



LEGAL NOTICE:

© **Copyright 2007 - 2017 NVM Express, Inc. ALL RIGHTS RESERVED.**

This erratum to the NVM Express revision 1.3 specification 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 erratum to the NVM Express revision 1.3 specification 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: "**© 2007 - 2016 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.

NVM Express Workgroup
c/o Virtual, Inc.
401 Edgewater Place, Suite 600
Wakefield, MA 01880
info@nvmexpress.org

NVM Express™ Technical Errata

Errata ID	002
Revision Date	09/18/2017
Affected Spec Ver.	NVM Express™ 1.3
Corrected Spec Ver.	

Errata Author(s)

Name	Company
Judy Brock	Samsung
Austin Bolen, David Black	Dell EMC
Fred Knight	NetApp Inc.
Michael Allison	SK Hynix Inc
Christoph Hellwig, Jason Gao	Western Digital
Edward Hsieh	Silicon Motion
Tom Friend	Toshiba

Errata Overview

Clarifications on MSI and MSI-X.

Clarifications on CMB and HMB.

Clarification on Firmware Download command and Firmware Commit command interactions.

Correct that the SMART Critical Composite Temperature Time covers greater than or equal to the threshold, not just greater than the threshold (to match the Warning Composite .. field).

Clarified that hosts are responsible for command coordination to meet CSE requirements from the Commands Supported and Effects information.

Clarify CMIC wording to focus on controllers rather than hosts.

Correct that NVI field indicates VI resources, not VQ resources.

Clarify that bits 6:0 of the FPI field are only valid when bit 7 is set to '1'.

The NVMe-MI Send and NVMe-MI Receive command descriptions will be moved to the NVMe-MI spec.

Clarify NS management command operations.

Delete NSes with NSID == 0xffffffff is allowed when zero or more active namespaces exist.

Clarification for Set Features on power cycles.

Clarification that IO queue pairs must be created by the host before IO commands are issued.

Clarifications to Get/Set Feature NSID settings.

Invalid Protection Information status clarification.

Clarification of meaning of NSFEAT bit set to '1'.

Clearing a Reservation clarification: all registrants are unregistered even if no reservation exists.

Allowed NVMe-MI commands during Sanitize will be included in the NVMe-MI spec.

Revision History

Revision Date	Change Description
04/29/2017	Clarifications to Get/Set Feature NSID settings
05/15/2017	Clarifications to a) Invalid Protection Information status, b) Clearing a Reservation, c) Deleting namespaces using the broadcast NSID
5/20/2017	Added removal of requirement to report status code that applies of lowest numerical value. Added various other items from the reflector.
06/06/2017	Add items from reflector.
06/08/2017 and 06/15/2017	Incorporate comments from con-call and reflector.
06/21/2017	Additional Firmware update clarifications
07/27/2017	Address issues from 30 day review; PI text and Feature value text
07/28/2017	Differentiate ILBRT field from EILBRT field in PI correction
08/03/2017	Editorial tweaks from the con-call
9/18/2017	Ratified

Incompatible Changes

The requirement that if multiple error status' applied (section 4.6.1.2) the lowest value be returned has changed to allow the device to select which of the applicable errors to return.

The SMART Critical Composite Temperature Time covers greater than or equal to the threshold, not just greater than the threshold.

The existing Feature Value text (section 7.8) contained conflicting requirements for the value of the CDW1.NSID field for features that applied to the controller (one sentence stated that 0h and FFFFFFFFh were permitted, but other text stated the NSID shall be 0h). Clarified that the NSID set to 0h or set to FFFFFFFFh are both permitted for controller feature values. The result of some Get Feature commands or Set Feature commands with an NSID value of 0h or FFFFFFFFh for a namespace specific feature was not specified; the missing cases are now specified as resulting in an error.

Description of Specification Changes

Modify a portion of section 1.6.20 as shown below:

1.6.20 NVM subsystem

An NVM subsystem includes one or more controllers, ~~one zero~~ or more namespaces, one or more ~~PCI Express~~ ports, a non-volatile memory storage medium, and an interface between the controller(s) and non-volatile memory storage medium.

Modify a portion of section 1.9 as shown below:

Technical input submitted to the NVM Express™ Workgroup is subject to the terms of the NVM Express™ Participant's agreement. Copyright © 2014-17 NVMe™ Corporation.

1.9 References

...

Single Root I/O Virtualization and Sharing Specification, revision 1.1. Available from http://www.pcisig.com/specifications/iov/single_root/.

Modify a portion of section 2.1.2 (CMD – Command) as shown below:

2.1.2 Offset 04h: CMD - Command

Bit	Type	Reset	Description
15:11	RO	0	Reserved
10	RW	0	Interrupt Disable (ID): Disables the controller from generating pin-based INTx# interrupts. This bit does not have any effect on MSI or MSI-X operation.
09	RO	0	Fast Back-to-Back Enable (FBE): Not supported by NVM Express.

Modify a portion of section 2.3.2 (MC – Message Signaled Interrupt Control) as shown below:

2.3.2 Offset MSICAP + 2h: MC – Message Signaled Interrupt Message Control

Bits	Type	Reset	Description
03:01	RO	Impl Spec	Multiple Message Capable (MMC): Indicates the number of messages the controller wants to assert.
00	RW	0	MSI Enable (MSIE): If set to '1', MSI is enabled and the traditional interrupt pins are not used to generate interrupts. If cleared to '0', MSI operation is disabled and the traditional interrupt pins are used.

Modify a portion of section 2.4 (MSI-X Capability) as shown below:

2.4 MSI-X Capability (Optional)

Start	End	Symbol	Name
MSIXCAP	MSIXCAP+1h	MXID	MSI-X Capability ID
MSIXCAP+2h	MSIXCAP+3h	MXC	MSI-X Message Control
MSIXCAP+4h	MSIXCAP+7h	MTAB	MSI-X Table Offset and Table BIR
MSIXCAP+8h	MSIXCAP+Bh	MPBA	MSI-X PBA Offset and PBA BIR

Note: It is recommended that the ~~controller~~ host allocate a unique MSI-X vector for each Completion Queue. The Table BIR and PBA BIR data structures may be allocated in either BAR0-1 or BAR4-5 in implementations.

Modify a portion of section 4.6.1.2 (Status Code) as shown below:

4.6.1.2 Status Code (SC)

The Status Code (SC) field in the completion queue entry indicates more detailed status information about the completion being reported.

Each Status Code set of values is split into three ranges:

- 00h – 7Fh: Applicable to Admin Command Set, or across multiple command sets.
- 80h – BFh: I/O Command Set Specific status codes.
- C0h – FFh: Vendor Specific status codes.

If there are multiple status codes that apply to a particular command failure, the controller shall report the status code with the lowest numerical value. Unless otherwise specified, if multiple status codes apply, then the controller selects the status code that is returned.

Modify a portion of Figure 31 (Status Code – Generic Command Status Values) as shown below:

4.6.1.2.1 Generic Command Status Definition

...

Figure 31: Status Code – Generic Command Status Values

Value	Description
...	...
02h	Invalid Field in Command: A reserved coded value or an unsupported value in a defined field (other than the opcode field). This status code should be used unless another status code is explicitly specified for a particular condition. The field may be in the command parameters as part of the Submission Queue Entry or in data structures pointed to by the command parameters.
...	...

Modify a portion of section 4.7 as shown below:

4.7 Controller Memory Buffer

The Controller Memory Buffer (CMB) is a region of general purpose read/write memory on the controller that may be used for a variety of purposes.

...

The controller may support data and metadata in the Controller Memory Buffer. All data ~~or~~ and metadata, if any, associated with a particular command shall be located in either the Controller Memory Buffer or host memory.

Modify a portion of Figure 48 (Asynchronous Event Information – SMART) as shown below:

5.2.1 Command Completion

...

Figure 48: Asynchronous Event Information – SMART / Health Status

Value	Description
0h	NVM subsystem Reliability: NVM subsystem reliability has been compromised. This may be due to significant media errors, an internal error, the media being placed in read only mode, or a volatile memory backup device failing.
1h	Temperature Threshold: A temperature is above an over temperature threshold or below an under temperature threshold (refer to section 5.21.1.4).
2h	Spare Below Threshold: Available spare space capacity has fallen below the threshold.
3h - FFh	Reserved

Modify a portion of section 5.11.1 (Firmware Commit -> Command Completion) as shown below:

5.11.1 Command Completion

~~A completion queue entry is posted to the Admin Completion Queue if the controller has completed the requested action (specified in the Commit Action field). When the command is completed, the controller posts a completion queue entry to the Admin Completion Queue indicating the status for the command.~~

Modify a portion of section 5.12 (Firmware Image Download command) as shown below:

5.12 Firmware Image Download command

The Firmware Image Download command is used to download all or a portion of an image for a future update to the controller. The Firmware Image Download command may be submitted while other commands on the Admin Submission Queue or I/O Submission Queues are outstanding. The Firmware Image Download command ~~copies the downloads~~ a new image (in whole or in part) to the controller.

...

Host software ~~shall~~ should not update Boot Partitions and firmware images simultaneously. ~~A downloaded image shall be committed using~~ After downloading an image, host software issues a Firmware Commit command before downloading another image. Processing of the first Firmware Download command after completion of a Firmware Commit command shall cause the controller to discard all remaining portion(s), if any, of downloaded images. If a reset occurs between a firmware download and completion of the Firmware Commit command, then the controller shall discard all portion(s), if any, of downloaded images. ~~If the controller does not receive a Firmware Commit command, then it shall delete the portion(s) of the new image in the case of a reset.~~

Modify a portion of Figure 93 (Get Log Page – SMART) as shown below:

5.14.1.2 SMART / Health Information (Log Identifier 02h)

...

Figure 93: Get Log Page – SMART / Health Information Log

0	Critical Warning: This field indicates critical warnings for the state of the controller. Each bit corresponds to a critical warning type; multiple bits may be set. If a bit is cleared to '0', then that critical warning does not apply. Critical warnings may result in an asynchronous event notification to the host. Bits in this field represent the current associated state and are not persistent.	
	Bit	Definition
	0	If set to '1', then the available spare space capacity has fallen below the threshold.
	1	If set to '1', then a temperature is above an over temperature threshold or below an under temperature threshold (refer to section 5.21.1.4).
	2	If set to '1', then the NVM subsystem reliability has been degraded due to significant media related errors or any internal error that degrades NVM subsystem reliability.
	3	If set to '1', then the media has been placed in read only mode.
	4	If set to '1', then the volatile memory backup device has failed. This field is only valid if the controller has a volatile memory backup solution.
7:5		Reserved
...		
199:196	Critical Composite Temperature Time: Contains the amount of time in minutes that the controller is operational and the Composite Temperature is greater than or equal to the Critical Composite Temperature Threshold (CCTEMP) field in the Identify Controller data structure in Figure 109. If the value of the CCTEMP field is 0h, then this field is always cleared to 0h regardless of the Composite Temperature value.	

Modify a portion Figure 97 (Get Log Page - Commands Supported and Effects) as shown below:

5.14.1.5 Commands Supported and Effects (Log Identifier 05h)

...

Host software may take command effects into account when determining how to submit commands and actions to take after the command is complete. It is recommended that if a command may change a particular capability that host software re-enumerate and/or re-initialize the associated capability after the command is complete. For example, if a namespace capability change may occur, then host software is recommended to pause the use of the associated namespace, submit the command that may cause a namespace capability change and wait for its completion, and then re-issue the Identify command.

If the namespace is attached to multiple controllers, the host(s) associated with those controllers should coordinate their commands to meet the Command Submission and Execution requirements (refer to Figure 97). The details of this coordination are outside the scope of this specification.

Figure 97: Get Log Page – Commands Supported and Effects Data Structure

Bits	Description										
31:19	Reserved										
18:16	Command Submission and Execution (CSE): This field defines the command submission and execution recommendations for the associated command. <table> <tr> <th>Value</th><th>Definition</th></tr> <tr> <td>000b</td><td>No command submission or execution restriction</td></tr> <tr> <td>001b</td><td>The Ccommand associated with this structure may be submitted when there is no other outstanding command to the same namespace and another command should not be submitted to the same namespace until this command is complete.</td></tr> <tr> <td>010b</td><td>The Ccommand associated with this structure may be submitted when there is no other outstanding command to any namespace and another command should not be submitted to any namespace until this command is complete.</td></tr> <tr> <td>011b – 111b</td><td>Reserved</td></tr> </table>	Value	Definition	000b	No command submission or execution restriction	001b	The C command associated with this structure may be submitted when there is no other outstanding command to the same namespace and another command should not be submitted to the same namespace until this command is complete.	010b	The C command associated with this structure may be submitted when there is no other outstanding command to any namespace and another command should not be submitted to any namespace until this command is complete.	011b – 111b	Reserved
Value	Definition										
000b	No command submission or execution restriction										
001b	The C command associated with this structure may be submitted when there is no other outstanding command to the same namespace and another command should not be submitted to the same namespace until this command is complete.										
010b	The C command associated with this structure may be submitted when there is no other outstanding command to any namespace and another command should not be submitted to any namespace until this command is complete.										
011b – 111b	Reserved										
15:05	Reserved										
04	Controller Capability Change (CCC): If this bit is set to '1', then this command may change controller capabilities. If this bit is cleared to '0', then this command does not modify controller capabilities. Controller capability changes include a firmware update that changes the capabilities reported in the CAP register.										

Modify a portion of Figure 109 (Identify – Identify Controller) Figure 112 (Secondary Controller Entry) and Figure 114 (Identify – Identify Namespace) as shown below:

5.15 Identify command

...

Figure 1: Identify – Identify Controller Data Structure

76	0	Controller Multi-Path I/O and Namespace Sharing Capabilities (CMIC): This field specifies multi-path I/O and namespace sharing capabilities of the controller and NVM subsystem. Bits 7:3 are reserved Bit 2: If set to '1' then the controller is associated with an SR-IOV Virtual Function. If cleared to '0' then the controller is associated with a PCI Function or a Fabrics connection . Bit 1: If set to '1' then the NVM subsystem may be connected to more than one host (e.g., it may contain two or more controllers) . If cleared to '0' then the NVM subsystem may only be connected to a single host (e.g., it contains only a single controller) . As described in section 1.4.1, an NVM subsystem that contains multiple controllers may be used by multiple hosts, or may provide multiple paths for a single host. Bit 0: If set to '1' then the NVM subsystem may contain more than one NVM subsystem port. If cleared to '0' then the NVM subsystem contains only a single NVM subsystem port.
----	---	--

Figure 112: Secondary Controller Entry

9:8	Virtual Function Number (VFN): If the secondary controller is an SR-IOV VF, this field indicates its VF Number, where VF Number > 0, and VF Number is no larger than the total number of VFs indicated by the TotalVFs register (refer to Single Root I/O Virtualization and Sharing Specification) in the PF's SR-IOV Extended Capability Capabilities structure. If the secondary controller is not an SR-IOV VF, then this field is cleared to zero.
11:10	Number of VQ Flexible Resources Assigned (NVQ): This field indicates the number of VQ Flexible Resources currently assigned to the indicated secondary controller.
13:12	Number of VI Flexible Resources Assigned (NVI): This field indicates the number of VQ VI Flexible Resources currently assigned to the indicated secondary controller.
31:14	Reserved

Figure 114: Identify – Identify Namespace Data Structure, NVM Command Set Specific

32	0	<p>Format Progress Indicator (FPI): If a format operation is in progress, this field indicates the percentage of the namespace that remains to be formatted.</p> <p>Bit 7 if set to '1' indicates that the namespace supports the Format Progress Indicator defined by bits 6:0 in this field. If this bit is cleared to '0', then the namespace does not support the Format Progress Indicator and bits 6:0 in this field shall be cleared to 0h.</p> <p>Bits 6:0 indicate the percentage of the Format NVM command that remains to be completed (e.g., a value of 25 indicates that 75% of the Format NVM command has been completed and 25% remains to be completed). If bit 7 is set to '1', then a value of 0 indicates that the namespace is formatted with the format specified by the FLBAS and DPS fields in this data structure and there is no Format NVM command in progress.</p>
----	---	--

Modify a portion of 5.17 (NVMe-MI Receive command) as shown below:

5.17 NVMe-MI Receive command

Refer to the NVM Express Management Interface Specification for details on the NVMe-MI Receive command.

~~The NVMe-MI Receive command transfers an NVMe-MI Response Message from the controller to the host that corresponds to an NVMe-MI Request Message that was previously submitted to the controller (refer to section 5.18). Refer to the NVM Express Management Interface (NVMe-MI) specification for the format and servicing of the NVMe-MI Response Message.~~

~~The fields used are Data Pointer and Command Dword 10 fields. All other command specific fields are reserved.~~

Figure 2: NVMe-MI Receive—Data Pointer

Bit	Description
427:00	Data Pointer (DPTR): This field specifies the the start of the data buffer that contains an NVMe-MI Response Message. Refer to Figure 11 for the definition of this field.

Figure 3: NVMe-MI Receive—Command Dword 10

Bit	Description
31:08	Reserved
07:00	NVMe-MI Specific 0 (NMSP0): The value of this field contains bits 07:00 of the NVMe-MI Specific field as defined in the NVMe-MI specification.

5.17.1 Command Completion

When the command is completed successfully, the controller shall post a completion queue entry to the Admin Completion Queue indicating successful completion status. An NVMe-MI Receive command is completed successfully when the NVMe-MI Response Message has been transferred to the data buffer pointed to by Data Pointer.

The definition of Dword 0 of the completion queue entry is in Figure 119.

Figure 4: NVMe-MI Receive — Completion Queue Entry Dword 0

Bit	Description
31:00	Number of Dwords (NUMD): This field specifies the length of the data structure returned in Dwords. This is a 0's based value.

Modify a portion of 5.18 (NVMe-MI Send command) as shown below:

5.18 NVMe-MI Send command

Refer to the NVM Express Management Interface Specification for details on the NVMe-MI Send command.

The NVMe-MI Send command is used to transfer an NVMe-MI Request Message to the controller. Refer to the NVM Express Management Interface (NVMe-MI) specification for the format and servicing of the NVMe-MI Request Message.

The fields used are Data Pointer, Command Dword 10, and Command Dword 11 fields. All other command specific fields are reserved.

Figure 5: NVMe-MI Send — Data Pointer

Bit	Description
127:00	Data Pointer (DPTR): This field specifies the start of the data buffer that contains an NVMe-MI Request Message. Refer to Figure 11 for the definition of this field.

Figure 6: NVMe-MI Send — Command Dword 10

Bit	Description
31:08	Reserved
07:00	NVMe-MI Specific 0 (NMSP0): The value of this field contains bits 07:00 of the NVMe-MI Specific field as defined in the NVMe-MI specification.

Figure 7: NVMe-MI Send — Command Dword 11

Bit	Description
31:00	Number of Dwords (NUMD): This field specifies the length of the data structure transferred in Dwords. This is a 0's based value.

5.18.1 Command Completion

When the command is completed successfully, the controller shall post a completion queue entry to the Admin Completion Queue indicating successful completion status. An NVMe-MI Send command is completed successfully when the NVMe-MI Request Message has been transferred to the controller.

Modify portions of section 5.20 (Namespace Management command) as shown below:

5.20 Namespace Management command

The Namespace Management command is used to manage namespaces (refer to section 8.12), including create and delete operations. Note: The controller continues to execute commands submitted to I/O Submission Queues while this operation is in progress.

Host software uses the Namespace Attachment command to attach or detach a namespace to or from a controller. The create operation does not attach the namespace to a controller. As a side effect of the delete operation, the namespace is detached from ~~any-all~~ controllers as ~~it-the namespace~~ is no longer present in the system. It is recommended that host software detach all controllers from a namespace prior to deleting the namespace. If the namespace is attached to another controller (i.e., a controller other than the controller processing the operation) and that controller has Namespace Attribute Notices enabled (refer to Figure 148), when a delete operation is requested, then as part of the delete operation a Namespace Attribute Notice is issued by that controller to indicate a namespace change ~~(if Namespace Attribute Notices are supported)~~.

...

The Namespace Identifier (CDW1.NSID) field is used as follows for create and delete operations:

- Create: The CDW1.NSID field is reserved for this operation; host software shall set this field to a value of 0h. The controller shall select an available Namespace Identifier to use for the operation.
- Delete: This field specifies the previously created namespace to delete in this operation. Specifying a value of FFFFFFFFh ~~may be~~ is used to delete all namespaces in the NVM subsystem. ~~If the value of FFFFFFFFh is specified and there are zero valid namespaces, the command completes successfully.~~

Modify a portion of Figure 133 (Set Features – Command Dword 10) as shown below:

5.21 Set Features command

...

Figure 133: Set Features – Command Dword 10

Bit	Description
31	Save (SV): This field specifies that the controller shall save the attribute so that the attribute persists through all power states and resets. The controller indicates in bit 4 of the Optional NVM Command Support field of the Identify Controller data structure in Figure 109 whether this field is supported. If the Feature Identifier specified in the Set Features command is not saveable by the controller and the controller recieves-receives a Set Features command with the Save bit set to one, then the command shall be aborted with a status of Feature Identifier Not Saveable.
30:08	Reserved
07:00	Feature Identifier (FID): This field indicates the identifier of the Feature that attributes are being specified for.

Modify a portion of section 5.21.1 (Feature Specific Information) as shown below:

5.21.1 Feature Specific Information

Figure 134 defines the Features that may be configured with Set Features and retrieved with Get Features. Figure 135 defines Features that are specific to the NVM Command Set. Some Features utilize a memory buffer to configure or return attributes for a Feature, whereas others only utilize a Dword in the command or completion queue entry. Feature values that are not persistent across power cycles and resets are **restored reset** to their default values as part of a controller reset operation. The default value for each Feature is vendor specific and set by the manufacturer unless otherwise specified; it is not changeable. For more information on Features, including default, saveable, and current value definitions, refer to section 7.8.

...

Modify a portion of Figure 135 as shown below:

Figure 135: Set Features, NVM Command Set Specific – Feature Identifiers

Feature Identifier	O/M ⁶	Persistent Across Power Cycle States and Reset ¹	Uses Memory Buffer for Attributes	Description
80h	O	Yes	No	Software Progress Marker
81h	O ²	No	Yes	Host Identifier
82h	O ³	No	No	Reservation Notification Mask
83h	O ³	Yes	No	Reservation Persistence
84h – BFh				Reserved
NOTES: 1. This column is only valid if bit 4 in the Optional NVM Command Support field of the Identify Controller data structure in Figure 109 is cleared to '0'. This column is only valid if the feature is not saveable (refer to section 7.8). If the feature is saveable, then this column is not used and any feature may be configured to be saved across power cycles and reset. 2. Mandatory if reservations are supported as indicated in the Identify Controller data structure. 3. Mandatory if reservations are supported by the namespace as indicated by a non-zero value in the Reservation Capabilities (RESCAP) field in the Identify Namespace data structure. 4. O/M: O = Optional, M = Mandatory.				

Modify a portion of Figure 153, 155, 156, and 158 (Host Memory Buffer) as shown below:

5.21.1.13 Host Memory Buffer (Feature Identifier 0Dh), (Optional)

...

Figure 153: Host Memory Buffer– Command Dword 13

Bit	Description
31:0400	Host Memory Descriptor List Lower Address (HMDLLA): This field specifies the lower 32 bits of the physical location of the Host Memory Descriptor List (refer to Figure 156) for the Host Memory Buffer. This address shall be 16 byte aligned, indicated by bits 3:0 being cleared to 0h. Note: The controller shall operate as if bits 3:0 are cleared to 0h. However, the controller is not required to check that bits 3:0 are cleared to 0h.
03:00	Reserved

...

The Host Memory Descriptor List Address (HMDLLA/HMDLUA) **points to-specifies the address of** a physically contiguous data structure in host memory that describes the address and length pairs of the Host Memory Buffer. The number of address and length pairs is specified in the Host Memory Descriptor List Entry Count in Figure 155. The Host Memory Descriptor List is described in Figure 156.

Figure 155: Host Memory Buffer – Command Dword 15

Bit	Description
31:00	Host Memory Descriptor List Entry Count (HMDLEC): This field specifies the number of entries provided in the Host Memory Descriptor List.

Figure 156: Host Memory Buffer – Host Memory Descriptor List

Bytes	Description
15:0	Host Memory Buffer Descriptor (refer to Figure 157) Entry 0
31:16	Host Memory Buffer Descriptor Entry 1
47:32	Host Memory Buffer Descriptor Entry 2
63:48	Host Memory Buffer Descriptor Entry 3
...	...
16*n+15:16*n	Host Memory Buffer Descriptor Entry n (where n = HMDLEC - 1) (refer to Figure 155)

...

Figure 158: Host Memory Buffer – Attributes Data Structure

Byte	Description
3:0	Host Memory Buffer Size (HSIZE): This field specifies indicates the size of the host memory buffer allocated in memory page size units.
7:4	Host Memory Descriptor List Address Lower (HMDLAL): This field specifies indicates the lower 32 bits of the physical location of the Host Memory Descriptor List for the Host Memory Buffer. This address shall be 16 byte aligned. The lower 4 bits shall be cleared to zero.
11:8	Host Memory Descriptor List Address Upper (HMDLAU): This field specifies indicates the upper 32 bits of the physical location of the Host Memory Descriptor List for the Host Memory Buffer. This address shall be 16 byte aligned. The lower 4 bits shall be cleared to zero.
15:12	Host Memory Descriptor List Entry Count (HMDLEC): This field specifies indicates the number of valid Host Memory Descriptor Entries.
4095:16	Reserved

Modify a portion of section 5.21.1.19 (Host Identifier) as shown below:

5.21.1.19 Host Identifier (Feature Identifier 81h), (Optional¹)

...

The Host Identifier is contained in the data structure indicated in Figure 166. The attributes are specified in Command Dword 11. If a Get Features command is issued for this Feature, the data structure specified in Figure 166 is returned in the data buffer for that command.

The requirements and use of the Host Identifier feature is dependent on whether the implementation supports NVMe over PCIe or NVMe over Fabrics. Refer to section 5.21.1.19.1 and section 5.21.1.19.2.

¹ Mandatory if reservations are supported as indicated in the Identify Controller data structure.

Modify Figures 203, 217, 242, and 250 as shown below:

6.6.1 Command Completion

...

Figure 203: Compare – Command Specific Status Values

Value	Description
81h	Invalid Protection Information: The Protection Information Field (PRINFO) (refer to Figure 200) settings specified in the command are invalid for the Protection Information with which the namespace was formatted (refer to the PI field in Figure 176 and the DPS field in Figure 114) or the EILBRT field is invalid (refer to section 8.3.1.5).

...

6.9.1 Command Completion

...

Figure 217: Read – Command Specific Status Values

Value	Description
80h	Conflicting Attributes: The attributes specified in the command are conflicting.
81h	Invalid Protection Information: The Protection Information Field (PRINFO) (refer to Figure 213) settings specified in the command are invalid for the Protection Information with which the namespace was formatted (refer to the PI field in Figure 176 and the DPS field in Figure 114) or the EILBRT field is invalid (refer to section 8.3.1.5).

...

6.14.1 Command Completion

...

Figure 242: Write – Command Specific Status Values

Value	Description
80h	Conflicting Attributes: The attributes specified in the command are conflicting.
81h	Invalid Protection Information: The Protection Information Field (PRINFO) (refer to Figure 238) settings specified in the command are invalid for the Protection Information with which the namespace was formatted (refer to the PI field in Figure 176 and the DPS field in Figure 114) or the ILBRT field is invalid (refer to section 8.3.1.5).
82h	Attempted Write to Read Only Range: The LBA range specified contains read-only blocks.

...

6.16.1 Command Completion

...

Figure 250: Write Zeroes – Command Specific Status Values

Value	Description
81h	Invalid Protection Information: The Protection Information Field (PRINFO) (refer to Figure 247) settings specified in the command are invalid for the Protection Information with which the namespace was formatted (refer to the PI field in Figure 176 and the DPS field in Figure 114) or the ILBRT field is invalid (refer to section 8.3.1.5).
82h	Attempted Write to Read Only Range: The LBA range specified contains read-only blocks.

Modify a portion of the last paragraph of section 7.1 (Controller Architecture -> Introduction) as shown below:

7 Controller Architecture

7.1 Introduction

...

There are no ordering restrictions for completions to the host. Each completion queue entry identifies the Submission Queue Identifier and Command Identifier of the associated command. Host software uses this information to correlate the completions with the commands submitted to the Submission Queue(s).

Host software is responsible for creating **all required I/O Submission Queues** and **I/O Completion Queues** prior to **using those queue pairs to submit submitting** commands to the controller. I/O Submission **Queues** and I/O Completion Queues are created using **the Create I/O Submission Queue command (refer to section 5.4) and the Create I/O Completion Queue command (refer to section 5.3)-Admin commands defined in section 5.**

Modify a portion of section 7.8 (Feature Values) as shown below:

7.8 Feature Values

...

Set Features may be used to modify the saveable and current value for a Feature. Get Features may be used to read the default, saveable, and current value for a Feature. If the controller does not support a saveable value for a Feature, then the default value is returned for the saveable value in Get Features.

Feature settings may apply to:

- a) the controller (i.e., the feature is not namespace specific); or
- b) a namespace (i.e., the feature is namespace specific).

~~the entire controller (and all associated namespaces) or may apply to each namespace individually. To change or retrieve a value that applies to the controller and all associated namespaces, host software sets CDW1.NSID to 0h or FFFFFFFFh in the Set Features or Get Features command. Features that are not namespace specific shall have the CDW1.NSID field set to 0h.~~

~~To change or retrieve a value that applies to a specific namespace, host software sets CDW1.NSID to the identifier of that namespace in the Set Features or Get Features command. If host software specifies a valid CDW1.NSID value that is not 0h or FFFFFFFFh and the Feature is not namespace specific, then a Set Features command returns the Feature Not Namespace Specific status code, whereas a Get Features command returns the Feature value that applies to the entire controller.~~

~~For feature values that apply to the controller:~~

- ~~a) if the CDW1.NSID field is set to 0h or FFFFFFFFh, then:
 - the Set Features command shall set the specified feature value for the controller; and
 - the Get Features command shall return the current setting of the requested feature value for the controller,~~
- ~~and~~
- ~~b) if the CDW1.NSID field is set to a valid namespace identifier (refer to section 6.1), then:
 - the Set Features command shall fail with a status code of Feature Not Namespace Specific;
 - and
 - the Get Features command shall return the current setting of the requested feature value for the controller.~~

~~For feature values that apply to a namespace:~~

- ~~a) if the CDW1.NSID field is set to an active namespace identifier (refer to section 6.1), then:
 - the Set Features command shall set the specified feature value of the specified namespace;
 - and~~

- the Get Features command shall return the current setting of the requested feature value for the specified namespace,
- b) if the CDW1.NSID field is set to FFFFFFFFh, then
 - the Set Features command shall set the specified feature value for all namespaces attached to the controller processing the command; and
 - the Get Features command shall fail with a status code of Invalid Namespace or Format, and
- c) if the CDW1.NSID field is set to any other value, then the Set Features command and the Get Features command shall fail as described in the rules for namespace identifier usage in Figure 11.

If the controller supports the Save field in the Set Features command and the Select field in the Get Features command, then any Feature Identifier that is namespace specific may be saved on a per namespace basis.

Modify a portion of the last paragraph of section 7.11 (Unique Identifier) as shown below:

7.11 Unique Identifier

...

The Identify Namespace data structure contains the IEEE Extended Unique Identifier (EUI64) and the Namespace Globally Unique Identifier (NGUID) fields. EUI64 is an 8-byte EUI-64 identifier and NGUID is a 16-byte identifier based on EUI-64. When creating a namespace, the controller specifies a globally unique value in the EUI64 or NGUID field (the controller may optionally specify a globally unique value in both fields). In cases where the 64-bit EUI64 field is unable to ensure a globally unique namespace identifier, the EUI64 field shall be cleared to 0h. When not implemented, these fields contain a value of 0h. ~~A controller may reuse a non-zero NGUID or EUI64 value for a new namespace after the original namespace using the value has been deleted.~~ If bit 3 in NSFEAT is cleared to '0', then a controller may reuse a non-zero NGUID/EUI64 value for a new namespace after the original namespace using the value has been deleted. ~~If bit 3 in NSFEAT is set to '1', then a controller shall not reuse a non-zero NGUID/EUI64 for a new namespace after the original namespace using the value has been deleted.~~

Modify the last paragraph of section 8.1 (Firmware Update Process) as shown below:

8.1 Firmware Update Process

The process for a firmware update to be activated by a reset is:

...

Host software ~~shall~~ ~~should~~ not update multiple firmware images simultaneously. ~~A downloaded image shall be committed using~~ After downloading an image, host software issues a Firmware Commit command before downloading additional firmware images. Processing of the first Firmware Download command after completion of a Firmware Commit command shall cause the controller to discard remaining portions, if any, of downloaded images. If a reset occurs between a firmware download and completion of the Firmware Commit command, then the controller shall discard all portion(s), if any, of downloaded images. ~~If the controller does not receive a Firmware Commit command, then it shall delete the portion(s) of the new image in the case of a reset.~~

Modify a portion of section 8.3.1.5 (Control of Protection Information Checking) as shown below:

8.3.1.5 Control of Protection Information Checking - PRCHK

...

For Type 1 protection, if bit 0 of the PRCHK field is set to '1', then the controller compares the protection information Reference Tag field to the computed reference tag. The value of the computed reference tag for

the first LBA of the command is the value contained in the Initial Logical Block Reference Tag (ILBRT) or Expected Initial Logical Block Reference Tag (EILBRT) field in the command. If the namespace is formatted for Type 1 or Type 2 protection, the computed reference tag is incremented for each subsequent logical block. If the namespace is formatted for Type 3 protection, the reference tag for each subsequent logic block remains the same as the initial reference tag. Unlike SCSI Protection Information Type 1 protection which implicitly uses the least significant four bytes of the LBA, the controller always uses the ILBRT or EILBRT field and requires host software to initialize the ILBRT or EILBRT field to the least significant four bytes of the LBA when Type 1 protection is used. In Type 1 protection, the controller should check the ILBRT field or the EILBRT field for the correct value; ~~if there is any miscompare, the command completes with an error of Invalid Protection Information. If the ILBRT or EILBRT field;~~ if the value does not match the least significant four bytes of the LBA, then the controller completes the command with a status of ~~an Invalid Protection Information~~ ~~status code~~.

For Type 2 protection, if bit 0 of the PRCHK field is set to '1', then the controller compares the protection information Reference Tag field from each logical block to the computed reference tag. The computed reference tag is incremented for each subsequent logical block. The value of the computed reference tag for the first LBA of the command is the value contained in the ILBRT or EILBRT field in the command. Host software may set the ILBRT and EILBRT fields to any value.

For Type 3 protection, if bit 0 of the PRCHK field is set to '1', then the command ~~should be aborted with status Invalid Protection Information, but~~ may be aborted with status Invalid Field in Command. The controller may ignore the ILBRT and EILBRT fields when Type 3 protection is used because the computed reference tag remains unchanged.

Modify a portion of section 8.8.8 (Clearing a Reservation) as shown below:

8.8.8 Clearing a Reservation

A host that is a registrant may clear a reservation (i.e., force the release of a reservation held on the namespace and unregister all registrants) by executing a Reservation Release command, setting the Reservation Release Action (RRELA) field to 001b (i.e., Clear), and supplying the current reservation key associated with the host in the Current ~~reservation~~ Reservation Key (CRKEY) field. If the value in the CRKEY field does not match the value used by the host to register with the namespace, then the command shall be aborted with status Reservation Conflict. If the host is not a registrant, then the command is aborted with a status of Reservation Conflict. When a ~~command to clear a~~ reservation is ~~executed cleared~~ the following occur as an atomic operation: ~~any~~the reservation held on the namespace, ~~if any~~, is released, and all registrants are unregistered from the namespace.

A reservation preempted notification occurs on all controllers in the NVM subsystem that are associated with hosts that have their registrations removed as a result of actions taken in this section except those associated with the host that issued the Reservation Release command.

Modify Figure 287 (Sanitize Operations – Admin Commands Allowed) as shown below:

8.15 Sanitize Operations (Optional)

...

8.15.1 Command Restrictions

...

Figure 287: Sanitize Operations – Admin Commands Allowed

Admin Command	Additional Restrictions														
Abort															
Asynchronous Event Request															
Create I/O Completion Queue															
Create I/O Submission Queue															
Delete I/O Completion Queue															
Delete I/O Submission Queue															
Get Features															
Get Log Page	<p>The log pages allowed are listed below.</p> <table> <tr> <th>Log Pages</th><th>Additional Restrictions</th></tr> <tr> <td>Error Information</td><td>Return zeros in the LBA field.</td></tr> <tr> <td>SMART / Health Information</td><td></td></tr> <tr> <td>Changed Namespace List</td><td></td></tr> <tr> <td>Reservation Notification</td><td></td></tr> <tr> <td>Sanitize Status</td><td></td></tr> </table>	Log Pages	Additional Restrictions	Error Information	Return zeros in the LBA field.	SMART / Health Information		Changed Namespace List		Reservation Notification		Sanitize Status			
Log Pages	Additional Restrictions														
Error Information	Return zeros in the LBA field.														
SMART / Health Information															
Changed Namespace List															
Reservation Notification															
Sanitize Status															
Identify															
Keep Alive															
NVMe-MI Receive	Prohibited unless explicitly allowed in the NVM Express Management Interface Specification.														
NVMe-MI Send															
Set Features															
Opcode 7Fh	<p>The Fabric Commands allowed are listed below. Refer to the NVMe over Fabrics specification.</p> <table> <tr> <th>Fabrics Commands</th><th>Additional Restrictions</th></tr> <tr> <td>Property Set</td><td></td></tr> <tr> <td>Connect</td><td></td></tr> <tr> <td>Property Get</td><td></td></tr> <tr> <td>Authentication Send</td><td></td></tr> <tr> <td>Authentication Receive</td><td></td></tr> <tr> <td>Vendor Specific</td><td>Commands are allowed that do not affect or retrieve user data.</td></tr> </table>	Fabrics Commands	Additional Restrictions	Property Set		Connect		Property Get		Authentication Send		Authentication Receive		Vendor Specific	Commands are allowed that do not affect or retrieve user data.
Fabrics Commands	Additional Restrictions														
Property Set															
Connect															
Property Get															
Authentication Send															
Authentication Receive															
Vendor Specific	Commands are allowed that do not affect or retrieve user data.														
Vendor Specific	Commands are allowed that do not affect or retrieve user data.														