



LEGAL NOTICE:

© **Copyright 2008 to 2022 NVM Express, Inc. ALL RIGHTS RESERVED.**

This erratum 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, 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 2022 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.
PCI-SIG®, PCI Express®, and PCIe® are registered trademarks of PCI-SIG.
InfiniBand™ is a trademark and servicemark of the InfiniBand Trade Association.

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

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant’s agreement. Copyright © 2008 to 2022 NVM Express, Inc.

NVM Express® Technical Errata

Errata ID	110
Revision Date	07/10/2022
Affected Spec Ver.	NVM Express® Base Specification Revision 2.0b NVM Express® NVM Command Set Specification Revision 1.0b NVM Express® Zoned Namespace Command Set Specification Revision 1.1b NVM Express® Key Value Command Set Specification Revision 1.0b NVM Express® TCP Transport Specification 1.0b NVM Express® PCIe Transport Specification 1.0b NVM Express® RDMA Transport Specification 1.0b
Corrected Spec Ver.	

Errata Author(s)

Name	Company
Judy Brock, Mike Allison, Bill Martin	Samsung
David Black	DellEMC
Fred Knight	NetApp
Paul Suhler, John Geldman	Kioxia
Curtis Ballard	HPE

Errata Overview

This ECN updates and clarifies various text within the NVM Express Base Specification Revision 2.0b, NVM Express NVM Command Set Specification Revision 1.0b, NVM Express Zoned Namespace Command Set Specification 1.1b, NVM Express Key Value Command Set Specification Revision 1.0b, NVM Express TCP Transport Specification 1.0b, NVM Express PCIe Transport Specification 1.0b, and the NVM Express RDMA Transport Specification 1.0a.

Revision History

Revision Date	Change Description
2/16/2021	Initial creation
2/22/2022	Added more reported items.
2/23/2022	Aligned the log page figure titles and the introduction of the Log Specific field use.
2/24/2022	Editorial cleanup for initial review.
2/28/2022	Clarified the definition of CAP.CSS bit 6 in terms of the value specified by CAP.CSS bit 0. Clarified the simultaneously outstanding Asynchronous Event Request commands limits. Clarified the aborting of a Get Log Page command when media is not ready when the host is clearing the event associated with that log page.

3/3/2022	Added the Over Temperature field in the Thermal Excursion Event. Fixed a trademark.
3/8/2022	Updating constancy is using "NVM Express Base Specification" in the NVM Express family of specifications.
3/10/2022	Removed the User Data reference in the LBA field in the Error Information Log Entry data structure and in the LBA Failing field in the Self-test Result data structure. Edits from Technical WG review. Updated a "should" to a "shall" for the Get Log Page command.
3/24/2022	Clarified allocated and deallocated logical blocks. Edits from review in Technical WG meeting.
3/30/2022	Removed comments already reviewed.
3/31/2022	Removed comments and accepted changes for 30 day member review.
5/29/2022	Integrated
7/6/2022	Editorial changes per Mike Allison
7/10/2022	Editorial changes per Mike Allison

Description of Changes

NVM Express Base Specification 2.0b:

Backward Incompatible Changes:

- Corrected the status code returned for the Get Log Page command for a Log Identifier not support from Invalidated Field in Command to Invalid Log Page.
- Corrected the Identify command with CNS 05h to return
- Clarified that the Over Temperature field in the Thermal Excursion Event is the absolute value of the difference.
- If an Admin command or a Feature Identifier is prohibited by a Lockdown command for the entire NVM subsystem and a controller that does not support the Admin command or Feature Identifier, then the specification has multiple requirements on the status code returned for the aborted command. The change in this specification states that the controller shall issue a status code of Command Prohibited by Command and Feature Lockdown if the controller:
 - Supports the Admin command or Feature Identifier; and
 - The Admin command or Feature Identifier is prohibited from being processed by a Lockdown command.

Editorial Changes:

- Replaced "NVMe controller" with "controller" when not associated with physical NVMe controller.
- Removed "NVMe" from "NVMe Controller Level Reset".
- Changed the use of "log page" to lowercase text.
- Corrected the reference to a log page to use "log page" instead of "log"
- Clarified the Namespace Identification Descriptor list for the Identify command with CNS set to 03h.
- Aligned the titles of all log pages to use a consistent pattern.
- Clarified that references to Error Information Log Entries in relation to counting is about those entries inserted into the Error Information log page.
- Aligned the text User Data fields in log pages.
- Aligned the text to be consistent for the Log Specific field in all log pages that use the field.
- Clarified the definition of CAP.CSS bit 6.

- Clarified the simultaneously outstanding Asynchronous Event Request commands limits.
- Clarified the aborting of a Get Log Page command when media is not ready when the host is clearing the event associated with that log page.
- Clarified CAP.CSS bit 0 set to '1' and CAP.CSS bit 6 cleared to '0' indicates that the I/O Identify I/O Command Set data structure is not supported.
- Updated the Memory-based Transport Controller Initialization and the Message-based Transport Controller Initialization step 8 to clarify the enabled I/O Command Set.
- Fixed the consistent use of "n/a" versus "N/A"

NVM Express NVM Command Set Specification 1.0b:

Backward Incompatible Changes:

- Clarified that a logical block is marked as allocated on a Copy command.

Editorial Changes:

- Removed the reserved field in bytes 4095:4080 in the Source Range Entries Descriptor Format 1h example.
- Updated the definition of CNS 00h in the table of CNS values.
- Updated the LBA Data Size field that when cleared to 0h to refer to the NLBAF field.
- Aligned the titles of all log pages to use a consistent pattern.
- Clarified the User Data field in the Error Information log page.
- Clarified the User Data field in the Device Self-test log page.
- Clarified the proper name of the NVM Express Base Specification
- Clarified that a logical block may be allocated by a sanitize operation.
- Clarified that a logical block may be allocated or deallocated due to Vendor Specific means.

NVM Express Zoned Namespace Command Set 1.1b:

Editorial Changes:

- Aligned the titles of all log pages to use a consistent pattern.
- Clarified the use of "it" by stating the exact noun.
- Fixed the consistent use of "n/a" versus "N/A"
- Clarified the proper name of the NVM Express Base Specification

NVM Express Key Value Command Set 1.0b:

Backward Incompatible Changes:

- Clarified the User Data field in the Error Information log page is undefined.
- Clarified the User Data field in the Device Self-test log page is undefined.
- Clarified the proper name of the NVM Express Base Specification

Editorial Changes:

- None

NVM Express TCP Transport Specification 1.0b:

Editorial Changes:

- Corrected the Sub Type value illustration of the command that performs a 12,288 bytes (3000h bytes) controller to host Command Data Buffer transfer.

- Clarified the proper name of the NVM Express Base Specification

NVM Express PCIe Transport Specification 1.0b:

Editorial Changes:

- Clarified the proper name of the NVM Express Base Specification

NVM Express RDMA Transport Specification 1.0a:

Editorial Changes:

- Clarified the proper name of the NVM Express Base Specification

Editor's Note:

BLACK text indicates unchanged text. **BLUE** text indicates newly inserted text. **RED stricken** text indicates deleted text; **ORANGE** text indicates changes from another ECN. **Purple** text indicates moved text without changes; **GREEN** text indicates editor notes.

Description of NVM Express Base Specification 2.0b changes

Modify a portion of section 1.6 as shown below:

1 Introduction

...

1.6 I/O Command Set specific definitions used in **this specification** ~~the NVMe Base specification~~

...

Modify a portion of section 3.1.3.1 as shown below:

3 NVM Express Architecture

...

3.1 NVM Controller Architecture

...

3.1.3 Controller Properties

...

3.1.3.1 Offset 0h: CAP – Controller Capabilities

...

Figure 36: Offset 0h: CAP – Controller Capabilities

Bits	Type	Reset	Description
...			

Figure 36: Offset 0h: CAP – Controller Capabilities

Bits	Type	Reset	Description															
44:37	RO	Impl Spec	Command Sets Supported (CSS): This field indicates the I/O Command Set(s) that the controller supports. Bit 7 is set to '1' if no I/O Command Set is supported. For Discovery controllers, this field should have only bit 0 set to 1.															
			Bit	Definition	0	NVM command set	5:1	Reserved	6	Controller supports one or more I/O Command Sets and supports the Identify I/O Command Set data structure (refer to section 5.17.2.21). Controllers that support I/O Command Sets other than the NVM Command Set shall set this bit to '1'. Controllers that only support the NVM Command Set may set this bit to '1' to indicate support for the Command Set Identifier field in commands that use the Command Set Identifier field.	7	No I/O Command Set is supported (i.e., only the Admin Command Set is supported). <added period> This bit shall be set to '1' if no I/O Command Set is supported.	6	Controller supports one or more I/O Command Sets and supports the Identify I/O Command Set data structure (refer to section 5.17.2.21). Controllers that support I/O Command Sets other than the NVM Command Set shall set this bit to '1'. Controllers that only support the NVM Command Set may set this bit to '1'.	5:1	Reserved	0	NVM Command Set or a Discovery controller
			Bit	Definition														
			0	NVM command set														
			5:1	Reserved														
			6	Controller supports one or more I/O Command Sets and supports the Identify I/O Command Set data structure (refer to section 5.17.2.21). Controllers that support I/O Command Sets other than the NVM Command Set shall set this bit to '1'. Controllers that only support the NVM Command Set may set this bit to '1' to indicate support for the Command Set Identifier field in commands that use the Command Set Identifier field.														
			7	No I/O Command Set is supported (i.e., only the Admin Command Set is supported). <added period> This bit shall be set to '1' if no I/O Command Set is supported.														
			6	Controller supports one or more I/O Command Sets and supports the Identify I/O Command Set data structure (refer to section 5.17.2.21). Controllers that support I/O Command Sets other than the NVM Command Set shall set this bit to '1'. Controllers that only support the NVM Command Set may set this bit to '1'.														
5:1	Reserved																	
0	NVM Command Set or a Discovery controller																	

...

Modify a portion of section 3.3.2.9 as shown below:

3.3 NVM Queue Models

...

3.3.2 Message-based Transport Queue Model

...

3.3.2.9 Transport Requirements

...

The NVMe Transport may support NVMe Transport error detection and report errors to the NVMe layer in command status values. The controller may record NVMe Transport specific errors in the Error Information Log page. Transport errors that cause loss of a message or loss of data in a way that the low-level NVMe Transport cannot replay or recover should cause:

- the deletion of the individual I/O Queues (refer to section 3.3.2.4) and the associated NVMe Transport connection on which that NVMe Transport level error occurred; or
- termination of the NVMe Transport connection and the association between the host and controller.

...

Modify a portion of section 3.3.3.2.1 as shown below:

3.3.3 Queueing Data Structures

...

3.3.3.2 Common Completion Queue Entry

...

3.3.3.2.1 Status Field Definition

...

Figure 92: Completion Queue Entry: Status Field

Bits	Description
...	
30	More (M): If set to '1', there is more status information for this command as part of the Error Information log page that may be retrieved with the Get Log Page command. If cleared to '0', there is no additional status information for this command. Refer to section 5.16.1.2.
...	

...

Modify a portion of section 3.5.1 as shown below:

3.5.1 Memory-based Transport Controller Initialization

...

- ~~8. The host determines any I/O Command Set specific configuration information as follows:~~
- ~~a. If the CC.CSS field is set to 000b, then the host should determine the configuration of each namespace by issuing the Identify command for each namespace, specifying the Identify Namespace data structure (CNS 00h);~~
 - ~~b. If the CAP.CSS bit 6 is set to '1', then the host does the following:~~
 - ~~i. Issue the Identify command specifying the Identify I/O Command Set data structure (CNS 1Ch);~~
 - ~~ii. Issue the Set Features command with the I/O Command Set Profile Feature Identifier (FID 19h) specifying the index of the I/O Command Set Combination (refer to Figure 289) to be enabled;~~
 - ~~iii. For each I/O Command Set that is enabled:~~
 - ~~1. Issue the Identify command specifying the I/O Command Set specific Active Namespace ID list (CNS 07h) with the appropriate Command Set Identifier (CSI) value of that I/O Command Set;~~
 - ~~2. For each NSID that is returned:~~
 - ~~a. If the enabled I/O Command Set is the NVM Command Set or an I/O Command Set based on the NVM Command Set (e.g., the Zoned Namespace Command Set) issue the Identify command specifying the Identify Namespace data structure (CNS 00h);~~
 - ~~b. Issue the Identify command specifying each of the following data structures (refer to Figure 274): the I/O Command Set specific Identify Namespace data structure;~~

~~the I/O Command Set specific Identify Controller data structure, and the I/O Command Set independent Identify Namespace data structure;~~

8. The host determines any I/O Command Set specific configuration information as follows:
 - a. If the CAP.CSS bit 6 is set to '1', then the host does the following:
 - i. Issue the Identify command specifying the Identify I/O Command Set data structure (CNS 1Ch); and
 - ii. Issue the Set Features command with the I/O Command Set Profile Feature Identifier (FID 19h) specifying the index of the I/O Command Set Combination (refer to Figure 289) to be enabled;
and
 - b. For each I/O Command Set that is enabled (Note: the NVM Command Set is enabled if the CC.CSS field is set to 000b):
 - i. Issue the Identify command specifying the I/O Command Set specific Active Namespace ID list (CNS 07h) with the appropriate Command Set Identifier (CSI) value of that I/O Command Set; and
 - ii. For each NSID that is returned:
 1. If the enabled I/O Command Set is the NVM Command Set or an I/O Command Set based on the NVM Command Set (e.g., the Zoned Namespace Command Set) issue the Identify command specifying the Identify Namespace data structure (CNS 00h); and
 2. Issue the Identify command specifying each of the following data structures (refer to Figure 274): the I/O Command Set specific Identify Namespace data structure, the I/O Command Set specific Identify Controller data structure, and the I/O Command Set independent Identify Namespace data structure;

...

Modify a portion of section 3.5.3 as shown below:

3.5.2 Message-based Transport Controller Initialization

...

- ~~8. The host determines any I/O Command Set specific configuration information as follows:~~
 - ~~a. If the CC.CSS field is set to 000b, then the host should determine the configuration of each namespace by issuing the Identify command for each namespace, specifying the Identify Namespace data structure (CNS 00h);~~
 - ~~b. If the CAP.CSS bit 6 is set to '1', then the host does the following:~~
 - ~~i. Issue the Identify command specifying the Identify I/O Command Set data structure (CNS 1Ch);~~
 - ~~ii. Issue the Set Features command with the I/O Command Set Profile Feature Identifier (FID 19h) specifying the index of the I/O Command Set Combination (refer to Figure 289) to be enabled;~~
 - ~~iii. For each I/O Command Set that is enabled:~~
 - ~~1. Issue the Identify command specifying the I/O Command Set specific Active Namespace ID list (CNS 07h) with the appropriate Command Set Identifier (CSI) value of that I/O Command Set;~~
 - ~~2. For each NSID that is returned:~~
 - ~~a. If the enabled I/O Command Set is the NVM Command Set or an I/O Command Set based on the NVM Command Set (e.g., the~~

~~Zoned Namespace Command Set) issue the Identify command specifying the Identify Namespace data structure (CNS 00h);~~
~~b. Issue the Identify command specifying each of the following data structures (refer to Figure 274): the I/O Command Set specific Identify Namespace data structure, the I/O Command Set specific Identify Controller data structure, and the I/O Command Set independent Identify Namespace data structure;~~

8. The host determines any I/O Command Set specific configuration information as follows:

- a. If the CAP.CSS bit 6 is set to '1', then the host does the following:
 - i. Issue the Identify command specifying the Identify I/O Command Set data structure (CNS 1Ch); and
 - ii. Issue the Set Features command with the I/O Command Set Profile Feature Identifier (FID 19h) specifying the index of the I/O Command Set Combination (refer to Figure 289) to be enabled;and
- b. For each I/O Command Set that is enabled (Note: the NVM Command Set is enabled if the CC.CSS field is set to 000b):
 - i. Issue the Identify command specifying the I/O Command Set specific Active Namespace ID list (CNS 07h) with the appropriate Command Set Identifier (CSI) value of that I/O Command Set; and
 - ii. For each NSID that is returned:
 1. If the enabled I/O Command Set is the NVM Command Set or an I/O Command Set based on the NVM Command Set (e.g., the Zoned Namespace Command Set) issue the Identify command specifying the Identify Namespace data structure (CNS 00h); and
 2. Issue the Identify command specifying each of the following data structures (refer to Figure 274): the I/O Command Set specific Identify Namespace data structure, the I/O Command Set specific Identify Controller data structure, and the I/O Command Set independent Identify Namespace data structure;

...

Modify a portion of section 4.1.2 as shown below:

4.1.2 Scatter Gather List (SGL)

...

The value of the SDT field shall be less than or equal to the value of the Maximum SGL Data Block Descriptors ~~field~~ (MSDBD) field in the Identify Controller data structure (refer to the NVMe over Fabrics specification for the definition of the MSDBD field).

...

Modify a portion of section 5.2 as shown below:

5 Admin Command Set

...

5.2 Asynchronous Event Request command

...

Host software may submit multiple Asynchronous Event Request commands to reduce event reporting latency. The total number of simultaneously outstanding Asynchronous Event Request commands is limited by the **value indicated in the Asynchronous Event Request Limit field specified** in the Identify Controller data structure in Figure 275.

Asynchronous events are grouped into event types. The event type is indicated in the Asynchronous Event Type field in Dword 0 of the completion queue entry for the Asynchronous Event Request command. When the controller posts a completion queue entry for an outstanding Asynchronous Event Request command and thus reports an asynchronous event, subsequent events of that event type are automatically masked by the controller until the host clears that event. Unless otherwise stated, an event is cleared by reading the log page associated with that event using the Get Log Page command (refer to section 5.16). If that log page is not accessible because media is not ready (i.e., the controller **aborts the Get Log Page command with returns** a status code of Admin Command Media Not Ready **for the Get Log Page command requesting the log page**), then the controller shall not post a completion queue entry for that asynchronous event until the controller is able to successfully return the log page that is required to be read to clear the asynchronous event.

The following event types are defined:

- a) **Error event:** Indicates a general error that is not associated with a specific command (refer to Figure 144). To clear this event, host software reads the Error Information log **page** (refer to section 5.16.1.2) using the Get Log Page command with the Retain Asynchronous Event bit cleared to '0';

...

Asynchronous events are reported due to a new entry being added to a log page (e.g., Error Information log **page**) or a status update (e.g., status in the SMART / Health log **page**). A status change may be permanent (e.g., the media has become read only) or transient (e.g., the temperature reached or exceeded a threshold for a period of time). Host software should modify the event threshold or mask the event for transient and permanent status changes before issuing another Asynchronous Event Request command to avoid repeated reporting of asynchronous events.

...

Modify a portions of section 5.16 as shown below:

5.16 Get Log Page command

...

There are mandatory and optional Log Identifiers defined in section 3.1.2.1.2, section 3.1.2.2.2, and section 3.1.2.3.3. If a Get Log Page command is processed that specifies a Log Identifier that is not supported, then the controller should **<editor: for the next major revision, this "should" is changing to "shall">** abort the command with a status code of **Invalid Field in Command Invalid Log Page** with the exception defined in **Figure 268**.

...

5.16.1 Log Specific Information

...

5.16.1.1 Supported Log Pages (Log Identifier 00h)

...

Figure 203: Supported Log Pages —Log Page

Bytes	Description
...	

...

Figure 204: ~~Get Log Page~~—LID Supported and Effects Data Structure

Bits	Description
...	

...

5.16.1.2 Error Information (Log Identifier 01h)

...

Figure 206: ~~Get Log Page~~— Error Information Log Entry Data Structure (~~Log Identifier 01h~~)

Bytes	Description
...	
23:16	LBA: This field indicates I/O Command Set specific data about the error condition, if applicable. The description is described in the applicable I/O Command Set specification. NOTE: This field applies to all User Data. The original field name has been retained for historical continuity.
...	

..

5.16.1.3 SMART / Health Information (Log Identifier 02h)

...

Figure 207: ~~Get Log Page~~—SMART / Health Information Log Page

Bytes	Description
...	
191:176	Number of Error Information Log Entries: Contains the number of Error Information Log Entries over the life of the controller.
...	

...

Figure 208: ~~Get Log Page~~—Temperature Sensor Data Structure

Bits	Description
...	

...

5.16.1.4 Firmware Slot Information (Log Identifier 03h)

...

Figure 209: ~~Get Log Page~~—Firmware Slot Information Log Page

Bytes	Description
...	

...

5.16.1.6 Commands Supported and Effects (Log Identifier 05h)

...

Figure 210: ~~Get Log Page~~—Commands Supported and Effects Log Page

Bytes	Description
...	

...

Figure 211: ~~Get Log Page~~—Commands Supported and Effects Data Structure

Bits	Description
...	

...

5.16.1.7 Device Self-test (Log Identifier 06h)

...

Figure 212: ~~Get Log Page~~—Device Self-test Log Page

Bytes	Description
...	

...

Figure 213: ~~Get Log Page~~—Self-test Result Data Structure

Bytes	Description
...	
23:16	Failing LBA¹ : This field is I/O Command Set specific and is described in the applicable I/O Command Set specification. NOTE: This field applies to all I/O Command Sets. The original field name has been retained for historical continuity.
...	
NOTE: 1. I/O Command Set specific value that is described in the applicable I/O Command Set specification.	

...

5.16.1.8 Telemetry Host-Initiated (Log Identifier 07h)

This log consists of a header describing the log and zero or more Telemetry Data Blocks (refer to section 8.24). All Telemetry Data Blocks are 512 bytes in size. The controller shall initiate a capture of the controller's internal controller state to this log if the controller processes a Get Log Page command for this log with the Create Telemetry Host-Initiated Data bit set to '1' in the Log Specific field. If the host specifies a Log Page Offset Lower value that is not a multiple of 512 bytes in the Get Log Page command for this

log, then the controller shall abort the command with a status code of Invalid Field in Command. This log page is global to the controller or global to the NVM subsystem.

The Log Specific Parameter field in Command Dword 10 (refer to Figure 197) for this log page is defined in Figure 214.

Figure 214: Telemetry Host-Initiated Command Dword 10 – Log Specific Parameter Field

Bits	Description
...	

...

Figure 215: Get Log Page – Telemetry Host-Initiated Log Page (Log Identifier 07h)

Bytes	Description
...	

...

5.16.1.9 Telemetry Controller-Initiated (Log Identifier 08h)

...

Figure 216: Get Log Page – Telemetry Controller-Initiated Log Page (Log Identifier 08h)

Bytes	Description
...	

...

5.16.1.10 Endurance Group Information (Log Identifier 09h)

...

Figure 217: Get Log Page – Endurance Group Information Log Page (Log Identifier 09h)

Bytes	Description
...	
159:144	Number of Error Information Log Entries: Contains the number of Error Information Log Entries over the life of the controller for the Endurance Group.
...	

...

5.16.1.11 Predictable Latency Per NVM Set (Log Identifier 0Ah)

...

Figure 218: Get Log Page – Predictable Latency Per NVM Set Log Page

Bytes	Description
...	

...

5.16.1.12 Predictable Latency Event Aggregate **Log Page** (Log Identifier 0Bh)

...

Figure 219: **Get Log Page**—Predictable Latency Event Aggregate Log Page

Bytes	Description
...	

...

5.16.1.13 Asymmetric Namespace Access (Log Identifier 0Ch)

...

If the host performs multiple Get Log Page commands to read the ANA log page (e.g., using the LPOL field or the LPOU field), the host should re-read the header of the log page and ensure that the Change Count field in the Asymmetric Namespace Access log matches the original value read. If it does not match, then the data captured is not consistent and the ANA log page should be re-read.

The Log Specific Parameter field in Command Dword 10 (refer to [Figure 197](#)) for this log page is defined in [Figure 220](#).

Figure 220: **Asymmetric Namespace Access Command Dword 10**— Log Specific **Parameter** Field

Bits	Description
...	

...

Figure 221: **Get Log Page**—Asymmetric Namespace Access —Log **Page**

Bytes	Description
...	

...

5.16.1.14 Persistent Event Log (Log Identifier 0Dh)

The Persistent Event Log page contains information about significant events not specific to a particular command. The information in this log page shall be retained across power cycles and resets. NVM subsystems should be designed for minimal loss of event information upon power failure. This log consists of a header describing the log and zero or more Persistent Events (refer to section 5.16.1.14.1).

This log page is global to the NVM subsystem.

The Log Specific Parameter field in Command Dword 10 (refer to [Figure 197](#)) for this log page is defined in [Figure 223](#).

...

Figure 223: **Persistent Event Command Dword 10**— Log Specific **Parameter** Field

Bits	Description
...	

...

Figure 224: ~~Get Log Page~~—Persistent Event Log Page (Log Identifier 0Dh)

Bytes	Description
...	

...

5.16.1.14.1 Persistent Event Log Events

...

5.16.1.14.1.13 Thermal Excursion Event (Event Type 0Dh)

...

Figure 243: Thermal Excursion Event Data Format (Event Type 0Dh)

Bytes	Description
0	Over Temperature: Contains the absolute value of the difference (i.e., delta) in Kelvins between the temperature indicated in the threshold field and temperature measured at the time of the event.
1	Threshold: Contains an indicator for the temperature threshold crossing that is being reported.
...	

...

5.16.1.15 Endurance Group Event Aggregate (Log Identifier 0Fh)

...

Figure 247: ~~Get Log Page~~—Endurance Group Event Aggregate Log Page

Bytes	Description
...	

...

5.16.1.16 Media Unit Status (Log Identifier 10h)

...

<Editor – Move the Figure title into the table as the 1st row of the header>

Figure 248: ~~Get Log Page~~—Media Unit Status Log Page

Bytes	Description
...	

...

<Editor – Move the Figure title into the table as the 1st row of the header>

Figure 249: Media Unit Status Descriptor

Bytes	Description
...	

...

5.16.1.17 Supported Capacity Configuration List (Log Identifier 11h)

...

<Editor – Move the Figure title into the table as the 1st row of the header>

Figure 250: ~~Get Log Page~~—Supported Capacity Configuration List ~~Log Page~~

Bytes	Description
...	

...

5.16.1.18 Feature Identifiers Supported and Effects (Log Identifier 12h)

...

Figure 255: ~~Get Log Page~~—Feature Identifiers Effects ~~Log Page~~

Bytes	Description
...	

...

Figure 256: ~~Get Log Page~~—FID Supported and Effects Data Structure

Bits	Description
...	

...

5.16.1.19 NVMe-MI Commands Supported and Effects (Log Identifier 13h)

...

Figure 257: ~~Get Log Page~~—NVMe-MI Commands Supported and Effects ~~Log Page~~

Bytes	Description
...	

...

Figure 258: ~~Get Log Page~~—NVMe-MI Commands Supported and Effects Data Structure

Bits	Description
...	

...

5.16.1.20 Command and Feature Lockdown (Log Identifier 14h)

This log page is used to indicate which commands and Set Features Feature Identifiers are supported to be prohibited from execution using the Command and Feature Lockdown capability (refer to section 8.4) and which commands are currently prohibited if received on an NVM Express controller Admin Submission Queue or received out-of-band on a Management Endpoint (refer to the NVM Express Management Interface Specification). This log page uses the Log Specific ~~Parameter Field~~ field in ~~Command Dword 10 (refer to Figure 197) as defined in (refer to Figure 259). This log page -and-~~ may use the UUID Index field in the Get Log Page command to specify the scope and content of the list returned in the Command and Feature Identifier List field of this log page. The UUID Index field may be used if the Scope field is set to 2h, allowing returning of vendor specific Set Features Feature Identifier lockdown information.

Figure 259: **Command and Feature Lockdown Command Dword 10** – Log Specific **Parameter** Field

Bits	Description
...	

...

<Editor – Move the Figure title into the table as the 1st row of the header>

Figure 260: **Get Log Page** – Command and Feature Lockdown Log **Page**

Bytes	Description
...	

...

5.16.1.21 Boot Partition (Log Identifier 15h)

...

A host reading this log page has no effects on the BPINFO (refer to section 3.1.3.13), BPRSEL, and BPMBL (refer to section 3.1.3.15) registers.

The Log Specific Parameter field in Command Dword 10 (refer to **Figure 197**) for this log page is defined in **Figure 261**.

Figure 261: **Boot Partition Command Dword 10** – Log Specific **Parameter** Field

Bits	Description
...	

...

Figure 262: **Get Log Page** – Boot Partition Log **Page** (~~Log Identifier 15h~~)

Bytes	Description
Boot Partition Header	
...	

...

5.16.1.22 Rotational Media Information Log (Log Identifier 16h)

..

Figure 263: **Get Log Page** – Rotational Media Information Log **Page**

Bytes	Description
...	

...

5.16.1.23 Discovery Log Page (Log Identifier 70h)

...

Figure 264: ~~Get Log Page~~—Discovery Log Page Entry Data Structure

Bytes	Description
...	

...

Figure 265: ~~Get Log Page~~—Discovery Log Page

Bytes	Description
...	

...

5.16.1.24 Reservation Notification (Log Identifier 80h)

...

Figure 266: ~~Get Log Page~~—Reservation Notification Log Page

Bytes	Description
...	

...

5.16.1.25 Sanitize Status (Log Identifier 81h)

...

Figure 267: ~~Get Log Page~~—Sanitize Status Log Page

Bytes	Description
...	

...

Modify a portion of section 5.16.2 as shown below:

5.16.2 Command Completion

...

Figure 268: Get Log Page – Command Specific Status Values

Value	Description
9h	Invalid Log Page: The log page indicated is invalid or not supported. This error condition is also returned if a reserved log page is requested. Controllers compliant with NVM Express Base Specification revision 4.3 2.0 and earlier may return Invalid Field in Command for this condition.
29h	I/O Command Set Not Supported: The specified I/O Command Set is not supported by the controller.

...

Modify a portion of section 5.17.2.1 as shown below:

5.17 Identify command

...

5.17.2 Identify Data Structures

...

5.17.2.1 Identify Controller Data Structure (CNS 01h)

...

Figure 275: Identify – Identify Controller Data Structure, I/O Command Set Independent

Bytes	I/O ¹	Admin ¹	Disc ¹	Description																	
...																					
262	M	M	M	Error Log Page Entries (ELPE): This field indicates the maximum number of Error Information Log Entries that are stored by the controller. This field is a 0's based value.																	
...																					
331:328	O	O	R	Sanitize Capabilities (SANICAP): This field indicates attributes for sanitize operations. If the Sanitize command is supported, then this field shall be non-zero. If the Sanitize command is not supported, then this field shall be cleared to 0h. Refer to section 8.21.																	
				Bits	Description	31:30	No-Deallocate Modifies Media After Sanitize (NODMMAS): This field indicates if media is additionally modified by the controller after a sanitize operation successfully completes that had been started by a Sanitize command with the No-Deallocate After Sanitize bit set to '1'. The work required for the associated additional media modification is included both in the estimated time for each sanitize operation and in the Sanitize Progress field (refer to Figure 267).	Value	Definition	00b	Additional media modification after sanitize operation completes successfully is not defined. Only controllers compliant with NVM Express Base Specification revision 1.3 and earlier or that have bits 2:0 of the SANICAP field cleared to 0h shall be allowed to return this value.	01b	Media is not additionally modified by the NVMe controller after sanitize operation completes successfully.	10b	Media is additionally modified by the NVMe controller after sanitize operation completes successfully. The Sanitize Operation Completed event does not occur until the additional media modification associated with this field has completed.	11b	Reserved	If bits 2:0 of the SANICAP field are cleared to 000b, then the controller shall clear this field to 00b.		...	
				Bits	Description																
				31:30	No-Deallocate Modifies Media After Sanitize (NODMMAS): This field indicates if media is additionally modified by the controller after a sanitize operation successfully completes that had been started by a Sanitize command with the No-Deallocate After Sanitize bit set to '1'. The work required for the associated additional media modification is included both in the estimated time for each sanitize operation and in the Sanitize Progress field (refer to Figure 267).																
					Value		Definition	00b	Additional media modification after sanitize operation completes successfully is not defined. Only controllers compliant with NVM Express Base Specification revision 1.3 and earlier or that have bits 2:0 of the SANICAP field cleared to 0h shall be allowed to return this value.	01b	Media is not additionally modified by the NVMe controller after sanitize operation completes successfully.	10b	Media is additionally modified by the NVMe controller after sanitize operation completes successfully. The Sanitize Operation Completed event does not occur until the additional media modification associated with this field has completed.	11b	Reserved						
Value	Definition																				
00b	Additional media modification after sanitize operation completes successfully is not defined. Only controllers compliant with NVM Express Base Specification revision 1.3 and earlier or that have bits 2:0 of the SANICAP field cleared to 0h shall be allowed to return this value.																				
01b	Media is not additionally modified by the NVMe controller after sanitize operation completes successfully.																				
10b	Media is additionally modified by the NVMe controller after sanitize operation completes successfully. The Sanitize Operation Completed event does not occur until the additional media modification associated with this field has completed.																				
11b	Reserved																				
If bits 2:0 of the SANICAP field are cleared to 000b, then the controller shall clear this field to 00b.																					
...																					
...																					

...

Modify a portion of section 5.17.2.3 as shown below:

5.17.2.3 Namespace Identification Descriptor list (CNS 03h)

A list of Namespace Identification Descriptor structures (refer to Figure 277) is returned to the host for the namespace specified in the Namespace Identifier (NSID) field if it is an active NSID. Namespace Identification Descriptor structures consist of one or more Namespace Identifiers (NID) of various types as indicated by the Namespace Identifier Type (NIDT) field in each descriptor. Each NID is assigned to a namespace at namespace creation and remains fixed throughout the life of that namespace. If the NSID field does not specify an active NSID, then refer to section 3.2.1.5 for the status code to return.

The contents of the Namespace Identification Descriptor list is preserved across namespace and controller operations (e.g., Controller Level Reset, namespace format, etc.).

The controller may return any number of variable length Namespace Identification Descriptor structures that fit into the 4,096 byte Identify payload. All remaining bytes after the Namespace Identification Descriptor structures should be cleared to 0h, and the host shall interpret a Namespace Identifier Descriptor Length (NIDL) value of 0h as the end of the list. The host should ignore any Namespace Identification Descriptor with a Namespace Identifier Type not supported by the host.

A controller shall not return multiple Namespace Identification Descriptors with the same Namespace Identifier Type (NIDT). A controller shall return:

- at least one Namespace Identification Descriptor identifying the namespace (i.e., NIDT field set to 1h, ~~NIDT=2h~~, or ~~NIDT=3h~~); and
- a Namespace Identification Descriptor identifying the I/O Command Set (i.e., NIDT field set to 4h) ~~if CAP.CSS bit 6 is set to '1', then the Command Set Identifier NID (i.e., NIDT=4) shall be returned.~~

Figure 277: Identify – Namespace Identification Descriptor

Bytes	Description																					
00	Namespace Identifier Type (NIDT): This field indicates the data type contained in the Namespace Identifier field and the length of that type as defined in the following table.																					
	<table><tr><th>Value</th><th>Length (NIDL)</th><th>Definition</th></tr><tr><td>0h</td><td></td><td>Reserved</td></tr><tr><td colspan="3">...</td></tr><tr><td>2h</td><td>10h</td><td>Namespace Globally Unique Identifier: The NID field contains a copy of the NGUID field in the Identify Namespace data structure (refer to the applicable I/O Command Set specification). If the NGUID field of the Identify Namespace data structure is not supported (i.e., the NGUID field is cleared to 0h), the controller shall not report a Namespace Identification Descriptor with a value of type 2h.</td></tr><tr><td>3h</td><td>10h</td><td>Namespace UUID: The NID field contains a 128-bit Universally Unique Identifier (UUID) as specified in RFC 4122. Refer to section 4.3.6. If the namespace does not support an IEEE Extended Unique Identifier (i.e., EUI64 field is cleared to 0h) and does not support a Namespace Globally Unique Identifier (i.e., the NGUID field is cleared to 0h), then the namespace shall report a Namespace Identification Descriptor with a value of type 3h.</td></tr><tr><td>4h</td><td>1h</td><td>Command Set Identifier (CSI): The NID field contains the I/O Command Set that operates on this namespace. Refer to Figure 274.</td></tr><tr><td>5h to FFh</td><td></td><td>Reserved</td></tr></table>	Value	Length (NIDL)	Definition	0h		Reserved	...			2h	10h	Namespace Globally Unique Identifier: The NID field contains a copy of the NGUID field in the Identify Namespace data structure (refer to the applicable I/O Command Set specification). If the NGUID field of the Identify Namespace data structure is not supported (i.e., the NGUID field is cleared to 0h), the controller shall not report a Namespace Identification Descriptor with a value of type 2h.	3h	10h	Namespace UUID: The NID field contains a 128-bit Universally Unique Identifier (UUID) as specified in RFC 4122. Refer to section 4.3.6. If the namespace does not support an IEEE Extended Unique Identifier (i.e., EUI64 field is cleared to 0h) and does not support a Namespace Globally Unique Identifier (i.e., the NGUID field is cleared to 0h), then the namespace shall report a Namespace Identification Descriptor with a value of type 3h.	4h	1h	Command Set Identifier (CSI): The NID field contains the I/O Command Set that operates on this namespace. Refer to Figure 274.	5h to FFh		Reserved
	Value	Length (NIDL)	Definition																			
	0h		Reserved																			
	...																					
	2h	10h	Namespace Globally Unique Identifier: The NID field contains a copy of the NGUID field in the Identify Namespace data structure (refer to the applicable I/O Command Set specification). If the NGUID field of the Identify Namespace data structure is not supported (i.e., the NGUID field is cleared to 0h), the controller shall not report a Namespace Identification Descriptor with a value of type 2h.																			
	3h	10h	Namespace UUID: The NID field contains a 128-bit Universally Unique Identifier (UUID) as specified in RFC 4122. Refer to section 4.3.6. If the namespace does not support an IEEE Extended Unique Identifier (i.e., EUI64 field is cleared to 0h) and does not support a Namespace Globally Unique Identifier (i.e., the NGUID field is cleared to 0h), then the namespace shall report a Namespace Identification Descriptor with a value of type 3h.																			
	4h	1h	Command Set Identifier (CSI): The NID field contains the I/O Command Set that operates on this namespace. Refer to Figure 274.																			
5h to FFh		Reserved																				
01	Namespace Identifier Length (NIDL): This field contains the length in bytes of the Namespace Identifier (NID) field. The total length of the Namespace Identification Descriptor in bytes is the value in this field plus four. If this field is cleared to 0h it indicates the end of the Namespace Identifier Descriptor list.																					
02:03	Reserved																					

Figure 277: Identify – Namespace Identification Descriptor

Bytes	Description
(NIDL + 3):04	Namespace Identifier (NID): For an NIDT field value of 1h, 2h, and 3h, this field contains a value that is globally unique and assigned to the namespace when the namespace is created. This field remains fixed throughout the life of the namespace and is preserved across namespace and controller operations (e.g., Controller Level Reset, namespace format, etc.). The type of the value is specified by the Namespace Identifier Type (NIDT) field, and the size is specified by the Namespace Identifier Length (NIDL) field.

...

Modify a portion of section 5.17.2.5 as shown below:

5.17.2.5 I/O Command Set specific Identify Namespace data structure (CNS 05h)

...

If the controller supports the Namespace Management capability (refer to section 8.11), the I/O Command Set requested in the CSI field ~~is has been enabled with the I/O Command Set Profile feature (refer to CC.CSS in Figure 46 section 5.27.1.21),~~ and the NSID field is set to FFFFFFFFh, then the controller returns an I/O Command Set specific Identify Namespaces data structure that specifies capabilities that are common across namespaces for the I/O Command Set specified in the CSI field (refer to Figure 274).

<Editor: Make the follow sentence to be in a separate paragraph>

If the controller does not support the Namespace Management capability and the NSID field is set to FFFFFFFFh, then the controller shall abort the command with a status code of Invalid Namespace or Format.

...

Modify a portion of section 5.23 as shown below:

5.23 Namespace Management command

...

5.23.1 Command Completion

...

Figure 301: Namespace Management – Command Specific Status Values

Value	Description
...	
15h	Namespace Insufficient Capacity: Creating the namespace requires more unallocated capacity than is currently available. The Command Specific Information field of the Error Information Log page specifies the total amount of unallocated NVM capacity required to create the namespace in bytes.
...	

...

Modify a portion of section 8.2 as shown below:

8 Extended Capabilities

...

8.2 Boot Partitions

...

An NVMe controller that supports Boot Partitions has two Boot Partitions of equal size using Boot Partition identifiers 0h and 1h. The two Boot Partitions allow the host to update one and verify the contents before marking the Boot Partition active. Controllers in the NVM subsystem may share the same Boot Partitions.

...

Modify a portion of section 8.12 as shown below:

8.12 Namespace Write Protection

...

Figure 429 defines the write protection states that may be supported for a namespace. All states persist across power cycles and NVMe Controller Level Resets (refer to section 3.7.2) except Write Protect Until Power Cycle state, which transitions to the No Write Protect state on the occurrence of a power cycle.

...

Modify a portion of section 8.13.4.1 as shown below:

8.13 NVMe over Fabrics Secure Channel and In-band Authentication

8.13.4 Common Authentication Messages

8.13.4.1 AUTH_Negotiate Message

Figure 438: Secure channel protocol identifiers

Value	Description	Transport Applicability
00h	No secure channel concatenation	N/A n/a
...		

...

Modify a portion of section 8.13.5.1 as shown below:

8.13.5 DH-HMAC-CHAP Protocol

8.13.5.1 Protocol Operations

Figure 447: DH-HMAC-CHAP Diffie-Hellman group identifiers

Identifier	DH group size	Generator (g)	Modulus (p) and Reference
00h	NULL	N/A n/a	N/A n/a
...			

Description of NVM Express NVM Command Set specification 1.0b changes

Modify any portion of section as shown below:

<Editor: Globally change the following “NVM~~e~~ Express Base Specification”>

Modify a portion of section 2.1.1 as shown below:

2 NVM Command Set Model

...

2.1 Theory of operation

...

2.1.1 Namespaces

...

A logical block shall be marked as allocated when it is written with:

- a User Data Out Command ~~Write-command~~;
- a Write Uncorrectable command; or
- a Write Zeroes command that does not deallocate the logical block (refer to section 3.2.3.2.1).

A logical block may be marked as allocated as the result of:

- a User Data Out Command ~~Write-command~~ not addressing the logical block (e.g., NPWG field may indicate sequential logical blocks placed and tracked together on the media (refer to section 5.8.2.1)); ~~or~~
- a Write Zeroes command not addressing the logical block (refer to section 3.2.3.2.1); or
- a sanitize operation.

A logical block may be marked deallocated as the result of:

- a Dataset Management command (refer to section 3.2.3); ~~or~~
- a Write Zeroes command addressing the logical block (refer to section 3.2.3.2.1); or
- a sanitize operation.

Vendor specific means are able to allocate or deallocate logical blocks.

...

Modify a portion of section 3.2.2 as shown below:

3 I/O Commands for the NVM Command Set

...

3.2 NVM Command Set Commands

...

3.2.2 Copy command

...

<The reserved at the end of the table is not correct as the protocol passes complete SREs>

Figure 35: Copy – Source Range Entries Descriptor Format 1h

Range	Bytes	Description
	15:08	Starting LBA
	...	
		...
Source Range 1	...	
		...
Source Range 101	...	
	4095:4080	Reserved

...

Modify a portion of section 4.1.4.1 as shown below:

4 Admin Commands for the NVM Command Set

...

4.1 Admin Command behavior for the NVM Command Set

...

4.1.4.1 Error Information (Log Identifier 01h)

The Error Information Log Page is as defined in the NVMe Base Specification. Figure 90 describes the NVM Command Set specific definition of the ~~LBA User-Data~~ field.

Figure 90: ~~Get-Log-Page~~—Error Information Log Entry Data Structure – User Data

Bytes	Description
23:16	LBA: This field indicates the first LBA that experienced the error condition, if applicable.

...

Modify a portion of section 4.1.4.3 as shown below:

4.1.4.3 Device Self-test (Log Identifier 06h)

The Device Self-test Log Page is as defined in the NVMe Base Specification. Figure 91 describes the NVM Command Set specific definition of the Failing ~~LBA User-Data-Reference~~ field.

Figure 91: ~~Get-Log-Page~~—Device Self-test Result Data Structure Log

Bytes	Description
...	

...

Modify a portion of section 4.1.4.5 as shown below:

4.1.4.5 LBA Status Information (Log Identifier 0Eh)

...

Figure 93: LBA Status Information Log Page

Bytes	Description
03:00	LBA Status Log Page Length (LSLPLEN): This field indicates the length in bytes of the LBA Status Information log page.
...	

...

Modify a portion of section 4.1.5 as shown below:

4.1.5 Identify Command

...

Figure 96: CNS Values

CNS Value	O/M ¹	Definition	NSID ²	CNTID ³	CSI ⁴	Reference Section
Active Namespace Management						
00h	M	Identify I/O Command Set Specific Namespace data structure for the controller processing the command. ⁶ Identify Namespace data structure for the specified NSID or the namespace capabilities for the NVM Command Set. ⁶	Y	N	Y	4.1.5.1
01h	M	Identify Controller data structure for the controller processing the command. ⁶	N	N	N	4.1.5.2
05h	M ⁵	Identify I/O Command Set specific Namespace data structure for the specified NSID for the I/O Command Set specified in the CSI field. ⁶	Y	N	Y	4.1.5.3

...

Modify a portion of section 4.1.5.1 as shown below:

...

4.1.5.1 NVM Command Set Identify Namespace Data Structure (CNS 00h)

...

Figure 98: LBA Format Data Structure, NVM Command Set Specific

Bits	Description
...	
23:16	LBA Data Size (LBADS): This field indicates the LBA data size supported. The value is reported in terms of a power of two (2^n). A non-zero value less smaller than 9 (i.e., 512 bytes) is not supported. If the value reported is 0h, then the LBA format is not supported / used or is not currently available (refer to the Number of LBA Formats field in Figure 97).
...	

Description of NVM Express Zoned Namespace Command Set specification 1.1b changes

Modify any portion of section as shown below:

<Editor: Globally change the following “NVM Express Base Specification”>

Modify a portion of section 4.1.4.1 as shown below:

4 Admin Commands for the Zoned Namespace Command Set

...

4.1 Admin Command behavior for the Zoned Namespace Command Set

...

4.1.4 Get Log Page Command

...

4.1.4.1 Changed Zone List (Log Identifier BFh)

...

The Changed Zone List log page is 4,096 bytes in size. This log page ~~it~~ contains a Zone Identifier List with up to 511 Zone Identifier entries. The Changed Zone ~~Identifier~~ List log page is defined in Figure 46.

...

Figure 46: Changed Zone ~~Identifier~~ List Log Page Data Structure

Bytes	Description
...	

...

Modify a portion of section 4.1.7 as shown below:

4.1.7 Sanitize command

...

Figure 51: Sanitize Behavior for the Zoned Namespace Command Set

No Deallocate After Sanitize	No-Deallocate Modifies Media After Sanitize (NODMMAS)	No-Deallocate Inhibited (NDI)	No-Deallocate Response Mode (NODRM)	Results of a successful sanitize operation		
				Zone State ¹	Logical Block Content ²	Sanitize Status ⁴
0b	N/A n/a ³	N/A n/a ³	N/A n/a ³	ZSE:Empty	Refer to section 2.1.1.2.1.2	001b

Technical input submitted to the NVM Express® Workgroup is subject to the terms of the NVM Express® Participant’s agreement. Copyright © 2008 to 2022 NVM Express, Inc.

Figure 51: Sanitize Behavior for the Zoned Namespace Command Set

No Deallocate After Sanitize	No-Deallocate Modifies Media After Sanitize (NODMMAS)	No-Deallocate Inhibited (NDI)	No-Deallocate Response Mode (NODRM)	Results of a successful sanitize operation		
				Zone State ¹	Logical Block Content ²	Sanitize Status ⁴
1b	00b	0b	N/A n/a ³	ZSE:Empty	Refer to section 2.1.1.2.1.2	001b
1b	00b	1b	0b			N/A n/a ⁵
1b	00b	1b	1b	ZSE:Empty	Refer to section 2.1.1.2.1.2	001b
1b	01b (does not modify)	0b (not inhibited)	N/A n/a ³	ZSF:Full	Block Erase: not defined in this specification Crypto Erase: not defined in this specification Overwrite: Shall be the overwrite pattern specified in the Sanitize command that requested the sanitize operation	001b
1b	01b	1b (inhibited)	1b	ZSE:Empty	Refer to section 2.1.1.2.1.2	100b
1b	01b	1b	0b			N/A n/a ⁵
1b	10b (does modify)	0b	N/A n/a ³	ZSF:Full	Block Erase: not defined in this specification Crypto Erase: not defined in this specification Overwrite: Overwrite: Shall be the overwrite pattern specified in the Sanitize command that requested the sanitize operation	001b
1b	10b	1b	1b	ZSE:Empty	Refer to section 2.1.1.2.1.2	100b
1b	10b	1b	0b			N/A n/a ⁵
N/A n/a ³	11b	N/A n/a ³	N/A n/a ³	Reserved		
NOTES:						
1. ZSO:Offline state is a valid zone state as a result of a successful sanitize operation.						
2. This field describes the read value from a deallocated logical block. Refer to the Deallocated or Unwritten Logical Blocks section in the NVM Command Set Specification.						
3. This value is not relevant in the setup conditions defined in that row.						
4. Value reported in bits 2:0 of the Sanitize Status (SSTAT) field in the Sanitize Status (Log Identifier 81h) (refer to the NVMe Base Specification).						
5. Sanitize command is aborted with a status code of Invalid Field in Command.						

...

Description of NVM Express Key Value Command Set specification

1.0b changes

Modify any portion of section as shown below:

<Editor: Globally change the following “NVM Express Base Specification”>

Modify a portion of section 4.1.4 as shown below:

4 Admin Commands for the Key Value Command Set

...

4.1 Admin Command behavior for the Key Value Command Set

...

4.1.4 Get Log Page command

The Get Log Page command operates as defined in the NVMe Base Specification. In addition to the requirements in the NVM Express Base Specification, mandatory, optional, and prohibited Log Identifiers are defined in Figure NEW. If a Get Log Page command is processed that specifies a Log Identifier that is not supported, then the controller should abort the command with a status code of Invalid Field in Command.

Log page scope is as defined in the NVMe Base Specification.

The rules for namespace identifier usage are specified in the NVMe Base Specification.

Figure NEW: Get Log Page – Log Page Identifiers

Log Identifier	Scope and Support	Log Page Name	Reference
01h	Refer to the NVM Express Base Specification	Error Information	4.1.4.TBD1
06h	Refer to the NVM Express Base Specification	Device Self-test	4.1.4.TBD2

4.1.4.TBD1 Error Information (Log Identifier 01h)

The Error Information log page is as defined in the NVM Express Base Specification. Figure NEW-1 describes the Key Value Command Set specific definition of the LBA field.

Figure NEW-1: Error Information Log Entry Data Structure

Bytes	Description
23:16	LBA: This field is reserved.

4.1.4.TBD2 Device Self-test (Log Identifier 06h)

The Device Self-test log page is as defined in the NVM Express Base Specification. Figure NEW-2 describes the Key Value Command Set specific definition of the Failing LBA field.

Figure NEW-2: Self-test Result Data Structure

Bytes	Description
23:16	Failing LBA: This field is reserved.

...

Description of NVM Express TCP Transport Specification 1.0b changes

Modify any portion of section as shown below:

<Editor: Globally change the following “NVM~~e~~ Express Base Specification”>

Modify a portion of section 3.3.2.1 as shown below:

3 Transport Binding

...

3.3 Data Transfer Model

...

3.3.2 Command Data Buffers and SGLs

...

3.3.2.1 Controller to Host Command Data Buffer Transfers

...

Figure 15 illustrates a command that performs a 12,288 bytes (3000h bytes) controller to host Command Data Buffer transfer:

1. The host sends a Command Capsule PDU (CapsuleCmd) to the controller containing an SQE with a Transport SGL Data Block descriptor with a Sub Type value of ~~4h~~Ah in SGL1. The Length field in the descriptor has a value of 3000h;
2. The controller processes the command and sends an C2HData PDU that transfers 1,000 bytes (3E8h bytes). The Data Offset (DATAO) field is cleared to 0h, the Data Length (DATAL) field is set to 3E8h, and the Last (LAST_PDU) flag is cleared to '0' since this is not the last PDU of the data transfer;
3. The controller sends a subsequent C2HData PDU that transfers 2,000 bytes (7D0h bytes). The DATAO field is set to 3E8h, the DATAL field is set to 7D0h, and the LAST_PDU flag is cleared to '0' since this is not the last PDU of the data transfer;
4. The controller sends a subsequent C2HData PDU that transfers 8,192 bytes (2000h bytes). The DATAO field is set to BB8h, the DATAL field is set to 2000h, and the LAST_PDU flag is cleared to '0' since this is not the last PDU of the data transfer;
5. The controller sends a subsequent C2HData PDU that transfers 1,096 bytes (448h bytes). The DATAO field is set to 2BB8h, the DATAL field is set to 448h, and the LAST_PDU flag is set to '1' since this is the last PDU of the data transfer; and
6. The controller sends a Response Capsule PDU (CapsuleResp) to the host containing a CQE.

...

Description of NVM Express PCIe Transport Specification 1.0b changes

Modify any portion of section as shown below:

<Editor: Globally change the following “NVM Express Base Specification”>

Description of NVM Express RDMA Transport Specification 1.0a changes

Modify any portion of section as shown below:

<Editor: Globally change the following “NVM~~e~~ Express Base Specification”>