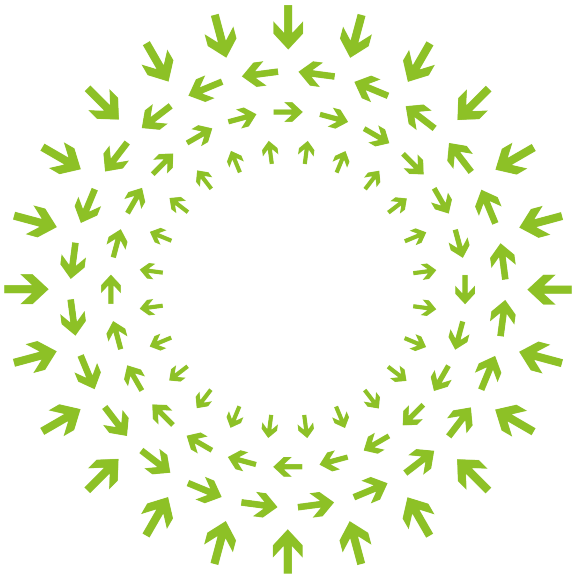

OPEN

Compute Project

Hardware Management
Type A IPM Controller

Version 0.03

Draft

Author: Hank Bruning, hank@jblade.com

1 Scope

This document defines the technical specifications for platform management of Open Compute Project servers, storage devices and network switches. The specification is limited to the data format and commands defined in Intelligent Platform Management Interface specification and does not require the presence of an operating system on the device that is managed.

Table of Contents

46 Revision History

Date	Revision	Description
December 31, 2013	0.2	Initial revision
January 3, 2014	0.3	Added XFP external calibration requirements.

2 Overview

This describes the Intelligent Platform Management Interface (IPMI) software interface implemented by a IPM Controller. It extends the IPMI 2.0 specification allowing Data Centers to implement a uniform System Management interface to servers, storage devices and network switches.

This specification does not contain any requirements for hardware dimensions, connectors or performance characteristics.

2.1 License

2.2 Reference Documents

These documents are referenced by this specification.

2.2.1 Specification Documents

Acronym	Date	Specification
DDR3 SPD	9/1/2011	JEDEC Standard No. 21-C, Annex K: Serial Presence Detect (SPD) for DDR3 SDRAM Modules, Release 4
DDR4 SPD	November 2013	JEDEC Standard No. 21-C Release 23, Annex L: Serial Presence Detect (SPD) for DDR4 SDRAM Modules

Acronym	Date	Specification
DDR Thermal Sensor	November 2009	JEDEC Standard No. 21-C Release 19, Section 4.1.5 TS3000 Standalone Thermal Sensor Component
IPMI 2.0	10/1/2013	Intelligent Platform Management Interface Specification Second Generation v2.0, Document Revision 1.1
SFF Diagnostic	1/30/2009	SFF-8472, Diagnostic Monitoring Interface for Optical Transceivers, Revision 10.4

2.3 Keywords

shall
A keyword indicating a mandatory requirement; designers are required to implement all such mandatory requirements to ensure interoperability with other products that conform to this specification.

shall not
A keyword used to describe a feature, function, or coded value that is defined in a specification to which this specification makes a normative reference where the use of said feature, function, or coded value is not allowed for implementations of this specification.

should
A keyword indicating flexibility of choice with a strongly preferred alternative.

vendor specific
Something (e.g., a bit, field, code value) that is not defined by this specification and may be used differently in various implementations.

3 IPM Controller Type A

This specification defines a set of requirements for an IPM Controller implementing the Intelligent Platform Management Interface (IPMI 2.0). They enable a Data Center System Manager to control rack mounted hardware with minimum operator intervention. The requirements automate interaction with the hardware to enable a single technician to operate 10,000 to 50,000 servers, storage devices or network switchs.

This specification provides uniform access to Servers, Storage Devices and Network Switchs which this document collectively calls IT hardware. No distinction is made between the different types of IT hardware, The benefits to the Data Center operator are a uniform, vendor and hardware neutral methods for:

- inventory data collection of model/serial numbers
- on-site customer acceptance tests
- operator training
- test procedures
- identification of DDR3 and DDR4 memory capacity and module errors
- diagnostic sensors and control for optical XFP interconnects

The requirements in this document are specific to IPM Controllers in IT hardware and no requirements are made on the System Manager itself. If a System Manager is not present, the IT Hardware will function normally.

This specification is not mandatory for IT Hardware. There is no architectural limitation preventing a single chassis from containing IPM Controllers implementing this specification and IPM Controllers implementing proprietary IPM Controllers.

3.1 Out of Scope

This specification does not contain any requirements for hardware dimensions, connectors or performance characteristics and does not mandate Hot Swap of IPMI Controllers.

3.2 Private Enterprise Number

The IPMI Commands and FRU records defined in this document utilize the Private Enterprise Number 42623 assigned to OCP by the Internet Assigned Number Authority , www.iana.org. In a twist of fate that only a IPMI implementer will enjoy, the number assigned to OCP ends in the IPMI UDP port number, 623.

3.3 FRU Records

All FRU records defined in this specification contain the Private Enterprise Number as the first three bytes after the record header checksum and is written Least Significant byte first.

3.3.1 OCP Version

The OCP Version Description Record identifies which version of this specification that

is implemented by a IPM Controller. It is used for both Chassis and Node IPM Controllers. The version number can be used to validate that all FRU records defined in this specification are present in the IPM Controllers FRU Information Area.

Table 1. OCP Version Description Record

Offset	Field Length	Field Name
0	1	*Record Type ID*. For all records defined in this specification a value of D0h (OEM) is used.
1	1	End of List/Version [7:7]- End of List. Set to one for the last record. [6:4]- Reserved. Write as 0h. [3:0]- Record Format Version. For this specification 0h.
2	1	Record Length
3	1	*Record Checksum*. The zero Checksum of the record.
4	1	*Header Checksum.* The zero Checksum of the header.
5	3	*Manufacturer ID*. The Private Enterprise number assigned to OCP. Write as 42623 (A67Fh) . Least significant byte first.
8	1	*OCP Record ID*. 00h
9	1	*Record Format Version*. [7:4] Reserved. Write as 0h. [3:0] Format Version ID. Use 0h to identify this table.
10	1	*Version number*. The version of the Type A controller implemented by the IPM Controller. To identify the version of this specification use 1h.
11	1	*LUN 0 Use Flags*. [7:1] Reserved. Write az zero. [0:0] L*UN 0 Use.* If set LUN 0 contains addressable FRU IDs and the field *LUN 0 MAX FRU ID* contains valid data. If not set LUN 0 does not contain addressable FRU IDs and field *LUN 0 MAX FRU ID* does not contain valid data.
12	1	*LUN 0 Max Fru ID*. The highest FRU ID addressable using LUN 0.
13	1	*LUN 1 Use Flags*. [7:1] Reserved. Write az zero. [0:0] L*UN 1 Use.* If set LUN 0 contains addressable FRU IDs and the field *LUN 1 MAX FRU ID* contains valid data. If not set LUN 1 does not contain addressable FRU IDs and field *LUN 1 MAX FRU ID* does not contain valid data.
14	1	*LUN 1 Max Fru ID*. The highest FRU ID addressable using LUN 1.
15	1	*LUN 2 Use Flags*. [7:1] Reserved. Write az zero. [0:0] L*UN 2 Use.* If set LUN 2 contains addressable FRU IDs and the field *LUN 2 MAX FRU ID* contains valid data. If not set LUN 2 does not contain addressable FRU IDs and field *LUN 2 MAX FRU ID* does not contain valid data.
16	1	*LUN 2 Max Fru ID*. The highest FRU ID addressable using LUN 2.
17	1	*LUN 3 Use Flags*. [7:1] Reserved. Write az zero.

Offset	Field Length	Field Name
		[0:0] *LUN 3 Use.* If set LUN 0 contains addressable FRU IDs and the field *LUN 3 MAX FRU ID* contains valid data. If not set LUN 0 does not contain addressable FRU IDs and field *LUN 3 MAX FRU ID* does not contain valid data.
18	1	*LUN 3 Max Fru ID*. The highest FRU ID addressable using LUN 3.

REQ-3.1 An IPM Controller **shall** include a single OCP Version Description record in the FRU Multi Record Area at address LUN 0, FRU ID 0.

3.4 IPMI Commands

The IPMI commands defined in this specification use the OEM/Group extension method documented in IPMI 2.0, Table 5-1, Network Function Codes.

REQ-3.2 IPMI Request Commands defined in this specification **shall** use the Network Function Code 2Eh.

REQ-3.3 IPMI Response Commands defined in this specification **shall** use the Network Function Code 2Fh.

4 Chassis Identification

Type A IPM Controllers contain FRU information that allows identifying the manufacture, model number, serial number and physical size of the Chassis.

4.1 Chassis Info Area Record

IPM Controllers that act as a Chassis Manager are required to have a Chassis Info record in the FRU Information. The Chassis Info Area Record contains the model and serial number of the chassis and is defined by the FRU Info Specification.

REQ-4.1 IPM Controllers representing a Chassis Manager **shall** populate Common Header field Chassis Info Area with a valid offset to the Chassis Info Area.

REQ-4.2 IPM Controllers not representing a Chassis Manager **shall** populate Common Header field Chassis Info Area with a 0h.

REQ-4.3 If more than one IPM Controller implements the Chassis Manager for a single chassis all IPM Controllers controlling the chassis **shall** contain an identical Chassis Info Area.

REQ-4.4 The length specified in the Chassis Part Number field of the Chassis Info Area Record record **shall** be greater than zero.

REQ-4.5 The length specified in the Chassis Serial Number field of the Chassis Info Area Record record **shall** be greater than zero.

REQ-4.6 If a IPM Controller representing the Chassis Manager is transferred from a chassis it **shall not** retain the Serial Number from the chassis it was removed from.

4.2 Chassis Description

The OCP Rack specification defines the minimum size of the hardware installed in a rack to be a 1/2 U high. To enable visualizing the physical layout of a rack without preexisting knowledge of the hardware within the rack Chassis Managers contain a static FRU Info record describing the size of chassis.

This document describes the dimensions of a chassis where the front and rear of the chassis have and identical dimension. No requirements are defined for a chassis with front that has a different dimension than the rear. A chassis compliant with this specification does not have to physically attach to both the front and the rear of the rack.

Table 2. Chassis Description Record

Offset	Field Length	Field Name
0	1	*Record Type ID*. For all records defined in this specification a value of D0h (OEM) is used.
1	1	End of List/Version [7:7]- End of List. Set to one for the last record. [6:4]- Reserved. Write as 0h.

Offset	Field Length	Field Name
		[3:0]- Record Format Version. For this specification 0h.
2	1	Record Length
3	1	*Record Checksum*. The zero Checksum of the record.
4	1	*Header Checksum.* The zero Checksum of the header.
5	3	*Manufacturer ID*. The Private Enterprise number assigned to OCP. Write as 42623 (A67Fh) . Least significant byte first.
8	1	*OCP Record ID*. 01h
9	1	*Record Format Version*. [7:4] Reserved. Write as 0h. [3:0] Format Version ID. Use 0h to identify this table.
10	1	*Chassis Height Major.* [7:6] Reserved [5:0] *Height Integer*. The number of OCP Rack Units completely filled by the chassis.
11	1	*Chassis Height Minor.* [7:1] Reserved. Write as 0h. [0:0] *Half Height Flag*. If set add 1/2 OCP Rack unit to the Chassis Height. If not set add nothing to the Chassis Height.

REQ-4.7 An IPM Controller representing a Chassis Manager **shall** have a Multi Record containing a Chassis Description Record at address LUN 0, FRU ID 0.

REQ-4.8 An IPM Controller that is not Chassis Manager **shall not** have a Multi Record containing a Chassis Description Record at address LUN 0, FRU ID 0.

156 5 IPM Controller Addressing

157 An IPM Controller Type A may have up to 255 IPM Controllers it can address using
158 multiple IPMB buses. Each IPMB bus is limited to 127 IPM Controllers.

159 The Type A IPM Controller IPMB Address Record(below) allows hardware vendors to
160 differentiate their products by distributing the subordinate IPM Controllers across one
161 or IPMI Channels, each implementing an IPMB bus. Cost sensitive implementations or
162 hardware with limited address range may have a single IPM Channel.

163 Hardware vendors creating systems with large number of IPM addresses may
164 differentiate their products by distributing the IPM Controllers over multiple channels
165 to reduce IPMB bus traffic or to minimize faults due to IPMB bus communication
166 errors.

167 5.1 IPMB Bus Address Record

168 All locations that a Type A IPM Controller may access an IPM Controller are defined in
169 the IPMB Address record.

170 **Table 3. IPMB Address Record**

Offset	Field Length	Field Name
0	1	*Record Type ID*. For all records defined in this specification a value of D0h (OEM) is used.
1	1	End of List/Version [7:7]- End of List. Set to one for the last record. [6:4]- Reserved. Write as 0h. [3:0]- Record Format Version. For this specification 0h.
2	1	Record Length
3	1	*Record Checksum*. The zero Checksum of the record.
4	1	*Header Checksum.* The zero Checksum of the header.
5	3	*Manufacturer ID*. The Private Enterprise number assigned to OCP. Write as 42623 (A67Fh) . Least significant byte first.
8	1	*OCP Record ID*. 02h.
9	1	*Record Format Version*. [7:4] Reserved. Write as 0h. [3:0] Format Version ID. Use 0h to identify this table.
10	1	*Channel Number*. [7:4] Reserved [3:0] Channel Number. The channel number of the IPM Controller containing the IPMB Bus.
11	1	*IPMB Address Count.* [7:7] Reserved. Write as 0h. [6:0] *Address Entry Count*. The number N of of IPMB addresses in this record.

Offset	Field Length	Field Name
12	1*N	*IPMB Address List*. A variable size list of IPMB Addresses. Each IPMB address is in its own byte.

REQ-5.1 An IPM Controller **shall** have one IPMB Address Record for each IPMI channel containing an IPMB Bus.

REQ-5.2 All IPMB Address Record **shall** be located at the IPMB address of the IPM Controller with and the address of LUN ID 0, FRU ID 0.

REQ-5.3 Each IPMB Address Record **shall** be contained in a single FRU Multi Record.

6 IPMI Implementation

To implement high performance System Managers that use the IPMI RMCP protocol a Type A IPM Controller must meet these requirements. The requirements in this section are part of the IPMI 2.0 specification but not explicitly state.

6.1 IPMI Command Address

IPMI 2.0 allows System Managers to send IPMI commands using a source IPMI address that contains LUN IDs 0,1,2 and 3. To enable multiple outstanding IPMI commands to concurrently execute the IPM controller must support all four LUNs as source addresses.

REQ-6.1 All IPMI requests sent to the IPM Controller using a source LUN ID zero **shall** have the response returned using a destination LUN ID zero.

REQ-6.2 All IPMI requests sent to the IPM Controller using a source LUN ID one **shall** have the response returned using a destination LUN ID one.

REQ-6.3 All IPMI requests sent to the IPM Controller using a source LUN ID two **shall** have the response returned using a destination LUN ID two.

REQ-6.4 All IPMI requests sent to the IPM Controller using a source LUN ID three **shall** have the response returned using a destination LUN ID three.

6.2 Fan Speed Sensor

IPMI 2.0 allows the Full Sensor SDR to contain a value of zero for the nominal and maximum sensor readings. When this is done for Fan Speed sensors the technician is unable to determine if a fan is operating normally or is at it's maximum speed.

To present a graph or table with valid information on the fan speed to a technician the nominal fan speed needs to be specified with non zero values. This allows verification that the fan is operating within the manufacturers nominal limits.

REQ-6.5 The Full Sensor SDR describing a fan speed **shall** have a value greater than zero for the fields Nominal Reading(byte 32), Normal Maximum(byte 33) and Normal Minimum(byte 34).

REQ-6.6 The Full Sensor SDR describing a fan speed **shall** have a value for the field Sensor Maximum Reading(byte 35) that is greater than or identical to the field Normal Maximum(byte 33).

Question: Should OCP mandate that at least one low and one high threshold be populated with a non-zero value?

6.3 Air Temperature Sensor

To standardize the reporting of temperature readings the Full Sensor SDR should not indicate the position in the air flow of a temperature sensor by encoding it in the sensor name. Instead use the IPMI 2.0 field Sensor Direction.

REQ-6.7 The Full Sensor SDR describing an intake air temperature sensor **shall** use a

value of 01h for the field Sensor Direction(byte 29) bits [1:0].

REQ-6.8 The Full Sensor SDR describing an exhaust air temperature sensor **shall** use a value of 10h for the field Sensor Direction(byte 29) bits [1:0].

REQ-6.9 The Full Sensor SDR describing an air temperature sensor that is neither an intake or exhaust sensor **shall** use a value of 00h for the field Sensor Direction(byte 29) bits [1:0].

REQ-6.10 The Full Sensor SDR describing an intake air temperature sensor **shall** use an Entity ID of TBD. Question: use 37h or 40h(DCMI 1.0) compatible ?

REQ-6.11 The Full Sensor SDR describing an exhaust air temperature sensor **shall** use an Entity ID of F3h. Question: This is an OEM value is there something better ?

7 DRAM

Type A IPM Controllers provide a view to the System Manager on the total number DDR3 or DDR4 slots supported by main CPUs and what type of memory modules, if any, are installed in the slots. This allows a System Manager to maintain a real time database of the type, memory capacity of each memory module and the total installed memory capacity available to each CPU.

This allows a System Manager to:

1. detect failed or incorrectly seated memory modules. The System Manager's database contains the expected number of modules in server and can detect if the actual number is lower.
2. detect which servers can be upgraded to higher memory capacity by populating unused slots or replacing low capacity modules with higher capacity modules.
3. compare memory module thermal design parameters to choose lowest cost cooling solutions. For example a Data Center may find it more cost effective to replace 85° C maximum operating temperate memory modules with 95° C modules than to redesign air conditioning systems.
4. select hardware based on actual memory capacity to install operating systems and the applications.
5. predict the number of memory modules which can be reallocated to other IT hardware when permanently deactivating running systems. For example when RAID is going to be removed from the Data Center how many DDR3 modules can be reallocated to compute servers.

This section is specific to DDR3 and DDR4 memory modules and does not provide requirements for IT hardware with embedded memory or memory soldered to a printed circuit board. The FRU Information Records (below) have been designed so DDR3/DDR4 memory modules that are soldered to a board can be coexist with removable memory that on the same circuit board. This allows a System Manager to determine the correct amount of memory when a single CPU uses low cost soldered DDR memory modules and the higher cost removable modules.

7.1 Slot Identification

Each Type A IPM Controller identifies the memory module slots associated with the main CPU(s) with a DRAM Address Record in the FRU Info.

Table 4. DRAM Address Record

Offset	Field Length	Field Name
0	1	*Record Type ID*. For all records defined in this specification a value of D0h (OEM) is used.
1	1	End of List/Version [7:7]- End of List. Set to one for the last record. [6:4]- Reserved. Write as 0h. [3:0]- Record Format Version. For this specification 0h.
2	1	Record Length

Offset	Field Length	Field Name
3	1	*Record Checksum*. The zero Checksum of the record.
4	1	*Header Checksum.* The zero Checksum of the header.
5	3	*Manufacturer ID*. The Private Enterprise number assigned to OCP. Write as 42623 (A67Fh) . Least significant byte first.
8	1	*OCP Record ID*. 03h
9	1	*Record Format Version*. [7:4] Reserved. Write as 0h. [3:0] Format Version ID. Use 0h to identify this table.
10	1	*DRAM Slot Type*.The values are identical to the DDR4 SPD, field Key Byte / DRAM Device Type, byte 2. 0h Reserved 1h Standard FPM DRAM 2h EDO 3h Pipelined Nibble 4h SDRAM 5h ROM 6h DDR SGRAM 7h DDR SDRAM 8h DDR2 SDRAM 9h DDR2 SDRAM FB-DIMM Ah DDR2 SDRAM FB-DIMM PROBE Bh DDR3 SDRAM Ch DDR4 SDRAM All other values are reserved.
11	1	*DRAM Slot Count*. [7:5] Reserved. Write as 0h. [4:0] Slot Address Entry Count. The number N, of DRAM Addresses in this record.
13	4*N	*Slot Address List*. A variable length list containing Slot Address Count entries. Each entry is seven bytes formatted according to Table 5. DRAM Slot Descriptor.

Each memory module slot available to the main CPU(s) is identified with a DRAM Slot Descriptor. It identifies the FRU address that represents the memory slot. A field *Circuit Board ID* contains the ASCII characters that uniquely identify the slot either in manufacturer documentation or the characters are printed on the circuit board adjacent to the memory slot. The *Circuit Board ID* may contain a maximum of two characters. This allows the System Manager to create error messages that are specific to a DRAM Slot.

Table 5. DRAM Slot Descriptor

Offset	Field Length	Field Name
1	1	*IPMB Address*. The IPMB Address of the DRAM slot.
2	1	*Internal Address*. [7:4] *Channel number*. The Channel number of the DRAM slot. This

Offset	Field Length	Field Name
		is identical to the value found in the FRU Device Locator Record field *Channel Number*. [3:2] Reserved. Write as zero. [1:0] *LUN ID*. The LUN ID within the IPMB address where the FRU is addressed. Identical to the value found in the FRU Device Locator Record field *Access LUN.*
3	1	*FRU ID*. The FRU ID of the DRAM Slot.
4	1	*Slot Description*. [7:1] Reserved. Write as 0h. [0:0] *Removable Memory Flag*. 0h the memory is embedded or soldered to the PCB and can not be removed. 1h the memory in the slot can be removed.

REQ-7.1 For each memory module slot present on the device being managed the IPM Controller's Device SDR located at LUN 0 **shall** contain an IPMI FRU Device Locator Record with an entity ID 8h.

REQ-7.2 Each IPMI FRU Device Locator Record for a memory module slot **shall** have the field *Device String* set to a value that uniquely identifies the slot on the Printed Circuit Board. **ToDo**: Has the OCP other board sub committee standardized on what is printed on the PCB next to a memory slot ?

REQ-7.3 The IPM Controller's FRU Info located at LUN 0, FRU ID 0 **shall** contain a single DRAM Address Record.

REQ-7.4 Each DRAM Slot Descriptor **shall** have it's Channel Number and IPMB Address field present in the IPM Controller Address Record.

REQ-7.5 Each memory module slot, populated by a DDR module or un-populated **shall** have a single DRAM Slot Descriptor.

REQ-7.6 The contents of the DRAM Address Record **shall not** change once a RMCP session has been activated.

7.2 IPM Controller to Module Interface

This specification has no requirements on the interface between the IPM Controller and the memory module. It may be proprietary. One method is an I^2C interface between the IPM Controller and a memory module slot. The I^2C interface is defined by the JDEC C-21 DDR Thermal Sensor specification. It allows up to eight DDR memory modules per I^2C bus.

Using the I^2C interface to each memory Slot a Type A IPM Controller can detect if a DDR3 or DDR4 memory module is present in the slot. Configuration information and operating status can be read from the memory module.

7.3 Module Identification

An IPM Controller maps each memory module's Serial Presence Detect(SPD) data bytes to a different FRU ID's FRU Information area as a OCP defined SDR Multi Record. An IPM Controller is not required to read the SPD data until it received an IPMI Read FRU Info command for the FRU ID. Once the IPM Controller or the System Manager

has read the SPD data no changes are allowed to that data until the IPM Controller is rebooted.

The format of the SPD data is not compatible between the DDR3 (DDR3 SPD Appendix K) and DDR4 (DDR4 SPD Appendix L). They share a common format for Byte 2 of the SPD data that differentiates between the types of memory. Note that the DDR4, Appendix L is only present in the JEDEC Standard No. 21-C Release 23 and later versions.

The the SPD data contain a Cyclical Redundancy Code(CRC) . The IPM Controller is not required to validate that the CRC is correct.

7.3.1 DDR3 SPD

The DDR3 SPD specification documents a sequence 128 bytes on each DDR3 module that identify the size, capabilities, vendor and serial number.

The thermal characteristics of the memory module are defined by JDEC 21-C as:

- Normal temperature range, 0 to 85 degrees C
- Extended temperature range, 0 to 95 degrees C.

This information is present in the SPD as field *SDRAM Thermal and Refresh Options*(byte 31).

7.3.2 DDR4 SPD

The DDR4 SPD specification documents a sequence 512 bytes on each DDR4 module that identify the size, capabilities, vendor and serial number.

The thermal characteristics of the memory module are defined by the hardware vendor an present in the SPD as field *SDRAM Thermal and Refresh Options*(byte 8).

7.3.3 DRAM Multi Record Description

An IPM Controller maps each memory module's Serial Presence Detect data to a single FRU's Multi Record area containing a DRAM Module Description Record.

Table 6. DRAM Module Description Record

Offset	Field Length	Field Name
0	1	*Record Type ID*. For all records defined in this specification a value of D0h (OEM) is used.
1	1	End of List/Version [7:7]- End of List. Set to one for the last record. [6:4]- Reserved. Write as 0h. [3:0]- Record Format Version. For this specification 0h.
2	1	Record Length
3	1	*Record Checksum*. The zero Checksum of the record.
4	1	*Header Checksum.* The zero Checksum of the header.
5	3	*Manufacturer ID*. The Private Enterprise number assigned to OCP. Write as 42623 (A67Fh) . Least significant byte first.
8	1	*OCP Record ID*. 04h
9	1	*Record Format Version*. [7:4] Reserved. Write as 0h.

Offset	Field Length	Field Name
		[3:0] Format Version ID. Use 0h to identify this table.
10	9+N	*DRAM Module Description*. The memory module Serial Presence Detect data formatted according to DDR3 SPD or DDR4 SPD specification. The size, N, is determined from the SPD data byte offset zero, bits [3:0].

REQ-7.7 If a memory module is detected in a memory module slot the IPM Controller **shall** set the FRU ID's Common Header field *MultiRecord Area Starting Offset* to the starting offset of the FRU Multi record containing the DRAM Module Description

7.4 Temperature Sensor

The DDR3 and DDR4 SPD data contains a field indicating the presence of a temperature sensor on the memory module. The sensor is optional but when present this section documents how Type A IPM Controller interacts with the sensor. When present, the sensor allows each DDR memory module to provide temperature readings that provide information on the health and status of the hardware. If a server is implemented without Fan speed sensors the rise in DDR temperatures is often the first indication that Fan is in process of, or has failed.

The DDR Thermal Sensor specification defines the sensor thresholds and how the thermal sensor is read.

REQ-7.8 If the IPM Controller detects that a memory module implements a temperature sensor the IPMI Controller **shall** create a Full Sensor Record with the upper non-critical threshold from the data in DDR Thermal Sensor, Section 6.5.1 Alarm Temperature Upper Boundary Register.

REQ-7.9 If the IPM Controller detects that a memory module implements a temperature sensor the IPMI Controller **shall** create a Full Sensor Record with the lower non-critical threshold from the data in DDR Thermal Sensor, Section 6.5.2 Alarm Temperature Lower Boundary Register.

REQ-7.10 If the IPM Controller detects that a memory module implements a temperature sensor the IPMI Controller **shall** create a Full Sensor Record with the upper critical threshold from the data in DDR Thermal Sensor, Section 6.5.3 Critical Temperature Register.

REQ-7.11 The entity ID in the Full sensor Record of a memory module **shall** be 8h (memory module).

8 XFP Modules

The 10 Gigabit Small Form Factor Pluggable (XFP) module optical interface allows IT hardware to communicate using Ethernet, Fibre Channel, InfiniBand and SONET/SDH at rates up to 10Gig/sec. The SFF Diagnostic specification mandates a 2 wire serial interface that a IPM Controller uses to provide identification of the type XFP, maximum transceiver separation, temperature sensors and diagnostics.

When two pieces of IT hardware are connected using a XFP interface they may be in the same rack or at a maximum distance of 80 Kilometers. The XFP standard allows the laser transmitters to be placed in diagnostic mode and to preform bidirectional end-to-end testing with the receiver reporting DB loss.

This specification creates an interface allowing the System Manager a vendor neutral end-to-end testing independent of the protocol used by the XFP module. This allows detection of faults in fiber optic cables without a technician's physical interaction with the cables. The cable interconnect can be tested in place without the a person touching the IT hardware and detect cables broken by being pinched in rack doors or broken due to cabling changes between racks or buildings.

The transmit output power of a XFP modules varies based on input voltage to the XFP and the temperature of XFP module. The XFP modules has sensors to monitor both and the Type A controller standardizes access to the sensor readings and the alarm thresholds.

A System Manager operating a Type A IPM Controller at both ends of a fiber optic link can detect mismatches in optical transceivers. This can be done prior to physical connecting fiber optic cable by examining the wavelength and output power of the laser.

A Type A IPM Controller provides a sensor to measure received optical power in 0.1 uWatt increments. Differences in fiber optical cable length, fiber diameter and the temperature of the transmitter cause variations in received optical power. Each XFP module provides alarm thresholds which the IPM Controller uses to to create an IPMI Full Sensor Record.

8.1 IPM Controller to XFP Interface

This specification has no requirements on the interface between the IPM Controller and a XFP. It may be proprietary. One method is a derivative of the I^2C interface with additional packet error checking and is defined by the Error: Reference source not found specification. **Question**: Is calling this an I^2C interface accurate ? I need some help here !

The SFF Diagnostic specification defines two blocks of 128 bytes.

The block at address A0h, in this document is called the Identity Block, is read only and contains information that identify the optical compatibility,

maximum fiber length, signaling rate and vendor identification of the XFP.

The second block at address A2h, in this document called the Diagnostic Block, is writable by both the XFP and the IPM Controller. It contains the thermal/voltage thresholds, current sensor readings, measured transmit/receive laser power and diagnostic status. This block contains a byte, offset 110, that disables/enables the laser output. This function is an optional feature of the XFP and if present allows the IPM Controller to test the connectivity with remote optical transceiver.

Question: Is there any vendor independent way to set/get the wavelength of the XFP? The ITU-T defines the wavelength and frequency but that usually involves a vendor defined ID written to a I2C address.

8.2 Bay Identification

Each Type A IPM Controller identifies the XFP Bays with a XFP Address Record in the FRU Info.

Table 7. XFP Address Record

Offset	Field Length	Field Name
0	1	*Record Type ID*. For all records defined in this specification a value of D0h (OEM) is used.
1	1	End of List/Version [7:7]- End of List. Set to one for the last record. [6:4]- Reserved. Write as 0h. [3:0]- Record Format Version. For this specification 0h.
2	1	Record Length
3	1	*Record Checksum*. The zero Checksum of the record.
4	1	*Header Checksum.* The zero Checksum of the header.
5	3	*Manufacturer ID*. The Private Enterprise number assigned to OCP. Write as 42623 (A67Fh) . Least significant byte first.
8	1	*OCP Record ID*. 05h
9	1	*Record Format Version*. [7:4] Reserved. Write as 0h. [3:0] Format Version ID. Use 0h to identify this table.
10	1	*XFP Bay Count*. [7:6] Reserved. Write as 0h. [5:0] *Bay Address Entry Count*. The number N, of XFP Addresses in this record.
11	6*N	*XFP Address Lis*t. A variable length list containing Bay Address Count entries. Each entry is seven bytes formatted according to Table 8. XFP Bay Descriptor.

Each XFP Bay is identified with a XFP Bay Descriptor. It identifies the FRU address that represents the XFP Bay. A field Face Plate *ID* contains the ASCII characters that uniquely identify the bay either in manufacturer documentation or the characters are printed on the face pate adjacent to the Bay. This allows the System Manager to create error messages that are specific to a XFP Bay.

Each XFP Bay is identified with a physical location in a grid so System Managers can draw the face plate and visually indicate the bay.

Table 8. XFP Bay Descriptor

Offset	Field Length	Field Name
0	1	*IPMB Address* . The IPMB Address of the XFP Bay. Derived from the value found in the FRU Device Locator Record field *Device Access Address* .
1	1	*Internal Address*. [7:4] *Channel number*. The Channel number of the XFP Bay. Identical to the value found in the FRU Device Locator Record field *Channel Number*. [3:2] Reserved. Write as zero. [1:0] *LUN ID*. The LUN ID within the IPMB address where the FRU is addressed. This is identical to the value found in the FRU Device Locator Record field *Access LUN.*
2	1	*FRU ID*. The FRU ID of the XFP Bay. This is identical to the value found in the FRU Device Locator Record field FRU Device ID.
3	1	*XFP Bay Flag*. [7:3] Reserved. Write as zero. [2:0] *XFP Bay location*. 0h Rear of chassis 1h front of chassis 2h top of module 3h bottom of module 4h internal interface.
4	1	*Row Count*. The row number where the XFP module is located. Row Zero is the top most row in the chassis. **ToDo**: Identify row when using internal modules.
5	1	*Column Count*. The column within the row where the XFP Bay is located. Column zero is the left most column.

REQ-8.1 For each XFP Bay present on the device being managed the IPM Controller's Device SDR located at LUN 0 **shall** contain an IPMI FRU Device Locator Record with an entity ID E2h.

REQ-8.2 Each IPMI FRU Device Locator Record for a XFP Bay **shall** have the field *Device String* set to a value that uniquely identifies the XFP Bay.

REQ-8.3 If an XFP Bay is present on the device being managed the IPM Controller's FRU Info located at LUN 0, FRU ID 0 **shall** contain one or more XFP Address Records.

REQ-8.4 Each XFP Bay Descriptor **shall** have a single IPMI FRU Device locator record with a matching IPMB Address, Channel Number and FRU ID.

REQ-8.5 Each XFP Bay, populated by an XFP or un-populated **shall** have a single XFP Bay Descriptor.

REQ-8.6 The contents of the XFP Address Record **shall not** change once a RMCP session has been activated.

REQ-8.7 If the XFP Bay Descriptor field *XFP Bay location* is Rear then the descriptor **shall**

have a unique Row Count/Column Count combination for all XFP Bays on the rear face plate.

REQ-8.8 If the XFP Bay Descriptor field *XFP Bay location* is Front then the descriptor **shall** have a unique Row Count/Column Count combination for all XFP Bays on the front face plate.

8.3 Module Identification

An IPM Controller maps the XFP's Identity Blocks data bytes to a different FRU ID's FRU Information area as a OCP defined SDR Multi Record. An IPM Controller is not required to read the Identify data until it received an IPMI Read FRU Info command for the FRU ID. Once the IPM Controller or the System Manager has read the Identity Block data no changes are allowed to that data until the IPM Controller is rebooted.

The Diagnostic block data is only accessible by the System Manager using IPMI read/write commands.

Table 9. XFP Module Description Record

Offset	Field Length	Field Name
0	1	*Record Type ID*. For all records defined in this specification a value of D0h (OEM) is used.
1	1	End of List/Version [7:7]- End of List. Set to one for the last record. [6:4]- Reserved. Write as 0h. [3:0]- Record Format Version. For this specification 0h.
2	1	Record Length
3	1	*Record Checksum*. The zero Checksum of the record.
4	1	*Header Checksum.* The zero Checksum of the header.
5	3	*Manufacturer ID*. The Private Enterprise number assigned to OCP. Write as 42623 (A67Fh) . Least significant byte first.
8	1	*OCP Record ID*. 06h
9	1	*Record Format Version*. [7:4] Reserved. Write as 0h. [3:0] Format Version ID. Use 0h to identify this table.
10	128	*XFP Module Description*. The block of 128 byes starting at the XFP address A0h.

8.4 Sensors

The XFP Identity block contains a flag(byte 92, bit 6) that when set indicates the XFP implements a group of five diagnostic sensors. The XFP module's Diagnostic block contains sensor thresholds and readings for optical input power, input bais current, temperature, laser transmit and optical receive power. The IPM Controller reads the Diagnostic Block and creates a IPMI Full Sensor Record for each of the five sensors.

XFP Module measures optical receive power. It ranges from 0 to 6.5 milli Watt in 0.1 uWatt increments and varies by wavelength, temperature and input power to the transmitter.

XFP Modules include threshold values for the receive power level and Data Centers may use sensor readings to determine which IT hardware would most likely be affected when a room temperate is raised. The optical link can be monitored in realtime and if the link margin falls below Data Center defined thresholds a technician can investigate and rectify the degradation before it affects traffic on the optical link.

Todo: Add Compact sensor for laser receive fault. What to do about 1.0 vs 1.5 watt optical output ?

Question: What to do in the case the module is external calibrated. The calibration values for the sensors are in the Diagnostic block(bytes 56-91).

Question: Should the Device SDR have a fixed number of Full Sensor Records (five for each XFP) and use the sensor scanning bit to enable/disable reading of them? This spec is currently written to remove the Full SDRs.

REQ-8.9 For each XFP Module detected the IPM Controller **shall** validate the Diagnostic Block CRC value at offset 95 is valid for bytes 0 through 94 before creating the XFP Module's Full Sensor SDR Records.

8.4.1 Module Presence sensor

Each XFP bat has a IPMI sensor to detect when the an XFP is present,

REQ-8.10 For each XFP Bay the IPM Controller **shall** have in the Device SDR a Compact Sensor Record with the Event/Reading Type code of 8h, Device Absent/Device Present.

8.4.2 Module Temperature

Each XFP Module has an internal temperature sensor. This document does not define how often the IPM Controller reads the XFP Module Temperature.

REQ-8.11 If bit 6, address 92 of the Identity Block is set the IPM Controller **shall** create a Full Sensor Record for the XFP module containing the module temperature with an entity ID of E3h.

REQ-8.12 The XFP Module temperature Full Sensor SDR **shall** contain an upper critical threshold created from the Diagnostic Block byte offsets 0 and 1.

REQ-8.13 The XFP Module temperature Full Sensor SDR **shall** contain lower critical threshold created from the Diagnostic Block byte offsets 2 and 3.

REQ-8.14 The XFP Module temperature Full Sensor SDR **shall** contain an upper non-critical threshold created from the Diagnostic Block byte offsets 4 and 5.

REQ-8.15 The XFP Module temperature Full Sensor SDR **shall** contain an upper non-critical threshold created from the Diagnostic Block byte offsets 6 and 7.

REQ-8.16 The IPM Controller **shall read** the Diagnostic Block byte offsets 96 and 97 as when determining the XFP Module temperature.

REQ-8.17 If bit 4, address 92 of the Identity Block is set the IPM Controller **shall** convert XFP Module temperature sensor data to IPMI sensor readings using the values in the Identity Block fields bytes 84 through 87.

REQ-8.18 If bit 4, address 92 of the Identity Block is set the XFP Module temperature

Full Sensor SDR threshold values **shall** contain values calculated using the Identity Block field's bytes 84 through 87.

8.4.3 Module Input Voltage

Each XFP Module has an input voltage sensor. This document does not define how often the IPM Controller reads the XFP Module input voltage.

REQ-8.19 If bit 6, address 92 of the Identity Block is set the IPM Controller **shall** create a Full Sensor Record for the XFP module containing the module input voltage with an entity ID of E3h.

REQ-8.20 The XFP Module voltage Full Sensor SDR **shall** contain an upper critical threshold created from the Diagnostic Block byte offsets 8 and 9.

REQ-8.21 The XFP Module input voltage Full Sensor SDR **shall** contain lower critical threshold created from the Diagnostic Block byte offsets 10 and 11.

REQ-8.22 The XFP Module input voltage Full Sensor SDR **shall** contain an upper non-critical threshold created from the Diagnostic Block byte offsets 12 and 13.

REQ-8.23 The XFP Module input voltage Full Sensor SDR **shall** contain an upper non-critical threshold created from the Diagnostic Block byte offsets 14 and 15.

REQ-8.24 The IPM Controller **shall read** the Diagnostic Block byte offsets 98 and 99 as when determining the XFP Module input voltage.

REQ-8.25 The XFP Module input voltage Full Sensor SDR **shall** have the field Sensor Direction(byte 24) set to 01h(input sensor).

REQ-8.26 If bit 4, address 92 of the Identity Block is set the IPM Controller **shall** convert XFP Module input voltage sensor data to IPMI sensor readings using the values in the Identity Block fields bytes 88 through 91.

REQ-8.27 If bit 4, address 92 of the Identity Block is set the XFP Module input voltage Full Sensor SDR threshold values **shall** contain values calculated using the Identity Block field's bytes 88 through 91.

8.4.4 Optical Transmit Power

The optical transmit power can vary from 0 to 6.5 milli Watt. Each XFP Module has a optical transmit power sensor. This document does not define how often the IPM Controller reads the XFP Module optical transmit power sensor.

REQ-8.28 If bit 6, address 92 of the Identity Block is set the IPM Controller **shall** create a Full Sensor Record for the XFP module containing the module optical transmit power with an entity ID of E3h.

REQ-8.29 The XFP Module optical transmit power Full Sensor SDR **shall** contain an upper critical threshold created from the Diagnostic Block byte offsets 24 and 25.

REQ-8.30 The XFP Module optical transmit power Full Sensor SDR **shall** contain lower critical threshold created from the Diagnostic Block byte offsets 26 and 27.

REQ-8.31 The XFP Module optical transmit power Full Sensor SDR **shall** contain an upper non-critical threshold created from the Diagnostic Block byte offsets 28 and 29.

REQ-8.32 The XFP Module optical transmit power Full Sensor SDR **shall** contain an upper

non-critical threshold created from the Diagnostic Block byte offsets 30 and 31.

REQ-8.33 The IPM Controller **shall read** the Diagnostic Block byte offsets 98 and 99 as when determining the XFP Module optical transmit power.

REQ-8.34 The XFP Module optical transmit power Full Sensor SDR **shall** have the field Sensor Direction(byte 24) set to 10h(output sensor).

REQ-8.35 If bit 4, address 92 of the Identity Block is set the IPM Controller **shall** convert XFP Module laser transmit sensor data to IPMI sensor readings using the values in the Identity Block fields bytes 80 through 83.

REQ-8.36 If bit 4, address 92 of the Identity Block is set the XFP Module laser transmit Full Sensor SDR threshold values **shall** contain values calculated using the Identity Block field's bytes 80 through 83.

8.4.5 Optical Receive Power

Each XFP Module has a optical receive power sensor. This document does not define how often the IPM Controller reads the XFP Module optical receive power sensor.

REQ-8.37 If bit 6, address 92 of the Identity Block is set the IPM Controller **shall** create a Full Sensor Record for the XFP module containing the module optical receive power with an entity ID of E3h.

REQ-8.38 The XFP Module optical receive power Full Sensor SDR **shall** contain an upper critical threshold created from the Diagnostic Block byte offsets 32 and 33.

REQ-8.39 The XFP Module optical receive power Full Sensor SDR **shall** contain lower critical threshold created from the Diagnostic Block byte offsets 34 and 35.

REQ-8.40 The XFP Module optical receive power Full Sensor SDR **shall** contain an upper non-critical threshold created from the Diagnostic Block byte offsets 36 and 37.

REQ-8.41 The XFP Module optical receive power Full Sensor SDR **shall** contain an upper non-critical threshold created from the Diagnostic Block byte offsets 38 and 39.

REQ-8.42 The IPM Controller **shall** read the Diagnostic Block byte offsets 104 and 105 as when determining the XFP Module optical receive power.

REQ-8.43 The XFP Module optical receive power Full Sensor SDR **shall** have the field Sensor Direction(byte 24) set to 01h(input sensor).

REQ-8.44 If bit 4, address 92 of the Identity Block is set the IPM Controller **shall** convert XFP Module optical receive power sensor data to IPMI sensor readings using the values in the Identity Block fields bytes 56 through 75.

REQ-8.45 If bit 4, address 92 of the Identity Block is set the XFP Module optical receive power Full Sensor SDR threshold values **shall** contain values calculated using the Identity Block field's bytes 56 through 75.

8.4.6 Laser Bias

Each XFP Module has a transmission bias current sensor. This document does not define how often the IPM Controller reads the XFP Module transmission bias current sensor.

REQ-8.46 If bit 6, address 92 of the Identity Block is set the IPM Controller **shall** create a Full Sensor Record for the XFP module containing the module laser bias with an entity ID of E3h.

REQ-8.47 The XFP Module transmission bias current Full Sensor SDR **shall** contain an upper critical threshold created from the Diagnostic Block byte offsets 16 and 17.

REQ-8.48 The XFP Module transmission bias current Full Sensor SDR **shall** contain lower critical threshold created from the Diagnostic Block byte offsets 18 and 19.

REQ-8.49 The XFP Module transmission bias current Full Sensor SDR **shall** contain an upper non-critical threshold created from the Diagnostic Block byte offsets 20 and 21.

REQ-8.50 The XFP Module transmission bias current Full Sensor SDR **shall** contain an upper non-critical threshold created from the Diagnostic Block byte offsets 22 and 23.

REQ-8.51 The IPM Controller **shall** read the Diagnostic Block byte offsets 100 and 101 as when determining the XFP Module transmission bias current.

REQ-8.52 The XFP Module transmission bias current Full Sensor SDR **shall** have the field Sensor Direction(byte 24) set to 10h(output sensor).

REQ-8.53 The XFP Module transmission bias current Full Sensor SDR **shall** have the measure units in milliamperes.

REQ-8.54 If bit 4, address 92 of the Identity Block is set the IPM Controller **shall** convert XFP Module transmission bias current sensor data to IPMI sensor readings using the values in the Identity Block fields bytes 76 through 79.

REQ-8.55 If bit 4, address 92 of the Identity Block is set the XFP Module transmission bias current Full Sensor SDR threshold values **shall** contain values calculated using the Identity Block field's bytes 76 through 79.

399 8.5 Hot Plug

400 The XFP specification allows modules to be inserted and removed while the IT
401 hardware is running. This capability is called Hot Plug.

REQ-8.56 The IPM Controller **shall** detect the insertion of a XFP Module into a bay and complete the addition of all the XFP module's Compact and Full sensor data record into the device SDR within ten seconds.

REQ-8.57 The IPM Controller **shall** detect the removal of a XFP Module into a bay and remove the XFP's Compact and Full sensor data record from the device SDR within ten seconds.

402 8.6 Laser Control

403 If a Type A IPM Controller is at both ends of the fiber optic cable it can validate that
404 the cable is connected to the correct port by toggling the laser state and reading the
405 optical receive power. Type A IPM Controllers create an IPMI Compact Sensor record
406 for both the transmit and receive state of each XFP modules laser.

407 The IPM Controller may enable/disable the laser state using a proprietary method or
408 use the Diagnostic block's byte(at offset 110). The SFF Diagnostic specification
409 identifies that functions in this byte is optional and not all XFP modules implement
410 software control of the laser.

8.6.1 Laser Transmit Command

The IPM Controller enables or disables the optical transmitter when it receives an Set XFP Laser Transmit command.

Table 10: Set Laser Transmit State Command

	Byte	Data Field
Request Data	0:2	*OCP Private Enterprise Number*. The Private Enterprise number assigned to OCP. The value 42623 (A67Fh) . Least significant byte first.
	3	Laser Output Flags. [7:1] Reserved. Write as zero. [0:0] Laser Transmit State. If set the laser output will be enabled. If not set the laser output will be disabled.
Response Data	0	*Completion Code*.

9 Reset Firmware

Question: This may not be the best way to implement the reset command. Currently it is written so that vendors can have unique defaults but that causes the OCP Compliance & Interoperability subcommittee to have unique test cases for each vendor. Should this command be rewritten so it is not vendor dependent ?

Type A IPM Controllers accept an IPMI command that resets the IPMI firmware to a known state defined by each hardware vendor. On receipt of the reset command all RMCP accounts will be deleted on all of the IPM Controller's channels. Then for each channel a single administrator account is created utilizing User ID 1. The password used for the account is defined by the hardware vendor.

The reset command sets the BMC key, Serial Mux setting and all LAN Parameters(IP Address, VLAN ID, etc) to values defined by the hardware vendor.

9.1 Command Format

The IPMI command to reset the IPMI Firmware state is

Table 11: Reset Firmware Command

	Byte	Data Field
Request Data	0:2	*OCP Private Enterprise Number*. The Private Enterprise number assigned to OCP. The value 42623 (A67Fh) . Least significant byte first.
Response Data	0	*Completion Code*.

REQ-9.1 If more than one RMCP session is active on any channel of the IPM Controller then the IPM Controller **shall** return a completion code of D5h(Command not supported in present state).

REQ-9.2 On receipt of the Reset Firmware command all RMCP account names on all channels **shall** be removed then a single RMCP account name of “admin” shall be added as User ID 1.

REQ-9.3 The password used when creating the “admin” account **shall** be reset to the manufacturer default.

REQ-9.4 While the Reset Firmware Request command is executing all requests to activate a RMCP session **shall** be refused using a 1h(Insufficient resources to create a session) in the RAKP 2 or RAKP 4 field *RMCP+ status code*(byte 2).

REQ-9.5 On receipt of the Reset Firmware command all IPMI LAN Parameters **shall** be reset to manufacturer defaults.

REQ-9.6 After the Reset Firmware response is sent with a completion code of normal the number of entries in the System Event Log **shall** be set to zero.

REQ-9.7 After a Reset Firmware Command request is received any subsequent Reset Firmware requests **shall** be reject until a Reset Firmware response is sent.

REQ-9.8 Five seconds after a Reset Firmware response command with a completion code

of normal is sent the controller **shall** terminate all RMCP sessions.

433 10 ID Assignment

434 **Table 12: IPMI Command ID Assignments**

Command Name	Table Number	Command ID	Minimum Privilege Level
Set Laser Transmit State Command	10	0h	Administrator
Reset Firmware Command	11	1h	Administrator

435 **Table 13: FRU Information Record ID Assignments**

FRU Record Name	Table Number	Record ID	Chassis Manager	Node Manager
OCP Version Description Record	1	0h	Yes	Yes
Chassis Description Record	2	1h	Yes	No
IPMB Address Record	3	2h	Yes	Yes
DRAM Address Record	4	3h	Yes	Yes
DRAM Module Description Record	6	4h	Yes	Yes
XFP Address Record	7	5h	Yes	Yes
XFP Module Description Record	9	6h	Yes	Yes

436 **Table 14: Entity ID Assignments**

Entity ID	Entity Name
E2h	XFP Bay
E3h	XFP Module