Standalone System Interactive Diagnostics User's Guide

MOTOROLA

Standalone System

Interactive Diagnostics

User's Guide

(SSIDUG/D6)

Motorola, Inc.
Computer Group
2900 South Diablo Way
Tempe, Arizona 85282

PREFACE

The *Standalone System Interactive Diagnostics (SSID) User's Guide* describes the program that tests and diagnoses system problems and how to use it. This manual assumes that you are familiar with the operating system of your host system and with the operation of the devices you want to diagnose and/or test.

The *SSID User's Guide* contains seven chapters and two appendices:

- Chapter 1 provides an introduction to SSID.
- Chapter 2 describes SSID basics, including invoking and using SSID.
- Chapter 3 describes the SSID system commands available in menu mode.
- Chapter 4 focuses on confidence, fault, board, and peripheral testing; specific board tests are described in Chapters 5 through 7.
- Chapter 5 describes CPU, memory, and miscellaneous board tests, including MVME147, MVME167/MVME187, and M8120.
- Chapter 6 covers mass storage controller board tests.
- Chapter 7 describes communication controller board tests.
- Appendix A describes the command line mode commands.
- Appendix B provides wiring information for the loopbacks used in SSID testing.

Conventions Used in this Manual

The following notation conventions are used throughout this document:

bold
: is used for user input that you type just as it appears. **Bold** is also used for commands, options and arguments to commands, and names of programs, directories, and files.

italics
is used for names of *variables* to which you must assign values. *italics* is also used for comments in screen displays and examples.

`fixed font`
is used for system output (e.g., screen displays, reports), examples, and system prompts.

|
is used to separate two or more items and indicates that a choice is to be made; only one of the items separated by this symbol should be selected.

[ ]
enclose an optional symbol.

...
shows that you may repeat the previous argument.

0x or **$**
immediately preceding a number indicates it is a hexadecimal number.

RETURN
represents the single key that performs the return function. Other function keys are also boxed, e.g., BREAK .

<CR>
in screen displays, the RETURN function is indicated as <CR>.

^D
represents a control character and a letter. You execute control characters by pressing the CTRL key and the letter simultaneously.

NOTE

The screen displays shown in this document may vary slightly from your system's screen displays.

Much of the information contained in this manual is accessible online through the **help** system command. Refer to **help** in Chapter 3 for details.

Printed in the United States of America
August 1992

CONTENTS

APPENDICES

INDEX

FIGURES

TABLES

CHAPTER 1
INTRODUCTION

SSID Overview

SSID was derived from VMEsystest, a program that was first shipped with Motorola 1130 Series systems. VMEsystest proved to be an effective diagnostic and evaluation tool on these systems, thus was selected as the base for SSID diagnostics. SSID extends the functionality of VMEsystest by adding user-friendly menu structures with extensive predefined test scripts for factory, field, and development use.

SSID is a flexible, expandable set of system level tests and fault isolation diagnostics that execute on a VME-based system. The package is designed to serve the following four distinct end user groups:

1. *Manufacturing*: Verifies final system configuration by running the components in the actual target environment.
2. *Engineering*: Checks and verifies new designs in a particular system configuration.
3. *Field Service*: Measures performance and isolates faults in installed systems.
4. *System Integration*: The Extensible SSID (ESSID), which is an optional linkable version of SSID, offers a basis on which test modules for user-supplied products may be added to provide a complete system diagnostic. Refer to the *SYSTEM V/68 and SYSTEM V/88 Extensible SSID Driver Programmer's Guide.*

The many features and utilities of SSID are designed to meet the needs of each end user group. Fault isolation is the goal of SSID, and it is assumed that faults can be isolated to a functional block by using the test package. Component level fault isolation (i.e., testing specific integrated circuits or passive components such as resistors or capacitors) is generally beyond the scope of SSID.

SSID runs in a stand-alone environment; it does not interact with the operating system at any time. It can reside on a bootable floppy, tape cartridge, or within the operating system **root** file system. SSID is selected at system boot time in place of the operating system. Once booted, it has complete control over all hardware and peripherals. Remote communication capabilities are also available to allow off-site test and diagnosis via an optional internal modem.

Features

SSID possesses capabilities that set it apart from typical test software and enable it to serve a variety of applications. SSID features include:

- Tests that execute under a real-time, multitasking kernel to simulate the target environment in which the operating system would normally reside.
- Programs developed in C language to ensure ease of porting.
- The ability to configure test conditions, to match changing hardware configurations or to pinpoint a specific fault or measurement state.
- Elaborate test environment controls for fault isolation.
- Timer functions to support system performance measurement.
- Error logger and I/O redirection capabilities for test record keeping.

Concurrent Testing

Most test and diagnostic software packages use sequential testing, i.e., one test runs to completion before the next test starts. Sequential testing is easy to use and generally provides good fault coverage. In the VME environment, individual components such as CPU boards, communication boards, or disk controller boards are often tested in this way before being integrated into the final system configuration. At this point, you can execute another set of sequential tests to verify gross functionality.

General fault coverage may be acceptable; however, subtle interactions between modules that could appear in an operating system environment may not occur or be exercised at all. SSID provides an additional form of testing to simulate these interactions while still in a controlled environment.

Besides supporting sequential testing, SSID allows concurrent testing, which better simulates the asynchronous nature of the target systems. Concurrent operation in SSID is a blend of real-time execution (interrupt or exception processing), protected sequential execution (kernel operations that help synchronize and protect shared resources), and time-slice execution (each task is allotted a fixed portion of the available CPU time). Each test module or group of modules can execute in any or all modes, limitations being established only by the individual system components. After verifying system integrity with concurrent testing, the risk of additional failures while running with the operating system is significantly reduced or eliminated.

Dynamic Test Configuration

You can accomplish dynamic or run-time changes to SSID in three ways:

1. Specify the desired test order from the command line.
2. Use various SSID utilities to tailor the execution of tests. For example, enable or disable concurrent test mode or select tests to run once or in a loop-continuous mode. (Refer to **setopt** in Chapter 3 or **set** in Appendix A.)

NOTE

Changing options with **set/setopt** does not affect the options in **confid/fault** and peripheral test scripts.

3. Use the **cf** command line utility. (Refer to **cf** in Appendix A.)

I/O Support

SSID provides the end user with a variety of I/O support capabilities. Normal operator interface, terminal I/O, is the obvious start of the I/O system. You can add new or additional device drivers to the I/O system with the ESSID package. You can use I/O redirection to read or write the default or standard input, output, and error output functions to any device that has a device driver installed in SSID. A basic error logger function logs and maintains a summary count of errors occurring on each device so that errors that occur during unattended testing are not lost.

SSID Requirements

To run all SSID functions, the following items are required:

HARDWARE

- A Motorola microcomputer system acting as the host system.
- The host system terminal from which you invoked SSID, which is the current SSID control terminal.
- Write-enabled removable media.
- Loopback cables.
- Formatted disk drives (SSID versions dated April 1989 or later provide a diagnostic formatter when destructive testing is enabled for some drives).

SOFTWARE

- The operating system of the host system to format disk drives for drives that do not have a format program implemented in SSID.
- SSID as a bootable file on the operating system tape or disk, as a bootable floppy disk, or as a tape cartridge.

Table 1-1 provides a summary of SSID Tests and Diagnostics.

Table 1-1. Summary of SSID Tests and Diagnostics

Test Type	Approximate Time	Loop-Back Cable	Write-Enabled Medium
Confidence Test Suites			
Quick System Check with Communication	30 sec. - 3 min.	yes	yes
Quick System Check without Communication	30 sec. - 3 min.	--	yes
Continuous Running Check with Communication	user-determined	yes	yes
System Installation Check with Communication	30 min. - 1 hour	yes	yes
Continuous Running Intensive I/O	user-determined	yes	yes
Fault Test Suites			
OS System Panic	10 - 35 min.	--	yes
System Halt	10 - 35 min.	--	yes
Memory Fault	2 - 15 min.	--	--
Disk Fault	45 min. - 1 hr.	--	--
Tape Fault	1 hour	--	yes
Communication Fault	5 - 10 min.	yes	--
Board Tests			
MVME13X 68020 CPU	depends on	--	--
MVME135/136 68020 CPU	test(s) chosen	no	no
MVME141 68030 CPU	"	--	--
MVME143 68030 CPU	"	no	no
MVME147/147XX Monoboard Microcomputer	"	yes	some tests
MVME167 Monoboard Microcomputer	"	yes	some tests
MVME181 VMEmodule RISC CPU	"	no	no
MVME187 Monoboard Microcomputer	"	yes	some tests
MVME188 1P VMEmodule RISC CPU	"	no	no
MVME188 2P VMEmodule RISC CPU	"	no	no
MVME188 4P VMEmodule RISC CPU	"	no	no
M8120 system	"	yes	some tests

Table 1-1. Summary of SSID Tests and Diagnostics (cont'd)

Test Type	Approximate Time	Loop-Back Cable	Write-Enabled Medium
Board Tests (cont'd)			
Interactive Memory Tests	"	--	--
MVMEenv Environmental Monitor	"	--	--
MVME320 Wini/Floppy Controller	"	--	some tests
MVME323 ESDI Disk Drive Controller	"	--	(4)
MVME327 SCSI Drive Controller	"	--	some tests (4)
MVME328 SCSI Drive Controller	"	--	some tests (4)
MVME330 LAN Controller	"	(1)	--
MVME332/332XT 8-Port Communication Boards	"	yes	--
MVME333 WAN Controller	"	yes	--
MVME333 X.25 Controller	"	yes	--
MVME335 Async Communication Controller	"	yes	--
MVME336 DeltaLINK Async Communication Controller	"	yes	--
MVME337-1/337A I/O Engine with MVSB741 16 Channel Asynchronous Extensible VSB Module	"	yes	--
MVME350 Streaming Tape Controller	"	--	some tests
MVME355 9-Track Tape Controller	"	--	some tests
MVME360 SMD Hard Disk Controller	"	(2)	--
MVME374 Multiprotocol Ethernet Controller	"	(1)	--
MVME376 Ethernet Controller	5 sec.	--	--
MVME393 Multichannel Graphics Controller	"	(3)	--
MVME395 Graphics Controller	"	(3)	--
MVME050 Utility Board	"	--	--

Table 1-1. Summary of SSID Tests and Diagnostics (cont'd)

Test Type	Approximate Time	Loop-Back Cable	Write-Enabled Medium
Peripheral Test Suites			
VME320 Winchester Drive (2)	45 min. - 1 hr.	--	--
VME360 SMD Drive	35 min. - 1 hr.	--	--
VME320 Floppy Drive (DESTRUCTIVE)	15 min.	--	yes
VME350 1/4" Streaming Tape Drive (DESTRUCTIVE)	1 hr.	--	yes
VME355 9-Track Tape Drive (DESTRUCTIVE)	35 min. - 1 hr.	--	yes
VME323 ESDI Drive (2)	1 hr.	--	--
VME147/327 SCSI Tape Drive (DESTRUCTIVE)	1 hr.	--	yes
VME147/327 SCSI Disk Drive	25 - 45 min.	--	--
VME147 SCSI Floppy Drive (DESTRUCTIVE)	5 - 10 min.	--	--
VME327 Floppy Drive	5 - 10 min.	--	yes

(1) The board must be connected to Ethernet coax via transceiver/tap to run external loopback test.
(2) Format must include bad track list for software bad track support.
(3) The External Uart Test requires loopback cables.
(4) Destructive tests are available, but you must use them with care.

Related Documentation

It is recommended that you have the following manuals at hand for ready reference when using SSID:

- The *System Administrator's Guide* and *System Administrator's Reference Manual* for your host system's operating system.
- Manuals that specify the cable and port configuration for your host system.
- Manuals related to the devices that you want to diagnose and/or test.

The following publications may provide additional helpful information. If not shipped with this product, they may be purchased from the Motorola Technical Literature Center, 1919 W. Fairmont, Suite 8, Tempe, AZ 85282; telephone 1-800-458-6443.

Document Title	Motorola Publications Number
SYSTEM V/68 and SYSTEM V/88 Extensible SSID Driver Programmer's Guide	EXSSIDPG
MVME050 System Controller Module User's Manual	MVME050
MVME130/MVME131 32-Bit Microprocessor VMEmodule User's Manual	MVME130
MVME130XT and MVME130XT-1 Microcomputer User's Manual	MVME130XT
MVME131XT and MVME131XT-1 Microcomputer User's Manual	MVME131XT
MVME134 VMEmodule 32-Bit Monoboard Microcomputer User's Manual	MVME134
MVME135/135A, MVME135-1, and MVME136/136A 32-Bit Microcomputers User's Manual	MVME135
MVME141 32-Bit VMEbus/VSB-Based Microcomputer User's Manual	MVME141
MVME143 MPU VMEmodule User's Manual	MVME143
MVME147 MPU VMEmodule User's Manual	MVME147
MVME167 Single Board Computer User's Manual	MVME167
MVME167Bug Debugging Package User's Manual	MVME167BUG
MVMR167/MVME187 Single Board Computers SCSI Software User's Manual	MVME187FW
MVME167/MVME187 Single Board Computers Programmer's Reference Guide	MVME187PG
MVME167 Single Board Computer Installation Guide	MVME167IG
System Manual for Model 8120	SYS8120

Document Title	Motorola Publications Number
MVME147S MPU VMEmodule User's Manual	MVME147S
MVME147 SCSI Firmware User's Manual	MVME147FW
MVME181 181bug Debugging Package User's Manual	MVME181BUG
MVME181 VMEmodule RISC Microcomputer User's Manual	MVME181
MVME187 RISC Single Board Computer User's Manual	MVME187
MVME187Bug Debugging Package User's Manual	MVME187BUG
MVME187 RISC Single Board Computer Installation Guide	MVME187IG
MVME188 188Bug Debugging Package User's Manual	MVME188BUG
MVME188 VMEmodule RISC Microcomputer User's Manual	MVME188
HM88K HYPERmodule 32-Bit RISC Processor Mezzanine Module User's Manual	HM88KUM
MVME204-1/-2 Dual Ported Dynamic Memory VMEmodule User's Manual	MVME204
MVME204-2F Dual Ported Dynamic Memory VMEmodule User's Manual	MVME204F
MVME224-1/-2 4/8 Mb DRAM Memory Modules User's Manual	MVME224
MVME224A Series of DRAM Memory Modules User's Manual	MVME224A
MVME230 Series of Error Detection and Correction DRAM Memory Modules User's Manual	MVME230
MVME236 Series of DRAM Memory Modules User's Manual	MVME236
MVME246 Series of DRAM Memory Modules User's Manual	MVME246
MVME320B VMEbus Disk Controller Module User's Manual	MVME320B

Document Title	Motorola Publications Number
MVME327A VMEbus To SCSI Bus Adapter and MVME717 Transition Module User's Manual	MVME327A
MVME327A Firmware User's Manual	MVME327AFW
MVME330 Ethernet Controller User's Manual	MVME330
MVME332 Intelligent Communication Controller User's Manual	MVME332
MVME332XT Intelligent Communication Controller User's Manual	MVME332XT
MVME333-2 Intelligent Communication Controller User's Manual	MVME333
MVME335 Serial and Parallel I/O Module User's Manual	MVME335
MVME336 VME Hub Module and MVME751 Transition Module User's Manual	MVME336
SYS336M16 Terminal Server User's Manual	SYS336M16
MVME337 IO Engine User's Manual	MVME337
MVSB741 16 Channel Asynchronous Extensible VSB (EVSB) Module User's Manual	MVSB741
MVME350 Streaming Tape Controller VMEmodule User's Manual	MVME350
MVME360 SMD Disk Controller User's Manual	MVME360
MVME374 Multi-Protocol Ethernet Interface Module User's Manual	MVME374
MVME393 Multi-Channel Graphics Display Controller User's Manual	MVME393
MVME395 Graphics Display Controller User's Manual	MVME395

NOTE: Although not shown in the above list, each Motorola MCD manual publication number is suffixed with characters that represent the revision level of the document, such as /D2 (the second revision of a manual);

each supplement bears the same number as the manual but has a suffix, such as /A1 (the first supplement to the manual).

The following publications are available from the sources indicated:

VMERAM User Information Manual: Clearpoint Inc., 99 South Street, Hopkinton, MA 01748. (Provides information on the MVME205.)

V/ESDI 4201 Panther High-performance VMEbus Enhanced Small Device Interface (ESDI) Disk Controller User's Guide: Interphase Corporation, 13800 Senlac, Dallas, TX 75234. (Provides information on the MVME323.)

V/SCSI 4210 Jaguar High-performance VMEbus Dual SCSI Host Adapter User's Guide: Interphase Corporation, 13800 Senlac, Dallas, TX 75234. (Provides information on the MVME328.)

V/Tape 3209 High-performance VMEbus 1/2 Inch Tape Controller User's Guide: Interphase Corporation, 13800 Senlac, Dallas, TX 75234. (Provides information on the MVME355.)

Small Computer System Interface - 2 (SCSI-2) Specification as defined by American National Standard Institute (ANSI), number X3.131-198: Global Engineering Documents, 2805 McGaw, Irvine, CA 92714. (Provides information on the the SCSI interface as used by the MVME167/187, M8120, MVME327, and MVME328.)

V/Ethernet 3207 Hawk High-Performance VMEbus Ethernet Communications Controller User's Guide: Interphase Corporation, 13800 Senlac, Dallas, TX 75234. (Provides information on the MVME376.)

Intel 82596 User's Manual: Intel Corporation, Literature Sales, P.O. Box 58130, Santa Clara, CA 95052-8130. (Provides information on the LAN coprocessor built into the MVME167/187 and M8120.)

2

CHAPTER 2
SSID BASICS

This chapter explains how to invoke SSID and how to use its menu system for running tests and diagnostics.

Invoking SSID on M68000-based Products

Booting and Executing SSID (Except MVME136 or MVME167)

To boot and execute SSID on the MVME132, MVME134, MVME141, MVME143, and MVME147 perform, the following steps:

1. Reset or power off the system; then power on.
2. Type **h** (halt) immediately after the following message displays:

```
System Self Test    Rev. x.y
FPC status
PMMU status
```

If the second phase of System Self Test (SST) begins before you type **h**, press the **BREAK** key to stop SST.

2

3. Type **2** (`Select Alternate Boot Device`) in response to the following service menu:

```
1) Continue System Start-up
2) Select Alternate Boot Device
3) Go to System Debugger
4) Initiate Service Call
5) Display System Test Errors
6) Dump Memory to Tape
Enter menu #:  2 <CR> (User input shown in bold)
```

4. Choose the controller, drive, and file from which to boot SSID.

   `Enter Alternate Boot Device(Controller,Drive,File) :` *x,y,z* RETURN

   where:

   x is the controller to be accessed, *y* is the drive, and *z* is the file.

   Use Table 2-1 to determine non-SCSI boot device options and Table 2-2 to determine SCSI boot device addresses. MC68030-based processors allow mapping of logical controller numbers and physical controller numbers. To determine boot device addresses on these boards, it may be necessary to execute the bug command **iot;h** to determine the correct device address. Refer to the appropriate processor debugger manual for further information on the **iot** command.

Table 2-1. 680x0 Non-SCSI Boot Device Options

Controller	Drive	File
Winchester **0** = first MVME320 drive **1** = second MVME320	**0** = first Winchester **1** = second Winchester drive	**diag/test***PN**
Floppy **0** = first MVME320	**2** = first floppy drive **3** = second floppy drive	**test***PN**
Storage Module Device (SMD) **2** = first MVME360 **3** = second MVME360	**0** = first SMD drive **2** = second SMD drive	**diag/test***PN**
Streaming Tape **4** = first MVME350	**0** = streaming tape drive	**test***PN**
ESDI Disk Drive **8** = first MVME323	**0** = first ESDI drive **1** = second ESDI drive **2** = third ESDI drive **3** = fourth ESDI drive	**diag/test***PN**

* *PN* = Processor Number:
132 for MVME130-132-based systems, **134** for MVME134-based systems,
136 for MVME135-136-based systems, **141** for MVME141-based systems,
143 for MVME143-based systems.

For example, to boot SSID from the first floppy drive on an MVME132-based system, type:

0,2,test132 (RETURN)

where:

0 is the first MVME320 controller

2 is the first floppy drive (device #2)

test132 is the file to be loaded

At this point, SSID displays several messages, including the address (in hexadecimal) at which the Initial Program Load (IPL) was loaded and the amount of memory detected by SSID.

When booting SSID from an MVME147-based system with 147BUG 1.0 using SCSI devices attached to the onboard SCSI controller, use the following boot device options. The target addresses shown assume that the drives have been addressed according to instructions provided by Motorola in Motorola installation instructions. On MVME147-based systems with 147BUG 2.0 or higher, use the **iot;h** bug command to determine drive LUN numbers for booting. Once SSID is booted, use the SCSI physical address for testing.

Table 2-2. SCSI Boot Device Addresses with 147BUG 1.0 or 141/327ABUG

Controller (target)	Drive	Description	File
0	0	150Mb CDC WREN III	diag/test147
1	0	155Mb Micropolis 1375	diag/test147
2	0	300Mb CDC WREN IV/600Mb CDC WREN	diag/test147
3	0	85Mb Seagate ST296N/40Mb ST157N	diag/test147
4	0	Teac/Archive Tape	test147
5	0	Teac/Archive Tape	test147

5. When the service menu displays again, type **1** to continue the system self-test and startup sequence.

   Before the system boots from the alternate boot device, your controller, drive, and file selections display. SSID takes approximately two minutes to load.

Booting and Executing SSID on MVME136

To boot and execute SSID on the MVME136, proceed with the following steps:

NOTE

The MVME136 and MVME135 share the same firmware, so most of the displays show MVME135 instead of MVME136. An MVME135 can be tested with the MVME136 SSID with the following restriction: If SSID is to be booted from tape, there must be contiguous memory starting at address 0 and ending at a minimum of 4Mb (0x3FFFFF).

1. Reset or power off the system; then power on.
2. Press the BREAK key immediately after the following message displays:

```
VME135 Debugger Diagnostics Version x.y mm/dd/yy
FPC status
PMMU status
Local Memory Size is 4 MEG
Autoboot in progress... To abort hit BREAK
```

This stops the autoboot process and enters the debugger. The following displays:

```
Autoboot aborted ! ! !
135bug>
```

3. Choose the controller, drive, and file from which to boot SSID; use Table 2-1. For example, to boot SSID from the first ESDI drive on an MVME136-based system, type:

   135bug>**bo 0,8,test136** RETURN

   where:

0	is the first ESDI drive
8	is the first MVME323 controller
test136	is the file to be loaded

2

NOTE

For the MVME136, the controller and drive fields are reversed as compared to the MVME141.

At this point SSID displays several messages, including the address (in hexadecimal) at which the IPL was loaded and the amount of memory detected by SSID.

Booting and Executing SSID on MVME167

To boot and execute SSID on the MVME167, perform the following steps:

1. Reset or power off the system; then power on.
2. Type **h** (halt) immediately after a message similar to the following is displayed:

```
CopyrightMotorolaInc. 1988, 1989, 1990, 1991, AllRightsReserved

MVME167 Debugger/Diagnostics Release Version     x.y - date
COLD Start

Local Memory Found =00800000 (&8388608)

MPU Clock Speed =25Mhz
```

If the second phase of System Self Test (SST) begins before you type **h,** press the BREAK key to stop SST.

3. Type **2** (`Select Alternate Boot Device`) in response to the following service menu:

```
1) Continue System Start-up
2) Select Alternate Boot Device
3) Go to System Debugger
4) Initiate Service Call
5) Display System Test Errors
6) Dump Memory to Tape
Enter menu #:  2 <CR> (User input shown in bold)
```

4. Choose the controller (CLUN), drive (DLUN), and file from which to boot SSID.

```
Enter Alternate Boot Device: [RETURN]
Controller: x [RETURN]
Drive     : y [RETURN]
File      : z [RETURN]
```

where:

x is the controller to be accessed, *y* is the drive, and *z* is the file.

Use the following tables to determine the CLUN, DLUN, and FILE to be used to boot your system.

2

Table 2-3. MVME167 Boot Options

MVME327 - SCSI Controller			
Base Address		**CLUN**	
$FFFFA600		2	
$FFFFA700		3	
Drive	**Drive ID**	**DLUN**	**File Name**
SCSI Hard Disk	0	0	**diag/test167**
SCSI Hard Disk	1	10	**diag/test167**
SCSI Hard Disk	2	20	**diag/test167**
SCSI Hard Disk	3	30	**diag/test167**
SCSI Streaming Tape	4	40	**test167**
SCSI Streaming Tape	5	50	**test167**
SCSI Cassette Tape	6	60	**test167**
Local Floppy Disk	0	80	**test167**
Local Floppy Disk	1	81	**test167**

MVME328 - Single/Dual Channel SCSI Controller			
Base Address		**CLUN**	
$FFFF9000		6	
$FFFF9800		7	
Drive	**Drive ID**	**DLUN**	**File Name**
SCSI Hard Disk	Primary 0	0	**diag/test167**
SCSI Hard Disk	Primary 1	8	**diag/test167**
SCSI Hard Disk	Primary 2	10	**diag/test167**
SCSI Hard Disk	Primary 3	18	**diag/test167**
SCSI Sequential Access	Primary 4	20	**test167**
SCSI Sequential Access	Primary 5	28	**test167**
SCSI Teac Floppy Disk	Primary 6	30	**test167**
SCSI Hard Disk	Daughter 0	40*	**diag/test167**
SCSI Hard Disk	Daughter 1	48*	**diag/test167**
SCSI Hard Disk	Daughter 2	50*	**diag/test167**
SCSI Hard Disk	Daughter 3	58*	**diag/test167**
SCSI Sequential Access	Daughter 4	60*	**test167**
SCSI Sequential Access	Daughter 5	68*	**test167**
SCSI Teac Floppy Disk	Daughter 6	70*	**test167**

* This device is available only if the daughter card is installed.

2

Table 2-3. MVME167 Boot Options (cont'd)

MVME350 - Streaming Tape Controller			
Base Address		**CLUN**	
$FFFF5000 $FFFF5100		4 5	
Drive	**Drive ID**	**DLUN**	**File Name**
QIC-02 Streaming Tape	-	0	**test167**

MVME323 - ESDI Winchester Controller			
Base Address		**CLUN**	
$FFFFA000 $FFFFA200		8 9	
Drive	**Drive ID**	**DLUN**	**File Name**
ESDI Winchester Hard Disk	0	0	**diag/test167**
ESDI Winchester Hard Disk	1	1	**diag/test167**
ESDI Winchester Hard Disk	2	2	**diag/test167**
ESDI Winchester Hard Disk	3	3	**diag/test167**

MVME167 - Single Channel SCSI Controller			
Base Address		**CLUN**	
N/A		0	
Drive	**Drive ID**	**DLUN**	**File Name**
SCSI Hard Disk	0	0	**diag/test167**
SCSI Hard Disk	1	10	**diag/test167**
SCSI Hard Disk	2	20	**diag/test167**
SCSI Hard Disk	3	30	**diag/test167**
SCSI Sequential Access	4	40	**test167**
SCSI Sequential Access	5	50	**test167**
SCSI Teac Floppy Disk	6	60	**test167**

2

Invoking SSID on M88000-Based Products

To boot and execute SSID, follow these steps:

1. Reset or power off the system; then power on.
2. Press the (space) bar immediately after the first lines are displayed.
3. Type **3** (`Go to System Debugger`) in response to the following service menu:

```
1) Continue System Start-up
2) Select Alternate Boot Device
3) Go to System Debugger
4) Initiate Service Call
5) Display System Test Errors
6) Dump Memory to Tape
Enter menu #:  3 <CR> (User input shown in bold)
```

4. At the ROM debugger prompt, type the following command:

   bo *CLUN DLUN FILE* (RETURN)

   where:

   CLUN is the controller logical unit number

   DLUN is the drive logical unit number

   FILE is the file to be loaded

   Use the following tables to determine the CLUN, DLUN and FILE to be used to boot on your system.

2

Table 2-4. M88000 Boot Options

MVME320 - Winchester/Floppy Controller			
Base Address		**CLUN**	
$FFFFB000 $FFFFAC00		0 1	
Drive	**Drive ID**	**DLUN**	**File Name**
Winchester Hard Disk	0	0	**diag/test18x**
Winchester Hard Disk	1	1	**diag/test18x**
5-1/4 DS/DD 96 TPI Floppy	2	2	**test18x**
5-1/4 DS/DD 96 TPI Floppy	3	3	**test18x**

MVME327 - SCSI Controller			
Base Address		**CLUN**	
$FFFFA600 $FFFFA700		2 3	
Drive	**Drive ID**	**DLUN**	**File Name**
SCSI Hard Disk	0	0	**diag/test18x**
SCSI Hard Disk	1	10	**diag/test18x**
SCSI Hard Disk	2	20	**diag/test18x**
SCSI Hard Disk	3	30	**diag/test18x**
SCSI Streaming Tape	4	40	**test18x**
SCSI Streaming Tape	5	50	**test18x**
SCSI Cassette Tape	6	60	**test18x**
Local Floppy Disk	0	80	**test18x**
Local Floppy Disk	1	81	**test18x**

MVME350 - Streaming Tape Controller			
Base Address		**CLUN**	
$FFFF5000 $FFFF5100		4 5	
Drive	**Drive ID**	**DLUN**	**File Name**
QIC-02 Streaming Tape	-	0	**test18x**

2

Table 2-4. M88000 Boot Options (cont'd)

MVME328 - Single/Dual Channel SCSI Controller

Base Address	CLUN
$FFFF9000	6
$FFFF9800	7

Drive	Drive ID	DLUN	File Name
SCSI Hard Disk	Primary 0	0	diag/test18x
SCSI Hard Disk	Primary 1	8	diag/test18x
SCSI Hard Disk	Primary 2	10	diag/test18x
SCSI Hard Disk	Primary 3	18	diag/test18x
SCSI Sequential Access	Primary 4	20	test18x
SCSI Sequential Access	Primary 5	28	test18x
SCSI Teac Floppy Disk	Primary 6	30	test18x
SCSI Hard Disk	Daughter 0	40*	diag/test18x
SCSI Hard Disk	Daughter 1	48*	diag/test18x
SCSI Hard Disk	Daughter 2	50*	diag/test18x
SCSI Hard Disk	Daughter 3	58*	diag/test18x
SCSI Sequential Access	Daughter 4	60*	test18x
SCSI Sequential Access	Daughter 5	68*	test18x
SCSI Teac Floppy Disk	Daughter 6	70*	test18x

* This device is available only if the daughter card is installed.

MVME323 - ESDI Winchester Controller

Base Address	CLUN
$FFFFA000	8
$FFFFA200	9

Drive	Drive ID	DLUN	File Name
ESDI Winchester Hard Disk	0	0	diag/test18x
ESDI Winchester Hard Disk	1	1	diag/test18x
ESDI Winchester Hard Disk	2	2	diag/test18x
ESDI Winchester Hard Disk	3	3	diag/test18x

Table 2-4. M88000 Boot Options (cont'd)

MVME187/M8120 - Single Channel SCSI Controller			
Base Address		**CLUN**	
N/A		0	
Drive	**Drive ID**	**DLUN**	**File Name**
SCSI Hard Disk	0	0	**diag/test18x**
SCSI Hard Disk	1	10	**diag/test18x**
SCSI Hard Disk	2	20	**diag/test18x**
SCSI Hard Disk	3	30	**diag/test18x**
SCSI Sequential Access	4	40	**test18x**
SCSI Sequential Access	5	50	**test18x**
SCSI Teac Floppy Disk	6	60	**test18x**

For example, to boot SSID from the second ESDI drive attached to the first MVME323, type:

bo 8 1 diag/test18x (RETURN)

where:

8	is the first MVME323 controller
1	is the second ESDI drive
diag/test18x	is the file to be loaded from the disk

5. The executable SSID will now be loaded and execution begins. A screen identifying the version of SSID, the memory configuration, and board identification will display. Press (RETURN) to display the SSID Main Menu.

SSID Main Menu

Once SSID loads, an intermediate menu displays that identifies SSID and some information about the system.

2

M68000-based Initial Display

```
VME System Diagnostics - Version 6.2
Thu Jul 28  16:42:03  MST 1989

System has PMMU installed.

Total memory detected = 0x00400000, (4194303 bytes)

Seg          Start            End         Size
 0         0x00000000      0x003fffff  0x00400000

Depress Return to continue.
```

This display shows the SSID version number that has been booted plus its creation date and time. It also displays whether a Memory Management Board or Controller (MMB/C), Paged Memory Management Unit (PMMU), or no MMU was detected by SSID. The total memory detected by SSID is shown in hexadecimal and decimal. In addition, the start, end, and size of each contiguous block of memory displays.

Some systems include additional information in the display that indicate special items. For example, if an MVME147RF is the main processor board, the following message displays before the version message:

```
NO LANCE detected on this board
```

2

M88000-based Initial Display

```
VME System Diagnostic - Version n.n
Day Month Date Time Year
Software Product ID: DHn.n

     •
     Information specific to the processor board set displays here.
     •

Enabling parity checking of on-board Memory.

Depress Return to continue.
```

Items shown in italics in this display vary depending on the type and configuration of CPU board, amount of memory, and the version and date of the SSID release.

2

SSID Main Menu Display

Press (RETURN) to display the SSID Main Menu:

```
**********************************************************************
************* SYSTEM COMMANDS **************** MENU SELECTIONS *******
*    help        remote      disperr     **    confid      bdtest    *
*    /           cmdline     version     **    fault       ptest     *
*    disphrd     clear       setopt      **                          *
*    view        slctdev     gotobug     **                          *
**********************************************************************

Current Menu is / - "System monitor"

help          - Using Diagnostics                 (informational screens)
confid        - Confidence Testing                (test suites)
fault         - Test Selection by Fault           (test suites)
bdtest        - Test by Board                     (individual board tests)
ptest         - Peripheral Testing                (test suites)
ptesti        - Peripheral Testing continued      (test suites)
cmdline       - Command Line Operations           (operations for advanced
                                                   users)

SA:
```

Figure 2-1. SSID Main Menu

The command review block (the upper part of the menu display) provides a quick reference to the SSID system. It is divided into `System Commands` and `Menu Selections`. System commands are used to perform functions unrelated to testing. You can execute a system command any time a test is not in progress by typing its name. Table 2-5 gives a brief description of the system commands; they are explained further in the *System Commands* chapter.

NOTE

The command review block is not shown in the menu displays in this manual. When SSID is booted, default devices will be defined for each SCSI controller. If the system being tested does not contain the default configuration or if the user does not wish to test a device in the default configuration, **slctdev** must be executed to modify the SCSI configuration(s). The default configuration can be displayed and changed by executing **slctdev**. Refer to **slctdev** in the *System Commands* chapter.

Table 2-5. System Commands

Command	Description
help	Displays the help menu which gives access to online information similar to that contained in this manual.
/	Returns to the SSID Main Menu shown in Figure 2-1.
disphrd	Displays the boards whose presence is sensed by SSID.
remote	Allows dial-in to a remote terminal connected to a modem.
cmdline	Exits the menu mode to perform various tasks. **NOTE**: **cmdline** is intended for advanced users of SSID.
clear	Clears all *system* error logs and pass counters. A menu of *local* error logs will be displayed. The *local* error logs are cleared only if selected by the user.
disperr	Lists the accumulated *system* error summary and the pass count. A menu of *local* error logs will be displayed. The contents of the *local* error logs are displayed only if selected by the user.
version	Displays the current revision data for SSID.
setopt	Sets options for tests.
view	Displays the individual tests making up any confidence, fault, or peripheral suite.
slctdev	Allows the display and online reconfiguration of SCSI controllers and devices. **slctdev** is required only if you do not have the default SCSI devices and/or target addresses. To view the default SCCI configuration, execute the **slctdev** command.
gotobug	Allows the user to exit SSID and return to the BUG prompt (e.g., `147Bug>`). At the BUG prompt, enter **g** RETURN to return to SSID. You can execute any bug commands, except reset. If memory locations containing code are modified, it may be necessary to reboot SSID.

You use `Menu Selections` to access the menus that prompt you for information necessary to perform the tests and diagnostics. Table 2-6 describes the menu selections. The *Confidence, Fault, Board, and Peripheral Testing* chapter discusses the **confid, fault,** and **ptest** menu selections; the *CPU, Memory, and Misc. Controller Board Tests, Mass Storage Device Controller Board Tests,* and *Communications Controller Board Tests* chapters cover **bdtest** menu selections.

Table 2-6. Menu Selections

Selection	Description
confid	Test suites that verify total system health. They are useful at initial system installation and after a major system component replacement (refer to the *Confidence, Fault, Board, and Peripheral Testing* chapter).
fault	Test suites that test and diagnose specific system failures (refer to the *Confidence, Fault, Board, and Peripheral Testing* chapter).
bdtest	Individual tests that verify operation of specific boards in the system. Such testing is especially valuable whenever a board is replaced (refer to the *Confidence, Fault, Board, and Peripheral Testing* chapter).
ptest/ ptesti	Test suites that test and diagnose disks and tape devices attached to the system (refer to the *Confidence, Fault, Board, and Peripheral Testing* chapter).

Guidelines for Using SSID

The following guidelines will assist you in using SSID:

- Press RETURN after every keyboard selection.
- Correct typing errors by using ^h or a BACKSPACE .
- Stop a display from scrolling off the screen by using ^s . Resume scrolling with ^q . Some terminals have a HOLD SCREEN function key that stops and restarts scrolling.
- Before executing any test or test suite, read the **help** item or relevant section of this manual. Remember that some floppy disk and tape tests are destructive to the medium.
- Before executing any test suite, view the individual tests making up the suite by using the **view** command described in the *System Commands* chapter.
- Access any menu by entering its name and pressing RETURN .
- When you make a selection from a **confid, fault, ptest**, or **ptesti** menu, a suite of tests is performed; each **bdtest** menu selection executes only one test.
- SSID allows you to set various test options, such as verbose mode, concurrent mode, and display-all-errors mode. Refer to the **setopt** command in the *System Commands* chapter for details.

- If a test requiring a loopback cable is executed without the cable installed, the test or diagnostic may execute improperly. If this happens, install the cable and reboot SSID.
- If a test requiring a scratch floppy disk or tape is executed without the medium installed, the test or diagnostic may execute improperly. If this happens, install the scratch medium and reboot SSID.

Terminating SSID

When stopping an SSID test or the program itself, follow these guidelines:

- **Stopping a Test**
  Press the BREAK key to stop an SSID test. Any errors detected by the test or test suite before the test is stopped will not appear in the error log.

> CAUTION
>
> **Using the BREAK key sometimes causes SSID to become corrupted. If this happens, reboot SSID.**

- **Console Power-Off**
  Never power off or remove the console while executing SSID tests. If this happens, reboot SSID.
- **Stopping SSID**
  To exit the SSID program, power off or reset the system. No disk should be under test during power off; however, you may reset the system during a disk test.

CHAPTER 3
SYSTEM COMMANDS

3

System commands are used to perform functions unrelated to testing. You can execute a system command any time a test is not in progress. The system commands are listed below and described in detail in this chapter.

```
***********************************************************
***************** SYSTEM COMMANDS *********************
**         help          remote          disperr        **
**         /             cmdline         version        **
**         disphrd       clear           setopt         **
**         view          slctdev         gotobug        **
***********************************************************
```

help Command

This command causes the main **help** menu to display. Once in the **help** program, you can access online information about SSID operations and all SSID tests and test suites. This manual is based on the information in the **help** program.

To access the main **help** menu, type:

`SA:` **help**

The following menu displays:

3

```
  Main Help Menu
1) How To Use System Diagnostics
2) Confidence Testing ("confid")
3) Testing of Specific Faults ("fault")
4) Testing of Individual Boards ("bdtest")
5) Peripheral Testing ("ptest")
Enter TOPIC NUMBER or "6" to return to system test menu:
```

NOTE

The **help** menu can also be exited by one of the following characters: **s**, **S**, **q**, **Q**, **e**, **E**, or **/**.

The following describes each topic in the **help** menu:

1) `How To Use System Diagnostics`
   Items in this section provide general information about system commands and guidelines for using SSID menus.

2) `Confidence Testing ("confid")`
   Items in this section describe the confidence checks that verify total system health. The tests are useful at initial system installation and after a major system component replacement. Refer to the *Confidence, Fault, Board, and Peripheral Testing* chapter for details.

3) `Testing of Specific Faults ("fault")`
   Items in this section describe tests and diagnostics that deal with a specific system failure. For example, if you decide to run the OS System Panic test because you received an operating system panic message, the `OS Panic` item provides you with information about the test before you run it. Refer to the *Confidence, Fault, Board, and Peripheral Testing* chapter for details.

4) **`Testing of Individual Boards ("bdtest")`**
Items in this section describe tests which verify the operation of a specific board in the system. Such testing is valuable whenever you replace a board. Refer to the *CPU, Memory, and Misc. Controller Board Tests* chapter, *Mass Storage Device Controller Board Tests* chapter, and *Communication Controller Board Tests* chapter.

5) **`Peripheral Testing ("ptest")`**
Items in this section describe test suites and diagnostics available for disks and tape devices attached to the system. Refer to the chapter on *Confidence, Fault, Board, and Peripheral Testing* for details. Individual tests in the test suites can be run separately through the **bdtest** menu (refer to the *CPU, Memory, and Misc. Controller* chapter, *Mass Storage Device Controller Board Tests* chapter, and *Communications Controller Board Tests* chapter).

Information on a given topic may fill several screens. Press the **RETURN** key for additional help information; type **p** to return to a previous screen in a **help** selection; type **l** to return to the last menu.

Help Example

This example demonstrates how to obtain information on quick system tests.

3

```
SA: help <CR>
  Main Help Menu
1) How To Use System Diagnostics
2) Confidence Testing ("confid")
3) Testing of Specific Faults ("fault")
4) Testing of Individual Boards ("bdtest")
5) Peripheral Testing ("ptest")
Enter TOPIC NUMBER or "6" to return to system test menu: 2 <CR>
                         (to access the Confidence Testing help menu)
Confidence Testing ("confid")
 1) Quick System Tests   2) System Installation  3) Continuous Burn-in

Enter TOPIC NUMBER, or "4" to return to main help menu: 1 <CR>
                         (to access information about quick system tests)
                                  (continued)
```

3

(continued)

CONFIDENCE TESTS - Quick System Testing

The Quick System Tests verify that all major system components are healthly in 30 sec. to 3 min. depending on system memory and disk configuration.

The Quick System Tests come in two versions. One contains comm testing and requires a loopback cable on ports one and two of each 331/332 comm board, a loopback cable on ports three and four of each 333 comm board in the system. The other Quick System Test does not contain comm testing thus can be run without a loopback cable. This test requires a write enabled tape installed in the tape drive.

Depress CR to display menu, or "p" for previous info: <CR>
(to return to last menu.)

Confidence Testing ("confid")
 1) Quick System Tests 2) System Installation
 3) Continuous Burn-in

Enter TOPIC NUMBER, or "4" to return to main help menu:

3

/ Command

This command returns the SSID program to the **System Monitor** (home) menu. To access the home menu, type:

```
SA: /
```

The main SSID menu appears.

```
******************************************************************
********** SYSTEM COMMANDS ************ MENU SELECTIONS *******
*  help        remote      disperr    **     confid    bdtest    *
*  /           cmdline     version    **     fault     ptest     *
*  disphrd     clear       setopt     **                         *
*  view        slctdev                **                         *
******************************************************************

Current Menu is / - "System monitor"

help                      - Using Diagnostics
confid                    - Confidence Testing
fault                     - Test Selection by Fault
bdtest                    - Test by Board
ptest                     - Peripheral Testing
ptesti                    - Peripheral Testing continued
cmdline                   - Command Line Operations

SA:
```

disphrd Command

This command displays the boards whose presence is sensed by SSID when it probes to determine which VME boards are installed. (If a board is completely dead it may not be recognized.) The display is identical to the main **bdtest** menu.

To display the hardware in your system, type:

SA: **disphrd**

3

A display similar to the following appears:

```
Current Menu is /bdtest - "Tests by Board"

131.0     - VME131 68020 CPU Tests
204.0     - VME204 Computer System Memory Tests
320.0     - VME320 Disk Controller Tests
G330.0    - VME330 GSP LAN Board Test
332.1     - VME332 8 Port Comm. Board Test  (second MVME332 board)
332.0     - VME332 8 Port Comm. Board Test  (first MVME332 board)
333.0     - VME333 WAN Communications Board Test
350.0     - VME350 Streaming Tape Board Tests
355.0     - VME355 Controller/Drive Test
360.0     - VME360 SMD Controller Tests

SA:
```

remote Command

For systems with an internal modem, this command allows you to run diagnostics at a remote terminal connected to a modem. In addition, **remote** allows connection to a remote terminal on port 2 of the host CPU transition board. Once you are connected to the remote terminal, all keyboard activity and screen displays are seen concurrently. All of your keystrokes are echoed to the remote site and vice versa. Likewise, all SSID functions can be performed by you or by the individual at the remote terminal.

While connected to the remote terminal, you can choose a `Conversation Mode` which allows you to communicate via typed messages to the individual at the remote terminal. SSID ignores all activity done in `Conversation Mode`.

3

NOTE

The **remote** command does not currently support automatic dial-in to the Customer Response Center. If you have an internal modem and the Remote Maintenance Facility, you can access the Customer Response Center by selecting menu item 4 (`Initiate Service Call`) from the service menu shown in the *Invoking SSID* section.

To initiate the remote link, type:

`SA:` **remote**

The following question appears:

```
Is the remote a terminal or modem (t/m)?
```

If you request a remote terminal connection (**t**), the **remote** program connects you to port 2 of the host CPU transition board at 9600 baud. The terminal connection is complete when the previous SSID menu reappears. At this point all SSID functions you perform are visible on the remote terminal; likewise, the individual at the remote terminal can perform any SSID function.

If you request a remote modem connection (**m**), the following prompt appears:

`Is modem already connected (y/n)?`

Answer **y** (yes)	Answer **n** (no)
`Remote link up` *(The link is at 1200 baud.)* *(See format below.)* `Depress Return to Continue` *(You are now linked to the remote modem.)*	`Enter phone number:` `Remote link up` *(The link is at 1200 baud.)* `Depress Return to Continue` *(You are now linked to the remote modem.)*

When you are prompted to enter the telephone number, include the following symbols as needed:

T	tone dial (default)	**,**	wait two seconds
P	pulse dial (rotary)	=	pause and search for another dial tone

For example:

P0,,,6023214567

where **P** indicates a pulse dial telephone and ,,, requests a six-second delay after dialing the **0**. The delay gives the PBX time to connect your call to an outside line.

If there are problems connecting to the remote modem, you may receive one of the following messages.

Hanging Modem up
Retrying (**remote** *retries four times.*)
Enter phone number:

Remote link failed:
DIALING
NO ABT (*ABT = answer back tone*)

Remote link failed:
DIALING
BUSY

Remote link failed:
DIALING
NO DIAL TONE

To disconnect the remote link or enter `Conversation Mode`, type:

`SA:` **remote**

You are then prompted:

`Do you wish to disconnect the remote link (y/n)?`

Answer **y** (yes)	Answer **n** (no)
`Depress Return to Continue`	`Do you wish "Conversation Mode" (y/n)?` **y <CR>**
(Remote link disconnected.)	*(SSID ignores all input in "Conversation Mode".)*
	`Enter "<cr>" to exit Conversation Mode` *(Press the Return key followed by a period (.) when ready to exit.)*

cmdline Command

This command causes SSID to exit the menu mode and enter another testing environment, the command line mode. (Refer to Appendix A for a discussion of the command line mode.) Once in **cmdline** mode, enter **set menu** to return to menu mode.

clear Command

This command clears all system error logs and pass counters and gives the user the option of clearing local error logs. Once a test fails, the `SYSTEM TEST FAIL` message displays. You must use the **clear** command to continue testing (unless you have enabled the `continue-on-error` test option as described in the *setopt Command*).

The error logging capabilities of some boards/devices have been enhanced by the addition of what is referred to as a *local error log*. The local error log allows information pertinent to a failure to be saved for viewing by the SSID user after the information has scrolled off the screen. These local error logs are not automatically cleared by execution of the **clear** command. A menu of the local error logs that have been initialized will be displayed and the user will be given the option of clearing any or all of these logs. If a test has an error logging feature implemented that does not conform to the local error log specifications, that test will *not* appear on the menu displayed by this function, although there may be a display/clear error log item on the test menu.

To clear the system error logs and pass counters and view the menu of local error logs, type:

SA: **clear**

```
*****               SSID ERROR LOG MENU            *****
 1 ) 376.0/lan376            2 ) env.0/envmon
     NOTE:  The individual device error logs will be out of
            sync with the error count summary if all of the
            logs are not cleared.
Enter 'a' to clear all error logs, the number from list above
to clear only one error log or 'q' to quit:
```

At this point, the system error logs and pass counters have been cleared and the user has the option of clearing any or all of the local error logs. To clear all of the local error logs, type **a**. To clear a specific error log, enter the menu selection number associated with that log. The SSID Error Log Menu will be redisplayed after each selection made by the user until a **q** is typed to quit the **clear** command.

3

```
*****                  SSID ERROR LOG MENU                *****
 1 ) 376.0/lan376              2 ) env.0/envmon
     NOTE:  The individual device error logs will be out of
            sync with the error count summary if all of the
            logs are not cleared.
Enter 'a' to clear all error logs, the number from list above
to clear only one error log or 'q' to quit:  q
Depress Return to Continue (RETURN)
```

If the user does not clear all of the local error logs during execution of the **clear** command, the local error logs will be out of sync with the system error log summary count. The **disperr** command system summary count would indicate that no errors had occurred but the information regarding any errors that occurred before execution of the **clear** command would still be available for viewing by the user. This feature allows clearing of the system error counts so that testing can continue and the `SYSTEM TEST FAIL` message will not be displayed but the error information will still be available to the user for errors that occurred on previous passes of SSID.

Notice that the **clear** command does not issue a message stating that the logs and counters have been cleared. Use **disperr** before and after the **clear** command to see its effect. Refer to the example in the *disperr Command* section.

disperr Command

This command lists the accumulated system error summary and the pass count since the last execution of the **clear** command and gives the user the option of displaying *local* error logs. Refer to the discussion of local error logs in the *clear Command* section. If a failure has occurred on a device in which the *local* error logger has not been implemented or the error logging does not conform to the error logger specifications, the summary error count will be displayed but there will be no selection for the device in the SSID Error Log Menu.

To display all errors, type:

SA: **disperr**

```
Pass-Count = 0
  /bdtest/env.0/envmon                :    error=0, fatal=1
  /bdtest/376.0/lan376                :    error=0, fatal=2
*****                SSID ERROR LOG MENU                *****
 1 ) 376.0/lan376               2 ) env.0/envmon
Enter 'a' to display all error logs, the number from list
above to display only one error log or 'q' to quit:
```

At this point, the user has the option of displaying any or all of the local error logs. To display all of the local error logs, type **a**. To display a specific error log, enter the menu selection number associated with that log. The SSID Error Log Menu will be redisplayed after each selection made by the user until a **q** is typed to quit the **disperr** command.

If the user chooses to display all of the local error logs, a display similar to the following will be output to the screen with the user prompted to enter a carriage return at appropriate intervals for paging the error information.

3

```
*****         ERROR LOG ENTRIES FOR 376.0/lan376           *****

 .PASS..TIME-PASS....TIME-BOOT...DEVICE-ID.....FAILING COMMAND
     0  0000:00:00    0000:00:26  376.0/lan376    Memory test
  ERROR: 0x00000101, Data Read does not match Data Written

Depress RETURN or ENTER key to continue (RETURN)

*****         ERROR LOG ENTRIES FOR env.0/envmon           *****

 .PASS..TIME-PASS....TIME-BOOT...DEVICE-ID.......FAILING COMMAND
     0  0000:00:00    0000:00:39  env.0/envmon    Verify Connect.
  ERROR: 0xfffffffc, The transition board is NOT properly conn.

     STATA REGISTER BITS:  0x8000
       Transition board is NOT connected.
     STATB REGISTER BITS:  0x0065
       UPS on port 2.
       UPS on port 3.
       External Chassis on port 3.
       External Chassis on port 1.
     COMMAND REGISTER BITS:  0x0048
       Interrupts enabled from Env. Monitor.
       Short overtemp timeout selected.
     INTERRUPT VECTOR NUMBER: 0x0080

Depress RETURN or ENTER key to continue (RETURN)

*****               SSID ERROR LOG MENU               *****

 1 ) 376.0/lan376             2 ) env.0/envmon

Enter 'a' to display all error logs, the number from list
above to display only one error log or 'q' to quit:  q

Depress Return to Continue (RETURN)
```

In this example, the first error shows the common information that will be logged and displayed for each failure. The second error shows how device-specific information might be displayed if it is available. For some devices, the register values shown above might be replaced with the expected and actual data in a data miscompare, SCSI sense data, or other such information relative to the failure that occurred. Further description of the error information related to a specific device or board can be found in the Board Tests description for that particular board/device.

The **PASS** field displays the pass number in which the error occurred.

The **TIME-PASS** field displays the time at which the error occurred as counted from the beginning of the test/suite pass.

The **TIME-BOOT** field displays the time at which the error occurred as counted from the time SSID was booted.

The **DEVICE-ID** field specifies the device on which the error occurred.

The **FAILING COMMAND** field describes the command being executed when the error occurred. For some devices, this may be the SSID test that was being executed. For other devices, it may indicate the actual operation that failed, e.g., a 147 SCSI attach error. This description is found by searching a table that relates the command code to a textual description. If a description of the failing command cannot be found, `Command unknown.` will be displayed in this field. If there is no command code translation table available, `Command unspecified.` will be displayed in this field.

The **ERROR** field gives the hexadecimal value of the error code returned by the device, followed by a textual description of the error code. This description is found by searching a table that relates the error code to a textual description. If a description of the error code cannot be found, `Unknown error, no definition found in error table.` will be displayed in this field. If there is no error code translation table available, `Unknown error, no error table specified.` will be displayed in this field.

version Command

This command displays the current revision and date/time of creation for SSID as shown in the following example:

```
SA: version <CR>
Version 6.1   Jan 19  13:22:17 MST 1989
Depress Return to Continue <CR>
```

3

setopt Command

This command allows you to enable or disable the following test options:

NOTE

setopt only affects menu level tests. The **set** command must be used for **cmdline** mode.

Options	Description
lc	loop-continue
le	loop-on-error
li	loop-on-intermittent-error
vb	verbose mode
ce	continue-on-error
ae	display all errors
dp	display packet
con	concurrent mode
time	execution time
menu	prompt by menu

The following describes the variable settings:

lc loop-continue (default: disabled)
When enabled, this variable causes SSID to repeat a test or test suite continuously until the **BREAK** key is pressed.

le loop-on-error (default: disabled)
When enabled, this variable causes SSID to repeat a failing test until the test passes. It is useful if you want to see an error occur repeatedly.

li loop-on-intermittent-error (default: disabled)
When enabled, this variable causes SSID to repeat a test that fails randomly.

vb verbose mode (default: enabled)
When enabled, this variable determines how much status and error information displays while tests are running. When disabled, the information about individual tests is suppressed; only the execution time and test outcome (passed or failed) displays.

ce continue-on-error (default: disabled)
When enabled, this variable causes a test to continue to run even when an error has been detected. When **ce** is disabled, SSID stops after the current test completes if an error has occurred. Enable **ce** if you want to see all possible errors in a given test suite or sequence.

ae display-all-errors (default: disabled)
When enabled, this variable permits the display of all errors that result from data comparisons during a test. If you are interested only in whether a test passes or fails, disable **ae** so that only the first data comparison error message displays.

dp display packets (default: disabled)
When enabled, this variable displays packet activity as it occurs. When disabled, no packets are displayed unless an error occurs. On the MVME147, this option is used to display SCSI packets if an error occurs during execution of the packet. (A packet is a block of data containing commands and control information and will vary in content between controllers (refer to the appropriate board information in the following chapters.) The figure below shows a packet as it might be displayed if **dp** were enabled.

```
|cmd -p| stat _p| pkt_no |cmd |type|am|dw|pu|su|pri_addr|sec_addr|
| 01   |   ff   |00000005|2001| 02 |05|01|00|00|00000000|0000ffff|
| count  |ctrl0|ctrl1|ctrl2|comp_st  |ss|ls|rcnt|pri_sadr|sec_sadr|
|00000096|ffff |ffff |ffff |c20000000|00|00|0000|0000a4fe|00000000|
|term_cnt|stat0|stat1|stat2|
|00000000|0000 |4031 |0000 |
```

con concurrent mode (default: enabled)
When enabled, this variable allows multiple tests to be run on the current command line concurrently or in parallel. Disable **con** if you want to run a group of tests on the command line sequentially.

time display elapsed time (default: disabled)
When enabled, this variable causes the elapsed time of the test or test suite to display in the following format:

`Execution time (hrs:min:sec.ms)` = *hrs:min:sec.ms*

This information is not shown if **time** is disabled.

menu prompt by menu (default: enabled)
This option causes SSID to return to the previous menu screen after the execution of any command, test, or test suite. Switching to command line mode automatically disables this option.

3

NOTE

Most **confid**, **fault**, **ptest**, and **ptesti** test suite options are set in the test scripts themselves and are, therefore, unaffected by option settings made using **setopt**.

To change any of the preceding test options, type:

SA: **setopt** RETURN

A display similar to the following appears.

3

```
SA: setopt
Setopt
Current option settings are:      (This display shows the default settings.)
1)              lc - disabled loop-continue
2)              le - disabled loop-on-error
3)              li - disabled loop-on-intermittent-error
4)              vb - enabled verbose-mode
5)              ce - disabled continue-on-error
6)              ae - disabled display all errors
7)              dp - disabled display packet
8)             con - enabled concurrent-mode
9)            time - disabled execution-time
10)           menu - enabled prompt by menu
11)                  set all options to default setting
12)                  to exit setopt
             (This example shows how to disable the verbose mode option.)
Enter the number of the option to be changed
: 4 <CR>
vb - enabled verbose-mode
Enter: 1 to enable or 2 to disable
2 <CR>
             (The setopt display screen reappears with verbose mode disabled.)
Enter the number of the option to be changed
: 12 <CR> (to exit setopt.)
Depress Return to Continue <CR>
```

view Command

The **view** command displays the individual board tests making up any confidence, fault, or peripheral test suite. Note that the tests are not run with **view**; it is a display only command.

To display any confidence suite, type:

`SA:` **view confid**

3

The confidence menu displays. Enter the number of the suite you wish to view. If you select `Confidence Test 1`, a display similar to the following appears:

```
                Quick System Confidence Check with Comm.:

set +vb -ce +time;mem.0 2;131.0;320.0/hd.0 01;332.0/sp.0;350.0/tp.0
c0120c0120:

Do you wish to select another confid script for viewing?
y - reselect,  n - exit view
```

The suite displays in the same format used to specify multiple board tests. Refer to *Performing Multiple Board Tests* in *Confidence, Fault, Board, and Peripheral Testing* chapter. Testing environment options set with the **set/setopt** command do not affect options in confid/fault and peripheral tests. Refer to Appendix A.

If you want to view another confidence suite, type **y** for reselect. The confidence screen redisplays, and you can enter the number of the desired suite.

To exit the view confidence suites, type **n**; the menu present when **view confid** was entered redisplays. After exiting **view**, selecting suites causes them to run instead of display.

You may use **view** in a similar manner for fault (**view fault**) and peripheral (**view ptest, view ptesti**) suites.

NOTE

view is not available for board tests.

slctdev Command

This command displays the current SCSI configuration for all configurable controllers, and allows you to select different devices for each target address.

To display and/or change the SCSI configuration, type:

`SA:` **slctdev**

A display similar to the following appears:

```
Select from the following:

q)  Quit slctdev program
1)  147.0
2)  327.0
3)  327.1

Enter Selection:
```

Enter the menu number of the controller for the SCSI devices to be configured and press RETURN .

NOTE

If the number is out of range, **slctdev** defaults to the first selection.

The current SCSI configuration for the selected controller displays.

3

```
CURRENT DEVICE CONFIGURATION
Current configuration for target[0]: System V/68 CDC III 150 MB SCSI
Current configuration for target[1]: NONE
Current configuration for target[2]: NONE
Current configuration for target[3]: NONE
Current configuration for target[4]: Archive Streaming Tape Drive
Current configuration for target[5]: NONE
Current configuration for target[6]: NONE

DEPRESS RETURN TO CONTINUE
```

NOTE

You can have **slctdev** exit from either of the `Current Device Configuration` displays by entering **q**.

Press (RETURN) to display the first target menu.

3

```
                              SYSTEM COMMANDS

Select a controller for target 0 from the following list:

- 512 byte sectors:  SYSTEM V/68 and V/88 hard disks and DSHD floppies
- 256 byte sectors:  VERSAdos hard disks and SYSTEM V/68 and V/88 DSDD floppies

 1 * CDC III (150MB) 512 byte sectors        2 - Mic1375 (150MB) 512 byte sectors
 3 - CDC IV (300MB) 512 byte sectors         4 - CDC V (600MB) 512 byte sectors
 5 - Seagate (40MB) 512 byte sectors         6 - Seagate (85MB) 512 byte sectors
 7 - Swift (104MB) 512 byte sectors          8 - Swift (172MB) 512 byte sectors
 9 - CDC III (150MB) 256 byte sectors       10 - CDC IV (300MB) 256 byte sectors
11 - Seagate (40MB) 256 byte sectors        12 - Seagate (85MB) 256 byte sectors
13 - 655KB 5.25" Floppy 256 byte sectors    14 - 1.2MB 5.25" Floppy 512 byte sectors
15 - 750KB 3.5" Floppy 512 byte sectors     16 - 1.5MB 3.5" Floppy 512 byte sectors
17 - 2.9MB 3.5" Floppy 512 byte sectors     18 - 1.2MB 5.25" Floppy 256 byte sectors
19 - ARCHIVE or TEAC Streamer               20 - KENNEDY 9 Track Tape 96X0
21 - EXABYTE 2GB Tape

r - no controller                      n - go to next controller
b - start over at controller 0         q - quit selection program
```

The display lists all possible devices for target address 0. The entry number followed by an asterisk (*) specifies the currently selected device for the displayed target address. On an MVME328, the list is simpler: the exact device need not be specified, only the device type (i.e., hard disk, tape, or floppy disk). To select a different device for the displayed target address, type the number associated with the desired device and press (RETURN) . The selections for the next target address then displays. If you do not want to change the device for the target address you are currently viewing, type **n** or (RETURN) to go to the next target address.

To remove a configured device from a SCSI target address enter **r**. Enter **b** to restart the configuration for this controller at SCSI target 0. Enter **q** to exit configuration for this controller.

3

After displaying the device options for all target addresses or typing **q** (quit), the CURRENT DEVICE CONFIGURATION displays again, showing the changes made with **slctdev**; press RETURN to redisplay the **slctdev** menu. Select another controller to configure or enter **q** to exit the **slctdev** program. If **slctdev** was invoked on a system without configurable SCSI devices, the following message displays:

```
There is no selectable SCSI in this system.
```

All SSID tests now reflect the changes made with **slctdev**. Note, however, that the **ptest** scripts work only if target addresses 0, 1, 2, 3, and 6 are disk drives, and target addresses 4 and 5 are tape drives. **ptest** scripts are not configurable.

There is a short cut method for direct selection of a target. This short cut is available at the first display of `Current Device Configuration`. If, at the `Depress Return To Continue` prompt, the letter **s** is entered, the user is prompted for the target number for which the device selection information will be displayed. This menu will allow the user to work on just the target(s) in which the user is interested.

gotobug Command

The **gotobug** Command returns the user to the BUG prompt (e.g., `147BUG>` or `188-Bug>`) to execute bug commands. As long as a reset has not been executed, you can return to SSID at the point where **gotobug** was executed by entering **g** RETURN . If memory locations containing code were modified, it may be necessary to reboot SSID.

CHAPTER 4
CONFIDENCE, FAULT, BOARD, AND PERIPHERAL TESTING

Confidence Testing

Confidence testing verifies total system health by running a dynamically configured suite of tests. Before performing a confidence test, read this chapter or read the **help** screens related to confidence testing. Both sources describe the various confidence test suites available through the **confid** menu. You can display the individual tests making up the suites with the **view** command.

To access the **confid** menu, type:

SA: **confid**

The following display appears:

```
Current Menu is /confid - "Confidence Testing"

1                 - Quick System Confidence Check with Comm.
2                 - Quick System Confidence Check without Comm.
3                 - Continuous Running Confidence Check with Comm.
4                 - System Installation Confidence Check with Comm.
5                 - Continuous Running Intensive I/O Check with Comm.
```

(Tests 1 through 5 require write-enabled tapes installed in all tape drives attached to the system.)
(Tests 1, 3, 4, and 5 require loopback cables.)

SA:

Quick System Confidence Checks

The Quick System Confidence Checks (menu selections 1 and 2) verify that all major system components are healthly. The tests take from 30 seconds to 5 minutes depending on system memory and disk configuration.

There are two types of Quick System Confidence Checks. The first type tests the system with communication boards connected. To run this test suite, loopback cables are required on ports 3 and 4 of the MVME147 monoboard computer; on ports 1 and 2 of each MVME331/MVME332 or MVME332XT communication board; on selected ports of each MVME333 WAN or X.25 communication board; on all ports of each MVME335 communication board; on all 16 ports of the first cluster controller attached to each MVME336 board; and on all port pairs of each octopus cable on each MVSB741 asynchronous communications controller. Refer to the *MVME333 WAN Controller and MVME333 X.25 Controller* section in the *Communication Controller Board Tests* chapter for a description of the MVME333 loopback cable configuration.

Connect the loopback cables as follows:

```
         MVME147 :  SP3 <------> SP4
 MVME331/MVME332 :  SP1 <------> SP2
       MVME332XT :  SP1 <------> SP2
         MVME335 :  SP1 <------> SP2
                    SP3 <------> SP4
         MVME336 :  Individual Port Loopbacks
 MVME337/MVSB741 :  (each octopus cable)
                    Channel 0 <------> Channel 4
                    Channel 1 <------> Channel 5
                    Channel 2 <------> Channel 6
                    Channel 3 <------> Channel 7
```

Refer to the *MVME333 WAN Controller and MVME333 X.25 Controller* section of the *Communication Controller Board Tests* chapter for a description of the MVME333 loopback cable pinouts.

See Figure 5-1 in the *CPU, Memory, and Misc. Controller Board Tests* chapter for proper cabling of the 9-pin loopback cable used for the MVME147.

Figure 4-1 in this chapter shows the proper pinout for the 25-pin loopback cables used for the MVME331/MVME332/MVME332XT/MVME333/MVME335 and MVME337/MVSB741.

Figure 7-3 in the *Communication Controller Board Tests* chapter shows the proper pinout for MVME336 loopback cables.

The second type of Quick System Test does not include communication testing and can thus be run without loopback cables.

Both Quick System Confidence Checks require write-enabled tapes installed in all tape drives attached to the system.

```
Male DB-25                                              Male DB-25
 connector                                               connector
   pin #        (cable length 3 inches)                   pin #
-------------------------------------------------------------------
(Tx)     2     <------------------------->    3    (Rx)
(Rx)     3     <------------------------->    2    (Tx)
(RTS)    4     <------------------------->    5    (CTS)
(CTS)    5     <------------------------->    4    (RTS)
(DCD)    8     <------------------------->   20    (DTR)
(DTR)   20     <------------------------->    8    (DCD)
```

Figure 4-1. 25 Pin Loopback Cable Pinout

Continuous Running Confidence Check and Continuous Running Intensive I/O Check

This confidence check (also known as Continuous Burn-In) allows you to observe what happens to the system after many hours of continuous use. It is designed to rerun a specific set of tests until the BREAK key is pressed. Loopback cables are required on ports 3 and 4 of the MVME147 monoboard computer; on all port pairs of all MVME331, MVME332, MVME332XT; on selected ports of each MVME333 WAN or X.25 communication board; on all ports of each MVME335 communication board; on all 16 ports of the first cluster controller attached to each MVME336 board; and on all port pairs of each octopus cable on each MVSB741 asynchronous communications controller. Refer to the *MVME333 WAN Controller and MVME333 X.25 Controller* section in the *Communication Controller Board Tests* chapter for a description of the MVME333 loopback cable configuration.

Connect the loopback cables as follows:

MVME147:	SP3 <------> SP4
MVME331/MVME332/MVME332XT :	SP7 <------> SP8
	SP5 <------> SP6
	SP3 <------> SP4
	SP1 <------> SP2
MVME335:	SP1 <------> SP2
	SP3 <------> SP4
MVME336:	Individual Port Loopbacks
MVME337/MVSB741:	(each octopus cable)
	Channel 0 <------> Channel 4
	Channel 1 <------> Channel 5
	Channel 2 <------> Channel 6
	Channel 3 <------> Channel 7

Refer to the *MVME333 WAN Controller and MVME333 X.25 Controller* section of the *Communication Controller Board Tests* chapter for a description of the MVME333 loopback cable pinouts.

See Figure 5-1 in *CPU, Memory, and Misc. Controller Board Tests* chapter for proper cabling of the 9-pin loopback cable used for the MVME147.

Figure 4-1 in this chapter shows the proper pinout for the 25-pin loopback cables used for the MVME331/MVME332/MVME332XT/MVME333/MVME335 and MVME337/MVSB741.

Figure 7-3 in the *Communication Controller Board Tests* chapter shows the proper pinout for the MVME336 loopback cables.

System Installation Confidence Check

You can run this confidence test after system installation to verify the total System, including communication. The test takes from 30 minutes to longer than one hour depending on system memory and disk configuration. Loopback cables are required on ports 3 and 4 of the MVME147 monoboard computer; on ports 1 and 2 of each MVME331/MVME332/MVME332XT communication board, on selected ports of each MVME333 WAN or X.25 communications board, on all ports of each MVME335 communications board, on all 16 ports of the first cluster controller attached to each MVME336 board, and on all port pairs of each octopus cable on each MVSB741 asynchronous communications controller. Refer to the *MVME333 WAN Controller and MVME333 X.25 Controller* section in the *Communication Controller Board Tests* chapter for a description of the MVME333 loopback cable configuration.

MVME147 :	`SP3 <------> SP4`
MVME331/MVME332 :	`SP1 <------> SP2`
MVME332XT :	`SP1 <------> SP2`
MVME335 :	`SP1 <------> SP2` `SP3 <------> SP4`
MVME336 :	`Individual Port Loopbacks`

MVME337/MVSB741 :	`(each octopus cable)`
	`Channel 0 <------> Channel 4`
	`Channel 1 <------> Channel 5`
	`Channel 2 <------> Channel 6`
	`Channel 3 <------> Channel 7`

Refer to the *MVME333 WAN Controller and MVME333 X.25 Controller* section of the *Communication Controller Board Tests* chapter for a description of the MVME333 loopback cable pinouts.

See Figure 5-1 in the *CPU, Memory, and Misc. Controller Board Tests* chapter for proper cabling of the 9-pin loopback cable used for the MVME147.

Figure 4-1 in this chapter shows the proper pinout for the 25-pin loopback cables used for the MVME331/MVME332/MVME332XT/MVME333/MVME335 and MVME337/MVSB741.

Figure 7-3 in the *Communication Controller Board Tests* chapter shows the proper pinout for the MVME336 loopback cables.

Fault Testing

Fault testing checks and diagnoses specific system failures by running a dynamically configured suite of tests. Before performing a fault test, read this chapter or read the **help** screens related to fault testing. Both sources describe the various fault test suites available through the **fault** menu. The individual tests making up the suites may be displayed with the **view** command.

The **fault** menu contains test suites used to diagnose specific system failures. These suites test the requested category intensively and check the interaction between other related components in the system. For example, if the operating system reports a memory fault, run the memory fault test. This test intensively tests system memory and checks the interaction with disk and CPU operations.

To access the **fault** menu, type:

`SA:` **fault**

The following display appears:

```
Current Menu is /fault - "Test Selection by Fault"
1              - Test to be run for OS System Panic
2              - Test to be run for System Halt
3              - Memory Fault Test
4              - Disk Fault Test
5              - Tape Fault Test
6              - Communication Fault Test
```

(Tests 1 *and* 5 *require write-enabled tapes installed in all tape drives attached to the system.)*

```
SA:
```

OS System Panic

Run this test suite if an operating system panic is reported. The test suite performs an intensive memory test with short disk, tape, and CPU tests, ensuring complete system test. You must install write-enabled media in all tape drives attached to the system. In a minimal system configuration the tests take approximately 35 minutes.

NOTE

No communication board or communication line testing is performed.

The following display shows a sample OS System Panic test sequence that is initiated by typing the following in response to the **fault** menu shown above.

SA: 1 (RETURN)

```
/bdtest/131.0       1)interrupter          .................starts
/bdtest/131.0       1)interrupter          .................passed
      .                 .                                    .
      .                 .                                    .
      .                 .                                    .
      .                 .                                    .
/bdtest/204.0/mem.0 9)Long AA/55 Pattern .................starts
/bdtest/204.0/mem.0 9)Long AA/55 Pattern .................passed

PASS-0, FATAL-000, ERROR-000, 0:02:50.0
/bdtest/204.0/mem.0 : running

Execution time (hrs:min:sec.ms) - 0:30:04.180
*** SYSTEM TEST PASSED ***
Depress Return to Continue <CR>
```
(Press <CR> to return to the main **fault** *menu.)*

System Halt

Run this test suite if the system-fail LED (light emitting diode) on the CPU board is lit, or if the system stopped with no error indication.

The test suite performs intensive memory tests with short disk, tape, and CPU tests, ensuring complete system test coverage. You must install write-enabled tapes in all tape drives attached to the system. In a minimal system configuration the tests take approximately 35 minutes.

NOTE

No communication board or communication line testing is performed.

Memory Fault Test

Run the memory fault test if an operating system panic message about memory appears. The test suite determines if there is a problem with the RAM chips, VMEbus, VSBbus, ECC logic, or parity logic.

NOTE

You must perform all SST tests to correctly test *all* memory.

Disk Fault Test

Run this test suite if any type of disk fault is reported. The tests ensure complete system test coverage by performing intensive disk testing on the first disk drive of every disk controller in the system, as well as short memory and CPU testing. In a minimal system configuration, the tests take approximately one hour.

Tape Fault Test

Run this test suite if any type of tape fault is reported. The tests ensure complete system test coverage by performing intensive tape testing on the first tape drive on every tape controller in the system, as well as short disk, memory and CPU testing. You must install write-enabled tapes in all tape drives attached to the system. In a minimal system configuration, the tests take about one hour.

Communication Fault Test

Run this test suite if any type of communication fault is reported. The test suite automatically tests all properly configured communication boards in the system.

The test suite performs the following tests on the first port pair (1&2) on each MVME332/MVME331 communication board:

```
0)  ext loop back
1)  single char loop back
2)  break detection
3)  abort read
4)  abort write/read
5)  echo
6)  full duplex
7)  overrun
8)  parity error
9)  hw flow control
```

Loopback cables are required on the first port pair to run this test suite. See Figure 4-1 for the proper pinout of the loopback cables.

If the port pair in error is not the first port pair, you can still test it through the **bdtest** menu. Find the appropriate board test menu under **bdtest** (for example, **332.0/sp.2**) and specify the test sequence as follows:

SA: **0123456789**

For additional information, refer to the *MVME332 and MVME331 Asynchronous Communications Controller* section in the *Communication Controller Board Tests* chapter.

If the system contains an MVME332XT board, the first port pair (1 & 2) are tested:

```
1)  open port a        a)  DCD a -> b
2)  open port b        b)  DCD b -> a
3)  loopback a -> b    c)  baud rate a -> b
4)  loopback b -> a    d)  baud rate b -> a
5)  full duplex        e)  parity a -> b
6)  event a -> b       f)  parity b -> a
7)  event b -> a       g)  frame a -> b
8)  break a -> b       h)  frame b -> a
9)  break b -> a       i)  HW flow ctl a -> b
                       j)  HW flow ctl b -> a
```

Loopback cables are required on the first port pair to run this test suite. See Figure 4-1 for the proper pinout of the loopback cables.

For additional information, refer to the *MVME332XT Asynchronous Communications Controller* section in the *Communication Controller Board Tests* chapter.

If the system contains an MVME333 WAN board or an MVME333 X.25 board, selected ports are tested. Refer to the *MVME333 WAN Controller and MVME333 X.25 Controller* section in the *Communication Controller Board Tests* chapter to determine which ports are tested. In any case, the following tests are run:

```
0)  Board Confidence Test
1)  Board Hardware Test
2)  Internal Loopback Test
3)  External Loopback Test
```

Loopback cables are required as described in the *MVME333 WAN Controller and MVME333 X.25 Controller* section in the *Communication Controller Board Tests* chapter. Refer to this same section for a description of the loopback cable pinouts and for information on the tests.

If the system contains a G330 or T330 board, the following tests are performed:

```
0)  LANCE CSR
1)  LANCE Init
2)  LANCE Internal Loopback
```

Refer to the *MVME330 Local Area Network Controller* section in the *Communication Controller Board Tests* chapter for information on these tests.

If the system contains an MVME335 communications board, the following tests are performed:

```
0)  resisters
1)  internal-loop
2)  baud rate
3)  parity
4)  external-loop
5)  rx intr
6)  tx intr
7)  interface intr
8)  frame error
```

Loopback cables are required on all port pairs to run this test suite. See Figure 4-1 for proper pinout of the loopback cables. Refer to the *MVME335 Asynchronous Communications Controller* section in the *Communication Controller Board Tests* chapter for information on these tests.

If the system contains an MVME336 SIO/MUX board, the following tests are performed:

```
xpc.0

0)   XPC DMA

xp.0

C)   cluster confidence
0)   internal loopback
1)   force break
2)   external loopback
3)   force framing error
4)   modem status change
5)   force parity error

siomem

0)   zeros
1)   ones
5)   5555's
a)   AAAA's
d)   address
```

Loopback cables are required on all ports to run this test suite. See Figure 7-3 in the *Communication Controller Board Tests* chapter for proper cabling of the loopback.

Refer to the *MVME336 DeltaLINK Asynchronous Communications Controller* section in the *Communication Controller Board Tests* chapter for information on these tests.

If the system contains an MVME337/MVSB741 board set, the following tests are performed:

```
0)   Run 337 Local Memory Test
5)   Run EVSB Burn-in Confidence Test
```

Loopback cables are required on all port pairs of each octopus cable on all MVSB741 boards. See Figure 4-1 for the proper pinout of the looopback cables. Refer to the *MVME337 IO Engine with MVSB741 Asynchronous Communications Controller* section in the *Communication Controller Board Tests* chapter for information on these tests.

If the system contains an MVME147 monoboard computer, the following tests are performed:

```
psp
1)  Internal Loopback
2)  External Loopback

lan147
1)  LANCE init
2)  LANCE Internal Loopback
```

Loopback cables are required on MVME147 ports 3 and 4 to run this test suite. See Figure 5-1 in the *CPU, Memory, and Misc. Controller Board Tests* chapter for proper cabling of the 9-pin loopback cable used for the MVME147.

If the system contains an MVME167/MVME187 monoboard computer, the following tests are performed:

```
lan1x7
2)  LAN Dump of Registers
1)  LAN Addressing Function
3)  LAN Self Test
4)  LAN Receive Frame Area Ready
5)  LAN Internal Loopback
6)  LAN External Loopback
7)  LAN Command Chaining

llp0
1)  Polled Loopback

lsp3
0)  Execute tests 1-9

lsp2
0)  Execute tests 1-9
```

For lsp3 and lsp2 test descriptions, see MVME167/MVME187 1x7sp Test Menu in the *CPU, Memory, and Misc. Controller Board Tests* chapter. Loopback connectors are required on MVME167/MVME187 serial ports 3 and 4 and the printer port to run this test suite. See Appendix B for proper wiring of the loopback connectors for the MVME167 and MVME187.

If the system is an M8120, the following tests are performed:

```
lan1x7

2)  LAN Dump of Registers
1)  LAN Addressing Function
3)  LAN Self Test
4)  LAN Receive Frame Area Ready
5)  LAN Internal Loopback
6)  LAN External Loopback
7)  LAN Command Chaining

lsp5

0)  Execute tests 1-9

lsp4

0)  Execute tests 1-9

lsp3

0)  Execute tests 1-9

lsp2

0)  Execute tests 1-9

lsp1

0)  Execute tests 1-9
```

For lsp5 through lsp1 test descriptions, see MVME167/MVME187 1x7sp Test Menu in the *CPU, Memory, and Misc. Controller Board Tests* chapter. Loopback connectors are required on M8120 serial ports 1 through 5 to run this test suite. See Appendix B for proper wiring of the loopback connectors for the M8120.

Board Testing

Board testing verifies the operation of a specific board in the system. Before performing a board test, read the section containing information about the board you plan to test or read the **help** screens related to board testing. Both sources describe the various board tests available through the **bdtest** menu. The **view** command is not available for board tests, although it appears in the menus on the screens.

4

To access the **bdtest** menu, type:

`SA:` **bdtest**

A display similar to the following appears, showing all the boards in your system recognized by SSID:

```
Current Menu is /bdtest - "Tests by Board"

131.0      - VME131 68020 CPU Tests
mem.0      - Memory Test
320.0      - VME320 Disk Controller Tests
G330.0     - VME330 GSP LAN Board Test
332.1      - VME332 8 Port Comm. Board Test   (second MVME332 board)
332.0      - VME332 8 Port Comm. Board Test   (first MVME332 board)
333.0      - VME333 WAN Comm. Board Test
x333.0     - VME333 X.25 Comm. Board Test
350.0      - VME350 Streaming Tape Board Tests
355.0      - VME355 Controller/Drive Test
360.0      - VME360 SMD Controller Tests

SA:
```

Figure 4-2. bdtest Main Menu

The **bdtest** Main Menu varies from system to system, displaying those boards that SSID has found in the system. SSID finds the boards during the boot sequence when it probes the VMEbus for each board in the system. If a board is completely dead, it does not appear in the menu. If a board is known to be in the system but is not displayed in the **bdtest** menu, check for proper board and backplane strapping and proper board seating. If the board is still not displayed, replace it.

It is possible to configure the SSID menu selections so that you can access menus for boards that are not actually in the system. Refer to **cf** in Appendix A for details on how to change the board configuration.

To perform specific tests on an individual board, type the board number as displayed in the **bdtest** Main Menu. If you have more than one board of the same type, the first board in the series is designated *board_number*`.0`, the second *board_number*.1, the third *board_number*.2, e.g., **332.0, 332.1, 332.2**. Notice that these boards are listed in the menus in reverse order; the last board in the series is shown first (see Figure 4-2).

After choosing the board to be tested from the **bdtest** Main Menu, you may go through several menu levels before reaching the test selection menu. For example, if you type the following, the **VME320 Disk Controller Tests** menu displays:

SA: **320.0**

```
Current Menu is /bdtest/320.0 - "VME320 Disk Controller Tests"
hd.1                - Winchester Disk
hd.0                - Winchester Disk
fd.1                - Floppy Disk
fd.0                - Floppy Disk
SA:
```

The test selection menu appears only after one of the four menu items is selected. Typing the following brings up the Winchester disk test selection menu:

SA: **hd.0**

```
Selections for Test "Winchester Disk"
0) Recalibrate
1) Quick confidence read
2) Read entire OS disk
3) Random read OS disk
4) Pingpong read OS disk
d) Display OS bad tracks
SA:
```

Once a specific test starts, its progress and status display on the screen. If an error occurs, error information appears followed by a fail message. After a sequence of tests is run, a `SYSTEM TEST PASS` or `SYSTEM TEST FAIL` message displays. If the tests and their status scroll off the screen, use **disperr** to review any failures.

NOTE

> Once a test fails, the `SYSTEM TEST FAIL` message displays. Type the system command **clear** to continue testing.

Performing Multiple Board Tests

Unlike the **confid**, **fault**, **ptest**, and **ptesti** menus that display test suites, the **bdtest** menus display individual tests. Choosing a test from a **bdtest** menu causes only a single test to be performed. However, you can specify multiple tests; type each test identifier (the number or letter preceding the test name) without spaces and press (RETURN).

If, for example, the current menu is the **hd.0** test selection menu shown above, you can type the following to run both the `Recalibrate` and `Quick confidence read` tests:

`SA:` **01** (RETURN) *(no space between test numbers)*

Just as you can specify multiple tests on a single command line, you can also specify tests from other **bdtest** menus with tests from the current menu on a single command line. When using this feature of SSID, separate the test groups with a semicolon (;) and include the test directory of the other **bdtest**. For example, assume that **hd.0** is still the current menu. If you type the following command:

`SA:` **01;esdi.0 01234** (RETURN)
(a semicolon between test groups; a space between `esdi.0` *and* `01234`*)*

SSID will run the `Recalibrate` and `Quick confidence read` tests from the **hd.0** menu followed by the ESDI controller tests, `0` through `4`, on the first ESDI controller. The section *MVME323 ESDI Disk Drive Controller* later in this chapter describes these tests and shows the test selection menu. Notice that it is not necessary to identify the full pathname of the test from the other menu (i.e., **/bdtest/323.0/esdi.0**).

As another example, suppose you want to run the interrupter test on the MVME131.0 board (see the MVME131.0 test selection menu below) in addition to tests `4` and `d` from the current **hd.0** menu, the command line would be:

`SA:` **4d;131.0 1** (RETURN)
(a semicolon between test groups; a space between `131.0` *and* `1`*.)*

The remaining chapters explain the tests available in **bdtest** and show the test selection menus. Use this information as a reference if you decide to run multiple **bdtest** options from a single command line.

Peripheral Testing

Peripheral testing verifies the operation of system peripherals by running a suite of tests. Before performing a peripheral test suite, read this chapter or read the **help** screens related to peripheral testing. Both sources describe the various peripheral test suites available through the **ptest** and **ptesti** menus. The **ptest** menu displays the first 12 peripheral tests available. The **ptesti** menu displays the tests numbered 13 through 24 (if there are that many available). The individual tests making up the suites may be displayed with the **view ptest** and **view ptesti** commands.

NOTE

Only the peripherals attached to the first board of a given type will be tested in the ptest suites. To test peripherals attached to other boards of the same type, use **/bdtest**.

To access the **ptest** menu, type:

`SA:` **ptest**

4

A display similar to the following appears:

```
Current Menu is /ptest - "Peripheral Testing"
1         - 147 SCSI LUN 0,0 Test
2         - 147 SCSI LUN 1,0 Test
3         - 147 SCSI LUN 2,0 Test
4         - 147 SCSI LUN 3,0 Test
5         - 147 SCSI LUN 4,0 Test
6         - 147 SCSI LUN 5,0 Test
7         - 147 SCSI LUN 6,0 Test
8         - VME320 Winchester Drive 0,0 Test
9         - VME320 Winchester Drive 0,1 Test
10        - VME320 Winchester Drive 1,0 Test
11        - VME320 Winchester Drive 1,1 Test
12        - VME320 Floppy Drive 0,2 Test (DESTRUCTIVE)
ptesti    - Next menu. (displays additional drive test suites such as ESDI drive tests.)
SA:
```

Figure 4-3. ptest Main Menu

The drives listed in the **ptesti** and **ptesti** menus are numbered using a *controller,drive* or *target,drive* (for SCSI) Logical Unit Numbering (LUN) scheme. Table 4-1 summarizes the numbering scheme.

Table 4-1. ptest Logical Unit Numbers

LUN Controller,Drive	Device Controller,Drive
0,0	1st MVME320,1st Winchester
0,1	1st MVME320,2nd Winchester
0,2	1st MVME320,1st Floppy
0,3	1st MVME320,2nd Floppy
1,0	2nd MVME320,1st Winchester
1,1	2nd MVME320,2nd Winchester
2,0	1st MVME360,1st SMD
2,2	1st MVME360,2nd SMD
3,0	2nd MVME360,1st SMD
3,2	2nd MVME360,2nd SMD
4,0	1st MVME350,1st Streaming Tape
6,0	1st MVME355,1st 9-Track
8,0	1st MVME323,1st ESDI
8,1	1st MVME323,2nd ESDI
8,2	1st MVME323,3rd ESDI
8,3	1st MVME323,4th ESDI
MVME147/MVME327 SCSI LUN Target,Drive	**Device Target,Drive Physical Addresses**
0,0	SCSI target 0, drive 0
1,0	SCSI target 1, drive 0
2,0	SCSI target 2, drive 0
3,0	SCSI target 3, drive 0
4,0	SCSI target 4, drive 0
5,0	SCSI target 5, drive 0
6,0	SCSI target 6, drive 0

VME320 Winchester Drive Test Suite

The Winchester drive test suite performs a series of nondestructive read-only tests. To run these tests, you must format the disk drives with an operating system format that includes the bad track list for software bad track support. All read tests in the suite check the bad track list for the current track before performing the read operation. If the current track is found in the bad track list, the current track is not read, and the test continues to the next track.

4

Choose the **ptest** menu item that corresponds to the controller and Winchester drive you want to test. The test suite runs in the following order:

```
0) Recalibrate
1) Quick confidence read
2) Read entire OS disk
3) Random read OS disk
4) Pingpong read OS disk
```

To run individual tests in this suite or for a description of the tests, refer to the *Winchester Disk Tests* section in the *Mass Storage Device Controller Board Tests* chapter.

VME360 SMD Drive Test Suite

This test suite performs nondestructive read-only tests. The SMD drives to be tested must be initialized (formatted or have an operating system file system on them). They can be formatted with or without alternate track mapping. The tests automatically handle the alternate tracks.

Choose the **ptest** menu item that corresponds to the controller and SMD drive you want to test. The test suite runs in the following order:

```
0) Recalibrate
1) Confidence test
2) Read entire Disk
3) Random read Disk
4) Ping-Pong read Disk
```

To run individual tests in this suite, refer to the *MVME360 SMD Hard Disk Controller* section in the *Mass Storage Device Controller Board Tests* chapter. The tests are described in *Winchester Disk Tests* section also in that chapter.

VME320 Floppy Drive Test Suite

CAUTION
The Floppy Drive Test Suite performs tests that destroy data on the floppy disk!

All floppy disks used for the test suite must be formatted using the operating system. The first track is formatted by the operating system to single density; it is, therefore, not used in any of the floppy disk tests.

Before starting this test suite, insert a write-enabled, scratch floppy disk into the floppy disk drive. Choose the **ptest** menu item that corresponds to the controller and floppy drive you want to test. The test suite runs in the following order:

```
O) Recalibrate
p) Partial W/R DESTRUCTIVE
4) Pingpong read Disk
3) Random read Disk
```

To run individual tests in this suite or for a description of the tests, refer to the *Floppy Disk Tests* section in the *Mass Storage Device Controller Board Tests* chapter.

VME350 1/4-Inch Streaming Tape Drive Test Suite

> **CAUTION**
>
> **The 1/4-Inch Streaming Tape Drive Test Suite performs tests that destroy data on the tape!**

Before starting this test suite, insert a write-enabled, scratch QIC tape into the tape drive. Choose the **ptest** menu item that corresponds to the 1/4-inch Streaming Tape Drive Test. The test suite runs in the following order:

```
O) Init
r) Retension
t) Write EOT
O) Init
v) Read EOT
O) Init
```

To run individual tests in this suite or for a description of the tests, refer to the *MVME350 Streaming Tape Controller* section in the *Mass Storage Device Controller Board Tests* chapter.

4

VME355 9-Track Tape Drive Test Suite

> **CAUTION**
>
> **The 9-Track Tape Drive Test Suite performs tests that destroy data on the tape!**

Before starting this test suite, load a write-enabled, scratch 9-track tape into the tape drive. Choose the **ptest** menu item that corresponds to the 9-Track Tape Drive Test. The test suite runs in the following order:

```
0) Reset/Initialization
1) Drive Status
8) Rewind to Load Point
6) Write/Read to EOT
8) Rewind to Load Point
```

To run individual tests in this suite and for a description of the tests, refer to the *MVME355 9-Track 1/2-inch Tape Controller* section in the *Mass Storage Device Controller Board Tests* chapter.

VME323 ESDI Disk Drive Test Suite

This test suite performs nondestructive read-only tests. The ESDI drives to be tested must be initialized (formatted or have an operating system file system on them). They can be formatted with or without alternate track mapping; the tests automatically handle the alternate tracks.

Choose the **ptesti** menu item that corresponds to the ESDI drive you want to test. The test suite runs in the following order:

```
0) Recalibrate
1) Confidence test
2) Read entire Disk
3) Random read Disk
4) Ping-Pong read Disk
```

To run individual tests in this suite, refer to the *MVME323 ESDI Disk Drive Controller* section in the *Mass Storage Device Controller Board Tests* chapter. For a description of these tests, refer to the *Winchester Disk Tests* section also in that same chapter.

VME147/VME327 SCSI Tape Drive Test Suite

> **CAUTION**
>
> **The SCSI Tape Drive Test Suite performs tests that destroy data on the tape!**

Before starting this test suite, insert a write-enabled, scratch QIC tape into the tape drive. Choose the **ptest** menu item that corresponds to the SCSI Tape Test. The test suite runs in the following order for VME327:

```
O) Init
r) Retension
t) Write EOT
O) Init
v) Read EOT
O) Init
```

The test suite runs in the following order for VME147:

```
O) Init
r) Retension
w) Write/Read EOT
O) Init
```

To run individual tests in this suite or for a description of the tests, refer to SCSI tape tests in the *MVME147 CPU* section in the *CPU, Memory, and Misc. Controller Board Tests* chapter.

VME147/VME327 SCSI Hard Disk Drive Test Suite

Choose the **ptest** menu item that corresponds to the SCSI Hard Disk Test. The test suite runs in the following order:

```
0) Recalibrate
1) Confidence test
2) Read entire Disk
3) Random read Disk
4) Ping-Pong read Disk
```

To run individual tests in this suite or for a description of the tests, refer to the SCSI hard disk tests in the *CPU, Memory, and Misc. Controller Board Tests* chapter, *MVME147 CPU;* and to the SCSI hard disk tests in the *Mass Storage Device Controller Board Tests* chapter.

VME147 SCSI Floppy Drive Test Suite

You should format all floppy disks used for the test suite using the operating system.

Before starting this test suite, insert a write-enabled, scratch floppy disk into the floppy drive. Choose the **ptest** menu item that corresponds to the SCSI Floppy Drive Test. The test suite runs in the following order:

```
0) Recalibrate
1) Confidence test
2) Read entire Disk
3) Random read Disk
4) Ping-Pong read Disk
```

To run individual tests in this suite or for a description of the tests, refer to the SCSI hard disk and floppy drive tests in the *MVME147 CPU* section of the *CPU, Memory, and Misc. Controller Board Tests* chapter.

VME327 Floppy Drive Test Suite

No device testing is available for the MVME327 Floppy drives using the **ptest** or **ptesti** menu. To test the attached floppy devices, use **/bdtest/327.*n*/flop.*x*,** substituting the board number for *n* and the peripheral number for *x*. Refer to the *Mass Storage Device Controller Board Tests* chapter for a complete description of tests available for the MVME327.

VME328 SCSI ptest Test Suite

No device testing is available for the MVME328 SCSI Controller board using the **ptest** or **ptesti** menu. To test any attached SCSI device, use **/bdtest/328.n/scsi***x*, substituting the board number for *n* and the peripheral number for *x*. Refer to the *Mass Storage Device Controller Board Tests* Chapter for a complete description of tests available for the MVME328.

VME167/VME187/M8120 SCSI ptest Test Suite

No device testing is available for the MVME167/MVME187/M8120 SCSI Controller using the **ptest** or **ptesti** menu. To test any attached SCSI device, use **/bdtest/1x7scsi.0/scsi***x*, substituting the peripheral number for *x*. Refer to the SCSI tests in the MVME167/MVME187 section of the *CPU, Memory, and Misc. Controller Board Tests* Chapter for a complete description of tests available.

CHAPTER 5
CPU, MEMORY, AND MISC. CONTROLLER BOARD TESTS

This chapter describes the board tests available for CPU and memory boards, as well as miscellaneous boards, such as graphics and utility boards. Each section describes the tests available for a particular controller board, the board's test configuration, and, in some cases, the error codes associated with the board. For an introduction to board testing, refer to the *Board Testing* section in the *Confidence, Fault, Board, and Peripheral Testing* chapter.

MVME131XX/MVME132XX, MVME134, and MVME135/MVME136 CPUs

Two tests are available to test the MVME131XT CPU and one for the MVME134 and MVME135/MVME136 CPUs.

To access the MVME131XT CPU tests, type:

SA: 131.0 (RETURN)

To access the MVME134 CPU test, type:

SA: 134.0 (RETURN)

To access the MVME135/MVME136 CPU test, type:

SA: 136.0 (RETURN)

The following display appears for the MVME131XT and a similar one for the MVME134 and MVME135/MVME136:

```
***************************************************************************
************* SYSTEM COMMANDS ****************** MENU SELECTIONS ******
*    help       remote     disperr    **        confid         bdtest     *
*    /          cmdline    version    **        fault          ptest      *
*    disphrd    clear      setopt     **                                  *
*    view       slctdev               **                                  *
***************************************************************************
Selection for Test "VME131 68020 CPU Tests"
   0) Enable MMB      (Do NOT use this option with SSID Version 3.1 or earlier!)
                      (MMU on MVME134)
   1) interrupter     (Not available for the MVME134 or MVME135/MVME136)
SA:
```

For the MVME131XT, the CPU, Cache, and MMB three-board set is extensively tested at power-on and system reset. The `Enable MMB` and `interrupter` options are used to enhance the interactive testing of other system components.

For the MVME134 and MVME135/MVME136 CPUs, only the `Enable MMU` test is available:

`0) Enable MMB`
: Do not use this option with SSID version 3.1 or earlier because it can cause other tests to run incorrectly or not at all. If the MMB (or the MMU on the MVME134) is enabled, it can only be disabled by resetting the system and rebooting SSID.

`1) interrupter`
: This option enables and generates all seven levels of software interrupts on the board.

MVME141 and MVME143 CPUs

Two tests are available for testing the MVME141 CPU and one for the MVME143 CPU.

To access the MVME141 CPU tests, type:

SA: 141.0 (RETURN)

To access the MVME143 CPU test, type:

SA: 143.0 (RETURN)

A display similar to the following appears:

```
*************************************************************************
************* SYSTEM COMMANDS ***************** MENU SELECTIONS ******
*    help       remote     disperr    **      confid        bdtest     *
*    /          cmdline    version    **      fault         ptest      *
*    disphrd    clear      setopt     **                               *
*    view       slctdev               **                               *
*************************************************************************
Selection for Test "VME141 68030 CPU Tests"
   0) Enable MMU
   1) interrupter       (not available on the MVME143)
SA:
```

For the MVME141, the CPU, Cache, and MMU are extensively tested at power-on and system reset. The `Enable MMU` and `interrupter` options are used to enhance the interactive testing of other system components.

For the MVME143 CPU, only the `Enable MMU` test is available:

0) Enable MMU
This option turns on the Memory Management Unit.

1) interrupter
This option enables and generates four levels (1, 2, 4, 5) of software interrupts on the board.

MVME147 Monoboard Microcomputer

NOTE

When SSID is booted on an MVME147, the default SCSI configuration is for a 150Mb disk (WREN III) at SCSI address 0 and a tape drive at SCSI address 4. If the system contains any other SCSI configuration, the **slctdev** command must be executed to configure the proper SCSI devices before execution of any SCSI tests.

The following tests are available for the MVME147 series of Monoboard Microcomputers: LANCE tests (for boards that contain a LANCE chip), Serial Port tests, Enable MMU, Line Printer Test, SCSI tape tests, SCSI hard disk/floppy tests, and VMEchip Interrupter tests.

The MVME147 serial port external loopback test requires loopback cables connected between ports 3 and 4. See Figure 5-1 for the proper pinout for the 9-pin loopback cable. If a system has the older transition board with DB-25 connectors, see Figure 4-1 in the *Confidence, Fault, Board, and Peripheral Testing* chapter for the proper pinout for the 25-pin connector.

The SCSI tape tests verify SCSI tape controller operation by performing write and read tests on the SCSI tape drives.

The SCSI hard disk/floppy tests verify SCSI disk controller operation by performing nondestructive, read-only tests on the SCSI hard disk or floppy drives unless the user enables destructive tests with **cf**. The SCSI disks to be tested must be initialized (formatted or have an operating system file system on them). Refer to the operating system documentation for information on formatting or select the format test from the SCSI drive menus.

When a SCSI hard disk/floppy test displays an error, it also displays the command that failed, the physical sector, and other information about the controller and drive status. This information is saved in a local Error Log that is not cleared by the system command **clear**.

```
Male DB-9                                       Male DB-9
connector                                       connector
  pin #       (cable length 3 inches)             pin #
---------                                       --------
(DCD)  1    <------------------------->  4    (DTR)
(Rx)   2    <------------------------->  3    (Tx)
(Tx)   3    <------------------------->  2    (Rx)
(DTR)  4    <------------------------->  1    (DCD)
(RTS)  7    <------------------------->  8    (CTS)
(CTS)  8    <------------------------->  7    (RTS)
```

Figure 5-1. 9 Pin Loopback Cable Pinout

MVME147 Board Test Menu

To access the test menus for the MVME147 board, type:

SA: **147.0**

A display similar to the following appears:

```
Current Menu is /bdtest/147.0 - "VME147 Board Tests"
psp                    - 147 Serial Port Tests
mmu                    - Enable MMU
plp                    - 147 Line Printer Tests
scsi6                  - SCSI Ctrlr 6 LUN 0 Tests
scsi5                  - SCSI Ctrlr 5 LUN 0 Tests
scsi4                  - SCSI Ctrlr 4 LUN 0 Test
scsi3                  - SCSI Ctrlr 3 LUN 0 Tests
scsi2                  - SCSI Ctrlr 2 LUN 0 Tests
scsi1                  - SCSI Ctrlr 1 LUN 0 Tests
scsi0                  - SCSI Ctrlr 0 LUN 0 Tests
intrp                  - 147 Interrupter Test
lan147                 - 147 LANCE Tests
SA:
```

5

MVME147 mmu, plp, psp, intrp, and lan147 Test Menus

The test selection menus for the `mmu`, `plp`, `psp`, `intrp`, and `lan147` are:

```
SA: mmu <CR>    (to access the mmu test selection menu)

Selections for Test "Enable mmu"

     0) Enable MMU    (Once the MMU is enabled, it cannot be disabled.)

SA: plp <CR>  (to access the line printer test selection menu)

Selections for Test "147 Line Printer Test"

     o) Printer Output    (outputs pattern with length/#lines specified in cf)

SA: psp <CR>    (to access serial test selection menu)

Selections for Test "147 Serial Port Tests"

     1) Internal-loopback    (internal LB tests on ports 3 and 4)
     9) External-loopback    (external LB between ports 3 and 4)

SA: intrp    (to access VMEchip interrupter test selection menu)

Selections for Test "147 Interrupter Test"

     1)  interrupter    (VMEchip interrupter test)

SA:  lan147    (to access LANCE test selection menu)

Selections for Test "147 LANCE Tests"

     1)  LANCE Init    (initialize LANCE chip)
     2)  LANCE Internal Loopback    (internal LB and DMA test)
     a)  Display LANCE address in NVRAM    (shows contents of NVRAM
                  location containing LANCE address)
```

Enable MMU - mmu

Turns on the MMU so that all future memory accesses are done using the MMU. The only way to turn the MMU off again is to power cycle and reboot the system.

Printer Output - plp

Outputs a predefined character string to a printer attached to the MVME147 parallel port. You can change the number of lines to print and the length of each line to print by entering **cmdline** mode, executing **cf 147.0/plp,** and changing the displayed values. The default is 56 lines of 132 characters each.

Internal Loopback - psp

Puts each port of the serial controller chip into internal loopback mode and transfers and verifies a predefined character string. You can change the number of characters to transfer by entering **cmdline** mode, executing **cf 147.0/psp** and changing the displayed value. The default value is 1024 characters.

External Loopback - psp

Places port **3** in receive mode, port **4** in transmit mode, and transfers and verifies a predefined character string. If this process completes successfully, port **3** is placed in transmit mode, port **4** into receive mode, and the test is repeated. You can change the number of characters to transfer by entering **cmdline** mode, executing **cf 147.0/psp** and changing the displayed value. The default value is 1024 characters.

Interrupter - intrp

Checks interrupt levels 1, 2, 4, and 5 of the VMEchip for local interrupts.

LANCE Init - lan147

Checks the ability of the LANCE chip to read an initialization block from memory and configure itself accordingly. No data is transferred other than the initialization block. This test also verifies that the LANCE address location in NVRAM contains a non-zero value.

LANCE Internal Loopback - lan147

Checks the ability of the LANCE chip to access the transmit and receive rings and to transfer data to/from the LANCE buffers. The LANCE chip is placed into internal loopback mode and transfers a predefined character string. You can change the number of characters by entering **cmdline** mode, executing **cf 147.0/lan147**, and changing the displayed value. The default value is 1024. The total number of characters to be transmitted is broken into small packets so that multiple packets are transferred. This test also verifies that the LANCE address location in NVRAM contains a non-zero value.

Display LANCE address in NVRAM - lan147

Displays the contents of the LANCE address location in NVRAM preceded by the default address assigned to Motorola. If the NVRAM location contains 0x02b67, the test displays 0x08003e202b67.

5

MVME147 SCSI Tape Test Menu

```
SA: scsi4 <CR>
Selections for Test "147 SCSI Ctrlr 4 LUN 0 Tests"
(write-enabled scratch tape required)
(This menu is the same for all SCSI tape tests.)
    0) Init                    (returns tape to BOT)
    1) Write Log/EOF           (short write test)
    2) Read Log/EOF            (short read test; run after test 1)
    3) Write/Read EOF          (write short file and read if write passes)
    r) Retension               (fast forward, then rewind the tape)
    s) Erase                   (remove ALL data from tape and rewind)
    w) Write/Read EOT          (write to EOT and read if write passes)
    t) Write EOT               (long write test)
    v) Read EOT                (long read test; run after test t)
    g) Confidence Test         (verifies SCSI interface to drive)
    i) Inquiry                 (displays the vendor and drive ID)
    e) Displays Error Log      (displays the local error log)
    l) Clear Error Log         (clears the local error log)
    m) Mode Sense              (displays drive mode info)
```

Init

This test attaches to the tape device and rewinds the tape if it is not already at BOT.

Write Log/EOF

This test rewinds the tape, then writes two short files and a file mark.

Read Log/EOF

This test rewinds the tape, then reads two short files from the tape. If the data read is not the data that is written using the *Write Log/EOF* test, the test will fail.

Write/Read EOF

This is a combination of tests **1** and **2**. Test **1** is executed and if the files are successfully written, test **2** is executed to read the files.

Retension

On tape devices that support retension, this test will rewind and retension the tape. On devices that do not support retension, this test will rewind the tape.

Erase

This test rewinds and then erases the tape to EOT. The tape is then rewound to BOT.

Write/Read EOT

This is a combination of tests **t** and **v**. Test **t** is executed to write to the tape; if the tape is successfully written, test **v** is executed to read the tape.

Write EOT

This test rewinds the tape, writes blocks containing an incrementing data pattern until EOT is reached, writes a file mark, then rewinds the tape.

Read EOT

This test rewinds the tape, reads blocks that should contain the incrementing data pattern written by test **t** until EOT or file mark is reached, then rewinds the tape. If the data read is not the same incrementing pattern written by test **t**, the test will fail.

Confidence Test

This test verifies that the device can be accessed via the SCSI bus.

Inquiry

This test performs a SCSI inquiry and displays the vendor identification, product identification, and product revision level of the device.

Display Error Log

Refer to the explanation for the MVME147 SCSI disk tests in the following section.

Clear Error Log

Refer to the explanation for the MVME147 SCSI disk tests in the following section.

Mode Sense

This test performs an MVME147 SCSI firmware mode sense and displays the drive attributes, controller type, drive type and number of bytes per block. Refer to the *MVME147 SCSI Firmware User's Manual* for additional information.

MVME147 SCSI Disk Test Menus

```
SA: scsi0 <CR>   (to access the non-destructive menu for the first SCSI disk test)
Selections for Test "147 SCSI Ctrlr 0 LUN 0 Tests"
   0) Recalibrate                (seeks to and reads sector 0)
   1) Confidence test            (reads first sector, last sector, first sector)
   2) Read entire Disk           (reads twice and verifies every sector on the disk)
   3) Random read Disk           (randomly reads 5% of the sectors on the disk)
   4) Ping-Pong read disk        (reads start, then end, then start +1, then end -1,
                                   etc.)
   r) Read a Sector              (queries user for sector to be read)
   s) Read a range of sectors    (queries user for sector range to be read)
   c) Read Capacity              (hard disk only - displays drive capacity in sectors)
   i) Inquiry                    (displays information returned by device)
   e) Display Error Log          (displays local error log)
   l) Clear Error Log            (clears local error log)
```

Recalibrate

This test seeks to sector zero and then reads sector zero, simulating a recalibrate command.

Confidence test

This test reads the first sector, then the last sector, then the first sector a second time.

Read entire Disk

This test sequentially reads a block from the disk, reads the same block into a second buffer, and compares the two buffers. The entire disk will be verified.

Random read Disk

This test will read approximately 5% of the disk. Random numbers are generated for the block numbers to be read. This exercises the seek circuitry of the drive. Each block will be read into two buffers, and the buffers are compared to verify the data.

Ping-Pong read disk

This test reads the first block, then the last block, then first block +1, then last block -1 until the entire disk has been read in both directions. This actually results in the entire disk being read twice.

Read a Sector

This test allows the user to select, using hexadecimal or decimal sector numbers, any sector to be read.

Read a range of sectors

This test allows the user to select, using hexadecimal or decimal sector numbers, any range of sectors to be read.

Read Capacity

This test displays the drive capacity in sectors. The capacity is displayed in both hexadecimal and decimal.

Inquiry

This test performs a SCSI inquiry and displays the information returned by the device.

Display Error Log

The first 20 SCSI errors occurring on each drive are saved in the Error Log. After completion of SCSI tests, you can display the Error Log for each drive using this menu option. The floppy drive must be ready, i.e., floppy installed, for this command to execute. The system **clear** command does not clear, or give the user the option of clearing, this device Error Log. (Refer to the *Clear Error Log* section.) If no errors occurred on this device, the following displays:

```
SA: e
/bdtest/147.0/scsi4    e)Display Error Log       ... starts
End of SCSI Error Log
/bdtest/147.0/scsi4    e)Display Error Log       ... passed
```

5

A display similar to the following displays if errors occurred on this device:

```
SA: e
 /bdtest/147.0/scsi4     e)Display Error Log       ......starts

Pass    Code/Sector        Command              Error Message
 1      0x00000027         WRITE                PROTECTED_MEDIA
          Class 7 Sense is controller-dependent.
          Refer to vendor manuals to decode.
          Error Code: 0x70        Valid Code: 0x80
          Sense Key:  0x07    -   Media is write-protected
          Add'l. Sense Key:  0x00
          f0 00 07 00 00 00 80 06 00 00 00 00 00 00 45 20
          56 49 50 45 52 20 31 35 30 20 20 32 31 32 34 37
          00 00 00 00 00 00 00 00 00 00 00 00

Pass    Code/Sector        Command              Error Message
 1      0x00000022         TAPE_ATTACH          DEVICE_NOT_READY

Pass    Code/Sector        Command              Error Message
 2      0x00000022         TAPE_ATTACH          DEVICE_NOT_READY

 Press RETURN for next page

Pass    Code/Sector        Command              Error Message
 3      0x00000022         TAPE_ATTACH          DEVICE_NOT_READY

Pass    Code/Sector        Command              Error Message
 4      0x00000022         TAPE_ATTACH          DEVICE_NOT_READY

 Press RETURN for next page

 End of SCSI Error Log
 /bdtest/147.0/scsi4  e)Display Error Log           ......passed
***SYSTEM TEST FAILED***
 Depress Return to Continue
```

Clear Error Log

Clears the local device error log. This will have no effect on the system error log that is accessed using the **disperr/clear** system level commands.

Destructive Disk Tests

CAUTION

Destructive disk tests destroy all data on the drive. Be extremely careful with test selection after enabling destructive tests with either the "eval flag" and/or the "set test scripts" command.

5

Destructive and expert mode hard/floppy disk tests are available from the menu by going into the cmdline mode and executing **cf** for *each* SCSI device on which you wish to execute destructive tests. By changing the `eval flag` to `1`, and using **set menu** to return to the menu system, the following tests appear on the disk menus:

d) Display Defects *(hard disk only - displays the drive's defect lists)*
m) Mode Sense/Select/Verify/Display *(hard disk only - executes mode sense, displays sense - data, executes mode select)*
f) Format Drive (DESTRUCTIVE) *(formats drive)*
a) Assign Alternate (DESTRUCTIVE) *(hard disk only - adds defects to the drive's grown defect list)*
u) Wr/Rd/Cmpr Last 1% (DESTRUCTIVE) *(hard disk only - writes, reads twice and verifies - the last 1% of the drive)*
w) Write Entire Disk (DESTRUCTIVE) *(writes all sectors)*
z) Set Test Scripts *(toggles confid 5 to / from destructive tests)*

The Destructive Floppy menu contains one additional test:

Partial Write Test (DESTRUCTIVE)

Randomly writes some sectors on the floppy.

5

Display Defects

This test displays the manufacturer's and the grown defect list in logical sector number format, if that format is supported by the drive. This test does not appear on the floppy disk menu.

Mode Sense/Select/Verify/Display

This test executes a SCSI mode sense command for all changeable pages and displays, in hexadecimal, the mode sense data. A vendor manual for the device is required to decode the values. A mode select will be executed using the data received during execution of the mode sense. A second mode sense is executed and the data is compared to the data from the first mode sense to verify that no changes were made. This test does not appear on the floppy disk menu.

Format Drive (DESTRUCTIVE)

Formats the disk using the same parameters used during read tests (i.e., if the drive is configured as 256 bytes per sector, the drive is formatted using these parameters.)

NOTE

This is a diagnostic format only. You must format the drive from the operating system before use by the operating system.

Assign Alternate (DESTRUCTIVE)

This command allows a defective block to be redirected without reformatting the entire disk *but should be carefully used on any disk* containing data that should not be destroyed. Data contained in the block to be reassigned is lost.

Write Entire Disk (DESTRUCTIVE)

Writes all sectors on the hard disk, block by block, then reads the block and verifies that the pattern read is the same pattern that was written.

Wr/Rd/Cmpr Last 1% (DESTRUCTIVE)

Writes the last 1% of the total sectors on the disk with a unique pattern, reads the blocks into a buffer, and verifies that the pattern read is the same pattern that was written. This test does not appear on the floppy disk menu.

Set Test Scripts

Used to reconfigure **confid** string 5 (Continuous Running Intensive I/O check with Comm.) for destructive testing on hard and floppy disks with the `eval flag` set to "1". The destructive mode flag for 147 SCSI devices is toggled each time this command is executed at any disk menu. If destructive mode is enabled, the following message displays:

```
WARNING: DESTRUCTIVE DISK TESTS FOR ALL 147 SCSI FLOPPY AND
         HARD DISKS WITH THE EVAL FLAG SET HAVE BEEN ADDED TO
         CONFID #5 AND FAULT SCRIPTS.
```

If destructive mode for 147 SCSI devices is disabled, the following message displays:

```
ALL 147 DISK CONFID AND FAULT SCRIPTS ARE NOW READ ONLY
```


Destructive test *warning messages are disabled* during confid 5 testing to allow unattended destructive tests.

Using the **view confid** command shows destructive tests for *all* 147 disks, but the destructive tests are only run on drives with the `eval flag` set to 1. Any disk with the `eval flag` set to 0 displays the following message (where X is the SCSI controller ID and Y is the drive LUN):

```
Unable to perform test on 147 target X drive Y - eval flag
not set
```

Normally, destructive disks display a warning message and ask the user if they want to continue before any destructive tests are run. This warning message can be overridden by using **cf** to change the **conf** flag to a `1`. If the `Conf flag` is set to a "1", no warning messages display.

MVME147 Board Test Configuration

The MVME147 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 147.0
/bdtest/147.0 configuration
   base[0xFFFE1000] = 0xFFFE1000 ?       # base address of pcc chip
   level [0-7] = 6 ?                     # default interrupt level
   mode [0] = 0 ?                        # cannot be
   warning msgs [0] = 0 ?                # changed by user

> cf 147.0/psp
/bdtest/147.0/psp configuration
   xfer size [0-1024] = 1024 ?           # size per transfer
```

```
> cf 147.0/scsi0
/bdtest/147.0/scsi0 configuration
   Ctrlr number [2] = 2 ?                          # SCSI target number
   phy-drive [0-3] = 0 ?                           # drive number
   vector number [0-0xFF] = 0x4D ?                 # SCSI fw vector number
   Conf flag 0-Warnings ON 1-Warnings OFF [0-1] = 0 ? # disable warning
   Eval flag 0-NO write 1-WRITE TESTS [0-1] = 0 ? 0 # enable destructive tests
```

```
> cf plp
/bdtest/147.0/plp configuration
   line length [0-136] = 132 ?                     # length of line to print
   number lines [0-4096] = 56 ?                    # number of lines to print
```

5

```
> cf lan147
/bdtest/147.0/lan147 configuration
   xfer size [0-1024] = 1024 ?
```

MVME147 Error Messages

General Error Messages

XXXXX **Memory not available**
XXXXX YYYYY **Memory not available**
XXXXX YYYYY **NO Memory for Buffer**

Indicates that no memory was available when a buffer was requested by test *XXXXX*. *YYYYY* indicates the specific function within a test, if available. Possibly there are too many tests running, defective memory or 147 board.

XXXXX YYYYY **No system semaphore**
XXXXX YYYYY **No semaphores available**

Indicates that no semaphore was available when requested by test *XXXXX*. *YYYYY* indicates the specific function within a test, if available. Possibly there are too many tests running, defective memory or 147 board.

PCCchip is programmed for Lvl 7 IRQ on Bus Error

Indicates the PCCchip Bus Error Interrupt Control register has been programmed to generate a level 7 IRQ when a bus error occurs. This indicates a problem on the 147 board as SSID disables the interrupt during initialization at boot time.

Error in setvec from *XXXXX* **(***YYYYY***,** *ZZZZZ***)**
****Help setvec failed in** *XXXXX* **(***YYYYY***)**
****Help setvec failed in** *XXXXX* **(***YYYYY***)**

Indicates an error occurred while SSID was attempting to set interrupt vectors. *XXXXX* indicates the SSID function where the error occurred. *YYYYY* indicates the vector address SSID was attempting to change so that interrupt handler *ZZZZZ* is executed when the interrupt occurred. This indicates a problem on the 147 board.

Error in setvec in usrinit for 147 LANCE

Indicates an error occurred while setting up interrupt handler for the 147 LANCE chip. This indicates a problem on the 147 board.

Name of this routine is cpux147. Some program accessed the Location Monitor adrs $FFFE0000 thru $FFFEFFFF.

Indicates that the location monitor was accessed when it should not have been accessed. Some board in the system is incorrectly accessing memory.

SSID fatal error: get_stbl() - unrecognized SCC channel address
SCC channel address = 0x*XXXXX*
SSID IS NOW HALTED

An interrupt was received through the vector assigned to the SCC chips but an invalid SCC channel address (*XXXXX*) was received. Some device in the system is interrupting through the wrong vector or an SCC is returning an incorrect channel address. You must reboot the system for any further action to occur.

System Test error: Unrecognized SCC interrupt
SCC interrupt status = *XXXXX*
SSID IS NOW HALTED

An interrupt was received through the vector assigned to the SCC chips but an invalid interrupt status was received. Indicates a problem with the SCC chip(s) on the 147 board. The system is halted and must be rebooted for any further action to occur.

XXXXX Max number of errors exceeded

The device indicated by *XXXXX* has exceeded the maximum number of allowed errors. No additional errors will be logged for this device.

XXXXX **SIGLP_IRQ failed**
XXXXX **TIME_OUT -- SIGLP_IRQ**
XXXXX **LMO_IRQ failed**
XXXXX **TIME_OUT -- LMO_IRQ**
XXXXX **LM1_IRQ failed**
XXXXX **TIME_OUT -- LM1_IRQ**
XXXXX **SIGHP_IRQ failed**
XXXXX **TIME_OUT -- SIGHP_IRQ**

Indicates failure of the VMEchip interrupter test. Possibly a defective VMEchip or 147 board. *XXXXX* indicates the 147 board "intrp" test.

5

XXXXX **TIMEOUT**

Indicates a test on device *XXXXX* timed out before the device interrupted with completion status. This can indicate a problem with device *XXXXX* or possibly the loopback cable is loose if the device is a serial port.

LANCE Error Messages

XXXXX **LANCE test invalid on this board**

SSID has determined that no LANCE chip exists on the 147 board and therefore, no LANCE tests are supported.

NO LANCE detected on this board

Indicates that no LANCE tests will be run on 147 boards that do not contain a LANCE chip.

LANCE ADDRESS IS NOT PROGRAMMED. VALUE IS 0x08003e2*ZZZZZ*

Indicates that the value contained in the NVRAM location reserved for the LANCE address is an invalid value (usually 0). ZZZZZZ will indicate the value that is contained in the NVRAM location.

XXXXX **147 LANCE** *YYYYY* **FAILED**

Indicates a failure of 147 LANCE tests (*XXXXX*). *YYYYY* indicates the failing test.

REG *XX* **EXP:** *YY* **GOT:** *ZZ*

Indicates an unexpected reading from LANCE register *XX* during testing. SSID expected to read *YY* from the register but instead read ZZ.

147 LANCE TEST ERROR # *XXX*

Indicates a LANCE test failure where *XXX* is:

1	**147 board resource timeout**
2	**147 board ram parity error**
3	**147 board unexpected exception**
4	**147 board invalid test command from host**

8	147 board ACFAIL
20	csr error - set with one
40	csr error - set with zero
60	csr error - cannot set with one
80	csr error - cannot clear with one
100	csr error - cannot clear with zero
120	csr error - cleared with one
140	csr error - cleared with zero
160	csr error - unexpected bits on
164	csr error - unexpected bits on
168	csr error - unexpected bits on
172	csr error - unexpected bits on
176	csr error - unexpected bits on
180	csr error - unexpected bits on
184	csr error - unexpected bits on
188	should have STOP set, strt & INIT clear
189	CSR read while INIT bit set
191	IDON not set after INIT
192	INTR not set after INIT
193	unexpected bits on after INIT
194	RXON or TXON not set after START
195	unexpected bits on after START
196	IDON not set
197	unexpected bits on after INIT
198	no packet received
199	TINT not set after transmit
200	transmit interrupt count invalid
201	receiver interrupt count invalid
202	missed packet interrupt count invalid
203	expected packet not received
204	expected packet not received
205	net too busy - retry test
206	buffer space unavailable
207	buffer space unavailable
208	not enough transmit buffer
209	all transmit buffers used up
210	unexpected transmission error
211	expected transmission error not occurred
212	babbling transmitter
213	memory error in transmission
214	collision error after transmit
215	unexpected receive error
216	expected receive error not occurred
217	memory error in receive
218	received and sent data not match
219	trailing data bytes missing

5

220	`rmd1 has end of packet set only`
221	`bad CRC found in received packet`
222	`RINT not set after transmit in loopback`
223	`Expected packet not received`
224	`Transmitter should be disabled`
225	`Unexpected packet transmitted`
226	`Receiver should be disabled`
227	`Unexpected packet received`

`UNDEFINED LAN STATUS:` *XXXXX*

Failure code *XXXXX* was returned from the LANCE chip but this is not a defined error number.

5

Serial Port Error Messages

XXXXX `Data miscompare`

Indicates a miscompare of data during serial port tests.

Parallel Printer Error Messages

`plp Printer status: ACK`*xxx* `present,`
`FAULT`*xxx* `present, SELECT`*xxx* `present`
`PAPER/EMPTY`*xxx* `present, BUSY`*xxx* `present`

where *xxx* is either *is* or *is not* to indicate the presence/absence of the indicated status signal.

SCSI Error Messages

`SCSI timeout - no response from target`*X* `drive`*Y*

The SCSI command was sent but the device did not interrupt with completion status in the time period allowed by SSID for completion of the command. *X* indicates the SCSI target ID (controller), *Y* indicates the LUN (device) ID. This message also occurs if the SCSI bus is hung. Try the BUG reset command or power cycle the system.

`WARNING: Unexpected interrupt received from`*XXX*`.`
`Device is interrupting when it should not be OR`
`command completion took longer than the test expected OR`
`completion status was returned after the user pressed the BREAK`
`key.`

This message occurs if the SCSI device interrupts after a timeout has occurred or when SSID is not waiting for an interrupt from the device. *XXX* indicates from which SCSI device the interrupt was received. If the user presses the BREAK key during execution of a SCSI command that takes a long time to complete, e.g., a tape erase, this message can be ignored because it indicates the device has completed the command but SSID was no longer waiting for the completion status. (Thirty seconds after detection of the BREAK SSID stops all running processes. There is no method in SSID to tell the device to halt the command so the device continues and then interrupts to return completion status but the process that was waiting for the status no longer exists.) Any other time this message is received, try resetting the SCSI bus or power cycling the system and rebooting SSID. If the error continues, SSID could have a timeout that is too short or the SCSI device or 147 SCSI interface could be defective. Also check SCSI cabling and SCSI bus termination.

5

`UNDEFINED SCSI CMD:` *XXXXX*

Indicates an error during testing of an SCSI device. An undefined command (*XXXXX*) was indicated as failing. Possibly caused by defective memory or SCSI interface on 147 board.

`UNDEFINED SCSI STATUS:` *XXXXX*

A SCSI device returned status of *XXXXX*, which is undefined in the SCSI status tables. Refer to the vendor manual for the device or the *MVME147 SCSI Firmware User's Guide* for further information.

`No sector number returned from drive. Failing sector number could be in the range of` *XXXXX* through *YYYYY*.

SSID tried to read all sectors from *XXXXX* through *YYYYY*, inclusive, and a failure occurred but the device did not report the failing sector number. Actual failing sector could be any sector in the indicated range.

XXXXX `Unable to read disk capacity`

SSID was unable to execute the SCSI Read Capacity command on the device indicated by *XXXXX*. Indicates defective SCSI device or device that does not support this command or defective SCSI interface on 147 board.

`Target number` *X* `out of range - must be between 0 and` *Y*

The selected SCSI target number is less that 0 or greater than the maximum allowed. This error can occur if an incorrect value is entered in the configuration structure using the **cf** command line command or the SCSI device table in memory has been overwritten. Target addresses are between 0 and 6.

`No device assigned for target` *X*

The device pointer in the SCSI device table is NULL. This occurs if the SCSI device table in memory has been overwritten.

`Invalid device type` *X* `for target` *Z*

The device type X in the SCSI device table is invalid for target number Z. This occurs if the SCSI device table in memory has been overwritten.

XXXXX `The drive selected is invalid`

The selected SCSI drive number is invalid for device *XXXXX*. This error can occur if an incorrect value is entered in the configuration structure using the **cf** command, or if the SCSI device table in memory has been overwritten.

`Unable to perform test on 147 target` *X* `drive` *Y* `- eval flag not set`

Indicates destructive test will not be run on SCSI target *X* (controller) drive *Y* (LUN). This message displays during destructive confid/fault tests for drives that are not enabled for destructive tests.

5

`Unable to start test - last test TIMED OUT -` *XXXXX*`. System must be reset to further test` *XXXXX*.

The SCSI device specified by *XXXXX* timed out before completing the last test run. No other tests can be run on the device until a system or SCSI reset has been executed from the 147BUG prompt or by pressing the *RESET* switch. SSID must then be rebooted before testing can continue. This generally indicates a problem with the SCSI device but might also be caused by excessive retries causing a test to take longer than expected.

`Error in Modesense Command`

An error occurred during execution of the SCSI mode sense command.

`Error in Modeselect Command`

An error occurred during execution of the SCSI mode select command.

`Invalid read request on device` *XXXXX*`, want` *YYYYY*`, have` *ZZZZZ*

A request to read past the last sector on the disk was received. This would indicate a problem in memory, the SCSI interface, or SCSI bus.

`First pass read error`

A read error was reported by the drive on the first pass read of a test that does double read verification.

`Second pass read error`

A read error was reported by the drive on the second pass read of a test that does double read verification.

After the first and second pass reads have been completed for a test that does double read verification, the two buffers are compared. The following messages indicate the disk sector and memory buffer locations at which a miscompare occurred:

```
Data Comparison Error:
DISK INFO:
Read NNN (0xNNN) sectors starting at LLL (0xLLL)
Compare failed on byte BBB (0xBBB) which is
byte BBB (0xBBB) of sector SSS (0xSSS)
which is the number ZZZ (0xZZZ) sector read.
MEMORY BUFFER INFO:
Top of block 1: CCC
Top of block 2: DDD
Compare data read:
Address: BUF1 Value read: XXX
Address: BUF2 Value read: YYY
```

The values displayed have the following significance:

BBB	byte number within disk sector or memory buffer
BUF1	address in buffer 1 where the miscompare occurred
BUF2	address in buffer 2 where the miscompare occurred
CCC	starting address of buffer 1
DDD	starting address of buffer 2
LLL	starting logical sector number of the block in which the error occurred
NNN	number of sectors in the block read
SSS	sector number in which the failure occurred
XXX	contents of buffer 1 at the location where the miscompare occurred
YYY	contents of buffer 2 at the location where the miscompare occurred
ZZZ	sector count, from the first one read, in which the error occurred

`Unable to execute mode sense/select on controller type` *XXXXX*
According to the SCSI configuration, the drive being selected is of controller type XXXXX and is not capable of supporting this command.

5

5

SCSI Packet Display

If an error occurs during execution of a SCSI packet and the display packet (dp) is disabled, a message similar to the following appears:

```
/bdtest/147.0/scsi0       2) Read entire Disk      ...... starts
PASS=0, FATAL=000, ERROR=000 0:00:43 /bdtest/147.0/scsi0:waiting
  Final Status Code 23 NONRECOVERABLE_(HARD)_ERROR
  SCSI COMMAND 0, READ
  Error on SCSI target 0 drive 0
  SECTOR Adrs 0x00007c2d
/bdtest/147.0/scsi0       2) Read entire Disk      ...... failed
  NONRECOVERABLE_(HARD)_ERROR 0x00007c2d
*** SYSTEM TEST FAILED ***
```

In this example, the final status code 23 indicates that a nonrecoverable or hard error occurred while executing a SCSI **read** command on SCSI target 0, drive 0. The read error occurred on sector 0x00007c2d.

When the test is completed, a recap of the errors displays, listing the error type and the sector at which the error occurred.

If the display packet (dp) option is enabled, additional information similar to the following displays:

```
/bdtest/147.0/scsi0        2) Read entire Disk        ...... starts
PASS-0, FATAL-001, ERROR-000    0:00:43 /bdtest/147.0/scsi0   : waiting
 SCSI PKT STATUS 0-3 40 23 80 00
 Stat0:Final_status:External_stat_Valid:rte_return:No_restart_cmd:
No_que'ed_packet:Bit_Res_Low:Bit_Res_Low:Bit_Res_Low:
 Final Status Code 23 NONRECOVERABLE_(HARD)_ERROR
 Stat2: Phase code 00
 Stat3: Bit_Res_Low :Bit_Res_Low :_ :_ :
 :_ :_ :Bit_Res_Low :
 SCSI COMMAND 0, READ
 Error on SCSI target 0 drive 0
 SECTOR Adrs 0x00007c2d
/bdtest/147.0/scsi0        2) Read entire Disk        ...... failed
 NONRECOVERABLE_(HARD)_ERROR 0x00007c2d
*** SYSTEM TEST FAILED ***
```

The additional information for this example includes the following. Refer to the *MVME147 SCSI Firmware User's Manual* for details.

`SCSI PKT STATUS 0-3 40 23 80 00`
The values of the SCSI status bytes 0 through 3 are 40, 23, 80, and 00. The balance of the display defines the status represented by these bytes. You can also use Tables 5-1 and 5-2 to decode the status bytes.

`Stat0: Final_status`
This is final, not intermediate, status.

`:External_stat_Valid`
External status information is valid and available to the test program.

`:rte_return`
A rte is required by the test program to continue processing from where an interrupt occurred.

`Bit_Res_Low`
The bit is reserved and should be 0.

`Stat2: Phase code 00`
The SCSI bus phase in which the error occurred.

`Stat3: Bit_Res_Low . . . :Bit_Res_Low :`
An interpretation of status byte 3. In this example there is no pertinent information.

Table 5-1. MVME147 SCSI Packet Return Status

STAT 0 (15 – 08)	STAT 1 (07 – 00)
Control Flags	**Status Code (Refer to Table 5-2)**

Bits 12-8 (reserved)

Bit 13 (RTE FLAG) (of interest only to the programmer)

1 = This return was not preceded by an interrupt and is the first return since command entry. In this case, no RTE is required.

0 = This return was preceded by an interrupt and is not the first return, therefore, an RTE is required to continue processing from where an interrupt occurred. Register A3 has a pointer to a register save area (D0-D7, A0-A6).

Bit 14 (ADDITIONAL STATUS)

1 = External status is valid.

0 = External status is not valid.

Bit 15 (FINALSTAT)

1 = Intermediate return.

0 = Final status.
The script processing completed successfully, OR the script processing encountered a fatal error.

NOTE

This does not mean that the operation the user requested on the SCSI was successful. The status is contained in the status code (bits 7-0.)

Table 5-1. MVME147 SCSI Packet Return Status (cont'd)

STAT 2: 15–12	11–08	STAT 3: 07–04	03–00
SCSI Phase Status	**Reserved**		**Request Sense Flags (0 if not used)**

- Bit 0 (RES) Re-served)
- Bit 1 (ILI) Incorrect length indicator.
- Bit 2 (EOM) Logical end of media (early warning).
- Bit 3 (FM) Filemark
- Bit 4 (COMMAND RETRY) 1 = Retries were performed.
- Bit 5 (COMMAND RETRY OVERFLOW) 1 = Retry overflow. The command was retried "retry count" times.
- Bits 6 and 7 (reserved)
- Bit 15 (PHASE VALID) 1 = Phase code is in D14-D12. 0 = Data in D14-D12 is not valid.

5

Table 5-2. MVME147 Packet Status Codes

Code	Meaning	Notes
	Intermediate Return Codes	
$02	Wait for interrupt; command door open. OK to send new commands for other devices to firmware.	1
$04	A message has been received. User must interpret.	1
	Final Return Codes	
$00	Good. Script processing is OK.	2
$01	Undefined problem.	2
$02	Reserved.	2
$03	Interrupt handler was entered with no pending IRQ ($FFFE0788).	2
$04	Reselection not expected from this target.	2
$05	TARGET thinks it is working on linked commands but the command table does not.	2
$06	Linked command has error status code; command has been aborted.	2
$07	Received an illegal message.	2
$08	The message we have tried to send was rejected.	2
$09	Encountered a parity error in data-in phase, command phase (TARGET only), status phase, or message-in phase. (Refer to bits 15-12 of second status word.)	2
$0A	SCSI bus RESET received.	2
$0B	Command error (bad command code, bad timing, or command door was closed when a command was received) = 00. Custom SCSI sequence: controller level not equal to "147 local level", or interrupt not on. Format: format with defects on a controller type not supported. Controller reset: controller not SCSI type. Space (tape): undefined mode. Mode select (tape): undefined controller type. Mode sense (tape): undefined controller type.	2

Table 5-2. MVME147 Packet Status Codes (cont'd)

Code	Meaning	Notes
$0C	Size error (invalid format code).	2
$0D	Bad ID in packet or local ID ($FFFE07A6).	2
$0E	Error in attach (not previously attached, bad device LUN, unsupported controller, target SCSI address conflicts with initiator).	2
$0F	Busy error (device has a command pending).	2
$10	Byte Count Error. There is disagreement between initiator and TARGET about the number of bytes that are to be transferred. If bit 15 of status = 1, then bits 12-14 contain the phase code.	2
$11	Received a BERR* while in DMA mode from a device that did not respond fast enough.	2
$12	Selection time-out. TARGET does not respond.	2
$13	SCSI protocol violation. Controller reset: controller not SCSI.	2
$14	Script mismatch. CHECK STATUS. If SCSI status within Command Table (offset $14 for custom sequence, otherwise $64) is zero, then assume script mismatch, otherwise use SCSI packet status.	2
$15	Script mismatch. The TARGET sequence of operation did not match the script.	2
$16	Illegal SCSI state machine transition.	2
$17	Command has been received (in TARGET role).	2
$18	Script complete in TARGET role.	2
$19	Script complete and new command loaded (TARGET role).	2
$1A	TARGET module called. TARGET role not supported.	2
$1B	TARGET module rejected an initiator message and returned with this status to a particular LUN service routine.	2

5

5

Table 5-2. MVME147 Packet Status Codes (cont'd)

Code	Meaning	Notes
$1C	TARGET module sent a check status with an "illegal request" sense block to some initiator because the particular LUN that the initiator wanted was not enabled.	2
$1D	TARGET module sent a busy status to the calling initiator because the particular LUN that the initiator wanted was already busy servicing a command.	2
$1E	Reserved and unused.	2
$1F	Reserved.	2
	Request-Sense-Data Error-Class 7 Codes (Controller-Dependent)	
$20	NO SENSE. Indicates that there is no specific sense key information to be reported for the designated logical unit.	2,3
$21	RECOVERED ERROR. Indicates that the last command completed successfully with some recovery action performed by the TARGET. Details can be determined by examining the additional sense bytes and information bytes.	2,3
$22	NOT READY. Indicates that the logical unit addressed cannot be accessed. Operator intervention may be required to correct this condition.	2,3
$23	MEDIUM ERROR. Indicates that the TARGET detected a nonrecoverable error condition that was probably caused by a flaw in the medium or an error in recording data.	2,3
$24	HARDWARE ERROR. Indicates that the TARGET detected a nonrecoverable hardware failure (for example, controller failure, device failure, parity error) while performing the command or during self test.	2,3
$25	ILLEGAL REQUEST. Indicates that there was an illegal parameter in the command descriptor block or in the additional parameters supplied as data.	2,3

Table 5-2. MVME147 Packet Status Codes (cont'd)

Code	Meaning	Notes
$26	UNIT ATTENTION. Indicates that the removable media may have been changed or the TARGET has been reset.	2,3
$27	DATA PROTECT. Indicates that a command that Reads or Writes the medium was attempted on a block that is protected from this operation.	2,3
$28	BLANK CHECK. Indicates that a write-once read-multiple device or a sequential access device encountered a blank block while reading or a write-once read-multiple device encountered a nonblank block while writing.	2,3
$29	VENDOR UNIQUE. Used for reporting vendor unique conditions.	2,3
$2A	COPY ABORTED. Indicates that a copy or a copy and verify command was aborted due to an error condition.	2,3
$2B	ABORTED COMMAND. Indicates that the TARGET aborted the command. The initiator may be able to recover by trying the command again.	2,3
$2C	EQUAL. Indicates a search data command has satisfied an equal comparison.	2,3
$2D	VOLUME OVERFLOW. Physical EOM. Indicates that a buffered peripheral device has reached an end-of-medium and data remains in the buffer that has not been written to the medium. A recover buffered data command may be issued to read the unwritten data from the buffer.	2,3
$2E	MISCOMPARE. Indicates that the source data did not match the data read from the medium.	2,3
$2F	RESERVED. This sense key is reserved.	2,3

5

5

Table 5-2. MVME147 Packet Status Codes (cont'd)

Code	Meaning	Notes
	SCSI Status Returned in Status Phase	
$31	SCSI status = $02. CHECK.	2,4
$32	SCSI status = $04. CONDITION MET.	2,4
$34	SCSI status = $08. BUSY.	2,4
$38	SCSI status = $10. INTERMEDIATE / GOOD.	2,4
$3A	SCSI status = $14. INTERMEDIATE / CONDITION MET / GOOD.	2,4
$3C	SCSI status = $18. RESERVATION CONFLICT.	2,4
	Request-Sense-Data Error-Classes 0-6 Codes (Controller-Dependent)	
$40	NO ERROR STATUS.	2,5,6
$41	NO INDEX SIGNAL.	2,5,6
$42	NO SEEK COMPLETE.	2,5,6
$43	WRITE FAULT.	2,5,6
$44	DRIVE NOT READY.	2,5,6
$45	DRIVE NOT SELECTED.	2,5,6
$46	NO TRACK 00.	2,5,6
$47	MULTIPLE DRIVES SELECTED.	2,5,6
$49	CARTRIDGE CHANGED.	2,5,6
$4D	SEEK IN PROGRESS.	2,5,6
$50	ID ERROR. ECC error in the data field.	2,5,7
$51	DATA ERROR. Uncorrectable data error during a read.	2,5,7
$52	ID ADDRESS MARK NOT FOUND.	2,5,7
$53	DATA ADDRESS MARK NOT FOUND.	2,5,7

Table 5-2. MVME147 Packet Status Codes (cont'd)

Code	Meaning	Notes
$54	SECTOR NUMBER NOT FOUND.	2,5,7
$55	SEEK ERROR.	2,5,7
$57	WRITE PROTECTED.	2,5,7
$58	CORRECTABLE DATA FIELD ERROR.	2,5,7
$59	BAD BLOCK FOUND.	2,5,7
$5A	FORMAT ERROR. (Check track command.)	2,5,7
$5C	UNABLE TO READ ALTERNATE TRACK ADDRESS.	2,5,7
$5E	ATTEMPTED TO DIRECTLY ACCESS AN ALTERNATE TRACK.	2,5,7
$5F	SEQUENCER TIME OUT DURING TRANSFER.	2,5,7
$60	INVALID COMMAND.	2,5,8
$61	ILLEGAL DISK ADDRESS.	2,5,8
$62	ILLEGAL FUNCTION.	2,5,8
$63	VOLUME OVERFLOW.	2,5,8

5

5

Table 5-2. MVME147 Packet Status Codes (cont'd)

NOTES:

1. Intermediate return codes. Bit 15=1, actual word=$80xx, $90xx, etc.
2. Final return codes.
3. Sense key status codes for request-sense-data error — class 7. An offset of $20 is added to all sense key codes.
4. The SCSI status sent from the controller is ANDed with $1E, shifted right one bit, and $30 added.
5. Sense key status codes for request-sense-data error -- classes 0-6. An offset of $40 is added to all sense key codes.
6. Drive error codes.
7. Controller error codes.
8. Command errors.

M8120 Microcomputer System

The following tests are available for the M8120 system: LAN tests, Serial Port tests, and SCSI device tests.

The tests used on the M8120 are the same tests used for the MVME187; refer to the *MVME167/MVME187 Monoboard Microcomputer* section for further information.

The M8120 differs slightly from the MVME187. The M8120 does not have external connections for the printer port, so use of those tests is not recommended. For the M8120 serial port tests, special loopbacks are required. See the loopback wiring information in Appendix B. The M8120 has six serial ports. Ports 1 through 5 are tested by default. Port 0 is the console.

MVME167/MVME187 Monoboard Microcomputer

The following tests are available for the MVME167/MVME187 series of Monoboard Microcomputers: LAN tests, Serial Port tests, Line Printer Test, and SCSI device tests.

With the MVME167 and MVME187, there have been some changes in the display output when a communication device detects an error or break condition. The communication driver will report which channel a break is detected on. It will also report other types of errors to the display, displaying the channel number and type of error.

MVME167/MVME187 Board Test Menus

There are four separate test menus for the MVME167/MVME187 board. Each menu applies to a different function on the board, namely serial communications, LAN, SCSI, and printer port. Each function can be separately enabled or disabled through use of the **cf** command at the command line level. (See the MVME167/MVME187 Board Test Configuration section for more detail on **cf**.) To access the test menus for the MVME167/MVME187 board, type:

```
SA: bdtest
```

A display similar to the following appears:

```
Current Menu is /bdtest - "Tests by Board"

    1x7lp.0            - VME1x7 Printer Port Tests
    1x7sp.0            - VME1x7 COMM Port Tests
    1x7lan.0           - VME1x7 LAN Tests
    1x7scsi.0          - VME1x7 SCSI Tests
    mem.0              - Memory Test

SA:
```

5

The **bdtest** Main Menu varies from system to system, displaying those boards that SSID has found in the system. The display shown above is for a system which has only an MVME167/MVME187 board installed. Each of these menu items will be discussed in the next few pages.

MVME167/MVME187 1x7lp Test Menu

The test selection menu for the `1x7lp` is:

```
SA: bdtest/1x7lp.0 <CR>   (to access the printer port intermediate menu)

Current Menu is /bdtest/1x7lp.0 - "VME1x7 Printer Port Tests"

       llp0                 - VME1x7 Printer Port Tests
                             (Intermediate Menu)
```

The default tests selection menu for `llp0` is:

```
SA: bdtest/1x7lp.0/llp0 <CR> (to directly access the printer port test selection menu)
Selections for Test "VME1x7 Printer Port Tests"
    1) Polled loopback
    2) Interrupt loopback
    D) Display Error Log
    C) Clear Error Log
    z) Add/Remove from confid
```


1) Polled loopback

The printer port is tested by setting output signals to known states and verifying the input signals are in the expected state. This test requires an external printer loopback to execute correctly. See the loopback wiring information in *Appendix B.*

2) Interrupt loopback

The same as the Polled loopback, except that the expected interrupt is also checked. This test requires an external printer loopback to execute correctly. See the loopback wiring information in *Appendix B.*

D) Display Error Log

An error log has been provided that stores all error information for the Printer Port Tests. All errors print to the screen at the time of the error, and will also be stored in the error log. Although the error may have scrolled off the screen, the failure information can be viewed through the use of this display option. The only errors that cannot be logged are failures of the error logger itself, memory allocation, and semaphore allocation errors that would occur very early in the test execution.

C) Clear Error Log

This command clears the errors logged in the local error log during execution of tests. The system error log is not cleared by this command. In order to clear the system error log, the user must execute the **clear** system level command that will also give the option to clear this local error log.

z) Add/Remove from confid
This command allows the user to add or remove this test from the SSID scripts; e.g. confidence scripts.

The alternate test selection menu can be turned on by use of the **cf** command. To enable this menu, the "loopback mode" needs to be set to 0 and the "output only mode" needs to be set to 1 when prompted by **cf**. The alternate test selection menu for `llp0` is:

5

```
SA: bdtest/1x7lp.0/llp0 <CR> (to directly access the printer port test selection menu)
Selections for Test "VME1x7 Printer Port Tests"
    1) Polled output to printer
    2) Interrupt output to printer
    D) Display Error Log
    C) Clear Error Log
    z) Add/Remove from confid
```

1) Polled output to printer
The printer port is tested by output of the following test pattern to the printer:

`The quick brown fox jumps over the lazy dog.`

The BUSY and ACKNOWLEDGE signal are polled to determine when a character can be sent to the printer. If any error signals are detected during the test, a fatal error will be recorded. This includes paper errors.

2) Interrupt output to printer
The same as the Polled output to printer, except that the expected interrupt is also checked.

D) Display Error Log
An error log has been provided that stores all error information for the Printer Port Tests. All errors print to the screen at the time of the error, and will also be stored in the error log. Although the error may have scrolled off the screen, the failure information can be viewed through the use of this display option. The only errors that cannot be logged are failures of the error logger itself, memory allocation, and semaphore allocation errors that would occur very early in the test execution.

C) Clear Error Log

This command clears the errors logged in the local error log during execution of tests. The system error log is not cleared by this command. In order to clear the system error log, the user must execute the **clear** system level command that will also give the option to clear this local error log.

z) Add/Remove from confid

This command allows the user to add or remove this test from the SSID scripts; e.g. confidence scripts.

5

MVME167/MVME187 1x7sp Test Menu

The test selection menus for the `1x7sp` are:

```
SA: bdtest/1x7sp.0 <CR> (to access the Serial Port intermediate menu)

Current Menu is /bdtest/1x7sp.0 - "VME1x7 COMM Port Tests"

        lsp7      - VME1x7 Comm Port 7 Tests    (Intermediate Menu)
        lsp6      - VME1x7 Comm Port 6 Tests    (Intermediate Menu)
        lsp5      - VME1x7 Comm Port 5 Tests    (Intermediate Menu)
        lsp4      - VME1x7 Comm Port 4 Tests    (Intermediate Menu)
        lsp3      - VME1x7 Comm Port 3 Tests    (Intermediate Menu)
        lsp2      - VME1x7 Comm Port 2 Tests    (Intermediate Menu)
        lsp1      - VME1x7 Comm Port 1 Tests    (Intermediate Menu)
        lsp0      - VME1x7 Comm Port 0 Tests    (Intermediate Menu)

SA: bdtest/1x7sp.0/lsp3 <CR> (to directly access serial port 3's test selection menu)

Selections for Test "VME1x7 Comm Port 3 Tests"

    0) Execute tests 1-9
    1) Internal send/receive
    2) Internal baud rate
    3) Internal framing error
    4) Internal break detect
    5) External send/receive
    6) External baud rate
    7) External framing error
    8) External break detect
    9) External handshake
    a) Autodetect handshake
    c) Display CD2401 version
    d) Display detected handshake
    D) Display Error Log
    C) Clear Error Log
    z) Add/Remove from confid
```

1) Internal send/receive

Transmit data in internal loopback mode, receive it and compare it to what was transmitted. The test operates at 9600 baud, with no parity, eight bits per character, and one stop bit.

2) Internal baud rate

The same as "Internal send/receive" except that the test executes at each of the following baud rates: 50, 110, 300, 600, 1200, 2400, 4800, 7200, 9600, 19200, 38400, 56000, and 64000.

3) Internal framing error

Transmit data in internal loopback mode, receive it and check for a framing error. The test operates with transmitter set to 4800 baud, receiver set to 9600 baud, eight bits per character, odd parity, and one stop bit.

4) Internal break detect

Transmit a break in internal loopback mode, receive it and check for a break being detected. The test operates at 9600 baud, with no parity, eight bits per character, and one stop bit.

5) External send/receive

The same as "Internal send/receive" except that the test executes with an external loopback attached to the transition board. See the loopback wiring information in *Appendix B*.

6) External baud rate

The same as "Internal baud rate" except that the test executes with an external loopback attached to the transition board.

7) External framing error

The same as "Internal framing error" except that the test executes with an external loopback attached to the transition board.

8) External break detect

The same as "Internal break detect" except that the test executes with an external loopback attached to the transition board.

9) External handshake

Test external handshake connections. This test does one autodetect of external handshake connections and then uses that detected configuration to test against. A minimum of one handshake path must exist and not change for the duration of the test for the test to pass. This test requires an external loopback attached to the transition board.

a) Autodetect handshake

Detect and display current detected connections and the current state of the handshake connections. This test requires an external loopback attached to the transition board.

c) Display CD2401 version

Read and display the CD2401's version.

d) Display detected handshake

Display the detected handshake connections. This test requires an external loopback attached to the transition board.

D) Display Error Log

An error log has been provided that stores all error information for the Serial Port Tests. All errors print to the screen at the time of the error, and will also be stored in the error log. Although the error may have scrolled off the screen, the failure information can be viewed through the use of this Display option. The only errors that cannot be logged are failures of the error logger itself, memory allocation, and semaphore allocation errors that would occur very early in the test execution.

C) Clear Error Log

This command clears the errors logged in the local error log during execution of tests. The system error log is not cleared by this command. In order to clear the system error log, the user must execute the **clear** system level command that will also give the option to clear this local error log.

z) Add/Remove from confid

This command allows the user to add or remove this test from the SSID scripts; e.g. confidence scripts.

MVME167/MVME187 1x7lan Test Menu

The test selection menus for the `1x7lan` are:

```
SA: bdtest/1x7lan.0 <CR> (to access the LAN intermediate menu)
Current Menu is /bdtest/1x7lan.0 - "VME1x7 LAN Tests"
      lan                    - LAN Tests        (Intermediate Menu)
SA: bdtest/1x7lan.0/lan <CR>       (to directly access the LAN test selection menu)
Selections for Test "LAN Tests"
    1) LAN Addressing Function
    2) LAN Dump of Registers
    3) LAN Self Test
    4) LAN Receive Frame Area Ready
    5) LAN Internal Loopback
    6) LAN External Loopback
    7) LAN Command Chaining
    8) LAN Transceiver Loopback
    a) Display Ethernet Address in NVRAM
    t) Time Domain Reflectometry
    D) Display Error Log
    C) Clear Error Log
```

5

1) LAN Addressing Function

The first portion of this test reads NVRAM to determine what the assigned ethernet address is. If it does not fall within the range of addresses assigned to Motorola Ethernet products, a Fatal Error occurs. Next, the address found in NVRAM is written into the LAN chip. Lastly, the test checks that the value stored in the LAN Ethernet Address Register is the same as the address just written.

2) LAN Dump of Registers

The LAN Dump Test verifies that the contents of various LAN registers can be placed in a memory area specified by SSID. This verifies some of the data paths through the chip to RAM. This test also proves that the initialization of the LAN chip was successful. If the Dump test fails, it may mean that the memory pointers stored in the LAN chip are invalid. Running other LAN tests when the Dump test has failed may result in the LAN chip writing to invalid memory

5

locations. This problem would only occur if the user had changed the option settings to force SSID to continue on error. Normal SSID operation will halt testing if the Dump of Registers Test fails.

3) LAN Self Test

The LAN Self-Test will provide testing ROM, Parallel Registers, Bus Throttle Timers, and internally the serial subsystem. The Self-Test also causes the micro machine to issue an internal Diagnose command.

4) LAN Receive Frame Area Ready

The Receive Frame Area (RFA) Ready test verifies that activating the Receive Unit within the LAN chip prepares the receiver memory buffers. If successful, the receiver will interrupt the processor when the Receive Frame Area is ready. NOTE! Any user data received from an active LAN connection will be lost. The RFA Ready test should always be run prior to the Loopback Tests.

5) LAN Internal Loopback

During Internal Loopback tests, one packet of data is transmitted, looped back, and received by the LAN chip. The loopback of signals occurs within the Intel 82596 LAN chip itself. No data is propagated beyond the 82596 chip. The transmit clock is divided by 4 during this type of loopback test. The size of the data packet is 3 times the size of the receive and transmit buffers, in order to verify that the LAN chip can chain both receive and transmit buffers. If the received data does not match the transmitted data, an error occurs.

6) LAN External Loopback

During External Loopback tests, one packet of data is transmitted, looped back, and received by the LAN chip. The loopback of signals occurs within the Intel 82c501 chip, which is connected to the LAN chip itself. Note that this test verifies operation of the LAN circuitry's serial interface, but does not test all the way out to the DB15 LAN connector. There is no loopback cable or terminator of any type required for this test. No data is propagated beyond the 82c501 chip.

7) LAN Command Chaining Test

This test verifies that the Command Unit in the Intel 82596 LAN chip can independently execute two sequential commands in a Command Block List. This diagnostic will otherwise send individual commands to the LAN circuitry, verifying that each command has worked prior to moving on to the next command.

8) LAN Transceiver Loopback

During Transceiver Loopback tests, one self-addressed packet of data is transmitted, looped back, and received. A properly terminated transceiver is required for this test. Although any properly terminated, inactive network will suffice, the following equipment list is provided for the user's benefit:

Transceiver Cable
Thinnet Transceiver
Tee Connector for Thinnet Cable
Network Cable Terminators (need 2)

This test may fail if run on an active network; if a broadcast packet should happen to come across the network during execution of this loopback test, the test will fail. This test is not part of the confidence test suites.

a) Display Ethernet Address in NVRAM

This menu item prints the board's Ethernet address, which is stored in NVRAM, to the screen.

5

t) Time Domain Reflectometry

Time Domain Reflectometer (TDR) Testing is provided as a user-selectable menu option. This test is not part of the confidence test suite. It will determine the integrity of the Network cabling connected to the board under test. Using functions embedded in the Intel 82596 LAN chip itself, these conditions can be detected and displayed:

-- No Problem
-- Open on the Ethernet Cable
-- Short on the Ethernet Cable
-- Problem with the Transceiver or Transceiver Cable

CAUTION

It is STRONGLY recommended that all other LAN tests PASS prior to executing the TDR test on an active Network. This test sends a packet out onto the Network, and doing so with faulty hardware could give unpredictable results.

Because this is primarily a test of cabling outside of the system, errors are not logged for any of the three error conditions noted above. If, however, the 82596 returns an invalid condition code during this test, i.e., no condition bits set, an error will be logged.

For valid test results, the transceiver used must return transmitted data on the receive pair and activate the Carrier Sense Signal while transmitting. For transceivers that do not activate Carrier Sense, the test will indicate a problem with the Transceiver or Transceiver Cable, regardless of the condition of the cable.

The calculation of distance to the cable failure is dependent on the Network cable being standard trunk or transceiver cable with a Nominal Velocity of Propagation of approximately 79%. The distance values displayed are approximate.

The packet that is sent out on the Network is a misaligned packet, whose source and destination addresses are both equal to the LAN address of the board under test (from NVRAM). The format and content of the packet is controlled by the 82596 LAN chip itself.

D) Display Error Log

An error log has been provided that stores all error information for the LAN Tests. All errors print to the screen at the time of the error, and will also be stored in the error log. Although the error may have scrolled off the screen, the failure information can be viewed through the use of this Display option. The only errors that cannot be logged are failures of the error logger itself, and semaphore allocation errors that would occur very early in the test execution.

C) Clear Error Log

This command clears the errors logged in the local error log during execution of tests. The system error log is not cleared by this command. In order to clear the system error log, the user must execute the **clear** system level command that will also give the option to clear this local error log.

MVME167/MVME187 SCSI Device Tests

The MVME167/MVME187 SCSI Device tests verify the SCSI controller's operation by performing device tests through a software interface. This software interface is referred to as firmware.

MVME167/MVME187 Controller Tests

The `1x7scsi.0/controller` tests are used to reset the MVME167/ MVME187 NCR SCSI Controller chip or to reset the SCSI bus.

MVME167/MVME187 Hard Disk Tests

The Hard Disk Tests verify SCSI disk controller operation by performing nondestructive, read-only tests on the SCSI hard disk drives. The hard disks to be tested must be initialized (formatted or have an operating system file system on them). Refer to the operating system documentation for information on formatting. Destructive testing and format capability may be enabled from `cmdline` mode by setting the `Eval flag` to `1` using `cf 1x7scsi.0/scsi`*m* where *m* is the device number.

MVME167/MVME187 Tape Tests

The Tape Tests verify SCSI tape controller operation by performing write and read tests on the SCSI tape drives. A write-enabled tape is required.

MVME167/MVME187 9 Track Tape Tests

The SCSI 9 Track Tape Tests verify SCSI tape controller operation by performing write and read tests on SCSI 9 track tape drives. A write-enabled tape is required.

MVME167/MVME187 Floppy Disk Tests

The Floppy Disk Tests verify SCSI disk controller operation by performing nondestructive, read-only tests on the attached SCSI floppy disk drives. The floppy disks to be tested must be initialized (formatted or have an operating system file system on them). Refer to the operating system documentation for information on formatting using your operating system. Destructive testing and format capability may be enabled from `cmdline` mode by setting the `Eval flag` to `1` using `cf 1x7scsi.0/scsi`*m* where *m* is the device number.

MVME167/MVME187 CD-ROM Disk Tests

The CD-ROM Disk Tests verify SCSI disk controller operation by performing nondestructive, read-only tests on the attached SCSI CD-ROM disk drives.

MVME167/MVME187 Optical Memory Disk Tests

The Optical Memory Disk Tests verify SCSI disk controller operation by performing nondestructive, read-only tests on the attached SCSI Optical Memory disk drives. The Optical Memory disks to be tested must be initialized (formatted or have an operating system file system on them). Refer to the operating system documentation for information on formatting using your operating system. Destructive testing and format capability may be enabled from `cmdline` mode by setting the `Eval flag` to `1` using `cf 1x7scsi.0/scsi`*m* where *m* is the device number.

MVME167/MVME187 SCSI Controller Primary Menu

To access the MVME167/MVME187 test menus for the SCSI controller, type:

```
SA: 1x7scsi.0
```

A display similar to the following appears:

```
Current Menu is /bdtest/1x7scsi.0 - "VME1x7 SCSI Tests"
scsi7         - VME1x7 SCSI SCSI ID 7 Tests
scsi6         - VME1x7 SCSI SCSI ID 6 Tests
scsi5         - VME1x7 SCSI SCSI ID 5 Tests
scsi4         - VME1x7 SCSI SCSI ID 4 Tests
scsi3         - VME1x7 SCSI SCSI ID 3 Tests
scsi2         - VME1x7 SCSI SCSI ID 2 Tests
scsi1         - VME1x7 SCSI SCSI ID 1 Tests
scsi0         - VME1x7 SCSI SCSI ID 0 Tests
controller    - VME1x7 Controller Self Tests
SA:
```

MVME167/MVME187 SCSI Controller Self Test Menu

To execute the controller test suite, select `controller` from the `1x7scsi.0` menu and the following menu displays:

```
SA: controller<CR>                          (to access the controller menu)
Selections for Test "VME1x7 Controller Self Tests"
   B) Reset the SCSI BUS                    (DEBUG only)
   R) Reset the NCR SCSI Controller         (DEBUG only)
SA:
```

The following is a brief explanation of the tests:

R) Reset the SCSI BUS
This is primarily for debug and should be used only if the MVME167/MVME187 SCSI Bus fails to respond to SCSI commands.

R) Reset the NCR SCSI Controller
This is primarily for debug and should be used only if the MVME167/MVME187 fails to respond to SCSI commands.

MVME167/MVME187 SCSI Hard Disk Menu

To execute any hard disk test suite, select the appropriate SCSI ID from the **1x7scsi.0** menu and the following menu displays:

```
SA: scsi0<CR>                      (to access the first SCSI hard disk test)
Selections for Test "VME1x7 SCSI ID 0 Tests"
  1) Hard Disk Confidence Test(Reads first sector, last sector, then first sector)
  2) Read Entire Hard Disk        (Sequentially reads every sector on the disk)
  3) Hard Disk Random Read        (Randomly reads 5% sectors on the disk)
  4) Hard Disk Ping-Pong Read (Reads start, end, then start + 1, end - 1, etc.)
  5) Device Self Tests            (Executes device self tests)
  r) Read a Sector                (Allows reading of any sector; accepts
                                  hexadecimal or decimal input)
  b) Read a Block of Sectors      (Allows reading of any block of sectors; accepts
                                  hexadecimal or decimal input)
  c) Read Capacity                (Displays drive capacity in sectors)
  i) Inquiry                      (Displays drive vendor and model)
  s) Request Sense                (Reads and displays drive request sense information)
SA:
```

The following is a brief explanation of the tests:

1) **Hard Disk Confidence Test**
Read the first sector, the last sector, then the first sector on the disk. Each read is done twice and the results compared.

2) **Read Entire Hard Disk**
Reads the entire disk sequentially from sector zero to the last sector. Each read is done twice and the results compared.

3) **Hard Disk Random Read**
Reads randomly selected blocks of sectors. Approximately 5% of the available sectors are read. Each read is done twice and the results compared.

5

4) `Hard Disk Ping-Pong Read`
Reads the entire disk in the following sequence: sector zero, the last sector, sector one, sector (last - 1), sector two, sector (last - 2), and so on, until every sector on the disk has been read twice, once from each *direction*. Every read is done twice and the results compared.

5) `Device Self Tests`
This test issues a SCSI self test command. If the device fails its internal self tests, an error message is printed. Some devices do not support this command and will fail with an illegal command status.

r) `Read a Sector`
Requests user input of a sector number to be read, verifies that the sector number is valid, and if so, reads the sector, rereads the sector, and compares the results.

b) `Read a Block of Sectors`
Requests user input of a sector number at which to begin reading, verifies that these are valid sector numbers, then reads the block of sectors, rereads the block, and compares the results.

c) `Read Capacity`
Displays the drive capacity, in sectors. The display is in both hex and decimal.

NOTE

SCSI hard disk drives of the same type do not always have the same number of available sectors, but the numbers should be close in value (e.g. within 200 sectors of each other).

i) `Inquiry`
Displays the drive type, vendor ID, product ID, and product revision level obtained by performing a SCSI Inquiry command.

s) `Request Sense`
Displays the drive's request sense information. The information is obtained by performing a SCSI Request Sense command.

MVME167/MVME187 SCSI Tape Tests Menu

To execute any SCSI tape test suite, select the appropriate SCSI controller from the `1x7scsi.0` menu and the following menu displays:

```
SA: scsi4<CR>                          (to access the first SCSI tape test)
Selections for Test "VME1x7 SCSI ID 4 Tests"
(A write-enabled scratch tape is required)
  1) Tape Confidence Test              (Verifies presence of media and does rewind)
  2) Write/Append/Read Tape Files      (Write/Append/Read/Compare test)
  3) Write/Read to EOT                 (Write/Read/Compare test on entire tape)
  4) Read Entire Tape                  (Reads, without verification, all data on tape)
  5) Device Self Tests                 (Executes device self tests)
  r) Tape Retension                    (If supported by the drive, retensions media)
  e) Erase Tape                        (Removes ALL data from the tape then rewinds)
  b) Tape Block Data                   (Displays drive block size information)
  i) Inquiry                           (Displays drive vendor and model)
  s) Request Sense                     (Reads and displays drive request sense information)
  u) Unload and Eject the Media        (Unloads the tape)
SA:
```

The following is a brief explanation of the tests:

1) **`Tape Confidence Test`**
   Tests that the tape drive is ready and contains a tape. If so, a rewind is done.

2) **`Write/Append/Read Tape File tests`**
   At the beginning of media, two files are written. Each file consists of two blocks of data with one file mark. The tape is then rewound. Next, a space to End-Of-Data (EOD) is done. At EOD, a file with two blocks of data and a file mark is written or appended. The tape is then rewound. Now, each of the three files are read, the data blocks are verified, and the file mark is checked for. Next, EOD is checked for and the tape is rewound. If the test fails, an error message is printed.

3) **`Write/Read to EOT`**
   From the beginning of media, this test writes a file onto all the remaining media. The tape is then rewound and the data read and compared with the written data patterns. This requires a write-enabled tape.

4) **`Read Entire Tape`**
From the beginning of media, this test reads every file on the tape until either blank media or end-of-media is encountered. No data verification is done. This merely confirms that all the media is readable.

5) **`Device Self Tests`**
This test issues a SCSI self test command. If the device fails its internal self tests, an error message is printed. Some tape drives do not support this command and will fail with an illegal command status. For this reason, the confidence test scripts for tapes do *not* include this test.

r) **`Tape Retension`**
A SCSI Retension command is issued to the tape drive. On some devices, this requires over one hour to complete. On drives that do not support (or need) retensioning, such as the EXABYTE streaming cartridge tape drive, no tape movement takes place and the test simply passes.

e) **`Erase Tape`**
Beginning at the start-of-media, all tape contents are erased. A write-enabled tape is required. On some devices, this requires over one hour to complete.

b) **`Tape Block Data`**
Displays the tape drive's block size information obtained by performing a Read Block limits and a Mode Sense SCSI command.

i) **`Inquiry`**
Displays the drive type, vendor ID, product ID, and product revision level obtained by performing a SCSI Inquiry command.

s) **`Request Sense`**
Displays the drive's request sense information. The information is obtained by performing a SCSI Request Sense command.

u) **`Unload and Eject the Media`**
The tape is unloaded from the tape drive. If the drive supports tape ejection, then the tape is ejected from the drive. Tape unloading is accomplished by performing a SCSI Unload command.

MVME167/MVME187 SCSI 9 Track Tape Tests Menu

To execute any SCSI 9 track tape test suite, select the appropriate SCSI controller from the **1x7scsi.0/scsi*m*** menu and the following menu displays:

```
SA: 1x7scsi.0/scsim<CR>                  (to access the first SCSI 9 track tape test)

Selections for Test "VME1x7 SCSI ID m Tests"
(A write-enabled scratch tape is required)

   1) Tape Confidence Test               (Verifies presence of media and does rewind)
   2) Write/Append/Read Tape Files          (Write/Append/Read/Compare test)
   3) Write/Read to EOT                  (Write/Read/Compare test on entire tape)
   4) Read Entire Tape                   (Reads, without verification, all data on tape)
   5) Device Self Tests                  (Executes device self tests)
   r) Tape Retension                     (If supported by the drive, retensions media)
   e) Erase Tape                         (Removes ALL data from the tape then rewinds)
   b) Tape Block Data                    (Displays drive block size information)
   i) Inquiry                            (Displays drive vendor and model)
   s) Request Sense                      (Reads and displays drive request sense information)
   u) Unload and Eject the Media         (Unloads the tape)
SA:
```


Tests **1**, **3**, **4**, **5**, **r**, **e**, **b**, **i**, **s**, and **u** are identical to those explained in the *MVME167/MVME187 SCSI Tape Tests Menu* section. Test **2** is a little different and the following is a brief explanation of test **2**:

2) Write/Append/Read Tape File tests

At the beginning of media, two files are written. The first file consists of two blocks of data with one file mark. The last file consists of two blocks of data with two sequential file marks. The tape is then rewound. Next, the tape is spaced to the end of the two sequential file marks. Then the tape is backed up one file mark. The tape is now positioned at the End-Of-Data (EOD). At EOD, a file consisting of two blocks of data with two sequential file marks is written or appended. The tape is now rewound. Now, each of the three files are read, the data blocks are verified, and a file mark is checked for. Next, EOD is found by checking for the last file mark. The tape is then rewound. If the test fails, an error message is printed.

5

MVME167/MVME187 SCSI Floppy Disk Tests Menu

SSID can read, write, and format 5¼" and 3½" floppy disks in the following formats.

Table 5-3. Floppy Sizes and Formats Supported

Media Size	Description	Sectors Per Track	Cylinders	Capacity In Bytes
5¼"	Double sided, double density (PC-XT)	8	40	332 K
5¼"	Double sided, double density (PC-XT)	9	40	360 K
5¼"	Double sided, high density (PC-AT)	9	80	1.2 M
5¼"	Double sided, double density (Motorola dsdd5)	16	80	655 K
3½"	Double sided, double density (PC-AT)	9	80	737 K
3½"	Double sided, high density (PC-AT)	15	80	1.4 M
3½"	Double sided, extra density	36	80	2.9 M

In all read and write tests, a formatted floppy is required. SSID automatically detects the format of the disk. If the disk is unreadable in any of the appropriate formats listed in the above table, the test fails. During an SSID MVME167/MVME187 floppy disk format, you are asked which format to use. Be sure that the format you request matches the media density that you are formatting.

To execute any SCSI floppy disk test suite, select the appropriate SCSI controller from the `1x7scsi.0` menu and the following menu displays:

```
SA: scsi6<CR>                          (To access the first SCSI floppy disk test)

Selections for Test "VME1x7 SCSI ID 6 Tests"

  1) Floppy Disk Confidence Test  (Reads first sector, the last sector, then first)
  2) Read Entire Floppy Disk      (Sequentially reads every sector on the disk)
  3) Floppy Disk Random Read      (Randomly reads 5% sectors on the disk)
  4) Floppy Disk Ping-Pong Read   (Reads start, end, then start + 1, end - 1, etc.)
  5) Device Self Tests            (Executes device self tests)
  c) Read Capacity                (Determines capacity of disk)
  r) Read a Sector                (Allows reading of any sector; accepts
                                  hexadecimal or decimal input)
  b) Read a Block of Sectors      (Allows reading of any block of sectors;
                                  accepts hexadecimal or decimal input)
  i) Inquiry                      (Displays drive vendor and model)
  s) Request Sense                (Reads/displays drive request sense information)
SA:
```

Tests **1, 2, 3, 4, 5, r, b, i,** and **s** are identical to those explained in the *MVME167/MVME187 SCSI Hard Disk Menu* section. The following is a brief explanation of test **c**:

c) Read Capacity

Determines capacity of the floppy disk currently in the drive and displays it in decimal and hexadecimal. Requires a formatted floppy disk.

5

MVME167/MVME187 SCSI CD-ROM Menu

To execute any CD-ROM test suite, use the SSID **slctdev** command to select the CD-ROM for the MVME167/MVME187 SCSI ID address. Then select the appropriate SCSI controller from the `1x7scsi.`*n*`/scsim` menu and the following menu displays:

```
SA: 1x7scsi.0/scsim <CR>          (To access the SCSI CD-ROM test)
Selections for Test "VME1x7 SCSI ID m Tests"

  1) CD-ROM Confidence Test       (Reads first sector, last sector, then first sector)
  2) Read Entire CD-ROM           (Sequentially reads every sector on the disk)
  3) CD-ROM Random Read           (Randomly reads 5% sectors on the disk)
  4) CD-ROM Ping-Pong Read        (Reads start, end, then start + 1, end - 1, etc.)
  5) Device Self Tests            (Executes device self tests)
  r) Read a Sector                (Allows reading of any sector; accepts hexadecimal or
                                  decimal input)
  b) Read a Block of Sectors      (Allows reading of any block of sectors; accepts
                                  hexadecimal or decimal input)
  c) Read Capacity                (Displays drive capacity in sectors)
  i) Inquiry                      (Displays drive vendor and model)
  s) Request Sense                (Reads and displays drive request sense information)
  e) Unload and Eject the Media   (Ejects the CD-ROM)
  a) Audio Status                 (Displays audio playback status)
  d) Disk Information             (Displays CD-ROM disk information)
  p) Play Audio CD-ROM            (Plays an Audio CD-ROM)
  f) Audio Pause/Resume           (Pauses/Resumes Audio playback)

SA:
```

Tests **1, 2, 3, 4, 5, r, b, c, i,** and **s** are identical to those explained in the *MVME167/MVME187 SCSI Hard Disk Menu* section. The following is a brief explanation of tests **e, a, d, p,** and **f**:

`e) Unload and Eject the Media`
The CD-ROM is unloaded and ejected from the CD-ROM drive. CD-ROM unloading is accomplished by performing a Toshiba SCSI Unload and Eject command.

a) `Audio Status`
The current audio playback status is requested from the drive and displayed. The playback status is read by performing a Toshiba SCSI Read Audio Status command.

d) `Disk Information`
The current CD-ROM disk information is requested from the drive and displayed. The disk information is read by performing a Toshiba SCSI Read Disk Information command.

p) `Play Audio CD-ROM`
The number of tracks on the current CD-ROM is requested from the drive. The user is prompted for the starting and ending tracks to play. The starting and ending tracks are sent to the drive and it is told to play these tracks.

f) `Audio Pause/Resume`
If the CD-ROM drive is currently playing an audio track, then a Toshiba SCSI Audio Still command is sent to the drive. If the CD-ROM drive has been Stilled by a Audio Still command, then a Toshiba SCSI Audio Resume command is sent to the drive. If the CD-ROM drive is not in one of the preceding states, this command will just exit.

5

MVME167/MVME187 SCSI Optical Memory Menu

To execute any Optical Memory test suite, use the SSID **slctdev** command to select the Optical Memory for the MVME167/MVME187 SCSI ID address. Then select the appropriate SCSI controller from the `1x7scsi.0/scsim` menu and the following menu displays:

```
SA: 1x7scsi.0/scsim <CR>                    (To access the SCSI Optical Memory test)
Selections for Test "VME1x7 SCSI ID m Tests"

  1) Optical Memory Confidence Test  (Reads first sector, then last, then first)
  2) Read Entire Optical Memory      (Sequentially reads every sector on the disk)
  3) Optical Memory Random Read      (Randomly reads 5% sectors on the disk)
  4) Optical Memory Ping-Pong Read   (Reads start, end, start + 1, end - 1, etc.)
  5) Device Self Tests               (Executes device self tests)
  r) Read a Sector                   (Allows reading of any sector; accepts
                                     hexadecimal or decimal input)
  b) Read a Block of Sectors         (Allows reading of any block of sectors;
                                     accepts hexadecimal or decimal input)
  c) Read Capacity                   (Displays drive capacity in sectors)
  i) Inquiry                         (Displays drive vendor and model)
  s) Request Sense                   (Reads and displays drive request sense
                                     information)
  e) Unload and Eject the Media      (Ejects the Optical Media)
SA:
```

Tests **1, 2, 3, 4, 5, r, b, c, i,** and **s** are identical to those explained in the *MVME167/MVME187 SCSI Hard Disk Menu* section. The following is a brief explanation of test **e**:

`e) Unload and Eject the Media`
The Optical Media is unloaded and ejected from the Optical drive. Optical Media unloading is accomplished by performing a SCSI Stop Unit command.

MVME167/MVME187 Board Test Configuration

Each of the four major test functions, Serial Communications, LAN, SCSI, and Printer Port, can be separately enabled or disabled through use of the **cf** command in `cmdline` mode. The configuration relating specifically to the MVME167/MVME187 is shown below. Other boards have been deleted for simplicity. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf
/bdtest configuration
   1x7lp [0-1] = 1 ?             # MVME167/MVME187 Printer Port testing
   1x7sp [0-1] = 1 ?             # MVME167/MVME187 Serial Port testing
   1x7lan [0-1] = 1 ?            # MVME167/MVME187 LAN testing
   1x7scsi [0-1] = 1 ?           # MVME167/MVME187 SCSI testing
```


All MVME167/MVME187 board test configurations are shown below. Refer to the **cf** command in Appendix A for information on modifying these configurations.

```
> cf 1x7lp.0
/bdtest/1x7lp.0 configuration
   base [%0x10] = 0xffffce00 ?            # GCSR address
> cf 1x7lp.0/llp0
/bdtest/1x7lp.0/llp0 configuration
   Port is testable 0-False, 1-True [0,1] = 1
   Test in loopback mode 0-False [0,1] = 1
   Test in output only mode 0-False [0,1] = 0
   Columns per line to print [1-300] = 80
   Lines per page to print [1-1000] = 66
   Pages to print [0,1] = 1
   Use FormFeed 0-False [0,1] = 1
   Carriage Return after LineFeed 0-False [0,1] = 1
> cf 1x7sp.0
/bdtest/1x7sp.0 configuration
   base [%0x10] = 0xffffce00 ?            # GCSR address
> cf 1x7sp.0/lsp3
/bdtest/1x7sp.0/lsp3 configuration
   Port number [3] = 3
   Port is testable 0-False [0,1] = 1
   Port can do sync 0-False [0,1] = 0
   Port should driver clk 0-False [0,1] = 0
   Port should receive clk 0-False [0,1] = 0
   TX/RX encoding 0-NRZ,1-NRZI,2-Manchester [0,1,2] = 0
   Synchronous baud rate [300,1200,2400,4800,9600,19200] = 9600
```

```
> cf 1x7lan.0
/bdtest/1x7lan.0 configuration
   base [%0x10] = 0xffffce00 ?              # GCSR address
   level [0-7] = 4 ?                         # unused value
   mode [0] = 0 ?                            # cannot be changed by user
   warning msgs [0] = 0 ?                    # cannot be changed by user

> cf 1x7lan.0/lan
/bdtest/1x7lan.0/lan configuration
   xfer size [0-1024] = 1024 ?               # unused value

> cf 1x7scsi.0                               (To configure parameters for the VME1x7 SCSI port)
/bdtest/1x7scsi.0  configuration
  base [%0x10] = 0xffffce00 ?                # Base address
  level [0-7] = 4 ?                          # Default interrupt level
  SCSI ID [0-7] = 7 ?                        # SCSI ID for the NCR chip
  mode [0] = 0 ?                             # cannot be changed by user
  warning msgs [0] = 0 ?                     # cannot be changed by user
  Device timeout-secs [1-240] = 70 ?         # Set for time device takes to become ready
  Debug Flag [0-15] = 0 ?                    # Debug mode flag
```

To set the parameters for each individual device, use `cf 1x7scsi.0/scsim`:

```
> cf 1x7scsi.0/scsi0                         (To configure the first SCSI device)
/bdtest/1x7scsi.0/scsi0  configuration
  Ctrlr number[0] = 0 ?                      # scsi controller number
  phy-drive[0-7] = 0 ?                       # drive number
  Maximum sector size [1-10000] = 512 ?      # Used to set block size for tapes and
  sector/xfer [1-128] = 64 ?                 # Number of blocks to transfer
  W Data Pattern = 0x0 ?                     # Write/Read data pattern
  Device type flag [0] = 0 ?                 # Program device type flag
                                             # hard disk formatting
  Eval flag:Write/Expert Mode(0-Off 1-On) [0-1] = 0 ?# destructive test flag
  Conf flag:Warning messages(0-On 1-Off)[0-1] = 0 ?# To turn user warning about
                                                   # destructive disk tests on and off
  Write Type(0-1%,1-All,2-Random,3-Ping)[0-3] = 0 ?
                                             # (Selects the write test that will be
                                             # available on the expert menu)
```

> **CAUTION**
>
> **Setting the `Eval flag` variable to `1-On` changes the device test menu so that it will include destructive tests and expert commands.**

MVME167/MVME187 Hard Disk Expert Menu

If the `Eval flag` is set to `1-On`, the MVME167/MVME187 hard disk test menu is expanded to include destructive tests and menu items for closer disk control.

```
SA: 1x7scsi.0/scsim <CR>                  (To access the expert hard disk test)
Selections for Test "VME1x7 SCSI ID m Tests"
  1) Hard Disk Confidence Test            (Reads first sector, last sector, then first sector)
  2) Read Entire Hard Disk                (Sequentially reads every sector on the disk)
  3) Hard Disk Random Read                (Randomly reads 5% sectors on the disk)
  4) Hard Disk Ping-Pong Read             (Reads start, then end, then start + 1, then end - 1, etc.)
  5) Device Self Tests                    (Executes device self tests)
  r) Read a Sector                        (Allows reading of any sector; accepts hexadecimal or
                                           decimal input)
  b) Read a Block of Sectors              (allows reading of any block of sector; accepts hexadecimal
                                           or decimal input)
  d) Display Defect Lists                 (Displays manufacturer's and grown defect lists)
  v) Mode Sense/Select                    (Does SCSI Mode Sense, Mode Select test)
  w) User Selected Write Test (DESTRUCTIVE) (Write test as set by the device test configuration)
  f) Format Disk (DESTRUCTIVE)            (Format media - NOT COMPATIBLE
                                          WITH OPERATING SYSTEM)
  x) Re-Assign a Block  (DESTRUCTIVE)
                                          (Reassigns defective logical block and adds it to the
                                          grown defect list)
  c) Read Capacity                        (Displays drive capacity in sectors)
  i) Inquiry                              (displays drive vendor and model)
  s) Request Sense                        (Reads and displays drive request sense information)
  t) Test Unit Ready                      (Tests the drive's current state of readiness)
  m) Display Mode Sense                   (Displays the drive's Mode Sense data)
  z) Set Confid Test Script               (Modifies confid 5 script to enable DESTRUCTIVE
                                          testing)
SA:
```

Tests **1, 2, 3, 4, 5, r, b, c, i,** and **s** are identical to those explained in the *MVME167/MVME187 SCSI Hard Disk Menu* section. The following describes the expert and destructive tests:

`t) Test Unit Ready`
A SCSI Test Unit Ready command is performed on the drive. If the drive is currently ready, this test will pass. If the drive is **NOT** ready, this test will fail, displaying information about the failure. *NOTE: This test may fail one time and pass every time thereafter. This is the nature of a SCSI Test Unit Ready command. This command only indicates a drive's ready state.*

5

`m) Display Mode Sense`
Displays the Mode Sense information that is read from the drive. The user is prompted for the type of Mode Sense information that is to be displayed. There are four types of mode sense information; they are Current values, Changeable values, Default values, and Saved values. Note: Some drives do not support all four types of mode sense data and will cause this test to fail.

`d) Display Defect Lists`
This command displays the manufacturer's and the grown media defect lists contained on the hard disk. The defect lists are requested in one of three types of formats. The user is prompted to select the format type that will be used to request the defect lists. The three types of formats are as follows:

1. **Block Format** - This format as defined by **SCSI-2** is a four-byte defective block address that contains the defect. This format is **not** supported by all hard disk drives and drives that do not support this format, the display defect list command will automatically change to the **Bytes From Index** format.
2. **Physical Sector Format** - This format as defined by **SCSI-2** is an eight-byte sector defect location. Each defect location is comprised of a three-byte cylinder number of the defect, a one-byte head number of the defect, and a four-byte defect sector number.
3. **Bytes From Index Format** - This format as defined by **SCSI-2** is an eight-byte defect location. Each defect location is comprised of a three-byte cylinder number of the defect, a one-byte head number of the defect, and a four-byte defect bytes from index.

The manufacturer's defect list, which is called the **Primary Defects List,** is requested and displayed first. Next, the grown defect list, which is called the **Grown Defects List,** is requested and displayed. The user is allowed to exit this command by entering an **E** or **e** during format type selection or by entering a **Q** or **q** after the **Depress RETURN or ENTER key to continue** message.

v) `Mode Sense/Select`
A SCSI Mode Sense command is performed on the drive. The received Mode Sense information is then used to perform a Mode Select command. Next, another SCSI Mode Sense command is performed on the drive. The Mode Sense information received on the first SCSI Mode Sense command is compared to the Mode Sense information received from the second Mode Sense command. The test will fail if there were changes to the second Mode Sense information received.

5

w) *User Selected Write Test*
This test and the test name will be set to the type of test as selected in the device test configuration variable **Write Type** (see the board test configuration section for this variable). The **Write Type** variable can be set to one of four values. The **Write Type** value will determine the name of this test and the type of test this command will run. The test name and the test function is defined as follows for the **Write Type** values:

0 - **Write Last 1%** **(DESTRUCTIVE)**
This test will write then read and verify approximately the last 1% of the sectors on the hard disk. Though this is a *quick* write-verification, it is still *extremely dangerous. Use extreme caution!*

1 - **Write All Sectors** **(DESTRUCTIVE)**
This test will sequentially write then read and verify every sector on the disk. *This test destroys all data on the disk. Use extreme caution!*

2 - **Random 50% Write** **(DESTRUCTIVE)**
This test will randomly select 50% of the sectors to write then read and verify. *This test randomly destroys data on the disk. Use extreme caution!*

3 - **Ping-Pong Write** **(DESTRUCTIVE)**
This test will write, read and verify the entire disk in the following sequence: sector zero, the last sector, sector one, sector (last - 1), sector two, sector (last - 2), and so on, until every sector on the disk has been written and read, once from each *direction. This test destroys all data on the disk. Use extreme caution!*

f) `Format Disk` `(DESTRUCTIVE)`
This test formats the entire hard disk. Before the format begins, the user is asked if they would like to retain the grown list of defects. If the user selects to *not* retain the grown list of defects, the sectors listed in the grown defects list will be returned to the usable sector list, the grown defects list will be zeroed, and the hard disk will be formated as perfect media. A perfect media

format is where bad sector mapping is automatically handled by the disk controller. If the user selects to retain the grown list of defects, the disk controller will be instructed to keep the grown list of defects. The controller will automatically redirect the sectors list in the grown list of defects. *This test destroys all data on the disk. Use extreme caution!*

5

CAUTION

This format is not compatible with any operating system. It should be used only as a diagnostic procedure.

`x) Re-Assign a Block` (DESTRUCTIVE)

Requests that the TARGET hard disk drive reassign the user-entered defective logical block to an area on the logical unit reserved for this purpose and to record the defective logical block in the grown defect list if such a list is supported. *This can destroy data on the disk. No attempt is made to recover data from the reassigned block. Use extreme caution!*

`z) Set Confid Test Script`

Modifies `confid 5` (Continuous Running Intensive I/O Check) test script for this hard disk to include destructive disk tests. The destructive selection is TOGGLED each time this selection is executed. If destructive mode is enabled, the following message is displayed:

```
CAUTION: Destructive disk tests for drive n have been added
         to confidence and fault test scripts!!!
```

If destructive mode is disabled, the following message is displayed:

```
DRIVE n TEST SCRIPTS ARE NOW READ ONLY
```

Destructive test WARNING messages are disabled during `confid` 5 testing to allow unattended destructive tests.

MVME167/MVME187 Tape Drive Expert Menu

If the `Eval flag` is set to `1-On`, the MVME167/MVME187 tape drive test menu is expanded to include menu items for closer tape control.

```
SA: 1x7scsi.0/scsim <CR>              (To access the expert tape drive test)
Selections for Test "VME1x7 SCSI ID m Tests"
(A write-enabled scratch tape is required)

  1) Tape Confidence Test             (Verifies presence of media and does rewind)
  2) Write/Append/Read Tape Files     (Write/Append/Read/Compare test)
  3) Write/Read to EOT                (Write/Read/Compare test on entire tape)
  4) Read Entire Tape                 (Reads, without verification, all data on tape)
  5) Device Self Tests                (Executes device self tests)
  r) Tape Retension                   (If supported by the drive, retensions media)
  e) Erase Tape                       (Removes ALL data from the tape then rewinds)
  b) Tape Block Data                  (Displays drive block size data)
  i) Inquiry                          (Displays drive vendor and model)
  m) Display Mode Sense               (Displays the drive's Mode Sense data)
  s) Request Sense                    (Reads and displays drive request sense
                                      information)
  t) Test Unit Ready                  (Tests the drive's current state of readiness)
  u) Unload and Eject the Media       (Unloads the tape)
SA:
```

The added tests are **m** and **t**. These tests are identical to the tests that are explained in the *MVME167/MVME187 SCSI Hard Disk Expert Menu* section.

5

5

MVME167/MVME187 9 Track Tape Drive Expert Menu

If the `Eval flag` is set to `1-On`, the MVME167/MVME187 9 track tape drive test menu is expanded to include menu items for closer tape control.

```
SA: 1x7scsi.0/scsim <CR>                (To access the expert 9 track tape drive test)
Selections for Test "VME1x7 SCSI ID m Tests"
(A write-enabled scratch tape is required)
   1) Tape Confidence Test              (Verifies presence of media and does rewind)
   2) Write/Append/Read Tape Files      (Write/Append/Read/Compare test)
   3) Write/Read to EOT                 (Write/Read/Compare test on entire tape)
   4) Read Entire Tape                  (Reads, without verification, all data on tape)
   5) Device Self Tests                 (Executes device self tests)
   r) Tape Retension                    (If supported by the drive, retensions media)
   e) Erase Tape                        (Removes ALL data from the tape then rewinds)
   b) Tape Block Data                   (Displays drive block size data)
   i) Inquiry                           (Displays drive vendor and model)
   m) Display Mode Sense                (Displays the drive's Mode Sense data)
   s) Request Sense                     (Reads and displays drive request sense
                                        information)
   t) Test Unit Ready                   (Tests the drive's current state of readiness)
   u) Unload and Eject the Media        (Unloads the tape)
SA:
```

The added tests are **m** and **t**. These tests are identical to the tests that are explained in the *MVME167/MVME187 SCSI Hard Disk Expert Menu* section.

MVME167/MVME187 Floppy Disk Expert Menu

If the `Eval flag` is set to `1-On`, the MVME167/MVME187 floppy drive test menu is expanded to include destructive tests and menu items for closer disk control.

```
SA: 1x7scsi.0/scsim <CR>                 (To access the expert floppy disk test)
Selections for Test "VME1x7 SCSI ID m Tests"
   1) Floppy Disk Confidence Test        (Reads first sector, last sector, then first sector)
   2) Read Entire Floppy Disk            (Sequentially reads every sector on the disk)
   3) Floppy Disk Random read            (Randomly reads 5% sectors on the disk)
   4) Floppy Disk Ping-Pong read         (Reads start, end, then start + 1, end - 1, etc.)
   5) Device Self Tests                  (Executes device self tests)
   c) Read Capacity                      (Determines capacity of disk)
   r) Read a Sector                      (Allows reading of any sector; accepts
                                         hexadecimal or decimal input)
   b) Read a Block of Sectors            (Allows reading of any block of sectors;
                                         accepts hexadecimal or decimal input)
   w) Write All Sectors (DESTRUCTIVE)
                                         (Sequentially write then read every sector on the
                                         disk)
   f) Format Floppy (DESTRUCTIVE)        (Format media - NOT COMPATIBLE WITH
                                         OPERATING SYSTEM)
   i) Inquiry                            (Displays drive vendor and model)
   s) Request Sense                      (Reads and displays drive request sense
                                         information)
   t) Test Unit Ready                    (Tests the drive's current state of readiness)
   m) Display Mode Sense                 (Displays the drive's Mode Sense data)
SA:
```

The added tests are **t**, **m**, **w**, and **f**. Tests **t**, **m**, and **w** are identical to the tests that are explained in the *MVME167/MVME187 SCSI Hard Disk Expert Menu* section. Test **f** is explained as follows:

f) Format Disk (DESTRUCTIVE)
This test offers the user a menu of supported floppy formats and then formats the floppy as requested. A write-enabled floppy disk is required. This *destroys all data* on the floppy.

If a 5¼" floppy drive is attached to the MVME167/MVME187, the following menu of available formats is displayed:

```
SA: f                            (To format the floppy diskette)
Select the format type to use:
   0) 5 1/4" PC-XT 9 secs/trk - 360 kbytes
   1) 5 1/4" PC-XT 8 secs/trk - 332 kbytes
   2) 5 1/4" dsdd5 - Motorola format - 655 kbytes
   3) 5 1/4" PC-AT - 1.2 Mbytes
   e) - Exit the format command
Your selection: 0 ?              (Select 0-3, or e to exit <CR>)
```

If a 3½" floppy drive is attached to the MVME167/MVME187, the following menu of available formats is displayed:

```
SA: f                            (To format the floppy diskette)
Select the format type to use:
   0) 3 1/2" Low density - 720 kbytes
   1) 3 1/2" High density - 1.4 Mbytes
   2) 3 1/2" Extra density - 2.9 Mbytes
   e) - Exit the format command
Your selection: 0 ?              (Select 0-2, or e to exit <CR>)
```

MVME167/MVME187 CD-ROM Drive Expert Menu

If the `Eval flag` is set to `1-On`, the MVME167/MVME187 CD-ROM drive test menu is expanded to include menu items for closer disk control.

```
SA: 1x7scsi.0/scsim <CR>            (To access the expert CD-ROM test)
Selections for Test "VME1x7 SCSI ID m Tests"
   1) CD-ROM Confidence Test        (Reads first sector, last sector, then first sector)
   2) Read Entire CD-ROM            (Sequentially reads every sector on the disk)
   3) CD-ROM Random Read            (Randomly reads 5% sectors on the disk)
   4) CD-ROM Ping-Pong Read         (Reads start, then end, then start + 1, end - 1, etc.)
   5) Device Self Tests             (Executes device self tests)
   r) Read a Sector                 (Allows reading of any sector; accepts
                                    hexadecimal or decimal input)
   b) Read a Block of Sectors       (Allows reading of any block of sectors;
                                    accepts hexadecimal or decimal input)
   c) Read Capacity                 (Displays drive capacity in sectors)
   i) Inquiry                       (Displays drive vendor and model)
   m) Display Mode Sense            (Displays the drive's Mode Sense data)
   s) Request Sense                 (Reads and displays drive request sense information)
   t) Test Unit Ready               (Tests the drive's current state of readiness)
   e) Unload and Eject the Media    (Ejects the CD-ROM)
   v) Mode Sense/Select             (Does SCSI Mode Sense, Mode Select test)
   a) Audio Status                  (Displays audio playback status)
   d) Disk Information              (Displays CD-ROM disk information)
   p) Play Audio CD-ROM             (Plays an Audio CD-ROM)
   f) Audio Pause/Resume            (Pauses/Resumes Audio playback)
SA:
```

The added tests are **m**, **t**, and **v**. These tests are identical to the same tests that are explained in the *MVME167/MVME187 SCSI Hard Disk Expert Menu* section.

5

MVME167/MVME187 Optical Memory drive expert menu

If the `Eval flag` is set to `1-On`, the MVME167/MVME187 Optical Memory drive test menu is expanded to include menu items for closer disk control.

```
SA: 1x7scsi.0/scsim <CR>                   (To access the expert Optical Memory test)
Selections for Test "VME1x7 SCSI ID m Tests"

   1) Optical Memory Confidence Test        (Reads first sector, then last, then first)
   2) Read Entire Optical Memory            (Sequentially reads every sector on the disk)
   3) Optical Memory Random Read            (Randomly reads 5% sectors on the disk)
   4) Optical Memory Ping-Pong Read         (Reads start, end, start + 1, end - 1, etc.)
   5) Device Self Tests                     (Executes device self tests)
   r) Read a Sector                         (Allows reading of any sector; accepts
                                            hexadecimal or decimal input)
   b) Read a Block of Sectors               (Allows reading of any block of sectors;
                                            accepts hexadecimal or decimal input)
   c) Read Capacity                         (Displays drive capacity in sectors)
   d) Display Defect Lists                  (Displays manufacturers and grown defect lists)
   w) User Selected Write Test (DESTRUCTIVE) (Write test as set by the device test configuration)
   f) Format Disk (DESTRUCTIVE)             (Format media - NOT COMPATIBLE
                                            WITH OPERATING SYSTEM)
   x) Re-Assign a Block   (DESTRUCTIVE)
                                            (Reassigns defective logical block and adds it to the
                                            grown defect list)
   v) Mode Sense/Select                     (Does SCSI Mode Sense, Mode Select test)
   i) Inquiry                               (Displays drive vendor and model)
   s) Request Sense                         (Reads and displays drive request sense information)
   t) Test Unit Ready                       (Tests the drive's current state of readiness)
   m) Display Mode Sense                    (Displays the drive's Mode Sense data)
   e) Unload and Eject the Media            (Ejects the Optical media)
   z) Set Confid Test Script                (Modifies confid 5 script to enable
                                             DESTRUCTIVE testing)

SA:
```

The added tests are **d**, **m**, **t**, **w**, **v**, **f**, **x**, and **z**. These tests are identical to the same tests that are explained in the *MVME167/MVME187 SCSI Hard Disk Expert Menu* section.

MVME167/MVME187 Error Messages

Printer Port Test Error Messages

Nearly all Printer port test errors are logged to an error log.

The most likely cause for each error (or group of errors) is described in an indented paragraph that follows the error messages.

The `YYYY` in the following error messages would be replaced by a value specific to that error.


```
System Error: Unable to allocate system's memory.
System Error: Unable to initialize error log.
System Error: Unable to record error into error log.
Error: Request for 0xYYYY bytes of system memory failed.
Error: Request for system semaphore failed.
```

All of these errors indicate that the SSID operating system cannot supply the requested resource.

```
Internal code error. The error number of YYYY is invalid.
Error: Input buffer is empty.
Error: Unable to setup printer BUSY interrupt handler.
Error: Unable to setup printer Paper Error interrupt handler.
Error: Unable to setup printer SELECT interrupt handler.
Error: Unable to setup printer FAULT interrupt handler.
Error: Unable to setup printer Acknowledge interrupt handler.
Error: Unable to release printer BUSY interrupt handler.
Error: Unable to release printer Paper Error interrupt handler.
Error: Unable to release printer SELECT interrupt handler.
Error: Unable to release printer FAULT interrupt handler.
Error: Unable to release printer Acknowledge interrupt handler.
Internal ERROR:Invalid encoded irq handler error message
0xYYYY
```

An invalid parameter has been passed between functions in the test program. This could indicate the test program or its data has been corrupted. Possible causes: bad memory or a bad data transfer from another device in the system has corrupted the memory.

```
Error: Test timed out waiting for printer to become ready after
printer reset.
Error: Unexpected status from printer reset.
Error: Test timed out waiting for printer to become ready.
Error: Printer not ready.
```

```
Error: Test timed out waiting for printer ACKNOWLEDGE.
Error: Printer never became ready.
Error: Printer error occured during printing.
Error: Test timed out waiting for printer to finish current
page.
```

The printer is doing one of the above unexpectedly. The printer having a paper out error or paper path error could cause some of these errors.

```
Error: The PCC2 printer interface is indicating an unexpected
interrupt.
Error: The BUSY bit is not a one as expected.
Error: The Paper Error bit is not a one as expected.
Error: The SELECT bit is not a one as expected.
Error: The FAULT bit is not a one as expected.
Error: The Acknowledge bit is not a one as expected.
Error: The Interrupt Status bit is not a one as expected.
Error: The BUSY bit is not a zero as expected.
Error: The Paper Error bit is not a zero as expected.
Error: The SELECT bit is not a zero as expected.
Error: The FAULT bit is not a zero as expected.
Error: The Acknowledge bit is not a zero as expected.
Error: The Interrupt Status bit is not a zero as expected.
Error: An unexpected BUSY interrupt occured.
Error: An unexpected SELECT interrupt occured.
Error: An unexpected Paper Error interrupt occured.
Error: An unexpected FAULT interrupt occured.
Error: An unexpected Acknowledge interrupt occured.
Error: An expected BUSY interrupt did not occur.
Error: An expected SELECT interrupt did not occur.
Error: An expected Paper Error interrupt did not occur.
Error: An expected FAULT interrupt did not occur.
Error: An expected Acknowledge interrupt did not occur.
```

A loopback is not connected, there is a bad loopback or incorrect loopback, or there is a hardware failure. In the case of a printer being attached, the printer is doing one of the above unexpectedly.

```
Error: The Printer interrupt status bit is not set.
Error: The BUSY interrupt pending bit is not set.
Error: The Paper Error interrupt pending bit is not set.
Error: The SELECT interrupt pending bit is not set.
Error: The FAULT interrupt pending bit is not set.
Error: The Acknowledge interrupt pending bit is not set.
```

A hardware failure.

Serial Port Test Error Messages

Nearly all Serial port test errors are logged to an error log.

The most likely cause for each error (or group of errors) is described in an indented paragraph that follows the error messages.

The `YYYY` in the following error messages would be replaced by a value specific to that error.

```
Error: Request to own port failed.
```

The port on which the test is requested is the SSID console port, or host port when in concurrent mode, so a test cannot be executed against it.

A less likely cause is an invalid parameter has been passed between functions in the test program. This could indicate the test program or its data has been corrupted. Possible causes: bad memory or a bad data transfer from another device in the system has corrupted the memory.

```
Error: Unable to init async defaults.
Error: Unable to init internal loopback mode.
Error: Request to release port failed.
Error: Unable to clear status.
Error: Unable to get status.
Error: Unable to init transmitter and/or receiver to YYYY baud.
Error: Unable to init even parity, 1 start/stop bit(s), and
8 bit character.
Error: Unable to init tx to YYYY baud.
Error: Unable to init rx to YYYY baud.
Error: Unable to init Break sequence.
```

An invalid parameter has been passed between functions in the test program. This could indicate the test program or its data has been corrupted. Possible causes: bad memory or a bad data transfer from another device in the system has corrupted the memory.

```
Error: Multiple outputs are connected together on port YYYY.
Possible causes of the failure might be:
a bad loopback, an incorrect loopback, or a hardware failure.
```

As stated in the error message, a bad loopback, an incorrect loopback, or a hardware failure.

```
Error: No handshake connections detected.
```

A loopback is not connected, there is a bad loopback or incorrect loopback, or there is a hardware failure.

```
Error: Port YYYY, current handshake connections do not match the
orginal:
Original:
   CTS  connects to  RTS.   Current signal state of  CTS   is
   NOT ASSERTED.
   DCD  connects to  DTR.   Current signal state of  DCD   is
   NOT ASSERTED.
Current:
   CTS  connects to  UNKN.  Current signal state of  CTS   is
   NOT ASSERTED.
   DCD  connects to  UNKN.  Current signal state of  DCD   is
   NOT ASSERTED.
```

The loopback has been removed during testing, the loopback is intermittent, or there is a hardware failure.

```
Error: YYYY unexpected Framing error(s) occured.
Error: YYYY unexpected Break(s) detected.
```

A hardware failure or an intermittent loopback.

```
Error: YYYY unexpected Parity error(s) occured.
Error: YYYY unexpected Overrun error(s) occured.
```

A hardware failure.

```
Error: Test timed out waiting for data, 0 characters received.
```

If the count is zero characters, check to make sure the loopback is connected. Causes are a bad loopback or a hardware failure.

```
Error: Data miscompare: tx data 0x54, rx data 0x53, buffer
offset 0.
```

A hardware failure.

```
Error: Test failed at YYYY baud.
```

When the internal and external rate test fails, this error message will indicate the test's baud rate.

```
Error: Expected Framing error not detected.
Error: Expected Break not detected.
```

An intermittent loopback, a missing loopback, or a hardware failure.

```
Error: Internal code error port number out of range.
Error: Internal code error program trying to queue greater than
16 characters.
Internal code error. The error number of YYYY is invalid.
```

An invalid parameter has been passed between functions in the test program. This could indicate the test program or its data has been corrupted. Possible causes: bad memory or a bad data transfer from another device in the system has corrupted the memory.

```
Error: Initializing work structure failed.
t2401_getwrk(1) memory request for 0xYYYY bytes failed.
Error: Request for system semaphore failed.
System Error: Unable to allocate system's memory.
System Error: Unable to initialize error log.
Error log not available.
System Error: Unable to record error into error log.
```

All of these errors indicate that the SSID operating system cannot supply the requested resource.

5

LAN Test Error Messages

Nearly all LAN errors are logged to an error log. Each logged error is assigned an error code. In the following list, the error code in hexadecimal is followed by the error text. The error code and the error text make up the whole error message.

The most likely cause for each error (or group of errors) is described in an indented paragraph that follows the error messages.

```
Unable to Initialize LAN Error Log.
Unable to Allocate LAN Intr Semaphore.
0xfffffff4, LAN: Mem couldn't be allocated.
```

Indicates that the named system resource was not available, or was malfunctioning. Possible causes are too many tests running, or defective MVME167/MVME187 board. These three errors are those which, by their nature, may prevent their being logged.

```
0xfffffff3, LAN: can't claim interrupt vector.
0xfffffff2, LAN: can't relinquish interrupt vec.
```

Indicates that an error occurred while SSID was attempting to set up one of the two LAN interrupts. The most probable cause is a defective MVME167/MVME187 board.

```
0xfffffff1, Freecore failed in release_lan_mem!
```

Indicates that an error occurred while the LAN test was trying to relinquish (free) the system memory it had used. The memory which was not freed will not be tested by the system memory test until the malfunction is corrected. Resetting the system is strongly suggested prior to beginning another pass of SSID. The most probable cause of this failure is a defective MVME167/MVME187 board.

```
Ethernet Address in Hex: xxxxxxxyyyyy
```

This message does not always indicate an error, but is printed as part of the error message for incorrect Ethernet Address. The first 7 digits of this hexadecimal number are uniformly assigned to all Motorola Ethernet products, and will be 08003E2. The final 5 digits are unique for each board. All twelve digits are user-programmable through the use of the Rom Debugger CNFG command.

```
0xfffffff0, Ethernet Address may be incorrect.
            Ethernet Address not Motorola assigned.
            Use ROM Debugger CNFG command fix it.
```

These messages, which appear together, indicate that the first 7 digits of the Ethernet Address are not 08003E2, as assigned. This may indicate that NVRAM on the MVME167/MVME187 has not been initialized, or is defective. If the customer has intentionally modified his/her address, it will be necessary to either use **cf** to configure the LAN test out, or temporarily modify the Ethernet Address to begin with 08003E2.


```
0xffffffef, Invalid System Configuration Pointer.
0xffffffee, lan_cuc_cmd_write: Command out of limits...
0xffffffe9, lan_ruc_cmd_write: Command out of limits...
```

Indicates that an SSID code variable has been corrupted. In both cases, the most probable cause is the memory into which SSID itself has loaded.

```
0xffffffd4, Warning: it took 2 tries to free memory.
```

Indicates that when the LAN test tried to return memory to the kernel's free memory pool, the first attempt failed. The second attempt was successful. This is only a warning, and subsequent testing should not be affected.

```
0xffffffed, No interrupt after Channel Attention.
0xffffffec, No interrupt received after Address Setup.
0xffffffeb, Init failed, BUSY byte never zeroed.
0xffffffea, LAN command word always busy.
0xffffffe8, Lan chip couldn't accept Address Setup.
0xffffffe7, Lan chip didn't complete Address Setup.
0xffffffe6, Write of Ethernet Address to LAN failed.
0xffffffe5, lan_dump: Status ok bit not 0.
0xffffffe4, lan_dump: Dump never completed.
0xffffffe3, Dump Addr not found in Dumped Data.
0xffffffe2, Addr of System Control Block not found.
0xffffffe1, LAN Selftest never completed.
0xffffffe0, LAN Selftest, selftest bit failure.
0xffffffdf, LAN Selftest, diagnose bit failure.
0xffffffde, LAN Selftest, bus timer bit failure.
0xffffffdd, LAN Selftest, register result bit failure.
0xffffffdc, LAN Selftest, rom test bit failure.
```

```
0xffffffdb, Receive Unit Command didn't complete.
0xffffffda, Receive Unit Status Indicates Not Ready.
0xffffffd9, Incorrect Receiver Values in Dump Area.
0xffffffd8, Configuration for Loopback test failed.
0xffffffd7, Data Received does not match Transmitted Data.
            At address = xxxxxxxx, expected data was: xx
                                   actual data was: xx
0xffffffd6, Receiver not ready for Loopback test.
0xffffffd5, No Interrupt after data transmission.
0xffffffd3, Sysbus Byte incorrect in dumped data.
0xffffffd2, Transmit Command did not complete correctly.
0xffffffd1, Chained Commands did not complete correctly.
0xffffffd0, Chip reconfigure did not complete.
0xffffffcf, TDR Test results are invalid.
```


Each of the messages listed above indicates a malfunction in the LAN circuitry of the MVME167/MVME187. A fairly thorough knowledge of the LAN chip itself is required to understand the full meaning of each message, but the conclusion to be drawn from each is the same. For any of these errors, the most probable cause is a defective MVME167/MVME187 board.

SCSI Test Error Codes and Messages

The SCSI test modules print out error messages when unexpected test conditions are detected. If a SCSI test detects an error, the contents of the SCSI sense packet and the English translation of the error code are both displayed. The following paragraphs describe the SCSI sense packet and the error messages:

SCSI Sense Packet

The contents of this packet display whenever an error or recovered error condition is encountered by a SCSI test. The English translation of the error code also displays. The following is a typical example of the SCSI Sense display:

```
/bdtest/1x7scsi.0/scsim First pass read error                  LINE 1

/bdtest/1x7scsi.0/scsim Fatal Read Error                       LINE 2

    At block 221219 (0x36023),                                 LINE 3
    0x03, Non-recoverable media error                          LINE 4
    Sense code: 0x11, Sense qual: 0x0                          LINE 5

/bdtest/1x7scsi.0/scsim
    Read sectors 221219-221283 (0x36023-0x36063):failed        LINE 6
```


where:

LINE 1
: indicates the SCSI device that failed and whether the failure occurred on the first read, second read, write, or read following write.

LINE 2
: indicates the SCSI device that failed and whether the failure is fatal or recoverable.

LINE 3
: contains the logical block number of the error.

LINE 4
: contains the SCSI Sense Key and English description of the error that occurred.

LINE 5
: contains the additional sense code and sense qualifier returned for the error.

LINE 6
: indicates the SCSI device that failed, the subtest that failed and the range of sectors being accessed, in both decimal and hexadecimal.

The following table gives a further explanation of the SCSI Sense Key found on LINE 4. Refer to an SCSI Specification or the product specification for the drive that failed to interpret the meaning of the additional sense code and sense qualifier found on line 5.

Table 5-4. MVME167/MVME187 SCSI Sense Keys

Sense Key (Hex)	Error Description
$00	**NO SENSE.** Indicates that there is no specific sense key information to be reported for the designated logical unit. This would be the case for a successful command or a command that received CHECK CONDITION or COMMAND TERMINATED status because one of the filemark, END-OF-MEDIA, or ILLEGAL-LENGTH-INDICATOR bits is set to one.
$01	**RECOVERED ERROR.** Indicates that the last command completed successfully with some recovery actions performed by the target. Details may be determinable by examining the additional sense bytes and the information field. When multiple recovered errors occur during one command, the choice of which error to report (i.e., first, last, most severe) is device specific.
$02	**NOT READY.** Indicates that the logical unit addressed cannot be accessed. Operator intervention may be required to correct this condition.
$03	**MEDIA ERROR.** Indicates that the command terminated with a non-recovered error condition that was probably caused by a flaw in the medium or an error in the recorded data. This sense key may also be returned if the target is unable to distinguish between a flaw in the medium and a specific hardware failure (sense key $04).
$04	**HARDWARE ERROR.** Indicates that the target detected a non-recoverable hardware failure (e.g., controller failure, device failure, parity error) while performing the command or during a self test.

5

Table 5-4. MVME167/MVME187 SCSI Sense Keys (cont'd)

Sense Key (Hex)	Error Description
$05	**ILLEGAL REQUEST.** Indicates that there was an illegal parameter in the command descriptor block or in the additional parameters supplied as data for some commands. This may be indicative of memory failure in the CPU or on the device controller board.
$06	**UNIT ATTENTION.** Indicates that the removable medium may have been changed or the target has been reset.
$07	**DATA PROTECT.** Indicates that a command that reads or writes the medium was attempted on a block that is protected from this operation. The read or write operation is not performed.
$08	**BLANK CHECK.** Indicates that a sequential-access or write-once device encountered blank medium or format-defined end-of-data indication while reading or a write-once device encountered a non-blank medium while writing.
$09	**VENDOR SPECIFIC.** This sense key is available for reporting vendor specific conditions. Refer the device manual for cause of the condition.
$0A	**COPY ABORTED.** Indicates that a COPY, COMPARE, or COPY AND VERIFY command was aborted due to an error condition on the source device, the destination device, or both.
$0B	**ABORTED COMMAND.** Indicates that the target aborted the command.
$0C	**EQUAL.** Indicates that a SEARCH DATA command has satisfied an equal comparison.
$0D	**VOLUME OVERFLOW.** Indicates that a buffered peripheral device has reached the end-of-partition and data may remain in the buffer that has not been written to the medium.
$0E	**MISCOMPARE.** Indicates that the source data did not match the data read from the medium.
$0F	**RESERVED.**

Some MVME167/MVME187 SCSI errors are not caused by SCSI failures, but rather by the firmware that interfaces to the NCR chip, bus errors, communication failures between the NCR chip and local memory, etc. The following tables give further explanation of these errors.

Table 5-5. MVME167/MVME187 SCSI Firmware Error Codes

Error Code (Hex)	Error Description
$00	**GOOD.** The NCR firmware has successfully completed the command and no corrective action is necessary.
$01	**NOP.** The NCR firmware was sent a command with no command mode bit set. The user may have intentionally issued this command, in which case the returned status indicates that the desired no operation action has occurred. Otherwise the user has issued an command in error and must set one of the mode request bits.
$02	**SCSI BUS RESET.** The NCR firmware has either been issued or has detected a SCSI bus reset. All commands currently outstanding to the NCR firmware at the time of a SCSI bus reset will be terminated with this status, leaving the NCR firmware device queues empty. All commands which are terminated by this reset maybe re-issued by the SSID driver at its discreation.
$03	**DEVICE RESET.** This status is returned only for target mode commands outstanding to the NCR firmware. It indicates that a SCSI device reset message has been received by the NCR firmware in the target mode. The target application should reset itself in according to its reset mode and ready itself to receive more target commands. *NOTE: SSID does not support the target mode of operation.*
$04	**ABORT.** This status is returned only for target mode commands outstanding to the NCR firmware. It indicates that a SCSI abort command message has been received by the NCR firmware in the target mode for the specified logical unit. The attached target application should abort the current I/O and all queued I/O and instruct the NCR firmware to release the SCSI bus by going to the bus free state. *NOTE: SSID does not support the target mode of operation.*
$05	**ABORT TAG.** This status is returned only for target mode commands outstanding to the NCR firmware. It indicates that a SCSI abort command message has been received by the NCR firmware in the target mode for the specified logical unit. If the target application supports tagged queuing, it should abort only the current I/O and instruct the NCR firmware to release the SCSI bus by going to the bus free state. If the target application does not supports tagged queuing, it should respond with a reject message to the initiator of the abort message. *NOTE: SSID does not support the target mode of operation.*

5

Table 5-5. MVME167/MVME187 SCSI Firmware Error Codes (cont'd)

Error Code (Hex)	Error Description
$06	**CLEAR QUEUE.** This status is returned only for target mode commands outstanding to the NCR firmware. It indicates that a SCSI abort command message has been received by the NCR firmware in the target mode for the specified logical unit. If the target application supports tagged queuing, it should abort the current I/O and all queued I/O and instruct the NCR firmware to release the SCSI bus by going to the bus free state. If the target application does not supports tagged queuing, it should terminate the current I/O and instruct the NCR firmware to release the SCSI bus by going to the bus free state. *NOTE: SSID does not support the target mode of operation.*
$07	**DATA OVERFLOW.** The command was aborted due to a data overflow. The overflow may have occurred on a DATA-OUT phase or on a DATA-IN phase for non-scatter gather type commands. A data overflow is defined to occur if the supplied byte count is exhausted yet the SCSI bus remains in the DATA-IN or DATA-OUT phase. For the DATA-IN phase, the NCR firmware will read the remaining data from the target device into a bit bucket before terminating the command. For the DATA-OUT phase, the NCR firmware will write the data byte 0x0E for all requested bytes until the the DATA-IN phase is terminated.
$08	**DATA UNDERRUN.** The command was aborted due to a data underrun. The underrun may have occurred on a DATA-OUT phase or on a DATA-IN phase for non-scatter gather type commands. A data underrun is defined as a condition wherein the current data transfer completes yet the supplied byte count has not been exhausted.
$0B	**BAD QUEUE DEPTH.** This status is returned on a configuration mode command when the command is attempting to set the queue deth greater than the 255 item value allowed by the SCSI-2 spec.
$0C	**SELECTION TIMEOUT.** The command was aborted because the NCR firmware could not select a SCSI device within 250 milliseconds.
$0D	**RESELECTION TIMEOUT.** The command was aborted because the NCR firmware could not reselect a SCSI device within 250 milliseconds.
$0E	**BUS ERROR.** The command was aborted due to a BUS ERROR during a DATA phase to/from the local memory bus of the MVME167/MVME187. The NCR firmware will terminate a SCSI bus DATA-IN phase by dumping the remaining data or on a DATA-OUT phase by writing an error pattern.
$0F	**BUS ERROR.** The command was aborted due to a BUS ERROR during a NON-DATA phase to/from the local memory bus of the MVME167/MVME187.

Table 5-5. MVME167/MVME187 SCSI Firmware Error Codes (cont'd)

Error Code (Hex)	Error Description
$10	**ILLEGAL INSTR.** The command was aborted due to the NCR chip attempting to execute an illegal NCR chip script instruction or a non-longword aligned NCR chip script instruction.
$11	**UNEXPECTED DISC.** The command was aborted due to an unexpected disconnect.
$12	**UNEXPECTED PHASE CHANGE.** The command was aborted due to an unexpected phase change.
$13	**SCSI BUS HUNG.** The command was aborted because the SCSI bus hung during the command. This status is caused by any SCSI bus phase other than a selection phase where a device did not respond within 250 milliseconds.

5

The following class of errors are initialization errors.

Table 5-5. MVME167/MVME187 SCSI Firmware Error Codes (cont'd)

Error Code (Hex)	Error Description
$09	**CLOCK FAST.** The initialize command was aborted due to a clock speed greater than 75MHz. The clock speed must be less than or equal to the ASCII equivalent of 75 X 100 MHz (**0x37353030**). The clock speed must be within the range of 25 X 100Mhz to 75 X 100Mhz (**0x32353030 to 0x37353030**).
$0A	**BAD CLOCK PARAMETER.** The initialize command was aborted due to an invalid clock speed parameter. The clock speed must contain valid ASCII characters of the decimal values between 0 - 9 (**0x30 - 0x39**).
$1E	**SCRIPT PATCH.** The initialization was aborted due to the NCR chip failing to patch the NCR scripts. This failure maybe caused by a Bus Error, an Abort, or an Illegal NCR script Instruction.
$1F	**NO SCSI BUS.** The initialization was aborted because the SCSI bus was monitored and found to be in an illegal state. This illegal SCSI bus state could be because the NCR chip is not physically attached to a SCSI bus. This would result in the SCSI bus signals at the NCR 53C710 hardware interface to float in an undefined state. Another potential source for this error is that the SCSI bus is not terminated properly. A third possibility is that a device connected to the SCSI bus is driving the bus to an invalid state.

5

The following class of errors are termed protocol violation errors because each indicates a violation of the SCSI bus protocol. Generally, these violations are illegal phase transitions from a given phase. The NCR firmware will attempt to get the device off the bus in each instance of a protocol violation before returning the error status.

Table 5-5. MVME167/MVME187 SCSI Firmware Error Codes (cont'd)

Error Code (Hex)	Protocol Violation Error Description
$14	**PROTOCOL VIOLATION #1.** The command was aborted because the target device transition to some other SCSI bus phase other than the expected data phase (either DATA-IN or DATA-OUT).
$15	**PROTOCOL VIOLATION #2.** The command was aborted because the expected data phase was in the wrong direction. For example: If a DATA-IN phase was expected, then a DATA-OUT phase occurred.
$16	**PROTOCOL VIOLATION #3.** The command was aborted due to an incorrect phase following a selection.
$17	**PROTOCOL VIOLATION #4.** The command was aborted due to an incorrect phase following a message out phase.
$18	**PROTOCOL VIOLATION #5.** The command was aborted due to an incorrect phase following a data phase (either DATA-IN or DATA-OUT).
$19	**PROTOCOL VIOLATION #6.** The command was aborted due to an incorrect phase following a command phase.
$1A	**PROTOCOL VIOLATION #7.** The command was aborted due to an incorrect phase following a status phase.
$1B	**PROTOCOL VIOLATION #8.** The command was aborted due to an incorrect phase following a restore pointer message.
$1C	**PROTOCOL VIOLATION #9.** The command was aborted due to an incorrect phase following a save data pointers message.
$1D	**PROTOCOL VIOLATION #A.** The command was aborted because the expected identify message phase after reselection did not occur.

MVME188 Microcomputer Multiprocessor Tests

On any MVME188, a suite of tests is available to ensure proper operation of multiple 88100 CPU chips. If only one processor is present, all these tests automatically pass.

To access the MVME188 multiprocessor tests, type:

SA: 188.0/188mpu RETURN

This displays a menu of multiprocessor tests:


```
Selections for Test "188 Multiple MPUs"

   0) Multi-MPU Awake tests
   1) Multi-MPU Memory tests
   2) Multi-MPU SWI tests
   a) All Multi-MPU tests

SA:
```

`0) Multi-MPU Awake tests`
This test ensures that all available processors are at least minimally functional and are able to access shared memory.

`1) Multi-MPU Memory tests`
This test causes all slave processors to run a brief write-read test on a section of memory. By default, the size of the section is 32,768 (32K) bytes. This size may be modified by altering the memory_size parameter using `cf 188mpu`; the new size may be from 128 bytes to 65,536 (64K) bytes.

`2) Multi-MPU SWI tests`
During this test, the ability of the slave processors to cause and respond to software interrupts (SWIs) is tested. Each slave issues a SWI to the master processor which, in turn, issues a SWI to the slave. No inter-slave SWIs are tested. This capability can be tested by using the 188Bug to run SSID with another processor as master. Refer to the *MVME188BUG 188Bug Debugging Package User's Manual* for more details.

a) `All Multi-MPU tests`
Executing this menu command causes the MPU Awake, MPU Memory, and MPU SWI tests to be executed repeatedly. The default number of iterations is 200; this can be modified by altering the iterations parameter using `cf 188mpu` in the **cmdline** mode; allowable iterations are from 10 to 10000. Two hundred iterations of the test takes approximately 4.5 minutes.

MVME18X Error Statistics

On any M88100-based system, a test is available to display and clear a log of board errors. These errors are logged against the processor that fielded them. In general, these failures are not directly attributable to a specific board but rather indicate a malfunction of the processor board(s) or chassis itself.

NOTE

The logged errors can not be cleared by the system command **clear** or by the command line command **ce**.

To access the MVME18x Errors statistics, type:

`SA:` **log18x.0** (RETURN)

This displays a menu of MVME18x Error Statistics functions:

```
Selections for Test "18x Error Statistics"

   D) Display 18x errors
   C) Clear 18x errors

SA:
```

D) Display 18x errors

This function displays all processor board errors:

```
              88k Board Errors by Processor

      PROCESSOR                       0    1    2    3
Stray interrupts                      0    0    0    0
Uninitialized VME interrupts          0    0    0    0
Unset VME interrupts                  0    0    0    0
VME Sysfail interrupts                0    0    0    0
Parity errors                         0    0    0    0
Software interrupt errors             0    0    0    0
AC Fail errors                        0    0    0    0
Write Posting interrupts              0    0    0    0
```

- Stray interrupts are those whose source is unidentifiable. In other words, they are not VME interrupts, timer interrupts, or DUART interrupts. In general, it is *not* possible to successfully recover from this error, but SSID does attempt to continue.
- Uninitialized VME interrupts are those that are never assigned to a test and should never occur. In conformance with the VMEbus Specification, the 0x0F vector is never assigned in SSID. Reception of this vector is logged as an uninitialized VME interrupt. Recovery is usually successful.
- Unset VME interrupts are VME interrupts which are legal but unclaimed by any test. Recovery from this error is usually successful.
- VME Sysfail interrupts occur whenever any board asserts SYSFAIL. This is not an actual error. When the first Sysfail interrupt occurs, SSID logs the error and disables further VME Sysfail interrupts. Note that the MVME181 does not actually issue a Sysfail interrupt but, instead, asserts a SYSFAIL status line; this is treated as a Sysfail error.
- Parity errors can occur when accessing local memory or the VMEbus. Recovery is usually successful.

- A software interrupt (SWI) error can occur only on the MVME188 board set. It is caused by the receipt of an unexpected SWI. Recovery is usually successful.
- AC Fail errors can occur only on the MVME187 board. It is caused by the receipt of an AC Fail interrupt. Recovery is usually successful.
- Write Posting interrupt errors can occur only on the MVME187 board. It is caused by the receipt of an Write Posting interrupt. Recovery is usually successful.

`C) Clear 18x errors`
This function resets all error logs for all processors to 0. This *does not* reenable any disabled interrupts.

Memory Board Control

The SSID Memory Board Control functions enable a user to turn memory error reporting **ON** or **OFF** for specific memory boards.

SSID Memory Board Control Menu

To access the SSID Memory Board Control menu, type:

SA: logmem (RETURN)

A display similar to the following will appear:

```
Selections for Test "Memory Error logger"
Available memory board CSRs:
         204.0        224.0        2?6.0        230.0
         205.0        205.1        187.0
Available memory board commands:
   a) - Change  Memory Board CSR Scanner
   b) - Configure Memory Board table
   d) - Disable Memory Board Error Reporting
   e) - Enable  Memory Board Error Reporting
   m) - Display all Supported Memory Board Drivers
   s) - Display Memory Board Status
   C) - Clear Memory Board error log
   D) - Display logged Memory Board errors
SA:
```

5

The `Available memory board CSRs` section defines the memory boards that were found during the startup of SSID. Also, some commands require a memory board name as an argument. A memory board name is specified by using one of the names listed in the `Available memory board CSRs` section of the menu. For example, to disable memory board error reporting on the MVME187, type:

SA: d 187.0 [RETURN]

More than one board can be specified. For example:

SA: d 187.0 205.0 236.0 [RETURN]

A board name can be used without the period and the number. It will select all the boards with the same board name. For example, the following command will select both of the MVME205 boards:

SA: d 205 [RETURN]

A brief explanation of each command follows:

`a) - Change  Memory Board CSR Scanner`
This command allows the user to change how the memory board CSRs are scanned. This command supports three arguments; if none are specified, a usage message is printed. The arguments are defined as follows:

on

This argument starts the memory board CSR scanner if the scanner is not already running. The command has an optional second argument for the scanner delay time. The scanner delay time is the time between memory board CSR scans. The delay time is taken as a number specified in milliseconds. For example, the following command will turn the scanner on with a *100 millisecond* delay between scans:

`SA:` **a on 100** (RETURN)

The smallest allowable number is 10, which specifies a *10 millisecond* delay. If the delay time is not specified with the **on** argument, the default scanner delay time is used. The default scanner delay time is *100* for a *100 millisecond* delay. If the scanner is already running, the scanner is restarted with the new selected scanner time.

off

This argument stops the memory board CSR scanner when the scanner is running.

st

This argument prints the scanner's current run status.

`b) - Configure Memory Board table`
This command allows the user to configure any memory board CSR listed in the `Available memory board CSRs` section. The command supports four arguments; if none are specified, a usage message is printed. The arguments are defined as follows:

all

This argument instructs the configure command to edit all `Available memory board CSRs`.

add

This argument instructs the configure command to add a memory board CSR entry to the `Available memory board CSRs` section.

del

This command requires an additional argument. The configure function will delete the remaining memory board name arguments from the `Available memory board CSRs` section. The user will be asked to confirm memory board CSR deletion.

memory board name

This argument instructs the configure command to edit the named memory board CSR in the `Available memory board CSRs` section.

5

`d) - Disable Memory Board Error Reporting`

This command allows the user to disable error reporting on any memory board listed in the `Available memory board CSRs` section. This command supports several arguments. The **all** argument means to disable error reporting on all known memory board CSRs. The only exception to this is when memory boards do not have valid starting and ending addresses. The rest of the arguments to this command are taken as board name arguments. If no arguments are specified, a usage message will be printed.

`e) - Enable Memory Board Error Reporting`

This command allows the user to enable error reporting on any memory board listed in the `Available memory board CSRs` section. The command supports several options. One option is **BECC**, which enables bus error reporting on ECC correctable memory errors (single-bit errors) for the remaining memory board name arguments. This option works only for memory boards that are of the error correction type; it must be the first argument passed to this command. The next option can be **all**. This option enables error reporting for all of the known memory board CSRs; the only exception is when memory boards do not have valid starting and ending addresses. The rest of the arguments to this command are taken as board name arguments. If no arguments are specified, a usage message is printed.

`m) - Display all Supported Memory Board Drivers`

This command allows the user to show a list of the currently supported memory board drivers.

`s) - Display Memory Board Status`
This command allows the user to show information about the memory boards listed in the `Available memory board CSRs` section. The command supports two arguments. The first argument is **all**. When the **all** argument is used, the status information for all of the memory boards listed in the `Available memory board CSRs` section is printed. The second argument is the board name; it displays the status for only the named board. If no arguments are specified, a usage message is printed.

`C) - Clear Memory Board error log`
This command will clear all of the logged memory board errors. This command does **NOT** support any arguments.

`D) - Display logged Memory Board errors`
This command will display all of the logged memory board errors. This command does **NOT** support any arguments.

Interactive Memory Tests

The SSID interactive memory tests are designed to run interactively with other board tests. The memory diagnostics perform extensive tests of the memory board logic, RAM Chips, VSBbus interface, and VMEbus interface.

System memory boards must be strapped correctly for the interactive memory tests to work properly. During the memory probe at SSID boot time, a list is created of the physical memory found, mapping around any holes or non-exsistant memory so that all memory installed in the system will be tested. Any overlap in memory addresses is assumed to be bad RAM. For information on proper strapping of memory boards, refer to the System Manual for your particular system.

In M68000-based systems, two limitations are imposed on the total amount of memory in a given system:

1. In systems configured with either an MVME330A (OfficeLAN controller) or an MVME330B (RFS controller), total memory is limited to 12Mb.
2. In systems configured with an MVME320 (Winchester/Floppy Disk controller) but no MVME330A or MVME330B, total memory is limited to 16Mb.

Neither of these limitations apply to M88000-based systems.

The interactive memory tests work in the following manner:

The tests examine the diagnostic kernel's free list to determine which block of memory to test. Starting with the first block in the free list, the tests determine whether or not a memory block has already been tested. If it has not been tested, the memory tests access a predetermined portion of the memory block. The amount of memory accessed depends on the size of the contiguous block. The memory tests continue in this way until all available free memory has been tested.

At this point the memory tests check the process table to determine if other processes (using memory not yet tested) are running. The memory tests cannot access this untested memory until the processes quit. While waiting to access the memory, the memory tests take a free block and repeatedly test it. When all other processes quit, the memory tests access the remaining untested memory, test it, and then stop.

Memory tests check all RAM except the block of memory where SSID is loaded. SSID does not check parity because SST does this at system startup time.

If a memory board is failing, the interactive memory tests display a message identifying the bad address on the memory board. In some cases, the tests also display the expected and actual data. The interactive memory tests do not size memory but rather use the free memory found by the diagnostics kernel at startup time.

To access the test menus for the memory tests, type:

```
SA: mem.0
```

5

A display similar to the following appears:

```
Selections for Test "Memory Test"

   2) Interactive AA/55 Pattern
   3) Interactive 00/FF Pattern
   4) Interactive C3/3C Pattern
   5) Interactive Cross Address Test
   6) Interactive Walking Ones
   7) Interactive Walking Zeros
   8) Interactive X Move
   9) Burn-in Memory Test
```

The following briefly explains the tests:

`2)` through `8)`

By default, these tests perform the indicated pattern on every longword (4 bytes; 32 bits). All RAM is written to and a verification read is performed.

It is possible to change the memory configuration to test address increments other than every longword, thereby changing the execution time of the memory tests. Refer to the **cf** command (Example 3) in Appendix A for details.

`9) Burn-in Memory Test`

This test performs tests 2 through 8 on each block of memory, using the increment value defined in the memory configuration to determine which longwords are tested. Since interactive memory tests run until all other board tests have finished, only one memory test can be run in any given set of board tests. By combining all the memory tests into this one option, all memory tests can run concurrently with other board tests.

Interactive Memory Test Configuration

The interactive memory test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

> **cf mem.0**

```
/bdtest/mem.0 configuration
   walk_step = 1 ?                          # address increment value
```

MVME393 and MVME395 Graphics Display Controllers

NOTE

The **UART External Loopback** test for the MVME393 requires an MVME792-1 or MVME792-2 transition board and external serial port loopback cables. The **UART External Loopback** test for the MVME395 requires an MVME795 transition board and external serial port loopback cables. The **UART External Loopback** test for the MVME395 requires an MVME796 transition board and external serial port loopback cable. Refer to the next section of this chapter for the cable pin connections.

5

The MVME393 and MVME395 Graphics Controller tests check the functionality of the MVME393 and the MVME792 transition board; and of the MVME395 and the MVME795 or MVME796 transition board. To access the MVME393 or MVME395 tests, type:

SA: **393.0/gb** (RETURN)

The following display appears for the MVME393:

```
Selection for Test "VME393/5"
(This menu is the same for all MVME393 boards)
   1) MVME393 Download Test
   2) MVME393/792 Video Test Pattern
   3) MVME792 UART Internal Loopback
   e) MVME792 UART External Loopback
   r) MVME393 Board Reset
SA:
```

The following display appears for the MVME395:

```
Selection for Test "VME393/5"
(This menu is the same for all MVME395/795 boards)

    1) MVME395 Download Test
    2) MVME395/795 Video Test Pattern
    3) MVME795 UART Internal Loopback
    e) MVME795 UART External Loopback
    r) MVME395/795 Board Reset

SA:
```

The following display appears for the MVME395 and MVME796:

```
Selection for Test "VME393/5"
(This menu is the same for all MVME395/796 boards)

    1) MVME395 Download Test
    2) MVME395/796 Video Test Pattern
    3) MVME796 UART Internal Loop
    4) MVME796 Keyboard Port Internal Loop
    5) MVME796 Beeper Test
    e) MVME796 UART External Loop
    h) MVME796 UART Handshake Test
    x) MVME796 Keyboard Port External Loop
    r) MVME395/796 Board Reset

SA:
```

The first time the test menu is selected, the following steps are taken:

The program checks to see if the board has passed its on board self test before it tries to run the board. Next, the program determines if the board is in console mode. If the board is in console mode, the test reports that it is in console mode and does not test the board any farther. Otherwise, the program determines

5

what type of board it is (an MVME393 or an MVME395/795 or an MVME395/796) and sets up the correct testing modes and menus.

NOTE

> The RGB connections may be connected to a monitor during any of the tests. They are not required for a test to pass. They are only required if the user wants to view the video test patterns. Viewing the video test patterns is the only way to know if the analog portion of the transition card or video display unit is working.

1) `MVME393 Download Test` or `MVME395 Download Test`
   A mini program is moved into the Graphics Service Processor (GSP) memory and executed. Then the GSP is verified to have responded.

2) `MVME393/792 Video Test Pattern` or `MVME395/795 Video Test Pattern`
   This test moves a display program into the GSP memory and executes it.

   The Video Test Pattern program does the following:

   The test runs the sequence described below first in gray scale and then in color.

   The gray scale palette order is:

   black, 14 shades of gray, white.

NOTE

> On an MVME395, there are 254 shades of gray.

The color palette order is:

black, white, green, yellow, blue, magenta, cyan, red, light steel blue, coral, forest green, khaki, sky blue, plum, sienna, light gray.

NOTE

> On an MVME395, the palette is repeated for 15 more sets.

5

The following video test patterns display:

1. The display area is cleared and a large cross hair and display number are drawn. The cross hair is at the center of the display and is 100 pixels in width. The display number is in the upper left hand corner of the display and it is drawn in block numbers. (When in color mode, the cross hair is magenta and the number is red on a light gray background.)

   On the MVME393, the second display is the same as the first but the number on the display is the digit two.

   On the MVME395, a cross hair hardware cursor is displayed in coral along with a small arrow pointing at the center of the display. It is drawn full screen 1 pixel wide 1 pixel high through the center of the screen in both the vertical and horizontal directions. The cursor is also blinking. Drawn full screen means: something drawn from one edge of the screen to the other edge of the screen.

2. A series of squares are drawn:

   Four squares of 100 by 100 pixels at positions;

x = 0, y = 0	(When in color mode, this is red)
x = 0, y = 200	(When in color mode, this is coral)
x = 200, y = 0	(When in color mode, this is blue)
x = 200, y = 200	(When in color mode, this is forest green)

   Sixteen squares of 50 by 50 pixels at positions y = 400 and;
   x = 50, x = 102, x = 154, x = 206, x = 258, x = 310, x = 362, x = 414,
   x = 466, x = 518, x = 570, x = 622, x = 674, x = 726, x = 778, x = 830

   These sixteen squares are in palette order.

3. Fuli screen 16 bars. For the MVME393, the bar width is ((1024/16) - 2) and the bar height is (768). For the MVME395, the bar width is ((1280/256) - 2) and the bar height is (1024). These sixteen bars are in palette order.

4. Sixteen squares across and sixteen squares down with a two pixel gap between squares.
   The first row is in palette order starting at black.
   The second row starts at black plus one and ends with black.
   The pattern of shift continues through the sixteenth row.

5. Sixteen rectangles across and sixteen rectangles down.
   The first row is in palette order starting at black.
   The second row starts at black plus one and ends with black.
   The pattern of shift continues through the sixteenth row.

6. Sixteen rectangles across and sixteen rectangles down.
   The first row is in palette order starting at black.
   The second row starts at black plus one and ends with black.
   The pattern of shift continues through the sixteenth row.
   The pattern will move from right to left for one hundred rotations.

7. Sixteen rectangles across and sixteen rectangles down with a two pixel gap between rectangles.
   The first row is in palette order starting at black.
   The second row starts at black plus one and ends with black.
   The pattern of shift continues through the sixteenth row.

8. Sixteen rectangles across and sixteen rectangles down with a two pixel gap between rectangles.
   This will be a checkerboard pattern using palette positions 10 and 11.
   For the color part of the test, it is coral and forest green.

9. Sixteen rectangles across and sixteen rectangles down with a two pixel gap between rectangles.
   The first row is in palette order starting at black.
   The second row starts at black plus one and ends with black.
   The pattern of shift continues through the sixteenth row.
   The pattern will move from right to left for one hundred rotations.
   Then the pattern will move from left to right for one hundred rotations.

10. Quarter-screenful palette count of bars.

    The first row of bars are drawn left to right, starting with the lowest palette value.
    The second row starts with the highest palette value.
    The third row starts with the lowest palette value.
    The fourth row starts with the highest palette value.

    For the MVME393, the bar width is (1024/16) and the bar height is (768/4); the palette count is 16.

    For the MVME395, the bar width is (1280/256) and the bar height is (1024/4); the palette count is 256.

5

2) `MVME395/796 Video Test Pattern`
This test moves a display program into the GSP memory and executes it.

The Video Test Pattern program does the following:

The test runs the sequence described below.

The background is filled with a gray background.

Eight horizontal color bars are painted starting at the top of the screen. Each color bar is comprised of 64 20 pixel wide by 80 pixel high smaller bars. Each of the small bars increases in intensity from black to full intensity from left to right. Starting from the top of the display the first set of four color bars is blue, green, red, and white. The second set of four color bars is the same as the first except that this set decreases in intensity from left to right so maximum intensity is at the left and black is at the right.

NOTE

A white grid on a black background is drawn in the bottom left of the screen. A black grid on a white background is drawn in the bottom middle of the screen. A green cursor is drawn on the bottom right of the screen. A green on pixel border is drawn around the entire image.

3) `MVME792 UART Internal Loopback` or `MVME795 UART Internal Loopback`
This test can only be done on an MVME795/6 or MVME792-1 or MVME792-2 transition board; it does not use external loopbacks. This test downloads a UART process and executes it. Once the process has run, the results are verified.

4) `MVME796 Keyboard Port Internal Loop`
The Keyboard Internal Loopback test can only be done on an MVME796 transition board; it does not use external loopbacks. This test loops data within the keyboard interface chip and verifies it.

5) `MVME796 Beeper Test`
This test can only be done on an MVME796 transition board; it requires that a speaker be attached to pins 3 and 4 of jumper J2 of the MVME796. Either polarity is fine. This test produces a loud beep.

e) `MVME792 UART External Loopback` or `MVME795/6 UART External Loopback`
This test can only be done on an MVME795/6 or MVME792-1 or MVME792-2 transition board; it requires the use of external serial port loopback cables. This test downloads a UART process and executes it. Once the process has run, the results are verified.

h) `MVME796 UART Handshake`
The Handshake test can only be done on an MVME796 transition board; it requires the use of an external serial port loopback cable. This test verfies the handshake lines on the mouse port.

x) `MVME796 Keyboard Port External Loop`
This test can only be done on an MVME796 transition board; it requires the use of an AT compatible keyboard. The recommended Motorola part number is 01-W0346B02A. This test blinks a random pattern of lights on the keyboard LEDS and verifies the acknowledgment of the LED command by the keyboard.

r) `MVME393 Board Reset` or `MVME395 Board Reset`
This option resets the MVME393/5 via the reset bit in the Command Status Register (CSR), then verifies that some addresses in the dual-port static RAM contain zeroes.

MVME792-1/-2 Loopback Cabling and Pin Connections

Connect pin 1 to 4, pin 2 to 3, and pin 7 to 8 on the mouse port(J3). Connect the loopback cables and the loopback cable pin connectors as shown.

MVME792-1 `P1A <------> P1B`
(using DB-9 pin connectors) `P2A <------> P2B`

```
Male DB-9                                       Male DB-9
connector                                       connector
  pin #    (cable length 5 inches or more)        pin #
----------                                      ----------
   1       <-----|                  |----->         1
   4       <-----|                  |----->         4
   2       <------------------------------->        3
   3       <------------------------------->        2
   7       <-----|                  |----->         7
   8       <-----|                  |----->         8
```

MVME792-2	`P1` *(wires are internal to the connector)*
(using 17W5 pin connectors)	`P2` *(wires are internal to the connector)*
MVME795	`only has P1`

```
 17W5                                              17W5
connector                                        connector
 pin #        (wire length as needed)              pin #
----------                                      ----------

   1      <------------------------->              4
   2      <------------------------->              9
   5      <------------------------->              6
   7      <------------------------->             10
   8      <------------------------->              3
```

5

MVME393 and MVME395 Board Test Configuration

The MVME393 and MVME395 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

> **cf 393.0** *(This is for either the MVME393 or MVME395)*

```
/bdtest/393.0
   base[%0x10] = 0xfffa800 ?                       # base address
   am[0x0d,0x3d] = 0x0d ?                          # address modifier
   dsz[1-2] = 2 ?                                  # data bus size
   level[0-7] = 3 ?                                # default interrupt level
   vec[0x40-0x100] = 0x98 ?                        # default interrupt vector
   shared memory[%0x00080000] = 0xfd800000 # shared memory address
```

> **cf 393.0/gb** *(This is for either the MVME393 or MVME395)*

```
/bdtest/393.0/gb
   mode 0-use_default 1-use_cf[0-1] = 0  # xtal freq default
   393 control = 0x00cc                   # gsp control register
   393 dpyctl = 0xf010                    # see TI34010 GSP manual
   393 dpystrt = 0xfffc                   # see TI34010 GSP manual
   393 heblnk = 0x000e                    # see TI34010 GSP manual
   393 hesync = 0x0004                    # see TI34010 GSP manual
   393 hsblnk = 0x004e                    # see TI34010 GSP manual
   393 htotal = 0x0051                    # see TI34010 GSP manual
   393 veblnk = 0x0027                    # see TI34010 GSP manual
   393 vesync = 0x0009                    # see TI34010 GSP manual
   393 vsblnk = 0x0327                    # see TI34010 GSP manual
   393 vtotal = 0x032c                    # see TI34010 GSP manual
   395 control = 0x00cc                   # gsp control register
   395 dpyctl = 0xf040                    # see TI34010 GSP manual
```

```
395 dpystrt = 0xfffc                    # see TI34010 GSP manual
395 heblnk = 0x000b                     # see TI34010 GSP manual
395 hesync = 0x0005                     # see TI34010 GSP manual
395 hsblnk = 0x0033                     # see TI34010 GSP manual
395 htotal = 0x0035                     # see TI34010 GSP manual
395 veblnk = 0x0021                     # see TI34010 GSP manual
395 vesync = 0x0003                     # see TI34010 GSP manual
395 vsblnk = 0x0421                     # see TI34010 GSP manual
395 vtotal = 0x0424                     # see TI34010 GSP manual
```

The GSP video timing parameters can be adjusted by the **cf** command.

NOTE

Changing the timing parameters should only be done with the help of Hardware Engineering or Field Service. Changing these values requires knowledge of the TI34010 processor, the layout of the video RAM, the frequency of the crystal installed in the transition card, and the timing requirements for the monitor being used.

Refer to the manuals for the MVME393 or MVME395 for more information and references to other related manuals. The manual(s) containing the timing information for the monitor will also be needed.

The meaning of the labels, such as `dpyctl` in the **cf** configuration, is described in the GSP programming guide for the TI34010 processor.

MVME393 or MVME395 Error Messages

The following are error message descriptions for the MVME393 and MVME395 boards:

`The 393 or 395 board is in Console mode, no tests can be run`

This message is not an error; it is an indicator to let the operator know the board is not being tested.

NOTE

It is possible for the board to be erroneously in console mode. The two causes that have been observed are: 1) the board has the firmware incorrectly installed, and 2) the transition board is bad.

```
Board_fail flag in the CSR is on
The 393 or 395 board firmware has reported an error
no other tests can be run.
```

The board has detected some type of onboard hardware error. If these are the only two error messages, then the board could not report any more information.

```
Board_fail flag in the CSR is on
393/395 error data from VME dual port ram:
Composite status word (csw) = 6201, magic = 0,
loop count (lcnt) = 0, fatal = 1, error = 1,
fault address (faddr)= 400400,
expected data (expdata) = 123, read data (readata) = fffe
The 393 or 395 board firmware has reported an error
no other tests can be run.
```

The board has detected some type of onboard hardware error. The error message is decoded as follows:

The VME Dual Port RAM "Byte, Word, Long Word" test failed at address 400400: expected data 123, read data fffe.

The following is a breakdown of the Composite status word:

csw -

Bits D15 through D12 indicate the main test the board was running.
Bits D11 through D8 indicate the sub-test the board was running.
Bits D7 through D0 are ignored. An X means it may or may not be printed and could be any value. To read this message, assume the first digit from the left is bits D15 through D12.

CPU1 - the MC68010 on the MVME393 or MC68020 on the MVME395

11XX = CPU1 - SIMPLEI	Simple instruction test.
12XX = CPU1 - COMPLEXI	Complex instruction test.

CPU2 - the TI34010

21XX = CPU2 - CACHE	
22XX = CPU2 - COMPLEXA	Complex address test.
23XX = CPU2 - EXCEPT	Exception processing test.
24XX = CPU2 - TRP316	Trap test.
25XX = CPU2 - TRP1731	Trap test.
31XX = ROM - CRC	The ROM (firmware) check sum failed to compare.

LRAM - local CPU1 RAM

41XX = LRAM - WALK	Walking one bit test.
42XX = LRAM - BWL	Byte, Word, Long word test.
43XX = LRAM - MISA	
44XX = LRAM - MARCH	Marching address test.
45XX = LRAM - NONDES	Non-destructive test.

SDP - Graphics Dual Port RAM

51XX = SDP - WALK
52XX = SDP - BWL
53XX = SDP - MISA
54XX = SDP - MARCH
55XX = SDP - NONDES

DDP1 - VME Dual Port RAM

61XX = DDP1 - WALK
62XX = DDP1 - BWL
63XX = DDP1 - MISA
64XX = DDP1 - MARCH
65XX = DDP1 - NONDES

DDP2 - Graphics Off-Screen Display RAM

71XX = DDP2 - WALK
72XX = DDP2 - BWL
73XX = DDP2 - MISA
74XX = DDP2 - MARCH
75XX = DDP2 - NONDES

CSR - Command status register

81XX = CSR - CREGS	Command register test.
9XXX = HOSTDATA	Graphics host data port test.

5

Unknown board type. Not a 393 or 395 board.0x00000000...
The board type is unknown and the hex number is the value returned by the program.

Error board busy did not clear
The board busy bit in the CSR was not cleared as expected.

Error board busy did become set
The board busy bit in the CSR was not set as expected.

Error board bus available was not set
The board bus available bit in the CSR was not set as expected.

Could not halt Graphics Board
The protocol to take control of the board failed.

Memory not available for buffer
There was not enough memory in the memory pool to meet the request.

unable to create semaphore
There are no more system semaphores available.

MVME393/5 Create channel error
The program was unable to create a Buffered Pipe Protocol (BPP) channel to talk with the onboard firmware.

MVME393/5: Corrupted env on ctl 0
A BPP envelope pointer is bad on the board.

x393send: stale packet()
A BPP work packet was stale (bad) on the work queue.

envelope %x valid flag set
This BPP envelope has not been serviced on the board.

MVME393/5 error sending read descriptor command
The program was unable to send a read descriptor command to the onboard firmware.

MVME393/5 Warning no transition card detected
The onboard firmware reported that there was no transition card connected to the board.

Test is not applicable. Wrong transition board.
The attached transition board is not what the test expects.

`Unable to create the process.`
or
`gb395xloop: unable to create process`
The program was unable to create a process to run on the board.

`Internal UART error -- downloaded test did not start`
or
`External UART error -- downloaded test did not start`
The cross loaded UART test program did not begin execution.

`Internal UART error -- downloaded test failed`
or
`External UART error -- downloaded test failed`
The cross loaded UART test program reported a failure.

`Internal UART error at address 28c0000, expected = 55, read = 54`
or
`External UART error at address 28c0000, expected = 55, read = 54`
The cross loaded MVME393 UART test reported a data miscompare during the UART test.

`Internal UART error at address 28c0000, port 0, expected 55, read 54`
`status 0, code 0`
or
`External UART error at address 28c0000, port 0, expected 55, read 54`
`status 0, code 0`
The cross loaded MVME395 UART test reported a data miscompare during the UART test.

`No response from Graphics Board`
The graphics processor did not mark the handshake flag after it ran the downloaded program.

`Graphics CPU failed to start`
The graphics processor did not mark the handshake flag when it started the downloaded program.

`Graphics CPU failed to handshake`
The graphics processor did not mark the handshake flag to acknowledge the last command.

5

5

Error: Channel 0: (Mouse) Handshake: RTS stuck low or CTS stuck high

Error: Channel 0: (Mouse) Handshake: RTS stuck high or CTS stuck low

Error: Channel 0: (Mouse) Handshake: DTR stuck low or DCD stuck high

Error: Channel 0: (Mouse) Handshake: DTR stuck high or DCD stuck low

Error: Channel 0/1: (Mouse/Console) Loopback: TXRDY timeout

Error: Channel 0/1: (Mouse/Console) Loopback: Timeout waiting for character

Error: Channel 0/1: (Mouse/Console) Loopback: Receive errors - SR xx

The data path to the MVME796 transition board is malfunctioning. Check that you have the correct transition cable.

Error: Channel 0/1: (Mouse/Console) Loopback: Pattern wrong: Sent xx, Received xx

The data path to the MVME796 transition board is malfunctioning. Check that you have the correct transition cable.

Error: Palette Address: Wrote xxxx, Read xxxx.

The data path to the MVME796 transition board is malfunctioning. Check that you have the correct transition cable.

Error: Palette RGB Register (red/green/blue)
Palette Register: Address xxxxxxxx, Wrote xxxx, Read xxxx.

The data path to the MVME796 transition board is malfunctioning. Check that you have the correct transition cable.

Error: Palette Register: Address xxxxxxxx, Wrote xxxx, Read xxxx.

The data path to the MVME796 transition board is malfunctioning. Check that you have the correct transition cable.

Error: SendKey: Timeout waiting for TDRE to set.

The transmit data register did not empty as expected. The keyboard control chip may not be clocking out data or the keyboard is not receiving it.

Error: SKI Loopback: Timeout waiting for TDRE to be set.

The transmit data register did not empty as expected. The keyboard control chip may not be clocking out data or not clocking it in.

Error: SKI Loopback: Timeout waiting for RDRF to set.

The transmit data register did not empty as expected. The keyboard control chip may not be clocking out data or not clocking it in.

Error: SKI Loopback: Data, Wrote xxxx, Read xxxx.

The keyboard did not get the expected data. The keyboard control chip may not be clocking out data or not clocking it in.

Error: SKI Loopback: Receiver, ParityErr = 1/0, FrameErr = 1/0.

5

MVME050 Utility Board

The three test groups for the MVME050 Utility Board are shown below followed by the menu selections for each test.

```
SA: 050.0 <CR>        (to select the VME050 Utility Board Tests)
Selection for Test "VME050 Utility Board Tests"
sp  - Serial Port Tests
tod - Real Time Clock
mi  - Miscellaneous Board Functions
SA: sp <CR>
   1) Internal-loop        (no external cable required)
   2) Baud Rate
   3) Parity
   4) External-loop        (requires an external loopback cable)
   5) Rx intr              (receiver interrupt)
   6) Tx intr              (transmitter interrupt)
   7) Interface intr       (RS-232 signal interrupt)
SA: tod <CR>
   0) Reg & Ram            (Register and RAM)
   1) Counters
   2) Periodic Intr
   3) Update Intr
   4) Alarm Intr
SA: mi <CR>
   0) LED Display          (LED = light-emitting diode)
   1) Read Switch
   2) Global BIM           (Global Bus Interface Module)
   3) Printer loop
   4) Printer Intr
SA:
```

MVME050 Board Test Configuration

The MVME050 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 050.0
/bdtest/050.0 configuration
   base[%0x100] = 0xffff1000 ?            # base address

> cf 050.0/mi
/bdtest/050.0/mi configuration            # miscellaneous

> cf 050.0/tod
/bdtest/050.0/tod configuration           # real time clock

> cf 050.0/sp
/bdtest/050.0/sp configuration            # serial port - sp050
   xfer size = 512 ?
```

MVMEenv Environmental Monitor Board

The purpose of the MVMEenv environmental monitor board is to allow customers to interface an "Uninterruptible Power Supply" (UPS) to a Delta chassis, monitor temperature and voltage levels in the Delta and external chassis, and notify the operating system when a power loss or over-temperature condition exists. The environmental monitor board tests provide verification of the board functionality in a Delta system chassis. A maximum of one environmental monitor board is supported in each Delta chassis.

All interrupts generated by the environmental monitor board are level 7 VMEbus interrupts. This requires that there be *only* one board in the system — the master CPU board — that acknowledges level 7 VMEbus interrupts. This also requires other boards that currently acknowledge level 7 VMEbus interrupts to be programmed to disregard the interrupt. If the board cannot be programmed to disregard the interrupt, the board's firmware must be modified to eliminate the acknowledgement of the level 7 VMEbus interrupt or the board cannot be used in a system containing an environmental monitor board. If an MVME337 board is installed, it should have revision 1.2 or higher firmware in the board so that level 7 VMEbus interrupts are not acknowledged by the MVME337.

After power-up, the environmental monitor board has interrupts disabled. Interrupts are enabled as part of the board initialization procedure. Pressing the ABORT button on the environmental monitor transition board has no effect before board initialization is complete. Board initialization is done just before display of the **env.0/envmon** menu or execution of the first environmental monitor board test.

5

After pressing the ABORT button on the environmental monitor transition board, enter **g** (RETURN) at the `1xx-Bug` prompt to return to SSID. The SSID menu is not redisplayed unless (RETURN) is pressed a second time. If memory locations containing code are modified, it may be necessary to reboot SSID.

Four registers are on the environmental monitor board. The contents of these registers at the time an error occurs are saved in the local error log. These saved register values, along with text describing the bits that are turned on, are displayed by the **Display Error Log** command. Definitions of these register bits can be found in the *MVMEenv Board Register Definitions* section.

5

The STATA register indicates the cause of an interrupt from the environmental monitor board. Once a condition occurs, the associated bit in STATA stays set to a 1 until the condition disappears. This can cause multiple messages to be displayed by the environmental monitor board interrupt handler on subsequent interrupts. For instance, if an interrupt occurs due to an over-temperature condition, a message indicating this is displayed. If the over-temperature condition still exists when a second interrupt for the environmental monitor board is received, messages indicating the over-temperature condition and the condition causing the second interrupt may be displayed.

The following tests are available for the MVMEenv environmental monitor board: verification of proper connection of the transition board, external configuration display, MVMEenv interrupter test, and a test to execute multiple iterations of the tests previously mentioned. Also available are commands to display/clear the local error log. Enabling of the `expert` level menu allows execution of the following tests/commands in addition to those listed above: display the interrupt log, display register contents, generate a system reset, power down all chassis, power down external chassis, and set test flags/options.

These tests/commands are further described in the following pages.

MVMEenv Board Tests

To access the board menu for the MVMEenv environmental monitor board, type:

`SA:` **env.0**

A display similar to the following appears:

```
Current Menu is /bdtest/env.0 -  "VMEenv Envir. Monitor Board
Tests"

envmon - VMEenv Envir. Monitor Board Tests

SA:
```

5

To access the test menu for the MVMEenv environmental monitor board, type:

SA: envmon

A display similar to the following appears:

```
Selections for Test "VMEenv Envir. Monitor Board Tests"

   1) Verify Conn. of Transition Board
   2) Display Ext. Device Configuration
   3) Interrupter Test
   a) Multiple Iterations of Tests 1,2,3
   D) Display Error Log
   C) Clear Error Log

SA:
```

Verify Connection of Transition Board

This test determines whether the MVMEenv transition board is properly connected (by reading the proper connection bit of the board's STATA register) and prints a message to indicate the result. If the transition board is not properly connected, a fatal error is logged.

Display External Device Configuration

This test displays the type of device (UPS, external chassis, or none) connected to each external device port. If an invalid device type (i.e., both UPS and external chassis) is indicated, an "invalid type" message is displayed and a fatal error is logged. If the currently indicated device type differs from the device type saved during board initialization, a "type changed" message is displayed and a non-fatal error is logged.

Interrupter Test

This test requests an interrupt from the environmental monitor board. If an interrupt is received from the board within 10 seconds and the interrupt was caused by the interrupt request, the test has passed. If an interrupt is received but was not caused by the interrupt request, the test waits for another interrupt until the 10-second timeout period expires. If an interrupt caused by the interrupt request is not received before the timeout period ends, a fatal error is logged. If interrupts from the environmental monitor board have been disabled, a message so indicating is displayed; the test is not executed and no errors are logged.

Multiple Iterations of Tests 1, 2, 3

This test is not implemented in DH02.00 or earlier releases of SSID. This menu selection executes tests 1, 2, and 3 multiple times during each SSID pass. A user can change the number of iterations to be executed per pass by entering **cmdline** mode, executing **cf env.0**, and changing the `iterations` value. The default is 200 iterations per pass of SSID. To avoid flooding the screen with messages during **confid/fault** testing, the normal output messages from tests 1 and 2 indicating connection and device status are not displayed during execution via this menu selection.

Display Error Log

This command displays the errors that were logged in the local error log during execution of tests. If no errors occurred on the environmental monitor board, the following displays:

```
SA: e
/bdtest/env.0/envmon  D)Display Error Log      ... starts

No error information in log for env.0/envmon
/bdtest/env.0/envmon  D)Display Error Log      ... passed
```

5

A display similar to the following appears if errors have been logged during test execution:

```
SA: D
/bdtest/env.0/envmon      D) Display Error Log       ...... starts

PASS   TIME-PASS  TIME-BOOT  DEVICE-ID      FAILING COMMAND
   0   00:00:01   00:02:38   env.0/envmon   Display Ext. Dev. Config.
     ERROR: 0xfffffffb, Device type shows both UPS and external
      chassis.
     STATA REGISTER BITS:  0x0000
     STATB REGISTER BITS:  0x0065
       UPS on port 2.
       UPS on port 3.
       External Chassis on port 3.
       External Chassis on port 1.
     COMMAND REGISTER BITS:  0x0048
       Interrupts enabled from Env. Monitor.
       Short overtemp timeout selected.
     Interrupt Vector Number: 0x0080

Depress RETURN to continue

 /bdtest/env.0/envmon  D)Display Error Log       ......passed
***SYSTEM TEST FAILED***
 Depress Return to Continue
```

The pass number and time at which the error occurred, along with the device identification, failing command description, error code and a text message explaining the error code, and the board register values are displayed. The **TIME-PASS** field displays the time at which the error occurred as counted from the beginning of the test pass. The **TIME-BOOT** field displays the time at which the error occurred as counted from the time SSID was booted.

The actual value shown for the registers varies from what is shown here, depending on the configuration of the system and the vector number that was assigned during execution of SSID. Refer to the *MVMEenv Board Register Definitions* section for an explanation of each register bit. Unused bits in the register values displayed are shown as zero even though the board sets most unused bits to a value of 1.

Clear Error Log

This command clears the errors saved in the local error log during execution of tests. This has no effect on the system error count summary that is accessed using the **disperr/clear** system level commands. If this error log is not cleared but the system error summary count is cleared, the error information can still be viewed using *Display Error Log* but the system error count summary will not reflect the number of errors saved in this log.

MVMEenv Expert Mode Tests

Expert mode tests (i.e., tests that can cause a system reset or remove power from the system, and commands used primarily for design verification) are available from the menu by going into the **cmdline** mode, executing **cf env.0**, and changing the `expert flag` to `1` to enable expert test mode. When **set menu** is executed to return to the menu system, the following tests, in addition to the tests described above, appear on the **envmon** menu:

```
i)  Display Interrupt Log
r)  Display Register Contents
s)  Reset System
p)  Powerdown All Chassis
c)  Powerdown External Chassis
z)  Set Test Flags/Options
```

Display Interrupt Log

This test is not implemented in DH02.00 or earlier releases of SSID. This command is used to display the interrupt log. During initialization, a buffer is allocated that is used to log the clock time and register contents when an environmental monitor board interrupt occurs. This information is primarily used during design verification but can be displayed by the SSID user.

Display Register Contents

This test is not implemented in DH02.00 or earlier releases of SSID. This command displays the current contents of the four registers on the environmental monitor board and decodes the bits for the user as shown in the *Display Error Log* description.

5

Reset System

This test is not implemented in DH02.00 or earlier releases of SSID. This test causes the environmental monitor board to generate a VME SYSRESET. User intervention is required to determine whether a reset of the Delta chassis was actually performed, and to restart the system unless the system is configured for autoboot after any reset. This test is only available in the `expert mode`. Before execution, a message is displayed indicating that no other tests should be running when this test is executed. The user will be queried to determine if execution should continue. If the user chooses to continue and SSID is still running at the end of a 10-second timeout period, a fatal error is logged.

Powerdown All Chassis

This test is not implemented in DH02.00 or earlier releases of SSID. This test causes the Delta (or main) chassis power to be latched, which also causes a request to power down to be sent to all external devices (UPS/external chassis). User intervention is necessary to determine whether a powerdown was actually performed and to restart the system. This test is only available in the `expert mode`. Before execution, a message is displayed indicating that no other tests should be running when this test is executed. The user is then queried as to whether execution should continue. If the user chooses to continue and SSID is still running at the end of a 10-second timeout period, a fatal error is logged.

NOTE

The *RESET* button on the environmental monitor transition board must be pressed after execution of the Powerdown All Chassis test. The front panel and CPU board *RESET* switches cannot be used.

5

Powerdown External Chassis

This test is not implemented in DH02.00 or earlier releases of SSID. This test causes the external chassis to be powered down. User intervention is necessary to determine whether a powerdown was actually performed and, possibly, to restart the system. If the external device is an external chassis, it should power down. If the external device is a UPS which has not lost AC power, it will continue to run. If the external device is a UPS which has lost AC power, power to the Delta chassis will be removed. The Delta chassis will reboot when AC power is restored to the UPS and the UPS restores power to the Delta chassis. This test is only available in the `expert mode`. Before execution, a message is displayed indicating that no other tests should be running when this test is executed. If the user chooses to continue and SSID is still running at the end of a 10-second timeout period, a fatal error is logged.

Set Test Flags/Options

This command allows the user to change options that affect the confidence and fault script testing. The options that can be changed for the environmental monitor board are:

— `DISABLE POWER REMOVAL by Env. Mon. board after an overtemperature condition`
This option removes the ability of the environmental monitor board to pull power from the Delta chassis if the system is still running after the overtemperature timeout period has expired.

— `ENABLE POWER REMOVAL by Env. Mon. board after an overtemperature condition`
This option restores the ability of the environmental monitor board to pull power from the Delta chassis if the system is still running after the overtemperature timeout period has expired.

— `DISABLE INTERRUPTS from Env. Mon. board`
This option removes the ability of the environmental monitor board to interrupt the main CPU for any reason. If interrupts are disabled, the ABORT button on the environmental monitor transition board has no effect.

— `ENABLE INTERRUPTS from Env. Mon. board`
This option restores the ability of the environmental monitor board to interrupt the main CPU.

— **SET SHORT TIMEOUT for power removal after an overtemperature condition has been detected by the Env. Mon. board**
This option allows the environmental monitor board to remove power from the Delta chassis approximately one minute after an overtemperature condition has been detected unless the **DISABLE POWER REMOVAL** option has been chosen.

— **SET LONG TIMEOUT for power removal after an overtemperature condition has been detected by the Env. Mon. board**
This option allows the environmental monitor board to remove power from the Delta chassis approximately 15 minutes after an overtemperature condition has been detected unless the **DISABLE POWER REMOVAL** option has been chosen.

5

MVMEenv Board Test Configuration

The MVMEenv board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

> **cf env.0**

```
/bdtest/env.0 configuration
    base[%0x100]=0xfffa900        # base adrs for ENV board
                                  #  should not be changed by user
    am[0x29]=0x29                 # A16/D16 non-priv adrs modifier
                                  #  should not be changed by user
    iterations[10-10000]=200      # num of iterations for test 'a'
    expert[0-1]=0                 # 1 = expert mode, 0 = novice
                                  #   can be changed to select which
                                  #   env test menu is displayed
```

MVMEenv Board Error Messages

The following list describes the error messages that may be displayed due to failures while running the MVMEenv tests. Most likely causes of these failures are indicated in each description. Where the monitor board assembly is indicated as the possible cause of a failure, the failure could be the monitor board, its transition board or the cabling between the two.

If multiple error messages are displayed, descriptions of each message should be read in order to determine the actual cause of the failure.

The error messages are listed in alphabetical order.

5

`envmon: Device type IS NOT the same type indicated at initialization. Device type was XX at initialization.`

The external device configuration read during execution of the *Display Ext. Device Configuration* test does not match the external device configuration saved during board initialization. *XX* is the device type indicated at initialization time. The port number this message applies to should appear on the preceding screen line or in the register definitions in the error log. Either the device type connected to the port was changed after SSID started execution, the signal from the external device to indicate the device type changed, the environmental monitor board assembly is defective, or there is interference from other boards on the VMEbus.

`envmon: Environmental Monitor Transition board is NOT properly connected.`

The transition board connection bit that is read during execution of the *Verify Conn. of Transition Board* test indicates improper connection of the transition board. This could be a loose or defective cable, improper installation, a defective environmental monitor board or transition board, or interference from other boards on the VMEbus.

`envmon: Environmental Monitor board does not exist.`

A bus error occurred when the environmental monitor board's base address was accessed. Either the board is not installed, an incorrect base address was entered using **cf**, the master CPU board is defective, the environmental monitor board assembly is defective, or there is interference from other boards on the VMEbus.

`envmon: Environmental Monitor board initialization failed.`

A failure occurred during initialization of the environmental monitor board. There should be some additional messages displayed to indicate the cause of the failure. Failure to properly complete initialization prevents any test from running on the MVMEenv board. This *may* indicate a defective monitor board assembly or interference from other boards on the VMEbus. If a message indicating the failure was due to the inability to allocate/deallocate memory, semaphores, or vectors is also displayed, the failure is *not* attributable to the environmental monitor board.

`envmon: Environmental Monitor board interrupter test timed out.`

An interrupt from the environmental monitor board was not received within the timeout period after an interrupt was requested by the *Interrupter Test*. This generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

`envmon: Environmental Monitor test` *XX* `timed out.`

The test indicated by XX timed out and did not complete successfully. This *may* indicate a defective monitor board assembly or interference from other boards on the VMEbus.

`envmon: Error in board cleanup.`

An error occurred during board initialization and again when SSID tried to release resources that had been allocated for the board. This could be caused by corruption of memory or SSID structures in memory. It does *not* indicate a failure in the monitor board assembly.

`envmon: Error in interrupt log initialization.`

An error occurred during initialization of the interrupt log. Failure to properly complete initialization prevents any test from running on the MVMEenv board. This could be caused by corruption of memory or SSID structures in memory. It does *not* indicate a failure in the monitor board assembly.

`envmon: Interrupt received but status is not valid.`

An interrupt was received due to a request by the *Interrupter Test* but the status returned to the test was invalid. This could be caused by corruption of memory or SSID structures in memory. It does *not* indicate a failure with the monitor board assembly.

`envmon: Invalid device type indication on port` *XX*.

The monitor board status registers indicated that both a UPS and an external chassis were attached to port number *XX*. Either the device connected to the port is generating an invalid type signal, the environmental monitor board assembly is defective, or there is interference from other boards on the VMEbus.

`envmon: Unable to clear disable powerdown bit on board.`

SSID was unable to clear the environmental monitor board register bit associated with enabling power removal after detection of an overtemperature condition. This generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

`envmon: Unable to clear interrupt enable bit on board.`

SSID was unable to clear the environmental monitor board register bit associated with disabling interrupts from the board. This generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

`envmon: Unable to create interrupt semaphore for Environmental Monitor board.`

SSID was unable to allocate a semaphore during board initialization. This could be caused by corruption of memory or SSID structures in memory. It does *not* indicate a failure in the monitor board assembly.

5

`envmon: Unable to delete board access semaphore.`

An error occurred while attempting to deallocate a semaphore previously allocated to the monitor board tests. This could be caused by corruption of memory or SSID structures in memory. It does *not* indicate a failure with the monitor board assembly.

`envmon: Unable to enable interrupts on board.`

During board initialization, SSID attempts to program the monitor board to enable interrupts from the board. SSID was not able to program the board to enable interrupts. This generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

`envmon: Unable to execute Interrupter test - interrupts disabled.`

SSID is unable to execute the interrupter test because the interrupt enable bit in the board's register is not set. Try enabling interrupts with the "Set Test Flags/Options" menu selection. If you are unable to set the interrupt enable bit, this generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

`envmon: Unable to free vector` *XX*.

An error occurred while attempting to deallocate vector number *XX*, previously allocated to the monitor board. This could be caused by corruption of memory or SSID structures in memory. It does *not* indicate a failure in the monitor board assembly.

`envmon: Unable to get a semaphore.`

An error occurred while attempting to allocate a semaphore for use by the monitor board tests. This could be caused by corruption of memory or SSID structures in memory. It does *not* indicate a failure with the monitor board assembly.

`envmon: Unable to get a vector.`

An error occurred while attempting to allocate a vector through which interrupts from the monitor board would be received. This could be caused if memory or SSID structures in memory have been corrupted. This does *not* indicate a failure with the monitor board assembly.

`envmon: Unable to log error.`

An error occurred while attempting to add error information to the local error log. The error that was being logged will not be displayed by *Display Error Log*. This could be caused by corruption of memory or SSID structures in memory. This particular message does *not* indicate a failure in the monitor board assembly, but it does indicate that some other error did occur and will not be logged.

`envmon: Unable to program board for short timeout.`

During board initialization, SSID attempts to program the monitor board for a short timeout before removal of power after an overtemperature condition is detected. This message indicates that SSID was not able to program the board for a short timeout. Tests continue to execute, but the timeout period before the monitor board removes power after detecting an over-temperature condition is approximately 15 minutes. This failure generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

**`envmon: Unable to program vector register VCTR: 0x`*XXXX*
`REG: 0x`*YYYY*.

SSID was unable to program the monitor board so that it would interrupt through the assigned vector. *XXXX* is the vector number that SSID tried to program into the board. *YYYY* is the value read from the board's vector register after the attempt to write *XXXX* into the register. Failure to properly complete vector register initialization prevents any test from running on the MVMEenv board. This generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

5

`envmon: Unable to set disable powerdown bit on board.`

SSID was unable to set the environmental monitor board register bit associated with disabling the removal of power after detection of an overtemperature condition. This generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

`envmon: Unable to set interrupt enable bit on board.`

SSID was unable to set the environmental monitor board register bit associated with enabling interrupts from the board. This generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

`envmon: Unable to set long timeout on board.`

SSID was unable to clear the environmental monitor board register bit associated with the timeout period before power removal after an overtemperature condition is detected. This generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

`envmon: Unable to set short timeout on board.`

SSID was unable to set the environmental monitor board register bit associated with the timeout period before power removal after an overtemperature condition is detected. This generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

5

MVMEenv Board Interrupt Messages

env.0/envmon: ABORT button was pressed.

The ABORT button on the environmental monitor transition board was pressed. After displaying this message, the console should be displaying the `1xx-Bug>` prompt. Execute **g** RETURN to return to SSID. If the screen display does not change, press RETURN again to redisplay the menu that was on the screen at the time ABORT was pressed. If this message appears when the ABORT button has not been pressed, it could indicate a defective monitor board assembly.

env.0/envmon: Lost AC power to UPS on port *XX*.

An interrupt was received from the monitor board indicating AC power was lost to a UPS attached to port XX. If this message is displayed when a UPS is not indicating loss of AC power, it generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

env.0/envmon: Lost AC power to external chassis on port *XX*.

An interrupt was received from the monitor board indicating AC power was lost to an external chassis attached to port XX. If this message is displayed when an external chassis is not indicating loss of AC power, it generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

env.0/envmon: Low battery in UPS on port *XX*.

An interrupt was received from the monitor board indicating a low battery condition in a UPS attached to port XX. If this message is displayed when a UPS is not indicating low battery power, it generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

env.0/envmon: Overtemperature indication in Delta chassis on sensor *XX*.

An interrupt was received from the monitor board indicating detection of an overtemperature condition by internal temperature sensor XX. If this message is displayed when no overtemperature condition exists, it generally indicates a defective sensor, defective monitor board assembly, or interference from other boards on the VMEbus.

env.0/envmon: Overtemperature indication in external chassis on port *XX*.

An interrupt was received from the monitor board indicating detection of an overtemperature condition in the external chassis attached to port XX. If this message is displayed when an external chassis is not indicating it has detected an overtemperature condition, it generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

`env.0/envmon: Received interrupt from Environmental Monitor board with unknown or invalid cause indication.`

An interrupt was received from the monitor board but no cause was indicated when the board's status register was read by SSID. This generally indicates a defective monitor board assembly.

`env.0/envmon: Received test interrupt from Environment board when no test interrupt was requested.`

An interrupt was received from the monitor board indicating the interrupt was due to a test request, but no test interrupt had been requested. This generally indicates a defective monitor board assembly or interference from other boards on the VMEbus.

MVMEenv Board Register Definitions

STATA REGISTER BITS

Bits 0 - 3

When set to a 1 indicate, for sensors 1 - 4 respectively, that an overtemperature condition has been detected by the respective internal (Delta chassis) sensor.

Bits 4 - 7

When set to a 1 indicate, for ports 1 - 4 respectively, a low battery condition on the device attached to the port (if the device is a UPS) or an overtemperature condition (if the device is an external chassis).

Bits 8 - 11

When set to a 1 indicate, for ports 1 - 4 respectively, an AC fail condition (if the device attached to the port is a UPS) or a power loss condition (if the device is an external chassis).

Bit 12

Not used.

Bit 13 When set to a 1, indicates the cause of the interrupt is a request for a test interrupt.

Bit 14

When set to a 1, indicates the ABORT button on the environmental monitor transition board was pressed.

Bit 15

When set to a 1, indicates the environmental monitor transition board is *not* properly connected.

STATB REGISTER BITS

Bits 0 - 3

When set to a 1, indicate the device attached to ports 1 - 4, respectively, is an external chassis.

Bits 4 - 7

When set to a 1, indicate the device attached to ports 1 - 4, respectively, is an external chassis.

Bits 8 - 15

Not used.

COMMAND REGISTER BITS

Bit 0

When set to a 1, turns off the internal (Delta chassis) power supply.

Bit 1

When set to a 1, requests that the external device shut down power.

Bit 2

When set to a 1, generates a VME SYSRESET signal on the VMEbus.

Bit 3

When set to a 1, indicates the short overtemp timeout is selected.

Bit 5

When set to a 1, disables the ability of the environmental monitor board to remove power from the Delta chassis after an overtemperature timeout has occurred.

Bit 6

When set to a 1, indicates that interrupts from the environmental monitor board are enabled.

Bit 7

When set to a 1, indicates a pending interrupt; also used to request a test interrupt from the environmental monitor board.

Bits 4 and 8 - 15

Not used.

VECTOR REGISTER BITS

Bits 0 - 7

Indicate the interrupt vector through which the environmental monitor board will interrupt.

Bits 8 - 15

Not used.

CHAPTER 6
MASS STORAGE DEVICE CONTROLLER BOARD TESTS

This chapter describes the board tests available for mass storage device controllers. Each section describes the tests available for a particular controller board, the board's test configuration, and, in some cases, the error codes associated with the board. For an introduction to board testing, refer to the *Board Testing* section in the *Confidence, Fault, Board, and Peripheral Testing* chapter.

MVME320 Winchester/Floppy Controller

6

The MVME320 board tests verify controller operation by performing read-only tests on the Winchester disk drive and read only or write/read tests on the floppy disk drive. The floppy tests are set up for double-sided, double-density, 80-track floppies. You can test both 655Kb low speed floppy disks and 1.2Mb high speed floppy disks. If you are testing 1.2Mb floppies, you must change the default floppy disk configuration. (Refer to the **cf** command, Example 4, in Appendix A for details.)

You can install a maximum of two MVME320 controllers in a system. The Winchester tests are designed for Micropolis (default) and Toshiba drives. If you have a Toshiba drive, use the **cmdline** system command to change the Winchester drive type. (Refer to the **cf** command, Example 2, in Appendix A for details.)

Winchester Disk Tests

These tests are all nondestructive read-only tests. You must format the disk drives with SYSTEM V/68 format which includes the bad track list for software bad track support. All read tests check the bad track list for the current track before performing the read operation. If the current track is found in the bad track list, the current track is not read and the test continues to the next track.

To access the test menus for the Winchester and floppy disk drives, type:

```
SA: 320.0
```

A display similar to the following appears:

```
Current Menu is /bdtest/320.0 - "VME320 Disk Controller Tests"
hd.1                - Winchester Disk
hd.0                - Winchester Disk
fd.1                - Floppy Disk
fd.0                - Floppy Disk
SA: hd.0 <CR>   (to access the test selection menu for the first Winchester disk)
(The following menu is displayed for hd.0 and hd.1)
Selections for Test "Winchester Disk"
   0) Recalibrate
   1) Quick confidence read
   2) Read entire OS disk
   3) Random read OS disk
   4) Ping-pong read OS disk
   d) Display OS bad tracks
SA:
```

6

The following is a brief explanation of the tests:

`0) Recalibrate`
Causes the drive to recalibrate its track counter by issuing the restore command to the controller; it then waits until the heads reach track zero.

`1) Quick confidence read`
Seeks and reads tracks in the following sequence: track zero, last track, track zero. This is a quick check of drive and controller functions.

`2) Read entire OS disk`
Reads the entire disk sequentially from track zero to the last track.

`3) Random read OS disk`
Reads tracks picked at random.

`4) Ping-pong read OS disk`
Reads the entire disk in the following sequence: from track zero to last track, from track one to second to last track, from track two to third to last track, and so on.

d) `Display bad track list`
Reads and displays the bad track list from the disk.

Floppy Disk Tests

Some of these tests destroy the data on the scratch floppy disk. You must format all floppy disks for these tests using SYSTEM V/68 or the SSID format test, selection `f`; they do not need bad track support. Since the first track is formatted by the operating system to single density, this track is not used in any of the floppy tests.


```
SA: fd.0 <CR>          (The following menu displays for fd.0 and fd.1)
Selections for Test "Floppy Disk"  (write-enabled scratch floppy disk required)
   0) Recalibrate
   1) Quick confidence read
   2) Read entire OS Disk
   3) Random read Disk
   4) Ping-pong read Disk
   p) Partial W/R DESTRUCTIVE
   w) Fully W/R DESTRUCTIVE
   f) Format Disk
SA:
```

For a description of tests `0` through `4`, refer to the previous section, *Winchester Disk Tests*. The following describes tests `p`, `w`, and `f`:

p) `Partial W/R DESTRUCTIVE`
Writes a pattern, reads it back, and verifies the pattern on a small number of tracks.

w) `W/R DESTRUCTIVE`
Writes a pattern, reads it back, and verifies the pattern on every track except the first track.

f) `Format Disk`
Formats the entire disk (including the first track) based on the variables found in the floppy disk configuration. The following section, *MVME320 Board Test Configuration*, shows the default 655Kb floppy disk configuration.

Refer to the **cf** command, Example 4, in Appendix A for information on how to change the floppy disk configuration to accommodate 1.2Mb floppy disks.

6

MVME320 Board Test Configuration

TThe MVME320 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 320.0
/bdtest/320.0 configuration
   base[%0x400] = 0xffffb000 ?                   # base address
   am[0x0d,0x3d] = 0x3d ?                        # address modifier
   level[0-7] = 5 ?                              # default interrupt level
   data bus width[b,w,l] = l ?                   # date bus width (byte, word, long)
   type 0Micropolis 1Toshiba[0,1] = 0 ?          # type of hard disk
   fd[0-2] = 2 ?                                 # number of floppy disks
   hd[0-2] = 2 ?                                 # number of hard disks

> cf 320.0/hd.0
/bdtest/320.0/hd.0 configuration
   Drive type[MICR] = MICR ?                     # drive type Micropolis
   Micropolis-phy-drive[0-1] = 0 ?               # hard disk drive number
   M-test type[r,w] = r ?                        # read only test or
   M-sector/xfer[1,2,4,8,16,32] = 32 ?           # size per buffer
   M-retries[0-20] = 4 ?                         # retry count
   M-step rate(ms) = 0 ?                         # step rate
   M-recal step rate(ms) = 5 ?                   # recalibrate step rate
   M-format with[ECC,CRC] = ECC ?                # ECC or CRC
   M-buffered[y,n] = y ?                         # drive buffered
   M-post data gap =0xf ?                        # post data gap
   M-addr mark cnt = 1 ?                         # address mark count
   M-sectors/track = 32 ?                        # number of sectors per track
   M-tracks/cylinder = 8 ?                       # number of tracks per cylinder
   M-cylinders/drive = 1024 ?                    # number of cylinders per drive
   M-sector size[128,256] = 256 ?                # number of bytes per sector
   M-interleave factor[1-16] = 3 ?               # interleave factor
   M-precomp track = 1025 ?                      # precompensation
   M-low wrt current trk = 512 ?                 # low write current
   M-concurrent[y,n] = y ?                       # allow concurrent, no lock
   Drive type[TOSH] = TOSH ?                     # drive type Toshiba
   Toshiba-phy-drive[0-1] = 0 ?                  # hard disk drive number
   T-test type[r,w] = r ?                        # read only test or
   T-sector/xfer[1,2,4,8,16,32] = 32 ?           # size per buffer
   T-retries[0-20] = 4 ?                         # retry count
   T-step rate(ms) = 0 ?                         # step rate
```

```
    T-recal step rate(ms) = 5 ?                # recalibrate step rate
    T-format with[ECC,CRC] = ECC ?             # ECC or CRC
    T-buffered[y,n] = y ?                      # drive buffered
    T-post data gap =0xf ?                     # post data gap
    T-addr mark cnt = 1 ?                      # address mark count
    T-sectors/track = 32 ?                     # number of sectors per track
    T-tracks/cylinder = 10 ?                   # number of tracks per cylinder
    T-cylinders/drive = 830 ?                  # number of cylinders per drive
    T-sector size[128,256] = 256 ?             # number of bytes per sector
    T-interleave factor[1-16] = 3 ?            # interleave factor
    T-precomp track = 512 ?                    # precompensation
    T-low wrt current trk = 0 ?                # low write current
    T-concurrent[y,n] = y ?                    # allow concurrent, no lock
> cf 320.0/fd.0
/bdtest/320.0/fd.0 configuration  (for a 655Kb floppy disk)
    phy-drive[2-3] = 2 ?                       # floppy drive number
    test mode[r,w] = r ?                       # read only test or
    sector/xfer[1-16] = 16 ?                   # size per buffer
    retries[0-20] = 10 ?                       # retry count
    step rate(ms) = 3 ?                        # step rate
    post data gap = 0x36 ?                     # post data gap
    addr mark cnt = 3 ?                        # address mark count
    size(inch)[5,8] = 5 ?                      # 5-1/4 inch or 8 inch
    sides[1,2] = 2 ?                           # number of sides
    no of cylinder = 80 ?                      # number of tracks per side (DEBUG)
    track density[s,d] = d ?
    no of sector/track = 16 ?
    sector size[128,256,512] = 256 ?           # number of bytes per sector
    interleave factor[1-16] = 1 ?              # interleave factor
    precomp track = 0x50 ?                     # precompensation
    low wrt current trk = 0x50 ?               # low write current
> cf 320.0/fd.0
/bdtest/320.0/fd.0 configuration  (for a 1.2Mb floppy disk)
    phy-drive[2-3] = 2 ?                       # floppy drive number
    test mode[r,w] = r ?                       # read only test or
    sector/xfer[1-16] = 16 ? 15                # size per buffer
    retries[0-20] = 10 ?                       # retry count
    step rate(ms) = 3 ?                        # step rate
    post data gap = 0x36 ?                     # post data gap
    addr mark cnt = 3 ?                        # address mark count
    size(inch)[5,8] = 5 ? 8                    # 5-1/4 inch or 8 inch
    sides[1,2] = 2 ?                           # number of sides
    no of cylinder = 80 ?                      # number of tracks per side (DEBUG)
    track density[s,d] = d ?
    no of sector/track = 16 ?  15
```

```
sector size[128,256,512] = 256 ? 512 # number of bytes per sector
interleave factor[1-16] = 1 ?        # interleave factor
precomp track = 0x50 ?               # precompensation
low wrt current trk = 0x50 ?         # low write current
```

MVME320 Winchester/Floppy Controller Soft Errors

The MVME320 controller tests are configured to retry four times on soft errors (i.e., Cyclic Redundancy Check (CRC) errors, Identifier not found, Positioning errors, and Throughput errors on Direct Memory Access (DMA) operation). If one of these errors occurs, the test retries the configured number of times until the operation is successful or until the retry count expires, causing a FATAL error to display.

In the case of a CRC error, the controller hardware retries a configured number of times. The SSID software retries on CRC errors only after the controller hardware has retried the configured number of times.

The error counter is incremented after every retry in which a soft error is found; however, the fatal error counter is not incremented if the test completes successfully. Refer to *The clear Command* section in the *System Commands* chapter for a sample error display.

MVME323 ESDI Disk Drive Controller

The MVME323 board tests verify controller operation by performing both nondestructive *read only* tests and destructive *write/read/verify* tests on the ESDI Winchester disk drives.

You can install a maximum of one MVME323 controller in a system, with a maximum of four fixed ESDI Winchester drives per controller. You must format the disks to be tested.

The MVME323 test automatically configures itself for the correct number of sectors per track, number of heads, and number of cylinders. The test also accounts for a slip-sectored disk. To account for slip sectoring requires the disk drive's Disk Configuration Block to have the slip sector count something other then zero. The **cf** information is overridden by the test for these fields.

The **323.0** directory contains tests for the 323 controller. To run tests on the first ESDI drive, enter **esdi.0**; enter **esdi.1**, **esdi.2**, or **esdi.3** to run tests on the second, third, or fourth ESDI drive, respectively. These commands display a menu of tests you can run for this controller.

When a test encounters an error, it displays the command that failed, the logical sector tested, and other information about the controller and drive status.

To access the test menus for the MVME323 controller, type:

`SA:` **323.0**

A display similar to the following appears:

```
Current Menu is /bdtest/323.0 - "VME323 ESDI Controller Tests"
esdi.3         - ESDI Disk Tests
esdi.2         - ESDI Disk Tests
esdi.1         - ESDI Disk Tests
esdi.0         - ESDI Disk Tests
SA: esdi.0 <CR>    (to access the test selection menu for the first ESDI disk drive)
Selections for Test "ESDI Disk Test"
   0) Recalibrate
   1) Confidence Test
   2) Read Entire Disk
   3) Random Read Disk
   4) Ping-Pong Read Disk
   a) Read Sector Header
   b) Controller Check
   c) Get uib from Drive
   d) Read sector x
   e) Calculator
   r) Read Disk Descriptor
SA:
```

For a description of tests 0 through 4, refer to the *Winchester Disk Tests* section. Menu items `a` through `e` are tools available to isolate disk problems:

`a) Read Sector header`
Displays any arbitrary logical and physical sector number from the current cylinder and head position on the disk.

`b) Controller check`
Performs transfers between the host and the controller to verify proper controller function.

c) **Get uib from controller**
Displays the current drive parameters from the unit initialization block.

d) **Read sector x**
Allows the reading of any sector on the disk. It accepts either hexadecimal or decimal input.

e) **Calculator**
Converts logical sector numbers into cylinder, head, physical sector numbers. The calculator works with either hexadecimal or decimal numbers.

The following display shows how the calculator works:


```
/bdtest/323.0/esdi.0  c) Calculator             ........starts
Calculator
1 = logical sector
2 = physical
q = quit
<: 1  (to calculate the logical sector)
An 'e' to exit or Enter Cylinder number : 1 <CR>
An 'e' to exit or Enter Head/Track number : 1 <CR>
An 'e' to exit or Enter Physical sector number : 1 <CR>

Logical sector number = hexadecimal 0x00000169 decimal 361
<: 2  (to calculate physical sector number)
An 'e' to exit or Enter Logical sector number : 1 <CR>

Cylinder number     = hexadecimal 0x00000000 decimal 0
Head/track number   = hexadecimal 0x00000000 decimal 0
Physical sector     = hexadecimal 0x00000001 decimal 1
<: q <CR>
```

r) **Read Disk Descriptor**
Displays device descriptor information for the drive requested. This information is from the ESDI drive itself.

The following destructive tests are available for hard disks, but you must enter command line mode and execute **cf** to set the `eval flag` before you can run these tests on a hard disk:

> **CAUTION**
>
> **The following selections destroy data on the disk: Fix BAD Spot, Format Track, Map Alternate Track, Format Entire Disk, and Full W/R Destructive.**

Fix Bad Spot

This selection includes the following:

Fix Bad Spot

Fixes a bad spot on the disk without reformatting the entire disk by mapping the track that contains the bad spot to an alternate track. Requires the user to input the physical cylinder and head to be fixed. Fix bad spot will automatically select an alternate track.

Display Grown Defect List

Displays the defect list stored on the disk.

Format Track

Format a specified track.

Map Alternate Track

Map a bad track to an alternate track. Maps a bad track on the disk without reformatting the entire disk by mapping the track that contains the bad spot to an alternate track. Requires the user to input the physical cylinder and head to be fixed and the physical cylinder and head of where the alternate track is at.

Format Entire Disk

Formats the disk as specified by the disk descriptor parameters using alternate track mapping.

Full W/R DESTRUCTIVE

Sequentially writes a data pattern over the entire disk, then sequentially reads and verifies the pattern.

6

Set Test Scripts
Allows you to:

- set the last cylinder for testing.
- set the data pattern used for write/read/verify testing.
- set seek or read test mode.
- enable verify option for read testing.
- set the number of random seeks for speed tests.
- modify confid string 5 (Continuous Running Intensive I/O Check) for `esdi.0` to include destructive disk tests. The destructive selection is TOGGLED each time this selection is executed. If destructive mode is enabled, the following message displays:

```
CAUTION: Destructive disk tests for drive 0 have been
         added to confidence and fault test scripts!!!
```

If destructive mode is disabled, the following message displays:

```
DRIVE 0 TEST SCRIPTS ARE NOW READ ONLY
```

Destructive test WARNING messages are disabled during confid 5 testing to allow unattended destructive tests.

Read Sectors From X
Reads the specified number of sectors from the specified position on the disk.

Seek for Speed
Displays the time required to seek randomly on the disk for a specified number of seeks.

Display O/S ALT-list
Displays the alternate track or alternate track slip sector list from the disk drive. This list is created by the operating system's format program; SSID cannot create or modify this list.

The following abbreviations may display during the tests:

Control and Status Register		Others	
sled	status led state	uib	unit initialization block
bok	board okay	iopb	input/output parameter block
sfdis	sysfail enable	SSE	spare sector enable
bdclr	board clear	CE	cache enable
berr	bus error enable	SC_int_on	status change interrupt
go_busy	board running command		
done	operation done interrupt		
err_last	error last command		

MVME323 Board Test Configuration

The MVME323 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 323.0
/bdtest/323.0 configuration
   base[%0x200] = 0xfffa000 ?             # base address
   am[0x0d,0x3d] = 0x0d ?                 # address modifier
   level[0-7] = 4 ?                       # default interrupt level
   dbw[b,w,l] = w ?                       # data bus width (byte, word, long)
   esdi[0-4] = 4 ?                        # number of drives
```

> **CAUTION**
>
> **Setting the `eval flag` variable to `1-on` allows destructive disk testing!**

```
> cf 323.0/esdi.0
/bdtest/323.0/esdi.0 configuration
   Controller type [ESDI,SMD] = ESDI ?
   phy-drive[0-3] = 0 ?                   # number of hard disk drives
   Alt track 0-no 1-yes[0-1] = 1 ?        # alternate track handling
   sector/xfer[1-128] = 36 ?              # test buffer size
   dma burst[1-255] = 10 ?                # dma burst rate
   start head = 0 ?
   tracks/cylinder = 9 ?                  # number of tracks per cylinder
   sectors/track = 36 ?                   # number of sectors per track
```

```
spiral skew = 2 ?
sector size[128,256,512,1024] = 512 ?   # number of bytes in a sector
gap1 = 10 ?
gap2 = 10 ?
interleave factor[1-16] = 1 ?           # interleave factor
retries[0-255] = 3 ?                    # retry count
cylinders/drive = 968 ?                 # number of cylinders per drive
attribute = 0x5 ?                       # inc by head, reSeek
format with[ECC,NON] = ECC ?            # format with ECC or without ECC
Eval Flag 0-off 1-on[0-1] = 0 ?         # not recommended for general use
lockout cylinder = 0 ?                  # ending range of cylinder tested
lockout logical sector number = 0 ?     # calculated log. position of
                                        # lockout cyl.
read/seek flag 0-read 1-seek[0-1] = 0 ?
read/check flag 0-no 1-yes[0-1] = 0 ?
write data pattern = 0xdb60db6 ?
Max number of random seeks = 10000 ?    # applies to speed test only
Max number of alternate tracks = 144 ? # automatically determined by test
Conf test flag 0-off 1-on[0-1] = 0 ?    # confidence test flag for
                                        # destructive disk tests
```

MVME323 Error Codes

The following MVME323 error codes may be returned while running the MVME323 tests. All error codes are in hexadecimal format.

10 - Drive not ready

The disk ready signal output is tested at the beginning of any command requiring disk data movement. Error 10 is posted if the disk is not ready. This code is typically posted when an attempt is made to access a disk before the MVME323 has received the READY signal for the drive.

11 This error code is not used.

12 - Seek error - wrong cylinder

If the MVME323 cannot find the required sector of data within two revolutions, it tries to verify that the head is on the right track by reading several sectors. If the cylinder number in the header is incorrect, then error 12 is issued (refer to related errors 29 and 2A).

13 - ECC code discrepancy

Issued if the computed ECC on the data did not agree with the ECC appended to the data on the disk, and if no error correction was attempted (refer to related error 23).

14 - Invalid command code (byte 0 of IOPB)
The command code in the IOPB (Input/Output Parameter Block) (byte 0) was not valid.

15 - Illegal fetch and execute attempted
Indicates that a Fetch and Execute command was encountered in external memory. A Fetch and Execute command is only valid when it occurs in the onboard Short I/O space.

16 - Sector number too large for drive
The target sector in the IOPB (byte 7) was greater than the capacity of the drive as specified for that drive in byte 4 of the UIB. This check is performed before the command is executed.

17 - Illegal memory type specified
Either the memory type specified for the buffer address is not 0, 2, or 3; or the IOPB address is not 0, 1, 2, or 3 as required.

18 - Bus not available in < 1 ms
Indicates that bus acquisition was not completed within 100 milliseconds of a request. This error is typically caused by a nonexistent address or address modifier in the data transfer IOPB.

19 - Header checksum error
Indicates that there was an error in the header field.

1A - Disk write protected
Issued when attempts are made to write to a disk that is write protected.

1B - No response from unit on select
Issued when a unit select was made and the unit failed to respond with Unit Selected. This occurs when either the drive unit number is incorrectly selected, the drive is not powered up, or the cable is not properly connected.

1C - No correct seek 3 secs after RESTORE
Occurs when a **Clear Fault** or **Restore** failed to correct a seek error from the drive within three seconds. If this error is issued, check to make sure the "B" cable is connected correctly.

1D - Fault not cleared 3 secs after RESTORE
Issued when a **Clear Fault** or **Restore** failed to correct a fault condition from the drive within three seconds. If this error is issued, check to make sure the "B" cable is connected correctly.

1E - Drive fault occurred, RESTORE required
Indicates that a fault condition exists in a selected unit. The fault should be cleared by a **Restore** command. This error is issued when the drive tries to access a nonexistent head or cylinder. Check the drive manual to ensure that the UIB contains the proper settings.

1F - Drive not ready 3 secs after RESTORE
Issued when a **Clear Fault** or **Restore** failed to bring the drive ready within three seconds.

20 - Multi sector I/O exceeded end of media
Indicates that a multisector transfer exceeded the end of the medium.

21 - UIB for specified volume contains zeros
This fault indicates that the volume specified in the IOPB contains zero heads in the UIB. This error is usually caused by an error in the UIB.

22 - Bad post header pad byte found
Indicates that an improper post-header pad byte was encountered.

23 - Failed to fix uncorrectable error
When this error is posted, error correction was attempted on the data field and the error was found to be uncorrectable.

24 - Logical sector num -> bad cylinder num
Indicates that the translation of a logical sector resulted in a bad cylinder number. If the drive's UIB is correct, then the logical sector is invalid.

25 - Logical sector num -> bad head num
Occurs when the translation of a logical sector resulted in a bad head number. If the drive's UIB is correct, then the logical sector is invalid.

26 - Logical sector num -> bad phys sector num
When posted, this error indicates that the translation of a logical sector resulted in a bad physical sector number. If the drive's UIB is correct, then the logical sector is invalid.

27 - Data overrun - suspect clock from drive
Indicates a data timeout error usually caused by a missing TX (transmit) or RX (receive) clock from the drive. If this error is issued, check to ensure the "B" cable is connected correctly. It may also occur if the UIB sectors/track is set to 0.

28 - No index pulse after 65ms on format cmd
During a Format operation, the MVME323 looks for the index pulse from the disk drive. If not found within 65 milliseconds, this error is posted.

29 - Sector not found during read or write
If the target sector cannot be found during a **Read** or **Write**, this error is issued (refer to related errors 12 and 2A). It is also issued if a bad sector on the disk is encountered or if a track is improperly formatted.

2A - Head number in header wrong
Issued if the head number read from the disk in the header field was wrong (refer to related errors 12 and 29).

2B - Sync word in data field not valid
Indicates that the first word read from the data field was not a valid sync character.

2C - No valid header found
Indicates that during the Read Header command, no valid header was found. After checking every sector (specified by the UIB) including the runt sector and short sector, every header was invalid. This means that the sync character, checksum, and/or post-header pad were invalid. This error is usually posted when attempting to read a disk that was not formatted for use with the MVME323.

2D - Seek not complete in 500ms
If issued, this error indicates that a seek was made and a normal complete response did not occur within 500 milliseconds.

2E - Drive held by dual port over 500ms
This error is set on a dual ported drive if Busy has been active for more than 500 milliseconds. It indicates that one of the controllers has held the drive too long.

2F - Not on cylinder 3 secs after select
The drive must be on cylinder within three seconds after being selected, or this error results.

30 - Not normal complete 3 secs after RESTORE
Issued when a **Restore** command was executed but a normal complete did not occur within three seconds.

31 - Sync word in header not valid
An invalid sync character in the header field causes this error code to occur.

32 through **3D**
These error codes are not used.

6

3E - UIB skew factor
Occurs if the skew factor set in the UIB exceeds the sectors per track specified in byte 4 of the UIB (plus any spares specified in byte 13 of UIB).

3F This error code is not used.

40 - Write or format attempted before INIT_DRV
Indicates that a **Write** or **Format** command was attempted on a unit that has not been initialized.

41 This error code is not used.

42 - Specified gap too small - minimum gap is 5
Occurs when the value for either Gap 1 or Gap 2 in the UIB is less than five.

43 through **4A**
These error codes are not used.

4B - Seek error reported by drive
Indicates that a seek error was reported by the disk drive.

4C - No sector pulse on track to be mapped
No sector pulse was found and the track cannot be mapped if no sector pulse is present.

4D through **4F**
These error codes are not used.

50 - Sectors per track in UIB invalid
Indicates that the number of sectors/track set in the UIB is zero or greater than 160.

51 - Bytes per sector in UIB > buffer size
The bytes per sector are specified in bytes 6 and 7 in the UIB. When the number is less than 256 or greater than 2048, this error is issued.

52 - Invalid interleave in UIB > SPT or 0
If this error occurs, the interleave factor set in byte 6 of the UIB is either zero or greater than the number of sectors per track. If this error is issued, check to make sure that the UIB pointer in the initialize command is pointing to the correct place in memory. If it is not, a UIB of random data is read during initialization.

53 - Head number in IOPB out of range per UIB

The capacity of the drive is specified in bytes 0 through 3 of the UIB. This error indicates that the target head address in byte 6 of the IOPB exceeded the capacity of the drive.

54 - Cylinder no. in IOPB out of range per UIB

The capacity of the drive is specified in bytes 12 through 13 of the UIB. This error indicates that the target cylinder in Word 2 of the IOPB exceeded the capacity of the drive.

55 - ESDI timeout

Occurs if an MVME323 command does not complete within the time limit indicated by the ESDI specification.

56 through 5C

These error codes are not used.

6

5D - Invalid DMA transfer count

Indicates that the specified transfer count caused the MVME323 to attempt to transfer an odd number of bytes.

5E and 5F

These error codes are not used.

60 - Bus error on external IOPB transfer

When this error is posted, a bus error occurred during the transfer of an external IOPB. The IOPB pointer (Words 10 and 11) shows the starting address of the IOPB on which the bus error occurred. (Refer to error code 61 for details on bus errors during DMA transfers.)

61 - Bus error on DMA transfer

Indicates that a bus error occurred during the DMA transfer of data to or from the buffer or the bus. Words 5 and 6 of the IOPB (the buffer address) point to the start of the sector block in system memory where the error occurred. Words 2 and 3 of the IOPB (the disk address) point to the disk location where the disk transfers were when the error occurred. If the disk was addressed in physical mode, the disk error location is a physical location. If logical address mode was used, the disk error location is a logical location.

62 - VME buffer address not aligned

For 8- or 16-bit transfers, the starting address of the VME buffer must fall on a word boundary (even address, multiple of two). For 32-bit transfers, the starting address of the VME buffer must fall on a longword boundary (even address, multiple of four).

`63` through `69`
These error codes are not used.

`6A - Unrecognized header field`
During a requested read/verify command, one or more of the requested headers were not found. This error differs from error 29 (**SECTOR NOT FOUND**) in that one or more headers were ignored because of invalid sync, checksum, or post-header pad fields. Possible causes could include unformatted sectors (UIB sectors/track less than disk sectors/track), UIB Gap one too small, or *short sector* pulse from drive.

`6B - Mapped header error`
Indicates that the sync field of a header appeared to be a valid mapped field, but the remainder of the header was unrecognizable. (Refer to Error code 6A for probable causes.)

`6C` through `6D`
These error codes are not used.

`6E - Spare sector already in use`
Indicates that the spare sector number to be mapped is beyond the end of the track. The spare sector number must be less than the number of sectors/track.

`6F - UIB indicates no spares enabled`
A Map Sector command was issued and the UIB did not specify spare sector mapping.

`70` through `76`
These error codes are not used.

`77 - Command aborted`
Indicates the MVME323 observed and serviced the ABORT bit in the CSR.

`78 - ACFAIL detected`
Indicates the MVME323 received the VME ACFAIL control signal. This error requires a Reset and ACFAIL release before normal operation can continue.

`79` through `7F`
These error codes are not used.

`80 - Transfer assertion timeout`
The ESDI command transfer acknowledge signal was not returned within the specified time limits.

81 - Transfer release timeout
The ESDI command transfer release signal was not returned within the specified time limits.

82 - Status transfer timeout
The ESDI status transfer acknowledge signal was not returned within the specified time limits.

83 - Status release timeout
The ESDI status release acknowledge signal was not returned within the specified time limits.

84 through **9F**
These error codes are not used.

A0 - S/G list too large
The Scatter/Gather list size exceeds the sector size. This list cannot be larger than a sector buffer. List size is calculated by multiplying the list element size (8 bytes) by the number of elements specified in byte 27 of the IOPB.

A1 - Illegal element byte count
For Scatter/Gather commands (A1 and A2), the element byte count must be a multiple of the bytes/sector parameter. If an illegal element byte count is specified, this error is returned by the Scatter/Gather commands code.

A2 through **AA**
These error codes are not used.

AB - Illegal element size
All Scatter/Gather elements must contain an even number of bytes. The controller cannot transfer less than a word; therefore, if an element contains an odd number of bytes, this error is returned.

AC - Illegal list byte count
The total byte count specified by the Scatter/Gather list must be a multiple of the bytes/sector parameter. The controller will not complete a Scatter or Gather command with partial sectors, and this error is returned.

AD - Illegal IOPB sector count
The IOPB sector count does not agree with the total bytes specified in the Scatter/Gather list.

AE through **BF**
These error codes are not used.

`C0 - Both bits set`
Returned if both the Spare Sector Enable bit and the Multiple Spare Enable bit are set.

`C1 - MSE without init long`
Returned if the Multiple Spare Enable bit is set and the unit is not Initialized with the Initialize Long command (7C).

`C2` through `EF`
These error codes are not used.

`F0 - Mapped header`
A mapped bad header was encountered on the current track.

`F1 - Sector not flagged`
The sector to be transferred was not flagged as error 29 in sector ID table (rdeach @ rdca100:).

`F2` through `FB`
These error codes are not used.

`FC - No write list`
No write list was available to start gather operation.

`FD - No write buffers`
No write buffers were available for gather operation.

`FE - Out of buffers`
All buffers were full before gathered data could be transferred.

`FF - Command not implemented`
The command issued will be supported in a later release.

`XX - Unexpected error code` *XX*
This is any error code not defined above that might occur due to erroneous operation. Contact your sales or field service representative for assistance.

MVME327 SCSI Disk Drive Controller

The MVME327 board tests verify controller operation by performing the following types of tests:

The SCSI tape tests verify SCSI tape controller operation by performing write and read tests on the SCSI tape drives. A write-enabled tape is required.

The SCSI hard disk tests verify SCSI disk controller operation by performing nondestructive, read-only tests on the SCSI hard disk drives. The SCSI disks to be tested must be initialized (formatted or have an operating system file system on them). Refer to the operating system documentation for information on formatting.

6

CAUTION

Do not use the format command in the SCSI test to format the drive. This test is for factory use only.

The local floppy disk tests verify local disk controller operation by performing destructive, write/read tests and nondestructive, read-only tests on the local floppy disk drives. The local floppy disks to be tested must be initialized (formatted or have an operating system file system on them). Refer to the operating system documentation for information on formatting or select the format test from the floppy drive menus.

NOTE

The MVME327 currently only supports local floppy drives and does not support SCSI floppy drives.

To access the test menus for the MVME327 board, type:

`SA:` **327.*n***

where *n* is the controller number, e.g., *0* for the first MVME327, *1* for the second board.

A display similar to the following appears:

```
Current Menu is /bdtest/327.n - VME327 SCSI Tests
/bdtest/327.0/flop.1      - 327 Floppy Disk
/bdtest/327.0/flop.0      - 327 Floppy Disk
/bdtest/327.0/scsi6       - 327 SCSI Ctrlr 6 LUN 0 Tests
/bdtest/327.0/scsi5       - 327 SCSI Ctrlr 5 LUN 0 Tests
/bdtest/327.0/scsi4       - 327 SCSI Ctrlr 4 LUN 0 Tests
/bdtest/327.0/scsi3       - 327 SCSI Ctrlr 3 LUN 0 Tests
/bdtest/327.0/scsi2       - 327 SCSI Ctrlr 2 LUN 0 Tests
/bdtest/327.0/scsi1       - 327 SCSI Ctrlr 1 LUN 0 Tests
/bdtest/327.0/scsi0       - 327 SCSI Ctrlr 0 LUN 0 Tests

SA: scsi0<CR>             (to access the first SCSI hard disk test)

Selections for Test "327 SCSI Ctrlr 0 LUN 0 Tests"

  0) Recalibrate            (reads sector zero on the disk)
  1) Confidence test        (reads last sector, then first sector)
  2) Read entire Disk       (reads every sector on the disk)
  3) Random read Disk       (randomly reads most sectors on the disk)
  4) Ping-Pong read Disk    (reads start, then end, then start + 1, then end - 1, etc.)
  r) Read a Sector          (allows reading of any sector; accepts hexadecimal
                             or decimal input)
  c) Read Capacity          (displays drive capacity in sectors)
  i) Inquiry                (displays drive vendor and model)
```

The following is a brief explanation of the tests:

0) Recalibrate
Causes the drive to read sector zero. This has the effect of moving the head to cylinder zero.

1) Disk Confidence Test
Read the first then the last sector on the disk. Each read is done twice and the results compared.

2) Read entire disk
Reads the entire disk sequentially from sector zero to the last sector. Each read is done twice and the results compared.

3) `Random read disk`
Reads randomly selected blocks of sectors. Most of the available sectors are read. Each read is done twice and the results compared.

4) `Ping-Pong read disk`
Reads the entire disk in the following sequence: sector zero, the last sector, sector one, sector (last - 1), sector two, sector (last - 2), and so on, until every sector on the disk has been read twice, once from each direction. Every read is done twice and the results compared.

r) `Read a Sector`
Requests user input of a sector number to be read, verifies that the sector number is valid, and if so, reads the sector, rereads the sector, and compares the results.

c) `Read Capacity`
Displays the drive capacity, in sectors. The display is in both hex and decimal.

NOTE

SCSI hard disk drives of the same type will not always have the same number of available sectors, but the numbers should be close in value (e.g. within 200 sectors of each other).

i) `Inquiry`
Displays the drive type, vendor ID, product ID, product revision level, and vendor unique data field obtained by performing a SCSI Inquiry command.

MVME327 SCSI Tape Tests Menu

To execute any SCSI tape test suite, select the appropriate SCSI controller from the `327.n/scsim` menu and the following menu displays:

```
SA: 327.n/scsim <CR>  (to access the first SCSI tape test)

Selections for Test "327 SCSI Ctrlr 4 LUN 0 Tests"
(write-enabled scratch tape required)

    0) Init               (returns tape to BOT)
    1) Write Log/EOF      (short write test)
    2) Read Log/EOF       (short read test; run after Write Log/EOF test)
    r) Retension          (fast forward, then rewind the tape)
    s) Erase              (remove ALL data from the tape and rewind)
    t) Write EOT          (long write test)
    v) Read EOT           (long Read test; run after Write EOT test))
    g) Confidence Test    (verifies SCSI interface to drive)
    i) Inquiry            (displays drive vendor and model)

SA:
```


where:

n is the board number.

m is the device number.

0) `Init`
Tests that the tape drive is ready and contains a tape. If so, a rewind is done. (BOT means Beginning Of Tape.)

1) `Write Log/EOF`
At the beginning of media, this test writes a file consisting of 2 blocks of data and an EOF (End-of-File) mark. The tape is then rewound. This requires a write-enabled tape.

2) `Read Log/EOF`
At the beginning of media, this test reads a file consisting of 2 blocks of data. The data read from the tape is compared with the data patterns written in test (1). This requires a tape that was written with the `Write Log/EOF` selection.

r) **`Retension Tape`**
A SCSI Retension command is issued to the tape drive. On some devices, this may require longer than one hour to complete. On drives that do not support (or need) retensioning, e.g., the EXABYTE streaming cartridge tape drive, no tape movement takes place and the test simply passes.

s) **`Erase Tape`**
Beginning at the start-of-media, all tape contents are erased. A write-enabled tape is required. On some devices, this may require over one hour to complete.

t) **`Write EOT`**
From the beginning of media, this test writes a file onto all the remaining media. The tape is then rewound. This requires a write-enabled tape. (EOT means End of Tape.)

v) **`Read EOT`**
From the beginning of media, this test reads all the remaining media. The data read from the tape is compared with the written data patterns. This requires a tape that was written with the `Write EOT` selection.

g) **`Confidence Test`**
Tests that the tape drive is ready and contains a tape.

i) **`Inquiry`**
Displays the drive type, vendor ID, product ID, product revision level, and vendor unique data field obtained by performing a SCSI Inquiry command.

MVME327 Local Floppy Tests Menu

To execute any local floppy test suite, select the appropriate floppy drive from the `327.n/flop.m` menu and the following menu displays:

```
SA: 327.n/flop.m <CR>    (to access the test selection menu for the first floppy drive)

Selections for Test "327 Floppy Disk"
(write-enabled scratch floppy required)

    0) Recalibrate          (reads sector zero on the disk)
    1) Confidence test      (reads last sector, then first sector)
    2) Read entire Disk     (reads every sector on the disk)
    3) Random read Disk     (randomly reads most sectors on the disk)
    4) Ping-Pong read Disk  (reads start, then end, then start + 1, then end - 1, etc.)
    r) Read a Sector        (allows reading of any sector; accepts hexadecimal
                             or decimal input)
    w) Write All Sectors    (write all sectors on floppy)
    f) Format Floppy        (format floppy)
```

where:

n is the board number.

m is the device number.

The following is a brief explanation of the tests:

0) Recalibrate
Causes the drive to read sector zero, reread it and compare the results. This has the affect of moving the head to cylinder zero.

1) Disk Confidence Test
Read the first then the last sector on the disk. Each read is done twice and the results compared.

2) Read entire disk
Reads the entire disk sequentially from sector zero to the last sector. Each read is done twice and the results compared.

3) `Random read disk`
Reads randomly selected blocks of sectors. Each read is done twice and the results compared.

4) `Ping-pong read disk`
Reads the entire disk in the following sequence: sector zero, the last sector, sector one, sector (last - 1), sector two, sector (last - 2), and so on, until every sector on the disk has been read twice, once from each direction. Every read is done twice and the results compared.

r) `Read a Sector`
Requests user input of a sector number to be read, verifies that the sector number is valid, and if so, reads the sector.

w) `Write All Sectors`
For every sector on the floppy disk, beginning at sector zero, this test writes the sectors. A write-enabled floppy disk is required. Note that this *destroys all* data on the floppy.

f) `Format floppy`
Formats the entire disk based on the variables found in the floppy disk configuration. The following section, *MVME327 Board Test Configuration*, shows the default 1.2Mb floppy disk configuration. A write-enabled floppy disk is required. Note that this *destroys all* data on the floppy.

MVME327 Board Test Configuration

The MVME327 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 327.0
/bdtest/327.0 configuration
  base[%0x100] = 0xffffa600 ?          # base address
  Scsi ID [0-7] = 7 ?                  # SCSI host id
  level[0-7] = 4 ?                     # default interrupt level
  am[0x0d,0x3d] = 0x0d ?               # address modifier
  data bus width[b,w,l] = l ?          # (byte, word, long)
  flop[0-2] = 2 ?                      # no of floppy disk
```

CAUTION

Setting the `Eval Flag` variable to `1-on` allows destructive disk testing!

> **cf 327.0/scsi0**

```
/bdtest/327.0/scsi0 configuration
  Ctrlr number[0] = 0 ?                        # scsi controller number
  phy-drive[0-3] = 0 ?                         # drive number
  Eval flag 0-NO write 1-WRITE TESTS[0-1] = 0 ?# destructive test flag
  Conf test flag Warnings 0-ON 1-OFF[0-1] = 0 ?# confidence test flag
                                               # for destructive disk tests
```

NOTE

The default setting is for high density double sided floppy.

> **cf 327.0/flop.0**

```
/bdtest/327.0/flop.0 configuration
  Ctrlr number[8] = 8 ?                        # controller number
  phy-drive[0-3] = 0 ?                         # floppy drive number
  Eval flag 0-NO write 1-WRITE TESTS[0-1] = 1 ?# destructive test flag
  Conf test flag Warnings 0-ON 1-OFF[0-1] = 0 ?# confidence test flag
                                               # for destructive disk tests
  test mode[r,w] = r ?                         # read only test or read/write test.
  sector/xfer[1-16] = 16 ?                     # size per xfer
  controller[0-0xff] = 0x27 ?                  # controller type
  peripheral[0-0xff] = 1 ?                     # peripheral type
  heads = 2 ?                                  # number of heads
  fixed/removable[0-1] = 1 ?
  cylinders/drive = 80 ?                       # number of cylinders per drive
  sector size[256,512,1024] = 512 ?            # bytes in a sector
  logical block size[256,512,1024] = 512 ? # bytes in a sector
  no of sector/track[15-16] = 15 ?             # number of sectors per track
  hard/soft sector flag = 0 ?
  interleave factor = 0 ?
  format char = 0x4e ?
  retry count = 10 ?                           # retry count
```

```
  step rate = 0x1e ?
  mot(0) or ibm (1) format = 1 ?
* track 0 density[0-1] = 1 ?
* track density[0-1] = 1 ?
* drive density[0-1] = 1 ?
* data rate[0-1] = 1 ?
  fm(0) or mfm(1)[0-1] = 1 ?
  precomp track = 0 ?              # precompensation
  low wrt current trk = 0 ?        # low write current
  alt type = 0 ?                   # the rest of these field are not used
  alt num = 0 ?
  ECC flag = 0 ?
  spiral offset = 0 ?
  gap1 = 0 ?
  gap2 = 0 ?
  gap3 = 0 ?
  gap4 = 0 ?
  strategy = 0 ?
  blocking = 0 ?
  skew = 0 ?
  cache = 0 ?

* 1=high and 0=low for the above selections.
```

6

If the `Eval flag` is set to 1 (=on), the MVME327 hard disk test menu is expanded to include destructive tests and menu items for closer disk control:

```
Selections for Test "327 SCSI Ctrlr 0 LUN 0 Tests"

 0) Recalibrate                 (reads sector zero on the disk)
 1) Confidence test             (reads last sector, then first sector)
 2) Read Entire Disk            (sequentially reads every sector on the disk)
 3) Random Read Disk            (randomly reads most sectors on the disk)
 4) Ping-Pong Read Disk         (reads start, then end, then start + 1, then end - 1, etc.)
 r) Read a Sector               (allows reading of any sector; accepts hexadecimal or
                                 decimal input)
 c) Read Capacity               (displays drive capacity in sectors)
 i) Inquiry                     (displays drive vendor and model)
 w) Write All Sectors (DESTRUCTIVE)
                                (sequentially write then read every sector on the disk)
 l) Write Last 1% of Sectors (DESTRUCTIVE)
                                (write then read last 1% sector on the disk)
 f) Format Disk (DESTRUCTIVE)
                                (format as perfect media - NOT COMPATIBLE WITH
                                 OPERATING SYSTEM)
 z) Set Confid Test Script      (modifies confid 5 script to enable DESTRUCTIVE
                                 testing)

SA:
```

6

Tests 0, 1, 2, 3, 4, r, b, c, and i are identical to those explained in the MVME327 SCSI Hard Disk Menu section. The following describes the destructive tests:

w) Write All Sectors
Causes the drive to sequentially write then read and verify every sector on the disk. *This destroys all data on the disk. Use extreme caution!*

l) Write Last 1% of Sectors
Causes the drive to write then read and verify approximately the last 1% of the sectors on the hard disk. Though this is a *quick* write-verification, it is still *extremely dangerous. Use extreme caution!*

f) `Format Disk`
Formats the entire hard disk as perfect media. In other words, bad sector mapping is handled automatically by the disk controller. *This destroys all data on the disk. Use extreme caution!*

> **CAUTION**
>
> **This format is not compatible with any operating system. It should be used only as a diagnostic procedure.**

6

z) `Set Confid Test Script`
Modifies `confid 5` (Continuous Running Intensive I/O Check) test script for this hard disk to include destructive disk tests. The destructive selection is TOGGLED each time this selection is executed. If destructive mode is enabled, the following message displays:

```
CAUTION: Destructive disk tests for drive n have been added
         to confidence and fault test scripts!!!
```

If destructive mode is disabled, the following message displays:

```
DRIVE n TEST SCRIPTS ARE NOW READ ONLY
```

Destructive test WARNING messages are disabled during confid 5 testing to allow unattended destructive tests.

MVME327 Error Codes

The MVME327 test module prints out error messages when unexpected test conditions are detected. If the MVME327 detects an error, the contents of the MVME327 command/status packet and the English translation of the error code both display. The following paragraphs describe the command/status packet and all error messages.

Command/Status Packet

The contents of this packet displays whenever an error or recovered error condition is encountered by the MVME327. The English translation of the error code also displays. The following is a typical example of the packet display:

```
                     packet address = 3f8820, ID = flop.0
|cmd|cmd_ctrl|pri_dev|pri_unit|sec_dev|sec_unit|am|dw|pri_addr|
|04 |   00   |  01   |   00   |  00   |   00   |0d|02|00000000|
|sec_addr| count  |sg_count|ctrl0|ctrl1|ctrl2|fatal_err|recover_err|
|003f7820|00000000|  0000  |0007 |0000 |3988 |   01    |     00    |
|additl_err|rcnt|res|err_sadr|term_cnt|stat0|stat1|stat2|
|   0000   | 00 |00 |00000000|00000024|0000 |0000 |0000 |

error: Bad descriptor
```

Each packet field is defined according to the *MVME327 Firmware User's Guide* and described below:

cmd
Command issued to MVME327

cmd_cntrl
Options for cmd

pri_dev
Device type (1=local floppy, 5=SCSI Bus, F=MVME327)

pri_unit
Unit Number (for SCSI upper nibble controller, lower nibble logical unit number; for local floppy, 0=floppy 0, 1=floppy 1)

sec_dev
Not used

sec_unit
Not used

am
Address modifier of memory space where the data buffer is located

dw
Data bus width (1=16 bit, 2=32 bit)

pri_addr
Primary address command dependent. Refer to the individual command description in the *MVME327 Firmware User's Guide* for usage.

sec_addr
Secondary address command dependent. Refer to the individual command description in the *MVME327 Firmware User's Guide* for usage.

count
Transfer count dependent. Refer to the individual command description in the *MVME327 Firmware User's Guide* for usage.

sg_count
Scatter/Gather count

ctrl0
Optional command parameters

ctrl1
Optional command parameters

ctrl2
Optional command parameters

fatal_err
Fatal error code (0 = no error, refer to Table 6-1)

recover_err
Recovered error status (0 = no error recovery attempted, refer to Table 6-1 for error that was recovered from)

additl_err
Additional error code/status (0 = no status, refer to Table 6-2)

rcnt
Retry count (if retry were required during this operation, this field indicates the number of retries attempted)

res
Not used

err_sadr
Error status address (either the sector address or logical block where the error occurred)

term_cnt
Termination transfer count (the number of bytes successfully transferred)

stat0
Not used

stat1
Not used

stat2
Status Parameter 3 (if fatal error code 1 or 2 is returned, then this status word contains the byte offset where the bad field is located). The *MVME327 Firmware User's Guide* refers to stat2 as Status Parameter 3.

Disk Descriptor Packet

This is for local floppy and SCSI hard disk drives:

```
                    Packet address = 3f8820, ID = flop.0
|ctlno|per_type|noheads|fix/remove| nocyls |bytes_sector| log_size|
| 00  |   01   |  02   |    01    |00000050|    0200    |   0200  |
|secs_per_track|res_secs_per_zone|hard/soft|interleave|format_char|
|      0f      |       00        |   00    |    00    |    4e     |
|retry_cnt|step_rate|floppy_format|pre_comp_cyl|reduced_wrt_cyl|
|   0a    |   01    |     00      |    0000    |     0000      |
|alt_type|noalts|ECC_flg|spiral_offset|cyl_skew|cache_entry_size|
|   00   |  00  |  00   |      00     |   00   |       01       |
```

Each packet field is defined according to the *MVME327 Firmware User's Guide* and described below:

ctlno
Controller type (a predefined SCSI controller type code; refer to the *MVME327 Firmware User's Guide* for values)

per_type
Peripheral type (acceptable codes: 1=floppy, 2=hard disk)

noheads
Number of heads on this disk

fix/remove
Fixed/removable media (0=fixed, 1=removable)

nocyls
Number of cylinders the disk contains

bytes_sector
Bytes per sector

log_size
Logical block size

secs_per_track
Logical sectors per track

res_secs_per_zone
Reserved sectors per zone

hard/soft
Hard/soft sector flag (0=soft, 1=hard)

interleave
Interleave factor

format_char
Format init character

retry_cnt
Retry count

step_rate
Step rate

floppy_format
Floppy format (only used by floppy)

pre_comp_cyl
Pre-compensation cylinder number

reduced_wrt_cyl
Reduced write current cylinder number

alt_type
Zone type

noalts
Number of alternates

ECC_flg
ECC correction (0=ecc off, 1=ecc on)

spiral_offset
Spiral offset

cyl_skew
Cylinder skew

cache_entry_size
Cache entry size

Tape Descriptor Packet

This is for the SCSI streaming tape:

```
                        Packet address = 3f8820, ID = scsi4
|ctlno|per_type|dri_trks|media_trks|ext_write|swab|buffered|res1|
| 00  |   00   |   00   |    00    |    00   | 00 |   00   | 00 |
|phys_bytes|log_size|qic_format|stream_cnt|res2|retry_cnt|
|   0000   |  0000  |   0000   |   0000   | 00 |    00   |
|read_size|write_size|
|  0000   |   0000   |
```

Each packet field is defined according to the *MVME327 Firmware User's Guide* and described below:

6

ctlno
Controller type (a predefined SCSI controller type code; refer to the *MVME327 Firmware User's Guide* for values)

per_type
Peripheral type (acceptable codes: 5=streaming tape)

dri_trks
Number of drive tracks

media_trks
Number of media tracks

ext_write
Extend on write flag (0=non-extend, 1=extend)

swab
Byte swap flag (0=no byte swap, 1=byte swap)

buffered
Buffered mode flag (0=non-buffered, 1=buffered)

res1
Not used

phys_bytes
Physical bytes per block

log_size
Logical block size

qic_format
QIC format

stream_cnt
Streaming count

res2
Not used

retry_cnt
Retry count

read_size
Minimum read transfer size

write_size
Minimum write transfer size

Start/Stop Tape Descriptor Packet

This is for the SCSI start/stop tape:

```
                    Packet address = 3f8820, ID = scsi4
|ctlno|per_type|speed|density|buffered|swab|resv0|retry_cnt|
| 00  |   00   | 00  |   00  |   00   | 00 | 00  |    00   |
|phys_bytes|log_size|
| 00000000 |00000000|
```

Each packet field is defined according to the *MVME327 Firmware User's Guide* and described below:

ctlno
Controller type (a predefined SCSI controller type code; refer to the *MVME327 Firmware User's Guide* for values)

per_type
Peripheral type (acceptable codes: 6=start/stop tape)

speed
Speed select code

density
Density select code

buffered
Flag 1. Buffered mode flag (0=non-buffered, 1=buffered)

swab
Flag 2. Byte swap flag (0=no byte swap, 1=byte swap)

res0
Not used

retry_cnt
Retry count

phys_bytes
Physical bytes per block

log_size
Logical block size

Additional Error Code/Status

The status word is returned in one of two formats:

Format 1
The upper byte is nonzero because a request sense (SCSI) command was issued at the request of the target. (A request sense (SCSI) command returns formatted error information about the last SCSI command to the target.) The upper byte contains the FM bit, EOM bit, Illegal Length Indicator (ILI) bit, and sense key (byte 02 of the sense data). The lower byte contains the additional sense code (byte 12 of the sense data), if available, from the sense data. The additional sense code byte is device dependent. Refer to Table 6-2 for a partial list of these error code definitions.

15	14	13	12	Bits 11-8	Bits 7-0
FM	EOM	ILI	Res	Sense Code	Additional Sense Code

Format 2
The upper byte is clear and the lower byte contains a specific MVME327A fatal error code that is relative to the current fatal error. Refer to Table 6-1 for a list of these error code definitions.

Bits 15-8	Bits 7-0
0	Error Code

Table 6-1. MVME327 Fatal Error Codes

Code (Hex)	Error Description	Notes
$00	Good	1
	$01-0F Command Parameter Errors	
$01	Bad descriptor	2,3
$02	Bad command	2,3
$03	Unimplemented command	
$04	Bad drive	3
$05	Bad logical address	3
$06	Bad scatter/gather table	
$07	Unimplemented device	
$08	Unit not initialized	3
	$10-1F Media Errors	
$10	No ID found on track	3
$11	Seek error	3
$12	Relocated track error	3
$13	Record not found, bad ID	3
$14	Data sync fault	3
$15	ECC error	3
$16	Record not found	3
$17	Media error	3
	$20-2F Drive Errors	
$20	Drive Fault	3
$21	Write protected media	3
$22	Motor not on	3
$23	Door open	3
$24	Drive not ready	3
$25	Drive busy	3
	$30-3F VME DMA Errors	
$30	VMEbus error	3,4
$31	Bad address alignment	3
$32	Bus time-out	3
$33	Invalid DMA transfer count	3

6

Table 6-1. MVME327 Fatal Error Codes (cont'd)

Code (Hex)	Error Description	Notes
	$40-4F Disk Format Errors	
$40	Not enough alternates	3
$41	Format failed	3
$42	Verify error	3
$43	Bad format parameters	3
$44	Cannot fix bad spot	3
$45	Too many defects	3
	$80-FF MVME327A Specific Errors	
$80	SCSI error, additional status available	3
$81	Indeterminate media error, no additional information	3
$82	Indeterminate hardware error	3
$83	Blank check (EOD or corrupted WORM)	3
$84	Incomplete extended message from TARGET	3,5
$85	Invalid reselection by an unthreaded TARGET	3,5
$86	No status returned from TARGET	3,5
$87	Message out not transferred to TARGET	3,5
$88	Message in not received from TARGET	3,5
$89	Incomplete data read to private buffer	3
$8A	Incomplete data write from private buffer	3
$8B	Incorrect CDB size was given	3,5
$8C	Undefined SCSI phase was requested	3,5
$8D	Time-out occurred during a select phase	3
$8E	Command terminated due to SCSI bus reset	3
$8F	Invalid message received	3,5
$90	Command not received	6
$91	Unexpected status phase	3
$92	SCSI script mismatch	3,5,9
$93	Unexpected disconnect caused command failure	3
$94	Request sense command was not successful	10
$95	No write descriptor for controller drive	7
$96	Incomplete data transfer	3
$97	Out of local resources for command processing	11

Table 6-1. MVME327 Fatal Error Codes (cont'd)

Code (Hex)	Error Description	Notes
$98	Local memory resources lost	
$99	Channel reserved for another VME host	12
$9A	Device reserved for another SCSI device	12
$9B	Already enabled, expecting TARGET response	6
$9C	Target not enabled	6
$9D	Unsupported controller type	7
$9E	Unsupported peripheral device type	7
$9F	Block size mismatch	8
$A0	Invalid cylinder number in format defect list	7
$A1	Invalid head number in format defect list	7
$A2	Block size mismatch —nonfatal	8
$A3	Our SCSI ID was not changed by command	13
$A4	Our SCSI ID has changed	6,14
$A5	No target enable has been completed	6
$A6	Cannot do longword transfers	7
$A7	Cannot do DMA transfers	7
$A8	Invalid logical block size	7,8
$A9	Sectors per track mismatch	7
$AA	Number of heads mismatch	7
$AB	Number of cylinders mismatch	7
$AC	Invalid floppy parameter(s)	
$AD	Already reserved	12
$AE	Was not reserved	12
$AF	Invalid sector number	7
$CC	Self Test failed	

6

Table 6-1. MVME327 Fatal Error Codes (cont'd)

NOTES:

1. The termination transfer count is always valid for a command that transfers data.
2. The bad byte is indicated by its offset value in status parameter 3. If the value is -1 ($FFFF), then the bad type is not indicated.
3. Additional status information may be available in the additional error code/status field of the BPP packet.
4. VMEbus error address contained in error status address field of BPP packet is currently not valid.
5. SCSI processing may not have finished. A SCSI bus reset command may need to be executed to put the SCSI bus in a known state.
6. TARGET mode only.
7. Designated parameter is in error.
8. Block size requested does not correspond to block size of device.
9. TARGET device did not behave as indicated by the script in the SCSI specific packet.
10. Error condition flagged by target device cannot be reported. Probably due to a hardware problem.
11. Command cannot be executed because local resources required exceed available local resources. Resubmit command when MVME327A is less busy.
12. Valid for the reserve/release commands.
13. Set SCSI address command was unsuccessful because at least one SCSI command was in progress.
14. Set SCSI address command was issued and all pending target wait commands were returned. Set SCSI address command may or may not be successful.

Table 6-2. Additional MVME327 Error Codes

Hex Value	Message
00	No additional error code
01	No index/sector signal
02	No seek complete
03	Write fault
04	Drive not ready
05	Drive not defined
06	Track ZERO not found
07	Multiple drives selected
08	Logical unit communications failure
09	Track following error
0A-0F	Reserved
10	ID CRC or ECC error
11	Unrecovered READ error
12	No address mark found for ID field
13	No address mark found for data area
14	No record found
15	Seek positioning error
16	Data synchronization mark error
17	Recovered data with target read retries
18	Recovered data with ECC correction
19	Defect list error
1A	Parameter overrun —parameter list too long
1B	Synchronous transfer error
1C	Primary defect list not found
1D	All bytes did not compare during a **VERIFY** command
1E	Recovered ID with ECC correction
1F	Reserved
20	Invalid command operation code
21	Illegal logical block address
22	Illegal function for device type
23	Reserved
24	Illegal field in CDB
25	Invalid LUN
26	Invalid field in parameter list
27	Disk is write protected
28	Medium change
29	Power on or bus device reset
2A	Mode select parameters have changed
2B-2F	Reserved

6

Table 6-2. Additional MVME327 Error Codes (cont'd)

Hex Value	Message
30	Incompatible cartridge
31	Medium format corrupted
32	No defect spare location available
33-3F	Reserved
40	RAM failure
41	Data path diagnostic failure
42	Power on diagnostic failure
43	Message reject error
44	Internal controller error
45	Select/reselect failed
46	Unsuccessful 'soft' reset
47	SCSI interface parity error
48	Initiator detected error
49	Inappropriate/illegal message
4A-4F	Reserved
50-5F	Reserved
60-6F	Reserved
70-7F	Reserved
80-FF	Vendor unique error codes

MVME328 SCSI Disk Drive Controller

The MVME328 board tests verify controller operation by performing the following types of tests:

VME328 Controller Tests

The MVME328 contains on-board firmware. The `328.n/controller` tests are used to cause the MVME328 to execute its self test capability or to display the revision level of the firmware currently installed.

VME328 Hard Disk Tests

The SCSI Hard Disk Tests verify SCSI disk controller operation by performing nondestructive, read-only tests on the SCSI hard disk drives. The SCSI disks to be tested must be initialized (formatted or have an operating system file system on them). Refer to the operating system documentation for information on formatting. Destructive testing and format capability may be enabled from `cmdline` mode by setting the `Eval flag` to `1` using `cf 328.n/scsim` where *n* is the board number and *m* is the device number.

VME328 Tape Tests

The SCSI Tape Tests verify SCSI tape controller operation by performing write and read tests on the SCSI tape drives. A write-enabled tape is required.

VME328 9 Track Tape Tests

The SCSI 9 Track Tape Tests verify SCSI tape controller operation by performing write and read tests on SCSI 9 track tape drives. A write-enabled tape is required.

MVME328 Floppy Disk Tests

The Floppy Disk Tests verify SCSI disk controller operation by performing nondestructive, read-only tests on the attached SCSI floppy disk drives. The floppy disks to be tested must be initialized (formatted or have an operating system file system on them). Refer to the operating system documentation for information on formatting using your operating system. Destructive testing and format capability may be enabled from `cmdline` mode by setting the `Eval flag` to `1` using `cf 328.n/scsim` where *n* is the board number and *m* is the device number.

MVME328 CD-ROM Disk Tests

The CD-ROM Disk Tests verify SCSI disk controller operation by performing nondestructive, read-only tests on the attached SCSI CD-ROM disk drives.

MVME328 Optical Memory Disk Tests

The Optical Memory Disk Tests verify SCSI disk controller operation by performing nondestructive, read-only tests on the attached SCSI Optical Memory disk drives. The Optical Memory disks to be tested must be initialized (formatted or have an operating system file system on them). Refer to the operating system documentation for information on formatting using your operating system. You may enable destructive testing and format capability from `cmdline` mode by setting the `Eval flag` to `1` using **cf 328.*n*/scsi*m***, where *n* is the board number and *m* is the device number.

MVME328 SCSI Controller Primary Menu

6

To access the test menus for any MVME328 board, type:

```
SA: 328.n
```

where *n* is the controller number, *0* for the first MVME328, *1* for the second board, and so on. A display similar to the following appears:

```
Current Menu is /bdtest/328.n - "VME328 SCSI Tests"
scsi13                 - 328 Secondary Channel, SCSI ID 6 Tests
scsi12                 - 328 Secondary Channel, SCSI ID 5 Tests
scsi11                 - 328 Secondary Channel, SCSI ID 4 Tests
scsi10                 - 328 Secondary Channel, SCSI ID 3 Tests
scsi9                  - 328 Secondary Channel, SCSI ID 2 Tests
scsi8                  - 328 Secondary Channel, SCSI ID 1 Tests
scsi7                  - 328 Secondary Channel, SCSI ID 0 Tests
scsi6                  - 328 Primary Channel, SCSI ID 6 Tests
scsi5                  - 328 Primary Channel, SCSI ID 5 Tests
scsi4                  - 328 Primary Channel, SCSI ID 4 Tests
scsi3                  - 328 Primary Channel, SCSI ID 3 Tests
scsi2                  - 328 Primary Channel, SCSI ID 2 Tests
scsi1                  - 328 Primary Channel, SCSI ID 1 Tests
scsi0                  - 328 Primary Channel, SCSI ID 0 Tests
controller             - 328 Controller Self Test
SA:
```

MVME328 SCSI Controller Self Test Menu

To execute the controller test suite, select `controller` from the `328.`*n* menu and the following menu displays:

```
SA: controller &lt;CR&gt;              (to access the controller menu)
Selections for Test "328 Controller Self Test"

    0) Self Test                  (To perform board self test)
    1) Display Controller Rev     (To display board revision information)
    R) Reset the Controller       (DEBUG only)

SA:
```

The following is a brief explanation of the tests:

0) Self Test
Causes the firmware residing on the MVME328 to execute its resident self test suite.

1) Display Controller Rev
Displays the board and firmware revision levels of this MVME328 board.

R) Reset the Controller
This is primarily for debug and should be used only if the MVME328 fails to respond to all input.

6

MVME328 SCSI Hard Disk Menu

To execute any hard disk test suite, select the appropriate SCSI controller from the `328.n` menu and the following menu displays:

```
SA: scsi0 <CR>                      (to access the first SCSI hard disk test)
Selections for Test "328 Primary Channel, SCSI ID 0 Tests"
  1) Hard Disk Confidence Test(Reads first sector, last sector, then first sector)
  2) Read Entire Hard Disk    (Sequentially reads every sector on the disk)
  3) Hard Disk Random Read    (Randomly reads 5% sectors on the disk)
  4) Hard Disk Ping-Pong Read(Reads start, end, then start + 1, end - 1, etc.)
  5) Device Self Tests        (Executes device self tests)
  r) Read a Sector            (Allows reading of any sector; accepts hexadecimal or
                              decimal input)
  b) Read a Block of Sectors (Allows reading of any block of sectors; accepts
                              hexadecimal or decimal input)
  c) Read Capacity            (Displays drive capacity in sectors)
  i) Inquiry                  (Displays drive vendor and model)
  s) Request Sense            (Reads and displays drive request sense information)
SA:
```

The following is a brief explanation of the tests:

1) **Hard Disk Confidence Test**
   Read the first sector, the last sector, then the first sector on the disk. Each read is done twice and the results compared.

2) **Read Entire Hard Disk**
   Reads the entire disk sequentially from sector zero to the last sector. Each read is done twice and the results compared.

3) **Hard Disk Random Read**
   Reads randomly selected blocks of sectors. Approximately 5% of the available sectors are read. Each read is done twice and the results compared.

4) Hard Disk Ping-Pong Read
Reads the entire disk in the following sequence: sector zero, the last sector, sector one, sector (last - 1), sector two, sector (last - 2), and so on, until every sector on the disk has been read twice, once from each *direction.* Every read is done twice and the results compared.

5) Device Self Tests
This test will issue a SCSI self test command. If the device fails its internal self tests, an error message is printed. Some devices do not support this command and will fail with an illegal command status.

r) Read a Sector
Requests user input of a sector number to be read, verifies that the sector number is valid, and if so, reads the sector, rereads the sector, and compares the results.

b) Read a Block of Sectors
Requests user input of a sector number at which to begin reading, verifies that these are valid sector numbers, then reads the block of sectors, rereads the block, and compares the results.

c) Read Capacity
Displays the drive capacity, in sectors. The display is in both hex and decimal.

NOTE

SCSI hard disk drives of the same type do not always have the same number of available sectors, but the numbers should be close in value (e.g. within 200 sectors of each other).

i) Inquiry
Displays the drive type, vendor ID, product ID, and product revision level obtained by performing a SCSI Inquiry command.

s) Request Sense
Displays the drive's request sense information. The information is obtained by performing a SCSI Request Sense command.

6

MVME328 SCSI Tape Tests Menu

To execute any SCSI tape test suite, select the appropriate SCSI controller from the `328.n` menu and the following menu displays:

```
SA: scsi4<CR>                        (to access the first SCSI tape test)
Selections for Test "328 Primary Channel, SCSI ID 4 Tests"
(A write-enabled scratch tape is required)
  1) Tape Confidence Test            (Verifies presence of media and does rewind)
  2) Write/Append/Read Tape Files    (Write/Append/Read/Compare test)
  3) Write/Read to EOT               (Write/Read/Compare test on entire tape)
  4) Read Entire Tape                (Reads, without verification, all data on tape)
  5) Device Self Tests               (Executes device self tests)
  r) Tape Retension                  (If supported by the drive, retensions media)
  e) Erase Tape                      (Removes ALL data from the tape then rewinds)
  b) Tape Block Data                 (Displays drive block size information)
  i) Inquiry                         (Displays drive vendor and model)
  s) Request Sense                   (Reads and displays drive request sense
                                     information)
  u) Unload and Eject the Media      (Unloads and Ejects the tape)
SA:
```

The following is a brief explanation of the tests:

1) **Tape Confidence Test**
   Tests that the tape drive is ready and contains a tape. If so, a rewind is done.

2) **Write/Append/Read Tape Files**
   At the beginning of media, two files are written. Each file consists of two blocks of data with one file mark. The tape is then rewound. Next, a space to End-Of-Data (EOD) is done. At EOD, a file with two blocks of data and a file mark is written or appended. The tape is then rewound. Now, each of the three files are read, the data blocks are verified, and the file mark is checked for. Next, EOD is checked for and the tape is rewound. If the test fails, an error message is printed.

3) **`Write/Read to EOT`**
From the beginning of media, this test writes a file onto all the remaining media. The tape is then rewound and the data read and compared with the written data patterns. This requires a write-enabled tape.

4) **`Read Entire Tape`**
From the beginning of media, this test reads every file on the tape until either blank media or end-of-media is encountered. No data verification is done. This merely confirms that all the media is readable.

5) **`Device Self Tests`**
This test will issue a SCSI self test command. If the device fails its internal self tests, an error message is printed. Some tape drives do not support this command and will fail with an illegal command status. For this reason, the confidence test scripts for tapes do *not* include this test.

6

r) **`Tape Retension`**
A SCSI Retension command is issued to the tape drive. On some devices, this requires over one hour to complete. On drives that do not support (or need) retensioning, such as the EXABYTE streaming cartridge tape drive, no tape movement takes place and the test simply passes.

e) **`Erase Tape`**
Beginning at the start-of-media, all tape contents are erased. A write-enabled tape is required. On some devices, this requires over one hour to complete.

b) **`Tape Block Data`**
Displays the tape drive's block size information obtained by performing a Read Block limits and a Mode Sense SCSI command.

i) **`Inquiry`**
Displays the drive type, vendor ID, product ID, and product revision level obtained by performing a SCSI Inquiry command.

s) **`Request Sense`**
Displays the drive's request sense information. The information is obtained by performing a SCSI Request Sense command.

u) **`Unload and Eject the Media`**
The tape is unloaded from the tape drive. If the drive supports tape ejection, then the tape is ejected from the drive. Tape unloading is accomplished by performing a SCSI Unload command.

MVME328 SCSI 9 Track Tape Tests Menu

To execute any SCSI 9 track tape test suite, select the appropriate SCSI controller from the `328.n/scsim` menu and the following menu displays:

```
SA: 328.n/scsim <CR> (to access the SCSI 9 track tape test)
Selections for Test "328 Primary Channel, SCSI ID m Tests"
(A write-enabled scratch tape is required)
   1) Tape Confidence Test         (Verifies presence of media and does rewind)
   2) Write/Append/Read Tape Files   (Write/Append/Read/Compare test)
   3) Write/Read to EOT            (Write/Read/Compare test on entire tape)
   4) Read Entire Tape             (Reads, without verification, all data on tape)
   5) Device Self Tests            (Executes device self tests)
   r) Tape Retension               (If supported by the drive, retensions media)
   e) Erase Tape                   (Removes ALL data from the tape then rewinds)
   b) Tape Block Data              (Displays drive block size information)
   i) Inquiry                      (Displays drive vendor and model)
   s) Request Sense                (Reads and displays drive request sense
                                   information)
   u) Unload and Eject the Media (Unloads and Ejects the tape)
SA:
```

Tests **1**, **3**, **4**, **5**, **r**, **e**, **b**, **i**, **s**, and **u** are identical to those explained in the *MVME328 SCSI Tape Drive Menu* section. Test **2** is a little different and the description follows:

2) `Write/Append/Read Tape Files`
At the beginning of media, two files are written. The first file consists of two blocks of data with one file mark. The last file consists of two blocks of data with two sequential file marks. The tape is then rewound. Next, the tape is spaced to the end of the two sequential file marks. Then the tape is backed up one file mark. The tape is now positioned at the End-Of-Data (EOD). At EOD, a file consisting of two blocks of data with two sequential file marks is written or appended. The tape is now rewound. Now, each of the three files are read, the data blocks are verified, and a file mark is checked for. Next, EOD is found by checking for the last file mark. The tape is then rewound. If the test fails, an error message is printed.

MVME328 SCSI Floppy Disk Tests Menu

SSID can read, write and format 5¼" and 3½" floppy disks in the following formats.

Table 6-3. Floppy Sizes and Formats Supported

Media Size	Description	Sectors Per Track	Cylinders	Capacity In Bytes
5¼"	Double sided, double density (PC-XT)	8	40	332 K
5¼"	Double sided, double density (PC-XT)	9	40	360 K
5¼"	Double sided, high density (PC-AT)	9	80	1.2 M
5¼"	Double sided, double density (Motorola dsdd5)	16	80	655 K
3½"	Double sided, double density (PC-AT)	9	80	737 K
3½"	Double sided, high density (PC-AT)	15	80	1.4 M
3½"	Double sided, extra density	36	80	2.9 M

In all read and write tests, a formatted floppy is required. SSID automatically detects the format of the disk. If the disk is unreadable in any of the appropriate formats listed in the above table, the test fails. During an SSID MVME328 floppy disk format, you are asked which format to use. Be sure that the format you request matches the media density that you are formatting.

To execute any SCSI floppy disk test suite, select the appropriate SCSI controller from the **328.*n*/scsi*m*** menu and the following menu displays:

```
SA: scsi6 <CR>                        (to access the first SCSI floppy disk test)

Selections for Test "328 Primary Channel, SCSI ID 6 Tests"

  1) Floppy Disk Confidence Test  (Reads first sector, last sector, then first sector)
  2) Read Entire Floppy Disk      (Sequentially reads every sector on the disk)
  3) Floppy Disk Random Read      (Randomly reads 5% sectors on the disk)
  4) Floppy Disk Ping-Pong Read   (Reads start, end, then start + 1, end - 1, etc.)
  5) Device Self Tests            (Executes device self tests)
  i) Inquiry                      (Displays drive vendor and model)
  s) Request Sense                (Reads/displays drive request sense information)
  c) Read Capacity                (Determines capacity of disk)
  r) Read a Sector                (Allows reading of any sector; accepts
                                  hexadecimal or decimal input)
  b) Read a Block of Sectors      (Allows reading of any block of sectors;
                                  accepts hexadecimal or decimal input)

SA:
```

Tests **1**, **2**, **3**, **4**, **5**, **r**, **b**, **i**, and **s** are identical to those explained in the *MVME328 SCSI Hard Disk Menu* section. The following is a brief explanation of test **c**:

c) Read Capacity
Determines capacity of the floppy disk currently in the drive and displays it in decimal and hexadecimal. Requires a formatted floppy disk.

6

MVME328 SCSI CD-ROM Menu

To execute any CD-ROM test suite, use the SSID **slctdev** command to select the CD-ROM for the MVME328 SCSI ID address. Then select the appropriate SCSI controller from the `328.n/scsim` menu and the following menu displays:

```
SA: 328.n/scsim <CR>  (to access the SCSI CD-ROM test)
Selections for Test "328 Primary Channel, SCSI ID m Tests"
   1) CD-ROM Confidence Test      (Reads first sector, last sector, then first sector)
   2) Read Entire CD-ROM          (Sequentially reads every sector on the disk)
   3) CD-ROM Random Read          (Randomly reads 5% sectors on the disk)
   4) CD-ROM Ping-Pong Read       (Reads start, end, then start + 1, end - 1, etc.)
   5) Device Self Tests           (Executes device self tests)
   r) Read a Sector               (Allows reading of any sector; accepts
                                  hexadecimal or decimal input)
   b) Read a Block of Sectors     (Allows reading of any block of sectors; accepts
                                  hexadecimal or decimal input)
   c) Read Capacity               (Displays drive capacity in sectors)
   i) Inquiry                     (Displays drive vendor and model)
   s) Request Sense               (Reads and displays drive request sense
                                  information)
   e) Unload and Eject the Media  (Ejects the CD-ROM)
   a) Audio Status                (Displays audio playback status)
   d) Disk Information            (Displays CD-ROM disk information)
   p) Play Audio CD-ROM           (Plays an Audio CD-ROM)
   f) Audio Pause/Resume          (Pauses/Resumes Audio playback)
SA:
```

Tests **1, 2, 3, 4, 5, r, b, c, i,** and **s** are identical to those explained in the *MVME328 SCSI Hard Disk Menu* section. The following is a brief explanation of tests **e, a, d, p,** and **f**:

`e) Unload and Eject the Media`
The CD-ROM is unloaded and ejected from the CD-ROM drive. CD-ROM unloading is accomplished by performing a Toshiba SCSI Unload and Eject command.

a) `Audio Status`
The current audio playback status is requested from the drive and displayed. The playback status is read by performing a Toshiba SCSI Read Audio Status command.

d) `Disk Information`
The current CD-ROM disk information is requested from the drive and displayed. The disk information is read by performing a Toshiba SCSI Read Disk Information command.

p) `Play Audio CD-ROM`
The number of tracks on the current CD-ROM is requested from the drive. The user is prompted for the starting and ending tracks to play. The starting and ending tracks are sent to the drive and it is told to play these tracks.

p) `Audio Pause/Resume`
If the CD-ROM drive is currently playing an audio track, then a Toshiba SCSI Audio Still command is sent to the drive. If the CD-ROM drive has been Stilled by a Audio Still command, then a Toshiba SCSI Audio Resume command is sent to the drive. If the CD-ROM drive is not in one of the preceding states, this command will just exit.

MVME328 SCSI Optical Memory Menu

To execute any Optical Memory test suite, use the SSID **slctdev** command to select the Optical Memory for the MVME328 SCSI ID address. Then select the appropriate SCSI controller from the `328.n/scsim` menu and the following menu displays:

```
SA: 328.n/scsim <CR> (to access the SCSI Optical Memory test)
Selections for Test "328 Primary Channel, SCSI ID m Tests"
  1) Optical Memory Confidence Test (Reads first sector, last sector, then first
                                     sector)
  2) Read Entire Optical Memory (Sequentially reads every sector on the disk)
  3) Optical Memory Random Read (Randomly reads 5% sectors on the disk)
  4) Optical Memory Ping-Pong Read (Reads start, end, then start + 1, end - 1,
                                     etc.)
  5) Device Self Tests              (Executes device self tests)
  r) Read a Sector                  (Allows reading of any sector; accepts hexadecimal
                                     or decimal input)
  b) Read a Block of Sectors        (Allows reading of any block of sectors; accepts
                                     hexadecimal or decimal input)
  c) Read Capacity                  (Displays drive capacity in sectors)
  i) Inquiry                        (Displays drive vendor and model)
  s) Request Sense                  (Reads and displays drive request sense
                                     information)
  e) Unload and Eject the Media (Ejects the Optical media)
SA:
```

Tests **1, 2, 3, 4, 5, r, b, c, i,** and **s** are identical to those explained in the *MVME328 SCSI Hard Disk Menu* section. The following is a brief explanation of test **e**:

`e) Unload and Eject the Media`

The Optical Media is unloaded and ejected from the Optical drive. Optical Media unloading is accomplished by performing a SCSI Stop Unit command.

6

6

MVME328 Board Test Configuration

The MVME328 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 328.n                                        (to configure parameters
                                                   for the nth 328 board)
/bdtest/328.n configuration
  base [%0x100] = 0xfff9000 ?                     (base address)
  level [0-7] = 4 ?                               (default interrupt level)
  mode [0] = 0 ?                                  (destructive mode flag)
  warning msgs [0] = 0 ?                          (Not modifiable)
  am[0x0d,0x3d] = 0x0d ?                          (address modifier)
  data bus width[w,l] = l ?                       (bus width set to long)
  DMA burst byte count [0-512] = 0 ?              (0 sets to default maximum)
  SCSI timeout [0-10000] = 500 ?                  (msecs to wait for a device select)
  Work queue 0 timeout [0-100] = 50 ?             (Do NOT modify)
  VMEbus timeout [0-100] = 0 ?                    (Do NOT modify)
  Device timeout-secs [1-500] = 70 ?              (Set for time device takes to become ready)
```

To set the parameters for each individual drive, use `cf 328.n/scsim`:

```
> cf 328.n/scsi0                                  (to configure first SCSI device)
/bdtest/328.n/scsi0 configuration
  Ctrlr number[0] = 0 ?                           (scsi controller number)
  phy-drive[0-3] = 0 ?                            (drive number)
  Maximum sector size [1-10000] = 512?            (Used to set block size for tapes and
                                                   hard disk formatting)
  sector/xfer [1-128] = 64 ?                      (Number of blocks to transfer)
  Device type flag [0] = 0 ?                      (Program device type flag)
  Eval flag:Write/Expert Mode(0-Off 1-On) [0-1] = 0 ?
                                                  (destructive test flag)
  Conf flag:Warning messages(0-On 1-Off)[0-1] = 0 ?
                                                  (To turn user warning about
                                                   destructive disk tests on and off)
  Write Type(0-1%,1-All,2-Random,3-Ping)[0-3] = 0 ?
                                                  (Selects the write test that will be
                                                   available on the expert menu)
```

CAUTION

Setting the `Eval flag` variable to `1-On` changes the device test menu so that it will include destructive tests and expert commands.

MVME328 Hard Disk Expert Menu

If the `Eval flag` is set to `1-On`, the MVME328 hard disk test menu is expanded to include destructive tests and menu items for closer disk control.

```
SA: 328.n/scsim <CR>   (to access the expert hard disk test)
Selections for Test "328 Primary Channel, SCSI ID m Tests"

  1) Hard Disk Confidence Test          (Reads first sector, last sector, then first sector)
  2) Read Entire Hard Disk              (Sequentially reads every sector on the disk)
  3) Hard Disk Random Read              (Randomly reads 5% sectors on the disk)
  4) Hard Disk Ping-Pong Read           (reads start, then end, then start + 1, then end - 1, etc.)
  5) Device Self Tests                  (Executes device self tests)
  r) Read a Sector                      (Allows reading of any sector; accepts hexadecimal or
                                         decimal input)
  b) Read a Block of Sectors            (allows reading of any block of sector; accepts hexadecimal
                                         or decimal input)
  c) Read Capacity                      (Displays drive capacity in sectors)
  i) Inquiry                            (displays drive vendor and model)
  s) Request Sense                      (Reads and displays drive request sense information)
  t) Test Unit Ready                    (Tests the drive's current state of readiness)
  m) Display Mode Sense                 (Displays the drive's Mode Sense data)
  d) Display Defect Lists               (Displays manufacturer's and grown defect lists)
  v) Mode Sense/Select                  (Does SCSI Mode Sense, Mode Select test)
  w) User Selected Write Test (DESTRUCTIVE) (Write test as set by the device test configuration)
  f) Format Disk (DESTRUCTIVE)          (Format media - NOT COMPATIBLE
                                         WITH OPERATING SYSTEM)
  x) Re-Assign a Block (DESTRUCTIVE)
                                        (Reassigns defective logical block and adds it to the
                                         grown defect list)
  z) Set Confid Test Script             (Modifies confid 5 script to enable DESTRUCTIVE
                                         testing)

SA:
```

Tests **1, 2, 3, 4, 5, r, b, c, i,** and **s** are identical to those explained in the *MVME328 SCSI Hard Disk Menu* section. The following describes the expert and destructive tests:

t) Test Unit Ready

A SCSI Test Unit Ready command is performed on the drive. If the drive is currently ready, this test will pass. If the drive is **NOT** ready, this test will fail, displaying information about the failure. *NOTE: This test may fail one time and pass every time thereafter. This is the nature of a SCSI Test Unit Ready command. This command only indicates a drive's ready state.*

m) Display Mode Sense

Displays the Mode Sense information that is read from the drive. The user is prompted for the type of Mode Sense information that is to be displayed.

There are four types of mode sense information; they are Current values, Changeable values, Default values, and Saved values. Note: Some drives do not support all four types of mode sense data and will cause this test to fail.

`d) Display Defect Lists`

This command displays the manufacturer's and the grown media defect lists contained on the hard disk. The defect lists are requested in one of three types of formats. The user is prompted to select the format type that will be used to request the defect lists. The three types of formats are as follows:

1. **Block Format** - This format as defined by **SCSI-2** is a four-byte defective block address that contains the defect. This format is *not* supported by all hard disk drives and drives that do not support this format, the display defect list command will automatically change to the **Bytes From Index** format.
2. **Physical Sector Format** - This format as defined by **SCSI-2** is an eight-byte sector defect location. Each defect location is comprised of a three-byte cylinder number of the defect, a one-byte head number of the defect, and a four-byte defect sector number.
3. **Bytes From Index Format** - This format as defined by **SCSI-2** is an eight-byte defect location. Each defect location is comprised of a three-byte cylinder number of the defect, a one-byte head number of the defect, and a four-byte defect bytes from index.

The manufacturer's defect list, which is called the **Primary Defects List**, is requested and displayed first. Next, the grown defect list, which is called the **Grown Defects List**, is requested and displayed. The user is allowed to exit this command by entering an **E** or **e** during format type selection or by entering a **Q** or **q** after the **Depress RETURN or ENTER key to continue** message.

`v) Mode Sense/Select`

A SCSI Mode Sense command is performed on the drive. The received Mode Sense information is then used to perform a Mode Select command. Next, another SCSI Mode Sense command is performed on the drive. The Mode Sense information received on the first SCSI Mode Sense command is compared to the Mode Sense information received from the second Mode Sense command. The test will fail if there were changes to the second Mode Sense information received.

6

`w)` *User Selected Write Test*

This test and the test name will be set to the type of test as selected in the device test configuration variable **Write Type** (see the board test configuration section for this variable). The **Write Type** variable can be set to one of four values. The **Write Type** value will determine the name of this test and the type of test this command will run. The test name and the test function is defined as follows for the **Write Type** values:

0 - **Write Last 1%** **(DESTRUCTIVE)**
This test will write then read and verify approximately the last 1% of the sectors on the hard disk. Though this is a *quick* write-verification, it is still *extremely dangerous. Use extreme caution!*

1 - **Write All Sectors** **(DESTRUCTIVE)**
This test will sequentially write then read and verify every sector on the disk. *This test destroys all data on the disk. Use extreme caution!*

2 - **Random 50% Write** **(DESTRUCTIVE)**
This test will randomly select 50% of the sectors to write then read and verify. *This test randomly destroys data on the disk. Use extreme caution!*

3 - **Ping-Pong Write** **(DESTRUCTIVE)**
This test will write, read and verify the entire disk in the following sequence: sector zero, the last sector, sector one, sector (last - 1), sector two, sector (last - 2), and so on, until every sector on the disk has been written and read, once from each *direction. This test destroys all data on the disk. Use extreme caution!*

`f) Format Disk (DESTRUCTIVE)`

This test will format the entire hard disk. Before the format begins, the user is asked if they would like to retain the grown list of defects. If the user selects to **not** retain the grown list of defects, the sectors listed in the grown defects list will be returned to the usable sector list, the grown defects list will be zeroed, and the hard disk will be formated as perfect media. A perfect media format is where bad sector mapping is automatically handled by the disk controller. If the user selects to retain the grown list of defects, the disk controller will be instructed to keep the grown list of defects. The controller will automatically redirect the sectors list in the grown list of defects. *This test destroys all data on the disk. Use extreme caution!*

CAUTION

This format is not compatible with any operating system. It should be used only as a diagnostic procedure.

x) `Re-Assign a Block` (DESTRUCTIVE)
Requests that the TARGET hard disk drive reassign the user-entered defective logical block to an area on the logical unit reserved for this purpose and to record the defective logical block in the grown defect list if such a list is supported. *This can destroy data on the disk. No attempt is made to recover data from the reassigned block. Use extreme caution!*

6

z) `Set Confid Test Script`
Modifies **`confid 5`** (Continuous Running Intensive I/O Check) test script for this hard disk to include destructive disk tests. The destructive selection is TOGGLED each time this selection is executed. If destructive mode is enabled, the following message is displayed:

CAUTION: Destructive disk tests for drive *n* have been added to confidence and fault test scripts!!!

If destructive mode is disabled, the following message is displayed:

DRIVE *n* TEST SCRIPTS ARE NOW READ ONLY

Destructive test WARNING messages are disabled during confid 5 testing to allow unattended destructive tests.

MVME328 Tape Drive Expert Menu

If the `Eval flag` is set to `1-On`, the MVME328 tape drive test menu is expanded to include menu items for closer tape control.

```
SA: 328.n/scsim <CR> (to access the expert tape drive test)
Selections for Test "328 Primary Channel, SCSI ID m Tests"
(A write-enabled scratch tape is required)

  1) Tape Confidence Test          (Verifies presence of media and does rewind)
  2) Write/Append/Read Tape Files  (Write/Append/Read/Compare test)
  3) Write/Read to EOT             (Write/Read/Compare test on entire tape)
  4) Read Entire Tape              (Reads, without verification, all data on tape)
  5) Device Self Tests             (Executes device self tests)
  r) Tape Retension                (If supported by the drive, retensions media)
  e) Erase Tape                    (Removes ALL data from the tape then rewinds)
  b) Tape Block Data               (Displays drive block size data)
  i) Inquiry                       (Displays drive vendor and model)
  m) Display Mode Sense            (Displays the drive's Mode Sense data)
  s) Request Sense                 (Reads and displays drive request sense information)
  t) Test Unit Ready               (Tests the drive's current state of readiness)
  u) Unload and Eject the Media    (Unloads and Ejects the tape)

SA:
```

The added tests are **m** and **t**. These tests are identical to the tests that are explained in the *MVME328 SCSI Expert Hard Disk Menu* section.

MVME328 9 Track Tape Drive Expert Menu

If the `Eval flag` is set to `1-On`, the MVME328 9 track tape drive test menu is expanded to include menu items for closer tape control.

```
SA: 328.n/scsim <CR> (to access the expert 9 track tape drive test)
Selections for Test "328 Primary Channel, SCSI ID m Tests"
(A write-enabled scratch tape is required)

  1) Tape Confidence Test           (Verifies presence of media and does rewind)
  2) Write/Append/Read Tape Files   (Write/Append/Read/Compare test)
  3) Write/Read to EOT              (Write/Read/Compare test on entire tape)
  4) Read Entire Tape               (Reads, without verification, all data on tape)
  5) Device Self Tests              (Executes device self tests)
  r) Tape Retension                 (If supported by the drive, retensions media)
  e) Erase Tape                     (Removes ALL data from the tape then rewinds)
  b) Tape Block Data                (Displays drive block size data)
  i) Inquiry                        (Displays drive vendor and model)
  m) Display Mode Sense             (Displays the drive's Mode Sense data)
  s) Request Sense                  (Reads and displays drive request sense information)
  t) Test Unit Ready                (Tests the drive's current state of readiness)
  u) Unload and Eject the Media     (Unloads and Ejects the tape)
SA:
```

The added tests are **m** and **t**. These tests are identical to the tests that are explained in the *MVME328 SCSI Expert Hard Disk Menu* section.

MVME328 Floppy Disk Expert Menu

If the `Eval flag` is set to `1-On`, the MVME328 floppy drive test menu is expanded to include destructive tests and menu items for closer disk control.

```
SA: 328.n/scsim <CR> (to access the expert floppy disk test)
Selections for Test "328 Primary Channel, SCSI ID m Tests"

  1) Floppy Disk Confidence Test    (Reads first sector, last sector, then first sector)
  2) Read Entire Floppy Disk        (Sequentially reads every sector on the disk)
  3) Floppy Disk Random Read        (Randomly reads 5% sectors on the disk)
  4) Floppy Disk Ping-Pong Read     (Reads start, then end, then start + 1, then end - 1, etc.)
  5) Device Self Tests              (Executes device self tests)
  i) Inquiry                        (Displays drive vendor and model)
  s) Request Sense                  (Reads and displays drive request sense information)
  c) Read Capacity                  (Determines capacity of disk)
  t) Test Unit Ready                (Tests the drive's current state of readiness)
  m) Display Mode Sense             (Displays the drive's Mode Sense data)
  r) Read a Sector                  (Allows reading of any sector; accepts hexadecimal or
                                    decimal input)
  b) Read a Block of Sectors        (Allows reading of any block of sectors; accepts
                                    hexadecimal or decimal input)
  w) Write All Sectors    (DESTRUCTIVE)
                                    (Sequentially write then read every sector on the disk)
  f) Format Floppy        (DESTRUCTIVE)
                                    (Format media - NOT COMPATIBLE WITH
                                    OPERATING SYSTEM)

SA:
```

The added tests are **t**, **m**, **w**, and **f**. Tests **t**, **m**, and **w** are identical to the tests that are explained in the *MVME328 SCSI Expert Hard Disk Menu* section. Test **f** is explained as follows:

`f) Format Floppy     (DESTRUCTIVE)`
This test offers the user a menu of supported floppy formats and then formats the floppy as requested. A write-enabled floppy disk is required. This *destroys all* data on the floppy.

If a 5¼" floppy drive is attached to the MVME328, the following menu of available formats is displayed:

```
SA: f                                   (to format the floppy diskette)
 Please select format to use:

   0: 5 1/4" PC-XT 9 secs/trk - 360 kbytes
   1: 5 1/4" PC-XT 8 secs/trk - 332 kbytes
   2: 5 1/4" dsdd5 - Motorola format - 655 kbytes
   3: 5 1/4" PC-AT - 1.2 Mbytes
   q: quit

 Select 0-3, or q to quit:
```

If a 3½" floppy drive is attached to the MVME328, the following menu of available formats is displayed:

```
SA: f                                   (to format the floppy diskette)
 Please select format to use:

   0: 3 1/2" Low density - 720 kbytes
   1: 3 1/2" High density - 1.4 Mbytes
   2: 3 1/2" Extra density - 2.9 Mbytes
   q: quit

 Select 0-2, or q to quit:
```

6

MVME328 CD-ROM Drive Expert Menu

If the `Eval flag` is set to `1-On`, the MVME328 CD-ROM drive test menu is expanded to include menu items for closer disk control.

```
SA: 328.n/scsim <CR>  (to access the expert CD-ROM test)
Selections for Test "328 Primary Channel, SCSI ID m Tests"
  1) CD-ROM Confidence Test  (Reads first sector, last sector, then first sector)
  2) Read Entire CD-ROM      (Sequentially reads every sector on the disk)
  3) CD-ROM Random Read      (Randomly reads 5% sectors on the disk)
  4) CD-ROM Ping-Pong Read   (Reads start, end, then start + 1, end - 1, etc.)
  5) Device Self Tests       (Executes device self tests)
  r) Read a Sector           (Allows reading of any sector; accepts hexadecimal or
                             decimal input)
  b) Read a Block of Sectors (Allows reading of any block of sectors; accepts
                             hexadecimal or decimal input)
  c) Read Capacity           (Displays drive capacity in sectors)
  i) Inquiry                 (Displays drive vendor and model)
  m) Display Mode Sense      (Displays the drive's Mode Sense data)
  s) Request Sense           (Reads and displays drive request sense information)
  t) Test Unit Ready         (Tests the drive's current state of readiness)
  e) Unload and Eject the Media  (Ejects the CD-ROM)
  a) Audio Status            (Displays audio playback status)
  d) Disk Information        (Displays CD-ROM disk information)
  p) Play Audio CD-ROM       (Plays an Audio CD-ROM)
  f) Audio Pause/Resume      (Pauses/Resumes Audio playback)
  v) Mode Sense/Select       (Does SCSI Mode Sense, Mode Select test)
SA:
```

The added tests are **m**, **t**, and **v**. These tests are identical to the same tests that are explained in the *MVME328 SCSI Expert Hard Disk Menu* section.

MVME328 Optical Memory Drive Expert Menu

If the `Eval flag` is set to `1-On`, the MVME328 Optical Memory drive test menu is expanded to include menu items for closer disk control.

```
SA: 328.n/scslm <CR> (to access the expert Optical Memory test)
Selections for Test "328 Primary Channel, SCSI ID m Tests"
  1) Optical Memory Confidence Test(Reads first sector, last sector, then first sector)
  2) Read Entire Optical Memory      (Sequentially reads every sector on the disk)
  3) Optical Memory Random Read      (Randomly reads 5% sectors on the disk)
  4) Optical Memory Ping-Pong Read (Reads start, end, then start + 1, end - 1, etc.)
  5) Device Self Tests               (Executes device self tests)
  r) Read a Sector                   (Allows reading of any sector; accepts hexadecimal or
                                     decimal input)
  b) Read a Block of Sectors         (Allows reading of any block of sectors; accepts
                                     hexadecimal or decimal input)
  c) Read Capacity                   (Displays drive capacity in sectors)
  i) Inquiry                         (Displays drive vendor and model)
  d) Display Defect Lists            (Displays manufacturers and grown defect lists)
  m) Display Mode Sense              (Displays the drive's Mode Sense data)
  s) Request Sense                   (Reads and displays drive request sense information)
  t) Test Unit Ready                 (Tests the drive's current state of readiness)
  w) User Selected Write Test (DESTRUCTIVE) (Write test as set by the device test configuration)
  f) Format Disk          (DESTRUCTIVE)
                                     (Format media - NOT COMPATIBLE
                                     WITH OPERATING SYSTEM)
  x) Re-Assign a Block   (DESTRUCTIVE)
                                     (Reassigns defective logical block and adds it to the
                                     grown defect list)
  e) Unload and Eject the Media      (Ejects the Optical media)
  z) Set Confid Test Script          (Modifies confid 5 script to enable
                                      DESTRUCTIVE testing)
SA:
```

The added tests are **d, m, t, w, f, x,** and **z**. These tests are identical to the same tests that are explained in the *MVME328 SCSI Expert Hard Disk Menu* section.

6

MVME328 Error Codes

The MVME328 test module prints out error messages when unexpected test conditions are detected. If the MVME328 detects an error, the contents of the MVME328 SCSI sense packet and the English translation of the error code both display. The following paragraphs describe the SCSI sense packet and error messages:

SCSI Sense Packet

The contents of this packet displays whenever an error or recovered error condition is encountered by the MVME328. The English translation of the error code also displays. The following is a typical example of the SCSI Sense display:


```
/bdtest/328.n/scsim First pass read error                          LINE 1

/bdtest/328.n/scsim Fatal Read Error                               LINE 2

              At block 221219 (0x36023),                           LINE 3
              0x03, Non-recoverable media error                    LINE 4
              Sense code: 0x11, Sense qual: 0x0                    LINE 5

/bdtest/328.n/scsim
       Read sectors 221219-221283 (0x36023-0x36063):failed         LINE 6
```

where:

LINE 1
: indicates the SCSI device that failed and whether the failure occurred on the first read, second read, write, or read following write.

LINE 2
: indicates the SCSI device that failed and whether the failure is fatal or recoverable.

LINE 3
: contains the logical block number of the error.

LINE 4
contains the SCSI Sense Key and English description of the error that occurred.

LINE 5
contains the additional sense code and sense qualifier returned for the error.

LINE 6
indicates the SCSI device that failed, the subtest that failed and the range of sectors being accessed, in both decimal and hexadecimal.

Table 6-4 gives a further explanation of the SCSI Sense Key found on Line 4. To interpret the meaning of the additional sense code and sense qualifier found on line 5, refer to an SCSI specification or to the product specification for the drive that failed.

Table 6-4. MVME328 SCSI Sense Keys

Sense Key (Hex)	Error Description
$00	**NO SENSE.** Indicates that there is no specific sense key information to be reported for the designated logical unit. This would be the case for a successful command or a command that received CHECK CONDITION or COMMAND TERMINATED status because one of the filemark, END-OF-MEDIA, or ILLEGAL-LENGTH-INDICATOR bits is set to 1.
$01	**RECOVERED ERROR.** Indicates that the last command completed successfully with some recovery actions performed by the target. Details may be determinable by examining the additional sense bytes and the information field. When multiple recovered errors occur during one command, the choice of which error to report (i.e., first, last, most severe) is device specific.
$02	**NOT READY.** Indicates that the logical unit addressed cannot be accessed. Operator intervention may be required to correct this condition.
$03	**MEDIA ERROR.** Indicates that the command terminated with a non-recovered error condition that was probably caused by a flaw in the medium or an error in the recorded data. This sense key may also be returned if the target is unable to distinguish between a flaw in the medium and a specific hardware failure (sense key $04).
$04	**HARDWARE ERROR.** Indicates that the target detected a non-recoverable hardware failure (e.g., controller failure, device failure, parity error) while performing the command or during a self test.

Table 6-4. MVME328 SCSI Sense Keys (cont'd)

Sense Key (Hex)	Error Description
$05	**ILLEGAL REQUEST.** Indicates that there was an illegal parameter in the command descriptor block or in the additional parameters supplied as data for some commands. This may be indicative of memory failure in the CPU or on the device controller board.
$06	**UNIT ATTENTION.** Indicates that the removable medium may have been changed or the target has been reset.
$07	**DATA PROTECT.** Indicates that a command that reads or writes the medium was attempted on a block that is protected from this operation. The read or write operation is not performed.
$08	**BLANK CHECK.** Indicates that a sequential-access or write-once device encountered blank medium or format-defined end-of-data indication while reading or a write-once device encountered a non-blank medium while writing.
$09	**VENDOR SPECIFIC.** This sense key is available for reporting vendor specific conditions. Refer the device manual for cause of the condition.
$0A	**COPY ABORTED.** Indicates that a COPY, COMPARE, or COPY AND VERIFY command was aborted due to an error condition on the source device, the destination device, or both.
$0B	**ABORTED COMMAND.** Indicates that the target aborted the command.
$0C	**EQUAL.** Indicates that a SEARCH DATA command has satisfied an equal comparison.
$0D	**VOLUME OVERFLOW.** Indicates that a buffered peripheral device has reached the end-of-partition and data may remain in the buffer that has not been written to the medium.
$0E	**MISCOMPARE.** Indicates that the source data did not match the data read from the medium.
$0F	**RESERVED.**

Some MVME328 errors are caused not by SCSI failures, but rather by SCSI bus errors, VMEbus errors, communication failures between the CPU and the MVME328, or by MVME328 or CPU failures. Tables 6-5 through 6-9 give a further explanation of these errors.

Table 6-5. MVME328 Command Queue/Controller Errors

Error Code (Hex)	Error Description
$00	**GOOD STATUS.** The controller has completed the command and no errors were detected.
$01	**QUEUE FULL.** The work queue specified for this command is full and cannot receive another entry. The command is not executed.
$02	**WORK QUEUE INITIALIZATION ERROR.** The work queue specified has not been initialized. The command is not executed.
$03	**FIRST COMMAND ERROR.** The first command sent to the board was not an Initialize Controller Command. This would indicate a communication error between the CPU and the MVME328.
$04	**COMMAND CODE ERROR.** The command field contains an invalid command type.
$05	**QUEUE NUMBER ERROR.** The work queue number specified in the Command Queue entry is invalid.
$06	**QUEUE ALREADY INITIALIZED.** The work queue number specified to be initialized has already been initialized.
$07	**QUEUE UNINITIALIZED.** An input/output parameter block (IOPB) was issued to a work queue that has not been initialized.
$08	**QUEUE MODE NOT READY.** The Start Queue Mode bit was set before the Initialize Controller Command was issued.
$09	**COMMAND UNAVAILABLE.** The command specified has not been implemented in the current firmware.
$0A	**PRIORITY ERROR.** The priority specified for the work queue is invalid.
$0B	**INVALID BURST COUNT.** This error indicates that the requested burst size in the Controller Initialization Block is odd, or is greater than 256.

6

6

Table 6-6. MVME328 General Errors

Error Code (Hex)	Error Description
$10	**RESERVED FIELD ERROR.** A reserved field in the IOPB has nonzero data in it. This is indicative of a communications failure between the CPU and the MVME328.
$11	**RESET BUS STATUS.** The SCSI Reset command has executed successfully and generated a Reset on the bus. This is *not* an error.
$12	**SECONDARY PORT UNAVAILABLE.** A command has been issued for the secondary port (Port 1), but the port is not installed. This is indicative of a MVME328 daughter board failure.
$13	**SCSI ID ERROR.** The SCSI device ID request is the MVME328 own device ID. This is indicative of a communications failure between the CPU and the MVME328.
$14	**SCSI BUS RESET STATUS.** The command could not execute because the SCSI bus is held in the reset state. This may be caused by an unpowered device on the bus, improper termination, or an inverted cable.
$15	**COMMAND ABORTED BY RESET.** The command has been aborted due to a SCSI reset condition received during execution of the command. This may be caused by stopping SSID using the BREAK key; in any other circumstance, it is indicative of an SCSI bus error.
$16	**PAGE SIZE ERROR.** The Page Size field of the Controller Initialization Block is invalid. This is indicative of a communications failure between the CPU and the MVME328.
$17	**INVALID COMMAND TAG.** A command tag of zero was received by the MVME328. This is indicative of a communications failure between the CPU and the MVME328.
$18	**BUSY COMMAND TAG.** A command tag is on the bus during a Command Status Inquiry. This is indicative of SCSI bus failure.

Table 6-7. MVME328 VMEbus Errors

Error Code (Hex)	Error Description
$20	**VMEbus BUS ERROR.** A VMEbus error occurred during the DMA transfer of the data to or from the buffer or the bus.
$21	**VMEbus TIMEOUT.** The VMEbus acquisition was not completed within the programmed timeout period. This error is typically caused by a nonexistent address or address modifier in the IOPB. This may be indicative of a communications failure between the CPU and the MVME328.
$23	**VMEbus ILLEGAL ADDRESS.** The starting address of the VMEbus buffer must fall on a word or longword boundary. This is indicative of a communications failure between the CPU and the MVME328.
$24	**VMEbus ILLEGAL MEMORY TYPE.** An illegal memory type has been specified. This is indicative of a communications failure between the CPU and the MVME328.
$25	**ILLEGAL COUNT SPECIFIED.** An invalid transfer length was received by the MVME328. This is indicative of a communications failure between the CPU and the MVME328.
$26	**VMEbus FETCH ERROR.** A VMEbus error occurred during an offboard IOPB fetch. SSID does not use offboard IOPBs; thus, this indicates a communications failure between the CPU and the MVME328.
$27	**VMEbus FETCH TIMEOUT.** A VMEbus timeout occurred on an offboard IOPB fetch. SSID does not use offboard IOPBs; thus, this indicates a communications failure between the CPU and the MVME328.
$28	**VMEbus POST ERROR.** A VMEbus error occurred on an offboard Command Response Block post. SSID does not use offboard Command Response Blocks; thus, this indicates a communications failure between the CPU and the MVME328.
$29	**VMEbus POST TIMEOUT.** A VMEbus timeout occurred on an offboard Command Response Block post. SSID does not use offboard Command Response Blocks; thus, this indicates a communications failure between the CPU and the MVME328.
$2A	**VMEbus ILLEGAL FETCH ADDRESS.** A illegal address error occurred during an offboard IOPB fetch. SSID does not use offboard IOPBs; thus, this indicates a communications failure between the CPU and the MVME328.
$2B	**VMEbus ILLEGAL POST ADDRESS.** A illegal address error occurred during an offboard Command Response Block post. SSID does not use offboard Command Response Blocks; thus, this is indicative of a communications failure between the CPU and the MVME328.

6

Table 6-7. MVME328 VMEbus Errors (cont'd)

Error Code (Hex)	Error Description
$2C	**VMEbus SCATTER/GATHER FETCH.** VMEbus error occurred on scatter/gather list fetch. SSID does not use scatter/gather lists; thus, this is indicative of a communications failure between the CPU and the MVME328.
$2D	**VMEbus SCATTER/GATHER TIMEOUT.** VMEbus timeout occurred on scatter/gather list fetch. SSID does not use scatter/gather lists; thus, this is indicative of a communications failure between the CPU and the MVME328.
$2E	**INVALID SCATTER/GATHER COUNT.** An invalid number of scatter/gather elements has been specified. SSID does not use scatter/gather lists; thus this is indicative of a communications failure between the CPU and the MVME328.

6

Table 6-8. MVME328 SCSI Transaction Errors

Error Code (Hex)	Error Description
$30	**SCSI SELECTION TIMEOUT ERROR.** The selection phase of the SCSI device has failed. The selected device is probably not present. Use **slctdev** to reconfigure the system.
$31	**SCSI DISCONNECT TIMEOUT ERROR.** A disconnected device has not reselected the board in the timeout period. This may be indicative of a drive hardware error.
$32	**ABNORMAL SCSI SEQUENCE.** The SCSI operation did not complete successfully due to a hardware error or an abnormal operation sequence.
$33	**SCSI DISCONNECT ERROR.** An invalid SCSI bus sequence has been detected. This usually indicates a device has disconnected without either issuing the disconnect or command complete message. This may be due to a drive controller or SCSI bus failure.
$34	**SCSI TRANSFER COUNT EXCEPTION.** SCSI transfer count did not match requested count. This is not necessarily an error.
$35	**SCSI PARITY ERROR.** A parity error occurred during the Information Transfer phase on the SCSI bus.
$40	**ILLEGAL SCATTER/GATHER COUNT.** Odd byte count in scatter/gather list. SSID does not use scatter/gather lists; thus, this is indicative of a communications failure between the CPU and the MVME328.

Table 6-8. MVME328 SCSI Transaction Errors (cont'd)

Error Code (Hex)	Error Description
$41	**ILLEGAL SCATTER/GATHER MEMORY TYPE.** Illegal memory type in scatter/gather list. SSID does not use scatter/gather lists; thus, this is indicative of a communications failure between the CPU and the MVME328.
$42	**ILLEGAL SCATTER/GATHER ADDRESS.** Illegal address in scatter/gather list. SSID does not use scatter/gather lists; thus, this is indicative of a communications failure between the CPU and the MVME328.

6

Table 6-9. MVME328 Error Handling Codes

Error Code (Hex)	Error Description
$50	**READ/WRITE BUFFER COUNT ERROR.** Buffer count is too large. This is indicative of a communications failure between the CPU and the MVME328.
$51	**ILLEGAL READ/WRITE.** Cannot execute because of offboard Command Response Block. SSID does not use offboard Command Response Blocks; thus, this is indicative of a communications failure between the CPU and the MVME328.
$80	**FLUSH ON ERROR IN PROGRESS.** This status is set when the IOPB is flushed because an error condition has occurred and the work queue has the abort enable option set. Because SSID does not set this option, this is probably indicative of a communications failure between the CPU and the MVME328.
$81	**FLUSH WORK QUEUE STATUS.** Buffer count is too large. This is indicative of a communications failure between the CPU and the MVME328.
$82	**MISSING COMMAND.** A device has reselected the MVME328 for which there is no currently pending command. This is indicative of a SCSI bus error or a malfunctioning device.
$83	**COUNTER EXHAUSTED.** The transfer counter has exhausted but more data is being requested by the target device. This is indicative of a SCSI bus error or a malfunctioning device.
$84	**DATA DIRECTION ERROR.** A data phase is being requested opposite of the direction set in the IOPB. This is either an internal error within SSID itself or a MVME328 firmware error.

6

The presence of any printer port error is indicative of a communications failure between the CPU and the MVME328 SCSI Controller board or an MVME328 board failure. Running **VME328.*n*/controller 0** *(board self test)* executes the MVME328 onboard diagnostics to aid in further fault isolation.

Table 6-10. MVME328 Printer Port Errors

Error Code (Hex)	Error Description
$90	**PRINTER STATUS CHANGE.** Because SSID does no printer port tests, this error is indicative of an MVME328 internal error.
$91	**PRINTER COUNT TOO SHORT.** Because SSID does no printer port tests, this error is indicative of a MVME328 internal error.
$92	**BAD DATA LENGTH FIELD.** Because SSID does no printer port tests, this error is indicative of a MVME328 internal error.
$93	**PRINTER UNAVAILABLE.** Because SSID does no printer port tests, this error is indicative of a MVME328 internal error.
$99	**SCATTER/GATHER SELECTED FOR PRINTER PORT.** Because SSID does no printer port tests, this error is indicative of a MVME328 internal error.

Table 6-11. Other MVME328 Errors

Error Code (Hex)	Error Description
$C0	**BAD IOPB TYPE.** The IOPB type field does not match a currently supported IOPB type. This is either an internal error within SSID itself or a MVME328 board failure.
$C1	**IOPB TIMEOUT ERROR.** The IOPB has timed out due to some type of serious error. This is usually caused by a device, SCSI bus, or VME bus failure.

MVME350 Streaming Tape Controller

The test group for the MVME350 Streaming Tape Controller board consists of the **tp.0** tape tests shown below. Some of these tests destroy the data on the scratch tape.

To access the test menus for the MVME350 Streaming Tape Controller, type:

`SA:` **350.0**

The following appears:

```
Current Menu is /bdtest/350.0 - "VME350 Streaming Tape Board Tests"
tp.0 - Tape Tests
SA: tp.0 <CR>      (to access the test selection menu)
Selections for Test "Tape Tests"  (write-enabled scratch tape required)
   c) TAS CSR File       (TAS = test and set; CSR = command status register)
   0) Init
   1) Write Log/Eof      (Eof = end of file)
   2) Read Log/Eof
   r) Retension
   s) Erase
   t) Write EOT          (EOT = end of tape)
   v) Read EOT
SA:
```

The tests are described briefly below:

c) TAS CSR File
Uses a TAS instruction to verify that the MVME350 board's CSR is working. If this test fails, the CPU is probably not at the correct revision level.

0) Init
Resets the tape drive and returns the tape to BOT.

1) Write Log/Eof
Performs a short tape write test.

2) Read Log/Eof
Performs a short tape read test that should be run after test 1 (Write Log/Eof).

r) `Retension`
Performs a fast forward, then rewinds the tape.

s) `Erase`
Removes all data from the tape.

t) `Write EOT`
Performs a long tape write test that writes to the end of the tape (EOT).

v) `Read EOT`
Performs a long tape read test that should be run after test t (Write EOT).

MVME350 Board Test Configuration

6

The MVME350 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 350.0
/bdtest/350.0 configuration
   base[%0x100] = 0xfff5000 ?          # base address
   block size[%512] = 1024 ?           # logical block size
   tp[0-1] = 1 ?                       # number of streamer

> cf 350.0/tp.0
/bdtest/350.0/tp.0 configuration
   am[0x0d, 0x3d] = 0x3d ?             # address modifier
   level[0-7] = 4 ?                    # default interrupt level
   dbw[b,w,l] = l ?                    # date bus width (byte, word, long)
   no of test block = 150 ?            # number of test blocks
```

MVME355 9-Track 1/2-Inch Tape Controller

These tests verify controller operation by performing tests on Pertec-type interface, 9-track, ½-inch tape drives. The tests also verify the operation of the specified tape drive. Some of these tests destroy the data on the scratch tape.

To access the MVME355 9-Track Tape Controller test menu, type:

`SA:` **355.0**

A display similar to the following appears:

```
Current Menu is /bdtest/355.0 - "VME355 Controller/Drive Test"
ntpd.0 - VME355
SA: ntpd.0 <CR>      (to access the test selection menu)
Selections for Test "VME355"    (write-enabled scratch tape required)
   0) Reset/Initialization
   1) Drive Status
   2) Write to EOT
   3) Write Blocks
   4) Read to EOT
   5) Read Blocks
   6) Write/Read to EOT
   7) Erase to EOT
   8) Rewind to Load Point
   9) Controller Revision Level
   a) Dump Blocks in Hexadecimal/ASCII Format
   b) Set NRZI
   c) Clear NRZI
SA:
```

The number and size of the blocks used in these tests is set in the test configuration file. Refer to *MVME355 Board Test Configuration* below for details. The MVME355 tests are explained below:

0) Reset/Initialization

Verifies that communication with the controller can take place. It resets the controller and sets the default unit initialization block. At this point the board-edge LED turns from red to green. The remaining tests in the sequence execute this initialization process provided that no other tape drive(s) is being tested.

6

1) `Drive Status`
Verifies that the tape drive is online and ready. The remaining tests in the sequence execute this status check of the tape drive before the body of the test is executed.

2) `Write to EOT`
Writes multiple blocks of data to the tape until an EOT error occurs. The diagnostics then report how many blocks of data were written to the tape. The test must start with the tape positioned at the beginning of tape. Upon completion, it returns the tape to BOT.

3) `Write Blocks`
Writes multiple blocks of data until the count is exhausted. Upon completion, it returns the tape to BOT.

4) `Read to EOT`
Reads multiple blocks of data from the tape until an EOT occurs. The diagnostics then report how many blocks of data were read from the tape. The test is dependent on data being present on the tape (refer to `Write to EOT`, above) and must start with the tape positioned at BOT. Upon completion, it returns the tape to BOT.

5) `Read Blocks`
Reads multiple blocks of data until the count is exhausted. The test is dependent on data being present on the tape (refer to `Write Blocks`, above). Upon completion, it returns the tape to BOT.

6) `Write/Read to EOT`
Writes multiple (256) blocks of 512-byte data in a single write. After the write, the tape is repositioned over the data blocks written. The tape is then read for 256 blocks (512-bytes in length). Next, the data read is compared to the data written; any errors are reported. The above process is repeated until an EOT error occurs. Upon completion, it returns the tape to BOT.

7) `Erase to EOT`
Removes all data from the tape. Upon completion, it returns the tape to BOT.

8) `Rewind to Load Point`
Rewinds the tape to the load point (position tape at the beginning of tape).

9) `Controller Revision Level`
Asks the controller its revision level and revision date, then displays them.

a) `Dump Blocks in Hexadecimal/ASCII Format`
This option is more a utility than a test. It prompts for the data block on the tape you want to view. Input may be in decimal or hexadecimal numbers. This function assumes that there is data on the tape at the specified block number. The data displayed is of the hexadecimal/ASCII format dump. The following options are recognized at the block number prompt: **c** – display current block position; **h** – display options list; **q** – exit. Pressing the **RETURN** key causes the utility to rewind to BOT.

b) `Set NRZI`
This sets the NRZI bit. It is used for the **TELEK 7942** tape drive while it is in low density mode (800bpi).

c) `Clear NRZI`
This clears the NRZI bit and is the Default setting.

6

MVME355 Board Test Configuration

The MVME355 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 355.0
/bdtest/355.0 configuration
   base[%0x100] = 0xfff0800 ?                    # base address
   am[0x0d, 0x3d] = 0x3d ?                        # address modifier
   level[0-7] = 3 ?                               # default interrupt level
   dbw[w,l] = w ?                                 # date bus width (word, long)
   ntpd[0-1] = 1 ?                                # number of drives/tests

> cf 355.0/ntpd.0
/bdtest/355.0/ntpd.0 configuration
   physical drive[0-1] = 0 ?                      # tape drive number
   dma rate[0-255] = 16 ?                         # dma transfer rate
   block size (bytes)[256-131072] = 512 ?         # bytes per block
   number of blocks = 200 ?                       # number of blocks
   retries[0-14] = 3 ?                            # retry count
   debug[0-1] = 0 ?                               # debug flag
   iopbug[0-1] = 0 ?                              # iopb debug flag
   J1-14[0-1] = 0 ?                               # j1-14, byte 0 of uib
   J1-16[0-1] = 0 ?                               # j1-16, byte 1 of uib
   J1-36[0-3] = 0 ?                               # j1-36, byte 2 of uib
   J1-44[0-2] = 0 ?                               # j1-44, byte 3 of uib
   J2-26[0-2] = 0 ?                               # j2-26, byte 4 of uib
   J2-50[0-2] = 0 ?                               # j2-50, byte 5 of uib
   NRZI[0-1] = 0 ?                                # NRZI,  byte 6 of uib
   poffts[0-255] = 255 ?                          # poffts, byte 0xf of uib
```

MVME355 Error Codes

All error codes are in hexadecimal format.

10 - tape not ready
Tape drive's ready signal output is tested at the beginning of any command requiring tape movement. Error 10 is posted if the tape is not ready.

14 - invalid command code
The command code, Byte 0 in the IOPB, is not valid.

15 - illegal fetch and execute command
Indicates that a Fetch and Execute command was encountered in external memory. A Fetch and Execute command is only valid when it occurs in the onboard Short I/O space.

6

17 - illegal memory type
The memory type specified for the buffer address or IOPB address is illegal.

18 - bus time out
A bus record transfer was not completed within one second after a request.

19 - not used
The command issued may be supported in a later release.

1A - tape write protected
Posted when attempts are made to write to a tape that is write protected.

20 - end of tape
A complete record write was attempted after EOT. No data was transferred.

21 - load point error
A reverse movement command encountered the load point before completion.

23 - uncorrectable error
Error correction was attempted by the tape drive on the data field and the error was found to be uncorrectable.

24 - parity error
While reading data from the tape, a parity error was detected by the controller. This error is not reported when the drive is operated at 800 bpi (NRZI).

25 - `read long error`

The number of bytes per record requested in a Read operation was less than the number of bytes actually contained in the tape record. Only the number of bytes requested was transferred into system memory. The Residual Record Count field of the IOPB will contain the number of non-transferred records, which includes the record that caused this error. The Residual Byte Count field of the IOPB will contain an approximate count of the bytes in the tape record that were not transferred.

26 - `short read error`

The number of bytes per record requested in a Read operation was greater than the number of bytes actually contained in the tape record. However, the number of bytes requested were transferred into system memory. Both the Residual Byte Count and the Residual Record Count fields of the IOPB are valid.

42 - `signal definition error`

An illegal value was specified in one or more of the first 6 bytes of the UIB.

52 - `VME bus error`

The VMEbus system controller activated the BUS ERROR signal during a transfer by the MVME355.

58 - `fifo error`

An internal hardware error occurred on the MVME355 causing the data buffer to overflow or underflow. This only happens if the MVME355 cannot access the bus for very long periods of time.

5A - `invalid address modifiers`

An invalid value (greater than 3F) was detected in the Address Modifier field of the IOPB. The following table lists valid (not reserved) address modifier codes.

Table 6-12. Address Modifier Codes

Code	Function
3E	Standard Supervisory Program Access
3D	Standard Supervisory Data Access
3A	Standard Non-Privileged Program Access
39	Standard Non-Privileged Data Access
2D	Short Supervisory I/O Access
29	Short Non-Privileged I/O Access
0E	Extended Supervisory Program Access
0D	Extended Supervisory Data Access
0A	Extended Non-Privileged Program Access
09	Extended Non-Privileged Data Access

All other address modifier codes are undefined.

`5B - invalid memory address`
An invalid value was detected in the Memory Buffer address field of the IOPB. Only memory addresses starting on word boundaries are supported.

`5C - invalid interrupt level`
An invalid value was detected in the Interrupt Level field of the IOPB; 1 to 7 are valid values.

`5D - illegal DMA burst count`
The DMA burst count specified was not within the valid range. Valid values are 8 to 255, or 0 for a VMEbus system using priority arbitration.

`61 - DMA failed`
A bus error occurred during the DMA transfer of data.

`62 - invalid UIB parameter`
An invalid parameter was detected in the UIB, other than the first 6 bytes. Errors in the first 6 bytes have an error code of 42.

`63 - tape timeout`
The specified tape drive did not respond properly to a command in a calculated period of time.

`64 - invalid IOPB parameter`
An invalid parameter was found in the IOPB, other than the parameters covered by error codes 52 through 5D or 65.

65 - invalid record size
This error is also reported if the Bytes per Record field in the IOPB is larger than the buffer size when a Read or Write Data Buffer command is issued.

66 - illegal transfer size
A tape record size of less than 16 bytes was issued in the IOPB.

67 - illegal erase
An erase command was issued for a drive without the Erase Enable bit set (Bit 5 in the UIB attributes byte) for that drive.

68 - filemark encountered
A filemark was unexpectedly found during a read or a tape movement command.

70 - time out on command
The selected drive did not return proper status within 500 milliseconds after the command was issued.

71 - command not accepted
The selected drive did not acknowledge the command from the controller.

80 - command aborted
The Abort bit of the CSR was set during command execution.

FF - command not implemented
The command issued will be supported in a later release.

XX - **unknown status**
This is any error code not defined above that might occur due to erroneous operation. Contact your sales or field service representative for assistance.

MVME360 SMD Hard Disk Controller

The MVME360 board tests verify controller operation by performing nondestructive, read-only tests on the SMD Winchester disk drives. Write tests are available but destroy any data on the disk.

A maximum of two MVME360 controllers can be installed in a system, with a maximum of two fixed SMD Winchester drives per controller.

NOTE

Removable SMDs are not supported.

The drives to be tested must be initialized (formatted or have SYSTEM V/68 or SYSTEM V/88 file system on them). They can be formatted with or without alternate track mapping. The tests automatically handle the alternate tracks.

To access the test menus for the first MVME360 board, type:

`SA:` **360.0**

A display similar to the following appears:

```
Current Menu is /bdtest/360.0 - "VME360 SMD Controller Tests"
smd.1 - SMD Disk Test
smd.0 - SMD Disk Test
SA: smd.0 <CR>      (to access the test selection menu for the first SMD disk drive)
Selections for Test "SMD Disk Test"
   0) Recalibrate
   1) Confidence test
   2) Read entire Disk
   3) Random read Disk
   4) Ping-Pong read Disk
   a) Read Sector header
   b) Controller check
   c) Get uib from drive
   d) Read sector x
   e) Calculator
SA:
```

For a description of tests `0` through `4`, refer to the *Winchester Disk Tests* section.

Menu items **a** through **e** and abbreviations that might appear during testing are described in the *MVME323 ESDI Disk Drive Controller* section.

MVME360 Board Test Configuration

The MVME360 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 360.0
/bdtest/360.0 configuration
   base[%0x200] = 0xfff0c00 ?        # base address
   am[0x0d, 0x3d] = 0x3d ?           # address modifier
   level[0-7] = 4 ?                  # default interrupt level
   dbw[b,w,l] = w ?                  # date bus width (byte, word, long)
   smd[0-2] = 2 ?                    # number of drives
```


> **CAUTION**
>
> **Testing an SMD drive with the `eval flag*` variable set to `1-on` will allow destruction of all data on the disk drive!**

```
> cf 360.0/smd.0
/bdtest/360.0/smd.0 configuration
   Controller type [ESDI,SMD] = SMD ?
   phy-drive[0-1] = 0 ?                       # number of hard disk drives
   Alt track 0-no 1-yes[0-1] = 1 ?            # alternate track handling
   sector/xfer[1-128] = 64 ?                  # size per buffer
   dma burst[1-255] = 10 ?                    # dma burst rate
   start head = 0 ?
   tracks/cylinder = 10 ?                     # number of tracks per cylinder
   sectors/track = 64 ?                       # number of sectors per track
   spiral skew = 0 ?
   sector size[128,256,512,1024] = 512 ?# number of bytes in a sector
   gap1 = 16 ?
   gap2 = 16 ?
   interleave factor[1-16] = 1 ?              # interleave factor
   retries[0-255] = 3 ?                       # retry count
   cylinders/drive = 823 ?                    # number of cylinders per drive
   attribute = 0x5 ?                          # inc by head, reSeek
   format with[ECC,NON] = ECC ?               # format with ECC or without ECC
 * Eval Flag 0-off 1-on[0-1] = 0 ?            # not recommended for general use
```

(The following variables are ignored unless **eval flag** *is set to* **1-on***.)*

```
lockout cylinder = 0 ?                      # ending range of cylinder tested
lockout logical sector number = 0 ?         # calculated log. position of
                                            # lockout cyl.
read/seek flag 0-read 1-seek[0-1] = 0 ?
read/check flag 0-no 1-yes[0-1] = 0 ?
write data pattern = 0xdb60db6 ?
Max number of random seeks = 10000 ?       # applies to speed test only
Max number of alternate tracks = 144 ?     # automatically determined by test
Conf test flag 0-off 1-on[0-1] = 0 ?       # confidence test flag for
                                            # destructive disk tests
```

6

CHAPTER 7
COMMUNICATION CONTROLLER BOARD TESTS

This chapter describes the board tests available for communication controller boards. Each section describes the tests available for a particular controller board, the board's test configuration, and, in some cases, the error codes associated with the board. For an introduction to board testing, refer to the *Board Testing* section in the *Confidence, Fault, Board, and Peripheral Testing* chapter.

MVME330 Local Area Network Controller

There are two versions of the MVME330 board:

`G330` Generic Support Package
with Buffer Pipe Protocol/Common Environment (BPP/CE) firmware

`T330` Transmission Control Protocol/Internet Protocol (TCP/IP)

G330 and T330 use the same tests but different downloading protocols. G330 follows BPP for downloading, while T330 downloads test codes to dual access RAM from host.

To access the test menus for the Generic Support Package version of the MVME330 board, type:

`SA:` **G330.0**

The following display appears for a G330 board:

```
Current Menu is /bdtest/G330.0 - "VME330 GSP LAN Board Test"
gsplan.0 - Ethernet GSP LAN Test
SA: gsplan.0 <CR>           (to access the test selection menu)
    0)  LANCE CSR           (LANCE = Local Area Network Controller for Ethernet)
    1)  LANCE Init
    2)  LANCE Internal Loopback
    a)  External Loopback   (Connect MVME330 to Ethernet coax via transceiver/tap)
    b)  Board Status Check
SA:
```

7

The following describes the MVME330 tests:

0) `LANCE CSR`
Checks each bit in the four command status registers, making sure that each bit can be cleared and set.

1) `LANCE Init`
Tests the initialization of the LANCE chip with interrupt enabled and disabled. It also verifies that an interrupt is received when initialization is finished.

2) `LANCE Internal Loopback`
Transmits and receives data between two points within the LANCE chip.

a) `External Loopback`
Transmits to and receives data from the Ethernet cable. The LAN board must be connected to Ethernet coax via transceiver/tap to run this test.

b) `Board Status Check`
A quick confidence check which ensures that there is some kind of board activity.

G330 Board Test Configuration

The G330 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf G330.0
/bdtest/G330.0 configuration
   base[%0x1000] = 0xdc1000 ?            # base address
   am[0x0d,0x3d] = 0x3d ?                # address modifier
   gsplan[0-1] = 1 ?                     # LAN test, can only be one
> cf G330.0/gsplan.0
/bdtest/G330.0/gsplan.0 configuration
   local status = 0x001806 ?             # local status location
   local command = 0x001805 ?            # local command location
   interrupt location = 0x01ee00 ?       # interrupt location
   xfer size = 1024 ?                    # buffer size
```

T330 Board Test Configuration

The T330 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf T330.0
/bdtest/T330.0 configuration
   base[%0x1000] = 0xde1000 ?            # base address
   am[0x0d,0x3d] = 0x3d ?                # address modifier
   tcplan[0-1] = 1 ?                     # LAN test, can only be one
> cf T330.0/tcplan.0
/bdtest/T330.0/tcplan.0 configuration
   local status = 0x001806 ?             # local status location
   local command = 0x001805 ?            # local command location
   interrupt location = 0x01ee00 ?       # interrupt location
   xfer size = 1024 ?                    # buffer size
```

7

MVME331 and MVME332 Asynchronous Communications Controller

These tests check all port pairs on the MVME331 and MVME332 communication boards. The MVME332 has four port pairs; the MVME331 has three port pairs. Loopback cables are required and should be connected as follows:

```
MVME331 :   SP5 <-------> SP6
            SP3 <-------> SP4
            SP1 <-------> SP2
MVME332 :   SP7 <-------> SP8
            SP5 <-------> SP6
            SP3 <-------> SP4
            SP1 <-------> SP2
```

See Figure 4-1 for the proper pinout for the loopback cables.

To access the test menus for the first VME332 8-port communication board, type:

SA: **332.0**

The following display appears:

```
Current Menu is /bdtest/332.0 - "VME332 8 Port Comm. Board Test"
sp.3                  - Serial Port Pair
sp.2                  - Serial Port Pair
sp.1                  - Serial Port Pair
sp.0                  - Serial Port Pair
SA: sp.0 <CR>   (to access the test selection menu for the communication ports SP1 and SP2)
          0)  ext loop back
          1)  single char loop back
          2)  break detection
          3)  abort read
          4)  abort write/read
          5)  echo
          6)  full duplex
          7)  overrun
          8)  parity error
          9)  hw flow control
          a)  frame error
          b)  Board alive check      (No loopback cable required for tests b, c, or d)
          c)  String output to device  (Do NOT run with loopback cable connected.)
          d)  Display EIA status
(This menu displays for all communication port pairs on the MVME331 and MVME332 boards.)
SA:
```


A description of the MVME331 and MVME332 tests follows:

`0) ext loop back`
`1) single char loop back`

Transmit and receive data between two ports; the tests write to the second port and read from the first port, then write to the first port and read from the second port. Test 0 transmits and receives a block of characters; test 1 transmits and receives a single character.

`2) break detection`

Forces a break in the line, then waits for a break detect message to be received.

3) **`abort read`**

4) **`abort write/read`**
Ensure that the read and write/read processes abort on cue.

5) **`echo`**
Tests the ability of the port to transmit information it receives.

6) **`full duplex`**
Similar to tests 0 and 1. This test checks that data can be transmitted to and received at both ports simultaneously.

7) **`overrun`**
Ensures that the controller can detect an overrun error.

8) **`parity error`**
Generates a parity error on one port, then checks if it is detected by the other port.

9) **`hw flow control`**
Checks the hardware signals that control data.

a) **`frame error`**
Generates the wrong baud rate to ensure that the controller can detect a frame error.

b) **`Board alive check`**
This quick confidence check verifies that there is board activity.

c) **`String output to device`**

NOTE

Do not run with loopback cable connected.

This menu-driven test ensures that data is output to the selected device. It is useful for checking transmission of characters to printers or terminals connected to the MVME331 or MVME332 board. This test displays the string it will output on the test console, then sends the string to the selected port.

d) **`Display EIA status`**
Displays the Electronic Industries Association Standard RS-232C signals.

MVME331 Board Test Configuration

The MVME331 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 331.0
/bdtest/331.0 configuration
   base[%0x100] = 0xfff3000 ?          # base address
   am[0x0d, 0x3d] = 0x3d ?             # address modifier
   level[0-7] = 3 ?                    # default interrupt level
   sp[0-3] = 3 ?                       # number of communication port pairs

> cf 331.0/sp.0
/bdtest/331.0/sp.0 configuration
   src port no[1-6] = 2 ?              # source serial port number
   dest port no[1-6] = 1 ?             # destination serial port number
   xfer size = 1024 ?                  # buffer size
   baud rate = 9600 ?                  # baud rate
```

7

MVME332 Board Test Configuration

The MVME332 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 332.0
/bdtest/332.0 configuration
   base[%0x100] = 0xfff6000 ?          # base address
   am[0x0d, 0x3d] = 0x3d ?             # address modifier
   level[0-7] = 3 ?                    # default interrupt level
   sp[0-4] = 4 ?                       # number of communication port pairs

> cf 332.0/sp.0
/bdtest/332.0/sp.0 configuration
   src port no[1-8] = 2 ?              # source serial port number
   dest port no[1-8] = 1 ?             # destination serial port number
   xfer size = 1024 ?                  # buffer size
   baud rate = 9600 ?                  # baud rate
```

MVME331 and MVME332 Error Codes

The VME33X test module, which covers MVME331 and MVME332 but not MVME332XT, prints out pertinent error messages when it detects unexpected test conditions. If the error is detected by the VME33X, the contents of the packet and the English translation of the error code both display. The following is a description of all error messages the test module uses followed by a description of the packet display:

TIMEOUT

The test module prints a timeout message if an event is not accomplished within a predefined time limit. In the VME33X test, the time out occurs when a certain time has elapsed after a command is sent to the VME33X module and the module has not yet responded.

DATA COMPARISON ERROR

Most VME33X tests transmit data through one port and receive from the other port. If the data read differs from the expected data, a data comparison error occurs. The error message is:

```
/bdtest/33X.0/sp.0  At address=001aff00, expect=44, read=00
```

This message means that a data comparison error occurred on serial port pair, logical number 0. The address of the data in error is at location 001aff00, the expected value of the data is 44, and the actual data read is 00.

OTHER ERRORS

Table 7-1 lists the status codes that the VME33X returns through the `stat` field in the packet.

Table 7-1. Status Code List

Code	Status
$00	Normal command termination
$01	End of system memory block
$02	Read timeout
$03	Write timeout
$04	Command aborted
$05	Insufficient local memory
$06	Invalid item number
$07	Invalid baud rate
$08	Invalid number of bits per character
$09	Invalid number of stop bits
$0A	Invalid type of parity
$0B	Invalid handshake selection
$0C	RX queue overrun
$0D	Invalid delete character string length
$10	Invalid start address
$14	Least number of characters received
$81	Parity error
$82	Overrun error
$83	Overrun and parity error
$84	Framing error
$85	Overrun and framing error

Status Packet Display

To provide sufficient status information, the contents of the packet are displayed when there is an error status returned by the VME33X. The packet is also displayed when there is a data comparison error detected by the test. The English translated error code in the main status and the extended status fields display after the packet contents. You can also force this packet to display every command and status with or without error by typing **set dp** in command line mode (refer to Appendix A). To clear the flag, type **set -dp**. Figure 7-1 is a typical example of the packet display.

```
            Packet address = 1f9aa8, ID = sp.0
|  flnk   |cmd |port|  saddr  |  len    |stat|term|  rlen   |
|ffffffff|01  | 01 |000187b8|00000400| 0c | 00 |00000010|

error:  Rx queue overrun
```

Figure 7-1. Packet Display

The message after `error:` is interpreted directly from the status returned by the controller. Each field of the packet display is defined according to the *MVME332 Intelligent Communication Controller User's Manual*. The following describes these terms:

`flnk`
: is one longword that is used internally by the VME33X firmware for packet linking purposes. `flnk` does not need to be set nor monitored by the host.

`cmd`
: is one byte set by the host. `cmd` is a code number that specifies the command. Valid command codes are: `$01` to `$07` and `$0B` to `$13`. Refer to the *VME33X User's Manual* for the command definition.

`port`
: is one byte set by the host. `port` is a number that specifies the serial port affected by the command. Valid `port` numbers are: `$01` to `$06` for the MVME331 and `$01` to `$08` for the MVME332.

`sad`
: is one longword (4 bytes) set by the host. `sad` defines the start address of a data area in system memory for command use. Such data may be characters to be transmitted or received, configuration tables, or other additional command parameters. Valid start addresses are any system RAM addresses accessible by the controller.

`len`
: is one longword set by the host. `len` defines the size of a data area in system memory for command use. Such data may be characters to be transmitted or received, configuration tables, or other additional command parameters. The sum of `sad` and `len` must not exceed the system RAM addresses accessible by the communications controller. The maximum value for `len` is `$7fffffff`.

`stat`
: is one byte returned by the VME33X. `stat` is a status code number that is zero when the command has been completed successfully. In the case of abnormal terminations, `stat` indicates the condition that aborted the command. Refer to the *VME33X User's Manual*.

`term`
: is one byte returned by the VME33X. `term` is used with the read and write commands and is the character that caused command termination. `term` can be the previously defined terminating character or the bad character in the case of a parity or framing error.

`rlen`
: is one longword returned from the VME33X. `rlen` is used by commands that involve the transfer of data to or from system memory. `rlen` is the number of bytes read from or written to system RAM at the time of command termination. In the case of abnormal command terminations, `rlen` can be used by the host to update the `sad` and `len` fields, and to reissue the command that continues the previous aborted operation.

Non-Fatal Errors

SSID records both fatal and non-fatal errors. To see the error count, use the **fe** command in command mode (refer to Appendix A). The non-fatal errors are the errors that could be corrected by software retry. In the case of the VME33X, all errors are considered fatal errors.

Fatal Errors

Fatal errors are hard, non-recoverable errors. In the VME33X test, all errors are fatal errors and fall into these categories:

- Errors flagged in the status returned by the VME33X.
- Data comparison errors.
- Timeout errors that occur when a packet is not returned by the controller after a certain time.

MVME333 Wide Area Network Controller and MVME333X25 X.25 Controller

The MVME333 test software supports a wide range of board firmware types and board positions. The test software supports up to 19 boards with the following names and addresses:

333a.0 - 0xFFFF3800	333b.0 - 0xFFFF3900	333c.0 - 0xFFFF3A00
333d.0 - 0xFFFF3B00	333e.0 - 0xFFFF3C00	333f.0 - 0xFFFF3D00
333g.0 - 0xFFFF3E00	333h.0 - 0xFFFF3F00	333i.0 - 0xFFFF4000
333j.0 - 0xFFFF4100	333k.0 - 0xFFFF4200	333l.0 - 0xFFFF4300
333m.0 - 0xFFFF4400	333n.0 - 0xFFFF4500	333o.0 - 0xFFFF4600
333p.0 - 0xFFFF4700	333q.0 - 0xFFFF4800	333r.0 - 0xFFFF4900
333s.0 - 0xFFFF4A00		

7

There are three types of firmware supported by the test: CE1, CE3, and X.25. They are supported at any of the above addresses in any combination. When the first of any of the following tests is run, that test will probe all of the above addresses and identify each board that is in the system. It will then cross load each of the boards using the correct interface and save the status for all the boards. Each of the above firmware has a different cross loader interface. The cross loader interface is used to load an object module from the host's memory into the MVME333 memory. This cross loaded object takes control of the MVME333 by first loading itself into the board, then initializing itself. When complete, it reports ready for work status.

All test reports about the communication ports are relative to the MVME333 communication ICs. For all MVME333 test reports about the communication port numbering, refer to the MVME333 Intelligent Communication Controller User's Manual for the interface pinout to the P2 bus connector. The ports are labeled 1 to 6. Port 1 is the A port of the first Z8530 communication IC, port 2 is the B port of the first Z8530, and so on. A transition board manual will be needed to determine the external connections of all signals. Each transition board has different and possibly configurable connections.

To access the test menus for the first MVME333 controller, type:

```
SA: 333a.0
```

The following appears:

```
Current Menu is /bdtest/333a.0-"VME333 Communications Board Test"
sp.0                  - Serial Port Pair
SA: sp.0 <CR>  (to access the test selection menu)

          (The following menu is displayed for all MVME333 controllers.)
       0)  Board Confidence Test
       1)  Board Hardware Test
       2)  Internal Loopback Test
       3)  External Loopback Test
       b)  Display EIA Signal Status
       d)  Display Loopback Connection(s)
       y)  Display Board Type And Address
                      NOTE: Only asynchronous mode is tested.
SA:
```

7

A description of the MVME333 and MVME333X25 tests follows:

0) **Board Confidence Test**
Causes the MVME333 to transfer a 1024-byte command block to the MVME333's on-board memory set status and write it back to the host's memory. The test then verifies the return status block.

1) **Board Hardware Test**
Causes the MVME333 to execute a write/read register test on the MVME333's communication ICs.

2) **Internal Loopback Test**
Causes the MVME333 to execute an internal loopback test on the MVME333's communication ICs. External loopbacks are not required for this test.

Note that this test transmits data on the Transmit Data pin of the transition board at variable baud rate and stop bit setings. External connections other than loopbacks should be disconnected if this is a problem.

3) **External Loopback Test**
Causes the MVME333 to execute an external loopback test on the MVME333's communication channels. External loopbacks are required for this test. The test automatically detects the loopback connections and reports them back as part of the test. The test will fail if no loopbacks are detected.

Note that the test will pass if only one loopback is detected, even if that is not the actual configuration. Before execution of the external loopback test, the display loopback connections test should be executed for each of the MVME333 boards in the system to determine if loopback connections are as desired.

Note also that the CE1 firmware comes up in its debug mode if there is a loopback connected to port 1. Thus, for boards with CE1 firmware, port 1 cannot be tested in external mode.

7

b) **Display EIA Signal Status**
Causes the MVME333 to execute a get EIA status test, which returns the current EIA status for the communication channels — i.e., the current states of the CTS, DCD, and DSR input signals.

d) **Display Loopback Connection(s)**
Causes the MVME333 to report the current detected external loopback connections.

y) **Display Board Type And Address**
Displays the detected board's type (CE1, CE3, or X.25) and the board's address in this system.

MVME333 WAN and MVME333X25 Board Test Configuration

The MVME333 WAN and MVME333X25 board test configurations are shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 333.0
/bdtest/333.0 configuration
   base[%0x100] = 0xfff3800 ?          # base address
   am[0x0d, 0x3d] = 0x0d ?             # address modifier
   level[0-7] = 1 ?                    # default interrupt level
   sp[0-1] = 1 ?                       # test available
```

MVME333 WAN and MVME333X25 Transition Board Loopback Connections

Table 7-2. MVME333 and MVME333X25 Loopback Connections

Board	Connections	Transition Board
MVME333 X.25	SP1 <-------> SP3	(MVME705B)
	SP1 <-------> SP2	(MVME705-1)
	SP1 <-------> SP3	(MVME705, MVME705A)
MVME333 WAN with CE1	SP3 <-------> SP5	(MVME705B)
	SP2 <-------> SP3	(MVME705-1)
	SP3 <-------> SP4 SP5 <-------> SP6	(MVME705, MVME705A)
MVME333 WAN with CE3	SP3 <-------> SP5	(MVME705B)
	SP2 <-------> SP3	(MVME705-1)
	SP1 <-------> SP2 SP3 <-------> SP4 SP5 <-------> SP6	(MVME705, MVME705A, MVME705A-1, MVME705A-2)

Table 7-3. DB9 Loopback Pinouts

Transition Board	Pinouts
MVME705A-1 RS422	DB9------DB9 2 <-------> 4 4 <-------> 2 6 <-------> 9 9 <-------> 6
MVME705A-2 RS232	DB9------DB9 2 <-------> 3 3 <-------> 2 4 <-------> 5 5 <-------> 4 6 <-------> 8 7 <-------> 9 8 <-------> 6 9 <-------> 7

For the proper pinout for the DB-25 loopback cables, see Figure 4-1.

MVME333 Error Messages

The following describes the MVME333 error messages:

`333a.0/sp.0: This board does not exist in the system`
The board is not being detected by the program.

`333a.0/sp.0: This board's firmware is unknown by the test package`
This board's startup status does not match any of the known startup status values.

`333a.0/sp.0: This board is in an error state and no more testing can be executed on it`
This board is in some unknown error state and is not responding to the test.

`333a.0/sp.0: This board is reporting a power up error condition`
This board has some type of power up self test failure. No more testing can be done on this board.

7

`333a.0/sp.0: Download failed for X.25 board`
`333a.0/sp.0: Download failed for CE1 board`
`333a.0/sp.0: Download failed for CE3 board`
These errors may or may not be for this board, since all boards are downloaded by the first test to reach the download code. When the test for which the download failed is executed, it will report and record a failure.

MVME332XT Asynchronous Communications Controller

These tests check all port pairs on the MVME332XT communication boards. To run these tests, loopback cables are required and should be connected as:

```
SP7 <-------> SP8
SP5 <-------> SP6
SP3 <-------> SP4
SP1 <-------> SP2
```

See Figure 4-1 for the proper pinout for the loopback cables.

All tests, except the baud rate tests, are executed with the ports set to 38.4K baud.

There is no specific download test provided for the MVME332XT board; however, a download is performed as part of the hardware flow control test, providing a check of the download capability of the board.

To access the test menus for the first MVME332XT communication board, type:

SA: x332.0

7

The following display appears:

```
Current Menu is /bdtest/x332.0 - "VME332xt Async Comm"
sp.3                - Serial Port Pair
sp.2                - Serial Port Pair
sp.1                - Serial Port Pair
sp.0                - Serial Port Pair
SA:sp.0 <CR> (to access the test selection menu for the communication ports SP1 and SP2)
    1)  open port a             a)  DCD a -> b
    2)  open port b             b)  DCD b -> a
    3)  loopback a -> b         c)  baud rate a -> b
    4)  loopback b -> a         d)  baud rate b -> a
    5)  full duplex             e)  parity a -> b
    6)  event a -> b            f)  parity b -> a
    7)  event b -> a            g)  frame a -> b
    8)  break a -> b            h)  frame b -> a
    9)  break b -> a            i)  HW flow ctl a -> b
                                j)  HW flow ctl b -> a
(This menu is displayed for all communication port pairs on the MVME332XT boards).
SA:
```

The following is a description of the MVME332XT tests:

1) `open port a`
2) `open port b`
   These tests open and then close a port as a simple *board alive* check.

3) `loopback a -> b`
4) `loopback b -> a`
   Transmit and receive data between two ports, comparing the results.

5) `full duplex`
   Checks that data can be transmitted to and received at both ports simultaneously.

6) `event a -> b`
7) `event b -> a`
   Verify that the ports return the correct event packets when interrupt and quit signals are received. Event packets are used for communication between the operating system and the MVME332XT board.

8) `break a -> b`
9) `break b -> a`
   Force a break in the line and verify a `break detect` message is received.

a) `DCD a -> b`
b) `DCD b -> a`
   Cause DCD to be negated on one port and verify that the other port detects the loss of DCD.

c) `baud rate a -> b`
d) `baud rate b -> a`
   Verify each port at baud rates of 50, 75, 110, 300, 1200, 2400, 4800, 9600, 19.2K, and 38.4K.

e) `parity a -> b`
f) `parity b -> a`
   Generate a parity error on one port and check to see if it is detected by the other port.

g) `frame a -> b`
h) `frame b -> a`
   Set one port to 4800 baud and the other to 9600 baud, transfer data from one port to the other, and verify that a framing error is detected.

`i) HW flow ctl a -> b`
`j) HW flow ctl b -> a`
Set each port to use hardware flow control, transfer data from one port to the other, and compare the results.

MVME332XT Board Test Configuration

The MVME332XT board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf x332.0
/bdtest/x332.0 configuration
   base[%0x10000] = 0xff780000            # base address
   am[0x0d, 0x3d] = 0xd                   # address modifier
   level[0-7] = 2                         # default interrupt level
   sp[0-4] = 4                            # number of communication port pairs
```

```
> cf x332.0/sp.0
/bdtest/x332.0/sp.0 configuration
   port a no[0-7] = 0                     # serial port number
   port b no[0-7] = 1                     # serial port number
   xfer size[0-2000] = 1024               # buffer size
```


MVME332XT Error Messages

`ring buffer allocation exceeds 64k`
: The ring buffer allocation exceeds the available MVME332XT dual ported RAM area of 64K.

`Initialization error`
: Board initialization was unsuccessful.

`TIMEOUT`
: A time out has occurred. This happens if a packet is sent to the MVME332XT, but the expected status packet is not returned within a certain period of time.

`port open timeout`
: A serial port open command is sent to the MVME332XT, but no response is received within a certain period of time.

`dev never ready`
: A serial port is expected to have DCD asserted status from the MVME332XT, but DCD is never asserted.

`Free packet pool is empty`
The test requests a free packet but none is available. This indicates there is memory corruption or a real memory shortage.

`DCD did not NEGATE`
DCD is expected to be negated on a serial port, but it is not.

`DCD did not ASSERT`
DCD is expected to be asserted on a serial port, but it is not.

`not enough memory`
There is not enough system memory available for the test to use. This indicates a real memory shortage or memory corruption.

`no INTR event`
An INTR event is expected from a serial port, but the INTR event is never reported by the MVME332XT event packet.

`no QUIT event`
A QUIT event is expected from a serial port, but the QUIT event is never reported by the MVME332XT event packet.

`missing BRK INTR`
A break interrupt is expected from a serial port, but the MVME332XT never reported one.

`cannot execute downloaded code`
A piece of code is downloaded to the MVME332XT for execution, but the MVME332XT detected an error and the code was not executed successfully.

`parity mark did not work`
The parity mark option is selected, but the MVME332XT did not mark parity as expected when a parity error occurs.

`at addr=`*xxxxxxxx*`, expect=`*xxxxxxxx*`, read=`*xxxxxxxx*
A data comparison is detected. The location of the error and the expected and read values are displayed.

`cannot download code`
A piece of code is downloaded to the MVME332XT dual ported memory, but the download was not successful.

`Board does not exist`
The host accesses the MVME332XT and a bus error occurs.

`board init timeout`
An initialization packet is sent to the MVME332XT, but no status is returned after an extended period of time.

`Controller never ready`
The MVME332XT indicates in its CSR space that the controller is never ready.

`create channel failed`
The buffer pipe communication channel between the MVME332XT and the host cannot be established. This could happen if 1) the controller is never ready, 2) the channel establishment protocol is never completed between the host and the MVME332XT, or 3) a bad status is reported by the MVME332XT.

`init packet failed`
The initialization packet is sent to the MVME332XT, but the initialization was not successful.

`board stay busy`
The MVME332XT CSR indicates that the board is staying busy. If the BUSY bit stays set, it will prevent the host from establishing a buffer pipe channel for host and MVME332XT communication.

`TAS never ready`
The MVME332XT CSR indicates that the board always has its test and set (TAS) bit set. If the TAS bit stays set, it will prevent the host from establishing a buffer pipe channel for host and MVME332XT communication.

`wait for VSTAT timeout`
The MVME332XT VSTAT bit in its CSR is not set during channel creation. This indicates that the MVME332XT is not able to complete the channel creation protocol due to board malfunction.

`channel bad status`
The channel creation was not successful. The channel is created so that the host and the MVME332XT can communicate through it.

MVME335 Asynchronous Communications Controller

The MVME335 Asynchronous Communications Controller tests allow for testing of both dual universal asynchronous receiver/transceivers (DUARTs) and the printer port.

The MVME335 Serial Port DUART tests 3 through 8 require loopback cables connected as:

```
SP3 <------> SP4
SP1 <------> SP2
```

See Figure 4-1 for the proper pinout for the loopback cables.

Some MVME335 Line Printer tests require a loopback jumper plug on the 36-Pin printer port connector. Figure 7-2 shows the printer port loopback connector pinout.

```
Pin #      to      Pin #
------------------------
  1     <------->    10
  2     <------->    13
  3     <------->    11
  4     <------->    32
  5     <------->    12
```


Figure 7-2. 36-Pin Printer Port Loopback Connector Pinout

To access the test menus for the first MVME335 board, type:

`SA:` **335.0**

A display similar to the following appears:

```
Current Menu is /bdtest/335.0 -"VME335 Quad SIO W/Printer Tests"
spd.1           - 335 Serial Port Duart
spd.0           - 335 Serial Port Duart
lp              - 335 Line Printer
SA: spd.0 <CR> (to access test selection menu for the first DUART connected to serial
               ports 1 & 2)
Selections for Test "335 Serial Port Duart"
(This menu is the same for both MVME335 port pairs.)
   0) Registers
   1) Internal-loop
   2) Baud Rate
   3) Parity Error      (Loopback cable required for tests 3 through 8)
   4) External-loop
   5) Rx intr
   6) Tx intr
   7) Interface intr
   8) Framing Error
SA: lp  (to access the MVME335 line printer test selection menu)
Selections for Test "335 Line Printer"
   0) Registers
   1) Printer loop
   2) Printer Intr
   o) Printer Output
SA:
```

The following describes the serial port DUART and line printer tests.

Serial Port DUART Tests

0) Registers

Writes and reads a pattern to verify the chip's bus interface.

`1) Internal-loop`
`4) External-loop`
These tests program the DUART to enter internal or external loopback mode. The tests then transmit, receive, and compare a pattern. The external-loop test requires a loopback cable.

`2) Baud rate`
Programs the DUART to enter internal loopback mode. The test then transmits, receives, and compares a pattern at all selectable baud rates, thereby verifying that all baud rates are working.

`3) Parity Error`
`8) Framing Error`
Check the ability of the DUART to detect parity errors and framing errors, respectively.

`5) Rx intr`
`6) Tx intr`
Verify receiver interrupts and transmitter interrupts, respectively.

`7) Interface intr`
Checks that a clear to send (CTS) status change generates an interrupt.

Line Printer Tests

`O) Registers`
Writes and reads a pattern to verify the chip's bus interface.

`1) Printer loop`
Sets bits in the printer port and looks for the associated jumpered status bit to follow.

`2) Printer Intr`
Verifies that the printer port can generate an interrupt when empty.

`o) Printer Output`
Sends 95 lines of a 130-column-wide pattern, then sends a form feed to a connected printer. This test requires a compatible, properly configured parallel printer.

MVME335 Board Test Configuration

The MVME335 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 335.0
/bdtest/335.0 configuration
   base [%0x100] = 0xffff3600 ?      # base address
   spd [0-2] = 2 ?                   # serial port DUART

> cf 335.0/spd.0
/bdtest/335.0/spd.0 configuration
   xfer size = 512 ?                 # buffer size
   duart [0-1] = 0 ?                 # dual universal async receiver/transceiver
```

MVME336 DeltaLINK Asynchronous Communications Controller

7

The MVME336 DeltaLINK product (also known as SIO/MUX or Serial Input Output/Multiplexer) consists of one MVME336 board or DeltaLINK hub (SIO) containing global memory, six XPCs (X.25 protocol controllers), and up to six DeltaLINK servers (MUX). Each server can support up to 16 RS-232 port connections with asynchronous communications.

The MVME336 diagnostics are divided into three parts: **gmem**, **xpc**, and **mx**. The **gmem** (global memory) tests verify that all of the hub's shared memory is functioning properly. The **xpc** tests verify that the XPCs DMA (direct memory access) paths are working. The **mx** tests verify that the 16 ports on each MUX server are functioning properly. Because the servers take commands from the hub (SIO), they can only be tested (using the **mx** tests) after the hub's global memory and XPCs are tested and found to be working.

You can test the MVME336 board without a MUX; however, the high speed interface link between the MVME336 and the MUX can only be tested if a MUX is connected.

The MVME336 DeltaLINK tests require loopback jumper plugs on all 16 of the 25-pin MUX connectors. Figure 7-3 shows the MUX Loopback connector pinout.

Pin #	to	Pin #
2	<-------->	3
4	<-------->	5
5	<-------->	6
8	<-------->	20

Figure 7-3. 25-Pin MUX Loopback Connector Pinout

To access the test menus for the MVME336 board, type:

`SA:` **336.0**

A display similar to the following appears:

```
Current Menu is /bdtest/336.0 - "VME336 XPC/Cluster Tests"

gmem      - 336SIO Global Memory Tests
mx.0       - 336MUX Tests          (to select the first 16-port MUX)
xpc.5      - 336SIO XPC Tests
xpc.4      - 336SIO XPC Tests
xpc.3      - 336SIO XPC Tests
xpc.2      - 336SIO XPC Tests
xpc.1      - 336SIO XPC Tests
xpc.0      - 336SIO XPC Tests (to select the first XPC connected to the first MUX)
```

The test selection menus for the `gmem`, `xpc`, and `mx.0` tests and a brief description of each test follow.

NOTE

If any of the `gmem` tests fail, the `xpc` and `mx.0` tests will also fail.

```
SA: gmem <CR>     (to access the test selection menu for the global memory tests)
Selections for Test "336SIO Global Memory Tests"
    0) Zeros
    1) Ones
    5) 5555s
    a) AAAAs
    d) Address
```


Global Memory Tests

The `gmem` selections test all the SIO board's global RAM, which is required for XPC command, status, and data buffers.

`0) Zeros`
`1) Ones`
`5) 5555s`
`a) AAAAs`

The RAM is written with a 0s, 1s, 5s, or As pattern and is then read and verified for correct contents.

`d) Address`

Writes all global RAM with a complementing address pattern, then verifies it.

```
SA: xpc.0 <CR>      (to access the test selection menu for the first XPC on the SIO board)
Selections for Test "336SIO XPC Tests"
   0) XPC DMA
```

7

XPC Tests

0) XPC DMA

Fills the SIO board's global RAM with a pattern and commands the SIO board's XPC to read and write it to a cleared buffer in global RAM. The destination is compared against the source buffer.

If this test fails, the Global RAM may be bad or the XPC interface to Global RAM may be failing.

```
SA: mx.0  (to access the MUX test selection menu)
Current Menu is /bdtest/336.0/mx.0 - "336MUX Tests"
p8.7           - 336MUX Ports 8-15 Tests
p8.6           - 336MUX Ports 8-15 Tests
p8.5           - 336MUX Ports 8-15 Tests
p8.4           - 336MUX Ports 8-15 Tests
p8.3           - 336MUX Ports 8-15 Tests
p8.2           - 336MUX Ports 8-15 Tests
p8.1           - 336MUX Ports 8-15 Tests
p8.0           - 336MUX Ports 8-15 Tests
p0.7           - 336MUX Ports 0-7 Tests
p0.6           - 336MUX Ports 0-7 Tests
p0.5           - 336MUX Ports 0-7 Tests
p0.4           - 336MUX Ports 0-7 Tests
p0.3           - 336MUX Ports 0-7 Tests
p0.2           - 336MUX Ports 0-7 Tests
p0.1           - 336MUX Ports 0-7 Tests
p0.0           - 336MUX Ports 0-7 Tests
SA: p0.0 (to access the first MVME336MUX port)
Selections for Test "336MUX Ports 0-7 Tests"
(The menu is the same for all 16 ports)
   c) Cluster Confidence
(Tests c, 0, and 1 do not require a loopback jumper plug)
   0) Internal Loopback
   1) Force Break
   2) External Loopback
(Tests 2, 3, 4, 5, o, 1, and e require a loopback jumper plug)
   3) Force Framing Error
   4) Modem Status Change
   5) Force Parity Error
   r) Display ROM Revision
   s) Display Modem Status
   o) String Output to Device (Press the BREAK key to stop test o, 1, or e)
   1) Endless Internal Loopback
   e) Endless External Loopback
```

7

MUX Tests

c) `Cluster Confidence`
Requests the results of the power-on self test performed by the MUX ROM. If the MUX ROM passes all self tests except the UART test, then the results of the UART test display. The results show which port failed by setting the failing port's bit in a byte. The results are returned for the first eight ports (if the requesting port is between 0 and 7) or for the second eight ports (if the port is between 8 and 15).

1) `Force Break`
3) `Force Framing Error`
5) `Force Parity Error`
These tests verify the detection logic for the forced condition.

0) `Internal Loopback`
2) `External Loopback`
l) `Endless Internal Loopback` *(Press the* BREAK *key to terminate)*
e) `Endless External Loopback` *(Press the* BREAK *key to terminate)*
These tests fill a buffer with a pattern and send it to the port. When all data sent is received, the send and receive buffers are compared.

4) `Modem Status Change`
Verifies that the CTS , DSR, and DCD signals can generate a change of status interrupt.

r) `Display ROM Revision`
Displays the revision level of the MUX ROM, e.g., 1.0.

s) `Display Modem Status`
Displays the level ON or OFF of the DSR, DCD, and RTS signals from the requested port.

o) `String Output to Device` *(Press the* BREAK *key to terminate)*
Sends a pattern to the port to verify output on a terminal. Like tests l and e, this is an endless test.

After a test is completed, the count of XPC retries from the MUX is requested. If the retry count is greater than zero, it displays. (An XPC retry is considered a soft or recoverable error condition.) An XPC retry may indicate that the condition of the high speed link between the SIO and MUX is marginal.

All MUX tests start the XPC link before sending any commands to the MUX. If the XPC cannot establish communications with the MUX after 90 seconds, a `link startup failed` or `TIMED OUT` error occurs. This may be caused by a bad link cable or bad SIO or MUX interface hardware.

MVME336 Board Test Configuration

The MVME336 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 336.0
/bdtest/336.0 configuration
   base [%0x40000] = 0xe00000            # base address
   level [1-7] = 3 ?                     # default interrupt level
   xpc [0-6] = 6 ?                       # number of X25 protocol controllers
   mx [0-6] = 1 ?                        # number of servers

> cf 336.0/xpc.0
/bdtest/336.0/xpc.0 configuration
   xpc [0-5] = 0 ?                       # X25 protocol controller

> cf 336.0/mx.0
/bdtest/336.0/mx.0 configuration
   xpc [0-5] = 0 ?                       # X25 protocol controller
   p0 [0-8] = 8 ?
   p8 [0-8] = 8 ?
```

7

MVME337 IO Engine with MVSB741 Asynchronous Communications Controller

The MVME337 I/O Engine supports from two to four MVSB741 Asynchronous Communications Controllers, depending on the size of the EVSB backplane. (A 2-slot and a 4-slot EVSB backplane are available.) There are two versions of the MVME337, the MVME337-1 (20.00 MHz clock with 1 Mb onboard memory) and the MVME337A (20.00 MHz clock with 4 Mb onboard.)

These tests check the MVME337 on-board memory and all port pairs on each MVSB741 communication controller installed. To run the MVSB741 tests, loopback cables are required; each MVSB741 has two octopus cables; the loopback arrangement for each octopus cable is as follows:

```
             "A" ports        "B" ports
             ---------        ---------
          Channel 0 <------> Channel 4
          Channel 1 <------> Channel 5
          Channel 2 <------> Channel 6
          Channel 3 <------> Channel 7
```

See Figure 4-1 for the proper pinout for the loopback cables.

To access the test menus for the first MVME337-1 I/O Engine, type:

SA: **337.0/evsb**

To access the test menus for the first MVME337A I/O Engine, type:

SA: **a337.0/evsb**

In either case, a display similar to the following appears:

```
Current Menu is /bdtest/337.0 - "VME337 Extensible VSB Tests"

evsb - VME337 Extensible VSB Tests

SA: evsb <CR>   (to access the test selection menu)

Selections for Test - "VME337 Extensible VSB Tests"

    0)  Run 337 Local Memory Test
    1)  Run evsb.0 Confidence Test
    2)  Run evsb.1 Confidence Test
    3)  Run evsb.2 Confidence Test
    4)  Run evsb.3 Confidence Test
    5)  Run BurnIn EVSB Confidence Test
    r)  Run User-specified EVSB Test
    c)  Display EVSB Configuration
    m)  Display EVSB Test Selection Menu
    D)  Display EVSB error log
    C)  Clear EVSB error log

(This menu is displayed for all MVME337-1 and MVME337A boards).

SA:
```

7

MVME337 Board Tests

Following is a description of each MVME337 test:

Run 337 Local Memory Test

This selection tests the MVME337 onboard memory using the *Burn-in Memory Test* described in the *Interactive Memory Tests* section in the *CPU, Memory, and Misc. Controller Board Tests* chapter.

Run evsb.*n* Confidence Test

These selections run a confidence test on all port pairs of the selected MVSB741 board (n = 0, 1, 2, 3). The confidence test includes MVSB741 tests 0) through i). These tests are listed below and described in the next section. Loopback cables are required.

Run Burn-in EVSB Confidence Test

This selection runs the confidence test described above on all port pairs of all MVSB741 boards in the EVSB backplane. The MVSB741 board tests are run concurrently. Loopback cables are required.

7

Run User-specified EVSB Test

This selection allows the user to specify a test or set of tests to be run on one or more of the MVSB741 boards installed in the EVSB backplane. Select the "Display EVSB Test Selection Menu" test to list the set of available tests. For example, to run the internal loopback test on the first MVSB741, enter:

741.0 0 (RETURN)

Display EVSB Configuration

This selection displays the current configuration of boards installed in the EVSB backplane. The test displays the EVSB slot number and type of EVSB board installed in that slot. If a slot is empty, `EMPTY` is displayed. This test probes for a total of six EVSB boards whether a 2-slot or 4-slot backplane is attached backplane to the MVME337. The additional slots past the end of the backplane are reported as `EMPTY`.

Display EVSB Test Selection Menu

This selection displays the Test Selection Menu for the MVSB741 board. The available tests are listed below and described in the next section.

0) Internal loopback
1) External Loopback A-B
2) External Loopback B-A
3) High baud rate A-B

4) High baud rate B-A
5) Low baud rate A-B
6) Low baud rate B-A
7) RTS Signal test A-B
8) RTS Signal test B-A
9) DTR Signal test A-B
a) DTR Signal test B-A
b) Parity err test A-B
c) Parity err test B-A
d) Frame error test A-B
e) Frame error test B-A
f) Overrun error A-B
g) Overrun error B-A
h) Send BREAK A-B
i) Send BREAK B-A
j) EEPROM test

Display EVSB Errors

This selection displays error messages from any failing EVSB test.

Clear EVSB Errors

This selection clears all the error messages from the EVSB error log.

MVSB741 Board Tests

The MVSB741 Board Tests are only available for selection through the MVME337 board test menu. (See the previous section.) Following is a description of each MVSB741 test:

`0) Internal loopback`

Run the internal loopback test on all ports; each port's TxD output is internally looped back to its RxD input. Loopback cables are not required.

`1) External Loopback A-B`
`2) External Loopback B-A`

Run external loopback test on all port pairs; the destination port echoes the received data back to the source port.

`3) High baud rate A-B`
`4) High baud rate B-A`

Run a transmit-receive test on all port pairs at 38.4 Kbaud.

5) Low baud rate A-B
6) Low baud rate B-A
Run a transmit-receive test on all port pairs at 110 baud.

7) RTS Signal test A-B
8) RTS Signal test B-A
RTS signal test.

9) DTR Signal test A-B
a) DTR Signal test B-A
DTR signal test.

b) Parity err test A-B
c) Parity err test B-A
Parity error test; transmit data with transmitter set for odd parity and receiver set for even parity; verify parity error occurred.

d) Frame error test A-B
e) Frame error test B-A
Framing error test; transmit 7-bit data with 1 stop bit; receive 8-bit data with 2 stop bits; verify framing error occurred.

f) Overrun error A-B
g) Overrun error B-A
Overrun error test; transmit data with receiver interrupt disabled; verify overrun error occurs.

h) Send BREAK A-B
i) Send BREAK B-A
Transmit BREAK character and verify BREAK detected on destination port.

j) EEPROM test
Non-destructive eeprom read test; read contents of EEPROM into two buffers and compare data.

MVME337-1 Board Test Configuration

The MVME337-1 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 337.0
/bdtest/337.0 configuration
    base[%0x100000].l+0xff00000=0xef300000# base address
    ram_addr.l=0xffe00000       # on-board ram start address
    control.w=0x0800                # multiprocessor control register offset
    address.w=0x0804                # multiprocessor address register offset
```

```
expert[0-1].b=0                    # expert mode flag
debug[0-1].b=0                     # debug mode flag
evsb.t
```

The *expert mode* and *debug mode* flags are used primarily for SSID test debug. These flags should remain off (set to zero) during normal SSID operation.

MVME337A Board Test Configuration

The MVME337A board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf a337.0
/bdtest/a337.0 configuration
   base[%0x400000].l+0xffc00000=0xedc00000  # base address
   ram_addr.l=0xff800000              # on-board ram start address
   control.w=0x0800                   # multiprocessor control register offset
   address.w=0x0804                   # multiprocessor address register offset
   expert[0-1].b=0                    # expert mode flag
   debug[0-1].b=0                     # debug mode flag
   evsb.t
```

The *expert mode* and *debug mode* flags are used primarily for SSID test debug. These flags should remain off (set to zero) during normal SSID operation.

MVME337 Error Messages

MVME337 did not start
An attempt was made to download the MVME337 board and start the download code running on the board, but the board did not start. Check that the front panel switches are set properly.

MVME337 failed to start command
A command was sent to the MVME337 for execution, but was never executed.

MVME337: Board is not running
A command was sent to the MVME337, but the board is not running, and cannot execute the command.

MVME337 TIMEOUT: command did not finish
The command sent to the MVME337 for execution did not finish in the allotted amount of time.

MVME337: BREAK detected
The MVME337 detected a BREAK condition.

MVME337: command failed
The command sent to the MVME337 failed.

MVSB741 Error Messages

The MVSB741 error messages adhere to the following format, except where noted:

741.*n*/sp: (*x*) Error message on port *j* (chip *k*)

where **741.*n*** (*n* = 0, 1, 2, 3) is the MVSB741 board under test, (*x*) is the test number from the EVSB Test Selection Menu, port *j* is the failing channel on the chip (j = 0, 1, 2, 3, 4, 5, 6, 7; k = 0, 1), and **Error message** is one of the following:

Receiver TIMEOUT
Indicates the receive FIFO is empty and no data has been received within the receive timeout period.

BREAK detected
Indicates that a BREAK has been detected.

PARITY ERROR detected
Indicates that a parity error has occurred.

FRAMING ERROR detected
Indicates that a bad stop bit has been detected.

OVERRUN detected
Indicates that new data has arrived but the FIFO and holding registers are full. The new data is lost.

Data mis-compare
Indicates that the data received does not match the data transmitted.

No CTS status
In the "RTS Signal" test, the receiving channel was set up to receive CTS, but it was not detected.

Unexpected CTS status
In the "RTS Signal" test, CTS was unexpectedly detected on the receiving channel.

`RTS no 1 to 0 transition`
`RTS no 0 to 1 transition`
: In the "RTS Signal" test, the expected state changes did not occur.

`CTS input unexpectedly changed`
: In the "RTS Signal" test, unexpected CTS status was detected after a modem signal interrupt has been serviced.

`No DSR status`
: In the "DTR Signal" test, the receiving channel was set up to receive DSR, but it was not detected.

`Unexpected DSR status`
: In the "DTR Signal" test, DSR was unexpectedly detected on the receiving channel.

`DSR no 1 to 0 transition`
`DSR no 0 to 1 transition`
: In the "DTR Signal" test, the expected state changes did not occur.

`DSR input unexpectedly changed`
: In the "DTR Signal" test, unexpected DSR status was detected after a modem signal interrupt has been serviced.

`No BREAK detected`
: In the "Send BREAK" test, the BREAK character was transmitted, but was not detected on the destination port.

`No FRAMING ERROR detected`
: In the "Frame error test", no framing error was detected, even though one should have been generated.

`No OVERRUN detected`
: In the "Overrun error" test, no overrun error was detected, even though receiver interrupts were disabled, and an overrun condition should have occurred.

`No PARITY ERROR detected`
: In the "Parity Error" test, no parity error was detected, even though the transmitter's parity did not match the receiver's parity.

Additional error messages:

`CD180 chip` *n* `initialization failed`
: The MVSB741's CD180 chip failed to initialize.

`Channel configuration failed on port` *j*
: A port's channel parameters could not be successfully changed.

MVME374 Multi-Protocol Ethernet Interface Module

In the following information about the MVME374, the word *host* refers to the system's main processor (e.g., MVME141, MVME147).

When running SSID tests, the MVME374 is taken over by the host processor via the following procedure:

1. The MVME374 tests are copied into the dual ported RAM.
2. The host loads a vector pointing to the tests in the MVME374 vector table.
3. The host loads a flag in the MVME374 control packet (CP).
4. The host interrupts the MVME374 by writing into the MVME374 interrupt RAM (IRAM).
5. When the test flag clears in the control packet, the MVME374 is ready for tests.

To access the test menus for the first MVME374 board, type:

```
SA: 374.0
```

A display similar to the following appears:

```
Current Menu is /bdtest/374.0 - "VME374 Lan Controller Tests"

lan - VME374 Lan Tests

SA: lan <CR>   (to access the test selection menu)

Selections for Test - "VME374 Lan Tests"

 1) Run a,b,c,d,e,f,g,h,i,j,k,l
 2) Run a,b,c,d,e,f,g,h,i,j,k,l,x
 a) Checksum test NVRAM
 b) Check station address
 c) Lance register
 d) Lance internal loopback
 e) On-board walking bit RAM
 f) On-board random RAM
 g) Off-board walking bit RAM
 h) Off-board random RAM
 i) On-board 374 down address RAM
 j) On-board 374 up address RAM
 k) Bus arbitration
 l) IRQ
 r) Reset 374
 x) Lance external loopback  (Connect MVME374 to Ethernet coax
                             via a transceiver/tap.)
SA:
```

7

The following describes the MVME374 tests:

1) **Run a,b,c,d,e,f,g,h,i,j,k,l**
Execute tests a through l. For tests descriptions, see individual descriptions.

2) **Run a,b,c,d,e,f,g,h,i,j,k,l,x**
Execute tests a through l and x. For tests descriptions, see individual descriptions.

a) `Check sum test NVRAM`
Performs checksum test of the MVME374 Non-Volatile Random Access Memory (NVRAM). This verifies that the NVRAM data has not been corrupted.

b) `Check station address`
Verifies the Ethernet station address. If the station address prefix does not match the Motorola 28 bit prefix (0x08003E2), this test displays a warning. This warning is not an error.

c) `Lance register`
A series of patterns are written to the LANCE registers. The patterns are then verified.

d) `Lance internal loopback`
The LANCE is programmed to be in its internal loopback mode. A series of special Ethernet packets (technically referred to as runt packets) are put into the LANCE transmit queue. The LANCE transmits the packets and loops them back to its receiver. The LANCE then deposits the packets into the receive queue where the program verifies the packets.

e) `On-board walking bit RAM`
Performs a walking bit RAM test on the MVME374. This test is done by the cross-loaded program on the MVME374. A walking bit test consists of the following steps:

1. A 1 is written into a 32-bit memory location.
2. The contents of the location is then verified.
3. The last value is shifted to the left one place.
4. This new value is written into the same 32 bit memory location.
5. It is then verified. If it is good, go to Step 3. This is repeated until the 32 bit value is equal to zero.
6. Move to next address and go to Step 1.
7. Continue until last address location is tested.

f) `On-board random RAM`
Performs a random value 16 bit RAM test on the MVME374. This test is done by the cross-loaded program on the MVME374. The program writes a random series of words to the range of RAM being tested. The program then verifies the random series. The *range of RAM being tested* is determined by the following criteria:

1. The starting address for the test is just past the end of the cross-loaded test module. This address can vary between test versions since the size of the cross-loaded module can change.

2. The ending address for the test is 0xFFFFFF00.

These are local onboard MVME374 RAM addresses.

g) `Off-board walking bit RAM`
Performs basically the same test as the On-board walking bit RAM test. The difference is that the memory being tested by MVME374 is the system's main memory.

h) `Off-board random RAM`
Performs basically the same test as the On-board random RAM test. The difference is that the memory being tested by MVME374 is the system's main memory.

i) `On-board 374 down address RAM`
Performs a down address RAM test on the MVME374. This test is done by the cross-loaded program on the MVME374. The program starts testing at the highest address to be tested and writes the address of that location into that location. The program proceeds down to the starting address. The program then verifies the pattern.

j) `On-board 374 up address RAM`
Performs an up address RAM test on the MVME374. This test is done by the cross-loaded program on the MVME374. The program starts testing at the lowest address to be tested and writes the address of that location into that location. The program proceeds up to the ending address. The program then verifies the pattern.

k) `Bus arbitration`
This test checks the bus arbitration logic on the MVME374 by executing the Off-board random RAM test from the MVME374 and, at the same time, executing an incremental pattern to the MVME374 onboard memory from the host system.

`l) IRQ`

During this test, the MVME374 generates a VMEbus Interrupt ReQuest (IRQ) at the level set in the configuration structure.

`r) Reset 374`

This test accesses the reset bit in the MVME374 CSR (Command Status Register) to reset the board.

NOTE

Some systems detect the system fail status if this test is executed. The MVME374 momentarily asserts the system fail line during a board reset.

`x) Lance external loopback`

NOTE

An external cable, transceiver, and transceiver cable are required to execute this test.

The LANCE is programmed to be in normal transmit/receive mode. A series of special Ethernet packets (technically referred to as runt packets) are put into the LANCE transmit queue. The LANCE transmits the packets to the cable, then receives them from the cable back into its receiver. The LANCE then deposits the packets into the receive queue where the program verifies the packets.

MVME374 Board Test Configuration

The MVME374 board test configuration is shown below. Refer to the **cf** command in Appendix A for information on modifying this configuration.

```
> cf 374.0
/bdtest/374.0 configuration
   base [%0x100000] = 0xff000000 ?                      # base address
   signal_address = 0xff0fffc0 ?
   signal_type = 0x0 ?
   am[0x0d,0x3d] = 0x0d ?                               # address modifier
   level[0-7] = 4 ?                                     # default interrupt level
   ExtLbck SQE Errors: Hard/Soft [0,1] = 0 ?            # default SQE errors to hard
```

MVME374 Error Messages

The following describes the MVME374 Error Messages:

`Board does not exist`
The board is not being detected by the program.

`t374init() getmem() failed`
The program tried to request some memory from the SSID main memory pool and there was not enough memory available to meet the request.

`Board failed to start.`
The cross-loaded program failed the startup handshake. Some of the possible causes are:

1. The MVME374 has a malfunction in its bus interface.
2. The MVME374 onboard memory is operating incorrectly.
3. The MVME374 has some other internal malfunction that prevents the program from operating correctly.

`Board fail LED is on.`
The board did not clear its board fail bit after a reset command. This indicates a defective board.

`Timeout on board reset.`
The board did not return status after a reset command. This indicates a defective board.

`374.0/lan: getmem(0x800) failed`
The program tried to request some memory from SSID main memory pool and there was not enough memory available to meet the request.

```
Bus arbitration test error.
374.0/lan: Random word RAM test error:
At address 0x001e0010 expected 0x0000e001, read 0x00000001
```

The MVME374 side of the bus arbitration test detected a memory miscompare. The MVME374 is testing the system's main memory during this test. This indicates one of the following is happening:

1. The system controller, which controls who owns the bus, has allowed an illegal bus cycle to occur.
2. The MVME374 has illegally taken control of the bus causing an illegal bus cycle to occur.
3. The MVME374 has a bad master bus interface causing incorrect data to be written or read.
4. The system's main memory is responding incorrectly to the MVME374.
5. Another board in the system is illegally taking control of the bus causing an illegal bus cycle to occur.
6. Another program or board in the system is illegally using the same memory as the MVME374 is testing.

```
374.0/lan: bus arbitration test error
At address 0xff0e0010 expected 0x0a, read 0x02
```

The host side of the bus arbitration test detected a memory miscompare. This miscompare is in the MVME374 memory. One of the following is happening:

1. The MVME374 has allowed an internal bus cycle and a external bus cycle coming into the board to happen at the same time.
2. The MVME374 memory has responded incorrectly.
3. The program executing on the MVME374 is illegally using the same memory on the MVME374 as the host test is testing.
4. Another program or board in the system is illegally using the same memory on the MVME374 as the host is testing.

```
IRQ test: IRQ setting is zero or greater than 7, test can not be run
```

The interrupt test detected that the configuration value for the IRQ level is not set to a valid value. This is only a warning. The test is skipped as long as the setting is invalid. This warning should only display if the **cf** command was used to change the IRQ level for the board.

7

374.0/lan: Walking bit RAM test error:
At address 0xfffe0010 expected 0x00100000, read 0x00000001

The cross-loaded test program has detected a data miscompare. If the address is between 0xFFF05000 to 0xFFFFFF00 it is on the MVME374. If the address is below that number range, it is in the system's main memory.

374.0/lan: Random word RAM test error:
At address 0xfffe0010 expected 0x0a00e001, read 0x00000001

The cross-loaded test program has detected a data miscompare. If the address is between 0xFFF05000 to 0xFFFFFF00 it is on the MVME374. If the address is below that number range, it is in the system's main memory.

374.0/lan: Up address RAM test error:
At address 0xfffe0010 expected 0x00100000, read 0x00000001

The cross-loaded test program has detected a data miscompare in the MVME374 memory.

374.0/lan: Down address RAM test error:
At address 0xfffe0010 expected 0x000000f1, read 0x00000001

The cross-loaded test program has detected a data miscompare in the MVME374 memory.

374.0/lan: NVRAM test error:
NVRAM is bad or not initialized

The NVRAM does not contain the correct values.

374.0/lan: Lance register test error:
incorrect data received error

The LANCE or the interface to the LANCE on the MVME374 is probably defective.

374.0/lan: Lance register test error:
Lance status error occurred status = 0x31

The LANCE on the MVME374 is probably defective. If the users are interested in the meaning of the status, refer to the LANCE manual referenced in the *MVME374 Multi-Protocol Ethernet Interface Module User's Manual* to determine the meaning of the status. The status displayed is from the LANCE status register.

374.0/lan: Lance register test error:
error code=0x3

The cross-load program timed out waiting on LANCE. The LANCE on the MVME374 is probably defective.

374.0/lan: Lance internal loopback test error:
incorrect data received error

The LANCE on the MVME374 is probably defective.

374.0/lan: Lance internal loopback test error:
Lance status error occurred status = 0x31

The LANCE on the MVME374 is probably defective.

374.0/lan: Lance internal loopback test error:
error code=0x3

The cross-load program timed out waiting on LANCE. The LANCE on the MVME374 is probably defective.

374.0/lan: Lance external loopback test error:
incorrect data received error

The following are possible causes:

1. The external cable, transceiver, or transceiver cable have a defective connection path, or they are defective themselves.
2. The LANCE or the interface logic on the MVME374 is defective.

374.0/lan: Lance external loopback test error:
Lance status error occurred status = 0x31

The following are possible causes:

1. The external cable, transceiver, or transceiver cable have a defective connection path, or they are defective themselves.
2. The LANCE or the interface logic on the MVME374 is defective.

374.0/lan: Lance external loopback test error:
error code=0x3

The cross-load program timed out waiting on LANCE. The LANCE on the MVME374 is probably defective. Another possible reason for the error is the Ethernet has a extremely heavy network load causing collisions.

374.0/lan: Lance external loopback test error:
Lance status error occurred status = 0xa6f3

A Collision Error occurred and it indicates that the collision input to the Lance failed to activate after a Lance-initiated transmission was completed. The probable cause is the SQE signal from the Ethernet transceiver has been disabled, is not functioning, or is bad.

374.0/lan: Lance external loopback SQE soft Errors = 129

When the change SQE errors flag is set, SQE errors are changed to soft errors and the number of soft errors that occurred during the test are reported by this error message.

374.0/lan: Ethernet station address test error:
WARNING Board has non-standard ethernet address

This is only a warning message. The address does not have a Motorola prefix.

374.0/lan: Ethernet station address test error:
NVRAM field has checksum error

The Ethernet address contained in the NVRAM is not verifiable.

MVME374 Other Error Messages

Any one of the following displays could occur for any of the above tests.

level seven IRQ occurred
on board RAM parity error occurred

The MVME374 on board parity logic detected wrong parity on a memory access.

level seven IRQ occurred
vme bus error occurred

The MVME374 got a bus time out when it accessed the VMEbus.

level seven IRQ occurred
local bus timeout occurred

The MVME374 got a bus time out while it was accessing one of its local on-board devices.

Command Timeout

The program being executed did not complete in the allotted time. The MVME374 has some malfunction that prevents the program from completing correctly.

MVME376 Ethernet Controller

To access the test menus for the first MVME376 board, type:

`SA:` **376.0**

A display similar to the following appears:

```
Current Menu is /bdtest/376.0 - "VME376 Board Tests"
lan376              - VME376 Tests
SA:      lan376 <CR> (to access the test selection menu)
Selections for Test "VME376 Tests"
   1) Run tests a,b,c,d,e,f,g,i
   2) Run tests a,b,c,d,e,f,g,h,i
   a) Memory test - Addressing
   b) Memory test - Long Access
   c) Memory test - Word Access
   d) Memory test - Byte Access
   e) Memory test - Walking 1's
   f) Memory test - Walking 0's
   g) LANCE Internal Loopback
   h) LANCE External Loopback
   i) Non-Volatile RAM
   j) Hardware Reset
   k) Show Hard & Soft Addresses
   n) Display Board Ethernet Address
   r) Display Snapshot Registers
   D) Display Error Log
   C) Clear Error Log
SA:
```

7

The following describes the MVME376 tests:

1) `Run tests a,b,c,d,e,f,g,i`
   Execute tests a through g, and i. The external loopback test is not included in this menu selection. This menu selection is what is used for confidence and fault testing. It does not require any loopback or external hardware. For tests descriptions, see individual test descriptions.

2) `Run tests a,b,c,d,e,f,g,h,i`
   Execute tests a through i. **DO NOT EXECUTE ON A LIVE NETWORK,** as external loopback testing is performed. For further detail, see individual descriptions.

a) `Memory test - Addressing`
   Performs address test to verify the integrity of the address lines on the VME376.

b) `Memory test - Long Access`
Performs memory test with a data width of 32 bits. The test is done by writing a pattern, reading, and verifying the pattern until all the on-board memory is tested. The test is then repeated with the opposite bit pattern.

c) `Memory test - Word Access`
Performs memory test with a data width of 16 bits. The test is done by writing a pattern, reading, and verifying the pattern until all the on-board memory is tested. The test is then repeated with the opposite bit pattern.

d) `Memory test - Byte Access`
Performs memory test with a data width of 32 bits. The test is done by writing a pattern, reading, and verifying the pattern until all the on-board memory is tested. The test is then repeated with the opposite bit pattern.

e) `Memory test - Walking 1's`
Performs a walking bit test with a data width of 16 bits. The test is done by writing a pattern, reading, and verifying the pattern until all the on-board memory is tested.

f) `Memory test - Walking 0's`
Performs a walking bit test with a data width of 16 bits. The test is done by writing a pattern, reading, and verifying the pattern until all the on-board memory is tested.

g) `LANCE Internal Loopback`
The LANCE is programmed to be in internal loopback mode and tests are run to verify data path and integrity of the chip. No testing is done in which data is transmitted off board.

h) `LANCE External Loopback`
The LANCE is programmed to be in external loopback mode and tests are run to verify data path and integrity of the chip and serial interface adapter. **DO NOT RUN THIS TEST ON AN ACTIVE NETWORK!** Packets of 32-byte length are transmitted out to the attached network. The destination address of these packets is always 000801205a25. To run this test, a properly terminated transceiver must be attached to each VME376 under test. Although any properly terminated, private network will suffice, the following equipment list is provided for the user's benefit:

Transceiver Cable
Thinnet Transceiver
Tee Connector for Thinnet Cable
Two Network Cable Terminators

It is important that *each* MVME376 under test have its own private network. No other active Ethernet controllers should be attached to the private test network.

i) **Non-Volatile RAM**
Non-volatile RAM test checks the Manufacturer's ID portion of the Ethernet node address which was written to NVRAM at the factory. If the ID portion is different from what was stored, the test will fail.

j) **Hardware Reset**
Performs a board reset and verifies the static state of registers and other responding logic.

k) **Show Hard & Soft Addresses**
Displays board addresses and software addresses, as follows:

```
VME376 Short I/O address....  0xffff1200
VME376 Memory Start           0xfd6c0000
VME376 Memory End...........  0xfd6fffff
VME376 Init Block addr        0xfd6fec10
TX MSG DESC BUFR............  0xfd6c0000
RX MSG DESC BUFR              0xfd6c0200
TX BUFR ADDR................  0xfd6c0400
RX BUFR ADDR                  0xfd6df800
Board Control Register......  0x006c
Current CSR0                  0x0004
Interrupt Vector............  0x2
Interrupt Level               0x3
```

n) **Display Ethernet Address**
Selection displays the board's Ethernet Address which is stored in non-volatile RAM.

r) **Display Snapshot Registers**
Displays contents of several internal board registers. A snapshot of these registers is taken during Internal and External Loopback testing. The register values shown are from the most recently taken snapshot. The format of the display is :

```
Register Values of Last Snapshot:

CSR0 ...................... 0x06f3
RMD1 FLAGS                  0x03
TMD1 FLAGS ................ 0x03
TMD3 FLAGS                  0x0000
BOARD Status Register ..... 0x000f
```

D) `Display Error Log`
An error log has been provided, which stores all error information for the VME376 tests. Each VME376 board under test will have its own Error Log. All errors will print to the screen at the time of the error, and will also be stored in the Error Log. Although the error may have scrolled off the screen, the failure information can be viewed through the use of the Display option. The only errors which cannot be logged are failures of the error logger itself, and system resource allocation errors which might occur very early in the test execution. In some cases, more information is logged than is printed to the screen at the time of the error. The contents of the Error Log should be checked for any VME376 failure.

C) `Clear Error Log`
This command will clear the errors logged to the local (board-specific) log. The system error log is not cleared by this command. In order to clear the system error log, the user must execute the **clear** system level command. There are two methods for clearing the local Error Log, as the **clear** command also provides the option of clearing this local error log.

MVME376 Board Test Configuration

The MVME376 board test configuration is shown below. Refer to the **cf** command in Appendix A of this guide for information on modifying this configuration.

> **cf 376.0**

```
/bdtest/376.0 configuration
   base[%0x200].1+0x200=0xffff1200          # base address
   level[0-7]=3                             # default interrupt level
   hwkphysaddr[%0x100].1+0xfffc0000=0xfd6c0000    # shared memory
```

>cf 376.0/lan376

```
/bdtest/376.0/lan376 configuration
   progress [0-1] = 0     # if set, show progress printouts
   intr_en [0-1] = 1      # if set, VME376 interrupts are enabled
```

MVME376 Error Messages

The following describes the MVME376 Error Messages:

LANCE Internal/External Loopback Errors

`LANCE_INITIALIZATION_FAILURE (0x001)`
Indicates that the LANCE did not read the initialization block and the IDON flag was not issued.

`TRANSMIT_TIMEOUT_ERROR (0x003)`
Indicates that the transmitter sent a packet and did not receive an interrupt.

7

`CRC_COMPARE_FAILURE (0x004)`
Indicates that the transmitted and received CRC were not identical.

`DATA RECEIVED DOES NOT MATCH DATA TRANSMITTED (0x005)`
Indicates that a comparison of the transmit buffer to receive buffer shows their contents are not identical. Additional information is written to the error log in the following format:

```
ERROR:0x00000005, Data received does not match data
transmitted
     At transmit buff addr:   0xaaaaaaaa   Data:   0xdd
     At receive buff addr:    0xaaaaaaaa   Data:   0xdd
```

`NO_LANCE_COMPLETION_INTERRUPT (0x006)`
Indicates that the LANCE did not receive an expected interrupt. This error will only occur when running SSID with interrupts enabled.

`NO_INIT_DONE_INTERRUPT (0x009)`
Indicates the interrupt was not received after the LANCE chip was initialized.

`POLLED_COMPLETION_TIMEOUT (0x00C)`
Indicates the Control/Status Register 0 never indicated task completion during internal or external loopback testing. This error will only occur when the test is running with interrupts disabled.

`TRANSMIT_PACKET_LOST (0x00E)`
Indicates the Control/Status Register 0 never indicated transmission of the packet during internal or external loopback testing. This error will only occur

when the test is running with interrupts disabled.

`RECEIVE_PACKET_LOST (0x010)`
Indicates the Control/Status Register 0 never indicated reception of the packet during internal or external loopback testing. This error will only occur when the test is running with interrupts disabled.

VME376 Resident Memory Errors

`Data Read does not match Data Written. (0x101)`
Indicates that the data which was read did not match what had been written. Additional detail regarding the failure will be written into the Error Log in this format:

```
Data Read does not match Data Written.
    At address      0xaaaaaaaa
    Expected data:  0xnn
    Actual Data:    0xnn
```

(The width of the expected and actual data field will depend on the access width of the test; either byte, word, or long.)

`BUS ERROR DURING VME376 MEMORY TEST (0x105)`
Indicates that a bus error occurred during one of the VME376 memory tests.

`PARITY_ERROR_BANK3 (0x119)`
Onboard memory parity error was detected on bank 3.

`PARITY_ERROR_BANK2 (0x11A)`
Onboard memory parity error was detected on bank 2.

`PARITY_ERROR_BANK1 (0x11B)`
Onboard memory parity error was detected on bank 1.

`PARITY_ERROR_BANK0 (0x11C)`
Onboard memory parity error was detected on bank 0.

Receive Descriptor One Errors

Receive Descriptor One is a control and status register within the LANCE chip. The error conditions shown below can be reported through this register.

`RECEIVE_FRAMING_ERROR (0x400)`
Indicates that the incoming packet contained a noninteger multiple of eight bits and there was a CRC error. Only valid on external loopback.

RECEIVE_OVERFLOW_ERROR (0x401)
Indicates that the receiver has lost all or part of the incoming packet due to an inability to store the packet in a memory buffer before the internal SILO overflowed.

RECEIVE_CRC_ERROR (0x402)
Indicates that the receiver has detected a CRC error on the incoming packet.

RECEIVE_BUFFER_ERROR (0x403)
Indicates that the LANCE does not own the next buffer while data chaining a received packet. This can occur in either of two ways: 1) the OWN bit of the next buffer is zero, or 2) SILO overflow occurred before the LANCE received the next STATUS.

Transmit Descriptor One Errors

Transmit Descriptor One is a control and status register within the LANCE chip. The error conditions shown below can be reported through this register.

TRANSMIT_MULTIPLE_RETRY (0x510)
Indicates that more than one retry was needed to transmit a packet.

TRANSMIT_SINGLE_RETRY (0x511)
Indicates that only one retry was needed to transmit a packet.

TRANSMIT_PACKET_DEFFERED (0x512)
Indicates that the LANCE had to defer while trying to transmit a packet. This condition occurs only if the channel is busy when the LANCE is ready to transmit.

Transmit Descriptor Three Errors

Transmit Descriptor Three is a control and status register within the LANCE chip. The error conditions shown below can be reported through this register.

TRANSMIT_BUFFER_ERROR (0x531)
Set by the LANCE during transmission when the LANCE does not find the ENP flag in the current buffer and does not own the next buffer.

TRANSMIT_UNDERFLOW_ERROR (0x532)
Indicates that the transmitter has truncated a message due to data late from memory.

`TRANSMIT_LATE_COLLISION (0x533)`
Indicates that a collision has occurred after the slot time of the channel has elapsed.

`TRANSMIT_LOSS_OF_CARRIER (0x534)`
Indicates that the carrier input to the LANCE went false during a LANCE initiated transmission.

`TRANSMIT_EXCESSIVE_RETRY (0x535)`
Indicates that the transmitter has failed in 16 attempts to successfully transmit a message due to repeated collisions on the medium.

`TIME_DOMAIN_REFLECTOMETRY (0x536)`
Indicates that the LANCE detected an error of the network cable when it performed a Time Domain Reflectometry (TDR) test. The VME376 will automatically perform a TDR test when 16 attempts to transmit a message fail due to collisions on the medium. One cause of this error is improper or missing installation of the transceiver, transceiver cable, Tee, or terminators, all of which are required for external loopback testing.

Control and Status Register Zero - Board General Errors

Control and Status Register Zero resides within the LANCE chip. The error conditions shown below can be reported through this register.

`BABBLING_TRANSMITTER (0x600)`
Indicates that the transmitter has been on the channel longer than the time required to send the maximum length packet.

`COLLISION_DETECTED (0x601)`
Indicates that the collision input to the LANCE failed to activate within 2 μs after a LANCE-initiated transmission was completed.

`RECEIVER_MISSED_PACKET (0x602)`
Indicates the receiver lost a packet because it does not own any receive buffers.

`MEMORY_ACKNOWLEDGE_ERROR (0x603)`
Indicates the LANCE is the Bus Master and has not received READY within 26.6 μs after asserting the address on the DAL lines.

`BERR_DURING_SOFT_RESET (0x605)`
A Bus Error occured when SSID attempted to write to VME376 board control registers.

Miscellaneous Errors

`Manufacturer's ID incorrect (0x700)`

Indicates that the Ethernet node address does not match the one programmed at the factory. Additional detail regarding the failure will be written into the Error Log in this format:

```
Expected Ethernet Address:  0x000077aaaaaa
Actual Address in NVRAM:    0xaaaaaaaaaaaa
```

`THE ERROR BIT FLAGS ARE INVALID. (0x701)`

Indicates that SSID found invalid error bit flags while attempting to report an error to the user.

`Received an unexpected interrupt.. (0x702)`

An interrupt was received on the VME376 interrupt vector which was not expected.

7

A

APPENDIX A
COMMAND LINE MODE

Command line mode is the testing environment on which the menu system is based. Designed for advanced users, command line mode offers many features not available within the menu system. In command line mode you can set test options directly on the command line, run a sequence of tests from different test menus, and configure the boards in the system.

To enter command line mode, type:

```
SA: cmdline
>
```

The > prompt indicates that you are in command line mode.

To run any of the SSID tests, simply type the name of the test and the test identifier(s). Typing the test name with no test identifier causes a preselected series of nondestructive tests to be run. To find out which test options are available for a given test, type the name of the test followed by a minus (–) sign.

EXAMPLES:

> **smd.1 0123** *(Run tests* **0, 1, 2,** *and* **3** *on the second SMD disk drive.)*

> **131.0** *(Run all preselected tests on the first MVME131 controller.)*

> **esdi.0 –** *(List the test options available for the first MVME323 disk drive.)*

To return to menu mode, type:

```
> set menu
SA:
```

Command Line Format

Command line commands have the following format:

cmd [*args*] [<*input_dev*] [[*descriptor*]>*output_dev*] [*descriptor*>*descriptor*] [; *cmd* . . .]

- Multiple command delimiter
- Duplicate standard output descriptor
- Redirect standard output descriptor to *output_dev*
- Redirect standard input descriptor to get data from *input_dev*
- ARGUMENT(S), separated by spaces or tabs; passed to *cmd* for optional selection
- COMMAND, can be any executable node name found in the output of the **he** command

where:

[ ]	= optional field
. . .	= repeatable field
<	= input redirection
>	= output redirection
input_dev	= input device such as **sp0** (console port) or **log** (logger)
output_dev	= output device such as **pr** (printer) or **log** (logger)
descriptor	= device descriptor

Standard device descriptors:

0	= input descriptor (default for input redirection)
1	= output descriptor (default for output redirection)
2	= error output descriptor

After receiving a user's command, the command line monitor evaluates its format, opens the I/O devices if needed for redirection, builds a list of arguments, updates the system status, traverses through the directory tree if the command field is a directory node, and finally gives control to all test nodes under the command tree.

I/O Redirection

I/O redirection is a mechanism that allows the user to specify input and output devices on the command line. In a typical computer system, input comes from the terminal keyboard and output goes to the terminal screen. I/O redirection allows the user to change the source of input or the destination for output by specifying a new source and/or destination on the command line.

In SSID, output is sent to a logical device called standard output or error output, both of which default to the user's terminal screen. Typical output from a program is sent to standard output, while error messages are sent to error output. All input is taken from a logical device called standard input, which defaults to the user's terminal keyboard.

Standard output and error output can be redirected to another terminal, a printer attached to the MVME050 board, if installed, or the log. Error output can be sent independent of standard output or to the same destination. Standard input can be redirected from another terminal or the log. At the present time all tests may not adhere to the error output path for errors but instead send errors to standard output. A future release of SSID will correct this problem.

EXAMPLES:

1) Redirect standard output from the **he** command to the line printer:

   he > pr

2) Redirect standard input to come from the log:

   echo < log

3) Recalibrate the first winchester disk (test 1) and redirect the error output to the log:

   hd.0 1 2 > log

4) Set loop continuous, run the default tests on the first winchester disk, and store the error output in the log:

   set lc; hd.0 2 > log *(Note the semicolon separating the commands)*

5) Run the default tests on the first MVME332XT and redirect both standard output and error output to the printer:

   x332.0 > pr 2 > 1

The first part of the command (**x332.0** > **pr**) redirects the standard output to the printer. The second part (**2** > **1**) redirects error output (descriptor 2) to be the same destination as the standard output (descriptor 1), which is redirected to the printer in the first part of the command.

NOTE

If the MVME050 attached printer is under test, standard output and error output of the test cannot be redirected to the printer because they can interfere with the printer test. The printer test requires a special loopback cable connected to the printer port, but for test results to be printed out, connection to a printer is required.

All references to the device **pr** refer to a printer connected via an MVME050 board.

Command Structures

To determine the tests available for a particular device, use the **he** (help) command, which displays the command directory tree. The tree structure allows you to concentrate on a particular test suite once the directory has been identified. Typing **he** *board_number* allows you to view the test suites associated with that board. Refer to the first two examples in the **he** command description later in this chapter.

When an executable node name, whether directory or test, is invoked as a command, every subnode under its tree structure is also a part of the command invocation. Thus, typing the following causes both **hd.0** and **fd.0** to be executed simultaneously or sequentially depending on how the system's environment is set up. Command line arguments are also passed to each subnode for as many levels as are constructed in the tree.

320.0

Arguments passed to the test module may allow specific subtest selection from the test module. The **hd.0** and **fd.0** subtest menus are obtained by using the following command:

he hd.0 fd.0

A display similar to the following appears:

```
bdtest/320.0/hd.0                      t--x     Winchester Disk
     0) Recalibrate
     1) Quick confidence read
     2) Read entire OS disk
     3) Random read OS disk
     4) Pingpong read OS disk
     d) Display OS bad tracks
bdtest/320.0/fd.0                      t--x     Floppy Disk
     0) Recalibrate
     1) Quick confidence read
     2) Read entire OS disk
     3) Random read disk
     4) Pingpong read disk
     p) Partial W/R DESTRUCTIVE
     w) Fully W/R DESTRUCTIVE
     f) Format Disk
```

The numbered subtest selections are the default test sequences that run if no test selection is specified. (This convention applies only when there are less than 11 default selections, i.e., 0 through 9.) For example, tests 0 through 4 run if command **hd.0** or **320.0** is invoked, but only test 2 runs if the command is **hd.0 2**. You can only invoke a subtest selection preceded by a letter by explicitly passing the selection letter as an argument to the test module from the command line. Test selections can be in any order and grouped into one or more arguments. You can enter multiple selections and repeat a selection any number of times. Each selection is executed in the order it appears on the command line.

EXAMPLES:

1) Run all default tests (0 through 4) on the first hard disk:

   hd.0 or **320.0**

2) Run several hard disk tests and several floppy disk tests:

   320.0/hd.0 23d ; 320.0/fd.0 4pw

3) Run the hard disk test 0 (Recalibrate) once, test 4 (Pingpong read OS disk) three times, and test 1 (Quick confidence read) and test 3 (Random read OS disk) twice:

hd.0 0 444 1313

Test Progress Display

When a command is executed in command line mode, it performs a table look-up on its own test selection menu against its arguments. If the argument matches any entry in the table, it displays a banner including the command pathname, test description, and the word `starts` (See Figure A-1). It then calls the appropriate routine for the selection. Upon completion of the routine, it displays the previous banner but ends with the word `passed` or `failed`.

```
/bdtest/320.0/fd.0    0) Recalibrate             ....... starts
/bdtest/320.0/hd.0    0) Recalibrate             ....... starts
/bdtest/320.0/fd.0    0) Recalibrate             ....... passed
/bdtest/320.0/fd.0    2) Read entire OS disk     ....... starts
/bdtest/320.0/hd.0    0) Recalibrate             ....... passed
/bdtest/320.0/hd.0    1) Quick confidence read   ....... starts
/bdtest/320.0/fd.0    2) Read entire OS disk     ....... passed
/bdtest/320.0/hd.0    1) Quick confidence read   ....... passed
/bdtest/320.0/fd.0    3) Random read OS disk     ....... starts
/bdtest/320.0/hd.0    2) Read entire OS disk     ....... starts
/bdtest/320.0/fd.0    3) Random read OS disk     ....... passed
/bdtest/320.0/hd.0    2) Read entire OS disk     ....... passed
/bdtest/320.0/hd.0    3) Random read OS disk     ....... starts
/bdtest/320.0/fd.0    4) Pingpong read OS disk   ....... starts
/bdtest/320.0/fd.0    4) Pingpong read OS disk   ....... passed
/bdtest/320.0/hd.0    3) Random read OS disk     ....... passed
/bdtest/320.0/hd.0    4) Pingpong read OS disk   ....... starts
/bdtest/320.0/hd.0    4) Pingpong read OS disk   ....... passed
```

pathname *selection* *description* *result*

Figure A-1. Test Progress Display

Test Result Display

Figure A-1 illustrates a typical output of the successful completion of all subtests in the **320.0** test module. This output is sent to the standard output device, which is the user console if there is no I/O redirection on the command line.

SSID maintains fatal and nonfatal error counters for each node above the test nodes. For the directory structure shown in Figure A-2, counters are maintained for **hd.0, fd.0, 320.0,** and **bdtest**. If an error occurs in any part of a test, the monitor updates the fatal and nonfatal error counters for each node above that test. Therefore, the total number of errors for the **320.0** node equals the number of errors in both the **fd.0** and **hd.0** tests, as illustrated in Figure A-2.

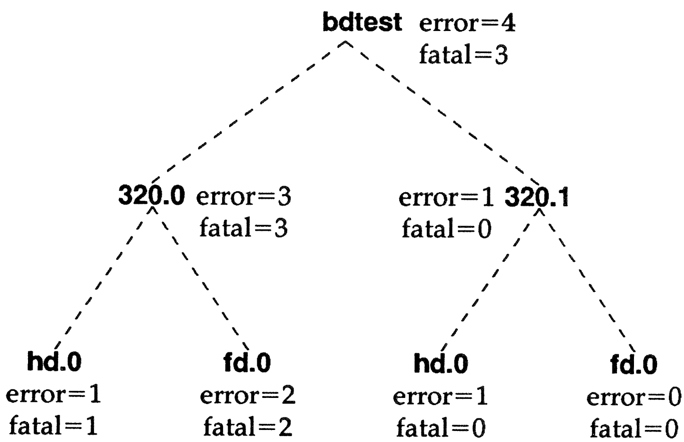


Figure A-2. Directory with Fatal and Nonfatal Error Counters

If the error is caused by a bad return status from the MVME320, the entire ECA (or packet) data structure displays on the standard error device. Figure A-3 shows the typical output for such an error.

```
/bdtest/320.0/fd.0        3) Scan                 ....... starts
|cmd|stat|ext |rtry|rcnt|dma|op|buf_addr|len |xfer|cyl |hd|sec|ccyl|
|03 | 01 |0008| 00 | 00 |00 |00|001f96b0|0400|0000|0006|00|0d |0006|
|n0|n1|n2|n3|n4|n5|sl|fill|type|surf|sec/trk|step|hst|hlt|seek|lwc |
|00|00|00|00|36|03|01| e5 | 05 | 02 |  10   | 0c |46 |46 | 00 |0028|
|prec|ecc0|ecc1|ecc2|aec0|aec1|aec2|wrk0|wrk1|wrk2|wrk3|wrk4|wrk5|
|0028|0000|0000|0000|0000|0000|0000|8888|8e8c|8480|0000|0000|0000|

error:  Nonrecoverable error, No identifier found.

/bdtest/320.0/fd.0        3) Scan                 ....... failed
```

Figure A-3. Typical Error Display

If the error is caused by the data comparison, the output shows where the data mismatch occurred in the buffer, as illustrated in Figure A-4.

```
/bdtest/320.0/hd.0        3) Partial W/R              ....... starts
/bdtest/320.0/hd.0        At address=001aff00, expect=44, read=00
/bdtest/320.0/hd.0        3) Partial W/R              ....... failed
```

Figure A-4. Data Comparison Error

Along with all above messages, a system status message periodically displays on the system console, reporting pass-count, error-count, elapsed time, and test activity as illustrated in Figure A-5.

Note that this type of message only displays if SSID is operating in concurrent mode (default condition), and at least one test has been dispatched in the system. Also note that this message is only terminated by pressing RETURN without the `LINE FEED`. Therefore, it always stays at the same place on the screen. This preserves the previous information, and provides a clear picture of progress through the tests.

```
PASS=3, FATAL=0, ERROR=01 0:00:20.130 /bdtest/320.0/hd.0:running
```

Figure A-5. System Status Message

Finally, a composite status message displays at the completion of all tests to indicate the overall test results as illustrated in Figure A-6.

```
*** SYSTEM TEST PASSED ***
            or
*** SYSTEM TEST FAILED ***
```

Figure A-6. Composite Status Message

The error count message remaining on the screen reflects the results of the last test to complete; it is not a summary of all tests run.

Command Descriptions

Table A-1 summarizes the commands available in command line mode. Following the table, each command is explained in detail.

Table A-1. Command Line Mode Commands

Command	Description
ce [*test*]	Clears error counters for the specified test. If no test is specified, all error counters are cleared. This command performs the same function as **clear** does in menu mode.
dml	Displays the actual detected installed memory (before allocations for SSID) for the system.
cf [*board*] [*test*]	Allows you to change the configuration of boards and tests recognized by SSID. **cf** with no option accesses the system level configuration (see Figure A-7).
echo *argument*	Echoes *argument* to the standard output. This command is similar to the operating system's **echo** command.
fe [*board*]	Finds errant test. **fe** displays the number of errors and fatal errors for all tests that have been run since reset or the last **ce** (or **clear** in menu mode) command. This command provides the same information as the **disperr** command available in menu mode.
he [*board*]	Displays the command structure of SSID. You can use the information given by **he** to bypass the SSID menus and initiate a test directly from the command line.
ml	Displays the memory free list in hexadecimal format. **ml** lists one line for each contiguous segment of memory.
reconf	Reconfigures test scripts for the **confid, fault, ptest,** and **ptesti** suites. Configuration information is taken from the system configuration list accessed by typing **cf** with no *option*.
set [±*option*]	Sets *option* for the test indicated on the command line. A preceding plus sign (+) or no prefix before *option* enables it; a preceding minus sign (–) disables *option*. Without *option*, **set** displays the current test options in command line mode.
set menu	Returns SSID to menu mode and the `SA:` prompt.

In addition, the system commands **disphrd, remote, help,** and **version** can be executed while in command line mode.

NAME

ce – clear error counters

SYNTAX

ce [*test*]

DESCRIPTION

ce clears the error counter for the specified test module. If no test module name is specified, all error counters are cleared. SSID maintains an error counter for each test module. These error counters are cleared at startup and may be cleared at other times by using **ce** (or **clear** in menu mode). No output is provided to confirm that the error counters have been cleared. Use **fe** (or **disperr** in menu mode) to confirm that errors have been cleared.

EXAMPLES

1. Clear all error counters:

   ce

2. Clear the error counter for the fd.0 test suite:

   ce fd.0

NAME

cf – configure SSID

SYNTAX

cf [*board*] [*test*]

DESCRIPTION

cf allows you to change variables that control many aspects of SSID execution. These variables include quantity and selection of boards in the system configuration, type of peripherals available for test, base addresses, interrupt levels, address modifiers, and so on.

SSID has three levels of configuration variables: system, board, and test. **cf** with no option accesses the system level configuration; **cf** *board* accesses the board level configuration; and **cf** *test* accesses the test level configuration.

Configuration variables at all three levels are presented one line at a time. Lines containing configuration variables end with a question mark (?). Changes to configuration variables are entered following the question mark. Pressing (RETURN) leaves the current variable unchanged and displays the next line in the configuration. It is not possible to scroll up in a configuration display. If you pass over a line you wish to change, finish listing the display and invoke **cf** again.

The configuration displays show the range of legal values for each configuration variable and the current value. The default configuration is set up based on the hardware detected when SSID is booted. At startup, all configuration variables are initialized to default values. The default values are placed in a configuration file that may be changed to remove boards from the displays and (using **reconf**) the test suites. (Note you must execute **reconf** before exiting command line mode to affect test scripts.) Sometimes, the range of values in the menu may be further restricted by the hardware configuration. Refer to the hardware user's manual for each board or peripheral being configured to determine proper values.

System Configuration

The system level configuration display is similar to the *System Configuration List*.

System Configuration List

```
> cf /bdtest configuration
    050 [0-1] = 0 ?                # number of MVME050
    13x [0-1] = 1 ?                # number of MVME13x
    mem [0-1] = 1 ?                # memory test enabled
    320 [0-2] = 1 ?                # number of MVME320
    321 [0-2] = 0 ?                # number of MVME321
    323 [0-2] = 0 ?                # number of MVME323
    G330 [0-1] = 0 ?               # number of MVME330GSP
    T330 [0-1] = 0 ?               # number of MVME330TCP
    331 [0-4] = 0 ?                # number of MVME331
    332 [0-4] = 1 ?                # number of MVME332
    x332 [0-16] = 1 ?              # number of MVME332XT
    333 [0-4] = 1 ?                # number of MVME333
    335 [0-2] = 1 ?                # number of MVME335
    336 [0-1] = 1 ?                # number of MVME336
    350 [0-2] = 1 ?                # number of MVME350
    355 [0-1] = 0 ?                # number of MVME355
    360 [0-2] = 1 ?                # number of MVME360
>
```

You may alter the configuration to include a new board or to remove an existing board which you do not want to have tested. Refer to Example 1.

Board and Test Configuration

To display the configuration variables for a given board, type:

>**cf** *board.n*

where *n* is the number of the board in the series with **0** designating the first board. For example, **360.1** refers to the second MVME360 board.

To display the configuration variables for a given test, type:

>**cf** *test*

If a particular test is associated with more than one board in the system configuration, you may need to specify the board as well as the test (e.g., **cf 331.0/sp.0** to distinguish it from **332.0/sp.0**).

The acceptable range of input values is represented as:

0x Hexadecimal. `0x100` is the hexadecimal number 100, or 256 in decimal. Numbers not preceded by `0x` are decimal.

% Modulo. `%0x100` indicates that the value of a variable must be modulo 256 or 0x100.

– Range. `0-7` represents any number in the range of 0 through 7.

`b,w,l` Bus Width. `b` is byte, `w` is word, and `l` is longword.

NOTE

You cannot configure a test until a board is configured to include that test; likewise, you cannot change a board configuration unless the system configuration includes that board.

EXAMPLES

1. Changing System Configuration

   Make it possible to run the communication fault test only on the MVME332 board although the system has both an MVME332 and MVME333 board.

   cf (RETURN)

   ```
   /bdtest configuration
           355 [0-1] = 0 ? (RETURN)
           360 [0-2] = 1 ? (RETURN)
           350 [0-2] = 1 ? (RETURN)
           333 [0-4] = 1 ? (RETURN)
           332 [0-4] = 1 ? (RETURN)
   ```

   (Continue pressing the (RETURN) *key until the* > *prompt reappears.)*

   Once the system configuration list has been changed, the test scripts for the **confid**, **fault**, **ptest**, and **ptesti** test suites must be reconfigured based on the new information. Refer to the following **reconf** command description.

2. Changing Default Drive Type

   Change the default drive type for the first MVME320 from Micropolis to Toshiba.

   cf 320.0 (RETURN)

   ```
   /bdtest/320.0  configuration
      base [%0x400] = 0xffffb000 ? (RETURN)
      am[0x0d,0x3d] = 0x3d ? (RETURN)
      level [0-7] = 5 ? (RETURN)
      data bus width [b,w,l] = l ? (RETURN)
      type 0Micropolis 1Toshiba [0,1] = 0 ? (RETURN)
   ```

   (Type **1** *to select a Toshiba drive.)*

   (Continue pressing the (RETURN) *key until the* > *prompt reappears.)*

3. Selecting Address Increment Value for Memory Tests

   Change the default address increment value for the memory tests from testing every longword to every fourth longword.

   cf mem.0 RETURN

```
/bdtest/mem.0 configuration
      walk_step = 1 ? 4 RETURN
```

4. Changing Default Drive Size

   Change the size of the first floppy disk drive from 655K to 1.2Mb.

   cf 320.0/fd.0 RETURN

```
/bdtest/320.0/fd.0 configuration
```

(Only parameters requiring changes are shown here.)

```
   sector/xfer[1-16] = 16 ?  15 RETURN
   size(inch) [5,8] = 5 ? 8 RETURN
   no of sector/track = 16 ? 15 RETURN
   sector size[128,256,512] = 256 ? 512 RETURN
```

NAME

dml – display detected memory list

SYNTAX

dml

DESCRIPTION

This command displays the actual detected installed memory for the system. Unlike the **ml** command, **dml** probes memory before any allocations being made for SSID. Therefore, it displays an accurate map of all memory installed in the system. Each block of contiguous space takes one line to display and all memory, contiguous or not, is represented.

EXAMPLES

Display detected memory.

dml

```
Total memory detected = 0x00400000, (4194304 bytes)

Seg     Start         End          Size
 0      0x00000000    0x001fffff   0x00200000
 1      0x00400000    0x005fffff   0x00200000
```

In this example, the total detected memory installed in the system is 4Mb, displayed in both hex and decimal. This is made up of two contiguous segments of memory as itemized:

The first segment starts at address 0, ends at address 0x1fffff and is 0x200000 bytes.

The second segment starts at address 0x400000, ends at address 0x5fffff, and is also 0x200000 bytes.

The gap between the end address of segment 0 (0x200000) and the start address of segment 1 (0x3fffff) constitutes a *hole* detected in the RAM address space.

NAME

echo – echo message to standard output

SYNTAX

echo [*argument*]

DESCRIPTION

This command echoes an argument to standard output. It is especially useful for examining the log.

EXAMPLES

1. Output a character string to the terminal:

   echo Good morning!

2. Display the log on the terminal by redirecting input from the log:

   echo < log

NAME

fe – find errant test

SYNTAX

fe [*test*]

DESCRIPTION

Like the **disperr** command in menu mode, **fe** displays the number of errors and fatal errors for all tests that have run since reset or the last **ce** (or **clear** in menu mode) command. If a test name is specified, only errors for that test and those under it display.

EXAMPLES

1. Display all errors:

   fe

```
/bdtest/320.0/fd.0        :error=1,  fatal=1
/bdtest/320.1/hd.0        :error=3,  fatal=1
/bdtest/320.1/hd.1        :error=5,  fatal=1
```

2. Display only errors from MVME320.0 tests:

   fe 320.0

```
/bdtest/320.0/fd.0        :error=1,  fatal=1
```

NAME

he – online help

SYNTAX

he [*test*]

DESCRIPTION

he displays a description of *test* and each subtest under it. If *test* is not specified, the entire SSID command structure, starting with root, displays.

The mode characters in the **he** display are interpreted as:

d = directory	r = read
t = test	w = write
D = device	x = executable

Use <^s> to stop the display from scrolling off the screen. Resume scrolling with <^q>.

EXAMPLES

1. Display the entire SSID command structure:

   he

```
/               d--- System monitor
   help         t--x   Using Diagnostics
   confid       d---   Confidence Testing
   .               .          .
   .               .          .
/bdtest         d--- Tests by Board
   131.0        t--x   VME131 68020 CPU Tests
   204.0        d--x   VME204 Computer System Memory Tests
   320.0        d--x   VME320 Disk Controller Tests
   .               .             .
   .               .             .
/bdtest/204.0   t--x VME204 Computer System Memory Tests
   mem.0        t--x   Memory Test
   .               .       .
   .               .       .
/cmdline        d--- Command Line Operations
   mcs/         d---   MCS options
   dv/          drw-   I/O devices
   opt/         d---   System options
```

```
/cmdline/mcs     d--- MCS options
   reconf        t--x   Reconfigure test scripts
   version       t--x   @(#)confile.src version date
   setopt        t--x   Set test options
   remote        t--x   Remote Diagnostic Link
   disphrd       t--x   display known hardware
/cmdline/dv      drw- I/O devices
   log           Drw-   logger
   sp1           Drw-   host port
   sp0           Drw-   console port
   pr            Drw-   printer 050
/cmdline/opt     d--- System Options
   he            t--x   help !!!!!!
   fe            t--x   find errant test
   cf            t--x   configure test
   ml            t--x   memory free list
   ce            t--x   clear error counters
   set           t--x   set variables
   echo          t--x   display msg
```

2. Display a test with descriptions of each subtest:
   he 320.0

```
/bdtest/320.0        d--x     VME320 Disk Controller Tests
   hd.1              t--x       Winchester Disk
   hd.0              t--x       Winchester Disk
   fd.1              t--x       Floppy Disk
   fd.0              t--x       Floppy Disk
```

NAME

ml – display memory free list

SYNTAX

ml

DESCRIPTION

This command displays the current allocation of available memory under SSID. SSID dynamically allocates memory on an as-needed basis. Areas of available memory are not necessarily contiguous. The memory free list contains one line for each contiguous segment of memory. `head` is the starting memory location, and `size` is the number of bytes of contiguous memory starting at the head address.

EXAMPLE

Display available memory:

ml

```
Free memory list
     head =      13150, size = e85b0
     head =      fb800, size = 400
```

In this example, the first segment starts at address `13150` and is `e85b0` bytes long; it thus ends with address `fb6ff`. The second segment starts at address `fb800` and is `400` bytes long; it thus ends with address `fbbff`. The memory between `fb700` and `fb800` (0x100 bytes) is currently allocated to some task within SSID.

NAME

reconf – reconfigure the menu mode test suites

SYNTAX

reconf

DESCRIPTION

The **reconf** command is used to reconfigure the **confid** and **fault** test suites to reflect changes made in the system configuration list (see *System Configuration List*). If **reconf** is not executed after changing the system configuration list, the test suites do not recognize the added or deleted board(s). The **bdtest** menus, on the other hand, automatically incorporate changes to the system configuration list.

EXAMPLE

1. Reconfigure test scripts **confid** and **fault** to reflect a change in the system configuration list:

   reconf

NAME

set – set environment options

SYNTAX

set [±*option*]

DESCRIPTION

The **set** command is used to view or change the state of options that control the operation of SSID. When *option* is not specified, **set** displays the current status of the test options. When *option* is preceded by a plus sign (+) or nothing, it is enabled. When preceded by a minus sign (–), *option* is disabled.

In command line mode, the options **lc, le, ce** and **time** are volatile; i.e., they must be set on the current command line, but they will return to their default values after processing of the command line is complete. The remaining values are non-volatile, holding their current values until changed by **set**.

EXAMPLES

1. Display the current list of options.

   set

```
  lc - disabled loop-continue
  le - disabled loop-on-error
  li - disabled loop-on-intermittent-error
  vb - enabled verbose-mode
  ce - disabled continue-on-error
  ae - disabled display all errors
  dp - disabled display packet
 con - enabled concurrent-mode
time - disabled execution-time
menu - enabled prompt by menu
```

   Refer to *The setopt Command* in the *System Commands* chapter for an explanation of these options.

2. Enable the option **lc** to cause SSID to loop continuously in the **tp.0** test.

   set lc;tp.0

   When the test is complete, **lc** is again disabled.

3. Disable the verbose mode, enable the display of elapsed time, and run test 1 on the first Winchester disk drive.

   set -vb time;hd.0 1

NOTE

Option settings made using **set** are valid in command line mode only; option settings made using **setopt** are valid in menu mode only but do not affect the options used in the **confid/fault** and peripheral test scripts.

NAME

Return to menu mode

SYNTAX

set menu

DESCRIPTION

set menu returns SSID to menu mode, displaying the last screen that was accessed before the **cmdline** command was executed. Menu mode provides confidence, fault, and peripheral test suites not available in command line mode.

EXAMPLE

1. Exit command line mode:

   set menu
   SA:

APPENDIX B
LOOPBACK WIRING

This appendix details the wiring information for the loopbacks used for SSID testing.

Connector Pin #		to		Connector Pin #
(Tx)	2	<——————>	3	(Rx)
(Rx)	3	<——————>	2	(Tx)
(RTS)	4	<——————>	5	(CTS)
(CTS)	5	<——————>	4	(RTS)
(DCD)	8	<——————>	20	(DTR)
(DTR)	20	<——————>	8	(DCD)

Figure B-1. DB-25 Pin Male Loopback Cable Pinout

The above port-to-port loopback is used with the following boards that have DB-25 connectors: MVME147, MVME331, MVME332, MVME332xt, MVME333, MVME335, MVSB741.

Connector Pin #		to		Connector Pin #
(DCD)	1	<——————>	4	(DTR)
(Rx)	2	<——————>	3	(Tx)
(Tx)	3	<——————>	2	(Rx)
(DTR)	4	<——————>	1	(DCD)
(RTS)	7	<——————>	8	(CTS)
(CTS)	8	<——————>	7	(RTS)

Figure B-2. DB-9 Pin Male Loopback Cable Pinout

The above port-to-port loopback is used with the MVME335 board with DB-9 connectors.

Connector Pin #		to		Connector Pin #
(Rx-)	2	<——————>	4	(Tx-)
(Tx-)	4	<——————>	2	(Rx-)
(Rx+)	6	<——————>	9	(Tx+)
(Tx+)	9	<——————>	6	(Rx+)

Figure B-3. RS422 DB-9 Pin Male Loopback Cable Pinout

The above port-to-port loopback is used with the MVME705A-1 board with DB-9 connectors.

Connector Pin #		to		Connector Pin #
(Rx)	2	<——————>	3	(Tx)
(Tx)	3	<——————>	2	(Rx)
(CTS)	4	<——————>	5	(RTS)
(RTS)	5	<——————>	4	(CTS)
(DTR)	6	<——————>	8	(DCD)
(TxCLK)	7	<——————>	9	(RxCLK)
(DCD)	8	<——————>	6	(DTR)
(RxCLK)	9	<——————>	7	(TxCLK)

Figure B-4. RS232 DB-9 Pin Male Loopback Cable Pinout

The above port-to-port loopback is used with the MVME705A-2 board with DB-9 connectors.

Pin #		to		Pin #
(Rx)	2	<——————>	3	(Tx)
(RTS)	4	<——————>	5	(CTS)
(CTS)	5	<——————>	6	(DSR)
(DCD)	8	<——————>	20	(DTR)

Figure B-5. DB-25 Pin Male Loopback Connector Pinout

The above single-port loopback is used with the following boards having DB-25 connectors: MVME167, MVME187, MVME336.

Pin #		to	Pin #	
(STROBE)	1	<————————>	10	(ACKNOWLEDGE)
(DATA0)	2	<————————>	13	(SELECT)
(DATA1)	3	<————————>	11	(BUSY)
(DATA2)	4	<————————>	32	(FAULT)
(DATA3)	5	<————————>	12	(PAPER ERROR)

Figure B-6. 36-Pin Male Printer Port Loopback Connector Pinout

The above printer port loopback is used with the following boards having 36-pin connectors: MVME167, MVME187, MVME335.

Pin #		to	Pin #	
(STROBE)	1	<————————>	10	(ACKNOWLEDGE)
(DATA0)	2	<————————>	13	(SELECT)
(DATA1)	3	<————————>	11	(BUSY)
(DATA2)	4	<————————>	15	(FAULT)
(DATA3)	5	<————————>	12	(PAPER ERROR)

Figure B-7. DB-25 Pin Male Printer Port Loopback Connector Pinout

The above printer port loopback is used with the following boards having DB-25 connectors: MVME167, MVME187.

Connector Pin #		to	Connector Pin #	
(DCD)	1	<——┐ ┌——>	1	(DCD)
(DTR)	4	<——┘ └——>	4	(DTR)
(Rx)	2	<————————>	3	(Tx)
(Tx)	3	<————————>	2	(Rx)
(RTS)	7	<——┐ ┌——>	7	(RTS)
(CTS)	8	<——┘ └——>	8	(CTS)

Figure B-8. DB-9 Pin Male Loopback Cable Pinout

The above port-to-port loopback is used with the MVME792-1 board with DB-9 connectors.

Pin #		to	Pin #	
(DCD)	1	<————————>	4	(DTR)
(Rx)	2	<————————>	9	(Tx)
(RTS)	5	<————————>	6	(CTS)
(DCD)	7	<————————>	10	(DTR)
(RX)	8	<————————>	3	(Tx)

Figure B-9. 17W5 Male Loopback Connector Pinout

The above loopback is used with the MVME792-2 and MVME795 boards with 17W5 connectors.

Pin #		to	Pin #	
(DCD)	1	<————————>	4	(DTR)
(Rx)	2	<————————>	3	(Tx)
(RTS)	7	<————————>	8	(CTS)

Figure B-10. DB-9 Pin Female Loopback Connector Pinout

The above single-port loopback is used with the following boards having DB-9 connectors: MVME167, MVME187.

Pin #		to	Pin #	
(DCD)	1	<————————>	4	(DTR)
(Rx)	2	<————————>	3	(Tx)
(RTS)	7	<————————>	8	(CTS)

Figure B-11. DB-9 Pin Female Loopback Connector Pinout (Mouse Port)

The above single-port loopback is used with the MVME796 for the DB-9 serial (mouse) port.

Connector Pin #		to		Connector Pin #
(DCD)	1	<-------------------->	4	(DTR)
(Rx)	2	<-------------------->	3	(Tx)
(Tx)	3	<-------------------->	2	(Rx)
(DTR)	4	<-------------------->	1	(DCD)
(RTS)	7	<-------------------->	8	(CTS)
(CTS)	8	<-------------------->	7	(RTS)

Figure B-12. DB-9 Pin Female Loopback Cable Pinout

The above port-to-port loopback is used with the MVME147 with DB-9 connectors.

Pin #		to		Pin #
(Rx)	2	<————————>	3	(Tx)
(RTS)	4	<————————>	5	(CTS)
(DSR)	6	<————————>	8	(DCD)
(DCD)	8	<————————>	20	(DTR)

Figure B-13. DB-25 Pin DCE Male Loopback Connector Pinout

The above single-port loopback is used with the M8120 on ports 1, 2, and 3. The signal names used above are what the serial port test will display.

Pin #		to		Pin #
(Rx)	2	<————————>	3	(Tx)
(RTS)	4	<————————>	5	(CTS)
(CTS)	5	<————————>	6	(DSR)
(DCD)	8	<————————>	20	(DTR)
(DTR)	20	<————————>	22	(RI)
(TXC)	15	<————————>	17	(RXC)
(RXC)	17	<————————>	24	(RXCO)

Figure B-14. DB-25 Pin DTE Male Loopback Connector Pinout

The above single-port loopback is used with the M8120 on ports 4 and 5. The signal names used above are what the serial port test will display.

GLOSSARY

board testing

Testing that verifies operation of a specific board in the system. Board testing is valuable whenever a board is replaced (refer to the *Confidence, Fault, Board, and Peripheral Testing* chapter).

ce

A command line mode command that clears error counters for the specified test. If no test is specified, all error counters are cleared. This command performs the same function as **clear** does in menu mode.

cf

A command line mode command that allows you to change the configuration of boards and tests recognized by SSID. **cf** with no option accesses the system level configuration (see Figure A-7). Refer to Appendix A for details.

clear

A system command that clears all error logs and pass counters. Refer to the *clear Command* section in the *System Commands* chapter for details.

cmdline

A system command that accesses command line mode. Command line mode offers features not available in menu mode, e.g., the capability to change configuration variables, reconfigure test suites, and list the entire SSID command structure.

NOTE

Command line mode is reserved for advanced users of SSID. Refer to Appendix A for details.

command packet

A set of parameters sent to a board or module to cause it to execute a specific command. A command packet might contain a command to read data, the code for the device to read data from, and the amount of data to read.

command review block

The upper portion of SSID menus that serves as a reminder of the SSID commands you can select when not testing.

concurrent testing

Testing that allows tests to run together, using time slicing for maximum use of the CPU for each test.

confidence testing
Testing that verifies total system health by running a dynamically configured suite of tests; it is useful at initial system installation and after a major system component replacement. Refer to the *Confidence, Fault, Board, and Peripheral Testing* chapter.

destructive test
A test that modifies the data stored on a storage device attached to a board. Do not do destructive testing on a disk until you have backed up the data or replaced the disk drive.

diagnostic
A kind of test that probes a device that has failed a go/no-go test to find the cause of the failure.

disperr
(display error) A system command that lists the accumulated errors and the pass count. If no errors occurred, only the pass count is shown. Refer to the *disperr Command* section in the *System Commands* chapter for details.

disphrd
(display hardware) A system command that displays the boards whose presence is sensed by SSID. (If a board is completely dead it may not be recognized.) Refer to the *disphrd Command* section in the *System Commands* chapter for details.

dml
(display memory list) A command line mode command that displays the actual installed memory for the system. Refer to Appendix A for details.

echo
A command line mode command that echoes a given argument to the standard output. This command is similar to the operating system's **echo** command. Refer to Appendix A for details.

fault isolation
The process of eliminating conditions and hardware components until the failing part and the test that causes the failure are identified.

fault testing
Testing that checks and diagnoses specific system failures by running a dynamically configured suite of tests.

fe

A command line mode command that finds errant tests. **fe** displays the number of errors and fatal errors for all tests that have run since reset or the last **ce** (or **clear** in menu mode) command. This command provides the same information as the **disperr** command available in menu mode. Refer to Appendix A for details.

hard error

Also known as fatal error, an error that is repeatable and cannot be corrected. Hard errors include nonexistent controller or device, illegal command, media not in place, or soft errors for which the retry count has been reached.

he

A command line mode command that displays the command structure of SSID. You can use the information given by **he** to bypass the SSID menus and initiate a test directly from the command line. Refer to Appendix A for details.

help

A system command that accesses the **help** program. Once in the **help** program, you can access online information concerning SSID operations, tests, and test suites. Refer to the *help Command* section in the *System Commands* chapter for details.

interference

In SSID, interference refers to situations in which the activity of one board causes failures in another board. Sometimes failures occur only if the interfering board is running a specific test. However, just the presence of the interfering board in the system may be sufficient in other cases. In certain circumstances, the presence of a module may even hide a failure in another module or cause multiple problems to show up at the same time.

intermittent failure

A failure that occurs at random rather than in a predictable pattern. The time between failures may be a few seconds or a few days.

ml

A command mode command that displays the memory free list. **ml** lists one line for each contiguous segment of memory. Refer to Appendix A for details.

non-destructive test

A test that does not modify the data on a storage device attached to a board. It is not necessary to back up data contained on a device undergoing non-destructive testing.

peripheral testing
Testing that checks and diagnoses disk drives and tape devices attached to the system by running a dynamically configured suite of tests.

reconf
A command mode command that reconfigures test scripts for the **confid** and **fault** suites. Refer to Appendix A for details.

remote
A system command that allows connection to a remote terminal connected to a modem or to a remote terminal on the host port of the CPU transition board. (This may be referred to as the `host port` or as `port 2` on the board itself.) Once connected to the remote site, all keyboard and display functions are in parallel. All SSID functions can then be performed from the remote site. Refer to the *remote Command* section in the *System Commands* chapter for details.

SCSI-2
A reference to the Small Computer System Interface version 2 as defined by American National Standards Institute (**ANSI**).

set
A command line mode command that displays current test options, and sets test options directly on the command line. Refer to Appendix A for details.

set menu
A command line mode command that returns SSID to menu mode and the `SA:` prompt. Refer to Appendix A for details.

setopt
A system command that allows you to enable or disable test options such as loop-on-error, verbose mode, and concurrent mode. Refer to the *setopt Command* section in the *System Commands* chapter for details.

soft error
Also known as non-fatal error, an error that may not occur again if the same operation is repeated. If the configured number of soft error retries is performed without getting correct data, then a hard (fatal) error has occurred.

test
A program that determines whether or not a device is performing correctly by subjecting it to go/no-go situations.

version
A system command that displays the current revision data for SSID. Refer to the *version Command* section in the *System Commands* chapter for details.

view

A system command that displays the individual board tests making up any confidence, fault, or peripheral suite. Refer to the *view Command* section in the *System Commands* chapter for details.

INDEX

When using this index, keep in mind that a page number indicates only where referenced material begins; it may extend to the following page or pages.

A

B

INDEX

C

INDEX

F

G

H

INDEX

INDEX

INDEX

S

INDEX

Computer Group
2900 South Diablo Way
Tempe, Arizona 85282
P. O. Box 2953
Phoenix, Arizona 85062-2953

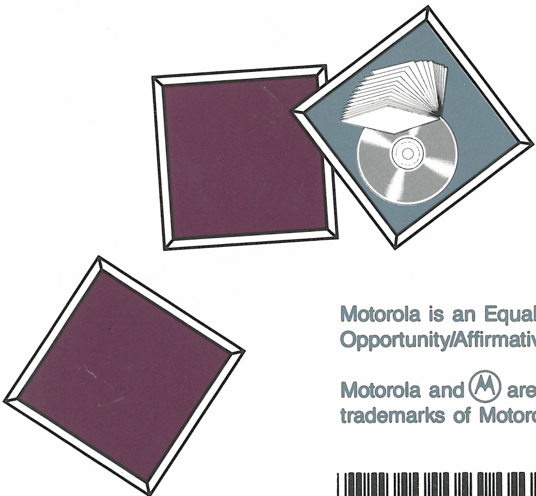
To order additional Computer Group literature:

Motorola, Inc.
Technical Literature Center
1919 W. Fairmont, Suite 8
Tempe, Arizona 85282

PHONE: 1-800-458-6443
FAX: 602-438-0240

To comment on Motorola hardware, software, or system products, contact:

Motorola Field Service Division
Customer Support Center
1-800-551-1016



Motorola is an Equal Employment Opportunity/Affirmative Action Employer

(P) SSIDUG/D6

11978 PRINTED IN USA 7-92 IRONWOOD LITHO 5,000