LEGAL NOTICE:

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

NVM Express Technical Proposal for New Feature

Technical Proposal ID	**4112 Implicit FW Pending Activation**
Change Date	**2021-07-12**
Builds on Specification	**NVM Express Base Specification 2.0**
References Specification	

Technical Proposal Author(s)

Name	Company
Mike Allison, Andres Baez, Kapil Karkra	Intel

This proposal enhances firmware update by:

- clarifying the effects on a Sanitize command and sanitize operation when firmware images become activated by a controller when not specifically activated by a host.
- clarifying that the firmware image is commented to the specified slot when the returned status code of a Firmware Activation command is Firmware Activation Requires Maximum Time Violation.

Revision History

Revision Date	Change Description
2021-04-09	Initial version
2021-04-14	Added the word “subsequent” and a comment to discuss multiple firmware slots use case.
2021-04-19	Minor edits. Added that a controller may abort a Firmware Commit command that overwrites the active FW slot without activation if the domain supports multiple firmware slots that host can store firmware images.
2021-04-29	Edits during review in WG. Ready for Phase 2 & 3
2021-05-20	Renamed file ready for member review.
2021-06-24	Updated content with latest Base Specification document available
2021-07-12	Integrated into the NVMe Base Specification, revision 2.0.

Description for NVMe Base Specification 2.0 Changes Document

Feature Enhancement:

- **Incompatible and New requirement**:
  - The status code Firmware Activation Requires Maximum Time Violation has been updated to state that the firmware image is committed to the firmware slot. Prior to this change nothing was stated on whether the firmware image was or was not committed.
- **New requirement:**
  - A Sanitize command is aborted if the host commits a firmware image to the currently active firmware slot without activation and that committed firmware image is allowed to be activated by a reset.
- **New requirement:**

- If a firmware image cannot be loaded as part of a reset and the controller reverts to a different firmware image while a sanitize operation is in progress, then that sanitize operation fails.
- If the host issues attempts to commit without activation to the active firmware slot.A controller may abort a Firmware Commit command if the host attempts to commit without activation to the active firmware slot.

Description of Specification Changes

Markup Conventions:

Black:	Unchanged (however, hot links are removed)
~~Red Strikethrough~~:	Deleted
Blue:	New
Blue Highlighted:	TBD values, anchors, and links to be inserted in new text.
<Green Bracketed>:	Notes to editor

Modify portions of NVM Express Base Specification 2.0 (dated 5/17/2021 Preratification) as shown below:

Modify a portion of section 3.11 as shown below:

3.11 Firmware Update Process

...

If the firmware image is not able to be successfully loaded, then the controller shall revert to the firmware image present in the most recently activated firmware slot or the baseline read-only firmware image, if available, and indicate the failure as an asynchronous event with a Firmware Image Load Error. If the controller changes (e.g., reverts) the firmware image while a sanitize operation is in progress, then that sanitize operation fails (refer to section 8.21.1).

If a host overwrites (i.e., updates) the firmware image in the active firmware slot, then the previously active firmware image may no longer be available. As a result, any action (e.g., power cycling the controller) that requires the use of that firmware slot may instead use the firmware image that is currently in that firmware slot. If the firmware image that is currently in that firmware slot would be activated by such an action replacing the currently active firmware, then:

- this is a pending firmware activation with reset; and
- a controller aborts subsequent Sanitize commands with a status code indicating the appropriate reset required to activate the pending activation as defined in section 5.24.

...

Modify a portion of Figure 183 in section 5.12.1 as shown below:

5.12 Firmware Commit command

Note: This command was known in NVM Express Base Specification revisions prior to revision 1.2 as “Firmware Activate.”

The Firmware Commit command is used to modify the firmware image or Boot Partitions.

When modifying a firmware image, the Firmware Commit command verifies that a valid firmware image has been downloaded and commits 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 host may determine the currently executing firmware revision by examining the Firmware Revision field in the Identify Controller data structure in Figure 275. The host may determine the firmware revision to be executed on the next Controller Level Reset by examining the Firmware Slot Information log page. All controllers in a domain share firmware slots and the same firmware image is applied to all controllers in that domain (i.e., all the controllers in the NVM subsystem if multiple domains are not supported or all the controllers in that domain if multiple domains are supported).

Activation of a firmware image may result in a change in controller behavior that is not expected by the host (e.g., an incompatible change in the UUID List (refer to section 8.25.2)). In this case, if the Commit Action field is set to 011b, then the controller shall abort the command with a status code of Firmware Activation Requires Conventional Reset.

When modifying Boot Partitions, the host may select the Boot Partition to mark as active or replace. A Boot Partition is only able to be written when unlocked (refer to section 8.2).

The Firmware Commit command uses the Command Dword 10 field. All other command specific fields are reserved.

Figure 181: Firmware Commit – Command Dword 10

<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>31</td><td>Boot Partition ID (BPID): Specifies the Boot Partition that shall be used for the Commit Action, if applicable.</td></tr>
<tr><td>30:06</td><td>Reserved</td></tr>
<tr><td>05:03</td><td>Commit Action (CA): This field specifies the action that is taken (refer to section 3.11) on the image downloaded with the Firmware Image Download command or on a previously downloaded and placed image. The actions are indicated in the following table.
<table>
<tr><th>Value</th><th>Definition</th></tr>
<tr><td>000b</td><td>Downloaded image replaces the existing image, if any, in the specified Firmware Slot. The newly placed image is not activated.</td></tr>
<tr><td>001b</td><td>Downloaded image replaces the existing image, if any, in the specified Firmware Slot. The newly placed image is activated at the next Controller Level Reset.</td></tr>
<tr><td>010b</td><td>The existing image in the specified Firmware Slot is activated at the next Controller Level Reset.</td></tr>
<tr><td>011b</td><td>Downloaded image replaces the existing image, if any, in the specified Firmware Slot and is then activated immediately. If there is not a newly downloaded image, then the existing image in the specified firmware slot is activated immediately.</td></tr>
<tr><td>100b to 101b</td><td>Reserved</td></tr>
<tr><td>110b</td><td>Downloaded image replaces the Boot Partition specified by the Boot Partition ID field.</td></tr>
<tr><td>111b</td><td>Mark the Boot Partition specified in the BPID field as active and update BPINFO.ABPID.</td></tr>
</table></td></tr>
<tr><td>02:00</td><td>Firmware Slot (FS): Specifies the firmware slot that shall be used for the Commit Action, if applicable. If the value specified is 0h, then the controller shall choose the firmware slot (i.e., slot 1 to slot 7) to use for the operation.</td></tr>
</table>

5.12.1 Command Completion

Upon completion of the Firmware Commit command, the controller posts a completion queue entry to the Admin Completion Queue indicating the status for the command.

For Firmware Commit commands that specify activation of a new firmware image at the next Controller Level Reset (i.e., the CA field was set to 001b or 010b) and complete with a status code value of 0h (i.e., Success Completion), a Controller Level Reset initiated by any of the methods defined in section 3.7.2 activates the specified firmware.

If the controller detects overlapping firmware/boot partition image update command sequences (refer to section 1.5.23) of more than one firmware image and/or Boot Partition or the use of more than one controller

and/or Management Endpoint to update a single firmware image, then the results of that detection are reported in Dword 0 of the completion queue entry as defined in Figure 182. Refer to section 3.11 and section 8.2.2.

If:

a) the Commit Action field is cleared to 000b (i.e., update the firmware image in the specified firmware slot but do not activate);
b) the Firmware Slot field is set to the active firmware slot (refer to Figure 209); and
c) the controller supports greater than one firmware slot for a host to store firmware images,

then the controller may abort the command with a status code of Command Sequence Error. A host may avoid this result by using a different firmware slot.

Figure 182: Firmware Commit – Completion Queue Entry Dword 0

<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>31:02</td><td>Reserved</td></tr>
<tr><td>01:00</td><td>Multiple Update Detected (MUD): This field indicates if a controller detected overlapping firmware/boot partition image update command sequences of Boot Partitions and/or firmware images (refer to section 3.11 and section 8.2.2). If the SMUD bit in the Firmware Update field of the Identify Controller data structure is cleared to ‘0’, then this field shall be cleared to 00b.

This field is valid if the command is successful or aborted.
<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>1</td><td>If set to ‘1’, then the controller detected an overlapping firmware/boot partition image update command sequence due to processing a command from a Management Endpoint. If cleared to ‘0’, then the controller did not detect an overlapping firmware/boot partition image update command sequence due to processing a command from a Management Endpoint.</td></tr>
<tr><td>0</td><td>If set to ‘1’, then the controller detected an overlapping firmware/boot partition image update command sequence due to processing a command from an Admin Submission Queue on a controller. If cleared to ‘0’, then the controller did not detect an overlapping firmware/boot partition image update command sequence due to processing a command from an Admin Submission Queue on a controller.</td></tr>
</table></td></tr>
</table>

Firmware Commit command specific status values are defined in Figure 183.

Figure 183: Firmware Commit – Command Specific Status Values

Value	Description
06h	**Invalid Firmware Slot:** The firmware slot indicated is invalid or read only. This error is indicated if the firmware slot exceeds the number supported.
07h	**Invalid Firmware Image:** The firmware image specified for activation is invalid and not loaded by the controller.
0Bh	**Firmware Activation Requires Conventional Reset:** The firmware commit was successful, however, activation of the firmware image requires a Conventional Reset. If an FLR or Controller Reset occurs prior to a Conventional Reset, the controller shall continue operation with the currently executing firmware image.
10h	**Firmware Activation Requires NVM Subsystem Reset:** The firmware commit was successful, however, activation of the firmware image requires an NVM Subsystem Reset. If any other type of Controller Level Reset occurs prior to an NVM Subsystem Reset, the controller shall continue operation with the currently executing firmware image.
11h	**Firmware Activation Requires Controller Level Reset:** The firmware commit was successful; however, the image specified does not support being activated without a Controller Level Reset. The image shall be activated at the next Controller Level Reset. This status code should be returned only if the Commit Action field in the Firmware Commit command is set to 011b (i.e., activate immediately).
12h	**Firmware Activation Requires Maximum Time Violation:** The firmware commit was successful; however, t~~T~~he image specified if activated immediately would exceed the Maximum Time for Firmware Activation (MTFA) value reported in the Identify Controller data structure (refer to Figure 275). To activate the firmware, the Firmware Commit command needs to be re-issued and the image activated using a reset.

Figure 183: Firmware Commit – Command Specific Status Values

Value	Description
13h	**Firmware Activation Prohibited:** The image specified is being prohibited from activation by the controller for vendor specific reasons (e.g., controller does not support down revision firmware).
14h	**Overlapping Range**: This error is indicated if the firmware image has overlapping ranges.
1Eh	**Boot Partition Write Prohibited:** This error is indicated if a command attempts to modify a Boot Partition while locked (refer to section 8.2.3).

Modify a portion of section 5.24 as shown below:

5.24 Sanitize command

...

If a firmware activation with reset is pending, then the controller shall abort any Sanitize command (refer to section 5.12.1 and section 3.11).

If the Firmware Commit command that established the pending firmware activation with reset condition returned a status code of:

a) Firmware Activation Requires Controller Level Reset;
b) Firmware Activation Requires Conventional Reset; or
c) Firmware Activation Requires NVM Subsystem Reset.

then the controller should abort the Sanitize command with that same status code.

If the Firmware Commit command that established the pending firmware activation with reset condition completed successfully or returned a status code other than:

a) Firmware Activation Requires Controller Level Reset;
b) Firmware Activation Requires Conventional Reset; or
c) Firmware Activation Requires NVM Subsystem Reset,

then the controller should abort the Sanitize command with a status code of Firmware Activation Requires Controller Level Reset.

Activation of new firmware is prohibited during a sanitize operation (refer to section 8.21.1).

…

Modify a portion of section 8.21.1 as shown below:

8.21.1 Sanitize Operation Restrictions

While performing a sanitize operation and while a failed sanitize operation has occurred but successful recovery from that failure has not occurred, all enabled controllers and namespaces in the NVM subsystem are restricted to performing only a limited set of actions.

While a sanitize operation is in progress:

- All controllers in the NVM subsystem shall only process the Admin commands listed in Figure 139 subject to the additional restrictions stated in that figure;
- All I/O Commands other than a Flush command shall be aborted with a status code of Sanitize In Progress;
- Processing of a Flush command is specified in section 7.1;
- Any command or command option that is not explicitly permitted in Figure 139 shall be aborted with a status code of Sanitize In Progress if fetched by any controller in the NVM subsystem; and
- The Persistent Memory Region shall be prevented from being enabled (i.e., setting PMRCTL.EN to ‘1’ does not result in PMRSTS.NRDY being cleared to ‘0’).

- If a controller changes (e.g., reverts) the firmware image (refer to section 3.11), then the sanitize operation shall fail.

While a failed sanitize operation has occurred, a subsequent sanitize operation has not started and successful recovery from the failed sanitize operation has not occurred:

- All controllers in the NVM subsystem shall only process the Sanitize command (refer to section 5.24) and the Admin commands listed in Figure 139 subject to the additional restrictions noted in that figure;
- All I/O Commands other than a Flush command (refer to section 7.1) shall be aborted with a status code of Sanitize Failed;
- The Sanitize command is permitted with action restrictions (refer to section 5.24);
- Aside from the Sanitize command, any other command or command option that is not explicitly permitted in Figure 139 shall be aborted with a status code of Sanitize Failed if fetched by any controller in the NVM subsystem; and
- The Persistent Memory Region shall be prevented from being enabled (i.e., setting PMRCTL.EN to '1' does not result in PMRSTS.NRDY being cleared to '0').