



LEGAL NOTICE:

© **Copyright 2008 to 2024 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 2024 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 2024 NVM Express, Inc.

NVM Express® Technical Proposal for New Feature

Technical Proposal ID	6037 – Modifications for I3C
Change Date	2024.08.01
Builds on Specification	NVM Express Base Specification 2.0d NVM Express® Management Interface 1.2d
References	TP6038 Management Interface Miscellaneous Maintenance TP6027b Reset Behavior Clarifications TP6034 NVMe-MI High Availability Support TP6035 Out-of-Band Asynchronous Events TP4029 Power Loss Signal Support ECN120, ECN116

Technical Proposal Author(s)

Name	Company
Yoni Shternhell	WDC
Myron Loewen	Solidigm

This Technical Proposal updates the name of the I2C/SMBus ports and communications to match the new conventions used by SNIA SFF TA's EDSFF and PCI-SIG specifications for I3C.

This proposal adds a new section on I3C with a quick summary of how I3C works, backwards compatibility with SMBus, speed negotiation, list of required/prohibited options, over which bytes the packet CRC is calculated, hot plug notifications, and ideas to make implementations more robust.

This proposal updates enumerated lists in various places for SMBus frequency to include the new higher frequencies for I3C, update how MTU size changes are made, specify retry mechanisms on errors, and how each reset types affects staying in I3C mode vs returning to power on default of SMBus.

The NVM Express Base Specification also changes for the new terminology and a capability bit.

Revision History

Revision Date	Change Description
2023.07.20	Initial Draft
2024.04.02	All Phase 2 content completed
2024.04.03	Clean version after MI meeting on Apr-08
2024-04-12	Integrating comments from Austin. Pulled text from TP6027b and ECN120. Editorial changes: <ul style="list-style-type: none">- “if this bit is set to....” convention.- Add acronym to fields name.- Add “specification” and not “spec”
2024-04-15	<ul style="list-style-type: none">- Added 2-Wire characteristics discovery to VPD: I3C Supported and I3C in Vaux.- Clean editorial text
2024-04-22	Added feedback from NVMe-MI review for Get/Set. I3C Binding rev. Fig 98 byte 11. Deleted Fig 147 row for I3C and moved its note to #1. Fixed Feature Enhancements
2024-04-24	Clean version after TWG meeting. Ready for Phase 2 exit ballot.
2024-04-30	Completed 2-Wire conversion exception table, SMBus and I3C examples in Appendix C, and feedback from Monday’s MI meeting.
2024-05-06	Integrating comments received during WG meeting
2024-05-13	Integrating comments received during WG meeting
2024-05-26	Integrating comments from Austin
2024-05-28	Added text from TP6035 that changed for I3C and prepped for final review session. Added Vendor ID to definitions to explain the case of multiple Vendor IDs
2024-05-29	Feedback from ad hoc meeting, changed the taxonomy of I3C reset to be Endpoint Reset while in I3C mode instead of MIPI’s reset signal
2024-05-30	More editorial changes in the TWG, fixed the I3C Reset text, and the incorrect Appendix C CRCs
2024-05-31	More fixes to I3C Reset text based on Austin’s feedback
2024-06-03	Clarifying section 2.2 and accept changes for 30-day member review
2024-06-26	Members review comments resolution
2024-07-11	Additional member review resolution resulting in the combination of SMBus Reset and I3C Reset sections into 2-Wire Reset with impacts on prior use of SMBus Reset
2024-07-16	Revert back 2-Wire Reset wording and figure to make the change simpler and postpone the larger change until a future ECN
2024-07-18	A few more wording clarifications from Dell in another round of feedback
2024-08-01	Fixed MIC and PEC in Appendix C example 12

Description for NVMe-MI Changes Document

- **Feature Enhancements:**
 - Adds optional communications using I3C protocol over SMBus which is now going to be called the 2-Wire port/element/mux (refer to the 2-Wire interface in the PCI Express Base Specification).
 - New bit in VPD’s NVM Subsystem Element Descriptor to indicate if MCTP over 2-Wire port is operational during the Auxiliary Power Only State.

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant’s agreement. Copyright © 2008 to 2024 NVM Express, Inc.

- Increased maximum VPD size from 4 KiB to 64 KiB.
- **Required changes:**
- **Backwards Incompatible:**
- **References:**
 - Technical Proposal 6037

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
Orange Strikethrough:	Text is deleted from a referenced Technical Proposal

Description of Specification Changes For NVM Express Base Specification 2.0d

5.1.12.1.18 Feature Identifiers Supported and Effects (Log Page Identifier 12h)

An NVM subsystem may support several interfaces for submitting a Get Log Page command such as an Admin Submission Queue, PCIe VDM Management Endpoint, or **SMBus/I2C 2-Wire** Management Endpoint (refer the NVM Express Management Interface Specification for details on Management Endpoints) and may have zero or more instances of each of those interfaces.

5.1.13.2.1 Identify Controller Data Structure (CNS 01h)

The Identify Controller data structure (refer to Figure 1) is returned to the host for the controller processing the command.

Figure 1: Identify – Identify Controller Data Structure, I/O Command Set Independent

Bytes	I/O ¹	Admin ¹	Disc ¹	Description								
Controller Capabilities and Features												
01:00	M	M	R	PCI Vendor ID (VID): Contains the company vendor identifier that is assigned by the PCI SIG. This is the same value as reported in the ID register in the PCI Header section of the NVMe over PCIe Transport Specification.								
...												
255	M	M	M	Management Endpoint Capabilities (MEC): This field indicates the support for Management Endpoints in the NVM subsystem. Refer to the NVM Express Management Interface Specification for details.								
				<table border="1"> <thead> <tr> <th>Bits</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>7:2</td> <td>Reserved</td> </tr> <tr> <td>1</td> <td>PCIe Port Management Endpoint (PCIEME): If the NVM subsystem contains one or more Management Endpoints on one or more PCIe ports, then this bit shall be set to '1'; otherwise, this bit shall be cleared to '0'.</td> </tr> <tr> <td>0</td> <td>SMBus/I2C 2-Wire Port Management Endpoint (SMBUSME TWPME): If the NVM subsystem contains one or more Management Endpoints on the SMBus/I2C 2-Wire port, then this bit shall be set to '1'; otherwise, this bit shall be cleared to '0'.</td> </tr> </tbody> </table>	Bits	Description	7:2	Reserved	1	PCIe Port Management Endpoint (PCIEME): If the NVM subsystem contains one or more Management Endpoints on one or more PCIe ports, then this bit shall be set to '1'; otherwise, this bit shall be cleared to '0'.	0	SMBus/I2C 2-Wire Port Management Endpoint (SMBUSME TWPME): If the NVM subsystem contains one or more Management Endpoints on the SMBus/I2C 2-Wire port, then this bit shall be set to '1'; otherwise, this bit shall be cleared to '0'.
				Bits	Description							
				7:2	Reserved							
1	PCIe Port Management Endpoint (PCIEME): If the NVM subsystem contains one or more Management Endpoints on one or more PCIe ports, then this bit shall be set to '1'; otherwise, this bit shall be cleared to '0'.											
0	SMBus/I2C 2-Wire Port Management Endpoint (SMBUSME TWPME): If the NVM subsystem contains one or more Management Endpoints on the SMBus/I2C 2-Wire port, then this bit shall be set to '1'; otherwise, this bit shall be cleared to '0'.											
...												

Description of Specification Changes For NVM Express Management Interface Specification

Spec wide change to 2-Wire port name, add term to definitions

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2024 NVM Express, Inc.

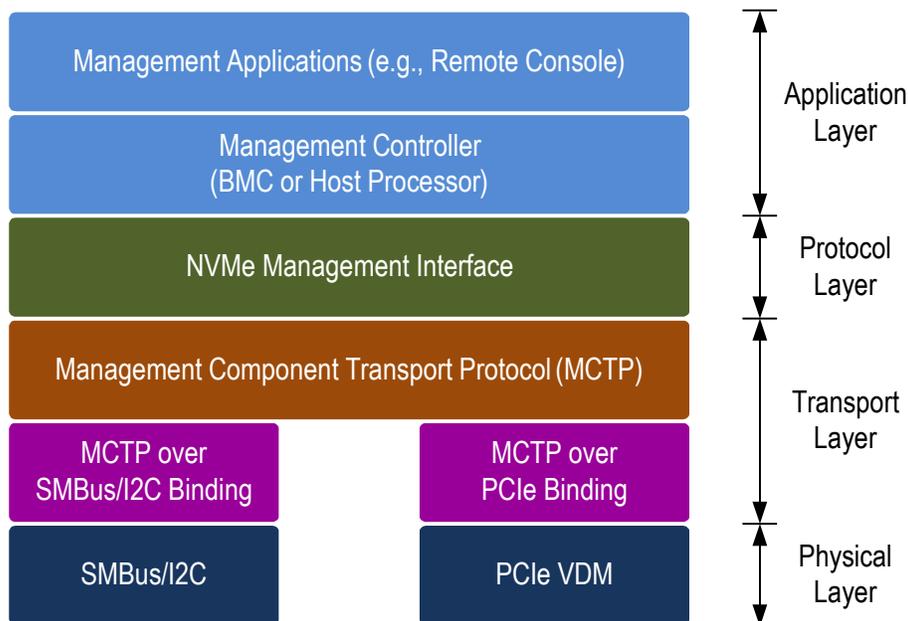
<Editor note> Replace all instances of “an SMBus/I2C” with “a 2-Wire” (about 38); and all additional instances of “SMBus/I2C” with “2-Wire”; (even the ones within figures and headings) **except** for the instances documented in the content of this technical proposal and in the following table. Replace all instances of “SMBus Reset” with “2-Wire Reset”, and “an SMBus Reset” with “a 2-Wire Reset”. Do not replace any of the other “SMBus” when not followed by “/I2C”. Note that this replacement should be applied after ECN116 is integrated and that the replacement should find any additional cases in the .next file that have been added since NVMe-MI 1.2d.

Section	Changes (most special cases keep SMBus/I2C with no change)
1.3.1.2	... either via SMBus/I2C as ...
Fig 6 heading	Dual-Port PCIe SSD with SMBus/I2C 2-Wire Port
Fig 7	SMBus/I2C With ARP
Fig 66 row 3	SMBus/I2C Frequency
8.1	... VPD SMBus/I2C access consists of...
8.2	...then the VPD shall be accessible at the SMBus/I2C address...
8.2	FRU Information Device access mechanisms (I2C Reads/I2C Writes) over SMBus/I2C .
Fig 167 byte 4	Maximum Speed: This field is set to the highest supported SMBus/I2C clock speed by the SMBus/I2C 2-Wire Mux.

1.3.1.1 Management Component Transport Protocol

The out-of-band mechanism utilizes the Management Component Transport Protocol (MCTP) as the transport and utilizes existing MCTP SMBus/I2C, I3C, and PCIe bindings for the physical layer. Command Messages are submitted to one of two Command Slots associated with a Management Endpoint contained in an NVM Subsystem. Figure 2 shows the NVMe-MI out-of-band protocol layering from the Requester’s point of view.

Figure 2: NVMe-MI Out-of-Band Protocol Layering



<Editor note> Copy the 2 SMBus/I2C blocks on the bottom left of Fig 2 and paste them into the white space on their right. Resize the new bottom 6 blocks to fit and change “SMBus/I2C” to “I3C” on the inserted blocks. Make the white space between columns a little wider than between rows.

1.5 NVMe Storage Device Architectural Model

...

An example U.2 form factor NVMe Storage Device with Expansion Connectors (i.e., a Carrier) is shown in Figure 7. This Carrier has two M.2 Expansion Connectors for connecting two M.2 NVMe Storage Device FRUs. The Carrier and each M.2 NVMe Storage Device are separate NVMe Storage Device FRUs, each with their own FRU Information Device. As defined by Figure 16, the FRU Information Device on the Carrier is at address A4h and the FRU Information Devices on each M.2 NVMe Storage Device has a default address of A6h and supports the SMBus Address Resolution Protocol (ARP). ARP is used after power is applied to reassign the conflicting A6h addresses before the M.2 FRU Information devices are read. ARP ~~would~~ ~~is~~ ~~be~~ also used to reassign the conflicting MCTP addresses and potentially additional elements. Alternatively, I3C capable elements have an I3C defined method to setup unique bus addresses after switching from SMBus to I3C mode.

1.8.TBD 2-Wire

A generalized term from the PCI-SIG PCI Express® Base Specification for the interface port that transfers compatible protocols requiring two physical wires (i.e., SMBus, I2C, and I3C).

1.8.TBD1 2-Wire Reset

A mechanism used to reset the 2-Wire elements in an NVMe Storage Device or NVMe Enclosure. For more information, see section 8.3.4.

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant’s agreement. Copyright © 2008 to 2024 NVM Express, Inc.

1.8. **TBD2** Vendor ID

An identification value assigned by PCI-SIG to the PCI-SIG member company. If an NVMe Subsystem consists of parts from multiple vendors or a Vendor has multiple Vendor IDs, then for the purpose of this specification any of the Vendor IDs may be used, but the same Vendor ID should be used for all fields that need a Vendor ID.

1.8.23 Out-of-Band

Per the Management Component Transport Protocol (MCTP) Overview White Paper, out-of-band management is management that operates with hardware resources and components that are independent of the operating system's control. The out-of-band communication paths supported by this specification are via MCTP over **2-Wire SMBus/I2C** or ~~MCTP over~~ PCIe VDM ports from a Management Controller to a Management Endpoint. In addition, this specification supports the out-of-band access mechanism defined by the IPMI Platform Management FRU Information Storage Definition specification for accessing a FRU Information Device from a Management Controller over SMBus/I2C.

...

1.8.29 **2-Wire SMBus/I2C** Mux

A bidirectional **2-Wire SMBus/I2C** fan-out multiplexer with one upstream channel and one or more downstream channels configured by an I2C command from a Management Controller to connect zero or more downstream channels to the upstream channel. Each downstream channel may be connected to devices with **2-Wire SMBus/I2C** ports. This multiplexer permits multiple devices to use the same **2-Wire SMBus/I2C** addresses if they are on separate channels. **2-Wire Mux elements may be designed to only support SMBus/I2C and such elements are unable to handle I3C traffic.**

1.8.43 **SMBus Reset**

~~A mechanism used to reset the SMBus/I2C elements in an NVMe Storage Device or NVMe Enclosure. For more information, see section 8.3.4.~~

1.11 References

...

MCTP Base Specification (DSP0236), version 1.3.1. Available from <https://www.dmtf.org>.

MCTP I3C Transport Binding Specification (DSP0233), version 1.0.1. Available from <https://www.dmtf.org>.

MCTP IDs and Codes (DSP0239), version 1.7.0. Available from <https://www.dmtf.org>.

MCTP Overview White Paper (DSP2016), version 1.0.0. Available from <https://www.dmtf.org>.

MCTP PCIe VDM Transport Binding Specification (DSP0238), version 1.1.0. Available from <https://www.dmtf.org>.

MCTP SMBus/I2C Transport Binding Specification (DSP0237), version 1.2.0. Available from <https://www.dmtf.org>.

MIPI I3C BasicSM Specification v1.1.1. Available from <https://www.mipi.org/>.

NVM Express Base Specification, revision 2.0. Available from <https://www.nvmexpress.org>.

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2024 NVM Express, Inc.

2.2 SMBus/I2C 2-Wire

This section defines the requirements for an NVMe Storage Device or NVMe Enclosure that implements an SMBus/I2C 2-Wire port. The SMBus/I2C 2-Wire physical layer is only applicable for the out-of-band mechanism.

~~If an NVMe Storage Device or NVMe Enclosure implements an NVM Subsystem with a Management Endpoint associated with an SMBus/I2C port, then that port shall comply to the MCTP SMBus/I2C Transport Binding Specification.~~

The 2-Wire port, protocols, and electricals are defined by multiple industry specifications. If an NVMe Storage Device or NVMe Enclosure implements an NVM Subsystem with a Management Endpoint associated with a 2-Wire port, then that port shall comply with the following specifications:

- PCI Express Base Specification;
- the applicable form-factor specification (e.g., U.2 or EDSFF);
- MCTP Base Specification;
- MCTP SMBus/I2C Transport Binding Specification; and
- SMBus Specification.

If the 2-Wire port also supports I3C mode, then that port shall also comply with the following specifications:

- MCTP I3C Transport Binding Specification; and
- MIPI I3C Basic Specification.

This section summarizes content from these other specifications and defines additional requirements for an NVMe Storage Device or NVMe Enclosure that implements an optional 2-Wire port.

The PCI Express Base Specification requires the 2-Wire port to default to SMBus mode after auxiliary or main power on and after specific resets. The PCI Express Base Specification also describes how to negotiate the 2-Wire Port into I3C mode for higher frequencies at lower power if supported by all elements on the bus.

An NVM Subsystem may also support the NVMe Basic Management Command for health and status polling while the 2-Wire port is in SMBus mode. The NVMe Basic Management Command is defined as an informative technical note in Appendix A, though it is not recommended for new designs.

Figure 16 lists 2-Wire SMBus/I2C elements that are supported on an NVMe Storage Device or NVMe Enclosure. For each 2-Wire SMBus/I2C element, the default SMBus/I2C address is provided as well as the conditions under which the 2-Wire SMBus/I2C element is required on an NVMe Storage Device or NVMe Enclosure. The presence or absence of Expansion Connectors on an NVMe Storage Device determines which of the two mutually exclusive SMBus/I2C addresses is used for the FRU Information Device. Using a different SMBus/I2C address for the FRU Information Device on NVMe Storage Devices that are Carriers versus non-Carriers avoids SMBus/I2C address conflict when Expansion Connectors are populated with NVMe Storage Devices.

~~ARP support on SMBus/I2C elements is optional unless multiple SMBus/I2C elements in the NVMe Storage Device or NVMe Enclosure with the same default SMBus/I2C address are present on the same SMBus/I2C channel.~~

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2024 NVM Express, Inc.

Figure 16: 2-Wire SMBus/I2C Elements and Requirements

SMBus/I2C 2-Wire Element	Default SMBus/I2C Address ³		SMBus ARP Support	I3C Support	Required Element Presence
	Hex Format	Binary Format ¹			
FRU Information Device	A6h	1010_011xb	Optional	No	Required on an NVMe Storage Device with no Expansion Connectors. Undefined on NVMe Enclosures.
FRU Information Device	A4h	1010_010xb	Optional	No	Required on Carriers (i.e., an NVMe Storage Device with one or more Expansion Connectors). Undefined on NVMe Enclosures.
SMBus/I2C 2-Wire Management Endpoint	3Ah	0011_101xb	Optional	Optional	Required if an NVMe Storage Device or NVMe Enclosure contains an one or more 2-Wire SMBus/I2C Management Endpoints at the SMBus address. May also support I3C using a dynamically assigned address.
SMBus/I2C 2-Wire Mux	E8h	1110_100xb	Optional	No	For NVMe Storage Devices, required if there is more than one SMBus/I2C element on any SMBus/I2C channel with the same SMBus/I2C address that does not support ARP. Undefined on NVMe Enclosures.
Basic Management Command ²	D4h	1101_010xb	Optional	No	For NVMe Storage Devices, not recommended for new designs. Undefined on NVMe Enclosures.
Notes:					
1. The x represents the SMBus/I2C read/write bit.					
2. The NVMe Basic Management Command is defined in Appendix A as an informative technical note.					
3. Per the PCI Express Base Specification, the SMBus/I2C addresses are not ACKed in I3C mode and I3C addresses are disabled in SMBus mode.					

Host platforms expecting to be used with one or more Management Endpoints (e.g., data center platforms and workstations) often isolate 2-Wire ports with separate channels SMBus/I2C channels to avoid address a Management Endpoint conflictings with the address of another SMBus/I2C element. An SMBus/I2C address conflicts may occur on when a Management Endpoints that does not support ARP is used with platforms that do not isolate SMBus/I2C 2-Wire ports channels (e.g., some client platforms).

SMBus-capable elements may support Address Resolution Protocol (ARP) to assign dynamic addresses for SMBus communications and eliminate address conflicts. If an NVMe Storage Device or NVMe Enclosure contains more than one 2-Wire element with the same default SMBus/I2C address that are not isolated from each other, then all such elements shall support ARP. ARP is used to dynamically reassign SMBus/I2C addresses in a system when supported by both the Management Controller and the NVMe Storage Devices or NVMe Enclosure. 2-Wire SMBus/I2C elements that support ARP should be implemented as DSA devices (refer to the SMBus Specification). These NVMe Storage dDevices should not issue “Notify ARP Master” commands.

If ARP the Get UDID command is supported by an NVM Subsystem, then all SMBus/I2C elements associated with that NVM Subsystem shall use the SMBus-Address-Resolution-Protocol Unique Device

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant’s agreement. Copyright © 2008 to 2024 NVM Express, Inc.

Identifier (UDID) shown in Figure 17. The UDID and the Get UDID command are defined by the SMBus Address Resolution Protocol (ARP) (refer to the SMBus Specification). ~~The ARP UDID is a unique identifier.~~ The UDID Type field ~~Vendor ID bits 31:30 allow~~ allows up to four 2-Wire SMBus/I2C elements to be grouped together ~~with~~ within the same NVM Subsystem if their SMBus UDID has the same Vendor ID, Device ID, and UDID Device ID. ~~Those elements shall indicate UDIDs that are identical except for their UDID Type fields.~~ The only difference within this group of UDIDs is the most significant two bits of the ~~Vendor Specific ID.~~ This fact may be used by the Management Controller to associate an SMBus/I2C Management Endpoint on the 2-Wire port with ~~it's~~ the corresponding FRU Information Device while in SMBus mode. If the 2-Wire port supports I3C mode, then the least significant 30 bits of the I3C Device ID (refer to Figure TBD) shall match the UDID Device ID. I3C uses the Device Configuration Register (DCR) to differentiate multiple device types instead of the UDID Type field. The SMBus UDID and I3C Provisioned ID shall be globally unique identifiers to prevent unreconcilable address assignment issues.

If there are multiple NVM Subsystems in an NVMe Storage Device or NVMe Enclosure that report an SMBus UDID or I3C Provisioned ID, ~~that support the Get UDID command in an SMBus ARP-capable NVMe Storage Device or NVMe Enclosure,~~ then for each NVM Subsystem the Unique NVM Storage value of the UDID Device ID fields ~~of the UDID~~ shall be sequential incremented by one for each NVM Subsystem. If the Upstream Connector has an ~~SMBus/I2C 2-Wire~~ port, then the FRU Information Device associated with that connector shall be present directly on the 2-Wire port ~~SMBus/I2C channel~~ connected to the Upstream Connector while in SMBus mode.

Clock stretching is allowed by the Management Controller, Management Endpoint, and the FRU Information Device ~~when operating in SMBus mode.~~ However, implementations are strongly discouraged from using clock stretching so that communications are more predictable with higher throughput. Clock stretching is prohibited when operating in I3C mode.

The PCI Express Base Specification describes how the 2-Wire port may be switched by the Management Controller from the default SMBus mode to I3C for higher frequencies at lower voltages. A quick summary is that only 2-Wire elements which support I3C respond with an ACK to address FCh. After this ACK, the SMBus/I2C functionality on that 2-Wire port is disabled and the Management Controller may scan for any remaining SMBus/I2C only elements. If no SMBus/I2C responders are found, the Management Controller should transmit the address FCh a second time to enter I3C only mode at the lower voltage of 1.8 V. Note that the MIPI I3C Basic Specification uses the 7-bit representation of the address (i.e., 7Eh) while this specification uses the 8-bit representation of the address (i.e., FCh). If a 2-Wire port is in I3C mode, then some 2-Wire Resets (refer to section 8.3.4) shall reset the 2-Wire port into SMBus mode.

If the 2-Wire port on an NVM Subsystem is successfully switched to I3C, then all internal elements on the NVMe Subsystem's 2-Wire port are either switched to I3C mode or disabled. Whenever the 2-Wire port enters I3C mode, the I3C elements still ACK address FCh but their own unique address is disabled. Refer to the MIPI I3C Basic Specification for Common Command Codes (CCC) like ENTDAAs which is used to assign a unique address. I3C capable NVM Subsystems shall support the ENTDAAs CCC which is used to assign dynamic addresses for I3C communications similarly to how ARP is used to assign addresses for SMBus communications.

I3C traffic has similar bit patterns to SMBus traffic. The I3C address byte tolerates collisions as SMBus does for arbitration and prioritization. However, after the address byte the bus is driven by CMOS push/pull drivers for higher frequencies. To achieve these frequencies the bus direction does not reverse for an ACK bit on every byte like SMBus. The ACK bit is renamed to T-bit for use as bit parity on writes

and as a transmission complete signal on reads. The clock is always driven by the Management Controller.

The Management Endpoint initiates traffic by asserting the data line and waiting for the Management Controller to start clocking out the Response Message or AEM. This is called an in-band interrupt (IBI) and the feature is enabled by default upon entering I3C mode. Note that because the 2-Wire port starts out in SMBus mode the Hot Join IBIs for I3C are never sent. The MCTP I3C Transport Binding Specification describes how IBIs, Private Reads, and Private Writes are used to transfer MCTP packets.

Typical I3C elements only respond to one dynamically assigned address and use a Mandatory Data Byte (MDB) to describe the traffic type. Management Endpoints shall use an MDB value of AEh and a DCR value of CCh as reserved for MCTP by MIPI. The I3C MTU shall default to 64 bytes per the MCTP I3C Transport Binding Specification. Maximum bus speed and MTU negotiation for I3C traffic is described in the MIPI I3C Basic Specification. The PCI Express Base Specification indicates the required and recommended I3C features.

When a NACK is received, a Management Endpoint shall follow the appropriate MCTP SMBus/I2C transport binding specification for the current mode of operation on the 2-Wire SMBus/I2C port. ~~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.

Figure 17: SMBus/I2C Element UDID

Bits	Field	Description								
127:120	Device Capabilities	This field describes the device capabilities.								
		<table border="1"> <thead> <tr> <th>Bits</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>7:6</td> <td>Address Type: This field describes the type of address contained in the device. Refer to the MCTP SMBus/I2C Transport Binding Specification.</td> </tr> <tr> <td>5:1</td> <td>Reserved</td> </tr> <tr> <td>0</td> <td>PEC Supported: All MCTP transactions shall include a Packet Error Code (PEC) byte. This bit shall be set to '1' to indicate support for PEC.</td> </tr> </tbody> </table>	Bits	Description	7:6	Address Type: This field describes the type of address contained in the device. Refer to the MCTP SMBus/I2C Transport Binding Specification.	5:1	Reserved	0	PEC Supported: All MCTP transactions shall include a Packet Error Code (PEC) byte. This bit shall be set to '1' to indicate support for PEC.
		Bits	Description							
		7:6	Address Type: This field describes the type of address contained in the device. Refer to the MCTP SMBus/I2C Transport Binding Specification.							
5:1	Reserved									
0	PEC Supported: All MCTP transactions shall include a Packet Error Code (PEC) byte. This bit shall be set to '1' to indicate support for PEC.									
119:112	Version and Revision	This field is used to identify the UDID version and silicon revision.								
		<table border="1"> <thead> <tr> <th>Bits</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>7:6</td> <td>Reserved</td> </tr> <tr> <td>5:3</td> <td>UDID Version: This field specifies the UDID version and shall be set to 001b.</td> </tr> <tr> <td>2:0</td> <td>Silicon Revision ID: This field is used to specify a vendor specific silicon revision level.</td> </tr> </tbody> </table>	Bits	Description	7:6	Reserved	5:3	UDID Version: This field specifies the UDID version and shall be set to 001b.	2:0	Silicon Revision ID: This field is used to specify a vendor specific silicon revision level.
		Bits	Description							
		7:6	Reserved							
5:3	UDID Version: This field specifies the UDID version and shall be set to 001b.									
2:0	Silicon Revision ID: This field is used to specify a vendor specific silicon revision level.									
111:96	Vendor ID	This field contains the PCI-SIG Vendor ID for the NVM Subsystem Management Endpoint.								
95:80	Device ID	This field contains a vendor assigned Device ID for the NVM Subsystem Management Endpoint.								

Figure 17: SMBus/I2C Element UDID

Bits	Field	Description																	
79:64	Interface	This field defines the SMBus version and the Interface Protocols supported.																	
		<table border="1"> <thead> <tr> <th>Bits</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>15:08</td> <td>Reserved</td> </tr> <tr> <td>07</td> <td>ZONE: This bit shall be cleared to '0'.</td> </tr> <tr> <td>06</td> <td>IPMI: This bit shall be cleared to '0'.</td> </tr> <tr> <td>05</td> <td>ASF: This bit shall be set to '1'. Refer to the MCTP SMBus/I2C Transport Binding Specification.</td> </tr> <tr> <td>04</td> <td>OEM: This bit shall be set to '1'.</td> </tr> <tr> <td>03:00</td> <td>SMBus Version: This field shall be set to 4h for SMBus Version 2.0, or to 5h for SMBus Version 3.0 and 3.1.</td> </tr> </tbody> </table>	Bits	Description	15:08	Reserved	07	ZONE: This bit shall be cleared to '0'.	06	IPMI: This bit shall be cleared to '0'.	05	ASF: This bit shall be set to '1'. Refer to the MCTP SMBus/I2C Transport Binding Specification.	04	OEM: This bit shall be set to '1'.	03:00	SMBus Version: This field shall be set to 4h for SMBus Version 2.0, or to 5h for SMBus Version 3.0 and 3.1.			
		Bits	Description																
		15:08	Reserved																
		07	ZONE: This bit shall be cleared to '0'.																
		06	IPMI: This bit shall be cleared to '0'.																
		05	ASF: This bit shall be set to '1'. Refer to the MCTP SMBus/I2C Transport Binding Specification.																
04	OEM: This bit shall be set to '1'.																		
03:00	SMBus Version: This field shall be set to 4h for SMBus Version 2.0, or to 5h for SMBus Version 3.0 and 3.1.																		
63:48	Subsystem Vendor ID	This field contains the PCI-SIG Vendor ID for the NVM Subsystem Management Endpoint.																	
47:32	Subsystem Device ID	This field contains a vendor assigned Device ID for the NVM Subsystem Management Endpoint.																	
31:00	Vendor Specific ID	This field ensures all UDIDs from a vendor are unique and is used to associate elements implemented within an NVMe Storage Device or NVMe Enclosure.																	
		<table border="1"> <thead> <tr> <th>Bits</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td rowspan="5">31:30</td> <td>UDID Type: This field distinguishes which NVM Subsystem that implements multiple SMBus elements is providing the UDID. Note that Management Controllers compliant implemented to versions 1.0 and earlier of this specification may be incompatible with devices NVM Subsystems using values 1h and 3h.</td> </tr> <tr> <td> <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0h</td> <td>FRU Information Device</td> </tr> <tr> <td>1h</td> <td>SMBus/I2C Mux</td> </tr> <tr> <td>2h</td> <td>Management Endpoint</td> </tr> <tr> <td>3h</td> <td>Vendor Specific Devices</td> </tr> </tbody> </table> </td> </tr> <tr> <td>29:00</td> <td>UDID Device ID: This field indicates contains a unique vendor assigned ID for the NVM Subsystem. The ID is different in each NVM Subsystem instance This field shall contain a value that results in a unique UDID as specified by the SMBus Specification, and remains static during the life of the device NVM Subsystem.</td> </tr> </tbody> </table>	Bits	Description	31:30	UDID Type: This field distinguishes which NVM Subsystem that implements multiple SMBus elements is providing the UDID. Note that Management Controllers compliant implemented to versions 1.0 and earlier of this specification may be incompatible with devices NVM Subsystems using values 1h and 3h.	<table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0h</td> <td>FRU Information Device</td> </tr> <tr> <td>1h</td> <td>SMBus/I2C Mux</td> </tr> <tr> <td>2h</td> <td>Management Endpoint</td> </tr> <tr> <td>3h</td> <td>Vendor Specific Devices</td> </tr> </tbody> </table>	Value	Description	0h	FRU Information Device	1h	SMBus/I2C Mux	2h	Management Endpoint	3h	Vendor Specific Devices	29:00	UDID Device ID: This field indicates contains a unique vendor assigned ID for the NVM Subsystem. The ID is different in each NVM Subsystem instance This field shall contain a value that results in a unique UDID as specified by the SMBus Specification, and remains static during the life of the device NVM Subsystem.
		Bits	Description																
31:30	UDID Type: This field distinguishes which NVM Subsystem that implements multiple SMBus elements is providing the UDID. Note that Management Controllers compliant implemented to versions 1.0 and earlier of this specification may be incompatible with devices NVM Subsystems using values 1h and 3h.																		
	<table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0h</td> <td>FRU Information Device</td> </tr> <tr> <td>1h</td> <td>SMBus/I2C Mux</td> </tr> <tr> <td>2h</td> <td>Management Endpoint</td> </tr> <tr> <td>3h</td> <td>Vendor Specific Devices</td> </tr> </tbody> </table>	Value	Description	0h		FRU Information Device	1h	SMBus/I2C Mux	2h	Management Endpoint	3h	Vendor Specific Devices							
	Value	Description																	
	0h	FRU Information Device																	
	1h	SMBus/I2C Mux																	
2h	Management Endpoint																		
3h	Vendor Specific Devices																		
29:00	UDID Device ID: This field indicates contains a unique vendor assigned ID for the NVM Subsystem. The ID is different in each NVM Subsystem instance This field shall contain a value that results in a unique UDID as specified by the SMBus Specification, and remains static during the life of the device NVM Subsystem.																		

Figure TBD: I3C Provisioned ID

Bits	Field	Description
47:33	MIPI Manufacturer ID	This field shall indicate the lower 15 bits of the MIPI Manufacturer ID.
32	Provisioned ID Type	This bit should be cleared to '0' to indicate that the Device ID is not a random value.
31:00	Device ID	This field shall indicate a value that results in a unique I3C Provisioned ID as specified by the MIPI I3C Basic Specification and remains static during the life of the NVM Subsystem. If SMBus ARP is also supported, then the value in the least-significant 30 bits shall match UDID Device ID.

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2024 NVM Express, Inc.

4.1.TBD4 Asynchronous Event Messages (AEMs) (Optional)

An Asynchronous Event Message (AEM) is an NVMe-MI Message that is transmitted by a Management Endpoint to a Management Controller after one or more Asynchronous Events (AEs) such as a health status change event, a temperature change event, or a firmware activation (refer to [Figure TBD10](#)) occurs. AEMs are posted (i.e., there is no NVMe-MI Message transmitted back from the Management Controller to the Management Endpoint in response to the AEM).

AEMs are permitted using the out-of-band mechanism. In-band communication uses the Asynchronous Event Request command (refer to the NVM Express Base Specification) for asynchronous events and therefore, AEMs are prohibited using the in-band tunneling mechanism. AEMs are optional for both NVMe Storage Devices and NVMe Enclosures.

AEMs are supported by Management Endpoints on a per port basis, and an implementation is permitted to support AEMs on Management Endpoints on a subset of the ports in the NVM Subsystem. Note that many host platforms are designed to connect a Management Controller to the 2-Wire SMBus/I2C Management Endpoint via an 2-Wire SMBus/I2C Mux. A Management Endpoint is not able to transmit an AEM while the 2-Wire SMBus/I2C Mux downstream channel connected to a Management Endpoint is not connected to the 2-Wire SMBus/I2C Mux upstream channel which, in turn, is connected to the Management Controller. Therefore, for example, an NVM Subsystem may choose to support AEMs on Management Endpoints on the PCIe port(s) but not on the 2-Wire SMBus/I2C port. However, host platforms that implement an I3C hub architecture resolve the problems caused by 2-Wire Mux.

If AEMs are supported on a given Management Endpoint (i.e., the Asynchronous Event Messages Supported bit is set to '1' in the Port Information data structure for the port associated with the Management Endpoint), then at least one AE shall be supported. The list of supported AEs is returned in the Response Message to a Configuration Get command for the Asynchronous Event configuration (refer to [Section 5.1.TBD5](#)).

Each AE is able to be enabled or disabled on a per Management Endpoint basis via the Configuration Set command for the Asynchronous Event configuration. The AEM servicing model is defined in [Section 4.TBD4](#).

4.TBD4.4 AEM Format

The format of an AEM is shown in [Figure TBD20](#) and the fields are described in [Figure TBD21](#).

Figure [TBD20](#): Asynchronous Event Message (AEM) Format

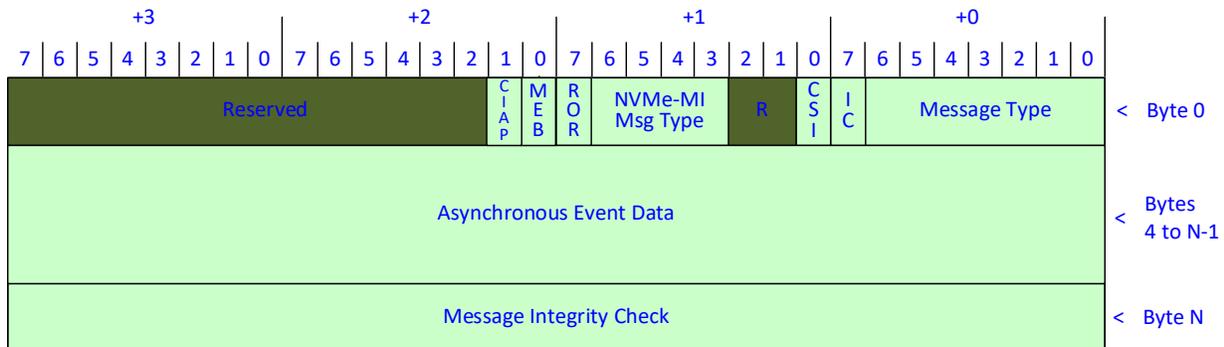


Figure TBD21: Asynchronous Event Message (AEM) Fields

Bytes	Description
3:0	NVMe-MI Message Header (NMH): Refer to Section 3.1 .
N-1:4	Asynchronous Event Data (AED): This field contains an AE Occurrence List data structure (refer to Figure TBD17).
N+3:N	Message Integrity Check (MIC): Refer to Section 3.1 .

An AEM shall be transmitted to the entity that most recently caused an AE Arm. All fields in the MCTP packet header (refer to [Figure 23](#)) of an AEM shall be set as specified by the MCTP Base Specification with the following additional requirements:

- the Msg tag shall be selected by the Management Endpoint and the Tag Owner bit shall be set to '1' since the AEM originates from the Management Endpoint; and
- the Destination Endpoint ID shall be set to the value of the Source Endpoint ID of the of the entity that most recently caused an AE Arm.

For an AEM originating from a PCIe VDM Management Endpoint, all PCIe VDM header fields shall be set as specified in the MCTP PCIe VDM Transport Binding Specification with the following additional requirements:

- the PCI Target ID field shall be set to the value of the PCI Requester ID of the entity that most recently caused an AE Arm; and
- bits 2:0 of the Type field shall be set to a value of 010b to indicate the PCIe message routing is Route by ID.

~~The values for any fields specific to an AEM originating from an SMBus/I2C Management Endpoint are outside the scope of this specification.~~ There are no special transport header field requirements to transmit an AEM from a 2-Wire Management Endpoint in either SMBus mode or I3C mode.

5.1 Configuration Get

...

Modify Figure 66 as follows:

Figure 66: NVMe Management Interface Configuration Identifiers

Configuration Identifier	Out-of-Band Mechanism O/M/P ¹	In-Band Tunneling Mechanism O/M/P ¹	Description
00h	-	-	Reserved
01h	M	P	SMBus/I2C Frequency
02h	M	M	Health Status Change
03h	M	P	MCTP Transmission Unit Size
04h	O ²	P	Asynchronous Event
05h to BFh	-	-	Reserved
C0h to FFh	O	O	Vendor Specific
Notes:			
1. O/M/P definition: O = Optional, M = Mandatory, P = Prohibited from being supported.			
2. This configuration is optional for both PCIe VDM Management Endpoints and 2-Wire SMBus/I2C Management Endpoints; however, the specifics of how this configuration works for SMBus/I2C Management Endpoints behind a 2-Wire Mux is outside the scope of this specification.			

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2024 NVM Express, Inc.

5.1.1 SMBus/I2C Frequency (Configuration Identifier 01h)

The SMBus/I2C Frequency configuration indicates the current frequency of each Management Endpoint on the SMBus port, if applicable. If the 2-Wire port is not in SMBus mode, then the indicated value is undefined.

The configuration specific fields in the NVMe Management Dword 0 field are shown in Figure 67. The configuration specific fields in the NVMe Management Dword 1 field are reserved. The current SMBus/I2C Frequency configuration is returned in the NVMe Management Response field as shown in Figure 68.

Figure 67: SMBus/I2C Frequency – NVMe Management Dword 0

Bits	Description
31:24	Port Identifier: This field specifies the port whose SMBus/I2C Frequency is indicated.
23:08	Reserved
07:00	Configuration Identifier: This field specifies the identifier of the Configuration that is being read. Refer to Figure 66.

Figure 68: SMBus/I2C Frequency – NVMe Management Response

Bits	Description												
23:04	Reserved												
03:00	<p>SMBus/I2C Frequency: This field shall indicate the frequency that the Management Endpoint transmissions are clocked at while the 2-Wire port is in SMBus mode. The current frequency of the SMBus/I2C port. The default value for this field following a reset or power cycle is 1h, if SMBus is supported.</p> <p>A Management Endpoint Reset (refer to section 8.3.3) shall set this field to 1h.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0h</td> <td>SMBus is not supported or is disabled. This value is obsolete for implementations compliant with versions of this specification later than 1.2.</td> </tr> <tr> <td>1h</td> <td>100 kHz</td> </tr> <tr> <td>2h</td> <td>400 kHz</td> </tr> <tr> <td>3h</td> <td>1 MHz</td> </tr> <tr> <td>4h to Fh</td> <td>Reserved</td> </tr> </tbody> </table>	Value	Description	0h	SMBus is not supported or is disabled. This value is obsolete for implementations compliant with versions of this specification later than 1.2.	1h	100 kHz	2h	400 kHz	3h	1 MHz	4h to Fh	Reserved
Value	Description												
0h	SMBus is not supported or is disabled. This value is obsolete for implementations compliant with versions of this specification later than 1.2.												
1h	100 kHz												
2h	400 kHz												
3h	1 MHz												
4h to Fh	Reserved												

5.1.3 MCTP Transmission Unit Size (Configuration Identifier 03h)

The MCTP Transmission Unit Size configuration indicates the current MCTP Transmission Unit Size of each Management Endpoint on the port corresponding to each Management Endpoint on the port corresponding to the Port Identifier specified in the NVMe Management Dword 0 field. If the 2-Wire port is in I3C mode, then the value indicated is determined with the SETMRL CCC mechanism defined by the MCTP I3C Transport Binding Specification.

...

5.2.1 SMBus/I2C Frequency (Configuration Identifier 01h)

The SMBus/I2C Frequency configuration specifies a new frequency for the SMBus port. [If the 2-Wire port is in I3C mode, then this command shall complete with a Success Response but have no effect.](#)

The configuration specific fields in NVMe Management Dword 0 are shown in Figure 73. The configuration specific fields in NVMe Management Dword 1 are reserved. NVMe Management Response field is reserved.

After successful completion of this command, the SMBus/I2C frequency is updated to the specified frequency. A Management Controller should not change this configuration while there are other Command Messages outstanding.

If the specified frequency is not supported, then the Management Endpoint shall respond with an Invalid Parameter Error Response with the PEL field indicating the SMBus/I2C Frequency field. If the Port Identifier specified is not [an 2-Wire SMBus/I2C](#) port, then the Management Endpoint shall respond with an Invalid Parameter Error Response with the PEL field indicating the Port Identifier field.

Figure 73: SMBus/I2C Frequency – NVMe Management Dword 0

Bits	Description		
31:24	Port Identifier: This field specifies the port whose SMBus/I2C Frequency is specified.		
23:12	Reserved		
11:08	SMBus/I2C Frequency: This field specifies the new frequency for each Management Endpoint on the specified SMBus/I2C port.		
		Value	Description
		0h	Reserved
		1h	100 kHz
		2h	400 kHz
		3h	1 MHz
4h to Fh	Reserved		
07:00	Configuration Identifier: This field specifies the identifier of the Configuration that is being written. Refer to Figure 66.		

5.2.3 MCTP Transmission Unit Size (Configuration Identifier 03h)

The MCTP Transmission Unit Size configuration specifies a new MCTP Transmission Unit Size for [each Management Endpoint on the port corresponding to](#) the specified Port Identifier, [if applicable.](#) This command is not applicable for the 2-Wire port while the port is in I3C mode. [If targeting a 2-Wire port that is in I3C mode, then:](#)

- [this command shall complete successfully but have no effect; and](#)
- [the SETMWL and SETMRL CCCs are used to change the MCTP Transmission Unit Size as defined by the MCTP I3C Transport Binding Specification.](#)

A Management Controller should check the maximum MCTP Transmission Unit Size for the port reported by the Management Endpoint using the Read NVMe-MI Data Structure command (refer to Figure 96).

The configuration specific fields in NVMe Management Dwords 0 and 1 are shown in Figure 76 and Figure 77 respectively. The NVMe Management Response field is reserved.

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2024 NVM Express, Inc.

After successful completion of this command, the MCTP Transmission Unit Size for MCTP packets on the specified port is updated to the specified size for future Command Messages. A Management Controller should not change this configuration while there are other commands outstanding. Changing this configuration while there are other Request Messages outstanding results in undefined behavior. If a Request Message is sent with a given MCTP Transmission Unit Size, then issuing a Replay Control Primitive after changing the MCTP Transmission Unit Size to a different value results in undefined behavior.

If the specified MCTP Transmission Unit Size is not supported, then the Management Endpoint shall abort the command and send a Response Message with an Invalid Parameter Error Response with the PEL field indicating the MCTP Transmission Unit Size field. If the Port Identifier specified is not valid, then the Management Endpoint shall abort the command and send a Response Message with an Invalid Parameter Error Response with the PEL field indicating the Port Identifier field.

Figure 76: MCTP Transmission Unit Size – NVMe Management Dword 0

Bits	Description
31:24	Port Identifier: This field specifies the port whose MCTP Transmission Unit Size is specified.
23:08	Reserved
07:00	Configuration Identifier: This field specifies the identifier of the Configuration that is being written. Refer to Figure 66.

Figure 77: MCTP Transmission Unit Size – NVMe Management Dword 1

Bits	Description
31:16	Reserved
15:00	MCTP Transmission Unit Size: This field contains the requested MCTP Transmission Unit Size in bytes to be used by each Management Endpoint on the port.

5.7 Read NVMe-MI Data Structure

...

Figure 96: Port Information Data Structure

Byte	Description															
00	<p>Port Type: Specifies the port type.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Definition</th> <th>Reference</th> </tr> </thead> <tbody> <tr> <td>0h</td> <td>Inactive</td> <td></td> </tr> <tr> <td>1h</td> <td>PCIe</td> <td>Figure 97</td> </tr> <tr> <td>2h</td> <td>SMBus2-Wire</td> <td>Figure 98</td> </tr> <tr> <td>3h to FFh</td> <td>Reserved</td> <td></td> </tr> </tbody> </table>	Value	Definition	Reference	0h	Inactive		1h	PCIe	Figure 97	2h	SMBus2-Wire	Figure 98	3h to FFh	Reserved	
Value	Definition	Reference														
0h	Inactive															
1h	PCIe	Figure 97														
2h	SMBus2-Wire	Figure 98														
3h to FFh	Reserved															
...	...															
03:02	<p>Maximum MCTP Transmission Unit Size: The maximum MCTP Transmission Unit size that all Management Endpoints on the port are capable of sending and receiving. If:</p> <ul style="list-style-type: none"> • if the port does not support MCTP, then this field shall be cleared to 0h; • if the Port Type is PCIe and the port supports MCTP, then this field shall be set to a value between 64 bytes and the PCIe Max Payload Size Supported (refer to the PCI Express Base Specification), inclusive. All PCIe ports within an NVM Subsystem should report the same value in this field; and • if the Port Type is 2-Wire SMBus and the port supports MCTP over SMBus, then this field shall be set to a value between 64 bytes and 250 bytes, inclusive. 															
...	...															

Figure 98: SMBus 2-Wire Port Specific Data

Bytes	Description																														
08	Current VPD SMBus/I2C Address: This field indicates the current VPD SMBus/I2C address. A value of 0h indicates there is no VPD.																														
09	<p>Maximum VPD Access SMBus/I2C Frequency: This field indicates the maximum SMBus/I2C frequency supported on the VPD interface.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Definition</th> </tr> </thead> <tbody> <tr> <td>0h</td> <td>Not supported</td> </tr> <tr> <td>1h</td> <td>100 kHz</td> </tr> <tr> <td>2h</td> <td>400 kHz</td> </tr> <tr> <td>3h</td> <td>1 MHz</td> </tr> <tr> <td>4h to FFh</td> <td>Reserved</td> </tr> </tbody> </table>	Value	Definition	0h	Not supported	1h	100 kHz	2h	400 kHz	3h	1 MHz	4h to FFh	Reserved																		
Value	Definition																														
0h	Not supported																														
1h	100 kHz																														
2h	400 kHz																														
3h	1 MHz																														
4h to FFh	Reserved																														
10	Current Management Endpoint SMBus/I2C Address: This field indicates the current 2-Wire MCTP SMBus/I2C address. A value of 0h indicates there is no Management Endpoint on this port.																														
11	<p>Maximum Management Endpoint SMBus/I2C 2-Wire Frequency Protocols Supported (2WSPRT): This field indicates which 2-Wire protocols are supported and the maximum supported SMBus/I2C frequency.</p> <table border="1"> <thead> <tr> <th>Bits</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>7</td> <td>I3C Support (I3CSPRT): If the port supports I3C mode, then this bit shall be set to '1'; otherwise, this bit shall be cleared to '0'.</td> </tr> <tr> <td>6:2</td> <td>Reserved</td> </tr> <tr> <td>1:0</td> <td> <p>Maximum SMBus/I2C Frequency (MSMBFREQ): This field shall indicate the support for SMBus/I2C and if supported, then this field shall indicate the maximum SMBus/I2C frequency supported by the Management Endpoint.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Definition</th> </tr> </thead> <tbody> <tr> <td>0h</td> <td>Not supported</td> </tr> <tr> <td>1h</td> <td>100 kHz</td> </tr> <tr> <td>2h</td> <td>400 kHz</td> </tr> <tr> <td>3h</td> <td>1 MHz</td> </tr> </tbody> </table> </td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th>Value</th> <th>Definition</th> </tr> </thead> <tbody> <tr> <td>0h</td> <td>Not supported</td> </tr> <tr> <td>1h</td> <td>100 kHz</td> </tr> <tr> <td>2h</td> <td>400 kHz</td> </tr> <tr> <td>3h</td> <td>1 MHz</td> </tr> <tr> <td>4h to FFh</td> <td>Reserved</td> </tr> </tbody> </table>	Bits	Description	7	I3C Support (I3CSPRT): If the port supports I3C mode, then this bit shall be set to '1'; otherwise, this bit shall be cleared to '0'.	6:2	Reserved	1:0	<p>Maximum SMBus/I2C Frequency (MSMBFREQ): This field shall indicate the support for SMBus/I2C and if supported, then this field shall indicate the maximum SMBus/I2C frequency supported by the Management Endpoint.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Definition</th> </tr> </thead> <tbody> <tr> <td>0h</td> <td>Not supported</td> </tr> <tr> <td>1h</td> <td>100 kHz</td> </tr> <tr> <td>2h</td> <td>400 kHz</td> </tr> <tr> <td>3h</td> <td>1 MHz</td> </tr> </tbody> </table>	Value	Definition	0h	Not supported	1h	100 kHz	2h	400 kHz	3h	1 MHz	Value	Definition	0h	Not supported	1h	100 kHz	2h	400 kHz	3h	1 MHz	4h to FFh	Reserved
Bits	Description																														
7	I3C Support (I3CSPRT): If the port supports I3C mode, then this bit shall be set to '1'; otherwise, this bit shall be cleared to '0'.																														
6:2	Reserved																														
1:0	<p>Maximum SMBus/I2C Frequency (MSMBFREQ): This field shall indicate the support for SMBus/I2C and if supported, then this field shall indicate the maximum SMBus/I2C frequency supported by the Management Endpoint.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Definition</th> </tr> </thead> <tbody> <tr> <td>0h</td> <td>Not supported</td> </tr> <tr> <td>1h</td> <td>100 kHz</td> </tr> <tr> <td>2h</td> <td>400 kHz</td> </tr> <tr> <td>3h</td> <td>1 MHz</td> </tr> </tbody> </table>	Value	Definition	0h	Not supported	1h	100 kHz	2h	400 kHz	3h	1 MHz																				
Value	Definition																														
0h	Not supported																														
1h	100 kHz																														
2h	400 kHz																														
3h	1 MHz																														
Value	Definition																														
0h	Not supported																														
1h	100 kHz																														
2h	400 kHz																														
3h	1 MHz																														
4h to FFh	Reserved																														
12	<p>NVMe Basic Management</p> <table border="1"> <thead> <tr> <th>Bits</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>7:1</td> <td>Reserved</td> </tr> <tr> <td>0</td> <td>If set to '1', then the port implements the NVMe Basic Management Command. If cleared to '0', then the port does not implement the NVMe Basic Management Command. It is strongly recommended that implementations clear this bit to '0'. The NVMe Basic Management Command is included in Appendix A for information purposes only and is not a part of the standard NVMe-MI protocol.</td> </tr> </tbody> </table>	Bits	Description	7:1	Reserved	0	If set to '1', then the port implements the NVMe Basic Management Command. If cleared to '0', then the port does not implement the NVMe Basic Management Command. It is strongly recommended that implementations clear this bit to '0'. The NVMe Basic Management Command is included in Appendix A for information purposes only and is not a part of the standard NVMe-MI protocol.																								
Bits	Description																														
7:1	Reserved																														
0	If set to '1', then the port implements the NVMe Basic Management Command. If cleared to '0', then the port does not implement the NVMe Basic Management Command. It is strongly recommended that implementations clear this bit to '0'. The NVMe Basic Management Command is included in Appendix A for information purposes only and is not a part of the standard NVMe-MI protocol.																														
31: 13	Reserved																														

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2024 NVM Express, Inc.

8.1 Out-of-Band Operational Times

< Note to editor: Header row text direction changed to horizontal. Delete the space before the superscript 2 after Main Power Only column heading >

Figure 147: Operations Supported During NVM Subsystem Power States

Operation ^{3, 4}	Powered Off -All Power Rails Off	Powered On -All Power Rails On	Auxiliary Power Only ² -Main Power Off -Auxiliary Power On	Main Power Only ² -Main Power On -Auxiliary Power Off
Main Power	Off	On	Off	On
Auxiliary Power ²	Off or Not Supported	On or Not Supported	On	Off
SMBus/I2C VPD and SMBus/I2C2-Wire Mux Access	Not Supported	Supported	Supported	Implementation Specific
SMBus/I2C2-Wire MCTP Access	Not Supported	Supported	Optional ¹	Implementation Specific
PCIe MCTP Access	Not Supported	Supported	Not Supported	Supported

Notes:

1. An implementation that supports ~~SMBus/I2C2-Wire~~ MCTP Access during Auxiliary Power may support a subset of the commands supported during the Powered On ~~this power~~ state. ~~The commands that are supported are implementation specific.~~ If a subset of commands is supported, then the subset shall include VPD Read. Power states that support MCTP over I3C shall also support MCTP over SMBus and both transports shall support the same set of commands.
2. The form factor defines whether Auxiliary power is supported. This means that Auxiliary Power Only and Main Power Only columns are not applicable to form factors that do not support ~~define~~ Auxiliary power.
3. For interactions with Power Loss Signaling processing, refer to section ~~5.3.TBD~~.
4. An ~~SMBus 2-Wire~~ Reset may prevent access as described in section 8.3.4. A PCIe Reset may prevent access as described in ~~5.3.TBD~~.
5. Firmware activation may impact access as described in section ~~5.3.2~~.

8.2 Vital Product Data

The Vital Product Data (VPD) is FRU Information (refer to the IPMI Platform Management FRU Information Storage Specification) describing an NVMe Storage Device. Each NVMe Storage Device FRU shall have a FRU Information Device with a size of 256 to ~~4,096~~65,536 bytes which contains the VPD.

...

If the NVM Subsystem has a ~~2-Wire port SMBus/I2C interface~~ and the ~~2-Wire port is in SMBus mode~~, then the VPD shall be accessible at the SMBus/I2C address of the FRU Information Device using I2C Reads. Updating the VPD using I2C Writes shall not be supported if the VPD Write command is supported. Refer to the IPMI Platform Management FRU Information Storage Definition for more information about the FRU Information Device access mechanisms (I2C Reads/I2C Writes) over SMBus/I2C.

If the NVM Subsystem has a 2-Wire port and the 2-Wire port is in I3C mode, then the FRU Information Device is not accessible using I2C Reads. In this case, the VPD is accessible with the VPD Read command if the VPD Read command is supported in the current NVM Subsystem power state.

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2024 NVM Express, Inc.

8.2.5.7 NVM Subsystem Element Descriptor

The NVM Subsystem Element Descriptor is shown in Figure 172 and is used to describe an NVM Subsystem contained in the NVMe Storage Device.

Figure 172: NVM Subsystem Element Descriptor

Bytes	Factory Default	Description						
00	07h	Type: This field indicates the type of the Element Descriptor. This field shall be set to the NVM Subsystem Element Descriptor Type (i.e., 7h). Refer to Figure 158.						
01	00h 01h	Revision: This field indicates the revision of the NVM Subsystem Element Descriptor. This field shall be cleared to 0h set to 1h.						
02	Impl Spec	Length: This field indicates the length of the NVM Subsystem Element Descriptor in bytes.						
03	3Ah or 3Bh	SMBus/I2C Address Info: If the NVM Subsystem supports an MCTP over on all SMBus/I2C Management Endpoints on the SMBus/I2C port, then this field indicates the SMBus/I2C address for the MCTP over SMBus/I2C port and whether or not SMBus ARP is supported; otherwise, this field shall be cleared to 0h.						
		<table border="1"> <thead> <tr> <th>Bits</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>7:1</td> <td>SMBus/I2C Address: This field contains the 7-bit SMBus/I2C address. Refer to Figure 16 for requirements.</td> </tr> <tr> <td>0</td> <td>ARP Capable: This bit is set to '1' If SMBus ARP is supported, then this bit shall be set to '1'; otherwise, else it this bit shall be is cleared to '0'. Refer to Figure 16 for requirements.</td> </tr> </tbody> </table>	Bits	Description	7:1	SMBus/I2C Address: This field contains the 7-bit SMBus/I2C address. Refer to Figure 16 for requirements.	0	ARP Capable: This bit is set to '1' If SMBus ARP is supported, then this bit shall be set to '1'; otherwise, else it this bit shall be is cleared to '0'. Refer to Figure 16 for requirements.
		Bits	Description					
7:1	SMBus/I2C Address: This field contains the 7-bit SMBus/I2C address. Refer to Figure 16 for requirements.							
0	ARP Capable: This bit is set to '1' If SMBus ARP is supported, then this bit shall be set to '1'; otherwise, else it this bit shall be is cleared to '0'. Refer to Figure 16 for requirements.							

Bytes	Factory Default	Description																																
04	Impl Spec	<p>SMBus/I2C2-Wire Capabilities (2WCAP): If the NVM Subsystem supports an SMBus/I2C2-Wire port then this field indicates the SMBus/I2C2-Wire capabilities; otherwise, this field shall be cleared to 0h.</p> <table border="1"> <thead> <tr> <th>Bits</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>7</td> <td> <p>Reset: This bit is set to '1' if all of the SMBus/I2C2-Wire reset mechanisms are supported as defined by the associated form factor specification. This bit is cleared to '0' if the form factor does not define SMBus2-Wire Reset or the NVMe Storage Device does not support all of the SMBus/I2C2-Wire reset mechanisms defined by the specification for the Form Factor in the Host Connector Element Descriptor.</p> </td> </tr> <tr> <td>TBD+2 TBD+1</td> <td> <p>MCTP over 2-Wire In Aux Power Support (M2WAS): This field indicates the 2-Wire port support during the Auxiliary Only power state (refer to Figure 147). This field shall not be cleared to 00b on implementations that are compliant with revisions later than 1.2 of this specification. <Note to Editor: Change "revisions later than 1.2" in the prior sentence to "revision 2.0 or later" in the integration specification.></p> <table border="1"> <thead> <tr> <th>Value</th> <th>Definition</th> </tr> </thead> <tbody> <tr> <td>00b</td> <td>Support for this capability is not indicated.</td> </tr> <tr> <td>01b</td> <td>The 2-Wire port does not support MCTP during the Auxiliary Only power state.</td> </tr> <tr> <td>10b</td> <td>The 2-Wire port supports MCTP during the Auxiliary Only power state.</td> </tr> <tr> <td>11b</td> <td>Reserved</td> </tr> </tbody> </table> </td> </tr> <tr> <td>TBD</td> <td> <p>I3C Support (I3CS): If the 2-Wire port supports MCTP during the Auxiliary Only power state (refer to Figure 147), then this bit shall be set to '1'; otherwise, this bit shall be cleared to '0'.</p> </td> </tr> <tr> <td>6TBD-1:2</td> <td>Reserved</td> </tr> <tr> <td>1:0</td> <td> <p>Maximum Speed: This field is set to the highest supported SMBus/I2C clock speed.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>100 kHz</td> </tr> <tr> <td>1</td> <td>400 kHz</td> </tr> <tr> <td>2</td> <td>1 MHz</td> </tr> <tr> <td>3</td> <td>Reserved</td> </tr> </tbody> </table> </td> </tr> </tbody> </table>	Bits	Description	7	<p>Reset: This bit is set to '1' if all of the SMBus/I2C2-Wire reset mechanisms are supported as defined by the associated form factor specification. This bit is cleared to '0' if the form factor does not define SMBus2-Wire Reset or the NVMe Storage Device does not support all of the SMBus/I2C2-Wire reset mechanisms defined by the specification for the Form Factor in the Host Connector Element Descriptor.</p>	TBD+2 TBD+1	<p>MCTP over 2-Wire In Aux Power Support (M2WAS): This field indicates the 2-Wire port support during the Auxiliary Only power state (refer to Figure 147). This field shall not be cleared to 00b on implementations that are compliant with revisions later than 1.2 of this specification. <Note to Editor: Change "revisions later than 1.2" in the prior sentence to "revision 2.0 or later" in the integration specification.></p> <table border="1"> <thead> <tr> <th>Value</th> <th>Definition</th> </tr> </thead> <tbody> <tr> <td>00b</td> <td>Support for this capability is not indicated.</td> </tr> <tr> <td>01b</td> <td>The 2-Wire port does not support MCTP during the Auxiliary Only power state.</td> </tr> <tr> <td>10b</td> <td>The 2-Wire port supports MCTP during the Auxiliary Only power state.</td> </tr> <tr> <td>11b</td> <td>Reserved</td> </tr> </tbody> </table>	Value	Definition	00b	Support for this capability is not indicated.	01b	The 2-Wire port does not support MCTP during the Auxiliary Only power state.	10b	The 2-Wire port supports MCTP during the Auxiliary Only power state.	11b	Reserved	TBD	<p>I3C Support (I3CS): If the 2-Wire port supports MCTP during the Auxiliary Only power state (refer to Figure 147), then this bit shall be set to '1'; otherwise, this bit shall be cleared to '0'.</p>	6TBD-1:2	Reserved	1:0	<p>Maximum Speed: This field is set to the highest supported SMBus/I2C clock speed.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>100 kHz</td> </tr> <tr> <td>1</td> <td>400 kHz</td> </tr> <tr> <td>2</td> <td>1 MHz</td> </tr> <tr> <td>3</td> <td>Reserved</td> </tr> </tbody> </table>	Value	Description	0	100 kHz	1	400 kHz	2	1 MHz	3	Reserved
		Bits	Description																															
		7	<p>Reset: This bit is set to '1' if all of the SMBus/I2C2-Wire reset mechanisms are supported as defined by the associated form factor specification. This bit is cleared to '0' if the form factor does not define SMBus2-Wire Reset or the NVMe Storage Device does not support all of the SMBus/I2C2-Wire reset mechanisms defined by the specification for the Form Factor in the Host Connector Element Descriptor.</p>																															
		TBD+2 TBD+1	<p>MCTP over 2-Wire In Aux Power Support (M2WAS): This field indicates the 2-Wire port support during the Auxiliary Only power state (refer to Figure 147). This field shall not be cleared to 00b on implementations that are compliant with revisions later than 1.2 of this specification. <Note to Editor: Change "revisions later than 1.2" in the prior sentence to "revision 2.0 or later" in the integration specification.></p> <table border="1"> <thead> <tr> <th>Value</th> <th>Definition</th> </tr> </thead> <tbody> <tr> <td>00b</td> <td>Support for this capability is not indicated.</td> </tr> <tr> <td>01b</td> <td>The 2-Wire port does not support MCTP during the Auxiliary Only power state.</td> </tr> <tr> <td>10b</td> <td>The 2-Wire port supports MCTP during the Auxiliary Only power state.</td> </tr> <tr> <td>11b</td> <td>Reserved</td> </tr> </tbody> </table>	Value	Definition	00b	Support for this capability is not indicated.	01b	The 2-Wire port does not support MCTP during the Auxiliary Only power state.	10b	The 2-Wire port supports MCTP during the Auxiliary Only power state.	11b	Reserved																					
		Value	Definition																															
		00b	Support for this capability is not indicated.																															
01b	The 2-Wire port does not support MCTP during the Auxiliary Only power state.																																	
10b	The 2-Wire port supports MCTP during the Auxiliary Only power state.																																	
11b	Reserved																																	
TBD	<p>I3C Support (I3CS): If the 2-Wire port supports MCTP during the Auxiliary Only power state (refer to Figure 147), then this bit shall be set to '1'; otherwise, this bit shall be cleared to '0'.</p>																																	
6TBD-1:2	Reserved																																	
1:0	<p>Maximum Speed: This field is set to the highest supported SMBus/I2C clock speed.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>100 kHz</td> </tr> <tr> <td>1</td> <td>400 kHz</td> </tr> <tr> <td>2</td> <td>1 MHz</td> </tr> <tr> <td>3</td> <td>Reserved</td> </tr> </tbody> </table>	Value	Description	0	100 kHz	1	400 kHz	2	1 MHz	3	Reserved																							
Value	Description																																	
0	100 kHz																																	
1	400 kHz																																	
2	1 MHz																																	
3	Reserved																																	

...

8.2.5.8 FRU Information Device Element Descriptor

The FRU Information Device Element Descriptor is shown in Figure 174 and is used to describe a FRU Information Device contained in the NVMe Storage Device.

Figure 174: FRU Information Device Element Descriptor

Byte Offset	Factory Default	Description																				
00	08h	Type: This field indicates the type of the Element Descriptor. This field shall be set to the FRU Information Device Element Descriptor Type (i.e., 8). Refer to Figure 158.																				
01	00h	Revision: This field indicates the revision of the FRU Information Device Element Descriptor. This field shall be cleared to 0h.																				
02	06h	Length: This field indicates the length of the FRU Information Device Element Descriptor in bytes.																				
03	A6h/A7h or 0h for NVM Storage Devices A4h/A5h or 0h for Carriers	SMBus/I2C Address Info: If the NVMe Storage Device contains an SMBus/I2C port, then this field indicates the default SMBus/I2C addressing per the table below; else, this field shall be cleared to 0h.																				
		<table border="1"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>7:1</td> <td>SMBus/I2C Address: This field contains the 7-bit SMBus/I2C address. Refer to Figure 16 for requirements.</td> </tr> <tr> <td>0</td> <td>ARP Capable: If this bit is set to '1', then SMBus ARP is supported. If this bit is cleared to '0', then SMBus ARP is not supported. Refer to the SMBus Specification for additional details.</td> </tr> </tbody> </table>	Bit	Description	7:1	SMBus/I2C Address: This field contains the 7-bit SMBus/I2C address. Refer to Figure 16 for requirements.	0	ARP Capable: If this bit is set to '1', then SMBus ARP is supported. If this bit is cleared to '0', then SMBus ARP is not supported. Refer to the SMBus Specification for additional details.														
		Bit	Description																			
7:1	SMBus/I2C Address: This field contains the 7-bit SMBus/I2C address. Refer to Figure 16 for requirements.																					
0	ARP Capable: If this bit is set to '1', then SMBus ARP is supported. If this bit is cleared to '0', then SMBus ARP is not supported. Refer to the SMBus Specification for additional details.																					
04	Impl Spec	SMBus/I2C Capabilities: If the NVMe Storage Device contains an SMBus/I2C port, then this field indicates the SMBus/I2C capabilities per the table below; else, this field shall be cleared to 0h.																				
		<table border="1"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>7</td> <td>Reset: If this bit is set to '1', then all of the SMBus/I2C reset mechanisms are supported as defined by the specification for the Form Factor in the Host Connector Element Descriptor. If this bit is cleared to '0', then the FRU Information Device does not support all of the SMBus/I2C reset mechanisms defined by the specification for the Form Factor in the Host Connector Element Descriptor.</td> </tr> <tr> <td>6</td> <td>I2C Writes Allowed: If this bit is set to '1', then the FRU Information Device is allowed to be written using an I2C Write operation. If this bit is cleared to '0', then the FRU Information Device is not allowed to be written using an I2C Write operation.</td> </tr> <tr> <td>5:2</td> <td>Reserved</td> </tr> <tr> <td>1:0</td> <td>Maximum Speed: This field is set to the highest supported SMBus/I2C clock speed supported by the FRU Information Device. <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>100 kHz</td> </tr> <tr> <td>1</td> <td>400 kHz</td> </tr> <tr> <td>2</td> <td>1 MHz</td> </tr> <tr> <td>3</td> <td>Reserved</td> </tr> </tbody> </table> </td> </tr> </tbody> </table>	Bit	Description	7	Reset: If this bit is set to '1', then all of the SMBus/I2C reset mechanisms are supported as defined by the specification for the Form Factor in the Host Connector Element Descriptor. If this bit is cleared to '0', then the FRU Information Device does not support all of the SMBus/I2C reset mechanisms defined by the specification for the Form Factor in the Host Connector Element Descriptor.	6	I2C Writes Allowed: If this bit is set to '1', then the FRU Information Device is allowed to be written using an I2C Write operation. If this bit is cleared to '0', then the FRU Information Device is not allowed to be written using an I2C Write operation.	5:2	Reserved	1:0	Maximum Speed: This field is set to the highest supported SMBus/I2C clock speed supported by the FRU Information Device. <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>100 kHz</td> </tr> <tr> <td>1</td> <td>400 kHz</td> </tr> <tr> <td>2</td> <td>1 MHz</td> </tr> <tr> <td>3</td> <td>Reserved</td> </tr> </tbody> </table>	Value	Description	0	100 kHz	1	400 kHz	2	1 MHz	3	Reserved
		Bit	Description																			
		7	Reset: If this bit is set to '1', then all of the SMBus/I2C reset mechanisms are supported as defined by the specification for the Form Factor in the Host Connector Element Descriptor. If this bit is cleared to '0', then the FRU Information Device does not support all of the SMBus/I2C reset mechanisms defined by the specification for the Form Factor in the Host Connector Element Descriptor.																			
		6	I2C Writes Allowed: If this bit is set to '1', then the FRU Information Device is allowed to be written using an I2C Write operation. If this bit is cleared to '0', then the FRU Information Device is not allowed to be written using an I2C Write operation.																			
5:2	Reserved																					
1:0	Maximum Speed: This field is set to the highest supported SMBus/I2C clock speed supported by the FRU Information Device. <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>100 kHz</td> </tr> <tr> <td>1</td> <td>400 kHz</td> </tr> <tr> <td>2</td> <td>1 MHz</td> </tr> <tr> <td>3</td> <td>Reserved</td> </tr> </tbody> </table>	Value	Description	0	100 kHz	1	400 kHz	2	1 MHz	3	Reserved											
Value	Description																					
0	100 kHz																					
1	400 kHz																					
2	1 MHz																					
3	Reserved																					
05	8h to 0Ch 10h inclusive	Maximum FRU Information Device Size: The maximum size of the FRU Information Device is 2 ^N bytes where N is the value in this field (e.g., a value of 8 in this field indicates a maximum FRU Information Device size of 2 ⁸ or 256 bytes).																				

8.3.4 2-Wire SMBus Resets

The 2-Wire port may be reset in multiple ways to restore 2-Wire communications. Some 2-Wire Resets are only applicable for specific 2-Wire modes or device form factors. Some 2-Wire Resets also reset the associated 2-Wire Management Endpoint.

SMBus eClock-low recovery is the ability to reset communication on all 2-Wire SMBus/I2C Management Endpoints on an SMBus/I2C 2-Wire port when the 2-Wire SMBus/I2C clock on that 2-Wire SMBus/I2C

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2024 NVM Express, Inc.

port is low for longer than $t_{\text{TIMEOUT,MIN}}$ (refer to the SMBus Specification). 2-Wire SMBus/I2C Management Endpoints shall support SMBus-clock-low recovery. It is strongly recommended that any 2-Wire SMBus/I2C element other than the 2-Wire SMBus/I2C Management Endpoint (refer to Figure 16 for a list of 2-Wire SMBus/I2C elements) should support SMBus-clock-low recovery. SMBus-clock-low recovery:

- shall cause an SMBus Management Endpoint Reset; ~~An SMBus Reset caused by SMBus clock-low recovery~~
- shall not cause reset ARP-assigned addresses to be reset to their default values; and
- shall switch the 2-Wire port to SMBus mode if the 2-Wire port was in I3C mode.

Some form factor specifications may also specify one or more form factor-specific mechanisms to reset the 2-Wire port (e.g., SMBRST# as defined in the SNIA SFF-TA-1009 Enterprise and Datacenter Standard Form Factor Pin and Signal Specification, or the rising edge of the +3.3 Vaux rail as defined in the PCI Express SFF-8639 Module Specification). Any form factor-specific mechanisms to reset the 2-Wire port SMBus supported by an NVMe Storage Device or NVMe Enclosure shall cause:

- a Management Endpoint ~~an SMBus~~ Reset; and
- shall switch the 2-Wire port to SMBus mode if the 2-Wire port was in I3C mode.

An NVM Subsystem Reset also causes a 2-Wire Reset and a Management Endpoint Reset. If the NVM Subsystem Reset is due to application of main power, then the 2-Wire port shall switch to SMBus mode; otherwise, the NVM Subsystem Reset shall not change the 2-Wire port's current SMBus or I3C mode.

The Target Reset Pattern as defined by the MIPI I3C Basic Specification may also cause a 2-Wire Reset for 2-Wire ports in I3C mode. The default case for the Target Reset Pattern shall reset the I3C physical layer. The default case shall not cause a Management Endpoint Reset and shall not reset the I3C Dynamically Assigned Address. Target Reset Patterns shall not change the 2-Wire port's current SMBus or I3C mode.

2-Wire ports in I3C mode may optionally support the RSTACT CCC as defined by the MIPI I3C Basic Specification. RSTACT uses Defining Bytes to modify the behavior of the immediately subsequent Target Reset Pattern as follows:

- Defining Byte value 0 shall not cause a 2-Wire Reset;
- Defining Byte value 1 shall behave the same as a default Target Reset Pattern; and
- Defining Byte value 2 shall cause a Management Endpoint Reset and shall reset the I3C Dynamically Assigned Address.

If a 2-Wire Reset causes a Management Endpoint Reset, then the 2-Wire physical layer shall also be reset. ~~An SMBus Reset shall cause a Management Endpoint Reset of the all SMBus/I2C Management Endpoints on the SMBus/I2C port.~~ For any 2-Wire SMBus/I2C element other than the SMBus/I2C Management Endpoint (refer to Figure 16 for a list of 2-Wire SMBus/I2C elements), it is strongly recommended that an SMBus-2-Wire Reset should reset that 2-Wire SMBus/I2C element.

An SMBus-2-Wire Reset shall cause an SMBus-2-Wire reset mechanism defined for the Expansion Connector to be applied to each Expansion Connector in the NVMe Storage Device.

If an 2-Wire SMBus/I2C Management Endpoint is transmitting a Response Message, then an 2-Wire SMBus Reset shall cause the 2-Wire SMBus/I2C port to attempt to generate a STOP condition (refer to the SMBus Specification) within 5 ms from the assertion of a 2-Wire SMBus Reset. The 2-Wire SMBus/I2C port's transmitter shall remain in the bus idle condition (refer to the SMBus Specification and MIPI I3C Basic Specification) and the 2-Wire port's receiver shall ignore any incoming traffic for the remainder of the 2-Wire SMBus/I2C Reset assertion ~~even if a Management Controller attempts to access the SMBus/I2C port.~~ An 2-Wire SMBus/I2C port shall support 2-Wire SMBus/I2C accesses starting from the de-assertion of a 2-Wire SMBus Reset within the same timing constraints as are applicable to transitioning from an unsupported to a supported power state as defined in section 8.1.

Additional requirements and recommendations for 2-Wire SMBus Resets are specified elsewhere in this specification. For example, an 2-Wire SMBus Reset:

- resets the value of the SMBus/I2C Frequency field as defined in Figure 76; and
- should clear the internal Command Offset for the VPD to 0h as defined in section 8.2.

Appendix A Technical Note: NVM Express Basic Management Command

This appendix describes the NVMe Basic Management Command and is included here for informational purposes only. The NVMe Basic Management Command is not formally a part of this specification and its features are not tested by the NVMe Compliance program. No further enhancements to the NVMe Basic Management Command are planned, and it is strongly recommended that any consumers of the NVMe Basic Management Command transition to using the standard NVMe-MI protocol.

This specification utilizes Management Component Transport Protocol (MCTP) messages. The NVMe Basic Management Command does not utilize MCTP. Support for the NVMe Basic Management Command is optional. [The NVMe Basic Management Command only works while the 2-Wire port is in SMBus mode.](#)

...

<Note to editor: Capitalize Vendor ID in the command code 8 row of Figure 176>

8	08	Length of identification: Indicates number of additional bytes to read before encountering PEC. This value should always be 22 (16h) in implementations of this version of the spec.
	10:09	Vendor ID: The 2 byte V endor ID, assigned by the PCI-SIG. Should match VID in the Identify Controller command response. Note the MSB is transmitted first.
	11:30	Serial Number: 20 characters that match the serial number in the NVMe Identify Controller command response. Note the first character is transmitted first.
	31	PEC: An 8 bit CRC calculated over the SMBus address, command code, second SMBus address, and returned data. The algorithm is defined in the SMBus Specification.

Appendix C Example NVMe-MI Messages ~~over SMBus/I2C~~

This section contains example NVMe-MI Messages ~~over SMBus/I2C~~ between a Management Controller ...

< Note to editor: Append the rest of the text in this TP onto the end of Appendix C. The indented examples in this text are the correct final colors and formatting, they do not follow the color scheme of the rest of the technical proposal.>

If each Request Message fits into a single packet, then the Management Controller's MCTP stack may be simplified to transmit fixed strings that are pre-coded with the correct headers and MIC. Receiving Response Messages also does not require a full MCTP stack as the Management Controller may ignore the Pkt Seq # field for single packet Response Messages but should compute CRCs to test the Response Message's PEC and MIC fields. Underlined bytes in the example Response Messages change with Pkt Seq # and thus may not match the example. Bold italic shadows mark bytes in the example Response Messages that change if an Error Response is returned.

The rest of this appendix follows the same color scheme for headers and footers as in the prior examples but byte labeling was eliminated to for easier copying of the byte strings into test tools. Each Request Message is a single packet and each Response Messages is a single packet. If implementations do not use a Management Controller's physical address of 20h (21h in the command string), then it should be updated as well as the PEC field. Some implementations may also need Port Identifier to be a value other than 0. Online tools like <https://www.crccalc.com> may be used to calculate the PEC and MIC fields.

Example 7: Set SMBus packet size to 250 bytes to make most Response Messages single packet:

3Ah 0Fh 19h 21h 01h 00h 00h 8Bh 84h 08h 00h 00h 03h 00h 00h 00h 03h 00h 00h 00h FAh 00h 00h 00h D2h 88h B4h 01h 0Ch

Successful Response:

20h 0Fh 11h 3Bh 01h 00h 00h D3h 84h 88h 00h 00h **00h 00h 00h 00h 24h 55h 77h 22h 21h**

Example 8: VPD Read command to fetch first 232 bytes of VPD on MCTP over SMBus (same data as I2C reads from offset 0):

3Ah 0Fh 19h 21h 01h 00h 00h 8Bh 84h 10h 00h 00h 05h 00h 00h 00h 00h 00h 00h 00h F0h 00h 00h 00h 51h 22h 32h 62h 9Ah

Successful Response (the 232 bytes of VPD are shown with '...' and CRCs are marked with 'xxh' since they depend on VPD content):

20h 0Fh **F9h** 3Bh 01h 00h 00h D3h 84h 88h 00h 00h **00h 00h 00h 00h ... xxh xxh xxh xxh xxh**

Example 9: Read NVMe-MI Data Structure command on MCTP over SMBus to retrieve data similar to what is returned by NVMe Basic Management Command:

3Ah 0Fh 19h 21h 01h 00h 00h 8Bh 84h 10h 00h 00h 02h 00h AFh 3Dh 20h 33h 41h

Successful Response (the 8 bytes of NSHDS are shown with '...' and CRCs are marked with 'xxh' since they depend on NSDHDS content, the first 4 bytes of NSHDS match Basic data):

20h 0Fh **19h** 3Bh 01h 00h 00h D3h 84h 88h 00h 00h **00h 00h 00h 00h ... xxh xxh xxh xxh xxh**

Technical input submitted to NVM Express® is subject to the terms of the NVM Express® Participant's agreement. Copyright © 2008 to 2024 NVM Express, Inc.

Refer to the PCI Express Base Specification for the algorithm to switch the 2-Wire port from SMBus mode into I3C mode. The next 4 examples show how the prior 3 examples are done in I3C mode. In I3C the clock signal is always driven by the Management Controller so there is no need to negotiate the bus frequency but a Management Endpoint may advertise the max timings supported.

Example 10: Assign address 30h to the I3C element using the ENTDAACCC (07h). The NVMe Storage vendor's MIPI Manufacturer ID is 1234h and the NVMe Storage Device's unique Device Id is 01020304h. The BCR for this element is 06h and the DCR is CCh for all MCTP elements. Red bytes are MIPI CCCs from the Management Controller with orange responses from the Management Endpoint. A 7Eh read becomes FCh, a 7Eh write becomes FDh, the ENTDAACCC is 07h, and a repeated start is indicated with 'Sr'. Interrupts are enabled by default so the NVMe Storage Device with an address is now ready to send and receive MCTP packets over I3C. Changes for error conditions are similar to MCTP over SMBus but not indicated for I3C.

FCh 07h Sr FDh 24h 68h 01h 02h 03h 04h 06h CCh 60h

Example 11: Configure the I3C device at address 30h to receive and transmit packets of 250 (FAh) bytes. Follow up reads are used to confirm that both settings were accepted. Direct CCCs for SETMWL=89h, SETMRL=8Ah, GETMWL=8B, and GETMRL=8Ch:

FCh 89h Sr 60h 00h FAh
FCh 8Ah Sr 60h 00h FAh
FCh 8Bh Sr 61h 00h FAh
FCh 8Ch Sr 61h 00h FAh

Example 12: VPD Read command to fetch first 8 bytes of VPD on MCTP over I3C (smaller read used to show CRC values):

60h 01h 00h 00h 8Bh 84h 10h 00h 00h 05h 00h 00h 00h 00h 00h 00h 08h 00h 00h 00h D9h
72h 31h 53h D3h

IBI notification that an MCTP packet is ready for Management Controller to read:
61h AEh

Successful Response privately read by Management Controller (VPD is last 8 bytes in black font):
61h 01h 00h 00h D3h 84h 88h 00h 00h 00h 00h 00h 01h 00h 00h 00h 01h 0Bh 00h F3h E9h
1Fh 3Dh 8Bh EFh

Example 13: Read NVMe-MI Data Structure command on MCTP over I3C to retrieve data similar to what is returned by NVMe Basic Management Command:

60h 01h 00h 00h 8Bh 84h 10h 00h 00h 02h 00h AFh
3Dh 20h 33h A7h

IBI notification that an MCTP packet is ready for Management Controller to read:
61h AEh

Successful Response privately read by Management Controller (data is last 8 bytes in black font):
61h 01h 00h 00h D3h 84h 88h 00h 00h 00h 00h 00h 34h FFh 25h 0Ah 00h 01h 00h 00h 1Ah
75h 4Ah 30h F2h