VadaTech MicroTCA

Carrier Manager Command Line Interface

Reference Manual

February 22, 2010

Version 2.1

vadatech inc
THE POWER OF VISION
TM

Copyright

Notice

While reasonable efforts have been made to assure the accuracy of this document, VadaTech, Inc. assumes no liability resulting from any omissions in this document or from the use of the information obtained herein. VadaTech reserves the right to revise this document and to make changes periodically and the content hereof without obligation of VadaTech to notify any person of such revision or changes.

Electronic versions of this material may be read online, downloaded for personal use, or referenced in another document as a URL to the VadaTech Incorporated Web site. The text itself may not be published commercially in print or electronic form, edited, translated, or otherwise altered without the permission of VadaTech, Inc.

It is possible that this publication may contain reference to or information about VadaTech products (machines and programs), programming, or services that are not available in your country. Such references or information must not be construed to mean that VadaTech intends to announce such products, programming, or services in your country.

Trademarks

Revision History

Doc Rev	Description of Change	Revision Date
1.0.0	Preliminary document	08/31/2008
1.1.0	Updated to reflect UTC001 application release version 1.1.1	10/31/2008
1.1.1	Updated to reflect Get Event Code Description changes	11/05/2008
1.1.2	Updated MCH FRU ID values	01/06/2009
1.2.0	Release	01/06/2009
1.3.0	Updated for UTC001 release 1.3.0.1	02/10/2009
1.4.0	Updated for UTC001 release 1.3.3	03/12/2009
2.0.0	Updated for MCH release 1.4.0	05/20/2009
2.0.1	Updated Fro MCH release 1.4.4	02/22/2010

Table of Contents

Tables

1 Overview

This document covers the initial connection to the Command Line Interface (CLI), and the available commands to the interface.

1.1 Applicable Products

- UTC001
- UTC003
- VT850

1.2 Document References

- Intelligent Platform Management (IPMI) Interface Specification v2.0
- PICMG® 3.0 Revision 3.0 AdvancedTCA® Base Specification
- PICMG® AMC.0 R2.0 Advanced Mezzanine Card Base Specification
- PICMG Specification MTCA.0 R1.0 (MicroTCA)
- VadaTech MCH Software Management Manual
- VadaTech MicroTCA Management Interface Specification

1.3 Acronyms Used in this Document

Acronym	Description
ATCA	Advanced Telecommunications Computing Architecture
CLI	Command Line Interface
EMMC	Enhanced Module Management Controller
MC	Management Controller
MicroTCA	MicroTelecommunication Computing Architecture
PEF	Platform Event Filter(ing)
SEL	System Event Log

Table 1: Acronyms

2 MicroTCA Carrier Manager Command Line Interface

The MicroTCA Carrier Command Line Interface (CLI) provides an interface to a MicroTCA Carrier Manager. The CLI is based on the IPMI 2.0, AdvancedTCA™ PICMG 3.0, and MicroTCA 1.0 specifications. It uses a subset of commands that can be accessed directly or though a higher-level Management Application. Administrators can access the CLI though a telnet session, SSH, or the VT Carrier Manager serial port. Using the CLI, users can access information about the current state of the Carrier and obtain information such as the FRU population and monitor alarms, power management, current sensor values, and the overall health of the Carrier. The interface can also be used to update sensor thresholds and other Carrier-configurable parameters.

The information provided in this section is a supplement to the information found in the MicroTCA Management Interface Specification and the device's respective Getting Started documentation.

2.1 Carrier Manager Interfaces

The various VT MicroTCA Carrier Managers support the following front panel interfaces:

- Two Ethernet connections via RJ-45 connectors
- An RS-232 serial management port interface via an RJ-45 connector

Any of these interfaces can be used to log into the VT MicroTCA Carrier Manager. To configure the system IP address the first time, log in using the serial port console.

2.2 Logging in the First Time

If the system IP address is not configured / not configured properly for its network, logging onto the console the first time must be done via the serial port console.

The default administrative user name and password are '`root`' and '`root`', respectively, for console authentication.

2.3 Starting the CLI

Once logged on, the user can communicate with an active Carrier Manager, if present. There are two CLI operating modes the user can use to communicate with the Carrier Manager:

- **Interactive Mode** – use the "`cliUTCC`" command to connect to the Carrier Manager. You will be prompted for a username and password. Following the login, the CLI will start in an interactive mode. Interactive mode supports additional debugging commands not listed in this document. Enter 'help' at the command line to get the capabilities of the interface.

```
# cliUTCSH
Username: carrier
Password: carrier

SilverFox Version   5.1
Copyright (C) 2004-2009 by VadaTech Inc., all rights
reserved.

NOTE: Session inactivity timeout is set to 60 secs.

fox(127.0.0.1:624)#
```

  To exit this mode, enter '`exit`' or '`q`'.

- **Command Mode** – The user can execute CLI commands from the console prompt (see Section 5: Command Reference). This enables the user to write and execute scripts using the MicroTCA Carrier CLI implicitely.

```
# list_frus_present
```

Note: If a MicroTCA Shelf Manager is configured to be active on the same MCH as the Carrier Manager, the Command Mode defaults to the Shelf CLI. For information on how to use the Shelf CLI, refer to the MicroTCA Shelf Manager Command Line Interface Reference Manual.

Note: The CLI can be accessed from either the passive Manager's MCH or the active Manager's MCH using the above methods. Connections are forwarded to the active Manager.

2.4 Remote Access

If the MicroTCA Carrier Manager IP address is configured properly, the user can communicate with the Carrier Manager over the network. The CLI can run on a Remote

Management Console and connect to the Carrier Manager through the RMCP interface to send IPMI messages. The port address to connect to is **624**. By default, the Carrier Manager is configured with a CLI user account with administrative privileges.

Note: If a MicroTCA Shelf Manager is configured to be active on the same MCH as the Carrier Manager, the Command Mode defaults to the Shelf CLI. For information on how to use the Shelf CLI, refer to the MicroTCA Shelf Manager Command Line Interface Reference Manual.

By default, the Carrier Manager IP Connection uses address "192.168.40.230" to communicate with the active Carrier Manager on eth0:1. This can be changed during the initial configuration via the serial console.

View the available IP connections to the Carrier Manager:

```
cli_server> get_ip_conection

      MicroTCA Shelf IP Address     : 192.168.16.17
      MicroTCA Carrier IP Addresss  : 192.168.16.0
      MCH 1 IP Address              : 192.168.16.0
      MCH 2 IP Address              : 192.168.16.0
      Subnet Mask                   : 255.255.255.0
      Gateway Address 0             : 192.168.1.1
      Gateway Address 1             : 0.0.0.0
      Username                      : shelf
      Password                      : ********
```

To change the IP connections to the Carrier Manager, use the `set_ip_connection` CLI command. The changes take effect on the next power cycle.

For example, once the IP connections are set, if an IP connection is configured to "192.168.1.220", the remote CLI access to the active Carrier Manager is as follows:

```
# cliUTCC --host 192.168.1.220 --port 624
```

2.5 CLI Starting Options

The CLI accepts several starting options:

```
cliUTCC                 [-h|--host HOST_NAME]
                        [-p|--port PORT_NUMBER]
                        [-a]--keepalive]
                        [-l|--cmd_list]
                        [-v|--version]
                        [-U|--username USERNAME]
                        [-P|--password PASSWORD]
```

Host – specify a host address other than the default (local host)
Port – specify a port other than the default (624 for the MicroTCA Carrier CLI)
Keepalive – overrides the timeout for a CLI run in the Interactive Mode
Cmd-List – display a list of available CLI commands and exit; see `cli_commands`
Version – display the CLI version and exit; see `get_version`
Username – specify the login username; by default, the CLI will prompt for it
Password – specify the login password; by default, the CLI will prompt for it

3 Command Summary

The following is a summary of the commands supported by the VadaTech MicroTCA Carrier Manager CLI.

The commands are listed alphabetically by Group in Table 2. For more details about each command, refer to Section 5: Command Reference.

Note: The following commands are only applicable to a CLI connected to a MicroTCA Carrier Manager.

Group	Command	Description
Alarms	`alarm_clear`	remove or clear a triggered alarm from the list of active alarms
	`alarm_reset`	clear alarms for a given time (specified in minutes)
	`alarm_status`	display the active alarms and whether the alarm cut-off is enabled
	`alarm_test`	test the alarm subsystem
	`list_event_code_descriptions`	translate the MicroTCA Carrier diagnostic event codes
	`get_telco_alarm_state`	display the state of Telco alarms
	`get_telco_capabilities`	display the supported Telco alarms and alarm modes
	`set_telco_alarm_state`	set a given Telco alarm's state
Alerting	`get_pef_config_parameters`	display the configuration of a given Platform Event Filter (PEF) parameter, such as the configuration of the Event Filter Table and alert strings, as well as whether PEF is enabled/disabled
	`get_snmp_trap_info`	display the status of SNMP traps and available trap destinations
	`set_pef_config_parameters`	configure a given Platform Event Filter (PEF) parameter, such as the Event Filter Table and alert strings, as well as whether PEF is enabled/disabled
	`snmp_trap_disable`	disable SNMP traps for a given channel
	`snmp_trap_enable`	enable SNMP traps for a given channel
	`snmp_trap_get_address`	display a list of SNMP trap destinations for a given channel
	`snmp_trap_remove_address`	remove an SNMP trap destination from a given channel
	`snmp_trap_set_address`	modify an SNMP trap destination for a given channel

	`snmp_trap_test`	send a test SNMP trap to a given destination; get/clear status of test alert sent to a given destination
CLI	`cli_commands`	list all available CLI commands
	`cli_options`	describe the shorthand notation used for common CLI options
	`exit`	exit the CLI
	`get_version`	display the application and CLI versions
	`help`	display help for a given command, or display all commands, organized by group
	`q`	exit the CLI
Cooling	`get_cooling_parameters`	display the MicroTCA Carrier cooling management parameters
	`get_fan_geography`	display the Fan Tray to FRU connections
	`get_fan_info`	display the Fan Tray properties and hot-swap status for a given Fan Tray
	`get_fan_level`	display a given Fan Tray's current operating speed level
	`list_fan_trays`	display the locations of all Fan Trays installed in the MicroTCA Carrier
	`set_cooling_parameters`	configure the MircoTCA Carrier cooling management parameters
	`set_fan_level`	set the current operating speed level for a given Fan Tray
E-Keying	`get_amc_ptp`	display AMC e-keying information
	`get_carrier_ptp`	display the Carrier point-to-point connectivity information
	`get_clock_configuration`	display available clocks in a given FRU
	`get_clock_ptp`	display clock routing information
	`get_clock_state`	display current state information for a given clock ID
	`get_port_state`	display link status for a given FRU
	`set_clock_state`	configure a given clock
FRU Management	`activate`	activate a given FRU, bring it to M4 state
	`deactivate`	deactivate a given FRU, bring it to M1 state
	`fru_control`	change the state of a given FRU's payload
	`get_address_info`	display a given FRU's address information
	`get_device_id`	retrieve device information from a given FRU
	`get_event_receiver`	display the location of the event receiver for a given FRU
	`get_fru_activation_policy`	display the activation policy for a given FRU
	`get_fru_power_levels`	display a given FRU's power level
	`get_fru_state`	display a given FRU's hot-swap status
	`get_fru_temperature`	display the status of all temperature sensors for a given FRU
	`get_module_info`	display the configuration and hot-swap information for a given Module
	`hpm`	upgrade firmware or check version
	`list_device_sdr`	display a list of SDRs in a given FRU's Device SDR

	`list_fru_storages`	display the list of all FRU Inventory Devices located in the MicroTCA Carrier
	`list_sdr`	display a list of SDRs in the SDR Repository
	`read_fru_storage`	display content from a given FRU Inventory Device
	`set_event_receiver`	change the location of the event receiver on a given FRU
	`set_fru_extracted`	inform the MicroTCA Carrier Manager that a given Power Module is no longer installed
	`set_fru_power_level`	set the FRU power level for a given FRU
	`update_fru_version`	change the product version number for a given FRU
	`write_fru_storage`	write data to a given FRU inventory device
	`write_sdr_repository`	write data to a given SDR repository device
LAN	`get_channel_access`	display whether a given channel is enabled or disabled, whether alerting is enabled or disabled, and under what system modes the channel can be accessed
	`get_channel_cipher_suites`	display supported authentication, integrity, and confidentiality algorithms
	`get_channel_info`	display media and protocol information about a given channel
	`get_ip_connection`	display available network interfaces to a MicroTCA Carrier
	`get_lan_config_parameters`	display a given parameter related to IPMI LAN operation, such as network addressing information
	`get_session_info`	display session information
	`list_active_sessions`	display a list of active sessions
	`set_channel_access`	modify whether a given channel is enabled or disabled, whether alerting is enabled or disabled, and privilege level limit
	`set_ip_connection`	add or modify available network interfaces to the MicroTCA Carrier
	`set_lan_config_parameters`	set a given parameter required for IPMI LAN operation, such as the network addressing information
	`set_session_privilege_level`	request the ability to perform operations at a given privilege level for the active session
LEDs	`get_led_color_capabilities`	display a list of LEDs and the colors supported by the given FRU
	`get_led_properties`	display a list of LEDs controlled by a given FRU
	`get_led_state`	display the state of a given LED
	`set_led_state`	set the state for a given LED
Power	`get_power_channel_status`	display global and local power status information for a given power channel
	`get_power_feed_info`	display the power information for a given Power Module

	`get_power_feed_status`	display status information, such as the hot-swap status, role, and power status, for a given Power Module
	`get_power_management_info`	see `get_fru_activation_sequence`
	`get_power_policy`	display configured information for Power Module control, such as role, current limit, and associated power channels for each Power Module
	`power_feed_control`	requests a Chassis control request be sent to a given Power Module
	`power_feed_reset`	resets a given Power Module (requires redundancy)
SEL	`clear_sel`	erase the contents of a given FRU's System Event Log
	`get_sel`	display the contents of a given FRU's System Event Log
	`get_sel_info`	display information about a given FRU's System Event Log
Sensors	`get_sensor_event_enable`	display sensor event generation capabilities
	`get_sensor_hysteresis`	display sensor hysteresis values
	`get_sensor_info`	display sensor information
	`get_sensor_reading`	display sensor reading
	`get_sensor_threshold`	display sensor threshold
	`list_sensors`	display the list of sensors on a given FRU
	`set_sensor_event_enable`	set sensor event generation capabilities for a given sensor
	`set_sensor_hysteresis`	set sensor hysteresis values for a given sensor
	`set_sensor_threshold`	set sensor thresholds for a given sensor
System Administration	`get_user_access`	display privilege level and channel accessibility for a given user
	`list_users`	display the list of available users
	`list_users_access`	display channel access information for all users on a given channel
	`set_user_access`	configure privilege level and channel accessibility associated with a given user
	`set_user_info`	add user, set / change a given user ID's associated user name or password, and/or enable/disable a given user ID
System Management	`chassis_control`	change the power state of the Chassis or issue a diagnostic interrupt
	`check_ipmb0_status`	report the current status of all IPMB-0 links
	`failover`	initiate system failover (requires redundant MCH)
	`get_address_table`	display the FRU address table
	`get_diagnostics`	run diagnostics and display the results
	`get_fru_activation_sequence`	display the FRU activation sequence; see `get_power_management_info`
	`get_health`	provide a summary of Carrier alarm and health statuses

	`get_ipmb0_status`	display IPMB-0 sensor data for each FRU
	`get_location_info`	display location information for a given FRU
	`list_frus_present`	display the list of installed FRUs
	`list_modules_present`	display the list of installed Modules
	`set_carrier_number`	update the Carrier Number to a given number

Table 2: Command List

4 FRU Addressing

This section covers, in detail, the FRU addressing according to the PICMG® MicroTCA Base Specification. The first section covers general FRU addressing. The second section covers addressing as it relates to Inventory commands.

4.1 FRU Addressing

When using the CLI, all intelligent Field Replaceable Units (FRUs) on a MicroTCA Carrier can be addressed by the target FRU ID.

A FRU may also be specified using its physical address. The physical address is the combination of site type and site number.

The `get_address_info` command may be used to translate from one type of addressing into the other.

The physical address describes the physical location of a FRU in a Carrier, identifying both the type of FRU and a particular FRU of that type (e.g. AMC #1 versus AMC #2).

For example, to activate AMC #4's Management Controller, which is FRU ID 8 at site number 4, the following two options for FRU addressing can be used.

Carrier Manager address (20h) and target FRU ID (8):

```
    # activate –f 8
```

AMC site type (7h) and target site number (4):

```
    # activate –t 0x7 –n 4
```

Note: the Carrier address is 20h for method 1), and is optional. The examples in this document will omit the optional address field. Also note that throughout this document, “IPM Controller address” is used interchangeably with “Carrier address” and “Carrier Manager address”, all referring to the same access address of 20h.

The following table describes the various addressing options available for all applicable FRUs:

Entity ID	Entity	Slave Address	FRU Device ID	Physical Address	
				Site Type	Site Number
C2h	MCH 1	10h	3	0Ah	1
	MCH 2	12h	4	0Ah	2
C1h	AMC 1	72h	5	07h	1
	AMC 2	74h	6	07h	2
	AMC 3	76h	7	07h	3
	AMC 4	78h	8	07h	4
	AMC 5	7Ah	9	07h	5
	AMC 6	7Ch	10	07h	6
	AMC 7	7Eh	11	07h	7
	AMC 8	80h	12	07h	8
	AMC 9	82h	13	07h	9
	AMC 10	84h	14	07h	10
	AMC 11	86h	15	07h	11
	AMC 12	88h	16	07h	12
1Eh	CU 1	A8h	40	04h	1
	CU 2	AAh	41	04h	2
0Ah	PM 1	C2h	50	0Bh	1
	PM 2	C4h	51	0Bh	2
	PM 3	C6h	52	0Bh	3
	PM 4	C8h	53	0Bh	4
D0h-DFh	OEM Module 1	42h	60	C0h – CFh	1
	OEM Module 2	44h	61	C0h – CFh	2
	OEM Module 3	46h	62	C0h – CFh	3
	OEM Module 4	48h	63	C0h – CFh	4
	OEM Module 5	4Ah	64	C0h – CFh	5
	OEM Module 6	4Ch	65	C0h – CFh	6
	OEM Module 7	4Eh	66	C0h – CFh	7
	OEM Module 8	50h	67	C0h – CFh	8
	OEM Module 9	52h	68	C0h – CFh	9
	OEM Module 10	54h	69	C0h – CFh	10
	OEM Module 11	56h	70	C0h – CFh	11
	OEM Module 12	58h	71	C0h – CFh	12
	OEM Module 13	5Ah	72	C0h – CFh	13
	OEM Module 14	5Ch	73	C0h – CFh	14
	OEM Module 15	5Eh	74	C0h – CFh	15
	OEM Module 16	60h	75	C0h – CFh	16

Table 3: Physical Addressing

4.2 FRU Storage Addressing

Certain FRU IDs are reserved for Inventory use, and do not represent any managed FRU. FRU Inventory commands can address the FRUs outlined in Section 4.1: FRU Addressing, as well as those reserved for Inventory use. These FRUs have no physical address.

Entity ID	Entity	FRU Device ID
C2h	Physical Shelf Information 1	1
	Physical Shelf Information 2	2
D1h	Logical Carrier FRU Information	253

Table 4: Carrier Manager FRU Inventory

To read the complete FRU Inventory Information of Shelf Information 2, use the following syntax:

```
# read_fru_storage –f 2
```

To read the complete FRU Inventory Information of the logical Carrier Information, use the following syntax:

```
# read_fru_storage –f 253
```

5 Command Reference

This section covers, in detail, the commands of the CLI, providing the syntax, usage, and examples for each of the available commands.

5.1 Alerting

5.1.1 Alarm Clear

NAME

`alarm_clear` – remove a triggered alarm from the list of active alarms

SYNOPSIS

```
alarm_clear             [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        {-c|--event-code EVENT_CODE}
                        [-u|--unit-id UNIT_ID/SENSOR_NUMBER]
                        [-s|--sensor-type SENSOR_TYPE]

alarm_clear             {-t|--site-type SITE_TYPE}
                        {-f|--site-number SITE_NUMBER}
                        {-c|--event-code EVENT_CODE}
                        [-u|--unit-id UNIT_ID/SENSOR_NUMBER]
                        [-s|--sensor-type SENSOR_TYPE]
```

DESCRIPTION

This command is used to attempt to remove a triggered alarm from the list of active alarms. Some alarms cannot be cleared. Such alarms will persist until the problem is resolved.

IPMB Address and **FRU Identifier** specify the FRU that generated the alarm.

The target FRU can also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

Event Code is the code of the alarm to be removed. Some alarms are associated with a sensor. For these alarms, **Unit ID/Sensor Number** and **Sensor Type** must also be specified.

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Numbers
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 5: Alarm Clear applicable target FRUs

EXAMPLE

Remove alarm event represented by code 7Ch associated with the Carrier Manager located at FRU ID 0:

```
# alarm_clear –f 0 –c 0x7c
```

Remove alarm event represented by code 85h associated with the Fan Tray (site type 04h) at site number 1:

```
# alarm_clear –t 0x4 –n 1 –c 0x7c
```

SEE ALSO

```
alarm_status
list_event_code_descriptions
get_diagnostics
```

5.1.2 Alarm Reset

NAME

`alarm_reset` – turn off alarm LEDs and silence Telco alarms for a given time (specified in minutes)

SYNOPSIS

```
alarm_reset             {-t|--timeout MINUTES}
```

DESCRIPTION

This command is used to turn off alarm LEDs for a given time. If Telco alarms are supported, they are shut off for the same amount of time, as well.

Timeout is the time, in minutes, to keep the LEDs off and Telco alarms shut off.

EXAMPLE

Reset alarms for 3 minutes:

```
# alarm_reset --timeout 3
```

SEE ALSO

`alarm_status`

5.1.3 Alarm Status

NAME

`alarm_status` – display the active alarms and whether the alarm cut-off is enabled

SYNOPSIS

```
alarm_status
```

DESCRIPTION

This command is used to display the active alarms on the MicroTCA Carrier.

The command displays the following information:

- [!] Minor alarm is active
- [!!] Major alarm is active
- [!!!] Critical alarm is active

If the alarms are reset and/or silenced, the command displays the time for which the alarms are silenced.

SEE ALSO

```
alarm_reset
alarm_test
get_diagnostics
get_health
```

5.1.4 Alarm Test

NAME

`alarm_test` – enable test alarm(s)

SYNOPSIS

```
alarm_test              {[-m|--minor]|
                         [-M|--major]|
                         [-c|--critical]|
                         [-a|--all]}
```

DESCRIPTION

This command is used to test the Carrier Alarms and Telco Alarms.

Minor, **Major**, and **Critical** alarms may be tested individually. If **All** is specified, all alarms are tested.

Note: use `alarm_clear` to disable the alarms turned on for the test.

EXAMPLE

Test the minor alarm:

```
# alarm_test -m
```

Test all alarms:

```
# alarm_test -a
```

SEE ALSO

```
alarm_clear
alarm_status
```

5.1.5 Get Telco Alarm State

NAME

`get_telco_alarm_state` – display the state of Telco alarm(s)

SYNOPSIS

```
get_telco_alarm_state   [[-c|--critical]|
                         [-M|--major]|
                         [-m|--minor]|
                         [-p|--power-indicator]|
                         [-C|--alarm-cutoff]]
```

DESCRIPTION

This command is used to display the state of Telco alarms.

The user may specify one of the following alarms. If a target is not specified, the command displays information for all alarm states:

- Critical Alarm
- Major Alarm
- Minor Alarm
- Power Indicator
- Alarm Cutoff

EXAMPLE

Display all Telco alarm states:

```
# get_telco_alarm_state
```

Display the major Telco alarm state:

```
# get_telco_alarm_state –-major
```

SEE ALSO

```
get_telco_alarm_capabilities
set_telco_alarm_state
```

5.1.6 Get Telco Capabilities

NAME

`get_telco_capabilities` – display the Telco alarm states and modes available to a given MicroTCA Carrier

SYNOPSIS

```
get_telco_capabilities
```

DESCRIPTION

This command is used to display the Telco alarm states and modes available.

The command displays the following information:

- Critical Alarm, whether it can be controlled by `set_telco_alarm_state`
- Major Alarm, whether it can be controlled by `set_telco_alarm_state`
- Minor Alarm, whether it can be controlled by `set_telco_alarm_state`
- Power Alarm, whether it can be controlled by `set_telco_alarm_state`
- Test Mode, whether implemented or not
- Alarm Cutoff, whether implemented or not
- Minor Reset, whether implemented or not
- Major Reset, whether implemented or not

SEE ALSO

```
get_telco_alarm_state
set_telco_alarm_state
```

5.1.7 List Event Code Description

`list_event_code_description` – list the MicroTCA Carrier Manager diagnostic event codes and their descriptions

SYNOPSIS

```
list_event_code_descriptions
```

DESCRIPTION

This command is used to translate the event codes as reported by the `get_diagnostics` command.

The following is a table of the MicroTCA Carrier event codes, and recommended specification document(s) and document section(s) (if applicable) to refer to for more information. A value of "N/A" implies an internal VadaTech event type.

Event Code	Event Description	Specification Reference	Comments
60h	Normal	IPMI Intelligent Platform Management Interface Specification V2.0 Section : *Sensor Types and Data Conversion*	Threshold-based sensors, such as Voltage, Temperature, Fan RPM, etc, shall report these errors and assert that a threshold was crossed.
61h	Lower non-critical going low asserted		
62h	Lower non-critical going high asserted		
63h	Lower critical going low asserted		
64h	Lower critical going high asserted		
65h	Lower non-recoverable going low asserted		
66h	Lower non-recoverable going high asserted		
67h	Upper non-critical going low asserted		
68h	Upper non-critical going high asserted		
69h	Upper critical going low asserted		
6Ah	Upper critical going high asserted		
6Bh	Upper non-recoverable going low asserted		
6Ch	Upper non-recoverable going high asserted		
6Dh	Lower non-critical going low asserted		
6Eh	Lower non-critical going high asserted		
6Fh	Lower critical going low asserted		
70h	Lower critical going high asserted		
71h	Lower non-recoverable going low asserted		
72h	Lower non-recoverable going high asserted		
73h	Upper non-critical going low asserted		
74h	Upper non-critical going high asserted		
75h	Upper critical going low asserted		
76h	Upper critical going high asserted		
77h	Upper non-recoverable going low asserted		
78h	Upper non-recoverable going high asserted		

79h	IPMB-A disabled, IPMB-B disabled asserted	IPMI Intelligent Platform Management Bus Communications Protocol Specification 1.0 Section: *IPMB Sensors and event messages*	IPMB Fault events on the IPMB A or IPMB B.
7Ah	IPMB-A enabled, IPMB-B disabled asserted		
7Bh	IPMB-A disabled, IPMB-B enabled asserted		
7Ch	IPMB-A enabled, IPMB-B enabled asserted		
7Dh	Minor alarm test enabled	N/A	VadaTech Test only
7Eh	Major alarm test enabled		
7Fh	Critical alarm test enabled		
80h	-48V fuse alarm asserted	PICMG® 3.0 Revision 3.0 AdvancedTCA® Base Specification Section: *IPM Controller / FRU sensors*	Power Feed status sensors for the power Feed fuses have triggered an event.
81h	48V fuse alarm deasserted		
82h	Unable to read SDR	N/A	Unable to read from the Sensor Data Repository.
83h	Unable to read/write to sensor	N/A	Sensor initialization failed due to a communication error with the device.
84h	EEPROM read/write error	N/A	Communication error with an EEPROM device.
85h	SEL is full. Events are not being logged	N/A	System Event Log (SEL) is full; therefore, the Carrier Manager has disabled logging until the log is cleared by the user.
86h	Unable to read FRU information	N/A	Unable to successfully read FRU information due to internal failure.
C0h	Port enable failed	N/A	E-Keying port enable failed.
C1h	Cache point-to-point connectivity information failed	N/A	Caching point-to-point E-Key information from the specified FRU failed.
C2h	Communication lost	PICMG® 3.0 Revision 3.0 AdvancedTCA® Base Specification Section: *FRU hot swap event message*	A Device or FRU is currently at Communication lost state.
C3h	Unable to set FRU power level for FRU. FRU activation failed	N/A	Unable to activate FRU due to failure to Carrier Manager’s failure to set FRU active power level

C4h	FRU was deactivated due to overheating components	N/A	A FRU was deactivated by the Carrier Manager due to over-heating components.
C5h	One or more cooling units are absent or at communication lost state		Carrier Manger could not communicate with one of more cooling units and Carrier cooling is compromised.
C6h	Cooling configuration not found. Cooling disabled		Carrier Manger could not locate the Cooling configuration information and therefore has disabled cooling.
C7h	Cooling Management failure detected		Carrier Cooling management is not functional due to an internal failure.
C8h	Power Module failure	PICMG® Specification MTCA.0 R1.0 Section: *Abnormal power condition handling*	An abnormal power condition handling event occurred.
100h	Power Subsystem - redundancy lost		Power Module redundancy is lost.
101h	Power supply failure detected		Power supply failure detected in the specified Power Module.
102h	Power supply failure input lost (AC/DC)		Power supply input was lost in the specified Power Module.
130h	Payload power overcurrent condition detected	PICMG® Specification MTCA.0 R1.0 Section: *Power management commands and sensors*	Payload power for a FRU has encountered an overcurrent condition.
131h	Power Module max current inconsistency	N/A	The primary Power Module maximum current is not the same as the redundant Power Module maximum current.
132h	Power Module communication error	PICMG® Specification MTCA.0 R1.0 Section: *Abnormal power condition handling*	The Carrier Manager lost communication with a Power Module.
133h	Power Module redundancy error		A problem with the redundant Power Module caused an issue with the system redundant power status.
134h	Carrier FRU Information read error	N/A	Unable to read from the Carrier FRU Information.
135h	Shelf FRU Information read error		Unable to read from the Shelf FRU Information.
136h	Carrier FRU Information synchronization error	PICMG® Specification MTCA.0 R1.0 Section: *Redundant MCH Operation*	Unable to synchronize the FRU information from the active Carrier Manager's MCH to the backup MCH.

137h	Minor MCMC fault has occurred	N/A	VadaTech MCMC internal errors
138h	Major MCMC fault has occurred		
139h	Critical MCMC fault has occurred		

Table 6: Event Code Descriptions and References

SEE ALSO

`get_diagnostics`

5.1.8 Set Telco Alarm State

NAME

`set_telco_alarm_state` – enable, disable, or test supported Telco alarms

SYNOPSIS

```
set_telco_alarm_state   {-e|--enable}
                        {[-c|--critical] |
                         [-M|--major] |
                         [-m|--minor] |
                         [-p|--power-indicator] |
                         [-C|--alarm-cutoff]}

set_telco_alarm_state   {-d|--disable}
                        {[-c|--critical] |
                         [-M|--major] |
                         [-m|--minor] |
                         [-p|--power-indicator] |
                         [-C|--alarm-cutoff]}

set_telco_alarm_state   {-t|--test}
                        {[-c|--critical] |
                         [-M|--major] |
                         [-m|--minor] |
                         [-p|--power-indicator] |
                         [-C|--alarm-cutoff]}
```

DESCRIPTION

This command is used to enable, disable, or test supported Telco alarms.

An alarm state can be **Enabled**, **Disabled**, or, if supported, put into the **Test State.**

Only one of the following alarm states can be configured at a time:

- Critical Alarm
- Major Alarm
- Minor Alarm
- Power Indicator
- Alarm Cutoff

EXAMPLE

Enable power indicator Telco alarm:

```
set_telco_alarm_state –e -p
```

Test critical Telco alarm:

```
set_telco_alarm_state –-test –-critical
```

SEE ALSO

```
get_telco_alarm_capabilities
get_telco_alarm_state
```

5.2 Alerting

5.2.1 Get PEF Configuration Parameters

NAME

`get_pef_config_parameters` – display parameters related to PEF alerting, such as whether PEF is enabled or disabled, or the configuration of the event filter table and alert strings

SYNOPSIS

```
get_pef_config_parameters   {-p|--parameter PARAMETER_SELECTOR}
                            [-s|--set SET_SELECTOR]
                            [-b|--block BLOCK_SELECTOR]
                            [-r|--dump-raw]

get_pef_config_parameters   [-R|--revision-only]
```

DESCRIPTION

This command displays the parameters related to platform event filter (PEF) alerting. The full descriptions of the associated parameters are listed below.

Specifying the **Revision-Only** option will display only the parameters' revision.

Specifying the **Dump Raw** option will display the requested information, as well as the message response in raw hexadecimal format.

Parameter Selector is a 0-based index used to identify a setting parameter. The following parameters are supported:

0 = Set in progress. This parameter is used to indicate when LAN parameters are being updated, and when the updates are completed.
- Complete
- In progress

1 = PEF control
- PEF, whether enabled or disabled

2 = PEF action global control
- Alert, whether enabled or disabled

5 = Number of event filters

6 = Event Filter Table
- Filter, whether enabled or disabled and manufacturer or software configurable

- Event filter action
- Group control
- Diagnostic interrupt
- OEM
- Power cycle
- Reset
- Power off
- Alert
- Alert policy number
- Event severity to be used in a PET alert
- Generating FRU, channel number, and LUN
- Sensor type to match
- Sensor number to match
- Event/reading type to match
- Event data 1 event offset mask. This bit field is used to match different values of the least significant nibble of the Event Data 1 field. This enables a filter to provide a match on multiple offset values.
- Mask and compare fields for Event Data 1, 2, and 3. These fields are used in combination to allow wildcarding, 'one or more bit(s)', and exact comparisons to be made between bits in corresponding event data byte.

7 = Event Filter Table data 1. This parameter provides an aliased access to the first byte of the event filter data. This is provided to simplify the act of enabling and disabling individual filters by avoiding the need to do a read-modify-write of the entire filter data.

8 = Number of alert policy entries

9 = Alert Policy Table

- Policy number
- Policy
- Always send alert to destination
- If alert to previous destination was successful, do not send alert to this destination. Proceed to next entry in this policy set.
- If alert to previous destination was successful, do not send alert to this destination. Do not process any more entries in this policy set.
- If alert to previous destination was successful, do not send alert to this destination. Proceed to next entry in this policy set that is to a different channel.
- If alert to previous destination was successful, do not send alert to this destination. Proceed to next entry in this policy set that is to a different destination type.

10 = System GUID

- Use value returned from Get System GUID or the following value
- System GUID

11 = Number of alert strings

12 = Alert string keys. If, in the Alert Policy Table, alert uses an event-specific alert string, this parameter is used as an alert string set. An alert string set, along with the event filter number, is used to look up the string associated with a particular event. Otherwise, this parameter is used as an alert string selector.

- Alert string selector, 0-based
- Event filter number, 1-based, 00h = unspecified
- Alert string set, 1-based, 00h = unspecified

13 = Alert strings

- Alert string selector, 0-based
- Alert string

For the following parameters, the user may specify a target alert string selector using the **Set Selector** parameter. The set (alert string) selector is a 0-based index. Alert string 0 is always present as a volatile destination that is used with `snmp_trap_test`. If a target is not specified, the command displays information for all sets (strings).

- 12 = Alert strings
- 13 = Alert string keys

For the following parameters, the set selector may also be specified:

- 6 = Event Filter Table, 1-based
- 7 = Event Filter Table Data 1, 1-based
- 9 = Alert Policy Table, 1-based
- 15 = Group Control Table, 1-based

Block is unused in the current implementation.

The command displays a given parameter setting, as described above.

EXAMPLE

View event filter number 9:

```
# get_pef_config_parameters --parameter 6 –-set 9
```

View all alert strings:

```
# get_pef_config_parameters –p 13
```

View PEF configuration parameter revision:

```
# get_pef_config_parameters –R
```

SEE ALSO

```
set_pef_config_parameters
```

5.2.2 Get SNMP Trap Information

NAME

`get_snmp_trap_info` – display status of SNMP traps and available trap destinations

SYNOPSIS

```
get_snmp_trap_info      [-c|--channel-number CHANNEL_NUMBER]
```

DESCRIPTION

This command is used to display the status of SNMP traps and available trap destinations.

The user may specify a target **Channel Number**, which is a 0-based index used to identify a messaging channel. To display information for the channel used for the active session, the value 0Eh may be used. If a target is not specified, the command displays information for all available channels.

This command displays the following information:

- List of available channels
- SNMP traps, whether enabled or disabled. Alerting may also be disabled on a per-channel basis using `set_channel_access`.
- List of available trap destinations

EXAMPLE

Get trap information for channel 3:

```
# get_snmp_trap_info –-channel-number 3
```

SEE ALSO

```
set_channel_access
snmp_trap_disable
snmp_trap_enable
snmp_trap_get_address
snmp_trap_remove_address
snmp_trap_set_address
```

5.2.3 Set PEF Configuration Parameters

NAME

`set_pef_config_parameters` – modify setting parameters, such as Platform Event Filtering (PEF) enable/disable, and set the configuration of the event filter table and alert strings

SYNOPSIS

```
set_pef_config_parameters   {-p|--parameter PARAMETER_SELECTOR}
                            {[-d|--data DATA] |
                             [-S|--string STRING]}
```

DESCRIPTION

This command modifies the setting parameters, such as PEF enable/disable, and sets the configuration of the event filter table and alert strings.

Parameter Selector is a 0-based index used to identify a setting parameter (all changes are saved to both volatile and non-volatile inventories):

0 = Set in progress. This parameter is used to indicate when LAN parameters are being updated, and when the updates are completed.

- Complete
- In progress

2 = PEF action global control

- Alert, whether enabled or disabled

6 = Event Filter Table

- Filter, whether enabled or disabled and manufacturer- or software-configurable
- Event filter action
- Group control
- Diagnostic interrupt
- OEM
- Power cycle
- Reset
- Power off
- Alert
- Alert policy number
- Event severity to be used in a PET alert
- Generating FRU, channel number, and LUN
- Sensor type to match
- Sensor number to match

- Event/reading type to match
- Event data 1 event offset mask. This bit field is used to match different values of the least significant nibble of the Event Data 1 field. This enables a filter to provide a match on multiple offset values.
- Mask and compare fields for Event Data 1, 2, and 3. These fields are used in combination to allow wildcarding, one or more bit(s), and exact comparisons to be made between bits in corresponding event data byte.

7 = Event Filter Table data 1. This parameter provides an aliased access to the first byte of the event filter data. This is provided to simplify the act of enabling and disabling individual filters by avoiding the need to do a read-modify-write of the entire filter data.

9 = Alert Policy Table

- Policy number
- Policy
- Always send alert to destination
- If alert to previous destination was successful, do not send alert to this destination. Proceed to the next entry in this policy set.
- If alert to previous destination was successful, do not send alert to this destination. Do not process any more entries in this policy set.
- If alert to previous destination was successful, do not send alert to this destination. Proceed to the next entry in this policy set that is to a different channel.
- If alert to previous destination was successful, do not send alert to this destination. Proceed to next entry in this policy set that is to a different destination type.

10 = System GUID

- Use value returned from Get System GUID or the following value
- System GUID

12 = Alert string keys. If, in the Alert Policy Table, alert uses an event-specific alert string, this parameter is used as an alert string set. An alert string set, along with the event filter number, is used to look up the string associated with a particular event. Otherwise, this parameter is used as an alert string selector.

- Alert string selector, 0-based
- Event filter number, 1-based, 00h = unspecified
- Alert string set, 1-based, 00h = unspecified

13 = Alert strings

- Alert string selector, 0-based
- Alert string

Data is a variable length of byte values in hexadecimal. The length varies by parameter. Refer to the “Set PEF Configuration Parameters Command” section of the Intelligent Platform Management Interface Specification v2.0. If specifying an alert string, the **String** option must also be specified.

EXAMPLE

Add/modify event filter number 3 that sends an alert, with non-critical severity, for upper-critical going high assertion events from any voltage sensor. This event filter is software configurable and will execute alert policy number 5:

```
# set_pef_config_parameters --parameter 6 --data “03 00 01 05 08
      ff ff 02 ff 01 00 02 00 00 00 00 00 00 00 00 00”
```

Enable event filter number 3:

```
# set_pef_config_parameters –p 7 –d “03 80”
```

Change alert string to “testing” for selector 2:

```
# set_pef_config_parameters –p 13 –d 2 –S testing
```

SEE ALSO

`get_pef_config_parameters`

5.2.4 SNMP Trap Disable

NAME

`snmp_trap_disable` – disable SNMP traps for a given channel

SYNOPSIS

```
snmp_trap_disable       {-c|--channel-number CHANNEL_NUMBER}
```

DESCRIPTION

This command is used to disable SNMP traps on a given channel. By default, SNMP traps are not enabled. The changes are saved in non-volatile storage.

Channel Number is a 0-based index used to identify a messaging channel. SNMP trap are disabled on a per-channel basis. To disable SNMP trap on the channel used for the active session, the value 0Eh may be used.

EXAMPLE

Disable traps on channel 2:

```
# snmp_trap_disable –c 2
```

SEE ALSO

```
get_snmp_trap_info
snmp_trap_enable
```

5.2.5 SNMP Trap Enable

NAME

`snmp_trap_enable` – enable SNMP traps on a given channel

SYNOPSIS

```
snmp_trap_enable        {-c|--channel-number CHANNEL_NUMBER}
```

DESCRIPTION

This command is used to enable SNMP traps on a given channel. Alerting can be enabled on a per-channel basis using `set_channel_access`. By default, SNMP traps are not enabled. Changes are saved in non-volatile storage.

Channel Number is a 0-based index used to identify a messaging channel. SNMP traps are enabled on a per-channel basis. To enable SNMP traps on the channel used for the active session, the value 0Eh may be used.

EXAMPLE

Enable traps on the active channel:

```
# snmp_trap_enable –-channel-number 0xE
```

SEE ALSO

```
get_snmp_trap_info
set_channel_access
snmp_trap_disable
```

5.2.6 SNMP Trap Get Address

NAME

`snmp_trap_get_address` – display a list of SNMP trap destinations for a given channel

SYNOPSIS

```
snmp_trap_get_address   {-c|--channel-number CHANNEL_NUMBER}
                        [-d|--destination DESTINATION_NUMBER]
```

DESCRIPTION

This command is used to display a list of SNMP trap destinations for a given channel.

Channel Number is a 0-based index used to identify a messaging channel. SNMP trap destinations are configured on a per-channel basis. To display information for the channel used for the active session, the value 0Eh may be used.

Destination Number is a 0-based index used to identify an IP address that a SNMP trap can be sent to. The user may specify the target destination. If a target is not specified, the command displays all destinations available for the Carrier.

The command displays the alerting IP address as returned by `get_lan_config_parameters -p 19` with the corresponding channel number and set selector (destination number).

EXAMPLE

List SNMP trap destinations for channel 2:

```
# snmp_trap_get_address –c 2
```

List SNMP trap destinations for the channel used for the active session:

```
# snmp_trap_get_address –-channel-number 0xE
```

View the SNMP trap destination at index 3 for channel 2:

```
# snmp_trap_get_address –-channel-number 2 --destination 3
```

SEE ALSO

```
get_lan_config_parameters
snmp_trap_set_address
snmp_trap_remove_address
```

5.2.7 SNMP Trap Remove Address

NAME

`snmp_trap_remove_address` – remove an SNMP trap destination for a given channel

SYNOPSIS

```
snmp_trap_remove_address  {-c|--channel-number CHANNEL_NUMBER}
                          {-d|--destination DESTINATION_NUMBER}
```

DESCRIPTION

This command is used to remove an SNMP trap destination for a given channel. The changes are saved in non-volatile storage.

Channel Number is a 0-based index used to identify a messaging channel. SNMP trap destinations are configured on a per-channel basis. To display information for the channel used for the active session, the value 0Eh may be used.

Destination Number is a 0-based index used to identify an IP address that an SNMP trap can be sent to. Destination 0 on all channels represents the default destination number, and cannot be removed.

EXAMPLE

Remove SNMP trap destination 3 for channel 2:

```
# snmp_trap_remove_address –c 2 –d 3
```

Remove SNMP trap destination 3 for the channel used for the active session:

```
# snmp_trap_remove_address –c 0xE --destination 3
```

SEE ALSO

```
snmp_trap_get_address
snmp_trap_set_address
```

5.2.8 SNMP Trap Set Address

NAME

`snmp_trap_set_address` – modify an SNMP trap destination for a given channel

SYNOPSIS

```
snmp_trap_set_address   {-c|--channel-number CHANNEL_NUMBER}
                        {-d|--destination DESTINATION_NUMBER}
                        {-i|--ip-address IP_ADDRESS}
```

DESCRIPTION

This command is used to modify an SNMP trap destination for a given channel on the MicroTCA Carrier. The changes are saved in non-volatile storage.

Channel Number is a 0-based index used to identify a messaging channel. SNMP trap destinations are configured on a per-channel basis. To display information for the channel used for the active session, the value 0Eh may be used.

Destination Number is a 0-based index used to identify an IP address that a SNMP trap can be sent to. The user may specify the target destination. If a target is not specified, the command displays all destinations available.

IP Address is in decimal dotted notation.

The command is the same as using `set_lan_config_parameters` to change the following values:

- Destination type (parameter selector 18)
- Destination type = PET trap destination
- Alert acknowledge = no
- Alert acknowledge timeout / retry interval = 0 seconds
- Retries = 0
- Destination address (parameter selector 19)
- Address format = IPv4
- Address = IP_ADDRESS
- Gateway selector = use default
- Alerting MAC address = 00:00:00:00:00:00
- VLAN TAG (parameter selector 25)
- Address format = VLAN ID (not used)
- VLAN ID = 0
- CFI = 0
- User Priority = 0

EXAMPLE

Add IP address 10.1.12.36 as destination number 2 for channel 3:

```
# snmp_trap_set_address –c 3 –d 2 –a 10.1.12.36
```

Change destination 3 for the channel used for the active session to 10.1.14.109:

```
# snmp_trap_set_address –-channel-number 0xE --destination 3 --
      ip-address 10.1.14.109
```

SEE ALSO

```
set_lan_config_parameters
snmp_trap_get_address
snmp_trap_remove_address
```

5.2.9 SNMP Trap Test

NAME

`snmp_trap_test` – send a test SNMP trap to a given destination; get/clear status of test alert sent to a given destination

SYNOPSIS

```
snmp_trap_test          {-c|--channel CHANNEL_NUMBER}
                        {-d|--destination DESTINATION_NUMBER}
                        [{-g|--get-status} |
                         {-C|--clear-status}]
```

DESCRIPTION

This command is used to send a test SNMP trap to a given destination. It can also be used to get/clear the status of the test alert.

Channel Number is a 0-based index used to identify a messaging channel. SNMP trap destinations are configured on a per-channel basis. To send an alert on the channel used for the active session, the value 0Eh may be used.

Destination Number is a 0-based index used to identify an IP address to which a SNMP trap is sent. By default, destination 0 is always available to use for this command.

If **Get Status** is specified, the command displays the status of the test SNMP trap:

- No status – there was no SNMP trap sent. It is waiting its turn in the queue, there are no pending traps, or the channel specified is invalid.
- Normal end – the SNMP trap was sent successfully.
- Failed – timeout waiting for acknowledge on all retries.
- In progress – the SNMP trap is being sent out and the status is still pending.

If **Clear Status** is specified, the status is set to “No status.”

EXAMPLE

Send test SNMP trap to destination number 2 on channel 4:

```
# snmp_trap_test –c 4 –d 2
```

Get status of test SNMP trap sent to destination 3 on the channel used for the active session:

```
# snmp_trap_test –-channel-number 0xE --destination 3 –-get-
      status
```

Clear status of test SNMP trap sent to destination 5 on channel 4:

```
# snmp_trap_test –-channel-number 4 --destination 5 –C
```

SEE ALSO

```
get_snmp_trap_info
snmp_trap_get_address
```

5.3 CLI Utilities

5.3.1 CLI Commands

NAME

`cli_commands` – list all available commands

SYNOPSIS

```
cli_commands            [{-a|--alarm}       |
                         {-A|--alerting}    |
                         {-c|--cooling}     |
                         {-e|--e-keying}    |
                         {-f|--fru-mgmt}    |
                         {-i|--cli-info}    |
                         {-l|--lan}         |
                         {-L|--led}         |
                         {-p|--power}       |
                         {-E|--sel}         |
                         {-s|--sensor}      |
                         {-S|--system-admin}|
                         {-m|--system-mgmt} |
                         {-o|--all}]
```

DESCRIPTION

This command displays supported commands for the Carrier platform.

The following options can be specified to list the commands grouped by their function. The command is identical to using the `--cmd-list` option when starting the CLI; see **Section 2.5: CLI Starting Options** for more details.

- **alarm** – Alarm commands
- **alerting** – Alerting commands
- **cooling** – Cooling commands
- **e-keying** - Electronic Keying commands
- **fru-mgmt** – FRU Management commands
- **cli-info** – CLI commands
- **lan** – LAN commands
- **led** – LED commands
- **power** – Power commands
- **sel** – Sensor Event Log commands

- **sensor** - Sensor commands
- **system-admin** – Adminstration commands
- **system-mgmt** – System Management commands
- **all** – display all commands

Refer to Table 2 for more information on the command groups.

EXAMPLE

Display the supported System Management commands:

```
# cli_commands –m
```

Display all supported commands

```
# cli_commands -o
# cli_commands
```

SEE ALSO

```
cli_options
help
```

5.3.2 Describe CLI Options

NAME

`cli_options` – Describe the shorthand notations used for some CLI options

SYNOPSIS

```
cli_options
```

DESCRIPTION

This command describes the CLI shorthand options of `site options`, `address-fruid-options`, and `sensor-filter-options`.

SEE ALSO

```
cli_commands
help
```

5.3.3 Exit an Open CLI Session

NAME

`exit` – exit an open CLI session

SYNOPSIS

```
exit
```

DESCRIPTION

Note: This command is only available when using the CLI in Interactive Mode.

This command exits an open CLI session. This command functions identically to the `q` command.

SEE ALSO

`q`

5.3.4 Get Version Of CLI Software

NAME

`get_version` - display the release version of the MicroTCA Carrier CLI software

SYNOPSIS

```
get_version
```

DESCRIPTION

This command can only be issued in an open CLI session. This command displays the release version of the Carrier CLI. Alternatively, use CLI starting options to retrieve the version information; see Section 2.5: CLI Starting Options for more information.

5.3.5 Help

NAME

`help` - display an organized list of available CLI commands, or help for a given command

SYNOPSIS

```
help

<cli_command>           [-h | --help]
```

DESCRIPTION

Note: The first version of the command is only available when using the CLI in Interactive Mode.

If a command is provided, the `help` command lists the options available for a given CLI command. If a command is not provided, an organized list of available commands is displayed, identical to `cli_commands` without filtering capabilities.

EXAMPLE

See the help for the `get_address_info` command:

```
# get_address_info --help
```

Display all supported commands:

```
# help
```

SEE ALSO

```
cli_commands
get_options
```

5.3.6 Quit an Open CLI Session

NAME

`q` – quit an open CLI session

SYNOPSIS

```
q
```

DESCRIPTION

Note: This command is only available when using the CLI in Interactive Mode.

This command quits an open CLI session. This command functions identically to the `exit` command.

SEE ALSO

`exit`

5.4 Cooling Management

5.4.1 Get Cooling Parameters

NAME

`get_cooling_parameters` – display the MicroTCA Carrier cooling management parameters

SYNOPSIS

```
get_cooling_parameters
```

DESCRIPTION

This command will display the currently configured readiness allowance and monitor cycle, as well as the normal, ramp up, and ramp down incremental levels for handling temperature events.

The Carrier may contain Fan Trays that are used to maintain airflow within the Carrier. When the IPM Controllers within that region send temperature events to the Carrier Manager in configurations without an active MicroTCA Shelf, it is the responsibility of the Carrier Manager to adjust the fan levels in this region in an effort to alleviate the problem.

The command displays the following information:

- Normal operating level
- Increment level
- Decrement level
- Readiness allowance; time interval from power on until the Carrier Manager begins monitoring the cooling regions.
- Monitor cycle; minimum time between fan speed changes, in seconds
- Whether cooling is enabled/disabled

SEE ALSO

```
set_cooling_parameters
```

5.4.2 Get Fan Geography

NAME

`get_fan_geography` – display the Fan Tray to FRU connections as defined in the MicroTCA Carrier ATCA Fan Geography record

SYNOPSIS

```
get_fan_geography
```

DESCRIPTION

This command describes the FRUs associated with each configured Fan Tray in a Carrier.

The command displays the following information:

- Hardware address / FRU ID of a Fan Tray
- Site numbers of FRUs cooled by the specified Fan Tray

SEE ALSO

```
get_fan_info
get_fan_level
```

5.4.3 Get Fan Information

NAME

`get_fan_info` – display the Fan Tray properties and hot-swap status

SYNOPSIS

```
get_fan_info            [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}

get_fan_info            {-n|--site-number SITE_NUMBER}
```

DESCRIPTION

This command is used to obtain the Fan Tray properties for a given Fan Tray, its current hot-swap status, and its current operating speed level.

The target Fan Tray is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target Fan Tray may also be specified using the **Site Number**. The site number identifies a specific fan tray FRU.

The command displays the following information:

- Location of the Fan Tray identified by its IPMB address and FRU identifier
- Minimum speed level
- Maximum speed level
- Normal operating speed level
- If it is currently in the override control state, its override speed level

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Number(s)
CU	20h	40 – 41	1 – 2

Table 7: Get Fan Info applicable target FRUs

EXAMPLE

View the fan information for the Fan Tray located with FRU ID 40:

```
# get_fan_info –f 40
```

View the fan information for the Fan Tray located at site number 1:

```
# get_fan_info –n 1
```

SEE ALSO

```
get_fan_level
set_fan_level
```

5.4.4 Get Fan Level

NAME

`get_fan_level` – display current Fan Tray operating speed level

SYNOPSIS

```
get_fan_level          [-a|--ipmb-address IPMB_ADDRESS]
                       {-f|--fru-id FRU_ID}

get_fan_level          {-n|--site-number SITE_NUMBER}
```

DESCRIPTION

This command displays the current operating speed level for a given Fan Tray. This value can be anywhere between the minimum to maximum speed level for the Fan tray.

The target Fan Tray is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target Fan Tray may also be specified using the **Site Number.**

The command displays the following information:

- Location of the Fan Tray identified by its IPMB address and FRU identifier
- Whether the Fan Tray is in the local or override control state
- Current operating speed level

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Number(s)
CU	20h	40 – 41	1 – 2

Table 8: Get Fan Level applicable target FRUs

EXAMPLE

View the current operating speed level for the Fan Tray located with FRU ID 41:

```
# get_fan_level –f 41
```

View the current operating speed level for the Fan Tray located at site number 2:

```
# get_fan_level –n 2
```

SEE ALSO

```
get_fan_info
set_fan_level
```

5.4.5 List Fan Trays Installed

NAME

`list_fan_trays` – display the location of all Fan Trays installed in the MicroTCA Carrier

SYNOPSIS

```
list_fan_trays
```

DESCRIPTION

This command is used to obtain a list of all the Fan Trays installed in the Carrier and their locations.

The command displays the following information for each installed Fan Tray:

- Location
- Entity ID / Instance
- Device Name
- Hotswap State

SEE ALSO

```
get_fan_info
get_fan_level
```

5.4.6 Set Cooling Parameters

NAME

`set_cooling_parameters` – set the MicroTCA Carrier cooling management parameters

SYNOPSIS

```
set_cooling_parameters  {[-n|--normal-level NORMAL_LEVEL]
                         [-i|--increment-level INCREMENT_LEVEL]
                         [-I|--decrement-level DECREMENT_LEVEL]
                         [-a|--readiness-allowance READINESS_ALLOWANCE]
                         [-m|--monitor-cycle MONITOR_CYCLE]
                         [[-e|--enable]|
                          [-d|--disable]]}
```

DESCRIPTION

The Carrier may contain Fan Trays that are used to maintain airflow within the Carrer. When the IPM Controllers within that region send temperature events to the Carrier Manager, it is the responsibility of the Carrier Manager to adjust the fan levels in this region in an effort to alleviate the problem.

The Carrier Manager is responsible for monitoring all the Fan Trays in the Carrier when the MicroTCA Shelf is unavailable to do so. The Carrier Manager begins monitoring the region after the **Readiness Allowance** interval expires, which begins at power on. The Carrier Manager may begin lowering the Fan Tray speed depending on the region’s ambient temperature, which is determined by monitoring the various temperature sensors located on the FRUs and Chassis.

The readiness allowance is a time interval between 1-255 seconds. The default setting for the Carrier Manager cooling readiness allowance is 60 seconds. It is recommended to allow at least 60 seconds of readiness allowance during initial power on since all the payloads are being enabled.

The **Monitor Cycle** is the time interval in seconds that is used by the Carrier Manager to periodically monitor the Carrier’s temperature health. The default setting for the Carrier Manager cooling monitor cycle is 20 seconds. It is recommended to have a minimum of 20 seconds per monitor cycle for optimal performance of the Carrier Manager.

When temperature events require Carrier Manager intervention of the Fan Tray operating levels, the **Normal Level**, the normal run level of the Fan Trays during standard operation, is incremented or decremented by the level specified by the **Increment Level** and **Decrement Level** fields in order to raise or lower, respectively, the temperature in the troubled region.

A user may also enable or disable cooling monitoring by specifying **Enable** or **Disable.**

EXAMPLE

Enable intelligent cooling on the Carrier, specifying a cooling readiness allowance of 60 seconds, monitor cycle of 30 seconds, normal run level of 40, an increment level value of 10, and a decrement level of 3:

```
# set_cooling_parameters –a 60 –M 30 –e –i 10 –I 3 –n 40
```

SEE ALSO

`get_cooling_parameters`

5.4.7 Set Fan Level

NAME

`set_fan_level` – set the current operating speed level for a given Fan Tray

SYNOPSIS

```
set_fan_level           [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        {-l|--level FAN_LEVEL}

set_fan_level           {-n|--site-number SITE_NUMBER}
                        {-l|--level FAN_LEVEL}
```

DESCRIPTION

This command is used to set the current operating speed level for a given Fan Tray. This speed level can be anywhere from the minimum to the maximum speed level for the Fan tray. This command will override the MicroTCA Carrier or MicroTCA Shelf (whoever is the active cooling monitor) cooling management and set the Fan Tray at a given speed level and shall remain at override control. Note that overriding the cooling manager is not recommended. When this command is used, it is the responsibility of the user to monitor the Carrier cooling in the override state.

The target Fan Tray is specified using the IPM Controller's **IPMB Address** and **FRU Identifier.**

The target Fan Tray may also be specified using the **Site Number.**

The command displays the following information:

- Location of the Fan Tray identified by its IPMB address and FRU identifier
- Whether the Fan Tray is at local or override control
- Current operating speed level

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Number(s)
CU	20h	40 – 41	1 – 2

Table 9: Set Fan Level applicable target FRUs

EXAMPLE

Set the current operating speed level to 30 for the Fan Tray located with FRU ID 41:

```
# set_fan_level –f 41 –l 30
```

Set the current operating speed level to 30 for the Fan Tray at site number 1:

```
# set_fan_level –n 1 –l 30
```

SEE ALSO

```
get_fan_info
get_fan_level
```

5.5 E-Keying

5.5.1 Get AMC Point-to-Point Connectivity Information

NAME

`get_amc_ptp` – display e-keying information for an AMC, or for on-Carrier devices

SYNOPSIS

```
get_amc_ptp            [-a|--ipmb-address IPMB_ADDRESS]
                       {-f|--fru-id FRU_ID}

get_amc_ptp            {-t|--site-type SITE_TYPE}
                       {-n|--site-number SITE_NUMBER}
```

DESCRIPTION

This command is used to examine the available communication interfaces on an AMC or an on-Carrier device. This information is retrieved from the device's multi-records.

The target FRU is specified using the Carrier Manager's **IPMB Address** and the **FRU Identifier.**

The target FRU can also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

The command displays the following information for GUID link types, if any are defined:

- OEM GUID count

- Number of channels defined

The command displays the following information for each channel:

- Channel number
- Lane-to-port mapping

The command displays the following information for each point-to-point link:

- Channel number
- Lane mask
- Link type, in numeric and translated form
- Link type extension
- Link grouping ID
- Asymmetric match

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12

Table 10: AMC PTP Connectivity Information applicable target FRUs

EXAMPLE

View e-keying information for the AMC (site type 07h) located at site number 1:

```
# get_amc_ptp –t 7 –n 1
```

View e-keying information for on-Carrier devices on the MCMC located with FRU ID 3

```
# get_amc_ptp –f 3
```

SEE ALSO

```
get_carrier_ptp
```

5.5.2 Get Carrier Point-to-Point Connectivity Information

`get_carrier_ptp` – display MicroTCA Carrier point-to-point connectivity information

SYNOPSIS

```
get_carrier_ptp
```

DESCRIPTION

This command is used to examine the connections that a Carrier provides among on-board devices and AMC bays.

The command displays the following information for each Point-to-Point AMC Resource Descriptor defined by the Carrier:

- AMC Bay ID or on-Carrier device ID defining one endpoint common to a list of point-to-point connections
- List of point-to-point connections sharing one endpoint
- Port on the common endpoint
- AMC Bay ID or on-Carrier device ID defining the remote endpoint
- Port on the remote endpoint

SEE ALSO

```
get_amc_ptp
get_port_state
```

5.5.3 Get Clock Configuration

NAME

`get_clock_configuration` – display clock configuration descriptors for a given FRU

SYNOPSIS

```
get_clock_configuration  [-a|--ipmb-address IPMB_ADDRESS]
                         {-f|--fru-id FRU_ID}

get_clock_configuration  {-t|--site-type SITE_TYPE}
                         {-n|--site-number SITE_NUMBER}
```

DESCRIPTION

This command is used to display clock configurations that can be enabled or disabled on a FRU. A clock device may support multiple configurations, but only one of the configurations can be enabled at a time.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU can also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

The command displays the following information for each configuration:

- Descriptor number
- PLL status
- Clock family
- Clock accuracy
- Nominal frequency
- Minimum guaranteed frequency

- Maximum guaranteed frequency

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
MCH	20h	3 – 4	0Bh	1 – 2
AMC	20h	5 – 16	07h	1 – 12

Table 11: Clock Configuration applicable target FRUs

EXAMPLE

View clock configurations on the MCH located with FRU ID 3:

```
# get_clock_configuration –f 3
```

View clock configurations for the AMC (site type 07h) located at site number 2:

```
# get_clock_configuration –t 0x7 –n 2
```

SEE ALSO

```
get_clock_state
set_clock_state
get_clock_ptp
```

5.5.4 Get Clock Point-to-Point

NAME

`get_clock_ptp` – display backplane clock connectivity information

SYNOPSIS

```
get_clock_ptp
```

DESCRIPTION

This command is used to display clock point-to-point records. These records describe physical connections among clock ports on AMCs and on-Carrier devices.

The command displays the following information for each connection:

- Local clock ID
- Remote clock resource ID
- Remote clock ID
- Remote resource type
- Site number

SEE ALSO

```
set_clock_state
get_clock_state
get_carrier_ptp
```

5.5.5 Get Clock State

NAME

`get_clock_state` – display clock status for a given FRU

SYNOPSIS

```
get_clock_state         [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        {-i|--clock-id CLOCK_ID}
                        [-I|--device-id DEVICE_ID]

get_clock_state         {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
                        {-i|--clock-id CLOCK_ID}
                        [-I|--device-id DEVICE_ID]
```

DESCRIPTION

This command is used to display the clock configuration that is currently enabled on a given FRU. A clock configuration is enabled when there is an E-Keying match between two clock resources. The clock family, clock accuracy, and clock tolerance are compared to find a match. One clock resource must be a clock source, and the other clock resource must be a clock receiver.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU can also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

Clock ID is a number used to identify a clock on an AMC or on-Carrier device. Clock IDs for AMCs and backplanes are 1-based and defined by the AdvancedTCA specification. Clock IDs for on-Carrier devices are implementation-dependant, and are relative to the on-Carrier device ID.

The on-Carrier **Device Identifier** must be used when the target FRU is an MCMC.

The command displays the following information for each link:

- Clock state, enabled or disabled
- Clock direction, source or receiver
- PLL control status
- Configuration descriptor index
- Clock family
- Clock accuracy
- Nominal clock frequency

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12

Table 12: Get Clock State applicable target FRUs

EXAMPLE

View the status of Telco Clock A on the AMC located with FRU ID 8:

```
# get_clock_state -f 8 –i 1
```

View the status of clock 1 of on-Carrier device 3 on the MCH (site type 0Ah) located at site number 2:

```
# get_clock_state –t 0xa –n 2 –i 1 –I 3
```

SEE ALSO

```
set_clock_state
get_clock_configuration
get_clock_ptp
```

5.5.6 Get Port State

NAME

`get_port_state` – display link status for a given FRU

SYNOPSIS

```
get_port_state         [-a|--ipmb-address IPMB_ADDRESS]
                       {-f|--fru-id FRU_ID}
                       [-c|--channel-number CHANNEL_NUMBER]
                       [-d|--device-id DEVICE_ID]

get_port_state         {-t|--site-type SITE_TYPE}
                       {-n|--site-number SITE_NUMBER}
                       [-c|--channel-number CHANNEL_NUMBER]
                       [-d|--device-id DEVICE_ID]
```

DESCRIPTION

This command is used to display links that are enabled or disabled on a FRU. A link is enabled when there is an E-Keying match on the remote resource. The link type, extension, and designator are compared to find a match. The asymmetric match fields must be compatible as well.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU can also be specified using the Site Type and Number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

Channel Number is a 0-based number used to identify a channel on an AMC or on-Carrier device. The user may specify a target channel. If a target is not specified, the command displays information for all channels.

The on-Carrier **Device Identifier** may be used to filter results for a given on-Carrier device.

The command displays the following information for each link:

- Channel number
- Lanes used by link
- Link type
- Link extension
- Group ID
- Status of link, whether enabled or disabled

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12

Table 13: Get Port State applicable target FRUs

EXAMPLE

View the status of links on channel 2 for the AMC located with FRU ID 5:

```
# get_port_state –f 5 –c 2
```

View the status of all links with Device ID 7 on the MCH (site type 0Ah) located at site number 2:

```
# get_port_state –t 0xA –n 2 –d 7
```

SEE ALSO

```
get_amc_ptp
get_carrier_ptp
```

5.5.7 Set Clock State

NAME

`set_clock_state` – enable or disable clock.

SYNOPSIS

```
set_clock_state         [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        {-i|--clock-id CLOCK_ID}
                        [[-I|--device-id DEVICE_ID]
                         [-b|--backplane-clock]]
                        {[-s|--clock-source] |
                         [-S|--clock-receiver]}
                        {-d|--disable}

set_clock_state         [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        {-i|--clock-id CLOCK_ID}
                        [[-I|--device-id DEVICE_ID]
                         [-b|--backplane-clock]]
                        {[-s|--clock-source] |
                         [-S|--clock-receiver]}
                        {-e|--enable}
                        {-c|--clock-family CLOCK_FAMILY}
                        {-l|--accuracy-level ACCURACY_LEVEL}
                        {-F|--frequency FREQUENCY}
                        [[-m|--megahertz] |
                         [-k|--kilohertz] |
                         [-h|--hertz]]

set_clock_state         [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        {-i|--clock-id CLOCK_ID}
                        [[-I|--device-id DEVICE_ID]
                         [-b|--backplane-clock]]
                        {[-s|--clock-source] |
                         [-S|--clock-receiver]}
                        {[-U|--pll-use] |
                         [-B|--pll-bypass]}
                        {-c|--clock-family CLOCK_FAMILY}
                        {-l|--accuracy-level ACCURACY_LEVEL}
                        {-F|--frequency FREQUENCY}
                        [[-m|--megahertz] |
                         [-k|--kilohertz] |
                         [-h|--hertz]]
```

```
set_clock_state         {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
                        {-i|--clock-id CLOCK_ID}
                        [[-I|--device-id DEVICE_ID]
                         [-b|--backplane-clock]]
                        {[-s|--clock-source] |
                         [-S|--clock-receiver]}
                        {-d|--disable}

set_clock_state         {-t|--site-type SITE_TYPE)
                        {-n|--site-number SITE_NUMBER}
                        {-i|--clock-id CLOCK_ID}
                        [[-I|--device-id DEVICE_ID]
                         [-b|--backplane-clock]]
                        {[-s|--clock-source] |
                         [-S|--clock-receiver]}
                        {-e|--enable}
                        {-c|--clock-family CLOCK_FAMILY}
                        {-l|--accuracy-level ACCURACY_LEVEL}
                        {-F|--frequency FREQUENCY}
                        [[-m|--megahertz] |
                         [-k|--kilohertz] |
                         [-h|--hertz]]

set_clock_state         {-t|--site-type SITE_TYPE)
                        {-n|--site-number SITE_NUMBER}
                        {-i|--clock-id CLOCK_ID}
                        [[-I|--device-id DEVICE_ID]
                         [-b|--backplane-clock]]
                        {[-s|--clock-source] |
                         [-S|--clock-receiver]}
                        {[-U|--pll-use] |
                         [-B|--pll-bypass]}
                        {-c|--clock-family CLOCK_FAMILY}
                        {-l|--accuracy-level ACCURACY_LEVEL}
                        {-F|--frequency FREQUENCY}
                        [[-m|--megahertz] |
                         [-k|--kilohertz] |
                         [-h|--hertz]]
```

DESCRIPTION

This command is used to enable and disable clock sources and receivers.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU can also be specified using the Site Type and Number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

Clock ID is a number used to identify a clock on an AMC or on-Carrier device. Clock IDs for AMCs and backplanes are 1-based and defined by the AdvancedTCA specification. Clock IDs for on-Carrier devices are implementation-dependant, and are relative to the on-Carrier device ID.

The **Device ID** identifies an on-Carrier clock device. This field must be specified when enabling or disabling an on-Carrier clock device.

Enable and **Disable** enables and disables a given clock.

Every clock device is either a **Clock Source** or a **Clock Receiver**. The **–s** or **–S** option must match the clock device that is being enabled or disabled.

Some clock devices provide an optional PLL circuit. The clock e-keying process will always use the default PLL state of the clock device. The **–U** or **–B** option can be used to force the clock device to **Use** or **Bypass** the PLL. When specifying **–U** or **–B**, the remaining parameters must match the values reported by `get_clock_state`, such that any user input for the remaining parameters are ignored in lieu of the values already in the system.

The **Clock Family** is used to match clock sources with receivers. It must be provided when enabling a clock or changing the PLL status.

The **Accuracy Level** is smaller for more accurate clocks and larger for less accurate clocks. The specific meaning of an accuracy value is defined per clock family. This field must be provided when enabling a clock or changing the PLL status.

The **Frequency** is the nominal frequency of the clock, and must be provided when enabling a clock or changing the PLL status. The **–m**, **–k** and **–h** options allow the frequency to be specified in **Megahertz**, **Kilohertz**, or simply **Hertz**. By default, hertz is specified.

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12

Table 14: Set Clock State applicable target FRUs

EXAMPLE

Enable Telco Clock A, a SONET stratum 3 receiver running at 1.544 MHz, on the AMC located with FRU ID 7:

```
# set_clock_state –f 7 –i 1 –S –e –c 1 –l 60 –F 1544 -k
```

Enable the PLL for Telco Clock A, a SONET stratum 3 receiver running at 1.544 MHz, on the AMC (site type 07h) located at site number 3:

```
# set_clock_state –t 0x7 –n 3 –i 1 –S -U –c 1 –l 60 –F 1544 -k
```

Disable Telco Clock A on the AMC located with FRU ID 6:

```
# set_clock_state –f 6 –i 1 –S -d
```

Enable clock 1, a SONET stratum 3 source running at 8KHz, of on-Carrier device 3 on the MCH located with FRU ID 3:

```
# set_clock_state –f 3 –i 1 –I 3 –s –e –c 1 –l 60 –F 8 -k
```

SEE ALSO

```
get_clock_configuration
get_clock_state
```

5.6 FRU Management

5.6.1 Activate FRU

NAME

`activate` – activate a FRU, bringing it to state M4

SYNOPSIS

```
activate                [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}

activate                {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
```

DESCRIPTION

This command brings a FRU from M1 to M4 by clearing the activation-locked bit. This command is typically used in conjunction with the `deactivate` command. These commands can restart a FRU remotely, without using the hot-swap handle.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 15: FRU Activation applicable target FRUs

EXAMPLE

Activate the AMC located with FRU ID 13:

```
# activate –f 13
```

Activate the Fan Tray (site type 04h) located at site number 2:

```
# activate –t 0x4 –n 2
```

SEE ALSO

```
deactivate
```

5.6.2 Deactivate FRU

NAME

`deactivate` – deactivate a FRU, bringing it to M1

SYNOPSIS

```
deactivate              [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}

deactivate              {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
```

DESCRIPTION

This command brings a FRU from M4 to M1 by clearing the deactivation-locked bit. This command can be used to shut down a FRU remotely, without using the hot-swap handle. This command can be used in conjunction with activate to re-start a FRU.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 16: FRU Deactivation applicable target FRUs

EXAMPLE

Deactivate the AMC located with FRU ID 5:

```
# deactivate –f 5
```

Deactivate the Fan Tray (site type 04h) located at site number 1:

```
# deactivate –t 4 –n 1
```

SEE ALSO

```
activate
```

5.6.3 FRU Control

NAME

`fru_control` – change the state of a FRU's payload

SYNOPSIS

```
fru_control            [-a|--ipmb-address IPMB_ADDRESS]
                       {-f|--fru-id FRU_ID}
                       {[-c|--cold-reset]|
                        [-w|--warm-reset]|
                        [-r|--graceful-reboot]|
                        [-i|--diagnostic-interrupt]
                        [-q|--quiesce]}

fru_control            {-t|--site-type SITE_TYPE}
                       {-n|--site-number SITE_NUMBER}
                       {[-c|--cold-reset]|
                        [-w|--warm-reset]|
                        [-r|--graceful-reboot]|
                        [-i|--diagnostic-interrupt]
                        [-q|--quiesce]}
```

DESCRIPTION

This command is used to control the FRU's payload. Several control commands are available, as described below. The behavior of the FRU's payload in response to these commands will vary according to individual requirements. The command does not change the operational state of the FRU, which is typically M4 or FRU Active. All FRUs must support the Cold Reset control command. Support for the other control commands is optional.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

Only one of the following can be specified:

- **Cold Reset** causes a hardware reset, similar to a power-on reset.
- **Warm Reset** causes the payload to reset to a stable condition, preserving its operational state. Some FRUs may not support this control command.
- **Graceful Reboot** initiates a graceful shutdown and reboot of the Payload operating system. Some FRUs may not support this control command.
- **Diagnostic Interrupt** triggers a diagnostic interrupt to the FRU. Some FRUs may not support this control command.
- **Quiesce** brings the payload of a Module to a quiesced state. Some FRUs may not support this control command.

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 17: FRU Control applicable target FRUs

EXAMPLE

Perform a cold reset of the payload on the AMC located with FRU ID 12:

```
# fru_control –f 12 -c
```

Reboot the payload of the AMC (site type 07h) located at site number 2:

```
# fru_control –t 7 –n 2 --graceful-reboot
```

5.6.4 Get Address Information

NAME

`get_address_info` – display the FRU address information

SYNOPSIS

```
get_address_info        [[-a|--ipmb-address IPMB_ADDRESS]
                         {-f|--fru-id FRU_ID}]

get_address_info        [{-t|--site-type SITE_TYPE}
                         {-n|--site-number SITE_NUMBER}]
```

DESCRIPTION

This command can be used to determine the hardware address, IPMB address, and FRU ID of a FRU at a known site. It can also be used to determine the site type and number of a FRU at a known address. It also provides the Bus, Channel, MCH location, and Carrier number related to a given FRU.

The user may specify the target FRU using the IPM Controller’s **IPMB Address** and the **FRU Identifier**. If a target is not specified, the command displays the information of the FRU hosting the MicroTCA Carrier Manager.

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

The command displays the following information:

- Active MCH site number
- Active MCH IPMB address
- Device channel; if no device specified, defaults to 0
- Device bus ID; if no device specified, defaults to 0
- Device IPMB address and FRU ID; defaults to 0 and FFh, respectively, if no device is specified
- Device site type; defaults to 0 if no device is specified
- Device site ID; defaults to 0 if no device is specified
- Device associated Carrier number; defaults to 0 if no device is specified

APPLICABLE TARGET FRUS

Entity	Access Address	Slave Address(es)	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	20h	0	N/A	N/A
MCH	20h	10h – 12h	3 – 4	0Ah	1 – 2
AMC	20h	72h – 90h	5 – 16	07h	1 – 12
CU	20h	A8h – Aah	40 – 41	04h	1 – 2
PM	20h	C2h – C8h	50 – 53	0Bh	1 – 4

Table 18: Address Info applicable target FRUs

EXAMPLE

View the address information common for all address requests

```
# get_address_info
```

View the address information for a Power Module located with FRU ID 50:

```
# get_address_info –f 50
```

View the address information for an MCH (site type 0Ah) located at site number 1:

```
# get_address_info –n 1 –t 0xA
```

View the address information for an AMC at slave address 76h, FRU ID 0:

```
# get_address_info –a 0x76 f 0
```

SEE ALSO

```
get_location_info
```

5.6.5 Get Device Identifier

NAME

`get_device_id` – display the FRU device identifier information

SYNOPSIS

```
get_device_id           [[-a|--ipmb-address IPMB_ADDRESS]
                         {-f|--fru-id FRU_ID}]

get_device_id           [{-t|--site-type SITE_TYPE}
                         {-n|--site-number SITE_NUMBER}]
```

DESCRIPTION

The command displays the following information about the FRU:

- FRU device identifier
- Device revision
- Firmware revision
- IPMI version
- Device support flags
- Manufacturer ID
- Product ID

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 19: Get Device ID applicable target FRUs

EXAMPLE

View device information for the AMC located with FRU ID 7:

```
# get_device_id –f 7
```

View device information for an AMC (site type 07h) at site 2:

```
# get_device_id –-site-type 0x7 –-site-number 2
```

SEE ALSO

```
get_module_info
```

5.6.6 Get Event Receiver Location

NAME

`get_event_receiver` – display the location of the event receiver

SYNOPSIS

```
get_event_receiver      [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}

get_event_receiver      {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
```

DESCRIPTION

This command is used to display the location of the event receiver. The IPM Controller sends all event messages to the assigned event receiver. This command is only applicable to IPM Controllers that act as IPMB event generators.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

If event generation is enabled, by default the event receiver is assigned to FRU 0 at IPMB address 20h.

The command displays the location of the event receiver by IPMB address and Logical Unit Number (LUN).

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 20: Get Event Receiver applicable target FRUs

EXAMPLE

View the address of the event receiver for the AMC with FRU ID 5:

```
# get_event_receiver –f 5
```

View the address of the event receiver for the AMC (site type 07h) at site number 3:

```
# get_event_receiver –t 0x7 –n 3
```

SEE ALSO

set_event_receiver

5.6.7 Get FRU Activation Policy

NAME

`get_fru_activation_policy` – display current FRU activation policy

SYNOPSIS

```
get_fru_activation_policy   [[-a|--ipmb-address IPMB_ADDRESS]
                             {-f|--fru-id FRU_ID}]

get_fru_activation_policy   [{-t|--site-type SITE_TYPE}
                             {-n|--site-number SITE_NUMBER}]
```

DESCRIPTION

This command is used to display the current FRU activation policy. The activation policy is used during certain state transitions. The activation-locked bit is a software equivalent of the handle switch. The FRU cannot proceed from state M1 to M2 if the activation-locked bit is set. The deactivation-locked bit indicates whether an extraction condition exists. The FRU cannot proceed from state M4 to M5 if the deactivation-locked bit is set.

The user may specify a target FRU by using the IPM Controller's **IPMB Address** and **FRU Identifier.** If a target is not specified, the command displays information of the FRU hosting the MicroTCA Carrier Manager.

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

The command displays the following information:

- Activation-locked, whether enabled or disabled
- Deactivation-locked, whether enabled or disabled

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)s
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 21: FRU Activation Policy applicable target FRUs

EXAMPLE

Display the FRU activation policy for the Power Module located at FRU 50:

```
# get_fru_activation_policy –f 50
```

Display the FRU activation policy for the Power Module (site type 0Bh) located at site number 2:

```
# get_fru_activation_policy –t 0xB –n 2
```

SEE ALSO

```
get_fru_activation_sequence
get_power_management_info
```

5.6.8 Get FRU Power Levels

NAME

`get_fru_power_levels` – display the power levels information for a given FRU

SYNOPSIS

```
get_fru_power_levels    [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}

get_fru_power_levels    {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
```

DESCRIPTION

This command is used to display the current power level of a given FRU.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

The command displays the following power level information for a given FRU:

- Location
- Power draw information:
- Current power level (1 – ON, 0 – OFF)
- Power draw, in watts

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 22: FRU Power Levels applicable target FRUs

EXAMPLE

Display the power level for the Fan Tray with FRU ID 40:

```
# get_fru_power_levels –f 40
```

Display the power levels for the AMC (site type 07h) located at site number 2:

```
# get_fru_power_levels –t 0x7 -n 2
```

SEE ALSO

`set_fru_power_level`

5.6.9 Get FRU Hot-Swap State

`get_fru_state` – display the hot-swap information for a given FRU

SYNOPSIS

```
get_fru_state           [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}

get_fru_state           {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
```

DESCRIPTION

This command is used to display the hot-swap information of a given FRU.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

The command displays the following information:

- Physical address
- Module name
- Module entity ID/instance
- Current hot-swap state and reason for last state change

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 23: FRU State applicable target FRUs

EXAMPLE

Display the hot-swap information for the AMC with FRU ID 7:

```
# get_fru_state –f 7
```

Display the hot-swap information for the Fan Tray (site type 04h) located at site number 2, site type Cooling Unit:

```
# get_fru_state –t 4 –n 2
```

SEE ALSO

```
list_frus_present
```

5.6.10 Get FRU Temperature Status

NAME

`get_fru_temperature` – display the status of all temperature sensors for a given FRU

SYNOPSIS

```
get_fru_temperature     [[-a|--ipmb-address IPMB_ADDRESS]
                         {-f|--fru-id FRU_ID}]
                        [-T|--threshold]

get_fru_temperature     [{-t|--site-type SITE_TYPE}
                         {-n|--site-number SITE_NUMBER}]
                        [-T|--threshold]
```

DESCRIPTION

This command is used to display the status of all the temperature sensors.

The user may specify a target FRU by using the IPM Controller's **IPMB Address**. If a target is not specified, the command displays information for all FRUs in the current MicroTCA Carrier.

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

The command displays the following information for each FRU:

- FRU location
- Device name
- Sensor number
- Current sensor reading in degree centigrade

Specifying the **Threshold** option will also display the following:

- Current threshold status:
- Normal
- LowerNonCritical
- LowerCritical
- LowerNonRecoverable
- UpperNonCritical
- UpperCritical
- UpperNonRecoverable

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 24: Get FRU Temperature applicable target FRUs

EXAMPLE

Display the temperature status for the AMC with FRU ID 6:

```
# get_fru_temperature –f 6
```

Display the temperature status for the MCH (site type 04h) located at site number 2:

```
# get_fru_temperature –t 0x04 –n 2
```

Display the temperature status for all the FRUs installed on the Carrier:

```
# get_fru_temperature
```

SEE ALSO

```
get_sensor_reading
```

5.6.11 Get Module Information

NAME

`get_module_info` – display configuration and hot-swap information for an AMC Module

SYNOPSIS

```
get_module_info         [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}

get_module_info         {-n|--site-number SITE_NUMBER}
                        {-t|--site-type SITE_TYPE}
```

DESCRIPTION

This command displays configuration and hot-swap information for a given AMC Module. The information is collected from several sources, including the Module's FRU Inventory and the ATCA "Get Device ID" command.

The command displays the following information:

- Location of the Module identified by its physical and logical slot number and IPMB and physical addresses
- Module's device name, device ID, and entity ID / instance (optional)
- Module's FRU ID (normally 0)
- Module's current hot-swap state
- Revision information

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 25: Get Module Information applicable target FRUs

EXAMPLE

Display Module information for the Power Module (site type 0Bh) at site number 1:

```
# get_module_info –n 1 –t 0xb
```

Display Module information for the AMC with FRU ID 12:

```
# get_module_info –f 12
```

SEE ALSO

`list_modules_present`

5.6.12 HPM

NAME

hpm – upgrade firmware or check version information

SYNOPSIS

```
hpm                     {-a|--ipmb-address IPMB_ADDRESS}
                        {-f|--fru-id FRU_ID}
                        {-v|--version}

hpm                     {-a|--ipmb-address IPMB_ADDRESS}
                        {-f|--fru-id FRU_ID}
                        {-F|--file HPM FILE}
```

DESCRIPTION

This command uses the PICMG HPM.1 protocol to upgrade FRUs and check version information.

When used with the –v option, this command displays the following information:

The name and version of each upgradeable component

When used with the –F option, this command will upgrade the target as directed by the **HPM File.** The target device information is compared to the file, and the upgrade will not proceed if they are not compatible. Progress is displayed during the upgrade.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

APPLICABLE TARGET FRUS

Entity	Access Address(es)	FRU ID(s)
AMC	20h	5 – 16
CU	20h	40 – 41
PM	20h	50 – 53

Table 26: hpm applicable target FRUs

EXAMPLE

Show the version information for the Power Module at FRU ID 51:

```
# hpm –a 0x20 –f 51 -v
```

Upgrade the AMC at FRU ID 5:

```
# hpm –a 0x20 –f 5 –F Upgrade_AMC000_2.1.0.hpm
```

5.6.13 List Device SDRs

NAME

`list_device_sdr` – display a list of SDRs in a given MicroTCA Carrier Device SDR Repository.

SYNOPSIS

```
list_device_sdr         [[-a|--ipmb-address IPMB_ADDRESS]
                         {-f|--fru-id FRU_ID}]

list_device_sdr         [{-t|--site-type SITE_TYPE}
                         {-n|--site-number SITE_NUMBER}]
```

DESCRIPTION

This command is used to obtain a list of the SDRs in a given FRU's SDR Device Repository.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

The command displays the following information for each SDR:

- Record ID
- Version
- Length of the record
- Entity ID/instance
- Type of record
    - FS – Full sensor
    - CS – Compact sensor
    - EO – Entity only sensor
    - EA – Entity association
    - DREA- Device-Relative entity association
    - GDL – Generic device locator
    - FRUDL – FRU device locator
    - MCDL – Management device locator
    - MCC – Management Controller confirmation
    - BMC – Message channel information
    - OEM
    - Device name

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 27: List Device SDR applicable target FRUs

EXAMPLE

List the SDRs from the repository on the Power Module with FRU ID 51:

```
# list_device_sdr –f 51
```

List the SDRs from the repository on the AMC (site type 07h) at site number 8:

```
# list_device_sdr –t 0x7 –n 8
```

SEE ALSO

```
list_sdr
list_sensors
```

5.6.14 List FRU Storage Devices

NAME

`list_fru_storages` – display a list of all the FRU Inventory Devices located in the MicroTCA Carrier

SYNOPSIS

```
list_fru_storages
```

DESCRIPTION

This command is used to obtain a list of accessible FRU Inventory Devices.

The command displays the following information for each FRU Inventory Device:

- Location
- Entity ID/instance
- Device name

SEE ALSO

```
read_fru_storage
```

5.6.15 List SDRs

NAME

`list_sdr` – display a list of SDRs in the MicroTCA Carrier's SDR Repository.

SYNOPSIS

```
list_sdr
```

DESCRIPTION

This command displays all the merged SDRs in the MicroTCA Carrier SDR Repository along with the SDRs belonging to FRU address 20h.

The command displays the following information for each SDR:

- Record ID
- Version
- Length of the record
- Entity ID/instance
- Type of record
  - FS – Full sensor
  - CS – Compact sensor
  - EO – Entity only sensor
  - EA – Entity association
  - DREA- Device-Relative entity association
  - GDL – Generic device locator
  - FRUDL – FRU device locator
  - MCDL – Management device locator
  - MCC – Management Controller confirmation
  - BMC – Message channel information
  - OEM
- Device name

SEE ALSO

```
list_device_sdr
list_sensors
```

5.6.16 Read FRU Inventory Device

NAME

`read_fru_storage` – display content from a FRU Inventory Device

SYNOPSIS

```
read_fru_storage        [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        [-B|--board-info]
                        [-P|--product-info]
                        [-C|--chassis_info]
                        [-M|--multi-record]
                        [-T|--multi-record-type MULTI_RECORD_TYPE]
                        [-r|--dump-raw]

read_fru_storage        {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
                        [-B|--board-info]
                        [-P|--product-info]
                        [-C|--chassis_info]
                        [-M|--multi-record]
                        [-T|--multi-record-type MULTI_RECORD_TYPE]
                        [-r|--dump-raw]
```

DESCRIPTION

This command is used to read the content from a FRU Inventory Device.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The user may also specify a target FRU by using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

By default, all sections of the FRU Inventory Device are displayed.

Board Info displays the Board Information Area.

Product Info displays the Product Information Area.

Chassis Info displays the Chassis Information Area.

Multi-Record displays the Multi-Record Information Area.

Multi-Record Type displays records with a given PICMG Record ID within the Multi-Record Information Area. Refer to the PICMG specifications for Record ID Definitions,

Dump Raw will also include the information in raw format. This option is only applicable for information from the Multi-Record Information Area.

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
Physical Shelf FRU Info 1	20h	1	N/A	N/A
Physical Shelf FRU Info 2	20h	2	N/A	N/A
Logical Carrier FRU Info	20h	253	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 28: Read FRU Inventory Device applicable target FRUs

EXAMPLE

Read the product information from FRU Information on the AMC (site type 07h) located at site number 3:

```
# read_fru_storage –t 0x7 –n 3 -P
```

Read OEM PICMG multi-records of type 4 (Backplane Point-to-Point Connectivity Record), displaying information in raw format. The records will be read from FRU Information on the Carrier Manager (FRU ID 0):

```
# read_fru_storage –f 0 –T 4 -r
```

SEE ALSO

```
list_fru_storages
```

5.6.17 Set Event Receiver

NAME

`set_event_receiver` – change the location of a FRU's event receiver

SYNOPSIS

```
set_event_receiver      [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        {-r|--rx-address RECEIVER_IPMB_ADDRESS}
                        {-l|--lun LUN}

set_event_receiver      {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
                        {-r|--rx-address RECEIVER_IPMB_ADDRESS}
                        {-l|--lun LUN}
```

DESCRIPTION

This command is used to change the location of the event receiver. The IPM Controller sends each event message to the assigned event receiver. This command is only applicable to IPM Controllers that act as IPMB event generators.

The target is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The user may also specify a target FRU by using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

The location of the event receiver is identified by **Receiver IPMB address** and **Logical Unit Number (LUN).**

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 29: Set Event Receiver applicable target FRUs

EXAMPLE

Change the location of the event receiver to 20h, LUN 0 for the AMC with FRU ID 5:

```
# set_event_receiver –f 5 –r 0x20 –l 0
```

Change the location of the event receiver for the Power Module (site type 0Bh) at site number 2, LUN 0 to 20h:

```
# set_event_receiver –t 0xB –n 2 –r 0x20 –l 0
```

SEE ALSO

`get_event_receiver`

5.6.18 Set FRU Extracted

NAME

`set_fru_extracted` – inform the MicroTCA Carrier Manager that the given Power Module is no longer installed

SYNOPSIS

```
set_fru_extracted       [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}

set_fru_extracted       {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
```

DESCRIPTION

This command is used to inform the Carrier Manager that the FRU at a given address is no longer installed. All the information pertaining to this FRU is removed by the Carrier Manager. This command is only supported for extracting Power Modules.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 30: FRU Extraction applicable target FRUs

EXAMPLE

Set the Power Module with FRU ID 51 as extracted:

```
# set_fru_extracted –f 51
```

Set the Power Module (site type 0Bh) located at site number 2 as extracted.

```
# set_fru_extracted ––site-type 0xB --site-number 2
```

SEE ALSO

```
list_frus_present
```

5.6.19 Set FRU Power Level

NAME

`set_fru_power_level` – set the FRU power level for the given FRU

SYNOPSIS

```
set_fru_power_level     [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        {-l|--power-level POWER_LEVEL}

set_fru_power_level     {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
                        {-l|--power-level POWER_LEVEL}
```

DESCRIPTION

This command is used to set the power level of the FRU specified.

The target FRU is specified using the IPM Controller’s **IPMB Address** and the **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

The **Power Level** is a number used to specify which steady power draw, as returned by the get_fru_power_levels command, should be used. The user may set the power level between 0 (off) and the maximum level.

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 31: Set FRU Power Levels applicable target FRUs

EXAMPLE

Power off the AMC with FRU ID 10:

```
# set_fru_power_level –f 10 –l 0
```

Power on the AMC (site type 07h) located at site number 2:

```
# set_fru_power_level –t 0x7 –n 2 –l 1
```

SEE ALSO

```
get_fru_power_levels
```

5.6.20 Update FRU Version

NAME

`update_fru_version` – change the product version number for the given FRU

SYNOPSIS

```
update_fru_version      [-a|--ipmb-address IPMB_ADDRESS]
                        [-f|--fru-id FRU_ID]
                        {-v|--version VERSION}

update_fru_version      [-t|--site-type SITE_TYPE]
                        [-n|--site-number SITE_NUMBER]
                        {-v|--version VERSION}
```

DESCRIPTION

This command is used to change the product version number, as stored in the FRU Inventory Device, of the FRU specified.

The user may specify a target FRU by using the IPM Controller's **IPMB Address** and the **FRU Identifier**. If a target is not specified, the command changes the product version number of the MicroTCA Carrier Manager.

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

If successful, the product version number will be changed to the value specified by **Version**.

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 32: Update FRU Version applicable target FRUs

EXAMPLE

Change the product version number of the Power Module (site type 0Bh) located at site number 2 to ‘PM1.3’:

```
# update_fru_version –t 0xB –n 2 –v PM1.3
```

Change the product version number of the Cooling Unit with FRU ID 40, to ‘CU1.0’:

```
# update_fru_version –f 40 –v CU1.0
```

Change the product version number of the Carrier Manager to 1.2.2:

```
# update_fru_version –-version 1.2.2
```

SEE ALSO

read_fru_storage, write_fru_storage

5.6.21 Write FRU Storage

NAME

`write_fru_storage` – write data to the FRU Information for the given FRU

SYNOPSIS

```
update_fru_version      {-a|--ipmb-address IPMB_ADDRESS}
                        {-f|--fru-id FRU_ID}
                        {-F|--file FRU FILE}
```

DESCRIPTION

This command is used to change the data stored in the FRU Inventory Device, of the FRU specified.

The user may specify a target FRU by using the IPM Controller's **IPMB Address** and the **FRU Identifier**. This command writes the contents of **FRU File** to the target FRU, starting at the beginning of the FRU Information device. This operation is not reversible, and can interfere with the operation of the FRU. The FRU Information includes serial numbers and may include GUIDs that should be unique for every device, so the same file should not be written to two different FRUs. Do not use this command unless instructed to by the manufacturer of the target FRU. Normally, the target FRU will require a power cycle for the changes to take effect.

APPLICABLE TARGET FRUS

Entity	Access Address(es)	FRU ID(s)
Carrier Manager	20h	0
MCH	20h	3 – 4
AMC	20h	5 – 16
CU	20h	40 – 41
PM	20h	50 – 53

Table 33: write_fru_storage applicable target FRUs

EXAMPLE

Overwrite the FRU Information of the Power Module at FRU ID 50:

```
# write_fru_storage –a 0x20 –f 50 –F fru_3170028
```

SEE ALSO

`read_fru_storage`

5.6.22 Write SDR Repository

NAME

`write_sdr_repository` – update the SDRs in a given FRU's Device SDR Repository.

SYNOPSIS

```
write_sdr_repository    {-a|--ipmb-address IPMB_ADDRESS}
                        {-f|--fru-id FRU_ID}
                        {-F|--file-name SDR_FILE}
                        [-L|--sdr-alignment BYTES]
                        [-u|--updatemode]
```

DESCRIPTION

This command is used to update the SDRs in a given FRU's SDR Device Repository.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.** The SDRs to write are stored in the standard IPMI binary format in the **SDR File**. The alignment, in **Bytes**, specifies the alignment of SDRs in the SDR File. For example, with at 16-byte alignment, if an SDR ends on byte 17, the next SDR will not start until byte 32. If no alignment is specified, the SDRs in the file must be contiguous. The **Updatemode** option must be used to update FRUs that only provide a modal SDR.

This operation is not reversible, and can interfere with the operation of the FRU. Do not use this command unless instructed to by the manufacturer of the target FRU. The target FRU may require a power cycle for the changes to take effect.

APPLICABLE TARGET FRUS

Entity	Access Address(es)	FRU ID(s)
Carrier Manager	20h	0
MCH	20h	3 – 4
AMC	20h	5 – 16
CU	20h	40 – 41
PM	20h	50 – 53

Table 34: Write SDR Repository applicable target FRUs

EXAMPLE

Update the SDRs on the Power Module located at FRU ID 51:

```
# write_sdr_repository –a 0x20 –f 51 –F sdr –L 16
```

SEE ALSO

```
list_sdr
list_sensors
list_device_sdr
```

5.7 LAN

5.7.1 Get Channel Access

NAME

`get_channel_access` – display whether channel is enabled or disabled, whether alerting is enabled or disabled, and under which system modes the channel can be accessed

SYNOPSIS

```
get_channel_access      {-c|--channel-number CHANNEL_NUMBER}
                        [-V|--non-volatile]
```

DESCRIPTION

This command is used to check whether the channel is enabled or disabled, whether alerting is enabled or disabled, and under which system modes the channel can be accessed.

Channel Number is a 0-based index used to identify a messaging channel. To display information for the channel used for the active session, the value 0Eh may be used.

By default, the information is retrieved from volatile (active) storage. Specifying the **Non-volatile** option will retrieve information from non-volatile storage.

The command displays the following information:

- PEF alerting, whether it is enabled or disabled
- Per-message authentication, whether it is enabled or disabled
- User-level authentication, whether it is enabled or disabled
- Access mode
- Channel privilege level limit

EXAMPLE

View the active channel's access settings for the active session:

```
# get_channel_access –c 0xE
```

View the non-volatile channel access for channel 3:

```
            # get_channel_access –c 3 --non-volatile
```

SEE ALSO

```
      get_channel_info
      get_channel_cypher_suites
      set_channel_access
```

5.7.2 Get Channel Cipher Suites

NAME

`get_channel_cipher_suites` – display supported authentication, integrity, and confidentiality algorithms

SYNOPSIS

```
get_channel_cipher_suites   {-c|--channel-number CHANNEL_NUMBER}
                            [-A|--all]
```

DESCRIPTION

This command is used to look up which authentication, integrity, and authentication algorithms are supported for a given LAN channel that supports RMCP+ sessions. This allows the remote console to determine which authentication, integrity, and confidentiality algorithms can be used for establishing the connection. The algorithms are used in combinations called Cipher Suites. This command only applies to IPMI v2.0/RMCP+ sessions.

Channel Number is a 0-based index used to identify a messaging channel. To display information for the channel used for the active session, the value 0Eh may be used.

By default, only cipher suites using supported algorithms are displayed. To display all cipher suites, specify the **All** option.

The command displays the following information:

- Cipher suite ID
- OEM IANA, if applicable
- Authentication algorithm
- Integrity algorithms
- Confidentiality algorithms

EXAMPLE

View all cipher suites on channel 3:

```
# get_channel_cipher_suites –c 3 --all
```

View supported cipher suites on channel 2:

```
# get_channel_cipher_suites –c 2
```

View supported cipher suites on the channel used for the active session:

```
# get_channel_cipher_suites –c 0xE
```

5.7.3 Get Channel Information

NAME

`get_channel_info` – display media and protocol information about a given channel

SYNOPSIS

```
get_channel_info        {-c|--channel-number CHANNEL_NUMBER}
```

DESCRIPTION

This command is used to look up media and protocol information about a given channel. The user can use this command in combination with `get_session_info` to obtain the address of parties with open sessions and their present privilege levels.

Channel Number is a 0-based index used to identify a messaging channel. To display information for the channel used for the active session, the value 0Eh may be used.

The command displays the following information:

- Channel number
- Channel medium type
- Channel protocol type
- Session support
- Session-less
- Single-session
- Multi-session
- Session-based
- Number of active sessions
- Vendor IANA enterprise number

EXAMPLE

View information about the channel used for the active session:

```
# get_channel_info –c 0xE
```

View information about channel 2:

```
# get_channel_info –c 2
```

SEE ALSO

```
get_channel_access
get_channel_cipher_suites
get_session_info
```

5.7.4 Get IP Connection Information

NAME

`get_ip_connection` – display system network configuration

SYNOPSIS

```
get_ip_connection
```

DESCRIPTION

This command is used to view the system network configuration with which an external application may communicate with the MicroTCA Carrier Manager, and how the Carrier Manager may communicate with the MicroTCA Shelf Manager and MCH(s).

The command displays the following information:

- Shelf IP address
- Carrier IP address
- Gateway IP address 0
- Gateway IP address 1
- Subnet mask
- MCH 1 IP address
- MCH 2 IP address
- Username for the Carrier to use to communicate with the Shelf
- Password for the Carrier to use to communicate with the Shelf (not displayed for security purposes)

For each of the IP addresses and subnet mask above, a value of 0.0.0.0 indicates that no subnet mask is defined

SEE ALSO

`set_ip_connection`

5.7.5 Get LAN Configuration Parameters

NAME

`get_lan_config_parameters` – display parameters related to IPMI LAN operation, such as the network addressing information

SYNOPSIS

```
get_lan_config_parameters   {-c|--channel-number CHANNEL_NUMBER}
                            {-p|--parameter PARAMETER_SELECTOR}
                            [-s|--set SET_SELECTOR]
                            [-r|--dump-raw]

get_lan_config_parameters   {-c|--channel-number CHANNEL_NUMBER}
                            [-R|--revision-only]
```

DESCRIPTION

This command displays the parameters, such as the network addressing information, required for IPMI LAN operation.

Specifying the **Revision Only** option will display the parameters' revision.

Specifying the **Dump Raw** option will display the following, as well as the message response in raw hex format.

Channel Number is a 0-based index used to identify a messaging channel. To display information for the channel used for the active session, the value 0Eh may be used.

Parameter Selector is a 0-based index used to identify a setting parameter. The following parameters are supported:

0 = Set in progress. This parameter is used to indicate when LAN parameters are being updated, and when the updates are completed.
- Complete
- In progress

1 = Authentication type support. This displays the possible authentication types (algorithms) that are available for the given channel.
- MD2
- MD5
- Straight password/key
- OEM proprietary

2 = Authentication type enables. This displays the authentication types (algorithms) that are available for use when a remote session activates a connection for a requested maximum privilege level.

3 = IP address

4 = IP address source

- Static address (manually configured)
- Address obtained by BMC running DHCP
- Address loaded by BIOS or system software
- Address obtained by BMC running other address assignment protocol
- Unspecified

5 = MAC address

6 = Subnet mask

7 = IPv4 header parameters

- Time-to-live parameter in IP header for RMCP packets and PET traps transmitted from this channel
- Value of bit 1 in Flags field in the IP header for packets transmitted by this channel
- Type of service
- Precedence

12 = Default gateway address

13 = Default gateway MAC address

14 = Backup gateway address

15 = Backup gateway MAC address

16 = Community string. This value is used to fill in the ‘Community String’ field in a PET trap.

17 = Number of destinations

18 = Destination type

- Destination type
- Whether alert is acknowledged
- Alert acknowledge timeout / retry interval
- Number of alert retries to given destination

19 = Destination addresses

- Address format
- Address
- Gateway, whether default or backup
- Alerting MAC address

20 = 802.1q VLAN ID

- 12-bit VLAN ID
- VLAN ID, whether enabled or disabled

21 = 802.1q VLAN priority. This value is used for Priority field of 802.1q fields.

22 = RMCP+ messaging cipher suite entry support. This parameter provides a count of the number of cipher suites available to be enabled for use with IPMI messaging on the given channel.

23 = RMCP+ messaging cipher suite entries. This parameter contains the cipher suite IDs for cipher suites that can be used for establishing an IPMI messaging session.

24 = RMCP+ messaging cipher suite privilege levels. This parameter assigns a maximum privilege level for each cipher suite.

- Callback
- User
- Operator
- Administrator
- OEM proprietary

25 = Destination address VLAN TAGs

- Whether VLAN TAG is used
- VLAN TAG

For the following parameters, the user may specify a target destination using the **Set Selector**. The set (destination) selector is a 0-based index. Destination 0 is always present as a volatile destination that is used with the `snmp_trap_test` command. If a target is not specified, the command displays information for all sets (destinations).

- 18 = Destination type
- 19 = Destination addresses
- 25 = Destination address VLAN TAGs

The command displays a given parameter setting, as described above.

EXAMPLE

View the primary RMCP port number (parameter 8) on the channel used for the active session:

```
# get_lan_config_parameters --channel-number 0xE --parameter 8
```

View all available destination addresses (parameter 19) on channel 3, and display the raw information, as well:

```
# get_lan_config_parameters –c 3 –p 19 -r
```

View the destination type (parameter 18) for the destination with index 5 on channel 2:

```
# get_lan_config_parameters –c 2 –p 18 –s 5
```

View the LAN configuration parameter revision for channel 4:

```
# get_lan_config_parameters –c 4 –R
```

SEE ALSO

```
    set_lan_config_parameters
```

5.7.6 Get Session Information

NAME

`get_session_info` – display session information

SYNOPSIS

```
get_session_info        [-I|--session-index SESSION_INDEX]

get_session_info        [-h|--session-handle SESSION_HANDLE]
```

DESCRIPTION

This command is used to get information regarding which users presently have active sessions, and, when available, addressing information for the user that has established the session.

The user may also specify a target session by **Session Index** or **Session Handle.** The session index is used to select the entry in the active session table, based on its location within the table. The session handle is a 1-based number used to identify a specific session. If a target is not specified, the command displays information about the active session.

The command displays the following information:

- Number of possible active sessions
- Number of currently active sessions

The command also displays the following information if an active session is found:

- Session handle
- User ID
- Operating privilege level
- Channel number
- IPMI/RMCP version (for 802.3 LAN channel types)
- IP address of remote console
- MAC address of remote console
- Port number of remote console

EXAMPLE

View information for the active session:

```
get_session_info
```

View information for the session with session index 3:

```
get_session_info –I 3
```

View information for the session with session handle 2:

```
get_session_info –h 2
```

SEE ALSO

```
get_channel_info
```

5.7.7 List Active Sessions

NAME

`list_active_sessions` – display a list of active sessions

SYNOPSIS

```
list_active_sessions
```

DESCRIPTION

This command is used to display a list of active sessions for a given channel.

The command displays the following information for each active session:

- Session Index
- Session handle
- Channel number
- IP address used by the session
- Port used by the session
- User ID
- Operating privilege level

SEE ALSO

`get_session_info`

5.7.8 Set Channel Access

NAME

`set_channel_access` – modify whether channel is enabled or disabled, whether alerting is enabled or disabled, and privilege level limit

SYNOPSIS

```
set_channel_access      {-c|--channel-number CHANNEL_NUMBER}
                        {[-A|--pef-enable 0|1]
                         [-M|--per-msg-auth 0|1]
                         [-U|--user-level-auth 0|1]
                         [-l|--privilege-limit PRIVILEGE_LEVEL]
                         [-V|--non-volatile]}
```

DESCRIPTION

Session-based channels can be configured to provide IPMI messaging access only when the system is in certain states. This allows the system user to configure various levels of security and remotely-accessible features. This command is used to enable/disable PEF alerting, per-message authentication, and user-level authentication, as well as changing the system mode.

Channel Number is a 0-based index used to identify a messaging channel. To display information for the channel used for the active session, the value 0Eh may be used.

By default, only the information in volatile (active) settings is changed. Specifying the **Non-volatile** option will change information for both volatile and non-volatile settings. The volatile (active) settings are overwritten from the non-volatile settings whenever the system is reset or powered on.

One or more of the following can be enabled (**1**) or disabled (**0**):

- **PEF Alerting.** Enabling PEF alerting has no effect when alerting is disabled on the system, as reported in get_snmp_trap_info.
- **Per-Message Authentication.** Whether authentication is required to activate any session
- **User-Level Authentication.** Whether all User Level commands are to be authenticated

A maximum **Privilege Level** can be set on a given channel. This is the maximum privilege level that can be accepted on the channel:

- 01h = Callback
- 02h = User
- 03h = Operator
- 04h = Administrator
- 05h = OEM

EXAMPLE

Enable PEF alerting and disable user-level authentication in volatile storage for channel 3:

```
# set_channel_access –c 3 –A 1 –U 0
```

Change the maximum privilege level to operator in non-volatile storage on the channel used for the active session:

```
# set_channel_access –l 3 –V –c 0xE
```

SEE ALSO

```
get_channel_access
get_channel_cipher_suites
get_snmp_trap_info
```

5.7.9 Set IP Connection Information

NAME

`set_ip_connection` – modify available network interfaces

SYNOPSIS

```
set_ip_connection       {[-s|--shelf-ip-address SHELF_IP_ADDRESS]
                         [-c|--carrier-ip-address CARRIER_IP_ADDRESS]
                         [-m|--mch1-ip-address MCH_1_IP_ADDRESS]
                         [-M|--mch2-ip-address MCH_2_IP_ADDRESS]
                         [-n|--netmask SUBNET_MASK]
                         [-g|--gw-address GATEWAY_ADDRESS_0
                         [-G|--gw-address1 GATEWAY_ADDRESS_1]
                         [-U|--username USERNAME]
                         [-P|--password PASSWORD]}
```

DESCRIPTION

This command modifies the network interfaces with which an external application may communicate with the MicroTCA Carrier Manager, and how the Carrier Manager may communicate with a MicroTCA Shelf and MCH(s).

The IP connection record consists of the **Carrier IP Address**, its **Gateway Address 0** and **1**, and **Subnet Mask**. It also contains the **Shelf IP Address**, **MCH 1 IP Address**, and **MCH 2 IP Address**, as well as the username and password to be used to connect to the Shelf. At least one of these values must be specified when adding or modifying a record.

If a value is specified, the record will be updated with the value. If a value is not specified, the current value remains.

The changes will take effect on the next power cycle.

EXAMPLE

Update the IP Link record to Carrier IP address 192.168.0.13, gateway address 0 10.1.0.33, and subnet mask 255.255.255.0:

```
# set_ip_connection –c 192.168.0.13 –g 10.1.0.33 –n 255.255.255.0
```

Switch the target Shelf Manager IP information, where the new Shelf IP address is 192.168.1.14, username root, password root:

```
# set_ip_connection –s 192.168.1.14 –U root –P root
```

Add a redundant MCH at 192.168.1.7:

```
# set_ip_connection –M 192.168.1.7
```

SEE ALSO

`get_ip_connection`

5.7.10 Set LAN Configuration Parameters

NAME

`set_lan_config_parameters` – modify parameters required for IPMI LAN operation, such as the network addressing information

SYNOPSIS

```
set_lan_config_parameters   {-c|--channel-number CHANNEL_NUMBER}
                            {-p|--parameter PARAMETER_SELECTOR}
                            {[-d|--data DATA] |
                             [-S|--string STRING]}
```

DESCRIPTION

This command modifies the parameters, such as the network addressing information, required for IPMI LAN operation.

Channel Number is a 0-based index used to identify a messaging channel. To display information for the channel used for the active session, the value 0Eh may be used.

Parameter Selector is a 0-based index used to identify a setting parameter. All changes are saved to both volatile and non-volatile inventories. The following parameters are supported:

0 = Set in progress. This parameter is used to specify when LAN parameters are being updated, and when the updates are completed.
- Complete
- In progress

2 = Authentication type enables. This parameter is used to specify the authentication types (algorithms) that are available for use when a remote session activates for each privilege level.
- Callback
- User
- Operator
- Administrator
- OEM proprietary

3 = IP address

4 = IP address source
- Static address (manually configured)
- Address obtained by BMC running DHCP
- Address loaded by BIOS or system software
- Address obtained by BMC running other address assignment protocol

- Unspecified

5 = MAC address
6 = Subnet mask
7 = IPv4 header parameters

- Time-to-live parameter in IP header for RMCP packets and PET traps transmitted from this channel
- Value of bit 1 in Flags field in the IP header for packets transmitted by this channel
- Precedence
- Type of service

12 = Default gateway address
13 = Default gateway MAC address
14 = Backup gateway address
15 = Backup gateway MAC address
16 = Community string. This parameter is used to fill in the 'Community String' field in a PET trap.
18 = Destination type

- Destination selector
- Destination type
- Whether alert is acknowledged
- Alert acknowledge timeout / retry interval
- Number of alert retries to given destination

19 = Destination addresses

- Destination selector
- Address format
- Address
- Gateway, whether default or backup
- Alerting MAC address

20 = 802.1q VLAN ID

- VLAN ID, whether enabled or disabled
- 12-bit VLAN ID

21 = 802.1q VLAN priority. This parameter is used for Priority field of 802.1q fields.
24 = RMCP+ messaging cipher suite privilege levels. This parameter assigns a maximum privilege level for each cipher suite.

- Callback
- User
- Operator
- Administrator
- OEM proprietary

25 = Destination address VLAN TAGs

- Destination selector
- Whether VLAN TAG is used
- VLAN TAG

Data is a variable length list of byte values in hexadecimal. The length varies by parameter. Refer to the "Set PEF Configuration Parameters Command" section of the Intelligent Platform Management Interface Specification v2.0. If specifying a community string, the String option may be used instead.

EXAMPLE

Change the primary RMCP port (parameter 8) to 623 on the channel used for the active session:

```
# set_lan_config_parameters –-channel-number 0xE --parameter 8 --
      data "6f 02"
```

Change the LAN alert destination IP address (parameter 19) to 10.1.12.33, using the backup gateway, for destination with index 5 on channel 2:

```
# set_lan_config_parameters –c 2 –p 19 –d "05 00 01 0a 01 0c 21
      00 00 00 00 00 00"
```

Change the community string (parameter 16) to "testing" on channel 3:

```
# set_lan_config_parameters –c 3 –p 16 –S testing
```

SEE ALSO

```
get_lan_config_parameters
```

5.7.11 Set Session Privilege Level

NAME

`set_session_privilege_level` – request ability to perform operations at a particular privilege level for the active session

SYNOPSIS

```
set_session_privilege_level   {-p|--privilege-level PRIVILEGE_LEVEL}
```

DESCRIPTION

This command requests the ability to perform operations at a particular privilege level for the active session. The command can only be used to set privilege levels that are less than or equal to the privilege level limit for the entire channel, regardless of the privilege level of the user making the request.

This command is only relevant for a CLI used in **Interactive Mode**. See Section: 2.3: Starting the CLI for more details on this mode.

Privilege Level can be one of the following:

- 00h = No change, just return privilege level
- 01h = Callback
- 02h = User
- 03h = Operator
- 04h = Administrator
- 05h = OEM

The command displays the new current operating privilege level for the active session.

EXAMPLE

Request operator privilege level:

```
# set_session_privilege_level –p 3
```

SEE ALSO

```
get_channel_access
get_session_info
get_user_access
```

5.8 LEDs

5.8.1 Get LED Color Capabilities

NAME

`get_led_color_capabilities` – display a list of LEDs and the colors supported by each for a given FRU

SYNOPSIS

```
get_led_color_capabilities   [-a|--ipmb-address IPMB_ADDRESS]
                             {-f|--fru-id FRU_ID}
                             [-i|--led-id LED_ID]

get_led_color_capabilities   {-t|--site-type SITE_TYPE}
                             {-n|--site-number SITE_NUMBER}
                             [-i|--led-id LED_ID]
```

DESCRIPTION

This command is used to display a list of LEDs and the colors supported by each for a given Management Controller.

The target FRU may be specified using the IPM Controller’s **IPMB Address** and **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h – CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

The command displays the following information for each LED at a given address:

- List of colors supported
- Default color when in local control state
- Default color when in override state

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 35: LED Color Capabilities applicable target FRUs

EXAMPLE

View a list of LEDs and the colors supported by each for the AMC (site type 07h) located at site number 5:

```
# get_led_color_capabilities –n 5 –t 0x7
```

View the colors supported by LED 3 on the Power Module with FRU ID 50:

```
# get_led_color_capabilities –f 50
```

SEE ALSO

```
get_led_state
```

5.8.2 Get LED Properties

NAME

`get_led_properties` – display a list of LEDs controlled by a given FRU

SYNOPSIS

```
get_led_properties      [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}

get_led_properties      [-t|--site-type SITE_TYPE]
                        [-n|--site-number SITE_NUMBER]
```

DESCRIPTION

This command is used to display a list of LEDs controlled by the given Management Controller. LEDs are separated into general status LEDs and application specific LEDs.

The target FRU may be specified using the IPM Controller's **IPMB Address** and **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

The command displays the following information:

- Name of all general status LEDs, if any; otherwise, specifies "None"
- Number of application specific LEDs

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 36: Get LED Properties applicable target FRUs

EXAMPLE

View the list of LEDs on the AMC with FRU ID 8:

```
# get_led_properties –f 8
```

View the list of LEDs for the AMC (site type 07h) located at site number 3:

```
# get_led_properties –t 0x7 –n 3
```

SEE ALSO

```
get_led_color_capabilities
get_led_state
```

5.8.3 Get LED State

NAME

`get_led_state` – display the state for a given LED

SYNOPSIS

```
get_led_state           [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        [-i|--led-id LED_ID]

get_led_state           {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
                        [-i|--led-id LED_ID]
```

DESCRIPTION

This command is used to display the state of a FRU's LED(s), including whether a Lamp Test or Override State are enabled for it.

The target FRU may be specified using the IPM Controller's **IPMB Address** and **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

LED ID is a 0-based number used to identify an LED. The user may specify a target LED. If a target is not specified, the command displays information for all LEDs on a given Management Controller.

The command displays the following information for each LED:

- Whether local control state is supported
- Lamp test duration, if lamp test is enabled
- Override state and color, if override is enabled
- On and off duration, if override (blinking) is enabled
- Local state and color

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 37: Get LED State applicable target FRUs

EXAMPLE

View the state of LED 1 on the AMC with FRU ID 8:

```
# get_led_state –f 8 –i 1
```

View the state of all LEDs on the MCH with FRU ID 4:

```
# get_led_state –f 4
```

View the state of all LEDs on the MCMC (site type 0Ah) located at site number 2:

```
# get_led_state –t 0xA –n 2
```

SEE ALSO

`set_led_state`

5.8.4 Set LED State

NAME

`set_led_state` – set the state for a given LED

SYNOPSIS

```
set_led_state           [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        {-i|--led-id LED_ID}
                        {-o|--on-override}
                        [-c|--color COLOR]

set_led_state           [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        {-i|--led-id LED_ID}
                        {-x|--off-override}

set_led_state           [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        {-i|--led-id LED_ID}
                        {-b|--blink}
                        {-D|--off-duration OFF_DURATION}
                        {-U|--on-duration ON_DURATION}
                        [-c|--color COLOR]

set_led_state           [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        {-i|--led-id LED_ID}
                        {-l|--lamp-test}
                        {-U|--on-duration ON_DURATION}

set_led_state           [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        {-i|--led-id LED_ID}
                        {-r|--restore-local}

set_led_state           {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
                        {-i|--led-id LED_ID}
                        {-o|--on-override}
                        [-c|--color COLOR]

set_led_state           {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
                        {-i|--led-id LED_ID}
                        {-x|--off-override}
```

```
set_led_state           {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
                        {-i|--led-id LED_ID}
                        {-b|--blink}
                        {-D|--off-duration OFF_DURATION}
                        {-U|--on-duration ON_DURATION}
                        [-c|--color COLOR]

set_led_state           {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
                        {-i|--led-id LED_ID}
                        {-l|--lamp-test}
                        {-U|--on-duration ON_DURATION}

set_led_state           {-t|--site-type SITE_TYPE}
                        {-n|--site-number SITE_NUMBER}
                        {-i|--led-id LED_ID}
                        {-r|--restore-local}
```

DESCRIPTION

This command is used to set the state for a given LED.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

LED ID is a 0-based number used to identify an LED. The user must specify a target LED. For lamp tests, the value FFh may be used to test all LEDs controlled by a given Management Controller.

Only one of the following can be specified:

- **On Override** puts the LED in override state and turns on the LED.
- **Off Override** puts the LED in override state and turns of the LED.

Blinking puts the LED in override state and blinks the LED. The time the LED is on and off is specified by **On Duration** and **Off Duration**, in tens of milliseconds. The valid range of duration is 10ms – 2.5s.
Lamp Test turns on the LED for the time, in hundreds of milliseconds, specified in on duration. Valid range of duration is 100ms – 12.8s.
Restore Local puts the LED in local control state and sets the LED to the default local control state.

Color may be specified if the LED is in override on or blinking state.

- 1h = Blue
- 2h = Red
- 3h = Green
- 4h = Amber
- 5h = Orange
- 6h = White
- Eh = Do not change
- Fh = Use default color
- All other values reserved

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 38: Set LED State applicable target FRUs

EXAMPLE

Turn off LED 2 on the AMC with FRU ID 5:

```
# set_led_state –f 5 –i 2 -x
```

Turn on LED 1 using the color green (3) on the AMC with FRU ID 12:

```
# set_led_state –-fru-id 12 –-led-id 1 --on-override –-color 3
```

Blink LED 1 on the Fan Tray with FRU ID 40; 2 seconds on and 3 seconds off:

```
# set_led_state –f 40 –i 1 –b –U 20 –D 30
```

Lamp test LED 1 on the AMC with FRU ID 8 for 2 seconds:

```
# set_led_state –f 8 –i 1 –l –U 20
```

Restore LED 1 on the Power Module (site type 0Bh) located at site number 2:

```
# set_led_state –t 0xB –n 2 –i 1 –r
```

SEE ALSO

```
get_led_color_capabilities
get_led_state
```

5.9 Power Management

5.9.1 Get Power Channel Status

NAME

`get_power_channel_status` – display the status for the given power channel(s)

SYNOPSIS

```
get_power_channel_status  [-a|--ipmb-address IPMB_ADDRESS]
                          {-f|--fru-id FRU_ID}
                          [-c|--power-channel POWER_CHANNEL]

get_power_channel_status  {-n|--site-number SITE_NUMBER}
                          [-c|--power-channel POWER_CHANNEL]
```

DESCRIPTION

A MicroTCA Chassis and all its Modules are provided payload power by one or more Power Modules. This command is used to display information about the current status of the channels for which a Power Module is providing power.

The target Power Module is specified using the Power Module's IPM Controller's **IPMB Address** and the **FRU Identifier**.

The target Power Module may also be specified using the site number. **Site Number** is a 1-based index used to identify a Power Module in the MicroTCA Carrier Power Distribution and Management subsystem.

Power Channel is a number between 1 and 16 representing the power connection between a Power Module and any other Module on the Carrier. The user may specify a target power channel number. If a target is not specified, the command displays information for all power channels.

The command displays global power status and channel-specific power status:

- Power Module address
- Device Name
- Entity ID/Instance
- FRU state
- Power Module role
- Management power status (healthy/unhealthy)
- Payload power status (healthy/unhealthy)
- Count of power channels associated with this Power Module

For each power channel, the following information is displayed:

- Module name
- PS1# signal asserted/de-asserted
- Enable signal asserted/de-asserted
- Management power on/off and status (healthy/unhealthy)
- Payload power on/off and status (healthy/unhealthy)
- Channel power on/off
- Lastly, the command displays whether or not the redundant Power Module is providing payload power.

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Number(s)
PM	20h	50 – 53	1 – 4

Table 39: Power Channel Status applicable target FRUs

EXAMPLE

View power channel 5's status on the Power Module located at site number 1:

```
# get_power_channel_status –n 1 –c 5
```

View the status of all power channels on the Power Module with FRU ID 51:

```
# get_power_channel_status –f 51
```

SEE ALSO

```
get_power_feed_info
get_power_feed_status
get_power_management_info
get_power_policy
power_feed_control
power_feed_reset
```

5.9.2 Get Power Feed Information

NAME

`get_power_feed_info` – display the power information for a given Power Module

SYNOPSIS

```
get_power_feed_info     [-a|--ipmb-address IPMB_ADDRESS]
                        [-f|--fru-id FRU_ID]
                        [-d|--draws]

get_power_feed_info     [-n|--site-number SITE_NUMBER]
                        [-d|--draws]
```

DESCRIPTION

This command displays the amount of power being provided by the Power Modules in the MicroTCA Chassis.

The target Power Module is specified using the Power Module’s IPM Controller’s **IPMB Address** and the **FRU Identifier.**

The target Power Module may also be specified using the site number. **Site Number** is a 1-based index used to identify a Power Module in the MicroTCA Carrier Power Distribution and Management subsystem.

If a target is not specified, the command displays information for all Power Modules.

The command displays the following information:

- Power Module role (primary or redundant)
- Count of power channels associated with the respective Power Module
- Maximum internal current, in amps, that the internal MicroTCA Carrier circuitry can handle for this feed
- Maximum power, in Watts, available to the system
- Power, in Watts, that is currently available for powering MicroTCA Carrier FRUs
- Power, in Watts, that is currently being consumed by MicroTCA Carrier FRUs
- Feed to FRU mapping table that includes the FRU hardware address and FRU device ID. A value of FEh indicates that all FRU Device IDs at the hardware address is considered as a unit

If the **Draws** are requested:

- Maximum current (Amps) allotted to the respective power channels
- Current power draw (Watts) on the respective power channels

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Number(s)
PM	20h	50 – 53	1 – 4

Table 40: Power Management Info applicable target FRU

EXAMPLE

View power information for the Power Module located at site number 1:

```
# get_power_feed_info –n 1 -d
```

View power information, including draws, for the Power Module with FRU ID 50:

```
# get_power_feed_info –f 50 -d
```

View power information about all Power Modules in the Carrier:

```
# get_power_feed_info
```

SEE ALSO

```
get_power_channel_status
get_power_feed_status
get_power_management_info
get_power_policy
power_feed_control
power_feed_reset
```

5.9.3 Get Power Feed Status

NAME

`get_power_feed_status` – display the Power Module status for a given Power Module

SYNOPSIS

```
get_power_feed_status   [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}

get_power_feed_status   {-n|--site-number SITE_NUMBER}
```

DESCRIPTION

This command requires a redundant Power Module. The system will locate a secondary Power Module to query about a given Power Module.

The target Power Module may be specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target Power Module may also be specified using the site number. **Site Number** is a 1-based index used to identify a specific Power Module in the MicroTCA Carrier Power Distribution and Management subsystem.

The command displays the following information:

- FRU address
- Device name
- Entity ID/Instance
- FRU hot-swap state, previous state
- Role, whether primary or redundant
- Power Feed status (healthy/unhealthy)
- Management power status (healthy/unhealthy)
- Payload power status (healthy/unhealthy)
- Whether the redundant Power Feed is supplying power or not

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Number(s)
PM	20h	50 – 53	1 – 4

Table 41: Power Feed Status applicable target FRUs

EXAMPLE

View the Power Module status for the Power Module located at site number 1:

```
get_power_feed_status –n 1
```

View the Power Module status for the Power Module with FRU ID 50:

```
get_power_feed_status –f 50
```

SEE ALSO

```
get_power_channel_status
get_power_feed_info
get_power_management_info
get_power_policy
power_feed_control
power_feed_reset
```

5.9.4 Get Power Management Information

NAME

`get_power_management_info` – display FRU activation sequence

SYNOPSIS

```
get_power_management_info
```

DESCRIPTION

See `get_fru_activation_sequence`.

SEE ALSO

```
get_fru_activation_sequence
get_power_channel_status
get_power_feed_status
get_power_feed_info
power_feed_control
power_feed_reset
```

5.9.5 Get Power Policy

NAME

`get_power_policy` – display the power policy configuration

SYNOPSIS

```
get_power_policy
```

DESCRIPTION

This command displays the following information for all configured Power Feeds:

- Power Module role (primary or redundant)
- Site number
- Maximum current override (amps)
- Associated power channels

SEE ALSO

```
get_power_channel_status
get_power_feed_status
get_power_feed_info
get_power_management_info
power_feed_control
power_feed_reset
```

5.9.6 Power Feed Control

NAME

`power_feed_control` – send a Chassis control request to a given Power Module

SYNOPSIS

```
power_feed_control      [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}
                        {[-d|--power-down] |
                         [-u|--power-up] |
                         [-r|--hard-reset] |
                         [-i|--diagnostic-interrupt] |
                         [-s|--soft-shutdown]}

power_feed_control      {-n|--site-number SITE_NUMBER}
                        {[-d|--power-down] |
                         [-u|--power-up] |
                         [-r|--hard-reset] |
                         [-i|--diagnostic-interrupt] |
                         [-s|--soft-shutdown]}
```

DESCRIPTION

This command provides power control, reset, and diagnostic interrupt control of a given Power Module. The implementation of the command is optional, and may not be supported in some Power Modules.

The target Power Module may be specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target Power Module may also be specified using the site number. **Site Number** is a 1-based index used to identify a specific Power Module in the MicroTCA Carrier Power Distribution and Management subsystem.

Only one of the following can be specified:

- **Power Down** forces the system into the SOFT OFF state. This is for emergency management power down actions.
- **Power Up** can be used to power up the Power Module.
- **Hard Reset** can be used to pulse the system reset signal regardless of the power state.
- **Diagnostic Interrupt** will pulse a version of a diagnostic interrupt that goes directly to the processor. This is typically used to cause the operating system to do a diagnostic dump. This may not be available in some implementations.
- **Soft Shutdown** behaves similarly to power down.

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Number(s)
PM	20h	50 – 53	1 – 4

Table 42: Power Feed Control applicable target FRUs

EXAMPLE

Send a power-down request to the Power Module located at site number 2:

```
# power_feed_control –n 2 –d
```

Send a power-up request to the Power Module with FRU ID 50:

```
# power_feed_control –f 50 –-power-up
```

SEE ALSO

```
chassis_control
get_power_feed_status
get_power_feed_info
```

5.9.7 Power Feed Reset

NAME

`power_feed_reset` – resets a given Power Module

SYNOPSIS

```
power_feed_reset        {-f|--fru-id FRU_ID}
                        {-F|--reset-fru-id RESET_FRU_ID}

power_feed_reset        {-n|--site-number SITE_NUMBER}
                        {-N|--reset-site-number RESET_SITE_NUMBER}
```

DESCRIPTION

This command will result in the reset Power Module to be reset. This is done by sending the reset request to another present Power Module.

FRU ID and **Reset FRU ID** is the FRU identifier of a present Power Module and the Power Module to be reset, respectively. Alternatively, the Power Modules may be specified by site number. **Site Number** and **Reset Site Number** are 1-based indexed numbers used to identify specific Power Modules in the MicroTCA Carrier Power Distribution and Management subsystem.

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Number(s)
PM	20h	50 – 53	1 – 4

Table 43: Reset Power Feed applicable target FRUs

EXAMPLE

Reset the Power Module located at site number 2 with the Power Module loated at site number 1:

```
# power_feed_reset –n 1 –N 2
```

Reset the Power Module with FRU ID 51 with the Power Module with FRU ID 50:

```
# power_feed_reset –f 50 –F 51
```

SEE ALSO

```
get_power_feed_status
power_feed_control
```

5.10 System Event Log

5.10.1 Clear SEL

NAME

`clear_sel` – erase the contents of a System Event Log

SYNOPSIS

```
clear_sel               [-a|--ipmb-address IPMB_ADDRESS]
                        [-f|--fru-id FRU_ID]

clear_sel               [-t|--site-type SITE_TYPE]
                        [-n|--site-number SITE_NUMBER]
```

DESCRIPTION

This command is used to clear a System Event Log.

The user may specify a target FRU by using the IPM Controller's **IPMB Address** and the **FRU Identifier**. If a target is not specified, the command clears the SEL of the FRU hosting the MicroTCA Carrier Manager.

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 44: Clear SEL applicable target FRUs

EXAMPLE

Clear the SEL on the Carrier Manager:

```
# clear_sel
```

Clear the SEL on the Power Module (site type 0Bh) located at site number 1:

```
# clear_sel –t 0xB –n 1
```

Clear the SEL on the Fan Tray with FRU ID 40:

```
# clear_sel –f 40
```

SEE ALSO

```
get_sel
get_sel_info
```

5.10.2 Get SEL

NAME

`get_sel` – display the contents of a System Event Log

SYNOPSIS

```
get_sel                  [-a|--ipmb-address IPMB_ADDRESS]
                         [-f|--fru-id FRU_ID
                         [[-i|--item INDEX] |
                          [-S|--sensor-number SENSOR_NUMBER] |
                          [[-s|--sensor-type SENSOR_TYPE]
                           [-A|--fru-address GENERATING_IPMB_ADDRESS]]]
                         [-r|--dump-raw | [-v|--verbose]

get_sel                  [-t|--site-type SITE_TYPE]
                         [-n|--site-number SITE_NUMBER]
                         [[-i|--item INDEX] |
                          [-S|--sensor-number SENSOR_NUMBER] |
                          [[-s|--sensor-type SENSOR_TYPE]
                           [-A|--fru-address GENERATING_IPMB_ADDRESS]]]
                         [-r|--dump-raw | [-v|--verbose]
```

DESCRIPTION

This command is used to display the contents of a System Event Log. A System Event Log is a non-volatile repository for system events and certain system configuration information.

The user may specify a target FRU by using the IPM Controller's **IPMB Address** and the **FRU Identifier**. If a target is not specified, the command displays information of the FRU hosting the MicroTCA Carrier Manager.

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

By default, all entries of a System Event Log are displayed.

Sensor Type displays only the entries in a System Event Log that are generated by sensors matching a given sensor type.

Generating IPMB Address displays only the entries in a System Event Log that is generated by the FRU matching a given IPMB address.

Sensor Number displays only the entries in a System Event Log that are generated by sensors matching a given sensor number.

Index displays the System Event Log entry with an event ID that matches a given index.

Verbose will interpret events from hot-swap, temperature, voltage, and IPMB-0 status sensors.

Dump Raw will convert the values into the response's hexadecimal values.

The command displays the following information:

- Date/timestamp
- Event ID
- Generator FRU
- Event revision
- Sensor type/number
- Event direction
- Event type
- Event data 1
- Event data 2
- Event data 3

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 45: Get SEL applicable target FRUs

EXAMPLE

Display the MicroTCA Carrier's System Event Log:

```
# get_sel
```

Display entries from the System Event Log that are generated by hot-swap sensors:

```
# get_sel –s 0xF0
```

Display entries from the System Event Log that are generated by temperature sensors, with interpretation

```
# get_sel –s 0x1 -v
```

Display entries from the System Event Log that are generated by FRUs at IPMB address AAh, by temperature (type 01h) sensors, in raw format:

```
# get_sel –A 0xAA –s 1 -r
```

Display entries from the System Event Log on the Cooling Unit at FRU 41:

```
# get_sel –f 41
```

Display entries from the System Event Log on the Cooling Unit (site type 04h) located at site number 2:

```
# get_sel –t 0x4 –n 2
```

SEE ALSO

`get_sel_info`

5.10.3 Get SEL Information

NAME

`get_sel_info` – display information about a Management Controller's System Event Log

SYNOPSIS

```
get_sel_info            [-a|--ipmb-address IPMB_ADDRESS]
                        [-f|--fru-id FRU_ID]

get_sel_info            [-t|--site-type SITE_TYPE]
                        [-n|--site-number SITE_NUMBER]
```

DESCRIPTION

This command is used to display information about a System Event Log. A System Event Log is a non-volatile repository for system events and certain system configuration information.

The user may specify a target FRU by using the IPM Controller's **IPMB Address** and the **FRU Identifier**. If a target is not specified, the command displays information of the FRU hosting the MicroTCA Carrier Manager.

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

The command displays the following information:

- SEL version
- Number of events logged
- Available free space
- Timestamp of most recent addition
- Timestamp of most recent erasure
- Command support
- Delete SEL support
- Partial add SEL entry support
- Reserve SEL support
- SEL allocation information

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 46: Get SEL Information applicable target FRUs

EXAMPLE

Display the System Event Log information for the Carrier Manager:

```
# get_sel_info
```

Get the System Event Log information for the Power Module (site type 0Bh) located at site number 1:

```
# get_sel_info –t 0xB –n 1
```

Get the System Event Log information for the Fan Tray with FRU ID 40:

```
# get_sel_info –f 40
```

SEE ALSO

```
get_sel
```

5.11 Sensor Management

5.11.1 Get Sensor Event Generation Capabilities

NAME

`get_sensor_event_enable` – display sensor event generation capabilities

SYNOPSIS

```
get_sensor_event_enable   [-l|--lun LUN]
                          [-S|--sensor-number SENSOR_NUMBER]
                          [-s|--sensor-type SENSOR_TYPE]
```

DESCRIPTION

The Management Controller monitors the state of a number of sensors. When a change is detected, the sensor event status is updated and an event is generated if the sensor supports event generation. The process of updating the event status is referred to as sensor scanning. As long as scanning is enabled, the sensor event status will be updated. This is independent of whether Event Messages are enabled. This command is used to display the status of the sensor event generation capabilities and whether sensor scanning is enabled.

Individual sensor events may be enabled or disabled. For discrete sensors, assertion and de-assertion events may be generated on a state change. For threshold sensors, the following events may be generated:

- an assertion event upon exceeding a threshold
- a de-assertion event upon clearing a threshold

The user may specify a target sensor location by sensor. If a target is not specified, the command displays the readings for all the sensors in the MicroTCA Carrier.

Logical Unit Number (**LUN**) displays readings for sensors on a given LUN.

Sensor Number displays readings for sensors matching a given number.

Sensor Type displays readings for sensors matching a given type.

The command displays the following information:

- Sensor number/name
- Entity ID/Instance
- Sensor type
- Event/reading type
- Event messages, whether enabled or disabled
- Sensor scanning, whether enabled or disabled
- Assertion/de-assertion event masks and supported events

EXAMPLE

View event generation capabilities for sensor 10h:

```
# get_sensor_event_enable –S 0x10
```

View event generation capabilities for all temperature (type 01h) sensors:

```
# get_sensor_event_enable –-sensor-type 0x1
```

View event generation capabilities for all sensors on the Chassis:

```
# get_sensor_event_enable
```

SEE ALSO

```
get_sensor_info
set_sensor_event_enable
```

5.11.2 Get Sensor Hysteresis Values

NAME

`get_sensor_hysteresis` – display sensor hysteresis values

SYNOPSIS

```
get_sensor_hysteresis   [-l|--lun LUN]
                        [-S|--sensor-number SENSOR_NUMBER]
                        [-s|--sensor-type SENSOR_TYPE]
```

DESCRIPTION

This command is used to display the sensor hysteresis values, applicable only to threshold sensors. Threshold and hysteresis values are used to determine when assertion and de-assertion events are generated.

Threshold events are classified as either high-going or low-going. The differences between high-going and low-going threshold events are in which direction the reading needs to be going for an event to occur and how hysteresis affects when de-assertion events occur.

For a high-going threshold:

- an assertion event is generated when the reading is greater than or equal to the threshold
- a de-assertion event is generated when the reading is positive_going_hysteresis+1 below the threshold

For a low-going threshold:

- an assertion event is generated when the reading is less than or equal to the threshold
- a de-assertion event is generated when the reading is negative_going_hysteresis+1 above the threshold

The user may specify a target sensor location by sensor. If a target is not specified, the command displays the readings for all the sensors in the MicroTCA Carrier.

Logical Unit Number (**LUN**) displays readings for sensors on a given LUN.

Sensor Number displays readings for sensors matching a given number.

Sensor Type displays readings for sensors matching a given type.

The command displays the following information:

- Sensor number/name
- Entity ID/Instance
- Sensor type
- Event/Reading Type
- Positive-going hysteresis
- Negative-going hysteresis

EXAMPLE

View hysteresis for sensor 10h:

```
# get_sensor_hysteresis –S 0x10
```

View hysteresis for all temperature (type 01h) sensors:

```
# get_sensor_hysteresis –s 0x1
```

View hysteresis for all sensors on the MicroTCA Carrier:

```
# get_sensor_hysteresis
```

SEE ALSO

```
get_sensor_event_enable
get_sensor_info
get_sensor_reading
get_sensor_threshold
get_diagnostics
set_sensor_hysteresis
```

5.11.3 Get Sensor Information

NAME

`get_sensor_info` – display sensor information

SYNOPSIS

```
get_sensor_info         [-l|--lun LUN]
                        [-S|--sensor-number SENSOR_NUMBER]
                        [-s|--sensor-type SENSOR_TYPE]
                        [-r|--dump-raw]
```

DESCRIPTION

This command is used to display the sensor information.

The user may specify a target sensor location by sensor. If a target is not specified, the command displays the information for all the sensors in the MicroTCA Carrier.

Logical Unit Number (LUN) displays readings for sensors on a given LUN.

Sensor Number displays readings for sensors matching a given number.

Sensor Type displays readings for sensors matching a given type.

Dump Raw displays the information in raw format, without interpretation.

The command displays the following information:

- Sensor number/name
- Entity ID/Instance
- Sensor type
- Event/Reading type
- Sensor initialization
- Assertion/De-assertion event mask, if applicable
- Sensor units
- Analog data format
- Rate type/unit
- Modifier type/unit
- Percentage, yes or no

Also shown for compact sensor records:

- Number of sensors sharing the same sensor data record

The command displays the following information for each threshold sensor:

- Settable/reading mask
- Sensor direction
- Hysteresis

Also shown for full sensor records:

- Linearization
- M, B, K1, and K2 values
- Tolerance and accuracy
- Nominal reading, if specified
- Normal maximum, if specified
- Normal minimum, if specified
- Sensor maximum and minimum

The command displays the following information for each discrete sensor:

- Discrete reading mask

EXAMPLE

View information for sensor 10h:

```
# get_sensor_info –S 0x10
```

View information for sensors located on LUN 0:

```
# get_sensor_info –l 0
```

View information for all temperature (type 01h) sensors:

```
# get_sensor_info –-sensor-type 0x1
```

View information for all sensors on the MicroTCA Carrier:

```
# get_sensor_info
```

SEE ALSO

`get_sensor_reading`

5.11.4 Get Sensor Reading

NAME

`get_sensor_reading` – display sensor reading

SYNOPSIS

```
get_sensor_reading      [-l|--lun LUN]
                        [-S|--sensor-number SENSOR_NUMBER]
                        [-s|--sensor-type SENSOR_TYPE]
```

DESCRIPTION

This command is used to display the sensor reading for threshold-based sensors and the asserted states for discrete sensors.

The user may specify a target sensor location by sensor. If a target is not specified, the command displays the readings for all the sensors in the MicroTCA Carrier.

Logical Unit Number (LUN) displays readings for sensors on a given LUN.

Sensor Number displays readings for sensors matching a given number.

Sensor Type displays readings for sensors matching a given type.

The command displays the following information:

- Sensor number/name
- Entity ID/Instance
- Sensor type
- Event/Reading type
- Sensor initialization status
- Sensor scanning (enabled/disabled)
- Event messages (enabled/disabled)

The command displays the following information for each threshold sensor:

- Raw reading
- Processed reading
- Current threshold status
- Sensor thresholds

The command displays the following information for each discrete sensor:

- Current asserted state
- Sensor reading

EXAMPLE

View sensor reading for sensor 10h:

```
# get_sensor_reading –S 0x10
```

View sensor readings for sensors on LUN 0:

```
# get_sensor_reading –l 0
```

View sensor readings for all hot-swap (type F0h) sensors:

```
# get_sensor_reading –s 0xF0
```

View sensor readings for all sensors on the MicroTCA Carrier:

```
# get_sensor_reading
```

SEE ALSO

```
get_sensor_info
get_sensor_threshold
set_sensor_reading
```

5.11.5 Get Sensor Thresholds

NAME

`get_sensor_threshold` – display sensor threshold values

SYNOPSIS

```
get_sensor_threshold    [-l|--lun LUN]
                        [-S|--sensor-number SENSOR_NUMBER]
                        [-s|--sensor-type SENSOR_TYPE]
```

DESCRIPTION

This command is used to display the sensor threshold values, applicable only to threshold sensors. Threshold and hysteresis values are used to determine when assertion and de-assertion events are generated.

The Management Controller monitors the state of a number of sensors. When a change is detected, the sensor event status gets updated and, if the sensor supports event generation, an event is generated.

For threshold sensors, the severity of an event depends on the threshold that was triggered or cleared. There are three levels of thresholds: non-critical, critical, and non-recoverable. The corresponding severities are minor, major, and critical. Each event triggers an alarm of the corresponding severity. Use get_health and get diagnostics to view the alarms triggered and the sensors that caused the events.

An event is specified as either high-going or low-going. The differences in high-going and low-going threshold events are in which direction the reading needs to be going for an event to occur and how hysteresis affects when de-assertion events occur.

For a high-going threshold:

- an assertion event is generated when the reading is greater than or equal to the threshold
- a de-assertion event is generated when the reading is positive_going_hysteresis+1 below the threshold

For a low-going threshold:

- an assertion event is generated when the reading is less than or equal to the threshold
- a de-assertion event is generated when the reading is negative_going_hysteresis+1 above the threshold

The user may specify a target sensor location by sensor. If a target is not specified, the command displays the readings for all the sensors in the MicroTCA Carrier.

Logical Unit Number (LUN) displays readings for sensors on a given LUN.

Sensor Number displays readings for sensors matching a given number.

Sensor Type displays readings for sensors matching a given type.

- The command displays the following information:
- Sensor number/name
- Sensor type
- Entity ID/instance
- Threshold values
- Lower/upper non-critical
- Lower/upper critical
- Lower/upper non-recoverable

EXAMPLE

View threshold for sensor 10h:

```
# get_sensor_threshold –-sensor-number 0x10
```

View threshold for sensors located on LUN 0:

```
# get_sensor_threshold –l 0
```

View threshold for all temperature (type 01h) sensors:

```
# get_sensor_threshold –s 0x1
```

View thresholds for all sensors on the MicroTCA Carrier:

```
# get_sensor_threshold
```

SEE ALSO

```
get_sensor_event_enable
get_sensor_info
get_sensor_hysteresis
get_sensor_reading
get_diagnostics
set_sensor_hysteresis
set_sensor_threshold
```

5.11.6 List Sensors

NAME

`list_sensors` – display a list of sensors on the MicroTCA Carrier

SYNOPSIS

```
list_sensors            [-l|--lun LUN]
                        [-s|--sensor-type SENSOR_TYPE]
                        [-r|--reading]
```

DESCRIPTION

This command is used to obtain a list of sensors on the MicroTCA Carrier.

The **Logical Unit Number** (**LUN**) is used to list the sensors on a given LUN. If not specified, then sensors located on all the LUNs are displayed.

The **Sensor Type** is used to list the sensors that match a given type. If not specified, then sensors of all types are displayed.
The logical unit and sensor type can be used together to filter the sensor list.

The command displays the following information for each sensor:

- Location
- Entity ID/instance
- Sensor number/type
- Name

Using the **Reading** option displays the raw sensor reading for each sensor in the list.

EXAMPLE

View all temperature (type 01h) sensors on the MicroTCA Carrier

```
# list_sensors –s 0x1
```

View all sensors on LUN 0 and display the raw readings:

```
# list_sensors–l 0 -r
```

SEE ALSO

```
get_sensor_event_enable
get_sensor_hysteresis
get_sensor_info
get_sensor_reading
get_sensor_threshold
```

5.11.7 Set Sensor Event Enable

NAME

`set_sensor_event_enable` – set event generation capabilities for a given sensor

SYNOPSIS

```
set_sensor_event_enable   {-S|--sensor-number SENSOR_NUMBER}
                          {-l|--lun LUN}
                          {[-e|--event-messages 0|1]
                           [-s|--scanning 0|1]
                           [{-i|--selected-event-messages 0|1}
                            {[-A|--event-assertion-mask
                               ASSERTION_EVENT_MASK]
                             [-D|--event-deassertion-mask DE-
                               ASSERTION_EVENT_MASK]}]}
```

DESCRIPTION

The Management Controller monitors the state of a number of sensors. When a change is detected, the sensor event status gets updated and an event is generated if the sensor supports event generation. The process of updating the event status is referred to as sensor scanning. As long as scanning is enabled, the sensor event status will be updated. This is independent of whether Event Messages are enabled. This command is used to modify the event generation capabilities for a given sensor, as well as enable/disable sensor scanning.

Logical Unit Number (**LUN**) and **Sensor Number** specify which sensor threshold values are to be changed.

Sensor Scanning option for the sensor must be enabled (**1**) or disabled (**0**). If enabled, the sensor event status will be updated.

Event Messages option for the sensor must be enabled (**1**) or disabled (**0**). If enabled, events are generated when there is a change in the sensor event status. Event generation can also be modified on a per-event basis.

To change an individual sensor event capability, specify the **Assertion Event Mask** and/or **De-assertion Event Mask**. The mask is a two-byte value which can be specified in hexadecimal or decimal format. For the selected events, they will be enabled (**1**) or disabled (**0**) according to the **Selected Event Messages** option. For discrete sensors, an assertion and de-assertion event may be generated on a state change. For threshold sensors, the following events may be generated:

- an assertion event upon exceeding a threshold
- a de-assertion event upon clearing a threshold

Refer to the “Set Sensor Event Enable Command” section and “Generic Event/Reading Type Codes” table of the Intelligent Platform Management Interface Specification v2.0 for the assertion and de-assertion event mask bit-assignments.

EXAMPLE

Disable event generation and enable sensor scanning for sensor 10h located on LUN 0:

```
# set_sensor_event_enable –l 0 –S 0x10 –e 0 –s 1
```

Enable sensor scanning and event generation for sensor 10h located on LUN 1, but disable upper critical and upper non-recoverable going high event generation:

```
# set_sensor_event_enable–l 1 –S 0x10 -s 1 –e 1 –E 0 -D 0x0A00
```

SEE ALSO

```
get_sensor_event_enable
```

5.11.8 Set Sensor Hysteresis

NAME

`set_sensor_hysteresis` – set sensor hysteresis values for a given sensor

SYNOPSIS

```
set_sensor_hysteresis   {-l|--lun LUN}
                        {-S|--sensor-number SENSOR_NUMBER}
                        {[-P|--positive-hysteresis
                              POSITIVE_GOING_HYSTERESIS]
                         [-N|--negative-hysteresis
                              NEGATIVE_GOING_HYSTERESIS]}
```

DESCRIPTION

This command is used to set the sensor hysteresis values, applicable only to threshold sensors. Threshold and hysteresis values are used to determine when assertion and de-assertion events are generated. The ability to set hysteresis values is optional. Therefore, some FRUs may not support this command.

Threshold events are classified as either high-going or low-going. The differences between high-going and low-going threshold events are in which direction the reading needs to be going for an event to occur and how hysteresis affects when de-assertion events occur.

For a high-going threshold:

- an assertion event is generated when the reading is greater than or equal to the threshold
- a de-assertion event is generated when the reading is positive_going_hysteresis+1 below the threshold

For a low-going threshold:

- an assertion event is generated when the reading is less than or equal to the threshold
- a de-assertion event is generated when the reading is negative_going_hysteresis+1 above the threshold

Logical Unit Number (**LUN**) and **Sensor Number** specify which sensor threshold values are to be changed.

At least one hysteresis value must be specified. If either **Positive-Going** or **Negative-Going** hysteresis values are not specified, it will remain unchanged.

EXAMPLE

Change positive-going hysteresis to 3 for sensor 10h located on LUN 0:

```
# set_sensor_hysteresis –l 0 –S 0x10 –P 3
```

Change positive-going hysteresis to 5 and negative-going hysteresis to 6 for sensor 10h located on LUN 1:

```
# set_sensor_hysteresis –-lun 1 –-sensor-number 0x10 –-positive-
      hysteresis 5 –-negative-hysteresis 6
```

SEE ALSO

```
get_sensor_event_enable
get_sensor_hysteresis
get_sensor_info
get_sensor_reading
get_sensor_threshold
get_diagnostics
```

5.11.9 Set Sensor Thresholds

NAME

`set_sensor_threshold` – set sensor threshold values for a given sensor

SYNOPSIS

```
set_sensor_threshold    {-l|--lun LUN}
                        {-S|--sensor-number SENSOR_NUMBER}
                        {[-o|--lower-non-critical
                              LOWER_NON_CRITICAL_THRESHOLD]
                         [-c|--lower-critical LOWER_CRITICAL_THRESHOLD]
                         [-r|--lower-non-recoverable
                              LOWER_NON_RECOVERABLE_THRESHOLD]
                         [-O|--upper-non-critical
                              UPPER_NON_CRITICAL_THRESHOLD]
                         [-C|--upper-critical UPPER_CRITICAL_THRESHOLD]
                         [-R|--upper-non-recoverable
                              UPPER_NON_RECOVERABLE_THRESHOLD]}
```

DESCRIPTION

This command is used to set the sensor threshold values for a given threshold, applicable only to threshold sensors. Threshold and hysteresis values are used to determine when assertion and de-assertion events are generated.

The Management Controller monitors the state of a number of sensors. When a change is detected, the sensor event status is updated and an event is generated if the sensor supports event generation.

For threshold sensors, the severity of an event depends on the threshold that was triggered or cleared. There are three levels of thresholds: non-critical, critical, and non-recoverable. The corresponding severities are minor, major, and critical. Each event triggers an alarm of the corresponding severity. Use get_health and get_diagnostics to view the alarms triggered and the sensors that caused the events.

An event is specified as either high-going or low-going. The differences in high-going and low-going threshold events are in which direction the reading needs to be going for an event to occur and how hysteresis affects when de-assertion events occur.

For a high-going threshold:

- an assertion event is generated when the reading is greater than or equal to the threshold

- a de-assertion event is generated when the reading is positive_going_hysteresis+1 below the threshold

For a low-going threshold:

- an assertion event is generated when the reading is less than or equal to the threshold
- a de-assertion event is generated when the reading is negative_going_hysteresis+1 above the threshold

Logical Unit Number (LUN) and **Sensor Number** specify which sensor threshold values are to be changed.

At least one threshold value must be specified. If **Lower Non-Critical**, **Lower Critical**, **Lower Non-Recoverable**, **Upper Non-Critical**, **Upper Critical**, or **Upper Non-Recoverable** threshold values are not specified, it will remain unchanged.

EXAMPLE

Change upper critical threshold to 80 for sensor 10h located on LUN 0:

```
# set_sensor_threshold –l 0 –S 0x10 –C 80
```

Change lower non-recoverable threshold to 20 and upper non-recoverable to 100 for sensor 10h located on LUN 0:

```
# set_sensor_threshold –l 0 –S 0x10 –r 20 –R 100
```

SEE ALSO

```
get_health
get_sensor_event_enable
get_sensor_info
get_sensor_hysteresis
get_sensor_reading
get_sensor_threshold
get_diagnostics
```

5.12 System Administration

5.12.1 Get User Accessibility Information

NAME

`get_user_access` – display privilege level and channel accessibility for a given user

SYNOPSIS

```
get_user_access         {-c|--channel-number CHANNEL_NUMBER}
                        {-i|--user-id USER_ID}
```

DESCRIPTION

This command is used to retrieve the privilege level and channel accessibility associated with a given user ID

Channel Number is a 0-based index used to identify a messaging channel. A user is configured on a per-channel basis.

User ID is a 1-based number used to identify a user record.

The command displays the following information:

- Maximum number of user IDs supported on this channel
- Count of currently enabled user IDs on this channel
- Count of user IDs with fixed names on this channel
- Access, whether restricted to callback connection or unrestricted for callback and call-in connections
- Link authentication, whether it is enabled or disabled
- IPMI messaging, whether it is enabled or disabled
- User privilege limit

EXAMPLE

Display access information for user with ID 1 on channel 3:

```
# get_user_access –c 3 –i 1
```

SEE ALSO

`get_channel_access`

set_user_access

5.12.2 List Users

NAME

`list_users` – display the list of available users

SYNOPSIS

```
list_users
```

DESCRIPTION

This command is used to display the list of available users.

The command displays the following information:

- User ID
- User name
- Whether the user is enabled or not

SEE ALSO

```
get_user_access
get_user_info
list_users_access
```

5.12.3 List Users Access

NAME

`list_users_access` – display all users’ access properties for a given channel

SYNOPSIS

```
list_users_access       {-c|--channel-number CHANNEL_NUMBER}
```

DESCRIPTION

This command is used to display a list of users and their access properties for a given channel.

The **Channel Number** specifies the channel from which to retrieve information.

The command displays the information for all users that `get_user_access` displays for a given user:

- User ID
- User name
- User privilege limit
- Whether user is enabled or not
- IPMI messaging, whether it is enabled or disabled
- Link authentication, whether it is enabled or disabled
- Access, whether restricted to callback connection or unrestricted for callback and call-in connections

EXAMPLE

List users’ access on channel 3:

```
# list_users_access –c 3
```

SEE ALSO

```
get_user_access
get_user_info
list_users
```

5.12.4 Set User Accessibility Information

NAME

`set_user_access` – configure privilege level and channel accessibility associated with a given user ID

SYNOPSIS

```
set_user_access         {-c|--channel-number CHANNEL_NUMBER}
                        {-i|--user-id USER_ID}
                        [-l|--privilege-limit PRIVILEGE_LIMIT]
                        [-s|--session-limit SESSION_LIMIT]
                        [-r|--restricted 0|1]
                        [-a|--authentication 0|1]
                        [-m|--messaging 0|1]
```

DESCRIPTION

This command is used to configure the privilege level and channel accessibility associated with a given user ID. The limits set for channel access take precedence over the ones set here. The changes will take effect the next time the user establishes a session. The changes are saved in non-volatile storage.

Channel Number is a 0-based index used to identify a messaging channel. A user is configured on a per-channel basis. To configure information for the channel used for the active session, the value 0Eh may be used.

User ID is a 1-based number used to identify a user record.

A maximum **Privilege Limit** can be set for a user. This is the maximum privilege level the user can be switched to:

- 01h = Callback
- 02h = User
- 03h = Operator
- 04h = Administrator
- 05h = OEM
- 0Fh = No access

The user **Session Limit** sets the number of simultaneous sessions that can be activated with the username associated with the user ID.

The following can be enabled (**1**) or disabled (**0**):

- **Restricted.** User access restricted to callback

- **Authentication.** User link authentication
- **Messaging.** IPMI messaging

EXAMPLE

Change privilege level to operator (03h) for user with ID 2 on channel 4:

```
# set_user_access –c 4 –i 2 –l 3
```

Disable callback restriction, enable IPMI message for user with ID 2 on channel 3:

```
# set_user_access –c 3 –i 2 –r 0 –m 1
```

SEE ALSO

```
get_channel_access
get_user_access
```

5.12.5 Set User Information

NAME

`set_user_info` – add user, change username, set/change password, or enable/disable user

SYNOPSIS

```
set_user_info           {-i|--user-id USER_ID}
                        {-n|--username USER_NAME}

set_user_info           {-i|--user-id USER_ID}
                        {-p|--password PASSWORD}

set_user_info           {-i|--user-id USER_ID}
                        {-t|--test-password PASSWORD}

set_user_info           {-i|--user-id USER_ID}
                        {-e|--enable}

set_user_info           {-i|--user-id USER_ID}
                        {-d|--disable}
```

DESCRIPTION

This command is used to add a new user, change a user name, set and change user passwords, and enable and disable users. The changes are saved in non-volatile storage.

When adding a new user, by default, the user is disabled with the following access levels:

- Access is available only during callback connection
- Link authentication is disabled
- IPMI messaging is disabled
- Current and maximum operating level of NO ACCESS. This value does not add to, or subtract from, the number of enabled users

User ID is a 1-based number used to identify a user record. User ID 1 is reserved for the null user name.

User Name is an ASCII string with maximum length of 16.

Password is a string of no more than 20 characters. This option is required when assigning a password.

Test Password verifies the password value against the password saved in storage.

The user must be **Enabled** before the username assigned to the user can be used. Similarly, the user can be **Disabled**.

EXAMPLE

Assign a user name to ID 2:

```
# set_user_info -i 2 –n david
```

Set password for a new user with ID 2:

```
# set_user_info -i 2 -p password
```

Verify the password was set for user with ID 2:

```
# set_user_info –i 2 -t newPassword
```

Enable the user with ID 2:

```
# set_user_info ––user-id 2 --enable
```

SEE ALSO

```
set_user_access
get_user_info
```

5.13 System Management

5.13.1 Chassis Control

NAME

`chassis_control` – change the power state of the Chassis or issue a diagnostic interrupt

SYNOPSIS

```
chassis_control         {[-d|--power-down]|
                         [-u|--power-up]|
                         [-r|--hard-reset]|
                         [-i|--diagnostic-interrupt]|
                         [-s|--soft-shutdown]}
```

DESCRIPTION

This command provides power control, reset, and diagnostic interrupt control of the Chassis.

Only one of the following can be specified:

- **Power Down** forces the system into the SOFT OFF state. This is for emergency management power down actions.
- **Power Up** can be used to power up the Chassis.
- **Hard Reset** can be used to pulse the system reset signal regardless of the power state.
- **Diagnostic Interrupt** will pulse a version of a diagnostic interrupt that goes directly to the processor. This is typically used to cause the operating system to do a diagnostic dump. This may not be available in some implementations.
- **Soft Shutdown** behaves similarly to power down.

EXAMPLE

Send a power down request to the MicroTCA Carrier:

```
# chassis_control --power-down
```

SEE ALSO

```
fru_control
power_feed_control
```

5.13.2 Check IPMB-0 Bus Status

NAME

`check_ipmb0_status` – report current status of all IPMB-0 links on the MicroTCA Carrier

SYNOPSIS

```
check_ipmb0_status
```

DESCRIPTION

Physical IPMB-0 sensors are implemented by each EMMC and are used to monitor the state of the IPMBs.

This command lists the IPMB-A and IPMB-B enable status for each IPMB-0 link configured on the Carrier.

For each FRU address on the link, this command will display the connection status between the FRU and the Carrier Manager.

SEE ALSO

```
get_ipmb0_status
```

5.13.3 Failover

NAME

`failover` – transfer control of MicroTCA Carrier from the active MCH to the backup MCH

SYNOPSIS

```
failover
```

DESCRIPTION

This command is used to simulate a failure on the primary (active) MCH, thus transferring control of the Carrier to the backup MCH. If the redundant Carrier is unhealthy or is absent, no actions are taken.

The following are preserved when performing a failover:

- FRU state information
- Power settings of all FRUs
- E-Keying
- Alarms
- Fan level and cooling settings

5.13.4 Get Address Table

NAME

`get_address_table` – display the MicroTCA Carrier address table

SYNOPSIS

```
get_address_table       [-t|--site-type SITE_TYPE]
```

DESCRIPTION

This command is used to obtain the mapping between a hardware address and physical address within a Carrier. Each IPM Controller in a Carrier is assigned a unique hardware address on the backplane. The hardware address can be derived from the upper 7 bits of the IPMB address. The physical address describes the physical location of the FRU in the Carrier. The physical address identifies both the type of the FRU (site type) and the instance of the FRU of that type (site number). The site number within the site type will always start from 1.

The user may specify the target site type. If a target is not specified, the command displays all site types.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The command displays the following information:

- Hardware address
- Device Description
- Site type/number

EXAMPLE

Display FRUs supported by the Carrier with site type AMC (07h):

```
# get_address_table –t 0x7
```

Display all FRUs supported by the Carrier:

```
# get_address_table
```

SEE ALSO

```
get_address_info
```

5.13.5 Get Diagnostics

NAME

`get_diagnostics` – run diagnostics and display the results

SYNOPSIS

```
get_diagnostics         [-a|--ipmb-address IPMB_ADDRESS]
                        [-f|--fru-id FRU_ID]
                        [-r|--report]

get_diagnostics         [-t|--site-type SITE_TYPE]
                        [-n|--site-number SITE_NUMBER]
                        [-r|--report]
```

DESCRIPTION

This command is used to run the MicroTCA Carrier diagnostics and obtain the current status of the major, minor, and critical alarms in the Carrier.

The user may specify a target FRU by using the IPM Controller's **IPMB Address** and the **FRU Identifier.** If a target is not specified, the command displays information of the FRU hosting the MicroTCA Carrier Manager.

The target FRU can also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

-
- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

The **Report** option displays the results in a list format, unlike the non-report table version displayed otherwise.

The command displays the following information:

- FRU device name
- Location of the FRU
- Current FRU alarm state - major, minor, or critical
- Event code of the event that caused the alarm
- Sensor type and number of the sensor that generated this event

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 47: Diagnostics applicable target FRUs

EXAMPLE

View diagnostics for the AMC (site type 07h) located at site number 2:

```
# get_diagnostics –t 0x7 –n 2
```

View diagnostics for the AMC with FRU ID 5, with results in list format:

```
# get_diagnostics –f 5 -r
```

View diagnostics for all modules:

```
# get_diagnostics
```

SEE ALSO

```
list_event_code_descriptions
get_health
```

5.13.6 Get FRU Activation Sequence

NAME

`get_fru_activation_sequence` – display FRU activation sequence

SYNOPSIS

```
get_fru_activation_sequence
```

DESCRIPTION

This command is used to display the FRU activation sequence. This information is retrieved from the device's multi-records.

The MicroTCA Carrier FRU Information may include several Carrier Activation and Power Management records. Each record describes the amount of time and power allowable for any number of FRUs. The order of the records determines the order in which the FRUs will be brought to M3 (see AMC.0 R1.0 specification for exceptions).

The command displays the following information for each Carrier Activation and Power Management record:

- Number of seconds from startup that FRUs have in order to transition to M3 (activation in progress)
- Number of entries per entity type
- Physical address
- Power channel number
- Maximum FRU current capability
- Delay before powering next FRU
- Whether it is the MicroTCA Carrier Manager, MicroTCA Shelf Manager, or System Manager that controls activation, activating the FRU when it reaches M2 (activation request)

SEE ALSO

`get_power_management_info`

5.13.7 Get Health

NAME

`get_health` – provide a summary of the FRU alarm status

SYNOPSIS

```
get_health
```

DESCRIPTION

This command is used to obtain a summary of the currently active major, minor, and critical alarms for all the FRUs installed in the MicroTCA Carrier. If no alarms are active for a FRU, then the FRU is deemed 'healthy'.

Further information on an alarm status can be obtained by using the get_diagnostics command.

SEE ALSO

`get_diagnostics`

5.13.8 Get IPMB-0 Status

NAME

`get_ipmb0_status` – get EMMC IPMB-0 sensor status

SYNOPSIS

```
get_ipmb0_status
```

DESCRIPTION

Physical IPMB-0 sensors are implemented by each EMMC and are used to monitor the state of the IPMBs. For all EMMCs, querying these sensors returns information about their local IPMBs.

The Carrier Manager connects to IPMB-0 in a bused topology to the EMMCs.

SEE ALSO

`check_ipmb0_status`

5.13.9 Get Location Information

NAME

`get_location_info` – display location information for a given Module

SYNOPSIS

```
get_location_info       [-a|--ipmb-address IPMB_ADDRESS]
                        {-f|--fru-id FRU_ID}

get_location_info       {-n|--site-number SITE_NUMBER}
                        {-t|--site-type SITE_TYPE}
```

DESCRIPTION

This command describes the physical location of a Module within a MicroTCA Carrier.

The target FRU is specified using the IPM Controller's **IPMB Address** and the **FRU Identifier.**

The target FRU may also be specified using the site type and number.

Site Types are defined in the MicroTCA.0 R1.0 specification as follows:

- 04h = Cooling Unit
- 07h = AdvancedTCA™ Module (Mezzanine)
- 0Ah = MicroTCA Carrier Hub
- 0Bh = Power Module
- C0h - CFh = OEM defined
- FFh = Unknown
- All other values reserved

The **Site Number** identifies a specific FRU of a given site type.

The command displays the following information for a given Module:

- IPMB address
- Entity ID/instance
- Physical address
- Tier, and whether it is a 0-based or 1-based number
- Slot, and whether it is a 0-based or 1-based number
- Cartesian coordinates, in mm, from the lower left corner of the Carrier
- Carrier orientation, whether horizontal or vertical

APPLICABLE TARGET FRUS

Entity	Access Address	FRU ID(s)	Site Type	Site Number(s)
Carrier Manager	20h	0	N/A	N/A
MCH	20h	3 – 4	0Ah	1 – 2
AMC	20h	5 – 16	07h	1 – 12
CU	20h	40 – 41	04h	1 – 2
PM	20h	50 – 53	0Bh	1 – 4

Table 48: Location Information applicable target FRUs

EXAMPLE

View the location information for the MCH with FRU ID 3:

```
# get_location_info –f 3
```

View the location information for the Fan Tray (site type 04h) located at site number 1:

```
# get_location_info –t 0x4 –n 1
```

SEE ALSO

```
get_address_info
```

5.13.10 List FRUs Present

NAME

`list_frus_present` – provide a summary of the FRUs installed in the MicroTCA Carrier

SYNOPSIS

```
list_frus_present       [-e|--entity-id ENTITY_ID]
```

DESCRIPTION

This command is used to obtain a list of FRUs currently installed.

The user may specify a target FRU by using the IPM Controller's **Entity ID.** If a target is not specified, the command displays all FRUs currently installed in the MicroTCA Carrier.

The command displays the following information for each installed FRU:

- Location
- Entity ID/instance
- Device name
- Current hot-swap state
  - M1 – Inactive
  - M2 – Activation request
  - M3 – Activation in progress
  - M4 – Active
  - M5 – Deactivation request
  - M6 – Deactivation in progress
  - M7 – Communication lost

EXAMPLE

List FRUs installed with entity type A0h:

```
# list_frus_present –e 0xA0
```

List all FRUS installed

```
# list_frus_present
```

SEE ALSO

```
list_modules_present
```

5.13.11 List Modules Present

NAME

`list_modules_present` – provide a summary of the Modules installed in the current MicroTCA Carrier

SYNOPSIS

```
list_modules_present
```

DESCRIPTION

This command is used to obtain a list of Modules currently installed.

The command displays the following information for each installed FRU:

- Location
- Entity ID/instance
- Device name
- Current hot-swap state
  - M1 – Inactive
  - M2 – Activation request
  - M3 – Activation in progress
  - M4 – Active
  - M5 – Deactivation request
  - M6 – Deactivation in progress
  - M7 – Communication lost

SEE ALSO

```
list_carriers_present
list_fan_trays
list_frus_present
```

5.13.12 Set Carrier Number

NAME

`set_carrier_number` – update the Carrier Number of the Carrier

SYNOPSIS

```
set_carrier_number      {-i|--carrier-number CARRIER_NUMBER}
```

DESCRIPTION

This command updates the **Carrier Number** of the Carrier. The change is applied after restarting the Carrier Manager.

EXAMPLE

Update the current Carrier's Carrier Number to 3:

```
# set_carrier_number –i 3
```

SEE ALSO

`get_address_info`

Index