Sun Fire™ Entry-Level Midrange System Administration Guide

Firmware Release 5.19.0

Sun Microsystems, Inc.
www.sun.com

Part No. 819-1269-10
July 2005, Revision A

Submit comments about this document at: http://www.sun.com/hwdocs/feedback

Please
Recycle

Contents

Figures

Tables

Code Samples

Preface

This book provides an overview of the system and presents a step-by-step description of common administration procedures. It explains how to configure and manage system controller firmware on the Sun Fire™ family of entry-level midrange servers–the Sun Fire E2900 and the Sun Fire V1280/Netra 1280 systems. It explains how to remove and replace components and perform firmware upgrades. This book also contains information about security, troubleshooting, and a glossary of technical terms.

How This Book Is Organized

Chapter 1 describes the system controller, and explains board states, describes redundant system components, minimum system configurations, and reliability, serviceability, and availability.

Chapter 2 describes how to power on and set up the system for the first time.

Chapter 3 describes how to navigate within the system controller.

Chapter 4 explains system controller message logging.

Chapter 5 describes how to use the LOM from the Solaris™ console.

Chapter 6 describes how to run the power-on self-test (POST).

Chapter 7 describes the automatic diagnosis and domain restoration features of the firmware.

Chapter 8 describes security guidelines.

Chapter 9 describes the Capacity on Demand (COD) option and how to allocate, activate, and monitor COD resources.

Chapter 10 describes troubleshooting information including LEDs, system faults, displaying diagnostic information, displaying system configuration information, disabling components (blacklisting), and mapping device path names to physical system devices.

Chapter 11 provides information on firmware updates, including how to update the flash PROMs, and the procedure for updating the system controller firmware.

Chapter 12 describes Dynamic Reconfiguration and the procedures you can use.

Appendix A describes the Watchdog timer, Alarm 3, and procedures for their use.

Using UNIX Commands

This book assumes you are experienced with the UNIX® Operating System. If you are not experienced with the UNIX operating system, see one or more of the following for this information:

- AnswerBook2™ online documentation for the Solaris Operating System.
- Other software documentation that you received with your system.

Typographic Conventions

Typeface	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	Read Chapter 6 in the *User's Guide*. These are called *class* options. You *must* be superuser to do this.
	Command-line variable; replace with a real name or value	To delete a file, type `rm` *filename*.

Shell Prompts

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

Related Documentation

Type of Book	Title	Part Number
Command reference	*Sun Fire Entry-Level Midrange System Controller Command Reference Manual*	819-1268-10

Accessing Sun Documentation

You can view and print a broad selection of Sun™ documentation, including localized versions, at:

http://www.sun.com/documentation

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. You can email your comments to Sun at:

docfeedback@sun.com

Please include the part number of your document (819-1269-10) in the subject line of your email.

CHAPTER 1

Overview

This chapter provides a basic understanding of the features of Sun Fire entry-level midrange servers—Sun Fire E2900 and Sun Fire V1280/Netra 1280 systems.

The procedural overview of how to set up your system with detailed procedures, is covered in Chapter 2.

This chapter describes the following topics:

System Controller

The system controller (SC) is an embedded system resident on the IB_SSC assembly, which connects to the system baseplane. The SC is responsible for providing the Lights Out Management (LOM) functions which include power on sequencing, sequencing module powe- on self-tests (POST), environmental monitoring, fault indication, and alarms.

The SC provides an RS 232 serial interface and one 10/100 Ethernet interface. Access to the LOM command line interface and the Solaris and OpenBoot™ PROM consoles are shared and obtained through these interfaces.

System controller functions include:

- Monitoring the system
- Providing the Solaris and OpenBoot PROM consoles
- Providing the virtual TOD (time of day)
- Performing environmental monitoring
- Performing system initialization
- Coordinating POST

The software application running on the SC provides a command-line interface for you to modify system settings.

I/O Ports

The following ports are on the rear of the system:

- Console serial (RS-232) port (RJ-45)
- Reserved serial (RS-232) port (RJ-45)
- Two Gigabit Ethernet ports (RJ-45)
- Alarms port (DB-15)
- System controller 10/100 Ethernet port (RJ-45)
- UltraSCSI port
- Up to six PCI ports (five 33 MHz, one 66 MHz)

Their locations are shown in FIGURE 1-1.

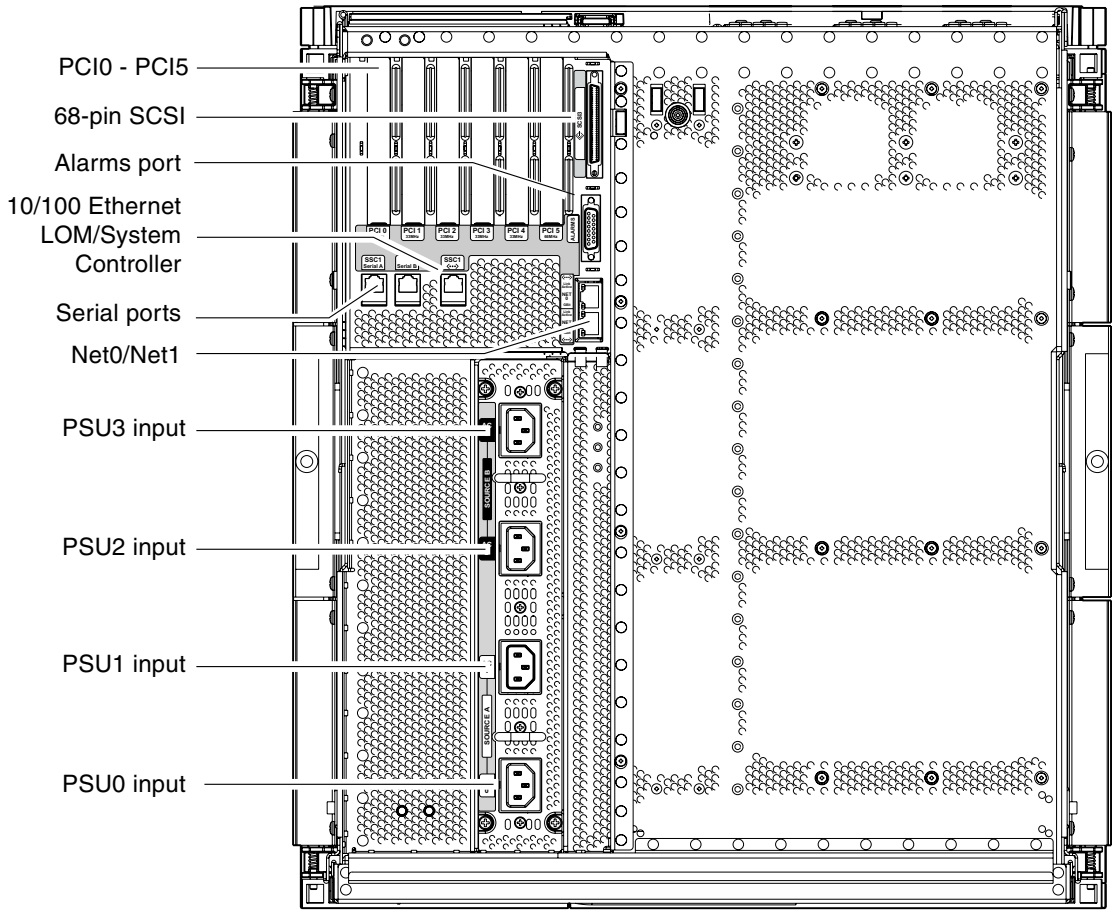

FIGURE 1-1 I/O Ports

The console serial port and 10/100 Ethernet port can be used to access the system controller.

Use the console serial port to connect directly to an ASCII terminal or an NTS (network terminal server). Connecting the System Controller board with a serial cable enables you to access the system controller command line interface with an ASCII terminal or an NTS.

Use the 10/100 Ethernet port to connect the SC to the network.

LOM Prompt

The LOM prompt provides the command-line interface for the SC. It is also the place where console messages are displayed:

```
lom>
```

Some of the system management tasks are shown in TABLE 1-1.

TABLE 1-1 Selected System Controller Management Tasks

Tasks	Commands
Configuring thesystem controller	`password`, `setescape`, `seteventreporting`, `setupnetwork`, `setupsc`
Configuring the system	`setalarm`, `setlocator`
Powering boards on and off and powering the system on or off	`poweron`, `poweroff`, `reset`, `shutdown`
Testing the CPU/Memory board	`testboard`
Resetting the system controller	`resetsc`
Marking components as faulty or OK	`disablecomponent`, `enablecomponent`
Upgrading firmware	`flashupdate`
Displaying the current system controller settings	`showescape`, `showeventreporting`, `shownetwork`, `showsc`
Displaying the current system state	`showalarm`, `showboards`, `showcomponent`, `showenvironment`, `showfault`, `showhostname`, `showlocator`, `showlogs`, `showmodel`, `showresetstate`
Setting the date, time, and time zone	`setdate`
Displaying the date and time	`showdate`

Solaris Console

If the Solaris Operating System, the OpenBoot PROM, or POST is running, you can access the Solaris console. When you connect to the Solaris console, you will be in one of the following modes of operation:

- Solaris Operating System console (`%` or `#` prompts).
- OpenBoot PROM (`ok` prompt).
- System will be running POST and you can view the POST output.

To switch between these prompts and the LOM prompt, see "Switching Between the Different Consoles" on page 37.

Environmental Monitoring

There are sensors that monitor temperature, voltage, and cooling.

The SC polls these sensors in a timely manner and makes the environmental data available. If necessary, the SC shuts down various components to prevent damage.

For instance, in the case of an overtemperature, the SC notifies the Solaris Operating System of the overtemperature and the operating system takes action. In the case of extreme overtemperature, the SC software can shut down the system without first notifying the operating system.

System Indicator Board

The system indicator board contains the On/Standby switch and indicator LEDs as shown in FIGURE 1-2.

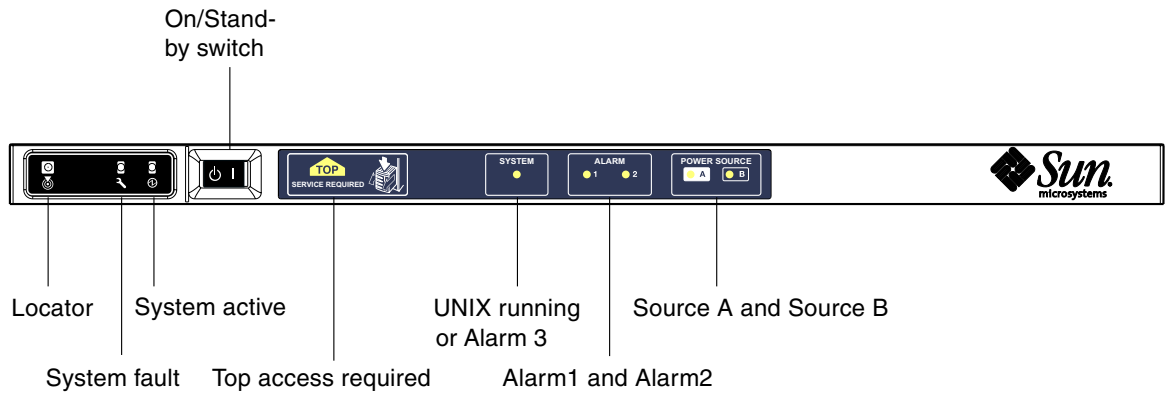

FIGURE 1-2 System Indicator Board

The indicator LEDs function as shown in TABLE 1-2.

TABLE 1-2 System Indicator LED Functions

Name	Color	Function
Locator*	White	Normally off; can be lit by user command
System Fault*	Amber	Lights when the LOM detects a fault
System Active*	Green	Lights when power is applied to the system
Top Access	Amber	Lights when a fault occurs in a FRU, which can only be replaced from the top of the system
UNIX Running	Green	Lights when Solaris is running. Off while the system is powering up Can be reset by watchdog timeout, or by assertion of user-defined Alarm 3 (for further information, see "Programming Alarm 3" on page 159
Alarm1 and Alarm2	Green	Light when triggered by events as specified in the LOM
Source A and Source B	Green	Light when the relevant power feeds are present

* This indicator is duplicated on the rear of the system.

Reliability, Availability, and Serviceability (RAS)

Reliability, availability, and serviceability (RAS) are features of this system.

- *Reliability* is the probability that a system stays operational for a specified time period when operating under normal environmental conditions. Reliability differs from availability in that reliability involves only system failure, whereas availability depends on both failure and recovery.
- *Availability*, also known as average availability, is the percentage of time that a system is available to perform its functions correctly. Availability can be measured at the system level or in the context of the availability of a service to an end client. The "system availability" is likely to impose an upper limit on the availability of any products built on top of that system.
- *Serviceability* measures the ease and effectiveness of maintenance and system repair for the product. There is no single well-defined metric, because serviceability can include both Mean Time to Repair (MTTR) and diagnosability.

The following sections provide details on RAS. For more hardware-related information on RAS, see the *Sun Fire E2900 System Service Manual* or *Sun Fire V1280/Netra 1280 Service Manual*. For RAS features that involve the Solaris Operating System, see the *Sun Hardware Platform Guide.*

Reliability

The software reliability features include:

- Disabling Components or Boards and Power-On Self-Test (POST)
- Manual Disabling of Components
- Environmental Monitoring

The reliability features also improve system availability.

Disabling Components or Boards and Power-On Self-Test (POST)

The power-on self-test (POST) is part of powering on the system. If the board or component failed testing, POST disables components or boards. The `showboards` command displays the board as either being failed or degraded. The system, running the Solaris Operating System, is booted only with components that have passed POST testing.

Manual Disabling of Components

The system controller provides component-level status and user-controlled modification of component status.

Set the component location status by running the setls command from the console. The component location status is updated at the next domain reboot, board power cycle, or POST execution (for example, POST is run whenever you perform a setkeyswitch on or off operation).

Note – The `enablecomponent` and `disablecomponent` commands have been replaced by the `setls` command. These commands were formerly used to manage component resources. While the `enablecomponent` and `disablecomponent` commands are still available, it is suggested that you use the `setls` command to control the configuration of components into or out of the system.

The `showcomponent` command displays status information about the component, including whether or not it has been disabled.

Environmental Monitoring

The SC monitors the system's temperature, cooling, and voltage sensors. The SC provides the latest environmental status information to the Solaris Operating System. If hardware needs to be powered off, the SC notifies the Solaris Operating System to perform a system shutdown.

Availability

The software availability features include:

- Dynamic Reconfiguration
- Power Failure
- System Controller Reboot
- Host Watchdog

Dynamic Reconfiguration

The following components can be dynamically reconfigured:

- Hard disk drives
- CPU/Memory boards
- Power supplies
- Fans

Power Failure

On recovery from a power outage, the SC attempts to restore the system to its previous state.

System Controller Reboot

The SC can be rebooted and will start up and resume management of the system. The reboot does not disturb the currently running Solaris Operating System.

Host Watchdog

The SC monitors the state of the Solaris Operating System and will initiate a reset if the system stops responding.

Serviceability

The software serviceability features promote the efficiency and timeliness of providing routine as well as emergency service to the system.

- LEDs
- Nomenclature
- System controller error logging
- System controller XIR (eXternally Initiated Reset) support

LEDs

All field-replaceable units (FRUs) that are accessible from outside the system have LEDs that indicate their state. The SC manages all the LEDs in the system, with the exception of the power supply LEDs, which are managed by the power supplies. For a discussion of LED functions, see the appropriate board or device chapter of the *Sun Fire V1280/Netra 1280 Service Manual*.

Nomenclature

The SC, the Solaris Operating System, the power-on self-test (POST), and the OpenBoot PROM error messages use FRU name identifiers that match the physical labels in the system. The only exception is the OpenBoot PROM nomenclature used for I/O devices, which use the device path names as described in Chapter 10 to indicate I/O devices during device probing.

System Controller Error Logging

SC error messages are automatically notified to the Solaris Operating System. The SC also has an internal buffer where error messages are stored. You can display the SC logged events, stored in the SC message buffer, by using the `showlogs` command.

System Controller XIR (eXternally Initiated Reset) Support

The SC `reset` command enables you to recover from a hung system and extract a Solaris Operating System `core` file.

Capacity on Demand Option

The Capacity on Demand (COD) option provides additional processing resources (additional CPUs) on systems containing UltraSPARC IV CPU/Memory boards, (such as the Sun Fire E2900 servers) when you need them. These additional CPUs are provided on COD CPU/Memory boards that are installed in your system. However, to access these COD CPUs, you must first purchase the COD right-to-use (RTU) licenses for them. After you obtain the COD RTU licenses for your COD CPUs, you can activate those CPUs as needed. For details on COD, see "Capacity on Demand" on page 85.

CHAPTER 2

Starting and Setting Up Sun Fire Entry-Level Midrange Systems

This chapter describes how to power on your system using the system controller command line interface (LOM prompt), how to set up the SC using the `setupnetwork` command, and how to boot the Solaris Operating System.

This chapter contains the following topics:

The major steps you must perform to power on and set up the system, are as follows:

1. Installing and cabling the hardware.
2. Applying external power to the hardware.
3. Setting the date and time for the system.
4. Setting the password for the SC.
5. Setting up system-specific parameters with the `setupnetwork` command.
6. Powering on all hardware with the `poweron` command.
7. If the Solaris Operating System is not pre-installed, installing it.
8. Booting the Solaris Operating System.
9. Installing the Lights Out Management packages from the Solaris Supplementary CD.

Setting Up the Hardware

▼ To Install and Cable the Hardware

1. **Connect a terminal to the System Controller board serial port (**FIGURE 1-1**).**
2. **Set up the terminal to use the same baud rate as the SC serial port (9600 8N1).**

   The serial port settings of the System Controller board are:

   - 9600 baud
   - 8 data bits
   - No parity
   - 1 stop bit

   You can find more details in the *Sun Fire E2900 System Installation Guide* or the *Sun Fire V1280/Netra 1280 Systems Installation Guide*.

Using the Power On/Standby Switch

The power (On/Standby) switch of the Sun Fire entry-level midrange systems system is a rocker type, momentary action switch. It controls only low voltage signals and no high voltage circuits pass through it.

Note – The power switch is not an On/Off switch, it is an On/Standby switch. It does not isolate the equipment.

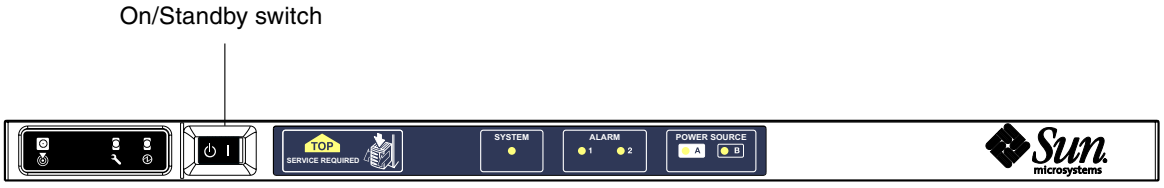

FIGURE 2-1 Power On/Standby Switch

The following table describes the switch positions

TABLE 2-1 Power On/Standby Switch Description

Symbol	Description
| On	Press and release to power on the server. This is the equivalent of the LOM `poweron` command.
⏻ Standby	• Press for less than four seconds to initiate an orderly shutdown of the system into Standby mode. This is equivalent to issuing the `shutdown` command at the `lom>` prompt. This is the method to use under normal operation. • Press and hold down for more than four seconds to perform a system power down to standby mode. This is equivalent to issuing the `poweroff` command at the `lom>` prompt. This process is not interruptible. You should ensure that Solaris is cleanly shut down before powering a system to standby mode otherwise data is at risk of being lost. The recommended method of powering down to standby is by using the `shutdown` command at the LOM prompt.

Use the LOM `setupsc` command to prevent accidental operation of the On/Standby switch.

Powering On and Off

When all the power cables are connected and external circuit breakers are switched on, the system enters Standby mode. The Source A and Source B indicators are the only indicator LEDs to be illuminated on the system indicator board. The IB_SSC assembly Active LED is lit, but not visible from the front of the system.

Powering the system on from Standby mode can be achieved in either of two ways:

- Operating the On/Standby switch
- Sending the `poweron` command via the LOM port.

If the `auto-boot?` variable has been set in the OBP, the system will automatically boot into the Solaris Operating System.

▼ To Power On Using the On/Standby Switch

1. **Check that power is applied to the system and that it is correctly in Standby mode.**

   The only indicator LEDs to be illuminated on the system indicator board are the Source A and Source B indicators. The IB_SSC assembly Active LED will also be illuminated, but not visible from the front of the system.

2. **Momentarily press the On/Standby switch to the right.**

   The system will power on completely. The System Active indicator will be illuminated in addition to the Source A and Source B indicators. The system will execute the power on self tests (POST).

▼ To Power On Using the LOM `poweron` Command

- **At the `lom>` prompt, type:**

```
lom>poweron
```

The SC first powers on all the power supplies, followed by the fan tray. Finally the SC powers on the system boards. If the value of the OpenBoot PROM variable `auto-boot?` is `true` then the system also boots the Solaris Operating System.

Individual modules can also be powered on using the `poweron` command. For further details see the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual*.

The System Active indicator is lit. The system will execute the power-on self-tests (POST).

Note – The `poweron all` command only powers on individual components; it does not boot the Solaris software.

See the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual* for a full description of the `poweron` command.

Bringing the System to Standby Mode

This can be achieved five ways:

- Using the UNIX `shutdown` command.
- Sending the `shutdown` command through the LOM port
- Sending the `shutdown` command using the On/Standby switch
- Sending the `poweroff` command through the LOM port
- Sending the `poweroff` command using the On/Standby switch

Note – Ensure that the system is cleanly shut down before powering to Standby mode, otherwise data is at risk of being lost.

▼ To Use the Solaris `shutdown` Command

- **At the system prompt, type:**

```
# shutdown -i5
```

The system powers off to Standby mode. The Source A and Source B indicators are the only indicator LEDs to be illuminated on the system indicator board. The IB_SSC assembly Active LED is lit, but not visible from the front of the system.

▼ To Use the LOM `shutdown` Command

Use the LOM `shutdown` command to perform a clean shutdown of the system followed by a power off of all modules and the system chassis to Standby mode.

Note – If the Solaris software is running, this command attempts to halt the system cleanly before powering down the system to Standby mode, and is the equivalent of the Solaris `init 5` command.

- **At the `lom>` prompt, type:**

```
lom>shutdown
```

After Solaris has been stopped, the system powers off to standby mode. The Source A and Source B indicators are the only indicator LEDs to be illuminated on the system indicator board. The IB_SSC assembly Active LED is lit, but not visible from the front of the system.

See the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual* for a full description of the LOM `shutdown` command.

▼ To Use the `shutdown` Command With the On/Standby Switch

- **Press the left side of the system On/Standby switch.**

  This initiates an orderly shutdown of the system into Standby mode. This is equivalent to issuing the `shutdown` command at the `lom>` prompt.

▼ To Use the LOM `poweroff` Command

Use the `poweroff` command when you wish to power off all modules and system chassis to Standby mode.

1. **At the `lom>` prompt, type:**

```
lom>poweroff

This will abruptly terminate Solaris.
Do you want to continue? [no]
```

Answer `yes` only if you wish to forcibly power off the system regardless of the state of the Solaris system. Under normal operation, you should use the `shutdown` command.

2. **Type `y` to continue or press Return to cancel the command.**

   The system powers off to Standby mode. The Source A and Source B indicators are the only indicator LEDs to be illuminated on the system indicator board. The IB_SSC assembly Active LED is lit, but not visible from the front of the system.

   See the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual* for a full description of the `poweroff` command.

▼ To Use the `poweroff` Command With the On/Standby Switch

Use this method only if you wish to forcibly power off the system regardless of the state of the Solaris system. Under normal operation you should send the `shutdown` command either from the `lom>` prompt or from the On/Standby switch (see "To Use the `shutdown` Command With the On/Standby Switch" on page 16).

- **Press the left side of the On/Standby switch and hold it for at least four seconds.**

  The system powers off to Standby mode. The Source A and Source B indicators are the only indicator LEDs to be illuminated on the system indicator board. The IB_SSC assembly Active LED is lit, but not visible from the front of the system.

After Powering On

You will see the following output on the system controller serial port connection:

CODE EXAMPLE 2-1 Hardware Reset Output From the System Controller

```
Hardware Reset...


@(#) SYSTEM CONTROLLER(SC) POST 23 2002/03/22 18:03
PSR = 0x044010e5
PCR = 0x04004000


Basic sanity checks done.
Skipping POST ...
ERI Device Present
Getting MAC address for SSC1
Using SCC MAC address
MAC address is 0:3:xx:xx:xx:xx
Hostname: some_name
Address: xxx.xxx.xxx.xxx
Netmask: 255.255.255.0
Attached TCP/IP interface to eri unit 0
Attaching interface lo0...done
Gateway: xxx.xxx.xxx.xxx
interrupt: 100 Mbps half duplex link up


          Copyright 2001-2002 Sun Microsystems, Inc.  All rights reserved.
          Use is subject to license terms.

Lights Out Management Firmware
RTOS version: 23
ScApp version: 5.13.0007 LW8_build0.7
SC POST diag level: off

The date is Friday, July 19, 2002, 3:48:50 PM BST.

Fri Jul 19 15:48:51 some_name lom: Boot: ScApp 5.13.0007, RTOS 23
Fri Jul 19 15:48:54 some_name lom: SBBC Reset Reason(s): Power On Reset
Fri Jul 19 15:48:54 some_name lom: Initializing the SC SRAM
Fri Jul 19 15:48:59 some_name lom: Caching ID information
Fri Jul 19 15:49:00 some_name lom: Clock Source: 75MHz
Fri Jul 19 15:49:02 some_name lom: /N0/PS0: Status is OK
Fri Jul 19 15:49:03 some_name lom: /N0/PS1: Status is OK
Fri Jul 19 15:49:03 some_name lom: Chassis is in single partition mode.
Fri Jul 19 15:49:05 some_name lom: Cold boot detected: recovering active domains
```

CODE EXAMPLE 2-1 Hardware Reset Output From the System Controller *(Continued)*

```
Hardware Reset...
Fri Jul 19 15:49:06 some_name lom: NOTICE: /N0/FT0 is powered off

Connected.

lom>
```

Setting Up the System

After powering on, you must set up your system using the SC `setdate` and `setupnetwork` commands described in this chapter.

This section contains the following topics:

▼ To Set the Date and Time

If your time zone area is using daylight or summer time, this is set automatically.

- **Set the date, time, and time zone for the system by using the `setdate` command at the LOM prompt:**

  The following example shows setting the time zone to Pacific Standard Time (PST) using the offset from Greenwich mean time (GMT), date, and time to Tuesday, April 20, 2004 at 18 hours 15 minutes and 10 seconds.

```
lom>setdate -t GMT-8 042018152004.10
```

If Solaris software is running, use the Solaris `date` command instead.

For more information on the `setdate` command, see the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual.*

▼ To Set Up the Password

1. **At the LOM prompt, type the SC `password` command.**

2. **At the** `Enter new password`**: prompt, type your password.**

3. **At the** `Enter new password again:` **prompt, re-type your password.**

```
lom>password
Enter new password:
Enter new password again:
lom>
```

In the event that your password has been lost or forgotten, contact Sun™ Service for advice.

▼ To Configure Network Parameters

The Sun Fire entry-level midrange system can be administered from the SC LOM prompt and from the Solaris software. There are two ways to access the LOM Console connection:

- Through the SC serial port connection.
- Through a remote (network) connection using the 10/100 Ethernet port.

Note – The system can be administered solely through the serial port, but if you want to use the 10/100 Ethernet port, use a separate secure subnet for this connection. Remote connection capability is not enabled by default. If you use SSH or Telnet to administer the system, you must set the connection type to SSH or Telnet, using the `setupnetwork` command.

- **At the LOM prompt, type** `setupnetwork`

```
lom>setupnetwork
```

Note – If you press the Return key after each question, the current value will not be changed.

See the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual* for full details of the `setupnetwork` command. CODE EXAMPLE 2-2 shows an example of the `setupnetwork` command.

CODE EXAMPLE 2-2 Output From the `setupnetwork` Command

```
lom> setupnetwork

Network Configuration
---------------------
Is the system controller on a network? [yes]:
Use DHCP or static network settings? [static]:
Hostname [hostname]:
IP Address [xxx.xxx.xxx.xxx]:
Netmask [xxx.xxx.xxx.x]:
Gateway [xxx.xxx.xxx.xxx]:
DNS Domain [xxxx.xxx.xxx]:
Primary DNS Server [xxx.xxx.xxx.xx]:
Secondary DNS Server [xxx.xxx.xx.x]:
Connection type (ssh, telnet, none) [ssh]:

Rebooting the SC is required for changes in the above network settings to take
effect.
lom>
```

Use the information in CODE EXAMPLE 2-2 as a guide for the information you need to enter for each parameter value entry.

Installing and Booting the Solaris Operating System

To use LOM commands you must install the Lights Out Management 2.0 packages (`SUNWlomu`, `SUNWlomr` and `SUNWlomm`) from the Solaris Supplementary CD.

▼ To Install and Boot the Solaris Operating System

1. **Access the LOM prompt.**

   For detailed instructions on how you can access the LOM prompt, see Chapter 3.

2. **To power on the system, type** `poweron`

   Depending on the setting of the OpenBoot PROM `auto-boot?` parameter, the system attempts to boot the Solaris OS or remains at the OpenBoot PROM `ok` prompt. The default setting is `true`, which attempts to initiate a boot into the Solaris OS. If the setting of `auto-boot?` is `false` or there is no bootable Solaris image installed, then you get the OpenBoot PROM `ok` prompt.

```
lom>poweron
<POST messages displayed here . . . >
. . .
. . .
ok
```

3. **If necessary, install the Solaris Operating System.**

   See your installation documentation, which is available with your Solaris Operating System release.

   At the `ok` prompt, boot the Solaris Operating System by typing the OpenBoot PROM `boot` command:

```
ok boot [device]
```

   For the optional *device* parameter, see the OpenBoot PROM `devalias` command, which displays the predefined aliases.

After the Solaris Operating System is booted, the `login:` prompt is displayed.

```
login:
```

Installing the Lights Out Management Packages

There are three LOM packages needed on a Sun Fire entry-level midrange system available on the Solaris Supplementary CD.

- `SUNWlomu` (LOMlite Utilities (usr))
- `SUNWlomm` (LOMlite manual pages)
- `SUNWlomr` (LOM drivers).

Note – The latest patches to these packages are available from SunSolve™. It is strongly advised that you obtain the latest versions of patches from SunSolve and install them on your Sun Fire E2900 and Sun Fire V1280/Netra 1280 to make use of the latest LOM utility updates.

▼ To Install the LOM Drivers

- **As root, type:**

CODE EXAMPLE 2-3 Installing the LOM Drivers

```
# pkgadd -d . SUNWlomr

Processing package instance <SUNWlomr> from </var/tmp>

LOMlite driver (root)
(sparc) 2.0,REV=2000.08.22.14.14
Copyright 2000 Sun Microsystems, Inc. All rights reserved.
## Executing checkinstall script.
Using </> as the package base directory.
## Processing package information.
## Processing system information.
   9 package pathnames are already properly installed.
## Verifying package dependencies.
## Verifying disk space requirements.
## Checking for conflicts with packages already installed.
```

CODE EXAMPLE 2-3 Installing the LOM Drivers *(Continued)*

```
## Checking for setuid/setgid programs.

This package contains scripts which will be executed with super-user
permission during the process of installing this package.

Do you want to continue with the installation of <SUNWlomr> [y,n,?] y

Installing LOMlite driver (root) as <SUNWlomr>

## Installing part 1 of 1.
20 blocks
i.drivers (INFO): Starting
i.drivers (INFO): Installing /var/tmp/SUNWlomr/reloc/platform/sun4u/kernel/drv/lom
i.drivers (INFO): Installing /var/tmp/SUNWlomr/reloc/platform/sun4u/kernel/drv/lomp
i.drivers (INFO): Installing /var/tmp/SUNWlomr/reloc/platform/sun4u/kernel/drv/sparcv9/lom
i.drivers (INFO): Installing /var/tmp/SUNWlomr/reloc/platform/sun4u/kernel/drv/sparcv9/lomp
i.drivers (INFO): Installing /var/tmp/SUNWlomr/reloc/platform/sun4u/kernel/drv/sparcv9/lomv

i.drivers (INFO): Identified drivers 'lom lomp lomv'
i.drivers (INFO): Cleaning up old driver 'lom'...
Cleaning up old devlink entry  'type=ddi_pseudo;name=SUNW,lom   lom'
i.drivers (INFO): Cleaning up old driver 'lomp'...
Cleaning up old devlink entry  'type=ddi_pseudo;name=lomp       lomp'
i.drivers (INFO): Cleaning up old driver 'lomv'...
Cleaning up old devlink entry  'type=ddi_pseudo;name=SUNW,lomv  \M0
type=ddi_pseudo;name=lomv       \M0'
i.drivers (INFO): Cleaning up old driver 'lomh'...
Cleaning up old devlink entry  'type=ddi_pseudo;name=SUNW,lomh  lom'

i.drivers (INFO): Adding driver 'lomp'...
  driver  = 'lomp'
  aliases = ''
  link    = 'lomp'
  spec    = 'lomp'

Adding devlink entry 'type=ddi_pseudo;name=lomp lomp'
adding driver with aliases '' perm '* 0644 root sys'
devfsadm: driver failed to attach: lomp
Warning: Driver (lomp) successfully added to system but failed to attach

i.drivers (INFO): Adding driver 'lomv'...
  driver  = 'lomv'
  aliases = 'SUNW,lomv'
  link    = 'SUNW,lomv lomv'
  spec    = '\M0'

Adding devlink entry 'type=ddi_pseudo;name=SUNW,lomv    \M0'
Adding devlink entry 'type=ddi_pseudo;name=lomv \M0'
adding driver with aliases 'SUNW,lomv' perm '* 0644 root sys'
devfsadm: driver failed to attach: lomv
Warning: Driver (lomv) successfully added to system but failed to attach
```

CODE EXAMPLE 2-3 Installing the LOM Drivers *(Continued)*

```
i.drivers (INFO): Adding driver 'lom'...
  driver  = 'lom'
  aliases = 'SUNW,lomh SUNW,lom'
  link    = 'SUNW,lomh SUNW,lom'
  spec    = 'lom'

Adding devlink entry 'type=ddi_pseudo;name=SUNW,lomh    lom'
Adding devlink entry 'type=ddi_pseudo;name=SUNW,lom     lom'
adding driver with aliases 'SUNW,lomh SUNW,lom' perm '* 0644 root sys'
devfsadm: driver failed to attach: lom
Warning: Driver (lom) successfully added to system but failed to attach
i.drivers (SUCCESS): Finished

[ verifying class <drivers> ]

Installation of <SUNWlomr> was successful.
#
```

Note – The WARNING messages concerning `lomp`, `lomv`, and `lom` driver attachment seen during the installation of the `SUWNlomr` package can be safely ignored since the `SUNWlomr` package is not used on the Sun Fire entry-level midrange systems. However, the presence of the package is required to allow successful upgrade through future patches.

▼ To Install the LOM Utility

- **As superuser, type:**

CODE EXAMPLE 2-4 Installing the LOM Utility

```
# pkgadd -d . SUNWlomu

Processing package instance <SUNWlomu> from
</cdrrom/suppcd_s28u7_multi_s28u7_supp.08al1/Lights_Out_Management_2.0/Product>

LOMlite Utilities (usr)
(sparc) 2.0,REV=2000.08.22.14.14
Copyright 2000 Sun Microsystems, Inc. All rights reserved.
Using </> as the package base directory.
## Processing package information.
## Processing system information.
   4 package pathnames are already properly installed.
## Verifying package dependencies.
## Verifying disk space requirements.
## Checking for conflicts with packages already installed.
## Checking for setuid/setgid programs.
```

CODE EXAMPLE 2-4 Installing the LOM Utility

```
Installing LOMlite Utilities (usr) as <SUNWlomu>

## Installing part 1 of 1.
1432 blocks

Installation of <SUNWlomu> was successful.
#
```

▼ To Install the LOM Manual Pages

- **As superuser, type:**

CODE EXAMPLE 2-5 Installing the LOM Manual Pages

```
# pkgadd -d . SUNWlomm

Processing package instance <SUNWlomm> from
</cdrom/suppcd_s28u7_multi_s28u7_supp.08a11/Lights_Out_Management_2.0/Product>

LOMlite manual pages
(sparc) 2.0,REV=2000.08.22.14.14
Copyright 2000 Sun Microsystems, Inc. All rights reserved.
Using </> as the package base directory.
## Processing package information.
## Processing system information.
   5 package pathnames are already properly installed.
## Verifying disk space requirements.
## Checking for conflicts with packages already installed.
## Checking for setuid/setgid programs.

Installing LOMlite manual pages as <SUNWlomm>

## Installing part 1 of 1.
71 blocks

Installation of <SUNWlomm> was successful.
```

Resetting the System

The `reset` command resets the system in the event of a system hang or hardware problem. If the Solaris OS is running then you will be prompted to confirm this action:

▼ To Forcibly Reset the System

- **Type:**

```
lom>reset

This will abruptly terminate Solaris.
Do you want to continue? [no] y
NOTICE: XIR on CPU 3
```

By default, reset uses XIR (externally initiated reset) to reset the CPU processors in the system. The externally initiated reset forces control of the processors into the OpenBoot PROM and begins the OpenBoot PROM error reset recovery actions. The error reset recovery actions preserve most of the Solaris states to allow the collection of data need for debugging the hardware and software, including a Solaris Operating System core file. After saving the debug information, if the value of the OpenBoot PROM variable `auto-boot?` is `true`, the system boots the Solaris OS. The OpenBoot PROM error reset recovery actions are controlled by setting the OpenBoot PROM `error-reset-recovery` configuration variable.

`reset` is prevented in Standby mode, and the message `reset not allowed, domain A keyswitch is set to off` is displayed.

Note – If the system is still hung (you cannot log into the Solaris Operating System), type the `break` command to force control of the system back into the OpenBoot PROM `ok` prompt. If that did not work after you type the `reset` command for the first time, you must next type `reset -a` in order to reset everything.

The `reset -a` command is equivalent to the OpenBoot PROM `reset-all` word.

▼ To Reset the System Controller

- **Use the `resetsc` command to reset the SC. This can be used in the event of a hardware or software problem causing the system controller application to malfunction.**

```
lom>resetsc
Are you sure you want to reboot the system controller now? [no] y
```

This causes the SC to reset, executes the SC POST level specified using the `setupsc` command, and restarts the LOM software.

CHAPTER 3

Navigation Procedures

This chapter explains step-by-step procedures and provides illustrations for connecting to the system and navigating between the LOM shell and the console. It also explains how to terminate an SC session.

This chapter includes the following topics:

Establishing a LOM Console Connection

There are two ways to access the LOM console connection.

- Through the SC serial port (direct) connection.
- Through a Telnet (network) connection using the 10/100 Ethernet port.

Caution – Starting with 5.17.0 firmware, network connections are disabled by default. Unless you enable network connections using the `setupnetwork` command, you must use a serial (direct) connection to access the LOM console.

Under normal operation (when Solaris is running or the system is in the OpenBoot PROM) connecting to the LOM console will automatically select a connection to the Solaris console, otherwise a connection to the LOM prompt is made.

The LOM prompt is:

```
lom>
```

Accessing the LOM Console Using the Serial Port

With the serial port, you can connect to one of three devices:

- ASCII terminal
- Network terminal server
- Workstation

See the *Sun Fire E2900 System Installation Guide* or *Sun Fire V1280/Netra 1280 Systems Installation Guide* for details of how to make the physical connections. The procedure is different for each type of device.

▼ To Connect to an ASCII Terminal

If the LOM password has been set (and the previous connection was logged out) you are prompted for a password.

1. **Enter the correct password as previously set up using the `password` command.**

```
Enter Password:
```

If the password is accepted the SC indicates that a connection has been made.

If the system is in Standby mode, the `lom` prompt is automatically displayed.

```
Connected.

lom>
```

2. **If not, press Return and the Solaris console prompt is displayed.**

```
Connected.

#
```

3. **If a connection to the LOM console is already established over the network port, then you are offered the opportunity to forcefully connect by logging out the other connection:**

```
Enter Password:

The console is already in use.

Host:      somehost.acme.com
Connected: May 24 10:27
Idle time: 00:23:17

Force logout of other user? (y/n) y

Connected.

lom>
```

Otherwise press Return and the Solaris console prompt is displayed.

```
Connected.

#
```

▼ To Connect to a Network Terminal Server

You will be provided with a menu of various servers to which you can connect. Select the required server.

If the LOM password has been set (and the previous connection was logged out) you will be prompted for a password.

1. **Enter the correct password as previously set up using the** `password` **command.**

```
Enter Password:
```

If the password is accepted the SC indicates that a connection has been made.

If the system is in Standby mode, the `lom` prompt is automatically displayed.

```
Connected.

lom>
```

2. **If not, press Return and the Solaris console prompt is displayed.**

```
Connected.

#
```

3. **If a connection to the LOM console is already established over the network port then you are offered the opportunity to forcefully connect by logging out the other connection:**

```
Enter Password:

The console is already in use.

Host:      somehost.acme.com
Connected: May 24 10:27
Idle time: 00:23:17

Force logout of other user? (y/n) y

Connected.

lom>
```

4. **If not, press Return and the Solaris console prompt is displayed.**

```
Connected.

#
```

▼ To Connect to Serial Port B of a Workstation

1. **At the Solaris shell prompt type:**

```
# tip hardwire
```

See the `tip` man page for a complete description of the `tip` command.

If the LOM password has been set (and the previous connection was logged out) you will be prompted for a password.

2. **Enter the correct password as previously set up using the `password` command.**

```
Enter Password:
```

If the password is accepted the SC indicates that a connection has been made.

If the system is in standby mode the `lom` prompt is automatically displayed.

```
Connected.

lom>
```

If not, press Return and the Solaris console prompt is displayed.

```
Connected.

#
```

3. **If a connection to the LOM console is already established over the network port then you are offered the opportunity to forcefully connect by logging out the other connection:**

```
Enter Password:

The console is already in use.

Host:      somehost.acme.com
Connected: May 24 10:27
Idle time: 00:23:17

Force logout of other user? (y/n) y

Connected.

lom>
```

▼ To Access the LOM Console Using a Remote Connection

In order to be able to access the LOM console through a remote connection (for example: an SSH connection) to the 10/100 Ethernet port you must first set up the interface.

Refer to "To Configure Network Parameters" on page 20.

1. **Type the `ssh` command at the Solaris prompt to connect to the SC.**

```
% ssh hostname
```

2. **If the LOM password has been set up you are prompted for a password.**

```
# Enter password:
```

3. **Enter the correct password as previously set up using the `password` command.**

   If the password is accepted the SC indicates that a connection has been made.

   If the system is in Standby mode the `lom` prompt is automatically displayed.

```
Connected.

lom>
```

4. **If not, press Return and the Solaris console prompt will be displayed.**

```
Connected.

#
```

5. **If a connection to the LOM console is already established over the serial port then you will be offered the opportunity to forcefully connect by logging out the other connection:**

```
# ssh hostname

The console is already in use.

Host:      somehost.acme.com
Connected: May 24 10:27
Idle time: 00:23:17

Force logout of other user? (y/n) y

Connected.

lom>
```

In this case you should first use the LOM `logout` command on the serial connection to make the connection available. Refer to the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual* for further details.

▼ To Disconnect From the LOM Console

When you have finished using the LOM console you can disconnect your connection by using the `logout` command.

On the serial port the response is:

```
lom>logout
Connection closed.
```

When connected over the network the response is:

```
lom>logout
Connection closed.
Connection to hostname closed by remote host.
Connection to hostname closed.Connection closed.
$
```

Switching Between the Different Consoles

The SC console connection provides access to the SC LOM command line interface, the Solaris OS, and the OpenBoot PROM.

This section describes the procedures to navigate between the following:

- LOM prompt.
- Solaris OS.
- OpenBoot PROM.

These procedures are summarized in FIGURE 3-1

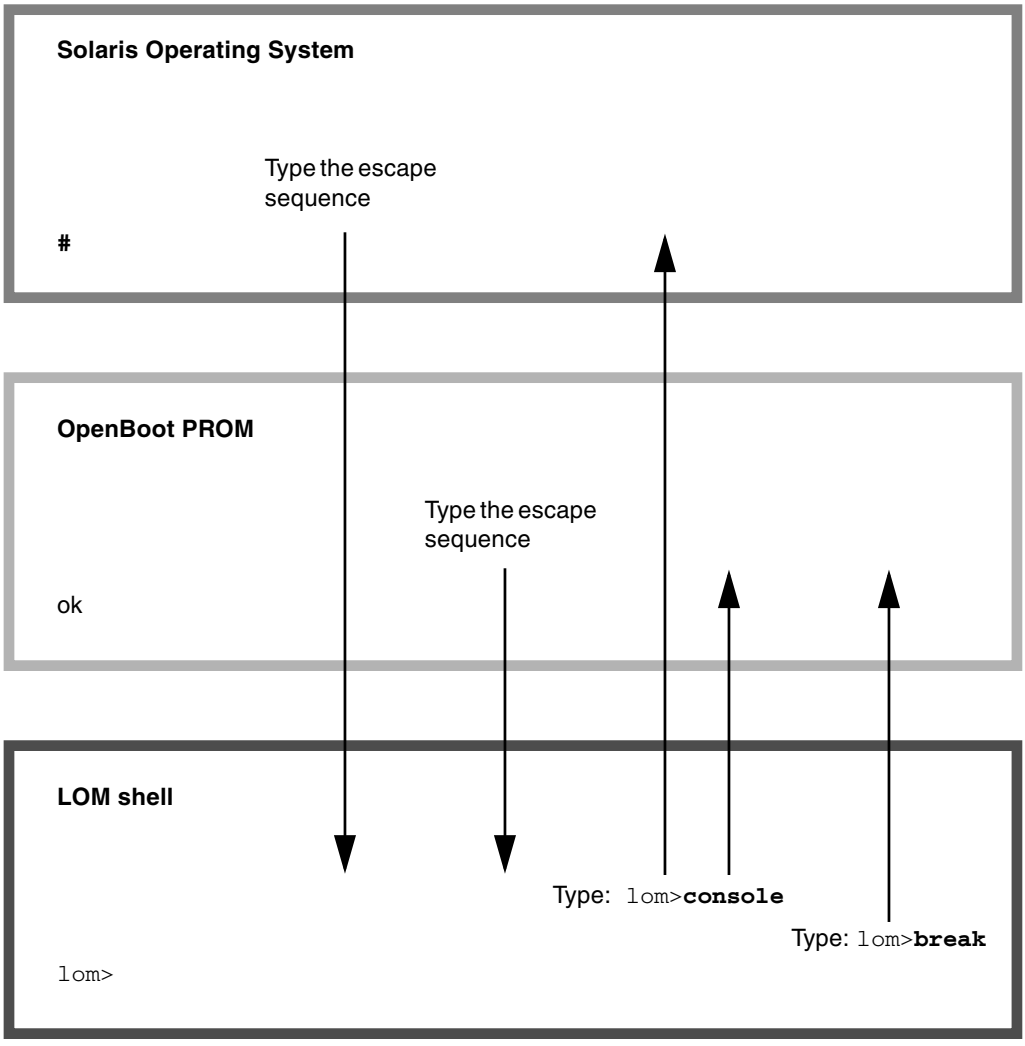

FIGURE 3-1 Navigation Procedures Between Consoles

▼ To Break to the LOM Prompt

- **When connected to the Solaris console, typing the *escape sequence* will take the console into the LOM prompt.**

  By default the escape sequence is set to '**#.**'. That is, a # sign followed by a period.

  For instance, if the escape sequence is the default of `#.` you will see:

```
lom>
```

  When you type the first character of the escape sequence, there is a one second delay before the character appears on the screen. This is because the system waits to see if the next character in the escape sequence is about to be typed. The second character must be typed within this one-second window. If all the characters in the escape sequence are typed then the `lom>` prompt appears. If the next character to be typed is not the next character in the escape sequence, then the characters belonging to the escape sequence that were typed are output to the screen.

  Choose an escape sequence that does not start with a sequence of characters that is frequently typed at the console, otherwise the delay between your striking the keys and the character appearing on the screen may be confusing.

▼ To Connect to the Solaris Console From the LOM Prompt

- **Use the `console` command from the LOM prompt, then type a carriage return.**

  If Solaris software is running, the system responds with the Solaris prompt:

```
lom>console
#
```

If the system was in the OpenBoot PROM, then the system responds with the OpenBoot PROM prompt:

```
lom>console
{2} ok
```

If the system is in Standby mode, the following message is generated:

```
lom>console
Solaris is not active
```

▼ To Break to the LOM Prompt From the OpenBoot PROM

- **The process of moving from the OpenBoot PROM to the LOM prompt is the same as moving from Solaris OS to the LOM prompt.**

  Type the sequence of escape characters (default #.).

```
{2} ok
lom>
```

▼ To Break to the OpenBoot Prompt When Solaris Is Running

- **When the Solaris Operating System is running the usual effect of sending a break signal to the console is to force entry to the OpenBoot PROM or kernel debugger.**

  Do this by using the break command from the LOM prompt:

```
lom>break
This will suspend Solaris.
Do you want to continue? [no] y
Type 'go' to resume
debugger entered.

{1} ok
```

▼ To Terminate a Session if You Are Connected to the System Controller Through the Serial Port

- If you are at the Solaris prompt or the OpenBoot PROM go to LOM prompt by typing the escape sequence, then terminate the LOM prompt session by typing `logout` and pressing Return:

```
lom>logout
```

- If you are connected through a terminal server invoke the terminal server's command to disconnect the connection.
- If the connection was established using a `tip` command then type the `tip` exit sequence `~.`(tilde and a period):

```
~.
```

▼ To Terminate a Session If You Are Connected to the System Controller with a Network Connection

- **If you are at the Solaris prompt or the OpenBoot PROM, go to the LOM prompt by typing the escape sequence and terminate the LOM prompt session by using the `logout` command.**

  The remote session (SSH or Telnet, depending upon the setting chosen using the `setupnetwork` command) terminates automatically:

```
lom>logout
Connection closed by foreign host.
%
```

CHAPTER 4

System Controller Message Logging

The SC generates timestamped messages for system events, processes such as when powering on, booting, powering off, changes to hot-pluggable units, environmental warnings, and so forth.

The messages are initially stored in the SC onboard memory in a circular 128-message buffer (note that a single message can span multiple lines). In addition, the SC sends the messages to the Solaris host when it is running Solaris software, and these are processed by the system log daemon (`syslogd`). When Solaris software is running, messages are sent at the time they are generated by the SC. Retrieval of messages not already copied from the SC takes place at Solaris OS boot time or when the SC is reset.

Messages can also be displayed at the Solaris prompt by using the `lom(1m)` utility (see Chapter 5).

Typically, the messages are stored on the Solaris host in the `/var/adm/messages` file, the only limiting factor being the available disk space.

Messages that are held in the SC message buffer are volatile. Messages are not retained if:

- Power is removed from the SC by loss of both power sources
- Less than two power supplies are operational
- The IB_SSC is removed
- The SC is reset

Messages stored on the system disk are available when the Solaris OS is rebooted.

The display of the messages on the shared Solaris/SC console port, when at the `lom>` prompt, is controlled by the `seteventreporting` command (see the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual*). This determines whether a message is printed at the `lom>` prompt at the time the message is logged, and also whether it is posted to the Solaris logging system so that it is written to `/var/adm/messages`.

Note – Systems equipped with the enhanced memory SC (also known as SC V2), have an additional 112 Kbytes area of SC memory that is used to store firmware messages. This memory is nonvolatile; messages stored there are not deleted when the SC is powered off. The original LOM history buffer is dynamic, losing information when powered off. The messages stored in the persistent history logs of the SC V2 can be displayed at the `lom>` prompt by using the `showlogs -p` command or the `showerrorbuffer -p` command. See the appropriate sections in *Sun Fire Entry-Level Midrange System Controller Command Reference Manual* for their description.

FIGURE 4-1 illustrates the two message buffers.

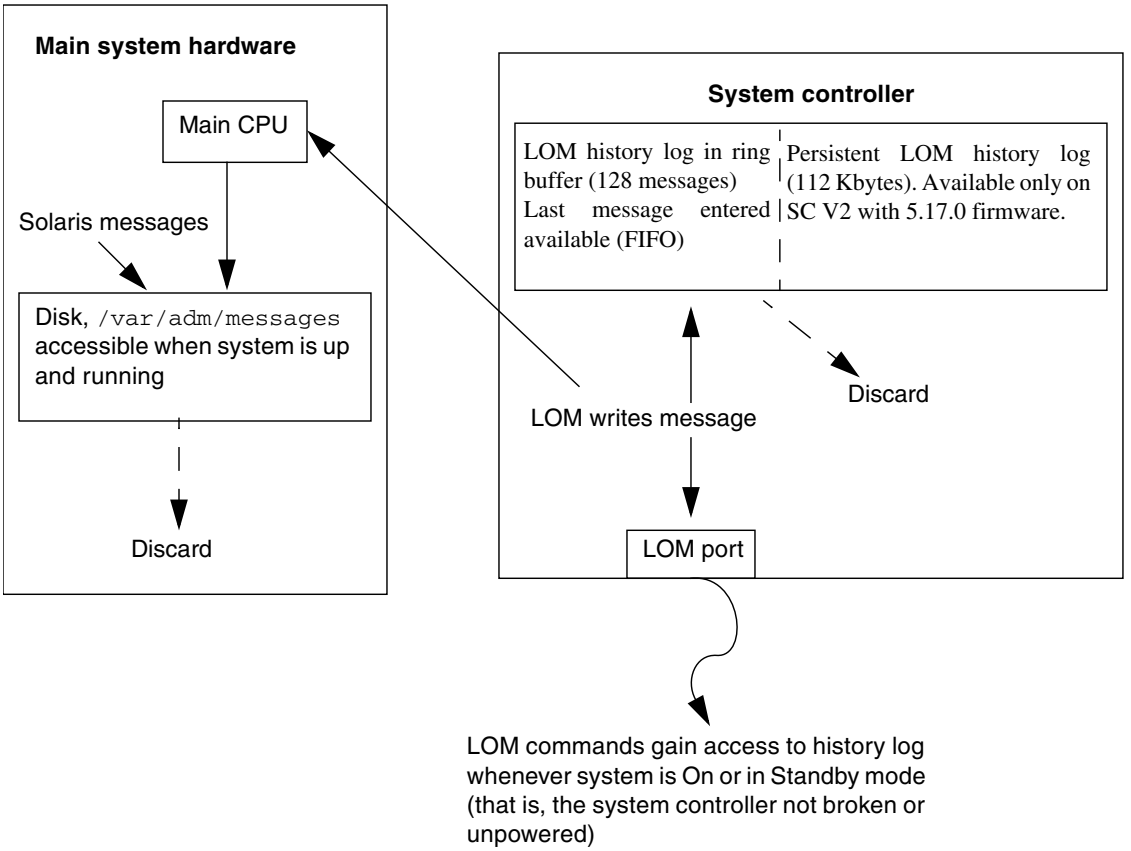

FIGURE 4-1 System Controller Logging

CHAPTER 5

Using Lights Out Management and the System Controller From the Solaris OS

This chapter explains how to use the LOM-specific commands available in the Solaris OS for monitoring and managing a Sun Fire entry-level midrange system. To use these commands, you should install the Lights Out Management 2.0 packages (`SUNWlomr`, `SUNWlomu` and `SUNWlomm`) from the Solaris Supplemental CD. See "Installing the Lights Out Management Packages" on page 23 for a description of how to install the LOM packages.

Note – The latest patches to these packages is available from SunSolve in patch 110208. It is strongly advised that the latest version of patch 110208 be obtained from SunSolve and be installed on the Sun Fire entry-level midrange system to make use of the latest LOM utility updates.

This chapter includes the following topics:

LOM Command Syntax

```
lom [-c] [-l] [-f] [-v] [-t] [-a] [-G] [-X]
lom -e <n>, [x]
lom -A on|off <n>
lom -E on|off
```

where:

`-c` displays LOM configuration.

`-l` displays the status of the Fault and Alarms LEDs.

`-e` displays the event log.

`-f` displays fan status. This information is also displayed in the output from the Solaris `prtdiag -v` command.

`-v` displays the status of the voltage sensors. This information is also displayed in the output from the Solaris `prtdiag -v` command.

`-t` displays temperature information. This information is also displayed in the output from the Solaris `prtdiag -v` command.

`-a` displays all component status data.

`-A` turns alarms on and off.

`-X` changes the escape sequence.

`-E` switches event logging to the console on and off.

`-G` upgrades the firmware.

Monitoring the System From the Solaris OS

There are two ways of interrogating the LOM device (SC) or of sending it commands to perform:

- By executing LOM commands from the `lom>` shell prompt (see Chapter 3).
- By executing LOM-specific Solaris commands from the UNIX # prompt as described in this chapter.

The Solaris commands described in this section, which are all available from the UNIX # prompt, run the `/usr/sbin/lom` utility.

Where appropriate, the command lines given in this section are accompanied by typical output from the commands.

Viewing Online LOM Documentation

- **To view the manual pages for the LOM utility, type:**

```
# man lom
```

Viewing the LOM Configuration (`lom -c`)

- **To view the current LOM configuration, type:**

CODE EXAMPLE 5-1 Sample Output from the `lom -c` Command

```
# lom -c
LOM configuration settings:
serial escape sequence=#.
serial event reporting=default
Event reporting level=fatal, warning & information
firmware version=5.17.0, build 5.0
product ID=Netra T12
```

Checking the Status of the Fault LED and Alarms (`lom -l`)

- **To check whether the System Fault LED and alarms are on or off, type:**

CODE EXAMPLE 5-2 Sample Output from the `lom -l` Command

```
# lom -l
LOM alarm states:
Alarm1=off
Alarm2=off
Alarm3=on
Fault LED=off
#
```

Alarms 1, and 2 are software flags. They are associated with no specific conditions but are available to be set by your own processes or from the command line (see "Turning Alarms On and Off (`lom -A`)" on page 54). For information about alarm 3 (the *system* alarm) and its relation to the watchdog timer, see "Programming Alarm 3" on page 159.

Viewing the Event Log (`lom -e`)

- **To see the event log, type:**

```
# lom -e n,[x]
```

where *n* is the number of reports (up to 128) that you want to see and *x* specifies the level of reports you are interested in. There are four levels of event:

1. Fatal events
2. Warning events
3. Information events
4. User events (not used on Sun Fire entry-level midrange systems systems)

If you specify a level, you will see reports for that level and above. For example, if you specify level 2, you will see reports of level 2 and level 1 events. If you specify level 3, you will see reports of level 3, level 2, and level 1 events.

If you do not specify a level, you will see reports of level 3, level 2, and level 1 events.

CODE EXAMPLE 5-3 shows a sample event log display.

CODE EXAMPLE 5-3 Sample LOM Event Log (Oldest Event Reported First)

```
# lom -e 11
LOMlite Event Log:
 Fri Jul 19 15:16:00 commando-sc lom: Boot: ScApp 5.13.0007, RTOS
23
 Fri Jul 19 15:16:06 commando-sc lom: Caching ID information
 Fri Jul 19 15:16:08 commando-sc lom: Clock Source: 75MHz
 Fri Jul 19 15:16:10 commando-sc lom: /N0/PS0: Status is OK
 Fri Jul 19 15:16:11 commando-sc lom: /N0/PS1: Status is OK
 Fri Jul 19 15:16:11 commando-sc lom: Chassis is in single
partition mode.
 Fri Jul 19 15:27:29 commando-sc lom: Locator OFF
 Fri Jul 19 15:27:46 commando-sc lom: Alarm 1 ON
 Fri Jul 19 15:27:52 commando-sc lom: Alarm 2 ON
 Fri Jul 19 15:28:03 commando-sc lom: Alarm 1 OFF
 Fri Jul 19 15:28:08 commando-sc lom: Alarm 2 OFF
```

Checking the Fans (`lom -f`)

- **To check status of the fans, type:**

CODE EXAMPLE 5-4 Sample Output From the `lom -f` Command

```
# lom -f
Fans:
1 OK speed self-regulating
2 OK speed self-regulating
3 OK speed self-regulating
4 OK speed self-regulating
5 OK speed self-regulating
6 OK speed self-regulating
7 OK speed self-regulating
8 OK speed self-regulating
9 OK speed 100 %
10 OK speed 100 %
#
```

If you need to replace a fan, contact your local Sun sales representative and quote the part number of the component you need. For information, see the *Sun Fire V1280/Netra 1280 Service Manual*.

The information output from this command is also contained in the output from the Solaris `prtdiag -v` command.

Checking the Internal Voltage Sensors (`lom -v`)

The `-v` option displays the status of the Sun Fire entry-level midrange system internal voltage sensors.

- **To check the status of the supply rails and internal voltage sensors, type:**

CODE EXAMPLE 5-5 Sample Output From the `lom -v` Command

```
# lom -v
Supply voltages:
 1 SSC1        v_1.5vdc0    status=ok
 2 SSC1        v_3.3vdc0    status=ok
 3 SSC1        v_5vdc0      status=ok
 4 RP0         v_1.5vdc0    status=ok
 5 RP0         v_3.3vdc0    status=ok
 6 RP2         v_1.5vdc0    status=ok
 7 RP2         v_3.3vdc0    status=ok
 8 SB0         v_1.5vdc0    status=ok
```

CODE EXAMPLE 5-5 Sample Output From the `lom -v` Command *(Continued)*

```
 9 SB0         v_3.3vdc0   status=ok
10 SB0/P0      v_cheetah0  status=ok
11 SB0/P1      v_cheetah1  status=ok
12 SB0/P2      v_cheetah2  status=ok
13 SB0/P3      v_cheetah3  status=ok
14 SB2         v_1.5vdc0   status=ok
15 SB2         v_3.3vdc0   status=ok
16 SB2/P0      v_cheetah0  status=ok
17 SB2/P1      v_cheetah1  status=ok
18 SB2/P2      v_cheetah2  status=ok
19 SB2/P3      v_cheetah3  status=ok
20 IB6         v_1.5vdc0   status=ok
21 IB6         v_3.3vdc0   status=ok
22 IB6         v_5vdc0     status=ok
23 IB6         v_12vdc0    status=ok
24 IB6         v_3.3vdc1   status=ok
25 IB6         v_3.3vdc2   status=ok
26 IB6         v_1.8vdc0   status=ok
27 IB6         v_2.4vdc0   status=ok
System status flags:
 1 PS0         status=okay
 2 PS1         status=okay
 3 FT0         status=okay
 4 FT0/FAN0    status=okay
 5 FT0/FAN1    status=okay
 6 FT0/FAN2    status=okay
 7 FT0/FAN3    status=okay
 8 FT0/FAN4    status=okay
 9 FT0/FAN5    status=okay
10 FT0/FAN6    status=okay
11 FT0/FAN7    status=okay
12 RP0         status=okay
13 RP2         status=okay
14 SB0         status=ok
15 SB0/P0      status=online
16 SB0/P0/B0/D0 status=okay
17 SB0/P0/B0/D1 status=okay
18 SB0/P0/B0/D2 status=okay
19 SB0/P0/B0/D3 status=okay
20 SB0/P1      status=online
21 SB0/P1/B0/D0 status=okay
22 SB0/P1/B0/D1 status=okay
23 SB0/P1/B0/D2 status=okay
24 SB0/P1/B0/D3 status=okay
25 SB0/P2      status=online
26 SB0/P2/B0/D0 status=okay
27 SB0/P2/B0/D1 status=okay
```

CODE EXAMPLE 5-5 Sample Output From the `lom -v` Command *(Continued)*

```
 28 SB0/P2/B0/D2 status=okay
 29 SB0/P2/B0/D3 status=okay
 30 SB0/P3     status=online
 31 SB0/P3/B0/D0 status=okay
 32 SB0/P3/B0/D1 status=okay
 33 SB0/P3/B0/D2 status=okay
 34 SB0/P3/B0/D3 status=okay
 35 SB2        status=ok
 36 SB2/P0     status=online
 37 SB2/P0/B0/D0 status=okay
 38 SB2/P0/B0/D1 status=okay
 39 SB2/P0/B0/D2 status=okay
 40 SB2/P0/B0/D3 status=okay
 41 SB2/P1     status=online
 42 SB2/P1/B0/D0 status=okay
 43 SB2/P1/B0/D1 status=okay
 44 SB2/P1/B0/D2 status=okay
 45 SB2/P1/B0/D3 status=okay
 46 SB2/P2     status=online
 47 SB2/P2/B0/D0 status=okay
 48 SB2/P2/B0/D1 status=okay
 49 SB2/P2/B0/D2 status=okay
 50 SB2/P2/B0/D3 status=okay
 51 SB2/P3     status=online
 52 SB2/P3/B0/D0 status=okay
 53 SB2/P3/B0/D1 status=okay
 54 SB2/P3/B0/D2 status=okay
 55 SB2/P3/B0/D3 status=okay
 56 IB6        status=ok
 57 IB6/FAN0   status=okay
 58 IB6/FAN1   status=okay
 #
```

The information output from this command is also contained in the output from the Solaris `prtdiag -v` command.

Checking the Internal Temperature (`lom -t`)

- **To check the internal temperature of the system and also the system's warning and shutdown threshold temperatures, type:**

CODE EXAMPLE 5-6 Sample Output from the `lom -t` Command

```
# lom -t
System Temperature Sensors:
 1 SSC1       t_sbbc0       36 degC : warning 102 degC : shutdown 107 degC
 2 SSC1       t_cbh0        45 degC : warning 102 degC : shutdown 107 degC
 3 SSC1       t_ambient0    23 degC : warning 82 degC : shutdown 87 degC
 4 SSC1       t_ambient1    21 degC : warning 82 degC : shutdown 87 degC
 5 SSC1       t_ambient2    28 degC : warning 82 degC : shutdown 87 degC
 6 RP0        t_ambient0    22 degC : warning 82 degC : shutdown 87 degC
 7 RP0        t_ambient1    22 degC : warning 53 degC : shutdown 63 degC
 8 RP0        t_sdc0        62 degC : warning 102 degC : shutdown 107 degC
 9 RP0        t_ar0         47 degC : warning 102 degC : shutdown 107 degC
10 RP0        t_dx0         62 degC : warning 102 degC : shutdown 107 degC
11 RP0        t_dx1         65 degC : warning 102 degC : shutdown 107 degC
12 RP2        t_ambient0    23 degC : warning 82 degC : shutdown 87 degC
13 RP2        t_ambient1    22 degC : warning 53 degC : shutdown 63 degC
14 RP2        t_sdc0        57 degC : warning 102 degC : shutdown 107 degC
15 RP2        t_ar0         42 degC : warning 102 degC : shutdown 107 degC
16 RP2        t_dx0         53 degC : warning 102 degC : shutdown 107 degC
17 RP2        t_dx1         56 degC : warning 102 degC : shutdown 107 degC
18 SB0        t_sdc0        48 degC : warning 102 degC : shutdown 107 degC
19 SB0        t_ar0         39 degC : warning 102 degC : shutdown 107 degC
20 SB0        t_dx0         49 degC : warning 102 degC : shutdown 107 degC
21 SB0        t_dx1         54 degC : warning 102 degC : shutdown 107 degC
22 SB0        t_dx2         57 degC : warning 102 degC : shutdown 107 degC
23 SB0        t_dx3         53 degC : warning 102 degC : shutdown 107 degC
24 SB0        t_sbbc0       53 degC : warning 102 degC : shutdown 107 degC
25 SB0        t_sbbc1       40 degC : warning 102 degC : shutdown 107 degC
26 SB0/P0     Ambient       29 degC : warning 82 degC : shutdown 87 degC
27 SB0/P0     Die           57 degC : warning 92 degC : shutdown 97 degC
28 SB0/P1     Ambient       27 degC : warning 82 degC : shutdown 87 degC
29 SB0/P1     Die           51 degC : warning 92 degC : shutdown 97 degC
30 SB0/P2     Ambient       27 degC : warning 82 degC : shutdown 87 degC
31 SB0/P2     Die           53 degC : warning 92 degC : shutdown 97 degC
32 SB0/P3     Ambient       29 degC : warning 82 degC : shutdown 87 degC
33 SB0/P3     Die           50 degC : warning 92 degC : shutdown 97 degC
34 SB2        t_sdc0        51 degC : warning 102 degC : shutdown 107 degC
35 SB2        t_ar0         40 degC : warning 102 degC : shutdown 107 degC
36 SB2        t_dx0         52 degC : warning 102 degC : shutdown 107 degC
37 SB2        t_dx1         54 degC : warning 102 degC : shutdown 107 degC
38 SB2        t_dx2         61 degC : warning 102 degC : shutdown 107 degC
39 SB2        t_dx3         53 degC : warning 102 degC : shutdown 107 degC
```

CODE EXAMPLE 5-6 Sample Output from the `lom -t` Command *(Continued)*

```
 40 SB2          t_sbbc0       52 degC : warning 102 degC : shutdown 107 degC
 41 SB2          t_sbbc1       42 degC : warning 102 degC : shutdown 107 degC
 42 SB2/P0       Ambient       27 degC : warning 82 degC : shutdown 87 degC
 43 SB2/P0       Die           54 degC : warning 92 degC : shutdown 97 degC
 44 SB2/P1       Ambient       26 degC : warning 82 degC : shutdown 87 degC
 45 SB2/P1       Die           53 degC : warning 92 degC : shutdown 97 degC
 46 SB2/P2       Ambient       27 degC : warning 82 degC : shutdown 87 degC
 47 SB2/P2       Die           51 degC : warning 92 degC : shutdown 97 degC
 48 SB2/P3       Ambient       27 degC : warning 82 degC : shutdown 87 degC
 49 SB2/P3       Die           51 degC : warning 92 degC : shutdown 97 degC
 50 IB6          t_ambient0    29 degC : warning 82 degC : shutdown 87 degC
 51 IB6          t_ambient1    29 degC : warning 82 degC : shutdown 87 degC
 52 IB6          t_sdc0        68 degC : warning 102 degC : shutdown 107 degC
 53 IB6          t_ar0         77 degC : warning 102 degC : shutdown 107 degC
 54 IB6          t_dx0         76 degC : warning 102 degC : shutdown 107 degC
 55 IB6          t_dx1         78 degC : warning 102 degC : shutdown 107 degC
 56 IB6          t_sbbc0       51 degC : warning 102 degC : shutdown 107 degC
 57 IB6          t_schizo0     48 degC : warning 102 degC : shutdown 107 degC
 58 IB6          t_schizo1     53 degC : warning 102 degC : shutdown 107 degC
```

The information output from this command is also contained in the output from the Solaris `prtdiag -v` command.

Viewing all Component Status Data and the LOM Configuration Data (`lom -a`)

- **To view all LOM status and configuration data, type:**

```
# lom -a
```

Other LOM Tasks Performed From Solaris

This section explains how to:

- Turn the alarm indicators on and off
- Change the LOM escape sequence
- Stop the LOM from sending reports to the console
- Upgrade the firmware

Turning Alarms On and Off (`lom -A`)

There are two alarms associated with the LOM. They are associated with no specific conditions but are software flags available to be set by your own processes or from the command line.

1. **To turn an alarm on from the command line, type:**

```
# lom -A on,n
```

where *n* is the number of the alarm you want to set: 1 or 2.

2. **To turn the alarm off, type:**

```
# lom -A off,n
```

where *n* is the number of the alarm you want to turn off: 1 or 2.

Changing the lom> Prompt Escape Sequence (lom -X)

The character sequence `#.` (hash, dot) enables you to escape from the Solaris OS to the `lom>` prompt.

- **To change the default escape sequence, type:**

```
# lom -X xy
```

where *xy* are the alpha-numeric characters you want to use.

Note – Quotes may be required for special characters to be interpreted by the shell.

If you are typing at the console and type the first character of the escape sequence, there is a one-second delay before the character appears on the screen. This is because the system waits to see if the next character in the escape sequence is about to be typed. If all the characters in the escape sequence are typed then the `lom>` prompt appears. If the next character to be typed is not the next character in the escape sequence, then the characters belonging to the escape sequence that were typed are output to the screen.

Stopping LOM From Sending Reports to the Console When at the LOM Prompt (lom -E off)

LOM event reports can interfere with information you are attempting to send or receive on the console.

To prevent LOM messages displaying when you are at the LOM prompt, turn off serial event reporting. This is equivalent to the `seteventreporting` command described in the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual*.

1. **To stop the LOM from sending reports to the console, type:**

```
# lom -E off
```

2. **To turn serial event reporting on, type:**

```
# lom -E on
```

Upgrading the Firmware (`lom -G` *filename*)

For a full description, see Chapter 11.

CHAPTER 6

Running POST

Each of the system boards (CPU/Memory boards and IB_SSC assembly) contains a flash PROM that provides storage for power-on self-test (POST) diagnostics. POST tests the following:

- CPU chips
- External cache
- Memory
- Bus interconnect
- I/O ASICs
- I/O buses

POST provides several diagnostic levels that can be selected using the OpenBoot PROM variable `diag-level`. In addition, the `bootmode` command enables the POST settings to be declared for the next system reboot.

There is a separate POST that runs on the SC, which can be controlled using the `setupsc` command.

This chapter includes the following topics:

OpenBoot PROM Variables for POST Configuration

The OpenBoot PROM enables you to set variables that configure how POST runs. These are described in the *OpenBoot 4.x Command Reference Manual*.

You can use the OpenBoot `printenv` command to display the current settings:

```
{3} ok printenv diag-level
diag-level                      init          (init)
```

You can use the OpenBoot PROM `setenv` command to change the current setting of a variable:

```
{1} ok setenv diag-level quick
diag-level=quick
```

For example, you can configure POST to run faster by using:

```
{1} ok setenv diag-level init
diag-level=init
{1} ok setenv verbosity-level off
verbosity-level=off
```

This has the same effect as using the SC command `bootmode skipdiag` at the LOM prompt. The difference is that by using the OpenBoot command the settings remain permanent until you change them again.

TABLE 6-1 POST Configuration Parameters

Parameter	Value	Description
`diag-level`	`init` (default value)	Only system board initialization code is run. No testing is done. This is a very fast pass through POST.
	`quick`	All system board components are tested using few tests with few test patterns.
	`min`	Core functionalities of all system board components are tested. This testing performs a quick sanity check of the devices under test.
	`max`	All system board components are tested with all tests and test patterns, except for memory and Ecache modules. For memory and Ecache modules, all locations are tested with multiple patterns. More extensive, time-consuming algorithms are not run at this level.
	`mem1`	Runs all tests at the default level plus more exhaustive DRAM and SRAM test algorithms.
	`mem2`	This is the same as `mem1` with the addition of a DRAM test that does explicit compare operations of the DRAM data.

TABLE 6-1 POST Configuration Parameters *(Continued)*

Parameter	Value	Description
verbosity-level	off	No status messages are displayed.
	min (default value)	Test names status messages, and error messages are displayed.
	max	Subtest trace messages are displayed.
error-level	off	No error messages are displayed.
	min	The failing test name is displayed.
	max (default value)	All relevant error statuses are displayed.
interleave-scope	within-board (default value)	The memory banks on a system board will be interleaved with each other.
	across-boards	The memory will be interleaved on all memory banks across all of the boards in the system.
interleave-mode	optimal (default value)	The memory is mixed-size interleaving in order to gain optimal performance.
	fixed	The memory is fixed-size interleaving.
	off	There is no memory interleaving.
reboot-on-error	false (default value)	The system is paused when there is an error.
	true	The system is rebooted.
use-nvramrc?		This parameter is the same as the OpenBoot PROM nvramrc? parameter. This parameter uses aliases that are stored in nvramrc.
	true	The OpenBoot PROM executes the script stored in nvramrc if this parameter is set to true.
	false (default value)	The OpenBoot PROM does not evaluate the script stored in nvramrc if this parameter is set to false.
auto-boot?		Controls booting of the Solaris Operating System.
	true (default value)	If this value is true, the system boots automatically after POST has run.
	false	If this parameter value is set to false, you will obtain the OpenBoot PROM ok prompt after POST runs, from which you must type a boot command to boot the Solaris Operating System.
error-reset-recovery		Controls the behavior of the system after an externally initiated reset (XIR) as well as a red mode trap.

TABLE 6-1 POST Configuration Parameters *(Continued)*

Parameter	Value	Description
	`sync` (default value)	The OpenBoot PROM invokes `sync`. A core file is generated. If the invocation returns, the OpenBoot PROM performs a reboot.
	`none`	The OpenBoot PROM prints a message describing the reset trap that triggered the error reset and passes control to the OpenBoot PROM `ok` prompt. The message describing the reset trap type is platform-specific.
	`boot`	The OpenBoot PROM firmware reboots the system. A core file is not generated. Rebooting a system occurs using the OpenBoot PROM settings for `diag-device` or `boot-device`, depending on the value of the OpenBoot PROM configuration variable `diag-switch?` If `diag-switch?` is set to `true`, the device names in `diag-device` will be the default for boot. If `diag-switch?` is set to `false`, the device names in `boot-device` will be the default for boot.

The default output from POST is similar to CODE EXAMPLE 6-1.

CODE EXAMPLE 6-1 POST Output Using `max` Setting

```
Testing CPU Boards ...
Loading the test table from board SB0 PROM 0 ...
{/N0/SB0/P0} Running CPU POR and Set Clocks
{/N0/SB0/P1} Running CPU POR and Set Clocks
{/N0/SB0/P2} Running CPU POR and Set Clocks
{/N0/SB0/P3} Running CPU POR and Set Clocks
{/N0/SB0/P0} @(#) lpost 5.13.0007       2002/07/18 12:45
{/N0/SB0/P2} @(#) lpost 5.13.0007       2002/07/18 12:45
{/N0/SB0/P1} @(#) lpost 5.13.0007       2002/07/18 12:45
{/N0/SB0/P0} Copyright 2001 Sun Microsystems, Inc. All rights reserved.
{/N0/SB0/P0} Subtest: Setting Fireplane Config Registers
{/N0/SB0/P0} Subtest: Display CPU Version, frequency
{/N0/SB0/P0} Version register = 003e0015.21000507
{/N0/SB0/P0} Cpu/System ratio = 6, cpu actual frequency = 900
{/N0/SB0/P1} Copyright 2001 Sun Microsystems, Inc. All rights reserved.
. . .
. . .
. . . <more POST ouput>
. . .
. . .
pci bootbus-controller pci
Probing /ssm@0,0/pci@18,700000 Device 1  Nothing there
Probing /ssm@0,0/pci@18,700000 Device 2  Nothing there
Probing /ssm@0,0/pci@18,700000 Device 3  ide disk cdrom
Probing /ssm@0,0/pci@18,600000 Device 1  Nothing there
```

CODE EXAMPLE 6-1 POST Output Using `max` Setting *(Continued)*

```
Probing /ssm@0,0/pci@18,600000 Device 2  scsi disk tape scsi disk tape
pci pci
Probing /ssm@0,0/pci@19,700000 Device 1  Nothing there
Probing /ssm@0,0/pci@19,700000 Device 2  Nothing there
Probing /ssm@0,0/pci@19,700000 Device 3  Nothing there
Probing /ssm@0,0/pci@19,600000 Device 1  network
Probing /ssm@0,0/pci@19,600000 Device 2  network

Sun Fire V1280
OpenFirmware version 5.13.0007 (07/18/02 12:45)
Copyright 2001 Sun Microsystems, Inc.  All rights reserved.
SmartFirmware, Copyright (C) 1996-2001.  All rights reserved.
16384 MB memory installed, Serial #9537054.
Ethernet address 8:0:xx:xx:xx:xx, Host ID: 80xxxxxx.


NOTICE: obp_main: Extended diagnostics are now switched on.
{0} ok
```

Controlling POST With the `bootmode` Command

The SC `bootmode` command allows you to specify the boot configuration for the next system reboot only. This removes the necessity for taking the system down to the OpenBoot PROM to make these changes, for instance to the `diag-level` variable.

For example, use the following commands to force the highest level of POST tests to be run prior to the next reboot:

```
lom>shutdown
lom>bootmode diag
lom>poweron
```

To force the lowest level of POST tests to be run prior to the next reboot, use:

```
lom>shutdown
lom>bootmode skipdiag
lom>poweron
```

If the system is not rebooted within 10 minutes of the `bootmode` command being issued, the `bootmode` setting is returned to `normal` and the previously-set values of `diag-level` and `verbosity-level` are applied.

For a fuller description of these commands, see the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual*.

Controlling the System Controller POST

The SC power-on self-test is configured using the LOM `setupsc` command. This enables the SC POST level to be set to `off`, `min` or `max`. For a fuller description of this command, see the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual*.

SC POST output appears only on the SC serial connection.

To set the SC POST diagnostic level default to `min`:

CODE EXAMPLE 6-2 Setting SC POST Diagnostic Level to `min`

```
lom>setupsc

 System Controller Configuration
 -------------------------------
 SC POST diag Level [off]: min
 Host Watchdog [enabled]:
 Rocker Switch [enabled]:
 Secure Mode [off]:

 PROC RTUs installed: 8
 PROC Headroom Quantity (0 to disable, 4 MAX) [0]:
 Tolerate correctable memory errors [false]:

 lom>
```

When SC POST `diag-level` is set to `min` you see the following output on the serial port whenever the SC is reset:

CODE EXAMPLE 6-3 SC POST Output With Diagnostic Level Set to `min`

```
@(#) SYSTEM CONTROLLER(SC) POST 21 2001/12/11 17:11
PSR = 0x044010e5
PCR = 0x04004000

        SelfTest running at DiagLevel:0x20

SC Boot PROM             Test
        BootPROM CheckSum               Test
IU        Test
        IU instruction set              Test

        Little endian access            Test
FPU       Test
        FPU instruction set             Test
SparcReferenceMMU     Test
        SRMMU TLB RAM                   Test
        SRMMU TLB Read miss             Test
        SRMMU page    probe             Test
        SRMMU segment probe             Test
        SRMMU region  probe             Test
        SRMMU context probe             Test
. . .
. . .
. . . <more SCPOST ouput>
. . .
. . .
Local I2C AT24C64     Test
        EEPROM       Device             Test
        performing eeprom sequential read

Local I2C PCF8591     Test
        VOLT_AD      Device             Test
        channel[00000001] Voltage(0x00000099) :1.49
        channel[00000002] Voltage(0x0000009D) :3.37
        channel[00000003] Voltage(0x0000009A) :5.1
        channel[00000004] Voltage(0x00000000) :0.0
Local I2C LM75        Test
        TEMP0(IIep)  Device             Test
        Temparature : 24.50 Degree(C)

Local I2C LM75        Test
        TEMP1(Rio)   Device             Test
        Temparature : 23.50 Degree(C)
```

CODE EXAMPLE 6-3 SC POST Output With Diagnostic Level Set to `min` *(Continued)*

```
Local I2C LM75         Test
        TEMP2(CBH)    Device             Test
        Temparature : 32.0 Degree(C)

Local I2C PCF8574      Test
        Sc CSR        Device             Test
Console Bus Hub          Test
        CBH Register Access              Test
POST Complete.
```

CHAPTER 7

Automatic Diagnosis and Recovery

This chapter describes the error diagnosis and domain recovery capabilities included with the firmware for Sun Fire entry-level midrange systems.

This chapter explains the following topics:

Automatic Diagnosis and Recovery Overview

The diagnosis and recovery features are enabled by default on Sun Fire midrange systems. This section provides an overview of how these features work.

Depending on the type of hardware errors that occur and the diagnostic controls that are set, the system controller performs certain diagnosis and recovery steps, as FIGURE 7-1 shows. The firmware includes an *auto-diagnosis* (AD) engine, which detects and diagnoses hardware errors that affect the availability of a system.

Note – Although entry-level midrange systems do *not* support the multiple domains that other midrange systems support, by convention, diagnostic output provides system status as the status for *Domain A*.

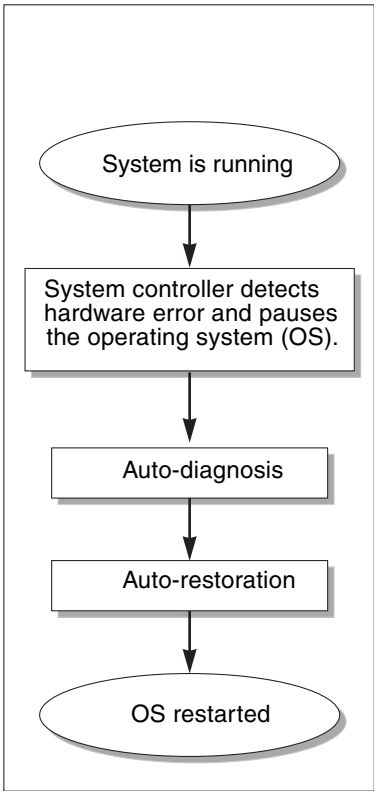

FIGURE 7-1 Auto Diagnosis and Recovery Process

The following summary describes the process shown in FIGURE 7-1:

1. **The SC detects hardware error and pauses the operating system.**
2. **Auto-diagnosis.** The AD engine analyzes the hardware error and determines which field-replaceable units (FRUs) are associated with the hardware error.

   The AD engine provides one of the following diagnosis results, depending on the hardware error and the components involved:

   - Identifies a single FRU that is responsible for the error.
   - Identifies multiple FRUs that are responsible for the error. Be aware that not all components listed may be faulty. The hardware error could be related to a smaller subset of the components identified.
   - Indicates that the FRUs responsible for the error cannot be determined. This condition is considered to be "unresolved" and requires further analysis by your service provider.

The AD engine records the diagnosis information for the affected components and maintains this information as part of the *component health status* (CHS).

The AD reports diagnosis information through console event messages.

- CODE EXAMPLE 7-1 shows an auto-diagnosis event message that appears on the console. In this example, a single FRU is responsible for the hardware error. See "Reviewing Auto-Diagnosis Event Messages" on page 71 for details on the AD message contents.

CODE EXAMPLE 7-1 Example of Auto-Diagnosis Event Message Displayed on the Console

```
[AD] Event: E2900.ASIC.AR.ADR_PERR.10473006
     CSN:  DomainID: A ADInfo: 1.SCAPP.17.0
     Time: Fri Dec 12 09:30:20 PST 2003
     FRU-List-Count: 2; FRU-PN: 5405564; FRU-SN: A08712; FRU-LOC: /N0/IB6
                        FRU-PN: 5404974; FRU-SN: 000274; FRU-LOC: /N0/RP2
     Recommended-Action: Service action required
```

Note – Contact your service provider when you see these auto-diagnosis messages. Your service provider will review the auto-diagnosis information and initiate the appropriate service action.

- Output from the `showlogs`, `showboards`, `showcomponent`, and `showerrorbuffer` commands (see "Obtaining Auto-Diagnosis and Recovery Information" on page 71 for details on the diagnosis-related information displayed by these commands).

  The output from these commands supplements the diagnosis information presented in the event messages and can be used for additional troubleshooting purposes.

3. **Auto-restoration.** During the auto-restoration process, POST reviews the component health status of FRUs that were updated by the AD engine. POST uses this information and tries to isolate the fault by deconfiguring (disabling) any FRUs from the domain that have been determined to cause the hardware error. Even if POST cannot isolate the fault, the system controller then automatically reboots the domain as part of domain restoration.

Note – To take advantage of the automatic recovery feature, make sure that the Openboot PROM variable `hang-policy` is set to `reset`.

Automatic Recovery of a Hung System

The system controller automatically monitors systems for hangs when either of the following occurs:

- The operating system heartbeat stops within a designated timeout period.

  The default timeout value is three minutes, but you can override this value by setting the `watchdog_timeout_seconds` parameter in the domain `/etc/systems` file. If you set the value to less than three minutes, the system controller uses three minutes (the default value) as the timeout period. For details on this system parameter, see the `system`(4) man page of your Solaris Operating System release.

- The system does not respond to interrupts.

When the `host watchdog` (as described in the `setupsc` command) is enabled, the system controller automatically performs an externally initiated reset (XIR) and reboots the hung operating system. If the OpenBoot PROM nvram variable, `error-reset-recovery` is set to `sync`, a core file is also generated after an XIR and can be used to troubleshoot the operating system hang.

CODE EXAMPLE 7-2 shows the console message displayed when the operating system heartbeat stops.

CODE EXAMPLE 7-2 Example of Message Output for Automatic Domain Recovery After the Operating System Heartbeat Stops

```
Tue Dec 09 12:24:47 commando lom: Domain watchdog timer expired.
Tue Dec 09 12:24:48 commando lom: Using default hang-policy (RESET).
Tue Dec 09 12:24:48 commando lom: Resetting (XIR) domain.
```

CODE EXAMPLE 7-3 shows the console message displayed when the operating system does not respond to interrupts.

CODE EXAMPLE 7-3 Example of Console Output for Automatic Recovery After the Operating System Does Not Respond to Interrupts

```
Tue Dec 09 12:37:38 commando lom: Domain is not responding to interrupts.
Tue Dec 09 12:37:38 commando lom: Using default hang-policy (RESET).
Tue Dec 09 12:37:38 commando lom: Resetting (XIR) domain
```

Diagnosis Events

Certain nonfatal hardware errors are identified by the Solaris Operating System and reported to the system controller. The system controller does the following:

- Records and maintains this information for the affected resources as part of the component health status
- Reports this information through event messages displayed on the console.

The next time that POST is run, POST reviews the health status of affected resources and if possible, deconfigures the appropriate resources from the system.

CODE EXAMPLE 7-4 shows an event message for a nonfatal domain error. When you see such event messages, contact your service provider so that the appropriate service action can be initiated. The event message information provided is described in "Reviewing Auto-Diagnosis Event Messages" on page 71.

CODE EXAMPLE 7-4 Domain Diagnosis Event Message – Nonfatal Domain Hardware Error

```
[DOM] Event: SFV1280.L2SRAM.SERD.0.60.10040000000128.7fd78d140
      CSN:  DomainID: A ADInfo: 1.SF-SOLARIS-DE.5_8_Generic_116188-01
      Time: Wed Nov 26 12:06:14 PST 2003
     FRU-List-Count: 1; FRU-PN: 3704129; FRU-SN: 100ACD; FRU-LOC: /N0/SB0/P0/E0
      Recommended-Action: Service action required
```

You can obtain further information about components deconfigured by POST by using the `showboards` and `showcomponent` commands, as described in "Reviewing Component Status" on page 72.

Diagnostic and Recovery Controls

This section explains the various controls and parameters that affect the restoration features.

Diagnostic Parameters

TABLE 7-1 describes the parameter settings that control the diagnostic and operating system recovery process. The default values for the diagnostic and operating system recovery parameters are the recommended settings.

Note – If you do not use the default settings, the restoration features will not function as described in "Automatic Diagnosis and Recovery Overview" on page 65.

TABLE 7-1 Diagnostic and Operating System Recovery Parameters

Parameter	Set Using	Default Value	Description
`Host Watchdog`	`setupsc command`	`enabled`	Automatically reboots the domain when a hardware error is detected. Also boots the Solaris Operating System when the `OBP.auto-boot` parameter is set to `true`.
`Tolerate correctable memory errors`	`setupsc command`	`False`	If set to true it allows the Solaris Operating System to boot with memory exhibiting correctable ECC errors. The Solaris 10 Operating System incorporates features that automatically isolate faulty parts of such memory modules, thus avoiding the need to completely disable these modules and increasing system availability. If set to false, memory modules exhibiting correctable ECC errors are disabled by POST and not allowed to participate in the Solaris domain.
`reboot-on-error`	`OBP setenv`	`true`	Automatically reboots the domain when a hardware error is detected. Also boots the Solaris Operating System when the `OBP.auto-boot` parameter is set to `true`.
`auto-boot`	`OBP setenv`	`true`	Boots the Solaris Operating System after POST runs.
`error-reset-recovery`	`OBP setenv`	`sync`	Automatically reboots the system after an XIR occurs and generates a core file that can be used to troubleshoot the system hang. However, be aware that sufficient disk space must be allocated in the swap area to hold the core file.

Obtaining Auto-Diagnosis and Recovery Information

This section describes various ways to monitor hardware errors and obtain additional information about components associated with hardware errors.

Reviewing Auto-Diagnosis Event Messages

Auto-diagnosis `[AD]` and domain `[DOM]` event messages are displayed on the console and also in the following:

- The `/var/adm/messages` file, provided that you have set up the event reporting appropriately, as described in Chapter 4.
- The `showlogs` command output, which displays the event messages logged on the console.

  In systems with enhanced-memory system controllers (SC V2s), log messages are maintained in a persistent buffer. You can selectively view certain types of log messages according to message type, such as fault event messages, by using the `showlogs -p -f` *filter* command. For details, see the `showlogs` command description in the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual.*

The `[AD]` or `[DOM]` event messages (see CODE EXAMPLE 7-1, CODE EXAMPLE 7-4, CODE EXAMPLE 7-5, and CODE EXAMPLE 7-6) include the following information:

- `[AD]`or `[DOM]` – Beginning of the message. `AD` indicates that the ScApp or POST automatic diagnosis engine generated the event message. `DOM` indicates that the Solaris Operating System on the affected domain generated the automatic diagnosis event message.
- `Event` – An alphanumeric text string that identifies the platform and event-specific information used by your service provider.
- `CSN` – Chassis serial number, which identifies your Sun Fire midrange system.
- `DomainID` – The domain affected by the hardware error. Entry-level midrange systems are always *Domain A*.
- `ADInfo` – The version of the auto-diagnosis message, the name of the diagnosis engine (SCAPP or SF-SOLARIS_DE), and the auto-diagnosis engine version. For domain diagnosis events, the diagnosis engine is the Solaris Operating System (SF-SOLARIS-DE) and the version of the diagnosis engine is the version of the Solaris Operating System in use.

- `Time` – The day of the week, month, date, time (hours, minutes, and seconds), time zone, and year of the auto-diagnosis.
- `FRU-List-Count` – The number of components (FRUs) involved with the error and the following FRU data:
  - If a single component is implicated, the FRU part number, serial number, and location are displayed, as CODE EXAMPLE 7-1 shows.
  - If multiple components are implicated, the FRU part number, serial number, and location for each component involved is reported, as CODE EXAMPLE 7-5 shows.

    In some cases, be aware that not all the FRUs listed are necessarily faulty. The fault may reside in a subset of the components identified.
  - If the SCAPP diagnosis engine cannot implicate specific components, the term `UNRESOLVED` is displayed, as CODE EXAMPLE 7-6 shows.
- `Recommended-Action: Service action required` – Instructs the administrator to contact their service provider for further service action. Also indicates the end of the auto-diagnosis message.

CODE EXAMPLE 7-5 Example of Auto-Diagnostic Message

```
Tue Dec 02 14:35:56 commando lom: ErrorMonitor: Domain A has a SYSTEM ERROR
.
.
.
Tue Dec 02 14:35:59 commando lom: [AD] Event: E2900
 CSN:  DomainID: A ADInfo: 1.SCAPP.17.0
 Time: Tue Dec 02 14:35:57 PST 2003
 FRU-List-Count: 0; FRU-PN:  ; FRU-SN:  ; FRU-LOC: UNRESOLVED
     Recommended-Action: Service action required
Tue Dec 02 14:35:59 commando lom: A fatal condition is detected on Domain A.
Initiating automatic restoration for this domain
```

Reviewing Component Status

You can obtain additional information about components that have been unconfigured as part of the auto-diagnosis process or disabled for other reasons by reviewing the following items:

- The `showboards` command output after an auto-diagnosis has occurred

  CODE EXAMPLE 7-6 shows the location assignments and the status for all components in the system. The diagnostic-related information is provided in the `Status` column for a component. Components that have a `Failed` or `Disabled`

status are deconfigured from the system. The `Failed` status indicates that the board failed testing and is not usable. `Disabled` indicates that the board has been deconfigured from the system, because it was disabled using the `setls` command or because it failed POST. `Degraded` status indicates that certain components on the boards have failed or are disabled, but there are still usable parts on the board. Components with degraded status are configured into the system.

You can obtain additional information about `Failed`, `Disabled`, or `Degraded` components by reviewing the output from the `showcomponent` command.

CODE EXAMPLE 7-6 `showboards` Command Output – `Disabled` and `Degraded` Components

```
Slot      Pwr Component Type                  State       Status
----      --- --------------                  -----       ------
SSC1      On  System Controller V2            Main        Passed
/N0/SCC   -   System Config Card              Assigned    OK
/N0/BP    -   Baseplane                       Assigned    Passed
/N0/SIB   -   Indicator Board                 Assigned    Passed
/N0/SPDB  -   System Power Distribution Bd.   Assigned    Passed
/N0/PS0   On  A166 Power Supply               -           OK
/N0/PS1   On  A166 Power Supply               -           OK
/N0/PS2   On  A166 Power Supply               -           OK
/N0/PS3   On  A166 Power Supply               -           OK
/N0/FT0   On  Fan Tray                        Auto Speed  Passed
/N0/RP0   On  Repeater Board                  Assigned    OK
/N0/RP2   On  Repeater Board                  Assigned    OK
/N0/SB0   On  CPU Board                       Active      Passed
/N0/SB2   On  CPU Board V3                    Assigned    Disabled
/N0/SB4   On  CPU Board                       Active      Degraded
/N0/IB6   On  PCI I/O Board                   Active      Passed
/N0/MB    -   Media Bay                       Assigned    Passed
```

- The `showcomponent` command output after an auto-diagnosis has occurred

  The `Status` column in CODE EXAMPLE 7-7 shows the status for components. The status is either `enabled` or `disabled`. The disabled components are deconfigured from the system. The POST status `chs` (abbreviation for component health status) flags the component for further analysis by your service provider.

Note – Disabled components that have a POST status of `chs` cannot be enabled by using the `setls` command. Contact your service provider for assistance. In some cases, subcomponents belonging to a “parent” component associated with a hardware error will also reflect a disabled status, as will the parent. You cannot re-enable the subcomponents of a parent component associated with a hardware error. Review the auto-diagnosis event messages to determine which parent component is associated with the error.

CODE EXAMPLE 7-7 `showcomponent` Command Output – Disabled Components

```
schostname: SC> showcomponent

Component           Status    Pending POST   Description
---------           ------    ------- ----   -----------
/N0/SB0/P0          disabled  -       chs    UltraSPARC-IV, 1050MHz, 16M ECache
/N0/SB0/P1          disabled  -       chs    UltraSPARC-IV, 1050MHz, 16M ECache
/N0/SB0/P2          disabled  -       chs    UltraSPARC-IV, 1050MHz, 16M ECache
/N0/SB0/P3          disabled  -       chs    UltraSPARC-IV, 1050MHz, 16M ECache
/N0/SB0/P0/B0/L0    disabled  -       chs    empty
/N0/SB0/P0/B0/L2    disabled  -       chs    empty
/N0/SB0/P0/B1/L1    disabled  -       chs    2048M DRAM
/N0/SB0/P0/B1/L3    disabled  -       chs    2048M DRAM
.
.
.
/N0/SB0/P3/B0/L0     disabled -       chs    empty
/N0/SB0/P3/B0/L2     disabled -       chs    empty
/N0/SB0/P3/B1/L1     disabled -       chs    1024M DRAM
/N0/SB0/P3/B1/L3     disabled -       chs    1024M DRAM
/N0/SB4/P0           enabled  -       pass   UltraSPARC-IV, 1050MHz, 16M ECache
/N0/SB4/P1           enabled  -       pass   UltraSPARC-IV, 1050MHz, 16M ECache
/N0/SB4/P2           enabled  -       pass   UltraSPARC-IV, 1050MHz, 16M ECache
/N0/SB4/P3           enabled  -       pass   UltraSPARC-IV, 1050MHz, 16M ECache
.
.
.
```

Reviewing Additional Error Information

For systems configured with enhanced-memory SCs (SC V2s), the `showerrorbuffer -p` command shows the system error contents maintained in the persistent buffer.

However, for systems that do not have enhanced-memory SCs, the `showerrorbuffer` command shows the contents of the dynamic buffer and displays error messages that otherwise might be lost when your domains are rebooted as part of the domain recovery process.

In either case, the information displayed can be used by your service provider for troubleshooting purposes.

CODE EXAMPLE 7-8 shows the output displayed for a domain hardware error.

CODE EXAMPLE 7-8 `showerrorbuffer` Command Output—Hardware Error

```
EX07:
lom>showerrorbuffer
ErrorData[0]
  Date: Fri Jan 30 10:23:32 EST 2004
  Device: /SSC1/sbbc0/systemepld
  Register: FirstError[0x10] : 0x0200
            SB0 encountered the first error
ErrorData[1]
  Date: Fri Jan 30 10:23:32 EST 2004
  Device: /SB0/bbcGroup0/repeaterepld
  Register: FirstError[0x10]: 0x0002
            sdc0 encountered the first error
ErrorData[2]
  Date: Fri Jan 30 10:23:32 EST 2004
  Device: /SB0/sdc0
  ErrorID: 0x60171010
  Register: SafariPortError0[0x200] : 0x00000002
               ParSglErr [01:01] : 0x1 ParitySingle error
```

CHAPTER 8

Security Guidelines

This chapter provides important information about securing the system, explains security recommendations, discusses domain minimization, and provides references to Solaris Operating System security.

This chapter includes the following topics:

Securing the System

The following are security practices to consider:

- Ensure that all passwords comply with security guidelines.
- Change your passwords on a regular basis.
- Scrutinize log files on a regular basis for any irregularities.

The practice of configuring a system to limit unauthorized access is called *hardening*. There are several configuration steps that can contribute to hardening your system. These steps are guidelines for system configuration:

- Implement security modifications immediately after updating the Sun Fire RTOS and SC application firmware and before configuring or installing any Sun Fire domains.
- Aim, in general, to restrict access to the SC operating system, RTOS.
- Limit physical access to serial ports.
- Expect to reboot, depending upon the configuration changes.

Defining the Console Password

The only restrictions on SC console passwords are the character set supported by ASCII and the terminal emulator in use. The SC uses the MD5 algorithm to generate a hash of the password entered. Correspondingly, all characters entered are significant.

A minimum password length of 16 characters promotes the use of pass-phrases instead of passwords. Passwords should be composed of a mixture of lowercase, uppercase, numeric, and punctuation characters. For information on how to set the console password, see "To Set Up the Password" on page 19.

Using the SNMP Protocol Default Configuration

Simple Network Management Protocol (SNMP) is commonly used to monitor and manage networked devices and systems. By default, SNMP is disabled.

Note – The use of Sun Management Center software requires SNMP. However, since the SC does not support a secure version of the SNMP protocol, do not enable SNMP unless you must use Sun Management Center software.

▼ To Reboot the SC to Implement Settings

The SC needs to be rebooted if a console message similar to the following is displayed:

```
Rebooting the SC is required for changes in network settings to
take effect.
```

- **Type `resetsc -y` to reboot the SC**

The SC can be rebooted while the Solaris domain is up and running.

After rebooting the SC, use the `shownetwork` command to validate that all the network modifications were implemented.

For information about using the Sun Security Toolkit to create secure configurations for systems running the Solaris Operating System, see the following web site:

`http://www.sun.com/security/jass`

Selecting a Remote Connection Type

The SSH and Telnet services on the SC are disabled by default.

Enabling SSH

If the SC is on a general purpose network, you can ensure secure remote access to the SC by using SSH rather than Telnet. SSH encrypts data flowing between host and client. It provides authentication mechanisms that identify both hosts and users, enabling secure connections between known systems. Telnet is fundamentally insecure because the Telnet protocol transmits information (including passwords) unencrypted.

Note – SSH does not help with FTP, HTTP, SYSLOG, or SNMPv1 protocols. These protocols are insecure and should be used cautiously on general purpose networks.

The SC provides limited SSH functionality, supporting only SSH version 2 (SSHv2) client requests. TABLE 8-1 identifies the various SSH server attributes and describes how the attributes are handled in this subset. These attribute settings are not configurable.

TABLE 8-1 SSH Server Attributes

Attribute	Value	Comment
Protocol	2	SSH v2 support only
Port	22	Listening port
ListenAddress	0.0.0.0	Support multiple IP addresses
AllowTcpForwarding	no	Port forwarding not supported
RSAAuthentication	no	Public key authentication disabled
PubkeyAuthentication	no	Public key authentication disabled

TABLE 8-1 SSH Server Attributes *(Continued)*

Attribute	Value	Comment
PermitEmptyPasswords	yes	Password authentication controlled by the SC
MACs	hmac-sha1,hmac-md5	Same SSH server implementation as the Solaris 9 Operating System
Ciphers	aes128-cbc,blowfish-cbc,3des-cbc	Same SSH server implementation as the Solaris 9 Operating System

▼ To Enable SSH

1. **To enable SSH, type:**

```
lom> setupnetwork
```

You are prompted to enter the network configuration and connection parameters. For example:

```
lom> setupnetwork

Network Configuration
---------------------
Is the system controller on a network? [yes]:
Use DHCP or static network settings? [static]:
Hostname [hostname]:
IP Address [xxx.xxx.xxx.xxx]:
Netmask [xxx.xxx.xxx.x]:
Gateway [xxx.xxx.xxx.xxx]:
DNS Domain [xxxx.xxx.xxx]:
Primary DNS Server [xxx.xxx.xxx.xx]:
Secondary DNS Server [xxx.xxx.xx.x]:
Connection type (ssh, telnet, none) [ssh]:

Rebooting the SC is required for changes in the above network
settings to take effect.
lom>
```

For detailed information on the `setupnetwork` command, see the command description in the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual*.

Features Not Supported by SSH

The SSH server on Sun Fire entry-level midrange systems does not support the following features:

- Remote command-line execution
- `scp` command (secure copy program)
- `sftp` command (secure file transfer program)
- Port forwarding
- Key-based user authentication
- SSHv1 clients

If you try to use any of the above features, an error message is generated. For example, if you type the following command

```
# ssh SCHOST showboards
```

The following messages are generated:

- On the SSH client:

```
Connection to SCHOST closed by remote host.
```

- On the SC console:

```
[0x89d1e0] sshdSessionServerCreate: no server registered
           for showboards
[0x89d1e0] sshd: Failed to create sshdSession
```

Changing SSH Host Keys

It is good security practice for well-managed machines to get new host keys periodically. If you suspect that the host key might be compromised, you can use the `ssh-keygen` command to regenerate system host keys.

Host keys, once generated, can only be replaced and not deleted without resorting to the `setdefaults` command. For newly generated host keys to be activated, the SSH server must be restarted either by running the `restartssh` command or through a reboot. For further information on the `ssh-keygen` and `restartssh` commands (with examples), see the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual*.

Note – You can also use the `ssh-keygen` command to display the host key fingerprint on the SC.

Additional Security Considerations

This section discusses the following topics:

- Special Key Sequences Allow RTOS Shell Access
- Domain Minimization
- Solaris Operating System Security

Special Key Sequences Allow RTOS Shell Access

Special key sequences can be issued to the SC, over its serial connection, while it is booting. These key sequences have special capabilities if entered at the serial port within the first 30 seconds after an SC reboot.

The special capabilities of these key sequences are automatically disabled 30 seconds after the Sun copyright message is displayed. Once the capability is disabled, the key sequences operate as normal control keys.

Because of the risk that the security of the SC could be compromised by unauthorized access to the RTOS shell, you should control access to the serial ports of the SC.

Domain Minimization

One way to contribute to the security of a Sun Fire midrange system is to tailor the installation of software to an essential minimum. By limiting the number of software components installed on each domain (called *domain minimization*), you can reduce the risks of security holes that can be exploited by potential intruders.

For a detailed discussion of minimization, with examples, see *Minimizing Domains for Sun Fire V1280, 6800, 12K, and 15K Systems* (two-part article) available online at:

`http://www.sun.com/security/blueprints`

Solaris Operating System Security

For information on securing the Solaris Operating System, see the following books and articles:

- *Solaris Security Best Practices* – available online at

  `http://www.sun.com/security/blueprints`

- *Solaris Security Toolkit* – available online at

  `http://www.sun.com/security/jass`

- *Solaris 8 System Administration Supplement* or the *System Administration Guide: Security Services* in the Solaris 9 System Administrator Collection

CHAPTER 9

Capacity on Demand

Sun Fire entry-level midrange systems are configured with processors (CPUs) on CPU/Memory boards. These boards are purchased as part of your initial system configuration or as add-on components. The right to use the CPUs on these boards is included with the initial purchase price.

For systems using UltraSPARC IV CPU/Memory boards, such as the Sun Fire E2900, the Capacity on Demand (COD) option provides additional processing resources that you pay for when you use them. Through the COD option, you purchase and install unlicensed COD CPU/Memory boards in your system. Each COD CPU/Memory board contains four CPUs, which are considered as available processing resources. However, you do not have the right to use these COD CPUs until you also purchase the right-to-use (RTU) licenses for them. The purchase of a COD RTU license entitles you to receive a license key, which enables the appropriate number of COD processors.

You use COD commands included with the firmware for Sun Fire entry-level midrange systems to allocate, activate, and monitor your COD resources.

This chapter covers the following topics:

COD Overview

The COD option provides additional CPU resources on COD CPU/Memory boards that are installed in your system. Although your midrange system comes configured with a minimum number of standard (active) CPU/Memory boards, your system can have a mix of both standard and COD CPU/Memory boards installed, up to the maximum capacity allowed for the system. At least one active CPU is required.

If you want the COD option, and your system is not currently configured with COD CPU/Memory boards, contact your Sun sales representative or authorized Sun reseller to purchase COD CPU/Memory boards. Your salesperson will work with your service provider to install the COD CPU/Memory boards in your system.

The following sections describe the main elements of the COD option:

COD Licensing Process

COD RTU licenses are required to enable COD CPU resources. COD licensing involves the following tasks:

1. Obtaining COD RTU License Certificates and COD RTU license keys for COD resources to be enabled

   You can purchase COD RTU licenses at any time from your Sun sales representative or reseller. You can then obtain a license key (for the COD resources purchased) from the Sun License Center.

2. Entering the COD RTU license keys in the COD license database

   The COD license database stores the license keys for the COD resources that you enable. You record this license information in the COD license database by using the `addcodlicense` command. The COD RTU licenses are considered as floating licenses and can be used for any COD CPU resource installed in the system.

For details on completing the licensing tasks, see "To Obtain and Add a COD RTU License Key to the COD License Database" on page 89.

COD RTU License Allocation

With the COD option, your system is configured to have a certain number of COD CPUs available, as determined by the number of COD CPU/Memory boards and COD RTU licenses that you purchase. The COD RTU licenses that you obtain are handled as a pool of available licenses.

When you activate a domain containing a COD CPU/Memory board or when a COD CPU/Memory board is connected to a domain through a dynamic reconfiguration (DR) operation, the following occurs automatically:

- The system checks the current COD RTU licenses installed.
- The system obtains a COD RTU license (from the license pool) for each CPU on the COD board.

The COD RTU licenses are allocated to the CPUs on a "first come, first serve" basis. However, you can allocate a specific quantity of RTU licenses by using the `setupsc` command. For details, see "To Enable or Disable Instant Access CPUs and Reserve RTU Licenses" on page 93.

If there is an insufficient number of COD RTU licenses and a license cannot be allocated to a COD CPU, the COD CPU is considered as unlicensed. The COD CPU is also assigned a COD-disabled status. If a COD CPU/Memory board does not have sufficient COD RTU licenses for its COD CPUs, the system will fail the COD CPU/Memory board during the poweron operation. For additional details and examples, see "COD-Disabled CPUs" on page 97.

When you remove a COD CPU/Memory board through a Dynamic Reconfiguration operation or when a COD CPU/Memory board is shut down normally, the COD RTU licenses for the CPUs on those boards are released and added to the pool of available licenses.

You can use the `showcodusage` command to review COD usage and COD RTU license states. For details on `showcodusage` and other commands that provide COD information, see "Monitoring COD Resources" on page 94.

Instant Access CPUs

If you require COD CPU resources before you complete the COD RTU license purchasing process, you can temporarily enable a limited number of resources called *instant access CPUs* (also referred to as *headroom)*. These instant access CPUs are available as long as there are unlicensed COD CPUs in the system. The maximum number of instant access resources available on Sun Fire entry-level midrange systems is four CPUs.

Instant access CPUs are disabled by default on Sun Fire midrange systems. If you want to use these resources, you activate them by using the `setupsc` command. Warning messages are logged on the console, informing you that the number of instant access CPUs (headroom) used exceeds the number of COD licenses available. Once you obtain and add the COD RTU license keys for additional instant access CPUs to the COD license database, these warning messages stop.

For details on activating instant access CPUs, see, "To Enable or Disable Instant Access CPUs and Reserve RTU Licenses" on page 93.

Instant Access CPUs as Hot-Spares

You can temporarily enable an available, instant access CPU to replace a failed non-COD CPU. In this case, the instant access CPU is considered as a *hot-spare* (a spare CPU that can be used immediately to replace a failed non-COD CPU). However, once the failed non-COD CPU has been replaced, you must deactivate the instant access CPU (see "To Enable or Disable Instant Access CPUs and Reserve RTU Licenses" on page 93). Contact your Sun sales representative or reseller to purchase a COD RTU license for the instant access CPU in use if you want to continue using it.

Resource Monitoring

Information about COD events, such as the activation of instant access CPUs (headroom) or license violations, is recorded in the console log messages and also in the output for the `showlogs` command.

Other commands, such as the `showcodusage` command, provide information on COD components and COD configuration. For details on obtaining COD information and status, see "Monitoring COD Resources" on page 94.

Getting Started With COD

Before you can use COD on Sun Fire entry-level midrange systems, you must complete certain prerequisites. These tasks include the following:

- Installing the same version of the firmware (starting with release 5.18.0) on both the system controller (SC) and the system boards.

  For details on upgrading the firmware, see "Firmware Update Procedures" on page 119.

Note – Sun Fire entry-level midrange systems firmware before version 5.18.0 will not recognize COD CPU/Memory boards.

- Contacting your Sun sales representative or reseller and doing the following:
  - Signing the COD contract addendum, in addition to the standard purchasing agreement contract for your Sun Fire entry-level midrange system.
  - Purchasing COD CPU/Memory boards and arranging for their installation.
- Performing the COD RTU licensing process as described in "To Obtain and Add a COD RTU License Key to the COD License Database" on page 89.

Managing COD RTU Licenses

COD RTU license management involves the acquisition and addition of COD RTU licenses keys to the COD license database. You can also remove COD RTU licenses from the license database if needed.

Note that COD license key information is always associated with a particular system. You may encounter invalid COD RTU licenses if you do any of the following:

- Move an IB_SSC board from one system to another.
- Replace the SCC card with another card (that is, one with another hostid).

Any COD RTU license keys for the original system now reside on the second system, but the license keys remain associated with the original system. These license keys will be considered invalid. To prevent invalid COD RTU license keys, run the `setdefaults` command on the first system (to set the default system configuration values) before you remove an IB_SSC board. If you do not run the `setdefaults` command on the first system, you can run the command on the second system after you insert the IB_SSC board.

▼ To Obtain and Add a COD RTU License Key to the COD License Database

1. **Contact your Sun sales representative or authorized Sun reseller to purchase a COD RTU license for each COD CPU to be enabled.**

   Sun sends you a COD RTU License Certificate for each CPU license that you purchase. The COD RTU license sticker on the License Certificate contains a right-to-use serial number used to obtain a COD RTU license key.

2. **Contact the Sun License Center and provide the following information to obtain a COD RTU license key:**
   - The COD RTU serial number from the license sticker on the COD RTU License Certificate
   - The Chassis HostID of the system, which identifies your system

     To obtain the Chassis HostID of your system, run the `showsc` command.

   For instructions on contacting the Sun License Center, see the COD RTU License Certificate that you received or check the Sun License Center Web site:

   `http://www.sun.com/licensing`

   The Sun License Center will send you an email message containing the RTU license key for the COD resources that you purchased.

3. **Add the license key to the COD license database by using the `addcodlicense` command. From the conso;e on the SC, type:**

   ```
   lom> addcodlicense license-signature
   ```

   where:

   license-signature is the complete COD RTU license key assigned by the Sun License Center. You can copy the license key string that you receive from the Sun License Center.

4. **Verify that the specified license key was added to the COD license database by running the `showcodlicense -r` command (see "To Review COD License Information" on page 91).**

   The COD RTU license key that you added should be listed in the `showcodlicense` output.

▼ To Delete a COD License Key From the COD License Database

1. **From the console on the SC, type:**

```
lom> deletecodlicense license-signature
```

where:

license-signature is the complete COD RTU license key to be removed from the COD license database.

The system verifies that the license removal will not cause a COD RTU license violation, which occurs when there is an insufficient number of COD licenses for the number of COD resources in use. If the deletion will cause a COD RTU license violation, the SC will not delete the license key.

Note – You can force the removal of the license key by specifying the `-f` option with the `deletecodlicense` command. However, be aware that the license key removal could cause a license violation or an overcommittment of RTU license reservations. An RTU license overcommittment occurs when there are more RTU domain reservations than RTU licenses installed in the system. For additional details, see the `deletecodlicense` command description in the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual*.

---

2. **Verify that the license key was deleted from the COD license database by running the `showcodlicense -r` command, described in the next procedure.**

The deleted license key should not be listed in the `showcodlicense` output.

▼ To Review COD License Information

- **From the console on the SC, do one of the following to display COD license information:**
  - To view license data in an interpreted format, type:

```
lom> showcodlicense
```

For example:

```
lom> showcodlicense
Description  Ver   Expiration  Count  Status
-----------  ---   ----------  -----  ------
PROC          01         NONE      4 GOOD
```

TABLE 9-1 describes the COD license information in the `showcodlicense` output.

TABLE 9-1 COD License Information

Item	Description
Description	Type of resource (processor).
Ver	Version number of the license.
Expiration	None. Not supported (no expiration date).
Count	Number of RTU licenses granted for the given resource.
Status	One of the following states: • GOOD – Indicates the resource license is valid. • EXPIRED – Indicates the resource license is no longer valid.

- To view license data in raw license key format, type:

```
lom> showcodlicense -r
```

The license key signatures for COD resources are displayed. For example:

```
lom> showcodlicense -r
01:83198b89:86017912:0201000000:4:00000000:VW03IcpXYAIO8DYqaF/wSQ
```

Note – The COD RTU license key listed above is provided as an example and is not a valid license key.

For details on the `showcodlicense` command, see the command description in the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual*.

Activating COD Resources

To activate instant access CPUs and allocate COD RTU licenses, use the `setupsc` command. For details on the `setupsc` command options, refer to the command description in the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual*.

▼ To Enable or Disable Instant Access CPUs and Reserve RTU Licenses

1. **From the console on the SC, type:**

```
lom> setupsc
```

You are prompted to enter the COD parameters (headroom quantity and RTU information). For example:

```
lom> setupsc
System Controller Configuration
-------------------------------
SC POST diag Level [off]:
Host Watchdog [enabled]:
Rocker Switch [enabled]:
Secure Mode [off]:

PROC RTUs installed: 4
PROC Headroom Quantity (0 to disable, 4 MAX) [0]: 2
Tolerate correctable memory errors [false]:
```

Note the following about the prompts displayed:

- Instant access CPU (headroom) quantity

  The text in parentheses indicates the maximum number of instant access CPUs (headroom) allowed. The value inside the brackets is the number of instant access CPUs currently configured.

  To disable the instant access CPU (headroom) feature, type 0. You can disable the headroom quantity only when there are no instant access CPUs in use.

2. **Verify the COD resource configuration with the `showsc` command:**

```
lom> showsc
```

For example:

```
lom> showsc

SC: SSC1
System Controller V2
Clock failover disabled.

SC date: Mon May 03 10:22:33 EDT 2004
SC uptime: 3 days 18 hours 4 minutes 4 seconds

ScApp version: 5.18.0
RTOS version: 38

Solaris Host Status: Active - Solaris

Chassis HostID: 83198b89
PROC RTUs installed: 4
PROC Headroom Quantity: 2
```

Monitoring COD Resources

This section describes various ways to track COD resource use and obtain COD information.

COD CPU/Memory Boards

You can determine which CPU/Memory boards in your system are COD boards by using the `showboards` command.

▼ To Identify COD CPU/Memory Boards

- **From the console on the SC, type:**

```
lom> showboards
```

COD CPU/Memory boards are identified as COD CPU boards. For example:

```
lom> showboards

Slot     Pwr Component Type                 State      Status
----     --- --------------                 -----      ------
SSC1     On  System Controller V2           Main       Passed
/N0/SCC  -   System Config Card             Assigned   OK
/N0/BP   -   Baseplane                      Assigned   Passed
/N0/SIB  -   Indicator Board                Assigned   Passed
/N0/SPDB -   System Power Distribution Bd.  Assigned   Passed
/N0/PS0  On  A166 Power Supply              -          OK
/N0/PS1  On  A166 Power Supply              -          OK
/N0/PS2  On  A166 Power Supply              -          OK
/N0/PS3  On  A166 Power Supply              -          OK
/N0/FT0  On  Fan Tray                       Auto Speed Passed
/N0/RP0  On  Repeater Board                 Assigned   OK
/N0/RP2  On  Repeater Board                 Assigned   OK
/N0/SB0  On  COD CPU Board                  Active     Degraded
/N0/SB2  On  COD CPU Board V3               Assigned   Disabled
/N0/SB4  On  COD CPU Board                  Assigned   Disabled
/N0/IB6  On  PCI I/O Board                  Active     Passed
/N0/MB   -   Media Bay                      Assigned   Passed
```

COD Resource Usage

To obtain information on how COD resources are used in your system, use the `showcodusage` command.

▼ To View COD Usage

- **From the console on the SC, type:**

```
lom> showcodusage -v
```

The output includes the status of CPUs. For example:

```
lom>showcodusage -v
Domain/Resource  In Use  Installed  Reserved  Status
---------------  ------  ---------  --------  ------
A - PROC              4          4         0
     SB0 - PROC       4          4
     /N0/SB0/P0                                Licensed
     /N0/SB0/P1                                Licensed
     /N0/SB0/P2                                Licensed
     /N0/SB0/P3                                Licensed
Unused - PROC         0          0         0
```

TABLE 9-2 describes the COD resource information displayed.

TABLE 9-2 `showcodusage` Information

Item	Description
Domain/Resource	The COD resource (processor). An unused processor is a COD CPU that has not yet been assigned.
In Use	The number of COD CPUs currently used.
Installed	The number of COD CPUs installed.
Reserved	The number of COD RTU licenses allocated.
Status	One of the following CPU states: • Licensed – The COD CPU has a COD RTU license. • Unused – The COD CPU is not in use. • Unlicensed – The COD CPU could not obtain a COD RTU license and is not in use.

COD-Disabled CPUs

When you activate a domain that uses COD CPU/Memory boards, any COD CPUs that did not obtain a COD RTU license are disabled by the SC. You can determine which COD CPUs were disabled by reviewing the following items:

- The console log for a `poweron` operation

  Any COD CPUs that did not obtain a COD RTU license are identified as `Cod-dis` (abbreviation for COD-disabled). If all the COD CPUs on a COD/Memory board are disabled, the `poweron` operation will also fail the COD CPU/Memory board, as CODE EXAMPLE 9-1 shows.

CODE EXAMPLE 9-1 Console Log Output Containing Disabled COD CPUs

```
lom> poweron
{/N0/SB0/P0} Passed
{/N0/SB0/P1} Passed
{/N0/SB0/P2} Passed
{/N0/SB0/P3} Passed
{/N0/SB0/P0} Cod-dis
{/N0/SB0/P1} Cod-dis
{/N0/SB0/P2} Cod-dis
{/N0/SB0/P3} Cod-dis
.
.
.
Entering OBP ...
Jun 27 19:04:38 schostname Domain-A.SC: Excluded unusable, unlicensed, failed
or disabled board: /N0/SB0
```

- The `showcomponent` command output

  CODE EXAMPLE 9-2 shows the type of status information displayed for each component in the system. If a COD RTU license cannot be allocated to a COD CPU, the COD CPU status is listed as `Cod-dis` (abbreviation for COD-disabled).

CODE EXAMPLE 9-2 `showcomponent` Command Output – Disabled COD CPUs

```
lom> showcomponent
Component          Status    Pending POST    Description
---------          ------    ------- ----    -----------
.
.
.
/N0/SB2/P0         Cod-dis   -       untest UltraSPARC-IV, 1050MHz, 16M ECache
```

CODE EXAMPLE 9-2 `showcomponent` Command Output – Disabled COD CPUs *(Continued)*

```
/N0/SB2/P1          Cod-dis  -       untest UltraSPARC-IV, 1050MHz, 16M ECache
/N0/SB2/P2          Cod-dis  -       untest UltraSPARC-IV, 1050MHz, 16M ECache
/N0/SB2/P3          Cod-dis  -       untest UltraSPARC-IV, 1050MHz, 16M ECache
/N0/SB2/P0/B0/L0    Cod-dis  -       untest 2048M DRAM
/N0/SB2/P0/B0/L2    Cod-dis  -       untest 2048M DRAM
/N0/SB2/P0/B1/L1    Cod-dis  -       untest 2048M DRAM
/N0/SB2/P0/B1/L3    Cod-dis  -       untest 2048M DRAM
/N0/SB2/P1/B0/L0    Cod-dis  -       untest 2048M DRAM
/N0/SB2/P1/B0/L2    Cod-dis  -       untest 2048M DRAM
/N0/SB2/P1/B1/L1    Cod-dis  -       untest 2048M DRAM
/N0/SB2/P1/B1/L3    Cod-dis  -       untest 2048M DRAM
/N0/SB2/P2/B0/L0    Cod-dis  -       untest 2048M DRAM
.
.
.
```

Other COD Information

TABLE 9-3 summarizes the COD configuration and event information that you can obtain through other system controller commands. For further details on these commands, see their descriptions in the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual*.

TABLE 9-3 Obtaining COD Configuration and Event Information

Command	Description
`showlogs`	Displays information about COD events, such as license violations or headroom activation, that are logged on the console.
`showsc`	Displays the current COD resource configuration and related information: • Number of instant access CPUs (headroom) in use • Chassis HostID

CHAPTER 10

Maintenance and Troubleshooting

This chapter provides maintenance and troubleshooting information for a system administrator. The chapter describes the following topics:

Setting PCI Mode

This release of system firmware (5.19.0) supports the installation of both PCI and PCI-X cards in PCI-X I/O boards on Sun Fire entry-level midrange systems. By default, the firmware enables PCI-X mode on any I/O cards installed into a PCI-X I/O board.

Note – Earlier generation (non-PCI-X) PCI I/O boards are ignored by the PCI-X software, and require no additional user administration.

Caution – While PCI-X I/O boards support both PCI-X and PCI I/O cards, you must use the `forcepci` command to specify the appropriate mode of operation for (non-PCI-X) PCI cards. The `forcepci` command enables you to change the mode from the default (PCI-X mode which has a maximum speed of 100 MHz), to PCI mode (which has a maximum speed of 66 MHz).

▼ To Set a Card in a PCI-X Board to PCI Mode

- **Use the** `forcepci` *board_specification* `on` **Command**

  The command in this example, changes the default operating mode of the card in board slot IB6/P0/B0 to PCI mode. The mode change takes effect after the next power cycle of the I/O assembly.

```
lom> forcepci IB6/P0/B0 on
```

▼ To Display The Mode Status of a Card in a PCI-X Board

- **Use the** `forcepci` *board_specification* `info` **Command**

  This command used in this example displays the mode status of the card in the specified PCI-X board slot.

```
lom> forcepci IB6/P0/B0 info
IB6/P0/B0 is in forced PCI mode
```

▼ To Reset a Card in a PCI-X Board Slot to PCI-X Mode

- **Use the** `forcepci` *board_specification* `off` **Command**

  The command in this example changes the default operating mode of the card in board slot IB6/P0/B0 to PCI-X mode (the default mode). The mode change takes effect after the next power cycle of the I/O assembly.

```
lom> forcepci IB6/P0/B0 off
```

Device Mapping

The physical address represents a physical characteristic that is unique to the device. Examples of physical addresses include the bus address and the slot number. The slot number indicates where the device is installed.

You reference a physical device by the node identifier—agent ID (AID). The AID ranges from 0 to 31 in decimal notation (0 to 1f in hexadecimal). In the device path beginning with `ssm@0,0` the first number, 0, is the node ID.

CPU/Memory Mapping

CPU/Memory board and memory agent IDs (AIDs) range from 0 to 23 in decimal notation (0 to 17 in hexadecimal). The system can have up to three CPU/Memory boards.

Each CPU/Memory board has four CPUs, depending on your configuration. Each CPU/Memory board has up to four banks of memory. Each bank of memory is controlled by one memory management unit (MMU), which is the CPU. The following code example shows a device tree entry for a CPU and its associated memory:

```
/ssm@0,0/SUNW/UltraSPARC-III@b,0 /ssm@0,0/SUNW/memory-controller@b,400000
```

where:

in `b,0`

- b is the CPU agent identifier (AID)
- 0 is the CPU register

in `b,400000`

- b is the memory agent identifier (AID)
- 400000 is the memory controller register

There are up to four CPUs on each CPU/Memory board (TABLE 10-1):

- CPUs with agent IDs 0–3 reside on board name SB0

- CPUs with agent IDs 8–11 on board name SB2, and so on.

TABLE 10-1 CPU and Memory Agent ID Assignment

CPU/Memory Board Name	Agent IDs On Each CPU/Memory Board			
	CPU 0	CPU 1	CPU 2	CPU 3
SB0	0 (0)	1 (1)	2 (2)	3 (3)
SB2	8 (8)	9 (9)	10 (a)	11 (b)
SB4	16 (10)	17 (11)	18 (12)	19 (13)

The first number in the columns of agent IDs is a decimal number. The number or letter in parentheses is in hexadecimal notation.

IB_SSC Assembly Mapping

TABLE 10-2 lists the types of I/O assembly, the number of slots each I/O assembly has, and the systems the I/O assembly types are supported on.

TABLE 10-2 I/O Assembly Type and Number of Slots

I/O Assembly Type	Number of Slots Per I/O Assembly
PCI	6

TABLE 10-3 lists the number of I/O assemblies per system and the I/O assembly name.

TABLE 10-3 Number and Name of I/O Assemblies per System

Number of I/O Assemblies	I/O Assembly Name
1	IB6

Each I/O assembly hosts two I/O controllers:

- I/O controller 0
- I/O controller 1

When mapping the I/O device tree entry to a physical component in the system, you must consider up to five nodes in the device tree:

- Node identifier (ID)
- I/O controller agent ID (AID)
- Bus offset
- PCI slot
- Device instance

TABLE 10-4 lists the AIDs for the two I/O controllers in each I/O assembly.

TABLE 10-4 I/O Controller Agent ID Assignments

Slot Number	I/O Assembly Name	Even I/O controller AID	Odd I/O Controller AID
6	IB6	24 (18)	25 (19)

The first number in the column is a decimal number. The number (or a number and letter combination) in parentheses is in hexadecimal notation.

The I/O controller has two bus sides: A and B.

- Bus A, which is 66 MHz, is referenced by offset 600000.
- Bus B, which is 33 MHz, is referenced by offset 700000.

The board slots located in the I/O assembly are referenced by the device number.

This section describes the PCI I/O assembly slot assignments and provides an example of the device path.

The following code example gives a breakdown of a device tree entry for a SCSI disk:

```
/ssm@0,0/pci@19,700000/pci@3/SUNW,isptwo@4/sd@5,0
```

Note – The numbers in the device path are hexadecimal.

where:

in `19,700000`

- 19 is the I/O controller agent identifier (AID)
- 700000 is the bus offset

in `pci@3`

- 3 is the device number

`isptwo` is the SCSI host adapter

in `sd@5,0`

- 5 is the SCSI target number for the disk
- 0 is the logic unit number (LUN) of the target disk

This section describes the PCI I/O assembly slot assignments and provides an example of the device path.

TABLE 10-5 lists, in hexadecimal notation, the slot number, I/O assembly name, device path of each I/O assembly, the I/O controller number, and the bus.

TABLE 10-5 IB_SSC Assembly PCI Device Mapping

I/O Assembly Name	Device Path	Physical Slot Number	I./O Controller Number	Bus
IB6	/ssm@0,0/pci@18,700000/*@1	0	0	B
	/ssm@0,0/pci@18,700000/*@2	1	0	B
	/ssm@0,0/pci@18,700000/*@3	x	0	B
	/ssm@0,0/pci@18,600000/*@1	5	0	A
	/ssm@0,0/pci@18,600000/*@2	w	0	A
	/ssm@0,0/pci@19,700000/*@1	2	1	B
	/ssm@0,0/pci@19,700000/*@2	3	1	B
	/ssm@0,0/pci@19,700000/*@3	4	1	B
	/ssm@0,0/pci@19,600000/*@1	y	1	A
	/ssm@0,0/pci@19,600000/*@2	z	1	A

where:

w = onboard LSI1010R SCSI controller

x = onboard CMD646U2 EIDE controller

y = onboard Gigaswift Ethernet controller 0

z = onboard Gigaswift Ethernet controller 1

and * is dependent upon the type of PCI card installed in the slot.

Note the following:

- `600000` is the bus offset and indicates bus A, which operates at 66 MHz.
- `700000` is the bus offset and indicates bus B, which operates at 33 MHz.
- `*@3` is the device number. In this example `@3` means it is the third device on the bus.

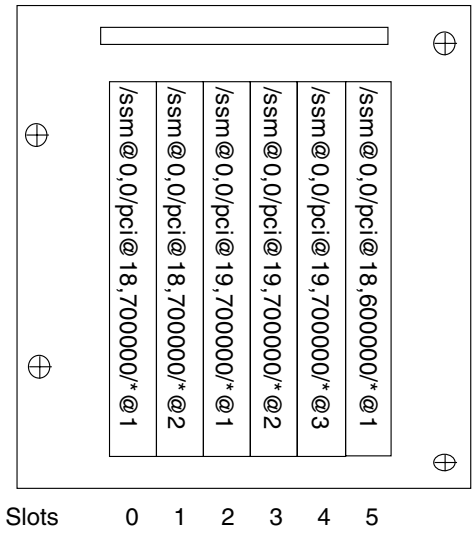

FIGURE 10-1 Sun Fire Entry-Level Midrange Systems IB_SSC PCI Physical Slot Designations for IB6

where * is dependent upon the type of PCI card installed in the slot.

For instance:

- Dual Differential Ultra SCSI card (375-0006) in Slot 4
- FC-AL card (375-3019) in Slot 3
- FC-AL card (375-3019) in Slot 2

These would generate device paths as follows:

```
/ssm@0,0/pci@19,700000/scsi@3,1
/ssm@0,0/pci@19,700000/scsi@3,1 (scsi-2)
/ssm@0,0/pci@19,700000/scsi@3,1/tape (byte)
/ssm@0,0/pci@19,700000/scsi@3,1/disk (block)
/ssm@0,0/pci@19,700000/scsi@3 (scsi-2)
/ssm@0,0/pci@19,700000/scsi@3/tape (byte)
/ssm@0,0/pci@19,700000/scsi@3/disk (block)

/ssm@0,0/pci@19,700000/SUNW,qlc@2 (scsi-fcp)
/ssm@0,0/pci@19,700000/SUNW,qlc@2/fp@0,0 (fp)
/ssm@0,0/pci@19,700000/SUNW,qlc@2/fp@0,0/disk (block)

/ssm@0,0/pci@19,700000/SUNW,qlc@1 (scsi-fcp)
/ssm@0,0/pci@19,700000/SUNW,qlc@1/fp@0,0 (fp)
/ssm@0,0/pci@19,700000/SUNW,qlc@1/fp@0,0/disk (block)
```

System Faults

A system fault is any condition that is considered to be unacceptable for normal system operation. When the system has a fault, the Fault LED ( ) turns on. The system indicators are shown in FIGURE 10-2.

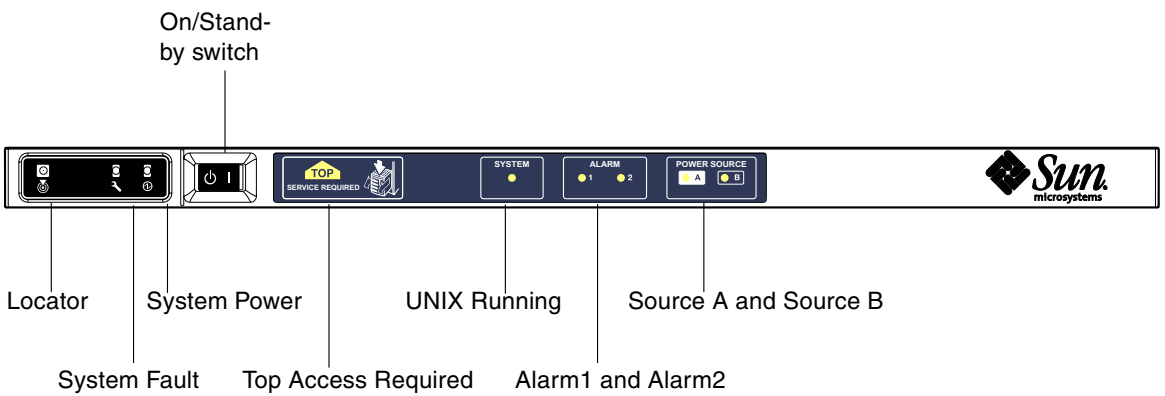

FIGURE 10-2 System Indicators

The indicator states are shown in TABLE 10-6. You must take immediate action to eliminate a system fault.

TABLE 10-6 System Fault Indicator States

FRU name	Fault indicator lit when fault detected*	System Fault indicator lit on FRU fault*	Top Access lit on FRU fault[1]	Comments
System Board	Yes	Yes	Yes	Includes processors, Ecache and DIMMs
Level 2 repeater	Yes	Yes	Yes	
IB_SSC	Yes	Yes	Yes	
System Controller	No	Yes	Yes	IB_SSC fault LED lit
Fan	Yes	Yes	Yes	IB Fan fault LED lit
Power Supply	Yes (by hardware)	Yes	No	All power supply indicators are lit by the power supply hardware. There is also a predicted fault indicator. Power supply EEPROM errors do not cause degraded state as there is no indicator control.
Power distribution board	No	Yes	Yes	Can only be degraded.
Baseplane	No	Yes	Yes	Can only be degraded.
System indicator board	No	Yes	Yes	Can only be degraded.
System configuration card	No	Yes	No	
Fan tray	Yes	Yes	No	
Main fan	Yes	Yes	No	
Media bay	No	Yes	Yes	
Disk	Yes	Yes	No	

* This includes faults where the FRU is only degraded.

1 If lit, indicates the failing FRU is accessed from the top of the platform. It is important that you employ the anti-tip legs on the cabinet before extending the platform out on its rails.

Customer Replaceable Units

The following topics describe the field replaceable units, by system.

Sun Fire E2900 System

The following FRUs are considered to be ones on which you can deal with faults:

- Hard disks – hot-swappable
- PSUs (PS0/PS1/PS2/PS3) – hot-swappable
- CPU/Memory boards (SB0/SB2/SB4) – can be blacklisted if considered faulty
- Repeater boards (RP0/RP2) – can be blacklisted if considered faulty

If a fault is indicated on any other FRU or a physical replacement of blacklisted FRUs above is required, then Sun Service should be called.

Sun Fire V1280 System

The following FRUs are considered to be ones on which you can deal with faults:

- Hard disks – hot-swappable
- PSUs (PS0/PS1/PS2/PS3) – hot-swappable
- CPU/Memory boards (SB0/SB2/SB4) – can be blacklisted if considered faulty
- Repeater boards (RP0/RP2) – can be blacklisted if considered faulty

If a fault is indicated on any other FRU or a physical replacement of blacklisted FRUs above is required, then Sun Service should be called.

Netra 1280 System

The following FRUs are considered to be ones on which you can deal with faults:

- Hard disks – hot swappable
- PSUs (PS0/PS1/PS2/PS3) – hot-swappable

Note – Only suitably trained personnel or Sun Service are permitted to enter the Restricted Access Location to hot-swap PSUs or hard disk drives.

- CPU/Memory boards (SB0/SB2/SB4) – can be blacklisted if considered faulty
- Repeater boards (RP0/RP2) – can be blacklisted if considered faulty

If a fault is indicated on any other FRU or a physical replacement of blacklisted FRUs above is required, then Sun Service should be called.

Manual Blacklisting (While Waiting for Repair)

The SC supports the blacklisting feature, which allows you to disable components on a board (TABLE 10-7).

Blacklisting provides a list of system board components that will not be tested and will not be configured into the Solaris Operating System. The blacklist is stored in nonvolatile memory.

TABLE 10-7 Blacklisting Component Names

System Component	Component Subsystem	Component Name
CPU system		*slot/port/physical-bank/logical-bank*
	CPU/Memory boards (*slot*)	SB0, SB2, SB4
	Ports on the CPU/Memory board	P0, P1, P2, P3
	Physical memory banks on CPU/Memory boards	B0, B1
	Logical banks on CPU/Memory boards	L0, L1, L2, L3
I/O assembly system		*slot*/*port*/*bus* or *slot*/*card*
	I/O assembly	IB6
	Ports on the I/O assembly	P0, P1
	Buses on the I/O assembly	B0, B1
	I/O cards in the I/O assemblies	C0, C1, C2, C3, C4, C5
Repeater system		<*slot*>
	Repeater board	RP0, RP2

Blacklist a component or device if you believe it might be failing intermittently or is failing. Troubleshoot a device you believe is having problems.

There are two system controller commands for blacklisting:

- `setls`
- `showcomponent`

Note – The `enablecomponent` and `disablecomponent` commands have been replaced by the `setls` command. These commands were formerly used to manage component resources. While the `enablecomponent` and `disablecomponent` commands are still available, it is suggested that you use the `setls` command to control the configuration of components into or out of the system.

The `setls` command updates only the blacklist. It does not directly affect the state of the currently configured system boards.

The updated lists take effect when you do one the following:

- Reboot the system.
- Use dynamic reconfiguration to configure the board containing the blacklisted component out of and then back into the system.

In order to use `setls` on the Repeater boards (RP0/RP2), the system first has to be shut down to Standby using the `poweroff` command.

When the `setls` command is issued for a Repeater board (RP0/RP2), the SC will be automatically reset to make use of the new settings.

If a replacement Repeater board is inserted, it is necessary to manually reset the SC using the `resetsc` command. See the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual* for a description of this command.

Special Considerations for CPU/Memory Boards

In the unlikely event that a CPU/Memory board fails the interconnect test during POST, a message similar to the following appears in POST output:

```
Jul 15 15:58:12 noname lom: SB0/ar0 Bit in error P3_ADDR [2]
Jul 15 15:58:12 noname lom: SB0/ar0 Bit in error P3_ADDR [1]
Jul 15 15:58:12 noname lom: SB0/ar0 Bit in error P3_ADDR [0]
Jul 15 15:58:12 noname lom: AR Interconnect test: System board SB0/ar0 address
repeater connections to system board RP2/ar0 failed
Jul 15 15:58:13 noname lom: SB0/ar0 Bit in error P3_INCOMING [0]
Jul 15 15:58:17 noname lom: SB0/ar0 Bit in error P3_PREREQ [0]
Jul 15 15:58:17 noname lom: SB0/ar0 Bit in error P3_ADDR [18]
Jul 15 15:58:17 noname lom: SB0/ar0 Bit in error P3_ADDR [17]
```

A CPU/Memory board failing the interconnect test might prevent the `poweron` command from completely powering on the system. The system then drops back to the `lom>` prompt.

As a provisional measure, before service intervention is obtained, the faulty CPU/Memory board can be isolated from the system using the following sequence of commands at the SC `lom>` prompt:

```
lom>disablecomponent SBx
.
.
lom>poweroff
.
.
lom>resetsc -y
```

A subsequent `poweron` command should now be successful.

Recovering a Hung System

If you cannot log into the Solaris Operating System, and typing the `break` command from the LOM shell did not force control of the system back to the OpenBoot PROM `ok` prompt, then the system has stopped responding.

In some circumstances the host watchdog detects that the Solaris Operating System has stopped responding and automatically resets the system.

Assuming that the host watchdog has not been disabled (using the `setupsc` command), then the Host Watchdog causes an automatic reset of the system.

Also, you can issue the `reset` command (default option is `-x` which causes an XIR to be sent to the processors) from the `lom>` prompt. The `reset` command causes the Solaris Operating System to be terminated.

Caution – When the Solaris Operating System is terminated, data in memory might not be flushed to disk. This could cause a loss or corruption of the application file system data. Before the Solaris Operating System is terminated, this action requires confirmation from you.

▼ To Recover a Hung System Manually

1. **Complete the steps in "Assisting Sun Service Personnel in Determining Causes of Failure" on page 116.**
2. **Access the LOM shell.**

   See Chapter 3.
3. **Type the `reset` command to force control of the system back to the OpenBoot PROM.**

   The `reset` command sends an externally initiated reset (XIR) to the system and collects data for debugging the hardware.

```
lom>reset
```

Note – An error is displayed if the `setsecure` command has been used to set the system into secure mode. You cannot use the `reset` or `break` commands while the system is in secure mode. See the *Sun Fire Entry-Level Midrange System Controller Command Reference Manual* for more details.

4. **This step depends on the setting of the Open Boot PROM `error-reset-recovery` configuration variable.**
   - If the `error-reset-recovery` configuration variable is set to `none`, the system returns immediately to the OpenBoot PROM. When the OpenBoot PROM takes control, it takes actions based on the setting of the OpenBoot PROM `error-reset-recovery` configuration variable. You can type any OpenBoot PROM command from the `ok` prompt, including rebooting the Solaris Operating System with the `boot` command. Also, you can force a core file with the `sync` command. The actions that can be configured by this variable might mean that the system will not return to the `ok` prompt.
   - If the `error-reset-recovery` configuration variable is *not* set to `none`, the OpenBoot PROM automatically takes recovery actions.
   - If the `error-reset-recovery` configuration variable is set to `sync` (default), the system generates a Solaris Operating System core file and reboots the system.
   - If the OpenBoot PROM `error-reset-recovery` configuration variable is set to `boot`, the system is rebooted.

5. **If the previous actions fail to reboot the system, use the `poweroff` and `poweron` commands to power cycle the system.**

   To power off the system, type:

```
lom>poweroff
```

   To power on the system, type:

```
lom>poweron
```

Moving System Identity

You might decide that the simplest way to restore service is to use a complete replacement system. In order to facilitate the rapid transfer of system identity and critical settings from one system to its replacement, the System Configuration Card (SCC) can be physically removed from the SCC Reader (SCCR) of the faulty system and inserted into the SCCR of the replacement system.

The following information is stored on the System Configuration Card (SCC):

- MAC addresses
  - System controller 10/100 Ethernet Port
  - Onboard Gigabit Ethernet port NET0
  - Onboard Gigabit Ethernet port NET1
- Hostid
- Critical LOM configurations
  - LOM password
  - escape sequence
  - SC network settings (IP address / DHCP / gateway, and so on)
  - `eventreporting` level
  - host watchdog enabled/disabled
  - On/Standby enabled/disabled
  - secure mode enabled/disabled
- Critical OpenBoot PROM configurations
  - `auto-boot?`
  - `boot-device`
  - `diag-device`
  - `use-nvramrc?`
  - `local-mac-address?`

Temperature

One indication of problems might be overtemperature of one or more components. Use the `showenvironment` command to list current status.

TABLE 10-8 Checking Temperature Conditions Using the `showenvironment` Command

```
lom>showenviroment

Slot Device    Sensor    Value  Units     Age     Status
---- --------- --------- ------ --------- ------- ------
SSC1 SBBC 0    Temp. 0    34    Degrees C   1 sec OK
SSC1 CBH 0     Temp. 0    41    Degrees C   1 sec OK
SSC1 Board 0   Temp. 0    22    Degrees C   1 sec OK
SSC1 Board 0   Temp. 1    22    Degrees C   1 sec OK
SSC1 Board 0   Temp. 2    28    Degrees C   1 sec OK
SSC1 Board 0   1.5 VDC 0   1.49 Volts DC    1 sec OK
SSC1 Board 0   3.3 VDC 0   3.35 Volts DC    1 sec OK
SSC1 Board 0   5 VDC 0     4.98 Volts DC    1 sec OK
/N0/PS0 Input 0   Volt. 0        - -           1 sec OK
/N0/PS0 48 VDC 0  Volt. 0    48.00 Volts DC    1 sec OK
/N0/PS1 Input 0   Volt. 0        - -           5 sec OK
```

TABLE 10-8 Checking Temperature Conditions Using the `showenvironment` Command *(Continued)*

```
/N0/PS1 48 VDC 0  Volt. 0    48.00 Volts DC    5 sec OK
/N0/FT0 Fan 0     Cooling 0   Auto             5 sec OK
/N0/FT0 Fan 1     Cooling 0   Auto             5 sec OK
/N0/FT0 Fan 2     Cooling 0   Auto             5 sec OK
/N0/FT0 Fan 3     Cooling 0   Auto             5 sec OK
/N0/FT0 Fan 4     Cooling 0   Auto             5 sec OK
/N0/FT0 Fan 5     Cooling 0   Auto             5 sec OK
/N0/FT0 Fan 6     Cooling 0   Auto             5 sec OK
/N0/FT0 Fan 7     Cooling 0   Auto             5 sec OK
/N0/RP0 Board 0   1.5 VDC 0   1.49 Volts DC    5 sec OK
/N0/RP0 Board 0   3.3 VDC 0   3.37 Volts DC    5 sec OK
/N0/RP0 Board 0   Temp. 0    20    Degrees C   5 sec OK
/N0/RP0 Board 0   Temp. 1    19    Degrees C   5 sec OK
/N0/RP0 SDC 0     Temp. 0    55    Degrees C   5 sec OK
/N0/RP0 AR 0      Temp. 0    45    Degrees C   5 sec OK
/N0/RP0 DX 0      Temp. 0    57    Degrees C   5 sec OK
/N0/RP0 DX 1      Temp. 0    59    Degrees C   5 sec OK
/N0/RP2 Board 0   1.5 VDC 0   1.48 Volts DC    5 sec OK
/N0/RP2 Board 0   3.3 VDC 0   3.37 Volts DC    5 sec OK
/N0/RP2 Board 0   Temp. 0    22    Degrees C   5 sec OK
/N0/RP2 Board 0   Temp. 1    22    Degrees C   5 sec OK
/N0/RP2 SDC 0     Temp. 0    53    Degrees C   5 sec OK
/N0/RP2 AR 0      Temp. 0    43    Degrees C   5 sec OK
/N0/RP2 DX 0      Temp. 0    49    Degrees C   5 sec OK
/N0/RP2 DX 1      Temp. 0    52    Degrees C   5 sec OK
/N0/SB0 Board 0   1.5 VDC 0   1.51 Volts DC    5 sec OK
/N0/SB0 Board 0   3.3 VDC 0   3.29 Volts DC    5 sec OK
/N0/SB0 SDC 0     Temp. 0    46    Degrees C   5 sec OK
/N0/SB0 AR 0      Temp. 0    39    Degrees C   5 sec OK
/N0/SB0 DX 0      Temp. 0    45    Degrees C   5 sec OK
/N0/SB0 DX 1      Temp. 0    49    Degrees C   5 sec OK
/N0/SB0 DX 2      Temp. 0    53    Degrees C   5 sec OK
/N0/SB0 DX 3      Temp. 0    48    Degrees C   5 sec OK
/N0/SB0 SBBC 0    Temp. 0    49    Degrees C   5 sec OK
/N0/SB0 Board 1   Temp. 0    24    Degrees C   5 sec OK
/N0/SB0 Board 1   Temp. 1    24    Degrees C   6 sec OK
/N0/SB0 CPU 0     Temp. 0    47    Degrees C   6 sec OK
/N0/SB0 CPU 0     1.8 VDC 0   1.72 Volts DC    6 sec OK
/N0/SB0 CPU 1     Temp. 0    47    Degrees C   6 sec OK
/N0/SB0 CPU 1     1.8 VDC 1   1.72 Volts DC    6 sec OK
/N0/SB0 SBBC 1    Temp. 0    37    Degrees C   6 sec OK
/N0/SB0 Board 1   Temp. 2    24    Degrees C   6 sec OK
/N0/SB0 Board 1   Temp. 3    24    Degrees C   6 sec OK
/N0/SB0 CPU 2     Temp. 0    49    Degrees C   6 sec OK
/N0/SB0 CPU 2     1.8 VDC 0   1.71 Volts DC    6 sec OK
/N0/SB0 CPU 3     Temp. 0    46    Degrees C   6 sec OK
/N0/SB0 CPU 3     1.8 VDC 1   1.72 Volts DC    7 sec OK
```

TABLE 10-8 Checking Temperature Conditions Using the `showenvironment` Command *(Continued)*

```
/N0/SB2 Board 0   1.5 VDC 0   1.51 Volts DC    6 sec OK
/N0/SB2 Board 0   3.3 VDC 0   3.29 Volts DC    6 sec OK
/N0/SB2 SDC 0     Temp. 0     55    Degrees C   6 sec OK
/N0/SB2 AR 0      Temp. 0     37    Degrees C   6 sec OK
/N0/SB2 DX 0      Temp. 0     47    Degrees C   6 sec OK
/N0/SB2 DX 1      Temp. 0     50    Degrees C   6 sec OK
/N0/SB2 DX 2      Temp. 0     53    Degrees C   6 sec OK
/N0/SB2 DX 3      Temp. 0     47    Degrees C   6 sec OK
/N0/SB2 SBBC 0    Temp. 0     48    Degrees C   6 sec OK
/N0/SB2 Board 1   Temp. 0     23    Degrees C   7 sec OK
/N0/SB2 Board 1   Temp. 1     24    Degrees C   7 sec OK
/N0/SB2 CPU 0     Temp. 0     45    Degrees C   7 sec OK
/N0/SB2 CPU 0     1.8 VDC 0   1.72 Volts DC    7 sec OK
/N0/SB2 CPU 1     Temp. 0     46    Degrees C   7 sec OK
/N0/SB2 CPU 1     1.8 VDC 1   1.73 Volts DC    7 sec OK
/N0/SB2 SBBC 1    Temp. 0     37    Degrees C   7 sec OK
/N0/SB2 Board 1   Temp. 2     24    Degrees C   7 sec OK
/N0/SB2 Board 1   Temp. 3     25    Degrees C   7 sec OK
/N0/SB2 CPU 2     Temp. 0     47    Degrees C   7 sec OK
/N0/SB2 CPU 2     1.8 VDC 0   1.71 Volts DC    7 sec OK
/N0/SB2 CPU 3     Temp. 0     45    Degrees C   7 sec OK
/N0/SB2 CPU 3     1.8 VDC 1   1.71 Volts DC    7 sec OK
/N0/IB6 Board 0   1.5 VDC 0   1.50 Volts DC    7 sec OK
/N0/IB6 Board 0   3.3 VDC 0   3.35 Volts DC    7 sec OK
/N0/IB6 Board 0   5 VDC 0     4.95 Volts DC    7 sec OK
/N0/IB6 Board 0   12 VDC 0   11.95 Volts DC    7 sec OK
/N0/IB6 Board 0   Temp. 0     29    Degrees C   7 sec OK
/N0/IB6 Board 0   Temp. 1     28    Degrees C   7 sec OK
/N0/IB6 Board 0   3.3 VDC 1   3.30 Volts DC    7 sec OK
/N0/IB6 Board 0   3.3 VDC 2   3.28 Volts DC    7 sec OK
/N0/IB6 Board 0   1.8 VDC 0   1.81 Volts DC    7 sec OK
/N0/IB6 Board 0   2.5 VDC 0   2.51 Volts DC    7 sec OK
/N0/IB6 Fan 0     Cooling 0   High             7 sec OK
/N0/IB6 Fan 1     Cooling 0   High             7 sec OK
/N0/IB6 SDC 0     Temp. 0     63    Degrees C   7 sec OK
/N0/IB6 AR 0      Temp. 0     77    Degrees C   7 sec OK
/N0/IB6 DX 0      Temp. 0     69    Degrees C   7 sec OK
/N0/IB6 DX 1      Temp. 0     73    Degrees C   8 sec OK
/N0/IB6 SBBC 0    Temp. 0     51    Degrees C   8 sec OK
/N0/IB6 IOASIC 0  Temp. 0     46    Degrees C   8 sec OK
/N0/IB6 IOASIC 1  Temp. 1     52    Degrees C   8 sec OK
```

Power Supplies

Each power supply unit (PSU) has its own LEDs as follows:

- Power/Active – lit if PSU is supplying main power; blinks if PSU is in Standby mode
- Faulty – lit if PSU has detected a fault condition and has turned off its main output
- Predictive Fail – lit if PSU has detected a pending internal fault but is still providing main output power (degraded PSU fan speed is the only trigger for this condition).

In addition there are two system LEDs labelled SourceA and SourceB. These show the state of the power feeds to the system. There are four physical power feeds and they are split into A and B.

Feed A supplies PS0 and PS1, feed B supplies PS2 and PS3. If either PS0 or PS1 receives input power then the SourceA indicator is lit. If either PS2 or PS3 receives input power then the SourceB indicator is lit. If neither of the supplies receives input power, the indicator is turned off.

These indicators are set on the basis of periodic monitoring at least once every 10 seconds.

Displaying Diagnostic Information

For information on displaying diagnostic information, see the *Sun Hardware Platform Guide*, which is available with your Solaris Operating System release.

Assisting Sun Service Personnel in Determining Causes of Failure

Provide the following information to Sun service personnel so that they can help you determine the causes of your failure:

- A verbatim transcript of all output written to the system console leading up to the failure. Also include any output printed subsequent to user actions. If the transcript does not show certain user actions, in a separate file include comments on what actions prompted particular messages.
- A copy of the system log file from `/var/adm/messages` from the time leading up to the failure.
- The following system controller commands output from the LOM shell:
  - `showsc -v` command
  - `showboards -v` command
  - `showlogs` command
  - `history`
  - `date`
  - `showresetstate`
  - `showenvironment`

CHAPTER 11

Firmware Update Procedures

This chapter explains how to update the system firmware.

The firmware on the Sun Fire entry-level midrange systems system can be updated by two mechanisms:

- The `flashupdate` command from the SC LOM prompt.
- The `lom -G` command in the Solaris Operating System.

The first method requires that the SC 10/100 Ethernet port is connected to a suitable network and is configured so that it can see an external FTP or HTTP server that contains the new firmware images to be downloaded.

This chapter includes the following topics:

Using the `flashupdate` Command

The `flashupdate` command requires that the 10/100 Ethernet port can access an external FTP or HTTP server.

The `flashupdate` command updates the flash PROMs in the SC and the system boards (CPU/Memory boards and I/O Assembly). The source flash image is normally held on an NFS server. In the case of CPU/Memory boards you can update one board with the flash image from another.

The syntax for the `flashupdate` command is:

```
flashupdate [-y|-n] -f url all|systemboards|scapp|rtos|board . . .

flashupdate [-y|-n] -c source_board destination_board . . .

flashupdate [-y|-n] -u
```

where:

-y does not prompt for confirmation.

-n does not execute this command if confirmation is required.

-f specifies a URL as the source of the flash images. This option requires a network connection with the flash image held on an NFS server. Use this option to install new firmware.

url is the URL of the directory containing the flash images and must be of the form:

ftp://[*userid*:*password*@]*hostname*/*path*

or

http://*hostname*/*path*

all causes all boards (CPU/Memory, I/O Assembly and System Controller) to be updated. This action reboots the SC.

systemboards causes all CPU/Memory boards and the I/O Assembly to be updated.

scapp causes the SC application to be updated. This action reboots the SC.

rtos causes the SC RTOS to be updated. This action reboots the SC.

board names a specific board to be updated (sb0, sb2, sb4 or ib6).

-c specifies a board as the source of flash images. Use this option to update replacement CPU/Memory boards.

source_board is a pre-existing CPU/Memory board to be used as the source of the flash image (sb0, sb2 or sb4).

destination_board is the CPU/Memory board to be updated (sb0, sb2 or sb4).

-u automatically updates all CPU/Memory boards with the image from the board that currently has the highest firmware revision. Use this option to update replacement CPU/Memory boards.

-h displays help for this command.

A power cycle is required in order to activate the updated OpenBoot PROM.

Note – flashupdate cannot retrieve flash images from a secure (userid/password) protected HTTP URL. A message of the form flashupdate: failed, URL does not contain required file: *file* will be returned, although the file may exist.

Caution – Do not interrupt the `flashupdate` operation. If the `flashupdate` command is terminated abnormally, the SC will go into single use mode and will only be accessible from the serial port.

Caution – Before performing a flashupdate, check the firmware revisions of all boards using the `showboards -p version` command.

Caution – If the SC application (`scapp`) or RTOS are to be updated, run the `flashupdate` command from a LOM shell running on the serial connection, so that the results can be fully monitored.

Caution – Before updating CPU/Memory boards or the I/O Assembly, ensure that all boards to be updated are powered on by using the `poweron` command.

▼ To Upgrade a Sun Fire V1280 or Netra 1280 System Running Firmware Version 5.13.*x* to 5.17.0 Using the `flashupdate` Command

1. **Upgrade the firmware on the SC:**

```
lom>flashupdate -f url scapp rtos
```

2. **Power on all boards:**

```
lom>poweron all
```

3. **Upgrade the firmware on the system boards:**

```
lom>flashupdate -f url sb0 sb2 sb4 ib6
```

This step has brought `sb0, sb2,sb4,` and `IB6` up to the same firmware level as the system controller.

4. **Shutdown the Solaris OS.**

5. **Power off the system.**
6. **Power on the system.**

▼ To Upgrade a Sun Fire E2900, Sun Fire V1280 or Netra 1280 System Running Firmware Version 5.17.*x* or 5.18.*x* to 5.19.0 Using the `flashupdate` Command

1. **Power on all boards:**

```
lom>poweron all
```

2. **Upgrade the firmware on the SC:**

```
lom>flashupdate -f url all
```

   This step has brought the CPU/Memory boards, `IB6`, and the system controller up to the same firmware level.

3. **Shutdown the Solaris OS.**
4. **Power off the system.**
5. **Power on the system.**

▼ To Downgrade a Sun Fire V1280 or Netra 1280 System Running Firmware Version 5.17.*x* to 5.13.*x* Using the `flashupdate` Command

1. **Downgrade the firmware on the SC:**

```
lom>flashupdate -f url scapp rtos
```

2. **Power on all boards:**

```
lom>poweron all
```

3. **Downgrade the firmware on the system boards:**

```
lom>flashupdate -f url sb0 sb2 sb4 ib6
```

4. **Shutdown the Solaris OS.**
5. **Power off the system.**
6. **Power on the system.**

Note – 5.13.*x* firmware does not support UltraSPARC IV CPU/Memory boards or any features introduced in 5.17.*x*, 5.18.*x*, or 5.19.0 firmware.

▼ To Downgrade a Sun Fire E2900, Sun Fire V1280 or Netra 1280 System Running Firmware Version 5.18.*x* or 5.19.0 to 5.17.*x* Using the `flashupdate` Command

1. **Power on all boards:**

```
lom>poweron all
```

2. **Downgrade the firmware on the SC:**

```
lom>flashupdate -f url all
```

This step has brought the CPU/Memory boards, `IB6`, and the system controller down to the same firmware level.

3. **Shutdown the Solaris OS.**
4. **Power off the system.**
5. **Power on the system.**

Using the `lom -G` Command

There are four image types which may need to be transferred using this method with names of the following form:

- `lw8pci.flash` (contains I/O board Local POST)
- `lw8cpu.flash` (contains CPU/Memory Board Local POST and OBP)
- `sgsc.flash` (contains LOM/SC firmware)
- `sgrtos.flash` (contains LOM/SC Real Time Operating System)

You must place these in a suitable directory, for instance `/var/tmp`, and issue the `lom -G` command with the filename of the file to be downloaded. The firmware knows from header information contained in the file which image type is being upgraded.

These images will be provided in a patch downloadable from `www.sunsolve.sun.com` or from your Sun Service representative.

The patch README file should contain full instructions for installing these new firmware images. It is very important that the instructions are followed exactly otherwise you may render your system unbootable.

Caution – Do not interrupt the `lom -G` operation. If the `lom -G` command is terminated abnormally, the SC goes into single use mode and is accessible only from the serial port.

Caution – Before performing a `lom -G`, check the firmware revisions of all boards using the `showboards -p version` command.

Caution – Run the `lom -G` command from a Solaris console running on the serial connection so that the results can be fully monitored.

Caution – Before updating CPU/Memory boards or the I/O Assembly, ensure that all boards to be updated are powered on by using the `poweron` command.

Examples

Downloading the `lw8pci.flash` image:

CODE EXAMPLE 11-1 Downloading the `lw8pci.flash` Image

```
# lom -G lw8pci.flash
This program will replace LOM firmware with version 5.17.0
Are you sure you want to continue?
Enter 'C' and return to Continue or anything else to Terminate
C
Transferring 346 kB image to the system controller.
This may take several minutes.
.................................................

Validating image...
346 kB IO image transferred.
Programming /N0/IB6/FP0
Comparing image and flash
# Image and flash are different, proceeding with update.
Erasing     ...... Done
Programming ...... Done
Verifying   ...... Done
Fri Dec 12 08:20:42 commando lom: /N0/IB6/FP0 updated with version 5.17.0
12/12/2003.
Dec 12 11:20:41 commando-a lw8: /N0/IB6/FP0 updated with version 5.17.0
12/12/2003.

Firmware update complete.

You must reboot Solaris to load the new firmware.
```

Downloading the `lw8cpu.flash` image:

CODE EXAMPLE 11-2 Downloading the `lw8cpu.flash` Image

```
# lom -G lw8cpu.flash
This program will replace LOM firmware with version 5.17.0
Are you sure you want to continue?
Enter 'C' and return to Continue or anything else to Terminate
C
Transferring 906 kB image to the system controller.
This may take several minutes.
.................................................

Validating image...
# 906 kB CPU image transferred.
Programming /N0/SB0/FP0
Comparing image and flash
```

CODE EXAMPLE 11-2 Downloading the `lw8cpu.flash` Image *(Continued)*

```
Image and flash are different, proceeding with update.
Erasing     ..............  Done
Programming ..............  Done
Verifying   ..............  Done
Fri Dec 12 08:23:43 commando lom: /N0/SB0/FP0 updated with version 5.17.0
12/12/2003.
Dec 12 11:23:42 commando-a lw8: /N0/SB0/FP0 updated with version 5.17.0
12/12/2003.
Programming /N0/SB0/FP1
Comparing image and flash
Image and flash are different, proceeding with update.
Erasing     ..............  Done
Programming ..............  Done
Verifying   ..............  Done
Fri Dec 12 08:24:24 commando lom: /N0/SB0/FP1 updated with version 5.17.0
12/12/2003.
Dec 12 11:24:23 commando-a lw8: /N0/SB0/FP1 updated with version 5.17.0
12/12/2003.
Programming /N0/SB2/FP0
Comparing image and flash
Image and flash are different, proceeding with update.
Erasing     ..............  Done
Programming ..............  Done
Verifying   ..............  Done
Fri Dec 12 08:25:06 commando lom: /N0/SB2/FP0 updated with version 5.17.0
12/12/2003.
Dec 12 11:25:06 commando-a lw8: /N0/SB2/FP0 updated with version 5.17.0
12/12/2003.
Programming /N0/SB2/FP1
Comparing image and flash
Image and flash are different, proceeding with update.
Erasing     ..............  Done
Programming ..............  Done
Verifying   ..............  Done
Fri Dec 12 08:25:48 commando lom: /N0/SB2/FP1 updated with version 5.17.0
12/12/2003.
Dec 12 11:25:48 commando-a lw8: /N0/SB2/FP1 updated with version 5.17.0
12/12/2003.
Programming /N0/SB4/FP0
Comparing image and flash
Image and flash are different, proceeding with update.
Erasing     ..............  Done
Programming ..............  Done
Verifying   ..............  Done
Fri Dec 12 08:26:31 commando lom: /N0/SB4/FP0 updated with version 5.17.0
12/12/2003.
```

CODE EXAMPLE 11-2 Downloading the `lw8cpu.flash` Image *(Continued)*

```
Dec 12 11:26:30 commando-a lw8: /N0/SB4/FP0 updated with version 5.17.0
12/12/2003.
Programming /N0/SB4/FP1
Comparing image and flash
Image and flash are different, proceeding with update.
Erasing     ..............  Done
Programming ..............  Done
Verifying   ..............  Done
Fri Dec 12 08:27:11 commando lom: /N0/SB4/FP1 updated with version 5.17.0
12/12/2003.
Dec 12 11:27:10 commando-a lw8: /N0/SB4/FP1 updated with version 5.17.0
12/12/2003.

Firmware update complete.

You must reboot Solaris to load the new firmware.
```

▼ To Upgrade the Firmware on a Sun Fire E2900, Sun Fire V1280 or Netra 1280 System Using the `lom -G` Command

The same upgrade procedure applies when updating firmware from 5.13.*x*x to 5.17.*x*, 5.18.*x*, or to 5.19.0.

1. **Upgrade the firmware on the SC:**

```
# lom -G sgsc.flash
# lom -G sgrtos.flash
```

Make sure that you upgrade the SC with both packages from the selected release (`sgsc.flash` *and* `sgrtos.flash`) before proceeding to the next step. The packages are a matched pair and require each other.

2. **Escape to `lom>` and reset the SC:**

```
lom>resetsc -y
```

3. **Upgrade the firmware on the system boards:**

```
# lom -G lw8cpu.flash
# lom -G lw8pci.flash
```

4. **Shutdown the Solaris OS.**
5. **Power off the system.**
6. **Power on the system.**

▼ To Downgrade the Firmware on a Sun Fire E2900, Sun Fire V1280 or Netra 1280 System Using the `lom -G` Command

The same downgrade procedure applies when updating firmware from 5.19.0 or 5.18.*x* to 5.17.*x* or to 5.13.*x*.

Note – 5.13.*x* firmware does not support UltraSPARC IV CPU/Memory boards or any features introduced in 5.17.*x*, 5.18.*x*, or 5.19.0 firmware.

---

1. **Downgrade the firmware on the SC:**

```
# lom -G sgsc.flash
# lom -G sgrtos.flash
```

2. **Escape to `lom>` and reset the SC:**

```
lom>resetsc -y
```

3. **Downgrade the firmware on the other boards:**

```
# lom -G lw8cpu.flash
# lom -G lw8pci.flash
```

4. **Shutdown the Solaris OS.**
5. **Power off the system.**

6. **Power on the system.**

CHAPTER 12

CPU/Memory Board Replacement and Dynamic Reconfiguration (DR)

This chapter describes how to dynamically reconfigure the CPU/Memory boards on the Sun Fire entry-level midrange systems system.

This chapter includes the following topics:

Dynamic Reconfiguration

DR software is part of the Solaris Operating System. With the DR software you can dynamically reconfigure system boards and safely remove them or install them into a system while the Solaris Operating System is running and with minimum disruption to user processes running on the system. You can use DR to do the following:

- Minimize the interruption of system applications while installing or removing a board.
- Disable a failing device by removing it before the failure can crash the operating system.
- Display the operational status of boards.
- Initiate system tests of a board while the system continues to run.

Command-Line Interface

The Solaris `cfgadm`(1M) command provides the command-line interface for the administration of DR functionality.

DR Concepts

Quiescence

During the unconfigure operation on a system board with permanent memory (OpenBoot PROM or kernel memory), the operating system is briefly paused, which is known as operating system quiescence. All operating system and device activity on the baseplane must cease during a critical phase of the operation.

Note – Quiescence may take several minutes, depending on workload and system configuration.

Before it can achieve quiescence, the operating system must temporarily suspend all processes, CPUs, and device activities. It might take a few minutes to achieve quiescence depending on system usage and activities currently in progress. If the operating system cannot achieve quiescence, it displays the reasons, which might include the following:

- An execution thread did not suspend
- Real-time processes are running
- A device exists that cannot be paused by the operating system

The conditions that cause processes to fail to suspend are generally temporary. Examine the reasons for the failure. If the operating system encountered a transient condition—a failure to suspend a process—you can try the operation again.

RPC or TCP Time-out or Loss of Connection

Time-outs occur by default after two minutes. Administrators might need to increase this time-out value to avoid time-outs during a DR-induced operating system quiescence, which might take longer than two minutes. Quiescing a system makes the system and related network services unavailable for a period of time that can exceed two minutes. These changes affect both the client and server machines.

Suspend-Safe and Suspend-Unsafe Devices

When DR suspends the operating system, all of the device drivers that are attached to the operating system must also be suspended. If a driver cannot be suspended (or subsequently resumed), the DR operation fails.

A *suspend-safe* device does not access memory or interrupt the system while the operating system is in quiescence. A driver is suspend-safe if it supports operating system quiescence (suspend/resume). A suspend-safe driver also guarantees that when a suspend request is successfully completed, the device that the driver manages will not attempt to access memory, even if the device is open when the suspend request is made.

A *suspend-unsafe* device allows a memory access or a system interruption to occur while the operating system is in quiescence.

Attachment Points

An attachment point is a collective term for a board and its slot. DR can display the status of the slot, the board, and the attachment point. The DR definition of a board also includes the devices connected to it, so the term *occupant* refers to the combination of board and attached devices.

- A slot (also called a receptacle) has the ability to electrically isolate the occupant from the host machine. That is, the software can put a single slot into low-power mode.
- Receptacles can be named according to slot numbers or can be anonymous (for example, a SCSI chain). To obtain a list of all available logical attachment points, use the `-l` option with the `cfgadm`(1M) command.

There are two formats used when referring to attachment points:

- A *physical* attachment point describes the software driver and location of the slot. An example of a physical attachment point name is:

```
/devices/ssm@0,0:N0.SBx
```

where

`N0`	`is node 0 (zero)`
`SB`	`is a system board`
x	`is a slot number. A slot number can be 0, 2 or 4 for a system board`

- A *logical* attachment point is an abbreviated name created by the system to refererence the physical attachment point. Logical attachment points take the following form:

```
N0.SBx
```

- Note that `cfgadm` will also show the I/O assembly `N0.IB6`, but as this is non-redundant, no DR actions are allowed on this attachment point.

DR Operations

There are four main types of DR operation.

TABLE 12-1 Types of DR Operation

Type	Description
Connect	The slot provides power to the board and monitors its temperature.
Configure	The operating system assigns functional roles to a board, and loads device drivers for the board, and brings the devices on that board into use by the Solaris Operating System.
Unconfigure	The system detaches a board logically from the operating system. Environmental monitoring continues, but devices on the board are not available for system use.
Disconnect	The system stops monitoring the board, and power to the slot is turned off.

If a system board is in use, stop its use and disconnect it from the system before you power it off. After a new or upgraded system board is inserted and powered on, connect its attachment point and configure it for use by the operating system. The `cfgadm`(1M) command can connect and configure (or unconfigure and disconnect) in a single command, but if necessary, each operation (connection, configuration, unconfiguration, or disconnection) can be performed separately.

Hot-Plug Hardware

Hot-plug devices have special connectors that supply electrical power to the board or module before the data pins make contact. Boards and devices that have hot-plug connectors can be inserted or removed while the system is running. The devices have control circuits to ensure they have a common reference and power control during the insertion process. The interfaces are not powered on until the board is home and the SC instructs them to.

The CPU/Memory boards used in the Sun Fire entry-level midrange systems system are hot-plug devices.

Conditions and States

A state is the operational status of either a receptacle (slot) or an occupant (board). A condition is the operational status of an attachment point.

Before you attempt to perform any DR operation on a board or component from a system, you must determine state and condition. Use the `cfgadm`(1M) command with the `-la` options to display the type, state, and condition of each component and the state and condition of each board slot in the system. See the section "Component Types" on page 137 for a list of the component types.

Board States and Conditions

This section contains descriptions of the states and conditions of CPU/Memory boards (also known as system slots).

Board Receptacle States

A board can have one of three receptacle states: empty, disconnected, or connected. Whenever you insert a board, the receptacle state changes from empty to disconnected. Whenever you remove a board the receptacle state changes from disconnected to empty.

Caution – Physically removing a board that is in the connected state, or that is powered on and in the disconnected state, crashes the operating system and can result in permanent damage to that system board.

TABLE 12-2 Board Receptacle States

Name	Description
`empty`	A board is not present.
`disconnected`	The board is disconnected from the system bus. A board can be in the disconnected state without being powered off. However, a board must be powered off and in the disconnected state before you remove it from the slot.
`connected`	The board is powered on and connected to the system bus. You can view the components on a board only after it is in the connected state.

Board Occupant States

A board can have one of two occupant states: configured or unconfigured. The occupant state of a disconnected board is always unconfigured.

TABLE 12-3 Board Occupant States

Name	Description
`configured`	At least one component on the board is configured.
`unconfigured`	All of the components on the board are unconfigured.

Board Conditions

A board can be in one of four conditions: unknown, ok, failed, or unusable.

TABLE 12-4 Board Conditions

Name	Description
`unknown`	The board has not been tested.
`ok`	The board is operational.
`failed`	The board failed testing.
`unusable`	The board slot is unusable.

Component States and Conditions

This section contains descriptions of the states and conditions for components.

Component Receptacle States

A component cannot be individually connected or disconnected. Thus, components can have only one state: connected.

Component Occupant States

A component can have one of two occupant states: configured or unconfigured.

TABLE 12-5 Component Occupant States

Name	Description
`configured`	Component is available for use by the Solaris Operating System.
`unconfigured`	Component is not available for use by the Solaris Operating System.

Component Conditions

A component can have one of three conditions: unknown, ok, failed.

TABLE 12-6 Component Conditions

Name	Description
`unknown`	Component has not been tested.
`ok`	Component is operational.
`failed`	Component failed testing.

Component Types

You can use DR to configure or to unconfigure several types of component.

TABLE 12-7 Component Types

Name	Description
`cpu`	Individual CPU
`memory`	All the memory on the board

Nonpermanent and Permanent Memory

Before you can delete a board, the environment must vacate the memory on that board. Vacating a board means flushing its nonpermanent memory to swap space and copying its permanent (that is, kernel and OpenBoot PROM memory) to another memory board. To relocate permanent memory, the operating system on a system must be temporarily suspended, or quiesced. The length of the suspension depends on the system configuration and the running workloads. Detaching a board with permanent memory is the only time when the operating system is suspended; therefore, you should know where permanent memory resides so that you can avoid significantly impacting the operation of the system. You can display the permanent memory by using the `cfgadm`(1M) command with the `-v` option. When permanent memory is on the board, the operating system must find another memory component of adequate size to receive the permanent memory. If that is not possible the DR operation will fail.

Limitations

Memory Interleaving

System boards cannot be dynamically reconfigured if system memory is interleaved across multiple CPU/Memory boards.

Reconfiguring Permanent Memory

When a CPU/Memory board containing nonrelocatable (permanent) memory is dynamically reconfigured out of the system, a short pause in all domain activity is required which may delay application response. Typically, this condition applies to one CPU/Memory board in the system. The memory on the board is identified by a nonzero permanent memory size in the status display produced by the `cfgadm -av` command.

DR supports reconfiguration of permanent memory from one system board to another only if one of the following conditions is met:

- The target system board has the same amount of memory as the source system board.
- The target system board has more memory than the source system board. In this case, the additional memory is added to the pool of available memory.

Command-Line Interface

The following procedures are discussed in this section:

Note – There is no need to enable dynamic reconfiguration explicitly. DR is enabled by default.

The `cfgadm` Command

The `cfgadm`(1M) command provides configuration administration operations on dynamically reconfigurable hardware resources. TABLE 12-8 lists the DR board states.

TABLE 12-8 DR Board States from the System Controller (SC)

Board States	Description
`Available`	The slot is not assigned.
`Assigned`	The board is assigned, but the hardware has not been configured to use it. The board may be reassigned by the chassis port or released.
`Active`	The board is being actively used. You cannot reassign an active board.

▼ To Display Basic Board Status

The `cfgadm` program displays information about boards and slots. See the `cfgadm(1)` man page for options to this command.

Many operations require that you specify the system board names.

- **To obtain these system names, type:**

```
# cfgadm
```

When used without options, `cfgadm` displays information about all known attachment points, including board slots and SCSI buses. The following display shows a typical output.

CODE EXAMPLE 12-1 Output of the Basic `cfgadm` Command

```
# cfgadm
Ap_Id  Type Receptacle Occupant Condition
N0.IB6 PCI_I/O_Boa connected configured ok
N0.SB0 CPU_Board connected configured unknown
N0.SB4 unknown emptyunconfigured unknown
c0     scsi-bus connected configured unknown
c1     scsi-bus connected unconfigured unknown
c2     scsi-bus connected unconfigured unknown
c3     scsi-bus connected configured unknown
```

▼ To Display Detailed Board Status

- **Use the command `cfgadm -av` for a more detailed status report**

The `-a` option lists attachment points and the `-v` option turns on expanded (verbose) descriptions.

CODE EXAMPLE 12-2 is a *partial* display produced by the `cfgadm -av` command. The output appears complicated because the lines wrap around in this display. (This status report is for the same system used in CODE EXAMPLE 12-1.) FIGURE 12-1 provides details of each display item.

CODE EXAMPLE 12-2 Output of the `cfgadm -av` Command

```
# cfgadm -av
Ap_Id Receptacle Occupant Condition Information
When Type Busy Phys_Id
N0.IB6 connected configured ok powered-on, assigned
Apr 3 18:04 PCI_I/O_Boa n /devices/ssm@0,0:N0.IB6
N0.IB6::pci0 connected configured ok device
/ssm@0,0/pci@19,70000
Apr 3 18:04 io n /devices/ssm@0,0:N0.IB6::pci0
N0.IB6::pci1 connected configured ok device
/ssm@0,0/pci@19,600000
Apr 3 18:04 io n /devices /ssm@0,0:N0.IB6::pci1
N0.IB6::pci2 connected configured ok device
```

CODE EXAMPLE 12-2 Output of the `cfgadm -av` Command *(Continued)*

```
/ssm@0,0/pci@18,700000
Apr 3 18:04 io n /devices/ssm@0,0:N0.IB6::pci2
N0.IB6::pci3 connected configured ok device
/ssm@0,0/pci@18,600000
Apr 3 18:04 io n /devices/ssm@0,0:N0.IB6::pci3
N0.SB0 connected configured unknown powered-on, assigned
Apr 3 18:04 CPU_Board n /devices/ssm@0,0:N0.SB0
N0.SB0::cpu0 connected configured ok cpuid 0, speed 750 MHz,
ecache 8 MBytes
Apr 3 18:04 cpu n /devices/ssm@0,0:N0.SB0::cpu0
N0.SB0::cpu1 connected configured ok cpuid 1, speed 750 MHz,
ecache 8 MBytes
Apr 3 18:04 cpu n /devices/ssm@0,0:N0.SB0::cpu1
N0.SB0::cpu2 connected configured ok cpuid 2, speed 750 MHz,
ecache 8 MBytes
Apr 3 18:04 cpu n /devices/ssm@0,0:N0.SB0::cpu2
```

FIGURE 12-1 shows details of the display in CODE EXAMPLE 12-2:

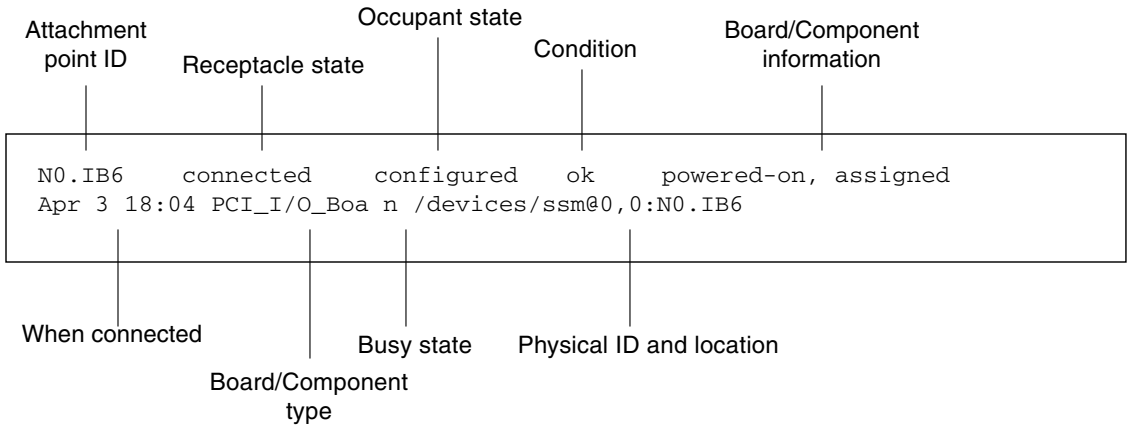

FIGURE 12-1 Details of the Display for `cfgadm -av`

Command Options

The options to the `cfgadm -c` command are listed in TABLE 12-9.

TABLE 12-9 `cfgadm -c` Command Options

cfgadm -c Option	Function
`connect`	The slot provides power to the board and begins monitoring the board. The slot is assigned if it was not previously assigned.
`disconnect`	The system stops monitoring the board, and power to the slot is turned off.
`configure`	The operating system assigns functional roles to a board and loads device drivers for the board and for the devices attached to the board.
`unconfigure`	The system detaches a board logically from the operating system and takes the associated device drivers offline. Environmental monitoring continues, but any devices on the board are not available for system use.

The options provided by the `cfgadm -x` command are listed in TABLE 12-10.

TABLE 12-10 `cfgadm -x` Command Options

cfgadm -x Option	Function
`poweron`	Powers on a CPU/Memory board.
`poweroff`	Powers off a CPU/Memory board.

The `cfgadm_sbd` man page provides additional information on the `cfgadm -c` and `cfgadm -x` options. The `sbd` library provides the functionality for hot-plugging system boards of the class `sbd`, through the `cfgadm` framework.

Testing Boards and Assemblies

▼ To Test a CPU/Memory Board

Before you can test a CPU/Memory board, it must first be powered on and disconnected. If these conditions are not met, the board test fails.

1. **You can type the Solaris `cfgadm` command to test CPU/memory boards (as superuser):**

```
# cfgadm -t ap-id
```

To change the level of diagnostics that `cfgadm` runs, supply a diagnostic level for the `cfgadm` command as follows:

```
# cfgadm -o platform=diag=level -t ap-id
```

where

level is a diagnostic level

ap-id is one of the following: `N0.SB0`, `N0.SB2` or `N0.SB4`.

If you do not supply *level*, the diagnostic level `init` is used. The diagnostic levels are:

TABLE 12-11 Diagnostic Levels

Diagnostic Level	Description
`init`	Only system board initialization code is run. No testing is done. This is a very fast pass through POST.
`quick`	All system board components are tested with few tests and test patterns.
`min`	Core functionalities of all system board components are tested. This testing performs a quick sanity check of the devices under test.
`default`	All system board components are tested with all tests and test patterns, except for memory and ecache modules. Note that `max` and `default` are the same definition (and that `default` is not the default value).
`max`	All system board components are tested with all tests and test patterns, except for memory and ecache modules. Note that `max` and `default` are the same definition.
`mem1`	Runs all tests at the `default` level, plus more exhaustive DRAM and SRAM test algorithms. For memory and ecache modules, all locations are tested with multiple patterns. More extensive, time-consuming algorithms are not run at this level.
`mem2`	The same as `mem1`, with the addition of a DRAM test that does explicit compare operations of the DRAM data.

Installing or Replacing CPU/Memory Boards

Caution – Physical board replacement should only be carried out by qualified service personnel.

▼ To Install a New Board

Caution – For complete information about physically removing and replacing CPU/Memory boards, see the *Sun Fire E2900 System Service Manual* or *Sun Fire V1280/Netra 1280 Service Manual*. Failure to follow the stated procedures can result in damage to system boards and other components.

Note – When replacing boards, you sometimes need filler panels.

If you are unfamiliar with how to insert a board into the system, read the *Sun Fire E2900 System Service Manual* or *Sun Fire V1280/Netra 1280 Service Manual* before you begin this procedure.

1. **Make sure you are properly grounded with a wrist strap.**
2. **After locating an empty slot, remove the system board filler panel from the slot.**
3. **Insert the board into the slot within one minute to prevent the system overheating.**

   See the *Sun Fire E2900 System Service Manual* or *Sun Fire V1280/Netra 1280 Service Manual* for complete step-by-step board insertion procedures.
4. **Power on, test, and configure the board using the `cfgadm -c configure` command:**

```
# cfgadm -c configure ap-id
```

where *ap-id* is one of the following: `N0.SB0`, `N0.SB2` or `N0.SB4`.

▼ To Hot-Swap a CPU/Memory Board

Caution – For complete information about physically removing and replacing boards, see the *Sun Fire E2900 System Service Manual* or *Sun Fire V1280/Netra 1280 Service Manual*. Failure to follow the stated procedures can result in damage to system boards and other components.

1. **Make sure you are properly grounded with a wrist strap.**
2. **Power off the board with the `cfgadm` command.**

```
# cfgadm -c disconnect ap-id
```

where *ap-id* is one of the following: `N0.SB0`, `N0.SB2` or `N0.SB4`.

This command removes the resources from the Solaris Operating System and the OpenBoot PROM, and powers off the board.

3. **Verify the state of the Power and Hot-plug OK LEDs.**

The green Power LED will flash briefly as the CPU/Memory board is cooling down. In order to safely remove the board from the systems the green Power LED must be off and the amber Hotplug OK LED must be on.

4. **Complete the hardware removal and installation of the board.**

For more information see the *Sun Fire E2900 System Service Manual* or *Sun Fire V1280/Netra 1280 Service Manual*.

5. **After removing and installing board, bring the board back to the Solaris Operating System with the dynamic reconfiguration `cfgadm` command.**

```
# cfgadm -c configure ap-id
```

where *ap-id* is one of the following: `N0.SB0`, `N0.SB2` or `N0.SB4`.

This command powers the board on, tests it, attaches the board, and brings all of its resources back to the Solaris Operating System.

6. **Verify that the green Power LED is lit.**

▼ To Remove a CPU/Memory Board From the System

Note – Before you begin this procedure, make sure you have ready a system board filler panel to replace the system board you are going to remove. A system board filler panel is a metal board with slots that allow cooling air to circulate.

---

1. **Detach and power off the board from the system by using the `cfgadm -c disconnect` command.**

```
# cfgadm -c disconnect ap-id
```

where *ap-id* is one of the following: `N0.SB0`, `N0.SB2` or `N0.SB4`.

Caution – For complete information about physically removing and replacing boards, see the *Sun Fire E2900 System Service Manual* or *Sun Fire V1280/Netra 1280 Service Manual*. Failure to follow the stated procedures can result in damage to system boards and other components.

---

2. **Remove the board from the system.**

   Refer to the *Sun Fire E2900 System Service Manual* or *Sun Fire V1280/Netra 1280 Service Manual* for complete step-by-step board removal procedures.

3. **Insert a system board filler panel into the slot within one minute of removing the board to prevent system overheating.**

▼ To Disconnect a CPU/Memory Board Temporarily

You can use DR to power off the board and leave it in place. For example, you might want to do this if the board fails and a replacement board or a system board filler panel is not available.

- **Detach and power off the board using the `cfgadm -c disconnect` command.**

```
# cfgadm -c disconnect ap-id
```

where *ap-id* is one of the following: `N0.SB0`, `N0.SB2` or `N0.SB4`.

Troubleshooting

This section discusses common types of failure:

- Unconfigure Operation Failure
- Configure Operation Failure

The following are examples of `cfgadm` diagnostic messages. (Syntax error messages are not included here.)

```
cfgadm: hardware component is busy, try again
cfgadm: operation: Data error: error_text
cfgadm: operation: Hardware specific failure: error_text
cfgadm: operation: Insufficient privileges
cfgadm: operation: Operation requires a service interruption
cfgadm: System is busy, try again
WARNING: Processor number number failed to offline.
```

See the following man pages for additional error message detail: `cfgadm`(1M), `cfgadm_sbd`(1M), and `config_admin`(3X).

CPU/Memory Board Unconfiguration Failures

An unconfigure operation for a CPU/Memory board can fail if the system is not in a correct state before you begin the operation.

- Memory on a board is interleaved across boards before an attempt to unconfigure the board.
- A process is bound to a CPU before an attempt to unconfigure the CPU.
- Memory remains configured on a system board before you attempt a CPU unconfigure operation on that board.
- The memory on the board is configured (in use). See "Unable to Unconfigure Memory on a Board With Permanent Memory" on page 149.
- CPUs on the board cannot be taken off line. See "Unable to Unconfigure a CPU" on page 151.

Cannot Unconfigure a Board Whose Memory Is Interleaved Across Boards

If you try to unconfigure a system board whose memory is interleaved across system boards, the system displays an error message such as:

```
cfgadm: Hardware specific failure: unconfigure N0.SB2::memory: Memory is
interleaved across boards: /ssm@0,0/memory-controller@b,400000
```

Cannot Unconfigure a CPU to Which a Process is Bound

If you try to unconfigure a CPU to which a process is bound, the system displays an error message such as the following:

```
cfgadm: Hardware specific failure: unconfigure N0.SB2::cpu3: Failed to off-line:
/ssm@0,0/SUNW,UltraSPARC-III
```

- **Unbind the process from the CPU and retry the unconfigure operation.**

Cannot Unconfigure a CPU Before All Memory is Unconfigured

All memory on a system board must be unconfigured before you try to unconfigure a CPU. If you try to unconfigure a CPU before all memory on the board is unconfigured, the system displays an error message such as:

```
cfgadm: Hardware specific failure: unconfigure N0.SB2::cpu0: Can't unconfig cpu
if mem online: /ssm@0,0/memory-controller
```

- **Unconfigure all memory on the board and then unconfigure the CPU.**

Unable to Unconfigure Memory on a Board With Permanent Memory

To unconfigure the memory on a board that has permanent memory, move the permanent memory pages to another board that has enough available memory to hold them. Such an additional board must be available before the unconfigure operation begins.

Memory Cannot Be Reconfigured

If the unconfigure operation fails with a message such as the following, the memory on the board could not be unconfigured:

```
cfgadm: Hardware specific failure: unconfigure N0.SB0: No available memory
target: /ssm@0,0/memory-controller@3,400000
```

Add to another board enough memory to hold the permanent memory pages, and then retry the unconfigure operation.

To confirm that a memory page cannot be moved, use the verbose option with the `cfgadm` command and look for the word `permanent` in the listing:

```
# cfgadm -av -s "select=type(memory)"
```

Not Enough Available Memory

If the unconfigure fails with one of the messages below, there will not be enough available memory in the system if the board is removed:

```
cfgadm: Hardware specific failure: unconfigure N0.SB0: Insufficient memory
```

- **Reduce the memory load on the system and try again. If practical, install more memory in another board slot.**

Memory Demand Increased

If the unconfigure fails with the following messages, the memory demand has increased while the unconfigure operation was proceeding:

```
cfgadm: Hardware specific failure: unconfigure N0.SB0: Memory operation failed
```

```
cfgadm: Hardware specific failure: unconfigure N0.SB0: Memory operation refused
```

- **Reduce the memory load on the system and try again.**

Unable to Unconfigure a CPU

CPU unconfiguration is part of the unconfiguration operation for a CPU/Memory board. If the operation fails to take the CPU offline, the following message is logged to the console:

```
WARNING: Processor number failed to offline.
```

This failure occurs if:

- The CPU has processes bound to it.
- The CPU is the last one in a CPU set.
- The CPU is the last online CPU in the system.

Unable to Disconnect a Board

It is possible to unconfigure a board and then discover that it cannot be disconnected. The `cfgadm` status display lists the board as not detachable. This problem occurs when the board is supplying an essential hardware service that cannot be relocated to an alternate board.

CPU/Memory Board Configuration Failure

Cannot Configure Either CPU0 or CPU1 While the Other Is Configured

Before you try to configure either CPU0 or CPU1, make sure that the other CPU is unconfigured. When both CPU0 and CPU1 are unconfigured, it is possible to configure both of them.

CPUs on a Board Must Be Configured Before Memory

Before configuring memory, all CPUs on the system board must be configured. If you try to configure memory while one or more CPUs are unconfigured, the system displays an error message such as:

```
cfgadm: Hardware specific failure: configure N0.SB2::memory: Can't
config memory if not all cpus are online: /ssm@0,0/memorycontroller
```

APPENDIX **A**

Understanding the Watchdog Timer Application Mode

This appendix gives information on the watchdog timer application mode on the Netra 1280 server.

The application mode allows you to:

- Configure the watchdog timer – Your applications running on the host can configure and use the watchdog timer, enabling you to detect fatal problems from applications and to recover automatically.
- Program Alarm 3 – This enables you to generate this alarm in case of critical problems in your applications.

This appendix provides the following sections to help you understand how to configure and use the watchdog timer and program Alarm3:

Note – Once the application watchdog timer is in use, it is necessary to reboot the Solaris operating system in order to return to the default (non-programmable) watchdog timer and default LED behavior (no Alarm3).

Understanding the Watchdog Timer Application Mode

The watchdog mechanism detects a system hang, or an application hang or crash, should they occur. The watchdog is a timer that is continually reset by a user application as long as the operating system and user application are running.

When the application is rearming the application watchdog, an expiration can be caused by:

- Crash of the rearming application
- Hang or crash of the rearming thread in the application
- System hang

When the system watchdog is running, a system hang, or more specifically, the hang of the clock interrupt handler causes an expiration.

The system watchdog mode is the default. If the application watchdog is not initialized, then the system watchdog mode is used.

The `setupsc` command, an existing command on the SC Lights Out Management can be used to configure the recovery for the system watchdog *only*:

```
lom> setupsc
```

The system controller configuration should be as follows:

```
SC POST diag Level [off]:
Host Watchdog [enabled]:
Rocker Switch [enabled]:
Secure Mode [off]:

PROC RTUs installed: 0
PROC Headroom quantity (0 to disable, 4 MAX) [0]:
```

The recovery configuration for the application watchdog is set using Input/Output Control codes (IOCTLs) that are issued to the `ntwdt` driver.

Using the ntwdt Driver

To use the new application watchdog feature, you must install the ntwdt driver. To enable and control the watchdog's application mode, you must program the watchdog system using the LOMIOCDOGxxx IOCTLs, described in the section "Understanding the User API".

If the ntwdt driver, as opposed to the system controller, initiates a reset of the Solaris OS on application watchdog expiration, the value of the following property in the ntwdt driver's configuration file (ntwdt.conf) is used:

```
ntwdt-boottimeout="600";
```

In case of a panic, or an expiration of the application watchdog, the ntwdt driver reprograms the watchdog time-out to the value specified in the property.

Assign a value representing a duration that is longer than the time it takes to reboot and perform a crash dump. If the specified value is not large enough, the SC resets the host if reset is enabled. Note that this reset by the SC occurs only once.

Understanding the User API

The ntwdt driver provides an application programming interface by using IOCTLs. You must open the /dev/ntwdt device node before issuing the watchdog ioctls.

Note – Only a single instance of open() is allowed on /dev/ntwdt; more than one instance of open() will generate the following error message: EAGAIN – The driver is busy, try again.

You can use the following IOCTLs with the watchdog timer:

- LOMIOCDOGTIME
- LOMIOCDOGCTL
- LOMIOCDOGPAT
- LOMIOCDOGSTATE
- LOMIOCALCTL
- LOMIOCALSTATE

Setting the Time-out Period

The LOMIOCDOGTIME IOCTL sets the timeout period of the watchdog. This IOCTL programs the watchdog hardware with the time specified in this IOCTL. You must set the time-out period (LOMIOCDOGTIME) before attempting to enable the watchdog timer (LOMIOCDOGCTL).

The argument is a pointer to an unsigned integer. This integer holds the new timeout period for the watchdog in multiples of 1 second. You can specify any timeout period in the range of 1 second to 180 minutes.

If the watchdog function is enabled, the time-out period is immediately reset so that the new value can take effect. An error (EINVAL) is displayed if the timeout period is less than 1 second or longer than 180 minutes.

Note – The LOMIOCDOGTIME is not intended for general purpose use. Setting the watchdog time-out to too low a value may cause the system to receive a hardware reset if the watchdog and reset functions are enabled. If the time-out is set too low, the user application must be run with a higher priority (for example, as a real time thread) and must be rearmed more often to avoid an unintentional expiration.

Enabling or Disabling the Watchdog

The LOMIOCDOGCTL IOCTL enables or disables the watchdog, and it enables or disables the reset capability. (See "Finding and Defining Data Structures" on page 157 for the correct values for the watchdog timer.)

The argument is a pointer to the lom_dogctl_t structure (described in greater detail in "Finding and Defining Data Structures" on page 157).

Use the reset_enable member to enable or disable the system reset function. Use the dog_enable member to enable or disable the watchdog function. An error (EINVAL) is displayed if the watchdog is disabled but reset is enabled.

Note – If LOMIOCDOGTIME has not been issued to set up the time-out period prior to this IOCTL, the watchdog is NOT enabled in the hardware.

Rearming, or Patting, the Watchdog

The LOMIOCDOGPAT IOCTL rearms, or pats, the watchdog so that the watchdog starts ticking from the beginning; that is, to the value specified by LOMIOCDOGTIME. This IOCTL requires no arguments. If the watchdog is enabled, this IOCTL must be used at regular intervals that are less than the watchdog timeout, or the watchdog expires.

Getting the State of the Watchdog Timer

The LOMIOCDOGSTATE IOCTL gets the state of the watchdog and reset functions and retrieves the current time-out period for the watchdog. If LOMIOCDOGSTATE was never issued to set up the time-out period prior to this IOCTL, the watchdog is not enabled in the hardware.

The argument is a pointer to the lom_dogstate_t structure (described in greater detail in "Finding and Defining Data Structures" on page 157). The structure members are used to hold the current states of the watchdog reset circuitry and current watchdog time-out period. Note that this is not the time remaining before the watchdog is triggered.

The LOMIOCDOGSTATE IOCTL requires only that open() be successfully called. This IOCTL can be run any number of times after open() is called, and it does not require any other DOG IOCTLs to have been executed.

Finding and Defining Data Structures

All data structures and ioctls are defined in lom_io.h, which is available in the SUNWlomh package.

The data structures for the watchdog timer are shown here:

1. The watchdog/reset state data structure is as follows:

CODE EXAMPLE A-1 Watchdog/Reset State Data Structure

```
typedef struct {
        int reset_enable; /* reset enabled if non-zero */
        int dog_enable; /* watchdog enabled if non-zero */
        uint_t dog_timeout; /* Current watchdog timeout */
} lom_dogstate_t;
```

2. The watchdog/reset control data structure is as follows:

CODE EXAMPLE A-2 Watchdog/Reset Control Data Structure

```
typedef struct {
        int reset_enable; /* reset enabled if non-zero */
        int dog_enable; /* watchdog enabled if non-zero */
} lom_dogctl_t;
```

Using the Sample Watchdog Program

Following is a sample program for the watchdog timer.

CODE EXAMPLE A-3 Example Watchdog Program

```
#include "sys/types.h"
#include "lom_io.h"
#include "fnctl.h"
#include "unistd.h"
#include "sys/stat.h"

int
main()
{
      uint_t timeout = 30; /* 30 seconds */
      lom_dogctl_t dogctl;
      int fd;

      dogctl.reset_enable = 1;
      dogctl.dog_enable = 1;

      fd = open("/dev/ntwdt", O_EXCL);
```

CODE EXAMPLE A-3 Example Watchdog Program *(Continued)*

```
        /* Set timeout */
        ioctl(fd, LOMIOCDOGTIME, (void *)&timeout);

        /* Enable watchdog */
        ioctl(fd, LOMIOCDOGCTL, (void *)&dogctl);

        /* Keep patting */
        While (1) {
              ioctl(fd, LOMIOCDOGPAT, NULL);
              sleep (5);
        }

        return (0);
}
```

Programming Alarm 3

Alarm 3 is available to Solaris Operating System users irrespective of the watchdog mode. Alarm 3 or system alarm ON and OFF have been redefined (see the table below.)

Set the value of Alarm 3 using the `LOMIOCALCTL` IOCTL. You can program Alarm 3 like you set and clear Alarm 1 and Alarm 2.

The following table presents the behavior of Alarm 3:

TABLE A-1 Alarm 3 Behavior

	Alarm 3	Relay	System LED (Green)
Poweroff	ON	COM -> NC	OFF
Poweron/LOM up	ON	COM -> NC	OFF
Solaris running	OFF	COM -> NO	ON
Solaris not running	ON	COM -> NC	OFF
Host WDT expires	ON	COM -> NC	OFF
User sets to ON	ON	COM -> NC	OFF
User sets to OFF	OFF	COM -> NO	ON

To summarize the data in the table:

Alarm3 ON = Relay(COM->NC), System LED OFF
Alarm3 OFF = Relay(COM->NO), System LED ON

When programmed, you can check Alarm3 or the system alarm with the `showalarm` command and the argument `system`.

For example:

```
sc> showalarm system
system alarm is on
```

The data structure used with the `LOMIOCALCTL` and `LOMIOCALSTATE` IOCTLs is as follows:

CODE EXAMPLE A-4 `LOMIOCALCTL` and `LOMIOCALSTATE` IOCTL Data Structure

```
#include <lom_io.h>

#define ALARM_NUM_1 1
#define ALARM_NUM_2 2
#define ALARM_NUM_3 3

#define ALARM_OFF 0
#define ALARM_ON 1

typedef struct {
      int alarm_no;
      int alarm_state;
} lom_aldata_t;
```

Understanding Error Messages

Following are the error messages that might be displayed and what they mean.

EAGAIN

This error message is displayed if you attempt to open more than one instance of `open()` on `/dev/ntwdt`.

EFAULT

This error message is displayed if a bad user-space address is specified.

EINVAL

This error message is displayed if a non-existant control command was requested or invalid parameters were supplied.

EINTR

This error message is displayed if a thread awaiting a component state change is interrupted.

ENXIO

This error message is displayed if the driver is not installed in the system.

Knowing Unsupported Features and Limitations

1. In the case of the watchdog timer expiration detected by the SC, the recovery is attempted only once; there are no further attempts of recovery if the first attempt fails to recover the domain.
2. If the application watchdog is enabled and you break into the OpenBoot™ PROM (OBP) by issuing the `break` command from the system controller's `lom` prompt, the SC automatically disables the watchdog timer.

Note – The SC displays a console message as a reminder that the watchdog, from the SC's perspective, is disabled.

However, when you reenter the Solaris OS, the watchdog timer is still ENABLED from the Solaris Operating System's perspective. To have both the SC and the Solaris OS view the same watchdog state, you must use the watchdog application to either enable or disable the watchdog.

3. If you perform a dynamic reconfiguration (DR) operation in which a system board containing kernel (permanent) memory is deleted, then you must disable the watchdog timer's application mode before the DR operation and enable it after the DR operation. This is required because Solaris software quiesces all system IO and disables all interrupts during a memory-delete of permanent memory. As a result, system controller firmware and Solaris software can not communicate during the DR operation. Note that this limitation affects neither the dynamic addition of memory nor the deletion of a board not containing permanent memory. In those cases, the watchdog timer's application mode can run concurrently with the DR implementation.

   You can execute the following command to locate the system boards that contain kernel (permanent) memory:

   ```
   sh> cfgadm -lav | grep -i permanent
   ```

4. If the Solaris Operating System hangs under the following conditions, the system controller firmware cannot detect the Solaris software hang:

   - Watchdog timer's application mode is set
   - Watchdog timer is not enabled
   - No rearming is done by the user

5. The watchdog timer provides partial boot monitoring. You can use the application watchdog to monitor a domain reboot.

   However, domain booting is not monitored for:

   - Bootup after a cold powerup
   - Recovery of a hung or failed domain

   In the latter cases, a boot failure is not detected and no recovery attempts are made.

6. The watchdog timer's application mode provides no monitoring for application startup. In application mode, if the application fails to start up, the failure is not detected and no recovery is provided.

Glossary

ap-id Attachment point identifier; an `ap-id` specifies the type and location of the attachment point in the system and is unambiguous. There are two types of identifier: physical and logical. A physical identifier contains a fully specified path name, while a logical identifier contains a shorthand notation.

attachment point A collective term for a board and its card cage slot. A *physical* attachment point describes the software driver and location of the card cage slot. A *logical* attachment point is an abbreviated name created by the system to reference the physical attachment point.

`cfgadm` **command** `cfgadm` is the primary command for dynamic reconfiguration on the Sun Fire Entry-Level Midrange Systems system. For information about the command and its options, see the `cfgadm`(1M), `cfgadm_sbd`(1M), and `cfgadm_pci`(1M) man pages. For any late-breaking news about this and related commands, see the Solaris 8 section at the DR web site. See Chapter 12.

Capacity on Demand (COD) Capacity on Demand (COD) is an option that provides additional processing resources (CPUs) when you need them. These additional CPUs are provided on COD CPU/Memory boards that are installed in Sun Fire entry-level midrange systems. You can access the COD CPUs after you purchase the COD right-to-use (RTU) licenses for them.

condition The operational status of an attachment point.

configuration (board) The operating system assigns functional roles to a board and loads device drivers for the board and for devices attached to the board.

configuration (system) The collection of attached devices known to the system. The system cannot use a physical device until the configuration is updated. The operating system assigns functional roles to a board and loads device drivers for the board and for devices attached to the board.

connection A board is present in a slot and is electrically connected. The temperature of the slot is monitored by the system.

detachability The device driver supports `DDI_DETACH` and the device (such as an I/O board or a SCSI chain) is physically arranged so that it can be detached.

disconnection The system stops monitoring the board and power to the slot is turned off. A board in this state can be unplugged.

DR See Dynamic Reconfiguration.

Dynamic Reconfiguration Dynamic Reconfiguration (DR) software allows the administrator to (1) view a system configuration; (2) suspend or restart operations involving a port, storage device, or board; and (3) reconfigure the system (detach or attach hot-swappable devices such as disk drives or interface boards) without the need to power down the system. When DR is used with IPMP or Solstice DiskSuite software (and redundant hardware), the server can continue to communicate with disk drives and networks without interruption while a service provider replaces an existing device or installs a new device. DR supports replacement of a CPU/Memory board, provided the memory on the board is not interleaved with memory on other boards in the system.

hot-plug Hot-plug boards and modules have special connectors that supply electrical power to the board or module before the data pins make contact. Boards and devices that do not have hot-plug connectors cannot be inserted or removed while the system is running.

hot-swap A hot-swap device has special DC power connectors and logic circuitry that allow the device to be inserted without the necessity of turning off the system.

instant access CPUs Unlicensed COD CPUs on COD CPU/Memory boards installed in Sun Fire entry-level midrange systems. You can access up to a maximum of four COD CPUs for immediate use while you are purchasing the COD right-to-use (RTU) licenses for the COD CPUs. Also referred to as *headroom*.

IP Multipathing (IPMP) Internet Protocol multipathing. Enables continuous application availability by load balancing failures when multiple network interface cards are attached to a system. If a failure occurs in a network adapter, and if an alternate adapter is connected to the same IP link, the system switches all the network accesses from the failed adapter to the alternate adapter. When multiple network adapters are connected to the same IP link, any increases in network traffic are spread across multiple network adapters, which improves network throughput.

logical DR A DR operation in which hardware is not physically added or removed. An example is the deactivation of a failed board that is then left in the slot (to avoid changing the flow of cooling air) until a replacement is available.

occupant Hardware resource such as a system board or a disk drive that occupies a DR receptacle or slot.

platform A specific Sun Fire system model, such as the Sun Fire entry-level midrange systems.

physical DR A DR operation that involves the physical addition or removal of a board. See also "Logical DR."

quiescence A brief pause in the operating system to allow an unconfigure and disconnect operation on a system board with non-pageable OpenBoot PROM (OBP) or kernel memory. All operating system and device activity on the backplane must cease for a few seconds during a critical phase of the operation.

port A board connector.

receptacle A receiver such as a board slot or SCSI chain.

RTU license Right-to-use license.

SNMP Simple Network Management Protocol. SNMP is any system listening to SNMP events.

state The operational status of either a receptacle (slot) or an occupant (board).

suspendability To be suitable for DR, a device driver must have the ability to stop user threads, execute the `DDI_SUSPEND` call, stop the clock, and stop the CPUs.

suspend-safe A suspend-safe device is one that does not access memory or interrupt the system while the operating system is in quiescence. A driver is considered suspend-safe if it supports operating system quiescence (suspend/resume). It also guarantees that when a suspend request is successfully completed, the device that the driver manages will not attempt to access memory, even if the device is open when the suspend request is made.

suspend-unsafe A suspend-unsafe device is one that allows a memory access or a system interruption while the operating system is in quiescence.

SC software The main application that performs all of the SC hardware management functions.

unconfiguration The system detaches a board logically from the operating system and takes the associated device drivers off-line. Environmental monitoring continues, but any devices on the board are not available for system use.

Index

N

O

P

Q

R

S

T

U

V