

Evaluation Copy



Compute Express Link™ (CXL™)

Engineering Change Notice to the Specification

December 2021

Type 3 Management Using MCTP CCI

LEGAL NOTICE FOR THIS PUBLICLY-AVAILABLE SPECIFICATION FROM COMPUTE EXPRESS LINK CONSORTIUM, INC.

© 2019-2021 COMPUTE EXPRESS LINK CONSORTIUM, INC. ALL RIGHTS RESERVED.

This CXL Specification (this “**CXL Specification**” or this “**document**”) is owned by and is proprietary to Compute Express Link Consortium, Inc., a Delaware nonprofit corporation (sometimes referred to as “**CXL**” or the “**CXL Consortium**” or the “**Company**”) and/or its successors and assigns.

NOTICE TO USERS WHO ARE MEMBERS OF THE CXL CONSORTIUM:

If you are a Member of the CXL Consortium (sometimes referred to as a “**CXL Member**”), and even if you have received this publicly-available version of this CXL Specification after agreeing to CXL Consortium’s Evaluation Copy Agreement (a copy of which is available <https://www.computeexpresslink.org/download-the-specification>, each such CXL Member must also be in compliance with all of the following CXL Consortium documents, policies and/or procedures (collectively, the “**CXL Governing Documents**”) in order for such CXL Member’s use and/or implementation of this CXL Specification to receive and enjoy all of the rights, benefits, privileges and protections of CXL Consortium membership: (i) CXL Consortium’s Intellectual Property Policy; (ii) CXL Consortium’s Bylaws; (iii) any and all other CXL Consortium policies and procedures; and (iv) the CXL Member’s Participation Agreement.

NOTICE TO NON-MEMBERS OF THE CXL CONSORTIUM:

If you are **not** a CXL Member and have received this publicly-available version of this CXL Specification, your use of this document is subject to your compliance with, and is limited by, all of the terms and conditions of the CXL Consortium’s Evaluation Copy Agreement (a copy of which is available at <https://www.computeexpresslink.org/download-the-specification>).

In addition to the restrictions set forth in the CXL Consortium’s Evaluation Copy Agreement, any references or citations to this document must acknowledge the Compute Express Link Consortium, Inc.’s sole and exclusive copyright ownership of this CXL Specification. The proper copyright citation or reference is as follows: “© 2019-2021 COMPUTE EXPRESS LINK CONSORTIUM, INC. ALL RIGHTS RESERVED.” When making any such citation or reference to this document you are not permitted to revise, alter, modify, make any derivatives of, or otherwise amend the referenced portion of this document in any way without the prior express written permission of the Compute Express Link Consortium, Inc.

Except for the limited rights explicitly given to a non-CXL Member pursuant to the explicit provisions of the CXL Consortium’s Evaluation Copy Agreement which governs the publicly-available version of this CXL Specification, nothing contained in this CXL Specification shall be deemed as granting (either expressly or impliedly) to any party that is **not** a CXL Member: (i) any kind of license to implement or use this CXL Specification or any portion or content described or contained therein, or any kind of license in or to any other intellectual property owned or controlled by the CXL Consortium, including without limitation any trademarks of the CXL Consortium.; or (ii) any benefits and/or rights as a CXL Member under any CXL Governing Documents.

LEGAL DISCLAIMERS FOR ALL PARTIES:

THIS DOCUMENT AND ALL SPECIFICATIONS AND/OR OTHER CONTENT PROVIDED HEREIN IS PROVIDED ON AN “**AS IS**” BASIS. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, COMPUTE EXPRESS LINK CONSORTIUM, INC. (ALONG WITH THE CONTRIBUTORS TO THIS DOCUMENT) HEREBY DISCLAIM ALL REPRESENTATIONS, WARRANTIES AND/OR COVENANTS, EITHER EXPRESS OR IMPLIED, STATUTORY OR AT COMMON LAW, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, VALIDITY, AND/OR NON-INFRINGEMENT.

In the event this CXL Specification makes any references (including without limitation any incorporation by reference) to another standard’s setting organization’s or any other party’s (“**Third Party**”) content or work, including without limitation any specifications or standards of any such Third Party (“**Third Party Specification**”), you are hereby notified that your use or implementation of any Third Party Specification: (i) is not governed by any of the CXL Governing Documents; (ii) may require your use of a Third Party’s patents, copyrights or other intellectual property rights, which in turn may require you to independently obtain a license or other consent from that Third Party in order to have full rights to implement or use that Third Party Specification; and/or (iii) may be governed by the intellectual property policy or other policies or procedures of the Third Party which owns the Third Party Specification. Any trademarks or service marks of any Third Party which may be referenced in this CXL Specification is owned by the respective owner of such marks.

NOTICE TO ALL PARTIES REGARDING THE PCI-SIG UNIQUE VALUE PROVIDED IN THIS CXL SPECIFICATION:

NOTICE TO USERS: THE UNIQUE VALUE THAT IS PROVIDED IN THIS CXL SPECIFICATION IS FOR USE IN VENDOR DEFINED MESSAGE FIELDS, DESIGNATED VENDOR SPECIFIC EXTENDED CAPABILITIES, AND ALTERNATE PROTOCOL NEGOTIATION ONLY AND MAY NOT BE USED IN ANY OTHER MANNER, AND A USER OF THE UNIQUE VALUE MAY NOT USE THE UNIQUE VALUE IN A MANNER THAT (A) ALTERS, MODIFIES, HARMS OR DAMAGES THE TECHNICAL FUNCTIONING, SAFETY OR SECURITY OF THE PCI-SIG ECOSYSTEM OR ANY PORTION THEREOF, OR (B) COULD OR WOULD REASONABLY BE DETERMINED TO ALTER, MODIFY, HARM OR DAMAGE THE TECHNICAL FUNCTIONING, SAFETY OR SECURITY OF THE PCI-SIG ECOSYSTEM OR ANY PORTION THEREOF (FOR PURPOSES OF THIS NOTICE, “**PCI-SIG ECOSYSTEM**” MEANS THE PCI-SIG SPECIFICATIONS, MEMBERS OF PCI-SIG AND THEIR ASSOCIATED PRODUCTS AND SERVICES THAT INCORPORATE ALL OR A PORTION OF A PCI-SIG SPECIFICATION AND EXTENDS TO THOSE PRODUCTS AND SERVICES INTERFACING WITH PCI-SIG MEMBER PRODUCTS AND SERVICES).



CXL ENGINEERING CHANGE NOTICE

TITLE:	Type 3 Management Using MCTP CCI
DATE:	Introduced date (01/26/2021) Updated date (11/16/2021)
AFFECTED DOCUMENT:	CXL 2.0
SPONSOR:	Vincent Haché, Microchip Mahesh Natu, Intel Chris Petersen, Facebook Joe Cowan, HPe

Part I

1. Summary of Functional Changes

This ECN contains changes required to define support for Type 3 device management over MCTP-capable interfaces via CXL-specific transport binding specifications. It also adds updates and clarifications to some management command definitions to improve the behavior of those commands when transmitted over MCTP-capable interfaces.

It defines a new command, "Get Supported Logs Sub-List", to improve compatibility for requesters whose receive buffer size is limited, and it clarifies the definition of command input parameters used to limit response payload size.

2. Benefits as a Result of the Changes

CXL 2.0 defines a comprehensive management scheme for Type 3 devices that is only accessible through the device's Mailbox Registers. However, component management using MCTP-capable interfaces is a well-established convention.

Existing open management standards can be used but will add additional requirements and complexity to devices that implement support for both the Mailbox Registers and MCTP-based management. This ECN simplifies component implementation, allowing for the reuse of application-layer processing of management commands over both the Mailbox Register and MCTP-capable interfaces. It also enables light-weight BMC implementations that do not require the sophistication and complexity of existing open management standards.

3. Assessment of the Impact

This new management path is optional, so there is no impact to existing systems. This ECN is limited to Type 3 components, so it only impacts implementations of Type 3 devices and systems managing Type 3 devices supporting this ECN. Future systems may implement support for MCTP-capable interfaces, as needed. Systems that support existing open management standards for generic management activities (eg. FW

Copy
Evaluation
Copy

update) may be upgraded to support this new management scheme if one or more components in that system only support CXL-defined management commands.

4. Analysis of the Hardware Implications

There is no impact to existing system hardware. Components adding support for this optional management capability may implement MCTP-capable interfaces. Systems with existing MCTP-capable interfaces to their CXL components may use that connectivity to implement support for these new management capabilities.

5. Analysis of the Software Implications

There is no impact to existing software. Out-of-band management software may implement support for the new transport binding, as necessary. Existing management software that uses the Mailbox Registers may implement support for the new, optional “Get Supported Logs Sub-List” command.

6. Analysis of the Compliance and Test Implications

This ECN does not repurpose any reserved bits and as such, does not impact existing compliance tests. No new compliance tests are required.

Part II

Detailed Description of the change

- 1.2 Terminology
 - Added definitions for ‘CCI’, ‘MB’, and ‘Sideband’
- 7.6.2 Fabric Management Model
 - Generalized the discussion to include references to CCIs
- 7.6.3 CCI Message Format and Transport Protocol
 - Generalized the definition of the transport layer so it can cover all component management commands
- [NEW] 7.6.3.1 Transport Details for MLD Components
 - New section added clarifying transport details for commands tunneled to MLDs
- 7.6.6.4 Event Notifications
 - Updated to reflect new command names for Get MCTP Event Interrupt Policy and Set MCTP Event Interrupt Policy
- 7.6.7.2.1 Get Virtual Switch Info
 - Added minimum values to fields in the Get VCS Info command
- 7.6.7.4.1 Tunnel Management Command
 - Expanded scope of this command so that it can be used on MCTP-based CCIs to pass commands to LDs of an MLD
- 7.6.7.5 MLD Component Command Set
 - Added minimum values to fields controlling response size in multiple commands
- 8.2.8.4 Mailbox Registers
 - Moved the general discussion about CCIs to 9.16
- 8.2.8.4.2 Attributes of the Second Mailbox
 - Added a requirement that CEL output for secondary mailbox must match that of the primary mailbox
- 8.2.8.4.5.1 Command Return Codes

- Updated description of “Unsupported Mailbox” field to account for MCTP CCIs
- 8.2.9 Component Command Interface
 - Updated Requirements column to specify permissions for commands to mailbox and MCTP-based CCIs
- 8.2.9.1.6 Get MCTP Event Interrupt Policy
 - Renamed command
 - Added notifications for the completion of Background Operations
- 8.2.9.1.7 Set MCTP Event Interrupt Policy
 - Renamed command
- 8.2.9.1.8 Event Notification
 - Updated to reflect new command name for Set MCTP Event Interrupt Policy and the addition of notifications for the completion of Background Operations
- 8.2.9.2.2 Transfer FW
 - Added clarifications for retransmission of package parts and handling “Transfer FW” requests from multiple CCIs
- 8.2.9.4.1 Get Supported Logs
 - Added implementation note recommending the use Get Supported Logs Sub-List and a requirement for Type 3 devices supporting Logs on an MCTP-based CCI
- 8.2.9.4.2.1 Command Effects Log
 - Specifying “Secondary Mailbox Supported” is reserved when queried from any other CCI than a mailbox CCI
- [NEW] 8.2.9.4.3 Get Supported Logs Sub-List
 - New command added for reading in supported log types in small sub-lists
- 8.2.9.5 Memory Device Commands
 - Renamed “Identify” command set to “Identify Memory Device”
 - Added Permission Info to all opcodes
- 8.2.9.5.1 Identify Memory Device
 - Renamed “Identify” command set to “Identify Memory Device”
- 8.2.9.6 FM API Commands
 - Added Permission Info, Input Payload Length, and Output Payload Length to all opcodes
 - Fixed a typo in the names of the “Send LD CXL.io Configuration Request” and “Send LD CXL.io Memory Request” commands
- 8.2.9.7.1 Identify
 - Added component type field to the Identify command response payload
- 9.15 Manageability Model for CXL Devices
 - Replace performance counter example with event log
- [NEW] 9.16 Component Command Interface
 - Created new subsection with general theory of operation of CCIs, with some content moved from 8.2.8.4
 - Defines “Component Command Interface (CCI)” as the target of all commands
 - Defines specific properties of MCTP-based CCIs
- 12.2.3.2 Memory Error Logging and Signaling Enhancements
 - Made the references to the “MMIO mailbox” more generic to account for CCIs
- [NEW] 14.9.107.7 MCTP-Based CCI Test
 - Compliance test added for MCTP-based CCIs

1.2 Terminology / Acronyms

Please refer to the PCI Express Specification for additional terminology and acronym definitions beyond those listed in Table 1.

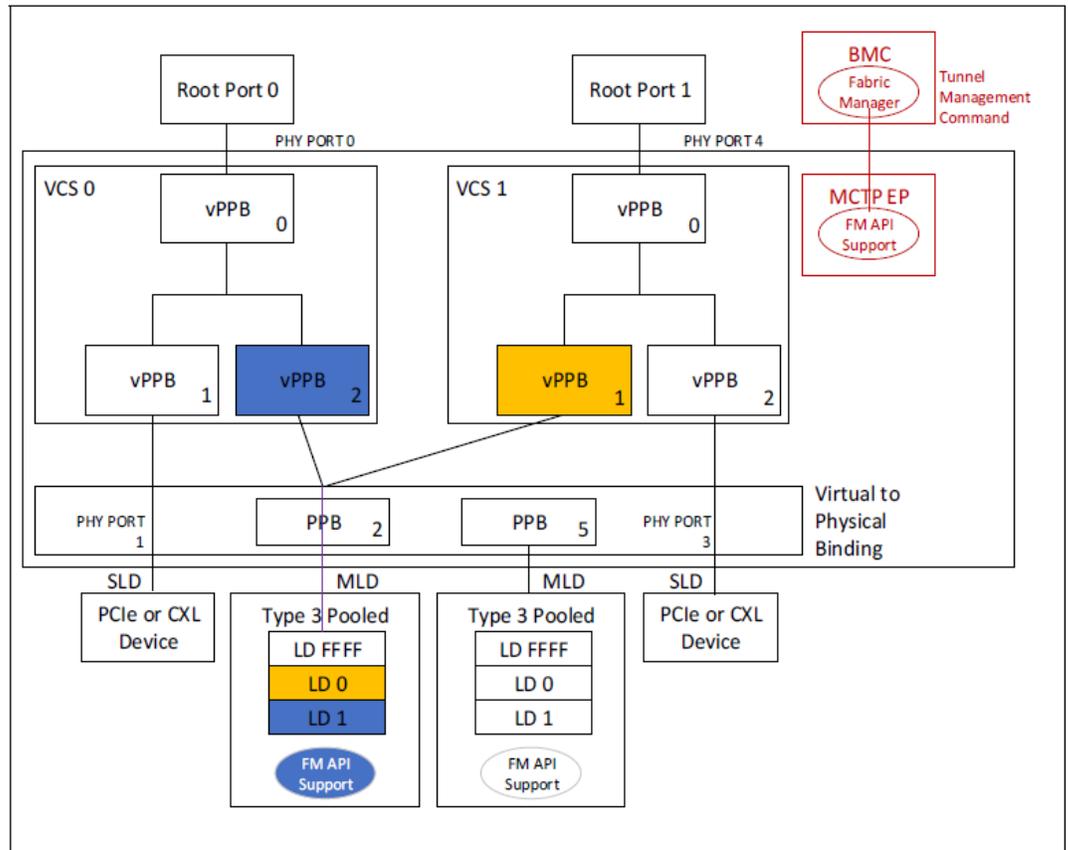
Table 1. Terminology / Acronyms

Term / Acronym	Definition
...	
CCI	Component Command Interface (Refer to Section 9.16)
...	
MB	Megabyte - 2²⁰ bytes (1,048,576 bytes)
...	
Sideband	Signal used for device detection, configuration, and hot plug in PCIe connectors, as defined in the PCIe Express Base Specification
...	

7.6.2 Fabric Management Model

CXL devices are configured by FMs using the Fabric Manager Application Programming Interface (FM API) Command Sets, as defined in Section 8.2.9.6, exposed through an MCTP-capable interface. Refer to Section 9.16 for details on the CCI processing these commands.

Figure 123. Example of Fabric Management Model



FMs issue request messages and CXL devices issue response messages. CXL components may also issue the "Event Notification" request if notifications are supported by the component and the FM has requested notifications from the component using the **Set ~~00B~~MCTP Event**

Interrupt Policy command. Refer to Section 7.6.3 for transport protocol details. ~~MCTP may be used as the transport protocol.~~

~~MCTP messages may be sent over a variety of interfaces.~~ The following list provides a number of examples of connectivity between an FM and a component's CCI, but should not be considered a complete list:

- An FM directly connected to a CXL device through any ~~⊖~~ **MCTP-capable** interconnect can issue FM commands directly to the device. This includes delivery over ~~sideband~~ **MCTP-capable interfaces buses** such as SMBus as well as VDM delivery over a standard PCIe tree topology where the responder is mapped to a CXL attached device.
- An FM directly connected to a CXL switch may use the switch to tunnel FM commands to MLD components directly attached to the switch. In this case, the FM issues the "Tunnel Management Command" command to the switch specifying the switch port to which the device is connected. Responses are returned to the FM by the switch. In addition to MCTP message delivery, the FM command set provides the FM with the ability to have the switch proxy config cycles and memory accesses to a Downstream Port on the FM's behalf.
- An FM may also be embedded within a CXL device. The communication interface between such a FW module and the device hardware is considered a vendor implementation detail and is not covered in this specification.

~~Support for multiple Management Ports and coordination of multiple active FMs falls outside the scope of this specification.~~

7.6.3 **CCI Message Format and FM Command Transport Protocol**

~~CCI commands are transmitted across MCTP-capable interconnects as MCTP messages using the format defined in Figure 124. FM API commands may be transported as MCTP Message as defined in CXL FM API over MCTP Binding Specification³.~~

Figure 124. FM API CCI Message Body

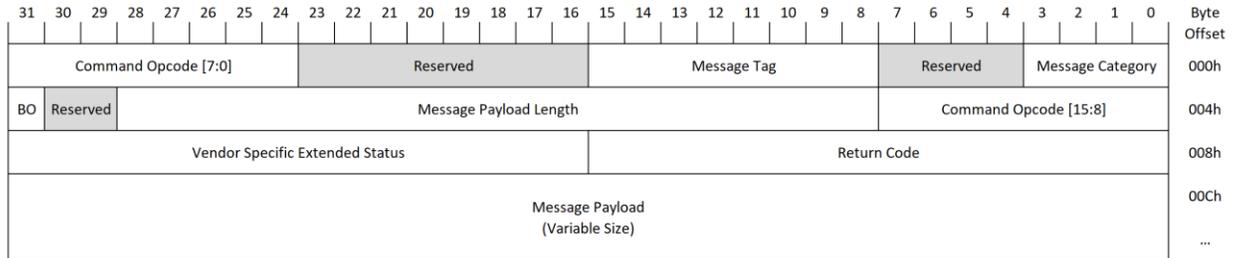


Table 84. CCI FM API Message Format

Byte Offset	Description
000h	Bits (3:0) - Message Category: Type of CCI-FM-API message: <ul style="list-style-type: none"> • 0h = Request • 1h = Response • All other encodings reserved Bits (7:4) - Reserved
001h	Message Tag: Tag number assigned to request messages by the FM requester used to track response messages when multiple request messages are outstanding. Response messages shall use the tag number from the corresponding Request message.
002h	Reserved
003h	Command Opcode [7:0]: As defined in Table 152 , Table 174 and Table 205 .
004h	Command Opcode [15:8]
005h	Message Payload Length [7:0]: As defined in Table 152 , Table 174 and Table 205 .
006h	Message Payload Length [15:8]
007h	Bits (4:0) - Message Payload Length [20:16] Bit (6:5) - Reserved Bit (7) - Background Operation (BO): As defined in Section 8.2.8.4.6. Must be 0 for Request messages.
008h	Return Code [7:0]: As defined in Table 150 . Must be 0 for Request messages
009h	Return Code [15:8]

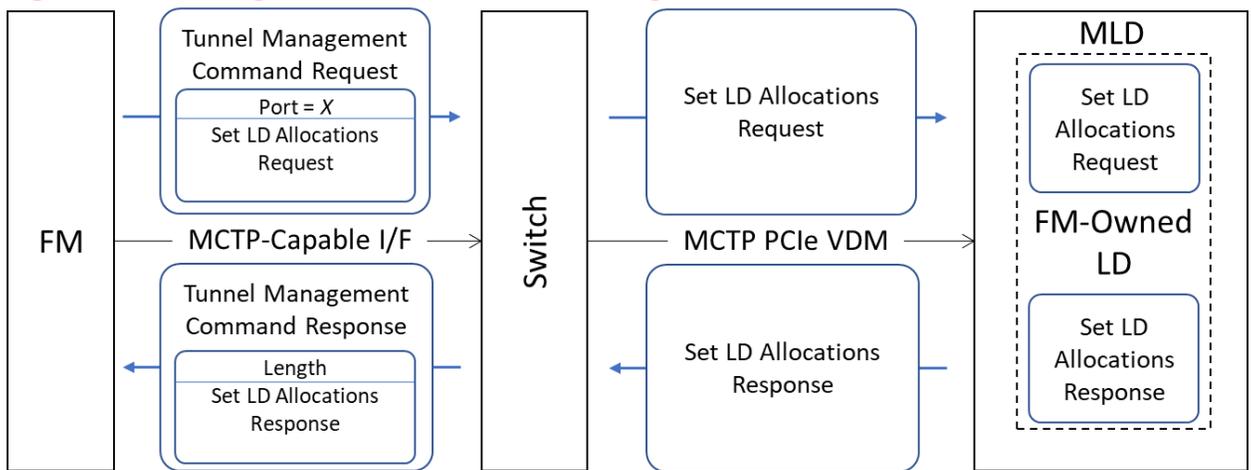
00Ah	Vendor Specific Extended Status [7:0]: As defined in Section 8.2.8.4.6 . Must be 0 for Request messages
00Bh	Vendor Specific Extended Status [15:8]
00Ch	Message Payload: Variably-sized payload for message in little endian. The length of this field is specified in the Message Payload Length field above. The format depends on Opcode and Message Category , as defined in Section 7.6.7 .

Commands from the FM API Command Sets may be transported as MCTP messages as defined in [CXL FM API over MCTP Binding Specification](#)¹. All other CCI commands may be transported as MCTP messages as defined by the respective binding specification.

7.6.3.1 Transport Details for MLD Components

MLD components that do not implement an MCTP-capable interconnect other than their CXL interface shall expose a CCI through their CXL interface(s) using the [MCTP PCIe VDM Transport Binding Specification](#). FMs shall use the Tunnel Management Command to pass requests to the FM-owned LD, as illustrated in Figure X.

Figure X. Tunneling Commands to an MLD through a CXL Switch



7.6.6.4 Event Notifications

- To facilitate some system operations, the FM requires event notifications so it can execute its role in the process in a timely manner (e.g., notifying hosts of an asserted Attention Button on an MLD during Managed Hot-Removal). If supported by the device, the FM can check the current event notification settings with the Get [ΘΘBMCTP](#) Event Interrupt Policy command and modify them with the Set [ΘΘBMCTP](#) Event Interrupt Policy command.

Table 97. Get Virtual CXL Switch Info Request Payload

Bytes	Description
0	Start vPPB: Specifies the ID of the first vPPB requested in the vPPB information list (bytes 4 – 7 in Table 99). This enables compatibility with devices that have small maximum command message sizes.
1	vPPB List Limit: The maximum number of vPPB information entries to include in the response (bytes 4 – 7 in Table 99). This enables compatibility with devices that have small maximum command message sizes. <i>This field shall have a minimum value of 1.</i>
2	Number of VCSs: Number of VCSs requested. <i>This field shall have a minimum value of 1.</i>
(Number of VCSs + 2) - 3	VCS ID List: 1 byte ID of requested VCS, repeated Number of VCSs times.

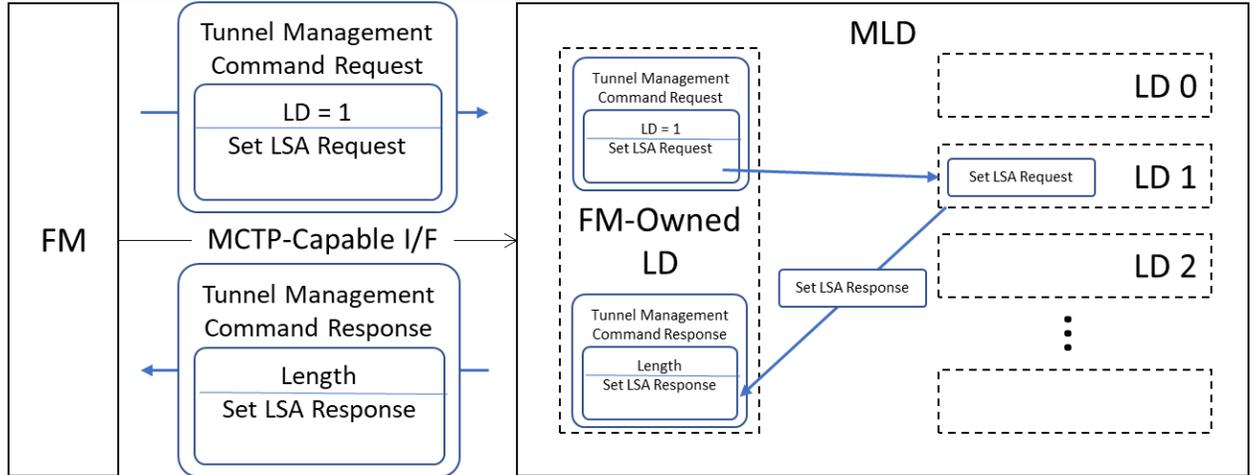
7.6.7.4.1 Tunnel Management Command (Opcode 5300h)

This command is used to tunnel FM-C commands *encapsulated as an MCTP request* to LDs within an MLD.

When sent to a CXL switch, the provided command is tunneled by the switch to the FM-owned LD (LD=0xFFFF) of the MLD on the specified port using the transport defined in Section 7.6.3.1.

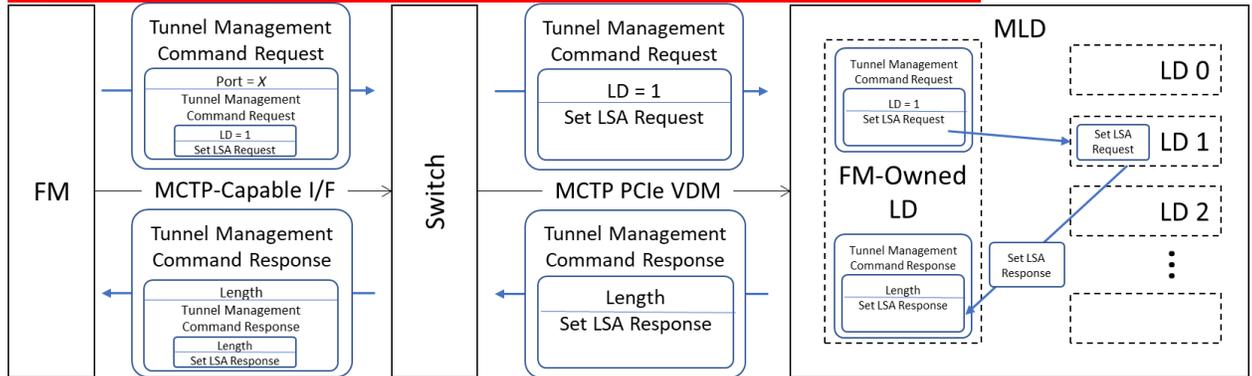
When sent to an MLD, this provided command is tunneled by the FM-owned LD to the specified LD, as illustrated in Figure Y.

Figure Y. Tunneling Commands to an LD in an MLD



The Management Command input payload field includes the tunneled command encapsulated in the CCI Message Format, as defined in Figure 124. This can include an additional layer of tunneling for commands issued to LDs in an MLD accessible only through a CXL switch’s MLD port, as illustrated in Figure Z.

Figure Z. Tunneling Commands to an LD in an MLD through a CXL Switch



Response size varies based on the tunneled FM command’s definition. Valid targets for the tunneled commands include switch MLD ports and valid LDs within an MLD. Tunneled FM commands sent to any ~~port other targets than an MLD port~~ shall be discarded and this command’s will indicate complete with an “Invalid Parameter” return code-failure. The FM-owned LD (LD=0xFFFF) is an invalid target.

The Tunnel Management Command itself does not cause any Command Effects, but the Management Command provided in the request will cause Command Effects as per its definition.

Possible Command Return Codes:

- Success

- Invalid Parameter
- Unsupported
- Internal Error
- Retry Required

Command Effects:

- None

Table 104 Tunnel Management Command Input Payload

Byte Offset	Length	Description
0	1	Port or LD ID: Egress port ID <i>for commands sent to a switch, or LD ID for commands sent to an MLD</i>
1	1	Reserved
3-2	2	Command Size: Number of valid bytes in Management Command
Varies-4	Varies	Management Command: Raw MCTP Message Body to transmit Request message formatted in the CCI Message Format as defined in Figure 124.

Table 105 Tunnel Management Command Output Payload

Byte Offset	Length	Description
1-0	2	Response Length: Number of valid bytes in Response Message .
3-2	2	Reserved
Varies-4	Varies	Response Message: Response message sent by MLD formatted in the CCI Message Format as defined in Figure 124.

...

Table Z Get LD Allocations Request Payload

Bytes	Description
0	Start LD ID: ID of the first LD in the LD Allocation List
1	LD Allocation List Limit: Maximum number of LD information blocks returned. <i>This field shall have a minimum value of 1.</i>

...

Table 114 Set LD Allocations Request Payload

Bytes	Description
0	Number of LDs: Number of LDs to configure. <i>This field shall have a minimum value of 1.</i>
1	Start LD ID: ID of the first LD in the LD Allocation List
3-2	Reserved
Varies-4	LD Allocation List: LD Allocation blocks for each LD starting at Start LD ID , as defined in Table 113, repeated Number of LDs times.

...

Table A Payload for Get QoS Allocated BW Request

Bytes	Description
0	Number of LDs: Number of LDs configured. <i>This field shall have a minimum value of 1.</i>
1	Start LD ID: ID of the first LD in the LD Allocation List

...

Table B Payload for Get QoS BW Limit Request

Bytes	Description
0	Number of LDs: Number of LDs to query. <i>This field shall have a minimum value of 1.</i>
1	Start LD ID: ID of the first LD in the LD Allocation List

...

8.2.8.4 Mailbox Registers (Offset Varies)

The mailbox registers provide the ability to issue a command to the device. ~~Refer to Section 8.2.9 for details about the commands.~~ There are two types of mailboxes provided through the device's register interface: primary and secondary. Each mailbox represents a unique CCI instance in the device and the properties of each instance are defined in Section 9.16.1. The secondary mailbox does not support background operations. The status of a background operation issued to a device's primary mailbox can be retrieved from the Background Command Status Register, as detailed in Section 8.2.8.4.7.

The register interface for both types of mailboxes is the same and is described in this section. The difference between the two types of mailboxes is their intended use and commands

allowed. Details on these differences are described in [Section 8.2.8.4.1](#) and [Section 8.2.8.4.2](#). ~~Devices implementing more than one mailbox shall process commands from those mailboxes in a manner which avoids “starvation,” so that commands submitted to one mailbox do not prevent commands from other mailboxes from being handled. The exact algorithm for accepting commands from multiple mailboxes is implementation specific.~~

The [primary and secondary](#) mailbox interfaces shall be used in a single-threaded manner only. It is software’s responsibility to avoid simultaneous, uncoordinated access to [the](#) mailbox [registers](#) using techniques such as locking.

...

8.2.8.4.2 Attributes of the Secondary Mailbox

The secondary mailbox, if implemented by a device, supports only a subset of the commands described in [Section 8.2.9](#). The Command Effects Log shall specify which commands are allowed on the secondary mailbox, and all other commands shall return the error Unsupported Mailbox [or CCI](#). The secondary mailbox does not support mailbox completion interrupts. Implementation of the secondary mailbox is optional.

...

[The secondary mailbox shall return identical information as the primary mailbox for a Get Log command issued with Log Identifier=CEL.](#) Devices shall indicate which commands are allowed on the secondary mailbox by setting the Secondary Mailbox Supported flag for the supported opcodes in the Command Effects Log. Exactly which commands are supported on the secondary mailbox is implementation specific. It is recommended (but not required) that the secondary mailbox supports all commands in the Events, Logs, and Identify command sets defined in [Section 8.2.9](#).

...

8.2.8.4.5.1 Command Return Codes

In general, retries are not recommended for commands that return an error except when indicated in the return code definition.

Table 150. Command Return Codes

Value	Definition
...	
0015h	Unsupported Mailbox or CCI: The command is not supported on the mailbox or CCI it was issued on. Used to indicate an unsupported command issued on the secondary mailbox.
...	

8.2.9 ~~EXL~~ Device Component Command Interface

...

Table 152. Generic ~~EXL~~ Device Component Command Opcodes

Command Set Bits[15:8]		Opcode			Required*			Input Payload Size (B)	Output Payload Size (B)***
		Command Bits[7:0]		Combined Opcode	Type 1/2/3		Switch		
					Mailbox	Type 3 MCTP			
00h	Information and Status	01h	Identify (Section 8.2.9.7.1)	0001h	<u>PMB</u>	<u>MO</u>	<u>MS</u>	0	+7 12h
		02h	Background Operation Status (Section 8.2.9.7.2)	0002h	<u>PMB</u>	<u>MO</u>	<u>MS</u>	0	8
		03h	Get Output Payload Limit (Section 8.2.9.7.3)	0003h	<u>PMB</u>	<u>MO</u>	<u>MS</u>	0	1
		04h	Set Output Payload Limit (Section 8.2.9.7.4)	0004h	<u>PMB</u>	<u>MO</u>	<u>MS</u>	1	1
01h	Events	00h	Get Event Records (Section 8.2.9.1.2)	0100h	<u>M</u>	<u>Q</u>	<u>Q</u>	1	20h+

		01h	Clear Event Records (Section 8.2.9.1.3)	0101h	M	Q	Q	8+	0
		02h	Get Event Interrupt Policy (Section 8.2.9.1.4)	0102h	M	P	P	0	4
		03h	Set Event Interrupt Policy (Section 8.2.9.1.5)	0103h	M	P	P	4	0
		04h	Get ΘΘB MCTP Event Interrupt Policy (Section 8.2.9.1.5)	0104h	PMB	Q	Q	2	0
		05h	Set ΘΘB MCTP Event Interrupt Policy (Section 8.2.9.1.5)	0105h	PMB	Q	Q	2	2
		06h	Event Notification (Section 8.2.9.1.5)	0106h	PMB	Q	Q	2	0
02h	Firmware Update	00h	Get FW Info (Section 8.2.9.2.1)	0200h	Q	Q	Q	0	50h
		01h	Transfer FW (Section 8.2.9.2.2)	0201h	Q	Q	Q	80h+	0
		02h	Activate FW (Section 8.2.9.2.3)	0202h	Q	Q	Q	2	0
03h	Timestamp	00h	Get Timestamp (Section 8.2.9.3.1)	0300h	Q	Q	Q	0	8
		01h	Set Timestamp (Section 8.2.9.3.2)	0301h	Q	Q	Q	8	0
04h	Logs	00h	Get Supported Logs (Section 8.2.9.4.1)	0400h	M	O**	Q	0	8+
		01h	Get Log (Section 8.2.9.4.2)	0401h	M	Q	Q	18h	0+
		05h	Get Supported Logs Sub-List (Section 8.2.9.4.3)	0405h	Q	O**	Q	2	8+

*M = mandatory for all devices that implement the CXL Device Register entry (Identifier=03h) in the Register Locator DVSEC (Section 8.1.9); PMB = prohibited on the primary and secondary mailboxes; P = prohibited; MO = mandatory for components that implement an MCTP-based CCI; MS = mandatory for components that support the FM API MCTP message type; O = Optional.
 ** Type 3 devices implementing support for the Get Supported Logs opcode on an MCTP-based CCI shall also support the Get Supported Logs Sub-List opcode
 *** Indicates the minimum output payload size for a successful completion. Commands with variable output payload sizes are marked with '+'. Actual valid bytes in the output payload are indicated by the 'Payload Length' field in the Mailbox registers or in the Response packet size for MCTP-based CCIs.

8.2.9.1.6 Get ~~ΘΘB~~MCTP Event Interrupt Policy (Opcode 0104h)

Retrieve the settings for interrupts that are signaled for component events over ~~ΘΘB~~MCTP management media. When notifications are enabled for a particular log on a component, the component may issue up to 1 Event Notification for that log type when the log contents transition from zero entries to one or more entries. The FM must be able to receive at least 1 Event Notification message from the component for each log for which interrupts have been enabled.

When notifications are enabled for Background Operation completion, the component shall issue 1 Event Notification upon the completion of every Background Operation. This setting shall be ignored if the component does not support Background Operations.

Possible Command Return Codes:

- Success
- Internal Error
- Retry Required
- Invalid Payload Length

Command Effects:

- None

Table XX Payload for Get ~~ΘΘB~~MCTP Event Interrupt Policy Output, Set ~~ΘΘB~~MCTP Event Interrupt Policy Input, and Set ~~ΘΘB~~MCTP Event Interrupt Policy Output

Byte Offset	Length	Description
0	2	Event Interrupt Settings: Bitmask indicating whether event notifications are enabled (1) or disabled (0) for a particular event Bit[0]: New uncleared Informational Event Log record(s) Bit[1]: New uncleared Warning Event Log record(s) Bit[2]: New uncleared Failure Event Log record(s)

		Bit[3]: New uncleared Fatal Event Log record(s) Bits[14-4]: Reserved Bit[15]: Background Operation completed
--	--	--

8.2.9.1.7 Set ~~00B~~MCTP Event Interrupt Policy (Opcode 0105h)

...

8.2.9.1.8 Event Notification (Opcode 0106h)

This command is used by a CXL component to send notifications to the FM. It is only sent by CXL components. Any commands of this type received by CXL components shall be silently discarded.

This command is a notification to the FM that there are new events to be read from the event logs specified in the "Event-Log" payload field. The A notification for log-based events is triggered when both of the following conditions are met: the log transitions from having no entries to having one or more entries and ~~00B~~MCTP interrupts have been enabled with the Set ~~00B~~MCTP Event Interrupt Policy command. Log entries are cleared with the Clear Event Logs command. After clearing log entries, the FM should use the Get Event Records command to confirm that all entries have been cleared from the log to ensure that the notification trigger has been re-armed. A notification for Background Operations is sent when a Background Operation completes.

The FM acknowledges a notification by returning a response. The component shall retransmit the notification every 1 ms using the same Message Tag value in the transport header until the FM has returned a response with the 'Success' return code, up to a maximum of 10 retries. No additional Event Notifications shall be sent until the component has received a response from the FM.

Possible Command Return Codes:

- Success
- Invalid Parameter
- Internal Error
- Retry Required

Command Effects:

- None

Table Y Event Notification Input Payload

Byte Offset	Length	Description
0	2	Event: Bitmask indicating the event: Bit 0 = Informational Event Log has uncleared record(s) Bit 1 = Warning Event Log has uncleared record(s) Bit 2 = Failure Event Log has uncleared record(s) Bit 3 = Fatal Event Log has uncleared record(s) Bits 14-4 = Reserved <u>Bit 15 = Background Operation completed</u>

...

8.2.9.2.2 Transfer FW (Opcode 0201h)

Transfer all or part of a FW package from the caller to the device. FW packages shall be 128 byte aligned.

If the FW package is transferred in its entirety, the caller makes one call to Update FW with Action = Full FW Transfer.

If a FW package is transferred in parts, the caller makes one call to Transfer FW with Action = Initiate FW Transfer, zero or more calls with Action = Continue FW Transfer, and one call with Action = End Transfer or Abort Transfer. The FW package parts shall be transferred in order,

otherwise the device shall return the FW Transfer Out of Order return code. Back-to-back retransmission of any FW package part is permitted during a transfer.

If ~~the~~ a FW package transfer initiated on a Mailbox CCI is interrupted by a Conventional or CXL reset, the FW package transfer shall be aborted by the device. If a FW package transfer is aborted prior to the entire FW package being successfully stored on the device, the device shall require the FW package transfer to be started from the beginning of the FW package.

Only one FW package may be transferred to a single CCI at a time. The device shall return the FW Transfer in Progress return code if it receives a Transfer FW command with Action = Full FW Transfer or Action = Initiate FW Transfer until the current FW package transfer is completed or aborted.

In case of a device with multiple CCIs, the device shall return the FW Transfer in Progress return code if it receives a Transfer FW request on any other CCI than the CCI that initiated the current FW package transfer until that transfer is completed or aborted. The timeout between FW package parts is 30 seconds, after which point the device shall consider the current FW package transfer aborted.

Once the entire FW package is fully transferred to the device, the device shall verify the FW package and store it in the specified slot. Verification of the FW package is vendor specific.

...

8.2.9.4.1 Get Supported Logs (Opcode 0400h)

Retrieve the list of device specific logs (identified by UUID) and the maximum size of each Log.

Possible Command Return Codes:

- Success
- Internal Error
- Retry Required
- Invalid Payload Length

Command Effects:

- None

IMPLEMENTATION NOTE

It is strongly recommended that Get Supported Logs Sub-List is supported by components and used by management software instead of Get Supported Logs so that requesters may control the size of the output payload, as needed. Type 3 devices implementing support for the Get Supported Logs opcode on an MCTP-based CCI shall also support the Get Supported Logs Sub-List opcode.

...

8.2.9.4.2.1 Command Effects Log

The Command Effect Log (CEL) is a variable length log page that reports ~~support for~~ each command supported by the CCI that was queried (with the exception of the secondary mailbox, as described in Section 8.2.8.4.2) and the effect each command will have on the device subsystem.

Devices shall implement the CEL for all commands supported by the device, including any vendor specific commands that extend beyond those specified in this specification.

Some host drivers may not allow unspecified commands to be passed through to the device if the commands are not advertised in the CEL.

...

Table 173. CEL Entry Structure

Byte Offset	Length	Description
0	2	Opcode: The command opcode.
2	2	<p>Command Effect: Bit mask containing one or more effects for the command opcode</p> <ul style="list-style-type: none"> • Bit[0]: Configuration Change after Cold Reset - When set, this opcode makes a driver visible change to the configuration of the device or data contained within persistent memory regions of the device. The change does not take effect until a device cold reset. • Bit[1]: Immediate Configuration Change - When set, this opcode makes an immediate driver visible change to the configuration of the device or data contained within persistent memory regions of the device. • Bit[2]: Immediate Data Change - When set, this opcode makes an immediate driver visible change to the data written to the device. • Bit[3]: Immediate Policy Change - When set, this opcode makes an immediate change to the policies utilized by the device. • Bit[4]: Immediate Log Change - When set, this opcode makes an immediate change to a device log. • Bit[5]: Security State Change - When set, this opcode results in an immediate driver visible change in the security state of the device. Security state changes that require a reboot to take effect do not use this effect. • Bit[6]: Background Operation - When set, this opcode is executed in the background. • Bit[7]: Secondary Mailbox Supported - When set, submitting this opcode via the secondary mailbox is supported, otherwise this opcode will return Unsupported Mailbox or CCI if issued on the secondary mailbox. <u>Only valid when returned on the primary or secondary mailbox. This bit is reserved if the CEL is being returned from any CCI other than the primary or secondary mailbox.</u> • Bits[15:8]: Reserved, shall be set to zero.

8.2.9.4.3 Get Supported Logs Sub-List (Opcode 0405h)

Retrieve a sub-list of device specific logs (identified by UUID) and the maximum capacity of each log. The output of this command shall be consistent with the output of the Get Supported Logs command.

Possible Command Return Codes:

- Success
- Internal Error
- Retry Required
- Invalid Payload Length

Command Effects:

- None

IMPLEMENTATION NOTE

It is strongly recommended that Get Supported Logs Sub-List is supported by components and used by management software instead of Get Supported Logs so that requesters may control the size of the output payload, as needed. Type 3 devices implementing support for the Get Supported Logs opcode on an MCTP-based CCI shall also support the Get Supported Logs Sub-List opcode.

Table C Get Supported Logs Input Payload

Byte Offset	Length	Description
0	1	Maximum Number of Supported Log Entries: Maximum number of Supported Log Entries requested. This field shall have a minimum value of 1.
1	1	Start Log Entry Index: Index of the first Supported Log Entry requested

Table D Get Supported Logs Output Payload

Byte Offset	Length	Description
0	2	Number of Supported Log Entries: The number of Supported Log Entries returned in the output payload
2	2	Total Number of Supported Log Entries: The total number of Supported Log Entries supported by the component
4	1	Start Log Entry Index: Index of the first Supported Log Entry in the output payload
5	3	Reserved
8	Varies	Supported Log Entries: Device specific list of supported log identifier UUIDs and the maximum capacity of each log, as defined in Table 169.

8.2.9.5 Memory Device Command Sets

Table 174. CXL Memory Device Command Opcodes

Command Set Bits[15:8]		Opcode			Required*		Input Payload Size (B)	Output Payload Size (B)***
					Type 1/2/3			
					Mailbox	Type 3 MCTP		
40h	Identify <u>Memory Device</u>	00h	Identify Memory Device (Section 8.2.9.5.1.1)	4000h	M	<u>M</u>	0	43h
41h	Capacity Config and Label Storage	00h	Get Partition Info (Section 8.2.9.5.2.1)	4100h	O	<u>Q</u>	0	20h
		01h	Set Partition Info (Section 8.2.9.5.2.2)	4101h	O	<u>Q**</u>	9	0
		02h	Get LSA (Section 8.2.9.5.2.3)	4102h	PM	<u>Q</u>	8	0+
		03h	Set LSA (Section 8.2.9.5.2.4)	4103h	PM	<u>Q**</u>	8+	0
42h	Health Info and Alerts	00h	Get Health Info (Section 8.2.9.5.3.1)	4200h	M	<u>Q</u>	0	12h
		01h	Get Alert Configuration (Section 8.2.9.5.3.2)	4201h	M	<u>Q</u>	0	10h
		02h	Set Alert Configuration (Section 8.2.9.5.3.3)	4202h	M	<u>Q</u>	0Ch	0
		03h	Get Shutdown State (Section 8.2.9.5.3.4)	4203h	PM	<u>Q</u>	0	1
		04h	Set Shutdown State (Section 8.2.9.5.3.5)	4204h	PM	<u>P</u>	1	0
43h	Media and Poison Mgmt	00h	Get Poison List (Section 8.2.9.5.4.1)	4300h	PM	<u>Q</u>	10h	20h+
		01h	Inject Poison (Section 8.2.9.5.4.2)	4301h	O	<u>P</u>	8	0
		02h	Clear Poison (Section 8.2.9.5.4.3)	4302h	O	<u>Q**</u>	48h	0
		03h	Get Scan Media Capabilities (Section 8.2.9.5.4.4)	4303h	PM	<u>Q</u>	10h	4
		04h	Scan Media (Section 8.2.9.5.4.5)	4304h	PM	<u>Q**</u>	11h	0
		05h	Get Scan Media Results (Section 8.2.9.5.4.6)	4305h	PM	<u>Q</u>	0	20h+
44h	Sanitize	00h	Sanitize (Section 8.2.9.5.5.1)	4400h	O	<u>Q**</u>	0	0
		01h	Secure Erase (Section 8.2.9.5.5.2)	4401h	O	<u>Q**</u>	0	0
45h	Persistent Memory Data-at-rest Security	00h	Get Security State (Section 8.2.9.5.6.1)	4500h	O	<u>Q</u>	0	4
		01h	Set Passphrase (Section 8.2.9.5.6.2)	4501h	O	<u>Q**</u>	60h	0
		02h	Disable Passphrase (Section 8.2.9.5.6.3)	4502h	O	<u>Q**</u>	40h	0
		03h	Unlock (Section 8.2.9.5.6.4)	4503h	O	<u>Q**</u>	20h	0
		04h	Freeze Security State (Section 8.2.9.5.6.5)	4504h	O	<u>Q**</u>	0	0
		05h	Passphrase Secure Erase (Section 8.2.9.5.6.6)	4505h	O	<u>Q**</u>	40h	0
46h	Security Passthrough	00h	Security Send (Section 8.2.9.5.7.1)	4600h	O	<u>Q**</u>	8+	0
		01h	Security Receive (Section 8.2.9.5.7.2)	4601h	O	<u>Q</u>	8	0+
47h	SLD QoS Telemetry	00h	Get SLD QoS Control (Section 8.2.9.5.8.1)	4700h	O	<u>Q</u>	0	4
		01h	Set SLD QoS Control (Section 8.2.9.5.8.2)	4701h	O	<u>Q**</u>	4	0
		02h	Get SLD QoS Status (Section 8.2.9.5.8.3)	4702h	O	<u>Q</u>	0	1

*M = mandatory for all CXL memory devices; PM = mandatory for CXL memory devices that support persistence; O = Optional; P = Prohibited. Note: It is prohibited for switches to support any commands from the Memory Device Command Set.

**Systems capable of management from Mailbox Registers and an MCTP-based CCI shall ensure that these commands are not issued as MCTP messages while a device's mailboxes are operational.

*** Indicates the minimum output payload size for a successful completion. Commands with variable output payload sizes are marked with '+'. Actual valid bytes in the output payload are indicated by the 'Payload Length' field in the Mailbox registers or in the Response packet size for MCTP-based CCIs.

8.2.9.5.1 Identify Memory Device

8.2.9.6 FM API Commands

Table 205. CXL FM API Command Opcodes

Command Set Bits[15:8]		Opcode Command Bits[7:0]			Combined Opcode			Required*				
								Type 1/2/3		Switch	Input Payload Size (B)	Output Payload Size (B)***
								Mailbox*	Type 3 MCTP			
51h	Physical Switch	00h	Identify Switch Device (Section 8.2.9.5.1.1)	5100h	P	P	MSW	0	49h			
		01h	Get Physical Port State (Section 7.6.7.1.4)	5101h	P	P	MSW	2+	14h+			
		02h	Physical Port Control (Section 7.6.7.1.5)	5102h	P	P	MSW	2	0			
		03h	Send PPB CXL.io Configuration Request (Section 7.6.7.1.6)	5103h	P	P	MSW	4+	0+			
52h	Virtual Switch	00h	Get Virtual CXL Switch Info (Section 7.6.7.2.1)	5200h	P	P	O	4+	8+			
		01h	Bind vPPB (Section 7.6.7.2.2)	5201h	P	P	O	6	0			
		02h	Unbind vPPB (Section 7.6.7.3)	5202h	P	P	O	3	0			
		03h	Generate AER Event (Section 7.6.7.3.1)	5203h	P	P	O	28h	0			
53h	MLD Port	00h	Tunnel Management Command (Section 7.6.7.4.1)	5300h	P	MO	O	4+	4+			
		01h	Send LD CXL.io Configuration Request (Section 7.6.7.4.2)	5301h	P	P	O	8+	0+			
		02h	Send LD CXL.io Memory Request (Section 7.6.7.4.3)	5302h	P	P	O	10h+	4+			
54h	MLD Components	00h	Get LD Info (Section 7.6.7.5.1)	5400h	MM	MMS	P	0	Bh			
		01h	Get LD Allocations (Section 7.6.7.5.2)	5401h	MM	MO	P	2	14h+			
		02h	Set LD Allocations (Section 7.6.7.5.3)	5402h	MO	MO	P	14h+	14h+			
		03h	Get QoS Control (Section 7.6.7.5.4)	5403h	MM	MO	P	0	7			
		04h	Set QoS Control (Section 7.6.7.5.5)	5404h	MM	MO	P	7	7			
		05h	Get QoS Status (Section 7.6.7.5.6)	5405h	MO	MO	P	0	1			
		06h	Get QoS Allocated BW (Section 7.6.7.5.7)	5406h	MM	MO	P	2	3+			
		07h	Set QoS Allocated BW (Section 7.6.7.5.8)	5407h	MM	MO	P	3+	3+			
		08h	Get QoS BW Limit (Section 7.6.7.5.9)	5408h	MM	MO	P	2	3+			
		09h	Set QoS BW Limit (Section 7.6.7.5.10)	5409h	MM	MO	P	3+	3+			

* O = Optional, P = Prohibited, MM = Mandatory for all MLD components (prohibited for SLD components), MO = Optional for MLD components (prohibited for SLD components), MMS = Mandatory for all MLD components that implement an MCTP-based CCI (prohibited for SLD components), MSW = mandatory for all switches that support the FM API MCTP message type, PSW = Prohibited for Switches, OSW = Optional for Switches, MMLD = Mandatory for all MLD components, PMLD = Prohibited for all MLD components, OMLD = Optional for all MLD components

**The FM-owned LD in MLD components is accessible using the transport defined in Section 7.6.3.1

*** Indicates the minimum output payload size for a successful completion. Commands with variable output payloads sizes are marked with '+'. Actual valid bytes in the output payload are indicated by the 'Payload Length' field in the Mailbox registers or in the Response packet size for MCTP-based CCIs.

8.2.9.7.1 Identify (Opcode 0001h)

Retrieves status information about the component, including whether it is ready to process commands. A component that is not ready to process commands shall return 'Retry Required'.

Possible Command Return Codes:

- Success
- Internal Error
- Retry Required

Command Effects:

- None

Identify Output Payload

Byte Offset	Length	Description
...		
17	1	Component Type: Indicates the type of component: <u>0 - Switch</u> <u>1 - Reserved</u> <u>2 - Reserved</u> <u>3 - Type 3 device</u> <u>All other encodings reserved</u>

9.15 Manageability Model for CXL Devices

Manageability is the set of capabilities that a managed entity exposes to a management entity. In the context of CXL, CXL devices and switches are managed entities. These capabilities are generally classified in sensors and effectors. ~~Performance counter~~An Event Log is an example of a sensor, whereas the ability to update the device firmware is an example of an effector. Sensors and effectors can either be accessed in-band, i.e., by OS/VMM resident software, or out of band, i.e., by firmware running on a management controller that is OS independent.

In-band software can access CXL device's manageability capabilities by issuing PCIe configuration read/write or MMIO read/write transactions to its Mailbox Registers. These accesses are generally mediated by CXL device driver. This is consistent with how PCIe adapters are managed.

Out of band manageability in S0 state can leverage transports for which an MCTP binding specification has been defined over PCI Express infrastructure. This assumes CXL.io path will decode and forward MCTP over PCIe VDMs in both directions. Form factors such as PCIe CEM provision two SMBUS pins (clock and data). The SMBUS path can be used for out of band manageability in Sx state or link down case. This is consistent with PCIe adapters. ~~The exact set of sensors and effectors exposed by the CXL adapter over SMBUS interface or PCIe are outside the scope of this specification. These can either be found in other specifications such as PLDM (Platform Level Data Model). CXL components may also support additional management capabilities defined in other specifications such as PLDM (Platform Level Data Model).~~

9.16 Component Command Interface

Run-time management of CXL components is facilitated by a Component Command Interface (CCI). A CCI represents a command target to process management and configuration commands issued to the component. Table 152, Table 174, and Table 205 define the commands that can be supported by a CCI.

A component can implement multiple CCIs of varying types that operate independently and have uniquely defined list of supported commands. There are 2 types of CCIs:

1. Mailbox Registers: a component can expose up to 2 mailboxes through its Mailbox Registers, as defined in Section 8.2.8.4. Each mailbox represents a unique CCI instance.

2. MCTP-based CCIs: components with MCTP-capable interconnects can expose up to 1 CCI per interconnect. There is a 1:1 relationship between the component's MCTP-based CCIs and MCTP-capable interconnects. Transfer of commands via MCTP uses the transport protocol defined in Section 7.6.3.

All CCIs shall comply with the properties listed in Section 9.16.1.

9.16.1 CCI Properties

Components implementing more than one CCI shall process commands from those CCIs in a manner which avoids starvation, so that commands submitted to one CCI do not prevent commands from other CCIs from being handled. The exact algorithm for accepting commands from multiple CCIs is implementation specific. Each CCI in a component reports its supported command list through the Command Effects Log, as described in Section 8.2.9.4.5.1.

Interface-specific properties of commands, background operation, and timeouts are defined in Section 8.2.8.4 for mailbox CCIs and in Section 9.16.2 of MCTP-based CCIs. Each CCI can support executing only one background command at a time.

When a command is successfully started as a background operation, the component shall return the Background Command Started return code defined in Section 8.2.8.4.5.1. While the command is executing in the background, the component should update the percentage complete at least once per second.

The component shall return the "Busy" return code if a command is sent to initiate a Background Operation while a Background Operation is already running on any other CCI.

IMPLEMENTATION NOTE

It is recommended that components with multiple CCIs that support commands that runs as Background Operations only advertise support for those commands on one CCI.

Each CCI within a component shall maintain a unique context with respect to the following capabilities:

- o Events, including reading contents, clearing entries, and configuring interrupt settings, and
- o Command Effects Log content

IMPLEMENTATION NOTE

Coordination between management entities attempting concurrent commands over separate CCIs that have component-level impact (e.g. FW update, etc.) is out of the scope of this specification.

9.16.2 MCTP-Based CCI Properties

The CCI command timeout is 2 seconds, measured from when the command has been received by the component to when the component has begun to transmit its response. Components should respond within this time limit, otherwise requesters may timeout. Requesters must account for round trip transmission time in addition to the command timeout.

IMPLEMENTATION NOTE

MCTP-based CCIs are intended to provide a dedicated management interface that operates independently from the state of any of the component's CXL interfaces; it is strongly recommended, but not required, that commands initiated on MCTP-based CCIs execute across Conventional Resets or any other changes of state of a component's CXL interface(s).

MCTP-based CCIs report background operation status using the "Background Operation Status" command as defined in Section 9.16.3.1.1.2.

In the event of a command timeout, the requester may re-transmit the request. New Message Tags shall be used every time a request is re-transmitted. Requesters may discard responses that arrive after the command timeout period has lapsed.

Commands sent to MCTP-based CCIs on MLD components are processed by the FM-owned LD.

12.2.3.2 Memory Error Logging and Signaling Enhancements

...

A CXL device that supports Memory Error Logging and Signaling Enhancements capability, must log such errors locally and expose the error log to system software via [its CCI\(s\)](#) (Section 8.2.8.4.3). Reading an error record from the mailbox will not automatically result in deletion of the error record on the device. An explicit clear operation is required to delete an error record from the device. To support error record access and deletion, the device shall implement the Get Event Records and Clear Event Records commands.

14.9.107-7 MCTP-Based CCI Test

Test Setup

The DUT is an MCTP-based CCI capable component connected directly to a CXL RP or downstream port of a Switch

Test Steps

1. Issue the Identify (opcode 0001h) command to the DUT as an MCTP message transmitted over an MCTP capable interconnect
2. Wait for 2s or until the DUT sends the Identify response message

Pass Criteria

The Identify response payload matches the expected value for the DUT

Fail Criteria

DUT fails to send an Identify response message within 2s or the response payload does not match the expected value for the DUT