



LEGAL NOTICE:

© Copyright 2007 - 2018 NVM Express, Inc. ALL RIGHTS RESERVED.

This NVM Express revision 1.3 technical proposal is proprietary to the NVM Express, Inc. (also referred to as "Company") and/or its successors and assigns.

NOTICE TO USERS WHO ARE NVM EXPRESS, INC. MEMBERS: Members of NVM Express, Inc. have the right to use and implement this NVM Express revision 1.3 technical proposal subject, however, to the Member's continued compliance with the Company's Intellectual Property Policy and Bylaws and the Member's Participation Agreement.

NOTICE TO NON-MEMBERS OF NVM EXPRESS, INC.: If you are not a Member of NVM Express, Inc. and you have obtained a copy of this document, you only have a right to review this document or make reference to or cite this document. Any such references or citations to this document must acknowledge NVM Express, Inc. copyright ownership of this document. The proper copyright citation or reference is as follows: "© 2007 - 2018 NVM Express, Inc. ALL RIGHTS RESERVED." When making any such citations or references to this document you are not permitted to revise, alter, modify, make any derivatives of, or otherwise amend the referenced portion of this document in any way without the prior express written permission of NVM Express, Inc. Nothing contained in this document shall be deemed as granting you any kind of license to implement or use this document or the specification described therein, or any of its contents, either expressly or impliedly, or to any intellectual property owned or controlled by NVM Express, Inc., including, without limitation, any trademarks of NVM Express, Inc.

LEGAL DISCLAIMER:

THIS DOCUMENT AND THE INFORMATION CONTAINED HEREIN IS PROVIDED ON AN "AS IS" BASIS. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, NVM EXPRESS, INC. (ALONG WITH THE CONTRIBUTORS TO THIS DOCUMENT) HEREBY DISCLAIM ALL REPRESENTATIONS, WARRANTIES AND/OR COVENANTS, EITHER EXPRESS OR IMPLIED, STATUTORY OR AT COMMON LAW, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, VALIDITY, AND/OR NONINFRINGEMENT.

All product names, trademarks, registered trademarks, and/or servicemarks may be claimed as the property of their respective owners.

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

NVM Express Technical Proposal for New Feature

Technical Proposal ID	4024 – Traffic Based Keep Alive
Change Date	9/17/2018
Builds on Specification	NVM Express 1.3

Technical Proposal Author(s)

Name	Company
Sagi Grimberg	Lightbits Labs
Michael Allison	Intel
Fredrick Knight	NetApp

This technical proposal defines Traffic Based Keep Alive mode of operation.

A host is expected to issue Keep Alive commands expecting a completion within the KATO period, if the Keep Alive command does not complete in the KATO period, a host should teardown the controller association. Similarly, a controller expects Keep Alive commands to be issued by the host every KATO period, if the controller does not receive a Keep Alive command within the KATO period, the controller shall teardown the association as well.

Under heavy load, Keep Alive commands might not complete during the KATO period. It can be due to heavy congestion which causes packet drops and fabric reconnect attempts, or simply because the controller device is overloaded and not able to service the Keep Alive command in time.

Given that normal I/O submissions and completions are sufficient to detect host and/or controller health, processing of periodic Keep Alive commands is redundant.

Revision History

Revision Date	Change Description
10/23/2017	Initial proposal
10/26/2017	Added review comments from Michael Allison <ul style="list-style-type: none"> - Naming: Keep Alive - Changed Initiated to executed - Stating that for TBKAS, Admin and/or I/O commands execution during the Keep Alive timeout interval allows a host to expand the Keep Alive timeout by one more Keep Alive timeout interval. This is instead of stating that Admin and/or I/O commands reset the Keep Alive timer.
1/11/2017	<ul style="list-style-type: none"> - Moved TBKAS from controller capability register to Controller Attributes (in controller identify data structure). - Changed “may”/“allowed” to “shall” - Clarified that in TBKAS, Admin and/or I/O commands expand the Keep Alive timeout by one more Keep Alive Timeout Interval and the maximum Keep Alive timeout is twice the Keep Alive Timeout Interval.
11/06/2017	Made TBKAS just indicate support without the detail and refer to the section with the detail
11/26/2017	Separating Traffic Based Keep Alive to a dedicated section
11/28/2017	Addressing comments from Michael Allison
11/29/2017	Remove in-paragraph carriage returns, delete addressed comments and remove some extra spaces
11/30/2017	Addressed comments from Fred Knight <ul style="list-style-type: none"> - Rework duplicated description of the Keep Alive Feature - Added reference to KATO where appropriate - Defined the undefined Keep Alive Timeout Interval - Added a condition on non-zero KATO set for Keep Alive Activation - Few more minor edits
12/6/2017	<ul style="list-style-type: none"> - Rework the phrasing of “extendig the Keep Alive Timeout Interval” to “restart the Keep Alive Timer”
16/1/2016	<ul style="list-style-type: none"> - Separated Keep Alive into “Keep Alive Command Based Keep Alive” section and “Traffic Based Keep Alive” section
1/21/18	<ul style="list-style-type: none"> - Incorporated Comments from Michael Allison, Hugh Curley and Fred Knight
2/14/18	<ul style="list-style-type: none"> - Clarified set features keep alive from host/controller perspective - Minor editing (capitalize)
2/18/18	<ul style="list-style-type: none"> - Eeditorial comments from Mike - Removed “milliseconds” after KATO as KATO is already in milliseconds - Colored removals in red - Added text in 7.12.2 that the host needs to send a Keep Alive if not Admin or I/O commands were executed during the Keep Alive timeout.
3/22/18	<ul style="list-style-type: none"> - Say that host sends keep alive at half of the keep alive timeout interval - Modify Figure 31 “Keep Alive Timer Expired”
3/25/18	<ul style="list-style-type: none"> - Added Fred to the author list - Aligned to NVMe1.3b (moved TBKAS to bit 6 of controller attributes)
3/28/18	<ul style="list-style-type: none"> - Modified section 7.12.2 so the host should (and not shall) send keep alive commands in half of the keep alive timeout interval
4/6/2018	<ul style="list-style-type: none"> - Fixed capitalization - Reword forms of ‘execution’ to ‘processing’ to avoid ambiguity - Other minor editorial fixes

6/5/2018	<ul style="list-style-type: none"> - Final rewording for final comments from Judy and Fred - Added Summary of Technical Proposal paragraph
8/22/2018	<ul style="list-style-type: none"> - Updated the date to match WG request for ratification. - Put history in chronological order. - Removed comments and tracking.
8/23/2018	<ul style="list-style-type: none"> - Technical writer editorial update
9/17/2018	<ul style="list-style-type: none"> - Ratified

Summary of Technical Proposal

Introduction of a Keep Alive mode where a host and a controller reset the Keep Alive Timer in the presence of Admin commands or I/O commands. Admin or I/O command processing indicate proper health of a host, controller and transport connectivity. Hence, Traffic Based Keep Alive allows for a host and controller to restart the Keep Alive Timer in the presence of an Admin command or an I/O command and not only upon processing of a Keep Alive command.

Description of Specification Changes

Modify a portion of Figure 109 (Identify – Identify Controller Data Structure) as shown below:

Modify the Controller Attributes entry as follows:

99:96	M	<p>Controller Attributes (CTRATT): This field indicates attributes of the controller.</p> <p>Bits 31:67 are reserved.</p> <p>Bit 6 (Traffic Based Keep Alive Support - TBKAS): If set to '1', then the controller supports restarting the Keep Alive Timer if an Admin command or an I/O command is processed during the Keep Alive Timeout Interval (refer to section 7.12.2). If cleared to '0', then the controller supports restarting the Keep Alive Timer only if a Keep Alive command is processed during the Keep Alive Timeout Interval (refer to section 7.12.1).</p> <p>Bit 5 (Predictable Latency Mode): If set to '1' then the controller supports Predictable Latency Mode (refer to section 8.18). If cleared to '0' then the controller does not support Predictable Latency Mode.</p> <p>Bit 4 (Endurance Groups): If set to '1', then the controller supports Endurance Groups (refer to section 8.17). If cleared to '0', then the controller does not support Endurance Groups.</p> <p>Bit 3 (Read Recovery Levels): If set to '1', then the controller supports Read Recovery Levels (refer to section 8.16). If cleared to '0', then the controller does not support Read Recovery Levels.</p> <p>Bit 2 (NVM Sets): If set to '1', then the controller supports NVM Sets (refer to section 4.9). If cleared to '0', then the controller does not support NVM Sets.</p> <p>Bit 1 (Non-Operational Power State Permissive Mode): If set to '1', then the controller supports host control of whether the controller may temporarily exceed the power of a non-operational power state for the purpose of executing controller initiated background operations in a non-operational power state (i.e., Non-Operational Power State Permissive Mode supported). If cleared to '0', then the controller does not support host control of whether the controller may exceed the power of a non-operational state for the purpose of executing controller initiated background operations in a non-operational state (i.e., Non-Operational Power State Permissive Mode not supported). Refer to section 5.21.1.17.</p> <p>Bit 0 if set to '1' then the controller supports a 128-bit Host Identifier. Bit 0 if cleared to '0' then the controller does not support a 128-bit Host Identifier.</p>
-------	---	--

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

Modify a Status code entry as follows:

19h	Keep Alive Timerout Expired: The Keep Alive Timer out expired.
-----	---

Modify a portion of section 5.16 (Keep Alive command) as shown below:

5.16 Keep Alive command

The Keep Alive command (refer to section 5.21.1.15) and associated functionality is used by the host to determine that the controller is operational and by the controller to determine that the host is operational. The host and controller are operational when each is accessible and able to issue or ~~execute~~ process commands. The controller indicates the granularity of the Keep Alive Timer in the KAS field in the Identify Controller data structure (refer to Figure 114).

If a Keep Alive Timeout has been enabled on the Admin Queue, the Keep Alive Timer is ~~reset when this command is executed~~ restarted when:

- A Keep Alive command (refer to section 7.12.1) is processed; or
- At the end of the Keep Alive Timeout (refer to section 7.12.2) when TBKAS is set to '1' and an Admin command or an I/O command is processed during the Keep Alive Timeout Interval.

All command specific fields are reserved.

Modify a portion of section 7.12 (Keep Alive) as shown below:

7.12 Keep Alive

~~The Keep Alive feature (refer to section 5.21.1.15) is used by the host to determine that the controller is operational and by the controller to determine that the host is operational. The host and controller are operational when each is accessible and able to issue or execute commands. The controller indicates the granularity of the Keep Alive Timer in the Identify Controller data structure.~~

The Keep Alive is a watchdog timer intended to detect a malfunctioning connection, controller, or host. The Keep Alive Timeout Interval is the period during which the Keep Alive Timer is activated.

A Keep Alive Timeout Interval on the controller starts when:

- a successful Completion Queue entry is posted for a Set Features command with Feature Identifier 0Fh and a non-zero KATO field.

A Keep Alive Timeout Interval on the host starts when:

- a Set Features command with Feature Identifier 0Fh and a non-zero KATO field is posted to the Admin submission queue; or
- a Keep Alive command was posted to the Admin submission queue.

Both on the host and the controller the Keep Alive Timeout Interval ends the time specified by the KATO field after the interval started (refer to Figure 161). A Keep Alive Timeout occurs when the Keep Alive Timer expires. The Keep Alive Timer expires:

- if TBKAS is cleared to '0', at the end of the Keep Alive Timeout Interval and no Keep Alive Command was processed during the Keep Alive Timeout Interval (refer to section 7.12.1); and
- if TBKAS is set to '1', at the end of the Keep Alive Timeout Interval and no Admin command or I/O command was processed during the Keep Alive Timeout Interval (refer to section 7.12.2).

The Keep Alive timer is active ~~only for an enabled controller, i.e., the Keep Alive timer is active~~ if:

- CC.EN is set to '1' and CSTS.RDY is set to '1'; ~~and~~
- CC.SHN is cleared to '0'; ~~and CSTS.SHST is cleared to '0'~~
- CSTS.SHST is cleared to '0'; and
- the Keep Alive Timer feature has been enabled with a KATO field set to a non-zero value,

~~Otherwise~~, the Keep Alive timer is inactive and a Keep Alive Timeout shall not occur. Activating an inactive Keep Alive timer (e.g., enabling a controller with the Keep Alive feature in use) shall initialize the Keep Alive timer to the Keep Alive Timeout (KATO) value.

~~When~~ If a Keep Alive Timer ~~for the Admin Queue~~ expires:

- the controller records an Error Information Log Entry with the status code Keep Alive Timer Expired and sets the Controller Fatal Status (CSTS.CFS) bit to '1'; and
- the host assumes all outstanding commands are not completed and need to be re-issued.

Add a new section 7.12.1 (Traffic Based Keep Alive) as shown below:

7.12.1 Keep Alive Command Based Keep Alive

Keep Alive Command Based Keep Alive restricts the Keep Alive Timer on both the host and the controller to be restarted only upon the processing of a Keep Alive command. This mode is in use if TBKAS is cleared to '0'.

The Keep Alive Timeout is the maximum time a connection remains established without processing a Keep Alive command. If the Keep Alive Timer in the controller expires and a Keep Alive command has not been processed within the Keep Alive Timeout Interval, then the controller may consider a Keep Alive Timeout to have occurred. If the Keep Alive Timer in the host expires and a completion of a Keep Alive command has not been received within the Keep Alive Timeout Interval, then the host may consider a Keep Alive Timeout to have occurred. The host should send Keep Alive commands at half of the Keep Alive Timeout accounting for transport roundtrip times, transport delays, command processing times, and the Keep Alive Timer granularity.

Add a new section 7.12.2 (Traffic Based Keep Alive) as shown below:

7.12.2 Traffic Based Keep Alive

Traffic Based Keep Alive (TBKAS) allows the host and controller to restart the Traffic Based Keep Alive Timer in the presence of Admin or I/O command processing. The controller support for TBKAS is indicated in the Controller Attributes in the Identify Namespace data structure (refer to Figure 114). If the Controller does not support Traffic Based Keep Alive (TBKAS is cleared to '0'), then the operation of the Keep Alive feature is described in section 7.12.1.

The Traffic Based Keep Alive Timeout occurs if a connection remains established without processing an Admin command or an I/O command during the Keep Alive Timeout Interval. If an Admin command or an I/O command is processed within the Keep Alive Timeout Interval, then upon the expiration of the Keep Alive Timer, the Keep Alive Timer shall be restarted.

The controller may consider a Keep Alive Timeout to have occurred if no Admin command or no I/O command is transferred to the controller (as defined in section 4.11) within the Keep Alive Timeout Interval. If an Admin command or an I/O command is transferred to the Controller within the Keep Alive Timeout Interval, then upon the expiration of the Keep Alive Timer the controller shall restart the Keep Alive Timer.

The host may consider a Traffic Based Keep Alive Timeout to have occurred if the host does not receive a completion of any Admin command or any I/O command within the Keep Alive Timeout Interval. If an Admin command or an I/O command is completed within the Keep Alive Timeout Interval, then upon expiration of the Keep Alive Timer, the host shall restart the Keep Alive Timer. The host should check for a command completion entry for any Admin commands and I/O commands at half of the Keep Alive Timeout accounting for transport roundtrip times, transport delays, command processing times, and the Keep Alive Timer granularity. To prevent the controller from detecting a Keep Alive Timeout, if no Admin command and no I/O command is sent to the controller during half of the Keep Alive Timeout Interval, the host should send a Keep Alive Command.