SunVTS 3.0 Test Reference Manual

Please
Recycle

Adobe PostScript™

Contents

Figures

Tables

Preface

SunVTS 3.0 is Sun's online Validation Test Suite. SunVTS is a comprehensive software diagnostic package that tests and validates Sun™ hardware by verifying the configuration and functionality of most hardware controllers, devices, and platforms.

SunVTS is primarily used from a graphical user interface (GUI), which may be either OPEN LOOK or Common Desktop Environment™ (CDE). This book describes SunVTS tests that run on machines with SPARC™ architectures. The descriptions include specific test options, procedures, and error messages.

This book is primarily written as a reference for SunVTS test specific information. Refer to the *SunVTS 3.0 User's Guide* for overall SunVTS information. Developers or experienced users who want to run the SunVTS diagnostic application will find these documents useful.

Before You Read This Book

In order to make full use of the information in this document, you may need access to the following documents:

- *SunVTS 3.0 User's Guide*
- *SunVTS 3.0 Quick Reference Card*

How This Book Is Organized

This book is organized as follows:

Chapter 1 describes SunVTS requirements, test modes, user interfaces, the collection of tests, and how to run a test from the command line.

The remaining chapters describe the individual SunVTS tests, their options, command line syntax, other applicable test modes, and error messages. These chapters are arranged in alphabetical order according to each test name.

Appendix A provides information about the serial and parallel port loopback connectors that are required by some of the SunVTS tests.

Using UNIX Commands

This document does not contain information on basic UNIX® commands and procedures such as shutting down the system, booting the system, and configuring devices.

See one or more of the following for this information:

- *Solaris Handbook for Sun Peripherals*
- AnswerBook™ online documentation for the Solaris™ 7 software environment
- Other software documentation that you received with your system

Typographic Conventions

TABLE P-1   Typographic Conventions

Typeface or Symbol	Meaning	Examples
`AaBbCc123`	The names of commands, files, and directories; on-screen computer output.	Edit your `.login` file. Use `ls -a` to list all files. `% You have mail.`
`AaBbCc123`	What you type, when contrasted with on-screen computer output.	`% `**`su`** `Password:`
AaBbCc123	Book titles, new words or terms, words to be emphasized. Command-line variable; replace with a real name or value.	Read Chapter 6 in the *User's Guide.* These are called *class* options. You *must* be `root` to do this. To delete a file, type `rm` *filename.*

Shell Prompts

TABLE P-2   Shell Prompts

Shell	Prompt
C shell	*machine_name*%
C shell superuser	*machine_name*#
Bourne shell and Korn shell	$
Bourne shell and Korn shell superuser	#

Related Documentation

This manual covers SunVTS revision 3.0 tests. It serves as an updated reference companion to the SunVTS 3.0 manuals listed below.

TABLE P-3   Related Documentation

Title	Part Number
SunVTS 3.0 User's Guide	805-4442
SunVTS 3.0 Quick Reference	805-4444

Sun Documentation on the Web

The `docs.sun.com` web site enables you to access Sun technical documentation on the World Wide Web. You can browse the `docs.sun.com` archive or search for a specific book title or subject at `http://docs.sun.com.`

Sun Welcomes Your Comments

We are interested in improving our documentation and welcome your comments and suggestions. You can email your comments to us at `docfeedback@sun.com`. Please include the part number of your document in the subject line of your email.

Introduction

The Sun Validation and Test Suite (SunVTS) software runs multiple diagnostic hardware tests from a single user interface. SunVTS verifies the configuration, functionality, and reliability of most hardware controllers and devices.

SunVTS is comprised of many individual tests that support testing of a wide range of products and peripherals. Many of the tests are capable of testing devices in a 32-bit or 64-bit Solaris environment.

Use SunVTS to test one device or multiple devices. Some of the major test categories are:

- Audio Tests
- Communication (Serial and Parallel) Tests
- Graphic/Video Tests
- Memory Tests
- Network Tests
- Peripherals (Disks, Tape, CD-ROM, Printer, Floppy) Tests
- Processor Tests
- Storage Tests

Such flexibility means that the proper test modes and options need to be selected to maximize its effectiveness. This book covers the individual test options and requirements. For overall test configuration modes and options refer to the *SunVTS 3.0 User's Guide.*

This manual describes SunVTS Version 3.0 tests, that are distributed on the Sun Computer Systems Supplement CD.

The default installation directory for SunVTS is `/opt/SUNWvts`. However, when you are installing SunVTS, you can specify a different directory. Refer to the *SunVTS 3.0 User's Guide* for installation information.

Requirements

SunVTS revision 3.0 was first introduced, and designed to run in the Solaris 7 operating environment.

The operating system kernel must be configured to support all peripherals that are to be tested.

The system must be booted and the SunVTS packages must be installed. Refer to the *SunVTS 3.0 User's Guide* for more information.

Some SunVTS tests have special requirements such as the connection of loopback connectors, installation of test media, or the availability of disk space. These requirements are listed for each test in the corresponding chapter in this book.

Test Modes

Two modes of testing are available—Connection test and Functional test. These modes differ in their assumptions about the state of the system you are testing and your objectives.

Only one test mode can be selected per test session. Each mode is described below:

Connection Test Mode

In Connection test mode, the tests determine if the devices are connected to the system you are testing and verifies that they are accessible.

Device functionality is not verified, however, you can safely run this mode when the system is online.

Functional Test Mode

In Functional test mode, the tests fully exercise all aspects of the selected devices and their associated device drivers.

In Functional test mode you can run a single test in to verify the functionality of a single device, or run multiple tests on all system devices for full system verification.

For thorough testing, the tests use a significant portion of the system resources. For this reason, do not run critical applications on the system or use the system for production purposes while testing in Functional test mode.

In Functional test mode, tests expect all system resources that are associated with the device to be available for testing. If the test cannot access a device, it registers a failure. The tests do not economize on runtime, but focus on achieving complete coverage and thoroughly exercising the device.

Note – SunVTS does not verify that the system is in a safe offline state for Functional mode testing. You must be sure that your applications and the SunVTS diagnostic will not compete for system resources. For more information regarding runtime considerations refer to the *SunVTS 3.0 User's Guide*.

Functional Test Mode via Solstice SyMON

Solstice SyMON identifies a range of hardware and system status states quickly.

Solstice SyMON has an online diagnostic interface, so you can access SunVTS when running SyMON. In this case, Functional test mode does operational testing to find and isolate faults, while minimizing the impact on other applications and users.

When Functional test mode is accessed from SyMON, the system may be running critical production software. The tests are sensitive to this fact and usually try to achieve as much coverage as possible within the constraints imposed. In this mode, some of the test parameters, execution options, and some system level options are fixed (have preassigned values) that cannot be changed. This ensures that the system state is not violated by selecting an option or combination of options which could trigger unsafe actions.

TABLE 1-1 shows the default values of the test execution options in different modes.

TABLE 1-1   Test Execution Options

Option	Connection Test Mode	Functional Test Mode	Functional Test Mode via SyMON
Stress	Disabled (fixed)	Disabled	Disabled (fixed)
Verbose	Disabled (fixed)	Disabled	Disabled (fixed)
Core File	Disabled (fixed)	Disabled	Disabled (fixed)
Run On Error	Disabled (fixed)	Disabled	Disabled (fixed)

TABLE 1-1   Test Execution Options

Option	Connection Test Mode	Functional Test Mode	Functional Test Mode via SyMON
Max Passes	1 (fixed)	0	1 (fixed)
Max Time	0 (fixed)	0	0 (fixed)
Number of Instances	1 (fixed)	Dependent on the number of processors	1 (fixed)

SunVTS User Interfaces

You can run SunVTS tests from various interfaces: The CDE and OL graphical user interfaces, or the TTY interface. SunVTS tests can also be run individually from a shell command line, using the command line syntax for each test (refer to "Running a Test from the Command Line" on page 6). Table 1-1 describes the various SunVTS system interfaces. Refer to the *SunVTS 3.0 User's Guide* for more information on these interfaces.

TABLE 1-2   SunVTS System Interfaces

SunVTS System Interfaces	Description
Graphical User Interfaces (GUIs)	Users can select tests and test options by pointing and clicking with a mouse button in the CDE or OL interface.
TTY Interface	Users can run SunVTS from a terminal or modem attached to a serial port. This feature requires that you use the keyboard instead of the mouse, and it displays one screen of information at a time.
Command Line execution	Lets users run each of the SunVTS tests individually from a shell command line using the command line syntax. Each test description in this book contains the corresponding command line syntax.
SyMON GUI	Provides access to SunVTS testing through the SyMON monitoring tool. With this interface, the SunVTS tests run in a mode that is non-intrusive to other system activity.

Collection of SunVTS tests

Many individual tests make up the collection of tests in the SunVTS application. Each test is a separate process from the SunVTS kernel. Each test can be run individually from the command line or from the SunVTS user interface.

When SunVTS is started, the SunVTS kernel automatically probes the system kernel to determine the hardware devices. The devices are then displayed on the SunVTS control panel with the appropriate tests and test options. This provides a quick check of your hardware configuration, and no time is wasted trying to run tests that are not applicable to your configuration.

During testing, the hardware tests send the test status and messages to the SunVTS kernel through interprocess communication (IPC) protocols. The kernel passes the status to the user interface and logs the messages.

SunVTS has a shared object library that contains test-specific probing routines. At runtime, the SunVTS kernel dynamically links in and calls these probing routines to initialize its data structure with test-specific information. You can add new tests into the SunVTS environment without recompiling the SunVTS source code.

As of SunVTS 3.0, the SunVTS kernel and most tests support 32-bit and 64-bit operating environments. When the `sunvts` command is used to start SunVTS, the appropriate tests (32-bit or 64-bit versions) are presented.

32-Bit and 64-Bit Tests

Because each test is a separate program, you can run individual tests directly from the command line. When this is done, care must be taken to run the appropriate test (32-bit or 64-bit) that corresponds to the operating system that is running (32-bit or 64-bit). This is done by running tests from specific directories as follows:

- 32-bit tests—`/opt/SUNWvts/bin/`*testname*
- 64-bit tests—`/opt/SUNWvts/bin/sparcv9/`*testname*
    - The test is an actual 64-bit binary test if *testname* is a binary file
    - The test is a 32-bit test capable of running in the 64-bit environment if *testname* is a symbolic link

---

Note – The SUNWvtsx package must be installed for 64-bit SunVTS support. For more information on SunVTS packages and installation procedures refer to the *SunVTS 3.0 User's Guide.*

If you use the `sunvts` command to run SunVTS, SunVTS will automatically allocate 32-bit or 64-bit tests based on the 32-bit or 64-bit Solaris operating environment that is running. Therefore, the only time that you need to be concerned with the 32-bit or 64-bit operation is when you run the SunVTS kernel or SunVTS tests from the command line.

If you are not sure which operating system is running, refer to the *Solaris 7 System Administration* manuals. In Solaris 7, the following command can be used to identify the application support of your system.

```
# isainfo -v
```

Note – The `isainfo` command is not available in Solaris 2.6 or earlier releases.

Running a Test from the Command Line

In some cases it may be more convenient to run a single SunVTS test from the command line rather than through a SunVTS user interface. The following information describes how to do this.

Unless specified, the test runs without the SunVTS kernel (`vtsk`). All events and errors are sent to `stdout` or `stderr`, and are not logged in the log files.

When you run a test in this way, you must specify all test options in the form of command line arguments.

There are two types of command line arguments:

■ Standard arguments—common to all tests. Refer to TABLE 1-3 for details.
■ Test specific arguments—unique to a specific test. Refer to the test specific chapters in this book for details.

The standard syntax for all SunVTS tests is:

```
testname [-scruvdtelnf] [-p number][-i number] [-w number]
        [-o test specific arguments]
```

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

Standard Command Line Arguments

The following table defines the standard SunVTS command line arguments:

TABLE 1-3 Standard Command-line Arguments

Argument	Definition
-s	Run a test as though it were invoked from the SunVTS kernel (`vtsk`). This causes the test to send output to an RPC port that the SunVTS kernel is using. The default is to send the output to `stdout` or `stderr`.
-c	Enables a core image of the test process to be created in the current working directory upon receipt of certain signals, otherwise those signals are caught and handled to prevent a core from being generated. The default is to disable the creation of a core image.
-r	Enables run on error so that when an error occurs, the test continues with the next test sequence instead of exiting. The default is false.
-u	Displays command line usage information.
-v	Runs the test in Verbose mode and displays messages with more detailed information about the testing process. The default is false.
-d	Runs the test in debug mode and displays messages to help programmers debug their test code. The default is false.
-t	Runs the test in test Trace mode and displays messages that track function calls and sequences currently in use by the test code. The default is false.
-e	Runs the test in Stress Mode by increasing the system load. The default is false.
-l	Runs the test in Online Functional mode. This is the same mode that tests run in when executed through SyMON. It is a non-intrusive version that will not significantly affect other applications. See the note below. The default is true.
-n	Runs the test in Connection Mode. See the note below. The default is false.

TABLE 1-3　　Standard Command-line Arguments

Argument	Definition
-f	Runs the test in full Functional test mode. This mode assumes that the test has complete control of the device under test. See the note below. The default is false.
-i *number*	Defines the number of instances for scalable tests.
-p *number*	Defines the number of passes.
-w *number*	For scalable tests, defines to which instance the test is assigned.
-o	Used to indicate that the options and arguments that follow are test specific. Note – separate each test specific argument by commas, with no space after each comma.

Note – If you choose to specify a test mode with the l, n, or f option, specify only one option at a time because only one test mode can be selected at a time.

Test-Specific Arguments

There are test-specific arguments, as described in TABLE 1-4 Test-specific arguments follow the format specified in the *getsubopt*(3c) man page. For information about test-specific arguments refer to the specific test chapter in this book.

TABLE 1-4　　SunVTS Test-Specific Arguments

Argument	Definition
-o	Separate each test-specific argument by commas, with no space after the each comma. For example: `#./sample -v -o dev=/dev/audio,volume=78` The test option format is specified by the man page `getsubopt`(3C).

To Test Multiple Frame Buffers

The following rules apply when you test multiple frame buffers (displays) simultaneously:

- You can test multiple frame buffers on a system at the same time, but only one frame buffer can run the window environment.

- To avoid incorrect test failures, for those frame buffer tests that have a window locking option, the frame buffer that runs the window environment, such as CDE or OPEN LOOK software, must have window locking enabled. Any other frame buffers must have window locking disabled.

Caution – If window locking is disabled (unlocked) on frame buffers that are running, the SunVTS tests can return false error messages if you move the mouse during testing. Even a slight mouse movement can cause a test to fail.

- By default, SunVTS enables window locking on the console monitor (frame buffers that are the /dev/fb named device).

- If you are running a frame buffer test from a command line, you can disable Window Locking by specifying a command line argument (see the test command line descriptions in this manual). For example, when running the generic frame buffer test (fbtest), use the lock=e/d option to enable or disable Window Locking. The example below shows the command that enables Window locking (frame buffer locking):

```
#./fbtest -o dev=cgthree0,lock=e
```

Remote Testing of Frame Buffers

The frame buffer locking option does not work when you start sunvts or vtsk remotely. In this case, disable the window locking option to d. Do not run any graphic programs (including vtsui) on that frame buffer during graphic testing.

Advanced Frame Buffer Test (`afbtest`)

`afbtest` verifies the functionality of the Advanced Frame Buffer.

`afbtest` can detect and adapt to the various video modes of the advanced frame buffer (AFB). Instead of only running in one standard graphics mode, all tests can run in any mode. In stereo mode, all tests write into the right and left eyes unless you specify otherwise.

You can interrupt `afbtest` using Control-C. Turn off all other keyboard input if OPEN LOOK is running on the unit being tested.

Test accuracy is checked using a checksum algorithm. Possible locations of failing pixels are colored chartreuse to help visually identify their position.

Caution – Do not run any other application or screen saver program that uses the AFB accelerator port while running `afbtest`. This combination causes SunVTS to return incorrect errors.

Note – Disable all screen savers before testing any graphics device. Type **xset s off** at a UNIX prompt to disable the Solaris screen saver.

Note – DO NOT run Open Windows across multiple monitor while running `afbtest`, otherwise the test will return errors.

Note – `afbtest` requires approximately 29 Mbytes of disk space in the `/tmp` directory to extract its working files. If this space is not available, the diagnostic will fail and report warning and error messages, indicating a lack of disk space.

`afbtest` Options

By default, all `afbtest` options are enabled.

Elite3D Graphics

Options :

3DRAM:	Enable | Disable
3DRAM Logic:	Enable | Disable
RAMDAC:	Enable | Disable
Micro code:	Enable | Disable
Rendering Pipeline:	Enable | Disable
Fast Fill/Vertical Scroll:	Enable | Disable
Pixel Processor:	Enable | Disable
AFB Dots:	Enable | Disable
AFB Lines:	Enable | Disable
AFB Triangles:	Enable | Disable
Lighting:	Enable | Disable
Texture Processor:	Enable | Disable
AFB Mix:	Enable | Disable
Picking:	Enable | Disable
Arbitration:	Enable | Disable
Stereo:	Enable | Disable
UART:	Enable | Disable

Subtest Repeat: 1___ 1 ▢════════ 99

Test Loop Repeat: 1___ 1 ▢════════ 99

Pattern: ▽ Random

Processor Affinity: ▽ None

(Reset)　(Apply)

FIGURE 2-1 `afbtest` Option Menu

TABLE 2-1 `afbtest` Options

`afbtest` **Options**	**Description**
3DRAM Test	The 3DRAM test thoroughly tests the video memory in the AFB using 512-bit reads and writes. 3DRAM makes a full screen pass, consisting of a write and a read to each pixel location, for each access mode on the list below. The data used can be either random or specified by the user. A second pass is made with the one's complement of the data used in the first pass so that each memory location is tested with both a zero and a one.
	Errors in this subtest are attributes to the 3DRAM. A failing chip is indicated by (x, y) locations and device-specific "U" numbers.
	• DFB8R, DFB8G, DFB8B, DFB8X - Buffer A • DFB24 - Buffer A • DFB32 - Buffer A • SFB8R, SFB8G, SFB8B, SFB8X - Buffer A • SFB8R, SFB8G, SFB8B, SFB8X - Buffer B • SFB32 - Buffer A • SFB32 - Buffer B • SFB32 - Buffer C • SFB64 - Buffers A and C • SFB64 - Buffers B and C

TABLE 2-1 `afbtest` Options

`afbtest` **Options**	**Description**
3DRAM Logic Test	The 3DRAM Logic test provides logical functionality to the AFB. The following services are tested: • Compare Controls - Match AB • Compare Controls - Magnitude AB • Compare Controls - Match C • Compare Controls - Magnitude C • Match Mask - AB • Magnitude Mask - AB • Match Mask - C • Magnitude Mask - C • Raster Operations - RGB • Raster Operations - X • Raster Operations - YZ • Plane Mask - RGB • Plane Mask - X • Plane Mask - Y • Plane Mask - Z • Group Enable - R, G, B, X • Group Enable - Y, Z Each function is tested separately with a series of SFB64 writes. A total of 16 writes are made for each different test case with Y coordinate values varying from 0 to 30 in increments of 2 pixels. This dotted column organization provides page thrashing and block flashing in all screen resolutions. For each operation all possible combinations are tested. For example, in ROP RGB *new==old* there are three possible values: *new < old, new == old, and new > old.* Each of these cases are tested. Five passes of the functions are made. Each pass writes into a different AFB address space: SFB32-A, SFB32-B, SFB32-C, SFB64-AC, and SFB64-BC. Note that the passes that write into the SFB32 address spaces are writing two pixels at a time because the tests use SFB64 writes. Care is taken to ensure that all 3DRAM chips are tested. Errors in this subtest are attributed to the 3DRAM.

TABLE 2-1 `afbtest` Options

afbtest **Options**	Description
RAMDAC Test	RAMDAC registers are tested using simple read/write patterns to determine if there are any bad bits. This includes all LUTs (4 CLUTs, PWLUT and OWLUT). `afbtest` ensures that data is actually being read from the RAMDAC and not being supplied by the driver.
	RAMDAC on AFB can be in SEP8 or Combined mode. RAMDAC test detects the RAMDAC mode and tests the RAMDAC output for that mode. The RAMDAC Signature Register captures the pixels going to the screen. This test determines that all of the different data paths within the RAMDAC are functioning properly.
	The data pattern is designed so all the data paths are tested. i.e., All CLUTs, PWLUTs and OWLUTS. A cursor is also displayed on the screen.
	Errors in this test are attributed to the RAMDAC.
Microcode Test	Micro code test generates the checksum for the microcode of the each enabled float and compares all the check sums for equality.
	Errors in this test are attributed to the Microcode PROMS & SRAMS.
Rendering Pipeline Test	Rendering Pipeline uses the rendering pipeline tests developed for the FFB stand-alone diagnostics. Each FFB primitive is tested thoroughly with a variety of sources and configurations.
	• Dots • Anti-aliased dots • Lines using all four line drawing primitives • Triangles • Polygons • Rectangles • Fonts
	Errors in this test are attributed to the Draw Chips.
Fast Fill/Vertical Scroll Test	Fast Fill/Vertical Scroll primitives are separated from the Rendering Pipeline tests because of their dependence on screen type. There are three different tests, one for each screen type. Each test uses both block and page mode fast_fills.
	Errors in this test are attributed to the Draw Chips.

TABLE 2-1   `afbtest` Options

`afbtest` **Options**	**Description**
Pixel Process Test	The Pixel Processor test, a subtest, exercises the options selected by the AFB's Pixel Processor Control (PPC) register. • Auxiliary clipping (additive and subtractive) • Depth cueing • Alpha blend • Viewport clip (2D and 3D) • Area pattern (transparent and opaque) Errors in this test are attributed to the Draw Chips.
AFB Dots Test	This test uses the AFB primitive tests developed for the AFB stand-alone diagnostics. AFB Dots are tested thoroughly with a variety of sources and configurations. • Dots • Anti-aliased dots • Big dots Errors in this test are attributed to the Command & Draw Chips.
AFB Lines Test	This test uses the AFB primitive tests developed for the AFB stand alone diagnostics. AFB Lines are tested thoroughly with a variety of sources and configurations. • Jaggy lines • Anti-aliased lines • Lines with patterns • Bresenham Lines • Wide Lines drawn as lines and Triangles Errors in this test are attributed to the Command & Draw Chips.
AFB Triangles Test	This test uses the AFB primitive tests developed for the AFB stand alone diagnostics. AFB Triangles are tested thoroughly with a variety of sources and configurations. • Triangles drawn clock wise & counter clockwise • Triangles drawn as stripes • Independent Triangles • Triangles drawn as stars • Triangles with facet normals Errors in this test are attributed to the Command & Draw Chips.
Lighting Test	The Lighting test exercises AFB float and lighting microcode. This test lights an object with maximum number of lights (32) that AFB can handle in hardware. A check sum is generated for the rendered image and compared with the check sum generated for the same image on a known good system. Errors in this test are attributed to the Float & Microcode SRAMS.

TABLE 2-1   `afbtest` Options

`afbtest` **Options**	Description
Texture Processor Test	The Texture Processor test exercises the different options of the AFB's Texture Pixel Processor Control (TPPC) register. • Texture Minification • Texture Magnification • Blend • Decal • Modulation Errors in this test are attributed to the Draw Chips.
AFB Mix Test	The AFB Mix test draws different primitives with variety combinations of sources and configurations, exercising all the Draw, Float, Microcode and 3DRAM chips on AFB. This test is to stress the AFB. Errors in this test are attributed to Draw, Float, Microcode and/or 3DRAM Chips.
Picking Test	The Picking test exercises the pick detect login of the 3DRAM. We define a pick detect window and make sure that writes to the window are picked, and writes outside the window are not picked. The test is repeated once for each 3DRAM. Errors in this test are attributed to the 3DRAM.
Arbitration Test	The Arbitration test, a subtest, continuously renders an object into the accelerator port while doing reads and writes through the direct port. A picture is rendered into all 32 planes of the B buffer while the other process does 32-bit DFB reads and writes in the A plane. This subtest simulates conditions in the real world, where rendering processes and windows operations run concurrently. Errors in this test are attributed to the Context switching between DFB and SFB.

TABLE 2-1   `afbtest` Options

`afbtest` **Options**	**Description**
Stereo Test	Stereo test displays an object in stereo mode with different images for the right and left eye. The user can verify proper operation by looking at the screen with stereo glasses and following the instructions being displayed. If the monitor type is not 1280x1024 @ 76MHz, this test prints a warning message and does not execute. To prevent this message from being displayed or written to the SunVTS information log, disable the stereo test in the test option menu. Only Sony P4 and N2 monitors support stereo resolutions. This test temporarily switches the monitor into stereo mode, renders a stereo image, performs a signature analysis on the stereo image (using the RAMDAC signature capture register), and after displaying the image for five seconds, restores the monitor to its previous resolution.
	Errors in this test are attributed to the RAMDAC.
UART Test	The UART test tests both UART0 and UART1. First, UART memory is tested using simple read/write patterns to determine if there are any bad bits. Then data is written to UART 0/1 and the written data is read using the internal loopback in polling mode. The read data is verified with written data. Errors in this test are attributed to UART and its SRAM memory chip.

`afbtest` Test Modes

Due to the nature of graphic tests, reading data from, or writing data to the frame buffer during graphic tests will disturb user operation. For this reason, `afbtest` is only available in Functional mode.

`afbtest` Command Line Syntax

/opt/SUNWvts/bin/afbtest *standard_arguments* **-o dev=**device_name**,**
S=subtest_number**,F=**#_of_subtest_loops**,B=**#_of_test_loops**,P=**test_pattern

TABLE 2-2 `afbtest` Command Line Syntax

Argument	Explanation
`dev=`*device_name*	*device_name* is the relative path name of the device being tested with respect to `/dev/fbs`; The default is `afb0`.
`S=`*subtest_number*	*subtest_number* is the test number of the subtest to be run. Select from the subtests below. You can run multiple subtests by adding the subtest numbers together. For example, *n=0x3* runs both test 1 and test 2; *n=0x180* runs both test *0x080* and test *0x0100*. You do not need the leading zeros. *n* – `0x00001 3DRAM` *n* – `0x00002 3DRAM Logic` *n* – `0x00004 RAMDAC` *n* – `0x00008 Micro code` *n* – `0x00010 Rendering Pipeline` *n* – `0x00020 FastFill/Vertical Scroll` *n* – `0x00040 Pixel Processor` *n* – `0x00080 AFB Dots` *n* – `0x00100 AFB Lines` *n* – `0x00200 AFB Triangles` *n* – `0x00400 Lighting` *n* – `0x00800 Texture Processor` *n* – `0x02000 AFB Mix Test` *n* – `0x04000 Picking` *n* – `0x08000 Arbitration` *n* – `0x10000 Stereo` *n* – **ox40000 UART**
`F=`*#_of_subtest_loops*	The number of times to repeat each subtest; the default is 1.
`B=`*#_of_test_loops*	The number of times to repeat a test loop before passing; the default is 1.
`P=`*test_pattern*	The test pattern number. The default is **r**, for random patterns. You may also choose `0` **for** `0x0000000`, `3` **for** `0x3333333`, `5` **for** `0x5555555`, or `9` **for** `0x9999999`.

`afbtest` Error Messages

Errors returned by `afbtest` are nonspecific: It is not possible to determine which component caused a failure. In all error conditions, the field replaceable unit (FRU) is the entire AFB.

TABLE 2-3   `afbtest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	*string* `error: 3DRAM Memory error` *string* `Plane at` *(data,data)* `Expected = 0x`*%02x*`, actual = 0x`*%02x*`, xor = 0x`*%02x*	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6002	*string* `error: 3DRAM Memory error` *string* `Plane at` (*data*,*data*) `Expected = 0x%06x, actual = 0x%06x, xor = 0x%06x`		
6004	*string* `error: 3DRAM Memory error` *string* `Plane at` (*data,data*) `Expected = 0x%08x, actual = 0x%08x, xor = 0x%08x`		
6006	*string* `error: 3DRAM Memory error` *string* `Mode, Plane A at` (*data*,*data*) `Expected = 0x%08x, actual = 0x%08x, xor = 0x%08x`	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.

TABLE 2-3   `afbtest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6008	*string* error: 3DRAM Memory error *string* Mode, Plane B at (*data*,*data*) Expected = 0x%08x, actual = 0x%08x, xor = 0x%08x	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6010	*string* error: 3DRAM Memory error *string* Mode, Plane C at (*data*, *data*) Expected = 0x%08x, actual = 0x%08x, xor = 0x%08x		
6012	Misaligned raster	Software error.	Reinstall software or contact your system administrator.
6013	File Not Found Could not open file *string*		
6014	Could not open file *string*: *string*		
6015	Keyword not Found, line *data*		
6016	Miscompares, starting at *string*	Faulty frame buffer.	Contact your authorized Sun service representative.
6017	Picking test error. Got a pick hit with picking disabled		
6018	Picking test error. Got a pick hit outside the picking area		
6019	Picking test error. Should have detected a pick hit		
6020	Out of memory	Software System Error.	Contact your system administrator.
6021	*string* failed	Software error.	Reinstall software or contact your system administrator. Check for existence and/or permissions on *string*. Also check for available space in /tmp.

TABLE 2-3   afbtest Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6022	*string* failed	Software error.	Reinstall software or contact your system administrator. Check for existence and/or permissions on *string*.
6023	fork1(2) failed	Software error.	Reinstall software or contact your system administrator.
6024	execl(2) failed		
6025	Unable to open *string*	Software error.	Reinstall software or contact your system administrator. Check device for existence and/or permissions on *string*.
6026	ioctl(2) failed trying to read CLUT.	Possible hardware failure, more likely a software driver error.	Reinstall software first, if failure continues, see a System Administrator or your Sun Service provider.
6027	ioctl(2) failed trying to turn on AFB video.	Possible hardware failure, more likely a software driver error.	Reinstall software first, if failure continues, see a System Administrator or your authorized Sun service provider.
6028	ioctl(2) failed trying to post CLUT.		
6029	Miscompare found in Green channel of CLUT entry *data*. Expected = 0x%02x, Observed = 0x%02x, XOR = 0x%02x.	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6030	Miscompare found in Green channel of CLUT entry *data*. Expected = 0x%02x, Observed = 0x%02x, XOR = 0x%02x.		
6031	Miscompare found in Blue channel of CLUT entry *data*. Expected = 0x%02x, Observed = 0x%02x, XOR = 0x%02x.		
6032	ioctl(2) failed trying to read WID LUT.	Possible hardware failure, more likely a software driver error.	Reinstall software first, if failure continues, see a System Administrator or your authorized Sun service provider.
6033	ioctl(2) failed trying to post WID LUT.		
6034	ioctl(2) failed trying to get RAMDAC signature analysis.		

TABLE 2-3 afbtest Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6035	Miscompare found in WID LUT entry *data*. Expected = 0x%06x, Observed = 0x%06x, XOR = 0x%06x.	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6036	RAMDAC test failed. The on-screen image is incorrect.		
6037	sysinfo(2) failed getting hostname.	Software error.	Must be superuser
6038	can't acquire console semaphore	Software error.	Reinstall software or contact your system administrator.
6039	Cannot gain control of mouse: *string*		
6040	Cannot gain control of keyboard: *string*		
6046	ioctl(2) failed trying to get current video mode.	Possible hardware failure, more likely a software driver error.	Reinstall software first, if failure continues, see a System Administrator or your authorized Sun service provider.
6047	ioctl(2) failed trying to set current video mode.		
6048	ioctl(2) failed trying to get AFB system info.		
6049	ioctl(2) failed trying to go into Diagnostic mode.		
6050	mmap(2) failed: *string*.	Software system error.	Reinstall software or contact your system administrator.
6051	munmap(2) failed: *string*.		
6052	close(2) failed: *string*.		
6053	DGA error grabbing window on screen *data*.		
6054	DGA error getting window from token on screen *data*.		

TABLE 2-3 afbtest Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6055	`ioctl(2) failed trying to read Transparent Overlay registers.`	Possible hardware failure, more likely a software driver error.	Reinstall software first, if failure continues, see a System Administrator or your authorized Sun service provider.
6056	`ioctl(2) failed trying to post Transparent Overlay registers.`		
6057	`ioctl(2) failed trying to post cursor.`		
6058	`ioctl(2) failed trying to read cursor.`		
6059	`No AFB device found on display 0.`	The environmental variable DISPLAY is set wrong or the display being used is not using an AFB.	Set the variable to a display that has the AFB plugged into it.
6060	`There is an inconsistency with the version of the X Window System.`	Software error.	Contact your system administrator.
6061	`stat(2) failed trying to get info about Device Under Test.`	Software error.	Reinstall software or contact your system administrator.
6062	`stat(2) failed trying to get info about screen 0:`*data*`.`		
6063	`Can't access XWindows.`	Software error.	Try running xhost(1) or contact your system administrator.
6064	`wait(2) failed:` *string*	Software error.	Reinstall software or contact your system administrator.
6065	`signal(2) failed:` *string*		
6066	`thr_kill(2) failed`		
6067	`thr_join(2) failed:` *string*		
6071	`To get signatures, run afbtest from the command line.`		
6072	`Stereo image incorrect`	Faulty frame buffer.	If the problem persists, contact your Sun Service provider.

TABLE 2-3 `afbtest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6075	System LWP limit exceeded.	Software error.	Reinstall software or contact your system administrator.
6076	Error *data* creating thread.		
6104	Error in getting the length of the file. *string*		
6105	Stencil test failed *string*	Faulty 3DRAM.	If the problem persists, contact your Sun Service provider.
6106	Passin test failed *string*		
6107	Blend2 test failed *string*		
6108	Couldn't create lock file *string*	No permission or lack of space in /tmp.	Free some space in /tmp directory. Check permissions of /tmp.
6115	*string* error: GCopy error *string* at (*data*,*data*) Expected = 0x%02x, actual = 0x%02x, xor = 0x%02x.	Faulty frame buffer.	If the problem persists, contact your Sun Service provider.
6116	*string* error: GCopy AFB to Memory error *string* at (*data*,*data*) Expected = 0x%02x, actual = 0x%02x, xor = 0x%02x.		
6117	*string* error: GCopy Memory to AFB error *string* at (*data*,*data*) Expected = 0x%02x, actual = 0x%02x, xor = 0x%02x.		
6208	ioctl(2) failed trying to post Overlay WID LUT.	Possible hardware failure, more likely a software driver error.	Reinstall software first, if failure continues, see a System Administrator or your authorized Sun service provider.
6209	ioctl(2) failed trying to post Primary WID LUT.		
6210	Miscompare found in Overlay WID LUT entry *data*. Expected = 0x%06x, Observed = 0x%06x, XOR = 0x%06x.	Faulty frame buffer.	If the problem persists, contact your Sun Service provider.
6211	Miscompare found in Primary WID LUT entry *data*.		

TABLE 2-3   `afbtest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6212	`ioctl(2) failed trying to read CLUT #` *data*.	Possible hardware failure, more likely a software driver error.	Reinstall software first, if failure continues, see a System Administrator or your authorized Sun service provider.
6213	`ioctl(2) failed trying to post CLUT #` *data*.		
6214	`Miscompare found in Red channel of CLUT #` *data* `in entry` *data*. `Expected = 0x%02x, Observed = 0x%02x, XOR = 0x%02x.`	Faulty frame buffer.	If the problem persists, contact your Sun Service provider.
6215	`Miscompare found in Green channel of CLUT #` *data* `in entry` *data*. `Expected = 0x%02x, Observed = 0x%02x, XOR = 0x%02x.`		
6216	`Miscompare found in Blue channel of CLUT #` *data* `in entry` *data*. `Expected = 0x%02x, Observed = 0x%02x, XOR = 0x%02x.`	Faulty frame buffer.	If the problem persists, contact your Sun Service provider.
6217	`RAMDAC test failed. The on-screen image in sep 4 mode is incorrect.`		
6218	`RAMDAC test failed. The on-screen image in sep 8 mode is incorrect.`		
6219	`RAMDAC test failed. The on-screen image in combined mode is incorrect.`		
6220	`ioctl(2) failed trying to read Overlay WID LUT.`	Possible hardware failure, more likely a software driver error.	Reinstall software first, if failure continues, see a System Administrator or your authorized Sun service provider.
6221	`ioctl(2) failed trying to read Primary WID LUT.`		
6222	Couldn't bind afbtest and its threads to processor# *data*.	User does not have permission or System error.	If the problem persists, contact you system administrator, or an authorized Sun service provider.

TABLE 2-3 `afbtest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6230	L-core checksum mismatch, L-core Checksum of Float #*data* = 0x*value*	Possible hardware failure.	If the problem persists, contact your Sun Service provider
6231	S-core checksum mismatch, S-core Checksum of Float #*data* = 0x*value*		
6232	F-core checksum mismatch, F-core Checksum of Float #*data* = 0x*value*		
6233	F-core SRAM error in Float #*data*, Failing address = 0x*value*, expected data = 0x*value*, actual data = 0x*value*, XOR of expected and actual data = 0x*value*, F-core checksum = 0x*value*		
6234	L-core SRAM error in Float #*data*, Failing address = 0x*value*, expected data = 0x*value*, actual data = 0x*value*, XOR of expected and actual data = 0x*value*, L-core checksum = 0x*value*		
6235	S-core SRAM error in Float #*data*, Failing address = 0x*value* expected data = 0x*value*, actual data = 0x*value*, XOR of expected and actual data = 0x*value*, S-core checksum = 0x*value*		
6300	Couldn't reset UART*n*	Faulty UART or Some other application/test is using the UART*n*	Try stopping other applications/tests that use the UART. If the problem persists, contact your authorized Sun service provider.

TABLE 2-3 `afbtest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6301	UART memory test failed. Error occurred in memory of UART*n*. data at location 0x*value* expected 0x*value* = *data* actual data = *data*	Faulty UART. Or some other application/test is using the UART.	Try stopping other applications/tests that use the UART. If the problem persists, contact your authorized Sun service provider.
6302	Couldn't baud rate of UART*n* *data* to *data*.		
6303	UART loopback test failed. Loopback cable is missing on UART n.		
6304	UART loopback test failed. Couldn't send data through UART*n*.		
6305	UART loopback test failed. TxRdy not ready. Timed out while polling UART*n*.		
6306	UART loopback test failed. Buffer pointer of UARTn data didn't advance.		
6307	UART loopback test failed. Couldn't read data from UART*n*.		
6308	UART Fill Buffer test for UART*n* failed. Transmitter not ready.		
6309	UART Fill Buffer test for UART*n* failed. Data mismatch at memory location ox%08*x*.		
8000	Real user ID of current instance of afbtest is not root.	Not logged in as root.	Log in as root. If the problem persists, contact your authorized Sun service provider.

SunATM Adapter Test (`atmtest`)

`atmtest` checks the functionality of the SunATM-155 and SunATM-622 SBus and PCI bus adapters.

It runs only in loopback (external or internal) mode. The Asynchronous Transfer Mode (ATM) adapter, and ATM device driver must be present. To run the `atmtest` in external loopback mode, a loopback connector must be attached to the ATM adapter. The internal loopback mode does not require a loopback connector.

`atmtest` uses DLPI RAW mode to talk to the device driver. It establishes a virtual circuit (VC) to send a message, receive a message, and compare messages. If the message does not match, or the message is out of sequence, it displays an error message.

Using a random number generator, `atmtest` sends data into a data buffer and then sends each message from a different starting point. This assures that no two consecutive messages are the same.

`atmtest` can test more than one virtual circuit. The more virtual circuits used increases the stress level of the test. `atmtest` automatically selects the virtual circuit number which is unique to the test. `atmtest` is nonscalable as it provides for more than one virtual circuit to be tested for each instance, which is adequate for the purpose of a stress test.

`atmtest` can only be selected when the Intervention mode is enabled since it requires a loopback connector for external loopback testing. While Intervention mode is enabled, `atmtest` and `nettest` are both available as default selections, however, you must deselect `nettest` when testing the ATM device.

Note – Do not run `nettest` while running `atmtest`.

Note – Bring the ATM interface down to make sure that the interface is in offline mode before running `atmtest`.

atmtest Options

```
┌─────────────────────────────────────────────────────┐
│  ──      ba0(atmtest) Test Parameter Options         │
│                                                       │
│  Configuration:                                       │
│      Host_Name: ss5-4-net191                          │
│      Host Address: 172.18.191.34                      │
│      Host ID: 80720852                                │
│      Domain Name: nettest.eng.sun.com                 │
│                                                       │
│  Options:                                             │
│         Total_packets:  ▲▼ 10000                      │
│                                                       │
│        Number_of_VC:    ▲▼ 2                          │
│                                                       │
│         MAX_PKT_LEN:    ▲▼ 9140                       │
│                                                       │
│      Outstanding_Pkts:  ▲▼ 4                          │
│                                                       │
│          First_VC_no:   ▲▼ 30                         │
│                                                       │
│            Bandwidth:   ▲▼ 14                         │
│                                                       │
│            Loopback:  ◉ External   ◯ Internal         │
│                                                       │
│        Print_Warning:   ◯ Enable  ◉ Disable           │
│                                                       │
│             Instance:   ▲▼ 1                          │
│  ───────────────────────────────────────────────     │
│      Apply          Reset           Cancel            │
└─────────────────────────────────────────────────────┘
```

FIGURE 3-1 `atmtest` Option Menu

TABLE 3-1 `atmtest` Options

Options	Description
Configuration	The Post Address, Host ID, and Domain Name of the system being tested.
Total Packets	The total number of packets sent. The default number of packets sent is 10000.
Number of VC	The number of virtual circuits to be set up by one instance. The default number of virtual circuits is 2for each instance. The `atmtest` uses these two virtual circuits to send out messages simultaneously. The message is received in sending order.
Loopback	Enables the user to select either the external loopback field or internal loopback field. The default selection is the external loopback field. A loopback connector is only needed for external loopback testing.
MAX_PKG_LEN	The maximum packet length to be used by the test to send out the data. The default number is 9140.
Outstanding_pkts	Describes the maximum number of outstanding packets. `atmtest` stops sending messages when the outstanding packet count is more than the number of packets this field specifies.
First_VC_no	Enables the user to set up the starting virtual circuit number to be used for each `atmtest` instance. `atmtest` can automatically avoid virtual circuit numbers that have already been used.
Bandwidth	Enables the user to select different bandwidths to test. The default number is 14.

`atmtest` Test Modes

`atmtest` can only be run in Functional test mode.

`atmtest` Command Line Syntax

/opt/SUNWvts/bin/atmtest *standard_arguments* **-o**
dev=*device***,tpkts=***n***,nv=***n***,ml=***n***,bw=***n***,opkts=***n,ld,sd,sl,nc,ns,vcf*

TABLE 3-2 `atmtest` Command Line Syntax

Argument	Explanation
dev=*device*	Specifies the device name to be tested, such as `ba0` or `sa0`.
tpkts=n	Specifies the number of packets to loopback [1.2147483647].
nv=num_vc	Specifies the number of simultaneous virtual circuits to be tested.
ml=max_len	Specifies the maximum length of the random packet.
bw=bandwidth	Specifies the bandwidth in MBits/s of a virtual circuit.
opkts=n	Specifies the number of packets for each virtual circuit that can be transmitted without receiving a corresponding packet.
lb	The internal loopback mode is selected.
sd	Changes the payload data to static instead of random.
sl	Changes all packets to their maximum length.
nc	Instructs the test not to check the receive payload (Improves throughput).
ns	Instructs the test not to exit on a packet reception failure.
vcf=n	Specifies the first virtual circuit number used.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If the test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`atmtest` Error Messages

The `atmtest` error messages are listed below with explanations or probable causes:

TABLE 3-3   `atmtest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	`putmsg failed, errmsg=`*data*		
6001	`getmsg failed, errmsg=`*data*		
6002	`wrong vc: exp:` *number*, `obs:` *number*		
6003	`VC` *number* `bogus pkt, seq: exp=`*number*`, obs=`*number*`; len: exp=`*number*`, obs=`*number*		
6004	`VC` *number* `bad pkt len, EXP:` *number*`, OBS:` *number*`, seq=`*number*		
6005	`VC number memcmp error`		
6006	`VC number failed to rcv a packet`		
6007	`Complete Usage:` *string* `u`		
6008	*string*`: alarm`		
6009	*string*`: getmsg`		
6010	`receive` *string* `for` *string* `with` *string* `error`		
6011	`DL_OK_ACK was not M_PCPROTO`		
6012	`short response ctl.len:` *number*		
6013	`ctl.len sizeof (dl_ok_ack_t):` *number*		
6014	`receive` *string* `for string with` *string* `error`		
6015	`DL_BIND_ACK was not M_PCPROTO`		
6016	`ioctl DLIOCRAW failed, errmsg` *string*		

TABLE 3-3 atmtest Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6017	ioctl DL_IOC_HDR_INFO failed, errmsg *string*		
6018	len=*number* hdrmax=*number*		
6019	ioctl ADDVC failed, errmsg *string*		
6020	ioctl ALLOCBW failed, errmsg *string*		
8000	open device failed, errmsg=*string*		
8001	sa_add_vpci failed, errmsg=*string*		
8002	sa_allocatebw failed, errmsg=*string*		
8003	atm_attach failed, errmsg=*string*		

Audio Test (`audiotest`)

The `audiotest` verifies the hardware and software components of the `audio` subsystem. This test supports all Sun `audio` implementations.

The `audio` device is an *exclusive use device*. Only one process or application can interface with it at a time. This test is not scalable.

The availability of the following subtests depends on the particular `audio` implementation being tested.

`audiotest` Subtests

TABLE 4-1   `audiotest` Subtests

Subtest	Description
Record/Play Test	This test plays and records one second of data. It does not check data. This test is run on all audio implementations.
Crystal Test	The crystal test measures the accuracy of the crystal that generates the sample rate clock. It does this by playing a one-second signal and then measuring the actual time required to play the signal. This measurement is performed for each of the eight standard sample rates. This test is available for `dbri`(7) and `audiocs`(7) audio implementations
Loopback Tests	This test verifies the functionality and signal quality of the audio ports. The test simultaneously plays and records a known signal. The recorded signal is analyzed for loop gain and signal-to-noise ratio plus distortion. This is repeated at various sample rates, encodings, precisions and channels. The audio ports that are supported depend on the audio implementation under test. The `audiocs(7)` implementation supports loopbacks from/to headphone, line-out, microphone, and line-in ports. The `dbri(7)`/speakerbox implementation supports fewer ports. The `audioamd(7)`implementation does not support loopback tests. Most tests require a stereo loopback cable. NOTE: The microphone loopback tests require special hardware and are used by manufacturing centers and special test facilities. Do not invoke the microphone loopback tests unless you have the required hardware.
Controls Test	This test verifies the three control buttons on the Sun Speakerbox. The Controls test plays music while the user is prompted to press the Volume Down, Volume Up, and Mute buttons in a specific order. If no button is pressed in 30 seconds the test fails. This test is only supported on the `dbri`(7)/speakerbox implementation.
Audio Test	This test plays a 30-second music file out of the speaker or headphone. The full benefit of this test is only realized if the user listens to the output. Badly distorted audio, or not audible music indicates a problem. This test is supported on all audio implementations.

`audiotest` Options

Upon start-up, the SunVTS probe utility determines which audio implementation is present and adjusts the `audiotest` option menu appropriately. The possible option menus are as follows:

```
┌─────────────────────────────────────────────┐
│        sound0(audio) Test Parameter Options  │
│                                              │
│   Configuration:                             │
│       Audio Device Type: AMD79C30            │
│                                              │
│   Options:                                   │
│       Audio Output:   ◈ speaker   ◇ headphone│
│                                              │
│          Volume:   [▲ ▼ 80        ]          │
│                                              │
│       Audio Test:   ◇ disable   ◈ enable     │
│                                              │
│   ─────────────────────────────────────────  │
│                                              │
│      [ Apply ]    [ Reset ]    [ Cancel ]    │
│                                              │
└─────────────────────────────────────────────┘
```
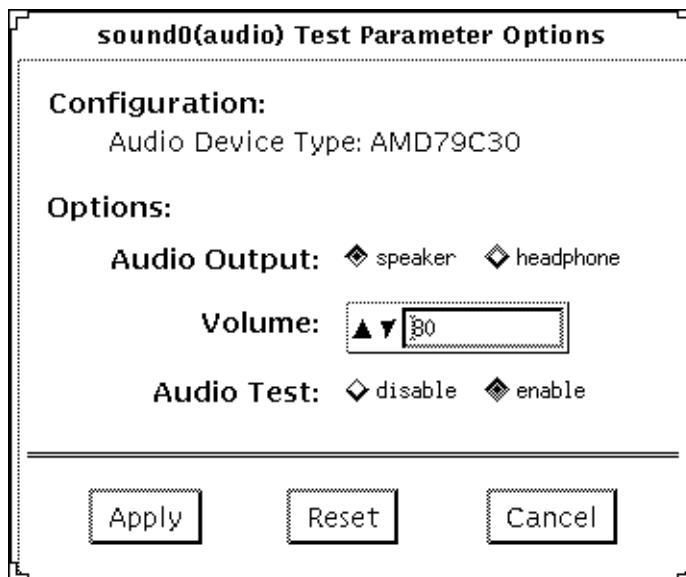
FIGURE 4-1 `audiotest` Test Option Menu for `audioamd(7)`

The `audioamd(7)` is an 8-bit mono, telephone-quality, audio device.
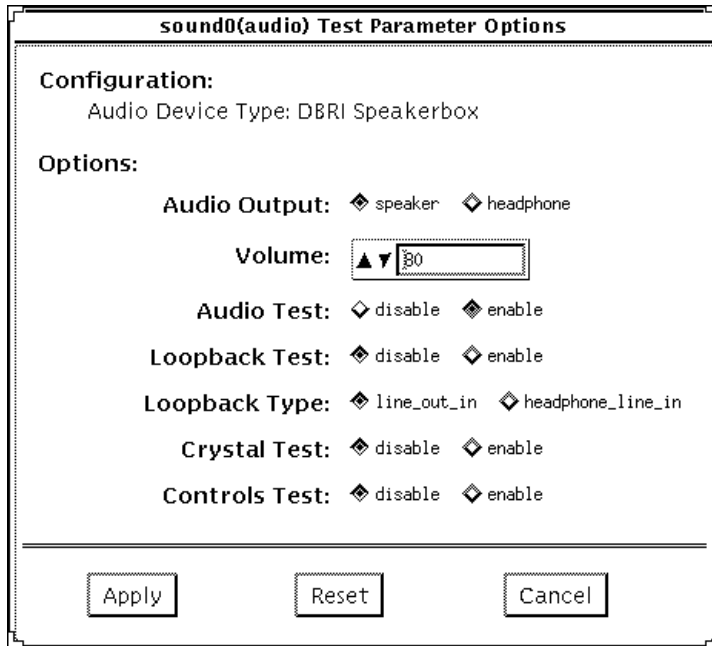
FIGURE 4-2 `audiotest` Test Option Menu for Speakerbox `dbri(7)` audio

The speakerbox `dbri` is a high-quality 16-bit stereo multimedia (coder/decoder (codec).
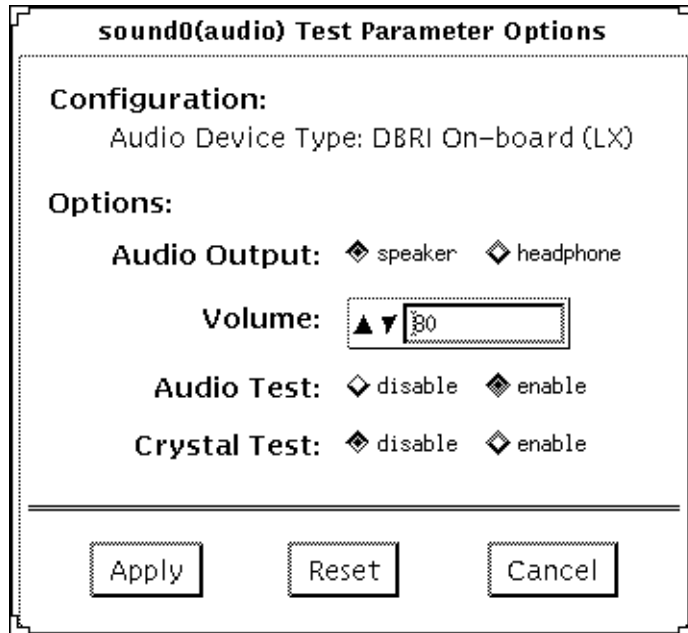
FIGURE 4-3 `audiotest` Test Option Menu for SPARCstation LX `dbri(7)` without a Speakerbox attached

The on-board DBRI interface is a high-quality, 16-bit stereo multimedia codec.

```
┌─────────────────────────────────────────────────────────────┐
│              sound0(audio) Test Parameter Options             │
├─────────────────────────────────────────────────────────────┤
│                                                               │
│  Configuration:                                               │
│      Audio Device Type: DBRI On-board                         │
│                                                               │
│  Options:                                                     │
│                                                               │
│              Audio Output:  ◈ speaker   ◇ headphone           │
│                                                               │
│                   Volume:   ▲ ▼ 80                            │
│                                                               │
│               Audio Test:  ◇ disable   ◈ enable               │
│                                                               │
│            Loopback Test:  ◈ disable   ◇ enable               │
│                                                               │
│            Loopback Type:  ◈ line_out_in  ◇ headphone_line_in │
│                                                               │
│              Crystal Test:  ◈ disable   ◇ enable              │
│                                                               │
│  ─────────────────────────────────────────────────────────   │
│                                                               │
│     ┌─────────┐      ┌─────────┐      ┌─────────┐             │
│     │  Apply  │      │  Reset  │      │ Cancel  │             │
│     └─────────┘      └─────────┘      └─────────┘             │
│                                                               │
└─────────────────────────────────────────────────────────────┘
```
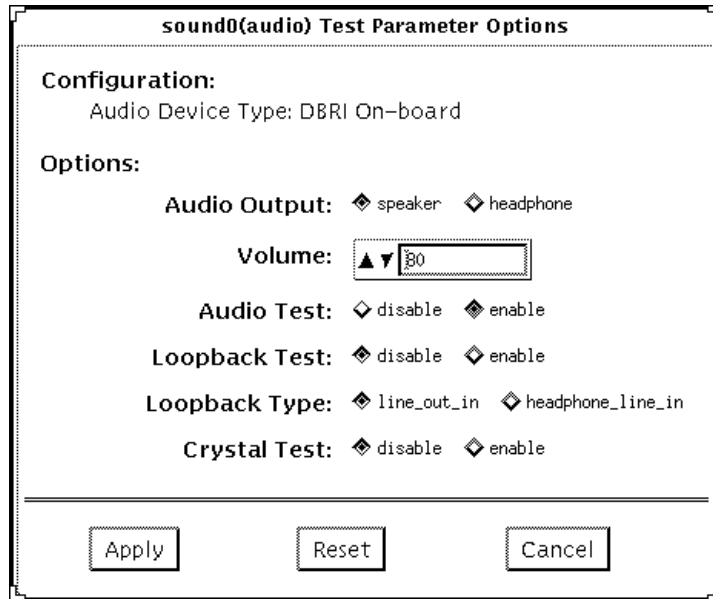
FIGURE 4-4 `audiotest` Test Option Menu for on-board `dbri(7)` for SPARCstation 20 and S240

The SPARCstation 20 On-board dbri is a high-quality 16-bit stereo multimedia codec.
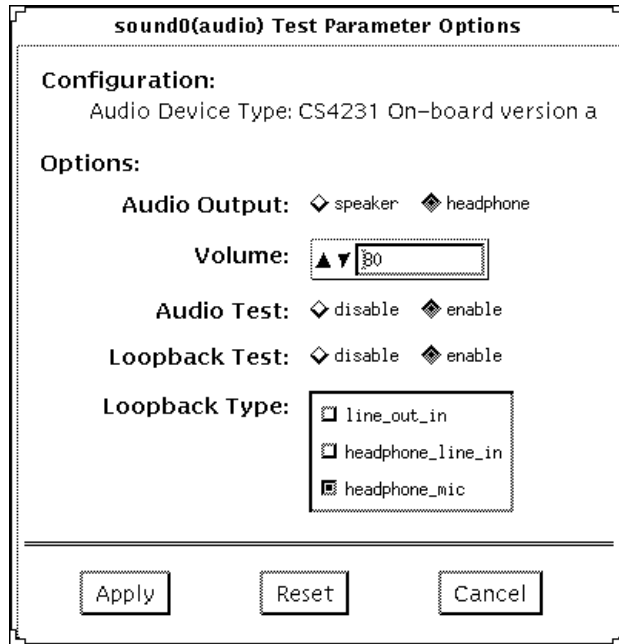
FIGURE 4-5 `audiotest` Test Option Menu for `audiocs(7)`

The `audiocs(7)` is an on-board, high-quality 16-bit stereo multimedia codec.
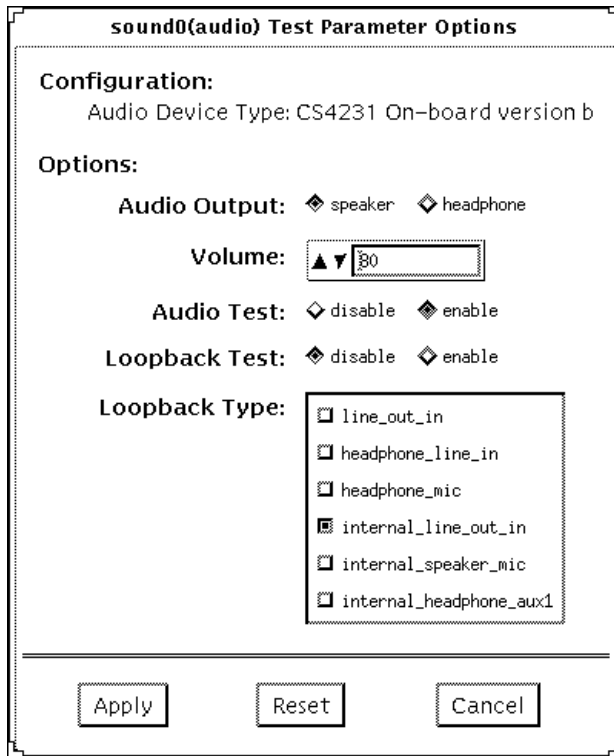
FIGURE 4-6 `audiotest` Test Option Menu for `audiocs`(7) with Internal Loopbacks

The `audiocs`(7) is an on-board, high-quality 16-bit stereo multimedia codec.

Note – The internal loopbacks are only active if the audio jacks are unused (nothing connected).

Some options can only be selected through the command line. See the command line option descriptions in "`audiotest` Command Line Syntax" on page 46.

TABLE 4-2 `audiotest` Options

Option	Description
Audio Test	Enables or disables the music play test. This test is enabled by default.
Audio Output	Selects the output port for the music play test.
Volume	Sets the volume for the music play test.

TABLE 4-2 `audiotest` Options (Continued)

Option	Description
Loopback Test	Enables or disables the loopback test. A loopback cable must be installed between the selected ports to run external loopback tests. This test is disabled by default.
Loopback type	Selects the type of loopback test to run.
Crystal Test	Enables or disables the crystal test. This test is disabled by default.
Controls Test	Enables or disables the speakerbox Controls test. This is an interactive test. The user is prompted to press the control buttons on the speakerbox. This test is disabled by default.

Note – Do not run the crystal test while running other SunVTS tests. The crystal test is timing-dependent. If the system is too busy, it fails due to time-out errors.

`audiotest` Test Modes

TABLE 4-3   `audiotest` Test Modes

Test Mode	Description
Connection Test	A simple open and close is performed. No data is transferred. The test returns a pass if the device can be opened and closed successfully. If the device cannot be opened because it is busy, then it is assumed that the device is successfully connected to another process and the test passes.
Functional Test	The record/play test is run and you can choose to run any of the tests described earlier. In this mode, the test will fail if the device is busy.
	If you invoke SunVTS through SyMON, the following Functional Test description applies: The record/play test is run. If the device cannot be opened because it is busy, then it is assumed that the device is successfully connected to another process and the test passes.

`audiotest` Command Line Syntax

/opt/SUNWvts/bin/audiotest *standard_arguments* **-o dev=/dev/sound/** *unit_no***,I=/** **dev***ioctl_device***,M,L,Q,S,T=***loopback_test_type***,X,E,LE,CD,CDD=***CD_device_name***,** **CDT=***track_number***,CDG=***play_gain***,CDL=***play_time***,W,MF=***filename***,TF=**filename

TABLE 4-4   `audiotest` Command Line Syntax

Argument	Explanation
dev=/dev/ audio_device	Specifies the audio device to be tested. The default is dev=/ dev/audio
I=/dev/ioctl_device	Specifies the `audio ioctl` device to be tested; the default is /dev/audioctl.
M	Enables the music play test.
L	Enables the loopback test.

TABLE 4-4 `audiotest` Command Line Syntax (Continued)

Argument	Explanation (Continued)
Q	Enables the quality test. This option does the same thing as L option except that it prints an extra status message upon completion.
S	Enables the speakerbox controls test.
T=loopback_test_type	Specifies the type of loopback test. The default is 1; the choices are listed below. 0 Codec Internal Loopback (CS4231 audio only) 1 Line-in/Line-out 2 Headphone/Line-in 3 Headphone/Microphone 4 Speaker/CD-input I1 Internal Line-in/Line-out I2 Internal Spk/Mic I3 Internal Headphone/Aux1 I4 Internal Speaker/Aux1 I5 Internal Headphone/Mic Note: Test type 0 is always run by default on CS4231 audio implementations. Test types 3, and 4 require special hardware, and are used by manufacturing centers and special test facilities. Do not invoke these tests unless you have the required hardware.
X	Enables the Audio Crystal test.
E	Continues testing if an error occurs.
LE	Loops on Error. This plays the signal data in a continuous loop.
CD	Enables the cdtest. This is for systems with an internal CD-ROM drive. A CD disc with music tracks must be loaded prior to running this test.
CDD=CD_device_name	Specifies the raw device name for the CD-ROM drive. The default is `CDD=/dev/rdsk/c0t6d0s0`.
CDT=number	Specifies the track number of CD-ROM to play. The default is to play the first track on the disc.
CDG=play_gain	Specifies the play gain of the CD Play test (0 to 255). The default is 120.
CDL=play_time	Specifies the number of seconds to run the CD Play test. The default is 30 seconds.

TABLE 4-4 `audiotest` Command Line Syntax (Continued)

Argument	Explanation (Continued)
W	Shows warning messages during the loopback test.
MF=filename	Selects an optional music file.
TF=filename	Specifies an optional tolerance file. Note: The tolerance file is used by manufacturing centers and special test facilities. Do not use this option unless you are familiar with the tolerance file format.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If the test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`audiotest` Error Messages

TABLE 4-5 `audiotest` Error Messages

	Error Message	Probable Cause(s)	Suggested Action (if applicable)
6000	Signal To Noise ratio too low (*name*) on *name*, SNR =*number* db, Min SNR =*number* db (*text*)	Loopback cable is missing or faulty. Audio hardware (usually consistent failures). System software problem (usually intermittent failures).	
6001	Channel Separation too low (*name*), SEP =*number* db, Min SEP =*number* db (*text*)	Loopback cable is faulty. Audio hardware is faulty (usually consistent failures). System software problem (usually intermittent failures).	
6002	Loop gain is out of range (*name*), *name* GAIN =*number* db, Min =*number* Max =*number* db (*text*)	Loopback cable is missing or faulty. Audio hardware is faulty (usually consistent failures). System software problem (usually intermittent failures).	
8000	Must be super user (*root*) to execute	The user does not have root privileges.	

TABLE 4-5 `audiotest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Suggested Action (if applicable)
8012	`Invalid audio device` (*device_name*) `for Crystal test`	The crystal test is not supported on the system audio device.	
8013	`Invalid audio device` (*device_name*) `for Controls test`	The system audio device does not support a speakerbox.	
8014	`Invalid audio device` (*device_name*) `for Loopback Quality test`	No loopback tests are supported on the system audio device.	
8015	`Invalid audio device` (*device_name*) `for Loopback Quality test` `(T=`*number*`)`	The specified loopback test is not supported on the system audio device. The specified loopback type is not supported on this audio device.	
8020	`Incomplete button press sequence`	The required button presses were not detected in the time allowed.	
8023	`Underrun/Overrun error failure`	The audio driver detected underrun or overrun errors while transferring data to the audio device. These errors usually happen when the loopback test is run while during heavy system activity.	
8027	*name*`:` `'`*name*`(`*name*`=`*number*`,` *name*`=`*number*`,` *name*`=`*number*`)'` `system call timeout. No response after` *number* `seconds. Device =` *device_name*	The system call is hanging.	
		Audio hardware (usually consistent failures).	
		System software problem (usually intermittent failures).	
8028	`Audio crystal test did not complete in the expected time, rate =` *number* `Hz, time =` *number* `usecs, min =` *number*`, max =` `number`	Excessive loading on the system.	Disable all other tests and try again.
8029	`Speaker is an invalid output port for device`	The system does not have an on-board speaker.	
8032	`Failed ioctl name` (*error_message*)	The `ioctl()` system call failed.	See the appended error message.

TABLE 4-5   `audiotest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Suggested Action (if applicable)
8033	`Failed mmap (`*error_message*`)`	The `mmap()` system call failed.	See the appended error message.
8034	`Failed fstat (`*error_message*`)`	The `fstat()` system call failed.	See the appended error message.
8035	`Failed ftruncate (`*error_message*`)`	The `ftruncate()` system call failed.	See the appended error message.
8036	`Write to audio device returned error condition (`*error_message*`)`	The `write()` system call failed.	See the appended error message.
8037	`Read from audio device returned error condition (`*error_message*`)`	The `read()` system call failed.	See the appended error message.
8038	`Failed to allocate number bytes of memory (`*error_message*`)`	The memory allocation system call failed, most likely due to a lack of system memory resource.	See the appended error message.
8051	`Unknown audio device (`*name=device_name*`, config=`*name*`)`		
8052	`Unable to get platform name`		
8053	`Unknown architecture/audio = `*name*` / `*device_name*		
8055	`Could not open `*device_name*`:` *error_message*	The `open()` system call failed Examine system message files (`/var/adm/messages`) for other information. Device does not respond. No `mmcodec` device found. Unable to communicate with speakerbox.	
8075	`Invalid audio file format`	Specified audio file is not in expected format.	
8077	`Invalid audio encoding`	An unknown audio encoding type was specified.	
8090	`Only one `*name*` loopback can be selected at a time`	The user specified more than one loopback test on the same port.	
8091	`Cannot enable loopback testing without selecting a loopback type`	The user enabled loopback testing but did not specify a loopback type.	

Bidirectional Parallel Port Printer Test (`bpptest`)

`bpptest` verifies the functionality of the bidirectional parallel port. SBus printer cards have two printer ports: one for SPARCprinters™, and one for a parallel port printer. Use the `lpvitest` to verify SPARCprinter functionality.

The `bpptest` verifies that your SBus card and its parallel port are working properly by attempting to transfer a data pattern from the SBus card to the printer.

Two indications show that the card and printer are functioning properly: First, you can see from the SunVTS Status window that `bpptest` made a successful pass, and second, that the pattern transmitted to the printer printed correctly.

If the `bpptest` passes successfully, you know that the SBus DMA circuitry, the printer, and the device driver are functioning properly.

Note – Large Postscript files or raster files may require that the printer has 2 Mbytes or more of memory. Otherwise, the printout may appear on two different sheets of paper.

Printer Test Hardware and Software Requirements

The SBus printer card and device drivers must be installed to run `lpvitest` or `bpptest`. A printer must be connected to the SPARCprinter or bidirectional parallel port, and be powered-up. If both a SPARCprinter and a parallel port printer are connected to the SBus card, you can test both devices at the same time.

Note – For a SPARCstation 10, SPARCstation LX, or SPARCclassic system, you can connect a printer directly to the onboard parallel port to run `bpptest`.

If you are testing the SPARCprinter port, be sure the magnets on the SPARCprinter paper tray are set to the correct paper size. For more information, see the *SPARCprinter Installation and User's Guide* and the label on the paper tray.

`bpptest` Options

TABLE 5-1 `bpptest` Options

Options	Description
Access	Determines the direction of data transfer; this field is informational only. `Writeonly` is the only option currently available. Data is transferred only from the SBus printer card to the printer.
Mode	Sets the print intervals. This option lets you select the intervals at which the test image is printed. The default setting is Fast. In the online and Functional tests this option is set to medium. In the Functional test, the choices are: `Fast`: Prints an image every 10 seconds. `Medium`: Prints an image every 12 minutes. `Extended`: Prints an image every 30 minutes.

`bpptest` Test Modes

This test supports Connection and Functional test modes.

TABLE 5-2 `bpptest` Test Modes

Test Mode	Description
Connection Test	In his mod, `bpptest` verifies that a bidirectional parallel port is configured on the system. The success of the `bpptest` in this mode indicates that the system has the bidirectional parallel port hardware and the software driver is installed.
Functional Test	The testing done in this mode is the same as in the online mode, except that the test registers a failure if the port is found busy. This is because SunVTS tests make the assumption that all the resources will be available for testing in the Functional test and therefore the unavailability of the device is interpreted as an indication of a fault condition.
	If you invoke SunVTS through SyMON, the following Functional test description applies: the test attempts to verify the functionality of the parallel port and the path between the parallel port and the host memory. An internal loopback test is performed and if a printer is attached to the port, a data pattern is transferred to the printer.

`bpptest` Command Line Syntax

/opt/SUNWvts/bin/bpptest *standard_arguments* **-o**
dev=*device_name***,access=***writeonly* | *readonly***,mode=***mode***,loop=***disable* | *enable*

TABLE 5-3   `bpptest` Command Line Syntax

Argument	Explanation
dev=*device_name*	Specifies the name of the device. This should be of the form `/dev/bpp#`, where # is the minor number of the device.
access=*writeonly* | *readonly*	Determines test mode: **writeonly** or **readonly**. Note: This flag is mandatory.
mode=*mode*	Sets the test image print rate. The test image is a continuous printout of the ASCII character set. Possible rates are: `Fast`: Prints the test image at 10-second intervals. `Medium`: Prints the test image at 12-minute intervals. `Extended`: Prints the test image at 30-minute intervals.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`bpptest` Error Messages

TABLE 5-4   `bpptest` Error Messages

	Error Message	Probable Cause(s)	Suggested Action (if applicable)
6000	Read compare failed, expected:*value*, observed:*value*	Suspect DMA circuitry or FIFO associated with bpp.	
6001	Error occurred while accessing *device_name*; Time out Error: *value*; Bus Error: *value*; Pin Status: value	System error message.	

TABLE 5-4   bpptest Error Messages (Continued)

	Error Message	Probable Cause(s)	Suggested Action (if applicable)
8000	could not open *device_name*	System error message.	
		Device does not exist.	
		Device not configured correctly.	
8001	Encountered error while writing to *device_name*; Requested bytes to write: *value*; Bytes successfully written: value	Device not configured correctly or does not exist.	
		Too many print requests queued up.	
		Printer out of paper.	
8002	Failed to obtain memory from malloc() system call	System heavily loaded.	
8003	Error occurred while attempting to fork a child process	System error message.	
8004	ioctl fail on *device_name*	System error message.	
8005	error=*number*	System error message.	
8006	Device *device_name* unavailable for testing	Device is busy.	Retry later.

Compact Disc Test (`cdtest`)

`cdtest` checks the CD-ROM unit by reading the CD. `cdtest` is not a scalable test. Each track is classified as follows:

- Mode 1 uses error detection/correction code (288 bytes).

- Mode 2 uses that space for auxiliary data, or as an audio track.

Note – Load a compact disc into the drive before starting the test. See the explanation for CD Type in TABLE 6-2 in the chart in "`cdtest` Test Modes," later in this chapter.

Volume Management and Compact Discs

`cdtest` tests the CD-ROM drive(s) even if the Volume Manager is not running. If the Volume Manager is running and no media is installed in the CD-ROM drive(s), SunVTS prompts you to install media in the drive before selecting the test.

The test fails if you try to run it without a CD in the drive.

`cdtest` Options
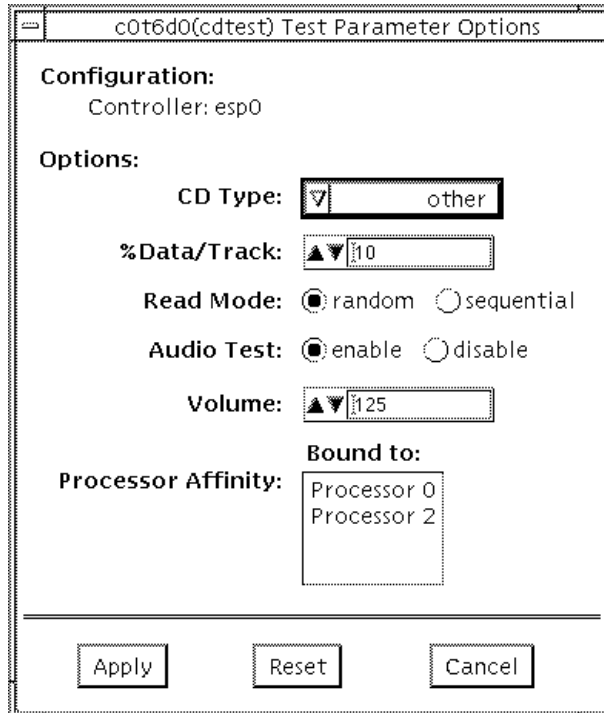
FIGURE 6-1   cdtest Option Menu

TABLE 6-1   cdtest Options

Options	Description
CD Type	The types of compact discs that can be tested are listed in the CD Type menu. The choices are: pdo, multi-session, or other (the default CD type is other).In the Connection test, this option has a default value of other. NOTE: Your choice must correspond with the disc used for testing.
% Data/Track	Tests a percentage of data on each track; type a value between 0 and 100 in this field to indicate the percentage. In the online and connection tests this option has a canned value of 2%.
Read Mode	cdtest reads the CD either in Random or Sequential mode. In Random mode, data blocks are read from random track positions; in Sequential mode, data blocks are read in sequence. For both modes, the total number of blocks read is determined by the %_of_data option. In the online and Connection tests this option has a canned value of random.
Audio Test	Enables or disables the audio test. You must connect headphones or a speaker to the audio jack on the CD player to hear audio output. In the Connection test, this option has a default value of disable.
Volume	Adjusts the volume; type a value between 0 and 255 in this field. In the online and connection tests this option has a default value of 125.

`cdtest` Test Modes

This test supports Connection and Functional tests.

TABLE 6-2   `cdtest` Test Modes

Test Mode	Description
Connection Test	In this mode, `cdtest` verifies that a CD-ROM drive is connected to and configured in the system.
Functional Test	This mode is similar to Online mode except that the test registers a failure if the device is found to be busy. This is because SunVTS tests make the assumption that all the resources will be available for testing in the Functional test and the unavailability of a device is interpreted as an indication of a fault condition.
	If you invoke SunVTS through SyMON, the following Functional Test description applies: This mode attempts to verify the functionality of the drive by accessing and reading the media in the drive. If the media contains audio and the audio test is enabled, `cdtest` tries to play it. Only a small percentage of the media is read, as this is sufficient to verify the functionality of the drive and keeps the test runtime from being unnecessarily extended.
	If the device is found to be busy at the time of testing, the test exits after printing a message indicating that the device is unavailable for testing. The pass and error counts are unaffected.

`cdtest` Command Line Syntax

/opt/SUNWvts/bin/cdtest *standard_arguments* **-o dev=***raw_device_name,*
mode=*mode,***read=***random | sequential,***data=***%_of_data,***vol=***volume,*
audio=*enable | disable,***type=***CD_type*

TABLE 6-3   `cdtest` Command Line Syntax

Argument	Explanation
dev=*raw_device_name*	Specifies the name of the raw device to be tested.
read=*random	sequential*	Indicates random or sequential read access.
data=*%_of_data*	Sets the percentage of data to be tested; you can specify **0** to **100** percent.
vol=*volume*	Controls the audio volume; you can specify **0** through **255**; the default is **255**.
audio=*enable	disable*	Enables or disables the audio test; you must connect headphones or a speaker to the audio jack on the CD player to hear audio output.
type=*CD_type*	Specifies the type of CD used for the test; the choices are: **pdo, multi-session, sunos** and **other**; the default is other.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/ sparcv9/`*testname.* If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`cdtest` Error Messages

TABLE 6-4 `cdtest` Error Messages

	Error Message	Probable Cause(s)	Suggested Action (if applicable)
6000	Fail to seek at block *value*	System error message.	Retry with a different CD-ROM.
		Bad media.	
6001	Fail to read *number* bytes at block *number*	System error message.	
		Bad media	
		Incorrect CD-ROM type passed to `cdtest`.	
6002	Data mode *value* not supported	Datamode has a value other than `CD_DATAMODE1` or `CD_DATAMODE2` (see `cdio(7)`).	
6003	Error occurred while playing audio track *number*		
6004	Failed to obtain memory from malloc() system call	Audio stop before complete at track *number*.	
6005	Unexpected audio status *value* received at track *number*	System error message	
6006	Fail in reading offset of the multi-session, offset = *value*	An incorrect value for cdtype has been passed to the `cdtest`.	
		Bad media.	
6007	This is not a CD ROM Multi-Session disc!	An incorrect value for `cdtype` was passed to the `cdtest`.	
8000	Fail to open device *device_name*	System error message.	
8001	Fail ioctl *name*	System error message.	
8002	Fail ioctl *name*	System error message.	
		No partitions on the disc	
8003	Fail to allocate *number* bytes of memory		

TABLE 6-4 `cdtest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Suggested Action (if applicable)
8004	Error occurred while reading TOC header	Track numbers out of range; First track: *value*, Last track: *value*.	
8005	Data miscompare while reading number bytes starting from block *number*	An incorrect value for cdtype has been passed to `cdtest`.	
		Bad media.	
8006	ioctl CDROMSTART failure (number of retries: *value*)		

Color Graphics Frame Buffer Test (`cg14test`)

`cg14test` checks the `cg14` frame buffer card. `cg14test` is specific to the VSIMM (Video SIMM) devices in the SPARCstation 10 SX and the SPARCstation 20 SX.

> ⚠️ **Caution –** Due to possible conflicts between SunVTS cg14 frame buffer tests and OPEN LOOK applications that use the cg14 frame buffer, the following restrictions apply when running `cg14test`:
>
> Do not run graphic applications other than OPEN LOOK while SunVTS is running frame buffer tests.
>
> Do not run OPEN LOOK programs that generate video updates outside or on top of the SunVTS window.
>
> Do not close the SunVTS window to an icon while it is running frame buffer tests.
>
> Ensure that the frame buffer locking option is enabled from the Options window (see the section about testing frame buffers in *SunVTS 3.0 User's Guide* for details).

`cg14test` Groups

There are nine test groups in `cg14test`:

1. MDI and VBC Chip Control Registers

2. Memory Chips

3. MDI Chip Cursor Registers

4. MDI Chip CLUT Registers

5. DAC Chip Registers

6. MDI Chip XLU Registers

7. CG14 Display (visual only)

8. MDI Chip Testmode Readback in 8-bit mode

9. Driver IOCTLs

TABLE 7-1   `cg14` Test Groups

Test Groups	Description
Hardware (Groups 1-6)	These test groups are tested by opening `/dev/fbs/cgfourteenX`, mapping the MDI Control Address Space, modifying the target test location (using direct writes to the mapped address space), reading from the mmapped address space for verification, and closing the device.
Visual Pattern (Group 7)	This subtest loads a visual pattern of 256 colors, then rotates the pattern around. You verify this test by seeing it display.
Data Propagation (Group 8)	This test group is tested by loading the frame buffer (FB) memory with four neutral data patterns, then setting a target FB pixel with data that triggers the test mode readback latch. The result is read from the readback register after vertical blanking occurs. Two different trigger patterns are used at each FB pixel. All four MDI pixel paths (A - D) are used, and the pixel locations for each trigger are designed to detect gross MDI input data opens or short, VRAM SAM addressing, and VRAM-to-SAM transfer addressing. The screen shows four horizontal bars, which are either greyscale or colored. These bars change each time the trigger data is inverted, and as it completes the testing of a raster pattern. NOTE — If the resolution and VRAM size permits, 8-bits per pixel mode are tested.

TABLE 7-1   cg14 Test Groups (Continued)

Test Groups	Description
Driver (Group 9)	Test all IOCTL calls that have not yet been used to verify proper driver communication to the hardware. Call the driver to perform a hardware update, and then confirm that the update was successful by using the complementary driver read, or reading the mmap'ed address space and comparing it against the stimulus. cg14test performs the appropriate steps before and after each test (if possible) to maintain context and prevent visual confusion by saving the register data before it is overwritten, disabling video (if possible), performing the specific test, and restoring the saved register data information. The data used for register testing is optimized to include all 0's, all 1's, and walking a 1 through each bit being tested.
MDI and VBC Chip Control Registers (Group 1)	Master Control Register bits 7-0 write/read verify Packed Pixel Register bits 3-0 write/read verify Master Status Register bits 7-4 read-only verify 0x00 and 0x30 occur Horizontal Blank Start Register bits 9-0 write/read verify Horizontal Blank Clear Register bits 9-0 write/read verify Horizontal Sync Set Register bits 9-0 write/read verify Horizontal Sync Clear Register bits 9-0 write/read verify Composite Sync Clear Register bits 9-0 write/read verify Vertical Blank Start Register bits 11-0 write/read verify Vertical Blank Clear Register bits 11-0 write/read verify Vertical Sync Set Register bits 11-0 write/read verify Vertical Sync Clear Register bits 11-0 write/read verify Transfer Cycle Set Register bits 9-0 write/read verify (MDI revision 0 only) Transfer Cycle Clear Register bits 9-0 write/read verify (MDI revision 0 only) Fault Status Address Register bits 15-0 write/read verify Auto-increment Address Space Register bits 7-0 write/read verify Video Base Register bits 23-12 write/read verify
Memory Chips (Group 2)	The Memory Chips test group includes VRAM Testing, Memory Retention, and Test Write Recovery. VRAM Testing The Data Bus Test uses 18 NTA patterns (Nair, Thatte, and Abraham's testing procedure for RAM) to check for data and address faults. This test is performed in MDI_CHUNKY_XBGR_MAP access mode only. See Table 7-2.

TABLE 7-1 `cg14` Test Groups (Continued)

Test Groups	Description
Memory Chips (Group 2) Continued	VRAM Testing (Continued) The test ascends through the Frame buffer memory, clearing it to 0's. The NTA pattern test number *x* reads a location to make sure test data *y* is present. It then writes new data *z* to that location. The location ascends through the FB sequentially. See Table 7-2.
	Memory Retention VRAM Data Retention checks for gross problems with the VRAM refresh. Since refresh is active during this test, no retention problems should occur unless the refresh is defective.
	This test turns off the video, writes 0's to all the VRAM, waits the specified `memory_hold` time (the default is five seconds), then reads and compares all VRAM data. This process is repeated with data of f's, then the video is restored and the test is complete.
	There are two new command line parameters related to this test: **R=***number* and **H=***number*. **R=** lets the user specify the refresh interval from 128-1023. The time between refresh cycles and the system default is 123. H**=** lets the user specify the retention test hold time in seconds.
	Test Write Recovery A write recovery test is used in all the EMC mapping modes to write data to 0's followed by immediately reading that data location to see if the VRAM can recover from a write correctly. This is done to all sequential ascending locations. Next, a second independent pass of memory is made with the complementary data of `0xffffffff` written to descending locations of the FB memory buffer.
	The EMC mapping access modes are: `MDI_CHUNKY_XGBR_MAP` `MDI_CHUNKY_BGR_MAP` `MDI_PLANAR_X16_MAP` `MDI_PLANAR_C16_MAP` `MDI_PLANAR_X32_MAP` `MDI_PLANAR_B32_MAP` `MDI_PLANAR_G32_MAP` `MDI_PLANAR_R32_MAP`

TABLE 7-1   `cg14` Test Groups *(Continued)*

Test Groups	Description
MDI Chip Cursor Registers (Group 3)	The MDI Chip Cursor Registers are: Cursor Plane 0 Register bits 31-0 write/read verify Cursor Plane 1 Register bits 31-0 write/read verify Cursor Plane 0 Register bits 31-0 write/read verify (with auto increment) Cursor Plane 1 Register bits 31-0 write/read verify (with auto increment Cursor Control Register bits 2-0 write/read verify Cursor Color Register 1 bits 28-0 write/read verify Cursor Color Register 2 bits 28-0 write/read verify X-Cursor Location Register bits 11-0 write/read verify Y-Cursor Location Register bits 11-0 write/read verify Cursor Plane 0 Non-Auto Registers test Cursor Plane 0 Auto Registers test Cursor Plane 1 Non-Auto Registers test Cursor Plane 1 Auto Registers test Cursor Planes Retry A test Cursor Planes Retry B test
MDI Chip CLUT Registers (Group 4)	The MDI Chip CLUT Registers are: LUT1 Registers 0-255 bits 31-27 & 23-0 write/read verify LUT1 Registers 0-255 bits 31-27 & 23-0 write/read verify (with auto increment) LUT1D Registers 0-255 bits 31-27 & 23-0 write/read verify LUT1D Registers 0-255 bits 31-27 & 23-0 write/read verify (with auto increment) LUT2 Registers 0-255 bits 31-27 & 23-0 write/read verify LUT2 Registers 0-255 bits 31-27 & 23-0 write/read verify (with auto increment) LUT2D Registers 0-255 bits 31-27 & 23-0 write/read verify LUT2D Registers 0-255 bits 31-27 & 23-0 write/read verify (with auto increment) LUT3 Registers 0-255 bits 31-27 & 23-0 write/read verify LUT3 Registers 0-255 bits 31-27 & 23-0 write/read verify (with auto increment) LUT3D Registers 0-255 bits 31-27 & 23-0 write/read verify LUT3D Registers 0-255 bits 31-27 & 23-0 write/read verify (with auto increment)

TABLE 7-1   `cg14` Test Groups (Continued)

Test Groups	Description
DAC Chip Registers (Group 5)	The DAC Chip Registers test group includes the RAMDAC registers and control registers. RAMDAC Registers Address Register bits 7-0 (0x7 maximum) write/read verify Mode Register bits 7-0 (skip bit 5) bits write/read verify Control Registers ID Register bits 7-0 r/o verify data is 0x8C Pixel-Mask Register bits 7-0 write/read verify (skipped if dac rev = 2) Command2 Register bits 7-0 write/read verify (skipped if dac rev = 2) Command3 Register bits 7-0 write/read verify (skipped if dac rev = 2)
MDI Chip XLUT Registers (Group 6)	The MDI Chip XLUT Registers are: XLUT Registers 0-255 bits 7-0 write/read verify XLUT Registers 0-255 bits 7-0 write/read verify (with auto increment) XLUTD Registers 0-255 bits 7-0 write/read verify XLUTD Registers 0-255 bits 7-0 write/read verify (with auto increment)
CG14 Display (visual only) Group 7	This test visually displays 256 boxes on the screen (each in a different color), and then shifts the CLUT1 entries giving the visual impression of the pattern mirroring itself from left to right horizontally. The pattern then rotates up, down, followed by mirroring itself horizontally left to right.
MDI Chip Test Mode Readback Register (Group 8)	This Test Mode reads back register bits 23-0 in read-only and verify modes.

TABLE 7-1 `cg14` Test Groups (Continued)

Test Groups	Description
Driver IOCTLs (Group 9)	`MDI_GET_CFGINFO` check # of CLUT's, pixel height, pixel width, and pixel mode against hardware `FBIOGATTR` check `real_type`, `fb_height`, `fb_width`, `fb_depth`, `fb_cmsize`, and `fb_size` against cfginfo values `FBIOGTYPE` check `fb_type`, `fb_height`, `fb_width`, `fb_depth`,`fb_size`, and `fb_cmsize` against driver defines or cfginfo values `FBIOGVIDEO` check status returned against hardware `FBIOSVIDEO` set off, off, on, on, off verifying against hardware `FBIOVERTICAL` (imbedded in `FBIOSVIDEO`) `MDI_VRT_CNTL` turn off, off, on, on, off the video interrupt enable and verify the hardware agrees`MDI_SET_PIXELMODE` set different modes and verify against the hardware `MDI_SET_PPR` set the different modes and verify against the hardware `MDI_SET_COUNTERS` set HSS, HSC, XCC, HBC, XCS, HBS, CSC, VSS, VSC, VBC, VBS, HCT, and VCT then verify against hardware `MDI_SET_XLUT` set xlut and verify against hardware `MDI_GET_XLUT` get xlut and verify against hardware `MDI_SET_CLUT` set clut (1-3 as applicable) and verify against hardware `MDI_GET_CLUT` get clut (1-3 as applicable) and verify against hardware `FBIOPUTCMAP` set and verify clut1 matches `FBIOGETCMAP` verify clut1 matches get `FBIOSATTR` set `emu_type` to `FBTYPE_MDICOLOR` and verify `FBIOGATTR` check `FBIOGCURMAX` verify x and y size match driver defines `FBIOSCURSOR` verify set at 3 locations matches hardware `FBIOGCURSOR` verify driver knows what set(s) just did `FBIOSCURPOS` verify set at three locations matches hardware `FBIOGCURPOS` verify driver knows what set(s) just did `MDI_SET_CURSOR` set then check CCR, XCU, and YCU cursor hardware registers

1.

TABLE 7-2 `cg14test` NTA Testing Patterns

NTA Test Pattern Number = x	Test Data = y	New Data = z
1.0	0x00000000	0x01010101
1.5	0x01010101	0xffffffff
2.1	0xffffffff	0xf1f1f1f1

TABLE 7-2   `cg14test` NTA Testing Patterns (Continued)

NTA Test Pattern Number = x	Test Data = y	New Data = z
2.2	0xf1f1f1f1	0x33333333
3.1	0x33333333	0xf0f0f0f0
3.2	0xf0f0f0f0	0x0f0f0f0f
4.1	0x0f0f0f0f	0x55555555
4.2	0x55555555	0xaaaaaaaa
5.1	0xaaaaaaaa	0x05050505 (1x) 0x88888888 (2x)
5.2	0x88888888	0xf5f5f5f5
6.1	0xf5f5f5f5	0x00000000 (1x) 0x5f5f5f5f (2x)
6.2	0x5f5f5f5f	0x11111111
7.1	0x11111111	0x00000000 (1x) 0xcccccccc (2x)
7.2	0xcccccccc	0xdbdbdbdb
8.1	0xdbdbdbdb	0x6d6d6d6d
8.2	0x6d6d6d6d	0x6b6b6b6b
9.1	0x6b6b6b6b	0x0000000
9.2	0x00000000	–

`cg14test` Options
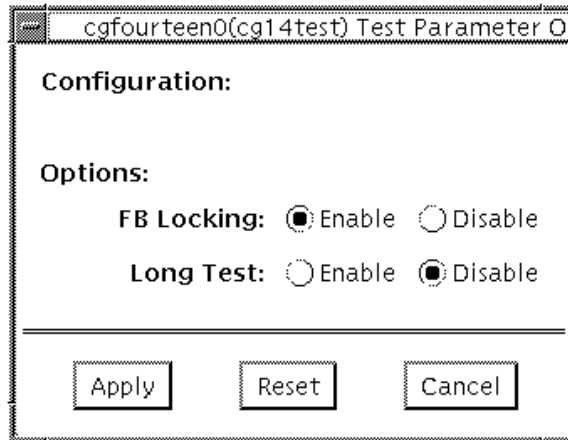
FIGURE 7-1   `cg14test` Option Menu

TABLE 7-3   `cg14test` Options

Options	Description
FB Locking	See the section about Testing Multiple Frame Buffers in the *SunVTS 3.0 User's Guide* for details.
Long Test	When enabled, the color bar screen(s) in the MDI Testmode Readback test checks all SAM transfers in clock=0 mode and clock=1 mode. If Long test is disabled, clock=1 runs checks on the first eight addresses and the first SAM transfer only.
Processor Affinity	For multiprocessor systems, indicates the processor to be tested.

`cg14test` Command Line Syntax

/opt/SUNWvts/bin/cg14test *standard_arguments* **-o dev=***device_name,*
lock=*E(nable)/D(isable)***,L,I**

TABLE 7-4 `cg14test` Command Line Syntax

Argument	Explanation
dev=*device_name*	Specifies the path of the cg14 device file to be tested; for example: **/dev/fbs/***device_name.*
lock=*E(nable)/* *D(isable)*	*Enables and disables the window system locking option. See the Testing Multiple Frame Buffers sections in the SunVTS 3.0 User's Guide for details. Do not use when your device is the window system display.*
L	Enables the long TMRB test.
I	Enables optional driver ioctl tests for cursor. **Note–** Do not move the mouse during the `cg14test` when you run this option.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`cg14test` Test Modes

Due to the nature of graphic tests, reading from or writing to the frame buffer during graphic tests will disturb user operation. This test is only available in Functional test mode.

The Functional test uses all subtests to test the cg14 frame buffer. The user can select the long mode for TRMB subtest.

`cg14test` Error Messages

TABLE 7-5 `cg14test` Error Messages

	Error Message	Probable Cause(s)	Suggested Action (if applicable)
6002	`MDI` *name* `register` = *number*	CG14 video board	
6004	`MDI Cursor planes retry test maximum retry limit exceeded`	CG14 video board	
6006	*name* `error message`	CG14 video boar	
6008	`Failed open of file` *name*, `errno=`*number*	Disk	
		CPU board	
6010	`Failed write to file` *name*, `errno=`*number*	CG14 video board	
		CPU board	
		Disk	
6012	`Failed close of file` *name*, `errno=`*number*	Disk	
		CPU board	
6014	`MDI chip TestMode Readback,` *number*`-bit` *name* `mode,` `offset=` *number* `pixelpipe=`*name* `clock=`*number* `exp=`*number* `obs=`*number*		
6016	`MEM (`*name*`), Data Retention offset=` *number* `exp=`*number* `obs=number`	CG14 video board	
		CPU board	
6018	`MEM (`*name*`), NTA` *name* `offset=` *number* `exp=`*number* `obs=`*number*	CG14 video board	
		CPU board	
6020	`MEM (`*name*`), WRRD` *name* `offset=` *number* `exp=`*number* `obs=`*number*	CG14 video board	
		CPU board	
6022	`failed, mapping` *name* `space,` `errno =` *number*	CG14 device file	
		SunOS	
		CG14 video board	
		CPU board	

TABLE 7-5 `cg14test` Error Messages (Continued)

	Error Message	Probable Cause(s)	Suggested Action (if applicable)
6024	MEM (*name*), WRRD *name* offset= *number* exp=*number* obs=*number*	SunOS	
		CG14 video board	
		CPU board	
6026	VBC Control Register exp=*number* obs=*number*	CG14 video board	
		CPU board	
8002	*name* is an invalid parameter for *name*!	Operator error	
8004	unable to close device *name*	CG14 device file	
		SunOS	
		CG14 video board	
8006	invalid CG14 device type from *name*	CG14 device file	
		SunOS	
		CG14 video board	
8008	IOCTL Error: *name*	CPU board	
		CG14 video board	
8008	IOCTL(*name*) *name*	CPU board	
		CG14 video board	
8010	unable to unlock OL windows	SunOS	
		Operator error	
8012	IOCTL(*name*,CLUT*number*) *name*	CG14 device file	
		SunOS	
		CG14 video board	
8014	unable to open CG14 device *name*	Incorrect device name	
		No existing device	
		CG14 video board	

Frame Buffer, GX, GX+ and TGX Options Test (`cg6`)

The `cg6` test verifies the cgsix frame buffer and the graphics options offered with most SPARC based workstations and servers. This test stresses the frame buffer with the subtests described below.

Note – Disable all screen savers before testing any graphics device. Type **xset s off** at the prompt to disable the Solaris screen saver.

To start SunVTS with `vtsui`, but without `vtsk`, add the host name to `xhost` as: **xhost +** *hostname.*

`cg6` Subtests

TABLE 8-1   `cg6` Subtests

Subtests	Description
Cursor Test	`cg6` visually checks the overlay registers of the RAMDAC. A pointer is drawn on the screen and moved to predetermined locations. There is a problem if the pointer disappears. This visual test ensures that the overlay is working properly
Fast Copy in Double Buffer Test Mode	Creates two full-size screen raster images in double buffer mode, writing different patterns to each. The hidden buffer is copied to the visible buffer, and the data is compared. An error message is returned if there are inconsistencies. Then the buffer is flipped and the process is repeated. Note: This test only applies to Sun Microsystems GX+ graphic accelerators with double-buffering capacity.
TEC Test	Verifies that the Transformation Engine and Cursor control logic are accessible to confirm that further TEC access is performed correctly.
FBC Test	Verifies that the Frame Buffer Controller logic is accessible to confirm that further FBC access is performed correctly.
Frame Buffer Test	Verifies that the frame buffer memory is working. A "walking ones" pattern is written to memory, with a specific color signifying one of eight bits. The screen is divided into eight equally wide vertical stripes. A "walking one" is written to each stripe, causing eight iterations of these stripes. The value written is read back and checked. If the values do not match, an error is reported.
Screen Test Using Blits	Draws blocks of color and performs `blit` transfers to other portions of the screen. First, the entire screen is drawn with cyan, then a black block is placed in the upper-left corner. This subtest `blits` this block on the upper-right, lower-right, and lower-left corners, then "or's" the whole image.
Blit Test	Draws a block of data and `blit` into a location at the bottom-right rectangle.

TABLE 8-1   cg6 Subtests

Subtests	Description
Line Test	Draws lines on the screen in different colors with different data values. The data is read back and compared with the expected values. An error is returned in the case of a mismatch.
Polygon Test	Draws hourglass-shaped polygons on the screen, using the four vertices. After all the polygons are rendered in the video memory, they are read back and the data compared with expected values. If there is a mismatch, an error is displayed.
Colormap Test	Loads all 256 locations in the color map with a greyscale, both backward and forward. This means decreasing values are loaded to all R, G, and B values. Note: If the system being tested has a monochrome or greyscale monitor, visual color problems are undetectable.

cg6 Options
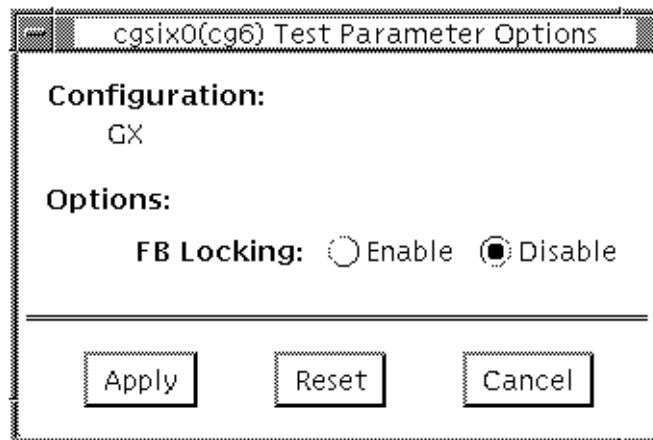
FIGURE 8-1   cg6 Option Menu

Note – See the section about Testing Frame Buffers in the *SunVTS 3.0 User's Guide* for details about frame buffer locking.

`cg6` Test Modes

Due to the nature of graphic tests, reading from or writing to the frame buffer during graphic tests will disturb user operation. This test is only available in Functional test.

`cg6` Command Line Syntax

/opt/SUNWvts/bin/cg6test *standard_arguments* **-o dev=***device_name***, lock=***E(nable)/D(isable)***,Passes=***number*

TABLE 8-2   `cg6` Command Line Syntax

Argument	Explanation
dev=*device_name*	Specifies the path of the cg14 device file to be tested; for example: **/dev/fbs/***device_name*.
lock=*E(nable)/D(isable)*	Enables/disables the window system locking option. See the section about Testing Multiple Frame Buffers in the *SunVTS 3.0 User's Guide* for details. Do not use when device is the window system display.
Passes=*number*	The number of passes to run; the default is 1.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

Note – Extra swap space of 5 MBytes is required.

`cg6` Error Messages

TABLE 8-3 `cg6` Error Messages

	Error Message	Probable Cause(s)	Recommended Action(s)
6000	Resolution not supported for double buffering	Unsupported resolution.	Change to other resolutions.
6002	data error: x_pos: *number*, y_pos: *number*. *mobs: *number*, *mexp: *number*	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
6004	FB Screen Test data error: xpos=*number*, ypos=*number*, exp(*number*), obs(*number*)	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
6006	Data Error: Screen location x: *number*, y: number, Obs: *number*, Exp: *number*	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
6007	Multibuffering extension does not exists	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
6008	Cannot create CMAP	System error.	If the problem persists, call your authorized Sun service provider.
		Faulty frame buffer.	
6009	name off(*number*)exp(*number*), obs(*number*), xor(*number*)	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
6010	Could not get color	System error.	If the problem persists, call your authorized Sun service provider.
		Faulty frame buffer.	
6011	*name* off(*number*)exp(*number*), obs(*number*), xor(*number*) *name*	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
6012	Checksum test failed for PROM. Observed: *number*, Expected: *number*	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
6014	Unable to lock frame buffer	FB lock was enabled while window system was not running.	Disable the FB lock.
6016	Found error in sine test	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.

TABLE 8-3   cg6 Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action(s)
6018	Found error in cursor test	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
6020	Found error in fb test	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
6022	Found error in polygon test	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
6024	Found error in cmap test	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
6026	Found error in db test	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
6028	Found error in line test	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
6030	Found error in dac test	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
8000	*name* not mapped	System error.	If the problem persists, call your authorized Sun service provider.
8002	can't acquire console semaphore	System error.	If the problem persists, call your authorized Sun service provider.
8004	XCreateSimpleWindow failed	System error.	If the problem persists, call your authorized Sun service provider.
8008	Couldn't create enough buffers	Not enough memory available.	Reduce the system load.
		System error.	If the problem persists, call your authorized Sun service provider.
8010	grab of pointer unsuccessful	Not enough memory available.	Reduce the system load.
		System error.	If the problem persists, call your authorized Sun service provider.

TABLE 8-3 cg6 Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action(s)
8012	cmap mapping failed	Not enough memory available.	Reduce the system load.
		System error.	If the problem persists, call your authorized Sun service provider.
8014	Blit status. Could not launch draw	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
8016	Draw Wait FBC_DRAW status=*number*	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
8018	Could not get cmap	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
8020	Could not get color	Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
8022	Could not create child raster	Incorrect device name.	Re-enter the correct name.
		Not enough memory available.	Reduce the system load.
8024	Failed to get malloc	Not enough memory available.	Reduce the system load.
		System error.	
8026	Cannot create CMAP	System error.	If the problem persists, call your authorized Sun service provider.
		Faulty frame buffer.	
8028	Couldn't get hostname of machine under test	System error.	If the problem persists, call your authorized Sun service provider.
8030	Failed in opening device *name*	Incorrect device name.	Re-enter the correct name.
		Faulty frame buffer.	If the problem persists, call your authorized Sun service provider.
8032	Failed to create raster	Incorrect device name.	Re-enter the correct name.
		Not enough memory available.	Reduce the system load.

TABLE 8-3   `cg6` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action(s)
8034	`Failed to create context`	Not enough memory available.	Reduce the system load.
		System error.	If the problem persists, call your authorized Sun service provider.
8036	`fbs/cgsix0 FATAL: "mmap() failed`	Overloaded system.	Reduce the system load.
		System error.	Check OS release number.

Disk and Floppy Drives Test (`disktest`)

`disktest` verifies the functionality of hard disk drives and floppy drives using three subtests (see TABLE 9-1); Media, File System, and Asynchronous I/O. Most disk drives, such as SCSI disks, native or SCSI floppy disks, IPI, and so on, are supported. The type of drive being tested is displayed at the top of the Test Parameter option menu.

The `disktest` Test Parameter option menu shows all the partitions that are available for testing. The file System subtest can only be run if the selected partition is mounted (described below). The WriteRead option of the Media subtest is allowed only if a selected partition is *not* mounted.

To have SunVTS pre-mount all mountable partitions, set the environment variable `BYPASS_FS_PROBE` to zero before starting SunVTS. Disable pre-mounting by unsetting `BYPASS_FS_PROBE` or changing it to a value other than zero.

The mount point used by `disktest` is the word *disktest* appended by the name of the disk partition. For example, if the disk partition name is `/dev/dsk/c0t3d0s0`, `disktest` mounts it as superuser under the name `/disktest_c0t3d0s0`

> **Caution –** If a power failure occurs while the Media subtest is running in WriteRead mode, disk data will be destroyed.

> **Caution –** Running the Media subtest on a disk partition in the WriteRead mode may cause data corruption if the same partition is being used by other programs. Only select this mode when the system is offline (not used by any other users or programs).

`disktest` tests the floppy drive regardless of whether the Volume Management software is running or not. The following mount point names are used:

- If the Volume Management software *is* running, `disktest` tests the disk drive with the mount point name in the `/etc/mnttab` file.
- If the Volume Management software is *not* running, `disktest` tests the disk drive with the device name `dev=/dev/diskette`. Do not edit the `/etc/vold.conf` file to change the floppy drives. Currently, the SunVTS software is hard-coded to use these path names as the default logic names.

Table 2-1 describes the `disktest` subtests

TABLE 9-1   `disktest` Subtests

Subtest	Description
Media Subtest	Verifies disk media by writing data to and reading data from the disk. The Media subtest treats a disk as one large chunk of contiguous data. This is a scalable test that can run multiple copies in read/write mode on the same disk partition. To avoid data corruption, all simultaneous instances of `disktest` communicate through a shared memory service. This ensures that different copies of the media subtest do not overlay the same disk block at the same time.
File System Subtest	Verifies the disk system's integrity. The file system subtest exercises the partition being tested to determine if it is mounted. If the partition is not already mounted or pre-mounted, then the test is blocked. The test opens two temporary files (of the size specified on `File System File Size`) and performs a Read/Write test.
Asynchronous I/O Subtest	Uses the asynchronous read/write feature of the Solaris disk driver to exercise the disk. In read-only mode, the test sends a maximum of four asynchronous read packets, each with a random size and a random offset into the selected partition. The test then waits for all outstanding I/O activity to complete before issuing another round of packets. This process continues until the whole area is tested. In read-write mode, one write packet is issued in every four read packets as a spot check of the write operation. Before data is written to a particular location, data is backed up, write-verified, and restored to its original state.

`disktest` Test Options
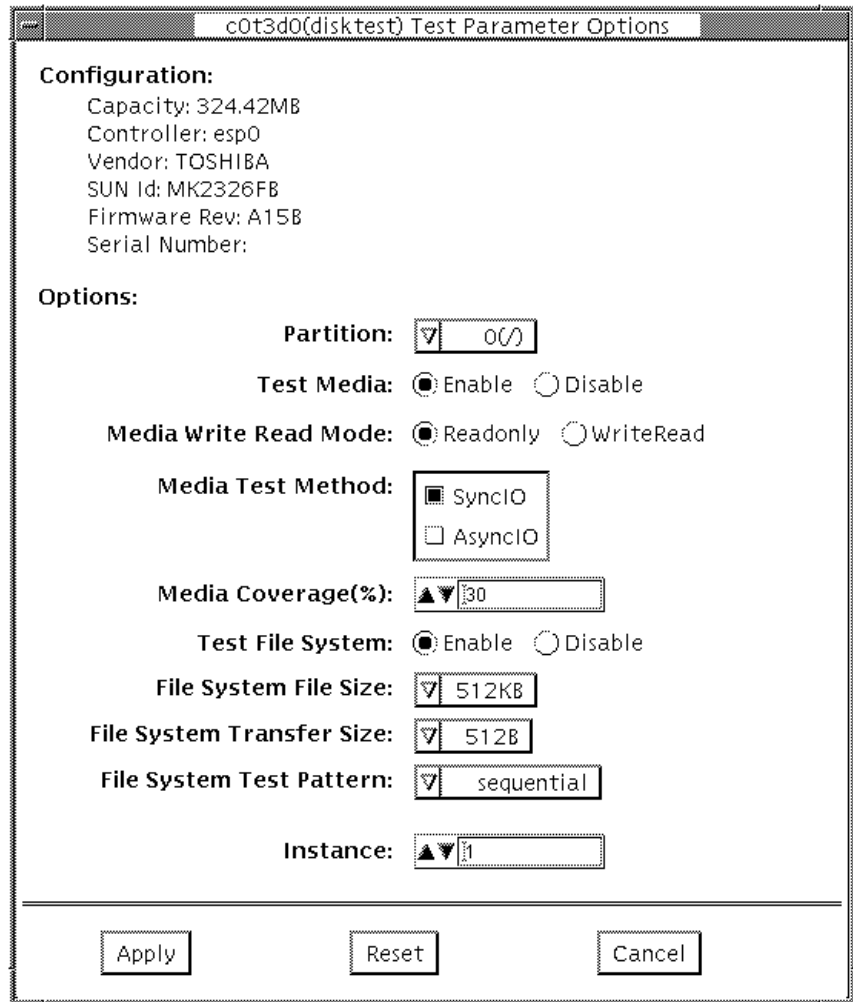
FIGURE 9-1   `disktest` Test Parameter Options Menu

The `disktest` test (see TABLE 9-2) has different option menus for different test modes.

TABLE 9-2   `disktest` Configurations and Options

`disktest` Options	Description
Partition	Displays the partition for the Media subtest. If a partition is mounted, its mount point is appended after the partition number, such as `1(/usr)`, where `1` is the partition number, and `/usr` is the mount point.
Test Media	Enables or disables the Media subtest.
Media Write Read Mode	Enables Read-Only or Read mode after write, with or without backup.
Media Coverage (%)	Enables users to test all or part of a partition (in percentages).
Test File System	Enables or disables the File System subtest; File System File Size: creates a file system file size twice the size as what is specified.
File System Transfer Size	Displays the transfer size of the File System subtest. File System Test Pattern: Test pattern of File System subtest.
Connection Test for Hard Disk	Option Menu for hard disk partition: 0 - 7 [default] Test Media: [Enable~](fixed to Enable) Test Mode: [Read Only~](fixed to Read Only) Media Coverage(%): [1](default - can be changed) Media Transfer Size: [2KB] Test File System: [Disable~](fixed to Disable)

TABLE 9-2 `disktest` Configurations and Options (Continued)

disktest **Options**	**Description**
Online Mode for Hard Disk	Online Mode for hard disk partition: 0 - 7 [default] Test Media: [Enable]- [Disable] Test Mode: [Read-only~](fixed to Read-only) Media Coverage(%): [10] Media Transfer Size: [2KB] [10KB] [20KB] - [40KB] [60KB] Test File System: [Disable~](fixed to Disable)
Functional Test for Hard Disk	(under SCSI-devices group): partition: 0 - 7 [default] Test Media: [Enable]- [Disable] Media Write Read Mode: [Read-only]- [BackupWriteRead] [NoBackupWriteRead] Media Coverage(%): [30] Media Transfer Size: [2KB] [10KB] [20KB] - [40KB] [60KB] Test File System: [Enable] [Disable] - File System File Size:[512KB]-[2MB][8MB][20MB] [100MB] [200MB] File System Transfer Size:[512B]- [1024B][10KB][40KB] [100KB] File System Test Pattern:[sequential] [0x00000000] [0xffffffff] [0x5aa55aa5] [0xdb6db6db] [random]
Functional Test for Floppy Disk	(under Other-Devices group): partition: 0 - 7 [default] Test Media: [Enable]- [Disable] Media Write Read Mode: [Read-only]- [BackupWriteRead] [NoBackupWriteRead] Media Test Method Media Coverage(%): [30] Media Transfer Size: [2KB]- [10KB] [20KB] Test File System: [Enable] [Disable] - Floppy File Size:[512KB]- [2MB] [8MB] [20MB] [100MB] [200MB] Floppy Transfer Size:[512B]- [1024B] [10KB] File System Test Pattern:[sequential] [0x00000000] [0xffffffff] [0x5aa55aa5] [0xdb6db6db] [random] NOTE: The floppy `disktest` can only be run in Functional test mode.

`disktest` Test Modes.

TABLE 9-3   `disktest` Test Modes

Test Mode	Description
Connection Test	This mode is only available for the hard disk test. There is no Connection test for the floppy driver test. Only one instance of `disktest (which monitors UNIX error messages)` is allowed for each disk device. `disktest` displays messages and reports errors. The test also opens the hard disk, checks the disk configuration, reads a few blocks, and then closes the hard disk. No File System subtest is run. No Write option is available in Connection test.
Functional Test	This mode does not allow `disktest` to monitor UNIX error messages. More than one instance of `disktest` is allowed for one disk device. Both the File System and Media subtests can be run in Functional test.The floppy test can also be run in Functional test mode.
	If you invoke SunVTS through SyMON, the following Functional Test description applies: This mode is only available for the hard disk test. There is no Online mode for the floppy driver test. Only one instance of `disktest` is allowed for each disk device, which monitors UNIX error messages. `disktest` displays messages and reports errors.
	In this mode, `disktest` also opens the hard disk, checks the disk configuration, and executes the Media subtest. Next, `disktest` performs random seek checks. By default the media sub-test performs a read-only test on 30% of the partition under test. No File System subtest is run. When the test finishes, `disktest` closes the disk device being tested. No Write option is available in Online mode.

`disktest` Command Line Syntax

/opt/SUNWvts/bin/disktest *standard_arguments* **-o p=***n***,-i=***n***,**
-w=*n***,dev=***device_name***,partition=***0-7***,rawsub=***E/D***,rawrw=,**

`rawcover=,rawiosize=,fssub=,fssize=,fsiosize=, fspattern=`

TABLE 9-4 `disktest` Command Line Syntax

Argument	Explanation
`p=number`	Sets the number of passes for the test to run; default is 1.
`i=number`	Sets the number of total instances for the test; default is 1.
`w=number`	Determines which instance this test is assigned; default is 0.
`dev=`*device_name*	Specifies the name of the disk to be tested, such as c0t3d0.
`partition=`*0-7*	Specifies the partition number to test as `partition=6`(`/export/s6`) if mounted on partition 6.
`rawsub=Enable/Disable`	Enables or disables the media subtest.

`rawrw=`
`Readonly`
`BackupWriteRead`
`NoBackupWriteRead`
	Specifies the Media subtest Read and Write mode:
	-Read Only
	-Write, read then backup
	-Write, read without backup

`rawcover=`	Specifies media coverage from 0-100% of the partition.
`rawiosize=`*2KB,10KB* *20KB,40KB,60KB*	Specifies the media size to transfer.
`fssub=Enable/Disable`	Enables or disables the file system subtest.
`fspattern=`*data pattern*	Specifies the file system data pattern as sequential or random. {seq(uential)/0x0(0000000)/0xf(fffffff)/0xa (5a5a5a5)/ 0x5(a5a5a5a)/ran(dom)/0xd(b6db6db)}
`fssize=`*file system size*	Indicates the file system subtest size in kilobytes or megabytes: K/k/KB/kb:kilobytes, M/m/MB/mb: megabytes.
`fsiosize=`*file system I/O* *transfer size*	Indicates the size of the file system subtest I/O transfer in bytes or kilobytes: {512B/1024B/10KB/40KB/100KB}

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`disktest` Error Messages

TABLE 9-5   `disktest` Error Messages

Error Message	Probable Cause(s)	Recommended Action
6000 `Re-reading and recomparing block` *`number`* `on name`	Media error.	If the problem persists, call your authorized Sun service provider.
	Faulty cable, disk, or controller.	
6002 `Error on recomparing block` *`number`* `on` *`name`*	Media error.	If the problem persists, call your authorized Sun service provider.
	Faulty cable, disk, or controller.	
6004 *`name`* `failed on` *`name`*`, blk` *`number`*`:` *`error_message`*	Faulty cable, disk, or controller.	If the problem persists, call your authorized Sun service provider.
6006 `Compare error on` *`name`*`, block` *`number`*`, offset` *`number`*	Faulty cable, disk, or controller.	If the problem persists, call your authorized Sun service provider.
6008 `Compare error: Block` *`number`* `on` *`name`* `was written with a repeating hex pattern of` *`number`*	Media error.	If the problem persists, call your authorized Sun service provider.
	Faulty cable, disk, or controller.	
6010 `TIME OUT!`	System too busy.	Reduce the system load.
	Faulty cable, disk, or controller.	If the problem persists, call your authorized Sun service provider.
6012 `Couldn't close` *`name`*	System error.	
6014 *`name`* `read failed on disk, in-between blocks` *`number`* `and` *`number`*`:` *`name`*	Media error.	If the problem persists, call your authorized Sun service provider.
	Faulty cable, disk, or controller.	
6016 *`name`* `read failed due to unexpected end of media`		
6018 *`name`* `write failed on disk, in-between blocks` *`number`* `and` *`number`*`:` *`name`*	Media error.	If the problem persists, call your authorized Sun service provider.
	Faulty disk or controller.	
6020 *`name`* `write failed due to unexpected end of media:`		

TABLE 9-5　disktest Error Messages (Continued)

Error Message		Probable Cause(s)	Recommended Action
6022	*name* compare failed on *name*, block *number*, offset *number*	Media error.	If the problem persists, call your authorized Sun service provider.
		Faulty disk or controller.	
6024	Couldn't close name	Faulty disk or controller.	If the problem persists, call your authorized Sun service provider.
		System error.	
6026	failed automount *name* onto *name*:*error_message*	Un-supported file system.	Disable the file system subtest.
		System error.	Disable the file system subtest.
			If the problem persists, call your authorized Sun service provider.
6028	*error_message*		
8000	File system subtest only allow to run on off-line mode	Parameter error.	Re-enter the command line correctly.
8002	malloc() failed	System load too heavy.	
		System error.	
8006	File operation error: Couldn't *name* file *name* on *name*: *error_message*	Lack of system resource.	Reduce the system load.
		Permission error.	Check directory or file permission.
		System error.	If the problem persists, call your authorized Sun service provider.
8008	Not enough free blocks on *name* (partition *number*)		Try another partition or delete files.
8010	No writable partition on *name*		Try another partition.
8012	No file system on device: *name*		Use newfs to create the file system.
			Disable the file system subtest.

TABLE 9-5   `disktest` Error Messages (Continued)

Error Message	Probable Cause(s)	Recommended Action
8014 `Couldn't run fstest on any partition of device:` *name*`(except 2)`	File system not supported.	Disable the file system subtest.
	System error.	If the problem persists, call your authorized Sun service provider.
8016 `Cannot open device:` *name*	Parameter error.	Enter the correct name of disk as `dev=c0t3d0`.
	System error.	If the problem persists, call your authorized Sun service provider.
8018 `Couldn't mount /dev/dsk/` *name*	Parameter error.	Enter the correct name of disk as `dev=c0t3d0`.
	System error.	If the problem persists, call your authorized Sun service provider.
8020 `Couldn't remove temporary dir` *name*: *error_message*	System error.	If the problem persists, call your authorized Sun service provider.
8022 `fail get semaphore:` *error_message*	Lack of system resource.	Reduce the system load.
	System error.	If the problem persists, call your authorized Sun service provider.
8024 `failed lock mtab semaphore: "semap" system call failure, errmsg` *error_message*	`disktest` instances exceed system semaphore operation limitation (default system limit for seminfo_semmnu=30)system software error.	Add the line "set semsys: seminfo_semmnu=0x100" to your `/etc/system` file and reboot the machine. If the problem persists, call your authorized Sun service provider.
8026 `fail unlock mtab semaphore:` *error_message*	System error.	If the problem persists, call your authorized Sun service provider.
8028 `failed get_volmgr_name()`		
8030 `Couldn't open` *name*: *error_message*	Cable loose or disconnected.	Check the cable.
	Device off line or missing.	Check the device on line.
	Device not configured.	Configure the device.
8032 `Get file state` *name* `failed:` *error_message*	System error.	If the problem persists, call your authorized Sun service provider.

TABLE 9-5 `disktest` Error Messages (Continued)

Error Message	Probable Cause(s)	Recommended Action
8034 `Couldn't open` *name*	Permission error.	Check the permissions.
	System error.	If the problem persists, call your authorized Sun service provider.
8036 `failed putmntent():` *error_message*	System error.	If the problem persists, call your authorized Sun service provider.
8038 `fail rename file`	Permission error.	Check permission.
	System error.	If the problem persists, call your authorized Sun service provider.
8040 `Couldn't get mount table entry for` *name*		
8042 `Invalid device name in` *name*	Parameter error.	Enter the correct name of disk as `dev=c0t3d0`.
8044 `Message table overflow on device` *name*	Parameter error.	Enter the correct name of disk as `dev=c0t3d0`.
	System error.	If the problem persists, call your authorized Sun service provider.
	Program error.	
8046 `Invalid controller id` *name* `for` *name*	Parameter error.	Re-enter the command line correctly.
8048 *name*, `disk DKIOCINFO ioctl:` *name*	Controller is not supported.	Disable the file system subtest.
	System error.	If the problem persists, call your authorized Sun service provider.
8050 `failed getting shared memory for monitor` *number*: *error_message*:	Lack of system resource.	Reduce the system load.
	System error.	If the problem persists, call your authorized Sun service provider.
8052 `failed attaching monitor shared memory:` *error_message*	Lack of system resource.	Reduce the system load.
	System error.	If the problem persists, call your authorized Sun service provider.

TABLE 9-5 `disktest` Error Messages (Continued)

Error Message	Probable Cause(s)	Recommended Action
8053 `failed create monitor shared memory` *number*:	Lack of system resource.	Reduce the system load.
	System error.	If the problem persists, call your authorized Sun service provider.
8054 `Shm remove error:` *error_message*	System error.	If the problem persists, call your authorized Sun service provider.
8056 `New error messages in system log. Extracted below:` *error_message*	Faulty cable/disk/controller.	See /var/adm/messages for more information.
		If the problem persists, consult your authorized Sun service provider.
8060 `fails to open any partition of` *name*	File system not mounted.	Mount the file system then run the test.
		Disable the file system subtest.
		If the problem persists, call your authorized Sun service provider.
8060 `failed getting semaphore:` *error_message*	System error.	If the problem persists, call your authorized Sun service provider.
8062 `failed getting shared memory:` *error_message*	System error.	If the problem persists, call your authorized Sun service provider.
8064 `failed attaching shared memory:` *error_message*	System error.	If the problem persists, call your authorized Sun service provider.
8068 `failed unlocking semaphore:` *error_message*	System error.	If the problem persists, call your authorized Sun service provider.
8070 `fail open floppy disk`	No floppy disk in drive.	Check the floppy disk.
8072 `failed FDIOGCHAR on floppy ioctl`	System error.	If the problem persists, call your authorized Sun service provider.
8074 `ioctl DKIOCGGEOM failed`	File system not supported.	If the problem persists, call your authorized Sun service provider.
	System error.	

TABLE 9-5 `disktest` Error Messages (Continued)

Error Message	Probable Cause(s)	Recommended Action
8074 `ioctl DKIOCGAPART failed`	System error.	If the problem persists, call your authorized Sun service provider.
8076 `fail initialize shared memory`	System error.	If the problem persists, call your authorized Sun service provider.
8078 `Illegal parameters on Online Mode`	Parameter error.	Enter the correct option.
8080 `Illegal parameters on Conn. Mode`	Parameter error.	Enter the correct option.
8082 `failed get_volmgr_name()`	No floppy disk in drive.	Check the floppy disk.
8084 `Mounted file system on device!`	Not enough disk space.	Reduce the system load.
	Permission error.	Check directory or file permissions.
	System error.	If the problem persists, call your authorized Sun service provider.
8086 `Invalid controller id` *number* `from` *name*	Parameter error.	Enter the correct name of disk as `dev=c0t3d0`.
8088 `Couldn't get file system information on` *name* `errno=`*number*	System error.	If the problem persists, call your authorized Sun service provider.
8090 `ioctl() failed on` *name*`:` *name*	System error.	If the problem persists, call your authorized Sun service provider.
8092 `Fail to get disk information`	No supported disk.	No UNIX message monitor feature in this disk(2).
	System load is too heavy.	Reduce the system load
8094 `Failed to create a message base for controller` *number*	No disk under this controller.	Connect the disk driver then run the test again.
8096 `fail to make directory` *name*	Lack of system resource.	Reduce the system load.
	Permission error.	Check directory or file permissions.
	System error.	If the problem persists, call your authorized Sun service provider.
8098 `Failed to mount partition`	Does this disk only have partition 2.	Partition 2 must be manually mounted.

TABLE 9-5 `disktest` Error Messages *(Continued)*

Error Message	Probable Cause(s)	Recommended Action
8010 `r/w is on /dev/r`*name*	System error.	If the problem persists, call your authorized Sun service provider.
8102 write/read count error	Out of disk space.	Try another partition or delete some files.
8104 failed to create buffer	Not enough memory.	Reduce the system load. If the problem persists, call your authorized Sun service provider.
8106 `failed aioread() :` *offset error number error* `string`	Faulty cable, disk, or controller.	If the problem persists, call your authorized Sun service provider.
8108 `failed aiowrite() :` *offset error number error string*	Faulty cable, disk, or controller.	If the problem persists, call your authorized Sun service provider.
8110 `failed io operation() :` *number text*	Faulty cable, disk, or controller.	If the problem persists, call your authorized Sun service provider.

ECP 1284 Parallel Port Printer Test (`ecpptest`)

The `ecpptest` verifies the functionality of the `ecpp`(7) IEEE 1284 parallel printer port device.

The `ecpp`(7) device is an *exclusive use device*. Only one application can interface with it at a time. This test is not scalable.

Printer Test Hardware and Software Requirements

The `ecpp`(7) driver is installed with the OS if the system being tested supports the `ecpp`(7) device. To run the optional printer subtest, a Centronics or ECP-mode printer must be attached to the printer port. To run the optional external loopback test, a passive loopback connector must be installed on the printer port.

Note – The external loopback test is intended for Sun internal manufacturing use only. It requires a custom loopback connector not available to an external customer.

`ecpptest` Subtests

TABLE 10-1 `ecpptest` Subtests

Subtest	Description
Internal Test FIFO Loopback	Verifies DMA and PIO accesses to the `ecpp` device. It uses an internal test fifo on the `ecpp` device. There are no printer or loopback connectors required. This test is always executed.
External Passive Loopback	This verifies the parallel port I/O connections to the back panel connector. This test requires a passive loopback connector (Sun part no. 270-2965-01). This test is disabled by default and must be manually enabled by the user. Note: The external loopback test is intended for Sun internal manufacturing use only. It requires a custom loopback connector that is not available to an external customer.
Printer Test	This verifies the parallel port printer operation. It will output a half page of ASCII character data. The output mode (for example, ECP and Centronics) is determined by which mode the printer and `ecpp` driver automatically negotiate. It will not change the current mode. The user must verify that data printed properly. This test is disabled by default and must be manually enabled by the user.

`ecpptest` Options
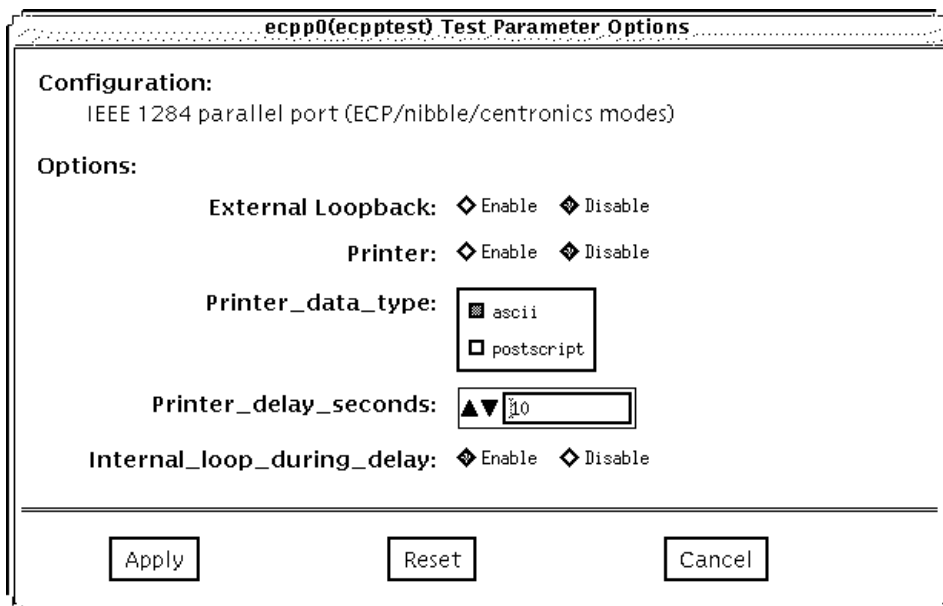
FIGURE 10-1 `ecpptest` Option Menu

TABLE 10-2 `ecpptest` Options

`ecpptest` **Options**	**Description**
External Loopback	`Enables or disables the external loopback test.` `This test requires a special loopback plug`

Printer	`Enables or disables the printer test; this test`
	`requires a printer to be attached to the parallel`
	`port`

Printer_data_type	Choose whether ascii text or postscript data is sent to the printer. The printer test must be enabled for this to be meaningful. A postscript printer must be attached to print postscript data.
Printer_delay _seconds	Allows the user to choose a delay between passes of the printer test. This prevents continuous printing of data that could quickly empty the paper supply. This is only meaningful if the printer test is enabled.
Internal_loop_durin g_delay	Enables the Internal Test FIFO loopback test during the printer delay time. This is only meaningful if the printer test is enabled.

`ecpptest` Test Modes

TABLE 10-3   `ecpptest` Test Modes

Test Mode	Description
Connection Test	Opens and closes the `ecpp`(7) device. No data is transferred. The test passes if the device can be open and closed successfully. The test also passes if the device cannot be opened because the device is busy with another process
Functional Test	Runs the internal loopback test and the user can optionally run the External loopback test and Printer test. The test will fail if the device is busy
	If you invoke SunVTS through SyMON, the following Functional Test description applies: Runs the internal loopback test. If the device cannot be opened because it is busy, then it is assumed that the device is successfully connected to another process

`ecpptest` Command Line Syntax

/opt/SUNWvts/bin/ecpptest *standard_arguments* **-o [dev=***device_name***]** **[ext_loop=***Enable*|*Disable***][printer=***Enable*|*Disable***][,data=***ascii*|*postscript***][,delay=***0-86400***][,dloop=***Enable*|*Disable***]**

TABLE 10-4   `ecpptest` Command Line Syntax

Argument	Explanation
dev=*device_name*	Specifies the name of the device. This should be of the form **/dev/ecpp#**, where # is the minor number of the device. The default device is /dev/ecpp0.
ext_loop=*Enable*|*Disable*	If enabled, the external loopback test is run. The External loopback plug must be attached to the printer port.
printer=*Enable/Disable*	If enabled, the printer test is run. A Centronics or ECP mode parallel port printer must be attached

TABLE 10-4 `ecpptest` Command Line Syntax

Argument	Explanation (Continued)
`data=`*ascii|postscript*	Choose whether ascii text or postscript data is sent to the printer. A postscript printer must be attached to print postscript data.
`delay=`*0-86400*	Allows the user to choose a delay between passes of the printer test. This prevents continuous printing of data that could quickly empty the paper supply.
`dloop=`*Enable|Disable*	Enables the Internal Test FIFO loopback test during the printer delay time.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`ecpptest` Error Messages

TABLE 10-5 ecpptest Error Messages

	Error Message	Probable Cause(s)	Recommended Action
8000	`External loop test and printer test cannot be enabled at the same time`	Operator error.	Examine system message files (`/var/adm/messages`) for other information.
8001	*device_name*`: Device does not exist or is otherwise inaccessible`	Correct `ecpp` driver not installed.	Examine system message files (`/var/adm/messages`) for other information.
		Faulty `ecpp` device or motherboard hardware.	
		System software problem.	
8002	`Must be super user (`*root*`) to execute`	Operator error.	Examine system message files (`/var/adm/messages`) for other information.

TABLE 10-5   ecpptest Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8003	Failed to open *device_name* (*error_message*)	Correct ecpp driver not installed.	Examine system message files (/var/adm/messages) for other information.
		Faulty ecpp device or motherboard hardware.	
		System software problem.	
8004	Failed ioctl *name*, device = *device_name*, error = *error_message*	Correct ecpp driver is not installed.	Examine system message files (/var/adm/messages) for other information.
		Faulty ecpp device or motherboard hardware.	
		System software problem.	
8005	Device not in correct mode, device = *device_name*, expected mode = *number*, observed mode = *number*	Correct ecpp driver is not installed.	Examine system message files (/var/adm/messages) for other information.
		Faulty ecpp device or motherboard hardware.	
		System software problem.	
8006	*device_name*: Status line miscompare, expected = *number*, observed = *number*, xor = *number*	Faulty or missing external loopback connector.	Examine system message files (/var/adm/messages) for other information.
		Correct ecpp driver is not installed.	
		Faulty ecpp device or motherboard hardware.	
		System software problem.	
8007	ECPP_nACK in wrong state, expected = *number*, observed = *number*, xor = *number*	Faulty or missing external loopback connector.	Examine system message files (/var/adm/messages) for other information.
		Correct ecpp driver not installed.	
		Faulty ecpp device or motherboard hardware.	
		System software problem.	

TABLE 10-5 ecpptest Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8008	*subtest_name*: Data miscompare, expected = *number*, observed = *number*, xor = *number*	Faulty or missing external loopback connector.	Examine system message files (/var/adm/messages) for other information.
		Correct ecpp driver not installed.	
		Faulty ecpp device or motherboard hardware.	
		System software problem.	
8009	Could not set correct port type, expected = *number* (*device_name*), observed = *number*, device = *device_name*	Correct ecpp driver not installed.	Examine system message files (/var/adm/messages) for other information.
		System software problem.	
		Faulty ecpp device or motherboard hardware.	
8010	*subtest_name*: Failed to allocate *number* bytes of memory, error = *error_message*	System resources temporarily not available, try again.	Examine system message files (/var/adm/messages) for other information.
		Correct ecpp driver not installed.	
		Faulty ecpp device or motherboard hardware.	
		System software problem.	
8011	*subtest_name*: Write() failed, device = *device_name*, error = *error_message*	Correct ecpp driver not installed.	Examine system message files (/var/adm/messages) for other information.
		Faulty ecpp device or motherboard hardware.	
		System software problem.	
8012	*subtest_name*: Write count is wrong, expected = *number*, observed = *number*	Correct ecpp driver is not installed.	Examine system message files (/var/adm/messages) for other information.
		Faulty ecpp device or motherboard hardware.	
		System software problem.	

TABLE 10-5   ecpptest Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8013	*subtest_name*: Data miscompare, expected = *number*, observed = *number*, xor = *number*, offset = *number*	Correct ecpp driver not installed	Examine system message files (/var/adm/messages) for other information.
		Faulty ecpp device or motherboard hardware.	
		System software problem.	
8014	*device_name*: Device busy	Another process is using the device.	Make sure no other process is using the device.
		Someone is printing with the device.	Make sure no one is printing to the device.
		Faulty ecpp device or motherboard hardware.	Examine system message files (/var/adm/messages) for other information.
		System software problem.	
8015	Bad state for device *name*, expected mode = *number*, observed mode = *number*		
8016	ecpp device busy: device *name*, expected mode = *number*, observed mode = *number*		
8017	Paper out: device *name*, expected mode = *number*, observed mode = *number*		
8018	Printer offline: device *name*, expected mode = *number*, observed mode = *number*		

TABLE 10-5 ecpptest Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8019	Printer error: device *name*, expected mode = number, observed mode = *number*		
8020	Printer unplugged: device *name*, expected mode = *number*, observed mode = *number*		
8021	Unrecognized printer error: device *name*, expected mode = *number*, observed mode = *number*		

Sun Enterprise Network Array Test(`enatest`)

`enatest` is used to provide configuration verification, fault isolation, and repair validation of the Sun Enterprise Network Array.

The Sun Enterprise Network Array is a high availability mass storage subsystem consisting of:

- SCSI fibre channel protocol host adapters with dual 100-Megabyte FC-AL ports.

- A disk enclosure.

- A Front panel display for configuration information.

- Up to two interface boards in the enclosure, which provide FC-AL connections to the enclosure and also provide status information and control of the conditions within the enclosure.

- Other field-replaceable units (FRUs) within the enclosure include power supply units, fan trays and backplane.

`enatest` detects all Sun Enterprise Network Array enclosures connected to the host and collects relevant configuration information. FIGURE 11-1 shows the Options menu which contains a sample configuration listing and test parameters. TABLE 11-1 describes the extent of the test coverage and provides samples of the configuration information that is displayed.

TABLE 11-1 `enatest` Coverage

Test Coverage	Description
Host Connections	`enatest` searches for all the active and inactive connections between the host and the enclosure and reports the number of existing active connections. If the VERBOSE mode is enabled, the socal port on the host side and the GBIC port on the enclosure side is reported for each active connection. The test also diagnoses any inactive connection(s) and reports the possible causes for the failure. The test will fail if there are one or more inactive connections. See the section on "Fault Isolation Capability" on page 114" for more information.

```
Sample Output:

SUNWvts.enatest.1010 06/05/97 13:48:53 enatest ses0 VERBOSE:
"MYBOX: Lower-Right GBIC connected to host via /devices/sbus@1f,0/
SUNW,socal@0,0:1"
SUNWvts.enatest.1006 06/05/97 13:48:53 enatest ses0 VERBOSE:
"MYBOX: Interface Board (Bottom one in the enclosure) detected to be
installed
and OK"
SUNWvts.enatest.6023 06/05/97 13:48:53 enatest ses0
ERROR: "MYBOX: Cannot communicate with the enclosure via
/devices/sbus@1f,0/SUNW,socal@0,0:0; possibly connected to Lower-Left
GBIC in the enclosure"
Probable_Cause(s):
    (1)Signal too low at the GBIC module in the enclosure
    (2)Faulty cable or cable disconnected
    (3)Faulty GBIC module on the host side
Recommended_Action(s):
    (1)Ensure the cables are properly connected
    (2)Please contact your service representative

SUNWvts.enatest.2006 06/05/97 13:48:53 enatest ses0 INFO:
"MYBOX: Number of connections to the host: 1"
```

Test Coverage	Description
Disk Access	During the testing, each disk is accessed through each active connection leading to that disk. The `enatest` opens partition 2 on the disk and reads 512 bytes of raw data. If there are any failures, the test tries to isolate the fault to either an enclosure element, the cable, the host adapter card or the OE module on the host adapter. See the section on "Fault Isolation Capability" on page 114"for more information.

TABLE 11-1 `enatest` Coverage

Test Coverage	Description
Enclosure Status	The status of the enclosure is obtained by querying the SCSI Enclosure Services (SES) device in the enclosure. Detailed information regarding the status of the elements within the enclosure is reported. The test fails if a critical condition is detected in the enclosure. The table below shows how the status information is reported.

TABLE 11-2 Element Enclosure Status

Enclosure Element	Information
Disk	Fault Sensed: Yes/No Status of ports A and B: Connected or Bypassed
Power Supply	Status: ON/OFF Temperature: OK/Critical Overtemp/Abnormal AC Input: OK/Not OK DC Output: OK/Not OK
Fan	Status: On/Off Speed: High/Low/Stopped
Backplane	Status: OK/Failed Status of ports A and B: Connected/Bypassed
Interface Board	Temperature: OK/Critical Overtemp Loop A status: OK/Failed Loop B status: OK/Failed
GBIC	Status: Disabled/Enabled Signal Level: OK/Too low Transmitter: OK/Failed

`enatest` Options

```
Configuration:
     Product: SENA        1.01
     Interface Board Firmware rev: 1.01
     DISKS(Front):2 Installed
       0: Installed
       1: Not Installed
       2: Not Installed
       3: Not Installed
       4: Not Installed
       5: Not Installed
       6: Installed
     DISKS(Rear):2 Installed
       0: Installed
       1: Not Installed
       2: Not Installed
       3: Not Installed
       4: Not Installed
       5: Not Installed
       6: Installed
     Interface Boards: 2 Installed
       Top: Installed
         GBIC Modules:Not Installed on Right;Not Installed on Left
       Bottom: Installed
         GBIC Modules:Installed on Right;Installed on Left
     Power Supplies: 3 Installed
       Front: 2 Installed;   Rear: 1 Installed
     Fans: 2 Installed
       Front: Installed;   Rear: Installed
     Loop Configuration: Single Loop

Options:
                              Disk Access:  ○ enable  ○ disable
                  Display Enclosure Status:  ○ enable  ○ disable
         Enclosure Services Functional Test:  ○ enable  ○ disable
                     Check All Connections:  ○ enable  ○ disable
                               Pass Delay:  ▲▼ 30

                                            Bound to:
                         Processor Affinity:  Processor 0
                                             Processor 1
```

FIGURE 11-1 `enatest` Option Menu

TABLE 11-3 `enatest` Options

`enatest` **Options**	**Description**
Enclosure Services Functional Test (General Description)	Certain control operations are performed on devices in the enclosure through the SES device and verified that the operation was performed successful. This functional test involves the following steps: **(a)** Perform control operation. **(b)** Verify control operation was successful. **(c)** Restore state to what it was before (a). **(d)** Verify restore operation was successful. The test will fail if any one of the above steps fails. This test targets the Disks and the Fans in the enclosure.
Enclosure Services Functional Test (Detailed disk test description)	**(a)** Control Operation: Each port of the disk is toggled from its original state: a port that was originally connected will be bypassed and vice-versa. **(b)** Verify Control Operation: This is done in two ways. First, the new status of the disk ports is verified by reading the status through the SES device. The test will fail if the status read back does not reflect the change. Next, disk access is attempted through the port that was originally connected but has now been bypassed. The test will fail if the access attempt is successful. **(c)** Restore State: The port states are restored to what they were before the Control Operation. **(d)** Verify Restore Operation: This is done in two ways. First the status of the disk ports is verified by reading the status through the SES device. The test will fail if the status read back does not reflect the change. Next, disk access is attempted through the port that have been reconnected. The test will fail if the access attempt is unsuccessful.
Enclosure Services Functional Test (Detailed Fan test description)	**(a)** Control Operation: The speed of each fan is toggled. Possible speeds are HIGH and LOW. **(b)** Verify Control Operation: The status is read back through the SES device and the speeds are compared. Failure to vary the speed in this case will result in an INFO message indicating that the fan speed could not be changed but does not result in a test failure. This is because SES can ignore fan speed change requests if required because of existing ambient temperature conditions. **(c)** Restore State: Restore the fan speed to the original speed. **(d)** Verify Restore Operation: This is similar to the Verify Control Operation step above.

Fault Isolation Capability

In the case of a failure, the test aids in fault isolation by reporting the possible cause(s) of failure. The fault isolation capability varies depending on the nature of the fault and the system configuration. enatest can detect and isolate hard faults. The following table shows the fault isolation capability for different configurations. A Yes indicates that fault isolation capability is available for that component in that configuration and a No indicates lack of fault isolation capability for that component in that configuration.

TABLE 11-4   enatest Fault Isolation Configurations

Connections to Enclosure	System Architecture	SOC+ Host Adapter	Host Side GBIC or Cable	Enclosure Elements			
				Disk	Backplane	IB	GBIC
Multiple	sun4u	Yes	Yes	Yes	Yes	Yes	Yes
Multiple	sun4d	Yes	No	Yes	Yes	No	No
Single	sun4u	Yes	No	No	No	No	No
Single	sun4d	Yes	No	No	No	No	No

`enatest` Test Modes

TABLE 11-5 `enatest` Test Modes

Test Mode	Description
Connection Test	In this mode, the host connections and the status of the enclosure are checked. The test fails if there are any broken connections or if a critical enclosure condition is detected. Non-critical conditions result in a warning. A sample of the output follows.

```
Connection test starting....

    ses0
        Status: Connected
        Enclosure:
            Product Anemones Enterprise Network Array,
            Enclosure Name=MYBOX,
        Host Connections:
            Number of Active Connections=2,
        Enclosure State:
            Critical Conditions=None, Non-Critical Conditions=None

Connection test complete
```

Functional	When invoked through SyMON, The Enclosure Services Functional Test is not allowed and the Disk Access option is disabled by default. When the test is invoked through regular SunVTS GUI, all the options are allowed in this mode.

`enatest` Command Line Syntax

/opt/SUNWvts/bin/enatest *standard_arguments* **-o dev=***device_name***,
disk_access=***enable | disable***,disks=***disk1:disk2:disk3:...diskn***,disp=***enable | disable,*
esfunc=*enable | disable*

TABLE 11-6 `enatest` Command Line Syntax

Argument	Explanation
dev	The name of a ses device in the enclosure.

TABLE 11-6   `enatest` Command Line Syntax

Argument	Explanation (Continued)
disk	The disks will be accessed only if this option is enabled.
disks	This argument consists of a ":" separated list of disks contained within this enclosure, that the test will attempt to access if the disks_access option is enabled. If this option is absent the test will access all the disks that it can find in the enclosure.
disp	Detailed status information regarding the enclosure elements will be displayed only if this option is enabled.
esfunc	The Enclosure Services Functional Test will be performed only if this option is enabled.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`enatest` Error Messages

This table describes the `enatest` error messages, probable causes, and recommended actions.

TABLE 11-7   `enatest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	*enclosure_name*: `Unable to access disk` *diskname* (*enclosure name*, *Front*|*Rear*, `Slot`)	Bad disk.	`Contact your authorized Sun service representative.`
6001	`Unable to access any disk through host adapter:` *host adapter device*	Bad host adapter.	Run SunVTS `socaltest` on the host adapter.

TABLE 11-7 `enatest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6003	*enclosure_name*: `Unable to access disk` *diskname* (*enclosure_name*, *Front*|*Rear*, `Slot`) `through` *Host adapter device name*. `This disk is still accessible through other path(s)`	Port A or B of the disk is faulty.	Contact your authorized Sun service representative.
6004	*enclosure_name*: `Failed to determine the location of disk` *diskname*	Invalid device name.	Contact your authorized Sun service representative.
6005	*enclosure_name*: `Could not access` *diskname* (*enclosure_name*, *Front*|*Rear*, `Slot`) `through` *Host adapter device name*	Host adapter is faulty.	Contact your authorized Sun service representative.
6006	*enclosure_name*: `Disk` *diskname* `cannot be accessed through` *Host adapter device name*	Critical failure in the Interface Board (in the enclosure) through which this disk is accessed.	Contact your authorized Sun service representative.
6007	*enclosure_name*: `Disk` *diskname* `cannot be accessed through` *Host adapter device name*	Interface Board (in the enclosure) through which this disk is accessed is not installed.	
6008	*enclosure_name*: `Disk` *diskname* `cannot be accessed through` *Host adapter device name*	Critical failure (transmitter fault) in the GBIC module (in the enclosure) through which this disk is accessed.	Contact your authorized Sun service representative.
6009	*enclosure_name*: `Disk` *diskname* `cannot be accessed through` *Host adapter device name*	`Loss of light at the GBIC module (in the enclosure) through which this disk is accessed.`	Contact your authorized Sun service representative.
		`Faulty cable.`	
		`Faulty GBIC module on the host side.`	
6013	*enclosure_name*: `Could not access` *diskname* (*enclosure name*, *F*ront|*Rear*, `Slot`) `through` *Host adapter device* `name`	Faulty GBIC module on host side.	
		Bad cable connection.	
6014	*enclosure_name*: `Could not access` *diskname* (*enclosure name*, `Front|Rear, Slot`) `through` *Host adapter* `device name`	Faulty Host Adapter.	
		Faulty GBIC on host side.	
		Faulty cable.	

TABLE 11-7 `enatest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6015	*enclosure_name*: `Could not` `access` *diskname* `through` *Host adapter device name*	Disk is not installed.	
6016	*enclosure_name*: `Could not` `access` *diskname* `through` *Host adapter device name*	Disk detected to be installed, but may be bypassed or powered off.	
6017	*enclosure_name*: `Disk` *diskname* `cannot be accessed` `through` *Host adapter device name*	Critical failure in the enclosure backplane through which this disk is accessed.	`Contact your authorized` `Sun service` `representative.`
6018	*enclosure_name*: `Disk` *diskname* `cannot be accessed` `through` *Host adapter device name*	Enclosure backplane through which this disk is accessed is not installed.	
6019	*enclosure_name*: `Disk` *diskname* `cannot be accessed` `through` *Host adapter device name*	GBIC module (in the enclosure) through which this disk is accessed is not installed.	
6020	*enclosure_name*: `Cannot` `communicate with the` `enclosure via` *Host adapter device name*	Critical failure in the Interface Board (top/bottom one in enclosure).	`Contact your authorized` `Sun service` `representative.`
6021	*enclosure_name*: `Cannot` `communicate with the` `enclosure via` *Host adapter device name*	Interface board (top or bottom) is not installed in the enclosure.	
6022	*enclosure_name*: `Cannot` `communicate with the` `enclosure via` *Host adapter device name*`; possibly` `connected to` *location* `GBIC` `in the enclosure`	Critical failure (transmitter fault in the GBIC module in the enclosure).	`Contact your authorized` `Sun service` `representative.`
6023	*enclosure_name*: `Cannot` `communicate with the` `enclosure via` H*ost adapter device name*`; possibly` `connected to` *location* `GBIC` `in the enclosure`	Signal level too low at the GBIC module in the enclosure.	Ensure the cables are properly connected. `Contact your` `authorized Sun service` `representative.`
		Faulty cable or cable disconnected.	
		Faulty GBIC module on the host side.	
6024	*enclosure_name*: `Cannot` `communicate with the` `enclosure via` *Host adapter device name*	GBIC module (*location*) is not installed in the enclosure.	

TABLE 11-7 `enatest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6026	*enclosure_name*: `Failed to` *Restore|Toggle* `disk` (*Front|Rear*, `slot` *slot_number*`) state.` `Expected:` `Port A:` *Bypassed|Connected;* `Port B:` *Bypassed|Connected;* `Observed:` `Port A:` *Bypassed|Connected;* `Port B:` *Bypassed|Connected*	Error in Enclosure Services or Enclosure Element.	`Contact your authorized` `Sun service` `representative.`
6027	*enclosure_name*: `Unable to` `access disk` *diskname* (*Front|Rear* `slot` *slot_number*`)` `through port` *A|B* `even` `after bringing the port` `online`	Error in Enclosure Services or Enclosure Element.	`Contact your authorized` `Sun service` `representative.`
6028	*enclosure_name*: `Disk access` `succeeded for` *diskname* (*Front|Rear* `slot` *slot_number*`)` `through port` *A|B* `even after requesting` `that the port be bypassed`	Error in Enclosure Services or Enclosure Element.	`Contact your authorized` `Sun service` `representative.`
6035	*enclosure_name*: `Cannot` `access enclosure through` *Host adapter device name*	Critical GBIC failure in the enclosure; location GBIC: disabled|enabled Signal: *signal_level*; Transmitter: *Failed|OK*; *location* GBIC: *disabled|enabled* Signal: *signal_level*; Transmitter: *Failed|OK*	`Contact your authorized` `Sun service` `representative.`
6036	*enclosure_name*: `Critical` `condition detected in` `enclosure!`		Run this test in 'Functional' mode for more information. `Contact your authorized` `Sun service` `representative.`
8001	`Could'nt determine the` `host adapter through` `which diskname being` `accessed`	Software error.	`Contact your authorized` `Sun service` `representative.`
8003	`Internal error`	Software error in test. system error message.	`Contact your authorized` `Sun service` `representative`

TABLE 11-7   `enatest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8004	*enclosure_name*: `Could not communicate with enclosure via` *Host adapter device name*	Bad connection.	Ensure the cables are properly connected. Contact your authorized Sun service representative.
8005	`Could not communicate with the enclosure`	Faulty connection.	Ensure the cables are properly connected. Run SunVTS `socaltest` to verify the host adapter. Contact your authorized Sun service representative.

StorEdge 1000 Enclosure Test (`enctest`)

`enctest` tests the StorEdge 1000 enclosures. The enclosure can support either 12 1" 4GB drives or 8 1.6" 9GB drives and have redundant power and cooling. Two enclosure models are available:

■ StorEdge A1000 – Disk Tray with the hardware RAID controller

■ StorEdge D1000 – Disk Tray without the hardware RAID controller.

`enctest` can be used for validation, configuration verification, repair verification, and fault isolation of both models.

The `enctest` probe detects all the connected StorEdge enclosures and displays the status of the various elements in the enclosure.

`enctest` sets the StorEdge D1000 LEDs as follows:

TABLE 12-1   `enctest` Status LEDs

LED State	Indicated Condition
On	Unrecoverable or critical condition
Blinking	Non-critical condition
Off	Informational or no condition to report

Note – NOTE: The `ses` driver must be present on the system with the StorEdge enclosure(s).   If the `ses` driver is not present, SunVTS will not detect the StorEdge hardware. Use the following command to see if the `ses` driver is present:
pkginfo SUNWses
Install the package if necessary.

`enctest` reports the status of the various elements in the enclosure. An error is registered if an unrecoverable or critical condition is detected. Non-critical conditions are reported through warning messages. The table below describes the information that is reported for each enclosure element.

TABLE 12-2 `enctest` Enclosure Status Report

Enclosure Element	Information reported (per device slot)
Disk	Present/Not Present Failed/OK
Power supply	Present/Not Present Failed/OK
Fan	Present/Not Present Failed/OK
Temperature	OK/Over temperature
RPA cache Battery (StorEdge A1000 only)	OK/Low Charge

enctest Options
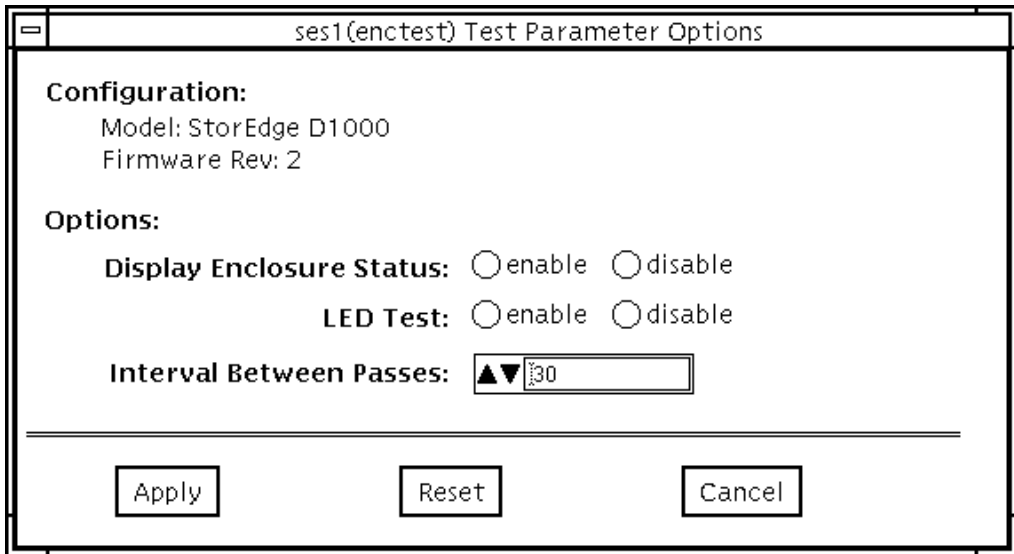
FIGURE 12-1 enctest Options

TABLE 12-3 enctest Options

enctest **Options**	**Description**
Display Enclosure Status	When enabled, the status of the enclosure elements are displayed as messages in the SunVTS console. These messages are also logged in the SunVTS info log file.
LED Test	When enabled, the LEDs associated with the disks are flashed for a short period of time.
Interval Between Passes	Sets the minimum delay between successive invocations of the test.

`enctest` Test Modes

TABLE 12-4 `enctest` Test Modes

Test Mode	Description
Connectivity Mode	The enclosure status summary bits are read. Detailed information is reported only if an unrecoverable or critical condition is detected.
Functional Mode	When started through Symon, the LED test is not allowed and the "Display Enclosure Status" option is disabled by default. When started through the regular SunVTS GUI, all options are allowed.

`enctest` Command Line Syntax

/opt/SUNWvts/bin/enctest *standard_arguments* **-o dev=***device_name,*
disp=*enable* | *disable,***led=***enable* | *disable,***delay=***delay_in_seconds*

TABLE 12-5 `enctest` Command Line Syntax

Argument	Explanation
dev	This is the name of a ses device in the enclosure.
disp	Detailed status information regarding the enclosure elements will be displayed if enabled
l*ed*	Option to enable or disable the LED test.
delay	Minimum delay between successive invocations of the test.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNVvts/bin/sparcv9/`*testname.* If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`enctest` Error Messages

This table describes the `enctest` error messages, probable causes, and recommended actions.

TABLE 12-6   `enctest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
8000	Failed to open device.	System error. Cable loose or disconnected or device not configured.	Verify the cable is properly connected and the device configured.Contact your authorized Sun service representative.
8001	Failed to communicate with the enclosure	System error. Cable loose or disconnected or device not configured.	Verify the cable is properly connected and the device configured.Contact your authorized Sun service representative.
8002	Cannot proceed further due to lack of memory	Heavy system load.	Please try later.
8003	*unrecoverable*|*critical*|*non-critical* informational condition detected in the enclosure		
8004	Failed to obtain list of enclosure elements	System error	Contact your authorized Sun service representative.
8005	Failed to obtain status of an enclosure element	System error	Contact your authorized Sun service representative.
8006	Failed to set the status of an enclosure element	System error	Contact your authorized Sun service representative.

Environmental Test (`envtest`)

`envtest` exercises the I2C bus on the Sun Enterprise 450 product line. `envtest` contains five subtests to test and report on the power supply status, system temperature status, fan speed, disk LEDs, and front panel and keyswitch. `envtest` is not scalable.

`envtest` Options

FIGURE 13-1 `envtest` Options Menu

TABLE 13-1 `envtest` Options

`envtest` **Options**	Description
Disk LEDs Test	Illuminates each LED on the disk backplane(s) to green, then amber, and then back to its original state. The test then illuminates all disk LEDs to green, then amber, and then back to their original state. This test is only enabled in Functional test mode.
Fan Test	Cycles each fanbank speed to low, medium, and high, then verifies the correct speed. Next, each fanbank is stopped, one at a time. The test then verifies that a fan fault has occurred. Next, the watchdog timer is invoked to simulate a catastrophic failure. The test verifies that the system set all fanbanks to high and then resets the fan speed to normal. This test is only enabled in Functional test.

TABLE 13-1   `envtest` Options

`envtest` **Options**	**Description**
Front Panel and Keyswitch Test	Flashes each individual LED on the front panel to ON (green or amber), then OFF, and then back to its original state. The test then illuminates all front panel LEDs then sets them back to their original state. The power on LED is Read Only and will not be cycled. The test then displays the current keyswitch position. This test is only enabled in Functional test mode.
Power Supply Status	Identifies the number of power supplies that are in the system, and the state of each power supply, and verifies that the power supply temperatures are within normal operating parameters. This test is enabled in all modes.
Temperature Status	Identifies the current temperature of each CPU in the system, and the ambient temperature of the system, and `envtest` verifies that all temperatures are within normal operating parameters. This test is enabled in all modes.

`envtest` Test Modes

`envtest` supports Connection and Functional tests.

TABLE 13-2   `envtest` Test Modes

Test Mode	**Description**
Connection Test	Reports the status of the power supplies, the temperature sensors within the system, and verifies normal operating parameters.
Functional Test	Tests the disk back panel, front panel LEDs, and fan control circuitry. Also uses the same functionality as Online mode and connection test If you invoke SunVTS through SyMON, the following Functional Test description applies: `envtest` runs in Connection Test mode (see description above).

`envtest` Command Line Syntax

/opt/SUNWvts/bin/envtest [*standard arguments*]
-o dev=*raw_device_name***,diskleds=***E/D***,fans=***E/D***,fpanel=***E/D***,**
psupply=*E/D***,temp=***E/D*

TABLE 13-3 `envtest` Command Line Syntax

Argument	Explanation
dev=*raw_device_name*	Specifies the name of the raw device to test.
diskleds=*enable|disable*	Enables or disables the diskleds test.
fans=*enable|disable*	Enables or disables the fans test.
fpanel=*enable|disable*	Enables or disables the front panel test.
psupply=*enable|disable*	Enables or disables the power supply test
temp=*enable|disable*	Enables or disables the temperature test.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`envtest` Error Messages

TABLE 13-4   `envtest` Error Messages

Error Message	Probable Cause(s)	Recommended Action
8000 `Unable to open driver` *driver_name*: *sys error msg*	Device name is wrong or driver is not loaded.	Check the device name (if entered) or load driver.
	Device busy.	Kill all other processes accessing driver.
8001 `kstat_open failed`	No kernel resource available.	Consult `/var/adm/messages` file for more information.
	System software error	
8002 `kstat_lookup failed for` *envtest structure* `in` *kstat module*	`envtest` module not found.	Check revision and reload `envctrl` driver.
	No kernel resource available.	Consult `/var/adm/messages` file for more information.
	System software error.	
8003 `kstat_read failed` *envtest structure* `in` *kstat module*	`envtest` module not found.	Check revision /reload `envctrl` driver.
	No kernel resource available.	Consult the `/var/adm/messages` file for more information.
	System software error.	
8004 `Unable to close kstat`	No kernel resource available.	Consult the `/var/adm/messages` file for more information.
	System software error	
8005 `Unable to set mode to DIAG using ioctl.:` *sys error msg*	System software error	Consult the `/var/adm/messages` file for more information.
8006 `Unable to set disk LEDs:` *sys error msg*	System software error	Consult the `/var/adm/messages` file for more information.
8007 `Unable to get disk LEDs:` *sys error msg*	System software error.	Consult the `/var/adm/messages` file for more information.
8008 `Unable to set` *fan type* `fan speed to` *speed*: *sys error msg*	Fan hardware failure.	Manually check the fan speed and replace the fan.
	System software error.	Consult the `/var/adm/messages` file for more information.
8009 `Unable to get` *fan type* `fan speed:` *sys error msg*	System software error.	Consult the `/var/adm/messages` file for more information.

TABLE 13-4 `envtest` Error Messages

Error Message	Probable Cause(s)	Recommended Action
8010 `fan type` fan speed: `speed` out of range	Fan hardware failure.	Manually check the fan speed and replace the fan.
	System software error.	Consult the `/var/adm/messages` file for more information.
8011 `Forced fan fault did not occur for` *fan type* `fanbank`	Hardware failure.	Replace the hardware.
	System software error.	Consult the `/var/adm/messages` file for more information.
8012 `Forced fan fault did not clear for` *fan type* `fanbank`	Hardware failure.	Replace the hardware.
	System over temperature.	Consult the `/var/adm/messages` file for more information.
	System software error.	
8013 `Unable to set watchdog timer`	Hardware failure.	Replace the hardware.
	System software error.	Consult the `/var/adm/messages` file for more information.
8014 `Unable to set temperature to` *temperature*	System software error.	Consult the `/var/adm/messages` file for more information.
8015 `Power Supply` *supply* `has failed: Limit` *0=PASS|1=FAIL* `Current Share` *0=PASS|1=FAIL* `Temperature` *temperature*	Power supply hardware failure.	Replace the power supply.
8016 `Unable to allocate memory using malloc. :` *sys error msg*	System software error.	Consult the `/var/adm/messages` file for more information.
8017 `Unable to set mode to NORMAL using ioctl. :` *sys error msg*	System software error.	Consult the `/var/adm/messages` file for more information.
8018 `Error encountered during call to` *function name*		
8019 `Unable to set front panel LEDs using ioctl.:` *sys error msg*	System software error.	Consult the `/var/adm/messages` file for more information.
8020 `Unable to reset front panel LEDs to original state`	System software error.	Consult the `/var/adm/messages` file for more information.
8021 `Unable to set front panel LEDs to` *led mask*	System software error	Consult the `/var/adm/messages` file for more information.

TABLE 13-4 `envtest` Error Messages

Error Message		Probable Cause(s)	Recommended Action
8022	`Ambient temperature:` *temperature* `Celsius out of range`	`envctrl` hardware error.	
8023	`CPU CPU # temperature:` *temperature* `Celsius out of range`	`envctrl` hardware error.	
8024	`Data Miscompare: ioctl diskled mask =` *diskled mask* `kstat diskled mask =` *diskled mask*	`envctrl` hardware error	
		System software error.	Consult the `/var/adm/messages` file for more information.
8025	`Illegal options passed to envtest` *option*	User error.	Check usage by typing `envtest -u`.
8026	Unable to retrieve power supply information.	I2C hardware failure	Replace power supply.
		I2C bus wedged	Check for other I2C devices that may be corrupting the I2C bus.
8027	Power Supply Over Temperature: PS Slot *slot* PS Temperature *temperature*.	Bad Power Supply.	Replace Power Supply.

Environmental Test (`env2test`)

`env2test` exercises and validates the I2C bus on the Sun Enterprise 250 systems.

Five subtests are in `env2test` to test and report the status of the power supply, system temperature, fan speed, disk LEDs, front panel, and keyswitch.
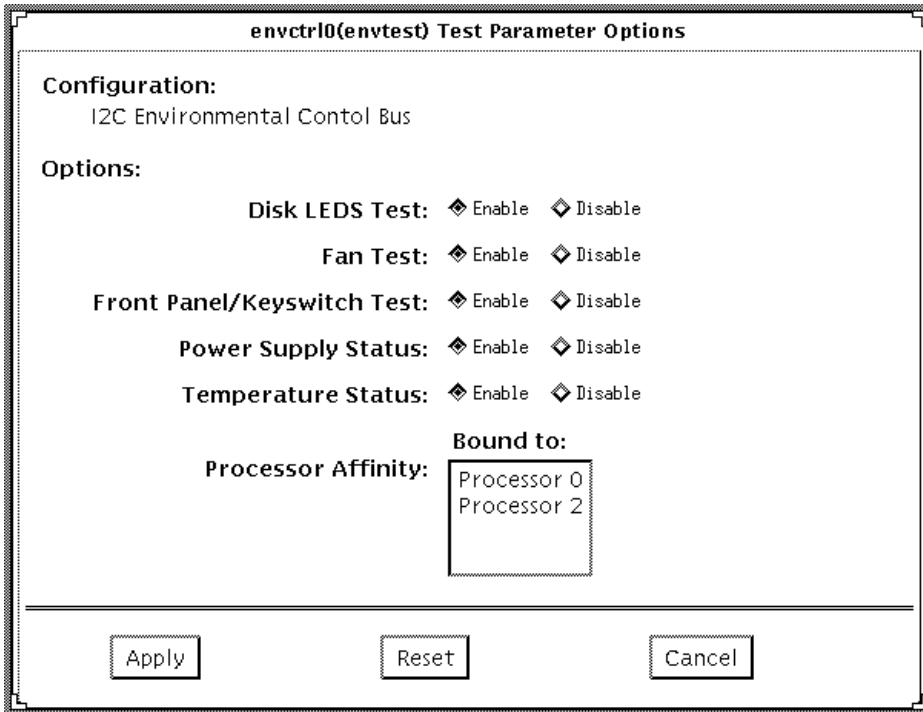
`env2test` is not scalable.

`env2test` Options

```
envctrltwo0(env2test) Test Parameter Options
```

Configuration:
    I2C Environmental Control Bus

Options:

Disk LEDS Test:	⦿ Enable ◯ Disable
Fan Test:	⦿ Enable ◯ Disable
Front Panel/Keyswitch Test:	⦿ Enable ◯ Disable
Power Supply Status:	⦿ Enable ◯ Disable
Temperature Status:	⦿ Enable ◯ Disable

Apply   Reset   Cancel

FIGURE 14-1 `env2test` Options Menu

TABLE 14-1 `env2test` Options

`env2test` **Options**	**Description**
Disk LEDs Test	Determines the number of disks that are present and the state of each disk. Illuminates each LED on the disk backplane to amber, and then back to its original state. The test then illuminates all disk LEDs to amber, and then back to their original state. This test is only enabled in Functional test.
Fan Test	Sets the fanbank speed to halfway between the current speed and max speed, then verifies the correct speed. Next, the fanbank speed is set to max speed, and the speed is verified. The fanbank speeds are then returned to normal, and then verified. This test is only enabled in Functional test.
Front Panel and Keyswitch Test	Flashes each individual LED on the front panel to ON (green or amber), then OFF, and then back to its original state. The test then illuminates all front panel LEDs then sets them back to their original state. The power on LED is read-only and will not be cycled. The test then displays the current keyswitch position. This test is only enabled in Functional test.
Power Supply Status	Identifies the number of power supplies that are in the system and the state of each power supply. This test is enabled in all modes.
Temperature Status	Identifies the current temperature of each CPU in the system, the ambient temperatures of the system, the temperature on the SCSI and power distribution boards, and verifies that all temperatures are within normal operating parameters. This test is enabled in all modes.

env2test Test Modes

env2test supports Connection and Functional tests as descripbed in the table below.

TABLE 14-2 env2test Test Modes

Test Mode	Description
Connection Test	Reports the status of the power supplies, the temperature sensors within the system, and verifies normal operating parameters
Functional Test	Tests the disk back panel, front panel LEDs, and fan control circuitry. Also uses the same functionality as Online mode and connection test
	If you invoke SunVTS through SyMON, the following Functional Test description applies: Uses the same functionality as connection test (above)

env2test Command Line Syntax

/opt/SUNWvts/bin/env2test [*standard arguments*]
-o dev=*device=name***,diskleds=***E/D***,env_mon=***poll_interval***,fans=***E/D***,fpanel=***E/D***,psupply=***E/D***,temp=**E/D

TABLE 14-3 env2test Command Line Syntax

Argument	Explanation
dev=raw_device_name	Specifies the name of the raw device to test
diskleds=enable|disable	Enables or disables diskleds test
env_mon=poll_interval	Displays all system environmental statics every "poll_interval" seconds. (Display ONLY, Does not test.)
fans=enable|disable	Enables or disables fans test
fpanel=enable|disable	Enables or disables front panel test
psupply=enable|disable	Enables or disables power supply test
temp=enable|disable	Enables or disables temperature test

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`env2test` Error Messages

TABLE 14-4 `env2test` Error Messages

Error Message	Probable Cause(s)	Recommended Action
8000 `Unable to open driver` *driver name*: *sys error msg*	Device name is wrong or driver is not loaded.	Check the device name (if entered) or load driver.
	Device busy.	Kill all other processes accessing driver.
8001 `kstat_open failed`	No kernel resource available.	Consult `/var/adm/messages` file for more information.
	System software error	
8002 `kstat_lookup failed for` *env2test structure* `in` *kstat module*	`env2test` module not found.	Check revision and reload `envctrl` driver.
	No kernel resource available	Consult `/var/adm/messages` file for more information.
	System software error.	
8003 `kstat_read failed` *env2test structure* `in` *kstat module*	`env2test` module not found.	Check revision /reload `envctrl` driver.
	No kernel resource available.	Consult the `/var/adm/messages` file for more information.
	System software error.	
8004 `Unable to close kstat`	No kernel resource available.	Consult the `/var/adm/messages` file for more information.
	System software error.	
8005 `Unable to set mode to DIAG using ioctl.: sys error msg`	System software error	Consult the `/var/adm/messages` file for more information.
8006 `Unable to set disk LEDs:` *sys error msg*	System software error	Consult the `/var/adm/messages` file for more information.
8007 `Unable to get disk LEDs:` *sys error msg*	System software error.	Consult the `/var/adm/messages` file for more information.

TABLE 14-4 `env2test` Error Messages

Error Message	Probable Cause(s)	Recommended Action
8008 `Unable to set fan speed to` *speed*`:` *sys error msg*	Fan hardware failure	Manually check the fan speed and replace the fan.
	System software error.	Consult the `/var/adm/messages` file for more information.
8009 `Unable to get fan speed:` *sys error msg*	System software error	Consult the `/var/adm/messages` file for more information.
8010 `Fan speed` *speed* `out of range: MIN` *speed* `MAX` *speed*	Fan hardware failure.	Manually check the fan speed and replace the fan.
	System software error.	Consult the `/var/adm/messages` file for more information.
8011 `Forced fan fault did not occur for` *fan type* `fanbank`	Hardware failure.	Replace the hardware.
	System software error.	Consult the `/var/adm/messages` file for more information.
8015 `Power Supply` *supply* `has failed: State` *state*	Power supply hardware failure.	Replace the power supply.
8016 `Unable to allocate memory using malloc.` `:` *sys error msg*	System software error.	Consult the `/var/adm/messages` file for more information.
8017 `Unable to set mode to NORMAL using ioctl.` `:` *sys error msg*	System software error.	Consult the `/var/adm/messages` file for more information.
8018 `Error encountered during call to` *function name*		
8019 `Unable to set front panel LEDs using ioctl.:` *sys error msg*	System software error.	Consult the `/var/adm/messages` file for more information.
8020 `Unable to reset front panel LEDs to original state`	System software error.	Consult the `/var/adm/messages` file for more information.
8021 `Unable to get front panel LEDs.`	System software error.	Consult the `/var/adm/messages` file for more information.
8022 `Ambient temperature` *temperature* `Celsius out of range`	`envctrl` hardware error.	
8023 `CPU CPU # temperature` *temperature* `Celsius out of range`	`envctrl` hardware error.	

TABLE 14-4 `env2test` Error Messages

Error Message	Probable Cause(s)	Recommended Action
8024 `Data Miscompare: ioctl diskled mask = ` *diskled mask* `kstat diskled mask = ` *diskled mask*	hardware error	
	System software error.	Consult the `/var/adm/messages` file for more information.
8025 `Illegal options passed to env2test` *option*	User error.	Check usage by typing **`env2test -u.`**
8026 Unable to retrieve power supply information.	I2C hardware failure.	Replace power supply.
	I2C bus wedged.	Check for other I2C devices that may be corrupting the I2C bus.
8033 Temperature *temperature* celsius out of range.	Hardware error.	Check all thermistors.

Frame Buffer Test (`fbtest`)

`fbtest` is a generic test for all dumb frame buffers used with the Solaris 2.x and Solaris 7 software.

The `fbtest` checks the frame buffer by sequentially writing, reading, and verifying small blocks of random patterns across the entire video RAM. The block size is 64 x 64 pixels. If a miscompare occurs, the test stops with an error message that indicates the location of the error.

If a generic frame buffer device name (`dvc/fb`) is specified, `fbtest` automatically detects the depth of the frame buffer, and adjusts testing to the frame buffer size.

`fbtest` Options
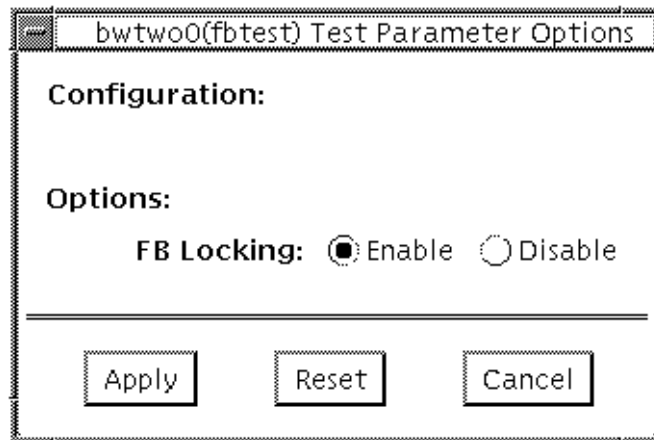
FIGURE 15-1 `fbtest` Option Menu

`fbtest` Command Line Syntax

/opt/SUNWvts/bin/fbtest *standard_arguments* **-o dev=***device_name* **lock=***E(nable)/D(isable)*

TABLE 15-1   `fbtest` Command Line Syntax

Argument	Explanation
dev=*device_name*	Specifies which frame buffer to be test.
lock=*E(nable)/ D(isable)*	Enables or disables the window system locking option. See the section about Testing Multiple Frame Buffers in the *SunVTS 3.0 User's Guide* for details. Frame buffer locking is enabled by default on the window server running the Open Windows software.

`fbtest` Test Modes

Due to the nature of graphic tests, reading from or writing to the frame buffer during graphic tests disturbs user operation. This test is only available in Functional test.

`fbtest` Error Messages

TABLE 15-2 `fbtest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	Data compare error	FB lock is disabled.	Enable FB lock.
		Bad frame buffer.	If the problem persists, call your authorized Sun service provider.
6001	Video memory error at *address*, expected: *number*, observed: *number*	FB lock is disabled.	Enable FB lock.
		Bad frame buffer.	If the problem persists, call your authorized Sun service provider.
6002	Unable to lock frame buffer	FB lock was enabled while the window system was not running.	
8000	can't acquire console semaphore	System load is too heavy.	Reduce the system load.
		System error.	If the problem persists, contact your Sun service provider.
8001	Parameter error – Improper test parameters supplied	fbtest can only run in Functional test.	
8002	Device name missing!	Parameter error.	Must provide device name
8003	Create_Raster() failed for *name*	Parameter error.	
		No such device.	
8004	Create_Raster() failed for type RAS_CHILD	Operator error.	Check test options, mode and retry, if appropriate.
		Software error.	Contact your Sun service provider.
8005	Malloc failed	System load too heavy.	Reduce the system load
		System error	If the problem persists, call your authorized Sun service provider.

Fast Frame Buffer Test (`ffbtest`)

`ffbtest` verifies the functionality of the Fast Frame Buffer.

`ffbtest` can detect and adapt to the video modes of single- and double-buffer versions of the fast frame buffer (FFB). All `ffbtest` tests can run in several screen resolutions such as standard, stereo, and high resolution. In stereo mode, all tests write into the right and left eyes unless you specify otherwise. This test also supports FFB, FFB2, and FFB2+ fast frame buffer configurations. Use the `ffbconfig -prconf` command to display the configuration of the frame buffer you want to test.

You can interrupt `ffbtest` using Control-C. Turn off all other keyboard input if OPEN LOOK is running on the unit being tested.

Test accuracy is checked using a checksum algorithm. Possible locations of failing pixels are identified, as well as the likely failing FRU.

> ⚠️ **Caution –** DO NOT run any other application or screen saver program that uses the FFB accelerator port while running `ffbtest`. Do not run power management software. These programs causes SunVTS to return incorrect errors.

> **Note –** Disable all screen savers before testing any graphics device. Type **`xset s off`** at a UNIX prompt to disable the Solaris screen saver. Disable power management software if it is running.

> **Note –** DO NOT run Open Windows across multiple monitors.

Note – `ffbtest` requires approximately 7 Mbytes of disk space in the `/tmp` directory to extract its working files. If this space is not available, the diagnostic will fail and report warning and error messages, indicating a lack of disk space.

Note – To start SunVTS with `vtsui`, but without `vtsk`, you must add the host name to `xhost` as: **xhost + <*hostname*>**.

`ffbtest` Options

ffb0(ffbtest) Test Parameter Options

Options:

3DRAM:	Enable Disable
3DRAM Logic:	Enable Disable
RAMDAC:	Enable Disable
Rendering Pipeline:	Enable Disable
Fast Fill/Vertical Scroll:	Enable Disable
Pixel Processor:	Enable Disable
Picking:	Enable Disable
Arbitration:	Enable Disable
Stereo:	Enable Disable

Subtest Repeat: 1

Test Loop Repeat: 1

Pattern: Random

Apply   Reset   Cancel
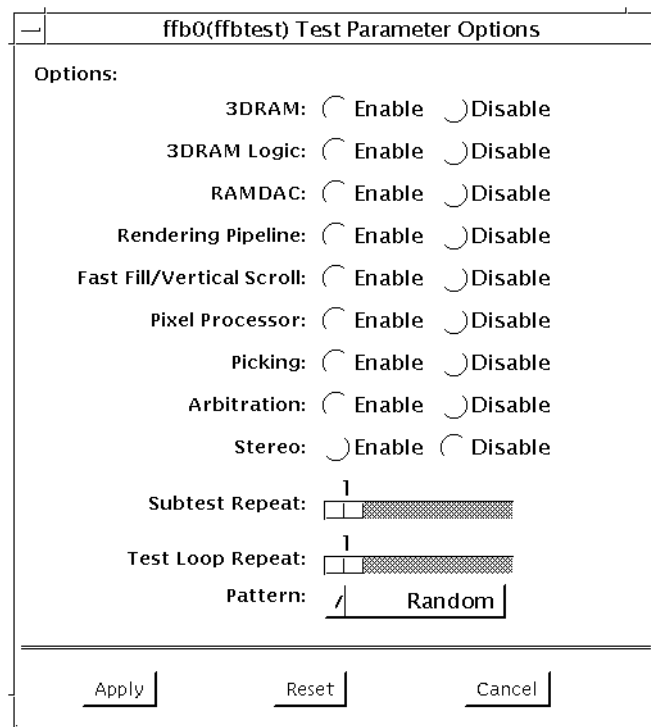
FIGURE 16-1 `ffbtest` Option Menu

By default, all tests are enabled except the stereo test.

TABLE 16-1 `ffbtest` Options

ffbtest **Options**	**Description**
3DRAM Test	The 3DRAM Test thoroughly tests the video memory in the FFB using 512-bit reads and writes. 3DRAM makes a full screen pass, consisting of a write and a read to each pixel location, for each access mode on the list below. You can use either random data or specify data at the command line. A second pass is made with the one's complement of the data used in the first pass so that each memory location is tested with both a zero and a one. Notice that some passes are skipped on the single-buffered FFB. Errors in this subtest are attributes to the 3DRAM. A failing chip is indicated by (x, y) locations and device-specific "U" numbers. DFB8R, DFB8G, DFB8B, DFB8X - Buffer A DFB24 - Buffer A DFB32 - Buffer A SFB8R, SFB8G, SFB8B, SFB8X - Buffer A SFB8R, SFB8G, SFB8B, SFB8X - Buffer B (double buffer only) SFB32 - Buffer A SFB32 - Buffer B (double buffer only) SFB32 - Buffer C (double buffer only) SFB64 - Buffers A and C (double buffer only) SFB64 - Buffers B and C (double buffer only)
3DRAM Logic Test	3DRAM Logic provides logical functionality to the FFB. The following services are tested: Compare Controls - Match AB Compare Controls - Magnitude AB Compare Controls - Match C (double buffer only) Compare Controls - Magnitude C (double buffer only) Match Mask - AB Magnitude Mask - AB Match Mask - C (double buffer only) Magnitude Mask - C (double buffer only) Raster Operations - RGB Raster Operations - X Raster Operations - YZ (double buffer only) Plane Mask - RGB Plane Mask - X Plane Mask - Y Plane Mask - Z Group Enable - R, G, B, X Group Enable - Y, Z (double buffer only)

TABLE 16-1 `ffbtest` Options (Continued)

`ffbtest` **Options**	**Description**
3DRAM Logic Test (Continued)	`ffbtest` tests each function separately with a series of SFB64 writes. A total of 16 writes are made for each different test case with Y coordinate values varying from 0 to 30 in increments of 2 pixels. This dotted column organization provides page thrashing and block flashing in all screen resolutions. For each operation, all possible combinations are tested. For example, `ROP RGB new==old` has three possible values are: `new < old`, `new == old`, *and* `new > old`. `ffbtest` tests each of these cases.

Five passes of the functions are made. Each pass writes into a different FFB address space: SFB32-A, SFB32-B, SFB32-C, SFB64-AC, and SFB64-BC. Note that the passes that write into the SFB32 address spaces are writing two pixels at a time because the tests use SFB64 writes.

For FFB2+ boards, additional testing is performed on the new stencil and passin capabilities if the board is DBZ.

Care is taken to ensure that all 3DRAM chips are tested. Errors in this subtest are attributed to the 3DRAM. |
| RAMDAC Test | RAMDAC registers are tested using simple read/write patterns to determine if there are any bad bits. This includes all LUTs. `ffbtest` ensures that data is actually being read from the RAMDAC and not being supplied by the driver.
Next, the RAMDAC Signature Register captures the pixels going to the screen. This test determines that all of the different data paths within the RAMDAC are functioning properly.

The following modes are tested:

24-bit true color from A
24-bit true linear color from A
24-bit direct color from A
24-bit true color from B (double buffer only)
24-bit true linear color from B
24-bit direct color from B (double buffer only)
8-bit pseudo color (from each plane in RGB) from A
8-bit pseudo color (from each plane in RGB) from B (double buffer only)
8-bit non-linear grayscale (from each plane in RGB) from A
8-bit non-linear grayscale (from each plane in RGB) from B (double buffer only)
8-bit linear grayscale (from each plane in XRGB) from A
8-bit linear grayscale (from each plane in XRGB) from B (double buffer only)
8-bit overlay pseudo color (from buffer A, X plane) |

TABLE 16-1 `ffbtest` Options (Continued)

`ffbtest` **Options**	**Description**
RAMDAC Test (Continued)	This test displays a total of 11 different types of windows on the screen for the single-buffered configuration; 22 for double-buffered. A cursor is also displayed on the screen. RAMDAC on FFB2+ board supports three modes (SEP8, SEP4 and Combined). This test detects the RAMDAC type and tests the original and additional features like increased number of CLUTs, increased WLUT size, additional overlay WLUT. Errors in this test are attributed to the RAMDAC.
Rendering Pipeline Test	Rendering Pipeline uses the rendering pipeline tests developed for the FFB stand-alone diagnostics. Each primitive is tested thoroughly with the following sources and configurations: • Dots • Anti-aliased dots • Lines using all four line drawing primitives • Triangles • Polygons • Rectangles • Fonts Errors in this test are attributed to the FBC.
Fast Fill/Vertical Scroll Test	The Fast Fill/Vertical Scroll primitives are separated from the Rendering Pipeline tests because of their dependence on screen type. There are three different tests, one for each screen type. Each test uses both block and page mode fistfuls. Errors in this test are attributed to the FBC.
Pixel Process Test	Pixel Processor, a subtest, exercises the following options selected by the FFB's Pixel Processor Control (PPC) register. • Auxiliary clipping (additive and subtractive) • Depth cueing • Alpha blend • Viewport clip (2D and 3D) • Area pattern (transparent and opaque) Errors in this test are attributed to the FBC.

TABLE 16-1 `ffbtest` Options (Continued)

`ffbtest` **Options**	**Description**
Picking Test	The Picking test exercises the pick detect login of the 3DRAM. `ffbtest` defines a pick detect window and checks that writes to the window are picked, and writes outside the window are not picked. The test is repeated once for each 3DRAM. Errors in this test are attributed to the 3DRAM.
Arbitration Test	Arbitration, a subtest, continuously renders an object into the accelerator port while performing reads and writes through the direct port. For single-buffered configuration, a picture is rendered into the RGB planes while another process does DFB reads and writes in the X plane. For doubled buffered configuration, a picture is rendered into all 32 planes of the B buffer while the other does 32-bit DFB reads and writes in the A plane. This subtest simulates conditions in the read world, where rendering processes and windows operations run concurrently. Errors in this test are attributed to the FBC.

TABLE 16-1   `ffbtest` Options (Continued)

`ffbtest` **Options**	**Description**
Stereo Test	The Stereo test displays an object in stereo mode with different images for the right and left eye. The user can verify proper operation by looking at the screen with stereo glasses and following the displayed instructions. If the monitor type is not 1280x1024 @ 76MHz, this test prints a warning message and does not run. To prevent this message from being displayed or written to the SunVTS information log, disable the stereo test in the test option menu. Only Sony P4 and N2 monitors support stereo resolutions. This test temporarily switches the monitor into stereo mode, renders a stereo image, performs a signature analysis on the stereo image (using the RAMDAC signature capture register), and after displaying the image for five seconds, restores the monitor to its previous resolution.
	Errors in this test are attributed to the RAMDAC.
	NOTE — If vertical lines are displayed on the console when running SunVTS, this could be caused by the `ffbtest` stereo test. There is a time-critical period in the FFB hardware when trying to change the screen resolution from standard to stereo and back to standard. When the system is heavily loaded or running all of the SunVTS tests, the FFB device driver may get interrupted while changing screen resolution. If this occurs, FB ASIC and RAMDAC get out of synchronization, resulting in an unusual display on the FFB screen. This problem could also cause a system hang condition.
	This test is disabled by default because it is only needed when a stereo monitor and stereo glasses are present.
	To avoid this type of display problem, disable the `ffbtest` stereo test when other SunVTS tests are enabled.

`ffbtest` Test Modes

Due to the nature of graphic tests, reading from or writing to the frame buffer during graphic tests will disturb user operation. This test is only available in Functional Test and Stress Mode.

TABLE 16-2 `ffbtest` Test Modes

Test Mode	Description
Functional Test	The `ffbtest` verifies both the single- (SFB) and double-buffered (DBZ) fast frame buffer boards.
Stress Mode	Stress mode exercises the frame buffer as much as possible. The Random Test generator, constructed as part of the verification effort, is used. Starting from a known seed, random primitives with random attributes are generated. The primitives are checked to ensure that they were rendered in the same way as on a known good system. The test is repeated ten times, with each random picture overlaying the previous one. Stress is mode is not available on FFB2+ boards.

`ffbtest` Command Line Syntax

/opt/SUNWvts/bin/ffbtest *standard_arguments* **-o dev=***device_name,* **S=***subtest_number,* **F=***#_of_subtest_loops,* **B=***#_of_test_loops,* **P=***test_pattern*

TABLE 16-3 `ffbtest` Command Line Syntax

Argument	Explanation
dev=*device_name*	*device_name* is the relative path name of the device being tested with respect to `/dev/fbs`; The default is `ffb0`.

TABLE 16-3   `ffbtest` Command Line Syntax

Argument	Explanation (Continued)
S=*subtest_number*	*subtest_number* is the test number of the subtest to be run. Select from the subtests below. You can run multiple subtests by adding the subtest numbers. For example, `n=0x3` runs both test 1 and test 2; `n=0x180` runs both test 0x080 and test 0x0100. Note that you do not need the leading zeros. **n - 0x00001 3DRAM** **n - 0x00002 3DRAM Logic** **n - 0x00004 RAMDAC** **n - 0x00008 Rendering Pipeline** **n - 0x00010 FastFill/Vertical Scroll** **n - 0x00020 Pixel Processor** **n - 0x00040 Picking** **n - 0x00080 Arbitration** **n - 0x00100 Stereo** More than one test can be selected by ORing subtest numbers. For example: n = 0x00009 selects 3DRAM and Rendering Pipeline tests. A hex number must be preceded by `0x`, decimal numbers are also acceptable.
F=*#_of_subtest_loops*	Specifies the number of times to repeat each subtest; the default is 1.
B=*#_of_test_loops*	Specifies the number of times to repeat a test loop before passing; the default is 1.
P=*test_pattern*	Specifies the test pattern number. The default is **r**, for random patterns. You may also choose **0 for 0x0000000**, **3 for 0x3333333**, **5 for 0x5555555**, or **9 for 0x9999999**.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`ffbtest` Error Messages

Errors returned by `ffbtest` are nonspecific. It is not possible to determine which component caused a failure. In all error conditions, the field replaceable unit (FRU) is the entire FFB.

TABLE 16-4 `ffbtest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	`3DRAM Memory error in` *string* `Plane`	Faulty 3DRAM, or possibly host memory error during gcopy operation.	If the problem persists, contact your authorized Sun service provider.
6002	`3DRAM Memory error in` *string* `Plane`	Faulty 3DRAM, or possibly host memory error during gcopy operation.	If the problem persists, contact your authorized Sun service provider.
6004	`3DRAM Memory error in` *string* `Plane`	Faulty 3DRAM, or possibly host memory error during gcopy operation.	If the problem persists, contact your authorized Sun service provider.
6006	`3DRAM Memory error in Buffer A`	Faulty 3DRAM, or possibly host memory error during gcopy operation.	If the problem persists, contact your authorized Sun service provider.
6008	`3DRAM Memory error in Buffer B`	Faulty 3DRAM, or possibly host memory error during gcopy operation.	If the problem persists, contact your authorized Sun service provider.
6010	`3DRAM Memory error in Buffer C`	3DRAM Memory error in Buffer C.	If the problem persists, contact your authorized Sun service provider.
6012	`Misaligned raster.`	Software error.	Reinstall software. If the problem persists, contact your authorized Sun service provider.
6013	`File Not Found. Could not open file` *string*.	Software error	Reinstall software. If the problem persists, contact your authorized Sun service provider.
6014	`Could not open file` *string*`:` *string*.	Software error.	Reinstall software. If the problem persists, contact your authorized Sun service provider.

TABLE 16-4 `ffbtest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6015	Keyword not Found, line *data*	Software error, Suspect data file corruption.	Reinstall software. If the problem persists, contact your authorized Sun service provider.
6016	*string* error. Miscompares, starting at *string*.	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6017	Picking test error. Got a pick hit with picking disabled.	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6018	Picking test error. Got a pick hit outside the picking area.	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6019	Picking test error. Should have detected a pick hit.	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6020	Out of memory	Software System Error.	Contact system administrator
6021	*string* failed	Software error.	Check for existence and/or permissions on *string*. Also check for available space in tmp. Reinstall software or contact your system administrator.
6022	*string* failed	Software error.	Check for existence and/or permissions on *string*.
6023	fork1(2) failed	Software error.	Reinstall software or contact your system administrator.
6024	execl(2) failed	Software error.	Reinstall software or contact your system administrator.
6025	Unable to open *string*.	Software error.	Check device for existence and/or permissions on *string*. Reinstall software or contact your system administrator.
6026	ioctl(2) failed trying to read CLUT.	hardware failure or software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.

TABLE 16-4 `ffbtest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6027	`ioctl(2) failed trying to turn on FFB video.`	hardware failure or software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.
6028	`failed trying to post CLUT.`	hardware failure or software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.
6029	`Miscompare found in Red channel of CLUT entry` *data*.	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6030	`Miscompare found in Green channel of CLUT entry` *data*	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6031	`Miscompare found in Blue channel of CLUT entry` *data*	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6032	`ioctl(2) failed trying to read WID LUT.`	hardware failure or software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.
6033	`ioctl(2) failed trying to post WID LUT`	hardware failure or software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.
6034	`failed trying to get RAMDAC signature analysis.`	hardware failure or software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.
6035	`Miscompare found in WID LUT entry` *data*	Faulty frame buffer	If the problem persists, contact your authorized Sun service provider.
6036	`RAMDAC test failed. The on-screen image is incorrect`	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6037	`sysinfo(2) failed getting hostname`	Software error.	Make sure you are superuser.

TABLE 16-4 ffbtest Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6038	can't acquire console semaphore	Software error.	Reinstall software or contact your system administrator.
6039	Cannot gain control of mouse: *string*	Software error.	Reinstall software or contact your system administrator.
6040	Cannot gain control of keyboard: *string*	Software error	Reinstall software or contact your system administrator.
6046	ioctl(2) failed trying to get current video mode.	Possibly hardware failure, likely software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.
6047	ioctl(2) failed trying to set current video mode	Possibly hardware failure, likely software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.
6048	ioctl(2) failed trying to get FFB system info	Possibly hardware failure, likely software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.
6049	ioctl(2) failed trying to go into Diagnostic mode	Possibly hardware failure, likely software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.
6050	mmap(2) failed: *string*	Software system error.	Reinstall software or contact your system administrator.
6051	munmap(2) failed: *string*	Software system error.	Reinstall software or contact your system administrator.
6052	close(2) failed: *string*	Software system error.	Reinstall software or contact your system administrator.
6053	DGA error grabbing window on screen *data*	Software error.	Reinstall software or contact your system administrator.
6054	DGA error getting window from token on screen *data*	Software error.	Reinstall software or contact your system administrator.
6055	failed trying to read Transparent Overlay registers	Possibly hardware failure, likely software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.

TABLE 16-4 `ffbtest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6056	`ioctl(2) failed trying to post Transparent Overlay registers`	Possibly hardware failure, likely software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.
6057	`failed trying to post cursor`	Possibly hardware failure, likely software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.
6058	`ioctl(2) failed trying to read cursor`	Possibly hardware failure, likely software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.
6059	`No FFB device found on display 0`	The environmental variable DISPLAY is set wrong or the display being used is not using an FFB.	Set the variable to a display that has the FFB plugged into it.
6060	`There is an inconsistency with the version of the X Window System`	Software error.	Contact your system administrator
6061	`stat(2) failed trying to get info about Device Under Test`	Software error.	Reinstall software or contact your system administrator.
6062	`stat(2) failed trying to get info about screen 0:`*data*	Software error.	Reinstall software or contact your system administrator.
6063	`Can not access X Windows`	Software error.	Try running `xhost(1)` or contact your system administrator.
6064	`wait(2) failed:` *string*	Software error.	Reinstall software or contact your system administrator.
6065	`signal(2) failed:` *string*	Reinstall software or contact your system administrator.	Reinstall software or contact your system administrator.
6066	`thr_kill(2) failed`	Software error.	Reinstall software or contact your system administrator.
6067	`thr_join(2) failed:` *string*	Software error.	Reinstall software or contact your system administrator.

TABLE 16-4 ffbtest Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6071	get signatures	operator error.	To get signatures, run ffbtest from the command line
6072	Stereo image incorrect	Stereo image incorrect.	If the problem persists, contact your authorized Sun service provider.
6075	System LWP limit exceeded	Software error.	Reinstall software or contact your system administrator.
6076	Error *data* creating thread	Software error.	Reinstall software or contact your system administrator.
6104	Error in getting the length of the file. *string*	Software error.	If the problem persists, contact your authorized Sun service provider.
6105	Stencil test failed *string*	Faulty 3DRAM.	If the problem persists, contact your authorized Sun service provider.
6106	Passin test failed *string*	Faulty 3DRAM.	If the problem persists, contact your authorized Sun service provider.
6107	Blend2 test failed *string*	Faulty 3DRAM.	If the problem persists, contact your authorized Sun service provider.
6115	3DRAM Memory error testing *string*	Faulty 3DRAM, or possibly GCopy error.	If the problem persists, contact your authorized Sun service provider.
6116	*string* error: 3DRAM Memory error testing *string*	Faulty 3DRAM, or possibly GCopy FFB to Memory error.	If the problem persists, contact your authorized Sun service provider.
6117	*string* error: 3DRAM Memory error testing *string*	Faulty 3DRAM, or possibly GCopy Memory to FFB error.	If the problem persists, contact your authorized Sun service provider.
6208	ioctl(2) failed trying to post Overlay WID LUT.	Possibly hardware failure, likely software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.
6209	ioctl(2) failed trying to post Primary WID LUT	Possibly hardware failure, likely software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.

TABLE 16-4 `ffbtest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6210	Miscompare found in Overlay WID LUT entry *data*	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6211	Miscompare found in Primary WID LUT entry *data*	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6212	ioctl(2) failed trying to read CLUT # *data*	Possibly hardware failure, likely software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.
6213	ioctl(2) failed trying to post CLUT # *data*	Possibly hardware failure, likely software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.
6214	Miscompare found in Red channel of CLUT # *data* in entry *data*	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6215	Miscompare found in Green channel of CLUT # *data* in entry *data*	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6216	Miscompare found in Blue channel of CLUT # *data* in entry *data*	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6217	RAMDAC test failed. The on-screen image in sep 4 mode is incorrect.	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6218	RAMDAC test failed. The on-screen image in sep 8 mode is incorrect.	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6219	RAMDAC test failed. The on-screen image in combined mode is incorrect.	Faulty frame buffer.	If the problem persists, contact your authorized Sun service provider.
6220	ioctl(2) failed trying to read Overlay WID LUT	Possibly hardware failure, likely software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.

TABLE 16-4 `ffbtest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6221	`ioctl(2) failed trying to read Primary WID LUT`	Possibly hardware failure, likely software driver error.	Reinstall software first, if failure continues, see a system administrator or your authorized Sun service provider.

Floating Point Unit Test (`fputest`)

`fputest` checks the floating point unit on machines with the SPARC-based architecture. `fputest` performs the following subtests:

Instruction tests:

- FSR Register test
- Registers test
- NACK test
- Move Registers test
- Positive to Negative test
- Negative to Positive test
- Absolute test
- Single-Precision Integer to Floating Point test
- Double-Precision Integer to Floating Point test
- Single-Precision Floating Point to Integer test
- Double-Precision Floating Point to Integer test
- Single-Precision Round Toward Zero test
- Double-Precision Round Toward Zero test
- Single to Double-Precision Format Conversion test
- Double to Single-Precision Format Conversion test
- Single and Double-Precision Addition, Subtraction, Multiplication, Square-root, Division, and Compare tests
- Single and Double-Precision Compare and Exception if Unordered tests
- Branching and no Branching on Condition Instructions tests
- Single and Double-Precision Chaining tests
- Weitek Status tests
- Lock test
- Single and Double-Precision Datapath tests
- Timing (load) test

Benchmark tests:

- Linpack test
- Cparanoia test
- Cora test

- Kcsqrt test
- Kcdiv test
- Clorenz test
- Cvector test

`fputest` Options
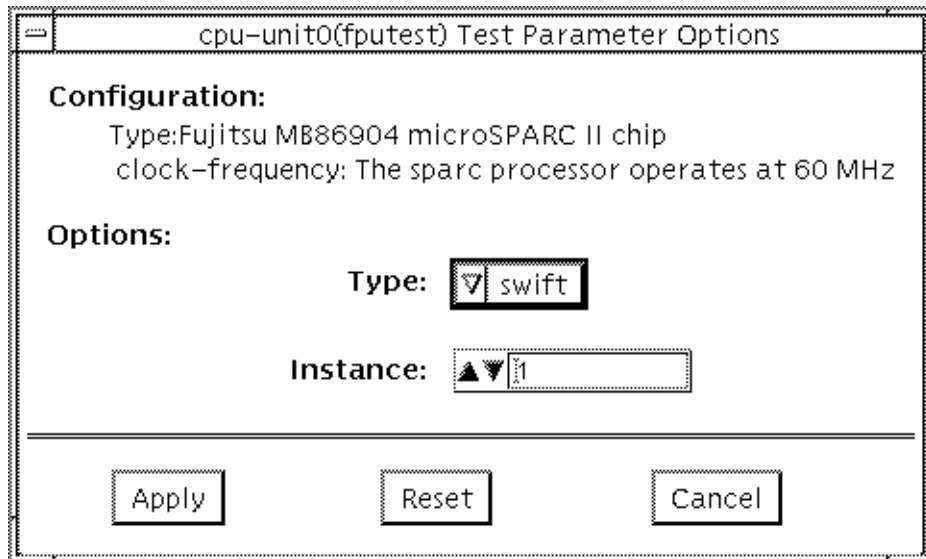
FIGURE 17-1 `fputest` Option Menu

`fputest` Test Modes

TABLE 17-1   `fputest` Test Modes

Test Mode	Description
Connection Test Mode	In this mode, `fputest` includes all the instruction tests.
Functional Test Mode	Functional test includes all the Instruction Tests and all the benchmark tests.
Functional Test mode through SyMON	If you invoke SunVTS through SyMON, the following Functional Test description applies: `fputest` runs Linpack test and all the instruction tests.
Stress Mode	When stress mode is selected, several `fpu` benchmark tests are run.

`fputest` Command Line Syntax

/opt/SUNWvts/bin/fputest *standard_arguments*

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`fputest` Error Messages

TABLE 17-2   `fputest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6012	`Precision worse than 5 decimal figures`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6013	Radix is too big: roundoff problems	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6014	Radix is not as good as 2 or 10	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6015	(1-U1)-1/2 < 1/2 is FALSE, prog. fails?	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6016	X=1 but X-1/2-1/2!= 0	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6017	Subtraction is not normalized X=Y,X+Z != Y+Z!	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6018	Multiplication lacks a Guard Digit, so 1*X != X	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6019	Multiplication gets too many final digits wrong	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6020	Division lacks a Guard Digit	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6021	Division lacks a Guard Digit, so X/1 != X	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6022	Computed value of 1/1.000.1 >= 1	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6023	Multiplication and/or Division gets too many last digits wrong	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6024	Subtraction lacks Guard Digit, so cancellation is obscured	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6025	X * (1/X) differs from 1	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6026	Radix * (1 / Radix) differs from 1	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6027	Incomplete carry-propagation in Addition	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6028	Square root of 0.0, -0.0 or 1.0 wrong	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6029	sqrt gets too many last digits wrong	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6030	Square root is neither chopped nor correctly rounded	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6031	PseudoZero prints out as: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6032	PseudoZero prints out as: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6033	Underflow prints out as: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6034	Accuracy deteriorates as numbers approach a threshold = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6035	Underflow confuses Comparison	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6036	X = *value* is not equal to Z = *value* yet X - Z yields *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6037	X = *value* is not equal to Z = *value* yet X - Z yields *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6038	Calculation yields: *value* :This is not between 0 and underflow threshold = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6039	Calculation yields: *value* :This is not between 0 and underflow threshold = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6040	Calculated *value* for (1 + (*value*) ^ (*value*); differs from correct value by *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6041	Calculated *value* for (1 + (*value*) ^ (*value*); differs from correct value by *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6042	Overflow(`Z = -Y') test on Y = *value* finds that -(-Y) differs from Y	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6043	Overflow(`Z = -Y') test on Y = *value* finds that -(-Y) differs from Y	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6044	Overflow past *value* shrinks to *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6045	Overflow past *value* shrinks to *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6046	Comparisons involving +- *value*, +-*value* and +-*value* are confused by Overflow	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6047	Comparisons involving +- *value*, +-*value* and +-*value* are confused by Overflow	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6048	A total of *value* floating point exceptions were registered	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6049	(*value*) ^ (*value*) yielded *value*; which compared unequal to correct *value*; they differ by *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6050	(*value*) ^ (*value*) yielded *value*; which compared unequal to correct *value*; they differ by *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6051	What prints as Z = *value* compares different from Z * 1 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6052	What prints as Z = *value* compares different from 1 * Z == *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6053	What prints as Z = *value* compares different from Z / 1 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6054	What prints as Z = *value* compares different from Z * 1 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6055	What prints as Z = *value* compares different from 1 * Z == *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6056	What prints as Z = *value* compares different from Z / 1 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6057	Multiplication does not commute:Comparison alleges that 1 * Z = *value* differs from Z * 1 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6058	Multiplication does not commute:Comparison alleges that 1 * Z = *value* differs from Z * 1 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6059	*subtest_name* test appears to be inconsistent.	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6060	Comparison alleges that - 0.0 is Non-zero!	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6061	Disagreements among the values X1, Y1, Z1 respectively *value value value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6062	Disagreements among the values X1, Y1, Z1 respectively *value*, *value*, *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6063	Comparison alleges (1-U1) < 1 although subtraction yields (1-U1) - 1 = 0	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6064	Multiplication appears to chop	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6065	Multiplication is neither chopped nor correctly rounded	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6066	Division appears to chop	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6067	Division is neither chopped nor correctly rounded	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6068	Addition/Subtraction appears to be chopped	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6069	Addition/Subtraction neither rounds nor chops	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6070	(X - Y) + (Y - X) is non zero	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6071	Sticky bit used incorrectly or not at all	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6072	X * Y == Y * X trial fails	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6073	Sqrt(X) is non-monotonic for X near *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2   `fputest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6074	Sqrt(X) is non-monotonic for X near *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6075	Anomalous arithmetic with Integer Radix-Precision = *value* fails test whether sqrt rounds or chops	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6076	Anomalous arithmetic with Integer Radix-Precision = *value* fails test whether sqrt rounds or chops	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6077	Square root appears to be chopped	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6078	Comparison alleges that Z = *value* is too far from sqrt(Z) ^ 2 (*value*)	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6079	Comparison alleges that Z = *value* is too far from sqrt(Z) ^ 2 (*value*)	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6080	Unbalanced range; UfThold * V = *value* is too far from 1	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6081	Unbalanced range; UfThold * V = *value* is too far from 1	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6082	Sqrt(*value*) - *value* = *value* instead of correct value 0	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6083	Sqrt(*value*) - *value* = *value* instead of correct value 0	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6084	Similar discrepancies have occurred *value* times	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6085	X / X traps when X = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6086	X / X traps when X = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6087	X / X differs from 1 when X = *value* instead, X / X - 1/2 - 1/2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6088	X / X differs from 1 when X = *value* instead, X / X - 1/2 - 1/2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6089	`Computed:`*character* `Expected:`*character*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6090	`Failed Vector test`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6091	`Error: Invalid operator` (*character*)	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6092	`Error: Invalid version` (*character*)	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6093	`Error: Invalid precision` (*character*)	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6094	`Error: Illegal input` (*character*)	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6095	`Failed Vector` *subtest_name* `sub-test[`*text*`],` `Operand 1` *text value value,* `Operand 2` *text value value,* *charactercharactercharacter* `Correct` *text value value* (*text*), `Computed` *text value value* (*text*)	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6096	`Failed Vector` *subtest_name* `sub-test[`*text*`],` `Operand 1` *text value value,* `Operand 2` *text value value value,* *charactercharactercharacter* `Correct` *text value value value* (*text*), `Computed` *text value value value* (*text*)	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6097	`Failed Vector` *subtest_name* `sub-test[`*text*`],` `Operand 1` *text value value value value,* `Operand 2` *text value value value value,* *charactercharactercharacter* `Correct` *text value value value value* (*text*), `Computed` *text value value value value* (*text*)	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6098	`Failed Vector` *subtest_name* `sub-test[`*text*`]` `Operand 1` *text* `value` *value* `Operand 2` *text* `value value` *value* *charactercharactercharacter* `Correct` *text* *value value value* (*text*) `Computed` *text value value value* (*text*)	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6099	*charactercharactercharacter value* (*text*) *character value* (*text*) `<computed` *character*`>`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6100	*charactercharactercharacter* `Op1` (*text*) *character* `Op2` (*text*) `<computed` *character*`>`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6101	`Flags: Correct` (*text*) `Computed` (*text*)	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6102	`FPU initialization failure`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6103	`Multiplication failed, result was` (*value*)`, expected` (*value*)	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6104	`Failed Single-Precision FPA math test`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6105	`Failed Double-Precision FPA math test`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6106	`Failed linpack test`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6107	`Failed cparanoia test`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6108	`Failed cora test`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6109	`Failed clorenz test`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6110	`Failed kcsqrt test`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6111	`Failed kcdiv test`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6112	Error: a + b, Expected: 2.2221000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6113	Error a - b,, Expected: 0.2469000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6114	Error a * b, Expected: 1.2191922 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6115	Error a / b, Expected: 1.2500000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6116	Error: a + (a + b), Expected: 1.4814000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6117	Error: a - (a + b), Expected: -0.9876000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6118	Error: a + (a * b), Expected: 2.4536924 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6119	Error: a - (a * b), Expected: 0.0153078 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6120	a + (a / b), Expected: 2.4845002 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6121	Error: a - (a / b), Expected: -0.0155000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6122	Error: a * (a + b), Expected: 2.7431825 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6123	Error: a * (a - b), Expected: 0.3047980 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6124	Error: a / (a - b), Expected: 0.5555550 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6125	Error: a / (a - b), Expected: 5.0000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action

6126	`Error: a * ( a / b),`
	`Expected: 1.5431250`
	`Actual:` *value*)
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6127	`Error: a / ( a * b),`
	`Expected: 1.0125557`
	`Actual:` *value*)
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6128	`Error: a + b, Expected:`
	`2.222100000000000`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6129	`Error: a – b, Expected:`
	`0.246899999999999`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6130	`Error: a * b, Expected:`
	`1.219192199999999`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6131	`Error: a / b, Expected:`
	`1.249999999999999`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6132	`Error: a + (a – b),`
	`Expected:1.481399999999999`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6133	`Error: a – (a + b),`
	`Expected: –`
	`0.987600000000000`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6134	`Error: a + (a * b),`
	`Expected:2.453692200000000`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6135	`Error: a – (a * b),`
	`Expected:0.015307800000000`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6136	`Error: a + (a / b),`
	`Expected:2.484500000000000`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6137	`Error: a – (a / b),`
	`Expected: –`
	`0.015499999999999`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6138	`Error: a * (a + b),`
	`Expected:2.743182449999999`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action

6139	`Error: a * (a - b),`
	`Expected:0.304798049999999`
	`Actual:` *value*
		FFaulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6140	`Error: a / (a + b),`
	`Expected:0.555555555555555`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6141	`Error: a / (a - b),`
	`Expected:5.000000000000002`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6142	`Error: a * (a / b),`
	`Expected:1.543124999999999`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6143	`Error: a / (a * b),`
	`Expected:`
	`1.0125555690562980`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6144	`Error: sin(-2pi), Expected:`
	`-0.000000000820413`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6145	`Error: sin(-3pi 2),`
	`Expected:`
	`1.0000000000000000`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6146	`Error: sin(-3pi/2),`
	`Expected:`
	`1.0000000000000000`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6147	`Error: sin(-pi), Expected:`
	`0.000000000410206`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6148	`Error: sin(-pi/2),`
	`Expected:`
	`-1.0000000000000000`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6149	`Error: sin(0), Expected:`
	`0.0000000000000000`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6150	`Error: sin(pi/2), Expected:`
	`1.0000000000000000`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6151	Error: sin(pi), Expected: – 0.000000000410206 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6152	Error: sin(3pi/2), Expected: – 1.0000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6153	Error: sin(2pi), Expected: 0.000000000820143 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6154	Error: cos(-2pi), Expected: 1.0000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6155	Error: cos(-3pi/2), Expected: 0.000000000615310 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6156	Error: cos(-pi), Expected: -1.0000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6157	Error: cos(-pi/2), Expected: – 0.000000000205103 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6158	Error: cos(0), Expected: 1.0000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6159	Error: cos(pi/2), Expected: -0.000000000205103 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6160	Error: cos(pi), Expected: – 1.0000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6161	Error: cos(3pi/2), Expected: 0.000000000615310 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6162	Error: cos(pi/2), Expected: 1.0000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6163	Error: sin(pi/4), Expected: 0.707106781259062 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action

6164	`Error: sin(3pi/4),`
	`Expected: 0.707106780969002`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6165	`Error: sin(5pi/4),`
	`Expected: –`
	`0.707106781549122 Actual:`
	value
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6166	`Error: sin(7pi/4),`
	`Expected: –`
	`0.707106780678942 Actual:`
	value
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6167	`Error: cos(pi/4), Expected:`
	`0.707106781114032 Actual:`
	value
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6168	`Error: cos(3pi/4),`
	`Expected: –`
	`0.707106781404092 Actual:`
	value
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6169	`Error: cos(5pi/4),`
	`Expected: –`
	`0.707106780823972 Actual:`
	value
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6170	`Error: cos(7pi/4),`
	`Expected: 0.707106781694152`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6171	`Error: exp(0), Expected:`
	`1.0000000000000000 Actual:`
	value
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6172	`Error: exp(1), Expected:`
	`2.718281828459045 Actual:`
	value
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6173	`Error: exp(2), Expected:`
	`7.389056098930650 Actual:`
	value
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6174	`Error: exp(5), Expected:`
	`148.413159102576600 Actual:`
	value
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6175	`Error: exp(10), Expected:`
	`22026.465794806718000`
	`Actual:` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action

6176	`Error: exp(-1), Expected:`
	`0.367879441171442`
	`Actual: value`
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6177	`Error: exp(-2), Expected:`
	`0.135335283236612`
	`Actual: value`
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6178	`Error: exp(-5), Expected:`
	`0.006737946999085`
	`Actual: value`
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6179	`Error: exp(-10), Expected:`
	`0.000045399929762`
	`Actual: value`
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6180	`Error: exp(log(1),`
	`Expected:`
	`1.0000000000000000`
	`Actual: value`
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6181	`Error: exp(log(10),`
	`Expected 10.000000000000002`
	`Actual: value`
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6182	`Error: log(1), Expected:`
	`0.0000000000000000`
	`Actual: value`
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6183	`Error: log(2), Expected:`
	`0.693147180559945`
	`Actual: value`
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6184	`Error: log(10), Expected:`
	`2.302585092994045`
	`Actual: value`
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6185	`Error: log(100), Expected:`
	`4.605170185988091`
	`Actual: value`
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6186	`Error: log(exp(0)),`
	`Expected:`
	`0.0000000000000000`
	`Actual: value`
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6187	`Error: log(exp(1)),`
	`Expected:`
	`1.0000000000000000`
	`Actual: value`
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6188	Error: log(exp(10)), Expected: 10.0000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6189	Error: tan(-2pi), Expected: -0.000000000820414 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6190	Error: tan(-7pi/4), Expected: 0.999999998564275 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6191	Error: tan(-5pi/4), Expected: - 1.000000001025517 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6192	Error: tan(-pi, Expected: 0.000000000410207 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6193	Error: tan(-3pi/4), Expected: 0.999999999384690 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6194	Error: tan(-pi/4), Expected: - 1.000000000205103 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6195	Error: tan(0.0), Expected: 0.000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6196	Error: tan(pi / 4), Expected: 1.000000000205103 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6197	Error: tan(3pi/4), Expected: - 0.999999999384690 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6198	Error: tan(pi), Expected: 0.000000000410207 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6199	Error: tan(5pi/4), Expected: 1.000000001025517 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6200	Error: tan(7pi/4), Expected: – 0.999999998564275 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6201	Error: tan(2pi), Expected: 0.000000000820414 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6202	Error: sqrt(0), Expected: 0.000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6203	Error: sqrt(1), Expected: 1.000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6204	Error: sqrt(4), Expected: 2.000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6205	Error: sqrt(9), Expected: 3.000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6206	Error: sqrt(16), Expected: 4.000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6207	Error: sqrt(25), Expected: 5.000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6208	Error: sqrt(36), Expected: 6.000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6209	Error: sqrt(49), Expected: 7.000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6210	Error: sqrt(64), Expected: 8.000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6211	Error: sqrt(81), Expected: 9.000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6212	Error: sqrt(100), Expected: 10.000000000000000 Actual: *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action

6213	`log-test failed: int = `*value*`,`
	`expected / observed = `*value*` / `
	value
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6214	`kvm_read(mem,nl[0].n_value,` `..) failed`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6215	`kvm_open()` `failed:<error_message>`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6216	`kvm_nlist()` `failed:<error_message>`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6217	`Add SP failed: expected /` `read = `*value*` / `*value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6218	`Multiply SP failed:expected` `/ read = `*value*` / `*value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6219	`Add DP failed: msw :`
	`expected / read = `*value*` /`
	value`,                 lsw`
	`: expected / read = `*value*` /`
	value
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6220	`Multiply DP failed: msw :`
	`expected / read = `*value*` /`
	value`,  lsw : expected /`
	`read = `*value*` / `*value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6221	`Single-Precision: add,`
	`expected / observed =`
	`0x41200000 / 0x`*value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6222	`Single-Precision: Multiply,`
	`expected / observed =`
	`0x43470000 / 0x`*value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6223	`Double-Precision: Add, MSW`
	`: expcected / observed =`
	`0x40240000 / 0x`*value*`,`
	`LSW : expected / observed =`
	`0x0  / 0x`*value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6224	`Double-Precision: Multiply,`
	`MSW : expected / observed =`
	`0x4034000 / 0x`*value*`,`
	`LSW : expected / observed =`
	`0x0  / 0x`*value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6225	`Error: expected / observed` `= `*value*` / 0x`*value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6226	Error: expected / observed = *value* / 0x*value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6227	fitos failed: int = *value*, expected / observed = *value* / *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6228	fitod failed: int = *value*, expected / observed = *value* / *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6229	fstoi failed: int = *value*, expected / observed = *value* / *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6230	fdtoi failed: int = *value*, expected / observed = *value* / *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6231	fstod failed: int = *value*, expected / observed = *value* / *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6232	fdtos failed: int = *value*, expected / observed = *value* / *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6233	fmovs failed : written *value* to f0, read from f31 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6234	fnegs failed(from pos to neg): int = *value*, expected / observed = *value* / *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6235	fnegs failed (from neg. to pos): int = *value*, expected / observed = *value* / *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6236	fabs failed: int = *value*, expected / observed = *value* / *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6237	fadds failed: int = *value*, f0 = *value*, f2 = *value*, f0+f2 = f4 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6238	faddd failed: int = *value*, f0 = *value*, f2 = *value*, f0+f2 = f4 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6239	`fsubs failed:int =` *value*`, f0 =` *value*`, f2 =` *value*`, f0-f2 = f4 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6240	`fsubd failed: int =` *value*`, f0 =` *value*`, f2 =` *value*`, f0-f2 = f4 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6241	`fsqrt(single) failed: written / read =` *value* `/` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6242	`fsqrt(double) failed: written / read =` *value* `/` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6243	`fdivs failed: int =` *value*`, f0 =` *value*`, f2 =` *value*`, f0 / f2 = f4 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6244	`fdivd failed: int =` *value*`, f0 =` *value*`, f2 =` *value*`, f0 / f2 = f4 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6245	`fmuls failed: int =` *value*`, f0 =` *value*`, f2 =` *value*`, f0 / f2 = f4 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6246	`fmuld failed: int =` *value*`, f0 =` *value*`, f2 =` *value*`, f0 / f2 = f4 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6247	`fcmps failed: f0 =` *value*`, f2 =` *value* `: expected / observed = 0 /` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6248	`fcmps failed: f0 =` *value*`, f2 =` *value* `: expected / observed = 1 /`*value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6249	`fcmps failed: f0 =` *value*`, f2 =` *value* `: expected / observed = 2 /`*value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6250	`fcmps failed: f0 =` *value*`, f2 = NaN : expected / observed = 3 /`*value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6251	`fcmpd failed: f0 =` *value*`, f2 =` *value* `: expected / observed = 0 /` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6252	`fcmpd failed: f0 =` *value*`, f2 =` *value* `: expected / observed = 1 /`*value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6253	`fcmpd failed: f0 =` *value*`, f2 =` *value* `: expected / observed =  2  /`*value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6254	`fcmpd failed: f0 =` *value*`, f2 = NaN : expected / observed =  3  /`*value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6255	`FBU failed. result =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6256	`FBG failed: f0 =` *value*`, f2 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6257	`FBUG (unordered) failed`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6258	`FBUG (greater) failed: f0 =` *value*`, f2 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6259	`FBL failed: f0 =` *value*`, f2 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6260	`FBUL (unordered) failed`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6261	`FBUL (Less) failed: f0 =` *value*`, f2 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6262	`FBLG (Less) failed: f0 =` *value*`, f2 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6263	`FBLG (Greater) failed: f0 =` *value*`, f2 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6264	`FBNE failed: f0 =` *value*`, f2 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6265	`FBE failed : f0 =` *value*`, f2 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6266	`FBUE (unordered) failed`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6267	`FBUE (equal) failed : f0 =` *value*`, f2 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6268	`FBGE (equal) failed : f0 =` *value*`, f2 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6269	`FBGE (greater) failed: f0 =` *value*`, f2 =` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6270	`FBUGE (unordered) failed`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6271	FBUGE (equal) failed : f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6272	FBUGE (greater) failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6273	FBLE (Less) failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6274	FBLE (equal) failed : f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6275	FBULE (unordered) failed	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6276	FBULE (Less) failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6277	FBULE (equal) failed : f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6278	FBO failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6279	FBA failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6280	FBN failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6281	FBU failed	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6282	FBG failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6283	FBUG failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6284	FBLfailed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6285	FBUL failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6286	FBLG failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6287	FBNE failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6288	FBE failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6289	FBUE failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6290	FBGE failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6291	FBUGE failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6292	FBLE failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6293	FBULE failed: f0 = *value*, f2 = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6294	FBO failed	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6295	fcmpxs failed: Exception did not occur. fsr = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6296	fcmpxd failed: Exception did not occur. fsr = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6297	Error: Bus error occured. ftt = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6298	Error: Bus error did not occur(IEEE exception). ftt = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6299	Error: Bus error did not occur(Unfinished exception). ftt = *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6300	register read/write failed : reg = *value*, expected / observed = *value* / *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6301	register test-2 read/write failed : reg = *value*, expected / observed = *value* / *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6302	register read/write failed : reg = *value*, expected / observed = *value* / *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6303	FSR Error: expected / observed = *value* / *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6304	Wrong code is given	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6305	`FPU Trap did not occur ,` `i = ` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6306	`FPU Trap Should not occur` `but occured`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6307	`Bus Error did not occur`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6308	`Did not create correct IEEE`
	`exception (Inexact):`
	`expected = 1, observed = `
	value
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6309	`Did not create correct IEEE`
	`exception (Divide By zero)`
	`: expected = 2, observed = `
	value
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6310	`Did not create correct IEEE`
	`exception (Overflow) :`
	`expected = 8, observed = `
	value
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6311	`Did not create correct IEEE`
	`exception (Invalid) :`
	`expected = 10, observed = `
	value
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6312	`Sun FPU Reliability Test`
	`Failed due to segment`
	`violation error`
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6313	`Sun FPU Reliability Test` `Failed due to fpu bus error`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6314	`Number of failures (out of` *value* `cases) = ` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6315	`malloc(0x`*value*`) failed` `:<error_message>`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6316	`Failed ` *value* ` precision` `linpack test`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

6317	`DIVTEST ` *value*`: Number of`
	`failures (out of ` *value* ` cases)`
	`= ` *value*
		Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.

TABLE 17-2 `fputest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6318	`,    expected t` *value*`,` `observed`*value*`,` `expected x` *value*`,`observed*value*`,` `expected y` *value*`, observed` *value*`,    expected z` *value*`,` `observed` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6319	`,    expected t` *value*`,` `observed` *value* `/n  expected x` *value*`, observed` *value* `,` `expected y` *value*`, observed` *value* `,   expected z` *value*`,` `observed` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6320	`Failed` *value* `precision Cora` `test:Check Sum =` *value* `Expected` *value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6321	*value*	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6322	`Sun FPU Reliability Test` `Failed due to floating` `point exception error`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
6323	`Sun FPU Reliability Test` `Failed due to illegal` `instruction error`	Faulty CPU module.	Retry test. If the problem persists, contact your Sun service provider.
8000	`Failed systest for FPU`		
8001	`Floating point exception` `interrupt`		

Sun GigabitEthernet Test (`gemtest`)

The `gemtest` provides functional test coverage of the Sun GigabitEthernet SBus and PCI bus Adapters. It runs in loopback (external/internal) mode and must be selected mutually exclusive with the nettest. The `gemtest` provides better fault isolation as compared to `nettest`.

No network configuration is necessary, however, you must have the GigabitEthernet card and the `ge` device driver installed and a loopback connector in place before running `gemtest`.

The `gemtest` uses DLPI RAW mode to talk to the device driver. For the purpose of this test, a packet is defined as an Ethernet header followed by the Ethernet data payload (refer to IEEE 802.3z). The test generates and sends out the desired number of packets (a tunable parameter) and expects to receive the same number of packets through the loopback interface, external or internal. If an error occurs (for example, packet mismatch or timeout), an error message indicating the type of error, its probable cause(s) and recommended action(s) is displayed on the SunVTS console.

The data sent out is generated by a random number generator, and put into a data buffer. Each time the packet sent is selected from a different starting point of the data buffer, so that any two consecutively transmitted packets will not be the same.

Note – Do NOT run `nettest` and `gemtest` at the same time.

Note – To run `gemtest`, you must have a loopback connector connected to the Sun GigabitEthernet card. `gemtest` cannot run if the network interface is connected to a live network, however the link must be up. A loopback connector provides the network interface driver the necessary link for testing, while maintaining isolation from a live network. The loopback connector is required for both the internal and external tests.

`gemtest` Options

FIGURE 18-1 `gemtest` Option Menu

Refer to TABLE 18-1 for test parameter descriptions.

TABLE 18-1   `gemtest` Options

`gemtest` **Options**	**Description**
Configuration	Specifies the Port Address, Host ID and Domain Name of the system under test.
Total Packets	Specifies the total number of the packets to send. The default number of packet is 1000.
Packet size	Determines the size (in bytes) of the packets to be transmitted. 60 <= packet size <= 1514. The default packet size is 1000 bytes
Loopback	Determines the external and internal loopback mode. The default setting is internal loopback mode.
Print_Warning	Enables or disable the printing of warning messages. The default setting is Disable
Processor Affinity	Binds the test to a specific processor. If no processor is specified, the test migrates between processors. This option is only available on multiprocessor systems.

`gemtest` Test Modes

The `gemtest` can only be run in Functional test mode. It is assumed that the host is not connected to the network through the intended test device(s).

gemtest Command Line Syntax

/opt/SUNWvts/bin/gemtest *standard_arguments*
-o dev=*device***,tpkts=***n***,pksz=***pkt_size***,lb=***Internal*
,warn=*Disable*

TABLE 18-2 gemtest Command Line Syntax

Argument	Explanation
dev=*device_name*	Specifies the device to test such as ge0
tpkts=*n*	[1..100000], count of packets to loopback
pksz=*pkt_size*	[60 ... 1514], packet size in bytes
lb=*Internal*	select internal (or external) loopback mode
warn=*Disable*	enable or disable printing of warning messages

Note – 64-bit tests are located in the sparcv9 subdirectory: /opt/SUNWvts/bin/
sparcv9/*testname*. If a test is not present in this directory, then it may only be
available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on
page 5.

`gemtest` Error Messages

TABLE 18-3 `gemtest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	`Timed out for receiving pkt` *number*	Card and/or driver not configured properly.	Configure driver or reseat card. Check loopback connector. If the problem persists, contact your authorized Sun service Provider.
6001	system error	System may have run out of memory.	Reduce the number of processes currently running or increase system memory.
6002	`Data mismatch between transmitted and received pkt data`	Link loss, CRC error, Alignment Error.	If the frequency of errors is large, replace the card and/or contact your authorized Sun service Provider.
8000	*errmsg* `for device`	Card not present	Check for card.
8001	`Memory allocation failure`	System ran out of memory.	Reduce the number of processes currently running. Increase system memory.
8002	`No card found`	No card found.	Install card.
8003	`Failed to get the link up`	Loopback connector not connected.	Faulty loopback connector. Check it an replace it if necessary. If problem persists, contact your authorized Sun service Provider
8004	`Transmit failed` *error*	STREAMS was unable to accept more data, perhaps due to insufficient STREAMS memory resources.	Reduce the system load.
8005	`Receive failed, error for pkt` *number*		Try again, if the problem persists, then it is a system error. Contact your authorized Sun service provider.
8006	DLPI error for device *string*	Wrong Instance number X specified in geX	Specify correct instance number.

Intelligent Fibre Channel Processor Test (`ifptest`)

`ifptest` tests the functionality of the PCI FC_AL card when there are no devices attached to the loop. The driver checks for devices on the fibre loop. If devices are detected the driver blocks any diagnostic commands.

Note – When devices are attached to the loop, do not run `ifptest`. Instead, run `disktest` tests on the individual devices. This will test the whole subsystem including the FC_AL controller.

`ifptest` uses the "mailbox" interface to the card. This interface allows certain firmware operations to be performed that normally would not be available to the application layer.

There are four subtests that are run in online and functional modes:

■ Mailbox loopback Test

Loads a series of registers into the input mailboxes on the card and then reads the output mailboxes and compares results. This verifies that the system side of the card is operating correctly, and that the internal data paths are ok.

■ Firmware revision check

Reads the firmware revision from the firmware and compares it against a revision loaded by the driver.

■ Checksum firmware test

Runs an internal checksum test on the installed firmware. This verifies that the RISC ram on the card is fully functional and that the installed firmware is still intact. This test also serves as a quick ram check of the RISC ram.

■ Dump revision levels

Extracts the hardware and firmware revision levels of different sub modules on the card.

`ifptest` Options

```
┌──────────────────────────────────────────────────────────┐
│ ─┤            IFP FC_AL Controller                        │
├──────────────────────────────────────────────────────────┤
│ Configuration :                                           │
│                                                           │
│         IFP controller at /devices/pci@1f;2000/pci1077;2100@2 │
│                                                           │
│ Options :                                                 │
│                                                           │
│              Mailbox Loopback Test:  [ Enable | Disable ] │
│                                                           │
│            Firmware Revision Check:  [ Enable | Disable ] │
│                                                           │
│            Firmware Checksum Test:   [ Enable | Disable ] │
│                                                           │
│           Check Module Revisions:    [ Enable | Disable ] │
│                                                           │
│        ( Reset )     ( Apply )                            │
└──────────────────────────────────────────────────────────┘
```

`ifptest` Option Menu

TABLE 19-1   `ifptest` Options

`ifptest` **Options**	**Description**
Mailbox Loopback Test	Enable	Disable the mailbox loopback command. This test writes data patterns into the mailboxes and then reads them back from the output mailboxes and verifies the data is correct.
Firmware Revision Check	Enable	Disable the firmware revision check command. This test extracts the firmware revision from the RISC firmware code and verifies against expected values.
Firmware Checksum Test	Enable	Disable the firmware checksum command. This command instructs the interface's RISC processor to calculate the checksum on the current microcode and then compare it against the checksum that was loaded in with the microcode.
Check Module Revisions	Enable	Disable the firmware checksum command. This command returns the revision level of several sub-modules on the interface card. Although this test is executed when enabled, the module revision levels are only printed out in VERBOSE mode.

`ifptest` Command Line Syntax

`/opt/SUNWvts/bin/ifptest` *standard_arguments*
`-o dev=`*device name*`,mbox=`*Enable | Disable*`,fwrevcheck=`*[Enable | Disable]*`,`
`checksum=`*[Enable | Disable]*`,modrevcheck=`*[Enable | Disable]*

TABLE 19-2   `ifptest` Command Line Syntax

Argument	Explanation
`dev`	The name of the device to test.
`mbox`	*Enable	Disable* the mailbox loopback command. This test writes data patterns into the mailboxes and then reads them back from the output mailboxes and verifies the data is correct.
`fwrevcheck`	*Enable	Disable* the firmware revision check command. This test extracts the firmware revision from the RISC firmware code and verifies against expected values.
`checksum`	*Enable	Disable* the firmware checksum command. This command instructs the interface's RISC processor to calculate the checksum on the current microcode and then compare it against the checksum that was loaded in with the microcode.
`modrevcheck`	*Enable	Disable* the firmware checksum command. This command returns the revision level of several sub-modules on the interface card. Although this test is executed when enabled, the module revision levels are only printed out in VERBOSE mode.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`ifptest` Test Modes

TABLE 19-3 `ifptest` Test Modes

Test Mode	Description
Functional mode	`ifptest` runs the full set of mailbox tests
Functional mode through SyMON	`ifptest` runs the full set of mailbox tests
Connectivity	`ifptest` only performs an open/close

Note – Connectivity mode will only open the controller to verify that the path is still viable.

`ifptest` Error Messages

TABLE 19-4 `ifptest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	`Mailbox loopback failed`	FC_AL board, System software.	Replace FC_AL board. If the problem persists, contact your authorized Sun service provider.
6001	`Error parsing command line options`	Error in command line.	Check your command line and re-enter it if necessary.
6002	`Can not associate device with` *string*`.`	New hardware added since VTS probe or system software.	Stop all testing and re-probe. If the problem persists, contact your authorized Sun service provider.
6003	`FW Stop command returned` *data*`, expected 0.`	IFP Firmware or a system software.	Reload IFP firmware. If the problem persists, contact your authorized Sun service provider.
6004	`Command #`*data* `(cmd=value) did not successfully complete, status: Obs 0x`*value*`, Exp 0x`*value*	IFP Firmware or a system software.	Reload IFP firmware. If the problem persists, contact your authorized Sun service provider.

TABLE 19-4 `ifptest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6005	Chained command number *data* failed to complete.	IFP Firmware or a system software.	Reload IFP firmware. If the problem persists, contact your authorized Sun service provider.
6006	*data* errors were observed during this cycle.		
6007	Command chain aborted: *data* commands queued, *data* commands run		
6008	Command Issued: mbox[0]=0x*value*, mbox[1]=0x*value*, mbox[2]=0x*value*, mbox[3]=0x*value*, mbox[4]=0x*value*, mbox[5]=0x*value*, mbox[6]=0x*value*, mbox[7]=0x*value*		
6009	Result Mboxes: mbox[0]=0x*value*, mbox[1]=0x*value*, mbox[2]=0x*value*, mbox[3]=0x*value*, mbox[4]=0x*value*, mbox[5]=0x*value*, mbox[6]=0x*value*, mbox[7]=0x*value*		
6010	HCCR=0x*value*, BUS_SEMA=0x*value*, ISR=0x*value*, ICR=0x*value*, CDMA_COUNT=0x*value*, CDMA_ADDR=0x*value*		
6011	CDMA_STATUS=0x*value*, CDMA_CONTROL=0x*value*, RDMA_COUNT=0x*value*, RDMA_ADDR=0x*value*, RDMA_STATUS=0x*value*, RDMA_CONTROL=0x*value*		
6012	TDMA_COUNT=0x*value*, TDMA_ADDR=0x*value*, TDMA_STATUS=0x*value*, TDMA_CONTROL=0x*value*, RISC_PSR=0x*value*, RISC_IVR=0x*value*		

TABLE 19-4 `ifptest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6013	RISC_REGS: RR0=0x*value*, RR1=0x*value*, RR2=0x*value*, RR3=0x*value*, RR4=0x*value*, RR5=0x*value*, RR6=0x*value*, RR7=0x*value*		
6014	RISC_REGS: RR8=0x*value*, RR9=0x*value*, RR10=0x*value*, RR11=0x*value*, RR12=0x*value*, RR13=0x*value*, RR14=0x*value*, RR15=0x*value*		
6015	RISC_PCR=0x*value*, RISC_RAR0=0x*value*, RISC_RAR1=0x*value*, RISC_LCR=0x*value*, RISC_PC=0x*value*, RISC_SP=0x*value*		
6016	REQUEST_IN=0x*value*, REQUEST_OUT=0x*value*, RESPONSE_IN=0x*value*, RESPONSE_OUT=0x*value*		
6017	CURRENT_REQ_PTR=0x*value*, BASE_REQ_PTR=0x*value*, CURRENT_RESP_PTR=0x*value*, BASE_RESP_PTR=0x*value*		
6018	Interface at string is currently not testable, there may be devices attached.		
6019	Bad option passed to test: *string*		
6020	No device detected for ifp*data*		
6021	Cannot associate an IFP device with *string*.		
6021	No IFP devices detected on this system.		
8000	*string*: invalid file descriptor *data* passed in.	Programming error or system software.	If the problem persists, contact your authorized Sun service provider.
8001	*string*: IOCTL call failed: *string*.	wrong/missing IFP driver.	Contact your system administrator to load correct driver. If the problem persists, contact your authorized Sun service provider.

TABLE 19-4   `ifptest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8002	*string*: Firmware checksum failed, Rec: *value*, Exp: 0x0	Unsupported or bad firmware loaded or bad ifp controller.	Load the correct ifp driver. If the problem persists, contact your authorized Sun service provider.
8003	*string*: Get Revision mailbox command failed, return code 0x*value*.	Incorrect ifp firmware loaded or a system software problem.	Load the correct ifp driver. If the problem persists, contact your authorized Sun service provider.
8004	*B*ad device name passed to test: *string*	Wrong entry on command line.	Check and retype the command line. If the problem persists, contact your authorized Sun service provider.
8005	Bad option passed to test: *string*	Wrong entry on command line.	Check and retype the command line If the problem persists, contact your authorized Sun service provider.
8006	Open of IFP device string failed; *string*	System software problem.	Contact your system administrator. If the problem persists, contact your authorized Sun service provider.
8007	Open of /etc/path_to_inst file failed; *string*	System software problem.	Contact your system administrator. If the problem persists, contact your authorized Sun service provider.
8008	Get firmware revision IOCTL failed: *string*	System software problem.	Contact your system administrator. If the problem persists, contact your authorized Sun service provider.
8009	Device *string* is untestable, devices attached to loop	Devices attached to controller or a system software problem.	Remove interface connections and try again.
8010	Open ctrlr string as RDWR failed. Reason: *string*	Device name incorrect (command line only), or a system software problem.	Check and re-enter the command line. If the problem persists, contact your authorized Sun service provider.
8011	Device string is untestable, reason: *string*	Device name is incorrect (command line only), or the wrong driver is installed or a system software problem.	Check and re-enter the command line. Verify that the correct driver is installed. If problem persists, contact your authorized Sun service provider.

M64 Video Board Test (`m64test`)

`m64test` tests the PCI-based M64 video board by performing the following subtests:

- Video Memory test
- RAMDAC test
- Accelerator Port test

Caution – DO NOT run any other application or screen saver program that uses the Pineapple accelerator port while running `m64test`. Do not run power management software. These programs cause SunVTS to return incorrect errors.

Note – Disable all screen savers before testing any graphics device. Type **xset s off** at a UNIX prompt to disable the Solaris screen saver. Disable power management software if it is running.

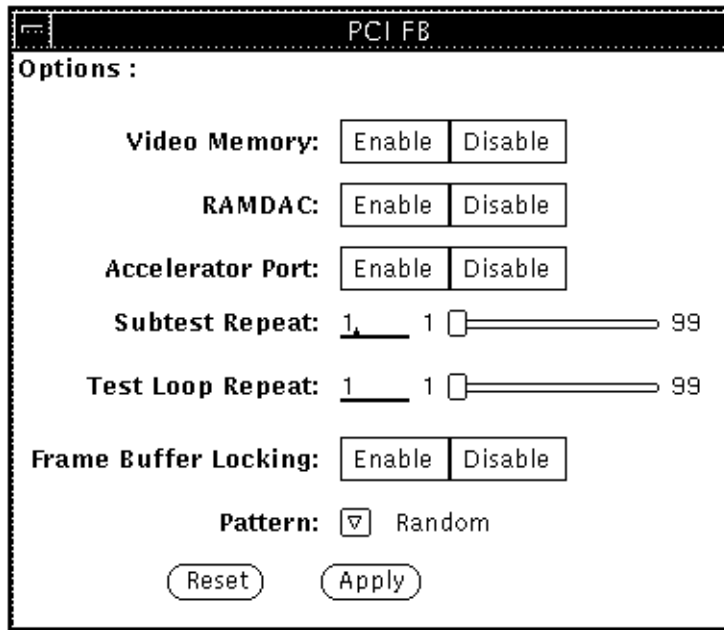**Note –** DO NOT run Open Windows across multiple monitors.

Note – To start SunVTS with `vtsui`, but without `vtsk`, you must add the host name to `xhost` as: **xhost +** *hostname.*

`m64test` Options

By default, all options are enabled except frame buffer locking.

FIGURE 20-1 `m64test` Option Menu

TABLE 20-1 `m64test` Options

`m64test` **Options**	**Description**
Video Memory Test	Thoroughly tests the on-screen video memory (the memory part that is mapped on to the monitor) of the M64 video board in 8-bit, 16-bit, 32-bit, 64-bit, and 64 byte (block) modes. Entire on screen video memory is tested by testing 512 bit blocks at a time (8x8 pixel block). Each block is tested in two passes. Each pass consists of a data write and read. In the first pass user specified data or random data is used and in the second pass one's complement of the data used in the first pass is used so that each on-screen video memory location (bit) is tested with a zero (electrical low state) and one (electrical high state).

TABLE 20-1   `m64test` Options

`m64test` **Options**	**Description**
RAMDAC Test	Tests the RAMDAC in three phases. In the first phase the RAMDAC CLUT (Color LookUp Table) is tested using simple write/read patterns to determine if there are any bad bits in CLUT. The data patterns used are: • Random data • Complement of the random data (used as first data pattern) • The data pattern 0101 • The data pattern 10101 In the second phase four different patterns are drawn on the screen. Each pattern stays on the screen for approximately three seconds. The four patterns that are drawn are listed below. For each pattern signature is captured and compared with the signature obtained for the same pattern on a known good board. This test verifies that all the different data paths within the RAMDAC are functioning properly. Patterns drawn on screen: • Red ramp with cursor at top-left corner of the screen • Blue ramp with cursor at top-right corner of the screen • Green ramp with cursor at bottom-left of the screen • Grey ramp with cursor at bottom-right of the screen In the last (third) phase of the RAMDAC test the Vertical Retrace Interrupt is tested for 300 interrupts.
Accelerator Port Test	Tests all the data paths (sources: fixed color, host data, blit, fixed pattern), arithmetic and logic unit (ALU), color comparator, primitives (destinations: line, rectangle), and mono to color expansion logic. Primitives are drawn using a combination of different data paths (allowed), ALU functions, and color comparator functions. A checksum is generated for each data combination and is compared with the checksum generated for the same data combination on a known good board.
Frame Buffer Locking	This option is set to *disable* if the M64 is not the console device. When Sunvts GUI is brought up FB Locking is enabled by default if M64 is console device. If M64 is not console device, FB Locking is disabled by default.

`m64test` Test Modes

Due to the nature of graphic tests, reading from or writing to the frame buffer during graphic tests will disturb user operation. This test is only available in the Functional Test mode.

TABLE 20-2 `m64test` Test Mode

Test Mode	Description
Functional Test	The `m64test` verifies the M64 video board.

`m64test` Command Line Syntax

/opt/SUNWvts/bin/m64test *standard_arguments* **-o dev=***device_name,*
S=*subtest_number,***F=***#_of_subtest_loops,***B=***#_of_test_loops,***L=disable,P=***test_pattern*

TABLE 20-3 `m64test` Command Line Syntax

Argument	Explanation
dev=*device_name*	*device_name* is the relative path name of the device being tested with respect to /dev/fbs; The default is m640.
S=*subtest_number*	*subtest_number* is the test number of the subtest to be run. Select from the subtests below. You can run multiple subtests by adding the subtest numbers. For example, `n=0x00003` runs both test 00001 and test 00002; `n=0x00005` runs both test 0x00001 and test 0x00004. Note that you do not need the leading zeros. **n - 0x00001 VRAM** **n - 0x00002 RAMDAC** **n - 0x00004 Accelerator port test (Rendering Pipeline)** More than one test can be selected by ORing subtest numbers. For example: n = 0x00005 means VRAM and Rendering Pipeline tests. A hex number must be preceded by 0x, decimal numbers are also acceptable.
F=*#_of_subtest_loops*	Specifies the number of times to repeat each subtest; the default is 1.

TABLE 20-3 `m64test` Command Line Syntax

Argument	Explanation (Continued)
B=_#_of_test_loops_	Specifies the number of times to repeat a test loop before passing; default is 1.
L=disable	_Disables the frame buffer lock._Disable the lock when the m64 is not the console or when the server is not running on the m64 under test.
P=_test_pattern_	Specifies the test pattern number. The default is **r**, for random patterns. You may also choose **0 for 0x0000000**, **3 for 0x3333333**, **5 for 0x5555555**, or **9 for 0x9999999**.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`_testname_. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`m64test` Error Messages

Errors returned by `m64test` are nonspecific. It is not possible to determine which component caused a failure. In all error conditions, the field replaceable unit (FRU) is the entire M64 video board. The errors messages are listed below, along with probable causes and recommended actions.

TABLE 20-4 `m64test` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	Out of memory	System error.	If problem persists, call your authorized Sun service provider.
6002	_string_ error: Misaligned raster.	Software error.	If problem persists, call your authorized Sun service provider.
6003	Unable to open _string_.	Device not present.	Check for device presence.
		Device failure.	Replace device.
		Permission restrictions.	Check device permissions.

TABLE 20-4 `m64test` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6004	`ioctl(2) Error in loading the Frame Buffer structure fbtype.`	`IOCTL call to driver failure.`	Try replacing device.
		`Device    failure.`	Call your authorized Sun service provider.
6005	`ioctl(2) Error in loading the Original Color Map Values.`	IOCTL call to driver failure.	Try replacing device.
		Device failure.	Call your authorized Sun service provider.
6006	`ioctl(2) Error in Setting the Cursor Values.`	IOCTL call to driver failure.	Try replacing device.
		Device failure.	Call your authorized Sun service provider.
6007	`ioctl(2) Error in loading the GXInfo Structure fbtype.`	IOCTL call to driver failure.	Try replacing device.
		Device failure.	Call your authorized Sun service provider.
6008	*string* `(1-byte access) error: x/y (`*data*`) Expected 0x%02x, Actual  0x%0  2x,  Xor 0x%02x.`	Device failure.	Replace device.
			Call your authorized Sun service provider.
6009	*string* `(2-byte access) error:` `x/y (`*data*`,`*data*`) Expected 0x%04x,   Actual   0x%0 4x,   Xor    0x%04x.`	Device failure.	Replace device.
			Call your authorized Sun service provider
6010	*string* `(4-byte access) error: x/y (`*data*`,`*data*`) Expected 0x%08x, Actual  0x%0  8x,  Xor 0x%08x.`	Device failure	Replace device.
			Call your authorized Sun service provider.
6011	*string* `(8-byte access) error: x/y (`*data*`,`*data*`) Expected 0x%08x, Actual  0x%0  8x,  Xor 0x%08x.`	Device failure	Replace device.
			Call your authorized Sun service provider.
6012	*string* `(64-byte access) error: x/y (`*data*`,`*data*`) Expected 0x%08x, Actual  0x%  08x,  Xor 0x%08x.`	Device failure.	Replace device.
			Call your authorized Sun service provider.

TABLE 20-4 `m64test` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6013	`Error in Allocating Memory mmap call Failure.`	System error.	If problem persists, call your authorized Sun service provider.
6014	`Error in DeAllocating Memory unmap call Failure.`	System error.	If problem persists, call your authorized Sun service provider.
6015	`ioctl(2) failed trying to turn on M64 video.`	IOCTL call to driver failure.	Replace device.
		Device failure.	Call your authorized Sun service provider.
6016	`ioctl(2) failed trying to post CLUT.`	IOCTL call to driver failure.	Replace device.
			Call your authorized Sun service provider.
		Device failure.	
6017	`ioctl(2) failed trying to read CLUT.`	IOCTL call to driver failure.	Replace device.
		Device failure.	Call your authorized Sun service provider.
6018	`Miscompare found in Red channel of CLUT entry` *data*`. Expected 0x%02x, Ob served 0x%02x, XOR 0x%02x.`	RAMDAC failure.	Replace device.
		Device failure.	Call your authorized Sun service provider.
6019	`Miscompare found in Green channel of CLUT entry` *data*`. Expected 0x%02x, Observed 0x%02x, XOR 0x%02x.`	RAMDAC failure	Replace device.
		Device failure	Call your authorized Sun service provider.
6020	`Miscompare found in Blue channel of CLUT entry` *data*`. Expected 0x%02x, O served 0x%02x, XOR 0x%02x.`	RAMDAC failure.	Replace device.
		Device failure	Call your authorized Sun service provider.
6021	`RAMDAC test screen CRC signature error message. Monitor type` *data*`, Cursor Position` *data*`, Expected Signature 0x`*value*`, Actual Signature 0x`*value*	Device failure.	Call your authorized Sun service provider.

TABLE 20-4 m64test Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6022	sysinfo(2) failed getting hostname.	No system permissions.	Become superuser. If problem persists, call your authorized Sun service provider.
6023	Can't acquire console semaphore	System error.	If problem persists, call your authorized Sun service provider.
6024	Cannot gain control of mouse:*string*	System error.	If problem persists, call your authorized Sun service provider.
6025	Cannot gain control of keyboard:*string*	System error.	If problem persists, call your authorized Sun service provider.
6026	Already grabbed by another client		Try stopping other similar actions. If problem persists, call your authorized Sun service provider.
6027	grabbed at invalid time	System error.	If problem persists, call your authorized Sun service provider.
6028	grab window is not viewable		If problem persists, call your authorized Sun service provider.
6029	pointer/keyboard is frozen		If problem persists, call your authorized Sun service provider.
6030	Unknown cause		If problem persists, all your authorized Sun service provider.
6031	ioctl(2) failed trying to get current video mode.	IOCTL call to driver failure Device failure.	Replace device. Call your authorized Sun service provider.
6032	ioctl(2) failed trying to set current video mode.	IOCTL call to driver failure. Device failure.	Replace device. Call your authorized Sun service provider.

TABLE 20-4 m64test Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6033	ioctl(2) failed trying to get M64 system info.	IOCTL call to driver failure.	Replace device.
		Device failure.	Call your authorized Sun service provider.
6034	mmap(2) failed:*string*.	System error.	If problem persists, all your authorized Sun service provider.
6035	munmap(2) failed:*string*.	System error	If problem persists, all your authorized Sun service provider.
6036	close(2) failed:*string*.	System error.	If problem persists, all your authorized Sun service provider.
6045	Can't access X Windows	Permission denial from Host.	Run xhost(1).
		System error.	If X server is not running, run X server.
		X Server is not running.	
			If problem persists, all your authorized Sun service provider.
6046	wait(2) failed:*string*	System error.	If problem persists, all your authorized Sun service provider.
6047	signal(2) failed:*string*	System error.	If problem persists, all your authorized Sun service provider.
6048	ioctl(2) Error in vertical retrace test.	IOCTL call to driver failure.	Replace device.
		Device failure.	Call your authorized Sun service provider.
6049	Can't open display	Permission denial from Host.	Run xhost(1).
		System error.	If X server is not running, run X server.
		X Server is not running.	
			If problem persists, call your authorized Sun service provider.

TABLE 20-4 m64test Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6050	m64 open file failed Could not open the signature file *string*: *string*	No file permission or file does not exist.	Check the file permissions. If problem persists, call your authorized Sun service provider.
6051	m64 FTRUNCATE failed in checker. Accelerator could not extend or truncate the signature file.	Insufficient disk space.	Create more space by deleting unnecessary files. If problem persists, call your authorized Sun service provider.
6052	m64 close failed in checker. Accelerator could not close signature file.	System error.	If problem persists, call your authorized Sun service provider.
6053	m64 check error in checker	Device failure.	Replace device. If problem persists, call your authorized Sun service provider.
6054	m64 ROP test failed	Device failure.	Replace device. If problem persists, call your authorized Sun service provider.
6055	m64 IOCTL fails FBIOPUTCMAP Error while putting color map.	Device failure.	Check software installation. Replace device. If problem persists, call your authorized Sun service provider.
6056	m64 IOCTL fail FBIOGETCMAP. Error while getting color map. IOCTL call to deriver failure	Device failure.	Check software installation. Replace device. If problem persists, call your authorized Sun service provider.

TABLE 20-4 `m64test` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6057	`Internal error Mon XLATOR Internal error occurred while translating monitor type to index into signature file`	Monitor types are modified.	If problem persists, call your authorized Sun service provider.

Multiprocessor Test (`mptest`)

`mptest` verifies the functionality of multiprocessing hardware. This test allocates a page of virtual memory for the test, declaring the page shared, locks the page against swapping, and creates threads to each of the processors being tested. Up to 256 processors can be tested by `mptest` in a CPU.

The processor mask argument can used during test probing. The `mptest` verifies that the current processor mask matches the argument you entered in the command line or from the GUI/TTYUI.
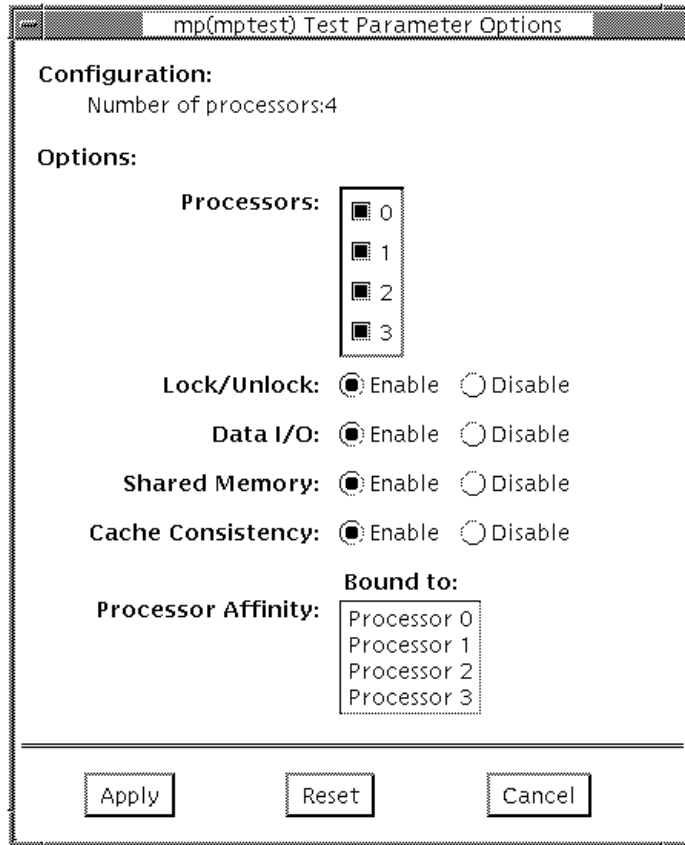
`mptest` Options

FIGURE 21-1 `mptest` Option Menu

The processors that can be tested are listed in the Configuration area of the menu. The multiprocessing test can be enabled or disabled for individual processors on this menu.

The options listed in Table 17-1 can be run alone or concurrently with other options.

TABLE 21-1 `mptest` Options

mptest Options	Description
Processors	You can test specific processors by clicking SELECT on the check boxes to enable or disable each processor. A check mark means the processor is enabled for testing; the default setting is all processors enabled. Note that `mptest` requires at least two enabled processors to test multiprocessing systems.
Lock/Unlock	Tests the lock/unlock mechanism that guarantees exclusive access to a physical page to one processor. A thread is created at each of the processors. Each processor uses the SPARC atomic instruction `ldstub` to write to the same shared physical memory page. While one processor is attempting the write, the other processors should be free spinning for their turn. As each processor acquires the lock, it writes an ordinal number to a shared trace buffer using a shared write pointer. After the test cycle is complete, the trace buffer is dumped for analysis. This test fails and returns an error message if the trace buffer does not contain an equal number of ordinal numbers for each processor. For example, if the specified loop count is 5, the trace buffer should contain five 0s, five negative 1s, five 2s, and so on.
Data I/O	Requires two or more threads, each of which locks onto one of the processors. Each processor, in turn, writes data to a temporary file that has been mapped to the physical address. The modified data is immediately read by other processors being tested. This test hangs and fails if the processors do not recognize the expected data.
Shared Memory	A shared memory buffer is divided into a number of contiguous chunks, one for each of the CPUs participating in the test. Each CPU is assigned a unique chunk based upon its ID (1-N). This subtest has two parts. First, each CPU locks and writes data to its data chunk. Identical data is written for each CPU. Then each CPU reads and compares the information on its data chunk with that of another CPU. If two CPUs do not confirm consistent data, the test fails and returns an error message. If that happens, testing stops and this test is run again in verbose mode to return more detailed information.
Cache Consistency	Requires two or more processors to access and write to the same physical address. This test verifies that a change in physical address by one processor is confirmed by another. If two processors do not confirm consistent data, the test continues to run, but the Pass Count in the SunVTS status window stops incrementing. If this happens, stop testing and run the test again in verbose mode for a more detailed picture of the problem.

`mptest` Test Modes

TABLE 21-2 `mptest` Test Modes

Test Mode	Description
Connection Test	The Connection test checks the current processors on the system with the original processor mask. An error is reported if they do not match. The original processor mask is set during probing, which shows the processors on system during the probe. The status of each selected processor is checked by `procsesor_bind`.
Functional Test	This test mode verifies that the current processor mask is the same as that from the command line, or the same as that from the GUI/TTYUI
Functional mode through SyMON	If you invoke SunVTs through SyMON, the following Functional test description applies: This test verifies that the current processor mask is the same as that from the command line, or the same as that from the GUI/TTYUI

`mptest` Command Line Syntax

/opt/SUNWvts/bin/mptest *standard_arguments*
-o M=0+1+2+3...,NL,ND,NS,NC,omask=*hexidecimal_number*

TABLE 21-3 `mptest` Command Line Syntax

Arguments	Description
NL	Disable the Lock/Unlock subtest.
ND	Disable the Data I/O subtest.
NS	Disable the Shared Memory subtest.
NC	Disable Cache Consistency subtest
M=*0+1+2+3...*	Use *0, 1, 2* to specify the processors to test.
omask=*hexidecimal_number*	Original mask of processors. Bit 0 represents processor 0 and bit 1 represents processor 1. An example: 03333320

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/` `sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`mptest` Error Messages

TABLE 21-4   `mptest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	`mp lock failed`	Faulty processor.	If the problem persists, call your authorized Sun service provider.
6001	`All subtests are disabled`	Parameter error.	Enable at least one subtest.
6002	`thr_setconcurrency error`	System load is too heavy.	Reduce the system load.
		System error.	If the problem persists, call your authorized Sun service provider.
6003	`Error in creating threads:` *name*	System load is too heavy.	Reduce the system load.
		System error.	If the problem persists, call your authorized Sun service provider.
6004	`thr_join() thread` *number* `error status:` *number*	System load is too heavy.	Reduce the system load.
		System error	If the problem persists, call your authorized Sun service provider.
6005	`Processor Mask Error:` `Expect` *number* `Was` *number*	Some processors are disappeared from processor mask.	If the problem persists, call your authorized Sun service provider.
6006	`Failed in system call` `uadmin, errno =` *number*`,` `sys_errlist =` *name*	System load is too heavy.	Reduce the system load.
		System error.	If the problem persists, call your authorized Sun service provider.
6007	`Failed in function` `getnextbitmsk!`	System error.	If the problem persists, call your authorized Sun service provider.
6008	`Processor Mask Error:` `check mask` *number* `cpu mask` *number*	Some processors disappeared from the processor mask.	If the problem persists, call your authorized Sun service provider.

TABLE 21-4 `mptest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6009	`Failed to get shared memory, err = `*name*	System load is too heavy.	Reduce the system load.
		System error.	If the problem persists, call your authorized Sun service provider.
6010	`Failed to create shared memory, err = `*name*	System load is too heavy.	Reduce the system load.
		System error	If the problem persists, call your authorized Sun service provider.
6011	`Shared Memory Subtest: Process `*number*`, Processor `*number*`, *curr_add `*number* `curr_addr `*number*`, cpu_mask `*number* `failed`	System load too heavy.	Reduce the system load.
		System error.	If the problem persists, call your authorized Sun service provider.
6014	`Failed on read & compare: Processor `*number*`, read=`*number*`, exp=`*number*		
6016	`Not a MP system!`	Faulty Processor.	
		Do not run `mptest` on this system.	
8000	`Failed to open `*name*	Not enough disk space.	Allocate disk space.
		Permission error.	Check directory/file permissions.
		System error.	If the problem persists, call your authorized Sun service provider.
8001	`Failed to create a page`	System load too heavy.	Reduce the system load.
		System error.	If the problem persists, call your authorized Sun service provider.
8002	`mmap address space to device`	System load too heavy.	Reduce the system load.
		System error.	If the problem persists, call your authorized Sun service provider.
8003	`mlock error. `*name*	Processor error.	Isolate the bad processor.
8005	`All subtests are disabled`	Parameter error: None subtest is selected.	Re-enter command line correctly: Select at least one subtest.
8006	`Can not run mptest! You need to enable at least two processors`	Parameter error You have selected less than two processors for testing.	Re-enter command line correctly; Select at least two processors.

TABLE 21-4 `mptest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8008	`alloc error:`*error_message*	system load too heavy or system error.	Reduce the system load. If the problem persists, call your authorized Sun service provider.
8010	CPU number *number*: Expected ID *number*, was *number*	Some processors do not appear in the processor mask.	If the problem persists, call your authorized Sun service provider.
8014	Processor *number* is not online or does not exist	Non-existant processor was specified on the command line. Processor selected is off-line.	Check the command line and option file. If the problem persists, contact your authorized Sun service provider.

Network Hardware Test (`nettest`)

`nettest` checks all the networking hardware on the system CPU board and separate networking controllers (for example, a second SBus Ethernet controller). For this test to be meaningful, the machine under test must be attached to a network with at least one other system on the network.

Note – This version of `nettest` is used for *all* networking devices, including Ethernet (`ie` and `le`), token ring (`tr`, `trp`), quad Ethernet (`QED`), fiber optic (`fddi`, `nf`, `bf`, `pf`), SPARCcluster™ 1 System (`em`), ATM (`sa`, `ba`), and 100-Mbits per second Ethernet (`be`,`hme`) devices.

The `nettest` mainly uses the Internet Control Message Protocol (ICMP), and requires at least two machines on a network—the machine under test and another machine reliable enough to be a test target. Both machines must support the transport control protocol/interface program (TCP/IP) (ICMP is part of TCP/IP). The target machine must either be configured to respond to ICMP broadcast or to RPC broadcast.

First `nettest` determines the target machine(s) to test against. If no targets are specified, it sends an ICMP broadcast to find them. If it fails to find all necessary targets, it tries RPC broadcast to the RPC port mapper daemon. If you specify the targets, `nettest` uses the specified target(s) instead.

After finding the necessary targets, `nettest` performs the following tests:

- `Random test` sends out 256 packets with random data length and random data.
- `Incremental test` sends out packets with length from minimum to maximum packet size using incremental data. (Minimum and maximum values differ for each device.)
- `Pattern test` sends 256 packets of maximum length, where each packet contains one test pattern, and all byte patterns (0 to 0xFF hex) are used. That is, the first packet contains pattern 0, the second packet contains pattern 1, and so on, until the last packet pattern of 0xFF.

Note – `nettest` is a scalable test. However, the maximum number of networked devices allowed on a system is 255, and the number of instances for each device is limited to 200. So, if you start the SunVTS exerciser using the `-i` option to specify a default number of instances for all tests, `nettest` cannot assign more than 200 instances per each networked device.

`nettest` Options
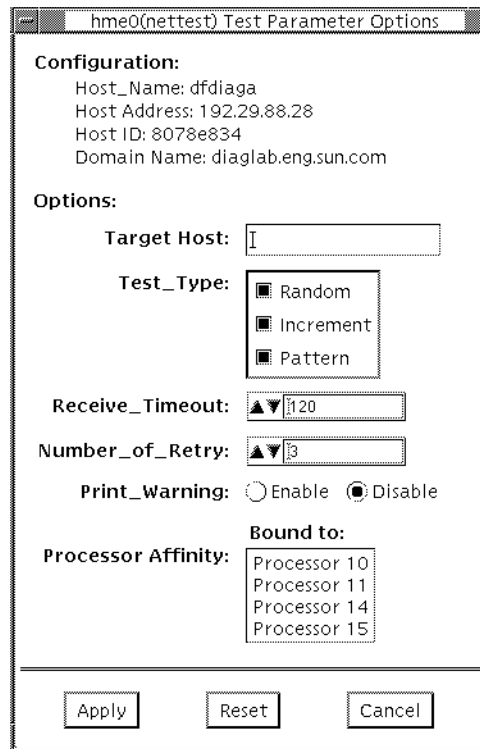
FIGURE 22-1 `nettest` Option Menu

The Configuration section specifies the host name, host ID, host address, and domain name of the system being tested.

TABLE 22-1 `nettest` Options

`nettest` **Options**	**Description**
Target Host	Specifies one or more targets to be tested against. Target host entries can be either a host name or an Internet address. When no target host is specified, the test finds necessary targets through broadcasting. The default setting leaves this field empty.
Receive Timeout field	The default is 120 seconds, but can be changed. Use a range between 0 - 600 seconds.
Number of Retries field	The default number of retries before flagging an error is three, but can be changed. Use a range between 0 to 128 retries.
Print Warning	Disabled by default. Click Enable to see warning errors, such as retry on timeout.

`nettest` Test Modes

All three modes are supported by `nettest`. Different test schemes are performed on the network device based on the mode selected.

TABLE 22-2 `nettest` Test Modes

Test Mode	Description
Connection Test	Checks whether the device is connected. It searches through all the network interfaces fora specified device name. If nettest does not find the device connected the test fails, otherwise it returns: `device is connected.`
Functional Test	Performs all three tests (Random test, Incremental test, and Pattern test) sequentially. It allows you to specify options that will perform heavy stress testing. If you invoke SunVTS through SyMON, the following Functional Test description applies: Online mode, nettest is aware that the user's application may be running, it tries to minimize the influence on the application while it is testing. So the `nettest` uses only random test in online mode. Since the network device can be shared and the system resource is available, applications shall be affected fairly small
Functional mode through SyMON	If you invoke SunVTS through SyMON, the following Functional test description applies: `nettest` is aware that the user's application may be running, it tries to minimize the influence on the application while it is testing. So `nettest` uses only random test in Online mode. Since the network device is shared and system resources are available, applications are minimally affected.

`nettest` Command Line Syntax

`/opt/SUNWvts/bin/nettest` *standard_arguments* `-o target=`*h1+h2+...*`,`
`dev=`*interface*`,test=`*type*`,packets=`*n*`,pattern=`*hex*`,delay=`*seconds*`,`
`timeout=`*seconds*`,retry=`*n*`,warn`

TABLE 22-3 `nettest` Command Line Syntax

Argument	Explanation
`target=`*h1+h2+*`...`	A list of test targets by host name or Internet address.
`dev=`*interface*	Network interface name. The default value is `le0` for Ethernet networks.
`test=`*type*	The test type. Type **Random**, **Increment**, or **Pattern** for the desired test. The default value is Random+Increment+Pattern where all tests run.
`packets=`*n*	Number of random/pattern packets. The default is 256.
`pattern=`*hexn*	Specifies a data pattern, in hexadecimal form. The default is all patterns from `0` to `0xff`.
`delay=`*seconds*	Indicates the time between subtests in seconds; the default is 30 seconds.
`timeout=`*seconds*	Indicates the number of seconds to wait before a timeout; the default is 20 seconds.
`retry=`*n*	Indicates the number of test timeout retries; the default is three retries.
`warn`	When enabled, prints warning messages.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/` `sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`nettest` Error Messages

TABLE 22-4 `nettest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	`RPC broadcast failure` *error_message*		
6001	`Unable to resolve` *host_name* `Ethernet address`		
6002	`No ICMP echo reply from` *host_name*	System load too heavy.	Reduce system load or increase timeout time.
		No cable connection.	Check cable connection!
		Target machine too busy.	Reduce target machine load.
6003	`ICMP echo reply incorrect length from` *host_name*`, exp` *data* `obs` *data*		
6004	`ICMP echo reply length` *number* `from` *host_name*`, data mismatch at byte` *number*`, exp` *value* `obs` *value*		
6005	`Unable to find any test target`		
6006	`Unable to find necessary test targets for the ethernet switch`		
8000	`Unable to open` *device_name* *error_message*		
8001	`Bad ioctl` *name* *error_message*		
8002	`Unable to allocate` *number* `bytes of memory` *error_message*		
8003	`System call` *name* `returned` *value* *error_message*		
8004	`Unable to map network entry for` *host_name*		
8005	`ATM device` *device_name* `requires target specification`		

TABLE 22-4 `nettest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8006	Failed to allocate *number* bytes of swap space for configuration		
8007	dl_primitive response = *value*		
8008	dl_ok_ack: short response		
8009	dl_phys_addr_ack: short response		
8010	Invalid Argument string		
8011	Interface *device_name* not configured in system		

PCMCIA Modem Card Test (`pcsertest`)

`pcsertest` verifies the functionality of the PCMCIA modem card. The `pcsertest` issues a series of commands to the modem that instructs the modem's firmware to run an internal analog loopback diagnostic test. Upon completion, the firmware sends back a three-digit status message indicating whether the test passed or failed.

As an option, the `pcsertest` tests socket I/O cards. This test writes an 8-Kbyte incrementing data pattern to the I/O card, which is then looped back, read, and verified.

Note – When testing socket I/O cards, a 9-pin loopback connector is required. However, no loopback connector is required when testing the default modem card. See Appendix A for loopback connector wiring instructions.

`pcsertest` Options
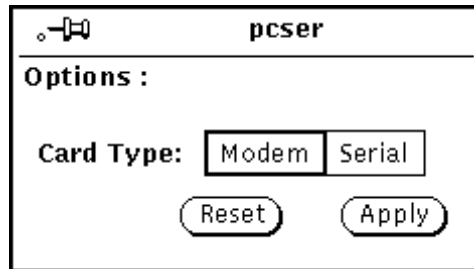
FIGURE 23-1 `pcsertest` Option Menu

The default card type for each PCMCIA slot is a modem card. If only one modem card is plugged in, the empty slot is ignored.

From the `pcsertest` Option window, you can choose to test socket I/O cards. To test a socket card, choose Socket on the Card Type switch. If you choose to test a socket I/O card in an empty slot, the test fails.

Any combination of modem and socket I/O cards can be placed in the PCMCIA slots. However, you must select the correct type of card in the Option window. If you select an incorrect card type, the test fails.

`pcsertest` Test Mode

This test only supports Functional Test mode.

`pcsertest` Command Line Syntax

/opt/SUNWvts/bin/pcsertest *standard_arguments* **-o dev=***device_name*,
type=*card_type*

TABLE 23-1 `pcsertest` Command Line Syntax

Argument	Explanation
dev=*device_name*	The device name (for example, dev=pc0 and pc1)
type=*card_type*	The card type for the previously specified device name (for example, type=serial). You do not need to specify the type if the device is a modem, since modem is the default card type.

Note – 64-bit tests are located in the `sparcv9` subdirectory: /opt/SUNWvts/bin/ sparcv9/*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`pcsertest` Error Messages

TABLE 23-2 `pcsertest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	tcgetattr failed for modem card *number error message*		
6001	tcsetattr failed for modem card *number* error message		
6002	Write failed while setting modem card *number error message*		
6003	ioctl TIOCSSOFTCAR failed for serial card *number error message*		

TABLE 23-2 `pcsertest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6004	`tcgetattr failed for serial card` *number* *error message*		
6005	`tcsetattr failed for serial card` *number* *error message*		
6006	`Card in slot` *number* `is not a serial card`		
6007	`Data miscompare error`		
6008	`Modem card` *number* `returned unknown status`		
6009	`ioctl TIOCMSET failed for serial card` *number* *error message*		
6010	`ioctl TIOCMBIS failed for serial card` *number* *error message*		
6011	`Write to serial card` *number* `failed.` *error message*		
6012	`Timeout waiting for data from serial card` *number*		
6013	`Read from serial card` *number* `failed.` *error message*		
6014	`Status timeout from modem card` *number*		
6015	`Read from modem card` *number* `failed.` *error message*		
6016	`Card in slot %d is not a modem card`		
8000	`can't open <device>` *error message*		
8001	`Serial card` *number* `loopback defective`		
8002	`Write to modem card` *number* `failed.` *error message*		

SPARCstorage Array Controller Test (`plntest`)

`plntest` checks the functionality of the controller board on the SPARCstorage™ Array. The SPARCstorage Array (SSA) is a large disk storage I/O subsystem capable of housing up to 30 SCSI hard drives. The SSA communicates with a host system over a fiber-optic link provided by an SBus-based host adapter card in the host system and the corresponding SSA controller board hardware.

The SSA controller card is an intelligent, CPU-based board with its own memory and ROM-resident software. In addition to providing a communications link to the disk drives, it also buffers data between the host system and disk drives in its nonvolatile RAM (NVRAM). For data to go from the host to a particular disk, it must first be successfully transferred to this NVRAM space.

The host machine, SBus host adapter card, fiber-channel connection, and the SSA controller board must be working properly to perform this data transfer operation. By verifying and stressing this operation, `plntest` can isolate failures on the SSA disk drives from failures on the SSA controller board.

Note – `disktest` transfers data on the SSA disk drives over the same path mentioned above. However, `disktest` does not transfer data as quickly as `plntest`.

`plntest` Controller Test

The `plntest` exercises the hardware and software by invoking SCSI read buffer commands of various sizes to the NVRAM. These operations exercise the host fiber channel hardware, the SSA fiber channel hardware, the SSA resident management

software, and the hardware component interaction on the SSA controller card (all components except the SCSI devices). In addition, the `plntest` reports failure of the fan module and the NVRAM battery module of the SPARC storage array.

`plntest` Options
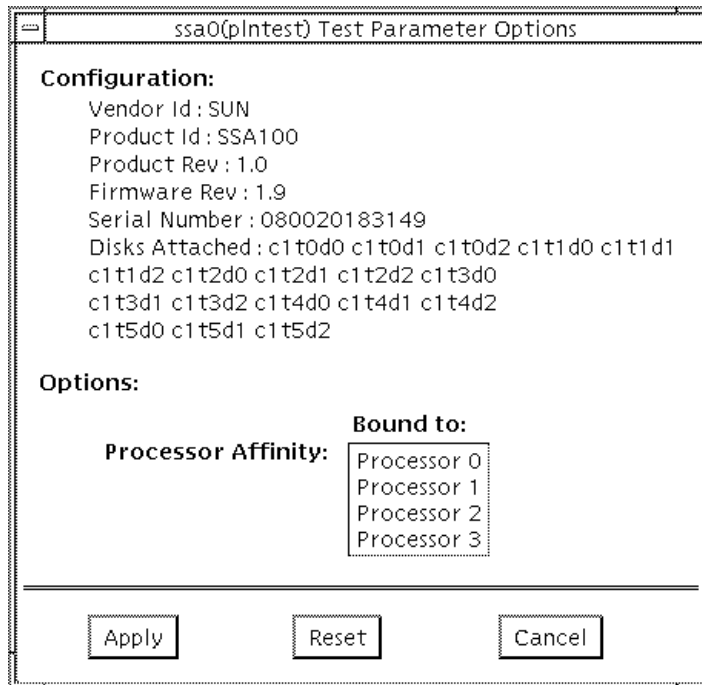
FIGURE 24-1 `plntest` Option Menu

Configuration lists the names of all the logical disk drives (both single and grouped) that are attached to the SPARCstorage Array controller board. `plntest` is not a scalable test.

Note – If no disks are present, `none` is displayed under the Attached Disks heading.

`plntest` Test Modes

TABLE 24-1 `plntest` Test Modes

Test Mode	Description
Connection Test	In this mode, `plntest` checks the state of the battery module and the fan module of the SPARCstorage array.
Functional Test	Performs the same type of tests as Online mode.
Functional mode through SyMON	If you invoke SunVTS through SyMON, the following Functional test description applies: `plntest` checks the state of the battery module and the fan module of the SPARCstorage array. The `plntest` issues a SCSI command, `READBUFFER`, causing the entire NVRAM to be read. This test uses different transfer buffer sizes

`plntest` Command Line Syntax

`/opt/SUNWvts/bin/plntest` *standard_arguments* **`-o ?,dev=`***device_name***`,x,`**

TABLE 24-2 `plntest` Command Line Syntax

Argument	Explanation
`?`	Probes the system for valid SSA controller devices and prints them to screen (see "Probing for SSA Controller Devices.")
`dev=`*device_name*	Specifies the physical path name of the SSA controller card to be tested. This argument *must* be included when running `plntest` from the command line, unless the `?` argument is used.
`x`	Probes the specified SSA controller card for the single and grouped disks attached to the controller card, and prints their logical names to the screen. Note: The **`dev=`***device_name* must be specified for this option to work.

> **Note –** 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

Probing for SSA Controller Devices

Unlike most other hardware devices, the SSA controller card does not have a logical device name (one you would find in the `/dev` directory). Therefore, the SSA controller card is identified by its longer, physical device name.

> **Note –** The physical device name of the SSA controller card cannot be used to run `plntest`.

When running `plntest` from the command line, the physical device name of the SSA controller card cannot be used, so a logical name must be specified. ANSI standards require commas as delimiters between items. Since the physical name of the SSA controller contains embedded commas, if you use a physical name that contains commas as a command line option, `plntest` misinterprets the option.

There are two ways that you can create a logical name:

- Run the SunVTS kernel (vtsk), which automatically creates a logical name entry for the SSA controller under the `/dev` directory, such as:

```
# /dev/ssaXX , where XX represents the decimal number of the
controller
```

Use this name as the parameter for the dev= option of `plntest`.

- Manually make a soft link from the actual physical device name to a logical name of your choice (under `/dev`). Use this name as the parameter for the dev= option of `plntest`, as shown in the following example:

```
machine# ./plntest "?"

1: /devices/io-unit@f,e3200000/sbi@0,0/SUNW,soc@1,0/
SUNW,pln@0c0d,0e0f0102:ctlr
2: /devices/io-unit@f,e0200000/sbi@0,0/SUNW,soc@3,0/
SUNW,pln@0c0d,0e0f0102:ctlr

machine# ln -s \ /devices/io-unit@f,e3200000/sbi@0,0/SUNW,soc@1,0/
SUNW,pln@0c0d,0e0f0102:ctlr \
/dev/ssa1
machine# /opt/SUNWvts/bin/plntest dev=/dev/ssa1
```

plntest Error Messages

TABLE 24-3 plntest Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6010	*device_name text1* during *text2*. *text1* indicates SCSI sense information for the SCSI command *text2*		
6011	*device_name* SCSI status returned is *text*, Command issued was *text*		
6012	*device_name*: NVRAM Battery is low or has failed	Suspect battery module.	Fast writes will be disabled. Consult your authorized Sun service provider.
8000	*device_name* Failed to open()(*error_message*	Suspect fiber optic cable.	Consult your authorized Sun service provider.
		Suspect SSA controller board.	
		Suspect SOC host adapter.	

TABLE 24-3 `plntest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8010	*device_name* *text1* during *text2*. *text1* indicates SCSI sense information for the SCSI command *text2*		
8011	*device_name* SCSI status returned is *text*, Command issued was *text*		
8012	*device_name* Fans have failed	Fan failure.	Consult your authorized Sun service provider.

Physical Memory Test (`pmemtest`)

The `pmemtest` checks the physical memory of the system. The `pmemtest` locates parity errors, hard and soft error correction code (ECC) errors, memory read errors, and addressing problems. The pseudo driver `mem(7)` is used to read the physical memory.

This test reads through all available physical memory. It does not write to any physical memory location.

`pmemtest` Options
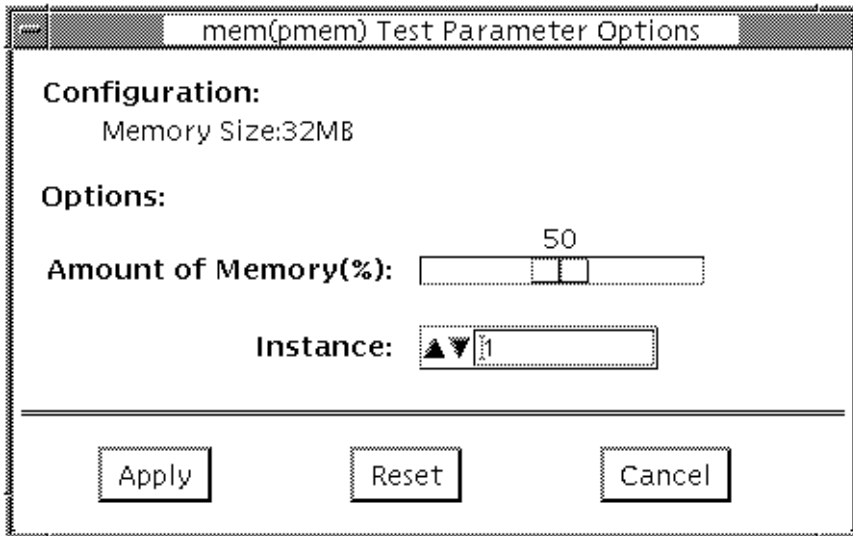
FIGURE 25-1 `pmemtest` Option Menu

Refer to TABLE 25-1 for test parameter descriptions.
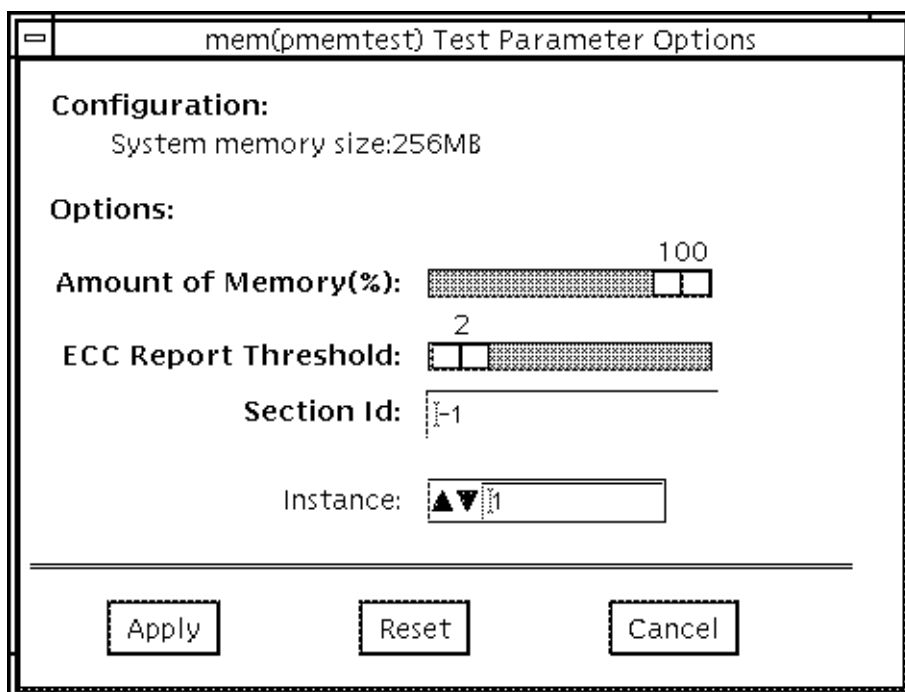
FIGURE 25-2 `pmemtest` UltraSPARC Server Option Menu

Refer to TABLE 25-1 for test parameter descriptions.

TABLE 25-1   `pmemtest` Options

`pmemtest` **Options**	**Description**
Configuration	Shows the total amount of physical memory, rounded up to the nearest megabyte, probed by the SunVTs kernel.
Amount of Memory	Specifies the percent of the physical memory to be tested.

TABLE 25-1 pmemtest Options

pmemtest **Options**	**Description**
ECC Report Threshold	Determines how many correctable ECC errors occur before they are reported as an error causing pmemtest to report a failure. A value of zero results in no report of any correctable ECC errors. The default is 2. This option is only available on UltraSPARC systems.
Section ID	When set to -1, pmemtest will test one memory section in each pass, automatically testing each subsequent memory section as testing progresses. When set to a number other than -1, only the section specified will be tested.A section is defined by the pass and instance number settings. This option is only available on UltraSPARC systems.
Instance	The number of copies of pmemtest to run simultaneously on the memory you are testing.

`pmemtest` Test Modes

TABLE 25-2   `pmemtest` Test Modes

Test Mode	Description
Connection Test	In this mode, 1 percent of the memory is read; `pmemtest` also informs the user how much physical memory is available. For sun4m, sun4u, and UltraSPARC servers, the test reports the ECC errors that have occurred since it was last invoked. The test reports ECC errors for a particular CPU or memory board when physical mapping is selected, otherwise it provides the SIMM number of the ECC memory error.
Functional Test	In Functional test, the amount of memory to be read can vary; the default is 100 percent. For UltraSPARC servers, this test mode reports all ECC errors that have occurred since the system was brought online
Functional Test mode through SyMON	When `pmemtest` is invoked through SyMON, the following Functional Test description applies: A variable amount of memory is read; the default is 100 percent. For UltraSPARC servers, the ECC error reporting is the same as for the Connection test.

`pmemtest` Command Line Syntax

/opt/SUNWvts/bin/pmemtest *standard_arguments*
-o size=*[1-100]***,dev=***device_name***,threshold=***report_threshold*
,bdinfo=*number***,section=***section_id*

TABLE 25-3   `pmemtest` Command Line Syntax

Argument	Explanation
size=*[1-100]*	Specifies the percentage of memory to be tested.
dev=*device_name*	Specifies the device to test such as `mem`.

TABLE 25-3 `pmemtest` Command Line Syntax

Argument	Explanation
threshold=*report_ threshold*	Determines how many correctable ECC errors occur before they are reported as an error causing `pmemtest` to report a failure. A value of zero results in no report of any correctable ECC errors. The default is 2. This option is only available on UltraSPARC systems.
bdinfo=*number*	For UltraSPARC servers, indicates board number information. For example, if board 0 and board 5 have memory and you want the test to read the memory on both boards, then bdinfo=33 (2**5+2**0).
section=*section_id*	When set to -1, `pmemtest` will test one memory section in each pass, automatically testing each subsequent memory section as testing progresses. When set to a number other than -1, only the section specified will be tested.A section is defined by the pass and instance number settings. This option is only available on UltraSPARC systems.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`pmemtest` Error Messages

TABLE 25-4 `pmemtest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6003	*number* intermittent errors on *text*. *text* would indicate the SIMM locations		
6004	Symbol *text* not found in kernel image. *text* will indicate the name of the symbol not found in the kernel image		
6005	*number* persistent errors on *text* SIMM(s). *text* would indicate the SIMM locations	SIMM(s), check the SIMM location	Call your Sun service representative.
8000	kvm_open() failed: *error_message*		
8001	sysinfo(SI_PLATFORM) failed: *error_message*		
8002	Symbol *text* not found in kernel image. *text* will indicate the name of the symbol not found in the kernel image		
8003	Couldn't open file /dev/mem: *error_message*		
8004	Unable to read pointer to memlist structure		
8005	Unable to read "size" and "address" in memlist		
8006	llseek() to address *address* failed (*error_message*)		
8007	read() at address *address* failed (*error_message*)		
8008	sysconf(_SC_NPROCESSORS_ONLN) failed: *error_message*		

TABLE 25-4 `pmemtest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8009	`llseek to address` *address* `failed (`*error_message*`)`		
8010	`llseek to address` *address* [*text*] `failed (`*error_message*`)`	This message is seen on UltraSPARC servers only. The *text* indicates to which board and bank the physical address maps.	
8011	`read() at address` *address* [*text*] `failed (`*error_message*`)`	This message is seen on UltraSPARC servers only. The *text* indicates to which board and bank the physical address maps.	

Prestoserve Test (`pstest`)

Prestoserve™ is an NFS accelerator. It reduces the frequency of disk I/O access by caching the written data blocks in nonvolatile memory. Prestoserve then flushes the cached data to disk asynchronously, as necessary.

`pstest` verifies the Prestoserve accelerator's functionality with the following three checks:

■ Board Battery Check

To ensure proper battery power level, the test runs this check before running the other two checks. If it finds a bad battery, it exits `pstest` immediately with a fatal error message.

■ Board Memory Check

This check maps the entire board memory to a process address space and locks the board to prevent multiple accesses. The test then travels through the mapped address spaces sequentially, doing a character, short, and long comparison on each space. The `pstest` runs this check twice.

■ Board Performance and File I/O Access Check

This check exercises only synchronous read/write access. `pstest` writes data equal to the amount of on-board memory to the memory cache and reads the data back for comparison. The time taken to write that data is measured twice: the first time with the Prestoserve accelerator disabled, the second time with Prestoserve enabled. The first value is divided by the second to get the performance ratio. If the ration is less than 1.5 on three passes of the test, the Prestoserve board may or may not have a problem.

Since the Prestoserve accelerator speeds up the `/opt` partition during testing and this partition may be mounted remotely, there may or may not be a problem with the Prestoserve board itself, a network performance problem could be the problem. In either case, a warning message is displayed if the performance ratio is less than 1:5.

> ⚠ **Caution –** To ensure consistent results, run as many concurrent tests as possible when a Prestoserve product has been installed. Tests are selected from the Option menu. The default is 2; the maximum is 10. However, do not run `pstest` and `kmem` tests at the same time. Running these tests together causes SunVTS to report erroneous errors.

`pstest` Options
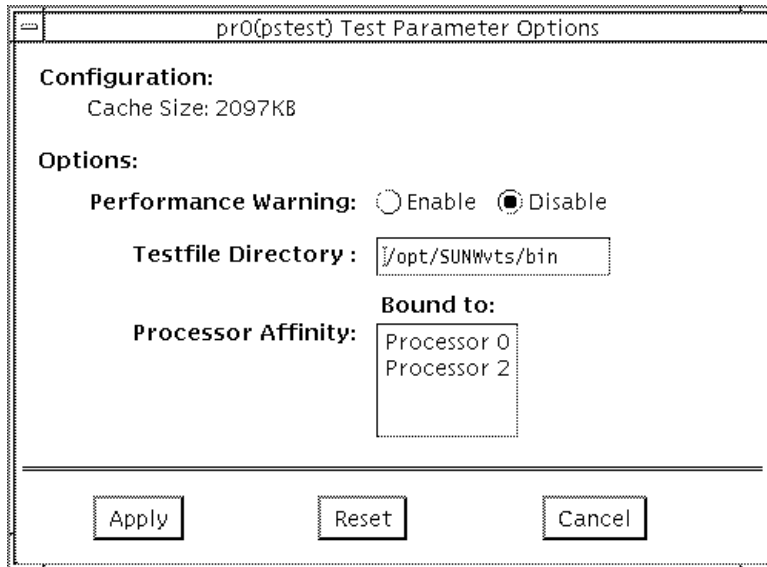
FIGURE 26-1 `pstest` Option Menu

`pstest` Test Modes

TABLE 26-1 `pstest` Test Modes

Test Mode	Description
Functional Test	The test runs the Board Battery Check subtest, Board Memory Check subtest, and Board Performance and File I/O Access Check subtest.

`pstest` Command Line Syntax

/opt/SUNWvts/bin/pstest *standard_argument* **-o e,l,f=***testfile_directory*

TABLE 26-2 `pstest` Command Line Syntax

Argument	Explanation
e	Enables performance warning, which displays a warning message if the performance ratio is less than 1:5.
l	Enables the long memory test.
f=*testfile_directory*	Specifies the directory of a test file.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`pstest` Error Messages

TABLE 26-3   `pstest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6001	Checking prstatus failed:		
6002	Failed to turn prestoserve DOWN:		
6003	File I/O test failed with low performance ratio: *number*		Check that Prestoserve is configured for /tmp.
6004	byte compare: location = *address*, observed = *number*, expected = *number*		
6005	word compare: location = *address*, observed = *number*, expected =*number*		
6006	(data type) long compare: location = *address*, observed = *number*, expected = *number*		
6007	lseek() failed:		
6008	NVRAM write failed:		
6009	NVRAM read failed:		
6010	Failed to get status of Prestoserve failed:		
6011	Failed to reinitialize Prestoserve:		
6012	Failed to turn prestoserve UP:		
6013	Failed to munmap prestoserve:		
6014	File write failed:		
6015	File read failed:		
6016	Data compare failed		
6017	Failed to restore prestoserve state: %s		

TABLE 26-3 `pstest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6018	Prestoserve test is applicable in Functional test only		
8001	Failed to open prestoserve board:	Prestoserve hardware not installed.	
		NVSIMM/NVRAM SBus board not seated correctly.	
		Prestoserve software not installed.	
		Not in superuser mode.	
8002	Prestoserve is in the ERROR state	Errors occurred on a disk drive.	
8003	Some batteries are not good or not enabled		
8004	memory_check (pass *pass_id*) failed		
8005	fileio_check (pass *pass_id*) failed		
8006	File I/O test failed (Prestoserve down)		
8007	File I/O test failed (Prestoserve up)		
8008	Open %s also failed: %s. Please enter test file directory		
8009	PRESTOSERVE TESTS FAILED!		

Remote System Control (`rsctest`)

The `rsctest` exercises the Remote System Control (RSC) feature on the Sun Enterprise 250.

The RSC provides secure remote access for system monitoring, firmware updates and failure recovery. The RSC hardware consists of the controller, flash, SEEPROM, 10MB Ethernet port and an external console serial port. The RSC communicates with the host through two internal serial lines, the I2C bus, and reset lines.

The `rsctest` consists of four subtests which test and report on the Ethernet, flash checksum, SEEPROM checksum, and the internal and external serial ports. The subtests call test modlets that are written in the native Real Time Operating System (RTOS) which resides in the RSC firmware. The `rsctest` subtests execute the test modlets, passes parameters, and retrieves results from the RSC using a test protocol on the host to RSC internal serial lines. The `rsctest` is not scalable.

rsctest Options

```
              rsc(rsctest) Test Parameter Options

Configuration:
    Remote System Control

Options:
          Enet_Test: ( ) Enable   ( ) Disable

  Data_Pattern_Type: [✓] Seq
                     [✓] Rand

        Packet_Size: [▲▼] 200

        Num_Packets: [▲▼] 50

        Target_Host: [ ]

          ETest_Type: [✓] Internal
                      [✓] External
                      [✓] Ping

         Flash_Test: ( ) Enable   ( ) Disable
       SEEPROM_Test: ( ) Enable   ( ) Disable
        Serial_Test: ( ) Enable   ( ) Disable

          Data_Size: [▲▼] 200

      Loopback_Type: [✓] Internal
                     [✓] External

  Data_Pattern_Type: [✓] Seq
                     [✓] Rand

          STest_Type: [✓] u_u
                      [✓] c_c
                      [✓] d_d

                      Bound to:
   Processor Affinity: Processor 0
                       Processor 1


       [ Apply ]      [ Reset ]      [ Cancel ]
```

FIGURE 27-1 `rsctest` Options Menu

TABLE 27-1 `rsctest` Options

`rsctest` **Options**	**Description**
Enet Test	Used to Enable or Disable RSC Ethernet testing.
Data Pattern Type	Select Sequential, Random, or both types of data patterns.
Packet Size	Define the size of each data packet to be sent for all tests.
Num Packets	Specify the number of data packets to send in one test loop.
Target Host	Specify the IP address of a host to use for the ping test.
Enet Test Type	Select any or all Internal, External, or ping test.
Flash Test	Used to Enable or Disable the flash checksum test.
seeprom Test	Used to Enable or Disable the SEEPROM checksum test.
Serial Test	Used to Enable or Disable the RSC serial test.
Data Size	Used to define the data size to be sent.
Loopback Type	Select Internal, External or both. External requires a loopback plug.
Data Pattern Type	Select Sequential, Random, or both types of data patterns.
Test Type	Select serial ports to be tested, u to u, c to c, or d to d.

`rsctest` Test Modes

`rsctest` supports Connection and Functional tests as described in the table below.

TABLE 27-2 `rsctest` Test Modes

Test Mode	Description
Connection Test	Reports the status of the RSC.
Functional Test	Tests the RSC's Ethernet, flash, SEEPROM, and serial devices. All tests use internal modes as defaults. The `rsctest` will not run the serial test on ttyc if the console has been redirected to the RSC. The ttyu tests will not run if there is an open login on ttyu.
	If you invoke SunVTS through SyMON, the following Functional Test description applies: `rsctest` uses the same functionality as connection test (above).

`rsctest` Command Line Syntax

`/opt/`**`SUNWvts/bin/rsctest`** *standard_arguments*
`-o enet=`*E/D***`,epattype=`***seq+rand***`,esize=`***packet_size***`,`**
`epkts=`*number_packets***`,target=`***IP_address***`,etest=`***I+E+P***`,flash=`***E/D***`,seeprom=`***E/***
D`,`
`serial=`*E/D***`,sdatsize=`***data_size***`,slb=`***I+E***`,spattype=`***seq+rand***`,`**
`stest=`*u_u+c_c+d_d***

TABLE 27-3 `rsctest` Command Line Syntax

Argument	Explanation
`enet=`*enable|disable*	Enable or disables RSC Ethernet test.
`epattype=`*seq+rand*	Predefined pattern options used for enet test.
`esize=`*packet_size*	The data size for each packet in the enet test.
`epkts=`*number_packets*	The number of packets to send for enet test.
`target=`*IP_address*	IP Address of target system for enet ping test.
`etest=`*Internal+External+Ping*	Test modes for RSC enet test.
`flash=`*enable|disable*	Enables or disables RSC flash checksum test.
`seeprom=`*enable|disable*	Enables or disables RSC SEEPROM checksum test.
`serial=`*enable|disable*	Enables or disables RSC serial test.
`sdatsize=`*data_size*	Data size for the rsc serial tests.
`slb=`*Internal+External*	Loopback type. External N/A on ports c&d.
`spattype=`*seq+rand*	Predefined pattern options used for RSC serial test.
`stest=`*u_u+c_c+d_d*	Defines port and configuration to use for rsc serial test.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNVvts/bin/`
`sparcv9/`*testname*. If a test is not present in this directory, then it may only be
available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on
page 5.

`rsctest` Error Messages

TABLE 27-4 `rsctest` Error Messages

Error Message	Probable Cause(s)	Recommended Action
8000 `Unable to open driver` *driver name*`:` *sys error msg*	Device name is wrong or driver is not loaded.	Check the device name (if entered) or load driver.
	Device busy.	Kill all other processes accessing driver.
8001 `Unable to allocate memory using malloc:` *sys err msg*	System software error.	Consult `/var/adm/messages` file for more information.
8002 `Illegal options passed to rsctest` *option*	User error.	Check usage by typing **`rsctest -u`**.
8003 `Error encountered during call to` *function name*		
8004 `Error encountered during call to rsc_send(). Error Code:` *sys error msg*	RSC Hardware is not responding.	Reset RSC.
8005 `Error encountered during call to rsc_recv(). Error Code:` *sys error msg*	RSC RTOS test timed out.	Reset RSC.
8006 `Failing status received. Status:`*status*	RSC's RTOS test returned a failing status.	
8007 `Invalid message type received from rsc_recv(). Msg_type:` msg type	RSC returned a unexpected message type.	RSC RTOS/protocol error.
8008 `Error encountered during call to rsc_start(). Error Code:` *sys error msg*	Unable to initialize RSC test protocol.	RSC RTOS/protocol error. Reset RSC.

TABLE 27-4 `rsctest` Error Messages

Error Message	Probable Cause(s)	Recommended Action
8009 `RSC's` *Internal|External* `enettest failed for pattern type=` *patt type*	Internal: Bad hardware Improper RSC enet configuration.	Check RSC configuration. Check 10 MB hub and cable. Replace mother board.
	External: Bad hardware, Improper RSC enet configuration, Missing 10 MB hub and cable.	
8010 `RSC's PING test failed. Pattern type =` *patt type* `Target host=` *target IP*.	Invalid RSC enet configuration.	Check RSC's enet configuration.
	Missing/damaged cable Invalid target IP Hardware failure.	Check cable connection. Check target IP address option. Replace mother board.
8011 `RSC's flash checksum test failed. Status:` *status*	Corrupted flash firmware. Error on I2C bus. Bad flash hardware.	Reflash firmware. Check I2C bus using env2test. Replace Hardware.
8013 *term setting* `failed for TERMIO.:` *sys error msg*	RSC protocol error.	
	System software error.	Consult the `/var/adm/messages` file for more information.
8014 `Read failed.:` *sys error msg*	System software error.	Consult the `/var/adm/messages` file for more information.
8015 `Data read does not match data sent.`	Bad hardware or System software error.	
8016 `Data read exceeds data sent.`	Bad hardware or System software error.	
8017 `Select timedout for read.:`*sys error msg*	System software error. Excessive load on RSC/ system causing lost data.	Reduce RSC and System load. Consult the `/var/adm/messages` file for more information.
8018 `Write failed for` *num bytes* `bytes of data.:`*sys error msg*	System software error.	Consult the `/var/adm/messages` file for more information.
8019 `Select timedout for write.:` *sys error msg*	System software error.	Consult the `/var/adm/messages` file for more information.
8020 `RSC Serial test failed. Test Name=` *testname* `Loopback=` *Loopback type* `Pattern_Type=` *patt type* `Baud=` *baud rate*.	Bad hardware or System software error.	

TABLE 27-4 `rsctest` Error Messages

Error Message	Probable Cause(s)	Recommended Action
8021 `OPEN of `*`port`*` failed.:`*`sys error msg`*`.`	System software error.	Consult the `/var/adm/messages` file for more information.
8022 `RSC Status shows that the last run of POST failed.`	Bad hardware or System Firmware error.	
8023 `Unable to access lib `*`rsc lib`*`.: `*`sys error msg`*`.`	System software error.	Consult the `/var/adm/messages` file for more information.
8024 `Unable to close lib.: `*`sys error msg`*`.`	System software error.	Consult the `/var/adm/messages` file for more information.
8025 `Call to rsc_init() failed. Error Code: `*`rsc error msg`*	System software error.	Consult the `/var/adm/messages` file for more information.
8026 `Unable to access function `*`function name`*` in lib.: `*`sys error msg`*`.`	System software error.	Consult the `/var/adm/messages` file for more information.
8027 `RSC's SEEPROM checksum test failed. Status: `*`status`*`.`	Corrupted Firmware or problem with I2C bus or system software error.	
8028 `Select failed for [`*`Write`*`/`*`Read`*`].`	Hung RSC. RSC data protocol error.	Consult `/var/adm/messages` file for more information.
8029 `Sending OFF to Serial test failed. Test Name= `*`test_name`*` Loopback= `*`loopback_type`*` pattern_type=`*`pattern_type`*` Baud=`*`baud_rate`*	Hung RSC. RSC data protocol error.	Reset RSC.
8031 `Communication to RSC failed.`	RSC Hung	Reset RSC.
8032 `Unable to get RSC's console port from library.`	RSC lib not installed. System software error.	Check for RSC lib on host.
8033 `Unable to get RSC's control port from library.`	RSC lib not installed. System software error.	Check for RSC lib on host.
8034 `Unable to negotiate loopback mode on RSC for serial loopback.`	Excessive RSC load. RSC data protocol error.	Reduce RSC and system load. Reset RSC.

SunVideo Test (`rtvctest`)

`rtvctest` verifies the functionality of the SunVideo™ SBus card. SunVideo technology captures and compresses video input in realtime, making it possible to have realtime video conferencing over standard Ethernet networks.

`rtvctest` is available as a `customtest`. To display the test on the SunVTS user interface, you must:

1. Edit the `.customtest_OtherDevices` file in the SunVTS bin directory.

2. Uncomment the line starting with the label SunVideo.

The `rtvctest` was originally written for Sundiag™ and the messages from the test are displayed on the SunVTS console within a SunVTS INFO message wrapper.

The `rtvctest` is available under the `OtherDevices` group, and is only available in the Functional Test mode.

`rtvctest` Subtests

`rtvctest` is divided into four subtests: PROMCheck, Memory, Jalapeno, and CL4000. The PROMCheck subtest verifies the SunVideo card's programmable read-only memory. The memory test verifies all of the memory on the card, including the 2 Mbytes of memory on the CL4000 compression engine and the memory on the Jalapeno application-specific integrated circuit (ASIC). The Jalapeno subtest verifies the interface logic between the SBus, A/D conversion chips, and the CL4000 compression engine. The CL400 subtest verifies that the compression engine ASIC is able to compress digitized video data from the A/D chips and send this data to the SBus through the Jalapeno ASIC.

The `rtvctest` is composed of 49 verification test modules. The table below lists these modules and their associated test sequence numbers.

TABLE 28-1 `rtvctest` Verification Modules

SunVideo Verification Module Name	Test Sequence Number
RTVC SUNDIAG Start	0
RTVC Checksum	1
RTVC Jalapeno SMEM	2
RTVC CL4000 DMEM	3
RTVC Jalapeno SBus Interrupt Mask	4
RTVC CL4000 Interrupt Mask	5
RTVC DVMA Control Register	6
RTVC DVMA Transfer Size Counter	7
RTVC DVMA Memory Address Counter	8
RTVC DVMA Virtual Memory Address Counter	9
RTVC DVMA Slave SBus Rerun Register	10
RTVC IIC Control Register	11
RTVC IIC Data Register	12
RTVC Video DMA Control Register	13
RTVC Video DMA Transfer Size Counter	14
RTVC Video DMA Memory Address Counter	15
RTVC User Interrupt 0	16
RTVC User Interrupt 1	17
RTVC User Interrupt 2	18
RTVC User Interrupt 3	19
RTVC Video Control and Status Register	20
RTVC Video Control Field Line Interrupt 1	21
RTVC Video Control Field Line Interrupt 2	22
RTVC Video Scan Line Mask Registers	23
RTVC Video Input Format Type	25
RTVC Video Horizontal Lock	26
RTVC Video Even Odd Field	27
RTVC CL4000 Host Control	28

TABLE 28-1 `rtvctest` Verification Modules (Continued)

SunVideo Verification Module Name	Test Sequence Number
RTVC CL4000 Host Lock	29
RTVC CL4000 Video Port A Control	30
RTVC CL4000 Video Port B Control	31
RTVC CL4000 Video Port A FIFO	32
RTVC CL4000 Video Port B FIFO	33
RTVC CL4000 Address Memory Registers	34
RTVC CL4000 Instruction Memory Access Registers	35
RTVC Time Stamp Register	36
RTVC CL4000 Register Memory	37
RTVC CL4000 Scratch Memory	38
RTVC CL4000 DMA Mode	39
RTVC CL4000 Motion Estimation Registers	40
RTVC CL4000 PSW	41
RTVC CL4000 Variable Length Coder Registers	42
RTVC CL4000 Channel Memory Registers	43
RTVC CL4000 CPU Control	44
RTVC CL4000 Multiply Control	45
RTVC CL4000 DMA Interrupt Control	46
RTVC CL4000 Block Transfer Mode	47
RTVC CL4000 Accumulator MSB	48
RTVC CL4000 JPC Field	49
RTVC SUNDIAG Finish	50

`rtvctest` Options
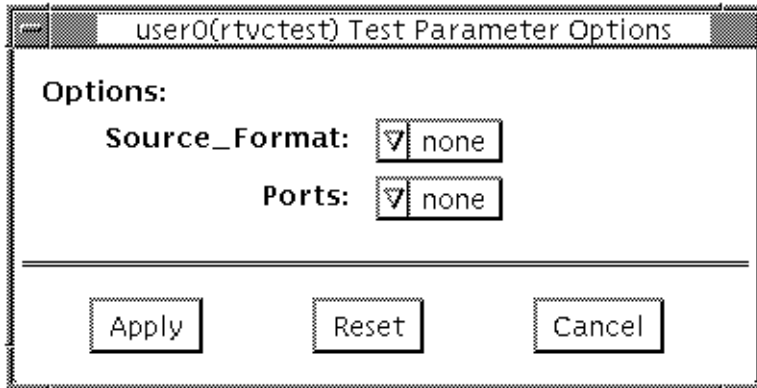
FIGURE 28-1 `rtvctest Option Window`

You may test the SunVideo card without any video device (camera, video disc player, or VCR) connected to an input port. However, if you connect a video device to the card, then you must state the format of the video source and the input port to which the device is connected.

Note – If you do not state both the video source and the input port, the `rtvctest` prints an error and terminates testing.

Source Format

Select the format of the video source used for testing. You may select from the sources shown in TABLE 28-2.:

TABLE 28-2 `rtvctest` Source Format Options

Video Source	Definition
None	No video source.
NTSC	National Television System Committee (NTSC) video source, which is the video standard in the United States and Japan.
PAL	Phase alternate line video source, which is the video standard in Europe.

Ports

If you have selected a video source for testing, then you must state which input port you have used to connect the source to the SunVideo card. You may select from the ports shown in TABLE 28-3:

TABLE 28-3 `rtvctest` Port Options

Input Port	Definition
None	No input port used.
CV1	Composite video input Port 1 (RCA type input).
CV2	Composite video input Port 2 (RCA type input).
SVHS	Super VHS input port.

`rtvctest` Command Line Syntax

/opt/SUNWvts/bin/rtvctest [fmt=none|ntsc|pal**]**
[port=none|cv1|cv2|svhs**] dev=**rtvc_device

TABLE 28-4 `rtvctest` Command Line Syntax

Arguments	Description
fmt	Select the format of the video source used for testing. Select either **ntsc** (video standard in the United States and Japan) or **pal** (video standard in Europe).
port	If you have selected a video source for testing, then you must state which input port you have used to connect the source to the SunVideo card. You may select from the following ports: • **cv1**Composite video input port 1 (RCA type) • **cv2**Composite video input port 2 (RCA type) • **svhs**Super VHS port
dev	You are required to state the SunVideo device when running the `rtvctest` from the command line. Replace *rtvc_device* with **rtvc**n, where *n* is the device number (from 0 to 31) of the SunVideo card being tested.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`rtvctest` Error Messages

TABLE 28-5 `rtvctest` Error Messages

Error Message	Description
`SunVideo 10`	*message* Where: *message* is one of the following: `Missing REQUIRED argument D=rtvc[0..31]`

`SunVideo 20`	*message* Error # *number*
	Where:
	message is one of the following:
	`ioctl RTVC_CMD_RESET fault`
	`ioctl RTVC_CMD_SET_VIDEO fault`
	`ioctl RTVC_CMD_GET_VIDEO fault`
	`open /dev/rtvc fault`
	`close /dev/rtvc fault`
	`open /dev/rtvcctl fault`
	`close /dev/rtvcctl fault`
	`Unknown Jalapeno Version`
	number is one of the following:
	`ioctl error`
	`module version`

`SunVideo 30`	*message name* Error # *number*
	Where:
	message is one of the following:
	`memory map fault for`
	`memory unmap fault for`
	selected /dev/rtvc and /dev/rtvcc1 not available.
	name is one of the following:
	`prom`
	`sram`
	`buses`
	`reset`
	`dmem`
	rtvc
	number is the error number

TABLE 28-5 `rtvctest` Error Messages (Continued)

Error Message	Description
SunVideo 40	RTVC Fault Detected via module *name* Physical Address *value* Expected Value *value* Actual Value *value* RTVC Sundiag Module number *number* Location *location* Message: *message(s)* Where: *name* is the name of the module under test *value* are the values associated with physical address, expected value, actual value *number* is the Sundiag Module under test. *location* is the U location of the faulty component. *message(s)* are any additional messages.
SunVideo 41	RTVC Fault Detected via module *name* Where: *name* is the name of the module under test.

SunVideo 42	*message value1 Actual value2*
	Where:
	message is one of the following:
	`Video Format Expected`
	`Video Port Expected`
	value1 is the expected value.
	value2 is the actual value.

SunVideo 50	*message number*
	Where:
	message is one of the following:
	`Unknown RTVC CL4000 test module`
	`Unknown RTVC Jalapeno test module`
	`Unknown RTVC test module`
	`Unknown RTVC memory test module`
	`Unknown Start Bit`
	number is the module number or start bit location.

Serial Asynchronous Interface (PCI) Test (`saiptest`)

`saiptest` checks the functionality of the Serial Asynchronous Interface card through its device driver.

`saiptest` Hardware Requirements

Before running the SunVTS diagnostics software, make sure you install the device driver and the cards to be tested. Also, you should reboot your system with the `boot -r` command to reconfigure the system and allow the SunVTS kernel to recognize the new driver.

Note – You must run the `saiptest` in intervention mode.

The following minimum hardware configuration is required to successfully run the Internal test:

- PCI-based SPARC desktop system with a PCI slot
- Serial Asynchronous Interface card, installed in one of the PCI slots

The following hardware is also required to run other SunVTS Serial Asynchronous Interface tests:

- Serial Asynchronous Interface Patch Panel (Part No. 7600-0027)
- 25-pin serial loopback plugs (Part No. 540-1558)
- RS-232 serial cables (Part No. 530-1685)
- TTY terminal

`saiptest` Options
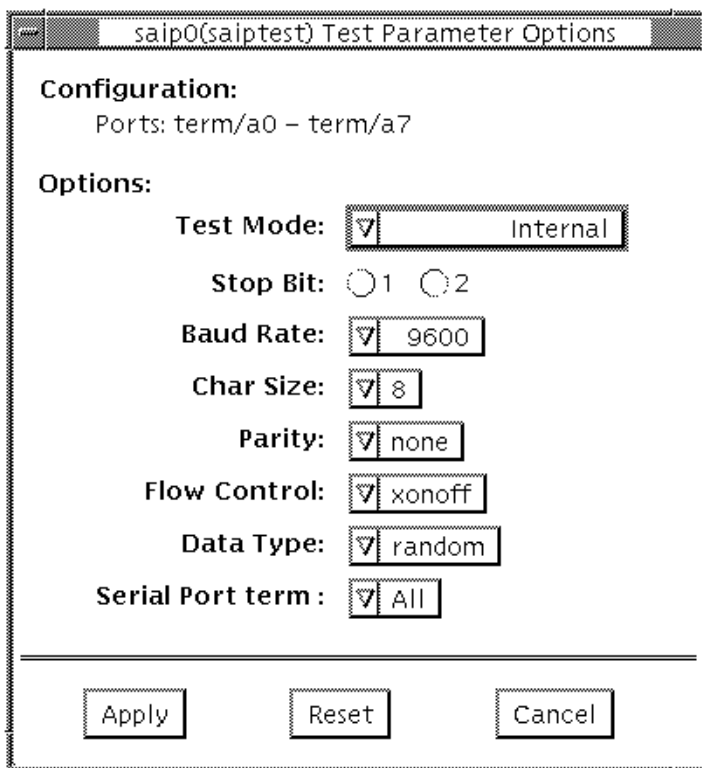
FIGURE 29-1 `saiptest` Options Menu

The Configuration section of the option menu displays the asynchronous serial ports available for the Serial Asynchronous Interface board. Table 4-1 shows the available ports.

TABLE 29-1 `saiptest` Asynchronous Serial Ports

Board Number	Board Device	Serial Ports
0	saip0	term/a000-a007
1	saip1	term/b000-b007
2	saip2	term/c000-c007

TABLE 29-1 `saiptest` Asynchronous Serial Ports (Continued)

Board Number	Board Device	Serial Ports
3	saip3	term/d000-d007
4	saip4	term/e000-e007
5	saip5	term/f000-f007
6	saip6	term/g000-g007
7	saip7	term/h000-h007
8	saip8	term/i000-i007
9	saip9	term/j000-j007
10	saip10	term/k000-k007
11	saip11	term/l000-l007

TABLE 29-2 `saiptest` Options

`saiptest` **Option**	**Description**
Internal Test	Performs internal loopback testing on the Serial Asynchronous Interface card(s) installed in PCI slots. You do not need to attach anything to the card(s) to perform this test.
25-pin Loopback	Provides full-duplex transmission and full-modem loopback testing of the serial port selected in the Serial Port section of the option menu. You must attach the 25-pin loopback plug to the serial port on the Serial Asynchronous Interface Patch Panel that is being tested. This test cannot be run concurrently with the Echo-TTY option enabled.
Echo-TTY	Checks the proper operation of the serial port selected in the Serial Port selection of the option menu by echoing characters typed on a TTY terminal keyboard to the TTY terminal screen. The characters you type on your TTY keyboard display on the TTY screen. NOTE: A TTY connection to the Serial Asynchronous Interface serial port requires corresponding character size se up. For example, if a TTY attachment is running with 8-bit character size, then the Char Size `saiptest` option should be set to 8- bits. If you do not type any characters within two minutes, this test times-out.
Baud Rate	Specifies the baud rate; choose 110, 300, 600, 1200, 2400, 4800, 9600, 19200, or 38400 baud. NOTE: The baud rate of 38400 can only be used if the Internal Test is disabled and you are testing one port at a time.
Char Size	Specifies the character length; choose 5, 6, 7, or 8 characters.

TABLE 29-2 `saiptest` Options (Continued)

`saiptest` **Option**	**Description**
Stop Bit	Specifies the number of stop bits; choose 1 or 2 bits.
Parity	Specifies the selectable parity; choose none, odd, or even.
Flow Control	Specifies the selectable flow control; choose XOnOff, rtscts, or both.
Data Type	Specifies the selectable data type pattern; choose 0x55555555 (0x55), 0xaaaaaaaa (0xaa), or random.
Serial Port	Specifies the serial port to be tested. The available ports are listed in the Configurations section at the top of the `saiptest` options menu.

`saiptest` Test Modes

`saiptest` is only available in the Functional Test mode.

`saiptest` Command Line Syntax

/opt/SUNWvts/bin/saiptest *standard_arguments* -o dev=*device_name*,
M=*test_mode*,B=baud_rate,Size=*character_size*,Stop=*#of_stop_bits*,
Parity=*parity*,F=*flow_control*,Data=*test_pattern*,sp=*serial_port*

TABLE 29-3   `saiptest` Command Line Syntax

Argument	Explanation
`dev=device_name`	Specifies the asynchronous serial ports of the PCI card slots tested. Since there is no default, you must type a device name–either a board(saip0-12) or an individual port (term/x000-term/x007, where x is a-l): •saip0 = the 8 asynchronous serial ports in the first card •saip1 = the 8 asynchronous serial ports in the second card •saip2 = the 8 asynchronous serial ports in the third card •saip3 = the 8 asynchronous serial ports in the fourth card •saip4 = the 8 asynchronous serial ports in the fifth card •saip5 = the 8 asynchronous serial ports in the sixth card •saip6 = the 8 asynchronous serial ports in the seventh card •saip7 = the 8 asynchronous serial ports in the eighth card •saip8 = the 8 asynchronous serial ports in the ninth card •saip9 = the 8 asynchronous serial ports in the tenth card •saip10 = the 8 asynchronous serial ports in the eleventh card •saip11 = the 8 asynchronous serial ports in the twelfth card or `/dev/term/x00`*m* Where *x* is a-l and m is 0 to 7 (any of the asynchronous serial ports in PCI card slots).
`M=test_mode`	Specifies Internal, 25_pin_loopback, or Echo_TTY test mode.
`B=baud_rate`	Sets the baud rate to 110, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400; the default is 9600.
`Stop=#of_stop_bits`	Toggles the number of stop bits between 1 or 2; the default is 1.
`Size=character_size`	Sets the character size as an integer between 5 and 8.
`Parity=parity`	Specifies the parity as none, odd, or even; the default is none.
F=flow_control	Specifies flow control as xonoff, rtscts, or both.
Data=test_pattern	Specifies test pattern as 0x55555555, 0xAAAAAAAA, or random.
sp=serial_port	Specifies the terminal and asynchronous serial port number, such as term/a003.

> **Note –** 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`saiptest` Error Messages

The `saiptest` error messages are generated when the SunVTS Serial Asynchronous Interface discovers errors. The error descriptions in the table below identify probable causes for the card or test failure, and identify the Field Replaceable Unit (FRU) and recommended action, if possible. The three FRUs under test are: the Serial Asynchronous Interface card, the 78-pin shielded cable, and the Patch panel

TABLE 29-4 `saiptest` Error Messages

Error Message		Probable Cause(s)	Recommended Action
6000	*error_message*		
6001	Expected = *value*, observed = *value*		
6002	Expected *value*:		
6003	Observed *value*:		
6004	Modem Loopback test failed on *device_name*		
6005	Off-line error on device *device_name*		
6007	Busy error on device *device_name*		
6008	Error on device *device_name*		
6009	Retry *number* exceeds limit *number* rc =s *value*		
6010	Expected *number* bytes, observed *number* bytes		
6011	Internal test failed on *device_name*		
6012	Data Loopback test failed on *device_name*		

TABLE 29-4 `saiptest` Error Messages (Continued)

Error Message	Probable Cause(s)	Recommended Action
6013 `Ioctl TIOCSSOFTCAR error on` *device_name*		
8000 `Must provide device name!`		
8001 `No SAI/P card found or device driver not installed`		
8002 `Ioctl TCSETS failed on` *device_name*		
8003 `Open error on` *device_name* `– device driver may not be installed properly`		
8004 `Device` *device_name* `is already opened`		
8005 `Ioctl SAIP_ENLOOP error in` *device_name*		
8006 `Ioctl SAIP_DISLOOP error in` *device_name*		
8011 `Ioctl TIOCMGET failed on` *device_name*		
8012 *device_name*`: Expected DSR set, observed clear`		
8013 *device_name*`: Expected DSR clear, observed set`		
8014 `Ioctl TIOCMSET failed on` *device_name*		
8015 `Ioctl TIOCMBIC failed on` *device_name*		
8016 `Read error on` *device_name*		
8017 `Write error on` *device_name*		
8020 `Timeout error on` *device_name*	No loopback plug.	Check the loopback plug.
8021 `Timeout error on` *device_name*	TTY terminal not connected.	Check the TTY terminal connected to the device.
8022 `Timeout error on` *device_name*		

.

Sun Enterprise Cluster 2.0 Network Hardware Test (`scitest`)

`scitest` verifies the functionality of the Sun Enterprise Cluster 2.0 by checking the networking hardware. For this test to be meaningful, the cluster must already be configured before the test is run. For details on how to configure the cluster, refer to *Sun Enterprise Cluster 2.0 Hardware Site Preparation, Planning, and Installation Guide*.

`scitest` reads the `/etc/sma.ip` file to determine the target nodes in the cluster. `scitest` mainly uses the Internet Control Message Protocol (ICMP) to test the connections between cluster nodes.

After finding the cluster nodes (targets), `scitest` performs the following tests:

- `Random test` sends out 256 packets with random data length and random data.
- `Incremental test` sends out packets with length from minimum to maximum packet size using incremental data.
- `Pattern test` sends 256 packets of maximum length, where each packet contains one test pattern and all byte patterns (0 to 0xFF) are used.

Note – `scitest` is a scalable test. The maximum number of instances is two per SCI card.

`scitest` Options

FIGURE 30-1 `scitest` Option Menu

TABLE 30-1 `scitest` Options

`scitest` **Options**	**Description**
Target Host	Not used.
Receive Timeout field	Specify a number between 0–600 seconds. The default is 120 seconds.
Number of Retries field	The number of retries before an error is flagged. Specify a number between 0–128.
Print Warning	Disabled by default. Choose Enable to see warning errors, such as `retry on timeout` errors.

`scitest` Test Modes

Connection, Functional and Online modes are supported by `scitest`. Different test schemes are performed on the network device based on the mode selected.

TABLE 30-2 `scitest` Test Modes

Test Mode	Description
Connection Test	`scitest` checks if the device is connected. It searches through all the network interfaces for the specified device name. `scitest` finds the device not connected the test fails, otherwise, it returns `device is connected`.
Functional Test	`scitest` performs all three tests (Random test, Incremental test, and Pattern test) sequentially. It allows you to specify an option in such a way that `scitest` performs a very stressful test.
Functional mode through SyMON	If you invoke SunVTS through SyMON, the following test description applies: `scitest` is aware that the user's application may be running, it tries to minimize the influence on the application while it is testing by only running the random test. Since the network device is shared and system resources are available, applications will not be heavily impacted.

`scitest` Command Line Syntax

/opt/SUNWvts/bin/scitest *standard_arguments* **-o
dev=**_interface_**,test=**_type_**,packets=**_n_**,pattern=**_hex_**,delay=**_seconds_**,
timeout=**_seconds_**,retry=**_n_**,warn**

TABLE 30-3 `scitest` Command Line Syntax

Argument	Explanation
dev=_interface_	Network interface name. The default value is **le0** for Ethernet networks.
test=_type_	The test type. Specify `random`, `increment`, or `pattern`. The default value is random+increment+pattern to run.
packets=_n_	Number of random/pattern packets. The default is 256.
pattern=_hex_	Specifies a data pattern in hexadecimal form. The default is all patterns from `0` to `0xff`.

TABLE 30-3 `scitest` Command Line Syntax

Argument	Explanation (Continued)
delay=*seconds*	Indicates the time between subtests in seconds. The default is 30 seconds.
timeout=*seconds*	Indicates the number of seconds to wait before a timeout. The default is 20 seconds.
retry=*n*	Indicates the number of test timeout retries. The default is three retries.
warn	When enabled, prints warning messages.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/ sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`scitest` Error Messages

TABLE 30-4   `scitest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
4001	`Unable to communicate from` *string*	SCI adapter failure or no cable connected.	Check cable connections. Replace the SCI adaptor.
4002	`Can not access /etc/ sma.ip file. "I WILL STOP TESTING SCI INTERFACES"`	`/etc/sma.ip` is missing, or the cluster is not configured, or this is an old cluster version.	Use `sm_config` to configure the cluster. Restore the files from a backup tape. Specify the cluster machines in the target host of test options before running the test.
6000	`RPC broadcast failure` *error_message*		
6001	`Unable to resolve` *host_name* `ethernet address`		
6002	`No ICMP echo reply from` *string*	SCI adaptor failure or no cable connected.	Check cable connections. Replace the SCI adaptor.
6003	`ICMP echo reply incorrect length from` *host_name*`, exp` *data* `obs` *data*		
6004	`ICMP echo reply length` *number* `from` *host_name*`, data mismatch at byte` *number*`, exp` *value* `obs` *value*		
6005	`Unable to find any test target`		
8000	`Unable to open` *device_name* *error_message*		
8001	`Bad ioctl` *name* *error_message*		
8002	`Unable to allocate` *number* `bytes of memory` *error_message*		
8003	`System call` *name* `returned` *value* *error_message*		
8004	`Unable to map network entry for` *host_name*		

TABLE 30-4 `scitest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
8006	Failed to allocate *number* bytes of swap space for configuration		
8007	dl_primitive response = *value*		
8008	dl_ok_ack: short response		
8009	dl_phys_addr_ack: short response		
8010	Invalid Argument *string*		
8011	Interface *device_name* not configured in system		
8012	SCITEST can not open/read /etc/sma.ip		

Environmental Sensing Card Test (`sentest`)

`sentest` checks the SCSI Environmental Sensing card (SEN) installed in the SPARCstorage RSM to monitor the enclosure environment. The SEN card monitors the enclosure's over-temperature condition, fan-failures, power-supply failures, and drive activity.

`sentest` verifies the SEN card by first setting each control function to a predetermined value, and then reading it back to verify if the value is correct.

`sentest` verifies the following control functions in the enclosure:

■ **Alarm** (enable/disable)–`sentest` toggles the alarm to the disable state, then to the enable state.

■ **Alarm time** (0-0xff seconds)–`sentest` sets the time (from 0 to 4095), then reads it back to verify the time setting.

■ **Drive fault LED** (DL0-DL6)–`sentest` toggles each LED to its OFF and ON states.

`sentest` is a nonscalable test. It cannot verify the control function settings correctly if another instance is changing the setting.

`sentest` Options
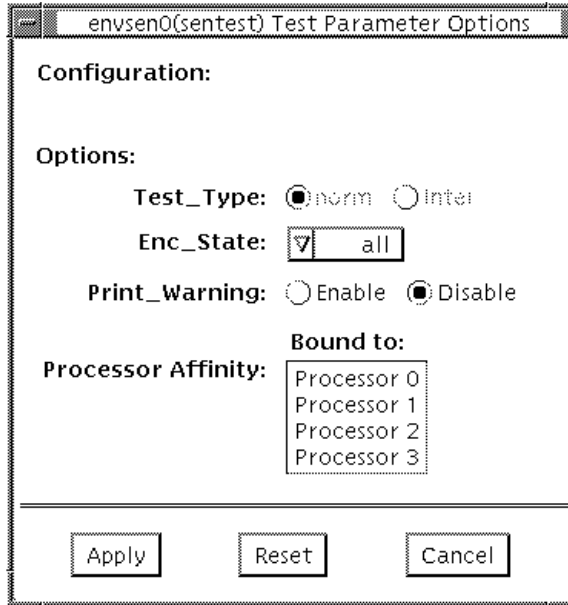
FIGURE 31-1 `sentest` Options Menu

TABLE 31-1 `sentest` Options

`sentest` **Option**	**Description**
Test Type	Specifies the test to be performed. `Norm` test type performs normal testing as alarm enable/disable, alarm time setting, and drive LED on/off testing; Interactive test type reports the current enclosure status.
Enc_state	Specifies which subsystem's status in the enclosure is reported. Default is ALL. This test is only used with the inter test type and in Functional test. The test options are: •Alarm enable/disable status •Drive present status •Drive LED status •Power modules status •Fan modules status •Over temperature, abs (abnormal, no immediate attention needed), chk (abnormal, immediate attention needed) status •All of the above

`sentest` Test Modes

`sentest` supports all three test modes. Each mode performs a different test scheme on the SEN card.

TABLE 31-2 `sentest` Test Modes

Test Mode	**Description**
Connection Test	Checks the device connection by opening the device. If the device does not open, the device is not connected.
Functional Test	Checks three components within the enclosure. It checks alarm enable/disable, alarm time setting, and the drive LEDs. It does not test the power on/off function (only functions whose values can be changed are tested).
Online Mode	Provides a means to run `sentest` through SyMON. In Online mode, the following Functional Test description applies: `sentest opens the device and reports the current enclosure status.`

`sentest` Command Line Syntax

/opt/SUNWvts/bin/sentest *standard arguments*
-o dev=*interface***,test=***type***,enc=***component*

TABLE 31-3   `sentest` Command Line Syntax

Argument	Explanation
dev=*interface*	SEN card device name; the default value is `ses0`.
test=*type*	Specifies the test type; select `Norm` for normal testing or `Inter` for interactive testing; the default value is `Norm`. Possible values are: `norm` and `inter`.
enc=*component*	Indicates which part of the enclosure status is reported; the default value is `ALL`. Possible values are: `enalm`, `dp`, `dl`, `pm`, `fan`, `ovt`, and `ALL`.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`sentest` Error Messages

TABLE 31-4   `sentest` Error Messages

Error Message	Probable Cause(s)	Recommended Action
6000 `Check alenb failed,` `exp=`*value*`, obs=`*value*		
6001 `Check alenb failed,` `exp=`*value*`, obs=`*value*		
6002 `Check alenb failed,` `exp=`*value*`, obs=`*value*		
6003 `Check altime failed, exp` `time=`*value*`, obs=`*value*		

TABLE 31-4 `sentest` Error Messages *(Continued)*

Error Message	Probable Cause(s)	Recommended Action
6004 Fan failed		
6005 Dual fan failed		
6006 Enclosure over temperature		
6007 Device failed, need immediate attention		
6008 Power module A failed		
6009 power module B failed		
8000 power module B failed		
8001 ioctl get state failed, errmsg=*message*		
8002 ioctl get state failed, errmsg=*message*		

Soc+ Host Adapter Card Test (`socaltest`)

`socaltest` aids the validation and fault isolation of the SOC+ host adapter card. In the case of a faulty card, the test tries to isolate the fault to the card, the GBIC module, or the DMA between the host adapter card and the host memory.

`socaltest` Options
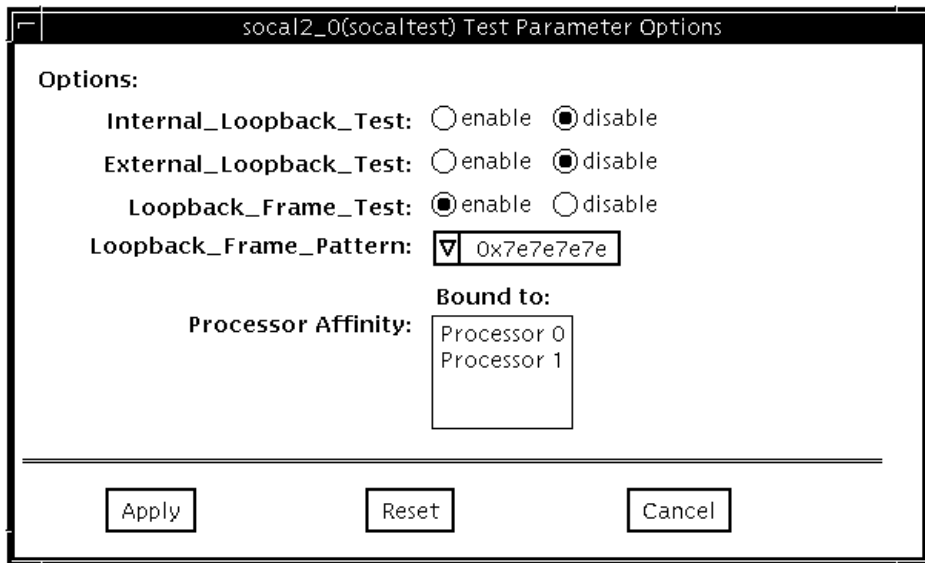
FIGURE 32-1 `socaltest` Options Menu

`socaltest` Options

TABLE 32-1   `socaltest` Options

`socaltest` **Options**	**Description**
Internal Loopback Test	Checks the host adapter card and the direct memory access (DMA) with the host system. This is accomplished as follows: • A frame is created in the host adapter local memory, sent out through the SOC+ transmitter and internally looped back to the SOC+ receiver. The received data is compared with the original data. • A frame is created in the host adapter local memory, sent out through the SOC+ transmitter and looped back through the SERDES (serialiser-deserialiser) chip on the host adapter card. The received data is compared with the original data. • A frame is created in the host main memory, transferred through the DMA to the host adapter transmitter, looped back within the SOC+ chip, and transferred from the receiver to the host main memory through the DMA. The received frame is compared with the original transmitted frame, which tests the host memory to the host adapter DMA path.
External Loopback Test	The External Loopback Test verifies the proper functioning of the GBIC module. A frame is created in the host adapter local memory, sent out and looped back through the external loopback connector attached to the port. If the external loopback test is run together with the internal loopback test, the DMA path is also tested by creating a frame in host main memory, transferring it to the host adapter through the DMA, looping it back through the external loopback connector and transferring the received frame back to the host main memory by DMA.
Loopback Frame Test	Sends out a buffer initialized with the selected pattern and compares it with the looped back frame. It passes if the two compare and fails if they do not.
Loopback Frame Pattern	List of selectable patterns for the Loopback Frame Test.

Note – In addition to the tests described above, `socaltest` also tests the basic functions of the SOC+ chip, the on-board XRAM, and the host control buffer by invoking the appropriate tests implemented in firmware.

`socaltest` Test Modes

`socaltest` can only be run in the Functional Test modes.

Note – You cannot run the internal and external loopback tests if the port is connected to a disk array.

`socaltest` Command Line Syntax

/opt/SUNWvts/bin/socaltest *standard arguments*
-o dev=*device*
*name,***elb=***enabled | disabled,***ilb=***enabled | disabled,***lbf=***enable | disable,***ptn=***pattern*

TABLE 32-2   `socaltest` Command Line Syntax

Argument	Explanation
dev=*device name*	The name of the socal port to be tested.
elb=*enabled	disabled*	Enables or disables the external loopback test.
ilb=*enabled	disabled*	Enables or disables the internal loopback test.
`lbf=`*enable	disable*	Enables or disables the loopback frame test.
`ptn=`*pattern*	Specify the pattern in hexadecimal, for example: `ptn=0x7e7e7e7e`

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`socaltest` Error Messages

TABLE 32-3 `socaltest` Error Messages

Error Message		Probable Cause(s)	Recommended Action
8000	`Couldn't open` *device_name*	SOC+ card not installed.	Verify that the target SOC+ card is installed properly.

8002	`{SOC+ host adapter}--`
	`{host memory} loopback`
	`test failed!`
		Fault in DMA between host and SOC+ host adapter.	Contact your authorized Sun service provider.

8006	`SOC+ host adapter`
	`external loopback test`
	`failed!`
		Loopback cable not connected.	Connect loopback cable.

		OEC module faulty or not installed properly.	Replace OEC module and retry.
			Contact your authorized Sun service provider.

8009	`SOC+ host adapter`
	`internal loopback test`
	`failed!`
		SOC+ host adapter is faulty or is not properly installed.	Verify that the SOC+ host adapter is installed properly.

			Replace the host adapter and retry.
			Contact your authorized Sun service provider.
8010	`Failed ioctl` *ioctl_name*	System error.	Contact your authorized Sun service provider.
8011	`Host control buffer test` `failed`	SOC+ host adapter is faulty.	Replace the host adapter and retry.
			Contact your authorized Sun service provider.
8012	`SOC+ selftest fails`	SOC+ host adapter is faulty.	Replace the host adapter and retry.
			Contact your authorized Sun service provider.

TABLE 32-3 `socaltest` Error Messages (Continued)

Error Message	Probable Cause(s)	Recommended Action
8013 `XRAM test fails`	SOC+ host adapter is faulty.	Replace the host adapter and retry.
		Contact your authorized Sun service provider.
8015 `Loopback frame test failed` `Mismatch in looped back frame` `Expected:` *value* `Observed:` *value*		

Serial Parallel Controller Test (`spiftest`)

The `spiftest` accesses card components such as the cd-180 and ppc2 chips, and the serial and parallel ports through the serial parallel controller device driver.

`spiftest` Hardware Requirements

Before running the SunVTS system exerciser, make sure you install the cards to be tested and the device driver. You should also reboot your system with the `boot -r` command to reconfigure the system and allow the SunVTS kernel to recognize the new driver.

Note – The `spiftest` must be run in Intervention mode.

The following minimum hardware configuration is required to successfully run the Internal test:

- SBus-based SPARC desktop system with an SBus slot
- Serial parallel controller card, installed in one of the SBus slots

The following hardware is also required to run the other SunVTS serial parallel controller tests:

- Serial parallel controller patch panel (part number 540-2007)
- 96-pin loopback plugs (part number 370-1366)
- 25-pin serial loopback plugs (part number 540-1558)
- RS-232 serial cables (part number 530-1685)
- TTY terminal

`spiftest` Options
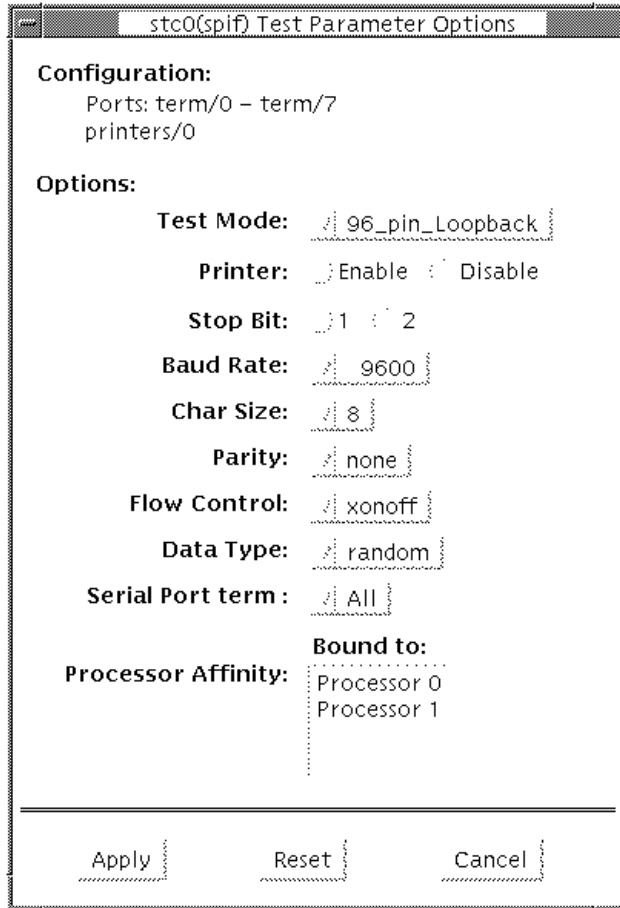
FIGURE 33-1 `spiftest` Option Menu

The Configuration section of the Option menu displays the serial ports available for the SPC/S board. The available ports are listed in the table below.

TABLE 33-1 `spiftest` Serial Ports for the SPC/S Board

Board Number	Board Device	Serial Ports	Parallel Ports
0	stc0	term/0-7	printers/0
1	stc1	term/8-15	printers/1
2	stc2	term/16-23	printers/2
3	stc3	term/24-31	printers/3
4	stc4	term/32-39	printers/4
5	stc5	term/40-47	printers/5
6	stc6	term/48-55	printers/6
7	stc7	term/56-63	printers/7

The `spiftest` options are described in the table below.

TABLE 33-2 `spiftest` Options

spiftest Option	Description
96-pin Loopback (LB)	Provides data transmission, full-modem loopback, and parallel port loopback testing. You *must* attach a 96-pin loopback connector (part number 370-1366) to the card before running this test (see Appendix A, "96-Pin Female Lookback Connector" section)..
Internal Test	Performs a quick internal check of the serial parallel controller card(s) installed in SBus slots. You do not need to attach anything to the card(s) to perform this test.
25-pin Loopback (LB)	Provides full-duplex transmission and full-modem loopback testing of the serial port selected in the Serial Port selection of this menu. You *must* attach the 25-pin Loopback plug to the serial port on the Serial Parallel Controller Patch Panel that is being tested (See Appendix A). This test cannot be run concurrently with the Echo-TTY option enabled.

TABLE 33-2 `spiftest` Options

spiftest Option	Description
Echo-TTY	Checks the proper operation of the serial port selected in the Serial Port section of this menu by echoing characters typed on a TTY terminal keyboard to the TTY terminal screen. The characters you type should be displayed on the TTY screen. Note: TTY connection to the `spif` serial port requires corresponding character size setup. For example, if a TTY attachment is running with 8-bit character size, then the `spiftest` option 'Char Size' should also be set to 8 bits. If you do not type within two minutes, this test will time-out. Terminate testing by pressing Control-C. After a short delay, the Status window updates the Pass Count. This test cannot be run concurrently with the 25-pin Loopback subtest.
Printer	Sends the entire ASCII character set to a parallel printer. You must attach a parallel printer to the parallel port on the Serial Parallel Controller patch panel. Observe the printer output to validate the test.
Baud Rate	Specifies the baud rate; choose 110, 300, 600, 1200, 2400, 4800, 9600, 19200, or 38400 baud. Note: The baud rate of 38400 can only be used if one port is tested at a time and the Internal Test is disabled.
Char Size	Indicates character length; choose 5, 6, 7, or 8 characters.
Stop Bit	Specifies the number of stop bits; choose 1 or 2 bits.
Parity	Specifies the selectable parity; choose `none`, `odd` or `even`.
Flow Control	Specifies the selectable flow control; choices are `XOnOff`, `rtscts`, or `both`.
Data Type	Specifies the selectable data type pattern; it can be 0x55555555 (0x55), 0xaaaaaaaa (0xaa), or random.
Serial Port	Specifies the serial port to be tested. The available ports are listed in the Configurations section at the top of the `spiftest` Options menu.

You can also change the test options by modifying the `/opt/SUNWvts/bin/ .customtest` file. See "Adding Your Own Tests" in `.customtest` in the *SunVTS 3.0 User's Guide.*

`spiftest` Test Modes

The `spiftest` supports only the Functional test mode.

`spiftest` Command Line Syntax

/opt/SUNWvts/bin/spiftest *standard_arguments* **-o dev=***device_name***,**
M=*test_mode***,Ptr=***enable/disable***,B=***baud_rate***,Size=***character_size***,S=***#of_stop_bits***,**
Parity=*parity***,F=***flow_control***,Data=***test_pattern***,sp=**serial_port

TABLE 33-3 `spiftest` Command Line Syntax

Argument	Explanation
dev=*device_name*	Specifies the serial ports in SBus card slots (0-63) being tested. Since there is no default, you must type a board name:
	`stc0` **=** the **8** serial ports in the first card
	stc1 = the **8** serial ports in the second card
	stc2 = the **8** serial ports in the third card
	`stc3` = the **8** serial ports in the fourth card
	`stc4` = the **8** serial ports in the fifth card
	`stc5` = the **8** serial ports in the sixth card
	`stc6` = the **8** serial ports in the seventh card
	`stc7` = the **8** serial ports in the eighth card
M=*test_mode*	Specifies `Internal`, `96_pin_Loopback`, `25_pin_loopback`, or `Echo_TTY` test mode.
Ptr=*printer_test*	Enables or disables the Printer subtest.
B=*baud_rate*	Sets the baud rate to 110, 300, 600, 1200, 2400, 4800, 9600, 19200, or 38400; the default is 9600. To use the 38400 rate, only one port at a time can be tested, and the `Internal` test must be disabled.
Stop=*#of_stop_bits*	Toggles the number of stop bits between 1 or 2; the default is 1.
Size=*character_size*	Sets character size as a number between 5 and 8.
P=*parity*	Specifies the parity as none, odd, or even; the default is none.

TABLE 33-3   `spiftest` Command Line Syntax

Argument	Explanation
F=*flow_control*	Specifies flow control as `xonxoff,rtscts`, or both.
Data=*test_pattern*	Specifies test pattern as 0x55555555, 0xAAAAAAAA, or random.
sp=*serial_port*	Specifies the terminal and serial port number, such as `term/3`.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`spiftest` Error Messages

These error messages are generated when the SunVTS serial parallel controller discovers any errors. The error descriptions below identify probable causes for the card or test failure, and identify the Field Replaceable Unit (FRU), if possible. The three FRUs affected are: the serial parallel controller card, the 96-pin shielded cable, and the patch panel.

TABLE 33-4   `spiftest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	*error_message*		
6001	`Expected = ` *value*`, observed = ` *value*		
6002	`Expected ` *value*`:`		
6003	`Observed ` *value*`:`		
6004	`Modem Loopback test failed on ` *device_name*		
6005	`Off-line error on device` *device_name*		
6006	`Paper out error on device` *device_name*		
6007	`Busy error on device` *device_name*		

TABLE 33-4 spiftest Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6008	Error on device *device_name*		
6009	retry *number* exceeds limit *number* rc =s *value*		
6010	Expected *number* bytes, observed *number* bytes		
6011	Internal test failed on *device_name*		
6012	Data Loopback test failed on *device_name*		
6013	ioctl TIOCSSOFTCAR error on *device_name*		
8000	Must provide device name!		
8001	No SPC/S card found or device driver not installed		
8002	ioctl TCSETS failed on *device_name*		
8003	Open error on *device_name* - device driver may not be installed properly		
8004	Device *device_name* is already opened		
8005	ioctl STC_SREGS(STC_IOREG-COR2) error in *device_name*		
8006	ioctl STC_GREGS(STC_IOREG-CCR) error in *device_name*		
8007	ioctl STC_S1REGS(STC_IOREG-CCR) error in *device_name*		
8008	ioctl STC_SREGS(STC_PPCREG-PDATA) error in *device_name*		
8009	ioctl STC_GREGS(STC_PPCREG-PDATA) error in *device_name*		
8011	ioctl TIOCMGET failed on *device_name*		
8012	*device_name*: Expected DSR set, observed clear		
8013	*device_name*: Expected DSR clear, observed set		

TABLE 33-4 `spiftest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8014	ioctl TIOCMSET failed on *device_name*		
8015	ioctl TIOCMBIC failed on *device_name*		
8016	Read error on *device_name*		
8017	Write error on *device_name*		
8018	ioctl STC_GPPC error on device *device_name*		
8019	Can't kill all processes		
8020	Timeout error on *device_name*.	There is no loopback plug	Check the loopback plug.
8021	Timeout error on *device_name*	TTY terminal not connected	Check the TTY terminal connection.
8022	Timeout error on *device_name*		

CHAPTER **34**

Serial Ports Test (`sptest`)

`sptest` checks the system's on-board serial ports (`zs[0,1]`, `zsh[0,1]`, `se[0,1]`, `se_hdlc[0,1]`), as well as any multi-terminal interface (ALM2) boards (`mcp[0-3]`). Data is written and read in asynchronous and synchronous modes utilizing various loopback paths.

Intervention mode must be enabled to run this test. This test is non-scalable.

TABLE 34-1   Serial Port Tests

Mode	Description
Asynchronous Testing	This mode uses the asynchronous communication protocol as described in the `zs(7D)` and `se(7D)` man pages. The `termio(7I)` interface is used for configuring port characteristics. The user can select the loopback paths to use, the amount of data to transfer, and the baud rate to use. The test writes and reads data through the loopback path and compares the data to the original data. The test first sends a single character. If no errors or timeouts are detected, the rest of the data is simultaneously written and read, then compared.

TABLE 34-1   Serial Port Tests (Continued)

Mode	Description
Synchronous Testing	This mode uses the synchronous hdlc-framing interface as described in the zsh(7D) and se_hdlc(7D) man pages. Data is written and read in checksum-protected packets.
	The user can select the loopback paths to use and the clock rate.
	The synchronous test runs in three phases:
	• The first phase looks for activity on the port. If no activity is detected for at least four seconds, the test proceeds to the next phase. If activity is detected sptest exits with an error. • The second phase attempts to send and receive one packet. If no packets are detected after five attempts, the test exits with an error. If a packet is returned, the result is compared to the original. If the length and content of the packets do not match exactly, the test exits with an error. • The third phase attempts to send many packets through the loop. Some packet drops are to be expected especially on a heavily loaded system. The test allows a percentage of the packets to be dropped. The user can set the drop tolerance between 0 percent and 99 percent. The default is 20 percent. If the system is extremely busy then the drop tolerance should be increased. Each packet is compared with its original for length and content. If a mismatch is detected, the test exits with an error.

Synchronous Testing Software Requirements

If you have zs(7D) serial ports on your machine, the synchronous devices may not exist. Look in the /dev directory for zs*h* (where *h*=0 and/or 1). If they do not exist, you can create them. Verify that the following two lines are in the /etc/devlink.tab file. If they are not there, add them.

```
type=ddi_pseudo;name=zsh zsh\M0
type=ddi_pseudo;name=clone;minor=zsh zsh
```

Caution – The white spaces in the lines above *must* be a single tab character before and after the zsh variables; using spaces will not work.

When these lines have been added to the /etc/devlink.tab file:

● **Change directories to** /kernel/drv, **and run the add_drv zsh command.**

If this command does not work:

● **run the rem_drv zsh command and then run the add_drv zsh command again.**

sptest Options

FIGURE 34-1 sptest Option Menu

There are a variety of loopback paths available. The internal loopback paths do not require an external connector. Their availability depends on the device. The zs(7D) device has an internal path for synchronous mode and the se(7D) device has an internal path for asynchronous mode. The external loopback connectors are described in Appendix A. The exact type of loopback connector required depends on the system I/O panel.

Legend: In the following table, *a* represents port a of the CPU board (motherboard), *b* represents port b of the CPU board, and the device names of the ports for each CPU board are listed.

TABLE 34-2   sptest Serial Devices

CPU	Port	Async Device	Sync Device
0	a b	zs0 or se0 zs1 or se1	zsh0 or se_hdlc0 zsh1 or se_hdlc1
1	a b	zs2 zs3	zsh2 * zsh3 *
2	a b	zs4 zs5	zsh4 * zsh5 *

* Currently, only zsh0 and zsh1 are supported by device drivers.

TABLE 34-3   sptest Options

sptest Options	Description
Test Type	Selects how the test will run. Test options include: • a= runs the test on port a • b= runs the test on port b • a_b= runs the test on ports a and b sequentially • a_b_concurrent= runs the test on port a and port b concurrently.
Test Mode	Selects the mode to put the serial device into before running the test. The modes available are Asynchronous, Synchronous or Both. When Both is selected, the test runs in Asynchronous mode then Synchronous mode.

TABLE 34-3   `sptest` Options

sptest Options	Description
Loopback Type	Selects the loopback test. Options include: •`Internal` is an internal path for a, b, a_b, and a_b_concurrent test types. •`Plug_a_to_a__b_to_b` is an external loopback plug for a, b, a_b, and a_b_concurrent test types. •`no_modem_a_to_b` is an external loopback cable for a_to_b and a_to_b_concurrent test types. •`Modem_a_to_b` is an external loopback cable with a modem attached to generate synchronous Transmit and Receive clocks in synchronous mode. The modem a_to_b external loopback type is intended for Sun internal use only. It requires custoe equipment that is not available.
Data Type	Selects the data pattern to transfer. The user can select: •`Random` •`Sequential` •`Alphanumeric` •`0x00-0xff`
Async Baud Rate	Selects the baud rate for Asynchronous mode testing. The valid rates are: 110, 300, 600, 1200, 4800, 9600, 19200, 38400, 57600, 76800, 115200, 153600, 230400, 307200, 460800, and ALL. The default rate is 9600 baud. Some platforms can only support up to 38400 or 76800. The test will return an error if you try to use a higher baud rate then is supported. For baud rates greater then 153600 the serial line drivers must be set for RS-423 mode and not RS-232 mode. The RS-423 and RS-232 modes are usually selected by a hardware jumper on the motherboard. Consult your hardware installation manual for more information.
Async Block Size	Selects the size (in bytes) of each write. This can range from 1 to 10000. The default is 100. If it is larger than the async size it will be truncated to the async size.
Async Data Size	Selects the total number of bytes to transfer in Asynchronous mode. This can range from 1 to 10000.
Async Parity	Selects the parity to use in asynchronous mode. The user can select even, odd, or none. The default is `none`.
Async Flow Control	Selects the type of flow control to use in asynchronous mode testing. The user can select `Hardware` (RTS/CTS), `Software` (XON/XOFF) or `None`. The default depends on the loopback type. Software flow control is not allowed on a, b, a_b, or a_b_concurrent loopback types.

TABLE 34-3 `sptest` Options

sptest Options	Description
Sync Baud Rate	Selects the device generated clock rate for synchronous mode testing. The valid rates are from 110 to 230400. The rate does not have to be a specific value as required for async mode baud rates. The default rate is 9600. Some platforms can only support up to 38400 or 76800. The test will return an error if you try to use a higher rate then is supported. For rates greater then 100000 the serial line drivers must be set for RS-423 mode and not RS-232 mode. The RS-423 and RS-232 modes are usually selected by a hardware jumper on the motherboard. Consult your hardware installation manual for more information.
Sync Packet Drop Tolerance	Selects the tolerance level of Synchronous mode dropped packets during the many_packets subtest. The default is 20 percent. The valid range is from 0 percent to 99 percent. Some packet drops are expected especially at higher clock rates and on a heavily loaded system.
Sync Poll Count	Selects the number of seconds in additional time to wait for a Synchronous mode packet to be sent. Additional time may be needed when there is heavy system activity and time-outs are being detected. In general, the user can decrease the value to 0 when the system load is light or increase the value when there is a heavy system load.

`sptest` Test Modes

`sptest` supports all three SunVTS test modes.

TABLE 34-4   `sptest` Test Modes

Test Mode	Description
Connection Test	In this mode, the `sptest` tries to open the port to determine if the device is connected. If it fails and the port is not busy, the test exits with an error. If it is successful or fails with a busy or exclusive use error, then the port is considered connected, and the test passes.
Functional Test	In Functional test, `sptest` performs the selected loopback test.
Functional mode through SyMON	If you invoke SunVTS through SyMON, the following test description applies: `sptest` tries to minimize the impact on the applications that are running. The Online test runs in synchronous and asynchronous modes. If the device supports an internal loopback then the functional test is run utilizing the internal loopback. The `zs(7D)` device supports an internal synchronous loopback and the `se(7d)` device supports an internal asynchronous loopback.

`sptest` Command Line Syntax

/opt/SUNWvts/bin/sptest *standard_arguments* **-o
dev=***device_name***,porta=first_port_name,T=***port***,L=***loopback_type***,M=***mode***,D=***data
_pattern***,AB=***async_baudrate***,BS=***async_write_size***,S=***async_total_size***,PAR=***parity***,F
=***flow_control***,B=***sync_speed***,DP=***sync_drop_tolreance***,P=***sync_timeout*

TABLE 34-5   `sptest` Command Line Syntax

Argument	Explanation

dev=*device_name*	Identifies the serial port(s) to test. There is no default value; you must specify a device name such as:
	`se0,`
	`zs0, zs1`
	`zs2, zs3`
	..., ...

porta=*port_name*	The name of the first device of a serial device pair. The default is `a`.

TABLE 34-5 `sptest` Command Line Syntax (Continued)

Argument	Explanation (Continued)
M=*mode*	The default test mode is asynchronous. Specify one of the following modes: • asynch • synch • both
T=*test_type*	Specifies the type of test to run: • a= runs the test on port a. • b= runs the test on port b. • a_b= runs the test on ports a and b sequentially. • a_b_concurrent= runs the test on port a and port b concurrently. • a_to_b= runs the test from port a to port b.
L=*test_loopback*	The type of loopback connector attached to ports: • No_modem_a_to_b • Internal_a_to_a__b_to_b • Plug_a_to_a__b_to_b • Modem_a_to_b
D=data_pattern	Selects the data pattern to transfer. The user can select: • Random • Sequential • Alphanumeric • 0x00-0xFF
AB=*baud_rate*	Asynchronous baud rate (default = 9600). The valid values are between **110 - 460800.** Note: Some platforms can only support asynchronous baud rates up to 38400 or 76800. For baud rates greater then 153600 the serial line drivers must be set for RS-423 mode and not RS-232 mode.
BS=*write_size*	Asynchronous mode write size; from 1 to 10000 bytes.
S=*size*	Asynchronous mode total number of bytes to write. 1 to 10000 bytes.
PAR=*parity*	Asynchronous mode parity type: • None • Even • Odd
F=*flow_control and default*	Asynchronous mode flow control • Hardware (RTS/CTS) • Software (xon/xoff) • None
B=*baud_rate*	Synchronous baud rate (default = 9600). The valid rates are between **110 - 256000.** Note: Some platforms can only support synchronous rates up to 38400 or 76800. For rates greater then 100000 the serial line drivers must be set for RS-423 mode and not RS-232 mode.

TABLE 34-5   `sptest` Command Line Syntax (Continued)

Argument	Explanation (Continued)
DP=*drop_tolerance*	Synchronous mode drop packet tolerance (default=20 percent)
P=*poll_seconds*	Synchronous mode additional wait time during poll (in seconds).

> **Note –** 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`sptest` Error Messages

TABLE 34-6   `sptest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6003	`data miscompare, cnt =` *number*`, exp =` *value*`, obs =` *value*`, receive device =` *name*`, source device =` *name*`, baud rate =` *value*	System load too heavy for selected baud rate.	Reduce the system load.
		Loopback connection defective.	Lower the baud rate.
		Bad serial port hardware.	Check loopback connection.
		System software error.	If the problem persists, contact your authorized Sun service provider.
6011	*name*`: packet read failed, error =` *error_message*`, speed =` *number*	System load too heavy.	Reduce the system load.
		Loopback connection missing or defective.	Check loopback connection.
		Bad serial port hardware.	If the problem persists, contact your authorized Sun service provider.
		System software error.	
6012	*name*`: packet write failed, error =` *error_message*`, speed =` *number*	System load too heavy.	Reduce the system load.
		Bad serial port hardware.	If the problem persists, contact your authorized Sun service provider.
		System software error	

TABLE 34-6 `sptest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6013	Device '*device_name*' not responding. No data detected after *number* seconds. Received *number* bytes out of *number* sent. Baud rate = *number*	System load too heavy for selected baud rate.	Reduce the system load.
		Loopback connection missing or defective.	Lower the baud rate.
		Bad serial port hardware.	Check the loopback connection.
		System software error.	If the problem persists, contact your authorized Sun service provider.
6014	*name*: expected *text* got *text*		
6015	Incomplete write on Port *name*, len *number* should be *number*, speed = *number*	System load too heavy.	Reduce the system load.
		Loopback connection missing or defective.	Check loopback connection.
		Bad serial port hardware.	If the problem persists, contact your authorized Sun service provider.
		System software error.	
6016	Incomplete read on Port *name*, len *number* should be *number*, speed = *number*	System load too heavy.	Reduce the system load.
		Loopback connection missing or defective.	Check loopback connection.
		Bad serial port hardware	If the problem persists, contact your authorized Sun service provider.
		System software error.	
6018	*name*: port not ready for writing after *number* seconds, speed = *number*	System load too heavy	Reduce the system load.
		Loopback connection missing or defective.	Check loopback connection.
		Bad serial port hardware.	If the problem persists, contact your authorized Sun service provider.
		System software error.	
6020	packet data miscompare, cnt = *number* exp = *value*, obs = *value*, device = *name*, speed = *number*	System load too heavy	Reduce the system load.
		Loopback connection defective.	Check loopback connection.
		Bad serial port hardware.	If the problem persists, contact your authorized Sun service provider.
		System software error.	

TABLE 34-6 `sptest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6021	*name*: port not ready for reading after *number* seconds, no data available, speed = *number*	System load too heavy.	Reduce the system load.
		Loopback connection missing or defective.	Check loopback connection.
		Bad serial port hardware.	If the problem persists, contact your authorized Sun service provider.
		System software error.	
6022	*name*: port not ready for writing after *number* seconds, wrote *number* bytes out of *number*. Baud rate = *number*	System load too heavy.	Reduce the system load.
		Loopback connection missing or defective.	Check loopback connection.
		Bad serial port hardware.	If the problem persists, contact your authorized Sun service provider.
		System software error	
8001	Test Type a_to_b is not allowed on Quick Test!		
8002	*name*: open() failed, error = *error_message*		
8003	Invalid Data Type: I=*text*		
8004	*name*: poll() error, (revents & (POLLERR | POLLHUP | POLLNVAL)) = *value*		
8005	Siemens serial device (se) does not support synchronous internal loopback test		
8006	Invalid port parameter: *name*		
8007	Invalid device name '*name*', must specify either zs*X* or se*X*, where X is the first instance number of a device pair (ex., zs0)		
8009	Failed ioctl *name*, device = *name*, error = *error_message*		
8010	*name*: Not able to push Tty compatibility STREAMS module		

TABLE 34-6 `sptest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8011	Invalid Test Type: T=*name*		
8012	Invalid Mode: M=*name*		
8013	*name* device does not exist		
8014	Invalid baud_rate: *number*		
8015	write() to *name* failed: *error_message*, baud rate = *number*		
8016	read() to *name* failed: *error_message*, baud rate = *number*		
8017	Invalid size, requested = *number* bytes, min = *number*, max *number*		
8018	Could not open *name* after waiting *number* seconds	Another process is using the port	Stop other process using the port.
		Port is hung	If the problem persists, contact your authorized Sun service provider.
		Bad serial port hardware	
		System software error	
8021	*name*: No valid packets returned after *number* attempts, last error = *error_message*, wrote *number* packets, read *number* packets, device accounting shows *number* packets sent and *number* packets received	System load too heavy	Reduce the system load.
		Timeout too short	Increase `Sync_Poll_Count` from pop menu.
		Loopback connection missing or defective	Check the loopback connection.
		Bad serial port hardware	If the problem persists, contact your authorized Sun service provider.
		System software error	
8022	printdlprim: unsupported primitive type *value*		
8023	stringtoaddr: invalid input string: *text*		
8025	dlinfoack: response ctl.len too short: *number*		
8026	dlinfoack: DL_INFO_ACK was not M_PCPROTO		
8027	dlokack: response ctl.len too short: *number*		

TABLE 34-6 `sptest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8028	dlokack: DL_OK_ACK was not M_PCPROTO		
8029	dlerrorack: response ctl.len too short: *number*		
8030	dlerrorack: DL_OK_ACK was not M_PCPROTO		
8031	dlbindack: response ctl.len too short: *number*		
8032	dlbindack: DL_OK_ACK was not M_PCPROTO		
8033	sigalrm: TIMEOUT		
8037	invalid packet length: *number*		
8038	*name*: poll() failed, error = *error_message*	System load too heavy	Reduce the system load.
		Bad serial port hardware	If the problem persists, contact your authorized Sun service provider.
		System software error	
8039	Failed I_STR ioctl, cmd = *name*, device = *name*, error = *error_message*		
8040	*name*: No packets sent, last error = *error_message*, wrote *number* packets, read *number* packets, device accounting shows *number* packets sent and *number* packets received	No transmit clock present	Reduce the system load.
		System load too heavy	Increase `Sync_Poll_Count` from the pop menu.
		Timeout too short	Check the loopback connection.
		Loopback connection missing or defective	If the problem persists, contact your authorized Sun service provider.
		Bad serial port hardware	
		System software error	
8041	Sporadic packets detected on *name*. The port must be quiescent before running test	Another process is generating activity on the port	Stop other process using the port.
		Bad serial port hardware	Stop other process using the port.
		System software error	
8042	syncinit: *name* missing minor device number		

TABLE 34-6 `sptest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8046	Test requires local or remote modem loopback (external data loop, external clocking)		
8047	*name* loopback type not allowed on Connection test		
8048	*name* loopback type not allowed on online mode		
8049	Invalid baud rate for device *name*, select between *number* and *number*		
8050	*name* internal loopback test not allowed on *name* device		
8051	Thread creation failed, *error_message*		
8054	*name*: At least *number* packets dropped (tolerance = *number*), last error = *error_message*. Wrote *number* packets, read *number* packets. Device statistics shows *number* packets sent and *number* packets received. *number* packets dropped in outbound queueing, *number* packets dropped in inbound queueing	System load too heavy	Reduce the system load.
		Loopback connection missing or defective	Increase the drop tolerance option.
		Bad serial port hardware	Check the loopback connection.
		System software error	If the problem persists, contact your authorized Sun service provider.
8055	*name*: *number* bad packets received, last error = *error_message*, wrote *number* packets, read *number* packets, device accounting shows *number* packets sent and *number* packets received	System load too heavy	Reduce the system load.
		Loopback connection missing or defective	Check the loopback connection.
		Bad serial port hardware	If the problem persists, contact your authorized Sun service provider.
		System software error	

SunButtons Test (`sunbuttons`)

The `sunbuttons` test verifies that the SunButtons graphics manipulation device is working correctly.

Note – Install SunVTS 3.0 making sure that the following packages are installed: `SUNWdial` and `SUNWdialh`.

Note – Do not run `sunbuttons` and `sundials` at the same time.

Running the `sunbuttons` Tests

The following tests verify the functionality of SunButtons:

▼ Using the `.customtest` File

Modify the `.customtest` file as described below to verify that each button functions; each button lights up in a round-robin fashion.

Note – The `.customtest` file is located in in two places. Modify the 32-bit (in the `/opt/SUNWvts/bin` directory) or the 64-bit (in the `/opt/SUNWvts/bin/sparcv9` directory) `.customtest` file according to the Solaris environment that you are using.

● **Edit the** `.customtest` **file to include the following lines:**

```
sunbuttons;sunbuttons; Mode<CYCLE|diag|diag|mode>
sundials;sundials; Mode<CYCLE|diag|diag|mode>
(you may have to execute the xhost + command)
```

▼ Configure the Port Using the `bdconfig` Command

To configure the port, perform the following steps:

1. **Run the** `/usr/sbin/bdconfig` **command to configure the** `/dev/term/a` **or** `/dev/term/b` **port and enable it. The following example shows how to check a configured serial port b:**

```
# ls -l /dev/bd*
lrwxrwxrwx  1 root   other     11 Nov 12 15:39 /dev/bd -> /dev/term/b
#
```

2. **Connect the sunbuttons/sundials device to your configured serial port (a or b).**

▼ Start SunVTS

After starting SunVTS, you should see Customtest as one of the available options with `sundials` and `sunbuttons` available for selection.

1. **Change to the Functional test mode and enable intervention.**

2. **Enable the sunbuttons and sundials tests.**

3. **Start testing.**

   In Diag mode, a pop-up window is displayed.

4. **Select the Diagnostics button.**

   Let the test run its course (the Diagnostics button will be selectable again).

5. **Close the pop-up window.**

   The test should register a single pass.

▼ Running the `sunbuttons` Test From a Shell Command Line

This is an interactive test. The test displays a screen representation of the buttonbox (see FIGURE 35-1) where you can press each of the buttons and see the corresponding button's display change.

● To run the diagnostic test, select the Diagnostics Button on the top of the menu representation. The buttonbox buttons on the screen do not change while the diagnostics test is running.

Be sure that the buttonbox is connected to one of the serial ports, and that the buttonbox has a power transformer.

There is no option menu for this test.
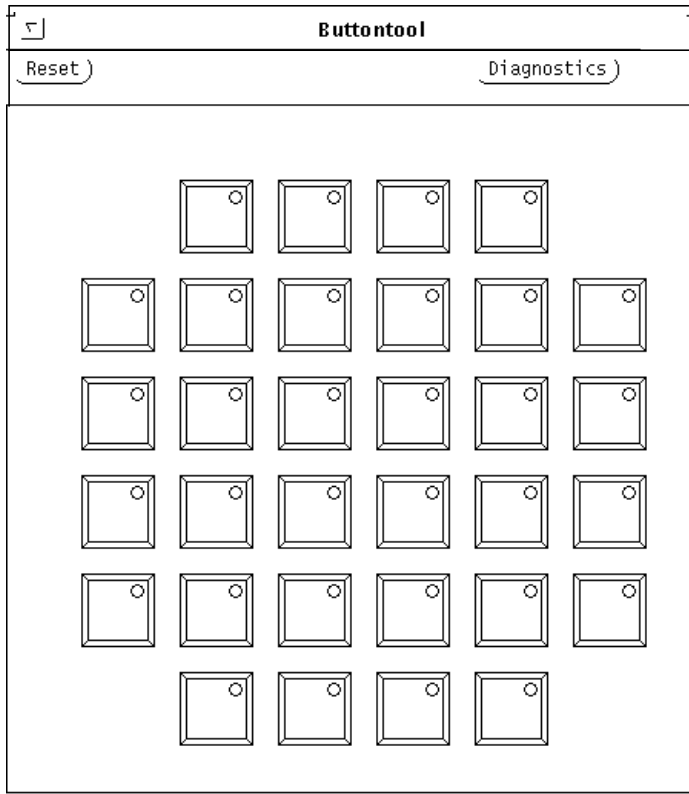
FIGURE 35-1 `sunbuttons` Test Menu

`sunbuttons` Command Line Syntax

/opt/SUNWvts/bin/sunbuttons diag *standard_arguments*

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`sunbuttons` Test Modes

Available only in the Functional test mode. In the Functional test mode, the `sunbuttons` test verifies the functionality of each button, and that each button lights up in sequence.

`sunbuttons` Error Messages

TABLE 35-1   `sunbuttons` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	Cannot open device		
6001	ioctl(VUIDSFORMAT, VUID_NATIVE)		
6002	Writing diag command failed		
6003	No response from Buttonbox		
6004	Selftest failed		
6005	Weird unknown kind of error		
6006	Can not open device /dev/db		
6007	ioctl(VUIDSFORMAT, VUID_NATIVE) -- you may need to run bdconfig		

SunDials Test (`sundials`)

The `sundials` test verifies that the SunDials graphics manipulation device controls are working properly. `sundials` also verifies the connection between the dialbox and serial port.

Note – Do not run `sundials` and `sunbuttons` at the same time.

Running the `sundials` Test

The following tests verify SunDial functionality.

▼ Using the `.customtest` File

● **Modify the** `.customtest` **file to test the SunDials.**

Here is an example of a `.customtest` entry for `sundials`:

```
sundials;sundials;text<TEXT|20|-s|>
```

Note – The `.customtest` file is located in in two places. Modify the 32-bit (in the `/opt/SUNWvts/bin` directory) or the 64-bit (in the `/opt/SUNWvts/bin/sparcv9` directory) `.customtest` file according to the Solaris environment that you are using.

▼ From a Command Line

Running the sundials test from a command line starts an interactive test that displays a screen representation of the dialbox (see FIGURE 36-1). You can move each of the dials and see the corresponding dial's display change.

- **To run the interactive test, select the Diagnostics button on the top of the window representation. The dialbox has a power transformer..**

  Be certain that the dialbox is connected to one of the serial ports, and that the dialbox has a power transformer.

  There is no option menu for this test.

  The sundials Test menu is shown in FIGURE 36-1.
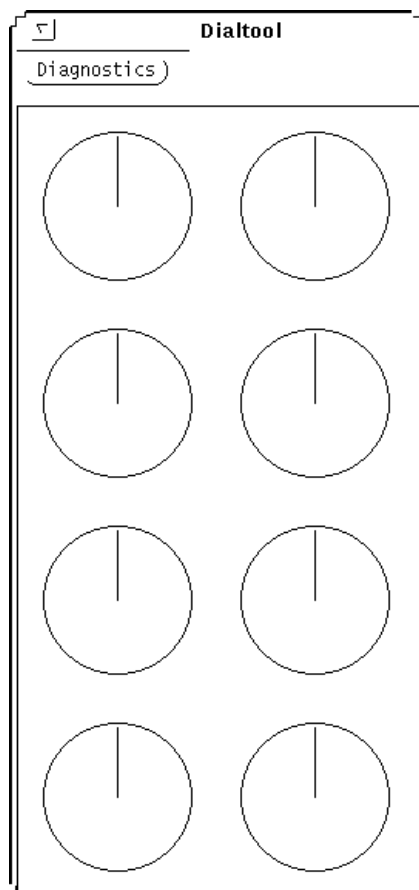
FIGURE 36-1 sundials Test Menu

`sundials` Test Modes

This test is available only in the Functional test mode. The `sundials` test verifies the connection between the dialbox and serial port. The test also verifies all manipulation device controls.

`sundials` Command Line Syntax

/opt/SUNWvts/bin/sundials diag *standard_arguments*

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`sundials` Error Messages

TABLE 36-1 `sundials` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	Open failed for /dev/bd		
6001	ioctl(VUIDSFORMAT, VUID_NATIVE) error		
6002	ioctl(VUIDSFORMAT, VUID_EVENT) error		
6003	Could not talk to Dialbox		
6004	Cannot open device		
6005	Writing Diag Command failed		
6006	No response from Dialbox		

TABLE 36-1 `sundials` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6007	Selftest failed		
6008	Weird unknown kind of error		
6009	ioctl(VUIDSFORMAT, VUID_NATIVE) -- you may need to run bdconfig		

HSI/S Board Test (`sunlink`)

The `sunlink` test verifies the functionality of the SBus HSI boards. `sunlink` tests the HDLC and SDLC protocol of SBus HSI boards and PCI bus HSI boards. `sunlink` downloads the DCP microcode, initializes the selected channel, and configures the selected channel to the protocol being tested.

Next, `sunlink` opens a datagram socket and tries to modify the socket to accept `ioctl` communications with the driver, and receive Synchronous mode information from it.

`sunlink` then opens the ports, linking the upper and lower layers with ioctl calls. After initialization, this test checks for activity before attempting to send or receive data. An error message is returned if activity is detected; otherwise the transmit buffer is filled with random data. Random data is used by default; other patterns may be specified. The data is then transmitted. If the transmission succeeds, `sunlink` then receives the returned data and verifies that it is identical to what was sent. Finally, statistics about the send and receive are gathered from the socket.

A full `sunlink` test takes approximately eight minutes per port and makes a brief check of the board ports before the actual test begins. If the port is bad, the test immediately aborts and returns an error message.

Note – This test will not pass unless you install the correct loopback connectors or port to port cables on the ports you are testing. The ports specified for test in the option menu must have loopback connectors attached. See Appendix A for loopback connector part numbers and wiring instructions.

`sunlink` Options

The Configuration field displays the available ports. (See FIGURE 37-1.)
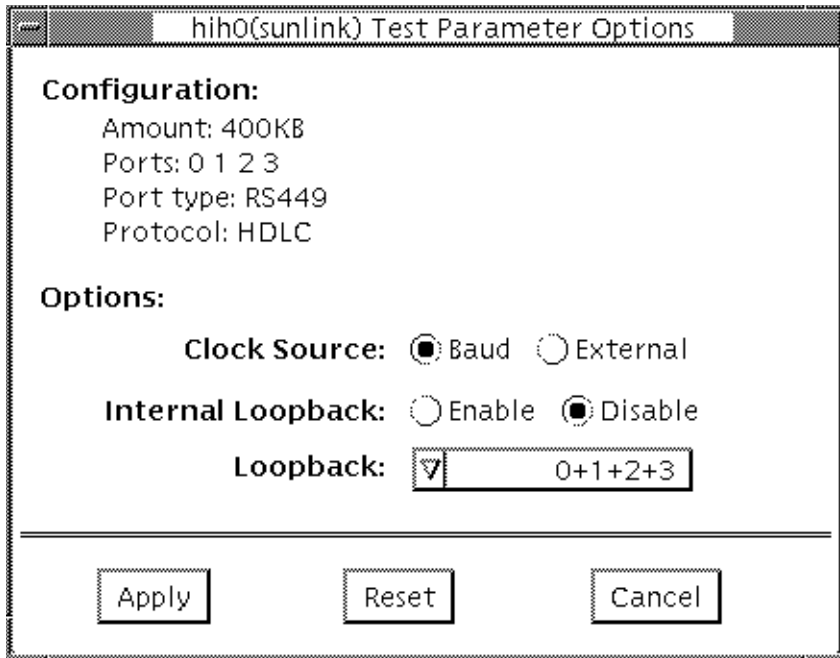
FIGURE 37-1 `sunlink` Options menu

TABLE 37-1 `sunlink` Options

`sunlink` **Options**	**Description**
Clock source	Select either the on-board clock or an external clock for use when using `sunlink`. To use the external clock option, the transmit, receive, and clock data lines must be physically loopbacked.
Internal Loopback	Enables or disables internal loopback tests. Internal Loopback is only needed when the Loopback setting is not port-to-port, and the clock source is on-board.
Loopback	Specifies the loopback type: simple single external port loopback, multiple external port loopback, and port-to-port external loopback.

`sunlink` Command Line Syntax

/opt/SUNWvts/bin/sunlink *standard_arguments* **-o dev=***device_name***,P=***port#***,**
p=*data_pattern***,I,C=**B/E

TABLE 37-2 `sunlink` Command Line Syntax

Argument	Explanation
dev=*device_name*	Specifies the device to be tested.
	hih0 HDLC and SDLC protocols
P=*ports*	Specifies the port number to be tested.
p=*data_pattern*	Specifies the *data_pattern* as one of the following:
	c: Character (0x55)
	i: Incrementing
	d: Decrementing
	r: Random (default)
I	Enables internal loopback for HSI/S (external).
c=*clocksource*	Specifies the clock source value as one of the following:
	B: On-board clock source
	E: External clock source

The following is a typical command line syntax for testing an HSI/S Sbus card:

```
# /opt/SUNWvts/bin/sunlink -o dev=hih0,P=0_to_1+2_to_3
```

Typing this at the command line tests the internal loopback for port 0.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`sunlink` Test Modes

The `sunlink` test only supports the Functional test mode.

`sunlink` Loopback Connectors

Refer to Appendix A of this manual, and the High Speed Serial Interface hardware manuals for information on loopback connectors.

`sunlink` Error Messages

TABLE 37-3   `sunlink` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	Device name must be provided!		
6001	Invalid port specification		
6002	Invalid port number		
6003	BSC protocol is not implemented for *device_name*		
6004	ASYNC protocol is not implemented for *device_name*		
6005	Illegal protocol specified for '*device_name*'		
6006	Could not open file: *file_name*		
6007	Could not open file devname *device_name*		
6008	Could not open file (clonename)*file_name*		
6009	Can't get sync mode info for *device_name*		

TABLE 37-3   sunlink Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6010	Can't set sync mode info for *device_name*		
6011	Couldn't successfully execute '/usr/sunlink/dcp/dcp *file_name*		
6012	Packet received but none sent! Activity on-line. Quiesce other end before starting		
6013	Transmit failed on '*device_name*'		
6014	Receive failed on '*device_name*'		
6015	Data compare error on '*device_name*', exp = *value*, actual = *value*, offset = *number*		
6016	*device_name* does not respond	No loopback plug.	Check the loopback plug.
		System load is too heavy.	Reduce the system load.
6017	sigalrm: TIMEOUT	No loopback plug.	Check the loopback plug.
		System load is too heavy.	Reduce the system load.
8000	Unsupported primitive type *value*		
8001	*text* (*error_message*)		
8002	Response ctl.len too short: *value*		
8003	DL_INFO_ACK was not M_PCPROTO		
8004	DL_OK_ACK was not M_PCPROTO		

Pixel Processor Test (`sxtest`)

`sxtest` checks models of SPARCstation 10 and SPARCstation 20 machines equipped with an onboard Pixel Processor module. `sxtest` is specific to the VSIMM (Video SIMM)/SX Memory Controller) devices in the SPARCstation 10 SX and SPARCstation 20 SX.

Note – Because of possible conflicts between `cg14` SunVTS frame buffer tests and OpenWindows applications that use the `cg14` frame buffer, the following restrictions apply when running the `sxtest` SunVTS test:

Do not run any graphic applications other than OpenWindows while running SunVTS frame buffer tests.

To start SunVTS with `vtsui`, but without `vtsk`, you must add the host name to `xhost` as: **xhost +** *hostname.*

Do not run any OpenWindows programs that generate video updates outside or on top of the SunVTS window.

Do not close the SunVTS window to an icon while it is running frame buffer tests.

Enable the frame buffer locking option from the Options window for the system console `cg14` device (see FIGURE 38-1).

If you run `sxtest` with VRAM enabled, then frame buffer locking *must* be enabled.

`sxtest` locates load error, store error, ALU error, logic error, and so on, of the pixel processor by reading and verifying data from the control registers of the pixel processor, virtual memories, or video memories.

This test also verifies the integration function of the `cg14` frame buffer and its device driver, video memories, and data memories. `sxtest` also writes a test pattern to the frame buffer for visual verification. The accuracy of this test is dependant on other

hardware, such as the CPU, memory card(s), and `cg14` graphics card. Run the `cg14test` and other tests before running `sxtest`. `sxtest` is a series of 13 modules, described FIGURE 38-1.

sxtest Options
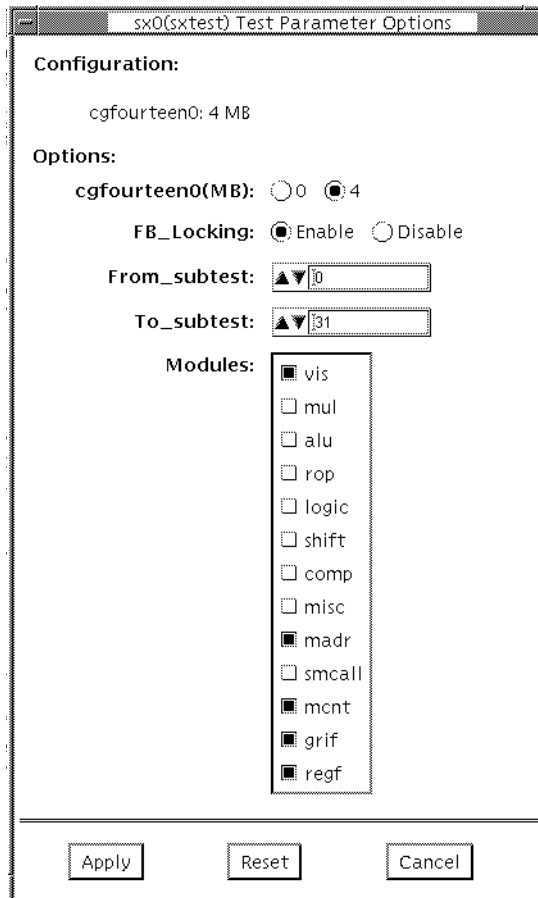
FIGURE 38-1 `sxtest` Option Menu (CMEM:0MB, none, `cg14` board)

Note – The `sxtest` option menu is customized, based on the configuration of the test system.
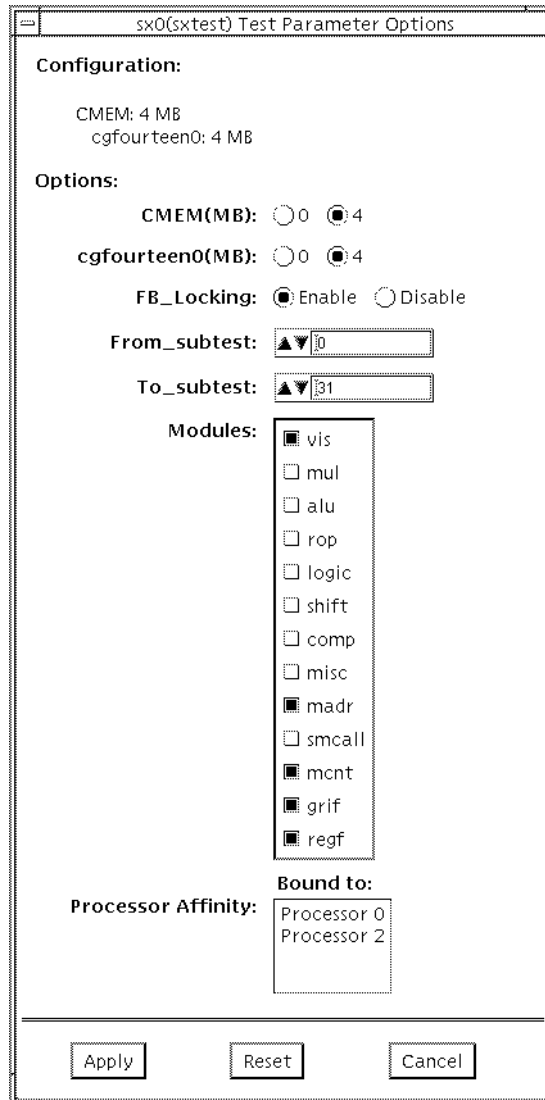
FIGURE 38-2 `sxtest` Option Menu (CMEM:4MB, cg14:4MB)

sxtest Test Parameters

The sxtest test parameter options are described in the table below.

TABLE 38-1 sxtest Test Parameter Options

Option	Description
FB Locking	Frame buffer locking is enabled by default on the window server running the OpenWindows software. See the section about testing multiple frame buffers in the *SunVTS 3.0 User's Guide* for details.

CMEM (Contiguous memory)	Choose either 0 or 4 if your system has 4 Mbytes (or more) of contiguous memory available.
	To set CMEM on a SPARCstation 10, enter the following:
	`cd /usr/platform/SUNW,SPARCstation10,sx/sbin`
	The system must have 32 Mbytes of memory or more to set CMEM. For example:
	`# sxconfig -s 4 -1 28 -f`

VRAM (Video Random Access Memory)	Choose either 01418 video rand on access memory. CAUTION: If sxtest is run with its VRAM enabled, then frame buffer locking must be enabled or SunVTS results in errors.

sxtest Module Descriptions

The 13 sxtest modules are described in the table below.

TABLE 38-2 sxtest Modules

Module	Description
Display (module0)	Click Enable to display visual patterns. Three subtests call the SPAM library and display pictures to verify the integrity of a subset of the kernel and the SPAM libraries through the SPARCstation 10 SX video system. These routines are ported from the SPAM demonstration programs. All subtests in this module are skipped if the cg14 frame buffer does not exist, or if the VRAM is disabled. One of the following subtests displays screens between other subtests. This helps you determine whether the system is hung or if sxtest is running. rect_test: The screen is filled with random rectangles. The rectangles are drawn in CHUNKY_XBGR mode if 32-bit mode OpenWindows is running. If not, they are drawn in CHUNKY_C8 mode with the SPAM library routine sl_rect_fill_32. shaa: A picture of shaded lines is drawn in CHUNKY_BGR mode with the SPAM library routines sl_line_shaa_32, sl_span_load_8, and sl_rect_fill_8. NOTE: The shaa test is skipped if the test is running on an 8-bit window. lines — The screen is filled with lines of various colors. These lines are drawn in CHUNKY_XBGR mode if 32-bit mode OpenWindows is running; If not, they are drawn in CHUNKY_C8 mode with SPAM library routine sl_line_fill_8.

TABLE 38-2 `sxtest` Modules (Continued)

Module	Description
MUL (module1)	Click Enable to test the multiplier operations.

	Eight subtests are called, each having 2500 randomly generated MUL SPAM macros.
	`sp_mul0`
	`sp_mul1`
	`sp_mul2`
	`sp_mul3`
	`sp_mul4`
	`sp_mul5`
	`sp_mul6`
	`sp_mul7`

	Each subtest tests SPAM MUL instruction sets by running random SPAM MUL macro patterns, for example:
	`spam_dot(S_0,R42,R45,R31,5)`
	`spam_mulr(L_16,R44,R29,R52,1)`
	`spam_mul(S_15,R115,R114,R58,4)`
	`spam_mul(L_16,R89,R110,R81,8)`
	`spam_mulr(S_8,R21,R76,R53,1)`
	`spam_saxpr(S_8,R54,R46,R98,2)`
	`spam_dotr(L_16,R75,R40,R20,5)`
	`spam_dot(L_16,R44,R45,R84,4)`
	`spam_saxp(L_0,R93,R96,R44,8)`
	`spam_mulr(L_0,R86,R56,R56,5)`
	`spam_dotr(L_0,R14,R62,R40,2`
	`spam_saxpr(S_15,R112,R85,R95,7)`

ALU (Module2)	Click Enable to test ALU operations.

	Five subtests are called, each having 2500 randomly generated ALU SPAM macros.
	`sp_alu0`
	`sp_alu1`
	`sp_alu2`
	`sp_alu3`
	`sp_alu4`

	Each subtest tests SPAM ALU instruction sets by running random SPAM ALU macro patterns, for example:
	`spam_subv(R101,R31,R42,1)`
	`spam_subs(R90,R44,R90,14)`
	`spam_subv(R44,R70,R29,14)`
	`spam_sum(R58,R95,R114,9)`
	`spam_adds(R54,R46,R98,10)`
	`spam_addi(R9,51,R68,9)`
	`spam_abs(R76,R28,7)`
	`spam_addv(R80,R59,R93,11)`

TABLE 38-2 `sxtest` Modules (Continued)

Module	Description
ROP (Module3)	Click Enable to test the ROP operations.

	Five subtests are called, each having 2500 randomly generated ROP SPAM macros.
	`sp_rop0`
	`sp_rop1`
	`sp_rop2`
	`sp_rop3`
	`sp_rop4`

	Each subtest tests SPAM ROP instruction sets by running random SPAM ROP macro patterns, for example:
	`spam_selb(R101,R31,R42,1)`
	`spam_ropl(R90,R27,R44,14)`
	`spam_sels(R19,R16,R112,15)`
	`spam_ropm(R47,R29,R96,16)`
	`spam_selb(R52,R43,R29,5)`
	`spam_ropb(R115,R114,R58,7)`
	`spam_selv(R57,R75,R16,2)`
	`spam_ropm(R110,R93,R83,13)`

LOGIC (Module4)	Click Enable to test the logical operations.

	Five subtests are called, each having 2500 randomly generated LOGIC SPAM macros.
	`sp_logic0`
	`sp_logic1`
	`sp_logic2`
	`sp_logic3`
	`sp_logic4`

	Each subtest tests SPAM LOGIC instruction sets by running random SPAM LOGIC macro patterns, for example:
	`spam_xors(R101,R31,R42,1)`
	`spam_xori(R90,101,R90,14)`
	`spam_xorv(R30,R19,R95,13)`
	`spam_ands(R108,R16,R125,1)`
	`spam_andv(R115,R114,R58,7)`
	`spam_ors(R46,R89,R8,16)`
	`spam_orv(R57,R75,R16,2)`
	`spam_andi(R9,51,R68,9)`

TABLE 38-2 `sxtest` Modules (Continued)

Module	Description

SHIFT (Module5)	Click Enable to test the shift operations.
	
	Five subtests are called, each having 2500 randomly generated SHIFT SPAM macros.
	`sp_shift0`
	`sp_shift1`
	`sp_shift2`
	`sp_shift3`
	`sp_shift4`
	
	Each subtest checks SPAM SHIFT instruction sets by running random SPAM SHIFT macro patterns, for example:
	`spam_sllv(R101,R31,R42,1)`
	`spam_slli(R90,5,R90,14)`
	`spam_srai(R30,19,R95,13)`
	`spam_srli(R108,16,R125,1)`
	`spam_sllv(R52,R43,R29,5)`
	`spam_slfi(R46,25,R8,16)`
	`spam_slfs(R57,R75,R16,2)`
	`spam_srav(R54,R44,R93,8)`
	`spam_srlv(R58,R60,R96,16)`

COMP (Module6)	Click Enable to test the compare operations.
	
	Five subtests are called, each having 2500 randomly generated COMP SPAM macros.
	`sp_comp0`
	`sp_comp1`
	`sp_comp2`
	`sp_comp3`
	`sp_comp4`
	
	Each subtest checks SPAM COMP instruction sets by running random SPAM COMP macro patterns. For example:
	`spam_cmpv_gt(R101,R31,R42,1)`
	`spam_cmps_lt(R90,R44,R90,14)`
	`spam_cmps_eq(R95,R112,R19,12)`
	`spam_cmpv_gt(R44,R43,R29,14)`
	`spam_cmpv_lt(R115,R114,R58,7)`
	`spam_cmps_gt(R46,R89,R8,16)`
	`spam_cmps_eq(R57,R75,R16,2)`
	`spam_cmpv_le(R54,R46,R98,10)`
	`spam_cmpv_eq(R9,R51,R68,9)`
	`spam_cmps_gt(R76,R103,R28,7)`
	`spam_cmpv_eq(R52,R37,R50,8)`
	`spam_cmpv_ge(R61,R86,R16,12)`

TABLE 38-2 `sxtest` Modules (Continued)

Module	Description
MISC (Module7)	Click Enable to test the miscellaneous operations.

	Five subtests are called, each having 2500 randomly generated MISC SPAM macros.
	`sp_misc0`
	`sp_misc1`
	`sp_misc2`
	`sp_misc3`
	`sp_misc4`

	Each subtest checks SPAM MISC instruction sets by running random SPAM MISC macro patterns, for example:
	`spam_scat(R45,-1,R29,1)`
	`spam_gath(R95,-6,R114,9)`
	`spam_delt(R89,R9,R16,16)`
	`spam_plot(R54,R46,R98,10)`
	`spam_plot(R53,R20,R75,16)`
	`spam_scat(R91,-2,R70,9)`
	`spam_gath(R120,-2,R51,15)`
	`spam_delt(R59,R95,R120,1)`

MADR (Module8)	Click Enable to test the address lines of `sx`.

	Eight subtests are called; each subtest verifies the 0x100000 SPAM address with `spam_stld` and `spam_ldld` instructions. All address bits and data bits of 4 Mbytes of VRAM and 4 Mbytes of DRAM are tested after running through the eight subtests.
	`0x00000000-0x000fffff`
	`0x00100000-0x001fffff`
	`0x00200000-0x002fffff`
	`0x00300000-0x003fffff`
	`0xfc000000-0xfc0fffff  *`
	`0xfc100000-0xfc1fffff  *`
	`0xfc200000-0xfc2fffff  *`
	`0xfc300000-0xfc3fffff  *`

	* These subtests are skipped if the CMEM option is disabled.

TABLE 38-2   `sxtest` Modules (Continued)

Module	Description
SMCALL (Module9)	Click Enable for a brief test of `sxtest` functionality.
	Eleven subtests are called from `spam.smcall` to verify the general function of the SMC chip. All subtests have a cg14 version and a non-cg14 version. These subtests repeat four times, each time with the `IQ FIFO` programmed to a different number of entries (8, 16, 32, or 64). `shift_ldst` `instr_mix *` `arith_ldst` `cmp_ldst` `select_ldst` `interlock_all *` `logic_ldst` `mult_ldst` `rop` `scat_ldst` `delt_ldst` * These subtests are skipped if the VRAM option is set to disable.
MCNT (Module10)	Click Enable to test the load and store functions with different repeat counts.
	Twelve subtests are called; they test the SPAM store functions by varying address offset and item count. `spsd_stba_cnt` `spsd_stbd_cnt` `spsd_stbds_cnt` `spsd_stcd_cnt` `spsd_stla_cnt` `spsd_stld_cnt` `spsd_stlds_cnt` `spsd_stpd_cnt` `spsd_stqd_cnt` `spsd_stsa_cnt` `spsd_stsd_cnt` `spsd_stsds_cnt`

TABLE 38-2 `sxtest` Modules (Continued)

Module	Description
GRIF (Module11)	Click Enable to test the graphic interface logic.

	Thirty-six subtests are called; they test the SPAM graphic interface login with load/store instructions. All subtests are skipped if `cg14` does not exist.
	`spsd_stbd_dram`
	`spsd_stbd_xbgr`
	`spsd_stbd_bgr`
	`spsd_stbd_8x`
	`spsd_stbd_8c`
	`spsd_stbd_x32`
	`spsd_stbd_b32`
	`spsd_stbd_g32`
	`spsd_stbd_r32`
	`spsd_stsd_dram`
	`spsd_stsd_xbgr`
	`spsd_stsd_bgr`
	`spsd_stsd_8x`
	`spsd_stsd_8c`
	`spsd_stsd_x32`
	`spsd_stsd_b32`
	`spsd_stsd_g32`
	`spsd_stsd_r32`
	`spsd_ldbd_dram`
	`spsd_ldbd_xbgr`
	`spsd_ldbd_bgr`
	`spsd_ldbd_8x`
	`spsd_ldbd_8c`
	`spsd_ldbd_x32`
	`spsd_ldbd_b32`
	`spsd_ldbd_g32`
	`spsd_ldbd_r32`
	`spsd_ldsd_dram`
	`spsd_ldsd_xbgr`
	`spsd_ldsd_bgr`
	`spsd_ldsd_8x`
	`spsd_ldsd_8c`
	`spsd_ldsd_x32`
	`spsd_ldsd_b32`
	`spsd_ldsd_g32`
	`spsd_ldsd_r32`

TABLE 38-2 `sxtest` Modules (Continued)

Module	Description
REGF (Module12)	Click Enable to test the register file pointer logic.
	Twenty-two subtests are called from `spam.regfile` to verify the register file's logic with assorted SPAM instructions.
	`readpointer1` *
	`readpointer2` *
	`readpointer3` *
	`readpointer4` *
	`writepointer1` *
	`writepointer2` *
	`writepointer3` *
	`writepointer4` *
	`readpointer5` *
	`writepointer5` *
	rdptr0 †
	wrptr0 †
	rdptr1 †
	wrptr1 †
	rdptr2 †
	wrptr2 †
	rdptr3 †
	wrptr3 †
	rdptr4 †
	wrptr4 †
	rdptr5 †
	wrptr5 †
	* These subtests are skipped if the VRAM option is disabled.
	† These subtests are skipped if the CMEM option is disabled.

`sxtest` Test Modes

Due to the nature of graphic tests, reading from or writing to the frame buffer during testing will disturb user production. `sxtest` is available only in the Functional test mode.

In Functional test, `sxtest` runs the visual subtest, memory address subtest, memory count subtest, graphic interface subtest, and register file subtest by default. The user can select or de-select other subtests as needed.

`sxtest` Command Line Syntax

/opt/SUNWvts/bin/sxtest *standard_arguments* **-o dev=***device_name***,
lock=***E(nable)/D(isable)***,tm=***to_module#***,cmem=***n***,vram=***n***,md=**XXX

TABLE 38-3   `sxtest` Command Line Syntax

Argument	Explanation
`dev=device_name`	Specifies the `device_name` to be tested.
`lock`**=***E(nable)/* *D(isable)*	Enables/disables frame buffer locking. See "Testing Multiple Frame Buffers" in Chapter 3 of the *SunVTS 3.0 User's Guide* for details. Frame buffer locking is enabled by default on the window server running the OpenWindows software.
`tm`**=***to_module#*	Specifies an ending module number; use these last four arguments to narrow `sxtest` to a specific test scope.
`cmem=`*0/4*	Enables or disables the contiguous memory; choose either 1 to enable or 0 to disable. Note: You must choose disable (0) if your system is equipped with less then 4 Mbytes of contiguous memory.
`cgfourteen0=`*0/4* */8*	Enables or disables the video random access memory; choose 1 to enable or 0 to disable.
`md=`**vis+alu+...**	Selects which modules are tested in a pass.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname.* If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`sxtest` Error Messages

TABLE 38-4   `sxtest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6002	`Unable to lock frame buffer`	FB lock was enabled while Window System was not running.	
6004	`Can`t get sem` *number*		
6006	`Failed lock sem:dev` *number*, `type` *number*		
6008	`Failed unlock sem`		
6010	`Failed GETVAL sem=`		
6012	`Failed sem_val`		
6014	`console_p() failed:`		
6016	`Failed sem_setall`		
6017	`Failed sem_rmid`		
6018	`Failed get sem`		
6020	`ioctl command` *name* `failed`		
6022	*error_message*		
6024	`Failed to malloc for` *name*		
6026	`mem ccitt[`*name*+*number*`,`*name*+*number*`]: o:`*number* `e:`*number* `o^e:`*number*		
6028	`mem crc16[`*name*+*number*`,`*name*+*number*`]: o:`*number* `e:`*number* `o^e:`*number*		
6030	`reg ccitt: o:`*number* `e:`*number* `o^e:`*number*		
6032	`reg crc16: o:`*number* `e:`*number* `o^e:`*number*		
6034	`Failed to open` *name* `device`		
6035	`First Open of sx device`		
6036	`Modules selected improperly!!!`		

TABLE 38-4 `sxtest` Error Messages *(Continued)*

	Error Message	Probable Cause(s)	Recommended Action
6038	*name*, i_module = *name*		
6040	FB unlock request failed		
8000	Can't acquire console semaphore	System load too heavy.	Reduce the system load.
		System error.	If the problem persists, call your authorized Sun service provider.
8002	Can`t get sem *number*		
8004	Unable to set page bounds		
8006	mmap: unable to map MDI control register space		
8008	Failed to mmap *name*		
8010	sxtest is only running in Functional test		
8012	Failed to open *name*		

System Test (`systest`)

`systest` checks the CPU board by exercising the I/O, memory, and CPU channels simultaneously as threads. There is no quick test option for `systest`; it is a CPU stress test.

`systest` Options
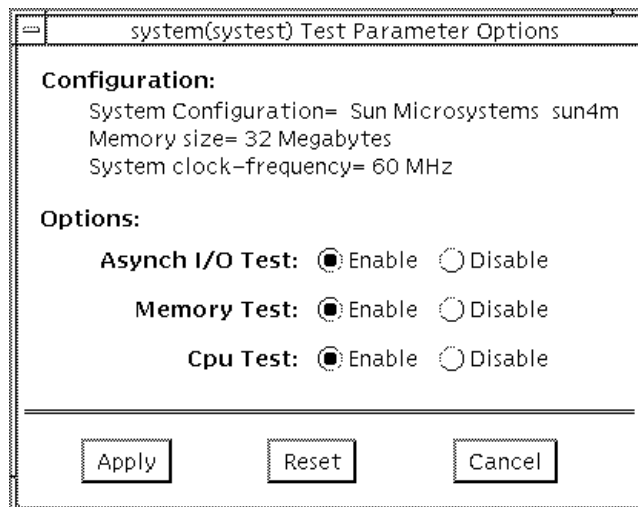
FIGURE 39-1 `systest` Option Menu

`systest` Test Modes

The Functional test is the only test mode available for `systest`.

`systest` Command Line Syntax

/opt/SUNWvts/bin/systest *standard_arguments*

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

---

systest Error Messages

TABLE 39-1   systest Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6005	(*thread_id*)Couldn't open file / dev/mem: *error_message*		
6006	(*thread_id*)'valloc' results incorrect: *error_message*		
6007	(*thread_id*)Bad 'mmap' results: *error_message*, page = *value*/*value*, addr = *address*		
6008	(*thread_id*)kvm_open() failed: *error_message*		
6009	(*thread_id*)kvm_nlist() failed: *error_message*		
6010	(*thread_id*)Defective namelist in '/vmunix'		
6011	(*thread_id*)Couldn't read physical memory list: *error_message*		
6012	(*thread_id*)Couldn't read physical memory address		
6013	(*thread_id*)Couldn't read physical memory size: *error_message*		
6014	Cannot open *device_name*		
6015	Test Failed due to segment violation error	Lack of sufficient memory resources.	Retry the test after increasing available memory.
6016	Test Failed due to fpu bus error	Lack of sufficient memory resources.	Retry test after increasing available memory.
6017	Test Failed due to fpu exception error	Lack of sufficient memory resources.	Retry test after increasing available memory.
6018	Test Failed due to illegal instruction	Lack of sufficient memory resources.	Retry test after increasing available memory.
8000	Failed system test (FPU, VMEM, I/O)		

Tape Drive Test (`tapetest`)

The `tapetest` synchronous I/O test writes a pattern to a specified number of blocks (or, for a SCSI tape, writes to the end of the tape). The `tapetest` then rewinds the tape and reads and compares the data just written. The `tapetest` asynchronous I/O test sends a series of up to five asynchronous read/write requests to the tape drive, writing to the tape and then reading and comparing the data. The `tapetest` file test writes four files to the tape and then reads them back, comparing the data. For tape library testing, the pass count is incremented only after all tapes in the library have been tested.

Note – If you have a tape drive in your system, load a blank writable tape (scratch tape) before you start SunVTS. If you fail to do this, the `tapetest` option displays `drive type:unknown` on the option menu for the `tapetest`.

`tapetest` Options

`tapetest` supports 4-mm, 8-mm, DLT, 1/4-inch cartridge, and 1/2-inch front-load tape drive testing. The options available for each of the tape devices differ slightly. An example of the option pop-up menu for a device is shown in FIGURE 40-1.

The Async I/O subtest uses the asynchronous read and write feature of the Solaris tape driver to exercise tape drives. In read-only mode the test sends a maximum of four asynchronous read packets, each with a random size and a random offset, to the tape drive. The test then waits for all outstanding I/O activity to complete before issuing another round of packets. This process continues until the whole area being tested has been covered. In read-write mode, one write packet is issued for every four read packets to ensure a spot check of the write operation. The area of the tape to be tested is written to first in order for the test to work correctly. This test is only supported under Solaris 2.6, Solaris 7 and compatible releases.
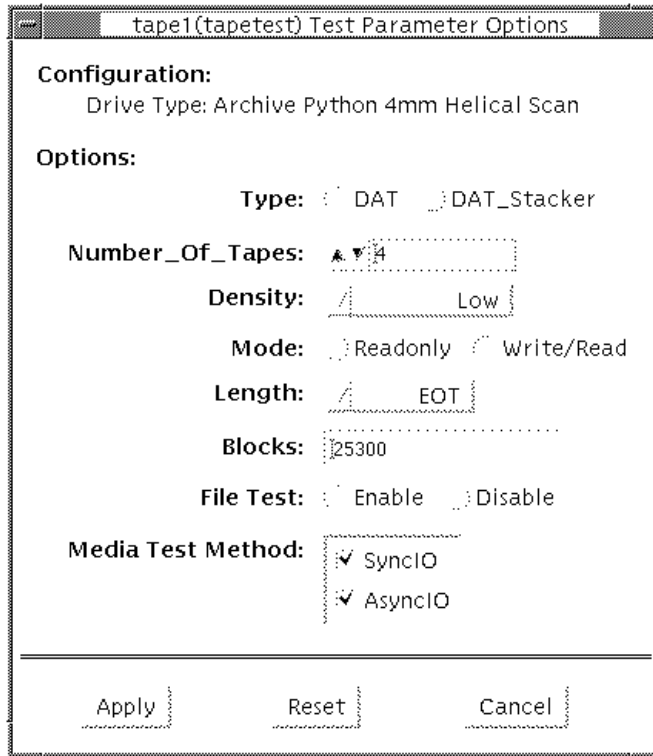
FIGURE 40-1 `tapetest` Option Menu (4-mm tape drives)

Note – This test does not immediately stop after being disabled.

FIGURE 40-1 shows an example of the options menu for a 4-mm tape drive.

Note – The option menus for the 1/4-inch, 1/2-inch, DLT, and 8-mm tape drives differ slightly from FIGURE 40-1.

TABLE 40-1 `tapetest` Options

`tapetest` **Options**	**Description**
Type	Normal tape drive or tape library (stacker).
# of Tapes	The number of tapes in the tape library. `tapetest` registers a single tape library pass only after all tapes in the library pass.
Density	The following settings are available for most tape drives: -Low: Tests the `l` tape device. -Medium: Tests the `m` tape device. -Compression: Tests the `c` tape device. -All: Tests the `l`, `m`, and `c` tape devices. For half-inch tape drives, the available settings are 800, 1600, and 6250 BPI (blocks per inch). For certain QIC drives select QIC-11 (1-byte block ID) mode, QIC-24 (4-byte block ID) mode, or both.
Mode	If you enable Write/Read mode, the test first writes to the tape and then reads it back to compare. If you enable Read_Only mode, the test assumes the tape has been properly written and merely reads and compares. This mode is useful to check proper head alignment.
Length	The amount of the tape to be tested. The choices are: `-EOT`: The default; tests to the entire tape. `-Long`: The SCSI tape tests 70,000 blocks of the tape. `-Short`: Only the first 1000 blocks are tested. `-Specified`: You must type the number of blocks to be tested in the `# of blocks` field.
# of Blocks	If you select Specified under the Length option, you must type the number of blocks you want to test.
Blocksize	Block size specification. This option is only available for Tandberg QIC tape drives. There are two possible values; 512-bytes for use with older tape media that have transfer size restrictions, or 64-kbytes for use with current, high-capacity tape media.

TABLE 40-1 `tapetest` Options

`tapetest` **Options**	**Description**
File Test	The tape file test sequence is as follows: -Writes three files -Rewinds -Reads part of the first file -Forward spaces to the start of the second file -Reads the second file -Forward spaces to the start of the third file -Tries to read to the end of that file for SCSI tapes only, the tape file test tries to backspace to the start of the second file and read it
Retension	When enable is selected, the program retensions the tape.
Media Test Method	Sync I/O: tapetest reads and or writes the number of blocks selected in Length. Async I/O: tapetest makes four asynchronous read requests to the tape drive. If read and write testing is selected, one asynchronous write request is also sent. The test continues after completing the requests. Note – When testing Tandberg QIC drives, Async I/O testing is restricted to read-only due to asynchronous behavior differences with other tape drives.

`tapetest` Test Modes

The `tapetest` supports all three modes. It performs different test schemes on the tape device, according to the mode you select.

TABLE 40-2 `tapetest` Test Modes

Test Mode	**Description**
Connection Test	In this mode, `tapetest` verifies that the drive can be opened and that the drive type can be determined. If both checks are successful, or if the drive is currently busy, then the test passes. The `tapetest` fails if the open operation is unsuccessful for any reason other than the drive is busy.
Functional Test	`tapetest` checks the status, rewinds the tape, erases and retensions it. If the device is a cartridge tape, `tapetest` writes a pattern to `nblks` or `eot` (default), rewinds the tape, and then reads and compares of the pattern. If you invoke SunVTS through SyMON, the following Functional test description applies: `tapetest` opens the device and reads several blocks from it. If the read operation is successful, the test passes. On the other hand, if the device is busy or if no tape cartridge can be found in the drive, the test cannot run and fails.

tapetest Command Line Syntax

`/opt/SUNWvts/bin/tapetest` *standard_arguments* `-o dev=`*device_name*`,s=`*block_count*`,d=`*density*`,m=`*mode*`,l=`*length*`,method=method,ft=`*enables/disables*`,ret=`*enables/disables*`,dat=`*dat_type*`,8mm=`*8mm_type*`,num=`*magazine_size*`,blocksize=`*block_size*

TABLE 40-3 `tapetest` Command Line Syntax

Argument	Explanation
dev=_device_name_	Specifies the *device_name* of the tape drive (required).
s=_block_count_	Specifies the number of blocks to be tested.
d=_density_	Specifies the density of the tape to open.
m=_mode_	Enables either the Write_Read or Read_Only tests.
l=_length_	Specifies the length of the test (EOT, Specified, Long, or Short).
method=_method_	Specifies the media test method (SyncI/O and or AsyncI/O) used.
ft=_enables/disables_	Enables or disables the file test.
ret=_enables/disables_	Enable or disables tape retention.
dat=_dat_type_	If you are testing a digital audio tape drive, specify whether it is a regular DAT drive or a DAT stacker. The choices are **DAT** and **DAT_Stacker**.
8mm=_8mm_type_	If you are testing an 8-mm tape drive, specify whether it is a regular 8-mm tape drive or a tape library. The command line choices are **8mm** and **8mm_Library**.
num=_magazine_size_	If you are testing a tape library, specify the magazine size.
blocksize=_block_size_	If you are testing a Tandberg QIC drive, specify whether to use a 64 kbyte block transfer or a 512 byte block transfer. Use 512 bytes when testing older media in the drive.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`tapetest` Error Messages

TABLE 40-4   `tapetest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	`Missing device name`	No device name is specified.	Retry test with correct device name specification.
6001	`failed ioctl on unload and load next tape: err_code =` *number*`!`	This stacker/library unit may lack the Automatic Cycle option.	Run the test with Single Pass Option enabled.
6002	`Cannot open` *device_name*`:` *error_message*	No tape media in drive.	Load tape media and retry test.
6005	`Cannot close` *device_name*`:` *error_message*	Bad drive.	Retry test; if problem persists, check or replace the drive.
6006	*text* `write failed on` *error_message*`, block` *number*`: EOF reached`	EOF reached prematurely.	Retry test with reduced blocksize specification or until EOF.
6007	*text* `write failed on` *device_name*`, block` *number*`:` *error_message*`, sense key(0x`*value*`) =` *text*	Drive needs attention.	Perform action to reset sense key and retry the test.
6008	*text* `read failed on` *device_name*`, block` *number*`: EOF reached`	EOF reached prematurely.	Retry test with reduced blocksize specification or until EOF.
6009	*text* `read failed on` *device_name*`, block` *number*`:` *error_message*`, sense key(0x`*value*`) =` *text*	Drive needs attention.	Perform action to reset sense key and retry test.
6010	*text* `compare failed on` *device_name*`, block` *number*`, offset` *number*`, pattern 0x`*value*`, data= 0x`*value*	Bad drive.	Retry test; if problem persists, check or replace the drive.
6011	*device_name* `tape MTIOCGETDRIVETYPE ioctl:` *number*	Bad drive.	Retry test; if problem persists, check or replace the drive.
6012	*text* `failed on` *device_name*`:` *error_message*	Bad drive.	Retry test; if problem persists, check or replace the drive.
6013	`Failed Connection test on` *device_name*`:`*error_message*	Broken cable.	Check or replace the cable.
		Bad drive.	Check or replace the drive.

TABLE 40-4 `tapetest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6014	Failed Online test:*device_name*	Device may be in use.	Retry later.
6015	Failed Online test:*device_name*:*error_message*	No tape media in drive.	Load tape media and retry test.
6016	Online test was unsuccessful:*device_name*	No tape media in drive.	Load tape media and retry test.
6017	Failed Online read test:*device_name*	Bad drive.	Retry test; if problem persists, check or replace the drive.
6018	Open Failed on *device_name*: *error_message*	Broken cable.	Check or replace the cable.
		Bad drive.	Check or replace the drive.
6018	recon: No Corresponding SCSI disk device		
6019	Needs both SCSI disk and SCSI tape to run the test		
6020	Couldn't retension '*device_name*'		
6021	Couldn't read file '*device_name*'		
6022	SCSI disconnect/ reconnect failed		
6023	Spurious signal received from child		
2005	Tandberg QIC Read/Write media failure.	Incorrect test settings for older QIC tape media.	Retry the test and specify a 512-byte blocksize. Retry the test with a read-only setting. Use the highest capacity tape media that the drive will allow (refer to tape drive documentation for QIC media compatibilities.).

S24 Frame Buffer Test (`tcxtest`)

Through a series of protocol, memory, acceleration, and colormap tests, `tcxtest` checks the functionality of the S24 Frame Buffer SBus card used on the SPARCstation5 and checks the FSV (fast SBus video) ASIC on the SPARCstation4 motherboard.

Note – Disable all screen savers before testing any graphics device. Type **xset s off** at a UNIX prompt to disable the Solaris screen saver.

Test Groups

`tcxtest` has four distinct test groups:

- AFX Protocol tests (in 8/16/32/64-bit mode)
- `WRC`
- Frame Buffer Memory tests (in 8/16/32/64-bit14 mode)
- `address`
- `constant`
- `random`
- Acceleration tests (both User and Raw modes)
- `blit`
- `stip`
- Colormap and Cursor tests
- `cursor` (does not apply to SPARCstation4)
- `colormap`

`tcxtest` Subtests

TABLE 41-1   `tcxtest` Subtests

tcxtest Subtests	Description
WRC	By performing multiple writes and reads, and then verifying the results, the WRC test exercises the FIFO inside the S24 chip. The WRC test is composed of these three subtests: `test_afx_alt_wr, test_memafx, and test_afx_random.` If these tests fail, they print an error message showing the expected and observed data.
Test_afx_alt_wr	This test performs 16 writes to alternative pages (for example: WR (Page1), WR (Page2), WR (Page1+off), WR (Page2+off), etc.). It then reads back the data and compares it with the expected results. This test also writes to the frame buffer space 16 times, followed by a write to a different page in the frame buffer space. The test then reads this data back and verifies it with the expected results.
Test_memafx	The CPU in the SWIFT chip has closely coupled interfaces for the DRAM and the AFX bus. This test checks the arbitration between the two accesses. This test performs a number of alternating writes to the AFX and the CPU memory. After writing to different locations, the test reads and verifies the data. By performing an access across the page boundaries, the test covers both the cached and non-cached accesses.
Test_afx_random	After writing to one page in the DRAM memory, the test performs a few random writes/reads to random locations in the AFX space. The test then writes to a different page in the DRAM space, where it performs random accesses. This test does not perform any data verification, it just checks to see if any of these random accesses caused a time out.
Constant	This test writes a data pattern to the whole memory. This pattern is read back and compared with the expected data. Once the memory fill operation is completed, the test reads the memory back and verifies that the value read is correct.
Address	This test writes a data pattern (which is same as the value of the address) to the whole memory. This pattern is then read back to verify that it is the correct value.
Random	This test writes a random data pattern to the whole memory. This pattern is read back and compared with the expected data. After the memory fill operation is completed, the test reads the memory and verifies the values read are correct.

TABLE 41-1 `tcxtest` Subtests

tcxtest Subtests	Description
Blit	This test has two parts; the raw `blit` test and the user `blit` test. The raw `blit` test draws a 64x64x24 pixel image at the top-left corner of screen. Next it `blits` the image to the screen. The destination images are read back and compared with the original image to verify the raw `blit` operation has run correctly. The user `blit` test draws a 64x64x24 pixel image at the top left corner of screen. It then `blits` the image to the screen. The destination images are read back and compared with the original image. The user `blit` test is the same as the raw `blit` test, except the user `blit` test uses the user data space for the `blit` command.
Stip	This test performs numerous corner cases for stipple. The test writes to the destination with different data values using a stipple operation. The destination data is read back and verified. For the fast SBus video (SV), the following is checked: • Walks 1 through pixel mask • Walks 1 through ROP bits • Walks 1 through destination byte • Walks 1 through IDX byte
Cursor (does not apply to SPARCstation 4)	This test performs a data register regression test. It writes a walking 1 pattern to the cursor data registers. The data is then read back and verified with the expected results. The test is repeated using a walking 0 as the data pattern
Colormap	Loads all locations in the colormap with varying values of RGB. Note: If the system being tested has a monochrome or greyscale monitor, visual color problems are undetectable.

`tcxtest` Options

FB Locking is the only test parameter options for this test. See the section about testing multiple frame buffers in the *SunVTS 3.0 User's Guide* for details.

● **Click enable or disable to configure Frame Buffer locking.**
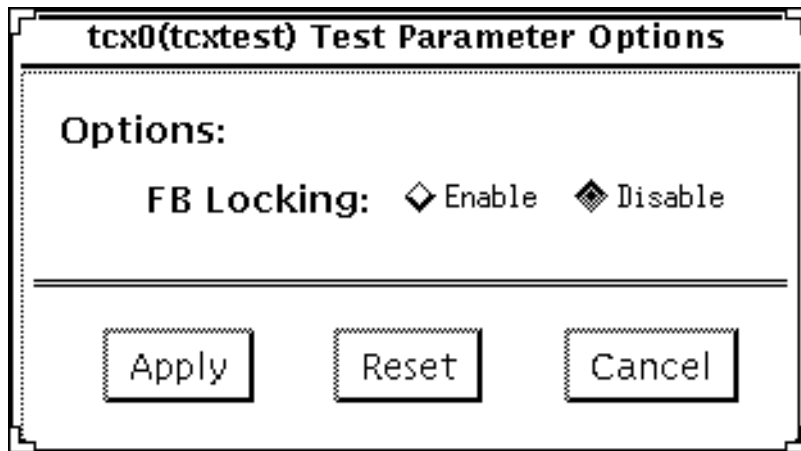
FIGURE 41-1 `tcxtest` Option Menu

`tcxtest` Test Modes

Available only in the Functional test mode. Functional test is the default for tcxtest. `tcxtest` runs the appropriate subtests for the hardware platform being tested.

`tcxtest` Command Line Syntax

`/opt/SUNWvts/bin/tcxtest` *standard_arguments* `-o`
`dev=`*device_name*`,lock=E(nable)/D(isable),` `X=`*bit_mode*`,T=`*test*`,S=`[dfb8, dfb24, dfb32]

TABLE 41-2   `tcxtest` Command Line Syntax

Argument	Explanation
dev=*device_name*	Specifies the filename of the device to be tested, for example: **dev=tcx0.**
lock=*E(nable)/ D(isable)*	Enables or disables the window system locking option. See the section about Testing Frame Buffers in *SunVTS 3.0 User's Guide* for details. Do not use when device is the window system display.
X=*bit_mode*	Specifies the data transfer size; supported values are: **8**byte **16**short **32**long **64**double word
T=*test*	Specifies a particular test; to specify an individual test, replace *test* with: a=Address c=Constant r=Random b=Blit s=Stipple h=Cursor w=WRC Note: When you select either the `Blit` or `Stipple` test, both the User and Raw mode tests are run.
S=*[dfb8, dfb24, dfb32]*	Specifies which frame buffer memory space to use. `-dfb8`Dumb frame buffer 8-bit space. Memory is accessed only by bytes. `-dfb24`Dumb frame buffer 24-bit space. Memory is accessed only by 24-bit reads and writes. `-dfb32`Dumb frame buffer 8-bit space. Memory is accessed by 8-bit reads and writes.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`tcxtest` Error Messages

TABLE 41-3   `tcxtest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6000	`FBIOGATTR failed in is_24bit subroutine`	Driver not installed.	Examine system message files (`/var/adm/messages`) for other information.
		OS driver problem.	
		AFX bus error.	
6010	`Colormap failure, Address:` *address*`, red, Expected:` *value* `Observed:` *value*	Bad RAMDAC.	Replace the frame buffer card.
		Bad a24 board.	Examine system message files (`/var/adm/messages`) for other information.
		AFX bus error.	
6011	`Colormap failure, Address:` *address*`, green, Expected:` *value* `Observed:` *value*	Bad RAMDAC.	Replace the frame buffer card.
		Bad a24 board.	Examine system message files (`/var/adm/messages`) for other information.
		AFX bus error	
6012	`Colormap failure, Address:` *address*`, blue, Expected:` *value* `Observed:` *value*	Bad RAMDAC.	Replace the frame buffer card.
		Bad a24 board.	Examine system message files (`/var/adm/messages`) for other information.
		AFX bus error.	
6013	`THC Cursor Regression failed Address:` *address*`, Expected:` *value*`, Observed:` *value*	Bad a24 board.	Replace the frame buffer card.
		AFX bus error.	Examine system message files (`/var/adm/messages`) for other information.

TABLE 41-3   tcxtest Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6014	Can't clear frame buffer, dst_start: *address*, src_start: *address*, Address: *value*, Observed: *value* Expected: *value*	Bad a24 board.	Replace the frame buffer card.
		AFX bus error.	Examine system message files (/var/adm/messages) for other information.
6015	Frame buffer pattern test failed, dst_space: *address*, src_space: *address*, check_x = *value*, check_y = *value*, Address: *value*, Observed: *value* Expected: *value*	Bad a24 board.	Replace frame buffer card
		AFX bus error.	Examine system message files (/var/adm/messages) for other information.
6016	Can't clear frame buffer pattern, dst_space: *address*, src_space: *address*, check_x: *value*, check_y: *value*, Address: *value* Observed: *value* Expected: *value*	Bad a24 board.	Replace the frame buffer card.
		AFX bus error.	Examine system message files (/var/adm/messages) for other information.
6017	Can't clear frame buffer after write, Address: *value*, Expected: *value* Observed: *value*	Bad a24 board.	Replace the frame buffer card.
		AFX bus error.	Examine system message files (/var/adm/messages) for other information.
6018	Raw Blit test failed, Offset: *address*, Expected: *value* Observed: *value*	Bad a24 board.	Replace the frame buffer card.
		AFX bus error.	Examine system message files (/var/adm/messages) for other information.
6019	Blit test failed CHECKING, Offset: *address*, Expected: *value* Observed: *value*	Bad a24 board.	Replace the frame buffer card.
		AFX bus error.	Examine system message files (/var/adm/messages) for other information.
6020	Blit test failed, Offset: *value*, Expected: *value* Observed: *value*	Bad a24 board.	Replace the frame buffer card.
		AFX bus error.	Examine system message files (/var/adm/messages) for other information.
6021	Blit accel test failed. Offset: *value*, Expected: *value* Observed: *value*	Bad a24 board.	Replace the frame buffer card.
		AFX bus error.	Examine system message files (/var/adm/messages) for other information.

TABLE 41-3 `tcxtest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6022	Stipple test failed. Offset: *address*, Expected:*value* Observed: *value*	Bad a24 board.	Replace the frame buffer card.
		AFX bus error.	Examine system message files (/var/adm/messages) for other information.
6025	Raw Stipple test failed. Offset: *address*, Expected:*value* Observed: *value*	Bad a24 board.	Replace the frame buffer card.
		AFX bus error.	Examine system message files (/var/adm/messages) for other information.
6027	Frame buffer pattern test miscompare, y: *value* x: *value* Observed: *value* Expected: *value* Address: *value*	Bad RAMDAC.	Replace the frame buffer card.
		Bad a24 board.	Examine system message files (/var/adm/messages) for other information.
		AFX bus error.	
6029	Afx protocol test *message* Expected: *value*	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	
6030	Null allocation to membase	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	
6031	Afx_mem test failure, Address: *address*, Expected: *value*	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	
6032	Afx_mem Mem test failure, Address+0x4000 Value: *value*, Exp 0x5555aaaa	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	

TABLE 41-3 `tcxtest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6033	Pattern test failed in Byte mode for DFB*value*. Offset: *address*, Expected: *value* OBserved: *value* xor(*value*)	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	
6034	Pattern test failed in Short mode for DFB*value*. Offset: *value*, Expected: *value* OBserved: *value* xor(*value*)	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	
6035	Pattern test failed in Long mode for DFB*value*. Offset: *value*, Expected: *value* OBserved: *value* xor(*value*)	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	
6036	Pattern test failed in Long mode for DFB*value* Offset: *value*, Expected: *value* OBserved: *value*	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	Examine system message files (/var/adm/messages) for other information.
6039	Random test failed in Byte mode for DF*value*, Offset: *value*, Expected: *value* Observed: *value* xor: *value*	AFX bus error.	
		Bad a24 board.	
		Bad CPU board.	
6040	Random test failed in Short mode for DFB*value*, Offset: *value*, Expected: *value* Observed: *value* xor: *value*	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	
6041	Random test failed in Long mode for DFB*value*, Offset: *value*, Expected: *value* Observed: *value* xor: *value*	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	

TABLE 41-3 `tcxtest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
6042	Pattern test failed in Dblword mode for DFB*value*, Offset: *value*, Expected: *value* Observed:*value*, xor:*value*	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	
6043	Address test failed in Char mode for DFB*value*. Offset: *value*, Expected: *value* Observed: *value* xor: *value*	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	
6044	Address test failed in Short mode for DFB*value*, Offset: *value*, Expected: *value* Observed: *value* xor: *value*	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	
6045	Address test failed in Long mode for DFB*value*, Offset: *value*, Expected: *value* Observed: *value* xor: *value*	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	
6046	Address test failed in Dblword mode for DFB*value*, Offset: *value*, Expected: *value* Observed: *value* xor: value>	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	
6047	Verifying Double:word writes	AFX bus error.	Examine system message files (/var/adm/messages) for other information.
		Bad a24 board.	
		Bad CPU board.	
8000	Open failure for device *device name*	Driver not installed.	Examine system message files (/var/adm/messages) for other information.
		OS driver problem.	
		AFX bus error.	

TABLE 41-3 `tcxtest` Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8001	Pattern test failed in Long mode, Offset: *value*, Expected: *value* Observed: *value*	Driver not installed.	Examine system message files (/var/adm/messages) for other information.
		AFX bus error.	
		Bad CPU board.	
8002	can't acquire console semaphor	Not enough memory.	Examine system message files (/var/adm/messages) for other information.
		Not enough swap space.	
8003	XCreateSimpleWindow failed	Not enough memory.	Examine system message files (/var/adm/messages) for other information.
		Not enough swap space.	
8004	Multibuffering extension does not exists	Not enough memory.	Examine system message files (/var/adm/messages) for other information.
		Not enough swap space.	
8005	Couldn't create enough buffers	Not enough memory.	Examine system message files (/var/adm/messages) for other information.
		Not enough swap space.	

Virtual Memory Test (`vmemtest`)

The `vmemtest` checks virtual memory; that is, it tests the combination of physical memory and the swap partitions of the disk(s).

Note – This test may not stop immediately after being disabled.

This test uses the Solaris `valloc` (page aligned) system call to allocate, write, read, and compare virtual memory. These operations normally cause heavy paging activity on the system and simulates a stressful environment for the operating system. This test also detects ECC parity errors, memory read errors, addressing problems, and displays the corresponding virtual memory addresses on failure.

Handling Swap Space

Running this test places a significant burden on the operating system, since it uses the majority of swap space available for testing. You should use the `vmemtest` swap space reserve option when non-SunVTS test processes are started after SunVTS testing has started. See "Swap Space Requirements" in the *SunVTS User's Guide* for a complete discussion of swap space requirements.

`vmemtest` Options

FIGURE 42-1 `vmemtest` Option Menu

TABLE 42-1 `vmemtest` Options

vmemtest Options	Description
vmemtest Configuration	The amount of memory listed in the configuration field is equivalent to the sum of the used and available swap space amounts returned by the `swap -s` command. It indicates the amount of virtual memory found, rounded up to the nearest Kbyte.
Test Amount	An amount can be specified to test the virtual memory, instead of the default. Specifying a number greater than the available memory, can cause `vmemtest` or other running tests to fail.
Mode	Two modes are available: • Regular mode tests the amount of memory and is limited by the amount of physical memory available to the system under test. • Page mode tests assign virtual memory one page at a time. Each page is mapped to the temporary file `/tmp/vmem.page` and is then paged out to storage once test data is written. Next, the temporary page is paged back into memory for a read and compare. When the stress test execution option is selected, the `vmemtest` allocates the entire assigned system memory (through `valloc`), writing from beginning to end. The memory is then read back and compared with the original pattern, one long word at a time.
Reserve	The Reserve option specifies the amount of memory to reserve from being tested by `vmemtest`. The reserved space is used for other processes running concurrently with the or SunVTS tests. The Reserve option can be used to reserve memory in addition to the default. This option applies only to a specific instance. Trying to reserve more memory than what is assigned to be tested by this instance will cause the test to fail.

`vmemtest` Test Modes

This test is only available in the Functional test mode. In this mode the, `vmemtest` writes a pattern to an amount of virtual memory specified by the user. Then the data is read back and compared. If there is a miscompare, the data is read again and

compared. Whenever there is a miscompare, the virtual address is reported. When there is a miscompare on recomparison, an attempt is made to convert the virtual address to the physical address if the SunVTS diagnostic driver is installed.

`vmemtest` Command Line Syntax

/opt/SUNWvts/bin/vmemtest *standard_arguments* **-o mode=***type***, reserve=***number***,amount=***number***,bdinfo=***number***,cerr=***number*

TABLE 42-2 `vmemtest` Command Line Syntax

Argument	Explanation
mode=*type*	Specifies which mode of the `vmemtest` you want to run; you can choose between `page` mode, which tells the write/read memory test to proceed one system memory page at a time, and `regular`, which `valloc`'s the entire assigned memory which is read and compared, one long word at a time.
reserve=*number*	Specifies the amount of Mbytes of virtual memory to reserve in addition to the default amount.
amount=*number*	Specifies the number of Mbytes of memory to be tested instead of the default amount.
bdinfo=*number*	Provides the board number information for all the CPU/memory boards in the system. For example, if board 0 and board 5 have memory, then the `bdinfo`=33 (2**5+2**0).
cerr=number	Specifies the maximum number of contiguous errors to be dumped when a memory error occurs.

Note – 64-bit tests are located in the `sparcv9` subdirectory: `/opt/SUNWvts/bin/sparcv9/`*testname*. If a test is not present in this directory, then it may only be available as a 32-bit test. For more information refer to "32-Bit and 64-Bit Tests" on page 5.

`vmemtest` Error Messages

TABLE 42-3 `vmemtest` Error Messages

	Error Message	Probable Cause(s)	Recommended Action
6001	`setrlimit(RLIMIT_DATA)` `failed to set to` `RLIM_INFINITY:` *error_message*	Failed to increase the size of test process's heap to 2GB (fatal error).	
6002	`Test terminated after` `finding maximum number` `(`*number*`) of noncontiguous` `errors`	Fatal error.	
6003	`msync() failed:` *error_message*	Failed to synchronize memory with physical storage (fatal error).	
6004	`Test terminated after` `finding miscompared data`	Test failed (fatal error).	
8001	`Test completed. Found` *data* `noncontiguous miscompare` `errors`	Suspect swap partition(s) (fatal error).	Consult your authorized Sun service provider.
		Suspect SCSI controller.	
		Faulty system software.	
8002	`Increase swap space to` `allocate at least` *number* `MB from the process'` `heap`	Failed to increase `mem` process' heap storage.	
8003	`Attempt to reserve more` `than what's been assigned`	Fatal error.	Reduce the "reserve" amount.
8004	`valloc(`*address*`) failed:` *error_message*	Fatal error.	Increase swap space or reduce the reserve amount.
8005	`Open(`*file_name*`) failed:` *error_message*	Failed to open the file *file_name* used in the "page" mode (fatal error).	
8006	`mmap() virtual address` *address* `to` *file_name* `failed:` *error_message*	Failed to map a region at the virtual address of the file *file_name* used in the "page" mode (fatal error).	

TABLE 42-3 vmemtest Error Messages (Continued)

	Error Message	Probable Cause(s)	Recommended Action
8007	munmap(*address*) failed: *error_message*	Failed to unmap a region at the virtual address of the file *file_name* used in the "page" mode (fatal error).	
8008	close() failed on *file_name*: *error_message*	Fatal error.	
8009	Not enough space to store miscompare information	Failed to allocate space to store miscompare data (fatal error).	Increase swap space or reduce the load on the system.
8010	swapctl() failed: *error_message*	Failed to obtain the amount of swap space configured in the system (fatal error)	
8011	calloc failed: *error*	Fatal error.	
8012	getitimes() error: *error*	Fatal error.	

Loopback Connectors

Loopback connectors are designed for the testing of communication ports. They take the form of either a single plug or a port-to-port cable with some communication connections shorted (looped-back).

Note – Loopback connectors must be wired properly and connected firmly for the Serial Port tests to work correctly. Miswired, poorly soldered, or missing loopback connectors can cause erroneous diagnostic error messages.

The table below depicts the pin assignments for most loopback plugs and cables that may be used when testing a system.

TABLE A-1 Loopback Connector Pin Assignments

Signal Description	EIA	CCITT #	RS-449 "A"	RS-449 "B"	DIN 8 8-pin round	DB9 9-pin	DB25 25-pin	Direction	Alpha ID
Chassis/ Frame Ground	AA	101	1	NC*	NC*	NC*	1	none	AA
Transmit Data (TxDa)	BA	103	4	22	3	3	2	output	BA
Receive Data (RxDa)	BB	104	6	24	5	2	3	input	BB
Request To Send (RTSa)	CA	105	7	25	6	7	4	output	CA
Clear To Send (CTSa)	CB	106	9	27	2	8	5	input	CB

TABLE A-1   Loopback Connector Pin Assignments

Signal Description	EIA	CCITT #	RS-449 "A"	RS-449 "B"	DIN 8 8-pin round	DB9 9-pin	DB25 25-pin	Direction	Alpha ID
Data Set Ready (DSRa)	CC	107	11	29	NC*	6	6	input/ output	CC
Signal Ground (SG)	AB	102	19	NC*	4	5	7	none	AB
Data Carrier Detect (DCDa)	CF	109	13	31	7	1	8	input	CF
Transmit Clock In (TRxCa)	DB	114	5	23	NC*	NC*	15	input	DB
Receive Clock in (RTxCa)	DD	115	8	26	8	NC*	17	input	DD
Data Terminal Ready (DTRa)	CD	108	12	30	1	4	20	output	CD
External Clock Out (TRxCa)	DA	113	17	35	NC*	NC*	24	output	DA
Secondary Data Carrier Detect (DCDb)	SCF	122	NC*	NC*	NC*	NC*	12	input	SCF
Secondary Clear to Send (CTSb)	SCB	121	NC*	NC*	NC*	NC*	13	input	SCB
Secondary Transmit Data (TxDb)	SBA	118	NC*	NC*	NC*	NC*	14	output	SBA
Secondary Receive Data (RxDb)	SBB	119	NC*	NC*	NC*	NC*	16	input	SBB
Secondary Request to Send (RTSb)	SCA	120	NC*	NC*	NC*	NC*	19	output	SCA

NC = No connection

25-Pin RS-232 Loopback Plug

The RS-232 and RS-423 single-port loopback plug is a specially wired male DB-25 connector. It is plugged in to a serial port in the back of the system under test. The wiring is shown in FIGURE A-1.

Connect:
pin 3 to pin 2
pins 6 and 8 to pin 20
pins 15 and 17 to pin 24

Male
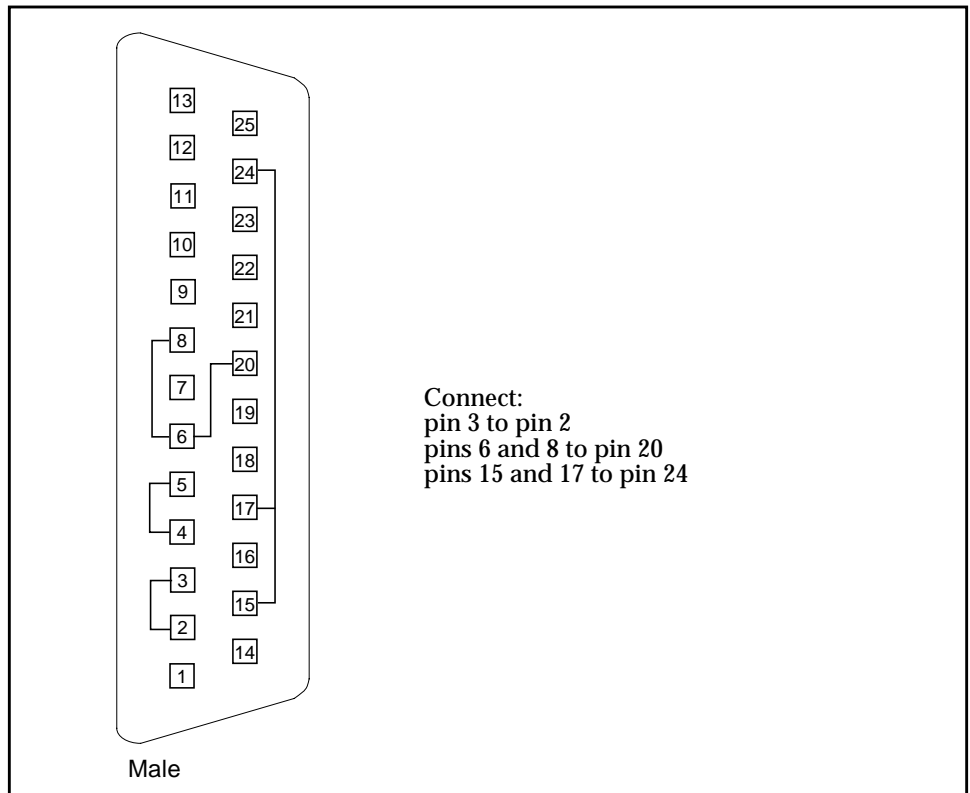
FIGURE A-1   25-pin RS-232 Loopback Plug

25-pin RS-232 Port-to-Port Loopback Cable

Use these wiring instructions for 25-pin RS-232 and RS-423 port to 25-pin RS 232 and RS 423 port loopback cables (two DB-25 connections). It is plugged into a pair of serial ports in the back of the system under test. Both connectors are male. The wiring is shown in FIGURE A-2.

Connect:

First Connector	to	Second Connector
pin 2		pin 3
pin 3		pin 2
pin 4		pin 5
pin 5		pin 4
pins 6 and 8		pin 20
pin 7		pin 7
pins 15 and 17		pin 24
pin 20		pins 6 and 8
pin 24		pins 15 and 17
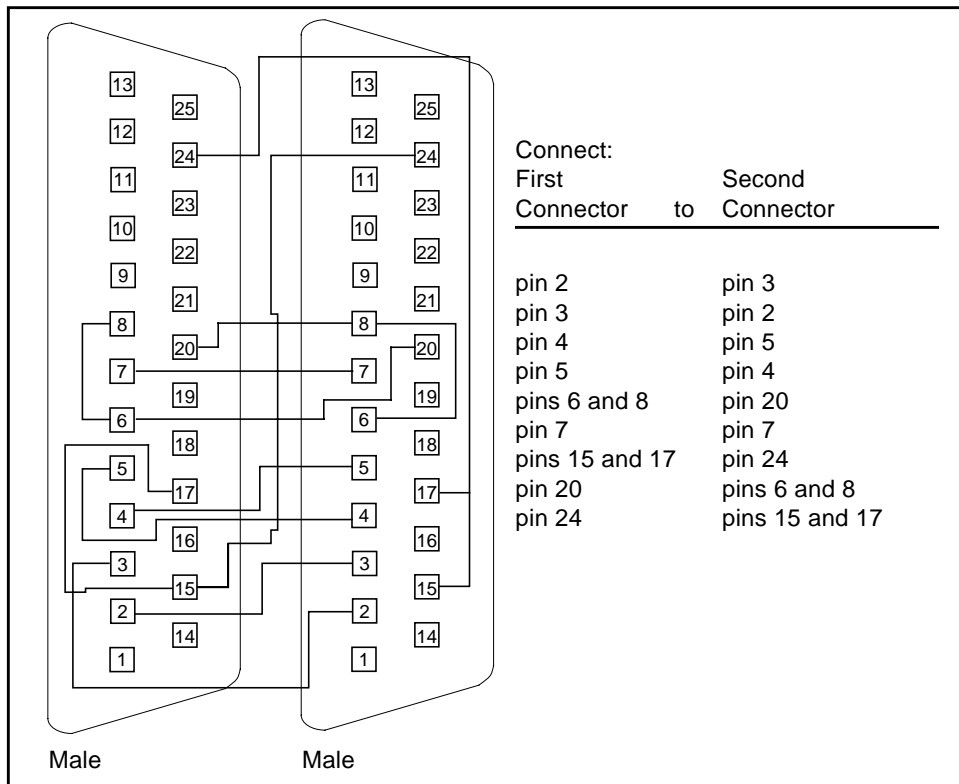
Male   Male

FIGURE A-2   25-pin RS-232 Port-to-Port Loopback Cable

8-Pin to 8-Pin Loopback Cable

Use these wiring directions for 8-pin round DIN RS-232 port to RS-423 to 8-pin round-DIN RS-232 and RS-423 port loopback cable (see FIGURE A-3). Both connectors are male.

	Connect:		
	First Connector	to	Second Connector
	pin 3		pin 5
	pin 5		pin 3
	pin 6		pin 2
	pin 2		pin 6
	pin 7		pin 1
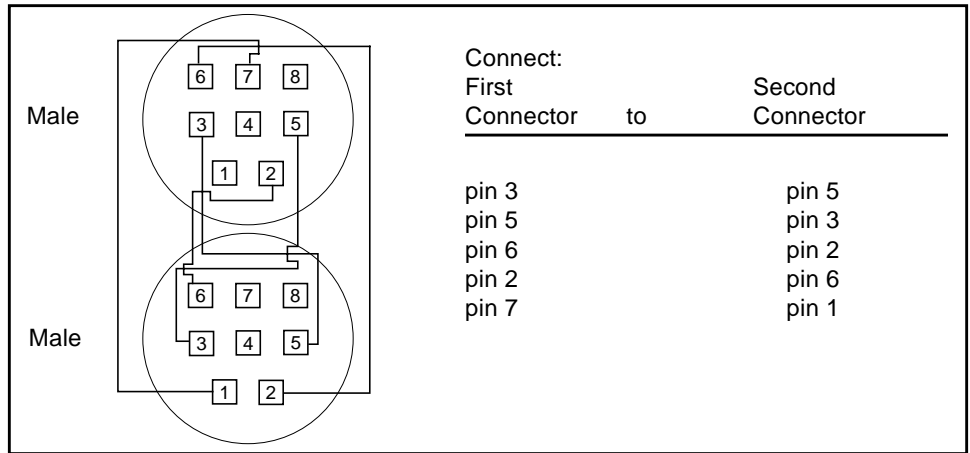
FIGURE A-3   8-Pin to 8-Pin Loopback Cable

Pin 8, Receive clock In (DD), remains unconnected.

8-Pin Loopback Plug

Use these wiring directions for male 8-pin round-DIN RS-232 and RS-423 Single port loopback plugs (see FIGURE A-4).
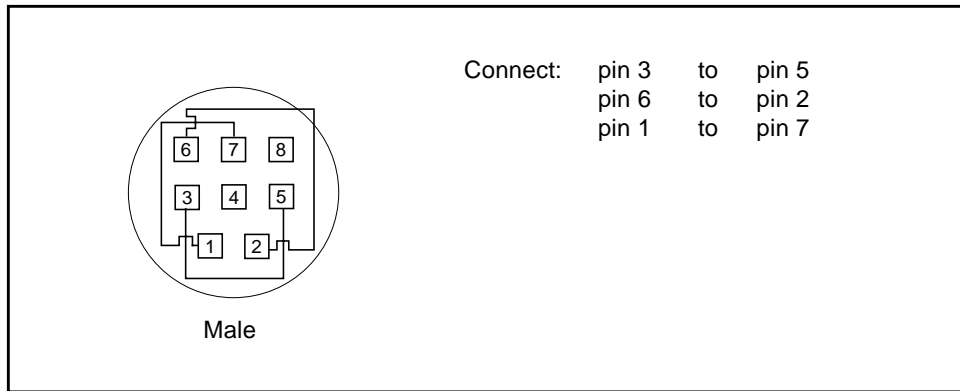
FIGURE A-4   8-Pin Loopback Plug

Pin 8, Receive Clock In (DD), remains unconnected.

25-pin Port A-to-Port B Loopback Plug

Use these wiring directions for a 25-pin Port A to Port B loopback plug for most systems (see FIGURE A-5).

```
Connect:    pin 16         to   pin 2
            pin 3          to   pin 14
            pin 13         to   pin 4
            pin 5          to   pin 19
            pins 6 and 8   to   pin 11
            pin 12         to   pin 20
            pin 18         to   pin 24
            pins 15 and 17 to   pin 25
```
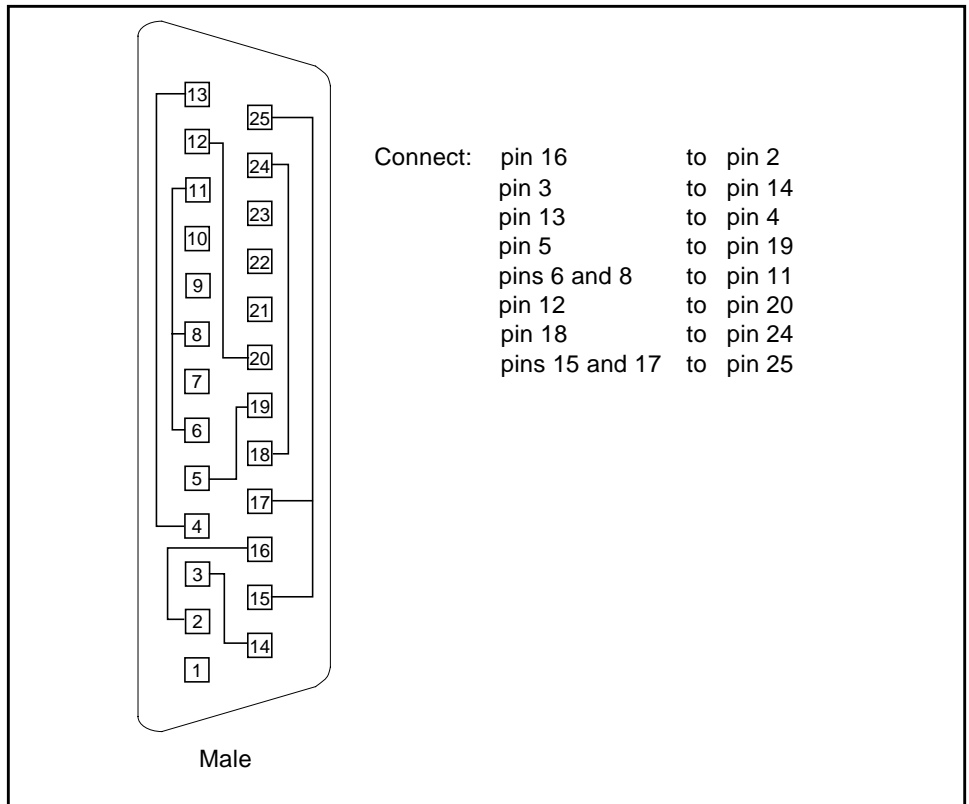
Male

FIGURE A-5   Port A-to-Port B Loopback Plug

25-pin Port A-to-A Port B-to-B Loopback Plug

If your system has a single communication port to connect it to peripherals, use these wiring instructions for making a male 25-pin loopback plug for that communication port (see FIGURE A-6):
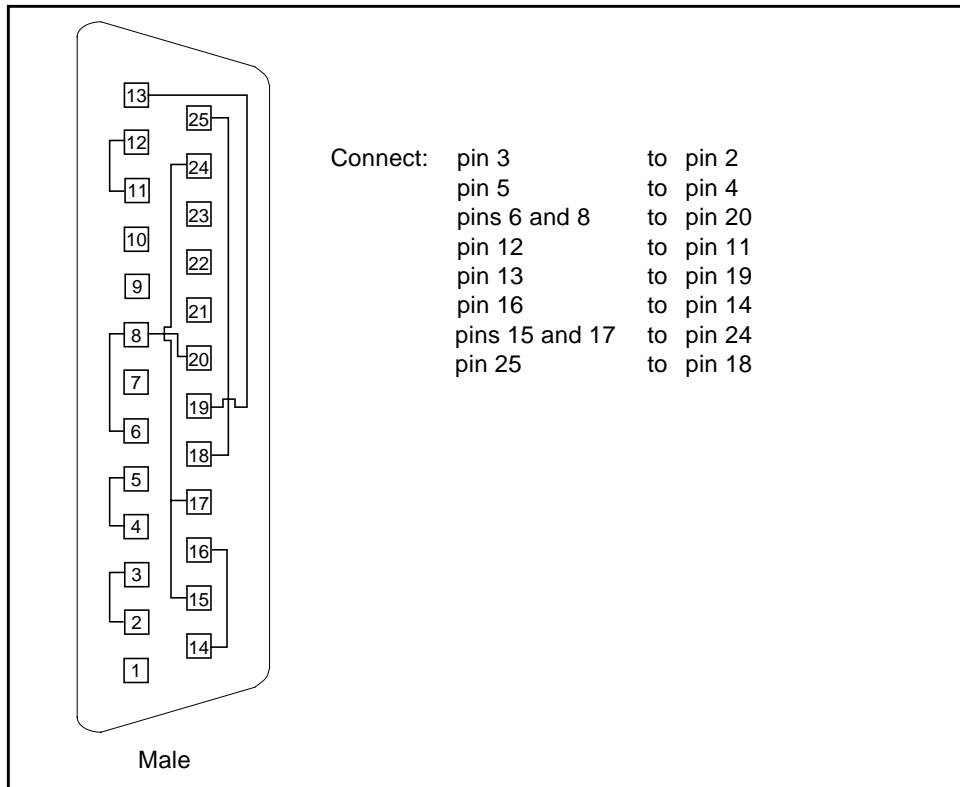
FIGURE A-6   Port A-to-A, Port B-to-B Loopback Plug

96-Pin Female Loopback Connector

This 96-pin connector (see FIGURE A-7) can be ordered from Sun Microsystems (Part Number 370-1366).

Female

```
1 ┌─┐  ┌─┐        ┌─┐  ┌─┐  49
4 │ └──┘ │        │ └──┘ │
5 │      │        │      │
  │      │        │      │
12│      │        │      │  77
  │      │        │      │
20│      │        │      │
  │      │        │      │
  │ ┌──┐ │        │ ┌──┐ │
48└─┘  └─┘        └─┘  └─┘  96
```

Connect: pins 4 and 12 to   pin 77
            pin 5   to   pin 20
            pin 6   to   pin 36
            pin 7   to   pin 44
            pin 10   to   pin 11
            pin 13   to   pin 16
            pin 14   to   pin 15
            pin 18   to   pin 19
            pin 21   to   pin 24
            pin 28   to   pin 60
            pin 29   to   pin 68
            pin 30   to   pin 34
            pin 37   to   pin 40
            pin 38   to   pin 39
            pin 42   to   pin 43
            pin 45   to   pin 48
            pin 46   to   pin 47
            pin 52   to   pin 78
            pin 53   to   pin 55
            pin 54   to   pin 75
            pin 58   to   pin 59
            pin 61   to   pin 64
            pin 62   to   pin 63
            pin 66   to   pin 67
            pin 69   to   pin 72
            pin 76   to   pin 79
            pin 82   to   pin 83
            pin 85   to   pin 88
            pin 86   to   pin 87
            pin 90   to   pin 91
            pin 93   to   pin 96
            pin 94   to   pin 95

Materials:
    PCR-E96FA   (1)
    PCS-E96LKPA   (1)
    3751 Metal Plug   (1)
      (9563K42)
    AWG28 Madison Cable (8" long)
      UL/CSA Approved

FIGURE A-7   96-Pin Female Loopback Connector

96-Pin Female Special Loopback Connector

This 96-pin connector (see FIGURE A-8) can be ordered from Sun Microsystems (Part Number 370-1381).

Female Special

Connect:

pin 3	to	pin 4
pin 5	to	pin 7
pins 8 and 9	to	pin 12
pin 10	to	pin 11
pin 13	to	pin 14
pin 15	to	pin 17
pins 18 and 19	to	pin 22
pin 20	to	pin 21
pin 27	to	pin 28
pin 29	to	pin 31
pins 32 and 33	to	pin 36
pin 34	to	pin 35
pin 37	to	pin 38
pin 39	to	pin 41
pins 42 and 43	to	pin 46
pin 44	to	pin 45
pin 51	to	pin 52
pin 53	to	pin 55
pins 56 and 57	to	pin 60
pin 58	to	pin 59
pin 61	to	pin 62
pin 63	to	pin 65
pins 66 and 67	to	pin 70
pin 68	to	pin 69
pin 75	to	pin 76
pin 77	to	pin 79
pins 80 and 81	to	pin 84
pin 82	to	pin 83
pin 85	to	pin 86
pin 87	to	pin 89
pins 90 and 91	to	pin 94
pin 92	to	pin 93

Materials:
    PCR-E96FA   (1)
    PCS-E96LKPA   (1)
    9563K999 Cap   (1)
    Madison Cable   (6" long)
      (#28 SCSI UL/CSA Approved)

Open Contacts:
    1, 2, 6,16, 23, 24, 25, 26, 30, 40, 47, 48, 49, 50, 54, 64, 71, 72, 73, 74, 78, 88, 95, 96

FIGURE A-8   96-Pin Female Special Loopback Connector

37-Pin RS-449 Loopback Cable

Use these wiring instructions for a loopback cable for two 37-pin RS-449
synchronous ports (see FIGURE A-9).
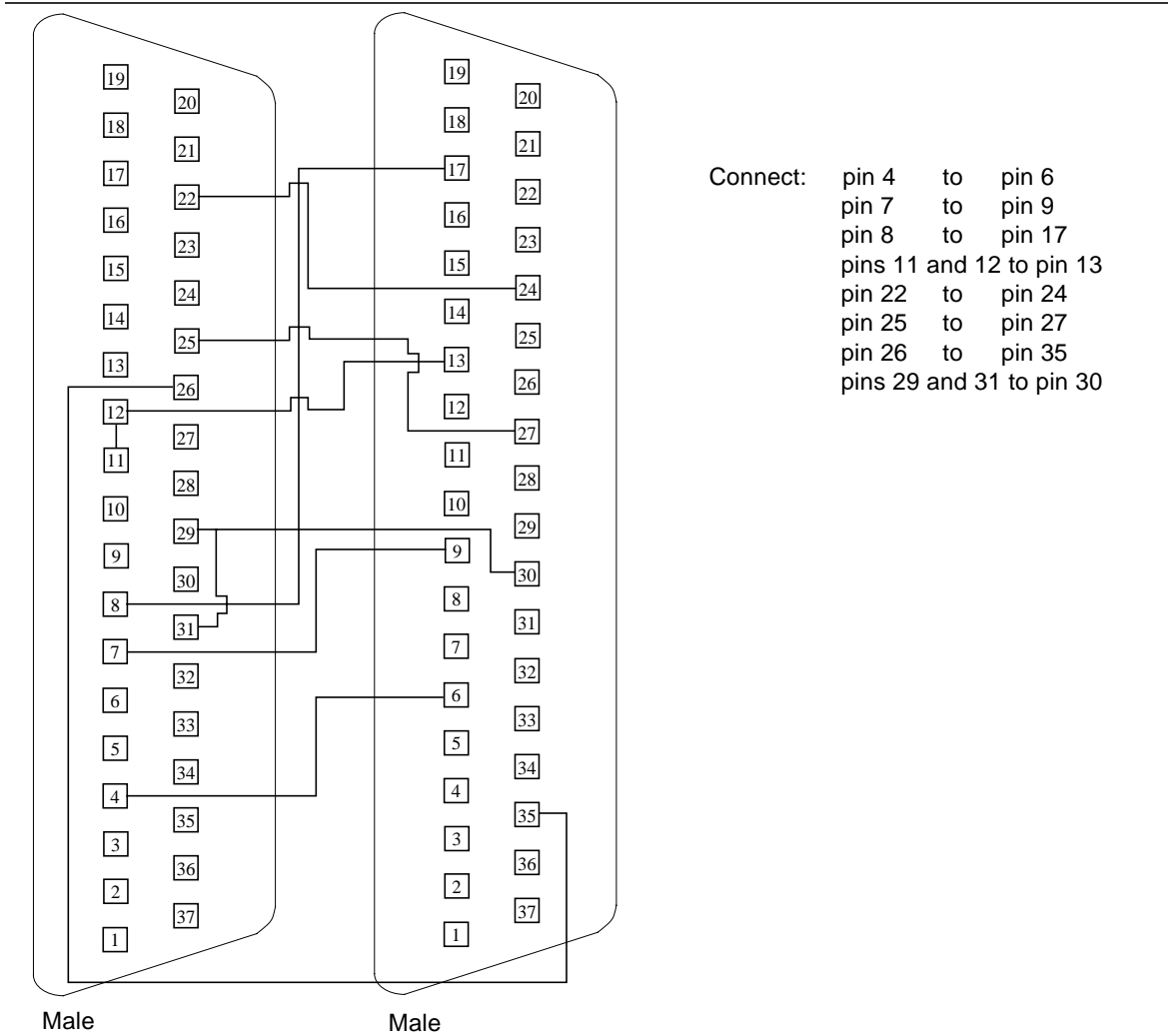
FIGURE A-9   37-Pin RS-449 Loopback Cable

37-Pin RS-449 Loopback Plug

Use these wiring instructions for making a male 37-pin RS-449 loopback plug (see FIGURE A-10). This connector is also available from Sun (Part Number 530-1430).
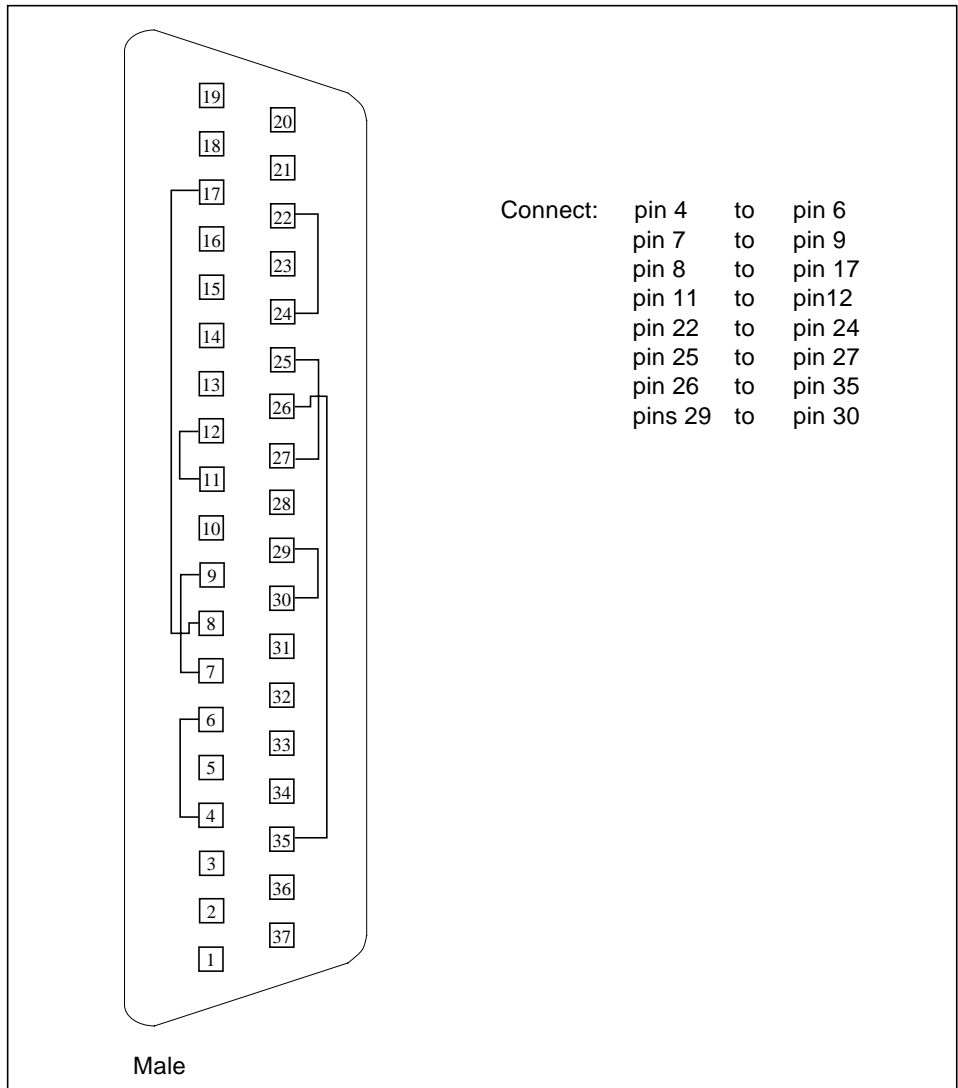
FIGURE A-10 37-Pin RS-449 Loopback Plug

9-pin Male Single-port Loopback Plug

Use these wiring instructions for male 9-pin RS-232 and RS-423 single-port loopback plugs (see FIGURE A-11):
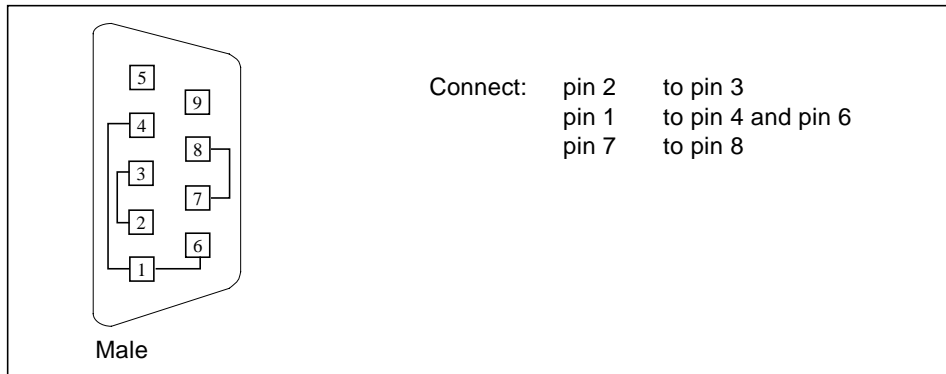
FIGURE A-11 9-Pin Male Single-port Loopback Plug

9-pin Female Single-port Loopback Plug

Use these wiring directions for female 9-pin RS-232 and RS-423 single-port loopback plugs (see FIGURE A-12):
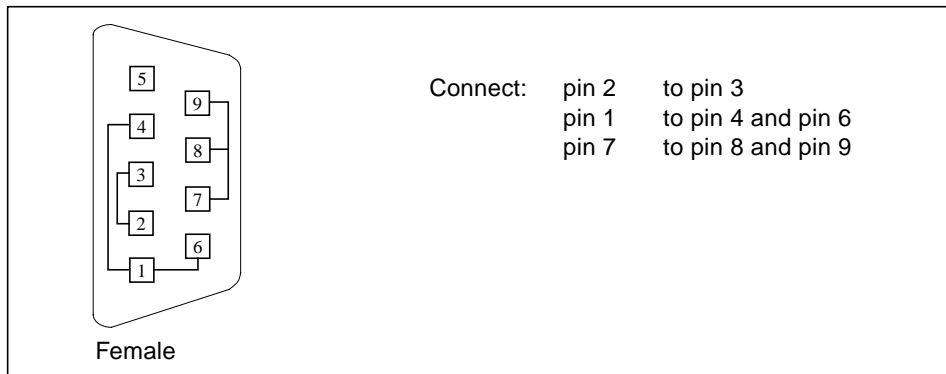
FIGURE A-12 9-Pin Female Single-port Loopback Plug

Note – Use this loopback plug with the `pcmciatest`.

9-Pin to 25-Pin Port-to-Port Loopback Cable

Use these wiring instructions for a 9-pin RS-232 and RS-423 port to 25-pin RS-232 and RS 423 port loopback cables (see FIGURE A-13). Both connectors are male.

Connect:

First Connector	to	Second Connector
pin 2		pin 2
pin 3		pin 3
pin 8		pin 4
pin 7		pin 5
pin 5		pin 7
pin 4		pins 6 and 8
pins 1 and 6		pin 20
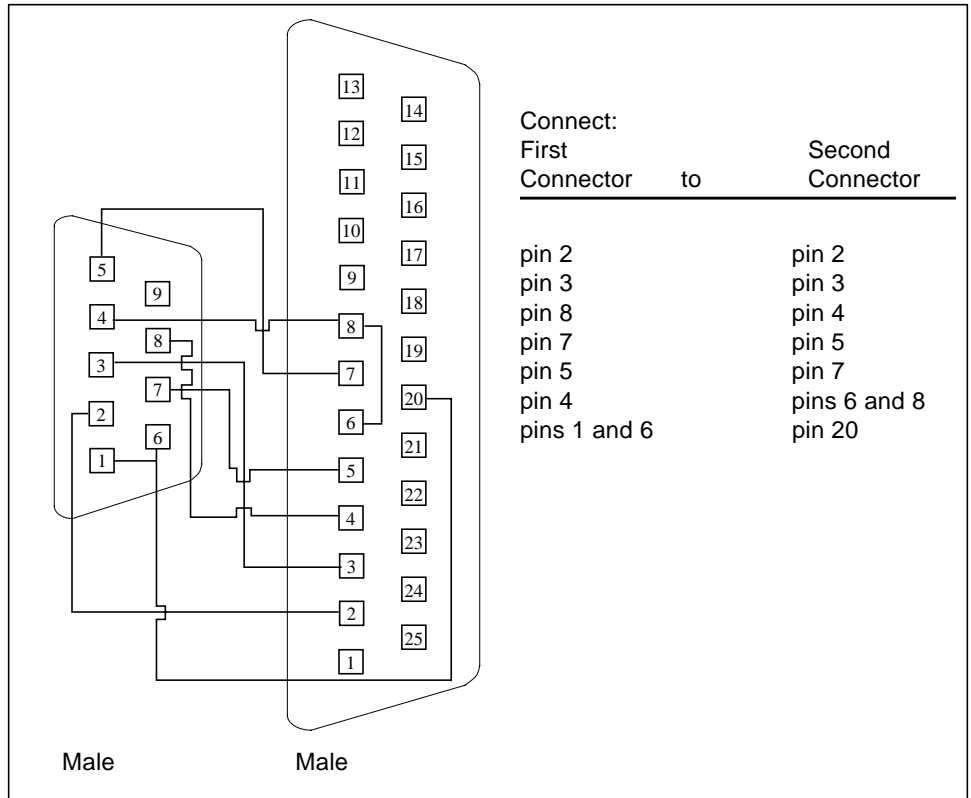
FIGURE A-13 9-Pin to 25-Pin Port-to-Port Loopback Cable

9-Pin to 9-Pin Port-to-Port Loopback Cable

Use these wiring instructions for 9-pin RS-232 and RS 423 port to 9-pin RS-232 and RS-423 port loopback cables (see FIGURE A-14). Both connectors are male.

Connect:		
First Connector	to	Second Connector
pins 1 and 6		pin 4
pin 2		pin 3
pin 3		pin 2
pin 4		pins 1 and 6
pin 5		pin 5
pin 7		pin 8
pin 8		pin 7
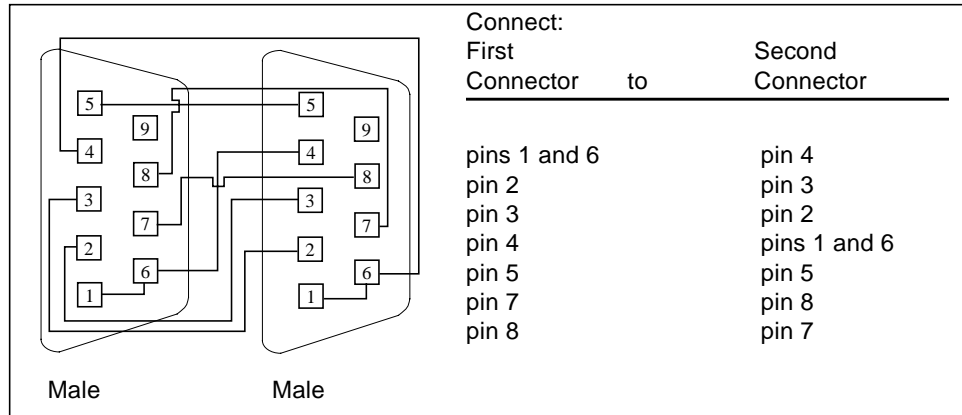
FIGURE A-14 9-Pin to 9-Pin Port-to-Port Loopback Cable

Please note that this cable has no Sun part number assigned to it.

NT to TE Loopback Cable

Using two standard RJ45 connectors, and connect pin1 to pin1, pin 2 to pin 2, and so on, for all pins. This loopback is a "straight-through" connection.

Index