SAM1411: Control via IPMI

Programmer's Reference

P/N: 6806800N05C

January 2020

Trademarks

The stylized "S" and "SMART" is a registered trademark of SMART Modular Technologies, Inc. and “SMART Embedded Computing” and the SMART Embedded Computing logo are trademarks of SMART Modular Technologies, Inc. All other names and logos referred to are trade names, trademarks, or registered trademarks of their respective owners. These materials are provided by SMART Embedded Computing as a service to its customers and may be used for informational purposes only.

Disclaimer*

SMART Embedded Computing (SMART EC) assumes no responsibility for errors or omissions in these materials. **These materials are provided "AS IS" without warranty of any kind, either expressed or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.** SMART EC further does not warrant the accuracy or completeness of the information, text, graphics, links or other items contained within these materials. SMART EC shall not be liable for any special, indirect, incidental, or consequential damages, including without limitation, lost revenues or lost profits, which may result from the use of these materials. SMART EC may make changes to these materials, or to the products described therein, at any time without notice. SMART EC makes no commitment to update the information contained within these materials.

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 The text itself may not be published commercially in print or electronic form, edited, translated, or otherwise altered without the permission of SMART EC.

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

Limited and Restricted Rights Legend

If the documentation contained herein is supplied, directly or indirectly, to the U.S. Government, the following notice shall apply unless otherwise agreed to in writing by SMART Embedded Computing.

Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data clause at DFARS 252.227-7013 (Nov. 1995) and of the Rights in Noncommercial Computer Software and Documentation clause at DFARS 252.227-7014 (Jun. 1995).

SMART Embedded Computing, Inc.

2900 S. Diablo Way, Suite 190

Tempe, Arizona 85282

USA

*For full legal terms and conditions, visit *www.smartembedded.com/ec/legal*

Table of Contents

List of Tables

List of Tables

About this Manual

Overview of Contents

This manual is intended for users such as system designers and system integrators qualified in electronics or electrical engineering. Users must have a working understanding of system design, Advanced Telecom Computing Architecture (AdvancedTCA) design and specifications, Intelligent Platform Management Interface (IPMI), and telephony telecommunications.

It provides a detailed description of default values of FRU information and sensor data records (SDRs) contained on your SAM1411 board and contains the following chapters and appendix:

Chapter 1, Introduction on page 11 gives a short overview on the scope of the document.

Chapter 2, Supported Commands on page 13 lists IPMI commands supported by the IPMC.

Chapter 3, FRU Information and Sensor Data Records on page 29 lists the blade's FRU information as well as sensors that are accessible via IPMI.

Appendix A, Related Documentation on page 41 lists related documentation and specifications.

Abbreviations

This document uses the following abbreviations:

Abbreviation	Definition
AdvancedTCA	Advanced Telecom Computing Architecture
AMC	Advanced Mezzanine Card
ASF	Alert Standard Forum
BCD	Binary-Coded Decimal
BIB	Board Information Block
BIOS	Basic Input/Output System
BMC	Baseboard Management Controller
CMC	Common Mezzanine Card
CO	Central Office
FPGA	Field Programmable Gate-Array
FRU	Field Replaceable Unit

Abbreviation	Definition
FW	Firmware
HPI	Hardware Platform Interface
IANA	Internet Assigned Numbers Authority
ID	Identifier
IMC	IPM Master Controller
IPMB	Intelligent Platform Management Bus
IPMC	Intelligent Platform Management Interface Controller
IPMI	Intelligent Platform Management Interface
ISC	IPM Slave Controller
LAN	Local Area Network
LSB	Least Significant Bit
LUN	Logical Unit Number
MAC	Media Access Control
MSB	Most Significant Bit
OEM	Original Equipment Manufacturer
OS	Operating System
PEM	Power Entry Module
PICMG	PCI Industrial Computer Manufacturers Group
POST	Power On Self Test
RTM	Rear Transition Module
SDR	Sensor Data Record
SEL	System Event Log
SOL	Serial over LAN

Conventions

The following table describes the conventions used throughout this manual.

Notation	Description
0x00000000	Typical notation for hexadecimal numbers (digits are 0 through F), for example used for addresses and offsets
0b0000	Same for binary numbers (digits are 0 and 1)
bold	Used to emphasize a word
Screen	Used for on-screen output and code related elements or commands. `Sample of Programming used in a table (9pt)`
`Courier + Bold`	Used to characterize user input and to separate it from system output
Reference	Used for references and for table and figure descriptions
File > Exit	Notation for selecting a submenu
<text>	Notation for variables and keys
[text]	Notation for software buttons to click on the screen and parameter description
...	Repeated item for example node 1, node 2, ..., node 12
. . .	Omission of information from example/command that is not necessary at the time
..	Ranges, for example: 0..4 means one of the integers 0,1,2,3, and 4 (used in registers)
|	Logical OR
	Indicates a hazardous situation which, if not avoided, could result in death or serious injury
	Indicates a hazardous situation which, if not avoided, may result in minor or moderate injury

Notation	Description
	Indicates a property damage message
	Indicates a hot surface that could result in moderate or serious injury
	Indicates an electrical situation that could result in moderate injury or death
	Indicates that when working in an ESD environment care should be taken to use proper ESD practices
	No danger encountered, pay attention to important information

Summary of Changes

Document Number	Date	Description
6806800N05C	January 2020	Rebrand to SMART Embedded Computing template
6806800N05B	August 2014	Re-branded to Artesyn
6806800N05A	September 2011	First version

Introduction

1.1 Overview

This document provides a detailed description of default values of FRU information and sensor data records (SDRs) contained on SAM1411 board.

For details on the IPMI commands that are used to read FRU information and SDRs refer to the IPMI Specification Version 1.5.

The board provides the following information:

Feature	Available
Provide sensor data records	yes
Provide FRU inventory	yes
Provide system event log	yes
Event generator	yes
SDR repository	yes

Supported Commands

2.1 Standard IPMI Commands

The IPMC is fully compliant to the Intelligent Platform Management Interface v.1.5. This section provides information on which IPMI commands are supported.

2.1.1 Global IPMI Commands

The IPMC supports the following global IPMI commands.

Table 2-1 Supported Global IPMI Commands

Command	NetFn (Request/Response)	CMD	Comments
Get Device ID	0x06/0x07	0x01	-
Cold Reset	0x06/0x07	0x02	-
Warm Reset	0x06/0x07	0x03	-
Master Write-Read	0x06/0x07	0x52	Only for accessing private I2C buses.

2.1.2 System Interface Commands

Table 2-2 Supported System Interface Commands

Command	NetFn (Request/Response)	CMD
Set BMC Global Enables	0x06/0x07	0x2E
Get BMC Global Enables	0x06/0x07	0x2F
Clear Message Flags	0x06/0x07	0x30
Get Message Flags	0x06/0x07	0x31
Enable Message Channel Receive	0x06/0x07	0x32
Get Message	0x06/0x07	0x33
Send Message	0x06/0x07	0x34
Read Event Message Buffer	0x06/0x07	0x35
Get BT Interface Capabilities	0x06/0x07	0x36

2.1.3 Watchdog Commands

Table 2-3 Supported Watchdog Commands

Command	NetFn (Request/Response)	CMD
Reset Watchdog Timer	0x06/0x07	0x22
Set Watchdog Timer	0x06/0x07	0x24
Get Watchdog Timer	0x06/0x07	0x25

2.1.4 SEL Device Commands

Table 2-4 Supported SEL Device Commands

Command	NetFn (Request/Response)	CMD	Comments
Get SEL Info	0x0A/0x0B	0x40	-
Get SEL Allocation Info	0x0A/0x0B	0x41	-
Reserve SEL	0x0A/0x0B	0x42	-
Get SEL Entry	0x0A/0x0B	0x43	-
Add SEL Entry	0x0A/0x0B	0x44	-
Delete SEL Entry	0x0A/0x0B	0x46	-
Clear SEL	0x0A/0x0B	0x47	-
Get SEL Time	0x0A/0x0B	0x48	Since the Set SDR Repository Time command is not supported, the SEL and SDR repository of the SMART Embedded Computing IPMC share the same time stamp. Therefore, the Get SDR Repository Time and Get SEL Time commands return the same time stamp.
Set SEL Time	0x0A/0x0B	0x49	Since the Set SDR Repository Time command is not supported, the SEL and SDR repository of the SMART Embedded Computing IPMC share the same time stamp. Therefore, the Set SEL Time command sets the time stamp for the SEL and the SDR repository.

2.1.5 SDR Repository Commands

Table 2-5 Supported SDR Repository Commands

Command	NetFn (Request/Response)	CMD	Comments
Get SDR Repository Info	0x0A/0x0B	0x20	-
Get SDR Repository Allocation Info	0x0A/0x0B	0x21	-
Reserve SDR Repository	0x0A/0x0B	0x22	-
Get SDR	0x0A/0x0B	0x23	-
Add SDR	0x0A/0x0B	0x24	-
Partial Add SDR	0x0A/0x0B	0x25	
Delete SDR	0x0A/0x0B	0x26	
Clear SDR Repository	0x0A/0x0B	0x27	
Get SDR Repository Time	0x0A/0x0B	0x28	Since the Set SDR Repository Time command is not supported, the SEL and SDR repository of the SMART Embedded Computing IPMC share the same time stamp. Therefore, the Get SDR Repository Time and Get SEL Time commands return the same time stamp.

2.1.6 FRU Inventory Commands

Table 2-6 Supported FRU Inventory Commands

Command	NetFn (Request/Response)	CMD	Comments
Get FRU Inventory Area Info	0x0A/0x0B	0x10	-
Read FRU Data	0x0A/0x0B	0x11	-
Write FRU Data	0x0A/0x0B	0x12	Address table, Board Power Distribution Record of FRU ID 0.

2.1.7 Sensor Device Commands

Table 2-7 Supported Sensor Device Commands

Command	NetFn (Request/Response)	CMD	Comments
Get Device SDR Info	0x04/0x05	0x20	-
Get Device SDR	0x04/0x05	0x21	-
Reserve Device SDR Repository	0x04/0x05	0x22	-
Get Sensor Hysteresis	0x04/0x05	0x25	-
Set Sensor Threshold	0x04/0x05	0x26	Most of the threshold-based sensors have fixed thresholds. Before using this command, check whether threshold setting is supported by using the Get Device SDR command.
Get Sensor Threshold	0x04/0x05	0x27	-
Set Sensor Event Enable	0x04/0x05	0x28	-
Get Sensor Event Enable	0x04/0x05	0x29	-
Rearm Sensor Events	0x04/0x05	0x2A	-
Get Sensor Event Status	0x04/0x05	0x2B	-
Get Sensor Reading	0x04/0x05	0x2D	-
Get Sensor Type	0x04/0x05	0x2F	-
Set Event Receiver	0x04/0x05	0x00	-

Table 2-7 Supported Sensor Device Commands (continued)

Command	NetFn (Request/Response)	CMD	Comments
Get Event receiver	0x04/0x05	0x01	-
Platform Event	0x04/0x05	0x02	The SAM1411 can write events to the SEL repository and acknowledge them.

2.1.8 LAN Device Commands

Table 2-8 Supported LAN Device Commands

Command	NetFn (Request/Response)	CMD
Get Channel Authentication Capabilities	0x0C/0x0D	0x38
Get Session Challenge	0x0C/0x0D	0x39
Activate Session	0x0C/0x0D	0x3A
Set Session Privilege Level	0x0C/0x0D	0x3B
ASF Presence Ping Message	-	-
ASF Presence Pong Message	-	-

2.2 PICMG 3.0 Commands

The SMART EC IPMC is a fully compliant AdvancedTCA Intelligent Platform Management Controller i.e., it supports all required and mandatory AdvancedTCA commands as defined in the PICMG 3.0 and AMC 0.9x specifications.

Table 2-9 Supported PICMG 3.0 Commands

Command	NetFn (Request/Response)	CMD	Comments
Get PICMG Properties	0x2C/0x2D	0x00	-
Get Address Info	0x2C/0x2D	0x01	-
Get Shelf Address Info	0x2C/0x2D	0x02	-
Set Shelf Address Info	0x2C/0x2D	0x03	-

Table 2-9 Supported PICMG 3.0 Commands (continued)

Command	NetFn (Request/Response)	CMD	Comments
FRU Control	0x2C/0x2D	0x04	The SAM1411 only supports the cold reset option.
Get FRU LED Properties	0x2C/0x2D	0x05	-
Get FRU LED Color Capabilities	0x2C/0x2D	0x06	-
Set FRU LED State	0x2C/0x2D	0x07	-
Get FRU LED State	0x2C/0x2D	0x08	-
Set IPMB State	0x2C/0x2D	0x09	-
Set FRU Activation Policy	0x2C/0x2D	0x0A	-
Get FRU Activation Policy	0x2C/0x2D	0x0B	-
Set FRU Activation	0x2C/0x2D	0x0C	-
Get Device Locator Record ID	0x2C/0x2D	0x0D	The SMART Embedded Computing IPMCs support the standard PICMG 3.0 and the extended AMC 0.9x versions of this command.
Set Port State	0x2C/0x2D	0x0E	-
Get Port State	0x2C/0x2D	0x0F	-
Compute Power Properties	0x2C/0x2D	0x10	-
Set Power Level	0x2C/0x2D	0x11	-
Get Power Level	0x2C/0x2D	0x12	-
Renegotiate Power	0x2C/0x2D	0x13	-
Bused Resource	0x2C/0x2D	0x17	-
Get IPMB Link Info	0x2C/0x2D	0x18	-

2.3 SMART EC Specific Commands

The SMART Embedded Computing IPMC supports several commands which are not defined in the IPMI or PICMG 3.0 specification but are introduced by SMART EC: Firmware upgrade and status change commands.

Before sending any of these commands, the shelf management software must check whether the receiving IPMI controller is an SMART EC IPMI controller, that means IPMC, by using the IPMI command 'Get Device ID'. Sending SMART EC specific commands to IPMI controllers which are not delivered by SMART EC leads to no or undefined results.

Implementing any of the SMART EC specific IPMI commands means that the software is not portable to other IPMI controllers that do not use the SMART EC IPMC firmware.

Make sure to use these commands with care.

2.3.1 Firmware Upgrade Commands

SMART EC offers three commands to upgrade the IPMC firmware which can be used to write an upgrade function:

- Start Firmware Upgrade
- Continue Firmware Upgrade
- Finish Firmware Upgrade

The firmware upgrade session has to start with the Start Firmware Upgrade command which makes the target IPMC enter the firmware upgrade mode. The firmware image is sent to the target IPMC in several parts with multiple Continue Firmware Upgrade commands. Each part can have the size of an IPMB message length. When the whole firmware image is on the target IPMC, the process has to be finished with the Finish Firmware Upgrade command. During the firmware upgrade mode, the SMART Embedded Computing IPMC may only execute the Continue Firmware Upgrade and Get Device ID commands.

The following table shows the firmware upgrade commands together with their network function and command code.

Table 2-10 Firmware Upgrade Commands

Command Name	NetFn (Request/Response)	CMD	Description
Start Firmware Upgrade	0x08/0x09	0x1B	See *Start Firmware Upgrade on page 20*
Continue Firmware Upgrade	0x08/0x09	0x1C	See *Continue Firmware Upgrade on page 20*
Finish Firmware Upgrade	0x08/0x09	0x1E	See *Finish Firmware Upgrade on page 21*

2.3.1.1 Start Firmware Upgrade

The Start Firmware Upgrade command puts the target IPMC into firmware upgrade mode. Only the Firmware Upgrade commands and the Get Device ID command are supported in firmware upgrade mode.

2.3.1.1.1 Request Data

No request data needs to be provided for this command.

2.3.1.1.2 Response Data

The following table lists the response data applicable to the Start Firmware Upgrade command.

Table 2-11 Response Data of Start Firmware Upgrade

Byte	Data Field
1	Completion Code 0x00: Command executed successfully and target IPMC entered firmware upgrade mode 0x01..0xFF: Error, that means IPMC cannot enter into firmware upgrade mode

2.3.1.2 Continue Firmware Upgrade

The Continue Firmware Upgrade command writes a part of the firmware image to the target IPMC. It also checks file integrity and makes the target IPMC leave the firmware upgrade mode if an error occurs. If an error occurs, the whole firmware upgrade sequence must be repeated beginning from the Start Firmware Upgrade command and the whole firmware upgrade image must be retransmitted.

2.3.1.2.1 Request Data

The following table lists the request data applicable to the Continue Firmware Upgrade command.

Table 2-12 Request Data of Continue Firmware Upgrade

Byte	Data Field
1..23	Firmware content to be sent to the target IPMC. The firmware image is an extended INTEL hex file. The whole message length is defined by the maximum IPMB message length.

2.3.1.2.2 Response Data

The following table lists the response data of the Continue Firmware Upgrade command.

Table 2-13 Response Data of Continue Firmware Upgrade

Byte	Data Field
1	Completion Code 0x00: Command executed successfully 0x1..0xFF: Error, that means the IPMC left the firmware upgrade mode

2.3.1.3 Finish Firmware Upgrade

The Finish Firmware Upgrade command makes the target IPMC leave the firmware upgrade mode.

2.3.1.3.1 Request Data

The following table lists the request data applicable to the Finish Firmware Upgrade command.

Table 2-14 Request Data of Finish Firmware Upgrade

Byte	Data Field
1..23	None

2.3.1.3.2 Response Data

The following table lists the response data applicable to the Finish Firmware Upgrade command.

Table 2-15 Response Data of Finish Firmware Upgrade

Byte	Data Field
1	Completion Code 0: Command executed successfully 0x01..0xFF: Error

2.3.2 Status Change Commands

The following table shows all SMART Embedded Computing specific status change commands together with their network function and command code.

Table 2-16 Status Change Commands

Command Name	NetFn (Request/Response)	CMD	Description
Set Discrete Sensor Value	0x30/0x31	0x01	See *Set Discrete Sensor Value on page 22*
Set Analog Sensor Value	0x30/0x31	0x02	See *Set Analog Sensor Value on page 23*
Change Active/Standby Role	0x30/0x31	0x03	See *Change Active/Standby Role on page 24*

2.3.2.1 Set Discrete Sensor Value

This command is used to set the value of an SMART EC Original Equipment Manufacturer (OEM) discrete sensor. It can only be used on sensors where the sensor description in this document explicitly states it.

2.3.2.1.1 Request Data

The following table lists the request data applicable to the Set Discrete Sensor Value command.

Table 2-17 Request Data of Set Discrete Sensor Value

Byte	Bits	Data Field
1	0..7	Sensor number (0xFF is reserved)
2..3	15	Reserved

Table 2-17 Request Data of Set Discrete Sensor Value (continued)

Byte	Bits	Data Field
	0..14	Change mask (LSB first) 0: State remains unchanged 1: State is changed
4..5	15	Reserved
	0..14	Value to be written into the discrete sensor. For further information, refer to the row of the sensor description tables.

2.3.2.1.2 Response Data

The following table lists the response data of the Set Discrete Sensor Value command.

Table 2-18 Response Data of Set Discrete Sensor Value

Byte	Data Field
1	Completion Code 0x00: Command was executed successfully 0x01..0xFF: Error

2.3.2.2 Set Analog Sensor Value

This command is used to set the value of an analog sensor. It can only be used on sensors where the sensor description in this document explicitly states it.

2.3.2.2.1 Request Data

The following table lists the request data of the Set Discrete Sensor Value command.

Table 2-19 Request Data of Set Analog Sensor Value

Byte	Bits	Data Field
1	0..7	Sensor number (0xFF is reserved)
2	0..7	Value to be written into the analog sensor.

2.3.2.2.2 Response Data

The following table lists the response data of the Set Discrete Sensor Value command.

Table 2-20 Response Data of Set Analog Sensor Value

Byte	Data Field
1	Completion Code 0x00: Command was executed successfully 0x01..0xFF: Error

2.3.2.3 Change Active/Standby Role

This command sets the role of the IPMC on the SAM1411 performing shelf management functions to active or standby.

2.3.2.3.1 Request Data

The following table lists the request data of the Change Active/Standby Role command.

Table 2-21 Request Data of Change Active/Standby Role

Byte	Bits	Data Field
1	0..7	Role: 0: Active 2: Standby

2.3.2.3.2 Response Data

The following table lists the response data of the Change Active/Standby Role command.

Table 2-22 Response Data of Change Active/Standby Role

Byte	Data Field
1	Completion Code 0x00: Command was executed successfully 0x01..0xFF: Error

2.3.3 Boot Bank Swap Commands

The following table lists the boot bank swap commands.

Table 2-23 Boot Bank Swap Commands

Command Name	NetFn (Request/Response)	CMD	Description
Get Upgrade Flag	0x30/0x31	0xE2	See *Get Upgrade Flag on page 25*
Set Upgrade Flag	0x30/0x31	0xE3	See *Set Upgrade Flag on page 26*
Get Swap Flag	0x30/0x31	0xE4	See *Get Swap Flag on page 26*
Set Swap Flag	0x30/0x31	0xE5	See *Set Swap Flag on page 27*

2.3.3.1 Get Upgrade Flag

The Get Upgrade Flag command returns the current value of the upgrade flag. This flag indicates that an upgrade has just taken place on the current boot bank. This information is used in conjunction with the swap flag below to determine whether or not to swap boot banks on an IPMI watchdog timeout and subsequent restart. This flag will then be cleared after this boot bank swap is performed.

2.3.3.1.1 Request Data

No request data needs to be provided for this command.

2.3.3.1.2 Response Data

The following table lists the response data of the Get Upgrade Flag command.

Table 2-24 Response Data of Get Upgrade Flag Command

Byte	Data Field
1	Completion Code 0x00: Command was executed successfully 0x01..0xFF: Error mode
2	Value of the Upgrade Flag 0x00: Upgrade not performed immediately prior to previous boot 0x01: Upgrade performed immediately prior to previous boot

2.3.3.2 Set Upgrade Flag

The Set Upgrade Flag command sets the value of the upgrade flag. This can be used to indicate an upgrade has been performed on the current boot bank. The value of this flag will automatically be set to 0 after an IPMC reset or in the event of an IPMI watchdog timeout and boot bank swap.

2.3.3.2.1 Request Data

The following table lists the request data of the Set Upgrade Flag command.

Table 2-25 Request Data of Set Upgrade Flag Command

Byte	Data Field
1	Value of the Upgrade Flag 0x00: Indicates an upgrade has not been performed 0x01: Indicates an upgrade has been performed

2.3.3.2.2 Response Data

The following table lists the response data of the Set Upgrade Flag command.

Table 2-26 Response data of the Set Upgrade Flag command

Byte	Data Field
1	Completion Code 0x00: Command was executed successfully 0x01..0xFF: Error mode

2.3.3.3 Get Swap Flag

The Get Swap Flag command returns the current value of the swap flag. This flag determines whether or not a boot bank swap is dependent on the upgrade flag. If this flag is set to 1 and the upgrade flag is set to 0, there will be no boot bank swap on an IPMI watchdog timeout. All other combinations of these two flags will result in a boot bank swap before a restart.

2.3.3.3.1 Request Data

No request data needs to be provided for this command.

2.3.3.3.2 Response Data

The following table lists the response data of Get Swap Flag command.

Table 2-27 Response Data of Get Swap Flag Command

Byte	Data Field
1	Completion Code 0x00: Command was executed successfully 0x01..0xFF: Error mode
2	Value of the Swap Flag 0x00: No boot bank swap performed on IPMI watchdog timeout 0x01: Boot Bank Swap performed on IPMI watchdog timeout after IPMC upgrade

2.3.3.4 Set Swap Flag

The Set Swap Flag command sets the value of the swap flag. This can be used to indicate that a boot bank swap should or should not be performed on a IPMI watchdog timeout after an upgrade. This is a persistent flag and will retain its value after an IPMC reset.

2.3.3.4.1 Request Data

The following table lists the request data applicable to the Set Swap Flag command.

Table 2-28 Request Data of Set Swap Flag Command

Byte	Data Field
1	Value of the Swap Flag 0x00: Boot bank swap will always occur after an IPMI watchdog timeout. 0x01: Boot bank swap will only occur after an IPMI watchdog timeout if the upgrade flag is set.

2.3.3.4.2 Response Data

The following table lists the response data applicable to the Set Swap Flag command.

Table 2-29 Response Data of Set Swap Flag Command

Byte	Data Field
1	Completion Code 0x00: Command was executed successfully 0x01..0xFF: Error mode

FRU Information and Sensor Data Records

3.1 FRU Information

The SAM1411 board is a dedicated shelf manager. It supports FRU ID 0 and FRU ID 254 according to the AdvancedTCA specification, PICMG 3.0.

FRU ID 254 consists of cashed shelf FRU data. The HPI system management determines the content of FRU ID 254.

FRU ID 0 consists of the SAM1411 data. It provides the following FRU information.

Table 3-1 FRU Information, FRU ID 0

Area	Description	Value	Access
Internal use area	not used		
Board info area	Manufacturing date/time	According to Intel's Platform Management FRU information Storage Definition v1.0	r
	Board manufacturer	SMART EC	r
	Board product name	Defined by SMART EC	r
	Board serial number	Defined by SMART EC	r
	Board part number	Defined by SMART EC	r
Product info area	Product manufacturer	SMART EC	r
	Product name	Defined by SMART EC	r
	Product serial number	Defined by SMART EC	r
	Product part number	Defined by SMART EC	r
	Product Version	Defined by SMART EC	r
Multi record area	Blade Point-To-Point Connectivity Record	The Multi record Area contains the ATCA-blade Point to Point Connectivity Record according to PICMG 3.0, Rev.1.0 Type: Ethernet, Channel: Base 1; Port: Port 0 Type: Ethernet, Channel: Base 2; Port: Port 0	r

Table 3-1 FRU Information, FRU ID 0 (continued)

Area	Description	Value	Access
	User Info Area	SMART Embedded Computing OEM user info area record ID: 0x48, 0x0E, 0x00, 0x00 followed by 251 byte of user info area data	r/w
	Custom usage	Min. 256 byte available	r/w

3.2 MAC Address Record

The blade provides one OEM FRU record which contains information about on-board MAC addresses. The MAC address information is stored in either of the following two ways:

- The MAC address record holds the total number of on-board MAC addresses and the base MAC address
- The MAC address record holds all on-board MAC addresses

The format of the record is described in the following table.

Table 3-2 MAC Address Record

Offset	Length	Description
0	1	Record type ID. value is: 0xC0 (OEM)
1	1	End of List/Version [7]: End of list. Set to one for the last record [6:4]: Reserved [3:0]: Record format version. Value for MAC address record is: 0x2
2	1	Record Length
3	1	Record Checksum. Holds the zero checksum of the record.
4	1	Header Checksum. Holds the zero checksum of the header.
5	3	Manufacturer ID. LS byte first. Value is: 0x000E48.
8	1	Record ID. For the MAC Address Record, the value is 0x11.
9	1	Record format version. Value is: 0x0.

Table 3-2 MAC Address Record (continued)

Offset	Length	Description
Case 1: the MAC address record holds the MAC address count and the base address:		
10	1	Total number of on-board MAC addresses
11	6	Base MAC address
Case 2: the MAC address record holds all MAC addresses		
10	n * 6	n MAC addresses (MS byte first in each MAC address)

3.3 Sensor Data Records

The SAM1411 board provides the following sensors:

Table 3-3 Sensor Data Records

Sensor Number	Sensor Name	Sensor Type	Event Data Byte 1	Event Data Byte 2	Event Data Byte 3	Event Threshold/ Description	Event Reading Type
1(0x01)	SAM1411 IPMC	OEM 0xD5	OEM 0xD5	0x0 0x1 0x2 0x3 0x4 0x5 0x6 0x7 0x8 0x9 0xa 0xb		0x0: Power on reset 0x1: Brown-out reset 0x2: Brown-out and power on reset 0x3: Watchdog reset 0x4: External reset occurred 0x5: Internal error occurred 0x6: IMC storage error occurred 0x7: ISCO storage error occurred 0x8: Reserved 0x9: BIB read error occurred 0xa: IMC heartbeat error 0xb: ISC0 heartbeat error	0x6F: Sensor-specific discrete
23 (0x17)	Boot Bank	OEM 0xD2	0x0			0x0: No events for this sensor	0x6F: Sensor-specific discrete
54 (0x36)	Active Boot	OEM 0xD2	0x0			0x0: No events for this sensor	0x6F: Sensor-specific discrete
135 (0x87)	SAM1411 FPGA Version	OEM 0xD6	0x0			0x0: No events for this sensor	0x6F: Sensor-specific discrete
136 (0x88)	FW Revision ISC	OEM 0xD6	0x0			0x0: No events for this sensor	0x6F: Sensor-specific discrete

Table 3-3 Sensor Data Records (continued)

Sensor Number	Sensor Name	Sensor Type	Event Data Byte 1	Event Data Byte 2	Event Data Byte 3	Event Threshold/ Description	Event Reading Type
133(0x85)	SAM IPMB0 State	IPMB 0xF1	[7:4] = Ah (OEM code in Event Data 2, OEM code in Event Data 3) [3:0] = Offset 00h – IPMB-A disabled, IPMB-B disabled 01h – IPMB-A enabled, IPMB-B disabled 02h – IPMB-A disabled, IPMB-B enabled 03h – IPMB-A enabled, IPMP-B enabled	[7:4] = Channel Number. For AdvancedTCA®, this will typically be 0h to indicate IPMB-0 [3:0] = Reserved	[7] – IPMB B Override state 0b = Override state, bus isolated 1b = Local Control state – IPM Controller determines state of bus. [6:4] = IPMB B Local Status 0h = No Failure. Bus enabled if no override in effect. 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 if it is causing a bus hang) 7h = Undiagnosed communications failure [3] – IPMB A Override status 0b = Override status, bus isolated 1b = Local Control state – IPM Controller determines state of bus. [2:0] = IPMB A Local status 0h = No failure. Bus enabled if no override in effect. 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 if it is causing a bus hang) 7h = Undiagnosed communications failure	0x3 = IPMB A enabled, IPMB-B enabled 0x2 = IPMB A disabled, IPMB-B enabled 0x1 = IPMB-A enabled, IPMB-B disabled 0x0 = IPMB A disabled, IPMB-B disabled	0x6F: Sensor-specific discrete

Table 3-3 Sensor Data Records (continued)

Sensor Number	Sensor Name	Sensor Type	Event Data Byte 1	Event Data Byte 2	Event Data Byte 3	Event Threshold/ Description	Event Reading Type
127(0x7F)	Ejector State	Button Switch 0x14	0x0 0x1 0x2 0x3 0x4			0x0 = Power button pressed 0x1 = Sleep button pressed 0x2 = Reset button pressed 0x3 = FRU latch open (Switch indicating FRU latch is in 'unlatched' position and FRU is mechanically removable) 0x4 = FRU service request button (1 = pressed,service, example: removal/replacement, requested)	0x6F: Sensor-specific discrete

Table 3-3 Sensor Data Records (continued)

Sensor Number	Sensor Name	Sensor Type	Event Data Byte 1	Event Data Byte 2	Event Data Byte 3	Event Threshold/ Description	Event Reading Type
134(0x86)	SAM1411 Watchdog	Watchdog 2 0x23	0x0 0x1 0x2 0x3 0x8	The Event Data 2 field for this command can be used to provide an event extension code, with the following definition: 7:4 interrupt type 0h = none 1h = SMI 2h = NMI 3h = Messaging Interrupt Fh = unspecified all other = reserved 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		0x0: Timer expired, status only (no action, no interrupt) 0x1: Hard Reset 0x2: Power Down 0x3: Power Cycle 0x8: Timer interrupt	0x6F: Sensor-specific discrete
52(0x34)	SYS FW PROGRESS	System Firmware Progress 0x0F	0x0 0x1 0x2	See IPMI Specifications.		0x0: System firmware error (POST Error) 0x1: System firmware hang 0x2: System firmware progress	0x6F: Sensor-specific discrete

Table 3-3 Sensor Data Records (continued)

Sensor Number	Sensor Name	Sensor Type	Event Data Byte 1	Event Data Byte 2	Event Data Byte 3	Event Threshold/ Description	Event Reading Type
53(0x35)	Boot Error	Boot Error 0x1E	0x0 0x1 0x2 0x3 0x4			0x0: No bootable media 0x1: Non-bootable diskette 0x2: PXE server not found 0x3: Invalid boot sector 0x4: Timeout waiting for user selection	0x6F: Sensor-specific discrete
126(0x7E)	OS Boot	OS Boot 0x1F	0x0 0x1 0x2 0x3 0x4 0x5 0x6			0x0: A: boot completed 0x1: C: boot completed 0x2: PXE boot completed 0x3: Diagnostic boot completed 0x4: CD_ROM boot completed 0x5: ROM boot completed 0x6: boot completed	0x6F: Sensor-specific discrete
65(0x41)	SAM1411 Over current	Current 0x03	0x0 0x1			0x0: Limit Not Exceeded 0x1: Limit Exceeded	0x5: Digital Discrete
56 (0x38)	System Boot	System Boot 0x1D	0x0 0x1 0x2 0x3 0x4			0x0: Initiated by power up 0x1: Initiated by hard reset 0x2: Initiated by warm reset 0x3: User requested PXE boot 0x4: Automatic boot to diagnostic	0x6F: Sensor-specific discrete
0 (0x0)	SAM1411 Temp	Temperature 0x01		Reading	Threshold		0x1: Threshold

Table 3-3 Sensor Data Records (continued)

Sensor Number	Sensor Name	Sensor Type	Event Data Byte 1	Event Data Byte 2	Event Data Byte 3	Event Threshold/ Description	Event Reading Type
132 (0x84)	SAM1411 Hotswap	Hotswap 0xF0	0x00 0x02 0x04 0x08 0x10 0x20 0x40 0x80	Event cause: 0x00: Normal state change 0x01: SHMC commanded change 0x02: Handle switch operator change 0x03: Fru programmatic action 0x04: Communication lost 0x05: Local failure 0x06: Extraction surprise change 0x07: Provided Info change 0x08: Invalid HW address 0x09: Unexpected deactivation 0x0A: Surprise state change	FRU ID	0x00: M0 0x02: M1 0x04: M2 0x08: M3 0x10: M4 0x20: M5 0x40; M6 0x80: M7	0x6F: Sensor-specific discrete
14(0x0E)	SAM1411 FEED A	Power Supply 0x08	0x0 0x1 0x2 0x3 0x4 0x5			0x0: Presence detected 0x1: Power supply failure detected 0x2: Predictive failure 0x3: Power supply AC lost 0x4: AC lost or out-of-range 0x5: AC out-of-range, but present	0x4: Digital discrete

Table 3-3 Sensor Data Records (continued)

Sensor Number	Sensor Name	Sensor Type	Event Data Byte 1	Event Data Byte 2	Event Data Byte 3	Event Threshold/ Description	Event Reading Type
15(0x0F)	SAM1411 FEED B	Power Supply 0x08	0x0 0x1 0x2 0x3 0x4 0x5			0x0: Presence detected 0x1: Power supply failure detected 0x2: Predictive failure 0x3: Power supply AC lost 0x4: AC lost or out-of-range 0x5: AC out-of-range, but present	0x4: Digital discrete
125(0x7D)	Shelf Alarm	Telco Alarm 0xF4					0x6F: Sensor-specific discrete

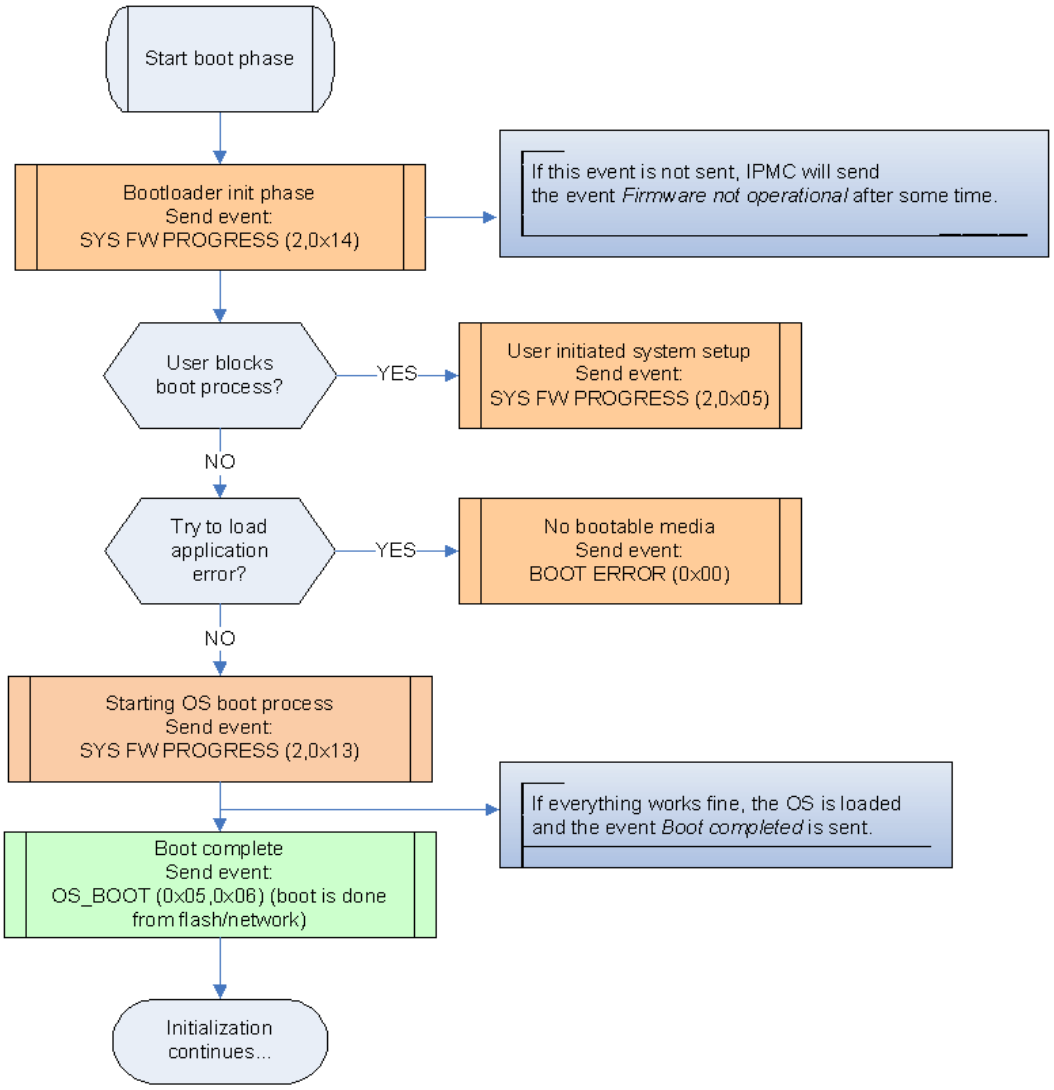
Table 3-3 Sensor Data Records (continued)

Sensor Number	Sensor Name	Sensor Type	Event Data Byte 1	Event Data Byte 2	Event Data Byte 3	Event Threshold/ Description	Event Reading Type
131(0x83)	ShelfFRU HotSwap	Hotswap 0xF0	0x00 0x02 0x04 0x08 0x10 0x20 0x40 0x80	Event cause: 0x00: Normal state change 0x01: SHMC commanded change 0x02: Handle switch operator change 0x03: Fru programmatic action 0x04: Communication lost 0x05: Local failure 0x06: Extraction surprise change 0x07: Provided Info change 0x08: Invalid HW address 0x09: Unexpected deactivation 0x0A: Surprise state change	FRU ID	0x00: M0 0x02: M1 0x04: M2 0x08: M3 0x10: M4 0x20: M5 0x40; M6 0x80: M7	0x6F: Sensor-specific discrete

3.4 Events during Boot Phase

During the boot phase of the product, the payload software sends events as described in the following figure. After the last event is sent, the OS is loaded and the product continues its initialization phase.

Figure 3-1 Boot Sequence



Orange indicates the bootloader/u-boot phase, green the OS phase, comments are blue.

Related Documentation

A.1 SMART Embedded Computing Documentation

The documentation listed is referenced in this manual. Technical documentation can be found by using the Documentation Search at *https://www.smartembedded.com/ec/support/* or you can obtain electronic copies of SMART EC documentation by contacting your local sales representative.

Table A-1 SMART EC Documents

Document Title	Document Number
SAM1411 Installation and Use	6806800M91
System Management Interface Based on HPI-B (Centellis CO 31kX /4100/2000/4410) User's Guide	6806800D84

A.2 Related Specifications

For additional information, refer to the following table for related specifications. As an additional help, a source for the listed document is provided. Please note that, while these sources have been verified, the information is subject to change without notice.

Table A-2 Related Specifications

Organization	Document Title
Intel developer.intel.com/design/servers/ipmi	Platform Management FRU Information Storage Definition v1.0 IPMI Specification v1.5
PICMG picmg.org/v2internal/specifications.htm	PICMG 3.0 Revision 3.0 Advanced TCA Base Specification