SGI® L1 and L2 Controller
Software User's Guide

CONTRIBUTORS

Written by Linda Rae Sande

Revised by Francisco Razo and Terry Schultz

Illustrated by Dan Young

Production by Glen Traefald

Engineering contributions by Don Adams, Michael T. Brown, Dick Brownell, Jason Chang, Steve Hein, Jill Heitpas, Nancy Heller, Matt Hoy, Hao Pham, Craig Schultz, and Lisa Steinmetz.

New Features in This Guide

This manual has been updated with information to support the SGI Origin 3900 and SGI Altix 3000 series of servers.

New Features Documented

This release adds information about the following:

- L1 targeting commands "iia, iib, iic, and iid" on page 52
- Show voltages command "margin" on page 61
- Enable or disable LED command "fuel" on page 66
- L1 targeting commands "nia and nib" on page 67
- JTAG boundary scan hard reset or soft reset command "scan" on page 78

Major Documentation Changes

The following sections were revised for this release:

- "Console Selection" on page 12
- "Viewing System Configuration" on page 7
- "L1 Controller Commands" on page 34
- "Identifying Bricks" on page 23

Record of Revision

Version	Description
001	May 2002 Initial revision.
002	February 2003 Updated to support the SGI Origin 3900 and SGI Altix 3000 series of servers.

Contents

Figures

Tables

Examples

About This Guide

This guide describes how to use the L1 and L2 controller commands at your system console to monitor and manage the following systems:

- SGI Origin 3000 series of servers

- SGI Origin 300 series of servers

- SGI Origin 300 systems with NUMAlink

- SGI Onyx 3000 series of graphics systems

- SGI Onyx 300 series of graphics systems

- SGI Altix 3000 series of servers

The following topics are covered in this guide:

- Chapter 1, "Introducing the L1 and L2 Controllers," describes the function of the L1 and L2 controllers.

- Chapter 2, "Using the L1 and L2 Controllers," describes how to use the L1 and L2 controller to monitor and manage the systems.

- Chapter 3, "L1 and L2 Controller Commands," describes the L1 and L2 system controller commands.

Related Publications

The following publications contain additional information that may be helpful:

- S*GI Origin 3000 Series Owner's Guide* introduces the Origin 3000 servers and describes how to operate, manage, modify some aspects of, and maintain the Origin 3000 server.

> **Note:** The L1 and L2 controller command information contained in this user's guide supersedes the L1 and L2 controller command information in SGI Origin 3000 Series Owner's Guide and other owner's and user's guides.

- *SGI Origin 300 User's Guide* introduces the Origin 300 server and describes how to set up and operate the server.

- *SGI Origin 300 System with NUMAlink Module User's Guide* introduces the server and describes how to set up and operate the server.

- *SGI Onyx 3000 Series Graphics System Hardware User's Guide* introduces the Onyx 3000 graphics system and describes how to use and administer the graphics system.

- *SGI Onyx 300 Graphics System User's Guide* describes the Onyx 300 graphics system and describes how to set up and operate the graphics system.

- *SGI Altix 3000 User's Guide* provides an overview of the architecture and descriptions of the major components that make up the SGI Altix 3000 computer system. It also describes the standard procedures for powering up and powering down the system, basic troubleshooting information, and it includes important safety and regulatory specifications.

Obtaining Publications

You can obtain SGI documentation in the following ways:

- See the SGI Technical Publications Library at http://docs.sgi.com. Various formats are available. This library contains the most recent and most comprehensive set of online books, release notes, man pages, and other information.

- If it is installed on your SGI system, you can use InfoSearch, an online tool that provides a more limited set of online books, release notes, and man pages. With an IRIX system, select **Help** from the Toolchest, and then select **InfoSearch**. Or you can type `infosearch` on a command line.

> **Note:** InfoSearch is not available on Linux systems.

- You can also view release notes by typing either `grelnotes` or `relnotes` on a command line. (These commands are not available on Linux systems.

- You can also view man pages by typing `man <title>` on a command line.

Conventions

The following conventions are used throughout this publication:

Convention	Meaning
`command`	This fixed-space font denotes literal items such as commands, files, routines, path names, signals, messages, and programming language structures.
variable	Italic typeface denotes variable entries and words or concepts being defined.
`user input`	This bold, fixed-space font denotes literal items that the user enters in interactive sessions. (Output is shown in nonbold, fixed-space font.)
[]	Brackets enclose optional portions of a command or directive line.
...	Ellipses indicate that a preceding element can be repeated.
`manpage(x)`	Man page section identifiers appear in parentheses after man page names.
GUI element	This font denotes the names of graphical user interface (GUI) elements such as windows, screens, dialog boxes, menus, toolbars, icons, buttons, boxes, fields, and lists.

Reader Comments

If you have comments about the technical accuracy, content, or organization of this document, contact SGI. Be sure to include the title and document number of the manual with your comments. (Online, the document number is located in the front matter of the manual. In printed manuals, the document number is located at the bottom of each page.)

You can contact SGI in any of the following ways:

- Send e-mail to the following address:

  techpubs@sgi.com

- Use the Feedback option on the Technical Publications Library Web page:

  http://docs.sgi.com

- Contact your customer service representative and ask that an incident be filed in the SGI incident tracking system.

- Send mail to the following address:

  Technical Publications
  SGI
  1600 Amphitheatre Pkwy, M/S 535
  Mountain View, California 94043-1351

- Send a fax to the attention of "Technical Publications" at +1 650 932 0801.

SGI values your comments and will respond to them promptly.

Introducing the L1 and L2 Controllers

The L1 and L2 controllers are used to monitor and manage the following systems:

- SGI Origin 3000 series servers

- SGI Origin 300 servers

- SGI Origin 300 systems with NUMAlink

- SGI Onyx 3000 graphics systems

- SGI Onyx 300 graphics systems.

- SGI Altix 3000 series servers

This chapter describes the L1 and L2 controller functions in the following sections:

- "L1 Controller" on page 1

- "L2 Controller" on page 3

Note: Hardware descriptions for the L1 controller can be found in your server or graphics system user's guide, and hardware descriptions for the L2 controller can be found in the user's guides of those servers and graphics systems that support an L2 controller.

L1 Controller

The L1 controller is designed to manage and monitor the individual bricks in the Origin 3000 servers, Altix 3000 series servers, and Onyx 3000 graphics systems. It is also used to monitor the Origin 300 servers, Origin 300 systems with NUMAlink, and Onyx 300 graphics systems.

You monitor the system with the L1 controller either at the L1 controller's liquid crystal display (LCD) or at a system console.

See Chapter 2, "Using the L1 and L2 Controllers," for details on using the L1 controller, and see Chapter 3, "L1 and L2 Controller Commands," for a list of L1 controller commands.

The following are some of the functions performed by the L1 controller. (For a list of functions that are specific to a product or component, see your server or graphics system user's guide.

- Drives the L1 controller display.

- Reports all failure conditions.

- Monitors and controls LEDs.

- Monitors the power On/Off switch.

- Monitors the reset switch, and the nonmaskable interrupt (NMI) switch.

- Monitors and acts on the state of the power, reset, and NMI switches.

- Drives the reset, soft reset, and NMI signals to the local node electronics on command from software.

- Provides the time of day (TOD).

- Reads and reports memory and processor configuration within the node. Reads dual inline memory module (DIMM) and system identification (ID) PROMs.

- Reads and writes contents of module identification EEPROMs.

- Controls voltage regulator modules (VRMs).

- Monitors voltage and reports failures.

- Controls and sequences DC power output of voltage regulators. Monitors failures and reports them. Drives DC on and failure LEDs.

- Controls voltage margining within the brick or server.

- Controls and monitors fan speed and reports the number of the failed fan.

- Monitors and reports operating temperature and status of 48 VDC input power and DC power.

- Provides a USB hub chip that has six master ports: one port connects internally to the R-brick's L1 controller, four ports connect to the L1 controllers of four C-bricks (via the NUMAlink3 cable), and a master port connects to the L2 controller.

- Provides a high-speed serial channel to communicate between the system controller network and the hub ASIC.

- Provides a high-speed serial channel to communicate with the L1 controller as a second brick or server. This is implemented as an RS-485 ICMB interface in the NUMAflex cable.

- Controls Joint Test Action Group (JTAG) boundary scan.

- Logs information in Non-Volatile Random Access Memory (NVRAM).

- Provides a USB slave port to communicate with upstream system controllers.

- Provides an external high-speed serial console port.

- Provides the ability to flash the L1 firmware, which can be updated.

- Reports the population of the PCI cards and the power levels of the PCI slots.

- Powers on the PCI slots and their associated LEDs.

- Monitors the slot power for PCI boards (currently takes no action).

- Reports the power levels of the XIO slots.

- Controls the termination voltage margins of the XIO cards.

L2 Controller

The L2 controller is designed to monitor and manage the individual bricks in the Origin 3000 servers, Altix 3000 series servers, and Onyx 3000 graphics systems. It can also be used to monitor and manage the Origin 300 server systems, Origin 300 server systems with NUMAlink, and Onyx 300 graphics systems.

Note: The L2 controller monitoring and management functions are only available for systems that include a hardware L2 controller.

You monitor the system with the L2 controller at a system console. You can also enter L2 controller commands to manage your system at your system console.

See Chapter 2, "Using the L1 and L2 Controllers," for details on using the L2 controller, and see Chapter 3, "L1 and L2 Controller Commands," for a list of L2 controller commands you can use to manage and monitor the different systems.

The following are functions performed by the L2 controller. (For a list of functions specific to a product or component, see your server or graphics system user's guide.)

- Controls resource sharing.

- Controls the L1 controllers on the server or graphics system.

- Maintains controller configuration and topology information between the L1 and hardware L2 controllers.

- Routes data between upstream devices and downstream devices.

  Upstream devices provide control for the system, initiate commands for the downstream devices, and act on the messages that they receive from downstream devices.

  Downstream devices perform the actions that are specified by the L2 controller commands, send responses to the L2 controller that indicate the status of the commands, and send error messages to the L2 controller.

- Allows remote maintenance via a modem.

- Enables the entering of commands to toggle between L2, L1, and the console mode. This means you can power on your server with L1 or L2 controller commands, and monitor the power-on activity by changing to the console mode.

In a system with more than one hardware L2 controller, all hardware L2 controllers are peers and each propagates configuration information to the other L2 controllers. Each L2 controller monitors its associated L1 controllers and propagates this information to the other L2 controllers.

Note: For more details about the L2 controller hardware and using the L2 controller touch display to perform server functions such as powering on and powering off the server or graphics system, see your user's guide.

Using the L1 and L2 Controllers

This chapter describes how to use the L1 and L2 controllers to monitor and manage the SGI Origin 3000 series servers, SGI Origin 300 servers, SGI Origin 300 servers with NUMAlink, SGI Altix 3000 series servers, SGI Onyx 3000 graphics systems, and SGI Onyx 300 graphics systems in the following sections:

- "Monitoring Feedback and Entering Commands" on page 6
- "Operating the L2 Controller" on page 6
- "Operating the L1 Controller" on page 15
- "Upgrading L1 and L2 Firmware" on page 22
- "Identifying Bricks" on page 23
- "Status and Error Messages" on page 25

Note: All L2 controller commands can be used with all devices, however, you may find that some specific examples of using L2 controller commands in this chapter are not applicable to your device.

Note: Most, but not all, of the L1 controller commands can be used with all devices. You may also find that some specific examples using L1 controller commands in this chapter may show components not applicable to your device, but this does not indicate that these commands cannot be used to monitor and manage your device.

Monitoring Feedback and Entering Commands

You can monitor the L1 controller status and error messages on the L1 controller's liquid crystal display (LCD) located on the front panel of the individual bricks. The L1 controller and L2 controller status and error messages can also be monitored at your system console.

The L2 controller hardware includes L2 controller firmware. To access the L2 controller firmware, you must connect a system console, such as the SGIconsole or a dumb terminal, to the L2 controller. For instructions on connecting a console to the L2 controller, see your server or graphics system owner's guide.

The L2 firmware is always running as long as power is supplied to the L2 controller. If you connect a system console to the L2 controller's console port, the L2 prompt appears.

Note: See "Upgrading L1 and L2 Firmware" on page 22 for instructions to upgrade your L1 controller firmware and L2 controller firmware.

The system console allows you to monitor and manage your server or graphics system by entering L1 controller commands. You can also enter L2 controller commands to monitor and manage your system if your system has L2 controller hardware and a system console or if you are using an SGIconsole as your system console. See Chapter 3, "L1 and L2 Controller Commands," for a list of L1 and L2 controller commands you can use to monitor and manage the various devices.

Operating the L2 Controller

The L2 firmware operates in one of the following three modes, each of which is discussed in the sections that follow.

- **L2 mode**. The L2 prompt is visible and all input is directed to the L2 command processor.

- **Console mode from L2**. Output from the system is visible and all input is directed to the system.

- **L1 mode from L2**. The prompt from a single L1 is visible, and all input is directed to that L1 command processor.

L2 Mode

After a connection to the L2 controller, the following prompt appears, indicating that the L2 is ready to accept commands:

```
L2>
```

Common operations are discussed in the following sections:

- "Viewing System Configuration" on page 7
- "Setting Command Targeting" on page 8
- "Viewing Information, Warnings, and Error Messages" on page 11
- "Powering On, Powering Off, and Resetting the System" on page 11

Viewing System Configuration

You can use the L2 config command to view the current system configuration from a brick level, as follows:

```
L2> config
127.0.0.1:
127.0.0.1:0:0 - 003c01
127.0.0.1:0:1 - 004c01
127.0.0.1:0:2 - 002c01
127.0.0.1:0:3 - 001x01
L2>
```

As shown above, config produces a list of bricks in the system and the system controller address of each brick. This is similar to the output from using the config command on the L1 with the addition of the L2 IP address and USB port number. The structure of the brick's address is as follows:

```
a.b.c.d:x:y - rrrtss.p
```

where:

a.b.c.d	is the IP address of the L2. (In the example above, the IP address is 127.0.0.1.)
x	is the USB port number. (In the example above, the port number is zero.)
y	is the L1 index, as follows:
	0 - local brick (the brick to which the USB cable is attached)

1 - I/O brick attached to the local brick

3 - C-brick attached to the local brick

4 - I/O brick (attached to the C-brick) that is attached to the local brick

`rrr`	is the rack number
`t`	is the type of brick (C-brick, I-brick, and so on)
`ss`	is the slot number
`p`	is the partition (not present if the system is not partitioned).

A brick is identified by its rack and slot. In the example shown above, 003c01 is a C-brick in rack 3 and unit position 1.

Setting Command Targeting

If a command is not understood by the L2 system controller, in general it is passed on to the L1 system controllers. The destination determines which L1s receive the command. A destination is a range of racks and slots, specified as follows:

`rack` *<rack list>* `slot` *<slot list>*

The *<rack list>* specifies a list of racks. This can be a list delimited by commas, such that 2,4,7 specifies racks 2, 4, and 7. You can use a dash to specify a range of racks, such that 2-4 specifies racks 2, 3, and 4. Both nomenclatures can be combined, such that 2-4,7 specifies racks 2, 3, 4, and 7.

You can specify the *<slot list>* using the same nomenclature. The slot number, sometimes referred to as a bay number, is the unit position number located on the rack, slightly above where the bottom of the brick sits. Each rack unit position number is located toward the top of the two lines that mark the unit position that the number represents. For example, the rack numbering for a brick located in slot 10 would appear on the left front side of the rack, as shown in Figure 2-1:

```
 __
|  |
| 10
|  |
| [] |
|  |
| __ |(The bottom of the brick sits here.)
```
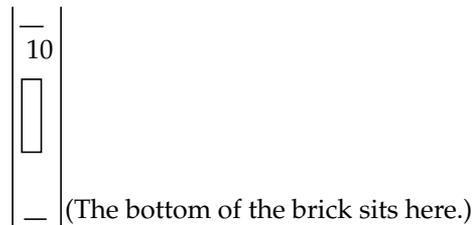
Figure 2-1   Rack Numbering

The slot *<slot list>* is optional; if not given, then all slots in the specified rack(s) are implied. You should avoid specifying a rack list and slot list that include multiple racks and slots, such as rack 2-4,7 slot 1-8,11,13. Generally, a rack and slot together are used to specify an individual brick.

You can use the aliases r and s to specify rack and slot, respectively. You can use the alias all or * in either or both the *<rack list>* and the *<slot list>* to specify all racks and all slots.

To send a command to all bricks in a partition, type the following:

l2> **partition** *<partition>* *<cmd>*

Default Destination

When the L2 starts, the default destination is set to all racks and all slots. You can determine the default destination by using the destination command, as follows:

```
L2> destination
all racks, all slots
L2>
```

The following command sets the destinations to rack 2 and 3, all slots:

```
L2> r 2,3 destination
2 default destination(s) set
L2>
```

The following example shows what bricks are found in the default destination. If you type a command not understood by the L2, the command is sent to these bricks.

Note: In the current implementation, adding a brick to either rack 2 or 3 would not automatically include it in the default destination. You would need to reset the default destination.

```
L2> destination
002c01 (127.0.0.1:0:2)
003c01 (127.0.0.1:0:0)
L2>
```

The following command resets the default destination to all racks and all slots:

```
L2> destination reset
default destination reset to all racks and slots
L2>
```

Current Destination

The current destination is a range of racks and slots for a given command. For example, the following command sends the command *<L1 command>* to all bricks in racks 2, 3, 4, and 7:

```
L2> r 2-4,7 <L1 command>
```

This is a one-time destination.

Command Interpretation

Some L2 commands are the same as the L1 commands. In many cases, this is intentional because the L2 provides sequencing that is necessary for a command to function correctly.

When L1 and L2 commands are similar, you can assure that an L1 command is entered for the bricks in the current destination by preceding *<L1 command>* with the L1 command (this is a one-time destination), as follows:

```
L2> r 2-4,7 l1 <L1 command>
```

Viewing Information, Warnings, and Error Messages

All information, warnings, and error messages generated by any of the system controllers are in the following form:

```
002c01 ERROR: invalid arguments for 'ver' command, try "help ver"
```

The general format includes a brick identification and the type of message, followed by the message. A message may be the result of an invalid command, as shown in the example, or the result of tasks running on the L1, such as the environmental monitor.

Each L1 has a log of local events. Use the L1 command `log` to view events on any of the L1s.

Powering On, Powering Off, and Resetting the System

The system can be powered on and off with the `power` command. This command is interpreted by the L2, because the bricks must be powered on in a specific order.

```
L2> power up
L2>
```

The `power` command may require several seconds to several minutes to complete. In the example above, all racks and slots in the default destination are affected. Any errors or warnings are reported as described in the prior "Viewing Information, Warnings, and Error Messages" section.

To power on or power off a specific brick, specify a current destination, as follows:

```
L2> r 2 s 5 power up
L2>
```

You can enter the `power down` and `reset` commands in a similar way, as follows:

```
L2> partition <partition number> <power down or reset>
```

Console Mode from L2

In console mode, all output from the system is visible and all input is directed to the system.

To enter console mode from L2 mode, press `Ctrl+D` at the L2 prompt and observe the response, as follows:

```
L2> Ctrl+D
entering console mode 002c01 console, <CTRL_T> to escape to L2
.
<system output appears here>
.
```

To return to L2 mode from console mode, press `Ctrl+T`, as follows:

```
Ctrl+T
escaping to L2 system controller
L2>
```

At this point, you can enter any L2 or L1 command. When the command completes, the L2 returns to console mode, as follows:

```
Re-entering console mode 002c01 console, <CTRL_T> to escape to L2
```

To permanently engage the L2 mode, press `Ctrl+T` and then type the `l2` command, as follows:

```
Ctrl+T
escaping to L2 system controller
L2> l2
L2 command processor engaged, <CTRL_D> for console mode.
L2>
```

Console Selection

The brick with which the L1 communicates in console mode is the system console or global master, and you can view and set it with the `select` command. By default, the C-brick attempts to communicate with its local CPUs when console mode is entered. If the system has been powered on and either one of the bricks received a request to be the system console, then the C-brick attempts to communicate with that brick. Enter the `select` command alone to show the current console mode settings as follows:

```
001c05-L1> select
console input: 001c05 console0
console output: not filtered.
```

The following are six common subchannels associated with console communications:

- Subchannel 0A specifies Node 0, CPU A.

- Subchannel 0C specifies Node 0, CPU B.

- Subchannel 1A specifies Node 1, CPU A.

- Subchannel 1C specifies Node 1, CPU B.

- Node 0 console subchannel.

- Node 1 console subchannel.

The output `console input: 001c05 console0` shows that the L2 will send console input to brick 001c05 and the subchannel to be used is the console0 subchannel.

To change system console status from one brick to the attached C-brick, use the `select <rack> <slot>` command:

```
001c05-L1> select r 2 s 1
console input: 001c05 console
console output: not filtered.
001c05-L1>
```

To change the subchannel used on the selected brick, use the `select` command followed by the subchannel number or the word `console`:

```
001c05-L1> select sub 0A
console input: 001c05 CPU 0A
console output: not filtered.
001c05-L1>
```

During the boot process on a multi-rack system, there is a window of time in which both C-bricks are producing output. The resulting output maybe somewhat jumbled at the L1. However, you can filter the console output so that the L1 shows output from only the brick chosen to receive console input. You can turn filtering on and off with the `select filter` command.

If you attempt to communicate with a brick that is not responding, a time-out condition results:
```
001c05-L1>

entering console mode 001c05 console, <CTRL-T> to escape to L1
no response from 001c05 junk bus console UART:UART_TIMEOUT
```

When this time-out condition occurs, either the brick is hung or the subchannel is incorrect.

L1 Mode from L2

In L1 mode, the prompt from a single L1 is visible, and all input is directed to that L1 command processor.

To enter L1 mode, type the l1 command and specify a rack and a slot, as follows:

```
L2> r 2 s 1 l1
enterling L1 mode 002c01, <CTRL-T> to escape to L2

002c01-L1>
```

To return to L2 mode, press Ctrl+T, as follows:

```
002c01-L1> Ctrl+T
escaping to L2 system controller, <CTRL-T> to send escape to L1
L2>
```

At this point, you can enter any L2 command. When the command completes execution, the L2 returns to L1 mode, as follows:

```
002c01-L1>
```

To permanently engage the L2 mode, press Ctrl+T and type the l2 command, as follows:

```
002c01-L1> Ctrl+T
escaping to L2 system controller, <CTRL-T> to send escape to L1
L2> l2
L2 command processor engaged, <CTRL-T> for console mode.
L2>
```

Note: If you press Ctrl+D while in L1 mode, the L1 goes into console mode. Output from the system console will not be visible because the L2 never shows system console output unless the L2 is in console mode. To return to the L1 prompt at this point, press Ctrl+T twice, followed by the L1 command, to lock the L1 back into L1 mode.

```
003c01> Ctrl+D

entering console mode 002c01 console, <CTRL-T> to escape to L1
Ctrl+T
escaping to L2 system controller, <CTRL-T> to send escape to L1
L2> Ctrl+T
escaping to L1 system controller
003c01-L1> l1
L1 command processor engaged, <CTRL-T> to exit.
003c01-L1>
```

Operating the L1 Controller

The L1 controller operates in one of the following two modes, each of which is discussed in the sections that follow:

- **L1 mode**. The L1 prompt is visible and all input is directed to the L1 command processor.

- **Console mode from L1 mode**. Output from the system is visible and all input is directed to the system.

Note: The console mode from L1 mode is not supported if the system contains an L2 controller.

L1 Mode

When you see a prompt of the following form, the L1 is ready to accept commands.

```
001c19-L1>
```

Common operations include the following and are discussed in the sections that follow:

- "Viewing System Configuration" on page 16
- "Command Targeting" on page 18
- "Viewing Information, Warnings, and Error Messages" on page 19
- "Powering On, Powering Off, and Resetting the Brick" on page 19

Viewing System Configuration

An L1 has limited knowledge of the system configuration. A C-brick only has information about its attached I/O brick and, if another C-brick is attached to it, information about that C-brick and its attached I/O brick. An I/O brick only has information about its attached C-brick. An R-brick only has information about itself.

You can view a brick's configuration information with the `config` command, as follows:

```
003c01-L1> config
:0 - 003c01
:1 - 004i01
:2 - 002c01
:3 - 001x01
003c01-L1>
```

Bricks are referenced by their racks and slot or bay locations. These values are stored in nonvolatile memory on the L1. Virtually all system controller communications require that each brick has a valid and unique rack and slot.

If a brick is not set with its rack and slot number, it appears in the output of an L2 `config` command, as shown in the following example:

```
L2> config
137.38.88.82.1.0 ---c-- (no rack/slot set)
L2>
```

To set the rack and slot for a brick, address it by its IP address, USB port, and L1 controller index. Note the following example:

```
L2> 137.38.88.82:1:0 brick rack 1
L2> 137.38.88.82:1:0 brick slot 8
L2> 137.38.88.82:1:0 reboot_l1
INFO: closed USB /dev/sgil1_0
INFO: opened USB /dev/sgil1_0
L2>config
137.38.88.82:1:0 001c08
L2.
```

The following example shows how to set rack 1, slot 8, for the C-brick with an IP address of 127.0.0.1:

```
L2> config
127.0.0.1:
127.0.0.1:0:0 - ---c--
127.0.0.1:0:0 - 001i01
127.0.0.1:0:0 - 001c05
L2> :0:0 brick rack 1
brick rack set to 001.
L2> :0:0 brick slot 8
brick slot set to 08.
L2> :0:0 reboot_l1
INFO: closed USB /dev/sgil1_0
INFO: opened USB /dev/sgil1_0
L2>
L2> config
127.0.0.1:
127.0.0.1:0:0 - 001c05
127.0.0.1:0:0 - 001i01
127.0.0.1:0:0 - 001c08
L2>
```

To set the rack and slot from the L1 prompt, simply use the `brick rack` and `brick slot` commands. To set the rack and slot on one of the attached bricks (an attached I/O brick, C-brick, or a C-brick's I/O brick), use the L1 targeting commands `nia`, `nib`, `iia`, or `iib` as follows:

```
001c05-L1> config
:0 - 001c05
:1 - ---i--
:5 - 001c08
:6 - 001p01
001c05-L1> iia brick rack 4
---i--:
brick rack set to 004.
001c05-l1> iia brick slot 1
---i--
brick slot set to 01
001c05-l1> iia reboot_l1
001c05 ERROR: no response from ---i--
001c05-L1> config
:0 - 001c05
:1 - 004i01
:5 - 001c08
:6 - 001p01
001c05-L1>
```

In the preceding code, the number after the colon symbol (:) indicates the following:

```
0 = local brick
1 = IIA
2 = IIB
5 = NIA
10 = NIB
```

To obtain a detailed configuration explanation from the L1 perspective, enter the following:

```
001c05-L1> config verbose
```

Command Targeting

All commands affect only the local brick, unless the command is prefixed with an asterisk (*). To target a command to all bricks (including the local brick), prefix the command, as shown in the following example:

```
003c01-L1> * version
003c01:
L1 0.7.37 (Image A), Built 05/24/2001 14:59:42 [P1 support]
004i01:
L1 0.7.37 (Image A), Built 05/24/2001 14:59:42 [P1 support]
002c01:
L1 0.7.37 (Image A), Built 05/24/2001 14:59:42 [P1 support]
001x01:
L1 0.7.37 (Image A), Built 05/24/2001 14:59:42 [P1 support]
003c01-L1>
```

You can also target commands to a single attached brick with either the cti, ctc, or ctci command, as follows:

```
003c01-L1> cti version
004i01:
L1 0.7.37 (Image A), Built 05/24/2001 14:59:42 [P1 support]
003c01-L1> ctc version
002c01:
L1 0.7.37 (Image A), Built 05/24/2001 14:59:42 [P1 support]
003c01-L1> ctci version
001x01:
L1 0.7.37 (Image A), Built 05/24/2001 14:59:42 [P1 support]
003c01-L1>
```

Viewing Information, Warnings, and Error Messages

All information, warnings, and error messages generated by any of the system controllers are in the following form:

```
002c01 ERROR: invalid arguments for 'ver' command, try "help ver"
```

The general format of the message includes a brick identification (this is not present if the command was to the local brick only), type of message, and the message. These messages can be the result of an invalid command (as shown in the example) or from tasks running on the L1, such as the environmental monitor.

Each L1 has a log of local events. Use the L1 command `log` to view the event on any of the L1s.

Powering On, Powering Off, and Resetting the Brick

You can power on and power off the brick with the `power` command, as follows:

```
003c01-L1> power up
003c01-L1>If an L2 is not present, you need to power on, power off, and
reset the system from one of the C-bricks. You do so by targeting all
bricks, as follows:
```

```
003c01-L1> * power up
003c01-L1>
```

This command can require from several seconds to several minutes to complete.

You can enter the `power off` and `reset` commands in similar ways.

Console Mode from L1

In console mode, output from the system is visible and all input is directed to the system.

To enter console mode, press `Ctrl+D` at the L1 prompt, as follows:

```
003c01-L1> Ctrl+D
entering console mode 003c01 console, <CTRL-T> to escape to L1
.
<system output appears here>
.
```

To return to L1 mode, press `Ctrl+T`, as follows:

```
Ctrl+T
escaping to L1 system controller
003c01-L1>
```

At this point, you can enter any L1 command. When the command completes execution, the L1 returns to console mode, as follows:

```
re-entering console mode 003c01 console, <CTRL-T> to escape to L1
```

To permanently engage the L1 mode, press `Ctrl+T` and then type the `l1` command, as follows:

```
Ctrl+T
escaping to L1 system controller
003c01-L1> l1
L1 command processor engaged, <CTRL-D> for console mode.
003c01-L1>
```

Console Selection

The brick with which the L1 communicates in console mode is the system console or global master, and you can view and set it with the `select` command. By default, the C-brick attempts to communicate with its local CPUs when it enters console mode. If the system has been powered on and either one of the bricks has a request to be the system console, then the C-brick attempts to communicate with that brick. Enter the `select` command alone to show the current console mode settings, as follows:

```
003c01-L1> select
console input: 003c01 console
console output: not filtered.
```

The following five common subchannels are associated with console communications:

- Subchannel 0 specifies CPU A.
- Subchannel 1 specifies CPU B.
- Subchannel 2 specifies CPU C.
- Subchannel 3 specifies CPU D.
- Subchannel 4 is the console subchannel.

The output `console input: 003c01 console` shows that the L1 will send console input to brick 003c01 and the console subchannel will be used.

To change system console status from one brick to the attached C-brick, use the `select` command, followed by `ctc` or the rack and slot number of the attached C-brick, as follows:

```
003c01-L1> select ctc
console input: 002c01 console
console output: not filtered.
003c01-L1> select r 2 s 1
console input: 002c01 console
console output: not filtered.
003c01-L1>
```

To change the subchannel used on the selected brick, use the `select` command, followed by the subchannel number or the word `console`, as follows:

```
003c01-L1> select 2
console input: 002c01 CPU C
console output: not filtered.
003c01-L1>
```

During the boot process on a multi-rack system, there is a window of time during which both C-bricks are producing output. This resulting output may be somewhat jumbled at the L1. However, you can filter the console output so that the L1 shows output only from the brick chosen to receive console input. You can turn filtering on and off with the `select filter` command.

If you try to communicate with a brick that is not responding, a time-out condition results, as follows:

```
003c01-L1>

entering console mode 002c01 console, <CTRL-T> to escape to L1
no response from 002c01 bedrock console UART:UART_TIMEOUT
```

When this time-out condition occurs, either the brick is hung or the subchannel is incorrect.

Upgrading L1 and L2 Firmware

L1 and L2 firmware is currently distributed as part of your software package. This collection of software packages contains L1 and L2 firmware.

The L1 and L2 firmware binary, and the utilities used to update it, are stored in `/usr/cpu/firmware/sysco`.

Upgrading L1 Firmware

The L1 firmware consists of the following three parts:

- Boot image
- Image A
- Image B

At boot time, the boot image validates images A and B and, if not instructed otherwise, it executes the newer of the two images. Because the L1 is running one of the two images, the image not in use is the image that will be overwritten when the firmware is upgraded. You need to reboot any L1 update either by power-cycling the brick or by using the L1 command `reboot_l1`. See the `flash` and `reboot_l1` commands in Chapter 3, "L1 and L2 Controller Commands," for descriptions of these commands.

Typically, you will upgrade the firmware through the network connection from the SGIconsole to the L2, as follows:

```
$> /usr/cpu/firmware/sysco/flashsc --l2 10.1.1.1
/usr/cpu/firmware/sysco/l1.bin all
```

This updates all the bricks in the system. You can update individual bricks by replacing *all* with a rack and slot number, as follows:

```
$> /usr/cpu/firmware/sysco/flashsc --l2 10.1.1.1
/usr/cpu/firmware/sysco/l1.bin 1.19
```

This updates only the brick in rack 1, slot 19.

Upgrading L2 Firmware

The L2 firmware consists of the following two parts:

- Boot image

- Kernel image

Typically, you will upgrade the firmware through the network connection from the SGIconsole to the L2, as follows:

```
$> /usr/cpu/firmware/sysco/flashsc --l2 10.1.1.1
/usr/cpu/firmware/sysco/l2.bin local
```

Once this command has executed, You must power-cycle the L2 to run the new image. To do this, you can use the L2 command `reboot_l2`.

If the L2 update fails, there is no back-up second image as there is with the L1. The L2, however, will not run the kernel image if it is not valid. At this point, the L2 is intelligent enough for you to upgrade it through its console port, as follows:

```
$> /usr/cpu/firmware/sysco/flashsc --l2recover /usr/cpu/firmware/sysco
/l2.bin <device>
```

where *<device>* equals --dev or --serial with the appropriate argument for the option entered.

Output will indicate that the firmware image is being erased and then rewritten. The flash image is quite large (almost 2 MB), so updating the flash takes several minutes. You must power-cycle the L2 to run the new image by using the L2 command `reboot_l2`.

Identifying Bricks

Bricks are referenced by their racks and slot or bay locations. These values are stored in non-volatile memory on the L1. Virtually all system controller communication requires that each brick have a valid and unique rack and slot. If a brick does not have these, the output of an L2 `config` command will reflect that as shown in the following example:

```
L2> config
137.38.88.82.1.0 ---c-- (no rack/slot set)
L2>
```

To set the rack and slot for a brick, address it by its IP address, USB port, and L1 controller index. The following is an example:

```
L2> 137.38.88.82:1:0 brick rack 3
L2> 137.38.88.82:1:0 brick slot 10
L2> 137.38.88.82:1:0 reboot_l1
L2> config
137.38.88.82:1:0 003c10
L2>
```

The following example shows how to set rack 3, slot 1, for the C-brick with the IP address 127.0.0.1:

```
L2> config
127.0.0.1:
127.0.0.1:0:0 - ---c--
127.0.0.1:0:0 - 004i01
127.0.0.1:0:0 - 002c01
127.0.0.1:0:0 - 001x01
L2> :0:0 brick rack 3
brick rack set to 003.
L2> :0:0 brick slot 1
brick slot set to 01.
L2> :0:0 reboot_l1
WARNING: can't read packet on L1 connection (/dev/sgil1_0), status:
IRouter:read failed - read error
INFO: closed USB /dev/sgil1_0
INFO: opened USB /dev/sgil1_0
WARNING: last error on L1 connection (/dev/sgil1_0) repeated 64 times

L2>
L2> config
127.0.0.1:
127.0.0.1:0:0 - 003c01
127.0.0.1:0:0 - 004i01
127.0.0.1:0:0 - 002c01
127.0.0.1:0:0 - 001x01
L2>
```

If the brick is connected to an L2 other than the local L2, you would enter the following:

L2>*<ipaddress>*:*<USB port>*:*<L1 index>* *<command>*

To set the rack and slot from the L1 prompt, simply use the `brick rack` and `brick slot` commands. To set the rack and slot on one of the attached bricks (an attached I/O brick, C-brick, or a C-brick's I/O brick), use the L1 targeting commands `cti`, `ctc`, or `ctci`. See the following example.

```
003c01-L1> config
:0 - 003c01
:1 - ---i--
:2 - 002c01
:3 - 001x01
003c01-L1> cti brick rack 4
---i--:
brick rack set to 004.
003c01-l1> cti reboot_l1
003c01 ERROR: no response from ---i--
003c01-L1> config
:0 - 003c01
:1 - 004i01
:2 - 002c01
:3 - 001x01
003c01-L1>
```

Status and Error Messages

This section lists and describes the status and error messages generated by the L1 and L2 controllers. It also explains how to resolve the errors, if action is necessary.

L1 Controller Messages

The L1 controller front panel display, located on the front panel of individual bricks, consists of a 2-line, 12-character liquid crystal display (LCD) that provides the following:

- Brick identification

- System status

- Warning of required service or failure

- Identification of failed components

Note: Besides the L1 control display, if you have an L2 controller, you can see the L1 controller messages on the L2 controller touch display located on the front door of the leftmost compute rack (position 001). If you have a system console, you can also see the L1 controller messages on your system console.

Table 2-1 lists the L1 controller messages.

Note: Note that in Table 2-1, a voltage warning occurs when a supplied level of voltage is below or above the nominal (normal) voltage by 10 percent. A voltage fault occurs when a supplied level is below or above the nominal by 20 percent.

Table 2-1   L1 Controller Messages

L1 System Controller Message	Message Meaning and Action Needed
Internal voltage messages:	
ATTN: x.xV high fault limit reached @ x.xxV	30-second power off sequence for the brick (or system, if no backup is available), server, or module.
ATTN: x.xV low fault limit reached @ x.xxV	30-second power off sequence for the brick (or system, if no backup is available), server, or module.
ATTN: x.xV high warning limit reached @ x.xxV	A higher than nominal voltage condition is detected.
ATTN: x.xV low warning limit reached @ x.xxV	A lower than nominal voltage condition is detected.
ATTN: x.xV level stabilized @ x.xV	A monitored voltage level has returned to within acceptable limits.
Fan messages:	
ATTN: FAN # x fault limit reached @ xx RPM	A fan has reached its maximum RPM level. The ambient temperature may be too high. Check to see if a fan has failed.
ATTN: FAN # x warning limit reached @ xx RPM	A fan has increased its RPM level. Check the ambient temperature. Check to see if the fan stabilizes.
ATTN: FAN # x stabilized @ xx RPM	An increased fan RPM level has returned to normal.
Temperature messages: low alt.	
ATTN: TEMP # advisory temperature reached @ xxC xxF	The ambient temperature at the brick's, server's, or module's air inlet has exceeded 30 ˚C.
ATTN: TEMP # critical temperature reached @ xxC xxF	The ambient temperature at the brick's, server's, or module's air inlet has exceeded 35 ˚C.
ATTN: TEMP # fault temperature reached @ xxC xxF	The ambient temperature at the brick's or server's air inlet has exceeded 40 ˚C.

Table 2-1 L1 Controller Messages **(continued)**

L1 System Controller Message	Message Meaning and Action Needed
Temperature messages: high alt.	
ATTN: TEMP # advisory temperature reached @ xxC xxF	The ambient temperature at the brick's, server's, or module's air inlet has exceeded 27 ˚C.
ATTN: TEMP # critical temperature reached @ xxC xxF	The ambient temperature at the brick's, server's, or module's air inlet has exceeded 31 ˚C.
ATTN: TEMP # fault temperature reached @ xxC xxF	The ambient temperature at the brick's, server's, or module's air inlet has exceeded 35 ˚C.
Temperature stable message:	
ATTN: TEMP # stabilized @ xxC/xxF	The ambient temperature at the brick's, server's, or module's air inlet has returned to an acceptable level.
Power off messages:	
Auto power down in xx seconds	The L1 controller has registered a fault and is shutting down. The message displays every 5 seconds until shutdown.
Brick or server appears to have been powered down	The L1 controller has registered a fault and has shut down.

L2 Controller Messages

The L2 controller performs the following functions:

- Controls resource sharing.
- Controls L1 controllers.
- Resets the system.
- Issues non-maskable interrupts (NMIs).
- Displays voltage margin information.
- Routes data between upstream devices and downstream devices.

Upstream devices (for example, rack display, console, and modem) provide control for the system, initiate commands for the downstream devices, and act on the messages that they receive from downstream devices.

Downstream devices (for example, C-bricks, the USB hub of the R-brick, and L1 controllers of the bricks) perform the actions that are specified by the L2 controller commands, send responses to the L2 controller that indicate the status of the commands, and send error messages to the L2 controller.

- Allows remote maintenance.

You use the L2 controller touch display to do the following:

- Power the system on and off.

- Monitor voltage margins.

- Reset the system

- Enter a non-maskable interrupt (NMI).

The L2 controller also monitors and generates status and error messages related to the rack chassis items, such as the power bay and other rack items. The L2 controller also displays status and error messages generated by each individual brick's L1 controller. (See "L1 Controller Messages" on page 25 for L1 controller message descriptions.)

The L2 controller information is displayed on the L2 controller touch display located in the front door of your server system. (The actual L2 controller is located on the top of your rack enclosure.)

Note: If you have a system console, you can also see the L2 controller messages on the system console.

The L2 controller contains a software component that transfers data from a send client to the appropriate receive client. The clients with which the L2 controller communicates are local to the L2 controller.

The software allows the router clients to do the following:

- Register with the router (identifies the client with a unique ID).
- Register to receive messages from other clients (local or remote).
- Receive commands and send corresponding responses.
- Send commands and receive corresponding responses.
- Receive messages that they are registered to receive.

The L2 controller logs the following information in separate files:

- Messages and command responses from the L1 controllers (includes the I/O bricks).
- Messages and output from the system console.
- Debugging messages that the L2 controller produces.
- Commands and responses from the L2 controller touch display.
- Messages and output that are sent to the console (attached to the L2 controller).
- Messages and output that are sent to the modem port (attached to the L2 controller).

L1 and L2 Controller Commands

You can use L1 and L2 controller commands to monitor and manage the SGI Origin 3000 series servers, SGI Origin 300 servers, SGI Origin 300 servers with a NUMAlink module, SGI Altix 3000 series servers, SGI Onyx 3000 graphics systems, and SGI Onyx 300 graphics systems. This chapter consists of the following sections:

- "Commands for Different Devices" on page 31
- "Sending Commands to Specific Devices" on page 33
- "L1 Controller Commands" on page 34
- "L2 Controller Commands" on page 87

Note: Online command information is available. To view a list of L1 commands, enter `help` at the L1 prompt. To view a list of L2 commands, enter `help` at the L2 prompt. To view information about an individual command, enter `help` and the command name at the appropriate prompt.

Commands for Different Devices

This section explains which L1 and L2 controller commands can be used with which devices in the following sections:

- "L2 Commands" on page 32
- "L1 Commands" on page 32

L2 Commands

You can use all of the L2 controller commands to monitor and manage all of the devices (Origin 3000 series servers, SGI Altix 3000 series servers, Onyx 3000 graphics systems, Origin 300 servers, and Onyx 300 graphics systems) that include a hardware L2 controller. (You cannot use L2 controller commands with systems that do not include a hardware L2 controller.)

Note: Although some of the specific examples of using L2 controller commands in this chapter may show components not applicable to your device, all L2 controller commands in this chapter are applicable to all servers and graphics systems supported.

L1 Commands

You can use most, but not all, of the L1 controller commands to monitor and manage all devices (Origin 3000 series servers, SGI Altix 3000 series servers, Onyx 3000 graphics systems, Origin 300 servers, and Onyx 300 graphics systems). Each controller command description in "L1 Controller Commands" on page 34, gives the devices each command supports.

Note: If you try to use an L1 controller command that is not available for a particular system or brick component, a message will inform you of this.

Note: Although some specific examples of using the L1 controller commands in this chapter may show components not applicable to your particular device in the command description, the command will also work for your device if so indicated.

Sending Commands to Specific Devices

You can use the *<command>* variable to send a specific command to particular devices. (Use the `help` command to get the list of the many commands that you can enter with *<command>*.) The following are some of the commands in which you can use *<command>* as a variable:

- *<command>*

- *<ip>* : *<port>* : *<l1>* *<command>*

- *<rack>* . *<slot>* *<command>*

- : *<port>* : *<l1>* *<command>*

- *<ip>* *<command>*

- `rack` *<rng>* `slot` *<rng>* *<command>*

Use these commands as follows:

- Use *<command>* to send L1 or L2 commands to default destinations.

- Use *<ip>* : *<port>* : *<l1>* *<command>* to send an L1 command to a specified L1 controller, to a specified port identified by a specified IP address.

- Use *<rack>* . *<slot>* *<command>* to send a specified L1 controller command to an L1 controller located in a brick or server located in a specified slot of a specified rack.

- Use : *<port>* : *<l1>* *<command>* to send a specified L1 controller command to a specified port on a specified L1 controller located on the local L2 controller.

- Use *<ip>* *<command>* to send a specified L2 controller command to the L2 controller specified by the IP address.

- Use `rack` *<rng>* `slot` *<rng>* *<command>* to send a specified L1 or L2 controller command to selected racks and slots. The *<rng>* variable specifies the rack number(s) and slot number(s).

When entering these commands, be aware of the following rules:

- For the *<rng>* variable, you can enter a single number, two numbers separated by a hyphen to indicate a range, or numbers separated by commas to indicate separate items. For *<rng>*, you can also enter an asterisk or the word "all" to select all the items available.

- If no target is specified and *<command>* is an L2 controller command, it is executed on the local L2 controller only. If the L2 controller does not understand the command, it passes it on to all the L1 controllers in the default destination. (The `destination` command shows the default destinations.)

- If the target is specified and *<command>* is an L2 controller command, it is executed on the targeted L2 controllers. If the command is not understood by the L2 controller, it passes on the command to all L1 controllers on the target list.

- For overlapping L2 controller commands and L1 controller commands, precede the *<command>* with L1 to override the L2 interpretation.

L1 Controller Commands

The following subsections describe the L1 controller commands and command sets, which are listed alphabetically. Examples of output are included where applicable.

For a list of all L1 controller commands, enter the following:

```
T1-001c01-L1>help
Commands are:
*                  autopower|apwr    junkbus|jb|bedrock brick
partdb             cpu               nia|ni|ctc         nib
iia|ii|cti         iib               iic                iid
config|cfg         debug             display|dsp        env
fan                help|hlp          history|hist       l1dbg
link               log               ioport|ioprt       istat
l1                 leds              margin|mgn         network
pimm               port|prt          power|pwr          reset|rst
nmi                softreset|softrst select|sel          serial
eeprom             uart              usb                verbose
router|rtr         date              nvram              security
fuel          flash               reboot_l1         version|ver
pbay          test|tst             scan                pci
```

For more information on a single command, enter the following:

```
T1-001c01-L1>hlp <command>
```

* (asterisk character)

Use * *<command>* to broadcast the command specified. For systems with a C-brick, a command that is broadcasted is sent to all of the bricks that are attached to the C-brick that issued the command. For systems that do not have a C-brick, a command that is broadcasted is sent to all systems connected to the system that issued the command. Example 3-1 shows sample output from the * version command.

Example 3-1   * version Command Output

```
001c07-L1>* version
001c07:
L1 0.7.27 (Image A), Built 04/28/2000 13:06:43    [P1 support]
001i21:
L1 0.8.xx (Image B), Built 06/13/2000 09:54:32    [P1 support]
```

Used on a C-brick or on server bricks performing the compute function.

autopower

The following autopower command set enables, disables, and aborts the feature that enables the system to automatically power up if power is lost. The autopower command also displays the current autopower setting (see Example 3-2).

- autopower

- autopower on

- autopower off

- autopower abort

Example 3-2   autopower Command Output

```
001c20-L1>autopower
auto power on is disabled.
```

Used on a C-brick or on server bricks performing the compute function.

brick

The following brick command set provides the status of the brick and sets the brick location and type.

- `brick`
- `brick type` *<str>*
- `brick rack` *<rack number>*
- `brick slot` *<slot number>*
- `brick rackslot` *<rack number>* *<slot number>*
- `brick partition none`
- `brick partition` *<partition number>*

Note: For this command set, you can use `bay` or `upos` instead of `slot`, `rs` instead of `rackslot`, and `part` or `p` in the place of `partition`.

Use the `brick` command to determine the brick location and type. Example 3-3 shows sample output.

Example 3-3 `brick` Command Output

```
001c07-L1>brick
rack: 001 slot: 07 partition:0type: C source: EEPROM
```

Use `brick rack` *<rack number>* to set the rack number where the brick is located. Example 3-4 shows sample output.

Example 3-4 `brick rack` *<rack number>* Command Output

```
001c07-L1>brick rack 1
brick rack set to 001.
```

Use `brick slot` *<slot number>* to set the brick slot number in the rack. The variable *<slot number>* is a unit number from 01 to 39. Example 3-5 shows sample output.

Example 3-5   `brick slot` *<slot number>* Command Output

```
001c07-L1>brick slot 07
brick slot set to 07.
```

Use `brick rackslot` *<rack number>* *<slot number>* to set the brick rack and slot number. Example 3-6 shows sample output.

Example 3-6   `brick rackslot` *<rack number>* *<slot number>* Command Output

```
001c07-L1>brick rackslot 01 07
brick rack set to 01
brick slot set to 07
```

Use `brick partition none` to clear the brick partition number. Example 3-7 shows sample output.

Example 3-7   `brick partition none` Command Output

```
001c07-L1>brick partition none
brick partition cleared.
```

Use `brick partition` *<partition>* to set the brick partition number. Example 3-8 shows sample output.

Example 3-8   `brick partition` *<partition>* Command Output

```
001c07-L1>brick partition 1
brick partition set to 1.
```

Used on all bricks.

config

Use the following config command set to view and reset L1 controller configuration information, and to check network interface (NI) ring configuration.

- `config`
- `config verbose`
- `config reset`
- `config ring`

> **Note:** For this command set, you can use `cfg` instead of `config` and `v` instead of `verbose`,

Use the `config` command to view the L1 controller configuration information. Example 3-9 shows sample output from the `config` command for an Origin 300 server, Origin 300 server with NUMAlink, or Onyx 300 graphics system.

Example 3-9   `config` Command Output

```
001c20-L1>config
0: - 001c20
```

Example 3-10 shows sample output from the `config` command for an Origin 3000 server or an Onyx 3000 graphics system.

Example 3-10   `config` Command Output

```
003c01-L1>config
:0 - 003c01
:1 - 004i01
:2 - 002c01
:3 - 001x01
003c01-L1>
```

Bricks are referenced by their racks and slot or bay locations. These values are stored in non-volatile memory on the L1. Virtually all system controller communications require that each brick has a valid and unique rack and slot.

The number that follows the L1 index, after the dash, is the brick identification number (for example, 003c01). The first three digits of the brick identification indicate the rack in which the brick resides. The fourth digit indicates the type of brick (Table 3-1). The last two digits indicate the slot position in which the brick resides.

Table 3-1   Valid Brick Types

Type	Description
c	C-brick
i	I-brick
p	P-brick
r	R-brick
x	X-brick
n	N-brick
g	G-brick
?	Unknown brick type

Use the other commands in this set as follows:

- Use `config verbose` to view an expanded view of the L1 controller configuration information.

- Use `config reset` to reset the L1 controller configuration.

- Use `config ring` to check for network interface (NI) ring configuration.

Used on all bricks.

cpu

Use the following cpu command set to enable, disable, and provide the status of the central processing units in a C-brick or server bricks performing the compute function.

- `cpu`

- `cpu e`

- `cpu d`

- `cpu <exp> e`

- `cpu <exp> d`

Use the cpu command to view the status of all cpus. Example 3-11 shows a sample output from the cpu command on a C-brick:

Example 3-11   cpu Command Output

```
T1-038c01-L1>cpu

CPU Present Enabled
--- ------- -------
0    1       1
1    1       1
2    1       1
3    1       1
```

Use the other commands in this set as follows:

- Use cpu e to enable all cpus, and use the cpu d command to disable all cpus.

- Use cpu *<exp>* e to enable the number of the cpu entered.

- Use cpu *<exp>* d to disable the number of the cpu entered.

Used on a C-brick or on server bricks performing the compute function.

ctc

Use ctc *<command>* to send a command to a network interface (NI) port (port A). The NI port is found on devices such as a C-brick. Example 3-12 shows sample output from the ctc version command.

Example 3-12   ctc version Command Output

```
003c01-L1>ctc version
002c01:

L1 0.7.37 (Image A), Built 05/24/2000 14:59:42    [P1 support]
```

cti

Use cti *<command>* to send a command to an I/O interface (II) port (port A) found in I/O bricks (such as I-brick, P-brick, and so on) cabled to a C-brick. Example 3-13 shows sample output from the cti version command.

Example 3-13   cti version Command Output

```
003c01-L1>cti version
002c01:

L1 0.7.37 (Image A), Built 05/24/2000 14:59:42   [P1 support]
```

Used on a C-brick or on server bricks performing the compute function.

date

Use the following date command set to view and set the current date and time used by the L1 controller.

- date
- date *<str>*
- date tz
- date tz *<str>*

Use the date command to view the current date and time value used by the L1 controller. Example 3-14 shows sample output.

Example 3-14   date Command Output

```
001r19-L1>date
09/14/4655  22:47:07
```

Use date *<str>* to set the date and time value used by the L1 controller. The variable *<str>* is a time value in the form *yyyymmddHHMMSS* (where *yyyy* is the four-digit year, *mm* is a two-digit month, *dd* is a two-digit day, *HH* is a two-digit hour, *MM* is a two-digit minute, and *SS* is a two-digit second).

Use date tz to display the time zone offset used by the L1 controller. Use date tz *<str>* to set the time zone offset used by the L1 controller. The variable *<str>* is a

maximum of +12 (for 12 hours ahead of GMT) and a minimum of -12 (for 12 hours behind GMT).

Used on all bricks.

debug

The following debug command set provides the status of and sets the virtual debug switches. (In systems that include C-bricks, these commands are valid only for C-bricks.)

- `debug`

- `debug` *<switches>*

Use the `debug` command to determine the current settings for the virtual debug switches. Example 3-15 shows sample output.

Example 3-15   `debug` Command Output

```
001c07-L1>debug
debug switches set to 0x0000.
```

Use `debug` *<switches>* to set the virtual debug switches. The variable *<switches>* is a hexadecimal value for the switches. Example 3-16 shows sample output.

Example 3-16   `debug` *<switches>* Command Output
```
001c07-L1>debug 0x0001
debug switches set to 0x0001
```

Used on a C-brick or on server bricks performing the compute function.

display

The following display command set displays text on the front panel display and controls the LEDs on the front panel display.

- `display`

- `display` *<exp> <command>*

- `display power on`

- `display attention on`

- display failure on
- display power off
- display attention off
- display failure off

Note: For this command, you can use dsp instead of display, pwr instead of power, attn instead of attention, and fail for failure.

Use the display command to view the front panel display status. Example 3-17 shows sample output.

Example 3-17   display Command Output

```
001c07-L1>display
line 1: 001c10
line 2: powered up
```

Use the other commands in this set as follows:

- Use display *<exp> <command>* to write text on the L1 controller display (*<exp>* value is 1 or 2).
- Use display power on to turn on the power LED.
- Use display attention on to turn on the attention LED.
- Use display failure on to turn on the failure LED.
- Use display power off to turn off the power LED.
- Use display attention off to turn off the attention LED.
- Use display failure off to turn off the failure LED.

Used on all bricks.

eeprom

Use the following `eeprom` command set to view the eeprom data and to write eeprom data.

- `eeprom`

- `eeprom` *<eeprom>* *<offset>* *<length>*

- `eeprom fuel write default`

Use the commands in this set as follows:

- Use `eeprom` to view brick eeprom data. Example 3-18 shows sample output.

- Use `eeprom` *<eeprom>* *<offset>* *<length>* to show brick eeprom data for specified eeprom, specified offset, and specified length.

- Use `eeprom fuel write default` to write standard Fuel EEPROM data to MAC EEPROM.

Example 3-18   `eeprom` Command Output

```
001c07-L1>eeprom
PWR/LOG (UNKNOWN)
43 20 01 06 00 00 00 d9 00 02 17 c2 4e 41 c2 4e
41 c1 00 00 00 00 00 84 ff ff ff ff ff ff ff ff
ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff
00 09 00 3a 52 21 c9 43 45 4c 45 53 54 49 43 41
c4 49 50 33 35 c6 4b 48 4b 35 36 35 cc 30 33 30
5f 31 36 30 34 5f 30 30 31 00 c2 5f 46 01 02 c2
30 30 04 ff ff ff ff 04 ff ff ff ff 04 ff ff ff
ff c1 00 00 00 00 00 25 ff ff ff ff ff ff ff ff
PIMM 0 - no hardware detected
PIMM 1 - no hardware detected
DIMM 0    (SPD)
80 08 07 0c 0a 02 48 00 04 a0 80 02 80 08 08 01
0e 04 04 01 02 26 00 00 00 00 00 50 3c 50 30 40
b0 b0 60 60 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 51
ce 00 00 00 00 00 00 00 01 4b 4d 4d 33 20 34 36
4c 33 33 31 33 42 20 54 2d 47 30 20 42 05 00 14
16 02 09 ff ff ff ff ff ff ff ff ff ff ff ff ff
ff ff ff ff ff ff ff ff ff ff ff ff ff ff 64 ff
DIMM 2    - no hardware detected
DIMM 4    - no hardware detected
```

```
DIMM 6    - no hardware detected
DIMM 1    (SPD)
80 08 07 0c 0a 02 48 00 04 a0 80 02 80 08 08 01
0e 04 04 01 02 26 00 00 00 00 00 50 3c 50 30 40
b0 b0 60 60 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 51
ce 00 00 00 00 00 00 00 01 4b 4d 4d 33 20 34 36
4c 33 33 31 33 42 20 54 2d 47 30 20 42 05 00 14
19 02 09 ff ff ff ff ff ff ff ff ff ff ff ff ff
ff ff ff ff ff ff ff ff ff ff ff ff ff ff 64 ff
DIMM 3    - no hardware detected
DIMM 5    - no hardware detected
DIMM 7    - no hardware detected
```

Used on all bricks.

env

The following env command set provides the status of environmental conditions and turns environmental monitoring on and off.

- env
- env check
- env on|off
- env temp on|off
- env altitude high|low
- env reset
- env test warning
- env test fault
- env test tmpady
- env test tmpcrit

Note: For this set of commands, you can use rst in place of reset.

Use the commands in this set as follows:

- Use `env` to view the status of the voltage, fan speed, and internal temperature of a brick. The output of the `env` command is divided into four areas: environmental monitoring and auto start status, voltage status, fan status, and temperature status (see Example 3-19).

- Use `env check` to get a brief environmental status.

- Use `env on|off` to enable or disable environmental monitoring.

- Use `env temp on|off` to turn on or turn off temperature monitoring.

- Use `env altitude high|low` to enable a high altitude (greater than 5000 feet/1524 meters) or low-altitude (less than 5000 feet/1524 meters) temperature limit setting for the environmental monitoring. Example 3-20 shows sample output.

- Use `env reset` to reset all current warnings and faults that are set. Example 3-21 shows sample output.

Note: For this command, the mnemonic `rst` may replace the word `reset`.

- Use `env test warning` to run the environmental warning test.

- Use `env test fault` to run the environmental fault test.

- Use `env test tmpadv` to run the environmental advisory temperature test.

- Use `env test tmpcrit` to run the environmental critical temperature test.

Example 3-19 env Command Output

```
001c07-L1>env

Environmental monitoring is disabled.  Auto start at L1 boot is disabled.

Description     State      Warning Limits      Fault Limits        Current
--------------  ---------- ------------------- ------------------- -------
          DIMM  Enabled    10%   2.250/  2.750 20%   2.000/  3.000   2.509
          2.5V  Enabled    10%   2.250/  2.750 20%   2.000/  3.000   2.522
          1.5V  Enabled    10%   1.350/  1.650 20%   1.200/  1.800   1.509
      3.3V aux  Enabled    10%   2.970/  3.630 20%   2.640/  3.960   3.337
       3.3V L1  Enabled    10%   2.970/  3.630 20%   2.640/  3.960   3.302
        5V aux  Enabled    10%   4.500/  5.500 20%   4.000/  6.000   4.992
       12V bias Enabled    10%  10.800/ 13.200 20%   9.600/ 14.400  12.188
    PIMM0 SRAM  Enabled    10%   2.970/  3.630 20%   2.640/  3.960   3.276
     PIMM0 CPU  Enabled    10%   1.710/  2.090 20%   1.520/  2.280   1.875
```

```
    PIMM0 1.5V     Enabled  10%   1.350/  1.650  20%  1.200/  1.800    1.495
 PIMM0 3.3V aux    Enabled  10%   2.970/  3.630  20%  2.640/  3.960    3.320
   PIMM0 5V aux    Enabled  10%   4.500/  5.500  20%  4.000/  6.000    4.914
 PIMM0 12V bias    Enabled  10%  10.800/ 13.200  20%  9.600/ 14.400   12.125
     PIMM1 SRAM   Disabled  10%   2.970/  3.630  20%  2.640/  3.960   12.125
      PIMM1 CPU   Disabled  10%   1.710/  2.090  20%  1.520/  2.280   12.125
     PIMM1 1.5V   Disabled  10%   1.350/  1.650  20%  1.200/  1.800   12.125
 PIMM1 3.3V aux   Disabled  10%   2.970/  3.630  20%  2.640/  3.960   12.125
   PIMM1 5V aux   Disabled  10%   4.500/  5.500  20%  4.000/  6.000   12.125
 PIMM1 12V bias   Disabled  10%  10.800/ 13.200  20%  9.600/ 14.400   12.125

Description     State     Warning RPM   Current RPM
-------------   ---------  --------------  -------------
FAN 0           Enabled       2160           2421
FAN 1           Enabled       2160           2421
FAN 2           Enabled       2160           2404
Advisory   Critical  Fault     Current
Description     State     Temp      Temp      Temp      Temp
-------------  ---------- --------  --------  --------  ---------
        TEMP 3   Enabled  30C/ 86F  35C/ 95F  40C/104F  26c/ 78F
        TEMP 5  Disabled  30C/ 86F  35C/ 95F  40C/104F   0c/ 32F
```

The environmental monitoring and auto start status area has two statements. The first statement describes the current state of environmental monitoring. The environmental monitoring state is one of the following:

- "disabled - initialization error"
- "disabled - configuration error"
- "disabled - start error"
- "disabled"
- "disabled - unknown error (0x00)"
- "running"

The second statement describes the current state of the auto start. The auto start state is "enabled" or "disabled."

The voltage status contains five columns as follows:

- The Description column provides the name of the supply voltage (2.5 V, 3.3 V, 5 V, etc.).

- The State column provides the current state of the supply voltage (Enabled, Fault, Warning, Waiting, or Disabled).

- The Warning Limits column provides the range of voltages that, when exceeded, causes a supply voltage to be in a Warning state.

- The Fault Limits column provides the range of voltages that, when exceeded, causes a supply voltage to be in a Fault state.

- The Current column provides the current value of the supply voltage.

The fan status contains four columns, as follows:

- The Description column provides the name of each fan.

- The State column provides the current state of the fan (Warning, Enabled, Waiting, or Disabled).

- The Warning RPM column provides the lowest revolutions-per-minute allowed before a fan enters a Warning state.

- The Current RPM column provides the current value of the revolutions-per-minute for the fan.

The temperature status contains six columns as follows:

- The Description column provides the name of a temperature sensor.

- The State column provides the current state of the temperature sensor (Fault, Warning, Enabled, or Disabled).

- The Advisory Temp column provides the temperature that, when exceeded, causes a temperature advisory state.

- The Critical Temp column provides the temperature that, when exceeded, causes a critical temperature state.

- The Fault Temp column provides the temperature that, when exceeded, causes the temperature sensor to be in a Fault state.

- The Current Temp column provides the current temperature reading from the temperature sensor.

Example 3-20   `env altitude high` Command Output
```
001c07-L1>env altitude high
001c07-L1>
```

Example 3-21   `env reset` Command Output
```
001c07-L1>env reset
001c07-L1>
```

Used on all bricks.

fan

Use the `fan` command to determine whether the fans are on or off and to read the fan speeds. In the `fan` command output, the number in parentheses is the counter reading for the fan. The counter reading is a value provided by the fan tachometer. The system controller converts the counter reading into a revolutions-per-minute value.

Example 3-22 shows sample output from the `fan` command for an Origin 3000 server or Onyx 3000 graphics system.

Example 3-22   `fan` Command Output
```
001c07-L1>fan
fan(s) are on.
fan 0 rpm 2465 (339)
fan 1 rpm 2423 (352)
fan 2 rpm 2430 (349)
```

When the temperature of the brick is below 30 ˚C, the fans run at 2400 rpm. If a fan fails and the speed of the fan drops below 2100 rpm, the system controller increases the fan speed for the fans to 4400 rpm. If any two fans drop below 2100 rpm, the system controller shuts down the brick.

When the temperature of the brick is between 30 ˚C and 40 ˚C, the fans run at 3400 rpm. If a fan fails and the speed of the fan drops below 3100 rpm, the system controller shuts down the brick.

Used on all bricks.

flash

The following flash command set provides the status of and updates of the firmware images stored in flash memory.

- `flash status`
- `flash status verbose`
- `flash default <a|b>`
- `flash default current`
- `flash default new`
- `flash default old`
- `flash default reset`

Use `flash status` to view the status of the two firmware images stored in flash memory. Example 3-23 shows sample output. Each image has a checksum value that indicates whether an image is valid.

Example 3-23   `flash status` Command Output

```
001c07-L1>flash status
Flash image A currently booted

Image    Status      Revision     Built
-----    --------    ----------   -----
  A      default     0.7.27       04/28/2000 13:06:43
  B      valid       0.8.0        05/24/2000 10:50:23
```

Use the other commands in this set as follows:

- Use `flash status verbose` to display an expanded version of the flash status.
- Use `flash default <a|b>` to set firmware image A or firmware image B as the default image that the system controller uses when booting.
- Use `flash default current` to set the current image as the default flash image.
- Use `flash default new` to set the new image as the default flash image.
- Use `flash default old` to set the old image as the default flash image.

- Use `flash default reset` to set the firmware image with the latest time-stamp as the default image that the system controller uses when booting. If the selected firmware image is not valid, the flash default commands will return the following message: "cannot set default--image A (or B) is invalid!"

Used on all bricks.

help

The following help command set provides helpful information on the system controller commands.

- `help`
- `help <command>`

Use the `help` command to generate a list of all of the system controller commands. Use `help <command>` to display more information on a single command. The variable *<command>* is the name of a command.

Used on all bricks.

history

Use the `history` command to display a history of the system maintenance port (SMP) network connection.

In the `history` command output, the first number in the history length is the number of commands stored in the history array. The second number in the history length is the maximum number of commands that can be stored in the history array. Example 3-24 shows sample output.

Note: For this command, you can use `hist` in place of `history`.

Example 3-24   `history` Command Output

```
001c07-L1>history
History length: 3/20
2: fan speed 4
1: fan
0: env
```

Used on all bricks.

iia, iib, iic, and iid

Use the `iia` and `iib` targeting commands on a system with a C-brick with IP41 processors to send a command to the brick attached on the local II A port or local II B port, respectively. On SGI Origin 3900 systems, use the `iic` and `iid` targeting commands to send a command to the brick attached on the local II C port or local II D port, respectively. For more information, see "select" on page 81.

Used on C-bricks with IP41 processors and on SGI Origin 3900 systems.

ioport

The following ioport command set displays and sets the speeds and clock source of the I/O ports. This command set is valid only for C-, I-, N-, P-, PE-, X-, and V-bricks.

- `ioport`
- `ioport 400|600`
- `ioport a 400|600`
- `ioport b 400|600`
- `ioport clksrc a|b`

Note: For the ioport command set, the mnemonic `ioprt` may replace the word `ioport`. Use `ioport` to view the speed of the I/O ports in the C-, I-, N-, P-, PE-, X-, V-brick, or Origin 300 server and the clock source for the universal system time (UST). The I/O port speed is 400 MHz or 600 MHz. In the command output, the word Junkbus refers to the hub ASIC in the C-brick. The word XBridge refers to the crosstown ASIC in the I-, P-, PE-, X-, or V-brick. Example 3-25 and Example 3-26 show sample output.

Example 3-25   `ioport` Command Output for a C-Brick

```
001c07-L1>ioport
Junkbus I/O port speed: 600 MHz - current, attached I/O is detected
                        400 MHz - default, when no attached I/O
detected
```

Example 3-26   `ioport` Command Output for an X-Brick

```
001i21-L1>ioport
XBridge port A speed: 400 MHz - current, no attached C detected
XBridge port B speed: 600 MHz - current, from attached C
                      400 MHz - default, when no attached C detected
XBridge UST clock source: port A
```

Use `ioport 400|600` to set the speed of the I/O ports. You can set the I-, P-, PE-, X-, or V-bricks to 400 or 600 (400 MHz or 600 MHz). Example 3-27 shows sample output from the `ioport 600` command.

Example 3-27   `ioport 600` Command Output

```
001x004-L1>ioport 600
nvram parameter changed, reset required to affect change.
```

Use the other commands in this set as follows:

- Use `ioport a 400|600` to set the I/O port A to either 400 MHz or 600 MHz.

- Use `ioport b 400|600` to set the I/O port B to either 400 MHz or 600 MHz.

- Use `ioport clksrc a|b` to set the universal system time (UST) clock source in the I-, P-, PE-, X-, or V-brick to port A or port B. The options are A or B.

Example 3-28 shows sample output from the `ioport clksrc a` command. These commands are valid only for the I-, P-, PE-, X-, and V-bricks.

Example 3-28   `ioport clksrc a` Command Output

```
001x004-L1>ioport clksrc a
nvram parameter changed, reset required to affect change.
```

Used only on C-, I-, P-, PE-, X-, and V-bricks.

istat

Use the following istat command set to view the status of the memory, queues, tasks, and memory allocation.

- istat memory
- istat queues
- istat tasks
- istat pmalloc

Use istat memory to view the status of the L1 controller memory. Example 3-29 shows sample output.

Example 3-29   istat memory Command Output

```
001c07-L1>istat memory
SYSMEM  [0x30005cf8]    Size: 41656  Avail:  26792
SMLPOOL [0x30005cbc]    Size: 8704/ 272 Avail:     28
BIGPOOL [0x30005c80]    Size: 16640/1040 Avail:     15
```

Use istat queues to view the status of the L1 controller queues. Example 3-30 shows sample output.

Example 3-30   istat queues Command Output

```
001c07-L1>istat queues
CMD_REQQ [0x30008a54]   Size: 10   Avail: 10   Msgs:  0
CMD_RSPQ [0x30008a9c]   Size: 10   Avail: 10   Msgs:  0
SMP_RQUE [0x3000c8a0]   Size: 10   Avail: 10   Msgs:  0
SMP_WQUE [0x3000c8e8]   Size: 10   Avail: 10   Msgs:  0
SMP_IQUE [0x3000c930]   Size: 10   Avail: 10   Msgs:  0
CTI_WQUE [0x3000f38c]   Size: 10   Avail: 10   Msgs:  0
USB_WQUE [0x3000ee7c]   Size: 10   Avail: 10   Msgs:  0
SCAN_QUE [0x30002440]   Size: 20   Avail: 20   Msgs:  0
FLASH_Q  [0x30009c64]   Size:  5   Avail:  5   Msgs:  0
BDR_WQUE [0x3000a0c8]   Size: 10   Avail: 10   Msgs:  0
```

Use istat tasks to view the status of the L1 controller tasks. Example 3-31 shows sample output.

Example 3-31   istat tasks Command Output

```
001c07-L1>istat tasks
MAIN_TSK [0x300019f8] (SLEEP     )  STACK: 2048 @ 0x300011f8 ( 904 used, 1144 free)
USB_CNTL [0x30007f38] (DRV_SUSP  )  STACK: 1536 @ 0x30007fe0 ( 288 used, 1248 free)
CMD_ITSK [0x300096e4] (READY     )  STACK: 3072 @ 0x30008ae4 (1936 used, 1136 free)
SMP_RTSK [0x3000e1c8] (EVT_SUSP  )  STACK: 2048 @ 0x3000c9c8 ( 800 used, 1248 free)
SMP_ITSK [0x3000e318] (QUEUE_SUSP)  STACK: 2048 @ 0x3000d9c8 ( 768 used, 1280 free)
SMP_WTSK [0x3000e270] (QUEUE_SUSP)  STACK: 2048 @ 0x3000d1c8 ( 416 used, 1632 free)
ENV_PITK [0x3000a9ec] (SEM_SUSP  )  STACK: 1024 @ 0x3000addc ( 276 used,  748 free)
ENV_FITK [0x3000aa94] (SEM_SUSP  )  STACK: 1024 @ 0x3000b1dc ( 260 used,  764 free)
ENV_TITK [0x3000ab3c] (SEM_SUSP  )  STACK: 1024 @ 0x3000b5dc ( 264 used,  760 free)
ENV_PMTK [0x3000abe4] (SEM_SUSP  )  STACK: 1024 @ 0x3000b9dc ( 260 used,  764 free)
ENV_FMTK [0x3000ac8c] (SEM_SUSP  )  STACK: 1024 @ 0x3000bddc ( 268 used,  756 free)
ENV_TMTK [0x3000ad34] (SEM_SUSP  )  STACK: 1024 @ 0x3000c1dc ( 324 used,  700 free)
BDR_RTSK [0x30009f30] (DRV_SUSP  )  STACK: 2048 @ 0x30016524 (1204 used,  844 free)
BDR_WTSK [0x3000a020] (QUEUE_SUSP)  STACK: 1024 @ 0x30016d34 ( 532 used,  492 free)
CTI_RTSK [0x3000f23c] (DRV_SUSP  )  STACK: 2048 @ 0x3001717c ( 892 used, 1156 free)
CTI_WTSK [0x3000f2e4] (QUEUE_SUSP)  STACK: 2048 @ 0x3001798c ( 572 used, 1476 free)
USB_RTSK [0x3000edd4] (READY     )  STACK: 1800 @ 0x300181d4 ( 732 used, 1068 free)
USB_WTSK [0x3000eeec] (QUEUE_SUSP)  STACK: 1500 @ 0x300188ec ( 480 used, 1020 free)
SCAN_TSK [0x30002398] (QUEUE_SUSP)  STACK: 2048 @ 0x30001b98 ( 600 used, 1448 free)
I2C_HIGH [0x30007154] (SEM_SUSP  )  STACK: 1500 @ 0x300071fc ( 248 used, 1252 free)
I2C_LOW  [0x30006a2c] (SEM_SUSP  )  STACK: 1500 @ 0x30006ad4 ( 396 used, 1104 free)
FLASH_T  [0x30009bbc] (QUEUE_SUSP)  STACK: 1024 @ 0x300097bc ( 316 used,  708 free)
```

Use istat pmalloc to view the status of the memory allocation. Example 3-32 shows
sample output.

Example 3-32   istat pmalloc Command Output

```
001c07-L1>istat pmalloc

small pool size:    272
small pool avail:    28
small pool used:      3
small pool max:      25

big pool size:     1040
big pool avail:      15
big pool used:        0
big pool max:         3
pmalloc calls:        0
```

```
prealloc calls:      3

history:
```

Used on all bricks.

junkbus

The following junkbus command set provides the status of and sets the L1 controller-to-junkbus protocol setting. (For systems that include a C-brick, these commands are valid only for the C-brick.)

- `junkbus`
- `junkbus dump output`

Note: For the junkbus command set, you can use `jbus` instead of `junkbus`, `d` instead of `dump`, `o` for `output`, `i` for `input`, and `rst` for `reset`.

Use `junkbus` to determine the current settings for the L1-controller-to-junkbus protocol. Example 3-33 shows sample output for this command. PPP refers to point-to-point protocol. (Use the `l1dbg` command set to turn debugging on or off.)

Example 3-33   junkbus Command Output

```
001c003-L1>junkbus
junkbus protocol is PPP
```

Use `junkbus dump output` to dump the L1 controller-to-junkbus output capture.

Used on a C-brick or on server bricks performing the compute function, R-bricks, and the NUMAlink module. Note that if a C-brick is present, these commands are valid only for the C-brick.

l1

Use the `l1` command to engage the L1 controller command processor. Press Ctrl+D to disengage the command processor. Example 3-34 shows sample output.

Example 3-34   `l1` Command Output

```
001c003-L1>l1
L1 command processor engaged, <CTRL-D> to exit.
```

Used on all bricks.

l1dbg

The following l1dbg command set displays and sets the state of communication debugging features.

- `l1dbg`
- `l1dbg junkbus <on|off>`
- `l1dbg irtr <on|off>`
- `l1dbg env <exp>` (1=pwr, 2=fan, 4=tmp, 8=setup)
- `l1dbg env off`
- `l1dbg port <on|off>`
- `l1dbg i2c <on|off>`
- `l1dbg margin <on|off>`
- `l1dbg pppdump <on|off>`
- `l1dbg promreq <on|off>`
- `l1dbg qsusp <exp> <exp>`

Use `l1dbg` to display the state of the communication debugging features. Example 3-35 shows sample output.

Example 3-35   l1dbg Command Output

```
001c07-L1>l1dbg
L1 irouter debugging is off
L1 junkbus communication debugging is off
L1 environmental debugging is off
L1 port interrupt debugging is off
L1 i2c interrupt debugging is off
L1 voltage margin debugging is off
```

Use l1dbg junkbus <on|off> to turn on communication debugging for the L1 controller-to-junkbus-ASIC communication. Example 3-36 shows sample output from the l1dbg junkbus on command.

Example 3-36   l1dbg junkbus on Command Output

```
001c07-L1>l1dbg junkbus on
L1 irouter debugging is off
L1 junkbus communication debugging is on
L1 environmental debugging is off
L1 port interrupt debugging is off
L1 i2c interrupt debugging is off
L1 voltage margin debugging is off
```

The other l1dbg commands in this set function in the same manner as the l1dbg junkbus <on|off> command. Table 3-2 lists the debugging features that each command controls.

Table 3-2   l1dbg Commands

Command	Description
l1dbg junkbus <on|off>	Controls junkbus ASIC debugging.
l1dbg irtr <on|off>	Controls irouter debugging.
l1dbg env <exp>	Turns on environmental debugging. Options for <exp> are 1=power, 2=fan, 4=temperature, 8=setup.
l1dbg env off	Turns off environmental debugging.
l1dbg port <on|off>	Controls L1-port interrupt debugging.
l1dgb i2c <on|off>	Controls I²C interrupt debugging.
l1dbg margin <on|off>	Controls voltage margin debugging.

Table 3-2   l1dbg Commands   **(continued)**

Command	Description
`l1dbg pppdump <on|off>`	Controls data dump on junkbus ppp errors.
`l1dbg promreq <on|off>`	Controls PROM request debugging.
`l1dbg qsusp <exp> <exp>`	Sets queue suspend time req/rsp and evt.

Used on all bricks.

leds

Use the `leds` command to read the value of the group 0 and group 1 status LEDs. In systems that include a C-brick, this command is only valid for the C-brick. These LEDs connect to I/O expanders that monitor group 0 status LEDs 0 through 15 and group 1 status LEDs 0 through 15. In the output of the `leds` command, the CPUs correspond to the status LEDs as follows:

- CPU A corresponds to I/O expander A_0100001x - group 0 status LEDs 0 through 7.
- CPU B corresponds to I/O expander A_0100010x - group 0 status LEDs 8 through 15.
- CPU C corresponds to I/O expander A_0100011x - group 1 status LEDs 0 through 7.
- CPU D corresponds to I/O expander A_0100100x - group 1 status LEDs 8 through 15.

The number next to the CPU letter is a hexadecimal value that represents the value of the status LEDs that connect to the I/O expander (a bit set to 1 indicates the LED is on, a bit set to 0 indicates the LED is off). Example 3-37 shows sample output from the `leds` command. For systems with C-bricks containing IP41 processors, the output of the `leds` command matches the numbering of the CPUs.

Example 3-37   `leds` Command Output

```
001c07-L1>leds
CPU A: 0x02
CPU B: 0x02
CPU C: 0x0f
CPU D: 0x0f
```

Used on C-bricks or server bricks performing the compute function. Note that, if a C-brick is present, this command is only valid on the C-brick.

link

Use the `link` command to show the error status for the network interface (NI) and I/O interface (II) connector links.

Use the `link verbose` command to show the complete status for the NI and II connector links.

Note: For this command, you can use `v` in the place of `verbose`.

Used on all bricks.

log

The following log command set displays the contents of the log, clears or resets the log, and writes an entry into the log.

- `log`
- `log clear|reset`
- `log insert <entry>`

Use `log` to view the contents of the log. Example 3-38 shows sample output. If the log is empty, the output from the `log` command is "log is empty."

Example 3-38   `log` Command Output

```
001c07-L1>log
USB: registered as remote
USB-R: USB:device was reset
USB: unregistered
USB-R: IRouter:read failed - read error
USB: registered as remote
USB-R: USB:device was reset
USB: unregistered
USB-R: IRouter:read failed - read error
```

```
USB: registered as remote
SMP-R: UART:UART_NO_CONNECTION
L1 booting...
[L1-0] ALERT: eeprom.c line 367 ; eeprom 0 checksum error.
USB: registered as remote
L1 booting...
[L1-0] ALERT: eeprom.c line 367 ; eeprom 0 checksum error.
USB: registered as remote
CTI-R: UART:UART_BREAK_RECEIVED
CTI-R: IRouter:read failed - read error
USB-R: USB:device was reset
```

Use log clear or log reset to empty the log. Example 3-39 shows sample output.

Example 3-39   log reset Command Output

```
001c003-L1>log reset
log reset.
```

Use log insert *<entry>* to write a line in the log. The variable *<entry>* is text to enter into the log. Example 3-40 shows sample output from the log Start the Test command.

Example 3-40   log insert Start the Test Command Output

```
001c003-L1>log insert Start the Test
log entry made.
```

Used on all bricks.

margin

Use the margin command to show the margin status of all voltages.

*   margin|mgn

    All voltages margin status

*   margin|mgn default|low|norm|high

    Set all voltages to read-only memery (ROM) defaults

*   margin|mgn gbrick default|low|norm|high

    Set all G-brick voltages to ROM defaults

- `margin|mgn 1.2 default|low|norm|high`

  Set 1.2V to ROM default voltage

- `margin|mgn 1.2 <exp>`

  Set 1.2V margin, *exp* 0..255

- `margin|mgn 1.5 default|low|norm|high`

  Set 1.5V to ROM default voltage

- `margin|mgn 1.5 <exp>`

  Set 1.5V margin, *<exp>* 0..255

- `margin|mgn 1.8 default|low|norm|high`

  Set 1.8V to ROM default voltage

- `margin|mgn 1.8 <exp>`

  Set 1.8V margin, *<exp>* 0..255

- `margin|mgn 2.5 default|low|norm|high`

  Set 2.5V to ROM default voltage

- `margin|mgn 2.5 <exp>`

  Set 2.5V margin, *<exp>* 0..255

- `margin|mgn 3.3 default|low|norm|high`

  Set 3.3V to ROM default voltage

- `margin|mgn 3.3 <exp>`

  Set 3.3V margin, *<exp>* 0..255

- `margin|mgn 5 default|low|norm|high`

  Set 5V to ROM default voltage

- `margin|mgn 5 <exp>`

  Set 5V margin, *<exp>* 0..255

- `margin|mgn VTERM default|low|norm|high`

  Set VTERM to ROM default voltage

- `margin|mgn vid 1.5 default|low|norm|high`

  Set 1.5V to ROM default voltage

- `margin|mgn vid 1.5 <exp>`

  Set 1.5V margin, *<exp>* 0..31

- `margin|mgn vid 2.5 default|low|norm|high`

  Set 2.5V to ROM default voltage

- `margin|mgn VCPU default|low|norm|high`

  Set VCPU to ROM default voltage

- `margin|mgn VCPU <exp>`

  Set VCPU margin, *<exp>* 0..31

- `margin|mgn dimm default|low|norm|high`

  Set dual inline memory module (DIMM) to ROM default voltage

- `margin|mgn dimm <exp>`

  Set DIMM margin, *exp* 0..255

- `margin|mgn dimm v|voltage str`

  Set DIMM margin based on specified target voltage *<exp>*

- `margin|mgn 1.5p0 default|low|norm|high`

  Set processor inline memory module 0 (PIMM0) 1.5V to ROM default voltage

- `margin|mgn 1.5p0 <exp>`

  Set PIMM0 1.5V margin, *<exp>* 0..255

- `margin|mgn 1.5p1 default|low|norm|high`

  Set PIMM1 1.5V to ROM default voltage

- `margin|mgn 1.5p1 <exp>`

  Set PIMM1 1.5V margin, *<exp>* 0..255

- `margin|mgn p0sram default|low|norm|high`

  Set PIMM0 SRAM to ROM default voltage

- `margin|mgn p0sram <exp>`

  Set PIMM0 SRAM margin, *<exp>*0..255

- `margin|mgn p1sram default|low|norm|high`

  Set PIMM1 SRAM to ROM default voltage

- `margin|mgn p1sram <exp>`

  Set PIMM1 SRAM margin, *<exp>* 0..255

- `margin|mgn p0cpu default|low|norm|high`

  Set PIMM0 CPU to ROM default voltage

- `margin|mgn p0cpu <exp>`

  Set PIMM0 CPU margin, *<exp>* 0..255

- `margin|mgn p1cpu default|low|norm|high`

  Set PIMM1 CPU to ROM default voltage

- `margin|mgn p1cpu <exp>`

  Set PIMM1 CPU margin, *<exp>* 0..255

- `margin|mgn cpu default|low|norm|high`

  Set Fuel CPU to ROM default voltage

- `margin|mgn cpu <exp>`

  Set Fuel CPU margin, *exp* 0..255

- `margin|mgn sram default|low|norm|high`

  Set Fuel static random-access memory (SRAM) to ROM default voltage

- `margin|mgn cpu <exp>`

  Set Fuel CPU margin, *<exp>* 0..255

- `margin|mgn sram default|low|norm|high`

  Set Fuel SRAM to ROM default voltage

- `margin|mgn sram default|low|norm|high`

  Set Fuel SRAM voltage

- margin|mgn 2.5xio default|low|norm|high

  Set Fuel XIO voltage

Used on all bricks.

network

The following network command set displays and sets the mode of the network communication interface. (For systems that include a C-brick, this command is valid only for C-bricks.)

- `network`

- `network usb`

- `network 422`

- `network autodetect|auto on`

- `network autodetect|auto off`

Use `network` to view the current mode of the network communication interface. Example 3-41 shows sample output.

Example 3-41   `network` Command Output

```
001c07-L1>network
network interface communication is USB
```

Use `network usb` to set the network communication interface mode to Universal Serial Bus (USB) and disable autodetection. Use `network 422` to set the network communication interface mode to RS-422 protocol and disable autodetection. Example 3-42 shows sample output from the `network usb` command.

Example 3-42   `network usb` Command Output

```
001c003-L1>network usb
nvram parameter changed, reset required to affect change.
```

Use `network autodetect|auto on` to turn on the autodetection. Use `network autodetect|auto off` to turn off the autodetection.

Used on C-brick or on server bricks performing the compute function and I/O bricks. Note that, if a C-brick is present, this command is valid only for the C-brick.

fuel

Use the `fuel` command to enable or disable the Silicon Graphics Fuel visual workstation LEDs for debugging purposes.

- `fuel led red on`

  Turn on red Fuel LED

- `fuel led red OFF`

  Turn off red Fuel LED

- `fuel led blue on`

  Turn on blue Fuel LED

- `fuel led blue off`

  Turn off blue Fuel LED

- `fuel led red blink start`

  Start blinking red Fuel LED

- `fuel led red blink stop`

  Stop blinking red Fuel LED

- `fuel led blue blink start`

  Start blinking blue Fuel LED

- `fuel led blue blink stop`

  Stop blinking blue Fuel LED

Used on the Silicon Graphics Fuel visual workstation for debugging purposes.

nia and nib

Use the `nia` and `nib` targeting commands to send a command to the brick attached on the local NI A port or NI B port, respectively. For more information, see "select" on page 81.

Used on C-bricks with IP41 processors and on SGI Origin 3900 systems.

nmi

Use the `nmi` command to issue a non-maskable interrupt (NMI). For systems that include a C-brick, the `nmi` command is valid only for the C-brick.

After the L1 system controller receives an `nmi` command, it displays "NMI..." on the front panel display and asserts the NMI signal in I/O expander B_0100001x. If no errors occur while the system controller issues the NMI, it displays "NMI done" on the front panel display (see Example 3-43 and Figure 3-1). If an error occurs, the system controller displays "NMI Fail" on the front panel display.

Example 3-43   `nmi` Command Output

```
001c07-L1>nmi
001c07-L1>
```


Figure 3-1   Example of `nmi` Front Panel Display Output

Used on a C-brick or on server bricks performing the compute function. Note that, if a C-brick is present, this command is valid only for the C-brick.

nvram reset

The `nvram reset` command returns the NVRAM settings of the L1 controller to the factory default settings.

Used on all bricks.

partdb

The following partdb command set displays, enables, and disables partitioning.

- `<partdb>`
- `<partdb enable>`
- `<partdb disable>`

Note: For this command set, you can use `e` or `on` instead of `enable` and `d` or `off` instead of `disable`.

Use `partdb` to display partition information from the database. Use `partdb enable` to enable the partition for a system, and use `partdb disable` to disable the partition for a system.

Used on C-brick or on server bricks performing the compute function and I/O bricks. Note that this command is valid only for the C-brick if a C-brick is present.

pbay

The following pbay command set displays information about the power bay, distributed power supplies (DPSs), DC output ports, and field replaceable units (FRUs):

- `pbay`
- `pbay version`
- `pbay dps`
- `pbay dps` *<dps number 1 - 6>*
- `pbay dcport`

- `pbay dcport` *<DC port number 1 - 8>*

- `pbay fru`

- `pbay fru` *<0 for power bay, 1 - 6 for DPS>*

- `pbay reset`

- `pbay env`

- `pbay env on|off`

- `pbay init`

- `pbay serial`

Use the `pbay` command to view the status of the power bay (see Example 3-44).

Example 3-44 `pbay` Command Output

```
001r28-L1>pbay
Total current: 16.5 Amps (+/-11%)
```

Use `pbay version` to view the firmware version of the power bay (see Example 3-45).

Example 3-45 `pbay version` Command Output

```
001r28-L1>pbay version
PBay FW Version 00.18, Built 09/29/2001 14:00:00
```

Use the other commands in this set as follows:

- Use `pbay dps` to view the status of all distributed power supplies (DPS).

- Use `pbay dps` *<powerbay number>* to view the status of an individual distributed power supply. The distributed power supplies are identified by a number ranging from 1 to 6.

- Use `pbay dcport` to view the status of all DC output ports.

- Use `pbay dcport` *<port number>* to view the status for an individual DC port. The ports are identified by a number ranging from 1 to 8.

- Use `pbay fru` to view information about all the power bays and DPS FRUs.

- Use `pbay fru` *<powerbay number>* to view information about the individual power bay FRUs, use the number 0. To view information about a distributed power supply FRU, use the numbers 1 through 6.

- Use `pbay reset` to reset the power bay and the power supplies.
- Use `pbay env` to view the status of environmental monitoring.
- Use `pbay env on|off` to enable or disable environmental monitoring.
- Use `pbay init` to initialize communication with the power bay.
- Use `pbay serial` to display the power bay ICMB card serial number.

Used on C-bricks, I/O-bricks, R-bricks, and graphic bricks.

pci

The following pci command set displays the status of the PCI cards in an I/O brick, an Origin 300 server, or a PCI expansion module, and powers up, powers down, and resets a PCI card.

- `pci`
- `pci <u|d>`
- `pci <bus> u|d`
- `pci <bus> <slot> u|d`
- `pci reset`
- `pci <bus> reset`
- `pci <bus> <slot> reset`

Note: For this command set, the mnemonic `rst` may replace the word `reset`.

Use the `pci` command to view the value of the status register for each PCI card. The output of the `pci` command contains eight columns, as follows:

- The Bus column lists the number of the bus for each PCI card.
- The Slot column lists the slot value for each PCI card.
- The Stat column lists the hexadecimal value of the status register for each PCI card.
- The Card Type column lists the card type (7.5 W, 15 W, 25 W, or none) for each slot.

- The Power column lists the value (error & off, error & on, okay & off, or okay & on) of the power OK and power on bits.

- The Attn LED column lists the value (off or on) of the attention LED for the PCI card.

- The Enable column lists the value (off or on) of the bus enable bit for the PCI card.

- The Reset column lists the value (off or on) of the reset bit for the PCI card.

Example 3-46 shows sample output from the `pci` command.

Example 3-46   `pci` Command Output

```
001i21-L1>pci
Bus Slot Stat Card Type Power        Attn LED Enable Reset
--- ---- ---- --------- ----------- -------- ------ -----
  1    1 0x91      15W  okay &  on      off      on   off
  1    2 0x57     none  okay & off      off     off    on
  1    3 0x91      15W  okay &  on      off      on   off
  1    4 0xff  on board  N/A & N/A      N/A     N/A   off
  2    1 0x57     none  okay & off      off     off    on
  2    2 0x57     none  okay & off      off     off    on
```

Use `pci <u|d>` to power up (u) or power down (d) all of the PCI cards in an I/O brick, Origin 300 server, or PCI expansion module. Example 3-47 shows sample output.

Example 3-47   `pci d` Command Output

```
001i21-L1>pci d
001i21-L1>
```

Use `pci <bus> <u|d>` to power up (u) or power down (d) all of the PCI cards on a bus in an I/O brick, Origin 300 server, or PCI expansion module. The variable *<bus>* is the bus number. Example 3-48 shows sample output from the `pci 1 u` command.

Example 3-48   `pci 1 u` Command Output

```
001i21-L1>pci 1 u
001i21-L1>
```

Use `pci <bus> <slot> <u|d>` to power up (u) or power down (d) an individual PCI card. The variable *<bus>* is the bus number and the variable *<slot>* is the slot number. Example 3-49 shows sample output from the `pci 1 2 u` command.

Example 3-49   `pci 1 2 u` Command Output

```
001i21-L1>pci 1 2 u
001i21-L1>
```

Use `pci reset` to reset all of the PCI cards in an I/O brick, Origin 300 server, or PCI expansion module. Example 3-50 shows sample output.

Example 3-50   `pci reset` Command Output

```
001i21-L1>pci reset
001i21-L1>
```

Use `pci <bus> reset` to reset all of the PCI cards on a bus in an I/O brick, Origin 300 server, or PCI expansion module. The variable *<bus>* is the bus number. Example 3-51 shows sample output from the `pci 1 reset` command.

Example 3-51   `pci 1 reset` Command Output

```
001i21-L1>pci 1 reset
001i21-L1>
```

Use `pci <bus> <slot> reset` to reset an individual PCI card. The variable *<bus>* is the bus number and the variable *<slot>* is the slot number. Example 3-52 shows sample output from the `pci 1 2 reset` command.

Example 3-52   `pci 1 2 reset` Command Output

```
001i21-L1>pci 1 2 reset
001i21-L1>
```

Used on I/O bricks, an Origin 300 server, or a PCI expansion module.

pimm

The following pimm command set displays the state of and sets the processor inline memory module (PIMM) clock source and clock mode. (This command set is valid only for C-bricks.)

- `pimm`
- `pimm clksrc <external|internal>`
- `pimm clkmode <async|sync>`

> **Note:** For the pimm command set, you can use the mnemonics `int` and `ext` instead of the words `internal` and `external`.

Use `pimm` to determine the PIMM clock source (internal or external) and the clock mode (synchronous or asynchronous). Example 3-53 shows sample output.

Example 3-53   pimm Command Output

```
001c07-L1>pimm
PIMM clock source is external
PIMM mode is asynchronous
```

Use `pimm clksrc <external|internal>` to set the PIMM clock source to external or internal. Example 3-54 shows sample output.

Example 3-54   pimm clksrc internal Command Output

```
001c07-L1>pimm clksrc internal
nvram parameter changed, reset required to affect change.
```

Use `pimm clkmode` *<async|sync>* to set the PIMM clock mode to asynchronous (async) or synchronous (sync). Example 3-55 shows sample output.

Example 3-55   pimm clkmode sync Command Output

```
001c07-L1>pimm clkmode sync
nvram parameter changed, reset required to affect change.
```

Used on C-bricks.

port

Use the `port` command to view the value of the status register for each port on a brick. As shown in Example 3-56 and Example 3-57, the output of the `port` command contains six columns, as follows:

- The Port column lists the name of each port.

- The Stat column lists the hexadecimal value of the status register for each port.

- The Remote Pwr column lists the value, "okay" or "none," of the remote power OK bit (bit 0) for each port.

- The Local Pwr column lists the value, "okay" or "none," of the local power OK bit (bit 1) for each port.

- The Link LED column lists the value, "on" or "off," of the link LED bit (bit 2) for each port.

- The SW LED column lists the value, "on" or "off," of the software LED bit (bit 3) for each port.

Note: For this command, the mnemonic `prt` may replace the word `port`.

Example 3-56   `port` Command Output for an R-brick

```
001r19-L1>port
Port Stat Remote Pwr Local Pwr  Link LED SW LED
---- ---- ---------- ---------- -------- --------
   A 0x02       none       okay      off      off
   B 0x0f       okay       okay       on       on
   C 0x0f       okay       okay       on       on
   D 0x02       none       okay      off      off
   E 0x02       none       okay      off      off
   F 0x0b       okay       okay      off       on
   G 0x0b       okay       okay      off       on
   H 0x02       none       okay      off      off
```

Example 3-57   `port` Command Output for a C-brick

```
001c07-L1>port
Port Stat Remote Pwr Local Pwr  Link LED SW LED
---- ---- ---------- ---------- -------- --------
   A 0x0f       okay       okay       on       on
   B 0x0f       okay       okay       on       on
```

Used on all bricks.

power

The following power command set displays the status of the supplies, and powers up and powers down the supplies.

- `power`

- `power check`

- power vrm

- power <up|down>

- power up hold

Note: For the power command set, the mnemonics pwr, u, and d may replace the words power, up, and down.

Use power to view the detailed current state of the power and margin values for the power supplies in a brick or system. Example 3-58 shows sample output.

Example 3-58   power Command Output

```
001c07-L1>power
Supply          State Voltage    Margin  Value
--------------  ----- ---------  ------- -----
          48V     on       N/A      N/A
         DIMM     on    2.509V  default   132
         2.5V     on    2.522V  default   120
         1.5V     on    1.509V  default   110
     3.3V aux     NC    3.337V      N/A
      3.3V L1     NC    3.302V      N/A
       5V aux     NC    4.992V      N/A
      12V bias    NC   12.188V      N/A
    PIMM0 SRAM    on    3.276V  default   125
     PIMM0 CPU    on    1.875V  default   142
    PIMM0 1.5V    NC    1.495V  default     3
PIMM0 3.3V aux    NC    3.320V      N/A
  PIMM0 5V aux    NC    4.914V      N/A
PIMM0 12V bias    NC   12.125V      N/A
    PIMM1 SRAM    <not present>
     PIMM1 CPU    <not present>
    PIMM1 1.5V    <not present>
PIMM1 3.3V aux    <not present>
  PIMM1 5V aux    <not present>
PIMM1 12V bias    <not present>
```

Use power check to view the summary of the current state of the power and margin values for the power supplies in a brick or system. Example 3-59 shows sample output.

Example 3-59   `power check` Command Output

```
001c07-L1>power check
power appears on
```

Use `power <up|down>` to power up or power down all of the power supplies in a brick or system. Example 3-60 shows sample output.

Example 3-60   `power up` Command Output

```
001c07-L1>power up
001c07-L1>
```

Used on all bricks.

reboot_l1

The following reboot_l1 command set reboots the L1 controller.

- `reboot_l1`
- `reboot_l1 <a|b>`
- `reboot_l1 <current|old|new|other>`
- `reboot_l1 addr <exp>`
- `reboot_l1 raw <exp>`

Use the other commands in this set as follows:

- Use `reboot_l1` to reboot the L1 controller using the newest firmware image (firmware image A or firmware image B).
- Use `reboot_l1 <a|b>` to reboot the L1 controller using the specified firmware image (A or B).
- Use `reboot_l1 <current|old|new|other>` to reboot the L1 controller using either the current, older, newer, or other (non-current) flash image.
- Use `reboot_l1 addr <exp>` to validate and boot the L1 controller flash image at the flash image address listed for *<exp>*.

- Use `reboot_l1 raw` *<exp>* to boot the L1 controller flash image at the flash image address listed for *<exp>*.

Used on all bricks.

reset

Use the `reset` command to perform a reset of the system. After the system controller receives a `reset` command, it sets various control and status signals back to their default values, and reboots the operating system. Example 3-61 shows sample output.

Example 3-61   `reset` Command Output

```
001c07-L1>reset
001c07-L1>
```

Used on all bricks.

router

Use the following router command set to view and set router types, to set routers in different modes for reasons such as service, and to enable router ports. This command set is valid only for R-bricks and NUMAlink modules:

- `router`
- `router meta`
- `router repeater`
- `router ordinary`
- `router 6 port`
- `router 8 port`
- `router 8 port` *<str> <str> <str> <str>*
- `router service` *<str> <str> <str> <str>*
- `router service off`
- `router spare`

Note: For the router command set, you can use `rtr` instead of `router`, `rep` instead of `repeater`, and `ord` instead of `ordinary`.

Use `router` to view the current router type. Example 3-62 shows sample output.

Example 3-62   `router` Command Output

```
001r19-L1>router
router type is ordinary
```

Use the other commands in this set as follows:

- Use `router meta` to set the router type to meta.

- Use `router repeater` to set the router type to repeater.

- Use `router ordinary` to set the router type to ordinary.

- Use `router 6 port` to enable all 6-port router ports.

- Use `router 8 port` to enable all 8-port router ports.

- Use `router 8 port <str> <str> <str> <str>` to enable all the 8-port router ports of the router specified with the IP address listed for the *<str> <str> <str> <str>* variable.

- Use `router service <str> <str> <str> <str>` to set into service mode, for repair, the 8-port router specified with the IP address listed for the *<str> <str> <str> <str>* variable.

- Use `router service off` to disable the service mode after the repairs are completed.

- Use `router spare` to configure the router for spare tear down.

Used on R-bricks and NUMAlink modules.

scan

Use the `scan` command to perform a JTAG boundary scan hard reset or soft reset.

- `scan reset`

  Perform hard and soft JTAG reset

- `scan reset hard`

  Perform hard JTAG reset (via TRSTN)

- `scan reset soft`

  Perform soft JTAG reset (via TMS)

- `scan reset both`

  Perform hard and soft JTAG reset

- `scan sel` *<exp>* *<exp>*

  Select SIC: *<addr> <CER>*

- `scan sel` *<exp> <exp> <exp>*

  Select SIC: <addr> <CER> <MR>

- `scan sel` *<exp> <exp> <exp> <exp>*

  Select SIC: *<addr> <CER> <MR> <IOR>*

- `scan ids` *<exp>*

  Read and display IDCODE registers

- `scan set trst 0`

  Direct control of JTAG TRST signal

- `scan set trst 1`

  Direct control of JTAG TRST signal

- `scan set tck 0`

  Direct control of JTAG TCK signal

- `scan set tck 1`

  Direct control of JTAG TCK signal

- `scan set tms 0`

  Direct control of JTAG TMS signal

- `scan set tms 1`

  Direct control of JTAG TMS signal

- `scan set tdi 0`

Direct control of JTAG TDI signal

- `scan set tdi 1`

  Direct control of JTAG TDI signal

- `scan set psi 0`

  Direct control of JTAG PSI signal

- `scan set psi 1`

  Direct control of JTAG PSI signal

- `scan get`

  Display state of JTAG TAP signals

- `scan debug <exp>`

  Set scan debug message level

- `scan debug`

  Display scan debug message level

- `scan info`

  Display scan information

- `scan count`

  Count length of IR and BYPASS registers

- `scan count <exp>`

  Count length of IR and BYPASS registers

Used on all bricks.

security verify

Use the `security verify <3400|3800>` command to test the security readiness of SGI Origin 3400 and SGI Origin 3800 servers. Use `security verify 3400` to verify the security readiness of the Origin 3400 servers. Use `security verify 3800` to test the security readiness of Origin 3800 servers.

Used on C-bricks, I/O-bricks, R-bricks, and graphic bricks.

select

The following select command set displays and sets the device to serve as console I/O. (These commands are valid only for C-bricks, or other devices with CPUs.)

- `select`
- `select` *<rack>* *<slot>*
- `select subchannel console`
- `select subchannel` *<exp>*
- `select subchannel <a|b|c|d>`
- `select subchannel` *<str>*
- `select filter <on|off>`

Note: For this command set, you can use `sel` in the place of `select`, `sub` or `s` in the place of `subchannel`, and `con` in the place of `console`.

Use the `iia` and `iib` commands on a system with a C-brick with IP41 processors to send a command to the brick attached on the local II A port or local II B port, respectively. On SGI Origin 3900 systems, use the `iic` and `iid` commands to send a command to the brick attached on the local II C port or local II D port, respectively.

Use the `nia` and `nib` commands to send a command to the brick attached on the local NI A port or NI B port, respectively.

Use the `select` command to view the current mode of console I/O. Example 3-63 shows sample output.

Example 3-63   select Command Output

```
001c07-L1>select
console input:  001c07 console
console output: not filtered.
```

Use the other commands in this set as follows:

- Use `select` *<rack>* *<slot>* to set the rack and slot location of the device to be console I/O.

- Use `select subchannel console` to set the current console subchannel as console I/O.

- Use `select subchannel` *<exp>* to set the mode of console I/O to the subchannel specified for *<exp>* (possible values for *<exp>* are 0, 1, 2, 3, or 4).

- Use `select subchannel <a|b|c|d>` to set the mode of console I/O to a CPU subchannel (possible CPU subchannel values are a, b, c or d).

- Use `select subchannel` *<str>* to set the mode of console I/O to a specified CPU subchannel (0, 1, 2, 3, or 4).

- Use `select filter <on|off>` to enable (on) or disable (off) the console output filter.

Used on C-bricks or on server bricks performing the compute function.

serial

Use the following serial command set to view and set the system serial number (SSN) that is stored in each brick.

- `serial`
- `serial verify`
- `serial all`
- `serial dimm`
- `serial clear`
- `serial` *<str>* *<str>* *<str>* *<str>*
- `serial security on`

Use the commands in this set as follows:

- Use `serial` to view the secure system serial number (SSN) settings stored in the NVRAM.

- Use `serial verify` to test the brick's readiness for secure serial numbering.

- Use `serial all` to show the brick serial number (BSN) and the SSN settings in NVRAM.

- Use serial dimm to show the dual-inline memory module (DIMM) part and serial number.

- Use serial clear to clear the SSN.

- Use serial *<str> <str> <str> <str>* to erase and reassign the device's SSN. The variable *<str> <str> <str> <str>* is the value of a security key that is provided only to SGI employees.

- Use serial security on to enable the SSN security.

Used on all bricks.

softreset

Use the softreset command to issue a software reset. In systems that include a C-brick, the softreset command is valid only for the C-brick.

After receiving a softreset command, the system controller displays "Sft rst..." on the front panel display and asserts the soft reset signal in I/O expander B_0100001x. If no errors occur while the system controller issues the soft reset, it displays "Sft rst done" on the front panel display (see Example 3-64 and Figure 3-2). If an error occurs while the system controller issues the soft reset, it displays "Sft rst fail" on the front panel display.

Note: For this command, you can use the mnemonic softrst instead of the word softreset.

Example 3-64   softreset Command Output

```
001c07-L1>softreset
001c07-L1>
```
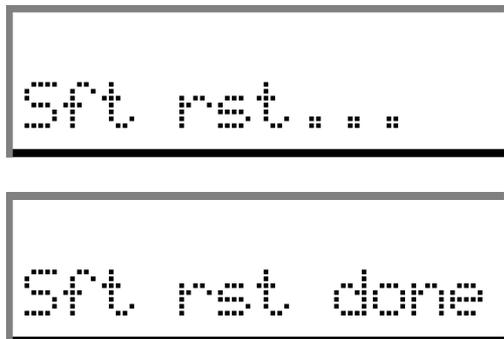
Figure 3-2   Example of `softreset` Command Output

Used on C-brick or on server bricks performing the compute function. Note that, if a C-brick is present, this command is valid only for the C-brick.

test

Use the following test command set to test various components in the L1 controller.

- `test i2c`
- `test i2c` *<passes>*
- `test ioexp set` *<index>* *<value>*
- `test ioexp get` *<index>*
- `test ioexp get all`
- `test intr`
- `test display`
- `test exception <mem|stack|wdog|fatal>`

Note: For this command set, you can use the mnemonic `tst` and `exc` in place of the words `test` and `exception`.

Use the commands in this set as follows:

- Use `test i2c` to test the I²C bus once.

- Use test i2c <*passes*> to test the I²C bus as many passes as you enter for the <*passes*> variable.

- Use test ioexp set <*index*> <*value*> to set a value for an I/O expander. The variable <*index*> is an I²C bus address and the variable <*value*> is the value for the I/O expander.

- Use test ioexp get <*index*> to read a value from an I/O expander. The variable <*index*> is an I²C bus address.

- Use test ioexp get all to read the values from all of the I/O expanders.

- Use test intr to view the I²C bus interrupt counts. Example 3-65 shows sample output.

- Use test display to reset the L1 controller liquid crystal display (LCD).

- Use test exception <mem|stack|wdog|fatal> to test the memory fault handling, stack corruption error handling, software watchdog reset, or fatal_error() call.

Example 3-65   test intr Command Output

```
001r19-L1>test intr
I2C interrupts, High: 25  Low: 406
DS1780  0, 1:    0 (0x00) 2:    0 (0x00)
IOExp  0:  197
IOExp  1:    0
IOExp  2:    0
IOExp  3:    0
IOExp  4:  137
IOExp  5:    0
IOExp  6:   11
IOExp  7:    0
```

Used on all bricks.

uart

The following uart command set displays the status of the following UARTs (universal asynchronous receivers/transmitters): CTI, CTC, SMP, and Junkbus. This command also enables you to reset the UART of the Junkbus ASIC.

- uart

- uart junkbus reset (You can use r in place of reset for this command)

Example 3-66 shows sample output from the uart command.

Example 3-66   uart Command Output

```
001c20-L1>uart
      Baud     Read   Read     Read     Read   Read   Write   Write   Write
UART  Rate     State  Status   Timeouts Breaks Errors State   Status  Timeouts
----  ----     -----  ------   -------- ------ ------ -----   ------  --------
CTI   107142   Discon Ready    0        0      0      Discon  Ready   0
CTC   107142   Discon Ready    0        0      0      Discon  Ready   0
SMP   37500    Discon Ready    0        0      1      Discon  Ready   0
BED   57692    Connect Suspend 0        0      13     Connect Ready   0
```

Used on all bricks.

usb

The usb command displays status information for the universal serial bus, (USB) L1 port. SSEs use this port to access L1 controllers in systems that do not have L2 controllers. Example 3-67 shows sample output.

Example 3-67   usb Command Output

```
001c20-L1>usb
Device: 0  Disconnects: 2  Bus Resets:  13
Endpoint State     Status     Stalls Errors Timeouts
-------- -----     ------     ------ ------ --------
Control  Active    Suspended 43      0      0
Read     Active    Ready     0       0      0
Write    Active    Ready     0       0      0
```

Used on all bricks.

verbose

Use the verbose command to get a system maintenance port (SMP) prompt and get into character-echo mode.

Used on all bricks.

version

Use the version command to view the version of the firmware that is currently running in the system controller. Example 3-68 shows sample output.

Example 3-68   version Command Output

```
001r19-L1>version
L1 0.7.27 (Image A), Built 04/28/2002 13:06:43    [P1 support]
```

Used on all bricks.

L2 Controller Commands

The following subsections describe the L2 controller commands and command sets, which are listed alphabetically. Examples of output are included where applicable.

autopower

The following autopower command set enables, disables, aborts, and shows the current auto power status.

- autopower
- autopower on
- autopower off
- autopower abort

Note: For these commands, the mnemonic apwr may replace the word autopower.

Use the `autopower` command to view the current auto power-up setting. Example 3-69 shows the sample output.

Example 3-69   `autopower` Command Output

```
L2>autopower
auto power up appears disabled
L2>
```

Use `autopower on` to enable the auto power on. Use `autopower off` to disable the auto power on. Use `autopower abort` to abort the auto power on.

config

The following config command set displays configuration information.

- `config`
- `config summary`
- `config verbose`
- `config l2`
- `config l2 verbose`
- `config rescan`
- `config auto`
- `config manual`
- `config print`
- `config print all`

Note: For these commands, the mnemonic `cfg` may replace the word `config`, and `s`, `v`, and `re` may replace the words `summary`, `verbose`, and `rescan`.

Use the `config` command to view configuration information for all the L1 controllers connected to all the L2 controllers in the server system (see Example 3-70). In the `config` command output, the first number is the IP address of the L2 controller (127.0.0.1 in the output below). The number that follows the IP address, after the first colon, is the USB port number (1 in the output below).

Example 3-70 `config` Command output

```
L2>config
127.0.0.1:
127.0.0.1:0:0 - 001r19
127.0.0.1:1:0 - 001c07
127.0.0.1:1:1 - 001i21
127.0.0.1:2:0 - 001c10
```

The number that follows the USB port number, after the second colon, is the L1 index.

Bricks are referenced by their racks and slot or bay locations. These values are stored in non-volatile memory on the L1. Virtually all system controller communications require that each brick have a valid and unique rack and slot For information how how to determine the L1 index value for a brick, see "Viewing System Configuration" on page 16.

The number that follows the L1 index, after the dash, is the brick identification number (for example, 001c07). The first three digits of the brick identification number indicate the rack in which the brick resides. The fourth digit indicates the type of brick (see Table 3-3). The last two digits indicate the slot position in which the brick resides.

Table 3-3 Valid Brick Types

Type	Description
c	C-brick
i	I-brick
p	P-brick
r	R-brick
x	X-brick
?	Unknown brick type

Use the other commands in this set as follows:

- Use `config summary` to see summary configuration information.

- Use `config verbose` to see more detailed configuration information for all the L1 controllers connected to all the server's L2 controllers.

- Use `config l2` to see configuration information for all the L1 controllers connected to the local L2 controller.

- Use `config l2 verbose` to see more detailed L2 controller configuration information.

- Use `config rescan` to force an L1 controller configuration update.

- Use `config auto` to enable automatic L1 controller configuration updates.

- Use `config manual` to disable automatic L1 controller configuration updates.

- Use `config print` to print out the local configuration for debugging purposes.

- Use `config print all` to print out a full configuration for debugging purposes.

date

Use the following date command set to view and set the current date and time used by the L2 controller.

- `date`

- `date <str>`

- `date tz`

- `date tz <str>`

Use the `date` command to view the current date and time value used by the L2 controller. Example 3-71 shows sample output.

Example 3-71   `date` Command Output

```
L2>date
09/14/4655  22:47:07
```

Use `date <str>` to set the date and time value used by the L2 controller. The variable *<str>* is a time value in the form *yyyymmddHHMMSS* (where *yyyy* is the four-digit year, *mm* is a two-digit month, *dd* is a two-digit day, *HH* is a two-digit hour, *MM* is a two-digit minute, and *SS* is a two-digit second).

Use `date tz` to display the time zone offset used by the L2 controller. Use `date tz <str>` to set the time zone offset used by the L2 controller. The variable *<str>* is a maximum of +12 (for 12 hours ahead of GMT) and a minimum of -12 (for 12 hours behind GMT).

destination

The following destination command set displays the brick identification numbers of bricks that are the destinations of L2 commands, and sets the bricks that are the destinations for L2 commands.

- `destination`
- `rack <rng> slot <rng> destination`
- `destination reset`

Note: For these commands, the mnemonic `dest` may replace the word `destination`. Also, the mnemonics `r` and `s` may replace the words `rack` and `slot`.

Use the `destination` command to view the brick identification numbers of the bricks that are the destinations of L2 commands. Example 3-72 shows sample output.

Example 3-72 `destination` Command Output

```
L2>destination
all racks, all slots
```

Use `rack <rng> slot <rng> destination` to set individual bricks that are the destinations for L2 commands. The variable *<rng>* specifies a rack number(s) and slot number(s).

Note: For the *<rng>* variable, you can enter a single number, or two numbers separated by a hyphen to indicate a range, or numbers separated by commas to indicate separate items. For *<rng>*, you can also enter an asterisk or the word "all" to select all the items available.

Example 3-73 `rack <rng> slot <rng> destination` Command Output

```
L2>r 1 s 7 destination
1 default destination(s) set
L2> destination
001c07 (127.0.0.1:1:0)
```

Use destination reset to reset the destination of L2 commands to all bricks in all racks and all slots. Example 3-74 shows sample output.

Note: For this command, the mnemonic rst may replace the word reset.

Example 3-74   destination reset Command Output

```
L2>destination reset
default destination reset to all bricks and slots
```

dhcpserver

The following dhcpserver command set displays the setting for the current DHCP server (see Example 3-75) and enables and disables the DHCP server.

- dhcpserver
- dhcpserver on|enable
- dhcpserver off|disable

Note: For this command, the mnemonic dhcps may replace the word dhcpserver.

Example 3-75   dhcpserver Command Output

```
L2>dhcpserver
DHCP server startup enabled (if no other DHCP server is found).
```

env

The following env command set provides an environmental status summary.

- env summary
- rack <*rng*> slot <*rng*> env summary

Note: For this command, r may replace the word rack; s may replace the word slot and may also replace the word summary.

Use env summary to generate an environmental summary of an entire system.
Example 3-76 shows sample output.

Example 3-76   env summary Command Output

```
L2>env summary
all environmental conditions appear normal
L2>
```

Use rack *<rng>* slot *<rng>* env summary to display an environmental summary
for specified rack number(s) and slot number(s).

Note: For the *<rng>* variable, you can enter a single number, or two numbers separated
by a hyphen to indicate a range, or numbers separated by commas to indicate separate
items. For *<rng>*, you can also enter an asterisk or the word "all" to select all the items
available.

---

help

Use the help command to generate a list of all of the system controller commands. Use
the help *<command>* command to display more information on a single command. The
variable *<command>* is the name of a command.

Note: For this command, hlp may replace the word help.

For information about entering commands, see "Sending Commands to Specific
Devices" on page 33.

ip

Use the following ip command set to set, clear, and show the L2 controller static IP
address configuration settings.

- ip
- ip *<addr>* *<netmask>*
- ip *<addr>* *<netmask>* *<broadcast>*
- ip clear|reset

Use the `ip` command to display the L2 controller static IP address settings. Example 3-77 shows sample output.

Example 3-77   ip Command Output

```
essc1-001-L2>ip
addr: 137.38.88.197 netmask: 255.255.255.0 broadcast addr: 137.38.88.255
essc1-001-L2>
```

Use the other commands in this set as follows:

- Use ip *<addr>* *<netmask>* to set the address and netmask L2 static IP configuration parameters.

- Use ip *<addr>* *<netmask>* *<broadcast>* to set the address, netmask, and broadcast L2 static IP configuration parameters.

- Use ip clear|reset to clear the L2 static IP address settings.

l1

The following l1 command set enters L1 mode or sends an L1 controller command to a specified L1 controller.

- l1
- rack *<rng>* slot *<rng>* l1
- *<rack>*.*<slot>* l1
- rack *<rng>* slot *<rng>* l1 *<command>*
- *<rack>*.*<slot>* l1 *<command>*
- *<ip>*:*<port>*:*<l1>* l1 *<command>*
- :*<port>*:*<l1>* l1 *<command>*
- l1 *<command>*

Note: For these commands, the mnemonics r and s may replace the words rack and slot.

Use l1 to engage the default L1 command processor or enter the l1 mode.

Use `rack <rng> slot <rng> l1` to engage the L1 command processor of a brick in a specified rack and slot. The *<rng>* variable following `rack` is one or more rack numbers, and the *<rng>* variable following `slot` is one or more slot numbers. Example 3-78 shows sample output.

Note: For the *<rng>* variable, you can enter a single number, or two numbers separated by a hyphen to indicate a range, or numbers separated by commas to indicate separate items. For *<rng>*, you can also enter an asterisk or the word "all" to select all the items available.

Example 3-78   `rack <rng> slot <rng> l1` Command Output

```
L2>r 1 s 19 l1
entering L1 mode 001r19, <CTRL-T> to escape to L2

001r19-L1>
```

Alternatively, you can use *<rack>*.*<slot>* `l1` to engage the L1 command processor for a specified rack and slot.

Use `rack <rng> slot <rng> l1 <command>` to send a command to a destination brick without changing the default destination value. The *<rng>* variable following `rack` is one or more rack numbers, the variable *<rng>* following `slot`, is one or more slot numbers, and the variable *<command>* is an L1 controller command such as `config`. Example 3-79 shows sample output.

Note: For the *<rng>* variable, you can enter a single number, or two numbers separated by a hyphen to indicate a range, or numbers separated by commas to indicate separate items. For *<rng>*, you can also enter an asterisk or the word "all" to select all the items available.

Example 3-79   `rack <rng> slot <rng> l1 config` Command Output

```
L2>r 1 s 7 l1 config
:0 - 001c07
:1 - 001i21
L2>
```

Alternatively, you can use *<rack>*.*<slot>* `l1 <command>` to send a command to a destination brick without changing the default destination value.

Use *<ip>*:*<port>*:*<l1>* l1 *<command>* to send a command to a destination brick that does not yet have a brick identification number assigned to it. The variable *<ip>* is an IP address, the variable *<port>* is the port number, the variable *<l1>* is the L1 controller index number, and the variable *<command>* is an L1 controller command.

Use :*<port>*:*<l1>* l1 *<command>* to send a command to a destination brick that does not yet have a brick identification number assigned to it. The variable *<port>* is the port number, the variable *<l1>* is an L1 controller index number, and the variable *<command>* is an L1 controller command.

Use l1 *<command>* to send a specified L1 controller command to a default destination.

l2

Use the following l2 command set to engage and lock a specified L2 command processor or to send a command to specified L2 command processor(s).

- l2
- rack *<rng>* slot *<rng>* l2
- rack *<rng>* slot *<rng>* l2 *<command>*
- *<ip>* l2 *<command>*

Note: For these commands, the mnemonics r and s may replace the word rack and slot.

Use the l2 command to engage and lock the L2 command processor. Example 3-80 shows sample output.

Example 3-80   l2 Command Output

```
L2>l2
L2 command processor engaged, <CTRL-D> for console mode.
```

Use `rack <rng> slot <rng> l2` to engage a specified L2 controller. The *<rng>* variable specifies the rack number(s) and slot number(s).

Note: For the *<rng>* variable, you can enter a single number, or two numbers separated by a hyphen to indicate a range, or numbers separated by commas to indicate separate items. For *<rng>*, you can also enter an asterisk or the word "all" to select all the items available.

Use `rack <rng> slot <rng> l2 <command>` to send a specified command to a specified L2 controller(s). Use `<ip> l2 <command>` to send a specified command to an L2 controller specified with its IP address.

l2find

The `l2find` command lists all of the L2 controllers that are connected together on the same subnet (see Example 3-81).

Example 3-81   `l2find` Command Output

```
L2>l2find
13 L2's discovered:

IP              SSN      NAME             RACK FIRMWARE
--------------- -------- ---------------- ---- ------------
[ L2's with System Serial Number NOT set ]
  137.38.82.101                           000 L3 controlle
  137.38.82.102                           000 L3 controlle
[ L2's with different System Serial Numbers ]
  137.38.82.156 L1000625                  001 1.8.0
   137.38.82.51 N1000405 itsys1           111 1.7.7
   137.38.82.58 L0000002 klsys2           002 1.7.7
  137.38.82.159 L0000138 klsys4           004 1.7.9
  137.38.82.162 L0000005 klsys5           005 1.7.7
   137.38.82.57 L0000007 klsys7           007 1.8.0
   137.38.82.55 L0000123 perch            009 1.7.7
  137.38.82.158 L0000018 shrimp           018 1.7.7
   137.38.82.50 N0000001 sn2-dbg2         022 1.7.7
   137.38.82.52 M0000114 snapper          015 1.7.7
  137.38.82.157 L0000119 whale            008 1.7.7
```

l2display

Use the following l2display command set to set the functioning of your L2 controller touch display.

- `l2display debug` *<exp>*
- `l2display debug`
- `l2display enable`
- `l2display disable`
- `l2display`

Note: For these commands, `l2dsp` can replace `l2display` in your entries.

Use the commands in this set as follows:

- Use `l2display debug` *<exp>* to set the L2 controller touch display to display the debug flags specified in *<exp>*.
- Use `l2display debug` to show the L2 controller touch display debug flags.
- Use `l2display enable` to enable the L2 controller touch display.
- Use `l2display disable` to disable and lock the L2 controller touch display.
- Use `l2display` to show whether the L2 controller touch display is locked.

log

The following `log` command set displays the contents of the log, resets the log, and writes an entry into the log.

- `log`
- `log clear|reset`
- `log insert` *<entry>*

Use the `log` command to view the contents of the log. If the log is empty, the output from the `log` command is `log is empty`.

Use `log clear|reset` to empty the log. Use `log insert <entry>` to write a line in the log. The variable *<entry>* is text to enter in the log. Example 3-82 shows sample output where the text for *<entry>* is "Start the Test."

Example 3-82   `log insert Start the Test` Command Output

```
L2>log insert Start the Test
log entry made.
```

multisys

Use the following multisys command set to enable, disable, and show the settings for multiple system network sharing.

- `multisys`

- `multisys on`

- `multisys off`

Note: For these commands, `msys` can replace the word `multisys`.

Use the `multisys` command to view the current setting for the L2 controller multiple system network sharing. Example 3-83 shows sample output.

Example 3-83   `multisys` Command Output

```
L2>multisys
L2 multiple system network support enabled.
L2's will only connect to L2s with same system SN.
L2>
```

Use `multisys on` to enable multiple L2 controller system network sharing. Network sharing enables L2s to communicate only with other L2s that have the same serial number. Use `multisys off` to disable multiple L2 controller system network sharing. When L2 controller system network sharing is disabled, L2s communicate with all L2s.

nvram reset

The `nvram reset` command returns the NVRAM settings of the L2 controller to the factory default settings.

pbay

Use `pbay config` to report the bricks connected to each power bay. Use `pbay <command>` to send a specified command to one L1 controller connected to each power bay.

ping

Use the following ping command set to set the ping function between L2 controller and L2 controllers or L2 controller and L1 controllers.

- `ping`
- `ping min` *<frequency in seconds>*
- `ping max` *<frequency in seconds>*
- `ping l2` *<frequency in seconds>*
- `ping sender` *<1 or 0>*
- `ping l1` *<frequency in seconds>*
- `ping reset`

Use the commands in this set as follows:

- Use `ping` to display the L2 controller to L2 controller or L2 controller to L1 controller ping configurations.
- Use `ping min` *<frequency in seconds>* to set the L2 controller to L2 controller ping response to a minimum specified in *<frequency in seconds>*.
- Use `ping max` *<frequency in seconds>* to set the L2 controller to L2 controller ping response to a maximum specified in *<frequency in seconds>*.
- Use `ping l2` *<frequency in seconds>* to set the L2 controller to L2 controller ping frequency.

- Use `ping sender` *<1 or 0>* to enable (*1*) or disable (*0*) sender ping as a valid response.

- Use `ping l1`*<frequency in seconds>* to set L2 controller to L1 controller ping frequency in seconds.

- Use `ping reset` to reset L2 controller to L2 controller and L2 controller to L1 controller ping configuration to default.

power

The following power command set displays the power status of the bricks and powers up and powers down the bricks.

- `power`

- `power up`

- `power down`

- `rack` *<rng>* `slot` *<rng>* `power`

- `rack` *<rng>* `slot` *<rng>* `power up`

- `rack` *<rng>* `slot` *<rng>* `power down`

- `power summary`

- `rack` *<rng>* `slot` *<rng>* `power summary`

- `power delay`

- `power delay` *<milliseconds>*

Note: For the power command set, the mnemonics `pwr`, `u`, `d`, and `s` may replace the words `power`, `up`, `down`, and `summary`. The mnemonics `r` and `s` may replace the words `rack` and `slot`.

Use the `power` command to view the power status of each brick identified by the destination set of commands. Use the `power up` and `power down` commands to power up or power down each brick identified by the destination set of commands. Example 3-84 shows sample output.

Example 3-84   power Command Output

```
L2>power
001i21:
power appears on
001r19:
power appears on
001c07:
power appears on
001c10:
power appears on
```

Use rack *<rng>* slot *<rng>* power to view the power status of a specified brick. Use rack *<rng>* slot *<rng>* power up or rack *<rng>* slot *<rng>* power down to power up or power down specified bricks. The variable *<rng>* following rack is one or more rack numbers and the variable *<rng>* following slot is one or more slot numbers. Example 3-85 shows sample output.

Example 3-85   rack *<rng>* slot *<rng>* power Command Output

```
L2>r 1 s 19 power
power appears on
```

Note: For the *<rng>* variable, you can enter a single number, or two numbers separated by a hyphen to indicate a range, or numbers separated by commas to indicate separate items. For *<rng>*, you can also enter an asterisk or the word "all" to select all the items available.

Use power summary to display a power status summary. Use rack *<rng>* slot *<rng>* power summary to display the power status summary for specified bricks. Example 3-86 shows sample output.

Example 3-86   power summary Command Output

```
L2>power summary
all 2 bricks appear powered off
margin: 1 default, 0 low, 0 normal, 0 high, 0 custom, 0 other
L2>
```

Use power delay to show the delay between power commands. Use power delay *<milliseconds>* to set the delay between power commands in milliseconds.

quit

Use the `quit` command to exit the L2 command processor.

rackid

Use the following rackid command set to display or set the L2 controller rack ID.

- `rackid`
- `rackid` *<decimal exp>*

Use the `rackid` command to display the rack ID of an L2 controller. Example 3-87 shows sample output.

Example 3-87   `rackid` Command Output

```
essc1-001-L2>rackid
rack ID is 001
essc1-001-L2>
```

Use `rackid` *<decimal exp>* to set the rack ID for an L2 controller, where *<decimal exp>* is the rack ID number for the L2 controller.

reboot_l2

The `reboot_l2` command reboots the L2 controller using the newest firmware image. The `reboot_l2 force` command reboots the L2 controller even if the firmware image is invalid.

reset

Use the following reset command set to reset brick(s).

- `reset`
- `rack` *<rng>* `slot` *<rng>* `reset`

Note: For the reset command set, you can use `rst` in the place `reset`, `r` in the place of `rack`, and `s` in the place of `slot`.

Use `reset` to reset the individual default brick or bricks.

Caution: If you enter a `reset` command at the L2 prompt, it will reset all the bricks in your server system, which can create problems for your operating system if the operating system is running when you enter the `reset` command. So make sure your operating system is shut down before using the reset command.

Use `rack <rng> slot <rng> reset` to reset the bricks specified by the rack number(s) and slot number(s) specified with the variable *<rng>*.

Note: For the *<rng>* variable, you can enter a single number, or two numbers separated by a hyphen to indicate a range, or numbers separated by commas to indicate separate items. For *<rng>*, you can also enter an asterisk or the word "all" to select all the items available.

select

The following select command set displays and sets the brick ID number of the compute brick and subchannel that receives console input.

- `select`
- `select terse`
- `select subchannel console`
- `select subchannel <a|b|c|d>`
- `select subchannel <exp>`
- `select <exp> <exp>`
- `select <rack>.<slot>`
- `select reset`

- select partition *<exp>*

- select filter on

- select filter off

Note: For this command set, the mnemonic sel, t, sub (or s), con, part (or p), res may replace the words select, terse, subchannel, console, partition, and reset.

Use the select command to view the brick ID number and subchannel that receives console input. Example 3-88 shows sample output. In this example, the brick ID is a C-brick in rack 2, slot 1 (002c01) and the subchannel is the console channel.

Example 3-88   select Command Output

```
L2>select
console input: 002c01 console
console output: not filtered
console detection: L2 detected
```

Alternatively, you can use select terse to provide a less detailed display of information identifying the brick ID number and the subchannel that receives the console input.

Use select subchannel console to select the current console as the subchannel of the brick that receives the console input. Use select subchannel <a|b|c|d> to select CPU a, b, c, or d of the brick that receives the console input. Alternatively, use select subchannel *<exp>* to select the subchannel CPU of the brick that receives that console input. In this command, the variable <a|b|c|d> is the subchannel letter (a, b, c, or d CPU) of the brick to receive the console input. Example 3-89 shows sample output.

Example 3-89   `select subchannel <a|b|c|d>` Command Output

```
L2>select subchannel c
console input: 003c01 CPUc
console output: not filtered

console detection: L2 detected
```

Use `select` *<rack>*.*<slot>* to select the brick (brick ID) that receives console input. The variable *<rack>* is a rack number and the variable *<slot>* is a slot number. In Example 3-90, the brick ID is a C-brick in rack 3, slot 1 (003c01).

Example 3-90   `select` *<rack>*.*<slot>* Command Output
```
L2>select 3.1
console input: 003c01 console
console output: not filtered
console detection: L2 detected
```

Alternatively, use `select` *<exp>* *<exp>* to enter the rack and slot of the brick that receives the console input.

Use `select reset` to reset the selection of the brick to receive the console input to the default setting. Use `select partition` *<exp>* to select the partition and system console to receive the console input.

Use `select filter on` to turn the console output filter on, and `select filter off` to turn the console output filter off. Example 3-91 shows sample output.

Example 3-91   `select filter on` Command Output

```
L2>select filter on
console input: 003c01 console
console output: filtered

console detection: L2 detected
```

serial

Use the following serial command set to view and set the L2 controller system serial number.

- `serial`

- `serial set <str>`

Use the `serial` command to view the L2 controller system serial number. Example 3-92 shows sample output.

Example 3-92   `serial` Command Output

```
L2>serial
L2 system serial number: not set.
```

Use `serial set <str>` to set the L2 controller system serial number. The variable *<str>* is the L2 system serial number. Example 3-93 shows sample output.

Example 3-93   `serial set <str>` Command Output

```
L2>serial set L0000010
L2 system serial number set to L0000010.
L2> serial
L2 system serial number: L0000010.
```

shell

Use the `shell` command to escape to the L2 operating system. Example 3-94 shows sample output.

Example 3-94   `shell` Command Output

```
L2>shell
bash$ rm core
bash$ exit
exit
L2>
```

Note: For this command, you may use the character `!` instead of the word `shell`.

smp

The following smp command set displays the status of the system maintenance port (SMP) network connection. These commands are also used to join a specified SMP group or leave a current SMP group.

- smp

- smp verbose

- smp join *<exp>*

- smp leave

Note: For this command set, the mnemonic v may replace the word verbose.

Use the smp command to view the status of the SMP network connection. Example 3-95 shows a sample output.

Example 3-95   smp Command Output

```
essc1-001-L2>smp
Session    Who            Group  Mode   Console
-------    -------------  -----  ----   ----------------------
>>>  6     network port      6   L2     038c01 console (default)
     1     modem port        1   L2     038c01 console (default)
     1     modem port        1   L2     038c01 console (default)

essc1-001-L2>
```

Use smp verbose to display an expanded status of the SMP network connection.

Use smp join *<exp>* to join a specified SMP group identified by the *<exp>* variable. Use the smp leave command to leave the current SMP group.

sysname

Use the following sysname command set to display or set the system name (name of the L2 controller).

- sysname
- sysname *<str>*

Use the sysname command to display the name of the system. Example 3-96 shows a sample output.

Example 3-96   sysname Command Output

```
essc1-001-L2>sysname
L2 system name : essc1.
essc1-001-L2>
```

Use sysname *<str>* to set a system name where the variable *<str>* is the system name. Example 3-97 shows sample output.

Example 3-97   sysname *<str>* Command Output

```
essc1-001-L2>sysname essc2
L2 system name : essc2.

essc2-001-L2>
```

version

Use the following version command set to view information about the L1 and L2 controller firmware versions running in the system controller.

- version
- l1 version
- version verbose
- version summary
- version check
- rack *<rng>* slot *<rng>* version summary
- rack *<rng>* slot *<rng>* version check

Note: For this command, `ver` may replace the word `version`, `v` may replace `verbose`, `s` may replace `summary`, `c` may replace `check`, `r` may replace `rack`, and `s` may replace `slot`.

The `version` command displays the L2 controller firmware version currently running in the system controller, as shown in Example 3-98.

Example 3-98   `version` Command Output

```
L2>version
L2 version 0.4.0
```

Use the other commands in this set as follows:

- Use `l1 version` to view the L1 controller firmware version currently running in the system controller.

- Use `version verbose` to display the verbose view of the L2 controller firmware version currently running in the system controller.

- Use `version summary` to display a summary of the L1 and L2 controller firmware versions currently running in the system controller.

- Use `version check` to display any L1 and L2 controller firmware version mismatches in the system.

- Use `rack <rng> slot <rng> version summary` to display the L1 and L2 controller firmware versions currently running in the rack(s) and slot(s) that you specify with the <rng> variable following `rack` and the <rng> variable following `slot`.

- Use `rack <rng> slot <rng> version summary` to display the L1 and L2 controller firmware versions mismatches in the rack(s) and slot(s) that you specify with the <rng> variable following `rack` and the <rng> variable following `slot`.

Note: For the <rng> variable, you can enter a single number, or two numbers separated by a hyphen to indicate a range, or numbers separated by commas to indicate separate items. For <rng>, you can also enter an asterisk or the word "all" to select all the items available.

Index