



LEGAL NOTICE:

© **Copyright 2008 to 2023 NVM Express®, Inc. ALL RIGHTS RESERVED.**

This Technical Proposal is proprietary to the NVM Express, Inc. (also referred to as “Company”) and/or its successors and assigns.

NOTICE TO USERS WHO ARE NVM EXPRESS, INC. MEMBERS: Members of NVM Express, Inc. have the right to use and implement this Technical Proposal subject, however, to the Member’s continued compliance with the Company’s Intellectual Property Policy and Bylaws and the Member’s Participation Agreement.

NOTICE TO NON-MEMBERS OF NVM EXPRESS, INC.: If you are not a Member of NVM Express, Inc. and you have obtained a copy of this document, you only have a right to review this document or make reference to or cite this document. Any such references or citations to this document must acknowledge NVM Express, Inc. copyright ownership of this document. The proper copyright citation or reference is as follows: “© 2008 to 2023 NVM Express, Inc. ALL RIGHTS RESERVED.” When making any such citations or references 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 NVM Express, Inc. Nothing contained in this document shall be deemed as granting you any kind of license to implement or use this document or the specification described therein, or any of its contents, either expressly or impliedly, or to any intellectual property owned or controlled by NVM Express, Inc., including, without limitation, any trademarks of NVM Express, Inc.

LEGAL DISCLAIMER:

THIS DOCUMENT AND THE INFORMATION CONTAINED HEREIN IS PROVIDED ON AN “**AS IS**” BASIS. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, NVM EXPRESS, 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 NONINFRINGEMENT.

All product names, trademarks, registered trademarks, and/or servicemarks may be claimed as the property of their respective owners.

The NVM Express® design mark is a registered trademark of NVM Express, Inc.

NVM Express
c/o VTM, Inc.
3855 SW 153rd Drive
Beaverton, OR 97003
USA
info@nvmexpress.org

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant’s agreement. Copyright © 2008 to 2023 NVM Express, Inc.

NVM Express® Technical Proposal for New Feature

Technical Proposal ID	6033 - MCTP Packet Timing
Change Date	2023.04.27
Builds on Specification	NVM Express® Management Interface 1.2b
References	TP6021 Status Reporting Enhancements TP6027 Reset Behavior Clarifications

Technical Proposal Author(s)

Name	Company
Myron Loewen	Intel
Myron Loewen	Solidigm
Austin Bolen	Dell

This Technical Proposal clarifies and expands packetization constraints for NVMe-MI traffic over MCTP. This should create more consistent implementations across vendors and enable simpler Management Controller error checking and recovery. In addition, the Pause flag was simplified to one per Endpoint, the Abort primitive was cleaned up, and defined behavior for receiving new messages on a non-idle slot.

Revision History

Revision Date	Change Description
2021.08.18	Initial Draft
2021.09.13	Incorporating first round of feedback from Dell
2021.09.22	Cleaned up Sect 4.2 and CMNICS changes
2021.10.04	Incorporated all wording revisions from workgroup review of entire document, no technical changes
2021.10.11	Updated base references to NVMe-MI 1.2a, numerous wording improvements from Samsung and Kioxia feedback
2021.10.18	Changed "shall" to "should" on 40 ms max time between packets
2021.11.01	Updated feature/required/incompatible changes and added a revision reference to the obsoleted WPTT field
2021.12.06	Moved and improved text for NACK retries converting to implicit pauses. Deleted text which stated no differences from MCTP spec
2021.12.29	Incorporated feedback. Change the pause flag to be Management Endpoint specific instead of command slot specific to eliminate the historical complexity of only one command slot being paused.
2022.02.14	Made Pause Flag not Command Slot specific in Figure 42 and pulled in the new description from TP6021. Changed Pause Primitive Response to deprecate the Pause status bits since they are always value '1'. Updated legal notice.
2022.03.01	Incorporated wording feedback from Dell and Samsung
2022.03.08	Additional wordsmithing from Dell, date formats from Kioxia, and trademark fixes
2022.03.21	Deeper review from Dell and Samsung, adding some impacts from rest of spec
2022.04.11	More thorough review and update of primitives section by Austin and Myron
2022.05.16	Fixed issue revealed for unable to abort with new error message, and incorporated wordsmithing from Solidigm feedback.
2022.06.20	Reverted to Phase 2 for another exit vote on the technical changes
2022.08.12	Phase 3 wordsmithing improvements after feedback from Dell and Solidigm
2022.08.19	Improvements for referenced TPs, timing definitions, and more wordsmithing
2022.08.29	Minor wordsmithing, cleaning up more abort primitive text for idle state
2022.09.02	Wordsmithing related to More Processing Required, abort, and replay
2022.10.17	Wordsmithing to CMNICS, MPR and Replay sections plus some typo corrections
2022.12.17	Wordsmithing MPR section
2023.02.06	Wordsmithing MPR section
2023.02.22	Integrated
2023.03.10	Incorporated feedback from integration review
2023.04.27	Removed reference links

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

Description for NVMe-MI Changes Document

- **Feature Enhancements:**
 - Comprehensive MCTP packet timing constraints
 - Management Endpoint should limit interpacket bus idle delays to less than 40ms
 - Management Endpoint should resume/replay content in less than 40ms
 - Clarify MCTP messages from Management Controller not dropped due to interpacket timeout
 - Explicit procedure for receiving new command on busy command slot
 - Simplified Pause Flag to a single flag for Management Endpoint scope instead of one per Command Slot scope
 - Cleaned up the unable to abort case with a new error message
- **Required changes:**
 - Deprecate Timeout Waiting for a Packet (WPTT) status flag since it cannot be set with new procedure
- **Backwards Incompatible:**
 - Pause Control Primitive and Abort Control Primitive now affect idle Command Slots
 - Abort and implied Abort now return error status when Abort is not possible
 - No Management Endpoint receiver timeout, and hence no pausing of the receiver timeout
- **References:**
 - Technical Proposal 6033

Markup Conventions:

Black:	Unchanged (however, hot links are removed)
Red Strikethrough:	Deleted
Blue:	New
Blue Highlighted:	TBD values, anchors, and links to be inserted in new text.
Green Strikethrough:	Deleted from this location and moved to another location
Green:	Deleted from another location and moved to this location
<Green Bracketed>:	Notes to editor
Orange	Text is pulled in from a referenced Technical Proposal

Description of Specification Changes For NVM Express Management Interface Specification

Add the following definitions to section 1.8:

1.8.TBD1 Request-To-Response Time

The time it takes for a Management Endpoint to respond to a Request Message. This time is measured at the Management Endpoint from the end of the reception of a Request Message to the beginning of the transmission of the first corresponding Response Message. The request-to-response timer may be restarted under certain conditions (refer to section 4.2.1.2).

1.8.TBD2 Interpacket Time

The time a Management Endpoint takes between transmitting packets. This time is measured at the Management Endpoint from the end of the successful transmission of any packet to the beginning of the transmission of the next packet. Note that the next packet may be part of the same Response Message as the prior packet or may be part of another Response Message.

1.8.TBD3 Transmission Delay

The worst-case time it takes to transmit an NVMe-MI Message from a Requester to a Responder or vice versa. This time is measured from the end of the transmission of an NVMe-MI Message at the transmitter to the end of the reception of the NVMe-MI Message at the receiver. The Transmission Delay is dependent on the transport type and the host platform implementation.

Modify the end of Section 2.2 as follows:

2.2 SMBus/I2C

...

Clock stretching is allowed by the Management Controller, Management Endpoint, and the FRU Information Device. However, implementations are strongly discouraged from using clock stretching so that communications are more predictable with higher throughput.

~~When a NACK is received, a Management Endpoint shall follow the MCTP SMBus/I2C Transport Binding Specification for a non-bridge endpoint. The Management Endpoint treats a STOP condition due to excessive SMBus NACKs as an implicit Pause Control Primitive. Refer to section 4.2.1.1.~~

Delete Section 2.3 as follows:

~~**2.3 Error Handling**~~

~~Physical layer errors are handled as specified by the corresponding physical layer specification and MCTP transport binding specification. This specification does not require any physical layer specific error handling requirements beyond those outlined in the MCTP transport binding specifications.~~

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

Modify Figure 27 in 4.1.2 as shown and so that the **highlighted** row does not repeat on next page:

4.1.2 Response Messages

...

Figure 27: Response Message Status Values

Value	Description	Response Message Format Section
Status Values that do not indicate an error (i.e., Success Response).		
00h	Success: The command completed successfully.	4.1.2.1
01h	More Processing Required: The Command Message is in progress and requires more time to complete processing. When this Response Message Status is used in a Response Message, a subsequent Response Message contains the result of the Command Message. This Response Message Status shall not be sent more than once per Command Message, except for retransmission due to a Replay Control Primitive as described in section 4.2.1.5.	4.1.2.3
Status Values that indicate an error (i.e., Error Response).		
02h	Internal Error: The Request Message could not be processed due to a vendor specific internal error.	4.1.2.1
03h	Invalid Command Opcode: The associated command opcode field is not valid. Invalid opcodes include reserved and optional opcodes that are not implemented.	4.1.2.1
04h	Invalid Parameter: Invalid parameter field value. Request Messages received with reserved or unimplemented values in defined fields shall be completed with an Invalid Parameter Error Response. Other error conditions that result in Invalid Parameter Error Response are specified elsewhere in this specification.	4.1.2.2
05h	Invalid Command Size: The size of the Message Body of the Request Message was different than expected due to a reason other than too much or too little Request Data (e.g., the Request Message did not contain all the required parameters or Request Data was present when not expected). The expected size of the Message Body is determined by the NVMe-MI Message Type and opcode assuming no other errors are detected (e.g., Invalid Command Opcode or Invalid Parameter).	4.1.2.1
06h	Invalid Command Input Data Size: The Command Message requires Request Data and contains too much or too little Request Data.	4.1.2.1
07h	Access Denied: A Request Message was prohibited from being processed due to a vendor specific protection mechanism or the Command and Feature Lockdown feature (refer to the NVM Express Base Specification).	4.1.2.1
08h	Unable To Abort: The Abort Control Primitive is unable to abort a Command Message.	4.1.2.1
09h 08h to 1Fh	Reserved	-
20h	VPD Updates Exceeded: More updates to the VPD are attempted than allowed.	4.1.2.1
21h	PCIe Inaccessible: The PCIe functionality is not available at this time.	4.1.2.1
...	...	

4.1.2.3 More Processing Required Response

A More Processing Required Response shall be returned when the Management Endpoint requires more than the maximum Request-To-Response Time (refer to section 4.2.2.1) to complete the Processing state of the Command Message ~~as described in section 4.2.2.1~~. If a More Processing Required Response is returned, then the Management Endpoint shall start to transmit the More Processing Required Response before the maximum Request-To-Response Time is exceeded.

After sending a More Processing Required Response, the Command Slot shall return to the Process state to finish servicing the Command Message. The Response Message that is transmitted after processing completes is permitted to exceed the maximum Request-To-Response Time by the amount specified in the MPRT field (refer to Figure 32).

A More Processing Required Response shall only be transmitted once for a Command Message unless a Replay Control Primitive replays the More Processing Required Response. A Management Endpoint shall not transmit a More Processing Required Response for Control Primitive Response Messages.

The format of a More Processing Required Response is shown in Figure 31 and the response specific fields are summarized in Figure 32.

The following are examples of situations where a More Processing Required Response shall be returned, unless otherwise specified (e.g., a Firmware Commit command that results in the Firmware Activation Requires Maximum Time Violation status code):

- a Command Message is not able to be processed within the maximum Request-To-Response Time due to waiting on conditions such as the following:
 - NVM Subsystem initialization to complete (e.g., firmware initialization or hardware self-test following a reset, power on, or firmware activation without reset);
 - a resource that is not yet ready (e.g., media initialization required after power on, exiting low power mode, or reset); or
 - serialized internal queues to become free (e.g., the Command Message is not able to be processed until another Command Message completes servicing);
- the processing time of a Command Message is expected to exceed the maximum Request-To-Response Time (e.g., the Format NVM); and
- any reason other than a failure in the NVM Subsystem that is expected to cause the maximum Request-To-Response Time to be exceeded.

The Management Endpoint shall complete any steps required to be able to process the Command Message and then process the Command Message. For example, if an NVMe Admin Command targeting a Controller that is in normal operation (i.e., the value of the CSTS.SHST field is set to 00b) requires media access and media has not been initialized, then the Management Endpoint shall initialize media and then the NVMe Admin Command shall be processed.

If a Command Message could be processed successfully given sufficient time but the Management Endpoint instead returns an Error Response for any Command Message or returns an error status code in the Status field in CQEDW3 in an NVMe Admin Command Response, then the Management Controller may erroneously flag the NVM Subsystem as failed.

Figure 31: More Processing Required Response

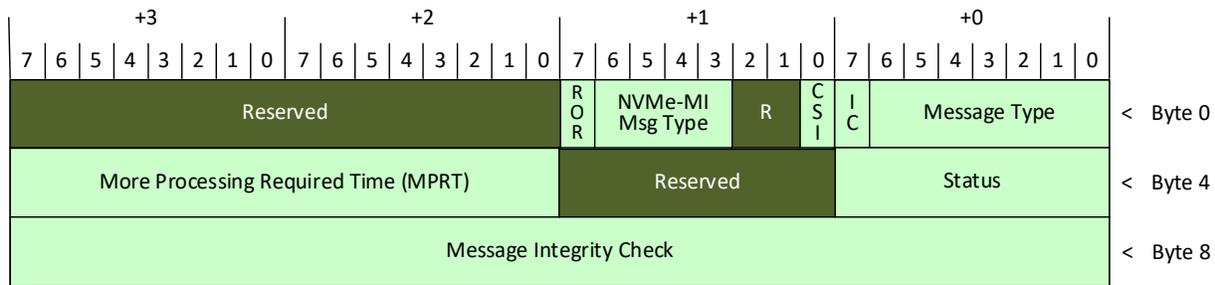


Figure 32: More Processing Required Response Fields

Byte	Description
7:6	More Processing Required Time (MPRT): This field indicates the worst-case amount of additional time in 100 ms units from the first attempt to transmit this More Processing Required Response until the first attempt to transmit the Response Message for the processed Command Message that the Management Controller should wait for the Management Endpoint to complete processing of the Command Message. A value of FFFFh in this field indicates that greater than or equal to 6,553.5 s more processing time is required.

Modify the end of Section 4.2 as follows (Modifies text from TP6027 for Idle):

4.2 Out-of-Band Message Servicing Model

...

This specification utilizes Command Slots for Command Message servicing. ~~A Management Controller should not send a new Command Message to a Command Slot until the Response Message for the previously issued Command Message to that Command Slot has been received.~~ Each Management Endpoint contains two Command Slots that each include state information **that is unique to each Command Slot** and a Pause Flag **that is shared between the two Command Slots.** The Command Slot state information and the value of the Pause Flag is returned by the Get State Primitive (refer to section 4.2.1.4).

A Management Controller sends a Command Message to a Management Endpoint that targets a specific Command Slot in the Management Endpoint. The Management Endpoint assembles MCTP packets into Command Messages separately for each Command Slot. Each Command Slot remains allocated to the Command Message until servicing of the Command Message has completed.

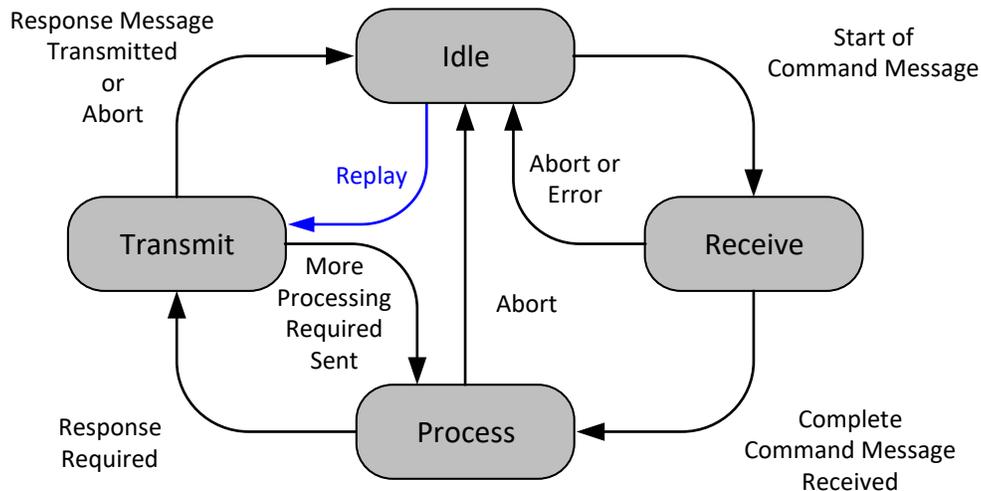
If a Command Slot that is not in the Idle state receives the start of a new Command Message, then:

- the Management Endpoint shall set the CMNICS bit to '1' (refer to Figure 42); and
- perform an implicit Abort Control Primitive (refer to section 4.2.1.3) with the exception that the Management Endpoint shall not transmit the Abort Control Primitive Response Message. If command servicing is:
 - unable to be aborted, then the Management Endpoint shall:
 - respond with an Unable To Abort Error Response;
 - silently discard the new Command Message; and
 - continue servicing the Command Message being serviced at the time the new Command Message was received;

- or
- aborted, then the Management Endpoint shall proceed with servicing the new Command Message.

...

Figure 33: Command Servicing State Diagram



Idle: A Command Slot is idle when it is not in the Receive, Process, or Transmit state. This is the initial state of the command servicing state machine following a Management Endpoint Reset (refer to section 8.3.3). Command servicing transitions from the Idle state to the Receive state when the first MCTP packet of a Command Message is received (i.e., an MCTP packet with the SOM bit in the MCTP packet header set to '1' and the Message Type set to 4h).

If a Replay Control Primitive is received and there is a Response Message available for retransmission, then command servicing shall transition to Transmit state (refer to section 4.2.1.5).

Receive: The state when the first packet of a Command Message has been received and the Command Message is being received, assembled, or validated. Command servicing transitions from Receive to the Idle state when:

- an Abort Control Primitive is received successfully;
- an error is detected in message assembly (refer to section 3.2.1.1); or
- the Message Integrity Check fails (refer to section 3.1.1.1).

Command servicing transitions from Receive state to the Process state when a Command Message is assembled and the message integrity check is successful.

Process: The state when a Command Message is processed. Processing of a Command Message consists of checking for errors with the Command Message and performing the actions specified by the Command Message or aborting the Command Message. Command servicing transitions from Process to the Transmit state when a Response Message is required to be sent. This occurs when the (i.e., processing of the Command Message has completed or command processing is expected to exceed the maximum Request-To-Response Time the corresponding MCTP transport binding specification response timeout (refer to section 4.2.2.1)). Command servicing transitions from the Process state to the Idle state due to an Abort Control Primitive unless the Command Message could not be aborted (refer to section 4.2.1.3).

Transmit: The state in which a Response Message for the Command Message is transmitted to the Management Controller. Command servicing transitions from the Transmit to the Idle state once the entire NVMe-MI Response Message associated with the response to the Command Message has been transmitted on the physical medium or due to an Abort Control Primitive (refer to section 4.2.1.3). Command servicing transitions from the Transmit to the Process state after transmitting a More Processing Required Response Message.

~~If, and only if, both the command servicing did not complete in the Process state and the Command Slot is not paused, then the Management Endpoint transmits a Response Message with status More Processing Required. If the Command Message requires more processing, then the Command Slot shall transition back to the Process state.~~

~~The behavior of receiving two or more overlapping Command Messages to the same Command Slot is undefined. If this results in the Management Endpoint discarding a Command Message, then this is considered receiving a Command Message to a non-Idle Command Slot (CMNICS). Refer to section 4.2.1.4.~~

Modify Figure 35 as follows:

4.2.1 Control Primitives

...

Figure 35: Control Primitive Fields

Bytes	Description
03:00	NVMe-MI Message Header (NMH): Refer to section 3.1.
04	Control Primitive Opcode (CPO): This field specifies the opcode of the Control Primitive to be processed. Refer to Figure 35.
05	Tag (TAG): This field specifies contains an opaque value tracking identifier that will be returned in the TAG field of the associated Control Primitive Response Message that is sent from the Management Controller in the Control Primitive and returned by the Management Endpoint in the associated Response Message. A Management Controller is allowed to use any value in this field.
07:06	Control Primitive Specific Parameter (CPSP): This field is used to pass Control Primitive specific parameter information.
11:08	Message Integrity Check (MIC): Refer to section 3.1.

Modify Figure 38 as follows:

Figure 38: Control Primitive Success Response Fields

Bytes	Description
03:00	NVMe-MI Message Header (NMH): Refer to section 3.1.
04	Status (STATUS): Refer to section 4.1.2.
05	Tag (TAG): This field shall contain a copy of the Tag specified in the TAG field of the associated Request Message contains an opaque value that is passed by the Management Endpoint from the Control Primitive to the associated Response Message. The Response Message contains the same value in this field as the corresponding Request Message.
07:06	Control Primitive Specific Response (CPSR): This field is used to return Control Primitive specific status.
11:08	Message Integrity Check (MIC): Refer to section 3.1.

Modify last paragraph of Section 4.2.1 as follows:

The ~~TAG field~~ value of the Tag field in the Control Primitive Request Message is a tracking identifier specified by the Management Controller. The Management Endpoint shall copy the value of the Tag field from the Control Primitive Request Message into the Tag field of the Control Primitive Response Message. ~~opaque value copied from the Control Primitive Request Message into the Response Message. By using unique TAG values, it is possible for the Management Controller to link Response Messages with Request Messages.~~

Modify Section 4.2.1.1 as follows:

4.2.1.1 Pause

The Pause Control Primitive ~~is used to suspend response transmission and suspend the timeout waiting for packet for both Command Slots in a Management Endpoint~~ shall set the Pause Flag to '1' and then suspend (i.e., pause) transmission of Response Messages for Command Messages on a packet boundary. Response Messages for Control Primitives shall never be paused.

It is not an error to process a Pause Control Primitive while the Pause Flag is set to '1'.

The CSI bit in a Pause Control Primitive is not used and ~~shall~~ should be cleared to 0h. If the CSI bit is set to '1', then the Management Endpoint ~~shall~~ should transmit an Invalid Parameter Error Response with the PEL field indicating the CSI bit.

~~Associated with each Command Slot is a Pause Flag that determines whether the slot is 'paused.' The Pause Flag status is included with a Success Response and may also be read using the Get State Control Primitive.~~

The CPSP field for the Pause Control Primitive is reserved.

The format of the CPSP field in the Control Primitive Success Response is shown in Figure 39.

Figure 39: Pause Control Primitive Success Response Fields

Bytes	Description	
07:06	Control Primitive Specific Response (CPSR): This field is used to return Control Primitive specific status.	
	Bits	Description
	15:02	Reserved
	01	Obsolete. Refer to NVM Express Management Interface Specification, Revision 1.2. This bit shall always be set to '1' for backwards compatibility. Pause Flag Status Slot 1 (PFSS1): This bit indicates whether or not Command Slot 1 is paused after completing the Pause Control Primitive. This bit set to '1' indicates the Command Slot is paused. This bit cleared to '0' indicates the Command Slot is not paused.
00	Obsolete. Refer to NVM Express Management Interface Specification, Revision 1.2. This bit shall always be set to '1' for backwards compatibility. Pause Flag Status Slot 0 (PFSS0): This bit indicates whether or not Command Slot 0 is paused after completing the Pause Control Primitive. This bit set to '1' indicates the Command Slot is paused. This bit cleared to '0' indicates the Command Slot is not paused.	

~~The result of a Pause Control Primitive on a Command Slot is dependent on the command servicing state of the Command Slot when the Pause Control Primitive is received, as described below:~~

~~**Idle:** The Pause Control Primitive has no effect, and the Pause Flag is not changed (i.e., remains cleared to '0'). Refer to section 4.2.1.4.~~

~~**Receive:** The Pause Control Primitive sets the Pause Flag to '1' (refer to section 4.2.1.4) and alerts the Management Endpoint that remaining MCTP packets associated with the Command Message may be delayed. Further packets sent to this Command Slot while the Pause Flag is set to '1' are received normally.~~

~~**Process:** The Pause Control Primitive sets the Pause Flag to '1' (refer to section 4.2.1.4). The Pause Flag has no effect on the command processing in the Command Slot. Upon completion of command processing, the Command Slot shall transition to the Transmit state.~~

~~**Transmit:** The Pause Control Primitive sets the Pause Flag to '1' (refer to section 4.2.1.4) suspending transmission of Response Messages on a packet boundary. The Management Endpoint should pause transmission as soon as possible after receiving a Pause Control Primitive.~~

~~The Management Endpoint shall transmit a Response Message with success status after receiving the Pause Control Primitive. It is not an error to issue a Pause Control Primitive when a Command Slot is already paused.~~

~~While the Pause Flag is set to '1', the Management Endpoint disables the timeout waiting for packet timer and does not transmit Response Messages to Command Messages. The timeout waiting for a packet is the lesser of 100 ms or the time defined in the appropriate MCTP transport binding specification. The Management Controller should not send Command Messages to a Command Slot that is paused.~~

Modify Section 4.2.1.2 as follows:

4.2.1.2 Resume

The Resume Control Primitive ~~is used to resume from a paused condition. This~~ is the complement to the Pause Control Primitive. The Resume Control Primitive shall clear the Pause Flag to '0'. After transmitting the Response Message for the Resume Control Primitive, the Management Endpoint shall resume paused transmissions from each Command Slot.

If the Pause Flag is set to '1' ~~Command Slot is paused~~ and a More Processing Required Response has not yet been transmitted for the Command Message being processed, then the request-to-response timer shall be reset and restarted (refer to section [4.2.2.1](#) for details on the Request-To-Response timer).

It is not an error to process a Resume Control Primitive while the Pause Flag is cleared to '0'.

~~Like the Pause Control Primitive, the Resume Control Primitive affects both slots and the CSI bit in a Resume Control Primitive shall be cleared to '0'. The CSI bit in a Resume Control Primitive is not used and should be cleared to '0'. If the CSI bit is set to '1', then the Management Endpoint shall respond with an Invalid Parameter Error Response with the PEL field indicating the CSI bit. If a Command Slot was not paused before receiving the Resume Control Primitive the Resume Control Primitive completes successfully and has no effect.~~

Note that the Resume Control Primitive causes a Management Endpoint to transmit the packet after the last packet the Management Endpoint transmitted prior to being paused. If the last packet transmitted was not received by the Management Controller, then, ~~after the first packet of the resumed Response Message is transmitted by the Management Endpoint,~~ the Management Controller should detect an out-of-sequence packet sequence number in the resumed Response Message and drop the Response Message. To avoid

this synchronization issue, the Management Controller should issue a Replay Controller Primitive specifying the packet number in the Response Replay Offset field from which the Response Message is replayed.

The CPSP field for the Resume Control Primitive is reserved. The CPSR field in the Control Primitive Success Response is reserved.

~~The result of a Resume Control Primitive is based on the state of a Command Slot when the Resume Control Primitive is received, as described below:~~

~~**Idle:** The Resume Control Primitive has no effect.~~

~~**Receive:** The Resume Control Primitive alerts the Management Endpoint that transmission of any remaining MCTP packets associated with the Command Message is resuming. The Pause Flag is cleared to '0' (refer to section 4.2.1.4).~~

~~**Process:** If the Command Slot is paused and a More Processing Required Response has not yet been transmitted for the Command Message being processed, then the request-to-response timer shall be reset and restarted (refer to section 4.2.2.1 for details on the request-to-response time). The Pause Flag is cleared to '0' (refer to section 4.2.1.4).~~

~~**Transmit:** The Management Endpoint resumes transmission of the Response Message corresponding to the Command Message associated with the Command Slot after responding to the Resume Control Primitive. The Pause Flag is cleared to '0' (refer to section 4.2.1.4).~~

~~The Management Endpoint shall transmit a Control Primitive Response Message with success status after receiving the Resume Control Primitive.~~

Modify Section 4.2.1.3 as follows:

4.2.1.3 Abort

The Abort Control Primitive attempts to stop Command Message servicing and return a Command Slot to the Idle state. It may not be possible for the Abort Control Primitive to abort some Command Messages (e.g., Shutdown) during the Process state.

Any Response Message associated with an aborted Command Slot shall be discarded (i.e., any Response Message associated with the Command Slot is no longer be available for replay).

~~The Abort Control Primitive is used to re-initialize a Command Slot to the Idle state, clear the Pause Flag associated with that Command Slot to '0', and attempt to abort command servicing associated with that Command Slot.~~

Attempting to abort ~~Aborting~~ a Command Message shall clear the Pause Flag to '0' even if the Command Message is unable to be aborted. ~~If have no effect on~~ the other Command Slot of the Management Endpoint, ~~other Management Endpoints, or NVMe Controllers in the NVM Subsystem. Subsequent command servicing in the Command Slot is not affected by the Abort Control Primitive.~~ has a paused transmission, then the paused transmission shall resume after the Abort Control Primitive Response Message has been transmitted.

~~A Management Controller may issue an Abort Control Primitive to clean-up resources associated with a Command Slot in an unknown state.~~

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

The CPSP field for the Abort Control Primitive is reserved. The format of the CPSR field in the Control Primitive Success Response is shown in Figure 40.

Figure 40: Abort Control Primitive Success Response Fields

Bytes	Description		
07:06	Control Primitive Specific Response (CPSR): This field is used to return Control Primitive specific status.		
	Bits	Description	
	15:02	Reserved	
	01:00	Command Processing Abort Status (CPAS): This field indicates the effect of the Abort Control Primitive on the processing of the Command Message associated with the Command Slot.	
		Value	Description
00b		Command aborted after processing completed or no command to abort.	
01b		Command aborted before processing began.	
10b	Command aborted after processing partially completed.		
11b	Reserved		

The resulting **Command Slot state and CPAS value** of the ~~an~~ Abort Control Primitive **Success Response** is based on the command servicing state of the specified Command Slot when the Abort Control Primitive is received, as described below:

Idle: ~~The Abort Control Primitive has no effect.~~ The Management Endpoint shall **respond with the transmit a Response Message with success status and the** CPAS field cleared to ~~0h~~**00b**.

Receive: The ~~Management Endpoint Command Slot shall discards the contents of the Command Slot, and~~ transitions to the Idle state. ~~The Management Endpoint shall transmit a Response Message with success status and the~~ and the Management Endpoint shall respond with the CPAS field set to ~~4h~~**01b**.

Process: Results depend on the type of Command Message that is being processed and how much processing has already completed. ~~The Abort Control Primitive causes processing of the command in the Command Slot to be aborted:~~

- a) if the ~~Abort Control Primitive was received~~ Management Endpoint is able to abort the Command Message before command processing affected the NVM Subsystem ~~started, then the Management Endpoint discards the contents of the Command Slot, and shall~~ transitions to the Idle state. ~~The Management Endpoint shall transmit a Success Response and the~~ and the Management Endpoint shall respond with the CPAS field set to ~~4h~~**01b**; ~~or~~
- b) if the ~~Abort Control Primitive was received while the command is being processed,~~ Management Endpoint is able to abort the Command Message after command processing affected the NVM Subsystem, then ~~the Management Endpoint discards the contents of the Command Slot, and shall~~ transitions to the Idle state. ~~The Management Endpoint shall attempts to abort the command:~~ and the Management Endpoint shall respond with the CPAS field set to 10b; ~~or~~
 - ~~if the command is aborted and had no effect on the NVM Subsystem, then the Management Endpoint shall transmit a Success Response and the CPAS field set to 4h; or~~
 - ~~if the Management Endpoint is not able to abort the command, then the Management Endpoint shall transmit a Success Response and set the CPAS field to 2h.~~
- c) if the Management Endpoint is unable to abort the Command Message (e.g., past the point-of-no-return in processing a Shutdown command), then the Management Endpoint

shall respond with a Response Message Status of Unable To Abort and continue processing the Command Message.

Transmit: Transmissions shall stop on a packet boundary as soon as possible.

If the Command Message has finished processing, then ~~t~~The Management Endpoint discards the contents of the Command Slot, ~~and shall transition~~s to the Idle state and the Management Endpoint shall respond with the ~~–The Management Endpoint transmits a Response Message with success status and the~~ CPAS field cleared to 0h00b.

If the Command Message has not finished processing because it is transmitting a More Processing Required Response, then the Command Slot transitions to the Process state after transmitting the More Processing Required Response and shall behave as if an Abort Control Primitive was received in the Process state.

~~It is not a Management Endpoint error if the Management Controller issues an Abort Control Primitive to a Command Slot that is paused. The state of Command Slot is reinitialized clearing the Pause Flag to '0'.~~

Modify Figure 42 in Section 4.2.1.4 as follows (modifies text from TP6021 for CMNICS):

4.2.1.4 Get State

Figure 42: Get State Control Primitive Success Response Fields

Description		
Control Primitive Specific Response (CPSR): This field is used to return Control Primitive specific status.		
Bits	Command Slot Specific ¹	Description
15	Yes No	Pause Flag (PFLG): This bit indicates whether or not the Management Endpoint Command Slot is paused. This bit set to '1' indicates If the Management Endpoint Command Slot is paused, then this bit shall be set to '1'. This bit cleared to '0' indicates If the Management Endpoint Command Slot is not paused, then this bit shall be cleared to '0'. While the Pause Flag is set, the Management Endpoint disables the timeout waiting for packet timer (refer to section 4.2.1.1) for the Command Slot and does not transmit responses to Command Messages.
14	No	NVM Subsystem Reset Occurred (NSSRO): This bit indicates when an NVM Subsystem Reset occurs while main power is applied. This bit is set to '1' if the last occurrence of an NVM Subsystem Reset occurred while main power was applied to the NVM Subsystem. This bit is cleared to '0' following a power cycle and following a Get State Control Primitive with the CESF bit set to '1'.
13	No	Bad Packet or Other Physical Layer (BPOPL): This bit is set to '1' if a packet sent to the Management Endpoint failed a transport specific packet integrity check since the last time Get State Control Primitive was processed with the CESF bit set to '1'.
12	No	Bad, Unexpected, or Expired Message Tag (BUEMT): This bit is set to '1' if the Management Endpoint detected an error of this type (refer to the MCTP Base Specification) since the last time Get State Control Primitive was processed with the CESF bit set to '1'.

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

Figure 42: Get State Control Primitive Success Response Fields

Description												
11	No	Out-of-Sequence Packet Sequence Number (OSPSN): This bit is set to '1' if the Management Endpoint detected an error of this type (refer to the MCTP Base Specification) since the last time Get State Control Primitive was processed with the CESF bit set to '1'.										
10	No	Unexpected Middle or End of Packet (UMEP): This bit is set to '1' if the Management Endpoint detected an error of this type (refer to the MCTP Base Specification) since the last time Get State Control Primitive was processed with the CESF bit set to '1'.										
09	No	Incorrect Transmission Unit (ITU): This bit is set to '1' if the Management Endpoint detected an error of this type (refer to the MCTP Base Specification) since the last time Get State Control Primitive was processed with the CESF bit set to '1'.										
08	No	Unknown Destination ID (UDSTID): This bit is set to '1' if the Management Endpoint detected an error of this type (refer to the MCTP Base Specification) since the last time Get State Control Primitive was processed with the CESF bit set to '1'.										
07	No	Bad Header Version (BHVS): This bit is set to '1' if the Management Endpoint detected an error of this type (refer to the MCTP Base Specification) since the last time Get State Control Primitive was processed with the CESF bit set to '1'.										
06	No	Unsupported Transmission Unit (UTUNT): This bit is set to '1' if the Management Endpoint detected an error of this type (refer to the MCTP Base Specification) since the last time Get State Control Primitive was processed with the CESF bit set to '1'.										
05	No	Obsolete. Refer to the NVM Express Management Interface Specification, Revision 1.2. Timeout Waiting for a Packet (WPTT): This bit is set to '1' if the Management Endpoint detected an error of this type (refer to the MCTP Base Specification) since the last time Get State Control Primitive was processed with the CESF bit set to '1'.										
04	No	Bad Message Integrity Check Error (BMICE): This bit is set to '1' if the Management Endpoint detected an error of this type (refer to the MCTP Base Specification) since the last time Get State Control Primitive was processed with the CESF bit set to '1'.										
03	No	Command Message to non-Idle Command Slot (CMNICS): If the Management Endpoint discards implicitly aborted one or more Command Messages due to receiving the first packet of a new overlapping Command Messages to a Command Slot to a Command Slot that was not in the Idle state, then this bit shall be set to '1'.										
02		Reserved										
01:00	Yes	<p>Slot Command Servicing State (SSTA): This field indicates the current command servicing state of the Command Slot. An implementation may choose to indicate only the Idle and Process states in this field. Refer to Figure 32.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0h</td> <td>Idle</td> </tr> <tr> <td>1h</td> <td>Receive</td> </tr> <tr> <td>2h</td> <td>Process</td> </tr> <tr> <td>3h</td> <td>Transmit</td> </tr> </tbody> </table>	Value	Description	0h	Idle	1h	Receive	2h	Process	3h	Transmit
Value	Description											
0h	Idle											
1h	Receive											
2h	Process											
3h	Transmit											
<p>NOTES:</p> <p>1. Command Slot Specific. A 'Yes' in this column indicates the value of the field is independent per Command Slot within a Management Endpoint. A 'No' in this column indicates the same value is reported for either Command Slot.</p>												

Modify Section 4.2.1.5 as follows, note that changes in idle state are on top of TP6027 changes:

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2023 NVM Express, Inc.

4.2.1.5 Replay

The Replay Control Primitive ~~is used to~~ shall cause the Management Endpoint to transmit a Response Message Status of Success and then shall clear the Pause Flag to '0' before retransmitting the Response Message for the last Command Message processed in ~~a the specified Command Slot and causes the Pause Flag for each Command Slot to be cleared to '0'~~. Packets within a given Response Message are transmitted in order as required by the MCTP Base Specification, but there are no packet ordering requirements between different Response Messages and so replayed packets from the specified Command Slot may be interleaved with packets from the other Command Slot. Control Primitive Response Messages shall not be replayed.

The replayed Response Message forms a new MCTP Response Message with Message Data starting from Response Replay Offset of the original Response Message and continuing to the end of the Response Message, including the original MIC. The first packet shall have SOM set to '1' and shall include the Message Header of the original Response Message even if the Response Replay Offset is not 0h. The Msg tag in each packet of the replayed Response Message shall be set to the value of the Msg tag in the associated Replay Control Primitive. Refer to the MCTP Base Specification for the definition of the Msg tag.

Note that the Management Controller requires extensions to the MCTP Base Specification in its MCTP layer in order to replay a Response Message using a non-zero Response Replay Offset. No extensions to the MCTP Base Specification are required to replay with Response Replay Offset equal to 0h. For the case where a Management Controller chooses to use a non-zero Response Replay Offset, the MCTP Base Specification requires terminating message assembly for certain errors (i.e., receiving a packet with bad packet data integrity).

If a Management Controller receives a number of packets with no errors in a Response Message and then gets an error on a packet that causes termination of message assembly, the Management Controller ~~may extend requires extensions in~~ its MCTP layer to forward the packets it received with no errors to its NVMe-MI layer prior to terminating message assembly. The Management Controller ~~can~~ may then issue a ~~R~~Replay Control Primitive to get the second part of the Response Message using a non-zero Response Replay Offset. The Management Controller's NVMe-MI layer then assembles the two partial Response Messages to create the whole Response Message. The MIC ~~can~~ may then be validated across the whole Response Message as described in section 3.1.1.1.

The format of the CPSP field in the Control Primitive Request Message is shown in Figure 43.

Figure 43: Replay Control Primitive Request Fields

Bytes	Description	
07:06	Control Primitive Specific Parameter (CPSP): This field is used to pass Control Primitive specific parameter information.	
	Bits	Description
	15:08	Reserved
07:00	Response Replay Offset (RRO): This field specifies the starting packet number from which the Response Message associated with the last Command Message processed in the Command Slot shall be replayed. This is a 0's based value. When this field is cleared to 0h, the first packet of the associated Response Message is the first packet replayed. If this field specifies an offset that is beyond the length of the Response Message, then processing of the Control Primitive is aborted and the Management Endpoint shall transmit an Invalid Parameter Error Response with the PEL field indicating this field.	

The format of the CPSR field in the Control Primitive Success Response is shown in Figure 44.

Figure 44: Replay Control Primitive Success Response Fields

Bytes	Description	
07:06	Control Primitive Specific Response (CPSR): This field is used to return Control Primitive specific status.	
	Bits	Description
	15:01	Reserved
00	Response Replay (RR): This bit indicates if a previous Response Message is to be retransmitted. This bit is set to '1' if the requested Response Message is to be retransmitted by the Management Endpoint. This bit is cleared to '0' if the requested Response Message is not retransmitted (<i>i.e., there was no Response Message to retransmit</i>).	

The result of a Replay Control Primitive is based on the command servicing state of the specified Command Slot when the Replay Control Primitive is received, as described below:

Idle: ~~The Replay Control Primitive requests retransmission of the result depends on the last Command Message processed in the specified Command Slot completion at the offset specified by the RRO field if such a completion is available as follows: Following an Abort Control Primitive or a Management Endpoint Reset (refer to section 8.3.3):~~

- a) if there is no Response Message available to retransmit (*i.e., a Replay Control Primitive is received after the Response Message has been discarded by an Abort Control Primitive or a Management Endpoint Reset has occurred but before any Command Messages are processed*), then :
 - ~~there is no Response Message available to retransmit.~~
 - ~~The~~ Management Endpoint shall transmit a Response Message with success status with the RR bit cleared to '0';

or

- b) if there is a Response Message available to retransmit (*i.e., a Replay Control Primitive is received following the processing of one or more Command Messages but before the Response Message has been discarded by an Abort Control Primitive or a Management*

Endpoint Reset has occurred), then the Management Endpoint shall ~~transmit a Response Message with success status with the RR bit set to '1'. The Management Endpoint transmits the MCTP packets associated with the requested Response Message after the Control Primitive Success Response.~~ complete the following steps in order:

1. transmit a Replay Control Primitive Response Message with the RR bit set to '1';
2. transition the Command Slot to the Transmit state; and
3. transmit a new Response Message containing the payload starting at the packet offset specified in the Response Replay Offset field of the Replay Control Primitive.

Receive: The Management Endpoint shall ~~transmits a~~ **Replay Control Primitive Success Response Message with success status** with the RR bit cleared to '0'.

Process: If a More Processing Required Response has not been transmitted for the Command Message being processed, then a **Replay Control Primitive Success Response** shall be transmitted with the RR bit cleared to '0'.

If a More Processing Required Response has been transmitted, then a **Replay Control Primitive Success Response** shall be transmitted with the RR bit set to '1' and then the More Processing Required Response shall be retransmitted. The Management Endpoint shall update the More Processing Required Time field in the Response Message with the current worst-case amount of additional time that the Management Controller should wait for the Management Endpoint to complete processing of the Command Message.

Transmit: The Management Endpoint shall complete the following steps in order:

1. stops ~~transmitting~~ ~~Response Message~~ packets for the Command Slot; ~~and then~~
2. ~~transmits a Response Message~~ **transmits a Replay Primitive Success Response** with ~~success status with~~ the RR bit set to '1'; ~~and. The Management Endpoint~~
3. ~~transmits a new~~ **transmits a new** Response Message containing the ~~packets~~ **payload** starting at the packet offset specified in the Response Replay Offset field of the Replay Control Primitive. ~~after the Control Primitive Success Response. The Command Slot remain in the Transmit state until retransmission is complete.~~

~~It is not an error to issue a Replay Control Primitive to a Command Slot that is paused. A Response Message is transmitted even if the Command Slot is paused at any time during the response, including before the first packet was transmitted. After successful completion of the Replay Control Primitive, neither Command Slot is paused (i.e., there is an implicit Resume Control Primitive affecting both Command Slots when processing the Replay Control Primitive except that the Management Endpoint shall not transmit a Response Message).~~

Modify Section 4.2.2 and its subsections as follows:

4.2.2 Out-of-Band Error Handling

This section describes **timing requirements and a packet retry mechanism** for error handling specific to the NVMe-MI out-of-band message processing model.

4.2.2.1 Command Response Message Timeouts

The timing parameters defined for NVMe-MI Messages are similar to the timing parameters for MCTP Control Messages in the MCTP Base Specification: Request-To-Response Time (refer to section

1.8.TBD1), Interpacket Time (refer to section 1.8.TBD2), and Transmission Delay (refer to section 1.8.TBD3).

The maximum Request-To-Response Time shall be 100 ms unless otherwise specified. If a Management Endpoint is paused then the request-to-response timer starts after the Pause Flag is cleared to '0'. A request-to-response timeout occurs when a Management Controller does not receive a Response Message within the Request-To-Response Time plus two times the Transmission Delay. The Management Controller should wait for at least this timeout before retrying a Request Message.

The maximum Interpacket Time shall be 40 ms while a Command Slot is in the Transmit state and the Pause Flag is cleared to '0' unless otherwise specified. An interpacket timeout occurs when a Requester does not receive a subsequent packet within the Interpacket Time plus the Transmission Delay after receiving the prior packet.

The timeouts in this section only apply while the physical transport to the Management Controller external to the NVMe Storage Device or NVMe Enclosure is available. If the physical transport is not available (e.g., busy or disconnected), then all transmissions will be delayed at least until the transport becomes available again.

Note that these timeouts assume that there is only one outstanding Response Message to be transmitted from a Management Endpoint at a time. Since there are no prioritization constraints on NVMe-MI Messages, a Response Message may be delayed until other Response Messages complete transmitting. The Management Endpoint shall not take longer than the maximum Interpacket Time to start transmitting a Response Message delayed by another Response Message. The Management Controller should adjust its timeouts to account for this added delay whenever multiple Response Messages are expected.

~~MCTP defines a maximum response time for MCTP control messages (refer to the appropriate MCTP transport binding specification).~~

~~If a Management Endpoint determines that command processing may not complete in less than or equal to the lesser of 100 ms or the request-to-response time specified in the appropriate MCTP transport binding specification the Management Endpoint shall return a More Processing Required Response as described in (section 4.1.2.3). The More Processing Required Response from the Management Endpoint is allowed to be delayed beyond this timeout while the transport is busy or unavailable.~~

~~A Management Endpoint should only return a More Processing Required Response for Command Messages that are expected take longer than the required time (e.g., Format NVM).~~

~~Implementations are strongly discouraged from using this response while processing Commands Messages that take less than or equal to the lesser of 100 ms or the request-to-response time specified in the appropriate MCTP transport binding specification.~~

4.2.2.2 Control Primitive Timeouts

~~A Management Endpoint shall attempt to respond to a Control Primitive within the lesser of 100 ms or the request-to-response time specified in the appropriate MCTP transport binding specification. The Response Message from the Management Endpoint is allowed to be delayed beyond this timeout while the transport is busy or unavailable.~~

4.2.2. **TBD5** SMBus/I2C Management Endpoint NACK and Packet Retry

A Management Endpoint that experiences a NACK anywhere during transmission shall attempt to retransmit that packet. The Management Endpoint shall treat a NACK during the ninth transmission attempt (the original transmission attempt plus 8 retry attempts) for a packet as if an implicit Pause Control Primitive was received with the exception that the Management Endpoint shall not transmit a Pause Control Primitive Response Message.

If the packet that was NACKed nine times was a Control Primitive Response Message, then it shall be discarded.