



**LEGAL NOTICE:**

© **Copyright 2007 - 2016 NVM Express, Inc. ALL RIGHTS RESERVED.**

This NVM Express revision 1.2 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 NVM Express revision 1.2 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

*Technical input submitted to the NVM Express™ Workgroup is subject to the terms of the NVM Express™ Participant’s agreement. Copyright © 2014-15 NVMe™ Corporation.*

respective owners.

NVM Express Workgroup  
c/o Virtual, Inc.  
401 Edgewater Place, Suite 600  
Wakefield, MA 01880  
[info@nvmexpress.org](mailto:info@nvmexpress.org)

## NVM Express™ Technical Errata

<b>Errata ID</b>	<b>007</b>
<b>Revision Date</b>	<b>1/17/2016</b>
<b>Affected Spec Ver.</b>	<b>NVM Express™ 1.2</b>
<b>Corrected Spec Ver.</b>	

### Errata Author(s)

Name	Company
John Carroll	Intel
Judy Brock	Samsung
Dave Landsman	SanDisk
Peter Onufryk	PMC
Austin Bolen	Dell
Amber Huffman	Intel
Ramdas Kachare	Seagate

### Errata Overview

Added clarifications to power management on how to handle an invalid power state, units in arbitration burst, admin command behavior when I/O queues are full, clarified firmware version is consistent across an NVM subsystem, requirement to initialize IO queue entry size, unique identifiers, and editorial changes.

Added missing capability bits for SGL enhancements and FW Activation Notices added in NVMe 1.2.

## Revision History

Revision Date	Change Description
10/01/2015	First draft (first two items pushed from ECN 006)
10/08/2015	Added Namespace and firmware commit clarifications. Replaced figure 103/139 with updates from 1.2a. Included feedback from 10/1 meeting.
10/22/2015	Added clarification on IOCQES/IOSQES. Made changes based on 10/08 workgroup feedback (remove read-only edit, add ^ operator). Added OAES bit for Firmware Activation Notices.
11/05/2015	Added NGUID Uniqueness clarification. Rephrased IOCQES/IOSQES text based on feedback from the workgroup.
11/11/2015	Removed NGUID Uniqueness clarification, changes to recommended error codes.
1/17/2016	Ratified.

## Description of Specification Changes

### ***Modify a portion of section 5.14.1.2 (Power Management) as shown below:***

This Feature allows the host to configure the power state. The attributes are indicated in Command Dword 11.

After a successful completion of a Set Features command for this feature, the controller shall be in the Power State specified.

If a Get Features command is submitted for this Feature, the attributes specified in Figure are returned in Dword 0 of the completion queue entry for that command.

**Figure 111: Power Management – Command Dword 11**

Bit	Description
31:08	Reserved
07:05	<b>Workload Hint (WH):</b> This field indicates the type of workload expected. This hint may be used by the NVM subsystem to optimize performance. Refer to section 8.4.3 for more details.
04:00	<b>Power State (PS):</b> This field indicates the new power state into which the controller should transition. This power state shall be one supported by the controller as indicated in the Number of Power States Supported (NPSS) field in the Identify Controller data structure. If the power state specified is not supported, the controller shall abort the command and should return an error of Invalid Field in Command.

### ***Modify a portion of Figure 90 (Identify Controller) as shown below:***

Bytes	O/M	Description																
Controller Capabilities and Features																		
539:536	O	<b>SGL Support (SGLS):</b> This field indicates if SGLs are supported for the NVM Command Set and the particular SGL types supported. Refer to section 4.4.																
		<table><tr><th>Bits</th><th>Description</th></tr><tr><td>31:19</td><td>Reserved</td></tr><tr><td>19</td><td>If set to '1', then use of a Metadata Pointer (MPTR) that contains an address of an SGL segment containing exactly one SGL Descriptor that is Qword aligned is supported. If cleared to '0', then use of a MPTR containing an SGL Descriptor is not supported.</td></tr><tr><td>18</td><td>If set to '1', then the controller supports commands that contain a data or metadata SGL of a length larger than the amount of data to be transferred. If cleared to '0', then the SGL length shall be equal to the amount of data to be transferred.</td></tr><tr><td>17</td><td>If set to '1', then use of a byte aligned contiguous physical buffer of metadata (the Metadata Pointer field in Figure 12) is supported. If cleared to '0', then use of a byte aligned contiguous physical buffer of metadata is not supported.</td></tr><tr><td>16</td><td>If set to '1', then the SGL Bit Bucket descriptor is supported. If cleared to '0', then the SGL Bit Bucket descriptor is not supported.</td></tr><tr><td>15:01</td><td>Reserved</td></tr><tr><td>00</td><td>If set to '1', then the controller supports SGLs for the NVM Command Set including the SGL Data Block, SGL Segment, and SGL Last Segment descriptor types. If cleared to '0', then the controller does not support SGLs for the NVM Command Set and all other bits in this field shall be cleared to '0'.</td></tr></table>	Bits	Description	31:19	Reserved	19	If set to '1', then use of a Metadata Pointer (MPTR) that contains an address of an SGL segment containing exactly one SGL Descriptor that is Qword aligned is supported. If cleared to '0', then use of a MPTR containing an SGL Descriptor is not supported.	18	If set to '1', then the controller supports commands that contain a data or metadata SGL of a length larger than the amount of data to be transferred. If cleared to '0', then the SGL length shall be equal to the amount of data to be transferred.	17	If set to '1', then use of a byte aligned contiguous physical buffer of metadata (the Metadata Pointer field in Figure 12) is supported. If cleared to '0', then use of a byte aligned contiguous physical buffer of metadata is not supported.	16	If set to '1', then the SGL Bit Bucket descriptor is supported. If cleared to '0', then the SGL Bit Bucket descriptor is not supported.	15:01	Reserved	00	If set to '1', then the controller supports SGLs for the NVM Command Set including the SGL Data Block, SGL Segment, and SGL Last Segment descriptor types. If cleared to '0', then the controller does not support SGLs for the NVM Command Set and all other bits in this field shall be cleared to '0'.
		Bits	Description															
		31:19	Reserved															
		19	If set to '1', then use of a Metadata Pointer (MPTR) that contains an address of an SGL segment containing exactly one SGL Descriptor that is Qword aligned is supported. If cleared to '0', then use of a MPTR containing an SGL Descriptor is not supported.															
		18	If set to '1', then the controller supports commands that contain a data or metadata SGL of a length larger than the amount of data to be transferred. If cleared to '0', then the SGL length shall be equal to the amount of data to be transferred.															
		17	If set to '1', then use of a byte aligned contiguous physical buffer of metadata (the Metadata Pointer field in Figure 12) is supported. If cleared to '0', then use of a byte aligned contiguous physical buffer of metadata is not supported.															
		16	If set to '1', then the SGL Bit Bucket descriptor is supported. If cleared to '0', then the SGL Bit Bucket descriptor is not supported.															
		15:01	Reserved															
00	If set to '1', then the controller supports SGLs for the NVM Command Set including the SGL Data Block, SGL Segment, and SGL Last Segment descriptor types. If cleared to '0', then the controller does not support SGLs for the NVM Command Set and all other bits in this field shall be cleared to '0'.																	

**Modify a portion of Figure 90 (Identify Controller) as shown below:**

Bytes	O/M	Description
72	M	<b>Recommended Arbitration Burst (RAB):</b> This is the recommended Arbitration Burst size. The value is in commands and is reported as a power of two ( $2^n$ ). This is the same units as the Arbitration Burst size. Refer to section 4.11.

**Modify a portion of Section 1.5 (Conventions) as shown below:**

When a size is stated in the document as KB, the convention used is 1KB = 1024 bytes.

The ^ operator is used to denote the power to which that number, symbol, or expression is to be raised.

Some parameters are defined as a string of ASCII characters. ASCII data fields shall contain only code values 20h through 7Eh. For the string "Copyright", the character "C" is the first byte, the character "o" is the second byte, etc. The string is left justified and shall be padded with spaces (ASCII character 20h) to the right if necessary.

**Modify a portion of section 5 (Admin Command Set) as shown below:**

The Admin Command Set defines the commands that may be submitted to the Admin Submission Queue.

The Submission Queue Entry (SQE) structure and the fields that are common to all Admin commands are defined in section 4.2. The Completion Queue Entry (CQE) structure and the fields that are common to all Admin commands are defined in section 4.6. The command specific fields in the SQE and CQE structures for the Admin Command Set are defined in this section.

For all Admin commands, Dword 14 and 15 are I/O Command Set specific.

Admin commands should not be impacted by the state of I/O queues (e.g., a full I/O completion queue should not delay or stall the Delete I/O Submission Queue command).

**Modify a portion of section 5.7 (Firmware Commit command) as shown below:**

The Firmware Commit command is used to verify that a valid firmware image has been downloaded and to commit that revision to a specific firmware slot. The host may select the firmware image to activate on the next Controller Level Reset as part of this command. The currently executing firmware revision may be determined from the Firmware Revision field of the Identify Controller data structure in Figure 90 or as indicated in the Firmware Slot Information log page. All controllers in the NVM subsystem share firmware image slots and the same firmware is applied to all controllers.

**Modify a portion of Figure 63 (Firmware Commit) as shown below:**

Value	Description
12h	<b>Firmware Activation Requires Maximum Time Violation:</b> The image specified if activated immediately would exceed the Maximum Time for Firmware Activation <del>(MFTA)</del> (MTFA) value reported in Identify Controller. To activate the firmware, the Firmware Commit command needs to be re-issued and the image activated using a reset.

**Modify a portion of Figure 86 (Identify – Data Structure Returned) as shown below:**

CNS Value	Definition
02h	A list of 1024 namespace IDs is returned containing active NSIDs in increasing order that are greater than the value specified in the Namespace Identifier (CDW1.NSID) field of the command. The controller should abort the command with status code Invalid Namespace or Format if CDW1.NSID is set to FFFFFFFEh or FFFFFFFFh. Note that CDW1.NSID may be cleared to 0h to retrieve a Namespace List including the namespace starting with NSID of 1h.  The data structure returned is a Namespace List (refer to section 4.8). Controllers that support specification revision 1.1 or later shall support this capability.

**Modify a portion of Figure 11 (Command Format – Admin Command Set) as shown below:**

Bytes	Description
07:04	<b>Namespace Identifier (NSID):</b> This field specifies the namespace ID that this command applies to. If the namespace ID is not used for the command, then this field shall be cleared to 0h. If a command shall be applied to all namespaces accessible by this controller, then this field shall be set to FFFFFFFFh.  Unless otherwise noted, specifying an inactive namespace ID in a command that uses the namespace ID shall cause the controller to abort the command with status Invalid Field in Command. Specifying an invalid namespace ID in a command that uses the namespace ID shall cause the controller to abort the command with status Invalid Namespace or Format, unless otherwise specified.

**Modify a portion of Figure 12 (Command Format – NVM Command Set) as shown below:**

Bytes	Description
07:04	<p><b>Namespace Identifier (NSID):</b> This field specifies the namespace that this command applies to. If the namespace is not used for the command, then this field shall be cleared to 0h. If a command shall be applied to all namespaces accessible by this controller, then this value shall be set to FFFFFFFFh.</p> <p>Unless otherwise noted, specifying an inactive namespace ID in a command that uses the namespace ID shall cause the controller to abort the command with status Invalid Field in Command. Specifying an invalid namespace ID in a command that uses the namespace ID shall cause the controller to abort the command with status Invalid Namespace or Format, <b>unless otherwise specified</b>.</p>

**Modify a portion of Figure 13 (Admin and NVM Vendor Specific Commands) as shown below:**

Bytes	Description
07:04	<p><b>Namespace Identifier (NSID):</b> This field indicates the namespace ID that this command applies to. If the namespace ID is not used for the command, then this field shall be cleared to 0h. If a command shall be applied to all namespaces accessible by this controller, then this field shall be set to FFFFFFFFh.</p> <p>The behavior of a controller in response to an inactive namespace ID for a vendor specific command is vendor specific. Specifying an invalid namespace ID in a command that uses the namespace ID shall cause the controller to abort the command with status Invalid Namespace or Format, <b>unless otherwise specified</b>.</p>

**Modify a portion of section 3.1.5 (Controller Configuration) as shown below:**

This register modifies settings for the controller. Host software shall set the Arbitration Mechanism (CC.AMS), the Memory Page Size (CC.MPS), and the Command Set (CC.CSS) to valid values prior to enabling the controller by setting CC.EN to '1'. **Attempting to create an I/O queue before initializing the I/O Completion Queue Entry Size (CC.IOCQES) and I/O Submission Queue Entry Size (CC.IOSQES) should cause a controller to abort Create I/O Completion Queue or Create I/O Submission Queue commands with status code Invalid Queue Size.**

**Modify a portion of Figure 90 (Identify Controller) as shown below:**

Bytes	O/M	Description
95:92	M	<p><b>Optional Asynchronous Events Supported (OAES):</b> This field indicates the optional asynchronous events supported by the controller. A controller shall not send optional asynchronous events before they are enabled by host software.</p> <p>Bits 31:9 10 are reserved.</p> <p><b>Bit 9 is set to '1' if the controller supports sending the Firmware Activation Notices. If cleared to '0' then the controller does not support the Firmware Activation Notices event.</b></p> <p><b>Bit 8 is set to '1' if the controller supports sending the Namespace Attribute <b>Changed event</b> Notices. If cleared to '0' then the controller does not support the Namespace Attribute <b>Changed event</b> Notices.</b></p> <p>Bits 7:0 are reserved.</p>

**Modify a portion of section 5.13 (Namespace Management) as shown below:**

Management command with the create operation, the namespace is formatted with the specified attributes. The fields that host software may specify in the create operation ~~is~~ **are** defined in Figure 98. Fields that are reserved shall be cleared to 0h by host software. There is no data structure transferred for the delete operation.