

**Proposed
Draft**

**Serial ATA
International Organization**

**Version 2
26 January 2015**

**Serial ATA Revision 3.2 Technical Proposal 068
Title : Clarification of the FPDMA state machine
and error reporting**

This is an internal working document of the Serial ATA International Organization. As such, this is not a completed standard and has not been approved. The Serial ATA International Organization may modify the contents at any time. This document is made available for review and comment only.

Permission is granted to the Promoters, Contributors and Adopters of the Serial ATA International Organization to reproduce this document for the purposes of evolving the technical content for internal use only without further permission provided this notice is included. All other rights are reserved and may be covered by one or more Non Disclosure Agreements including the Serial ATA International Organization participant agreements. Any commercial or for-profit replication or republication is prohibited. Copyright © 2000 to 2015 Serial ATA International Organization. All rights reserved.

This Draft Specification is NOT the final version of the Specification and is subject to change without notice. A modified, final version of this Specification ("Final Specification") when approved by the Promoters will be made available for download at this Web Site: <http://www.sata-io.org>.

THIS DRAFT SPECIFICATION IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION, OR SAMPLE. Except for the right to download for internal review, no license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted or intended hereunder.

THE PROMOTERS DISCLAIM ALL LIABILITY, INCLUDING LIABILITY FOR INFRINGEMENT OF ANY PROPRIETARY RIGHTS, RELATING TO USE OF INFORMATION IN THIS DRAFT SPECIFICATION. THE PROMOTERS DO NOT WARRANT OR REPRESENT THAT SUCH USE WILL NOT INFRINGE SUCH RIGHTS.

THIS DOCUMENT IS AN INTERMEDIATE DRAFT FOR COMMENT ONLY AND IS SUBJECT TO CHANGE WITHOUT NOTICE.

* Other brands and names are the property of their respective owners.

Copyright © 2000 to 2015 Serial ATA International Organization. All rights reserved.

Author Information

Author Name	Company	Email address
Paul Suhler	HGST	paul.suhler@hgst.com

Workgroup Chair Information

Workgroup (Phy, Digital, etc...)	Chairperson Name	Email address
Digital	James Hatfield	James.C.Hatfield@seagate.com

Document History

Version	Date	Comments
0	9 January 2015	Initial release.
1	26 January 2015	Version approved by committee.
2	26 January 2015	Member review, changed D194 to TPR068.

Introduction

SATA devices report errors in either the Register Device to Host FIS (RDH FIS) or in the Set Device Bits – Device to Host FIS (SDB FIS). If an FPDMA QUEUED command is judged to be “malformed,” then those command parameter errors are reported in the RDH FIS. Malformed commands are handled in the FPDMA Queued state machine transition DFPDMAQ1:2, as described in SATA-IO 3.2 subclause 11.15:

A command may be considered malformed as a result of any of its parameters being invalid, including the use of a TAG value that corresponds to an existing TAG value for a pending command.

This will cause the state machine to enter the DFPDMAQ12: BrokenHost_ClearBusy state, where the device transmits a RDH FIS with ERR set to one in the Status field, Interrupt bit set to one, BSY bit cleared to zero, DRQ bit cleared to zero, and ATA error code set to one in the Error field.

Devices employing hardware acceleration of FPDMA QUEUED commands must be able to evaluate the validity of command parameters quickly in order to return an RDH FIS promptly. However in a zoned device, some commands require a complicated analysis of zone IDs, LBAs, etc.; in many cases this analysis would be very expensive to perform in hardware.

We believe that the “may” in subclause 11.15’s statement “... command may be considered malformed ...” does not mean that DFPDMAQ1: AddCommandToQueue is the *only* state in which a malformed command may be detected, and it *is* permissible to defer checking of some parameters until after the RDH FIS has been sent.

Thus if a command parameter error is detected later – presumably while the device is in the DIO: Device_idle state – then transition DIO:6 takes the device to the DFPDMAQ11: ERROR state, where the SDB FIS is sent.

This does not require a technical change to the standard.

The clarification proposed is to add:

- a) a parenthetical phrase to the state table entry for transition DFPDMAQ1:2 indicating that a malformed command may not be detected; and
- b) a note to the paragraph explaining transition DFPDMAQ1:2 explaining further.

The author is uncertain whether a note in the description of one transition is allowed to refer to another transition, so alternative versions of the note are offered below.

Thanks to Ralph Weber, Joe Breher, et al. for their assistance in preparing this proposal.

[editor note: Existing text is black. New text is marked as underlined in blue color. Material to be deleted ~~is red with strikethrough markings.~~

Changes to Serial ATA 3.2

11 Device command layer protocol

...

11.15 FPDMA QUEUED command protocol

...

DFPDMAQ1: AddCommandToQueue	Append command to internal device command queue and store TAG value.		
1. Device successfully en-queued the command	→	DFPDMAQ2: ClearInterfaceBsy	
2. Command malformed Malformed command detected	→	DFPDMAQ12: BrokenHost_ClearBusy	

...

Figure 332 – Device command layer FPDMA queued state machine

...

Transition DFPDMAQ1:2, if the device determines that the received command is malformed, an error has occurred and the device shall transition to the DFPDMAQ12: BrokenHost_ClearBusy state. A command may be considered malformed as a result of any of its parameters being invalid, including the use of a TAG value that corresponds to an existing TAG value for a pending command.

[Note xx – Complete command parameter validation may not be possible when the device is in DFPDMAQ1: AddCommandToQueue state. In these cases, complete parameter validation occurs while the device is in DI0: Idle state \(see 11.3\).](#)