kontron

»Kontron Application Note«



AM42xx: IPMI Sensor User Guide

Document Revision 1.0
November 2009

Customer Service

Contact Information:	Kontron Canada, Inc.	Kontron Modular Computer GMBH
	4555 Rue Ambroise-Lafortune	Sudetenstrasse 7
	Boisbriand, Québec, Canada	87600 Kaufbeuren
	J7H 0A4	Germany
	Tel:(450) 979-5320	+49 (0) 8341 803 333
	(800) 354-4223	
	Fax: (450) 437-8053	+49 (0) 8341 803 339
	E-mail: support@ca.kontron.com	support-kom@kontron.com

Visit our web site at: www.kontron.com

© 2009 Kontron, an International Corporation. All rights reserved.

The information in this user's guide is provided for reference only. Kontron does not assume any liability arising out of the application or use of the information or products described herein. This user's guide may contain or reference information and products protected by copyrights or patents and do not convey any license under the patent rights of Kontron, nor the rights of others.

Kontron is a registered trademark of Kontron. All trademarks, registered trademarks, and trade names used in this user's guide are the property of their respective owners. All rights reserved. Printed in Canada. This user's guide contains information proprietary to Kontron. Customers may reprint and use this user's guide in other publications. Customers may alter this user's guide and publish it only after they remove the Kontron name, cover, and logo.

Kontron reserves the right to make changes without notice in product or component design as warranted by evolution in user needs or progress in engineering or manufacturing technology. Changes that affect the operation of the unit will be documented in the next revision of this user's guide.

Table of Contents

Scope 4
1. Sensor Introduction 4
Sensor Model 4
Sensor Classes 5
Event/Reading Type 6
Sensor Type 6
Sensor Reading 7
Event Data 8
Entity 8
Sensor ID 9
2. ipmitool 10
"*Get Sensor Reading*" Command 10
Sensor Command 10
Sdr Command 11
Sel Command 12
3. Pigeon Point's *clia* 13
Sensordata Command 13
Sel Command 14
4. Example 14
Analyzing the SEL 14
Read a "discrete" sensor 15
Annex A – List of AT8020/RTM8020 sensors 16
Annex B – Sensor-Specific Event 20
Annex C – Cause of State Change Values 21

Scope

This document's main purpose is to show how to analyze and understand events generated by sensors of the AM42xx which are stored in the System Event Log (SEL).

Typical situations in which the SEL needs to be consulted are as followed:

- Unexpected shutdown or reboot
- Front plate LEDs showing abnormality
- Any unusual behavior

In many cases, analyzing the SEL will allow to determine the root cause of the events and provide essential guidance in determining either preventive or corrective action.

This document also contains all the information needed to understand sensor readings. Readings provide useful information on the board's status. (e.g.: current POST code)

In order to be able to accomplish these tasks, the user will first be introduced to "Sensors" as defined in the *IPMI specification* v2.0. Once the first level knowledge has been acquired, detailed information will be provided on how to analyze and interpret the data collected from these sensors with tools such as *ipmitool* and Pigeon Point's *clia*. Last but not least, *"Annex A"* presents a detailed list of all the sensors implemented on the AM42xx.

1. Sensor Introduction

Sensor Model

"Access to monitored information, such as temperatures and voltages, fan status, etc., is provided via the IPMI Sensor Model. Instead of providing direct access to the monitoring hardware IPMI provides access by abstracted sensor commands, such as the Get Sensor Reading command, implemented via a management controller. This approach isolates software from changes in the platform management hardware implementation.

Sensors are classified according to the type of readings they provide and/or the type of events they generate. A sensor can return either an analog or discrete reading. Sensor events can be discrete or threshold-based.

The different event types, sensor types, and monitored entities are represented using numeric codes defined in the IPMI specification. IPMI avoids reliance on strings for management information. Using numeric codes facilitates internationalization, automated handling by higher level software, and reduces management controller code and data space requirements."[1]

For the purpose of this document, the two most important characteristics of a sensor are:

- *Event/Reading Type*
- *Sensor Type*

[1] IPMI v2.0 Section 1.7.5 p:13

Sensor Classes

Sensors fall into the following classes:

Discrete:

- These are *State Sensors*. The reading they return contains two bytes where each bit can represent a unique state.
- Up to 15 possible states (not 16 since bit15 from the returned reading is reserved)
- More than one state may be active simultaneously.
- Events are generated by a unique state. Thus, *Event Messages* do not return a bit field, just a single offset value corresponding to a single event.

'Digital' Discrete:

- A digital sensor is not really a unique class, but a term commonly used to refer to special case of a discrete sensor that only has two possible states.

Threshold:

- Threshold based.
- Changes event status on reading comparison to threshold values.
- Threshold enumerations may be considered a special case of the discrete sensor type.

***OEM*:**

- Special case of discrete where the meanings of the state's (offsets) are *OEM* defined.

Event/Reading Type

"Event/Reading Type codes are used in SDRs (sensor data records) and Event Messages to indicate the trigger type for an event. These codes are also used in SDRs to indicate what types of present reading a sensor provides.

Event/Reading Type Codes are used to specify a particular enumeration (offset) that identifies a set of possible events that can be generated by a sensor. For "Discrete" sensors, the specification of an Event/Reading Type code enumeration also indicates the type of reading the sensor provides."[2]

Event/Reading Type are listed in the following Table.

Table 1: *"Event/Reading Type"* Code Ranges[3]

Event/Reading Type Code category	7-bit Event/Reading Type Code Range	Sensor Class	Description
unspecified	00h	n/a	Event/Reading Type unspecified.
Threshold	01h	threshold	Threshold-based. Indicates a sensor that utilizes values that represent discrete threshold states in sensor access and/or events. The Event/Reading event offsets for the different threshold states are given in *Table 42-2, Generic Event/Reading Type Codes*, below.
Generic	02h-0Ch	discrete	Generic Discrete. Indicates a sensor that utilizes an Event/Reading Type code & State bit positions / event offsets from one of the sets specified for Discrete or 'digital' Discrete Event/Reading class in *Table 42-2, Generic Event/Reading Type Codes*, below.
Sensor-specific	6Fh	discrete	Sensor-specific Discrete. Indicates that the discrete state information is specific to the sensor type. State bit positions / event offsets for a particular sensor type are specified in the 'sensor-specific offset' column in *Table 42-3, Sensor Type Codes*, below.
OEM	70h-7Fh	OEM	OEM Discrete. Indicates that the discrete state information is specific to the OEM identified by the Manufacturer ID for the IPM device that is providing access to the sensor.

Sensor Type

"Discrete" sensors defined with an *Event/Reading Type* 6Fh (Sensor-specific) will use *"Sensor-Specific"* definition for their offset and *"Event Data"*. *"Sensor-specific"* definition is available for many *"Sensor Type"* and may be "OEM" defined for *OEM sensor types*.

[2] IPMI v2.0 Section 42.1, p:498
[3] IPMI v2.0 Table 42-1, Event/Reading Type Code Ranges, p:499

Sensor Reading

Reading from a sensor is available through the *"Get Sensor Reading"* command. All other more complex commands which provide sensor reading use this raw command. Therefore, it is important to understand the format in which data is returned.

Table 2: *"Get Sensor Reading"* Command[4]

Request Data	1	sensor number (FFh = reserved)
Response Data	1	Completion Code.
	2	Sensor reading Byte 1: byte of reading. Ignore on read if sensor does not return an numeric (analog) reading.
	3	[7] - 0b = All Event Messages disabled from this sensor [6] - 0b = sensor scanning disabled [5] - 1b = reading/state unavailable (formerly "initial update in progress"). This bit is set to indicate that a 're-arm' or 'Set Event Receiver' command has been used to request an update of the sensor status, and that update has not occurred yet. Software should use this bit to avoid getting an incorrect status while the first sensor update is in progress. This bit is only required if it is possible for the controller to receive and process a 'Get Sensor Reading' or 'Get Sensor Event Status' command for the sensor before the update has completed. This is most likely to be the case for sensors, such as fan RPM sensors, that may require seconds to accumulate the first reading after a re-arm. The bit is also used to indicate when a reading/state is unavailable because the management controller cannot obtain a valid reading or state for the monitored entity, typically because the entity is not present. See *Section 16.4, Event Status, Event Conditions, and Present State* and *Section 16.6, Re-arming* for more information. [4:0] - reserved. Ignore on read.
	(4)	For threshold-based sensors Present threshold comparison status [7:6] - reserved. Returned as 1b. Ignore on read. [5] - 1b = at or above (≥) upper non-recoverable threshold [4] - 1b = at or above (≥) upper critical threshold [3] - 1b = at or above (≥) upper non-critical threshold [2] - 1b = at or below (≤) lower non-recoverable threshold [1] - 1b = at or below (≤) lower critical threshold [0] - 1b = at or below (≤) lower non-critical threshold For discrete reading sensors [7] - 1b = state 7 asserted [6] - 1b = state 6 asserted [5] - 1b = state 5 asserted [4] - 1b = state 4 asserted [3] - 1b = state 3 asserted [2] - 1b = state 2 asserted [1] - 1b = state 1 asserted [0] - 1b = state 0 asserted
	(5)	For discrete reading sensors only. (Optional) (00h Otherwise) [7] - reserved. Returned as 1b. Ignore on read. [6] - 1b = state 14 asserted [5] - 1b = state 13 asserted [4] - 1b = state 12 asserted [3] - 1b = state 11 asserted [2] - 1b = state 10 asserted [1] - 1b = state 9 asserted [0] - 1b = state 8 asserted

Completion Code:
Will not be displayed if the Request Message completes successfully and normally.

Byte 1: Sensor Reading

- For *"Discrete"* Sensors, will return 00h
- For *"Threshold"* based sensors, will return the analog reading. This value is coded according to the *Event/Reading Type* and/or *Sensor type*. Tools such as *ipmitool* provide commands which will decode this information in a human readable format.

[4] IPMI v2.0 Table 35-15, Get Sensor Reading Command, p:464

Byte 2:

- Provides information on the sensor

Byte 3:

- For *"Threshold"* based sensor: Indicates were the reading stands against the threshold values.
- For *"Discrete"* sensors: Indicates which sensor offsets (states) are asserted for offset 00h to 07h.

Byte 4:

- For *"Threshold"* based sensor: 80h (since bit 7 is always 1b)
- For *"Discrete"* sensors: Indicates which sensor offsets (states) are asserted for offset 08h to 14h.

NOTE: Sensors have a reading mask which is *"OEM"* defined. This is used to ignore unused states during reading. Therefore, if a state that should be asserted is not read, the *"Reading Mask"* should be verified.

Event Data

When a sensor changes state, an *"Event Message"* is sent to the SEL only if the *"Event Mask"* indicates that the new state must generate an event.

The *"Event Data"* contains 3 bytes where only the first byte is used. The signification of these bytes is listed in *"Annex A"* for every sensors implemented on the AM42xx.

Entity

"An Entity ID is a standardized numeric code that is used in SDRs to identify the types of physical entities or FRUs in the system"[5]

The AM42xx's entity ID can differ since it can be used as entity of a carrier board or stand alone in a microTCA chassis. This is why the entity ID is not shown in "*Annex A*".

[5] IPMI v2.0 Section 39, p:488

Sensor ID

Sensors have a numerical ID used to identify them. The sensor ID as seen in the list from *"Annex A"* might not be the same in particular cases.

In the case where the AM42xx is in a carrier board, the sensor ID's are determined during the board's activation according to the order in which the entities are activated. First sensors to be designated an ID are the ones populated on *FRU0* (Entity: PICMG Front Board). Afterwards, it depends on which entity is the first to ask for activation. Therefore, the AM42xx's sensors might have an offset compared to the IDs from the *"Annex A"* list.

Sensors from the list should not be identified by a specific numerical ID but rather by their sensor name *(IE: "FRU Agent")*.

2. ipmitool

This section does not list all commands that can be used to get information on sensors. However, these commands provide most of the relevant information.

ipmitool can be obtained at: http://ipmitool.sourceforge.net/

"*Get Sensor Reading*" Command

This PICMG command, introduced in previous section, can be used by raw command:

ipmitool raw 0x04 0x2d <id>

0x04 : Network function Code for Sensor Event
0x2d : Get Sensor Reading command
<id> : Sensor ID

Sensor Command

This command provides various information on the board's sensors. It is also the only command (excluding raw commands) that lists the reading "Data Byte" 3 and 4 (see "Sensor Reading" from section "1. Sensor Introduction"). Other *ipmitool* command provides sensor reading "Data Byte" 3.

ipmitool sensor

Figure 1: "ipmitool sensor" command

Sensor	Analog Reading	Type	Reading Byte 3&4	Threshold Values					
IPMC Storage Err	0x0	discrete	0x0080	na	na	na	na	na	na
IPMC SEL State	0x0	discrete	0x0080	na	na	na	na	na	na
ME Availability	0x0	discrete	0x4280	na	na	na	na	na	na
Jumper Status	0x0	discrete	0x0080	na	na	na	na	na	na
IPMI Info-1	0x0	discrete	0x0080	na	na	na	na	na	na
IPMI Info-2	0x0	discrete	0x0080	na	na	na	na	na	na
RTM:IPMI Info-2	0x0	discrete	0x0080	na	na	na	na	na	na
RTM:IPMI Info-1	0x0	discrete	0x0080	na	na	na	na	na	na
RTM:MMC Stor Err	0x0	discrete	0x0080	na	na	na	na	na	na
RTM:MMC FwUp	0x0	discrete	0x0080	na	na	na	na	na	na
RTM:MMC Reboot	0x0	discrete	0x0180	na	na	na	na	na	na
RTM:FRU Agent	0x0	discrete	0x0180	na	na	na	na	na	na
RTM:IPMBL State	0x88	discrete	0x0880	na	na	na	na	na	na
RTM:Ver Change	0x0	discrete	0x0080	na	na	na	na	na	na
RTM:Health Error	0x0	discrete	0x0180	na	na	na	na	na	na
RTM:Pwr Good Ev	0x0	discrete	0x4780	na	na	na	na	na	na
RTM:Power Good	0x0	discrete	0x4780	na	na	na	na	na	na
RTM:Power State	0x0	discrete	0x0180	na	na	na	na	na	na
RTM:Disk Bay	0x0	discrete	0x0480	na	na	na	na	na	na
RTM:USB1 OC	0x0	discrete	0x0180	na	na	na	na	na	na
RTM:USB0 OC	0x0	discrete	0x0180	na	na	na	na	na	na
RTM:SFP-B OC	0x0	discrete	0x0180	na	na	na	na	na	na
RTM:SFP-B Pres	0x0	discrete	0x0080	na	na	na	na	na	na
RTM:SFP-A OC	0x0	discrete	0x0180	na	na	na	na	na	na
RTM:SFP-A Pres	0x0	discrete	0x0080	na	na	na	na	na	na
RTM:Vcc +12VDisk	12.100	Volts	ok	10.560	11.165	na	na	12.870	13.475
RTM:Vcc +12V	10.899	Volts	ok	0.000	10.395	na	na	14.301	16.002
RTM:Vcc BlueLED	6.612	Volts	ok	3.876	5.814	na	na	7.410	9.424
RTM:Vcc +1.2V	1.197	Volts	ok	0.000	1.121	na	na	1.285	1.588
RTM:Vcc +5V	5.079	Volts	ok	0.000	4.666	na	na	5.346	6.124
RTM:Vcc +3.3V	3.296	Volts	ok	0.000	3.072	na	na	3.536	4.032
RTM:Vcc +3.3VSUS	3.298	Volts	ok	0.000	2.941	na	na	3.672	4.335
RTM:Temp MMC	27.000	degrees C	ok	-20.000	-10.000	-5.000	60.000	70.000	80.000
RTM:Temp SASCtrl	56.000	degrees C	ok	-20.000	-10.000	-5.000	105.000	115.000	125.000
RTM:Temp Air Out	37.000	degrees C	ok	-10.000	0.000	5.000	60.000	70.000	80.000
RTM:Temp Air In	27.000	degrees C	ok	-10.000	0.000	5.000	60.000	70.000	80.000

*This example was made using an AT8050

Sdr Command

The following command will provide additional information on sensors.

ipmitool sdr list –v

Figure 2: "ipmitool sdr list –v" Command

```
Sensor ID              : RTM:SFP-A OC (0x82)
 Entity ID             : 192.96 (PICMG Rear Transition Module)
 Sensor Type (Discrete): Current
 States Asserted       : Digital State
                         [State Deasserted]
 Assertions Enabled    : Digital State
                         [State Asserted]
 Deassertions Enabled  : Digital State
                         [State Asserted]

Sensor ID              : RTM:SFP-A Pres (0x81)
 Entity ID             : 192.96 (PICMG Rear Transition Module)
 Sensor Type (Discrete): Slot / Connector
 Assertions Enabled    : Slot/Connector
                         [Fault Status]
                         [Device Installed]
 Deassertions Enabled  : Slot/Connector
                         [Fault Status]
                         [Device Installed]

Sensor ID              : RTM:Vcc +12VDisk (0x80)
 Entity ID             : 192.96 (PICMG Rear Transition Module)
 Sensor Type (Analog)  : Voltage
 Sensor Reading        : 12.100 (+/- 0.220) Volts
 Status                : ok
 Nominal Reading       : 11.990
 Normal Minimum        : 11.165
 Normal Maximum        : 12.870
 Positive Hysteresis   : 0.495
 Negative Hysteresis   : 0.495
 Minimum sensor range  : Unspecified
 Maximum sensor range  : Unspecified
 Event Message Control : Per-threshold
 Readable Thresholds   : lnr lcr ucr unr
 Settable Thresholds   : lnr lcr ucr unr
 Threshold Read Mask   : lnr lcr ucr unr
 Assertions Enabled    : lcr- lnr- ucr+ unr+
 Deassertions Enabled  : lcr- lnr- ucr+ unr+
```

*This example was made using an AT8050

Sel Command

The "ipmitool sel" command shows the sensor's NAME and ID and reports in a human readable format the "Event Data". In some cases, *ipmitool* is not able to analyze the "Event Data" and will print "Event Data" 1,2 and 3. When this happens, the *"Annex A"*'s list should be used to decode these bytes.

ipmitool sel list

Figure 3: "ipmitool sel list" Command

```
134 | 11/03/2009 | 14:55:15 | Voltage #0x6b | Lower Critical going low
135 | 11/03/2009 | 14:55:19 | FRU Hot Swap #0x02 | Transition to M4 | Asserted
136 | 11/03/2009 | 14:55:20 | FRU Hot Swap #0x03 | Transition to M4 | Asserted
137 | 11/03/2009 | 14:55:21 | FRU Hot Swap #0x01 | Transition to M4 | Asserted
138 | 11/03/2009 | 14:55:29 | Version Change #0x51 | Firmware or software change detected | Asserted
139 | 11/03/2009 | 14:55:33 | Reset #0x43 | State Asserted
13a | 11/03/2009 | 15:08:27 | System ACPI Power State #0x4e | S0/G0: working | Asserted
13b | 11/03/2009 | 15:56:59 | Reset #0x43 | State Asserted
13c | 11/03/2009 | 15:56:59 | Reset #0x43 | State Asserted
13d | 11/03/2009 | 15:57:33 | Reset #0x43 | State Asserted
13e | 11/03/2009 | 15:59:00 | Reset #0x43 | State Asserted
13f | 11/03/2009 | 16:18:22 | Reset #0x43 | State Asserted
140 | 11/03/2009 | 16:19:16 | System ACPI Power State #0x4e | S0/G0: working | Asserted
141 | 11/03/2009 | 16:22:57 | Reset #0x43 | State Asserted
142 | 11/03/2009 | 16:22:57 | Reset #0x43 | State Asserted
143 | 11/03/2009 | 16:23:58 | System ACPI Power State #0x4e | S0/G0: working | Asserted
144 | 11/03/2009 | 16:26:13 | Reset #0x43 | State Asserted
145 | 11/03/2009 | 16:26:14 | Reset #0x43 | State Asserted
146 | 11/03/2009 | 16:26:33 | Reset #0x43 | State Asserted
147 | 11/03/2009 | 16:27:45 | Reset #0x43 | State Asserted
148 | 11/03/2009 | 16:29:26 | System ACPI Power State #0x4e | S0/G0: working | Asserted
149 | 11/03/2009 | 20:56:15 | IPMB-0 Status #0x53 | IPMB-A enabled, IPMB-B disabled | Asserted
14a | 11/03/2009 | 20:56:26 | IPMB-0 Status #0x53 | IPMB-A enabled, IPMP-B enabled | Asserted
```

*This example was made using an AT8050

It is recommended to use Pigeon Point's "*clia* sel" command to analyze SEL data since it provides more details on sensors.

3. Pigeon Point's *clia*

In this section, two useful commands from the Shelf Manager's "Command Line Interpreter" will be detailed.

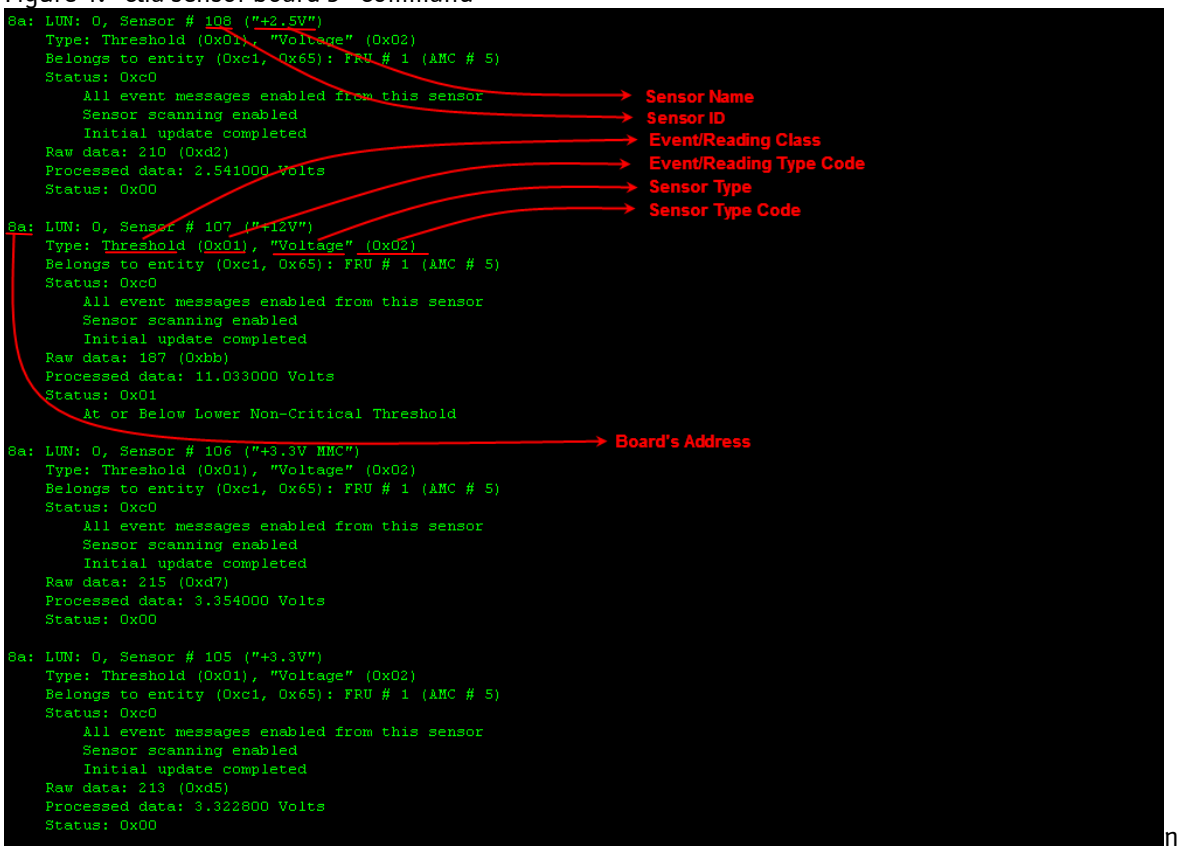
For more info consult Pigeon Point's web site: http://www.pigeonpoint.com/

Sensordata Command

This command can be used to get more details on sensors. It also indicates whether "Event Messages" are enabled or not.

clia sensor board 5 (in this example, the board is located in slot 5)

Figure 4: "clia sensor board 5" Command



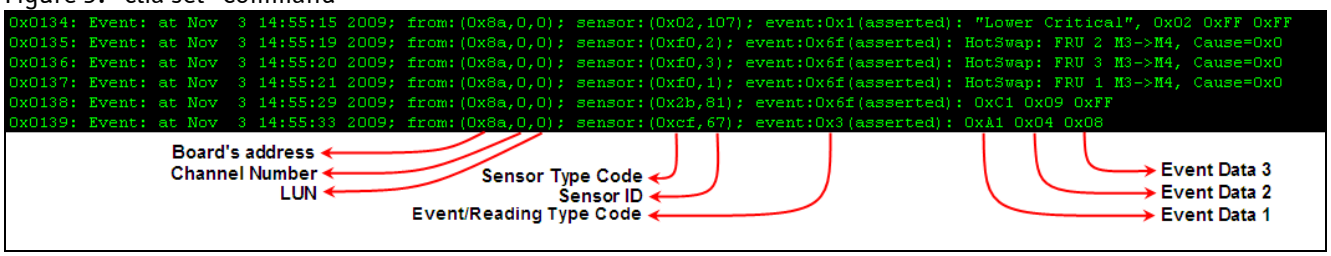
*This example was made using an AT8050

Sel Command

The "*clia* sel" command shows all the information you will need to find a definition for the event "Data Bytes" in *"Annex A"*'s list. Some Events are directly analyzed.

clia sel board 5 (in this example, the board is located in slot 5)

Figure 5: "clia sel" Command



*This example was made using an AT8050

4. Example

Examples from this section where made using an AT8050.

Analyzing the SEL

Whenever an unusual situation is reported, the SEL should be consulted:

Figure 6: SEL Event Example

```
0x0138: Event: at Nov  3 14:55:29 2009; from:(0x8a,0,0); sensor:(0x2b,81); event:0x6f(asserted): 0xC1 0x09 0xFF
```

Use the Sensor ID to determine the sensor's name with the help of "*clia* sensor" or "*ipmitool* sensor" command. With this name, run a search in the "*Annex A*" list and verify that the other information match since it may occur that two sensors have the same name if they belong to different "Entity".

Sensor ID	Sensor Name / Entity (ID)	Event/Reading Type (Class and Code) / Sensor Type (Code)	Offset	Data Byte 1	Data Byte 2	Data Byte 3
81	FW Ver Change / PICMG Front Board (160.96)	Sensor-specific (Discrete 0x6f) / Version Change (0x2b)	00h (bit 0): Hardware change detected with associated Entity. Informational. This offset does not imply whether the hardware change was successful or not. Only that a change occurred. 01h (bit 1): Firmware or software change detected with associated Entity. Informational. Success or failure not implied. 02h (bit 2): Hardware incompatibility detected with associated Entity. 03h (bit 3): Firmware or software incompatibility detected with associated Entity. 04h (bit 4): Entity is of an invalid or unsupported hardware version. 05h (bit 5): Entity contains an invalid or unsupported firmware or software version. 06h (bit 6): Hardware Change detected with associated Entity was successful. (deassertion event means 'unsuccessful'). 07h (bit 7): Software or F/W Change detected with associated Entity was successful. (deassertion event means 'unsuccessful')	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	bit[7:0]: Version change type 00h unspecified 01h management controller device ID (change in one or more fields from 'Get Device ID') 02h management controller firmware revision 03h management controller device revision 04h management controller manufacturer ID 05h management controller IPMI version 06h management controller auxiliary firmware ID 07h management controller firmware boot block 08h other management controller firmware 09h system firmware (EFI / BIOS) change 0Ah SMBIOS change 0Bh operating system change 0Ch operating system loader change 0Dh service or diagnostic partition change 0Eh management software agent change 0Fh management software application change 10h management software middleware change 11h programmable hardware change (e.g. FPGA) 12h board/FRU module change (change of a module plugged into associated entity) 13h board/FRU component change (addition or removal of a replaceable component on the board/FRU that is not tracked as a FRU)	FFh

Let's analyze the "Event Data":

Event Data 1: 0xC1

0xC1 = 1100 0001
Bit[7:6] = 11b : sensor-specific event extension code in byte 2
Bit[5:4] = 00b : unspecified byte 3
Bit[3:0] = 00001b : offset from Event/Reading code (offset which triggered the event)
→ 01h (bit 1): Firmware or software change detected with associated Entity. Informational. Success or failure not implied.

Event Data 2: 0x09

bit[7:0]: Version change type
→ system firmware (EFI / BIOS) change

Read a "discrete" sensor

In some cases, it can be useful to consult a discrete sensor's "State". For example, to know which jumpers are installed on a board without pulling it out of the chassis, the "Jumper Status" sensor should be consulted.

To do so, use the "*ipmitool sensor*" command or the raw "Get Sensor Reading" command. These commands will provide "Reading Byte" 3&4 (which correspond to "Response Data Byte" 4&5 as shown section "1. Sensor Introduction"). To analyze the reading, table 2 and the offset column on "*Annex A*" should be consulted.

Example: Reading Bytes 3&4 = 0xA193 for the "Jumper Status" sensor.

Meaning:

0xA193 = 1010 0001 1001 0011

- 00h (bit 0): Jumper 00 Present (JP1: 1-2)
- 05h (bit 5): Jumper 05 Present (JP1: 11-12)
- 07h (bit 7): Jumper 07 Present (JP2: 1-2)
- 08h (bit 8): Jumper 08 Present (JP2: 3-4)
- 09h (bit 9): Jumper 09 Present (JP2: 5-6)
- 0Ch (bit 12): Jumper 12 Present (JP2: 11-12)

Annex A – List of AM42xx sensors

Sensor ID	Sensor Name / Entity	Event/Reading Type (Class and Code) / Sensor Type (Code)	Description	Offset	Data Byte 1	Data Byte 2	Data Byte 3
0	FRU1 Hot Swap / PICMG AdvancedMC Module	(Discrete 0x6f) / PICMG FRU Hotswap (0xf0)	ATCA Board FRU Hot Swap Sensor for FRU 1 (AMC B1)	0h (bit 0): M0 – FRU Not Installed 1h (bit 1): M1 – FRU Inactive 2h (bit 2): M2 – FRU Activation Request 3h (bit 3): M3 – FRU Activation In Progress 4h (bit 4): M4 – FRU Active 5h (bit 5): M5 – FRU Deactivation Request 6h (bit 6): M6 – FRU Deactivation In Progress 7h (bit 7): M7 – FRU Communication Lost 8h-Fh : Reserved	[7:4] = Ah (OEM code in Event Data 2, OEM code in Event Data 3) [3:0] = Current State 0h = M0 – FRU Not Installed 1h = M1 – FRU Inactive 2h = M2 – FRU Activation Request 3h = M3 – FRU Activation In Progress 4h = M4 – FRU Active 5h = M5 – FRU Deactivation Request 6h = M6 – FRU Deactivation In Progress 7h = M7 – FRU Communication Lost 8h-Fh = Reserved	bit[7:4] = Cause of state change. See, "Cause of state change values," for values. (Annex C) bit[3:0] = Previous State 0h = M0 – FRU Not Installed 1h = M1 – FRU Inactive 2h = M2 – FRU Activation Request 3h = M3 – FRU Activation In Progress 4h = M4 – FRU Active 5h = M5 – FRU Deactivation Request 6h = M6 – FRU Deactivation In Progress 7h = M7 – FRU Comminication Lost 8h -Fh = Reserved	bit[7:0] = FRU Device ID
1	IPMI Info-1 / PICMG AdvancedMC Module	(Discrete 0x70) / OEM Firmware Info (0xc0)	Internal Management Controller firmware diagnostic	Internal use only, No additionnal info availble.	Should not generate Event. Please contact Kontron Canada Inc. Technical Support if an Event is triggered.	-	-
2	IPMI Info-2 / PICMG AdvancedMC Module	(Discrete 0x71) / OEM Firmware Info (0xc0)	Internal Management Controller firmware diagnostic	Internal use only, No additionnal info availble.	Should not generate Event. Please contact Kontron Canada Inc. Technical Support if an Event is triggered.	-	-
3	FRU Agent / PICMG AdvancedMC Module	(Discrete 0x0a) / OEM FRU Information Agent (0xc5)	FRU Information Agent - Data Error Detection	Only States (00h, 01h, 02h, 06h, 08h) trigger an event 00h (bit 0) = transition to Running 01h (bit 1) = transition to In Test 02h (bit 2) = transition to Power Off 03h (bit 3) = transition to On Line 04h (bit 4) = transition to Off Line 05h (bit 5) = transition to Off Duty 06h (bit 6) = transition to Degraded 07h (bit 7) = transition to Power Save 08h (bit 8) = Install Error	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	Event Data 2 is used as bit flag error Bit 7: unspecifiedError Bit 6: notPresentError Bit 5: multirecHeaderError Bit 4: multirecDataError Bit 3: timeout error Bit 2: ipmcError Bit 1: fruDataError Bit 0: commonHeaderError	Event Data 3 is used a bit flag error Bit 7: SetClockState Not Supported Bit 6: SetClockState Error Bit 5: SetPortState Not Supported Bit 4: SetPortState Error Bit 3: Clock Internal Mismatch Bit 2: Clock Match Error, Not a single clock matches Bit 1: Internal mismatch Bit 0: Match Error, Not in single link matches
4	ModuleHotSwap / PICMG AdvancedMC Module	(Discrete 0x6f) / OEM RTM Module Hotswap (0xf2)	AMC.0 mandatory Module Hot Swap state	0h (bit 0): Module Handle Closed 1h (bit 1): Module Handle Opened 2h (bit 2): Quiesced 3h (bit 3): Backend Power Failure 4h (bit 4): Backend Power Shut Down 5h-Fh = Reserved	bit[7:4] = 00h (unspecified Event Data 2 and 3) bit[3:0] = Current Event 0h = Module Handle Closed 1h = Module Handle Opened 2h = Quiesced 3h = Backend Power Failure 4h = Backend Power Shut Down 5h-Fh = Reserved	-	-
5	IPMBL State / PICMG AdvancedMC Module	(Discrete 0x6f) / OEM IPMBL Link State (0xc3)	IPMB-L fault detection sensor	00h (bit 0): IPMB-A disabled, IPMB-B disabled 01h (bit 1): IPMB-A enabled, IPMB-B disabled 02h (bit 2): IPMB-A disabled, IPMB-B enabled 03h (bit 3): IPMB-A enabled, IPMB-B enabled	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	Always 0	Bit[7:3]: Always 0 Bit[2:0]: 0h = No failure 1h = Unable to drive clock HI 2h = Unable to drive data HI 3h = Unable to drive clock LO 4h = Unable to drive data LO 5h = clock low timeout 6h = Under test (the IPM Controller is attempting to determine who is causing a bus hang) 7h = Undiagnosed Communication Failure
6	MMC Stor Err / PICMG AdvancedMC Module	(Discrete 0x6f) / Management Subsystem Health (0x28)	Management sub-system health: non volatile memory error	00h (bit 0): sensor access degraded or unavailable (A sensor that is degraded will still return valid results, but may be operating with a slower response time, or may not detect certain possible states. A sensor that is unavailable is not able to return any results (scanning is disabled.) 01h (bit 1): controller access degraded or unavailable (The ability to access the controller has been degraded, or access is unavailable, but the party that is doing the monitoring cannot determine which.) 02h (bit 2): management controller off-line (controller cannot be accessed for normal operation because it has been intentionally taken off-line for a non-error condition. Note that any commands that are available must function according to specification.) 03h (bit 3): management controller unavailable (controller cannot be accessed because of an error condition) 04h (bit 4): Sensor failure (the sensor is known to be in error. It may still be accessible by software) 05h (bit 5): FRU failure	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	See Sensor Specific Event (Annex B)	See Sensor Specific Event (Annex B)
7	MMC Reboot / PICMG AdvancedMC Module	(Discrete 0x03) / Platform Alert (0x24)	IPMC reboot detection	00h (bit 0): State Deasserted 01h (bit 1): State Asserted	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	-	-
8	MMC FwUp / PICMG AdvancedMC Module	(Discrete 0x6f) / OEM FWUM Status (0xc7)	Kontron OEM IPMC firmware upgrade Status	00h (bit 0): First Boot after upgrade 01h (bit 1): First Boot after roolback	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	-	-
9	Ver change / PICMG AdvancedMC Module	(Discrete 0x6f) / Version Change (0x2b)	Firmware Change Detection	00h (bit 0): Hardware change detected with associated Entity. Informational. This offset does not imply whether the hardware change was successful or not. Only that a change occurred. 01h (bit 1): Firmware or software change detected with associated Entity. Informational. Success or failure not implied. 02h (bit 2): Hardware incompatibility detected with associated Entity. 03h (bit 3): Firmware or software incompatibility detected with associated Entity. 04h (bit 4): Entity is of an invalid or unsupported hardware version. 05h (bit 5): Entity contains an invalid or unsupported firmware or software version. 06h (bit 6): Hardware Change detected with associated Entity was successful. (deassertion event means 'unsuccessful'). 07h (bit 7): Software or F/W Change detected with associated Entity was successful. (deassertion event means 'unsuccessful')	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	bit[7:0]: Version change type 00h unspecified 01h management controller device ID (change in one or more fields from 'Get Device ID') 02h management controller firmware revision 03h management controller device revision 04h management controller manufacturer ID 05h management controller IPMI version 06h management controller auxiliary firmware ID 07h management controller firmware boot block 08h other management controller firmware 09h system firmware (EFI / BIOS) change 0Ah SMBIOS change 0Bh operating system change 0Ch operating system loader change 0Dh service or diagnostic partition change 0Eh management software agent change 0Fh management software application change 10h management software middleware change 11h programmable hardware change (e.g. FPGA) 12h board/FRU module change (change of a module plugged into associated entity) 13h board/FRU component change (addition or removal of a replaceable component on the board/FRU that is not tracked as a FRU)	-

Sens or ID	Sensor Name / Entity	Event/Reading Type (Class and Code) / Sensor Type (Code)	Description	Offset	Data Byte 1	Data Byte 2	Data Byte 3
10	IPMI Watchdog / PICMG AdvancedMC Module	(Discrete 0x6f) / Watchdog (0x23)	IPMI Watchdog (payload watchdog)	00h (bit 0): Timer expired, status only (no action, no interrupt) 01h (bit 1): Hard Reset 02h (bit 2): Power Down 03h (bit 3): Power Cycle 04h-07h (bit[4:7]): reserved 08h (bit 8): Timer interrupt	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	The Event Data 2 field for this command can be used to provide an event extension code, with the following definition: bit[7:4]: interrupt type 0h = none 1h = SMI 2h = NMI 3h = Messaging Interrupt Fh = unspecified all other = reserved bit[3:0]: timer use at expiration: 0h = reserved 1h = BIOS FRB2 2h = BIOS/POST 3h = OS Load 4h = SMS/OS 5h = OEM Fh = unspecified all other = reserved	-
11	CPU Reset / PICMG AdvancedMC Module	(Discrete 0x03) / OEM Board Reset (0xcf)	CPU reset type and sources	00h (bit 0): State Deasserted 01h (bit 1): State Asserted	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	Reset Type 00h: Warm reset 01h: Cold reset 02h: Forced Cold [Warm reset reverted to Cold] 03h: Soft reset [Software jump] 04h: Hard Reset 05h: Forced Hard [Warm reset reverted to Hard]	Reset Source 00h: IPMI Watchdog [cold, warm or forced cold] (IPMI Watchdog2 sensors gives additionnal details) 01h: IPMI commands [cold, warm or forced cold] (chassis control, fru control) 02h: Processor internal checkstop 03h: Processor internal reset request 04h: Reset button [warm or forced cold] 05h: Power up [cold] 06h: Legacy Initial Watchdog / Warm Reset Loop Detection * [cold reset] 07h: Legacy Programmable Watchdog [cold, warm or forced cold] 08h: Software Initiated [soft, cold, warm of forced cold] 09h: Setup Reset [Software Initiated Cold] 0Ah: Power Cycle / Full Reset / Global Platform Reset FFh: Unknown
12	Temp Air Out / PICMG AdvancedMC Module	(Analog 0x01) / Temperature (0x01)	Ait Outlet Temperature (Degrees)	Threshold Values: 00h : Lower Non-critical - going low 01h : Lower Non-critical - going high 02h : Lower Critical - going low 03h : Lower Critical - going high 04h : Lower Non-recoverable - going low 05h : Lower Non-recoverable - going high 06h : Upper Non-critical - going low 07h : Upper Non-critical - going high 08h : Upper Critical - going low 09h : Upper Critical - going high 0Ah : Upper Non-recoverable - going low 0Bh : Upper Non-recoverable - going high	[7:6] - 00b = unspecified byte 2 01b = trigger reading in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = trigger threshold value in byte 3 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 Bit [3:0] = Offset from Event/Reading Code for threshold event.	Reading that triggered the event, FFh or not present if unspecified. Do not confuse reading with Threshold Value	threshold value that triggered event, FFh or not present if unspecified. If present, Event Data 2 must be present
13	Temp CPU / PICMG AdvancedMC Module	(Analog 0x01) / Temperature (0x01)	CPU Temperature (Degrees)	Threshold Values: 00h : Lower Non-critical - going low 01h : Lower Non-critical - going high 02h : Lower Critical - going low 03h : Lower Critical - going high 04h : Lower Non-recoverable - going low 05h : Lower Non-recoverable - going high 06h : Upper Non-critical - going low 07h : Upper Non-critical - going high 08h : Upper Critical - going low 09h : Upper Critical - going high 0Ah : Upper Non-recoverable - going low 0Bh : Upper Non-recoverable - going high	[7:6] - 00b = unspecified byte 2 01b = trigger reading in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = trigger threshold value in byte 3 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 Bit [3:0] = Offset from Event/Reading Code for threshold event.	Reading that triggered the event, FFh or not present if unspecified. Do not confuse reading with Threshold Value	threshold value that triggered event, FFh or not present if unspecified. If present, Event Data 2 must be present
14	Temp Dual GE / PICMG AdvancedMC Module	(Analog 0x01) / Temperature (0x01)	Dual GE Temperature (Degrees)	Threshold Values: 00h : Lower Non-critical - going low 01h : Lower Non-critical - going high 02h : Lower Critical - going low 03h : Lower Critical - going high 04h : Lower Non-recoverable - going low 05h : Lower Non-recoverable - going high 06h : Upper Non-critical - going low 07h : Upper Non-critical - going high 08h : Upper Critical - going low 09h : Upper Critical - going high 0Ah : Upper Non-recoverable - going low 0Bh : Upper Non-recoverable - going high	[7:6] - 00b = unspecified byte 2 01b = trigger reading in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = trigger threshold value in byte 3 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 Bit [3:0] = Offset from Event/Reading Code for threshold event.	Reading that triggered the event, FFh or not present if unspecified. Do not confuse reading with Threshold Value	threshold value that triggered event, FFh or not present if unspecified. If present, Event Data 2 must be present
15	Vcc 1.0V / PICMG AdvancedMC Module	(Analog 0x01) / Voltage (0x02)	Voltage on 1.0V payload power supply (Volts)	Threshold Values: 00h : Lower Non-critical - going low 01h : Lower Non-critical - going high 02h : Lower Critical - going low 03h : Lower Critical - going high 04h : Lower Non-recoverable - going low 05h : Lower Non-recoverable - going high 06h : Upper Non-critical - going low 07h : Upper Non-critical - going high 08h : Upper Critical - going low 09h : Upper Critical - going high 0Ah : Upper Non-recoverable - going low 0Bh : Upper Non-recoverable - going high	[7:6] - 00b = unspecified byte 2 01b = trigger reading in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = trigger threshold value in byte 3 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 Bit [3:0] = Offset from Event/Reading Code for threshold event.	Reading that triggered the event, FFh or not present if unspecified. Do not confuse reading with Threshold Value	threshold value that triggered event, FFh or not present if unspecified. If present, Event Data 2 must be present
16	Vcc 1.0V BMC / PICMG AdvancedMC Module	(Analog 0x01) / Voltage (0x02)	**Voltage on 1.0V BMC (Volts) (only on AM4220)**	Threshold Values: 00h : Lower Non-critical - going low 01h : Lower Non-critical - going high 02h : Lower Critical - going low 03h : Lower Critical - going high 04h : Lower Non-recoverable - going low 05h : Lower Non-recoverable - going high 06h : Upper Non-critical - going low 07h : Upper Non-critical - going high 08h : Upper Critical - going low 09h : Upper Critical - going high 0Ah : Upper Non-recoverable - going low 0Bh : Upper Non-recoverable - going high	[7:6] - 00b = unspecified byte 2 01b = trigger reading in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = trigger threshold value in byte 3 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 Bit [3:0] = Offset from Event/Reading Code for threshold event.	Reading that triggered the event, FFh or not present if unspecified. Do not confuse reading with Threshold Value	threshold value that triggered event, FFh or not present if unspecified. If present, Event Data 2 must be present
17	Vcc 1.1V / PICMG AdvancedMC Module	(Analog 0x01) / Voltage (0x02)	Voltage on 1.1V payload power supply (Volts)	Threshold Values: 00h : Lower Non-critical - going low 01h : Lower Non-critical - going high 02h : Lower Critical - going low 03h : Lower Critical - going high 04h : Lower Non-recoverable - going low 05h : Lower Non-recoverable - going high 06h : Upper Non-critical - going low 07h : Upper Non-critical - going high 08h : Upper Critical - going low 09h : Upper Critical - going high 0Ah : Upper Non-recoverable - going low 0Bh : Upper Non-recoverable - going high	[7:6] - 00b = unspecified byte 2 01b = trigger reading in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = trigger threshold value in byte 3 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 Bit [3:0] = Offset from Event/Reading Code for threshold event.	Reading that triggered the event, FFh or not present if unspecified. Do not confuse reading with Threshold Value	threshold value that triggered event, FFh or not present if unspecified. If present, Event Data 2 must be present
18	Vcc 1.2V / PICMG AdvancedMC Module	(Analog 0x01) / Voltage (0x02)	Voltage on 1.2V payload power supply (Volts)	Threshold Values: 00h : Lower Non-critical - going low 01h : Lower Non-critical - going high 02h : Lower Critical - going low 03h : Lower Critical - going high 04h : Lower Non-recoverable - going low 05h : Lower Non-recoverable - going high 06h : Upper Non-critical - going low 07h : Upper Non-critical - going high 08h : Upper Critical - going low 09h : Upper Critical - going high 0Ah : Upper Non-recoverable - going low 0Bh : Upper Non-recoverable - going high	[7:6] - 00b = unspecified byte 2 01b = trigger reading in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = trigger threshold value in byte 3 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 Bit [3:0] = Offset from Event/Reading Code for threshold event.	Reading that triggered the event, FFh or not present if unspecified. Do not confuse reading with Threshold Value	threshold value that triggered event, FFh or not present if unspecified. If present, Event Data 2 must be present

Sensor ID	Sensor Name / Entity	Event/Reading Type (Class and Code) / Sensor Type (Code)	Description	Offset	Data Byte 1	Data Byte 2	Data Byte 3
19	Vcc 1.8V / PICMG AdvancedMC Module	(Analog 0x01) / Voltage (0x02)	Voltage on 1.8V payload power supply (Volts)	Threshold Values: 00h : Lower Non-critical - going low 01h : Lower Non-critical - going high 02h : Lower Critical - going low 03h : Lower Critical - going high 04h : Lower Non-recoverable - going low 05h : Lower Non-recoverable - going high 06h : Upper Non-critical - going low 07h : Upper Non-critical - going high 08h : Upper Critical - going low 09h : Upper Critical - going high 0Ah : Upper Non-recoverable - going low 0Bh : Upper Non-recoverable - going high	[7:6] - 00b = unspecified byte 2 01b = trigger reading in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = trigger threshold value in byte 3 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 Bit [3:0] = Offset from Event/Reading Code for threshold event.	Reading that triggered the event, FFh or not present if unspecified. Do not confuse reading with Threshold Value	threshold value that triggered event, FFh or not present if unspecified. If present, Event Data 2 must be present
20	Vcc 3.3V / PICMG AdvancedMC Module	(Analog 0x01) / Voltage (0x02)	Voltage on 3.3V payload power supply (Volts)	Threshold Values: 00h : Lower Non-critical - going low 01h : Lower Non-critical - going high 02h : Lower Critical - going low 03h : Lower Critical - going high 04h : Lower Non-recoverable - going low 05h : Lower Non-recoverable - going high 06h : Upper Non-critical - going low 07h : Upper Non-critical - going high 08h : Upper Critical - going low 09h : Upper Critical - going high 0Ah : Upper Non-recoverable - going low 0Bh : Upper Non-recoverable - going high	[7:6] - 00b = unspecified byte 2 01b = trigger reading in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = trigger threshold value in byte 3 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 Bit [3:0] = Offset from Event/Reading Code for threshold event.	Reading that triggered the event, FFh or not present if unspecified. Do not confuse reading with Threshold Value	threshold value that triggered event, FFh or not present if unspecified. If present, Event Data 2 must be present
21	Vcc 3.3V SUS / PICMG AdvancedMC Module	(Analog 0x01) / Voltage (0x02)	Voltage on 3.3V suspend (management) power supply (Volts)	Threshold Values: 00h : Lower Non-critical - going low 01h : Lower Non-critical - going high 02h : Lower Critical - going low 03h : Lower Critical - going high 04h : Lower Non-recoverable - going low 05h : Lower Non-recoverable - going high 06h : Upper Non-critical - going low 07h : Upper Non-critical - going high 08h : Upper Critical - going low 09h : Upper Critical - going high 0Ah : Upper Non-recoverable - going low 0Bh : Upper Non-recoverable - going high	[7:6] - 00b = unspecified byte 2 01b = trigger reading in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = trigger threshold value in byte 3 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 Bit [3:0] = Offset from Event/Reading Code for threshold event.	Reading that triggered the event, FFh or not present if unspecified. Do not confuse reading with Threshold Value	threshold value that triggered event, FFh or not present if unspecified. If present, Event Data 2 must be present
22	Vcc 3.3VEARLY / PICMG AdvancedMC Module	(Analog 0x01) / Voltage (0x02)	**Voltage on 3.3V (Volts) (only on AM4204 rev0)**	Threshold Values: 00h : Lower Non-critical - going low 01h : Lower Non-critical - going high 02h : Lower Critical - going low 03h : Lower Critical - going high 04h : Lower Non-recoverable - going low 05h : Lower Non-recoverable - going high 06h : Upper Non-critical - going low 07h : Upper Non-critical - going high 08h : Upper Critical - going low 09h : Upper Critical - going high 0Ah : Upper Non-recoverable - going low 0Bh : Upper Non-recoverable - going high	[7:6] - 00b = unspecified byte 2 01b = trigger reading in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = trigger threshold value in byte 3 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 Bit [3:0] = Offset from Event/Reading Code for threshold event.	Reading that triggered the event, FFh or not present if unspecified. Do not confuse reading with Threshold Value	threshold value that triggered event, FFh or not present if unspecified. If present, Event Data 2 must be present
23	Vcc 12V / PICMG AdvancedMC Module	(Analog 0x01) / Voltage (0x02)	Voltage on 12V payload power supply (Volts)	Threshold Values: 00h : Lower Non-critical - going low 01h : Lower Non-critical - going high 02h : Lower Critical - going low 03h : Lower Critical - going high 04h : Lower Non-recoverable - going low 05h : Lower Non-recoverable - going high 06h : Upper Non-critical - going low 07h : Upper Non-critical - going high 08h : Upper Critical - going low 09h : Upper Critical - going high 0Ah : Upper Non-recoverable - going low 0Bh : Upper Non-recoverable - going high	[7:6] - 00b = unspecified byte 2 01b = trigger reading in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = trigger threshold value in byte 3 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 Bit [3:0] = Offset from Event/Reading Code for threshold event.	Reading that triggered the event, FFh or not present if unspecified. Do not confuse reading with Threshold Value	threshold value that triggered event, FFh or not present if unspecified. If present, Event Data 2 must be present
24	Health Error / PICMG AdvancedMC Module	(Discrete 0x03) / Platform Alert (0x24)	General health status, Aggregation of critical sensor. This list is flexable and could be adjust based on customer requirements.	00h (bit 0): State Deasserted 01h (bit 1): State Asserted	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	-	-
25	Pres SFP-1 / PICMG AdvancedMC Module	(Discrete 0x6f) / Slot / Connector (0x21)	SFP-1 Slot/Connector Status	00h (bit 0): Fault Status asserted 01h (bit 1): Identify Status asserted 02h (bit 2): Slot / Connector Device installed/attached [This can include dock events] 03h (bit 3): Slot / Connector Ready for Device Installation - Typically, this means that the slot power is off. The Ready for Installation, Ready for Removal, and Slot Power states can transition together, depending on the slot implementation. 04h (bit 4): Slot/Connector Ready for Device Removal 05h (bit 5): Slot Power is Off 06h (bit 6): Slot / Connector Device Removal Request - This is typically connected to a switch that becomes asserted to request removal of the device) 07h (bit 7): Interlock asserted - This is typically connected to a switch that mechanically enables/disables power to the slot, or locks the slot in the 'Ready for Installation / Ready for Removal states' - depending on the slot implementation. The asserted state indicates that the lock-out is active. 08h (bit 8): Slot is Disabled 09h (bit 9): Slot holds spare device	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	The Event Data 2 & 3 fields can be used to provide an event extension code, with the following definition: Event Data 2: bit[7]: reserved bit[6:0]: Slot/Connector Type 0h PCI 1h Drive Array 2h External Peripheral Connector 3h Docking 4h other standard internal expansion slot 5h slot associated with entity specified by Entity ID for sensor 6h AdvancedTCA 7h DIMM/memory device 8h FAN 9h PCI Express™ Ah SCSI (parallel) Bh SATA / SAS all other = reserved	bit[7:0]: Slot/Connector Number
26	Pres SFP-2 / PICMG AdvancedMC Module	(Discrete 0x6f) / Slot / Connector (0x21)	**SFP-2 Slot/Connector Status (only on AM4204 and AM4220)**	00h (bit 0): Fault Status asserted 01h (bit 1): Identify Status asserted 02h (bit 2): Slot / Connector Device installed/attached [This can include dock events] 03h (bit 3): Slot / Connector Ready for Device Installation - Typically, this means that the slot power is off. The Ready for Installation, Ready for Removal, and Slot Power states can transition together, depending on the slot implementation. 04h (bit 4): Slot/Connector Ready for Device Removal 05h (bit 5): Slot Power is Off 06h (bit 6): Slot / Connector Device Removal Request - This is typically connected to a switch that becomes asserted to request removal of the device) 07h (bit 7): Interlock asserted - This is typically connected to a switch that mechanically enables/disables power to the slot, or locks the slot in the 'Ready for Installation / Ready for Removal states' - depending on the slot implementation. The asserted state indicates that the lock-out is active. 08h (bit 8): Slot is Disabled 09h (bit 9): Slot holds spare device	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	The Event Data 2 & 3 fields can be used to provide an event extension code, with the following definition: Event Data 2: bit[7]: reserved bit[6:0]: Slot/Connector Type 0h PCI 1h Drive Array 2h External Peripheral Connector 3h Docking 4h other standard internal expansion slot 5h slot associated with entity specified by Entity ID for sensor 6h AdvancedTCA 7h DIMM/memory device 8h FAN 9h PCI Express™ Ah SCSI (parallel) Bh SATA / SAS all other = reserved	bit[7:0]: Slot/Connector Number

Sens or ID	Sensor Name / Entity	Event/Reading Type (Class and Code) / Sensor Type (Code)	Description	Offset	Data Byte 1	Data Byte 2	Data Byte 3
27	Pres SFP-3 / PICMG AdvancedMC Module	(Discrete 0x6f) / Slot / Connector (0x21)	**SFP-3 Slot/Connector Status (only on AM4204)**	00h (bit 0): Fault Status asserted 01h (bit 1): Identify Status asserted 02h (bit 2): Slot / Connector Device installed/attached [This can include dock events] 03h (bit 3): Slot / Connector Ready for Device Installation - Typically, this means that the slot power is off. The Ready for Installation, Ready for Removal, and Slot Power states can transition together, depending on the slot implementation. 04h (bit 4): Slot/Connector Ready for Device Removal 05h (bit 5): Slot Power is Off 06h (bit 6): Slot / Connector Device Removal Request - This is typically connected to a switch that becomes asserted to request removal of the device) 07h (bit 7): Interlock asserted - This is typically connected to a switch that mechanically enables/disables power to the slot, or locks the slot in the 'Ready for Installation / Ready for Removal states' - depending on the slot implementation. The asserted state indicates that the lock-out is active. 08h (bit 8): Slot is Disabled 09h (bit 9): Slot holds spare device	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	The Event Data 2 & 3 fields can be used to provide an event extension code, with the following definition: Event Data 2: bit[7]: reserved bit[6:0]: Slot/Connector Type 0h PCI 1h Drive Array 2h External Peripheral Connector 3h Docking 4h other standard internal expansion slot 5h slot associated with entity specified by Entity ID for sensor 6h AdvancedTCA 7h DIMM/memory device 8h FAN 9h PCI Express™ Ah SCSI (parallel) Bh SATA / SAS all other = reserved	bit[7:0]: Slot/Connector Number
28	Pres SFP-4 / PICMG AdvancedMC Module	(Discrete 0x6f) / Slot / Connector (0x21)	**SFP-4 Slot/Connector Status (only on AM4204)**	00h (bit 0): Fault Status asserted 01h (bit 1): Identify Status asserted 02h (bit 2): Slot / Connector Device installed/attached [This can include dock events] 03h (bit 3): Slot / Connector Ready for Device Installation - Typically, this means that the slot power is off. The Ready for Installation, Ready for Removal, and Slot Power states can transition together, depending on the slot implementation. 04h (bit 4): Slot/Connector Ready for Device Removal 05h (bit 5): Slot Power is Off 06h (bit 6): Slot / Connector Device Removal Request - This is typically connected to a switch that becomes asserted to request removal of the device) 07h (bit 7): Interlock asserted - This is typically connected to a switch that mechanically enables/disables power to the slot, or locks the slot in the 'Ready for Installation / Ready for Removal states' - depending on the slot implementation. The asserted state indicates that the lock-out is active. 08h (bit 8): Slot is Disabled 09h (bit 9): Slot holds spare device	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	The Event Data 2 & 3 fields can be used to provide an event extension code, with the following definition: Event Data 2: bit[7]: reserved bit[6:0]: Slot/Connector Type 0h PCI 1h Drive Array 2h External Peripheral Connector 3h Docking 4h other standard internal expansion slot 5h slot associated with entity specified by Entity ID for sensor 6h AdvancedTCA 7h DIMM/memory device 8h FAN 9h PCI Express™ Ah SCSI (parallel) Bh SATA / SAS all other = reserved	bit[7:0]: Slot/Connector Number
29	POST Value / PICMG AdvancedMC Module	(Discrete 0x6f) / OEM POST Value Sensor (0xc6)	Show current postcode value. No event generated by this sensor.	00h to 07h (bit[0:7]): Post Code low byte value 14h (bit 14): Post code Error All other offset are unused. Only offset 14h triggers an event	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	If offset 14h: POST Low Nibble	If offset 14h: POST High Nibble (always 00h)
30	POST Error / PICMG AdvancedMC Module	(Discrete 0x6f) / System Firmwares (0x0f)	CPU Power On Self Test Error	00h (bit 0): System Firmware Error (POST Error) 01h (bit 1): System Firmware Hang 02h (bit 2): System Firmware Progress	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	See Sensor Specific Event (Annex B)	-
31	Boot Error / PICMG AdvancedMC Module	(Discrete 0x6f) / Boot Error (0x1e)	Boot Error	00h (bit 0): No bootable media 01h (bit 1): Non-bootable diskette left in drive 02h (bit 2): PXE Server not found 03h (bit 3): Invalid boot sector 04h (bit 4): Timeout waiting for user selection of boot source	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	[7:4] - Optional offset from 'Severity' Event/Reading Code. (0Fh if unspecified). [3:0] - Optional offset from Event/Reading Type Code for previous discrete event state. (0Fh if unspecified.)	-
32	Diag Status / PICMG AdvancedMC Module	(Discrete 0x6f) / OEM OEM Diagnostic Status (0xc9)	OEM Kontron Diagnostic Status	00h (bit 0): Diagnostic Started 01h (bit 1): Diagnostic PASS 02h (bit 2): Diagnostic FAIL	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	-	-
33	Fwupg Status / PICMG AdvancedMC Module	(Discrete 0x6f) / OEM Component Firmware Upgrade (0xca)	OEM Kontron External Component Firmware Upgrade Status	00h (bit 0): Firmware Upgrade in Progress (no event) 01h (bit 1): Firmware upgrade succeeded 02h (bit 2): Firmware upgrade failed	[7:6] - 00b = unspecified byte 2 01b = previous state and/or severity in byte 2 10b = OEM code in byte 2 11b = sensor-specific event extension code in byte 2 [5:4] - 00b = unspecified byte 3 01b = reserved 10b = OEM code in byte 3 11b = sensor-specific event extension code in byte 3 [3:0] - Offset from Event/Reading Code for discrete event state	-	-

Annex B – Sensor-Specific Event

Sensor Type	Sensor Type Code	Sensor Offset	Event Data 2	Event Data 3
System Firmware Progress (formerly POST Error)	0Fh	00h	00h Unspecified. 01h No system memory is physically installed in the system. 02h No usable system memory, all installed memory has experienced an unrecoverable failure. 03h Unrecoverable hard-disk/ATAPI/IDE device failure. 04h Unrecoverable system-board failure. 05h Unrecoverable diskette subsystem failure. 06h Unrecoverable hard-disk controller failure. 07h Unrecoverable PS/2 or USB keyboard failure. 08h Removable boot media not found 09h Unrecoverable video controller failure 0Ah No video device detected 0Bh Firmware (BIOS) ROM corruption detected 0Ch CPU voltage mismatch (processors that share same supply have mismatched voltage requirements) 0Dh CPU speed matching failure 0Eh to FFh reserved	
		01h	uses same Event Data 2 definition as following System Firmware Progress offset	
		02h	00h Unspecified. 01h Memory initialization. 02h Hard-disk initialization 03h Secondary processor(s) initialization 04h User authentication 05h User-initiated system setup 06h USB resource configuration 07h PCI resource configuration 08h Option ROM initialization 09h Video initialization 0Ah Cache initialization 0Bh SM Bus initialization 0Ch Keyboard controller initialization 0Dh Embedded controller/management controller initialization 0Eh Docking station attachment 0Fh Enabling docking station 10h Docking station ejection 11h Disabling docking station 12h Calling operating system wake-up vector 13h Starting operating system boot process, e.g. calling Int 19h 14h Baseboard or motherboard initialization 15h reserved 16h Floppy initialization 17h Keyboard test 18h Pointing device test 19h Primary processor initialization 1Ah to FFh reserved	
Management Subsystem Health	28h	00h	-	-
		01h	-	-
		02h	-	-
		03h	-	-
		04h	[7:0] - Sensor Number. Number of the failed sensor corresponding to event offset 04h or 00h.	-
		05h	[7] - logical/physical FRU device 0b = device is not a logical FRU Device 1b = device is logical FRU Device (accessed via FRU commands to mgmt. controller) [6:5] - reserved. [4:3] - LUN for Master Write-Read command or FRU Command. 00b if device is non-intelligent device directly on IPMB. [2:0] - Private bus ID if bus = Private. 000b if device directly on PMB, or device is a logical FRU Device.	For LOGICAL FRU DEVICE (accessed via FRU commands to mgmt. controller): [7:0] - FRU Device ID within controller that generated the event.FFh = reserved. For non-intelligent FRU device: [7:1] - 7-bit I2C Slave Address of FRU device . This is relative to the bus the device is on. For devices on the IPMB, this is the slave address of the device on the IPMB. For devices on a private bus, this is the slave address of the device on the private bus. [0] - reserved.

1. To track the relationship between timestamps, the timestamp change events should be logged in pairs - the first event being logged just before the timestamp clock update followed by a second event that is logged after the timestamp clock has been updated. This enables software that reads the SEL to be able to determine time relationship between events that were logged before the update and those logged afterward. The generation of these events is normally the responsibility of the software that changes the timestamp clock. Note that some implementations may queue events prior to their being logged. It is recommended that generic software read the SEL to verify that the first event has been recorded with the relative timestamp before setting the new timestamp value and generating the second event.

Annex C – Cause of State Change Values

Cause offset	Cause of State Change Description
0h	Normal State Change. This is used when the FRU is proceeding normally through the state chart. For instance, an M3 to M4 transition is a normal state change. Other values in this table can be used to provide greater levels of detail about what initiated a transition. Valid for the M0 to M1, M1 to M2, M2 to M3, M3 to M4, M4 to M5, M5 to M6, and M6 to M1 transitions.
1h	Change Commanded by Shelf Manager with Set FRU Activation. The Shelf Manager has issued a command to change states, typically during an insertion or extraction. Valid for the M2 to M1, M2 to M3, M4 to M6, M5 to M4, and M5 to M6 transitions.
2h	State Change due to operator changing a Handle Switch. The FRU has changed states as a result of an operator changing the state of a Handle Switch. Valid for the M1 to M2, M3 to M6, M4 to M5, and M5 to M4 transitions.
3h	State Change due to FRU programmatic action. The FRU has changed states due to some non-operator related internal requirement (such as Locked bit being cleared). Valid for the M1 to M2, M3 to M6, M4 to M5, and M5 to M4 transitions
4h	Communication Lost or Regained. The Shelf Manager has lost or regained contact with the FRU and generated an event on its behalf. Valid for the M2 to M7, M3 to M7, M4 to M7, M5 to M7, and M6 to M7, M7 to M1, M7 to M2, M7 to M3, M7 to M4, M7 to M5 and M7 to M6 transitions.
5h	Communication Lost or Regained–locally detected. The FRU has changed state as a result of an internal detection by the IPM Controller. This is only valid for FRUs represented by a physically separate IPM Controller (e.g., mezzanine cards). Valid for the M2 to M7, M3 to M7, M4 to M7, M5 to M7, M6 to M7, M7 to M1, M7 to M2, M7 to M3, M7 to M4, M7 to M5 and M7 to M6 transitions.
6h	Surprise State Change due to extraction. The FRU has changed state abruptly to M0 due to a non-compliant removal from the system. This is only valid for FRUs represented by a physically separate IPM Controller (e.g., mezzanine cards). Valid for the M2 to M0, M3 to M0, M4 to M0, M5 to M0, M6 to M0, and M7 to M0 transitions.
7h	State Change due to provided information. A new state is known for the FRU that could not be deduced previously. This is used when a user verifies that a FRU has been extracted from the Shelf and is no longer available. Valid for the M7 to M0 state transition.
8h	Invalid Hardware Address Detected. This is an error condition where the Hardware Address did not pass the parity check. Valid for the M0 to M0 transition.
9h	Unexpected Deactivation. The FRU has transitioned to deactivating without requesting permission from the Shelf Manager first. Valid for M4 to M6 transition.
Ah	Surprise State Change due to power failure. Case 1: The FRU has abruptly changed state to M0 due to a serious power failure that precludes further use of the FRU until it is extracted and reinserted or replaced (for example, due to overcurrent in the Management Power domain). This is valid for M0, M1, M2, M3, M4, M5, M6, and M7 to M0 transitions. Case 2: The FRU has cut off the power to its Payload or a managing FRU has cut off the Payload Power to its Managed FRU due to an unexpected power failure that does not preclude further use of the FRU (for example, due to overcurrent in the Payload Power domain). This is valid for M3, M4, M5, M7 to M6, and M6, M7 to M1 transitions.
Fh	State Change, Cause Unknown. No cause could be determined.
All other values	Reserved