Sun Fire™ 6800/4810/4800/3800 System Controller Command Reference Manual

Send comments about this document to: docfeedback@sun.com

Please
Recycle

Adobe PostScript

Contents

Tables

Code Samples

Preface

This book describes the Sun Fire 6800/4810/4800/3800 system controller command line interface. The system controller is responsible for controlling system functions, environmental monitoring, domain control, and hardware control of the server. Using the command line interface, you can configure the platform and domains, power on and off domains, power supplies, fans, and other components. Access to the Solaris domain console is also possible using the command line interface.

Before You Read This Book

This book is written for system administrators or people who have a working knowledge of the Solaris™ operating environment. If you do not have such knowledge, you should first read the Solaris User and System Administrator AnswerBooks and consider UNIX® system administration training. For the documentation URLs, see "Accessing Sun Documentation Online" on page xv.

How This Book Is Organized

This book contains the following chapters:

Chapter 1 introduces you to the system controller and briefly describes platform administration and domain administration.

Chapter 2 explains system controller command syntax, command names, and command arguments.

Chapter 3 provides in tabular form a summary of all of the system controller commands and describes each command.

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.
`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, such as *board_name.*	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	`#`
Platform shell	*schostname*`:SC>`
Domain shell	*schostname*`:A>` or `B>, C>, D>`

Related Documentation

Application	Title	Part Number
Service	*Sun Fire 6800/4810/4800/3800 Systems Service Manual*	805-7363
Service	*Sun Fire 4810/4800/3800 System Cabinet Mounting Guide*	806-6781
System Administration	*Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual*	805-7373

Accessing Sun Documentation Online

A broad selection of Sun system documentation is located at:

`http://www.sun.com/products-n-solutions/hardware/docs`

A complete set of Solaris documentation and many other titles are located at:

`http://docs.sun.com`

Ordering Sun Documentation

`fatbrain.com`, an Internet professional bookstore, stocks select product documentation from Sun Microsystems, Inc.

For a list of documents and how to order them, visit the Sun Documentation Center on `fatbrain.com` at:

`http://www.fatbrain.com/documentation/sun`

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 (805-7372-13) of your document in the subject line of your email.

Introduction to the System Controller

The system controller consists of the System Controller board and the system controller software.

The system controller software does the following:

■ Monitors and controls the system
■ Manages hardware
■ Configures domains
■ Provides the date and time to the Solaris operating environment
■ Provides the clock signal used on all system boards
■ Provides a platform console and a domain console
■ Provides system monitoring and control using SNMP for use with the Sun Management Center 3.0 software.

For more information on the system controller, refer to the "Overview" chapter of the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual* and also the *Sun Fire 6800/4810/4800/3810 Systems Overview Manual*.

Platform and Domain Administration

The platform administration function manages resources and services that are shared among the domains. With this function, you can configure how resources and services are configured and shared.

The domain administration function manages resources and services for a specific domain. With this function you can configure the domain, control the keyswitch position, and access the domain console.

For more information on the platform administration and domain administration functions, refer to the "Overview" chapter of the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual.*

Connecting to the System Controller

To connect to the system controller, refer to the "System Controller Navigation Procedures" chapter in the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual.*

System Controller Navigation

To navigate within the system controller, refer to the "System Controller Navigation Procedures" chapter in the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual.*

Terminating Sessions

To terminate system controller sessions, refer to the "System Controller Navigation Procedures" chapter in the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual.*

System Controller Syntax, Arguments, and Device Names

This chapter describes the following topics:

Command Syntax and Arguments

The general syntax of system controller commands is:

command_name [*flags*] [*arguments*]

Arguments are the words that follow the command name and are divided into two categories:

- Required arguments
- Optional arguments

Optional arguments always follow the option flags on the command line. In the following example, the optional *domainID* argument, a, follows the option flag, -d. The *boardname* argument, which is sb2 in this example, is a required argument and is the board name for a CPU/Memory board.

```
schostname:SC> addboard -d a sb2
```

Command Names

Most system controller commands are in the form of *verbnoun*. Command names are in the English language. System controller commands are case insensitive.

Note – When typing system controller commands, you can type the complete command name or type just enough of the command name to uniquely identify it.

Component Names

TABLE 2-1 shows the component names that are accessible from the platform. TABLE 2-2 shows the component names that are accessible from the domain. In both tables, you can type the component names in either upper case or lower case.

Note – The components in TABLE 2-1 and TABLE 2-2 depend on the system you have. For example, only the Sun Fire 6800 system can have six power supplies and six CPU/Memory boards.

TABLE 2-1   Components That Are Accessible From the Platform

Device Description	Device Name
Power grids*	GRID0, GRID1
Power supplies	PS0, PS1, PS2, PS3, PS4, PS5
CPU/Memory boards	SB0, SB1, SB2, SB3, SB4, SB5
I/O assemblies	IB6, IB7, IB8, IB9
Repeater boards	RP0, RP1, RP2, RP3
ID/Source board	ID0
Fan trays	FT0, FT1, FT2, FT3
System controller	SSC0, SSC1

* Power grids are not a component but a division of the Sun Fire 6800 system into two distinct halves of the system. Power supplies ps0, ps1, and ps2 comprise power grid 0. Power supplies ps3, ps4, and ps5 comprise power grid 1.

TABLE 2-2 shows the components that are accessible from a domain.

TABLE 2-2   Components That Are Accessible From a Domain

Device Description	Device Name
CPU/Memory boards	SB0, SB1, SB2, SB3, SB4, SB5
I/O Assemblies	IB6, IB7, IB8, IB9

Board States for CPU/Memory Boards and I/O Assemblies

TABLE 2-3 lists the board states for the CPU/Memory boards and I/O assemblies. To determine the board state, use the `showboards` command and look under the `State` header.

TABLE 2-3   Board States

Board States	Description
Available	The board is not assigned to any domain.
Assigned	The board belongs to a domain, but the hardware has not been configured or it is not in use.
Active	The board is being actively used by the domain to which it has been assigned. You cannot reassign an active board.

The System Controller board, ID board, power supplies, and Repeater boards do not have a State. A dash is displayed In the Board Status field for these boards and components (see TABLE 2-4). Fan trays have a state of off, low speed, and high speed.

Board Test Status

TABLE 2-4 describes the Status field of the `showboards` command.

TABLE 2-4   Status Field of the `showboards` Command

Test Status	Description
Passed	All board components passed testing.
Failed	The board failed testing and is not usable. This could also indicate corrupt or incompatible firmware.
OK	The component is functioning properly.
Under Test	The domain is running POST (power-on self-test); testing has been initiated using the `setkeyswitch` command, the `testboard` command, or an SNMP operation.

TABLE 2-4   Status Field of the `showboards` Command   *(Continued)*

Test Status	Description
Not Tested	The board has not been tested.
Degraded	Certain components on the board have failed or are disabled. A board is degraded when there are still usable parts on the board.
–	The slot is empty or not applicable for this device.

Domain Status

In the `showplatform` and `showdomain` commands, one of the fields in the command output is domain status. The main values of domain status are (TABLE 2-5):

TABLE 2-5   Values for the Domain Status Column in `showplatform` and `showdomain` Command Output

Domain State	Description
Powered Off	The domain is not active and the keyswitch is in the off position.
Standby	The domain is not active and the keyswitch is in the standby position.
Running POST	The domain is active and is running POST (power-on self-test).
Active	The domain is active.
Active - OpenBoot PROM	The domain is active and is running the OpenBoot PROM.
Active - Booting	The domain is active and is booting the Solaris operating environment.
Active - Solaris	The domain is active and is running the Solaris operating environment.
Active - Halted	The Solaris operating environment is halted in the domain.
Active - Reset	The domain has had an XIR reset and had not be rebooted.
Active - Panicking	The domain is active and is panicking. It is creating the core file.
Active - Debugger	The domain is active and the debugger is presently running.
Not Responding	The domain is not responding.
Paused due to an error	The domain is paused due to a hardware error.

Special Characters

To execute multiple commands on the same line, separate them with a semicolon (;).
The following example executes both the addboard and deleteboard commands.

> *schostname*:A> **addboard sb2;deleteboard sb3**

A pound sign (#) signifies the start of a comment on the current line. The following
example executes the addboard command. Everything you type after the # and
before pressing the Return key is ignored.

> *schostname*:A> **addboard sb2** #this text is ignored

Interactive Commands

Some commands may prompt for confirmation before executing the command. You
can disable prompting by specifying the -y or -n flags, which answer yes or no to
any prompted question.

Some commands, such as setupplatform and setupdomain, are always
interactive.

● **When an interactive command prompts for input, do the following:**

  ■ Press the Return key to keep the current setting, displayed in braces [ ].

  ■ Type a dash (-) to change the current setting to an empty string (if the input
    type is used to define a string).

Command Line Editing

The system controller has a basic command line editor that allows you to edit the command line. The keys you press are *not* echoed on the screen.

TABLE 2-6 lists the keystrokes used to perform basic command line editing.

- To use the Control key sequences, hold down the Control key while typing the following character (TABLE 2-6).
- To use Escape key sequences, press and release the Escape key. Then press and release the following character (TABLE 2-6).

TABLE 2-6   Command Line Editing Capabilities

Keystroke	Description
Backspace (Control-h)	Erases the character before the cursor.
Delete	Erases the character before the cursor.
Control-U	Erases the entire line.
Control-C	Cancels editing the command line. The command is not executed.
Return	Finishes editing the line.
Control-b	Moves backward one character.
Esc-b	Moves backward one word.
Control-f	Moves forward one character.
Esc-f	Moves forward one word.
Control-a	Moves backward to the beginning of the line.
Control-e	Moves forward to the end of the line.
Esc-h	Erases from the beginning of the word to just before the cursor. Stores erased characters in a save buffer.
Control-w	Erases from the beginning of the word to just before the cursor. Stores erased characters in a save buffer.
Control-d	Erases next character.
Esc-d	Erases from the cursor to the end of the word. Stores erased characters in a save buffer.
Control-k	Erases from the cursor to the end of the line. Stores erased characters in a save buffer.
Control-u	Erases the entire line. Stores erased characters in a save buffer.

TABLE 2-6   Command Line Editing Capabilities *(Continued)*

Keystroke	Description
Control-r	Retypes the line.
Control-q	Quotes the next character. Allows you to insert Control characters.
Control-y	Inserts the contents of the save buffer before the cursor.

Command Line History

Each shell keeps a short history of all entered commands. To display the command line history, use the `history` command. After recalling the commands, you can either edit them or execute them by pressing the Return key. A history of at least eight previously typed command lines are saved.

TABLE 2-7 lists keystrokes you type to recall previously typed commands.

TABLE 2-7   Keystrokes that Recall Previously Typed Command Lines

Keystroke	Description
Control-p	Selects and displays the previous line for subsequent editing.
Control-n	Selects and displays the next line for subsequent editing.
Control-l	Displays the entire command history list.

Command Completion

Using the command completion function makes it easier for you to enter long commands. After typing part of the command, use the completion keystroke (TABLE 2-8). This action searches the list of available commands for commands beginning with the characters you typed thus far.

- If there is only one possible command, the remainder of the characters are entered for you automatically.
- If there are several possibilities, any characters that are common to all of the possible selections are filled in.
- If there are no defined commands starting with the given characters, the characters are erased until there is a minimum of one selection for the remaining characters.

TABLE 2-8 lists the keystrokes to complete a command.

TABLE 2-8 Keystrokes Used to Complete System Controller Commands

Key Sequence	Description
Control-space	Complete this command.
Control-?	Shows all possible matches.
Control-/	Shows all possible matches.

CHAPTER **3**

System Controller Alphabetical Command Reference

This chapter lists a summary of the system controller commands designed for an end-user on the Sun Fire 6800/4810/4800/3800 systems, describes the system controller command line interface, and provides descriptions and examples for each command.

Command Summary

TABLE 3-1 lists and describes the system controller commands and how you can access them. Many commands are accessible from both the platform shell and the domain shell. The system controller commands may differ in how they are used, the effect of the command, and the scope of the command between the platform and domain shells.

TABLE 3-1   System Controller Command Summary

Command	Description	Platform Shell	Domain Shell
addboard	Assigns a board to a domain.	x	x
break	Sends a Break signal to the domain console.		x
connections	Displays connections to the system controller to a domain.	x	x
console	Connects to a domain from the platform.	x	
deleteboard	Unassigns a board from a domain.	x	x
disablecomponent	Adds a component to the blacklist.	x	x
disconnect	Disconnects the current or specified connection.	x	x

TABLE 3-1 System Controller Command Summary *(Continued)*

Command	Description	Platform Shell	Domain Shell
dumpconfig	Saves the system controller configuration to a server.	x	
enablecomponent	Deletes a component from the blacklist.	x	x
flashupdate	Updates the firmware.	x	
help	Provides basic help information for commands.	x	x
history	Shows the command history with date and time stamps.	x	x
password	Sets the shell password.	x	x
poweroff	Powers off components.	x	x
poweron	Powers on components.	x	x
reboot	Reboots the system controller.	x	
reset	Resets the domain in the domain shell.	x	x
restoreconfig	Restores the system controller configuration from a server.	x	
resume	Exits the domain shell and resumes access to the domain console.		x
setdate	Sets the date and time.	x	x
setdefaults	Sets the configuration to default values.	x	x
setkeyswitch	Sets the keyswitch position.	x	x
setupdomain	Configures the domain.		x
setupplatform	Configures the platform.	x	
showboards	Shows board information.	x	x
showcomponent	Shows state of a component.	x	x
showdate	Shows the date and time.	x	x
showdomain	Displays the configuration and status of the domain.		x
showenvironment	Shows environmental information.	x	x
showkeyswitch	Displays the virtual keyswitch setting.	x	x
showlogs	Displays the logs.	x	x
showplatform	Displays the configuration and information for the platform and status of the domains.	x	

TABLE 3-1　System Controller Command Summary　*(Continued)*

Command	Description	Platform Shell	Domain Shell
showresetstate	Shows CPUs after a reset or a RED MODE trap.		x
showsc	Shows the system controller version and uptime.	x	
testboard	Tests the CPU/Memory board.	x	x

Alphabetical Listing of System Controller Commands

The following sections describe the system controller commands.

addboard

Assigns a board to a domain.

Scope

platform shell, domain shell

Syntax for the Platform Shell

```
addboard -d  domainID system_board_name [system_board_name . . .]
addboard -h
```

Syntax for the Domain Shell

```
addboard  system_board_name [system_board_name . . .]
addboard -h
```

Options/Parameters

−d specifies a *domainID* as an option (platform shell only).

domainID is a, b, c, or d.

−h displays help for this command.

system_board_name is the board to be added. Possible values for *system_board_name* are sb0–sb5 (CPU/Memory board) or ib6 - ib9 (I/O assembly).

Description

Assigns *system_board_name* to the specified domain within the platform shell or to the current domain. The board state must be *Available*. To display the board state, use the `showboards` command. For more information on board states, see "Board States for CPU/Memory Boards and I/O Assemblies" on page 6.

To assign a board to a domain using a domain shell, the board name must be listed in the Access Control List (ACL) for the current domain. When a board is listed in the ACL, the system controller software is allowed to process `addboard` requests on that board. The platform shell does not use the ACL and can always add boards to a domain. The platform shell can also delete boards from a domain and either ignores the ACL or overrides the ACL. You configure the ACLs and also restrict the domains a board has access to using the `setupplatform` command.

If a board is assigned to an active domain, the board will not be used. To unassign a board from a domain, you must halt the Solaris operating environment in the domain. For an overview of steps to perform, see the section "Assigning and Unassigning Boards" in the "Maintenance" chapter of the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual*.

If a board is not present, the command assigns ownership of the slot to the specified domain.

See Also

`deleteboard, setkeyswitch, showboards, showdomain, showplatform`, "Assigning and Unassigning Boards" in the "Maintenance" chapter of the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual* (for a step-by-step procedure on assigning boards to a domain)

Example—Platform Shell

From the platform shell to add board name `sb2` to domain A (CODE EXAMPLE 3-1), type:

CODE EXAMPLE 3-1   `addboard` Example in the Platform Shell

```
schostname:SC> addboard -d a sb2
```

Example—Domain Shell

To assign board name `sb2` to the current domain (CODE EXAMPLE 3-2), type:

CODE EXAMPLE 3-2   `addboard` Example in the Domain Shell

```
schostname:A> addboard sb2
```

break

Sends a Break signal to the domain console.

Scope

domain shell

Syntax

```
break [-y|-n]
break -h
```

Options/Parameters

-h displays help for this command.

-y answers yes to the warning message. Does not prompt for confirmation.

-n answers no to the warning message. Does not execute this command if confirmation is requested.

Description

Sends a Break signal to the domain console. Resumes the domain console.

Note – This command pauses the Solaris operating environment.

When the Solaris operating environment is running in the domain, the usual effect of the break signal is to force entry into OpenBoot PROM or the debugger. The Solaris operating environment will ignore the break signal if the keyswitch is set to secure.

See Also

```
resume, setkeyswitch, showkeyswitch
```

Example

CODE EXAMPLE 3-3 shows using the `break` command to pause the Solaris operating environment and enter the OpenBoot PROM.

CODE EXAMPLE 3-3 `break` Command Example

```
schostname:A> break

This will suspend Solaris in domain A.
Do you want to continue? [no] yes
Type 'go' to resume.
debugger entered.

{1} ok
```

connections

Displays connections to the system controller or a domain.

Scope

platform shell, domain shell

Syntax for the Platform Shell

```
connections [-d domainID]
connections -h
```

Syntax for the Domain Shell

```
connections [-h]
```

Options/Parameters

-h displays help for this command.

-d *domainID* shows connections to the specified domain (a, b, c, or d).

Description

Displays hosts currently connected to the system controller. When run from the platform shell, this command shows the connections to the platform and to each domain.

When run from the domain shell, this command will only show the connection to the domain. There can only be one connection to each domain.

See Also

```
disconnect
```

Example—Platform Shell

CODE EXAMPLE 3-4   connections Command for the Platform Shell

```
schostname:SC> connections
ID       Hostname      Idle Time     Connected On     Connected To
--       --------      ---------     ------------     ------------
1        XXXXXXX       -             May 02 00:00     Platform
4        XXX           -             May 02 00:21     Platform

schostname:SC
```

TABLE 3-2 describes the headers in the output for CODE EXAMPLE 3-4.

TABLE 3-2   Header Description for the connections Command

Header Column in the connections **Command**	Description
ID	Connection ID. This is a unique number.
Hostname	Source of the connection for this session. If the value is localhost, the connection was established through the platform or through the serial port.
Idle Time	Amount of time the session is sitting without any activity. Displays a dash (-) if there has been any activity during the last 30 seconds.
Connected On	Date and time the connection was initiated. The format is Mon dd hh:mm.
Connected To	Shell or console this session is connected to. Possible values are Platform or *domainIDs* A, B, C, or D.

console

Connects to a domain from the platform.

Scope

platform shell

Syntax

```
console [-d] domainID
console -h
```

Options/Parameters

-d *domainID* is the domain to connect to and is a, b, c, or d. The -d parameter is optional and does not need to precede the *domainID*.

-h displays help for this command.

Description

Connect to a domain. You use the `console` command to navigate from the platform to a domain. If a password was created for connecting to this domain, you must enter the password.

If the domain is active (the domain is running the Solaris operating environment, OpenBoot PROM, or POST), you are connected to the domain console. Otherwise, you are connected to the domain shell. To obtain the domain shell from the domain console, see the chapter "System Controller Navigation Procedures" in the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual*.

See Also

```
disconnect
```

Examples—Platform Shell

When the domain is not active and there is no password initialized for this domain:

CODE EXAMPLE 3-5   `console` Example and No Password Set Accessing Domain B

```
schostname:SC> console b

Connected to Domain B

Domain Shell for Domain B

schostname:B>
```

When a domain is not active and a password is initialized for this domain:

CODE EXAMPLE 3-6   `console` Example and A Password Set Accessing Domain A

```
schostname:SC> console a

Enter Password:

Connected to Domain A

Domain Shell for Domain A

schostname:A>
```

When a domain is active and a password is not set for this domain:

```
schostname:SC> console a

Connected to Domain A
```

Note that no other output is seen. If the OpenBoot PROM is at the `ok` prompt or the Solaris operating environment is at the `login:` prompt, you must press the Enter key in order to see the prompt. Otherwise, you will need to wait for output to be generated by POST, the OpenBoot PROM, or the Solaris operating environment.

If the domain is hung, there will be no output. However, when a domain displays no output, this does not necessarily mean that the domain is hung.

deleteboard

Unassigns a board from the domain where it is currently assigned.

Scope

platform shell, domain shell

Syntax for the Platform Shell and the Domain Shell

deleteboard *system_board_name* [*system_board_name* . . .]

deleteboard [-h]

Options/Parameters

-h displays help for this command.

system_board_name is the board to be deleted. Values are sb0 to sb5 (CPU/Memory board) and ib6 to ib9 (I/O assembly).

Description

Unassigns a board from the domain where it is currently assigned. When you use this command from your current domain, you can only unassign boards that are assigned to the current domain. The CPU/Memory board or the I/O assembly board status must be in the *Assigned* state and the board must *not* be part of an active domain. To display the board states, use the showboards command. For more information on board states, see "Board States for CPU/Memory Boards and I/O Assemblies" on page 6. The board does *not* have to be in the Access Control List (ACL) to be unassigned from the domain.

To unassign a board from a domain, you must halt the Solaris operating environment in the domain. For an overview of steps to perform, see the section "Assigning and Unassigning Boards" in the "Maintenance" chapter of the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual*.

If a board is not present, the command unassigns ownership of the slot from the specified domain.

See Also

addboard, showboards, "Assigning and Unassigning Boards" in the "Maintenance" chapter of the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual* (for a step-by-step procedure on unassigning boards from a domain)

Example—Platform Shell

To delete I/O assembly ib7, type:

CODE EXAMPLE 3-7   deleteboard Example Showing Deleting I/O Assembly 7

```
schostname:SC> deleteboard ib7
```

Example—Domain Shell

To delete CPU/Memory board sb3 from the current domain, domain A, type:

CODE EXAMPLE 3-8   deleteboard Example Showing Deleting CPU/Memory Board 3

```
schostname:A> deleteboard sb3
```

disablecomponent

Adds a component to the blacklist.

Scope

platform shell, domain shell

Syntax

disablecomponent *component_name* [*component_name . . .*]

disablecomponent -h

Options/Parameters

–h displays help for this command.

component_name is (see TABLE 3-3 and TABLE 3-4):

- *system_board_name/port/physical_bank/logical_bank* for the CPU/Memory board
- *system_board_name/port/bus* for an I/O assembly
- *system_board_name/card* for an I/O assembly

Note – You can use partial component names.

TABLE 3-3 *component_name* Descriptions for the CPU/Memory Board Used in the `disablecomponent` Command

Board or Device	Component Name
CPU system	*board_name/port/physical_bank/logical_bank*
CPU/Memory boards (*system_board_name*)	SB0, SB1, SB2, SB3, SB4, SB5
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

TABLE 3-4 *component_name* Descriptions for the I/O Assembly Used in the `disablecomponent` Command

Board or Device	Component Name
I/O assembly system	*board_name/port/bus* or *board_name/card*
I/O assemblies (*system_board_name*)	IB6, IB7, IB8, IB9
Ports on the I/O assembly	P0, P1
Buses on the I/O assembly	B0, B1
I/O cards in the I/O assembly	C0, C1, C2, C3, C4, C5, C6, C7—the number of cards varies with the I/O assembly

Description

This command adds a component to the blacklist. Blacklisting provides a list of components that will not be tested and will not be configured into the domain when you change the domain's keyswitch from an inactive state to an active state or reboot the domain.

Blacklist a component or device if you believe it may be failing intermittently.

When you disable any component, the subcomponents are also disabled. For example, if you disable a CPU, the Ecache and memory that are also controlled by the CPU is automatically disabled.

Devices can be the following (see TABLE 3-3 and TABLE 3-4):
- Ports (CPU on a CPU/Memory board and I/O controller on an I/O assembly)
- Physical and logical memory banks
- I/O buses
- I/O cards

Blacklisting is done on a per domain basis. When you run the `disablecomponent` command in a domain shell, the component is blacklisted for the current domain. When you run the `disablecomponent` command from the platform shell, the component is disabled in all of the domains.

Using this command, the platform shell blacklists supersede the domain shell blacklists. For example, if a component is disabled in the platform shell, it will be disabled in all domains.

See Also

`enablecomponent`, `showcomponent`, and the "Disabling Components" section in the "Troubleshooting" chapter of the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual* for step-by-step procedure on adding a component to the blacklist.

Examples

CODE EXAMPLE 3-9 adds *system_board_name* sb4 to the blacklist.

CODE EXAMPLE 3-9 disablecomponent Command Example Adding sb4 to the Blacklists

```
schostname:A> disablecomponent sb4
```

CODE EXAMPLE 3-10 adds *system_board_name* sb0 and CPU port 3 to the blacklists. Note that any memory banks on this CPU port are unreachable and are implicitly disabled.

CODE EXAMPLE 3-10 disablecomponent Command Example Adding sb0 CPU Port 3 to the Blacklist

```
schostname:A> disablecomponent sb0/p3
```

CODE EXAMPLE 3-11 adds *system_board_name* sb2, CPU port 3, physical bank 0 to the blacklists. Note that any logical banks belonging to this physical bank are disabled.

CODE EXAMPLE 3-11 disablecomponent Command Example Adding sb2, CPU Port 3, and Physical Bank 0 to the Blacklists

```
schostname:A> disablecomponent sb2/p3/b0
```

CODE EXAMPLE 3-12 adds *system_board_name* ib9, port 0, bus 1 and *board_name* ib8, I/O card 2 to the blacklists. Disabling I/O card 2 shuts down the power to that I/O card.

CODE EXAMPLE 3-12 disablecomponent Command Example Adding I/O Assembly 9, Port 0 and Bus 1 and Also I/O Assembly 8, I/O Card 2 to the Blacklists

```
schostname:A> disablecomponent ib9/p0/b1 ib8/c2
```

disconnect

Disconnects the current or specified connection.

Scope

platform shell, domain shell

Syntax for the Platform Shell

```
disconnect [ID]
disconnect -h
```

Syntax for the Domain Shell

```
disconnect [-h]
```

Options/Parameters

-h displays help for this command.

ID (platform shell only) is the connection to be terminated. *ID* is the number displayed under the heading ID for the `connections` command.

Description

Terminates a connection to the system controller. If this command is used with no arguments, it disconnects the current session. For illustrations of the disconnect command, see the chapter "System Controller Navigation Procedures" in the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual*.

If the connection was initiated from another system, you will be returned to the remote host. If the session was initiated from the platform shell, you will see nothing. To obtain the prompt, press the Return key. If you are connected from the serial port, then you will return to the main menu.

See Also

`connections`

Examples

To disconnect a session from the system controller platform shell with a `telnet` connection (CODE EXAMPLE 3-13), type:

CODE EXAMPLE 3-13   Disconnecting a Session From the System Controller Platform Shell

```
schostname:SC> disconnect
Connection closed by foreign host.
```

If you are connected to the system controller platform console with the serial connection, you will see the following after typing `disconnect`
(CODE EXAMPLE 3-14):

CODE EXAMPLE 3-14   Disconnecting a Session From the System Controller Platform Console

```
schostname:SC> disconnect

    Type 0 for Platform Shell

    Type 1 for domain A
    Type 2 for domain B
    Type 3 for domain C
    Type 4 for domain D

    Input:
```

To disconnect a session from a system controller domain shell (CODE EXAMPLE 3-15), type:

CODE EXAMPLE 3-15   `disconnect` Example Showing Disconnecting a Session From the Domain A Shell

```
schostname:A> disconnect
Connection closed by foreign host.
```

CODE EXAMPLE 3-15 displays a direct connection to the domain and *is not* a connection to the domain made from the platform.

Note – When you disconnect from a domain shell, you will see nothing. Press the Return key to obtain the prompt.

---

dumpconfig

Saves the platform and domain configurations to a server.

Scope

platform shell

Syntax

`dumpconfig -f` *url*

`dumpconfig -h`

Options/Parameters

`-h` displays help for this command.

`-f` specifies the URL, which must use the `ftp` protocol. The URL must point to a directory and not a file. Examples are:

`ftp://`*userid*`:`*password@host*`/`*path*

`ftp://`*host*`/`*path*

Note – The hostname you enter can be the hostname or a hostname with a fully qualified domain name. You cannot use a partial domain name. For example: *schostname*, *schostname*`.eng.sun.com` can be a valid hostname .

Note – The directory specified in the URL path must have write permission for the specified user. If a user was not specified, then anonymous ftp is used.

Description

Saves the platform and domain configurations to a server for recovery in case of a system controller failure. This command creates two data files at the specified URL: *schostname*.nvci and *schostname*.tod.

Use this command when you complete setting up the initial configuration of the platform and the domains, each time you modify the configuration, or change the hardware configuration. For example, when you type one of the following commands: setupplatform, setupdomain, setdate, addboard, deleteboard, enablecomponent, disablecomponent, and password. The data files include *schostname*.nvci and *schostname*.tod. By running this command again, the new platform and domain configurations are saved to a server.

Caution – This command should be run any time you change the platform or domain configuration (see the list of commands in the first paragraph). Invoking this command is *very important* because if the System Controller board fails and you did not use this command to save the platform and domain configurations, you will have to manually reconfigure the platform and the domains.

This command is intended to be used as part of the procedure in replacing a failed System Controller board. You *should not* use this command to revert to an old configuration. The showplatform, showdomain, showcomponent, and showboards system controller commands display all of the configuration and can be used as a reference.

If you need to replace the System Controller board, this configuration information can be used to restore the platform and domain configurations to the replacement System Controller board.

See Also

restoreconfig

Example

CODE EXAMPLE 3-16   dumpconfig Example

```
schostname:SC> dumpconfig -f ftp://hostname/path
Created: ftp://hostname/path/schostname.nvci
Created: ftp://hostname/path/schostname.tod
```

enablecomponent

Deletes a component from the blacklist.

Scope

platform shell, domain shell

Syntax

enablecomponent *component_name* [*component_name . . .*]

enablecomponent -h

Options/Parameters

–h displays help for this command.

component_name is (TABLE 3-5 and TABLE 3-6)

- *system_board_name/port/physical_bank/logical_bank* for CPU/Memory boards
- *system_board_name/port/bus* for an I/O assembly
- *system_board_name/card* for an I/O assembly

TABLE 3-5   *component_name* Descriptions for the CPU/Memory Board Used in the
enablecomponent Command

Board or Device	Component Name
CPU system	*board_name/port/physical_bank/logical_bank*
CPU/Memory boards (*system_board_name*)	SB0, SB1, SB2, SB3, SB4, SB5
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

TABLE 3-6 *component_name* Descriptions for the I/O Assembly Used in the
`enablecomponent` Command

Board or Device	Component Name
I/O assembly system	*board_name*/*port*/*bus* or *board_name*/*card*
I/O assemblies (*system_board_name*)	IB6, IB7, IB8, IB9
Ports on the I/O assembly	P0, P1
Buses on the I/O assembly	B0, B1
I/O cards in the I/O assembly	C0, C1, C2, C3, C4, C5, C6, C7—the number of cards varies with the I/O assembly

Description

Removes a component from the blacklist. Blacklisting provides a list of components that will not be tested and will not be configured into the domain when you change the keyswitch setting from an inactive state to an active state or when the domain is rebooted.

Components can be the following:

- Ports (CPU on the CPU/Memory board or I/O controller on the I/O assembly)
- Physical and logical memory banks
- I/O buses
- I/O cards

Blacklisting is done on a per domain basis. When you run the `enablecomponent` command from the domain, the component is removed from the blacklist for the current domain. When the `enablecomponent` command is run from the platform shell, the component is removed from the blacklist for all domains.

Using this command, the platform shell blacklists supersede the domain shell blacklists. For example, if a component is enabled in the platform shell, it will be enabled in all domains.

See Also

`disablecomponent`, `showcomponent`, and the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual* for step-by-step procedure on removing a component from the blacklist. This information is in the 'Troubleshooting" chapter in the section "Disabling Components."

Examples

CODE EXAMPLE 3-17 enables CPU/Memory board `sb4`. This removes CPU/Memory board `sb4` from the blacklist.

CODE EXAMPLE 3-17   `enablecomponent` Command Example Enabling CPU/Memory Board 4

```
schostname:A> enablecomponent sb4
```

CODE EXAMPLE 3-18 enables I/O assembly 6, port 1. This removes port 1 of I/O assembly 6 from the blacklist.

CODE EXAMPLE 3-18   `enablecomponent` Command Example Enabling I/O Assembly 6 and Port 1

```
schostname:A> enablecomponent ib6/p1
```

flashupdate

Updates the firmware on the system controller and the system boards (CPU/Memory boards and I/O assemblies). The source flash image can be on a server or another board of the same type.

Scope

platform shell

Syntax

```
flashupdate [-y|-n] -f url all|systemboards|scapp|rtos|domainID
flashupdate [-y|-n] -f url systemboards|scapp|rtos|domainID
flashupdate [-y|-n] -f url board [board ... ]
flashupdate [-y|-n] -u
flashupdate [-y|-n] -c source_board destination_board [destination_board ...]
flashupdate -h
```

Options/Parameters

-h displays help for this command.

-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
url is the URL to the directory containing the flash images.
where:

```
ftp://hostname/path
ftp://userid:password@hostname/path
http://hostname/path
```

board is the board name.

-c specifies that the image should be copied from another board.

- *source_board* is the source board for the flash images.
- *destination_board* is the destination board for the flash images.

-u automatically updates all of the boards from the board with the highest revision.

all updates the system controller, all system boards (CPU/Memory boards and I/O assemblies), and the system controller real time operating system (rtos).

system_boards are all CPU/Memory boards and I/O assemblies.

scapp updates the current system controller. Updating the system controller reboots the system controller and disconnects all the current connections. To update the other system controller, you must run the flashupdate command from the other system controller. Use showsc to determine which system controller you are currently using so you know the system controller you must go to next to update it.

rtos updates the real time operating system for the system controller. If you have a second System Controller board installed, when you update the real time operating system also update the rtos on the second System Controller board. To update the rtos on the second system controller, run the flashupdate command from the second System Controller board.

domainID (a, b, c, or d) upgrades all boards that are assigned to this domain.

Description

Updates the firmware on the system controller, CPU/Memory boards, and I/O assemblies. There is no firmware on Repeater boards. Boards must be powered on in order to be updated with flashupdate. If a board does not have power, the flashupdate command will do nothing and stop with an error message. Boards that are in a secure domain *cannot* be updated.

Caution – When you update the firmware on the system controller, update only one system controller. DO NOT update both system controllers at the same time.

Note – Any time the firmware is upgraded, check the `Install.info` file for the firmware upgrade procedure. This file is provided with your latest software release. Also refer to the *Release Notes* for any notes or special procedures.

If you install a replacement board into the system:

1. Check if the firmware level of the replacement board is the same as the board you replaced. Incompatible boards are noted by the `Failed` board state in the output of the `showboards` command.

2. If the board level is not the same as the board you replaced, perform the `flashupdate` procedure on the board.

3. If a board was noted by the `Failed` state in `showboards`, after you `flashupdate` a compatible version, power off the board to clear the `Failed` state.

Before you update the flash images, make sure the firmware on the replacement board is compatible with the firmware of the board you are replacing for `scapp`, CPU/Memory board, and I/O assembly PROMs. New firmware is loaded if you reboot the domain or turn the domain keyswitch off with `setkeyswitch off` and then turn it on with `setkeyswitch on`.

■ To determine the version number of the current firmware:

1. Use the `showboards -p proms` command.

2. Compare the *x.y.z* number format. If *x* and *y* are the same, then the flash images are compatible.

    The new firmware may also contain a new `rtos` image. If the `rtos` image that is currently installed is different than the `rtos` supplied with the software, you must upgrade the `rtos` image.

■ To determine the version number of the current `rtos`:

1. Use `showsc`. The README file contains the version number of the new `rtos` image.

    There is currently no method to determine that the `scapp` and `rtos` versions are compatible.

Note – If you fail to check all versions, the domains can crash or the system controller would fail to reboot normally.

Caution – While upgrading the firmware, heed the following precautions:

- Read the README and Install.info files before you upgrade the firmware.
- *Do not* change the keyswitch position of any domain.
- *Do not* use DR.
- *Do not* power off any boards.
- *Do not* reboot the system controller.

If you upgrade the scapp or rtos images for status:

1. It is important to watch the console during the flashupdate procedure.

2. Monitor the console if errors are reported.

If the images that are already installed are *incompatible* with the new images:

1. Shut down the domains before performing the firmware upgrade.

2. After you perform the firmware upgrade, reboot the domains.

If the images that are already installed are *compatible* with the new images:

1. Reboot the domains after performing the flashupdate procedure. Even though the firmware is compatible with the older firmware version, you must upgrade the firmware to obtain new features and bug fixes.

If you install a replacement board into the system:

1. Check if the firmware level of the replacement board is the same as the board you replaced.

2. If is it not, perform the flashupdate procedure on the board.

Caution – Under normal circumstances, the flashupdate command will complete successfully. However, if the flashupdate command is terminated abnormally (such as a power failure, a failed network connection, and so on), the system controller goes into maintenance mode.

To bring the system controller out of maintenance mode and back into its normal operating mode, type the flashupdate command with a valid URL or use the reboot command to boot the system controller with an older firmware version.

See Also

Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual

`flashupdate` Command Platform Shell—Examples

Note – In the following examples, since the output is very long, only the command you type is listed in the code box.

Note – The `flashupdate all`, `flashupdate rtos`, and `flashupdate scapp` commands cause the system controller to reboot once.

- To update the active System Controller board and the system controller real time operating system (`rtos`), perform this procedure from the platform console and watch the console output. This command reboots the system controller. Type:

CODE EXAMPLE 3-19 `flashupdate` Example Updating the Active System Controller
Board and the System Controller Real Time Operating System

```
schostname:SC> flashupdate -f ftp://host/path scapp rtos
```

- To upgrade the firmware on a replacement CPU/Memory board, `sb4` (destination_board), which is a replacement CPU/Memory board for CPU/Memory board, `sb0` (*source_board*). Type:

CODE EXAMPLE 3-20 `flashupdate` Example Updating Replacement CPU/Memory Board
sb4

```
schostname:SC> flashupdate -c sb0 sb4
```

help

Provides basic help information for commands.

Scope

platform shell, domain shell

Syntax

`help` [*command_name*] | [*partial_command_name*]

`help -h`

Options/Parameters

`-h` displays help for this command.

command_name is the name of the command.

partial_command_name can be one letter of the command or a portion of the command name, such as `show`.

Description

The `help` command, without arguments, lists currently available commands. When an argument is supplied, the `help` command displays a list of commands that begin with the specified argument. If only one command is found, full help is displayed. Otherwise, a short description is displayed for each command, beginning with the specified argument.

You can also type a partial command name and basic help information will be returned for all commands matching the partial name. The partial command name must contain enough of the command name in order for the command to be recognized by help. The `help` command returns help information for all commands beginning with the characters you typed.

The `help` command is shell sensitive. It displays help information for the current shell only.

Examples

CODE EXAMPLE 3-21 displays help information on the `addboard` command.

CODE EXAMPLE 3-21 `help` Command Example Displaying Information on the `addboard` Command

```
schostname:SC> help addboard

addboard -- assign a board to a domain

Usage: addboard -d <domain> <board> ...
       addboard -h

    -d -- the domain to assign the board to
    -h -- display this help message
```

CODE EXAMPLE 3-22 displays all commands in the platform shell beginning with show.

CODE EXAMPLE 3-22 `help` Command Example Displaying Commands Beginning With the Word `show`

```
schostname:SC> help show

showboards         -- show board information
showcomponent      -- show state of a component
showdate           -- show the current date and time for the platform
showenvironment    -- show environmental information
showkeyswitch      -- show the keyswitch positions
showlogs           -- show the logs
showplatform       -- show platform configuration, information and domain status
showsc             -- show system controller version and uptime
```

CODE EXAMPLE 3-23 displays all commands in the platform shell beginning with the letter a.

CODE EXAMPLE 3-23   help Command Example Displaying Commands Beginning With the Letter A

```
schostname:SC> help a

addboard -- assign a board to a domain

Usage: addboard -d <domain> <board> ...
       addboard -h

    -d -- the domain to assign the board to
    -h -- display this help message
```

history

Shows the command history with date and time stamps.

Scope

platform shell, domain shell

Syntax

`history [-h]`

Options/Parameters

`-h` displays help for this command.

Description

Shows the command history with date and time stamps for when the commands were executed. This command is shown for your shell and displays the last twenty commands only.

See Also

Command line editing can be used to edit the command history. For more information on how to use command line editing, see "Command Line Editing" on page 9.

Example

CODE EXAMPLE 3-24   `history` Command

```
schostname:SC> history
May 07 16:29:21 : showboards
May 07 16:29:24 : showdate
May 07 16:29:29 : history
```

password

Sets the password for the platform or the domain.

Scope

platform shell, domain shell

Syntax—Platform Shell

```
password [-d domainID]
password -h
```

Syntax—Domain Shell

```
password
password [-h]
```

Options/Parameters

-d *domainID* is domain a, b, c, or d.

-h displays help for this command.

Description

Sets the password for the platform or the domain. There are separate passwords for each domain and for the platform. If you set a password, entering the password is required for access to the shell or console.

It is very important to set the password for the platform and each domain even if a domain is not being used. This prevents people from creating and activating unauthorized domains.

Prior to allowing the password to be changed, the current password will be authenticated. Changed passwords take effect immediately. The old password will no longer be accepted.

You can remove the password by pressing Return at the `Enter new password` and `Enter new password again` prompts.

OpenBoot PROM passwords are different from the platform and domain passwords. The OpenBoot PROM in each domain supports the OpenBoot PROM security mode, which is a standard feature of the OpenBoot PROM software. For more information on the OpenBoot PROM security mode password, see your OpenBoot PROM documentation.

See Also

"Security" chapter of the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual*

Examples

You will see the following prompt (CODE EXAMPLE 3-25), when a password is already set and you type the `password` command at either the platform shell or the domain shell.

CODE EXAMPLE 3-25 `password` Command Example With a Password Already Set

```
schostname:SC> password
Enter current password:
Enter new password:
Enter new password again:
schostname:SC>
```

If currently there is not a password assigned to the shell you are entering, you will not be prompted for the current password (CODE EXAMPLE 3-26).

CODE EXAMPLE 3-26 `password` Command Example With No Password Set

```
schostname:SC> password
Enter new password:
Enter new password again:
schostname:SC>
```

poweroff

Powers off components.

Scope

platform shell, domain shell

Syntax for the Platform Shell

```
poweroff [-y|-n] all|grid#|component_name [component_name ... ]
```

```
poweroff -h
```

Syntax for the Domain Shell

```
poweroff [-y|-n] all|component_name [component_name ... ]
```

```
poweroff -h
```

Options/Parameters

−h displays help for this command.

−y will answer yes to any question. This option is potentially hazardous. You can forcefully power off a component with the −y option.

−n answers no to any question. You cannot forcefully power off a component with the −n option.

component_name is the component name.

Platform shell *component_name*(s):

- all turns off all currently controllable components.
- Power grid (grid0, grid1). The Sun Fire 6800 system has two power grids: grid0 and grid1. Grid 1 controls power supplies ps3, ps4, and ps5. All other mid-range systems have one power grid, grid0. This grid controls power supplies ps0, ps1, and ps2.
- Power supply (ps0 - ps5)
- CPU/Memory board (sb0 - sb5)
- I/O assembly (ib6 - ib9)
- Repeater board (rp0 - rp3)
- Fan tray (ft0 - ft3)

Domain shell *component_name(s)*:

- CPU/Memory board (sb0 - sb5)
- I/O assembly (ib6 - ib9*)*

Note – The specified board must be in the current domain.

Description

Powers off a component or a list of components.

For the domain shell, the specified board must be in the current domain.

All slots listed by showboards can be powered off except for system controllers (SSC0 and SSC1) and the ID board (ID0). A powered-off component will not be monitored. Use the showboards command to display the power status of each board.

You can power off any component except when the board is in the Active state (see the showboards command to display the board state). When a component is in the Active state, a warning is displayed on the console that tells you the entire domain will go down. You are asked to confirm with a warning of the consequences.

If a board is active in a domain and you forcefully power off a device (board), the keyswitch for the domain will be set to standby. To change the keyswitch setting, use the setkeyswitch command.

Repeater boards do not have an Active state. You cannot power off a Repeater board if it is being used by a domain.

You can turn off power supplies if turning off the power supply would leave sufficient power in the power grid to support the current load of the power grid. This varies with the configuration. It you attempt to power off a power supply that is required to support the current load, the power supply will not be powered off and a message stating why is displayed. You can power off redundant power supplies.

If powering off a fan would result in insufficient cooling, it cannot be powered off.

See Also

poweron, setkeyswitch, showboards

Examples

To power off CPU/Memory board, `sb2`, from the platform shell
(CODE EXAMPLE 3-27), type:

CODE EXAMPLE 3-27 `poweroff` Command Example Showing Powering Off `sb2`

```
schostname:SC> poweroff sb2
```

To power off all power supplies, fan trays, and system boards (CODE EXAMPLE 3-28),
type:

CODE EXAMPLE 3-28 `poweroff` Command Example Showing Powering Off All Power
Supplies, Fan Trays, and System Boards in the Platform Shell

```
schostname:SC> poweroff all
```

Note – From the domain shell, you cannot power off power supplies, fan trays,
Repeater boards, or power grids. To power off these components, connect to the
platform shell.

poweron

Powers on components.

Scope

platform shell, domain shell

Syntax for the Platform Shell

poweron all|*grid#*|*component_name* [*component_name* . . .]

poweron -h

Syntax for the Domain Shell

poweron all|*component_name* [*component_name* . . .]

poweron -h

Options/Parameters

-h displays help for this command.

Platform shell *component_name*(s):

- ■ all is all power supplies and boards.
- ■ Power grid (grid0, grid1). The Sun Fire 6800 system has two power grids: grid0 and grid1. Grid 1 controls power supplies ps3, ps4, and ps5. All of the other mid-range systems have one power grid, grid0. This grid controls power supplies ps0, ps1, and ps2.
- ■ Power supply (ps0 - ps5)
- ■ CPU/Memory board (sb0 - sb5)
- ■ I/O assembly (ib6 - ib9*)*
- ■ Repeater board (rp0 - rp3*)*
- ■ Fan tray (ft0 - ft3)

Domain shell *component_names*:

- ■ all powers on all CPU/Memory boards and I/O assemblies in the domain
- ■ CPU/Memory board (sb0 - sb5)
- ■ I/O assembly (ib6 - ib9*)*

The specified board must be in the current domain.

Description

Powers on a component or a list of components You *must* specify *component_name*(s).

You can power on any component that is not already powered on if there is sufficient power to support the component when it is powered on. From the domain, only components that are assigned to the domain can be powered on. To power on a component, the power grid power must also be on.

Fan trays are automatically powered on when a power supply is powered on. Fan trays obtain power from the power grid(s).

The System Controller boards obtain their power from a secondary power output on the power supplies. The System Controller boards power on automatically when the system is supplied with power. For example, the System Controller boards will power on automatically on a Sun Fire 6800 system if the power cords are connected, the system has power, and the circuit breaker(s) are on. The System Controller board will automatically power on the Sun Fire 4810/4800/3800 systems if the power cords are connected to the power source, the system has power, and the power supply switches are turned on (Sun Fire 3800 system only).

The fan trays power on automatically if any power supply is on.

See Also

poweroff, setkeyswitch, showboards

Examples

To power on CPU/Memory board, sb2 from the platform shell (CODE EXAMPLE 3-29), type:

CODE EXAMPLE 3-29   poweron Example Powering On sb2

```
schostname:SC> poweron sb2
```

To power on CPU/Memory boards and I/O assemblies in the domain
(CODE EXAMPLE 3-30), type:

CODE EXAMPLE 3-30 `poweron` Example Showing Powering On All CPU/Memory Boards
and I/O Assemblies in Domain A

```
schostname:A> poweron all
```

Note – From the domain shell, you cannot power on power supplies, Repeater
boards, fan trays, or power grids. To power on these components, use the platform
shell.

reboot

Reboots the system controller.

Scope

platform shell

Syntax

```
reboot [-y|-n]
reboot -h
```

Options/Parameters

-y does not prompt for confirmation.

-n does not execute this command. Confirmation is requested.

-h displays help for this command.

Description

Reboots the system controller. The reboot command will not change any configuration settings. Active domains will continue to run.

The keyswitch may be set to off if the system controller reboots while a keyswitch operations is in progress.

⚠ **Caution –** Rebooting interrupts any current operation. This includes keyswitch operations, rebooting the Solaris operating environment, testing boards, SNMP, and so on.

Examples

To reboot the system controller (CODE EXAMPLE 3-31), type:

CODE EXAMPLE 3-31   `reboot` Command Example

```
schostname:SC> reboot
Are you sure you want to reboot the System Controller now? [no]
```

reset

Resets the domain.

Scope

domain shell

Syntax

```
reset [-y|-n][-x|-a]
reset -h
```

Options/Parameters

-y answers yes to the question asked (executes the command).

-n answers no to the question asked (does not execute the command).

-h displays help for this command.

Options/Parameters

With no arguments, typing reset is the same as typing reset -x.

-x resets via XIR (externally initiated reset). Use XIR to try to obtain diagnostic data (default).

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

Description

This command also resumes the domain console. Resetting the domain is not allowed if the keyswitch is in the secure position. Before you can reset the domain, change the keyswitch position to on. Resetting the domain is not possible if the keyswitch is configured in either the off or standby position.

By default, reset uses XIR (externally initiated reset) to reset the CPUs in the domain. The externally initiated reset forces control of the domain into the OpenBoot PROM and begins the OpenBoot PROM error reset recovery actions. The error reset recovery actions preserve most domain states to allow collecting data needed for debugging the hardware and software, including a Solaris operating environment core file. The OpenBoot PROM error reset recovery actions are

controlled by setting the OpenBoot PROM `error-reset-recovery` configuration variable. For the definition of this variable and the various settings, see TABLE 3-10 in "`setupdomain`" on page 76.

Note that the OpenBoot PROM `reset` command *does not* generate a core file as this command does.

You cannot reset a domain that has been paused. The domain is paused automatically when hardware detects an error. Messages on the domain console indicate that the domain is paused.

To take a domain out of the paused state:

1. Turn the keyswitch off with `setkeyswitch off`.

2. Turn the keyswitch on with `setkeyswitch on`.

See Also

`resume, setkeyswitch, setupdomain, showdomain, showkeyswitch, showresetstate`, "Domain Not Responding" section in the "Troubleshooting" chapter in the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual.* This section describes how to recover from a hung or paused domain.

Examples

To reset the domain (from a domain shell), type the following (CODE EXAMPLE 3-32). This command uses XIR (externally initiated reset) to try to obtain diagnostic data.

CODE EXAMPLE 3-32   `reset` Command Example From Domain A

```
schostname:A> reset
```

To perform the equivalent operation as the OpenBoot PROM `reset-all` command (CODE EXAMPLE 3-33), type:

CODE EXAMPLE 3-33   `reset` -a Command Example From Domain A

```
schostname:A> reset -a
```

restoreconfig

Restores the platform and domain configurations from a server.

Scope

platform shell

Syntax

```
restoreconfig [-y|-n] -f url
restoreconfig -h
```

Options/Parameters

-h displays help for this command.

-y does not prompt for confirmation.

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

-f specifies a URL, which must use the ftp protocol.

   url is the directory containing the data files. The URL must point to a directory and not a file. Examples are:

      `ftp://`*userid*`:`*password*`@`*hostname*`/`*path*

      `ftp://`*hostname*`/`*path*

Note – The hostname you enter can be the host name or the host name with a fully qualified domain name, such as *schostname* or *schostname*`.eng.sun.com`. You *cannot* use a partial domain name.

Description

Restores the platform and domain configurations from a server if a system controller board has failed and has been replaced. This is based on the two data files created by the dumpconfig command, *schostanme*`.nvci` and *schostname*`.tod,` at the specified URL, which must point to a directory and use the ftp protocol. This command prompts you for a new platform password.

Caution – The dumpconfig command is intended to be used as part of replacing a failed system controller. For instructions on how to use dumpconfig, see "To Use the dumpconfig Command to Save the Current Platform and Domain Configurations to a Server" in the chapter "System Power On and Setup" in the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual*. This command should have been executed when you set up the system.

This command requires that all domains be powered off with the setkeyswitch off command. The system controller automatically reboots when the configuration is restored. The date and time need to be set after the configuration has been restored and the system controller has been rebooted. You also need to activate all domains with setkeyswitch on, since this command restores all domains in the off keyswitch position.

Note – Check the configuration of the platform and the domains before activating any domains to ensure that the desired configuration has been restored. In the platform shell use showplatform, showboards, and showdate. In each domain shell use showdomain, showboards, and showdate.

See Also

dumpconfig, setkeyswitch, setdate, showboards, showdate, showdomain, showplatform

Example

CODE EXAMPLE 3-34 shows an example of the restoreconfig command with the required –f option.

CODE EXAMPLE 3-34   restoreconfig Example

```
schostname:SC> restoreconfig -f ftp://hostname/path

The system controller will be rebooted when the restore is complete. The date
will need to be set in the platform and each domain.

Do you want to restore the system controller configuration now [no] yes

Retrieving: ftp://hostname/path/schostname.nvci
Retrieving: ftp://hostname/path/schostname.tod
Verifying data
```

```
NVCI has been restored
TOD has been restored

Restore complete. The system controller is being rebooted.
The date will need to be set in the platform and each domain.

Software Reset . . .
```

resume

Exits the domain shell and resumes access to the domain console.

Scope

domain shell

Syntax

`resume [-h]`

Options/Parameters

`-h` displays help for this command.

Description

Exits the domain shell and resumes access to the domain console. This command requires an active domain. If the domain is not active, there is no domain console and there is nothing to resume. When you connect to the domain console, there may not be any output displayed. If either the OpenBoot PROM or the Solaris operating environment is running in the domain, press the Return key to get the prompt. Otherwise, POST output is displayed (may be a delay in displaying POST output).

Other system controller commands that also resume the domain console are listed in the following section.

See Also

`break`, `reset`, `setkeyswitch`

Example

CODE EXAMPLE 3-35   `resume` Command Example in Active Domain A

```
schostname:A> resume
```

Note that no other output is seen. If the OpenBoot PROM is at the `ok` prompt or the Solaris operating environment is at the `login:` prompt, you must press the Enter key in order to see the prompt. Otherwise, you will need to wait for output to be generated by POST, the OpenBoot PROM, or the Solaris operating environment.

If the domain is hung, there will be no output. However, when a domain displays no output, this does not necessarily mean that the domain is hung.

setdate

Sets the date and time.

Scope

platform shell, domain shell

Syntax

`setdate` [`-v`] [`-t` *time zone*] [*mmdd*]*HHMM*

`setdate` [`-v`] [`-t` *time zone*] *mmddHHMM* [[*cc*]*yy*][*.SS*]

`setdate` [`-v`] [`-r` *datehost*]

`setdate` [`-v`] `-t` *time zone*

`setdate` [`-v`] `-t` GMT*<+|->* *offset from GMT (*TABLE 3-7*)*

`setdate -h`

Options/Parameters

`-t` *time zone* sets the time zone using the time zone abbreviation (TABLE 3-7). Only non-daylight savings time zones can be specified. If you are in an area with daylight time or summer time, the time and time zone are adjusted automatically for daylight time or summer time.

`-t` GMT*<+|->offset* from Greenwich Mean Time—*GMT* (TABLE 3-7).

mm is the month number.

dd day is the number in the month.

HH is the hour number (24-hour clock).

MM is the minute number.

cc is the century minus 1.

yy is the last two digits of the year number.

SS is the second number. For the century minus 1 value, use 21 (Twenty first century) minus 1, which yields 20.

–r *datehost* sets the current time using `rdate`. The host must be a valid system name and the system must support `rdate` requests. The hostname you enter can be the host name or the host name with a fully qualified domain name, such as *schostname* or *schostname*.`eng.sun.com`. You *cannot* use a partial domain name.

–v is verbose mode. Displays detailed information about the time zone that is set.

–h displays help for this command.

TABLE 3-7   Time Zone Abbreviations, Time Zone Name, and Offsets From Greenwich Mean Time

Time Zone Abbreviation	Time Zone Name	Offset From Greenwich Mean Time (GMT)
ACT	Australian central time	GMT+9.5
AET	Australian eastern time	GMT+10
AGT	Argentina standard time	GMT-3
ART	Arabic (Egypt) standard time	GMT+2
AST	Alaska standard time	GMT-9
BET	Brazil eastern time	GMT-3
BST	Bangladesh standard time	GMT+6
CAT	Central African time	GMT+2
CNT	Canada Newfoundland time	GMT-3.5
CST	Central standard time	GMT-6
CTT	China Taiwan time	GMT+8
EAT	Eastern African time	GMT+3
ECT	European central time	GMT+1
EET	Eastern European time	GMT+2
EST	Eastern standard time	GMT-5
HST	Hawaii standard time	GMT-10
IET	Indiana eastern standard time	GMT-5
IST	India standard time	GMT+5.5
JST	Japan standard time	GMT+9
MET	Middle East time	GMT+3.5
MIT	Midway Islands time	GMT-11
MST	Mountain standard time	GMT-7
NET	Near East time	GMT+4

Time Zone Abbreviation	Time Zone Name	Offset From Greenwich Mean Time (GMT)
NST	New Zealand standard time	GMT+12
PLT	Pakistan Lahore time	GMT+5
PNT	Phoenix standard time	GMT-7
PRT	Puerto Rico and U S. Virgin Islands time	GMT-4
PST	Pacific standard time	GMT-8
SST	Solomon standard time	GMT+11
UTC	Universal Time Coordinated	GMT+0
VST	Vietnam standard time	GMT+7

Description

Sets the date and time for the platform and domains. This command, when invoked from the platform, will have no effect on the date and time in each domain and vice versa. You can set up to five different times and time zones; one time and time zone for the platform and different times and time zones for each of the four domains.

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

Note – You cannot set the date from the system controller in a domain while the Solaris operating environment is running. To set the date while the Solaris operating environment is running, use the Solaris operating environment date command.

See Also

showdate

Examples

When you type the `setdate` command from the platform, this sets the date and time for the platform. When you type the `setdate` command from the domain, this sets the date and time for the domain.

To set the date and time on the platform to Thursday, April 20, 2001, at 18 hours 15 minutes and 10 seconds, type:

CODE EXAMPLE 3-36 `setdate` Command Example in the Platform Shell

```
schostname:SC> setdate 042018152001.10
Thu Apr 20 18:15:10 PST 2001
```

To set the date from a date host:

CODE EXAMPLE 3-37 `setdate -r` Command Example Setting the Date From a Date Host

```
schostname:SC> setdate -r datehost
Thu Apr 20 18:15:10 PST 2001
```

To set the time zone to Pacific Standard Time (PST), using the offset from Greenwich mean time—GMT, and the date and time on the platform to Thursday, April 20, 2001, at 18 hours 15 minutes and 10 seconds (TABLE 3-7), type:

CODE EXAMPLE 3-38 `setdate -t` Command Example Setting the Time Zone to Pacific Standard Time Using the Offset From Greenwich Mean Time

```
schostname:SC> setdate -t GMT-8 042018152001.10
Thu Apr 20 18:15:10 PST 2001
```

To set the time zone to Eastern Standard Time (EST), using the time zone abbreviations, and the date and time on the platform to Thursday, April 20, 2001, at 18 hours 15 minutes and 10 seconds (TABLE 3-7), type:

CODE EXAMPLE 3-39 `setdate -t` Command Example Setting the Date and Time Zone to Eastern Standard Time Using Time Zone Abbreviations

```
schostname:SC> setdate -t EST 042018152001.10
Thu Apr 20 18:15:10 EST 2001
```

To set just the time zone for Japan Standard Time using offsets from Greenwich
Mean Time—GMT (TABLE 3-7) and *not* the date and time, type:

CODE EXAMPLE 3-40   `setdate -t` Command Example Setting the Time Zone to Japan
Standard Time Using the Offset From Greenwich Mean Time

```
schostname:SC> setdate -t GMT+9
Thu Apr 20 18:15:10 GMP+9 2001
```

To set just the time zone for European Central Time using the time zone
abbreviations (TABLE 3-7) and *not* the date and time, type:

CODE EXAMPLE 3-41   `setdate -t` Command Example Setting the Time Zone to Eastern
Central Time Using Time Zone Abbreviations

```
schostname:SC> setdate -t ECT
Thu Apr 20 18:15:10 ECT 2001
```

setdefaults

Sets the default configuration values.

> **Caution –** This is a destructive command. Use with caution and care.

Scope

platform shell, domain shell

Syntax

```
setdefaults [-y|-n] [-c][-p platform]
setdefaults -h
```

Options/Parameters—Platform Shell

-y does not prompt for confirmation. The option sets the defaults for the platform shell and each domain shell.

-n does not execute the command if confirmation is requested.

-c keeps Capacity on Demand (COD) licenses.

-h displays help for this command.

-p platform resets the defaults for the platform only.

Options/Parameters—Domain Shell

-y does not prompt for confirmation.

-n does not execute the command if confirmation is requested.

-h displays help for this command.

Description

 Caution – This command requires that all domains are inactive (not running the OpenBoot PROM, POST, or the Solaris operating environment) and the keyswitch be set to off.

When you run this command from the platform shell without options, the platform and domain values are set to the default values. When you run this command from the domain shell without options, only the current domain values are reset to the default values.

If a password is set, you will need to type the password in order to set the default values. When you run this command from the platform shell, the password for the platform shell is required. When you run this command from a domain shell, the password for the domain shell is required. There are separate passwords for the platform shell and each domain shell. In addition, this command also does the following:

- Sets the system controller to the default values. This command only resets the system controller setting. It does not affect the domain's configuration in the OpenBoot PROM.
- Erases all platform and domain configurations (platform).
- Removes passwords (domain and platform).

Example

CODE EXAMPLE 3-42 `setdefaults` Example

```
schostname:SC> setdefaults

You are about to reset all configuration data to default values.
All domain configurations, ACLs, passwords and data buffers will
be lost.

The system controller will be REBOOTED after the defaults are set.

Do you want to restore the default values and reboot now? [no]
```

If you answer yes, the system controller is rebooted after the defaults are set.

If you answer no, the default values are not restored.

If a password is set for the platform shell, the password will be required.

setkeyswitch

Changes the position of the virtual keyswitch to the specified value.

Scope

platform shell, domain shell

Syntax for the Platform Shell

setkeyswitch [-y|-n] -d *domainID* off

setkeyswitch -h

Syntax for the Domain Shell

setkeyswitch [-y|-n] [off|standby|on|diag|secure]

setkeyswitch -h

Options/Parameters

-h displays help for this command.

-y does not prompt for confirmation.

-n does not execute this command is confirmation is requested.

-d *domainID* (platform shell only) is the domain ID of the domain you want power off (a, b, c, or d).

For `setkeyswitch` parameters and descriptions, see TABLE 3-8.

TABLE 3-8   Description of the `setkeyswitch` Transition

Value	Description
off	Changes the board state of all the boards belonging to a domain to the Assigned board status, and places the boards in low-power mode, which allows you to remove the boards from the system. The domain will not be initialized at system power on.
standby	Changes the board state of all the boards belonging to a domain to the Assigned board state and turns on the boards. The domain will not be initialized at system power on, but the boards that comprise the domain will be powered on.
on	Powers on and initializes the domain. The system controller brings the domain into OpenBoot PROM through POST. If the OpenBoot PROM has `auto-boot?` set to `true`, then the Solaris operating environment boots automatically.
diag	Similar to the `setkeyswitch on` command except the POST verbose mode is set to on and the POST diagnostic level is set to maximum.
secure	Similar to the `setkeyswitch on` command except that the `break` command and the `reset` commands are ignored. CPU/Memory board and I/O assembly flash PROM updates are not allowed.

TABLE 3-9 shows the results when you change a keyswitch setting.

TABLE 3-9   Results of Changing the Keyswitch From the Current Setting to a New Setting

Current Setting	New Setting	Result
off	off	Keyswitch position not changed.
off	standby	Powers on all of the boards. The boards belonging to a domain should already be in the Assigned state. The domain will not be initialized at system power on.
off	on	Powers on and initializes the domain. The domain will go through POST and the OpenBoot PROM. The Solaris operating environment boots automatically unless you have set the OpenBoot PROM `auto-boot?` set to false.
off	diag	Similar to on, except POST verbose and diagnostic level is to set to maximum. The domain will boot from the `diag-device`.
off	secure	Similar to on, except the `reset` and `break` commands are ignored by the domain shell.

Current Setting	New Setting	Result
standby	off	All boards will be powered off (set to low -power mode and all monitoring of the boards will stop). The domain is shut down.
standby	on	Similar to setting the virtual keyswitch from off to on except that you do not have to wait for the system controller to turn on the boards and prepare them.
on	off	All boards will be powered off (set to low -power mode and all monitoring of the boards will stop). The domain is shut down.
on	standby	The domain becomes inactive. All of the boards remain powered on.
on	on	Keyswitch position not changed.
on	diag	Similar to the `setkeyswitch on` command except the POST verbose mode is set to on and the POST diagnostic level is set to maximum. This change is setting has no effect on a running domain.
on	secure	`break` and `reset` are ignored.
diag	off	All boards will be powered off (set to low -power mode and all monitoring of the boards will stop). The domain is shut down.
diag	standby	The domain becomes inactive. All of the boards remain powered on.
diag	on	Has no effect on a running domain. This operation only changes the keyswitch position and does not perform any other function since the domain is already running.
diag	diag	Keyswitch position not changed.
diag	secure	`break` and `reset` are ignored.
secure	off	All boards will be powered off (set to low -power mode and all monitoring of the boards will stop). The domain is shut down.
secure	standby	The domain becomes inactive. All of the boards remain powered on. The `break` and `reset` commands are no longer ignored.
secure	on	`break` and `reset` are not ignored.
secure	diag	This change is setting has no effect on a running domain.
secure	secure	Keyswitch position not changed.

Description

Each domain has a virtual keyswitch with five positions: off, standby, on, diag, and secure. The `setkeyswitch` command changes the position of the virtual keyswitch to the specified value. The virtual keyswitch replaces the need for a physical keyswitch for each domain.

If it is not possible to access the domain shell, set the keyswitch position for the domain to off from the platform shell. Setting the keyswitch position to off for a domain without first halting the Solaris operating environment may cause problems. If the domain is running the Solaris operating environment, the `setkeyswitch` command parameters `off` and `standby` require confirmation.

If the domain is already powered on, the `setkeyswitch on`, `setkeyswitch diag`, and `setkeyswitch secure` commands will change only the position of the virtual keyswitch. If components making up a domain such as a I/O assembly or CPU/Memory board need to be powered on, the system attempts to power on these boards or devices also.

When the keyswitch position changes to on, secure, or diag from any keyswitch position, the domain console is resumed.

If a failure occurred during keyswitch transition, the keyswitch position is reset to `off` or `standby` after recovering from the failure.

The system controller maintains the position of each keyswitch even during power failures or physical power cycles. When you power on the system after either a power failure or system power off, the system controller activates any domains that were active when the power went off.

See Also

`resume, showkeyswitch`

Examples

To turn the virtual keyswitch on and power on the domain (CODE EXAMPLE 3-43), type:

CODE EXAMPLE 3-43 `setkeyswitch on` Example

```
schostname:A> setkeyswitch on
```

To shut down the domain by turning the virtual keyswitch off (CODE EXAMPLE 3-44), type:

CODE EXAMPLE 3-44 `setkeyswitch off` Example

```
schostname:A> setkeyswitch off
```

To shut down domain B from the platform shell (CODE EXAMPLE 3-45), type:

CODE EXAMPLE 3-45 `setkeyswitch off` Example Shutting Down Domain B From the Platform Shell

```
schostname:SC> setkeyswitch -d b off
```

Caution – CODE EXAMPLE 3-45 does not properly shut down the Solaris operating environment in the domain before turning the keyswitch in domain B to the off position.

setupdomain

Configures the domain.

Scope

domain shell

Syntax

setupdomain [-p *part*]...

setupdomain -h

Options/Parameters

-h displays help for this command.

-p *part* configures the specified information and has the following options:

- bootparams configures the boot parameter information.
- loghost configures the log hosts
- snmp configures SNMP information.

TABLE 3-10 lists the parameter values for setupdomain. Note that for diag-level definition and function of default and max are the same. Because the tests for mem1 and mem2 take a long time to run, use the default or max values. However, if you suspect memory problems, then use the mem1 or mem2 values.

TABLE 3-10 Parameter Values for the setupdomain Command

Parameter	Value	Description
diag-level	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 using few tests with few test patterns.
	default (default value)	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. This is the same as max.

TABLE 3-10 Parameter Values for the `setupdomain` Command *(Continued)*

Parameter	Value	Description
	max	All system board components are tested with all tests and test patterns. This is the same as `default`.
	mem1	Runs all tests at the default level plus more exhaustive DRAM and SRAM test algorithms. These tests take a long time to run.
	mem2	This is the same as `mem1` with the addition of a DRAM test that does explicit compare operations of the DRAM data. These tests take a long time to run.
verbosity-level	off	No status messages are displayed. POST output is always displayed. However, with the value set to `off`, there is very little POST output displayed and there may be a long time between output displays. This depends on your hardware configuration and the setting of `diag-level`.
	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 status is displayed.
interleave-scope	within-cpu	The memory banks on a CPU will be interleaved with each other. Each CPU may have up to four logical memory banks.
	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 domain.
		With this value, you cannot remove a CPU/Memory board from the system using DR.
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 domain will be paused when the system controller detects a hardware error.

TABLE 3-10 Parameter Values for the `setupdomain` Command *(Continued)*

Parameter	Value	Description
	true	The domain is rebooted when a hardware error is detected. If `OBP.auto-boot?` is set to true, the Solaris operating environment is booted.
`OBP.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.
`OBP.auto-boot?`		Controls booting of the Solaris operating environment.
	true (default value)	If this value is true, the domain 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 environment.
`OBP.error-reset-recovery`		Controls the behavior of the domain after an externally initiated reset (XIR) as well as a red mode trap.
	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 domain. A core file is not generated. Rebooting a domain 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.

TABLE 3-10   Parameter Values for the `setupdomain` Command   *(Continued)*

Parameter	Value	Description
Loghost		The name or the IP address of the loghost for this domain (SNMP).
Log Facility		The log facility is defined by the Solaris operating environment `syslogd` in `/etc/syslog.conf`. The default is `local0`.
Domain Description		Enter a brief description for this domain. For example, its function.
		Enter the name of the primary domain administrator.
Trap Hosts		Enter the name or the IP address of the trap hosts for this domain. You can define multiple trap hosts.
Public Community String		Community string for SNMP readers. The default value is A-public, B-public, and so on.
		Note—For SNMP clients such as the Sun Management Center 3.0 software to access the system controller using SNMP, their community strings should be set to the same value as the value entered here.
Private Community String		Community string for readers and writers. The default value is A-private, B-private, and so on.
		Note—For SNMP clients such as the Sun Management Center 3.0 software to access the system controller using SNMP, their community strings should be set to the same value as the value entered here.

Description

Configures domain specific values. Configures the domain. You can configure each domain differently and independently This command is an interactive command. You are prompted for parameters to configure.

See Also

`password, setdate, showdate, showdomain,` and the section "To Configure Domain-Specific Parameters" in the "System Power On and Setup" chapter of the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual*.

Example

CODE EXAMPLE 3-46   Variables for the `setupdomain` Command

```
schostname:A> setupdomain

Domain Boot Parameters
----------------
diag-level [default]:
verbosity-level [min]:
error-level [max]:
interleave-scope [within-board]:
interleave-mode [optimal]:
reboot-on-error [false]:
OBP.use-nvramrc? [<OBP default>]:
OBP.auto-boot? [<OBP default>]:
OBP.error-reset-recovery [<OBP default>]:

Log hosts
---------
Loghost [ ]: The hostname or the IP address of the loghost for this domain.
Log Facility [local0]:  Defined by the Solaris operating environment syslogd in /etc/syslog.conf.

SNMP
-----
Domain Description[]:  A brief description for this domain (its function).
Domain Contact[ ]:  The name of the primary domain administrator.
Trap Hosts [ ]:  The IP address of the loghost for this domain.
Public Community String [ ]: The community string for SNMP readers.
Private Community String [ ]:  The community string for readers and writers.

schostname:A>
```

Note – The default value displayed by the software for the `OBP.*` parameters is `<OBP default>`.

If you enter an invalid value, the system controller displays "invalid entry" and prompts you to enter a proper value (TABLE 3-10). If a value is not specified, it retains its current value. When an invalid entry is entered, it also lists the valid responses. Typing a dash (-), clears the entry.

Current values are displayed in [ ]. TABLE 3-10 lists the `setupdomain` parameter values.

setupplatform

Configures the platform specific variables.

Scope

platform shell

Syntax

`setupplatform [-p part]. . .`

`setupplatform -h`

Options/Parameters

`-h` displays help for this command.

`-p` *part* configures the specified part and has the following options:

- `acls` is ACLs (Access Control List)

  - `-` clears the boards from the ACL

  - `+` adds all boards to the ACL

  - `-a` *board* adds a board

  - `-d` *board* deletes a board

- `loghost` configures loghost and facility with the default of local0. The loghost is the IP address or host name of the syslog loghost. The host name you enter must be a complete host name (not a partial one).
- `network` is the network settings.
- `partition` sets the partition mode.
- `sc` configures system controller behavior.
- `snmp` is SNMP configuration

Description

Configures parameters for the platform. This command is an interactive command. You are prompted for parameters to configure. For parameters to configure, see TABLE 3-11.

TABLE 3-11 `setupplatform` Parameter Values

Parameter	Description
`Is the System Controller on a Network?`	If the answer is no, leave all network parameters as unconfigured. You will be prompted to enter the hostname. If the answer if yes, you are prompted to configure the network settings.
`Use DHCP or static network settings?`	• `DHCP` means the system controller network configuration is retrieved from a DHCP server. • `Static` means the network settings will be manually configured. If you select the static setting, you will be asked for the following network parameters.
`Hostname` (Network parameter)	The human readable network identity for this system controller.
`IP address` (Network parameter)	The network identity used by computers.
`Netmask` (Network parameter)	For this value, specify how much of the address should be reserved for subdividing networks into subnetworks. Must be specified in dot-notation address. For example `255.255.255.0`.
`Gateway` (Network parameter)	IP address of the gateway.
`DNS Domain` (Network parameter)	Domain name. For example, *xxx*.*xxx*.`com`.
`Primary DNS Server` (Network parameter)	IP address of your primary DNS server.
`Secondary DNS Server` (Network parameter)	IP address of your secondary DNS server.
`Loghost`	The name or the IP address of the loghost for this domain (SNMP).
`Log Facility`	The log facility. Facility is defined by the Solaris operating environment `syslogd` in `/etc/syslog.conf`. The default is `local0`.
`SNMP Agent?`	The SNMP agent enables or disables SNMP. The default is SNMP disabled.
`Platform Description`	Default value is the platform model name.

TABLE 3-11   `setupplatform` Parameter Values   *(Continued)*

Parameter	Description
Platform Contact	The name of the person who is responsible for this system. This name will be used by SNMP. Note that if SNMP is not going to be enabled, this field is informational only.
Platform Location	Location of the system (such as a room and/or the location inside of the room). This information will be used by SNMP. Note that if SNMP is not going to be enabled, this field is informational only.
Trap Hosts	IP address or name of the trap hosts (SNMP). You can define multiple trap hosts.
Public Community String	Community string for SNMP readers. The default is `P-public`. **Note**—For SNMP clients such as the Sun Management Center 3.0 software to access the system controller using SNMP, their community strings should be set to the same value as the value entered here.
Private Community String	Community string for readers and writers. The default is `P-private`. **Note**—For SNMP clients such as the Sun Management Center 3.0 software to access the system controller using SNMP, their community strings should be set to the same value as the value entered here.
ACL for Domain A	Access Control List (ACL). If the board name is listed in the ACL for the specific domain, then you have permission to perform `addboard` or `deleteboard` requests on that board. Default value is to have all slots assigned to all domains. • Typing the prefix `-d` deletes the slot from ACLs of this domain. • Typing a prefix of `-a` (add) adds a slot. • Typing a dash (–) clears the entire list. • Typing a + (plus sign) adds all. • Pressing the Return key does not change the value displayed inside [ ].
ACL for Domain B	See the description for ACL for Domain A.
ACL for Domain C	See the description for ACL for Domain A.
ACL for Domain D	See the description for ACL for Domain A.
SC POST diag level	The system controller boot parameters (diag level) can be set to: `off`, `min`, or `max`.
off	Only system board initialization code is run. No testing is done. POST is not run.

TABLE 3-11   `setupplatform` Parameter Values   *(Continued)*

Parameter	Description
min	All System Controller board components are tested using few locations with few test patterns.
max	All System Controller board components are tested with all tests and test patterns.
`Configure platform for single or dual partition mode`	The two options are single or dual. The default is single. For a description of single and dual partition mode, refer to the "Overview" chapter in the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual.*

See Also

`password`, `setdate`, `setupdomain`, `showdate`, `showdomain`, `showplatform`, and the section "Setting Up the Platform" in the "System Power On and Setup" chapter of the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual.* This chapter describes a step-by-step procedure on setting up the platform using `setupplatform`.

Example

CODE EXAMPLE 3-47   `setupplatform` Output

```
schostname:SC> setupplatform

Network Configuration
---------------------
Use DHCP or static network settings? [DHCP]:
Network settings? [static]:
Hostname []:
IP Address []:
Netmask []:
Gateway []:
DNS Domain []:
Primary DNS Server []:
Secondary DNS Server []:

Loghosts
--------
Loghost [ ]:
Log Facility [local0]:
```

```
SNMP
----
Platform Description [Sun Fire 6800]: System type
Platform Contact [ ]:  Platform administrator's name
Platform Location [ ]:  Geographic system location
Enable SNMP Agent? [yes]:
Trap Hosts []: SNMP trap host IP address or name
Public Community String [P-public]: Community string for SNMP readers
Private Community String [P-private]: Community string for readers and writers

ACLs
----
ACL for domain A [SB0 SB1 SB2 SB3 SB4 SB5 IB6 IB7 IB8 IB9 ]:
ACL for domain B [SB0 SB1 SB2 SB3 SB4 SB5 IB6 IB7 IB8 IB9 ]:
ACL for domain C [SB0 SB1 SB2 SB3 SB4 SB5 IB6 IB7 IB8 IB9 ]:
ACL for domain D [SB0 SB1 SB2 SB3 SB4 SB5 IB6 IB7 IB8 IB9 ]:

SC POST
-------
SC POST diag Level [min]:

Partition Mode
--------------
Configure platform for single or dual partition mode? [single]:

schostname:SC>
```

showboards

Displays the assignment information and status for all of components in the system.

Scope

platform shell, domain shell

Syntax for the Platform Shell

`showboards` [-ev] [-y|-n][-d *domainID*] [-p *part*]

`showboards -h`

Syntax for the Domain Shell

`showboards` [-aev] [-y|-n] [-p *part*]

`showboards -h`

Options/Parameters

-a include available boards (domain only).

-e includes empty slots in the output.

-v verbose mode.

-d *domainID* specify a domain to display (platform only).

-p *part* shows only a specific part and can be:
- board shows the board status.
- clock shows the system clock status.
- cpu shows CPU type, speed, and Ecache size.
- io shows I/O information.
- memory shows memory information for each board.
- power shows grid information.
- version shows version information.

-y answers yes to all questions.

-n answers no to all questions.

-h displays help for this command.

Description

Displays the assignment information and status for all of the components in the system. For example, CPU/Memory boards, I/O assemblies, fan trays, and so on. For the domain shell, the display shows components assigned to the domain and can also include boards that are included in the ACL, and not assigned to any other domain.

See Also

addboard, deleteboard

Examples—Platform Shell

CODE EXAMPLE 3-48   showboards Command for the Platform Shell

```
schostname:SC> showboards

Slot      Pwr   Component Type            State       Status    Domain
----      ---   --------------            -----       ------    ------

SSC0      On    System Controller         -           Passed    -
SSC1      -     Empty Slot                -           -         -
ID0       On    Sun Fire 3800 Centerplane -           OK        -
PS0       On    A145 Power Supply         -           OK        -
PS1       On    A145 Power Supply         -           OK        -
PS2       On    A145 Power Supply         -           OK        -
FT0       On    Fan Tray                  Low Speed   OK        -
FT1       On    Fan Tray                  Low Speed   OK        -
FT2       On    Fan Tray                  Low Speed   OK        -
FT3       On    Fan Tray                  Low Speed   OK        -
RP0       On    Repeater Board (F3800)    -           OK        -
RP2       On    Repeater Board (F3800)    -           OK        -
/N0/SB0   On    CPU Board                 Active      Passed    A
/N0/SB2   Off   CPU Board                 Assigned    Not tested C
/N0/IB6   On    CPCI I/O board (F3800)    Active      Passed    A
/N0/IB8   Off   CPCI I/O board (F3800)    Assigned    Not tested C
```

The showboards command with the -v option displays the verbose option.

TABLE 3-12 Output Header Definitions for the `showboards` and the `showboards -v` Command

Header	Description
Slot	Slot designator. The N*x* in the slot descriptor is the node number.
Pwr	Indicates if the power status of the device is off or on.
Component type	Component description.
State	Describes board state. Possible values are: Active, Assigned, Available and – (dash). The - board state means that the board state does not apply to this slot. For more information on board states, see "Board States for CPU/Memory Boards and I/O Assemblies" on page 6.
Status	Current board status. For more information on board status, see "Board Test Status" on page 6.
Domain	Indicates which domain the board belongs to. For more information on domains, refer to the "Overview" chapter of the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual*.
Memory Information	
Component	Component, such as a board, CPU, or memory DIMM.
J-No	Component label on the board.
Size	Lists the size of the DIMM.
Reason	Describes information why the value is not reported.
Firmware Versions	
Component	Board.
Device	Lists the specific flashPROM on the board.
Type	Provides the image type for the device. Describes what the image is used for. Each device can have multiple types.
Version	System controller version number.
Date	Date.
Time	Time of day.
Clock Status	
Component	Board.
SSC*X* Signal	Lists the status of the SSC0 or SS1 clock on the board.
Signal Used	Defines which clock is used by the board.
Failover	Describes if the board will respond to clock failover.

TABLE 3-12 Output Header Definitions for the `showboards` and the `showboards -v` Command *(Continued)*

Header	Description
I/O Status	
Slot	I/O slot.
Populated	Defines if a card is installed in the slot.
I/O Status (continued)	
Slot Description	Describes the slot. The card type cannot be determined by the system controller. The card is determined by the Solaris operating environment.
CPU Information	
Component	CPU.
Cpu Mask	Lists the CPU version.
Description	Describes the CPU type, speed, and Ecache size.
Power Grids	
Component	Board.
Pwr	Describes if the board is on or off.
Grid	Describes which power grid the board is in.

Example—Domain Shell

CODE EXAMPLE 3-49 `showboards -a` Command for the Domain Shell

```
schostname:A> showboards -a

Slot       Pwr   Component Type        State    Status      Domain
----       ---   --------------        ------   ------      ------
/N0/SB0    On    CPU Board             Active   Passed      A
SB2        Off   CPU Board             Available Not tested Isolated
/N0/IB6    On    CPCI I/O Board (F3800)Active   Passed      A
IB8        Off   CPCI I/O Board (F3800)Available Not tested Isolated
```

showcomponent

Shows the state of a component.

Scope

platform shell, domain shell

Syntax for the Platform Shell

```
showcomponent [-v] [component] ...

showcomponent [-v] [-d domainID] ...

showcomponent -h
```

Syntax for the Domain Shell

```
showcomponent [-v] [component] ...

showcomponent -h
```

Options/Parameters

-h displays help for this command. Includes *component_name* syntax.

boardname for CPU/Memory boards is SB0 - SB5 and for I/O assemblies is IB6 - IB9.

-d *domainID* is a, b, c, or d. Shows components assigned to the domain (platform only).

Description

Without options, this command shows the state of all components on a board. With one or more options described in the "Syntax" section, the output shows the state of one or more components on a board.

Displays the components that have been blacklisted by the disablecomponent command or components that have been marked as failed when testing using the testboard command. For CPU/Memory boards, this command provides information about CPUs, memory modules (DIMMs), and Ecache. This command displays logical memory bank size.

See Also

enablecomponent, disablecomponent, and the "Disable Component" section of the "Troubleshooting" chapter in the *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual* for a step-by-step procedure on displaying a component.

Examples

CODE EXAMPLE 3-50 shows sample output for the showcomponent sb4 command. The abbreviations for the Component field are:

- N*x*   Node name
- SB*x*   CPU/Memory board, where x is 0 – 5
- P*x*   Port, where x is 0 – 3
- B*x*   Physical memory bank, where x is 0 – 1
- L*x*   Logical memory bank, where x is 0 – 3. There are two DIMMs per logical memory bank.

CODE EXAMPLE 3-50   showcomponent sb4 Sample Output

```
schostname:SC> showcomponent sb4

Component          Status    Pending POST    Description
--------           ------    ------- ----    -----------
/NO/SB4/P0         enabled   -       pass    UltraSPARC III+,750Mhz, 8M ECache
/NO/SB4/P1         enabled   -       pass    UltraSPARC III+,750Mhz, 8M ECache
/NO/SB4/P2         enabled   -       pass    UltraSPARC III+,750Mhz, 8M ECache
/NO/SB4/P3         enabled   -       pass    UltraSPARC III+,750Mhz, 8M ECache
/NO/SB4/P0/B0/L0   enabled   -       pass    256M DRAM
/NO/SB4/P0/B0/L2   enabled   -       pass    256M DRAM
/NO/SB4/P0/B1/L1   enabled   -       pass    256M DRAM
/NO/SB4/P0/B1/L3   enabled   -       pass    256M DRAM
/NO/SB4/P1/B0/L0   enabled   -       pass    256M DRAM
/NO/SB4/P1/B0/L2   enabled   -       pass    256M DRAM
/NO/SB4/P1/B1/L1   enabled   -       pass    256M DRAM
/NO/SB4/P1/B1/L3   enabled   -       pass    256M DRAM
/NO/SB4/P2/B0/L0   enabled   -       pass    256M DRAM
/NO/SB4/P2/B0/L2   enabled   -       pass    256M DRAM
/NO/SB4/P2/B1/L1   enabled   -       pass    256M DRAM
/NO/SB4/P2/B1/L3   enabled   -       pass    256M DRAM
/NO/SB4/P3/B0/L0   enabled   -       pass    256M DRAM
/NO/SB4/P3/B0/L2   enabled   -       pass    256M DRAM
```

CODE EXAMPLE 3-50 `showcomponent sb4` Sample Output *(Continued)*

```
/N0/SB4/P3/B1/L1   enabled   -       pass   256M DRAM
/N0/SB4/P3/B1/L3   enabled   -       pass   256M DRAM
```

Definitions of the headings in CODE EXAMPLE 3-50 and CODE EXAMPLE 3-51 follow:

- `Component` is the name of the board and its components.
- `Status` is the status of the board or component.
- `Pending` means that the component will change to the indicated status (enabled or disabled) at the next reboot.
- `POST` is the status of POST (passed or failed).
- `Description` is a description of the board or component.

CODE EXAMPLE 3-51 shows sample output for the `showcomponent ib6` command. The abbreviations for the Component field are:

- Nx Node name
- IBx I/O assembly, where x is 6 – 9
- Px Port, where x is 0 – 1
- Bx Bus, where x is 0 – 1
- Cx Card in the I/O assembly, where x is 0 – 7.

CODE EXAMPLE 3-51 `showcomponent ib6` Sample Output

```
schostname:SC> showcomponent ib6

Component          Status    Pending   POST      Description
---------          -------   -------   ----      -----------
/N0/IB6/P0         enabled   -         pass      IO controller 0
/N0/IB6/P1         enabled   -         pass      IO controller 1
/N0/IB6/P0/B1      enabled   -         untest    66/33MHz PCI Bus
/N0/IB6/P0/B0      enabled   -         untest    33MHz PCI Bus
/N0/IB6/P0/B1/C0   enabled   -         untest    33MHz 5V Short PCI card
/N0/IB6/P0/B1/C1   enabled   -         untest    33MHz 5V Short PCI card
/N0/IB6/P0/B1/C2   enabled   -         untest    33MHz 5V Long/Short PCI card
/N0/IB6/P0/B1/C3   enabled   -         untest    33MHz 5V Long/Short PCI card
```

showdate

Displays the date and time.

Scope

platform shell, domain shell

Syntax for the Platform Shell

```
showdate [-tv]

showdate -h

showdate [-v] [-d domainID]
```

Syntax for the Domain Shell

```
showdate [-tv]

showdate -h
```

Options/Parameters

-h displays help for this command.

-t lists available time zones.

-v is verbose mode.

-d *domainID* is a, b, c, or d. Shows the date for the domain from the platform.

Description

Displays the current date and time for the platform and for each domain. You can obtain the date for the platform shell and each domain shell from the platform shell. If you type the command from one of the domain shells, the date and time for the current domain is returned.

See Also

setdate

Example—Platform Shell

CODE EXAMPLE 3-52 showdate Command for the Platform Shell

```
schostname:SC> showdate
Mon Apr 03 12:31:40 EDT 2001
```

showdomain

Displays the configuration and status of the domain.

Scope

domain shell

Syntax

showdomain [-v] [-p *part*]...

showdomain -h

Options/Parameters

-v is verbose mode.

-p *part* is a parameter for the information to display:

- acls shows the ACLs (Access Control List)
- bootparams shows boot parameter information
- loghosts shows the log hosts
- mac shows MAC addresses
- snmp shows SNMP configuration
- status shows domain status

-h displays help for this command.

Description

Displays the configuration and status of the domain. Some of the information is not configurable, such as the MAC address and hostID, or is configurable in the platform, such as the Access Control List (ACL), and SNMP enabled. This command without any arguments displays all the parameter values configured by the setupdomain command.

See Also

setupdomain, showdate, setupplatform, showplatform

Examples

To display parameters assigned by the `setupdomain` command, type `showdomain` (CODE EXAMPLE 3-53).

CODE EXAMPLE 3-53   `showdomain` Command Example

```
schostname:A> showdomain

Domain   Solaris Nodename   Domain Status  Keyswitch
------   ---------------   -------------  ---------
A        -                 Powered Off    off

diag-level = default
verbosity-level = min
error-level = max
interleave-scope = within-board
interleave-mode = optimal
reboot-on-error = true
OBP.use-nvramrc? = <OBP default>
OBP.auto-boot? = <OBP default>
OBP.error-reset-recovery = <OBP default>

Loghost for Domain A:
Log Facility for Domain A: local0

SNMP Agent: enabled
Domain Description:
Domain Contact:
Trap Hosts:
Public Community String:
Private Community String:

ACL for Domain A: SB0 SB1 SB2 SB3 SB4 SB5 IB6 IB7 IB8 IB9

schostname:A>
```

To display the verbose output of parameters assigned by the setupdomain command, type showdomain –v (CODE EXAMPLE 3-54).

CODE EXAMPLE 3-54   showdomain –v Command Example

```
schostname:A> showdomain -v

Domain  Solaris Nodename  Domain Status Keyswitch
------  ----------------  ------------- ---------
A       -                 Powered Off   off

Domain Boot Parameters
----------------------
diag-level = default
verbosity-level = min
error-level = max
interleave-scope = within-board
interleave-mode = optimal
reboot-on-error = false
OBP.use-nvramrc? = <OBP default>
OBP.auto-boot? = <OBP default>
OBP.error-reset-recovery = <OBP default>


                    MAC Address       HostID
                    ----------        -----
Domain A            XX:XX:XX:XX:XX:XX  XXXXXXXX

Loghosts
--------
Loghost for Domain A:
Log Facility for Domain A: local0

SNMP
----
SNMP Agent: enabled
Domain Description:
Domain Contact:
Trap Hosts:
Public Community String:
Private Community String:

SNMP packets received: 0
SNMP packets sent: 0
SNMP traps sent: 0
```

CODE EXAMPLE 3-54 `showdomain -v` Command Example *(Continued)*

```
ACLs
----
ACL for Domain A: SB0 SB1 SB2 SB3 SB4 SB5 IB6 IB7 IB8 IB9

schostname:A>
```

To display boot parameter information (CODE EXAMPLE 3-55), type:

CODE EXAMPLE 3-55 `showdomain -p bootparams` Example Displaying Boot Parameter Information

```
schostname:A> showdomain -p bootparams

diag-level = default
verbosity-level = min
error-level = max
interleave-scope = within-board
interleave-mode = optimal
reboot-on-error = false
OBP.use-nvramrc? =<OBP default>
OBP.auto-boot? = <OBP default>
OBP.error-reset-recovery = <OBP default>

schostname:A>
```

showenvironment

Displays the current environmental status, temperatures, currents, voltages, and fan status, for the system (platform shell) or the current domain (domain shell).

Scope

platform shell, domain shell

Syntax for the Platform Shell

showenvironment [-ltvuw] [-d *domainID*] [-p *part*]

showenvironment [-ltvuw] *board_name*

showenvironment -h

Syntax for the Domain Shell

showenvironment [-ltvuw] [-p *part*]

showenvironment [-ltvuw] *board_name* [*board_name* . . .]

showenvironment -h

Options/Parameters

board_name is the name of the board. Displays information for this board. You can type multiple board names.

-d *domainID* is a, b, c, or d. Displays information for boards assigned to that domain.

-h displays help for this command.

-l displays the limits that apply to each selected measurement. These values are the threshold for each measurement. Exceeding the threshold causes the status to display Max or Min.

-p *part* shows a specific part (TABLE 3-13):

TABLE 3-13 Parts for `showenvironment -p` *part*

Report	Description
currents	Displays currents (power supplies only)
fans	Displays fan states.
faults	Displays values that are suspected to be invalid.
temps	Displays temperatures only.
voltage	Displays voltages only.

-t prints header titles.

-u updates data now. Polls all sensors for new values.

-v is verbose mode. This is the equivalent of the options -l and -w.

-w shows warning thresholds.

Description

Displays the current environmental status, temperatures, currents, voltages, and fan status for the system (platform shell) or the current domain (domain shell). In a domain shell, the domain can only display boards that have been assigned to the domain, such as CPU/Memory boards and I/O assemblies. The board(s) must be powered on. Boards that are not powered on are not monitored and have no environmental data.

This command displays the minimum and maximum values. If these values are exceeded, the component (and the domain using the component) will be shut down. It also shows the warning levels when the system controllers warns you that you are approaching the limits.

TABLE 3-14 describes the `showenvironment` output headers and values.

TABLE 3-14 `showenvironment` Output Header Description

Header	Value	Description
Slot		Slot ID
Device		Device being monitored by the sensor.
Sensor		Component that measures the environmental data of the device.

TABLE 3-14 `showenvironment` Output Header Description *(Continued)*

Header	Value	Description
Value		Value returned by the sensor.
Units		Applicable unit for the sensor.
	Degrees C	Celsius
	Volts DC	Volts
	Amps	Amps
Age		Age in seconds of the reading being displayed.
Status		Values for Status. See the Value column. When the values listed in the Value column are exceeded, a warning message will be logged.
	WARNING LOW	
	NOTICE Low	
	OK	
	NOTICE High	
	WARNING HIGH	
	error	
	failed	
	ignored	
	unavailable	
LoWarn		Displayed only with the $-v$ and $-w$ options. Lists values that when exceeded a warning message will be logged.
HiWarn		Displayed only with the $-v$ and $-w$ options. Lists values that when exceeded a warning message will be logged
Min		Displayed with the $-v$ and $-l$ options. This is not a warning message. Min is when the component will be powered off (including any domains using it) to protect it.
Max		Displayed with the $-v$ and $-l$ options. This is not a fault condition. Max is when the component will be powered off (including any domains using it) to protect it.

Example

The showenvironment command display for the platform shell returns information on the slot number, device, sensor, value, units, age, and status, as shown in CODE EXAMPLE 3-56.

CODE EXAMPLE 3-56 showenvironment Platform Shell and Domain Shell Example

```
md6-sc0:SC> showenvironment

Slot    Device         Sensor        Value   Units       Age      Status
----    ---------      ---------     ------  ------      ---      ------
SSC0    SBBC 0         Temp. 0       38      Degrees C   2 sec    OK
SSC0    CBH 0          Temp. 0       48      Degrees C   2 sec    OK
SSC0    Board 0        Temp. 0       27      Degrees C   2 sec    OK
SSC0    Board 0        Temp. 1       27      Degrees C   2 sec    OK
SSC0    Board 0        Temp. 2       27      Degrees C   2 sec    OK
SSC0    Board 0        1.5 VDC 0     1.49    Volts DC    2 sec    OK
SSC0    Board 0        3.3 VDC 0     3.35    Volts DC    2 sec    OK
SSC0    Board 0        5 VDC 0       5.01    Volts DC    2 sec    OK
RP0     Board 0        1.5 VDC 0     1.49    Volts DC    5 sec    OK
RP0     Board 0        3.3 VDC 0     3.37    3Volts DC   5 sec    OK
RP0     Board 0        Temp. 0       23      Degrees C   5 sec    OK
RP0     Board 0        Temp. 1       22      Degrees C   5 sec    OK
RP0     SDC 0          Temp. 0       56      Degrees C   5 sec    OK
RP0     AR 0           Temp. 0       49      Degrees C   5 sec    OK
RP2     Board 0        1.5 VDC 0     1.49    Volts DC    5 sec    OK
RP2     Board 0        3.3 VDC 0     3.37    Volts DC    5 sec    OK
RP2     Board 0        Temp. 0       22      Degrees C   5 sec    OK
RP2     Board 0        Temp. 1       23      Degrees C   5 sec    OK
RP2     SDC 0          Temp. 0       61      Degrees C   5 sec    OK
RP2     AR 0           Temp. 0       49      Degrees C   5 sec    OK
PS0     48 VDC 0       Current 0     3.94    Amps        2 sec    OK
PS0     48 VDC 0       Temp. 0       25      Degrees C   2 sec    OK
PS0     48 VDC 1       Current 0     0.18    Amps        2 sec    OK
PS0     48 VDC 1       48 VDC 0      55.28   Volts DC    2 sec    OK
PS0     48 VDC 0       48 VDC 0      56.59   Volts DC    2 sec    OK
PS1     48 VDC 0       Current 0     4.05    Amps        2 sec    OK
PS1     48 VDC 0       Temp. 0       25      Degrees C   2 sec    OK
PS1     48 VDC 1       Current 0     0.16    Amps        2 sec    OK
PS1     48 VDC 1       48 VDC 0      55.28   Volts DC    2 sec    OK
PS1     48 VDC 0       48 VDC 0      56.33   Volts DC    2 sec    OK
PS2     48 VDC 0       Current 0     4.88    Amps        1 sec    OK
PS2     48 VDC 0       Temp. 0       27      Degrees C   1 sec    OK
PS2     48 VDC 1       Current 0     0.88    Amps        1 sec    OK
PS2     48 VDC 1       48 VDC 0      55.0    Volts DC    2 sec    OK
PS2     48 VDC 0       48 VDC 0      56.07   Volts DC    2 sec    OK
FT0     Fan 0          Cooling 0             Low         1 sec    OK
```

FT1	Fan 0	Cooling 0		Low	1 sec	OK
FT2	Fan 0	Cooling 0		Low	7 sec	OK
IB8	Board 0	1.5 VDC 0	1.51	Volts DC	4 sec	OK
IB8	Board 0	3.3 VDC 0	3.35	Volts DC	5 sec	OK
IB8	Board 0	5 VDC 0	4.98	Volts DC	5 sec	OK
IB8	Board 0	12 VDC 0	12.11	Volts DC	5 sec	
IB8	Board 0	Temp. 0	30	Degrees C	5 sec	OK
IB8	Board 0	Temp. 1	24	Degrees C	5 sec	OK
IB8	SDC 0	Temp. 0	53	Degrees C	5 sec	OK
IB8	AR 0	Temp. 0	43	Degrees C	5 sec	OK
IB8	SBBC 0	Temp. 0	41	Degrees C	5 sec	OK
IB8	IOASIC 0	Temp. 0	41	Degrees C	5 sec	OK
IB8	IOASIC 1	Temp. 1	37	Degrees C	5 sec	OK
SB0	Board 0	1.5 VDC 0	1.50	Volts DC	6 sec	OK
SB0	Board 0	3.3 VDC 0	3.33	Volts DC	6 sec	OK
SB0	SDC 0	Temp. 0	52	Degrees C	6 sec	OK
SB0	AR 0	Temp. 0	43	Degrees C	6 sec	OK
SB0	SBBC 0	Temp. 0	44	Degrees C	6 sec	OK
SB0	Board 1	Temp. 0	22	Degrees C	6 sec	OK
SB0	Board 1	Temp. 1	22	Degrees C	6 sec	OK
SB0	Cheetah 0	Temp. 0	35	Degrees C	7 sec	OK
SB0	Cheetah 0	1.8 VDC 0	1.71	Volts DC	7 sec	OK
SB0	Cheetah 1	Temp. 0	34	Degrees C	7 sec	OK
SB0	Cheetah 1	1.8 VDC 1	1.72	Volts DC	7 sec	OK
SB0	SBBC 1	Temp. 0	39	Degrees C	7 sec	OK
SB0	Board 1	Temp. 2	23	Degrees C	7 sec	OK
SB0	Board 1	Temp. 3	23	Degrees C	7 sec	OK
SB0	Cheetah 2	Temp. 0	34	Degrees C	7 sec	OK
SB0	Cheetah 2	1.8 VDC 0	1.72	Volts DC	7 sec	OK
SB0	Cheetah 3	Temp. 0	36	Degrees C	7 sec	OK
SB0	Cheetah 3	1.8 VDC 1	1.72	Volts DC	7 sec	OK
IB6	Board 0	1.5 VDC 0	1.51	Volts DC	7 sec	OK
IB6	Board 0	3.3 VDC 0	3.35	Volts DC	7 sec	OK
IB6	Board 0	5 VDC 0	4.95	Volts DC	7 sec	OK
IB6	Board 0	12 VDC 0	12.11	Volts DC	7 sec	OK
IB6	Board 0	Temp. 0	28	Degrees C	7 sec	OK
IB6	Board 0	Temp. 1	23	Degrees C	7 sec	OK
IB6	SDC 0	Temp. 0	49	Degrees C	7 sec	OK
IB6	AR 0	Temp. 0	45	Degrees C	7 sec	OK
IB6	SBBC 0	Temp. 0	41	Degrees C	7 sec	OK
IB6	IOASIC 0	Temp. 0	39	Degrees C	7 sec	OK
IB6	IOASIC 1	Temp. 1	34	Degrees C	7 sec	OK

showkeyswitch

Displays the keyswitch setting.

Scope

platform shell, domain shell

Syntax for the Platform Shell

```
showkeyswitch [-v] -d domainID
showkeyswitch -h
```

Syntax for the Domain Shell

```
showkeyswitch [-v]
showkeyswitch -h
```

Options/Parameters

–h displays help for this command.

–v is verbose mode.

-d *domainID* (platform shell only) is a, b, c, or d.

See Also

`setkeyswitch, showdomain, showplatform`

Description

Displays the keyswitch setting.

Examples

CODE EXAMPLE 3-57 showkeyswitch Example Showing the Keyswitch Set to On

```
schostname:A> showkeyswitch
keyswitch is: on
```

CODE EXAMPLE 3-58 showkeyswitch Example Showing the Keyswitch Set to Off

```
schostname:A> showkeyswitch
keyswitch is: off
```

CODE EXAMPLE 3-59 showkeyswitch Example Showing the Keyswitch Set to Standby

```
schostname:A> showkeyswitch
keyswitch is: standby
```

showlogs

Displays the system controller logged events stored in the system controller message buffer.

Scope

platform shell, domain shell

Syntax for the Platform Shell

```
showlogs [-d domain ID] [-v]
showlogs -h
```

Syntax for the Domain Shell

```
showlogs [-v]
showlogs -h
```

Options/Parameters

-h displays help for this command.

-d *domainID* (platform shell only) specifies a domain (a, b, c, or d).

-v is verbose.

Description

Displays the messages that are stored in the message buffer. Each domain and the platform have separate and independent buffers. Once a buffer is filled, the old messages are overwritten. These are messages from the system controller (not the Solaris operating environment) that are logged to the consoles. This is a small buffer and it is cleared when you reboot the system controller or when it loses power.

There is a separate log for the platform and for each domain. If the loghost has been configured (the platform and each domain are configured separately—using setupplatform for the platform loghost and setupdomain for each domain

loghost), then the messages will also be logged to the loghost for storage. Note that storing the messages is only temporary and will not survive a system reboot or a loss in power.

See Also

`setupdomain, setupplatform, showdomain, showplatform`

Example

CODE EXAMPLE 3-60 displays the output of the `showlogs` command, run at the platform shell after rebooting the system.

CODE EXAMPLE 3-60   Sample Output of the `showlogs` Command Run After Rebooting the System

```
schostname:SC> showlogs
May 09 13:36:21 sp4-sc0 Platform.SC: [ID 506094 local0.notice] Boot: ScApp
5.12.5, RTOS 17

May 09 13:36:37 sp4-sc0 Platform.SC: [ID 595276 local0.notice] Clock Source:
75MHz
May 09 13:36:50 sp4-sc0 Platform.SC: [ID 223289 local0.notice] Master System
Controller

May 09 13:38:30 sp4-sc0 Platform.SC: [ID 588369 local0.notice] Starting SNMP
agent.
```

showplatform

Displays the configuration and information for the platform and domain status.

Scope

platform shell

Syntax

showplatform [-v] [-d *domainID*] [-p *part*]

showplatform [-p *part*]

showplatform -h

Options/Parameters

-h displays help for this command.

-d *domainID* specifies a domain to show (a, b, c, or d).

-p *part domainID* displays information on a specific part. Use the options in this list when you specify a domain with the -d option. These options are the same as the -p *part* options to showdomain.

where:

- acls shows the Access Control List (ACL).
- bootparams shows boot parameter information.
- loghosts shows the log hosts for the domain.
- mac shows MAC addresses for the domain.
- snmp shows SNMP configuration for the domain.
- status shows domain status.

-p *part* displays information on a specific part. Use the options to -p *part* when you *do not* specify a domain with the -d option.

where:

- acls shows the Access Control List (ACL).
- frame shows FrameManager information.
- log shows log hosts for the platform.
- mac shows the mac addresses, hostID, and the platform a serial number for the domain and the system controllers.
- network displays the network settings.
- partition shows the partition mode.

- ■ `sc` shows the system controller configuration.
- ■ `snmp` shows snmp information for the platform.
- ■ `status` shows domain status for all domains.

`-v` is verbose mode.

Description

This command displays all configuration variables for this platform, including:

- ■ Network attributes of the platform
- ■ ACLs for each domain
- ■ Domain status
- ■ SNMP status
- ■ Loghosts
- ■ Partition configuration
- ■ Other values that are used but are not configurable such as the MAC address, host ID, and system serial number

See Also

`setupplatform`, `showdate`

Example

CODE EXAMPLE 3-61   `showplatform` Output for a Sun Fire 6800 System

```
schostname:SC> showplatform

Domain  Solaris Nodename  Domain Status  Keyswitch
------  ----------------  -------------  ---------
A       -                 Powered Off    off
B       -                 Powered Off    standby
C       -                 Powered Off    off
D       -                 Powered Off    off

The system controller is configured to be on a network.
Network settings: DHCP
Hostname: schostname
IP Address: xxx.xxx.xxx.xx
Netmask: xxx.xxx.xxx.x
Gateway: xxx.xxx.xxx.xxx
```

```
DNS Domain: domainname
Primary DNS Server: XXX.XXX.X.XXX
Secondary DNS Server: XXX.XXX.X.XXX

Loghost for Platform:
Log Facility for Platform:local0

SNMP Agent: enabled
Chassis Description: Sun Fire 6800
Chassis Contact:
Chassis Location:
Trap Hosts:
Public Community String: P-public
Private Community String: P-private

ACL for Domain A: SB0 SB1 SB2 SB3 SB4 SB5 IB6 IB7 IB8 IB9
ACL for Domain B: SB0 SB1 SB2 SB3 SB4 SB5 IB6 IB7 IB8 IB9
ACL for Domain C: SB0 SB1 SB2 SB3 SB4 SB5 IB6 IB7 IB8 IB9
ACL for Domain D: SB0 SB1 SB2 SB3 SB4 SB5 IB6 IB7 IB8 IB9

SC POST diag Level: off

Chassis is in dual partition mode.

schostname:SC>
```

CODE EXAMPLE 3-62 shows the `showplatform -v` command output.

CODE EXAMPLE 3-62 `showplatform -v` Output for a Sun Fire 6800 System

```
schostname:SC> showplatform -v

Domain  Solaris Nodename  Domain Status  Keyswitch
------  ----------------  -------------  ---------
A       -                 Powered Off    off
B       -                 Powered Off    off
C       -                 Powered Off    off
D       -                 Powered Off    off

Network
-------
The system controller is configured to be on a network.
Network settings: DHCP
Hostname: schostname
IP Address: XXX.XXX.XXX.XX
```

```
Netmask: XXX.XXX.XXX.X
Gateway: XXX.XXX.XXX.XXX
DNS Domain: domainname
Primary DNS Server: XXX.XXX.X.XXX
Secondary DNS Server: XXX.XXX.X.XXX


                          MAC Address          HostID
                          -----------          ------
Domain A                  XX:XX:XX:XX:XX:XX     XXXXXXX
Domain B                  XX:XX:XX:XX:XX:XX     XXXXXXX
Domain C                  XX:XX:XX:XX:XX:XX     XXXXXXX
Domain D                  XX:XX:XX:XX:XX:XX     XXXXXXX
SSC0                      XX:XX:XX:XX:XX:XX     XXXXXXX
SSC1                      XX:XX:XX:XX:XX:XX     XXXXXXX


System Serial Number: XXXXXXX

Loghosts
--------
Loghost for Platform:
Log Facility for Platform: local0

SNMP
----
SNMP Agent: enabled
Chassis Description: Sun Fire 6800
Chassis Contact:
Chassis Location:
Trap Hosts:
Public Community String: P-public
Private Community String: P-private

SNMP packets received: X
SNMP packets sent: XXX
SNMP traps sent: XXX

ACLs
----
ACL for Domain A: SB0 SB1 SB2 SB3 SB4 SB5 IB6 IB7 IB8 IB9
ACL for Domain B: SB0 SB1 SB2 SB3 SB4 SB5 IB6 IB7 IB8 IB9
ACL for Domain C: SB0 SB1 SB2 SB3 SB4 SB5 IB6 IB7 IB8 IB9
ACL for Domain D: SB0 SB1 SB2 SB3 SB4 SB5 IB6 IB7 IB8 IB9

SC POST
-----
SC POST diag Level: min
```

`showplatform -v` Output for a Sun Fire 6800 System *(Continued)*

```
Partition Mode
--------------
Chassis is in dual partition mode.
Frame information is not available.

schostname:SC>
```

In CODE EXAMPLE 3-61 and CODE EXAMPLE 3-62, the Domain Status header has the following possible states:

- Powered Off
- Standby
- Running POST
- Active
- Active - OpenBoot PROM
- Active - Booting
- Active - Solaris
- Active - Halted
- Active - Reset
- Active - Panicking
- Active - Debugger
- Not Responding
- Paused due to an error

For a description of these states, see "Domain Status" on page 7.

showresetstate

Shows that state of CPUs after a reset or a RED MODE trap.

Scope

domain shell

Syntax

```
showresetstate [-w | -s | -v]
```

Options/Parameters

`-w` displays the contents of register windows. The first windows displayed in the output are the stack trace for the failing CPU. The windows that are not part of the stack trace follow in the same order.

`-s` displays the contents of the secondary save area. Normally, the command displays the contents of the primary save area that holds the registers from the time of the first failure. If more than one failure occurs on a CPU, the secondary save area holds the selected registers from the time of the last failure.

`-v` displays a summary of all available registers. Includes the registers available with:

- `-s` and `-w` options
- Default display (no options)

Description

Prints a summary report of the contents of registers from the save areas of every CPU in the domain that has a valid saved state. By specifying certain options, the format and content of the printed report can be controlled.

With no options, the command displays the following registers:

- Register window (globals, outs, locals, ins) that was current at the time of the failure.
- `CANSAVE`, `CANRESTORE`, `OTHERWIN`, and other selected privileged registers.
- `AFAR` and `AFSR`.
- `TICK` and some related CPU-specific registers.

Since a CPU that does not have a saved state does not produce output, there is no output when CPUs are reset or have not failed. After a failure, the saved state is available until the domain restarts the OpenBoot PROM, either by rebooting the Solaris operating environment or by turning the virtual keyswitch from off to on.

See Also

`reset`

showsc

Shows the system controller version and uptime.

Scope

platform shell

Syntax

```
showsc [-v]

showsc -h
```

Options/Parameters

-h displays help for this command.

-v is verbose mode.

Description

Shows the system controller version and uptime.

Example

CODE EXAMPLE 3-63   showsc Command

```
schostname:SC> showsc

SC: SSC0

SC date: Wed May 09 15:06:22 PDT 2001
SC uptime: 1 hour 30 minutes 11 seconds

ScApp version: 5.12.5
RTOS version: 17
```

RTOS is the Real Time Operating System for the system controller and ScApp is the system controller application.

testboard

Tests the CPU/Memory board.

Scope

platform shell, domain shell

Syntax

```
testboard board_name
```

```
testboard -h
```

Options/Parameters

-h displays help for this command.

board_name is sb0-sb5 for CPU/Memory boards.

Description

Tests the CPU/Memory board name specified on the command line. This board cannot be used in an active domain and must be powered on. If running from a domain, the board must be assigned to the domain. The Repeater boards required to run the domain must also be powered on.

The `testboard` command runs tests at the levels set in the domain. Testing a CPU/Memory board is most accurately done when it is part of a domain. It is best if the CPU/Memory board can be tested by POST. However, if that is not possible, testing in the domain is the next best test.

Caution – When testing a CPU/Memory board that is assigned to a domain and the domain is active, there is a very slight possibility that problems can occur with the active domain. To prevent this from being a problem, unassign the board from the domain and test the board from the platform shell.

See Also

showcomponent, showboards

Example

To test CPU/Memory board, sb0 in the domain A shell (CODE EXAMPLE 3-64), type:

CODE EXAMPLE 3-64   testboard Example Testing CPU/Memory Board sb0

```
schostname:A> testboard sb0
<Command output not shown.>
```

Glossary

domain shell   With the domain shell, you have access to system controller commands that you need to perform on a domain. There are up to four domain shells (A through D). The domain shell prompt is *schostname*:A> (or B>, C>, or D>).

domain   A set of one or more system boards that is capable of booting the operating system and running independently of other domains. Domains do not depend on each other and do not interact with each other.

domain console   If the Solaris operating environment or the OpenBoot PROM is running, you can access the domain console. With the domain console, you can have the ok, login, #, or % prompts.

environmental monitoring   All systems have a large number of sensors that monitor temperature, voltage, and current. The system controller polls devices in a timely manner and makes the environmental data available. The system controller will shut down various components to prevent damage.

keyswitch   See virtual domain keyswitch.

partition   A group of Repeater boards that are used together to provide communications between CPU/Memory boards and I/O assemblies in the same domain.

platform shell   The platform shell enables access to the entire system and provides: configuration control, environmental status, ability to rearrange domains, ability to power on and off power grids, ability to change the system controller password, and other generic system controller functions.

POST   Power-on self-test. This is the program that takes uninitialized system hardware and probes and tests its components, configures what seems worthwhile into a coherent initialized system, and hands it off to the OpenBoot PROM.

SRAM   Static Random Access Memory. A type of high-speed memory device used for Ecache modules.

system controller
The system controller consists of the System Controller board and the system controller software. The system controller software manages platform and domain resources, monitors and controls the platform and domains, configures domains and the platform, provides access to domain consoles, provides the date and time to the Solaris operating environment, provides the reference clock signal used throughout the system, provides console security, performs domain initialization, provides a mechanism for updating the firmware on the boards installed in the system, and provides an external management interface using SNMP.

System Controller board
A board containing a CPU (central processing unit), which oversees operation of the system and provides clocks and the console bus. These systems support two System Controller boards.

TOD
See virtual time of day (TOD).

virtual domain keyswitch
The system controller provides a virtual keyswitch for each domain. The `setkeyswitch` command controls the position of the virtual keyswitch for each domain.

virtual time of day (TOD)
The TOD/NVRAM chip is located on the System Controller board. The system controller provides TOD services for each domain and for the system controller.

Index