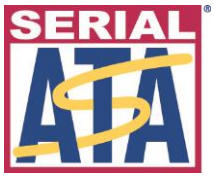


Serial ATA International Organization

Serial ATA Revision 3.5a

3/2/2021
Gold

SATA-IO Board Members

Dell Inc.

Intel Corporation

Marvell Semiconductor

Microchip Technology

Seagate Technology

Western Digital Corporation

The Serial ATA International Organization, Serial ATA Revision 3.5 specification is available for download at http://www.sata-io.org.

SPECIFICATION DISCLAIMER

THIS SPECIFICATION IS PROVIDED TO YOU “AS IS” WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE. THE AUTHORS OF THIS SPECIFICATION DISCLAIM ALL LIABILITY, INCLUDING LIABILITY FOR INFRINGEMENT OF ANY PROPRIETARY RIGHTS, RELATING TO USE OR IMPLEMNETATION OF INFORMATION IN THIS SPECIFICATION. THE AUTHORS DO NOT WARRANT OR REPRESENT THAT SUCH USE WILL NOT INFRINGE SUCH RIGHTS. THE PROVISION OF THIS SPECIFICATION TO YOU DOES NOT PROVIDE YOU WITH ANY LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS.

For more information about Serial ATA, refer to the Serial ATA International Organization website at http://www.sata-io.org.

Serial ATA International Organization contact information

SATA-IO
3855 SW 153rd Drive
Beaverton, Oregon 97003 USA
Tel +1 503-619-0572
Fax +1 503-644-6708
E-mail admin@sata-io.org

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 Revision history

1.1 Revision 2.5 (ratification date October 27, 2005)

Release that integrates and consolidates the following previously published specifications including all erratum against those specifications:

a) Serial ATA revision 1.0a;
b) Serial ATA II: Extensions to Serial ATA 1.0a revision 1.2;
c) Serial ATA II: Electrical Specification revision 1.0;
d) Serial ATA II: Cable and Connectors Volume 1 revision 1.0;
e) Serial ATA II: Cable and Connectors Volume 2 revision 1.0;
f) Serial ATA II: Port Multiplier revision 1.2; and
g) Serial ATA II: Port Selector revision 1.0.

1.2 Revision 2.6 (ratification date February 15, 2007)

Release that incorporates errata against Revision 2.5 and the following new features and enhancements:

a) Internal Slimline cable and connector;
b) Internal Micro SATA connector for 1.8” HDD;
c) Mini SATA Internal Multilane cable and connector;
d) Mini SATA External Multilane cable and connector;
e) NCQ Priority;
f) NCQ Unload;
g) Enhancements to the BIST Activate FIS; and
h) Enhancements for robust reception of the Signature FIS.

1.3 Revision 3.0 (ratification date June 6, 2009)

Release that incorporates errata against Revision 2.6:

a) ECN001 - Slimline Bump Correction;
b) ECN002 - Bump Correction;
c) ECN003 - State Name Corrections;
d) ECN004 - ATA Log & Subcode reservations;
e) ECN006 - fbaud/10 Jitter Parameter Removal;
f) ECN008 - fbaud/500 Jitter Parameter Clarification;
g) ECN009 - Correcting LBP references in the COMP data pattern and other locations;
h) ECN010 - Power State Resume Speed;
i) ECN011 - Data validity clarifications;
j) ECN012 - Cable & Connector Retention;
k) ECN013 - Section 13 Corrections;
l) ECN014 - PACKET State Names;
m) ECN016 - Long Term Frequency Accuracy and SSC Profile Tests for Transmitters;
n) ECN017 - OOB Burst/Gap Duration Clarification;
o) ECN018 - Figure 69 Pin Location Correction;
p) ECN019 - Cable ISI Test Source Risetime;
q) ECN021 - External plug height;
r) ECN022 - Editorial cleanup – 8KB, IDENTIFY DEVICE;
s) ECN023 - L-Key Opening Correction (Slimline Host Receptacle Connector) ;
t) ECN024 - Contact Current Rating procedure;
u) ECN025 - Rise Time Measurements;
v) ECN026 - Gen 3i TX TJ Measurement Location;
w) ECN027 - Gen3i Rx Differential Return Loss Text Description and Figure Change - Clarification Only;

x) ECN028 - Clarification of Test Patterns for Measurement Defined in 7.2 Electrical Specification;
y) ECN029 - Addition of Pattern to the TX AC Common Mode Voltage (Gen3i) Measurement;
z) ECN031 - Correction to ECN 004;
aa) ECN032 - Correction to TX AC Common Mode Voltage Table Value 'Units';
bb) ECN033 - Definition of Terms;
cc) ECN034 - Corrections to Technical Proposal 005;
dd) ECN035 - Clarification of Words 76 to 79;
ee) ECN036 - LIF-SATA Clarifications;
ff) ECN037 - Changes made by Technical Integration Work Group; and
gg) ECN038 - Key clarification,

and the following new features and enhancements:

a) TP002 - Gen3 register assignments;
b) TP004 - NCQ Clarifications;
c) TP005 - SATA Speed Indicator in ID String;
d) TP007 - Automatic Partial to Slumber Transitions;
e) TP009 - LIF Connector for 1.8” HDD for SATA Revision 2.6;
f) TP010 - Serial ATA NCQ Streaming Command;
g) TP011 - Serial ATA NCQ QUEUE MANAGEMENT Command;
h) TP012 - Gen 1 Clock to Data Jitter Definition;
i) TP013 - Connector for 7 mm slimline drives;
j) TP014 - Allow READ LOG DMA EXT to Clear NCQ Error;
k) TP015 - Remove Device Register from Signature;
l) TP016 - Add Write-Read-Verify to SSP support;
m) TP017 - Align SATA 2.6 with ATA8-ACS; and
n) TP018 - Specification Revisions For Gen3i.

1.4 Revision 3.1 (ratification date July 18, 2011)

Release that incorporates errata against Revision 3.0:

a) ECN039 - Gen3i TX Jitter Compliance Requirements;
b) ECN040 - DCO Corrections;
c) ECN041 -To correct SATA Internal 4 Lane Pin Assignments, Figure 44;
d) ECN042 - mSATA Connector Pin Counts of Vendor Specific and Reserved Pins;
e) ECN043 - Correction to description of ASR;
f) ECN044 - Mathematical CIC for Gen3i;
g) ECN045 - Interface Detect Pin for mSATA Connector and the following new features and enhancements;
h) ECN046 - IDENTIFY DEVICE words 63, 78 and 79;
i) ECN047 - Correction to description of APS status in Identify Packet Device;
j) ECN048 - RX Impedance Balance Correction;
k) ECN049 - P51 Pull-Down Resistor Value and Reference Circuit and P43 Definition;
l) ECN050 - Asymmetric Amplitude and revisions to Minimum Amplitude measurement methodology;
m) ECN051 - Change of receiver test pattern specification to include Logo Framed Composite Pattern;
n) ECN052 - Clarification of the Gen3i RX Tolerance Test Signal Amplitude Calibration Methodology;
o) ECN053 - Gen-III (6Gbps) RiseTime Specification Change;
p) ECN054 - Change to EMI Related Parameters: TX Rise/Fall Imbalance - Elimination, and TX Amplitude Imbalance - Margin Increase; and
q) ECN055 - Consistency of Register FIS nomenclature,

and the following new features and enhancements:

a) TPR003 - SSC Profile df/dt Excursion Limitation;
b) TPR019 - HOLDP/HOLDAP Protocol Change for 6G SATA;
c) TPR020 - Extensions to the FPDMA QUEUED Command Protocol to Support Fixed 512 Byte Block Transfer DMA Commands;
d) TPR021 - Speed Clarification;
e) TPR022 - Required Link Power Management;
f) TPR023 - Apply SATA 2.5 Design Guide 2 for all devices;
g) TPR024 - mSATA Connector;
h) TPR025 - NCQ Status bit 4;
i) TPR026 - Zero Power Slimline ODD;
j) TPR027 - Add Sanitize State to SSP support;
k) TPR028 - Micro SATA Connector P7 Definition;
l) TPR029 - Clarification of Speed Negotiation;
m) TPR030 - Gen1x, Gen2x Removal;
n) TPR031 - Hardware Control Feature Mechanism;
o) TPR032 - COMINIT after POR Timing;
p) TPR034 - SATA Universal Storage Module; and
q) TPR036 - NCQ Autosense.

1.5 Revision 3.2 (ratification date August 7, 2013)

Release that incorporates errata against Revision 3.1:

a) ECN056 - Tx AC Common Mode Voltage Change for Gen3;
b) ECN057 - Internal Micro SATA and Slimline Gen3;
c) ECN058 - Hardware Feature Control Correction;
d) ECN059 - Device Configuration Overlay Correction;
e) ECN061 - Dual Consecutive ALIGNp Sequence;
f) ECN062 - Tx Min Amplitude Cleanup;
g) ECN063 - Rx Rise/Fall Time Reinstated;
h) ECN064 - ErrorFlush Cleanup;
i) ECN065 - Identify Modification;
j) ECN066 - Tx AC Common Mode Voltage Procedural Simplification;
k) ECN067 - Port Multiplier Signature for Software Reset, Description of I field of Set Device Bits FIS, NCQ Queue Management Subcommand response, Typo Error in GSCR Reference;
l) ECN068 - Device Sleep Voltage Spec Adjustment;
m) ECN069 - Hardware Feature Control Bug;
n) ECN070 - PM5:PUIS Clarification;
o) ECN071 - DEVSLP Bit Interlock; and
p) ECN072 - Endianness of LBA Range List,

and the following new features and enhancements:

a) TPR033 - Relaxation of Minimum Transmit Rise/Fall Times for Gen 1 and Gen 2;
b) TPR035 - SATA BGA SSD;
c) TPR037 - Standard SATA Connector 3.3V Power Pin Assignments;
d) TPR038 - Device Sleep;
e) TPR039 - DEVSLP Assignment on Standard SATA Connector;
f) TPR040 - Software Settings Preservation for Device Initiated Interface Power Management;
g) TPR041 - 9 mm SATA USM;
h) TPR042 - Hybrid Information Feature;
i) TPR043 - Queuing Power Management;
j) TPR044 - Synch with ACS-4;
k) TPR045 - Rebuild Assist;
l) TPR046 - Transitional Energy Reporting;

m) TPR047 - SATA Express Specification;
n) TPR049 - Add QPM to SATA logs;
o) TPR050 - SATA MicroSSD Footprint Update;
p) TPR051 - Hybrid Information Update;
q) TPR053 - M.2 Card Formfactor for SSDs; and
r) TPR054 - SATA 8.5 mm Slimline ODD Connector Location.

1.6 Revision 3.3 (ratification date February 2, 2016)

Release that incorporates errata against Revision 3.2:
a) ECN073 - CIC Clarification;
b) ECN074 - SATA Express Pin Sequencing;
c) ECN075 - Identify Device data log and Word correction;
d) ECN076 - Flow control;
e) ECN077 - Automatic Partial to Slumber No NCQ;
f) ECN078 - Queued Error I-bit Correction;
g) ECN079 - TPR056 Corrections for Power Disable;
h) ECN080 - NCQ Feature Set Clarification;
i) ECN081 - Missing bit in ID data log for Rebuild Assist;
j) ECN082 - FCOMP D24.2 Correction;
k) ECN083 - Tx impedance balance cleanup;
l) ECN084 - Slimline Figure Move;
m) ECN085 - mSATA Figure Cleanup;
n) ECN086 - Clarifying NCQ Commands;
o) ECN087 - Missing bit in ID data log for Hybrid Information Enable;
p) ECN088 - Enclosure services signature; and
q) ECN089 - DSS/DAS Support Clarifications,

and the following new features and enhancements:
a) TPR056 - Enable new Power Disable feature on standard SATA connector P3;
b) TPR057 - Register Signature definition for Host Managed Zoned devices;
c) TPR058 - DAS/DSS/DHU Changes;
d) TPR059 - Emphasis Control for SATA Interface;
e) TPR060 - Modify/Cleanup Receive FPDMA Queued to support ZAC;
f) TPR061 - Define/Use Sequential NCQ command(s);
g) TPR062 - SEND/RECEIVE FPDMA Queued Cleanup;
h) TPR063 - Align Queued DATA SET MANAGEMENT with ACS-4;
i) TPR064 - Queued version of ACS-4 ZERO EXT command;
j) TPR065 - Modernize Aux Field in NCQ NON-DATA Command;
k) TPR066 - Reporting Current Write Pointer for NCQ Errors in Write Pointer Zones;
l) TPR067 - FPDMA Zone Management Commands;
m) TPR068 - Clarification of the FPDMA state machine and error reporting;
n) TPR069 - MFTP Measurement Method for Emphasis;
o) TPR070 - Retire ICC Field in SEND / RECEIVE FPDMA QUEUED;
p) TPR071 - Addition of ZAC Management In/Out to NCQ Non-Data log (12h) and NCQ Send and Receive log (13h);
q) TPR072 - Add Deferred Errors to NCQ Error log;
r) TPR073 - IDENTIFY DEVICE Revision Updates;
s) TPR074 - Obsolete Parallel ATA Emulation; and
t) TPR075 - Add DATA SET MANAGEMENT XL command.

1.7 Revision 3.4 (ratification date June 13, 2018)

Release that incorporates errata against Revision 3.3:
a) ECN090 - Correction of reference to SRST;

b) ECN091 - Correction to Power Disable;
c) ECN092 - Support Cache Behavior Bit Clarification;
d) ECN093 - DIPM and HIPM Corrections; and
e) ECN094 - Phy Event Counters Log Fix,

and the following new features and enhancements:
a) TPR076 - Durable/Ordered Write Notification;
b) TPR077 - DEADLINE HANDLING Correction;
c) TPR078 - Clarification on writeable logs;
d) TPR079 - Correction of references to tagged command queueing;
e) TPR080 - Additional DevSlp timing requirement
f) TPR081 - Out of band management control structures; and
g) TPR082 - Obsolete SATA Express.

1.8 Revision 3.5 (ratification date July 1, 2020)

Release that incorporates errata against Revision 3.4:
a) ECN095 - Misc Errors,

and the following new features and enhancements:
a) TPR083 - Gen3 Device Tx Emphasis Change;
b) TPR084 - Out of band management control structure additions;
c) TPR085 - Define/Use Ordered NCQ command(s);
d) TPR086 - Fast Fail; and
e) TPR087 - Fast Fail with Controls.

1.9 Revision 3.5a (ratification date March 2, 2021)

Changed the words ‘master’ and ‘slave’ to new terms throughout the specification.

2 Scope

This specification defines a high-speed serialized Advanced Technology Attachment (ATA) data link interface (specifying Physical (Phy), Link, Transport, and Application layers). The serialized interface uses the command set from the ACS-4 standard, augmented with Native Command Queuing (NCQ) commands optimized for the serialized interface. The serialized ATA interface is defined in a register-compatible manner with parallel ATA to enable backward compatibility with parallel ATA drivers. The physical interface is defined to ease integration (low pin count, low voltages) and enable scalable performance (with currently defined data rates of 1.5 Gbit/s, 3.0 Gbit/s, and 6.0 Gbit/s).

Complementary components are also specified including interconnect solutions for various applications, port expansion devices, and failover devices.

Normative information is provided to allow interoperability of components designed to this specification. Informative information, if provided, may illustrate possible design implementation.

3 Normative references

3.1 Normative references overview

The following standards contain provisions that, through reference in the text, constitute provisions of this standard. Every effort has been made to provide references to modern documents. Some referenced documents may no longer be available.

All standards are subject to revision, and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the standards listed below.

Copies of the following documents may be obtained from American National Standards Institute (ANSI):

a) approved ANSI standards;
b) approved and draft international;
c) regional standards (e.g., ISO, IEC, CEN/CENELEC, and ITUT); and
d) approved and draft foreign standards (e.g., including BSI, JIS, and DIN).

For further information, contact ANSI Customer Service Department at 212-642-4900 (phone), 212-302-1286 (fax) or via the World Wide Web at http://www.ansi.org.

Additional availability contact information is provided below as needed.

3.2 Approved references

The following approved ANSI standards, approved international and regional standards (e.g., ISO, IEC, CEN/CENELEC, and ITUT), may be obtained from the international and regional organizations who control them.

Information technology - AT Attachment with Packet Interface - 5 (ATA/ATAPI-5) [ANSI INCITS 340-2000]

Information technology - AT Attachment with Packet Interface - 6 (ATA/ATAPI-6) [ANSI INCITS 361-2002]

Information technology - AT Attachment with Packet Interface - 7 (ATA/ATAPI-7) [ANSI INCITS 397-2005]

Information technology - AT Attachment-8 ATA/ATAPI Command Set (ACS) [ANSI/INCITS 452-2008]

Information technology - ATA/ATAPI-8 Serial Transport (ATA8-AST) [ANSI/INCITS 493-2012 (R2017)]

Information technology - ATA/ATAPI-8 Parallel Transport (ATA8-APT) [ANSI INCITS 524-2016]

Information technology - ATA/ATAPI Command Set - 2 (ACS-2) [ANSI/INCITS 482-2012]

Information technology - ATA/ATAPI Command Set - 3 (ACS-3) [ANSI/INCITS 522-2014]

Information technology – ATA/ATAPI Command Set - 4 (ACS-4) [ANSI/INCITS 529-2018]

Information technology – Zoned Device ATA Command Set (ZAC) [ANSI INCITS 537-2016]

Information technology - Serial Attached SCSI - 4 (SAS-4) [ANSI INCITS 534-2019]

Information technology - Small Computer System Interface (SCSI) - Part 263:SAS Protocol Layer - 3 (SPL-3) [ISO/IEC 14776-263]

Information technology - SCSI-3 Enclosure Services (SES) Command Set [ANSI INCITS 305-1998]

Information technology - SCSI-3 Enclosure Services (SES) Amendment 1 [ANSI INCITS 305-1998/AM1-2000]

Information technology - SCSI Enclosure Services - 2 (SES-2) [ANSI INCITS 448-2008]

Information technology - Small Computer System Interface (SCSI) – Part 454: SCSI Primary Commands - 4 (SPC-4) [ISO/IEC 14776-454]

Information technology - ATA/ATAPI Host Adapters Standard [ANSI INCITS 370-2004]

ASME Y14.5M Dimensioning and Tolerancing

To obtain copies of these documents, contact Global Engineering or INCITS.

Global Engineering Documents, an IHS Company
15 Inverness Way East
Englewood, CO 80112-5704
USA
Web site http://global.ihs.com/
Telephone (303) 397-7956 or (303) 792-2181 or (800) 854-7179

Document Distribution
INCITS Online Store
managed by Techstreet
1327 Jones Drive
Ann Arbor, MI 48105
USA
Web site http://www.techstreet.com
Telephone (734) 302-7801 or (800) 699-9277

Additional material and draft versions available from http://www.T10.org and http://www.T13.org.

SAF-TE – SCSI Accessed Fault-Tolerant Enclosure version 1.00 [revision R041497, April 14, 1997]. Available for download at http://www.intel.com/design/servers/ipmi/pdf/sr041497.pdf.

I^2C-Bus Specification version 2.1. Available for download at https://www.i2c-bus.org.

IPMB - Intelligent Platform Management Bus Communications Protocol Specification version 1.0. Available for download at http://www.intel.com/design/servers/ipmi/spec.htm.

IPMI - Intelligent Platform Management Interface Specification version 1.5. Available for download at http://www.intel.com/design/servers/ipmi/spec.htm.

JEDEC Standards (Located on http://www.jedec.org):

a) JESD22-A114-B, Electrostatic Discharge (ESD) Sensitivity Testing Human Body Model (HBM);

b) JESD22-C101-A, Field-Induced Charged-Device Model Test Method for Electrostatic-Discharge-Withstand Thresholds of Microelectronic Components;
c) MO-300, mSATA SSD Assembly; and
d) MO-276E - Standard and Low Profile, Rectangular Fine Pitch Ball Grid Array Family 0.50mm Pitch.

The following specifications published by the Storage Networking Industry Association – SFF Technology Affiliate are referenced:

a) SFF-8086;
b) SFF-8087;
c) SFF-8088;
d) SFF-8111;
e) SFF-8144;
f) SFF-8146;
g) SFF-8201;
h) SFF-8301;
i) SFF-8470;
j) SFF-8482;
k) SFF-8484;
l) SFF-8553;
m) SFF-8609
n) SFF-8630;
o) SFF-8639;
p) SFF-8680; and
q) INF-8280.

For more information on the current status of SFF documents, contact the Storage Networking Industry Association (SNIA) (see www.snia.org/sff).

The following EIA-364-xx standards published by Electronic Industries Alliance (EIA) are referenced.
To obtain copies of these documents, contact Global Engineering:

a) EIA-364-09, Durability Test Procedure for Electrical Connectors and Contacts;
b) EIA-364-13, Mating and Unmating Forces Test Procedure for Electrical Connectors;
c) EIA-364-17, Temperature Life with or without Electrical Load Test Procedure for Electrical Connectors and Sockets;
d) EIA-364-18, Visual and Dimensional Inspection for Electrical Connector;
e) EIA-364-20, Withstanding Voltage Test Procedure for Electrical Connectors, Sockets and Coaxial Contacts;
f) EIA-364-21, Insulation Resistance Test Procedure for Electrical Connectors, Sockets, and Coaxial Contacts;
g) EIA-364-23, Low Level Contact Resistance Test Procedure for Electrical Connectors and Sockets;
h) EIA-364-27, Mechanical Shock (Specified Pulse) Test Procedure for Electrical Connectors;
i) EIA-364-28, Vibration Test Procedure for Electrical Connectors and Sockets;
j) EIA-364-31, Humidity Test Procedure for Electrical Connectors and Sockets;
k) EIA-364-32, Thermal Shock (Temperature Cycling) Test Procedure for Electrical Connectors and Sockets;
l) EIA-364-38, Cable Pull-Out Test Procedure for Electrical Connectors;
m) EIA-364-41, Cable Flexing Test Procedure for Electrical Connectors; and
n) EIA-364-65, Mixed Flowing Gas.

The following form factor standards published by EIA are referenced.
These standards may be obtained from Global Engineering:

a) EIA-740, specification for small form factor 3.5” disk drives; and

b) EIA-720, specification for small form factor 2.5” disk drives.

The following PCI Express® (PCIe) Specifications are available from http://www.pcisig.com:

a) PCI Firmware Specification, Revision 3.2;
b) PCI Express M.2 specification, Revision 1.2;
c) PCI Express Local Bus Specification, Revision 5.0; and
d) PCI Express Card Electromechanical Specification, Revision 4.0. Commonly known as the “PCIe CEM” spec.

3.3 References under development

The following ANSI standards under development are referenced.
Draft versions of these standards are available from http://www.T10.org or http://www.T13.org:

a) Information technology - ATA/ATAPI Command Set - 5 (ACS-5) [INCITS BSR T13/558]; and
b) Information technology – Zoned Device ATA Command Set - 2 (ZAC-2) [INCITS BSR T13/549].

3.4 Other references

National Institute of Standards and Technology (NIST), http://www.nist.gov

The 8b/10b code used in Serial ATA is based on the following published references:

a) A.X. Widmer and P.A. Franaszek, “A DC-Balanced, Partitioned-Block, 8B/10B Transmission Code.” IBM Journal of Research and Development, 27, no. 5: 440-451 (September, 1983); and
b) U.S. Patent 4,486,739. Peter A. Franaszek and Albert X. Widmer. Byte Oriented DC Balanced (0,4) 8B/10B Partitioned Block Transmission Code. (Dec. 4, 1984).

Serial Peripheral Interface Bus (SPI) – SPI is a de facto standard for a general purpose, synchronous serial interface.
See http://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus.

3.5 SATA-IO Style Guide

Serial ATA Design Guide #004 version 22 dated October 9, 2015 is the SATA-IO Style Guide used for this specification. This design guide is available from http://www.sata-io.org.

3.6 Design guides

Design guides are available from http://www.sata-io.org.

4 Definitions, abbreviations, and conventions

4.1 Terminology

4.1.1 Definitions and abbreviations

4.1.1.1 active port

The active port is the currently selected host port on a Port Selector.

4.1.1.2 Advanced Technology Attachment (ATA)

Advanced Technology Attachnment (ATA) defines the physical, electrical, transport, and command protocols for the attachment of storage devices.

4.1.1.3 Advanced Technology Attachment Packet Interface (ATAPI) device

A device implementing the Packet Command feature set.

4.1.1.4 $ALIGN_P$ sequence

A continuous stream of consecutive $ALIGN_P$ primitives.

4.1.1.5 bit synchronization

The state that a receiver has synchronized the internal receiver clock to the external transmitter and is delivering retimed serial data.

4.1.1.6 bitrate

Reciprocal of the unit interval (UI). Bitrate = 1 / UI.

4.1.1.7 burst

A short pulse of data starting from and ending with the idle condition on the interface. These are used for Out Of Band (OOB) signaling.

4.1.1.8 byte (B)

A byte is an ordered set of eight (8) bits. The least significant bit is bit 0 and the most significant bit is bit 7.

4.1.1.9 caching medium

A caching medium is an optional medium that may contain a subset of user data from the primary medium. The caching medium may contain data that is newer or identical to the data on the primary medium.

4.1.1.10 character

A character is a representation of a byte in the Zxx.y notation (see 9.3.2).

4.1.1.11 character alignment

Character alignment is a receiver action that resets the character boundary to that of the comma sequence found in the K28.5 control character of $ALIGN_P$, and establishes Dword synchronization of the incoming serial data stream.

4.1.1.12 character slipping

Character slipping is the receiver action that realigns the receiver's clock to the received bit stream by adding or removing bit times within the characters of $ALIGN_P$.

4.1.1.13 ClickConnect

An optional positive latch solution for internal single lane interconnects (see 6.2.4).

4.1.1.14 Closed Loop Transfer Function (CLTF)

For a feedback system, the Closed Loop Transfer Function (CLTF) is the ratio of the magnitude and phase of the output variable to the magnitude and phase of the input variable, as a function of frequency for sinusoidal excitation. This term is used in the Reference Clock sections of this specification.

4.1.1.15 code violation

A code violation is an error that occurs in the reception process as a result of:

a) a running disparity violation;
b) an encoded character that does not translate to a valid data or control character;
c) an encoded character that translates to a control character other than:
    A. K28.5; or
    B. K28.3 in byte 0 of a Dword;
    or
d) an encoded character that translates to any control character (valid or invalid) in bytes 1 to 3 of a Dword.

4.1.1.16 comma character

A comma character is a control character, that when encoded, contains the comma sequence. In Serial ATA the only comma character used is K28.5, and only $ALIGN_P$ contains the comma character. The comma sequence is the first seven bits of the encoded character.

4.1.1.17 comma sequence

The comma sequence is a seven bit sequence of 001 1111b or 110 0000b in an encoded stream. The comma sequence is unique in that it appears only in a single encoded character, and furthermore, may not appear in any subset of bits in adjacent encoded characters. This unique property allows the comma sequence to be used for determining alignment of the received data stream.

4.1.1.18 command aborted

Command aborted is command completion with the ABRT bit set to one in the Error register, and the ERR bit set to one in the Status register.

4.1.1.19 command completion

Command completion describes the completion of an action requested by command, applicable to the device.

Command completion also applies to the case where the command has terminated with an error, and the following actions occurred:

a) the appropriate bits of the Status register have been updated;
b) the BSY bit and the DRQ bit have been cleared to zero; and

c) assertion of INTRQ if nIEN is active-low, assuming that the command protocol specifies INTRQ to be asserted.

In Serial ATA, the register contents are transferred to the host using a Register Device to Host FIS.

4.1.1.20 command packet

A data structure transmitted to the device during the processing of a PACKET command that includes the command and command parameters.

4.1.1.21 Compliance Interconnect Channel (CIC)

A Compliance Interconnect Channel (CIC) is defined as a set of calibrated physical test circuits applied to the Transmitter mated connector (see 7.4.8).

4.1.1.22 concentrator

A concentrator is a generic term used to describe a component that has multiple Serial ATA ports to connect to Serial ATA devices plus some small number of ports to connect to a host. In the simplest case a concentrator may be a Host Bus Adapter (HBA) that is plugged into the host that connects to some number of Serial ATA devices (e.g., like a PCI Serial ATA controller card). A concentrator may also be an internal or external redundant array of independent disks (RAID) controller (e.g., a fibre-channel to Serial ATA RAID controller), or may be some element that expands the number of ports through a fan-out scheme, like a Port Multiplier.

4.1.1.23 Control Block registers

Control Block registers are interface registers used for device control and to post alternate status.

4.1.1.24 control character

A control character is a character where Z is equal to K (see 9.3.2).

4.1.1.25 control port

The Port Multiplier has one port address reserved for control and status communication with the Port Multiplier itself. The control port has port address Fh.

4.1.1.26 control variable

The control variable, Z, is a flag that determines the code set to be used to interpret a data byte. The control variable has the value D (for data characters) or K (for control characters).

4.1.1.27 Cyclic Redundancy Check (CRC)

An error checking mechanism that checks data integrity by computing a polynomial algorithm based checksum. In Serial ATA a 32 bit Cyclic Redundancy Check (CRC) is calculated over the contents of a Frame Information Structure. The Serial ATA CRC is the Dword in a frame that immediately precedes EOF_P.

4.1.1.28 data character

A data character is a character where Z is equal to D (see 9.3.2).

4.1.1.29 data signal source

An instrument that provides a Serial ATA data signal.

4.1.1.30 deterministic jitter (DJ)

All jitter sources that have bounded probability distribution functions (i.e., values outside the bounds have probability zero).
Four kinds of deterministic jitter are identified:

a) duty cycle distortion;
b) data dependent (ISI);
c) sinusoidal; and
d) uncorrelated (to the data) bounded.

Deterministic jitter (DJ) is characterized by its bounded, peak-to-peak value.

4.1.1.31 device

Device is a storage peripheral. Traditionally, a device on the interface has been a hard disk drive, but any form of storage device may be placed on the interface provided the device adheres to this specification and to an ATA standard.

4.1.1.32 device port

The device port is a port on a Port Multiplier or a Port Selector that may be connected to a device. Port Multipliers may have up to 15 device ports. Port Selectors have one device port.

4.1.1.33 device sleep (DEVSLP) signal

The signal which controls entry into and exit from the DevSleep interface power state (see 8.5).

4.1.1.34 differential signal

The differential signal is the voltage on the positive conductor minus the voltage on the negative conductor (i.e., Tx+ – Tx-).

4.1.1.35 Direct Current block (DCB)

The Direct Current block (DCB) is defined as a device that passes frequencies from 10 MHz to at least 12 GHz with minimal effect on the amplitude or phase of the signal.

4.1.1.36 Direct Head Unload (DHU)

An active high electrical signal used to request that a device with rotating media retract the heads from the media (see 6.13.2 and 13.19).

4.1.1.37 direct memory access (DMA)

Direct memory access (DMA) is a means of data transfer between device and host memory without host processor intervention.

4.1.1.38 dirty data

Dirty data is user data in a caching medium that is newer than the corresponding data in the primary medium.

4.1.1.39 Dword (DW)

A double word (DW) is an ordered set of thirty-two (32) bits. The least significant bit is bit 0 and the most significant bit is bit 31.

4.1.1.40 Dword synchronization

The state that a receiver has recognized the comma sequence and is producing an aligned data stream of Dwords (four contiguous bytes) from the zero-reference of the comma character.

4.1.1.41 elasticity buffer

The elasticity buffer is a portion of the receiver where character slipping or character alignment is performed.

4.1.1.42 encoded character

An encoded character is the output of the 8b/10b encoder – the result of encoding a character. An encoded character consists of 10 bits, where bit 0 is the most significant bit and bit 9 is the least significant. The bits in an encoded character are symbolically referred to as “abcdeifghj” where “a” corresponds to bit 0 and “j” corresponds to bit 9. Case is significant. See 9.3 for a description of the relationship between bytes, characters and encoded characters.

NOTE 1 - Note the out-of-order representation “abcdeifghj”

4.1.1.43 endpoint device

An endpoint device is an ATA or ATAPI device, as reported by the device signature after power-on or reset. This may include hard disk drives, optical disk drives, and tape drives.

4.1.1.44 eSATA

The system-to-system interconnects - External Desktop Applications usage model (see 5.3.7).

4.1.1.45 evict

Evict is a process within the device to remove data from the caching medium.

4.1.1.46 Explicit Clock

Bit rate clock derived from the pattern generator.

4.1.1.47 First-party DMA access

First-party DMA access is a method that a device accesses host memory.

4.1.1.48 First-party DMA Data Phase

The First-party DMA Data Phase is the period from the reception of a DMA Setup FIS until either the exhaustion of the associated data transfer count or the assertion of the ERR bit in the shadow Status register.

4.1.1.49 frame

A frame is an indivisible unit of information exchanged between a host and device. A frame consists of SOF_P, a FIS, a CRC calculated over the contents of the FIS, and EOF_P.

4.1.1.50 Frame Error Rate (FER)

Details as given in 7.4.3.1.3.

4.1.1.51 Frame Information Structure (FIS)

The user payload of a frame, does not include the SOF_P, CRC, and EOF_P delimiters.

4.1.1.52 Frequency baud (Fbaud)

The nominal rate of data through the channel, measured in GHz.

4.1.1.53 Gen1

Refers to first generation signaling characterized by a speed of 1.5 Gbit/s. Details as given in 7.2.

4.1.1.54 Gen1i

The internal electrical specifications at 1.5 Gbit/s with cable lengths up to 1 m.

4.1.1.55 Gen1m

The electrical specifications used in Short Backplane Applications, External Desktop Applications, and Data Center Applications with cable lengths up to two m, defined at 1.5 Gbit/s.

4.1.1.56 Gen1u

The electrical specifications defined at 1.5 Gbit/s for Universal Host (UHost) applications. Because a Gen1i/Gen2i/Gen3i endpoint device is a direct connection to the mating connection of the UHost, the electrical specifications for Gen1u allow a channel loss up to the approximate equivalence of a 1 m data cable plus mated connector pair within the UHost.

4.1.1.57 Gen1x (Obsolete)

The electrical specifications used in Long Backplane Applications and Data Center Applications supporting cable lengths up to and greater than two meters, defined at 1.5 Gbit/s.

4.1.1.58 Gen2

Refers to second generation signaling characterized by a speed of 3.0 Gbit/s. Details as given in 7.2.

4.1.1.59 Gen2i

The internal electrical specifications at 3.0 Gbit/s with cable lengths up to 1 m.

4.1.1.60 Gen2m

The electrical specifications used in Short Backplane Applications, External Desktop Applications, and Data Center Applications with cable lengths up to two meters, defined at 3.0 Gbit/s.

4.1.1.61 Gen2u

The electrical specifications defined at 3.0 Gbit/s for UHost applications. Because a Gen1i/Gen2i/Gen3i endpoint device is a direct connection to the mating connection of the

UHost, the electrical specifications for Gen2u allow a channel loss up to the approximate equivalence of a 1 m data cable plus mated connector pair within the UHost.

4.1.1.62 Gen2x (Obsolete)

The electrical specifications used in Long Backplane Applications and Data Center Applications supporting cable lengths up to and greater than two meters, defined at 3.0 Gbit/s.

4.1.1.63 Gen3

Refers to third generation signaling characterized by a speed of 6.0 Gbit/s.

4.1.1.64 Gen3i

The internal electrical specifications at 6.0 Gbit/s with cable lengths up to 1 m.

4.1.1.65 Gen3u

The electrical specifications defined at 6.0 Gbit/s for UHost applications. Because a Gen1i/Gen2i/Gen3i endpoint device is a direct connection to the mating connection of the UHost, the electrical specifications for Gen3u allow a channel loss up to the approximate equivalence of the Gen3i CIC (see 7.4.8) plus mated connector pair within the UHost.

4.1.1.66 Hardware Feature Control pin(s)

For the LIF-SATA connector, Hardware Feature Control (see 13.10) are connector pins P8 and P21. For the 1.8 inch Micro SATA connector, Hardware Feature Control is connector pin P7. For all other connectors Hardware Feature Control is connector pin P11.

4.1.1.67 High Bandwidth Scope (HBWS)

An oscilloscope with an analog bandwidth of 10 GHz or greater in the measurement path.

4.1.1.67.1 HBWS Gen1/Gen2 Requirement

For Gen1/Gen2 measurements, an oscilloscope with an analog bandwidth of 10 GHz or greater shall be used in the measurement path.

4.1.1.67.2 HBWS Gen3 Requirement

For Gen3 measurements, an oscilloscope with an analog bandwidth of 12 GHz or greater shall be used in the measurement path.

4.1.1.68 High Frequency Test Pattern (HFTP)

This pattern provides the maximum frequency allowed within the Serial ATA encoding rules. Pattern 1010 1010 1010 1010 1010b = encoded D10.2. The pattern is repetitive.

4.1.1.69 Host Bus Adapter (HBA)

Host Bus Adapter (HBA) is a component that connects to the host system’s expansion bus to provide connectivity for devices. HBAs are also often referred to as controller cards or merely controllers.

4.1.1.70 host port

The host port is the port that is used to connect the Port Multiplier or Port Selector to a host. Port Multipliers have one host port. Port Selectors have two host ports.

4.1.1.71 hot plug

The connection of a SATA device to a host system that is already powered. The SATA device is already powered or powered upon insertion/connection. See 7.4.6.1 for details on hot plug scenarios.

4.1.1.72 hybrid device

A hybrid device is a device that contains both a primary medium and a non-volatile caching medium.

4.1.1.73 immediate NCQ command

An immediate NCQ command is an NCQ command that is defined to be processed:

a) after any command previously accepted by the device that the device has transmitted a DMA Setup FIS and has not reached command completion; and
b) before any NCQ command previously accepted by the device that the device has not transmitted a DMA Setup FIS.

4.1.1.74 inactive port

The inactive port is the host port that is not currently selected on the Port Selector.

4.1.1.75 interrupt pending

Interrupt pending is an internal state of the device that exists when the device protocol requires the device to notify the host of an event by asserting INTRQ, given the condition where nIEN is asserted active-low to zero.

4.1.1.76 inter-symbol interference (ISI)

Data-dependent deterministic jitter caused by the time differences required for the signal to arrive at the receiver threshold when starting from different places in bit sequences (symbols).

Example - Media attenuates the peak amplitude of the bit sequence 0101b, more than it attenuates the peak amplitude of the bit sequence 0000 1111b, thus the time required to reach the receiver threshold with the 0101b sequence is less than required from the 0000 1111b sequence.

The run length of 4 produces a higher amplitude that takes more time to overcome when changing bit values and therefore produces a time difference compared to the run length of 1 bit sequence. When different run lengths are mixed in the same transmission the different bit sequences (symbols) therefore interfere with each other. Inter-symbol interference (ISI) is expected whenever any bit sequence has frequency components that are propagated at different rates by the transmission media. This translates into a high level of high-frequency, data-dependent, jitter.

4.1.1.77 jitter measuring device (JMD)

A device used to measure jitter.

EXAMPLE - Examples are a bit error rate tester (BERT), a timing interval analyzer (TIA), a single shot capture oscilloscope and processing software, or a HBWS.

4.1.1.78 Jitter Transfer Function (JTF)

In general terms, the jitter transfer function (JTF) of a system is the ratio of the jitter magnitude and phase of the output variable to the jitter magnitude and phase of the input variable, as a function of frequency for sinusoidal jitter excitation. In the case of a jitter definition, this defines the magnitude of the jitter, as a function of frequency allowed to be generated by the transmitter or tolerated by the receiver. In the case of a JMD, this defines the ratio of the reported jitter to the applied jitter, as a function of frequency for sinusoidal excitation.

4.1.1.79 junk

An 8b/10b encoded data Dword sent between $CONT_P$ and another primitive transmitted on the link. All junk Dwords shall be ignored by the receiver.

4.1.1.80 laboratory load or lab-load (LL)

An electrical test system connected to the unit under test (UUT). The laboratory load (LL) receives a signal from the UUT at the defined impedance level of 100 ohm differential and 25 ohm common mode.

4.1.1.81 laboratory sourced signal or Lab-Sourced Signal (LSS)

An instrument and electrical test system connected to the UUT. The laboratory sourced signal (LSS) provides a signal to the UUT at the defined impedance level of 100 ohm differential and 25 ohm common mode.

4.1.1.82 legacy mode

Legacy mode is the mode of operation that provides software-transparent communication of commands and status between a host and device using the ATA Command Block and Control Block registers.

4.1.1.83 legal character

A legal character is one for which there exists a valid decoding, either into the data character or control character fields. Due to running disparity constraints not all 10 bit combinations result in a legal character. Additional usage restrictions in Serial ATA result in a further reduction in the SATA defined control character space.

4.1.1.84 Linear Feedback Shift Register (LFSR)

See details as given in 9.5.7 for using a Linear Feedback Shift Register (LFSR) in scrambling.

4.1.1.85 Logical Block Address (LBA)

As defined in the ACS-4 standard.

4.1.1.86 Lone Bit Pattern (LBP)

This pattern as defined in 7.4.5.4.6. The pattern is repetitive.

4.1.1.87 low frequency test pattern (LFTP)

This pattern provides a low frequency, that is allowed within the Serial ATA encoding rules. Pattern 0111 1000 1110 0001 1100b = encoded D30.3. The pattern is repetitive.

4.1.1.88 mapping resource

A mapping resource is a vendor specific mechanism that may be used by the device internally to describe the physical location and attributes of user data.

4.1.1.89 mid frequency test pattern (MFTP)

This pattern provides a middle frequency that is allowed within the Serial ATA encoding rules. Pattern 1100 1100 1100 1100 1100b = encoded D24.3. The pattern is repetitive.

4.1.1.90 NCQ command

One of the mandatory or optional commands in the NCQ feature set (see 13.6.1).

4.1.1.91 NCQ Non-Streaming command

An NCQ Non-Streaming command is a command using the FPDMA QUEUED protocol that the Priority (PRIO) field is not set to 01b, Isochronous - deadline dependent priority.

4.1.1.92 NCQ Streaming command

An NCQ Streaming command is a command using the FPDMA QUEUED protocol that the PRIO field is set to 01b, Isochronous - deadline dependent priority.

4.1.1.93 Non-NCQ command

A command that is not an NCQ command.

4.1.1.94 Ordered NCQ command

An ordered NCQ command is an NCQ command that is defined to be processed:

a) after any NCQ commands previously accepted by the device have completed processing; and
b) before any NCQ commands subsequently accepted by the device have begun processing.

4.1.1.95 OS-aware hot plug

The insertion of a SATA device into a backplane that has power shutdown. The backplane is later powered, and both the device and the host power up, and the host-initiated OOB sequence determines the time that SATA operations begin.

4.1.1.96 OS-aware hot removal

The removal of a SATA device from a powered backplane, that has been first placed in a quiescent state.

4.1.1.97 Out Of Band Management interface

The management interface defined in 13.22.

4.1.1.98 Out Of Band (OOB) signaling

Out Of Band (OOB) signaling is a pattern of $ALIGN_P$ primitives or Dwords composed of D24.3 characters and idle time and is used to initialize the Serial ATA interface. OOB signaling is also used to recover from low power states and to signal specific actions during test modes (see clause 8).

4.1.1.99 PCI Express® (PCIe)

The PCI Express IO bus standard, see http://www.pcisig.com for a list of the most relevant PCIe documents available from Peripheral Component Interconnect - Special Interest Group (PCI-SIG).

4.1.1.100 PCI Express® Generation 2 (PCIe Gen2)

In this context PCIe Gen2 is used to refer to a link speed of 5 GT/s using 8b/10b encoding.

4.1.1.101 PCI Express® Generation 3 (PCIe Gen3)

In this context PCIe Gen3 is used to refer to a link speed of 8 GT/s using 128b/130b encoding with scrambling.

4.1.1.102 PCI Express® x2 (PCIe x2)

x2 means using 2 PCIe lanes for communication. This is 2 differential pairs in each direction. Total signals is 2 lanes times 2 (differential) times 2 (Transmit and receive) = 8 signals.

4.1.1.103 Phy offline

In this mode the host Phy is forced off and the host Phy does not recognize nor respond to COMINIT or COMWAKE. This mode is entered by setting the DET field of the SControl register to 0100b. This is a mechanism for the host to turn off its Phy.

4.1.1.104 Physical Region Descriptor (PRD)

A Physical Region Descriptor (PRD) table is a data structure used by DMA engines that comply with the ATA/ATAPI Host Adapters standard. The PRD describes memory regions to be used as the source or destination of data during DMA transfers. A PRD table is often referred to as a scatter/gather list.

4.1.1.105 port address

The control port and each device port present on a Port Multiplier have a port address. The port address is used to route FISes between the host and a specific device or the control port.

4.1.1.106 Power Disable

Feature that disables power to the SATA device circuitry within the SATA device when power is present at the SATA device power connector.

4.1.1.107 power disable (PWDIS) signal

Power Disable control signal.

4.1.1.108 primary medium

The primary medium is the medium to which all data is synchronized.

4.1.1.109 primitive

A primitive is a special Dword used by the Link layer for the transport control. Byte 0 of each primitive is a control character.

4.1.1.110 programmed input/output (PIO)

Programmed input/output (PIO) is a means of accessing device registers. PIO is also used to describe one form of data transfers. PIO data transfers are performed by the host processor utilizing PIO register accesses to the Data register.

4.1.1.111 protocol-based port selection

Protocol-based port selection is a method that may be used by a host to select the host port that is active on a Port Selector. Protocol-based port selection uses a sequence of Serial ATA OOB Phy signals to select the active host port.

4.1.1.112 quiescent power condition

Entering a quiescent power condition for a particular Phy is defined as the Phy entering the idle bus condition as defined in 7.7.2.

4.1.1.113 random jitter (RJ)

Random jitter (RJ) is Gaussian. Random jitter is equal to the peak to peak value of 14 times the 1σ standard deviation value given the 10^{-12} BER requirement.

4.1.1.114 reset speed negotiation (RSN)

The use of COMRESET to accomplish speed negotiation.

4.1.1.115 sector

A set of data bytes accessed and referenced as a unit.

4.1.1.116 Self-Monitoring, Analysis, and Reporting Technology (SMART)

Self-Monitoring, Analysis, and Reporting Technology (SMART) for prediction of device degradation or faults.

4.1.1.117 sequential NCQ command

A sequential NCQ command is an NCQ command that is defined to be processed:

a) after any sequential NCQ commands previously accepted by the device have completed processing; and
b) before any sequential NCQ commands subsequently accepted by the device have begun processing.

4.1.1.118 Serial ATA Enclosure Management Bridge (SEMB)

A Serial ATA Enclosure Management Bridge (SEMB) is a component that translates Serial ATA transactions into I^2C transactions to communicate enclosure services commands to a Storage Enclosure Processor (SEP).

4.1.1.119 Shadow Register Block registers

Shadow Register Block registers are interface registers used for delivering commands to the device or posting status from the device.

4.1.1.120 side-band port selection

Side-band port selection is a method that may be used by a host to select the host port that is active on a Port Selector. Side-band port selection uses a mechanism that is outside of the Serial ATA protocol for determining which host port is active. The port selection mechanism used in implementations that support side-band port selection is outside the scope of this specification.

4.1.1.121 Small Form Factor (SFF) standards

Small Form Factor (SFF) Committee is an ad hoc group that defined the 2.5 inch drive and related specification. Documents created by the SFF Committee are submitted to bodies such as Electronic Industries Association (EIA) or an Accredited Standards Committee (ASC).

4.1.1.122 software reset (SRST)

Software reset (SRST) performed by sending FISs setting and clearing the SRST bit (see 11.4).

4.1.1.123 Solid State Drive (SSD)

Solid State Drive (SSD) uses non-volatile memory to store data.

4.1.1.124 Spread Spectrum Clocking (SSC)

The technique of modulating the operating frequency of a signal slightly to spread its radiated emissions over a range of frequencies. This reduction in the maximum emission for a given frequency helps meet radiated emission requirements.

4.1.1.125 Storage Enclosure Processor (SEP)

A Storage Enclosure Processor (SEP) is a component that interfaces with the various enclosure sensors and actuators in an enclosure and is controlled through an I^2C interface to the Serial ATA Enclosure Management Bridge.

4.1.1.126 surprise hot plug

The insertion of a SATA device into a backplane that has power present. The device powers up and initiates an OOB sequence.

4.1.1.127 surprise hot removal

The removal of a SATA device from a powered backplane, without first being placed in a quiescent state.

4.1.1.128 sync

Sync is a process within the device where dirty data in the caching medium is copied to the primary medium.

4.1.1.129 SYNC Escape

The condition when $SYNC_P$ is used to escape from the present FIS transmission on the interface and resynchronize the link back to an IDLE state. The most common use of the SYNC Escape mechanism is to bring the link to an IDLE condition in order to send a Register Host to Device FIS with the C bit cleared to zero with a software reset to the device, but may be used at other times to recover link communication from erroneous conditions. A SYNC Escape is requested by the Transport layer to the Link layer.

4.1.1.130 time domain reflectometer (TDR)

An instrument used to test the impedance of the UUT.

4.1.1.131 timing interval analyzer (TIA)

Timing interval analyzer (TIA) with duty cycle distortion and ISI noise floor performance of better than 5 % of a UI for K28.5 with less than 67 picoseconds rise and fall times.

4.1.1.132 total jitter (TJ)

Unless further specified by including the BER level, total jitter (TJ) has a peak to peak value of (14 × RJ) + DJ. For Gen3i Tx jitter, TJ(10^{-12}) and TJ(10^{-6}) are specified and measured directly. RJ and DJ are not directly specified.

4.1.1.133 unit interval (UI)

Equal to the time required to transmit one bit (e.g., 666.667 picoseconds for Gen1).

4.1.1.134 unit under test (UUT)

The product under test and the other half of the “mated“ connector that is physically on the lab-load but considered part of the UUT.

4.1.1.135 Universal Host (UHost)

A SATA host that provides for attachment of a Gen1i/Gen2i/Gen3i endpoint device directly to the mating connection of the UHost. The UHost may include a channel loss up to the approximate equivalent of the 1 m cable plus mated connector pair or the Gen3i CIC (see 7.4.8) plus mated connector pair. The UHost does not support the attachment of additional interconnections between the UHost mating connection and the endpoint device connection. See 5.3, Table 1, and Table 2 for usage model applicability.

4.1.1.136 unrecoverable error

An unrecoverable error is defined as having occurred at any point when the device sets either the ERR bit or the DF bit set to one in the Status register at command completion.

4.1.1.137 vector network analyzer (VNA)

An instrument used to test the impedance of the UUT.

4.1.1.138 warm plug

Device connection with host controller powered and power at connector pins off (un-powered). This mechanism is used in Slimline applications, according to 6.4.

4.1.1.139 word

A word is an ordered set of sixteen (16) bits. The least significant bit is bit 0 and the most significant bit is bit 15.

4.1.1.140 xSATA

The system-to-system interconnects - Data Center Applications usage model (see 5.3.6).

4.1.1.141 zero crossing

To locate the zero crossing of a Data Eye, turn on the horizontal histogram function to horizontally enclose all waveforms associated with the “edge” and vertically limit to ± 5 % of the waveform voltage. The “zero crossing” is the location of the mean of the waveforms.

4.1.2 Abbreviations

Abbreviations used in this standard:

Abbreviation	Meaning
AC	Alternating Current
AFE	Analog Front End
AnyDword	Any Dword (see 9.7.1)
ASCII	American Standard Code for Information Interchange
ATA	Advanced Technology Attachment (see 4.1.1.2)
ATAPI	Advanced Technology Attachment Packet Interface (see 4.1.1.3)
AWG	American wire gauge
b	binary (see 4.2.4)
BBU	Battery Back-up Unit
BER	Bit Error Rate
BERT	Bit Error Rate Tester
BIOS	Basic Input/Output System
BIST	Built In Self Test
CBDS	Continuous Background Defect Scanning
CEM	Card Electromechanical
CIC	Compliance Interconnect Channel (see 4.1.1.21)
CLTF	Closed Loop Transfer Function (see 4.1.1.14)
CM	Common Mode
COMP	Composite Pattern (see 7.4.5.4.7)
CRC	Cyclic Redundancy Check (see 4.1.1.25)
DAS	Device Activity Signal (see 6.13.3)
DC	Direct Current
DCB	Direct Current Block (see 4.1.1.35)
DETO	DevSleep Exit Timeout (see 8.5)
DevSleep	Device Sleep interface power state (see 8.1)
DEVSLP	Device Sleep Signal (see 4.1.1.33)
DHU	Direct Head Unload (see 4.1.1.36)
DIPM	Device Initiated Power Management
DJ	Deterministic Jitter (see 4.1.1.30)
DMA	Direct Memory Access (see 4.1.1.37)
DMDT	DEVSLP Minimum Detection Time (see 8.5)
DNU	Do Not Use
DP	Device Present
DSM	Data Set Management

Abbreviation	Meaning
DSS	Disable Staggered Spinup (see 6.13.4)
DW	Dword (see 4.1.1.39)
Dword	Double Word (see 4.2.8)
EIA	Electronic Indurtries Alliance
EMI	Electromagnetic Interference
eSATA	eSATA usage model (see 5.3.7)
ESD	Electrostatic Discharge
Fbaud	Frequency baud (see 4.1.1.52)
FCOMP	Frame Composite Pattern (see 7.4.5.4.8)
FER	Frame Error Rate (see 4.1.1.50)
FIFO	First In First Out
FIS	Frame Information Structure (see 4.1.1.51)
FPC	Flexable Printed Circuit (see 5.3.11.6)
FS	Feature Specific
GPL	General Purpose Logging (see ACS-4)
GND	Ground
h	hexadecimal (see 4.2.4)
HBA	Host Bus Adapter (see 4.1.1.69)
HBWS	High Bandwidth Scope (see 4.1.1.67)
HDD	Hard Disk Drive
HFTP	High Frequency Test Pattern (see 4.1.1.68)
HTDP	High Transition Density Pattern (see 7.4.5.4.3)
ISI	Inter-Symbol Interference (see 4.1.1.76)
JBOD	Just a Bunch of Disks
JMD	Jitter Measurement Device (see 4.1.1.77)
JTF	Jitter Transfer Function (see 4.1.1.78)
LBA	Logical Block Address (see 4.1.1.85)
LBP	Lone Bit Pattern (see 4.1.1.86)
LED	Light Emitting Diode
LFSCP	Low Frequency Spectral Content Pattern (see 7.4.5.4.4)
LFSR	Linear Feedback Shift Register (see 4.1.1.84)
LFTP	Low Frequency Test Pattern (see 4.1.1.87)
LL	Lab-Load (see 4.1.1.80)
LSB	Least Significant Bit
LSS	Lab-Sourced Signal (see 4.1.1.81)
LTDP	Low Transition Density Pattern (see 7.4.5.4.2)
M.2	M.2 usage model (see 5.3.11.8)
MDAT	Minimum DEVSLP Assertion Time (see 8.5)
MFTP	Mid Frequency Test Pattern (see 4.1.1.89)
Micro SATA	Micro SATA connector (see 6.3)
MicroSSD	MicroSSD usage model (see 5.3.11.7)

Abbreviation	Meaning
mSATA	mSATA usage model (see 5.3.11.5)
MSB	Most Significant Bit
NCQ	Native Command Queuing
ODD	Optical Disk Drive
OOB	Out Of Band (see 4.1.1.98)
PCB	Printed Circuit Board
PCIe	Peripheral Component Interconnect Express (see 4.1.1.99)
Phy	Physical
PIO	Programmed Input/Output (see 4.1.1.110)
PLL	Phase Lock Loop
PRD	Physical Region Descriptor (see 4.1.1.104)
PWDIS	Power Disable (see 4.1.1.107)
Qword	Quad Word (see 4.2.8)
RA	Right Angle
RAID	Redundant Array of Independent Disks
RD+ or rd+	Running Disparity positive (see 9.3.3.3)
RD- or rd-	Running Disparity negative (see 9.3.3.3)
RJ	Random Jitter (see 4.1.1.113)
RSN	Reset Speed Negotiation (see 4.1.1.114)
Rx	Receiver
SAS	Serial Attached SCSI
SATA	Serial ATA
SATA Express	SATA Express usage model (obsolete)
SCSI	Small Computer System Interface
SDB	Set Device Bits (see 10.5.7)
SEMB	Serial ATA Enclosure Management Bridge (see 4.1.1.117)
SEP	Storage Enclosure Processor (see 4.1.1.125)
SES	SCSI Enclosure Services
SFF	Small Form Factor (see 4.1.1.121)
SMA	SubMiniture version A
SMART	Self-Monitoring, Analysis, and Reporting Technology (see 4.1.1.116)
SMT	Surface Mount Technology
SRIS	Separate RefClk Independent SSC
SRST	Software reset (see 4.1.1.122)
SSC	Spread Spectrum Clocking (see 4.1.1.124)
SSD	Solid State Drive (see 4.1.1.123)
SSHD	Solid State Hybrid Device (see 4.1.1.72)
SSOP	Simultaneous Switching Outputs Pattern (see 7.4.5.4.5)
SSP	Software Settings Preservation
TCTF	Transmitter Compliance Test Pattern (see 7.4.8)

Abbreviation	Meaning
TDR	Time Domain Reflectometry (see 4.1.1.130)
TIA	Time Interval Analyzer (see 4.1.1.131)
TJ	Total Jitter (see 4.1.1.132)
Tx	Transmitter
UHost	Universal Host (see 4.1.1.135)
UI	Unit Interval (see 4.1.1.133)
USM	Universal Storage Module (see 6.7)
UUT	Unit Under Test (see 4.1.1.134)
VNA	Vector Network Analyzer (see 4.1.1.137)

4.1.3 Units

Units used in this standard:

Units	Meaning
%	percent (i.e., 10^{-2})
% RH	percent relative humidity (i.e., absolute humidity relative to maximum humidity expressed as a percent)
A	Ampere (unit of electric current)
AWG	American Wire Gauge
B	byte (see 4.1.1.8)
bit	bit
dB	decibels (i.e., 10^{-1} bels)
dB/dec	decibels per decade (i.e., 10^{-1} bels per decade)
dBm	decibel milliwatts (i.e., power ratio in decibels of the measured power referenced to one milliwatt)
°C	degree Celsius (unit of temperature)
cycles per hour	cycles per hour (unit of repetition rate)
g's	Gravitational acceleration (i.e., unit of acceleration)
Gbit/s	Giga bits per second (i.e., 10^9 bits per second)
GHz	Gigahertz (i.e., 10^9 cycles per second)
GT/s	Giga Transfers per second (i.e., 10^9 transfers per second)
hour	hour (i.e., 60 minute)
Hz	Hertz (i.e., cycles per second)
in	inch (i.e., unit of length, 2.54 centimeters exactly)
kHz	kilohertz (i.e., 10^3 cycles per second)
kohm	kiloohm (i.e., 10^3 ohm)
m	meter (unit of distance)
mA	milliampere (i.e., 10^{-3} ampere)
MB	Megabyte (i.e., 10^6 byte)
MHz	Megahertz (i.e., 10^6 hertz)
mil	thousandth of an inch (i.e., 10^{-3} inch)
millimeter per minute	millimeter per minute (unit of speed)

Units	Meaning
minute	minute (i.e., 60 second)
mm	millimeter (i.e., 10^{-3} meter)
mmHg	millimeter of mercury (unit of pressure)
mohm	milliohm (i.e., 10^{-3} ohm)
Mohm	Megaohm (i.e., 10^{6} ohm)
ms	millisecond (i.e., 10^{-3} second)
mV	millivolt (i.e., 10^{-3} volt)
mVpp	millivolt peak to peak (i.e., 10^{-3} volt peak to peak)
mVppd	millivolt peak to peak differential (i.e., 10^{-3} volt peak to peak differential)
N	Newton (unit of force)
nF	nanofarad (i.e., 10^{-9} farad)
ns	nanosecond (i.e., 10^{-9} second)
ohm	ohm (unit of resistance)
pF	picofarad (i.e., 10^{-12} Farad)
ps	picosecond (i.e., 10^{-12} second)
s	second (unit of time)
sigma	standard deviation
uA	microampere (i.e., 10^{-6} ampere)
uF	microfarad (i.e., 10^{-6} Farad)
UI	Unit Interval (see 4.1.1.133)
us	microsecond (i.e., 10^{-6} second)
V	volt (unit of electromotive force)
VDC	volt direct current (unit of electromotive force)

4.1.4 Mathematical operators

Mathematical operators used in this standard:

Mathematical Operators	Meaning
$\|v\|$	the absolute value (i.e., magnitude) of v
Δ	delta
/	divide by
=	equal
XOR	exclusive logical OR
>	greater than
≥	greater than or equal to
∞	infinity
(	left parenthesis
<	less than
≤	less than or equal to
log	Log base 10

Mathematical Operators	Meaning
&&	logical AND
!	logical NOT (i.e., negation)
||	logical OR
×	multiplication
!=	not equal
+	plus
±	plus or minus
//	remarks
)	right parenthesis
-	subtract

4.2 Conventions

4.2.1 Capitalization

Lowercase is used for words having the normal English meaning. Certain words and terms used in this specification have a specific meaning beyond the normal English meaning. These words and terms are defined either in 4.1 or in the text where they first appear.

The names of abbreviations, commands, and acronyms used as signal names are in all uppercase (e.g., IDENTIFY DEVICE). Field and bit names are identified using SMALL CAPS followed by either the word field or the word bit (e.g., NCQ FEATURE SET SUPPORTED bit). Fields containing only one bit are usually referred to as the NAME bit instead of the NAME field.

Names of device registers and logs begin with a capital letter (e.g., Command).

Primitive names are followed by a “P” subscript (e.g., R_OK$_P$).

4.2.2 Precedence

If there is a conflict between text, figures, and tables, the precedence shall be tables, figures, and then text.

4.2.3 Keywords

4.2.3.1 Keywords scope

Several keywords are used to differentiate between different levels of requirements and optionality.

4.2.3.2 expected

A keyword used to describe the behavior of the hardware or software in the design models assumed by this standard. Other hardware and software design models may also be implemented.

4.2.3.3 mandatory

A keyword indicating items required to be implemented as defined by this standard.

4.2.3.4 may

A keyword that indicates flexibility of choice with no implied preference.

4.2.3.5 may not

A keyword that indicates flexibility of choice with no implied preference.

4.2.3.6 na

A keyword that indicates that a field or value is not applicable and has no defined value and should not be checked by the recipient.

4.2.3.7 obsolete

A keyword used to describe bits, bytes, fields, and code values that no longer have consistent meaning or functionality from one implementation to another. However, some degree of functionality may be required for items designated as “obsolete” to provide for backward compatibility. An obsolete bit, byte, field, or command shall never be reclaimed for any other use in any future standard. Bits, bytes, fields, and code values that had been designated as “obsolete” in previous standards may have been reclassified as “retired” in this standard based on the definitions herein for “obsolete” and “retired”.

4.2.3.8 optional

A keyword that describes features that are not required by this standard. However, if any optional feature defined by the standard is implemented, the feature shall be implemented in the way defined by the standard.

4.2.3.9 retired

A keyword indicating that the designated bits, bytes, fields, code values, or physical resources (e.g., such as pins on a connector) that had been defined in previous standards are not defined in this standard and may be reclaimed for other uses in future standards. Retired pins on a connector should be left not connected. If retired bits, bytes, fields, code values, or physical resources (e.g., such as pins on a connector) are utilized before they are reclaimed, they shall have the meaning or functionality as described in previous standards.

4.2.3.10 reserved

A keyword indicating reserved bits, bytes, words, fields, and code values, or physical resources (i.e., pins on a connector) that are set-aside for future standardization. Their use and interpretation may be specified by future extensions to this or other standards. A reserved bit, byte, word, or field shall be cleared to zero, or in accordance with a future extension to this standard. The recipient shall not check reserved bits, bytes, words, or fields. Receipt of reserved code values in defined fields shall be treated as a command parameter error and reported by returning command aborted. Reserved pins on a connector shall be left not connected.

4.2.3.11 shall

A keyword indicating a mandatory requirement. Designers are required to implement all such mandatory requirements to ensure interoperability with other standard conformant products.

4.2.3.12 should

A keyword indicating flexibility of choice with a strongly preferred alternative. Equivalent to the phrase “it is recommended”.

4.2.4 Numbering

Numbers that are not immediately followed by a lowercase "b" or "h" are decimal values. Numbers that are immediately followed by a lowercase "b" (e.g., 01b) are binary values. Numbers that are immediately followed by a lowercase "h" (e.g., 3Ah) are hexadecimal values.

4.2.5 Dimensions

All dimensions are shown in millimeters unless otherwise noted.

4.2.6 Signal conventions

Signal names are shown in all uppercase letters.

4.2.7 State machine conventions

For each function to be completed a state machine approach is used to describe the sequence requirements. Each function is composed of several states to accomplish a set goal. Each state of the set is described by an individual state figure. shows the general layout for each of the state figure that comprise the set of states for the function.

State Designator: State name	Action list[P | W]	
1) Transition condition [a]	→	Next state 1
2) Transition condition [b]	→	Next state 2
[a] footnote [b] footnote		

Figure 1 – State machine conventions

Each state is identified by a state designator and a state name. The state designator is unique among all states in all state machines in this specification. The state designator consists of a set of letters that are capitalized followed by a unique number. The state name is a brief description of the primary action taken during the state, and the same state name may appear in other state machines. If the same primary function occurs in other states in the same state machine, they are designated with a unique letter at the end of the name. Additional actions may be taken while in a state and these actions are described in the state description text.

Each transition is identified by a transition label and a numbered transition condition. The number does not indicate order, but is used as a way to reference each transition condition in the text. The transition label consists of the state designator of the state that the transition is being made followed by the state designator of the state that the transition is being made. The transition condition is a brief description of the event or condition that causes the transition to occur and may include a transition action that is taken when the transition occurs. This action is described fully in the transition description text.

Upon entry to a state, all actions to be performed in that state are performed. If a state is re-entered from itself, all actions to be performed in the state are performed again.

It is assumed that all actions are performed within a state and that transitions from state to state are instantaneous.

4.2.8 Byte, Word, Dword, and Qword Relationships

The most significant bit in a byte (i.e., bit 7) is shown on the left (see Figure 2 part 1).

A Word may be represented as an ordered set of two (2) bytes. The least significant byte (lower byte) is byte 0 and the most significant byte (upper byte) is byte 1. The most significant byte is shown on the left (see Figure 2 part 1).

A double word (Dword) may be represented as an ordered set of two (2) Words. The least significant Word (lower Word) is Word 0 and the most significant Word (upper Word) is Word 1. The most significant Word is shown on the left (see Figure 2 part 1).

A Dword may be represented as an ordered set of four (4) bytes. The least significant byte is byte 0 and the most significant byte is byte 3. The most significant byte is shown on the left (see Figure 2 part 1).

A quad word (Qword) may be represented as an ordered set of two (2) Dwords. The least significant Dword (lower Dword) is Dword 0 and the most significant Dword (upper Dword) is Dword 1. The most significant Dword is shown on the left (see Figure 2 part 2).

A Qword may be represented as an ordered set of four (4) Words. The least significant Word is Word 0 and the most significant Word is Word 3. The most significant Word is shown on the left (see Figure 2 part 2).

A Qword my be represented as an ordered set of eight (8) bytes. The least significant byte is byte 0 and the most significant byte is byte 7. The most significant byte is shown on the left (see Figure 2 part 2).

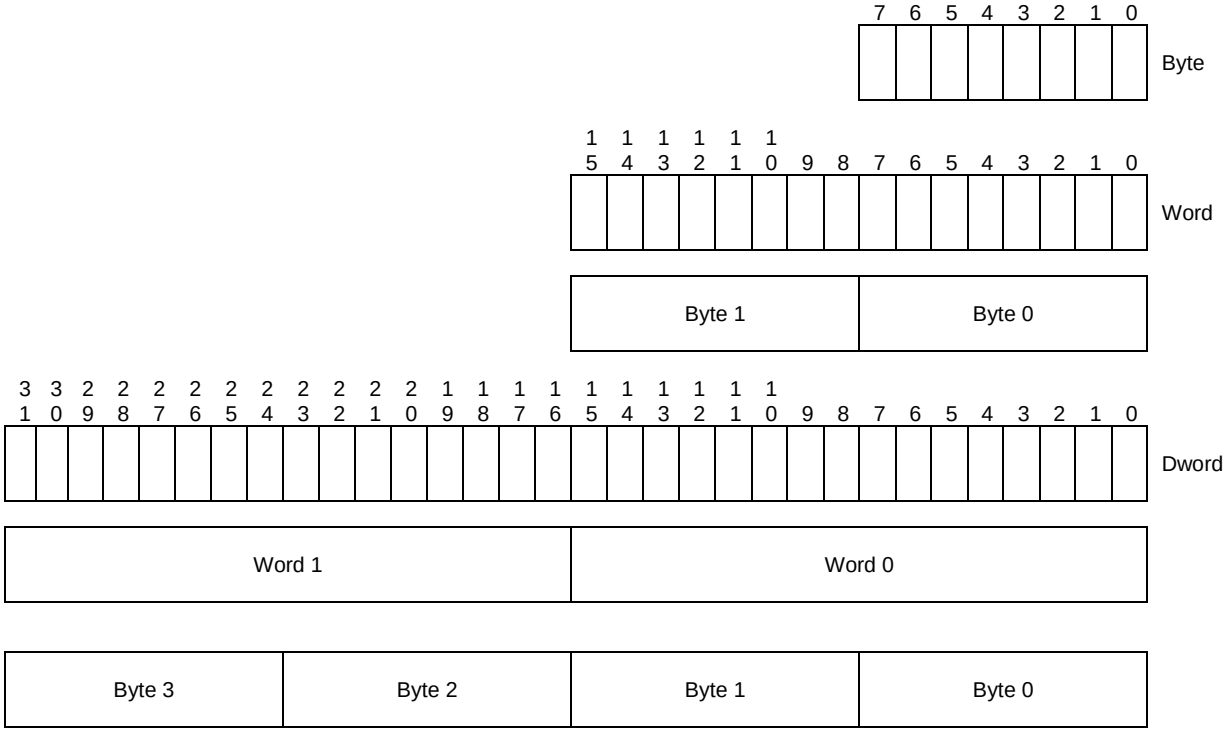


Figure 2 – Byte, Word, Dword, and Qword relationships (part 1 of 2)

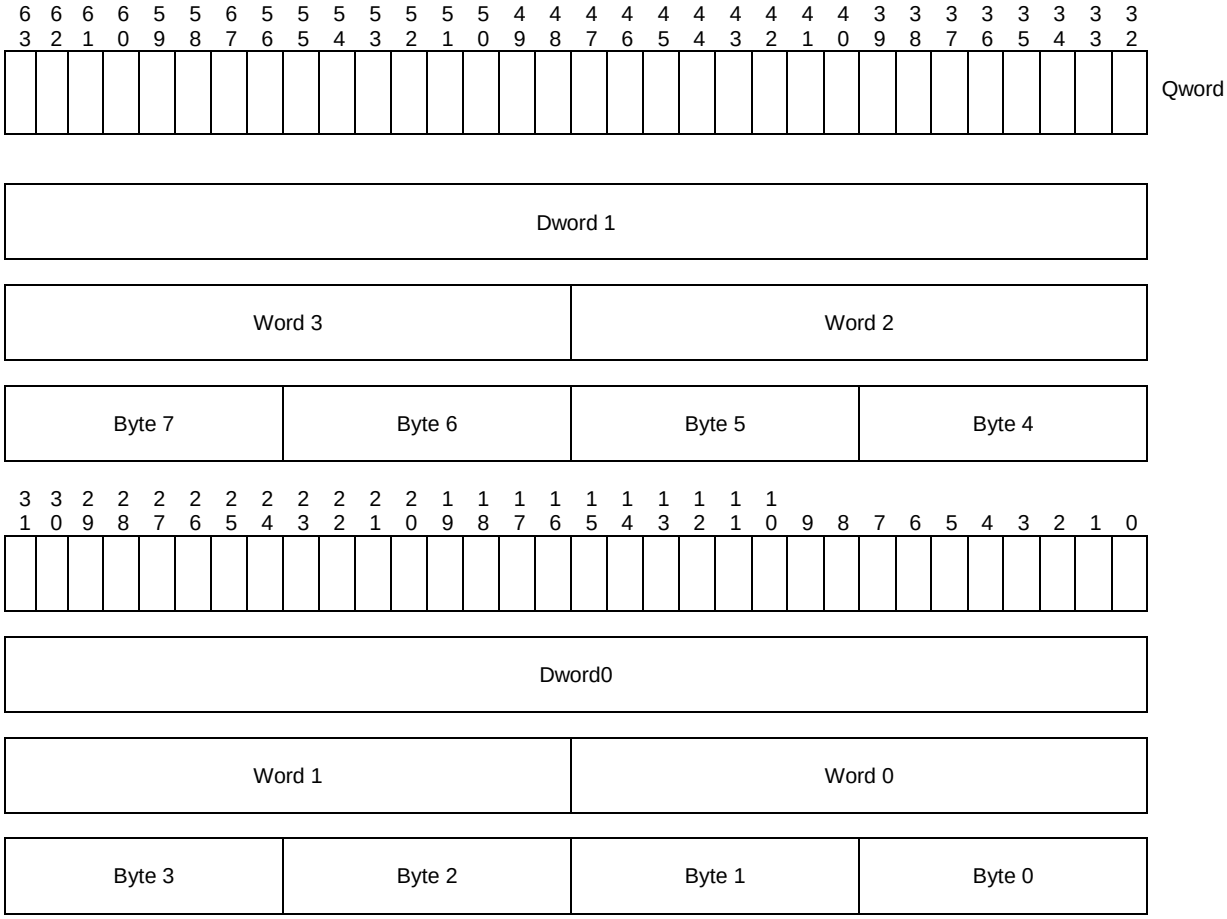


Figure 2 – Byte, Word, Dword, and Qword relationships (part 2 of 2)

5 General overview

5.1 Connectivity

Serial ATA is a high-speed serial link for mass storage devices. The serial link employed is a high-speed Phy layer that utilizes Gigabit technology and 8b/10b encoding.

Figure 3 shows an example of how the same two devices are connected using a Serial ATA Host Bus Adapter (HBA). In this figure the dark grey portion is functionally identical to the dark grey portion of **Error! Reference source not found.**. Host software that is only parallel ATA a ware accesses the Serial ATA subsystem in exactly the same manner and functions correctly. In this case, however, the software views the two devices as if they were on two separate ports. The right hand portion of the HBA is of a new design that converts the normal operations of the software into a serial data/control stream. The Serial ATA structure connects each of the two devices with their own respective cables in a point-to-point fashion.

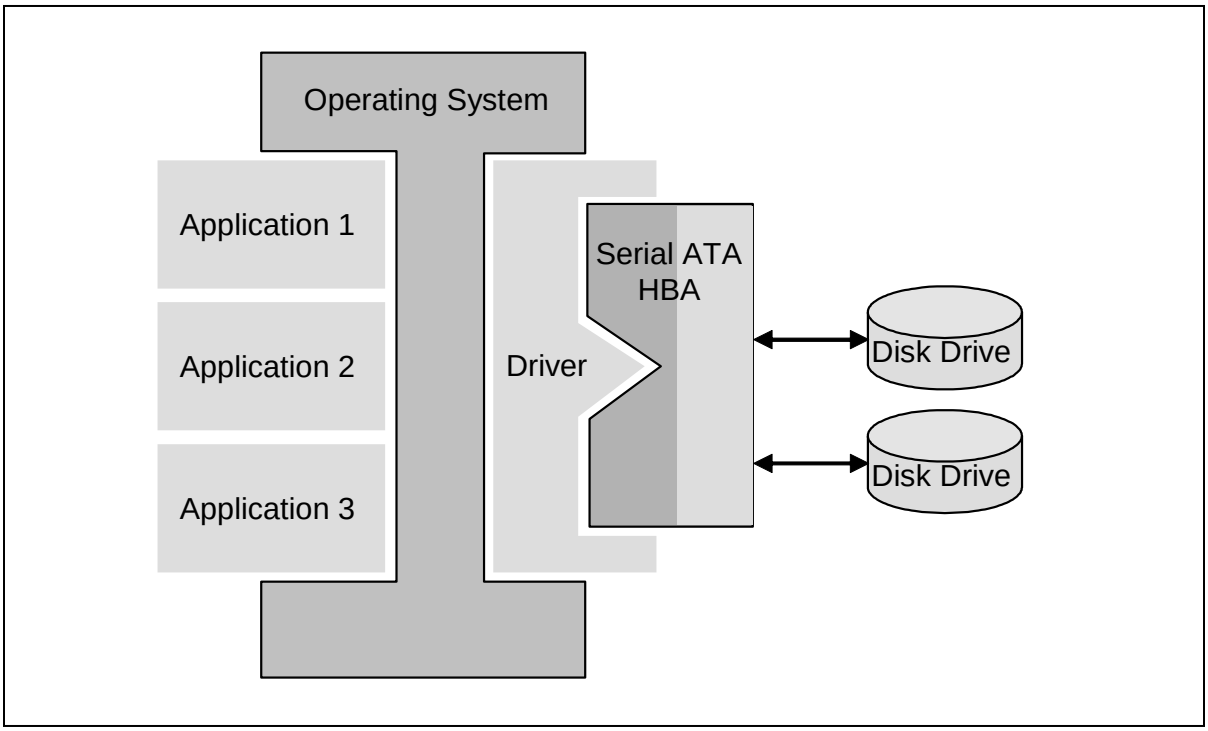


Figure 3 – Serial ATA connectivity

5.2 Architecture

There are four layers (see Figure 4) in the Serial ATA architecture:

a) Command and Application;
b) Transport;
c) Link; and
d) Phy.

The Application layer is responsible for overall ATA command processing, including controlling Command Block Register accesses. The Transport layer is responsible for placing control information and data to be transferred between the host and device in a packet/frame, known as a Frame Information Structure (FIS). The Link layer is responsible for taking data from the constructed frames, encoding or decoding each byte using 8b/10b, and inserting control characters such that the 10 bit stream of data may be decoded correctly. The Phy layer is responsible for transmitting and receiving the encoded information as a serial data stream on the wire.

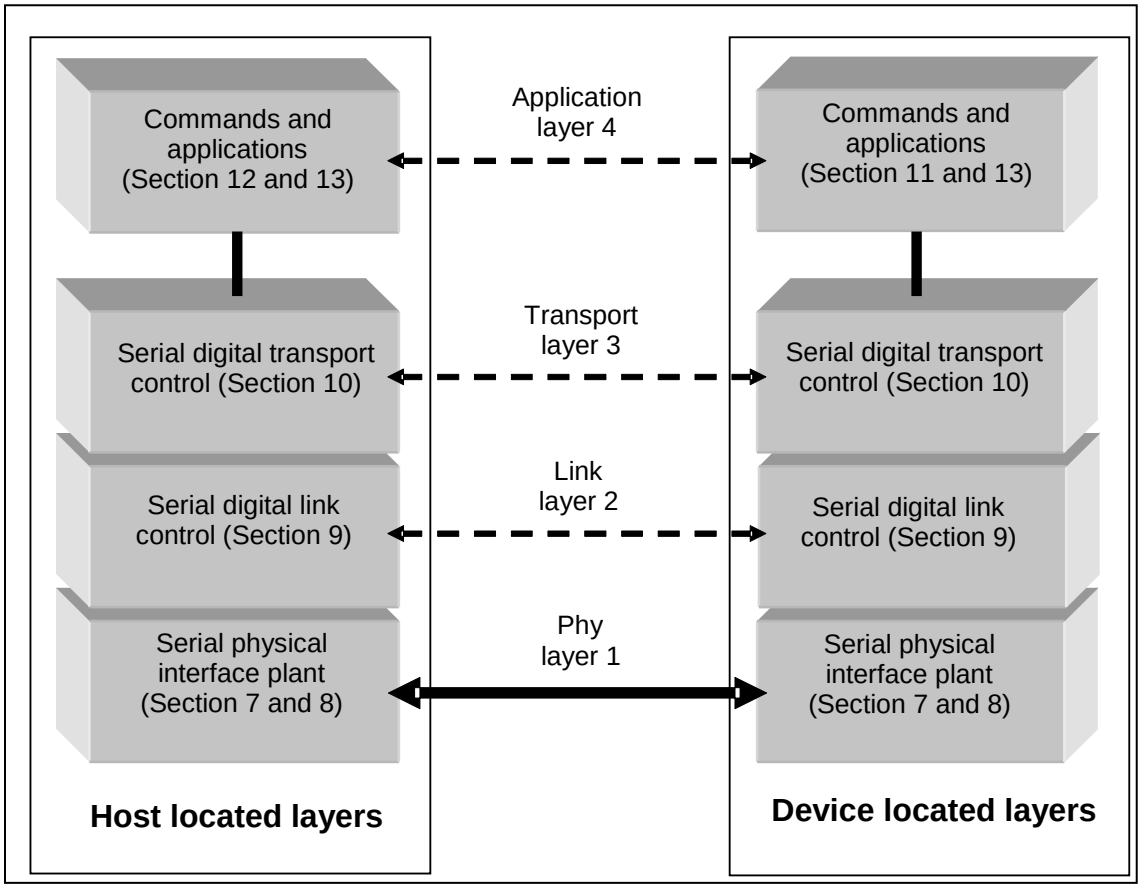


Figure 4 – Communication layers

The host may interact with the Application layer through a register interface (i.e., equivalent to that presented by a traditional parallel ATA host adapter). If using parallel ATA emulation, the host software follows existing ATA/ATAPI-6 standards and conventions if accessing the register interface and follows standard command protocol conventions.

5.3 Usage models

5.3.1 Usage models scope

This section describes some of the potential applications of Serial ATA, including usage models that take advantage of features (e.g., NCQ), enclosure management, Port Multipliers, and Port Selectors.

Table 1, Table 2, Table 3, and Table 4 outlines the different usage models described throughout the section as well as highlights the relative requirements applicable to those usage models.

Each table shows the characteristics in the columns that are necessary to support the usage models in the rows.

Feature specific (FS) is intended to indicate that Gen1 is required but higher data rates are optional.

NOTE 2 - The exact position of compliance points associated with the transmitter and the receiver are defined in 7.6.2 and 7.6.2.2.

Table 1 – Usage model descriptions (part 1 of 2)

Characteristic	Usage model section number	Cable or backplane type	Cable length	Cable Electrical	Attenuation at 4.5 GHz	Host side connector	Device side connector	Hot plug support
Internal 1 m Cabled Host to Device	5.3.2	Int SL	≤ 1 m	Table 36	-6 dB	6.2.5	6.2.3.1	NS
Short Backplane to Device	5.3.3	BP	-	P	P	P	6.2.3.1	R
Internal 4-lane Cabled Disk Arrays	5.3.5	Int ML	≤ 1 m	Table 36	-6 dB	6.2.11 or 6.2.12	6.2.3.1	NS
System to System Inter connects – Data Center Applications xSATA	5.3.6	Ext ML	≤ 2 m	Table 38	-8 dB	6.11.4 (key 7)	6.11.4 (key 7)	R
System to System Inter connects – External Desktop Applications eSATA	5.3.7	Ext SL	≤ 2 m	Table 37	-8 dB	6.11	6.11	R

Key:
R = Required configuration requires appropriate capabilities
NS = Not supported configuration is not supported by definition in specification
P = Proprietary implementation is vendor specific and not defined in specification
SL = single lane
ML = multi-lane
Int = Internal
Ext = External
BP = Backplane

NOTE - Many of the references in the table are section numbers or notations of clarification that do not require Key values.

Table 1 – Usage model descriptions (part 2 of 2)

Characteristic	Usage model section number	Cable or backplane type	Cable length	Cable Electrical	Attenuation at 4.5 GHz	Host side connector	Device side connector	Hot plug support
Proprietary Serial ATA Disk Arrays	5.3.8	BP and cable	-	Table 36	-6 dB	6.2.5 or P	6.2.3.1	R
Serial ATA and SAS	5.3.9	BP	-	P	P	SAS	6.2.3.1	R
LIF-SATA	5.3.11.6	P	P	Table 36	-6 dB	6.5.4	6.5.3	NS
mSATA	5.3.11.5	BP	-	P	P	6.6.3	6.6.4	NS
SATA USM	0	P	P	P	P	6.7.2	6.2.3.1	R
SATA MicroSSD	5.3.11.7	P	P	P	P	NS	NS	NS
Embedded M.2	5.3.11.8	BP	na	P	P	6.9.7	6.9.3	NS

Key:
R = Required configuration requires appropriate capabilities
NS = Not supported configuration is not supported by definition in specification
P = Proprietary implementation is vendor specific and not defined in specification
SL = single lane
ML = multi-lane
Int = Internal
Ext = External
BP = Backplane

NOTE - Many of the references in the table are section numbers or notations of clarification that do not require Key values.

Table 2 – Usage model electrical requirements (part 1 of 2)

Characteristic	**Gen1i** **1.5** Gbit/s	**Gen1m** **1.5** Gbit/s	**Gen1u** **1.5** Gbit/s	**Gen2i** **3.0** Gbit/s	**Gen2m** **3.0** Gbit/s	**Gen2u** **3.0** Gbit/s	**Gen3i** **6.0** Gbit/s	**Gen3u** **6.0** Gbit/s
Internal 1 m cabled host to device	R	NS	NS	FS [a]	NS	NS	FS [a]	NS
Short backplane to device	D (host to provide received signal)	H	NS	D (host to provide received signal)	H	NS	NS	NS
Internal 4-lane cabled disk arrays	R	NS	NS	FS [a]	NS	NS	FS [a]	NS
System to system inter connects – data center applications xSATA	NS	R (key 7)	NS	NS	R (key 7)	NS	NS	NS
System to system inter connects – external desktop applications eSATA	NS	R	NS	NS	FS [a]	NS	NS	NS
Proprietary Serial ATA disk arrays	R	NS	NS	FS [a]	NS	NS	FS [a]	NS
Serial ATA and SAS	D	NS	NS	D	NS	NS	D	NS
LIF-SATA	R	NS	NS	FS [a]	NS	NS	FS [a]	NS
mSATA	R	NS	NS	FS [a]	NS	NS	NS	NS
SATA USM	D	NS	H	D	NS	H	D	H

Key:
R = Required configuration requires appropriate capabilities
FS = Feature specific configuration is supported by specification but may be tied to an optional capability
NS = Not supported configuration is not supported by definition in specification
H = Host
D = Device

NOTE - Many of the references in the table are section numbers or notations of clarification that do not require Key values.

[a] Feature specific is intended to indicate that Gen1 is required but higher data rates are optional.

Table 2 – Usage model electrical requirements (part 2 of 2)

Characteristic	Gen1i **1.5** Gbit/s	Gen1m **1.5** Gbit/s	Gen1u **1.5** Gbit/s	Gen2i **3.0** Gbit/s	Gen2m **3.0** Gbit/s	Gen2u **3.0** Gbit/s	Gen3i **6.0** Gbit/s	Gen3u **6.0** Gbit/s
SATA MicroSSD	D	NS	H	D	NS	H	D	H
Embedded M.2	D	NS	H	D	NS	H	D	H

Key:
R = Required configuration requires appropriate capabilities
FS = Feature specific configuration is supported by specification but may be tied to an optional capability
NS = Not supported configuration is not supported by definition in specification
H = Host
D = Device

NOTE - Many of the references in the table are section numbers or notations of clarification that do not require Key values.

[a] Feature specific is intended to indicate that Gen1 is required but higher data rates are optional.

Table 3 – SATA Express usage models (obsolete)

Characteristic	Usage model section number	Host side connector	Device side connector	Cable length	Cable and connector electrical	PCIe lanes	SATA Lanes
SATA Express Internal 1 m Cabled Host to Device	6.10	6.10.10	6.10.7	≤ 1 m	6.10.15	x2	SL (single lane) muxed with PCIe
SATA Express Direct Connection to Device	6.10	6.10.9	6.10.7	BP	6.10.15	x2	SL (single lane) muxed with PCIe
SATA Express Short Backplane to Device	6.10	6.10.9	6.10.7	BP	6.10.15	x2	SL (single lane) muxed with PCIe

Key:
SL = single lane
ML = multi-lane
BP = Backplane
x2 = PCIe x2 (see 4.1.1.102)

NOTE - Many of the references in the table are section numbers or notations of clarification that do not require Key values.

[a] The SATA Express cabled topology requires the use of the PCIe Phy that supports SRIS since no RefClk is sent over the cable. PCIe Phys without SRIS capability do not support the cabled application.
[b] Optional RefClk pins are provided in the SATA Express host receptacle connector, that allows use of the PCIe Phy that requires common RefClk.
[c] System OEM is responsible for the management of PCIe devices with or without common RefClk.

Table 4 – SATA Express usage model electrical requirements (obsolete)

Characteristic	PCIe Gen2 [a b c]	PCIe Gen3 [a b c]	SATA Gen1i and SATA Gen2i	SATA Gen3i
SATA Express Interenal 1 m Cabled Host to Device	R	FS [1]	R	FS [1]
SATA Express Direct Connection to Device	R	FS [1]	R	FS [1]
SATA Express Short Backplane to Device	R	FS [1]	R	FS [1]

Key:
R = Required configuration requires appropriate capabilities
FS = Feature specific configuration is supported by specification but may be tied to an optional capability

NOTE 1 - Feature specific is intended to indicate that Gen1 is required but higher data rates are optional.

[a] The SATA Express cabled topology requires the use of the PCIe Phy that supports SRIS since no RefClk is sent over the cable. PCIe Phys without SRIS capability do not support the cabled application.

[b] Optional RefClk pins are provided in the SATA Express host receptacle connector, that allows use of the PCIe Phy that requires common RefClk.

[c] System OEM is responsible for the management of PCIe devices with or without common RefClk.

5.3.2 Internal 1 m cabled host to device

In this application (see Figure 5), Gen1i, Gen2i, or Gen3i electrical specifications compliant points are located at the Serial ATA mated connectors on both the host controller and device. The cable requirement given in clause 0, operates at 1.5 Gbit/s, 3.0 Gbit/s, and 6.0 Gbit/s. The application may comply with the electrical hot plug specification according to 7.4.5.4.8.

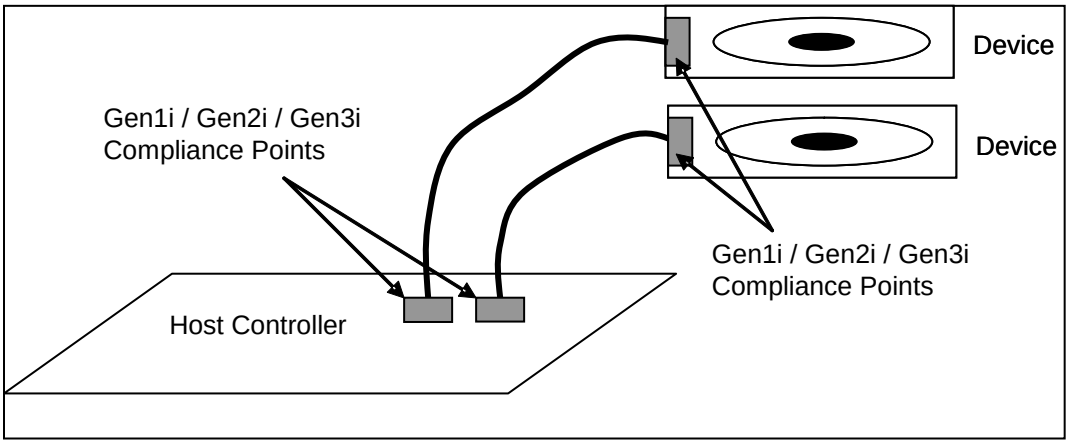


Figure 5 – Internal 1 m cabled host to device application

5.3.3 Short backplane to device

In this application (see Figure 6), Gen1i/2i disk devices within a small disk array are installed into drive canisters that are plugged into a “short” backplane. The host controller connecting to the Backplane shall contain a host component that exceeds Gen1i/2i transmitter and receiver Differential Swings specifications so that the signals at the host controller connector complies with or exceeds the Gen1m/2m electrical specifications. All other electrical specifications at this compliance point shall meet Gen1i/2i specifications. Compliance points are located at the Serial ATA mated connectors on both the device (Gen1i/2i) and the host controller (Gen1m/2m). The signaling at the host controller connector may exceed the Gen1m/Gen2m transmit maximum providing the Gen1i/Gen2i receiver maximum is not exceeded at the device connector. The application does not contain a cable but routes Serial ATA signals on printed circuit board at 1.5 Gbit/s and 3.0 Gbit/s where it is anticipated that the backplanes attenuate signals more than the compliant copper cable as defined in clause 0. The burden falls on the host controller to increase transmit signal swing and accommodate smaller receive swings at the host controller connector. The Gen1i/2i devices are not required to comply with the electrical hot plug according to 7.4.5.4.8. However, there are practical application benefits from complying with the hot plug electrical specifications.

NOTE 3 - At Gen2 speeds the designer faces significant challenges regarding signal integrity issues. Validation/feasibility data at Gen2 speeds has not been provided. Making this work is up to the system designer. Using a lab-load to measured the host controller amplitude or jitter at the device connector and comparing with Table 57 is not appropriate. Additional margin is required due to the non-ideal impedance match of transmitter and receiver. The receiver is tested to work at these amplitudes and jitter levels when the signal is applied from a Lab-Sourced Signal.

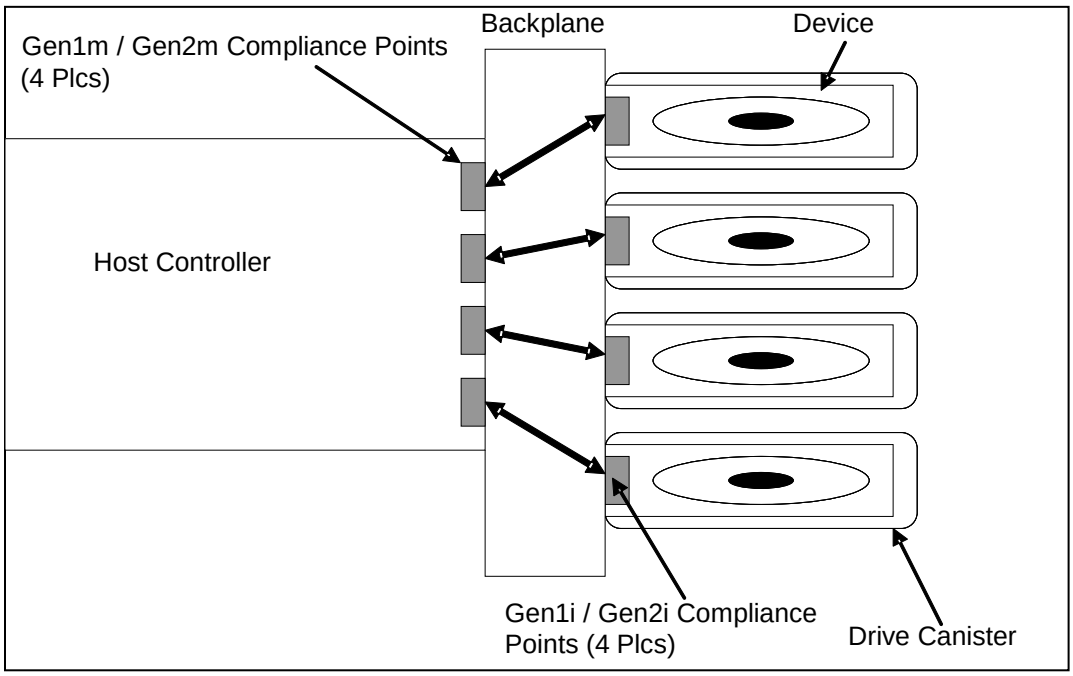


Figure 6 – Short backplane to device application

5.3.4 Long backplane to device (obsolete)

Obsolete.

5.3.5 Internal 4-lane cabled disk arrays

In this application (see Figure 7), Gen1i, Gen2i, and Gen3i Serial ATA devices are connected to the host controller via the internal 4-lane cable solution. Gen1i, Gen2i, and Gen3i specifications shall be met at each end of the 4-lane cable mated interface.

The internal 4-lane cable shall connect to the device by one of the following two approaches:

a) the host end of the internal 4-lane solution shall mate directly to the host controller board. The device end shall consist of four individual single lane Serial ATA cables for direct mate with up to four individual Serial ATA devices; or
b) the host end of the internal 4-lane solution shall mate directly to the host controller board. The device end of the internal 4-lane cable shall consist of a single internal 4-lane connector mated to a backplane that provides individual connection points to up to four Serial ATA devices. The backplane design is proprietary.

In the second solution, attenuation of the backplane reduces signal amplitude below specification limits at the compliance points. An IC, (e.g., a Port Selector), shall be placed between the device and the internal 4-lane cable mated interface. Gen1i, Gen2i, and Gen3i specifications shall be met at each end of the 4-lane cable mated interface and the device mated connector interface.

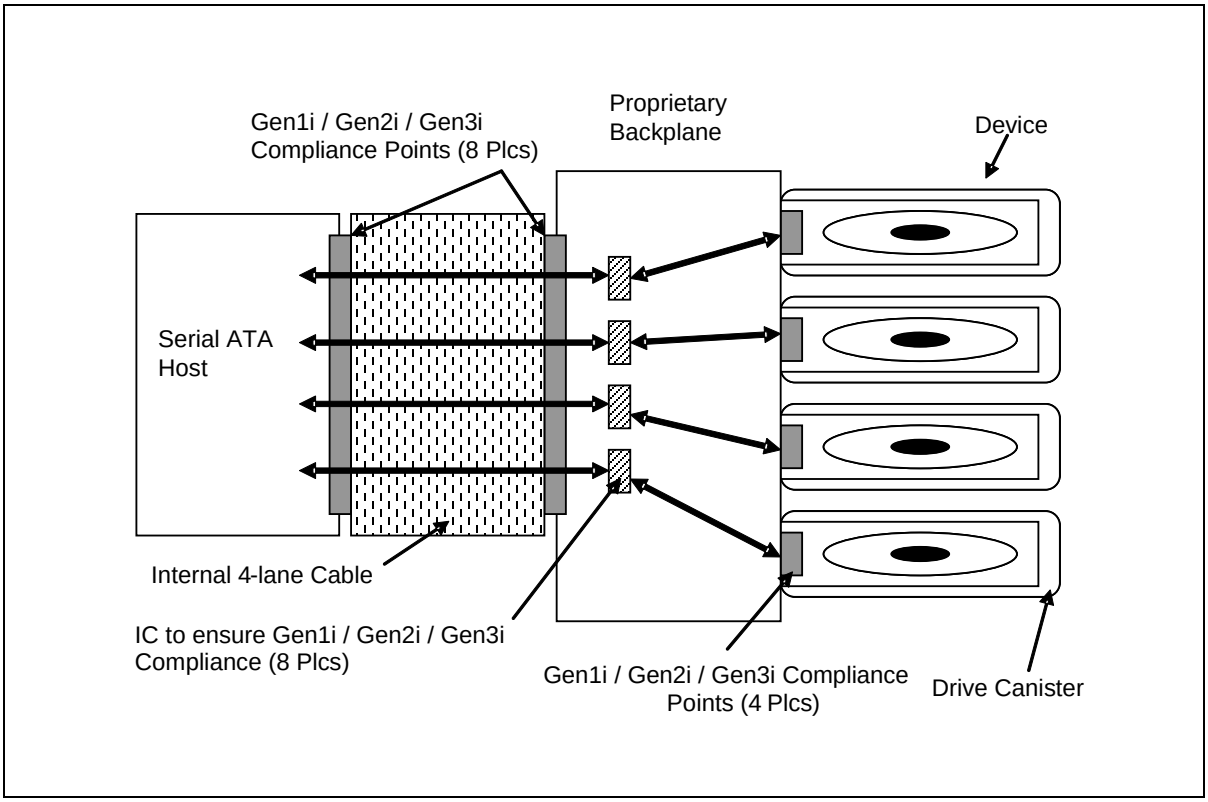


Figure 7 – Internal 4-lane cabled disk array

NOTE 4 - For Gen2i and Gen3i devices the link from the host to the drive (through the IC on the backplane) does not have an electrical specification for the “delivered signal” to the drive. Consider the compliance point for the backplane-drive connector (through the mated pair). If the signal at the compliance point, into a lab-load meets all the requirements of Table 52 and Table 54, and the backplane meets all the requirements of Table 53, then interoperability is confirmed.

However, it is anticipated that the additional trace length on the backplane between the IC and the backplane-device connector makes compliance to these specifications difficult. The burden for ensuring interoperability with all Gen1i/Gen2i/Gen3i devices falls upon the implementer of the system. Essentially the implementer is able to use simulations and empirical results to confirm that the compliance point at the backplane-device connector is equivalent to or better than a compliant host and cable combination.

5.3.6 System-to-system interconnects – data center applications (xSATA)

This application (see Figure 8) is defined as external storage applications that require more than one serial link between systems. This application uses the external Multilane cables as defined in 6.11.3 and 6.11.4, and may be referred to as xSATA.

For system-to-system interconnects that require cables of approximately two meters or less (i.e., pedestal to pedestal, blade to blade, or intrarack connections) Gen1m/Gen2m is used.

All external Serial ATA cables function at both 1.5 Gbit/s and 3.0 Gbit/s. Use of a cable that operates at 1.5 Gbit/s but not at 3.0 Gbit/s is allowed but this cable assembly shall not be interchangeable with standard Serial ATA cables.

EXAMPLE 1 - If this 1.5 Gbit/s cable uses Serial ATA specified connectors, the cable isb keyed to insure that it is not possible to plug into standard Serial ATA connections. Hot pluggability is a requirement in this application.

Compliance points are located at the mated bulkhead connectors of each system as shown in the example implementation in Figure 8. If a SATA endpoint device is included in the system, it shall not be connected directly to the external connectors. Instead, a bridge shall be used between the external connectors and the endpoint device.

EXAMPLE 2 - Some examples of bridges include repeaters/retimers, Port Multipliers, and redundant array of independent disks (RAID) controllers.

NOTE 5 - Gen3 is not defined for xSATA.

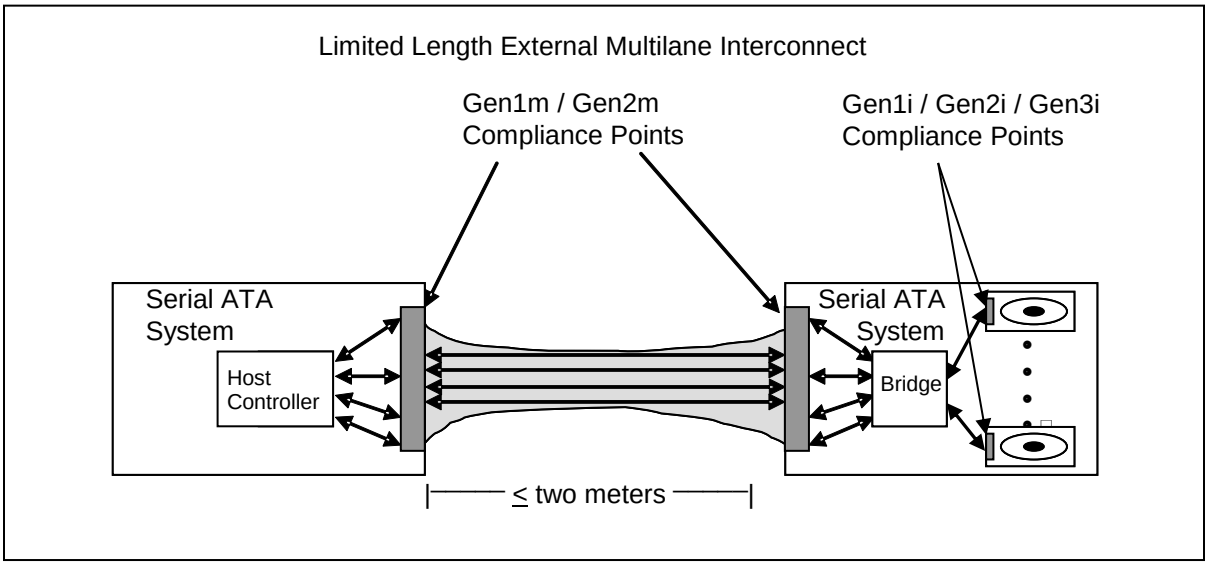


Figure 8 – System-to-system data center interconnects

5.3.7 System-to-system interconnects – external desktop applications (eSATA)

This application (see Figure 9) is aimed at external storage applications that require a single lane with approximately two meters or less of cable length. This application uses the external single lane connector system as defined in 6.11, and may be referred to as eSATA.

NOTE 6 - Gen3i is not defined for eSATA.

In the example shown below, a device enclosure contains a Gen1i / Gen2i device and an interposer card that contains an integrated circuit unless the device is specifically designed for external connection. Regardless of the implementation, the outside of the device enclosure shall meet Gen1m specifications if operating at Gen1 speeds and Gen2m specifications if operating at Gen2 speeds. A cable/connector has been defined for this application that operates at both Gen1 and Gen2. The host system has an external connector that meets Gen1m specifications if operating at Gen1 speeds and Gen2m specifications if operating at Gen2 speeds.

The entire system shall meet the following requirements:

a) the cable/connector shall operate at Gen1 and Gen2 speeds. Systems shall not deploy any cable unable to operate at Gen2 speeds if the host system and device enclosure both comply with Gen2m electrical specifications;
b) the host system and device enclosure shall comply with Gen1m specifications if operating at Gen1 speeds;
c) the host system and device enclosure may operate at Gen2 speeds. However, if they operate at Gen2 speeds the host system and device enclosure shall comply with Gen2m specifications if operating at Gen2 speeds and shall also be able to operate at Gen1 speeds using Gen1m electrical specifications; and
d) the host system and the device enclosure shall comply with the Hot Plug Specifications in this specification.

NOTE 7 – AC coupling on the transmitters and receivers of the host system and device enclosure is strongly recommended for Gen1m and required for Gen2m.

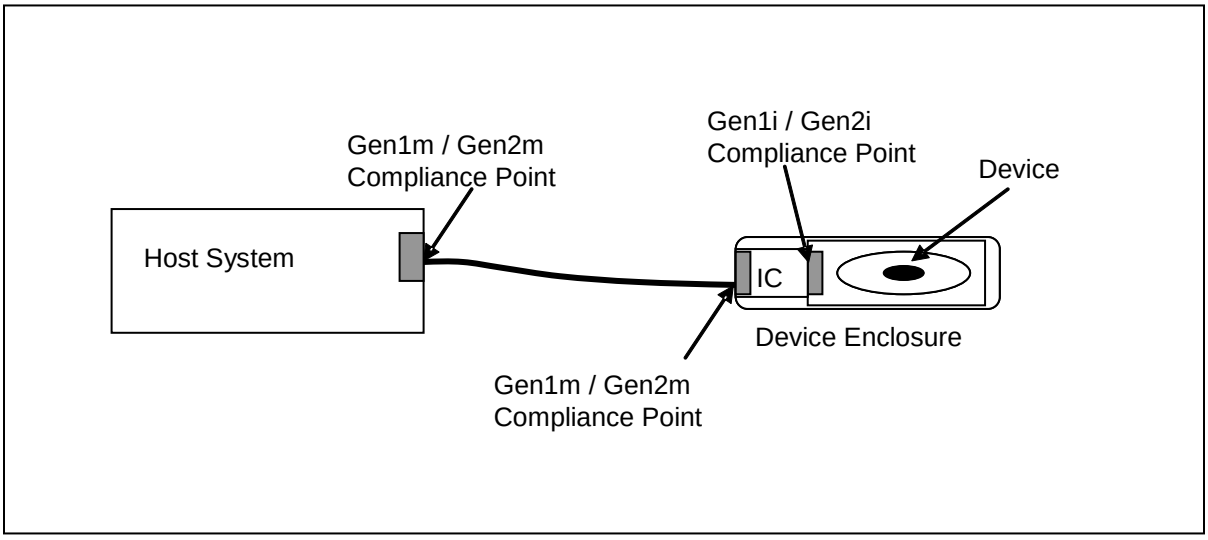


Figure 9 – External desktop application

5.3.8 Proprietary Serial ATA disk arrays

In this application (see Figure 10), Serial ATA devices are connected to a backplane and the links are routed over a combination of internal backplanes or cables as well as an external cable to a Serial ATA system. There are not semiconductors between the devices and the system so the intermediate connectors are not compliance points. Although this application is allowed, the external connectors on the disk array shall not be standard Serial ATA connectors. This is to prevent users from connecting standard external cables between the system and the disk array since they may not function reliably.

NOTE 8 - The designer faces significant challenges regarding signal integrity issues because of complexity of the interface that may require additional margin. Validation/feasibility data has not been provided. Making this work is up to the system designer. Using a lab-load to measure the host controller amplitude or jitter at the device connector, and comparing with Table 57 is not appropriate. The receiver is tested to work at these amplitudes and jitter levels when the signal is applied from a Lab-Sourced Signal.

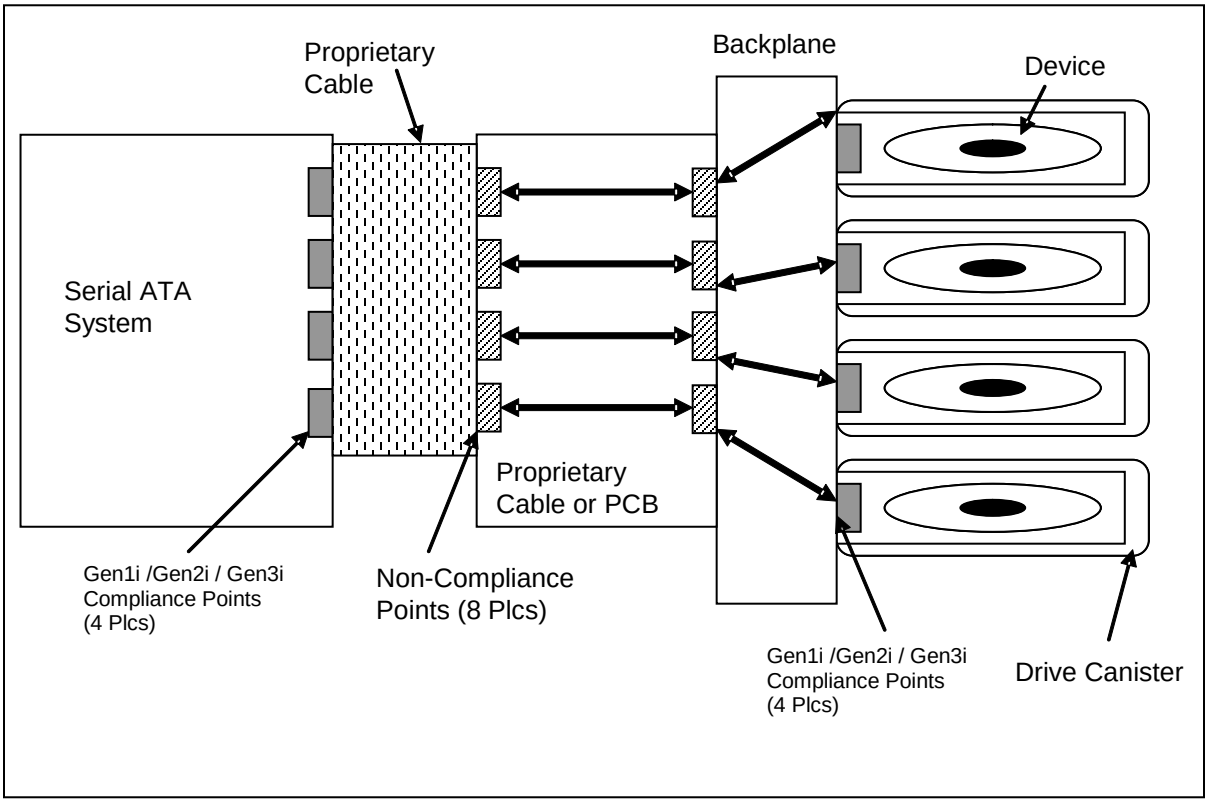


Figure 10 – SATA disk arrays

5.3.9 Serial ATA and SAS

The Serial Attached SCSI (SAS) references, within this specification, refer to a standard that specifies a Small Computer System Interface (SCSI) transport protocol over a serial link. The SAS standard borrows heavily from the SATA Phy, Link, and Transport layers. A SAS domain may support attachment to and control of unmodified SATA devices connected directly into the SAS domain using the Serial ATA Tunneled Protocol (STP).

5.3.10 Potential external SATA incompatibility issues

WARNING, the functionality of External Desktop has been defined and external Data Center cabled applications is not defined by this specification. Consequently, two systems are able to be connected that may not interoperate even though all the components comply with the electrical specifications defined in this specification. External applications are not required to support Port Multipliers or Port Selectors. As a result, a host with an External Data Center connector is able to be connected to a disk array containing a Port Multiplier and the resulting system may not operate correctly.

5.3.11 Mobile applications

5.3.11.1 Mobile applications overview

Applications and compliance points for Serial ATA devices within or connected to mobile computers are not defined in this document, except where otherwise specified. If any proprietary cables/connectors or electrical specifications are developed for this application, the system shall be designed so as to prevent connection with standard SATA components. If standard cables/connectors/electrical interfaces are used within the mobile computer, within the docking bay or to external storage components, these shall comply with the applicable requirements in this specification and interoperate properly with Serial ATA components.

5.3.11.2 Internal applications

It is expected that all internal interfaces comply with the Gen1i, Gen2i, or Gen3i specifications. Any mobile computer designer modifying electrical specifications of hosts and devices within the mobile computer is free to do so, however, all proprietary interfaces shall be designed so as to prevent connection with standard SATA components.

5.3.11.3 Docking bay applications

Proprietary docking bay interfaces shall be designed so as to prevent connection with standard SATA components.

5.3.11.4 External applications

Applications for external Serial ATA interfaces, on mobile computers, may use either the External Desktop cable/connector (Gen1m/Gen2m) or the System-to-System Data Center cable/connector (Gen1m/Gen2m). Proprietary solutions shall be designed so as to prevent connection with standard SATA components.

5.3.11.5 mSATA

Applications and compliance points for mSATA devices in the embedded applications are not defined in this specification. The mSATA interface shall comply with Gen1i and Gen2i specifications. The mSATA host and device shall comply with this standard (see Figure 11) and is equivalent to the Mobile Applications usage model.

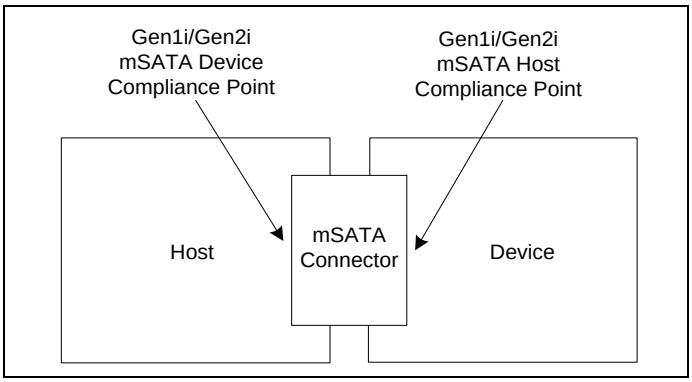


Figure 11 – mSATA application

5.3.11.6 LIF-SATA

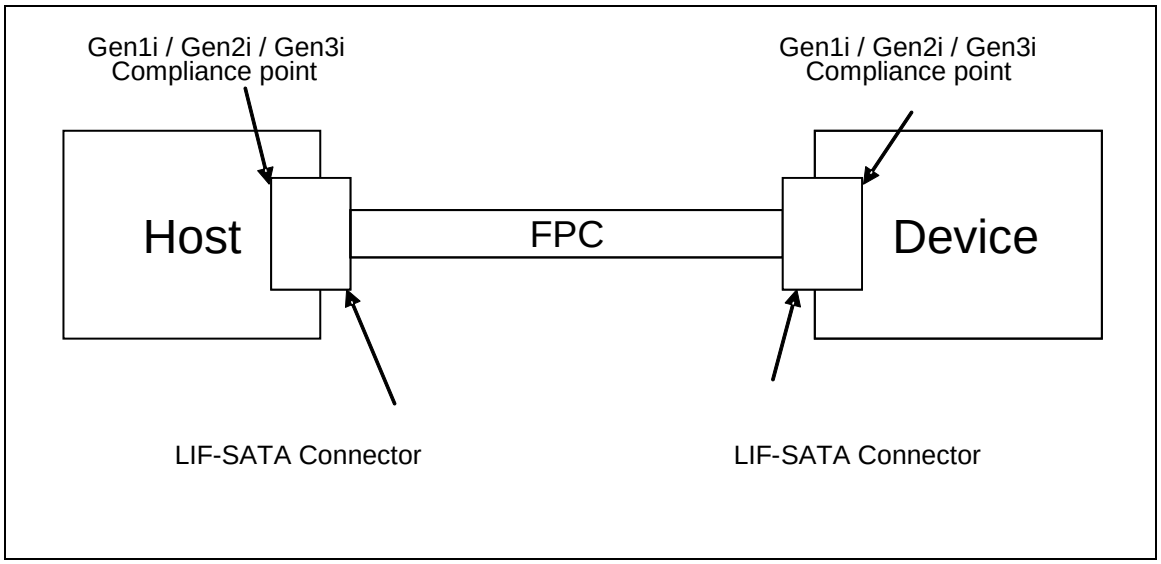


Figure 12 – Embedded LIF-SATA application

In this application (see Figure 12), the device is connected to a host controller via a flexible printed circuit cable (FPC). The FPC interfaces shall comply with Gen1i electrical requirements given in Table 36. The compliance points are shown in Figure 12. Gen2i or Gen3i compliance is feature specific, but if implemented, shall also comply with the electrical requirements given in Table 36.

5.3.11.7 SATA MicroSSD applications

In this application (see Figure 13), a Gen1i/Gen2i/Gen3i device is directly connected, via socket or solder connection, to the printed circuit board (PCB) of a Serial ATA host. Compliance points are defined at the point of connection between the SATA MicroSSD device and the SATA host. All electrical specifications at this compliance point shall meet Gen1i/Gen2i/Gen3i specifications for the device and Gen1u/Gen2u/Gen3u for the host. The signaling at the host controller may exceed the Gen1i/Gen2i/Gen3i transmit maximum providing the Gen1i/Gen2i/Gen3i receiver maximum is not exceeded at the ball grid array (BGA) interface.

The SATA MicroSSD host is an embedded host. It is the responsibility of the system manufacturer that the mounting location for the SATA MicroSSD provides AC coupling between the host and the device. AC coupling is not provided in the SATA MicroSSD.

The SATA MicroSSD device shall be tested for specification compliance as a SATA device. See 7.6.2.1.4 for SATA MicroSSD Lab-Load and 7.6.2.3 for SATA MicroSSD Lab-Sourced Signal using the compliance points shown in Figure 13.

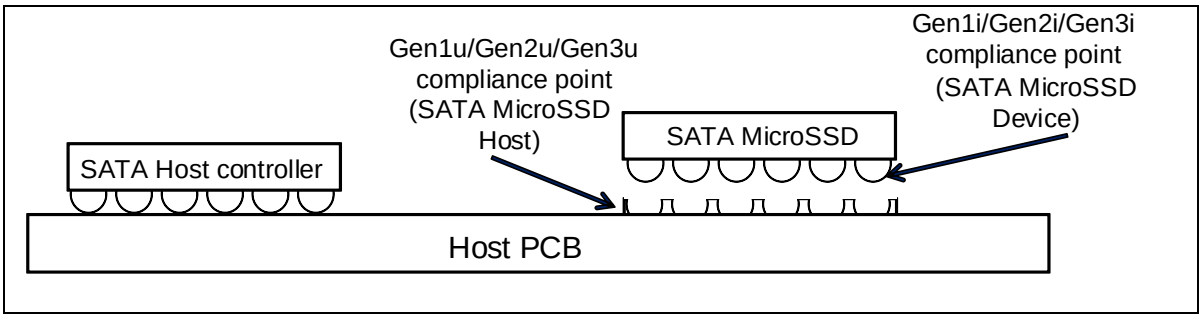


Figure 13 – Embedded SATA MicroSSD application

5.3.11.8 Embedded M.2 application

Applications and compliance points for M.2 devices in the embedded applications are not defined in this specification. The M.2 device shall comply with Gen1i, Gen2i, and Gen3i specifications. The M.2 host shall comply with Gen1u, Gen2u, and Gen3u specifications.

The M.2 host and device shall comply with this standard (see Figure 14) and is equivalent to the Mobile Applications usage model.

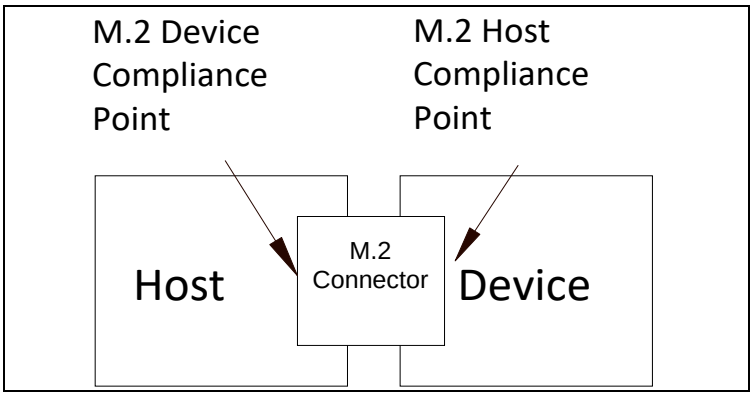


Figure 14 – Embedded M.2 application

5.3.12 SATA Universal Storage Module (SATA USM)

In this application (see Figure 15), a Gen1i/Gen2i/Gen3i device is mounted in an enclosure that provides the mechanism for the device to be inserted into or removed from a system accepting a SATA Universal Storage Module (USM) device. Compliance points are only defined at the SATA USM device mated connector pair. All electrical specifications at this compliance point shall meet Gen1i/Gen2i/Gen3i specifications for the drive and Gen1u/2u/3u specifications for the host. The signaling at the host controller may exceed the Gen1i/Gen2i/Gen3i transmit maximum providing the Gen1i/Gen2i/Gen3i receiver maximum is not exceeded at the SATA USM device connector. The host and the device shall comply with the electrical hot plug specification, details as given in 7.4.6. The SATA USM device may be tested for compliance with or without the SATA USM device enclosure. See INF-8280 for the SATA USM mechanical, power, and electromagnetic compatibility (EMC) requirements and recommendations. The SATA USM host shall be tested for compliance as a Universal Host (UHost).

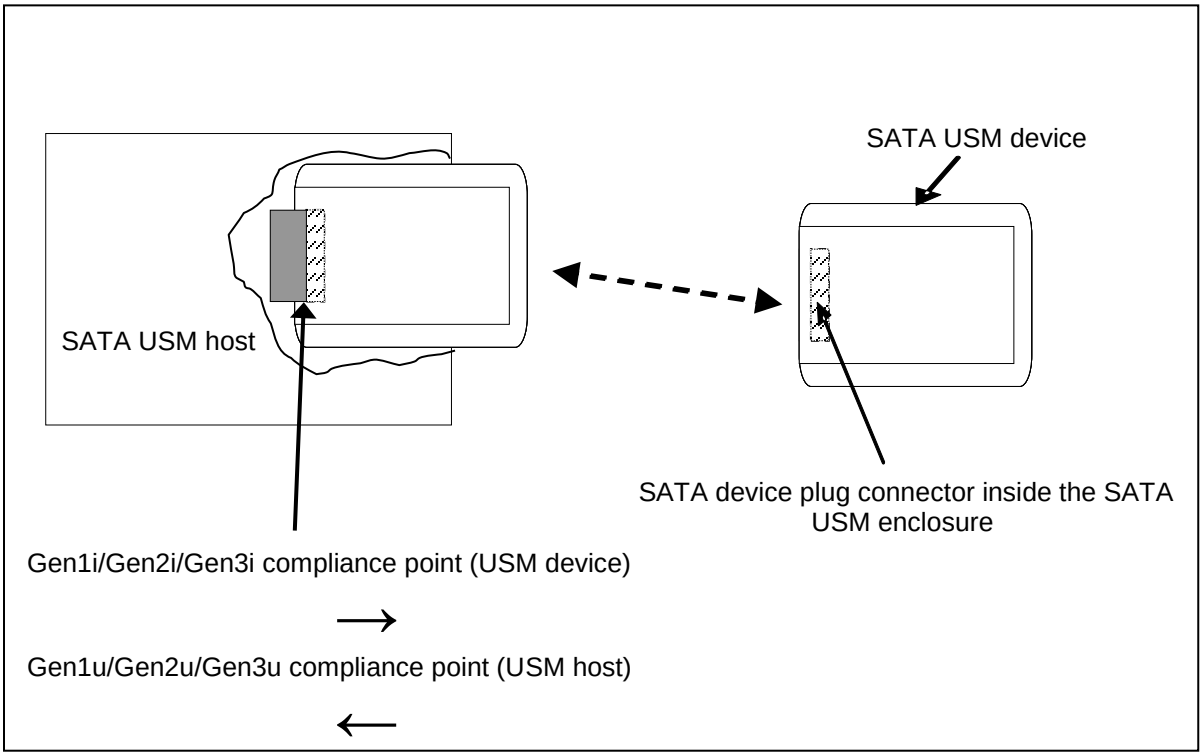


Figure 15 – SATA USM application

5.3.13 Port Multiplier example applications

One possible application of the Port Multiplier is to increase the number of Serial ATA connections in an enclosure that does not have a sufficient number of Serial ATA connections for all of the devices in the enclosure. An example is shown in Figure 16. A Multilane cable with two Serial ATA connections is delivered to the enclosure. The enclosure contains eight Serial ATA devices. To create the appropriate number of Serial ATA connections, two 1-to-4 Port Multipliers are used to create eight Serial ATA connections.

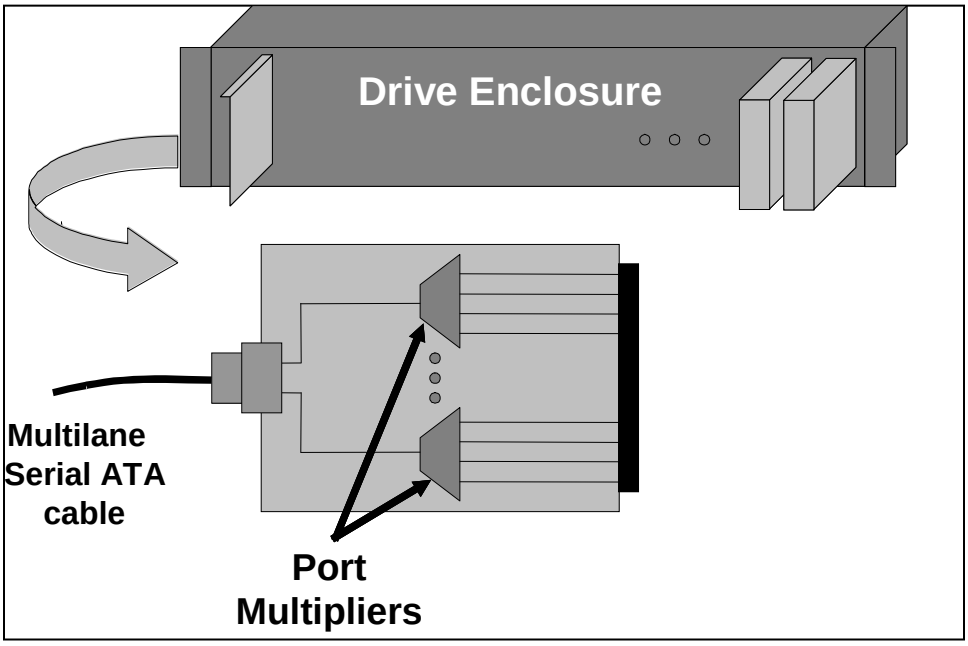


Figure 16 – Enclosure example using Port Multipliers with Serial ATA as the connection within the rack

Another example is shown in Figure 17. Fibre Channel, InfiniBand, or Gigabit Ethernet is used as the connection within the rack to the enclosure. Inside the enclosure, a host controller creates two Serial ATA connections from the connection delivered. The enclosure contains eight Serial ATA devices. To create the appropriate number of Serial ATA connections, two 1-to-4 Port Multipliers are used to create eight Serial ATA connections.

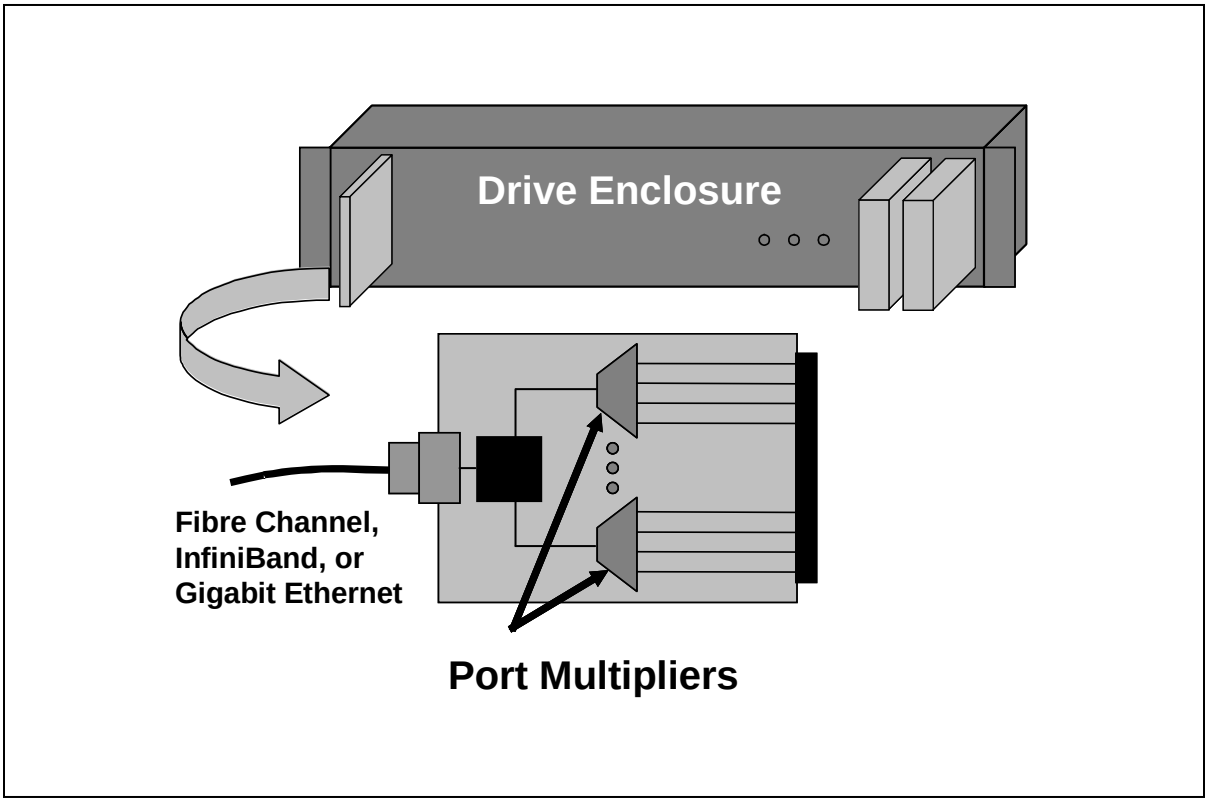


Figure 17 – Enclosure example using Port Multipliers with a different connection within the rack

The Port Multiplier allows host controllers with a modest number of connections to be used in these enclosures and then the connectivity is increased as product requirements dictate.

Another example application is using a Port Multiplier to increase the number of Serial ATA connections in a mobile docking station. The example shown in Figure 18 has a proprietary interface between the laptop and the docking station. The proprietary interface may route a Serial ATA connection from the laptop to the docking station or the docking station may create a Serial ATA connection itself. The docking station routes the Serial ATA connection to a Port Multiplier to create an appropriate number of Serial ATA connections for the number of devices to be attached.

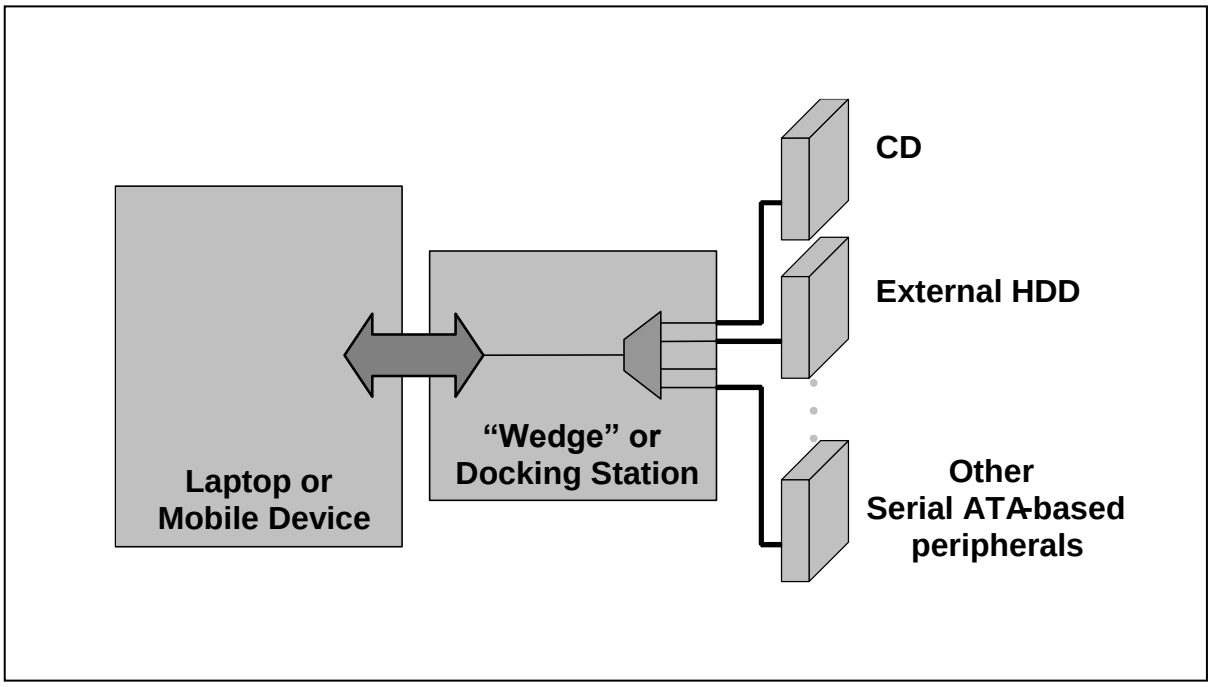


Figure 18 – Mobile docking station example using a Port Multiplier

These are a few examples of possible applications of the Port Multiplier and are not meant to be all encompassing.

6 Cables and connectors

6.1 Cables and connectors overview

This section defines the connectors and cable assemblies for Serial ATA.

It specifies:

a) the mating interfaces between the connectors;
b) the connector location on the Serial ATA device; and
c) the electrical, mechanical and reliability requirements of the connectors and cable assemblies.

The mating interfaces of Serial ATA connectors are defined in terms of their front end (i.e., separable) characteristics only. All SATA internal and external connector contact mating areas shall have a gold or gold-compatible finish. Unless otherwise specified, connector back end characteristics including finish, PCB mounting features, and cable termination features are not defined.

6.2 Internal cables and connectors

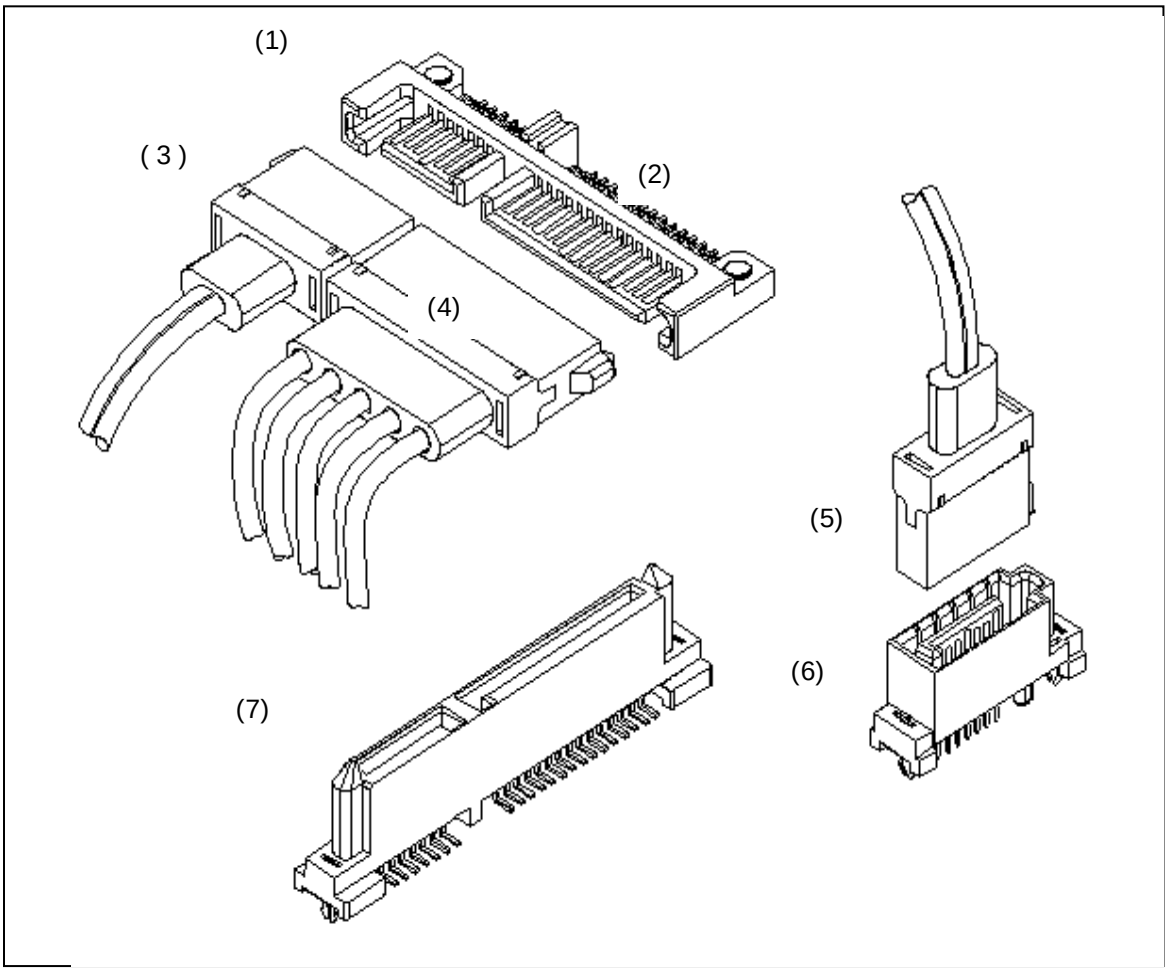
6.2.1 Internal single lane description

A Serial ATA device may be either directly connected to a host or connected to a host through a cable.

For direct connection, the device plug connector, shown as (1) and (2) in Figure 19, is inserted directly into a backplane connector, illustrated as (7) in Figure 19. The device plug connector and the backplane connector incorporate features that enable the direct connection to be hot pluggable and blind mateable.

For connection via cable, the device signal plug connector, shown as (1) in Figure 19, mates with the signal cable receptacle connector on one end of the cable, illustrated as (3) in Figure 19. The signal cable receptacle connector on the other end of the cable is inserted into a host signal plug connector, shown as (6) in Figure 19. The signal cable wire consists of two twinax sections in a common outer sheath.

Besides the signal cable, there is also a separate power cable for the cabled connection. A Serial ATA power cable includes a power cable receptacle connector, shown as (4) in Figure 19, on one end and may be directly connected to the host power supply on the other end or may include a power cable receptacle on the other end. The power cable receptacle connector on one end of the power cable mates with the device power plug connector, shown as (2) in Figure 19. The other end of the power cable is attached to the host as necessary.



Key:
1 = device signal plug segment or connector
2 = device power plug segment or connector
3 = signal cable receptacle connector, to be mated with (1)
4 = power cable receptacle connector, to be mated with (2)
5 = signal cable receptacle connector, to be mated with (6)
6 = the host signal plug connector
7 = backplane connector mating directly with device plug connector (1) and (2)

Figure 19 – Serial ATA connector examples

Figure 20 shows a direct cable / connector connection and highlights the signal path of the differential transmitter (Tx) and receiver (Rx) pairs.

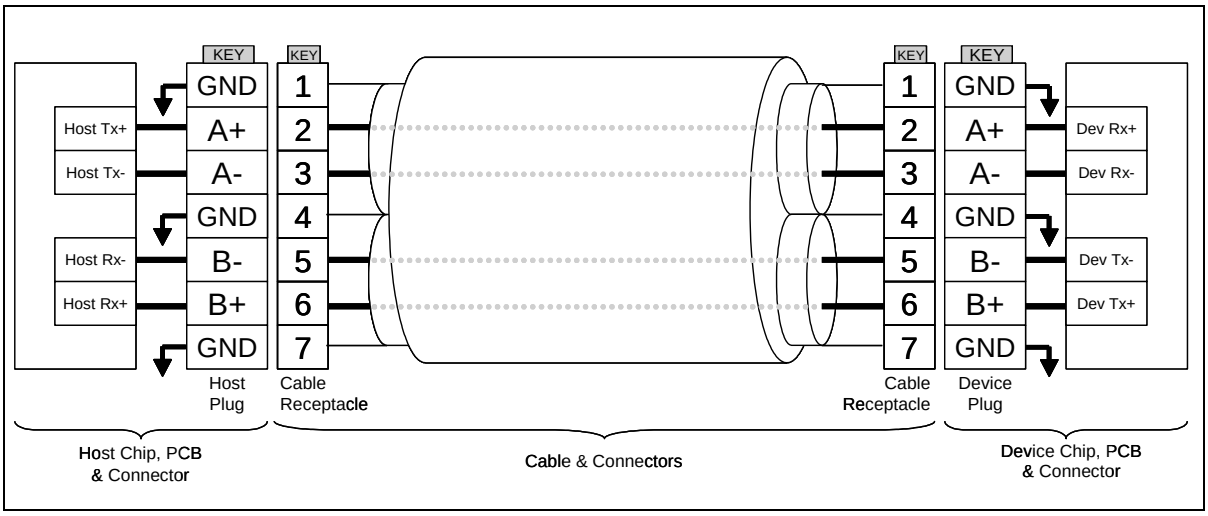


Figure 20 – SATA cable / connector connection diagram

The connector on the left represents the host with Tx/Rx differential pairs connected to a cable. The connector on the right shows the device with Tx/Rx differential pairs also connected to the cable. Notice also the ground path connecting the shielding of the cable to the Cable Receptacle.

Figure 21 shows the connection between host and device as a direct connection. It is similar to the cable/connector connection with the exception of the cable.

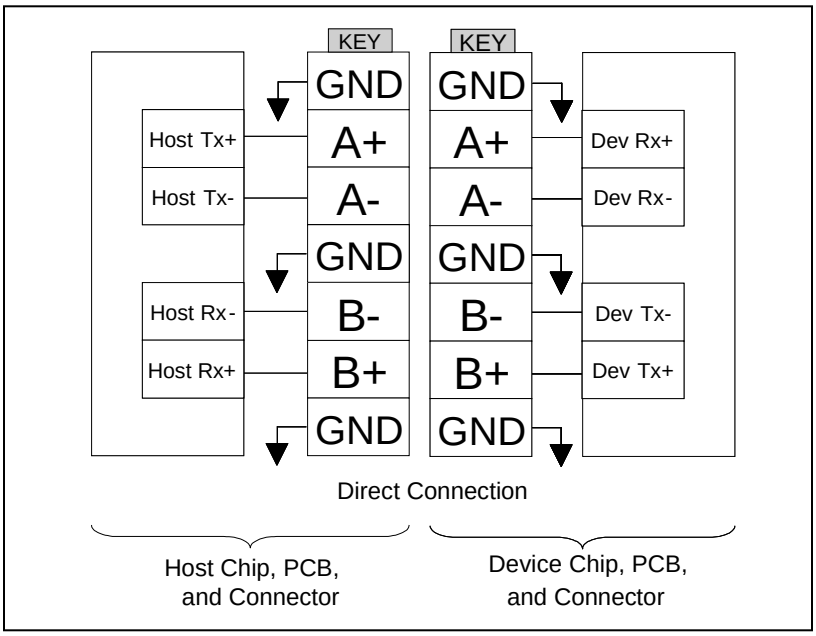


Figure 21 – SATA host / device direct connection diagram

In both cases the connection of the Tx differential signal pair on the host side to the Rx differential signal pair on the device side. A similar connection of the host Rx pair to the device Tx pair is also shown.

6.2.2 Connector locations

The device connector location is defined to facilitate blind mating.

Figure 22 and Figure 23 define the connector location on 5.25 inch devices. Optical devices shall locate the connector as indicated in the optical device connector location. Non-optical devices should locate the connector as indicated in the optical device connector location but may locate the connector as indicated in the non-optical device alternate connector location. The Serial ATA connector is nominally flush to the end of the device factor.

Figure 24 defines the connector location on side mounted 3.5 inch devices. Figure 25 defines the connector location on bottom mounted 3.5 inch devices. Refer to EIA-740 and SFF-8301 for 3.5 inch device form factor specifications. The Serial ATA connector is nominally flush to the end of the device factor.

Figure 26 defines the connector location on side mounted 2.5 inch devices. Figure 27 defines the connector location on bottom mounted 2.5 inch devices. Refer to EIA-720 and SFF-8201 for 2.5 inch device form factor specifications. The Serial ATA connector nominally protrudes 0.3 mm from the end of the device factor.

Figure 28 defines the Serial ATA connector location on side mounted 1.8 inch devices. Figure 29 defines the Serial ATA connector location on bottom mounted 1.8 inch devices. Refer to SFF-8111 for 1.8 inch device form factor specifications. The Serial ATA connector nominally protrudes 0.3 mm from the end of the device factor.

To ensure mating of devices to backplanes with proper mechanical and electrical interface and without physical conflict, Figure 41 illustrates the fully mated condition of the device to a nominally flush backplane receptacle and Figure 30 defines the keep out zones for devices of all form factors. The application shall ensure that these areas do not contain any materials or construction that prevents the fully mated condition.

The keying feature of the SATA connector is designed to prevent plugs and receptacles from being inverted. This key is the protrusion that meets the housing itself, not the ”L” shaped feature found on the contact tab. See Figure 33.

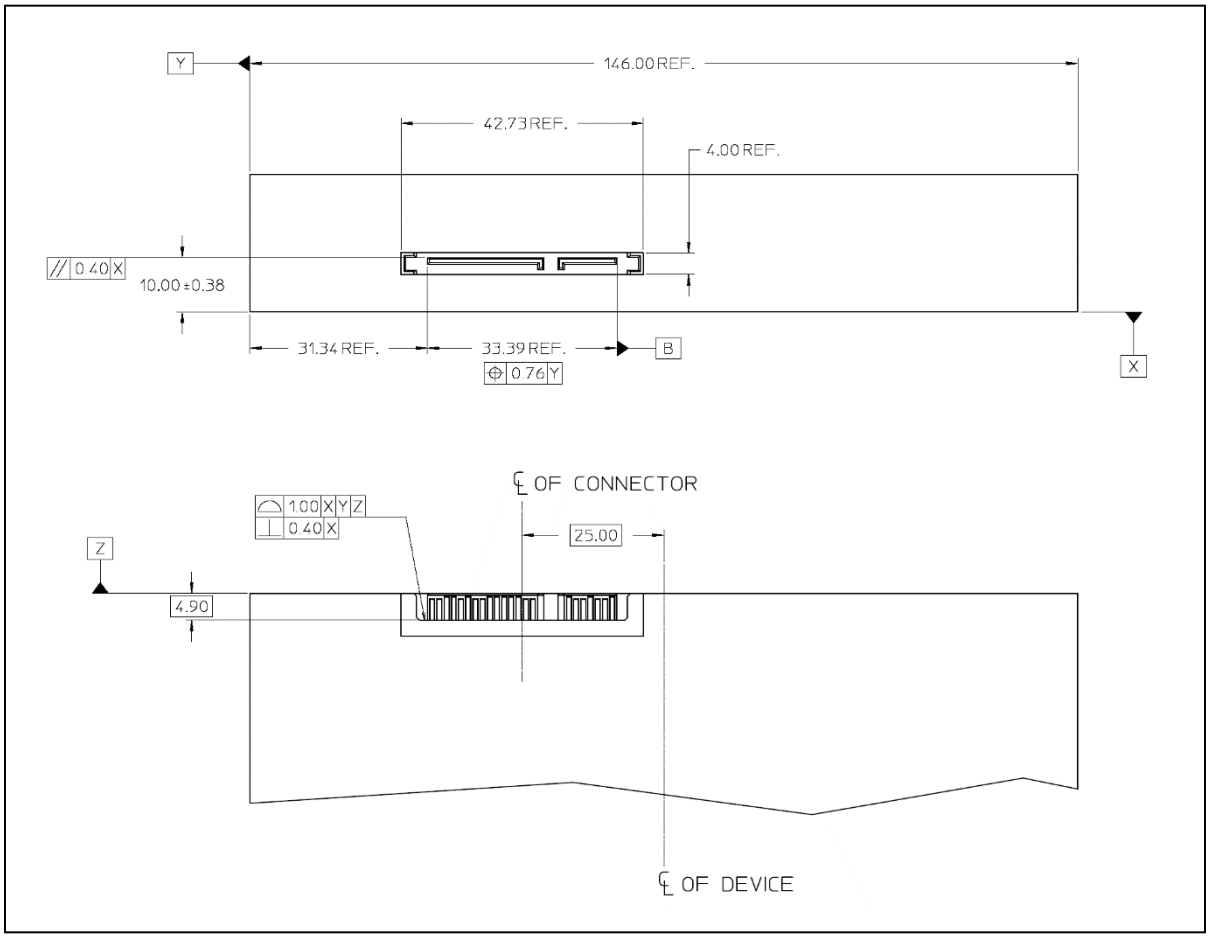


Figure 22 – Optical device plug connector location on 5.25 inch form factor

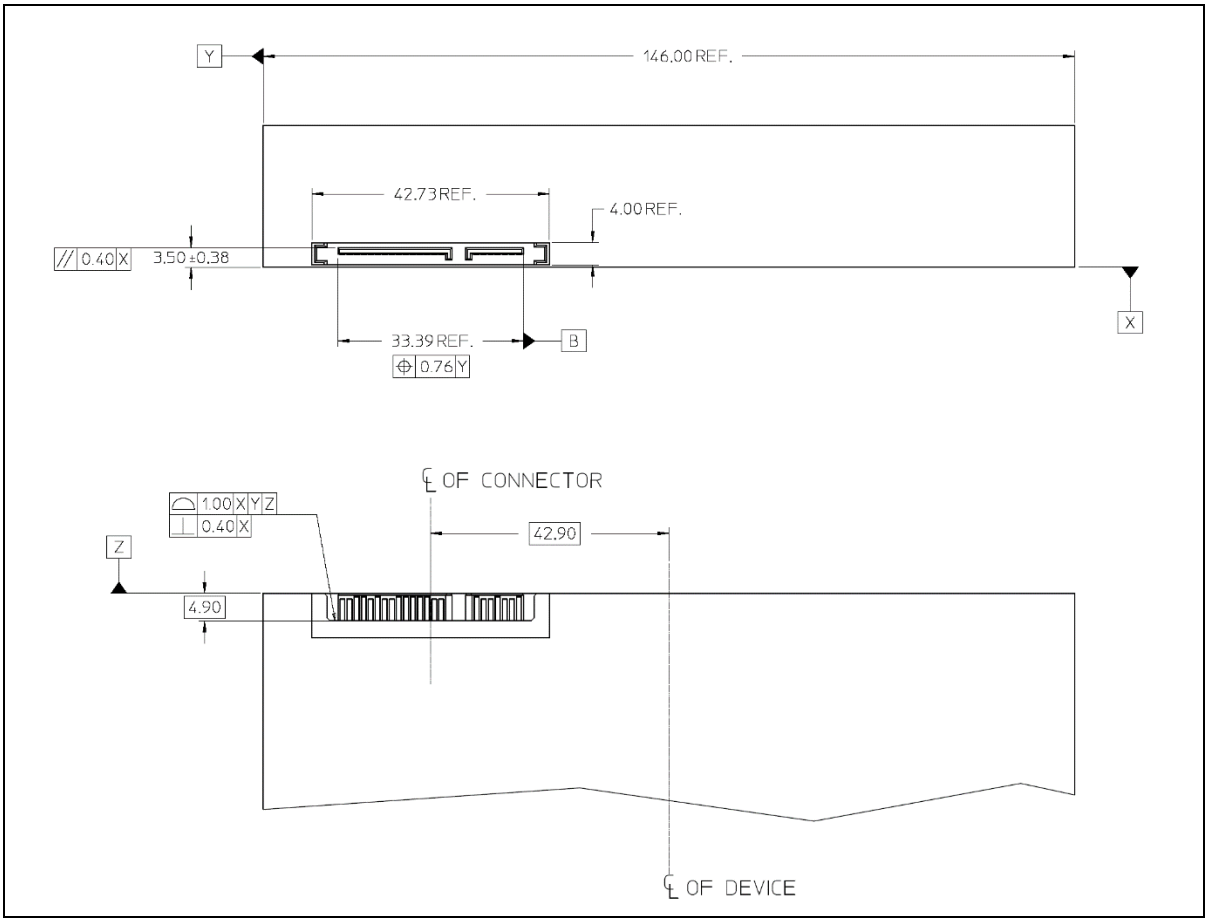


Figure 23 – Non-optical alternate device plug connector location on 5.25 inch form factor

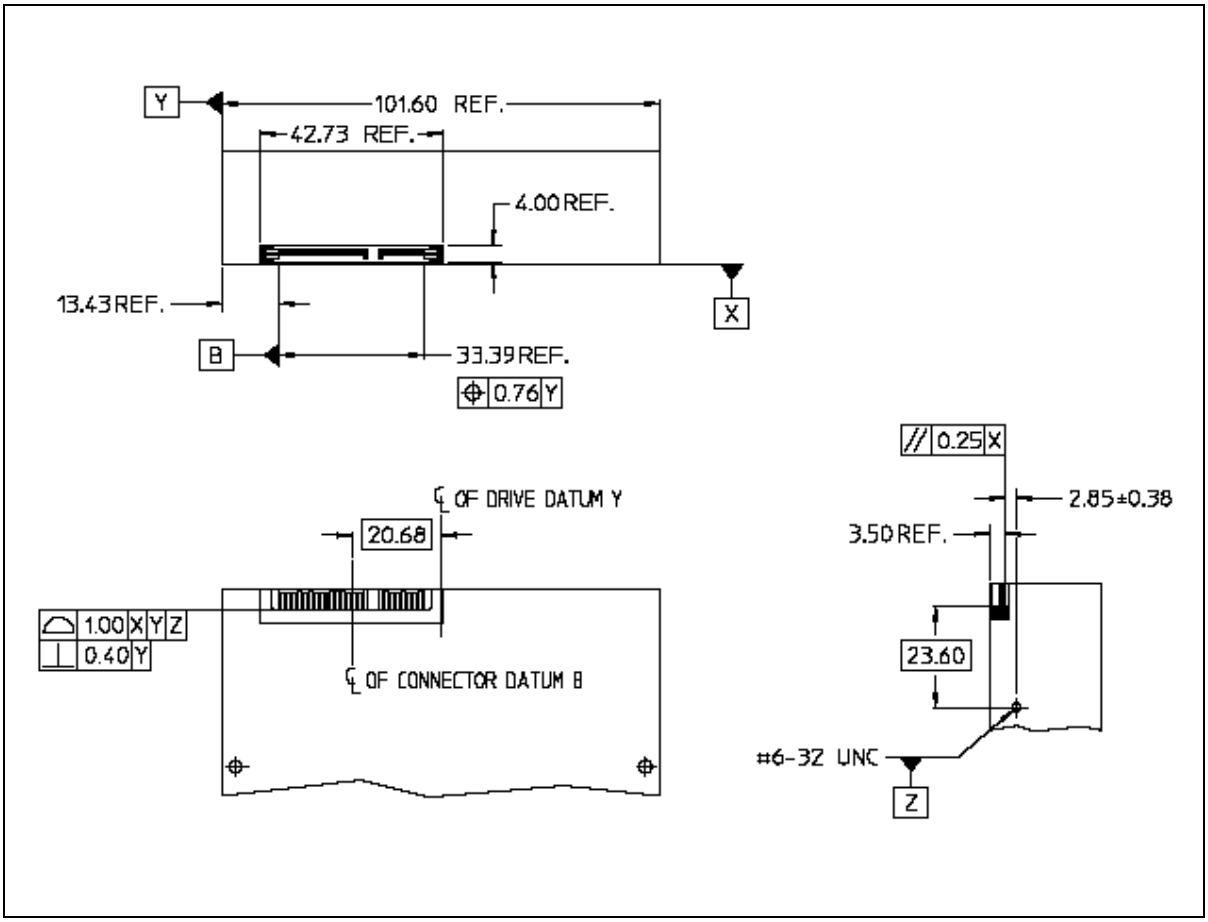


Figure 24 – Device plug connector location on 3.5 inch side mounted device

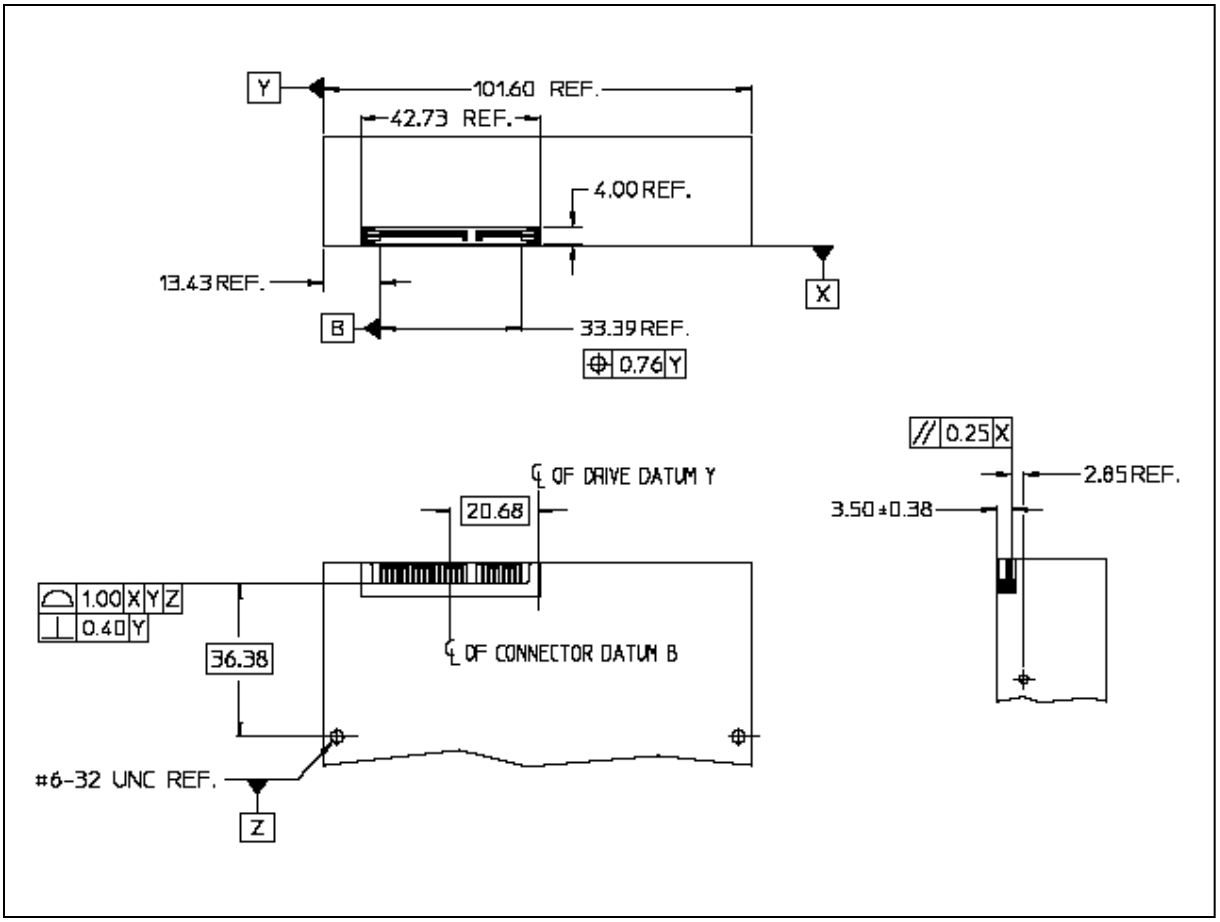


Figure 25 – Device plug connector location on 3.5 inch bottom mounted device

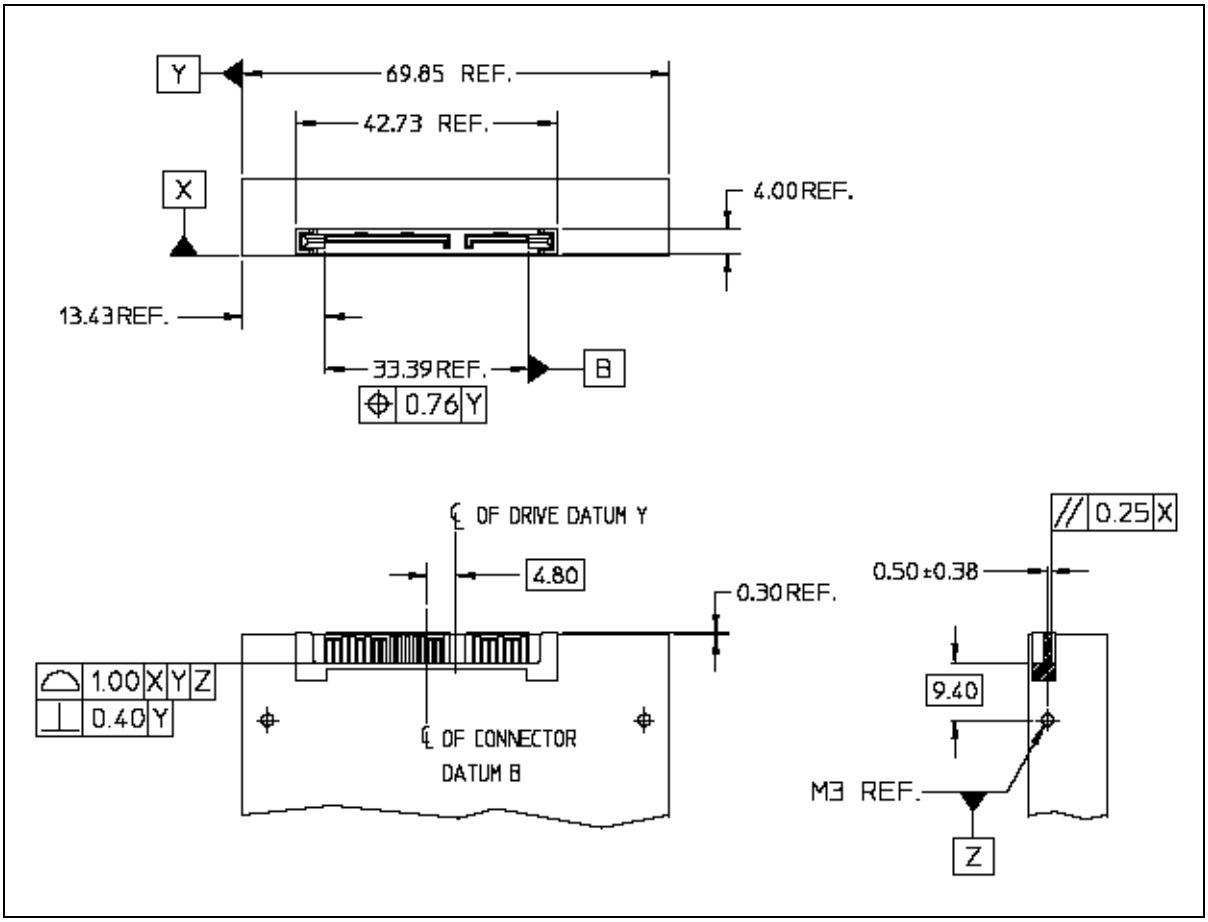


Figure 26 – Device plug connector location on 2.5 inch side mounted device

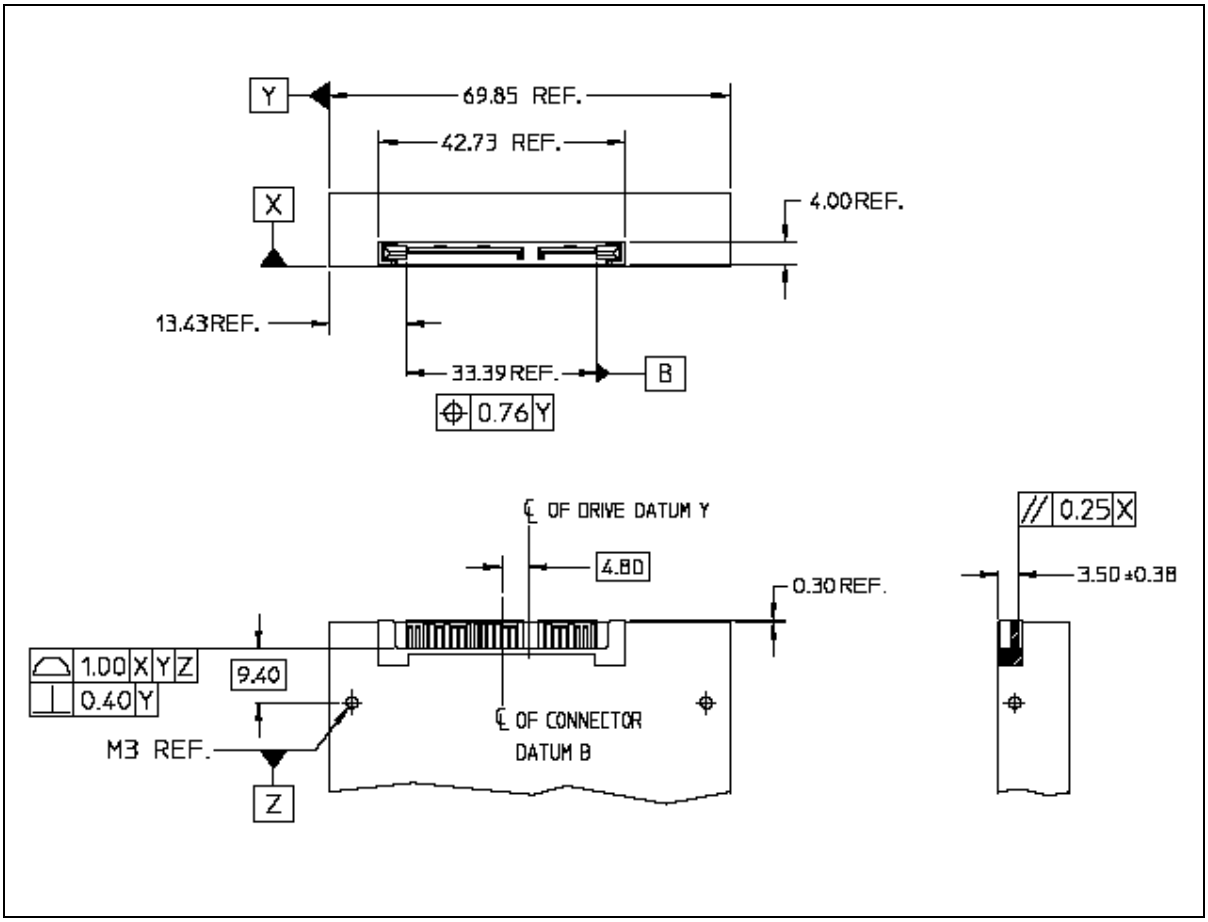


Figure 27 – Device plug connector location on 2.5 inch bottom mounted device

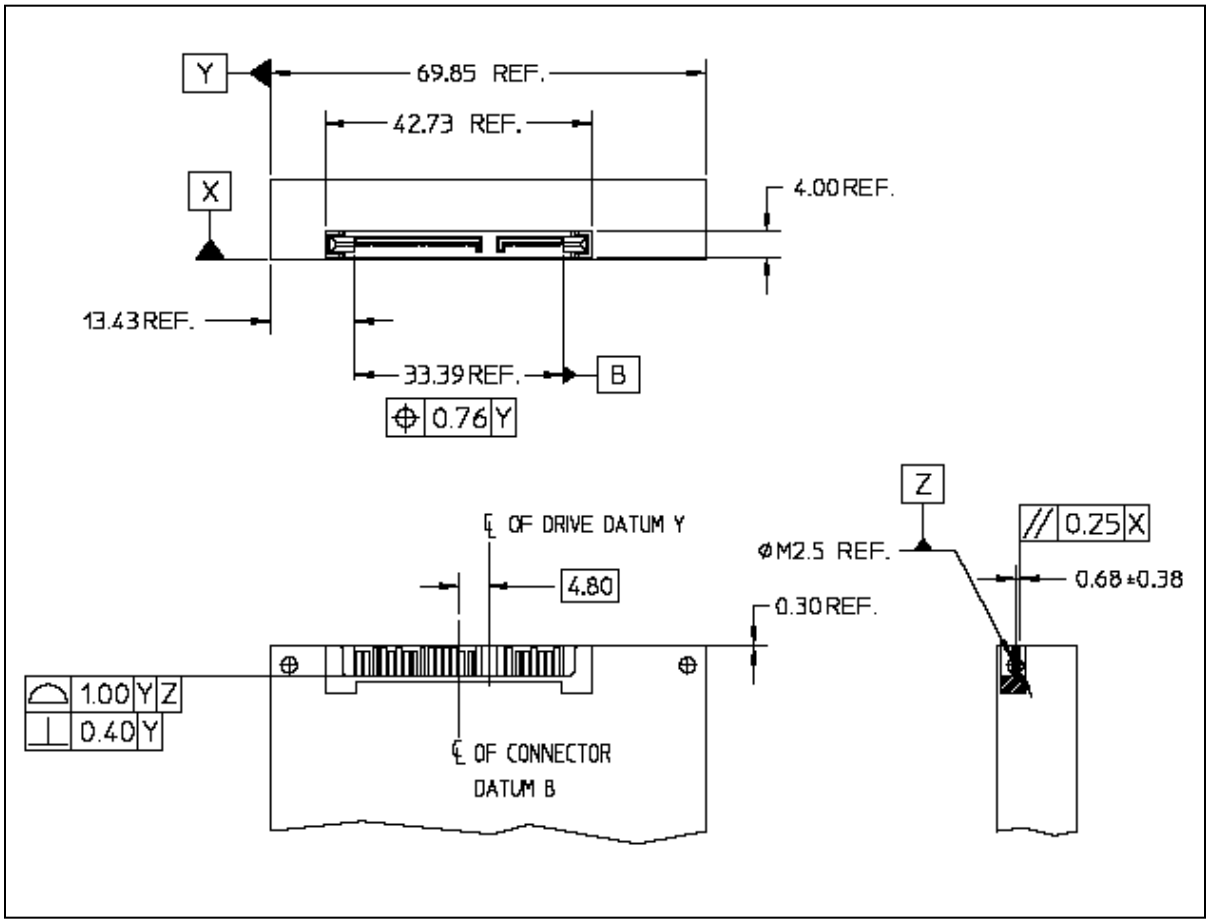


Figure 28 – Device plug connector location on 1.8 inch side mounted device

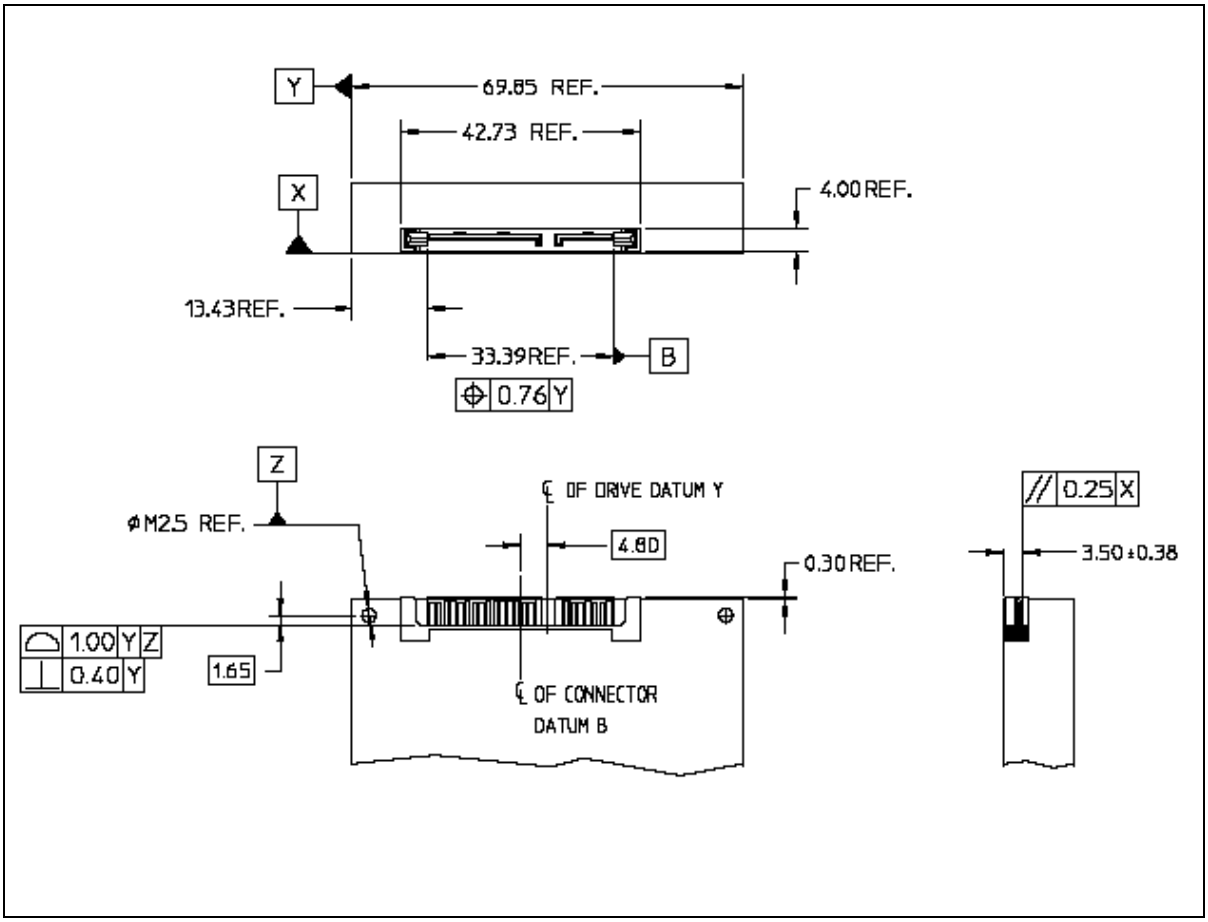


Figure 29 – Device plug connector location on 1.8 inch bottom mounted device

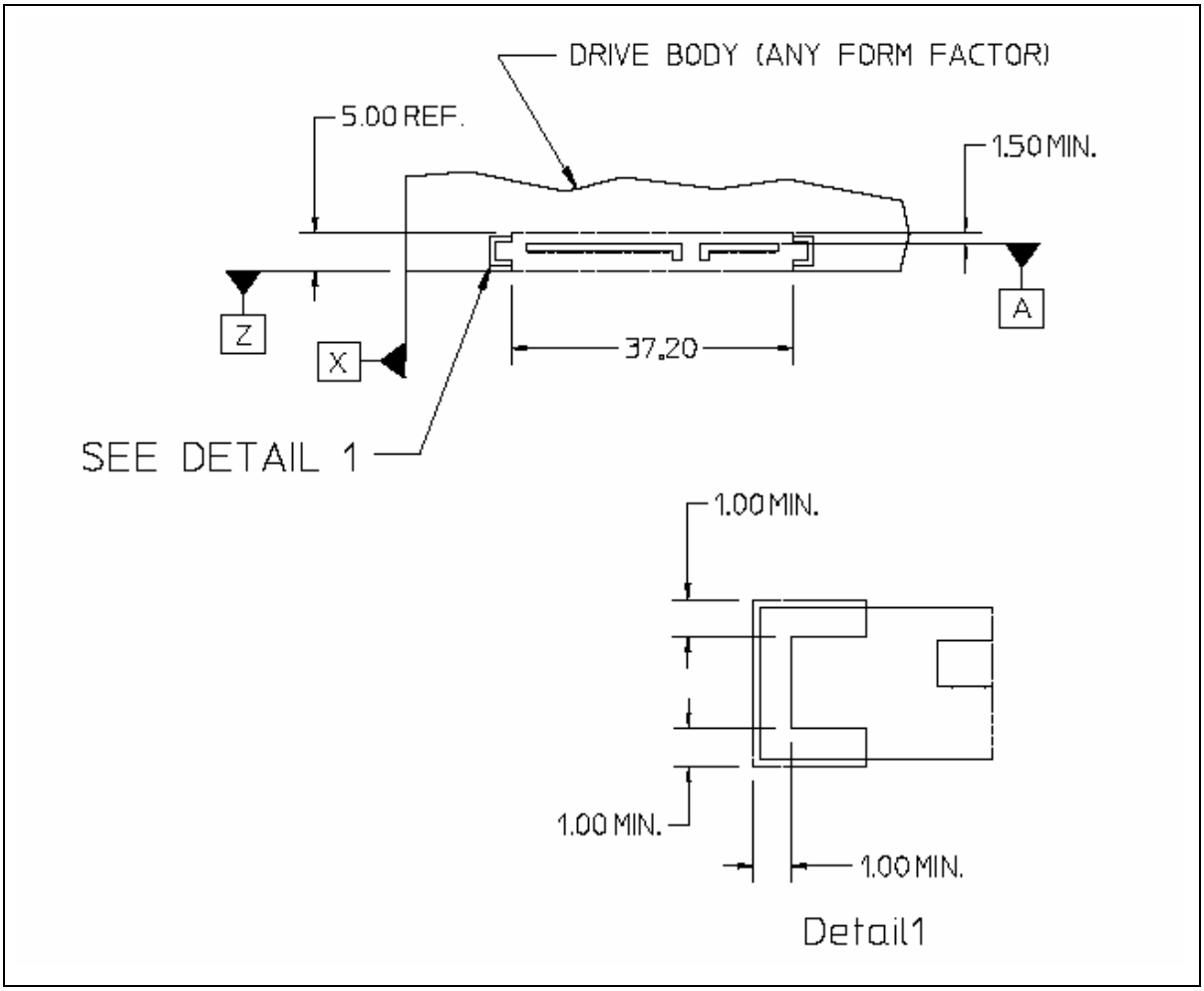


Figure 30 – Device plug connector keep out zones

NOTE 9 -The 1.50 mm keepout area above Datum A extends into the form factor to the connector Datum C in Figure 31.

NOTE 10 -The 1.00 mm keepout area shown in Detail 1 applies to both ends of the connector and extends from the connector housing outward to the outermost point of the form factor.

6.2.3 Mating interfaces

6.2.3.1 Device plug connector

Figure 31 and Figure 32 show the interface dimensions for the device plug connector with both signal and power segments. The device plug includes optional features to allow use of latching cables, fillets, and additional material to improve connector robustness. Table 5 defines the pin definitions and contact mating sequence for hot plug. These optional features should be included in all plug designs.

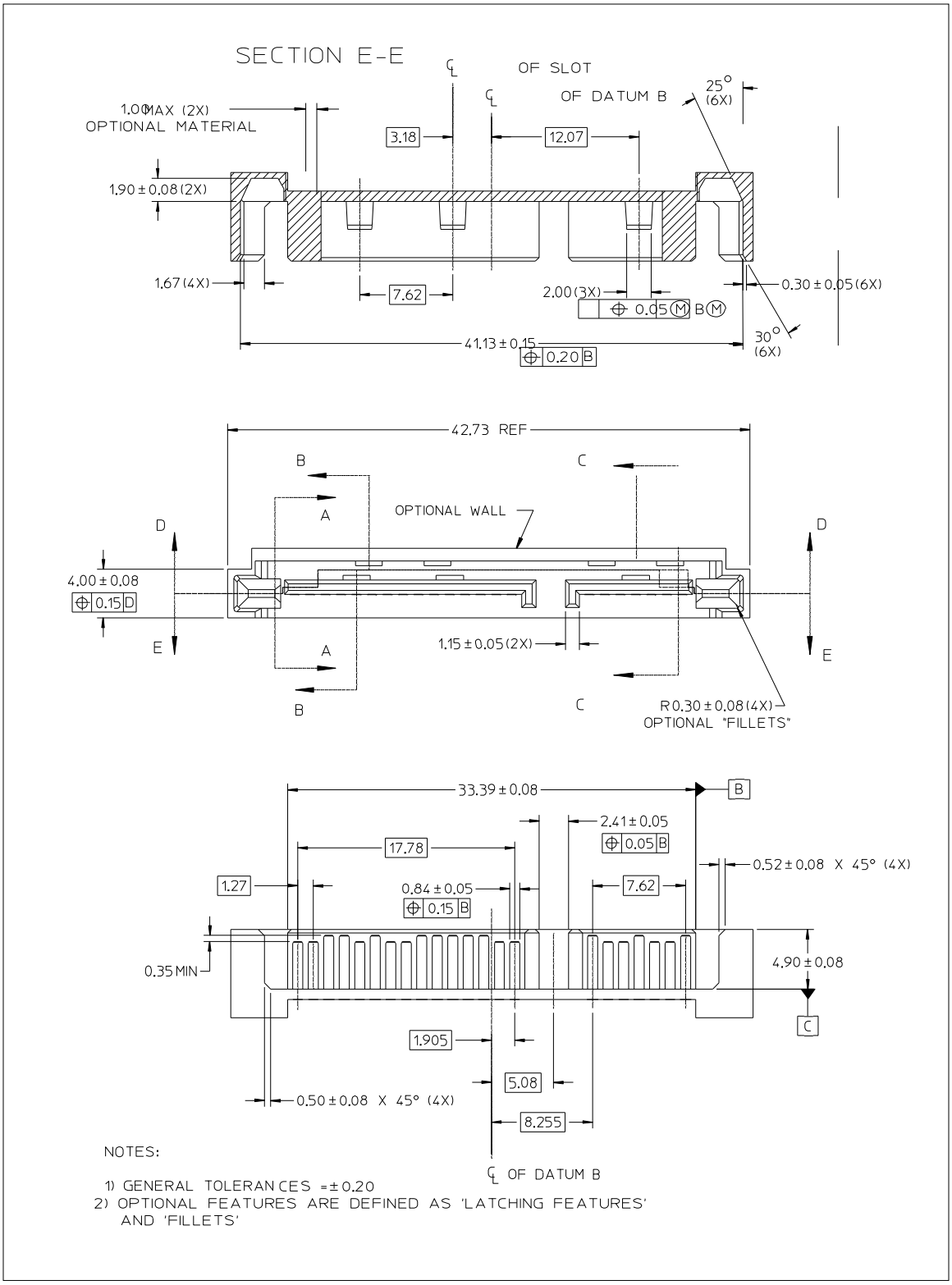


Figure 31 – Device plug connector

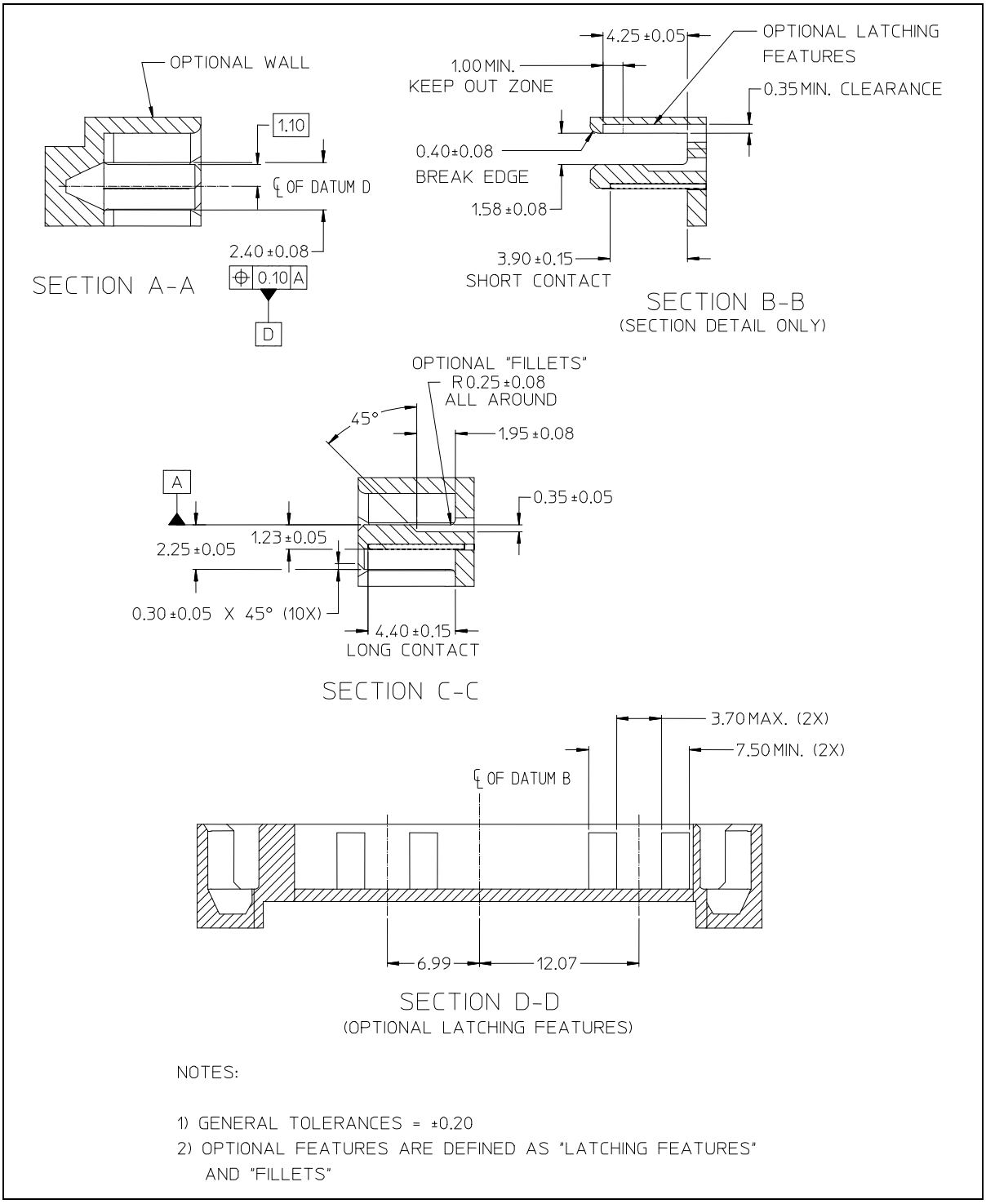


Figure 32 – Device plug connector (additional views)

6.2.3.2 Standard SATA connector (3.5 inch and 2.5 inch HDD)

Table 5 details the pin names, types, and contact order of the two SATA plug options. A brief description is also included for signal, ground and power pins. There are total of 7 pins in the signal segment and 15 pins in the power segment.

Table 5 – Standard SATA connector (3.5 inch and 2.5 inch HDD)

	Name	Type	Description	Cable Usage [b c]	Backplane Usage [c]
Signal Segment Key					
Signal Segment	S1	GND	Ground	1st Mate	2nd Mate
	S2	A+	Differential Signal Pair A	2nd Mate	3rd Mate
	S3	A-		2nd Mate	3rd Mate
	S4	GND	Ground	1st Mate	2nd Mate
	S5	B-	Differential Signal Pair B	2nd Mate	3rd Mate
	S6	B+		2nd Mate	3rd Mate
	S7	GND	Ground	1st Mate	2nd Mate
Signal Segment "L"					
Central Connector Gap [d]					
Power Segment "L"					
Power Segment	P1	Retired [e f g]		2nd Mate	3rd Mate
	P2	Retired [e f g]		2nd Mate	3rd Mate
	P3	DEVSLP [e] / PWDIS [e]	Enter/Exit DevSleep / Enter/Exit Power Disable	1st Mate	2nd Mate
	P4	GND	Ground	1st Mate	1st Mate
	P5	GND	Ground	1st Mate	2nd Mate
	P6	GND	Ground	1st Mate	2nd Mate
	P7	V_5	5 V Power, Pre-charge	1st Mate	2nd Mate
	P8	V_5	5 V Power	2nd Mate	3rd Mate
	P9	V_5	5 V Power	2nd Mate	3rd Mate
	P10	GND	Ground	1st Mate	2nd Mate
	P11	DAS/DSS/DHU	Device Activity Signal / Disable Staggered Spinup/ Direct Head Unload / Vendor Specific [a]	2nd Mate	3rd Mate
	P12	GND	Ground	1st Mate	1st Mate
	P13	V_{12}	12 V Power, Pre-charge	1st Mate	2nd Mate
	P14	V_{12}	12 V Power	2nd Mate	3rd Mate
	P15	V_{12}	12 V Power	2nd Mate	3rd Mate
Power Segment Key					

[a] For specific optional usage of pin P11 (see 6.13).
[b] Although the mate order is shown, hot plugging is not supported if using the cable connector receptacle.
[c] All mate sequences assume zero angular offset between connectors.
[d] The signal segment and power segment may be separate.
[e] Previous versions of this specification assigned 3.3 V to pins P1, P2, and P3. In addition, device plug pins P1, P2, and P3 were required to be bused together.
[f] If using DEVSLP, it is recommended to have P1 and P2 connected together for the purpose of legacy functionality.
[g] If using PWDIS, it is recommended to have P1 and P2 connected together for the purpose of legacy functionality.

Mating Configuration:

a) all pins are in a single row with 1.27 mm (0.050 inch) pitch;
b) all ground pins in the Serial ATA device plug power segment (i.e., connector pins P4, P5, P6, P10, and P12) shall be bussed together on the Serial ATA device;
c) the connection between the Serial ATA device signal ground and power ground is vendor specific;
d) the following sets of voltage pins in the Serial ATA device plug power segment shall be bussed together on the Serial ATA device:
    A) P7, P8, and P9, 5 V power delivery and precharge; and
    B) P13, P14, and P15, 12 V power delivery and precharge;
    and
e) the use of power delivery schemes that do not deliver all the specified voltages should only be used in scenarios where there is sufficient configuration control to ensure that the attached device does not require a supply voltage that is not provided.

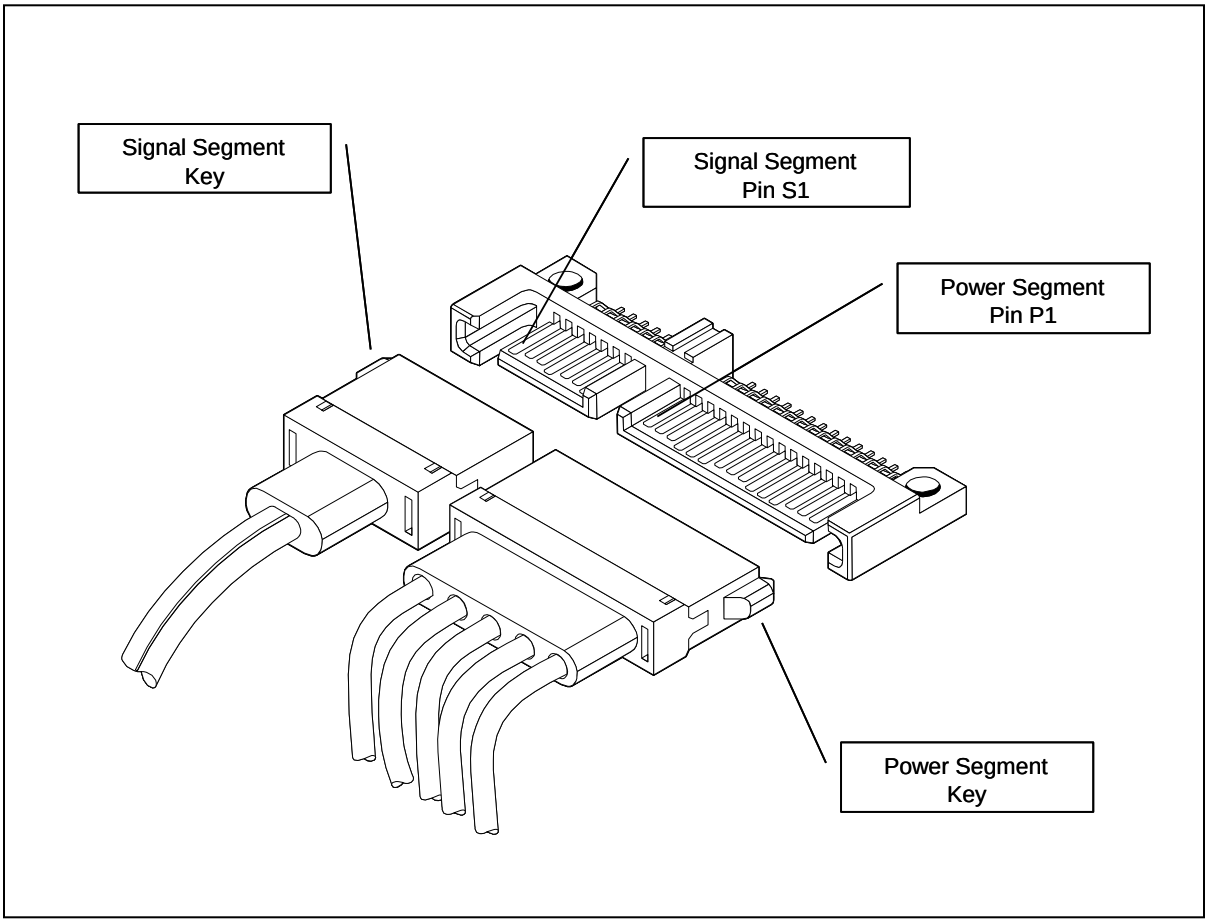


Figure 33 – Connector pin and feature locations

6.2.4 Signal cable receptacle connector

Figure 34 shows the interface dimensions for the signal cable receptacle connector. There are two identical receptacles at the two ends of the Serial ATA cable assembly. The cable receptacle mates with either the signal segment of the device plug connector on the device, or the host plug connector on the host.

Figure 35 defines an optional positive latch solution for internal cabled system applications. The latch requires the user to press and hold a release mechanism when disconnecting the cable. The latching feature option for device and host plug connectors are required in order to provide a latching surface. This latching feature option is called ClickConnect. Without a latching surface, there is no retention feature to hold a latching cable assembly in place.

It is optional to implement the latch on cable receptacles.

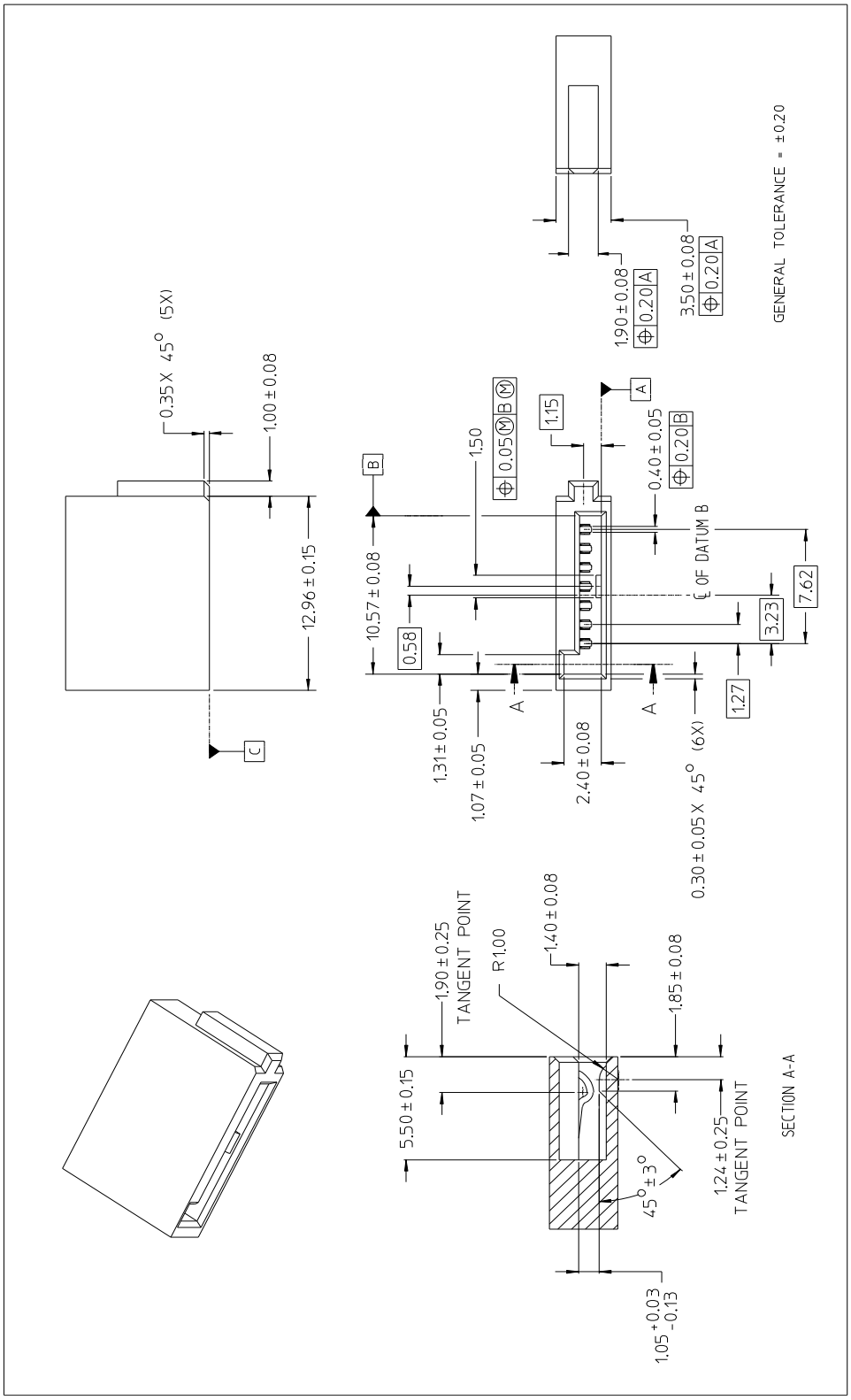


Figure 34 – Cable receptacle connector interface dimensions

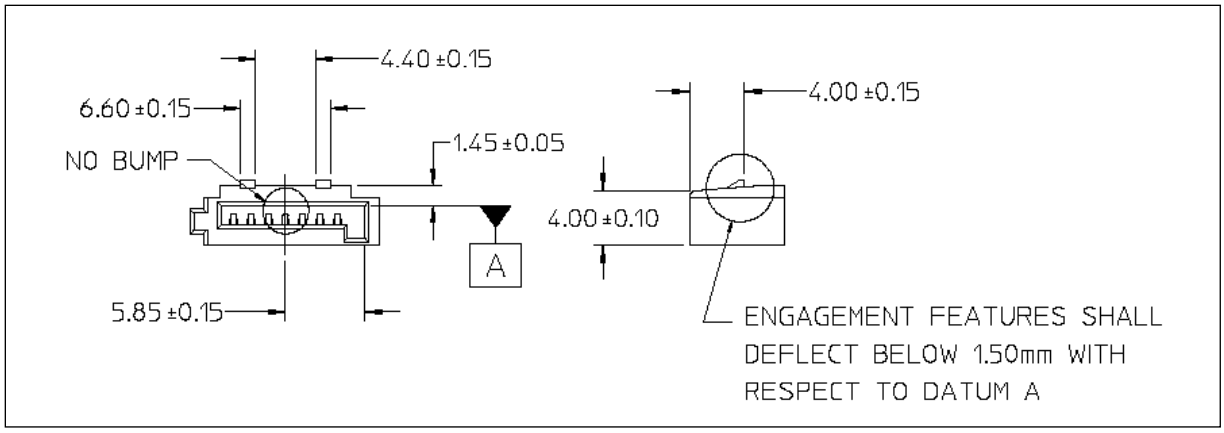


Figure 35 – Latching signal cable receptacle (ClickConnect)

The pin out of the cable receptacle connector is the mirror image of the signal segment of the device plug connector.

Notice that:

a) the two differential pin pairs are terminated with the corresponding differential cable pairs;
b) the ground pins are terminated with the cable drain wires, if it applies; and
c) the choice of cable termination methods, (e.g., crimping or soldering) is up to each connector vendor.

6.2.5 Host signal plug connector

6.2.5.1 Host signal plug dimentions

The host signal plug connector shall be mated with one end of the Serial ATA cable assembly. The pinout of the host plug connector is the mirror image of the signal cable receptacle. Figure 36 shows the host plug connector interface definition. The host plug includes optional features to allow use of latching cables, fillets, and additional material to improve connector robustness.

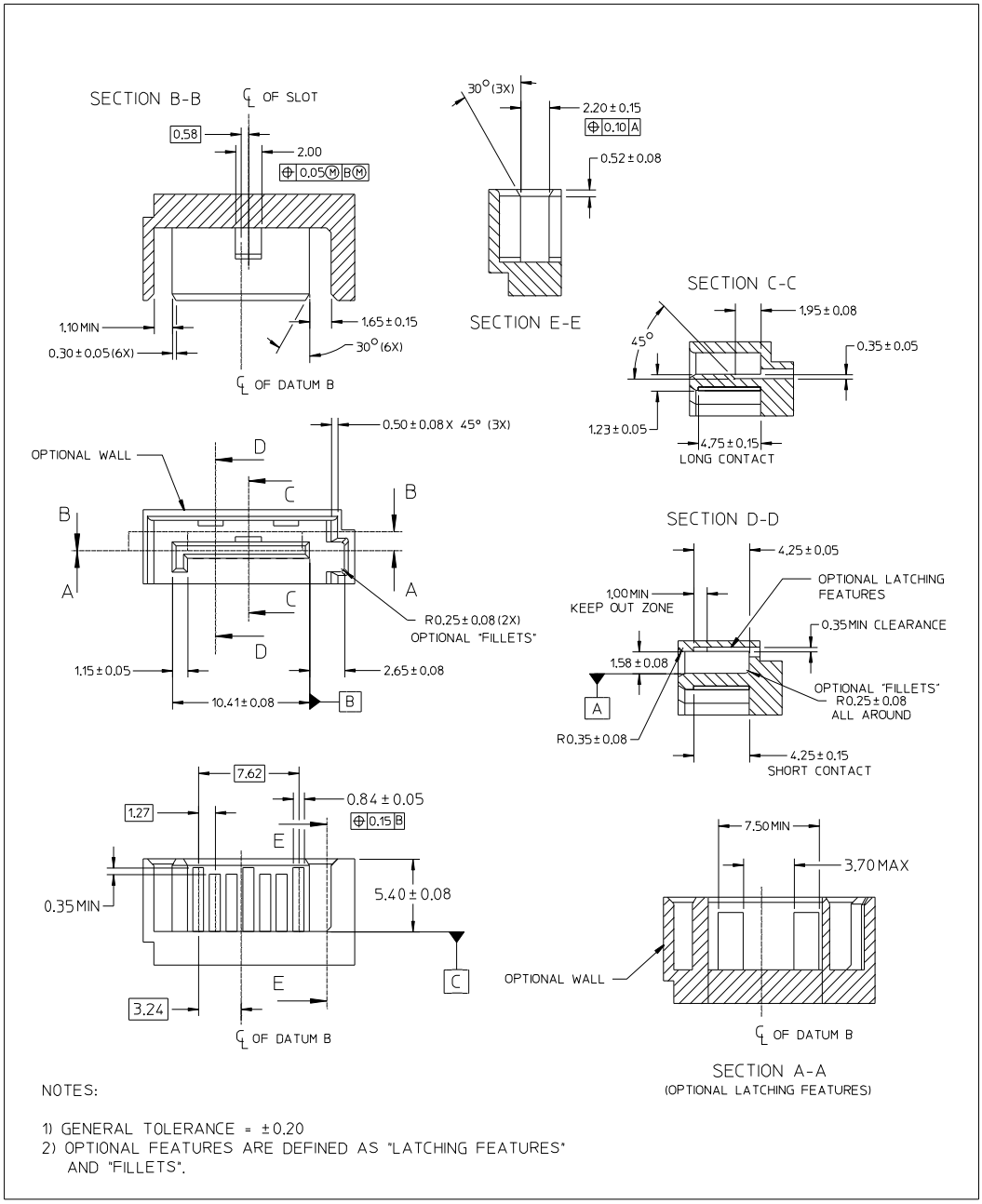


Figure 36 – Host signal plug connector interface dimensions

6.2.5.2 Internal plug stacking

The purpose of this recommended layout is to conserve motherboard, HBA and I/O controller printed circuit board space to support system density and size goals for such products.

For applications where multiple Serial ATA ports or connectors are stacked together on the host, there is a clearance or spacing requirement to prevent the cable assemblies from interfering with each other. Figure 37 shows the recommended clearance or spacing. Figure 38 shows the recommended clearance and orientation to allow access for latching cables.

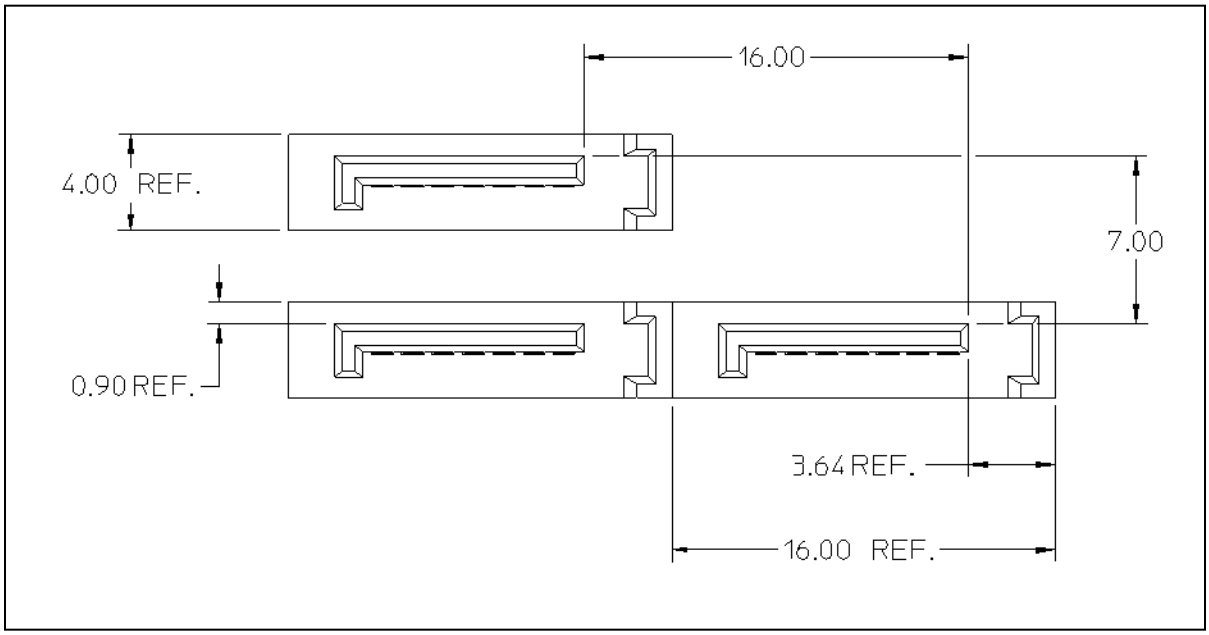


Figure 37 – Non-latching connector stack spacing and orientation

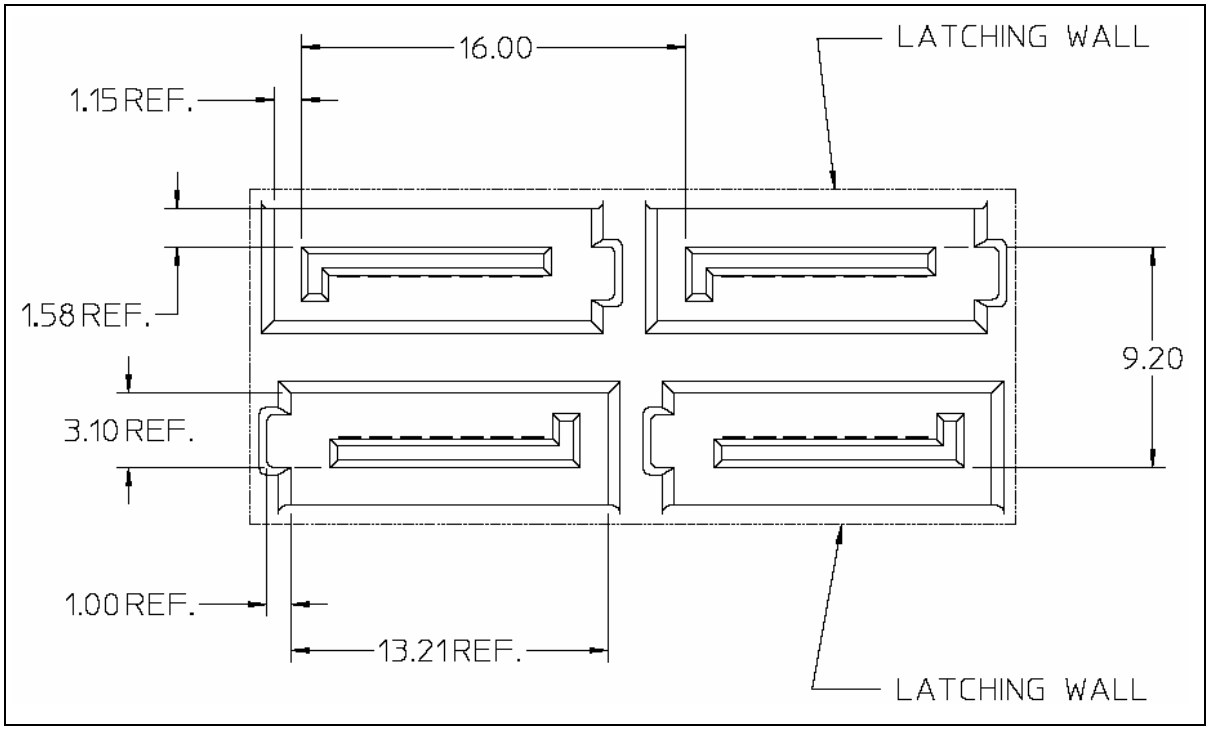


Figure 38 – Latching connector stack spacing and orientation

6.2.6 Backplane connector

6.2.6.1 Backplane connector dimentions

The backplane connector is to be blind-mated directly with the device plug connector. The interface dimensions for the backplane connector are shown in Figure 39.

NOTE 11 - Note that dimension B allows two values:

a) 8.15 mm; and
b) 14.15 mm.

There are two levels of contacts in the backplane connector. The advancing ground contacts P4 and P12 mate first with the corresponding ground pins on the device plug connector, followed by the engaging of the pre-charged power pins. An appropriate external retention mechanism independent of the connector is required to keep the host PCB and the device in place. The backplane connector is not designed with any retention mechanism.

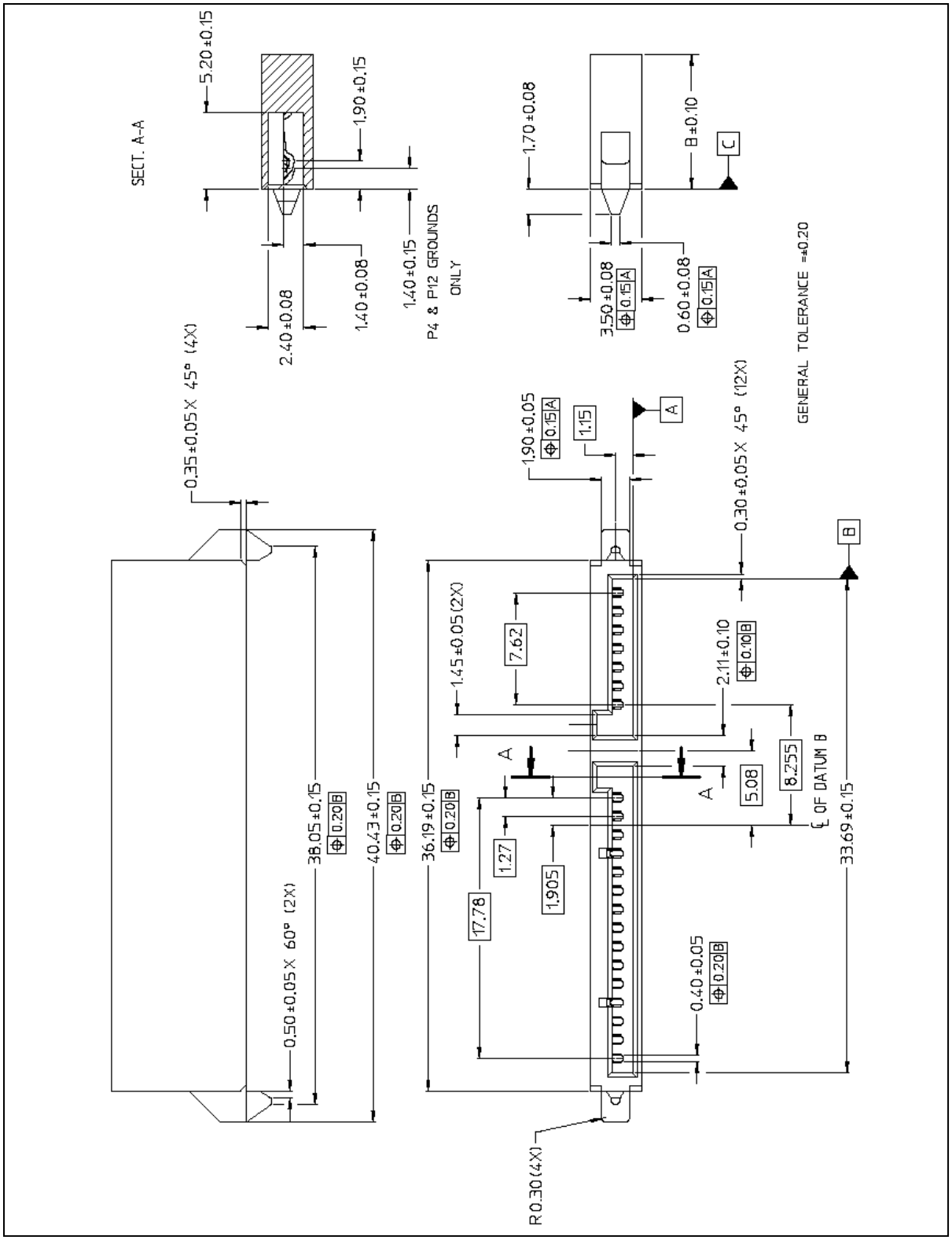


Figure 39 – Backplane connector interface dimensions

6.2.6.2 Backplane connector configuration and blind-mating tolerance

The maximum blind-mate misalignment tolerances are ±1.50 mm and ±1.00 mm, respectively, for two perpendicular axes illustrated in Figure 40. Any skew angle of the plug, with respect to the receptacle, reduces the blind-mate tolerances.

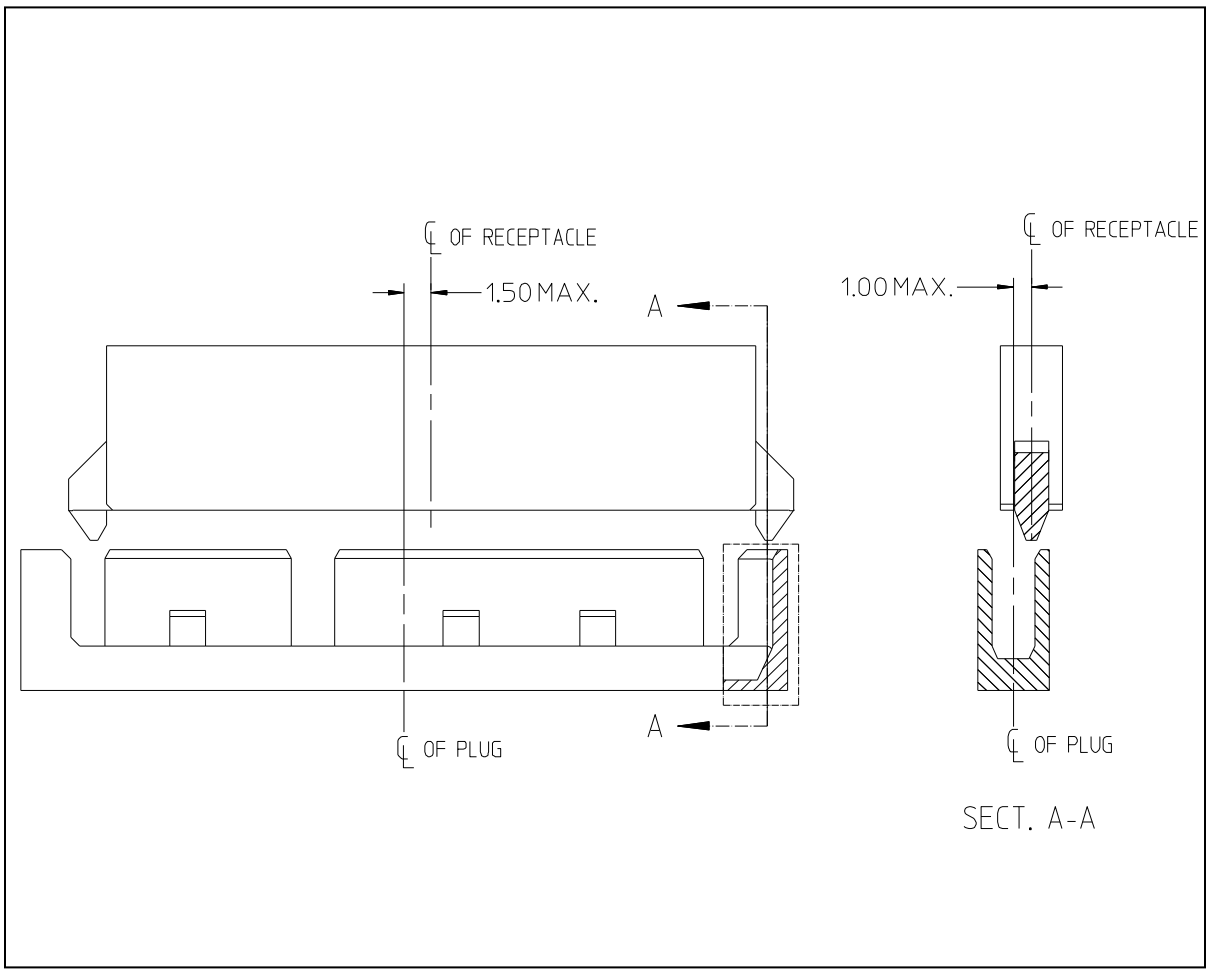


Figure 40 – Connector pair blind-mate misalignment tolerance

The device-to-backplane mating configuration is shown in Figure 41. The allowed values for dimension A and dimension B are shown in Table 6.

Table 6 – Allowed values for dimension A and B for device-to-backplane mating

Description	Standard	Extended
Device mated height (A)	8.45 mm	14.45 mm
Component clearance (B)	3.55 mm	9.55 mm

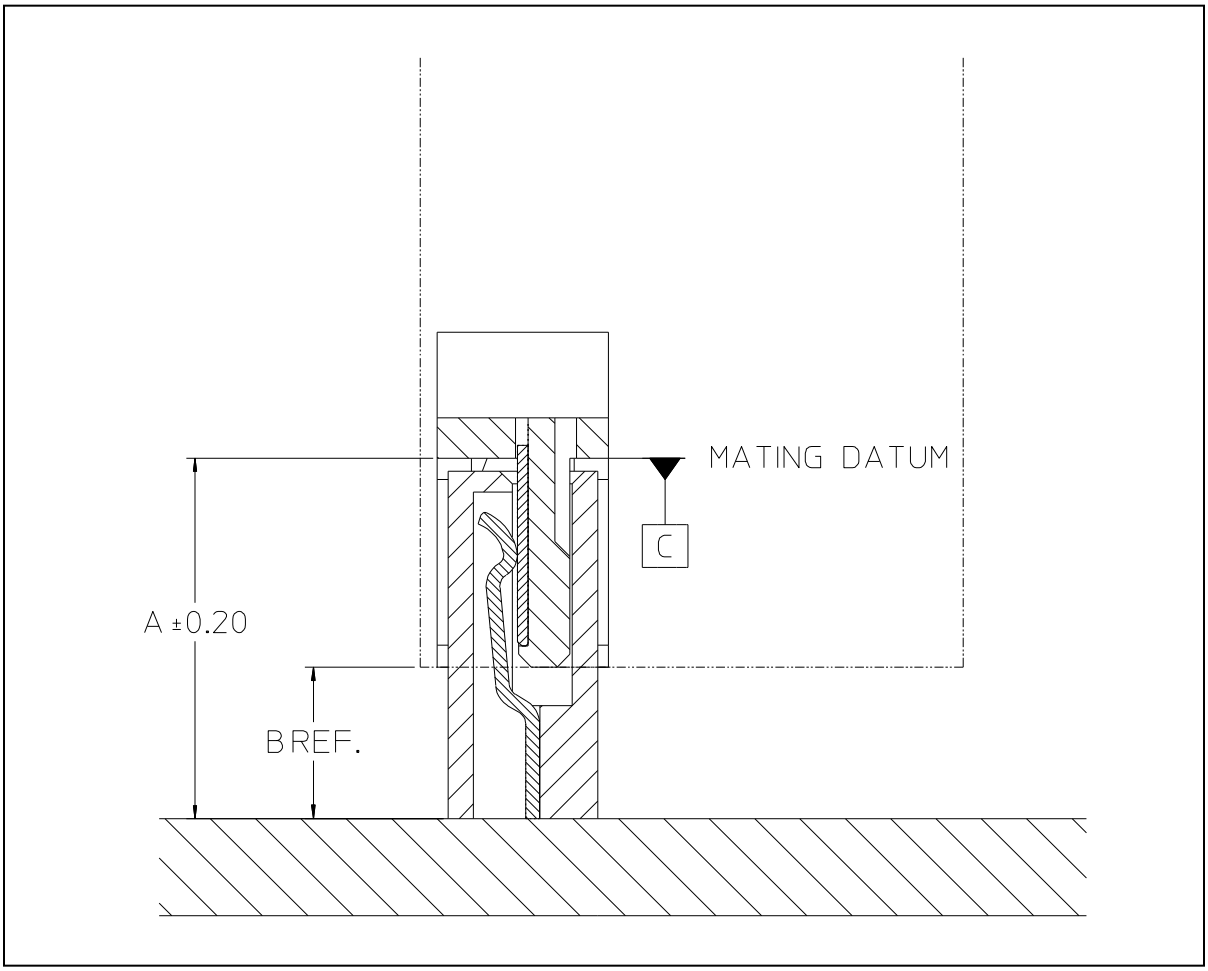


Figure 41 – Device-backplane mating configuration

6.2.7 Power cable receptacle connector

The power cable receptacle connector mates with the power segment of the device plug, bringing power to the device. Figure 42 shows the interface dimensions of the power receptacle connector. The pinout of the connector is the mirror image of the power segment of the device plug shown in Table 5. Figure 43 defines an optional positive latch solution for internal cabled system applications. The latch requires the user to press and hold a release mechanism if disconnecting the cable. The latching feature option for device plug connectors is required in order to provide a latching surface. Without a latching surface, there is no retention feature to hold a latching cable assembly in place.

It is optional to implement the latch on cable receptacles.

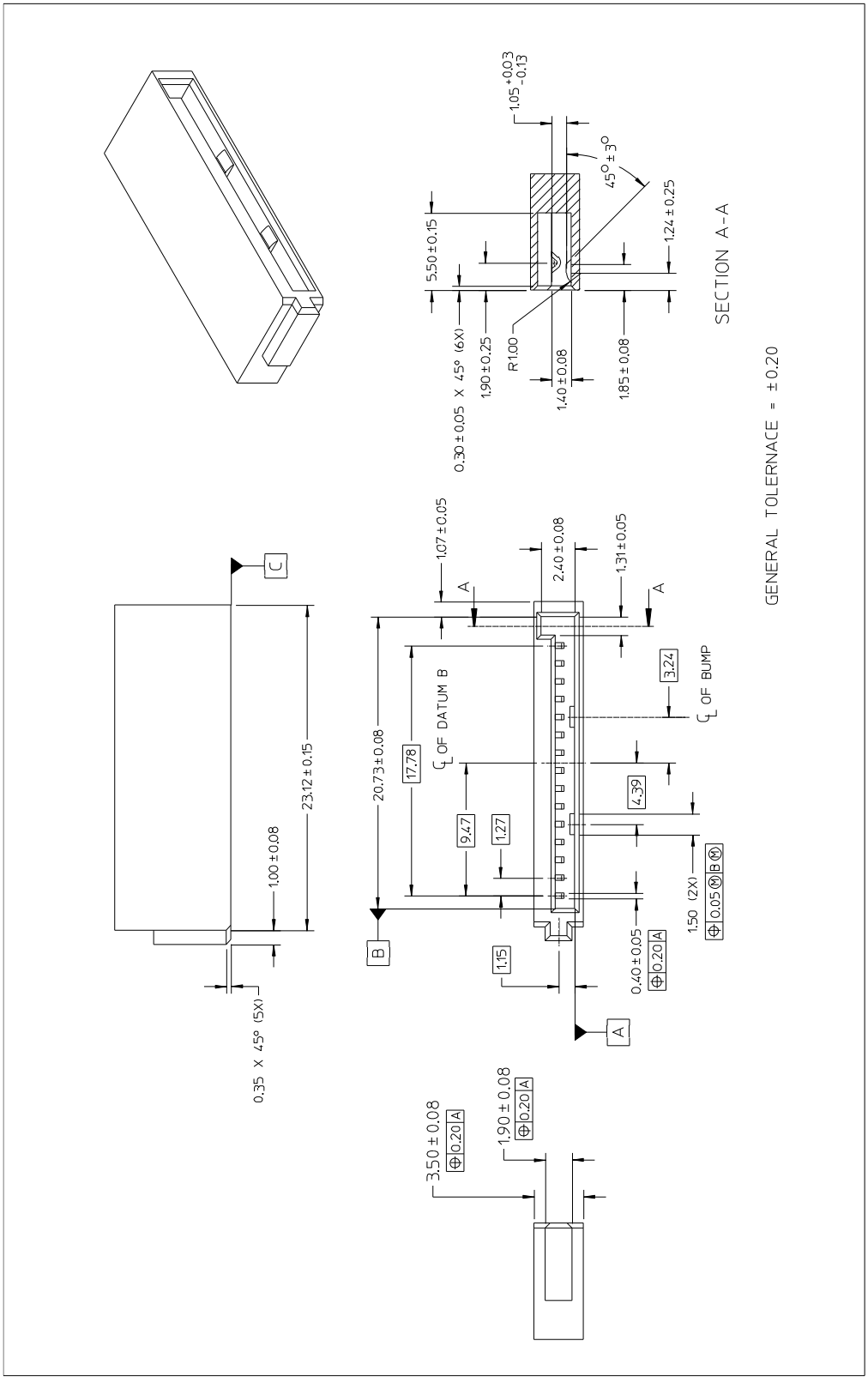


Figure 42 – Power receptacle connector interface dimensions

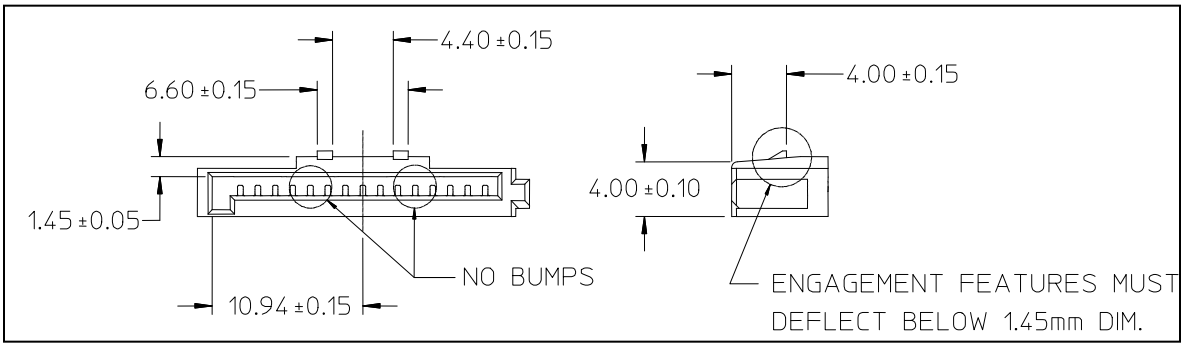


Figure 43 – Latching power cable receptacle

The power receptacle connector is terminated onto 18 American wire gauge (AWG) wires that are connected to the system power supply or other power sources. Five 18 AWG wires may be used, with three wires terminated to the nine power pins for the three voltages, while the remaining two wires to the six ground pins.

6.2.8 Internal single lane cable material

The internal single lane cable consists of four conductors in two differential pairs. If necessary, the cable may also include drain wires to be terminated to the ground pins in the Serial ATA cable receptacle connectors. The conductor size may be 30 AWG to 26 AWG. The cable maximum length is 1 m.

This specification does not specify a standard internal single lane cable. Any cable that meets the electrical requirements given in 6.4 is considered an acceptable internal single lane cable. The connector and cable vendors have the flexibility to choose cable constructions and termination methods based on performance and cost considerations. An example cable construction is given in Figure 44 for an informational purpose only.

Although construction methodologies are not specified, there are a few essential elements of the Serial ATA cable that should be considered.

Physical characteristics of the Serial ATA cable may include the following items:

a) shielded Pairs (i.e., 2);
b) solid Tinned Copper (e.g., 26 AWG);
c) white Foam Polyolefin (e.g., 43.5 mil Diameter);
d) parallel Drain Pairs (2 pair., 28 AWG – Solid Tinned Copper);
e) aluminized Polyester Foil (1 mil thick with 35 mil overlap);
f) foil may be the blue longitudinal wrap that is sealed with heat; and
g) jacket (20 mil polyvinyl chloride (PVC) wall).

See Figure 44 for details.

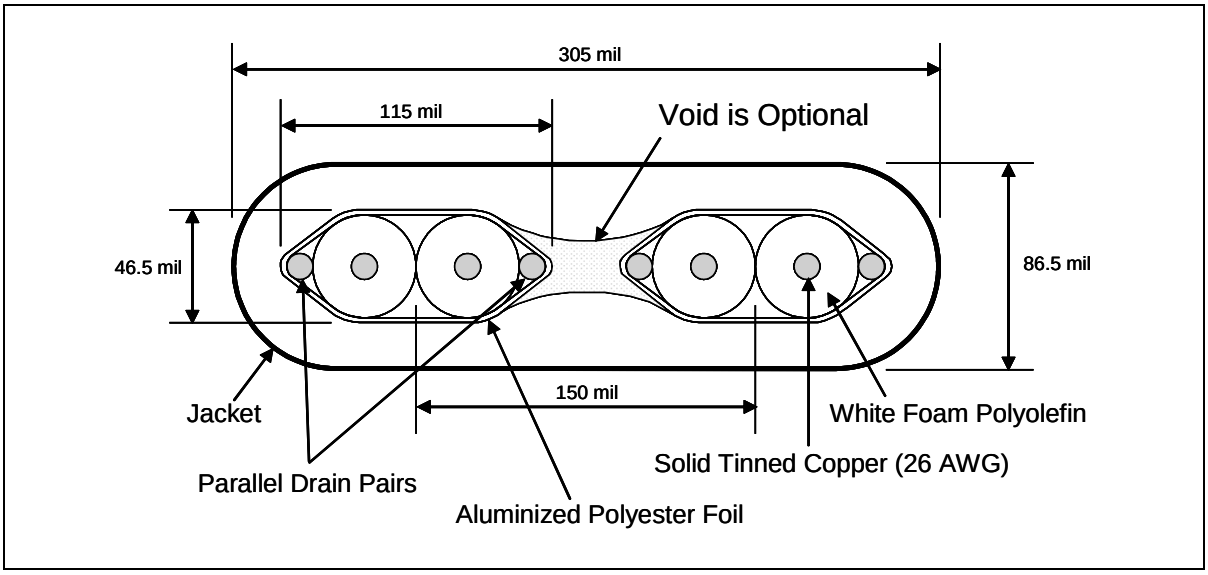


Figure 44 – Detailed cross-section of an example internal single lane cable

6.2.9 Connector labeling

Labeling on a connector with the connector manufacturer identifier or Serial ATA icon is optional.

6.2.10 Connector and cable assembly requirements and test procedures

6.2.10.1 Connector and cable assembly requirements and test procedures overview

Unless otherwise specified, all measurements shall be performed within the following lab conditions:

a) mated;
b) temperature in the range of 15 °C to 35 °C;
c) relative humidity in the range of 20 % to 80 %;
d) atmospheric pressure in the range of 650 mmHg to 800 mmHg.

If an Electronic Industry Association (EIA) test is specified without a letter suffix in the test procedures, the latest approved version of that test shall be used.

6.2.10.2 Housing and contact electrical requirements

Table 7 specifies the connector housing and contact electrical requirements.

Table 7 – Housing and contact electrical parameters, test procedures, and requirements

Parameter	Procedure	Requirement
Insulation resistance	EIA-364-21 After 500 VDC for one minute, measure the insulation resistance between the adjacent contacts of mated and unmated connector assemblies.	Min 1 000 Mohm
Dielectric withstanding voltage	EIA-364-20 Method B Test between adjacent contacts of mated and unmated connector assemblies.	The dielectric shall withstand 500 VAC for one minute at sea level.
Low level contact resistance (LLCR)	EIA-364-23 Subject mated contacts assembled in housing to max 20 mV open circuit at max 100 mA.	Initial max 30 mohm. Max resistance increase 15 mohm after stress.
Contact current rating (Power segment)	The test procedure is: 1) mount connector to a test PCB; 2) wire 3 adjacent pins in parallel for supply (or the minimum number required by the connector type); 3) wire 3 adjacent pins in parallel for return (or the minimum number required by the connector type); 4) apply a DC current of 3 times the current rating per contact to the supply pins, returning through the return pins; and 5) record temperature rise when thermal equilibrium is reached.	Min 1.5 A/pin. The temperature rise above ambient shall not exceed 30 °C at any point in the connector when contact positions are powered. The ambient condition is still air at 25 °C.

6.2.10.3 Mechanical and environmental requirements

Table 8 lists the mechanical parameters and requirements, while Table 9 the environmental and reliability tests and requirements.

Table 8 – Mechanical test procedures and requirements

Parameter	Procedure	Requirement
Visual and dimensional inspections	EIA-364-18 Visual, dimensional and functional per applicable quality inspection plan.	Meets product drawing requirements.
Cable pull-out	EIA-364-38 Condition A Subject a Serial ATA cable assembly to a 40 N axial load for a minimum of one minute while clamping one end of the cable plug.	No physical damage. Cable shall meet all connector and cable mechanical requirements before and after the completion of the test.
Cable flexing	For round cable use EIA-364-41 Condition I Dimension x = 3.7 × cable diameter, 100 cycles in each of two planes. For flat cable use EIA-364-41 Condition II 250 cycles using either method 1 or 2.	No physical damage. No discontinuity over 1 us during flexing.
Insertion force Cabled signal connector	EIA-364-13 Measure the force necessary to mate the connector assemblies at a max. rate of 12.5 millimeter per minute.	Max 45 N.
Removal force Cabled signal connector (Non-latching)	EIA-364-13 Measure the force necessary to unmate the connector assemblies at a max. rate of 12.5 millimeter per minute.	Min 10 N through 50 cycles.
Insertion force Cabled power connector	EIA-364-13 Measure the force necessary to mate the connector assemblies at a max. rate of 12.5 millimeter per minute.	Max 45 N.
Removal force Cabled power connector (Non-latching)	EIA-364-13 Measure the force necessary to unmate the connector assemblies at a max. rate of 12.5 millimeter per minute.	Min 15 N for cycles 1 to 5 Min 10 N through 50 cycles.
Insertion force Backplane connector	EIA-364-13 Measure the force necessary to mate the connector assemblies at a max. rate of 12.5 millimeter per minute.	Max 20 N.
Removal force Backplane connector	EIA-364-13 Measure the force necessary to unmate the connector assemblies at a max. rate of 12.5 millimeter per minute.	Min 4 N after 500 cycles.
Removal force Cabled Latching connector Includes power and signal connectors	EIA-364-13 Apply a static 25 N unmating test load	No damage.
Durability	EIA-364-09 50 cycles for internal cabled application; 500 cycles for backplane/blindmate application. Test done at a maximum rate of 200 cycles per hour.	No physical damage. Meet requirements of additional tests as specified in the test sequence in Table 11.

Table 9 – Environmental parameters, test procedures, and requirements

Parameter	Procedure	Requirement
Physical shock	EIA-364-27 Condition H Subject mated connectors to 30 g's half-sine shock pulses of 11 ms duration. Three shocks in each direction applied along three mutually perpendicular planes for a total of 18 shocks [b].	No discontinuities of 1 us or longer duration. No physical damage.
Random vibration	EIA-364-28 Condition V Test letter A Subject mated connectors to 5.35 g's RMS. Thirty minutes in each of three mutually perpendicular planes [b].	No discontinuities of 1 us longer duration.
Humidity	EIA-364-31 Method II Test Condition A. Subject mated connectors to 96 hours at 40 °C with 90 % RH to 95 % RH [a].	
Temperature life	EIA-364-17 Test Condition III Method A. Subject mated connectors to temperature life at +85 °C for 500 hours [a].	
Thermal shock	EIA-364-32 Test Condition I. Subject mated connectors to 10 cycles between –55 °C and +85 °C [a].	
Mixed Flowing Gas	EIA-364-65, Class 2A Half of the samples are exposed unmated for seven days, then mated for remaining seven days. Other half of the samples are mated during entire testing [a].	

[a] Shall meet EIA-364-18 Visual Examination requirements, show no physical damage, and shall meet requirements of additional tests as specified in the test sequence in Table 11.

[b] Shock and vibration test fixture is to be determined by each user with connector vendors.

An additional requirement is given in Table 10.

Table 10 – Additional requirement

Parameter	Procedure	Requirement
Flammability	UL 94V-0	Material certification or certificate of compliance required with each lot to satisfy the Underwriters Laboratories follow-up service requirements.

It should be pointed out that this specification does not attempt to define the connector and cable assembly reliability requirements that are considered application-specific. It is up to users and their connector suppliers to determine if additional requirements shall be added to satisfy the application needs.

EXAMPLE - For example, a user who requires a Surface Mount Technology (SMT) connector may want to include additional requirements for SMT connector reliability.

6.2.10.4 Sample selection

Samples shall be prepared in accordance with applicable manufacturers' instructions and shall be selected at random from current production. Each test group shall provide 100 data points for a good statistical representation of the test result. For a connector with greater than 20 pins, a test group shall consist of a minimum of five connector pairs. From these connector pairs, a minimum of 20 contact pairs per mated connector shall be selected and identified. For connectors with less than 20 pins, choose the number of connectors sufficient to provide 100 data points.

6.2.10.5 Test sequence

Table 11 shows the connector test sequences for five groups of tests.

Table 11 – Connector test sequences

Test or examination	Test group				
	A	B	C	D	E
Examination of the connector(s)	1, 5	1, 9	1, 8	1, 8	1, 7
Low-Level Contact Resistance (LLCR)	2, 4	3, 7	2, 4, 6		4, 6
Insulation resistance				2, 6	
Dielectric withstanding voltage				3, 7	
Current rating			7		
Insertion force		2			
Removal force		8			
Durability	3	4 [a]			2 [a]
Physical shock		6			
Vibration		5			
Humidity				5	
Temperature life			3		
Reseating (manually unplug/plug three times)			5		5
Mixed Flowing Gas					3
Thermal shock				4	

[a] Preconditioning, 20 cycles for the 50 durability cycle requirement, 50 cycles for the 500 durability cycle requirement. The insertion and removal cycle is at the maximum rate of 200 cycles per hour.

EXAMPLE - For example, in Test Group A, one is able to perform the following tests:

1) examination of the connector(s);
2) LLCR;
3) durability;
4) LLCR; and
5) examination of the connector(s).

6.2.11 Internal Multilane cables

6.2.11.1 Internal Multilane cable overview

This section defines standard cable assemblies and headers for connecting multiple Serial ATA links from a RAID HBA to a backplane within the same enclosure or server, or for connecting multiple Serial ATA links from a HBA to individual devices.

This cable/connector is based on the SFF-8484 specification. The SFF-8484 specification is also used by SAS.

6.2.11.2 Conformance criteria

6.2.11.2.1 Conformance criteria list

The conformance criteria is:

a) 2 or 4 lanes, Serial ATA signals;
b) either point to point with a high density connector on both ends of the cable or fanout with a high density connector on one end of the cable assembly and individual single lane connectors on the other end of the cable assembly;
c) additional pins/conductors for sideband signals;
d) specifications for PCB footprint for SMT, thru hole and press fit;
e) Rx, Tx, Rx, Tx pin sequencing to minimize crosstalk;
f) ground reference between each pair;
g) flexible cable for routing and airflow;
h) performance supporting 1.5 Gbit/s and 3.0 Gbit/s; and
i) compliance points are at the ends of a mated cable interface.

If additional interconnect media between host and device exists, that portion of the design is proprietary and shall ensure the mated cable interface compliance points are met.

6.2.11.2.2 Electrical requirements

The Internal Multilane cable assembly shall meet the electrical characteristics defined in Table 36. The Internal Multilane cable assembly shall operate at Gen1i and Gen2i levels and meet the electrical characteristics defined in Table 36. Since this cable is a Multilane and therefore has multi-aggressors, the additional requirement is to have crosstalk measured using the multilane crosstalk (ML-CXT) method. The measured crosstalk shall meet the requirements given in Table 36.

6.2.11.2.3 Component descriptions

Three components are defined in this section for Multilane applications. Each component has a 2 Lane and a 4 Lane version. Pin assignments are provided at the end of this section.

The three components are:

a) cable Receptacles and Backshells;
b) vertical Headers; and
c) right Angle Headers.

6.2.11.2.4 Cable receptacles and backshells

Figure 45 and Figure 46 show isometric drawings of the internal Multilane cables and connectors. Refer to SFF-8484 for dimensions and mechanical details.

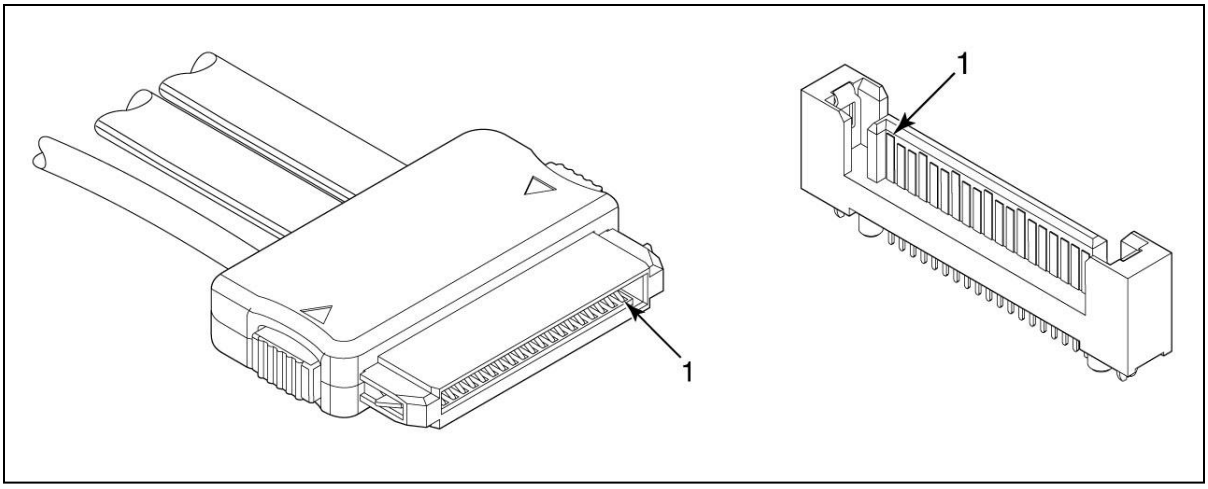


Figure 45 – Isometric drawings of the internal 2 lane cable and connector

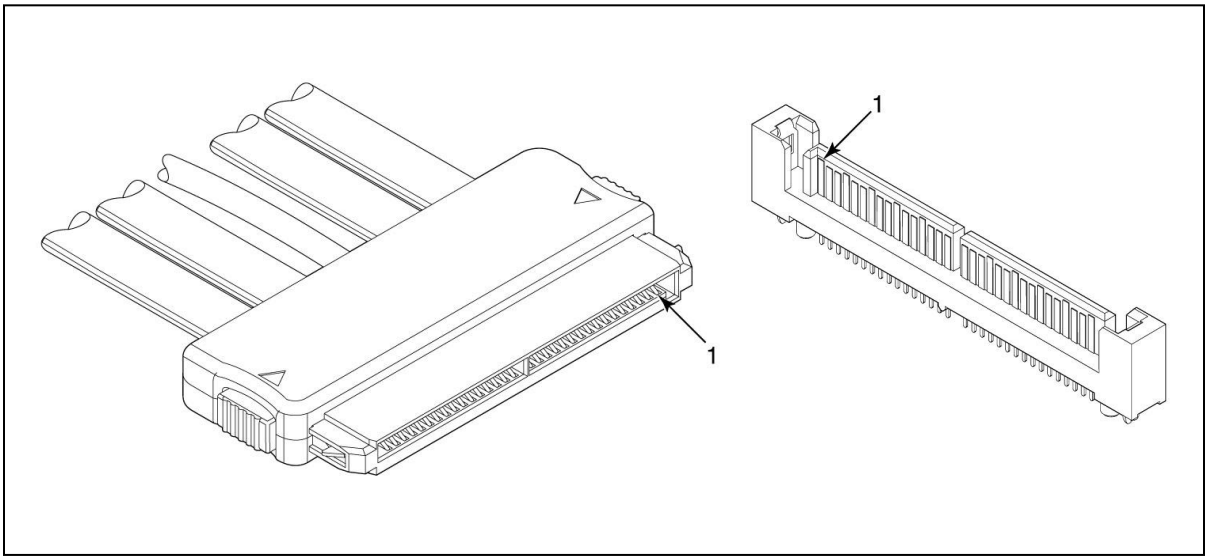
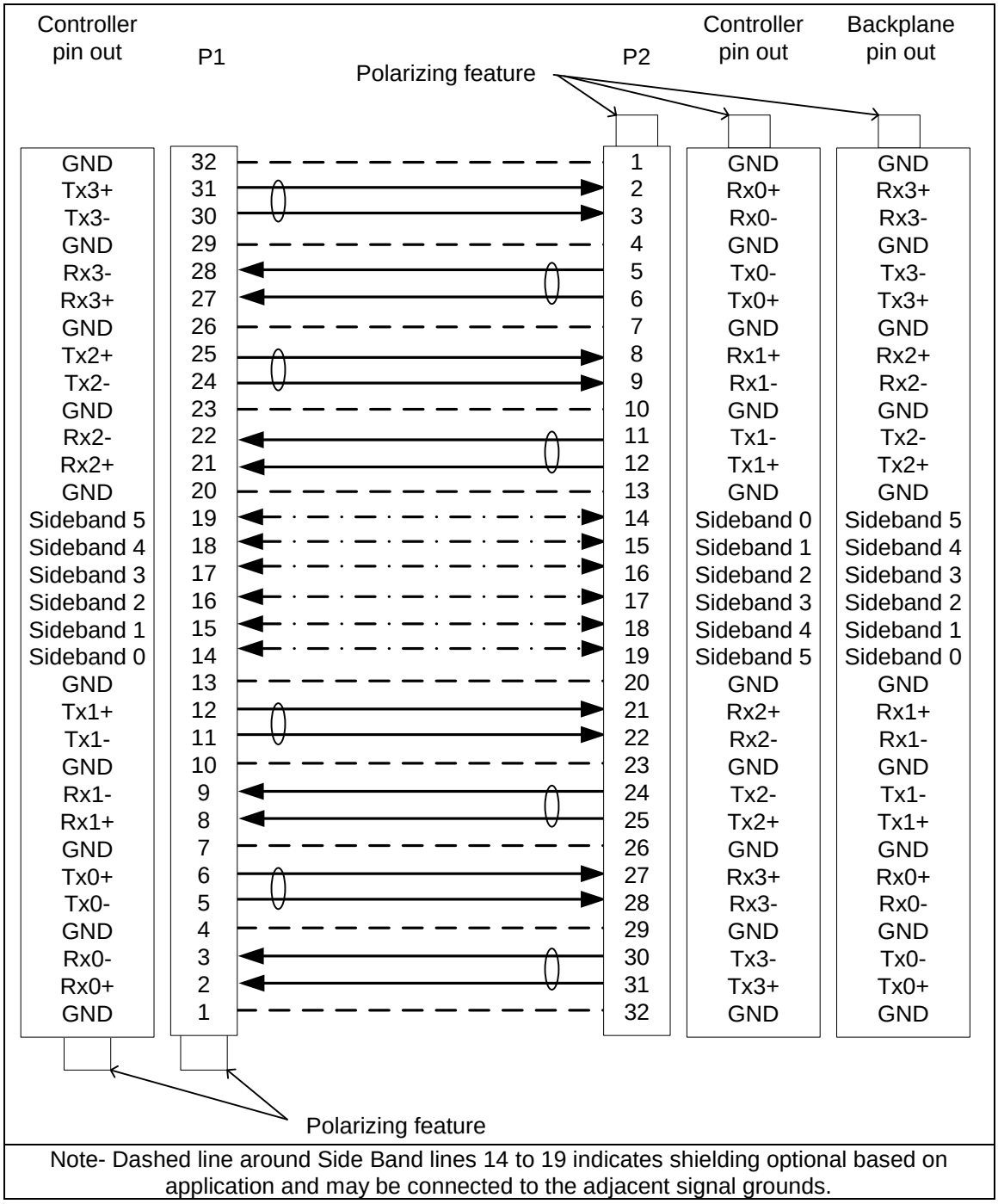


Figure 46 – Isometric drawings of the internal 4 lane cable and connector

6.2.11.3 4 lane pin assignments

The 4 lane pin assignments are shown in Figure 47.



Note- Dashed line around Side Band lines 14 to 19 indicates shielding optional based on application and may be connected to the adjacent signal grounds.

Figure 47 – 4 lane pin assignments

The 4 lane to 4 x 1 lanes, fanout implementation is shown in Figure 48).

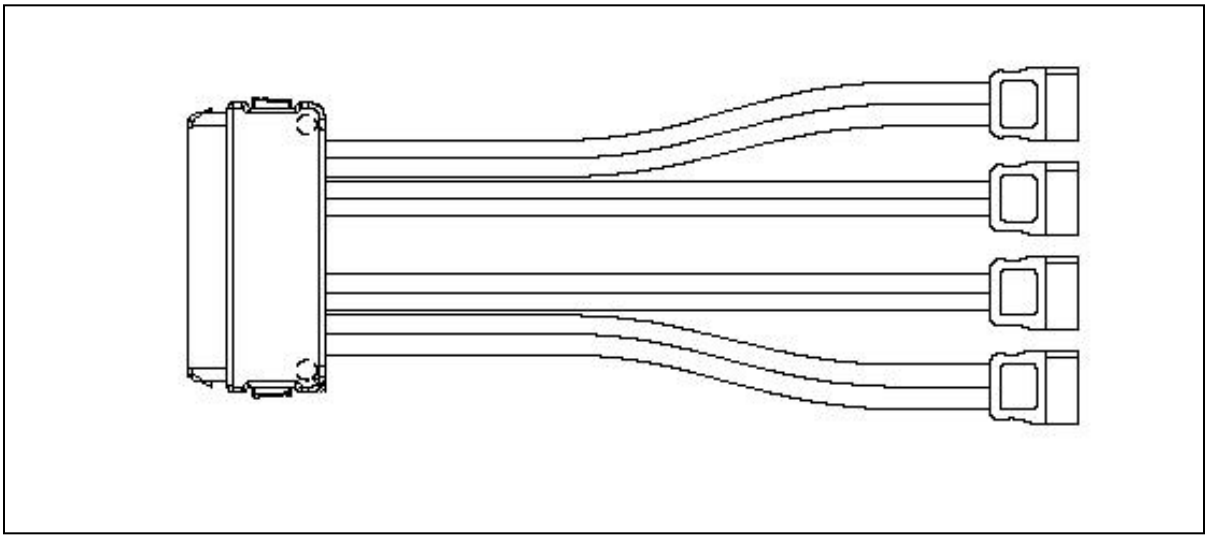


Figure 48 – 4 lane to 4 x 1 lanes, fanout implementation

The 4 lane fanout pin assignments is show in Figure 49.

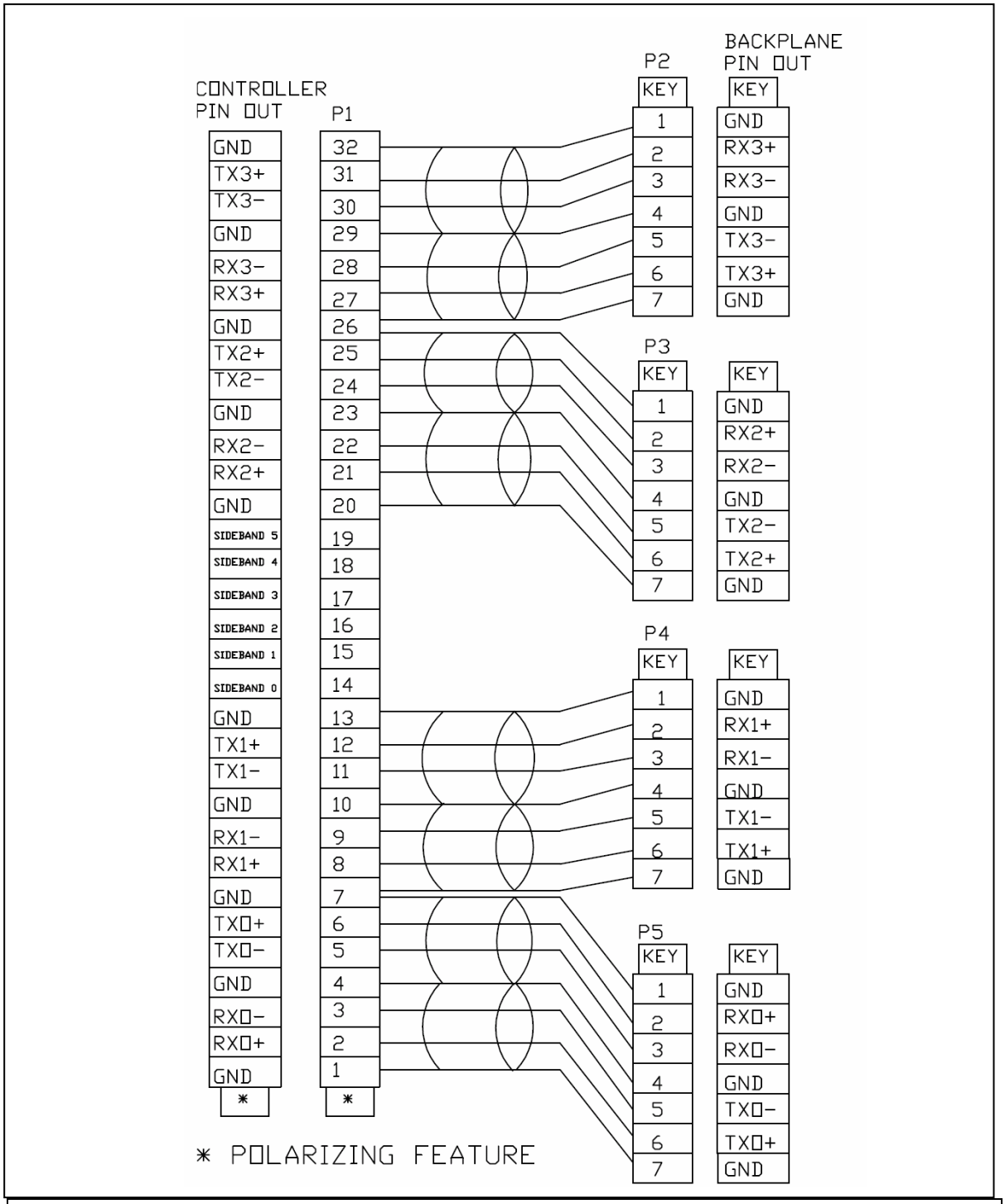


Figure 49 – 4 lane fanout pin assignments

6.2.11.4 2 lane pin assignments

The 2 lane pin assignments are shown in Figure 50.

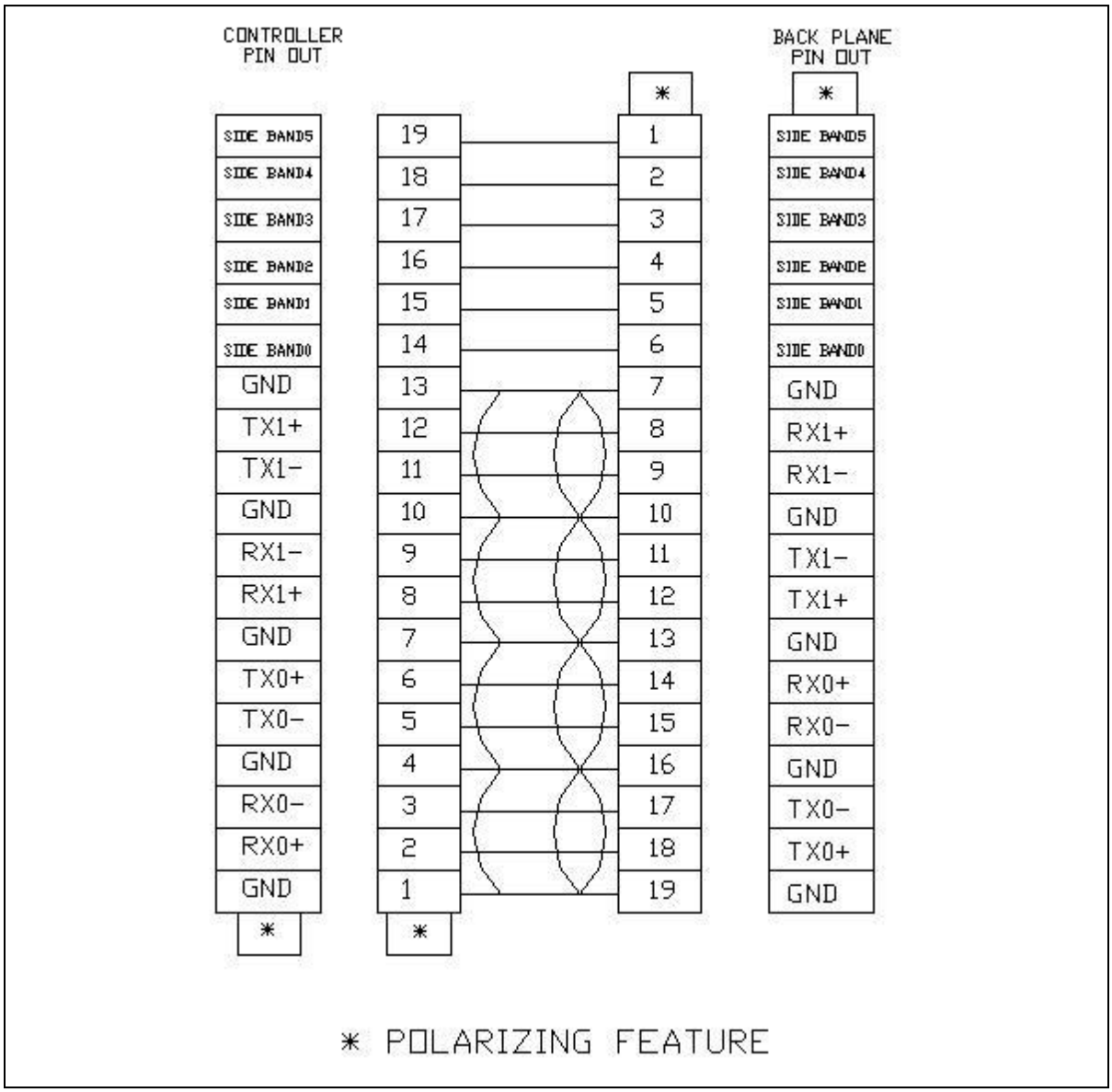


Figure 50 – 2 lane fanout pin assignments

6.2.12 Mini SATA Internal Multilane

6.2.12.1 Mini SATA Internal Multilane overview

This section defines standard cable assemblies and headers for connecting multiple Serial ATA links from a RAID HBA to a backplane within the same enclosure or server, or for connecting multiple Serial ATA links from a HBA to individual devices.

This cable/connector system is based on the SFF-8086 and SFF-8087 specifications. Both SFF-8086 and SFF-8087 specifications are also used by SAS.

6.2.12.2 Conformance criteria

6.2.12.2.1 Conformance criteria list

The conformance criteria is:

a) 4 lanes, Serial ATA signals;
b) cable max length is 1 m;
c) either point to point with a high density connector on both ends of the cable or fanout with a high density connector on one end of the cable assembly and individual single lane connectors on the other end of the cable assembly;
d) Rx, Tx, Rx, Tx pin sequencing to minimize crosstalk;
e) ground reference between each pair;
f) performance supporting 1.5 Gbit/s, 3.0 Gbit/s, and 6.0 Gbit/s; and
g) compliance points are at the ends of a mated cable interface.

If additional interconnect media between host and device exists, that portion of the design is proprietary and shall ensure the mated cable interface compliance points are met.

6.2.12.2.2 Electrical requirements

The Mini SATA Internal Multilane cable assembly shall operate at Gen1i, Gen2i, and Gen3i levels and meet the electrical characteristics defined in Table 36. Since this cable is a Multilane and therefore has multi-aggressors, the additional requirement is to have crosstalk measured using the ML-CXT method. The measured crosstalk shall meet the requirements given in Table 36.

6.2.12.2.3 Component descriptions

Detailed mechanical requirements are specified in SFF-8086 and SFF-8087.

6.2.12.2.4 Mechanical requirements

Figure 51 shows the isometric drawings of the Mini SATA Internal Multilane cables and connectors. Refer to SFF-8086 and SFF-8087 for dimensions and mechanical details.

The Mini SATA Internal Multilane cables and connectors shall use the 36-circuit version plug and receptacle defined in SFF-8086 and SFF-8087.

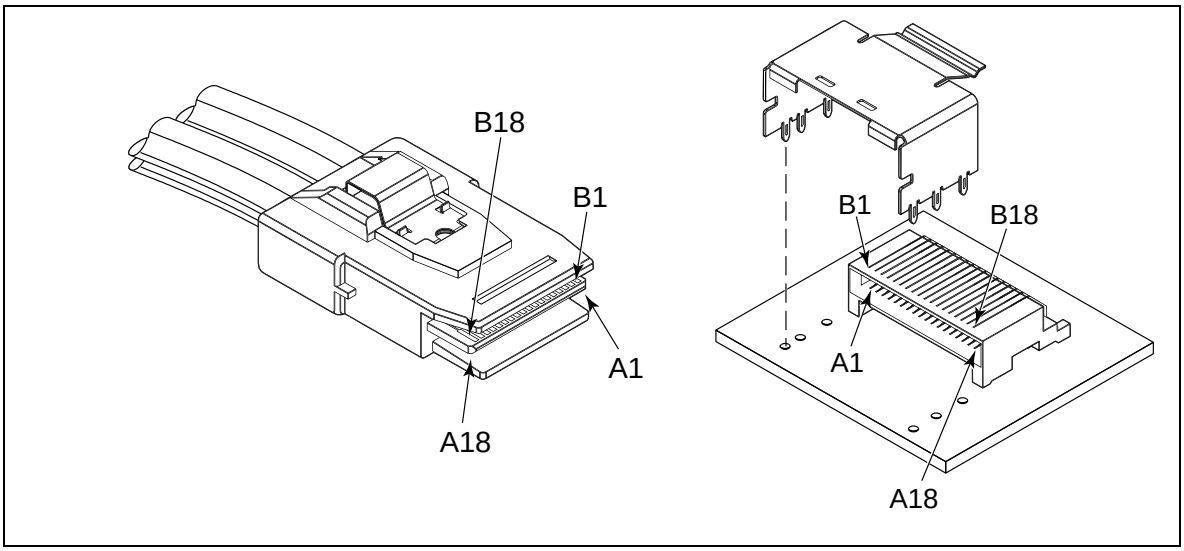


Figure 51 – Isometric drawings for Mini SATA Internal Multilane

6.2.12.3 Mini SATA Internal Multilane pin assignments

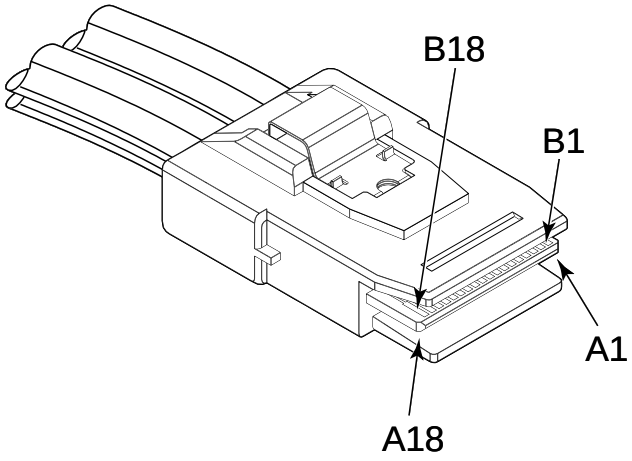
The Mini SATA Internal Multilane connector pin assignments are shown in Figure 52.

Pin assignments for sideband signals are based on the Internal Symmetrical Cable Assembly Implementation shown in Figure 53.

For host-to-backplane applications, sideband signals on the host are attached to the corresponding sideband signals on the backplane (e.g., SB0 of the host is attached to SB0 of the backplane). For host-to-host applications, sideband signals on one host are not attached to their corresponding sideband signals on the other host (e.g., SB0 of one host is attached to SB7 of the other host).

Figure 54 shows the Controller based fanout cable assembly.

Figure 55 shows the Backplane based fanout cable assembly



Signal	Signal pin
SIGNAL GND	A1
Rx 0+	A2
Rx 0-	A3
SIGNAL GND	A4
Rx 1+	A5
Rx 1-	A6
SIGNAL GND	A7
SB7 (host)/SB0 (backplane)	A8
SB3 (host)/SB1 (backplane)	A9
SB4 (host)/SB2 (backplane)	A10
SB5 (host)/SB6 (backplane)	A11
SIGNAL GND	A12
Rx 2+	A13
Rx 2-	A14
SIGNAL GND	A15
Rx 3+	A16
Rx 3-	A17
SIGNAL GND	A18
SIGNAL GND	B1
Tx 0+	B2
Tx 0-	B3
SIGNAL GND	B4
Tx 1+	B5
Tx 1-	B6
SIGNAL GND	B7
SB0 (host)/SB7 (backplane)	B8
SB1 (host)/SB3 (backplane)	B9
SB2 (host)/SB4 (backplane)	B10
SB6 (host)/SB5 (backplane)	B11
SIGNAL GND	B12
Tx 2+	B13
Tx 2-	B14
SIGNAL GND	B15
Tx 3+	B16
Tx 3-	B17
SIGNAL GND	B18

Figure 52 – Mini SATA Internal Multilane connector pin assignments

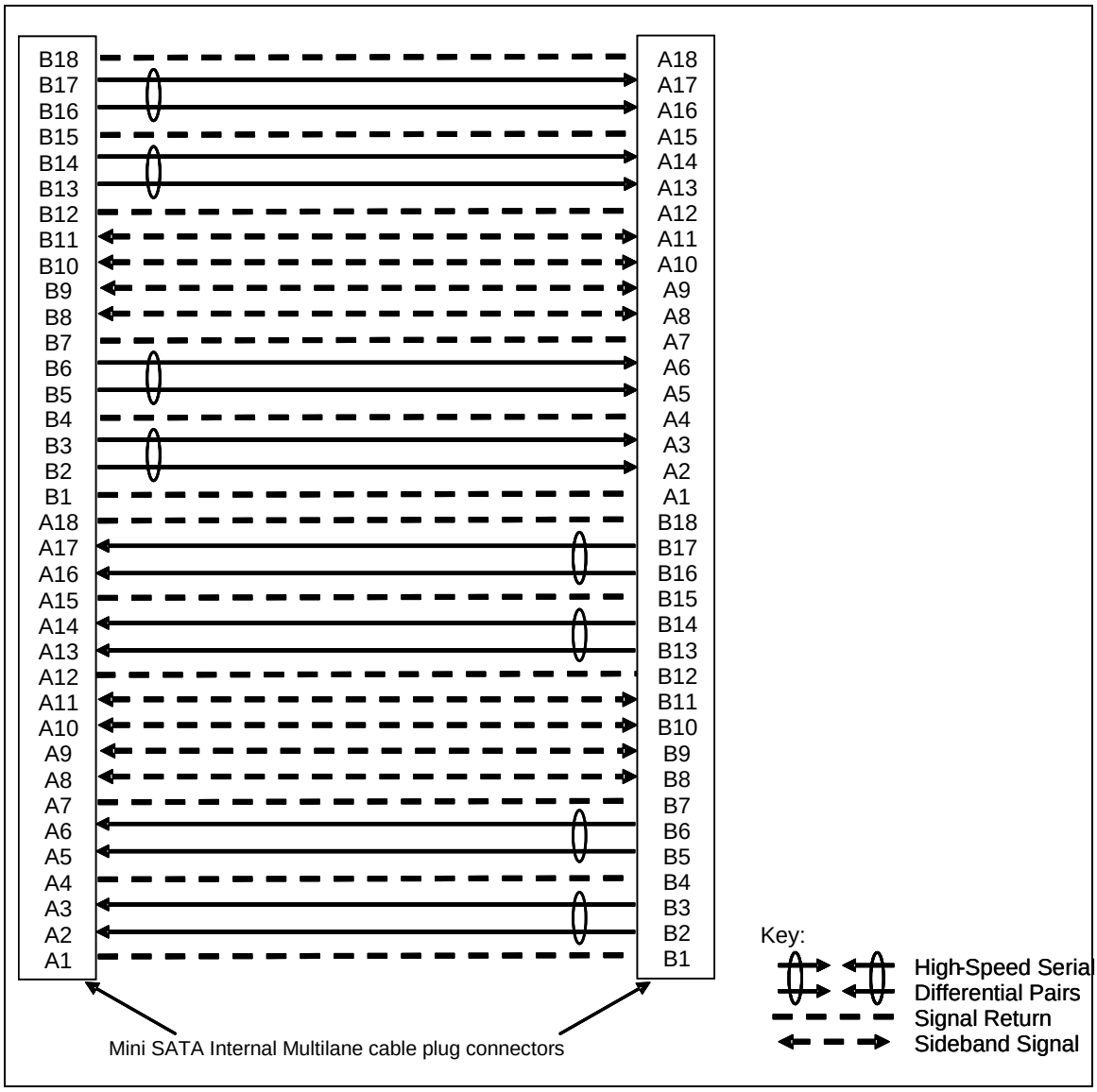
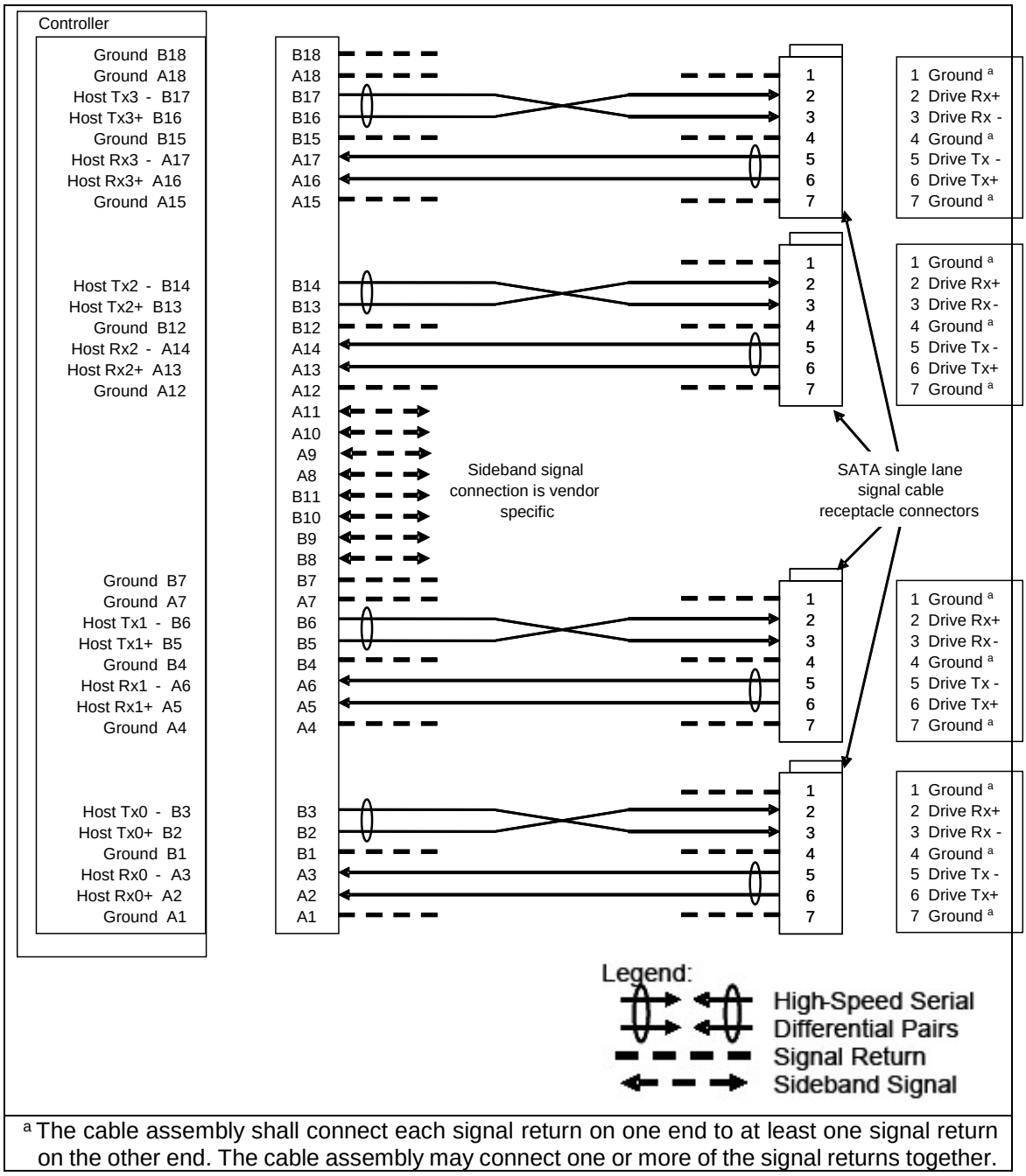
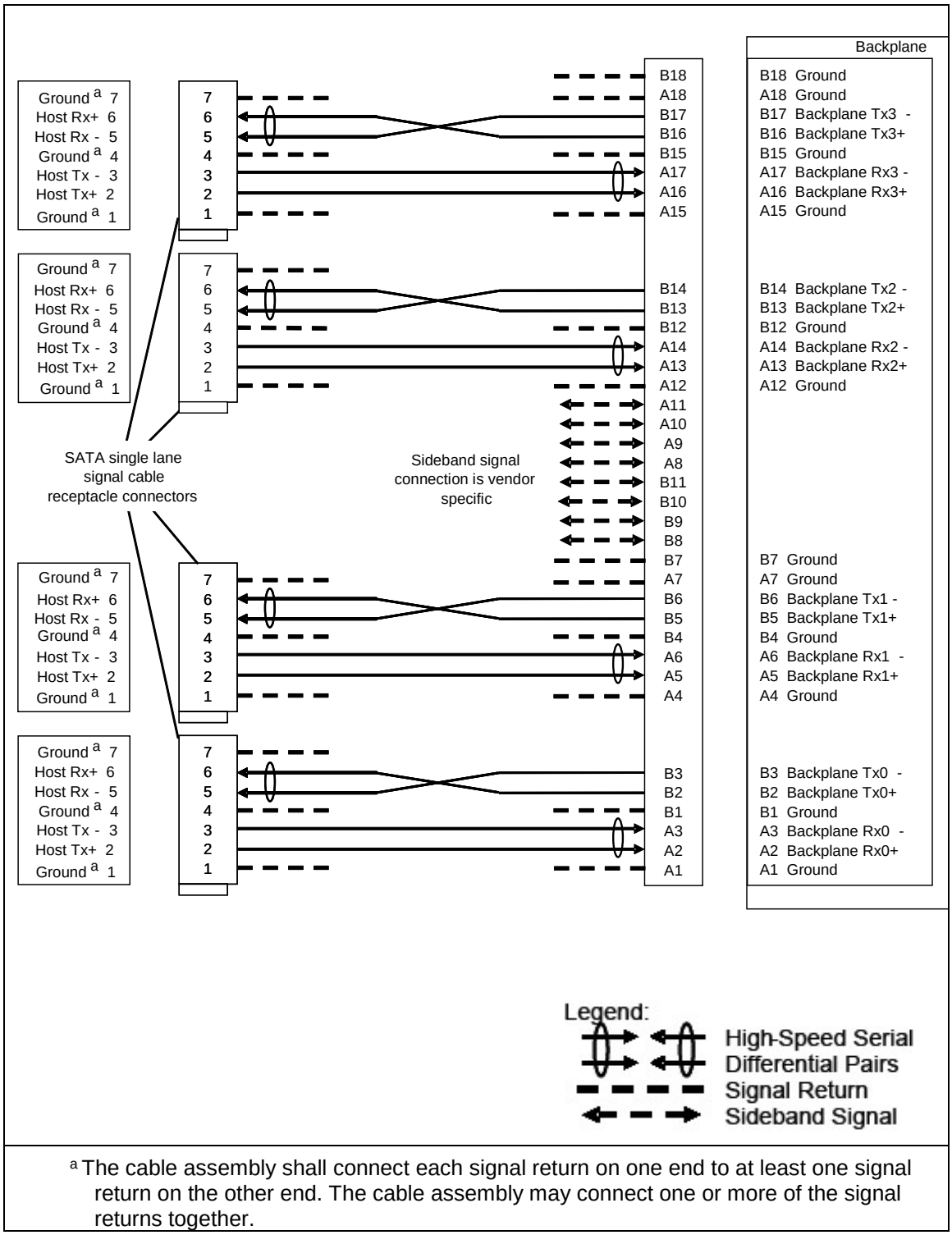


Figure 53 – Mini SATA Internal Multilane system, symmetric cable implementation



[a] The cable assembly shall connect each signal return on one end to at least one signal return on the other end. The cable assembly may connect one or more of the signal returns together.

Figure 54 – Mini SATA Internal Multilane system, controller based fanout cable implementation



[a] The cable assembly shall connect each signal return on one end to at least one signal return on the other end. The cable assembly may connect one or more of the signal returns together.

Figure 55 – Mini SATA Internal Multilane system, backplane based fanout cable implementation

6.3 Internal Micro SATA connector for 1.8 inch HDD

6.3.1 Internal Micro SATA connector for 1.8 inch HDD overview

This section provides capabilities required to enable a smaller Serial ATA 1.8 inch hard disk drive (HDD).

The definition supports the following capabilities:

a) supports Gen1 (1.5 Gbit/s), Gen2 (3.0 Gbit/s), and Gen3 (6.0 Gbit/s) transfer rates;
b) support for backplane (i.e., direct connection) and cable attachment usage models;
c) support for hot plug in backplane (non-cabled) applications;
d) support for 8.0 mm and 5.0 mm slim 1.8 inch form factor HDDs;
e) support of 3.3 V with 5 V to meet future product requirements; and
f) support optional pins, P8 and P9 for vendor specific use.

6.3.2 Usage model

The internal Micro SATA connector may be used for the mobile usage model as defined in 5.3.11. The definition only supports internal 8.0 mm and 5.0 mm slim 1.8 inch form factor HDDs.

6.3.3 General description

The internal Micro SATA connector is designed to enable connection of a slim 1.8 inch form factor HDD to the Serial ATA interface.

The internal Micro Serial ATA connector uses the 1.27 mm pitch configuration for both the signal and power segments. The signal segment has the same configuration as the internal standard Serial ATA connector. The power segment provides the present voltage requirement support of 3.3 V, and includes a provision for a future voltage requirement of 5 V. In addition, there is a reserved pin, P7. Finally, there are two optional pins, P8 and P9, for vendor specific use.

The internal Micro SATA connector is designed with staggered pins, for hot plug backplane (non-cabled) applications.

A special power segment key is located between pins P7 and P8. This feature prevents insertion of other Serial ATA cables.

Care should be taken in the application of this device so that excessive stress is not exerted on the device or connector. Backplane configurations should pay particular attention so that the device and connector are not damaged due to excessive misalignment.

6.3.4 Connector location

The internal Micro SATA connector location on the HDD is shown in Figure 56 and Figure 57 for reference purposes. See SFF-8144 for form factor definition and connector location.

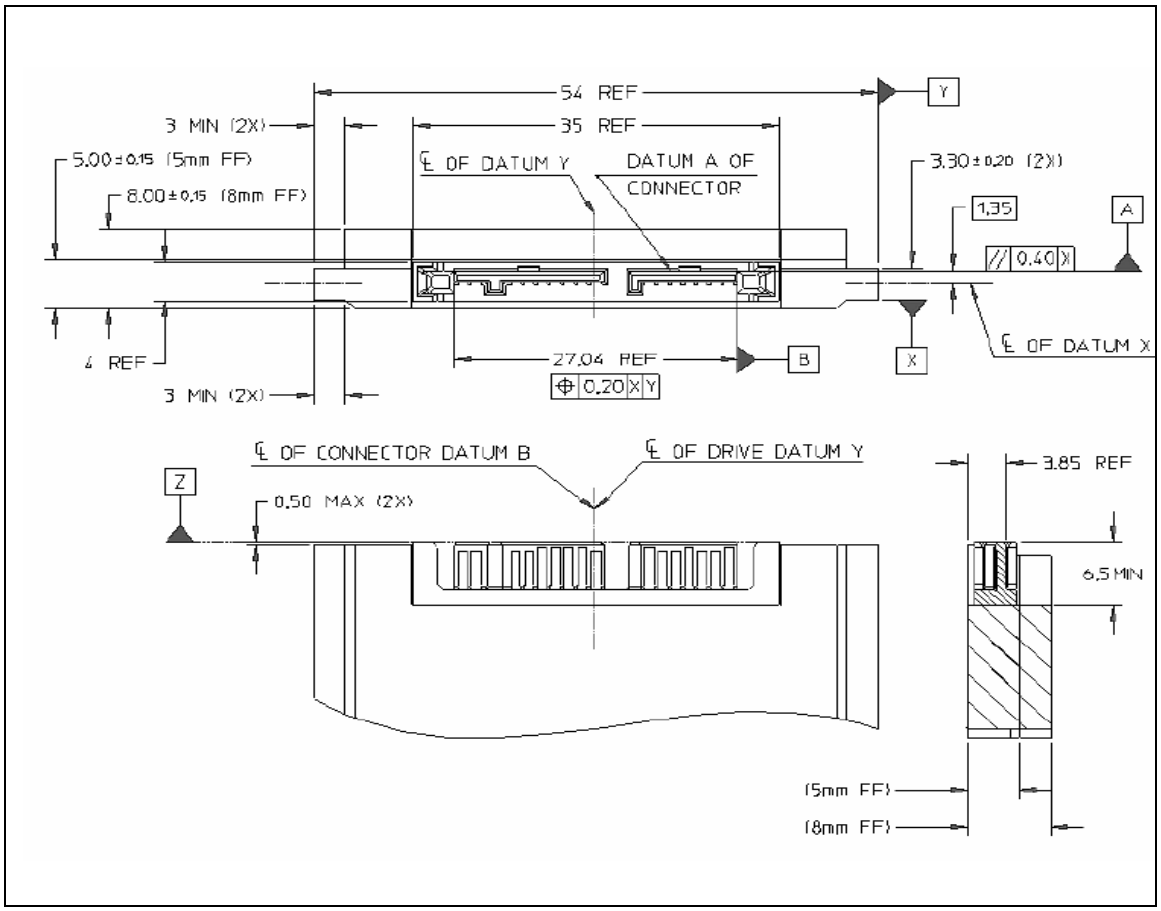


Figure 56 – Device internal Micro SATA connector location for 1.8 inch HDD

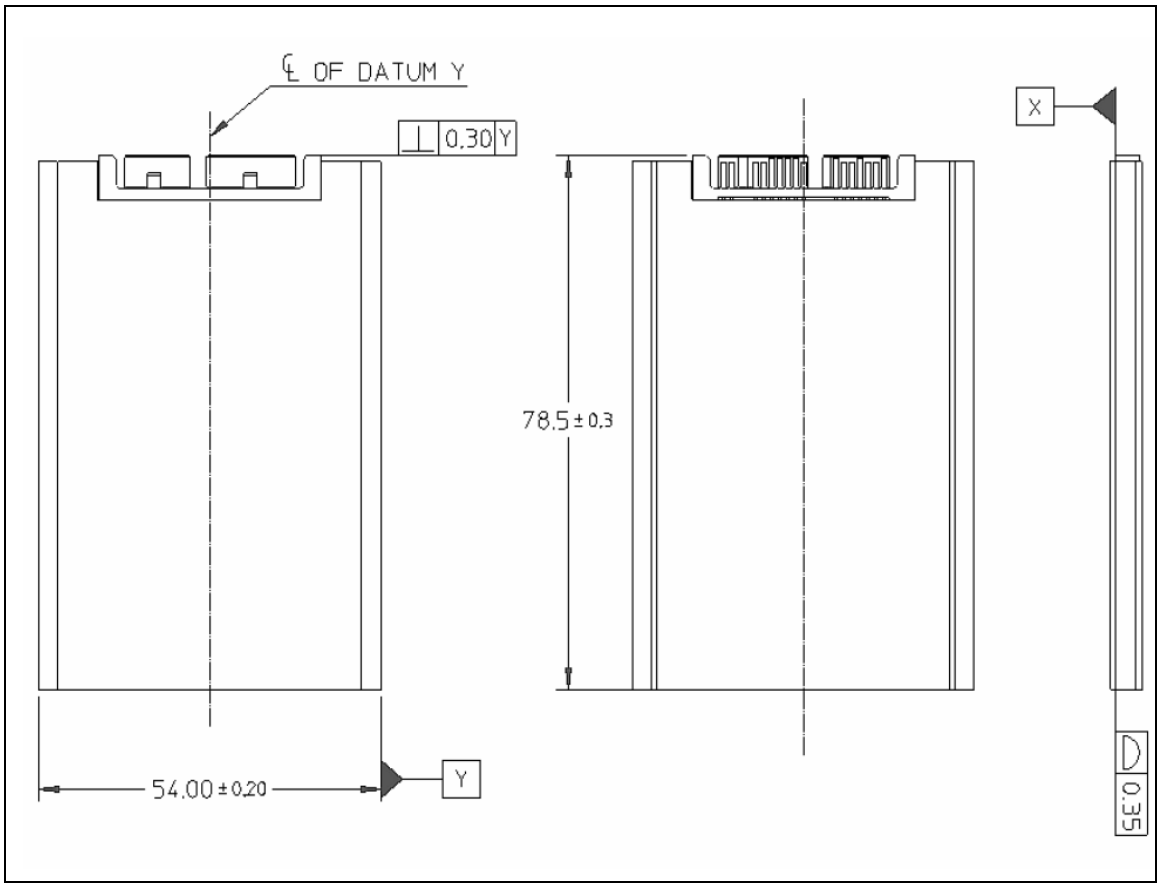


Figure 57 – Device internal Micro SATA connector location for 1.8 inch HDD

6.3.5 Mating interfaces

6.3.5.1 Device internal Micro SATA plug connector

Figure 58 defines the interface dimensions for the internal Micro SATA device plug connector with both signal and power segments.

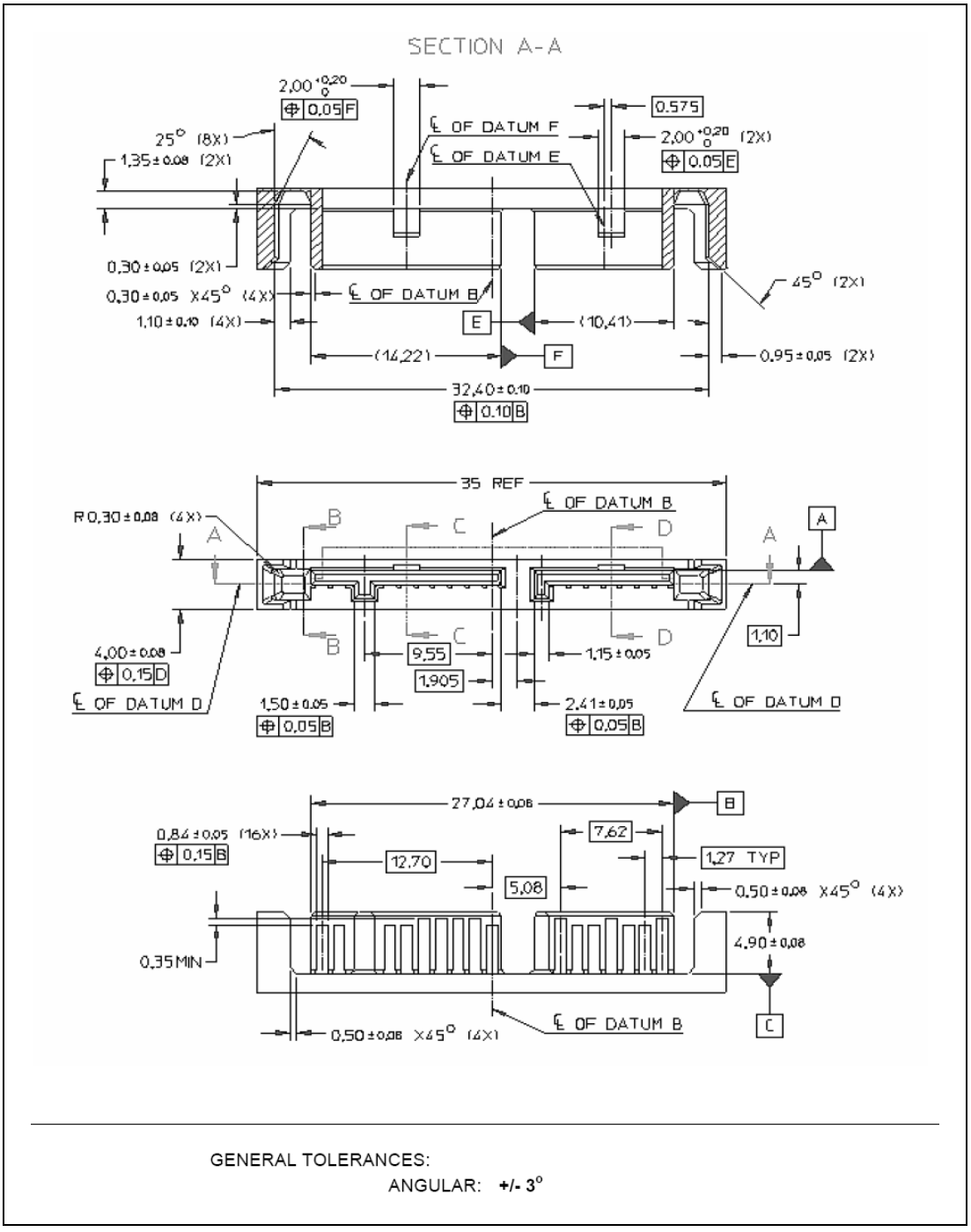


Figure 58 – Device internal Micro SATA plug connector (part 1 of 2)

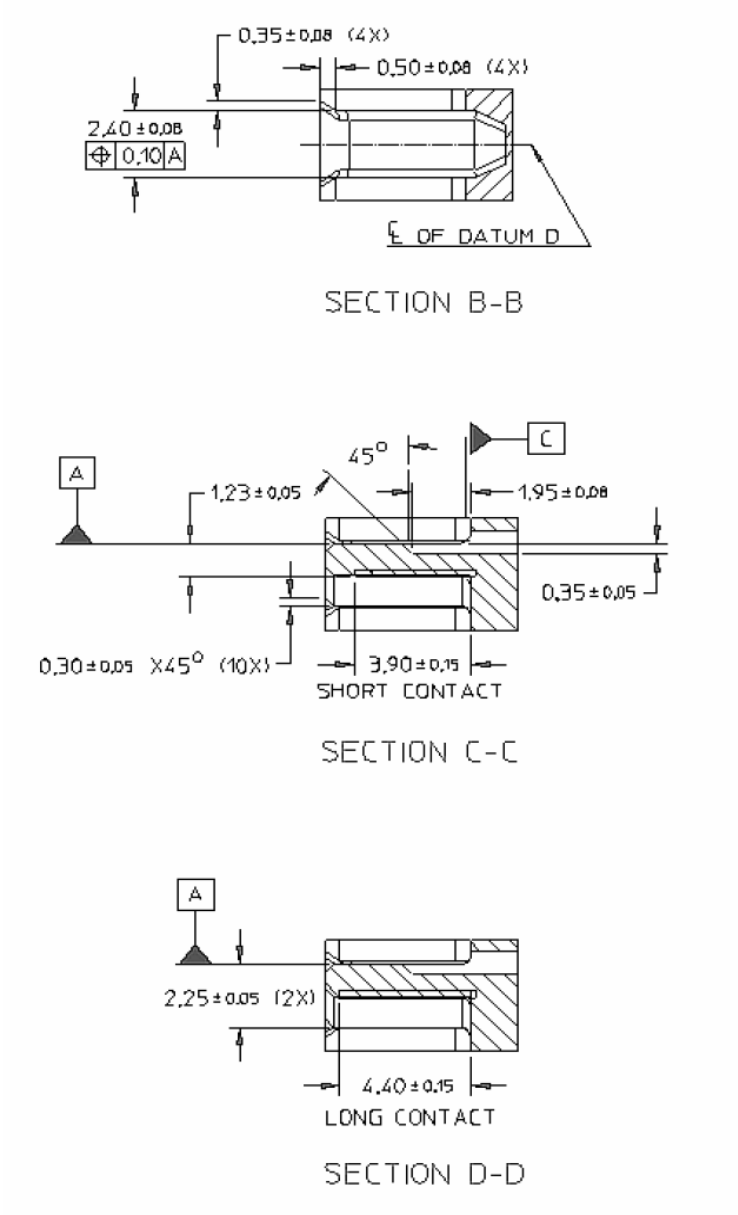


Figure 58 – Device internal Micro SATA plug connector (part 2 of 2)

6.3.5.2 Internal Micro SATA backplane connector

Figure 59 defines the interface dimensions for the internal Micro SATA backplane connector.

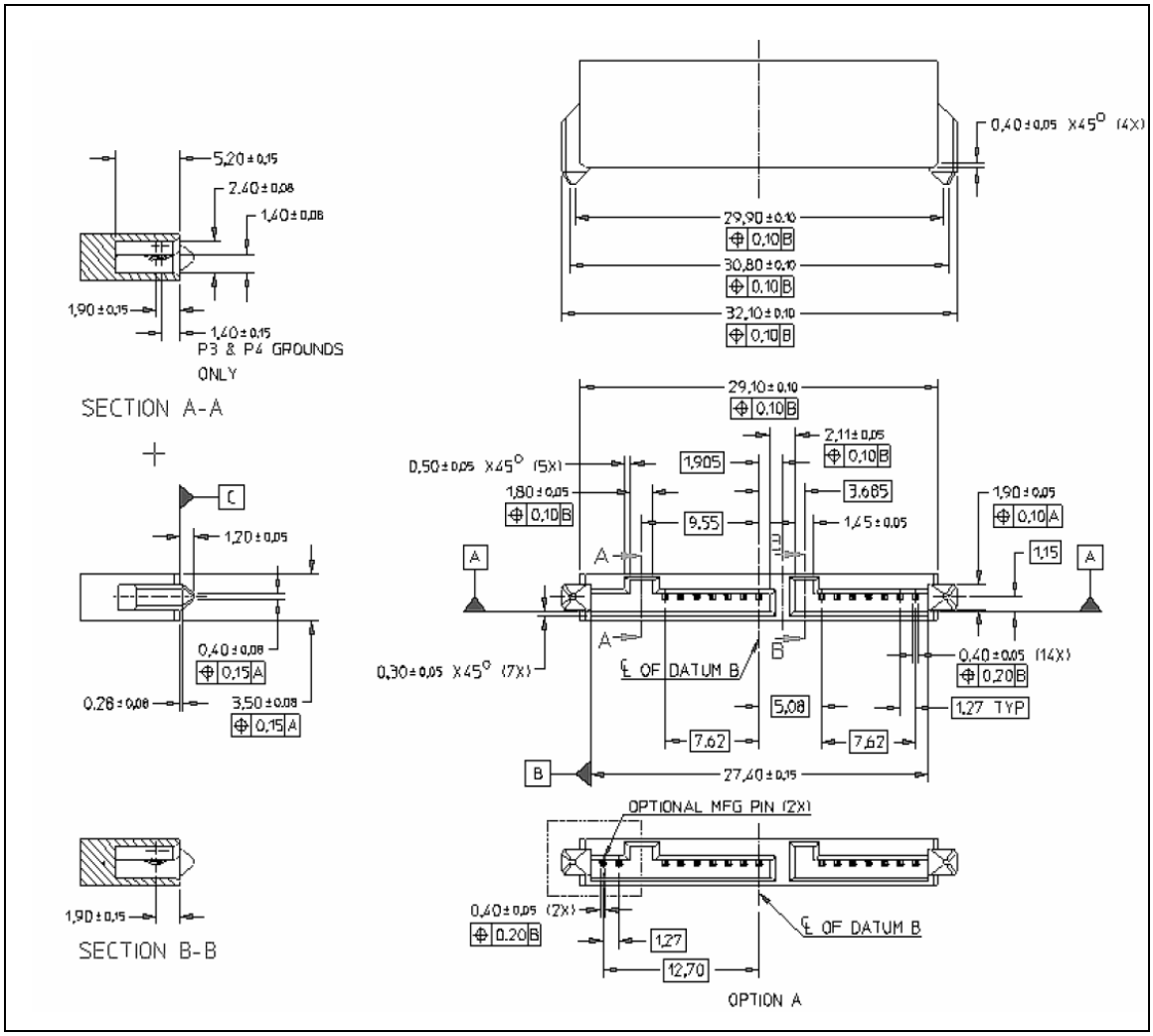


Figure 59 – Internal Micro SATA backplane connector

6.3.5.3 Internal Micro SATA power receptacle connector

Figure 60 defines the interface dimensions for the internal Micro SATA power receptacle connector.

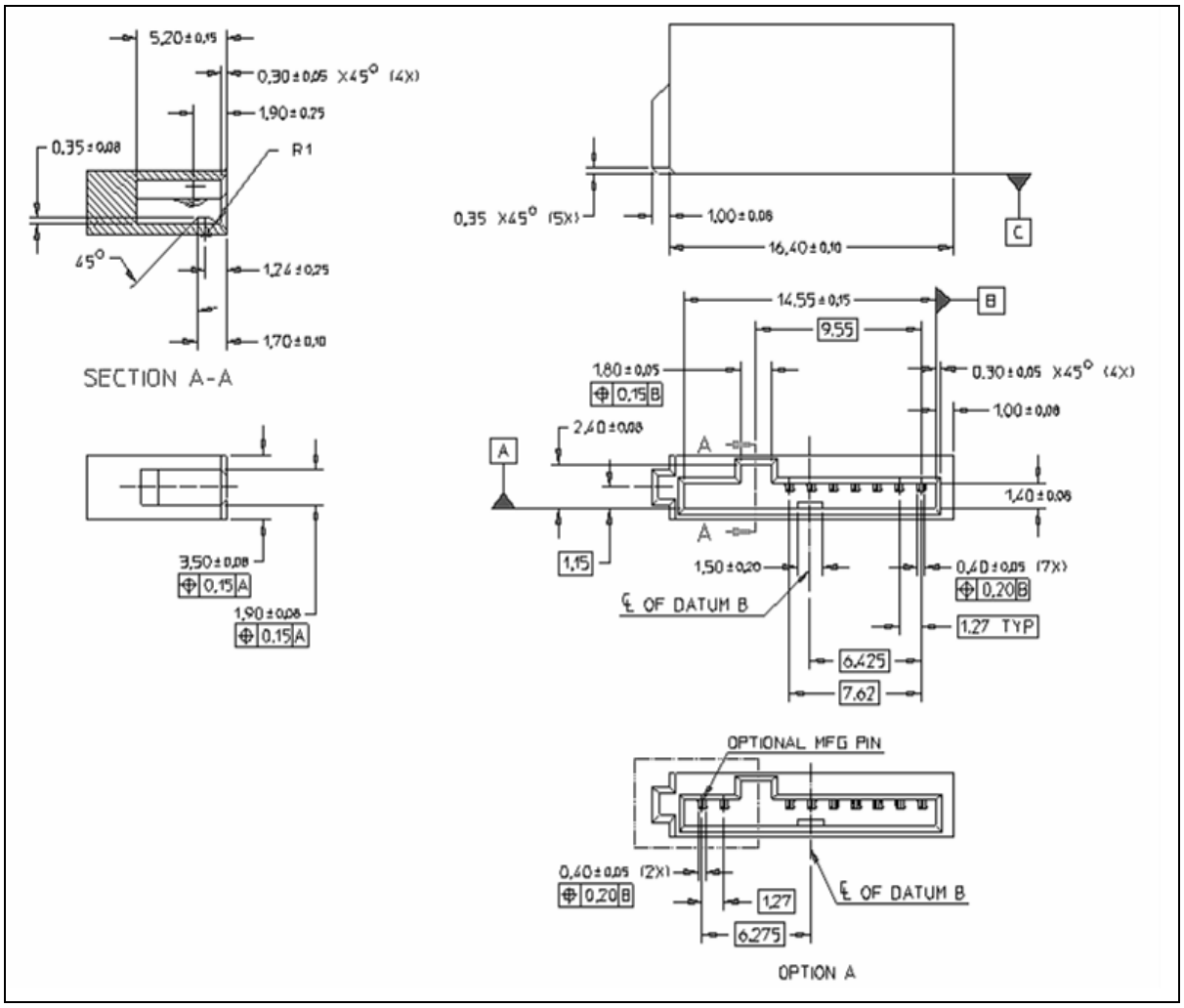


Figure 60 – Internal Micro SATA power receptacle connector

6.3.5.4 Internal Micro SATA connector pair blind-mate misalignment capability

The maximum blind-mate misalignment capabilities are ± 1.50 mm and ± 1.00 mm, respectively, for two perpendicular axes illustrated in Figure 61. Any skew angle of the plug, with respect to the receptacle, reduces the blind-mate capabilities.

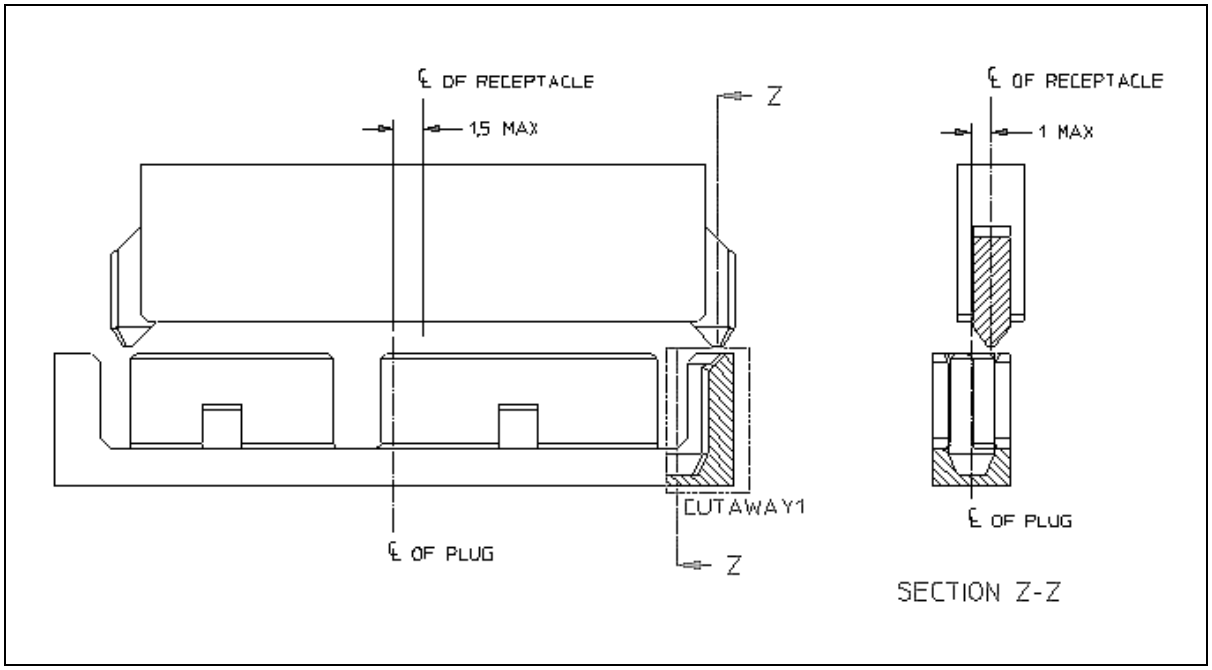


Figure 61 – Internal Micro SATA connector pair blind-mate misalignment capability

6.3.5.5 Internal Micro SATA pin signal definition and contact mating sequence

Table 12 details the pin names, types and contact order of the two internal Micro SATA Plug options. A brief description is also included for signal, ground and power pins. There are total of 7 pins in the signal segment and 9 pins in the power segment.

Table 12 – Signal and power Internal Micro SATA plug and nominal mate sequence

	Name	Type	Description	Cable Usage [a] [b]	Backplane Usage [b]
Signal Segment	Refer to Table 5.				
Spacing separate signal and power segments [c]					
Power Segment	P1	V_{33}	3.3 V Power	2nd Mate	3rd Mate
	P2	V_{33}	3.3 V Power, Pre-charge	1st Mate	2nd Mate
	P3	GND	Ground	1st Mate	1st Mate
	P4	GND	Ground	1st Mate	1st Mate
	P5	V_5	5 V Power, Pre-charge [d]	1st Mate	2nd Mate
	P6	V_5	5 V Power [d]	2nd Mate	3rd Mate
	P7	DAS/DHU	DAS/Direct Head Unload / Vendor Specific [e]	2nd Mate	3rd Mate
	Key	Key	Key	N/C	N/C
	P8	Optional	Vendor specific [f]	2nd Mate	3rd Mate
	P9	Optional	Vendor specific [f]	2nd Mate	3rd Mate

[a] Although the mate order is shown, hot plugging is not supported if using the cable connector receptacle.
[b] All mate sequences assume zero angular offset between connectors.
[c] The signal segment and power segment may be separate.
[d] The 5 V supply voltage pins are included to meet future product requirements and may optionally be provided on the power segment receptacle. Future revisions of this specification may require 5 V supply voltage be provided.
[e] For specific optional usage of pin P7, see 6.13.
[f] No connect on the host side.

6.3.5.6 Internal Micro SATA connector and cable assembly requirements and test procedures

The internal Micro SATA connector and cable shall meet the requirements as defined for standard internal SATA cables and connectors as defined in 6.2.10, with the exceptions given in Table 13.

Table 13 – Unique connector mechanical testing procedures and requirements

Parameter	Procedure	Requirement
Removal force Backplane connector	EIA-364-13 Measure the force necessary to unmate the connector assemblies at a max. rate of 12.5 millimeter per minute.	Min 2.5 N after 500 cycles
Insertion force Cabled power connector (non-latching)	EIA-364-13 Measure the force necessary to mate the connector assemblies at a max. rate of 12.5 millimeter per minute.	Max 45 N.
Removal force Cabled power connector (non-latching)	EIA-364-13 Measure the force necessary to unmate the connector assemblies at a max. rate of 12.5 millimeter per minute.	Min 10 N for cycles 1 to 5 Min 8 N through 50 cycles

6.4 Internal Slimline cables and connectors

6.4.1 Internal Slimline cables and connectors overview

This section provides capabilities required to enable Serial ATA in “Slimline” optical disk drives.

The definition supports the following capabilities:

a) supports Gen1 (1.5 Gbit/s), Gen2 (3.0 Gbit/s), and Gen3 (6.0 Gbit/s) transfer rates;
b) support for backplane (i.e., direct connection) and cable attachment usage models;
c) support for warm plug;
d) support for 12.7 mm, 9.5 mm, 8.5 mm, and 7.0 mm Slimline devices; and
e) 5 V only power delivery.

The definition has the following constraints:

a) no Device Activity Signal (DAS) support;
b) no support for DevSleep interface power state;
c) analog audio is not supported;
d) external cable and connector are not supported;
e) no hot plug support; and
f) warm plug support is usage model dependent.

6.4.2 Usage models

Support for three usage models. Internal fixed bay (i.e., direct connection), removable bay, and internal cable. The requirements for each usage model are specified in this section.

Internal Fixed Bay requirements:

a) support for 12.7 mm, 9.5 mm, 8.5 mm, and 7.0 mm Slimline devices;
b) direct connection support for 1.8 inch HDD not required;
c) direct connection support for 2.5 inch HDD not required;
d) minimum footprint; and
e) device presence detection not required.

Removable Bay requirements:

a) support for 12.7 mm, 9.5 mm, 8.5 mm, and 7.0 mm Slimline devices;
b) direct connection support for 1.8 inch HDD not required;

c) direct connection support for 2.5 inch HDD not required. Attachment in carrier adapter shall conform to cabled usage model for Gen1i, Gen2i, and Gen3i;
d) warm plug and blind mate required;
e) un-powered device presence detection required;
f) no floppy support required;
g) no battery connector support required;
h) support 500 insertions and removals (Design is scalable to higher cycle counts); and
i) un-powered device attention required.

Internal Cable requirements:

a) support for 12.7 mm, 9.5 mm, 8.5 mm, and 7.0 mm Slimline devices;
b) direct connection support for 1.8 inch HDD not required;
c) direct connection support for 2.5 inch HDD not required;
d) blind mate not supported;
e) device presence detection not supported;
f) support passive and latching retention mechanisms;
g) support internal single lane cables and connectors, as defined in 6.2; and
h) support 50 insertions and removals.

6.4.3 General description

The Slimline connector is designed to enable connection of a Slimline form factor device to the Serial ATA interface. The connector design accommodates a latching option for 8.5 mm, 9.5 mm, and 12.7 mm Slimline devices. A latching option may not be accommodated in 7.0 mm Slimline devices. This internal Serial ATA connector uses the 5 mm (ref) connector height as low profile.

The connector is designed to fit into the presently used Parallel ATA (PATA) connector Slimline designs with almost no changes to the Slimline device case, media tray, or internal mechanics. It is anticipated that a simple replacement of the PATA PC board with a SATA controller and connector in the present Slimline device designs may be possible.

The connector preserves the design of the signal portion of the present SATA connector and accommodates presently available SATA signal cables.

The power portion of the connector was reduced to six pins. Both +12 V and +3.3 V were removed from the connector leaving +5 V as the sole supply voltage.

The standard +A, -A, +B, and –B signals are on the signal portion of the connector.

In addition to +5 V and ground (GND), the following signals were added to the power portion:

a) Device Present (DP), active low signal indicating device connect to the host. The device shall connect the DP pin to ground with a resistance of 1 kohm with a relative tolerance of ± 10 %. This signal is not supported in a cabled environment. If a cabled connection is used and the cable is held in a fixed position, the cable shall follow backplane requirements. Host connection to the DP pin is optional. If un-used, a connection is not allowed. If the host requires the use of the DP function, the maximum current it shall source is 4 mA and the minimum is 0 mA; and
b) Manufacturing Diagnostic/Device Attention (MD/DA). In a manufacturing environment this signal pin is used by device vendors during device testing. In a PC application environment (e.g., a shipping system) it is used to signal that the device requires attention from the host.

Device implementation of Device Attention is optional and if supported, the device shall set bit 4 in Word 77 in the IDENTIFY PACKET DEVICE data structure to inform the host it supports this capability (see 13.2.3.15).

The maximum voltage applied to the pin from the host shall be 3.63 V. The maximum current the host shall source at any time is 4 mA, and the minimum is 0 mA. If the feature is unused by the host, then host connection of this pin is optional. If the host connects to MD/DA the host shall apply a voltage bias to the pin within the maximum and minimum values defined in this section. This voltage bias allows the device to detect whether the device is connected to a PC application environment, or a manufacturing test environment.

If MD/DA is pulled to ground by the host or if no voltage is applied by the host at the time 5 V is applied to the device, the device may assume that the device is in a manufacturing test environment and may apply the appropriate signaling to the pin necessary for the manufacturing test environment. In no case shall the device apply more than 3.3 V, with a relative tolerance of ± 10 %) on the pin. If IDENTIFY PACKET DEVICE bit 5 Word 77 is cleared to zero, the device may apply manufacturing diagnostic signaling on MD/DA at any time.

If MD/DA is pulled up to 3.3 V, with a relative tolerance of ± 10 %, at the time 5 V is applied to the device, or if 5 V and 3.3 V are applied within 100 ms of each (e.g., if the device is connected into a powered system) the device is in a PC application environment. In a PC application environment, the device shall not apply signaling intended for manufacturing diagnostic purposes.

In the PC application environment MD/DA is asserted on a low to high transition and indicates that the device requires attention by host SW or hardware. If the signal is low, the device shall present no more than 1 kohm, with a relative tolerance of ± 10 %, effective resistance to the host. If the signal is high, the device shall present a resistance of no less than 100 kohm, with a relative tolerance of ± 10 %.

If Device Attention function capability is desired by the host, the host shall ensure that MD/DA is pulled up to 3.3 V, with a relative tolerance of ± 10 %, before applying 5 V during the power up sequence to properly configure the device.

When in a PC application environment, if the device does not set the Supports Device Attentionon Slimline connected device bit (i.e., IDENTIFY PACKET DEVICE Word 77 bit 4), the device may assert MD/DA. Tray load optical devices shall assert Device Attention if the front panel button is released. Slot load optical devices shall assert Device Attention for at least 10 ms if the media is inserted. A Slot load optical device may optionally assert Device Attention as defined above if the media is ejected.

6.4.4 Connector location and keep out zones

6.4.4.1 Connector location and keep out zones overview

This section describes the location of the connector in the Slimline device referenced from the standard locations. Keep out zones are also defined to allow blind mate capability.

6.4.4.2 Location

Table 14 shows the connector location references for the 7.0 mm, 8.5 mm, 9.5 mm, and 12.7 mm Slimline devices.

Table 14 – Slimline connector location references

Slimline Drive	Horizontal	Vertical	Depth
7.0 mm Slimline	Drive left edge to connector CL	Drive bottom edge to connector tongue top edge	Drive back edge to connector back wall
8.5 mm Slimline	Drive left edge to connector CL	Drive bottom edge to connector tongue top edge	Drive back edge to connector back wall
9.5 mm Slimline	Drive left edge to connector CL	Drive bottom edge to connector tongue top edge	Drive back edge to connector back wall
12.7 mm Slimline	Drive left edge to connector CL	Drive bottom edge to connector tongue top edge	Drive back edge to connector back wall

6.4.4.3 Keep out zones

6.4.4.3.1 7.0 mm Slimline drive

The minimum panel opening is the maximum connector size plus the positional tolerance of the connector location within the Slimline device.

The internal SATA connector location on the 7.0 mm Slimline optical disk drive (ODD) is shown in Figure 62 for reference purposes only. Additional details may be found in SFF-8553.

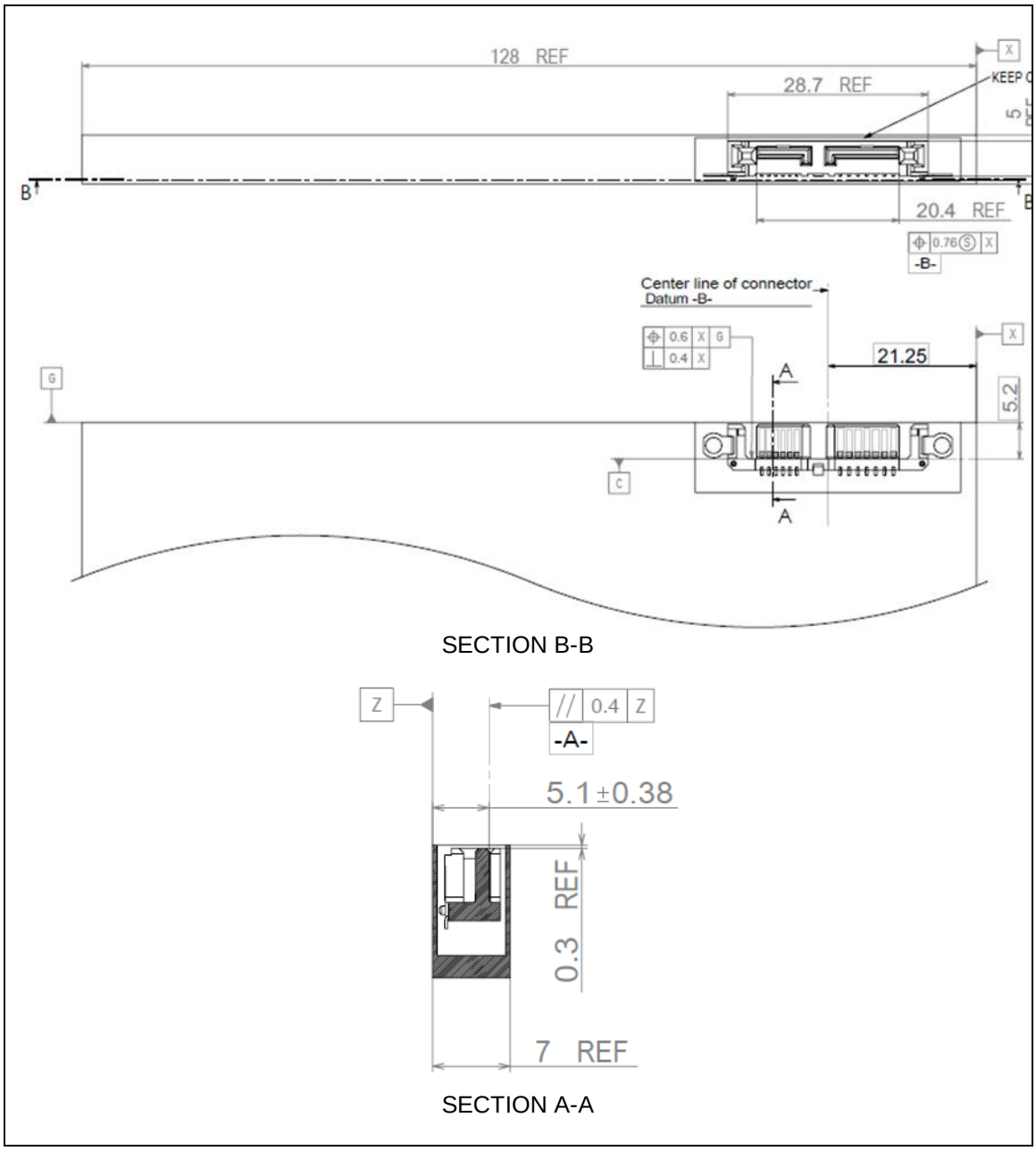


Figure 62 – 7.0 mm Slimline drive connector locations (informative)

6.4.4.3.2 8.5 mm Slimline drive

The internal SATA connector location on the 8.5 mm Slimline ODD is shown in Figure 63 for reference purposes only. Additional details may be found in SFF-8553.

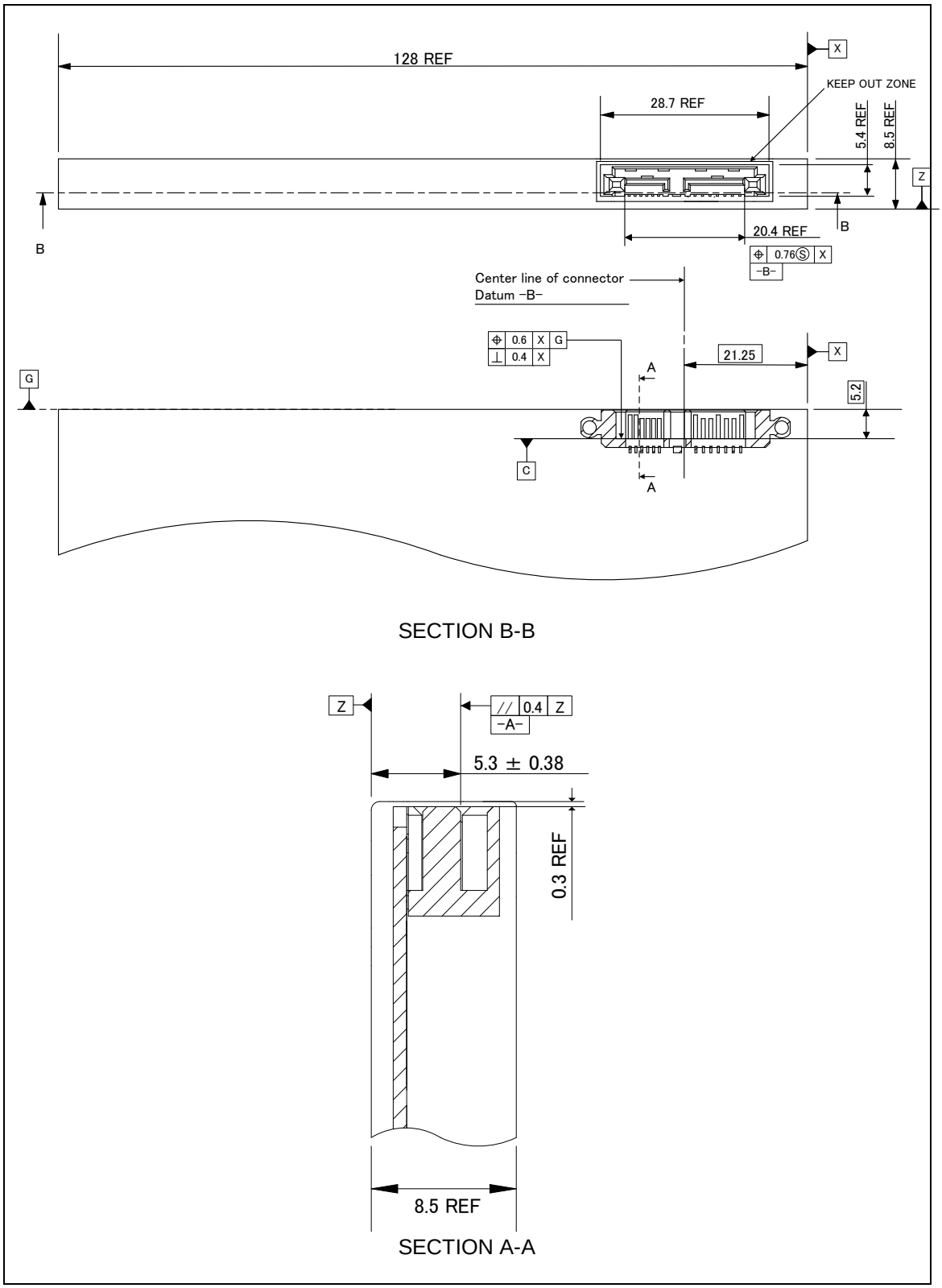


Figure 63 – 8.5 mm Slimline drive connector locations (informative)

6.4.4.3.3 9.5 mm and 12.7 mm Slimline drive

The internal SATA connector location on the 9.5 mm and 12.7 mm Slimline ODD is shown in Figure 64 for reference purposes only. Additional details may be found in SFF-8553.

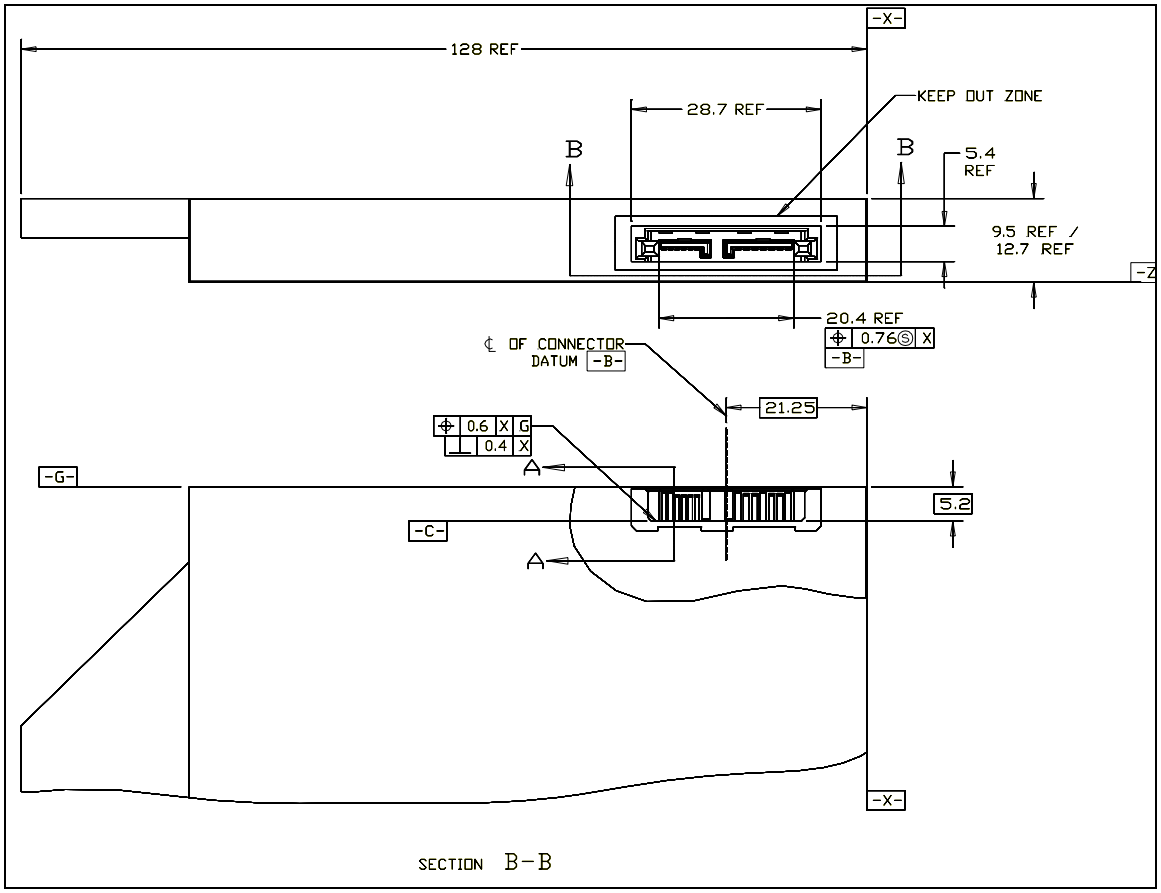


Figure 64 – 9.5 mm/12.7 mm Slimline drive connector locations (informative)

Figure 64 cross-section A for the 9.5 mm Slimline drive is shown in Figure 65. Figure 64 cross-section A for the 12.7 mm Slimline drive is shown in Figure 66.

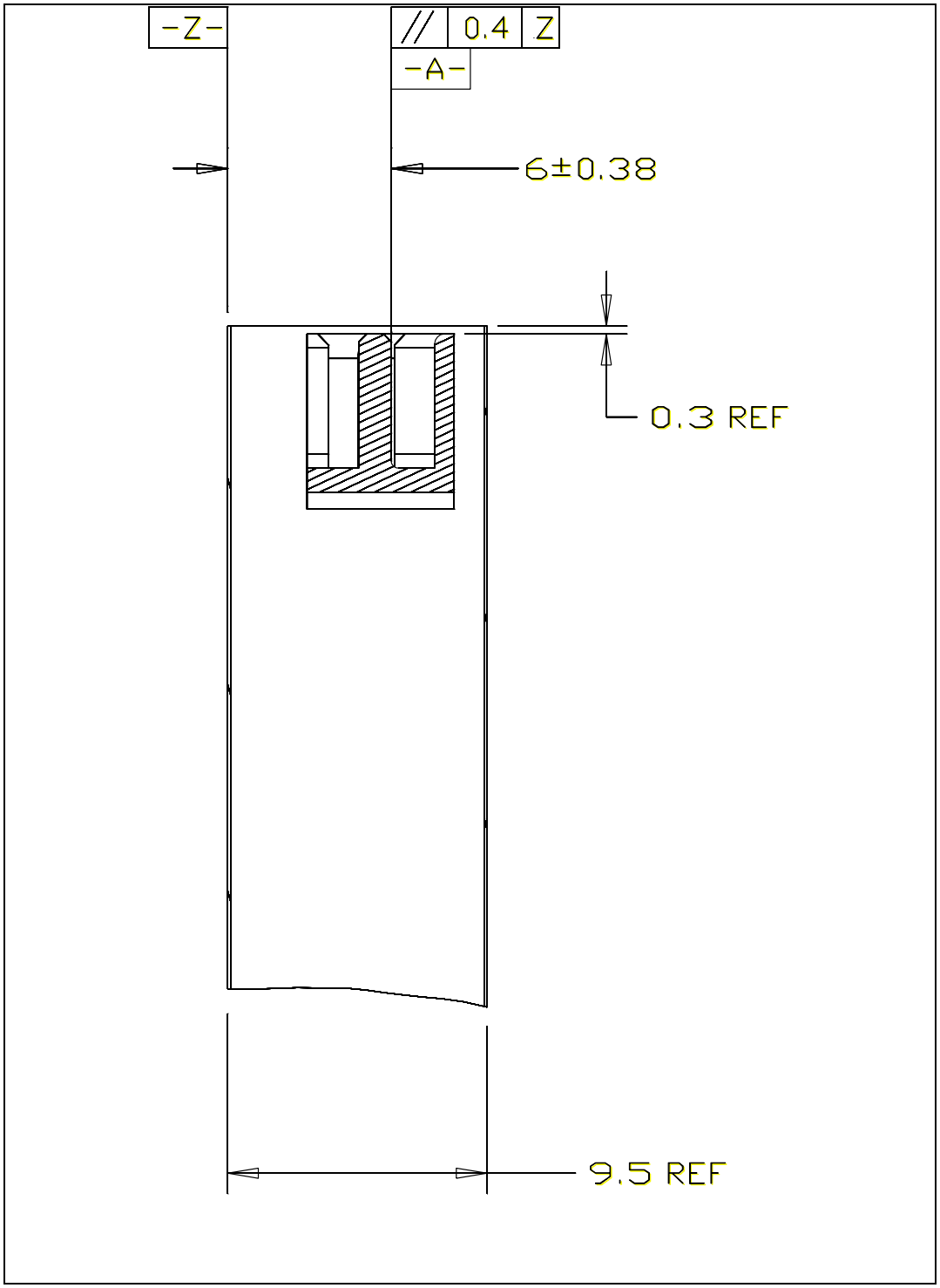


Figure 65 – 9.5 mm Slimline drive connector location (section A-A) (informative)

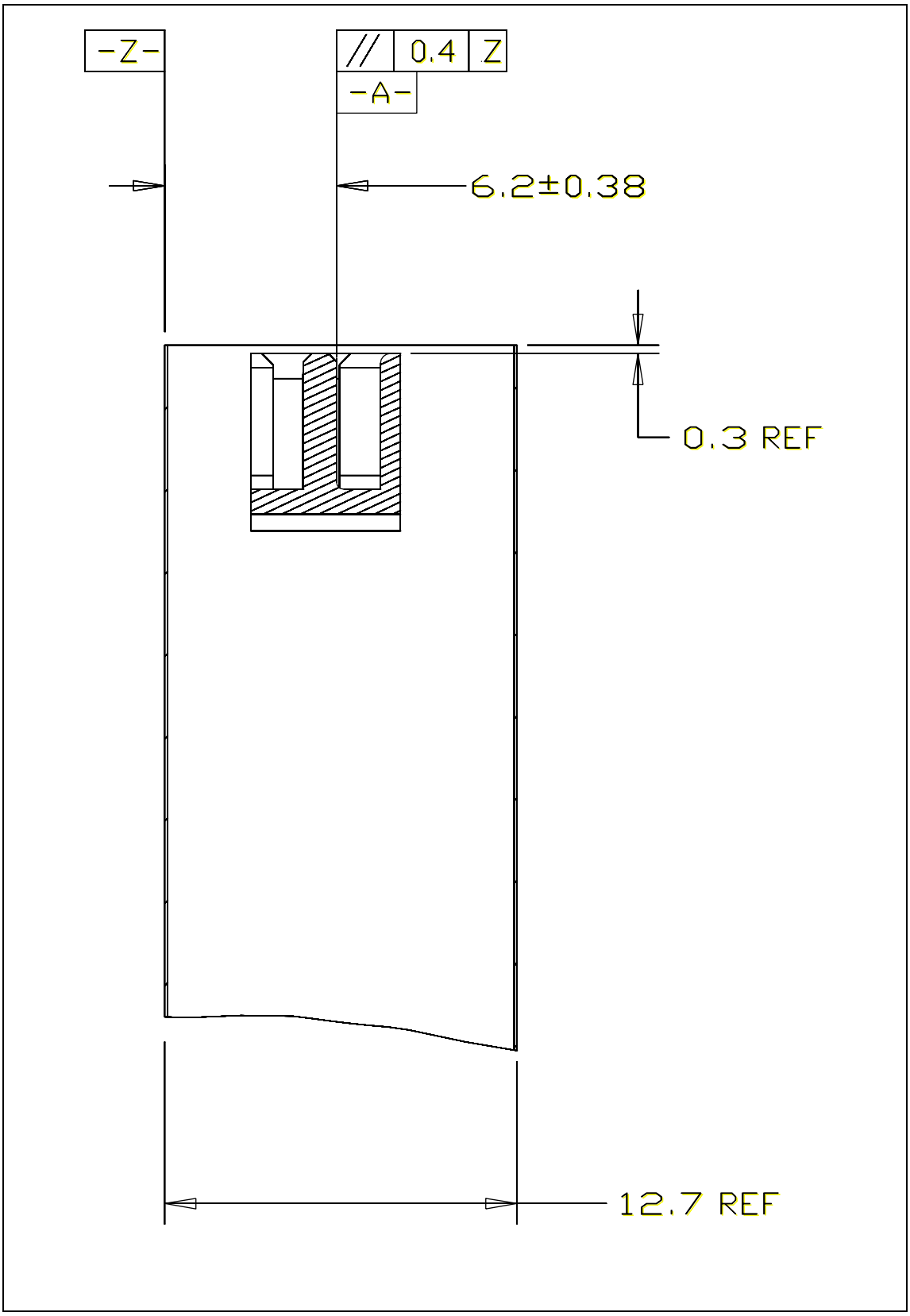


Figure 66 – 12.7 mm Slimline drive connector location (section A-A) (informative)

6.4.5 Mating interfaces

6.4.5.1 Mating interfaces overview

Serial ATA connectors are defined in terms of their mating interface or front end characteristics only. Connector back end characteristics including PCB mounting features and cable termination features are not defined.

Certain informative dimensions may be defined to indicate specific assumptions made in the design to meet certain requirements of form, fit, and function.

6.4.5.2 7.0 mm Slimline device plug connector

6.4.5.2.1 7.0 mm Slimline device plug dimentions

This is the connector used in the 7.0 mm Slimline device. Figure 67 to Figure 71 show the interface dimensions for the device plug connector with both signal and power segments. General tolerances on these drawings are ± 0.20 mm.

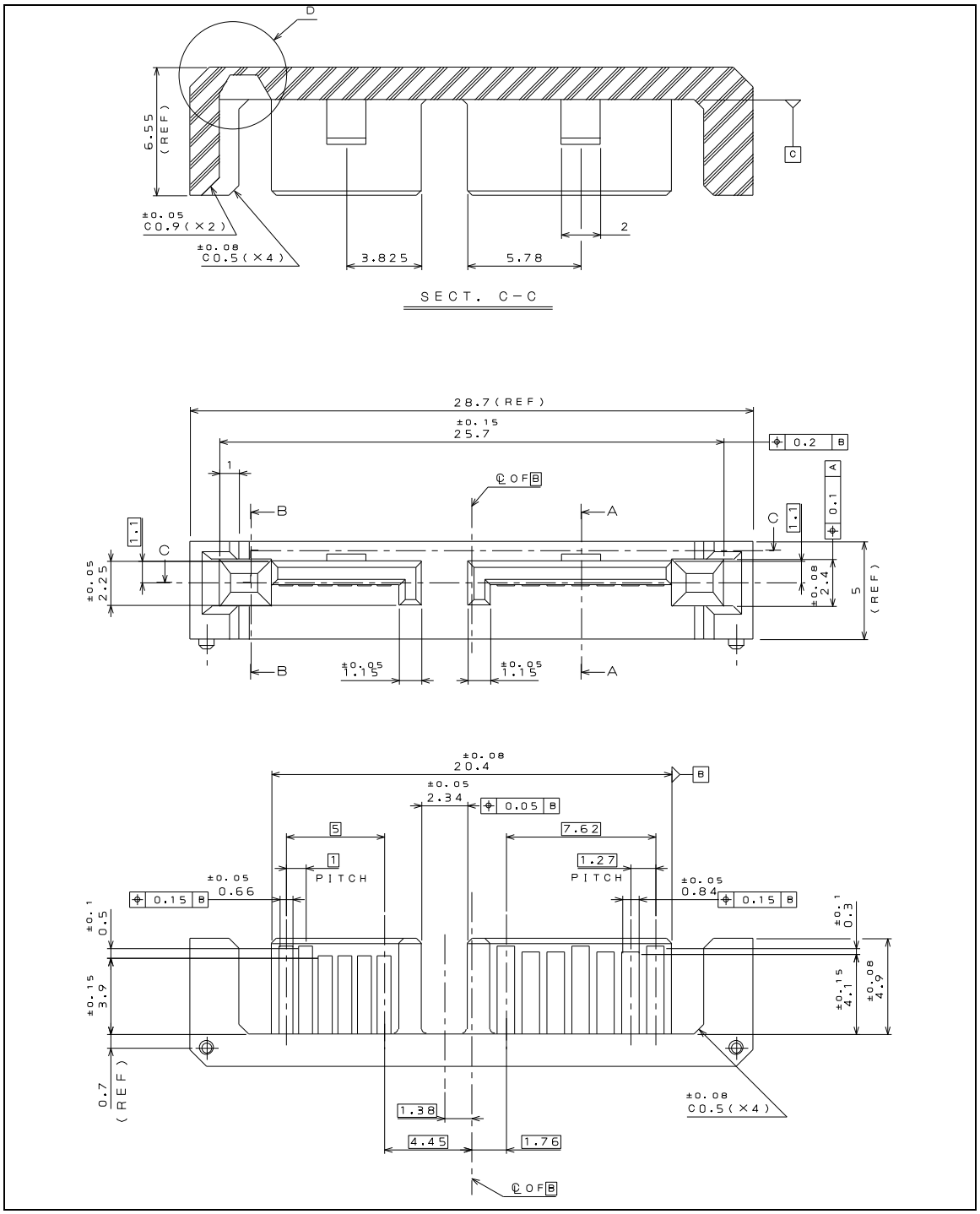


Figure 67 – 7.0 mm Slimline device plug connector interface dimensions

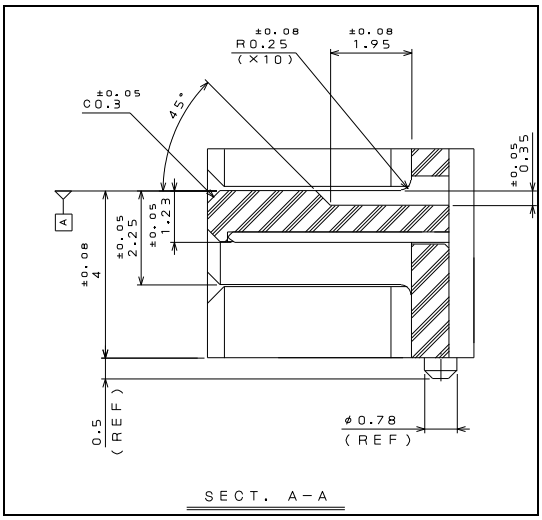


Figure 68 – 7.0 mm Slimline device plug connector interface dimensions section A-A

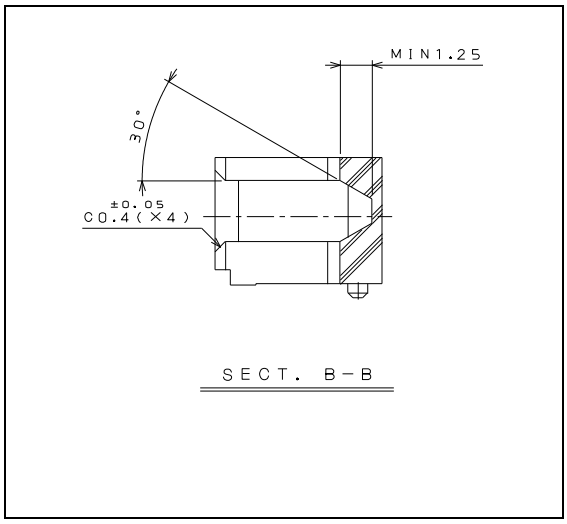


Figure 69 – 7.0 mm Slimline device plug connector interface dimensions section B-B

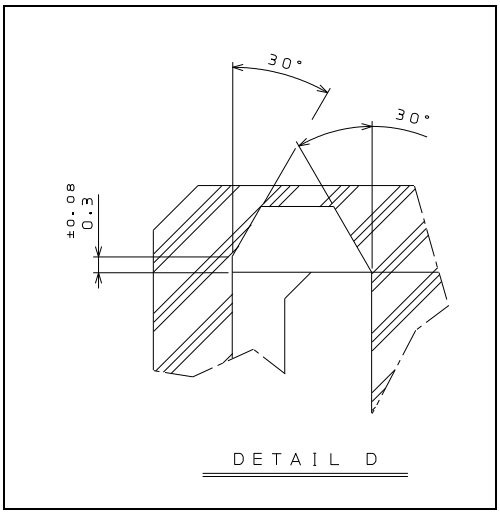


Figure 70 – 7.0 mm Slimline device plug connector interface dimensions detail D

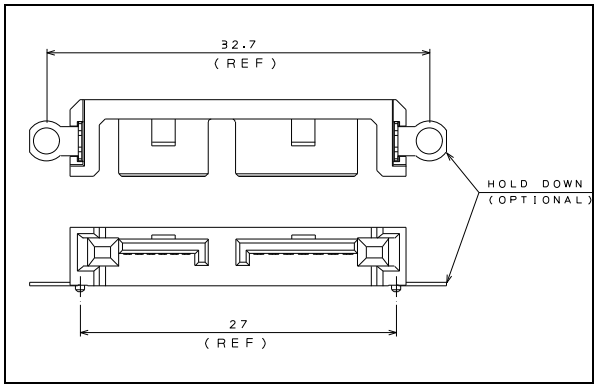


Figure 71 – 7.0 mm Slimline device plug connector interface dimensions optional hold down mounting

6.4.5.2.2 Connector pin signal definition

Refer to Table 15 for connector pin assignment and description.

6.4.5.2.3 Housing and contact electrical and mechanical requirement

The internal SATA connector and cable shall meet the requirements as defined for standard internal SATA cables and connectors in 6.2.10.2.

6.4.5.2.4 Connector and cable assembly requirements and test procedure

The connector and cable assembly requirements and test procedure shall confirm to those in 6.4.8.

6.4.5.3 8.5 mm, 9.5 mm, and 12.7 mm Slimline device plug connector

6.4.5.3.1 Slimline device plug

This is the connector used in the Slimline device.

Figure 72 to Figure 78 and Table 15 show the interface dimensions for the device plug connector with both signal and power segments. General tolerances on these drawings is ± 0.20 mm.

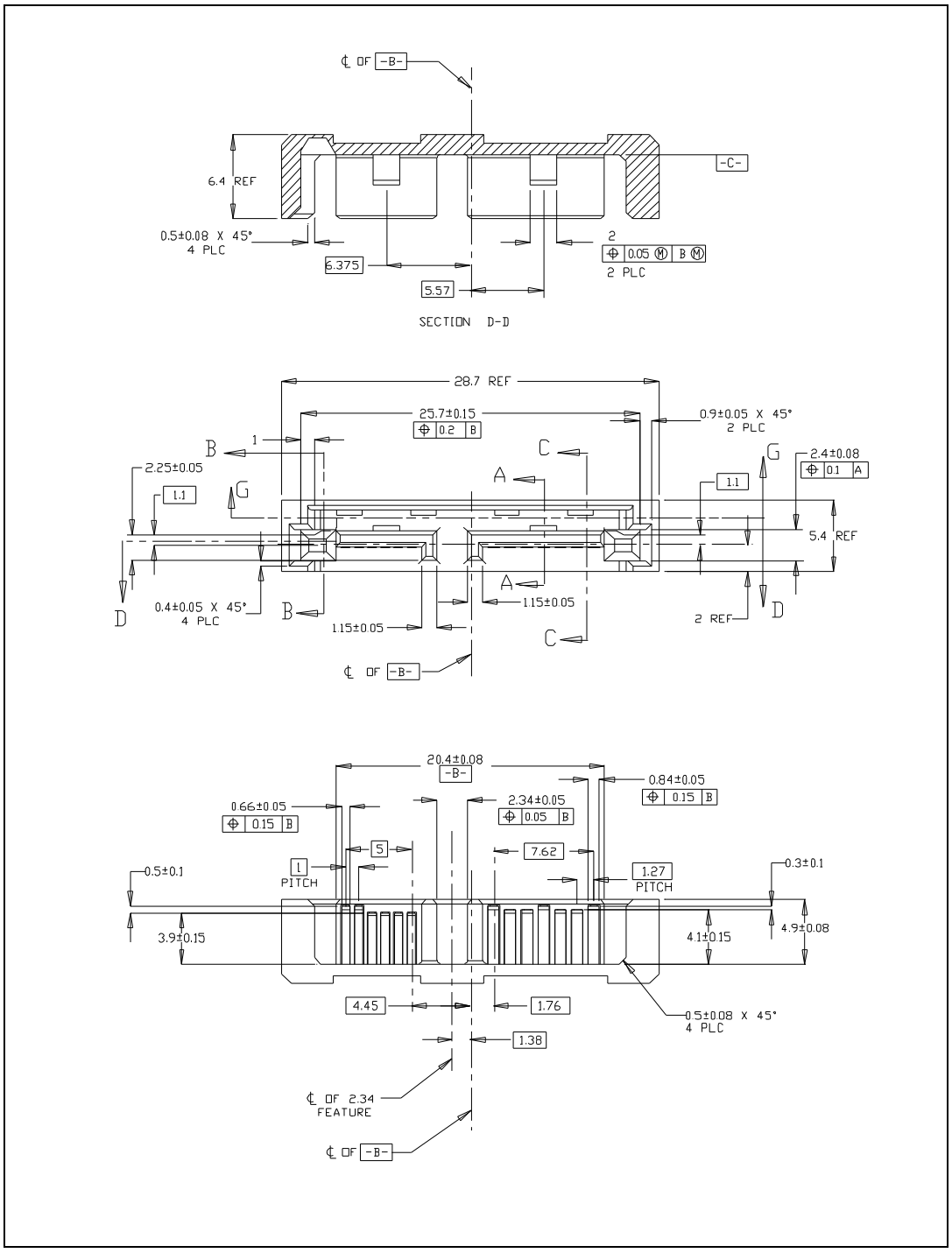


Figure 72 – Slimline device plug connector interface dimensions

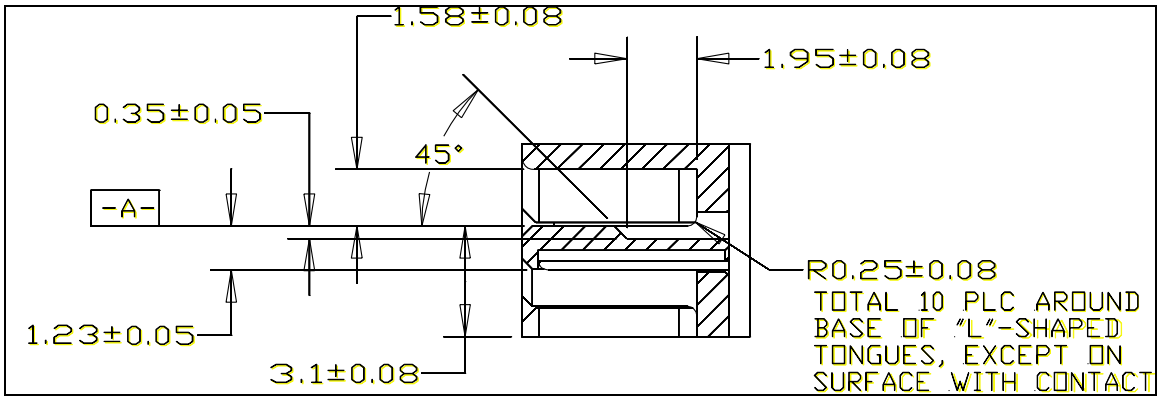


Figure 73 – Slimline device plug connector interface dimensions (section A-A)

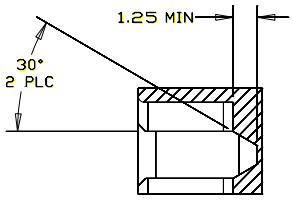


Figure 74 – Slimline device plug connector interface dimensions (section B-B)

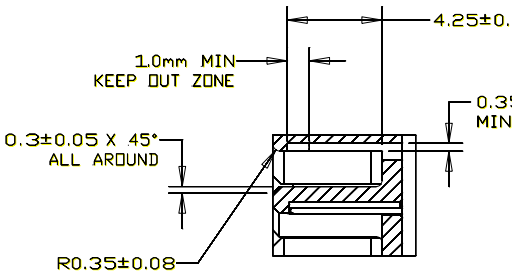


Figure 75 – Slimline device plug connector interface dimensions (section C-C)

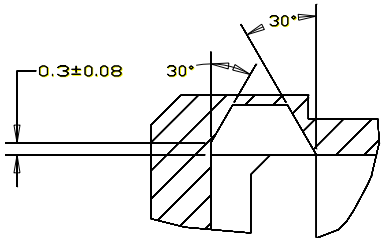


Figure 76 – Slimline device plug connector interface dimensions (detail F)

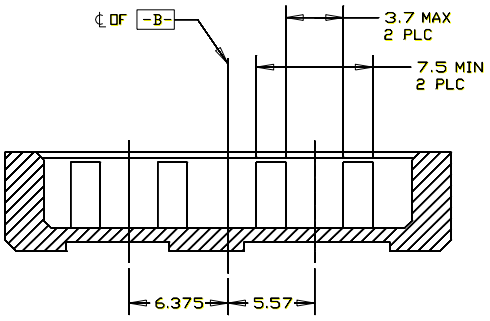


Figure 77 – Slimline device plug connector interface dimensions (section G-G)

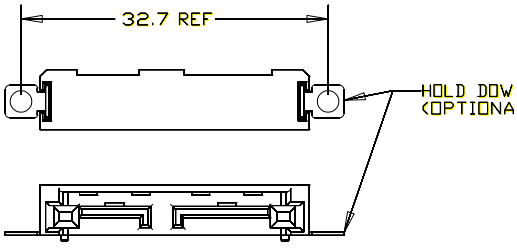


Figure 78 – Slimline device plug connector optional hold down mounting

There are total of 7 pins in the signal segment and 6 pins in the power segment (see Figure 79). The pin definitions are shown in Table 15.

Table 15 – Slimline device plug connector pin definition

	Name	Type	Description	Cable Usage [a] [b]	Backplane Usage [b]
Signal Segment	Refer to Table 5.				
Signal Segment "L"					
Central Connector Gap					
Power Segment "L"					
Power Segment	P1	DP	Device Present	3rd mate	3rd mate
	P2	+5 V [e]		2nd mate	2nd mate
	P3	+5 V [e]		2nd mate	2nd mate
	P4	MD/DA	Manufacturing Diagnostic/Device Attention [d]	2nd mate	2nd mate
	P5	Gnd [c]	Ground	1st mate	1st mate
	P6	Gnd [c]	Ground	1st mate	1st mate
Power Segment Key					

[a] All pins are in a single row with 1.00 mm (0.039 inch) pitch on the power segment portion.
[b] Ground pins in the Serial ATA Slimline device plug power segment (connector pins P5 and P6) shall be bussed together on the Serial ATA Slimline device.
[c] The connection between the Serial ATA Slimline device signal ground and power ground is vendor specific.
[d] The DP and MD/DA signals shall be referenced to the power portion ground pins, P5 and P6.
[e] The 5 V power delivery pins in the Serial ATA Slimline device plug power segment (connector pins P2 and P3) shall be bussed together in the Serial ATA Slimline device.

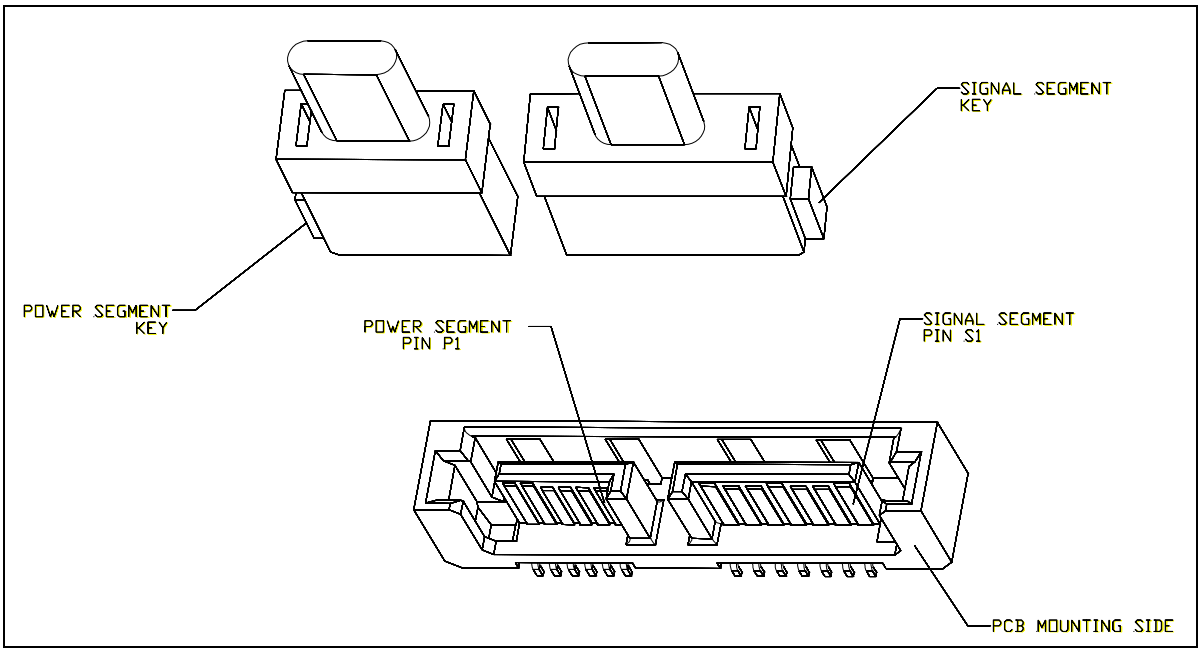


Figure 79 – Slimline connector pin and feature locations

6.4.5.3.2 Slimline Signal cable receptacle connector

The standard SATA signal cable receptacle is used (see 6.2.4).

6.4.5.3.3 Slimline power cable receptacle connector

The Slimline power cable receptacle connector mates with the power segment of the device plug, bringing power to the device. Figure 80 to Figure 82 shows the interface dimensions of the power receptacle connector. The pin out of the connector is the mirror image of the power segment of the device plug shown in Table 15. The MD and DP connector pins are optionally present in a cabled environment.

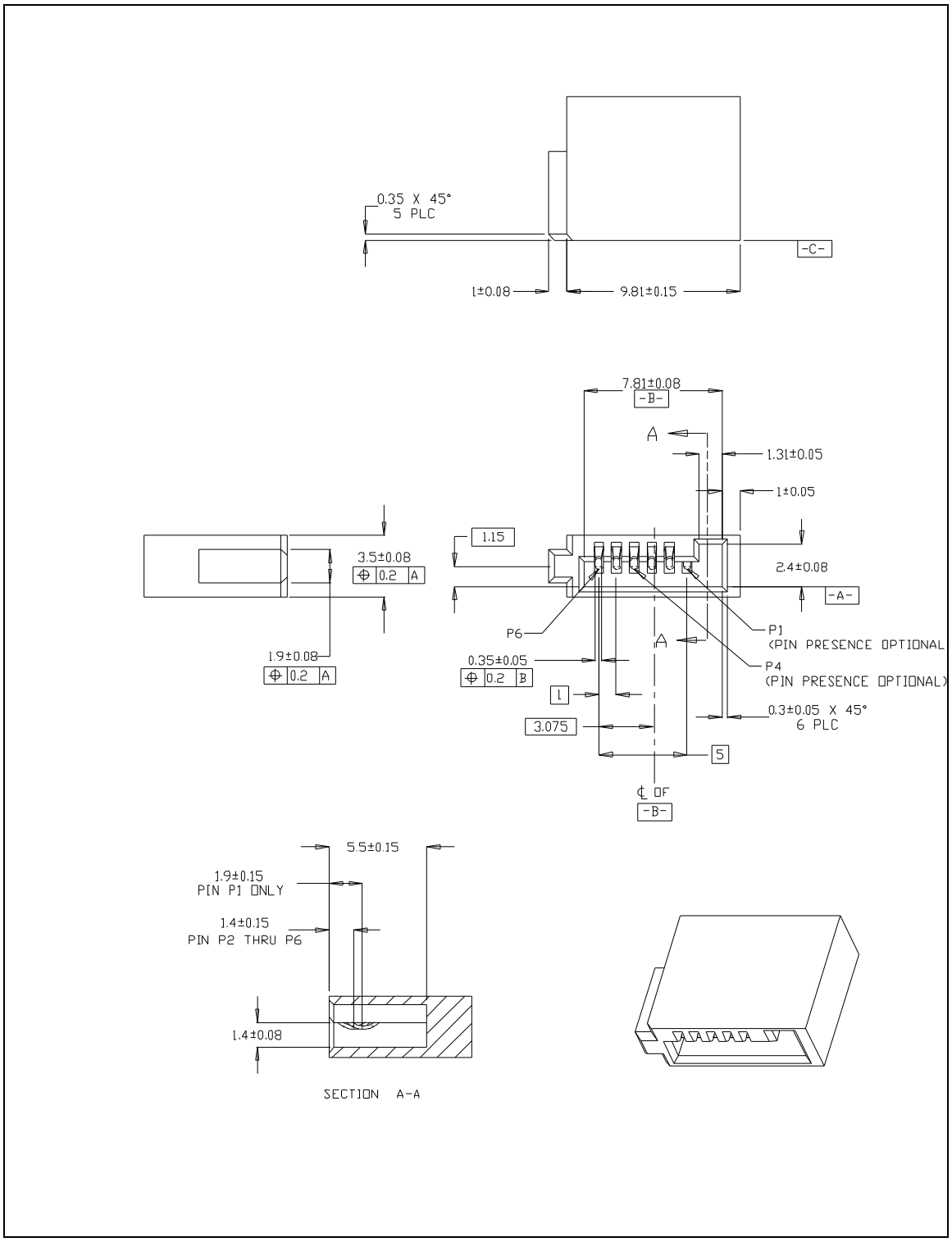


Figure 80 – Slimline power receptacle connector interface dimensions

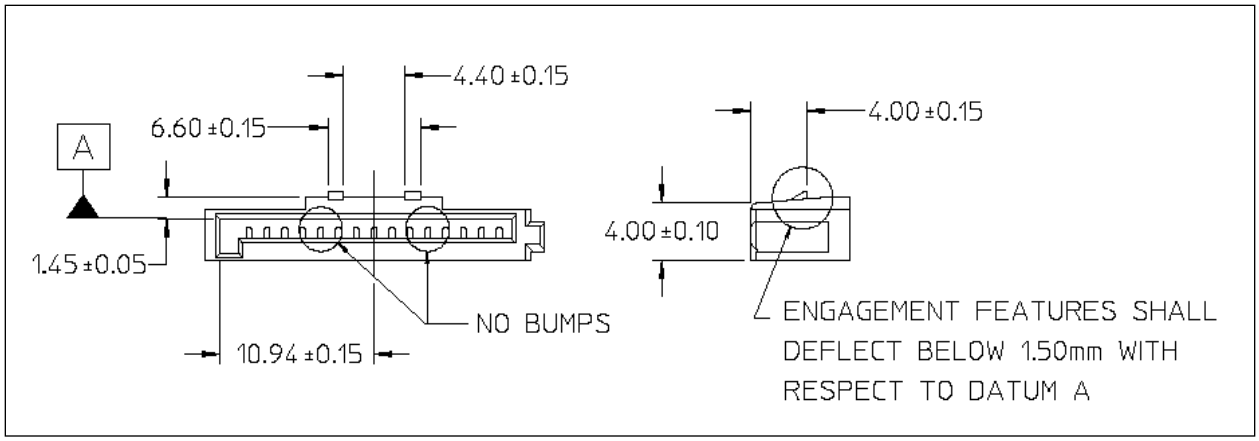


Figure 81 – Slimline power receptacle connector option with latch

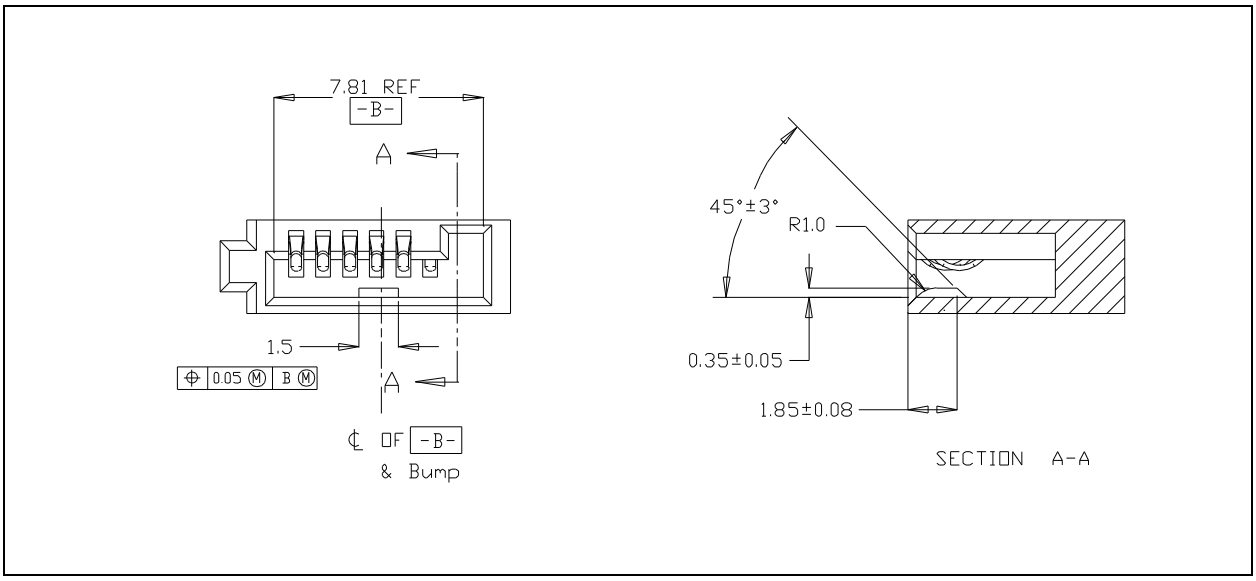


Figure 82 – Slimline power receptacle connector option with bump

6.4.5.4 Slimline host receptacle connector

The Slimline host receptacle connector is to be blind-mated directly with the device plug connector. The interface dimensions for the host receptacle connector are shown in Figure 83 to Figure 86. An appropriate external retention mechanism independent of the connector is required to keep the host PCB and the device in place. The Slimline host receptacle connector is not designed with any retention mechanism.

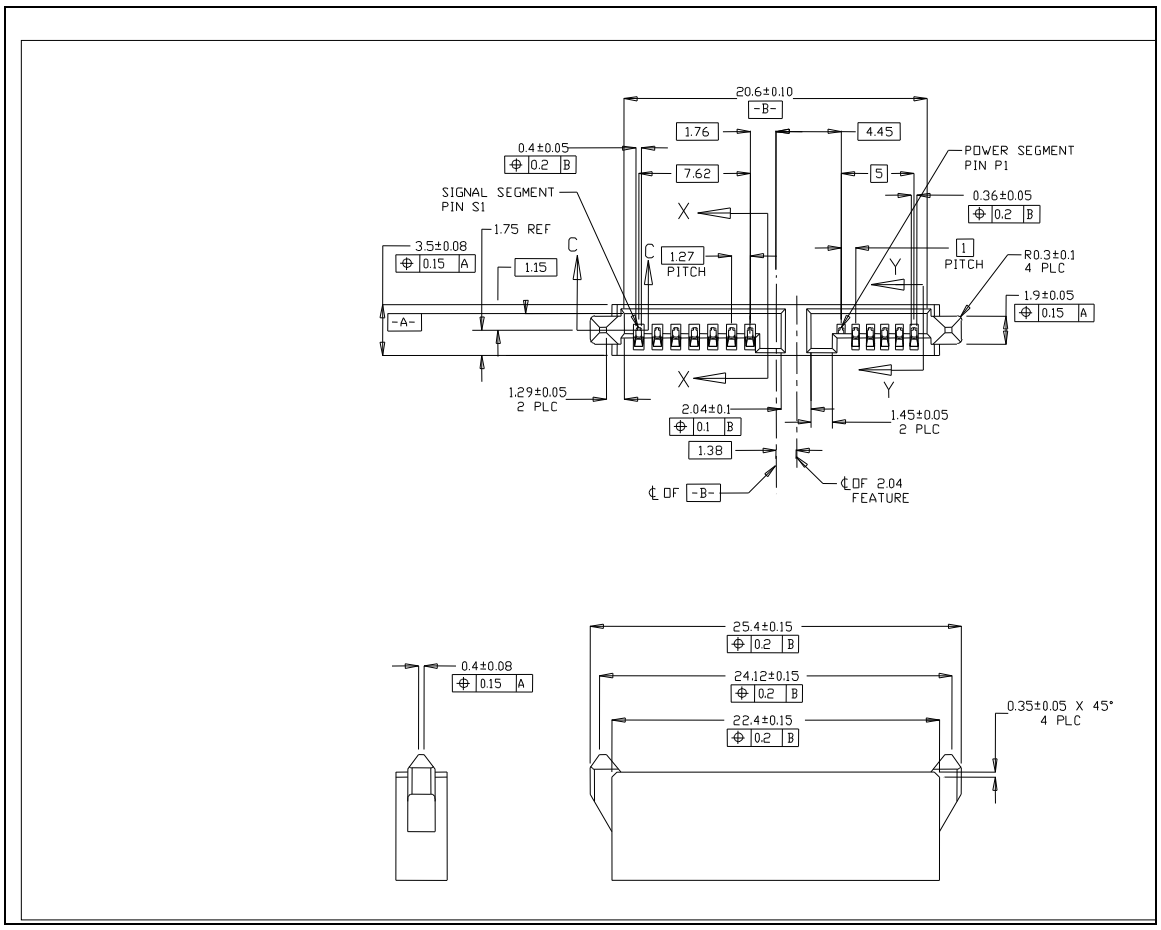


Figure 83 – Slimline host receptacle connector interface dimensions

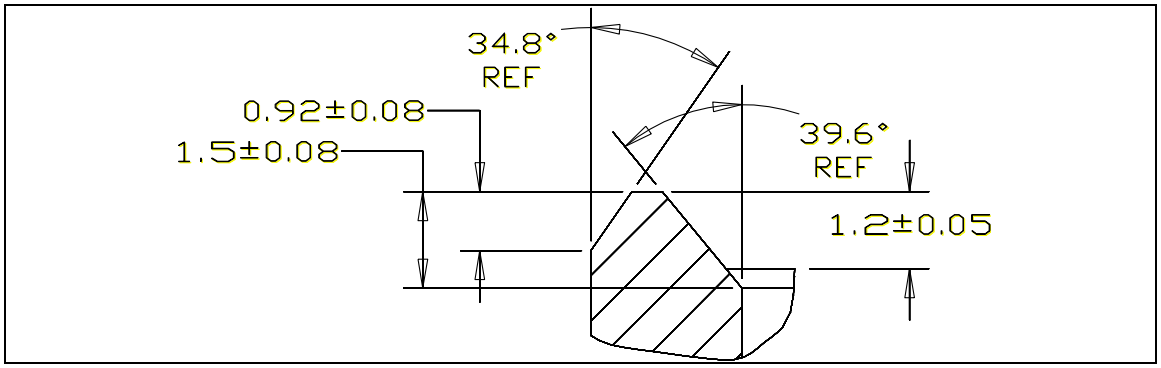


Figure 84 – Slimline host receptacle connector interface dimensions section C-C

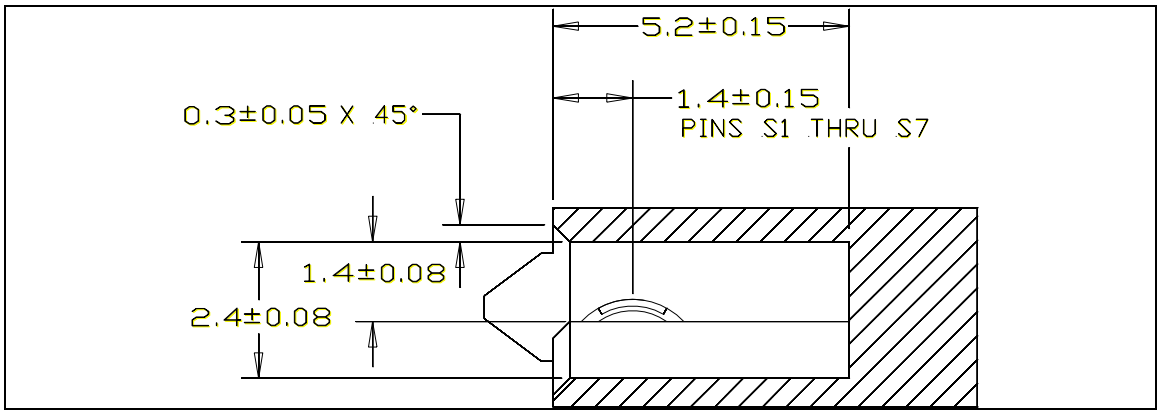


Figure 85 – Slimline host receptacle connector interface dimensions section X-X

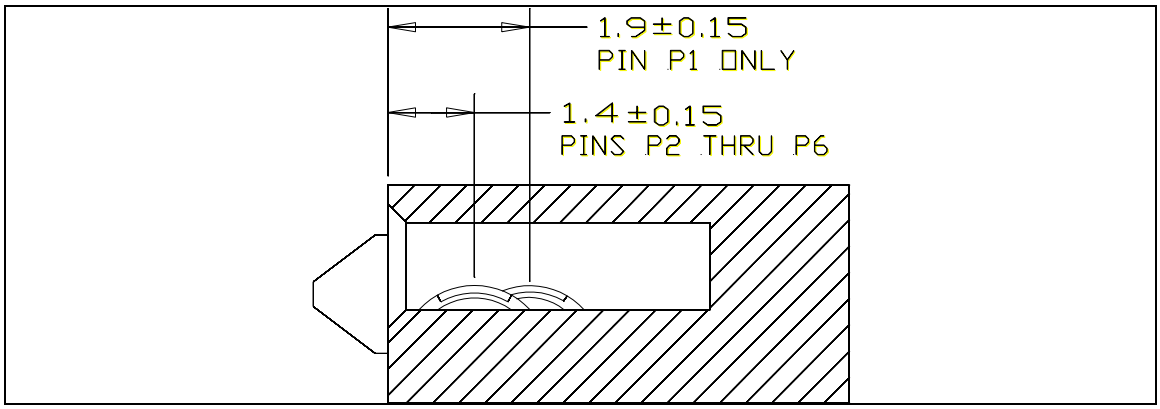


Figure 86 – Slimline host receptacle connector interface dimensions section Y-Y

6.4.6 Backplane connector configuration and blind-mating tolerance

The maximum blind-mate misalignment tolerances are ± 1.50 mm and ± 1.20 mm, respectively, for two perpendicular axes illustrated in Figure 87. Any skew angle of the plug, with respect to the receptacle, reduces the blind-mate tolerances.

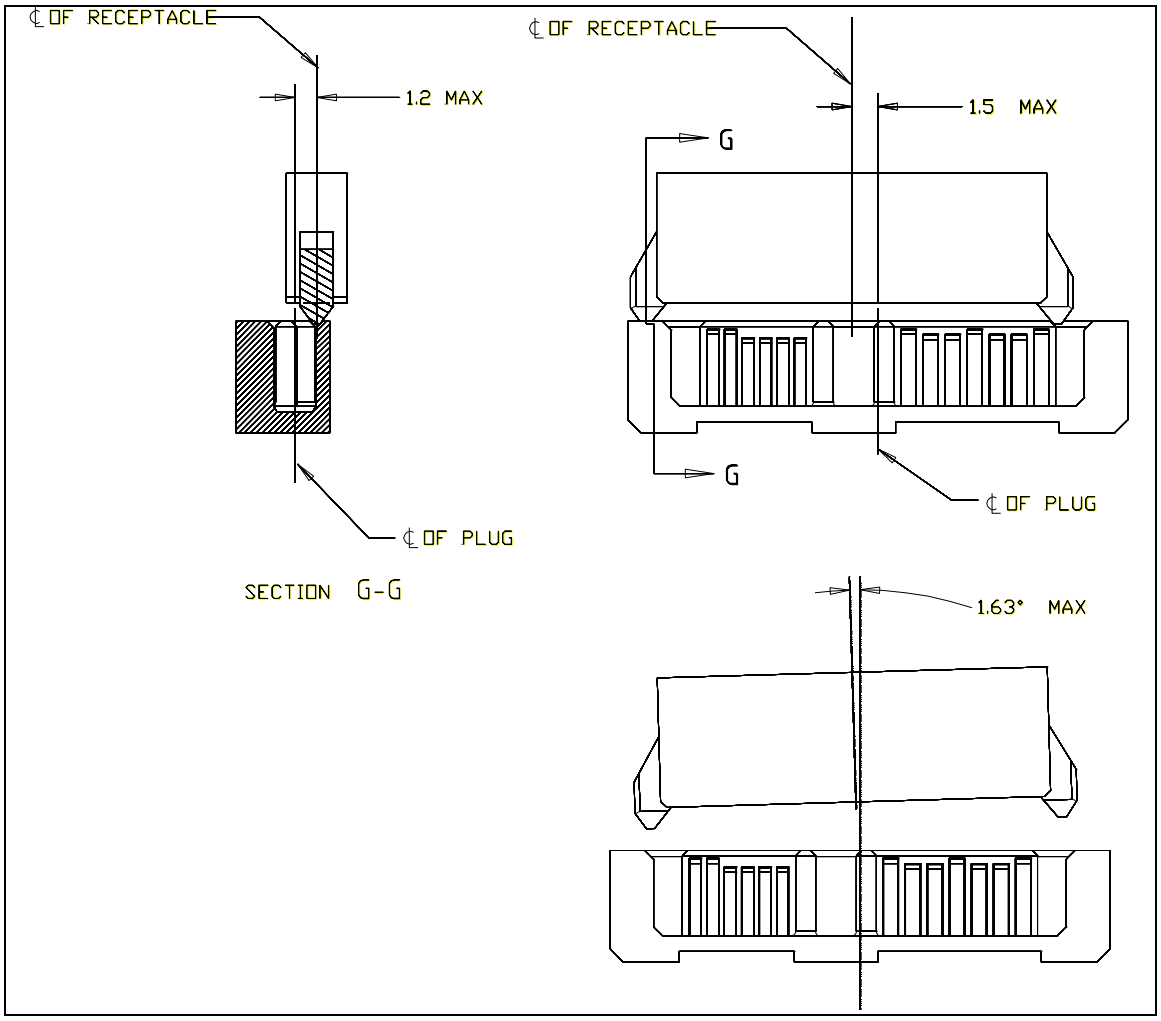


Figure 87 – Slimline connector pair blind-mate misalignment tolerance

6.4.7 Connector labeling

According to 6.2.9.

6.4.8 Connector and cable assembly requirements and test procedures

The connector and cable assembly requirements and test procedures shall conform to those in Table 8 with the following exceptions found in Table 16.

Table 16 – Slimline connector mechanical test procedures and requirements

Parameter	Procedure	Requirement
Removal force Backplane connector	EIA-364-13 Measure the force necessary to unmate the connector assemblies at a max. rate of 12.5 millimeter per minute.	Min 2.5 N after 500 cycles.
Insertion force Cabled power connector Without latch	EIA-364-13 Measure the force necessary to mate the connector assemblies at a max. rate of 12.5 millimeter per minute.	Max 45 N.
Removal force Cabled power connector Without latch	EIA-364-13 Measure the force necessary to unmate the connector assemblies at a max. rate of 12.5 millimeter per minute.	Min 10 N for cycles 1 to 5. Min 8 N through 50 cycles.
Insertion force Cabled power connector With latch	EIA-364-13 Measure the force necessary to mate the connector assemblies at a max. rate of 12.5 millimeter per minute.	Max 45 N.
Removal force Cabled power connector With latch	EIA-364-13 Apply a static 25 N unmating test load	No damage and no disconnect through 50 mating cycles.

6.5 Internal LIF-SATA connector for 1.8 inch HDD

6.5.1 Internal LIF-SATA connector for 1.8 inch HDD overview

This section provides capabilities required to enable a new smaller Serial ATA 1.8 inch HDD.

The definition supports the following capabilities:

a) supports Gen1 (1.5 Gbit/s) and Gen2 (3 Gbit/s) transfer rates;
b) support for FPC usage models;
c) support for 8.0 mm and 5.0 mm slim 1.8 inch form factor HDD's;
d) support of 3.3 V with 5 V to meet future product requirements;
e) support vendor pins, P18, P19, P20, and P21 reserved for HDD customer usage; and
f) support vendor pins, P22 and P23 for HDD manufacturing usage.

This LIF-SATA is only for internal 8.0 mm and 5.0 mm slim 1.8 inch form factor devices.

> NOTE 12 - It is expected that the LIF-SATA interfaces comply with Gen1i and Gen2i specifications. The LIF-SATA connector is only possible to be mated with FPC cable, the compliance point shall be after the mated assembly, where the necessary Tx and Rx measurements were measured.

6.5.2 General description

This internal LIF-SATA connector is designed to enable connection of a new family of slim 1.8 inch form factor HDD's to the Serial ATA interface.

The internal low insertion force (LIF) Serial ATA connector uses the 0.5 mm pitch configuration for both the signal and power segments. The signal segment has the same configuration as the internal standard Serial ATA connector but power segment provides the present voltage requirement

support of 3.3 V, and provision for a future voltage requirement of 5 V. In addition, there is P8 (i.e., defined as DAS/Disable Staggered Spinup (DSS)). Finally, there are 6 vendor pins, P18, 19, 20 and 21 for HDD customer usage, and P22 and P23 for HDD manufacturing usage.

Care should be taken in the application of this drive so that excessive stress is not exerted on the drive or connector.

6.5.3 Connector locations

6.5.3.1 Connector location on Hard Disk Drive (HDD) form factor, - (informative)

The internal LIF-SATA connector location on the HDD is shown in Figure 88 for reference purposes only. See SFF-8146 for form factor definition and connector location.

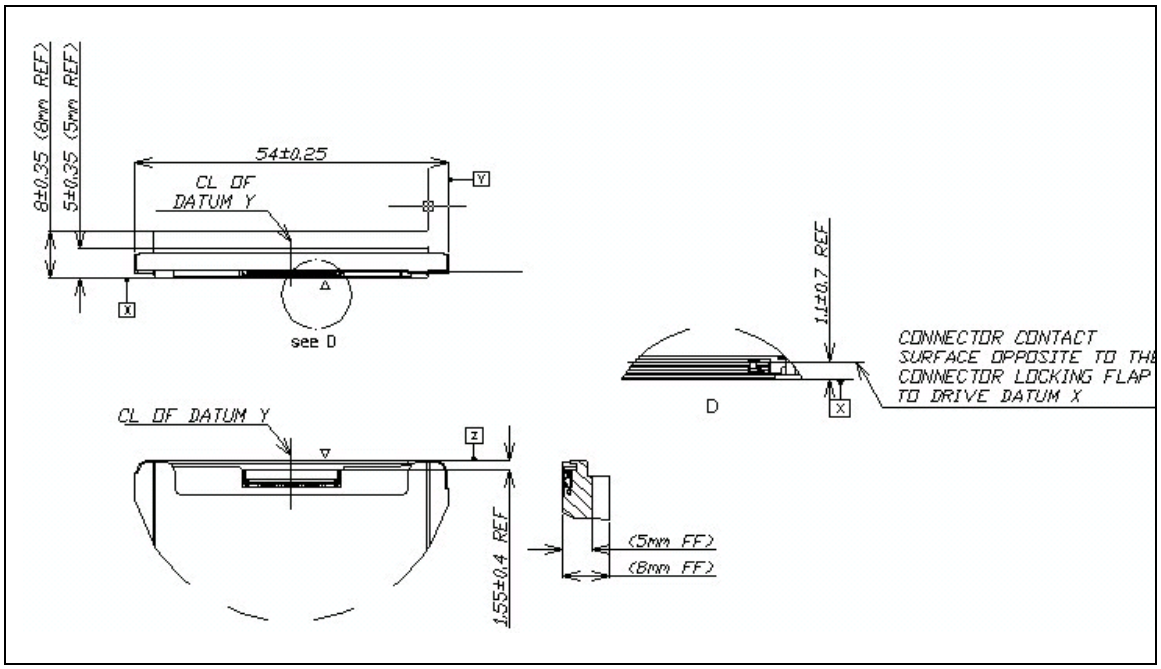


Figure 88 – Internal LIF-SATA connector location for 1.8 inch HDD

6.5.3.2 Connector location on Solid State Drive (SSD) form factor, - (informative)

The internal LIF-SATA connector location on Solid State Drive (SSD) Bulk Type is shown in Figure 89 for reference purposes only.

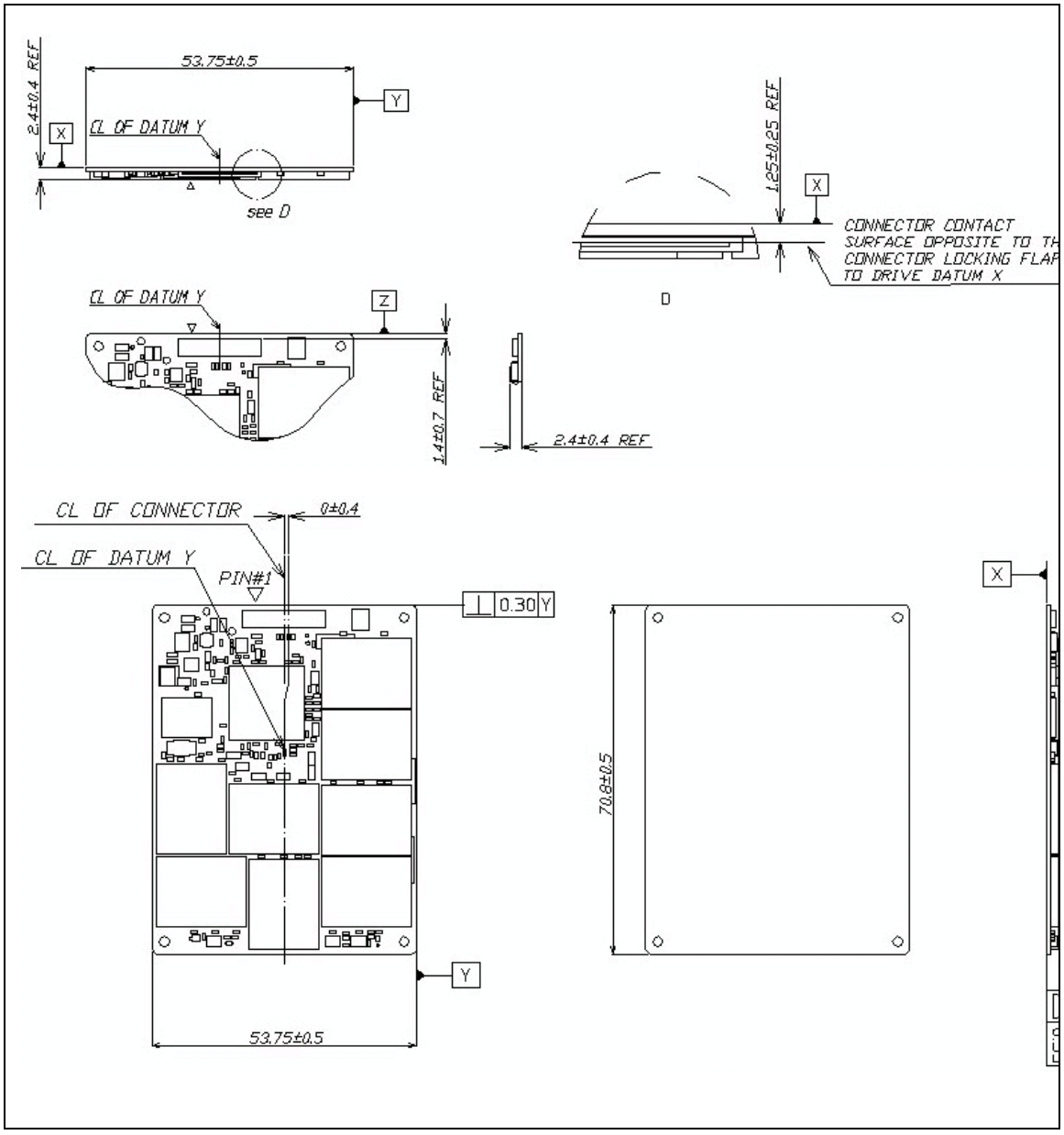
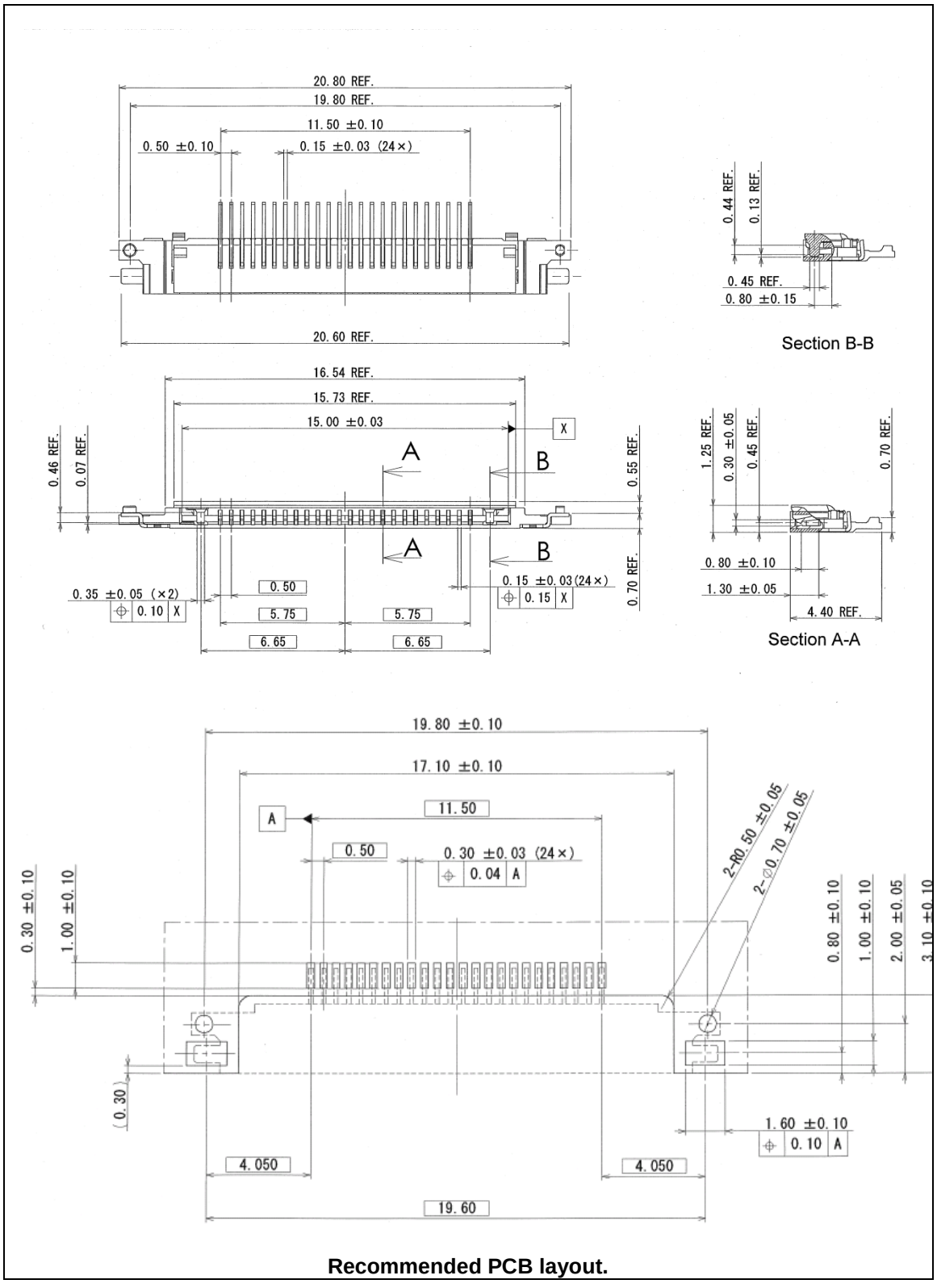


Figure 89 – Internal LIF-SATA connector location for 1.8 inch SSD bulk of single-sided mount type

6.5.4 Mating interfaces

6.5.4.1 Device internal LIF-SATA embedded type connector

Figure 90 defines the interface dimensions for the internal LIF-SATA embedded type connector with both signal and power segments.



Recommended PCB layout.

Figure 90 – Device internal LIF-SATA embedded type connector

6.5.4.2 Device internal LIF-SATA surface mounting type connector

Figure 91 defines the interface dimensions for the internal LIF-SATA device surface mounting type connector.

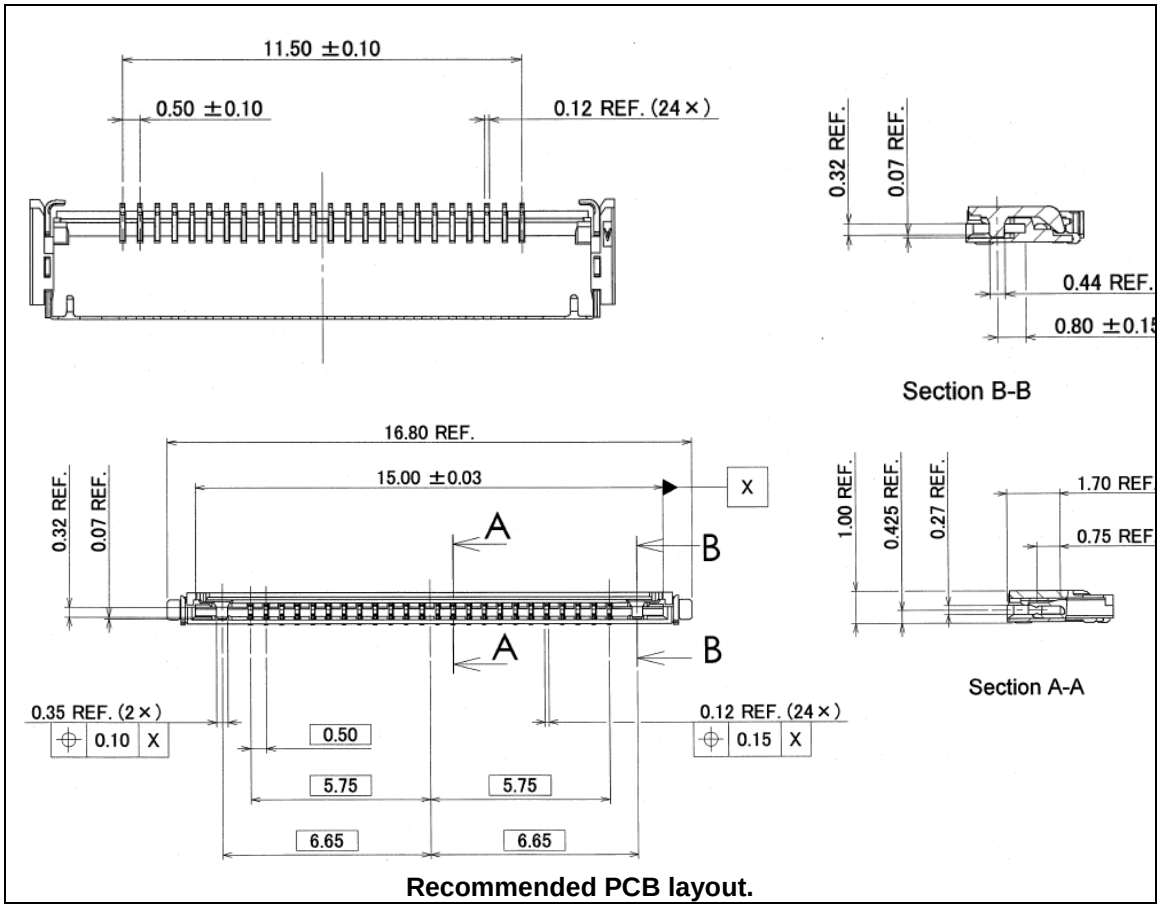


Figure 91 – Device internal LIF-SATA surface mounting type connector

6.5.4.3 FPC for internal LIF-SATA

Figure 92 defines the interface dimensions for the FPC of LIF-SATA.

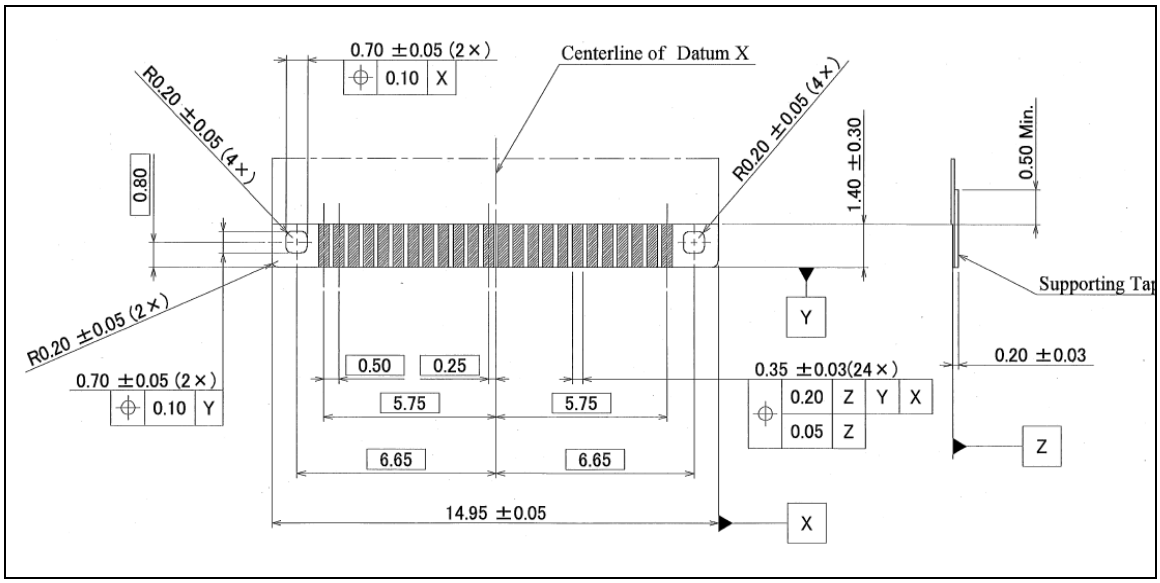


Figure 92 – FPC for internal LIF-SATA

6.5.5 Internal LIF-SATA pin signal definition and contact mating sequence

Table 17 details the pin names, types and contact order of the two internal LIF-SATA plug options. A brief description is also included for signal, ground and power pins. There are total 24 pins.

Table 17 – Signal and power internal LIF-SATA plug

Name	Type	Description
P1	GND	Ground
P2	V_{33}	3.3 V Power
P3	V_{33}	3.3 V Power
P4	GND	Ground
P5	V_5	5 V Power
P6	V_5	5 V Power [a]
P7	GND	Ground
P8	DAS/DSS/Vendor Specific	Device Activity Signal/Disable Staggered Spinup/Vendor Specific [b]
P9	GND	Ground
P10	GND	Ground
P11	A+	Differential Signal Pair A
P12	A-	
P13	GND	Ground
P14	B-	Differential Signal Pair B
P15	B+	
P16	GND	Ground
P17	GND	Ground
P18	Vendor	Vendor Specific
P19	Vendor	Vendor Specific
P20	Vendor	Vendor Specific
P21	DHU	Direct Head Unload [d]
P22	Vendor	Vendor Specific - Mfg pin [c]
P23	Vendor	Vendor Specific - Mfg pin [c]
P24	GND	Ground

[a] The 5 V supply voltage pins are included to meet future product requirements. Future revisions of this specification may require 5 V supply voltage be provided.
[b] For specific optional usage of pin P8 see 6.13.
[c] No connect on the host side.
[d] For specific optional usage of pin P21 see 6.13.

6.5.6 Housing and contact electrical requirement

The internal LIF-SATA connector and cable shall meet the requirements as defined for standard internal SATA cables and connectors in 6.2.10, Connector and FPC assembly requirements and test procedures, with the exceptions listed below in Table 18.

Table 18 – Unique connector mechanical testing procedures and requirements

Parameter	Procedure	Requirement
Insertion force for FPC	Measure the force necessary to mate the connector assemblies at a max. rate of 12.5 millimeter per minute.	Max 6 N.
Removal force for FPC	Measure the force necessary to unmate the connector assemblies at a max. rate of 12.5 millimeter per minute.	Min 6 N through 10 cycles.
Durability	10 cycles for internal cabled application; Test done at a maximum rate of 200 cycles per hour.	No physical damage.
Minimum current	Mount the connector to a test PCB: 1) wire power pins P2, P3, P5, P6 in parallel for power; 2) wire ground pins P1, P4, P7, P9, P10, P13, P16, P17, P24 in parallel for return; 3) supply 2 A total DC current to the power pins in parallel, returning from the parallel ground pins; and 4) record temperature rise when thermal equilibrium is reached.	Min 0.5 A/pin. The temperature rise above ambient shall not exceed 30 °C at any point in the connector when contact positions are powered. The ambient condition is still air at 25 °C.

6.6 mSATA connector

6.6.1 mSATA connector overview

This section defines the requirements of an mSATA configuration with a Serial ATA interface. The definition supports the following capabilities:

a) supports Gen1 (1.5 Gbit/s) and Gen2 (3 Gbit/s) transfer rates;
b) support for mSATA;
c) support of 3.3 V;
d) support 4 vendor pins; and
e) support 2 vendor pins, for drive or SSD manufacturing usage.

6.6.2 General description

The mSATA connector is designed to enable connection of a new family of small form factor devices to the Serial ATA interface.

The signal assignments are outlined in Table 19. All mSATA physical dimensions are under the control of JEDEC and provided in SATA as informative. See JEDEC MO-300 for all physical requirements.

6.6.3 mSATA host connector, footprint, and mounting

See JEDEC MO-300 for form factor definition and connector location.
The mSATA host connector physical dimensions are shown as informative in Figure 93. The mSATA host footprint and keepout are shown as informative in Figure 94 and Figure 95.

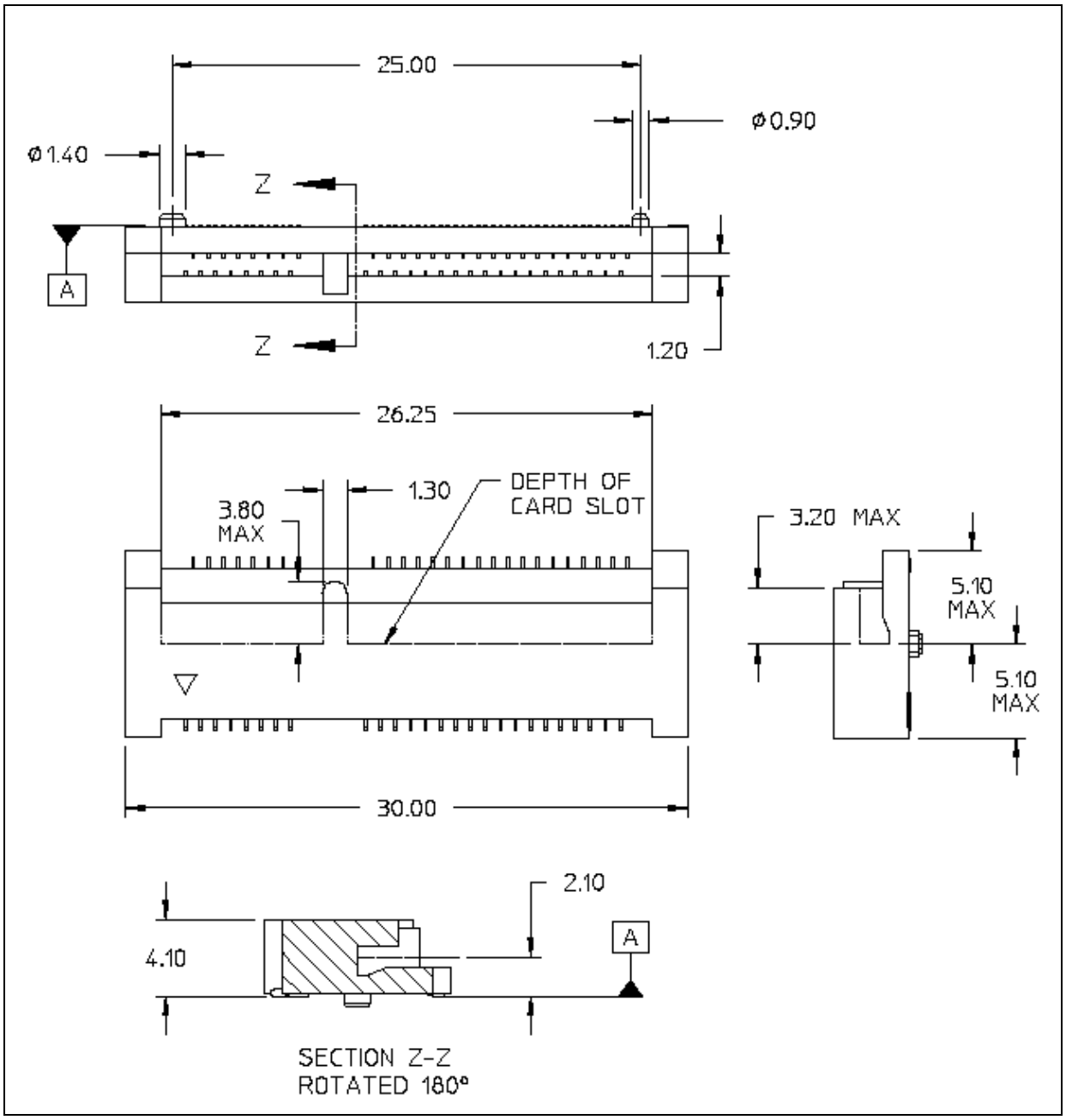


Figure 93 – mSATA card connector (informative)

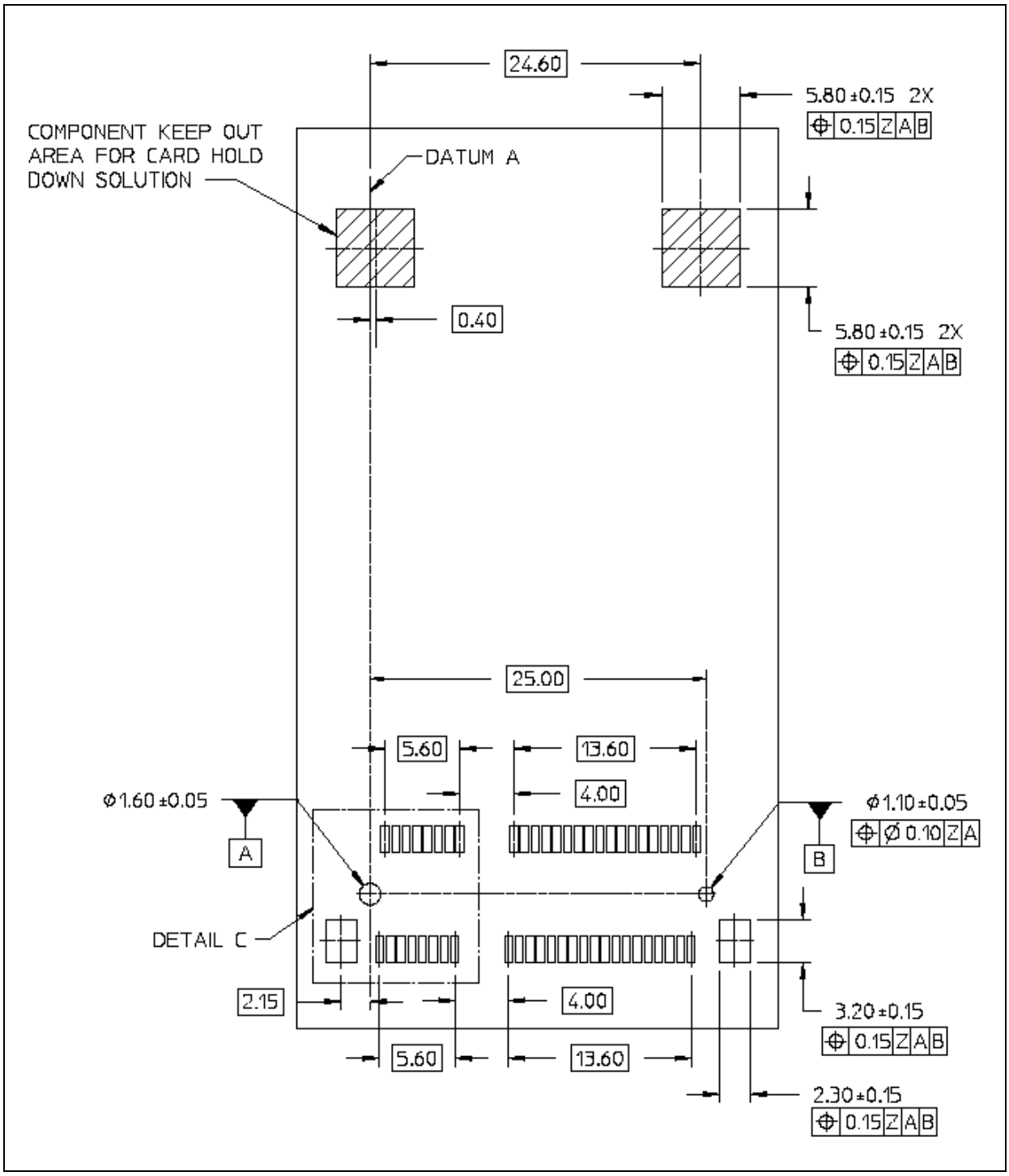


Figure 94 – mSATA card footprint and keepout (informative)

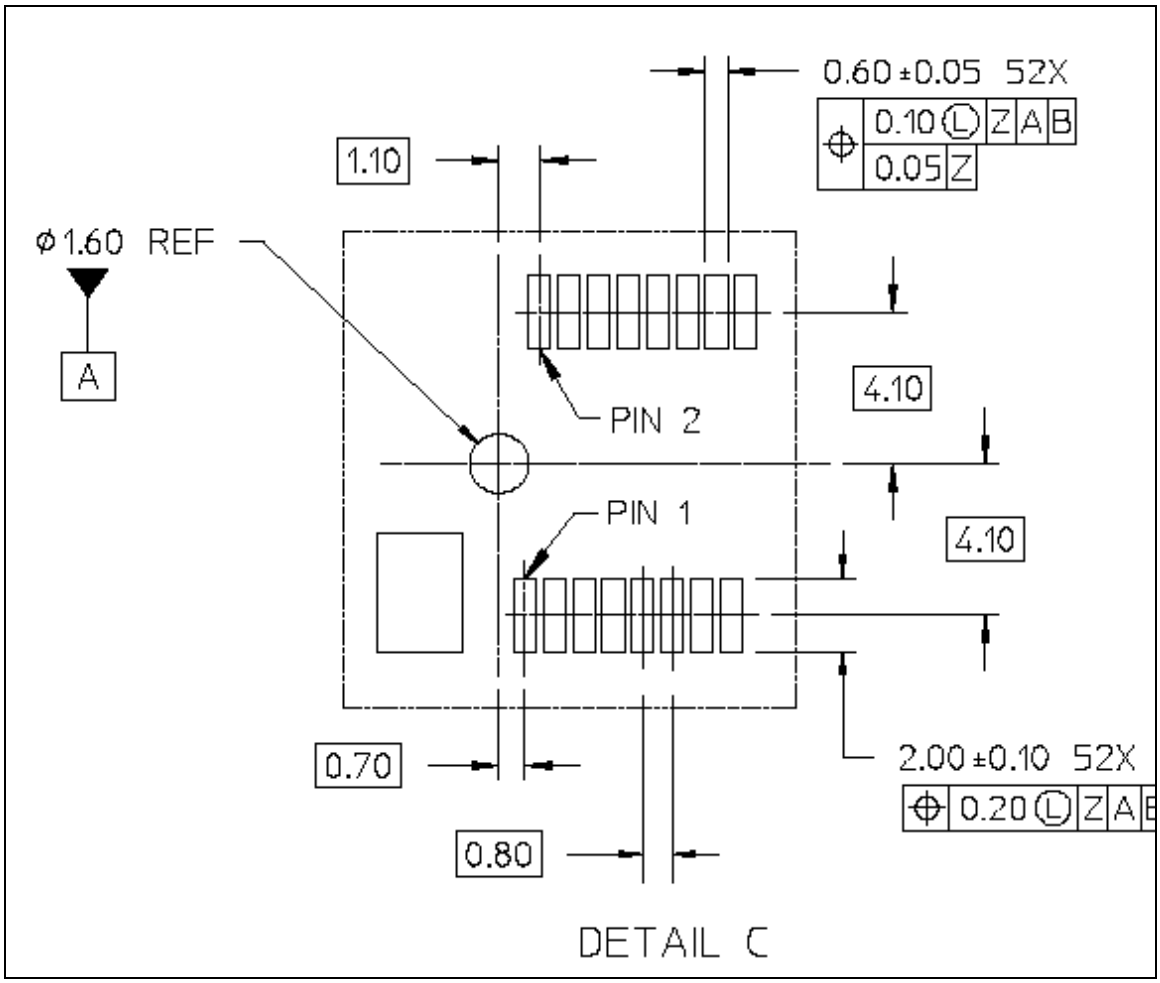


Figure 95 – mSATA card connector location detail C (informative)

6.6.4 mSATA device dimensions

Figure 96 defines the device mSATA card type internal connector.

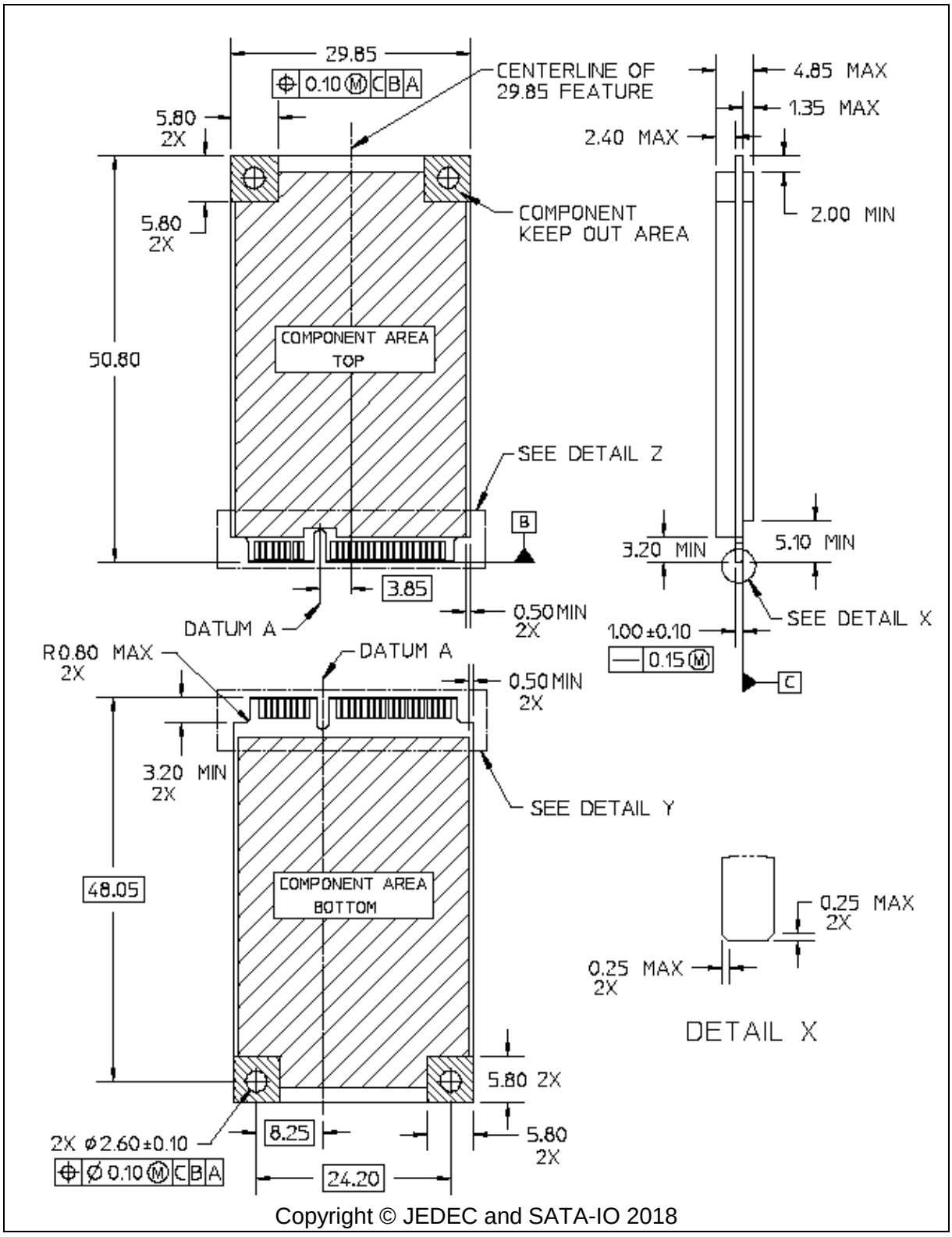


Figure 96 – Device mSATA card type internal connector (informative) (part 1 of 2)

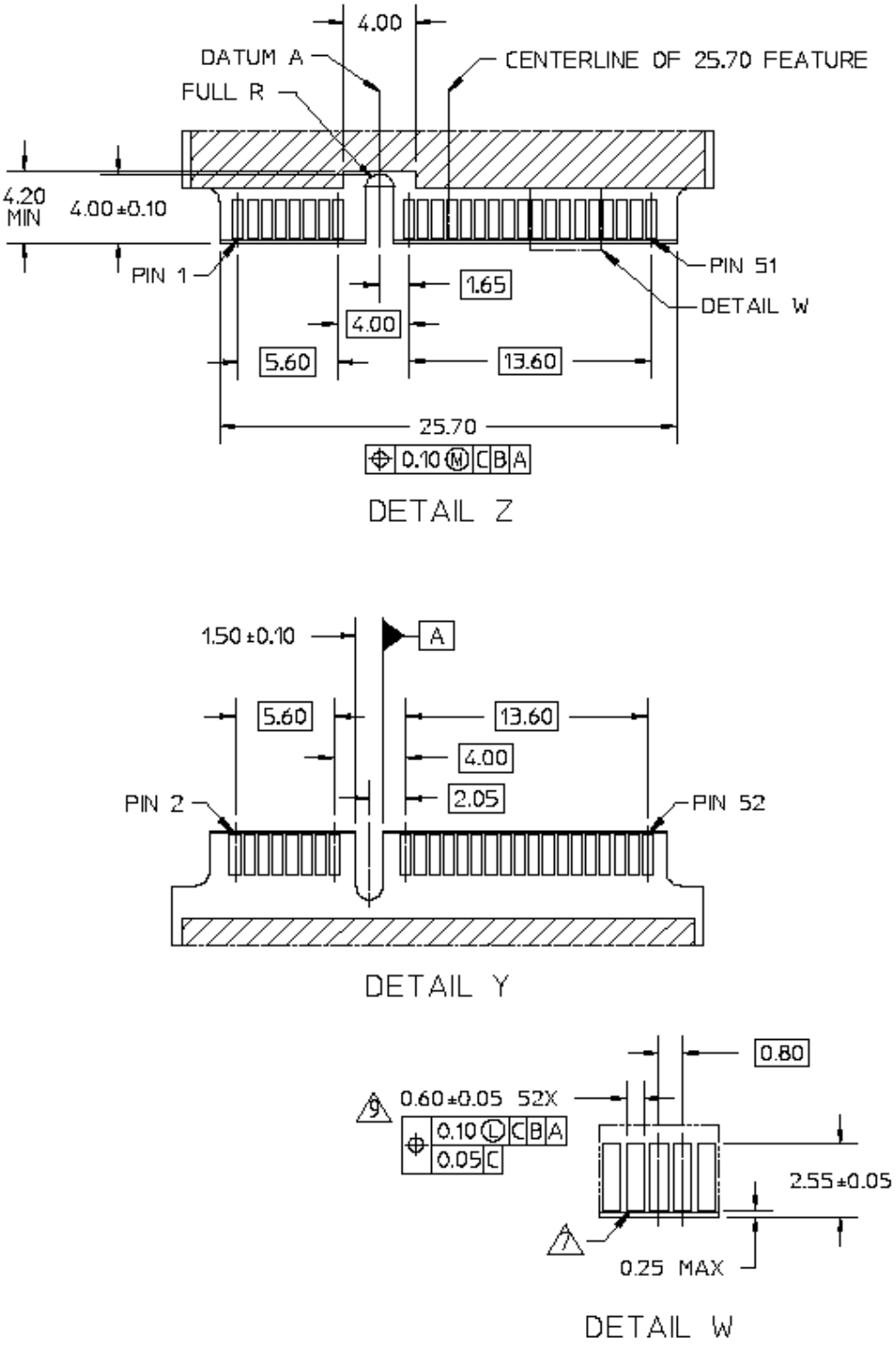


Figure 96 – Device mSATA card type internal connector (informative) (part 2 of 2)

6.6.5 mSATA pin signal definition

Table 19 defines the signal assignment of the mSATA connection. This connection does not support hot plug capability, so there is no connection sequence specified.

There are a total of 52 pins:

a) 5 pins for 3.3 V source;
b) 3 pins for 1.5 V source;
c) 13 pins for GND;
d) 4 pins for transmitter/receiver differential pairs;
e) 1 pin for DAS / DSS;
f) 1 pin for presence detection (PD) (see Figure 97);
g) 2 pins for Vendor Specific / Manufacturing;
h) 2 pins for Two Wire Interface;
i) 19 reserved pins (no connect) ; and
j) 1 pin to indicate mSATA use (no connect).

Table 19 – Signal assignments for mSATA (part 1 of 2)

Pin position	Type	Description
P1	Reserved	No Connect
P2	+3.3 V	3.3 V Source
P3	Reserved	No Connect
P4	GND	Ground
P5	Reserved	No Connect
P6	+1.5 V	1.5 V Source
P7	Reserved	No Connect
P8	Reserved	No Connect
P9	GND	Ground
P10	Reserved	No Connect
P11	Reserved	No Connect
P12	Reserved	No Connect
P13	Reserved	No Connect
P14	Reserved	No Connect
P15	GND	Ground
P16	Reserved	No Connect
P17	Reserved	No Connect
P18	GND	Ground
P19	Reserved	No Connect
P20	Reserved	No Connect
P21	GND	Ground
P22	Reserved	No Connect
P23	+B	Host Receiver Differential Signal Pair

[a] Presence detection pin indicates presence of an mSATA device. See Figure 97 for an informative host side bi-directional implementation for compatibility with non-mSATA devices that may use this pin as an input.
[b] No connect on the host side.
[c] Pins 30 and 32 are intended for use as a two wire interface to read a memory device to determine device information (i.e., for use as SMB bus pins). These pins are not designed to be active in conjunction with the SATA signal differential pairs.
[d] P43 to be a no connect on mSATA devices. Given that non-mSATA devices ground P43, configurable shared-socket designs may use this pin to identify mSATA and non-mSATA devices.

Table 19 – Signal assignments for mSATA (part 2 of 2)

Pin position	Type	Description
P24	+3.3 V	3.3 V Source
P25	-B	Host Receiver Differential Signal Pair
P26	GND	Ground
P27	GND	Ground
P28	+1.5 V	1.5 V Source
P29	GND	Ground
P30	Two Wire Interface	Two Wire Interface Clock [c]
P31	-A	Host Transmitter Differential Signal Pair
P32	Two Wire Interface	Two Wire Interface Data [c]
P33	+A	Host Transmitter Differential Signal Pair
P34	GND	Ground
P35	GND	Ground
P36	Reserved	No Connect
P37	GND	Ground
P38	Reserved	No Connect
P39	+3.3 V	3.3 V Source
P40	GND	Ground
P41	+3.3 V	3.3 V Source
P42	Reserved	No Connect
P43	Device Type	Shall be a No Connect on mSATA devices [d]
P44	DEVSLP	Enter/Exit DevSleep
P45	Vendor	Vendor Specific / Manufacturing Pin[b]
P46	Reserved	No Connect
P47	Vendor	Vendor Specific / Manufacturing Pin [b]
P48	+1.5 V	1.5 V Source
P49	DAS/DSS	Device Activity Signal / Disable Staggered Spinup
P50	GND	Ground
P51	Presence Detection	Shall be Connected to GND by a 0 ohm to 220 ohm Resistor on device [a]
P52	+3.3 V	3.3 V Source

[a] Presence detection pin indicates presence of an mSATA device. See Figure 97 for an informative host side bi-directional implementation for compatibility with non-mSATA devices that may use this pin as an input.

[b] No connect on the host side.

[c] Pins 30 and 32 are intended for use as a two wire interface to read a memory device to determine device information (i.e., for use as SMB bus pins). These pins are not designed to be active in conjunction with the SATA signal differential pairs.

[d] P43 to be a no connect on mSATA devices. Given that non-mSATA devices ground P43, configurable shared-socket designs may use this pin to identify mSATA and non-mSATA devices.

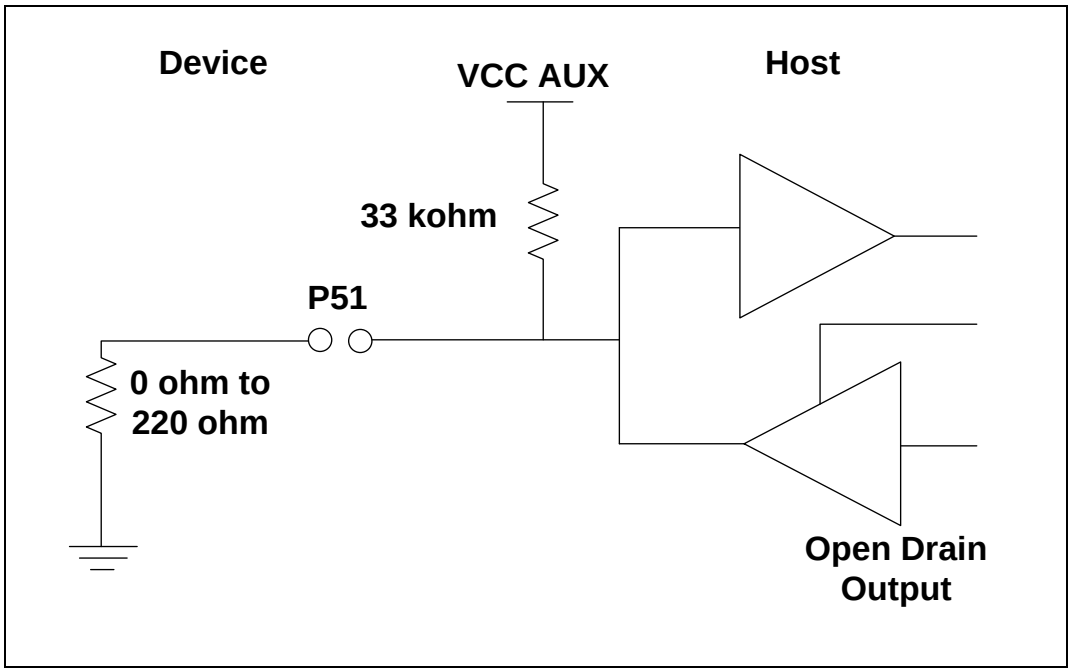


Figure 97 – Bi-directional host side implementation of P51 for compatibility with non-mSATA devices (informative)

6.7 SATA USM connector

6.7.1 SATA USM connector location

The SATA USM connector location is defined to facilitate blind mating. Figure 98 and Table 20 show the connector location on the 14.5 mm SATA USM device. Figure 99 and Table 21 show the connector location on the 9 mm SATA USM device. The Serial ATA connector is located inside the SATA USM housing as indicated. See INF-8280 for additional details regarding the SATA USM.

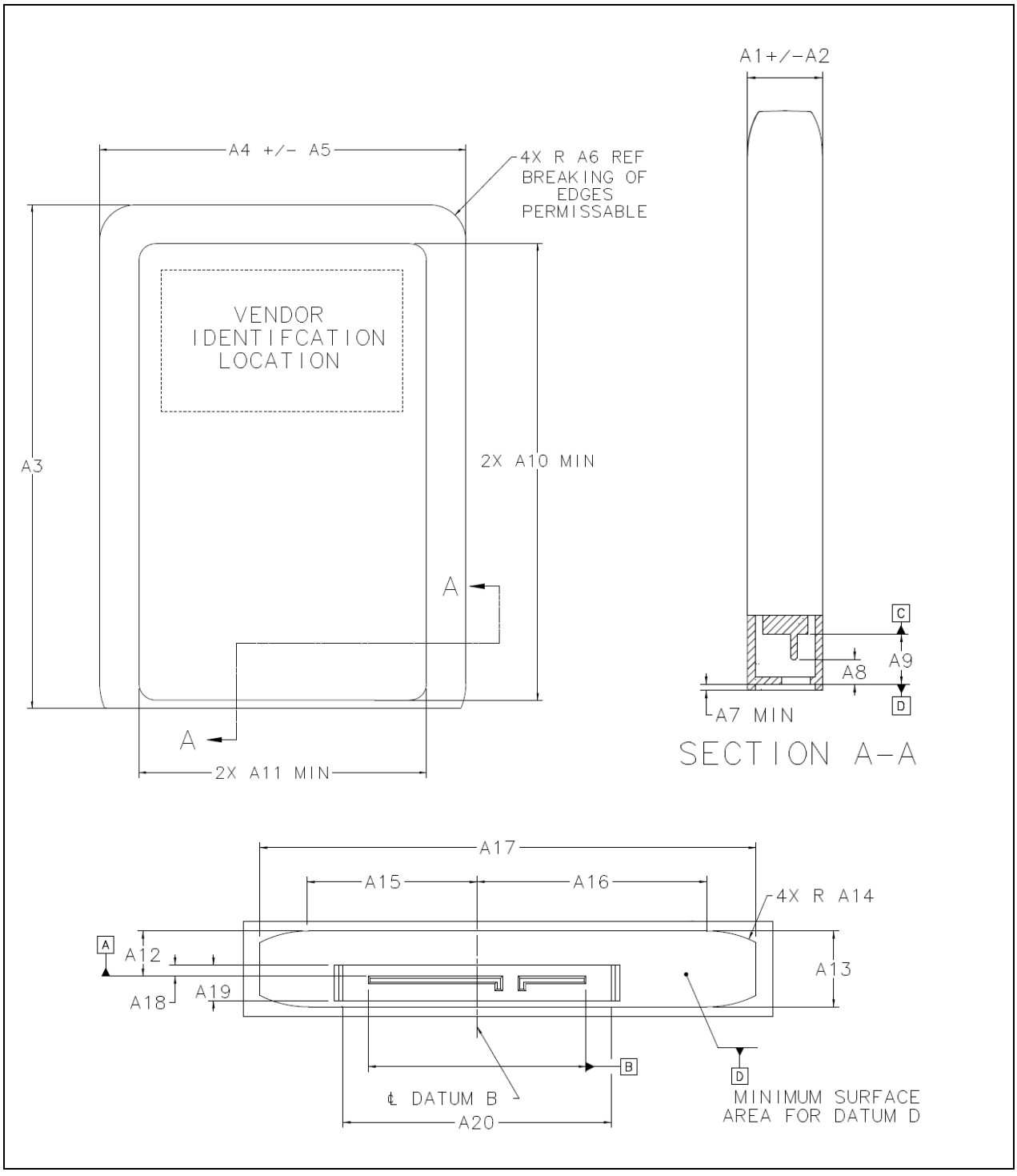


Figure 98 – 14.5 mm SATA USM physical dimensions (see INF-8280)

Table 20 – 14.5 mm SATA USM physical dimensions (see INF-8280)

Dimension	mm	inch
A1	14.50	0.571
A2	0.20	0.008
A3	111.68	4.397
A4	81.10	3.193
A5	0.20	0.008
A6	8.00	0.315
A7	1.14	0.045
A8	4.80	0.189
A9	9.70	0.382
A10	88.90	3.500
A11	50.80	2.000
A12	6.86	0.270
A13	11.53	0.454
A14	16.00	0.630
A15	26.04	1.025
A16	35.20	1.386
A17	76.02	2.993
A18	1.68	0.066
A19	5.46	0.215
A20	41.05	1.616

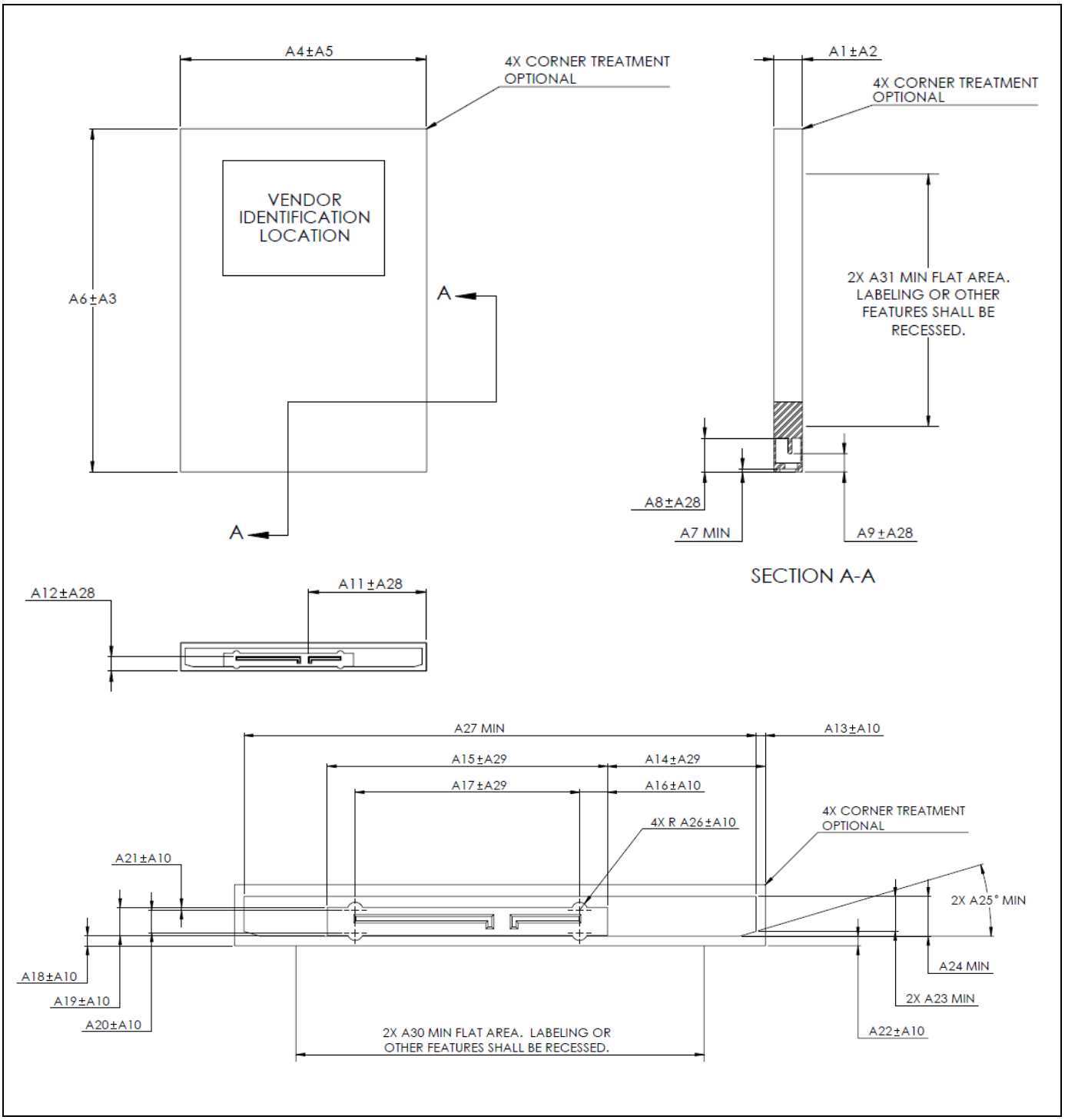


Figure 99 – 9 mm SATA USM physical dimensions (see INF-8280)

Table 21 – 9 mm SATA USM physical dimensions (see INF-8280)

Dimension	mm	inch
A1	9.00	0.354
A2	0.20	0.008
A3	1.0	0.039
A4	78.10	3.075
A5	0.20	0.008
A6	108.7	4.281
A7	1.00	0.039
A8	10.70	0.421
A9	5.80	0.228
A10	0.10	0.004
A11	37.55	1.478
A12	4.60	0.181
A13	1.41	0.056
A14	23.23	0.915
A15	41.22	1.623
A16	4.11	0.162
A17	33.00	1.299
A18	1.50	0.059
A19	4.13	0.163
A20	3.36	0.132
A21	0.36	0.014
A22	1.41	0.056
A23	5.14	0.202
A2	5.85	0.230
A25	15	15
A26	1.13	0.044
A27	75.28	2.964
A28	0.30	0.012
A29	0.20	0.008
A30	60.0	2.362
A31	80.00	3.150

6.7.2 USM mating interfaces

The USM device uses the device plug connector as defined in 6.2.3.1. The SATA USM receiver uses a custom SATA receptacle connector to properly attach to the module. The connectors shall be capable of a minimum of 1 500 insertion/removal cycles. The SATA USM receptacle connector is available in vertical and horizontal PCB mounting configurations as shown in Figure 100, Figure 101, Figure 102 and Figure 103.

The receptacle connector mating area is compliant with the backplane connector as defined in 6.2.6 with the following four exceptions:

a) side mounted retention springs to improve the connector retention (optional);
b) anti-wiggle bumps to reduce cable deflection (optional);
c) two alignment ribs on each of the long outside surfaces (required); and
d) extended reach/length to properly attach the SATA universal storage module (required).

NOTE 13 - For hot plug implementation, note that the side mounted retention springs are optional and that the only contacts present are the power and signal pins defined in Table 5.

See INF-8280 for USM receptacle connector details.

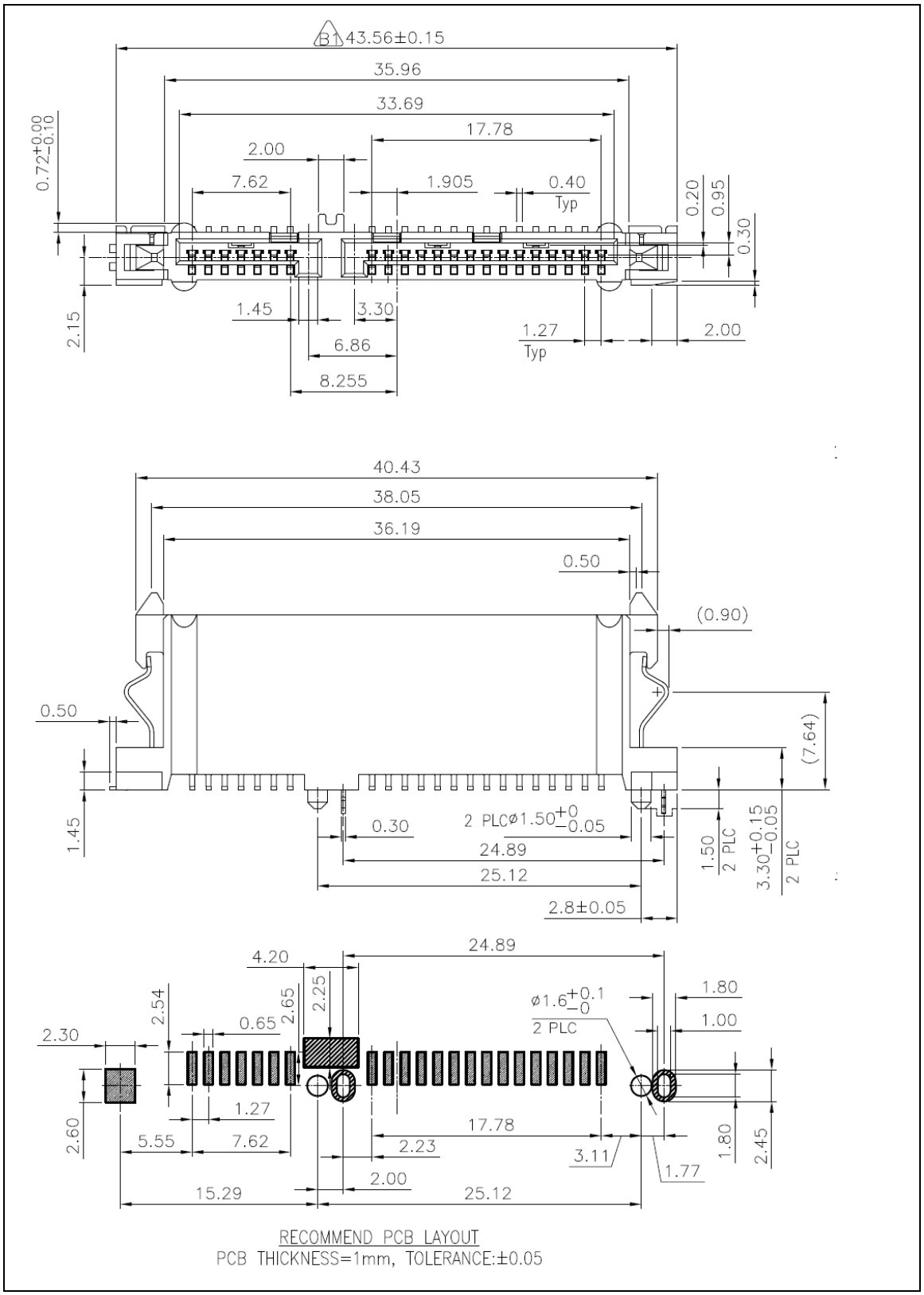


Figure 100 – SATA USM vertical receptacle (see INF-8280)

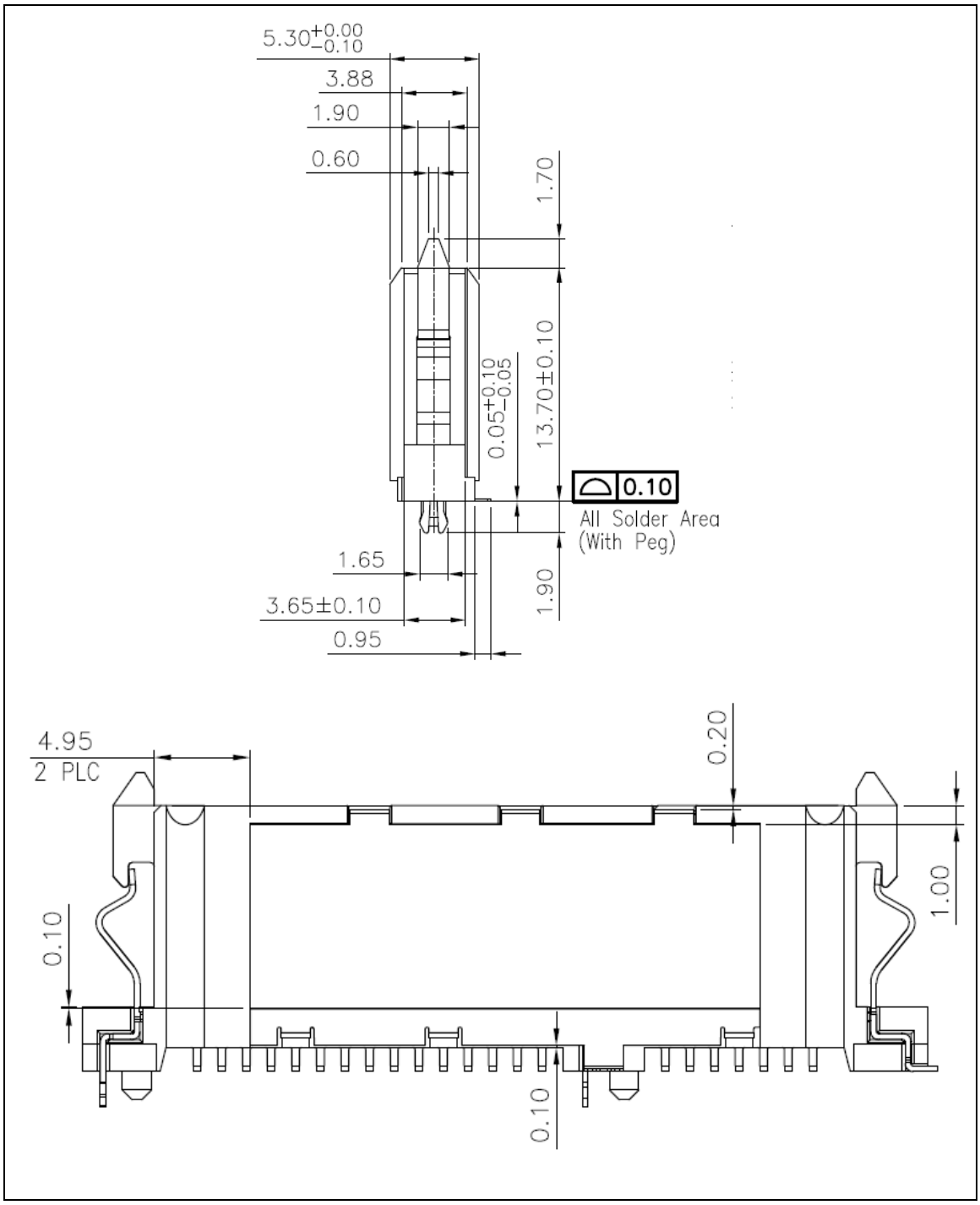


Figure 101 – SATA USM vertical receptacle continued (see INF-8280)

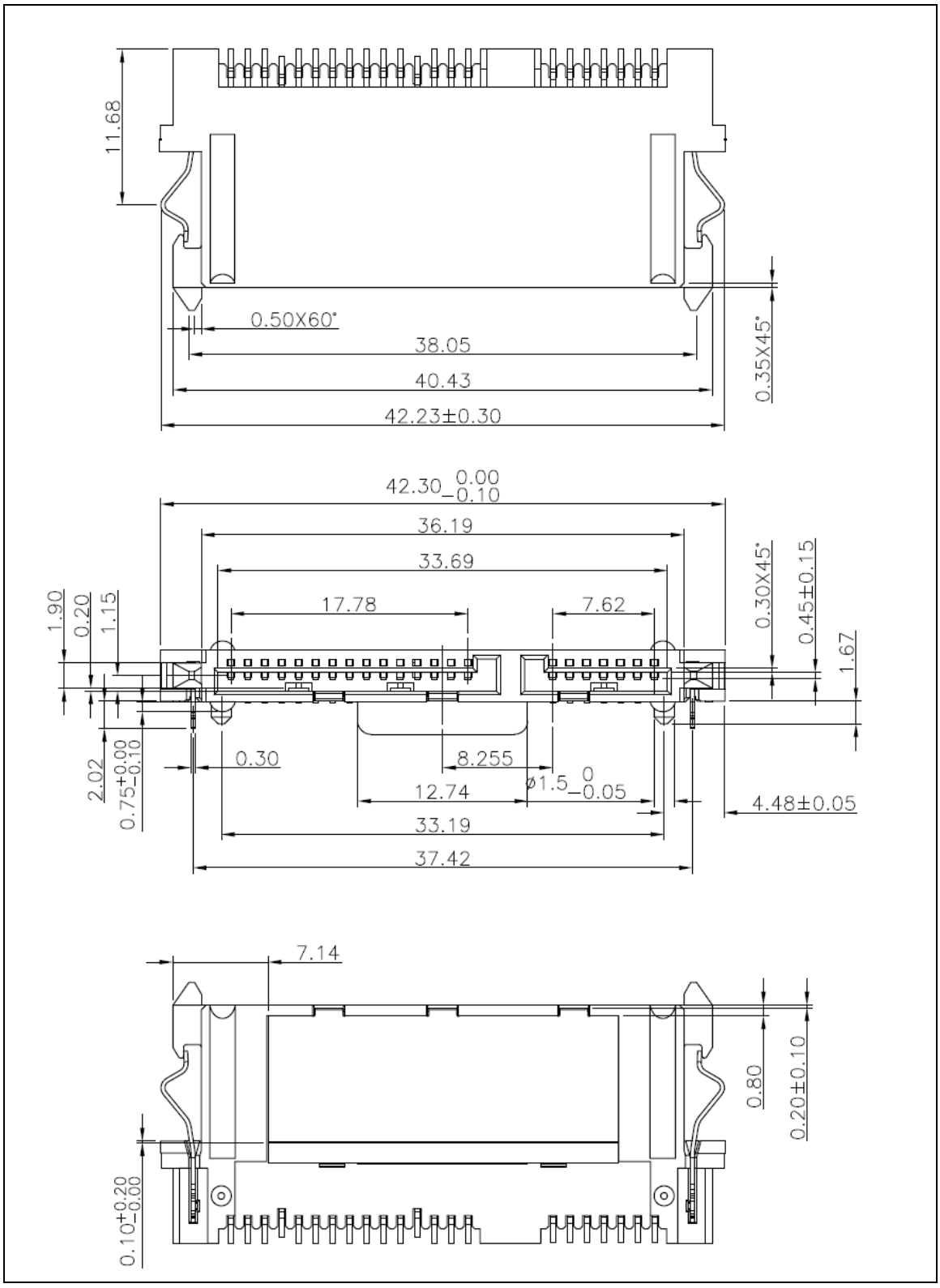


Figure 102 – SATA USM horizontal receptacle (see INF-8280)

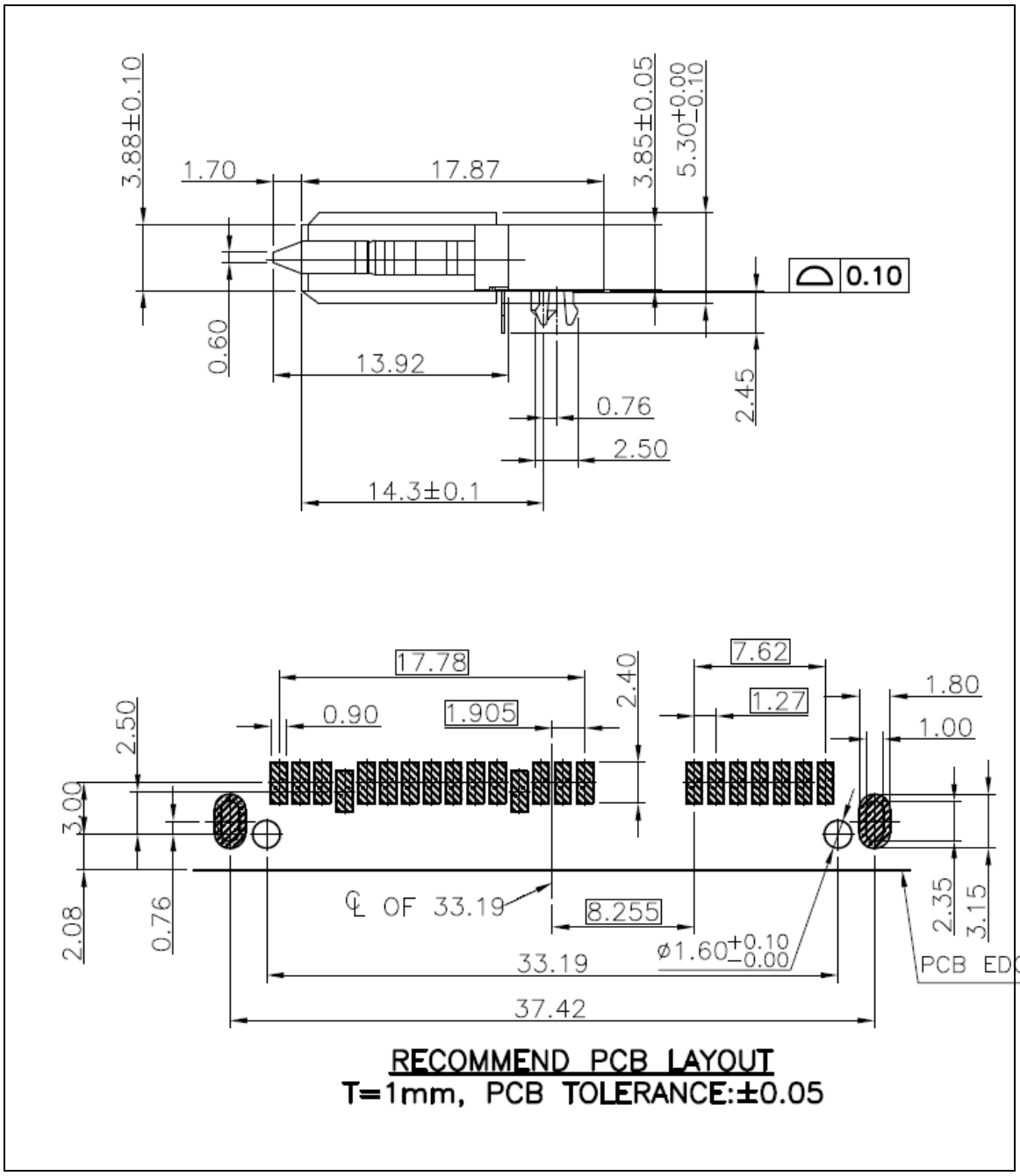


Figure 103 – SATA USM horizontal receptacle continued (see INF-8280)

6.8 SATA MicroSSD interface

6.8.1 SATA MicroSSD interface scope

This section defines the mechanical properties of the SATA MicroSSD device and device interface.

6.8.2 SATA MicroSSD mechanical specification

A SATA MicroSSD shall use package variants:

a) AC (see Figure 104);
b) AM (see Figure 107);
c) AK (see Figure 105);
d) AL (see Figure 106);
e) AR (see Figure 108);
f) CA (see Figure 107);
g) CB (see Figure 105);
h) DB (see Figure 108); or
i) DC (see Figure 108),

as defined in the JEDEC document “MO-276, Standard Profile and Low Profile Rectangular Fine Pitch Ball Grid Array Family, 0.50 mm Pitch”, revision E or later.

The rest of this section is informative only, refer to JEDEC MO-276E (or later) for the formal mechanical specification of SATA MicroSSD packages.

The specified package variants for a SATA MicroSSD use five distrinct package footprints. The package footprints (bottom view) are shown below. A “+” sign denotes a depopulated ball position. The functional balls (e.g., signals or power) for each footprint are the inner 3 or 4 rows on each footprint (i.e., square rows); the rest of the populated balls are mechanical only, for package stability. The normative signal assignment for the functional balls as defined in 6.8.3, SATA MicroSSD Ballout.

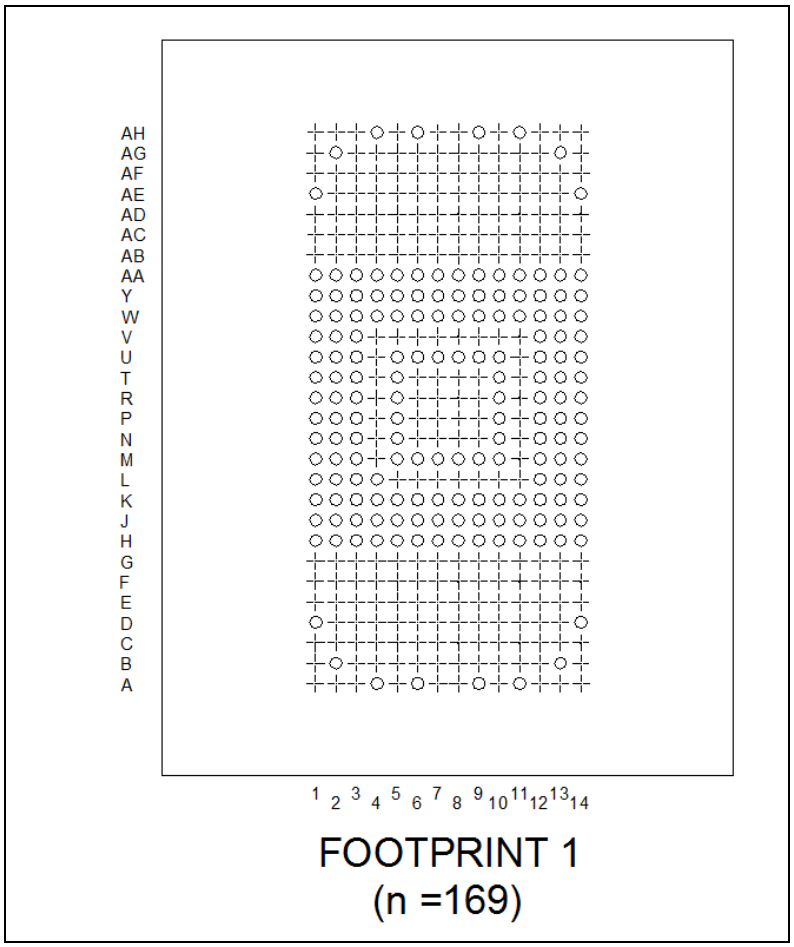


Figure 104 – Footprint 1, SATA MicroSSD variant AC, 169 balls (informative)

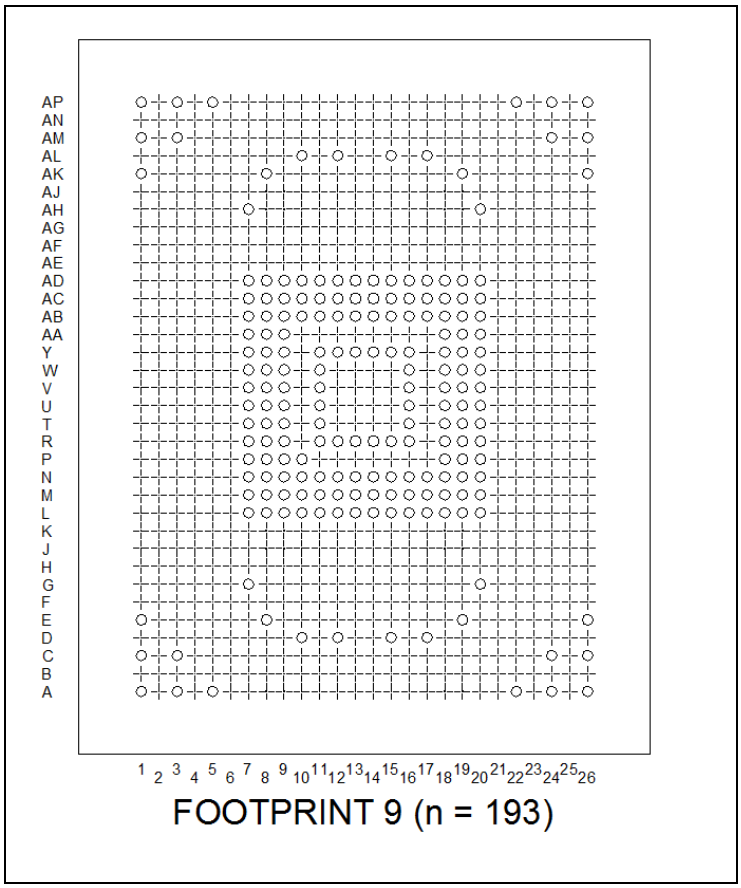


Figure 105 – Footprint 9, SATA MicroSSD variant AK and CB, 193 balls (informative)

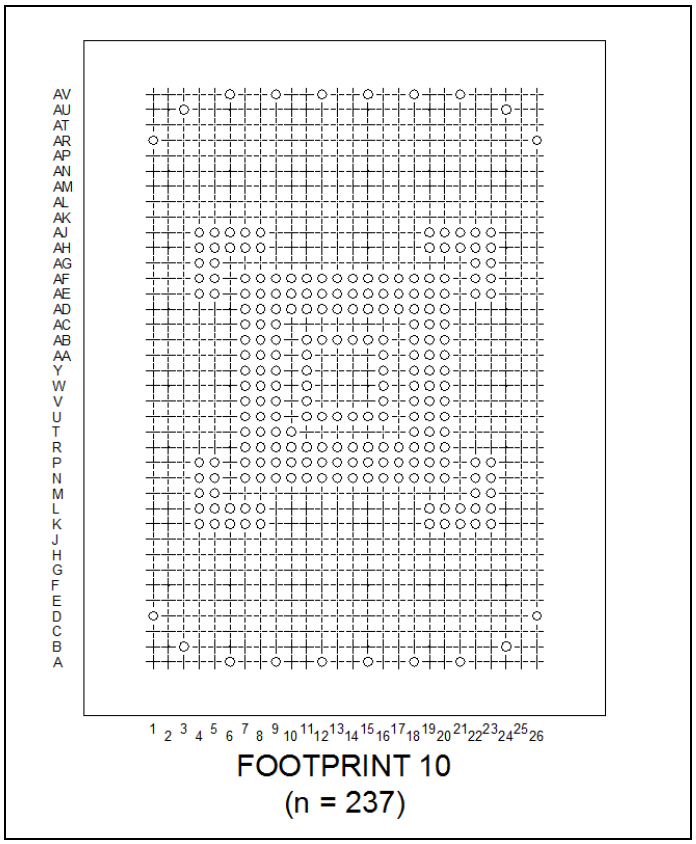


Figure 106 – Footprint 10, SATA MicroSSD variant AL, 237 balls (informative)

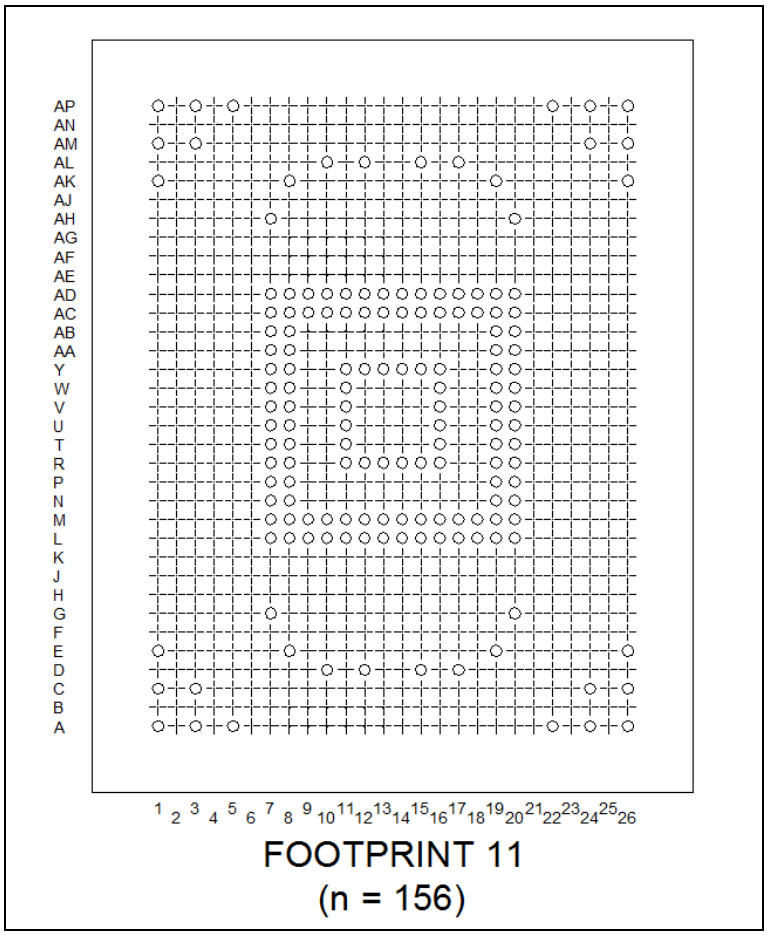


Figure 107 – Footprint 11, SATA MicroSSD variant AM and CA, 156 balls (informative)

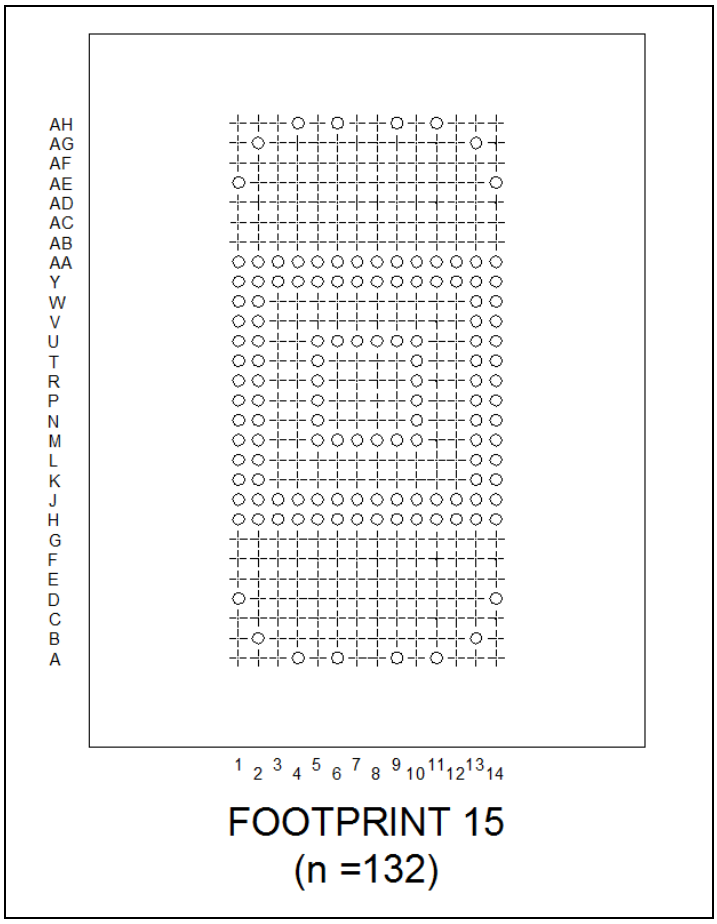


Figure 108 – Footprint 15, SATA MicroSSD variant AR, DB, and DC, 132 balls (informative)

6.8.3 SATA MicroSSD ballout - functional signal definition

Table 22 defines the signal assignment of the SATA MicroSSD connection for each of the package types defined in 6.8.2.

Some of the general catigories are:

a) VSP, vendor specific;
b) Reserved, reserved for future standardization (see 4.2.3.10);
c) Do Not Use (DNU), internal test only; shall not be connected on host; and
d) TEST, may be connected for test/diagnostic use, but not used during normal device operation.

Table 22 – Signal assignments for SATA MicroSSD (part 1 of 5)

BALL ASSIGNMENT # Balls (Footprint #)					BALL NAME	TYPE	DESCRIPTIONS
132 (15)	156 (11)	169 (1)	193 (9)	237 (10)			
SATA Interface Signals							
M1	R7	M1	R7	U7	SATA_RX_N	Input	Differential Signal Pair A (SATA Device Receive Signal Differential Pair)
L1	P7	L1	P7	T7	SATA_RX_P		
P1	U7	P1	U7	W7	SATA_TX_N	Output	Differential Signal Pair B (SATA Device Transmit Signal Differential Pair)
R1	V7	R1	V7	Y7	SATA_TX_P		
J7	M13	J7	M13	P13	DAS	Output	Device Activity Signal
P2	U8	P2	U8	W8	SATA_VCC	Supply	+3.3 V
R2	V8	R2	V8	Y8	SATA_VCC	Supply	+3.3 V
L2	P8	L2	P8	T8	SATA_VDD	Supply	+1.2 V
M2	R8	M2	R8	U8	SATA_VDD	Supply	+1.2 V
K1	N7	K1	N7	R7	SATA_VSS	GND	Signal Ground
N1	T7	N1	T7	V7	SATA_VSS	GND	Signal Ground
T1	W7	T1	W7	AA7	SATA_VSS	GND	Signal Ground
Optional Signals							
AA10	AD16	AA10	AD16	AF16	SPI_MISO	Input	MISO
AA9	AD15	AA9	AD15	AF15	SPI_CS0	Output	Chip Select
Y10	AC16	Y10	AC16	AE16	SPI_CLK	Output	Clock
AA12	AD18	AA12	AD18	AF18	SPI_CS1	Output	Chip Select
AA11	AD17	AA11	AD17	AF17	SPI_MOSI	Ouput	MOSI
V2	AA8	V2	AA8	AC8	TEST		
H9	L15	H9	L15	N15	TEST		
H10	L16	H10	L16	N16	TEST		
H11	L17	H11	L17	N17	TEST		
H12	L18	H12	L18	N18	TEST		
V14	AA20	V14	AA20	AC20	VSP		
J6	M12	J6	M12	P12	VSP		
Y3	AC9	Y3	AC9	AE9	DEVSLP	Input	Enter/Exit DevSleep
Y5	AC11	Y5	AC11	AE11	Reserved		Future Low Power
H7	L13	H7	L13	N13	Reserved		Future Low Power

[a] Optional pin reserved for system clock output to drive crystal or other system requirements. Frequency is system implementation dependent.
[b] Optional pin reserved for system clock input. Frequency is system implementation dependent.
[c] Optional pin reserved for Hardware Reset. Implementation is system dependent. For detailed timing information, consult device data sheet.

Table 22 – Signal assignments for SATA MicroSSD (part 2 of 5)

BALL ASSIGNMENT # Balls (Footprint #)					BALL NAME	TYPE	DESCRIPTIONS
132 (15)	156 (11)	169 (1)	193 (9)	237 (10)			
Control Signals							
H3	L9	H3	L9	N9	XTAL_OUT	Output	System Clock output [a]
J4	M10	J4	M10	P10	XTAL_IN	Input	System Clock input [b]
J3	M9	J3	M9	P9	PWR_RESETN	Input	Hardware Reset[c]
Power Supply Signals							
M7	R13	M7	R13	U13	VCC	Supply	+3.3 V
M8	R14	M8	R14	U14	VCC	Supply	+3.3 V
M9	R15	M9	R15	U15	VCC	Supply	+3.3 V
M10	R16	M10	R16	U16	VCC	Supply	+3.3 V
N10	T16	N10	T16	V16	VCC	Supply	+3.3 V
P10	U16	P10	U16	W16	VCC	Supply	+3.3 V
J5	M11	J5	M11	P11	VCC	Supply	+3.3 V
H6	L12	H6	L12	N12	VCC	Supply	+3.3 V
M13	R19	M13	R19	U19	VCC	Supply	+3.3 V
M14	R20	M14	R20	U20	VCC	Supply	+3.3 V
R5	V11	R5	V11	Y11	VCC	Supply	+3.3 V
U14	Y20	U14	Y20	AB20	VCC	Supply	+3.3 V
U13	Y19	U13	Y19	AB19	VCC	Supply	+3.3 V
V13	AA19	V13	AA19	AC19	VCC	Supply	+3.3 V
Y2	AC8	Y2	AC8	AE8	VCC	Supply	+3.3 V
R10	V16	R10	V16	Y16	VCCQ	Supply	+1.8 V
T10	W16	T10	W16	AA16	VCCQ	Supply	+1.8 V
U10	Y16	U10	Y16	AB16	VCCQ	Supply	+1.8 V
T5	W11	T5	W11	AA11	VDDC	Supply	+1.2 V
U5	Y11	U5	Y11	AB11	VDDC	Supply	+1.2 V
U6	Y12	U6	Y12	AB12	VDDC	Supply	+1.2 V
U7	Y13	U7	Y13	AB13	VDDC	Supply	+1.2 V
M5	R11	M5	R11	U11	VDD	Supply	+1.2 V
N5	T11	N5	T11	V11	VDD	Supply	+1.2 V

[a] Optional pin reserved for system clock output to drive crystal or other system requirements. Frequency is system implementation dependent.
[b] Optional pin reserved for system clock input. Frequency is system implementation dependent.
[c] Optional pin reserved for Hardware Reset. Implementation is system dependent. For detailed timing information, consult device data sheet.

Table 22 – Signal assignments for SATA MicroSSD (part 3 of 5)

BALL ASSIGNMENT # Balls (Footprint #)					BALL NAME	TYPE	DESCRIPTIONS
132 (15)	156 (11)	169 (1)	193 (9)	237 (10)			
GND signals							
M6	R12	M6	R12	U12	VSS	GND	Ground
P5	U11	P5	U11	W11	VSS	GND	Ground
H1	L7	H1	L7	N7	VSS	GND	Ground
H2	L8	H2	L8	N8	VSS	GND	Ground
J1	M7	J1	M7	P7	VSS	GND	Ground
H5	L11	H5	L11	N11	VSS	GND	Ground
H13	L19	H13	L19	N19	VSS	GND	Ground
H14	L20	H14	L20	N20	VSS	GND	Ground
J13	M19	J13	M19	P19	VSS	GND	Ground
J14	M20	J14	M20	P20	VSS	GND	Ground
K13	N19	K13	N19	R19	VSS	GND	Ground
L13	P19	L13	P19	T19	VSS	GND	Ground
J14	AC20	Y14	AC20	AE20	VSS	GND	Ground
AA14	AD20	AA14	AD20	AF20	VSS	GND	Ground
AA13	AD19	AA13	AD19	AF19	VSS	GND	Ground
AA2	AD8	AA2	AD8	AF8	VSS	GND	Ground
AA1	AD7	AA1	AD7	AF7	VSS	GND	Ground
N2	T8	N2	T8	V8	VSS	GND	Ground
U8	Y14	U8	Y14	AB14	VSS	GND	Ground
U9	Y15	U9	Y15	AB15	VSS	GND	Ground
P13	U19	P13	U19	W19	VSS	GND	Ground
L14	P20	L14	P20	T20	VSS	GND	Ground
P14	U20	P14	U20	W20	VSS	GND	Ground
R13	V19	R13	V19	Y19	VSS	GND	Ground
Y1	AC7	Y1	AC7	AE7	VSS	GND	Ground
W1	AB7	W1	AB7	AD7	VSS	GND	Ground
K2	N8	K2	N8	R8	VSS	GND	Ground
Do Not Use (DNU)							
AA5	AD11	AA5	AD11	AF11	DNU		
AA3	AD9	AA3	AD9	AF9	DNU		
Y7	AC13	Y7	AC13	AE13	DNU		
AA7	AD13	AA7	AD13	AF13	DNU		
Y8	AC14	Y8	AC14	AE14	DNU		
J10	M16	J10	M16	P16	DNU		

Table 22 – Signal assignments for SATA MicroSSD (part 4 of 5)

BALL ASSIGNMENT # Balls (Footprint #)					BALL NAME	TYPE	DESCRIPTIONS
132 (15)	156 (11)	169 (1)	193 (9)	237 (10)			
Reserved							
J8	M14	J8	M14	P14	Reserved		
J2	M8	J2	M8	P8	Reserved		
H4	L10	H4	L10	N10	Reserved		
H8	L14	H8	L14	N14	Reserved		
N13	T19	N13	T19	V19	Reserved		
R14	V20	R14	V20	Y20	Reserved		
N14	T20	N14	T20	V20	Reserved		
K14	N20	K14	N20	R20	Reserved		
J11	M17	J11	M17	P17	Reserved		
J12	M18	J12	M18	P18	Reserved		
W2	AB8	W2	AB8	AD8	Reserved		
W13	AB19	W13	AB19	AD19	Reserved		
W14	AB20	W14	AB20	AD20	Reserved		
Y11	AC17	Y11	AC17	AE17	Reserved		
AA8	AD14	AA8	AD14	AF14	Reserved		
Y9	AC15	Y9	AC15	AE15	Reserved		
AA6	AD12	AA6	AD12	AF12	Reserved		
Y13	AC19	Y13	AC19	AE19	Reserved		
Y12	AC18	Y12	AC18	AE18	Reserved		
T14	W20	T14	W20	AA20	Reserved		
T13	W19	T13	W19	AA19	Reserved		
V1	AA7	V1	AA7	AC7	Reserved		
U1	Y7	U1	Y7	AB7	Reserved		
T2	W8	T2	W8	AA8	Reserved		
U2	Y8	U2	Y8	AB8	Reserved		
AA4	AD10	AA4	AD10	AF10	Reserved		
Y6	AC12	Y6	AC12	AE12	Reserved		
J9	M15	J9	M15	P15	Reserved		
Y4	AC10	Y4	AC10	AE10	Reserved		
		K3	N9	R9	Reserved		
		L3	P9	T9	Reserved		
		M3	R9	U9	Reserved		
		N3	T9	V9	Reserved		
		P3	U9	W9	Reserved		
		R3	V9	Y9	Reserved		
		T3	W9	AA9	Reserved		
		U3	Y9	AB9	Reserved		

Table 22 – Signal assignments for SATA MicroSSD (part 5 of 5)

BALL ASSIGNMENT # Balls (Footprint #)					BALL NAME	TYPE	DESCRIPTIONS
132 (15)	156 (11)	169 (1)	193 (9)	237 (10)			
Reserved (Continued)							
		V3	AA9	AC9	Reserved		
		W3	AB9	AD9	Reserved		
		K12	N18	R18	Reserved		
		L12	P18	T18	Reserved		
		M12	R18	U18	Reserved		
		N12	T18	V18	Reserved		
		P12	U18	W18	Reserved		
		R12	V18	Y18	Reserved		
		T12	W18	AA18	Reserved		
		U12	Y18	AB18	Reserved		
		V12	AA18	AC18	Reserved		
		W12	AB18	AD18	Reserved		
		K4	N10	R10	Reserved		
		K5	N11	R11	Reserved		
		K6	N12	R12	Reserved		
		K7	N13	R13	Reserved		
		K8	N14	R14	Reserved		
		K9	N15	R15	Reserved		
		K10	N16	R16	Reserved		
		K11	N17	R17	Reserved		
		W4	AB10	AD10	Reserved		
		W5	AB11	AD11	Reserved		
		W6	AB12	AD12	Reserved		
		W7	AB13	AD13	Reserved		
		W8	AB14	AD14	Reserved		
		W9	AB15	AD15	Reserved		
		W10	AB16	AD16	Reserved		
		W11	AB17	AD17	Reserved		
		L4	P10	T10	Reserved		

[a] Optional pin reserved for system clock output to drive crystal or other system requirements. Frequency is system implementation dependent.
[b] Optional pin reserved for system clock input. Frequency is system implementation dependent.
[c] Optional pin reserved for Hardware Reset. Implementation is system dependent. For detailed timing information, consult device data sheet.

6.9 Internal M.2 connector

6.9.1 Internal M.2 connector overview

This section defines the requirements of an M.2 connector with support for SATA as well as PCI Express signaling.

This board format is specifically designed to match commonly used SSD memory components to ensure maximum use of circuit board area.

The definition supports the following capabilities:

a) SATA transfer rates:
   A) Gen1 (i.e., 1.5 Gbit/s);
   B) Gen2 (i.e., 3.0 Gbit/s); and
   C) Gen3 (i.e., 6.0 Gbit/s);
   and
b) PCI Express:
   A) V1 (i.e., 2.5 GT/s per lane);
   B) V2 (i.e., 5 GT/s per lane); and
   C) V3 (i.e., 8 GT/s per lane).

In addition this format supports the following concepts:

a) CONFIG pins that are set by the SSD to inform the host if the drive wishes to use the SATA or PCIe signaling scheme, as well as informing the system if a card is present and if the card is an SSD or another type of device;
b) DEVSLP (i.e., device sleep) that is a pin that the host may drive to inform an SSD that it should enter into a low power mode (if possible); and
c) MFG1/MFG2, two vendor pins for drive or SSD manufacturing usage.

6.9.2 M.2 mechanical (informative)

For SSD devices, the M.2 specification describes in detail a set of module sizes (e.g., 22 mm x 42 mm, 22 mm x 80 mm), connector heights (e.g., 2.25 mm, 2.75 mm, and 4.2 mm), and keying options (see Figure 111 to Figure 113) for use in M.2 SSD modules.

A SATA device built to meet the M.2 form factor specification shall use the module sizes and connector/key combinations as described in the M.2 specification (see PCIe M.2). Implementers should refer directly to the M.2 specification for detailed normative mechanical specifications for M.2-based SATA devices.

The internal M.2 connector is to be used for embedded applications (see Table 1, Table 2, and 5.3.11.8).

6.9.3 M.2 board connector (informative)

M.2 uses a dual-sided edge card connector with a 0.5 mm contact pitch (see Figure 109 and Figure 110). The connector provides for 75 pin locations.

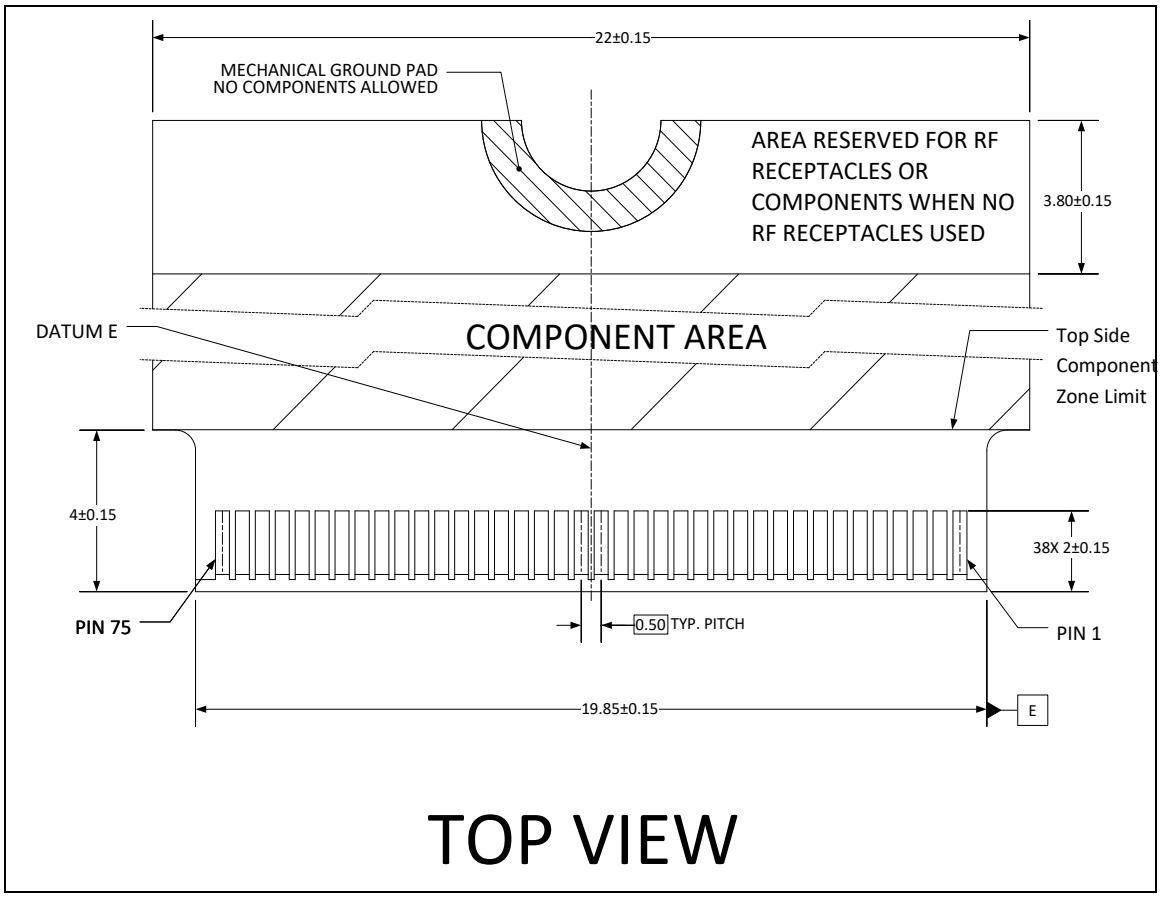


Figure 109 – M.2 board connector top details (informative)

The reserved area for radio frequency (RF) receptacles is only applicable if a SATA device is planned to be used in a shared use socket (see PCIe M.2). Components may be in this area for a SATA device, however be aware that a system may have antenna wires in this area.

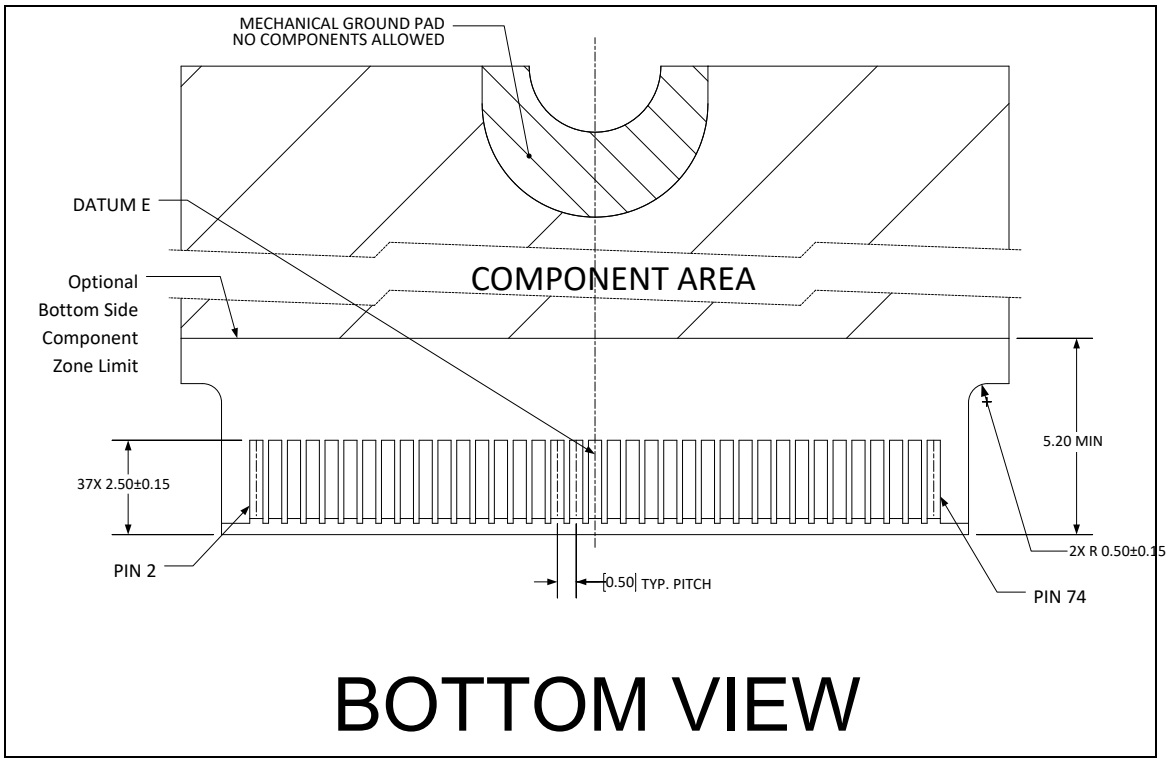
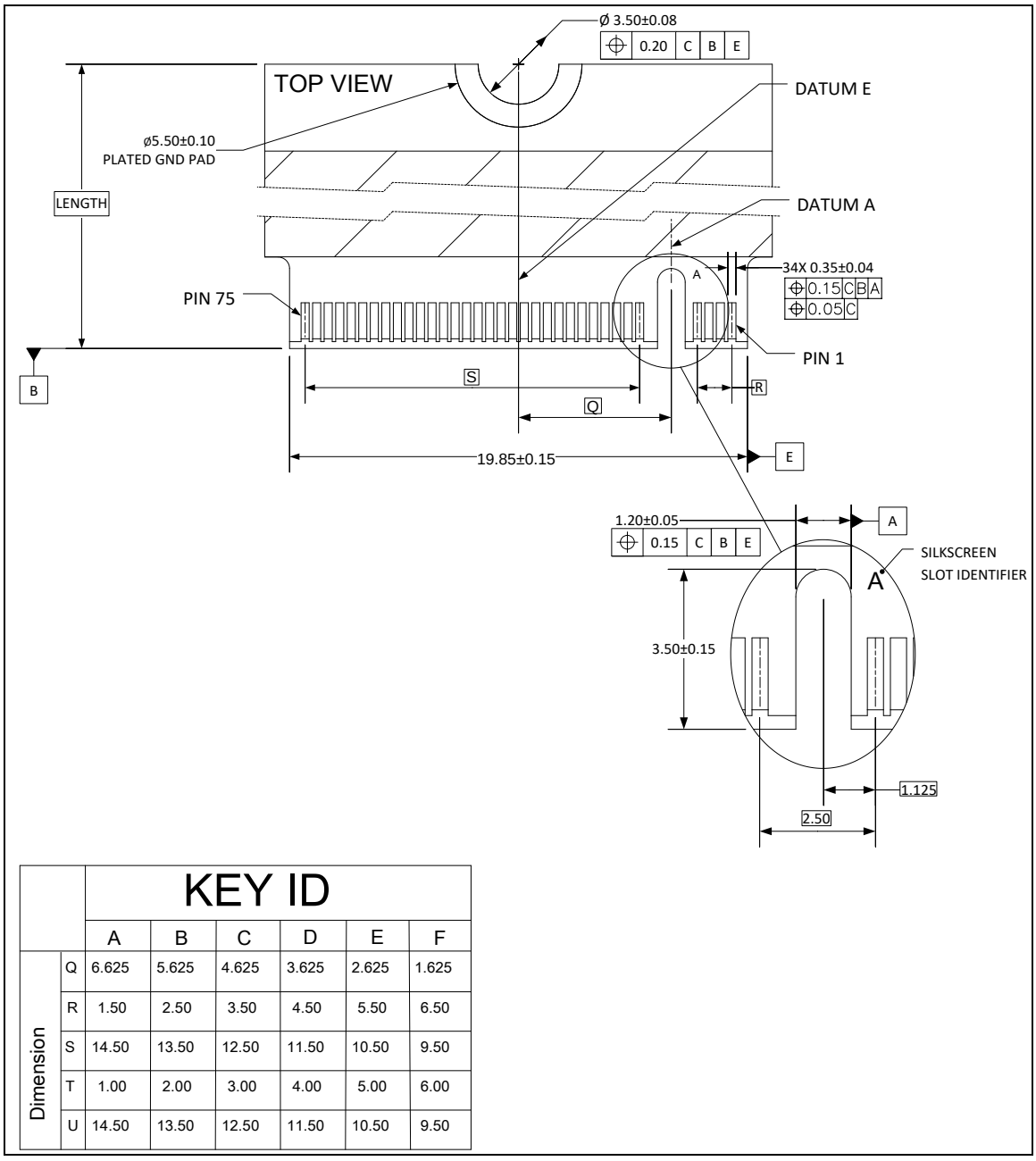
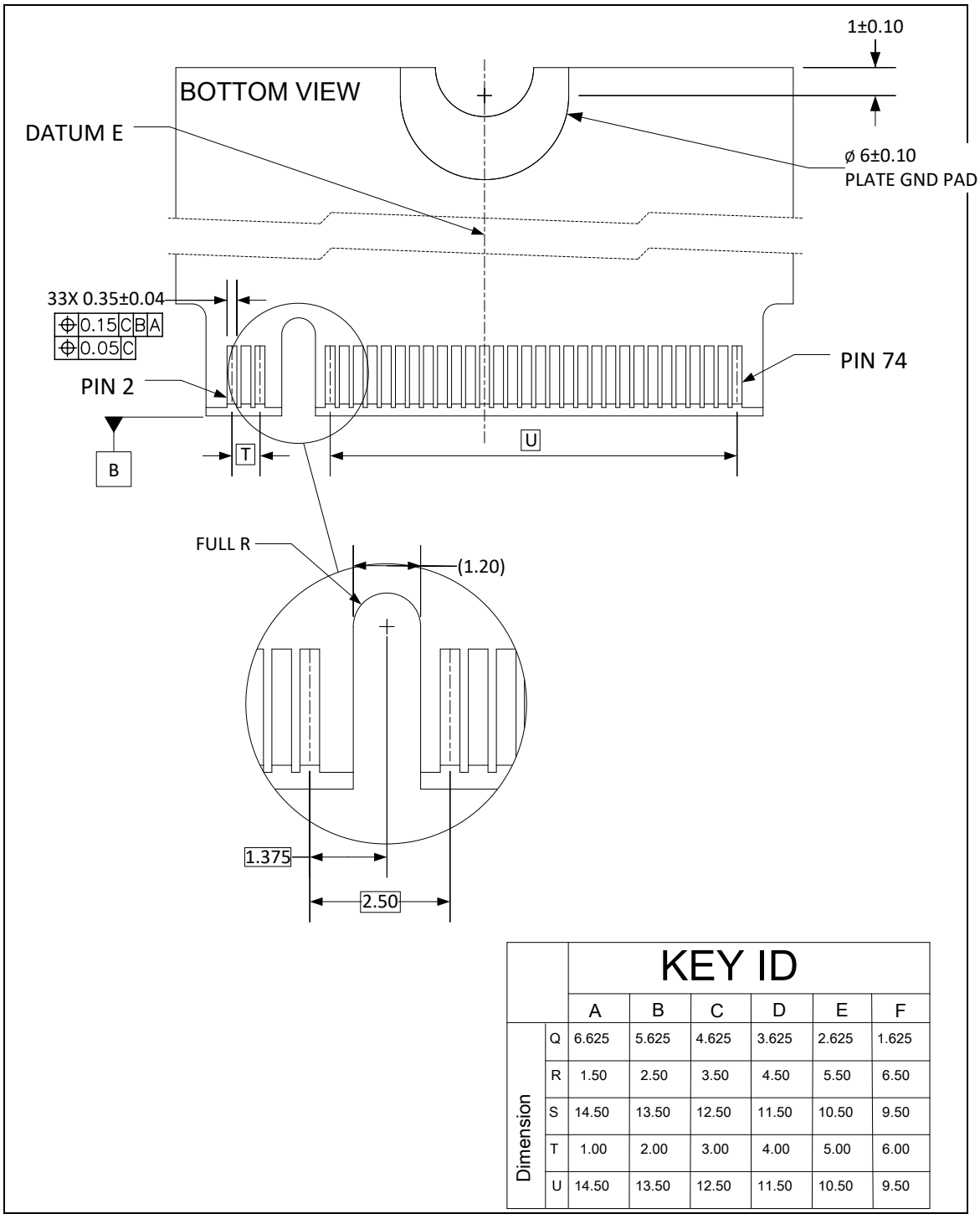


Figure 110 – M.2 board connector bottom details (informative)



		KEY ID					
		A	B	C	D	E	F
Dimension	Q	6.625	5.625	4.625	3.625	2.625	1.625
	R	1.50	2.50	3.50	4.50	5.50	6.50
	S	14.50	13.50	12.50	11.50	10.50	9.50
	T	1.00	2.00	3.00	4.00	5.00	6.00
	U	14.50	13.50	12.50	11.50	10.50	9.50

Figure 111 – M.2 board connector top slot details (informative)



		KEY ID					
		A	B	C	D	E	F
Dimension	Q	6.625	5.625	4.625	3.625	2.625	1.625
	R	1.50	2.50	3.50	4.50	5.50	6.50
	S	14.50	13.50	12.50	11.50	10.50	9.50
	T	1.00	2.00	3.00	4.00	5.00	6.00
	U	14.50	13.50	12.50	11.50	10.50	9.50

Figure 112 – M.2 board connector bottom slot details (informative)

6.9.4 M.2 keys (informative)

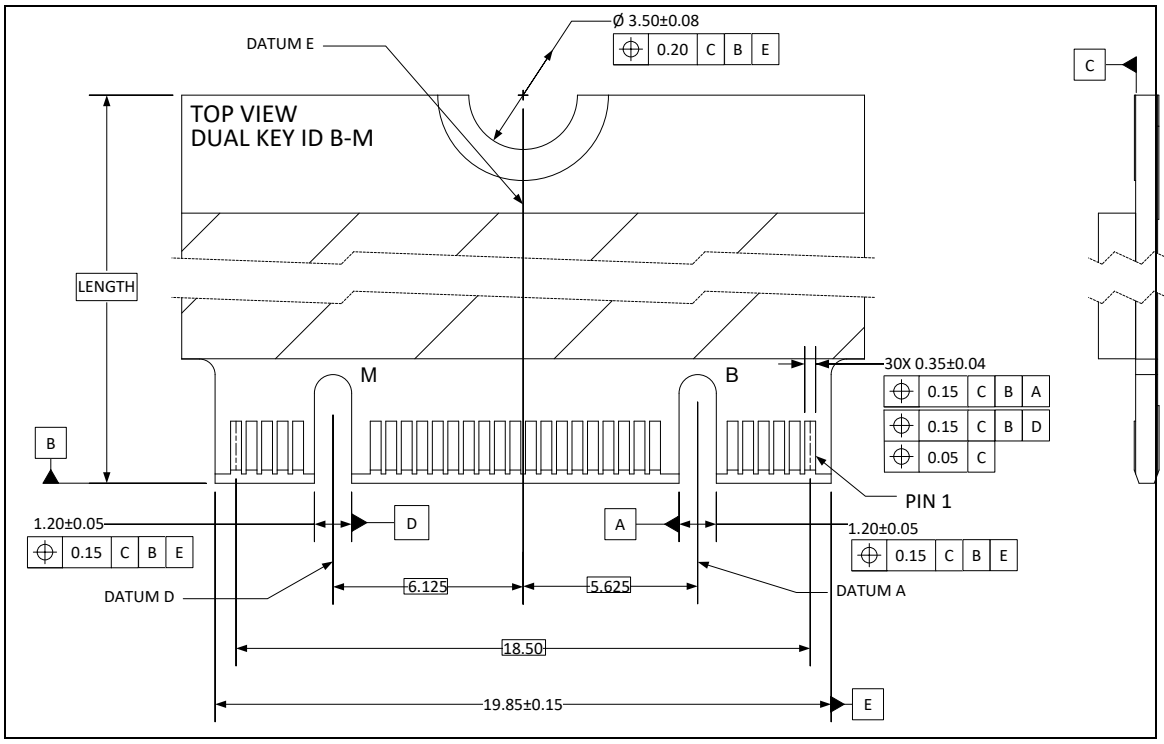


Figure 113 – M.2 keys (informative)

There are two mechanical keys defined for SSDs:

a) Key B pinout supports SSD/WWAN/Others:
   A) 1x SATA SSD; or
   B) 1x, 2x PCIe SSD (and WWAN) host interfaces;

   and

b) Key M pinout supports SSDs only:
   A) 1x SATA; or
   B) 1x, 2x, or 4x PCIe host interfaces.

Notch Location for Key B - Pins 12 to 19.
Notch Location for Key M - Pins 59 to 66.

SSD solutions targeting Key B host interface set should also employ two notches that coincide with Key B and Key M to enable these to be pluggable into both Socket 2 and Socket 3 (see 6.9.5). SSD solutions targeting Key M host interface set should only emply Key M notch and is only able to plug into Socket 3. It is not possible to plug a Key M only device into Socket 2 (with Key B).

6.9.5 M.2 sockets (informative)

Sockets are defined as follows:

a) Socket 1 accepts cards with an “A” key notch present;
b) Socket 2 accepts cards with a “B” key notch present; and
c) Socket 3 accepts cards with an “M” key notch present.

6.9.6 M.2 land pattern for top mount connector motherboard (informative)

Figure 114 is the informative M.2 land pattern for a top mount connector on a motherboard.

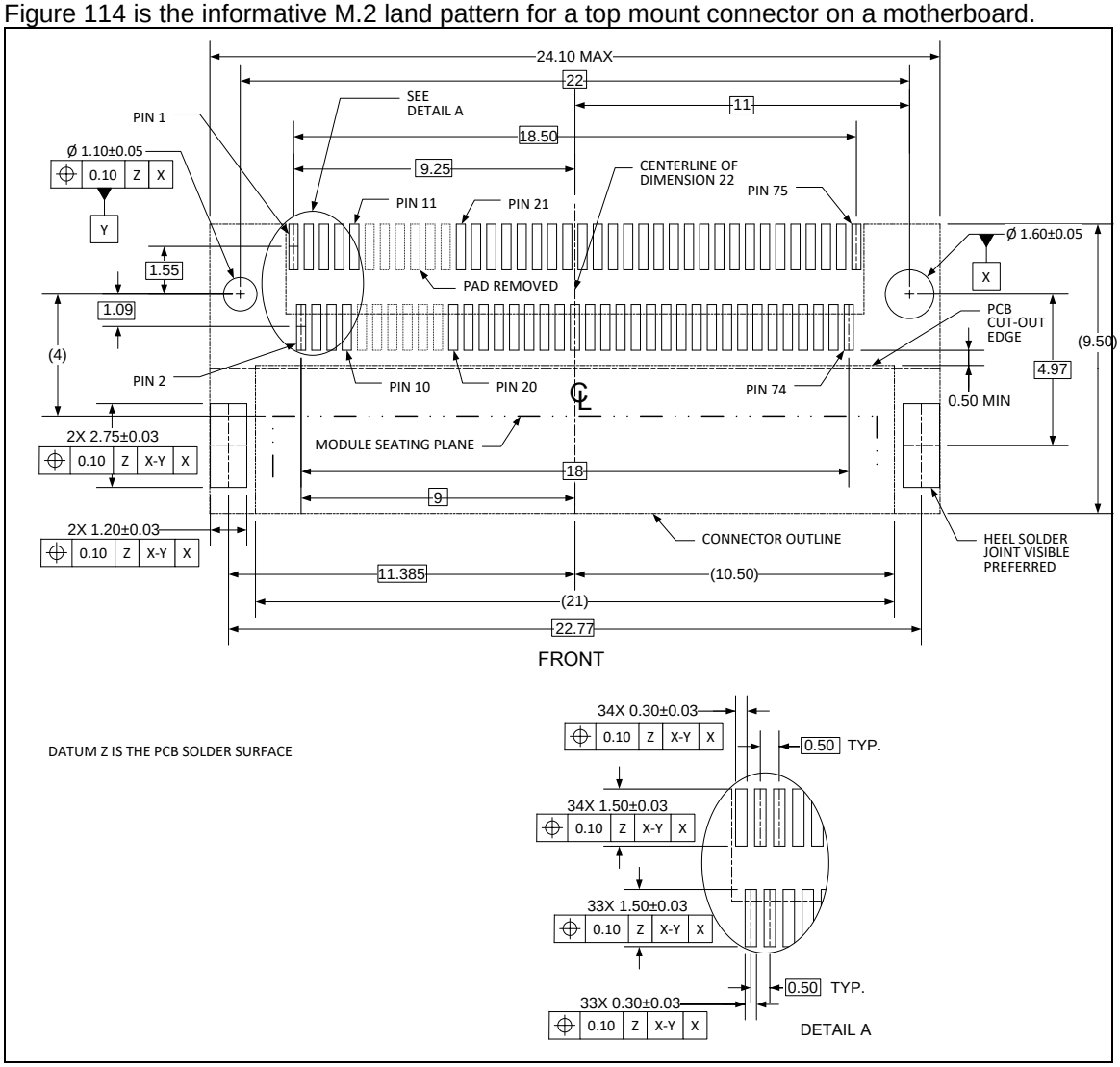


Figure 114 – M.2 land pattern for mother board (informative)

6.9.7 M.2 Z-height stack up (informative)

Figure 115 shows an informative view of the z-height stack up of the M.2 single sided assembly S2 profile.

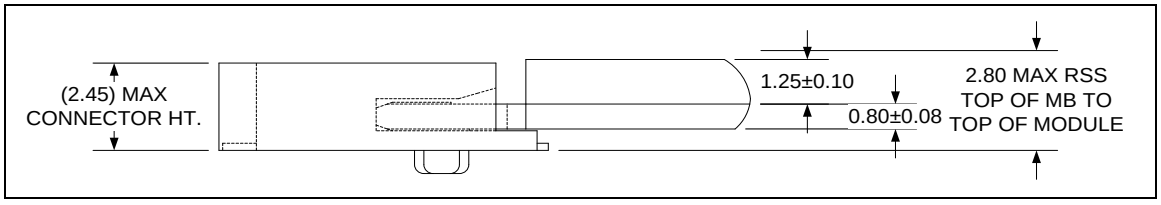


Figure 115 – M.2 single sided assembly – S2 profile (informative)

Figure 116 shows an informative view of the z-height stack up of the M.2 double sided assembly D2 profile.

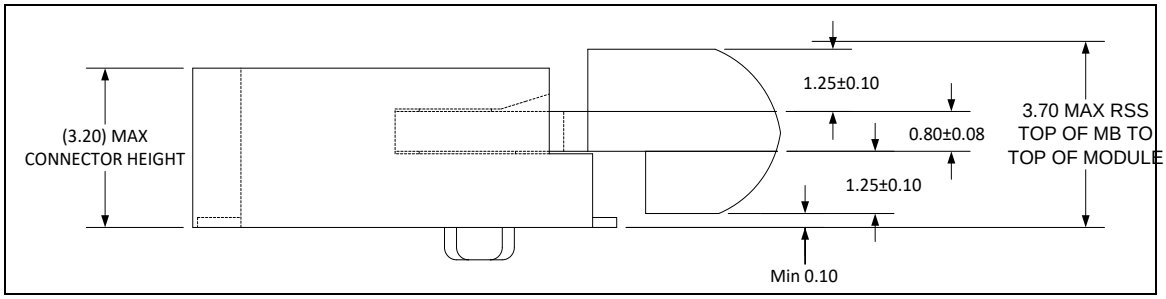


Figure 116 – M.2 double sided assembly – D2 (informative)

Figure 117 shows an informative view of the z-height stack up of the M.2 double sided assembly D3 profile.

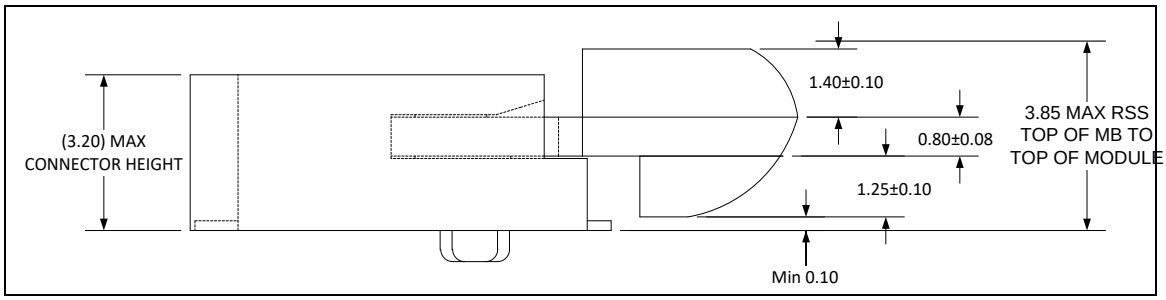


Figure 117 – M.2 double sided assembly – D3 (informative)

Figure 118 shows an informative view of the z-height stack up of the M.2 double sided assembly D5 profile.

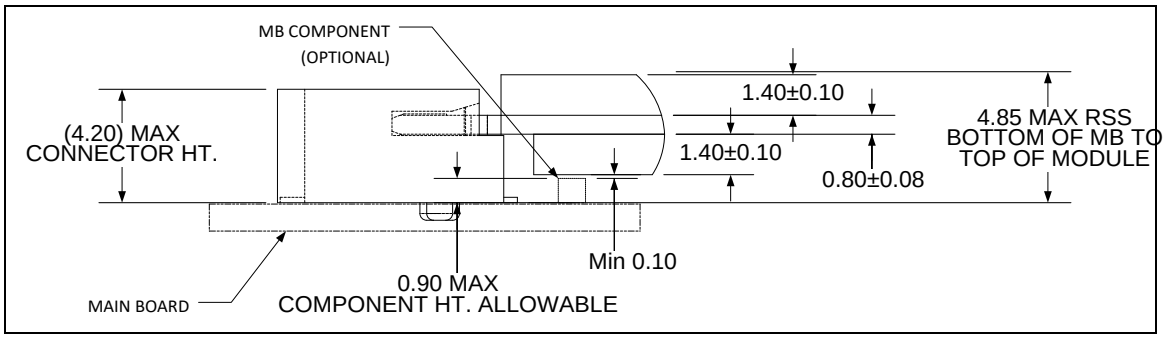


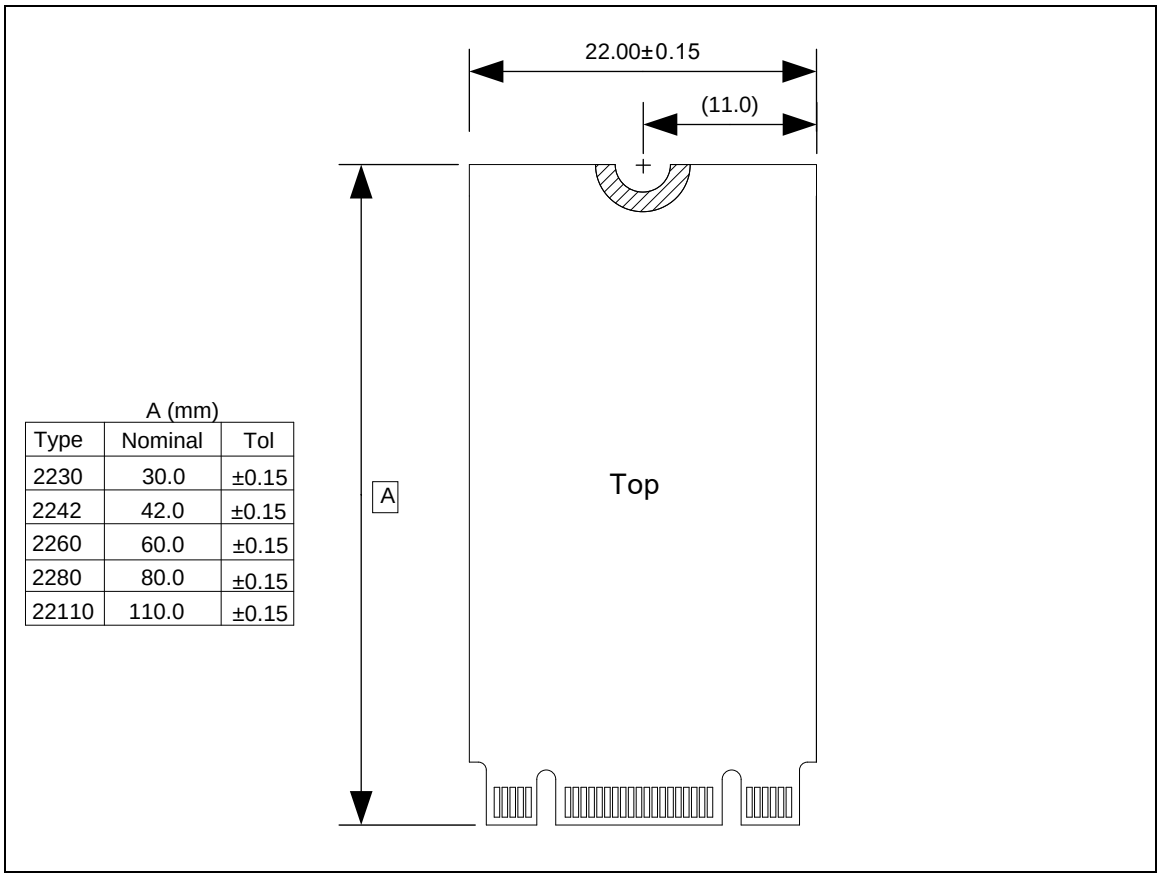
Figure 118 – M.2 double sided assembly – D5 (informative)

6.9.8 M.2 board sizes (informative)

Five board sizes are defined:

a) 30 mm;
b) 42 mm;
c) 60 mm;
d) 80 mm; and
e) 110 mm,

long as shown in Figure 119 (e.g., with both notch B and notch M present). All boards are nominally 22 mm wide.



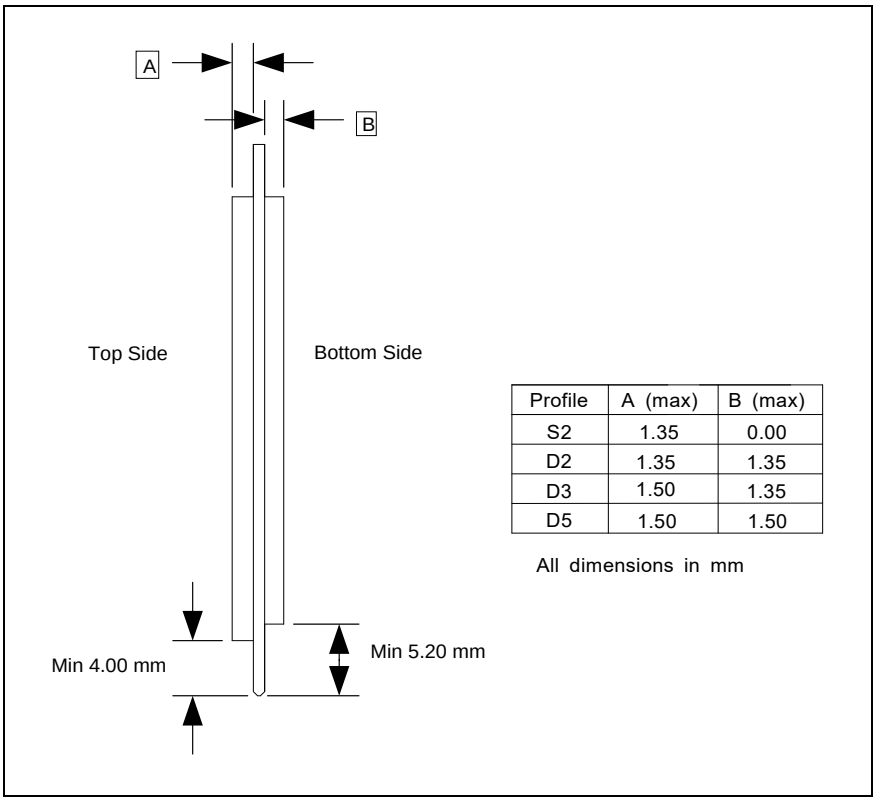
A (mm)

Type	Nominal	Tol
2230	30.0	±0.15
2242	42.0	±0.15
2260	60.0	±0.15
2280	80.0	±0.15
22110	110.0	±0.15

Figure 119 – M.2 board sizes (informative)

6.9.9 M.2 component placement and board thickness (informative)

Boards may be constructed as a single sided assembly (e.g., for absolute lowest z-height) or as double sided assembly. Four profiles are shown in Figure 120. Profile S2 is suitable for single sided SSDs. Profile D2 is suitable for SSD-only slots. Profiles D3 and D5 are suitable for slots that target multiple functions (e.g., SSDs or communication cards).



Profile	A (max)	B (max)
S2	1.35	0.00
D2	1.35	1.35
D3	1.50	1.35
D5	1.50	1.50

All dimensions in mm

Figure 120 – M.2 component profile and keep out zone (informative)

6.9.10 M.2 signal integrity

Table 23 – M.2 connector, electrical requirements

Parameter	Requirement
Differential impedance [a]	75 ohm to 95 ohm measured at 50 picoseconds rise time, from the 20 % threshold to the 80 % threshold
Differential insertion loss [a b]	≥ -0.5 dB up to 4 GHz and then ≥ -1 dB up to 8 GHz
Differential Near End Crosstalk [a b c]	≤ -36 dB up to 4 GHz and then ≤ -32 dB up to 8 GHz
Differential Far End Crosstalk [a b c]	≤ -40 dB up to 4 GHz and then ≤ -32 dB up to 8 GHz
[a] Mated connector and module including solder pad and gold finger. [b] The result is referenced to 85 ohm differential impedance. [c] The crosstalk shall be pair to pair between any two differential pairs.	

6.9.11 M.2 pad and anti-pad recommendations (informative)

Voiding planes under the pads reduces launch capacitance that improves signal integrity. The values shown in Figure 121 are typical of a board constructed with mainstream commercial grade FR-4 PCB material. Dimension of recommended pad sizes are in previous figures.

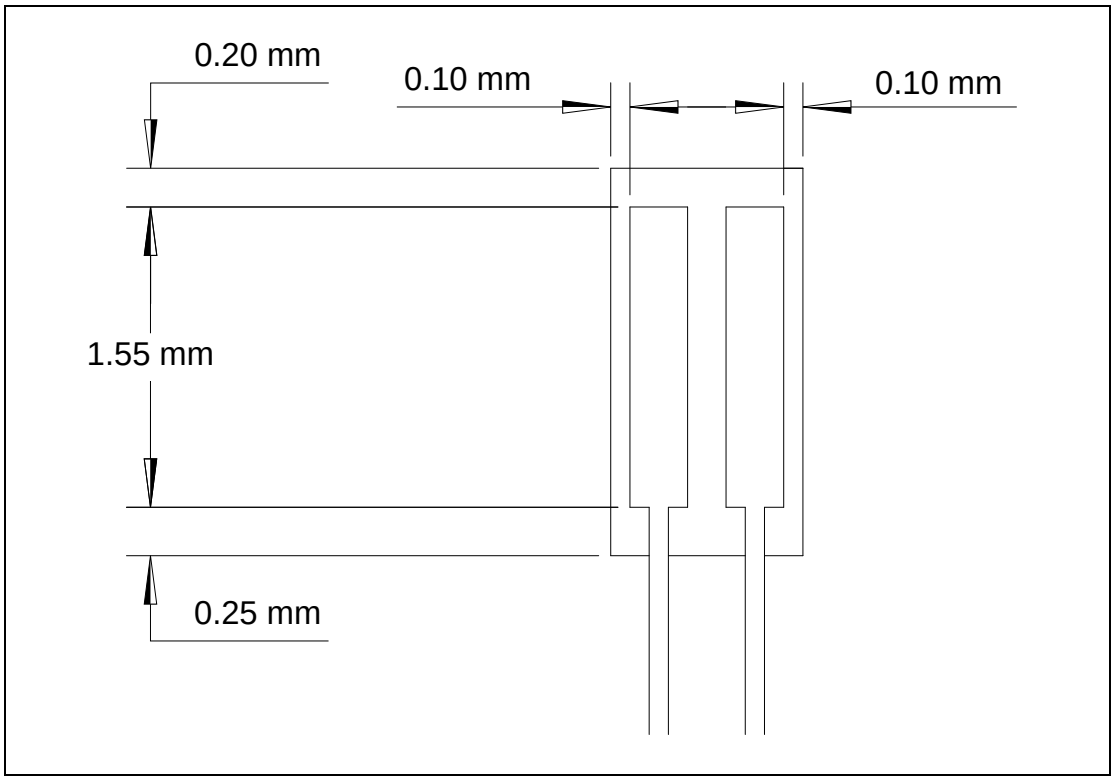


Figure 121 – M.2 mother board pad and void dimensions (informative)

6.9.12 M.2 minimum plane pull back from finger (informative)

Pad pull back from fingers in a typical FR-4 circuit board should improve signal integrity (see Figure 122).

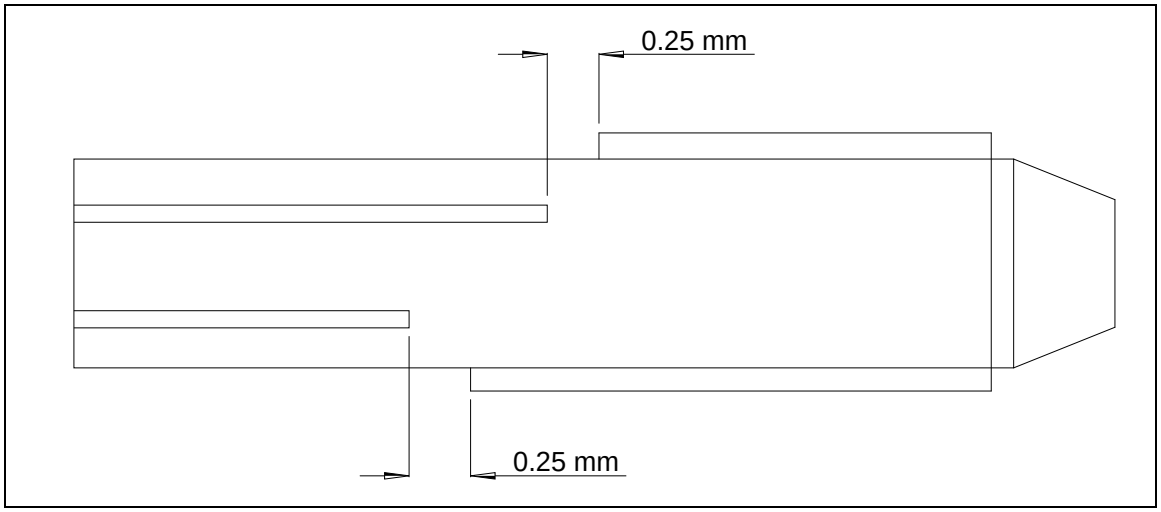


Figure 122 – M.2 pull back (informative)

6.9.13 M.2 socket 2 pin definition

Table 24 defines the signal assignment of the internal M.2 connector for device usage. This connector does not support hot plug capability, so there is no connection sequence specified. There are a total of 75 pins. 12 pin locations are used for mechanical key locations. This allows such a module to plug into both Key B and Key M connectors.

Socket 2 supports either a single lane SATA device, or a PCIe device with 1 lane or 2 lanes.

Pin direction is with respect to the module (i.e., Tx (transmit) is a signal driven by the module to the system).

Table 24 – M.2 device side signal assignments for key B (1x SATA, 2x PCIe) (part 1 of 2)

Pin position	Type	Description
1	CONFIG_3	Defines module type (see 6.9.16)
2	3.3 V	Supply pin, 3.3 V
3	GND	Ground
4	3.3 V	Supply pin, 3.3 V
5	No connect	No connect
6	Not Available	No connect (used for other purposes)
7	Not Available	No connect (used for other purposes)
8	Not Available	No connect (used for other purposes)
9	No connect	No connect
10	DAS/DSS	Device Activity Signal / Disable Staggered Spinup
11	No connect	No connect (used for other purposes)
12	(removed for key)	Mechanical notch B
13	(removed for key)	Mechanical notch B
14	(removed for key)	Mechanical notch B
15	(removed for key)	Mechanical notch B
16	(removed for key)	Mechanical notch B
17	(removed for key)	Mechanical notch B
18	(removed for key)	Mechanical notch B
19	(removed for key)	Mechanical notch B
20	Not Available	No connect (used for other purposes)
21	CONFIG_0	Defines module type (see 6.9.16)
22	Not available	No connect (used for other purposes)
23	Not available	No connect (used for other purposes)
24	Not available	No connect (used for other purposes)
25	Not available	No connect (used for other purposes)
26	Not available	No connect (used for other purposes)
27	GND	Ground
28	Not available	No connect (used for other purposes)
29	PERn1	PCIe signal (see PCIe M.2)
30	Not available	No connect (used for other purposes)
31	PERp1	PCIe signal (see PCIe M.2)
32	Not available	No connect (used for other purposes)
33	GND	Ground
34	Not available	No connect (used for other purposes)
35	PETn1	PCIe signal (see PCIe M.2)
36	Not available	No connect (used for other purposes)
37	PETp1	PCIe signal (see PCIe M.2)

[a] No connect on a host.

Table 24 – M.2 device side signal assignments for key B (1x SATA, 2x PCIe) (part 2 of 2)

Pin position	Type	Description
38	DEVSLP	Device Sleep, input. If driven high the host is informing the SSD to enter a low power state.
39	GND	Ground
40	Not available	No connect (used for other purposes)
41	SATA-B+/PERn0	Host receiver differential signal pair. If in PCIe mode, see PCIe M.2.
42	na	No connect (used for other purposes)
43	SATA-B-/PERp0	Host receiver differential signal pair. If in PCIe mode, see PCIe M.2.
44	Not available	No connect (used for other purposes)
45	GND	Ground
46	Not available	No connect (used for other purposes)
47	SATA-A-/PETn0	Host transmitter differential signal pair. If in PCIe mode, see PCIe M.2.
48	Not available	No connect (used for other purposes)
49	SATA-A+/PETp0	Host transmitter differential signal pair. If in PCIe mode, see PCIe M.2.
50	PERST#	PCIe signal (see PCIe M.2)
51	GND	Ground
52	CLKREQ#	PCIe signal (see PCIe M.2)
53	REFCLKN	PCIe signal (see PCIe M.2)
54	PEWAKE#	PCIe signal (see PCIe M.2)
55	REFCLKP	PCIe signal (see PCIe M.2)
56	MFG1	Manufacturing pin. Use determined by vendor. [a]
57	GND	Ground
58	MFG2	Manufacturing pin. Use determined by vendor. [a]
59	(removed for key)	Mechanical notch M
60	(removed for key)	Mechanical notch M
61	(removed for key)	Mechanical notch M
62	(removed for key)	Mechanical notch M
63	(removed for key)	Mechanical notch M
64	(removed for key)	Mechanical notch M
65	(removed for key)	Mechanical notch M
66	(removed for key)	Mechanical notch M
67	Not available	No connect (used for other purposes)
68	SUSCLK	PCIe signal (see PCIe M.2)
69	CONFIG_1	Defines module type (see 6.9.16)
70	3.3 V	Supply pin, 3.3 V
71	GND	Ground
72	3.3 V	Supply pin, 3.3 V
73	GND	Ground
74	3.3 V	Supply pin, 3.3 V
75	CONFIG_2	Defines module type (see 6.9.16)

[a] No connect on a host.

6.9.14 M.2 SSD socket 3

Slot M supports a one lane SATA device or a PCIe device with 1, 2, or 4 lanes.

Table 25 – M.2 signal assignments for card keyed for slot M (1x SATA, 1x, 2x, or 4x PCIe) (part 1 of 2)

Pin position	Type	Description
1	CONFIG_3	Defines module type (see 6.9.16)
2	3.3 V	Supply pin, 3.3 V
3	GND	Ground
4	3.3 V	Supply pin, 3.3 V
5	PERn3	PCIe signal, see PCI Express M.2 CEM
6	Not available	No connect (used for other purposes)
7	PERp3	PCIe signal, see PCI Express M.2 CEM
8	Not available	No connect (used for other purposes)
9	GND	Ground
10	DAS/DSS	Device Activity Signal / Disable Staggered Spinup
11	PETn3	PCIe signal, see PCI Express M.2 CEM
12	3.3 V	Supply pin, 3.3 V
13	PETp3	PCIe signal, see PCI Express M.2 CEM
14	3.3 V	Supply pin, 3.3 V
15	GND	Ground
16	3.3 V	Supply pin, 3.3 V
17	PERn2	PCIe signal, see PCI Express M.2 CEM
18	3.3 V	Supply pin, 3.3 V
19	PERp2	PCIe signal, see PCI Express M.2 CEM
20	Not available	No connect (used for other purposes)
21	CONFIG_0	Defines module type (see 6.9.16)
22	Not available	No connect (used for other purposes)
23	PETn2	PCIe signal, see PCI Express M.2 CEM
24	Not available	No connect (used for other purposes)
25	PETp2	PCIe signal, see PCI Express M.2 CEM
26	Not available	No connect (used for other purposes)
27	GND	Ground
28	Not available	No connect (used for other purposes)
29	PERn1	PCIe signal, see PCI Express M.2 CEM
30	Not available	No connect (used for other purposes)
31	PERp1	PCIe signal, see PCI Express M.2 CEM
32	Not available	No connect (used for other purposes)
33	Gen	Ground
34	Not available	No connect (used for other purposes)
35	PETn1	PCIe signal, see PCI Express M.2 CEM
36	Not available	No connect (used for other purposes)
37	PETp1	PCIe signal, see PCI Express M.2 CEM
38	DEVSLP	Device Sleep, Input. If driven high, the host is informing the SSD to enter a low power state.
39	GND	Ground
40	Not available	No connect (used for other purposes)
41	SATA-B+/PERn0	Host receiver differential signal pair. If in PCIe mode, see PCI Express M.2 CEM.
42	Not available	No connect (used for other purposes)
43	SATA-B-/PERp0	Host receiver differential signal pair. If in PCIe mode, see PCI Express M.2 CEM.
44	Not available	No connect (used for other purposes)
45	GND	Ground

[a] No connect on a host.

Table 25 – M.2 signal assignments for card keyed for slot M (1x SATA, 1x, 2x, or 4x PCIe) (part 2 of 2)

Pin position	Type	Description
46	Not available	No connect (used for other purposes)
47	SATA-A-/PETn0	Host transmitter differential pair. If in PCIe mode, see PCI Express M.2 CEM.
48	Not available	No connect (used for other purposes)
49	SATA-A+/PETp0	Host transmitter differential pair. If in PCIe mode, see PCI Express M.2 CEM.
50	PERST#	PCIe signal, see PCI Express M.2 CEM
51	GND	Ground
52	CLKREQ#	PCIe signal, see PCI Express M.2 CEM
53	REFCLKN	PCIe signal, see PCI Express M.2 CEM
54	PEWAKE#	PCIe signal, see PCI Express M.2 CEM
55	REFCLKP	PCIe signal, see PCI Express M.2 CEM
56	MFG1	Manufacturing pin. Use determined by vendor. [a]
57	GND	Ground
58	MFG2	Manufacturing pin. Use determined by vendor. [a]
59	(removed for key)	Mechanical notch M
60	(removed for key)	Mechanical notch M
61	(removed for key)	Mechanical notch M
62	(removed for key)	Mechanical notch M
63	(removed for key)	Mechanical notch M
64	(removed for key)	Mechanical notch M
65	(removed for key)	Mechanical notch M
66	(removed for key)	Mechanical notch M
67	Not available	No connect (used for other purposes)
68	SUSCLK	PCIe signal, see PCI Express M.2 CEM
69	CONFIG_1	Defines module type (see 6.9.16)
70	3.3 V	Supply pin, 3.3 V
71	Gnd	Ground
72	3.3 V	Supply pin, 3.3 V
73	Gnd	Ground
74	3.3 V	Supply pin, 3.3 V
75	CONFIG_2	Defines module type (see 6.9.16)

[a] No connect on a host.

6.9.15 M.2 electrical

Signal and 3.3 V Signal Requirements are defined in Table 26.

Table 26 – M.2 voltage and current requirements

Symbol	Parameter	Conditions	Min	Max	Units
+3.3 Vaux [c]	Supply voltage	-	3.135	3.465	V
V_{IH}	Input high voltage	-	2.0	3.6	V
V_{IL} [d]	Input low voltage	-	-0.5	0.5	V
I_{OL} [a]	Output low current for open-drain signals	0.4 V	4	-	mA
I_{IN}	Input leakage current	0 V to 3.3 V	-10	+10	uA
I_{LKG}	Output leakage current	0 V to 3.3 V	-50	+50	uA
C_{IN}	Input pin capacitance	-	-	7	pF
C_{OUT}	Output pin coapacitance	-	-	30	pF
T_{RISE} [b]	Power up time	-	-	10	ms

[a] Not applicable to LED# and DAS/DSS pins.
[b] Rise time of 3.3 V supply from 0 V to its minimum voltage under maximum load.
[c] Each pin shall be capable of supplying at least 500 mA.
[d] Only applicable to CONFIG pins.

6.9.16 M.2 signal definitions, configuration

Configuration pins (see Table 27) are used to inform the host system the type of card present in a socket. These pins are either ‘no connect’ or grounded on the device.

Table 27 – M.2 config pin settings

Type	CONFIG_0 (pin 21)	CONFIG_1 (pin 69)	CONFIG_2 (pin 75)	CONFIG_3 (pin 1)
SSD - SATA	Ground	Ground	Ground	Ground
PCIe	See PCIe M.2			

6.9.17 M.2 mated connector differential impedance (informative)

The purpose of the mated connector impedance requirement is to optimize signal integrity by minimizing reflections. The host may support the use of PCIe or SATA signaling over the same interconnect. The nominal characteristic differential impedance of PCIe is 85 ohm, while the nominal characteristic differential impedance of SATA is 100 ohm.

The differential impedance of a mated connector should be within 90 ohm ±12 ohm, as seen from a 50 picoseconds rise time, measured from the 20 % threshold to 80 % threshold of a differential time domain reflectometer (TDR). The impedance profile of a mated connector should fall within this range. Note that this mated connector differential requirement applies to all the connector mating interfaces defined in the M.2 specification. The measurement includes the connector footprints on both host and device PCBs (backplane application) and the cable connector wire termination area (e.g., cabled application).

6.10 SATA Express connector (obsolete)

6.10.1 SATA Express connector overview

SATA Express defines electrical and mechanical requirements for SATA Express that is a PCI Express (PCIe) connection to the existing standard 3.5 inch and 2.5 inch disk drive form factors for client applications. It is intended for providing a smooth transition path from SATA to PCIe storage, leveraging both PCIe specifications and 3.5 inch and 2.5 inch drive mechanical standards.

6.10.2 SATA Express connector goals

SATA Express is developed with the following characteristics:

a) PCIe (see 3.2) connection to client PCIe storage devices;
b) standardized connectors and form factors, fitting in the existing 3.5 inch and 2.5 inch drive mechanical enclosures;
c) muxed PCIe and SATA lanes on the host so the host connector work with either a PCIe or a SATA storage device;
d) both cabling and direct connection solutions to support desktop and notebook product needs; and
e) no PCIe reference clock sent over cable to allow the continued use of the low cost SATA-like cable solutions.

6.10.3 SATA Express technical summary

The technical summary of SATA Express is listed below:

a) the focus is on an SATA Express cable and motherboard connector for client applications and a device connector;
b) the form factors supported in SATA Express are SFF-8301 (3.5 inch drive) and SFF-8201 (2.5 inch drive) with no changes;
c) the device connector is based on SFF-8680 with a repurposed pin-out;
d) SATA Express supports x2 PCIe, Gen2 and Gen3, signaling and references PCIe Base and Card Electromechanical (CEM) specifications Rev 3.0 for electrical requirements;
e) SATA Express is compatible with SFF-8639 when server OEMs choose to mux PCIe with SAS/SATA lanes;
f) SATA Express supports legacy SATA devices;
g) PCIe and SATA lanes are muxed on the host so either a SATA or PCIe device operates with the host;
h) SATA Express delivers 5 V and 12 V power to the device with 3.3 V not supported (see 7.3.5); and
i) SATA Express allows existing SATA power supply cable infrastructure be used through the addition of a power supply dongle (see 6.10.12).

The PCIe reference clock (RefClk) is not included in the cabled interface. A separate RefClk with independent SSC (SRIS) architecture is used, similar to the clocking architecture used in SATA.

Figure 123 shows example configurations for SATA Express. In all cases shown, the backward compatibility with SATA is maintained by muxing PCIe and SATA together on the host.

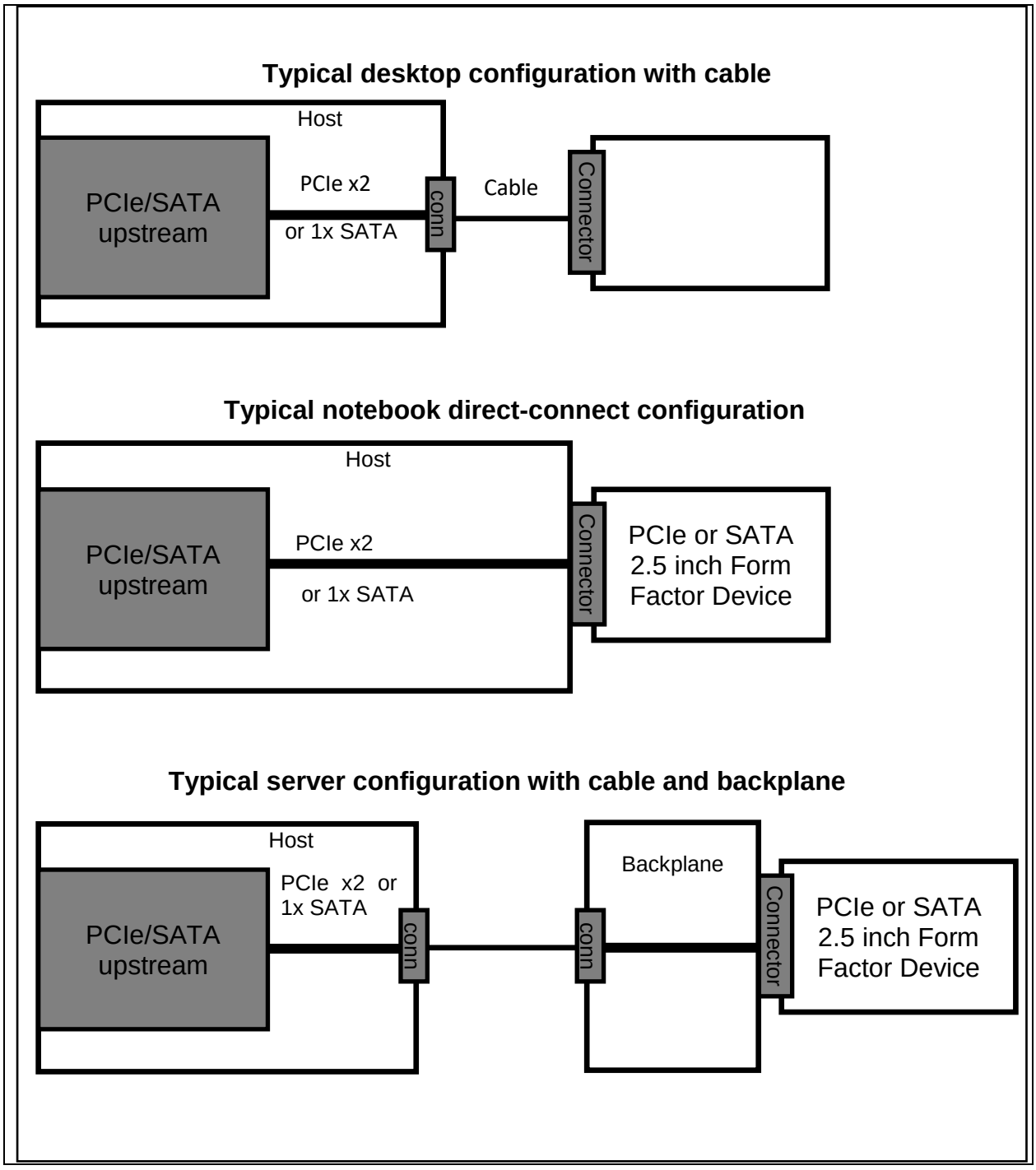


Figure 123 – Example configurations for client

Table 3 and Table 4 summarize the key characteristics of SATA Express.

6.10.4 SATA Express scope

SATA Express is a form factor specification that focuses on connectors and cables for PCIe storage. The connectors on the host are backward compatible extension of SATA connectors. The overall mechanical form factors are compliant with the 3.5 inch and 2.5 inch form factors.

SATA Express defines the following:

a) pin list and signal assignment of the connectors;
b) mechanical definition of the connectors and cable interfaces; and
c) limited electrical specification (largely reference to PCIe and legacy SATA Specifications).

6.10.5 SATA Express signal list

The signal list is a combination of SATA and PCIe signals. Table 28 provides an overview of the signals groups.

Table 29 gives the complete signal list, usage, and the document that governs the signal definition. Most of the signals used in SATA Express are definitively defined in PCIe spec or other sections of this spec (the SATA spec). Figure 124 shows the mapping of signals to the host receptacle connector pins.

Table 28 – SATA Express signal list summary

Usage	Signals	Contacts
x2 PCIe muxed with SATA	2 × (Tx and Rx pairs) + GND pins	14
Power	5 V and 12 V + GND pins	10
Device Activity Signal/Disable Staggered Spinup (optional)	DAS/DSS	1
PCIe CLKREQ# / SATA DEVSLP	CLKREQ#/ DEVSLP	1
PCIe Reset	PERST#	1
PCIe/SATA Interface Detect	IFDet	1
PCIe RefClk (optional)	RefClk+/RefClk- + ClkDet	3
Reserved	Reserved	1
Total		32

Table 29 – SATA Express pin group table (for host receptacle and device plug connectors) (part 1 of 2)

Pin Group	Pin Name	Signal Description	Usage	Defining Specification
PCIe/SATA Data	PETp0/A0+,PETn0/A0-	Host transmit PCIe Lane 0/SATA	PCIe/SATA	PCIe/SATA
	PETp1, PETn1	Host transmit PCIe Lane 1	PCIe	PCIe
	PERn0/B0-, PERp0/B0+	Host Receive PCIe Lane 0/SATA	PCIe/SATA	PCIe/SATA
PCIe Reset	PERST#	PCIe Reset	PCIe	PCIe
PCIe RefClk	RefClk+, RefClk-	PCIe common reference clock	PCIe, optional	PCIe
L1 PM Substate / DevSleep	CLKREQ#/DEVSLP	PCIe: Enter/exit L1 PM substate SATA: Enter/exit device sleep	PCIe/SATA	PCIe/SATA

Table 29 – SATA Express pin group table (for host receptacle and device plug connectors)
(part 2 of 2)

Pin Group	Pin Name	Signal Description	Usage	Defining Specification
PCIe/SATA Detect	IFDet	Detect a PCIe drive	PCIe/SATA	SATA
SATA Sideband (optional)	DAS/DSS	Device Activity Signal / Disable Staggered Spinup	SATA	SATA
Reserved	Reserved	Reserved for future use	Reserved	
Power	12 V	12 V power	PCIe/SATA	SATA
	5 V	5 V power	PCIe/SATA	SATA
	GND	Ground	PCIe/SATA	SATA

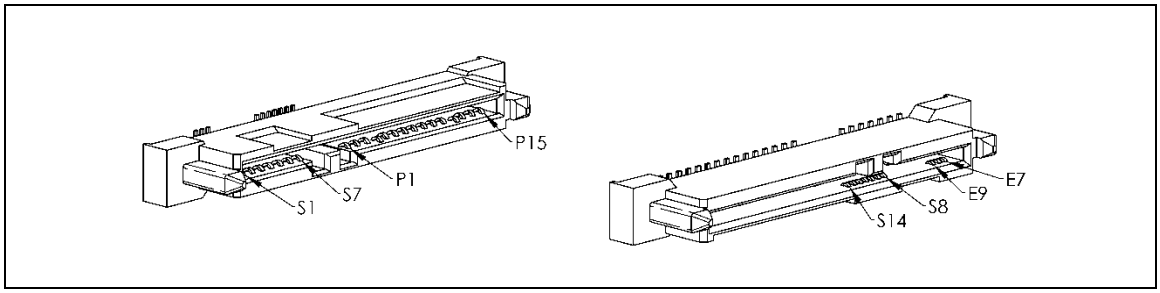


Figure 124 – SATA Express pinout for host receptacle connectors

Table 30 – SATA Express signal pin list for host receptacle and device plug connectors

Pin position	Name [a]	Signal Description [a]	Usage	Signal Direction	Mating Sequence [c]
S1	GND	Ground	PCIe/SATA		Second
S2	PETp0/A+	PCIe Lane 0 Host Tx / SATA Tx pair [b]	PCIe/SATA	Host Output	Third
S3	PETn0/A-				
S4	GND	Ground	PCIe/SATA		Second
S5	PERn0/B-	PCIe Lane 0 Host Rx / SATA Rx pair [b]	PCIe/SATA	Input to Host	Third
S6	PERp0/B+				
S7	GND	Ground	PCIe/SATA		Second
S8	GND	Ground	PCIe only		Second
S9	PETp1	PCIe Lane 1 Host Tx pair	PCIe only	Host Output	Third
S10	PETn1				
S11	GND	Ground	PCIe only		Second
S12	PERn1	PCIe Lane 1 Host Rx pair	PCIe only	Input to Host	Third
S13	PERp1				
S14	GND	Ground	PCIe only		Second

[a] The "Name" and "Signal Description" columns are relative to the host, as defined by each protocol specification.

[b] Muxing of the PCIe and SATA lanes is required for the host. Only the PCIe Lane 0 is required to mux with the SATA lane for the host receptacle connector since the client SATA device has only a single lane. For the host plug connector, both PCIe lanes (0 and 1) shall be muxed with SATA. This allows connections to two separate SATA devices via independent cables.

[c] Mating sequence defined here is for the host receptacle connector and device plug connector. The mating sequence for the cabled case is defined in 6.10.10.

Table 31 – SATA Express power pin list for host receptacle and device plug connectors

Pin position	Name [a]	Signal Description [a]	Usage	Signal Direction	Mating Sequence [c]
P1	Reserved [d]	Reserved for future use	PCIe/SATA		Third
P2	PERST# [d]	PCIe reset	PCIe only	Host Output	Third
P3	CLKREQ# / DEVSLP [d]	L1 PM substate / Device sleep	PCIe/SATA	Host Output	Second
P4	IFDet [e]	Interface (PCIe/SATA) detect	PCIe/SATA	Input to Host	First
P5	GND	Ground	PCIe/SATA		Second
P6	GND	Ground	PCIe/SATA		Second
P7	V5 [b]	5 V power, pre-charge	PCIe/SATA	Host Output	Second
P8	V5 [b]		PCIe/SATA	Host Output	Third
P9	V5 [b]		PCIe/SATA	Host Output	Third
P10	GND	Ground	PCIe/SATA		Second
P11	DAS/DSS	Device Activity Signal / Disable Staggered Spinup	SATA only	Bi-directional	Third
P12	GND	Ground	PCIe/SATA		First
P13	V12 [b]	12 V power, pre-charge	PCIe/SATA	Host Output	Second
P14	V12 [b]	12 V power	PCIe/SATA	Host Output	Third
P15	V12 [b]	12 V power	PCIe/SATA	Host Output	Third
E7	RefClk+ [g]	PCIe common RefClk	PCIe only, optional [f]	Host Output	Third
E8	RefClk- [g]				
E9	ClkDet [h]	PCIe RefClk detect	PCIe only, optional	Input to Host	Second

[a] The "Name" and "Signal Description" columns are relative to the host, as defined by each protocol specification.
[b] The power pins on the PCIe device shall be bused together for each supply voltage, in the same way as defined in the legacy SATA specification.
[c] Mating sequence defined here is for the host receptacle connector and device plug connector. The mating sequence for the cabled case is defined in 6.10.10.
[d] Pins P1, P2, and P3 were defined as the 3.3 V pins in previous versions of this specification (i.e., SATA rev 3.1) and have since been retired. Pins P1, P2, and P3 are repurposed to be PERST# and other signals in SATA Express. To avoid damage to a PCIe device, the components that connect to those pins shall be able to tolerate the application of 3.3 V. If CLKREQ#/DEVSLP is not implemented, then P3 should be a no connect.
[e] Pin P4 is a GND in previous revisions of this specification (i.e., SATA rev 3.1). It is named IFDet for interface detect in SATA Express. The detail interface detect mechanism is discussed in 7.3.3.
[f] The RefClk pins E7 and E8 are defined following SFF-8639. Support of the common RefClk is optional. Management of system compatibility with PCIe devices requiring the common RefClk is beyond the scope of SATA Express.
[g] For cabled applications, there is no reference clock (RefClk) included in the host cable plug connector (see 6.10.10), requiring an SRIS architecture. Requirements associated with the separate RefClk with independent SSC are defined in the PCIe Base Specification 3.0 ECN-Separate RefClk Independent SSC Architecture.
[h] ClkDet (i.e., pin E9) is used by the host to detect the device RefClk type. If a SATA Express device requires the common RefClk, then the device shall ground pin E9. If a SATA Express device supports SRIS, then the device shall not ground pin E9.

6.10.6 SATA Express mechanical

The following SATA Express connectors are defined, as illustrated in Figure 125:

a) device plug connector;
b) device cable receptacle connector;
c) host receptacle connector;
d) host plug connector; and
e) host cable receptacle connector.

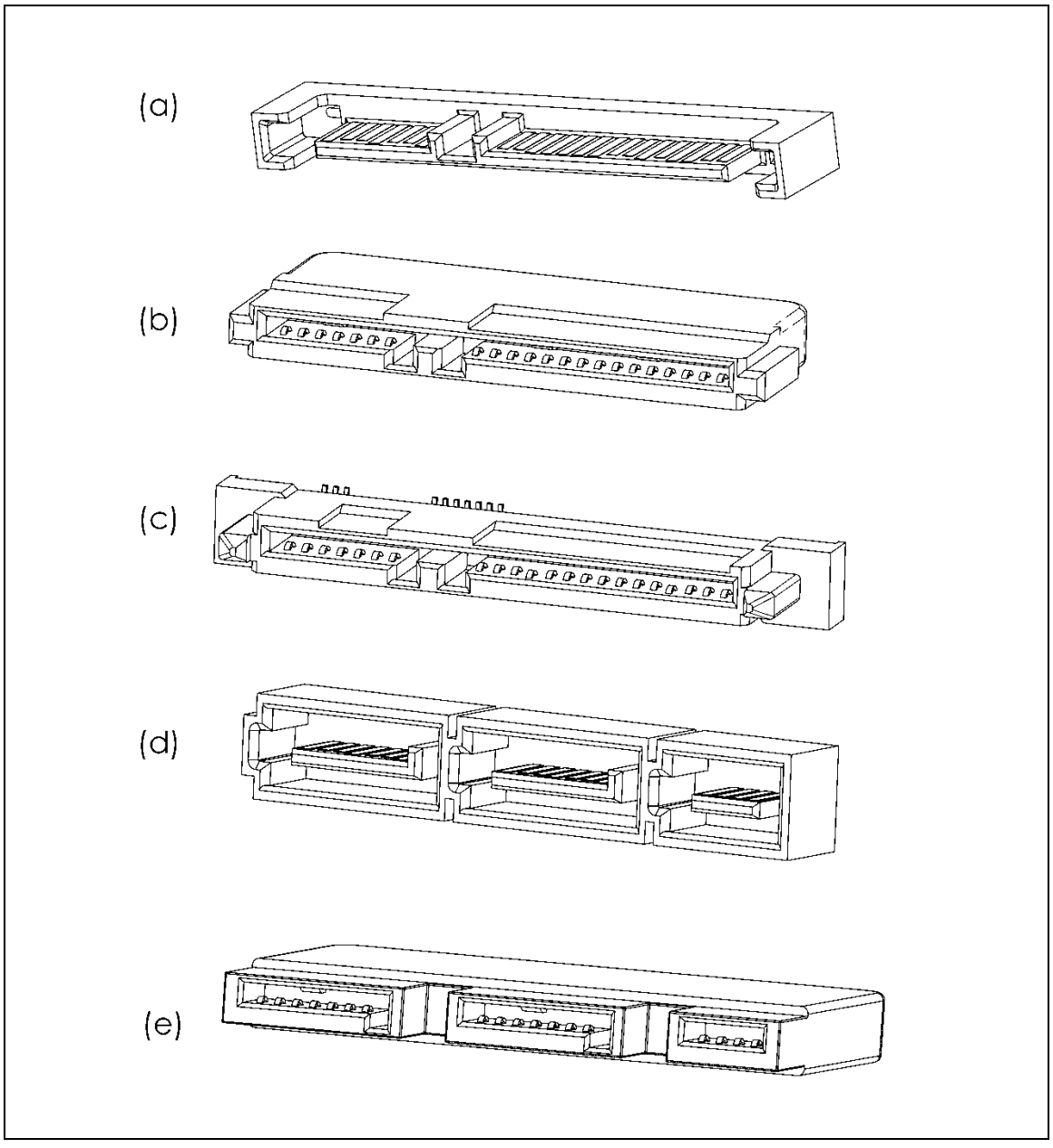


Figure 125 – Examples of SATA Express connectors

6.10.7 SATA Express device plug connector

The SATA Express device plug connector is physically similar to the standard SAS device connector defined in SFF-8680. The cable retention feature requirements are unchanged, the keying is different, and optional pins are added (i.e., E7, E8, and E9) for applications requiring RefClk.

The SATA Express device plug connector mates with:

a) SATA Express host receptacle connector;
b) SATA Express device cable receptacle connector;
c) SFF-8639 backplane connector; and
d) SFF-8482, SFF-8630, and SFF-8680 SAS receptacle connectors.

Management of the non-interoperability issues regarding mating the SATA Express device plug connector to SAS receptacles is beyond the scope of SATA Express.

Figure 126 shows an isometric view of the SATA Express device plug connector and Figure 127 defines the SATA Express device plug connector mating interface.

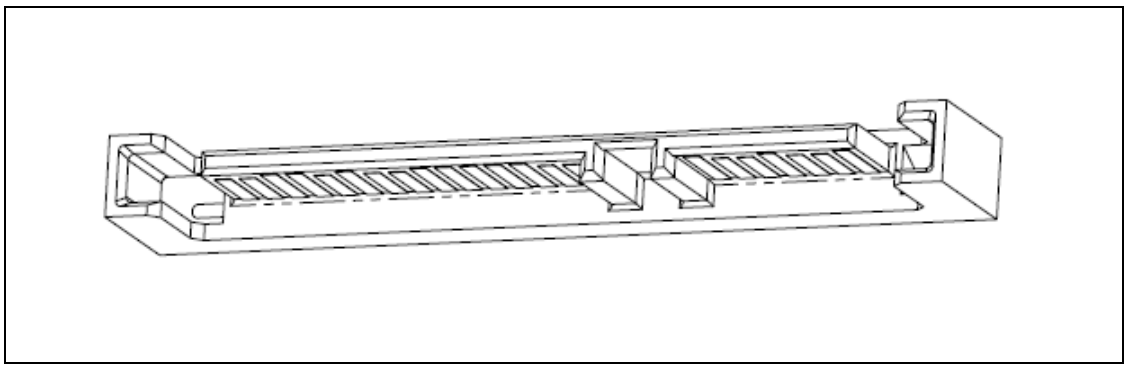


Figure 126 – SATA Express device plug connector isometric drawing

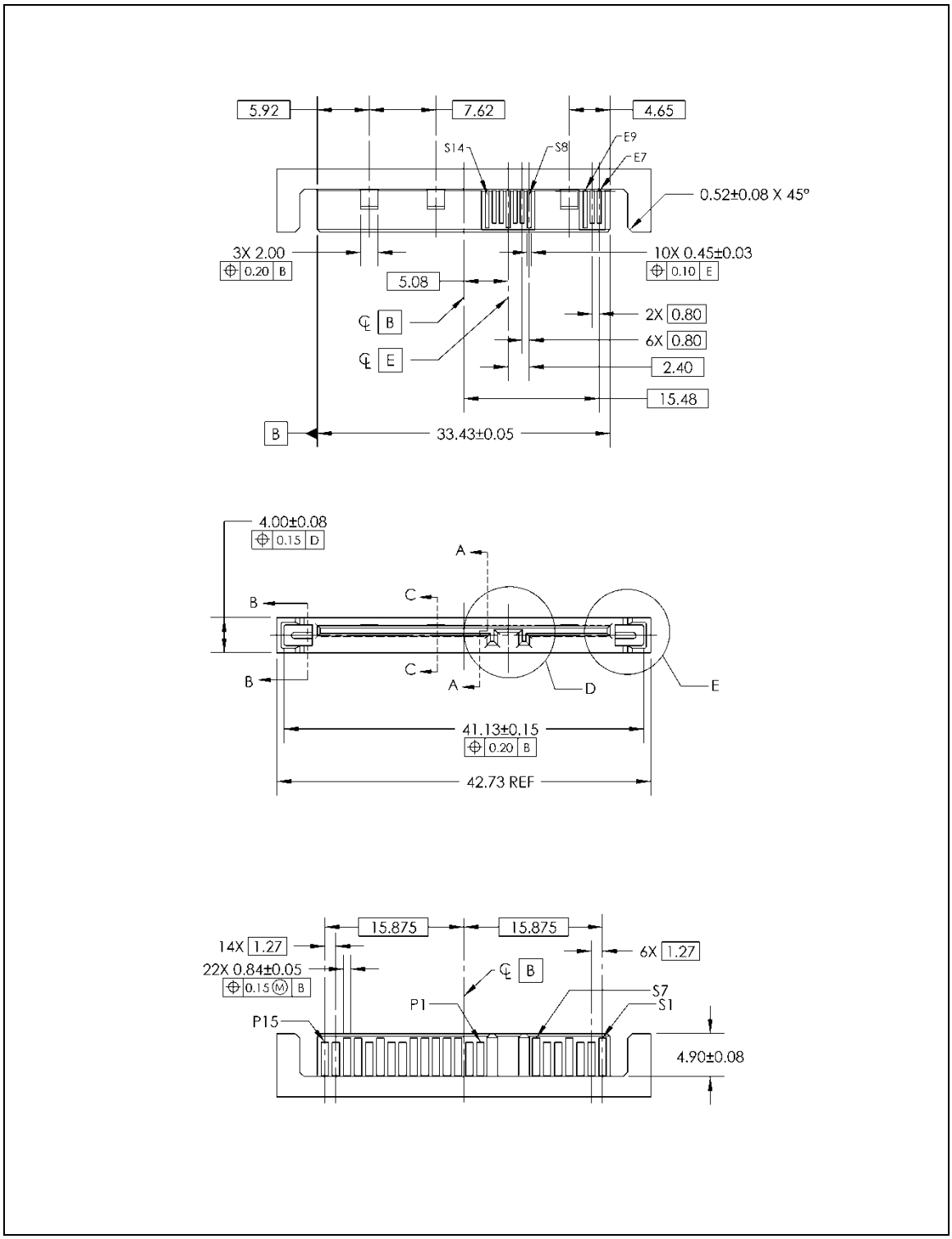


Figure 127 – SATA Express device plug connector drawing (part 1 of 2)

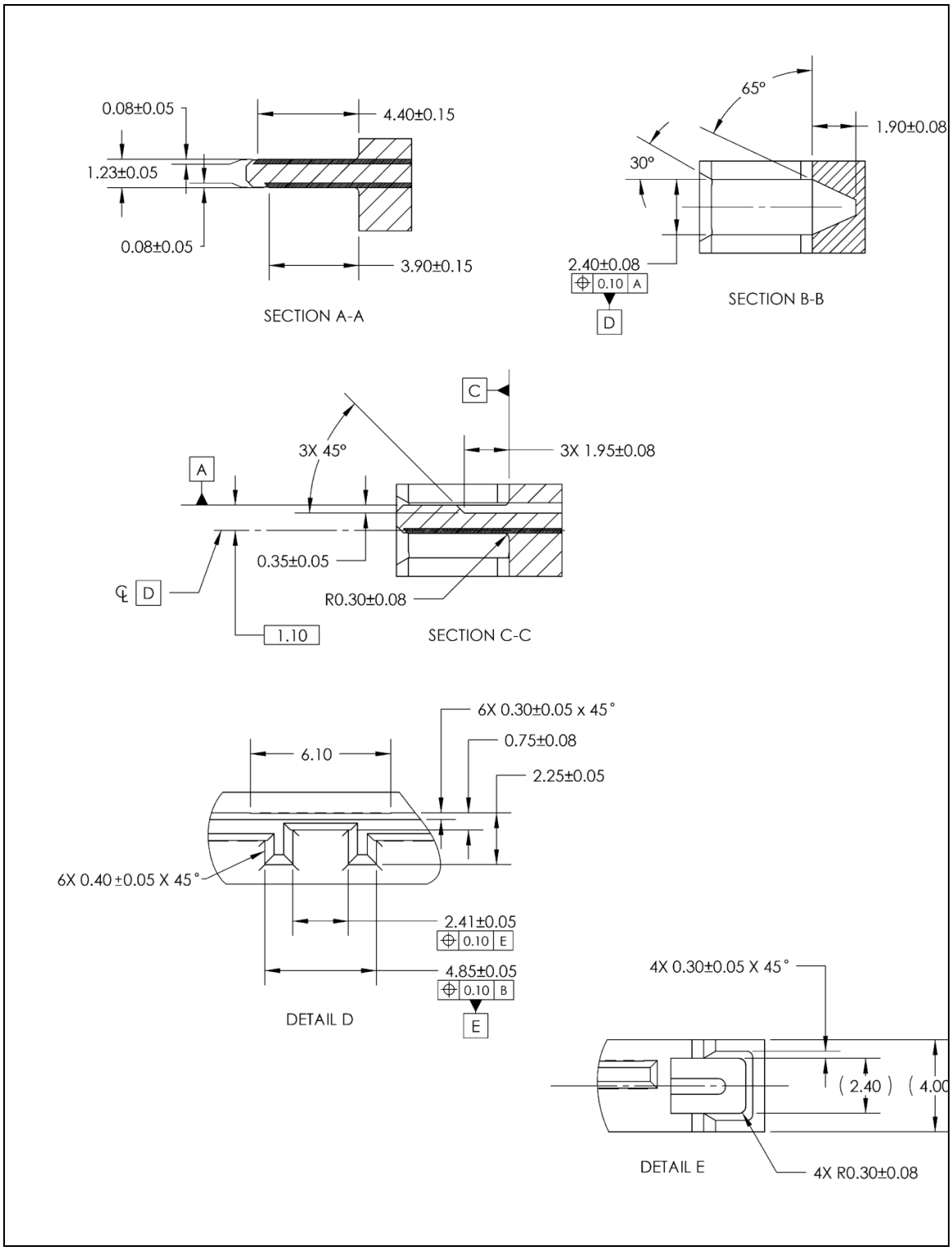


Figure 127 – SATA Express device plug connector drawing (part 2 of 2)

6.10.8 SATA Express device cable receptacle connector

The SATA Express device cable receptacle connector mates with the SATA Express device plug connector or the SATA device plug connector. It is physically similar to the SAS backplane

connector defined in SFF-8680, but has a different pin definition. The SATA Express device cable receptacle connector is keyed to prevent insertion of a SAS drive plug. Figure 128 defines the mating interface of the SATA Express device cable receptacle connector. RefClk is not supported in cabled applications since E7, E8, and E9 are not included in this connector.

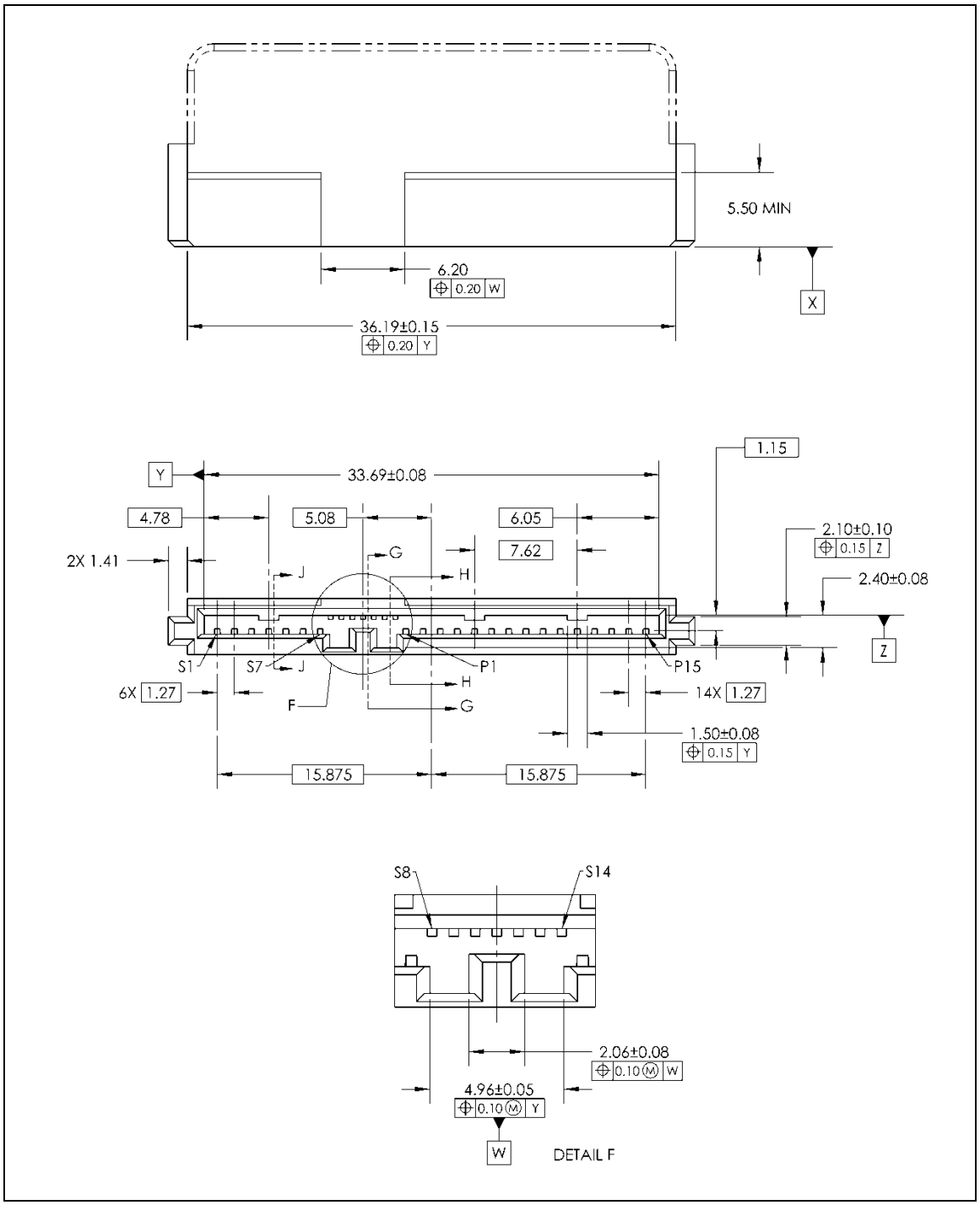


Figure 128 – SATA Express device cable receptacle connector drawing (part 1 of 2)

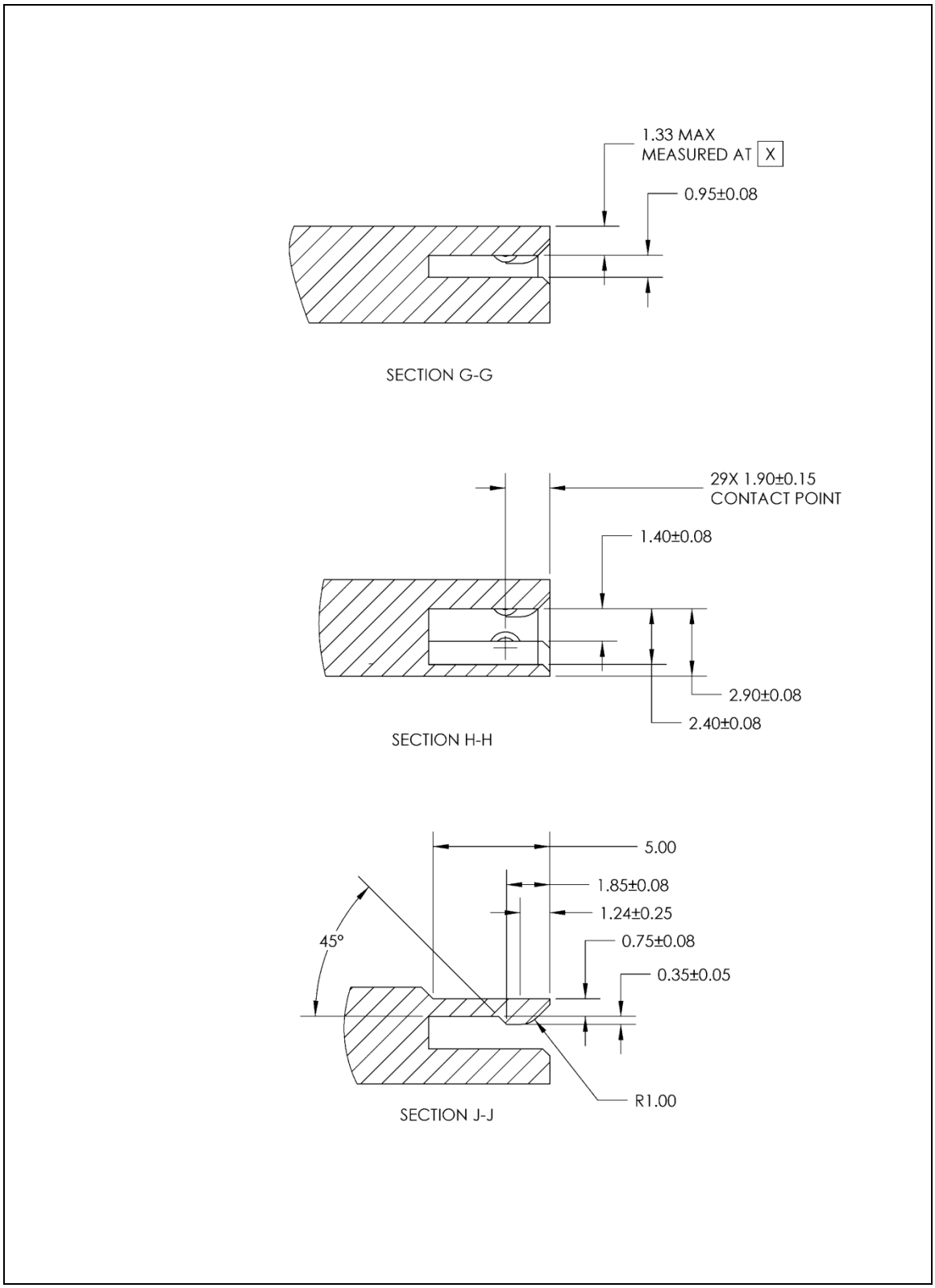


Figure 128 – SATA Express device cable receptacle connector drawing (part 2 of 2)

6.10.9 SATA Express host receptacle connector

The SATA Express host receptacle connector is identical to SATA Express device cable receptacle connector except for the following differences:

a) the SATA Express host receptacle connector may include optional contacts E7, E8, and E9 for RefClk support; and
b) the SATA Express host receptacle connector has no retention features (see View Section J-J in Figure 128).

The SATA Express host receptacle connector mates with the following connectors:

a) SATA Express device plug connector; and
b) SATA device plug connector.

Figure 129 shows an isometric view of the SATA Express host receptacle connector and Figure 130 defines the SATA Express host receptacle connector mating interface. Note that Figure 130 does not include all the necessary dimensions or section views to complete the interface definition. Refer to Figure 128 for completeness.

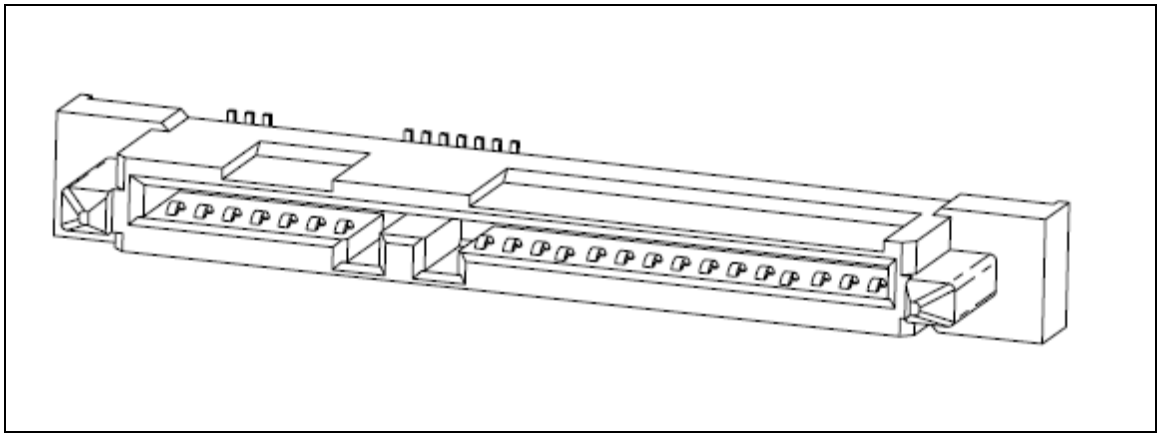


Figure 129 – SATA Express host receptacle connector isometric drawing

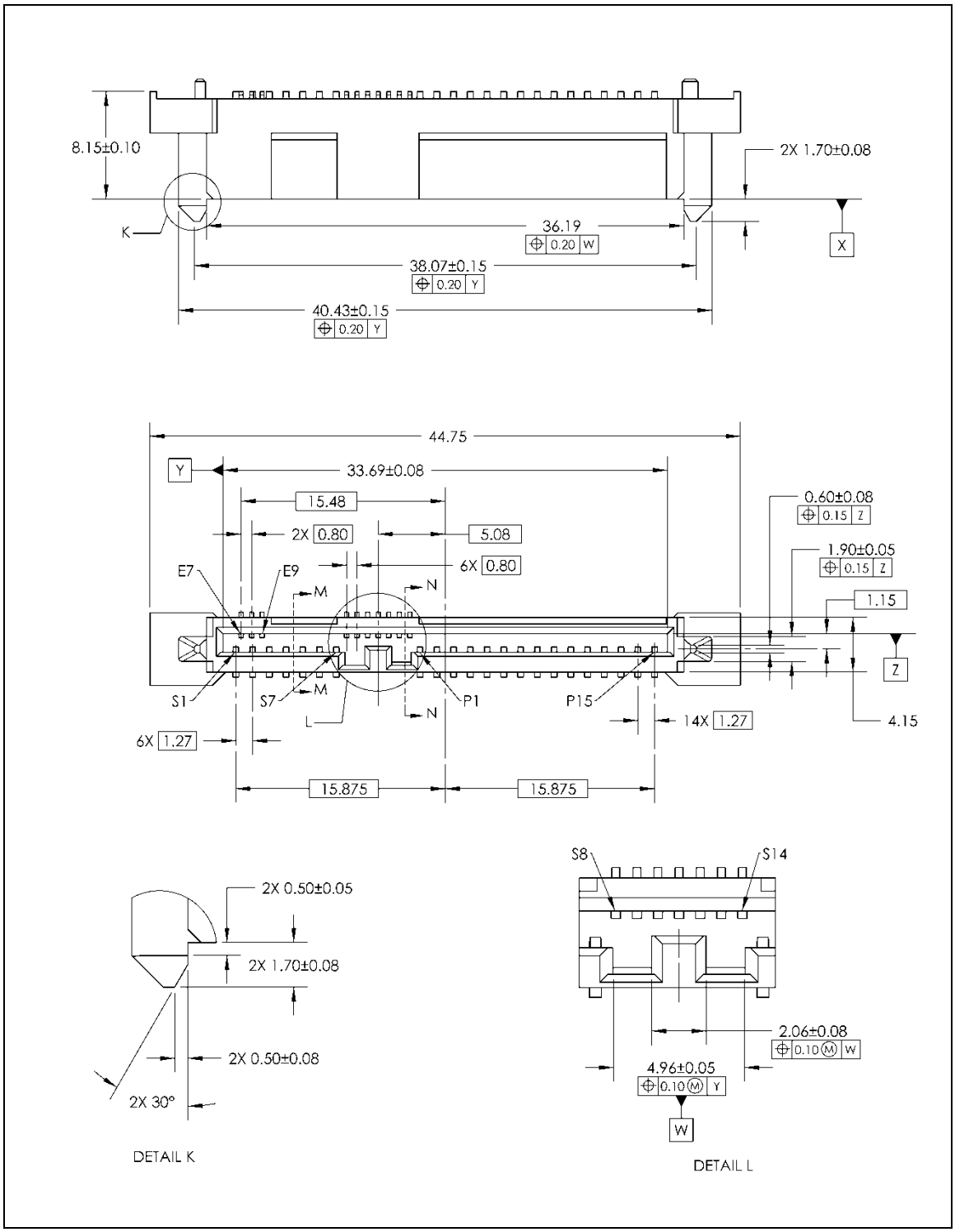


Figure 130 – SATA Express host receptacle connector drawing (part 1 of 2)

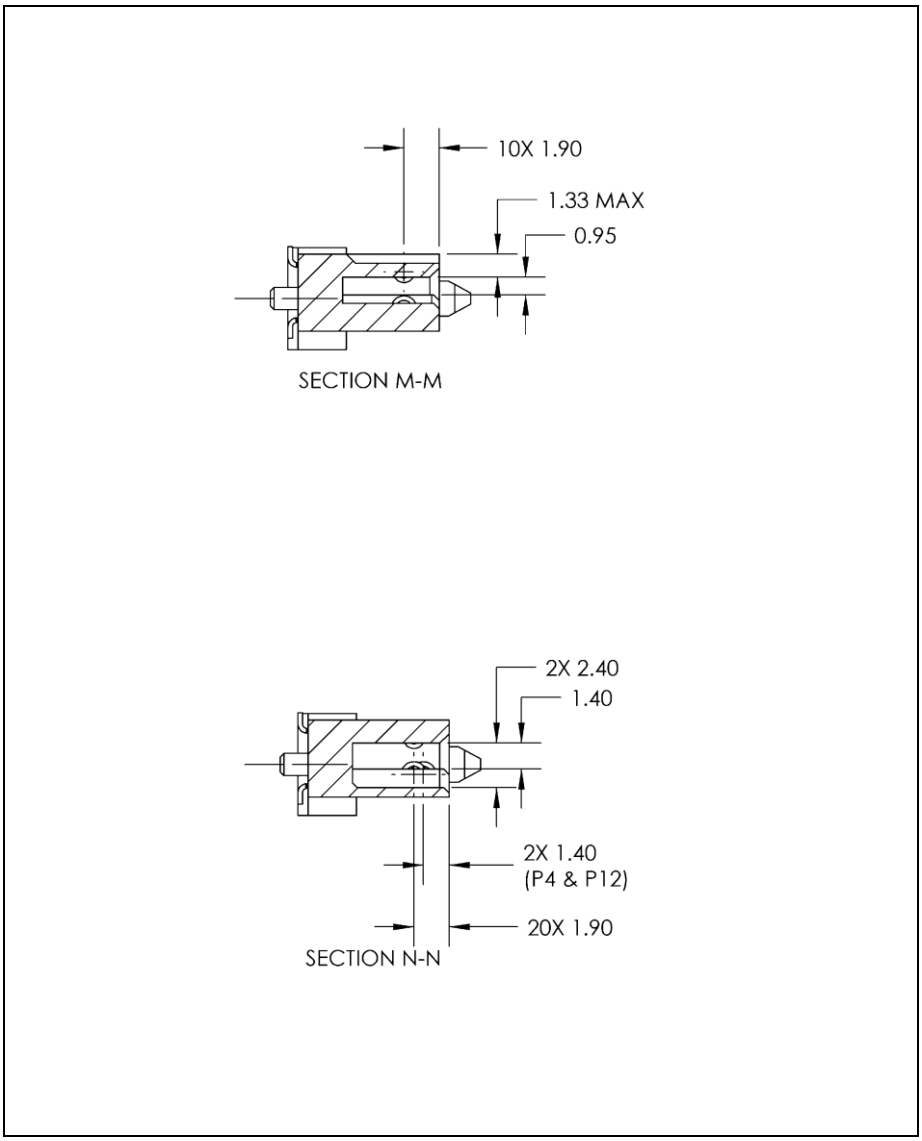


Figure 130 – SATA Express host receptacle connector drawing (part 2 of 2)

6.10.10 SATA Express host plug connector

The SATA Express host plug connector may be considered an extension of the SATA host plug connector. Figure 131 illustrates the SATA Express host plug connector. It is essentially two SATA host plug connectors joined together, plus an additional section for sidebands.

The SATA Express host plug connector mates with the following connectors:

a) the SATA Express host cable receptacle connector; and
b) the SATA single lane cable receptacle connector.

The pinout of the host plug connector in isometric view is shown in Figure 131 and the pinout table is given in Table 32. It contains only a subset of the pins listed in

Table 30. The reserved pin shall be N/C (i.e. no connect) on the host, cable and device.

Table 33 shows the SATA Express cable wire connection. Figure 132 and Figure 133 are the SATA Express host plug connector mechanical drawings.

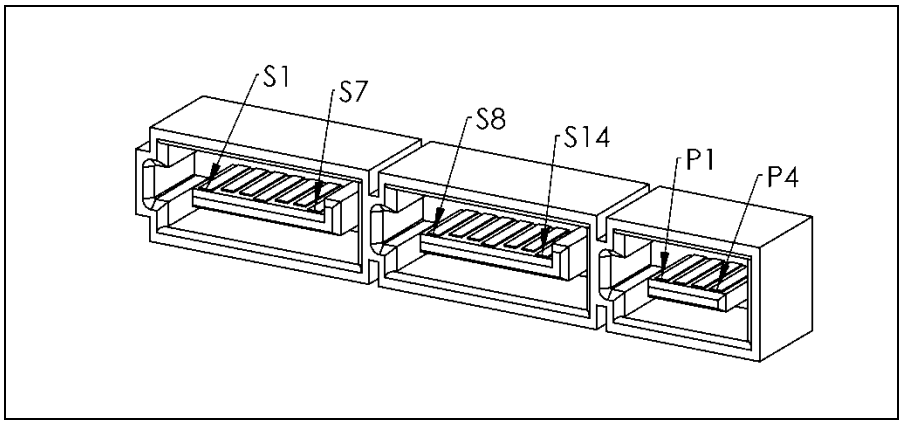


Figure 131 – SATA Express host plug connector pinout (isometric view)

Table 32 – SATA Express host plug connector pin list

Pin position	Name	Signal Description	Usage	Signal Direction	Mating
S1	GND	Ground	PCIe/SATA		First
S2	PETp0 / A0+	PCIe Lane 0 Tx / SATA Port 0 Tx pair	PCIe/SATA	Host Output	Second
S3	PETn0 / A0-				
S4	GND	Ground	PCIe/SATA		First
S5	PERn0 / B0-	PCIe Lane 0 Rx / SATA Port 0 Rx pair	PCIe/SATA	Input to Host	Second
S6	PERp0 / B0+				
S7	GND	Ground	PCIe/SATA		First
S8	GND	Ground	PCIe/SATA		First
S9	PETp1 / A1+	PCIe Lane 1 Tx / SATA Port 1 Tx pair	PCIe/SATA	Host Output	Second
S10	PETn1 / A1-				
S11	GND	Ground	PCIe/SATA		First
S12	PERn1 / B1-	PCIe Lane 1 Rx / SATA Port 1 Rx pair	PCIe/SATA	Input to Host	Second
S13	PERp1 / B1+				
S14	GND	Ground	PCIe/SATA		First
P1	Reserved	Reserved for future use	PCIe/SATA		First
P2	PERST#	PCIe reset	PCIe only	Host Output	First
P3	CLKREQ# / DEVSLP	PCIe L1 PM Substate / SATA device sleep	PCIe/SATA		First
P4	IFDet	Interface detect	PCIe/SATA	Input to Host	First

Table 33 – SATA Express cable wire connection

Host Cable Receptacle Connector		Wire	Device Cable Receptacle Connector	
Pin position	**Name**		**Name**	**Pin position**
S1	GND	Drain wire	GND	S1
S2	PETp0 / A0+	Shielded differential pair	PETp0 / A0+	S2
S3	PETn0 / A0-		PETn0 / A0-	S3
S4	GND	Drain wire	GND	S4
S5	PERn0 / B0-	Shielded differential pair	PERn0 / B0-	S5
S6	PERp0 / B0+		PERp0 / B0+	S6
S7	GND	Drain wire	GND	S7
S8	GND	Drain wire	GND	S8
S9	PETp1	Shielded differential pair	PETp1	S9
S10	PETn1		PETn1	S10
S11	GND	Drain wire	GND	S11
S12	PERn1	Shielded differential pair	PERn1	S12
S13	PERp1		PERp1	S13
S14	GND	Drain wire	GND	S14
P1	Reserved		Reserved	P1
P2	PERST#	Discrete wire	PERST#	P2
P3	CLKREQ# / DEVSLP	Discrete wire	CLKREQ# / DEVSLP	P3
P4	IFDet	Discrete wire	IFDet	P4

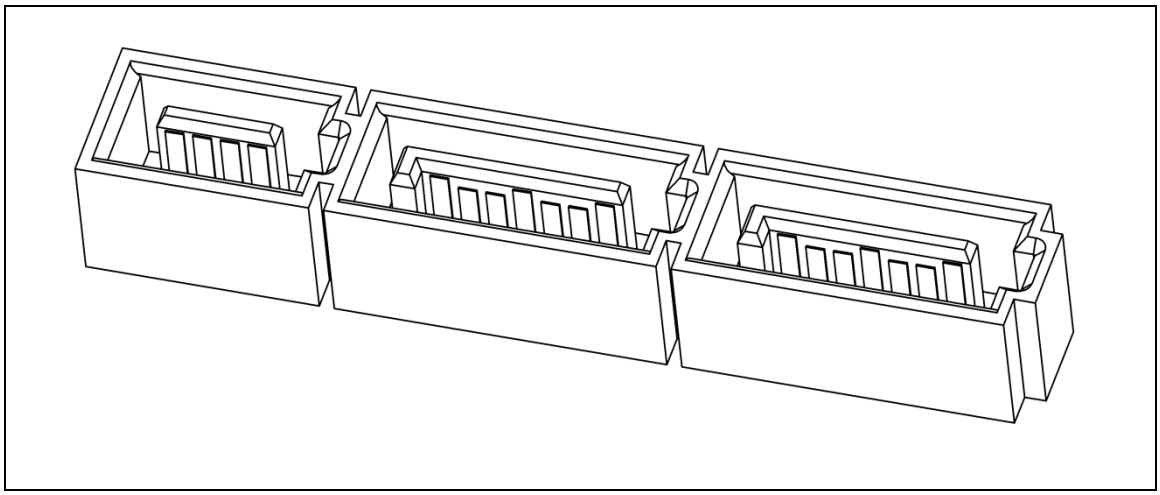


Figure 132 – SATA Express host plug connector isometric drawing

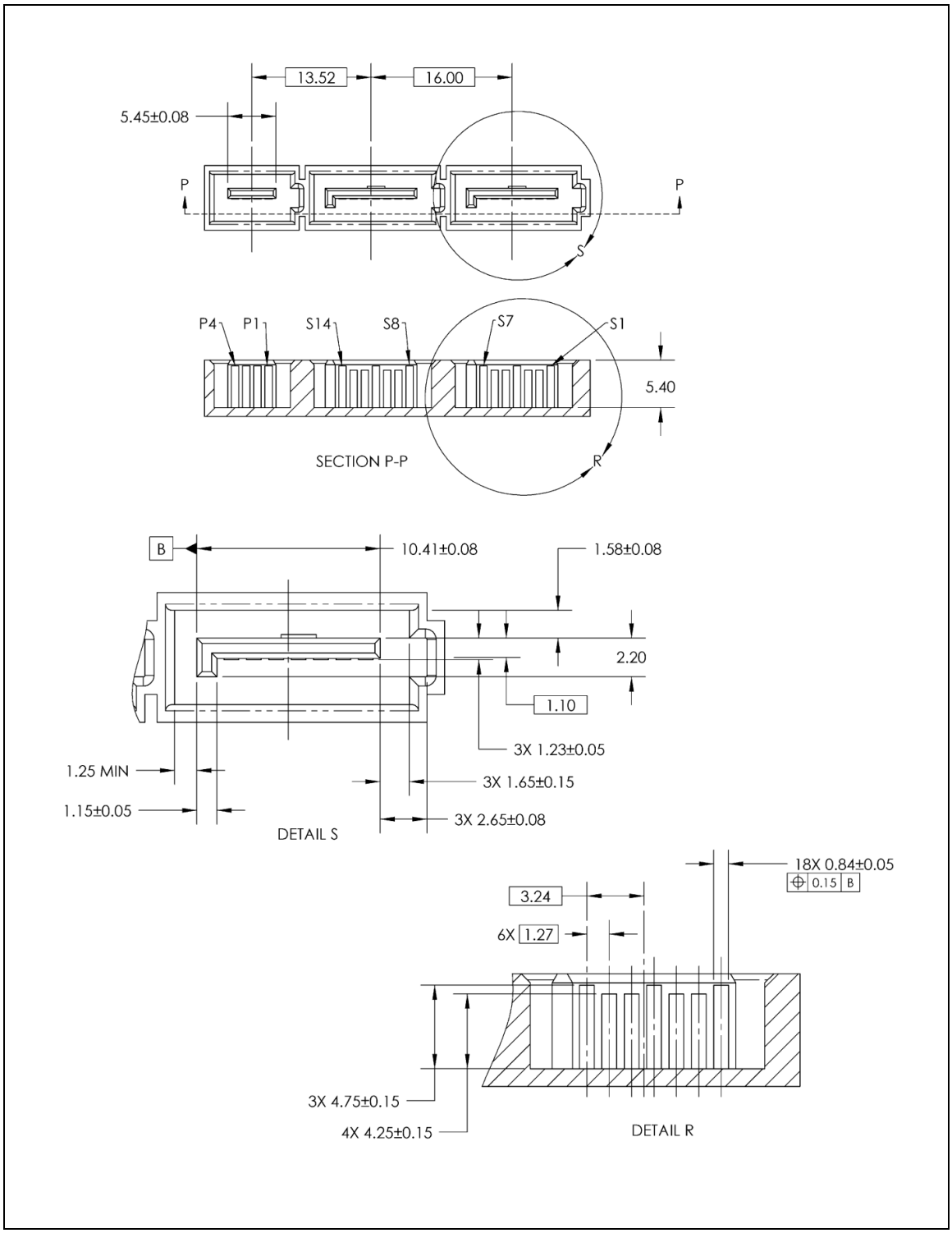


Figure 133 – SATA Express host plug connector drawing

6.10.11 SATA Express host cable receptacle connector

The SATA Express host cable receptacle connector mates with only the SATA Express host plug connector. Figure 134 shows an isometric view of the SATA Express host cable receptacle

connector. Figure 135 defines the mating interface of the connector. It may be considered an extension of the SATA signal cable receptacle connector.

The following descriptions apply to Figure 135:

a) power is not delivered through the cable; and
b) the cable retention solution is similar to SATA (i.e., both friction lock and metal latches may be used).

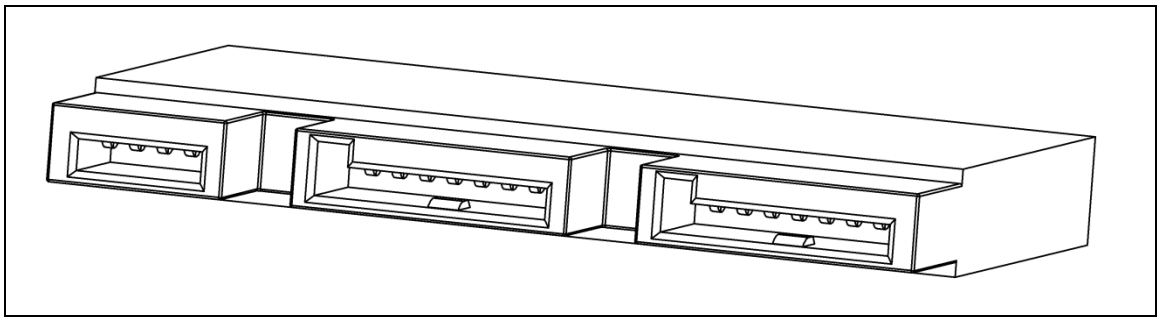


Figure 134 – SATA Express host cable receptacle connector isometric drawing

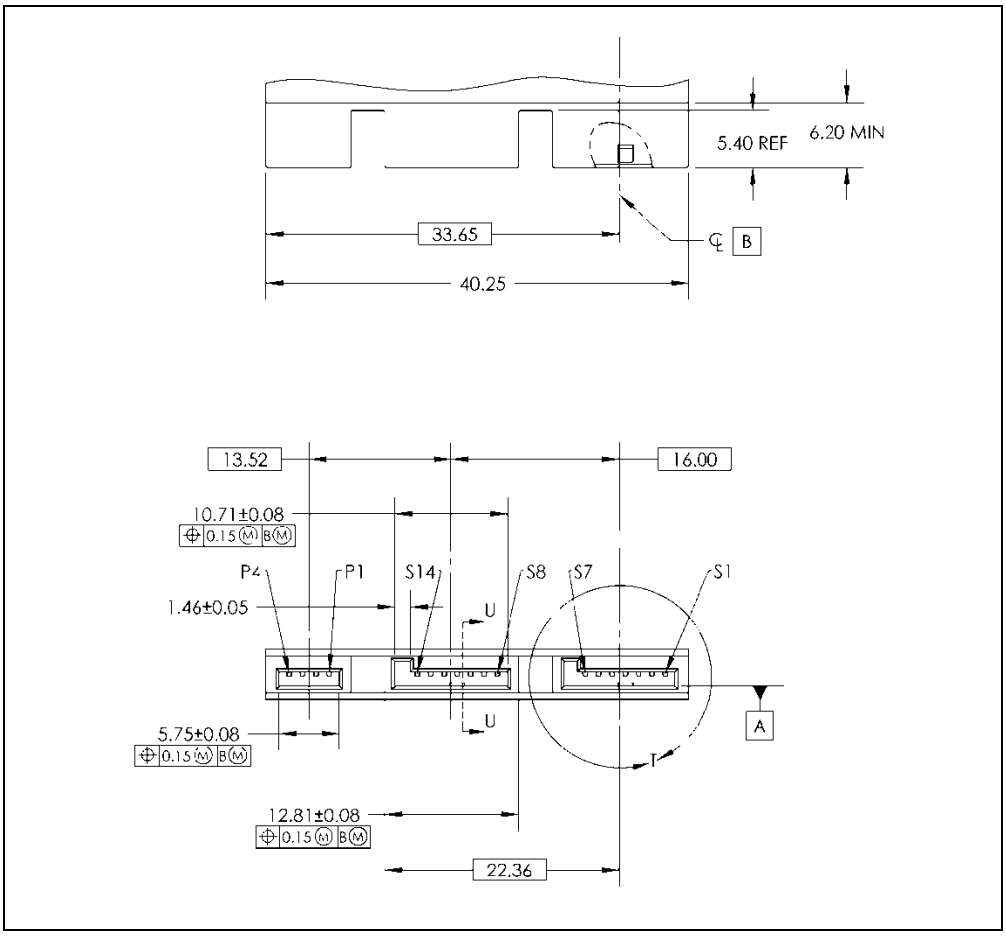


Figure 135 – SATA Express host cable receptacle connector drawing (part 1 of 2)

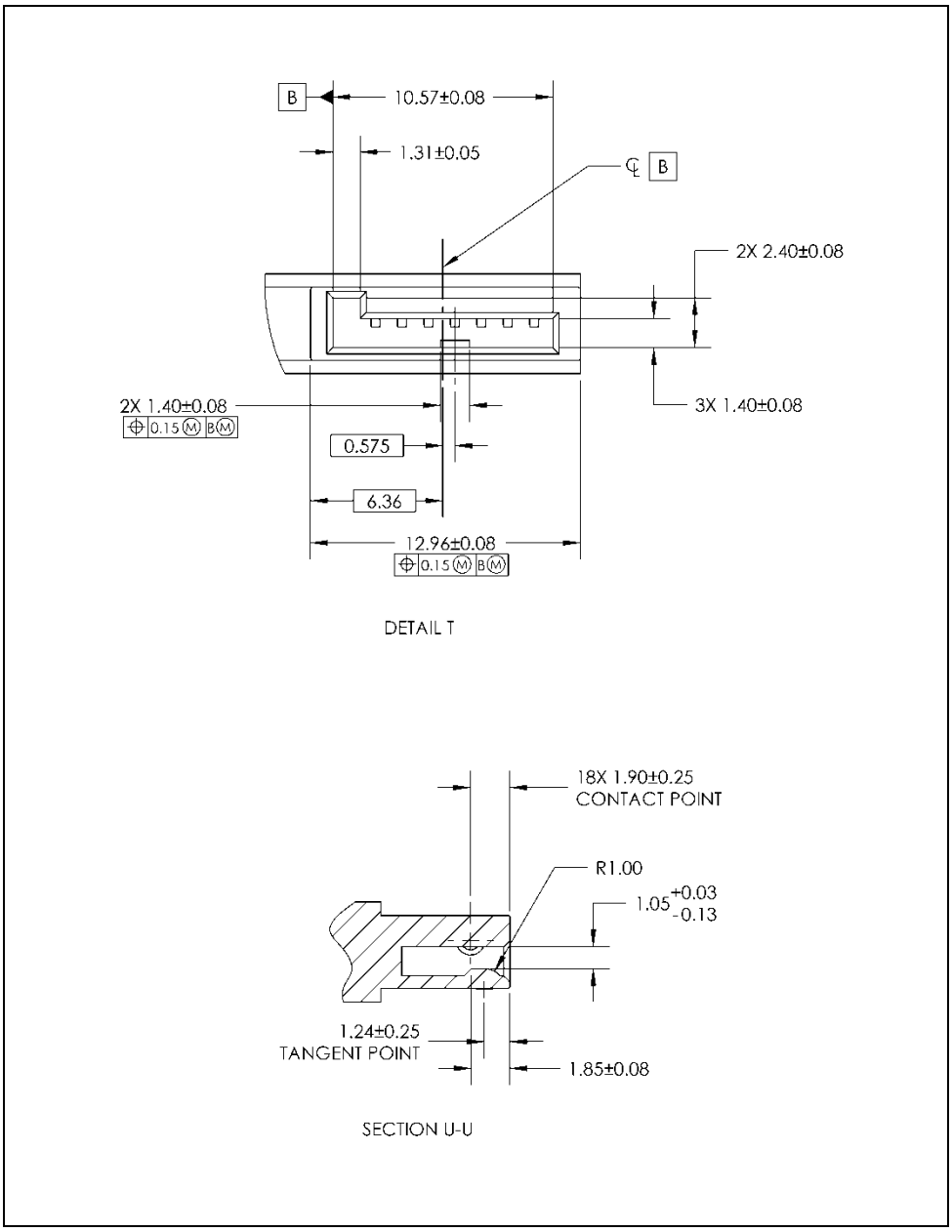


Figure 135 – SATA Express host cable receptacle connector drawing (part 2 of 2)

6.10.12 SATA Express power dongle connector

To allow the coexistence of SATA and SATA Express, reuse of the existing power supply infrastructure is desired in many usage models. The SATA Express device cable receptacle connector defined in 6.10.8 supports two PCIe lanes and power. This is different from SATA, that allows two separate connectors, one for signal and one for power, to plug into the device plug connector. To enable use of the standard SATA power cable receptacle connector with the SATA Express device cable receptacle connector, a power dongle is needed, as illustrated in Figure 136.

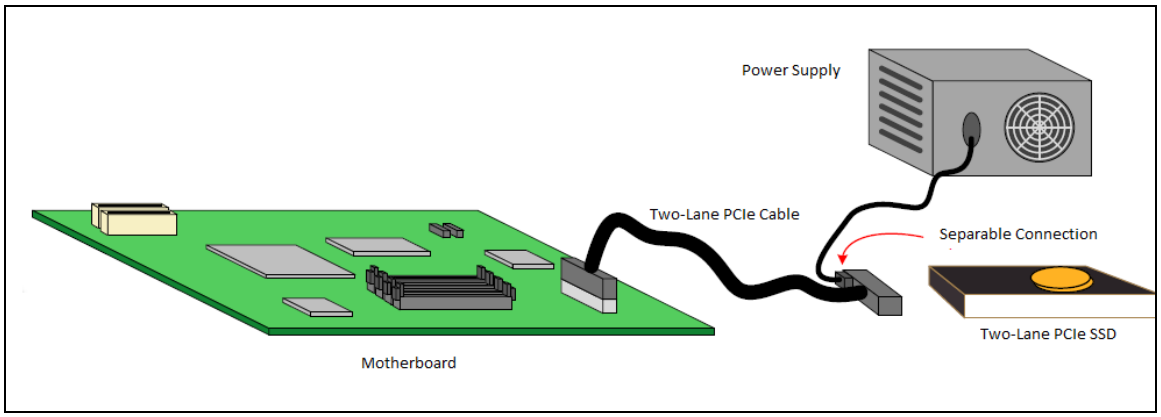


Figure 136 – Example of power dongle usage

Implementation of the power dongle on the SATA Express device cable receptacle connector is vendor specific. The example in Figure 137 allows the standard 15 pin SATA power cable receptacle connector to mate directly to the SATA Express device cable receptacle connector. The example in Figure 138 provides a pigtail power dongle from the SATA Express device cable receptacle connector. Implementation is not limited to the configurations illustrated in these two examples.

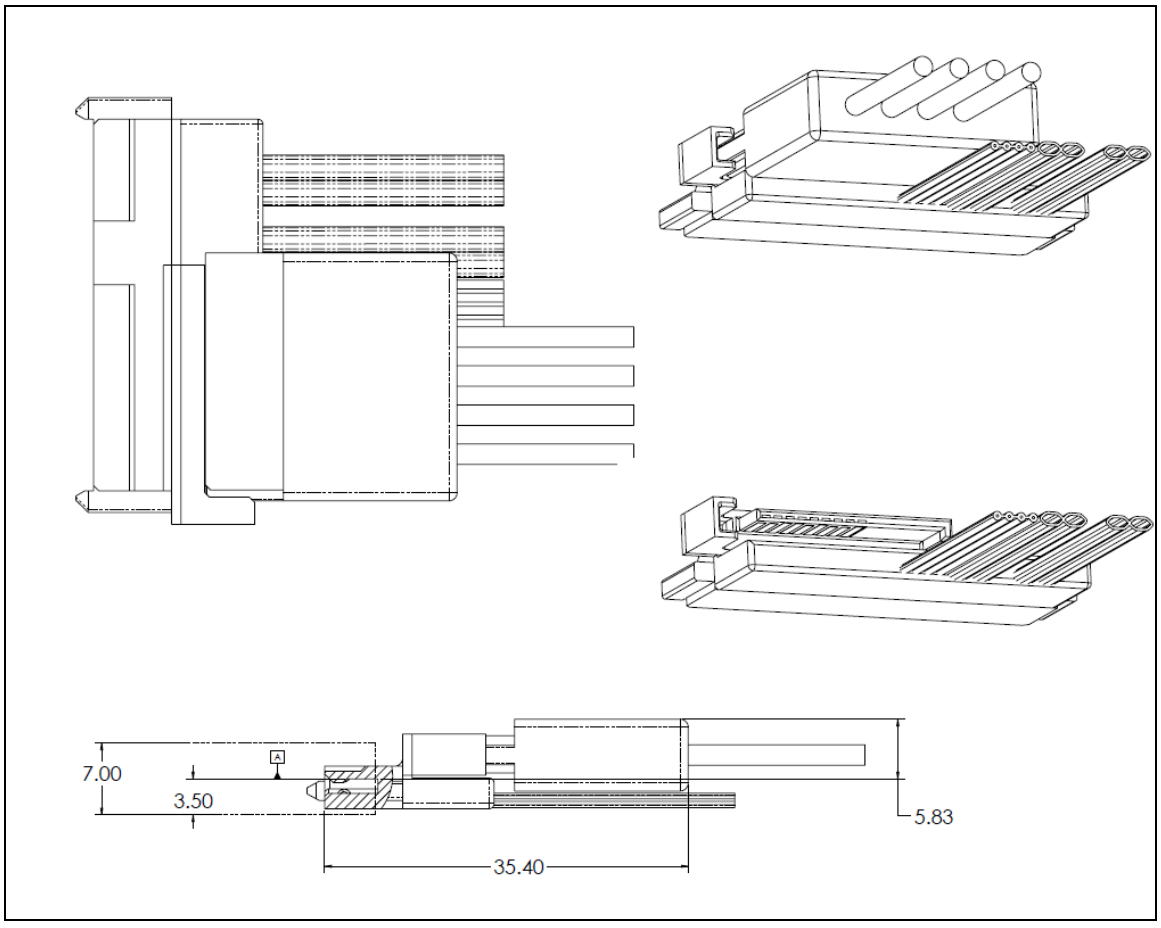


Figure 137 – Example power dongle to be used with the 15 pin SATA power connector

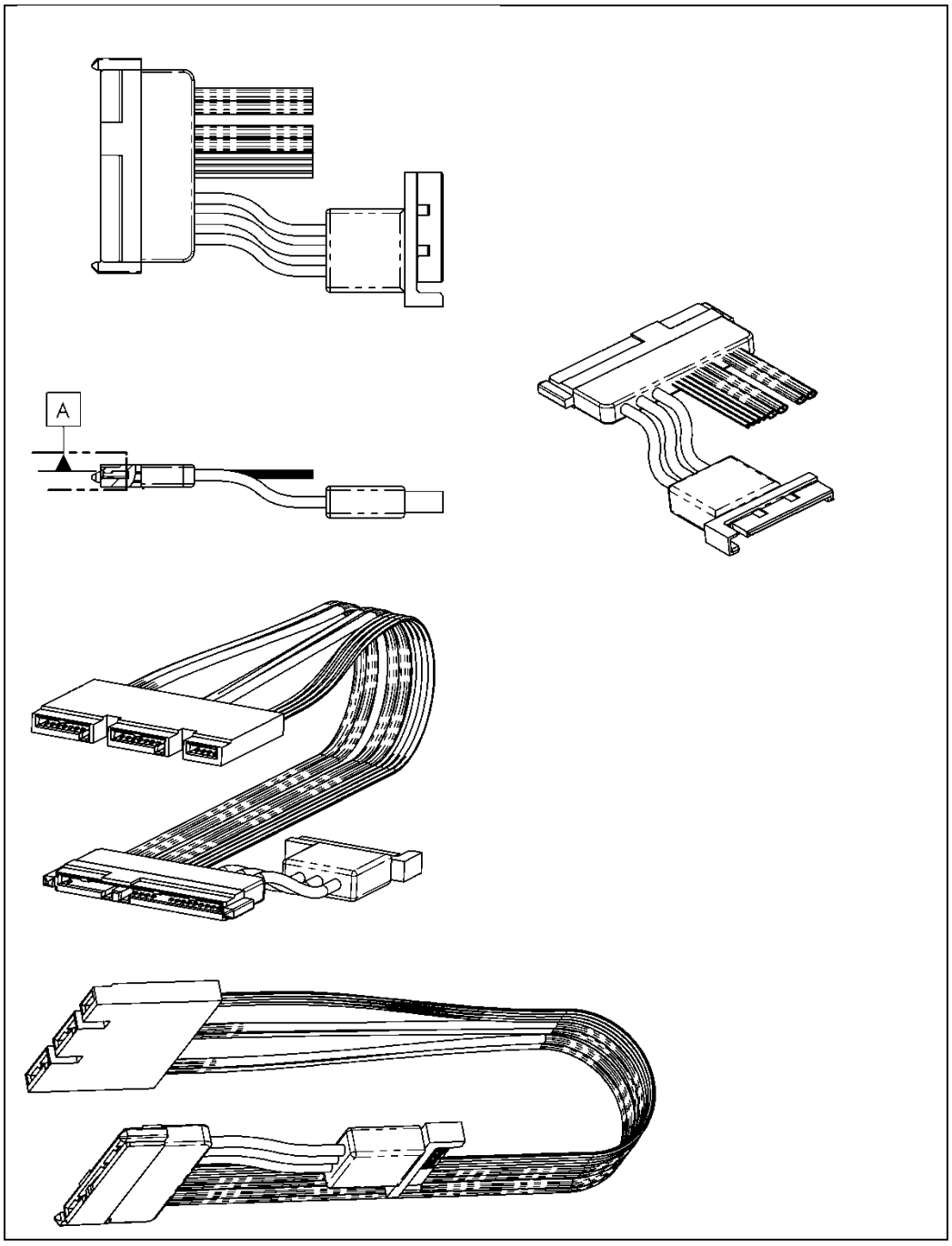


Figure 138 – Example power pigtail to be used with the 15 pin SATA power connector

6.10.13 SATA Express connector intermateability summary

The connector intermatability summary is provided in Table 34 for each of the connectors defined in SATA Express.

Table 34 – SATA Express connector intermatability summary

Connector	Intermateable with
SATA Express Host Receptacle Connector	SATA Express device plug connector (SATA spec) [a] SATA device plug connector (SATA spec) [a]
SATA Express Device Plug Connector	SATA Express host receptacle connector (SATA spec) [a] SFF-8639 backplane receptacle connector [b] SATA Express device cable receptacle connector (SATA spec) [a] SAS/SAS MultiLink backplane and cable receptacle connector [c]
SATA Express Host Plug Connector	SATA Express host cable receptacle connector (SATA spec) [a] 7-pin SATA single lane cable receptacle connector(s) [a]
SATA Express Host Cable Receptacle Connector	SATA Express host plug connector (SATA spec) [a]
SATA Express Device Cable Receptacle Connector	SATA Express device plug connector (SATA Spec) [a] SATA Device Plug Connector (SATA spec) [a]
Power Dongle (on the backside of the SATA Express device cable receptacle connector)	SATA power cable receptacle connector (SATA spec) [a]

[a] Interface is supported.
[b] Interface is supported if muxing of SATA/SAS/PCIe are implemented in SFF-8639 host.
[c] Interface support is beyond the scope of this specification.

6.10.14 SATA Express connector and cable electrical and mechanical requirements

SATA Express electrical requirements shall conform to the PCI Express Specifications. SATA Express cabled applications require the PCIe Phy that supports SRIS.

SATA Express covers only the necessary electrical and mechanical requirements in order for connector and cable manufacturers to make SATA Express cables and connectors. It includes the connector and cable assembly signal integrity requirements. References are provided if requirements are defined in other standards.

6.10.15 SATA Express connector and cable assembly signal integrity requirements

To ensure interoperability of the connectors and cable assemblies, the signal integrity requirements shall conform to PCIe CEM (Card Electromechanical Spec) connector spec, but with modification to the differential insertion loss and intra-pair skew requirements.

The mated cable assembly includes the raw cable plus two mated connectors on both ends of the cable. It is required that the mated cable assembly have a differential insertion loss not exceeding -4.2 dB to 4 GHz and -10.0 dB to 6 GHz. In addition, the insertion loss curve should be free of resonance up to 4.5 GHz.

The mated cable assembly shall have a different intra-pair skew ≤ 10 picoseconds, as measured with a differential Time Domain Transmission (TDT) (risetime = 50 picoseconds, measured from the 20 % threshold to 80 % threshold). The delays from each line of the pair are measured at the 50 % voltage crossing, and the delay difference from the two lines is reported as the intra-pair skew.

NOTE 14 - Connector PCB footprint is one of the major sources of impedance mismatches, mainly from the parasitic capacitance and inductance of a through-hole (TH) or SMT pad. The following recommendations are provided to minimize connector PCB launch mismatches:

a) for through-holes, a relatively small finished hole size of 0.61 mm (24 mil) is recommended with appropriate pad and anti-pad sizes (e.g., a 0.91 mm (36 mil) pad and a 1.52 mm (60 mil) differential anti-pad); and
b) for SMTs, the smallest pad that meets Design For Manufacturability (DFM) rules is recommended. Voiding the ground plane underneath the signal pads is recommended to remove excess parasitic capacitance.

The cable wire termination area is another major source of impedance mismatches. Careful wire management with minimum cable shield and dielectric strip-off is recommended.

6.10.16 SATA Express connector and cable shielding requirements for EMI

Since RefClk is not present on the cable with the SRIS architecture, there is no additional connector or cable shielding requirements beyond what is done for the SATA connector and cable. System designers should pay particular attention to RFI (Radio Frequency Interference), avoiding the placement of the SATA Express cable near wireless radios since this may cause RFI issues.

6.10.17 SATA Express connector and cable assembly DC electrical requirements

The connector and cable assembly direct current (DC) electrical requirements are the same as those specified for other SATA connectors in this specification.

6.10.18 SATA Express connector and cable assembly mechanical and environmental requirements

The connector and cable assembly mechanical and environmental requirements are the same as those specified for other SATA connectors in this specification.

6.11 External cables and connectors

6.11.1 External single lane

The External Single Lane system provides a single lane connection between a PC/Laptop and a commodity storage device using Gen1i/Gen2i Serial ATA devices. This interface is for the use of an external device that resides outside the PC chassis, similar to USB or 1394 hard drives and optical drives. While this does not exclude other usages, the requirements are derived based on this usage model.

Power is supplied to the external storage device via a separate means that is outside the scope of this specification. This separate means is expected to be similar to power delivery for USB or 1394 external drives.

This section defines the external interconnect, compliance points, and associated electrical requirements/parameters for device interoperability.

The primary implementation is:

a) an HBA connected to a shielded external connector. A buffer IC is required to interface to a Gen1i/Gen2i Serial ATA host unless the Serial ATA host is Gen1m/Gen2m compliant and designed for direct external connection;
b) a shielded Serial ATA cable designed for external usage; and

c) a Serial ATA device enclosure with a corresponding external connector. A buffer IC is required to interface to a Gen1i/Gen2i Serial ATA device unless the Serial ATA device is Gen1m/Gen2m compliant and designed for direct external connection.

This implementation is shown in Figure 139.

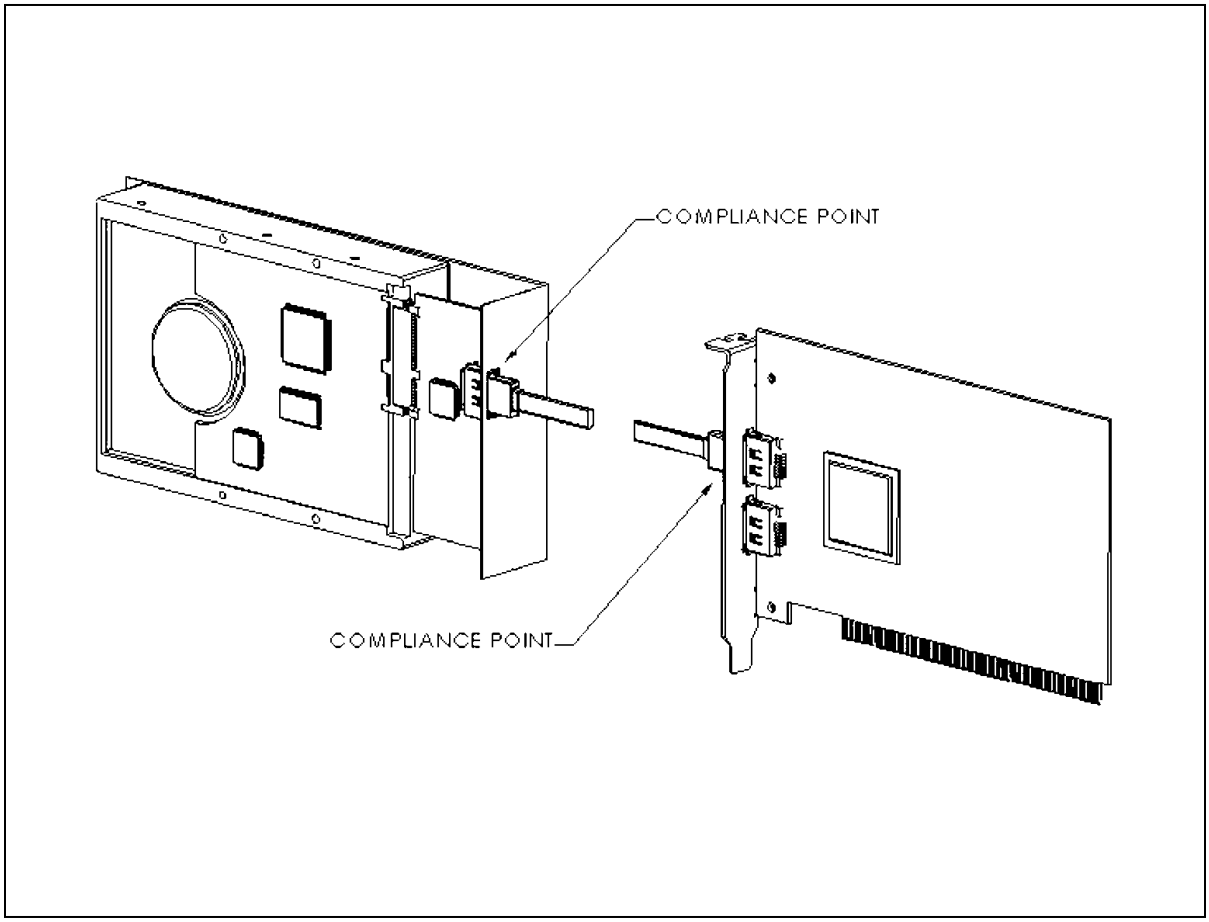


Figure 139 – Usage model for HBA with external cable and single device enclosure

A second potential implementation is a Serial ATA host directly assembled on the motherboard connected to a shielded external connector via a pigtail to the motherboard connection. In this implementation, the external Serial ATA cable and the device assembly are similar to Figure 184, but another cable and connector pair between the motherboard and the external cable is introduced, placing an additional discontinuity point between host and device.

There are two compliance points for the External Single Lane system, one at each shielded external connector. As with other cable and connector system descriptions, interconnect between the IC/Phy and the connectors at the mating interface are outside the scope of the definition and are considered part of the delivered Phy solution. Implementations that have additional connections between the Phy/IC and the shielded external connector shall provide such interconnects as part of the engineered solution. For an implementation shown in Figure 140, the compliance points remain at the shielded external connectors.

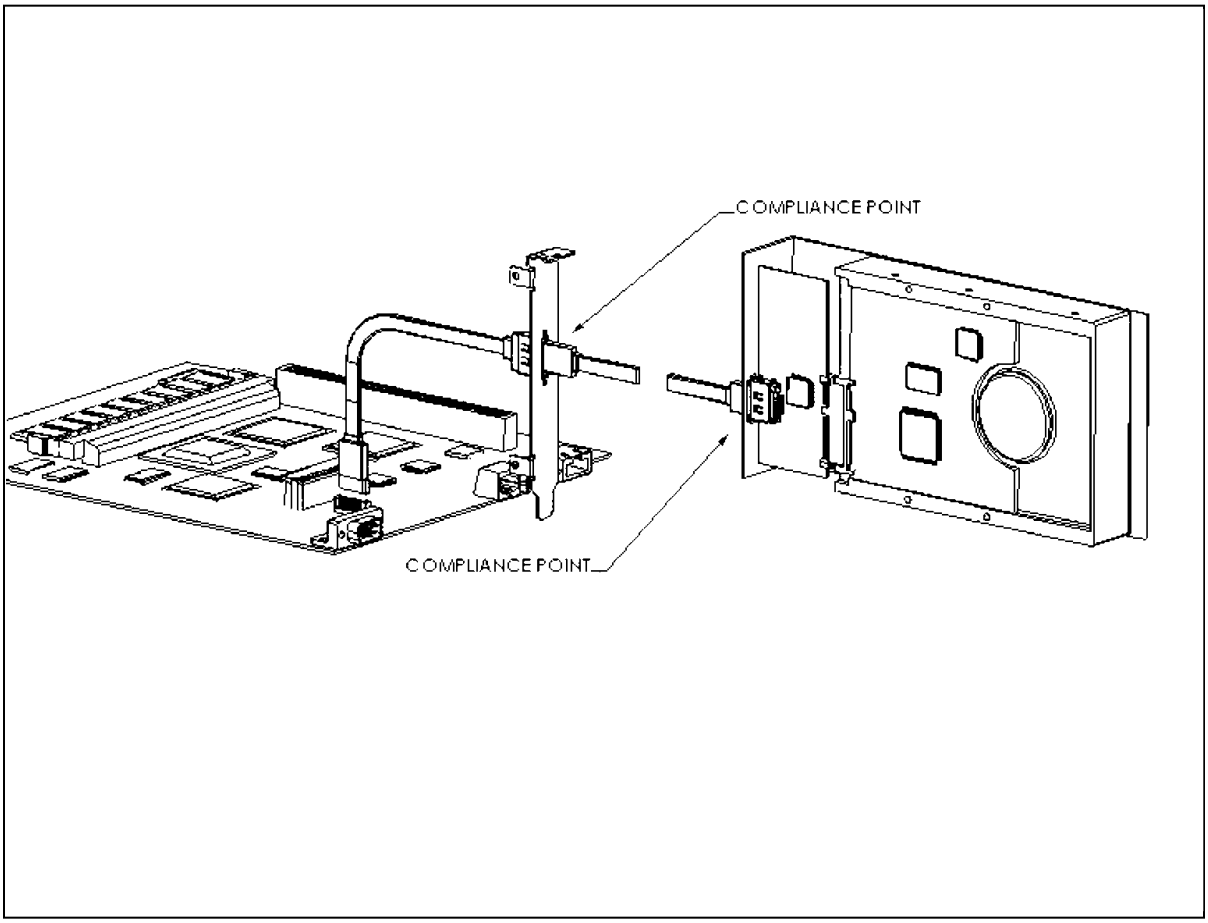


Figure 140 – Usage model for on-board Serial ATA connector with extension cable to external cable to disk

The typical cable length is two meters (six feet); long enough to reach from a floor mounted PC to a device placed on the desktop. The compliance points for both ends of the external cable shall meet the Gen2m electrical specification.

The use of a standard internal Serial ATA host or device not specifically designed for direct external connection requires a buffer IC as part of the implementation. Refer to Table 5 for both host and device connection signal assignments.

> NOTE 15 - Note that since Single Lane External Serial ATA cables are a straight through design, (pin to pin), the host and device external connections, using the same external connector, have the same pin one locations but opposite signal definitions.

6.11.2 External Serial ATA component general descriptions

6.11.2.1 External Serial ATA component general descriptions overview

Five components are defined in this section to support external Serial ATA:

a) a shielded external cable receptacle (see Figure 142) for use with shielded (external Serial ATA) cabling;

b) a fully shielded right angle (RA) PCB mounted SMT plug (see Figure 143), (and reversed pin-out version (see Figure 144));
c) a fully shielded RA PCB mounted through-hole plug (see Figure 145);
d) a fully shielded vertical PCB mounted SMT plug (see Figure 146); and
e) a fully shielded vertical PCB mounted through-hole plug (see Figure 147).

Footprints and recommended panel cutouts are included to encourage greater interoperability from multiple vendors.

The external cable connector is a shielded version of the internal single lane connector as defined in clause 0, with these basic differences:

a) the External connector has no “L” shaped feature, and the guide features are vertically offset and reduced in size. This prevents the use of unshielded internal cables in external applications;
b) to prevent ESD damage, the insertion depth is increased from 5 mm to 6.6 mm and the contacts are mounted further back in both the receptacle and plug; and
c) the retention features are springs built into the shield on both the top and bottom surfaces.

External Serial ATA Connector renderings (see Figure 141).

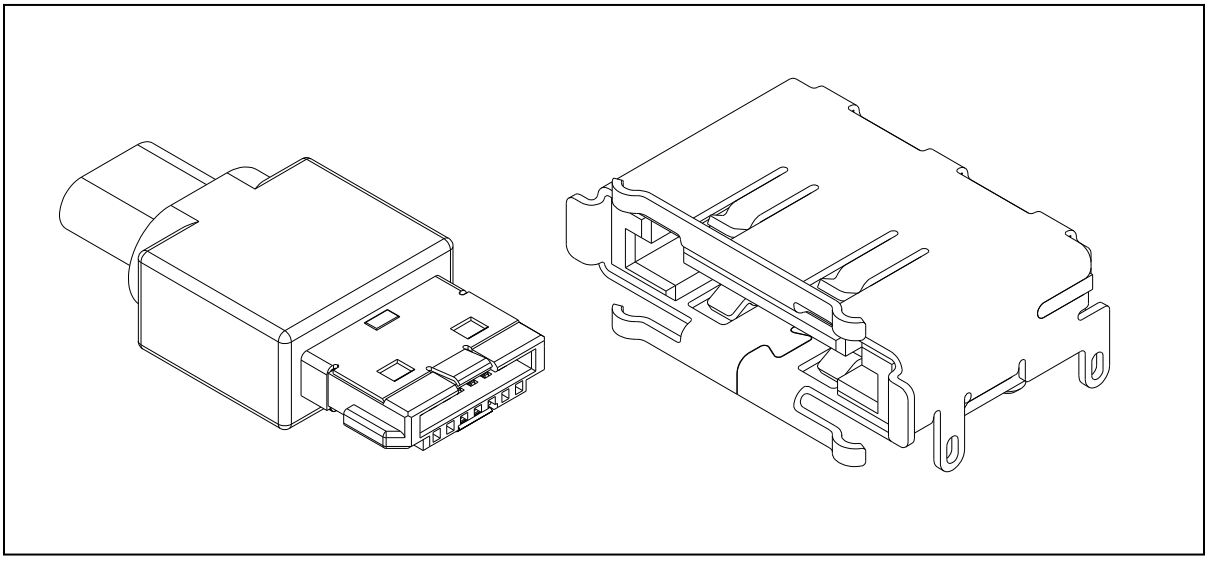


Figure 141 – Renderings of External Serial ATA cable receptacle and right angle plug

6.11.2.2 External Serial ATA connector mechanical drawings

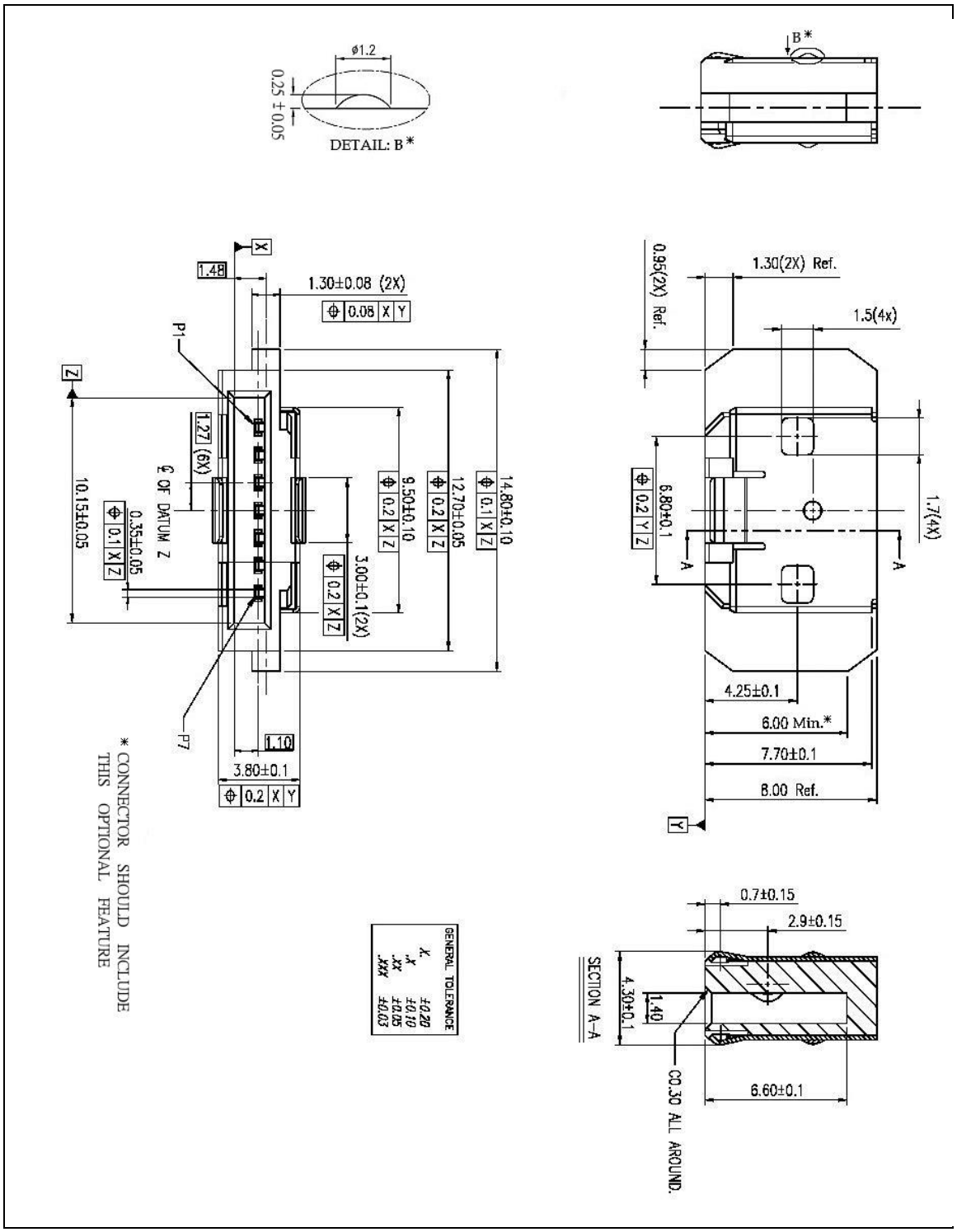


Figure 142 – Mechanical dimensions of External Serial ATA cable receptacle assembly

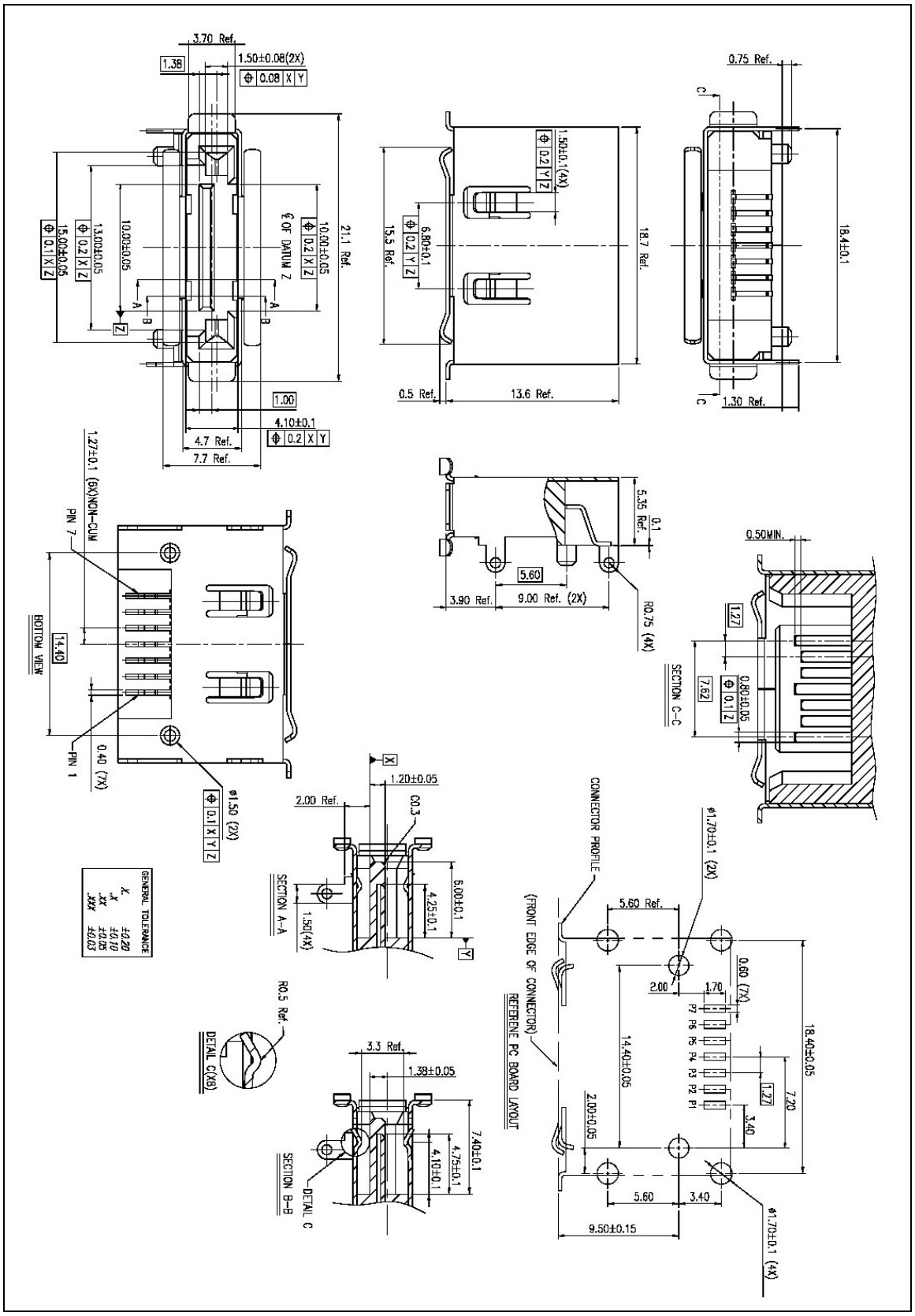


Figure 143 – Mechanical dimensions of External Serial ATA RA SMT plug

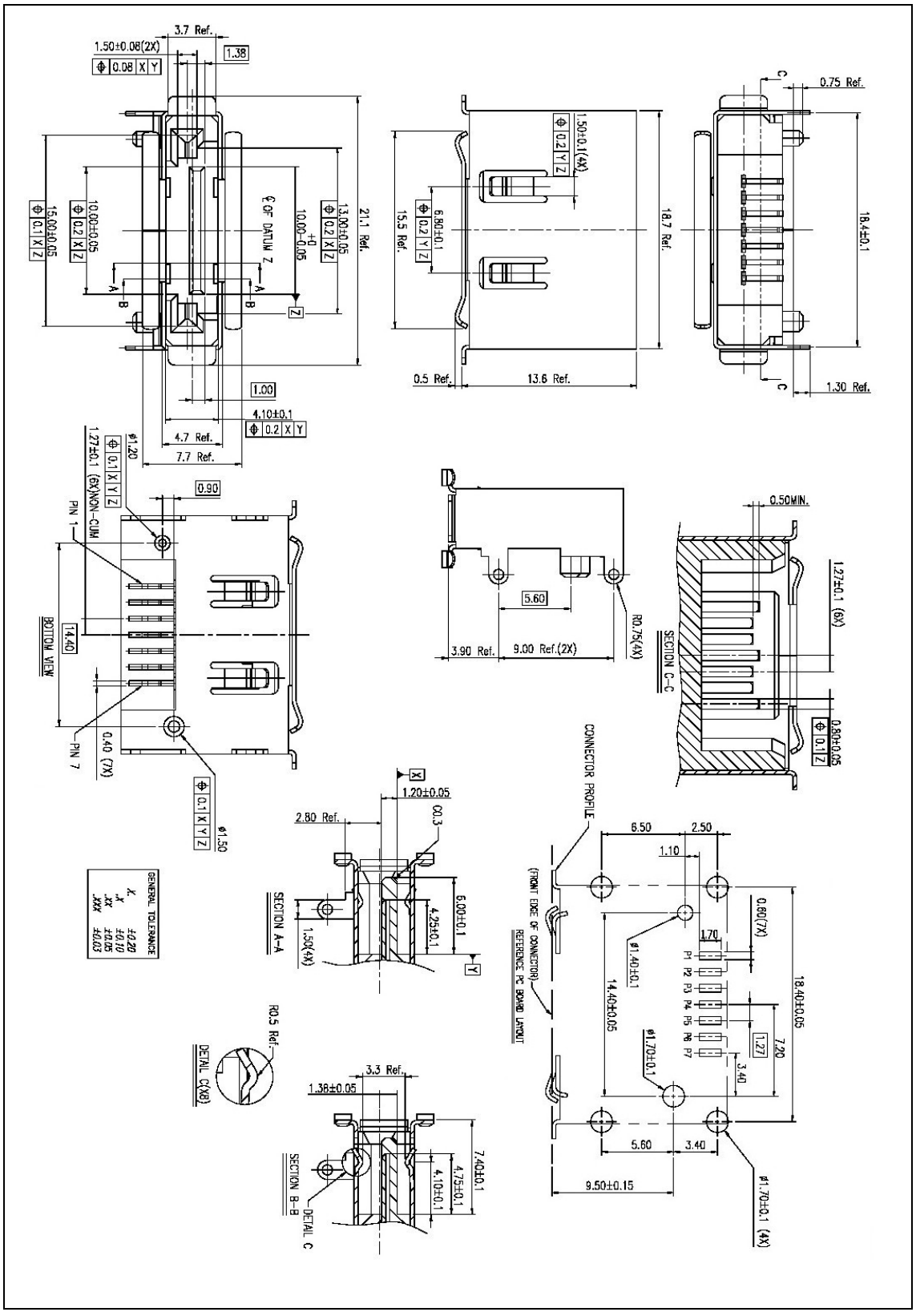


Figure 144 – Mechanical dimensions of External SATA RA SMT plug – reversed pin out

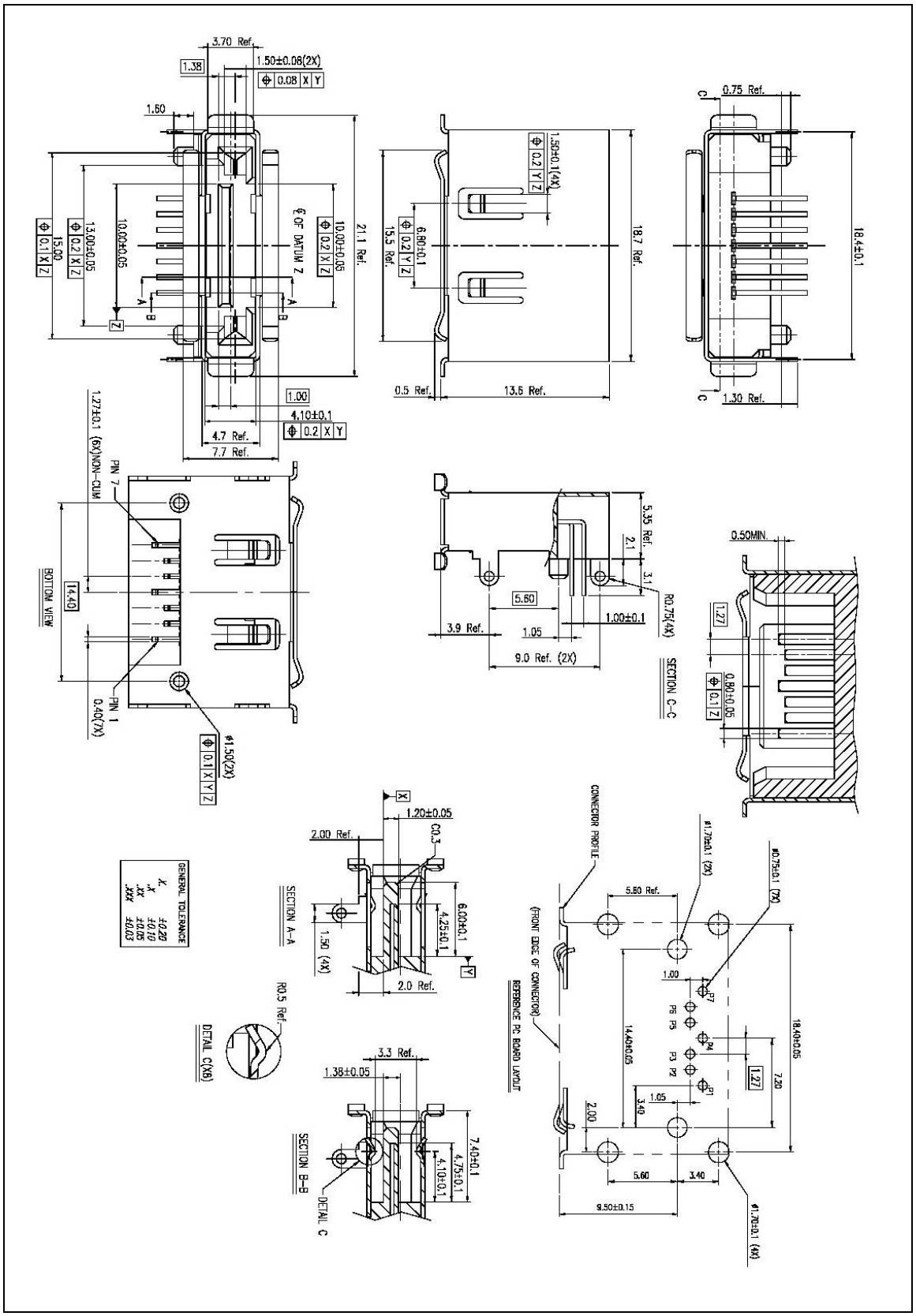


Figure 145 – Mechanical dimensions of External Serial ATA RA through-hole

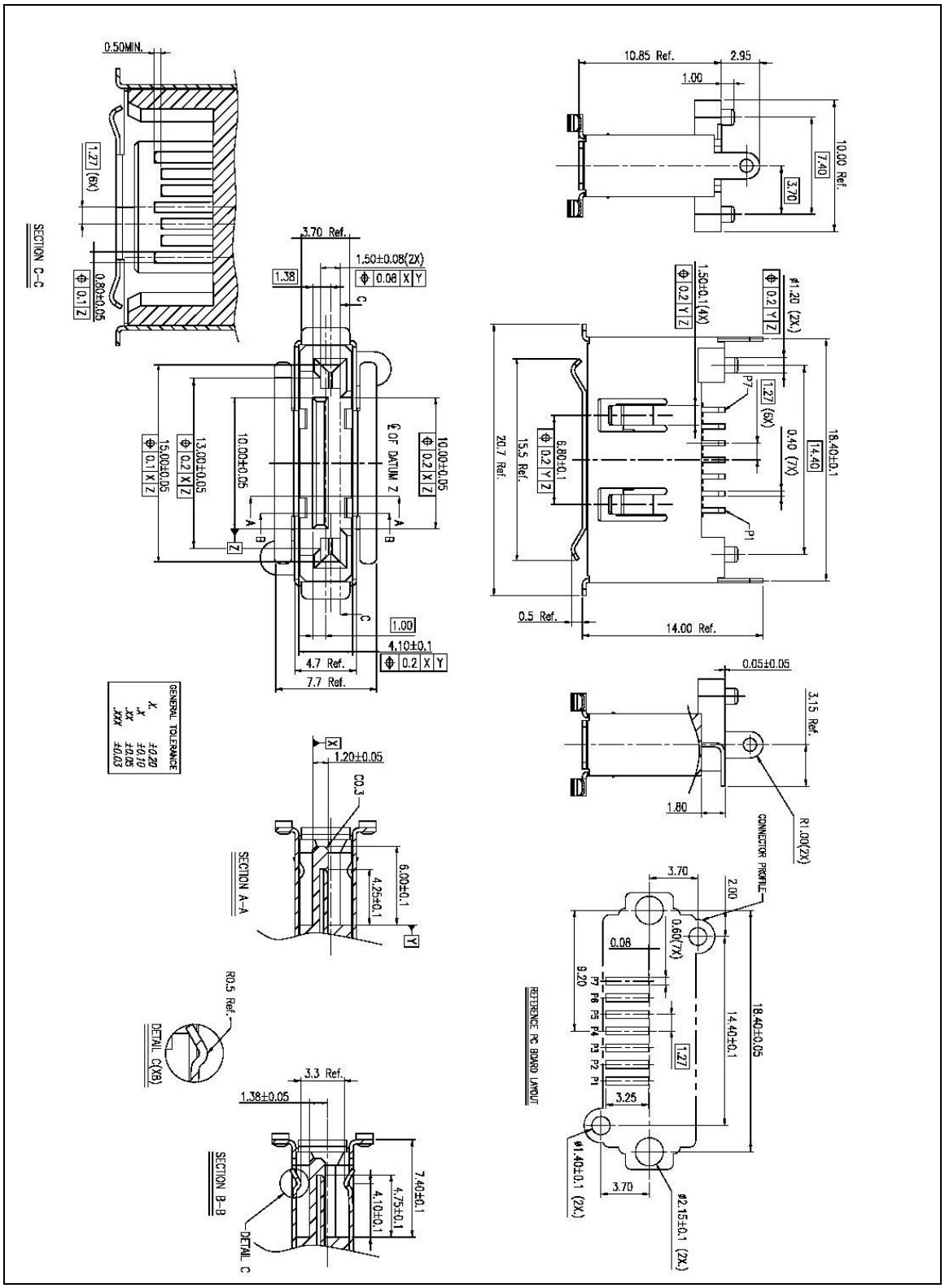


Figure 146 – Mechanical dimensions of External Serial ATA vertical SMT plug

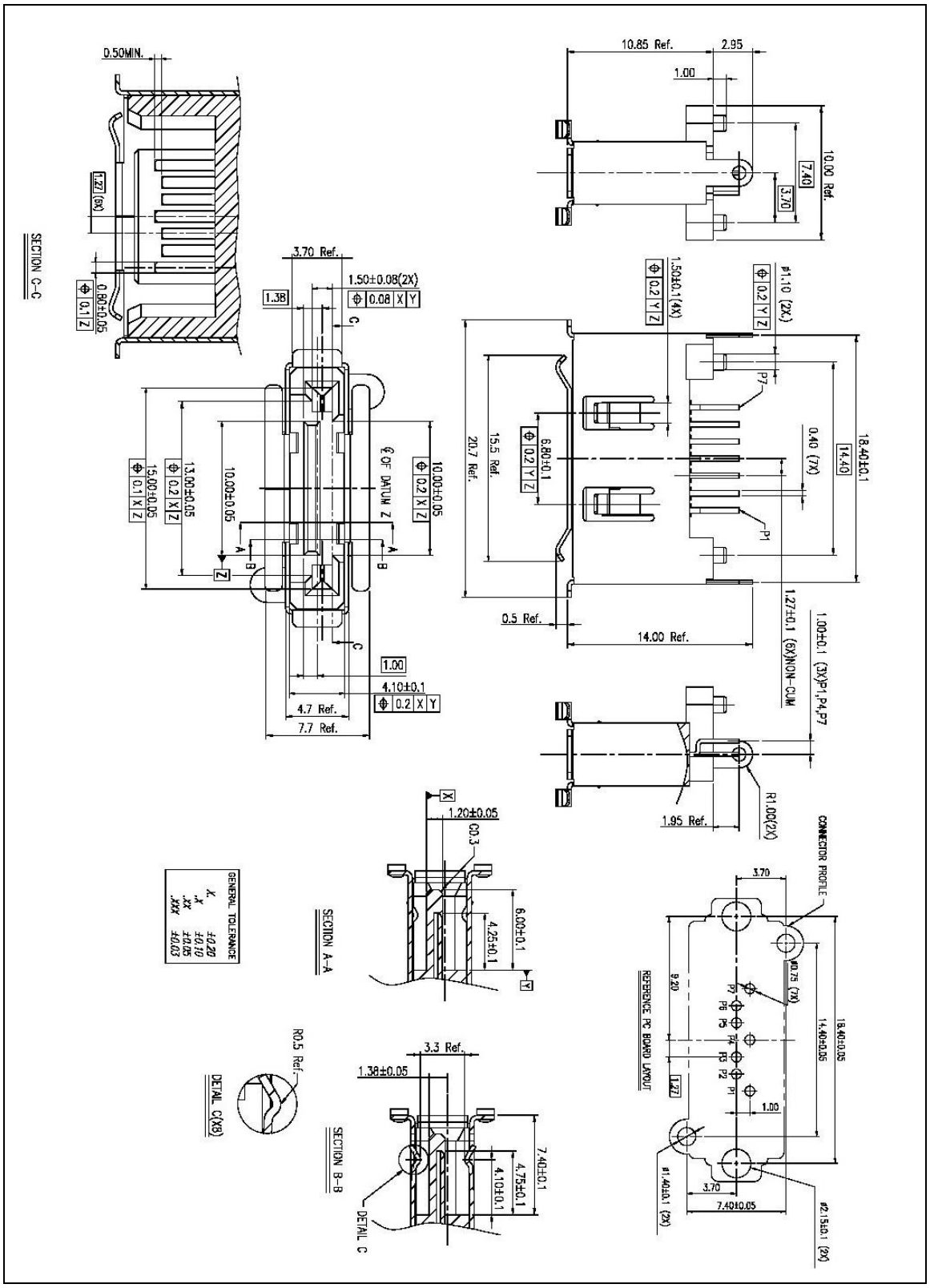


Figure 147 – Mechanical dimensions of External Serial ATA vertical through-hole plug

6.11.2.3 External Serial ATA electrical requirements

The external cable assembly shall meet the electrical characteristics defined in Table 37.

The Single Lane External Serial ATA Data Interface Phy electrical performance shall comply with the following:

a) electrical characteristics defined in Table 37, Gen1m or Gen2m at the shielded external connector compliance points;
b) support hot plugging and non-powered device attachment; and
c) AC coupling is required at the device interface and recommended at the host interface.

6.11.2.4 External Serial ATA mechanical requirements

The external connector mechanical performance specifications shall be consistent with the internal single lane connector specifications according to 6.11, with the following exceptions:

a) durability shall be 2 500 cycles with no exposure of the base metal of the signal contacts;
b) insertion force shall be a maximum of 40 N; and
c) removal force shall be a minimum of 10 N at the conclusion of the durability test.

6.11.2.5 External Serial ATA device direct connection requirements

Serial ATA devices may have data interfaces specifically designed for direct connection in the Single Lane External Serial ATA environment without requiring a buffer IC. The data interface of the Single Lane External Serial ATA device shall comply with all external Serial ATA mechanical requirements according to 6.11.2.4.

The Phy electrical performance shall comply with the following:

a) electrical characteristics as defined in 6.12, Gen1m or Gen2m at the shielded external connector compliance points;
b) support hot plugging and non-powered device attachment; and
c) AC coupling is required at the device interface and recommended at the host interface.

6.11.3 External Multilane

6.11.3.1 External Multilane overview

This section defines standard cable assemblies and headers for connecting multiple Serial ATA channels from a RAID HBA to an intelligent backplane in an adjacent just a bunch of disks (JBOD) unit. The RAID HBA and the JBOD units are envisioned as using different power supplies.

This cable/connector set is based on the SFF-8470 specification. The SFF-8470 specification also describes the cable/connector set used by SAS.

6.11.3.2 Multilane cable conformance criteria

6.11.3.2.1 Speed limitations

The External Multilane cable/connector shall be used with Gen1m/Gen2m signal levels only. If Gen1m/Gen2m signal levels are used, the cable length is limited to two meters.

6.11.3.2.2 Electrical parameters

The External Multilane cable assembly operating at Gen1m and Gen2m levels shall meet the electrical characteristics defined in Table 38.

6.11.3.2.3 Mechanical parameters

Detailed mechanical requirements are specified in SFF-8470, reference type 4X with thumbscrews. The PCI add-in card form factor supports two 4X interfaces (see Figure 148).

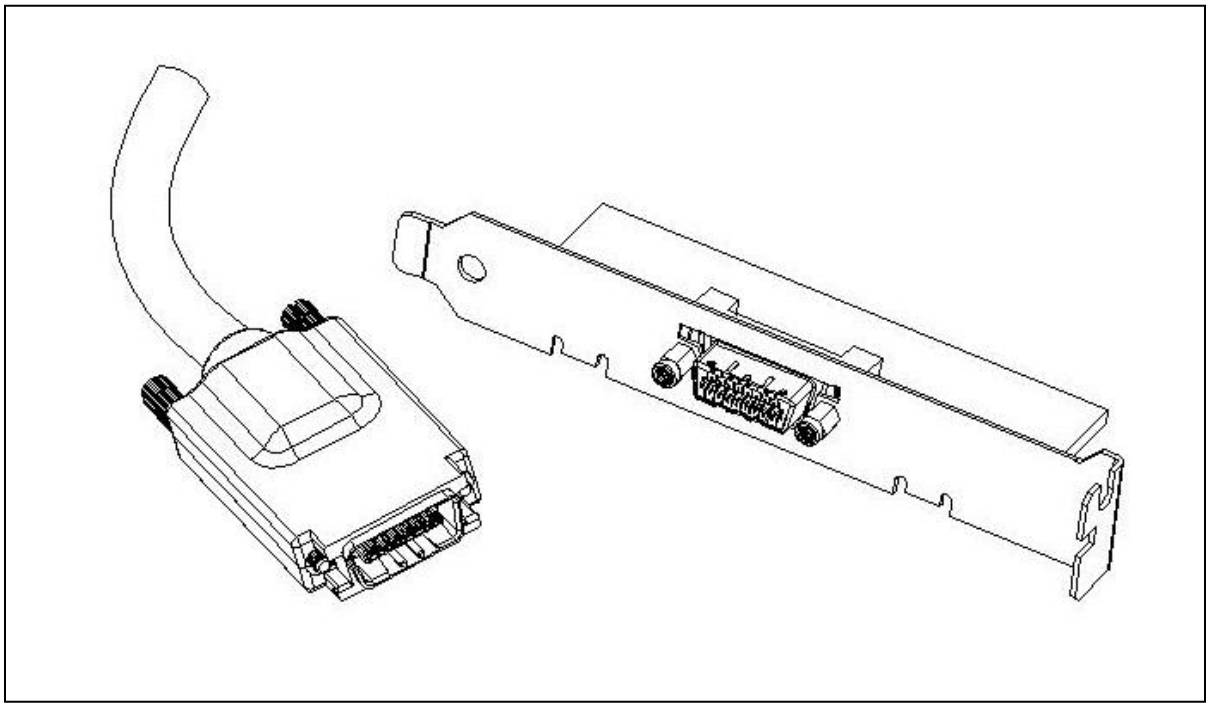


Figure 148 – External Multilane cable and connector

6.11.3.3 Keying requirements

The Serial ATA External Multilane cable/connector may include keying features, a variant from the SFF-8470. The Serial ATA key locations are shown in Figure 149.

Optional keying allows connection between Serial ATA HBAs and JBODs, but disallows connection to SAS HBAs or JBODs (see Figure 150), if the SAS units do not have connectors with key slots. If present, the External Multilane cable connector blocking key locations shall be 3, 4, and 5 and the corresponding mating connector blocking key locations shall be 1, 2, and 6.

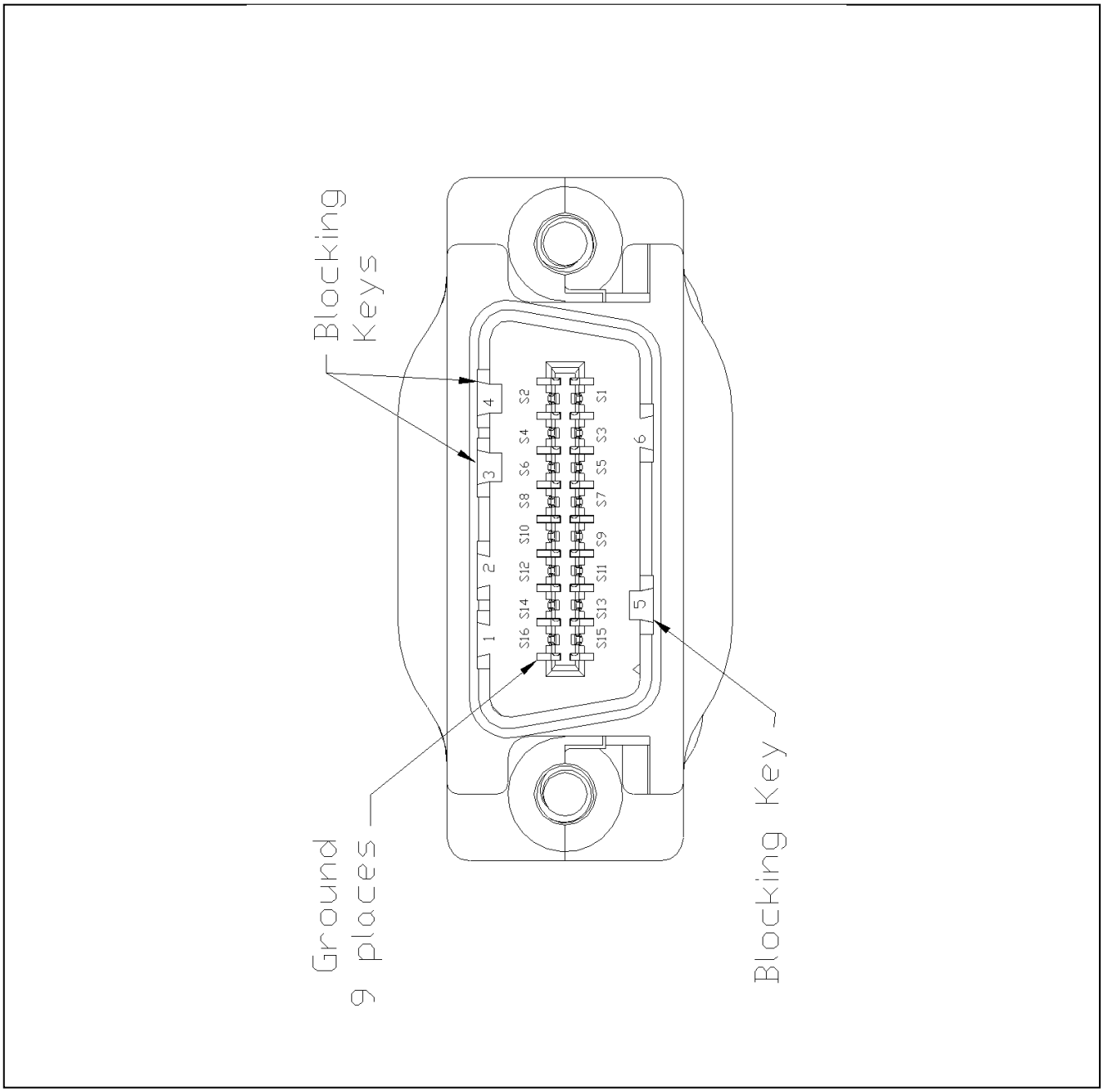


Figure 149 – Multilane cable connector blocking key locations

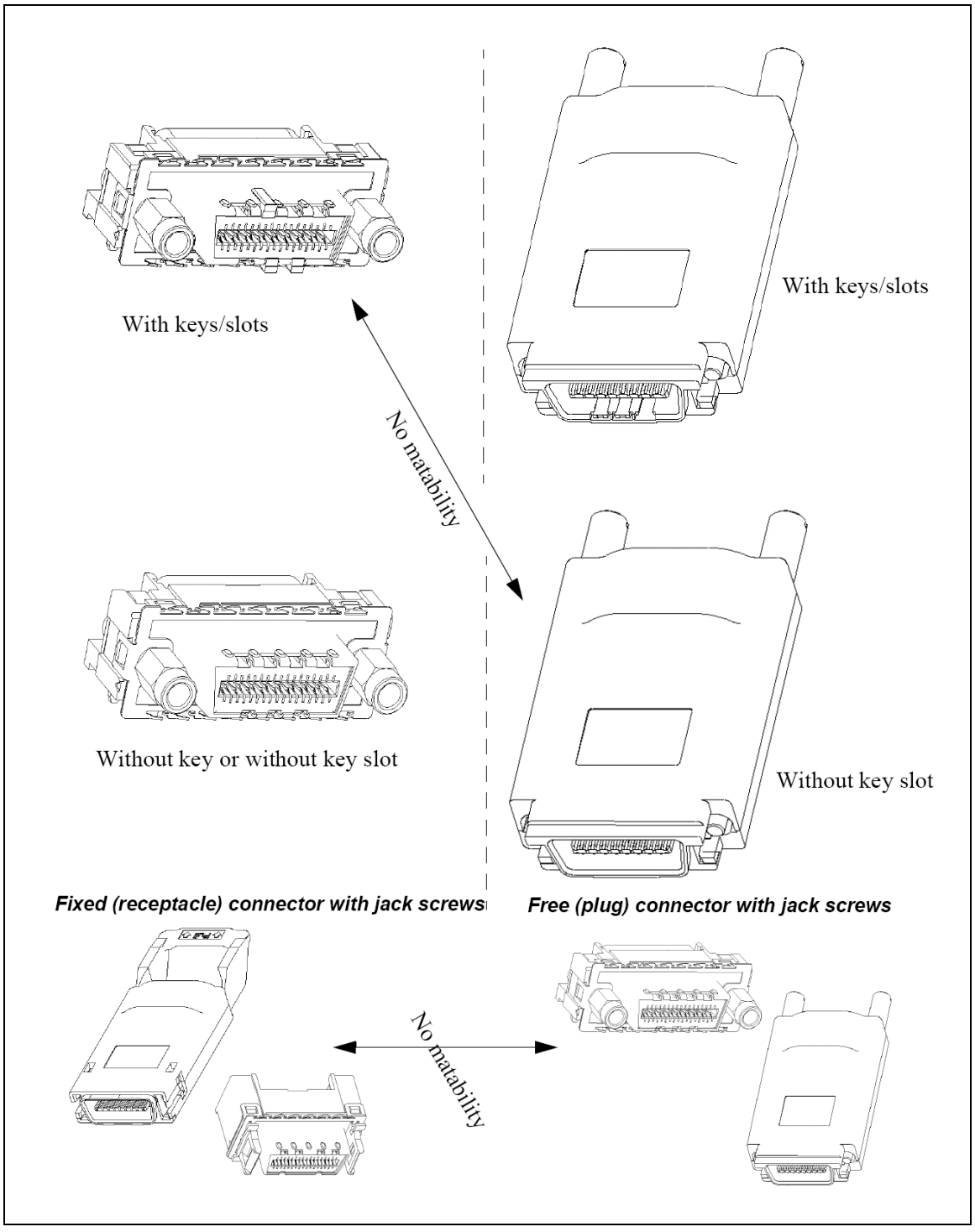


Figure 150 – Plug/receptacle keying

6.11.3.4 Four Multilane pin assignments

See Table 35 for multilane pin assignments.

Table 35 – Multilane pin assignments

Signal	Signal pin to use based on number of physical links supported by the cable			
	One	Two	Three	Four
Rx 0+	S1	S1	S1	S1
Rx 0-	S2	S2	S2	S2
Rx 1+	N/C	S3	S3	S3
Rx 1-	N/C	S4	S4	S4
Rx 2+	N/C	N/C	S5	S5
Rx 2-	N/C	N/C	S6	S6
Rx 3+	N/C	N/C	N/C	S7
Rx 3-	N/C	N/C	N/C	S8
Tx 3-	N/C	N/C	N/C	S9
Tx 3+	N/C	N/C	N/C	S10
Tx 2-	N/C	N/C	S11	S11
Tx 2+	N/C	N/C	S12	S12
Tx 1-	N/C	S13	S13	S13
Tx 1+	N/C	S14	S14	S14
Tx 0-	S15	S15	S15	S15
Tx 0+	S16	S16	S16	S16
SIGNAL GROUND	G1 to G9			
CHASSIS GROUND	Housing			
Key: N/C= Not connected				

6.11.4 Mini SATA External Multilane

6.11.4.1 Mini SATA External Multilane overview

This section defines standard cable assemblies and headers for connecting multiple Serial ATA channels from a RAID HBA to an intelligent backplane in an adjacent JBOD unit. The RAID HBA and the JBOD units are envisioned as using different power supplies.

This cable/connector system is based on the SFF-8086 and SFF-8088 specifications. Both SFF-8086 and SFF-8088 specifications are also used by SAS.

6.11.4.2 Conformance criteria

6.11.4.2.1 Conformance criteria list

The External Multilane cable/connector shall be used with Gen1m/Gen2m signal levels:

a) 4 lanes, SAS/Serial ATA signals;
b) cable length is two meters maximum for Gen1m and Gen2m applications;
c) Rx, Tx, Rx, Tx pin sequencing to minimize crosstalk;
d) ground reference between each pair;
e) performance for 3.0 Gbit/s; and
f) keying features for “m” cables.

6.11.4.2.2 Electrical parameters

The External Multilane cable assembly operating at Gen1m and Gen2m levels shall meet the electrical characteristics defined in Table 38.

6.11.4.2.3 Mechanical parameters

Detailed mechanical requirements are specified in SFF-8086 and SFF-8088.

The Mini SATA External Multilane cables and connectors shall use the 26-circuit version plug and receptacle defined in SFF-8086 and SFF-8088.

The pull-tab for the Mini SATA External Multilane connector, if present, shall be red (Pantone #207).

6.11.4.3 Mini SATA External Multilane keying requirements

The Mini SATA External Multilane cable/connector shall include keying features from SFF-8088. The Serial ATA defined key locations are shown in Figure 151.

Unique keying requirements for x-level signal levels are obsolete. Unique keying requirements for m-level signal levels are shown in Figure 152.

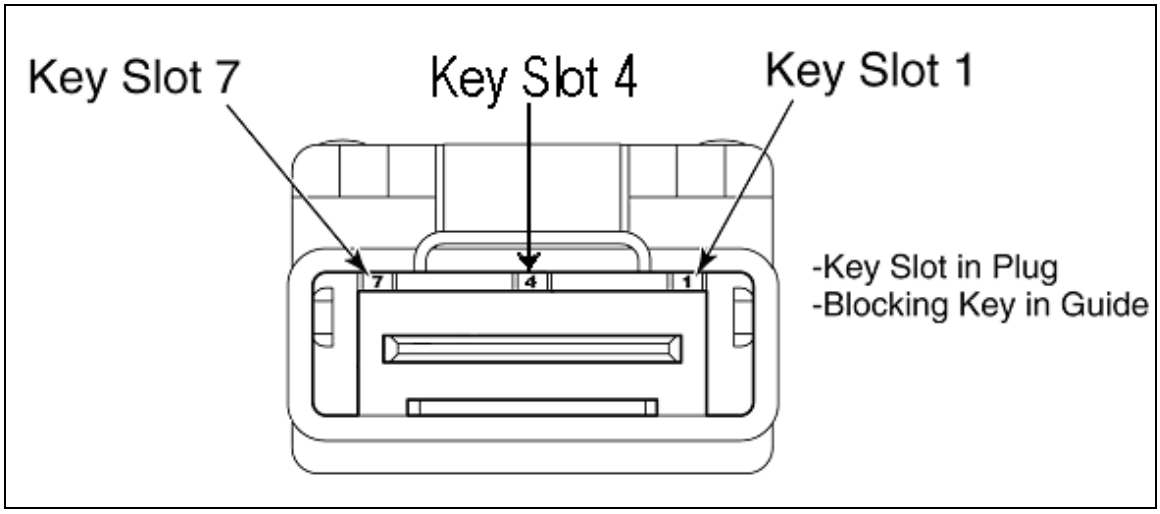


Figure 151 – Mini SATA External Multilane system, key features

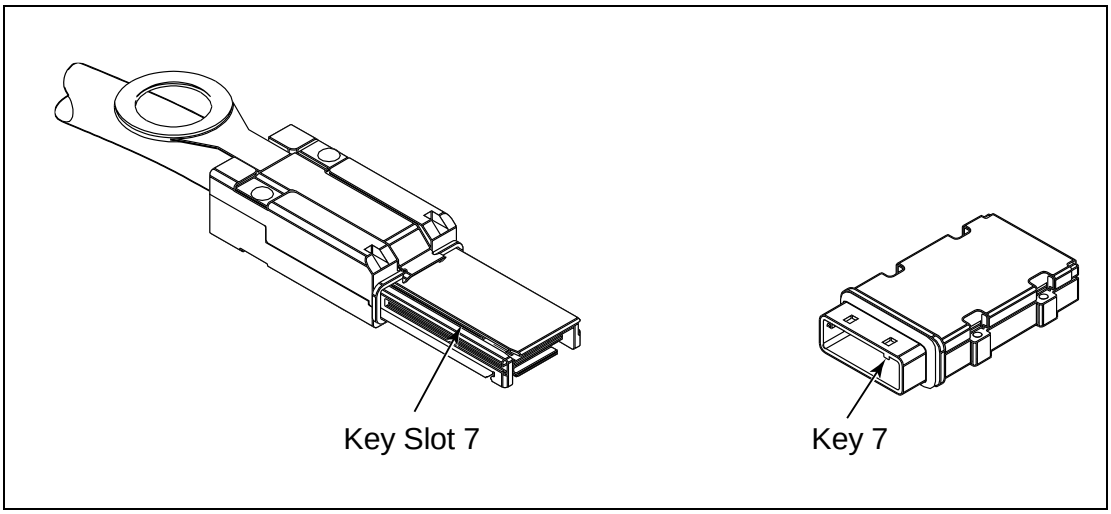
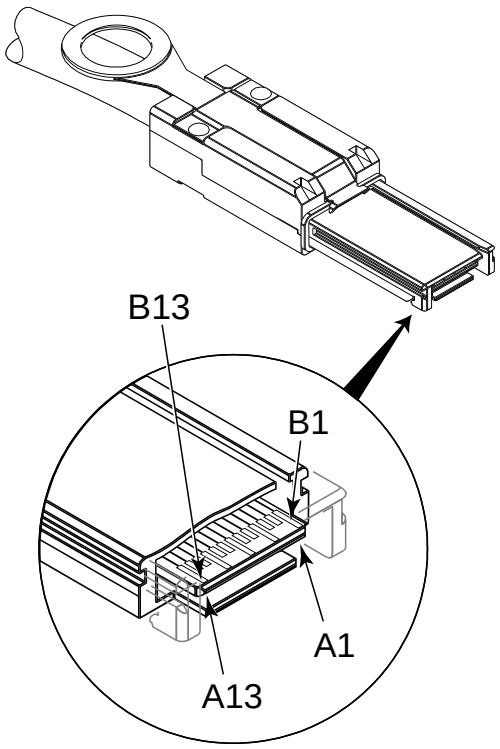


Figure 152 – Mini SATA External Multilane system, key slots 7 for m level signals

6.11.4.4 Mini SATA External Multilane pin assignments

The Mini SATA External Multilane connector pin assignments are shown in Figure 153.



Signal	Signal pin
SIGNAL GND	A1
Rx 0+	A2
Rx 0-	A3
SIGNAL GND	A4
Rx 1+	A5
Rx 1-	A6
SIGNAL GND	A7
Rx 2+	A8
Rx 2-	A9
SIGNAL GND	A10
Rx 3+	A11
Rx 3-	A12
SIGNAL GND	A13
SIGNAL GND	B1
Tx 0+	B2
Tx 0-	B3
SIGNAL GND	B4
Tx 1+	B5
Tx 1-	B6
SIGNAL GND	B7
Tx 2+	B8
Tx 2-	B9
SIGNAL GND	B10
Tx 3+	B11
Tx 3-	B12
SIGNAL GND	B13
CHASSIS GROUND	HOUSING

Figure 153 – Mini SATA External Multilane connector pin assignments

6.12 Cable and connector electrical specifications

6.12.1 Cable and connector electrical specifications overview

The purpose of this section is to specify the electrical characteristics of the cable and connector for all cabled usage models. The electrical characteristics defined herein describe relevant electrical characteristics required for high-speed signal transmission. An example test methodology is presented in Table 39 and Table 40 that may be used as a tool for characterizing cables, connectors, and PCB signal paths, (i.e., microstrip and stripline traces). Different test methodologies or equipment may be used as long as they provide equivalent results.

The cable and connector shall meet the electrical requirements listed below before and after all the tests given in Table 8 and Table 11 are performed.

6.12.2 Serial ATA cable electrical requirements

The electrical requirements for the internal single lane and Multilane Serial ATA cables and connectors for systems operating at Gen1i, Gen2i, or Gen3i levels are given in Table 36.

Table 36 – Internal cable / connector measurement parameter and requirements

Parameter	Requirement	Procedure
Mated Connector Differential Impedance	100 ohm ± 15 ohm	P1
Cable Absolute Differential Impedance	100 ohm ± 10 ohm	P2
Cable Pair Matching Impedance	± 5 ohm	P3
Common Mode Impedance	25 ohm to 40 ohm	P4
Maximum Insertion Loss of Cable (10 MHz to 4 500 MHz)	6 dB	P5
Maximum Crosstalk, single lane NEXT (10 MHz to 4 500 MHz)	26 dB loss	P6
Maximum Crosstalk, Multilane ML-CXT (10 MHz to 4 500 MHz)	30 dB loss	P7
Maximum Rise Time measured from 20 % threshold to 80 % threshold	85 picoseconds	P8
Maximum Inter-Symbol Interference	50 picoseconds	P9
Maximum Intra-Pair Skew	10 picoseconds	P10

NOTE 16 - The Internal Multilane and Mini SATA Internal Multilane maximum crosstalk is different than single lane. Since these cables are Multilane and have multi-aggressors, the additional requirement is to have crosstalk measured using the multilane crosstalk (ML-CXT) method.

The electrical requirements for the External Single Lane cable and connector for systems operating at Gen1m or Gen2m levels are defined in Table 37.

Table 37 – External Single Lane cable / connector measurement parameter and requirements

Parameter	Requirement	Procedure
Mated Connector Differential Impedance	100 ohm ± 15 ohm	P1
Cable Absolute Differential Impedance	100 ohm ± 10 ohm	P2
Cable Pair Matching Impedance	± 5 ohm	P3
Common Mode Impedance	25 ohm to 40 ohm	P4
Maximum Insertion Loss of Cable (10 MHz to 4 500 MHz)	8 dB	P5
Maximum Crosstalk NEXT (10 MHz to 4 500 MHz)	26 dB loss	P6
Maximum Rise Time measured from 20 % threshold to 80 % threshold	150 picoseconds	P8
Maximum Inter-Symbol Interference	50 picoseconds	P9
Maximum Intra-Pair Skew	20 picoseconds	P10

The electrical requirements for the External Multilane cable and connector for systems operating at Gen1m or Gen2m levels are defined in Table 38.

Table 38 – Limited External Multilane cable / connector measurement parameter and requirements

Parameter	Requirement	Procedure
Mated Connector Differential Impedance	100 ohm ± 15 ohm	P1
Cable Absolute Differential Impedance	100 ohm ± 10 ohm	P2
Cable Pair Matching Impedance	± 5 ohm	P3
Common Mode Impedance	25 ohm to 40 ohm	P4
Maximum Insertion Loss of Cable (10 MHz to 4 500 MHz)	8 dB	P5
Maximum Crosstalk ML-CXT (10 MHz to 4 500 MHz)	30 dB ML-CXT	P7
Maximum Rise Time measured from 20 % threshold to 80 % threshold	150 picoseconds	P8
Maximum Inter-Symbol Interference	50 picoseconds	P9
Maximum Intra-Pair Skew	20 picoseconds	P10

NOTE 17 - External Multilane cables for Gen1m or Gen2m signaling are limited to 2 m in length.

6.12.3 Cable/connector test methodology

6.12.3.1 Test equipment

The following list identifies the type and performance of suggested equipment to perform the characterization procedures outlined in Table 39 and Table 40:

a) High Bandwidth Sampling Oscilloscope;
b) TDR Module < 35 picoseconds measured from the 20 % threshold to 80 % threshold Edge Rate Step Response;
c) Vector Network Analyzer 4 port, may be 13.5 GHz, should be a bandwidth of 20 GHz;
d) High Performance Coaxial Cables = bandwidth 20 GHz; and
e) Low Jitter 3.0 Gbit/s Pattern Source with a minimum rise time of 136 picoseconds, measured from the 20 % threshold to 80 % threshold. The rise time should be as close to 136 picoseconds as is practical.

6.12.3.2 Test and measurement conditions

Unless otherwise specified, all tests and measurements shall be performed under the following conditions:

a) cable/connector mated;
b) temperature from 15 °C to 35 °C;
c) relative humidity from 20 % to 80 %; and
d) atmospheric pressure from 650 mmHg to 800 mmHg.

6.12.3.3 Test fixture considerations

Characterization of the cable/connector configuration requires an interface between the unit under test (UUT) and the test equipment and is commonly referred to as the test fixture. A primary objective in using a test fixture is to eliminate, as much as possible, the adverse signal integrity effects of the PCB.

The following guidelines should be followed to define the test fixture. Consider the following:

a) the test fixture should use differential microstrip traces (e.g., 100 ohm ± 5 ohm) over a ground plane (e.g., single ended 50 ohm ± 2.5 ohm);
b) open or shorted traces with the same length as the input signal traces shall be provided to enable the following:
    A) establish system input rise time;
    B) synchronize pulses; and
    C) establish reference plane;
c) traces for crosstalk measurements should diverge from each other; and
d) provisions for attenuation reference measurement should also be provided.

6.12.3.4 Test definition / methodology

There are a number of steps within the test procedures used in preparation of making a measurement and are referred to as common procedures. They consist of calibration, de-skewing, establishing a reference plane, and establishing a rise time reference trace.

Prior to performing any procedures or gathering data, ensure that the test equipment has been properly calibrated.

The methodology to complete each of the common procedures is outlined in Table 39.

Table 39 – Common interconnect measurement procedure methodologies

C1	**Minimizing Skew between V+ and V-, Diff Signals**
	The procedure is: 1) define Differential Channel Stimulus: a) channel 1 and channel 3 positive edge step response (V^+); and b) channel 2 and channel 4 negative edge step response (V^-); 2) differential response, identify the differential signal (V_{diff}) as a scope response, (e.g., use math function to obtain $V_{diff} = V^+ - V^-$ (CH1-CH2 or CH3-CH4)); and 3) minimize skew between the V+ (positive) and V- (negative) edges by adjusting either the V+ (CH1/3) or V- (CH2/4) edge forward or backward in time until both edges align to within 1 picosecond.
C2	**Establishing a reference plane at the connector**
	The procedure is: 1) follow calibration procedures outlined in the firmware of the oscilloscope; 2) select the define reference plane option within the scope firmware to establish a reference plane at the input of the test fixture; and 3) to establish a new reference plane, use precision 50 ohm loads or precision air lines that are terminated with 50 ohm loads for the test fixture.
C3	**Establishing the rise time reference trace**
	The procedure is: 1) configure the TDR modules to generate a differential step impulse response and identify the differential rising edge of the trace; 2) identify the high and low voltage values of the impulse response; 3) identify the 20 % and 80 % voltage levels and verify that the rise time of the step impulse is between 25 picoseconds and 35 picoseconds. There are two methods for adjusting the step impulse response to be within the desired range: a) the system rise time is to be set via equipment filtering techniques. The filter programmed equals $\sqrt{t^2_{r(observed)} - t^2_{r(stimulus)}}$; or b) capture the measurement data and perform a post processing step to filter the captured data to the desired rise time within a waveform viewer or TDR SW application; and 4) once the correct rise time has been established, verify the rise time using the reference traces on the PCB fixture.

The test methodologies and procedures outlined in Table 40 refer to the common procedures described in Table 39. The actual specification requirement values for each of these methodologies are defined in 6.12.2.

Table 40 – Interconnect test methodologies / procedures (part 1 of 5)

P1	**Mated Connector Differential Impedance**
	The procedure is: 1) calibrate the instrument and system using the measurement traces, then follow common procedures: 1) C1; 2) C2; and 3) C3; 2) the instrument rise time shall be set or the results filtered for a minimum of 55 picoseconds to a maximum of 70 picoseconds, measured from the 20 % threshold to 80 % threshold system risetime. The system risetime shall be set as close to 70 picoseconds, measured from the 20 % threshold to 80 % threshold as practical; and 3) measure and record the maximum and minimum values of the near end connector differential impedance.
P2	**Cable Absolute Differential Impedance**
	The procedure is: 1) calibrate the instrument and system using the measurement traces, then follow common procedures: 1) C1; 2) C2; and 3) C3; 2) the instrument rise time shall be set or the results filtered for a minimum of 55 picoseconds to a maximum of 70 picoseconds, measured from the 20 % threshold to 80 % threshold system risetime. The system risetime shall be set as close to 70 picoseconds, measured from the 20 % threshold to 80 % threshold as practical. 3) Measure and record maximum and minimum cable differential impedance values from the TDR trace in the first 500 picoseconds of cable response following any vestige of the connector response.
P3	**Cable Pair Matching**
	The procedure is: 1) calibrate the instrument and system using the measurement traces, then follow common procedures: 1) C1; 2) C2; and 3) C3; 2) The instrument rise time shall be set or the results filtered for a minimum of 55 picoseconds to a maximum of 70 picoseconds, measured from the 20 % threshold to 80 % threshold system risetime. The system risetime shall be set as close to 70 picoseconds, measured from the 20 % threshold to 80 % threshold as practical; 3) measure and record the single-ended cable impedance of each cable within a pair, (e.g., Z_{L1}, Z_{L2}); and 4) measure and record maximum and minimum cable impedance values from the TDR trace in the first 500 picoseconds of cable response following any vestige of the connector response, (e.g., $Z_{L1\text{-}max}$, $Z_{L1\text{-}min}$ and $Z_{L2\text{-}max}$, $Z_{L2\text{-}min}$). The desired parameter equals $Z_{max} = Z_{L1\text{-}max} – Z_{L2\text{-}max}$ and $Z_{min} = Z_{L1\text{-}min} – Z_{L2\text{-}min}$.

Table 40 – Interconnect test methodologies / procedures (part 2 of 5)

P4	**Common Mode Impedance**
	The procedure is: 1) calibrate the instrument and system using the measurement traces, then follow common procedures: 1) C1; 2) C2; and 3) C3; 2) the instrument rise time shall be set or the results filtered for a minimum of 55 picoseconds to a maximum of 70 picoseconds, measured from the 20 % threshold to 80 % threshold system risetime. The system risetime shall be set as close to 70 picoseconds, measured from the 20 % threshold to 80 % threshold as practical; 3) select the negative edge step response channel to be a positive edge step response such that both channels generate a positive edge step response; 4) measure the even mode impedance from the TDR trace of the first step generator in the first 500 picoseconds of cable response following any vestige of the connector response; 5) perform a math function on the waveform to divide the even mode impedance response by 2. The result is the Common Mode Impedance; and 6) make the same measurement and math calculation of the second step generator. The Common Mode Impedance for each step generator shall meet the requirement.
P5	**Insertion Loss**
	The procedure is: 1) calibrate the instrument and system using the measurement traces, then follow common procedures: 1) C1; and 2) C2; 2) measure and store the insertion loss (IL) of the fixturing using the IL reference traces provided on the board over a frequency range of 10 MHz to 4 500 MHz, (e.g., ILfixture); 3) measure and record the IL of the sample that includes fixturing IL, over a frequency range of 10 MHz to 4 500 MHz, (e.g., ILsystem); and 4) the insertion loss of the sample is calculated by ILsample = ILsystem – ILfixture.

Table 40 – Interconnect test methodologies / procedures (part 3 of 5)

P6	Differential to Differential Crosstalk NEXT
	The procedure is: 1) calibrate the instrument and system using the measurement traces, then follow common procedures; 1) C1; 2) C2; and 3) C3; 2) terminate the far ends of the reference trace with characteristic impedance loads of 50 ohm; 3) measure and record the system and fixturing crosstalk, it is defined as the noise floor, (e.g., V_{noise}) ; 4) Terminate the far ends of the device and listen lines with characteristic impedance loads of 50 ohm; 5) Connect the source to the device pair and the receiver to the near-end of the listen pair; 6) Measure the NEXT over a frequency range of 10 MHz to 4 500 MHz, (e.g., V_{NEXT}); and 7) Verify that the sample crosstalk is out of the noise floor, (e.g., $V_{NEXT} > V_{noise}$).

Table 40 – Interconnect test methodologies / procedures (part 4 of 5)

P7	**Multilane (Multi Disturber) Differential Crosstalk ML-CXT**

The procedure is:

1) calibrate the instrument and system using the measurement traces, then follow common procedures:
   1) C1;
   2) C2; and
   3) C3;
2) terminate the far ends of the reference trace with characteristic impedance loads of 50 ohm;
3) measure and record the system and fixturing crosstalk. It is defined as the noise floor, (e.g., V_{noise});
4) terminate the far ends of the device and listen lines with characteristic impedance loads of 50 ohm;
5) connect the source to the device pair and the receiver to the near-end of the listen pair;
6) measure the ML-CXT over a frequency range of 10 MHz to 4 500 MHz, (e.g., $V_{ML\text{-}CXT}$); and
7) verify that the sample crosstalk is out of the noise floor, (e.g., $V_{ML\text{-}CXT} > V_{noise}$) as in the following equation.

$$ML-CXT(f) = -20 \times \log\left(\sum_{A=1}^{7^1} 10^{-V_{ML-CXT}(f)_l/20}\right)$$

where:

ML-CXT(f) is the Multilane Cable assembly Crosstalk at frequency f observed on any given receive lane;

V_{ML-CXT}(f)A is the relative crosstalk at frequency f between the receiver/victim and any combination of aggressor A [a] that exhibits less than 40 dB of isolation;

f is the frequency ranging from 10 MHz to 4 500 MHz; and

A is the 1 to 7 (receiver/victim to aggressor [a] pair combinations).

[a] Aggressor is any lane identified as a Tx (input) shall be considered as a potential aggressor. This includes near end and far end Tx lane.

Note- ML-CXT summation calculations accounts for any aggressor [a] relative to a receiver/victim pair that exhibit less than 40 dB of isolation.

Table 40 – Interconnect test methodologies / procedures (part 5 of 5)

P8	**Differential Rise Time**
	The procedure is: 1) calibrate the instrument and system using the measurement traces, then follow common procedures: 1) C1; 2) C2; and 3) C3; 2) connect the TDR step impulse response generators to the near end of the signal path under test; and 3) record the output rise time at the far end of the signal path under test.
P9	**Inter-Symbol Interference** [a]
	The procedure is: 1) connect a differential pattern source at the input of the test fixture;. 2) the minimum rise time of the pattern source shall be 136 picoseconds, measured from the 20 % threshold to 80 % threshold, the rise and fall times should be as close to 136 picoseconds as is practical, to minimize the resulting DJ and produce the most accurate results; 3) Generate a LBP at 3.0 Gbit/s through the fixture. The Lone Bit Pattern emphasizes ISI; and 4) Using a JMD, evaluate the DJ introduced at the end of the cable.
P10	**Intra-Pair Skew**
	The procedure is: 1) calibrate the instrument and system using the measurement traces, then follow common procedures: 1) C1; 2) C2; and 3) C3; and 2) measure the propagation delay of each single ended signal within a pair at the mid point of the voltage swing, (e.g., t_{delay} = V_{mid+} - V_{mid-}) where $V_{mid} = \frac{V_{high} - V_{low}}{2}$.

[a] As incident (test system induced) DJ may not be de-convolved from the end results, it is critical one use a high quality (low jitter) fixture and stimulus system when performing this measurement.

6.13 Hardware Feature Control (optional)

6.13.1 Behavior

6.13.1.1 Behavior overview

Prior to processing a SET FEATURES Enable Hardware Feature Control (see 13.3.9) subcommand, the Hardware Feature Control pin(s) (see 4.1.1.66) operate using the default behavior in accordance with 6.13.1.2. Otherwise the Hardware Feature Control pin(s) operate using the extended behavior in accordance with 6.13.1.3.

6.13.1.2 Default behavior

There are two hardware control features listed as:

a) Disable Staggered Spinup (DSS); and
b) Device Activity Signal (DAS) (e.g., light emitting diode (LED)).

Due to various hardware issues, these features are mapped onto different physical pins depending on the connecter type as indicated in Table 41 and Table 125. Not all features are defined for all connecter types.

Table 41 – Default uses of DSS and DAS for various connectors

Standard Connector (3.5 inch and 2.5 inch)	1.8 inch Micro SATA Connector [a]	LIF-SATA Connector
Pin 11: a)DSS; or b)DAS.	Pin 7: a) DAS.	Pin 8: a)DSS; or b)DAS.
[a] DSS is not defined for 1.8 inch Micro SATA Connector.		

A Hardware Feature Control pin, (see Table 41) may be used by the device to provide the host with an activity indication and it may be used by the host to indicate whether staggered spinup should be used. To accomplish both of these goals, a Hardware Feature Control pin (see Table 41) acts as an input from the host to the device prior to PHYRDY for staggered spinup control and then acts as an output from the device to the host after PHYRDY for activity indication. The activity indication provided is primarily for use in backplane applications. See 13.15 for information on activity LED generation for desktop applications.

A host or device may optionally support activity indication, staggered spinup control, or both features. If neither feature is supported, then pin P11, P7, or P8 depending upon connector is a no connect at the device as specified in Table 5.

6.13.1.3 Extended behavior

The Hardware Feature Control pin(s) may be used by the device for one of the following:

a) default use of the Hardware Feature Control pin(s) (see 6.13.1.2);
b) Direct Head Unload (DHU) (see 6.13.2 , 13.10, and 13.19); or
c) vendor specific use (see 13.10).

6.13.2 Electrical requirements specification

See Table 42 for DHU electrical requirements.

Table 42 – Electrical requirements for DHU

Parameter	Min	Max	Description and Conditions
$V_{DHUactive}$	1.8 V	2.1 V	Host voltage presented to device to load the heads onto the ramp and keep them there. Value specified for all allowable I_{DInact} leakage currents.
$V_{DHUnegate}$	-0.1 V	225 mV	Host voltage presented to device to clear the state of the DHU. The timing to load the heads onto the media is vender specific. Value specified for all allowable I_{DInact} leakage currents.

6.13.3 Device Activity Signal (DAS)

6.13.3.1 Electrical definition

The signal the device provides for activity indication is a low-voltage low-current driver intended for efficient integration into current and future IC manufacturing processes. The signal is not suitable for directly driving an LED and shall first be buffered using a circuit external to the device before driving an LED.

The activity signal is based on an open-collector or open-drain active-low driver. The device shall tolerate the activity signal being shorted to ground. The device shall tolerate a no connect floating activity signal.

Table 43 and Table 44 define the electrical parameters and requirements for the activity signal for both the device and the host. Figure 154 is an example of an activity signal implementation for illustrative purposes. No direct support for wired-OR signals from multiple devices is accommodated. Host implementations that produce a single activity signal by combining multiple device inputs should buffer the signals prior to combining them.

> The host should not rely on a particular resistor pull-up value on the device side. The device should not rely on a particular host resistor value.

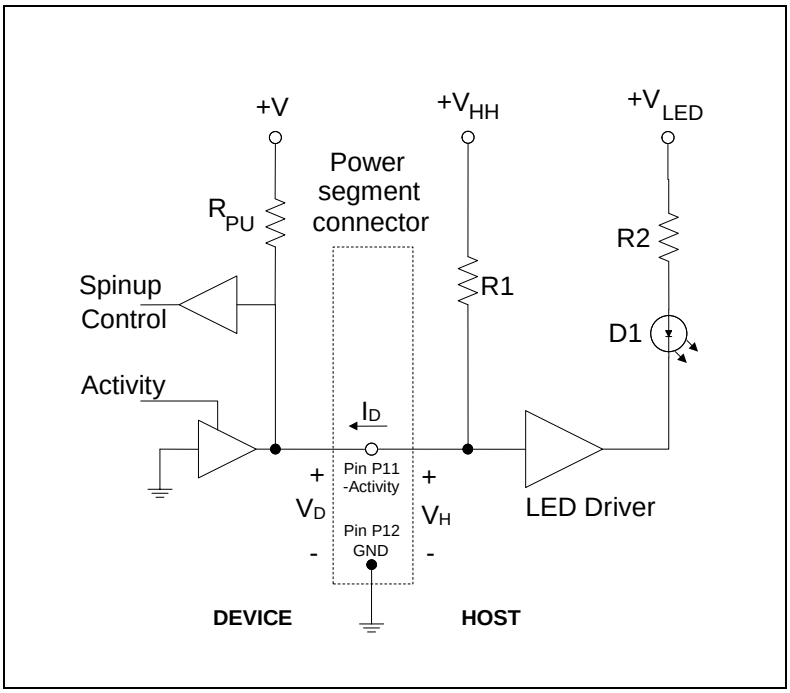


Figure 154 – Example Device Activity Signal (DAS) electrical block diagram

All voltage references in Table 43 and Table 44 are to ground pin P12 on the device connector. All voltages and currents in Table 43 and Table 44 are measured at pin P11 on the device connector.

Table 43 – Power segment pin P11 Device Activity Signal (DAS) electrical parameters

Parameter	Min	Max	Description and Conditions
V_{DIn}	-0.5 V	2.1 V	Tolerated input voltage.
V_{DAct}	0 mV	225 mV	Device output voltage when driving low under the condition I_D less than or equal to 300 uA.
V_{DInact}	-0.1 V	3.3 V	Device output voltage when not driving low.
I_{DInact}	-10 uA	100 uA	Device leakage current when not driven.

Table 44 – Host activity signal electrical parameters

Parameter	Min	Max	Description and Conditions
V_{HIn}	-0.5 V	3.3 V	Tolerated input voltage.
V_{HH}		2.1 V	Host voltage presented to device when device not driving signal low (see 6.13.4 for staggered spinup control).
V_{HL}	-0.1 V		Minimum allowable host voltage that may be presented to the device.
I_{HAct}		300 uA	Host current delivered to device when device driving signal low. Value specified at V_{DAct} voltage of 0 V.

6.13.3.2 LED driver circuit (informative)

The LED driver circuit provided by the host to drive an activity LED is vendor specific. Figure 155 illustrates two conceptual driver circuits that satisfy the electrical requirements and provide a signal suitable for driving an activity LED. Variations in the driver circuits may be employed to drive the LED when active or to drive the LED when the device is inactive through the use of an inverting or non-inverting buffering arrangement.

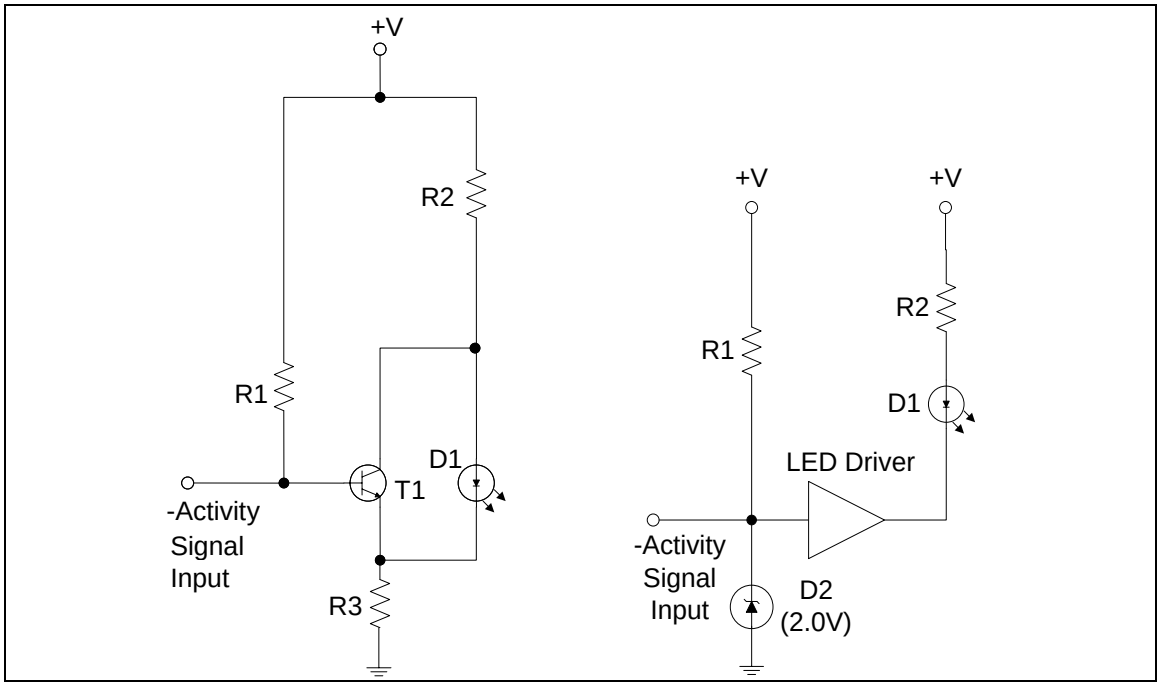


Figure 155 – Example host LED driver circuits

6.13.3.3 Functional definition

Table 45 defines the two activity signal states and the corresponding conditions.

Table 45 – Activity signal functional states

State	Condition [b]
Signal asserted (driven low)	Command(s) outstanding [a]
Signal negated (high impedance)	All other conditions

[a] Devices may omit asserting the activity signal for commands that do not access the media and have an expected service time too short to allow visual perception of the signal. Command(s) outstanding does not include the software reset, power-on reset, or COMRESET command protocols. As a consequence, pin P11 shall not be driven low by the device prior to return of the reset signature for the reset command protocols. This is behaviorally different than the parallel ATA DASP- signal.

[b] If the device supports the Out Of Band Management interface (see 13.22) (i.e., the OUT OF BAND MANAGEMENT INTERFACE SUPPORTED bit is set to one (see 13.7.11.2.29)) and the Out Of Band Management interface is enabled (i.e., the REPORTING ENABLED bit set is set to one (see 13.7.10.1)), then the device may toggle the activity signal as defined in SFF-8609 while transferring attribute information using the Out Of Band Management interface even if no commands are outstanding.

6.13.4 Disable Staggered Spinup (DSS) control

6.13.4.1 Electrical and functional definition

The staggered spinup feature is defined in 13.11. Hosts or devices may optionally provide support to DSS through pin P11 of the Standard Connector or pin P8 of the LIF-SATA Connector (see Table 125) of the power segment connector. The DSS control is an active asserted low host signal.

Before the device spins up its media, devices that support DSS control shall detect whether pin P11 of the Standard Connector or pin P8 of the LIF-SATA Connector is asserted low by the host. If pin P11 of the Standard Connector or pin P8 of the LIF-SATA Connector is asserted low the device shall disable staggered spinup and immediately initiate media spinup. If pin P11 of the Standard Connector or pin P8 of the LIF-SATA Connector is not asserted low in the host, devices that support DSS through pin P11 of the Standard Connector or pin P8 of the LIF-SATA Connector shall enable staggered spinup. Table 46 defines the electrical signal requirements for the device detection of DSS.

Table 46 – Host staggered spinup control electrical requirements

Parameter	Min	Max	Description and Conditions
V_{HEnb}	1.8 V	V_{HHmax}	Host voltage presented to device to not disable staggered spinup in devices that support staggered spinup control. Value specified for all allowable I_{DInact} leakage currents.
V_{HDis}	-0.1 V	225 mV	Host voltage presented to device to disable staggered spinup in devices that support staggered spinup control. Value specified for all allowable I_{DInact} leakage currents.

The staggered spinup control indication provided by a host or storage subsystem shall not be changed before PHYRDY is asserted. If the signal is pulled low by the host during the DSS detection period, the signal shall remain low at least until after PHYRDY is asserted.

If the staggered spinup feature is supported, the device shall:

a) sample the DSS condition after the time DC power is applied and before PHYRDY is asserted; and
b) not enable DAS before PHYRDY is asserted.

For devices utilizing either the Standard Connector or the 1.8 inch Micro SATA Connector, if the DHU feature is supported, the device shall disable DAS.

6.13.4.2 Disable Staggered Spinup (DSS) circuit (informative)

The host circuit for signaling DSS by pulling pin P11 low is vendor specific. Figure 156 illustrates a conceptual host circuit that is able to satisfy the electrical requirements for signaling DSS. It is permissible for the host to statically short pin P11 to ground or for the host to actively drive the signal low.

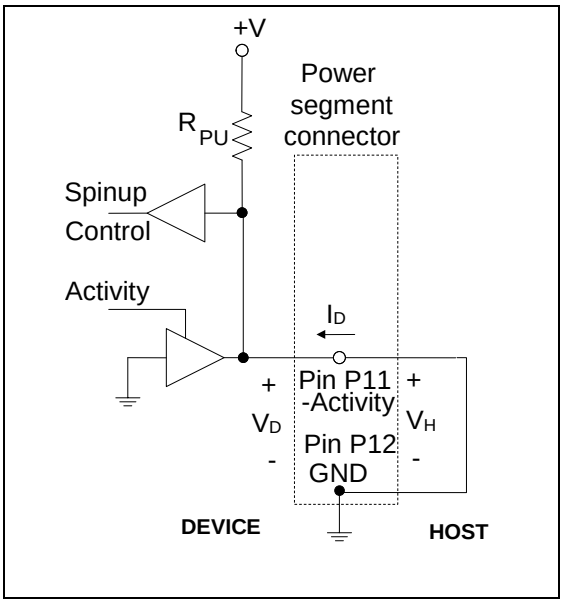


Figure 156 – Example circuit for Disable Staggered Spinup (DSS)

6.13.5 Micro SATA connector P7 definition (optional)

6.13.5.1 Micro SATA connector P7 definition overview

P7 of the Micro SATA connector may be used by the device to provide the host with an activity indication. The activity indication provided by P7 is primarily for use in backplane applications. See 13.15 for information on activity LED generation for desktop applications.

6.13.5.2 Device Activity Signal (DAS) (optional)

6.13.5.2.1 Electrical definition

The signal the device provides for activity indication is a low-voltage low-current driver intended for efficient integration into current and future IC manufacturing processes. The signal is not suitable

for directly driving an LED and shall first be buffered using a circuit external to the device before driving an LED.

The activity signal is based on an open-collector or open-drain active-low driver. The device shall tolerate the activity signal being shorted to ground. The device shall tolerate a no connect floating activity signal.

Table 47 and Table 48 define the electrical parameters and requirements for the activity signal for both the device and the host. Figure 157 is an example of an activity signal implementation for illustrative purposes. No direct support for wire-ORing signals from multiple devices is accommodated. Host implementations that produce a single activity signal by combining multiple device inputs should buffer the signals prior to combining them.

NOTE 18 - Note that the host is unable to rely on a particular resistor pull-up value on the device side, nor is it possible for the device rely on particular host resistor values.

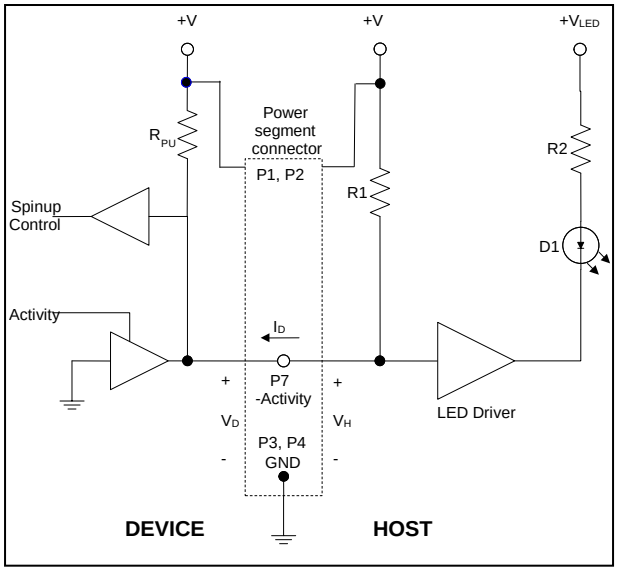


Figure 157 – Example Device Activity Signal (DAS) electrical block diagram

All voltage references in Table 47 and Table 48 are to ground pin 4 on the device connector. All voltages and currents in Table 47 and Table 48 are measured at P7 on the device connector.

Table 47 – Micro SATA connector P7 Device Activity Signal (DAS) electrical parameters

Parameter	Min	Max	Description and Conditions
V_{DIn}	-0.5 V	2.1 V	Tolerated input voltage.
V_{DAct}	0 mV	225 mV	Device output voltage when driving low under the condition I_D less than or equal to 300 uA.
V_{DInact}	-0.1 V	3.3 V	Device output voltage when not driving low.
I_{DInact}	-10 uA	100 uA	Device leakage current when not driven.

Table 48 – Host activity signal electrical parameters

Parameter	Min	Max	Description and Conditions
V_{HIn}	-0.5 V	3.3 V	Tolerated input voltage.
V_{HH}	-	2.1 V	Host voltage presented to device when device not driving signal low.
V_{HL}	-0.1 V	-	Minimum allowable host voltage that may be presented to the device.
I_{HAct}	-	300 uA	Host current delivered to device when device driving signal low. Value specified at V_{DAct} voltage of 0 V.

6.13.5.2.2 LED driver circuit (informative)

The LED driver circuit provided by the host to drive an activity LED is vendor specific. Figure 158 illustrates two conceptual driver circuits that is able to satisfy the electrical requirements and provide signal suitable for driving an activity LED. Variations in the driver circuits are able to be employed to drive the LED if active or to drive the LED if the device is inactive through the use of an inverting or non-inverting buffering arrangement.

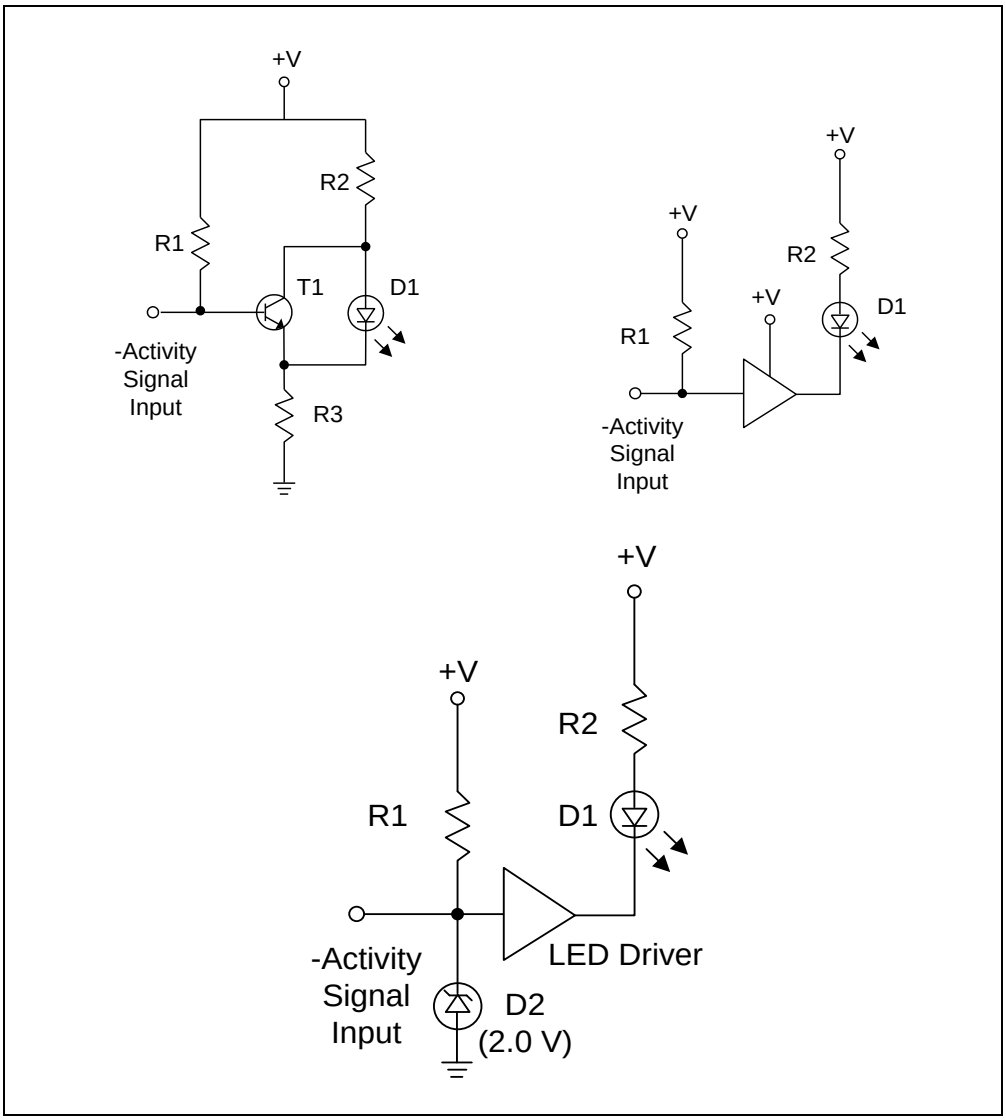


Figure 158 – Example host LED driver circuits

6.13.5.2.3 Functional definition

Table 49 defines the two activity signal states and the corresponding conditions.

Table 49 – Activity signal functional states

State	Condition
Signal asserted (driven low)	Command(s) outstanding [a]
Signal negated (high impedance)	All other conditions

[a] Devices may omit asserting the activity signal for commands that do not access the media and have an expected service time too short to allow visual perception of the signal. Command(s) outstanding does not include the software reset, power-on reset, or COMRESET command protocols. As a consequence, P7 shall not be driven low by the device prior to return of the reset signature for the reset command protocols. This is behaviorally different than the parallel ATA DASP- signal.

6.14 Precharge and device presence detection

6.14.1 Precharge and device presence detection overview

For a storage subsystem, hot plug capability is required as well as the ability to seamlessly handle presence detection in those cases where the storage subsystem may remove power to individual receptacles.

6.14.2 Device requirements

In order to accommodate hot-insertion with the use of the precharge feature as well as a means for presence detection, Serial ATA devices shall bus together all power delivery pins for each supply voltage.

6.14.3 Receptacle precharge (informative)

The Serial ATA device connector has been specifically designed to accommodate a robust hot plug capability. One feature of the device connector is the ability for receptacles to limit the instantaneous inrush current through the use of a precharge scheme. This scheme relies on one power delivery contact for each voltage being longer than the remaining contacts in order to allow power to be delivered through this longer contact through a current limiting device. Figure 159 illustrates one hot plug power delivery scheme that utilizes the precharge connector feature.

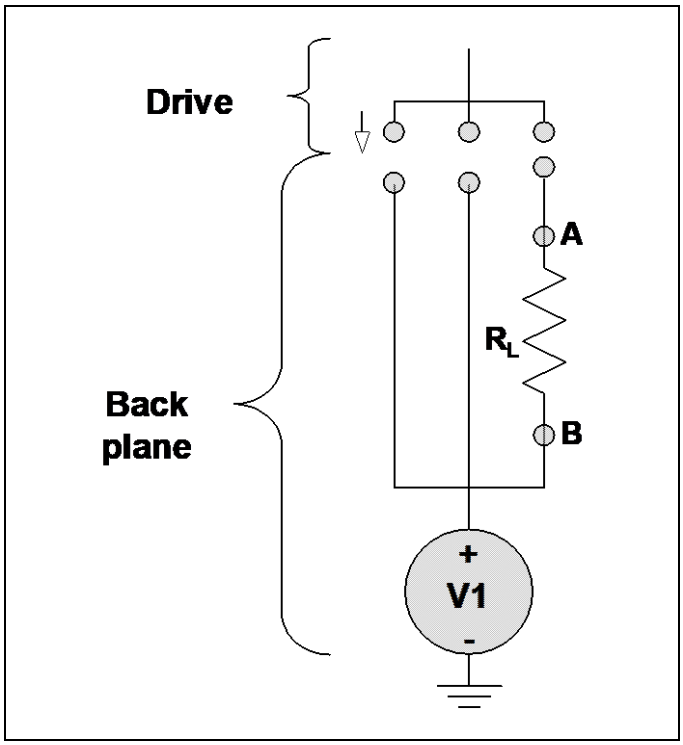


Figure 159 – Typical precharge configuration

All burden for limiting the inrush current for a newly inserted device is borne by the receptacle/backplane. The exact current limiting resistor size appropriate for a particular backplane solution depends on the details of the implementation. A few of the variables to be considered in sizing the current limiting resistor include:

a) device insertion velocity;
b) effective capacitance of the inserted device; and
c) contact current carrying capacity.

A survey of these variables by the group indicated that for one particular application, the maximum insertion velocity yielded a contact precharge time of approximately 3 ms. A poll of several disk drive vendors indicated a typical effective capacitance for disk drive devices of approximately 20 uF. For illustrative purposes, these values are presumed in an example scenario for estimating the precharge resistor value. The amount of time required to charge the effective capacitance to 90 % of full charge is roughly 2.2 × R × C. Thus:

$$T = 2.2 \times R \times C_{EQ}$$
$$R = T / (2.2 \times C_{EQ})$$

For the example charging time of 3 ms and an effective capacitance of 20 uF, the resultant precharge resistor value is approximately:

$$R = \frac{3 \text{ ms}}{2.2 \text{ x } 20 \text{ uF}} = 68 \text{ ohm}$$

Because the Serial ATA power conductors support currents up to 1.5 A, the computed resistor size may be substantially reduced without adverse consequence in order to reduce the sensitivity to the device's actual effective capacitance. For the 12 V supply rail, the resistor may be as small as

8 ohm and still not exceed the current carrying capacity of the precharge contact. Depending on the details of the actual enclosure subsystem design, typical precharge resistor values for the illustrative example scenario may therefore be in the range of 10 ohm to 20 ohm.

6.14.4 Presence detection (informative)

Presence detection relies on the device signaling the host using the Out Of Band (OOB) sequence to indicate its presence after a hot insertion. This approach presumes the device is inserted into a hot receptacle and also presumes the device inserted is not malfunctioning. In a storage subsystem, these assumptions may not be appropriate since such storage solutions may have the ability to unpower individual device receptacles in order to make device insertion/removal safer. Thus, a means for determining device presence in a receptacle that does not have power applied and without the device having to function is desired.

One possible device presence detection mechanism utilizes the precharge circuit according to 6.14.3. The basic approach is to determine presence of a device by measuring the impedance between points A and B in the diagram. Because devices bus together their respective power delivery contacts, the impedance between points A and B in the diagram is R_L with no device present and is effectively zero with a device inserted in the receptacle.

Figure 160 illustrates one possible circuit for handling device presence detect with the receptacle either powered or unpowered. The example circuit is subject to tolerance buildup of the selected components and the supply voltages, and are only presented as conceptual examples.

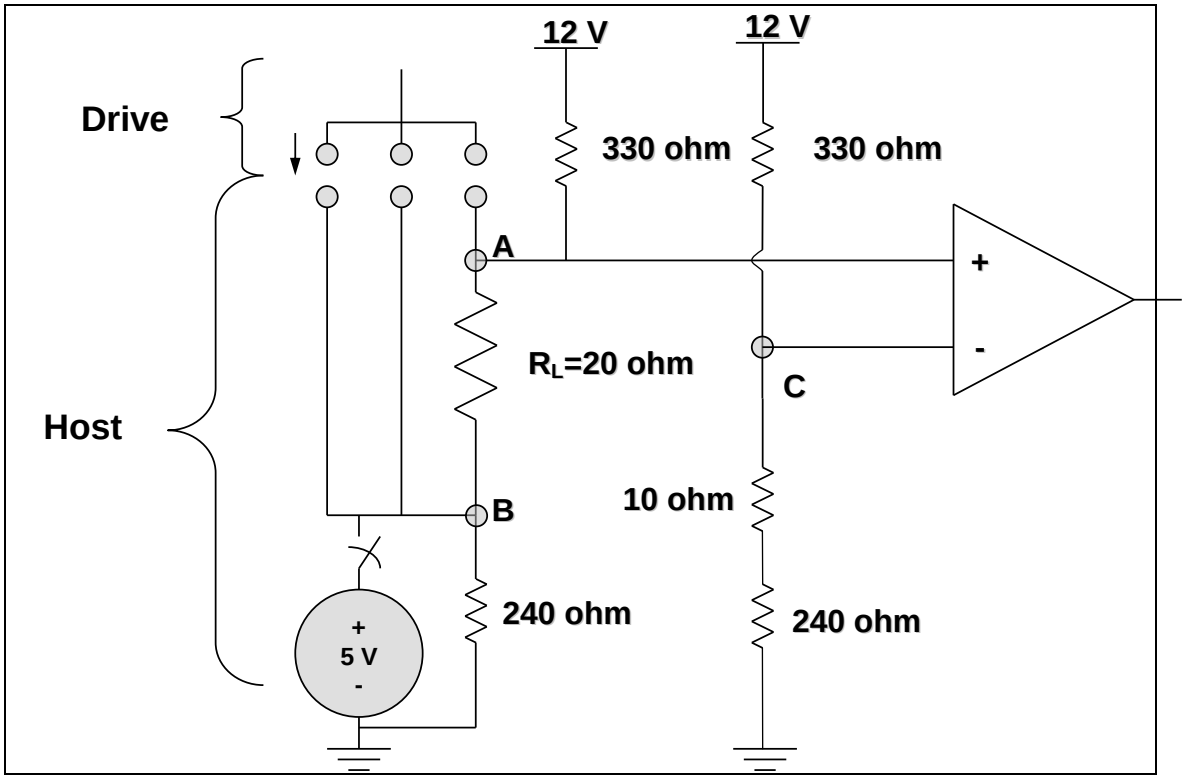


Figure 160 – Example presence detection implementation

Table 50 – Comparator voltages for alternate example presence detection circuit

	Receptacle Powered		Receptacle unpowered	
Device not present	V_A = 5.40 V	V_C = 5.17 V	V_A = 5.29 V	V_C = 5.17 V
Device present	V_A = 5.0 V	V_C = 5.17 V	[a] V_A = 5.05 V	V_C = 5.17 V
[a] If the inserted device provides a finite impedance to ground, then V_A should be lower than this value increasing the voltage differential further and increasing the margins.				

Table 50 provides an example of comparator voltages for a presence detection circuit.

7 Phy layer

7.1 Phy layer introduction

This section describes the physical layer of Serial ATA. The information that is provided is comprised of two types – informative and normative. Unless otherwise described, the information should be considered normative in nature and is included in this specification as a necessary requirement in order to properly allow a piece of equipment to attach to another piece of equipment. The normative information is deliberately structured to constrain and define areas only to the degree that is required for compatibility. The information that is provided and marked informative is provided only to help the reader better understand the normative sections and should be taken as examples only. Exact implementations may vary.

7.2 Descriptions of Phy electrical specifications

7.2.1 Terms overview

The following terms have been developed for the various Electrical Specifications:

a) Gen1i is generation 1 electrical specifications, these are the 1.5 Gbit/s electrical specifications for internal host to device applications;
b) Gen1m is generation 1 electrical specifications for Short Backplane and external cabling applications, these are the 1.5 Gbit/s electrical specifications aimed at short 1.5 Gbit/s internal backplane applications, External Desktop Applications using the external single lane cable, and System-to-System Data Center Applications using external Multilane cables up to two meters in length. These include only modified receiver Differential input specifications. All other electrical specifications relating to Gen1m compliance points are identical to Gen1i specifications. Gen1m is for the Short Backplane and External Desktop applications only and is not intended for any other system topology;
c) Gen1u is generation 1 electrical specifications defined at 1.5 Gbit/s for UHost applications. Because a Gen1i/Gen2i/Gen3i endpoint device is a direct connection to the mating connection of the UHost, the electrical specifications for Gen1u allow a channel loss up to the approximate equivalence of a 1 m data cable plus mated connector pair within the UHost;
d) Gen2i is generation 2 electrical specifications, these are 3.0 Gbit/s electrical specifications for internal host to device applications;
e) Gen2m is generation 2 electrical specifications for Short Backplane and External Desktop Applications, these are 3.0 Gbit/s electrical specifications aimed at short internal backplane applications, External Desktop Applications using the external single lane cable, and System-to-System Data Center Applications using external Multilane cables up to two meters in length. These include only modified receiver differential input specifications. All other electrical specifications relating to Gen2m compliance points are identical to Gen2i specifications. Gen2m is for the Short Backplane and External Desktop applications only and is not intended for any other system topology;
f) Gen2u is generation 2 electrical specifications defined at 3.0 Gbit/s for UHost applications. Because a Gen1i/Gen2i/Gen3i endpoint device is a direct connection to the mating connection of the UHost, the electrical specifications for Gen2u allow a channel loss up to the approximate equivalence of a 1 m data cable plus mated connector pair within the UHost;
g) Gen3i Generation 3 electrical specifications, these are 6.0 Gbit/s electrical specifications for internal host to device applications; and
h) Gen3u Generation 3 electrical specifications defined at 6.0 Gbit/s for UHost applications. Because a Gen1i/Gen2i/Gen3i endpoint device is a direct connection to the mating connection of the UHost, the electrical specifications for Gen3u allow a channel loss up to the approximate equivalence of the Gen3i CIC (see 7.4.8) plus mated connector pair within the UHost.

7.2.2 List of services

The list of services are:

a) transmit a 1.5 Gbit/s, 3.0 Gbit/s, or 6.0 Gbit/s differential non return to zero (NRZ) serial stream at specified voltage levels;
b) provide a 100 ohm matched termination (differential) at the transmitter;
c) serialize a 10 bit, 20 bit, 40 bit, or other width parallel input from the Link for transmission;
d) receive a 1.5 Gbit/s**,** 3.0 Gbit/s, or 6.0 Gbit/s differential NRZ serial stream with data rates of +350 ppm to -350 ppm with +0 ppm to -5 000 ppm (due to Spread Spectrum Clocking (SSC) profile) from the nominal data rate;
e) provide a 100 ohm matched termination (differential) at the receiver;
f) extract data (and, optionally, clock) from the serial stream;
g) de-serialize the serial stream;
h) detect the K28.5 comma character and provide a bit and Word aligned 10 bit, 20 bit, 40 bit, or other width parallel output;
i) provide specified OOB signaling detection and transmission;
j) use OOB signaling protocol for initializing the Serial ATA interface, and use this OOB sequence to process a pre-defined speed negotiation function;
k) perform proper power-on sequencing and speed negotiation;
l) provide device status to Link layer:
    A) device present;
    B) device absent; or
    C) device present but failed to negotiate communications;
m) optionally support power management modes;
n) optionally perform transmitter and receiver impedance calibration;
o) handle the input data rate frequency variation due to a spread spectrum transmitter clock; and
p) accommodate request to go into Far-End Retimed Loopback, and other BIST Activate FIS test modes of operation if commanded.

7.2.3 Low level electronics block diagrams (informative)

7.2.3.1 Block diagram overview

The following block diagrams (see Figure 161, Figure 162, and Figure 163) are provided as a reference for the following sections of this specification. Although informative in nature, the functions of the blocks described herein provide the basis that the normative specifications apply. The individual blocks provided are provided as an example of one possible implementation.

7.2.3.2 Physical plant block diagram

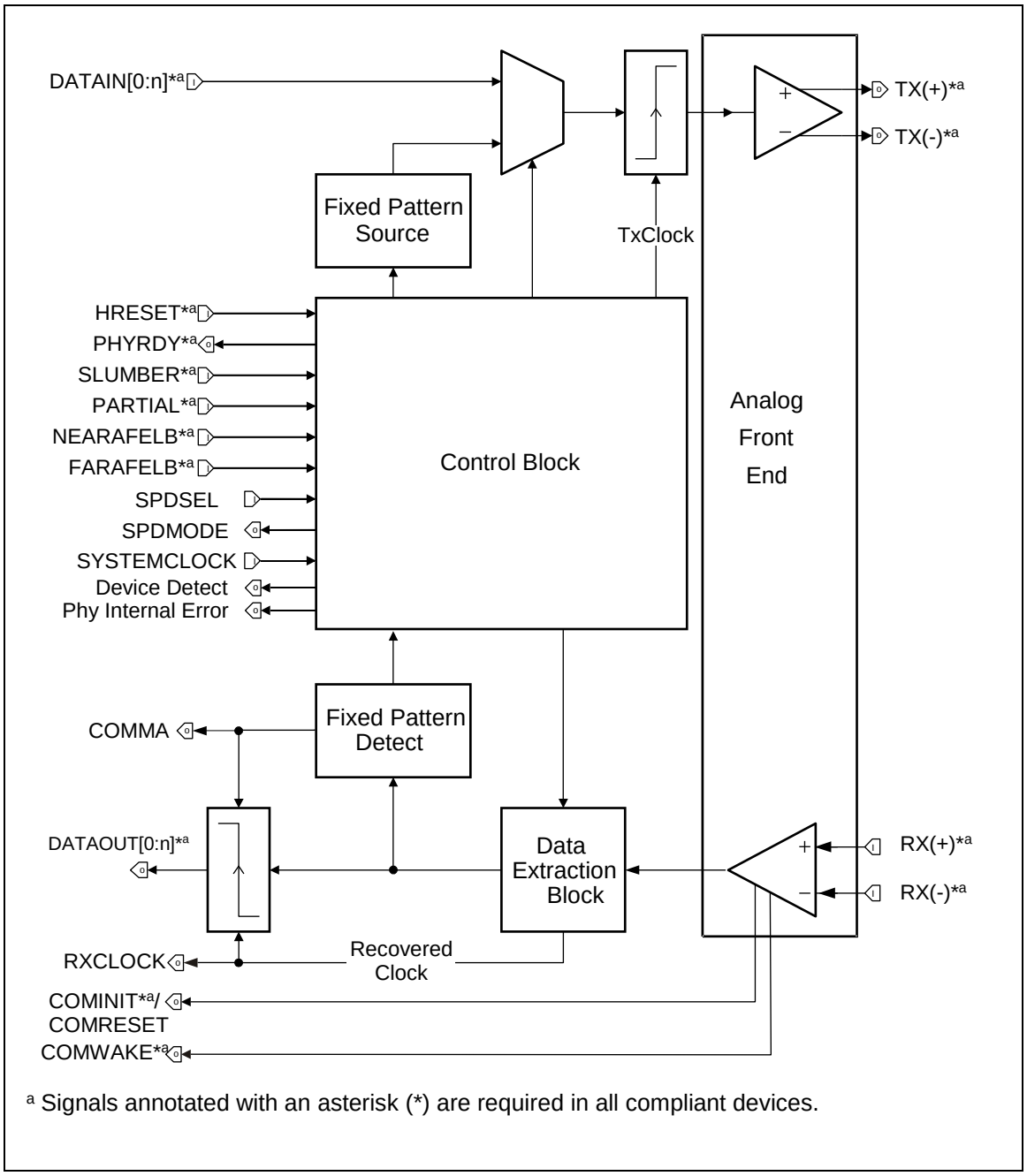


Figure 161 – Physical plant overall block diagram (informative)

7.2.3.3 Physical Plant Overall Block Diagram Description

Analog front end	This block is the basic interface to the transmission line. This block consists of the high-speed differential drivers and receivers as well as the OOB signaling circuitry.
Control block	This block is a collection of logic circuitry that controls the overall functionality of the Physical plant circuitry.
Fixed pattern source	This block provides the support circuitry that generates the patterns as needed to implement $ALIGN_P$ activity.
Fixed pattern detect	This block provides the support circuitry to allow proper processing of the $ALIGN_P$ primitives.
Data extraction block	This block provides the support circuitry to separate the clock and data from the high-speed input stream.
Tx clock	This signal is internal to the Physical plant and is a reference signal that regulates the frequency that the serial stream is sent via the high speed signal path.
Tx + / Tx -	These signals are the outbound high-speed differential signals that are connected to the Serial ATA cable.
Rx + / Rx -	These signals are the inbound high-speed differential signals that are connected to the Serial ATA cable.
DATAIN	Data sent from the Link layer to the Phy layer for serialization and transmission.
PHYRESET	This input signal causes the Phy to initialize to a known state and start generating the COMRESET OOB signal across the interface.
PHYRDY	Signal indicating Phy has successfully established communications. The Phy is maintaining synchronization with the incoming signal to its receiver and is transmitting a valid signal on its transmitter.
SLUMBER	Causes the Phy layer to transition to the Slumber power management state.
PARTIAL	Causes the Phy layer to transition to the Partial power management state.
NEARAFELB	Causes the Phy to loop back the serial data stream from its transmitter to its receiver.
FARAFELB	Causes the Phy to loop back the serial data stream from its receiver to its transmitter.
SPDSEL	Causes the control logic to automatically negotiate for a usable interface speed or sets a particular interface speed. The actual functionality of this input is vendor specific and varies from manufacturer to manufacturer.

SPDMODE — Output signal that reflects the current interface speed setting. The actual functionality of this signal is vendor specific and varies from manufacturer to manufacturer.

SYSTEMCLOCK — This input is the clock source for much of the control circuit and is the basis that the transmitting interface speed is established.

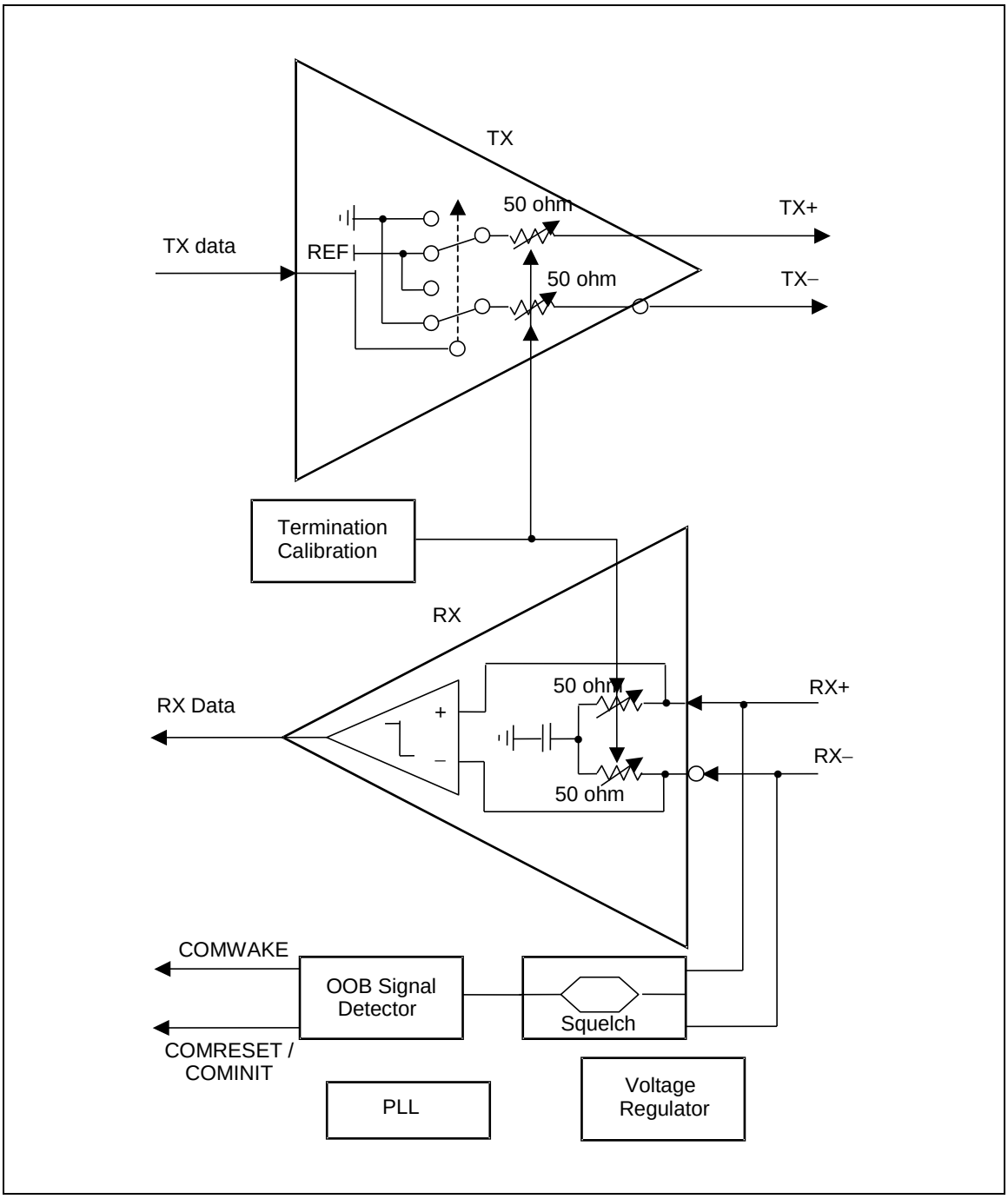
COMMA — This signal indicates that a K28.5 character was detected in the inbound high-speed data stream.

DATAOUT — Data received and de-serialized by the Phy and passed to the Link layer.

Rx CLOCK / Recovered clock
This signal is derived from the high speed input data signal and determines if parallel data has been properly formed at the DATAOUT pins and is available for transfer to outside circuitry.

COMRESET / COMINIT
Host: signal from the OOB detector that indicates the COMINIT OOB signal is being detected.
Device: signal from the OOB detector that indicates the COMRESET OOB signal is being detected.

COMWAKE — Signal from the OOB detector that indicates the COMWAKE OOB signal is being detected.

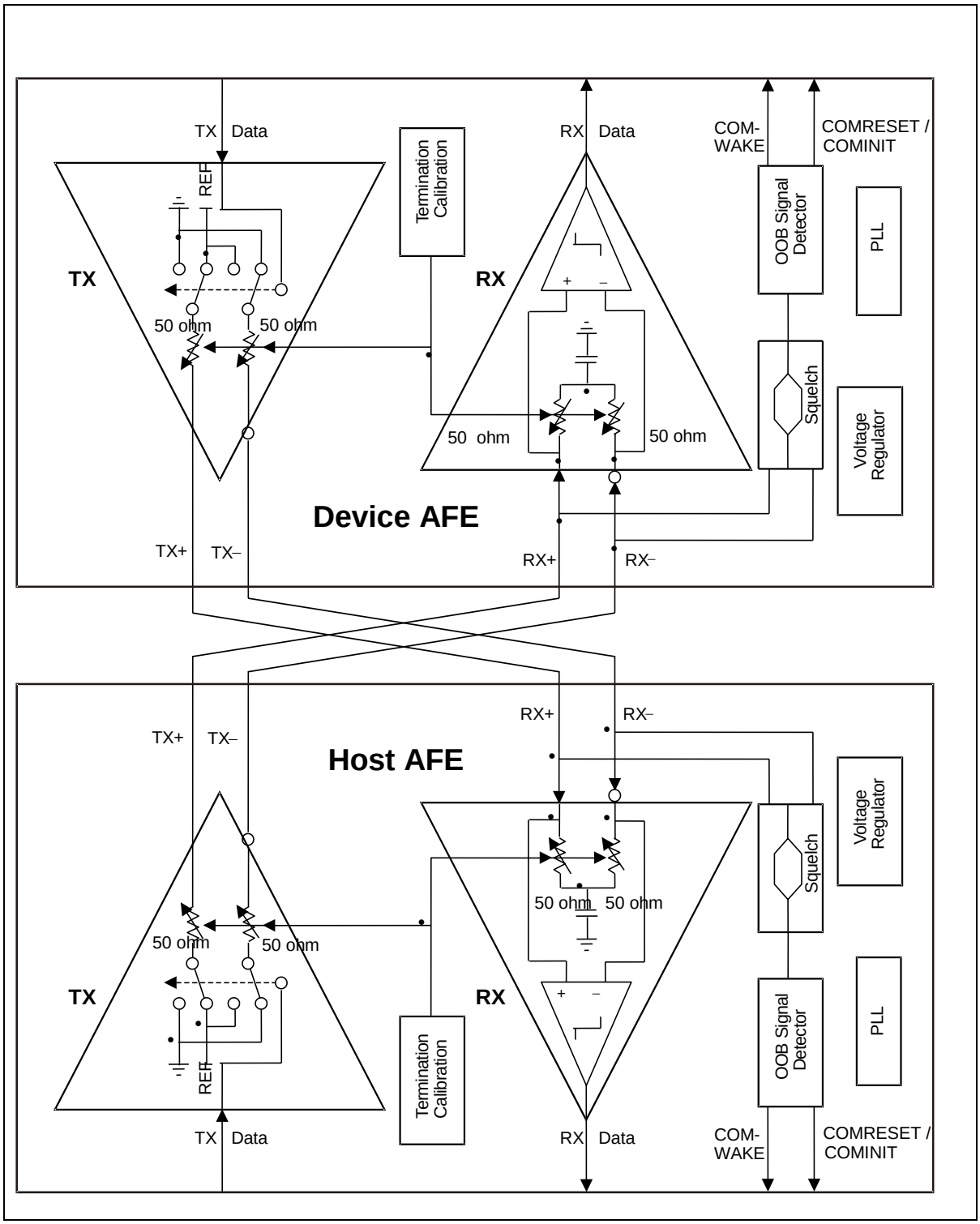


Note - External AC caps not shown

Figure 162 – Analog Front End (AFE) block diagram

7.2.3.4 Analog Front End (AFE) block diagram description

Tx	This block contains the basic high-speed driver electronics.
Rx	This block contains the basic high-speed receiver electronics.
Termination calibration	This block is used to establish the impedance of the Rx block in order to properly terminate the high-speed serial cable.
Squelch	This block establishes a limit so that detection of a common mode signal may be properly accomplished.
OOB signal detector	This block decodes OOB signal from the high-speed input signal path.
PLL	This block is used to synchronize an internal clocking reference so that the input high-speed data stream may be properly decoded.
Voltage Regulator	This block stabilizes the internal voltages used in the other blocks so that reliable operation may be achieved. This block may or may not be required for proper operation of the balance of the circuitry. The need for this block is implementation specific.
Tx+ / Tx-	This is the same signal as described in the previous section. Physical plant overall block diagram description.
Rx+ / Rx-	This is the same signal as described in the previous section. Physical plant overall block diagram description.
TxData	Serially encoded 8b/10b data attached to the high-speed serial differential line driver.
RxData	Serially encoded 8b/10b data attached to the high-speed serial differential line receiver.
COMWAKE	This is the same signal as described in the previous section. Physical plant overall block diagram description.
COMRESET / COMINIT	This is the same signal as described in the previous section. Physical plant overall block diagram description.



Note: - External AC coupling Capacitors not shown

Figure 163 – Analog Front End (AFE) cabling

7.2.4 Compliance testing

This specification provides electrical specifications that if met by hosts, devices, and interconnects, satisfy the link performance specifications if combined into a system. This section provides an overview of how to determine whether a host, device, or interconnect is compliant to the specifications of this specification.

Each electrical specification requires a specific measurement, test setup and data patterns. This section ties all of these requirements together to aid the reader in understanding what is needed for compliance testing.

Table 52, Table 53, Table 54, Table 55, Table 56, Table 57, Table 58, and Table 59 detail the electrical requirements for SATA compliance. Each requirement as defined in 7.4.3. Jitter as defined in 7.5. Measurement methods for each specification with details as given in 7.6. See 7.6.33.3 discuss Interface States relating to OOB and power management. See 6.12 describe the interconnect requirements.

The Phy layer is divided into a transmitter, interconnect, and a receiver (see Figure 164).

The SATA link is a full duplex point to point link as continuous data activity exists on each direction. For purposes of compliance testing of hosts and devices, the full duplex link is broken into two simplex links, one for the host transmitting to the device and the other for the device transmitting to the host. Each link is tested for compliance separately.

Each transmitter to receiver Link contains the following elements:

a) transmitter (IC/PCB/SATA Connector);
b) interconnect (Connector/Cable or PCB/Connector); and
c) receiver (SATA Connector / PCB / IC).

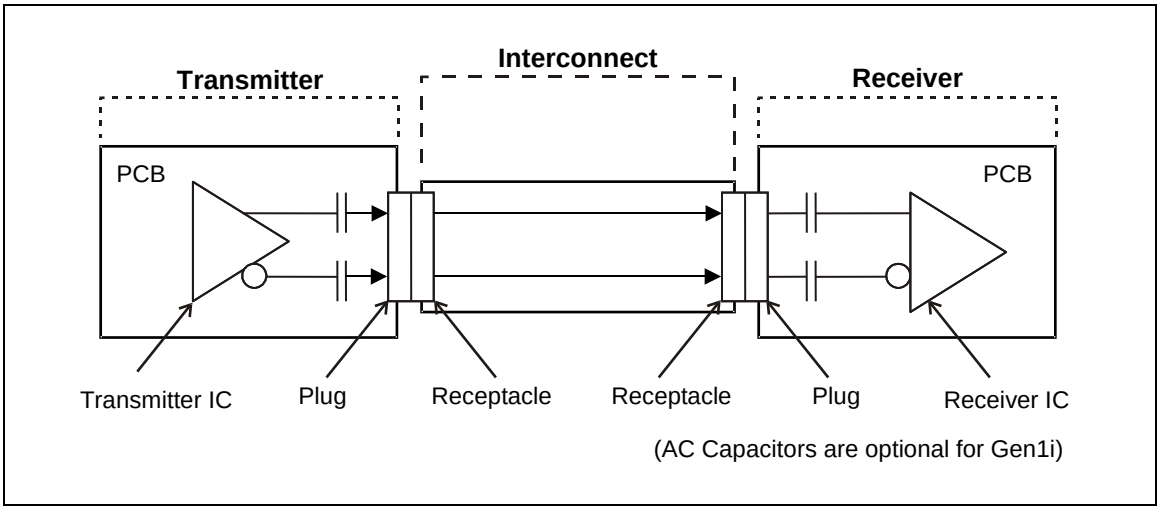


Figure 164 – The simplex link

In testing the compliance of SATA components that make up a system there are five Compliance Areas to be measured:

a) “Transmitted Signal”– examine the transmitted signal quality at the compliance point for the host/device into a Laboratory Load. Electrical specifications include amplitude, rise/fall time, frequency, jitter, etc. The Electrical specifications apply to the signal output from the Transmitter-Under-Test at the mated connector when driving a Laboratory Load. Unless a particular measurement requirement states otherwise, no attempt has been

made to specify the signal while attached to a cable, backplane, or directly into another device. Actual signals “In-System” may vary;

b) “Transmitter” examine all specified characteristics of the Transmitter from the compliance point. This includes specifications for differential and common-mode impedance. The “Transmitter” includes the IC that incorporates the transmitter, the PCB, the SATA connector as well as any additional components between the IC and the SATA connector;
c) “Receiver” examine all specified characteristics of the Receiver from the compliance point. This includes specifications for differential and common-mode impedance. The “Receiver” includes the SATA connector, the PCB and the IC that incorporates the receiver as well as any additional components between the IC and the SATA connector;
d) “Receiver Tolerance” The Receiver is presented with a worst-case Lab-Sourced Signal, and operating with its active transmitter, shall meet the specified Frame Error Rates. This requires carefully controlled signal sources in order to generate a worst-case signal; and
e) "Interconnect" examine all specified characteristics of the interconnect, using test equipment. The interconnect includes SATA connector pairs at each end. The testing requirements and procedures according to 6.12.

In order to determine compliance to this specification, measurements shall be performed separately (unless otherwise specified) with host, device, or interconnect being tested when connected to test equipment. Compliance tests are not done with a host, device, or interconnect connected together unless required by a particular measurement. Unless otherwise specified, all compliance measurements shall be taken through the mated connector pair.

NOTE 19 - The electrical specifications in the Receiver Tolerance Table do not describe the characteristics of the received signal; these describe the Lab-Sourced Signal calibrated into a Laboratory Load and subsequently applied to the Receiver. The Receiver Tolerance Table does not describe the characteristics of a signal from a Transmitter through an interconnect into a Laboratory Load. Received signals in a system are potentially worse due to the non-ideal impedance match of the transmitter and the receiver.

7.2.5 Link performance

The performance of a SATA system with host and device linked together with an interconnect is measured by the Frame Error Rate (FER), using a set of reference frames, defined by a specific set of ordered test patterns within the frame. An operating host-device duplex link that meets the Frame Error Rate specifications of Table 52 for both of its simplex links is deemed to fulfill Serial ATA performance levels. A host or device is commanded to generate the various test patterns through the use of the BIST Activate FIS or other vendor-specific commands to the device under test.

7.3 SATA Express system electrical requirements (obsolete)

7.3.1 SATA Express system electrical requirements overview

The SATA Express system electrical requirements cover:

a) AC coupling capacitance;
b) PCIe sidebands; and
c) power.

Most electrical characteristics that are defined as part of other standards, particularly PCIe, are referenced, rather than defined in SATA Express.

7.3.2 SATA Express AC coupling capacitance requirement

PCIe and SATA have different AC coupling capacitance requirements:

a) PCIe Gen1 and Gen2 min 75 nF, max 265 nF, Tx side only;
b) PCIe Gen3 min 176 nF, max 265 nF, Tx side only; or
c) SATA max 12 nF, both Tx and Rx sides (except for DC coupled Gen1i).

The following AC coupling capacitor is required for SATA Express:

a) there shall be no AC coupling capacitor placed on the SATA Express host, as illustrated in Figure 165; and
b) the PCIe device shall have AC coupling capacitors on both the Tx and Rx lanes, as shown in Figure 165. The capacitance value shall be 176 nF min to 265 nF max, as defined in the PCIe spec.

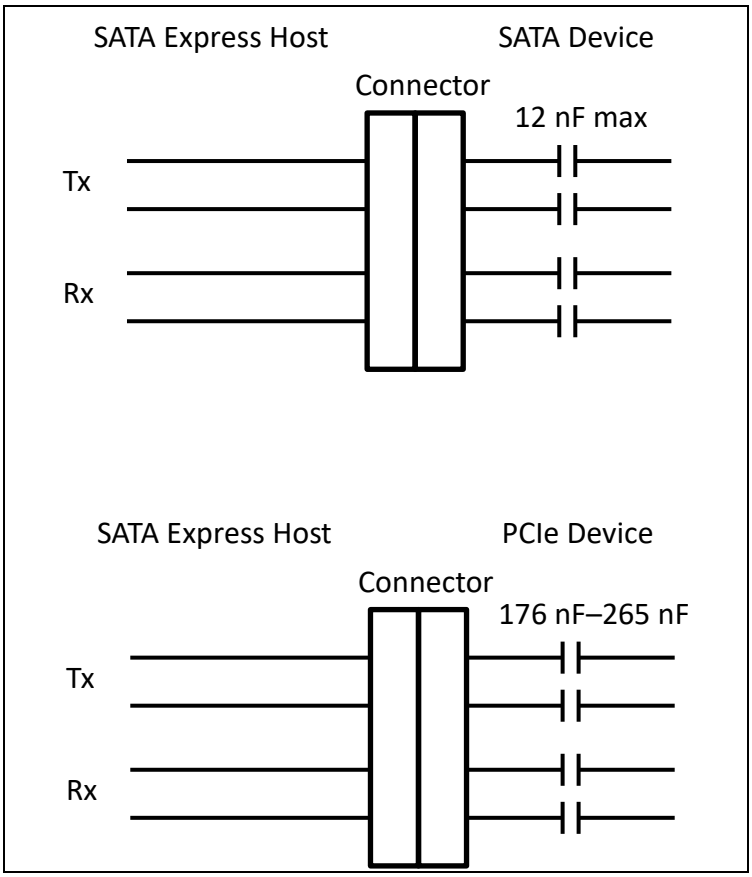


Figure 165 – AC coupling capacitor requirements

Such AC coupling capacitor requirements meet both the SATA and PCIe AC capacitor specifications. In the case of a SATA device working with a SATA Express host, a 12 nF max capacitor is provided by the SATA drive. A SATA Express host does not support SATA Gen1 DC coupled devices.

7.3.3 SATA Express interface detect

An interface detect (IFDet) is defined to detect if the device plugged in is SATA or PCIe. Figure 166 illustrates the interface detect mechanism for the cabled case.

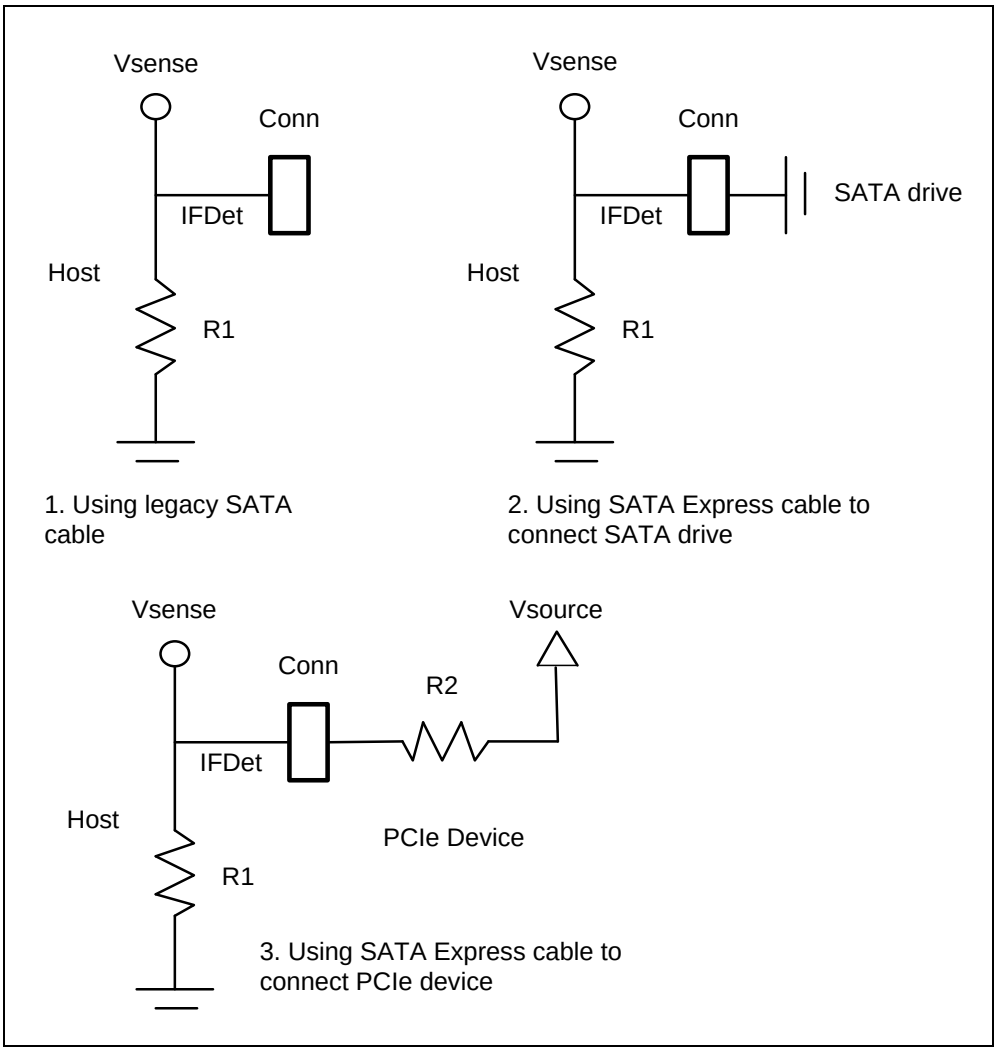


Figure 166 – SATA Express interface detect mechanism

To make this mechanism work, a pull-down resistor R1 is required on the host and a pull-up resistor R2 is needed on the PCIe device, connecting to voltage source Vsource. IFDet is always shorted to GND for SATA drives, but when using a legacy SATA cable the host IFDet is open requiring the host to apply the pull-down as illustrated in Figure 166.

Table 51 summarizes the logic states for Vsense for various cases.

Table 51 – Logic states of Vsense

Usage case	Vsense	Device type detected
SATA cable with SATA drive	Low	SATA
SATA Express cable with SATA drive	Low	SATA
SATA Express cable with PCIe drive	High	PCIe
SATA drive in SATA Express host receptacle connector	Low	SATA
PCIe drive in SATA Express host receptacle connector	High	PCIe

The followings are the requirements to support the interface detect:

a) the pull-down resistor R1 on the host may be on-die or discrete. The value of R1 shall be 20 kohm with a relative tolerance of ± 40 %;
b) the pull-up resistor R2 on the PCIe device of 5 kohm with a relative tolerance of ± 10 % shall be used, connecting to the 1.8 V rail with a relative tolerance of ± 5 %; and
c) the SATA Express cable shall have a wire to connect the IFDet pins at both ends.

NOTE 20 - To prevent DC drain when a PCIe device is connected to the SATA Express host, a switch may be implemented on host to shut down the DC drain path when detection is completed, as illustrated in Figure 167. The host/controller allows Tdelay amount of time for Vsense to stabilize. Then it latches in the detected voltage and opens the DC path to ground thus avoiding a constant DC power drain.

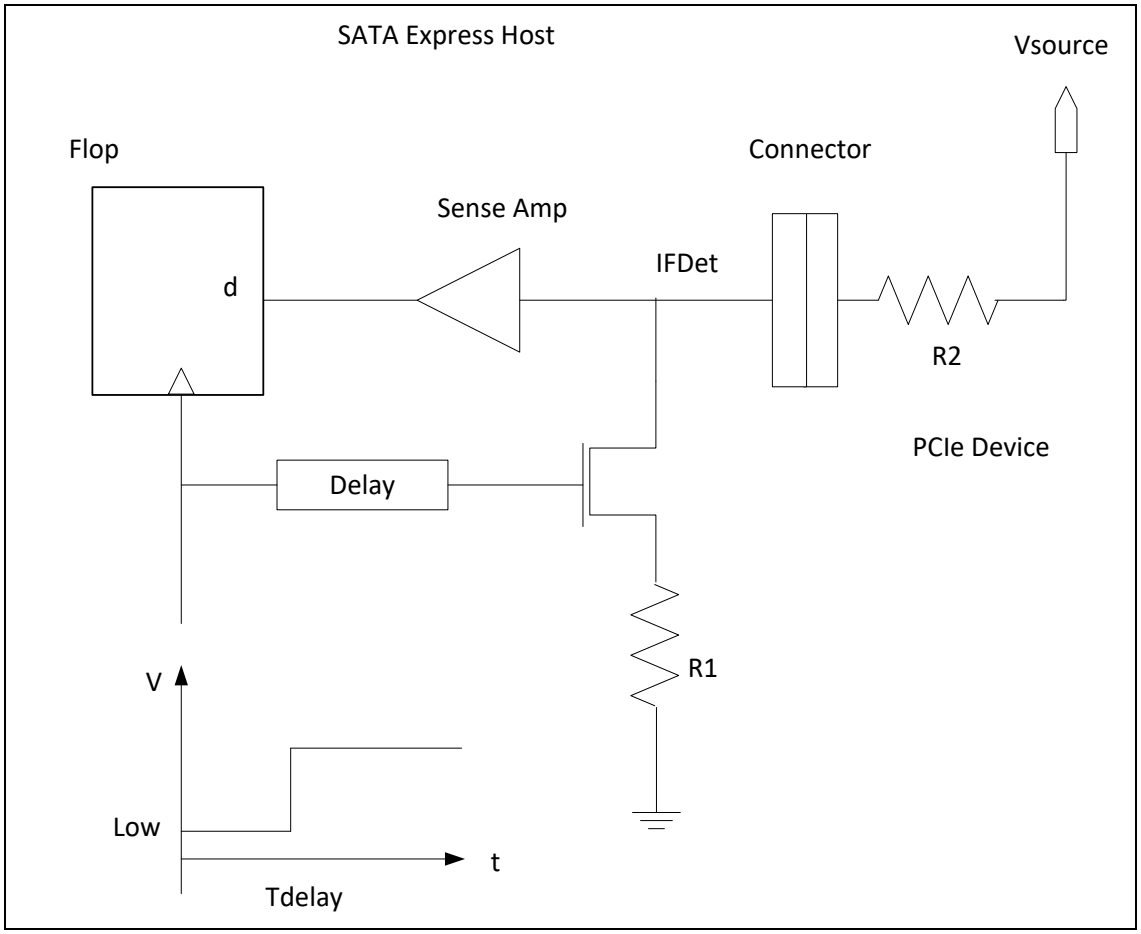


Figure 167 – Interface Detect, shutting down DC path after detection

7.3.4 PCIe sidebands

The PCIe reset (i.e., PERST#) is a required PCIe sideband. Its electrical requirements are given in the PCIe CEM Specification 3.0.

Clock request (i.e., CLKREQ#) is also required for a PCIe device to enter/exit the L1 PM substate. See PCIe Base Specification 3.0 for detail.

If the RefClk is not included in the SATA Express cabled interface, system manufacturers shall ensure that the PCIe receivers used for SATA Express cabling as discussed in this specification

support the SRIS capability, as specified in the PCIe Base Specification 3.0 ECN – Separate Refclk Independent SSC Architecture.

7.3.5 PCIe device power

PCIe devices get their power on 12 V or 5 V. 12 V shall be provided for the host systems that use the 3.5 inch drives. For systems that support only the 2.5 inch drives, 5 V shall be provided.

7.4 Electrical specifications

7.4.1 Electrical specifications overview

The goal of this specification is to provide a description of characteristics to ensure interoperability of SATA components (e.g., devices, hosts, and interconnects). Any combination of compliant components should provide the stated link performance. Secondly a means of validation to the requirements as defined in 7.6. Validation consists of performing tests on individual SATA components.

Serial ATA devices and hosts shall comply with the electrical specifications shown in Table 52, Table 53, Table 54, Table 56, Table 57, and Table 59**.** The transmitter consists of the driver integrated circuit (IC), printed circuit board, and mated connector pair. The receiver consists of the receiver IC, printed circuit board, and mated connector pair.

The Serial ATA UHost (see Table 1, Table 2 for applicability) shall comply with the electrical specifications shown in Table 52, Table 55**,** Table 56**,** and Table 58. Jitter transfer function (JTF) requirements in Table 54 apply to the Serial ATA UHost and are included through the measurement cross-references. The transmitter consists of the driver IC, mated connector pair and the channel between the driver IC and connector. The receiver consists of the receiver IC, mated connector pair and the channel between the receiver IC and connector.

Unless otherwise stated, all specifications include the mated connector pair.

7.4.2 Phy layer requirements tables

Table 52 – General specifications (part 1 of 2)

Parameters	Units	Limit	Electrical Specification								Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen1u	Gen2i	Gen2m	Gen2u	Gen3i	Gen3u		
Channel Speed	Gbit/s	Nom	1.5			3.0			6.0		7.4.3.1.2	-
Fbaud	GHz	Nom	1.5			3.0			6.0		-	-
FER, Frame Error Rate		Max	8.2e-8 at 95 % confidence level			8.2e-8 at 95 % confidence level			8.2e-8 at 95 % confidence level		7.4.3.1.3	7.6.3
$T_{UI,}$ Unit Interval	picoseconds	Min	666.433 3			333.216 7			166.608 3		7.4.3.1.4	7.6.16
		Nom	666.666 7			333.333 3			166.666 7			
		Max	670.233 3			335.116 7			167.558 3			
f_{tol}, Tx Frequency Long Term Accuracy	ppm of Fbaud	Min	-350			-350			-350		7.4.3.1.5	7.6.9
		Max	+350			+350			+350			
f_{SSC}, Spread-Spectrum Modulation Frequency	kHz	Min	30			30			30		7.4.3.1.6 0	7.6.16
		Max	33			33			33			
SSC_{tol}, Spread-Spectrum Modulation Deviation	ppm of Fbaud	Min	-5 350			-5 350			-5 350		7.4.3.1.7 0	7.6.16
		Max	+350			+350			+350			

Table 52 – General specifications (part 2 of 2)

Parameters	Units	Limit	Electrical Specification								Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen1u	Gen2i	Gen2m	Gen2u	Gen3i	Gen3u		
SSC_{tol}, Spread-Spectrum Modulation Rate	ppm/us	Max	1 250			1 250			1 250		7.4.3.1.70	7.6.16
$V_{cm,dc}$, DC Coupled Common Mode Voltage	mV	Min	200	(AC only)		(AC only)			(AC only)		7.4.3.1.8	7.6.16
		Nom	250	(AC only)		(AC only)			(AC only)			
		Max	450	(AC only)		(AC only)			(AC only)			
$V_{cm,ac\ coupled}$, AC Coupled Common Mode Voltage	mV	Min	0		-	-			-		7.4.3.1.9	7.6.31
		Max	2 000		-	-			-			
Z_{diff}, Nominal Differential Impedance	ohm	Nom	100			-			100		7.4.3.1.10	7.6.28
$C_{ac\ coupling}$ AC Coupling Capacitance	nF	Max	12			12			12		7.4.3.1.11	7.6.19
$t_{settle,cm}$, Common Mode Transient Settle Time	ns	Max	10			-			-		7.4.3.3.8	7.4.6.3
V_{trans}, Sequencing Transient Voltage	V	Min	-2.0			-2.0			-2.0		7.4.3.1.12	7.6.18
		Max	2.0			2.0			2.0			

Table 53 – Transmitter specifications (part 1 of 3)

Parameters	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
V_{trans}, Sequencing Transient Voltage LL	V	Min	-		-		-1.2	7.4.3.1.13	7.6.32
		Max	-		-		1.2		
Z_{diffTx}, Tx Pair Differential Impedance	ohm	Min	85		-		-	7.4.3.2.2	7.6.28
		Max	115		-		-		
$Z_{s\text{-}eTx}$, Tx Single-Ended Impedance	ohm	Min	40		-		-	7.4.3.2.3	7.6.29

Table 53 – Transmitter specifications (part 2 of 3)

Parameters	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
$RL_{DD11,Tx}$, Tx Differential Mode Return Loss (All Values Min)	dB	75 MHz to 150 MHz	14	14	-	-	-	7.4.3.2.4	7.6.15
		150 MHz to 300 MHz	8	8	14	14	-		
		300 MHz to 600 MHz	6	6	8	8	-		
		600 MHz to 1.2 GHz	6	6	6	6	-		
		1.2 GHz to 2.4 GHz	3	3	6	6	-		
		2.4 GHz to 3.0 GHz	1	-	3	3	-		
		3.0 GHz to 5.0 GHz	-	-	1	-	-		
$RL_{DD11,Tx}$, Tx Differential Mode Return Loss Start for slope	dB	Min at 300 MHz	-		-	-	14	7.4.3.2.7	7.6.15
Slope of Tx Differential Mode Return Loss	dB/dec	Nom	-		-	-	-13		
Tx Differential Mode Return Loss Max Frequency	GHz	Max	-		-	-	3		

Table 53 – Transmitter specifications (part 3 of 3)

Parameters	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
$RL_{CC11,Tx}$, Tx Common Mode Return Loss (all Values Min)	dB	150 MHz to 300 MHz	-		8	5	-	7.4.3.2.5	7.6.15
		300 MHz to 600 MHz	-		5	5	-		
		600 MHz to 1.2 GHz	-		2	2	-		
		1.2 GHz to 2.4 GHz	-		1	1	-		
		2.4 GHz to 3.0 GHz	-		1	1	-		
		3.0 GHz to 5.0 GHz	-		1	-	-		
$RL_{DC11,Tx}$, Tx Impedance Balance (all values Min)	dB	150 MHz to 300 MHz	-		30	30	30	7.4.3.2.6	7.6.15
		300 MHz to 600 MHz	-		20	20	30		
		600 MHz to 1.2 GHz	-		10	10	20		
		1.2 GHz to 2.4 GHz	-		10	10	10		
		2.4 GHz to 3.0 GHz	-		4	4	10		
		3.0 GHz to 5.0 GHz	-		4	-	4		
		5.0 GHz to6.5 GHz	-		-	-	4		

Table 54 – Transmitted signal requirements (part 1 of 5)

Parameter	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
$V_{diffTxdevice}$, Tx Differential Device Output Voltage	mVppd	Min	400		400		-	7.4.3.3.2	7.6.7
		Min	-		-		240		7.6.5 7.6.5.4
		Nom	500		-		-		7.6.7
		Max	600		700		-		
		Max	-		-		900		7.6.5
$V_{diffTxhost}$, Tx Differential Host Output Voltage	mVppd	Min	400		400		-	7.4.3.3.2	7.6.7
		Min	-		-		200		7.6.5 7.6.5.4
		Nom	500		-		-		7.6.7
		Max	600		700		-		
		Max	-		-		900		7.6.5
UIVminTx, Tx Minimum Voltage Measurement Interval	UI		0.45 to 0.55		0.45 to 0.55		-	7.4.3.3.3	7.6.7
			-		-		0.45 to 0.55		7.6.7.3
$V_{EmphasisDevice}$, Device TX Emphasis [a b c]	dB	Min	-		-		0.5	7.4.3.3.14	7.6.33
		Nom	-		-		2		
		Max	-		-		2.5		

[a] TX emphasis is measured at the mated Serial ATA connector at the device. Unless otherwise specified, no CIC is used for this measurement.
[b] The TX emphasis requirement does not apply to the Internal 4-lane cable mated to a backplane or devices using the Internal LIF-SATA connector, the SATA MicroSSD interface, or the Internal M.2 connector. Tx emphasis for these cases is vendor specific.
[c] The device shall pass at least one of the Device TX Emphasis test methods to achieve specification compliance for Device Tx Emphasis.
[d] This measurement includes the channel between the host IC and the device connector (e.g., motherboard, backplane, cable, and connectors). For internal 1 m cabled host to device applications (see section 5.3.2), the CIC is substituted for the actual system cable.

Table 54 – Transmitted signal requirements (part 2 of 5)

Parameter	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
$V_{EmphasisDevice}$, Peak-Mode Device TX Emphasis [a b c]	dB	Min	-		-		0.5	7.4.3.3.14	7.6.33
		Nom	-		-		-		
		Max	-		-		2.5		
$V_{EmphasisHost}$, Host TX Emphasis [a d]	dB	Min					-2	7.4.3.3.14	7.6.33
		Nom	0	-	0	-	0		
		Max					1.5		
t20-80Tx, Tx Rise/Fall Time	ps (UI)	Min 20 % to 80 %	50 (0.075)		50 (0.15)		33 (0.20)	7.4.3.3.4	7.6.6
		Max 20 % to 80 %	273 (0.41)		136 (0.41)		80 (0.48)		
t_{skewTx}, Tx Differential Skew	ps	Max	20		20		20	7.4.3.3.5	7.6.17
$V_{cm,acTx}$, Tx AC Common Mode Voltage	mVpp	Max	-		50		-	7.4.3.3.6	7.6.22
$V_{cm,acTx}$, Tx AC Common Mode Voltage	mVpp	Max	-		-		120	7.4.3.3.7	7.6.23
$D_{VdiffOOB}$, OOB Differential Delta	mV	Max	-		25		25	7.4.3.3.9	7.6.25

[a] TX emphasis is measured at the mated Serial ATA connector at the device. Unless otherwise specified, no CIC is used for this measurement.
[b] The TX emphasis requirement does not apply to the Internal 4-lane cable mated to a backplane or devices using the Internal LIF-SATA connector, the SATA MicroSSD interface, or the Internal M.2 connector. Tx emphasis for these cases is vendor specific.
[c] The device shall pass at least one of the Device TX Emphasis test methods to achieve specification compliance for Device Tx Emphasis.
[d] This measurement includes the channel between the host IC and the device connector (e.g., motherboard, backplane, cable, and connectors). For internal 1 m cabled host to device applications (see section 5.3.2), the CIC is substituted for the actual system cable.

Table 54 – Transmitted signal requirements (part 3 of 5)

Parameter	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
D_{VcmOOB}, OOB Common Mode Delta	mV	Max	-		50		50	7.4.3.3.10	7.6.24
Amp_{bal}, Tx Amplitude Imbalance	%	Max	-		30		30	7.4.3.3.12	7.6.20
TJ at Connector, Clk-Data, f_{BAUD}/500 JTF Defined	UI	Max	0.37		0.37		-	7.4.3.3.13 7.5	7.6.10 7.6.11
DJ at Connector, Clk-Data, f_{BAUD}/500 JTF Defined	UI	Max	0.19		0.19		-	7.4.3.3.13 7.5	7.6.10 7.6.11
Jitter Transfer Function Bandwidth (D24.3, high pass -3 dB)	MHz	Min	1.1		1.1		-	7.5.3	7.6.10
		Nom	2.1		2.1		-		
		Max	3.1		3.1		-		
Jitter Transfer Function Peaking	dB	Min	0		0		-	7.5.3	7.6.10
		Nom	0		0		-		
		Max	3.5		3.5		-		
Jitter Transfer Function Low Frequency Attenuation	dB	Min	69		69		-	7.5.3	7.6.10
		Nom	72		72		-		
		Max	75		75		-		

Table 54 – Transmitted signal requirements (part 4 of 5)

Parameter	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
Jitter Transfer Function Low Frequence Attenuation Measurement Frequency	kHz	Min	29.3		29.3		-	7.5.3	7.6.10
		Nom	30.0		30.0		-		
		Max	30.3		30.3		-		
Jitter Transfer Function Bandwidth (D24.3, high pass -3 dB) (Gen3)	MHz	Min	-		-		2.2	7.5.3.4	7.6.10
		Nom	-		-		4.2		
		Max	-		-		6.2		
Jitter Transfer Function Peaking (Gen3)	dB	Min	-		-		0	7.5.3.4	7.6.10
		Nom	-		-		0		
		Max	-		-		3.5		
Jitter Transfer Function Low Frequency Attenuation (Gen3)	dB	Min	-		-		35.2	7.5.3.4	7.6.10
		Nom	-		-		38.2		
		Max	-		-		41.2		

Table 54 – Transmitted signal requirements (part 5 of 5)

Parameter	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
Jitter Transfer Function Low Frequency Attenuation Measurement Frequency (Gen3)	kHz	Min	-		-		415.8	7.5.3.4	7.6.10
		Nom	-		-		420		
		Max	-		-		424.2		
TJ (10^{-12}) before and after CIC, Clk-Data JTF Defined (Gen3)	UI	Max	-		-		0.52	7.4.3.3.13 7.5	7.6.10 7.6.12
TJ (10^{-6}) before and after CIC, Clk-Data JTF Defined (Gen3)	UI	Max	-		-		0.46	7.4.3.3.13 7.5	7.6.10 7.6.12

Table 55 – UHost transmitted signal requirements (part 1 of 2)

Parameter [a]	Units	Limit	Electrical Specification			Detail Cross-Ref Section [b c]	Measurement Cross-Ref Section [b c]
			Gen1u	Gen2u	Gen3u [c]		
VdiffTx, Tx Differential Output Voltage	mVppd	Min	325	275	-	7.4.3.3.2	7.6.7
		Min	-	-	200		
		Nom	400	-	-		
		Max	600	750	-		
		Max	-	-	900		
UIVminTx, Tx Minimum Voltage Measurement Interval	UI		0.45 to 0.55	0.45 to 0.55	0.45 to 0.55	7.4.3.3.3	7.6.7
$V_{EmphasisHost}$, Host TX Emphasis	dB	Min	-	-	-2	7.4.3.3.14	7.6.33
		Nom	0	0	0		
		Max	-	-	1.5		
T20-80Tx, Tx Rise/Fall Time	ps (UI)	Min 20 % to 80 %	100 (0.15)	67 (0.20)	33 (0.20)	7.4.3.3.4	7.6.6
tskewTx, Tx Differential Skew	ps	Max	-	50	30	7.4.3.3.5	7.6.17
Vcm,acTx, Tx AC Common Mode Voltage	mVpp	Max	100	100	120	7.4.3.3.6 7.4.3.3.7	7.6.22 7.6.23
V_{trans}, Sequencing Transient Voltage LL	V	Min	-	-	-1.2	7.4.3.1.12	7.6.18
		Max	-	-	1.2		
Z_{diffTx}, Tx Pair Differential Impedance	ohm	Min	85	-	-	7.4.3.2.2	7.6.28
		Max	115	-	-		

[a] The UHost Tx requirements are defined at the target device attachment point and the UHost channel loss is not separable from the host transmitter.
[b] Referenced detail and measurement sections may indicate a different connection type in the figures than what applies to the specific UUT. Many figures are based on the Internal 1 m Cabled Host to Device usage model. Specific connection type may vary, depending on the type UHost under test.
[c] Gen3u measurements are made only with the lab-load and are not made using the Gen3i CIC.

Table 55 – UHost transmitted signal requirements (part 2 of 2)

Parameter [a]	Units	Limit	Electrical Specification			Detail Cross-Ref Section [b c]	Measurement Cross-Ref Section [b c]
			Gen1u	Gen2u	Gen3u [c]		
$Z_{s\text{-}eTx}$, Tx Single-Ended Impedance	ohm	Min	40	-	-	7.4.3.2.3	7.6.29
Amp_{bal}, Tx Amplitude Imbalance	%	Max	-	30	30	7.4.3.3.12	7.6.20
DVdiffOOB, OOB Differential Delta	mV	Max	-	25	25	7.4.3.3.9	7.6.25
DVcmOOB, OOB Common Mode Delta	mV	Max	-	50	50	7.4.3.3.10	7.6.24
TJ at Connector, Clk-Data, fBAUD/500 JTF Defined	UI	Max	0.52	0.52	-	7.4.3.3.13 7.5	7.6.10 7.6.11
DJ at Connector, Clk-Data, fBAUD/500 JTF Defined	UI	Max	0.34	0.34	-		
TJ (10^{-12}) without CIC, Clk-Data JTF Defined	UI	Max	-	-	0.52	7.4.3.3.13 7.5	7.6.10 7.6.12
TJ (10^{-6}) without CIC, Clk-Data JTF Defined	UI	Max	-	-	0.46		

[a] The UHost Tx requirements are defined at the target device attachment point and the UHost channel loss is not separable from the host transmitter.
[b] Referenced detail and measurement sections may indicate a different connection type in the figures than what applies to the specific UUT. Many figures are based on the Internal 1 m Cabled Host to Device usage model. Specific connection type may vary, depending on the type UHost under test.
[c] Gen3u measurements are made only with the lab-load and are not made using the Gen3i CIC.

Table 56 – Receiver specifications (part 1 of 3)

Parameter	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
Z_{diffRx}, Rx Pair Differential Impedance	ohm	Min	85		-		-	7.4.3.4.2	7.6.28
		Max	115		-		-		
Z_{s-eRx}, Rx Single-Ended Impedance	ohm	Min	40		-		-	7.4.3.4.3	7.6.29
$RL_{DD11,Rx}$, Rx Differential Mode Return Loss (all values Min)	dB	75 MHz to 150 MHz	18		18		-	7.4.3.4.4	7.6.15
		150 MHz to 300 MHz	14		14		18		
		300 MHz to 600 MHz	10		10		14		
		600 MHz to 1.2 GHz	8		8		10		
		1.2 GHz to 2.4 GHz	3		3		8		
		2.4 GHz to 3.0 GHz	1		-		3		
		3.0 GHz to 5.0 GHz	-		-		1		
$RL_{DD11,Rx}$, Rx Differential Mode Return Loss	dB	Min at 300 MHz	-		-		-18	7.4.3.2.7	7.6.15
Slope of Rx Differential Mode Return Loss	dB/dec	Nom	-		-		13		
Rx Differential Mode Return Loss Max Frequency	GHz	Max	-		-		6.0		

Table 56 – Receiver specifications (part 2 of 3)

Parameter	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
$RL_{CC11,Rx}$, Rx Common Mode Return Loss (all values Min)	dB	150 MHz to 300 MHz	-		5		5	7.4.3.4.5	7.6.15
		300 MHz to 600 MHz	-		5		5		
		600 MHz to 1.2 GHz	-		2		2		
		1.2 GHz to 2.4 GHz	-		1		1		
		2.4 GHz to 3.0 GHz	-		1		1		
		3.0 GHz to 5.0 GHz	-		1		-		
$RL_{DD11,Rx}$, Rx Differential Mode Return Loss	dB	Min at 300 MHz	-		-		-18	7.4.3.2.7	7.6.15
Slope of Rx Differential Mode Return Loss	dB/dec	Nom	-		-		13		
Rx Differential Mode Return Loss Max Frequency	GHz	Max	-		-		6.0		
$RL_{CC11,Rx}$, Rx Common Mode Return Loss (all values Min)	dB	150 MHz to 300 MHz	-		5	5	-	7.4.3.4.5	7.6.15
		300 MHz to 600 MHz	-		5	5	-		
		600 MHz to 1.2 GHz	-		2	2	-		
		1.2 GHz to 2.4 GHz	-		1	1	-		
		2.4 GHz to 3.0 GHz	-		1	1	-		
		3.0 GHz to 5.0 GHz	-		1	-	-		

Table 56 – Receiver specifications (part 3 of 3)

Parameter	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
$RL_{DC11,Rx}$, Rx Impedance Balance (all values Min)	dB	150 MHz to 300 MHz	-		30	30	30	7.4.3.4.6	7.6.15
		300 MHz to 600 MHz	-		30	30	30		
		600 MHz to 1.2 GHz	-		20	20	20		
		1.2 GHz to 2.4 GHz	-		10	10	10		
		2.4 GHz to 3.0 GHz	-		4	4	10		
		3.0 GHz to 5.0 GHz	-		4	-	4		
		5.0 GHz to 6.5 GHz	-		-	-	4		

Table 57 – Lab-Sourced Signal (for receiver tolerance testing) (part 1 of 2)

Parameter	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
$V_{diffRxdevice}$, Rx Differential Device Input Voltage	mVppd	Min	325	240	275	240	-	7.4.3.5.2	7.6.8
		Min	-		-		200		7.6.5 7.6.5.4 7.6.14
		Nom	400		-		-		7.6.8
		Max	600		750		-		
		Max	-		-		1 000		7.6.5 7.6.14
$V_{diffRxhost}$, Rx Differential Host Input Voltage	mVppd	Min	325	240	275	240	-	7.4.3.5.2	7.6.8
		Min	-		-		240		7.6.5 7.6.5.4 7.6.14
		Nom	400		-		-		7.6.8
		Max	600		750		-		
		Max	-		-		1 000		7.6.5 7.6.14
$t_{20-80Rx}$, Rx Rise/Fall Time	ps (UI)	Min 20 % to 80 %	100 (0.15)		67 (0.20)		-	7.4.3.5.3	7.6.6 7.6.13
		Min 20 % to 80 %	-		-		62 (0.37)		7.6.6 7.6.14
		Max 20 % to 80 %	273 (0.41)		136 (0.41)		-		7.6.6 7.6.13
		Max 20 % to 80 %	-		-		75 (0.45)		7.6.6 7.6.14

Table 57 – Lab-Sourced Signal (for receiver tolerance testing) (part 2 of 2)

Parameter	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
UI_{VminRx}, Rx Minimum Voltage Measurement Interval	UI		-		0.5		-	7.4.3.5.4	7.6.8
			-		-		0.45 to 0.55		7.6.5.3
t_{skewRx}, Rx Differential Skew	ps	Max	-		50		30	7.4.3.5.5	7.6.17
$V_{cm,acRx}$, Rx AC Common Mode Voltage	mVpp	Max	100		100		100	7.4.3.5.6	7.6.13
$f_{cm,acRx}$, AC Common Mode Frequency	MHz	Min	2		2		2	7.4.3.5.7	7.6.13
		Max	200		200		200		
TJ at Connector, Clk-Data, f_{BAUD}/500 JTF Defined	UI	Max	0.60		0.60		-	7.4.3.3.13 7.5	7.6.10 7.6.13
DJ at Connector, Clk-Data, f_{BAUD}/500 JTF Defined	UI	Max	0.42		0.42		-		
TJ after CIC, Clk-Data JTF Defined	UI	Max	-		-		0.6	7.4.3.5.9 7.5	7.6.10 7.6.14
RJ before CIC, MFTP Clk-Data JTF Defined	UI	Max	-		-		0.18 p-p (2.14 picoseconds 1 sigma)		

Table 58 – Lab-Sourced Signal (for UHost receiver tolerance testing) (part 1 of 2)

Parameter	Units	Limit	Electrical Specification			Detail Cross-Ref Section [a]	Measurement Cross-Ref Section [a]
			Gen1u	Gen2u	Gen3u		
V_{diffRx}, Rx Differential Input Voltage [b]	mVppd	Min	400	400	-	7.4.3.5.2	7.6.8
		Min	-	-	240		
		Max	600	700	-		
		Max	-	-	1 000		
$t_{20\text{-}80Rx}$, Rx Rise/Fall Time	ps (UI)	Min 20 % to 80 %	100 (0.15)	67 (0.20)	33 (0.20)	7.4.3.5.3	7.6.6
		Max 20 % to 80 %	273 (0.41)	136 (0.41)	68 (0.41)		
UI_{VminRx}, Rx Minimum Voltage Measurement Interval	UI		0.45 to 0.55	0.45 to 0.55	-	7.4.3.5.4	7.6.8
			-	-	0.45 to 0.55		7.6.5.4
t_{skewRx}, Rx Differential Skew	ps	Max	20	20	20	7.4.3.5.5	7.6.17
$V_{cm,acRx}$, Rx AC Common Mode Voltage	mVpp	Max	50	50	50	7.4.3.5.6	7.6.13 7.6.14
$f_{cm,acRx}$, AC Common Mode Frequency	MHz	Min	2	2	2	7.4.3.5.7	7.6.13 7.6.14
		Max	200	200	200		

[a] Referenced detail and measurement sections may indicate a different connection type in the figures than what applies to the specific UUT. Many figures are based on the Internal 1 m Cabled Host to Device usage model. Specific connection type may vary depending on the type UHost under test.

[b] Gen3u Lab-Sourced Signals for minimum Rx Differential Input Voltage and TJ are made adjusted using the Gen3i CIC into a lab-load. After setting these levels the Gen3i CIC is removed and the resulting signal is applied to the UHost receiver under test.

Table 58 – Lab-Sourced Signal (for UHost receiver tolerance testing) (part 2 of 2)

Parameter	Units	Limit	Electrical Specification			Detail Cross-Ref Section [a]	Measurement Cross-Ref Section [a]
			Gen1u	Gen2u	Gen3u		
Z_{diffRx}, Rx Pair Differential Impedance	ohm	Min	85	-	-	7.4.3.4.2	7.6.28
		Max	115	-	-		
$Z_{s\text{-}eRx}$, Rx Single-Ended Impedance	ohm	Min	40	-	-	7.4.3.4.3	7.6.29
TJ at Connector, Clk-Data, f_{BAUD}/500 JTF Defined	UI	Max	0.45	0.45	-	7.4.3.5.8 7.5	7.6.11 7.6.13
DJ at Connector, Clk-Data, f_{BAUD}/500 JTF Defned	UI	Max	0.27	0.27	-		
TJ Clk-Data JTF Defined [b]	UI	Max	-	-	0.60	7.4.3.5.9 7.5	7.6.12 7.6.14
RJ, MFTP Clk-Data JTF Defined	UI	Max	-	-	0.18 p-p (2.14 picoseconds 1 sigma)		

[a] Referenced detail and measurement sections may indicate a different connection type in the figures than what applies to the specific UUT. Many figures are based on the Internal 1 m Cabled Host to Device usage model. Specific connection type may vary depending on the type UHost under test.

[b] Gen3u Lab-Sourced Signals for minimum Rx Differential Input Voltage and TJ are made adjusted using the Gen3i CIC into a lab-load. After setting these levels the Gen3i CIC is removed and the resulting signal is applied to the UHost receiver under test.

Table 59 – OOB specifications

Parameter	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
V_{thresh}, OOB Signal Detection Threshold	mVppd	Min	50		75		75	7.4.3.6.3	7.6.26
		Nom	100		125		125		
		Max	200		200		200		
UI_{OOB}, UI During OOB Signaling	ps	Min	646.67		646.67		646.67	7.4.3.6.4	-
		Nom	666.67		666.67		666.67		
		Max	686.67		686.67		686.67		
COMINIT/ COMRESET and COMWAKE Transmit Burst Length	ns	Min	103.5					7.4.3.6.5 7.4.3.6.2	7.6.27
		Nom	106.7						
		Max	109.9						
COMINIT/ COMRESET Transmit Gap Length	ns	Min	310.4					7.4.3.6.6	7.6.27
		Nom	320.0						
		Max	329.6						
COMWAKE Transmit Gap Length	ns	Min	103.5					7.4.3.6.7	7.6.27
		Nom	106.7						
		Max	109.9						
COMWAKE Gap Detection Windows	ns	May detect	35 ≤ T < 175					7.4.3.6.8	7.6.27
		Shall detect	101.3 ≤ T ≤ 112						
		Shall not detect	T < 35 or T ≥175						
COMINIT/ COMRESET Gap Detection Windows	ns	May detect	175 ≤ T < 525					7.4.3.6.9	7.6.27
		Shall detect	304 ≤ T ≤ 336						
		Shall not detect	T < 175 or T ≥ 525						

7.4.3 Phy layer requirements details

7.4.3.1 General specifications details

7.4.3.1.1 General specifications scope

This section contains the details on Table 52, Table 53, and Table 55 entries.

7.4.3.1.2 Channel speed

A reference value showing the nominal rate of data through the channel.

7.4.3.1.3 Frame error rate (FER)

Frame error rate is the measure of link performance using all the intermediate circuit blocks in the chain from low-level Phy layer, Link layer, through Transport layer. Frame error rate is a system level test, not a compliance test. Error detection is at the frame level using the Cyclic Redundancy Check (CRC) error detection mechanism, and respective reporting to the higher layer levels.

7.4.3.1.4 Unit Interval (UI)

This is the operating data period (nominal value, architecture specific), excluding jitter. This value includes the long-term frequency accuracy and the Spread Spectrum Clock FM frequency deviation (rounded to 4 places).

This is the time interval value of each cycle of the Reference Clock.

7.4.3.1.5 Tx frequency long term accuracy

This specifies the allowed frequency variation from nominal. This does not include frequency variation due to jitter, Spread Spectrum Clocking, or phase noise of the clock source.

7.4.3.1.6 Spread spectrum modulation frequency

This is the modulation frequency of the SSC profile. See further details of Spread Spectrum as defined in 0.

7.4.3.1.7 Spread spectrum modulation deviation

This is the allowed frequency variation from the nominal Frequency baud (Fbaud) value in Table 52 if SSC is used. This deviation includes the long-term frequency variation of the transmitter clock source, and the SSC frequency modulation on the transmitter output. The frequency variation limits are measured using the SSC profile measurement see 7.6.16. See further details of Spread Spectrum as defined in 0.

7.4.3.1.8 DC coupled common mode voltage (Gen1i)

The Common mode DC level is defined as [(Tx+) + (Tx-)]/2 and [(Rx+) + (Rx-)]/2 measured at the mated connector.

This requirement only applies to Gen1i DC coupled designs (no blocking capacitors) that hold the common-mode DC level at the connector. The four possible common mode biasing configurations shown in Figure 168 demonstrate that only DC coupled designs need sustain the specified common-mode level to ensure interoperability. AC coupled designs may allow the DC level at the connector to float. The SATA interfaces defined as Gen1m, Gen1u, Gen2i, Gen2m, Gen2u, Gen3i, and Gen3u shall be AC coupled and this requirement does not apply to these.

A DC coupled receiver shall weakly hold the common-mode level of its inputs to the $V_{cm,dc}$ value specified in Table 52. A DC coupled transmitter shall transmit with the $V_{cm,dc}$ value specified in Table 52 while driving into a 100 ohm differential impedance.

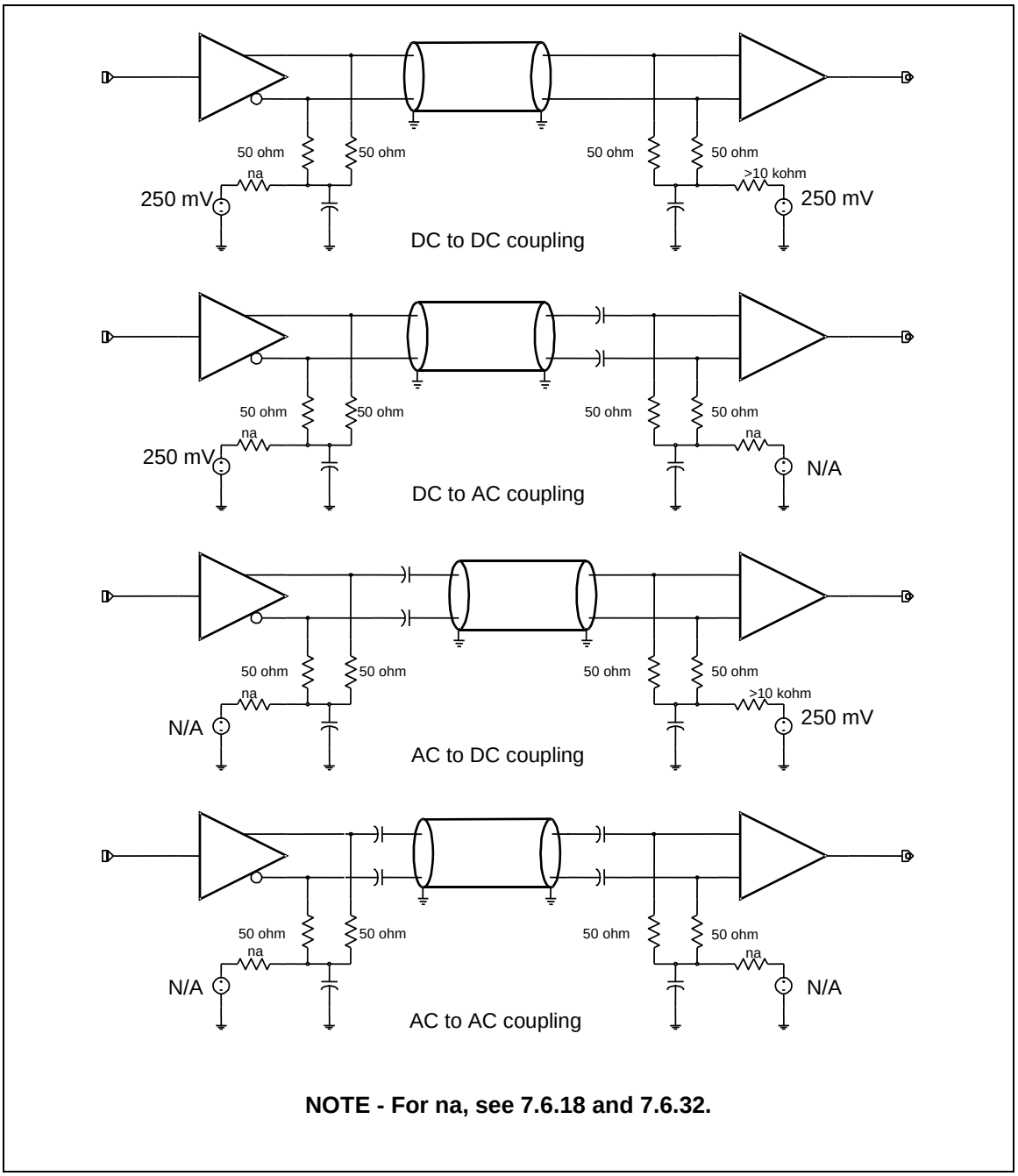
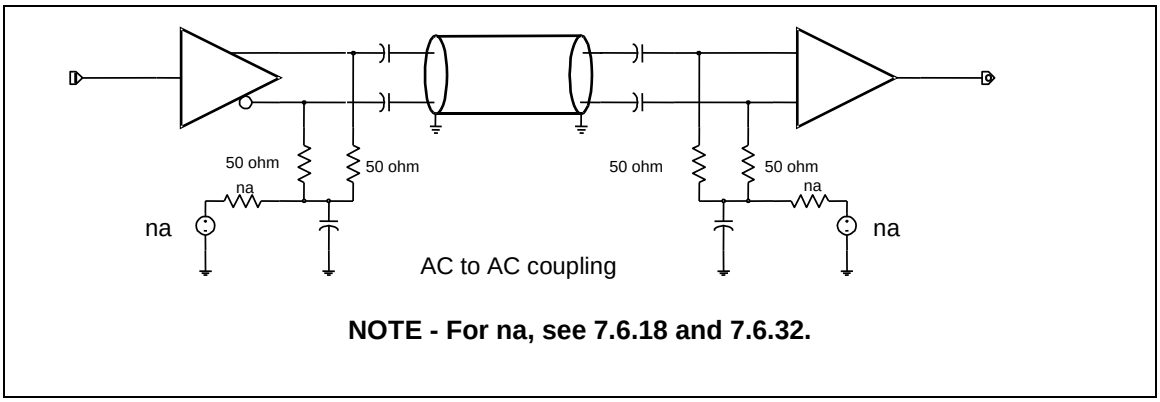


Figure 168 – Common mode biasing examples for Gen1i (informative)

7.4.3.1.9 AC coupled common mode voltage (Gen1m, Gen1u, Gen2i, Gen2m, Gen2u, Gen3i, Gen3u)

The SATA interfaces, defined as Gen1i, may be AC or DC coupled as shown in Figure 168. The SATA interfaces defined as Gen1m, Gen1u, Gen2i, Gen2m, Gen2u, Gen3i, and Gen3u shall be AC coupled. Figure 169 shows an example of a fully AC coupled system. The SATA MicroSSD application shall use the configuration shown in Figure 170.

Compliance points for SATA are defined at the connector. The AC coupled common mode voltage in Table 52 defines the open circuit DC voltage level of each single-ended signal at the IC side of the coupling capacitor in an AC coupled Phy and it shall be met during all possible power and electrical conditions of the Phy including power off and power ramping. Since the Gen2i, and Gen3i specification defines only the signal characteristics as observable at the connector, this value is not applicable to those specifications. The common mode transient requirements defined in Table 52 were determined sufficient to limit stresses on the attached components under transient conditions that was the sole intent of the AC coupled common mode voltage requirement. Due to this, the following is true even for Gen1i where $V_{cm,ac\ coupled}$ applies, AC coupled common mode voltage levels outside the specified range may be used provided that the transient voltage requirements of Table 52 are met.



NOTE - For na, see 7.6.18 and 7.6.32.

Figure 169 – Common mode biasing for Gen1m, Gen1u, Gen2i, Gen2m, Gen2u, Gen3i, and Gen3u

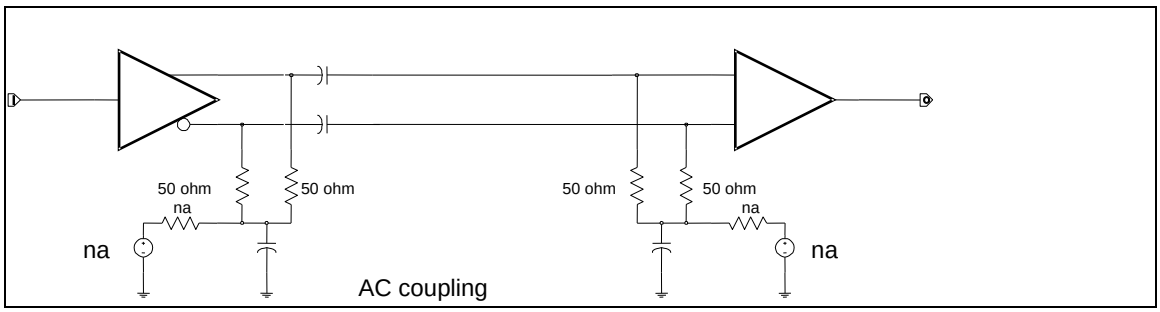


Figure 170 – Common mode biasing for SATA MicroSSD

7.4.3.1.10 Nominal differential impedance (Gen1i, Gen1m)

The nominal impedance of all components in a SATA system.

7.4.3.1.11 AC coupling capacitance

The value of the coupling capacitor used in AC coupled implementations. AC coupling is optional for Gen1i and mandatory for Gen1m, Gen1u, Gen2i, Gen2m, Gen2u, Gen3i, and Gen3u.

Coupling Capacitor Characteristics (Informative)**,** the physical size of the capacitor should be as small as practical to reduce the capacitance to ground. Body sizes larger than 0603 (or values less than 300 pF) should be avoided since they are likely to result in a failure of the return loss requirements in 7.4.3.2.4, 7.4.3.4.4, Table 53, and Table 54.

7.4.3.1.12 Sequencing transient voltage

This parameter addresses the transient voltages on the serial data bus during power sequencing and power mode changes. Since either the receiver or the transmitter may be affected by power sequencing transients, the term "aggressor" is used to indicate the sequencing interface circuit and the term "victim" is used for the interface circuit receiving the transient.

In order to limit the voltage and energy seen by the victim receiver or transmitter circuitry during power sequencing, several parameters of the aggressor and victim are involved. Although parameters of the victim, (e.g., common mode voltage and single ended impedance), affect the observed transient, this measurement addresses limiting the aggressor contribution.

The aggressor common mode voltage, single ended impedance, and AC coupling capacitor value determine the level of the sequencing transient. This measurement addressed the common mode voltage of the aggressor. The rate of change of the power on or power off ramp also affects this level. The limits provided allow for power up or power down ramps at rates faster than the time constants of the signal lines, although practical systems may not achieve this rate. This measurement shall include the test conditions of power on and power off ramping at the fastest possible rate expected in systems using the Phy, as well as any power mode transitions.

7.4.3.1.13 Sequencing transient voltage lab-load (Gen3i, Gen3u)

This parameter addresses the transient voltages on the serial data bus during power sequencing under the test condition of a lab-load (see 7.6.2 for lab-load details). Measuring the transient voltage with a lab-load combines the effects of the bias voltage and series termination. Separately measuring the impedance of the circuit is not required. Since some circuits calibrate the impedance after power ramping is compete, measuring impedance during a transient condition becomes challenging.

An open circuit voltage measurement, as outlined in 7.6.18, Sequencing Transient Voltage, shall be required to prevent overstressing victim circuits with high impedance common mode voltage.

7.4.3.2 Transmitter specification details

7.4.3.2.1 Transmitter specification scope

This section contains the details on Table 53 and Table 55 entries.

7.4.3.2.2 Tx pair differential impedance (Gen1i, Gen1m, Gen1u)

As seen by a differential TDR with max edge of 100 picoseconds looking into connector, measured from the 20 % threshold to the 80 % threshold. Measured with TDR in differential mode.

7.4.3.2.3 Tx single-ended impedance (Gen1i, Gen1m, Gen1u)

As seen by TDR with max edge of 100 picoseconds looking into connector, measured from the 20 % threshold to the 80 % threshold. The TDR is set to produce simultaneous positive pulses on both signals of the Tx pair. Single-ended impedance is the resulting (even mode) impedance of each signal. Both signals shall meet the single ended impedance requirement.

This requirement shall be met during all possible power and electrical conditions of the Phy including power off and power ramping.

7.4.3.2.4 Tx and Rx differential mode return loss (Gen2i, Gen2m) (Gen1i, Gen1m alternate)

This section describes transmitter output impedance and receiver input impedance in terms of both the peak value of a reflection given an incident step of known risetime and also in terms of return loss (see Figure 171). The return loss measurement shall be sufficient to verify compliance with Gen1 and Gen2 requirements. In order to ensure Gen1 designs passing the TDR differential impedance method as previously required are not invalidated due to this change, either method shall be sufficient to verify compliance with Gen1 requirements. Verification of compliance by both methods shall not be required.

The differential mode return loss is defined as the ratio (expressed in decibel (dB)) of differential mode incident power to differential mode reflected power both at a 100 ohm impedance level. In the system environment the purpose of controlling the return loss of devices and hosts is to limit signal reflections that cause data dependent jitter. These signal reflections in question are over and above those that exist in compliance testing if connected to a matched source or load.

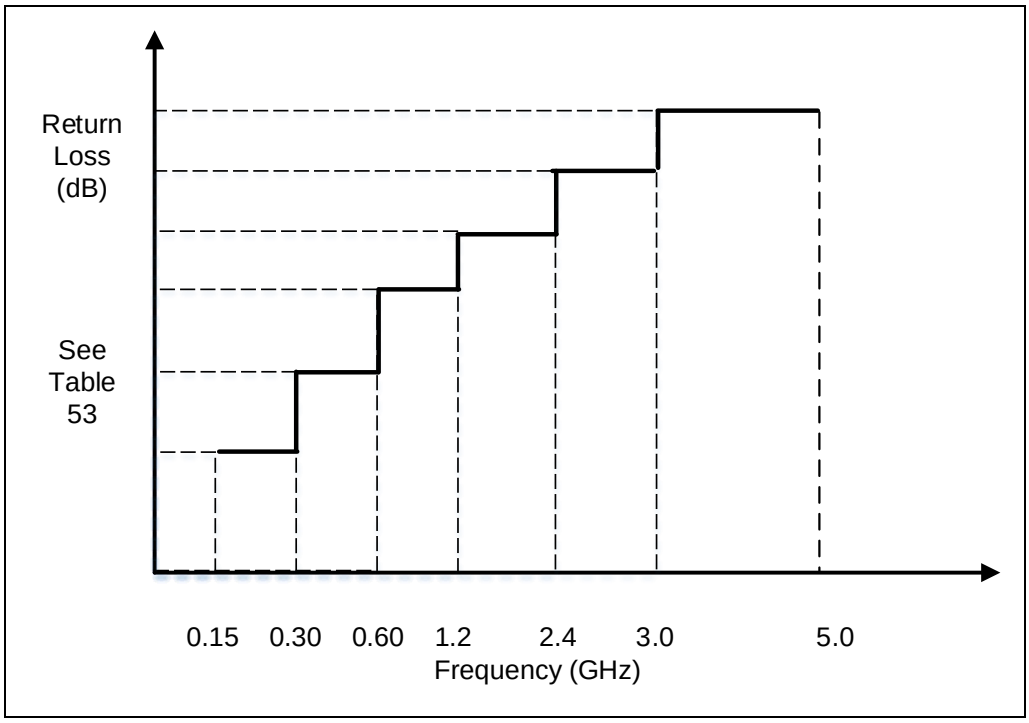


Figure 171 – Differential return loss limits

7.4.3.2.5 Tx common mode return loss (Gen2i, Gen2m)

The common mode return loss is defined as the ratio (expressed in dB) of common mode incident power to common mode reflected power both at a 25 ohm impedance level. The intended signal propagation mode in SATA is the differential mode. However, imperfections in the system create some coupling between the common and differential modes.

This has three consequences:

a) radiated emissions;
b) noise susceptibility; and
c) signal degradation.

Common mode reflections exacerbate these impairments. The common mode return loss is a bound on the magnitude of common mode reflections in the system (see Figure 172).

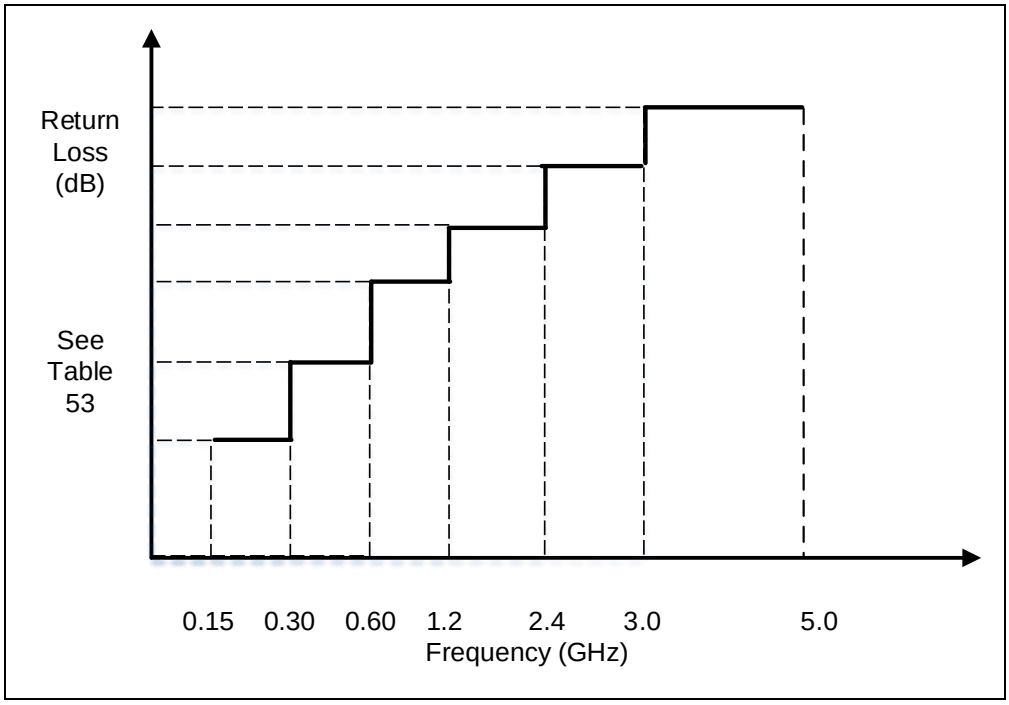


Figure 172 – Common mode return loss limits

7.4.3.2.6 Tx impedance balance (Gen2i, Gen2m, Gen3i)

Impedance balance is defined as the ratio (expressed in dB) of common mode incident power at a 25 ohm impedance level to differential mode reflected power at a 100 ohm impedance level. The impedance balance is a bound on the coupling between common and differential modes (see Figure 173).

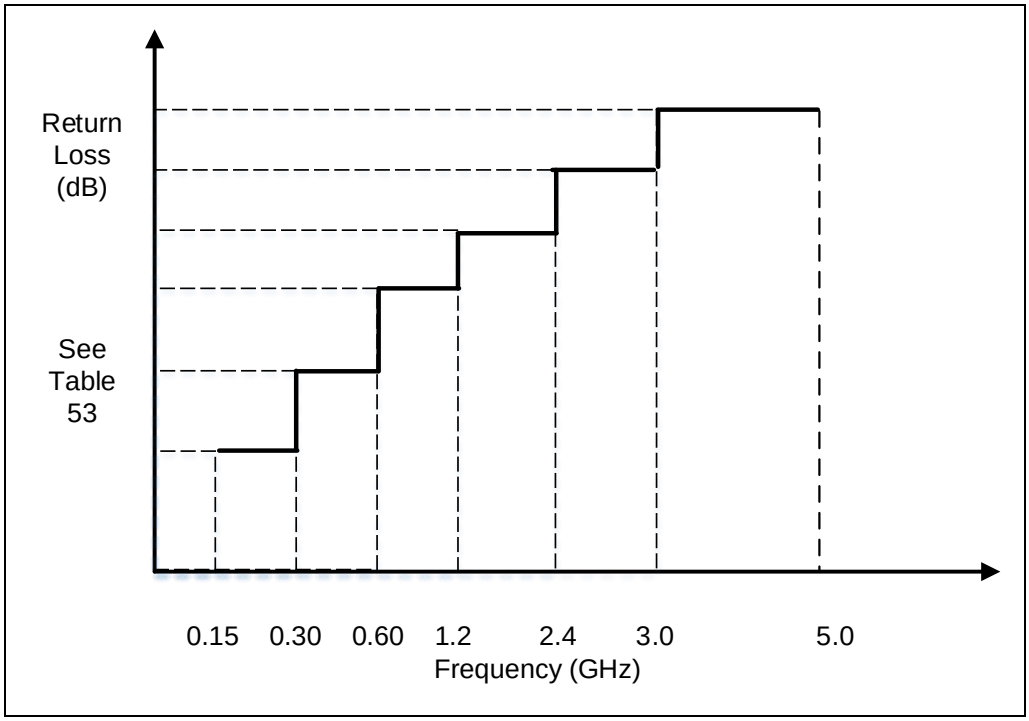


Figure 173 – Impedance balance limits

7.4.3.2.7 Tx and Rx differential mode return loss (Gen3i)

This section describes transmitter output impedance and receiver input impedance in terms of return loss. Return loss is specified as starting and ending points, with a defined slope between. The return loss shall remain below the line shown in Figure 174, from the starting frequency to the ending frequency.

The differential mode return loss is defined as the ratio (expressed in dB) of differential mode incident power to differential mode reflected power both at a 100 ohm impedance level. In the system environment the purpose of controlling the return loss of devices and hosts is to limit signal reflections that cause data dependent jitter. These signal reflections in question are over and above those that exist in compliance testing if connected to a matched source or load.

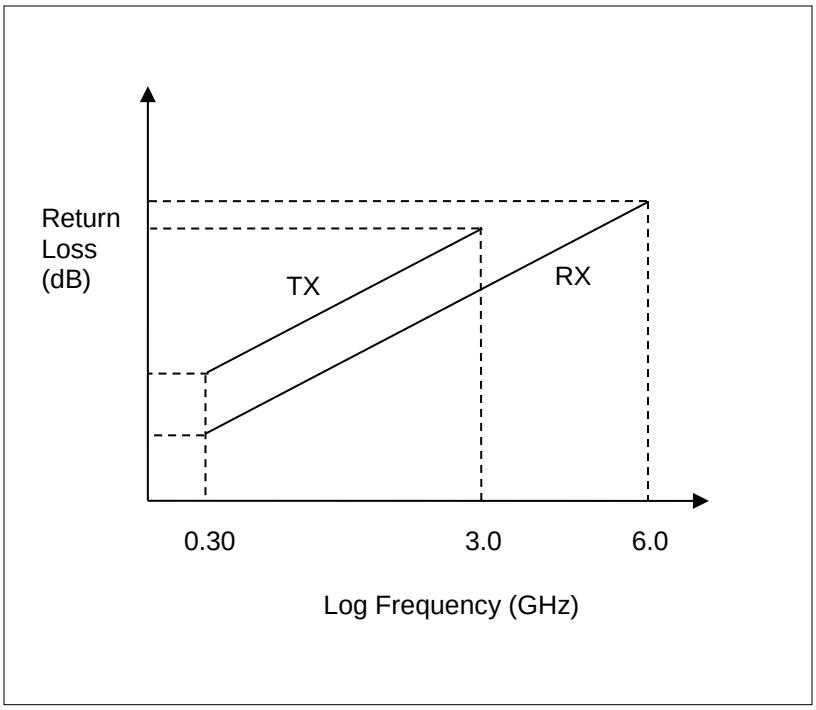


Figure 174 – Differential return loss limits, Gen3i, Tx and Rx

7.4.3.3 Transmitted signal requirements details

7.4.3.3.1 Transmitted signal requirements scope

This section contains the details on Table 52, Table 54, Table 55, and Table 58 entries.

7.4.3.3.2 Tx differential output voltage

The differential voltage [(Tx+) – (Tx-)] measured at the Transmitter shall comply with the respective electrical specifications as defined in Table 54 and Table 55.

This is measured at mated Serial ATA connector on transmit side including any pre-emphasis. For Gen3i and Gen3u the maximum differential output voltage is likewise measured at the Tx compliance point, but the minimum differential output voltage is measured after the Gen3i CIC. The minimum voltage for Gen3u is not measured after the Gen3i CIC (see 7.6.5).

7.4.3.3.3 Tx minimum voltage measurement interval

The point within a unit interval (UI) where the signal shall meet minimum levels.

7.4.3.3.4 Tx rise/fall time

Rise times and fall times are measured from the 20 % threshold and 80 % threshold of the signal, see Figure 175. The rise and fall time requirement $t_{r/f}$ applies to differential transitions (Tx+ – Tx-), for both normal and OOB signaling.

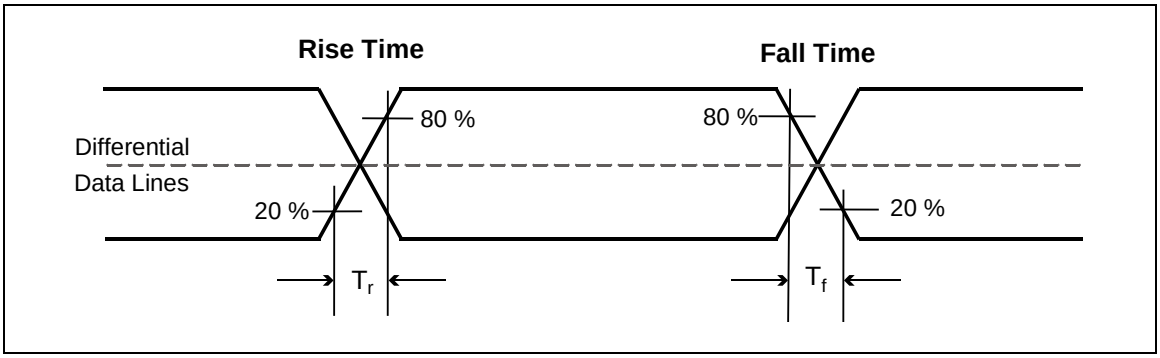


Figure 175 – Signal rise and fall times

7.4.3.3.5 Tx differential skew

Tx Differential Skew is the time difference between the single-ended mid-point of the Tx+ signal rising/falling edge, and the single-ended mid-point of the Tx- signal falling/rising edge (see Figure 176). It is an important parameter to control as excessive skew may result in increased high frequency jitter and common mode noise levels seen at the far end of the interconnect. The effects on the receiver are addressed in more detail (see 7.4.3.5.5). Excessive Tx Differential Skew also increases electromagnetic interference (EMI) emissions.

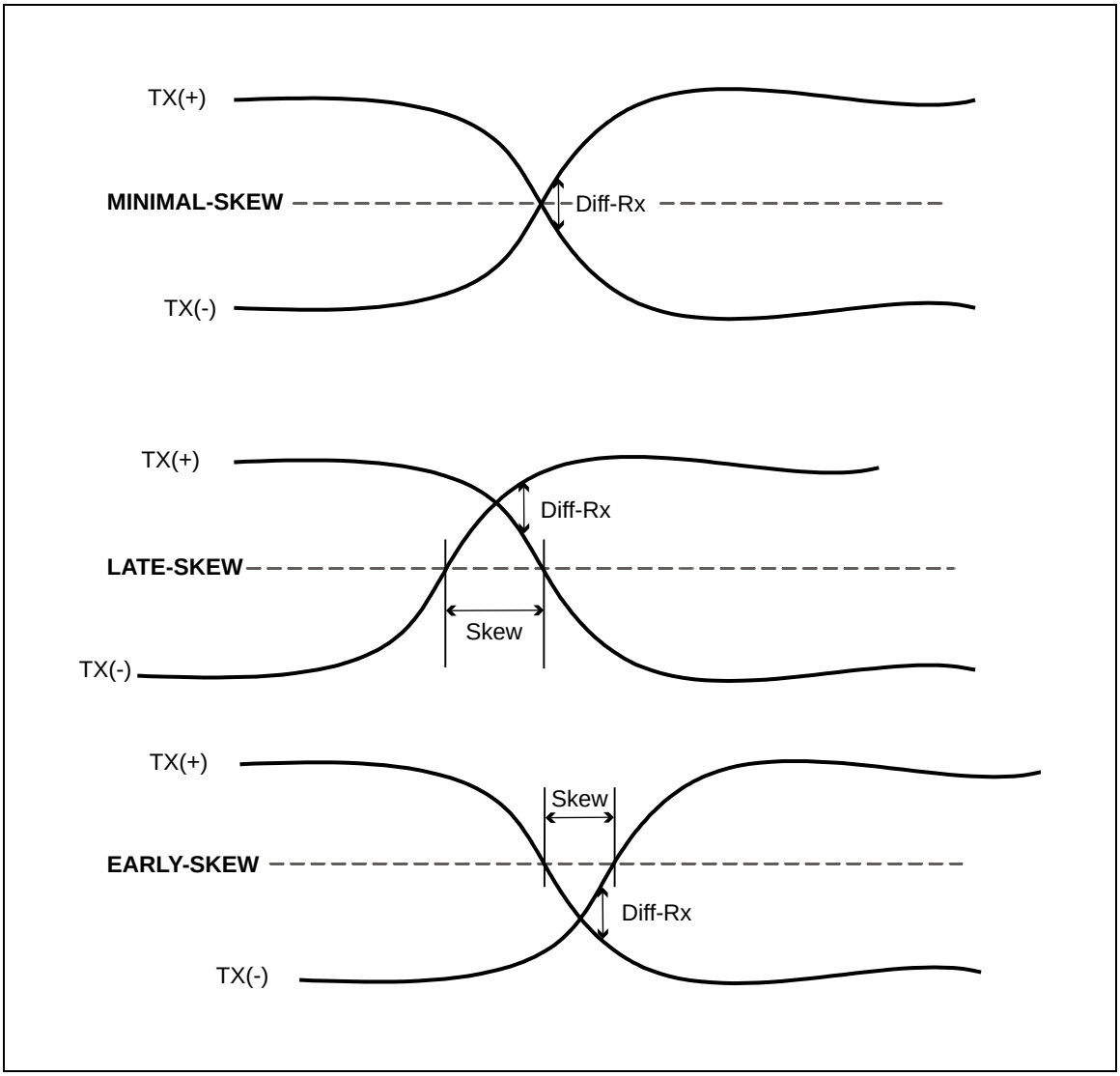


Figure 176 – Tx intra-pair skew

7.4.3.3.6 Tx AC common mode voltage (Gen2i, Gen2m)

Maximum peak-to-peak amplitude of common mode signal measured at the transmitter connector.

The Transmitter shall not deliver more output voltage than specified in Table 54 and Table 55 using the common mode voltage measuring technique defined in 7.6.22.

7.4.3.3.7 Tx AC common mode voltage (Gen1u, Gen2u, Gen3i, Gen3u)

Maximum peak-to-peak amplitude of common mode signal measured at the transmitter connector.

The Transmitter shall not deliver more output voltage than that specified in Table 54 and Table 55 using the common mode voltage measuring technique as defined in 7.6.23.

7.4.3.3.8 Common mode transient settle time (Gen1i, Gen1m)

In Gen1i transmitters, this is the maximum time for common-mode transients to settle to within 25 mV of their previous state common mode voltage during transitions to and from the idle bus condition.

7.4.3.3.9 OOB differential delta (Gen2i, Gen2m, Gen2u, Gen3i, Gen3u)

The difference between the average differential value during the idle bus condition and the average differential value during burst on transitions to and from the idle bus condition (see Figure 177).

During OOB transmission, imperfections and asymmetries in transmitters may generate error signals that impair proper detection by a receiver. The OOB Differential Delta describes an error from the difference in transmitter DC offset during the idle and active conditions. Since the transmitter is alternating between idle and active conditions each with different DC offsets, an Alternating Current (AC) error voltage is generated that is a square wave at about 1 / (2 × 106 ns) = 4.7 MHz. The AC error voltage propagates through the interconnect and causes an offset in the receiver OOB detector.

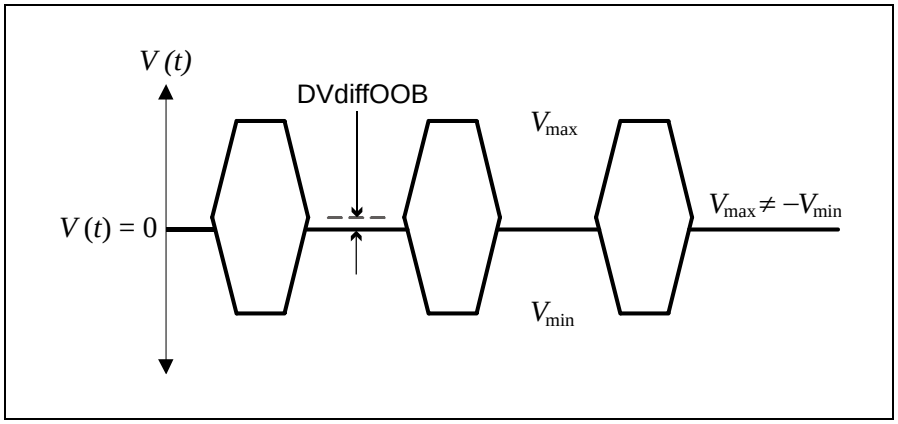


Figure 177 – OOB differential delta (at compliance point with AC coupling)

7.4.3.3.10 OOB common mode delta (Gen2i, Gen2m, Gen2u, Gen3i, Gen3u)

The difference between the common mode value during the idle bus condition and the common mode value during a burst on transitions to and from the idle bus condition.

7.4.3.3.11 Tx rise/fall imbalance (Obsolete)

The match in the rise of Tx+ and fall of Tx- determined by the functions:

absolute value (Tx+,rise – Tx-,fall)/average

where:

average = (Tx+,rise + Tx-,fall)/2

and all rise and fall times, measured from the 20 % threshold to the 80 % threshold.

The match in the fall of Tx+ and rise of Tx- determined by the function:

absolute value (Tx+,fall – Tx-,rise)/average

where:

average = (Tx+,fall + Tx-,rise)/2

and all rise and fall times, measured from the 20 % threshold to the 80 % threshold.

7.4.3.3.12 Tx amplitude imbalance (Gen2i, Gen2m, Gen2u, Gen3i, Gen3u)

The match in the amplitudes of Tx+ and Tx- determined by the function:

absolute value (Tx+ amplitude - Tx- amplitude)/average

where:

average is (Tx+ amplitude + Tx- amplitude)/2

and all amplitudes are determined by mode (most prevalent) voltage.

7.4.3.3.13 Clock-to-data transmit jitter (Gen1i, Gen1m, Gen1u, Gen2i, Gen2m, Gen2u, Gen3i, Gen3u)

Transmitters shall meet the jitter specifications for the Reference Clock characteristics specified in each case.

Table 54 and Table 55 shows the maximum amount of jitter that a transmitter may generate and still be SATA compliant, see 7.6.10 for a description of the measurement. Since this specification places the compliance point after the connector, any jitter generated at the package connection, on the printed circuit board, and at the board connector shall be included in the measurement.

7.4.3.3.14 Tx emphasis

7.4.3.3.14.1 Tx emphasis (Gen3i, Gen3u)

The emphasis measured at the transmitter shall comply with the respective electrical specifications as defined in Table 54 and Table 55.

For both host and device, Tx emphasis is measured at the mated Serial ATA connector at the device (unless otherwise specified). The Tx emphasis requirement does not apply at intermediate compliance points within the usage model channel (e.g., the internal 4-lane cable connector). Unless otherwise specified, no CIC is used for this measurement.

The TX emphasis requirement does not apply to devices using or mating with the following:

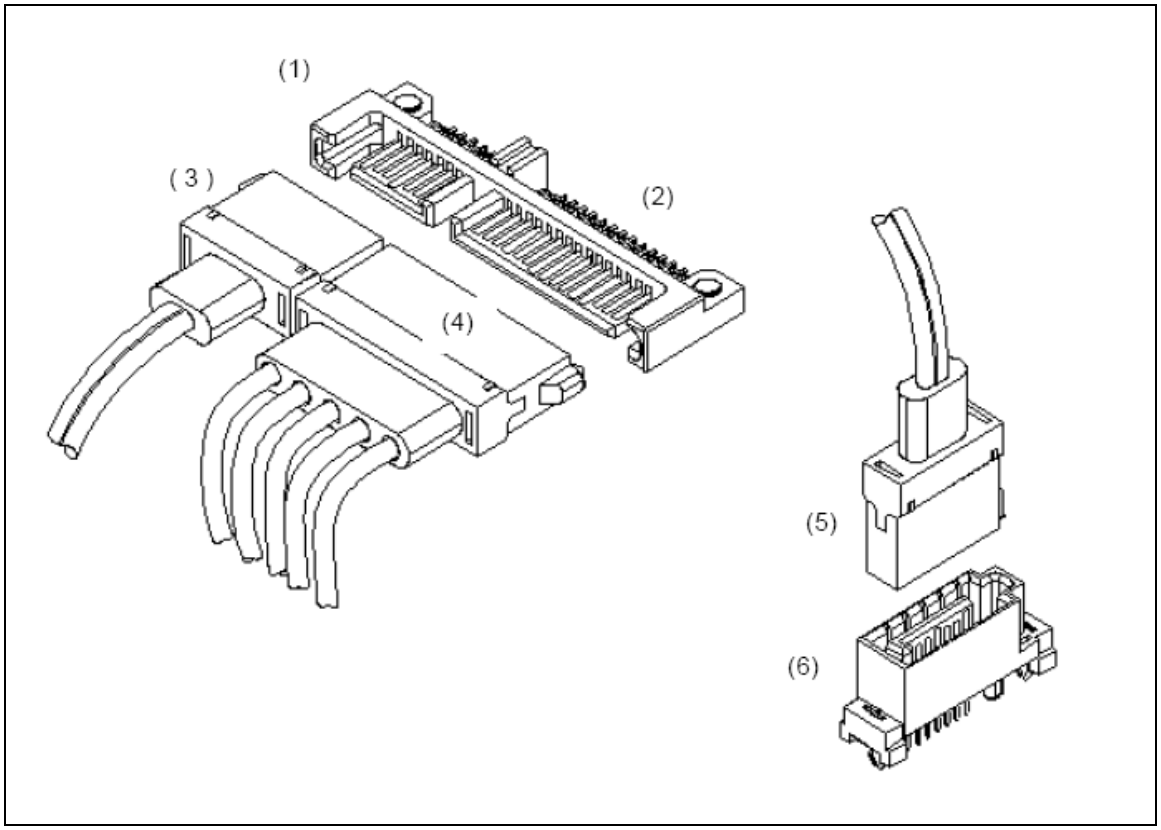
a) Internal LIF-SATA connector (see 6.5);
b) SATA MicroSSD interface (see 6.8); or

c) Internal M.2 connector (see 6.9).

TX emphasis for these devices is vendor specific.

7.4.3.3.14.2 Host Tx emphasis requirement for internal 1 m cabled host to device applications (Gen3i)

For connection via a standard SATA internal cable (see section 5.3.2), the device signal plug connector, shown as (1) in Figure 178, mates with the signal cable receptacle connector on one end of the cable, illustrated as (3) in Figure 178. The signal cable receptacle connector on the other end of the cable is inserted into a host signal plug connector (see section 6.2.5), shown as (6) in Figure 178. The signal cable wire consists of two twinax sections in a common outer sheath. When an application connects in this fashion, compliance with the host emphasis requirement shall be met by measuring the emphasis with the CIC rather than with the actual system cable.



Key:
1 = device signal plug segment or connector
2 = device power plug segment or connector
3 = signal cable receptacle connector, to be mated with (1)
4 = power cable receptacle connector, to be mated with (2)
5 = signal cable receptacle connector, to be mated with (6)
6 = the host signal plug connector

Figure 178 – Internal 1 m cabled host to device applications SATA connector examples

7.4.3.3.14.3 Device Tx emphasis (Gen1i, Gen2i) (informative)

The purpose of the Gen1 and Gen2 emphasis recommendation is to optimize signal integrity by adjusting the transmitted signal in such a way that it compensates for losses in the interconnect

channel. The particular implementation of emphasis is vendor specific. Since generally less emphasis is needed to compensate for losses at Gen1 and Gen2 than is needed for Gen3, it is recommended for devices with the capability of programming different levels of emphasis for different transfer rates that Gen2 device emphasis be limited to no more than 1.5 dB at the device connector and Gen1 device emphasis be limited to no more than 1 dB at the device connector.

7.4.3.3.14.4 Host Tx emphasis (Gen1i, Gen1u, Gen2i, Gen2u) (informative)

The purpose of the Gen1 and Gen2 emphasis recommendation is to optimize signal integrity by adjusting the transmitted signal in such a way that it compensates for losses in the interconnect channel. The particular implementation of emphasis is vendor specific. Since generally less emphasis is needed to compensate for losses at Gen1 and Gen2 than is needed for Gen3, it is recommended for hosts with the capability of programming different levels of emphasis for different transfer rates that Gen1 and Gen2 host emphasis be limited to no more than 1 dB at the device connector.

7.4.3.4 Receiver specification details

7.4.3.4.1 Receiver specification scope

This section contains the details on Table 56 and Table 58 entries.

7.4.3.4.2 Rx pair differential impedance (Gen1i, Gen1m, Gen1u)

As seen by a differential TDR with max edge of 100 picoseconds looking into connector, measured from the 20 % threshold to the 80 % threshold. Measured with TDR in differential mode.

7.4.3.4.3 Rx single-ended impedance (Gen1i, Gen1m, Gen1u)

As seen by TDR with max edge of 100 picoseconds looking into connector, measured from the 20 % threshold to the 80 % threshold.

TDR set to produce simultaneous positive pulses on both signals of the Rx pair. Single-ended impedance is the resulting (even mode) impedance of each signal. Both signals shall meet the single ended impedance requirement.

This requirement shall be met during all possible power and electrical conditions of the Phy including power off and power ramping.

7.4.3.4.4 Rx differential mode return loss (Gen2i, Gen2m)

Receiver differential mode return loss is measured similar to transmitter differential mode return loss (see 7.4.3.2.4).

7.4.3.4.5 Rx common mode return loss (Gen2i, Gen2m)

Receiver common mode return loss is measured similar to transmitter common mode return loss (see 7.4.3.2.5).

7.4.3.4.6 Rx impedance balance (Gen2i, Gen2m)

Receiver impedance balance is measured similar to transmitter impedance balance (see 7.4.3.2.6).

7.4.3.4.7 Rx differential mode return loss (Gen3i)

Receiver differential mode return loss is measured similar to transmitter differential mode return loss (see 7.4.3.2.7).

7.4.3.5 Receiver tolerance testing

7.4.3.5.1 Receiver tolerance testing scope

This section contains the details on Table 57 entries.

7.4.3.5.2 Rx differential input voltage

The Rx Differential Input Voltage is the range of input voltage under compliance test conditions that a receiver shall operate to the required link performance level. This is one range of input conditions a receiver shall tolerate (see 7.6.13).

The Serial ATA system has a transmitter and receiver with impedances near the nominal system impedance of 100 ohm. The voltage at compliance points is strongly dependent on the transmitter, receiver, and interconnect impedances. The Rx differential input voltage is delivered from an impedance matched signal source into a matched load (see Figure 179). If the actual receiver is substituted for the matched load, the voltage changes by an amount that is receiver design dependent. This change is part of the receiver design burden.

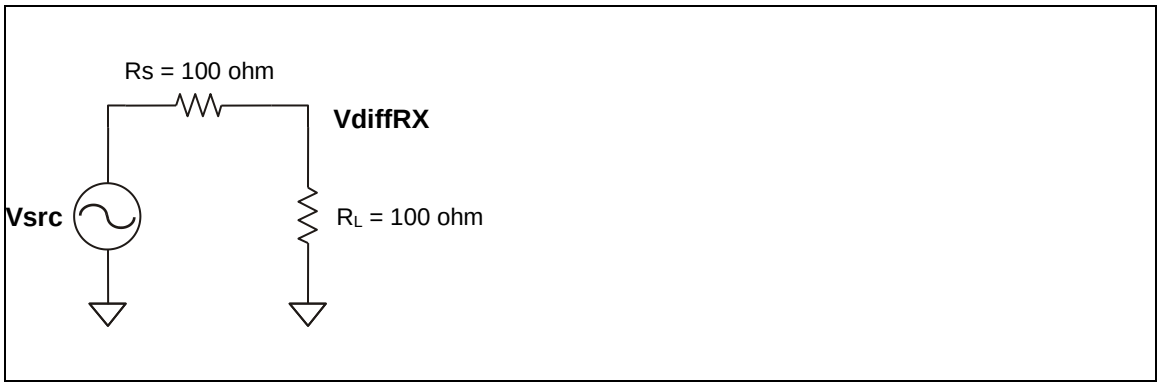


Figure 179 – Rx differential input Voltage conditions

The Rx differential input voltage does not describe the voltage delivered from the interconnect. The interconnect output impedance is not equal to the nominal system impedance over the entire frequency range. It is not the voltage at a matched load delivered from interconnects, nor is it the voltage at a receiver delivered from the interconnect. Example calculations and details as given in 7.6.7.

7.4.3.5.3 Rx rise/fall times

Rise times and fall times are measured from the 20 % threshold to the 80 % threshold of the signal. The rise and fall time requirement $t_{20\text{-}80Rx}$ applies to differential transitions (applied to Rx+ and Rx- for Gen1 and Gen2). For Gen3i the maximum and minimum Rx rise time and fall time requirements are applied to the Data Signal Source before the Gen3i CIC (see 7.6.14).

7.4.3.5.4 Rx minimum voltage measurement interval

The point in a UI that the signal shall meet minimum levels.

7.4.3.5.5 Rx differential skew (Gen2i, Gen2m, Gen2u, Gen3i, Gen3u)

Rx Differential Skew is the time difference between the single-ended mid-point of the Rx+ signal rising/falling edge, and the single-ended mid-point of the Rx- signal falling/rising edge, as measured at the Rx connector. The receiver should tolerate the Rx skew levels per Table 57 and Table 58, as generated by a Lab-Sourced Signal.

The receiver differential skew is an important parameter to consider, as excessive skew may result in increased high frequency jitter and high frequency common mode noise seen at the high-speed differential receiver. Figure 180 depicts how late and early skew signaling affect the time that the differential receiver resolves the differential input signals. For the minimal skew case, if the single-ended slew rate is at maximum, at the crossover, the UI width is also maximized. However, this is not the case for the early and late skew cases. The high frequency common mode noise is a result of the rapid changing of the operating point of the high-speed receiver. As described in 7.6.17 the applicable measurement method that should be used to calibrate the intentionally skewed Lab-Sourced Signal output into the receiver.

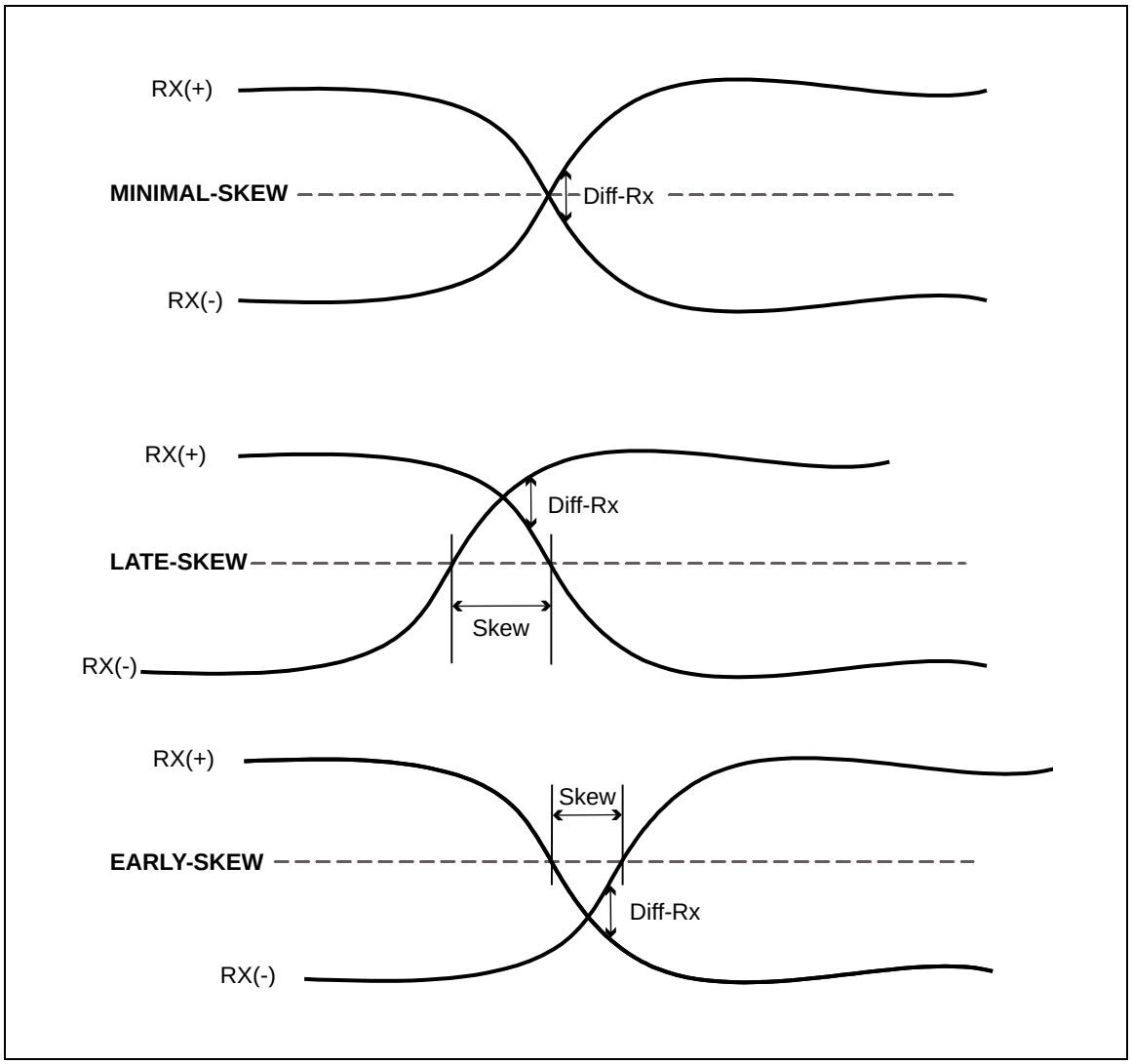


Figure 180 – Rx intra-pair skew

7.4.3.5.6 Rx AC common mode voltage

Max peak-to-peak sinusoidal amplitude of AC common mode signal [(Rx+) + (Rx-)]/2.
The Receiver shall operate to within the Frame Error Rate cited in Table 52, if subjected to a sinusoidal common mode interfering signal with peak-to-peak voltage VcmRx,ac defined in Table 57 and Table 58 and swept across the frequency range, fcm,acRx, defined in Table 57 and Table 58 at a sweep rate period no shorter than 33.33 us.

7.4.3.5.7 AC common mode frequency

All receivers shall be able to tolerate sinusoidal common-mode noise components inside this frequency range with amplitude of $V_{cm,acRx}$.

7.4.3.5.8 Clock-data receiver jitter tolerance (Gen1i, Gen1m, Gen1u, Gen2i, Gen2m, Gen2u)

Jitter tolerance is the ability of the receiver to recover data in the presence of jitter. The minimum amount of jitter that a receiver shall be able to tolerate is the jitter tolerance specification provided in Table 57 and Table 58. Section 7.6.13 describes the measurement method for Gen1 and Gen2.

7.4.3.5.9 Clock-data receiver jitter tolerance (Gen3i, Gen3u)

See 7.6.14 for Gen3i and Gen3u jitter tolerance measurement details.

7.4.3.6 OOB specifications details

7.4.3.6.1 OOB specifications details scope

This section provides details on Table 59**.**

7.4.3.6.2 OOB signal burst generation

Out Of Band (OOB) signals are groupings of waveforms made up of low frequency, waveform bursts, interspersed with idle gaps. These do not appear during normal data stream transfers, but are used to communicate low frequency identification information during initial notification and calibration periods, before data transfers begin. The OOB signal consists of a defined amount of idle time followed by a defined amount of burst time, this combination repeated for multiple iterations. During the idle time, the physical link carries DC idle. During the burst time, the physical link carries low frequency signal transitions intended to be repetitive waveshapes for envelope detection means. These bursts are translated into ON/OFF times as a means for very low speed communication. The OOB signal OFF time is determined by the length of idle time between the waveform bursts.

The signal patterns used during the OOB bursts shall be comprised of D24.3 characters (preferred) or $ALIGN_P$ primitives (allowed), transmitted at the Gen1 rate. The OOB burst is only required to generate an envelope for detection by AC coupled detection circuitry. A burst of D24.3 characters at Gen1 speed is equivalent to a square wave pattern that is a 1 for 2 UI_{OOB} periods and then a 0 for 2 UI_{OOB} periods, or simply, a squarewave with a period of 2.66 ns. All data speed generations shall use the Gen1 OOB burst speed, establishing a singular requirement for all OOB detection circuitry.

7.4.3.6.3 OOB signal detection threshold

Differential signal amplitude detected as activity by the squelch detector during OOB signaling.

V_{diffRx} signals less than the minimum V_{thresh} defined in Table 59 shall not be detected as activity. Signal levels greater than the maximum V_{thresh} defined in Table 59 shall be detected as activity.

7.4.3.6.4 UI during OOB signaling (UI_{OOB})

Average data period during OOB burst transmission (at Gen1 speed with a relative tolerance of ± 3 % of nominal).

7.4.3.6.5 COMINIT/COMRESET and COMWAKE transmit burst length

Burst length in nanoseconds, as measured from the first crossing point (+100 mV or –100 mV) of the burst to the last crossing point (+100 mV or –100 mV) of the burst.

7.4.3.6.6 COMINIT/COMRESET transmit gap length

Gap length in nanoseconds, as measured from the last crossing point (+100 mV or –100 mV) of one COMINIT/COMRESET burst to the first crossing point (+100 mV or –100 mV) of the following COMINIT/COMRESET burst.

7.4.3.6.7 COMWAKE transmit gap length

Gap length in nanoseconds, as measured from the last crossing point (+100 mV or –100 mV) of one COMWAKE burst to the first crossing point (+100 mV or –100 mV) of the following COMWAKE burst.

7.4.3.6.8 COMWAKE gap detection windows

Three timing ranges defining the validation and invalidation of COMWAKE gaps (see Table 59)**.**

Any OOB gap between bursts falling in the defined “may detect” range may be recognized as a valid COMWAKE gap.

Any OOB gap between bursts falling in the “shall detect” range shall be recognized as a valid COMWAKE gap.

Any OOB gap between bursts falling in the “shall not detect” ranges shall be recognized as an invalid COMWAKE gap (i.e., not be recognized as a valid COMWAKE gap).

7.4.3.6.9 COMINIT/COMRESET gap detection windows

Three timing ranges defining the validation and invalidation of COMINIT and COMRESET gaps (see Table 59).

Any OOB gap between bursts falling in the defined “may detect” range may be recognized as a valid COMINIT or COMRESET gap.

Any OOB gap between bursts falling in the “shall detect” range shall be recognized as a valid COMINIT or COMRESET gap.

Any OOB gap between bursts falling in the “shall not detect” ranges shall be recognized as an invalid COMINIT or COMRESET gap (shall not be recognized as a valid COMINIT or COMRESET gap).

7.4.4 Loopback

7.4.4.1 Loopback overview

In addition to meeting all electrical specifications in Table 52 to Table 58, all hosts and devices shall provide Far-End Retimed Loopback mode. Two other loopback modes are optional (see Table 60), but if implemented shall comply as defined in 7.4.4.3 and 7.4.4.4.

Table 60 – Loopback modes

Loopback modes	Required or Optional
Far-End Retimed	Required
Far-End Analog	Optional
Near-End Analog (Effectively Retimed)	Optional

7.4.4.2 Far-end retimed

Figure 181, illustrates the scope, at the architectural block diagram level, of the Far-End Retimed Loopback. As this loopback scheme needs a specific action from the far-end connected interface, this mode shall be entered by way of the BIST Activate FIS as defined in 10.5.10.

The Far-End Interface shall remain in this Far-End Retimed Loopback until receipt of the COMRESET/COMINIT OOB Signaling sequence.

As a minimum, Far-End Retimed Loopback shall involve far-end circuitry such that the data stream, at the Far-End interface, is extracted by the deserializer and data recovery circuit (DRC) before being sent back through the serializer and transmitter with appropriately inserted retiming $ALIGN_P$ primitives as defined in 7.8. The data may be decoded and descrambled in order to provide testing coverage for those portions of the device, provided the data is re-scrambled using the same sequence of scrambler syndromes. The returned data shall be the same as the received data with the exception that the returned data may be encoded with different starting running disparity.

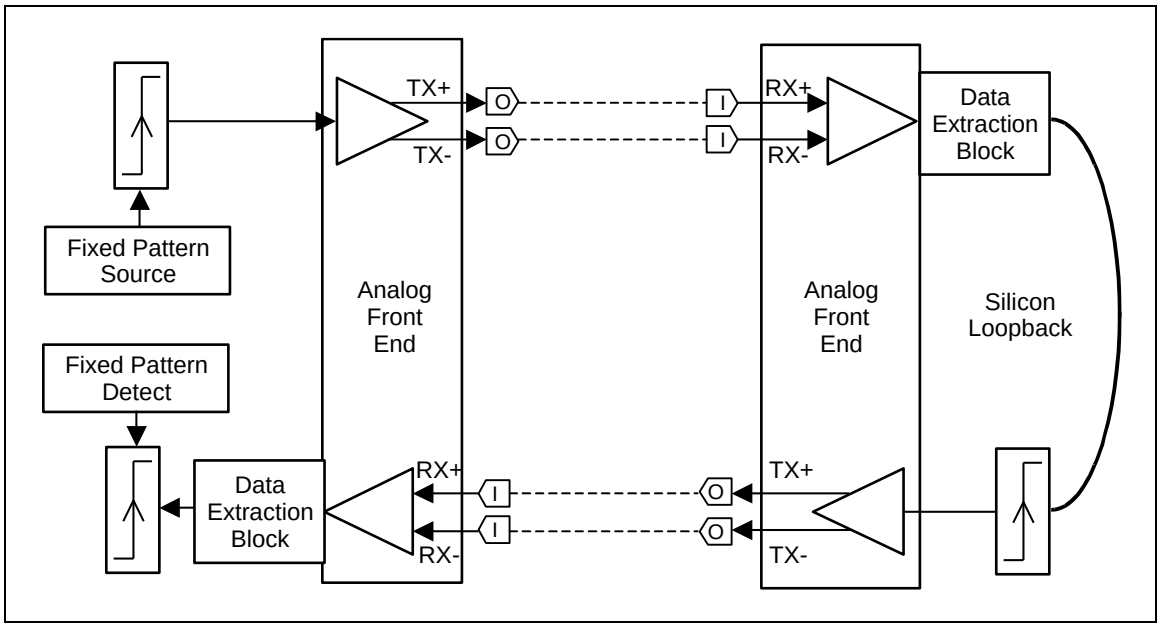


Figure 181 – Far-End Retimed Loopback

7.4.4.3 Far-end analog (optional)

Figure 182 illustrates the scope, at the architectural block diagram level, of the Far-End Analog Loopback. As this loopback scheme needs a specific action from the far-end connected interface, this mode shall be entered by way of the BIST Activate FIS as defined in 10.5.10.

The Far-End Interface shall remain in this Far-End Analog Loopback mode, until receipt of a COMRESET or COMINIT.

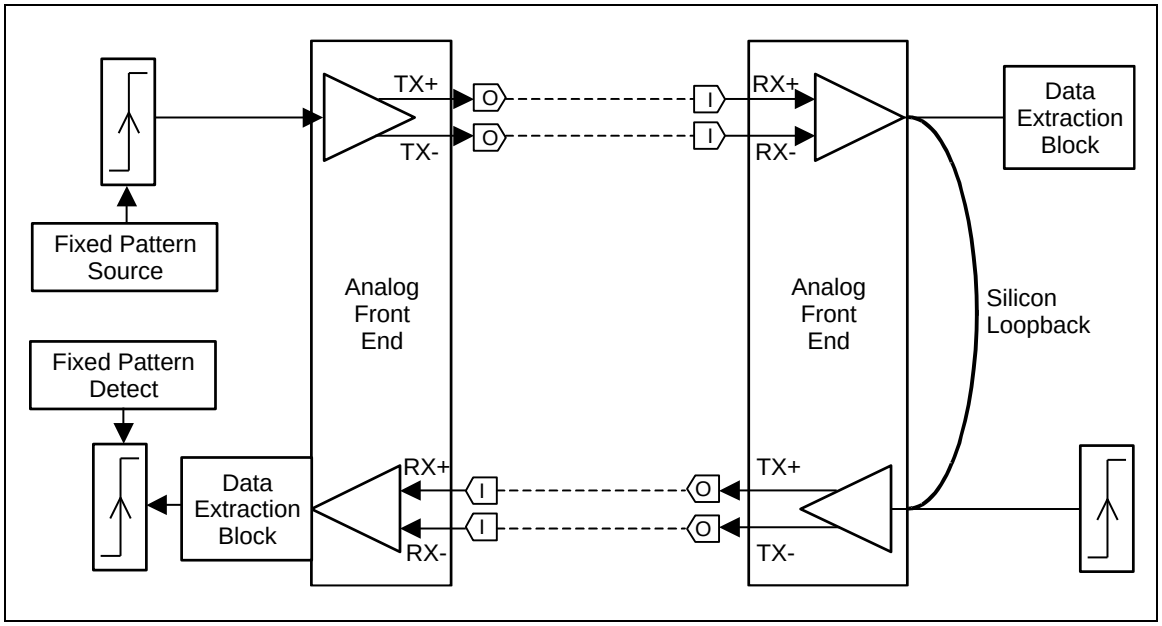


Figure 182 – Far-End Analog Loopback

7.4.4.4 Near-end analog (optional)

Figure 183 illustrates the scope, at the architectural block diagram level, of the Near-End Analog Loopback. This loopback scheme needs the far-end connected interface to be in a non-transmitting mode, (e.g., Slumber, or Partial interface power management states). Entry to and exit from this mode is vendor specific.

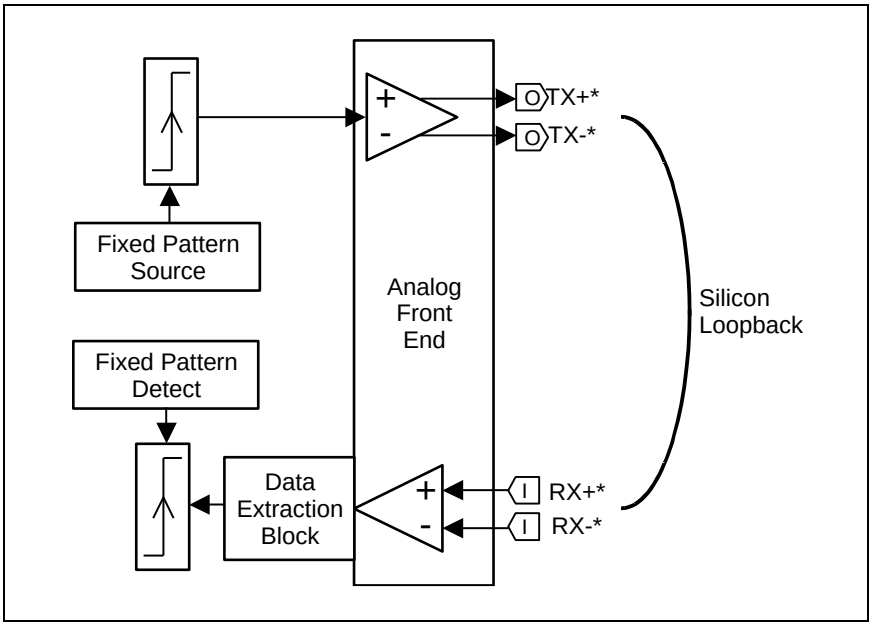


Figure 183 – Near-End Analog Loopback

7.4.5 Test pattern requirements

7.4.5.1 Test pattern requirements overview

Test patterns shall be used for compliance testing of the Serial ATA interfaces. This section defines various patterns to be used in compliance testing. The requirements as described in 7.6 define the patterns that are to be used for specific tests.

The patterns are classified in two categories:

a) non-compliant patterns; and
b) compliant patterns.

Non-compliant patterns are those patterns that are used for baseline jitter measurements and assessment of signal quality, given specified stimulus. These patterns do not comply with the required FIS formats, but are just a repeated selected set of 8b/10b characters.

Compliant patterns are those specified patterns that contain the leading SOF_P primitive, the specified pattern as data content, and trailing CRC and EOF_P primitive. There is no suppression of the two consecutive $ALIGN_P$ primitives during stimulus with this class of pattern.

Test patterns cited in this section are used as stimulus to verify interface compliance and signal integrity, using the following test models:

a) Non-compliant test patterns for:
    A) jitter measurements;
    B) physical connection media tests; and
    C) electrical parameter testing;

and

b) Compliant test patterns for:
   A) Frame Error Rate testing; and
   B) in-system tests.

7.4.5.2 Non-compliant patterns

Electrical parameters as defined in 7.4 shall be verified using the patterns identified in the measurement method in accordance with 7.6:

a) Lone Bit Patterns (LBP) as defined in 7.4.5.4.6;
b) High Frequency Test Pattern (HFTP) as defined in 4.1.1.68;
c) Mid Frequency Test Pattern (MFTP) as defined in 4.1.1.89; or
d) Low Frequency Test Pattern (LFTP) as defined in 4.1.1.87.

7.4.5.3 Compliant frame patterns

The Frame Error Rates as defined in 7.6.3.3 shall be tested for compliance when subjected to any implementation-determined worst-case compliant patterns, as well as the following set of compliant patterns:

a) Compliant Framed Composite patterns as defined in 7.4.5.4.8; and
b) Compliant Lone Bit Patterns as defined in 7.4.5.4.6.

Where the qualifying prefix term "compliant" signifies transmission of the cited pattern encapsulated in payload of a Data FIS, and used in a Serial ATA operational transmission context.

The N parameters of the reference patterns shall be extended to achieve the maximum frame length. These compliant patterns contain the necessary SOF_P leading primitive, the Dword header containing the FIS Type indicating a Data FIS, the specified test pattern, the calculated CRC, and the trailing EOF_P, as shown in Figure 184. To generate these patterns on the SATA link, scrambling needs to be taken into account.

NOTE 21 - Note that the cited patterns appear on the wire.

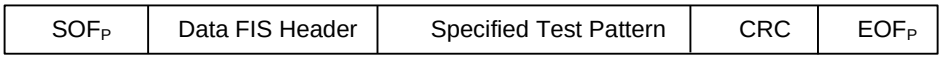


Figure 184 – Compliant test patterns

7.4.5.4 Test bit patterns and sequence characteristics

7.4.5.4.1 Test bit patterns and sequence characteristics overview

There are various types of bit sequence patterns that emphasize low/high transition density patterns, as well as low/high frequency patterns:

a) Low Transition Density Patterns (LTDP) are those patterns containing long runs of ones and zeroes, intended to create inter-symbol interference by varying the excursion times at either extreme of the differential signaling levels;
b) High Transition Density Patterns (HTDP) are those patterns containing short runs of ones and zeroes, also intended to create inter-symbol interference;
c) Low Frequency Spectral Content Patterns (LFSCP) are a good test of the input high pass filter circuitry, more specifically, introduced amplitude signal distortion, due to a marginal design. These bit patterns are a better test than those bit patterns having high frequency spectral content;
d) Simultaneous Switching Outputs Patterns (SSOP) are achieved by transmitting alternating ones complement bit patterns (10 bits) for recovery at the receiver. These patterns create worst case power supply, or chip substrate, noise, and are achieved by selecting bit test pattern sequences that maximize current extremes at the recovered bit pattern parallel

interface. These patterns induce Ldi/dt noise into substrate supply, and are a good test of the receiver circuitry;

e) The Lone Bit Patterns (LBP) are comprised of the consecutive combination of certain 8b/10b encoded patterns that result in a lone bit. These patterns create a condition where the preceding 4 bit run-length results in minimum amplitude of the lone bit as well as its time-width in comparison to its surrounding segments. This is often the worst-case condition that the receiving data recovery circuits may encounter; or

f) The intent of random bit patterns is to provide those patterns containing sufficiently broad spectral content, and minimal peaking, that should be used for both component, and system level architecture measurement of jitter output, and bit error rate (BER) performance. These patterns are also intended to be the common baseline pattern stimulus, for system/component vendor comparative testing, attributing the transmit jitter output measurement to the component performance, and not to the spectral profile of the data pattern used.

7.4.5.4.2 Low transition density patterns (LTDP)

Low transition density bit patterns (LTDP), as shown in Table 61 and Table 62, contain long runs of ones and zeroes. These patterns create jitter due to inter-symbol interference. This is aggravated by other sections of the composite pattern as defined in 7.4.5.4.7. Bit sequences are shown for both cases, where the starting running disparity is negative (RD-) or positive (RD+).

Table 61 – Low Transition Density Pattern (LTDP) starting with RD-

	Transmission Order →									
	D17.7(F1h)-			D30.7(FEh)+		D7.1(27h)+			D14.7(EEh)+	
-	1000b	1101b	1110b	0001b	1110b	0001b	1110b	0101b	1100b	1000b	-
	8h	Dh	Eh	1h	Eh	1h	Eh	5h	Ch	8h	
	D30.7(FEh)-			D7.6(C7h)-		D30.3(7Eh)-			D30.3(7Eh)+	
-	0111b	1000b	0111b	1000b	0110b	0111b	1000b	1110b	0001b	1100b	-
	7h	8h	7h	8h	6h	7h	8h	Eh	1h	Ch	
	D30.3(7Eh)-			D30.3(7Eh)+		D30.3(7Eh)-			D30.3(7Eh)+	
-	0111b	1000b	1110b	0001b	1100b	0111b	1000b	1110b	0001b	1100b	-
	7h	8h	Eh	1h	Ch	7h	8h	Eh	1h	Ch	
	Above Dword is repeated a total of 2 045 times for long version. Above Dword is repeated a total of 125 times for short version.									
	D3.7(E3h)-			D28.7(FCh)+		D3.7(E3h)-			D28.7(FCh)+	
-	1100b	0111b	1000b	1110b	0001b	1100b	0111b	1000b	1110b	0001b	-
	Ch	7h	8h	Eh	1h	Ch	7h	8h	Eh	1h	
	Long version total 3 + 2 045 = 2 048 Dwords. Short version total 3 + 125 = 128 Dwords.									

Table 62 – Low Transition Density Pattern (LTDP) starting with RD+

Transmission Order →

	D14.7(EEh)+			D30.7(FEh)-		D7.6(C7h)+			D17.7(F1h)-		
+	0111b	0010b	0001b	1110b	0001b	1110b	0001b	1010b	0011b	0111b	+
	7h	2h	1h	Eh	1h	Eh	1h	Ah	3h	7h	
	D30.7(FEh)+			D7.1(27h)+		D30.3(7Eh)+			D30.3(7Eh)-		
+	1000b	0111b	1000b	0111b	1001b	1000b	0111b	0001b	1110b	0011b	+
	8h	7h	8h	7h	9h	8h	7h	1h	Eh	3h	
	D30.3(7Eh)+			D30.3(7Eh)-		D30.3(7Eh)+			D30.3(7Eh)-		
+	1000b	0111b	0001b	1110b	0011b	1000b	0111b	0001b	1110b	0011b	+
	8h	7h	1h	Eh	3h	8h	7h	1h	Eh	3h	
	Above Dword is repeated a total of 2 045 times for long version. Above Dword is repeated a total of 125 times for short version.										
	D28.7(FCh)+			D3.7(E3h)-		D28.7(FCh)+			D3.7(E3h)-		
+	0011b	1000b	0111b	0001b	1110b	0011b	1000b	0111b	0001b	1110b	+
	3h	8h	7h	1h	Eh	3h	8h	7h	1h	Eh	

Long version total 3 + 2 045 = 2 048 Dwords
Short version total 3 + 125 = 128 Dwords

7.4.5.4.3 High transition density patterns (HTDP)

High transition density patterns are those patterns containing short runs of ones and zeroes, as shown in Table 63 and Table 64. These patterns create jitter due to inter-symbol interference, becoming more pronounced due to part of the composite pattern as defined in 7.4.5.4.7.

There are two types of high-transition density patterns of interest:

a) Full-rate high transition density bit pattern sequence; or
b) Half-rate high transition density bit pattern sequence.

Both types are used in the high transition density test pattern. Bit sequences are shown for both cases, where the starting running disparity is negative (RD-) or positive (RD+).

Table 63 – High Transition Density Pattern (HTDP) starting with RD-

Transmission Order →										
-	D21.5(B5h)-		D21.5(B5h)-			D21.5(B5h)-		D21.5(B5h)-		
	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b
	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah
	Above Dword is repeated a total of 512 times for long version. Above Dword is repeated a total of 32 times for short version.									
-	D24.3(78h)-		D24.3(78h)+			D24.3(78h)-		D24.3(78h)+		
	1100b	1100b	1100b	1100b	1100b	1100b	1100b	1100b	1100b	1100b
	Ch	Ch	Ch	Ch	Ch	Ch	Ch	Ch	Ch	Ch
	Above Dword is repeated a total of 512 times for long version. Above Dword is repeated a total of 32 times for short version.									
-	D10.2(4Ah)-		D10.2(4Ah)-			D10.2(4Ah)-		D10.2(4Ah)-		
	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b
	5h	5h	5h	5h	5h	5h	5h	5h	5h	5h
	Above Dword is repeated a total of 512 times for long version. Above Dword is repeated a total of 32 times for short version.									
-	D25.6(D9h)-		D6.1(26h)+			D25.6(D9h)-		D6.1(26h)+		
	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b
	9h	9h	9h	9h	9h	9h	9h	9h	9h	9h
	Above Dword is repeated a total of 512 times for long version. Above Dword is repeated a total of 32 times for short version.									
Long version total 4 × 512 = 2 048 Dwords Short version total 4 × 32 = 128 Dwords										

Table 64 – High Transition Density Pattern (HTDP) starting with RD+

	Transmission Order →										
	D21.5(B5h)+			D21.5(B5h)+		D21.5(B5h)+			D21.5(B5h)+		
+	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	+
	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	
	Above Dword is repeated a total of 512 times for long version. Above Dword is repeated a total of 32 times for short version.										
	D24.3(78h)+			D24.3(78h)+		D24.3(78h)+			D24.3(78h)+		
+	0011b	0011b	0011b	0011b	0011b	0011b	0011b	0011b	0011b	0011b	+
	3h	3h	3h	3h	3h	3h	3h	3h	3h	3h	
	Above Dword is repeated a total of 512 times for long version. Above Dword is repeated a total of 32 times for short version.										
	D10.2(4Ah)+			D10.2(4Ah)+		D10.2(4Ah)+			D10.2(4Ah)+		
+	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	+
	5h	5h	5h	5h	5h	5h	5h	5h	5h	5h	
	Above Dword is repeated a total of 512 times for long version. Above Dword is repeated a total of 32 times for short version.										
	D25.6(D9h)+			D6.1(26h)+		D25.6(D9h)+			D6.1(26h)+		
+	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	+
	9h	9h	9h	9h	9h	9h	9h	9h	9h	9h	
	Above Dword is repeated a total of 512 times for long version. Above Dword is repeated a total of 32 times for short version.										
Long version total 4 × 512 = 2 048 Dwords Short version total 4 × 32 = 128 Dwords											

7.4.5.4.4 Low frequency spectral content pattern (LFSCP)

Bit patterns that contain low frequency spectral components, as shown in Table 65 and Table 66, are a good test of the interconnect transmission, especially any AC coupling capacitors. Poor transmission through these components introduces signal distortion shown by this test pattern. Bit sequences are shown for both cases, where the starting running disparity is negative (RD-) or positive (RD+).

Table 65 – Low Frequency Spectral Content Pattern (LFSCP) starting with RD-

Transmission Order →

-	D20.2(54h)-			D20.2(54h)-			D20.2(54h)-			D20.2(54h)-	-
	0010b	1101b	0100b	1011b	0101b	0010b	1101b	0100b	1011b	0101b	
	2h	Dh	4h	Bh	5h	2h	Dh	4h	Bh	5h	

Above Dword is repeated a total of 1 023 times for long version.
Above Dword is repeated a total of 63 times for short version.

-	D20.2(54h)-			D20.7(F4h)-			D11.5(ABh)+			D11.5(ABh)+	+
	0010b	1101b	0100b	1011b	0111b	1101b	0010b	1011b	0100b	1010b	
	2h	Dh	4h	Bh	7h	Dh	2h	Bh	4h	Ah	

+	D11.5(ABh)+			D11.5(ABh)+			D11.5(ABh)+			D11.5(ABh)+	+
	1101b	0010b	1011b	0100b	1010b	1101b	0010b	1011b	0100b	1010b	
	Dh	2h	Bh	4h	Ah	Dh	2h	Bh	4h	Ah	

Above Dword is repeated a total of 1 023 times for long version.
Above Dword is repeated a total of 63 times for short version.

+	D11.5(ABh)+			D11.7(EBh)+			D20.2.(54h)-			D20.2(54h)-	-
	1101b	0010b	1011b	0100b	1000b	0010b	1101b	0100b	1011b	0101b	
	Dh	2h	Bh	4h	8h	2h	Dh	4h	Bh	5h	

Long version total 2 + (2 × 1 023) = 2 048 Dwords.
4 095 bytes of D11.5, 4 095 bytes of D20.2.
1 D11.7 transitional byte including 000 0010b run,
1 D20.7 transitional byte including 111 1101b run.

Short version total: 2 + (2 × 63) = 128 Dwords
255 bytes of D11.5, 255 bytes of D20.2
1 D11.7 transitional byte including 000 0010b run,
1 D20.7 transitional byte including 111 1101b run

Table 66 – Low Frequency Spectral Content Pattern (LFSCP) starting with RD+

Transmission Order →

+	D11.5(ABh)+				D11.5(ABh)+				D11.5(ABh)+				D11.5(ABh)+		+
	1101b	0010b	1011b	0100b	1010b	1101b	0010b	1011b	0100b	1010b	
	Dh	2h	Bh	4h	Ah	Dh	2h	Bh	4h	Ah	

Above Dword is repeated a total of 1 023 times.
Above Dword is repeated a total of 63 times for short version.

+	D11.5(ABh)+				D11.7(EBh)+				D20.2.(54h)-				D20.2(54h)-		-
	1101b	0010b	1011b	0100b	1000b	0010b	1101b	0100b	1011b	0101b	
	Dh	2h	Bh	4h	8h	2h	Dh	4h	Bh	5h	

-	D20.2(54h)-				D20.2(54h)-				D20.2(54h)-				D20.2(54h)-		-
	0010b	1101b	0100b	1011b	0101b	0010b	1101b	0100b	1011b	0101b	
	2h	Dh	4h	Bh	5h	2h	Dh	4h	Bh	5h	

Above Dword is repeated a total of 1 023 times.
Above Dword is repeated a total of 63 times for short version.

-	D20.2(54h)-				D20.7(F4h)-				D11.5(ABh)+				D11.5(ABh)+		+
	0010b	1101b	0100b	1011b	0111b	1101b	0010b	1011b	0100b	1010b	
	2h	Dh	4h	Bh	7h	Dh	2h	Bh	4h	Ah	

Long version total: 2 + (2 × 1 023) = 2 048 Dwords
4 095 bytes of D11.5, 4 095 bytes of D20.2
1 D11.7 transitional byte including 000 0010b run,
1 D20.7 transitional byte including 111 1101b run

Short version total: 2 + (2 × 63) = 128 Dwords
255 bytes of D11.5, 255 bytes of D20.2

7.4.5.4.5 Simultaneous switching outputs pattern (SSOP)

The simultaneous switching outputs bit pattern (SSOP), shown in Table 67 and Table 68, induces inductive switching (Ldi/dt) noise into substrate supply of a receiver providing a good test of noise control. The SSOP pattern, alternating 1's complement bit patterns (10 bits), are applied to a receiver. Bit sequences are shown for both cases, where the starting running disparity is negative (RD-) or positive (RD+).

Table 67 – Simultaneous Switching Outputs Pattern (SSOP) starting with RD-

	Transmission Order →										
	D31.3(7Fh)-			D31.3(7Fh)+		D31.3(7Fh)-			D31.3(7Fh)+		
-	1010b	1100b	1101b	0100b	1100b	1010b	1100b	1101b	0100b	1100b	-
	Ah	Ch	Dh	4h	Ch	Ah	Ch	Dh	4h	Ch	
	Above Dword is repeated a total of 2 048 times for long version. Above Dword is repeated a total of 128 times for short version.										
Long version total 1 × 2 048 = 2 048 Dwords. Short version total 1 × 128 = 128 Dwords.											

Table 68 – Simultaneous Switching Outputs Pattern (SSOP) starting with RD+

	Transmission Order →										
	D31.3(7Fh)+			D31.3(7Fh)-		D31.3(7Fh)+			D31.3(7Fh)-		
+	0101b	0011b	0010b	1011b	0011b	0101b	0011b	0010b	1011b	0011b	+
	5h	3h	2h	Bh	3h	5h	3h	2h	Bh	3h	
	Above Dword is repeated a total of 2 048 times for long version. Above Dword is repeated a total of 128 times for short version.										
Long version total 1 × 2048 = 2 048 Dwords. Short version total 1 × 128 = 128 Dwords.											

7.4.5.4.6 Lone Bit Pattern (LBP)

The Lone Bit Patterns, shown in Table 69 and Table 70, are comprised of the combination of adjacent 8b/10b encoded patterns, resulting in a lone one bit prefixed by a run length of four zeros, and suffixed by a run length of three zeros. It also results in a lone zero bit prefixed by a run length of two ones, one zero, two ones, one zero, four ones, and suffixed by a single one. This is a good test of the receiver jitter tolerance under adverse signaling conditions. The lone bit may be attenuated and narrower than expected. Bit sequences are shown for both cases, where the starting running disparity is negative (RD-) or positive (RD+).

Table 69 – Lone Bit Pattern (LBP) starting with RD-

	Transmission Order →										
-	D12.0(0Ch)-		D11.4(8Bh)+			D12.0(0Ch)-		D11.3(6Bh)+			+
	0011b	0110b	1111b	0100b	0010b	0011b	0110b	1111b	0100b	0011b	
	3h	6h	Fh	4h	2h	3h	6h	Fh	4h	3h	
+	D12.0(0Ch)+		D11.4(8Bh)-			D12.0(0Ch)+		D11.3(6Bh)-			-
	0011b	0101b	0011b	0100b	1101b	0011b	0101b	0011b	0100b	1100b	
	3h	5h	3h	4h	Dh	3h	5h	3h	4h	Ch	

Long version total 2 × 1 024 = 2 048 Dwords.
Short version total 2 × 64 = 128 Dwords.

Table 70 – Lone Bit Pattern (LBP) starting with RD+

	Transmission Order →										
+	D12.0(0Ch)+		D11.4(8Bh)-			D12.0(0Ch)+		D11.3(6Bh)-			-
	0011b	0101b	0011b	0100b	1101b	0011b	0101b	0011b	0100b	1100b	
	3h	5h	3h	4h	Dh	3h	5h	3h	4h	Ch	
-	D12.0(0Ch)-		D11.4(8Bh)+			D12.0(0Ch)-		D11.3(6Bh)+			+
	0011b	0110b	1111b	0100b	0010b	0011b	0110b	1111b	0100b	0011b	
	3h	6h	Fh	4h	2h	3h	6h	Fh	4h	3h	

Long version total 2 × 1 024 = 2 048 Dwords.
Short version total 2 × 64 = 128 Dwords.

7.4.5.4.7 Composite pattern (COMP)

For the measurement of jitter, the composite patterns (COMP), as shown in Table 71 and Table 72**,** should combine low frequency, low transition density, and high transition density patterns. All these combinations, but the low frequency spectral content class may be performed for relatively short test time intervals, for good jitter performance measurements.

The lower frequency pattern needs to be tested for longer interval periods to be able to observe the lower frequency jitter effects on the interface.

The composite pattern (COMP) stresses the interface components within the link with low and high frequency jitter, tests for component, and various amplitude distortions due to marginal receiver input circuitry, or interface components.

NOTE 22 - Note that the cited patterns appear on the wire.

NOTE 23 - Note that for the sequence that totals only 128 Dwords, the 128 Dword composite pattern is too short to get a sufficient number of continuous repeats for each pattern type.

Table 71 – Composite-Bit Pattern (COMP) starting with RD- (part 1 of 3)

Transmission Order →

	D31.3(7Fh)-			D31.3(7Fh)+		D31.3(7Fh)-			D31.3(7Fh)+		
-	1010b	1100b	1101b	0100b	1100b	1010b	1100b	1101b	0100b	1100b	-
	Ah	Ch	Dh	4h	Ch	Ah	Ch	Dh	4h	Ch	

Above Dword is repeated a total of 256 times for long version.
Above Dword is repeated a total of 16 times for short version.

	D21.5(B5h)-			D21.5(B5h)-		D21.5(B5h)-			D21.5(B5h)-		
-	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	-
	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	

Above Dword is repeated a total of 64 times for long version.
Above Dword is repeated a total of 4 times for short version.

	D24.3(78h)-			D24.3(78h)+		D24.3(78h)-			D24.3(78h)+		
-	1100b	1100b	1100b	1100b	1100b	1100b	1100b	1100b	1100b	1100b	-
	Ch	Ch	Ch	Ch	Ch	Ch	Ch	Ch	Ch	Ch	

Above Dword is repeated a total of 64 times for long version.
Above Dword is repeated a total of 4 times for short version.

	D10.2(4Ah)-			D10.2(4Ah)-		D10.2(4Ah)-			D10.2(4Ah)-		
-	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	-
	5h	5h	5h	5h	5h	5h	5h	5h	5h	5h	

Above Dword is repeated a total of 64 times for long version.
Above Dword is repeated a total of 4 times for short version.

	D25.6(D9h)-			D6.1(26h)+		D25.6(D9h)-			D6.1(26h)+		
-	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	-
	9h	9h	9h	9h	9h	9h	9h	9h	9h	9h	

Above Dword is repeated a total of 64 times for long version.
Above Dword is repeated a total of 4 times for short version.

	D17.7(F1h)-			D30.7(FEh)+		D7.1(27h)+			D14.7(EEh)+		
-	1000b	1101b	1110b	0001b	1110b	0001b	1110b	0101b	1100b	1000b	-
	8h	Dh	Eh	1h	Eh	1h	Eh	5h	Ch	8h	

	D30.7(FEh)-			D7.6(C7h)-		D30.3(7Eh)-			D30.3(7Eh)+		
-	0111b	1000b	0111b	1000b	0110b	0111b	1000b	1110b	0001b	1100b	-
	7h	8h	7h	8h	6h	7h	8h	Eh	1h	Ch	

Table 71 – Composite-Bit Pattern (COMP) starting with RD- (part 2 of 3)

Transmission Order →

	D30.3(7Eh)-			D30.3(7Eh)+		D30.3(7Eh)-			D30.3(7Eh)+		
-	0111b	1000b	1110b	0001b	1100b	0111b	1000b	1110b	0001b	1100b	-
	7h	8h	Eh	1h	Ch	7h	8h	Eh	1h	Ch	

Above Dword is repeated a total of 509 times for long version.
Above Dword is repeated a total of 29 times for short version.

	D3.7(E3h)-			D28.7(FCh)+		D3.7(E3h)-			D28.7(FCh)+		
-	1100b	0111b	1000b	1110b	0001b	1100b	0111b	1000b	1110b	0001b	-
	Ch	7h	8h	Eh	1h	Ch	7h	8h	Eh	1h	

	D12.0(0Ch)-			D11.4(8Bh)+		D12.0(0Ch)-			D11.3(6Bh)+		
-	0011b	0110b	1111b	0100b	0010b	0011b	0110b	1111b	0100b	0011b	+
	3h	6h	Fh	4h	2h	3h	6h	Fh	4h	3h	

	D12.0(0Ch)+			D11.4(8Bh)-		D12.0(0Ch)+			D11.3(6Bh)-		
+	0011b	0101b	0011b	0100b	1101b	0011b	0101b	0011b	0100b	1100b	-
	3h	5h	3h	4h	Dh	3h	5h	3h	4h	Ch	

Above 2 Dwords are repeated a total of 128 times for long version.
Above 2 Dwords are repeated a total of 8 times for short version.

	D20.2(54h)-			D20.2(54h)-		D20.2(54h)-			D20.2(54h)-		
-	0010b	1101b	0100b	1011b	0101b	0010b	1101b	0100b	1011b	0101b	-
	2h	Dh	4h	Bh	5h	2h	Dh	4h	Bh	5h	

Above Dword is repeated a total of 255 times for long version.
Above Dword is repeated a total of 15 times for short version.

	D20.2(54h)-			D20.7(F4h)-		D11.5(ABh)+			D11.5(ABh)+		
-	0010b	1101b	0100b	1011b	0111b	1101b	0010b	1011b	0100b	1010b	+
	2h	Dh	4h	Bh	7h	Dh	2h	Bh	4h	Ah	

	D11.5(ABh)+			D11.5(ABh)+		D11.5(ABh)+			D11.5(ABh)+		
+	1101b	0010b	1011b	0100b	1010b	1101b	0010b	1011b	0100b	1010b	+
	Dh	2h	Bh	4h	Ah	Dh	2h	Bh	4h	Ah	

Above Dword is repeated a total of 255 times for long version.
Above Dword is repeated a total of 15 times for short version.

	D11.5(ABh)+			D11.7(EBh)+		D20.2.(54h)-			D20.2.(54h)-		
+	1101b	0010b	1011b	0100b	1000b	0010b	1101b	0100b	1011b	0101b	-
	Dh	2h	Bh	4h	8h	2h	Dh	4h	Bh	5h	

Table 71 **– Composite-Bit Pattern (COMP) starting with RD-** (part 3 of 3)

Transmission Order →

-	D21.5(B5h)-				D21.5(B5h)-				D21.5(B5h)-				D21.5(B5h)-		-
	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	
	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	
	Above Dword is repeated a total of 64 times for long version. Above Dword is repeated a total of 4 times for short version.										
-	D24.3(78h)-				D24.3(78h)-				D24.3(78h)-				D24.3(78h)-		-
	1100b	1100b	1100b	1100b	1100b	1100b	1100b	1100b	1100b	1100b	
	Ch	Ch	Ch	Ch	Ch	Ch	Ch	Ch	Ch	Ch	
	Above Dword is repeated a total of 64 times for long version. Above Dword is repeated a total of 4 times for short version.										
-	D10.2(4Ah)-				D10.2(4Ah)-				D10.2(4Ah)-				D10.2(4Ah)-		-
	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	
	5h	5h	5h	5h	5h	5h	5h	5h	5h	5h	
	Above Dword is repeated a total of 64 times for long version. Above Dword is repeated a total of 4 times for short version.										
-	D25.6(D9h)-				D6.1(26h)+				D25.6(D9h)-				D6.1(26h)+		-
	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	
	9h	9h	9h	9h	9h	9h	9h	9h	9h	9h	
	Above Dword is repeated a total of 64 times for long version. Above Dword is repeated a total of 4 times for short version.										

Long version total 2 048 Dwords total
256 DW SSOP
256 DW HTDP (64 DW, 64 DW, 64 DW, 64 DW)
512 DW LTDP (1 DW, 1 DW, 509 DW, 1 DW)
256 DW LBP ((1 DW,1 DW) × 128)
512 DW LFSCP (255 DW, 1 DW, 255 DW, 1 DW)
256 DW HTDP (64 DW, 64 DW, 64 DW, 64 DW)

Short version total 128 Dwords total
16 DW SSOP
16 DW HTDP (4 DW, 4 DW, 4 DW, 4 DW)
32 DW LTDP (1 DW, 1 DW, 29 DW, 1 DW)
16 DW LBP ((1 DW,1 DW) × 8)
32 DW LFSCP (15 DW, 1 DW, 15 DW, 1 DW)
16 DW HTDP (4 DW, 4 DW, 4 DW, 4 DW)

Table 72 – Composite-Bit Pattern (COMP) starting with RD+ (part 1 of 3)

Transmission Order →

	D31.3(7Fh)+			D31.3(7Fh)-		D31.3(7Fh)+			D31.3(7Fh)-		
+	0101b	0011b	0010b	1011b	0011b	0101b	0011b	0010b	1011b	0011b	+
	5h	3h	2h	Bh	3h	5h	3h	2h	Bh	3h	
	Above Dword is repeated a total of 256 times for long version. Above Dword is repeated a total of 16 times for short version.										
	D21.5(B5h)+			D21.5(B5h)+		D21.5(B5h)+			D21.5(B5h)+		
+	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	+
	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	
	Above Dword is repeated a total of 64 times for long version. Above Dword is repeated a total of 4 times for short version.										
	D24.3(78h)+			D24.3(78h)+		D24.3(78h)+			D24.3(78h)+		
+	0011b	0011b	0011b	0011b	0011b	0011b	0011b	0011b	0011b	0011b	+
	3h	3h	3h	3h	3h	3h	3h	3h	3h	3h	
	Above Dword is repeated a total of 64 times for long version. Above Dword is repeated a total of 4 times for short version.										
	D10.2(4Ah)+			D10.2(4Ah)+		D10.2(4Ah)+			D10.2(4Ah)+		
+	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	+
	5h	5h	5h	5h	5h	5h	5h	5h	5h	5h	
	Above Dword is repeated a total of 64 times for long version. Above Dword is repeated a total of 4 times for short version.										
	D25.6(D9h)+			D6.1(26h)+		D25.6(D9h)+			D6.1(26h)+		
+	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	+
	9h	9h	9h	9h	9h	9h	9h	9h	9h	9h	
	Above Dword is repeated a total of 64 times for long version. Above Dword is repeated a total of 4 times for short version.										
	D14.7(EEh)+			D30.7(FEh)-		D7.6(C7h)+			D17.7(F1h)-		
+	0111b	0010b	0001b	1110b	0001b	1110b	0001b	1010b	0011b	0111b	+
	7h	2h	1h	Eh	1h	Eh	1h	Ah	3h	7h	
	D30.7(FEh)+			D7.1(27h)+		D30.3(7Eh)+			D30.3(7Eh)-		
+	1000b	0111b	1000b	0111b	1001b	1000b	0111b	0001b	1110b	0011b	+
	8h	7h	8h	7h	9h	8h	7h	1h	Eh	3h	
	D30.3(7Eh)+			D30.3(7Eh)-		D30.3(7Eh)+			D30.3(7Eh)-		
+	1000b	0111b	0001b	1110b	0011b	1000b	0111b	0001b	1110b	0011b	+
	8h	7h	1h	Eh	3h	8h	7h	1h	Eh	3h	
	Above Dword is repeated a total of 509 times for long version. Above Dword is repeated a total of 29 times for short version.										

Table 72 – Composite-Bit Pattern (COMP) starting with RD+ (part 2 of 3)

Transmission Order →

	D28.7(FCh)+			D3.7(E3h)-		D28.7(FCh)+			D3.7(E3h)-		
+	0011b	1000b	0111b	0001b	1110b	0011b	1000b	0111b	0001b	1110b	+
	3h	8h	7h	1h	Eh	3h	8h	7h	1h	Eh	
	D12.0(0Ch)+			D11.4(8Bh)-		D12.0(0Ch)+			D11.3(6Bh)-		
+	0011b	0100b	0011b	0100b	1101b	0011b	0101b	0011b	0101b	1100b	-
	3h	5h	3h	4h	Dh	3h	5h	3h	4h	Ch	
	D12.0(0Ch)+			D11.4(8Bh)-		D12.0(0Ch)+			D11.3(6Bh)-		
-	0011b	0110b	1111b	0100b	0010b	0011b	0110b	1111b	0100b	0011b	+
	3h	6h	Fh	4h	2h	3h	6h	Fh	4h	3h	

Above 2 Dwords are repeated a total of 128 times for long version.
Above 2 Dwords are repeated a total of 8 times for short version.

	D11.5(ABh)+			D11.5(ABh)+		D11.5(ABh)+			D11.5(ABh)+		
+	1101b	0010b	1011b	0100b	1010b	1101b	0010b	1011b	0100b	1010b	+
	Dh	2h	Bh	4h	Ah	Dh	2h	Bh	4h	Ah	

Above Dword is repeated a total of 255 times for long version.
Above Dword is repeated a total of 15 times for short version.

	D11.5(ABh)+			D11.7(EBh)+		D20.2.(54h)-			D20.2(54h)-		
+	1101b	0010b	1011b	0100b	1000b	0010b	1101b	0100b	1011b	0101b	-
	Dh	2h	Bh	4h	8h	2h	Dh	4h	Bh	5h	
	D20.2(54h)-			D20.2(54h)-		D20.2(54h)-			D20.2(54h)-		
-	0010b	1101b	0100b	1011b	0101b	0010b	1101b	0100b	1011b	0101b	-
	2h	Dh	4h	Bh	5h	2h	Dh	4h	Bh	5h	

Above Dword is repeated a total of 255 times for long version.
Above Dword is repeated a total of 15 times for short version.

	D20.2(54h)-			D20.7(F4h)-		D11.5(ABh)+			D11.5(ABh)+		
-	0010b	1101b	0100b	1011b	0111b	1101b	0010b	1011b	0100b	1010b	+
	2h	Dh	4h	Bh	7h	Dh	2h	Bh	4h	Ah	
	D21.5(B5h)+			D21.5(B5h)+		D21.5(B5h)+			D21.5(B5h)+		
+	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	+
	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	

Above Dword is repeated a total of 64 times for long version.
Above Dword is repeated a total of 4 times for short version.

Table 72 – Composite-Bit Pattern (COMP) starting with RD+ (part 3 of 3)

Transmission Order →

+	D24.3(78h)+				D24.3(78h)+				D24.3(78h)+		+
	0011b	0011b	0011b	0011b	0011b	0011b	0011b	0011b	0011b	0011b	
	3h	3h	3h	3h	3h	3h	3h	3h	3h	3h	
	Above Dword is repeated a total of 64 times for long version. Above Dword is repeated a total of 4 times for short version.										
+	D10.2(4Ah)+				D10.2(4Ah)+				D10.2(4Ah)+		+
	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	
	5h	5h	5h	5h	5h	5h	5h	5h	5h	5h	
	Above Dword is repeated a total of 64 times for long version. Above Dword is repeated a total of 4 times for short version.										
+	D25.6(D9h)+				D6.1(26h)+				D25.6(D9h)+		+
	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	
	9h	9h	9h	9h	9h	9h	9h	9h	9h	9h	
	Above Dword is repeated a total of 64 times for long version. Above Dword is repeated a total of 4 times for short version.										

Long version total 2 048 Dwords total
256 DW SSOP
256 DW HTDP (64 DW, 64 DW, 64 DW, 64 DW)
512 DW LTDP (1 DW, 1 DW, 509 DW, 1 DW)
256 DW LBP ((1 DW,1 DW) × 128)
512 DW LFSCP (255 DW, 1 DW, 255 DW, 1 DW)
256 DW HTDP (64 DW, 64 DW, 64 DW, 64 DW)

Short version total 128 Dwords total
16 DW SSOP
16 DW HTDP (4 DW, 4 DW, 4 DW, 4 DW)
32 DW LTDP (1 DW, 1 DW, 29 DW, 1 DW)
16 DW LBP ((1 DW,1 DW) × 8)
32 DW LFSCP (15 DW, 1 DW, 15 DW, 1 DW)
16 DW HTDP (4 DW, 4 DW, 4 DW, 4 DW)

NOTE 24 - Note that only 128 Dwords total for the composite pattern is too short to get a sufficient number of continuous repeats for each pattern type.

7.4.5.4.8 Framed composite pattern (FCOMP)

The Framed Composite Pattern (FCOMP) (see Table 73) is equivalent to the COMP pattern as defined in 7.4.5.4.7 with the following structured changes:

a) according to 7.4.5.3 the COMP pattern is framed;
b) 2 $ALIGN_P$ primitives inserted every 256 Dwords; and
c) a short Inter Gap region is introduced before and after the SOF_P/EOF_P to ensure that if repeated sequentially by a generator the 256 Dword $ALIGN_P$ primitives are perfectly and uniformly spaced 256 Dwords apart even after wrap-around by the generator.

Table 73 – Framed Composite Pattern (FCOMP) (part 1 of 7)

Transmission Order →

	K28.5(BCh)+		D10.2(4Ah)-			D10.2(4Ah)-			D27.3(7Bh)-		
+	1100b	0001b	0101b	0101b	0101b	0101b	0101b	0111b	0110b	0011b	+
	Ch	1h	5h	5h	5h	5h	5h	7h	6h	3h	

Above Dword is repeated a total of 2 times.
2 DW ALIGN$_P$.

	D10.2(4Ah)+		D10.2(4Ah)+			D10.2(4Ah)+			D10.2(4Ah)+		
+	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	+
	5h	5h	5h	5h	5h	5h	5h	5h	5h	5h	

Above Dword is repeated a total of 7 times.
7 DW HFTP

	K28.3(7Ch)+		D21.5(B5h)-			D23.2(57h)-			D23.2(57h)+		
+	1100b	0011b	0010b	1010b	1010b	1110b	1001b	0100b	0101b	0101b	-
	Ch	3h	2h	Ah	Ah	Eh	9h	4h	5h	5h	

X_RDY$_P$ with RD+

	K28.3(7Ch)-		D21.5(B5h)+			D23.2(57h)+			D23.2(57h)-		
-	0011b	1100b	1110b	1010b	1010b	0001b	0101b	0111b	1010b	0101b	+
	3h	Ch	Eh	Ah	Ah	1h	5h	7h	Ah	5h	

Above 2 Dword is repeated a total of 2 times.
X_RDY$_P$ with RD-

	K28.3(7Ch)+		D21.5(B5h)-			D23.2(57h)-			D23.2(57h)+		
+	1100b	0011b	0010b	1010b	1010b	1110b	1001b	0100b	0101b	0101b	-
	Ch	3h	2h	Ah	Ah	Eh	9h	4h	5h	5h	

X_RDY$_P$ with RD+
All together 5 DW X_RDY$_P$

	K28.3(7Ch)-		D21.5(B5h)+			D23.1(37h)+			D23.1(37h)-		
-	0011b	1100b	1110b	1010b	1010b	0001b	0110b	0111b	1010b	1001b	+
	3h	Ch	Eh	Ah	Ah	1h	6h	7h	Ah	9h	

SOF$_P$

	D11.6(CBh)+		D22.3(76h)+			D18.6(D2h)+			D3.0(C2h)+		
+	1101b	0001b	1001b	1010b	0011b	0100b	1101b	1001b	0010b	0110b	-
	Dh	1h	9h	Ah	3h	4h	Dh	9h	2h	6h	

Data FIS Header

	D31.3(7Fh)-		D31.3(7Fh)+			D31.3(7Fh)-			D31.3(7Fh)+		
-	1010b	1100b	1101b	0100b	1100b	1010b	1100b	1101b	0100b	1100b	-
	Ah	Ch	Dh	4h	Ch	Ah	Ch	Dh	4h	Ch	

Above Dword is repeated a total of 240 times.
SSOP

Table 73 – Framed Composite Pattern (FCOMP) (part 2 of 7)

Transmission Order →

-	K28.5(BCh)-		D10.2(4Ah)+			D10.2(4Ah)+			D27.3(7Bh)+		-
	0011b	1110b	1001b	0101b	0101b	0101b	0101b	0100b	1001b	1100b	
	3h	Eh	9h	5h	5h	5h	5h	4h	9h	Ch	
	Above Dword is repeated a total of 2 times. ALIGNP										
-	D31.3(7Fh)-		D31.3(7Fh)+			D31.3(7Fh)-			D31.3(7Fh)+		-
	1010b	1100b	1101b	0100b	1100b	1010b	1100b	1101b	0100b	1100b	
	Ah	Ch	Dh	4h	Ch	Ah	Ch	Dh	4h	Ch	
	Above Dword is repeated a total of 16 times. A total of 256 DW SSOP.										
-	D21.5(B5h)-		D21.5(B5h)-			D21.5(B5h)-			D21.5(B5h)-		-
	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	
	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	
	Above Dword is repeated a total of 64 times. HTDP										
-	D24.3(78h)-		D24.3(78h)+			D24.3(78h)-			D24.3(78h)+		-
	1100b	1100b	1100b	1100b	1100b	1100b	1100b	1100b	1100b	1100b	
	Ch	Ch	Ch	Ch	Ch	Ch	Ch	Ch	Ch	Ch	
	Above Dword is repeated a total of 64 times. HTDP										
-	D10.2(4Ah)-		D10.2(4Ah)-			D10.2(4Ah)-			D10.2(4Ah)-		-
	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	
	5h	5h	5h	5h	5h	5h	5h	5h	5h	5h	
	Above Dword is repeated a total of 64 times. HTDP										
-	D25.6(D9h)-		D6.1(26h)-			D25.6(D9h)-			D6.1(26h)-		-
	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	
	9h	9h	9h	9h	9h	9h	9h	9h	9h	9h	
	Above Dword is repeated a total of 46 times. HTDP										
-	K28.5(BCh)-		D10.2(4Ah)+			D10.2(4Ah)+			D27.3(7Bh)+		-
	0011b	1110b	1001b	0101b	0101b	0101b	0101b	0100b	1001b	1100b	
	3h	Eh	9h	5h	5h	5h	5h	4h	9h	Ch	
	Above Dword is repeated a total of 2 times. ALIGNP										
-	D25.6(D9h)-		D6.1(26h)-			D25.6(D9h)-			D6.1(26h)-		-
	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	
	9h	9h	9h	9h	9h	9h	9h	9h	9h	9h	
	Above Dword is repeated a total of 18 times. 256 DW HTDP (64 DW, 64 DW, 64 DW, 64 DW)										

Table 73 – Framed Composite Pattern (FCOMP) (part 3 of 7)

Transmission Order →

-	D17.7(F1h)-		D30.7(FEh)+			D7.1(27h)+		D14.7(EEh)+			-
	1000b	1101b	1110b	0001b	1110b	0001b	1110b	0101b	1100b	1000b	
	8h	Dh	Eh	1h	Eh	1h	Eh	5h	Ch	8h	
	LTDP										
-	D30.7(FEh)-		D7.6(C7h)-			D30.3(7Eh)-		D30.3(7Eh)+			-
	0111b	1000b	0111b	1000b	0110b	0111b	1000b	1110b	0001b	1100b	
	7h	8h	7h	8h	6h	7h	8h	Eh	1h	Ch	
	LTDP										
-	D30.3(7Eh)-		D30.3(7Eh)+			D30.3(7Eh)-		D30.3(7Eh)+			-
	0111b	1000b	1110b	0001b	1100b	0111b	1000b	1110b	0001b	1100b	
	7h	8h	Eh	1h	Ch	7h	8h	Eh	1h	Ch	
	Above Dword is repeated a total of 234 times. LTDP										
-	K28.5(BCh)-		D10.2(4Ah)+			D10.2(4Ah)+		D27.3(7Bh)+			-
	0011b	1110b	1001b	0101b	0101b	0101b	0101b	0100b	1001b	1100b	
	3h	Eh	9h	5h	5h	5h	5h	4h	9h	Ch	
	Above Dword is repeated a total of 2 times. $ALIGN_P$										
-	D30.3(7Eh)-		D30.3(7Eh)+			D30.3(7Eh)-		D30.3(7Eh)+			-
	0111b	1000b	1110b	0001b	1100b	0111b	1000b	1110b	0001b	1100b	
	7h	8h	Eh	1h	Ch	7h	8h	Eh	1h	Ch	
	Above Dword is repeated a total of 254 times. LTDP										
-	K28.5(BCh)-		D10.2(4Ah)+			D10.2(4Ah)+		D27.3(7Bh)+			-
	0011b	1110b	1001b	0101b	0101b	0101b	0101b	0100b	1001b	1100b	
	3h	Eh	9h	5h	5h	5h	5h	4h	9h	Ch	
	Above Dword is repeated a total of 2 times. $ALIGN_P$										
-	D30.3(7Eh)-		D30.3(7Eh)+			D30.3(7Eh)-		D30.3(7Eh)+			-
	0111b	1000b	1110b	0001b	1100b	0111b	1000b	1110b	0001b	1100b	
	7h	8h	Eh	1h	Ch	7h	8h	Eh	1h	Ch	
	Above Dword is repeated a total of 21 times. LTDP										
-	D3.7(E3h)-		D28.7(FCh)+			D3.7(E3h)-		D28.7(FCh)+			-
	1100b	0111b	1000b	1110b	0001b	1100b	0111b	1000b	1110b	0001b	
	Ch	7h	8h	Eh	1h	Ch	7h	8h	Eh	1h	
	512 DW LTDP (1 DW, 1 DW, 509 DW, 1 DW)										

Table 73 – Framed Composite Pattern (FCOMP) (part 4 of 7)

Transmission Order →

-	D12.0(0Ch)-		D11.4(8Bh)+			D12.0(0Ch)-		D11.3(6Bh)+			+
	0011b	0110b	1111b	0100b	0010b	0011b	0110b	1111b	0100b	0011b	
	3h	6h	Fh	4h	2h	3h	6h	Fh	4h	3h	
	LBP										
+	D12.0(0Ch)+		D11.4(8Bh)-			D12.0(0Ch)+		D11.3(6Bh)-			-
	0011b	0101b	0011b	0100b	1101b	0011b	0101b	0011b	0100b	1100b	
	3h	5h	3h	4h	Dh	3h	5h	3h	4h	Ch	
	Above 2 Dwords are repeated a total of 116 times.										
	LBP										
-	K28.5(BCh)-		D10.2(4Ah)+			D10.2(4Ah)+		D27.3(7Bh)+			-
	0011b	1110b	1001b	0101b	0101b	0101b	0101b	0100b	1001b	1100b	
	3h	Eh	9h	5h	5h	5h	5h	4h	9h	Ch	
	Above Dword is repeated a total of 2 times.										
	$ALIGN_P$										
-	D12.0(0Ch)-		D11.4(8Bh)+			D12.0(0Ch)-		D11.3(6Bh)+			+
	0011b	0110b	1111b	0100b	0010b	0011b	0110b	1111b	0100b	0011b	
	3h	6h	Fh	4h	2h	3h	6h	Fh	4h	3h	
	LBP										
+	D12.0(0Ch)+		D11.4(8Bh)-			D12.0(0Ch)+		D11.3(6Bh)-			-
	0011b	0101b	0011b	0100b	1101b	0011b	0101b	0011b	0100b	1100b	
	3h	5h	3h	4h	Dh	3h	5h	3h	4h	Ch	
	Above 2 Dwords are repeated a total of 12 times.										
	256 DW LBP ((1 DW,1 DW) × 128)										
-	D20.2(54h)-		D20.2(54h)-			D20.2(54h)-		D20.2(54h)-			-
	0010b	1101b	0100b	1011b	0101b	0010b	1101b	0100b	1011b	0101b	
	2h	Dh	4h	Bh	5h	2h	Dh	4h	Bh	5h	
	Above Dword is repeated a total of 230 times.										
	LFSCP										
-	K28.5(BCh)-		D10.2(4Ah)+			D10.2(4Ah)+		D27.3(7Bh)+			-
	0011b	1110b	1001b	0101b	0101b	0101b	0101b	0100b	1001b	1100b	
	3h	Eh	9h	5h	5h	5h	5h	4h	9h	Ch	
	Above Dword is repeated a total of 2 times.										
	$ALIGN_P$										
-	D20.2(54h)-		D20.2(54h)-			D20.2(54h)-		D20.2(54h)-			-
	0010b	1101b	0100b	1011b	0101b	0010b	1101b	0100b	1011b	0101b	
	2h	Dh	4h	Bh	5h	2h	Dh	4h	Bh	5h	
	Above Dword is repeated a total of 25 times.										
	LFSCP										

Table 73 – Framed Composite Pattern (FCOMP) (part 5 of 7)

Transmission Order →

-	D20.2(54h)-		D20.7(F4h)-			D11.5(ABh)+		D11.5(ABh)+			+
	0010b	1101b	0100b	1011b	0111b	1101b	0010b	1011b	0100b	1010b	
	2h	Dh	4h	Bh	7h	Dh	2h	Bh	4h	Ah	
	LFSCP										
+	D11.5(ABh)+		D11.5(ABh)+			D11.5(ABh)+		D11.5(ABh)+			+
	1101b	0010b	1011b	0100b	1010b	1101b	0010b	1011b	0100b	1010b	
	Dh	2h	Bh	4h	Ah	Dh	2h	Bh	4h	Ah	
	Above Dword is repeated a total of 228 times. LFSCP										
+	K28.5(BCh)+		D10.2(4Ah)-			D10.2(4Ah)-		D27.3(7Bh)-			+
	1100b	0001b	0101b	0101b	0101b	0101b	0101b	0111b	0110b	0011b	
	Ch	1h	5h	5h	5h	5h	5h	7h	6h	3h	
	Above Dword is repeated a total of 2 times. ALIGN$_P$										
+	D11.5(ABh)+		D11.5(ABh)+			D11.5(ABh)+		D11.5(ABh)+			+
	1101b	0010b	1011b	0100b	1010b	1101b	0010b	1011b	0100b	1010b	
	Dh	2h	Bh	4h	Ah	Dh	2h	Bh	4h	Ah	
	Above Dword is repeated a total of 27 times. LFSCP										
+	D11.5(ABh)+		D11.7(EBh)+			D20.2.(54h)-			D20.2.(54h)-		-
	1101b	0010b	1011b	0100b	1000b	0010b	1101b	0100b	1011b	0101b	
	Dh	2h	Bh	4h	8h	2h	Dh	4h	Bh	5h	
	512 DW LFSCP (255 DW, 1 DW, 255 DW, 1 DW)										
-	D21.5(B5h)-		D21.5(B5h)-			D21.5(B5h)-			D21.5(B5h)-		-
	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	1010b	
	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	Ah	
	Above Dword is repeated a total of 64 times. HTDP										
-	D24.3(78h)-		D24.3(78h)+			D24.3(78h)-			D24.3(78h)+		-
	1100b	1100b	1100b	1100b	1100b	1100b	1100b	1100b	1100b	1100b	
	Ch	Ch	Ch	Ch	Ch	Ch	Ch	Ch	Ch	Ch	
	Above Dword is repeated a total of 64 times. HTDP										
-	D10.2(4Ah)-		D10.2(4Ah)-			D10.2(4Ah)-			D10.2(4Ah)-		-
	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	
	5h	5h	5h	5h	5h	5h	5h	5h	5h	5h	
	Above Dword is repeated a total of 64 times. HTDP										

Table 73 – Framed Composite Pattern (FCOMP) (part 6 of 7)

Transmission Order →

-	D25.6(D9h)-		D6.1(26h)+			D25.6(D9h)-			D6.1(26h)+		-
	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	
	9h	9h	9h	9h	9h	9h	9h	9h	9h	9h	

Above Dword is repeated a total of 34 times.

HTDP

-	K28.5(BCh)-		D10.2(4Ah)+			D10.2(4Ah)+			D27.3(7Bh)+		-
	0011b	1110b	1001b	0101b	0101b	0101b	0101b	0100b	1001b	1100b	
	3h	Eh	9h	5h	5h	5h	5h	4h	9h	Ch	

Above Dword is repeated a total of 2 times.

ALIGN$_P$

-	D25.6(D9h)-		D6.1(26h)+			D25.6(D9h)-			D6.1(26h)+		-
	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	1001b	
	9h	9h	9h	9h	9h	9h	9h	9h	9h	9h	

Above Dword is repeated a total of 30 times.

HTDP

-	D11.6(CBh)-		D18.6(D2h)-			D29.6(DDh)-			D6.4(86h)+		-
	1101b	0001b	1001b	0011b	0110b	1011b	1001b	1001b	1001b	0010b	
	Dh	1h	9h	3h	6h	Bh	9h	9h	9h	2h	

CRC

-	K28.3(7Ch)-		D21.5(B5h)+			D21.6(D5h)+			D21.6(D5h)+		+
	0011b	1100b	1110b	1010b	1010b	1010b	1001b	1010b	1010b	0110b	
	3h	Ch	Eh	Ah	Ah	Ah	9h	Ah	Ah	6h	

EOF$_P$

+	K28.3(7Ch)+		D21.5(B5h)-			D24.2(58h)-			D24.2(58h)+		-
	1100b	0011b	0010b	1010b	1010b	1100b	1101b	0100b	1100b	0101b	
	Ch	3h	2h	Ah	Ah	Ch	Dh	4h	Ch	5h	

WTRM$_P$

-	K28.3(7Ch)-		D21.5(B5h)+			D24.2(58h)+			D24.2(58h)-		+
	0011b	1100b	1110b	1010b	1010b	0011b	0001b	0111b	0011b	0101b	
	3h	Ch	Eh	Ah	Ah	3h	1h	7h	3h	5h	

Above 2 Dwords are repeated a total of 2 times.

WTRM$_P$ (4 DW)

+	K28.3(7Ch)+		D21.4(95h)-			D21.5(B5h)+			D21.5(B5h)+		+
	1100b	0011b	0010b	1010b	1101b	1010b	1010b	1010b	1010b	1010b	
	Ch	3h	2h	Ah	Dh	Ah	Ah	Ah	Ah	Ah	

Above Dword is repeated a total of 2 times.

SYNC$_P$ (2 DW)

Table 73 **– Framed Composite Pattern (FCOMP)** (part 7 of 7)

Transmission Order →

	K28.3(7Ch)+		D10.5(AAh)-			D25.4(99h)-		D25.4(99h)+			
+	1100b	0011b	0001b	0101b	1010b	1001b	1011b	0110b	0110b	0010b	-
	Ch	3h	1h	5h	Ah	9h	Bh	6h	6h	2h	
	$CONT_P$										
	K28.3(7Ch)-		D10.5(AAh)+			D25.4(99h)+		D25.4(99h)-			
-	0011b	1100b	1101b	0101b	1010b	1001b	1000b	1010b	0110b	1101b	+
	3h	Ch	Dh	5h	Ah	9h	8h	Ah	6h	Dh	
	$CONT_P$										
	D10.2(4Ah)+		D10.2(4Ah)+			D10.2(4Ah)+		D10.2(4Ah)+			
+	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	0101b	+
	5h	5h	5h	5h	5h	5h	5h	5h	5h	5h	
	Above Dword is repeated a total of 214 times. HFTP (214 DW) Junk data/fill										

Total 2 304 Dwords total
2 DW $ALIGN_P$
7 DW HFTP Junk data/fill
5 DW X_RDY_P
1 DW SOF_P
1 DW Data FIS Header
256 DW SSOP with 2 DW $ALIGN_P$ at
256 DW HTDP (64 DW, 64 DW, 64 DW, 64 DW)
512 DW LTDP (1 DW, 1 DW, 509 DW, 1 DW)
256 DW LBP ((1 DW,1 DW) × 128)
512 DW LFSCP (255 DW, 1 DW, 255 DW, 1 DW)
256 DW HTDP (64 DW, 64 DW, 64 DW, 64 DW)
CRC (1 DW)
EOF_P (1 DW)
$WTRM_P$ (4 DW)
$SYNC_P$ (2 DW)
$CONT_P$ (2 DW)
HFTP (214 DW) Junk data/fill
$ALIGN_P$ (16 DW inserted in pairs at 256 DW interval)

7.4.6 Hot plug considerations

7.4.6.1 Hot plug overview

The purpose of this section is to provide the minimum set of normative requirements necessary for a Serial ATA host or device to be “Hot Plug Capable”. As there exists various Hot Plug events, there are relevant electrical and operational limitations for each of those types of events.

The events are defined below, and the Hot Plug Capability is further classified into:

a) surprise Hot Plug capable; or
b) OS-Aware Hot Plug capable.

If a host or device is Hot Plug Capable without any qualifier, this shall imply that the SATA interface is Surprise Hot Plug Capable.

For the purposes of this specification, Hot Plug operations are defined as insertion or removal operations, between SATA hosts and devices, if either side of the interface is powered.

Gen1m and Gen2m interfaces shall meet the requirements to be classified as Hot Plug Capable. These requirements are not applicable to Gen1i and Gen2i cabled interfaces, however, Gen1i/Gen2i devices used in Short Backplane applications shall be Hot Plug Capable.

Hot Plug Capable hosts/devices shall not suffer any electrical damage, or permanent electrical degradation, and shall resume compliant Tx/Rx operations after the applicable OOB operations, following the Hot Plug Events.

Hot Plug events are:

a) **Asynchronous Signal Hot Plug / Removal,** a signal cable is plugged or unplugged at any time. Power to the host/device remains on since it is sourced through an alternate mechanism that is not associated with the signal cable. This applies to External Single-Lane and Multilane Cabled applications;
b) **Unpowered OS-Aware Hot Plug / Removal,** this is defined as the insertion of a device into or removal of a device from a backplane connector (combined signal and power) that has power shutdown. Prior to removal, the host is placed into a quiescent state (not defined here) and power is removed from the backplane connector to the device. After insertion, the backplane is powered; both the device and host initialize and then operate normally. The mechanism for powering the backplane on/off and transitioning the host into/out of the “quiescent” state is not defined here. During OS-Aware events, the host is powered. This applies to “Short” and “Long” Backplane applications;
c) **Powered OS-Aware Hot Plug / Removal,** this is defined as the insertion of a device into or removal of a device from a backplane connector (combined signal and power) that has power on. After insertion, both the device and host initialize and then operate normally. Prior to insertion or removal, the host is placed into a quiescent state (not defined here) but the backplane connector to the device is powered at all times. The mechanism for transitioning the host into/out of the “quiescent” state is not defined here. During OS-Aware events, the host is powered. This applies to “Short” and “Long” Backplane applications; or
d) **Surprise Hot Plug / Removal,** this is defined as the insertion / removal of a host or device into / from a backplane connector (combined signal and power) that has power on. After insertion, both the device and host initialize and then operate normally. The powered host or device is not in a quiescent state.

NOTE 25 - Suprise hot Plug / Removal does not imply transparent resumption of system-level operation since data may be lost, the device may have to be re-discovered and initialized, etc. Suprise Hot Plug / Removal of a device is not recommended. System designs are recommended to prevent Surprise Hot Plug / Removal of devices.

7.4.6.2 Electrical requirements

AC coupling shall be required. Additional hot plug electrical characteristics should include considerations for common-mode transients, ESD, and drive body discharge.

7.4.6.3 Common-mode transients (informative)

This informative section highlights the maximum transient events encountered during hot plug operations. An example is presented (see Figure 185) depicting some of the Hot Plug relevant specifications of Table 52 (Sequencing Transient Voltage and Common Mode Transient Settle Time), where the impact on hosts/devices is shown.

The maximum current induced by a common mode transient is limited by Vcm and the minimum single-ended impedance of 42.5 ohm. Hence the worst possible surge current is able to be 2 V / 42.5 ohm = 47 mA. The duration of this current is limited by the time constant, C × Rtx ~ 0.5 us. This current should be further reduced by supplying common mode termination at the victim end, assuming ESD diodes do not turn on.

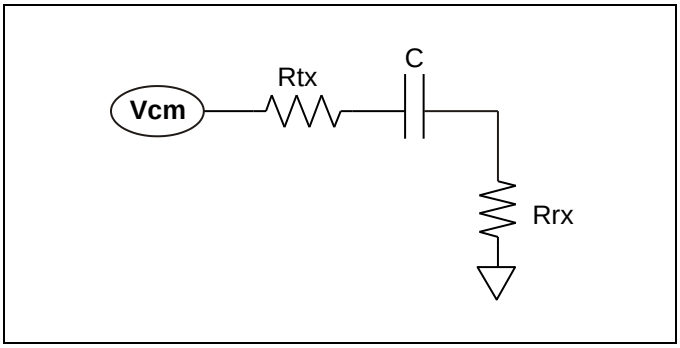


Figure 185 – Example circuit for common mode transients

0 V < Vcm < 2 V
21.25 ohm < Rtx < 40 ohm
21.25 ohm < Rrx < 40 ohm
C < 0.024 uF (Two 0.012 uF capacitors in parallel)

The maximum voltage step that may be transmitted to the "victim" end by a transient at the "aggressor" end is the maximum Vcm. This voltage is added to the existing bias voltage at the victim end. Since terminators have no maximum, single-ended limit, this step is not guaranteed to be reduced by any resistive divider. Voltage transients at the "victim" end, however, may be limited by clamping action of ESD diode structures.

7.4.6.4 ESD (informative)

There is no ESD requirement on the SATA connector interface pins. However, it is recommended that the semiconductors used in the Hot Plug Capable hosts and devices meet the following ESD specifications.

Receiver and Transmitter semiconductor signal pins and power pins should tolerate a minimum of 2 000 V using test methods per JEDEC EIA-JESD22-A114-B, Electrostatic Discharge (ESD) Sensitivity Testing Human Body Model (HBM).

Receiver and transmitter semiconductor signal pins should tolerate 500 V per JESD22-C101-A, Field-Induced Charged-Device Model Test Method for Electrostatic-Discharge-Withstand Thresholds of Microelectronic Components.

7.4.6.5 Drive body discharge (informative)

For all Serial ATA backplane systems, the device canister or enclosure should provide sufficient electrical bonding such that electrostatic potential is discharged from the device body ground to the enclosure ground prior to the connector mating. It is strongly advised for all Serial ATA backplane systems, and device canisters for hot plug capable devices, that the guide-rails are designed to be electrically conductive. The device canister should be designed to have an electrical ground connection to device ground, and the guide-rails within the canister system should be connected to system ground.

7.4.7 Mated connector pair definition

7.4.7.1 Standard mated connector pair definition

The compliance point for receiver and transmitter is at the device/host I/O including the mated connector pair.

Figure 186 shows the mated connector pair detail. The compliance point includes the "tails" of the receptacle pins. The physical description of the receptacle pin tails is shown in Figure 187.

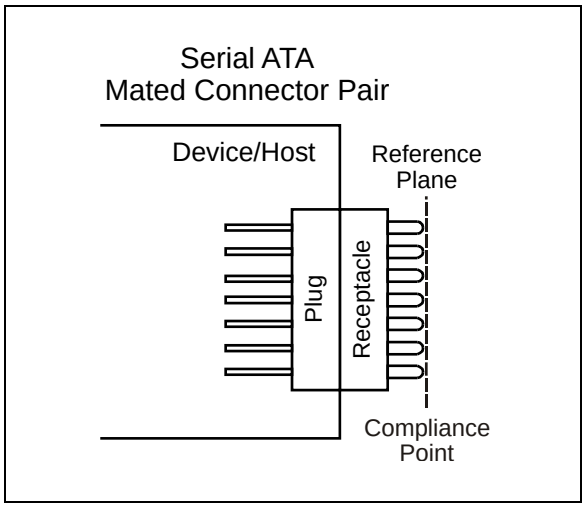


Figure 186 – Mated connector pair

The signal interface to the mated pair connector pin tails should be done with care to minimize parasitic capacitance or inductance. The connector pin, tails are a coplanar waveguide transmission line in a ground, signal, signal, ground (GSSG) configuration. The signal interface to the pin, tails should maintain the GSSG configuration.

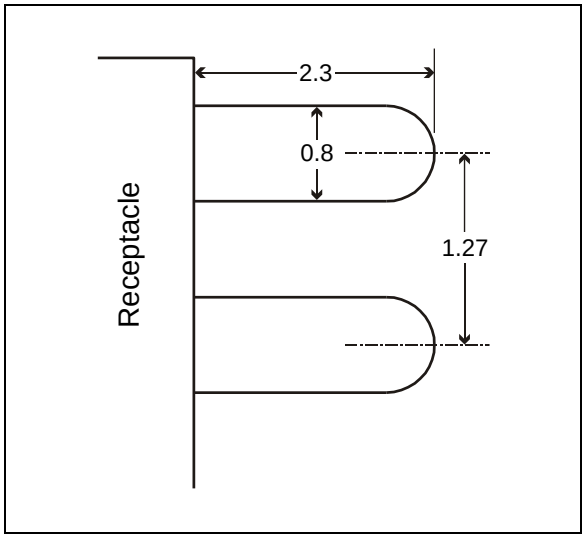


Figure 187 – Mated connector pair, pin tail detail

7.4.7.2 mSATA connector connection definition

The compliance points of mSATA are shown in Figure 188. The same concept of compliance point of SATA extends to the mSATA application. The detailed physical pin dimensions are shown in Figure 189.

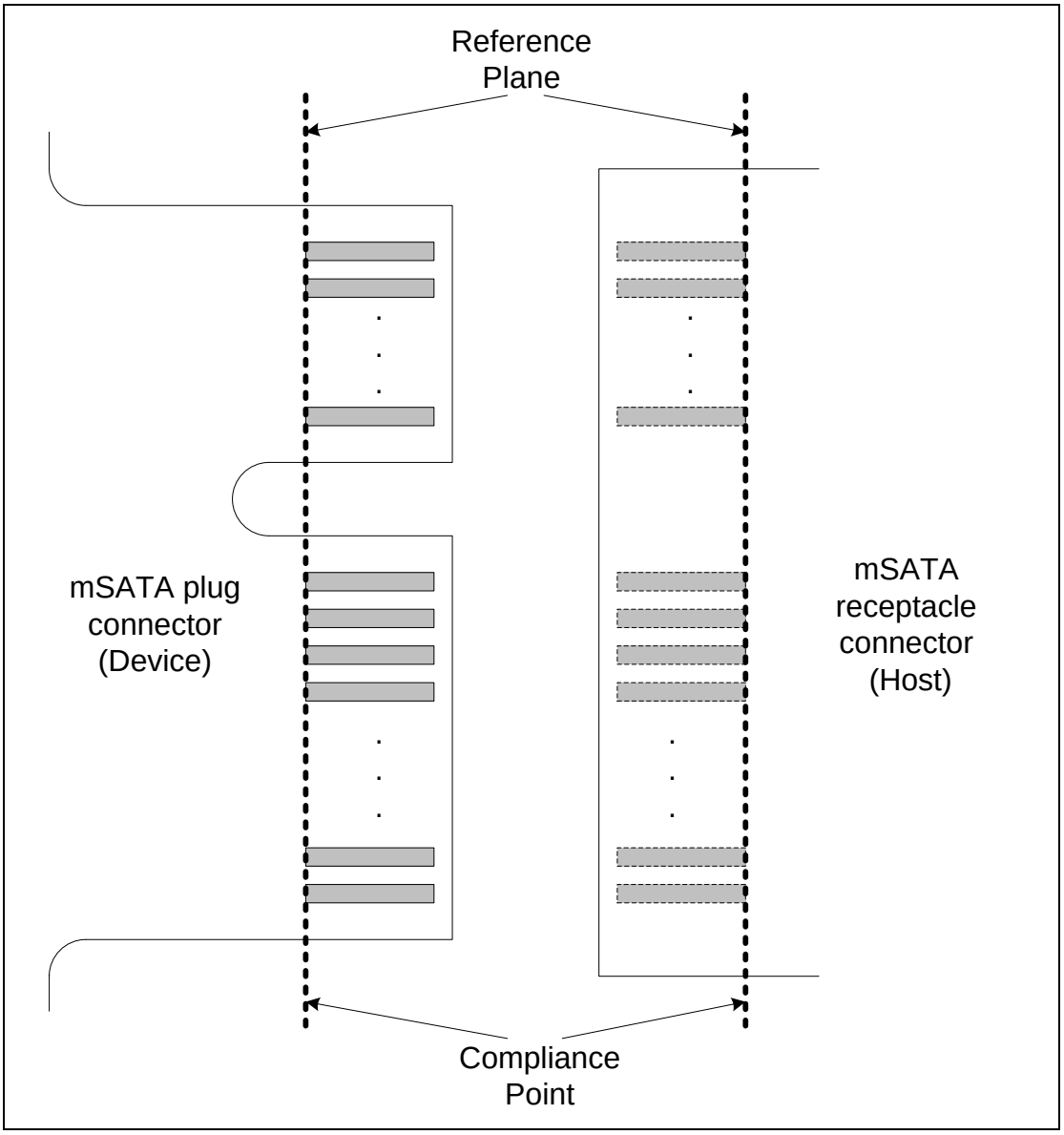


Figure 188 – Mated connector pair for mSATA

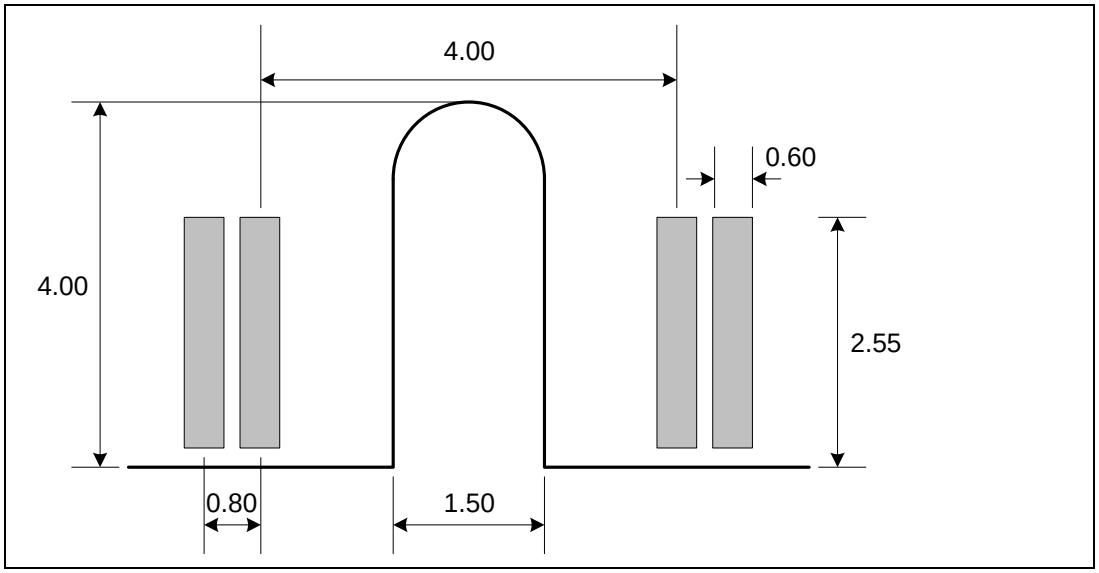


Figure 189 – mSATA connector pin detail

7.4.7.3 M.2 connector compliance point

The compliance points of M.2 are shown in Figure 190.

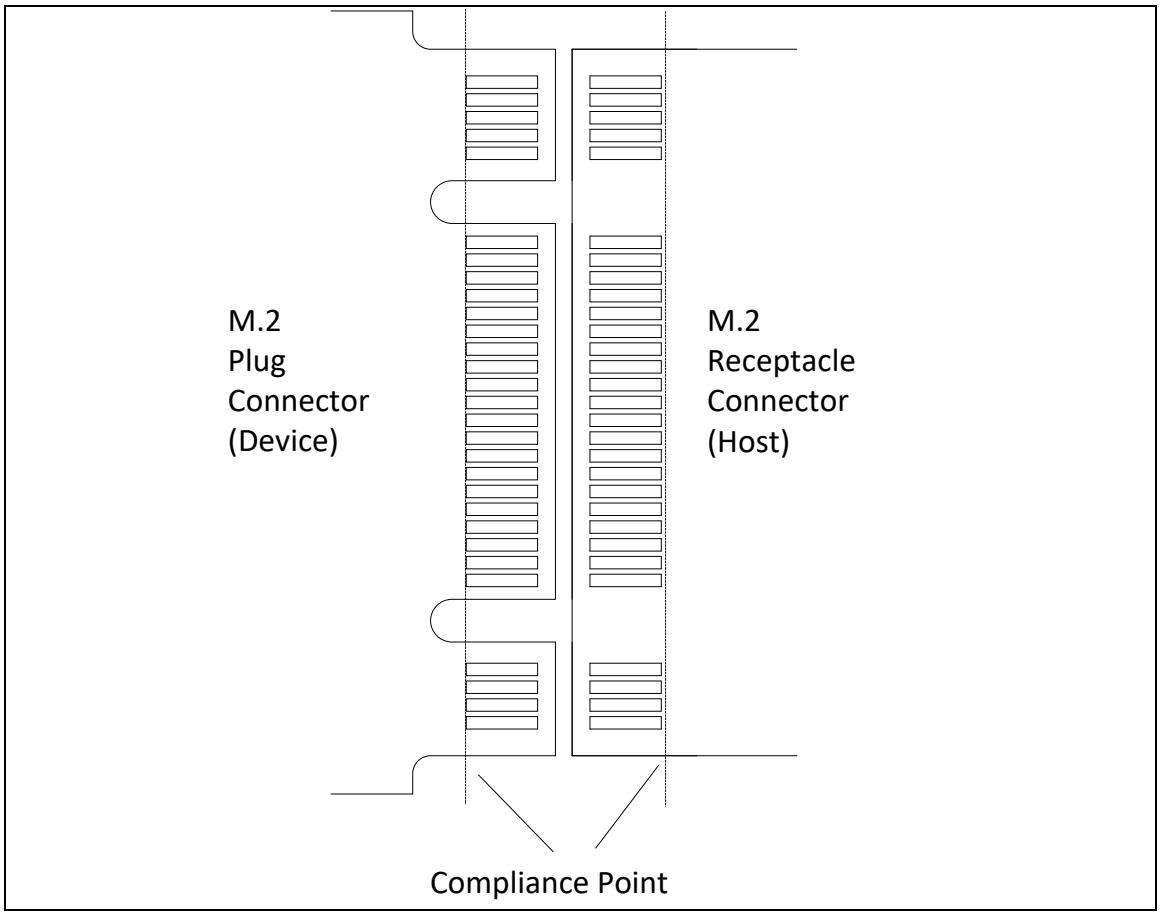


Figure 190 – M.2 compliance point

7.4.8 Compliance Interconnect Channels (CIC) (Gen3i, Gen3u)

7.4.8.1 Compliance Interconnect Channels (CIC) overview

For Gen3i, a Compliance Interconnect Channel (CIC) is defined as a set of calibrated physical test circuits applied to the Transmitter mated connector, intended to be representative of the highest-loss interconnect (see Figure 191). For Gen3u, the Gen3i CIC is used to set up the Lab-Sourced Signal and then removed prior to applying the signal to the UHost receiver under test (see Table 58). A CIC is not used in testing the UHost transmitter.

The CIC is used to verify that the signal electrical characteristics at the Transmitter mated connector are sufficient to ensure compliance to the input electrical specifications for Gen3i receivers as delivered through worst-case media. The magnitude of this worst-case loss as a function of frequency is defined mathematically as a Transmitter Compliance Transfer Function (TCTF). A CIC (see 7.4.8.2) is a linear, passive, differential two-port (e.g., a SATA cable, physical test fixture, or mathematical computation). A CIC shall have a loss greater than or equal to the TCTF loss at all frequencies and meet the ISI loss constraint (defined below). Any deviations of the CIC loss from the TCTF loss is a measurement error. Deviation of the CIC from the specified differential impedance results in measurement error. To reduce the measurement error, the CIC loss should match the TCTF without violation.

A combination of a zero-length test load (i.e., the Laboratory Load) plus the applicable CIC (Gen3i) is used for the specification of the host-controller or device transmitter characteristics.

A Gen3i transmitter signal is specified by meeting:

a) all parameters in Table 54 for Gen3i when transmitting into a Laboratory Load; and
b) Table 54 VdiffTx and total jitter (TJ) after CIC requirements for Gen3i when transmitting through the appropriate Gen3i CIC into a Laboratory Load while using the same transmitter settings (emphasis, amplitude, etc.) as in the first test.

NOTE 26 - Note that the Transmitter Compliance Specifications are defined and measured into a Laboratory Load. Received signal attenuation or amplification due to actual receiver terminator tolerance as well as additional received signal ISI due to the actual receiver return loss may further degrade the actual receiver's input signal. Transmitter Compliance Specifications are expected to be only slightly tighter than Receiver Specifications.

The transmission magnitude response, $|S_{DD21}|$, of the Gen3i TCTF satisfies the following two inequalities:

$$|S_{DD21}| \leq -20 \times \log_{10}(e) \times ((3.0 \times 10^{-6} (f^{0.5})) + (1.1 \times 10^{-10} (f))) \text{ dB}$$
for 50 MHz < f < 9.0 GHz, (f expressed in Hz),

$$|S_{DD21}| \text{ at } 600 \text{ MHz} - |S_{DD21}| \text{ at } 3\,000 \text{ MHz} > 2.7 \text{ dB}$$

NOTE 27 - Note that "e" in the first expression is the base of the natural logarithms, approximately 2.71828. Hence, the first factor, $20 \log_{10}(e)$, evaluates to approximately 8.6859. This value is the conversion factor from nepers (defined as the natural logarithm of a power ratio) to decibels.

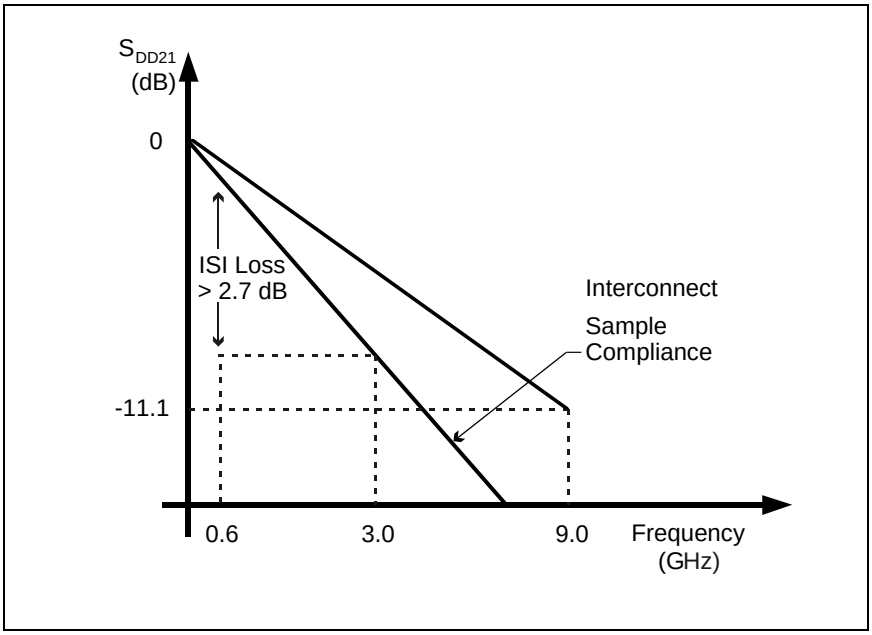


Figure 191 – Compliance Interconnect Channel (CIC) loss for Gen3i

The second constraint, termed ISI loss, may be motivated as follows, $|S_{DD21}|$ at one tenth the data rate is the attenuation of the fundamental component of a repeating five-ones-five-zeroes pattern, the longest possible run lengths in 8b/10b encoded data. Similarly, $|S_{DD21}|$ at one half the data rate is the attenuation of the fundamental component of a repeating 0101b pattern, the shortest possible run lengths in 8b/10b encoded data.

Hence, for an output waveform of this TCTF, ISI loss approximates the ratio between:

a) the peak-peak voltage (established by the long run lengths); and
b) the inside vertical eye opening (established by the high frequency pattern).

A TCTF with a flatter loss characteristic (i.e., with more broadband attenuation) generates less inter-symbol interference (ISI) and therefore less output jitter. This constraint prohibits such a TCTF.

7.4.8.2 Calibration of Compliance Interconnect Channels (CIC)

The TCTF defines the worst-case cable loss inclusive of the two SATA mated connector pairs in the path from transmitter to receiver. The loss due to these two mated connector pairs shall be included in the transmitter characterization. That is, the transmitter shall be tested with the TCTF-defined loss.

For a CIC implemented with SubMiniature version A (SMA) connectors, as seen in Figure 215 that the addition of a SATA adapter (plug) following the CIC and driving into a Laboratory Load provides the required combined total loss of TCTF (embodied in the CIC and adapter loss).

7.4.9 Impedance calibration (optional)

Hosts and devices may employ on-chip adaptive impedance matching circuits to ensure best possible termination for both its transmitter and receiver.

The host, since it is given the first opportunity to calibrate during the power on sequence, is undable to assume that the far end of the cable is calibrated yet. For this reason, the host controller should

utilize a separate reference to perform calibration. In a desktop system, the cable provides the optimal impedance reference for calibration.

Using TDR techniques, the host may launch a step waveform from its transmitter, so as to get a measure of the impedance of the transmitter, with respect to the cable, and adjust its impedance settings as necessary.

In a mobile system environment, where the cable is small or non-existent, the host controller should make use of a separate reference (e.g., an accurate off-chip resistor) for the calibration phase.

The device, on the other hand, may assume that the termination on the far side (host side) of the cable is fully calibrated, and may make use of this as the reference. Using the host termination as the calibration reference allows the devices operating in both the desktop and the mobile system environment to use the same hardware.

Signals generated for the impedance calibration process shall not duplicate the OOB signals, COMWAKE, COMINIT, or COMRESET. Signals generated for the impedance calibration process shall not exceed the normal operating voltage levels in accordance with 7.4. See the power management section for suggested times to perform calibration during power-on.

7.5 Jitter

7.5.1 Jitter overview

Jitter is the short-term variations of the zero crossings from ideal positions in time. A "Reference Clock" as defined in 7.5.3 determines the ideal positions in time. The Reference Clock method provides for the separation of jitter from SSC, tracking SSC, and other low frequency modulation but not jitter.

There are several types of jitter separated into two classes:

a) deterministic; and
b) random.

Deterministic jitter is bounded and random jitter is not. The amount of tolerable jitter is limited by the desired bit error rate performance of the channel. Two classes of jitter are used in analysis because they accumulate differently.

The Serial ATA data stream employs an embedded clock (i.e., no clock signal is separately sent). At the receiver, the Serial ATA data stream is re-clocked to form a parallel digital signal. Adequate timing margin is required for this process to function properly. Jitter analysis is the timing analysis used in systems with an embedded clock.

In SATA systems, random jitter is a significant portion of the total jitter causing occasional errors to occur. If a bit error occurs, the error is detected when an entire frame of bits is received. The bit error is corrected by retransmitting the frame. If two bit errors occur within a single frame, the corrective action is the same. The data throughput on the channel is diminished if frames are retransmitted. Frame Error Rate is the channel performance measure.

Since a portion of the jitter is random, a measurement of jitter also has a random nature (i.e., repeated measurements yield results that are somewhat different). As the sample size of each measurement increases, the spread of the measurement results decreases. A measured value of random jitter is determined to a known confidence level.

A Frame Error Rate test is a system performance test done on a combination of SATA compliant components. To achieve a statistically significant estimate of the Frame Error Rate a large sample size is necessary. A Frame Error Rate test on a SATA channel is lengthy requiring about an hour at Gen2 rates.

Jitter tests are compliance tests done on an individual SATA component, a device, host, or interconnect to ensure system performance. Compliance tests help to predict the performance of combinations of compliant components. It is often desirable to make jitter measurements in a short period of time rather than hours. Consequently, jitter measurements are done with small sample sizes and the results are extrapolated to predict results with larger sample size.

Extrapolation of results from small sample size to large sample size involves assumptions. This specification defines two assumptions as normative.

The assumptions are:

a) the random jitter has a Gaussian distribution; and
b) the total jitter (at a BER of 10^{-12}) is the sum of the deterministic jitter plus 14 times the standard deviation of the random jitter.

These allow the separation of deterministic from random jitter, and an estimate of the total jitter for an equivalent BER of 10^{-12} from a much smaller sample size.

7.5.2 Jitter definition

Jitter is defined as the difference in time between a data transition and the associated Reference Clock event. The jitter at the receiver is the result of the aggregate jitter in the transmission path. First, jitter is generated during clocking of the data in the transmitter. Then, each element in the channel between the transmitter and the receiver influences the jitter. Finally, the receiver shall be able to recover the data despite the jitter, otherwise errors occur. The receiver jitter tolerance shall be greater than the transmitter's generated jitter and the expected jitter accumulation through the channel.

Jitter budgets are dependent on the desired bit error rate (BER). SATA assumes a BER target of less than 10^{-12}. Jitter levels are defined as Reference Clock to data. The Reference Clock is extracted from a serial data stream using either a Phase Lock Loop (PLL) (hardware) or a clock recovery algorithm (software).

The Reference Clock to data jitter methodology allows for jitter measurements to be made on a device or host using a Spread Spectrum Clock or a non-spreading clock.

7.5.3 Reference clock definition

7.5.3.1 Reference clock definition overview

The Reference Clock is defined as that clock recovered from a Serial ATA data stream. The Reference Clock provides the distinction between Spread Spectrum Clocking (SSC) and jitter. The Reference Clock tracks SSC and wander, but not jitter. In addition, it provides a definition for determining the SSC profile. Reference Clock extraction is performed using either hardware or software PLLs.

7.5.3.2 Gen1i, Gen1m, Gen1u, Gen2i, Gen2m, and Gen2u normative requirements

For Gen1i, Gen1m, Gen1u, Gen2i, Gen2m, and Gen2u the Reference Clock characteristics are controlled by the resulting Jitter Transfer Function (JTF) characteristics obtained by taking the time difference between the Type 2 PLL output (the Reference Clock) and the data stream sourced to the PLL. The PLL Closed Loop Transfer Function (CLTF) -3 dB corner frequency, and other adjustable CLTF parameters such a peaking, are determined by the value required to meet the requirements of the JTF (see 7.6.10).

The JTF for Gen1i, Gen1m, and Gen1u shall have the following characteristics for an encoded Gen1 D24.3 pattern (e.g., 1100 1100 1100 1100 1100b). This is the Gen1 MFTP that is a test pattern having a clock-like characteristics and a transition density of 0.5. Gen1u shall use the values shown in Table 36 specified for Gen1i/Gen1m.

The JTF for Gen2i, Gen2m, and Gen2u shall have the following characteristics for an encoded Gen2 D24.3 pattern (e.g., 1100 1100 1100 1100 1100b). This is the Gen2 MFTP that is a test pattern having a clock-like characteristics and a transition density of 0.5. Gen2u shall use the values shown in Table 36 specified for Gen2i/Gen2m.

The calibration procedure is:

1) the -3 dB corner frequency of the JTF shall be as shown in Table 54 Jitter Transfer Function Bandwidth (D24.3, high pass -3 dB);
2) the magnitude peaking of the JTF shall be as shown in Table 54 Jitter Transfer Function Peaking; and
3) the attenuation at Jitter Transfer Function Low Frequency Attenuation Measurement Frequency in Table 54 shall be as shown in Table 54 Jitter Transfer Function Low Frequency Attenuation.

The JTF -3 dB corner frequency and the magnitude peaking requirements shall be measured with sinusoidal PJ applied, with a peak-to-peak amplitude of 0.3 UI with a relative tolerance of ± 10 %. The attenuation at 30 kHz shall be measured with sinusoidal phase (time) modulation applied, with a peak-to-peak amplitude of 20.8 ns with a relative tolerance of ± 10 %.

7.5.3.3 Gen1i, Gen1m, Gen1u, Gen2i, Gen2m, and Gen2u informative comments

Typically a CLTF -3 dB corner frequency of $f_{BAUD}/500$ is able to provide a JTF with characteristics close to the requirements, but due to differences in Type 2 PLL designs, the actual CLTF settings required to meet the required JTF are able to vary widely.

It is desired that the phase response of the JTF of a jitter measurement device (JMD) (Reported Jitter / Applied Jitter) be that of the JTF of the time difference of the output of a Type 2 PLL to the Data stream applied to the PLL. This is the reference design. In the presence of multiple jitter component frequencies, the relative phase at these frequencies determines how they are combined to construct the final reported jitter value. In the case of discrepancies between the reported jitter levels, between JMDs with the same JFT magnitude response, the JMD with the JTF phase characteristics closest to that of the reference design, shall be considered correct. The JTF phase response of a JMD is important, but it is not always possible to determine this without proprietary information concerning the JMD processing methods, and it is not externally observable in some classes of JMDs.

The JTF of the time difference of the output of a Type 2 PLL to the Data stream applied to the PLL, or the reference design, is defined with a pattern that has a transition density of 0.5. Since this Type 2 PLL contains a sampled data mode phase detector, with a gain that varies proportionally with transition density, the JTF -3 dB corner frequency should change with the transition density of the applied pattern. For a well designed PLL, with significant phase margin in the open loop response, the JTF -3 dB corner frequency, with shift proportionally with the change of pattern transition density.

EXAMPLE - For example, the 2.1 MHz JTF -3 dB corner frequency, set with a pattern with a transition density of 0.5, shifts to 4.2 MHz if a pattern with a transition density of 1.0, (e.g., the D10.2 pattern), is applied.

A proportional decrease of the JTF -3 dB corner frequency should also be observed for a decrease in pattern transition density compared to a 0.5 transition density. This is the expected JMD response to changes in pattern transition density as the reference design is able to exhibit. If a JMD shifts the JTF -3 dB corner frequency in a manner that does not match this characteristic, or does not shift at all, measurements of jitter with patterns with transition densities significantly different than 0.5 may lead to discrepancies in reported jitter levels. In the case of reported jitter discrepancies between JMDs, the JMD with the shift of the -3 dB corner frequency, closest to the proportional characteristic of the reference design, it to be considered correct. This characteristic may be measured using the conditions defined above for measuring the -3 dB corner frequency, using multiple patterns with different transition densities.

7.5.3.4 Gen3i and Gen3u normative requirements

For Gen3i and Gen3u the Reference Clock characteristics are controlled by the resulting Jitter Transfer Function (JTF) characteristics obtained by taking the time difference between the Type 2 PLL output (the Reference Clock) and the data stream sourced to the PLL. The PLL CLTF - 3 dB corner frequency, and other adjustable CLTF parameters such a peaking, are determined by the value required to meet the requirements of the JTF. (see 7.6.10)

The JTF for Gen3i shall have the following characteristics for an encoded Gen3 D24.3 pattern (1100 1100 1100 1100 1100b). This is the Gen3 MFTP that is a test pattern having a clock-like characteristics and a transition density of 0.5. Gen3u shall use the values shown in Table 36 specified for Gen3i.

The calibration procedure is:

1) the -3 dB corner frequency of the JTF shall be 4.2 MHz ± 2 MHz. as shown in Table 54 Jitter Transfer Function Bandwidth (D24.3, high pass -3 dB)(Gen3);
2) the magnitude peaking of the JTF shall be max 3.5 dB as shown in Table 54 Jitter Transfer Function Peaking (Gen3); and
3) the attenuation at 420 kHz, with a relative tolerance of ± 1 % Jitter Transfer Function Low Frequency Attenuation Measurement Frequency (Gen3) in Table 54 shall be 38.2 dB ± 3 dB as shown in Table 54 Jitter Transfer Function Low Frequency Attenuation (Gen3).

The JTF -3 dB corner frequency and the magnitude peaking requirements shall be measured with sinusoidal PJ applied, with peak-to-peak amplitude of 0.3 UI, with a relative tolerance of ± 10 %. The attenuation at 420 kHz shall be measured with sinusoidal phase (time) modulation applied, with peak-to-peak amplitude of 1.0 ns, with a relative tolerance of ± 10 %. The attenuation is measured on the 40 dB/dec slope of the JTF at 1/10 the -3 dB corner frequency nominal target value. This is equivalent to 72 dB at 60 kHz for an ideal 40 dB/dec slope that corresponds to a 2X increase of JTF BW for Gen3i compared to Gen2i. This shift in measurement point allows for improved practical measurements and lower test signal phase modulation level requirements.

7.5.4 Spread Spectrum Clocking

7.5.4.1 Spread Spectrum Clocking overview

Serial ATA allows the use of Spread Spectrum Clocking, or intentional low frequency modulation of the transmitter clock. The purpose of this modulation is to spread the spectral energy to mitigate the unintentional interference to radio services. The modulation frequency of SSC shall be in the range defined for f_{SSC} in Table 52.

The modulation frequency deviation shall be in the prescribed range for SSC_{tol} in Table 52. The instantaneous frequency (each period) of the Reference Clock shall fall within the prescribed T_{UI} range. If the rate of change of the instantaneous frequency is excessive, then jitter is increased.

The SSC modulation only moves the frequency below the nominal frequency. This technique is often called “down-spreading”.

7.5.4.2 Example SSC profile (informative)

An example triangular frequency modulation profile is shown in Figure 192. The modulation profile in a modulation period is expressed as:

$$f = (1 - \delta)\, f_{nom} + 2 \times f_m \times \delta \times f_{nom} \times t$$

when:

$$0 < t < \frac{1}{2 \times f_m}$$

and:

$$f = (1 + \delta)\, f_{nom} - 2 \times f_m \times \delta \times f_{nom} \times t$$

when:

$$\frac{1}{2 \times f_m} < t < \frac{1}{f_m}$$

where f_{nom} is the nominal frequency in the non-SSC mode, f_m is the modulation frequency, δ is the modulation amount, and t is time.

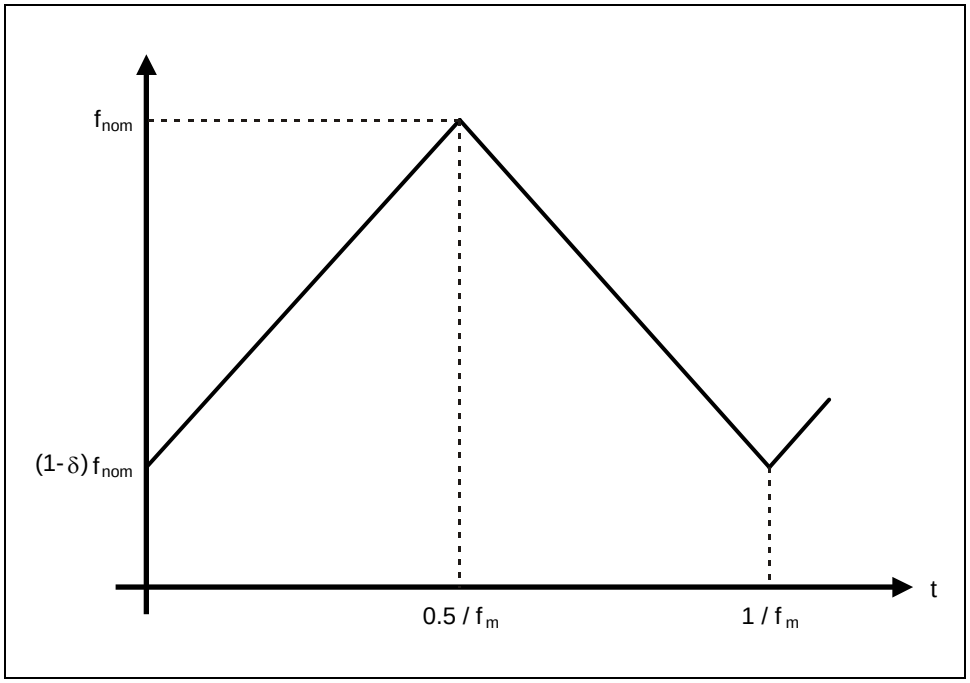


Figure 192 – SSC profile example, triangular

As an example, for triangular modulation, the absolute spread amount at the fundamental frequency is shown in Figure 193, as the width of its spectral distribution.

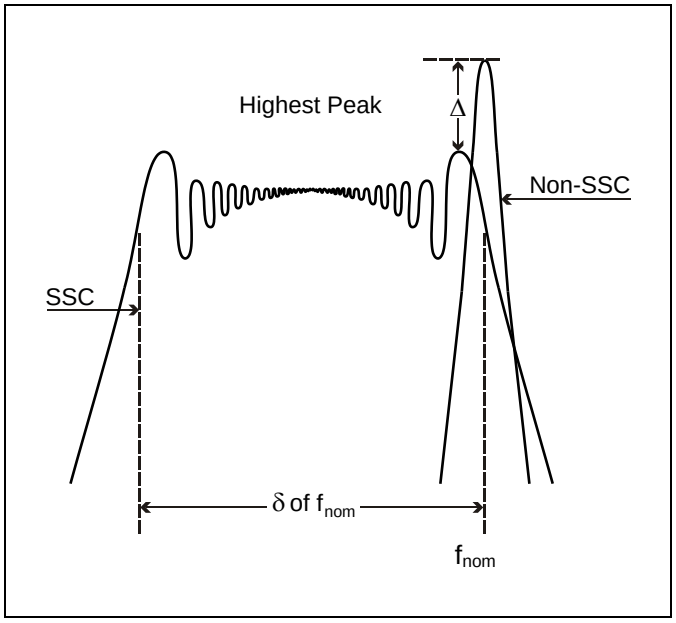


Figure 193 – Spectral fundamental frequency comparison

7.5.5 Jitter budget

There are two types of jitter, random jitter (RJ) and deterministic jitter (DJ). Random jitter is Gaussian and unbounded. For ease, the standard deviation (RJ_σ) is multiplied by a factor that

corresponds to the target BER. For a target BER = 10^{-12}, the associated multiplication factor for Serial ATA is 14.

TJ is peak-to-peak and defined as:

$$TJ = (14 \times RJ_{\sigma}) + DJ$$

Table 54, Table 55, Table 57, and Table 58, show the compliance jitter values. The measurement of jitter is according to 7.6.10.

7.6 Measurements

7.6.1 Measurements overview

The performance of a SATA system with host and device connected together is measured by the Frame Error Rate, using a set of reference frames, defined by a specific set of ordered test patterns within the frame. A host or device is commanded to generate the various test patterns through the use of the BIST Activate FIS or other vendor-specific commands to the device under test.

Measurements of devices and hosts are done to determine compliance with this specification. Compliance tests are done with a device or host connected to test equipment. Compliance tests are not done with devices and hosts connected together. Unless otherwise specified, all compliance measurements shall be taken through the mated connector pair.

The values specified in Table 54 and Table 55 refer to the output signal from the UUT at the mated connector into a Laboratory Load. The signals are not specified while attached to a system cable or backplane.

The values specified in Table 57 and Table 58 refer to the input signal from any signal source as measured at the UUT using a Laboratory Load.

The components that make up the system each affect the system's performance, not simply by summing up the low-level parameters. There are many interactions as well as protocol effects (e.g., the retry algorithms).

Fundamental to this specification is a clear definition of the UUT. The UUT consists of the host/device and the receptacle side of the mated pair connector. This places the signals at a point in the test measurement setup where all specifications of the UUT are defined at an impedance level of 50 ohm each signal line to ground which is 100 ohm differential and 25 ohm common mode impedance. This is the compliance point for hosts and devices.

The Serial ATA Phy layer compliance shall be tested using the Parametric Method. This method uses repetitive patterns and a Laboratory Load to allow accurate, repeatable measurements to be performed on the UUT.

Additional measurement methods for several parameters are aimed at providing quick No-Go testing. These use any valid data pattern and a Laboratory Load to quickly produce a visual "picture" of the performance of the UUT.

EXAMPLE - For example, one measurement method uses Data Eyes to quickly understand jitter.

Another uses a mode measurement for identifying a potentially complex signal with a single amplitude value. However, none of these No-Go measurement methods may be used for testing compliance to electrical specifications. These measurement methods are valuable for gaining useful information about the performance of the UUT that goes beyond specification compliance issues.

Both methods produce measurements of electrical performance, however, the parametric method shall be used for validation of the UUT to the Serial ATA requirements defined in 7.4 while other methods may be used as general No-Go tests.

7.6.2 Test fixtures

7.6.2.1 Lab-load

7.6.2.1.1 Standard lab-load

The lab-load is an electrical test system connected to the UUT. The serial transmitter signals from the UUT are connected through a “mated SATA connector pair” module consisting of connectors and cables to a High Bandwidth Scope (HBWS) terminated into two 50 ohm ± 5 ohm loads. The cables shall be 50 ohm ± 5 ohm impedance. The inputs of the Laboratory Load (from the back of the mated SATA connector to the 50 ohm load within the HBWS) shall have an individual return loss greater than 20 dB over a bandwidth of 100 MHz to 5.0 GHz, and greater than 10 dB from 5 GHz to 8 GHz. The skew between the channels under test shall have 10 picoseconds or less after compensation. The lab-load consists of this total assembly. The lab-load does not include the “other half of the mated connector” that is considered part of the UUT but is physically located on the lab-load. The lab-load is shown in Figure 194.

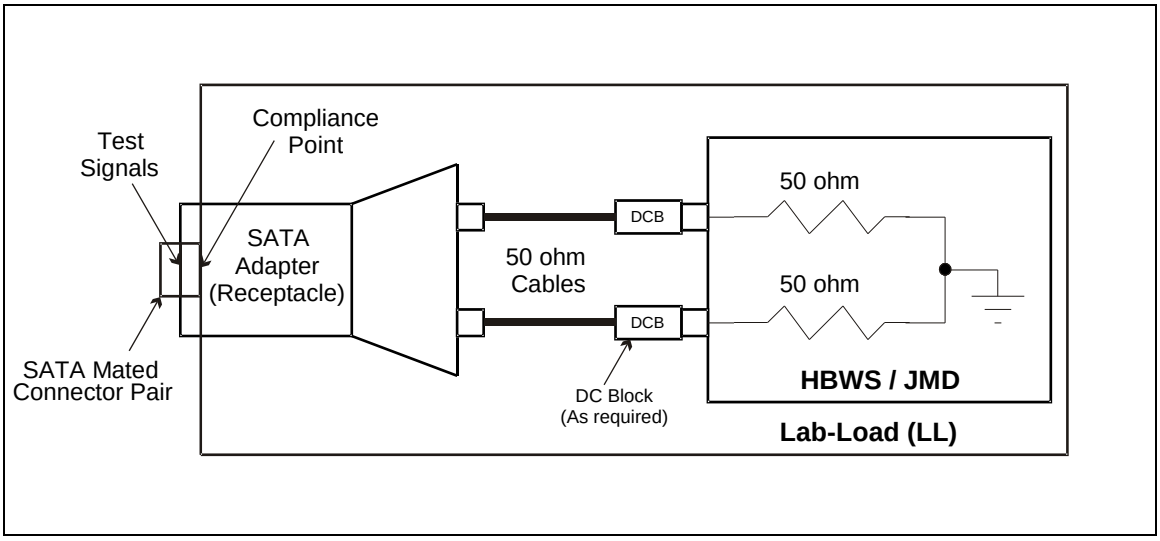


Figure 194 – Lab-Load (LL)

The electrical characteristics of the lab-load shall be greater than the required performance of the parameter being measured such that the effects of the lab-load on the parameter under test may be successfully compensated for, or de-embedded, in the measured data.

7.6.2.1.2 mSATA lab-load

Due to the direct connection of mSATA, two different types of mSATA adaptors are required as lab-loads. The device shall be mated to a female adaptor and the host shall be mated to a male connection as shown in Figure 195 and Figure 196, respectively.

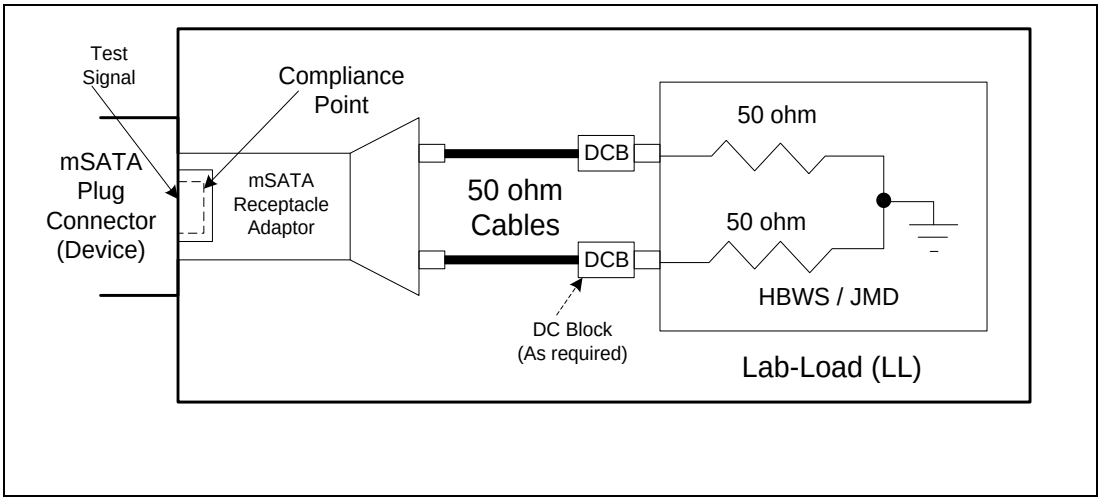


Figure 195 – Lab-Load (LL) for mSATA device

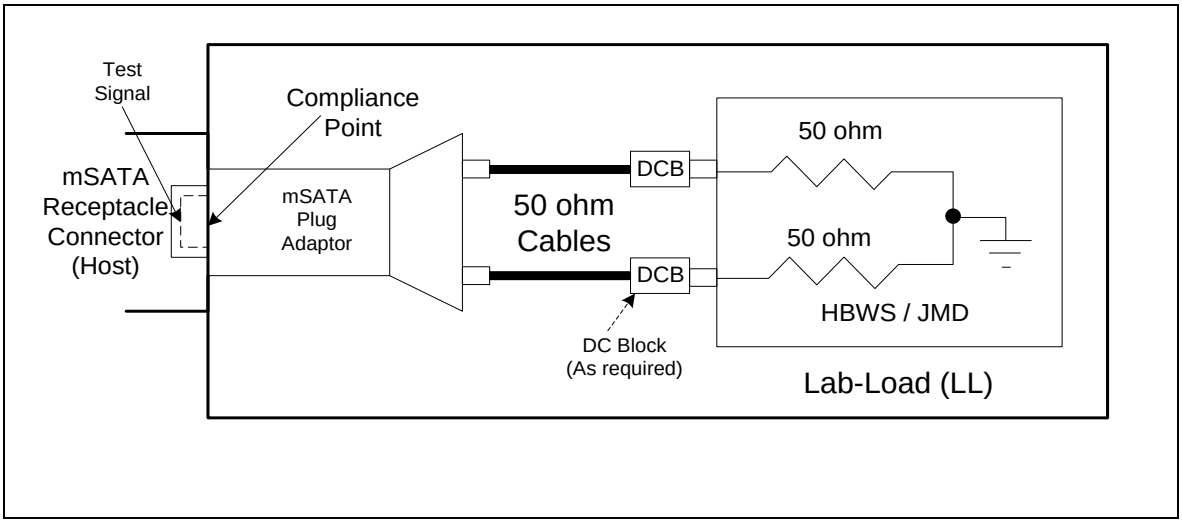


Figure 196 – Lab-Load (LL) for mSATA host

The electrical characteristics of the lab-load shall be greater than the required performance of the parameter being measured such that the effects of the lab-load on the parameter under test may be successfully compensated for, or de-embedded, in the measured data.

7.6.2.1.3 SATA USM host lab-load

Due to the direct connection of the SATA USM host, the host shall be mated to a plug connection as shown in Figure 197. The adaptor may require additional mechanical clearance to mate to the SATA USM host connector. See SFF publication INF-8280 for host connector details.

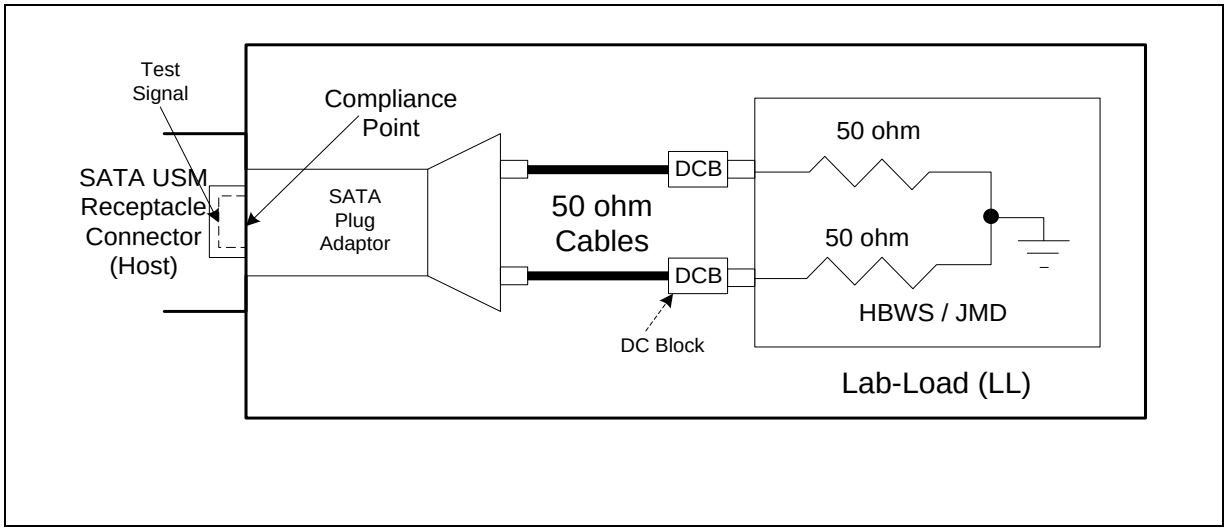


Figure 197 – Lab-Load (LL) for SATA USM host

The electrical characteristics of the lab-load shall be greater than the required performance of the parameter being measured such that the lab-load effects on the parameter under test may be successfully compensated for, or de-embedded, in the measured data.

7.6.2.1.4 SATA MicroSSD lab-load

Due to the embedded application of the SATA MicroSSD, two different types of SATA MicroSSD adaptors are required as lab-loads. The device shall be mated to an adaptor as shown in Figure 198 and the host shall be mated to an adaptor as shown in Figure 199, respectively. It is not possible to test the SATA MicroSSD host if the SATA MicroSSD device is soldered in place.

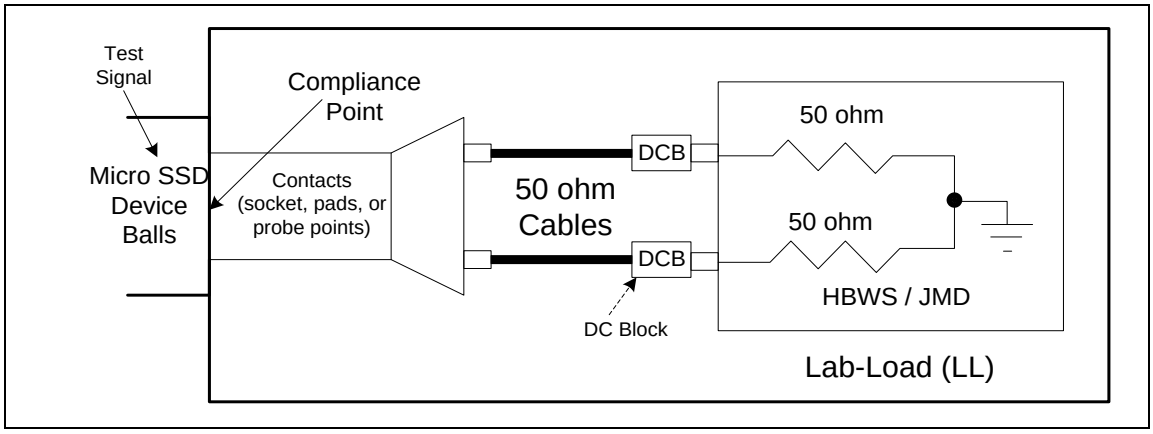


Figure 198 – Lab-Load (LL) for SATA MicroSSD device

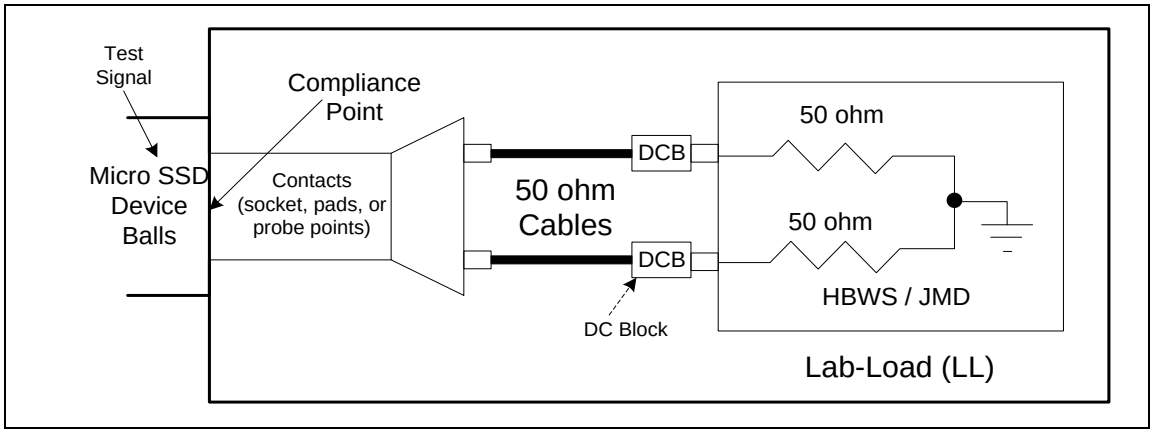


Figure 199 – Lab-Load (LL) for SATA MicroSSD host

The electrical characteristics of the lab-load shall be greater than the required performance of the parameter being measured such that the lab-load effects on the parameter under test may be successfully compensated for, or de-embedded, in the measured data.

In a test environment, a test fixture and methodology may be used that makes it possible to deembed the signal attenuation effects of the text fixture, and thus measure compliance at the solder balls on the device and/or the socket/pads on the host.

7.6.2.1.5 M.2 lab-load

Due to the direct connection of M.2, two different types of M.2 adaptors are required as lab-loads. The device shall be mated to a female adaptor and the host shall be mated to a male connection as shown in Figure 200 and Figure 201.

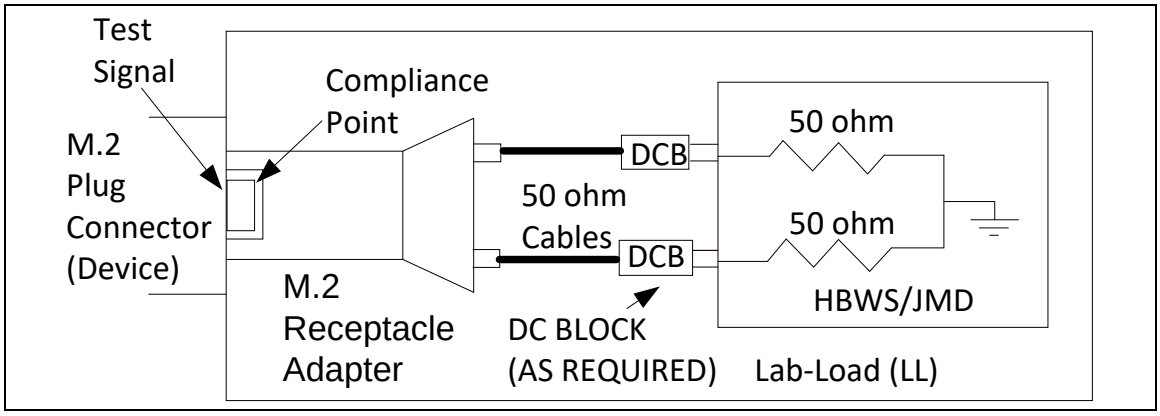


Figure 200 – Lab-Load (LL) for M.2 device

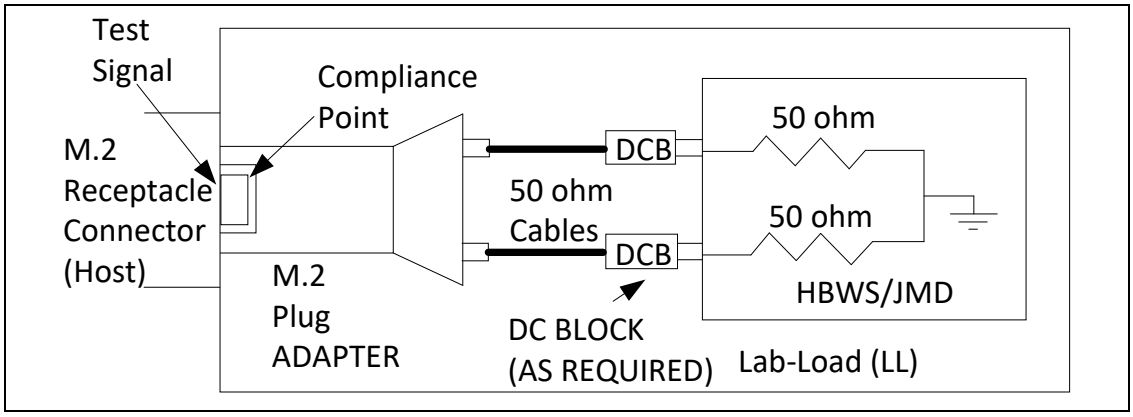


Figure 201 – Lab-Load (LL) for M.2 host

The electrical characteristics of the lab-load shall be greater than the required performance of the parameter being measured such that the lab-load effects on the parameter under test may be successfully compensated for, or de-embedded, in the measured data.

7.6.2.2 Lab-sourced signal

7.6.2.2.1 Lab-sourced signal overview

This section contains the details on Table 57 entries.

The laboratory sourced signal or Lab-Sourced Signal (LSS) is an instrument and electrical test system connected to the UUT (see Figure 202). The Lab-Sourced Signal provides a signal to the UUT at the defined impedance level of 100 ohm differential and 25 ohm common mode. The Lab-Sourced Signal may also provide a SATA signal with impairments (e.g., jitter and common mode noise). The Lab-Sourced Signal may consist of several instruments in combination with fixturing to create a signal with impairments.

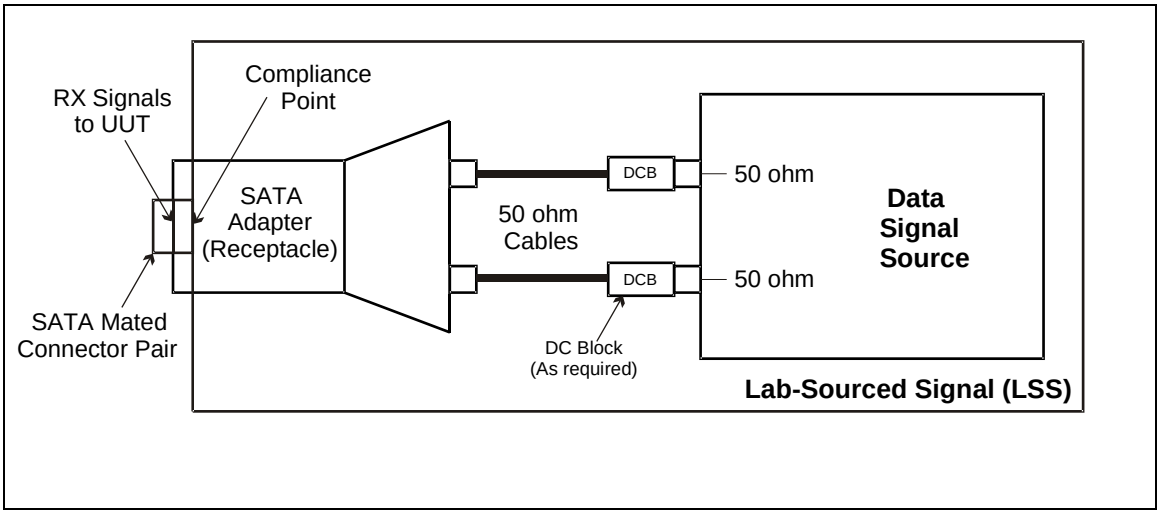


Figure 202 – Lab-Sourced Signal (LSS)

The Lab-Sourced Signal is a laboratory generated signal that is calibrated into an impedance matched load of 100 ohm differential and 25 ohm common mode and then applied to the Rx+ and Rx- signals of the Receiver Under Test. In the case of Gen3i, the Gen3i CIC is inserted in the signal

path applied to the Rx (see 7.6.14). The load used to calibrate the Lab-Sourced Signal shall have an individual return loss greater than 20 dB over a bandwidth of 100 MHz to 5.0 GHz, and greater than 10 dB from 5 GHz to 8 GHz. During calibration, the characteristics of the Lab-Sourced Signal shall comply with the specifications of Table 57. This signal is then applied to the Receiver Under Test, and the Frame Error Rate specifications of Table 52 shall be met.

7.6.2.2.2 mSATA Lab-Sourced Signal

As described in mSATA lab-load 7.6.2.1.2, to properly provide a SATA signal into mSATA device, both female and male type mSATA adaptors are required for device and host, respectively as shown in Figure 203 and Figure 204.

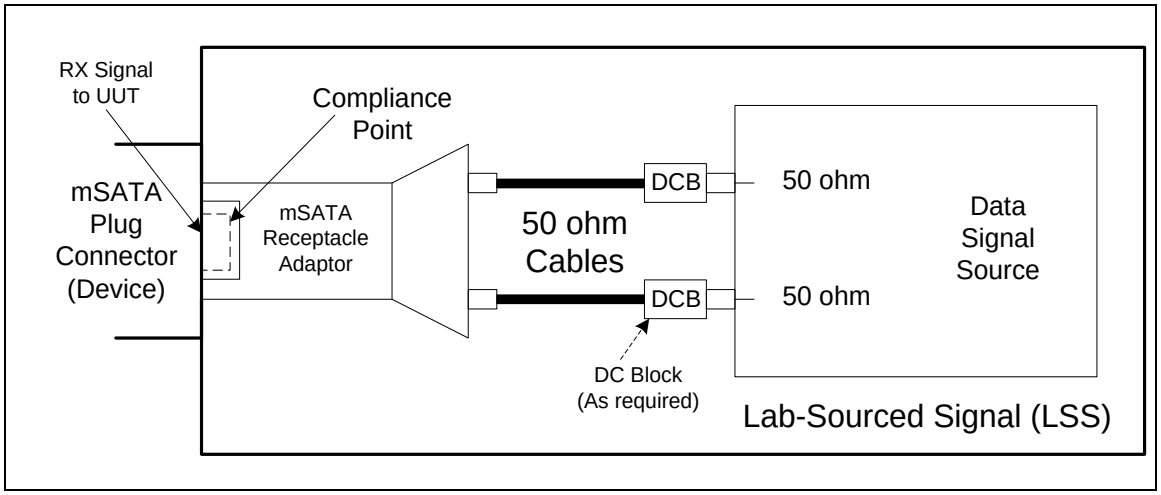


Figure 203 – Lab-Sourced Signal (LSS) for mSATA device

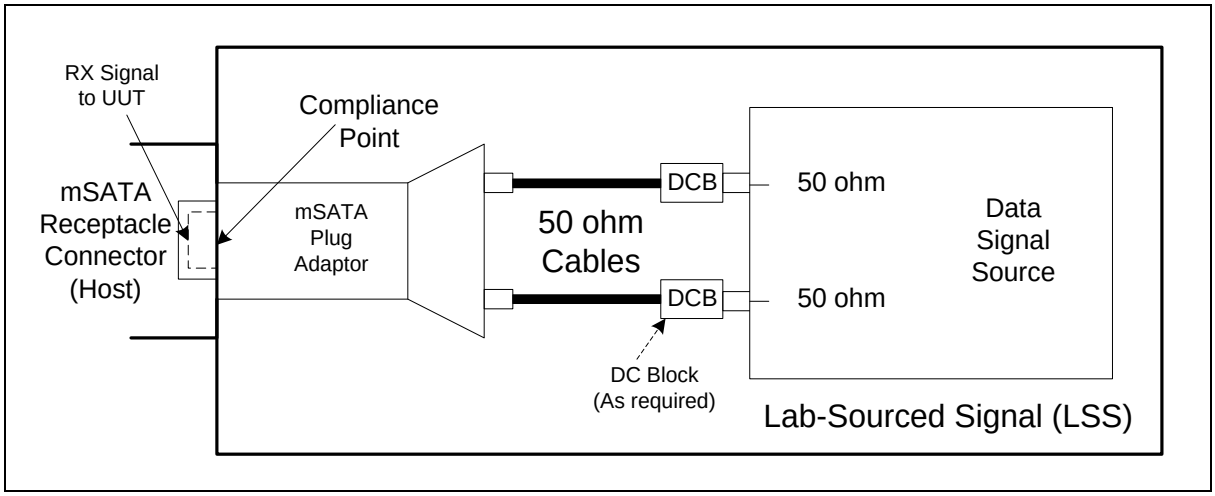


Figure 204 – Lab-Sourced Signal (LSS) for mSATA host

The Lab-Sourced Signal is a laboratory generated signal that is calibrated into an impedance matched load of 100 ohm differential and 25 ohm common mode and then applied to the Rx+ and Rx- signals of the Receiver Under Test. The load used to calibrate the Lab-Sourced Signal shall have an individual return loss greater than 20 dB over a bandwidth of 100 MHz to 5.0 GHz, and greater than 10 dB from 5 GHz to 8 GHz. During calibration, the characteristics of the Lab-Sourced

Signal shall comply with the specifications of Table 57. This signal is then applied to the Receiver Under Test the Frame Error Rate specifications of Table 52 shall be met.

7.6.2.2.3 SATA USM host Lab-Sourced Signal

Due to the direct connection of the SATA USM host, the host shall be mated to a plug connection as shown in Figure 205. The adaptor may require additional mechanical clearance to mate to the SATA USM host connector. See SFF publication INF-8280 for host connector details.

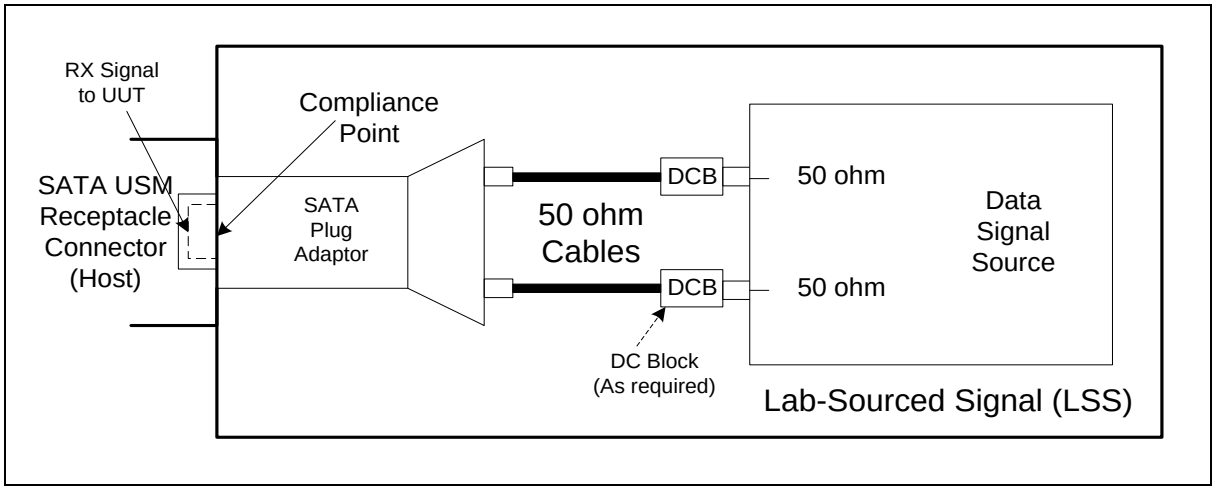


Figure 205 – Lab-Sourced Signal (LSS) for SATA USM host

The Lab-Sourced Signal is a laboratory generated signal that is calibrated into an impedance matched load of 100 ohm differential and 25 ohm common mode and then applied to the Rx+ and Rx - signals of the Receiver Under Test. The Lab-Sourced Signal represents the output of a device transmitter. In the case of Gen3i, the Gen3i CIC is used to calibrate the Lab-Sourced Signal as required, but removed from the signal path and applied to the UUT Rx. The Lab-Sourced Signal is signal is applied to the Receiver Under Test, the Frame Error Rate specifications of Table 52 shall be met.

7.6.2.3 SATA MicroSSD Lab-Sourced Signal

As described in SATA MicroSSD Lab-Load 7.6.2.1.4, to properly provide a SATA signaling to SATA MicroSSD devices and hosts, both types of SATA MicroSSD adaptors are required for device and host, as shown in Figure 206 and Figure 207, respectively. It is not possible to test the MicroSSD host if the SATA MicroSSD device is soldered in place.

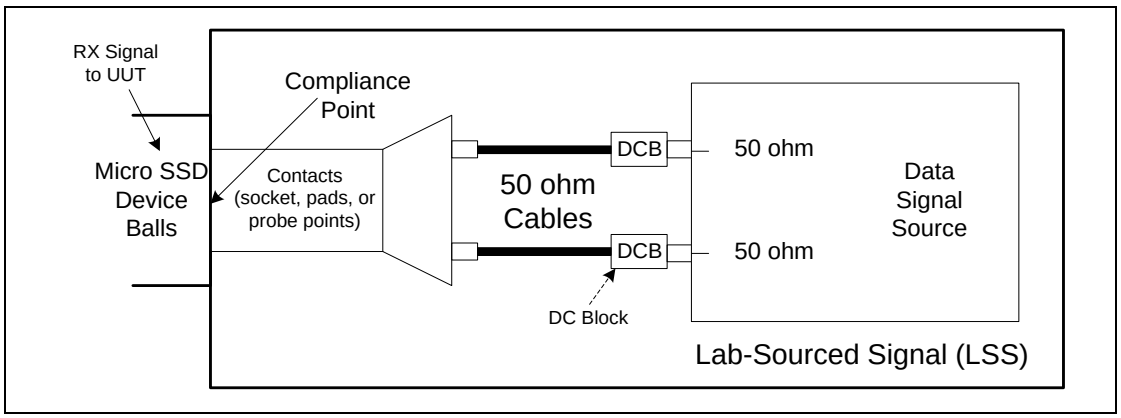


Figure 206 – Lab-Sourced Signal (LSS) for SATA MicroSSD device

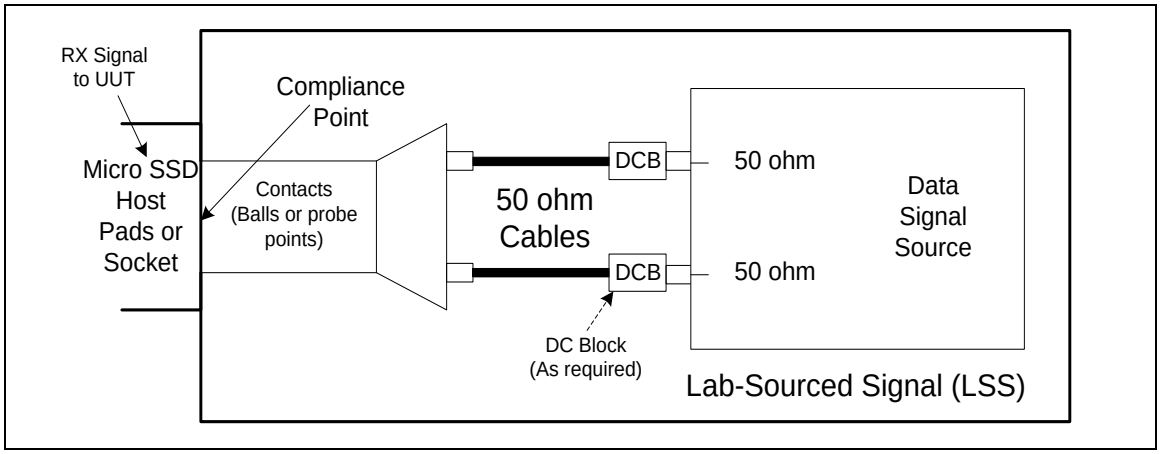


Figure 207 – Lab-Sourced Signal (LSS) for SATA MicroSSD host

The Lab-Sourced Signal is a laboratory generated signal that is calibrated into an impedance matched load of 100 ohm differential and 25 ohm common mode and then applied to the Rx+ and Rx- signals of the Receiver Under Test. The Lab-Sourced Signal represents the output of a device transmitter. In the case of Gen3i, the Gen3i CIC is used to calibrate the Lab-Sourced Signal as required, but removed from the signal path when applied to the UUT Rx. When the Lab-Sourced Signal is signal is applied to the Receiver Under Test, the Frame Error Rate specifications of Table 52 shall be met.

In a test environment, a test fixture and methodology may be used that makes it possible to deembed the signal attenuation effects of the text fixture, and thus measure compliance at the solder balls on the device and/or the socket/pads on the host.

7.6.2.4 M.2 Lab-Sourced Signal details

As described in Lab Load Details in 7.6.2.1, to properly provide a SATA signal into M.2 device, both female and male type M.2 adaptors are required for device and host, respectively as shown in Figure 208 and Figure 209.

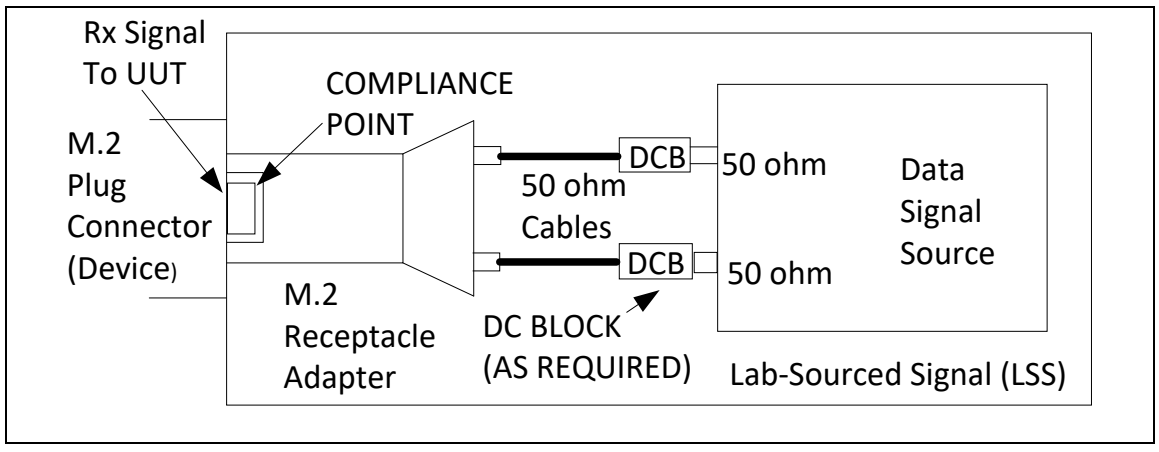


Figure 208 – Lab-Sourced Signal (LSS) for M.2 device

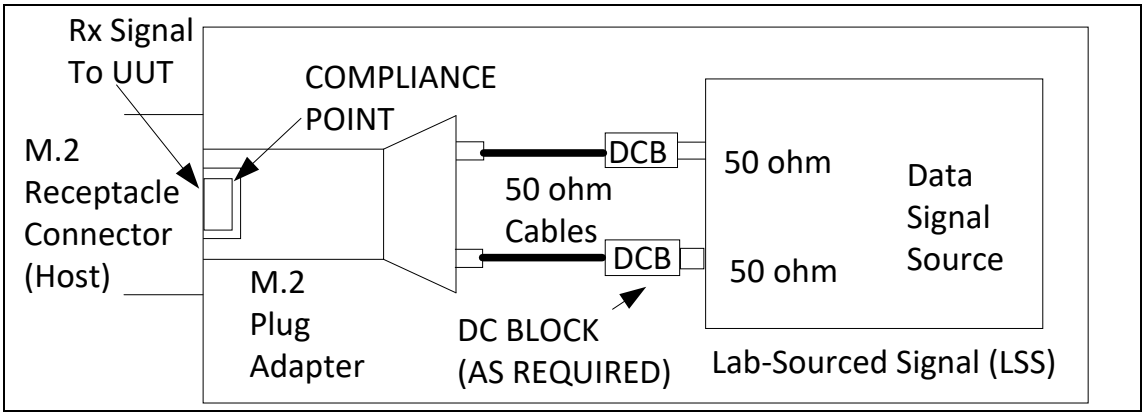


Figure 209 – Lab-Sourced Signal (LSS) for M.2 host

7.6.3 Frame error rate testing

7.6.3.1 Frame error rate testing overview

Frame error rate is the measure of link performance, a system level test. Since bit errors are ignored except during frames, Frame Error Rate testing is used as the method of measuring channel performance during system operation.

Serial ATA error detection at the frame level uses the CRC error detection mechanism, and respective reporting to the higher layer levels. Since all frames include a header and CRC, the calculation includes these overhead bytes in the Frame Error specification.

The bit error rate is a measurement of raw channel performance and is closely related to the Phy parameters. The 8b/10b encoding and SATA protocol complicate the measurement of bit error rate. A single bit error may result in several related errors occurring closely together that in turn may result in multiple bit-error counts. A character may have a single bit error in it that causes a code-violation error. A disparity error may occur on a following character, caused by the same single error. A single bit error has a high probability of causing a byte-wise error, or an 8b/10b code violation error, due to the 8b/10b encoding, thus a single bit error translates to 8 bits or 10 bits of error. A missing or an extra bit detected by the receiver translates into a series of errors that spans across multiple byte-boundaries until bit re-alignment via $ALIGN_P$ primitives.

Under the following condition, if BER = 0 then FER = 0 as well. BER = 0 therefore implies that FER = 0, and any product that achieves a receiver performance with zero BER, at the required confidence level, satisfies the FER requirement as well. The opposite is not necessarily the case.

7.6.3.2 Frame error rate patterns

Frame Error Rate patterns contain the elements of the Bit Error Rate test bit patterns and sequence of patterns, so as to thoroughly stress the serial interface in the system, while using the higher level CRC error detection and reporting from the lower protocol level layers to the Application layer.

The frame patterns shall be comprised of the set of the Composite Patterns, as defined in 7.4.5.4.7, but with the parameters extended so as to achieve the maximum frame length.

NOTE 28 -Note that the compliant patterns shown are the patterns that are expected on the wire.

If sent using a normal FIS payload mechanism the data within a FIS is scrambled. For the correct patterns to appear on the wire "pre-scrambling" needs to be performed so that the specified patterns appear on the link after payload scrambling is performed by the Transport layer.

7.6.3.3 Frame Error Rate measurements

The Frame Error Rate (FER) shall be measured and computed to be no greater than 8.200×10^{-8} at a 95 % confidence level if tested with any given 8b/10b encoded pattern, including the Frame Error Rate reference patterns as defined in 7.6.3.2. The Serial ATA CRC error detection mechanism is used to measure FER.

The Frame Error Rate is calculated based on the maximum size of a Data FIS, plus overhead for the FIS header and CRC Dwords. The Frame Error Rate assumes a target bit error rate of 10^{-12}.

$$FER = (8\ 192 + 8) \times 10 \times 10^{-12} = 8.200 \times 10^{-8}$$

The parameters of the reference patterns shall be extended to achieve the maximum frame length of 8 192 user payload bytes.

NOTE 29 - Note that the cited patterns appear on the wire.

7.6.3.4 Amount of data to transfer to achieve target confidence level (informative)

As this is a statistical process, there is a confidence level associated with each measurement that is related to the number of frames transferred.

EXAMPLE - One is recommended to only declare that the interface Frame Error Rate performance has been achieved with a confidence level for that given sample size, and Error-thresholds as shown in Table 74.

Table 74 – Frame Error Rate confidence levels versus sample size

Sample Size (Frames)	Number of Frame-Errors										
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1.22×10^7	63.21 %	26.42 %	8.03 %	1.90 %	0.37 %	0.06 %	0.01 %	<0.01 %	<0.01 %	<0.01 %	<0.01 %
1.22×10^8	>99.99 %	99.95 %	99.72 %	98.97 %	97.07 %	93.29 %	86.99 %	77.98 %	66.72 %	54.21 %	41.70 %

The sample size is taken as ten times the total number of frames transmitted for a given error rate.

$$\frac{1}{8.2 \times 10^{-8}} \times 10 = 1.22 \times 10^8$$

If a test is conducted where 1.22×10^8 frames are passed, the maximum number of frame errors to measure a Frame Error Rate of 8.200×10^{-8} with confidence level of > 95 % is four.

7.6.3.5 Bit error rate testing (informative)

7.6.3.5.1 Bit error rate testing overview

There are two basic classes of errors that affect the bit-error rate performance:

a) bit errors; and
b) burst errors.

In order to get a fair assessment of bit-error-rate performance, bit-errors, as well as burst errors, are considered separately. This is because a missing or an extra bit detected by the receiver

translates into a series of errors that spans across multiple byte boundaries until re-alignment via an alignment sequence. This series of errors are defined as burst errors.

Another type of byte-wise error exists when an entire byte is not received. As viewed by the higher-level protocol it appears as a loss of word synchronization. It causes a burst error whose span may be limited by higher layer protocol transmission conventions at the next alignment sequence.

Any of these errors may result in several related errors occurring closely together that in turn may result in multiple apparent bit-error events.

EXAMPLE - A character may have a single bit error in it that causes a code-violation error. A disparity error may occur on a following character, caused by the same single error.

All of these events eventually are recognized during the decoding process and result in a frame error.

NOTE 30 - Burst Error Rate measurements are not used for Compliance testing.

7.6.3.5.2 Bit error rate measurements

The Bit Error Rate, if measured and computed, byte-wise, should be no greater than 10^{-12} bit-errors if tested with the reference test patterns, as defined in 7.6.3.2.

Frame Error Rate measurements constitute the basis for the applicable test requirements for this specification**.** See 7.6.3 for an explanation of the relationship between FER and BER.

7.6.3.5.3 Amount of data to transfer to achieve target error rate

As this is a statistical process, there are confidence levels associated to each sample size.

EXAMPLE - One is recommended to only declare that the interface Bit Error Rate performance has been achieved with a confidence level of 95 % for that given sample size, and error thresholds as shown in Table 75.

Table 75 – Bit Error Rate confidence levels versus sample size

Sample Size (Bits)	Number of Bit-Error Events - Threshold										
	0	1	2	3	4	5	6	7	8	9	10
1.00×10^{12}	63.21 %	26.42 %	8.03 %	1.90 %	0.37 %	0.06 %	0.01 %	<0.01 %	<0.01 %	<0.01 %	<0.01 %
1.00×10^{13}	>99.99 %	99.95 %	99.72 %	98.97 %	97.07 %	93.29 %	86.99 %	77.98 %	66.72 %	54.21 %	41.70 %

If a test is conducted where 10^{13} bits are passed, the maximum number of error events to measure a Bit Error Rate of 10^{-12} with confidence level of > 95 % is four.

7.6.4 Measurement of differential voltage amplitudes (Gen1, Gen2)

7.6.4.1 Measurement of differential voltage amplitudes overview

The differential voltage amplitude, V_{diffTx}, shall be measured for bits in representative data patterns. It is necessary to use patterns that are DC balanced for this testing (otherwise, an offset is introduced that shifts the measured mean values).

The test setup shown in Figure 210 shows the connections.

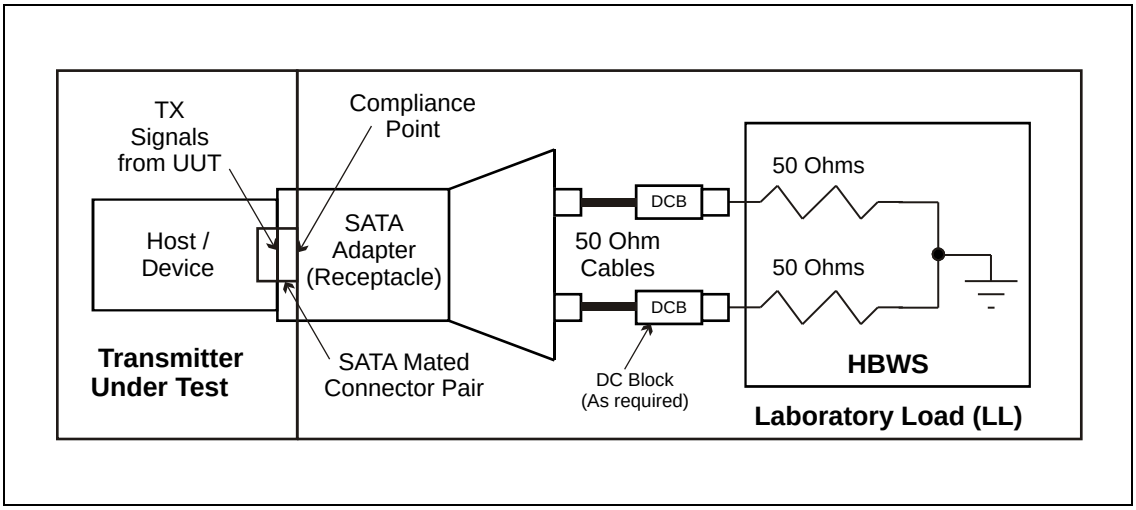


Figure 210 – Differential Voltage amplitude measurement

The transmitter under test sends the test pattern to a HBWS. The differential voltage waveform corresponding to one complete cycle of the N bit pattern has some unit intervals corresponding to zero bits and some unit intervals corresponding to ones bits. Figure 211 illustrates an example of a display on an equivalent time scope.

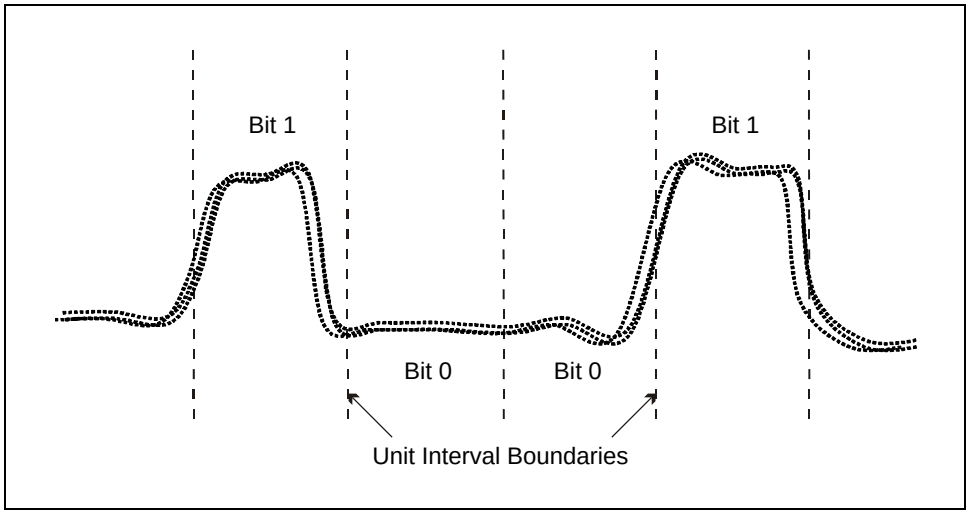


Figure 211 – Differential Voltage amplitude measurement pattern example

7.6.4.2 Testing for minimum differential voltage amplitude

7.6.4.2.1 Testing for minimum differential voltage amplitude overview

There are two separate procedures for this testing. However, each of the two procedures requires a common set of steps to be performed. These are labeled below as “Common Steps”. Following these steps are those that describe the rest of the procedure for one of two options (labeled as “Option 1” or “Option 2”).

7.6.4.2.2 Common steps

Common Step 1, transmitting a HFTP pattern, for a UI corresponding to a 1 bit, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

$\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode;

s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS; and

n is the number of samples that contribute to the histogram – this may also be read from the HBWS.

The inequality above is based on a requirement that enough samples are collected to define a confidence interval with at least 95 % probability and with a width no greater than 10 % of the sample mean.

Compute the following value:

$$UH = [\,\bar{x} - \frac{1.96 \times s}{\sqrt{n}}\,]$$

Common Step 2, transmitting a HFTP pattern, for a UI corresponding to a 0 bit, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

$\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode;

s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS; and

n is the number of samples that contribute to the histogram – this may also be read from the HBWS.

Compute the following value:

$$LH = [\,\bar{x} + \frac{1.96 \times s}{\sqrt{n}}\,]$$

Common Step 3, transmitting a MFTP pattern, for a UI corresponding to the second 1 bit of a string of two consecutive 1 bits, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

$\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode;

s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS;

n is the number of samples that contribute to the histogram – this may also be read from the HBWS.

Compute the following value:

$$UM = [\bar{x} - \frac{1.96 \times s}{\sqrt{n}}]$$

Common Step 4, transmitting a MFTP pattern, for a UI corresponding to the second 0 bit of a string of two consecutive 0 bits, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

$\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode

s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS

n is the number of samples that contribute to the histogram – this may also be read from the HBWS

Compute the following value:

$$LM = [\bar{x} + \frac{1.96 \times s}{\sqrt{n}}]$$

Common Step 5, compute the minimum of the following two differences:

$$DH = UH - LH$$

$$DM = UM - LM$$

Compute:

$$DHM = \min (DH, DM).$$

This value is used in the final step of each of the following two options.

7.6.4.2.3 Lone Bit Pattern measurements, option 1

If the test environment allows for the creation of a pattern trigger, the LBP pattern is used to make the following measurements. Continue the procedure from Common Step 5 above with the following steps for Option 1.

If the test environment does not allow for the creation of a pattern trigger, then continue the procedure from Common Step 5 above with the steps beginning with Option 2 Step 6 below.

Option 1 Step 6, transmitting a LBP pattern, for a UI corresponding to a lone 1 bit, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

$\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode;

s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS; and

n is the number of samples that contribute to the histogram – this may also be read from the HBWS.

Compute the following value:

$$A = [\,\bar{x} - \frac{1.96 \times s}{\sqrt{n}}\,]$$

Option 1 Step 7, transmitting a LBP pattern, for a UI corresponding to a lone 0 bit, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

$\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode;

s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS; and

n is the number of samples that contribute to the histogram – this may also be read from the HBWS.

Compute the following value:

$$B = [\bar{x} + \frac{1.96 \text{ x } s}{\sqrt{n}}]$$

Option 1 Step 8, from A and B obtained in steps 1 and 2, compute:

$$VTestLBP = A - B$$

Then take the minimum of VTestLBP and the previously computed DHM (from Common Step 5), that is:

$$VTest = \min (VTestLBP, DHM)$$

The test for minimum amplitude is passed if:

$$VTest > V_{diffTx}(min)$$

See Table 54 and Table 55 according to 7.4.2 for $V_{diffTx}(min)$. Otherwise, the test for minimum differential voltage amplitude has not been passed. If the test for minimum voltage amplitude is failed, the number of samples, n, is to be increased and the test shall be processed again for this larger number of samples. Failure to arrive at a value n that the test passes, means the requirement of the specification, for minimum differential voltage amplitude, has not been met.

7.6.4.2.4 Approximation to Lone Bit Pattern measurements, option 2

To test for minimum differential voltage amplitude without the ability to create a pattern trigger, continue the procedure from Common Step 5 above with the following steps for option 2.

Option 2 Step 6, transmitting a LFTP pattern, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI of the first 1 bit that follows either a string of three preceding 0 bits or a string of four preceding 0 bits. It is required that the histogram samples be the union of the samples collected for both cases. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

$\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode;

s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS; and

n is the number of samples that contribute to the histogram – this may also be read from the HBWS.

Compute the following value:

$$A = [\bar{x} - \frac{1.96 \times s}{\sqrt{n}}]$$

Option 2 Step 7, transmitting a LFTP pattern, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI of the first 0 bit that follows either a string of three preceding 1 bits or a string of four preceding 1 bits. It is required that the histogram samples be the union of the samples collected for both cases. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

$\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode;

s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS; and

n is the number of samples that contribute to the histogram – this may also be read from the HBWS.

Compute the following value:

$$B = [\bar{x} + \frac{1.96 \times s}{\sqrt{n}}]$$

Option 2 Step 8, transmitting a LFTP pattern, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI of the last 1 bit in a string of three or four 1 bits. It is required that the histogram samples be the union of the samples collected for both cases. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

$\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode;

s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS; and

n is the number of samples that contribute to the histogram – this may also be read from the HBWS.

Call the mean:

$$C = \bar{x}$$

Option 2 Step 9, transmitting a LFTP pattern, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI of the last 0 bit in a string of three or four 0 bits. It is required that the histogram samples be the union of the samples collected for both cases. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

$\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode;

s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS; and

n is the number of samples that contribute to the histogram – this may also be read from the HBWS.

Call the mean:

$$D = \bar{x}$$

Option 2 Step 10, transmitting a LFTP pattern, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI of the last 1 bit in a string of four 1 bits. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

$\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode;

s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS; and

n is the number of samples that contribute to the histogram – this may also be read from the HBWS.

Call the mean:

$$E = \bar{x}$$

Option 2 Step 11, transmitting a LFTP pattern, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI of the last 0 bit in a string of four 0 bits. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

$\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode;

s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS; and

n is the number of samples that contribute to the histogram – this may also be read from the HBWS.

Call the mean:

$$F = \bar{x}$$

Option 2 Step 12, from A and B obtained in steps 1 and 2, compute:

$$VTestAPP = (A + C + F) – (B + D + E)$$

Then take the minimum of VTestAPP and the previously computed DHM, that is:

$$VTest = \min (VTestAPP, DHM)$$

The test for minimum amplitude is passed if:

$$VTest > V_{diffTx}(min)$$

See Table 54 and Table 55, according to 7.4.2 for V_{diffTx}(min). Otherwise, the test for minimum differential voltage amplitude has not been passed. If the test for minimum voltage amplitude is failed, the number of samples, n, is to be increased and the test shall be processed again for this larger number of samples. Failure to arrive at a value n that the test passes, means the requirement of the specification for minimum differential voltage amplitude has not been met.

Figure 212 illustrates the locations of the sections of the LFTP as displayed on a scope that the measurements of the values for A, B, C, D, E, and F (see steps 1 to 6 above) are to be made.

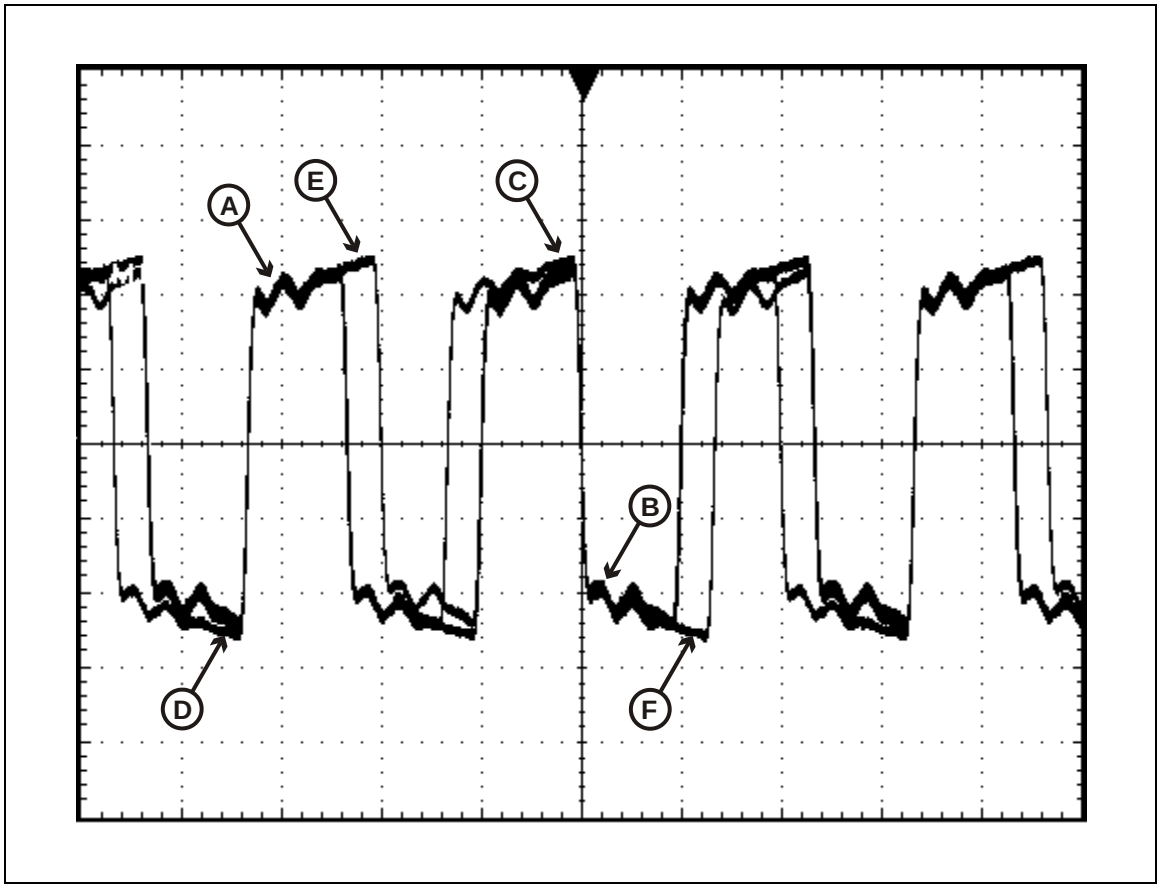


Figure 212 – LFTP Pattern on High BW Scope (HBWS)

7.6.4.3 Test for maximum differential voltage amplitudes

To test for maximum differential voltage amplitude for a given data pattern, perform the following steps using the LFTP and the MFTP as the data patterns.

The waveform sections to be examined are defined as:

a) High Test UI:
   A) for the LFTP, use the fourth 1 bit in a string of four 1 bits; and
   B) for the MFTP, use the first 1 bit in a string of two 1 bits;

and

b) Low Test UI:
   A) for the LFTP, use the fourth 0 bit in a string of four 0 bits; and
   B) for the MFTP, use the first 0 bit in a string of two 0 bits.

Step 1, for the High Test UI, construct a histogram in the waveform epoch [0.0 UI, 1.0 UI] for the UI. Position the upper edge of the histogram window at VU mV where:

$$VU = \frac{1}{2} V_{diffTx}(\max)$$

See Table 54 and Table 55, according to 7.4.2 for $V_{diffTx}(max)$.

Position the lower edge of the histogram window at 0 mV. Let the histogram acquire hits for a fixed time duration, T, such that the number of hits acquired is at least 10 000. Record the number of histogram hits as NU. This histogram may be based on data stored in the waveform database.

For the same High Test UI, construct a histogram in the waveform epoch [0.0 UI, 1.0 UI]. Position the upper edge of the histogram window at VU + 300 mV. Position the lower edge of the histogram window at VU mV. Record the number of histogram hits as nu.

Step 2, for the Low Test UI, construct a histogram in the waveform epoch [0.0 UI, 1.0 UI]. Position the upper edge of the histogram window at 0 mV. Position the lower edge of the histogram window at VL mV where:

$$VL = -\frac{1}{2}V_{diffTx}(\max)$$

Let the histogram acquire hits for the same fixed time duration, T, as used in step 1. (In practice, using the same waveform database as that collected in Step 1 insures that the same time duration is examined.) Record the number of histogram hits as NL.

For the same Low Test UI, construct a histogram in the waveform epoch [0.0 UI, 1.0 UI]. Position the lower edge of the histogram window at VL – 300 mV. Position the upper edge of the histogram window at VL mV. Record the number of histogram hits as nl.

Step 3, compute the values:

$$pu = \frac{nu}{nu + NU}$$

$$pl = \frac{nl}{nl + NL}$$

NOTE 31 - There are two values of pu and pl computed; one for the use of the LFTP and one for the use of the MFTP.

Step 4, the test for maximum amplitude is passed if:

$$pu < 0.05$$

and

$$pl < 0.05$$

NOTE 32 - Since there are two values of pl and pu, the test needs to be applied to each pair.

Otherwise, the test for maximum differential voltage amplitude has not been passed.

7.6.5 Measurement of differential voltage amplitudes (Gen3i, Gen3u)

7.6.5.1 Measurement of differential voltage amplitudes overview

The amplitude measurement of differential signals for Gen3i uses different methods for the maximum amplitude and the minimum amplitude compliance tests. The maximum amplitude is a peak-to-peak value measured at the Tx compliance point into a lab-load. This limits the magnitude of signals present in the interface. The minimum amplitude is a measurement of the minimum eye opening, using the specified method, after the Gen3i CIC, into a lab-load. This provides a minimum signal level for the receiver, measured in a manner that is representative of how a typical receiver processes the signal.

Achieving both the maximum and minimum differential amplitude compliance limits as described in Table 54 shall be required, using the same transmitter settings for both tests.

The same methods and patterns are used for setting up the Lab-Sourced Signal for Receiver Tolerance Testing, with the compliance limits specified in Table 57.

The Gen3u transmitted signal maximum amplitude is a peak-to-peak value measured at the Tx compliance point into the lab-load. The minimum amplitude is a measurement of the minimum eye opening, using the specified method, into a lab-load. The Gen3i CIC is not used for the Gen3u transmitted signal minimum amplitude. The measured values shall comply with the limits specified in Table 55.

The same methods and patterns are used for setting up the Lab-Sourced Signal for Gen3u receiver tolerance testing, with the compliance limits specified in Table 58. The Gen3i CIC is used to calibrate the Lab-Sourced Signal for Gen3u receiver tolerance testing, however, the Gen3i CIC losses are removed from the Lab-Sourced Signal prior to testing the Gen3u receiver under test for compliance.

7.6.5.2 Maximum differential voltage amplitude (Gen3i, Gen3u)

The maximum differential amplitude shall be measured at the Tx Compliance point into a lab-load. Figure 214 shows a drawing of this test connection. A Gen3 MFTP shall be used for this compliance measurement, although it is possible that with other patterns and signal path characteristics, additional peak-to-peak maximum amplitude values maybe present in the actual system. The MFTP should contain emphasis due to its run length, if the transmitter supports this signal conditioning, and allows for simple edge triggering for the signal capture.

The maximum amplitude is defined as the peak to peak value of the average of 500 waveforms measured over a time span of 4 Gen3 UI, using the HBWS.

7.6.5.3 Minimum differential voltage amplitude (Gen3i, Gen3u)

7.6.5.3.1 Minimum differential voltage amplitude overview

The Gen3i minimum Tx differential amplitude shall be measured with and without the Gen3i CIC as defined in 7.4.8 terminated into the lab-load. Figure 214, Figure 215 and Figure 216 shows drawings of these test connections. A Gen3 LBP shall be used for this compliance measurement, although it is possible that with other patterns and signal path characteristics, lower amplitudes may be present in the actual system.

The Gen3u transmitted minimum Tx differential amplitude shall be measured terminated into the lab-load. Figure 197 shows a drawing of this test connection. The Gen3i CIC is not used for this measurement. The measured value shall comply with the limits specified in Table 55.

The minimum amplitude is defined as the vertical eye height on a population of at least 5×10^6 UI of data measured at the minimum voltage measurement interval of the UI using the Gen3i

Reference Clock JTF as defined in 7.5.3. When calibrating a receiver stressed signal, an Explicit Clock shall be used instead of the JTF.

All test equipment (e.g., HBWS), requires a minimum signal amplitude to be able to measure the eye height to a population of 5×10^6 UI. This level varies with instrumentation type, hardware and software. This minimum required instrumentation amplitude introduces errors in the reported minimum amplitude measurement value. This error results in the minimum amplitude reported by the test equipment to be smaller than the actual signal minimum amplitude. This instrumentation error shall be corrected for, to determine the actual minimum amplitude value using recommended methods provided by the test equipment manufacturer. If no such recommended correction procedure is available for one piece of test equipment, alternate test equipment may be selected. Instrumentation performing this measurement shall be traceable to industry standards setting origination in the area of performing amplitude measurements (e.g., National Institute of Standards and Technology (NIST)) within the USA. HBWS Instruments shall be calibrated and be traceable to ISO/IEC 17025:2005 specifications.

7.6.5.3.2 Equipment sensitivity correction method (informative)

An alternate correction method is able to be used in the case of unsupported test equipment. The amplitude of a low noise Lab Source Gen3 MFTP test pattern with the fastest allowed Gen3i rise and fall times is reduced in several steps using passive calibrated attenuators. If the reported amplitude is plotted on the y-axis and the ideal amplitude calculated using the calibrated attenuators and source is plotted on the x-axis, the y-axis intercept represents the theoretical reported amplitude for a zero amplitude input. A linear curve fit to the measured data is able to extrapolate the measured data to the y-axis intercept. (a negative value) The absolute value of this y-axis intercept is then added as a positive number to the instrument reported minimum Eye Opening values to correct for this error term. Since this is a statistical measurement and the test equipment may contain significant random amplitude variations. This correction method is able to be in error since it does not convolve the random amplitude variation sources, but it reduces the error magnitude below an uncorrected measurement. This may result in a possible over correction of the instrumentation error term.

7.6.5.4 Minimum differential amplitude eye height (Gen3i)

7.6.5.4.1 Minimum differential amplitude eye height overview

The 4 sigma Eye Diagram shall be constructed from a population of at least 5×10^6 UI with either a JTF based PLL for transmitter testing, or an Explicit Clock for Lab-Sourced Signal calibration. The Explicit Clock, or equivalent, from an instrument grade generator shall be used during receive calibration.

7.6.5.4.2 Receiver test calibration levels and method details

$V_{diffRxdevice}$,Rx Differential Device Input Voltage for drive side receiver (Eye Height + CIC + Explicit Clock)
$V_{diffRxhost}$,Rx Differential Host Input Voltage for the host side receiver (Eye Height + CIC + Explicit Clock)

The Eye Height shall be evaluated at UI_{VminRx}, Rx Minimum Voltage Measurement Interval.

7.6.5.4.3 Transmitter test levels and method details

$V_{diffTxdevice}$,Tx Differential Device Output Voltage for drive side transmitter minimum (Eye Height + CIC + JTF PLL)

$V_{diffTxhost}$,Tx Differential Host Output Voltage for the host side transmitter minimum (Eye Height + CIC + JTF PLL)

The Eye Height shall be evaluated at UI_{VminTx}, Tx Minimum Voltage Measurement Interval.

7.6.6 Rise and fall times

The rise and fall times of the waveform under test are measured from the 20 % threshold to the 80 % thresholdfrom the reference levels (see Figure 213). High Reference level of the waveform under test is the “mode” of the top portion while the Low Reference level is the “mode” of the bottom portion. Mode is measured using Statistical Methods of the desired waveform and is the most common value of the probability density function. The minimum time span of the analysis zone for measuring the mode amplitude shall be 8 UI.

Therefore, Rise Time = X2 - X1; where X2 is the mean horizontal time value corresponding to 80 % of the distance between the Low and High value and X1 is the mean horizontal time value position corresponding to 20 % of the distance between the Low and High value.

And Fall Time = X1 - X2; where X1 is the mean horizontal time value corresponding to 20 % of the distance between the Low and High value and X2 is the mean horizontal time value position corresponding to 80 % of the distance between the Low and High value.

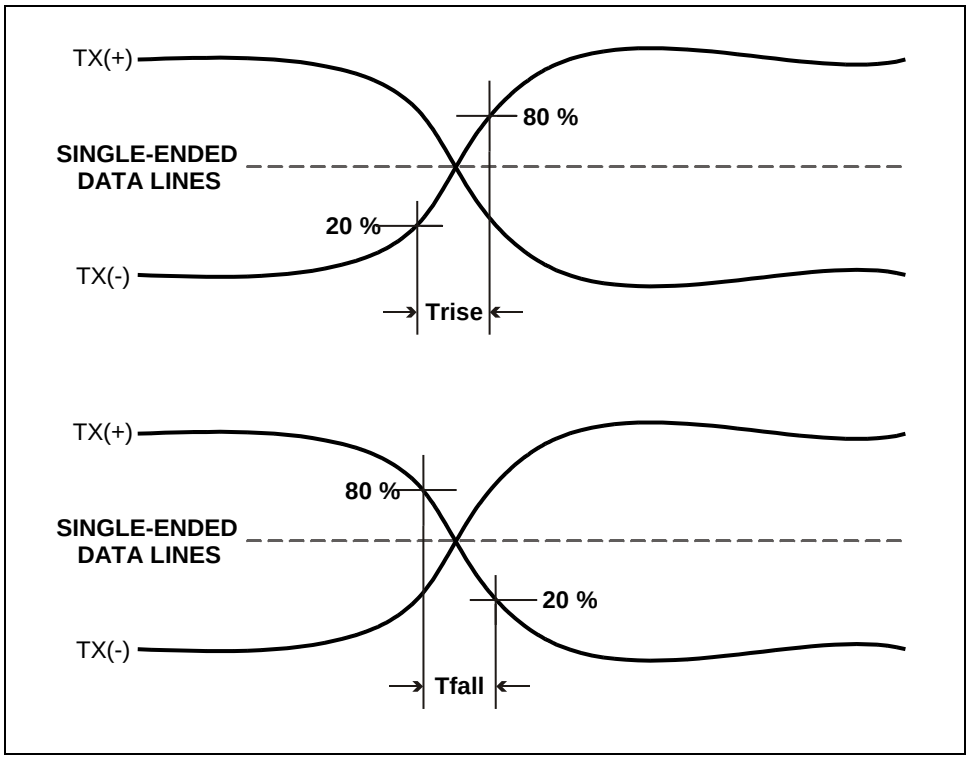


Figure 213 – Single ended rise and fall time

Rise and Fall values are measured using the LFTP Pattern previously defined, for all Gen1 and Gen2 Electrical Specifications. The average rise time of all rising edges and the separate average fall time of all falling edges within the 8 UI analysis zone shall both meet the required rise and fall time compliance limits.

For Gen3i and Gen3u, the Rise and Fall time values, measured from the 20 % threshold to the 80 % threshod, are measured using only the Gen3 LFTP. This minimizes errors in determining the 0 % and 100 % reference levels using the Mode Amplitude measurement method. The analysis zone of the measurement shall be made over a minimum time length of 8 UI. This is a lab-load measurement. The Rise and Fall time compliance limits, for the differential Tx test pattern as described in Table 54 and Table 55. The average Rise time of all rising edges and the separate average Fall time of all falling edges within the analysis zone shall meet the Rise and Fall time

compliance limits respectively. The Rise and Fall time compliance limits for the differential Data Signal Source (see Figure 231), for Receiver Tolerance testing, are also set with this method and pattern. The compliance limits for the Lab-Sourced Signal as described in Table 57 and Table 58**.**

The rise and fall times for transmitter differential buffer lines are measured with the load fixture shown in Figure 214. The rise and fall times shall be measured with an HBWS.

7.6.7 Transmitter amplitude

7.6.7.1 Transmitter amplitude overview

The transmitter amplitude values specified in Table 54 and Table 55 refer to the output signal from the UUT at the mated connector into a lab-load (for Gen1i, Gen1m, Gen1u, Gen2i, Gen2m, Gen2u, Gen3i, and Gen3u), or from the UUT through a CIC into a Laboratory Load (for Gen3i only). The signals are not specified while attached to a system cable or backplane.

7.6.7.2 Transmitter amplitude (Gen1 and Gen2)

Transmitter minimum amplitude is measured with each of three waveforms:

a) HFTP;
b) MFTP; and
c) the Lone Bit Pattern (LBP).

Amplitude specifications shall be met according to the measurement method as defined in 7.6.4.

The minimum amplitude value is measured during the Tx minimum voltage measurement interval defined in Table 54 and Table 55. The Reference Clock (see 7.5.3) defines the ideal (zero jitter) zero crossing times. The maximum amplitude is measured with the measurement method outlined according to 7.6.4.3 using waveforms LFTP and MFTP.

The transmit DC offset voltage (for Gen1i only) should be measured with the setup in Figure 214. The HBWS is measuring a DC voltage and the DC blocks (DCB) shall not be present.

Figure 214 shows the test setup for measuring transmitter amplitude. The HBWS is the standard for measuring amplitude. The losses in the test connections may be significant so it is prudent to minimize and estimate these.

Several methods may be used to estimate the cabling losses:

a) the first is to use two cables of different lengths and compare the losses of each;
b) the second is to rely on published data for the cables; or
c) the third is to obtain a separate means for measuring the cable loss (e.g., characterization with a network analyzer or power meter).

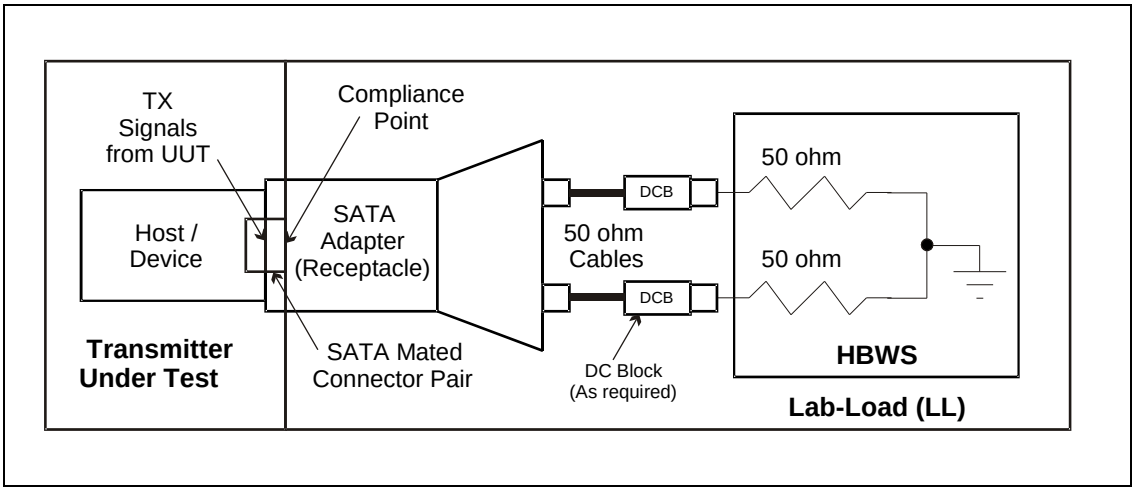


Figure 214 – Transmit amplitude test with Lab-Load (LL)

This specification describes transmitter levels in terms of voltage while driving a test load of 100 ohm differential (i.e., lab-load) and 50 ohm single ended to ground. To relate the specified maximum levels to the maximum values seen in a system requires a calculation. For an example of this calculation see 7.6.8.

7.6.7.3 Transmitter amplitude (Gen3i, Gen3u)

Transmitter minimum amplitude is measured with the Lone Bit Pattern (LBP). Amplitude specifications shall be met according to the measurement method as defined in 7.6.5.

The minimum amplitude value is measured during the Tx minimum voltage measurement interval defined in Table 54 and Table 55. The Reference Clock (see 7.5.3) defines the ideal (zero jitter) zero crossing times. The maximum amplitude is measured with the measurement method according to 7.6.5 using the MFTP waveform.

Figure 214 and if a CIC is used Figure 215 or Figure 216 show test setups for measuring transmitter amplitude. See 7.6.7.2 for suggestions on compensating for losses in the test connections.

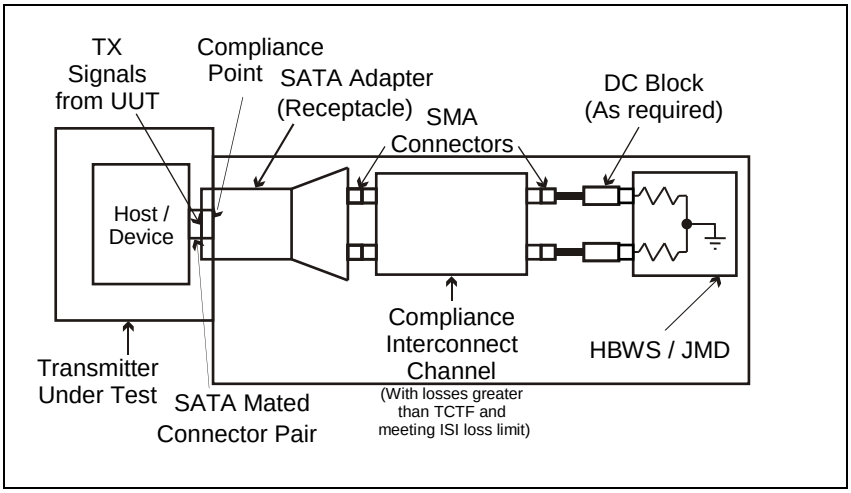


Figure 215 – Transmit amplitude test with CIC

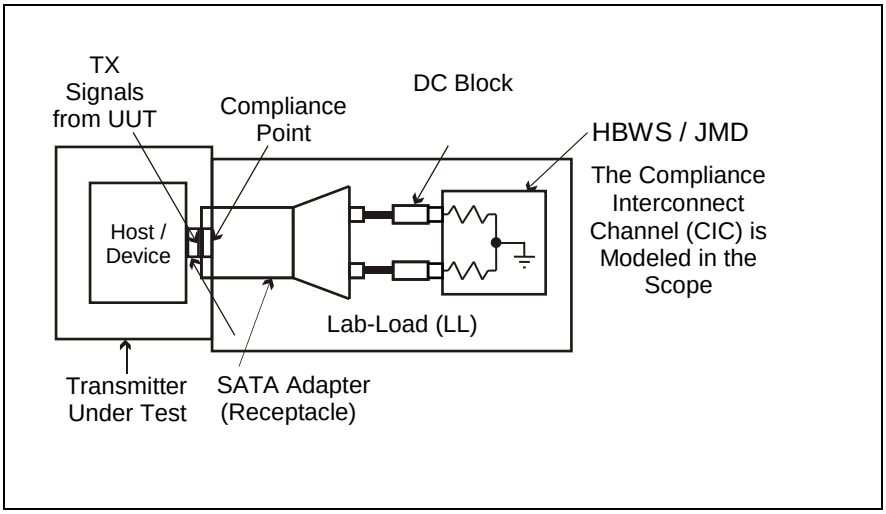


Figure 216 – Transmit amplitude test with simulated CIC

7.6.8 Receive amplitude

This section describes setting the receive amplitude, a test condition common to many tests. The proper operation of the receiver is its ability to receive a signal. An example of this testing as defined in 7.6.13. The values as specified in Table 57 and Table 58 refer to the input signal from any signal source as measured at the device under test using a Laboratory Load (see Figure 218).

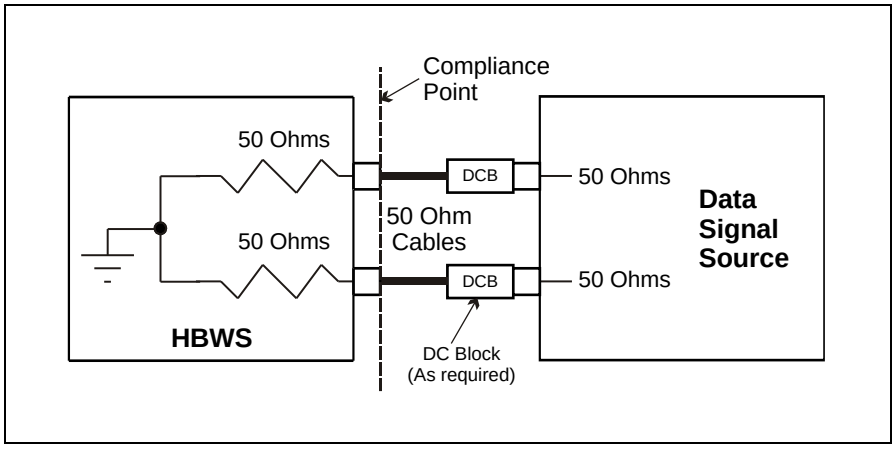


Figure 217 – Receiver amplitude test, setting levels

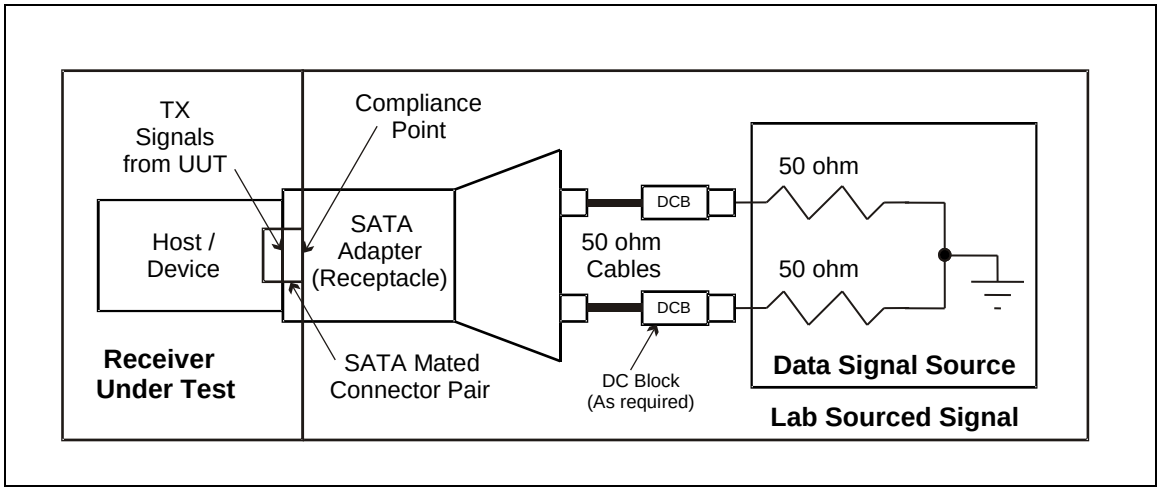


Figure 218 – Receiver amplitude test

Figure 217 shows an example to account for loss in the cabling and error in the signal source for receiver level testing. The loss in the SATA adapter is not accounted for here, but may be separately measured. The HBWS is used as the standard for amplitude while setting the levels for testing receivers. Equivalent methods to account for loss in the cabling are acceptable.

This specification describes receiver levels in terms of voltage driven from a differential source of 100 ohm impedance. A calculation is required to relate the specified maximum receiver level to the maximum receiver level in a system. The maximum receiver level is set at a HBWS by driving with a signal source impedance of 100 ohm. With the signal generator level set, it is then applied to the receiver under test. The voltage actually seen at the receiver inputs depends on the input impedance of the receiver. The maximum voltage at the receiver occurs if the receiver input impedance is at its maximum value (see Figure 219).

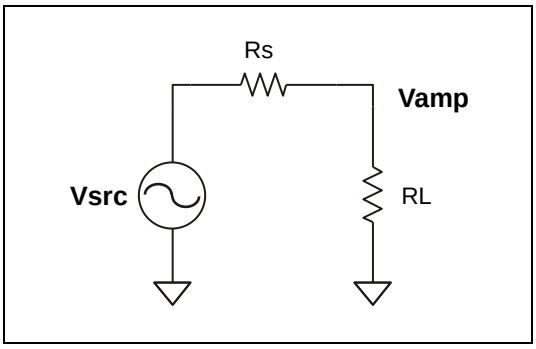


Figure 219 – Voltage at receiver input

$$V_{AMP} = \frac{V_{SRC}}{\left(1+\frac{R_S}{R_L}\right)} = \frac{V_A\left(1+\frac{R_S}{R_{LL}}\right)}{\left(1+\frac{R_S}{R_L}\right)}$$

The values of the receiver and transmitter resistor termination are set by the return loss specification at low frequency. Return loss is given by the following equation:

$$RL = -20 \log\left[\left|\frac{Z - Z_0}{Z + Z_0}\right|\right]$$

And solving this for the resistance, the real part of the impedance gives two solutions

$$R = Z_0 \frac{1-10^{-RL/20}}{10^{-RL/20}+1} = (100)\frac{1-10^{-18/20}}{10^{-18/20}+1} = 77.64$$

$$R = Z_0 \frac{10^{-RL/20}+1}{1-10^{-RL/20}} = (100)\frac{10^{-18/20}+1}{1-10^{-18/20}} = 128.8$$

$$R = Z_0 \frac{1-10^{-RL/20}}{10^{-RL/20}+1} = (100)\frac{1-10^{-14/20}}{10^{-14/20}+1} = 66.73$$

$$R = Z_0 \frac{10^{-RL/20}+1}{1-10^{-RL/20}} = (100)\frac{10^{-14/20}+1}{1-10^{-14/20}} = 149.9$$

The highest amplitude that may be seen at the receiver occurs if the receiver input resistance is highest.

$$V_{AMP} = \frac{0.7\left(1 + \frac{100}{100}\right)}{\left(1 + \frac{100}{128.8}\right)} = 0.788\ 1$$

The lowest amplitude at the receiver occurs if the receiver input resistance is lowest.

$$V_{AMP} = \frac{0.4\left(1 + \frac{100}{100}\right)}{\left(1 + \frac{100}{77.64}\right)} = 0.349\ 7$$

7.6.9 Long term frequency accuracy

There are several considerations for choosing instruments to measure long-term frequency accuracy. The long-term frequency accuracy of the instrument time base needs to be significantly better than the 350 ppm limit in this specification; many oscilloscopes do not have this frequency accuracy.

A method to measure the long-term frequency accuracy is to use a frequency counter. The test setup shown in Figure 220 shows the connections. The transmitter under test sends a HFTP (D10.2) signal to the frequency counter. The signal shall not have SSC modulation. The frequency counter should have a gating period set long enough to reduce the effects of noise; this may be done by setting the counter resolution to 10 Hz or better (350 ppm at 1.5 GHz is 525 kHz). The counter reads the long-term frequency of the transmitter; the accuracy is a percentage.

While SSC is present, long term frequency accuracy specification is not applicable, instead the SSC profile is measured (see 7.6.13).

There are other instruments that contain a frequency counter with an accuracy significantly better than 350 ppm.

EXAMPLE - Some BERT equipment has a frequency counter on the clock input.

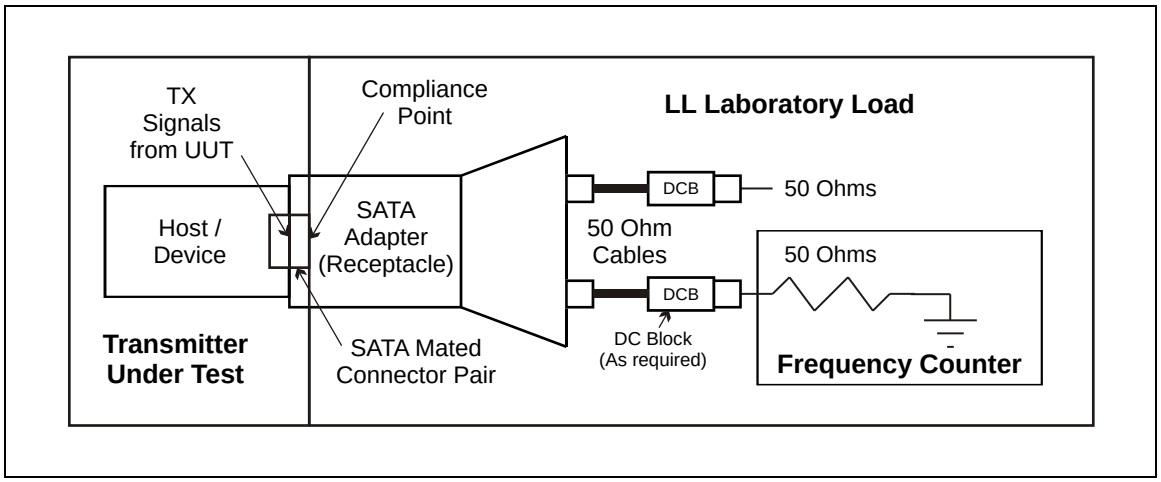


Figure 220 – Tx long term frequency measurement

7.6.10 Jitter measurements

7.6.10.1 Jitter measurements overview

The causes of jitter are categorized into random sources (RJ) and deterministic sources (DJ). Although the TJ is the convolution of the probability density functions for all the independent jitter sources, this specification defines the random jitter as Gaussian and the total jitter (at a BER of 10^{-12}) as the deterministic jitter plus 14 times the random jitter. The TJ specifications of Table 54, Table 55, Table 57, and Table 58 were chosen at a targeted BER of 10^{-12}. In Table 54 and Table 55, Gen3i and Gen3u Tx jitter is specified by providing limits for TJ(10^{-12}) and TJ(10^{-6}). The BERT scan method as described in 7.6.10.2 is the only method that measures the actual TJ and is used as the reference for all TJ estimation methods. The method for estimating TJ is unique to each measurement instrument.

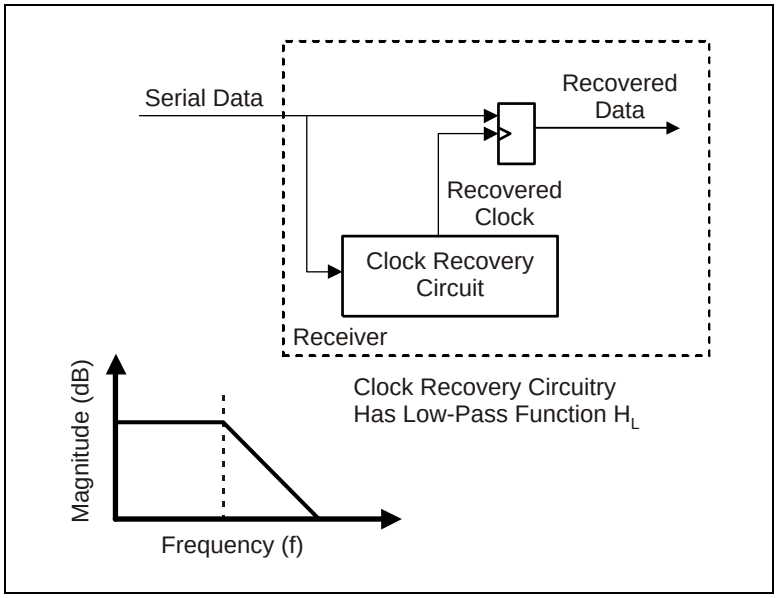


Figure 221 – Receiver model for jitter

The jitter measurement methodology is defined as a clock to data jitter measurement. Figure 221 shows a block diagram of a deserializer input. The serial data is split into two paths. One path feeds clock recovery circuitry that becomes the reference signal used to latch the data bits of the serial data stream. This clock recovery circuitry has a low pass transfer function H_L. This low pass function is the CLTF of the PLL or clock recovery circuit. The jitter seen by the receiver is the time difference of the recovered clock edge to the data edge position. This time difference function is shown in Figure 222. The resulting jitter seen by the receiver has a high pass function H_H shown in Figure 223. This high pass function is the Jitter Transfer Function (JTF) of the system. This defines the measurement function required by all jitter measurement methodologies. The required characteristics for the JTF and the CLTF corner frequency f_c as defined in 7.5.3. In the case of a JMD, the JTF may be simply viewed as the ratio of the reported jitter to the applied jitter, for a sinusoidal PJ input.

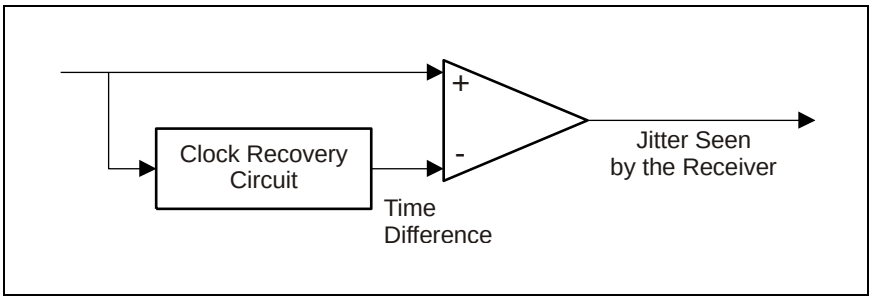


Figure 222 – Jitter at receiver

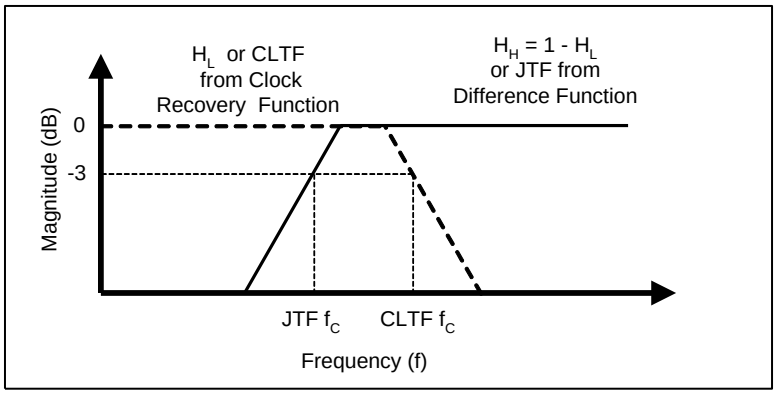


Figure 223 – Jitter at receiver, high pass function

This JTF (H_H in Figure 223) mimics the receiver's ability to track lower frequency jitter components (wander, SSC) and not include them in the jitter measurement. This measurement methodology enables any measurement instrument to accurately measure the jitter seen by a receiver and produce measurements that correlate from measurement instrument to measurement instrument.

It should be noted that the corner frequency of the JTF is not the corner frequency of the clock recovery CLTF. This may not be obvious until one considers the phase shift caused by the clock recovery circuit. In general the vector sum $H_L(f) + H_H(f) = 1$. All quantities consist of changing magnitude and phase as a function of frequency. This accounts for differences in corner frequencies and peaking in the two frequency dependant functions.

Figure 224 shows more detail into how the JTF and CLTF relate to the jitter that is able to be applied to a receiver. The subfigure A) represents a generic control system block diagram for a feedback loop based clock recovery system. Subfigure B) translates the same complex variables to the combined system of the clock recovery circuit and the time difference function. It is possible to be seen that E(s) is the jitter seen by the receiver, as well as being the error signal in the clock recovery circuit. Subfigure C) provides the defining equations for the clock recovery circuit CLTF and the combined system JTF function.

Both the CLTF and the JTF are uniquely defined by the open loop transfer function G(s). Defining a CLTF does not uniquely define the G(s) and subsequently the JTF due to the level of cancellation of G(s) in the numerator and denominator of the CLTF especially if G(s) is much greater than 1 that is necessary for jitter tracking by the clock recovery circuit. This is the rational for directly specifying the JTF rather than the CLTF of the clock recovery circuit. If the JTF of a JMD meets the requirements specified, the JMD reported jitter levels should closely represent the jitter applied to the receiver in this reference design.

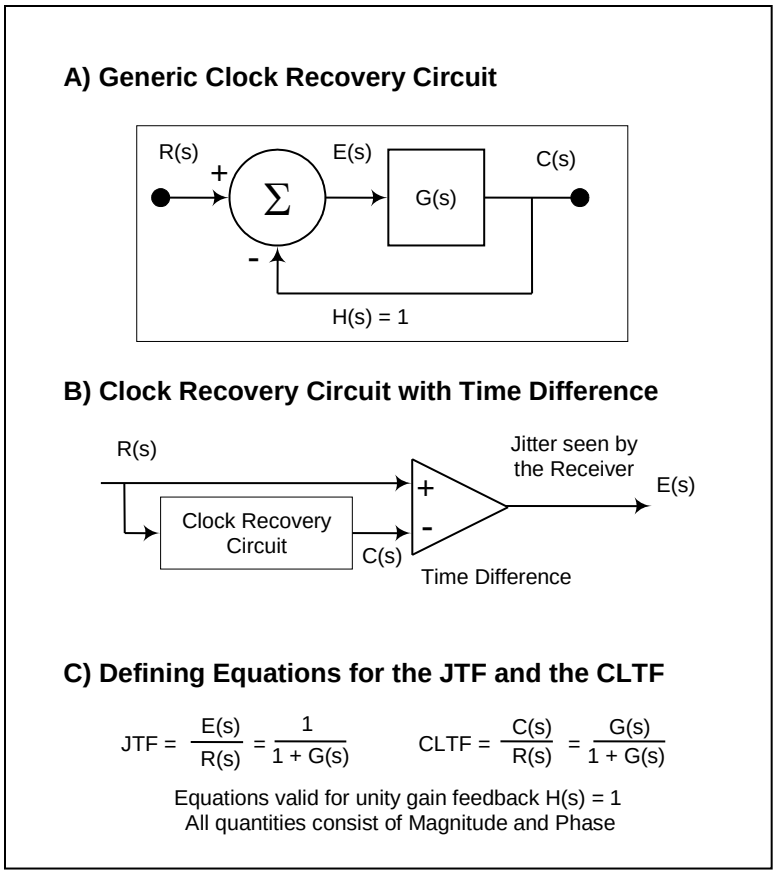


Figure 224 – JTF and CLTF definition

7.6.10.2 Jitter measurements with a bit error rate tester (BERT)

Most instruments used to measure jitter are unable to directly measure TJ at very low bit error rates like 10^{-12} due to the time it takes to capture sufficient transitions for a statistically significant direct measurement. Instead, these instruments capture a smaller sample size and extrapolate TJ using complex, and in some cases proprietary, algorithms. The determination of TJ through extrapolation may greatly reduce the amount of time required to measure jitter but experience has shown different extrapolation-based methods may produce different results. An alternate method to measure TJ is through the use of a BERT scan method. Since a BERT scan may directly measure jitter to 10^{-12} and even lower rates in a reasonable amount of time, it also provides a means of reconciling any differences in the extrapolated value of TJ.

A BERT scan method utilizes the variable clock-to-data timing path available on a BERT. In addition to this, a PLL inside or outside the BERT that meets the requirements as defined in 7.5.3 shall be used to generate the clock reference. The BERT scan systematically increases or decreases the clock-to-data timing and directly measures the BER performance at each increment of time. This is done until the time skew is found for the desired BER rate on each side of a Unit Interval. BER rates directly measured at each timing point on the left and right of a UI may be plotted to produce what is known as a bathtub curve. The time for one UI minus the time between the curves at the desired bit error rate is TJ. If those points on the bathtub curve are from directly measured data and not extrapolated, the TJ is a direct measure of TJ. Alternatively, the BERT scan is done to the left and right of the nominal zero crossing time relative to the Reference Clock to directly measure the tails of the Cumulative Distribution Function (CDF) histogram. The width of this histogram at the desired BER is TJ at that BER.

Methods do exist to extrapolate TJ on a BERT from time scan values at higher rates. While such methods may be used to predict TJ at a desired BER, only a direct measure all the way to the desired BER shall be used while using the BERT as a jitter standard for comparison to extrapolation methods.

The standard also requires a measure of DJ for compliance testing. All measures of DJ are statistically based including the estimations of DJ from a BERT. If a BERT is being used to measure TJ, the TJ values determined by that BERT may be used to estimate the DJ. Methods of estimating DJ from BERT TJ values are described in the public domain.[1] These methods involve the measurements of TJ at different BER levels using the BERT scan.

While measuring TJ and extracting the DJ and RJ components, it is common to encounter RJ measurements that are higher than actual random jitter. This is often encountered in systems where noise from the system causes jitter that is not correlated with the Serial ATA channel activity.

EXAMPLE - Power supply noise from a system that contaminates a transmitter's bit clock generator, may cause variations in the bit clock that impact jitter directly.

While making random jitter measurements, this non-correlated DJ is often included in the result that, if multiplied by 14 may lead to non-compliance to jitter specifications. This is inappropriate since non-correlated DJ is bounded, non-Gaussian and should not be multiplied by 14. Furthermore, non-correlated DJ is included in normal DJ measurements.

Extracting non-correlated DJ from RJ measurement lies beyond the scope of this specification since it usually requires in-depth knowledge of the characteristics of the non-correlated DJ and an appropriate algorithm for its measurement/extraction. Consequently, it is the readers' responsibility to characterize and then extract non-correlated DJ from their RJ measurements.

These jitter separation issues are not present for the Gen3i and Gen3u electrical specification for Tx jitter, if TJ(10^{-12}) and TJ(10^{-6}) are measured directly with a full population BERT scan. A full population BERT scan is one that has analyzed a sufficient population of bits versus errors to achieve a 95 % confidence level at the BER level being measured. Estimations of the TJ(BER) levels based on the separation of RJ and DJ components or measurements at lower population levels shall be validated by comparing the reported values to those of the full population BERT scan that is the TJ reference standard.

7.6.11 Transmit jitter (Gen1i, Gen1m, Gen1u, Gen2i, Gen2m, Gen2u)

The transmit jitter values specified in Table 54 and Table 55 refer to the output signal from the UUT at the mated connector into a lab-load (for Gen1i, Gen1m, Gen1u, Gen2i, Gen2m, and Gen2u). The signals are not specified while attached to a system cable or backplane. All the interconnect characteristics of the transmitter, package, printed circuit board traces, and mated connector pair are included in the measured transmitter jitter. Since the SATA adapter is also included as part of the measurement, good matching and low loss in the adapter are desirable to minimize its contributions to the measured transmitter jitter.

Transmit jitter is measured with each of the specified patterns according to 7.4.5.2. The measurement of jitter see 7.6.10. Transmit jitter is measured in one of the following two setups the transmitter is connected directly into the lab-load shown in Figure 225.

[1] "Estimation of Small Probabilities by Linearization of the Tail of a Probability Distribution Function" by S.B. Weinstein, IEEE Transactions on Communications Technology, Vol. COM-19, No. 6, December 1971.

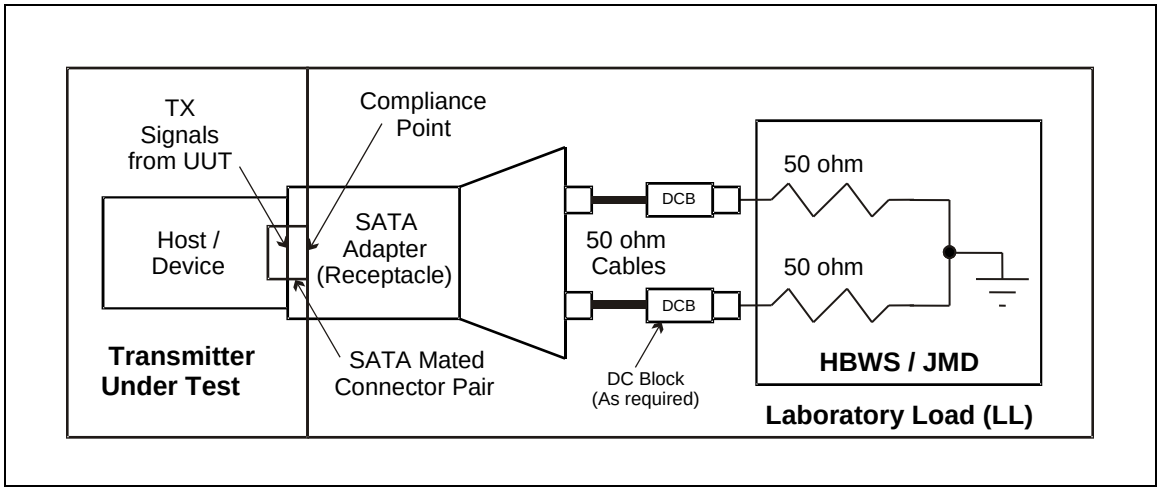


Figure 225 – Transmitter jitter test (Gen1i, Gen2i)

Transmitter jitter is measured into the lab-load, or in conjunction with the CIC; both have very good impedance matching. The jitter in an actual system is higher since load and interconnect mismatch results in reflections and additional data dependent jitter. It is generally not possible to remove the effects of the SATA adapter on jitter since jitter due to mismatch depends on the entire test setup.

7.6.12 Transmit jitter (Gen3i, Gen3u)

The Transmit Jitter values TJ(10^{-12}) and TJ(10^{-6}) specified in Table 54 and Table 55 refer to the output signal from the UUT at the mated connector into a lab-load, and Table 54 from the UUT through a CIC into a Laboratory Load. The signals are not specified while attached to a system cable or backplane. All the interconnect characteristics of the transmitter, package, printed circuit board traces, and mated connector pair are included in the measured transmitter jitter. Since the SATA adapter is also included as part of the measurement, good matching and low loss in the adapter are desirable to minimize its contributions to the measured transmitter jitter.

The Total Jitter parameters are measured with each of the specified patterns as defined in 7.4.5.2 and 7.4.5.4.5. The measurement of jitter as defined in 7.6.10.

One of the measurements of the Transmit Total Jitter parameters on the Tx signal shall be measured directly into the Laboratory Load as is shown in Figure 226.

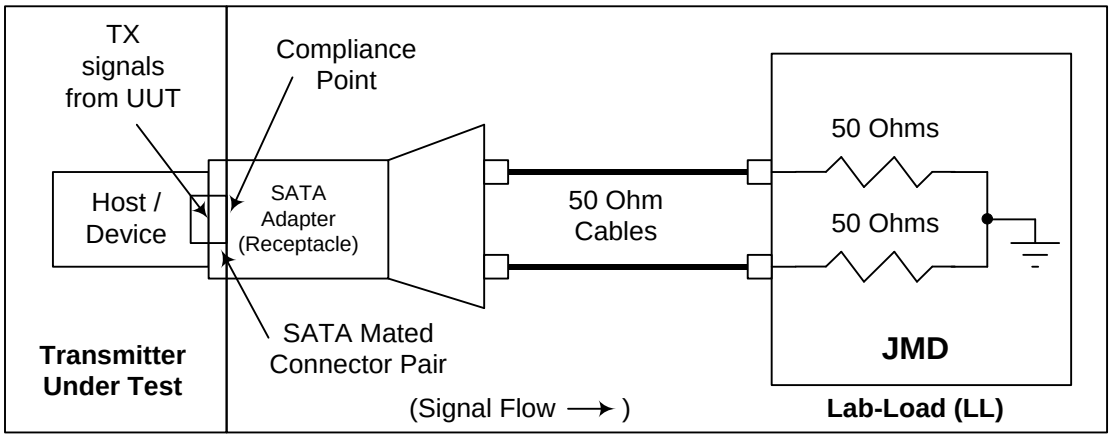


Figure 226 – Transmitter jitter test at Tx (Gen3i)

The second measurement of the Transmit Total Jitter parameters measures the jitter on the Tx signal after passing through the Gen3i CIC (see 7.4.8) into the Laboratory Load as is shown in Figure 215 or Figure 216. This measurement does not apply to Table 55 (i.e., Gen3u).

The Transmit Jitter shall meet both the TJ(10^{-12}) and TJ(10^{-6}) requirements as described in Table 54 both directly at the Tx and after the Gen3i CIC.

The TJ(10^{-6}) requirement is calculated from the TJ(10^{-12}) requirement of 0.52 UI pp and the specification for the maximum RJ pp value of 0.18 UI at a BER of 10^{-12}. If the nominal data rate UI time span is used for conversion to picoseconds, this equates to a TJ(10^{-12}) of 86.667 picosecondsand a RJ 1 sigma value of 2.143 picoseconds.

This calculation is performed using the dual Dirac equation TJ(BER) = ($2Q_{BER} \times \sigma$) + $DJ_{\delta\delta}$, where σ is the 1 sigma value of RJ, and $2Q_{BER}$ is based on a single Gaussian RJ distribution with a transition density of 0.5. This is equivalent to a dual Diac distribution with a transition density of 1.0.

At a BER level of 10^{-12}, $DJ_{\delta\delta}$ = TJ(10^{-12}) - ($14.069 \times \sigma$) = 86.667 picoseconds– (14.069 × 2.143 picoseconds) = 56.517 picoseconds.

At a BER level of 10^{-6}, TJ(10^{-6}) = ($9.507 \times \sigma$) + $DJ_{\delta\delta}$ = (9.507 × 2.143 picoseconds) + 56.517 picoseconds= 76.891 picosecondspp or 0.46 UI.

The tighter requirement at TJ(10^{-6}) is added to constrain the DJ component of TJ, without requiring actual jitter separation that is possible to contain variable results depending on the method used. A tradeoff between RJ versus DJ is possible with this criteria, but the DJ component is limited more than measuring TJ(10^{-12}) alone. If the RJ 1 sigma value is 1.0 picoseconds, then the allowable $DJ_{\delta\delta}$ is able to be 19 % larger than the 56.5 picoseconds derived in the calculation above. If the $DJ_{\delta\delta}$ was 20.0 picoseconds, then the RJ 1 sigma value is able to be 4.74 picoseconds. These calculations are based on the equations above as an example of the trade off between RJ and DJ allowed by these maximum jitter criteria. The $DJ_{\delta\delta}$ used in these equations should not be confused with DJ_{pp} that is not equivalent in the general case.

The full population BERT scan is the jitter measurement reference standard for both the TJ(10^{-6}) and TJ(10^{-12}) measurements for all JMD TJ estimation methods. A full population BERT scan is one that has analyzed a sufficient population of bits versus errors to achieve a 95 % confidence level.

Transmitter jitter is measured into the lab-load, or in conjunction with the CIC. Both have very good impedance matching. The jitter in an actual system is higher since load and interconnects impedance mismatch results in reflections and additional data dependent jitter. It is generally not possible to remove the effects of the SATA adapter on jitter since jitter due to mismatch depends on the entire test setup.

7.6.13 Receiver tolerance (Gen1i, Gen1m, Gen1u, Gen2i, Gen2m, Gen2u)

The performance measure for receiver tolerance and common mode interference rejection is the correct detection of data by the receiver. While measuring receiver and Common Mode tolerance it is necessary to set the maximum allowable jitter and common mode interference on the signal sent to the receiver and monitor data errors.

The data signal source provides a data signal with jitter, and a controlled rise/fall time, with matched output impedance. The sine wave source provides common mode interference with matched output impedance. The two sources are combined with resistive splitters into the receiver under test (see Figure 228). Equivalent signal generation methods that provide the data with jitter, common mode interference, and an impedance-matched output are allowed. All the interconnect

characteristics of the receiver, mated connector pair, printed circuit board traces, and package are included in the measured receiver jitter tolerance.

Figure 227 shows a setup to set the level of jitter and common mode signal at the compliance point, on the cable side of the mated pair connector. The JMD is used as the standard for measuring jitter, and the HBWS is used as the standard for measuring the common mode interference. Since the SATA adapter is not included while setting the level of jitter, good matching and low loss in the adapter are desirable to minimize contributions to the amount of receiver jitter used in testing. Unlike other measurements, it is generally not possible to remove the effects of the SATA adapter on jitter since jitter due to mismatch depends on the entire test setup. Figure 228 shows one example approach to generate the Lab-Sourced Signal.

The receiver tolerance test shall be conducted over variations in parameters:

a) SSC on and SSC off;
b) maximum rise and fall times and minimum rise and fall times;
c) minimum amplitude and maximum amplitude;
d) common mode interference over the specified frequency range; and
e) jitter:
    A) random; and
    B) deterministic:
        a) data dependent;
        b) periodic; and
        c) duty cycle distortion,

while sourcing the test pattern FCOMP as defined in 7.4.5.4.8. The receiver tolerance to the impairments is required over all signal variations.

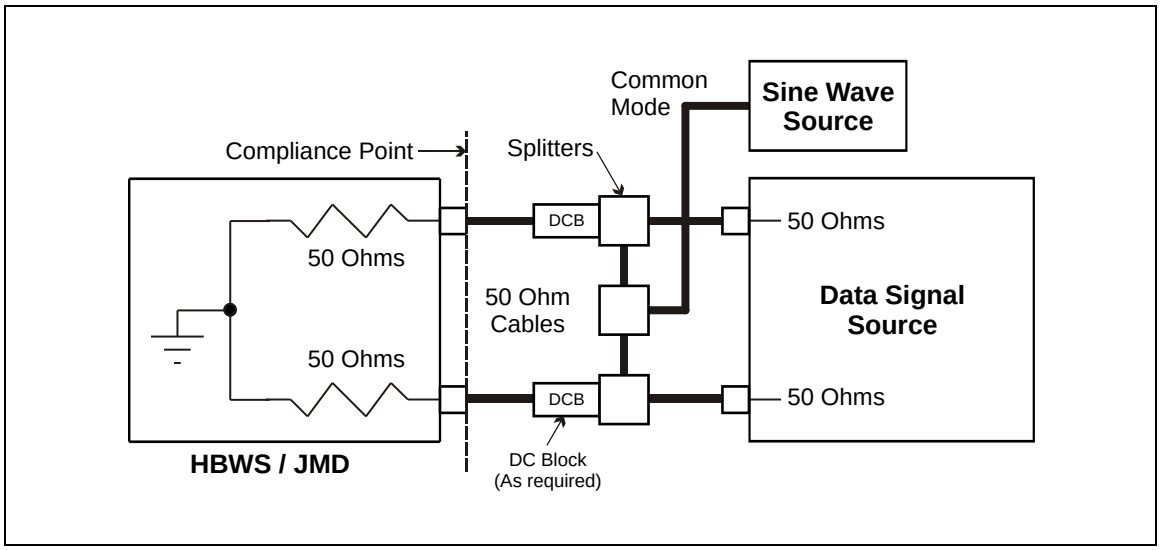


Figure 227 – Receiver jitter and CM tolerance test – setting levels

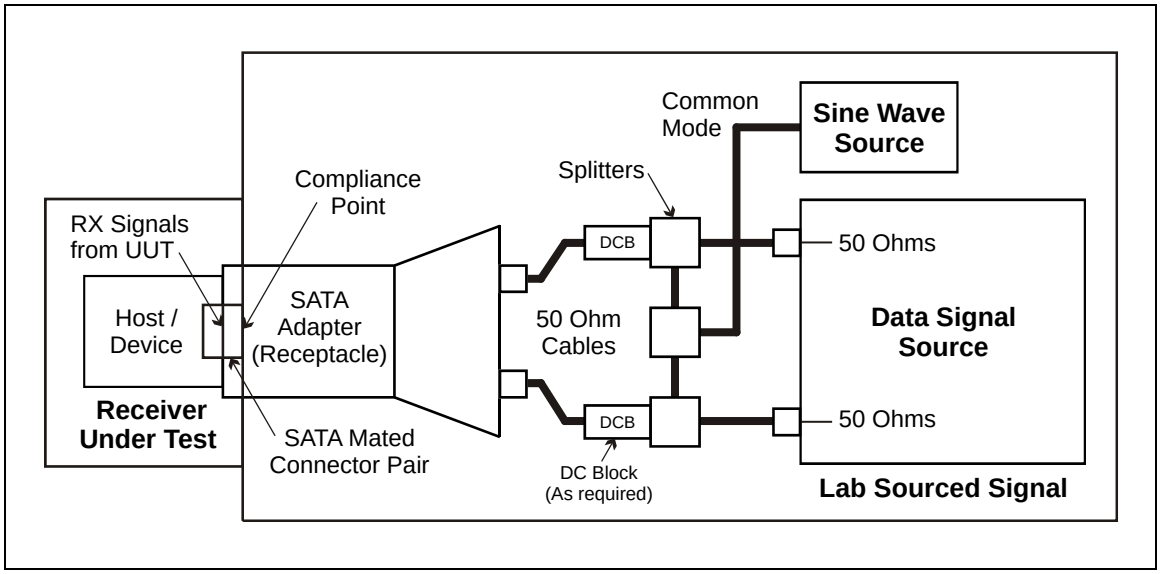


Figure 228 – Receiver jitter and CM tolerance test

7.6.14 Receiver tolerance (Gen3i, Gen3u)

The performance measure for receiver tolerance and Common Mode interference rejection is the correct detection of data by the receiver. While measuring receiver and Common Mode tolerance it is necessary to set the maximum allowable jitter and Common Mode interference on the signal sent to the receiver and monitor data errors.

The data signal source provides a data signal with jitter, and a controlled rise/fall time with a matched output impedance. Additional DJ (ISI) is added by the CIC. The sine wave source provides common mode interference with a matched output impedance. The two sources are combined with resistive splitters to calibrate the data signal source for the receiver under test. Equivalent signal generation methods that provide the data with jitter, common mode interference, and an impedance-matched output are allowed. All the interconnect characteristics of the receiver, mated connector pair, printed circuit board traces, and package are included in the measured receiver jitter tolerance.

To calibrate the test signal for Receiver Tolerance testing, the Data Signal Source is measured using two procedures, one for RJ and a second for TJ and the common mode signal content.

The rise time and fall time of the Data Signal Source in the following figures shall meet the requirements as described in Table 57 and Table 58 for the Gen3i and Gen3u Lab-Sourced Signal. This defines the signal rise time and fall time characteristics in the signal path before the CIC. This requirement shall be met using the rise time and fall time methods as defined in 7.6.5.4.

Figure 229 show the test configuration for setting the RJ level as is defined in Table 57 and Table 58 for the Gen3i and Gen3u Lab-Sourced Signal. The RJ level is set using a Gen3i MFTP pattern. This method minimizes the measurement errors of RJ, compared to the case while other signal degradations are present, and shall be done before adding additional jitter components and common mode signals.

This second procedure is performed after the RJ level of the Data Signal Source is set, as described above. Figure 230 shows one example approach for setting the TJ and the common mode signal level. The actual calibration plane is at the SMA connectors that shall be applied to the SATA to SMA adaptor during the Receiver Tolerance test. This is shown in Figure 230 as a dotted vertical line. The JMD is used as the standard for measuring jitter, and the HBWS is used as the standard

for measuring the common mode interference and signal amplitude. Since the SATA adapter is not included while setting the level of jitter, good matching and low loss in the adapter are desirable to minimize contributions to the amount of receiver jitter used in testing. Unlike other measurements, it is generally not possible to remove the effects of the SATA adapter on jitter since jitter due to mismatch depends on the entire test setup.

The measurement of the minimum and maximum amplitude levels of the test signal at the calibration plane, are performed in the same method used for these parameters for the Tx amplitude tests (see 7.6.5). In general the maximum peak-to-peak amplitude of a Gen3 MFTP pattern is the maximum limit, and the minimum eye height (see 7.6.5.4) of a Gen3 LBP is the minimum limit. The test signal minimum amplitude calibration shall be performed with SSC off, and all the jitter sources used during the tolerance test active, at their calibrated levels. The Gen3i CIC is used to calibrate the Lab-Sourced Signal minimum amplitude for Gen3u receiver tolerance testing, however, the Gen3i CIC losses shall be removed from the Lab-Sourced Signal prior to testing of the Gen3u receiver under test.

Figure 231 shows the calibrated Lab-Sourced Signal applied to the Gen3i Receiver Under Test. Figure 228 shows the calibrated Lab-Sourced Signal applied to the Gen3u Receiver Under Test.

The receiver tolerance test shall be conducted over variations in parameters:

a) SSC on and SSC off;
b) minimum amplitude and maximum amplitude;
c) common mode interference over the specified frequency range; and
d) jitter that includes the maximum:
    A) random; and
    B) deterministic jitter of various types:
        a) data dependent;
        b) periodic; and
        c) duty cycle distortion,

while sourcing the test pattern FCOMP as defined in 7.4.5.4.8.

The receiver tolerance to the impairments is required over all signal variations.

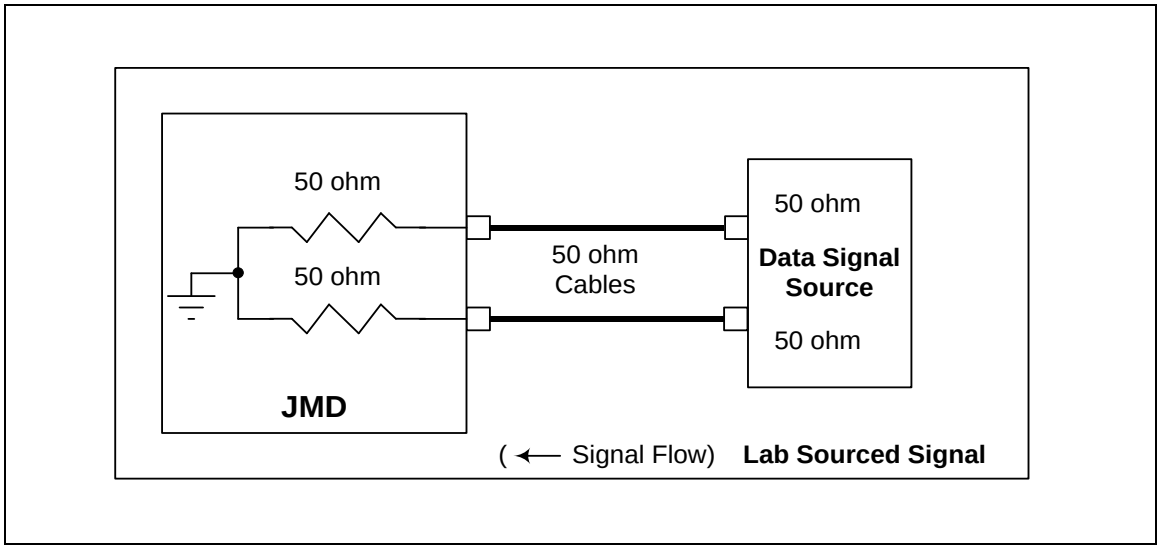


Figure 229 – Receiver jitter and CM tolerance test – setting RJ level (Gen3i)

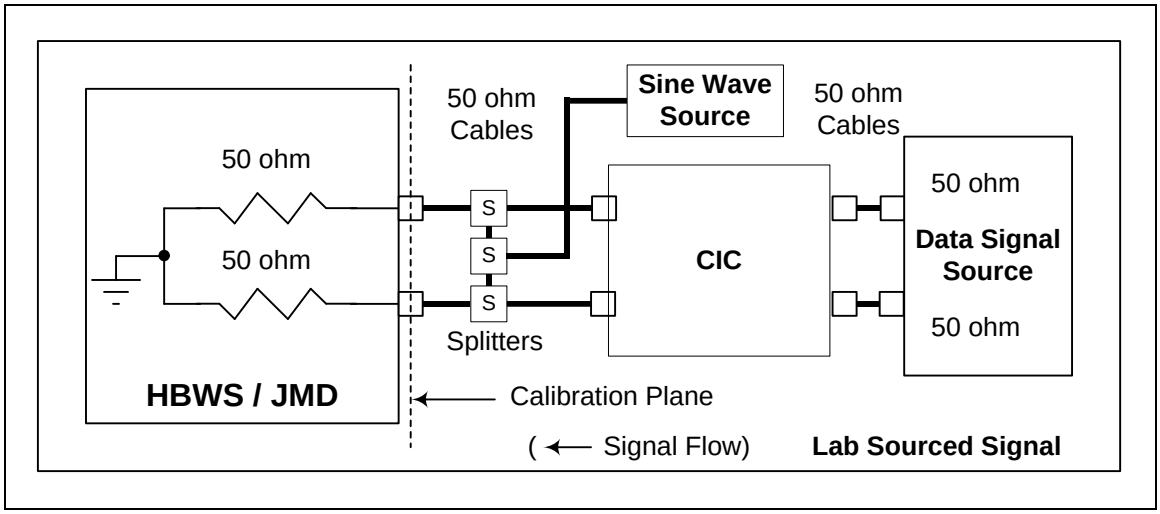


Figure 230 – Receiver jitter and CM tolerance test – setting TJ and CM levels (Gen3i)

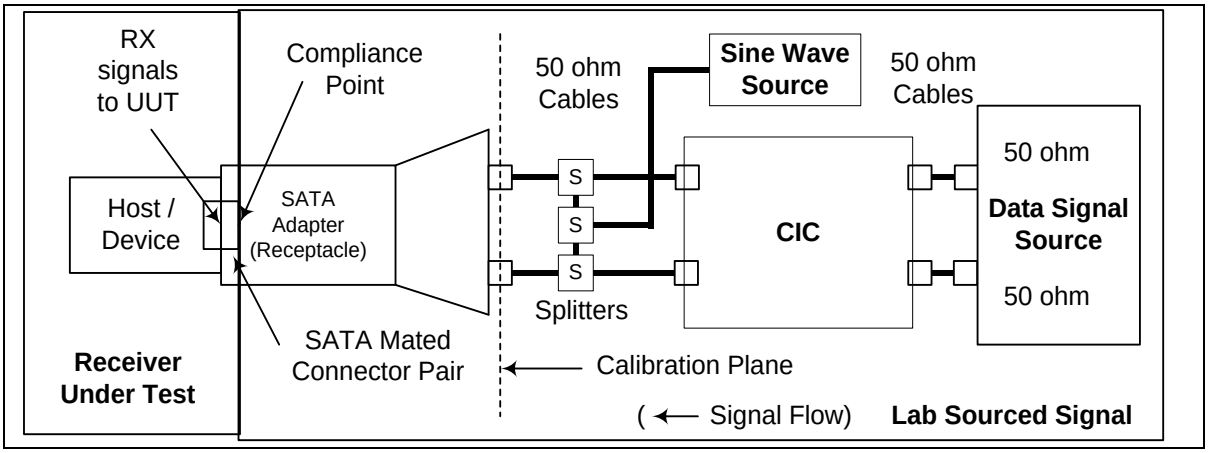


Figure 231 – Receiver jitter and CM tolerance test (Gen3i)

7.6.15 Return loss and impedance balance

The purpose of the return loss and impedance balance specifications (for rates above Gen1) is to bound the additional data dependent jitter incurred when attaching a host/device into a system. The test setup for hosts and devices is impedance matched in both differential and common modes and has good impedance balance whereas the system environment may not. Additional data dependent jitter occurs in a system from these imperfections. The return loss of a host/device quantifies the effect on the level of reflections in the system and the impedance balance controls the conversion between differential and common modes.

The differential return loss is defined as the magnitude of the differential mode reflection given a differential mode excitation, expressed in decibels. The common mode return loss is defined as the magnitude of the common mode reflection given a common mode excitation. The impedance balance is defined as the magnitude of the differential mode reflection given a common mode excitation. Each of these contributes to additional data dependent jitter in a system beyond that in a test setup.

The differential mode signal is defined by

$$v_{dm} = v_2 - v_1$$
$$i_{dm} = i_2 - i_1$$

The common mode signal is defined by

$$v_{cm} = \frac{v_2 + v_1}{2}$$
$$i_{cm} = \frac{i_2 + i_1}{2}$$

The return loss is defined by the magnitude of the reflection coefficient

$$RL = -20\log|\rho|$$

This specification describes transmitter output impedance and receiver input impedance in terms of the magnitude of a reflection of a sine wave. In a lossless line, the return loss remains constant over position. Attenuation loss in the test setup causes the measured return loss to appear higher (better matched) than actual. Figure 232 shows a setup to compensate for the loss in the cables. The short and load are assumed standards with RF connectors (i.e., SMA type connectors).

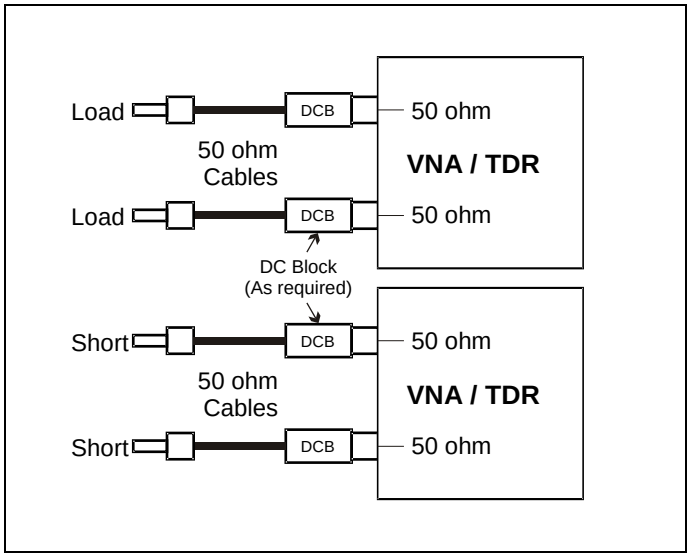


Figure 232 – Return loss test-calibration

A reflection test set allows the measurement of reflections in the differential mode, or in the common mode. It may consist of a TDR with processing software, a multiport vector network analyzer, hybrid couplers and directional bridges, or a 2-port vector network analyzer and processing software.

Differential return loss, common mode return loss, and impedance balance may be measured with a 2-port vector network analyzer. The vector network analyzer (VNA) is connected to the host/device and the S parameters are measured (with a 50 ohm reference impedance). The

differential return loss in terms of the mixed mode S parameters as well as the 2-port S parameters is given by

$$RL_{DD11} = -20\log|S_{DD11}| = -20\log\left|\frac{s_{11} + s_{22} - s_{12} - s_{21}}{2}\right|$$

The common mode return loss is given by

$$RL_{CC11} = -20\log|S_{CC11}| = -20\log\left|\frac{s_{11} + s_{22} + s_{12} + s_{21}}{2}\right|$$

The impedance balance is given by

$$RL_{DC11} = -20\log|S_{DC11}| = -20\log\left|\frac{s_{11} - s_{22} + s_{12} - s_{21}}{2}\right|$$

where the mixed mode S parameters are measured with a 4-port VNA, or alternatively the 2-port S parameters are measured with a 2-port VNA.

For the above equations, the mapping from single-ended to differential S parameters assumes 50 ohm single ended reference sources to 100 ohm differential reference sources and 25 ohm common-mode reference sources.

Figure 233 shows a test setup for measuring differential return loss. Since the SATA adapter is not included, good matching and low loss in the adapter are desirable to minimize its contributions to the measured return loss. If measurements and SATA adapter are characterized with S parameters, it is possible to remove adapter and test setup effects through a de-embedding process.

Test adapter imperfections affect the measurement of the UUT (e.g., they introduce measurement uncertainty).

EXAMPLE - The attenuation loss in the test adapter reduces the reflection from the UUT making the measured return loss higher than actual. An attenuation loss of 0.5 dB (about 1 inch of PCB trace on FR-4 at 5 GHz) causes the measured return loss to increase by 1 dB over actual.

The return loss of the adapter may affect the measured return loss higher or lower as the reflection from the adapter either adds or subtracts from the reflection from the UUT. A well-matched adapter with 20 dB return loss may affect the measurements of a UUT with return loss of 5 dB by ± 1 dB. These effects are most pronounced at higher frequencies.

The adapter affects the measured reflection by the following measurement uncertainty equation

$$s_{11m} = \varepsilon_{00} + \varepsilon_{01}\varepsilon_{10}s_{11a} + \varepsilon_{01}\varepsilon_{10}\varepsilon_{11}\left(s_{11a}\right)^2$$

where ε_{10} and ε_{01} are the attenuation loss, ε_{00} is the input reflection, and ε_{11} is the output reflection of the adapter. S_{11m} is the measured reflection, and S_{11a} is the actual reflection from the UUT. The return loss is related to the reflection amplitude.

For the most accurate measurements, the adapter effects should be characterized or calibrated, and then de-embedded or removed from the measurements.

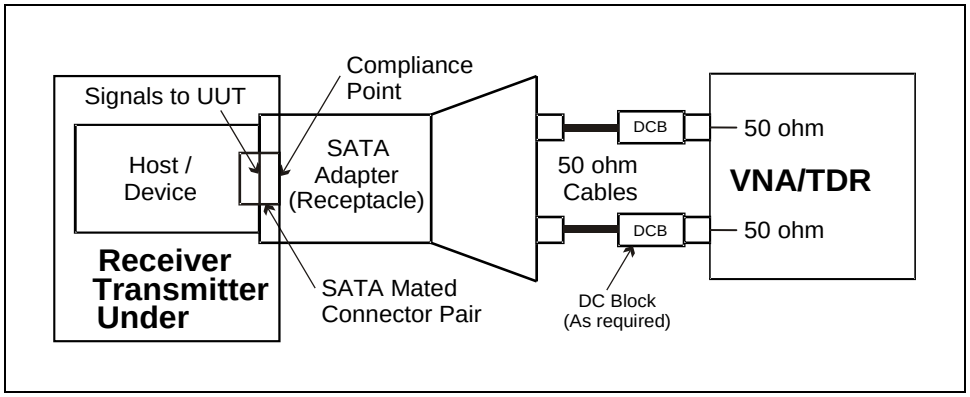


Figure 233 – Return loss test

While measuring output impedance of transmitters the operating condition shall be during transmission of MFTP. This is to assure the measurement is performed during a mode of operation that represents normal operation. The amplitude of external excitation applied shall not exceed -13.2 dBm 50 ohm (i.e., 139 mVpp) single ended on each differential port of the transmitter. This number is derived from the maximum reflected signal that may be present at a transmitter. A maximum transmitted signal of 700 mVppd (-5.14 dBm 50 ohm, each differential side, HFTP maximum rise/fall time), reflecting off a receiver with a differential return loss of 8 dB and direct connection.

While measuring input impedance of receivers the operating condition shall be during a PHYRDY Interface Power State (see 8.1). The amplitude of external excitation applied shall not exceed -6.48 dBm 50 ohm (i.e., 300 mVpp) on each differential port of the receiver. This number is derived from the maximum signal that may be present at a receiver, 600 mVppd (Gen1i, -6.48 dBm 50 ohm, each differential side).

7.6.16 SSC profile

The SSC profile is the modulation on the bit clock. To measure the SSC profile, a frequency demodulator and low pass filter are necessary. There are many possible realizations of this, in hardware and software. The low pass filter is necessary to reject undesired post-demodulation frequency components from bit patterns and jitter. To minimize these undesired signals the HFTP bit pattern shall be used. This may be produced using the BIST Activate FIS to invoke the Transmit-Only option. The SSC Profile measurement is also used to determine the UI values.

The Reference Clock as defined in 7.5.3 should be used with an additional low pass filter in the phase detector output to measure the SSC profile. The output is DC coupled and should be calibrated with a signal source with sufficient long-term frequency accuracy.

A single shot capture oscilloscope should be used to measure the times of zero crossings (through interpolation) and perform the FM demodulator and low pass filter function. The memory record of the oscilloscope shall be long enough to achieve the low pass filter cutoff frequency. The long term frequency accuracy of the oscilloscope time base should be significantly better than the 350 ppm limit in this specification. Oscilloscopes that do not have this frequency accuracy may be calibrated using a separate signal source of sufficient accuracy into a separate channel.

Modulation analysis tools with sufficient bandwidth provide alternative methods of measuring the SSC profile. These exist in some spectrum analyzers, modulation analyzers, or is able to be implemented as a separate frequency modulation receiver. Calibration is easier while the FM receiver has a DC coupled modulation path.

The low pass filter 3 dB cutoff frequency shall be 60 times the modulation rate. The filter stopband rejection shall be greater or equal to a second order low-pass of 20 dB/dec.

Evaluation of the maximum df/dt is possible to be achieved by inspection of the low-pass filtered waveform at its various high magnitude df/dt ppm-changes, making sure that the maximum df/dt ppm/us is never exceeded.

With the host/device under test continuously transmitting D10.2 patterns, this measurement is possible to be achieved by plotting the periods of the differential waveform versus time, and then using the cursor-data to calculate the instantaneous ppm-variations over the time segment of interest. The minimum time segment that should be used for this evaluation shall be 1.5 us, with a relative tolerance of ± 3 %.

7.6.17 Intra-pair skew

Intra-pair skew measurements are important measurements of transmitters and receivers. For transmitters they are a measure of the symmetry of the SATA transmitter silicon (see Table 54 and Table 55).

For receivers, they are a measure of the ability to handle signal degradation due to the interconnect. At a receiver, intra-pair skew adversely affects jitter levels. In a system, intra-pair skew has a direct impact on radiated emission levels. As the measurement values are typically just a few picoseconds, care should be taken to minimize measurement error.

Figure 234 illustrates a test setup for a measurement method using a HBWS and its built-in processing. Each single-ended channel of a transmitter is measured into a Laboratory Load with DC blocks. Use HFTP and MFTP as the test patterns when measuring transmitter skew. A new displayed signal is formed by mathematically changing the polarity (arithmetic sign) and displayed with the original signal. This creates crossover points for each single ended signal, one displayed on the upper and the other on the lower part of the display. The example shown in Figure 237 of a transmitter, C5 and C6 are the two single-ended signals of the differential pair, the M5-trace is the inverted C5 (-C5), and M6 is the inverted C6 (-C6). Vertical cursors are used to measure the time between crossovers as the intra-pair skew.

Receivers shall be tested (see Figure 236) to show required performance with the Rx Differential Skew set to maximum as specified in Table 57 and Table 58**.** Skew may be created using test cables of differing propagation delay or active control by the data signal source within the Lab-Sourced Signal generator. Receiver skew may be setup at the same time as receiver amplitude as seen in Figure 235. Use the HFTP as the pattern when setting the skew. The skew measurement is performed as described above for the transmitter.

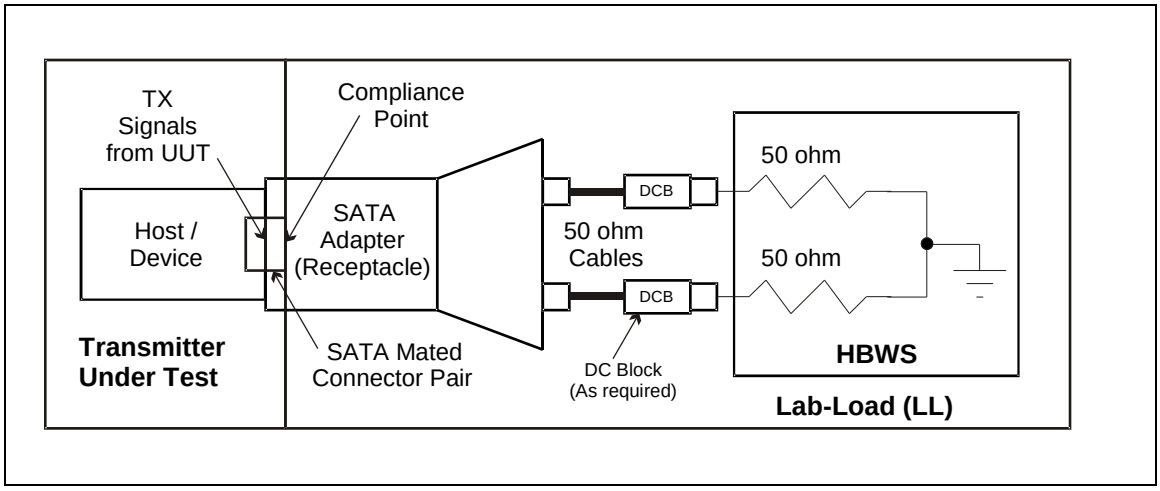


Figure 234 – Intra-pair skew test for a transmitter

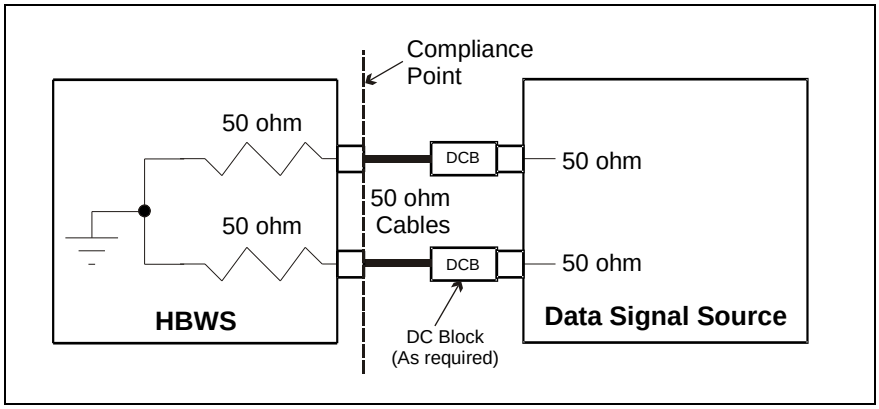


Figure 235 – Receiver intra-pair skew test – setting levels

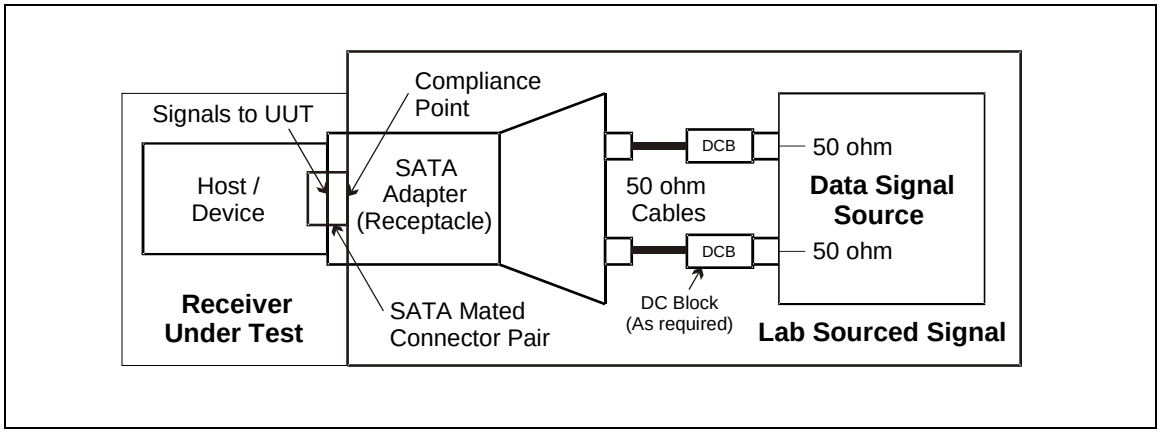


Figure 236 – Receiver intra-pair skew test

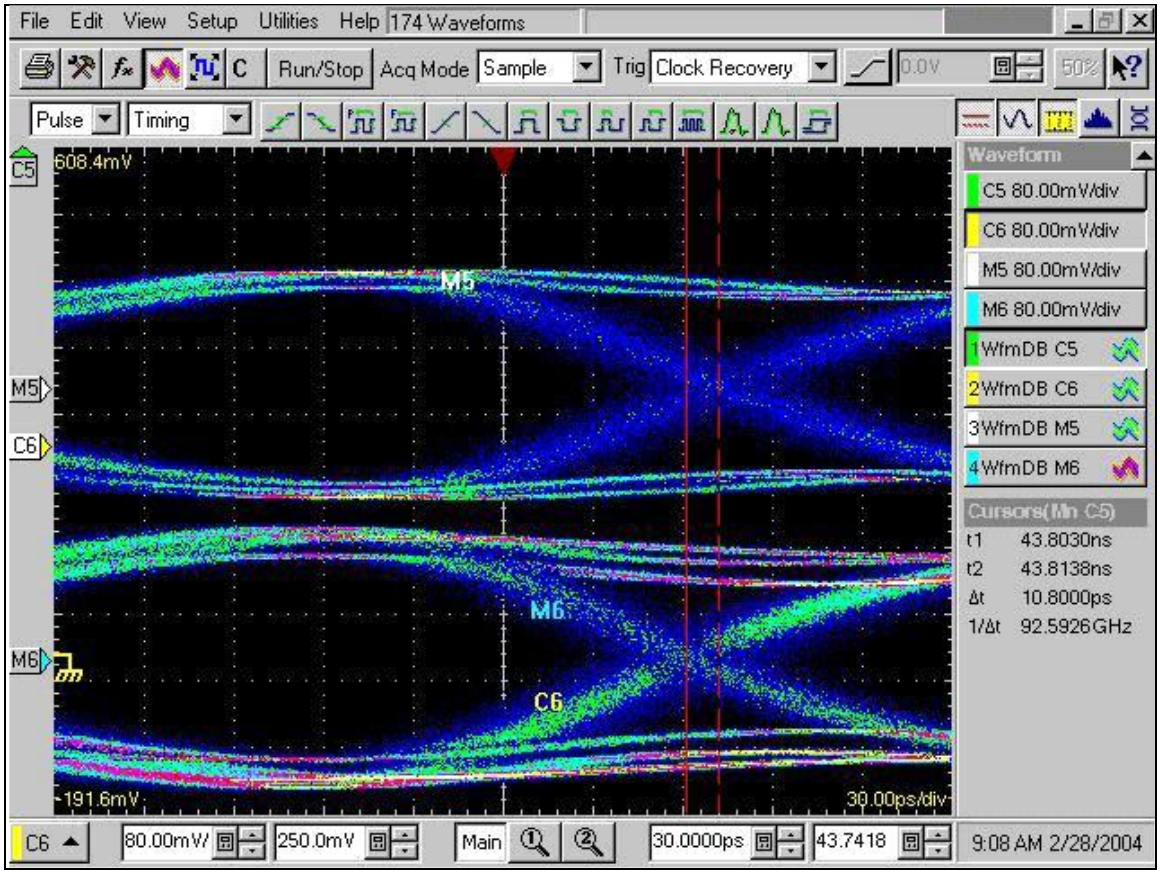


Figure 237 – Example intra-pair skew test for transmitter (10.8 picoseconds)

7.6.18 Sequencing transient voltage

Figure 238 shows the connections to the receiver or transmitter under test. Each Rx or Tx line is terminated to ground with a minimum impedance of 10 Mohm that includes the probe and any external load. The value of the voltage transients during power on or power off sequencing, or power state changes seen at Vp or Vn, shall remain in the voltage range specified.

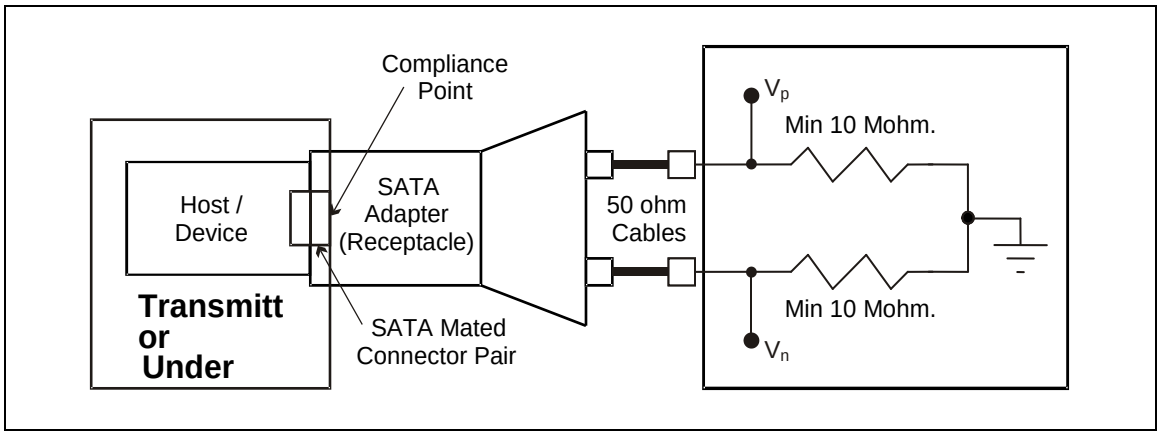


Figure 238 – Tx/Rx sequencing transient Voltage measurement

7.6.19 AC coupling capacitor

This measurement is only applicable to AC coupled transmitters and receivers. The AC coupling capacitor value is not directly observable at the SATA connector.

In order to measure this capacitance, each signal shall be probed on both sides of the AC coupling capacitor. The UUT is powered off and nothing is plugged into the SATA connector. In the case of coupling within the IC or where there is no access to the signals between the IC and external coupling capacitors, this parameter is not measurable as shown.

Figure 239 shows the connections to each coupling capacitor. Each coupling capacitor shall be lower than the specified maximum.

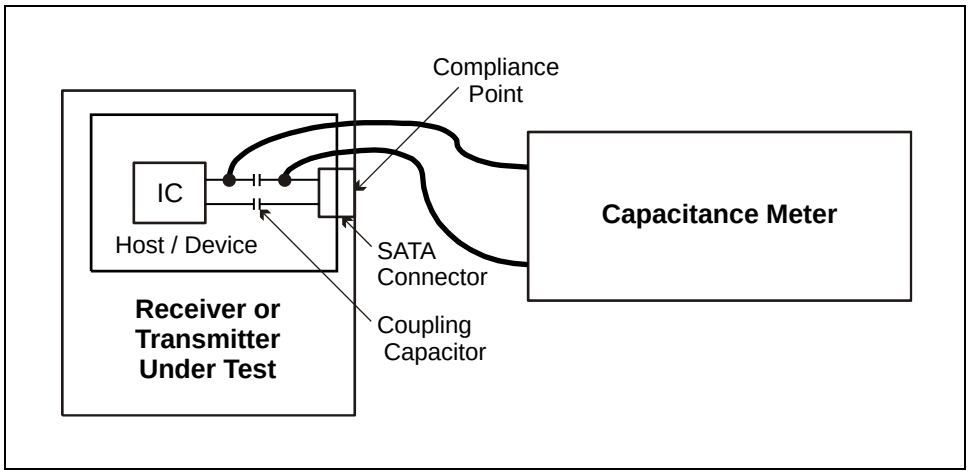


Figure 239 – AC coupled capacitance measurement

7.6.20 Tx amplitude imbalance

This parameter is a measure of the match in the single-ended amplitudes of the Tx+ and Tx- signals. The test setup shown Figure 214 shall be used for this measurement. This parameter shall be measured and met with both the HFTP and MFTP patterns. Clock-like patterns are used here to enable the use of standard mode-based amplitude measurements for the sole purpose of determining imbalance. Due to characteristics of the MFTP, it is required that the measurement points be taken between 0.45 UI to 0.55 UI of the second bit within the pattern. All amplitude values for this measurement shall be the statistical mode measured at 0.5 UI, nominal, over a minimum of 10 000 UI.

In order to determine the amplitude imbalance, single ended mode high and mode low based amplitudes of both Tx+ and Tx- over 10 cycles to 20 cycles of the clock-like pattern being used shall be determined. The amplitude imbalance value for that pattern is then determined by the equation:

$$\frac{\left| \text{TX+ amplitude - TX- amplitude} \right|}{\text{average}}$$

where:

$$\text{average} = \frac{\text{TX+ amplitude + TX- amplitude}}{2}$$

The amplitude imbalance value for each pattern shall be less than the maximum as described in Table 54.

7.6.21 Tx rise/fall imbalance (obsolete)

This parameter is a measure of the match in the simultaneous single-ended rise/fall or fall/rise times of the Transmitter. The test setup shown in Figure 214 shall be used for this measurement. This parameter shall be measured and met with both the HFTP and MFTP patterns.

In order to determine the imbalance, the single ended 20 % to 80 % rise and fall times of both Tx+ and Tx- shall be determined for a given pattern. Two imbalance values for that pattern are then determined by the two equations:

$$\frac{\left| \text{TX+, rise - TX-, fall} \right|}{\text{average}}$$

where:

$$\text{average} = \frac{\text{TX+, rise + TX-, fall}}{2}$$

and:

$$\frac{\left| \text{TX+, fall - TX-, rise} \right|}{\text{average}}$$

where:

$$\text{average} = \frac{\text{TX+, fall + TX-, rise}}{2}$$

Both values for each pattern shall be less than the maximum as described in Table 54.

7.6.22 Tx AC common mode voltage (Gen2i, Gen2m)

This parameter is a measure of common mode noise other than the common mode (CM) spikes during transitions due to Tx+/Tx- mismatch and skews that are limited by the rise/fall mismatch and other requirements. Measurement of this parameter is achieved by transmitting through a mated connector into a lab-load (see Figure 214). The transmitter shall use an MFTP during a data transfer only, not involving OOB transmissions. The measurement shall be done with a HBWS having a measurement bandwidth limited on the low end at 200 MHz and on the high end at bitrate/2 (fundamental), using first order filtering.

Separate channels shall be used for Tx+ and Tx-, with the common mode being (Tx+ + Tx-) / 2. The raw common mode is filtered to remove the noise contribution from the edge mismatches. The

peak-to-peak voltage of the filter output is the AC Common Mode Voltage, and shall remain below the specified limit.

7.6.23 Tx AC common mode voltage (Gen1u, Gen2u, Gen3i, Gen3u)

This parameter is a measure of common mode noise other than the CM spikes during transitions due to Tx+/Tx- mismatch and skews that are limited by the rise/fall mismatch and other requirements. Measurement of this parameter is achieved by transmitting through a mated connector into a lab-load such as shown in Figure 214. The transmitter shall use an MFTP and an HFTP during a data transfer only, not involving OOB transmissions. The measurement shall be done with a HBWS having a measurement bandwidth limited on the low end at 200 MHz and on the high end at bitrate/2 (fundamental), using first order filtering. Separate channels shall be used for Tx+ and Tx-, with the common mode being (Tx+ + Tx-) / 2. The raw common mode is filtered to remove the noise contribution from the edge mismatches. The peak-to-peak voltage of the filter output is the AC Common Mode Voltage, and shall remain below the specified limit.

7.6.24 OOB common mode delta

This parameter is a measure of the offset between the common mode voltage of idle times during OOB generation and the common mode voltage during the OOB bursts. The test setup shown in Figure 214 shall be used for this measurement. A HBWS or single-shot scope may be used for this measurement, the UUT shall be configured to send an OOB sequence or multiple OOB sequences, and the instrument shall be configured so that at least 40 Gen1 UI worth of idle time before the first OOB burst in a sequence and at least 40 Gen1 UI worth of burst activity in the first OOB burst of a sequence are observed.

The common mode signal is (Tx+ + Tx-)/2 and the common mode voltage during idle for this parameter is determined by averaging the common mode voltage of a 40 Gen1 UI span of idle time within the last 60 Gen1 UI worth of time prior to the first OOB burst in a sequence. The average common mode voltage during active time for this parameter is determined by averaging the common mode voltage of a 40 Gen1 UI span of time within the first 60 Gen1 UI of the first burst in a sequence. The reason that the active span is taken within the first 60 Gen1 UI of the first burst in a sequence is to minimize the affect of AC coupling RC time constant on the resulting common mode offset if one exists.

7.6.25 OOB differential delta

This parameter is a measure of the offset between the differential voltage of idle times during OOB generation and the average differential voltage during the OOB bursts. The test setup shown in Figure 214 shall be used for this measurement. A HBWS or single-shot scope may be used for this measurement, the UUT shall be configured to send an OOB sequence or multiple OOB sequences, and the instrument shall be configured so that at least 40 Gen1 UI worth of idle time before the first OOB burst in a sequence and at least 40 Gen1 UI worth of burst activity in the first OOB burst of a sequence are observed.

The differential signal is Tx+ - Tx- and the differential voltage during idle for this parameter is determined by averaging the differential voltage of a 40 Gen1 UI span of idle time within the last 60 Gen1 UI worth of time prior to the first OOB burst in a sequence. The average differential voltage during active time for this parameter is determined by averaging the differential voltage of a 40 Gen1 UI span of time within the first 60 Gen1 UI of the first burst in a sequence. The use of a span of 40 Gen1 UI ensures that no matter what the starting time within the burst, the signal is DC balanced and the average represents the differential mean. The reason that the active span is taken within the first 60 Gen1 UI of the first burst in a sequence is to minimize the affect of AC coupling RC time constant on the resulting differential offset if one exists.

7.6.26 Squelch detector tests

The squelch detector is an essential function in receiving OOB signaling.

There are two conditions to test:

a) if above the maximum threshold,then the detector shall detect; and
b) if below the minimum threshold, then the detector shall not detect.

Figure 240 shows the test setup to set the proper level of the OOB signal. To ensure the proper detection, multiple tests shall be done and the statistics of the results presented to show compliance.

NOTE 33 - Note the same method is used to calibrate the Lab-Sourced Signal amplitude as defined in 7.6.8.

NOTE 34 - Note the pattern content in the OOB may affect the detection.

The timing of the gaps in the OOB bursts shall be varied to ensure compliance to the OOB timing specification (see Table 59). Figure 241 shows how to connect a receiver under test for a squelch detector threshold test.

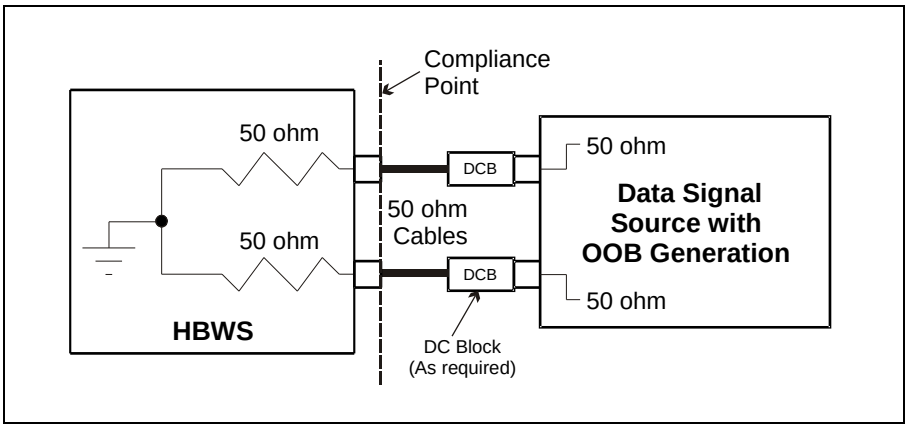


Figure 240 – Squelch detector threshold test – setting levels

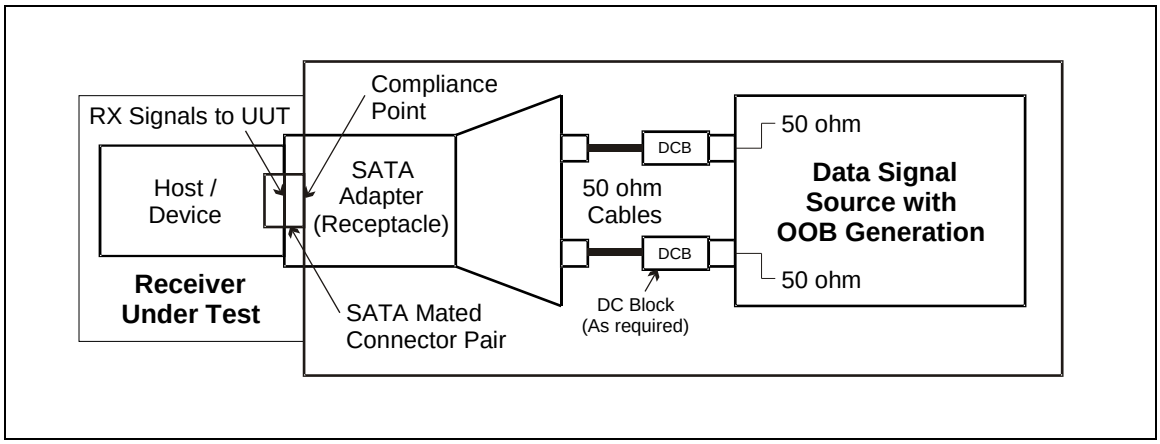


Figure 241 – Squelch detector threshold test

7.6.27 OOB signaling tests

7.6.27.1 OOB signaling tests overview

OOB signaling is used to signal specific actions during conditions where the receiving interface is in an active mode, a low interface power state, or a test mode.

This section specifies the set of test requirements to ensure that the OOB detector circuits comply with the OOB signaling sequences under various conditions.

7.6.27.2 Power-on sequence

7.6.27.2.1 Calibration

If the host controller performs impedance calibration, it shall adjust its own impedance such that the electrical requirements as defined in 7.4 are satisfied.

7.6.27.2.2 Speed negotiation

Speed negotiation and transition to lower serial interface data rates shall be implemented for higher data speed compatible interfaces, negotiating and transitioning down to lower data speed, as required. There is no requirement for speed negotiation and transition to lower speeds than Gen1. If Gen2 speed is supported, then Gen1 speed shall be supported. If Gen3 speed is supported, then Gen1 and Gen2 speeds shall be supported.

7.6.27.2.3 Interface power management sequences

7.6.27.2.3.1 Partial

The interface shall detect the OOB signaling sequence COMWAKE and COMRESET if in the Partial Interface power management state.

While in the Partial state, the interface shall be subjected to the low-transition density bit pattern (LTDP) sequences as defined in 7.4.5.4. The interface shall remain in the Partial state until receipt of a valid COMWAKE (or COMRESET) OOB signaling sequence.

Power dissipation in this Partial state shall be measured or calculated to be less than the Phy Active state, but more than the Slumber state as defined in 8.1.

The requirement for a "not-to-exceed" power dissipation limit in the Partial interface power management state is classified as vendor specific, and should be documented as part of the implementation performance specifications.

7.6.27.2.3.2 Slumber

The interface shall detect the OOB signaling sequence COMWAKE and COMRESET if in the Slumber Interface power management state.

While in the Slumber state, the interface shall be subjected to the low-transition density bit pattern (LTDP) sequences as defined in 7.4.5.4. The interface shall remain in the Slumber state until receipt of a valid COMWAKE (or COMRESET) OOB signaling sequence.

Power dissipation in this Slumber state shall be measured or calculated to be less than the Phy Ready state, and less than the Partial state as defined in 8.1.

The requirement for a "not-to-exceed" power dissipation limit in the Slumber interface power management state is classified as vendor specific, and should be documented as part of the implementation performance specifications.

7.6.28 TDR differential impedance (Gen1i, Gen1m, Gen1u)

This section describes transmitter output impedance and receiver input impedance in terms of both the peak value of a reflection given an incident step of known risetime and also in terms of return loss. The return loss measurement shall be sufficient to verify compliance with Gen1 requirements. In order to ensure replacement of the test outlined below does not invalidate Gen1 designs passing the TDR differential impedance, that method shall be sufficient to verify compliance with Gen1 requirements. Verification of compliance by both methods shall not be required.

To achieve consistent measurements it is important to control the test conditions at the compliance point (see Figure 243). These conditions include the signal launch (see 7.4.7), the source match looking back into the test setup and TDR, the risetime and shape of the TDR edge, and the attenuation loss on the reflection return path to the TDR. There are various methods to control and remove the test setup effects.

If measuring output impedance of transmitters the operating condition shall be during transmission of MFTP. This is to assure the measurement is performed during a mode of operation that represents normal operation. The amplitude of a TDR pulse or excitation applied to an active transmitter shall not exceed 139 mVpp (-13.2 dBm 50 ohm) single ended. This number is derived from the maximum reflected signal that may be present at a transmitter. A maximum transmitted signal of 700 mVppd reflecting off a receiver with a differential return loss of 8 dB and direct connection.

Source match is a constant 100 ohm differential impedance level on the TDR trace preceding the compliance point. This may be achieved by impedance controlled test setup or a calibration procedure.

Figure 242 shows the setup to set the risetime at the device under test. The risetime shall be set accurately at the compliance point. The shape of the TDR edge at the compliance point is affected by the edge shape of the TDR generator, the attenuation loss in the test setup, and averaging done on the received signal at the TDR.

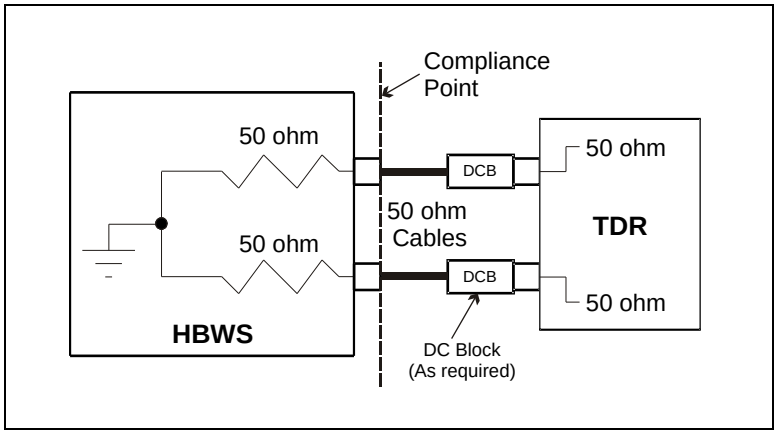


Figure 242 – TDR differential impedance test – setting risetime

Since the SATA adapter is not included while setting risetime, good matching and low loss are necessary in the adapter to minimize errors in the measured TDR impedance.

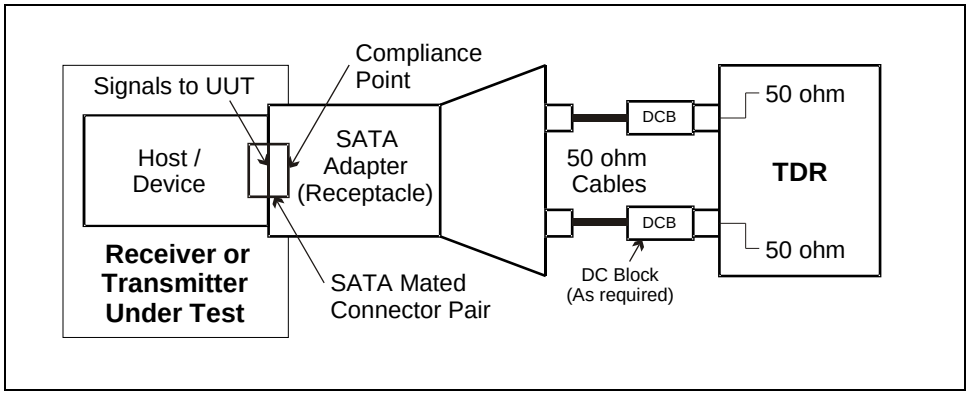


Figure 243 – TDR impedance test

7.6.29 TDR single-ended impedance (Gen1i, Gen1m)

This section describes transmitter single-ended output impedance and receiver single-ended input impedance in terms of the peak value of a reflection given an incident step of known risetime. To achieve consistent measurements it is important to control the test conditions at the compliance point. These conditions include the signal launch (see 7.4.7), the source match looking back into the test setup and TDR, the risetime and shape of the TDR edge, and the attenuation loss on the reflection return path to the TDR. There are various methods to control and remove the test setup effects.

Source match is a constant 50 ohm single-ended impedance level on the TDR trace preceding the compliance point. This may be achieved by impedance controlled test setup or a calibration procedure.

Figure 244 shows the setup to set the risetime at the device under test. The risetime shall be set accurately at the compliance point. The shape of the TDR edge at the compliance point is affected by the edge shape of the TDR generator, the attenuation loss in the test setup, and averaging done on the received signal at the TDR.

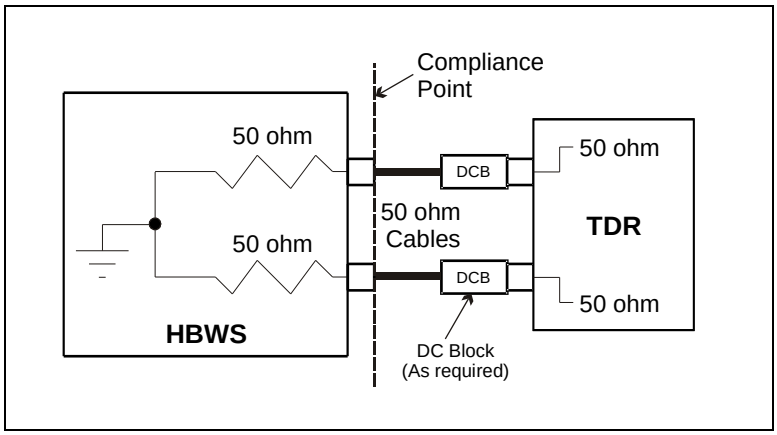


Figure 244 – TDR single-ended impedance test – setting risetime

Since the SATA adapter is not included while setting risetime, good matching and low loss are necessary in the adapter to minimize errors in the measured TDR impedance.

Figure 247 shows the connections to the receiver or transmitter under test. For single-ended measurements, the TDR shall be set to produce simultaneous positive pulses on both signals of the pair. Single-ended impedance is the resulting (even mode) impedance of each signal observed independently. Both signals shall meet the single-ended impedance requirement.

7.6.30 DC coupled common mode voltage (Gen1i)

This measurement is only applicable to DC coupled transmitters and receivers. The following measurement on an AC coupled signal or with AC coupled probing results in a value near or at 0 V. Figure 245 shows the connections to the receiver or transmitter under test. Each Rx or Tx line is terminated to ground with a minimum impedance of 10 Mohm that includes the probe and any external load. The common mode is (Vp + Vn)/2 and this term shall be in the range specified.

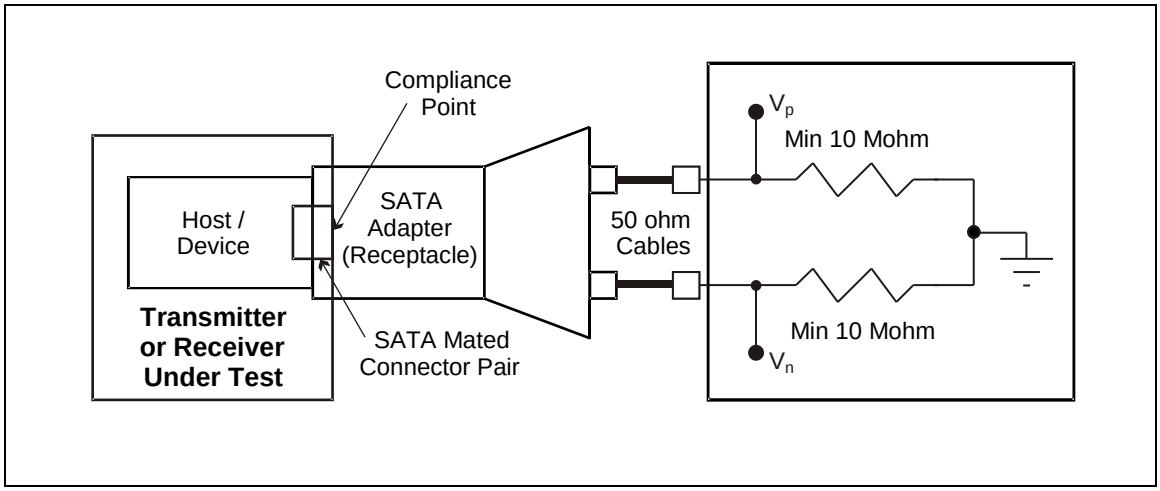


Figure 245 – DC coupled common mode Voltage measurement

7.6.31 AC coupled common mode voltage (Gen1i, Gen1m)

This measurement is only applicable to AC coupled transmitters and receivers. The AC coupled common mode voltage is not directly observable at the SATA connector.

In order to measure this voltage, each Rx or Tx signal shall be probed between the IC and AC coupling capacitor. In the case of coupling within the IC or where there is no access to the signals between the IC and external coupling capacitors, it is not measurable.

Figure 246 shows the connections to the receiver or transmitter under test. Each Rx or Tx line is terminated to ground with a minimum impedance of 10 Mohm that includes the probe and any external load. The common mode is (Vp + Vn)/2 and this term shall be in the range specified.

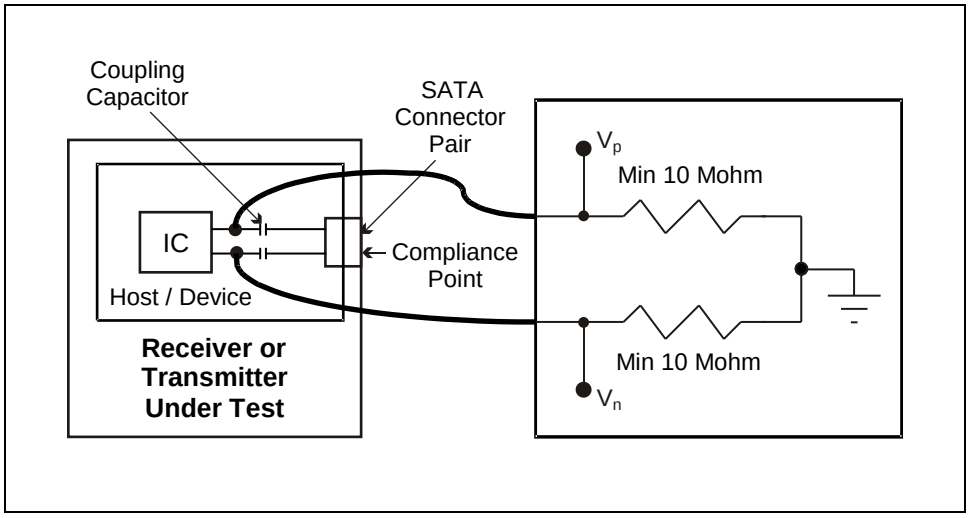


Figure 246 – AC coupled common mode Voltage measurement

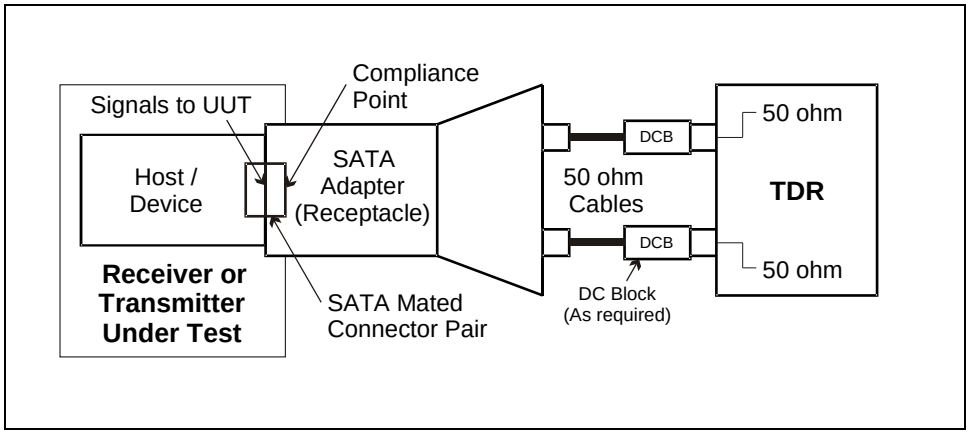


Figure 247 – TDR impedance test

7.6.32 Sequencing transient voltage - lab-load (Gen3i, Gen3u)

Figure 248 shows the connections to the receiver or transmitter under test. Each Rx or Tx line is terminated to ground using a lab-load (see 7.6.2 for lab-load definition). The value of the voltage transients during power on or power off sequencing, or power state changes seen at Vp or Vn, shall remain in the voltage range specified.

In some lab-load configurations additional DC blocking components are added. For this measurement there shall not be any additional DC blocking components added in the lab-load.

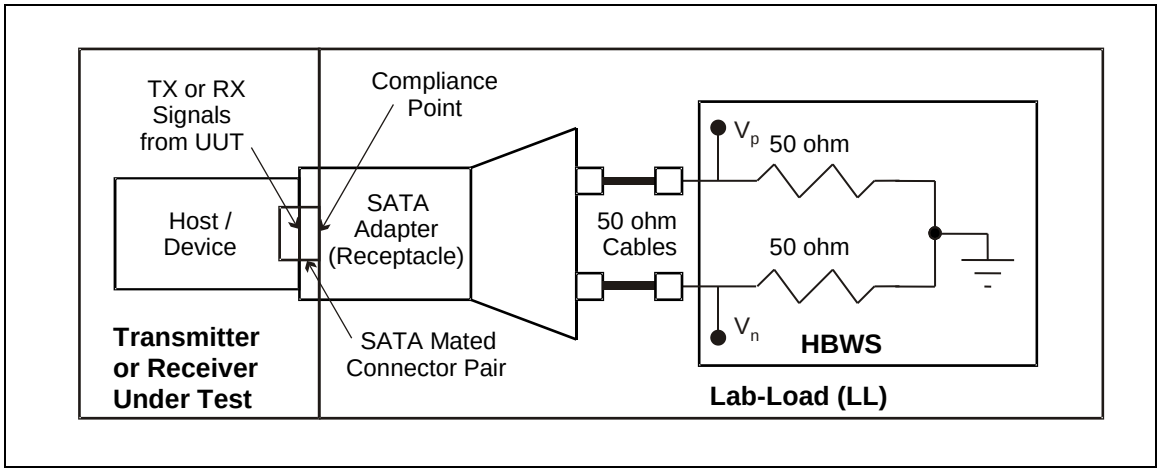


Figure 248 – Sequencing transient Voltage lab-load (LL)

7.6.33 Transmitter emphasis (Gen1i, Gen1u, Gen2i, Gen2u, Gen3i, Gen3u)

7.6.33.1 Transmitter emphasis overview

The transmitter emphasis values specified in Table 54 and Table 55 refer to the output signal from the UUT at the mated connector of the device. The host transmit emphasis is specified while attached to the system cable(s), connectors, motherboard, and/or backplane that are used when integrated with the device. For the alternate compliance method specified in Section 7.4.3.3.14.2, the measurement is made with the CIC rather than with the actual system cable.

The Device shall meet at least one of the Device Tx Emphasis specifications in Table 54 according to the measurement method defined in 7.6.33.3 or 7.6.33.4. Verification of compliance with both methods shall not be required. The measurement methods verify that Device Tx Emphasis is present and is bounded. It is up to the device manufacturer to determine the level of Device Tx Emphasis.

The Host shall meet the Host Tx Emphasis specifications in Table 54 and Table 55 according to the measurement method as defined in 7.6.33.3.

7.6.33.2 Transmitter emphasis measurement (Gen1i, Gen1u, Gen2i, Gen2u, Gen3i, Gen3u)

Figure 249 and Figure 250 show the test setups for measuring emphasis. The HBWS is the standard for measuring emphasis. The losses in the test connections may be significant so it is prudent to minimize and estimate these.

Several methods may be used to estimate the cabling losses:

a) use two cables of different lengths and compare the losses of each;
b) rely on published data for the cables; or
c) obtain a separate means for measuring the cable loss (e.g., characterization with a network analyzer or power meter).

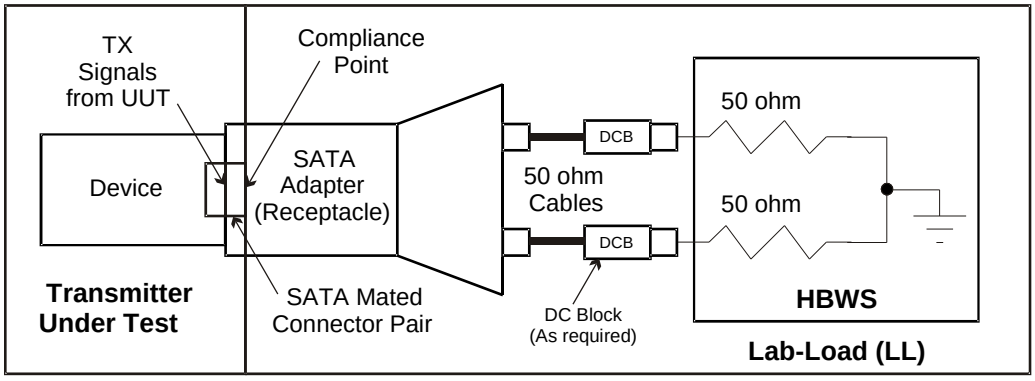


Figure 249 – Device transmit emphasis test with Lab-Load (LL)

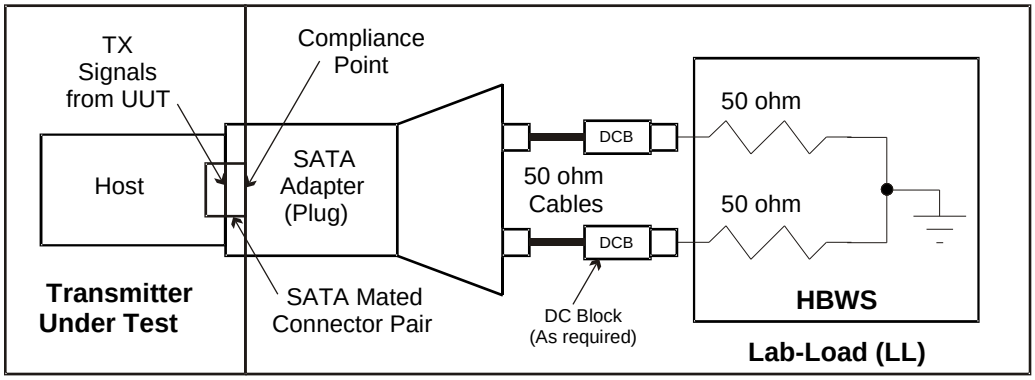


Figure 250 – Host transmit emphasis test with Lab-Load (LL)

This specification describes emphasis levels in terms of voltage amplitude ratio in dB while driving a test load of 100 ohm differential (i.e., lab-load) and 50 ohm single ended to ground.

7.6.33.3 Measurement of emphasis (Gen1i, Gen1u, Gen2i, Gen2u, Gen3i, Gen3u)

Transmitter emphasis is measured by comparing the mean differential voltage of the first bit of MFTP versus the mean differential voltage of the second bit of MFTP.

To test for emphasis, use the following steps:

Step 1, transmitting an MFTP pattern, for a UI corresponding to the first 1 bit, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

- $\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode;
- s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS; and
- n is the number of samples that contribute to the histogram – this may also be read from the HBWS.

The inequality above is based on a requirement that enough samples are collected to define a confidence interval with at least 95 % probability and with a width no greater than 10 % of the sample mean.

Call the mean:

$$A = \bar{x}$$

Step 2, transmitting an MFTP pattern, for a UI corresponding to the first 0 bit, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

- $\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode;
- s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS; and
- n is the number of samples that contribute to the histogram – this may also be read from the HBWS.

Call the mean,

$$B = \bar{x}$$

Step 3, transmitting an MFTP pattern, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI of the last 1 bit. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

- $\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode;
- s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS; and
- n is the number of samples that contribute to the histogram – this may also be read from the HBWS.

Call the mean,

$$C = \bar{x}$$

Step 4, transmitting an MFTP pattern, construct a histogram based on n samples collected in the waveform epoch [0.45 UI to 0.55 UI] for the UI of the last 0 bit. The number of samples in a histogram (n) for the UI shall be greater than or equal to 100 and shall meet the requirement that:

$$1\,537\,(s / \bar{x})^2 \leq n$$

where:

- $\bar{x}$ is the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode;
- s is the standard deviation of the voltage samples in the histogram that may also be read from the HBWS; and
- n is the number of samples that contribute to the histogram – this may also be read from the HBWS.

Call the mean,

$$D = \bar{x}$$

Step 5, from A, B, C, and D obtained in steps 1 to 4, compute:

$$V_{Emphasis} = 20 \times \log_{10}[(A - B)/(C - D)]$$

The test for minimum device emphasis is passed if:

$$V_{Emphasis} \geq V_{EmphasisDevice}(min)$$

The test for maximum device emphasis is passed if:

$$V_{Emphasis} \leq V_{EmphasisDevice}(max)$$

The test for minimum host emphasis is passed if:

$$V_{Emphasis} \geq V_{EmphasisHost}(min)$$

The test for maximum host emphasis is passed if:

$$V_{Emphasis} \leq V_{EmphasisHost}(max)$$

See Table 54 and Table 55, according to 7.4.2 for $V_{EmphasisDevice}$ and $V_{EmphasisHost}$, otherwise the test for emphasis has not been passed.

7.6.33.4 Peak-Mode Device TX Emphasis measurement (Gen1i, Gen2i, Gen3i)

The Peak-Mode device TX Emphasis measurement applies to devices only and shall be based on the following values:

a) VMA: a mode (i.e., the most frequent value of a set of data) measurement; and
b) V_{P-P}: a peak to peak measurement with a repeating LFTP pattern (i.e., D30.3).

The VMA and V_{P-P} measurements shall be made with the transmitter device terminated through the compliance point into a lab-load as shown in Figure 249.

The VMA and V_{P-P} measurements shall be made using a HBWS with a histogram function with the following or an equivalent procedure:

1) calibrate the HBWS for measurement of a 3 GHz signal; and
2) determine VMA and V_{P-P} as shown in Figure 251. A sample size of 1,000 minimum to 2,000 maximum histogram hits for VMA shall be used to determine the values. The histogram is a combination of two histograms (i.e., an upper histogram for TX+ and a lower histogram for TX-). The histograms on the left represent the test pattern signal displayed on the right. VMA and V_{P-P} are determined by adding the values measured for TX+ and TX-.

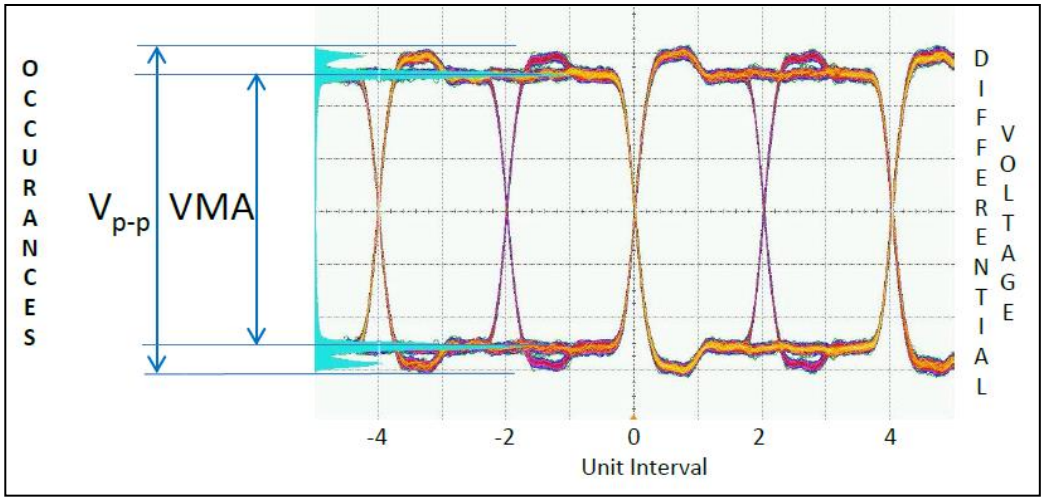


Figure 251 - Transmitter Emphasis Measurement (Alternate Method)

The following formula shall be used to calculate the transmitter equalization value:

$$\text{Transmitter equalization} = 20 \times \log(V_{P\text{-}P} / \text{VMA}) \text{ dB}$$

where:

$V_{P\text{-}P}$ is the peak to peak value; and
VMA is the mode value.

7.7 Interface states

7.7.1 Out Of Band (OOB) signaling

7.7.1.1 Out Of Band (OOB) signaling overview

There shall be three OOB signals used/detected by the Phy:

a) COMRESET;
b) COMINIT; and
c) COMWAKE.

COMINIT, COMRESET, and COMWAKE OOB signaling shall be achieved by transmission of either a burst of four Gen1 $ALIGN_P$ primitives or a burst composed of four Gen1 Dwords with each Dword composed of four D24.3 characters, each burst having a duration of 160 UI_{OOB}. Each burst is followed by idle periods (at common-mode levels), having durations as depicted in Figure 252 and Table 76.

Previous versions of Serial ATA allow only for the $ALIGN_P$ primitives as legitimate OOB signal content. The alternate OOB sequence defined in this section has different characteristics than the $ALIGN_P$ primitives in both the time and frequency domains. The use of alternate OOB signal content may lead to backwards incompatibility with Gen1 Phys designed to previous Serial ATA specification versions. Interoperability issues with Gen1 Phys designed to the earlier SATA specification arising from the use of alternate OOB signal content are the sole responsibility of the Phy transmitting this alternate content.

During OOB signaling transmissions, the differential and common mode levels of the signal lines shall comply with the same electrical specifications as for normal data transmission, as defined in 7.4. In Figure 252, COMRESET, COMINIT, and COMWAKE are shown. OOB signals are observed by detecting the temporal spacing between adjacent bursts of activity, on the differential pair. It is not required for a receiver to check the duration of an OOB burst.

Even though they are transmitted with apparent Gen1 timings, the OOB burst transmissions may be transmitted using Gen2 rise / fall times.

Any spacing less than or greater than the COMWAKE detector off threshold in Table 59 shall negate the COMWAKE detector output. The COMWAKE OOB signaling is used to bring the Phy out of the Partial or Slumber power down state as described in 8.4.4.3. The interface shall be held inactive for at least the maximum COMWAKE detector off threshold in Table 59 after the last burst to ensure far-end detector detects the negation properly. The device shall hold the interface inactive no more than the maximum COMWAKE detector off threshold plus two Gen1 Dwords (approximately 228.3 ns) at the end of a COMWAKE to prevent susceptibility to crosstalk.

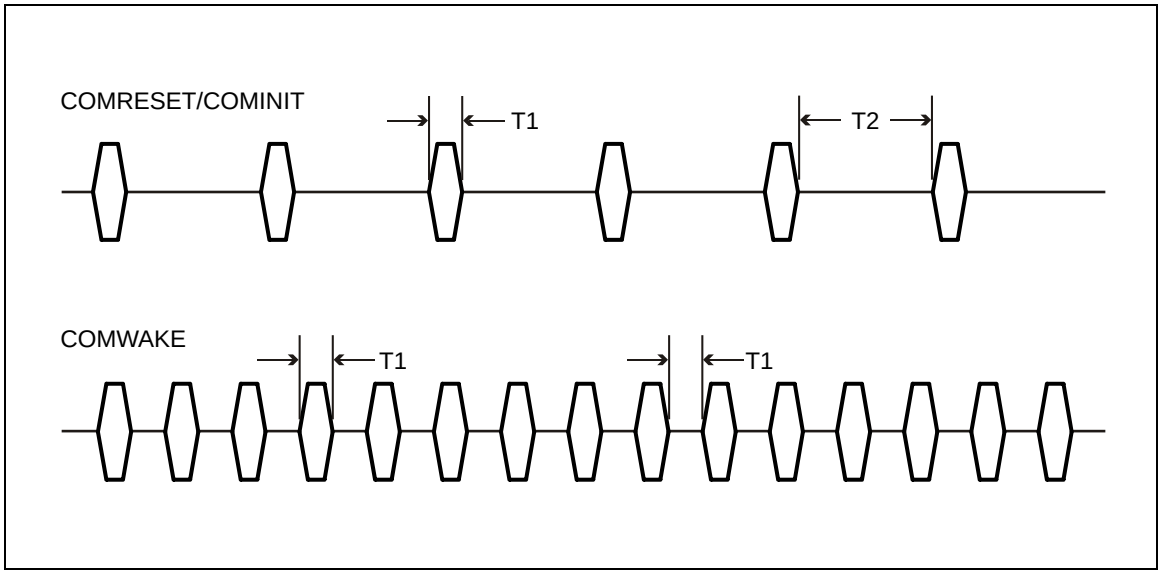


Figure 252 – OOB signals

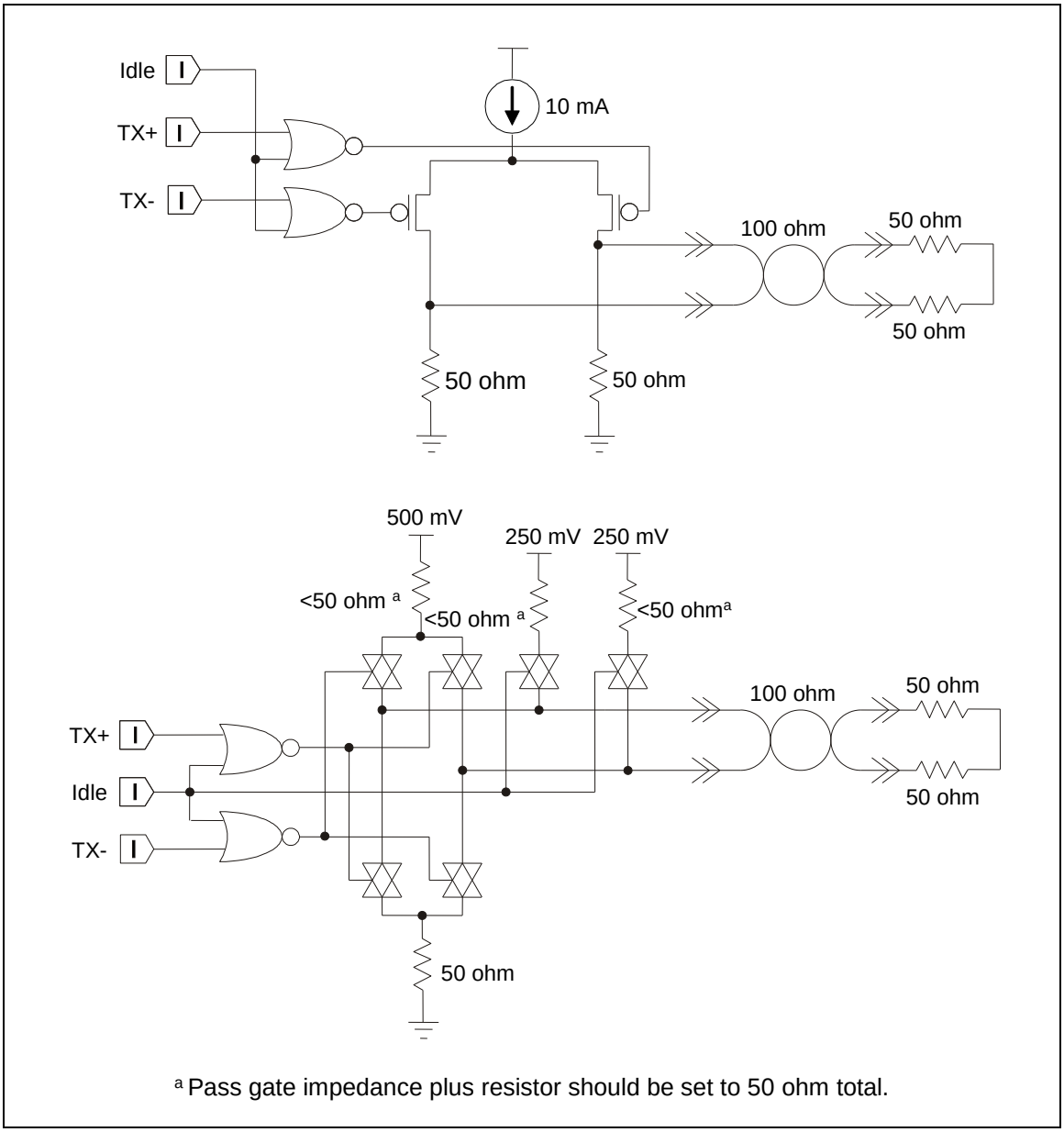
Table 76 – OOB signal times

Time	Value
T1	160 UI_{OOB} (Nom 106.7 ns)
T2	480 UI_{OOB} (Nom 320 ns)

7.7.1.2 Idle bus status

During the idle bus condition, the differential signal diminishes to zero while the common mode level remains.

Common-mode transients, shall not exceed the maximum amplitude levels (Vcm,ac) according to 7.4, and shall settle to within 25 mV of the previous state common mode voltage within Tsettle,cm, according to 7.4. Figure 253 shows several transmitter examples, and how the transition to and from the idle state may be implemented.



[a] Pass gate impedance plus resistor should be set to 50 ohm total.

Figure 253 – Transmitter examples (part 1 of 2)

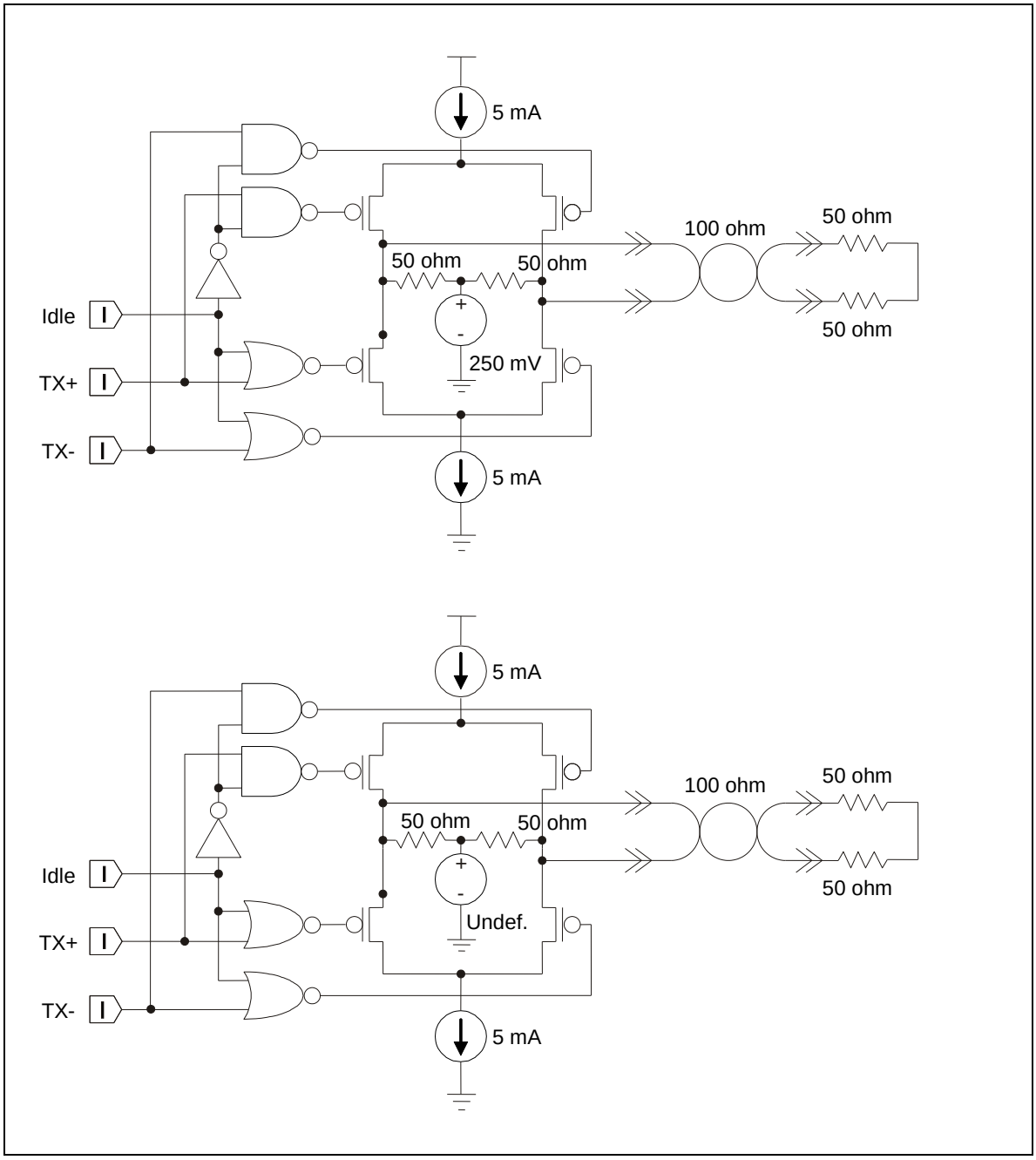


Figure 253 – Transmitter examples (part 2 of 2)

7.7.1.3 COMRESET

COMRESET always originates from the host controller, and forces a hardware reset in the device. It is indicated by transmitting bursts of data separated by an idle bus condition.

The OOB COMRESET signal shall consist of no less than six data bursts, including inter-burst temporal spacing.

The COMRESET signal shall be:

a) sustained/continued uninterrupted as long as the system hard reset is asserted;

b) started during the system hardware reset and ended some time after the negation of system hardware reset; or
c) transmitted immediately following the negation of the system hardware reset signal.

The host controller shall ignore any signal received from the device from the assertion of the hardware reset signal until the COMRESET signal is transmitted.

Each burst shall be 160 Gen1 UI long (e.g., 106.7 ns) and each inter-burst idle state shall be 480 Gen1 UI long (e.g., 320 ns). A COMRESET detector looks for four consecutive bursts with 320 ns spacing (nominal).

Any spacing less than 175 ns or greater than 525 ns shall invalidate the COMRESET detector output. The COMRESET interface signal to the Phy layer shall initiate the Reset sequence shown in Figure 254. The interface shall be held inactive for at least 525 ns after the last burst to ensure far-end detector detects the negation properly.

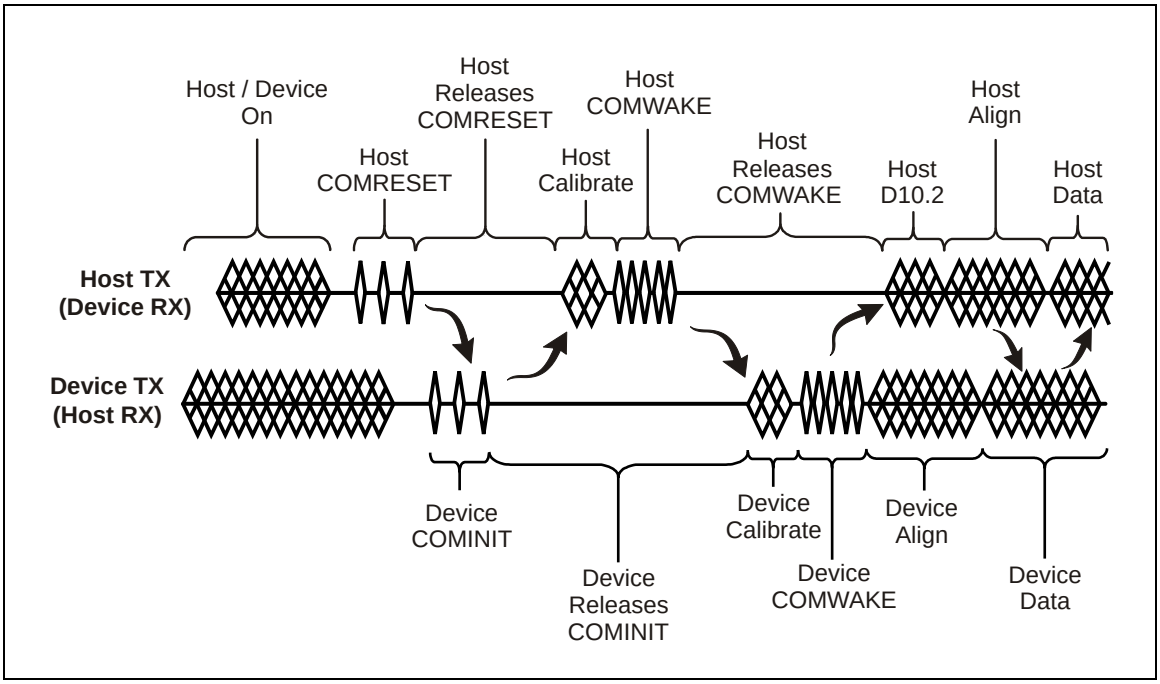


Figure 254 – COMRESET sequence

Description:
1) host/device are powered and operating normally with some form of active communication;
2) some condition in the host causes the host to issue COMRESET;
3) host releases COMRESET. Once the condition causing the COMRESET is released, the host releases the COMRESET signal and puts the bus in a quiescent condition;
4) device issues COMINIT. If the device detects the release of COMRESET, it responds with a COMINIT. This is also the entry point if the device is late starting. The device may initiate communications at any time by issuing a COMINIT;
5) host calibrates and issues a COMWAKE;
6) device responds, the device detects the COMWAKE sequence on its Rx pair and calibrates its transmitter (optional). Following calibration the device sends a six burst COMWAKE sequence and then sends a continuous stream of the $ALIGN_P$ sequence starting at the device's highest supported speed. After $ALIGN_P$ Dwords have been sent for 54.6 us (2 048 nominal Gen1 Dword times) without a response from the host as determined by detection of $ALIGN_P$ primitives received from the host, the device assumes that the host is unable to communicate at that speed. If additional speeds are available, then the device tries the

next lower supported speed by sending ALIGN$_P$ Dwords at that speed for 54.6 us (2 048 nominal Gen1 Dword times). This step is repeated for as many slower speeds as are supported. Once the lowest speed has been reached without response from the host, the device enters an error state;

7) host locks, after detecting the COMWAKE, the host starts transmitting D10.2 characters (see 7.8) at its lowest supported speed. Meanwhile, the host receiver locks to the ALIGN$_P$ sequence and, if ready, returns the ALIGN$_P$ sequence to the device at the same speed as received. A host shall be designed such that it acquires lock in 54.6 us (2 048 nominal Gen1 Dword times) at any given speed. The host should allow for at least 873.8 us (32 768 nominal Gen1 Dword times) after detecting the release of COMWAKE to receive the first ALIGN$_P$. This ensures interoperability with multi-generational and synchronous designs. If no ALIGN$_P$ is received within 873.8 us (32 768 nominal Gen1 Dword times) the host restarts the power-on sequence – repeating indefinitely until told to stop by the Application layer;
8) device locks, the device locks to the ALIGN$_P$ sequence and, if ready, sends SYNC$_P$ indicating it is ready to start normal operation; and
9) upon receipt of three back-to-back non-ALIGN$_P$ primitives, the communication link is established and normal operation may begin.

7.7.1.4 COMINIT

COMINIT always originates from the device and requests a communication initialization. It is electrically identical to the COMRESET signal except that it originates from the device and is sent to the host. It is used by the device to request a reset from the host in accordance to the sequence shown in Figure 255.

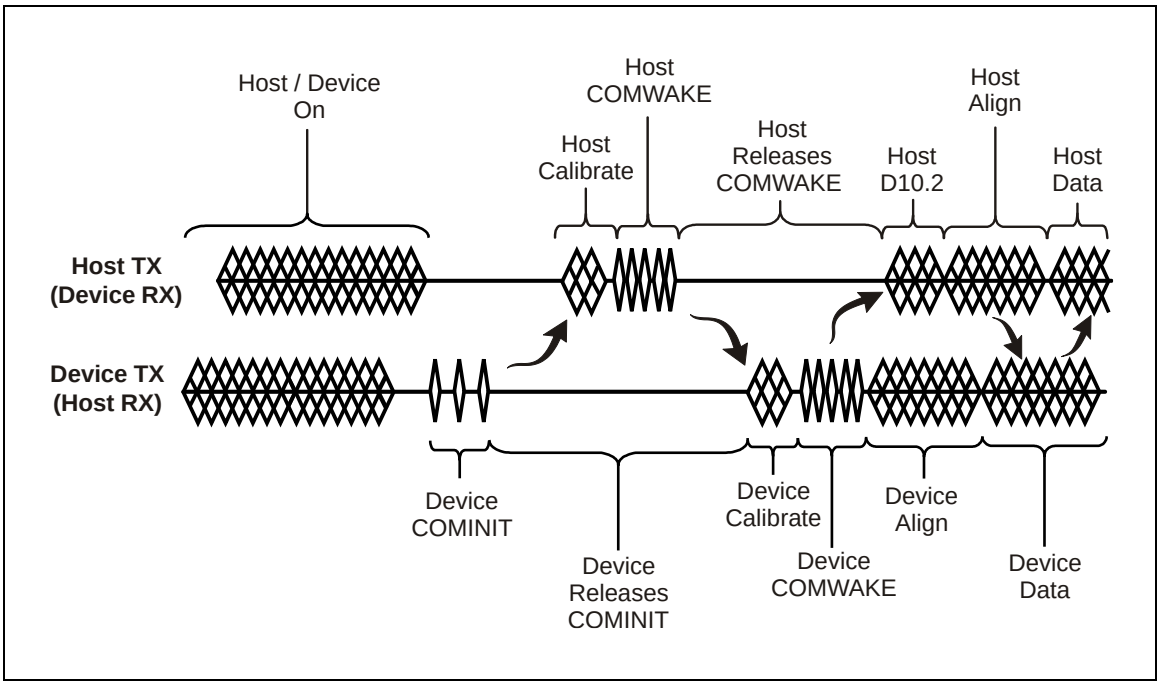


Figure 255 – COMINIT sequence

Description:

1) host/device are powered and operating normally with some form of active communication;
2) some condition in the device causes the device to issues a COMINIT;
3) host calibrates and issues a COMWAKE;

4) device responds, the device detects the COMWAKE sequence on its Rx pair and calibrates its transmitter (optional). Following calibration the device sends a six burst COMWAKE sequence and then sends a continuous stream of the $ALIGN_P$ sequence starting at the device's highest supported speed. After $ALIGN_P$ Dwords have been sent for 54.6 us (2 048 nominal Gen1 Dword times) without a response from the host as determined by detection of $ALIGN_P$ primitives received from the host, the device assumes that the host is unable to communicate at that speed. If additional speeds are available the device tries the next lower supported speed by sending $ALIGN_P$ Dwords at that speed for 54.6 us (2 048 nominal Gen1 Dword times). This step is repeated for as many slower speeds as are supported. Once the lowest speed has been reached without response from the host, the device enters an error state;
5) host locks, after detecting the COMWAKE, the host starts transmitting D10.2 characters (see 7.8) at its lowest supported speed. Meanwhile, the host receiver locks to the $ALIGN_P$ sequence and, if ready, returns the $ALIGN_P$ sequence to the device at the same speed as received. A host shall be designed such that it acquires lock in 54.6 us (2 048 nominal Gen1 Dword times) at any given speed. The host should allow for at least 873.8 us (32 768 nominal Gen1 Dword times) after detecting the release of COMWAKE to receive the first $ALIGN_P$. This ensures interoperability with multi-generational and synchronous designs. If no $ALIGN_P$ is received within 873.8 us (32 768 nominal Gen1 Dword times) the host restarts the power-on sequence – repeating indefinitely until told to stop by the Application layer;
6) device locks, the device locks to the $ALIGN_P$ sequence and, if ready, sends $SYNC_P$ indicating it is ready to start normal operation; and
7) upon receipt of three back-to-back non-$ALIGN_P$ primitives, the communication link is established and normal operation may begin.

7.7.1.5 COMWAKE

COMWAKE may originate from either the host controller or the device. It is signaled by transmitting six bursts of data separated by an idle bus condition.

The OOB COMWAKE signaling shall consist of no less than six data bursts, including inter-burst temporal spacing.

Each burst shall be 160 Gen1 UI long and each inter-burst idle state shall be 160 Gen1 UI long. A COMWAKE detector looks for four consecutive burst with a 106.7 ns spacing (nominal).

Any spacing less than 35 ns or greater than 175 ns shall invalidate the COMWAKE detector output. The COMWAKE OOB signaling is used to bring the Phy out of the Partial or Slumber interface power down state as defined in 8.1. The interface shall be held inactive for at least 175 ns after the last burst to ensure far-end detector detects the negation properly. The device shall hold the interface inactive no more then 228.3 ns (175 ns + two Gen1 Dwords) at the end of a COMWAKE to prevent susceptibility to crosstalk.

7.7.1.6 Design example (informative)

This section includes one possible design example for detecting COMRESET/COMINIT and COMWAKE. Other design implementations are possible as long as they adhere to the requirements listed in this specification.

The output of the squelch detector is fed into four frequency comparators. If the period is within the window determined by the RC time constants for three consecutive cycles, the appropriate signal is asserted.

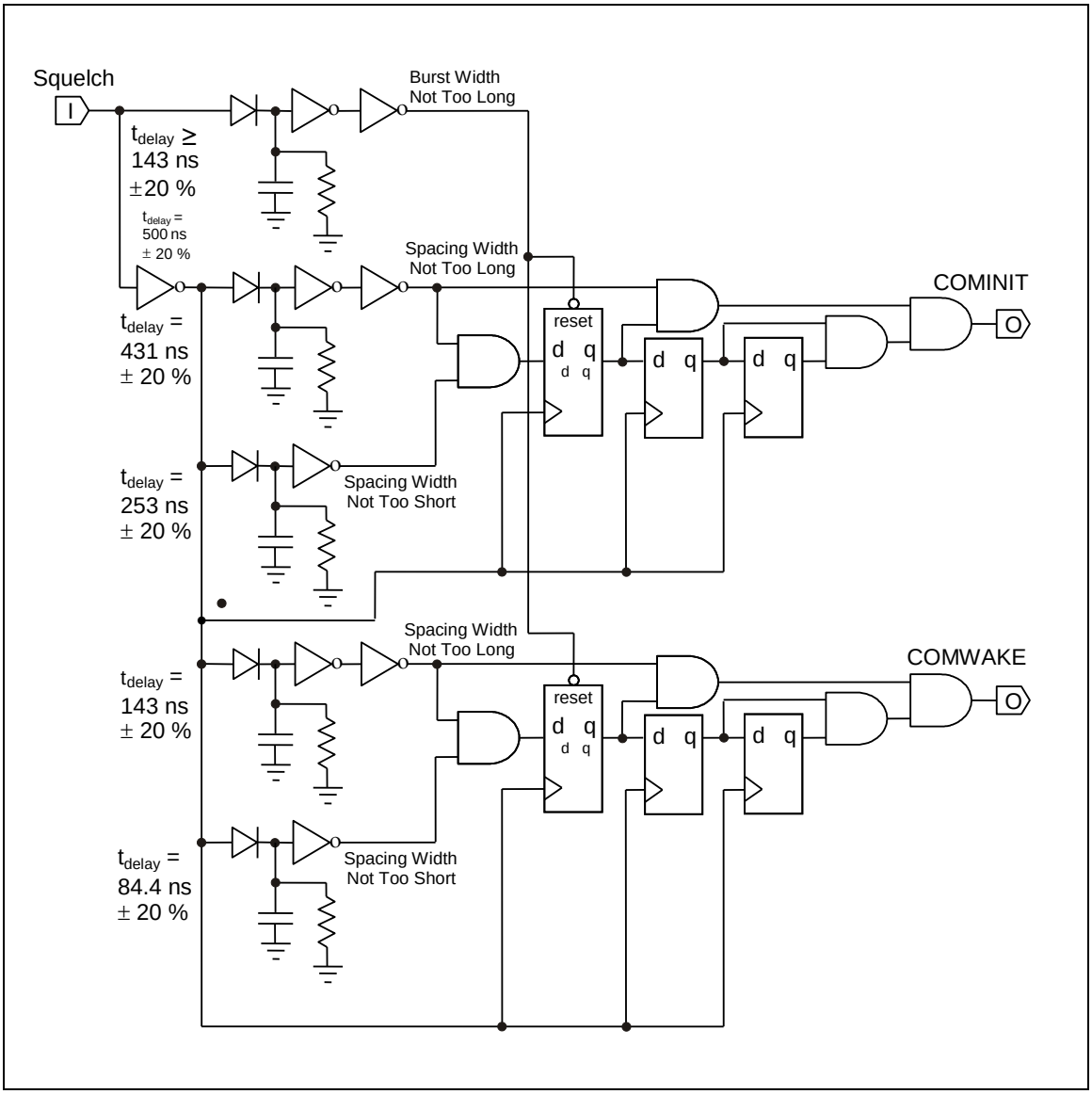


Figure 256 – OOB signal detector

The Squelch detector example below makes use of a receiver with built-in hysteresis to filter out any signal not meeting the minimum amplitude. The squelch detector receiver shall be true differential to ensure common-mode noise is rejected.

The full-swing output is fed into a pulse generator that charges up the capacitor through the diode. In the absence of signal, a resistor discharges the capacitor to ground. The circuit outputs a true signal if the capacitor voltage is below the turn-on threshold of the Schmitt trigger buffer – indicating insufficient signal level. This circuit shall be enabled in all power management states and should, therefore, be implemented with a small power budget.

Figure 257, like the OOB Signal Detector figure shown in Figure 256, is intended to show functionality (informative) only, and other solutions may be used to improve power consumption as long as they comply to the electrical specifications as defined in 7.4, for the worst case noise environment (common-mode) conditions.

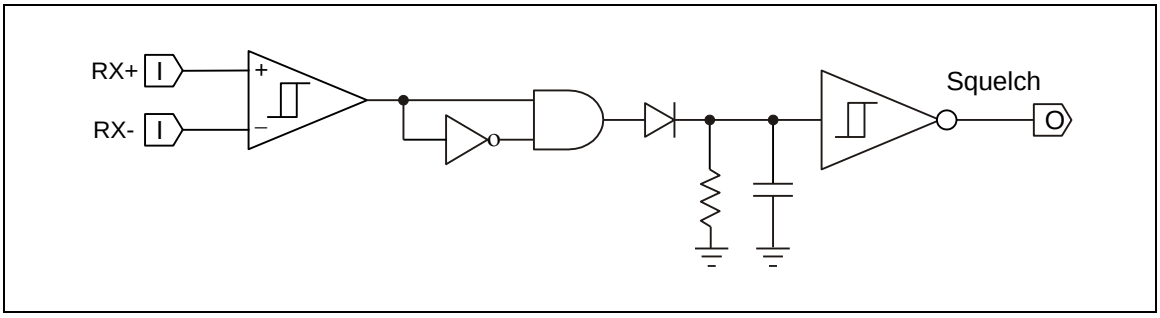


Figure 257 – Squelch detector

7.7.2 Idle bus condition

During power management states (Partial and Slumber), the electrical interface shall maintain the proper common-mode levels, according to 7.4, with zero differential on both signal pairs (all four conductors at 250 mV) for all interface scenarios, except for the case where both, the device and the host-controller, are AC coupled and the conductor pairs are allowed to float.

All transmitter designs shall ensure that transition to and from the idle bus condition does not result in a disturbance in the differential baseline on the conductors. To accomplish this, an AC coupled transmitter shall hold its outputs at zero differential with the same common-mode level as normal operation while in the Partial power management mode. If operating in the Slumber power management mode, the common mode level of the AC coupled transmitter is allowed to float (while maintaining zero differential) as long as it remains within the limits 7.4.

It is unacceptable to hold the Tx outputs at a logical zero or one state during the idle bus condition since this results in a baseline shift while communications are resumed.

7.8 Elasticity buffer management

For non-tracking implementations elasticity buffer circuitry may be required to absorb the slight differences in frequencies between the host and device. The greatest frequency difference results from a SSC compliant device talking to a non-SSC device. The average frequency difference is just over 0.25 % with excursions as much as 0.5 %.

This specification is written to support both tracking and non-tracking architectures. A non-tracking architecture shall contain the elasticity buffer within the Phy layer.

> NOTE 35 - Note that since this elasticity buffer is designed to have finite length, there needs to be a mechanism at the Phy layer protocol level that allows this receiver buffer to be reset without dropping or adding any bits to the data stream.

This is especially important during reception of long continuous streams of data. This Phy layer protocol not only supports oversampling architectures but also accommodates unlimited frame sizes (the frame size is limited by the CRC polynomial).

The Link layer shall keep track of a resettable counter that rolls over at most every 1 024 transmitted characters (i.e., 256 Dwords). Prior to, or at the pre-roll-over point (i.e., all ones), the Link layer shall trigger the issuance of two consecutive $ALIGN_P$ primitives that shall be included in the Dword count.

After communications have been established, the first and second Words out of the Link layer shall be the two consecutive $ALIGN_P$ primitives, followed by at most 254 non-$ALIGN_P$ Dwords. The cycle repeats starting with two consecutive $ALIGN_P$ primitives. The Link may issue more than a single

instance of two consecutive $ALIGN_P$ primitives but shall not send an odd number of $ALIGN_P$ primitives (i.e., $ALIGN_P$ primitives are always sent in pairs) except as noted for retimed loopback (see 10.5.10).

$ALIGN_P$ consists of four characters as shown in Table 77.

Table 77 – $ALIGN_P$

(rd+)	(rd-)	
11 0000 0101b	00 1111 1010b	Align1 (K28.5)
01 0101 0101b	01 0101 0101b	Align2 (D10.2)
01 0101 0101b	01 0101 0101b	Align3 (D10.2)
11 0110 0011b	00 1001 1100b	Align4 (D27.3)

8 OOB and Phy power states

8.1 Interface power states

Serial ATA interface power states are controlled by the device and host controller. The interface power states are defined as described in Table 78.

Table 78 – Interface power states

State	Description
PHYRDY	The Phy logic and main PLL are both on and active. The interface is synchronized and capable of receiving and sending data.
Partial	The Phy logic is powered, but is in a reduced power state. Both signal lines on the interface are at a neutral logic state (common mode voltage). The exit latency from this state shall be no longer than 10 us unless Automatic Partial to Slumber transitions is supported. If Automatic Partial to Slumber Transitions are enabled the exit latency from this state shall be no longer than the maximum Slumber exit latency.
Slumber	The Phy logic is powered but is in a reduced power state. The common mode level of the AC coupled transmitter is allowed to float (while maintaining zero differential) as long as it remains within the limits cited in Table 52 entry for AC coupled common mode voltage. The exit latency from this state shall be no longer than 10 ms.
DevSleep	The Phy logic may be powered down. The common mode level of the AC coupled transmitter is allowed to float (while maintaining zero differential) as long as it remains within the limits cited in Table 52 entry for AC coupled common mode voltage. The exit latency from this state shall be no longer than 20 ms, unless otherwise specified by DETO in Identify Device Data log (see 13.7.11.4).

8.2 Asynchronous signal recovery (optional)

8.2.1 Asynchronous signal recovery overview

Phys may support asynchronous signal recovery for those applications where the usage model of device insertion into a receptacle (power applied at time of insertion) does not apply.

If signal is lost, both the host and the device may attempt to recover the signal. A host or device shall determine loss of signal as represented by a transition from PHYRDY to PHYRDYn that is associated with entry into states LS1:L_NoCommErr or LS2:L_NoComm within the Link layer.

NOTE 36 - Note that negation of PHYRDY does not always constitute a loss of signal (e.g., Phy transition to Partial/Slumber).

Recovery of the signal is associated with exit from state LS2:L_NoComm. If the device attempts to recover the signal before the host by issuing a COMINIT, the device shall return its signature following completion of the OOB sequence that included COMINIT. If a host supports asynchronous signal recovery, and the host receives an unsolicited COMINIT, the host shall issue a COMRESET to the device. An unsolicited COMINIT is a COMINIT that was not in response to a preceding COMRESET, as defined by the host not being in the HP2:HR_AwaitCOMINIT state while the COMINIT signal is first received. If a host does not support asynchronous signal recovery, see 8.4.2.

If a COMRESET is sent to the device in response to an unsolicited COMINIT, the host shall set the Status register to 7Fh and shall set all other Shadow Command Block Registers to FFh. If the COMINIT is received in response to the COMRESET that is associated with entry into state

HP2B:HR_AwaitNoCOMINIT, the Shadow Status register value shall be updated to either FFh or 80h to reflect that a device is attached.

8.2.2 Device Sleep and Asynchronous Signal Recovery

If the host supports the Device Sleep feature (see 13.3.1), then the host should support Asynchronous Signal Recovery. If the device supports the Device Sleep feature, then the device should support Asynchronous Signal Recovery.

8.2.3 Unsolicited COMINIT usage (informative)

Issuing a COMRESET to the device causes the device to lose software settings, other than the cases where software settings preservation is supported as defined in 13.5. If the COMRESET was due to asynchronous signal recovery and legacy mode (see 4.1.1.82) software is in use, software does not replace the lost software settings. Issuing a non-commanded COMRESET to the device should be minimized in order to ensure robust operation with legacy mode software and avoid inadvertent loss of critical software settings.

The use of unsolicited COMINIT was originally intended to only be used if the signal is lost between host and device. Based on the Host Phy Initialization state machine, the host shall assume that if receiving an unsolicited COMINIT that either a new device was connected or that the cable was unplugged and communication was lost to the device. The proper host response to an unsolicited COMINIT is to issue a COMRESET, putting the device into a known state. The device issuing an unsolicited COMINIT leads to a COMRESET from the host that is able to change the software settings of the device in such a way that legacy mode software is unable to recover. To minimize potential for exposure to such indeterminate behavior, the device should only issue an unsolicited COMINIT if the Phy voltage threshold falls below the minimum value or as a last resort in error recovery.

8.3 OOB and Signature FIS return (informative)

After an OOB sequence, some devices compliant to previous revisions of this specification may send a Register Device to Host FIS with the device signature only if the device recognized COMRESET during the OOB. To ensure a robust host solution for compatibility with these older devices, the host may ensure at a system power-on event that the device always receives a valid COMRESET after power is determined good at the device. Hot plug aware software shall ensure that the device always receives a COMRESET on a hot plug event.

One mechanism as a host workaround is to implement the following software procedure while determining device presence:

1) wait for SError.DIAG.X to be set to one;
2) clear SError.DIAG.X to zero by writing a one to that bit location;
3) issue a COMRESET to the device (a valid COMINIT was received to set the X bit to one, thus power at the device is known to be good);
4) wait up to 10 ms for SError.DIAG.X to be set to one;
5) if SError.DIAG.X is not set after 10 ms, go back to step 3 or exit if number of retries is exceeded; and
6) at this point, the device is now required to transmit a Register Device to Host FIS with the device signature.

Other methods for ensuring that the device receives a COMRESET in these conditions are possible.

8.4 Power-on sequence state machine

8.4.1 Power-on sequence state machine overview

The following state machines specify the expected behavior of the host and device Phy from power-on to the establishment of an active communication channel.

In those states where the Phy relies on detection of received $ALIGN_P$ primitives or comma sequences for state transitions, the Phy shall ensure accurate detection of the $ALIGN_P$ primitives at the compatible signaling speed, with adequate implementation safeguards to ensure that there is no misdetection of $ALIGN_P$ in the HP6:HR_AwaitAlign state in light of aliasing effects given the different data rates of $ALIGN_P$ primitives and D10.2's in the incoming data streams.

8.4.2 Host Phy initialization state machine

As defined in 7.7.1.4, reception of a COMINIT signal shall cause the host to reinitialize communications with the device. Implementations that do not support asynchronous signal recovery shall unconditionally force the Host Phy Initialization state machine to transition to the HP2B:HR_AwaitNoCOMINIT state if a COMINIT is received regardless of other conditions. Implementations that do support asynchronous signal recovery shall unconditionally force the Host Phy Initialization state machine to transition to the HP1:HR_Reset state (see Figure 258) if an unsolicited COMINIT is received regardless of other conditions; if the COMINIT is not unsolicited the implementation shall force the Host Phy Initialization state machine to transition to the HP2B:HR_AwaitNoCOMINIT state regardless of other conditions. Reception of COMINIT is effectively an additional transition into the HP2B:HR_AwaitNoCOMINIT or HP1:HR_Reset state that appears in every Host Phy state. For the sake of brevity, this implied transition has been omitted from all the states.

A state variable called ResumePending is used to track whether the Host Phy has been to a power management state such that re-establishing communications is as a result of a resume from a low power state. If a COMWAKE signal is not received while resuming from a low power state, the Host Phy shall allow the device to retransmit COMWAKE and shall not transmit a COMRESET to the device unless a COMRESET is explicitly triggered from a higher layer. If a COMWAKE signal is not received from the device while resuming from a low power state, the host may retransmit COMWAKE to the device.

Designs that support asynchronous signal recovery have a state variable referred to as RetryInterval that determines the speed that optional signal recovery polling is attempted. The value for RetryInterval shall be no shorter than 10 ms. Implementations that do not implement optional retry polling may consider the RetryInterval value to be infinite.

HP1: HR_Reset [a]	Transmit COMRESET [b c d]. If asynchronous signal recovery is supported, then clear ResumePending to zero.

1. Power-on reset and explicit reset request negated.	→	HR_AwaitCOMINIT
2. Power-on reset or explicit reset request asserted.	→	HR_Reset

[a] This state is entered asynchronously any time in response to power-on reset or an explicit reset request. For hosts supporting asynchronous signal recovery, this state is entered in response to receipt of a COMINIT signal from any state other than the HP2:HR_AwaitCOMINIT or the HP2B:HR_AwaitNoCOMINIT state.
[b] Shall transmit COMRESET for a minimum of 6 bursts (and a multiple of 6).
[c] As defined in 7.7.1.3, COMRESET may be transmitted for the duration of this state, or it may be transmitted starting in this state and cease transmission after departure of this state, or it may be transmitted upon departure of this state.
[d] Hosts that support asynchronous signal recovery shall complete transmission of COMRESET in response to a received COMINIT that causes a transition to this state within 10 ms of the de-qualification of the received COMINIT signal.

HP2: HR_AwaitCOMINIT	Interface quiescent.

1. COMINIT detected from device.	→	HR_AwaitNoCOMINIT
2. COMINIT not detected from device and (asynchronous signal recovery not supported or RetryInterval not elapsed since entry into the HP2:HR_AwaitCOMINIT state).	→	HR_AwaitCOMINIT
3. COMINIT not detected from device and asynchronous signal recovery supported and RetryInterval elapsed since entry into the HP2:HR_AwaitCOMINIT state.	→	HR_Reset

HP2B: HR_AwaitNoCOMINIT [a]	Interface quiescent.

1. COMINIT not detected from device.	→	HR_Calibrate
2. COMINIT detected from device.	→	HR_AwaitNoCOMINIT

[a] For hosts that do not support asynchronous signal recovery, this state is entered asynchronously any time in response to COMINIT unless during a power-on reset or an explicit reset request that case HP1 is entered.

HP3: HR_Calibrate	Perform calibration [a].

1. Calibration complete or bypass not implemented.	→	HR_COMWAKE
2. Calibration not complete.	→	HR_Calibrate

[a] Calibration is optional. If bypassed or not implemented, proceed directly to HR_COMWAKE.

HP4: HR_COMWAKE	Transmit COMWAKE.

1. COMWAKE not detected from device.	→	HR_AwaitCOMWAKE
2. COMWAKE detected from device.	→	HR_AwaitNoCOMWAKE

Figure 258 – Host Phy initialization state machine (part 1 of 5)

HP5: HR_AwaitCOMWAKE	Interface quiescent.	
1. COMWAKE detected from device.	→	HR_AwaitNoCOMWAKE
2. COMWAKE not detected from device and (asynchronous signal recovery not supported or RetryInterval not elapsed since entry into the HP5:HR_AwaitCOMWAKE state).	→	HR_AwaitCOMWAKE
3. COMWAKE not detected from device and asynchronous signal recovery supported and RetryInterval elapsed since entry into the HP5:HR_AwaitCOMWAKE state and ResumePending = 0.	→	HR_Reset
4. COMWAKE not detected from device and asynchronous signal recovery supported and RetryInterval elapsed since entry into the HP5:HR_AwaitCOMWAKE state and ResumePending = 1.	→	HR_COMWAKE

HP5B: HR_AwaitNoCOMWAKE	Interface quiescent	
1. COMWAKE not detected from device.	→	HR_AwaitAlign
2. COMWAKE detected from device.	→	HR_AwaitNoCOMWAKE

Figure 258 –Host Phy initialization state machine (part 2 of 5)

HP6: HR_AwaitAlign	Host transmits D10.2 characters at lowest supported rate [b e]	
1. If reset speed negotiation is not supported and $ALIGN_P$ detected from device (at any supported speed) [c]. If reset speed negotiation is supported and $ALIGN_P$ detected from device at expected speed.	→	HR_AdjustSpeed
2. $ALIGN_P$ not detected from device and 873.8 us (i.e., 32 768 Gen1 Dwords) has elapsed since entry to HR_AwaitAlign. If reset speed negotiation is supported and $ALIGN_P$ is detected at a lower speed.	→	HR_Reset [a d f]
3. $ALIGN_P$ not detected from device and less than 873.8 us (i.e., 32 768 Gen1 Dwords) has elapsed since entry to HR_AwaitAlign.	→	HR_AwaitAlign

[a] Host retries the power-on sequence indefinitely unless explicitly turned off by the Application layer.
[b] Host shall start transmitting D10.2 characters no later than 533 ns (i.e., 20 Gen1 Dwords) after COMWAKE is negated as specified in the OOB signaling section.
[c] Host designers should be aware that the device is allowed 53.3 ns (i.e., 2 Gen1 Dwords) after releasing COMWAKE (by holding the idle condition for more than 175 ns) to start sending characters. Until this occurs, the bus is at an idle condition and may be susceptible to crosstalk from other devices. Care should be taken so that crosstalk during this window does not result in a false detection of an $ALIGN_P$.
EXAMPLE - A compliant host may detect the negation of COMWAKE in as little as 112 ns, such a host is recommended to wait at least 116.3 ns (i.e., 175 ns + 53.3 ns – 112 ns) after detecting the release of COMWAKE to start looking for $ALIGN_P$ primitives.
[d] The Host Phy Initialization state machine may use the transition to HR_Reset as a method of speed negotiation.
[e] The device may respond with D10.2 if out of lock (see DR_SendAlign).
[f] If $ALIGN_P$ is detected at a lower speed, the host may transition to HR_Reset before the 873.8 us timeout thus reducing the time for speed negotiation (RSN).

HP7: HR_SendAlign	Transmit $ALIGN_P$ at speed detected	
1. Three back-to-back non-$ALIGN_P$ primitives [b] detected from device.	→	HR_Ready
2. Three back-to-back non-$ALIGN_P$ primitives not detected from device.	→	HR_SendAlign [a]

[a] Host retries indefinitely unless explicitly turned off by the Application layer
[b] Non-$ALIGN_P$ primitives may be detected by the presence of the K28.3 control character in the Byte 0 position.

Figure 258 –Host Phy initialization state machine (part 3 of 5)

HP8: HR_Ready	Transmit Word from Link [a].	
1. Partial signal from Link asserted.	→	HR_Partial
2. Slumber signal from Link asserted.	→	HR_Slumber
3. No power management request received and (asynchronous signal recovery not supported, or signal recovery poll not initiated [b], or received signal detected).	→	HR_Ready
4. No power management request received, and asynchronous signal recovery supported, and received signal not detected, and signal recovery poll initiated[b].	→	HR_Reset

[a] PHYRDY asserted only if in the HR_Ready state and the Phy is maintaining synchronization with the incoming signal to its receiver and is transmitting a valid signal on its transmitter.
[b] The latency that a host elects to initiate an optional signal recovery poll is implementation specific but shall be greater than the $ALIGN_P$ transmit interval.

HP9: HR_Partial	Interface quiescent. If asynchronous signal recovery is supported, then set ResumePending = 1.	
1. Partial signal from Link negated and no COMWAKE detected from device [a b].	→	HR_COMWAKE
2. Partial signal from Link negated and COMWAKE detected from device [a b].	→	HR_AwaitNoCOMWAKE
3. Slumber signal from the Link asserted and host Automatic Partial to Slumber transitions are supported [c].	→	HR_Slumber
4. Partial signal from Link asserted.	→	HR_Partial

[a] Host Phy shall remember if COMWAKE was detected during Partial to determine if the wakeup request originated from the host or the Phy.
[b] The host Phy may take this transition only after it has recovered from Partial mode and the Phy is prepared to initiate communications. If Phy has not yet recovered from the Partial mode it shall remain in this state.
[c] The host Phy may transition to HR_Slumber if host Automatic Partial to Slumber transitions are supported by the host and device. See 13.17 for more information regarding Automatic Partial to Slumber transitions.

Figure 258 –Host Phy initialization state machine (part 4 of 5)

HP10: HR_Slumber	Interface quiescent. If asynchronous signal recovery is supported, then set ResumePending = 1.	
1. Slumber signal from Link negated and no COMWAKE detected from device [a] [b].	→	HR_COMWAKE
2. Slumber signal from Link negated and COMWAKE detected from device [a] [b].	→	HR_AwaitNoCOMWAKE
3. Slumber signal from Link asserted.	→	HR_Slumber
[a] Host Phy shall remember if COMWAKE was detected during Slumber to determine if the wakeup request originated from the host or the Phy. [b] The host Phy may take this transition only after it has recovered from Slumber mode and the Phy is prepared to initiate communications. If Phy has not yet recovered from the Slumber mode, then it shall remain in this state.		

HP11: HR_AdjustSpeed	Interface undefined but not quiescent [a]	
1 Transition to appropriate speed completed.	→	HR_SendAlign
2 Transition to appropriate speed not completed.	→	HR_AdjustSpeed
[a] Some implementations may undergo a transient condition where invalid signals are transmitted during the change in their internal transmission/reception speed. The host may transmit invalid signals for a period of up to 53 ns (i.e., 2 Gen1 Dwords) during the speed transition. Transmit jitter and UI timing requirements may not be met during this period but shall be met for all other bits transmitted in this state. A phase shift may occur across the speed transition time.		

Figure 258 –Host Phy initialization state machine (part 5 of 5)

8.4.3 Device Phy initialization state machine

As defined in 7.7.1.3, reception of a COMRESET signal shall be treated by the device as a hardware reset signal and shall unconditionally force the Device Phy Initialization state machine (see Figure 259) to transition to the DP1:DR_Reset initial state regardless of other conditions. Reception of COMRESET is effectively an additional transition into the DP1:DR_Reset state that appears in every Device Phy state. For the sake of brevity, this implied transition has been omitted from all the states.

DP1: DR_Reset [a]	Interface quiescent	
1. COMRESET not detected and power-on reset negated.	→	DR_COMINIT
2. COMRESET detected or power-on reset asserted.	→	DR_Reset
[a] This state is entered asynchronously any time in response to power-on reset or receipt of a COMRESET signal from the host.		

Figure 259 –Device Phy initialization state machine (part 1 of 4)

DP2: DR_COMINIT	Transmit COMINIT [a] [b]	
1. Unconditional	→	DR_AwaitCOMWAKE

[a] COMINIT transmitted for a 6 bursts duration.
[b] Devices shall respond with a COMINIT signal at the completion of the device power up sequence or within 10 ms of the de-qualification of a received COMRESET signal.

DP3: DR_AwaitCOMWAKE	Interface quiescent	
1. COMWAKE detected from host.	→	DR_AwaitNoCOMWAKE
2. COMWAKE not detected from host, and (asynchronous signal recovery not implemented, or RetryInterval not elapsed since entry into the DP3:DR_AwaitCOMWAKE state).	→	DR_AwaitCOMWAKE
3. COMWAKE not detected from host and asynchronous signal recovery implemented and RetryInterval elapsed since entry into the DP3:DR_AwaitCOMWAKE state.	→	DR_Reset

DP3B: DR_AwaitNoCOMWAKE	Interface quiescent	
1. COMWAKE not detected from host and part of power-on reset sequence [a].	→	DR_Calibrate
2. COMWAKE not detected from host and part of Partial/Slumber awake sequence [a].	→	DR_COMWAKE
3. COMWAKE detected from host.	→	DR_AwaitNoCOMWAKE

[a] Device shall remember if it was sent to Partial or Slumber mode for proper wakeup action.

DP4: DR_Calibrate	Perform calibration [a]	
1. Calibration complete or bypass not implemented.	→	DR_COMWAKE
2. Calibration not complete.	→	DR_Calibrate

[a] Calibration is optional. If bypassed or not implemented, proceed directly to DR_COMWAKE.

DP5: DR_COMWAKE	Transmit COMWAKE	
1. Unconditional	→	DR_SendAlign

Figure 259 – Device Phy initialization state machine (part 2 of 4)

DP6: DR_SendAlign	Transmit $ALIGN_P$ [a b c e]		
	1. $ALIGN_P$ detected from host (device locked to incoming data) [d].	→	DR_Ready
	2. $ALIGN_P$ not detected from host and $ALIGN_P$ primitives transmitted for 54.6 us (i.e., 2 048 [e] Gen1 $ALIGN_P$ primitives) at speed other than lowest [f].	→	DR_ReduceSpeed
	3. $ALIGN_P$ not detected from host and $ALIGN_P$ primitives transmitted for 54.6 us (i.e., 2 048 [e] Gen1 $ALIGN_P$ primitives) at lowest speed [f].	→	DR_Error
	4. $ALIGN_P$ not detected from host and $ALIGN_P$ primitives transmitted for less than 54.6 us (i.e., 2 048 Gen1 $ALIGN_P$ primitives).	→	DR_SendAlign
	[a] If this is part of a recovery from a Slumber or Partial power management state, the device shall send $ALIGN_P$ at the previously negotiated speed. For all other cases, $ALIGN_P$ should be sent at the device's fastest supported speed. [b] $ALIGN_P$ primitives should be sent only at valid frequencies (i.e., if PLL not locked, send D10.2). [c] After COMWAKE is released as specified in the OOB signaling section, the device shall ensure the interface is active (not quiescent). [d] Device designers should be aware that the host is allowed 533 ns (i.e., 20 Gen1 Dwords) after detecting the negation of COMWAKE to start sending D10.2 characters. Until this occurs, the bus is in an idle condition and may be susceptible to crosstalk from other devices. Care should be taken so that crosstalk during this window does not result in a false detection of an $ALIGN_P$. Devices may extend this timeout up to an additional 54.6 us (i.e., 2 048 Gen1 Dwords) (for a max total of 109.2 us), as necessary to allow their receiver time to lock to the host $ALIGN_P$. [e] Device shall not leave the bus idle more than 53.3 ns (i.e., 2 Gen1 Dwords) longer than the required 175 ns to negate COMWAKE. [f] If this is part of a recovery from the Slumber or Partial power management state, the device Phy shall not reduce its speed in response to failure to establish communications. Upon failing to establish communications it should instead transition directly to the DR_Error state to initiate a retry of the COMWAKE sequence.		

DP7: DR_Ready [a]	Transmit Word from Link		
	1. Partial signal from Link asserted.	→	DR_Partial
	2. Slumber signal from Link asserted.	→	DR_Slumber
	3. No power management request received and (asynchronous signal recovery not supported or received signal detected).	→	DR_Ready
	4. No power management request received, and asynchronous signal recovery supported, and received signal not detected.	→	DR_Error
	[a] PHYRDY asserted only if in the DR_Ready state and the Phy is maintaining synchronization with the incoming signal to its receiver and is transmitting a valid signal on its transmitter.		

Figure 259 – Device Phy initialization state machine (part 3 of 4)

DP8: DR_Partial	Interface quiescent		
	1. Partial signal from Link negated [a]	→	DR_COMWAKE
	2. Partial signal from Link negated, and COMWAKE detected from host [a].	→	DR_AwaitNoCOMWAKE
	3. Slumber signal from Link asserted, and device Automatic Partial to Slumber transitions enabled[b].	→	DR_Slumber
	4. Partial signal from Link asserted.	→	DR_Partial
	[a] The device Phy may take this transition only after it has recovered from Partial mode and the Phy is prepared to initiate communications. If Phy has not yet recovered from the Partial mode it shall remain in this state. [b] The device Phy may transition to DR_Slumber if device Automatic Partial to Slumber transitions are enabled. See 13.17 for more information regarding Automatic Partial to Slumber transitions.		

DP9: DR_Slumber	Interface quiescent		
	1. Slumber signal from Link negated [a].	→	DR_COMWAKE
	2. Slumber signal from Link negated, and COMWAKE detected from host [a].	→	DR_AwaitNoCOMWAKE
	3. Slumber signal from Link asserted.	→	DR_Slumber
	[a] The device Phy may take this transition only after it has recovered from Slumber mode and the Phy is prepared to initiate communications. If Phy has not yet recovered from the Slumber mode, then it shall remain in this state.		

DP10: DR_ReduceSpeed	Interface quiescent		
	1. Transition to a slower speed complete.	→	DR_SendAlign [a]
	2. Transition to a slower speed not complete.	→	DR_ReduceSpeed
	[a] Transition to a new speed is defined as being complete if the device is accurately transmitting a valid signal within the defined signaling tolerances for that speed.		

DP11: DR_Error	Interface quiescent		
	1. Error not due to failure to resume and ((asynchronous signal recovery not supported), or (asynchronous signal recovery supported and RetryInterval not elapsed since entry into the DP11:DR_Error state)).	→	DR_Error
	2. Resume from Slumber or Partial failed.	→	DR_COMWAKE
	3. Error not due to failure to resume, and asynchronous signal recovery supported, and RetryInterval elapsed since entry into the DP11:DR_Error state.	→	DR_Reset

Figure 259 – Device Phy initialization state machine (part 4 of 4)

8.4.4 Speed negotiation

8.4.4.1 Speed negotiation overview

In state HP6:HR_AwaitAlign, it is possible for the host to receive a signal at a speed different than what the host is awaiting (i.e., a Gen1 host may receive a Gen2 signal from the device or a Gen2 host may receive a Gen1 signal from the device). Some data recovery circuits may return unpredictable recovered data if presented with an incoming signaling speed higher than supported. Conversely, signal aliasing effects may impact the accuracy of decoded signals while a lower signaling speed than expected is received. Because the recovered data may be invalid, implementations shall insure that $ALIGN_P$ primitives are accurately decoded in the HP6:HR_AwaitAlign state in light of the possibility of recovered data in this state being the result of falsely decoding a signal at a speed different than the host is anticipating.

To reduce susceptibility to false $ALIGN_P$ detection/handshake, receivers should fully qualify the entire received $ALIGN_P$ primitives instead of relying on qualifying only a portion of each (e.g., just the comma sequence). Additional means for ensuring that the transition from the HP6:HR_AwaitAlign is accurately traversed and not traversed in response to a spurious signal from the data recovery circuit is to ensure that a series of contiguous $ALIGN_P$ primitives are successfully decoded. Other possible means for ensuring accuracy of the $ALIGN_P$ detection are also possible.

It is the responsibility of the designs to ensure that the conditions and state transitions associated with the Phy initialization state machines are accurately performed and are not susceptible to false decoding/transition as a result of receiving a signal at a speed different than currently selected or at an unsupported speed.

Devices shall not rely on the host transmission of D10.2 as a means for determining host communication speed since the D10.2 transmission is done at the lowest supported communication speed and not necessarily at the highest mutually supported data speed being negotiated. The D10.2 transmission is only for the purpose of crosstalk suppression and for providing a reference clock. There is no protocol interlock on the D10.2 reception in the Device Phy Initialization state machine.

8.4.4.2 Power-on sequence timing diagram

The following timing diagrams (see Figure 260 and Figure 261) and descriptions are provided for clarity and are informative. The state machines provided according to 8.4 comprise the normative behavior specification and is the ultimate reference.

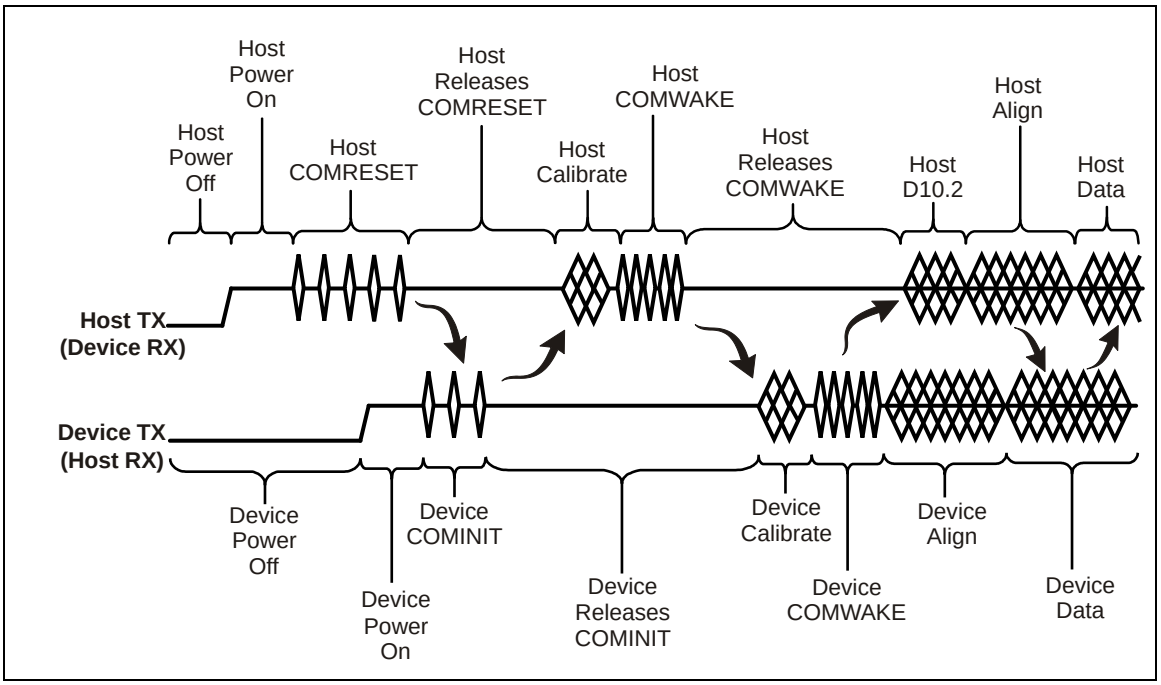


Figure 260 – Power-on sequence

Description:

1) host/device power-off, host and device power-off;
2) power is applied, host side signal conditioning pulls Tx and Rx pairs to neutral state (common mode voltage);
3) host issues COMRESET;
4) host releases COMRESET. Once the power-on reset is released, the host releases the COMRESET signal and puts the bus in a quiescent condition;
5) device issues COMINIT. If the device detects the release of COMRESET, it responds with a COMINIT. This is also the entry point if the device is late starting. The device may initiate communications at any time by issuing a COMINIT;
6) host calibrates and issues a COMWAKE;
7) device responds. The device detects the COMWAKE sequence on its Rx pair and calibrates its transmitter (optional). Following calibration the device sends a six burst COMWAKE sequence and then sends a continuous stream of the $ALIGN_P$ sequence starting at the device's highest supported speed. After $ALIGN_P$ primitives have been sent for 54.6 us (i.e., 2 048 nominal Gen1 Dword times) without a response from the host as determined by detection of $ALIGN_P$ primitives received from the host, the device assumes that the host is unable to communicate at that speed. If additional speeds are available the device tries the next lower supported speed by sending $ALIGN_P$ primitives at that speed for 54.6 us (i.e., 2 048 nominal Gen1 Dword times). This step is repeated for as many slower speeds as are supported. Once the lowest speed has been reached without response from the host, the device shall enter an error state;

8) host locks, after detecting the COMWAKE, the host starts transmitting D10.2 characters (see 7.8) at its lowest supported speed. Meanwhile, the host receiver locks to the ALIGNP sequence and, if ready, returns the ALIGNP sequence to the device at the same speed as received. A host shall be designed such that it acquires lock in 54.6 us (i.e., 2 048 nominal Gen1 Dword times) at any given speed. The host should allow for at least 873.8 us (i.e., 32 768 nominal Gen1 Dword times) after detecting the release of COMWAKE to receive the first ALIGNP. This insures interoperability with multi-generational and synchronous designs. If no ALIGNP is received within 873.8 us (i.e., 32 768 nominal Gen1 Dword times) the host restarts the power-on sequence, repeating indefinitely until told to stop by the Application layer;
9) device locks, the device locks to the ALIGNP sequence and, if ready, sends the SYNCP primitive indicating it is ready to start normal operation; and
10) upon receipt of three back-to-back non-ALIGNP primitives, the communication link is established and normal operation may begin.

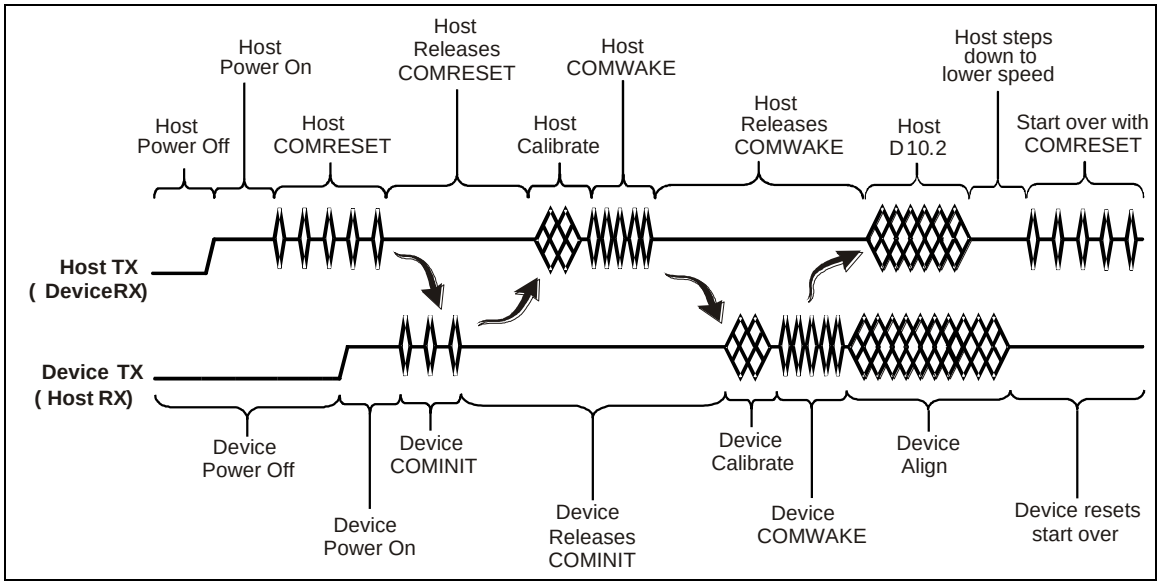


Figure 261 – Speed negotiation using COMRESET (RSN)

Description:

1) host/device are powered and operating normally with some form of active communication;
2) some condition in the host causes the host to issue COMRESET;
3) host releases COMRESET. Once the condition causing the COMRESET is released, the host releases the COMRESET signal and puts the bus in a quiescent condition;
4) device issues COMINIT. If the device detects the release of COMRESET, it responds with a COMINIT. This is also the entry point if the device is late starting. The device may initiate communications at any time by issuing a COMINIT;
5) host calibrates and issues a COMWAKE;
6) device responds. The device detects the COMWAKE sequence on its Rx pair and calibrates its transmitter (optional). Following calibration the device sends a six burst COMWAKE sequence and then sends a continuous stream of the ALIGNP sequence starting at the device's highest supported speed. After ALIGNP Dwords have been sent for 54.6 us (i.e., 2 048 nominal Gen1 Dword times) without a response from the host as determined by detection of ALIGNP primitives received from the host, the device assumes that the host is unable to communicate at that speed. If additional speeds are available the device tries the next lower supported speed by sending ALIGNP Dwords at that speed for 54.6 us (i.e., 2 048 nominal Gen1 Dword times). This step is repeated for as many slower speeds as are supported. Once the lowest speed has been reached without response from the host, the device enters an error state;

7) host locks, after detecting the COMWAKE, the host starts transmitting D10.2 characters (see 7.8) at its lowest supported speed. Meanwhile, the host receiver locks to the $ALIGN_P$ sequence and, if ready, returns the $ALIGN_P$ sequence to the device at the same speed as received. A host shall be designed such that it acquires lock in 54.6 us (i.e., 2 048 nominal Gen1 Dword times) at any given speed. The host should allow for at least 873.8 us (i.e., 32 768 nominal Gen1 Dword times) after detecting the release of COMWAKE to receive the first $ALIGN_P$. This ensures interoperability with multi-generational and synchronous designs. If no $ALIGN_P$ is received within 873.8 us (i.e., 32 768 nominal Gen1 Dword times) the host restarts the power-on sequence, repeating indefinitely until told to stop by the Application layer. A host, internally operating at a higher speed than the device, that has the capability of detecting the incoming stream of lower speed $ALIGN_P$ Dwords may issue a COMRESET on detection and rate verification of the lower speed $ALIGN_P$ Dwords. This host initiated COMRESET sequence allows the host to adjust its speed to match the lower speed device and resume the speed negotiation sequence as described above at the matched rate, thereby reducing the time for the speed negotiation process. This is illustrated in Figure 262;
8) device locks. The device locks to the $ALIGN_P$ sequence and, if ready, sends $SYNC_P$ indicating it is ready to start normal operation; and
9) upon receipt of three back-to-back non-$ALIGN_P$ primitives, the communication link is established and normal operation may begin.

8.4.4.3 Partial/Slumber to PHYRDY

8.4.4.3.1 Host initiated

The host may initiate a wakeup from the Partial or Slumber states by entering the power-on sequence at the “Host COMWAKE” point in the state machine. Calibration and speed negotiation is bypassed since it has already been performed at power-on and system performance depends on quick resume latency. The device, therefore, shall transmit $ALIGN_P$ primitives at the speed determined at power-on.

8.4.4.3.2 Device initiated

The device may initiate a wakeup from the Partial or Slumber states by entering the power-on sequence at the “Device COMWAKE” point in the state machine. Calibration and speed negotiation is bypassed since it has already been performed at power-on and system performance depends on quick resume latency. The device, therefore, shall transmit $ALIGN_P$ primitives at the speed determined at power-on.

8.4.4.4 PHYRDY to Partial/Slumber

8.4.4.4.1 Host initiated

Figure 262 shows the host initiated PHYRDY to Partial transition.

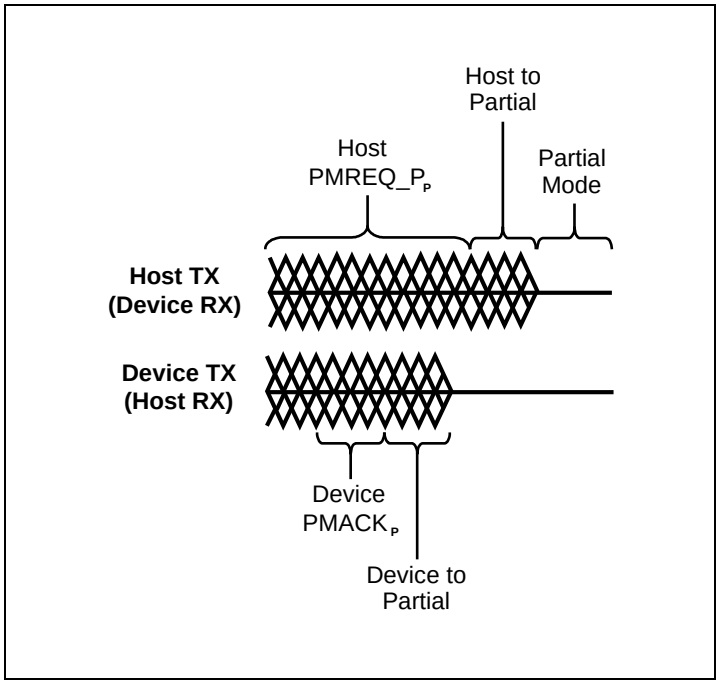


Figure 262 – PHYRDY to Partial – host initiated

NOTE 37 - For Slumber, the same sequence applies except PMREQ_P is replaced with PMREQ_S and Partial is replaced with Slumber.

Detailed Sequence:

1) host Application layer sends request to host Transport layer;
2) host Transport layer transmits request to host Link layer;
3) host Link layer encodes request as PMREQ_P primitive and transmits it to the host Phy layer;
4) host Phy layer serializes PMREQ_P primitives and transmits them to device Phy layer;
5) device Phy de-serializes PMREQ_P primitives and transmits them to device Link layer;
6) device Link layer decodes PMREQ_P primitives and transmits request to device Transport layer;
7) device Transport layer transmits request to device Application layer;
8) device Application layer processes and accepts request. Issues accept to device Transport layer;
9) device Transport layer transmits acceptance to device Link layer;
10) device Link layer encodes acceptance as PMACK primitive and transmits it four times to device Phy layer;
11) device Phy layer transmits between four and sixteen PMACK primitives to host Phy layer;
12) device Link layer places device Phy layer in Partial state;
13) host Phy layer de-serializes PMACK primitives and transmits them to host Link layer;
14) host Link layer decodes PMACK primitives and transmits acceptance to host Transport layer;
15) host Link layer places host Phy layer in Partial State; and
16) host Transport layer transmits acceptance to host Application layer.

8.4.4.4.2 Device initiated

Figure 263 shows device initiated PHYRDY to Partial transition.

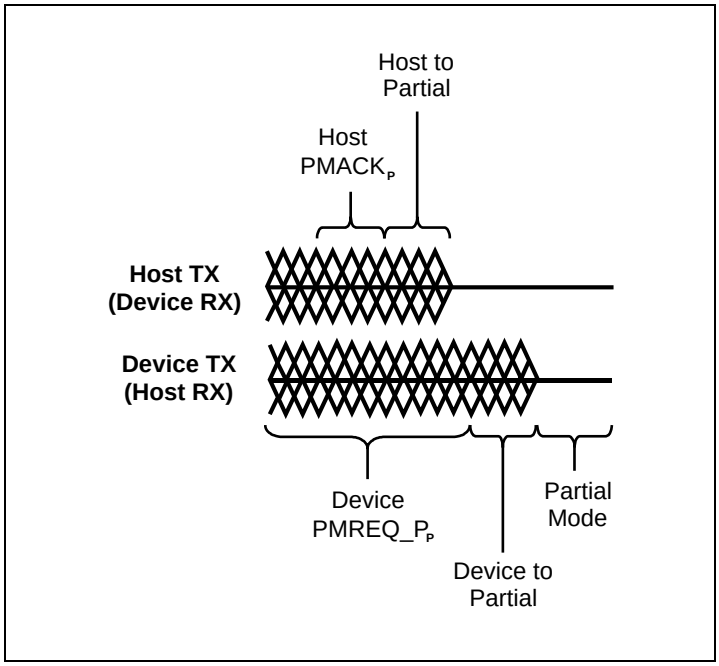


Figure 263 – PHYRDY to Partial – device initiated

Detailed Sequence:

1) device Application layer sends request to device Transport layer;
2) device Transport layer transmits request to device Link layer;
3) device Link layer encodes request as PMREQ_P$_P$ primitive and transmits it to device Phy layer;
4) device Phy layer serializes PMREQ_P$_P$ primitives and transmits them to host Phy layer;
5) host Phy de-serializes PMREQ_P$_P$ primitives and transmits them to host Link layer;
6) host Link layer decodes PMREQ_P$_P$ primitives and transmits request to host Transport layer;
7) host Transport layer transmits request to host Application layer;

   NOTE 38 - In this context, the host Application layer does not necessarily imply BIOS or other host CPU programming. Rather, the Application layer is the intelligent control section of the chipset logic.

8) host Application layer processes and accepts request. Issues accept to host Transport layer;
9) host Transport layer transmits acceptance to host Link layer;
10) host Link layer encodes acceptance as PMACK$_P$ and transmits it four times to host Phy layer;
11) host Phy layer transmits between four and sixteen PMACK$_P$ primitives to device Phy layer;
12) host Link layer asserts Partial signal and places host Phy layer in Partial state;
13) host Phy layer negates PHYRDY signal;
14) device Phy layer de-serializes PMACK$_P$ primitives and transmits them to device Link layer;
15) device Link layer decodes PMACK$_P$ primitives and transmits acceptance to device Transport layer;
16) device Link layer asserts Partial signal and places device Phy layer in Partial State;
17) device Phy layer negates PHYRDY signal; and
18) device Transport layer transmits acceptance to device Application layer.

8.5 DEVSLP signal protocol and timing

8.5.1 DEVSLP overview

Figure 264 and Table 79 provide an overview of the DEVSLP protocol.

The DEVSLP signal is a host-controlled signal which tells the device to enter the DevSleep interface power state (see 8.1). Together, the DEVSLP signal and DevSleep interface power state enable a SATA host and device to enter an ultra-low interface power state, including the possibility of completely powering down host and device Phys.

Support for the DEVSLP signal is advertised by the device in IDENTIFY DEVICE, Word 78, bit 8 (see 13.2.2) which defines the Device Sleep feature. If supported, the Device Sleep feature is enabled/disabled by the SET FEATURES command (see 13.6.6.8).

The following sections describe the DEVSLP signal protocol, as well as the electrical characteristics for the DEVSLP signal.

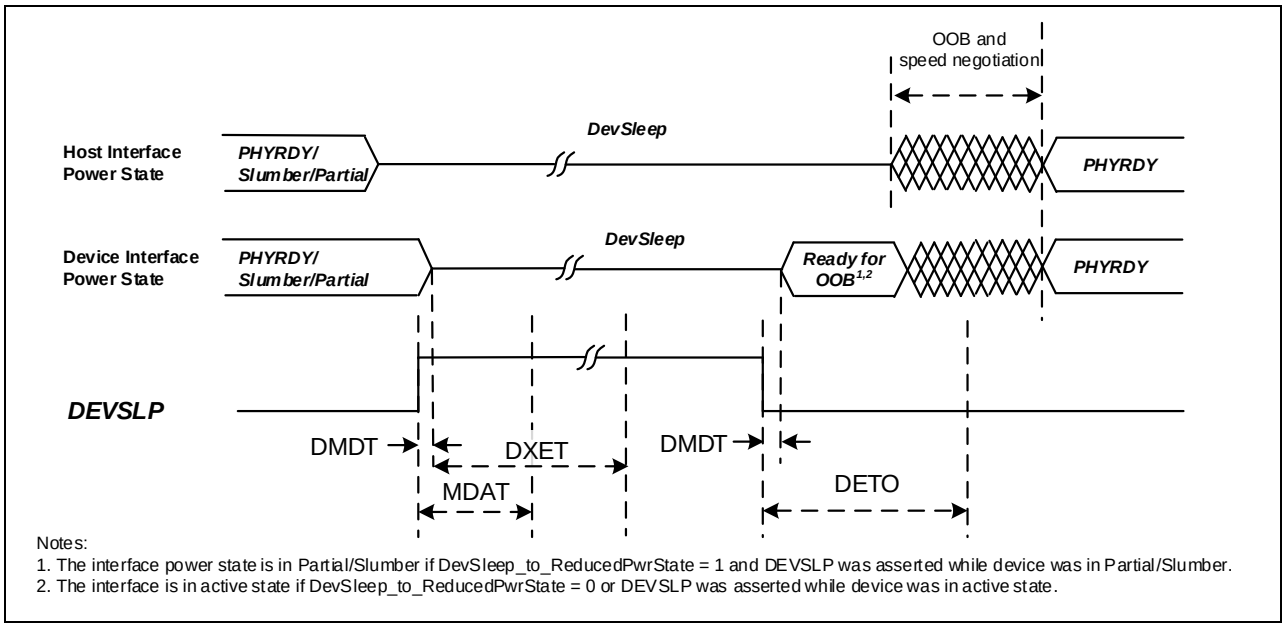


Figure 264 – DEVSLP protocol overview

Table 79 – DevSleep timing parameters

Symbol	Parameter	Value
MDAT	**Minimum DEVSLP Assertion Time,** minimum time that the host shall assert DEVSLP, once it has been asserted.	10 ms unless otherwise specified in Identify Device Data log.
DMDT	**DEVSLP Minimum Detection Time**, minimum time the device needs to de-bounce the DEVSLP signal after detecting it has been asserted.	10 us
DXET	**DevSleep maXimum Entry Time**, maximum time from the beginning of an interval in which DEVSLP is asserted and held, to when the device shall not initiate any device to host communications and the device shall ignore any host to device communications.	60 s
DETO	**DevSleep Exit Timeout,** maximum time from when DEVSLP is negated, to when the device shall be ready to detect OOB signals.	20 ms unless otherwise specified in Identify Device Data log.

8.5.2 Host requirements for DEVSLP

The host shall assert DEVSLP to request a device to enter the DevSleep interface power state (see 8.1). The host shall negate DEVSLP to request a device to exit from the DevSleep interface power state.

DEVSLP shall be asserted for the entire duration that a host wishes the device to remain in the DevSleep interface power state.

The host should not assert DEVSLP unless all of the following conditions are true:

a) the Device Sleep feature is supported (i.e., IDENTIFY DEVICE data Word 78 bit 8 is set to one);
b) the Device Sleep feature is enabled (i.e., IDENTIFY DEVICE data Word 79 bit 8 is set to one);
c) the Shadow Register Block reflects command completion (see 4.1.1.19); and
d) the SActive register is cleared to zero.

If the host asserts DEVSLP:

a) the host may power down its Phy or any other subsystems (e.g., PLLs, clocks, etc);
b) the host shall not initiate any host to device communications; and
c) the host shall ignore any device to host communications.

Once asserted, the host shall keep DEVSLP asserted for a minimum of 10 ms unless otherwise specified in the MDAT field (bits 4:0) of the DEVSLP Timing Variables Qword in Identify Device Data log (see 13.7.11).

After negating DEVSLP, the host shall keep it negated until the host Phy has reached the PHYRDY interface power state (see 8.1).

8.5.3 Device requirements for DEVSLP

If the Device Sleep feature is supported (i.e., IDENTIFY DEVICE data Word 78 bit 8 is set to one), then the requirements specified in this sub-clause shall apply.

Since the Power Disable feature (see 8.6) and the Device Sleep feature both use pin P3, these features are mutually exclusive.

After power up, the device shall ignore DEVSLP until the Device Sleep feature is enabled by a SET FEATURES command from the host (see 13.3).

If:

a) the Device Sleep feature is enabled (i.e., IDENTIFY DEVICE data Word 79 bit 8 is set to one);
b) the device detects that DEVSLP has been asserted for greater than or equal to DMDT (see Table 79);
c) no commands are outstanding; and
d) the device is in the DI0: Device_Idle state,

then:

a) the device shall enter the DevSleep interface power state;
b) the device may power down its Phy and any other subsystems (e.g., PLLs, clocks, media);
c) the device shall not initiate any device to host communications; and
d) the device shall ignore any host to device communications,

otherwise the device shall not enter the DevSleep interface power state, and shall not return an error to the host.

If the device detects that DEVSLP has been negated for greater than or equal to DMDT (see Table 79), then:

a) the device shall be ready to detect OOB signals in less than or equal to 20 ms unless otherwise specified in the DETO field (bits 12:5) of the DEVSLP Timing Variables Qword in Identify Device Data log (see 13.7.11);
b) if DevSleep_to_ReducedPwrState is not supported (see 13.2.2.18), then the device shall resume operation in the DP1: DR_Reset state; and
c) if DevSleep_to_ReducedPwrState is supported (see 13.2.2.18), then
   A) the device shall transition to the DP8: DR_Partial state if the DEVSLP signal was asserted while the device was in the DP8: DR_Partial state;
   B) the device shall transition to the DP9: DR_Slumber state if the DEVSLP signal was asserted while the device was in the DP9: DR_Slumber state; or
   C) the device shall transition to the DP1: DR_Reset state.

8.5.4 DEVSLP signal electrical characteristics

The DEVSLP signal shall be implemented with the electrical constraints in Table 80 and Table 81 DEVSLP is a level triggered signal, asserted high.

The device shall tolerate the DEVSLP signal being shorted to ground. The device shall tolerate a no connect floating DEVSLP signal.

Figure 265 is an example of a DEVSLP implementation for illustrative purposes.

NOTE 39 - The host is unable to rely on particular device resistor values.

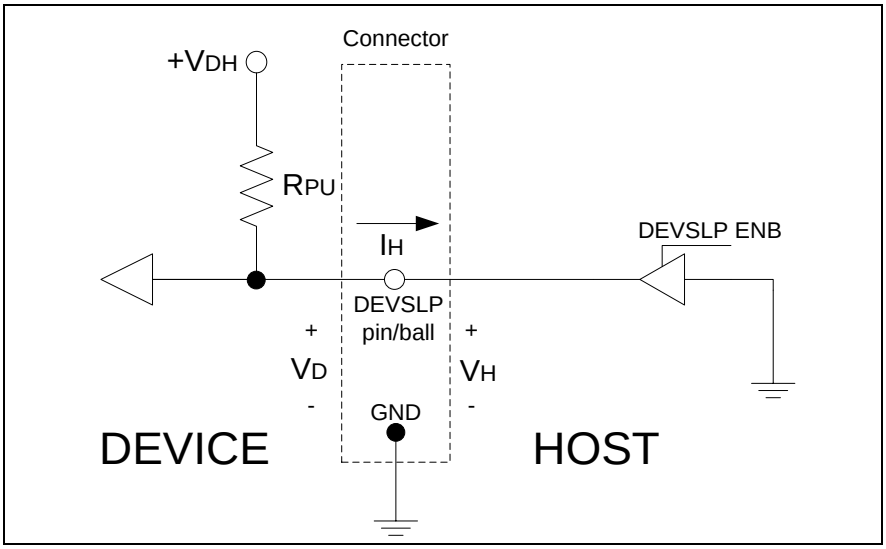


Figure 265 – Example DEVSLP electrical block diagram

All voltage references in Table 80 and Table 81 are to ground pin on the host connector. All voltages and currents in Table 80 and Table 81 are measured at DEVSLP pin on the host connector.

Table 80 – Device side DEVSLP electrical parameters

Parameter	Description and Conditions	SATA, mSATA		M.2, MicroSSD	
		Min	Max	Min	Max
V_{DIn}	Tolerated input voltage.	-0.5 V	3.6 V	-0.5 V	3.6 V
$V_{HAssert}$	Voltage presented to host if signal not driven low. Value specified for all allowable $I_{HAssert}$ (see Table 81).	-	2.1 V	-	1.89 V
$I_{HNegate}$	Device current delivered to host if host driving signal low. Value specified at $V_{HNegate}$ voltage of 0 V (see Table 81).	-	100 uA	-	100 uA

Table 81 – Host side DEVSLP electrical parameters

Parameter	Description and Conditions	SATA, mSATA		M.2, MicroSSD	
		Min	Max	Min	Max
V_{HIn}	Tolerated input voltage.	-0.5 V	2.1 V	-0.5 V	1.89 V
$I_{HAssert}$	Host leakage current if signal not driven. Value specified for all voltages between 0 V and $V_{HAssert}$ (see Table 80) of V_{HIn} maximum value.	-1 uA	10 uA	-1 uA	10 uA
$V_{HNegate}$	Host voltage presented to device if the signal driven low. Value specified for all allowable $I_{HNegate}$ (see Table 80).	0 V	0.225 V	0 V	0.225 V

8.6 Power Disable signal protocol and timing

The Power Disable feature, if supported and enabled, may be used to disable power to the device circuitry.

Since the Power Disable feature and the Device Sleep feature (see 8.5) both use pin P3, these features are mutually exclusive.

If the Power Disable feature is supported, then the device shall set IDENTIFY DEVICE data Word 78 bit 12 to one.

If the Power Disable feature is supported and the POWER DISABLE FEATURE ALWAYS ENABLED bit (see 13.7.11.2.16):

a) is set to one, then the Power Disable feature is always enabled; or
b) is cleared to zero, then the Power Disable feature:
   A. shall be disabled as a result of processing a power on reset;
   B. shall not be affected as a result of processing a hardware reset or a software reset; and
   C. may be enabled as a result of processing a SET FEATURES Enable/Disable Power Disable Feature subcommand (see 13.3.12).

If the Power Disable feature is supported and enabled, then the device shall:

a) allow power to be applied to the device circuitry if the PWDIS signal is not connected on the host connector;
b) allow power to be applied to the device circuitry if the PWDIS signal is negated as defined in Table 82;
c) disable power applied to the device circuitry if:
   1) the minimum negated hold time in Table 82 is met; and
   2) the PWDIS signal is asserted as defined in Table 82;
d) perform the actions defined for a power on reset if:
   1) the minimum negated hold time in Table 82 is met;
   2) the PWDIS signal is asserted as defined in Table 82; and
   3) the PWDIS signal transitions from asserted to negated;
   and
e) not respond to a change of the PWDIS signal from negated to asserted or asserted to negated until the PWDIS signal is held at the asserted or negated level for a minimum of 1 us as defined by T_{DS} in Table 82.

Figure 266 provides an overview of the Power Disable protocol. See 6.2.3.2 for additional requirements.

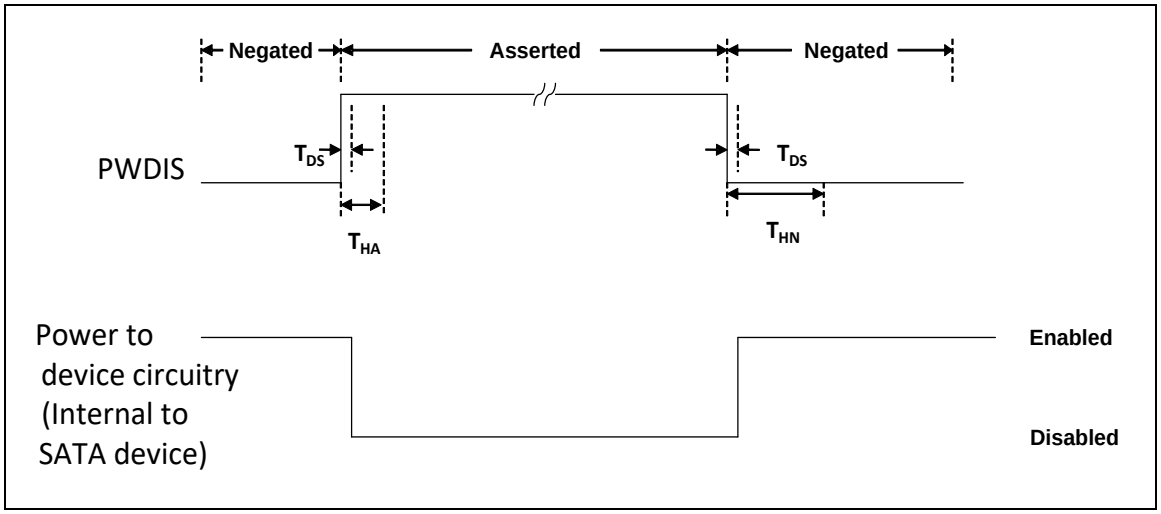


Figure 266 – Power Disable protocol overview

All voltages and currents in Table 82 are measured at the PWDIS pin (i.e., P3) on the device connector.

Table 82 – Characteristics of the PWDIS signal applied to the device

Parameter	Characteristic	Units	Minimum	Maximum
V_{DIn}	Absolute maximum voltage input range	V	-0.5	3.6
$V_{HNegate}$	Negated voltage (power enabled) [a] [b]	V	-0.5	0.7
$V_{HAssert}$	Asserted voltage (power disabled) [c]	V	2.1	3.6
A_{DSSCC}	Driver sink/source current capability [b] [c]	uA	100	-
T_{DS}	Device response time to a change in the PWDIS signal.	us	1.0	-
T_{HA}	PWDIS asserted hold time [d] [e]	s	5.0	-
T_{HN}	PWDIS negated hold time [d] [e]	s	30.0	-

[a] The device shall allow power to be applied to the device circuitry if P3 is not connected on the host connector.
[b] The PWDIS signal shall be actively negated.
[c] The PWDIS signal shall be actively asserted.
[d] The hold time is the length of time the PWDIS signal is asserted or negated. The length of time after the PWDIS signal is asserted or negated until the disabling or allowing of power to the device circuitry is vendor specific.
[e] The PWDIS signal should not transition from negated to asserted or asserted to negated for the negated hold time:
a) after power is applied to the host connector; or
b) after the detection of a hot plug event.

9 Link layer

9.1 Link layer overview

The Link layer transmits and receives frames, transmits primitives based on control signals from the Transport layer, and receives primitives from the Phy layer that are converted to control signals to the Transport layer. The Link layer need not be cognizant of the content of frames. Host and device Link layer state machines are similar, however the device is given precedence if both the host and device request ownership for transmission.

9.2 Link layer frame transmission and reception

9.2.1 Frame transmission

If requested by the Transport layer to transmit a frame, the Link layer provides the following services:

a) negotiates with its peer Link layer to transmit a frame, resolves arbitration conflicts if both host and device request transmission;
b) inserts frame envelope around Transport layer data (i.e., SOF_P, CRC, EOF_P, etc.);
c) receives data in the form of Dwords from the Transport layer;
d) calculates CRC on Transport layer data;
e) transmits frame;
f) provides frame flow control in response to requests from the first in first out (FIFO) or the peer Link layer;
g) receives frame receipt acknowledge from peer Link layer;
h) reports good transmission or Link/Phy layer errors to Transport layer;
i) performs 8b/10b encoding; and
j) scrambles data Dwords in such a way to distribute the potential EMI emissions over a broader range.

9.2.2 Frame reception

When data is received from the Phy layer, the Link layer provides the following services:

a) acknowledges to the peer Link layer readiness to receive a frame;
b) receives data in the form of encoded characters from the Phy layer;
c) decodes the encoded 8b/10b character stream into aligned Dwords of data;
d) removes the envelope around frames (i.e., SOF_P, CRC, EOF_P);
e) calculates CRC on the received Dwords;
f) provides frame flow control in response to requests from the FIFO or the peer Link layer;
g) compares the calculated CRC to the received CRC;
h) reports good reception or Link/Phy layer errors to Transport layer and the peer Link layer; and
i) descrambles data Dwords received from a peer Link layer.

9.3 Encoding method

9.3.1 Encoding method overveiw

Information to be transmitted over Serial ATA shall be encoded a byte (eight bits) at a time along with a data or control character indicator into a 10 bit encoded character and then sent serially bit by bit. Information received over Serial ATA shall be collected ten bits at a time, assembled into an encoded character, and decoded into the correct data characters and control characters. The 8b/10b code allows for the encoding of all 256 combinations of eight bit data. A subset of the control character set is utilized by Serial ATA.

9.3.2 Notation and conventions

Serial ATA uses a letter notation for describing data bits and control variables. A description of the translation process between these notations follows. This section also describes a convention used to differentiate data characters from control characters. Finally, translation examples for both a data character and a control character are presented.

An unencoded byte of data is composed of eight bits A,B,C,D,E,F,G,H and the control variable Z. The encoding process results in a 10 bit character a,b,c,d,e,i,f,g,h,j. A bit is either a binary zero or binary one. The control variable, Z, has a value of D or K. If the control variable associated with a byte has the value D, the byte is referred to as a data character. If the control variable associated with a byte has the value K, the byte is referred to as a control character.

If a data byte is not accompanied with a specific control variable value the control variable Z is assumed to be Z = D and the data byte shall be encoded as a data character.

Table 83 illustrates the association between the numbered unencoded bits in a byte, the control variable, and the letter-labeled bits in the encoding scheme.

Table 83 – Bit designations

Data Byte Notation	7	6	5	4	3	2	1	0	Control Variable
Unencoded bit notation	H	G	F	E	D	C	B	A	Z

Each character is given a name Zxx.y where Z is the value of the control variable (D for a data character, K for a control character), xx is the decimal value of the binary number composed of the bits E, D, C, B, and A in that order, and y is the decimal value of the binary number composed of the bits H, G, and F.

Figure 267, shows the relationship between the various representations.

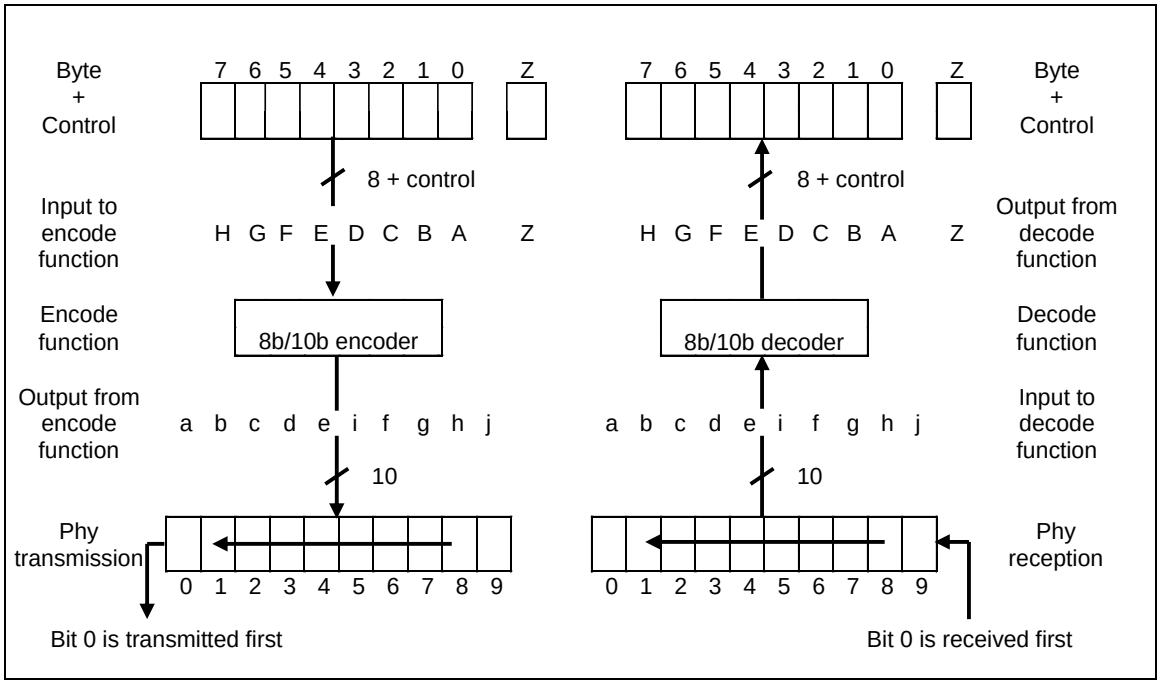


Figure 267 – Nomenclature reference

Table 84 shows conversions from byte notation to character notation for a control and data byte. The examples chosen have special significance and are also used during the conversion from data notation to the 8b/10b code values.

Table 84 – Conversion examples

Byte Notation	BCh, Control Character	4Ah, Data Character
Bit notation	7654 3210 Control variable 1011 1100b K	7654 3210 Control variable 0100 1010b D
Unencoded bit notation	HGF EDCBA Z 101 11100b K	HGF EDCBA Z 010 01010b D
Bit notation reordered to conform with Zxx.y convention	Z EDCBA HGF K 11100 101b	Z EDCBA HGF D 01010 010b
Character name	K 28 .5	D 10 .2

9.3.3 Character code

9.3.3.1 Character code overview

The coding scheme used by Serial ATA translates unencoded data and control bytes to characters. The encoded characters are then transmitted by the Phy layer over the serial line where they are received from the Phy layer and decoded into the corresponding byte and control value.

Serial ATA uses a subset of the 8b/10b coding method described by Widmer and Franaszek (see 3.4). The Serial ATA code uses all 256 data byte encodings while only two of the control codes are used. The reception of any unused code is a class of reception error referred to as a code violation.

9.3.3.2 Code construction

The 8b/10b encoding process is defined in two stages. The first stage encodes the first five bits of the unencoded input byte into a six bit sub-block using a 5b/6b encoder. The input to this stage includes the current running disparity value. The second stage uses a 3b/4b encoder to encode the remaining three bits of the data byte and the running disparity as modified by the 5b/6b encoder into a four bit value.

In the derivations that follow, the control variable (Z) is assumed to have a value of D, and thus is an implicit input.

9.3.3.3 The concept of running disparity

Running Disparity is a binary parameter with either the value negative (i.e., -) or the value positive (i.e., +).

After transmitting any encoded character, the transmitter shall calculate a new value for its Running Disparity based on the value of the transmitted character.

After a COMRESET, initial power-up, exiting any power management state, or exiting any diagnostic mode, the receiver shall assume either the positive or negative value for its initial Running Disparity. Upon reception of an encoded character the receiver shall determine whether the encoded character is valid according to the following rules and tables and shall calculate a new value for its Running Disparity based on the contents of the received character.

The following rules shall be used to calculate a new Running Disparity value for the transmitter after it sends an encoded character (transmitter's new Running Disparity) and for the receiver upon reception of an encoded character (receiver's new Running Disparity).

Running Disparity for an encoded character shall be calculated on two sub-blocks where the first six bits (abcdei) form one sub-block (e.g., the six bit sub-block). The last four bits (fghj) form the second sub-block – the four bit sub-block. Running Disparity at the beginning of the six bit sub-block is the Running Disparity at the end of the last encoded character or the initial conditions described above for the first encoded character transmitted or received. Running Disparity at the beginning of the four bit sub-block is the resulting Running Disparity from the six bit sub-block. Running Disparity at the end of the encoded character – and the initial Running Disparity for the next encoded character – is the Running Disparity at the end of the four bit sub-block.

Running Disparity for each of the sub-blocks shall be calculated as follows:

a) Running Disparity at the end of any sub-block is positive if the sub-block contains more ones than zeros. It is also positive at the end of the six bit sub-block if the value of the six bit sub-block is 00 0111b, and is positive at the end of the four bit sub-block if the value of the four bit sub-block is 0011b;
b) Running Disparity at the end of any sub-block is negative if the sub-block contains more zeros than ones. It is also negative at the end of the six bit sub-block if the value of the six bit sub-block is 11 1000b, and is negative at the end of the four bit sub-block if the value of the four bit sub-block is 1100b; or
c) otherwise, for any sub-block with an equal number of zeros and ones, the Running Disparity at the end of the sub-block is the same as at the beginning of the sub-block, sub-blocks with an equal number of zeros and ones are said to have neutral disparity.

The 8b/10b code restricts the generation of the 00 0111b, 11 1000b, 0011b and 1100b sub-blocks in order to limit the run length of zeros and ones between sub-blocks. Sub-blocks containing 00 0111b or 0011b are generated only if the running disparity at the beginning of the sub-block is positive, resulting in positive Running Disparity at the end of the sub-block. Similarly, sub-blocks containing 11 1000b or 1100b are generated only if the running disparity at the beginning of the sub-block is negative and the resulting Running Disparity shall also be negative.

The rules for Running Disparity shall result in generation of a character with disparity that is either the opposite of the previous character or neutral.

Sub-blocks with non-zero (non-neutral) disparity are of alternating disparity.

9.3.3.4 Data encoding

Table 85 and Table 86 describe the code and running disparity generation rules for each of the sub-blocks. The results may be used to generate the data in the data character tables.

The digital logic that is used to generate the results may be found in Franaszek and Widmer (see 3.4). The generation of control characters is also covered in the patent but not here.

In Table 85 and Table 86 rd+ or rd- represent the current (incoming) running disparity and rd' represents the resulting Running Disparity. The resulting Running Disparity columns use –rd to indicate a change in Running Disparity polarity while rd indicates the resulting sub-block has neutral disparity.

Table 85 – 5b/6b coding

Inputs		abcdei Outputs		rd'	Inputs		abcdei Outputs		rd'
Dx	EDCBA	rd+	rd-		Dx	EDCBA	rd+	rd-	
D0	00000b	011000b	100111b	-rd	D16	10000b	100100b	011011b	-rd
D1	00001b	100010b	011101b		D17	10001b	100011b		rd
D2	00010b	010010b	101101b		D18	10010b	010011b		
D3	00011b	110001b		rd	D19	10011b	110010b		
D4	00100b	001010b	110101b	-rd	D20	10100b	001011b		
D5	00101b	101001b			D21	10101b	101010b		
D6	00110b	011001b		rd	D22	10110b	011010b		
D7	00111b	000111b	111000b		D23	10111b	000101b	111010b	-rd
D8	01000b	000110b	111001b	-rd	D24	11000b	001100b	110011b	
D9	01001b	100101b			D25	11001b	100110b		rd
D10	01010b	010101b			D26	11010b	010110b		
D11	01011b	110100b		rd	D27	11011b	001001b	110110b	-rd
D12	01100b	001101b			D28	11100b	001110b		rd
D13	01101b	101100b			D29	11101b	010001b	101110b	
D14	01110b	011100b			D30	11110b	100001b	011110b	-rd
D15	01111b	101000b	010111b	-rd	D31	11111b	010100b	101011b	

Table 86 – 3b/4b coding

Inputs		fghj Outputs		rd'
Dx.y	HGF	rd+	rd-	
Dx.0	000b	0100b	1011b	-rd
Dx.1	001b	1001b		rd
Dx.2	010b	0101b		
Dx.3	011b	0011b	1100b	
Dx.4	100b	0010b	1101b	-rd
Dx.5	101b	1010b		rd
Dx.6	110b	0110b		
Dx.P7	111b	0001b	1110b	-rd
Dx.A7 [a]	111b	1000b	0111b	

[a] A7 replaces P7 if[(rd>0) and (e=i=0)] or [(rd<0) and (e=i=1)]

9.3.3.5 Encoding examples

The encoding examples in Table 87 illustrate how the running disparity calculations are done.

EXAMPLE 1- The first conversion example completes the translation of data byte value 4Ah (i.e., the character name of D10.2) into an encoded character value of “abcdei fghj” = “010101 0101b”. This value has special significance because it is of neutral disparity, and also contains an alternating zero/one pattern that represents the highest data frequency that may be generated.

EXAMPLE 2 - In the second example the 8b/10b character named D11.7 is encoded. Assuming a positive value for the incoming Running Disparity, this example shows the Dx.P7/Dx.A7 substitution. With an initial rd+ value, D11.7 translates to an abcdei value of 110100b, with a resulting Running Disparity of positive for

the 6 bit sub-block. Encoding the 4 bit sub-block triggers the substitution clause of Dx.A7 for Dx.P7 since [(rd>0) AND (e=i=0)].

Table 87 – Encoding examples

Initial rd	Character Name	abcdei Output	6 Bit Sub-Block rd	fghj Output	4 bit Sub-block rd	Encoded Character	Ending rd
-	D10.2	010101b	-	0101b	-	010101 0101b	-
+	D11.7	110100b	+	1000b	-	110100 1000b	-

9.3.3.6 8b/10b valid encoded characters

9.3.3.6.1 8b/10b valid encoded characters overview

The following tables define the valid data characters and valid control characters. These tables shall be used for generating encoded characters (encoding) for transmission. In the reception process, the table is used to look up and verify the validity of received characters (decoding).

In Table 88, each data character and control character has two columns that represent two encoded characters. One column represents the output if the current Running Disparity is negative and the other is the output if the current Running Disparity is positive.

9.3.3.6.2 Data characters

Table 88 – Valid data characters (part 1 of 4)

Name	Byte	abcdei fghj Output		Name	Byte	abcdei fghj Output	
		Current rd-	Current rd+			Current rd-	Current rd+
D0.0	00h	100111 0100b	011000 1011b	D0.1	20h	100111 1001b	011000 1001b
D1.0	01h	011101 0100b	100010 1011b	D1.1	21h	011101 1001b	100010 1001b
D2.0	02h	101101 0100b	010010 1011b	D2.1	22h	101101 1001b	010010 1001b
D3.0	03h	110001 1011b	110001 0100b	D3.1	23h	110001 1001b	110001 1001b
D4.0	04h	110101 0100b	001010 1011b	D4.1	24h	110101 1001b	001010 1001b
D5.0	05h	101001 1011b	101001 0100b	D5.1	25h	101001 1001b	101001 1001b
D6.0	06h	011001 1011b	011001 0100b	D6.1	26h	011001 1001b	011001 1001b
D7.0	07h	111000 1011b	000111 0100b	D7.1	27h	111000 1001b	000111 1001b
D8.0	08h	111001 0100b	000110 1011b	D8.1	28h	111001 1001b	000110 1001b
D9.0	09h	100101 1011b	100101 0100b	D9.1	29h	100101 1001b	100101 1001b
D10.0	0Ah	010101 1011b	010101 0100b	D10.1	2Ah	010101 1001b	010101 1001b
D11.0	0Bh	110100 1011b	110100 0100b	D11.1	2Bh	110100 1001b	110100 1001b
D12.0	0Ch	001101 1011b	001101 0100b	D12.1	2Ch	001101 1001b	001101 1001b
D13.0	0Dh	101100 1011b	101100 0100b	D13.1	2Dh	101100 1001b	101100 1001b
D14.0	0Eh	011100 1011b	011100 0100b	D14.1	2Eh	011100 1001b	011100 1001b
D15.0	0Fh	010111 0100b	101000 1011b	D15.1	2Fh	010111 1001b	101000 1001b
D16.0	10h	011011 0100b	100100 1011b	D16.1	30h	011011 1001b	100100 1001b
D17.0	11h	100011 1011b	100011 0100b	D17.1	31h	100011 1001b	100011 1001b
D18.0	12h	010011 1011b	010011 0100b	D18.1	32h	010011 1001b	010011 1001b
D19.0	13h	110010 1011b	110010 0100b	D19.1	33h	110010 1001b	110010 1001b
D20.0	14h	001011 1011b	001011 0100b	D20.1	34h	001011 1001b	001011 1001b
D21.0	15h	101010 1011b	101010 0100b	D21.1	35h	101010 1001b	101010 1001b
D22.0	16h	011010 1011b	011010 0100b	D22.1	36h	011010 1001b	011010 1001b
D23.0	17h	111010 0100b	000101 1011b	D23.1	37h	111010 1001b	000101 1001b
D24.0	18h	110011 0100b	001100 1011b	D24.1	38h	110011 1001b	001100 1001b
D25.0	19h	100110 1011b	100110 0100b	D25.1	39h	100110 1001b	100110 1001b
D26.0	1Ah	010110 1011b	010110 0100b	D26.1	3Ah	010110 1001b	010110 1001b
D27.0	1Bh	110110 0100b	001001 1011b	D27.1	3Bh	110110 1001b	001001 1001b
D28.0	1Ch	001110 1011b	001110 0100b	D28.1	3Ch	001110 1001b	001110 1001b
D29.0	1Dh	101110 0100b	010001 1011b	D29.1	3Dh	101110 1001b	010001 1001b
D30.0	1Eh	011110 0100b	100001 1011b	D30.1	3Eh	011110 1001b	100001 1001b
D31.0	1Fh	101011 0100b	010100 1011b	D31.1	3Fh	101011 1001b	010100 1001b

Table 88 – Valid data characters (part 2 of 4)

Name	Byte	abcdei fghj Output		Name	Byte	abcdei fghj Output	
		Current rd-	Current rd+			Current rd-	Current rd+
D0.2	40h	100111 0101b	011000 0101b	D0.3	60h	100111 0011b	011000 1100b
D1.2	41h	011101 0101b	100010 0101b	D1.3	61h	011101 0011b	100010 1100b
D2.2	42h	101101 0101b	010010 0101b	D2.3	62h	101101 0011b	010010 1100b
D3.2	43h	110001 0101b	110001 0101b	D3.3	63h	110001 1100b	110001 0011b
D4.2	44h	110101 0101b	001010 0101b	D4.3	64h	110101 0011b	001010 1100b
D5.2	45h	101001 0101b	101001 0101b	D5.3	65h	101001 1100b	101001 0011b
D6.2	46h	011001 0101b	011001 0101b	D6.3	66h	011001 1100b	011001 0011b
D7.2	47h	111000 0101b	000111 0101b	D7.3	67h	111000 1100b	000111 0011b
D8.2	48h	111001 0101b	000110 0101b	D8.3	68h	111001 0011b	000110 1100b
D9.2	49h	100101 0101b	100101 0101b	D9.3	69h	100101 1100b	100101 0011b
D10.2	4Ah	010101 0101b	010101 0101b	D10.3	6Ah	010101 1100b	010101 0011b
D11.2	4Bh	110100 0101b	110100 0101b	D11.3	6Bh	110100 1100b	110100 0011b
D12.2	4Ch	001101 0101b	001101 0101b	D12.3	6Ch	001101 1100b	001101 0011b
D13.2	4Dh	101100 0101b	101100 0101b	D13.3	6Dh	101100 1100b	101100 0011b
D14.2	4Eh	011100 0101b	011100 0101b	D14.3	6Eh	011100 1100b	011100 0011b
D15.2	4Fh	010111 0101b	101000 0101b	D15.3	6Fh	010111 0011b	101000 1100b
D16.2	50h	011011 0101b	100100 0101b	D16.3	70h	011011 0011b	100100 1100b
D17.2	51h	100011 0101b	100011 0101b	D17.3	71h	100011 1100b	100011 0011b
D18.2	52h	010011 0101b	010011 0101b	D18.3	72h	010011 1100b	010011 0011b
D19.2	53h	110010 0101b	110010 0101b	D19.3	73h	110010 1100b	110010 0011b
D20.2	54h	001011 0101b	001011 0101b	D20.3	74h	001011 1100b	001011 0011b
D21.2	55h	101010 0101b	101010 0101b	D21.3	75h	101010 1100b	101010 0011b
D22.2	56h	011010 0101b	011010 0101b	D22.3	76h	011010 1100b	011010 0011b
D23.2	57h	111010 0101b	000101 0101b	D23.3	77h	111010 0011b	000101 1100b
D24.2	58h	110011 0101b	001100 0101b	D24.3	78h	110011 0011b	001100 1100b
D25.2	59h	100110 0101b	100110 0101b	D25.3	79h	100110 1100b	100110 0011b
D26.2	5Ah	010110 0101b	010110 0101b	D26.3	7Ah	010110 1100b	010110 0011b
D27.2	5Bh	110110 0101b	001001 0101b	D27.3	7Bh	110110 0011b	001001 1100b
D28.2	5Ch	001110 0101b	001110 0101b	D28.3	7Ch	001110 1100b	001110 0011b
D29.2	5Dh	101110 0101b	010001 0101b	D29.3	7Dh	101110 0011b	010001 1100b
D30.2	5Eh	011110 0101b	100001 0101b	D30.3	7Eh	011110 0011b	100001 1100b
D31.2	5Fh	101011 0101b	010100 0101b	D31.3	7Fh	101011 0011b	010100 1100b

Table 88 – Valid data characters (part 3 of 4)

Name	Byte	abcdei fghj Output		Name	Byte	abcdei fghj Output	
		Current rd-	Current rd+			Current rd-	Current rd+
D0.4	80h	100111 0010b	011000 1101b	D0.5	A0h	100111 1010b	011000 1010b
D1.4	81h	011101 0010b	100010 1101b	D1.5	A1h	011101 1010b	100010 1010b
D2.4	82h	101101 0010b	010010 1101b	D2.5	A2h	101101 1010b	010010 1010b
D3.4	83h	110001 1101b	110001 0010b	D3.5	A3h	110001 1010b	110001 1010b
D4.4	84h	110101 0010b	001010 1101b	D4.5	A4h	110101 1010b	001010 1010b
D5.4	85h	101001 1101b	101001 0010b	D5.5	A5h	101001 1010b	101001 1010b
D6.4	86h	011001 1101b	011001 0010b	D6.5	A6h	011001 1010b	011001 1010b
D7.4	87h	111000 1101b	000111 0010b	D7.5	A7h	111000 1010b	000111 1010b
D8.4	88h	111001 0010b	000110 1101b	D8.5	A8h	111001 1010b	000110 1010b
D9.4	89h	100101 1101b	100101 0010b	D9.5	A9h	100101 1010b	100101 1010b
D10.4	8Ah	010101 1101b	010101 0010b	D10.5	AAh	010101 1010b	010101 1010b
D11.4	8Bh	110100 1101b	110100 0010b	D11.5	ABh	110100 1010b	110100 1010b
D12.4	8Ch	001101 1101b	001101 0010b	D12.5	ACh	001101 1010b	001101 1010b
D13.4	8Dh	101100 1101b	101100 0010b	D13.5	ADh	101100 1010b	101100 1010b
D14.4	8Eh	011100 1101b	011100 0010b	D14.5	AEh	011100 1010b	011100 1010b
D15.4	8Fh	010111 0010b	101000 1101b	D15.5	AFh	010111 1010b	101000 1010b
D16.4	90h	011011 0010b	100100 1101b	D16.5	B0h	011011 1010b	100100 1010b
D17.4	91h	100011 1101b	100011 0010b	D17.5	B1h	100011 1010b	100011 1010b
D18.4	92h	010011 1101b	010011 0010b	D18.5	B2h	010011 1010b	010011 1010b
D19.4	93h	110010 1101b	110010 0010b	D19.5	B3h	110010 1010b	110010 1010b
D20.4	94h	001011 1101b	001011 0010b	D20.5	B4h	001011 1010b	001011 1010b
D21.4	95h	101010 1101b	101010 0010b	D21.5	B5h	101010 1010b	101010 1010b
D22.4	96h	011010 1101b	011010 0010b	D22.5	B6h	011010 1010b	011010 1010b
D23.4	97h	111010 0010b	000101 1101b	D23.5	B7h	111010 1010b	000101 1010b
D24.4	98h	110011 0010b	001100 1101b	D24.5	B8h	110011 1010b	001100 1010b
D25.4	99h	100110 1101b	100110 0010b	D25.5	B9h	100110 1010b	100110 1010b
D26.4	9Ah	010110 1101b	010110 0010b	D26.5	BAh	010110 1010b	010110 1010b
D27.4	9Bh	110110 0010b	001001 1101b	D27.5	BBh	110110 1010b	001001 1010b
D28.4	9Ch	001110 1101b	001110 0010b	D28.5	BCh	001110 1010b	001110 1010b
D29.4	9Dh	101110 0010b	010001 1101b	D29.5	BDh	101110 1010b	010001 1010b
D30.4	9Eh	011110 0010b	100001 1101b	D30.5	BEh	011110 1010b	100001 1010b
D31.4	9Fh	101011 0010b	010100 1101b	D31.5	BFh	101011 1010b	010100 1010b

Table 88 – Valid data characters (part 4 of 4)

Name	Byte	abcdei fghj Output		Name	Byte	abcdei fghj Output	
		Current rd-	Current rd+			Current rd-	Current rd+
D0.6	C0h	100111 0110b	011000 0110b	D0.7	E0h	100111 0001b	011000 1110b
D1.6	C1h	011101 0110b	100010 0110b	D1.7	E1h	011101 0001b	100010 1110b
D2.6	C2h	101101 0110b	010010 0110b	D2.7	E2h	101101 0001b	010010 1110b
D3.6	C3h	110001 0110b	110001 0110b	D3.7	E3h	110001 1110b	110001 0001b
D4.6	C4h	110101 0110b	001010 0110b	D4.7	E4h	110101 0001b	001010 1110b
D5.6	C5h	101001 0110b	101001 0110b	D5.7	E5h	101001 1110b	101001 0001b
D6.6	C6h	011001 0110b	011001 0110b	D6.7	E6h	011001 1110b	011001 0001b
D7.6	C7h	111000 0110b	000111 0110b	D7.7	E7h	111000 1110b	000111 0001b
D8.6	C8h	111001 0110b	000110 0110b	D8.7	E8h	111001 0001b	000110 1110b
D9.6	C9h	100101 0110b	100101 0110b	D9.7	E9h	100101 1110b	100101 0001b
D10.6	CAh	010101 0110b	010101 0110b	D10.7	EAh	010101 1110b	010101 0001b
D11.6	CBh	110100 0110b	110100 0110b	D11.7	EBh	110100 1110b	110100 1000b
D12.6	CCh	001101 0110b	001101 0110b	D12.7	ECh	001101 1110b	001101 0001b
D13.6	CDh	101100 0110b	101100 0110b	D13.7	EDh	101100 1110b	101100 1000b
D14.6	CEh	011100 0110b	011100 0110b	D14.7	EEh	011100 1110b	011100 1000b
D15.6	CFh	010111 0110b	101000 0110b	D15.7	EFh	010111 0001b	101000 1110b
D16.6	D0h	011011 0110b	100100 0110b	D16.7	F0h	011011 0001b	100100 1110b
D17.6	D1h	100011 0110b	100011 0110b	D17.7	F1h	100011 0111b	100011 0001b
D18.6	D2h	010011 0110b	010011 0110b	D18.7	F2h	010011 0111b	010011 0001b
D19.6	D3h	110010 0110b	110010 0110b	D19.7	F3h	110010 1110b	110010 0001b
D20.6	D4h	001011 0110b	001011 0110b	D20.7	F4h	001011 0111b	001011 0001b
D21.6	D5h	101010 0110b	101010 0110b	D21.7	F5h	101010 1110b	101010 0001b
D22.6	D6h	011010 0110b	011010 0110b	D22.7	F6h	011010 1110b	011010 0001b
D23.6	D7h	111010 0110b	000101 0110b	D23.7	F7h	111010 0001b	000101 1110b
D24.6	D8h	110011 0110b	001100 0110b	D24.7	F8h	110011 0001b	001100 1110b
D25.6	D9h	100110 0110b	100110 0110b	D25.7	F9h	100110 1110b	100110 0001b
D26.6	DAh	010110 0110b	010110 0110b	D26.7	FAh	010110 1110b	010110 0001b
D27.6	DBh	110110 0110b	001001 0110b	D27.7	FBh	110110 0001b	001001 1110b
D28.6	DCh	001110 0110b	001110 0110b	D28.7	FCh	001110 1110b	001110 0001b
D29.6	DDh	101110 0110b	010001 0110b	D29.7	FDh	101110 0001b	010001 1110b
D30.6	DEh	011110 0110b	100001 0110b	D30.7	FEh	011110 0001b	100001 1110b
D31.6	DFh	101011 0110b	010100 0110b	D31.7	FFh	101011 0001b	010100 1110b

9.3.3.6.3 Control characters

Table 89 – Valid control characters

Name	Byte	abcdei fghj Output		Description
		Current rd-	Current rd+	
K28.3	7Ch	001111 0011b	110000 1100b	Occurs only at Byte 0 of all primitives except for ALIGN$_P$
K28.5	BCh	001111 1010b	110000 0101b	Occurs only at Byte 0 of ALIGN$_P$

In Serial ATA only the K28.3 and K28.5 control characters as defined in Table 89 are valid and are always used as the first byte in a four byte primitive. The K28.3 control character is used to prefix all primitives other than ALIGN$_P$, while the K28.5 control character is used to prefix ALIGN$_P$. The encoding of characters within primitives follow the same rules as that applied to non-primitives, while calculating the running disparity between characters and between subblocks of each character within the primitive. The control characters K28.3 and K28.5 invert the current running disparity.

ALIGN$_P$ primitives are of neutral disparity, (i.e., the running disparity at the end of ALIGN$_P$ is the same as the running disparity at the beginning of ALIGN$_P$).

9.3.4 Transmission order summary

9.3.4.1 Bits within a byte

The bits within an encoded character are labeled a,b,c,d,e,i,f,g,h,j. Bit “a” shall be transmitted first, followed in order by “b”, “c”, “d”, “e”, “i”, “f”, “g”, “h”, and “j”.

> NOTE 40 - Note that bit “i” is transmitted between bits “e” and “f”, and that bit “j” is transmitted last, and not in alphabetical order.

9.3.4.2 Bytes within a Dword

For all transmissions and receptions, Serial ATA organizes all values as Dwords. Even while representing a 32 bit value, the Dword shall be considered a set of four bytes. The transmission order of the bytes within the Dword shall be from the least-significant byte (i.e., byte 0) to the most-significant byte (i.e., byte 3). This right-to-left transmission order differs from Fibre Channel. Figure 268 illustrates how the bytes are arranged in a Dword and the order that bits are sent.

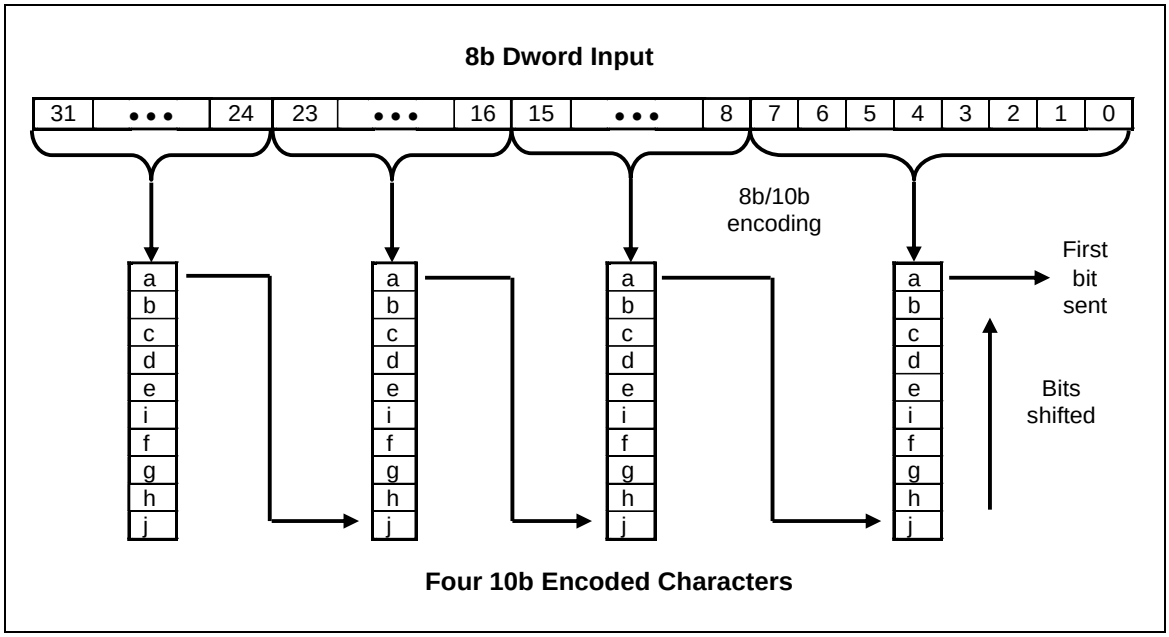


Figure 268 – Bit ordering and significance

9.3.4.3 Dwords within a frame

A frame (see 10.4.1) shall be transmitted sequentially in ascending Dword order starting with the SOF_P delimiter, followed by the Dwords of the frame contents, followed by the CRC, and ending with the EOF_P delimiter.

NOTE 41 - While this specification discusses a strict hierarchy of Dword transmission as an ordered series of bytes, it is not the intent to restrict implementations from implementing a wider data path. It is possible, and even desirable, to perform transmission in Word-sized fields. 8b/10b encoders with a 16 (i.e., unencoded) / 20 (i.e., encoded) data path do exist. The only restriction is the transmission order of each byte and running disparity for each sub-block are preserved.

9.3.5 Reception summary

9.3.5.1 Reception summary overview

Upon reception of an encoded character the column corresponding to the receiver's current Running Disparity shall be searched for the encoded character value. If the encoded character value is found in the table, then the received encoded character shall be considered a legal character and decoded, and the decoded character value is made available to the Link layer.

If the received encoded character is not found in that column, then the encoded character shall be marked as code violation and reported to the Link layer.

9.3.5.2 Disparity and the detection of a code violation

Due to the propagation characteristics of the 8b/10b code, it is possible that although most errors are detected, a single bit error may not be detected until several characters after the error occurred. The following examples illustrate this effect.

EXAMPLE 1 - The first example shows a bit error being propagated two characters before being detected (see Table 90).

EXAMPLE 2 - The second shows a single character of propagation (see Table 90).

It is important to note that Serial ATA sends data in Dword increments, but the transmitter and receiver operate in units of a byte (character). The examples do not show Dword boundaries, so it is possible that an error in either of these cases is able to be deferred one full Dword.

The frequency of disparity errors and code violations is an indicator of channel quality and corresponds directly to the bit error rate of the physical serial link between a host and device. Implementations may elect to count such events and make them available to external firmware or software.

Initial Running Disparity and the Running Disparity for each character is shown. In order to discover the errors note that Running Disparity is actually computed at the end of each sub-block and subsequently forwarded to the next sub-block. Footnotes indicate where the disparity error is detected. The error bit is underlined.

Table 90 – Single bit error with two character delay

	rd	Character	rd	Character	rd	Character	rd
Transmitted character stream.	-	D21.1	-	D10.2	-	D23.5	+
Transmitted bit stream.	-	101010 1001b	-	010101 0101b	-	111010 1010b	+
Received bit stream.	-	101010 10<u>1</u>1b [a]	+	010101 0101b	+	111010 [b] 1010b	+
Decoded character stream.	-	D21.0	+	D10.2	+	Code violation [b]	+ [c]

[a] Bit error introduced, 1001b changes to 1011b.
[b] Sub-blocks with non-neutral disparity alternate polarity (i.e., + changes to -). In this case, rd does not alternate (i.e., it stays positive for two sub-blocks in a row). The resulting encoded character does not exist in the rd+ column in the data or control code table, and so an invalid encoded character is recognized.
[c] Running disparity shall be computed on the received character regardless of the validity of the encoded character.

Table 91 – Single bit error with one character delay

	rd	Character	rd	Character	rd	Character	rd
Transmitted character stream	-	D21.1	-	D23.4	-	D23.5	+
Transmitted bit stream	-	101010 1001b	-	111010 0010b	-	111010 1010b	+
Received bit stream	-	101010 10<u>1</u>1b [a]	+	111010 [b] 0010b	-	111010 1010b	+
Decoded character stream	-	D21.0	+	Code violation [b]	-	D23.5	+ [c]

[a] Bit error introduced, 1001b changes to 1011b.
[b] Sub-blocks with non-neutral disparity alternate polarity (i.e., + changes to -). In this case, rd does not alternate (i.e., it stays positive for two sub-blocks in a row). The resulting encoded character does not exist in the rd+ column in the data or control code table, and so an invalid encoded character is recognized.
[c] Running disparity shall be computed on the received character regardless of the validity of the encoded character.

9.4 Transmission overview

The information on the serial line is a sequence of 8b/10b encoded characters. The smallest unit of communication is a Dword. The contents of each Dword are grouped to provide low-level control information or to transfer information between a host and an attached device.

The two types of structures are primitives and frames.

A primitive consists of a single Dword and is the simplest unit of information that may be exchanged between a host and a device. While the bytes of a primitive are encoded the resulting pattern is difficult to misinterpret as any other primitive or random pattern. Primitives are used primarily to convey real-time state information, to control the transfer of information, and coordinate host / device communication. All bytes in a primitive are constants and the first byte is always a special character. Since all of the bytes are constants, a primitive is unable to be used to convey variable information. Later sections describe the exact contents of the primitives used by Serial ATA.

A frame consists of multiple Dwords, and always starts with SOF_P, followed by a user payload called a Frame Information Structure (FIS), a CRC, and ends with EOF_P. The CRC is defined to be the last non-primitive Dword immediately preceding EOF_P. Some number of flow control primitives ($HOLD_P$ or $HOLDA_P$, or a $CONT_P$ stream to sustain a $HOLD_P$ or $HOLDA_P$ state) are allowed between the SOF_P and EOF_P primitives to throttle data flow for speed matching purposes.

Figure 269 shows an example of a sequence of transmissions.

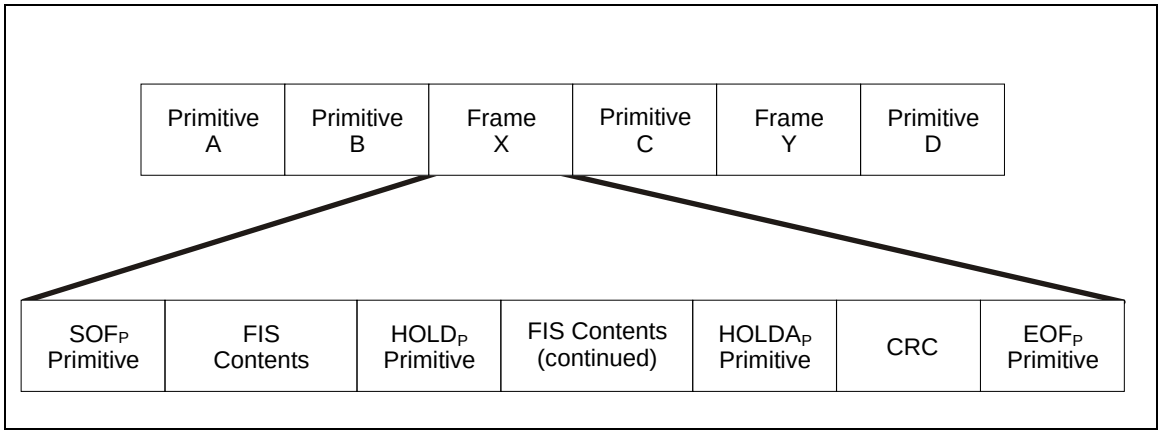


Figure 269 – Transmission structures

9.5 Primitives

9.5.1 Primitives overview

Primitives are Dword entities that are used to control and provide status of the serial line.

Primitives always begin with a control character, all primitives use the K28.3 control character to signify the beginning of a primitive except for ALIGN$_P$ that begins with the K28.5 control character. ALIGN$_P$ thus represents the only primitive that contains the comma character. Following the control character, three additional characters are encoded to complete the Dword. Table 92 is a summary of the character combinations that make up each primitive.

9.5.2 Primitive disparity

Primitives may begin with either positive or negative disparity and end in either positive or negative disparity. Normal 8b/10b encoding disparity rules are applied while encoding primitives.

ALIGN$_P$ is chosen to have neutral disparity so that it may be inserted into the stream without affecting the disparity of previously encoded characters. Disparity at the end of ALIGN$_P$ is the same as the ending disparity of the last character transmitted before ALIGN$_P$.

Each primitive is described and the encoding defined in the following sections.

9.5.3 Primitive handshakes

Some primitives are transmitted in response to receipt of other primitives to acknowledge receipt.

EXAMPLE 1 - HOLDA$_P$ is transmitted in response to the receipt of HOLD$_P$ primitives and R_OK$_P$ or R_ERR$_P$ is transmitted in response to WTRM$_P$.

Due to the different clock domains between to two ends of the cable, the number of response primitives may not match the number of primitives that they are responding.

EXAMPLE 2 - A device may send five HOLD$_P$ primitives but receive six HOLDA$_P$ primitives in response.

Neither the transmitter nor receiver of these primitives need count the number of primitives or match the number sent and received. There are boundary cases where a zero number of response primitives (e.g., HOLDA$_P$ may be sent).

9.5.4 Primitive descriptions

The following table (see Table 92) contains the primitive mnemonics and a brief description of each.

Table 92 – Description of primitives

Primitive	Name	Description
$ALIGN_P$	Phy layer control.	Upon receipt of an $ALIGN_P$, the Phy layer re-adjusts internal operations as necessary to perform its functions correctly. This primitive is always sent in pairs - there is no condition where an odd number of $ALIGN_P$ primitives shall be sent (except for retimed loopback).
$CONT_P$	Continue repeating previous primitive.	$CONT_P$ allows long strings of repeated primitives to be eliminated. $CONT_P$ implies that the previously received primitive be repeated as long as another primitive is not received.
$DMAT_P$	DMA terminate.	This primitive is sent as a request to the transmitter to terminate a DMA data transmission early by computing a CRC on the data sent and ending with EOF_P. The transmitter context is assumed to remain stable after EOF_P has been sent.
EOF_P	End of frame.	EOF_P marks the end of a frame. The previous non-primitive Dword is the CRC for the frame.
$HOLD_P$	Hold data transmission.	$HOLD_P$ is transmitted in place of payload data within a frame while the transmitter does not have the next payload data ready for transmission. $HOLD_P$ is also transmitted by the receiver while a receiver is not ready to receive additional payload data.
$HOLDA_P$	Hold acknowledge.	This primitive is sent while $HOLD_P$ is received.
$PMACK_P$	Power management acknowledge.	Sent in response to a $PMREQ\_S_P$ or $PMREQ\_P_P$ if a receiving node is prepared to enter a power mode state.
$PMNAK_P$	Power management denial.	Sent in response to a $PMREQ\_S_P$ or $PMREQ\_P_P$ if a receiving node is not prepared to enter a power mode state or if power management is not supported.
$PMREQ\_P_P$	Power management request to Partial.	This primitive is sent continuously until $PMACK_P$ or $PMNAK_P$ is received. If $PMACK_P$ is received, the current node (host or device) stops transmitting $PMREQ\_P_P$ and enters the Partial power management state.
$PMREQ\_S_P$	Power management request to Slumber	This primitive is sent continuously until $PMACK_P$ or $PMNAK_P$ is received. If $PMACK_P$ is received, the current node (host or device) stops transmitting $PMREQ\_S_P$ and enters the Slumber power management state.
R_ERR_P	Reception error.	Current node (host or device) detected error in received payload.
R_IP_P	Reception in Progress.	Current node (host or device) is receiving payload.
R_OK_P	Reception with no error.	Current node (host or device) detected no error in received payload.
R_RDY_P	Receiver ready.	Current node (host or device) is ready to receive payload.
SOF_P	Start of frame.	Start of a frame. Payload and CRC follow until EOF_P.
$SYNC_P$	Synchronization	Synchronizing primitive.
$WTRM_P$	Wait for frame termination	After transmission of an EOF_P, the transmitter sends $WTRM_P$ while waiting for reception status from receiver.
X_RDY_P	Transmission data ready.	Current node (host or device) has payload ready for transmission.

9.5.5 Primitive encoding

Table 93 defines the encoding for each primitive.

Table 93 – Primitive encoding

Primitive Name	Byte 3 Contents	Byte 2 Contents	Byte 1 Contents	Byte 0 Contents
ALIGN$_P$	D27.3	D10.2	D10.2	K28.5
CONT$_P$	D25.4	D25.4	D10.5	K28.3
DMAT$_P$	D22.1	D22.1	D21.5	K28.3
EOF$_P$	D21.6	D21.6	D21.5	K28.3
HOLD$_P$	D21.6	D21.6	D10.5	K28.3
HOLDA$_P$	D21.4	D21.4	D10.5	K28.3
PMACK$_P$	D21.4	D21.4	D21.4	K28.3
PMNAK$_P$	D21.7	D21.7	D21.4	K28.3
PMREQ_P$_P$	D23.0	D23.0	D21.5	K28.3
PMREQ_S$_P$	D21.3	D21.3	D21.4	K28.3
R_ERR$_P$	D22.2	D22.2	D21.5	K28.3
R_IP$_P$	D21.2	D21.2	D21.5	K28.3
R_OK$_P$	D21.1	D21.1	D21.5	K28.3
R_RDY$_P$	D10.2	D10.2	D21.4	K28.3
SOF$_P$	D23.1	D23.1	D21.5	K28.3
SYNC$_P$	D21.5	D21.5	D21.4	K28.3
WTRM$_P$	D24.2	D24.2	D21.5	K28.3
X_RDY$_P$	D23.2	D23.2	D21.5	K28.3

9.5.6 DMAT$_P$ primitive

No consistent use of the DMAT$_P$ facility is defined, and its use may impact software compatibility. Implementations should tolerate reception of DMAT$_P$ as defined in this specification but should avoid transmission of DMAT$_P$ in order to minimize potential interaction problems. One valid response to reception of DMAT$_P$ is to treat the DMAT$_P$ as R_IP$_P$ and complete the transfer.

EXAMPLE 1 - In the case of a direct memory access (DMA) read from device, a Serial ATA device may terminate the transfer with an EOF$_P$, and send a Register Device to Host FIS, with Error and Status registers updated appropriately.

In the case of a DMA write to device, the device sends a DMA Activate FIS to the host, and then after receiving an SOF$_P$, has to accept all data until receiving an EOF$_P$ from the host. Since the device is unable to terminate such a transfer once started, a special abort primitive is used.

The DMA Terminate (DMAT$_P$) primitive may be sent on the back channel during reception of a Data FIS to signal the transmitter to terminate the transfer in progress. It may be used for both host to device transfers and for device to host transfers. Reception of the DMAT$_P$ signal shall cause the recipient to close the current frame by inserting the CRC and EOF$_P$, and return to the idle state.

For host to device data transfers, upon receiving the DMAT$_P$ signal the host shall terminate the transfer in progress by deactivating its DMA engine and closing the frame with valid CRC and EOF$_P$. The host DMA engine shall preserve its state at the point it was deactivated so that the device may resume the transmission at a later time by transmitting another DMA Activate FIS to re-activate the DMA engine. The device is responsible for either subsequently resuming the

terminated transfer by transmitting another DMA Activate FIS or closing the affected command with appropriate status.

For device to host transfers, receipt of $DMAT_P$ signal by the device results in permanent termination of the transfer and is not resumable. The device shall terminate the transmission in progress and close the frame with a valid CRC and EOF_P, and shall thereafter clean up the affected command by indicating appropriate status for that command. No facility for resuming a device to host transfer terminated with the $DMAT_P$ signal is provided.

Some implementations may have an implementation-dependent latency associated with closing the affected Data FIS in response to the $DMAT_P$ signal.

EXAMPLE 2 - A host controller may have a small transmit FIFO, and in order for the DMA engine to accurately reflect a resumable state, the data already transferred by the DMA engine to the transmit FIFO may have to be transmitted prior to closing the affected Data FIS.

Conservative designs should minimize the $DMAT_P$ response latency while being tolerant of other devices having a long latency.

9.5.7 $CONT_P$ primitive

9.5.7.1 $CONT_P$ primitive overview

In order to accommodate EMI reductions, scrambling of data is incorporated in Serial ATA as defined in 9.6. The scrambling of data is simple, with a linear feedback shift register (LFSR) used in generating the scrambling pattern being reset at each SOF_P. However, the scrambling of primitives is not as effective or simple because of the small number of control characters available. In order to accommodate EMI reductions, repeated primitives are suppressed through the use of $CONT_P$.

Any repetitive primitive may be implied to continue repeating through the use of $CONT_P$. The recipient of $CONT_P$ shall ignore all data received after $CONT_P$ until the reception of any primitive, excluding $ALIGN_P$. Data following $CONT_P$ as defined in 9.5.7.2. The reception of $CONT_P$ shall cause the last valid primitive to be implied as repeated until the reception of the next valid primitive.

To improve overall protocol robustness and avoid potential timeout situations caused by a reception error in a primitive, all repeated primitives shall be transmitted a minimum of twice before $CONT_P$ is transmitted. The first primitive correctly received is the initiator of any action within the receiver.

EXAMPLE - This avoids scenarios where X_RDY_P is sent from the host, followed by a $CONT_P$, and the X_RDY_P is received improperly resulting in the device not returning an R_RDY_P and causing the system to deadlock until a timeout/reset condition occurs.

The transmission of $CONT_P$ is optional, but the ability to receive and properly process $CONT_P$ is required. The insertion of a single, or two repetitive primitives not followed by $CONT_P$ is valid (i.e., data, data, $HOLD_P$, data).

The following primitives may be followed by a $CONT_P$:

a) $HOLD_P$;
b) $HOLDA_P$;
c) $PMREQ\_P_P$;
d) $PMREQ\_S_P$;
e) R_ERR_P;
f) R_IP_P;
g) R_OK_P;
h) R_RDY_P;
i) $SYNC_P$;
j) $WTRM_P$; and
k) X_RDY_P.

The host Phy initialization state machine consumes the first few received primitives before communications between the host and device have been established (see state HP7:HR_SendAlign according to 8.4.2). In order to ensure proper synchronization between the host and device after entry into the L1:L_IDLE state from the LS3:L_SendAlign state or the LPM8:L_WakeUp2 state (see 9.7.2 and 9.7.5), the use of $CONT_P$ is not allowed after a transition from the LS3:L_SendAlign state or the LPM8:L_WakeUp2 state to the L1:L_IDLE state until either a minimum of 10 non-$ALIGN_P$ primitives have been transmitted or until receipt of a primitive other than $SYNC_P$ or $ALIGN_P$ has been detected.

Table 94 illustrates use of $CONT_P$ in the transmission of a FIS.

Table 94 – CONTP usage example (part 1 of 2)

Transmitter	Receiver
XXXX	XXXX
XXXX	XXXX
X_RDY_P	XXXX
X_RDY_P	XXXX
$CONT_P$	XXXX
XXXX	XXXX
XXXX	R_RDY_P
XXXX	R_RDY_P
XXXX	$CONT_P$
SOF_P	XXXX
DATA (FIS Type)	XXXX
DATA	XXXX
DATA	R_IP_P
DATA	R_IP_P
DATA	$CONT_P$
$HOLD_P$	XXXX
$HOLD_P$	XXXX
$CONT_P$	XXXX
XXXX	XXXX
XXXX	$HOLDA_P$
XXXX	$HOLDA_P$

Table 94 – CONTP usage example (part 2 of 2)

Transmitter	Receiver
$HOLD_P$	$CONT_P$
DATA	XXXX
DATA	XXXX
DATA	XXXX
CRC	XXXX
EOF_P	R_IP_P
$WTRM_P$	R_IP_P
$WTRM_P$	$CONT_P$
$WTRM_P$	XXXX
$CONT_P$	XXXX
XXXX	R_OK_P
XXXX	R_OK_P
XXXX	$CONT_P$
XXXX	XXXX
$SYNC_P$	XXXX
$SYNC_P$	XXXX
$CONT_P$	XXXX
XXXX	XXXX
XXXX	$SYNC_P$
XXXX	$SYNC_P$
XXXX	$CONT_P$
XXXX	XXXX
XXXX	XXXX

Key:
XXXX = Scrambled data values (non-primitives)
DATA = FIS payload data
Subscript p = primitive

9.5.7.2 Scrambling of data following the $CONT_P$ primitive

The data following the $CONT_P$ shall be the output of an LFSR that implements the same polynomial as is used to scramble FIS contents. That polynomial as defined in 9.6.

The resulting LFSR value shall be encoded using the 8b/10b rules for encoding data characters before transmission by the Link layer.

The LFSR used to supply data after $CONT_P$ shall be reset to the initial value upon detection of a COMINIT or COMRESET event.

Since the data following $CONT_P$ is discarded by the Link layer, the value of the LFSR is undefined between $CONT_P$ primitives (i.e., the LFSR result used for $CONT_P$ sequence N is not required to be continuous from the last LFSR result of $CONT_P$ sequence N-1).

The sequence of LFSR values used to scramble the payload contents of a FIS shall not be affected by the scrambling of data used during repeated primitive suppression (i.e., The data payload LFSR shall not be advanced during repeated primitive suppression and shall only be advanced for each

data payload character scrambled using the data payload LFSR). See details as described in 7.6.33.3 for additional information on scrambling and repeated primitive suppression.

9.5.7.3 Periodic retransmission of sustained primitives (informative)

In order to be able to determine the state that a bus is in, it is recommended that a sustained primitive periodically be retransmitted. The only requirement is that the interval that the retransmit occurs is large enough that EMI is not substantially affected. Since two consecutive ALIGN$_P$ primitives are required to be sent at an interval of at most 256 Dwords, one solution to providing visibility to a suppressed primitive stream is to retransmit the suppressed primitive sequence immediately after the ALIGN$_P$ primitives are inserted.

In the following example PRIM represents a suppressed primitive.

EXAMPLE - If the original sequence was:

1) PRIM;
2) PRIM;
3) CONT$_P$;
4) junk. . . ;
5) ALIGN$_P$;
6) ALIGN$_P$; and
7) junk. . . ;

the new sequence is able to look like:

1) PRIM;
2) PRIM;
3) CONT$_P$;
4) junk . . .;
5) ALIGN$_P$;
6) ALIGN$_P$;
7) PRIM;
8) PRIM;
9) CONT$_P$; and
10) junk.

The actual interval chosen by an implementation is able to be longer. The goal is to make visible the primitive stream being sustained within the normal depth of a logic analyzer.

9.5.8 ALIGN$_P$ primitive

The Link layer shall ignore reception of ALIGN$_P$ primitives. The Phy layer is free to consume received ALIGN$_P$ primitives. Implementations where the Phy does not consume received ALIGN$_P$ primitives shall effectively drop received ALIGN$_P$ primitives at the input to the Link layer or shall include Link layer processing that yields behavior equivalent to the behavior produced if all received ALIGN$_P$ primitives are consumed by the Phy and not presented to the Link.

9.5.9 Flow control signaling latency

9.5.9.1 Flow control signaling latency overview

There is a finite pipeline latency in a round-trip handshake across the Serial ATA interface. In order to avoid buffer overflow in flow control situations, the maximum tolerable latency from when a receiver sends a HOLD$_P$ signal until it receives the HOLDA$_P$ signal from the transmitter is specified. This allows the high-water mark to be set to one in the receive FIFO so as to avoid buffer overflow while avoiding excessive buffering/FIFO space.

In the case where the receiver wants to flow control the incoming traffic, it transmits HOLD$_P$ characters on the back channel. Some number of received Dwords later HOLDA$_P$ is received. The larger the latency between transmitting HOLD$_P$ until receiving HOLDA$_P$, the larger the receive FIFO needs to be. Within a single HOLD$_P$/HOLDA$_P$ sequence, there shall be a maximum allowed latency

from the time the most significant bit (MSB) of the initial HOLD$_P$ is on the wire, until the MSB of the initial HOLDA$_P$ is on the wire shall be no more than 20 Dword times. The least significant bit (LSB) is transmitted first.

If operating at Gen1 and Gen2 transfer speeds, a receiver shall be able to accommodate reception of 20 Dwords of additional data after the time it transmits HOLD$_P$ to the transmitter, and the transmitter shall respond with HOLDA$_P$ in response to receiving a HOLD$_P$ within 20 Dwords. The 20 Dword latency specification is not applicable to any subsequent transmissions of HOLD$_P$ within the same sequence. Upon each new instance of a HOLD$_P$/HOLDA$_P$ sequence, the receiver and transmitter operating at Gen1 and Gen2 transfer speeds shall meet the 20 Dword latency specification.

If operating at Gen3 transfer speeds, a receiver shall be able to accommodate reception of 24 Dwords of additional data after the time it transmits HOLD$_P$ to the transmitter, and the transmitter shall respond with HOLDA$_P$ in response to receiving a HOLD$_P$ within 20 Dwords. The Dword latency specification is not applicable to any subsequent transmissions of HOLD$_P$ within the same sequence. Upon each new instance of a HOLD$_P$/HOLDA$_P$ sequence, the receiver and transmitter operating at Gen3 transfer speeds shall meet this Dword latency specification.

There is no reference design in this specification. The specified maximum latency is based on the layers and states described throughout this specification. It is recognized that the Link layer may have two separate clock domains - transmit clock domain, and the receive clock domain. It is also recognized that a Link state machine there is a possibility to run at the Dword clock rate, implying synchronizers between three potential clock domains. In practice more efficient implementations are pursued and the actual latencies may be less than indicated here. The figures represent an almost literal interpretation of the spec into logic design. A synchronizer is assumed to be a worst case of 2.99 clocks of any clock domain and is rounded to three whole clocks. The Serial ATA cable contains less than half of a Dword of content at Gen1i and Gen2i speeds with 1 m internal cables, and is therefore negligible. For longer cable lengths, the effect of the cable should not be ignored. Two Dwords of pipeline delay are assumed for the Phy, and the FIFO is assumed to run at the Link state machine rate. No synchronization is needed between the two.

Table 95 outlines the origin of the 20 Dword latency specification. The example illustrates the components of a round trip delay when the receiver transmits HOLD$_P$ on the link until reception of the HOLDA$_P$ from the transmitter. This corresponds to the number of Dwords that the receiver shall be able to accept after transmitting a HOLD$_P$.

Table 95 – Example of components of a round trip delay

Receiver Sends $HOLD_P$	
1 Dword	Convert to 40 bit data.
1 Dword	10b/8b decoding.
1 Dword	De-scrambling.
3 Dwords	Synchronization between receive clock, and Link state machine clock.
1 Dword	Link state machine is notified that primitive has been received.
1 Dword	Link state machine takes action.
1 Dword	FIFO is notified of primitive reception.
1 Dword	FIFO stops sending data to Link layer.
1 Dword	Link is notified to insert $HOLDA_P$.
1 Dword	Link acts on notification and inserts $HOLDA_P$ into data stream.
1 Dword	Scrambling.
1 Dword	8b/10b encoding.
1 Dword	Synchronize to transmit clock (3 transmit clocks that are four times the Link state machine rate).
1 Dword	Convert to 10 bit data.
2 Dwords	Phy, transmit side.
$HOLDA_P$ on the Cable	

9.5.9.2 Cable length and flow control latency (informative)

For hosts and devices that are designed for use in data center applications that cables longer than 1 m are used, it is advised that these designs accommodate potential increased effects on overflow latencies. If operating at 3.0 Gbit/s with a 1 m cable, the cable contains less than half a Dword of content at any point in time and thus the latency effect from the cable is ignored in flow control calculations. If cables longer than 1 m are used, the effect of the cable on flow control latency should be accounted for in the system design.

A system design may account for these effects in a multitude of ways.

EXAMPLE - Some examples include:

a) hosts or devices may be selected that meet more stringent flow control requirements;
b) hosts or devices may be selected that have a larger flow control buffer and absorb more than 20 Dwords of latency; and
c) do not select excessive cables lengths over what is required for the environment as it impacts flow control latencies.

9.5.10 Examples of primitive usage (informative)

Table 96, Table 97, and Table 98 are examples that illustrate basic primitive usage. They do not show detailed lengthy sequences that invoke the use of CONT$_P$.

Table 96 – SRST write from host to device transmission breaking through a device to host data FIS

Host	Device	Description
…	…	Previous activity abbreviated for clarity.
R_IP$_P$	DATA n	Device transmitting data.
R_IP$_P$	DATA n+1	
R_IP$_P$	HOLD$_P$	Device transmit FIFO empty, and flow control applied.
R_IP$_P$	HOLD$_P$	Host receives and decodes HOLD$_P$ flow control.
HOLDA$_P$	HOLD$_P$	Host acknowledges flow control. Device internally deadlocked and no more data forthcoming (device hung) .
HOLDA$_P$	HOLD$_P$	
…	…	System in this state until host decides to reset device.
HOLDA$_P$	HOLD$_P$	Host detects SRST write to Device Control register, needs to break deadlock.
SYNC$_P$	HOLD$_P$	Host transmits SYNC$_P$ to abort current transmission.
SYNC$_P$	HOLD$_P$	Device receives and decodes SYNC$_P$, abandons transmission in progress.
SYNC$_P$	SYNC$_P$	Host sends SYNC$_P$ / device sends SYNC$_P$ (both returned to idle state).
SYNC$_P$	SYNC$_P$	Host receives and decodes SYNC$_P$, may now initiate new FIS transmission.
X_RDY$_P$	SYNC$_P$	Host ready to send Shadow Register Block registers for SRST write.
X_RDY$_P$	SYNC$_P$	Device decodes X_RDY$_P$.
X_RDY$_P$	R_RDY$_P$	Device indicates ready to receive.
X_RDY$_P$	R_RDY$_P$	Host decodes R_RDY$_P$.
SOF$_P$	R_RDY$_P$	Host starts a frame.
etc	etc	etc

Table 97 – Command Shadow Register Block register transmission example

Host	Device	Description
SYNCP	SYNCP	Idle condition.
SYNCP	SYNCP	Idle condition.
X_RDYP	SYNCP	Host ready to send Shadow Register Block registers.
X_RDYP	SYNCP	Device decodes X_RDYP.
X_RDYP	R_RDYP	Device indicates ready to receive.
X_RDYP	R_RDYP	Host decodes R_RDYP.
SOFP	R_RDYP	Host starts a frame.
DATA 0	R_RDYP	Host sends Register Host to Device FIS Dword 0 / device decodes SOFP.
DATA 1	R_IPP	Host sends Register Host to Device FIS Dword 1 / device stores DATA Dword 0.
…	…	…
DATA n	R_IPP	Host sends Register Host to Device FIS Dword n / device stores DATA Dword n-1.
CRC	R_IPP	Host sends CRC / device stores DATA Dword n.
EOFP	R_IPP	Host sends EOFP / device stores CRC.
WTRMP	R_IPP	Device decodes EOFP.
WTRMP	R_IPP	Device computes good CRC and releases TF contents.
WTRMP	R_OKP	Device sends good end.
WTRMP	R_OKP	Host decodes R_OKP as good results.
SYNCP	R_OKP	Host releases interface.
SYNCP	R_OKP	Device decodes release by host - is allowed to release.
SYNCP	SYNCP	Idle condition.

Table 98 – Data from host to device transmission example

Host	Device	Description
SYNC$_P$	SYNC$_P$	Idle condition.
SYNC$_P$	SYNC$_P$	Idle condition.
X_RDY$_P$	SYNC$_P$	Host ready to send Shadow Register Block registers.
X_RDY$_P$	SYNC$_P$	Device decodes X_RDY$_P$.
X_RDY$_P$	R_RDY$_P$	Device indicates ready to receive.
X_RDY$_P$	R_RDY$_P$	Host decodes R_RDY$_P$.
SOF$_P$	R_RDY$_P$	Host starts a frame.
DATA 0	R_RDY$_P$	Host sends DATA Dword 0 / device decodes SOF$_P$.
DATA 1	R_IP$_P$	Host sends DATA Dword 1 / device stores DATA Dword 0.
...	...	...
DATA x	R_IP$_P$	Host sends DATA Dword x / device stores DATA Dword (x-1) .
HOLD$_P$	R_IP$_P$	Host sends HOLD$_P$ / device stores DATA Dword (x) and decodes HOLD$_P$.
HOLD$_P$	HOLDA$_P$	Device acknowledges HOLD$_P$.
HOLD$_P$	HOLDA$_P$	Host decodes HOLDA$_P$ – host may release HOLD$_P$ at any time.
DATA(n-2)	HOLDA$_P$	Host sends (n-2)th DATA Dword / device decodes DATA Dword.
DATA(n-1)	R_IP$_P$	Host sends (n-1)th data Dword / device stores (n-2)th DATA Dword.
DATA(n)	R_IP$_P$	Host sends nth data Dword / device stores (n-1)th DATA Dword.
CRC	R_IP$_P$	Host sends CRC / device stores nth DATA Dword.
EOF$_P$	R_IP$_P$	Host sends EOF$_P$ / device stores CRC.
WTRM$_P$	R_IP$_P$	Device decodes EOF$_P$.
WTRM$_P$	R_IP$_P$	Device computes good CRC and releases data contents.
WTRM$_P$	R_OK$_P$	Device sends good end.
WTRM$_P$	R_OK$_P$	Host decodes R_OK$_P$ as good results.
SYNC$_P$	R_OK$_P$	Host releases interface.
SYNC$_P$	R_OK$_P$	Device decodes release by host - is allowed to release.
SYNC$_P$	SYNC$_P$	Idle condition.

9.6 Cyclic Redundancy Check (CRC) and scrambling

9.6.1 Cyclic Redundancy Check (CRC) and scrambling overview

The CRC of a frame is a Dword (32 bit) field that shall follow the last Dword of the contents of a FIS and precede EOF_P. The CRC calculation covers all of the FIS transport data between the SOF_P and EOF_P primitives, and excludes any intervening primitives and $CONT_P$ stream contents. The CRC value shall be computed on the contents of the FIS before encoding for transmission (scrambling) and after decoding upon reception.

The CRC shall be calculated on Dword quantities. If a FIS contains an odd number of Words the last Word of the FIS shall be padded with zeros to a full Dword before the Dword is used in the calculation of the CRC.

The CRC shall be aligned on a Dword boundary.

The CRC shall be calculated using the following 32 bit generator polynomial:

$$G(X) = X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$$

The CRC value shall be initialized with a value of 5232 5032h before the calculation begins.

The maximum number of Dwords between SOF_P to EOF_P shall not exceed 2 064 Dwords including the FIS Type and CRC.

The contents of a frame shall be scrambled before transmission by the Phy layer.

Scrambling shall be performed on Dword quantities by exclusive logical ORing (XORing) the data to be transmitted with the output of a linear feedback shift register (LFSR). The shift register shall implement the following polynomial:

$$G(X) = X^{16} + X^{15} + X^{13} + X^4 + 1$$

The serial shift register shall be initialized with the seed value of FFFFh before the first shift of the LFSR. The shift register shall be initialized to the seed value before SOF_P is transmitted. All data Words between the SOF_P and EOF_P shall be scrambled, including the CRC.

9.6.2 Relationship between scrambling of FIS data and repeated primitives

There are two separate scramblers used in Serial ATA. One scrambler is used for the data payload encoding and a separate scrambler is used for repeated primitive suppression. The scrambler used for data payload encoding shall maintain consistent and contiguous context over the scrambled payload characters of a frame (between SOF_P and EOF_P), and shall not have its context affected by the scrambling of data used for repeated primitive suppression.

Scrambling is applied to all data (non-primitive) Dwords. Primitives, including $ALIGN_P$, do not get scrambled and shall not advance the data payload LFSR register. Similarly, the data payload LFSR shall not be advanced during transmission of Dwords during repeated primitive suppression (i.e., after a $CONT_P$ primitive). Since it is possible for a repeated primitive stream to occur in the middle of a data frame – multiple $HOLD_P$/$HOLDA_P$ primitives are likely – care should be taken to insure that the data payload LFSR is only advanced for each data payload character that it scrambles and that it is not advanced for primitives or for data characters transmitted as part of repeated primitive suppression that uses a separate scrambler.

9.6.3 Relationship between scrambling and CRC

The order of application of scrambling shall be as follows. For a Dword of data following SOF_P the Dword shall be used in the calculation of the CRC. The same Dword value shall be XORed with the scrambler output, and the resulting Dword submitted to the 8b/10b encoder for transmission.

Similarly, on reception, the Dword shall be decoded using a 10b/8b decoder, the scrambler output shall be XORed with the resulting Dword, and the resulting Dword presented to the Link layer and subsequently used in calculating the CRC. The CRC Dword shall be scrambled according to the same rules.

9.6.4 Scrambling disable (informative)

Hosts and devices should provide a vendor-specific means of disabling the transmission/reception of scrambled data. Three independent controls are recommended – one to disable the scrambling of transmitted FIS payload data, the second to disable the $CONT_P$/junk method of repeated primitive suppression, and the third to disable the unscrambling of received FIS payload data.

Using the scrambling disable capabilities is intended for testability and design debug, and not recommended as an end-user feature. It is the responsibility of the engineer/operator to ensure that both ends of the cable are configured in such a way that the host and device may communicate (i.e., if scrambled transmission is disabled on the device, then scrambled reception shall be disabled on the host). Devices that disable payload scrambling may not interoperate with other devices that do not implement this recommendation. Systems that disable scrambling may not meet EMI regulatory requirements.

9.7 Link layer state machine

9.7.1 Terms used in Link layer transition tables

Terms used in Link layer transition tables are:

a) LRESET, Link layer COMRESET or COMINIT signal;
b) PHYRDYn, the negation of the PHYRDY signal;
c) PHYRDY, Phy status as defined in 7.2.3;
d) DecErr, bad decode of a 32 bit Dword transferred from Phy to Link:
    A) invalid 10b/8b decoded pattern;
    B) disparity error;
    C) primitive with a control character in the first byte but not an allowed control character; or
    D) any control character in other than the first byte of the Dword;
e) DatDword, a 32 bit pattern formed correctly, but does not have the primitive leading 10b/8b decode pattern (i.e., K28.5 or K28.3);
f) COMWAKE, signal from the OOB detector in the Phy indicating that the COMWAKE OOB signal is being detected; and
g) AnyDword, a 32 bit pattern of any type - even one with DecErr received from Phy.

9.7.2 Link idle state machine

The link idle state machine is defined in Figure 270.

L1: L_IDLE [d]	Transmit $SYNC_P$.		
1. Transport layer requests frame transmission and PHYRDY [b].		→	HL_SendChkRdy or DL_SendChkRdy [a]
2. Transport layer requests transition to Partial and PHYRDY [b e].		→	L_TPMPartial
3. Transport layer requests transition to Slumber and PHYRDY [b e].		→	L_TPMSlumber
4. X_RDY_P received from Phy.		→	L_RcvWaitFifo
5. Phy layer forwards ($PMREQ\_P_P$ or $PMREQ\_S_P$) and power modes are enabled and acceptable.		→	L_PMOff
6. Phy layer forwards ($PMREQ\_P_P$ or $PMREQ\_S_P$) and power modes are disabled or are unacceptable.		→	L_PMDeny
7. Phy layer forwards AnyDword other than (X_RDY_P or $PMREQ\_P_P$ or $PMREQ\_S_P$) and no transmit request from Transport layer [b c].		→	L_IDLE
8. PHYRDYn		→	L_NoCommErr

[a] The host Link layer makes a transition to the HL_SendChkRdy state, and the device Link layer makes a transition to the DL_SendChkRdy state.
[b] This transition is taken even if errors (i.e., 10b/8b decoding errors) are detected.
[c] This statement also ignores any unrecognized sequences or commands not defined in this specification.
[d] Upon entry to this state from the LS3:L_SendAlign state or the LPM8:L_WakeUp2 state, use of $CONT_P$ is not allowed until either a minimum of 10 non-$ALIGN_P$ primitives have been transmitted or until receipt of a primitive other than $SYNC_P$ or $ALIGN_P$ has been detected.
[e] Hosts shall not attempt initiating an interface power state transition between an issued reset and the receipt of the device reset signature. Hosts should not attempt initiating an interface power management request without first verifying the device has such capabilities as determined by the information in the device's IDENTIFY DEVICE (or IDENTIFY PACKET DEVICE) data structure.

Figure 270 – Link layer state machine (part 1 of 2)

L2: L_SyncEscape [a]	Transmit $SYNC_P$.	
1. AnyDword other than X_RDY_P or $SYNC_P$ received from Phy.	→	L_SyncEscape
2. X_RDY_P or $SYNC_P$ received from Phy.	→	L_IDLE
3. PHYRDYn	→	L_NoCommErr [b]
[a] This state is entered asynchronously from any other Link layer state where the Link layer has transmitted $SYNC_P$ to escape a FIS transfer, also known as a SYNC Escape. [b] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		

LS1: L_NoCommErr	Post Phy not ready error to Transport layer.	
1. Unconditional	→	L_NoComm

LS2: L_NoComm	Transmit $ALIGN_P$ [a].	
1. PHYRDYn	→	L_NoComm
2. PHYRDY	→	L_SendAlign
[a] Also deactivate any signal for Phy layer to abort operation.		

LS3: L_SendAlign	Transmit $ALIGN_P$.	
1. PHYRDYn	→	L_NoCommErr
2. PHYRDY	→	L_IDLE

LS4: L_RESET [a]	Reset Link state to initial conditions.	
1. LRESET Link reset signal asserted.	→	L_RESET
2. LRESET Link reset signal negated.	→	L_NoComm
[a] This state is entered asynchronously if the link reset control is active.		

Figure 270 – Link layer state machine (part 2 of 2)

L1: L_IDLE state, this state is entered if a frame transmission has been completed by the Link layer.

If in this state, the Link layer transmits $SYNC_P$ and waits for X_RDY_P from the Phy layer or a frame transmission request from the Transport layer.

Transition L1:1a, if the host Link layer receives a request to transmit a frame from the Transport layer and the Phy layer is ready, the Link layer shall make a transition to the LT1: HL_SendChkRdy state.

Transition L1:1b, if the device Link layer receives a request to transmit a frame from the Transport layer and the Phy layer is ready, the Link layer shall make a transition to the LT2: DL_SendChkRdy state.

Transition L1:2, if the Link layer receives a request to enter the Partial power mode from the Transport layer and the Phy layer is ready, the Link layer shall make a transition to the L_TPMPartial state.

Transition L1:3, if the Link layer receives a request to enter the Slumber power mode from the Transport layer and the Phy layer is ready, the Link layer shall make a transition to the L_TPMSlumber state.

Transition L1:4, if the Link layer receives an X_RDY$_P$ from the Phy layer, the Link layer shall make a transition to the LR2: L_RcvWaitFifo state.

Transition L1:5, if the Link layer receives a PMREQ_P$_P$ or PMREQ_S$_P$ from the Phy layer, is enabled to perform power management modes, and in a state to accept power mode requests, the Link layer shall make a transition to the LPM3: L_PMOff state.

Transition L1:6, if the Link layer receives a PMREQ_P$_P$ or a PMREQ_S$_P$ from the Phy layer and is not enabled to perform power management modes or is not in a state to accept power mode requests, the Link layer shall make a transition to the LR0: L_PMDeny state. This transition is still valid if interface power states are supported and enabled as verified by Word 76 bit 9 set to one in IDENTIFY (PACKET) DEVICE data.

Transition L1:7, if the Link layer does not receive a request to transmit a frame from the Transport layer, does not receive a request to go to a power mode from the Transport layer, does not receive an X_RDY$_P$ from the Phy layer or does not receive a PMREQ_x$_P$ from the Phy layer the Link layer shall make a transition to the L1: L_IDLE state.

Transition L1:8, if the Phy layer becomes not ready even if the Transport layer is requesting an operation, the Link layer transitions to the L_NoCommErr state.

L2: L_SyncEscape state, this state is entered if the Link layer transmits SYNC$_P$ to escape a FIS transmission. The Link layer may choose to escape a FIS transmission due to a request from the Transport layer or due to an invalid state transition. This state is only entered by the initiator of the SYNC Escape.

If in this state, the Link layer transmits SYNC$_P$ and waits for a SYNC$_P$ from the Phy layer before proceeding to L_IDLE. The Link layer also transitions to L_IDLE if X_RDY$_P$ is received in order to avoid a deadlock condition.

Transition L2:1, if the Link layer receives any Dword from the Phy that is not X_RDY$_P$ or SYNC$_P$, the Link layer shall make a transition to the L2: L_SyncEscape state.

Transition L2:2, if the Link layer receives X_RDY$_P$ or SYNC$_P$ from the Phy, the Link layer shall make a transition to the L1: L_IDLE state.

Transition L2:3, if the host Link layer detects that the Phy layer is not ready the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

LS1: L_NoCommErr state, this state is entered upon detection of a non ready condition of the Phy layer while attempting to process another state. The entry into this state heralds a relatively serious error condition in the Link layer. This state is processed only once so as to pass on the error condition up to the Transport layer.

Transition LS1:1, the transition is made to LS1:L_NoComm unconditionally.

LS2: L_NoComm state, this state is entered directly from the LS1:L_NoCommErr state or the LS4:L_RESET State. The Link layer remains in this state until the Phy signals that it has established communications and is ready.

Transition LS2:1, for as long as the Phy layer stays not ready, the transition is made to LS2: L_NoComm.

Transition LS2:2, if the Phy layer signals it is ready, a transition is made to LS3: L_SendAlign.

LS3: L_SendAlign state, this state is entered if an $ALIGN_P$ needs to be sent to the Phy layer.

Transition LS3:1, if the Phy layer becomes not ready, then a transition is made to LS1: L_NoCommErr.

Transition LS3:2, if the Phy layer indicates that it is ready, a transition is made to the L1: L_IDLE state.

LS4: L_RESET state, this state is entered if the Link LRESET control is active. All Link layer hardware is initialized to and held at a known state/value. While in this state all requests or triggers from other layers are ignored. While in this state, the Phy reset signal is also asserted.

Transition LS4:1, while the RESET control is active a transition is made back to the LS4: L_RESET state.

Transition LS4:2, if the RESET control goes inactive a transition is made to the LS2: L_NoComm state.

9.7.3 Link transmit state machine

The link transmitter state machine is defined in Figure 271.

LT1: HL_SendChkRdy	Transmit X_RDY_P.	
1. R_RDY_P received from Phy.	→	L_SendSOF
2. X_RDY_P received from Phy.	→	L_RcvWaitFifo
3. AnyDword other than (R_RDY_P or X_RDY_P) [a] received from Phy layer.	→	HL_SendChkRdy
4. PHYRDYn	→	L_NoCommErr [b]
[a] Any received errors (e.g., 10b/8b decoding errors and invalid primitives) are ignored. [b] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		

LT2: DL_SendChkRdy	Transmit X_RDY_P.	
1. R_RDY_P received from Phy.	→	L_SendSOF
2. AnyDword other than R_RDY_P received from Phy.	→	DL_SendChkRdy
3. PHYRDYn	→	L_NoCommErr

LT3: L_SendSOF	Transmit SOF_P	
1. PHYRDY [a]	→	L_SendData
2. PHYRDYn	→	L_NoCommErr [b]

LT3: L_SendSOF	Transmit SOF_P		
	3. $SYNC_P$ received from Phy.	→	L_IDLE [b]
	[a] Any received errors (e.g., 10b/8b decoding errors and invalid primitives) are ignored. [b] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		

Figure 271 – Link layer transmit state machine (part 1 of 4)

LT4: L_SendData	Transmit data Dword	
1. More data to transmit and AnyDword other than ($HOLD_P$ or $DMAT_P$ or $SYNC_P$) received from Phy [a] [b].	→	L_SendData
2. More data to transmit and $HOLD_P$ received from Phy.	→	L_RcvrHold
3. Data transmit not complete and data not ready to transmit and AnyDword other than $SYNC_P$ received from Phy.	→	L_SendHold
4. $DMAT_P$ received from Phy or data transmit complete and AnyDword other than $SYNC_P$ received from Phy.	→	L_SendCRC [e]
5. $SYNC_P$ received from Phy.	→	L_IDLE [c]
6. PHYRDYn	→	L_NoCommErr [c]
7. Transport layer indicates request to escape current frame [d].	→	L_SyncEscape

[a] Any received errors (e.g., 10b/8b decoding errors and invalid primitives) are ignored.
[b] This makes possible a back channel during this time.
[c] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.
[d] If this condition is true, the associated transition has priority over all other transitions exiting this state.
[e] The $DMAT_P$ signal is advisory and data transmission should be halted at the earliest opportunity but is not required to cease immediately. It is allowable to stay in the LT4: L_SendData state if there is more data to transmit and $DMAT_P$ is received.

Figure 271 – Link layer transmit state machine (part 2 of 4)

LT5: L_RcvrHold	Transmit HOLDA$_P$.	
1. More data to transmit and AnyDword other than (HOLD$_P$ or SYNC$_P$ or DMAT$_P$) received from Phy with no DecErr.	→	L_SendData
2. More data to transmit and HOLD$_P$ received from Phy or DecErr.	→	L_RcvrHold
3. More data to transmit and SYNC$_P$ received from Phy.	→	L_IDLE [a]
4. More data to transmit and DMAT$_P$ received from Phy.	→	L_SendCRC [c]
5. PHYRDYn	→	L_NoCommErr [a]
6. Transport layer indicates request to escape current frame [b].	→	L_SyncEscape
7. SYNC$_P$ received from Phy.	→	L_IDLE [a]

[a] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.
[b] If this condition is true, the associated transition has priority over all other transitions exiting this state.
[c] The DMAT$_P$ signal is advisory and data transmission should be halted at the earliest opportunity but is not required to cease immediately. It is allowable to stay in the LT5: L_RcvrHold state if there is more data to transmit and DMAT$_P$ is received.

LT6: L_SendHold	Transmit HOLD$_P$.	
1. More data ready to transmit and AnyDword other than (HOLD$_P$ or SYNC$_P$) received from Phy.	→	L_SendData
2. More data ready to transmit and HOLD$_P$ received from Phy.	→	L_RcvrHold
3. Data transmit not complete, and data not ready to transmit, and AnyDword other than (SYNC$_P$ or DMAT$_P$) received from Phy.	→	L_SendHold
4. DMAT$_P$ received from Phy or data transmit complete, and AnyDword other than SYNC$_P$ received from Phy.	→	L_SendCRC [c]
5. SYNC$_P$ received from Phy.	→	L_IDLE [a]
6. PHYRDYn	→	L_NoCommErr [a]
7. Transport layer indicates request to escape current frame [b].	→	L_SyncEscape

[a] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.
[b] If this condition is true, the associated transition has priority over all other transitions exiting this state.
[c] The DMAT$_P$ signal is advisory and data transmission should be halted at the earliest opportunity but is not required to cease immediately. It is allowable to stay in the LT6: L_SendHold state if there is more data to transmit and DMAT$_P$ is received.

Figure 271 – Link layer transmit state machine (part 3 of 4)

LT7: L_SendCRC	Transmit CRC.	
1. PHYRDY and $SYNC_P$ not received from Phy.	→	L_SendEOF
2. PHYRDYn	→	L_NoCommErr [a]
3. PHYRDY and $SYNC_P$ received from Phy.	→	L_IDLE [a]
[a] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		

LT8: L_SendEOF	Transmit EOF_P.	
1. PHYRDY and $SYNC_P$ not received from Phy.	→	L_Wait
2. PHYRDYn	→	L_NoCommErr [a]
3. PHYRDY and $SYNC_P$ received from Phy.	→	L_IDLE [a]
[a] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		

LT9: L_Wait	Transmit $WTRM_P$.	
1. R_OK_P received from Phy.	→	L_IDLE (good status)
2. R_ERR_P received from Phy.	→	L_IDLE (bad status)
3. $SYNC_P$ received from Phy.	→	L_IDLE [a]
4. AnyDword other than (R_OK_P or R_ERR_P or $SYNC_P$) received from Phy.	→	L_Wait
5. PHYRDYn	→	L_NoCommErr [a]
[a] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		

Figure 271 – Link layer transmit state machine (part 4 of 4)

LT1: HL_SendChkRdy state, this state is entered if a frame transmission has been requested by the host Transport layer.

If in this state, the Link layer transmits X_RDY_P and waits for X_RDY_P or R_RDY_P from the Phy layer.

It is possible that both the host and the device simultaneously request frame transmission by transmitting X_RDYP. If the host receives X_RDYP while transmitting X_RDYP, the host shall back off and enter the L_RcvWaitFifo state, postponing its desired frame transmission until the device has completed its frame transmission and the bus is idle.

Transition LT1:1, if the host Link layer receives R_RDY_P from the Phy layer, the Link layer shall make a transition to the LT3: L_SendSOF state.

Transition LT1:2, if the host Link layer receives X_RDY_P from the Phy layer, the Link layer shall make a transition to the LR2: L_RcvWaitFifo state.

Transition LT1:3, if the host Link layer receives any Dword other than R_RDY$_P$ or X_RDY$_P$ from the Phy layer, the Link layer shall make a transition to the LT1: HL_SendChkRdy state.

Transition LT1:4, if the host Link layer detects that the Phy layer is not ready the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

LT2: DL_SendChkRdy state, this state is entered if a frame transmission has been requested by the device Transport layer.

If in this state, the Link layer transmits X_RDY$_P$ and waits for R_RDY$_P$ from the Phy layer.

Transition LT2:1, if the device Link layer receives R_RDY$_P$ from the Phy layer, the Link layer shall make a transition to the LT3: L_SendSOF state.

Transition LT2:2, if the device Link layer does not receive R_RDY$_P$ from the Phy layer, the Link layer shall make a transition to the LT2: DL_SendChkRdy state.

Transition LT2:3, if the device Link layer detects that the Phy layer is not ready the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

LT3: L_SendSOF state, this state is entered if R_RDY$_P$ has been received from the Phy layer.

If in this state, the Link layer transmits SOF$_P$.

Transition LT3:1, if the device Link layer has transmitted SOF$_P$, the Link layer shall make a transition to the LT4: L_SendDATA state.

Transition LT3:2, if the Link layer detects that the Phy layer is not ready the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

Transition LT3:3, if the Link layer receives SYNC$_P$ from the Phy layer, the Link layer shall notify the Transport layer of the illegal transition error condition and shall make a transition to the L1:L_IDLE state.

LT4: L_SendData state, this state is entered if SOF$_P$ has been transmitted.

If in this state, the Link layer takes a data Dword from the Transport layer, encodes the Dword, and transmits it. The Dword is also entered into the CRC calculation before encoding.

Transition LT4:1, if the Link layer receives any Dword other than a HOLD$_P$, DMAT$_P$, or SYNC$_P$ primitive from the Phy layer and the Transport layer indicates a Dword is available for transfer, the Link layer shall make a transition to the LT4: L_SendData state. The DMAT$_P$ signal is advisory and data transmission should be halted at the earliest opportunity but is not required to cease immediately. It is therefore allowable to stay in the LT4: L_SendData state if there is more data to transmit and DMAT$_P$ is received.

Transition LT4:2, if the device Link layer receives HOLD$_P$ from the Phy layer, the Link layer shall make a transition to the LT5: L_RcvrHold state.

Transition LT4:3, if the Transport layer indicates that the next Dword is not available to transfer and any Dword other than SYNC$_P$ has been received from the Phy layer, the Link layer shall make a transition to the LT6: L_SendHold state.

Transition LT4:4, if the Transport layer indicates that all data for the frame has been transferred and any Dword other than SYNC$_P$ has been received from the Phy layer, the Link layer shall make a transition to the LT7: L_SendCRC state. If the Link layer receives DMAT$_P$ from the Phy layer, it

shall notify the Transport layer and terminate the transmission in progress as defined in 9.5.6 and shall transition to the LT7: L_SendCRC state.

Transition LT4:5, if the Link layer receives $SYNC_P$ from the Phy layer, the Link layer shall notify the Transport layer of the illegal transition error condition and shall make a transition to the L1:L_IDLE state.

Transition LT4:6, if the Link layer detects that the Phy layer is not ready the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

Transition LT4:7, if the Link layer receives notification from the Transport layer that the current frame transfer should be escaped, a transition to the L_SyncEscape state shall be made.

LT5: L_RcvrHold state, this state is entered if $HOLD_P$ has been received from the Phy layer.

If in this state, the Link layer shall transmit $HOLDA_P$.

Transition LT5:1, if the Link layer receives any Dword other than a $HOLD_P$, $SYNC_P$, or a $DMAT_P$ primitive from the Phy layer with no decoding error detected, and the Transport layer indicates that a Dword is available for transfer, the Link layer shall make a transition to the LT4: L_SendData state.

Transition LT5:2, if the device Link layer receives $HOLD_P$ from the Phy layer or a decoding error was detected, the Link layer shall make a transition to the LT5: L_RcvrHold state.

Transition LT5:3, if the Link layer receives $SYNC_P$ from the Phy layer, the Link layer shall make a transition to the L1: L_IDLE state. The Transport layer shall be notified of the illegal transition error condition.

Transition LT5:4, if the Link layer receives $DMAT_P$ from the Phy layer, it shall notify the Transport layer and terminate the transmission in progress as defined in 9.5.6 and shall transition to the LT7: L_SendCRC state.

Transition LT5:5, if the Link layer detects that the Phy layer is not ready the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

Transition LT5:6, if the Link layer receives notification from the Transport layer that the current frame should be escaped, a transition to the L_SyncEscape state shall be made.

Transition LT5:7, if the Link layer receives $SYNC_P$ from the Phy layer, the Link layer shall notify the Transport layer of the illegal transition error condition and shall make a transition to the L1:L_IDLE state.

LT6: L_SendHold state, this state is entered if the Transport layer indicates a Dword is not available for transfer and $HOLD_P$ has not been received from the Phy layer.

If in this state, the Link layer shall transmit $HOLD_P$.

Transition LT6:1, if the Link layer receives any Dword other than a $HOLD_P$ or $SYNC_P$ primitive from the Phy layer and the Transport layer indicates that a Dword is available for transfer, the Link layer shall make a transition to the LT4: L_SendData state.

Transition LT6:2, if the Link layer receives $HOLD_P$ from the Phy layer and the Transport layer indicates a Dword is available for transfer, the Link layer shall make a transition to the LT5: L_RcvrHold state.

Transition LT6:3, if the Transport layer indicates that a Dword is not available for transfer and any Dword other than $SYNC_P$ is received from the Phy layer, the Link layer shall make a transition to the LT6: L_SendHold state.

Transition LT6:4, if the Transport layer indicates that all data for the frame has been transferred and any Dword other than $SYNC_P$ has been received from the Phy layer, the Link layer shall make a transition to the LT7: L_SendCRC state. If the Link layer receives $DMAT_P$ from the Phy layer, it shall notify the Transport layer and terminate the transmission in progress as defined in 9.5.6 and shall transition to the LT7:L_SendCRC state.

Transition LT6:5, if the Link layer receives $SYNC_P$ from the Phy layer, the Link layer shall make a transition to the L1: L_IDLE state. The Transport layer shall be notified of the illegal transition error condition.

Transition LT6:6, if the Link layer detects that the Phy layer is not ready the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

Transition LT6:7, if the Link layer receives notification from the Transport layer that the current frame should be escaped, a transition to the L_SyncEscape state shall be made.

LT7: L_SendCRC state, this state is entered if the Transport layer indicates that all data Dwords have been transferred for this frame.

If in this state, the Link layer shall transmit the calculated CRC for the frame.

Transition LT7:1, if the CRC has been transmitted, the Link layer shall make a transition to the LT8: L_SendEOF state.

Transition LT7:2, if the Link layer detects that the Phy layer is not ready the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

Transition LT7:3, if the Link layer receives $SYNC_P$ from the Phy layer, the Link layer shall notify the Transport layer of the illegal transition error condition and shall make a transition to the L1:L_IDLE state.

LT8: L_SendEOF state, this state is entered if the CRC for the frame has been transmitted.

If in this state, the Link layer shall transmit EOF_P.

Transition LT8:1, if EOF_P has been transmitted, the Link layer shall make a transition to the LT9: L_Wait state.

Transition LT8:2, if the Link layer detects that the Phy layer is not ready the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

Transition LT8:3, if the Link layer receives $SYNC_P$ from the Phy layer, the Link layer shall notify the Transport layer of the illegal transition error condition and shall make a transition to the L1:L_IDLE state.

LT9: L_Wait state, this state is entered if EOF_P has been transmitted.

If in this state, the Link layer shall transmit $WTRM_P$.

Transition LT9:1, if the Link layer receives R_OK_P from the Phy layer, the Link layer shall notify the Transport layer and make a transition to the L1: L_IDLE state.

Transition LT9:2, if the Link layer receives R_ERR$_P$ from the Phy layer, the Link layer shall notify the Transport layer and make a transition to the L1: L_IDLE state.

Transition LT9:3, if the Link layer receives SYNC$_P$ from the Phy layer, the Link layer shall notify the Transport layer and make a transition to the L1: L_IDLE state.

Transition LT9:4, if the Link layer receives any Dword other than an R_OK$_P$, R_ERR$_P$, or SYNC$_P$ primitive from the Phy layer, the Link layer shall make a transition to the LT9: L_Wait state.

Transition LT9:5, if the Link layer detects that the Phy layer is not ready the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

9.7.4 Link receive state machine

The link receive state machine is defined in Figure 272.

LR1: L_RcvChkRdy	Transmit R_RDY$_P$.	
1. X_RDY$_P$ received from Phy.	→	L_RcvChkRdy
2. SOF$_P$ received from Phy.	→	L_RcvData
3. Any Dword other than (X_RDY$_P$ or SOF$_P$) received from Phy.	→	L_IDLE
4. PHYRDYn	→	L_NoCommErr [a]
[a] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		

LR2: L_ RcvWaitFifo	Transmit SYNC$_P$.	
1. X_RDY$_P$ received from Phy and FIFO space available.	→	L_RcvChkRdy
2. X_RDY$_P$ received from Phy and FIFO space not available.	→	L_RcvWaitFifo
3. Any Dword other than X_RDY$_P$ received from Phy.	→	L_IDLE
4. PHYRDYn	→	L_NoCommErr [a]
[a] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		

Figure 272 – Link layer receiver state machine (part 1 of 4)

LR3: L_RcvData	Transmit R_IP$_P$ or DMAT$_P$ [a].	
1. (DatDword received from Phy and FIFO space) or HOLDA$_P$ received from Phy.	→	L_RcvData
2. DatDword received from Phy and insufficient FIFO space.	→	L_Hold
3. HOLD$_P$ received from Phy.	→	L_RcvHold
4. EOF$_P$ received from Phy.	→	L_RcvEOF
5. WTRM$_P$ received from Phy.	→	L_BadEnd
6. SYNC$_P$ received from Phy.	→	L_IDLE
7. AnyDword other than (HOLD$_P$, EOF$_P$, HOLDA$_P$, SYNC$_P$, or WTRM$_P$) received from Phy.	→	L_RcvData
8. PHYRDYn	→	L_NoCommErr [b]
9. Transport layer indicates request to escape current frame.	→	L_SyncEscape
[a] If the Transport layer signals that it wishes to terminate the transfer, DMAT$_P$ is transmitted in place of R_IP$_P$. [b] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		

LR4: L_Hold	Transmit HOLD$_P$.	
1. FIFO space available and AnyDword other than HOLD$_P$ or EOF$_P$ received from Phy.	→	L_RcvData
2. FIFO space available and HOLD$_P$ received from Phy.	→	L_RcvHold
3. EOF$_P$ received from Phy.	→	L_RcvEOF
4. No FIFO space available and EOF$_P$ not received from Phy and SYNC$_P$ not received from Phy and PHYRDY.	→	L_Hold
5. PHYRDYn	→	L_NoCommErr [a]
6. SYNC$_P$ received from Phy.	→	L_IDLE
7. Transport layer indicates request to escape current frame.	→	L_SyncEscape
[a] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		

Figure 272 – Link layer receiver state machine (part 2 of 4)

LR5: L_RcvHold	Transmit $HOLDA_P$ or $DMAT_P$ [a].	
1. AnyDword other than ($HOLD_P$ or EOF_P or $SYNC_P$) received from Phy.	→	L_RcvData
2. $HOLD_P$ received from Phy.	→	L_RcvHold
3. EOF_P received from Phy.	→	L_RcvEOF
4. $SYNC_P$ received from Phy.	→	L_IDLE
5. PHYRDYn	→	L_NoCommErr [b]
6. Transport layer indicates request to escape current frame.	→	L_SyncEscape
[a] If the Transport layer signals that it wishes to terminate the transfer, $DMAT_P$ is transmitted in place of $HOLDA_P$. [b] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		

LR6: L_RcvEOF	Transmit R_IP_P.	
1. CRC check not complete.	→	L_RcvEOF
2. CRC good.	→	L_GoodCRC
3. CRC bad.	→	L_BadEnd
4. PHYRDYn	→	L_NoCommErr [a]
[a] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		

LR7: L_GoodCRC [a]	Transmit R_IP_P.	
1. Transport layer indicated good result.	→	L_GoodEnd
2. Transport layer indicates unrecognized FIS.	→	L_BadEnd
3. Transport layer has yet to respond.	→	L_GoodCRC
4. PHYRDYn	→	L_NoCommErr [b]
5. Transport or Link layer indicated error detected during reception of recognized FIS.	→	L_BadEnd
6. $SYNC_P$ received from Phy.	→	L_IDLE
[a] Upon entering this state for the first time, the Link layer shall notify the Transport layer that the CRC for this frame is valid. [b] The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		

LR8: L_GoodEnd	Transmit R_OK_P.	
1. $SYNC_P$ received from Phy.	→	L_IDLE
2. AnyDword other than $SYNC_P$ received from Phy.	→	L_GoodEnd
3. PHYRDYn	→	L_NoCommErr

Figure 272 – Link layer receiver state machine (part 3 of 4)

LR9: L_BadEnd	Transmit R_ERR$_P$.	
1. SYNC$_P$ received from Phy.	→	L_IDLE
2. AnyDword other than SYNC$_P$ received from Phy.	→	L_BadEnd
3. PHYRDYn	→	L_NoCommErr

Figure 272 – Link layer receiver state machine (part 4 of 4)

LR1: L_RcvChkRdy state, this state is entered if X_RDY$_P$ has been received from the Phy layer.

If in this state, the Link layer shall transmit R_RDY$_P$ and wait for SOF$_P$ from the Phy layer.

In the case that a device has not sent a Signature FIS (e.g., prior to completion of spinup) and receives X_RDY$_P$ from the host, a device may hold off responding with R_RDY$_P$ until its Application layer is in a state that is able to process a new command/control FIS from the host controller including SRST. In some cases, this may be held off until after transmission of the Signature FIS. A device is required to respond to the host's X_RDY$_p$ transmission, but the response may be deferred as mentioned above.

Transition LR1:1, if the Link layer receives X_RDY$_P$ from the Phy layer, the Link layer shall make a transition to the LR1: L_RcvChkRdy state.

Transition LR1:2, if the Link layer receives SOF$_P$ from the Phy layer, the Link layer shall make a transition to the LR3: L_RcvData state.

Transition LR1:3, if the Link layer receives any Dword other than an X_RDY$_P$ or SOF$_P$ primitive from the Phy layer, the Link layer shall notify the Transport layer of the condition and make a transition to the L1: L_IDLE state.

Transition LR1:4, if the Link layer detects that the Phy layer is not ready the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

LR2: L_RcvWaitFifo state, this state is entered if an X_RDY$_P$ has been received, and the FIFO is not ready to receive a FIS.

If in this state, the Link layer shall transmit SYNC$_P$.

Transition LR2:1, if the Link layer receives X_RDY$_P$ from the Phy layer and the FIFO is ready to accept data, the Link layer shall make a transition to the LR1: L_RcvChkRdy state.

Transition LR2:2, if the Link layer receives X_RDY$_P$ from the Phy layer and the FIFO is not ready to accept data, the Link layer shall make a transition to the LR2: L_RcvWaitFifo state.

Transition LR2:3, if the Link layer receives any Dword other than X_RDY$_P$ from the Phy layer, the Link layer shall notify the Transport layer of the condition and make a transition to the L1: L_IDLE state.

Transition LR2:4, if the Link layer detects that the Phy layer is not ready, the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

LR3: L_RcvData state, this state is entered if SOF$_P$ has been received from the Phy layer.

If in this state, the Link layer receives an encoded character sequence from the Phy layer, decodes it into a Dword, and passes the Dword to the Transport layer. The Dword is also entered into the CRC calculation. If in this state, the Link layer either transmits R_IP$_P$ to signal transmission to continue or transmits DMAT$_P$ to signal the transmitter to terminate the transmission.

Transition LR3:1, if the Transport layer indicates that space is available in its FIFO, the Link layer shall make a transition to the LR3: L_RcvData state.

Transition LR3:2, if the Transport layer indicates that sufficient space is not available in its FIFO, the Link layer shall make a transition to the LR4: L_Hold state.

Transition LR3:3, if the Link layer receives HOLD$_P$ from the Phy layer, the Link layer shall make a transition to the LR5: L_RcvHold state.

Transition LR3:4, if the Link layer receives EOF$_P$ from the Phy layer, the Link layer shall make a transition to the LR6: L_RcvEOF state.

Transition LR3:5, if the Link layer receives WTRM$_P$ from the Phy layer, the Link layer shall make a transition to the LR9: L_BadEnd state.

Transition LR3:6, if the Link layer receives SYNC$_P$ from the Phy layer, the Link layer shall notify the Transport layer that reception was aborted and shall make a transition to the L1: L_IDLE state.

Transition LR3:7, if the Link layer receives any Dword other than a HOLD$_P$, HOLDA$_P$, EOF$_P$, or SYNC$_P$ primitive from the Phy layer, the Link layer shall make a transition to the LR3: L_RcvData state.

Transition LR3:8, if the Link layer detects that the Phy layer is not ready, the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

Transition LR3:9, if the Link layer receives notification from the Transport layer that the current frame should be escaped, a transition to the L_SyncEscape state shall be made.

LR4: L_Hold state, this state is entered if the Transport layer indicates that sufficient space is not available in its receive FIFO.

If in this state, the Link layer shall transmit HOLD$_P$ and may receive an encoded character from the Phy layer.

Transition LR4:1, if the Link layer receives any Dword other than a HOLD$_P$ primitive from the Phy layer and the Transport layer indicates that sufficient space is now available in its receive FIFO, the Link layer shall make a transition to the LR3: L_RcvData state.

Transition LR4:2, if the Link layer receives HOLD$_P$ from the Phy layer and the Transport layer indicates that space is now available in its FIFO, the Link layer shall make a transition to the LR5: L_RcvHold state.

Transition LR4:3, if the Link layer receives EOF$_P$ from the Phy layer, the Link layer shall make a transition to the LR6: L_RcvEOF state.

> NOTE 42 - Note that due to pipeline latency, an EOF$_P$ may be received while in the L_Hold state that case the receiving Link to use its FIFO headroom to receive the EOF$_P$ and close the frame reception.

Transition LR4:4, if the Transport layer indicates that there is not sufficient space available in its FIFO and the Phy layer is ready, the Link layer shall make a transition to the LR4: L_Hold state.

Transition LR4:5, if the Link layer detects that the Phy layer is not ready, the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

Transition LR4:6, if the Link layer receives $SYNC_P$ from the Phy layer, the Link layer shall notify the Transport layer of the illegal transition error condition and shall make a transition to the L1: L_IDLE state.

Transition LR4:7, if the Link layer receives notification from the Transport layer that the current frame should be escaped, a transition to the L_SyncEscape state shall be made.

LR5: L_RcvHold state, this state is entered if $HOLD_P$ has been received from the Phy layer.

If in this state, the Link layer shall either transmit $HOLDA_P$ to signal transmission to proceed if the transmitter becomes ready or transmit $DMAT_P$ to signal the transmitter to terminate the transmission.

Transition LR5:1, if the Link layer receives any Dword other than a $HOLD_P$ or $SYNC_P$ primitive from the Phy layer, the Link layer shall make a transition to the LR3: L_RcvData state.

Transition LR5:2, if the Link layer receives $HOLD_P$ from the Phy layer, the Link layer shall make a transition to the LR5: L_RcvHold state.

Transition LR5:3, if the Link layer receives EOF_P from the Phy layer, the Link layer shall make a transition to the LR6: L_RcvEOF state.

Transition LR5:4, if the Link layer receives $SYNC_P$ from the Phy layer, the Link layer shall make a transition to the L1: L_IDLE state. The Transport layer shall be notified of the illegal transition error condition.

Transition LR5:5, if the Link layer detects that the Phy layer is not ready the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

Transition LR5:6, if the Link layer receives notification from the Transport layer that the current frame should be escaped, a transition to the L_SyncEscape state shall be made.

LR6: L_RcvEOF state, this state is entered if the Link layer has received EOF_P from the Phy layer.

If in this state, the Link layer shall check the calculated CRC for the frame and transmit one or more R_IP_P primitives.

Transition LR6:1, if the CRC calculation and check is not yet completed, the Link layer shall make a transition to the LR6: L_RcvEOF state.

Transition LR6:2, if the CRC indicates no error, the Link layer shall notify the Transport layer and make a transition to the LR7: L_GoodCRC state.

Transition LR6:3, if the CRC indicates an error has occurred, the Link layer shall notify the Transport layer and make a transition to the LR9: L_BadEnd state.

Transition LR6:4, if the Link layer detects that the Phy layer is not ready, the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

LR7: L_GoodCRC state, this state is entered if the CRC for the frame has been checked and determined to be good.

If in this state, the Link layer shall wait for the Transport layer to check the frame and transmit one or more R_IP$_P$ primitives.

Transition LR7:1, if the Transport layer indicates a good result, the Link layer shall transition to the LR8: L_GoodEnd state.

Transition LR7:2, if the Transport layer indicates an unrecognized FIS, the Link layer shall transition to the LR9: L_BadEnd state.

Transition LR7:3, if the Transport layer has not supplied status, then the Link layer shall transition to the LR7: L_GoodCRC state.

Transition LR7:4, if the Link layer detects that the Phy layer is not ready, the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

Transition LR7:5, if the Transport layer or Link layer indicates an error was encountered during the reception of the recognized FIS, the Link layer shall transition to the LR9: L_BadEnd state.

Transition LR7:6, if the Link layer receives SYNC$_P$ from the Phy layer, the Link layer shall notify the Transport layer of the illegal transition error condition and shall make a transition to the L1: L_IDLE state.

LR8: L_GoodEnd state, this state is entered if the CRC for the frame has been checked and determined to be good.

If in this state, the Link layer shall transmit R_OK$_P$.

Transition LR8:1, if the Link layer receives SYNC$_P$ from the Phy layer, the Link layer shall make a transition to the L1: L_IDLE state.

Transition LR8:2, if the Link layer receives any Dword other than a SYNC$_P$ primitive from the Phy layer, the Link layer shall make a transition to the LR7: L_GoodEnd state.

Transition LR8:3, if the Link layer detects that the Phy layer is not ready, the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

LR9: L_BadEnd state, this state is entered if the CRC for the frame has been checked and determined to be bad or if the Transport layer has notified the Link layer that the received FIS is invalid.

If in this state, the Link layer shall transmit R_ERR$_P$.

Transition LR9:1, if the Link layer receives SYNC$_P$ from the Phy layer, the Link layer shall make a transition to the L1: L_IDLE state.

Transition LR9:2, if the Link layer receives any Dword other than SYNC$_P$ from the Phy layer, the Link layer shall make a transition to the LR9: BadEnd state.

Transition LR9:3, if the Link layer detects that the Phy layer is not ready, the Link layer shall notify the Transport layer of the condition and make a transition to the LS1: L_NoCommErr state.

9.7.5 Link power mode state machine

The link power mode state machine is defined in Figure 273.

LPM1: L_TPMPartial	Transmit PMREQ_P_P.	
1. $PMACK_P$ received from Phy layer.	→	L_ChkPhyRdy
2. X_RDY_P received from Phy layer.	→	L_RcvWaitFifo [a]
3. $SYNC_P$ or R_OK_P received from Phy layer.	→	L_TPMPartial
4. AnyDword other than ($PMACK_P$, $PMNAK_P$, X_RDY_P, $SYNC_P$, R_OK_P, PMREQ_P_P [c], or PMREQ_S_P [c]) [a] received from Phy layer.	→	L_IDLE
5. PMREQ_P_P or PMREQ_S_P received from Phy layer.	→	L_TPMPartial [c]
6. PHYRDYn	→	L_NoCommErr [b]
7. $PMNAK_P$ received from Phy layer.	→	L_NoPmnak

[a] This transition aborts the request from the Transport layer to enter a power mode. A status indication to the Transport layer of this event is required.
[b] This is an unexpected transition and constitutes an error condition. An error condition needs to be sent to the Transport layer as a result.
[c] If PMREQ_P_P or PMREQ_S_P is received, the host shall make a transition to the L_IDLE state, but the device shall make a transition to the L_TPMPartial state.

LPM2: L_TPMSlumber	Transmit PMREQ_S_P.	
1. $PMACK_P$ received from Phy layer.	→	L_ChkPhyRdy
2. X_RDY_P received from Phy layer.	→	L_RcvWaitFifo [a]
3. $SYNC_P$ or R_OK_P received from Phy layer.	→	L_TPMSlumber
4. AnyDword other than ($PMACK_P$, $PMNAK_P$, X_RDY_P, $SYNC_P$, R_OK_P, PMREQ_P_P [c], or PMREQ_S_P [c]) [a] received from Phy layer.	→	L_IDLE
5. PMREQ_P_P or PMREQ_S_P received from Phy layer.	→	L_TPMSlumber [c]
6. PHYRDYn	→	L_NoCommErr [b]
7. $PMNAK_P$ received from Phy layer.	→	L_NoPmnak

[a] This transition aborts the request from the Transport layer to enter a power mode. A status indication to the Transport layer of this event is required.
[b] This is an unexpected transition and constitutes an error condition. An error condition needs to be sent to the Transport layer as a result.
[c] If PMREQ_P_P or PMREQ_S_P is received, the host shall make a transition to the L_IDLE state, but the device shall make a transition to the L_TPMSlumber state.

Figure 273 – Link layer power mode state machine (part 1 of 2)

LPM3: L_PMOff	Transmit PMACK$_P$ [a].		
	1. A total of 4 ≤ n ≤ 16 PMACK$_P$ primitives sent.	→	L_ChkPhyRdy
	2. Less than n PMACK$_P$ primitives sent.	→	L_PMOff
	[a] A flag is set according to whether a PMREQ_P$_P$ or PMREQ_S$_P$ was received from the Phy layer.		

LPM4: L_PMDeny	Transmit PMNAK$_P$.		
	1. PMREQ_P$_P$ or PMREQ_S$_P$ received from Phy layer.	→	L_PMDeny
	2. AnyDword other than (PMREQ_P$_P$ or PMREQ_S$_P$) received from Phy layer.	→	L_IDLE
	3. PHYRDYn	→	L_NoCommErr

LPM5: L_ChkPhyRdy	Assert Partial/Slumber to Phy layer (as appropriate).		
	1. PHYRDY	→	L_ChkPhyRdy
	2. PHYRDYn	→	L_NoCommPower

LPM6: L_NoCommPower	Maintain Partial/Slumber assertion (as appropriate).		
	1. Transport layer requests a wakeup or COMWAKE detected.	→	L_WakeUp1
	2. Transport layer not requesting wakeup and COMWAKE not detected.	→	L_NoCommPower

LPM7: L_WakeUp1	Negate both Partial and Slumber.		
	1. PHYRDY	→	L_WakeUp2
	2. PHYRDYn	→	L_WakeUp1

LPM8: L_WakeUp2	Transmit ALIGN$_P$.		
	1. PHYRDY	→	L_IDLE
	2. PHYRDYn	→	L_NoCommErr

LPM9: L_NoPmnak	Transmit SYNCP.		
	1. PMNAK$_P$ received from Phy layer.	→	L_NoPmnak
	2. AnyDword other than (PMNAK$_P$) received from Phy layer.	→	L_IDLE

Figure 273 – Link layer power mode state machine (part 2 of 2)

LPM1: L_TPMPartial state, this state is entered if the Transport layer has indicated that a transition to the Partial power state is desired.

Transition LPM1:1, if in this state, PMREQ_P$_P$ shall be transmitted. If the Link layer receives PMACK$_P$ a transition to the LPM5: L_ChkPhyRdy state shall be made.

Transition LPM1:2, if the Link layer receives X_RDY$_P$, a transition shall be made to the LR2: L_RcvWaitFifo state, effectively aborting the request to a power mode state.

Transition LPM1:3, if the Link layer receives a SYNC$_P$ or R_OK$_P$ primitive, then it is assumed that the opposite side has not processed PMREQ_P$_P$ and time is needed. A transition to the LPM1: L_TPMPartial state shall be made.

Transition LPM1:4, if the host Link layer receives any Dword from the Phy layer other than a PMACK$_P$, PMNAK$_P$, X_RDY$_P$, SYNC$_P$, or R_OK$_P$ primitive, then the request to enter the Partial state is aborted and a transition to L1: L_IDLE shall be made. If the device Link layer receives any Dword from the Phy layer other than a PMACK$_P$, PMNAK$_P$, X_RDY$_P$, SYNC$_P$, PMREQ_P$_P$, PMREQ_S$_P$, or R_OK$_P$ primitive, then the request to enter the Partial state is aborted and a transition to L1: L_IDLE shall be made.

Transition LPM1:5, the host Link layer shall not make this transition as it applies only to the device Link layer. If the device Link layer receives PMREQ_P$_P$ or PMREQ_S$_P$ from the host, it shall remain in this state by transitioning back to LPM1: L_TPMPartial.

Transition LPM1:6, if the Link layer detects that the Phy layer has become not ready, this is interpreted as an error condition. The Transport layer shall be notified of the condition and a transition shall be made to the LS1: L_NoCommErr state.

Transition LPM1:7, if the Link layer receives a PMNAK$_P$, then the request to enter the Partial state is aborted and a transition to LPM9: L_NoPmnak shall be made.

LPM2: L_TPMSlumber state, this state is entered if the Transport layer has indicated that a transition to the Slumber power state is desired.

Transition LPM2:1, if in this state, PMREQ_S$_P$ shall be transmitted. If the Link layer receives PMACK$_P$, a transition to the LPM5: L_ChkPhyRdy state shall be made.

Transition LPM2:2, if the Link layer receives X_RDY$_P$, a transition to the LR2: L_RcvWaitFifo state shall be made, effectively aborting the request to a power mode state.

Transition LPM2:3, if the Link layer receives SYNC$_P$ or R_OK$_P$, then it is assumed that the opposite side has not processed PMREQ_S$_P$ and time is needed. The transition to the LPM2: L_TPMSlumber state shall be made.

Transition LPM2:4, if the host Link layer receives any Dword from the Phy layer other than a:

a) PMACK$_P$;
b) PMNAK$_P$;
c) X_RDY$_P$;
d) SYNC$_P$; or
e) R_OK$_P$ primitive,

then the request to enter the Slumber state is aborted and a transition to L1: L_IDLE shall be made.

If the device Link layer receives any Dword from the Phy layer other than a:

a) PMACK$_P$;
b) PMNAK$_P$;
c) X_RDY$_P$;
d) SYNC$_P$;
e) PMREQ_P$_P$;
f) PMREQ_S$_P$; or
g) R_OK$_P$ primitive,

then the request to enter the Slumber state is aborted and a transition to L1: L_IDLE shall be made.

Transition LPM2:5, the host Link layer shall not make this transition as it applies only to the device Link layer. If the device Link layer receives PMREQ_P_P or PMREQ_S_P from the host, it shall remain in this state by transitioning back to LPM2: L_TPMSlumber.

Transition LPM2:6, if the Link layer detects that the Phy layer has become not ready, this is interpreted as an error condition. The Transport layer shall be notified of the condition and a transition shall be made to the L_NoCommErr state.

Transition LPM2:7, if the Link layer receives a $PMNAK_P$, then the request to enter the Slumber state is aborted and a transition to LPM9: L_NoPmnak shall be made.

LPM3: L_PMOff state, this state is entered if either PMREQ_S_P or PMREQ_P_P was received by the Link layer. The Link layer transmits $PMACK_P$ for each processing of this state.

Transition LPM3:1, if $4 \leq n \leq 16$ $PMACK_P$ primitives have been transmitted, a transition shall be made to the L_ChkPhyRdy state.

Transition LPM3:2, if less than n $PMACK_P$ primitives have been transmitted, a transition shall be made to L_PMOff state.

LPM4: L_PMDeny state, this state is entered if any primitive is received by the Link layer to enter a power mode and power modes are currently disabled. The Link layer shall transmit $PMNAK_P$ to inform the opposite end that a power mode is not allowed.

Transition LPM4:1, if the Link layer continues to receive a request to enter any power mode, then a transition back to the same LPM4: L_PMDeny state shall be made.

Transition LPM4:2, if the Link layer receives any Dword other than a power mode request primitive, then the Link layer assumes that the power mode request has been removed and shall make a transition to the L1: L_IDLE state.

Transition LPM4:3, if the Link layer detects that the Phy layer has become not ready, this is interpreted as an error condition. The Transport layer shall be notified of the condition and a transition shall be made to the LS1: L_NoCommErr state.

LPM5: L_ChkPhyRdy state, this state is entered if it is desired for the Phy layer to enter a low power condition. For each processing in this state a request is made to the Phy layer to enter the state and deactivate the PHYRDY signal. Partial or Slumber is asserted to the Phy layer as appropriate.

Transition LPM6:1, if the Phy layer has not yet processed the request to enter the power saving state and not deactivated the PHYRDY signal, then the Link layer shall remain in the LPM5: L_ChkPhyRdy state and continue to request the Phy layer to enter the power mode state.

Transition LPM6:2, if the Phy layer has processed the power mode request and has deactivated the PHYRDY signal, then a transition shall be made to the LPM6: L_NoCommPower state.

LPM6: L_NoCommPower state, this state is entered if the Phy layer has negated its PHYRDY signal indicating that it is in either Partial or Slumber state. In this state, the Link layer waits for the OOB detector to signal reception of the COMWAKE signal (for a wakeup initiated by the other device), or for the Transport layer to request a wakeup.

Transition LPM6:1, if the Transport layer requests a wakeup or the OOB signal detector indicates reception of the COMWAKE signal, then a transition shall be made to LPM7: L_WakeUp1.

Transition LPM6:2, if the Transport layer does not request a wakeup and the OOB detector does not indicate reception of the COMWAKE signal, then a transition shall be made to LPM6: L_NoCommPower.

LPM7: L_WakeUp1 state, this state is entered if the Transport layer has initiated a wakeup. In this state, the Link layer shall negate both Partial and Slumber to the Phy layer, and wait for the PHYRDY signal from the Phy layer to be asserted. While in this state the Phy layer is performing the wakeup sequence.

Transition LPM7:1, if the Phy layer asserts its PHYRDY signal, a transition shall be made to LPM8: L_WakeUp2.

Transition LPM7:2, if the Phy layer remains not ready, a transition shall be made to LPM7: L_WakeUp1.

LPM8: L_WakeUp2 state, this state is entered if the Phy layer has acknowledged an initiated wakeup request by asserting its PHYRDY signal. In this state, the Link layer shall transmit the $ALIGN_P$ sequence, and transition to the L1: L_IDLE state.

Transition LPM8:1, if the Phy layer keeps PHYRDY asserted, a transition shall be made to the L1: L_IDLE state.

Transition LPM8:2, if the Phy layer negates PHYRDY, this is an error condition. The Transport layer shall be notified of the condition and a transition shall be made to the LS1: L_NoCommErr state.

LPM9: L_NoPmnak state, this state is entered if the Link layer has indicated that a request to enter the Slumber or Partial state has been denied. The Link layer transmits $SYNC_P$ for each processing of this state. In this state, the Link layer waits for receipt of any Dword that is not $PMNAK_P$ from the Phy layer.

Transition LPM9:1, if the Link layer receives $PMNAK_P$, then the Link layer shall remain in the LPM9: L_NoPmnak state and continue to wait for receipt of a primitive that is not $PMNAK_P$ from the Phy layer.

Transition LPM9:2, if the Link layer receives any Dword from the Phy layer other than $PMNAK_P$, then the request to enter the power management state is aborted and a transition to L1: L_IDLE shall be made.

10 Transport layer

10.1 Transport layer overview

The Transport layer need not be cognizant of how frames are transmitted and received. The Transport layer simply constructs Frame Information Structures (FISes) for transmission and decomposes received FISes. Host and device Transport layer state differ in that the source of the FIS content differs. The Transport layer maintains no context in terms of ATA commands or previous FIS content.

10.2 FIS construction

If requested to construct a FIS by a higher layer, the Transport layer provides the following services:

a) gathers FIS content based on the type of FIS requested;
b) places FIS content in the proper order;
c) notifies the Link layer of required frame transmission and passes FIS content to Link;
d) manages buffer/FIFO flow, notifies Link of required flow control;
e) receives frame receipt acknowledge from Link layer; and
f) reports good transmission or errors to requesting higher layer.

10.3 FIS decomposition

If a FIS is received from the Link layer, the Transport layer provides the following services:

a) receives the FIS from the Link layer;
b) determines FIS Type;
c) distributes the FIS content to the locations indicated by the FIS Type;
d) for the host Transport layer, receipt of a FIS may also cause the construction of a FIS to be returned to the device; and
e) reports good reception or errors to higher layer.

10.4 Frame information structure (FIS)

10.4.1 Frame information structure (FIS) overview

A FIS is a group of Dwords that convey information between host and device as described previously. Primitives are used to define the boundaries of the FIS and may be inserted to control the rate of the information flow. This section describes the information content of the FIS - referred to as payload - and assumes the reader is aware of the primitives that are needed to support the information content.

The contents of the info field is divided into three categories:

a) register type;
b) setup type; and
c) data type.

For each category the organization of each frame is defined in the following section.

10.4.2 Payload content

The type and layout of the payload is indicated by the FIS Type field located in byte 0 of the first Dword of the payload. See Figure 275 as an example. This example type is used primarily to transfer the contents of the Shadow Register Block Registers from the host to the device. Table 99

may be referenced to refresh the reader's memory of a simplified version of the Shadow Register Block organization of an ATA adapter.

Table 99 – Simplified Shadow Register Block register numbering

				Register access operation	
	A2	**A1**	**A0**	**Read**	**Write**
CS 0 Active	0	0	0	Data Port	
	0	0	1	Error	Features
	0	1	0	Sector Count [15:8], [7:0]	
	0	1	1	LBA [31:24], [7:0]	
	1	0	0	LBA [39:32], [15:8]	
	1	0	1	LBA [47:40], [23:16]	
	1	1	0	Device / Head	
	1	1	1	Status	Command
CS 1 Active	1	1	0	Alternate Status	Device Control

The following sections detail the types of payloads that are possible. The SOF_P, EOF_P, and $HOLD_P$ primitives have been removed for clarity.

10.5 FIS Types

10.5.1 FIS Types scope

The following sections define the structure of each individual FIS.

10.5.2 FIS Type values

10.5.2.1 FIS Type values overview

The value for the FIS Type fields of all FISes has been selected to provide additional robustness. In minimally buffered implementations that may not buffer a complete FIS, the state machines may begin acting on the received FIS Type value prior to the ending CRC having been checked. Because the FIS Type value may be acted upon prior to the integrity of the complete FIS being checked against its ending CRC, the FIS Type field values have been selected to maximize the Hamming distance between them.

Figure 274 defines the FIS Type values and their assignments.

FIS Type field value	Description
27h	Register Host to Device FIS
34h	Register Device to Host FIS
39h	DMA Activate FIS – Device to Host
41h	DMA Setup FIS – Bi-directional
46h	Data FIS – Bi-directional
58h	BIST Activate FIS – Bi-directional
5Fh	PIO Setup FIS – Device to Host
A1h	Set Device Bits FIS – Device to Host
A6h	Reserved for future Serial ATA definition
B8h	Reserved for future Serial ATA definition
BFh	Reserved for future Serial ATA definition
C7h	Vendor specific
D4h	Vendor specific
D9h	Reserved for future Serial ATA definition

Figure 274 – FIS Type value assignments

10.5.2.2 Unrecognized FIS Types

A device or host may receive a FIS Type that is not defined or vendor specific in Figure 274. The receiver of a FIS determines whether to handle a FIS Type that is not defined or reserved as "unrecognized". There may be cases where a receiver accepts undefined or vendor specific FISes. The host and device should negotiate any undefined or vendor specific FIS Types that may be transmitted prior to their use.

If the receiver decides to treat a FIS Type as unrecognized, it shall follow the Link layer state machine as defined in 9.7 upon receipt of that FIS Type.

10.5.3 CRC errors on data FISes

Following a Serial ATA CRC error on a Data FIS, if the device transmits a Register Device to Host FIS it shall set the ERR bit to one and both the BSY bit and the DRQ bit cleared to zero in the Status field, and the ABRT bit set to one in the ERROR field. It is recommended for the device to also set the bit 7 (i.e., the ICRC bit) to one in the ERROR field. See ACS-4.

There is no Register Device to Host FIS transmitted after a Serial ATA CRC error on the last Data FIS of a PIO-in command nor following a Serial ATA CRC error on the Advanced Technology Attachment Packet Interface (ATAPI) command packet transfer. Thus, there is no mechanism for the device to indicate a Serial ATA CRC error to the host in either of these cases. The host should check the SError register to determine if a Link layer error has occurred in both of these cases.

10.5.4 All FIS Types

In all of the following FIS structures the following rules shall apply, all reserved fields shall be:

a) written or transmitted as all zeroes; and
b) ignored during the reading or reception process.

10.5.5 Register Host to Device FIS

10.5.5.1 Register Host to Device FIS layout

0	Features(7:0)								Command(7:0)								C	R	R	R	PM Port				FIS Type (27h)			
1	Device(7:0)								LBA(23:16)								LBA(15:8)								LBA(7:0)			
2	Features(15:8)								LBA(47:40)								LBA(39:32)								LBA(31:24)			
3	Control(7:0)								ICC(7:0)								Count(15:8)								Count(7:0)			
4	Auxiliary(31:24)								Auxiliary(23:16)								Auxiliary(15:8)								Auxiliary(7:0)			

Figure 275 – Register Host to Device FIS layout

If a field in this FIS is not defined by a command, it shall be Reserved for that command.

Field Definitions

Features(7:0)
: Contains the contents of the Features register of the Shadow Register Block.

Command
: Contains the contents of the Command register of the Shadow Register Block.

C
: This bit is set to one if the register transfer is due to an update of the Command register. The bit is cleared to zero if the register transfer is due to an update of the Device Control register. Setting C bit to one and SRST bit to one in the Device Control field is invalid and results in indeterminate behavior.

R
: Reserved, shall be cleared to zero.

PM Port
: If an endpoint device is attached via a Port Multiplier, specifies the device port address that the FIS should be delivered to. This field is set by the host.

FIS Type
: Set to a value of 27h. Defines the rest of the FIS fields. Defines the length of the FIS as five Dwords.

Device
: Contains the contents of the Device register of the Shadow Register Block.

LBA(23:16)
: Contains the contents of the LBA high register of the Shadow Register Block.

LBA(15:8)
: Contains the contents of the LBA mid register of the Shadow Register Block.

LBA(7:0)
: Contains the contents of the LBA low register of the Shadow Register Block.

Features(15:8)
: Contains the contents of the expanded address field of the Shadow Register Block.

LBA(47:40)
: Contains the contents of the expanded address field of the Shadow Register Block.

LBA(39:32)
: Contains the contents of the expanded address field of the Shadow Register Block.

LBA(31:24)
: Contains the contents of the expanded address field of the Shadow Register Block.

Control
: Contains the contents of the Device Control register of the Shadow Register Block.

ICC	The Isochronous Command Completion (ICC) field contains a value set by the host to inform device of a time limit. If a command does not define the use of this field, it shall be reserved.
Count(15:8)	Contains the contents of the expanded address field of the Shadow Register Block.
Count(7:0)	Contains the contents of the Sector Count register of the Shadow Register Block.
Auxiliary	Contains parameter values specified on a per command basis.

10.5.5.2 Description

The Register Host to Device FIS is used to transfer the contents of the Shadow Register Block from the host to the device. This is the mechanism for issuing ATA commands to the device.

10.5.5.3 Transmission

Transmission of a Register Host to Device FIS is initiated by a write operation to either the command register, or a write to the Device Control register with a value different than is currently in the Device Control register in the host adapter's Shadow Register Block. Upon initiating transmission, the current contents of the Shadow Register Block are transmitted and the C bit in the FIS is set according to whether the transmission was a result of the Command register being written or the Device Control register being written. The host adapter shall set the BSY bit in the shadow Status register to one within 400 ns of the write operation to the Command register that initiated the transmission. The host adapter shall set the BSY bit in the shadow Status register to one within 400 ns of a write operation to the Device Control register if the write to the Device Control register changes the state of the SRST bit from zero to one. The host adapter shall not set the BSY bit in the shadow Status register for writes to the Device Control register that do not change the state of the SRST bit from zero to one.

> NOTE 43 - It is important to note that Serial ATA host adapters enforce the same access control to the Shadow Register Block as parallel ATA devices enforce to the Command Block Registers.

Specifically, the host shall not write the FEATURES field (7:0), COUNT field (7:0), LBA field (7:0), LBA field (15:8), LBA field (23:16), or Device registers if either the BSY bit or the DRQ bit is set to one in the Status register. Any write to the Command register if the BSY bit or the DRQ bit is set is ignored unless the write is to issue a DEVICE RESET command.

10.5.5.4 Reception

Upon reception of a valid Register Host to Device FIS the device updates its local copy of the Command and Control Block Register contents. Then the device either initiates processing of the command indicated in the Command register or initiates processing of the control request indicated in the Device Control register, depending on the state of the C bit in the FIS.

There are legacy BIOS and drivers that write the Device Control register to enable the interrupt just prior to issuing a command. To avoid unnecessary overhead, this FIS is transmitted to the device only upon a change of state from the previous value.

10.5.6 Register Device to Host FIS

10.5.6.1 Register Device to Host FIS Layout

0	Error(7:0)								Status(7:0)								R	I	R	R	PM Port				FIS Type (34h)			
1	Device(7:0)								LBA(23:16)								LBA(15:8)								LBA(7:0)			
2	Reserved								LBA(47:40)								LBA(39:32)								LBA(31:24)			
3	Reserved								Reserved								Count(15:8)								Count(7:0)			
4	Reserved								Reserved								Reserved								Reserved			

Figure 276 – Register Device to Host FIS layout

Field Definitions

Error — Contains the new value of the Error register of the Shadow Register Block.

Status — Contains the new value of the Status (and Alternate status) register of the Shadow Register Block.

R — Reserved, shall be cleared to zero.

I — Interrupt (I) bit, this bit reflects the interrupt bit line of the device. Devices shall not modify the behavior of this bit based on the state of the nIEN bit received in Register Host to Device FISes.

PM Port — If an endpoint device is attached via a Port Multiplier, specifies the device port address that the FIS is received from. This field is set by the Port Multiplier. Endpoint devices shall clear this field to 0h.

FIS Type — Set to a value of 34h. Defines the rest of the FIS fields. Defines the length of the FIS as five Dwords.

Device — Contains the new value of the Device register of the Shadow Register Block.

LBA(23:16) — Contains the new value of the LBA high register of the Shadow Register Block.

LBA(15:8) — Contains the new value of the LBA mid register of the Shadow Register Block.

LBA(7:0) — Contains the new value of the LBA low register of the Shadow Register Block.

LBA(47:40) — Contains the contents of the expanded address field of the Shadow Register Block.

LBA(39:32) — Contains the contents of the expanded address field of the Shadow Register Block.

LBA(31:24) — Contains the contents of the expanded address field of the Shadow Register Block.

Count(15:8) — Contains the contents of the expanded address field of the Shadow Register Block.

Count(7:0) — Contains the new value of the Sector Count register of the Shadow Register Block.

10.5.6.2 Description

The Register Device to Host FIS (see Figure 276) is used to by the device to update the contents of the host adapter's Shadow Register Block. This is the mechanism that devices indicate command completion status or otherwise change the contents of the host adapter's Shadow Register Block.

10.5.6.3 Transmission

Transmission of a Register Device to Host FIS is initiated by the device in order to update the contents of the host adapter's Shadow Register Block. Transmission of the Register Device to Host FIS is typically as a result of command completion by the device.

The Register Device to Host FIS shall only be used to set the SERV bit to one in the Status register to request service for a bus released command if the BSY bit or the DRQ bit is currently set to one in the Status register; the Set Device Bits FIS shall be used to set to one the SERV bit if the BSY bit and the DRQ bit are both cleared to zero in the Status register. The SERV bit transmitted with the Register Device to Host FIS is written to the shadow Status register and so the bit should accurately reflect the state of pending service requests if the Register Device to Host FIS is transmitted as a result of a command completion by the device.

10.5.6.4 Reception

Upon reception of a valid Register Device to Host FIS the received register contents are transferred to the host adapter's Shadow Register Block.

If the BSY bit and the DRQ bit in the shadow Status register are both cleared, and a Register Device to Host FIS is received by the host adapter, then the host adapter shall discard the contents of the received Register Device to Host FIS and not update the contents of any shadow register.

10.5.7 Set Device Bits - Device to Host FIS

10.5.7.1 Set Device Bits - Device to Host FIS layout

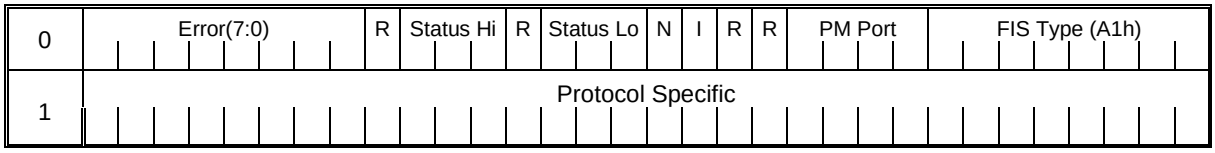


Figure 277 – Set Device Bits – Device to Host FIS layout

Field Definitions

Error — Contains the new value of the Error register of the Shadow Register Block.

R — Reserved, shall be cleared to zero.

Status Hi — Contains the new value of bits 6, 5, and 4 of the Status register of the Shadow Register Block.

Status-Lo — Contains the new value of bits 2, 1, and 0 of the Status register of the Shadow Register Block.

N — Notification (N) bit, this bit signals the host that the device needs attention. If the bit is set to one, the host should interrogate the device and determine what type of action is needed. If the bit is cleared to zero, the device is not requesting attention from the host (see 13.8.2).

I — Interrupt bit, this bit signals the host adapter to enter an interrupt pending state. If the host is processing tagged queued commands:

a) READ DMA QUEUED;
b) WRITE DMA QUEUED;
c) READ DMA QUEUED EXT;
d) WRITE DMA QUEUED EXT; or
e) WRITE DMA QUEUED FUA EXT,

with the device, the host should only enter the interrupt pending state if both the BSY bit and the DRQ bit in the shadow Status register are cleared to zero.
If the host is processing NCQ commands with the device, the interrupt pending state is entered regardless of the current state of the BSY bit or the DRQ bit in the shadow Status register. Devices shall not modify the behavior of this bit based on the state of the nIEN bit received in Register Host to Device FISes.

PM Port — If an endpoint device is attached via a Port Multiplier, specifies the device port address that the FIS is received from. This field is set by the Port Multiplier. Endpoint devices shall clear this field to 0h.

FIS Type — Set to a value of A1h. Defines the rest of the FIS fields. Defines the length of the FIS as two Dwords.

Protocol Specific
The value of this field is only defined for use with the NCQ Protocol. Refer to 13.6 for details. This field shall be cleared to zero for any uses other than NCQ (e.g., Asynchronous Notification).

10.5.7.2 Description

The Set Device Bits (SDB) FIS (see Figure 277) is used by the device to load Shadow Register Block bits that the device has exclusive write access. These bits are the eight bits of the Error register and six of the eight bits of the Status register. This FIS does not alter the BSY bit or the DRQ bit of the Status register.

The FIS includes a bit to signal the host adapter to generate an interrupt if the BSY bit and the DRQ bit in the shadow Status register are both cleared to zero.

Some Serial ATA to parallel ATA bridge solutions may elect to not support this FIS based on the requirements of their target markets.

10.5.7.3 Transmission

The device transmits a Set Device Bits FIS to alter one or more bits in the Error register or in the Status register in the Shadow Register Block. This FIS should be used by the device to set the SERV bit in the Status register to request service for a bus released command. If used for this purpose the device shall set the Interrupt bit to one.

10.5.7.4 Reception

Upon receiving a Set Device Bits FIS, the host adapter shall load the data from the ERROR field into the shadow Error register, the data from the Status-Hi field into bits 6, 5, and 4, of the shadow Status register, and the data from the Status-Lo field into bits 2, 1, and 0 of the shadow Status register. The BSY bit and the DRQ bit of the shadow Status register shall not be changed. If the Interrupt bit in the FIS is set to a one, and if both the BSY bit and the DRQ bit in the Shadow status register are cleared to zero, then the host adapter shall enter an interrupt pending state.

10.5.8 DMA Activate - Device to Host

10.5.8.1 DMA Activate - Device to Host Layout

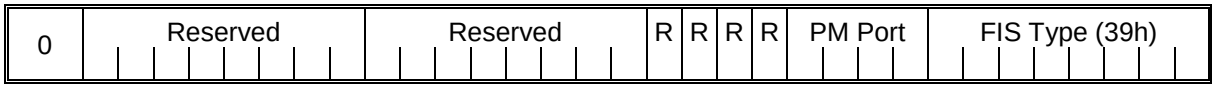


Figure 278 – DMA Activate – Device to Host FIS layout

Field Definitions

R	Reserved, shall be cleared to zero.
PM Port	If an endpoint device is attached via a Port Multiplier, specifies the device port address that the FIS is received from. This field is set by the Port Multiplier. Endpoint devices shall clear this field to 0h.
FIS Type	Set to a value of 39h. Defines the rest of the FIS fields. Defines the length of the FIS as one Dword.

10.5.8.2 Description

The DMA Activate FIS (see Figure 278) is used by the device to signal the host to proceed with a DMA data transfer of data from the host to the device.

A situation may arise where the host needs to send multiple Data FISes in order to complete the overall data transfer request. The host shall wait for a successful reception of a DMA Activate FIS before sending each of the Data FISes that are needed.

10.5.8.3 Transmission

The device transmits a DMA Activate to the host in order to initiate the flow of DMA data from the host to the device as part of the data transfer portion of a corresponding DMA write command. If transmitting this FIS, the device shall be prepared to subsequently receive a Data - Host to Device FIS from the host with the DMA data for the corresponding command.

10.5.8.4 Reception

Upon receiving a DMA Activate, if the host adapter's DMA controller has been programmed and armed, the host adapter shall initiate the transmission of a Data FIS and shall transmit in this FIS the data corresponding to the host memory regions indicated by the DMA controller's context. If the host adapter's DMA controller has not yet been programmed and armed, the host adapter shall set an internal state indicating that the DMA controller has been activated by the device, and as soon as the DMA controller has been programmed and armed, a Data FIS shall be transmitted to the device with the data corresponding to the host memory regions indicated by the DMA controller context.

10.5.9 DMA Setup – Device to Host FIS or Host to Device FIS (bidirectional)

10.5.9.1 DMA Setup – Device to Host FIS or Host to Device FIS (bidirectional) layout

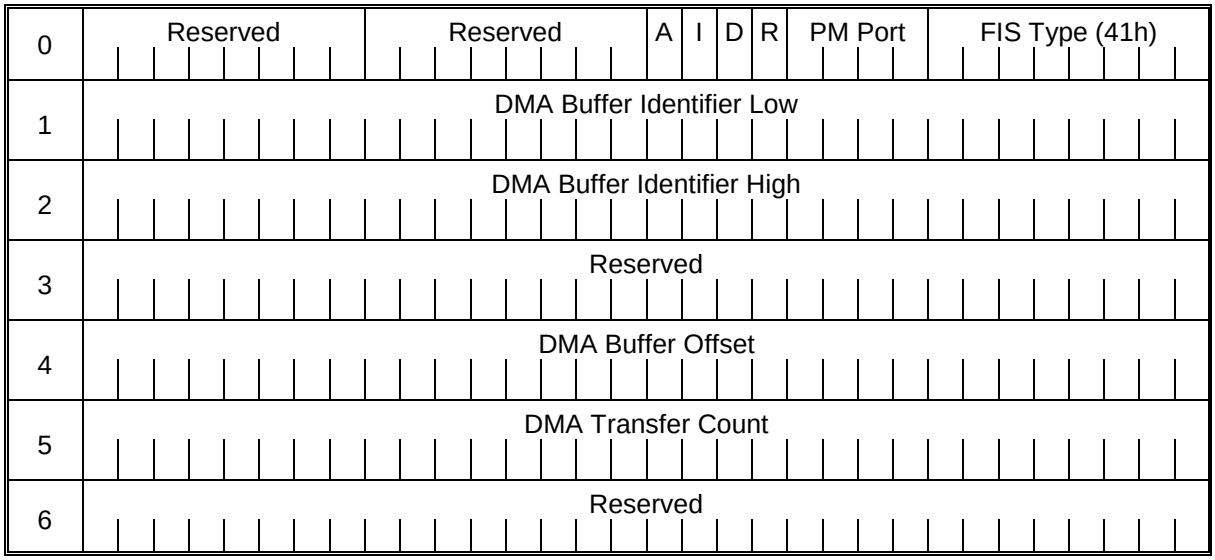


Figure 279 – DMA Setup – Device to Host or Host to Device FIS layout

Field Definitions

A — Auto-Activate (A) bit, if set to one, in response to a DMA Setup FIS with data transfer direction of Host-to-Device, causes the host to initiate transfer of the first Data FIS to the device after the DMA context for the transfer has been established. The device shall not transmit a DMA Activate FIS to trigger the transmission of the first Data FIS from the host. If cleared to zero, a DMA Activate FIS is required to trigger the transmission of the first Data FIS from the host if the data transfer direction is Host-to-Device.

I — Interrupt bit, if the Interrupt bit is set to one an interrupt pending shall be generated if the DMA transfer count is exhausted. Devices shall not modify the behavior of this bit based on the state of the nIEN bit received in Register Host to Device FISes.

D — Direction (D) bit, specifies whether subsequent data transferred after this FIS is from transmitter to receiver or from receiver to transmitter. If set to one, the direction is transmitter to receiver. If cleared to zero, the direction is receiver to transmitter.

R — Reserved, shall be cleared to zero.

PM Port — If an endpoint device is attached via a Port Multiplier, specifies the device port address that the FIS should be delivered to or is received from. This field is set by the host for host to device transmission and this field is set by the Port Multiplier for device to host transmission. Endpoint devices shall clear this field to 0h for device to host transmissions.

FIS Type — Set to a value of 41h. Defines the rest of the FIS fields. Defines the total length of the FIS as seven Dwords.

DMA Buffer Identifier Low/High
This field is used to identify a DMA buffer region in host memory. The contents are not described in this specification and are host dependent. The DMA Buffer Identifier is supplied by the host to the device and the device echoes it back to the host. This allows the implementation to pass a physical address or, in more

complex implementations, the DMA Buffer Identifier is able to be a scatter gather list or other information that may identify a DMA “channel”.

DMA Buffer Offset
This is the byte offset into the buffer. Bits (1:0) shall be cleared to zero.

DMA Transfer Count
This is the number of bytes to be read or written. Bit zero shall be cleared to zero.

10.5.9.2 Description

The DMA Setup – Device to Host or Host to Device FIS (see Figure 279) is the mechanism that first-party DMA access to host memory is initiated. This FIS is used to request the host or device to program its DMA controller before transferring data. The FIS allows the actual host memory regions to be abstracted (depending on implementation) by having memory regions referenced via a base memory descriptor representing a memory region that the host has granted the device access to. The specific implementation for the memory descriptor abstraction is not defined.

The device or host is informed of the 64 bit DMA Buffer Identifier/descriptor at some previous time by an implementation specific mechanism (e.g., a command issued to or as defined in a specification). Random access within a buffer is accomplished by using the buffer offset.

First party DMA is a superset capability not necessarily supported by legacy mode devices or legacy mode device drivers but essential for accommodating future capabilities.

10.5.9.3 Transmission

A device or host transmits a DMA Setup – Device to Host or Host to Device FIS as the first step in performing a DMA access. The purpose of the DMA Setup – Device to Host or Host to Device is to establish DMA hardware context for one or more data transfers.

A DMA Setup – Device to Host or Host to Device is required only if the DMA context is to be changed. Multiple Data – Host to Device or Device to Host FISes may follow in either direction.

EXAMPLE - If the transfer count exceeds the maximum Data – Host to Device or Device to Host transfer length or if a data transfer is interrupted.

If multiple Data – Host to Device FISes or Device to Host FISes follow a DMA Setup – Device to Host FIS or Host to Device FIS, then the device or host shall place the data contained in the FIS in sequential addresses; (i.e., if the last Dword of a FIS is placed in or obtained from address N, then the first Dword of a subsequent Data – Host to Device FIS or Device to Host FIS is placed in or obtained from address N+4 unless an intervening DMA Setup – Device to Host FIS or Host to Device FIS is used to alter the DMA context). This mechanism allows for the efficient streaming of data into a buffer.

10.5.9.4 Reception

10.5.9.4.1 Reception overview

Upon receiving a DMA Setup – Device to Host or Host to Device FIS, the receiver of the FIS shall validate the received DMA Setup request, and provided that the DMA Buffer Identifier and the specified offset/count are valid, program and arm the adapter’s DMA controller using the information in the FIS. The specific implementation of the DMA Buffer Identifier and buffer/address validation is not specified. After a valid DMA Setup – Device to Host or Host to Device FIS with the D bit cleared to zero, the receiver of the DMA Setup – Device to Host or Host to Device FIS responds with one or more Data – Host to Device or Device to Host FISes until the DMA count is exhausted. After a valid DMA Setup – Device to Host or Host to Device FIS with the D bit set to

one, the receiver of the FIS shall be prepared to accept one or more Data – Host to Device or Device to Host FISes until the DMA count is exhausted.

An interrupt pending condition shall be generated upon the completion of the DMA transfer if the Interrupt bit is set to one. The definition of DMA transfer completion is system dependent but typically includes the exhaustion of the transfer count or the detection of an error by the DMA controller.

NOTE 44 - First-party DMA accesses are categorized in two groups:
a) command/status transfers; and
b) user-data transfers.

Interrupts are not typically generated on user-data transfers. The optimal interrupt scheme for command/status transfers is not defined in this specification.

10.5.9.4.2 Auto-activate

First Party DMA transfers from the host to the device require transmission of both the DMA Setup FIS and a subsequent DMA Activate FIS in order to trigger the host transfer of data to the device. Because the device may elect to submit the DMA Setup FIS only if it is already prepared to receive the subsequent Data FIS from the host, the extra transaction for the DMA Activate FIS may be eliminated by merely having the DMA Setup FIS automatically activate the DMA controller by setting the Auto-Activate (A) bit to one in the DMA Setup FIS.

Devices shall not attempt to utilize this capability prior to the optimization having been explicitly enabled by the host as defined in 13.3.3. The host response to a DMA Setup FIS with the Auto-Activate bit set to one if the host has not enabled Auto-Activate is not defined.

10.5.9.5 HBA enforcement of first-party DMA data phase atomicity

The HBA shall ensure the First-party DMA Data Phase is uninterrupted. Unless the ERR bit in the shadow Status register is set to one, the host shall ensure no FIS other than requested data payload or a FIS for a software reset is transmitted from the host to device between the reception of a DMA Setup FIS and the exhaustion of the associated transfer count.

10.5.10 BIST Activate FIS - bidirectional

10.5.10.1 BIST Activate FIS – bidirectional layout

0	Reserved								Pattern Definition								R	R	R	R	PM Port				FIS Type (58h)							
									T	A	S	L	F	P	R	V															
1	Data1 (31:24)								Data1 (23:16)								Data1 (15:8)								Data1 (7:0)							
2	Data2 (31:24)								Data2 (23:16)								Data2 (15:8)								Data2 (7:0)							

Figure 280 – BIST Activate FIS – bidirectional

Field Definitions

Pattern Definition

T - Far end transmit only mode.
A - $ALIGN_P$ Bypass (Do not Transmit $ALIGN_P$ primitives) (valid only in combination with the T bit).
S - Bypass Scrambling (valid only in combination with the T bit).
L - Far End Retimed Loopback* Transmitter shall insert additional $ALIGN_P$ primitives.
F - Far End Analog (i.e., analog front end (AFE)) Loopback (Optional).
P - Primitive bit (valid only in combination with the T bit) (Optional).
R - Reserved, shall be cleared to zero.
V - Vendor Specific Test Mode. Causes all other bits to be ignored.

R — Reserved, shall be cleared to zero.

PM Port — If an endpoint device is attached via a Port Multiplier, specifies the device port address that the FIS should be delivered to or is received from. This field is set by the host for host to device transmission and this field is set by the Port Multiplier for device to host transmission. Endpoint devices shall clear this field to 0h for device to host transmissions.

FIS Type — Set to a value of 58h. Defines the rest of the FIS fields.

Data1 — Dword #1 of data information used to determine what pattern is transmitted as a result of the BIST Activate FIS. Applicable only if the T bit is set to one.

Data2 — Dword #2 of data information used to determine what pattern is transmitted as a result of the BIST Activate FIS. Applicable only if the T bit is set to one.

10.5.10.2 Description

The BIST Activate FIS (see Figure 280) shall be used to place the receiver in one of several loopback modes.

The BIST Activate FIS is a bi-directional FIS that may be sent by either the host or the device. The sender and receiver have distinct responsibilities in order to insure proper cooperation between the two parties. The state machines for transmission and reception of the FIS are symmetrical. The method of causing a BIST Activate FIS transmission is not defined in this specification.

The state machines for the transmission of the FIS do not attempt to specify the actions the sender takes once successful transmission of the request has been performed. After the Application layer is notified of the successful transmission of the FIS the sender's Application layer prepares its own Application, Transport, and Phy layers into the appropriate states that support the transmission of a stream of data. The FIS shall not be considered successfully transmitted until the receiver has acknowledged reception of the FIS as per normal FIS transfers documented in various sections of this specification. The transmitter of the BIST Activate FIS should transmit continuous $SYNC_P$

primitives after reception of R_OK$_P$ until such a time that it is ready to interact with the receiver in the build in self test (BIST) exchange.

Similarly, the state machines for the reception of the FIS do not specify the actions of the receiver's Application layer. Once the FIS has been received, the receiver's Application layer places its own Application, Transport, and Phy layers into states that perform the appropriate retransmission of the sender's data. The receiver shall not enter the BIST state until after it has properly received a good BIST Activate FIS (good CRC), indicated a successful transfer of the FIS to the transmitting side via R_OK$_P$ and has received at least one good SYNC$_P$. Once in the self-test mode, a receiver shall continue to allow processing of the COMINIT or COMRESET signals in order to exit from the self-test mode.

NOTE 45 - Note, that BIST mode is intended for Inspection/Observation Testing, as well as support for conventional laboratory equipment, rather than for in-system automated testing.

The setting of the F, L, and T bits is mutually exclusive. It is the responsibility of the sender of the BIST Activate FIS to ensure that only one of these bits is set. Refer to Table 100 for valid bit settings within a BIST Activate FIS.

Table 100 – BIST Activate FIS modes and bit settings

BIST Test Mode	F	L	T	P	A	S	V
Far End Analog Loopback	1	0	0	0	0	0	0
Far End Retimed Loopback	0	1	0	0	0	0	0
Far End Transmit with ALIGN$_P$ primitives, scrambled data	0	0	1	0	0	0	0
Far End Transmit with ALIGN$_P$ primitives, unscrambled data	0	0	1	0	0	1	0
Far End Transmit without ALIGN$_P$ primitives, scrambled data	0	0	1	0	1	0	0
Far End Transmit without ALIGN$_P$ primitives, unscrambled data	0	0	1	0	1	1	0
Far End Transmit primitives with ALIGN$_P$ primitives	0	0	1	1	0	na	0
Far End Transmit primitives without ALIGN$_P$ primitives	0	0	1	1	1	na	0
Vendor Specific	na	na	na	na	na	na	1
Key: 0 = bit shall be cleared to zero 1 = bit shall be set to one							

F, the Far End Analog (i.e., AFE) Loopback Mode is defined as a vendor optional mode where the raw data is received, and retransmitted, without any retiming or re-synchronization, etc. The implementation of Far End Analog Loopback is optional due to the round-trip characteristics of the test as well as the lack of retiming. This mode is intended to give a quick indication of connectivity, and test failure is not an indication of system failure.

L, the Far End Retimed LoopbackMode is defined as a mode where the receiver retimes the data, and retransmits the retimed data. The initiator of the retimed loopback mode shall account for the loopback device consuming up to two ALIGN$_P$ primitives every 256 Dwords transmitted and, if it requires any ALIGN$_P$ primitives to be present in the returned data stream, it should insert additional ALIGN$_P$ primitives in the transmitted stream. The initiator shall transmit additional two consecutive ALIGN$_P$ primitives in a single burst at the normal interval of every 256 Dwords transmitted (as opposed to inserting two consecutive ALIGN$_P$ primitives at half the interval).

The loopback device may remove zero, one, or two ALIGN$_P$ primitives from the received data. It may insert one or more ALIGN$_P$ primitives if they are directly preceded or followed by the initiator

inserted ALIGNP primitives (resulting in at least two consecutive ALIGNP primitives) or it may insert two or more ALIGNP primitives if not preceded or followed by the initiator's ALIGNP primitives. One side effect of the loopback retiming is that the returned data stream may have instances of an odd number of ALIGNP primitives, however, returned ALIGNP primitives are always in bursts and if the initiator transmitted four consecutive ALIGNP primitives, then the returned data stream shall include ALIGNP bursts that are no shorter than two ALIGNP primitives long (although the length of the ALIGNP burst may be odd). The initiator of the retimed loopback mode shall not assume any relationship between the relative position of the ALIGNP primitives returned by the loopback device and the relative position of the ALIGNP primitives sent by the initiator.

In retimed loopback mode, the initiator shall transmit only valid 8b/10b characters so the loopback device may 10b/8b decode it and re-encode it before retransmission. If the loopback device descrambles incoming data it is responsible for rescrambling it with the same sequence of scrambling syndromes in order to ensure the returned data is unchanged from the received data. The loopback device's running disparity for its transmitter and receiver are not guaranteed to be the same and thus the loopback initiator shall 10b/8b decode the returned data rather than use the raw 10b encoded returned stream for the purpose of data comparison. The loopback device shall return all received data unaltered and shall disregard protocol processing of primitives. Only the OOB signals and ALIGNP processing is acted on by the loopback device, while all other data is retransmitted without interpretation.

T, the Far-End Transmit Mode is defined as a mode that may be used to invoke the Far-End Interface to send data patterns, upon receipt of the BIST Activate FIS, as defined by the content located in Data1 and Data2. Data1 and Data2 shall be applicable only if the T bit is active, indicating "Far-End Transmit Mode". It is not required that the values within Data1 and Data2 are equal. These two Dwords are programmable to any value.

This data is modified by the following bits.

P, the transmit primitives bit. If this bit is set to one in far end transmit mode, the lowest order byte of the two following Dwords are treated as K Characters in order to identify the appropriate primitive(s) for transmission. The encoding for primitives as defined in 9.5. It is the responsibility of the sender of the BIST Activate FIS to ensure that the values contained within Data1 and Data2 are valid D character versions of the K character (i.e., BCh for K28.5). The setting of this bit is applicable only if the T bit is set.

A, ALIGNP primitive bypass mode. If set to one, no ALIGNP primitives are sent. If the A bit is not asserted, ALIGNP primitives are sent normally as defined in this specification. The setting of this bit is applicable only if the T bit is set.

S, the Bypass Scrambling Mode is defined as a mode that may be used to send data or patterns, during BIST activation, that are not scrambled, however are encoded and decoded to normal and legal 8b/10b values. The setting of this bit is applicable only if the T bit is set. The S bit is ignored if the P bit is set to one.

V, the vendor-specific mode is implementation specific and shall be reserved for individual vendor use. All other bits are ignored in this mode.

10.5.10.3 Transmission

The initiator transmits a BIST Activate to the recipient in order to initiate the BIST mode of operation.

10.5.10.4 Reception

Upon receiving a BIST Activate, the recipient shall begin operations as per the BIST Activate FIS, described in this specification.

10.5.11 PIO Setup – Device to Host FIS

10.5.11.1 PIO Setup – Device to Host FIS layout

0	Error(7:0)								Status(7:0)								R	I	D	R	PM Port				FIS Type (5Fh)							
1	Device(7:0)								LBA(23:16)								LBA(15:8)								LBA(7:0)							
2	Reserved								LBA(47:40)								LBA(39:32)								LBA(31:24)							
3	E_Status(7:0)								Reserved								Count(15:8)								Count(7:0)							
4	Reserved																Transfer Count(15:0)															

Figure 281 – PIO Setup – Device to Host FIS layout

Field Definitions

Error	Contains the new value of the Error register of the Command Block at the conclusion of all subsequent Data to Device frames.
Status	Contains the new value of the Status register of the Command Block for initiation of host data transfer.
R	Reserved, shall be cleared to zero.
I	Interrupt bit, this bit reflects the interrupt bit line of the device. Devices shall not modify the behavior of this bit based on the state of the nIEN bit received in Register Host to Device FISes.
D	Specifies the data transfer direction. If set to one, the transfer is from device to host. If cleared to zero, the transfer is from host to device.
PM Port	If an endpoint device is attached via a Port Multiplier, specifies the device port address that the FIS is received from. This field is set by the Port Multiplier. Endpoint devices shall clear this field to 0h.
FIS Type	Set to a value of 5Fh. Defines the rest of the FIS fields. Defines the length of the FIS as five Dwords.
Device	Contains the contents of the Device register of the Command Block.
LBA(23:16)	Contains the contents of the LBA(23:16) register of the Command Block.
LBA(15:8)	Contains the contents of the LBA(15:8) register of the Command Block.
LBA(7:0)	Contains the contents of the LBA(7:0) register of the Command Block.
LBA(47:40)	Contains the contents of the LBA(47:40) field of the Shadow Register Block.
LBA(39:32)	Contains the contents of the LBA(39:32) field of the Shadow Register Block.
LBA(31:24)	Contains the contents of the LBA(31:24) field of the Shadow Register Block.
E_Status	Contains the new value of the Status register of the Command Block at the conclusion of the subsequent Data FIS.

Count(15:8) Contains the contents of the Count(15:8) field of the Shadow Register Block.

Count(7:0) Contains the contents of the Count(7:0) register of the Command Block.

Transfer Count

Contains the number of bytes to be transferred in the subsequent Data FIS. The Transfer Count value shall be nonzero and the low order bit shall be cleared to zero (even number of bytes transferred).

10.5.11.2 Description

The PIO Setup – Device to Host FIS (see Figure 281) is used by the device to provide the host adapter with sufficient information regarding a Programmed Input/Output (PIO) data phase to allow the host adapter to efficiently handle PIO data transfers. For PIO data transfers, the device shall send to the host a PIO Setup – Device to Host FIS just before each and every data transfer FIS that is required to complete the data transfer. Data transfers from host to device as well as data transfers from device to host shall follow this algorithm. Because of the stringent timing constraints in the ATA standard, the PIO Setup FIS includes both the starting and ending status values. These are used by the host adapter to first signal to host software readiness for PIO write data (the BSY bit is cleared to zero and the DRQ bit is set to one), and following the PIO write burst to properly signal host software by clearing the DRQ bit to zero and possibly setting the BSY bit to one.

10.5.11.3 Transmission of PIO Setup by Device Prior to a Data Transfer from host to device

The device transmits a PIO Setup – Device to Host FIS to the host in preparation for a PIO data payload transfer just before each and every PIO data payload transfer required to complete the total data transfer for a command. The device includes in the FIS the values to be placed in the Shadow Status register at the beginning of the PIO data payload transfer and the value to be placed in the Shadow Status register at the end of the data payload transfer. The device shall be prepared to receive a Data FIS in response to transmitting a PIO Setup FIS.

10.5.11.4 Reception of PIO Setup by Host Prior to a Data Transfer from host to device

Upon receiving a PIO Setup – Device to Host FIS, the host shall update all Shadow registers and shall hold the E_Status value in a temporary register. The Transfer Length value shall be loaded into a countdown register. Upon detecting the change in the Shadow Status register, host software proceeds to perform a series of write operations to the Data shadow register that the host adapter shall collect to produce a Data FIS to the device. Each write of the Data shadow register results in another Word of data being concatenated into the Data FIS, and the countdown register being decremented accordingly. The E_Status value shall be transferred to the Shadow Status register within 400 ns of the countdown register reaching terminal count. In the case that the transfer length represents an odd number of Words, the last Word shall be placed in the low order (Word 0) of the final Dword and the high order Word (Word 1) of the final Dword shall be padded with zeros before transmission. This process is repeated for each and every data FIS needed to complete the overall data transfer of a command.

10.5.11.5 Transmission of PIO Setup by Device Prior to a Data Transfer from device to host

The device transmits a PIO Setup – Device to Host FIS to the host in preparation for a PIO data payload transfer just before each and every PIO data payload transfer required to complete the total data transfer for a command. The device includes in the FIS the values to be placed in the Shadow Status register at the beginning of the PIO data payload transfer and the value to be placed in the Shadow Status register at the end of the data payload transfer. The device shall be prepared to transmit a Data FIS following the transmittal of a PIO Setup FIS.

10.5.11.6 Reception of PIO Setup by Host Prior to a Data Transfer from device to host

Upon receiving a PIO Setup – Device to Host FIS for a device to host transfer, the host shall hold the Status, Error, and E_Status values in temporary registers. The Transfer Length value shall be loaded into a countdown register. Upon reception of a Data FIS from the device, the host shall update all Shadow registers and host software proceeds to perform a series of read operations from the Data shadow register. Each read of the Data shadow register results in a countdown register being decremented accordingly. The E_Status value shall be transferred to the Shadow Status register within 400 ns of the countdown register reaching terminal count. This process is repeated for each and every data FIS needed to complete the overall data transfer of a command.

10.5.12 Data - Host to Device FIS or Device to Host FIS (bidirectional)

10.5.12.1 Data - Host to Device FIS or Device to Host FIS (bidirectional) layout

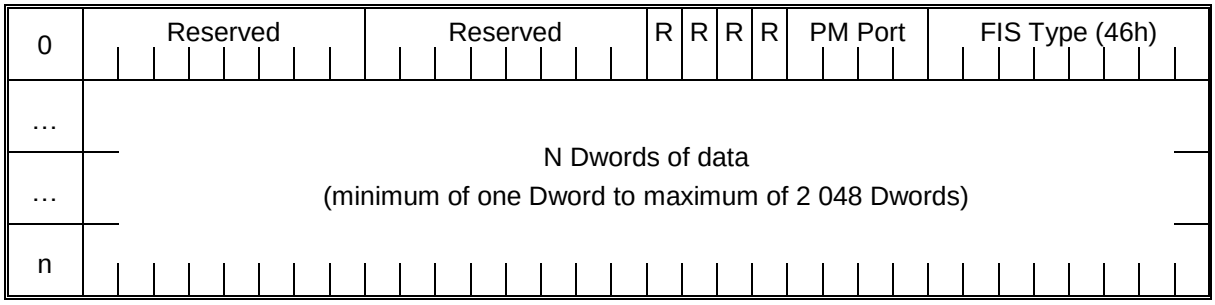


Figure 282 – Data – Host to Device or Device to Host FIS layout

Field Definitions

R Reserved, shall be cleared to zero.

PM Port If an endpoint device is attached via a Port Multiplier, specifies the device port address that the FIS should be delivered to or is received from. This field is set by the host for host to device transmission and this field is set by the Port Multiplier for device to host transmission. Endpoint devices shall clear this field to 0h for device to host transmissions.

FIS Type Set to a value of 46h. Defines the rest of the FIS fields. Defines the length of the FIS as n + 1 Dwords.

Dwords of data
Contain the actual data to transfer. Only 32 bit fields are transferred. The last Dword is padded with zeros if only a partial Dword is to be transmitted.

NOTE 46 - The maximum amount of user data that may be sent in a single Data – Host to Device or Data – Device to Host FIS is limited. See description.

10.5.12.2 Description

The Data – Host to Device and the Data – Device to Host FISes (see Figure 282) are used for transporting payload data, (e.g., the data read from or written to a number of sectors on a hard drive). The FIS may either be generated by the device to transmit data to the host or may be generated by the host to transmit data to the device. This FIS is generally only one element of a sequence of transactions leading up to a data transmission and the transactions leading up to and following the Data FIS establish the proper context for both the host and device.

The byte count of the payload is not an explicit parameter, rather it is inferred by counting the number of Dwords between SOF_P and EOF_P, and discounting the FIS Type and CRC Dwords. The payload size shall be no more than 2 048 Dwords (8 192 bytes). Non-packet devices, with or without bridges, should report a SET MULTIPLE limit of 16 sectors or less in Word 47 of their IDENTIFY DEVICE information.

In the case that the transfer length represents an odd number of Words, the last Word shall be placed in the low order (i.e., Word 0) of the final Dword and the high order Word (i.e., Word 1) of the final Dword shall be padded with zeros before transmission.

10.5.12.3 Transmission

The device transmits a Data – Device to Host FIS to the host during the data transfer phase of legacy mode PIO reads, DMA reads, and First-party DMA writes to host memory. The device shall precede a Data FIS with any necessary context-setting transactions as appropriate for the particular command sequence.

EXAMPLE 1 - A First-party DMA host memory write is preceded by a DMA Setup – Device to Host FIS to establish proper context for the Data FIS that follows.

The host transmits a Data – Host to Device FIS to the device during the data transfer phase of PIO writes, DMA writes, and First-party DMA reads of host memory. The FIS shall be preceded with any necessary context-setting transactions as appropriate for the particular command sequence.

EXAMPLE 2 - A legacy mode DMA write to the device is preceded by a DMA Activate – Device to Host FIS with the DMA context having been pre-established by the host.

If used for transferring data for DMA operations, multiple Data – Host to Device or Device to Host FISes may follow in either direction. Segmentation may occur if the transfer count exceeds the maximum Data – Host to Device or Device to Host transfer length or if a data transfer is interrupted.

If used for transferring data in response to a PIO Setup, the Data FIS shall contain the number of bytes indicated in the Transfer Count field of the preceding PIO Setup FIS.

In the event that a transfer is broken into multiple FISes, all intermediate FISes shall contain an integral number of full Dwords. If the total data transfer is for an odd number of Words, then the high order Word (Word 1) of the last Dword of the last FIS shall be padded with zeros before transmission and discarded on reception.

The Serial ATA protocol does not permit for the transfer of an odd number of bytes.

10.5.12.4 Reception

Neither the host nor device is expected to buffer an entire Data FIS in order to check the CRC of the FIS before processing the data. Incorrect data reception for a Data FIS shall be reflected in the overall command completion status.

10.6 Host transport states

10.6.1 Host transport states overview

FIS reception is asynchronous in nature. In the case of a non-Data FIS transmission, the host may be pre-empted by a non-Data FIS reception from the device. The host shall hold off on the pending transmission and process the incoming FIS from the device before attempting to retransmit the pending FIS.

10.6.2 Host transport idle state machine

The host transport idle state machine is defined in Figure 283.

HTI1: HT_HostIdle	Host adapter waits for frame or frame request.	
1. Register Host to Device FIS with the C bit set to one transmission pending	→	HT_CmdFIS
2. Register Host to Device FIS with the C bit cleared to zero transmission pending	→	HT_CntrlFIS [a]
3. Frame receipt indicated by Link layer	→	HT_ChkTyp [c]
4. DMA Setup FIS transmission pending	→	HT_DMASTUPFIS
5. BIST Activate FIS transmission pending	→	HT_XmitBIST
6. Previous FIS was PIO Setup and Application layer indicates data direction is host to device	→	HT_PIOOTrans2 [b]

[a] Transmission of a Register Host to Device FIS with the C bit cleared to zero is mandatory if the state of the SRST bit in the Device Control Shadow Register is changed, and is optional if the state of the SRST bit is not changed. If a Register Host to Device FIS with the C bit cleared to zero transmission with a modified value for the SRST bit is triggered while another Register Host to Device FIS with the C bit set to one transmission is already pending, all pending Register Host to Device FIS with the C bit set to one transmissions shall be aborted. If the state of the SRST bit is not modified from its previous value in a Register Host to Device FIS with the C bit cleared to zero transmission, then pending Register Host to Device FIS with the C bit set to one transmissions shall not be aborted.

[b] The PIO Setup FIS shall set an indication that PIO Setup was the last FIS received. Indication from the Application layer that it is transmitting data to the device may be determined from the Application layer performing write operations to the Data register in the Shadow Register Block.

[c] FIS reception shall have priority over all other transitions in this state.

Figure 283 – Host transport idle state machine (part 1 of 2)

HTI2: HT_ChkTyp	Received FIS Type checked.	
1. Register Device to Host FIS Type detected	→	HT_RegFIS
2. Set Device Bits FIS Type detected	→	HT_DB_FIS
3. DMA Activate FIS Type detected	→	HT_DMA_FIS
4. PIO Setup FIS Type detected	→	HT_PS_FIS
5. DMA Setup FIS Type detected	→	HT_DS_FIS
6. BIST FIS Type detected	→	HT RcvBIST
7. Data FIS Type detected and previous FIS was not PIO Setup	→	HT_DMAITrans [a]
8. Data FIS Type detected and previous FIS was PIO Setup	→	HT_PIOITrans1 [a]
9. Unrecognized FIS received	→	HT_HostIdle
10. Notification of illegal transition error received from Link layer	→	HT_HostIdle
[a] The PIO Setup FIS shall set an indication that PIO Setup was the last FIS sent, so that this state may determine whether to transition to DMA data transfer, or PIO data transfer.		

Figure 283 – Host transport idle state machine (part 2 of 2)

HTI1: HT_HostIdle state, this state is entered if a FIS transaction has been completed by the Transport layer.

If in this state, the Transport layer waits for the shadow Command register to be written, the shadow Device Control register to be written, or the Link layer to indicate that a FIS is being received.

TransitionHTI1:1, if a Register Host to Device FIS with the C bit set to one transmission is pending, the Transport layer shall make a transition to the HTCM1: HT_CmdFIS state. A Register Host to Device FIS with the C bit set to one becomes pending upon a write operation to the Command shadow register, and ceases pending at successful transmission of the FIS as indicated by the Link layer.

Transition HTI1:2, if a Register Host to Device FIS with the C bit cleared to zero transmission is pending, the Transport layer shall make a transition to the HTCR1: HT_CntrlFIS state. A Register Host to Device FIS with the C bit cleared to zero becomes pending upon a write operation to the Device Control shadow register that changes the state of the SRST bit from the previous value, or optionally upon a write operation to the Device Control shadow register that does not change the state of the SRST bit. A Register Host to Device FIS with the C bit cleared to zero ceases pending at successful transmission of the FIS as indicated by the Link layer.

Transition HTI1:3, if the Link layer indicates that a FIS is being received, the Transport layer shall make a transition to the HTI2: HT_ChkTyp state.

Transition HTI1:4, if the Application layer indicates that a DMA Setup FIS is to be sent, the Transport layer shall make a transition to the HT_DMASTUP0:HT_DMASTUPFIS state.

Transition HTI1:5, if the Application layer requests the transmission of a BIST request to the device, the Transport layer shall make a transition to the HTXBIST1:HT state.

Transition HTI1:6, if the Application layer requests the transmission of data to the device and the previous FIS was a PIOSetup type, the Transport layer shall make a transition to the

HTPS3:HT_PIOOTrans2 state. The Application layer signals transmission of PIO data to the device by performing writes to the Data register in the Shadow Register Block.

HTI2: HT_ChkTyp state, this state is entered if the Link layer indicates that a FIS is being received.

If in this state, the Transport layer checks the FIS Type of the incoming FIS.

Transition HTI2:1, if the incoming FIS is a register type, the Transport layer shall notify the Link layer that it has received a valid FIS, and make a transition to the HTR1: HT_RegFIS state.

Transition HTI2:2, if the incoming FIS is a Set Device Bits type, the Transport layer shall notify the Link layer that it has received a valid FIS and make a transition to the HTDB0:HT_DB_FIS state.

Transition HTI2:3, if the incoming FIS is a DMA Activate type, the Transport layer shall notify the Link layer that it has received a valid FIS, and make a transition to the HTDA1: HT_DMA_FIS state.

Transition HTI2:4, if the incoming FIS is a PIO Setup type, the Transport layer shall notify the Link layer that it has received a valid FIS, and make a transition to the HTPS1: HT_PS_FIS state.

Transition HTI2:5, if the incoming FIS is a DMA Setup type, the Transport layer shall notify the Link layer that it has received a valid FIS, and make a transition to the HTDS1: HT_DS_FIS state.

Transition HTI2:6, if the incoming FIS is a BIST Activate type, the Transport layer shall notify the Link layer that it has received a valid FIS, and make a transition to the HTRBIST1:HT_RcvBIST state.

Transition HTI2:7, if the incoming FIS is a Data type, and the previous FIS was not a PIO Setup type, the Transport layer shall notify the Link layer that it has received a valid FIS, and make a transition to the HTDA5:HT_DMAITrans state.

Transition HTI2:8, if the incoming FIS is a Data type, and the previous FIS was a PIO Setup type, the Transport layer shall notify the Link layer that it has received a valid FIS, and make a transition to the HTPS5:HT_PIOITrans1 state.

Transition HTI2:9, if the received FIS is of an unrecognized, or unsupported type, the Transport layer shall notify the Link layer that it has received an unrecognized FIS, and make a transition to the HTI1: HT_HostIdle state.

Transition HTI2:10, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HTI1: HT_HostIdle state.

10.6.3 Host transport transmit command FIS state machine

This protocol builds a FIS that contains the host adapter shadow register content and sends it to the device if the software driver or BIOS writes the host adapter shadow Command register (see Figure 284).

HTCM1: HT_CmdFIS	Construct Register Host to Device FIS with C bit set to one from the content of the shadow registers and notify Link to transfer.		
	1. FIS transfer complete	→	HT_CmdTransStatus
	2. Notification of illegal transition error received from Link layer	→	HT_HostIdle
	3. Frame receipt indicated by Link layer	→	HT_HostIdle

HTCM2: HT_CmdTransStatus	Check Link and Phy transmission results and if an error occurred take appropriate action.		
	1. Status checked and no error detected	→	HT_HostIdle
	2. Status checked and error detected [a]	→	HT_HostIdle
	[a] Upon return to the HT_HostIdle state in response to a detected error, the associated FIS remains pending for transmission.		

Figure 284 – Host transport transmit command FIS state machine

HTCM1: HT_CmdFIS state, this state is entered if the shadow Command register is written.

If in this state, the Transport layer shall construct a Register Host to Device FIS with C bit set to one, notify the Link layer that the Register Host to Device FIS is to be transmitted, and pass the Register Host to Device FIS to the Link layer.

Transition HTCM1:1, if the entire Register Host to Device FIS has been passed to the Link layer, the Transport layer shall indicate to the Link layer that the Register Host to Device FIS transmit is complete and make a transition to the HTCM2: HT_CmdTransStatus state.

Transition HTCM1:2, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HTI1: HT_HostIdle state.

Transition HTCM1:3, if the Link layer indicates that a FIS is being received, the Transport layer shall make a transition to the HTI1:HT_HostIdle state.

HTCM2: HT_CmdTransStatus state, this state is entered if the entire Register Host to Device FIS has been passed to the Link layer.

If in this state, the Transport layer shall wait for the Link and Phy layer ending status for the Register Host to Device FIS and take appropriate error handling action if required.

Transition HTCM2:1, if the Register Host to Device FIS status has been handled and no errors detected, the Transport layer shall transition to the HTI1: HT_HostIdle state.

Transition HTCM2:2, if the Register Host to Device FIS status has been handled and an error has been detected, the Transport layer shall transition to the HTI1: HT_HostIdle state. The associated Register Host to Device FIS remains pending for transmission.

10.6.4 Host transport transmit control FIS state machine

This protocol builds a Register Host to Device FIS that contains the host adapter shadow register content and sends it to the device if the software driver or BIOS writes the host adapter shadow Device Control register (see Figure 285).

HTCR1: HT_CntrlFIS	Construct Register Host to Device FIS with C bit cleared to zero from the content of the shadow registers and notify Link to transfer.	
1. FIS transfer complete	→	HT_CtrlTransStatus
2. Notification of illegal transition error received from Link layer	→	HT_HostIdle
3. Frame receipt indicated by Link layer	→	HT_HostIdle

HTCR2: HT_CtrlTransStatus	Check Link and Phy transmission results and if an error occurred take appropriate action.	
1. Status checked and no errors detected	→	HT_HostIdle
2. Status checked and error detected [a]	→	HT_HostIdle
[a] Upon return to the HT_HostIdle state in response to a detected error, the associated FIS remains pending for transmission.		

Figure 285 – Host transport transmit control FIS state machine

HTCR1: HT_Cntrl_FIS state, this state is entered if the shadow Device Control register is written.

If in this state, the Transport layer shall construct a Register Host to Device FIS with C bit cleared to zero, notify the Link layer that the Register Host to Device FIS is to be transmitted, and pass the Register Host to Device FIS to the Link layer.

Transition HTCR1:1, if the entire Register Host to Device FIS has been passed to the Link layer, the Transport layer shall indicate to the Link layer that the Register Host to Device FIS transmit is complete and make a transition to the HTCR2: HT_CtrlTransStatus state.

Transition HTCR1:2, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HTI1: HT_HostIdle state.

Transition HTCR1:3, if the Link layer indicates that a FIS is being received, the Transport layer shall make a transition to the HTI1:HT_HostIdle state.

HTCR2: HT_CtrlTransStatus state, this state is entered if the entire Register Host to Device FIS has been passed to the Link layer.

If in this state, the Transport layer shall wait for the Link and Phy ending status for the Register Host to Device FIS and take appropriate error handling action if required.

Transition HTCR2:1, if the Register Host to Device FIS status has been handled and no errors have been detected, the Transport layer shall transition to the HTI1: HT_HostIdle state.

Transition HTCR2:2, if the Register Host to Device FIS status has been handled and an error has been detected, the Transport layer shall transition to the HTI1: HT_HostIdle state. The associated Register Host to Device FIS remains pending for transmission.

10.6.5 Host transport transmit DMA Setup – Device to Host FIS or Host to Device FIS state machine

This protocol transmits a DMA Setup – Device to Host or Host to Device FIS to a receiver. This FIS is a request by a transmitter for the receiver to program its DMA controller for a First-party DMA transfer and is followed by one or more Data FISes that transfer data. The DMA Setup – Device to Host or Host to Device FIS request includes the transfer direction indicator, the host DMA Buffer Identifier, the host buffer offset, the byte count, and the interrupt flag (see Figure 286).

HTDMASTUP0: HT_DMASTUPFIS	Construct the DMA Setup – Host to Device or Device to Host FIS from the content provided by the Application layer and notifies Link to transfer.

1. FIS transfer complete	→	HT_DMASTUPTransStatus
2. Notification of illegal transition error received from Link layer	→	HT_HostIdle
3. Frame receipt indicated by Link layer	→	HT_HostIdle

HTPDMASTUP1: HT_DMASTUPTransStatus	Check Link and Phy transmission results and if an error occurred take appropriate action.

1. Status checked and no error detected	→	HT_HostIdle
2. Status checked and error detected [a]	→	HT_HostIdle
[a] Upon return to the HT_HostIdle state in response to a detected error, the associated FIS remains pending for transmission.		

Figure 286 – Host transport transmit DMA setup – Device to Host FIS or Host to Device FIS state machine

HTDMASTUP0: HT_DMASTUPFIS state, this state is entered if the Application requests the transmission of a DMA Setup – Host to Device or Device to Host FIS.

If in this state, the Transport layer shall construct a DMA Setup – Host to Device or Device to Host FIS, notify the Link layer that the FIS is to be transmitted, and pass the FIS to the Link layer.

Transition HTDMASTUP0:1, if the entire FIS has been passed to the Link layer, the Transport layer shall indicate to the Link layer that the FIS transmission is complete, and make a transition to the HTDMASTUP1: HT_DMASTUPTransStatus state.

Transition HTDMASTUP0:2, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HTI1: HT_HostIdle state.

Transition HTDMASTUP0:3, if the Link layer indicates that a FIS is being received, the Transport layer shall make a transition to the HTI1:HT_HostIdle state.

HTPDMASTUP1: HT_DMASTUPTransStatus state, this state is entered if the entire FIS has been passed to the Link layer.

If in this state, the Transport layer shall wait for the Link and Phy ending status for the FIS and take appropriate error handling action if required.

Transition HTDMASTUP1:1, if the FIS status has been handled, and no error detected, the Transport layer shall transition to the HTI1: HT_HostIdle state.

Transition HTDMASTUP1:2, if the FIS status has been handled, and an error detected, the Transport layer shall transition to the HTI1: HT_HostIdle state. The associated FIS remains pending for transmission.

10.6.6 Host transport transmit BIST Activate FIS state machine

This protocol builds a BIST Activate FIS that tells the device to prepare to enter the appropriate Built-in Self-test mode (see Figure 287). After successful transmission, the host Transport layer enters the idle state. The Application layer, upon detecting successful transmission to the device shall then cause the host's Transport layer, Link layer, and Phy layer to enter the appropriate mode for the transmission of the test data pattern defined by the FIS. The means that the Transport, Link, and Phy layers are placed into self-test mode are not defined by this specification.

HTXBIST1: HT_XmitBIST	Construct the BIST Activate FIS from the content provided by the Application layer and notify Link to transfer.	
1. FIS transfer complete	→	HT_TransBISTStatus
2. Notification of illegal transition error received from Link layer	→	HT_HostIdle
3. Frame receipt indicated by Link layer	→	HT_HostIdle

HTXBIST2: HT_TransBISTStatus	Check Link and Phy transmission results and if an error occurred take appropriate action.	
1. Status check completed	→	HT_HostIdle
2. Status check and at least one error detected	→	HT_HostIdle [a]
[a] Re-transmission of the BIST Activate FIS due to errors is not required but allowed.		

Figure 287 – Host transport transmit BIST Activate FIS state machine

HTXBIST1: HT_XmitBIST state, this state is entered to send a BIST FIS to the device.

Transition HTXBIST1:1, If the entire FIS has been passed to the Link layer, the Transport layer shall indicate to the Link layer that the FIS transmission is complete and make a transition to the HTXBIST2:HT_TransBISTStatus state.

Transition HTXBIST1:2, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HTI1: HT_HostIdle state.

Transition HTXBIST1:3, if the Link layer indicates that a FIS is being received, the Transport layer shall make a transition to the HTI1:HT_HostIdle state.

HTXBIST2: HT_TransBISTStatus state, this state is entered if the entire FIS has been passed to the Link layer.

Transition HTXBIST2:1, if the FIS transmission is completed, the Transport layer shall transition to the HTI1:HT_HostIdle state.

Transition HTXBIST2:2, if the FIS transmission is completed and at least one error is detected, the Transport layer shall transition to the HTI1:HT_HostIdle state. The associated FIS may remain pending for transmission.

10.6.7 Host transport decomposes Register FIS state machine

This protocol receives a Register Device to Host FIS from the device containing new shadow register content and places that content into the shadow registers (see Figure 288).

HTR1: HT_RegFIS	Place FIS contents from device into appropriate holding registers.	
1. FIS transfer complete	→	HT_RegTransStatus
2. Notification of illegal transition error received from Link layer	→	HT_HostIdle

HTR2: HT_RegTransStatus	Check Link and Phy transmission results and if an error occurred take appropriate action.	
1. Status checked	→	HT_HostIdle

Figure 288 – Host transport decomposes Register FIS state machine

HTR1: HT_RegFIS state, this state is entered if the Link layer has indicated that a FIS is being received and that the FIS is of the type Register Device to Host FIS.

If in this state, the Transport layer shall decompose the Register Device to Host FIS and place the contents into the appropriate holding registers.

Transition HTR1:1, if the entire Register Device to Host FIS has been placed into the holding registers, the Transport layer shall make a transition to the HTR2: HT_RegTransStatus state.

Transition HTR1:2, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HTI1: HT_HostIdle state.

HTR2: HT_RegTransStatus state, this state is entered if the entire Register Device to Host FIS has been placed into the holding registers.

If in this state, the Transport layer shall wait for the Link and Phy layer ending status for the Register Device to Host FIS and take appropriate error handling action if required.

Transition HTR2:1, if the Register Device to Host FIS status has been handled and no errors detected, the contents of the holding registers shall be placed in the shadow registers and if the interrupt bit is set to one, the Transport layer shall set the interrupt pending flag. The Transport layer shall transition to the HTI1: HT_HostIdle state. If the Register Device to Host FIS status has been handled and at least one error detected, the contents of the holding registers shall not be transferred to the shadow registers, error status shall be returned to the device, and the Transport layer shall transition to the HTI1: HT_HostIdle state.

10.6.8 Host transport decomposes a Set Device Bits FIS state machine

This protocol receives a Set Device Bits FIS from the device containing new Error and Status Shadow register content and places that content into the Error and Status Shadow registers (see Figure 289). The Set Device Bits FIS may also contain SActive register content and asynchronous notification content.

HTDB0:HT_DB_FIS	Receive Set Device Bits FIS		
	1. FIS status checked and no error detected	→	HT_Dev_Bits
	2. FIS status checked and error detected.	→	HT_HostIdle

HTDB1:HT_Dev_Bits	Load Error register and bits of the Status register		
	1. Register bits loaded	→	HT_HostIdle

Figure 289 – Host transport decomposes a Set Device Bits FIS state machine

HTDB0:HT_DB_FIS state, this state is entered if the Link layer has indicated that a FIS being received and that the FIS is a Set Device Bits type.

If in this state, the Transport layer shall wait for the FIS reception to complete and for Link and Phy layer ending status to be posted.

Transition HTDB0:1, if the FIS reception is complete with no errors detected, the Transport layer shall transition to the HTDB1:HT_Dev_Bits state.

Transition HTDB0:2, if the FIS reception is complete with errors detected, the Transport layer shall return error status to the device and transition to the HTI1:HT_HostIdle state.

HTDB1:HT_Dev_Bits state, this state is entered if a Set Device Bits FIS has been received with no errors.

If in this state, the data in the ERROR field of the received FIS shall be loaded into the host adapter's shadow Error register. The data in the Status-Hi field of the received FIS shall be loaded into bits 6, 5, and 4 of the shadow Status register. The data in the Status-Lo field of the received FIS shall be loaded into bits 2, 1, and 0 of the shadow Status register. The BSY bit and the DRQ bit in the shadow Status register shall not be changed. If the Interrupt bit in the FIS is set to one and if both the BSY bit and the DRQ bit in the shadow Status register are cleared to zero, then the host adapter shall enter an interrupt pending state.

Transition HTDB1:1, the Transport layer shall transition to the HTI1:HT_HostIdle state.

10.6.9 Host transport decomposes a DMA Activate FIS state machine

This protocol receives a DMA Activate FIS that requests a DMA data out transfer. The data transfer is from the host to the device and the DMA Activate FIS causes the host adapter to transmit the data in a subsequent Data FIS (see Figure 290).

HTDA1: HT_DMA_FIS		
1. Status checked and no error detected.	→	HT_DMAOTrans1
2. Status checked and error detected	→	HT_HostIdle
3. Notification of illegal transition error received from Link layer	→	HT_HostIdle

HTDA2: HT_DMAOTrans1	DMA controller initialized?	
1. DMA controller not initialized.	→	HT_DMAOTrans1
2. DMA controller initialized.	→	HT_DMAOTrans2
3. SRST asserted, or DEVICE RESET command requested	→	HT_HostIdle

HTDA3: HT_DMAOTrans2	Activate DMA controller	
1. Transfer not complete and < 2 048 Dwords transmitted	→	HT_DMAOTrans2
2. Transfer not complete and 2 048 Dwords transmitted	→	HT_DMAEnd
3. Abort notification from Link layer	→	HT_DMAEnd
4. Transfer complete	→	HT_DMAEnd
5. Notification of illegal transition error received from Link layer	→	HT_HostIdle
6. SRST asserted, or DEVICE RESET command requested	→	HT_HostIdle

HTDA4: HT_DMAEnd	Check DMA Controller completion	
1. DMA controller actions completed, no error detected	→	HT_HostIdle
2. DMA controller actions completed, and error detected.	→	HT_HostIdle
3. Abort notification from Link layer, no error detected	→	HT_HostIdle
4. Abort notification from Link layer, error detected.	→	HT_HostIdle

HTDA5: HT_DMAITrans	Activate DMA controller if initialized, receive Data FIS.	
1. Transfer not complete	→	HT_DMAITrans
2. SRST asserted, or DEVICE RESET command issued	→	HT_HostIdle
3. Transfer complete	→	HT_DMAEnd
4. Notification of illegal transition error received from Link layer	→	HT_HostIdle

Figure 290 – Host transport decomposes a DMA Activate FIS state machine

HTDA1: HT_DMA_FIS state, this state is entered if the Link layer has indicated that a FIS is being received and the Transport layer has determined that a DMA Activate FIS is being received.

If in this state, the Transport layer shall determine the direction of the DMA transfer being activated.

Transition HTDA1:1, the Transport layer shall make a transition to the HTDA2: HT_DMAOTrans1 state. This transition occurs if no error is detected.

Transition HTDA1:2, if an error is detected, status is conveyed to the Link layer and to the Application layer. The Transport layer shall make a transition to the HTI1:HT_HostIdle state.

Transition HTDA1:3, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HTI1: HT_HostIdle state.

HTDA2: HT_DMAOTrans1 state, this state is entered if it is determined that the DMA transfer that is being activated is a transfer from host to device.

If in this state, the Transport layer shall determine if the DMA controller has been initialized.

Transition HTDA2:1, if the DMA controller has not yet been initialized, the Transport layer shall transition to the HTDA2: HT_DMAOTrans1 state.

Transition HTDA2:2, if the DMA controller has been initialized, the Transport layer shall transition to the HTDA3: HT_DMAOTrans2 state.

Transition HTDA2:3, if the host has asserted the SRST bit by writing to the Device Control register, or the DEVICE RESET command is requested, the Transport layer shall inform the Link layer to send a SYNC Escape, and the Transport layer shall transition to the HTI1:HT_HostIdle state.

HTDA3: HT_DMAOTrans2 state, this state is entered if the DMA controller has been initialized.

If in this state, the Transport layer shall activate the DMA controller and pass data to the Link layer.

Transition HTDA3:1, if the transfer is not complete and less than 2 048 Dwords of payload data has been transmitted, the Transport layer shall transition to the HTDA3: HT_DMAOTrans2 state.

Transition HTDA3:2, if the transfer is not complete but 2 048 Dwords of payload data has been transmitted, the Link layer shall be notified to close the current frame, and the Transport layer shall deactivate the DMA engine, and transition to the HTDA4: HT_DMAEnd state.

Transition HTDA3:3, if notified by the Link layer that the DMA Abort primitive was received, the Transport layer shall transition to the HTDA4: HT_DMAEnd state.

Transition HTDA3:4, if the requested DMA transfer is complete, the Transport layer shall transition to the HTDA4: HT_DMAEnd state.

Transition HTDA3:5, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HTI1: HT_HostIdle state.

Transition HTDA3:6, if the host has asserted the SRST bit by writing to the Device Control register, or the DEVICE RESET command is requested, the Transport layer shall inform the Link layer to send a SYNC Escape, and the Transport layer shall transition to the HTI1:HT_HostIdle state.

HTDA4: HT_DMAEnd state, this state is entered if the DMA data transfer is complete.

If in this state, the Transport layer shall ensure that the activities of the DMA controller have completed.

Transition HTDA4:1, if the DMA controller has completed its activities, whether it has exhausted its transfer count or has been deactivated as a result of reaching the 2 048 Dword data payload

limit, the Transport layer shall transition to the HTI1: HT_HostIdle state. This transition occurs if no error is detected.

NOTE 47 - The host is unable to assume received data is valid (even with a valid CRC receipt for the FIS) until command completion status is returned by the device.

Transition HTDA4:2, if an error is detected, status shall be reported to the Link and Application layers. The Transport layer shall transition to the HTI1:HT_HostIdle state.

Transition HTDA4:3, if notified by the Link layer that a DMA Abort primitive was received, the transfer shall be truncated, and the Link layer notified to append CRC and end the frame. If it is determined that the transfer is completed with no error, the Transport layer shall make a transition to the HTI1:HT_HostIdle state.

Transition HTDA4:4, if notified by the Link layer that a DMA Abort primitive was received, the transfer shall be truncated, and the Link layer notified to append CRC and end the frame. If it is determined that the transfer is completed with an error, the Transport layer shall report status to the Application layer and make a transition to the HTI1:HT_HostIdle state.

HTDA5: HT_DMAITrans state, this state is entered if the Transport layer has determined that the DMA transfer being activated is from device to host.

If in this state, the Transport layer shall activate the DMA controller if the DMA controller is initialized. A data frame is received from the device and a received data Dword shall be placed in the data FIFO.

If in this state, the Transport layer shall wait until the Link layer has begun to receive the DMA data frame and data is available to be read by the host.

Transition HTDA5:1, if the transfer is not complete, the Transport layer shall transition to the HTDA5: HT_DMAITrans state. This includes the condition where the host DMA engine has not yet been programmed and the transfer is therefore held up until the DMA engine is prepared to transfer the received data to the destination memory locations.

Transition HTDA5:2, if the SRST bit is asserted by the host writing the Device Control register, or a DEVICE RESET command has been written to an ATAPI device, the Link layer shall be informed to send $SYNC_P$, and the Transport layer shall transition to the HTI1:HT_HostIdle state.

Transition HTDA5:3, if the requested DMA transfer is complete, the Transport layer shall transition to the HTDA4: HT_DMAEnd state.

Transition HTDA5:4, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HTI1: HT_HostIdle state.

10.6.10 Host transport decomposes a PIO Setup FIS state machine

This protocol receives a PIO Setup FIS that requests a PIO data transfer (see Figure 291). If the direction is from host to device, the Transport layer transmits a Data FIS to the device containing the PIO data. If the direction of transfer is from device to host, the Transport layer receives a Data FIS from the device. The PIO data shall be sent in a single Data FIS.

HTPS1: HT_PS_FIS	Determine the direction of the requested PIO transfer.	
1. Transfer host to device, no error detected. D bit cleared to zero.	→	HT_PIOOTrans1
2. Transfer device to host, no error detected. D bit set to one.	→	HT_HostIdle
3. Error detected.	→	HT_HostIdle
4. Notification of illegal transition error received from Link layer.	→	HT_HostIdle

HTPS2: HT_PIOOTrans1	Place initial register content received from FIS into shadow registers.	
1. Unconditional	→	HT_ HostIdle

HTPS3: HT_PIOOTrans2	Wait for Link layer to indicate data transfer complete.	
1. Transfer not complete.	→	HT_PIOOTrans2
2. Transfer complete.	→	HT_PIOEnd
3. Abort notification from Link layer.	→	HT_PIOEnd
4. Notification of illegal transition error received from Link layer.	→	HT_HostIdle
5. SRST asserted, or DEVICE RESET command requested.	→	HT_HostIdle

HTPS4: HT_PIOEnd	Place ending register content from PIO REQ FIS into shadow registers.	
1. No error detected.	→	HT_HostIdle
2. Error detected.	→	HT_HostIdle

HTPS5: HT_PIOITrans1	Wait until initial PIO data received in data frame.	
1. Received PIO data available.	→	HT_PIOITrans2 [a]
2. SRST asserted, or DEVICE RESET command requested.	→	HT_HostIdle
3. Notification of illegal transition error received from Link layer.	→	HT_HostIdle
[a] If transitioning to the HT_PIOITrans2 state, the starting status and Interrupt bit value from the PIO Setup FIS shall be transferred to the Shadow Status register and interrupt signal shall then reflect the value of the Interrupt bit.		

Figure 291 – Host transport decomposes a PIO Setup FIS state machine (part 1 of 2)

HTPS6: HT_PIOITrans2	Wait for Link layer to indicate data transfer complete.	
1. Transfer not complete.	→	HT_PIOITrans2
2. Transfer complete.	→	HT_PIOEnd
3. Abort notification from Link layer.	→	HT_PIOEnd
4. Notification of illegal transition error received from Link layer.	→	HT_HostIdle
5. SRST asserted, or DEVICE RESET command requested.	→	HT_HostIdle

Figure 291 – Host transport decomposes a PIO Setup FIS state machine (part 2 of 2)

HTPS1: HT_PS_FIS state, this state is entered if the Link layer has indicated that a FIS is being received and that the Transport layer has determined a PIO Setup FIS is being received.

If in this state, the Transport layer shall determine the direction of the requested PIO transfer and indicate that the last FIS sent was a PIO Setup.

Transition HTPS1:1, if the direction of transfer requested is from host to device (D bit cleared to zero), the Transport layer shall make a transition to the HTPS2: HT_PIOOTrans1 state. This transition occurs if no error is detected.

Transition HTPS1:2, if the direction of transfer requested is from device to host (D bit set to one), the Transport layer shall make a transition to the HTI1:HT_HostIdle state. This transition occurs if no error is detected.

Transition HTPS1:3, if an error is detected, status shall be reported to the Link layer. The Transport layer shall make a transition to the HTI1:HT_HostIdle state.

Transition HTPS1:4, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HTI1:HT_HostIdle state.

HTPS2: HT_PIOOTrans1 state, this state is entered if the direction of the requested PIO data transfer is from host to device.

If in this state, the Transport layer shall place the FIS initial register content into the shadow registers, may update the FIS byte count, and shall set the interrupt pending flag if the FIS indicates to do so.

Transition HTPS2:1, if the FIS initial register content has been placed into the shadow registers, interrupt pending set if requested, and the Transport layer is ready to begin transmitting the requested PIO data FIS, the Transport layer shall make a transition to the HTI1:HT_HostIdle state.

HTPS3: HT_PIOOTrans2 state, this state is entered if PIO data is available in the PIO FIFO to be passed the Link layer.

If in this state, the Transport layer shall wait for the Link layer to indicate that all data has been transferred.

NOTE 48 - Since the software driver or BIOS sees the DRQ bit set to one and the BSY bit cleared to zero, it continues writing the Data register filling the PIO FIFO.

Transition HTPS3:1, if the transfer is not complete, the Transport layer shall transition to the HTPS3: HT_PIOOTrans2 state.

Transition HTPS3:2, if the byte count for this DRQ data block is reached, the Transport layer shall transition to the HTPS4: HT_PIOEnd state.

Transition HTPS3:3, if notified by the Link layer that the DMA Abort primitive was received, the Transport layer shall transition to the HTPS4: HT_PIOEnd state.

Transition HTPS3:4, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HTI1:HT_HostIdle state.

Transition HTPS3:5, if the host has asserted the SRST bit by writing to the Device Control register, or the DEVICE RESET command is requested, the Transport layer shall inform the Link layer to send a SYNC Escape, and the Transport layer shall transition to the HTI1:HT_HostIdle state.

HTPS4: HT_PIOEnd state, this state is entered if the PIO data transfer is complete.

If in this state, the Transport layer shall place the ending register content from the received PIO request FIS into the shadow registers.

Transition HTPS4:1, if the ending register content for the PIO request FIS has been placed into the shadow registers and there were no errors detected with the transfer, the Transport layer shall transition to the HTI1: HT_HostIdle state.

> NOTE 49 - The host is unable to assume received data is valid (even with a valid CRC receipt for the FIS) until command completion status is returned by the device.

Transition HTPS4:2, if the ending register content from the previous PIO Setup FIS has been placed into the shadow registers, the Transport layer shall transition to the HTI1:HT_HostIdle state. For data in transfers, the Transport layer shall notify the Link layer of any error encountered during the transfer, and the error shall be reflected in the end of frame handshake. If the transfer was not the final transfer for the PIO data in command, the device shall reflect the error status by transmitting an appropriate Register Device to Host FIS to the host. If the transfer was the final transfer for the associated PIO data in command, the error condition is not detectable. For data out transfers, errors detected by the device shall be reflected in the end of frame handshake. The device shall reflect the error status by transmitting an appropriate Register Device to Host FIS to the host.

HTPS5: HT_PIOITrans1 state, this state is entered if the direction of the PIO data transfer is device to host.

If in this state, the Transport layer shall wait until the Link layer has begun to receive the PIO data frame and data is available to be read by the host.

Transition HTPS5:1, if data is available for the host to read in the shadow Data register, the Transport layer shall place the initial register content received in the PIO Setup frame into the shadow registers and transition to the HTPS6: HT_PIOITrans2 state.

Transition HTPS5:2, if the host has asserted the SRST bit by writing to the Device Control register, or the DEVICE RESET command is requested, the Transport layer shall inform the Link layer to send a SYNC Escape, and the Transport layer shall transition to the HTI1: HT_HostIdle state.

Transition HTPS5:3, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HTI1:HT_HostIdle state.

HTPS6: HT_PIOITrans2 state, this state is entered if PIO data is available in the PIO FIFO to be read by the host and the initial shadow register content has been set.

If in this state, the Transport layer shall wait for the Link layer to indicate that the data transfer is complete.

Transition HTPS6:1, if the transfer is not complete, the Transport layer shall transition to the HTPS6: HT_PIOITrans2 state.

Transition HTPS6:2, if the byte count for this DRQ data block is reached, the Transport layer shall transition to the HTPS4: HT_PIOEnd state.

Transition HTPS6:3, if notified by the Link layer that the DMA Abort primitive was received, the Transport layer shall transition to the HTPS4: HT_PIOEnd state.

Transition HTPS6:4, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HTI1:HT_HostIdle state.

Transition HTPS6:5, if the host has asserted the SRST bit by writing to the Device Control register, or the DEVICE RESET command is requested, the Transport layer shall inform the Link layer to send a SYNC Escape, and the Transport layer shall transition to the HTI1: HT_HostIdle state.

10.6.11 Host transport decomposes a DMA Setup FIS state machine

This protocol receives a FIS that sets up the host adapter DMA controller to allow the transfer of subsequent Data FISes according to the First-party DMA protocol (see Figure 292). For a First-party DMA write request if Auto-Activate is not used, a separate DMA Activate FIS is issued by the device to trigger the start of the data transfer from the host.

HTDS1: HT_DS_FIS	Initialize the DMA controller for a First-party DMA transfer with the content of the DMA Setup FIS.	
1. No error detected and (D bit set to one in DMA Setup FIS) or (D bit cleared to zero and Auto-Activate bit cleared to zero in DMA Setup FIS).	→	HT_HostIdle
2. Error detected.	→	HT_HostIdle
3. Notification of illegal transition error received from Link layer	→	HT_HostIdle
4. No error detected and (D bit cleared to zero and Auto-Activate bit set to one in DMA Setup FIS).	→	HT_DMAOTrans2

Figure 292 – Host transport decomposes a DMA Setup FIS state machine

HTDS1: HT_DS_FIS state, this state is entered if the Link layer has indicated that a FIS is being received and that the Transport layer has determined the FIS is of the DMA Setup type.

If in this state, the Transport layer shall initialize the DMA controller with content from the FIS.

Transition HTDS1:1, if the DMA controller has been initialized and the request is a read (i.e, Direction is set to one), or (the request is a write (i.e., Direction is cleared to zero) and Auto-Activate is cleared to zero in the DMA Setup FIS), the Transport layer shall transition to the HTI1: HT_HostIdle state. This transition is made if no error is detected.

Transition HTDS1:2, if an error is detected, status shall be reported to the Link layer. The Transport layer shall transition to the HTI1:HI_HostIdle state.

Transition HTDS1:3, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HTI1:HT_HostIdle state.

Transition HTDS1:4, if the DMA controller has been initialized and the request is a write (Direction is cleared to zero) and Auto-Activate is one in the DMA Setup FIS, the Transport layer shall transition to the HT_DMAOTrans2 state. This transition is made if no error is detected.

10.6.12 Host transport decomposes a BIST Activate FIS state machine

This protocol receives a BIST Activate FIS that instructs the host to enter one of several Built-in Self-test modes that cause the host to retransmit the data it receives (see Figure 293). If the mode is supported the host's Application layer places both the transmit and receive portions of the Transport, Link, or Phy layers into appropriate state to perform the loopback operation.

HTRBIST1: HT_RcvBIST	Determine validity of loopback mode requested.	
1. Status checked, no error detected and Loopback mode valid	→	HT_BISTTrans1
2. Status checked, no error detected and Loopback mode is invalid or not supported.	→	HT_HostIdle
3. Status checked and error detected	→	HT_HostIdle
4. Notification of illegal transition error received from Link layer	→	HT_HostIdle

HTRBIST2: HT_BISTTrans1	Notify Application layer of desired BIST modes	
1. Unconditional	→	HT_HostIdle

Figure 293 – Host transport decomposes a BIST Activate FIS state machine

HTRBIST1: HT_RcvBIST state, this state is entered if the Link layer has indicated that a FIS is being received and the Transport layer has determined that a BIST Activate FIS is being received.

If in this state, the Transport layer shall determine the validity of the loopback request.

Transition HTRBIST1:1, if no reception error is detected and the FIS contents indicate a form of loopback request that is supported by the host the Transport layer shall make a transition to the HTRBIST2: HT_BISTTrans1 state.

Transition HTRBIST1:2, if no reception error is detected and the FIS contents indicate a form of loopback request that is not supported by the host the Transport layer shall make a transition to the HTI1:HT_HostIdle state.

Transition HTRBIST1:3, if a reception error is indicated the Transport layer shall make a transition to the HTI1:HT_HostIdle state.

Transition HTRBIST1:4, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HTI1:HT_HostIdle state.

HTRBIST2: HT_BISTTrans1 state, this state is entered if the Transport layer has determined that a valid BIST Activate FIS has been received.

Having received a valid FIS, the Transport layer informs the Application layer that it should place the Transport, Link, and Phy layers into the appropriate modes to loop the received data back to the transmitter. The method that this is performed by is not defined in this specification.

Transition HTRBIST2:1, if the Application layer has been notified, the Transport layer shall transition to the HTI1:HostIdle state.

10.7 Device transport states

10.7.1 Device transport idle state machine

The device transport idle state machine is defined in Figure 294.

DTI0: DT_DeviceIdle	Device waits for FIS or FIS request.	
1. Transmission of Register Device to Host FIS requested by Application layer	→	DT_RegDHFIS
2. Transmission of Set Device Bits FIS requested by Application layer	→	DT_DB_FIS
3. Transmission of PIO Setup FIS requested by Application layer	→	DT_PIOSTUPFIS
4. Transmission of DMA Activate FIS requested by Application layer	→	DT_DMAACTFIS
5. Transmission of DMA Setup FIS requested by Application layer	→	DT_DMASTUPDHFIS
6. Transmission of Data FIS requested by Application layer	→	DT_DATAIFIS
7. Transmission of BIST Activate FIS requested by Application layer	→	DT_XmitBIST
8. Frame receipt indicated by Link layer	→	DT_ChkTyp

DTI1: DT_ChkTyp	Received FIS Type checked.	
1. Register Host to Device FIS Type detected	→	DT_RegHDFIS
2. Data FIS Type detected	→	DT_DATAOFIS
3. DMA Setup FIS Type detected	→	DT_DMASTUPHDFIS
4. BIST Activate FIS Type detected	→	DT_RcvBIST
5. Notification of illegal transition error received from Link layer or unrecognized FIS Type	→	DT_DeviceIdle

Figure 294 – Device transport idle state machine

DTI0: DT_DeviceIdle state, this state is entered if a FIS transaction has been completed by the Transport layer.

If in this state, the Transport layer waits for the Application layer to indicate that a FIS is to be transmitted or the Link layer to indicate that a FIS is being received.

Transition DTI0:1, if the Application layer indicates that a Register Device to Host FIS is to be transmitted, the Transport layer shall make a transition to the DTR0: DT_RegDHFIS state.

Transition DTI0:2, if the Application layer indicates that a Set Device Bits FIS is to be transmitted, the Transport layer shall make a transition to the DTDB0:DT_DB_FIS state.

Transition DTI0:3, if the Application layer indicates that a PIO Setup FIS is to be transmitted, the Transport layer shall make a transition to the DTPIOSTUP0: DT_PIOSTUPFIS state.

Transition DTI0:4, if the Application layer indicates that a DMA Activate FIS is to be transmitted, the Transport layer shall make a transition to the DTDMAACT0: DT_DMAACTFIS state.

Transition DTI0:5, if the Application layer indicates that a DMA Setup FIS is to be transmitted, the Transport layer shall make a transition to the DTDMASTUP0: DT_DMASTUPDHFIS state.

Transition DTI0:6, if the Application layer indicates that a Data FIS is to be transmitted, the Transport layer shall make a transition to the DTDATAI0: DT_DATAIFIS state.

Transition DTI0:7, if the Application layer indicates that a BIST Activate FIS is to be transmitted, the Transport layer shall make a transition to the DTXBIST1:DT XmitBIST state.

Transition DTI0:8, if the Link layer indicates that a FIS is being received, the Transport layer shall make a transition to the DTI1: DT_ChkTyp state.

DTI1: DT_ChkTyp state, this state is entered if the Transport layer is idle and Link layer indicates that a FIS is being received.

If in this state, the Transport layer checks the FIS Type of the incoming FIS.

Transition DTI1:1, if the incoming FIS is a Register Host to Device FIS Type, the Transport layer shall make a transition to the DTCMD0: DT_RegHDFIS state.

Transition DTI1:2, if the incoming FIS is a Data - Host to Device FIS Type, the Transport layer shall make a transition to the DTDATAO0: DT_DATAOFIS state.

Transition DTI1:3, if the incoming FIS is a DMA Setup FIS Type, the Transport layer shall make a transition to the DTSTP0: DT_ DMASTUPHDFIS state.

Transition DTI1:4, if the incoming FIS is a BIST Activate FIS Type, the Transport layer shall make a transition to the DTRBIST1:DT_RcvBIST state.

Transition DTI1:5, if the Transport layer receives notification from the Link layer of an illegal state transition or the FIS Type is not recognized, the Transport layer shall make a transition to the DTI0: DT_DeviceIdle state.

10.7.2 Device transport sends Register Device to Host state machine

This protocol builds a Register Device to Host FIS that contains the register content and sends it to the host if the Application layer requests the transmission (see Figure 295).

DTR0: DT_RegDHFIS	Construct Register Device to Host FIS from the content of the registers and notify Link to transfer.	
1. FIS transfer complete.	→	DT_RegTransStatus
2. Notification of illegal transition error received from Link layer.	→	DT_DeviceIdle

DTR1: DT_RegTransStatus	Check Link and Phy transmission results and if an error occurred take appropriate action.	
1. Status checked, and no error detected.	→	DT_DeviceIdle
2. Status checked, and error detected.	→	DT_RegDHFIS

Figure 295 – Device transport sends Register Device to Host state machine

DTR0: DT_RegDHFIS state, this state is entered if the Application requests the transmission of a Register Device to Host FIS.

If in this state, the Transport layer shall construct a Register Device to Host FIS, notify the Link layer that the FIS is to be transmitted, and pass the FIS to the Link layer.

Transition DTR0:1, if the entire FIS has been passed to the Link layer, the Transport layer shall indicate to the Link layer that the FIS transmission is complete and make a transition to the DTR1: DT_RegTransStatus state.

Transition DTR0:2, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the DTI0: DTDeviceIdle state.

DTR1: DT_RegTransStatus state, this state is entered if the entire FIS has been passed to the Link layer.

If in this state, the Transport layer shall wait for the Link and Phy layer ending status for the FIS and take appropriate error handling action if required.

Transition DTR1:1, if the FIS status has been handled, and no error detected, the Transport layer shall transition to the DTI0: DT_DeviceIdle state.

Transition DTR1:2, if the FIS status has been handled, and an error detected, the Transport layer shall report status to the Link layer, and retry this transfer by transitioning to the DT_RegDHFIS state.

10.7.3 Device transport sends Set Device Bits FIS state machine

This protocol sends a Set Device Bits FIS to the host adapter if the Application layer requests the transmission (see Figure 296).

DTDB0: DT_DB_FIS	Inform Link to transmit Set Device Bits FIS.	
1. FIS transfer complete.	→	DT_SDBTransStatus
2. Notification of illegal transition error received from Link layer.	→	DT_DeviceIdle

DTDB1: DT_SDBTransStatus	Check Link and Phy transmission results and if an error occurred take appropriate action.	
1. Status checked and no error detected.	→	DT_DeviceIdle
2. Status checked and error detected.	→	DT_DB_FIS

Figure 296 – Device transport sends Set Device Bits FIS state machine

DTDB0:DT_DB_FIS state, this state is entered if the Application layer requests the transmission of a Set Device Bits FIS.

If in this state, the Transport layer shall construct a Set Device Bits FIS, notify the Link layer that the FIS is to be transmitted, and pass the FIS to the Link layer.

Transition DTDB0:1, if the entire FIS has been passed to the Link layer, the Transport layer shall indicate to the Link layer that the FIS transmission is complete and make a transition to the DTDB1:DT_SDBTransStatus state.

Transition DTDB0:2, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the DTI0: DT_DeviceIdle state.

DTDB1:DT_SDBTransStatus state, this state is entered if the entire FIS has been passed to the Link layer.

If in this state, the Transport layer shall wait for the Link and Phy layer ending status for the FIS and take appropriate error handling action if required.

Transition DTDB1:1, if the FIS status has been handled and no error detected, the Transport layer shall transition to the DTI0:DT_DeviceIdle state.

Transition DTDB1:2, if the FIS status has been handled and an error detected, the Transport layer shall report status to the Link layer and retry this transfer by transitioning to the DTDB0:DT_DB_FIS state.

10.7.4 Device transport transmit PIO Setup – Device to Host FIS state machine

This protocol transmits a PIO Setup – Device to Host FIS to the host (see Figure 297). Following this PIO Setup frame, a single data frame containing PIO data shall be transmitted or received depending on the state of the D bit in the PIO Setup frame.

DTPIOSTUP0: DT_PIOSTUPFIS	Construct PIO Setup – Device to Host FIS from the content provided by the Application layer and notify Link to transfer. This FIS shall include the beginning and ending register content, the byte count, and the interrupt flag.	
1. FIS transfer complete.	→	DT_PIOSTUPTransStatus
2. Notification of illegal transition error received from Link layer.	→	DT_DeviceIdle

DTPIOSTUP1: DT_PIOSTUPTransStatus	Check Link and Phy transmission results and if an error occurred take appropriate action.	
1. Status checked, and no error detected.	→	DT_Device Idle
2. Status checked, and error detected.	→	DT_PIOSTUPFIS

Figure 297 – Device transport PIO Setup – Device to Host FIS state machine

DTPIOSTUP0: DT_PIOSTUPFIS state, this state is entered if the Application layer requests the transmission of a PIO Setup – Device to Host FIS.

If in this state, the Transport layer shall construct a PIO Setup – Device to Host FIS, notify the Link layer that the FIS is to be transmitted, and pass the FIS to the Link layer.

Transition DTPIOSTUP0:1, if the entire FIS has been passed to the Link layer, the Transport layer shall indicate to the Link layer that the FIS transmission is complete and make a transition to the DTPIOSTUP1: DT_PIOSTUPTransStatus state.

Transition DTPIOSTUP0:2, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the DTI0: DT_DeviceIdle state.

DTPIOSTUP1: DT_PIOSTUPTransStatus state, this state is entered if the entire FIS has been passed to the Link layer.

If in this state, the Transport layer shall wait for the Link and Phy ending status for the FIS and take appropriate error handling action if required.

Transition DTPIOSTUP1:1, if the FIS status has been handled and no error detected, the Transport layer shall transition to the DTI0: DT_DeviceIdle state.

Transition DTPIOSTUP1:2, if the FIS status has been handled and an error detected, the Transport layer shall report status to the Link layer and retry this transfer by transitioning to the DT_PIOSTUPFIS state.

10.7.5 Device transport transmit DMA Activate FIS state machine

This protocol transmits a DMA Activate FIS to the host adapter (see Figure 298). Following the DMA Activate FIS, a Data FIS shall be sent from the host to the device.

DTDMAACT0: DT_DMAACTFIS	Construct DMA Activate FIS from the content provided by the Application layer and notify Link to transfer.		
	1. FIS transfer complete.	→	DT_DMAACTTransStatus
	2. Notification of illegal transition error received from Link layer	→	DT_DeviceIdle

DTDMAACT1: DT_DMAACTTransStatus	Check Link and Phy transmission results and if an error occurred take appropriate action.	
1. Status checked, and no error detected.	→	DT_DeviceIdle
2. Status checked, and error detected.	→	DT_DMAACTFIS

Figure 298 – Device transport transmit DMA Activate FIS state machine

DTDMAACT0: DT_DMAACTFIS state, this state is entered if the Application layer requests the transmission of a DMA Activate FIS.

If in this state, the Transport layer shall construct a DMA Activate FIS, notify the Link layer that the FIS is to be transmitted, and pass the FIS to the Link layer.

Transition DTDMAACT0:1, if the entire FIS has been passed to the Link layer, the Transport layer shall indicate to the Link layer that the FIS transmission is complete and make a transition to the DTDMAACT1: DT_DMAACTTransStatus state.

Transition DTDMAACT0:2, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the DTI0: DT_DeviceIdle state.

DTDMAACT1: DT_DMAACTTransStatus state, this state entered is entered if the entire FIS has been passed to the Link layer.

If in this state, the Transport layer shall wait for the Link and Phy ending status for the FIS and take appropriate error handling action if required.

Transition DTDMAACT1:1, if the FIS status has been handled and no error detected, the Transport layer shall transition to the DTI0: DT_DeviceIdle state.

Transition DTDMAACT1:2, if the FIS status has been handled and an error detected, the Transport layer shall report status to the Link layer and retry this transfer by transitioning to the DT_DMAACTFIS state.

10.7.6 Device transport transmit DMA Setup – Device to Host FIS state machine

This protocol transmits a DMA Setup – Device to Host FIS to the host adapter (see Figure 299). This FIS is a request by the device for the host adapter to program the DMA controller for a First-

party DMA transfer and is followed by one or more Data FISes that transfer the data to or from the host adapter depending on the direction of the transfer. The DMA Setup – Device to Host request includes the transfer direction indicator, the host DMA Buffer Identifier, the host buffer offset, the byte count, and the interrupt flag.

DTDMASTUP0: DT_DMASTUPDHFIS	Construct the DMA Setup – Device to Host FIS from the content provided by the Application layer and notify Link to transfer.

1. FIS transfer complete.	→	DT_DMASTUPTransStatus
2. Notification of illegal transition error received from Link layer.	→	DT_DeviceIdle

DTDMASTUP1: DT_DMASTUPTransStatus	Check Link and Phy transmission results and if an error occurred take appropriate action.

1. Status checked, and no error detected.	→	DT_DeviceIdle
2. Status checked, and error detected.	→	DT_DMASTUPDHFIS

Figure 299 – Device transport transmit DMA Setup – Device to Host FIS state machine

DTDMASTUP0: DT_DMASTUPDHFIS state, this state is entered if the Application layer requests the transmission of a DMA Setup – Device to Host FIS.

If in this state, the Transport layer shall construct a DMA Setup – Device to Host FIS, notify the Link layer that the FIS is to be transmitted, and pass the FIS to the Link layer.

Transition DTDMASTUP0:1, if the entire FIS has been passed to the Link layer, the Transport layer shall indicate to the Link layer that the FIS transmission is complete and make a transition to the DTDMASTUP1: DT_DMASTUPTransStatus state.

Transition DTIDMASTUP0:2, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the DTI0: DT_DeviceIdle state.

DTDMASTUP1: DT_DMASTUPTransStatus state, this state is entered if the entire FIS has been passed to the Link layer.

If in this state, the Transport layer shall wait for the Link and Phy ending status for the FIS and take appropriate error handling action if required.

Transition DTDMASTUP1:1, if the FIS status has been handled and no error detected, the Transport layer shall transition to the DTI0: DT_DeviceIdle state.

Transition DTDMASTUP1:2, if the FIS status has been handled and an error detected, the Transport layer shall report status to the Link layer and retry this transfer by transitioning to the DT_DMASTUPDHFIS state.

10.7.7 Device transport transmit Data – Device to Host FIS state machine

This protocol builds a Data – Device to Host FIS (see Figure 300).

DTDATAI0: DT_DATAIFIS	Construct Data – Device to Host FIS content from the content provided by the Application layer and notify Link to transfer.	
1. Unconditional	→	DT_DATAITrans

DTDATAI1: DT_DATAITrans	Pass data Dwords from data FIFO to Link layer.	
1. Transfer not complete.	→	DT_DATAITrans
2. Transfer complete.	→	DT_DATAIEnd
3. Application layer requests termination of DMA in transfer.	→	DT_DATAIEnd
4. Notification of illegal transition error received from Link layer.	→	DT_DeviceIdle

DTDATAI2: DT_DATAIEnd	Check Link and Phy transmission results and if an error occurred take appropriate action.	
1. Status checked, and no error detected.	→	DT_DeviceIdle
2. Status checked, and error detected.	→	DT_DeviceIdle
3. Application layer requests termination of DMA in transfer, no error detected.	→	DT_DeviceIdle
4. Application layer requests termination of DMA in transfer, error detected.	→	DT_DeviceIdle
5. Notification of illegal transition error received from Link layer.	→	DT_DeviceIdle

Figure 300 – Device transport transmit data device to host state machine

DTDATAI0: DT_DATAFIS state, this state is entered if the Application layer has requested the transmission of a Data – Device to Host FIS.

If in this state, the Transport layer shall pass a portion of the DMA data to the Link layer.

Transition DTDATAI0:1, if ready and there is data in the FIFO to be passed to the Link layer, the Transport layer shall transition to the DTDATAI1: DT_DATAITrans state.

DTDATAI1: DT_DATAITrans state, this state is entered if data is available in the FIFO to be passed the Link layer.

If in this state, the Transport layer shall pass a Dword of data from the FIFO to the Link layer.

Transition DTDATAI1:1, if the transfer is not complete, the Transport layer shall transition to the DTDATAI1: DT_DATAITrans state.

Transition DTDATAI1:2, if the transfer is complete, the Transport layer shall transition to the DTDATAI2: DT_DATAIEnd state.

Transition DTDATAI1:3, if the Application layer requests that a DMA operation is to be aborted, the Transport layer shall transition to the DTDATAI2:DT_DATAIEnd state.

Transition DTDATAI1:4, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the DTI0: DT_DeviceIdle state.

DTDATAI2: DT_DATAIEnd state, this state is entered if the data transfer is complete or an abort has been requested by the Application layer.

If in this state, the Transport layer shall wait for the Link layer and Phy layer ending status for the FIS and take appropriate error handling action if required.

Transition DTDATAI2:1, if the FIS status has been handled and no error detected, the Transport layer shall transition to the DTI0: DT_DeviceIdle state.

Transition DTDATAI2:2, if the FIS status has been handled and an error detected, status shall be reported to the Link layer. The Transport layer shall transition to the DTI0: DT_DeviceIdle state.

Transition DTDATAI2:3, if the Application layer requests the termination of a DMA data in transaction, it reports the abort condition to the Link layer, waits for an EOF_P and, if no error is detected, shall transition to the DTI0:DT_DeviceIdle state.

Transition DTDATAI2:4, if the Application layer requests the termination of a DMA data in transaction, it reports the abort condition to the Link layer, waits for an EOF_P, and if an error is detected, reports the error to the Link layer, and shall transition to the DTI0:DT_DeviceIdle state.

Transition DTDATAI2:5, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall transition to the DTI0: DT_DeviceIdle state.

10.7.8 Device transport transmit BIST Activate FIS state machine

This protocol builds a BIST Activate FIS that tells the host to prepare to enter the appropriate Built-in Self Test mode (see Figure 301). After successful transmission, the device Transport layer enters the idle state. The Application layer, upon detecting successful transmission to the host shall then cause the device's Transport layer, Link layer, and Phy layer to enter the appropriate mode for the transmission of the Built-in Test data defined by the FIS. The means that the Transport layer, Link layer, and Phy layer are placed into self-test mode are not defined by this specification.

DTXBIST1: DT_XmitBIST	Construct the BIST Activate FIS from the content provided by the Application layer and notifies Link layer to transfer.	
1. FIS transfer complete	→	DT_TransBISTStatus
2. Notification of illegal transition error received from Link layer	→	DT_DeviceIdle

DTXBIST2: DT_TransBISTStatus	Check Link layer and Phy layer transmission results and if an error occurred take appropriate action.	
1. Status check completed	→	DT_DeviceIdle
2. Status check and at least one error detected	→	DT_XmitBIST [a]
[a] Re-transmission of the BIST Activate FIS due to errors is not required but allowed.		

Figure 301 – Device transport transmit BIST state machine

DTXBIST1: DT_XmitBIST state, this state is entered to send a BIST FIS to the host.

Transition DTXBIST1:1: If the entire FIS has been passed to the Link layer, the Transport layer shall indicate to the Link layer that the FIS transmission is complete and make a transition to the DTXBIST2:DT_TransBISTStatus state.

Transition DTXBIST1:2, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the DTI0: DT_DeviceIdle state.

DTXBIST2: DT_TransBISTStatus state, this state is entered if the entire FIS has been passed to the Link layer.

Transition DTXBIST2:1, if the FIS transmission is completed, the Transport layer shall transition to the DTI0: DT_DeviceIdle state.

Transition DTXBIST2:2, if the FIS transmission is completed and at least one error is detected, the Transport layer may transition to the DTXBIST1: DT_XmitBIST state.

10.7.9 Device transport decomposes Register Host to Device FIS state machine

This protocol receives a Register Host to Device FIS, places received register content into the device registers, and notifies the Application layer of the FIS receipt (see Figure 302).

DTCMD0: DT_RegHDFIS	Receive a Register Host to Device FIS.		
1.	FIS transfer complete, and no error detected.	→	DT_DeviceIdle
2.	FIS transfer complete, and error detected.	→	DT_DeviceIdle
3.	Notification of illegal transition error received from Link layer.	→	DT_DeviceIdle

Figure 302 – Device transport register host to device state machine

DTCMD0: DT_RegHDFIS state, this state is entered if the receipt of a DT_RegHDFISFIS is recognized.

If in this state, the Transport layer shall receive the FIS and place the contents of the FIS into the device registers if it is determined that the FIS was received without error.

Transition DTCMD0:1, if the entire FIS has been received from the Link layer without error, the Transport layer shall indicate to the Application layer that a command FIS was received and make a transition to the DTI0: DT_DeviceIdle state.

Transition DTCMD0:2, if the entire FIS has been received from the Link layer and an error has been detected, status shall be sent to the Link layer. The Transport layer shall make a transition to the DTI0:DT_DeviceIdle state.

Transition DTCMD0:3, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the DTI0: DT_DeviceIdle state.

10.7.10 Device transport decomposes Data (Host to Device) FIS state machine

This protocol receives a Data - Host to Device FIS (see Figure 303).

DTDATAO0: DT_DATAOFIS	Prepare to receive data.	
Unconditional	→	DT_DATAOREC

DTDATAO1: DT_DATAOREC	Place received data Dword into data FIFO and signal Link to continue transfer.	
1. Transfer not complete.	→	DT_DATAOREC
2. Transfer complete.	→	DT_DeviceIdle
3. Abort Transfer from Application layer.	→	DT_DeviceAbort
4. Notification of illegal transition error received from Link layer.	→	DT_DeviceIdle

DTDATAO2: DT_DeviceAbort	Signal Link to abort transfer.	
1. Transfer not complete.	→	DT_DATAOREC
2. Transfer complete.	→	DT_ DeviceIdle

Figure 303 – Device transport data host to device state machine

DTDATAO0: DT_DATAOFIS state, this state is entered if the Link layer has indicated that a FIS is being received and that the Transport layer has determined the FIS is of Data - Host to Device type.

If in this state, the Transport layer shall prepare to receive the data.

Transition DTDATAO0:1, if ready to receive the data, the Transport layer shall make a transition to the DTDATAO1: DT_DATAOREC state.

DTDATAO1: DT_DATAOREC state, this state is entered if the Transport layer is ready to receive the data.

If in this state, the Transport layer shall wait for the Link layer to indicate the transfer is complete.

Transition DTDATAO1:1, if Link layer has not indicated that the end of the FIS has been reached, the Transport layer shall transition to the DTDATAO1: DT_DATAOREC state.

Transition DTDATAO1:2, if the Link layer indicates that the end of the FIS has been reached, the Transport layer shall transition to the DTI0: DT_DeviceIdle state.

Transition DTDATAO1:3, if the Application layer indicates that the FIS is to be aborted, the Transport layer shall transition to the DTDATAO2: DT_DeviceAbort state.

Transition DTDATAO1:4, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the DTI0: DT_DeviceIdle state.

DTDATAO2: DT_DeviceAbort state, this state is entered if the Application layer indicates that the current transfer is to be aborted.

If in this state, the Transport layer shall signal the Link layer to Abort the incoming transmission and return to either DT_DATAOREC or DT_DeviceIdle, depending upon the current state of the FIFO. If the abort occurs coincident with an end of transfer indication from the Link layer, then the transition to DTI0: DT_DeviceIdle is also accommodated. After issuing an Abort, the Transport layer returns to normal data transfer, and awaits the end of transfer indication from the Link layer.

Transition DTDATAO2:1, inform Link layer to issue an abort. If transfer is not complete, the Transport layer shall transition to the DTDATAO1: DT_DATAOREC state.

Transition DTDATAO2:2, inform Link layer to issue an abort. If the Link layer indicates that the end of the FIS has been reached, the Transport layer shall transition to the DTI0: DT_DeviceIdle state.

10.7.11 Device transport decomposes DMA Setup – Host to Device state machine

This protocol receives a DMA Setup – Host to Device FIS, passes received DMA Setup content, and notification of FIS receipt to the Application layer (see Figure 304).

DTSTP0: DT_DMASTUPHDFIS	Receive a DMA Setup – Host to Device FIS.	
1. FIS transfer complete, and no error detected.	→	DT_DeviceIdle
2. FIS transfer complete, and error detected.	→	DT_DeviceIdle
3. Notification of illegal transition error received from Link layer.	→	DT_DeviceIdle

Figure 304 – Device transport DMA setup host to device state machine

DTSTP0: DT_DMASTUPHDFIS state, this state is entered if the receipt of a DT_DMASTUP FIS is recognized.

Transition DTSTP0:1, if the entire FIS has been received from the Link layer without error, the Transport layer shall indicate to the Application layer that a DMA Setup – Host to Device FIS was received and make a transition to the DTI0: DT_DeviceIdle state.

Transition DTSTP0:2, if the entire FIS has been received from the Link layer and an error has been detected, status shall be sent to the Link layer. The Transport layer shall make a transition to the DTI0: DT_DeviceIdle state.

Transition DTSTP0:3, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the DTI0: DT_DeviceIdle state.

10.7.12 Device transport decomposes a BIST Activate FIS state machine

This protocol receives a FIS that instructs the device to enter one of several Built-in Self-test modes that cause the device to retransmit the data it receives (see Figure 305). If the mode is supported, the Device's Application layer places both the transmit and receive portions of the Transport layer, Link layer, or Phy layer into appropriate state to perform the loopback operation.

DTRBIST1: DT_RcvBIST	Determine validity of loopback mode requested.	
1. Status checked, no error detected, and Loopback mode valid.	→	DT_BISTTrans1
2. Status checked, no error detected, and Loopback mode is invalid or not supported.	→	DT_DeviceIdle
3. Status checked, and error detected.	→	DT_DeviceIdle
4. Notification of illegal transition error received from Link layer.	→	DT_DeviceIdle

DTRBIST2: DT_BISTTrans1	Notify Application layer of desired BIST modes.	
1. Unconditional	→	DT_DeviceIdle

Figure 305 – Device transport BIST state machine

DTRBIST1: DT_RcvBIST state, this state is entered if the Link layer has indicated that a FIS is being received and the Transport layer has determined that a BIST Activate FIS is being received.

If in this state, the Transport layer shall determine the validity of the loopback request.

Transition DTRBIST1:1, if no reception error is detected and the FIS contents indicate a form of loopback request that is supported by the device, the Transport layer shall make a transition to the DTRBIST2: DT_BISTTrans1 state.

Transition DTRBIST1:2, if no reception error is detected and the FIS contents indicate a form of loopback request that is not supported by the device, the Transport layer shall make a transition to the DTI0:DT_DeviceIdle state.

Transition DTRBIST1:3, if a reception error is indicated, the Transport layer shall make a transition to the DTI0:DT_DeviceIdle state.

Transition DTRBIST1:4, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the DTI0: DT_DeviceIdle state.

DTRBIST2: DT_BISTTrans1 state, this state is entered if the Transport layer has determined that a valid BIST Activate FIS has been received.

Having received a valid FIS, the Transport layer informs the Application layer that it should place the Transport, Link, and Phy layers into the appropriate modes to loop the received data back to the transmitter. The method that this is performed is not defined by this specification.

Transition DTRBIST2:1: If the Application layer has been notified, the Transport layer shall transition to the DTI0:DT_DeviceIdle state.

11 Device command layer protocol

11.1 Device command layer protocol overview

In the following Device command layer protocols, if the host sends COMRESET before the device has completed processing a command layer protocol, then the device shall start processing the COMRESET protocol from the beginning. If the device receives a Register Host to Device FIS with C bit cleared to zero and the SRST bit set to one before the device has completed processing a command layer protocol, then the device shall start processing its software reset protocol from the beginning.

SYNC Escape is used by a host or device to bring the link back to a known state, and may be used for vendor specific recovery of error or hang conditions. After a SYNC Escape is performed, a software reset may be necessary prior to issuing the next command to the device.

11.2 Power-on and COMRESET protocol

If the host sends a hardware reset (power-on reset or COMRESET), regardless of the device power management mode (e.g., SLEEP, STANDBY) or the current Device command layer state, then the device shall perform the hardware reset protocol (see Figure 306). Assertion of hardware reset is associated with entry into state DP1:DR_Reset within the Phy state machine.

DHR0: Hardware_reset_asserted	Wait.	
1. Hardware reset negated.	→	DHR1: Execute_diagnostics

DHR1: Execute_diagnostics	Initialize hardware and process diagnostics.	
1. Initialization and diagnostics completed successfully.	→	DHR2: Send_good_status
2. Initialization completed and diagnostics failed.	→	DHR3: Send_bad_status

DHR2: Send_good_status	Request transmission of Register Device to Host FIS with good status.	
1. Register Device to Host FIS transmitted.	→	DI0: Device_idle

DHR3: Send_bad_status	Request transmission of Register Device to Host FIS with bad status.	
1. Register Device to Host FIS transmitted.	→	DI0: Device_idle

Figure 306 – Device command layer reset state machine

DHR0: Hardware_reset_asserted, this state is entered if the Transport layer indicates that a hardware reset (power-on reset or COMRESET) is asserted.

If in this state, the device awaits the negation of a hardware reset that is associated with exit from state DP1:DR_Reset within the Phy state machine.

Transition DHR0:1, if the Transport layer indicates that hardware reset has been negated, the device shall transition to the DHR1: Execute_diagnostics state.

DHR1: Execute_diagnostics, this state is entered if the Transport layer indicates that the COMRESET signal has been negated.

If in this state, the device initializes the device hardware and performs its power-up diagnostics.

Transition DHR1:1, if the device hardware has been initialized and the power-up diagnostics successfully completed, the device shall transition to the DHR2: Send_good_status state.

Transition DHR1:2, if the device hardware has been initialized and the power-up diagnostics failed, the device shall transition to the DHR3: Send_bad_status state.

DHR2: Send_good_status, this state is entered if the device hardware has been initialized and the power-up diagnostics successfully completed.

If in this state, the device requests that the Transport layer transmit a Register Device to Host FIS. If the device is an ATA device that is not a Host Managed Zoned device, the register signature shall be as defined in Figure 307.

COUNT field (7:0)	01h
LBA field (7:0)	01h
LBA field (15:8)	00h
LBA field (23:16)	00h
DEVICE field (7:0)	na
ERROR field (7:0)	01h
STATUS field (7:0)	00h..70h [a]
[a] Setting of bits 6:4 in the Status register are device specific.	

Figure 307 – DHR2: Send_good_status, ATA device not a Host Managed Zoned device implemented

If the device is an ATAPI device, the register signature shall be as defined in Figure 308.

COUNT field (7:0)	01h
LBA field (7:0)	01h
LBA field (15:8)	14h
LBA field (23:16)	EBh
DEVICE field (7:0)	na
ERROR field (7:0)	01h
STATUS field (7:0)	00h

Figure 308 – DHR2: Send_good_status, ATAPI device implemented

If the device is an ATA device that is a Host Managed Zoned device (see ZAC), the register signature shall be as defined in Figure 309.

COUNT field (7:0)	01h
LBA field (7:0)	01h
LBA field (15:8)	CDh
LBA field (23:16)	ABh
DEVICE field (7:0)	na
ERROR field (7:0)	01h
STATUS field (7:0)	00h..70h [a]
[a] Setting of bits 6:4 in the Status register are device specific.	

Figure 309 – DHR2: Send_good_status, ATAPI Host Managed Zoned device implemented

Transition DHR2:1, if the Transport layer indicates that the Register Device to Host FIS has been transmitted, the device shall transition to the DI0: Device_Idle state.

DHR3: Send_bad_status, this state is entered if the device hardware has been initialized and the power-up diagnostics failed.

If in this state, the device requests that the Transport layer transmit a Register Device to Host FIS. If the device is an ATA device that is not a Host Managed Zoned device, the register signature shall be as defined in Figure 310.

COUNT field (7:0)	01h
LBA field (7:0)	01h
LBA field (15:8)	00h
LBA field (23:16)	00h
DEVICE field (7:0)	na
ERROR field (7:0)	00h, 02h..7Fh
STATUS field (7:0)	00h..70h [a]
[a] Setting of bits 6:4 in the Status register are device specific.	

Figure 310 – DHR3: Send_bad_status, ATA device not a Host Managed Zoned device implemented

If the device is an ATAPI device, the register signature shall be as defined in Figure 311.

COUNT field (7:0)	01h
LBA field (7:0)	01h
LBA field (15:8)	14h
LBA field (23:16)	EBh
DEVICE field (7:0)	na
ERROR field (7:0)	00h, 02h..7Fh
STATUS field (7:0)	00h

Figure 311 – DHR3: Send_bad_status, ATAPI device implemented

If the device is an ATA device that is a Host Managed Zoned device (see ZAC), the register signature shall be as defined in Figure 312.

COUNT field (7:0)	01h
LBA field (7:0)	01h
LBA field (15:8)	CDh
LBA field (23:16)	ABh
DEVICE field (7:0)	na
ERROR field (7:0)	00h, 02h..7Fh
STATUS field (7:0)	00h..70h [a]
[a] Setting of bits 6:4 in the Status register are device specific.	

Figure 312 – DHR3: Send_bad_status, ATA Host Managed Zoned device implement

Transition DHR3:1, if the Transport layer indicates that the Register Device to Host FIS has been transmitted, the device shall transition to the DI0: Device_Idle state.

11.3 Device idle protocol

The state machine below describes the idle protocol for a device (see Figure 313). States and transitions utilized if NCQ or ATA Tagged Command Queuing commands are implemented have a footnote.

DI0: Device_idle	Wait.	
1. FIS receipt	→	DI1: Check_FIS
2. Ready to complete released command. [b]	→	DI4: Set_service
3. Ready to receive data for WRITE FPDMA QUEUED or SEND FPDMA QUEUED command and FIS receipt not indicated and no error encountered. [b]	→	DFPDMAQ4: DataPhase_ PreWriteSetup
4. Ready to transmit data for READ FPDMA QUEUED or RECEIVE FPDMA QUEUED command and FIS receipt not indicated and no error encountered. [b]	→	DFPDMAQ3: DataPhase_ ReadSetup
5. One or more FPDMA QUEUED or NCQ NON-DATA commands completed successfully and FIS receipt not indicated and no error encountered. [b]	→	DFPDMAQ10: SendStatus [a]
6. FPDMA QUEUED or NCQ NON-DATA command terminated with failure and FIS receipt not indicated. [b]	→	DFPDMAQ11: ERROR
7. Asynchronous Notification is enabled and event has occurred that requires notification and NotifyPending = 0 and FIS receipt not indicated.	→	AN0: Notify_host
[a] This condition may be true simultaneously with condition 3 or 4. Devices implementing status aggregation may select any of the transitions 3, 4, or 5 if their conditions evaluate to true. Devices not implementing status aggregation shall prioritize transition 5 over transitions 3 and 4. [b] States and transitions utilized if NCQ or ATA Tagged Command Queuing commands are implemented.		

DI1: Check_FIS	Check_ FIS Type and C bit.	
1. Register type, C bit cleared to zero, and SRST set to one.	→	DSR0: Software_reset_ asserted
2. Register type, C bit cleared to zero, and SRST cleared to zero.	→	DI0: Device_idle [a]
3. Register type and C bit set to one.	→	DI2: Check_command
4. DMA Setup FIS Received.	→	DI0: Device_idle
5. Unexpected FIS Type.	→	DI0: Device_idle
[a] A Register Device to Host FIS shall not be sent in response to the received Register Host to Device FIS.		

Figure 313 – Device command layer idle state machine (part 1 of 3)

DI2: Check_command [a]	Check the command to determine required command protocol. If asynchronous notification is supported, then NotifyPending is cleared to zero.	
1. Non-data command protocol and no native queued command outstanding.	→	DND0: Non-data
2. PIO DATA-IN command protocol and no native queued command outstanding.	→	DPIOI0: PIO_in
3. PIO DATA-OUT command protocol and no native queued command outstanding.	→	DPIOO0: PIO_out
4. READ DMA command protocol and no native queued command outstanding.	→	DDMAI0: DMA_in
5. WRITE DMA command protocol and no native queued command outstanding.	→	DDMAO0: DMA_out
6. PACKET command protocol and no native queued command outstanding.	→	DPKT0: PACKET
7. READ DMA QUEUED command protocol and no native queued command outstanding. [b]	→	DDMAQI0: DMA_queued_in
8. WRITE DMA QUEUED command protocol and no native queued command outstanding. [b]	→	DDMAQO0: DMA_queued_out
9. EXECUTE DEVICE DIAGNOSTIC command protocol and no native queued command outstanding.	→	DEDD0: Execute_device_diag
10. DEVICE RESET command protocol.	→	DDR0: Device_reset
11. Command not implemented and no native queued command outstanding.	→	DI3: No_command
12. SERVICE command protocol and no native queued command outstanding. [b]	→	DI5: Service_test
13. FPDMA QUEUED command protocol. [b]	→	DFPDMAQ1: AddCommand_ ToQueue
14. Non-NCQ command and not DEVICE RESET and native queued command(s) outstanding. [b]	→	DFPDMAQ12: BrokenHost_ ClearBusy
[a] This state shows transitions for all commands. If a device does not implement any particular command, then that transition should not be processed. [b] States and transitions utilized if NCQ or ATA Tagged Command Queuing commands are implemented.		

DI3: No_command	Request transmission of Register Device to Host FIS with ABRT bit set to one.	
1. FIS transmission complete.	→	DI0: Device_idle

DI4: Set_service [a]	Request transmission of Set Device Bits FIS with SERV set.	
1. FIS transmission complete.	→	DI0: Device_idle
[a] States and transitions utilized if NCQ or ATA Tagged Command Queuing commands are implemented.		

Figure 313 – Device command layer idle state machine (part 2 of 3)

DI5: Service_test [a]	Test command to see if Register Device to Host FIS is needed to set the DRQ bit and set Tag.	
1. PACKET PIO data-in or PACKET PIO data-out.	→	DI7: Service_decode
2. Other.	→	DI6: Service_send_tag
[a] States and transitions utilized if NCQ or ATA Tagged Command Queuing commands are implemented.		

DI6: Service_send_tag [a]	Request transmission of Register Device to Host FIS with the BSY bit cleared to zero, the DRQ bit set to one, and appropriate Tag	
1. FIS transmission complete.	→	DI7: Service_decode
[a] States and transitions utilized if NCQ or ATA Tagged Command Queuing commands are implemented.		

DI7: Service_decode [a]	Check command type to be serviced.	
1. PACKET PIO data-in.	→	DPKT4: PACKET_PIO_in
2. PACKET PIO data-out.	→	DPKT6: PACKET_PIO_out
3. PACKET DMA data-in.	→	DPKT9: PACKET_DMA_in
4. PACKET DMA data-out.	→	DPKT11: PACKET_DMA_out
5. READ DMA QUEUED.	→	DDMAQI1: Send_data
6. WRITE DMA QUEUED.	→	DDMAQO1: Send_DMA_activate
[a] States and transitions utilized if NCQ or ATA Tagged Command Queuing commands are implemented.		

Figure 313 – Device command layer idle state machine (part 3 of 3)

DI0: Device_Idle, this state is entered if the device has completed the processing of a command protocol, a COMRESET protocol, a software reset protocol, or a queued command has been released.

If in this state, the device is awaiting a command. If queuing is supported, the device may be waiting to acquire data or establish buffer space to complete a queued command.

Transition DI0:1, if the device receives a FIS from the Transport layer, the device shall transition to the DI1: Check_FIS state.

*** Transition DI0:2,** if the device is ready to complete the data transfer for a queued command, the device shall transition to the DI4: Set_service state.

*** Transition DI0:3,** if the device is ready to receive the data for a WRITE FPDMA QUEUED or a SEND FPDMA QUEUED command, the device shall transition to the DFPDMAQ4: DataPhasePreWriteSetup state. This condition also applies for the case where non-zero buffer offsets are used to complete a previous partial data transfer.

*** Transition DI0:4,** if the device is ready to transmit the data for a READ FPDMA QUEUED or a RECEIVE FPDMA QUEUED command, the device shall transition to the DFPDMAQ3:

DataPhaseReadSetup state. This condition also applies for the case where non-zero buffer offsets are used to complete a previous partial data transfer.

*** Transition DI0:5,** if the device has successfully completed a FPDMA QUEUED or an NCQ NON-DATA command, the device shall transition to the DFPDMAQ10: SendStatus state.

*** Transition DI0:6,** if the device has encountered an error in a command using the FPDMA QUEUED command protocol, the device shall transition to the DFPDMAQ11: ERROR state.

Transition DI0:7, if Asynchronous Notification is enabled:

a) an event requiring notification of the host has occurred;
b) the NotifyPending variable is cleared to zero; and
c) FIS receipt not indicated,

then the device shall transition to the AN0: Notify_host state.

DI1: Check_FIS state, this state is entered if the device receives a FIS from the Transport layer.

If in this state, the device shall check the FIS Type.

Transition DI1:1, if the FIS Type is a Register Host to Device FIS, the C bit in the FIS is cleared to zero, and the SRST bit in the FIS is set to one, the device shall transition to the DSR0: Software_reset_asserted state.

Transition DI1:2, if the FIS Type is a Register Host to Device FIS, the C bit in the FIS is cleared to zero, and the SRST bit in the FIS is cleared to zero, the device shall transition to the DI0: Device_idle state.

Transition DI1:3, if the FIS Type is a Register Host to Device FIS and the C bit in the FIS is set to one, the device shall transition to the DI2: Check_command state.

Transition DI1:4, if the FIS Type is a DMA Setup FIS, the device shall inform the Transport layer of the reception of the DMA Setup FIS, and transition to the DI0: Device_idle state.

Transition DI1:5, for any other FIS, the device shall transition to the DI0: Device_idle state.

DI2: Check_command state, this state is entered if the device recognizes that the received Register Host to Device FIS contains a new command.

This state shows transitions for all commands. If a device does not implement any particular command, then transition DI2:11 to state DI3:No_command shall be made.

If in this state, the device shall check the command protocol required by the received command and clears NotifyPending to zero if asynchronous notification is supported. Clearing NotifyPending to zero allows future asynchronous notification messages to be sent to the host.

Transition DI2:1, if the received command is a non-data transfer command, the device shall transition to the DND0: Non-data state.

Transition DI2:2, if the received command is a PIO DATA-IN command, the device shall transition to the DPIOI0: PIO_in state.

Transition DI2:3, if the received command is a PIO DATA-OUT command, the device shall transition to the DPIOO0: PIO_out state.

Transition DI2:4, if the received command is a READ DMA command, the device shall transition to the DDMAI0: DMA_in state.

Transition DI2:5, if the received command is a WRITE DMA command, the device shall transition to the DDMAO0: DMA_out state.

Transition DI2:6, if the received command is a PACKET command, the device shall transition to the DPKT0: PACKET state.

*** Transition DI2:7,** if the received command is a READ DMA QUEUED command, the device shall transition to the DDMAQI0: DMA_queued_in state.

*** Transition DI2:8,** if the received command is a WRITE DMA QUEUED command, the device shall transition to the DDMAQO0: DMA_queued_out state.

Transition DI2:9, if the received command is an EXECUTE DEVICE DIAGNOSTICS command, the device shall transition to the DEDD0: Execute_device_diag state.

Transition DI2:10, if the received command is a RESET DEVICE command, the device shall transition to the DDR0: Device_reset state.

Transition DI2:11, if the received command is not implemented by the device, the device shall transition to the DI3: No_command state.

*** Transition DI2:12,** if the received command is a SERVICE command, the device shall transition to the DI5: Service_test state.

*** Transition DI2:13,** if the received command uses the FPDMA QUEUED command protocol, the device shall transition to the DFPDMAQ1: AddCommandToQueue state.

*** Transition DI2:14,** if the received command is a non-NCQ command and there are native queued command(s) outstanding, an error has occurred and the device shall transition to the DFPDMAQ12: BrokenHost_ClearBusy state.

DI3: No_command state, this state is entered if the device recognizes that the received command is not implemented by the device.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS with register content as described in the command description in the ACS-4 standard and the Interrupt bit set to one.

Transition DI2:1, if the Transport layer has transmitted the Register Device to Host FIS, the device shall transition to the DI0: Device_idle state.

*** DI4: Set_service state,** this state is entered if ready to complete the data transfer for a queued command.

If in this state, the device shall request that the Transport layer transmit a Set Device Bits FIS with the SERV bit set to one in the Status register, and with all other bits in the Error and Status fields the same as the current contents of the respective registers, and the Interrupt bit set to one.

Transition DI4:1, if the Transport layer has transmitted the Set Device Bits FIS, the device shall transition to the DI0: Device_idle state.

*** DI5: Service_test state,** this state is entered if the SERVICE command has been received.

If in this state, the device shall determine the type of command that the device has requested service to complete. The PACKET command using the PIO protocol provides its own register

update to set the DRQ BIT and send the command tag, but other queued commands require a Register Device to Host FIS.

Transition DI5:1, if the command to be serviced is a PIO data-in or PIO DATA-OUT command, the device shall transition to the DI7: Service_decode state.

Transition DI5:2, if the command to be serviced is neither a PIO data-in nor a PIO DATA-OUT command, the device shall transition to the DI6: Service_send_tag state.

*** DI6: Service_send_tag state,** this state is entered if the SERVICE command has been received and sending a Register Device to Host FIS is necessary for the command being serviced.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS with register contents, including the desired command tag, as described in the command description of the ACS-4 standard for the command being serviced.

Transition DI6:1, if the Transport layer has transmitted the Register Device to Host FIS, the device shall transition to theDI7: Service_decode state.

*** DI7: Service_decode state,** this state is entered if a Register Device to Host FIS has been transmitted, if necessary to send the register contents, including the desired command tag, in response to a SERVICE command.

If in this state, the device shall again determine the type of command that the device has requested service to complete, and branch to that command's data transfer and completion.

Transition DI7:1, if the command to be serviced is a PIO DATA-IN command, the device shall transition to the DPKT4: PACKET_PIO_in state.

Transition DI7:2, if the command to be serviced is a PIO DATA-OUT command, the device shall transition to the DPKT6: PACKET_PIO_out state.

Transition DI7:3, if the command to be serviced is a DMA DATA-IN command, the device shall transition to the DPKT9: PACKET_DMA_in state.

Transition DI7:4, if the command to be serviced is a DMA DATA-OUT command, the device shall transition to the DPKT11: PACKET_DMA_out state.

Transition DI7:5, if the command to be serviced is a READ DMA QUEUED command, the device shall transition to the DDMAQI1: Send_data state.

Transition DI7:6, if the command to be serviced is a WRITE DMA QUEUED command, the device shall transition to the DDMAQO1: Send_DMA_activate state.

11.4 Software reset protocol

If the host sends a Register Host to Device FIS with a one in the SRST bit position (see AST-2) of the Device Control register byte, regardless of the device power management mode (e.g., SLEEP, STANDBY), the device shall perform the software reset protocol (see Figure 314).

DSR0: Software_reset_asserted	Begin initialization and diagnostic processing.	
1. Software reset negated.	→	DSR1: Execute_diagnostics
2. Software reset asserted.	→	DSR0: Software_reset_ asserted

DSR1: Execute_diagnostics	Complete Initialization and diagnostics processing.	
1. Initialization and diagnostics completed successfully.	→	DSR2: Send_good_status
2. Initialization completed and diagnostics failed.	→	DSR3: Send_bad_status

DSR2: Send_good_status	Request transmission of Register Device to Host FIS with good status.	
1. FIS transmitted.	→	DI0: Device_idle

DSR3: Send_bad_status	Request transmission of Register Device to Host FIS with bad status.	
1. FIS transmitted.	→	DI0: Device_idle

Figure 314 – Device command layer software reset state machine

DSR0: Software_reset_asserted, this state is entered if a Register Host to Device FIS is received with the C bit in the FIS cleared to zero and the SRST bit set to one in the Device Control register.

If in this state, the device begins its initialization and diagnostics processing and awaits the clearing of the SRST bit.

Transition DSR0:1, if a Register Host to Device FIS is received with the C bit in the FIS cleared to zero and the SRST bit cleared to zero in the Device Control register, the device shall transition to the DSR1: Execute_diagnostics state.

Transition DSR0:2, if a Register FIS is received with the C bit in the FIS set to one, or the SRST bit set to one in the Device Control register, the device shall transition to the DSR0: Software_reset_asserted state.

DSR1: Execute_diagnostics, this state is entered if a Register Host to Device FIS is received with the C bit in the FIS cleared to zero and the SRST bit cleared to zero in the Device Control register.

If in this state, the device completes initialization and processing of its diagnostics.

Transition DSR1:1, if the device has been initialized and the diagnostics successfully completed, the device shall transition to the DSR2: Send_good_status state.

Transition DSR1:2, if the device has been initialized and the diagnostics failed, the device shall transition to the DSR3: Send_bad_status state.

DSR2: Send_good_status, this state is entered if the device has been initialized and the diagnostics successfully completed.

If in this state, the device requests that the Transport layer transmit a Register Device to Host FIS. If the device does not implement the PACKET command feature set the register content shall be as defined in Figure 315.

COUNT field (7:0)	01h
LBA field (7:0)	01h
LBA field (15:8)	00h
LBA field (23:16)	00h
DEVICE field (7:0)	na
ERROR field (7:0)	01h
STATUS field (7:0)	00h..70h [a]
[a] Setting of bits 6:4 in the Status register are device specific.	

Figure 315 – DSR2: Send_good_status, PACKET command feature set not implemented

If the device implements the PACKET command feature set, the register signature shall be as defined in Figure 316.

COUNT field (7:0)	01h
LBA field (7:0)	01h
LBA field (15:8)	14h
LBA field (23:16)	EBh
DEVICE field (7:0)	na
ERROR field (7:0)	01h
STATUS field (7:0)	00h

Figure 316 – DSR2: Send_good_status, PACKET command feature set is implemented

Transition DSR2:1, if the Transport layer indicates that the Register Device to Host FIS has been transmitted, the device shall transition to the DI0: Device_Idle state.

DSR3: Send_bad_status, this state is entered if the device has been initialized and the diagnostics failed.

If in this state, the device requests that the Transport layer transmit a Register Device to Host FIS. If the device does not implement the PACKET command feature set, the register content shall be as defined in Figure 317.

COUNT field (7:0)	01h
LBA field (7:0)	01h
LBA field (15:8)	00h
LBA field (23:16)	00h
DEVICE field (7:0)	na
ERROR field (7:0)	00h, 02h..7Fh
STATUS field (7:0)	00h..70h [a]
[a] Setting of bits 6:4 in the Status register are device specific.	

Figure 317 – DSR3: Send_bad_status, PACKET command feature set not implemented

If the device implements the PACKET command feature set, the register content shall be as defined in Figure 318.

COUNT field (7:0)	01h
LBA field (7:0)	01h
LBA field (15:8)	14h
LBA field (23:16)	EBh
DEVICE field (7:0)	na
ERROR field (7:0)	00h, 02h..7Fh
STATUS field (7:0)	00h

Figure 318 – DSR3: Send_bad_status, PACKET command feature set is implemented

Transition DSR3:1, if the Transport layer indicates that the Register Device to Host FIS has been transmitted, the device shall transition to the DI0: Device_Idle state.

11.5 EXECUTE DEVICE DIAGNOSTIC command protocol

If the host sends COMRESET before the device has completed processing the EXECUTE DEVICE DIAGNOSTIC protocol, then the device shall immediately start processing the COMRESET protocol form the beginning (see Figure 319). If the host asserts SRST in the Device Control register before the device has completed processing the EXECUTE DEVICE DIAGNOSTIC protocol, then the device shall immediately start processing its software reset protocol from the beginning.

DEDD0: Execute_device_diag	Process diagnostics.	
1. Diagnostics completed successfully.	→	DEDD1: Send_good_status
2. Diagnostics failed.	→	DEDD2: Send_bad_status

DEDD1: Send_good_status	Request transmission of Register Device to Host FIS with good status.	
1. FIS transmitted.	→	DI0: Device_idle

DEDD2: Send_bad_status	Request transmission of Register Device to Host FIS with bad status.	
1. FIS transmitted.	→	DI0: Device_idle

Figure 319 – Device command layer diagnostic state machine

DEDD0: Execute_device_diag, this state is entered if an EXECUTE DEVICE DIAGNOSTIC command is received.

If in this state, the device performs its diagnostics.

Transition DEDD0:1, if the device successfully completed the diagnostics, the device shall transition to the DEDD1: Send_good_status state.

Transition DEDD1:2, if the device has failed the diagnostics, the device shall transition to the DEDD2: Send_bad_status state.

DEDD1: Send_good_status, this state is entered if the device has successfully completed the diagnostics.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS to the host, with the Interrupt bit set to one. If the device does not implement the PACKET command feature set, the register content shall be as defined in Figure 320.

COUNT field (7:0)	01h
LBA field (7:0)	01h
LBA field (15:8)	00h
LBA field (23:16)	00h
DEVICE field (7:0)	na
ERROR field (7:0)	01h
STATUS field (7:0)	00h..70h [a]
[a] Setting of bits 6:4 in the Status register are device specific.	

Figure 320 – DEDD1: Send_good_status, PACKET command feature set not implemented

If the device implements the PACKET command feature set, the register content shall be as defined in Figure 321.

COUNT field (7:0)	01h
LBA field (7:0)	01h
LBA field (15:8)	14h
LBA field (23:16)	EBh
DEVICE field (7:0)	na
ERROR field (7:0)	01h
STATUS field (7:0)	00h

Figure 321 – DEDD1: Send_good_status, PACKET command feature set is implemented

Transition DEDD1:1, if the Transport layer indicates that the Register Device to Host FIS has been transmitted, the device shall transition to the DI0: Device_Idle state.

DEDD2: Send_bad_status, this state is entered if the device has been initialized and the diagnostics failed.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS to the host, with the Interrupt bit set to one. If the device does not implement the PACKET command feature set, the register content shall be as defined in Figure 322.

COUNT field (7:0)	01h
LBA field (7:0)	01h
LBA field (15:8)	00h
LBA field (23:16)	00h
DEVICE field (7:0)	na
ERROR field (7:0)	00h, 02h..7Fh
STATUS field (7:0)	00h..70h [a]
[a] Setting of bits 6:4 in the Status register are device specific.	

Figure 322 – DEDD2: Send_bad_status, PACKET command feature set not implemented

If the device implements the PACKET command feature set, the register content shall be as defined in Figure 323.

COUNT field (7:0)	01h
LBA field (7:0)	01h
LBA field (15:8)	14h
LBA field (23:16)	EBh
DEVICE field (7:0)	na
ERROR field (7:0)	00h, 02h..7Fh
STATUS field (7:0)	00h

Figure 323 – DEDD2: Send_bad_status, PACKET command feature set implemented

Transition DEDD2:1, if the Transport layer indicates that the Register Device to Host FIS has been transmitted, the device shall transition to the DI0: Device_Idle state.

11.6 DEVICE RESET command protocol

If the host sends COMRESET before the device has completed processing the DEVICE RESET protocol, then the device shall immediately start processing the COMRESET protocol from the beginning (see Figure 324). If the host asserts SRST in the Device Control register before the device has completed processing the DEVICE RESET protocol, then the device shall immediately start processing its software reset protocol from the beginning.

DDR0: Device_reset	Stop processing and background activity.	
1. Activity ceased.	→	DDR1: Send_good_status

DDR1: Send_good_status	Request transmission of Register Device to Host FIS with good status.	
1. FIS transmitted.	→	DI0: Device_idle

Figure 324 – Device command layer reset state machine

DDR0: Device_reset, this state is entered if a DEVICE RESET command is received.

If in this state, the device stops any processing or activity in progress.

Transition DDR0:1, if the device has ceased any processing or activity and has completed its internal diagnostics, the device shall transition to the DDR1: Send_good_status state.

DDR1: Send_good_status, this state is entered if the device has been initialized and the diagnostics successfully completed.

If in this state, the device requests that the Transport layer transmit a Register Device to Host FIS.

The register content shall be as defined in Figure 325.

COUNT field (7:0)	01h
LBA field (7:0)	01h
LBA field (15:8)	14h
LBA field (23:16)	EBh
DEVICE field (7:0)	na
ERROR field (7:0)	01h
STATUS field (7:0)	00h

Figure 325 – Device command layer non-data state machine

Transition DDR1:1, if the Transport layer indicates that the Register Device to Host FIS has been transmitted, the device shall transition to the DI0: Device_Idle state.

11.7 Non-data command protocol

Processing of this class of command involves no data transfer (see Figure 326). See the NOP command description and the SLEEP command description in ACS-4 for additional protocol requirements.

DND0: Non-data	Process Non-data command.	
1. Command processing complete.	→	DND1: Send_status

DND1: Send_status	Request transmission of a Register Device to Host FIS.	
1. FIS transmission complete.	→	DI0: Device_idle

Figure 326 – Device command layer non-data state machine

DND0: Non-data State, this state is entered if a received command is a non-data command.

If in this state, the device shall perform the requested command if supported.

Transition DND0:1, if command processing completes, the device shall transition to the DND1: Send_status state.

DND1: Send_status State, this state is entered if the processing of the non-data command has been completed.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS with register content as described in the command description in the ACS-4 standard and the Interrupt bit set to one.

Transition DND1:1, if the FIS has been transmitted, then the device shall transition to the DI0: Device_idle state.

11.8 PIO DATA-IN command protocol

Processing of this class of command includes the PIO transfer of one or more blocks of data from the device to the host (see Figure 327).

DPIOI0: PIO_in	Prepare a DRQ data block for transfer to the host.		
	1. DRQ data block ready to transfer and no error encountered.	→	DPIOI1: Send_PIO_setup
	2. Error encountered during command processing.	→	DPIOI3: Error_status

DPIOI1: Send_PIO_setup	Request transmission of a PIO Setup FIS to host.		
	1. PIO Setup FIS transmitted.	→	DPIOI2: Transmit_data

DPIOI2: Transmit_data	Request transmission of a Data FIS to host.		
	1. Data FIS transmitted, no more data transfer required for this command.	→	DI0: Device_idle
	2. Data FIS transmitted, more data transfer required for this command, or 2 048 Dwords transmitted.	→	DPIOI0: PIO_in

DPIOI3: Error_status	Request transmission of a Register Device to Host FIS.		
	1. FIS transmission complete.	→	DI0: Device_idle

Figure 327 – Device command layer PIO data-in state machine

DPIOI0: PIO_in State, this state is entered if the device receives a PIO DATA-IN command or the transmission of one or more additional DRQ data blocks is required to complete the command.

If in this state, device shall prepare a DRQ data block for transfer to the host.

Transition DPIOI0:1, if the device has a DRQ data block ready to transfer and no error was encountered, the device shall transition to the DPIOI1: Send_PIO_setup state.

Transition DPIOI0:2, if the device has encountered an error during command processing, the device shall transition to the DPIOI3: Error_status state.

DPIOI1: Send_PIO_setup, this state is entered if the device is ready to transmit a DRQ data block to the host.

If in this state, the device shall request that the Transport layer transmit a PIO Setup FIS. The initial status shall have the BSY bit cleared to zero, the DRQ bit set to one, and with register content as described in the command description in the ACS-4 standard. The Interrupt bit shall be set to one. If this is the last DRQ data block requested by the command, the ending status shall have the BSY bit cleared to zero and the DRQ bit cleared to zero. If this is not the last data block requested by the command, the ending status shall have the BSY bit set to one and the DRQ bit cleared to zero.

Transition DPIOI1:1, if the PIO Setup FIS has been transferred, the device shall transition to the DPIOI2: Transmit_data state.

DPIOI2: Transmit_data, this state is entered if the device has transmitted a PIO Setup FIS to the host.

If in this state, the device shall request that the Transport layer transmit a Data FIS containing the DRQ data block.

Transition DPIOI2:1, If the Data FIS has been transferred and all data requested by this command have been transferred, the device shall transition to the DI0: Device_idle.

Transition DPIOI2:2, if the Data FIS has been transferred but all data requested by this command has not been transferred, or the 2 048 Dword transfer limit has been reached, then the device shall transition to the DPIOI0: PIO_in state.

DPIOI3: Error_status, this state is entered if the device has encountered an error that causes the command to abort before completing the transfer of the requested data.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS with register content as described in the command description in the ACS-4 standard and the Interrupt bit set to one. In addition to the ACS-4 requirements, the device may set bit 7 of the ERROR field in the FIS to one if a CRC error was encountered in transmission of a previous FIS for this command.

Transition DPIOI3:1, if the FIS has been transmitted, the device shall transition to the DI0: Device_idle state.

11.9 PIO DATA-OUT command protocol

Processing of this class of command includes the PIO transfer of one or more blocks of data from the host to the device (see Figure 328).

DPIOO0: PIO_out	Prepare to receive DRQ data block transfer from the host.		
	1. Ready to receive DRQ data block transfer.	→	DPIOO1: Send_PIO_setup
	2. All DRQ data blocks received or command aborted due to error.	→	DPIOO3: Send_status

DPIOO1: Send_PIO_setup	Request transmission of a PIO Setup FIS to host.		
	1. PIO Setup FIS transmitted.	→	DPIOO2: Receive_data

DPIOO2: Receive_data	Receive Data FIS from the Transport layer.		
	1. Data FIS received.	→	DPIOO0: PIO_out

DPIOO3: Send_status	Request transmission of a Register Device to Host FIS.		
	1. FIS transmission complete.	→	DI0: Device_idle

Figure 328 – Device command layer PIO data-out state machine

DPIOO0: PIO_out State, this state is entered if the device receives a PIO DATA-OUT command or the receipt of one or more DRQ data blocks is required to complete this command.

If in this state, device shall prepare to receive a DRQ data block transfer from the host.

Transition DPIOO0:1, if the device is ready to receive a DRQ data block, the device shall transition to the DPIOO1: Send_PIO_setup state.

Transition DPIOO0:2, if the device has received all DRQ data blocks requested by this command or the device has encountered an error that causes the command to abort before completing the transfer of the requested data, then the device shall transition to the DPIOO3: Send_status state.

DPIOO1: Send_PIO_setup, this state is entered if the device is ready to receive a DRQ data block from the host.

If in this state, the device shall request that the Transport layer transmit a PIO Setup FIS. The initial status shall have the BSY bit cleared to zero and the DRQ bit set to one. If this is the first DRQ data block for this command, the Interrupt bit shall be cleared to zero. If this is not the first DRQ data block for this command, the Interrupt bit shall be set to one. The ending status shall have the BSY bit set to one and the DRQ bit cleared to zero. The byte count for the DRQ data block shall be indicated.

Transition DPIOO1:1, if the PIO Setup FIS has been transferred, the device shall transition to the DPIOO2: Receive_data state.

DPIOO2: Receive_data, this state is entered if the device has transmitted a PIO Setup FIS to the host.

If in this state, the device shall receive the requested Data FIS from the Transport layer.

Transition DPIOO2:1, if the Data FIS has been received, the device shall transition to the DPIOO0: PIO_out state.

DPIOO3: Send_status, this state is entered if the device has received all DRQ data blocks requested by this command or the device has encountered an error that causes the command to abort before completing the transfer of the requested data.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS with register content as described in the command description in the ACS-4 standard and the Interrupt bit set to one. In addition to the ACS-4 requirements, the device may set bit 7 of the ERROR field in the FIS to one if a CRC error was encountered in transmission of a previous FIS for this command.

Transition DPIOO3:1, if the FIS has been transmitted, the device shall transition to the DI0: Device_idle state.

11.10 DMA DATA-IN command protocol

Processing of this class of command includes the transfer of one or more blocks of data from the device to the host using DMA transfer (see Figure 329).

DDMAI0: DMA_in	Prepare data for the transfer of a Data FIS.	
1. Data for Data FIS ready to transfer.	→	DDMAI1: Send_data
2. Command completed or aborted due to error.	→	DDMAI2: Send_status

DDMAI1: Send_data	Request transmission of a Data FIS to host.	
1. Data FIS transmitted. No more data transfer required for this command, or 2 048 Dwords transmitted.	→	DDMAI0: DMA_in

DDMAI2: Send_status	Request transmission of a Register Device to Host FIS.	
1. FIS transmitted.	→	DI0: Device_idle

Figure 329 – Device command layer DMA data in state machine

DDMAI0: DMA_in State, this state is entered if the device receives a DMA DATA-IN command or the transmission of one or more data FIS is required to complete the command.

If in this state, device shall prepare the data for transfer of a data FIS to the host.

Transition DDMAI0:1, if the device has the data ready to transfer a data FIS, the device shall transition to the DDMAI1: Send_data state.

Transition DDMAI0:2, if the device has transferred all of the data requested by this command or has encountered an error that causes the command to abort before completing the transfer of the requested data, then the device shall transition to the DDMAI2: Send_status state.

DDMAI1: Send_data, this state is entered if the device has the data ready to transfer a data FIS to the host.

If in this state, the device shall request that the Transport layer transmit a data FIS containing the data. The Device command layer shall request a Data FIS size of no more than 2 048 Dwords.

Transition DDMAI1:1, if the data FIS has been transferred, the device shall transition to the DDMAI0: DMA_in state.

DDMAI2: Send_status, this state is entered if the device has transferred all of the data requested by the command or has encountered an error that causes the command to abort before completing the transfer of the requested data.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS with register content as described in the command description in the ACS-4 standard and the Interrupt bit set to one.

Transition DDMAI2:1, if the FIS has been transmitted, the device shall transition to the DI0: Device_idle state.

11.11 DMA DATA-OUT command protocol

Processing of this class of command includes the transfer of one or more blocks of data from the host to the device using DMA transfer (see Figure 330). A single interrupt is issued at the completion of the successful transfer of all data required by the command.

DDMAO0: DMA_out	Prepare to receive a Data FIS from the host.	
1. Ready to receive Data FIS.	→	DDMAO1: Send_DMA_activate
2. All data requested for this command received or command aborted due to error.	→	DDMAO3: Send_status

DDMAO1: Send_DMA_activate	Request transmission of a DMA Activate FIS to host.	
1. DMA Activate FIS transmitted.	→	DDMAO2: Receive_data

DDMAO2: Receive_data	Receive Data FIS from the Transport layer.	
1. Data FIS received.	→	DDMAO0: DMA_out

DDMAO3: Send_status	Request transmission of a Register Device to Host FIS.	
1. FIS transmitted.	→	DI0: Device_idle

Figure 330 – Device command layer DMA data out state machine

DDMAO0: DMA_out State, this state is entered if the device receives a DMA DATA-OUT command or the receipt of one or more Data FIS is required to complete this command.

If in this state, device shall prepare to receive a Data FIS from the host.

Transition DDMAO0:1, if the device is ready to receive a Data FIS, the device shall transition to the DDMAO1: Send_DMA_activate state.

Transition DDMAO0:2, if the device has received all the data requested by this command or the device has encountered an error that causes the command to abort before completing the transfer of the requested data, then the device shall transition to the DDMAO3: Send_status state.

DDMAO1: Send_DMA_activate, this state is entered if the device is ready to receive a Data FIS from the host.

If in this state, the device shall request that the Transport layer transmit a DMA Activate FIS.

Transition DDMAO1:1, if the DMA Activate FIS has been transferred, the device shall transition to the DDMAO2: Receive_data state.

DDMAO2: Receive_data, this state is entered if the device transmitted a DMA Activate FIS to the host.

If in this state, the device shall receive the requested Data FIS from the Transport layer.

Transition DDMAO2:1, if the Data FIS has been received, the device shall transition to the DDMAO0: DMA_out state.

DDMAO3: Send_status, this state is entered if the device has received all the data requested by this command or the device has encountered an error that causes the command to abort before completing the transfer of the requested data.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS with register content as described in the command description in the ACS-4 standard and the Interrupt bit set to one.

Transition DDMAO3:1, if the FIS has been transmitted, the device shall transition to the DI0: Device_idle state.

11.12 PACKET protocol

The Device command layer PACKET protocol is defined in Figure 331.

DPKT0: PACKET	Request transmission of a PIO Setup FIS.	
1. FIS transmission complete.	→	DPKT1: Receive_command

DPKT1: Receive_command	Receive Data FIS containing command packet.	
1. FIS reception complete.	→	DPKT2: Check_command

DPKT2: Check_command	Determine the protocol required for the received command.	
1. Non-data command.	→	DPKT3: PACKET_non-data
2. PIO DATA-IN command.	→	DPKT4: PACKET_PIO_in
3. PIO DATA-OUT command.	→	DPKT6: PACKET_PIO_out
4. DMA DATA-IN command.	→	DPKT9: PACKET_DMA_in
5. DMA DATA-OUT command	→	DPKT11: PACKET_DMA_out

DPKT3: PACKET_non-data	Process Non-data command.	
1. Command processing complete.	→	DPKT14: Send_status

DPKT4: PACKET_PIO_in	Prepare a DRQ data block for transfer to the host.	
1. DRQ data block ready to transfer.	→	DPKT4a: PIO_in_setup
2. Transfer complete or command aborted due to error.	→	DPKT14: Send_status
3. DRQ data block is not ready for immediate transfer [a]	→	DPKT15: Release
[a] This transition is only utilized if queuing is implemented.		

DPKT4a: PIO_in_setup	Request transmission of a PIO Setup FIS to host.	
1. PIO Setup FIS transmitted.	→	DPKT5: Send_PIO_data

DPKT5: Send_PIO_data	Request transmission of a Data FIS to host.	
1. Data FIS transmitted.	→	DPKT4: PACKET_PIO_in

DPKT6: PACKET_PIO_out	Prepare to receive DRQ data block from the host.	
1. Ready to receive DRQ data block transfer.	→	DPKT7: PIO_out_setup
2. All DRQ data blocks received or command aborted due to error.	→	DPKT14: Send_status
3. Not ready to accept DRQ data block immediately. [a]	→	DPKT15: Release
[a] This transition is only utilized if queuing is implemented.		

Figure 331 – Device command layer packet state machine (part 1 of 2)

DPKT7: PIO_out_setup	Request transmission of a PIO Setup FIS to host.	
1. PIO Setup FIS transmitted.	→	DPKT8: Receive_PIO_data

DPKT8: Receive_PIO_data	Receive Data FIS from the Transport layer.	
1. Data FIS received.	→	DPKT6: PACKET_PIO_out

DPKT9: PACKET_DMA_in	Prepare data for the transfer of a Data FIS.	
1. Data for Data FIS ready to transfer.	→	DPKT10: Send_DMA_data
2. Command completed or aborted due to error.	→	DPKT14: Send_status
3. Data is not ready for immediate transfer. [a]	→	DPKT15: Release
[a] This transition is only utilized if queuing is implemented.		

DPKT10: Send_DMA_data	Request transmission of a Data FIS to host.	
1. Data FIS transmitted. No more data transfer required for this command, or 2 048 Dwords transmitted.	→	DPKT9: PACKET_DMA_IN

DPKT11: PACKET_DMA_out	Prepare to receive a Data FIS from the host.	
1. Ready to receive Data FIS.	→	DPKT12: Send_DMA_activate
2. All data requested for this command received or command aborted due to error.	→	DPKT14: Send_status
3. Not ready for immediate transfer. [a]	→	DPKT15: Release
[a] This transition is only utilized if queuing is implemented.		

DPKT12: Send_DMA_activate	Request transmission of a DMA Activate FIS to host.	
1. DMA Activate FIS transmitted.	→	DPKT13: Receive_DMA_data

DPKT13: Receive_DMA_data	Receive Data FIS from the Transport layer.	
1. Data FIS received.	→	DPKT11: PACKET_DMA_out

DPKT14: Send_status	Request transmission of a Register Device to Host FIS.	
1. FIS transmission complete.	→	DI0: Device_idle

DPKT15: Release [a]	Request transmission of a Register Device to Host FIS.	
1. FIS transmission complete.	→	DI0: Device_idle
[a] This state is only utilized if queuing is implemented.		

Figure 331 – Device command layer packet state machine (part 2 of 2)

DPKT0: PACKET, this state is entered if the device receives a PACKET command.

If in this state, the device shall request that the Transport layer transmit a PIO Setup FIS to acquire the command packet associated with this command. The initial status shall have the BSY bit cleared to zero and the DRQ bit set to one. The Interrupt bit shall be cleared to zero. The ending status shall have the BSY bit set to one and the DRQ bit cleared to zero. The byte count for the DRQ data block shall be indicated.

Transition DPKT0:1, if the PIO Setup FIS has been transferred, the device shall transition to the DPKT1: Receive_command state.

DPKT1: Receive_command, this state is entered if the device transmitted a PIO Setup FIS to the host to get the command packet.

If in this state, the device shall receive the requested Data FIS from the Transport layer.

Transition DPKT1:1, if the Data FIS has been received, the device shall transition to the DPKT2: Check_command state.

DPKT2: Check_command, this state is entered if the Data FIS containing the command packet has been received.

If in this state, the device shall determine the protocol for the command contained in the command packet.

Transition DPKT2:1, if the command is a non-data transfer command, the device shall transition to the DPKT3: PACKET_non-data state.

Transition DPKT2:2, if the command is a PIO data-in transfer command, the device shall transition to the DPKT4: PACKET_PIO_in state.

Transition DPKT2:3, if the command is a PIO data-out transfer command, the device shall transition to the DPKT6: PACKET_PIO_out state.

Transition DPKT2:4, if the command is a DMA data-in transfer command, the device shall transition to the DP9: PACKET_DMA_in state.

Transition DPKT2:5, if the command is a DMA data-out transfer command, the device shall transition to the DPKT11: PACKET_DMA_out state.

DPKT3: PACKET_non-data State, this state is entered if a received command is a non-data command.

If in this state, the device shall process the requested command.

Transition DPKT3:1, if command processing completes, the device shall transition to the DPKT14: Send_status state.

DPKT4: PACKET_PIO_in State, this state is entered if the device receives a PIO DATA-IN command or the transmission of one or more DRQ data blocks is required to complete the command.

If in this state, device shall prepare a DRQ data block for transfer to the host.

Transition DPKT4:1, if the device has a DRQ data block ready to transfer, the device shall transition to the DPKT4a: PIO_in_setup.

Transition DPKT4:2, if all of the data requested by this command has been transferred or the device has encountered an error that causes the command to abort before completing the transfer of the requested data, then the device shall transition to the DPKT14: Send_status state.

Transition DPKT4:3, if the device supports overlap and queuing and does not have a DRQ data block ready to transfer immediately, the device shall transition to the DPKT15: Release state. This transition is only utilized if queuing is implemented.

DPKT4a: PIO_in_setup, this state is entered if the device is ready to transfer a DRQ data block to the host.

If in this state, the device shall request that the Transport layer transmit a PIO Setup FIS. The initial status shall have the BSY bit cleared to zero and the DRQ bit set to one. The Interrupt bit shall be set to one. The ending status shall have the BSY bit set to one and the DRQ bit cleared to zero. The byte count for the DRQ data block shall be indicated.

Transition DPKT4a:1, if the PIO Setup FIS has been transferred, the device shall transition to the DPKT5:Send_PIO_data state.

DPKT5:Send_PIO_data, this state is entered if the device is ready to transfer a DRQ data block to the host.

If in this state, the device shall request that the Transport layer transmit a Data FIS containing the DRQ data block.

Transition DPKT5:1, if the Data FIS has been transferred, the device shall transition to the DPKT4: PACKET_PIO_in state.

DPKT6: PACKET_PIO_out State, this state is entered if the device receives a PIO DATA-OUT command or the receipt of one or more DRQ data blocks is required to complete the command.

If in this state, device shall prepare to receive a DRQ data block transfer from the host.

Transition DPKT6:1, if the device is ready to receive a DRQ data block transfer, the device shall transition to the DPKT7: PIO_out_setup state.

Transition DPKT6:2, if the device has received all DRQ data blocks requested by this command or the device has encountered an error that causes the command to abort before completing the transfer of the requested data, then the device shall transition to the DPKT14: Send_status state.

Transition DPKT6:3, if the device supports overlap and queuing and is not in a state that it is possible to accept a DRQ data block immediately, the device shall transition to the DPKT15: Release state. This transition is only utilized if queuing is implemented.

DPKT7: PIO_out_setup, this state is entered if the device is ready to receive a DRQ data block from the host.

If in this state, the device shall request that the Transport layer transmit a PIO Setup FIS. The initial status shall have the BSY bit cleared to zero and the DRQ bit set to one. The Interrupt bit shall be set to one. The ending status shall have the BSY bit set to one and the DRQ bit cleared to zero. The byte count for the DRQ data block shall be indicated.

Transition DPKT7:1, if the PIO Setup FIS has been transferred, the device shall transition to the DPKT8: Receive_PIO_data state.

DPKT8: Receive_PIO_data, this state is entered if the device transmitted a PIO Setup FIS to the host.

If in this state, the device shall receive the requested Data FIS from the Transport layer.

Transition DPKT8:1, if the Data FIS has been received, the device shall transition to the DPKT6: PACKET_PIO_out state.

DPKT9: PACKET_DMA_in State, this state is entered if the device receives a DMA DATA-IN command or the transmission of one or more Data FIS is required to complete the command.

If in this state, device shall prepare the data for transfer of a Data FIS to the host.

Transition DPKT9:1, if the device has the data ready to transfer a Data FIS, the device shall transition to the DPKT10: Send_DMA_data state.

Transition DPKT9:2, if the device has transferred all of the data requested by this command or has encountered an error that causes the command to abort before completing the transfer of the requested data, then the device shall transition to the DPKT14: Send_status state.

Transition DPKT9:3, if the device supports overlap and queuing and does not have data ready to transfer immediately, the device shall transition to the DPKT15: Release state. This transition is only utilized if queuing is implemented.

DPKT10: Send_DMA_data, this state is entered if the device has the data ready to transfer a Data FIS to the host.

If in this state, the device shall request that the Transport layer transmit a Data FIS containing the data.

Transition DPKT10:1, if the Data FIS has been transferred, the device shall transition to the DPKT9: PACKET_DMA_in state. The Device command layer shall request a data FIS size of no more than 2 048 Dwords.

DPKT11: PACKET_DMA_out State, this state is entered if the device receives a DMA DATA-OUT command or the receipt of one or more Data FIS is required to complete the command.

If in this state, device shall prepare to receive a Data FIS from the host.

Transition DPKT11:1, if the device is ready to receive a Data FIS, the device shall transition to the DPKT12: Send_DMA_activate state.

Transition DPKT11:2, if the device has received all the data requested by this command or the device has encountered an error that causes the command to abort before completing the transfer of the requested data, then the device shall transition to the DPKT14: Send_status state.

Transition DPKT11:3, if the device supports overlap and queuing and is not in a state that it is possible to accept a Data FIS immediately, the device shall transition to the DPKT15: Release state. This transition is only utilized if queuing is implemented.

DPKT12: Send_DMA_activate, this state is entered if the device is ready to receive a Data FIS from the host.

If in this state, the device shall request that the Transport layer transmit a DMA Activate FIS.

Transition DPKT12:1, if the DMA Activate FIS has been transferred, the device shall transition to the DPKT13: Receive_DMA_data state.

DPKT13: Receive_DMA_data, this state is entered if the device transmitted a DMA Activate FIS to the host.

If in this state, the device shall receive the requested Data FIS from the Transport layer.

Transition DPKT13:1, if the Data FIS has been received, the device shall transition to the DPKT11: PACKET_DMA_out state.

DPKT14: Send_status, this state is entered if the device has received all the data requested by this command or the device has encountered an error that causes the command to abort before completing the transfer of the requested data.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS with register content as described in the command description in the ACS-4 standard and the Interrupt bit set to one.

Transition DPKT14:1, if the FIS has been transmitted, then the device shall transition to the DI0: Device_idle state.

DPKT15: Release, this state is entered if the device is not able to do a data transfer immediately. This state is only utilized if queuing is implemented.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS with register content as described in the command description in the ACS-4 standard, with the REL bit set to one, and, if the bus release interrupt has been enabled by a previous Set Features Command, with the Interrupt bit set to one.

Transition DPKT15:1, if the FIS has been transmitted, then the device shall transition to the DI0: Device_idle state.

11.13 READ DMA QUEUED command protocol

Processing of this class of command includes the transfer of one or more blocks of data from the device to the host using DMA transfer (see Figure 332). All data for the command may be transferred without a bus release between the command receipt and the data transfer. This command may bus release before transferring data. The host shall initialize the DMA controller prior to transferring data. If data transfer is begun, all data for the request shall be transferred without a bus release.

DDMAQI0: DMA_queued_in	Determine whether to transfer or release.	
1. Data for Data FIS ready to transfer.	→	DDMAQI1: Send_data
2. Command aborted due to error.	→	DDMAQI3: Send_status
3. Bus release	→	DDMAQI4: Release

DDMAQI1: Send_data	Request transmission of a Data FIS to host.	
1. Data FIS transmitted. No more data transfer required for this command, or 2 048 Dwords transmitted.	→	DDMAQI2: Prepare_data

DDMAQI2: Prepare_data	Prepare data for the next Data FIS.	
1. Data ready.	→	DDMAQI1: Send _data
2. Command complete or aborted due to error.	→	DDMAQI3: Send_status

DDMAQI3: Send_status	Request transmission of a Register Device to Host FIS.	
1. FIS transmitted.	→	DI0: Device_idle

DDMAQI4: Release	Request transmission of a Register Device to Host FIS.	
1. FIS transmitted.	→	DI0: Device_idle

Figure 332 – Device command layer read DMA queued state machine

DDMAQI0: DMA_queued_in State, this state is entered if the device receives a READ DMA QUEUED command.

If in this state, device shall determine if the requested data is ready to transfer to the host.

Transition DDMAQI0:1, if the device has the requested data ready to transfer a Data FIS immediately, the device shall transition to the DDMAQI1: Send_data state.

Transition DDMAQI0:2, if the device has encountered an error that causes the command to abort before completing the transfer of the requested data, the device shall transition to the DDMAQI3: Send_status state.

Transition DDMAQI0:3, if the device does not have the requested data ready to transfer a Data FIS immediately, the device shall transition to the DDMAQI4: Release state.

DDMAQI1: Send_data, this state is entered if the device has the data ready to transfer a Data FIS to the host.

If in this state, the device shall request that the Transport layer transmit a Data FIS containing the data.

Transition DDMAQI1:1, if the Data FIS has been transferred, the device shall transition to the DDMAQI2: Prepare_data state. The Device command layer shall request a Data FIS size of no more than 2 048 Dwords .

DDMAQI2: Prepare_data, this state is entered if the device has completed the transfer a Data FIS to the host.

If in this state, the device shall prepare the data for the next Data FIS.

Transition DDMAQI2:1, if data is ready for the Data FIS, the device shall transition to the DDMAQI1: Send_data state.

Transition DDMAQI2:2, if all data requested for the command has been transmitted or an error has been encountered that causes the command to abort before completing the transfer of the requested data, the device shall transition to the DDMAQI3: Send_status state.

DDMAQI3: Send_status, this state is entered if the device has transferred all of the data requested by the command or has encountered an error that causes the command to abort before completing the transfer of the requested data.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS with register content as described in the command description in the ACS-4 standard and the Interrupt bit set to one.

Transition DDMAQI3:1, if the FIS has been transmitted, the device shall transition to the DI0: Device_idle state.

DDMAQI4: Release, this state is entered if the device does not have the requested data available for immediate transfer.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS with the REL bit set to one, with register content as described in the command description in the ACS-4 standard, and, if the bus release interrupt has been enabled by a previous Set Features Command, with the Interrupt bit set to one.

Transition DDMAQI4:1, if the FIS has been transmitted, then the device shall transition to the DI0: Device_idle state.

11.14 WRITE DMA QUEUED command protocol

Processing of this class of command includes the transfer of one or more blocks of data from the device to the host using DMA transfer (see Figure 333). All data for the command may be transferred without a bus release between the command receipt and the data transfer. This command may bus release before transferring data. The host shall initialize the DMA controller prior to transferring data. If data transfer is begun, all data for the request shall be transferred without a bus release.

DDMAQO0: DMA-queued_out	Determine whether to transfer or release.	
1. Ready to accept Data FIS	→	DDMAQO1: Send_DMA_activate
2. Command due to error.	→	DDMAQO4: Send_status
3. Bus release	→	DDMAQO5: Release

DDMAQO1: Send_DMA_activate	Request transmission of a DMA Activate FIS to host.	
1. DMA Activate FIS transmitted.	→	DDMAQO2: Receive_data

DDMAQO2: Receive_data	Receive Data FIS from the Transport layer.	
1. Data FIS received.	→	DDMAQO3: Prepare_data_buffer

DDMAQO3: Prepare_data_buffer	Prepare to receive the next Data FIS.	
1. Ready to receive.	→	DDMAQO1: Send DMA_activate
2. Command complete or aborted due to error.	→	DDMAQO4: Send_status

DDMAQO4: Send_status	Request transmission of a Register Device to Host FIS to host.	
1. FIS transmitted.	→	DI0: Device_idle

DDMAQO5: Release	Request transmission of a Register Device to Host FIS to host.	
1. FIS transmitted.	→	DI0: Device_idle

Figure 333 – Device command layer write DMA queued state machine

DDMAQO0: DMA_queued_out State, this state is entered if the device receives a WRITE DMA QUEUED command.

If in this state, device shall determine if it is ready to accept the requested data from the host.

Transition DDMAQO0: 1, if the device is ready to receive a Data FIS immediately, the device shall transition to the DDMAQO1: Send_DMA_activate state.

Transition DDMAQO0:2, if the device has encountered an error that causes the command to abort before completing the transfer of the requested data, the device shall transition to the DDMAQO4: Send_status state.

Transition DDMAQO0:3, if the device is not ready to receive a Data FIS immediately, the device shall transition to the DDMAQO5: Release state.

DDMAQO1:Send_DMA_activate, this state is entered if the device is ready to receive a Data FIS from the host.

If in this state, the device shall request that the Transport layer transmit a DMA Activate FIS.

Transition DDMAQO1:1, if the DMA Activate FIS has been transferred, the device shall transition to the DDMAQO2: Receive_data state.

DDMAQO2:Receive_data, this state is entered if the device transmitted a DMA Activate FIS to the host.

If in this state, the device shall receive the requested Data FIS from the Transport layer.

Transition DDMAQO2:1, if the Data FIS has been received, the device shall transition to the DDMAQO3: Prepare_data_buffer state.

DDMAQO3: Prepare_data_buffer, this state is entered if the device has completed receiving a Data FIS from the host.

If in this state, the device shall prepare for receipt of the next Data FIS.

Transition DDMAQO3:1, if ready to receive the Data FIS, the device shall transition to the DDMAQO1: Send_DMA_activate state.

Transition DDMAQO3:2, if all data requested for the command has been transmitted or an error has been encountered that causes the command to abort before completing the transfer of the requested data, the device shall transition to the DDMAQO4: Send_status state.

DDMAQO4: Send_status, this state is entered if the device has transferred all of the data requested by the command or has encountered an error that causes the command to abort before completing the transfer of the requested data.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS with register content as described in the command description in the ACS-4 standard and the Interrupt bit set to one.

Transition DDMAQO4:1, if the FIS has been transmitted, then the device shall transition to the DI0: Device_idle state.

DDMAQO5: Release, this state is entered if the device is unable to receive the requested data immediately.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS with REL set to one, with register content as described in the command description in the ACS-4 standard, and, if the bus release interrupt has been enabled by a previous Set Features Command, with the Interrupt bit set to one.

Transition DDMAQO5:1, if the FIS has been transmitted, then the device shall transition to the DI0: Device_idle state.

11.15 FPDMA QUEUED command protocol

The Device command layer FPDMA Queued state machine is defined in Figure 334.

This class includes NCQ commands.

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

DFPDMAQ2: ClearInterfaceBsy	Transmit Register Device to Host FIS with the BSY bit cleared to zero and the DRQ bit cleared to zero and Interrupt bit cleared to zero to mark interface ready for the next command.	
1. FIS transmission complete.	→	DI0: Device_idle

DFPDMAQ3: DataPhaseReadSetup	Transmit a DMA Setup FIS to the host with the DMA Buffer Identifier = TAG and D bit set to one (direction is device to host) and Interrupt bit cleared to zero	
1. FIS transmission complete.	→	DFPDMAQ8: DataXmitRead

DFPDMAQ4: DataPhasePreWriteSetup		
1. DMA Setup FIS Auto-Activate option supported and enabled.	→	DFPDMAQ5: DataPhase_ WriteSetup
2. DMA Setup FIS Auto-Activate option not supported or not enabled.	→	DFPDMAQ6: DataPhase_ OldWriteSetup

DFPDMAQ5: DataPhase_WriteSetup	Transmit a DMA Setup FIS to the host with the DMA Buffer Identifier = TAG and D bit cleared to zero (direction is host to device) and Auto-Activate bit set to one and Interrupt bit cleared to zero.	
1. DMA Setup FIS transmission complete.	→	DFPDMAQ9: DataXmitWrite

DFPDMAQ6: DataPhase_OldWriteSetup	Transmit a DMA Setup FIS to the host with the DMA Buffer Identifier = TAG and D bit cleared to zero (direction is host to device) and Auto-Activate bit cleared to zero and Interrupt bit I cleared to zero.	
1. DMA Setup FIS transmission complete.	→	DFPDMAQ7: DataPhase_ XmitActivate

DFPDMAQ7: DataPhase_XmitActivate	Transmit a DMA Activate FIS to the host.	
1. DMA Activate FIS transmission complete.	→	DFPDMAQ9: DataXmitWrite

Figure 334 – Device command layer FPDMA queued state machine (part 1 of 3)

DFPDMAQ8: DataXmitRead	Transmit Data FIS to the host.	
1. Transfer count for previous DMA Setup FIS not exhausted and no error encountered.	→	DFPDMAQ8: DataXmitRead
2. Transfer count for previous DMA Setup FIS exhausted and data transfer for this command not complete and no error encountered. [a]	→	DI0: Device_idle
3. Finished with data transfer for this command and no error encountered.	→	DI0: Device_idle
4. Unrecoverable error has occurred.	→	DI0: Device_idle
[a] This condition requires that non-zero buffer offsets be supported and enabled. The transition also applies if a device switches between multiple active commands and is performing partial data transfers for the multiple outstanding commands.		

DFPDMAQ9: DataXmitWrite	Receive Data FIS from host.	
1. Transfer count for previous DMA Setup FIS not exhausted and no error encountered.	→	DFPDMAQ7: DataPhase_ XmitActivate
2. Transfer count for previous DMA Setup FIS exhausted and data transfer for this command not complete and no error encountered. [a]	→	DI0: Device_idle
3. Finished with data transfer for this command and no error encountered.	→	DI0: Device_idle
4. Unrecoverable error has occurred.	→	DI0: Device_idle
[a] This condition requires that non-zero buffer offsets be supported and enabled. The transition also applies if a device switches between multiple active commands and is performing partial data transfers for the multiple outstanding commands.		

DFPDMAQ10: SendStatus	Transmit Set Device Bits FIS with the ERR bit cleared to zero, Interrupt bit set to one, and bit **n** in ACT field set to one where **n** = TAG for each command TAG value that has completed since the last status return.	
1. Set Device Bits FIS transmission complete.	→	DI0: Device_idle

DFPDMAQ11: ERROR	Halt command processing and transmit Set Device Bits FIS to host with the ERR bit in Status field set to one, Interrupt bit set to one, ATA error code set to one in the ERROR field, bits in ACT field cleared to zero for any outstanding queued commands, and bits set to one for any successfully completed queued commands that completion notification not yet delivered.	
1. Set Device Bits FIS transmission complete.	→	DFPDMAQ13: WaitforClear

Figure 334 – Device command layer FPDMA queued state machine (part 2 of 3)

DFPDMAQ12: BrokenHost_ClearBusy	Halt command processing and transmit Register Device to Host FIS with the ERR bit in Status field set to one, Interrupt bit set to one, the BSY bit cleared to zero, the DRQ bit cleared to zero, and the ERROR field = 04h. If error condition was due to reception of an Unload request and the device supports Unload, if NCQ commands are outstanding, the device shall unload/park the heads, if any.	
1. FIS transmission complete.	→	DFPDMAQ13: WaitforClear

DFPDMAQ13: WaitforClear	Wait for host to either issue a command to read the Queued Error log or issue SRST.	
1. READ LOG EXT command with Queued Error log received.	→	DFPDMAQ14: SendQueue_ CleanACK
2. READ LOG DMA EXT command with Queued Error log received. [a].	→	DFPDMAQ15: SendQueue_ CleanACKDMA
3. SRST received.	→	DSR0: Software_reset_ asserted
4. Any other command received.	→	DFPDMAQ12: BrokenHost_ ClearBusy
[a] See 13.7		

DFPDMAQ14: SendQueue_CleanACK	Discard all commands in the pending device queue. Transmit Set Device Bits FIS with the ERR bit in Status field cleared to zero, the ERROR field cleared to 00h, ACT field = FFFF FFFFh, and Interrupt bit cleared to zero.	
1. Set Device Bits FIS transmission complete.	→	DPIOI0: PIO_in

DFPDMAQ15: SendQueue_ CleanACKDMA	Discard all commands in the pending device queue. Transmit Set Device Bits FIS with the ERR bit in Status field cleared to zero, the ERROR field cleared to 00h, ACT field = FFFF FFFFh, and Interrupt bit cleared to zero.	
1. Set Device Bits FIS transmission complete.	→	DDMAI0: DMA_in

Figure 334 – Device command layer FPDMA queued state machine (part 3 of 3)

DFPDMAQ1: AddCommandToQueue, this state is entered if the device has checked the command and determined it to be a native queued type command, and NCQ is supported and enabled.

If in this state, the device shall check the TAG field validity and verify that it is not already assigned to an outstanding command. If valid, the device shall append the command to its internal command queue and store the new TAG value.

Transition DFPDMAQ1:1, if the device determines the TAG field is valid, and has added the command to its internal command queue, the device shall transition to the DFPDMAQ2: ClearInterfaceBusy state.

Transition DFPDMAQ1:2, if the device determines that the received command is malformed, then 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 50 – 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).

DFPDMAQ2: ClearInterfaceBusy, this state is entered if the device has appended the command to its internal queue and is ready to transmit a Register Device to Host FIS with the BSY bit cleared to zero and the DRQ bit cleared to zero to indicate that the interface is ready to receive the next command.

Transition DFPDMAQ2:1, if the Register Device to Host FIS has been transmitted, the device shall transition to the DI0: Device_idle state.

DFPDMAQ3: DataPhaseReadSetup, this state is entered if the device has determined that it is ready to transmit data for a previously queued READ FPDMA QUEUED or RECEIVE FPDMA QUEUED command.

If in this state, the device shall transmit a DMA Setup FIS to the host with the DMA Buffer Identifier set to the queued TAG value and the D bit set to one (host memory write).

Transition DFPDMAQ3:1, if the device completes the transmission of the DMA Setup FIS, the device shall transition to the DFPDMAQ8: DataXmitRead state.

DFPDMAQ4: DataPhasePreWriteSetup, this state is entered if the device has determined that it is ready to receive data for a previously queued WRITE FPDMA QUEUED or SEND FPDMA QUEUED command.

If in this state, the device shall determine if the DMA Setup Auto-Activate option is supported and enabled, and then make the appropriate state transition.

Transition DFPDMAQ4:1, if the DMA Setup FIS Auto-Activate option is enabled, the device shall transition to the DFPDMAQ5: DataPhase_WriteSetup state.

Transition DFPDMAQ4:2, if the DMA Setup FIS Auto-Activate option is not supported or not enabled, the device shall transition to the DFPDMAQ6: DataPhase_OldWriteSetup state.

DFPDMAQ5: DataPhase_WriteSetup, this state is entered if the device is ready to Auto Activate and receive data for a previously queued WRITE FPDMA QUEUED or SEND FPDMA QUEUED command.

If in this state, the device transmits a DMA Setup FIS to the host with the DMA Buffer Identifier set to the queued TAG value and the D bit cleared to zero (host memory read), and Auto-Activate bit set to one.

Transition DFPDMAQ5:1, if the device completes the transmission of the DMA Setup FIS, the device shall transition to the DFPDMAQ9: DataXmitWrite state.

DFPDMAQ6: DataPhase_OldWriteSetup, this state is entered if the device is ready to receive data for a previously queued WRITE FPDMA QUEUED or SEND FPDMA QUEUED command, and the device does not support Auto-Activate, or it is not enabled.

If in this state, the device transmits a DMA setup FIS to the host with the DMA Buffer Identifier set to the queued TAG value and the D bit cleared to zero (host memory read), and Auto-Activate bit cleared to zero.

Transition DFPDMAQ6:1, if the device completes the transmission of the DMA Setup FIS, the device shall transition to the DFPDMAQ7: DataPhase_XmitActivate state.

DFPDMAQ7: DataPhaseXmit_Activate, this state is entered after the device has completed transmission of a DMA Setup FIS for a WRITE FPDMA QUEUED or SEND FPDMA QUEUED command or the device has finished receiving a Data FIS for a WRITE FPDMA QUEUED or SEND FPDMA QUEUED command, and the transfer count is not exhausted.

If in this state, the device transmits a DMA Activate FIS to the host indicating readiness to receive Data FISes from the host.

Transition DFPDMAQ7:1, if the device completes the transmission of the DMA Activate FIS, the device shall transition to the DFPDMAQ9: DataXmitWrite state.

DFPDMAQ8: DataXmitRead, this state is entered after the device has completed transmission of a DMA Setup FIS for a READ FPDMA QUEUED or RECEIVE FPDMA QUEUED command.

If in this state, the device transmits a Data FIS to the host.

Transition DFPDMAQ8:1, if the transfer count for the previous DMA Setup FIS is not exhausted and no error is encountered, the device remains in the DFPDMAQ8: DataXmitRead state.

Transition DFPDMAQ8:2, if the transfer count for the previous DMA Setup FIS is exhausted, and the data transfer for this command is not complete, and no error is encountered, the device shall transition to the DI0: Device_idle state.

This condition requires that non-zero buffer offsets be supported and enabled. The transition also applies if a device switches between multiple active commands and is performing partial data transfers for the multiple outstanding commands.

Transition DFPDMAQ8:3, if the device has completed the data transfer for this command, and no error is encountered, the device shall transition to the DI0: Device_idle state.

Transition DFPDMAQ8:4, if the device determines that an unrecoverable error has occurred, the device shall transition to the DI0: Device_idle state.

DFPDMAQ9: DataXmitWrite, this state is entered after the device has completed transmission of a DMA Setup FIS for a WRITE FPDMA QUEUED or SEND FPDMA QUEUED command.

If in this state, the device receives a Data FIS from the host.

Transition DFPDMAQ9:1, after the data FIS reception is complete, and if the transfer count for the previous DMA Setup FIS is not exhausted and no error is encountered, the device shall transition to the DFPDMAQ7: DataPhase_XmitActivate state.

Transition DFPDMAQ9:2, if the transfer count for the previous DMA Setup FIS is exhausted, and the data transfer for this command is not complete, and no error is encountered, the device shall transition to the DI0: Device_idle state.

This condition requires that non-zero buffer offsets be supported and enabled. The transition also applies if a device switches between multiple active commands and is performing partial data transfers for the multiple outstanding commands.

Transition DFPDMAQ9:3, if the device has completed the data transfer for this command, and no error is encountered, the device shall transition to the DI0: Device_idle state.

Transition DFPDMAQ9:4, if the device determines that an unrecoverable error has occurred, the device shall transition to the DI0: Device_idle state.

DFPDMAQ10: SendStatus, this state is entered if the data transfer for this command, or aggregated commands, is completed and the device is ready to send status.

If in this state, the device transmits a Set Device Bits FIS to the host with the ERR bit cleared to zero, Interrupt bit set to one, and bit **n** in ACT field set to one where **n** = TAG for each command TAG value that has completed since the last status return.

Transition DFPDMAQ10:1, if the device completes the transmission of the Set Device Bits FIS, the device shall transition to the DI0: Device_idle state.

DFPDMAQ11: ERROR, this state is entered if the device has encountered an unrecoverable error.

If in this state, the device halts command processing and transmits a Set Device Bits FIS to the host with the ERR bit set to one, Interrupt bit set to one, ATA error code set to one in the ERROR field, and bits in ACT field cleared to zero for any outstanding queued commands (including the erring command) and bits set to one for any successfully completed queued command that a completion notification has not yet been provided to the host.

Transition DFPDMAQ11:1, if the device completes the transmission of the Set Device Bits FIS, the device shall transition to the DFPDMAQ13: WaitforClear state.

DFPDMAQ12: BrokenHost_ClearBusy, this state is entered if:

a) the device has received:
   A. a READ FPDMA QUEUED command;
   B. a WRITE FPDMA QUEUED command;
   C. a NCQ NON-DATA command;
   D. a RECEIVE FPDMA QUEUED command; or
   E. a SEND FPDMA QUEUED command,

   with a TAG that already exists in its command queue;

   or

b) the received command is not:
   A. a READ FPDMA QUEUED command;
   B. a WRITE FPDMA QUEUED command;
   C. a NCQ NON-DATA command;
   D. a RECEIVE FPDMA QUEUED command;
   E. a SEND DMA QUEUED command; and
   F. a DEVICE RESET command,

   and there are NCQ command(s) outstanding.

If in this state, the device halts command processing and transmits a Register Device to Host FIS with the ERR bit set to one in the Status field, Interrupt bit set to one, the BSY bit cleared to zero, the DRQ bit cleared to zero, and ATA error code set to one in the ERROR field. If error condition was due to reception of an Unload request and the device supports Unload if NCQ commands are outstanding (i.e., the UNLOAD WHILE NCQ COMMANDS ARE OURSTANDING SUPPORTED bit is set to one), the device shall unload/park the heads.

Transition DFPDMAQ12:1, if the device completes the transmission of the Register Device to Host FIS, the device shall transition to the DFPDMAQ13: WaitforClear state.

DFPDMAQ13: WaitforClear, this state is entered if the device has transmitted an error FIS to the host and is awaiting a command to read the Queued Error log (see 13.7.4) or a soft reset. Any other commands return Register Device to Host FIS with the ERR bit set to one in the Status field.

Transition DFPDMAQ13:1, if the device receives a READ LOG EXT command to read the Queued Error log, the device shall transition to the DFPDMAQ14: SendQueue_CleanACK state.

Transition DFPDMAQ13:2, if the device receives a READ LOG DMA EXT command to read the Queued Error log, and IDENTIFY DEVICE data Word 76 bit 15 is set to one, the device shall transition to the DFPDMAQ15: SendQueue_CleanACKDMA state.

Transition DFPDMAQ13:3, if the device receives a SRST, the device shall transition to the DSR0: Software_reset_asserted state.

Transition DFPDMAQ13:4, if the device receives any other command, the device shall transition to the DFPDMAQ12: BrokenHost_ClearBusy state.

DFPDMAQ14: SendQueue_CleanACK, this state is entered if the host has responded to an error FIS to a command to read the Queued Error log.

The device shall discard all commands in the pending queue and transmit a Set Device Bits FIS with the ERR bit in the Status field cleared to zero, the ERROR field cleared to 00h, ACT field = FFFF FFFFh, and Interrupt bit cleared to zero.

Transition DFPDMAQ14:1, if the Set Device Bits FIS transmission is complete, the device shall transition to the DPIOI0: PIO_in state.

DFPDMAQ15: SendQueue_CleanACKDMA, this state is entered if the host has responded to an error FIS to a command to read the Queued Error log.

The device shall discard all commands in the pending queue and transmit a Set Device Bits FIS with the ERR bit in the Status field cleared to zero, the ERROR field cleared to 00h, ACT field = FFFF FFFFh, and Interrupt bit cleared to zero.

Transition DFPDMAQ15:1, if the Set Device Bits FIS transmission is complete, the device shall transition to the DDMAI0: DMA_in state.

12 Host command layer protocol

12.1 FPDMA QUEUED command protocol overview

This high-level state machine describes the behavior of the host for the NCQ command protocol. The host behavior described by the state machine may be provided by host software or host hardware and the intent of the state machines is not to indicate any particular implementation.

This class includes NCQ commands.

12.2 FPDMA QUEUED command protocol

The Host command layer FPDMA Queued state machine is defined in Figure 335.

HFPI0: Idle [a]			
	1. Free TAG location and command waiting to have TAG assigned.	→	HFPDMAQ2: PresetACTBit
	2. Command with assigned TAG awaiting issue, and the BSY bit cleared to zero, and not First-party DMA Data Phase.	→	HFPDMAQ3: IssueCommand
	3. Interrupt received from device.	→	HFPDMAQ4: DeviceINT
	4. Default.	→	HFPI0: Idle
	[a] If more than one condition is true, the host may apply a vendor specific priority.		

HFPDMAQ1: AddCommandToQueue	Append command to internal host command queue.		
	1. Unconditional.	→	HFPI0: Idle

HFPDMAQ2: PresetACTBit	Assign free TAG value to command. Write SActive register with value that has bit set in bit position corresponding to assigned TAG value.		
	1. TAG value assigned to command and SActive register written with new TAG bitmask.	→	HFPI0: Idle

HFPDMAQ3: IssueCommand	If not First-party DMA Data Phase, transmit Register Host to Device FIS to device with new command and assigned TAG value.		
	1. FIS transmission complete (command issued).	→	HFPI0: Idle
	2. FIS transmission deferred (command not issued).	→	HFPI0: Idle

Figure 335 – Host command layer FPDMA queued state machine (part 1 of 3)

HFPDMAQ4: DeviceINT	Read Status register to clear pending interrupt flag and save value as SavedStatus.	
1. Unconditional.	→	HFPDMAQ5: CompleteRequests1

HFPDMAQ5: CompleteRequests1	Compare SActive register with stored SActive register from last interrupt to identify completed commands.	
1. SActive comparison indicates one or more commands are completed.	→	HFPDMAQ6: CompleteRequests2
2. SActive comparison indicates no commands are completed.	→	HFPDMAQ7: CompleteRequests3

HFPDMAQ6: CompleteRequests2	Retire host requests associated with TAG values corresponding to newly cleared bits in the SActive register and update stored SActive with new value.	
1. Unconditional.	→	HFPDMAQ7: CompleteRequests3

HFPDMAQ7: CompleteRequests3	Test the ERR bit in SavedStatus value.	
1. The ERR bit cleared to zero.	→	HFPI0: Idle
2. The ERR bit set to one.	→	HFPDMAQ8: ResetQueue

HFPDMAQ8: ResetQueue	Issue a command to read the Queued Error log to device.	
1. READ LOG EXT command was accepted.	→	HFPDMAQ9: CleanupACK
2. READ LOG DMA EXT command was accepted.	→	HFPDMAQ12: RetrieveRequest_ SenseDMA
3. Command was not accepted.	→	HFPDMAQ8: ResetQueue

HFPDMAQ9: CleanupACK	Wait for the DRQ bit set to one and the BSY bit cleared to zero. [a]	
1. the DRQ bit set to one and the BSY bit cleared to zero.	→	HFPDMAQ10: RetrieveRequest_ Sense
2. the DRQ bit cleared to zero or the BSY bit set to one.	→	HFPDMAQ9: CleanupACK
[a] The host may wait for this condition using any means including awaiting an interrupt and checking the the DRQ bit and the BSY bit status, spinning, or periodic timer.		

Figure 335 – Host command layer FPDMA queued state machine (part 2 of 3)

HFPDMAQ10: RetrieveRequest_Sense	Receive PIO Data FIS with Queued Error log contents.	
1. PIO Data FIS reception complete.	→	HFPDMAQ11: ErrorFlush

HFPDMAQ11: ErrorFlush	Retire failed queued command with status set to error condition reported by device. Flush all allocated native queued command tags. Flush pending native queued commands from host command queue with system-specific error condition or re-issue pending queued commands.	
1. Unconditional.	→	HFPI0: Idle

HFPDMAQ12: RetrieveRequest_ SenseDMA	Receive Data FIS with Queued Error log contents.	
1. Data FIS reception complete.	→	HFPDMAQ13: SendStatus

HFPDMAQ13: SendStatus	Request transmission of a Register Device to Host FIS.	
1. FIS transmitted.	→	HFPDMAQ11: ErrorFlush

Figure 335 – Host command layer FPDMA queued state machine (part 3 of 3)

HFPI0: Idle, if in this state, if queuing is supported and enabled, the Command layer is awaiting an NCQ command:

a) command from the higher level protocol;
b) awaiting an interrupt from the device indicating completion of previously queued commands; or
c) waiting for a TAG location to become available for a command waiting in the command queue.

Transition HFPI0:1, if an NCQ command is pending that has not had a TAG value assigned to it and there is a free TAG location available for assignment, then a transition shall be made to the HFPDMAQ2: PresetACTBit state.

Transition HFPI0:2, if an NCQ command with assigned TAG value is awaiting issue to the device, the BSY bit cleared to zero, and the interface is not in the First-party DMA Data Phase, then a transition shall be made to the HFPDMAQ3: IssueCommand state.

Transition HFPI0:3, if an interrupt is received from the device, indicating status is available for a previously queued NCQ command, it shall transition to the HFPDMAQ4: DeviceINT state.

Transition HFPI0:4, if the queuing is supported and enabled, and the Command layer is awaiting a free TAG, a new NCQ command, or an interrupt for a previously queued command, it shall transition to the HFPI0: Idle state.

HFPDMAQ1: AddCommandToQueue, the Command layer enters this state if it has received an NCQ command from the higher level protocol, and adds it to the internal host command queue.

Transition HFPDMAQ1:1: after the Command layer has added the NCQ command to the internal host command queue, it shall transition to the HFPI0: Idle state.

HFPDMAQ2: PresetACTBit, if in this state, the Command layer assigns a free TAG value to the previously queued NCQ command and writes the SActive register with the bit position corresponding to the assigned TAG value.

Transition HFPDMAQ2:1: After the Command layer has assigned a TAG value and written the corresponding bit to the SActive register, it shall transition to the HFPI0: Idle state.

HFPDMAQ3: IssueCommand, if in this state, the Command layer attempts to issue an NCQ command with preassigned TAG to the device by transmitting a Register Host to Device FIS with the new NCQ command and assigned TAG value if the interface state permits it.

Transition HFPDMAQ3:1: After the Command layer has transmitted the Register Host to Device FIS, it shall mark the corresponding NCQ command as issued and transition to the HFPI0: Idle state.

Transition HFPDMAQ3:2: After the Command layer has deferred transmission of the Register Host to Device FIS due to the interface state not permitting it to be delivered, it shall transition to the HFPI0: Idle state. The corresponding NCQ command is still considered as not having been issued.

HFPDMAQ4: DeviceINT, if in this state, the Command layer reads the Device Status register to reset the pending interrupt flag and save the value as SavedStatus.

Transition HFPDMAQ4:1: After the Command layer has read the Device Status register, it shall transition to the HFPDMAQ5: CompleteRequests1 state.

HFPDMAQ5: CompleteRequests1, if in this state, the Command layer compares the SActive register with the SavedStatus SActive register value that resulted from the last interrupt to identify completed NCQ commands.

Transition HFPDMAQ5:1, if the SActive comparison indicates one or more NCQ commands have completed, it shall transition to the HFPDMAQ6: CompleteRequests2 state.

Transition HFPDMAQ5:2, if the SActive comparison indicates no NCQ commands have completed, it shall transition to the HFPDMAQ7: CompleteRequests3 state.

HFPDMAQ6: CompleteRequests2, if in this state, the Command layer retires commands in its internal host command queue that are associated with TAG values corresponding to newly cleared bits in the SActive register and updates the stored SActive register with the new value.

Transition HFPDMAQ6:1: After updating the stored SActive value, it shall transition to the HFPDMAQ7: CompleteRequests3 state.

HFPDMAQ7: CompleteRequests3, if in this state, the Application layer tests the ERR bit in the SavedStatus value to determine whether the queue should be maintained or reset.

Transition HFPDMAQ7:1, if the SavedStatus value ERR bit cleared to zero, the queue is maintained and it shall transition to the HFPI0: Idle state.

Transition HFPDMAQ7:2, if the SavedStatus value ERR bit set to one, an error has been reported by the device and the Command layer shall transition to the HFPDMAQ8: ResetQueue state.

HFPDMAQ8: ResetQueue, if in this state, the Application layer issues a command to read the Queued Error log (see 13.7.4) to the device.

Transition HFPDMAQ8:1: After a READ LOG EXT command has been accepted, it shall transition to the HFPDMAQ9: CleanupACK state.

Transition HFPDMAQ8:2: After a READ LOG DMA EXT command has been accepted, it shall transition to the HFPDMAQ12: RetrieveRequest_SenseDMA .

Transition HFPDMAQ8:3, if the command was not accepted, the host shall transition to the HPFDMAQ8: ResetQueue state.

HFPDMAQ9: CleanupACK, if in this state, the Command layer tests the device for the DRQ bit set to one and the BSY bit cleared to zero in preparation for a PIO data FIS transfer.

Transition HFPDMAQ9:1, if the DRQ bit is set to one and the BSY bit is cleared to zero, it shall transition to the HFPDMAQ10: ReceiveRequestSense state.

Transition HFPDMAQ9:2, if the DRQ bit is cleared to zero or the BSY bit is set to one, it shall transition to the HFPDMAQ9: CleanupACK state.

HFPDMAQ10: RetrieveRequest_Sense, if in this state, the Command layer completes the PIO Data FIS that retrieves the Queued Error log contents.

Transition HFPDMAQ10:1: After the completion of the PIO Data FIS, it shall transition to the HFPDMAQ11: ErrorFlush state.

HFPDMAQ11: ErrorFlush, if in this state, the Command layer retires the failed queued command with the error status set to the error condition reported by the device. It flushes all allocated NCQ command tags, and flushes pending NCQ commands from the host command queue with system-specific error condition or re-issue pending NCQ commands.

Transition HFPDMAQ11:1: After the error flush actions have been completed, it shall transition to the HFPI0: Idle state.

HFPDMAQ12: RetrieveRequest_SenseDMA, this state is entered if the device has the data ready to transfer a data FIS to the host containing the Queued Error log contents.

If in this state, the device shall request that the Transport layer transmit a data FIS containing the data. The Device command layer shall request a Data FIS size of no more than 2 048 Dwords.

Transition HFPDMAQ12:1, if the FIS has been transmitted, the device shall transition to the HFPDMAQ13: SendStatus state.

HFPDMAQ13: SendStatus, this state is entered if the device has transferred all of the data requested by the command or has encountered an error that causes the command to abort before completing the transfer of the requested data.

If in this state, the device shall request that the Transport layer transmit a Register Device to Host FIS with register content as described in the command description in the ACS-4 standard and the Interrupt bit set to one.

Transition HFPDMAQ13:1, if the FIS has been transmitted, the device shall transition to the HFPDMAQ11: ErrorFlush state.

13 Application layer

13.1 Parallel ATA emulation (obsolete)

For information on Parallel ATA emulation see SATA Revision 3.2.

13.2 IDENTIFY (PACKET) DEVICE

13.2.1 IDENTIFY (PACKET) DEVICE overview

In the IDENTIFY DEVICE command various parameters are communicated to the host from the device. The following sections define those Words that are different from and additions to the ACS-4 standard definition of the data contents. Serial ATA features and capabilities include a means that their presence and support may be determined, and a means for enabling them if optionally supported.

The IDENTIFY (PACKET) DEVICE settings requirements shall be implemented by native Serial ATA devices. The IDENTIFY (PACKET) DEVICE settings requirements are optional for parallel ATA devices with an external Serial ATA bridge attached.

13.2.2 IDENTIFY DEVICE

13.2.2.1 IDENTIFY DEVICE information

The IDENTIFY DEVICE information structure is defined in Table 101.

Table 101 – IDENTIFY DEVICE information (part 1 of 4)

Word	O/M	F/V	Description
0..46			Set as indicated in ACS-4
47	M		Multiple Count
		F	15:8 80h
		R	7:0 00h = Reserved 01h..10h = Maximum number of sectors that shall be transferred per interrupt on READ/WRITE MULTIPLE commands 11h..FFh = Reserved
48			Set as indicated in ACS-4
49	M		Capabilities
			15:12 Set as indicated in ACS-4
		F	11 Shall be set to one
		F	10 Shall be set to one
			9:0 Set as indicated in ACS-4
50..52			Set as indicated in ACS-4
53	M		Field validity
		R	15:3 Reserved
		F	2 1=the fields reported in Word 88 are valid 0=the fields reported in Word 88 are not valid
		F	1 1=the fields reported in Words (70:64) are valid 0=the fields reported in Words (70:64) are not valid
		F	0 Obsolete
54..62			Set as indicated in ACS-4
63	M		Multiword DMA transfer
			15:3 Set as indicated in ACS-4
		F	2 Multiword DMA mode 2 and below are supported
		F	1 Multiword DMA mode 1 and below are supported
		F	0 Multiword DMA mode 0 is supported
64	M		PIO transfer modes supported
			15:2 Set as indicated in ACS-4
		F	1:0 PIO modes 3 and 4 supported
65	M		Minimum Multiword DMA transfer cycle time per Word
		F	15:0 Cycle time in nanoseconds
66	M		Manufacturer's recommended Multiword DMA transfer cycle time
		F	15:0 Cycle time in nanoseconds
67	M		Minimum PIO transfer cycle time without flow control
		F	15:0 Cycle time in nanoseconds

Key:
M = Support of the Word is mandatory.
O = Support of the Word is optional.
F = the content of the bit, field, or Word is fixed and does not change. For removable media devices, these values may change if media is removed or changed.
V = the contents of the bit, field, or Word is variable and may change depending on the state of the device or the commands processed by the device.
R = the content of the bit, field, or Word is reserved and shall be cleared to zero.

Table 101 – IDENTIFY DEVICE information (part 2 of 4)

Word	O/M	F/V	Description
68	M		Minimum PIO transfer cycle time with IORDY flow control
		F	15:0 Cycle time in nanoseconds
69..74			Set as indicated in ACS-4
75	O		Queue depth
		R	15:5 Reserved
		F	4:0 Maximum queue depth – 1
76	O		Serial ATA capabilities
		F	15 Supports READ LOG DMA EXT as equivalent to READ LOG EXT
		F	14 Supports Device Automatic Partial to Slumber transitions
		F	13 Supports Host Automatic Partial to Slumber transitions
		F	12 Supports NCQ priority information
		F	11 Supports Unload while NCQ commands outstanding
		F	10 Supports Phy event counters
		F	9 Supports receipt of host-initiated interface power management requests
		F	8 Supports NCQ feature set
		R	7:4 Reserved for future Serial ATA signaling speed grades
		F	3 Supports Serial ATA Gen3 signaling speed (6.0 Gbit/s)
		F	2 Supports Serial ATA Gen2 signaling speed (3.0 Gbit/s)
		F	1 Supports Serial ATA Gen1 signaling speed (1.5 Gbit/s)
		F	0 Shall be cleared to zero
77	O		Serial ATA Additional capabilities
		R	15:10 Reserved
			9 Supports Out Of Band Management Interface
		F	8 Power Disable feature always enabled
		F	7 DevSleep_to_ReducedPwrState
		F	6 Supports RECEIVE FPDMA QUEUED and SEND FPDMA QUEUED commands
		F	5 Supports NCQ NON-DATA command
		F	4 Supports NCQ Streaming
		V	3:1 Coded value indicating current negotiated Serial ATA signal speed
		F	0 Shall be cleared to zero

Key:
M = Support of the Word is mandatory.
O = Support of the Word is optional.
F = the content of the bit, field, or Word is fixed and does not change. For removable media devices, these values may change if media is removed or changed.
V = the contents of the bit, field, or Word is variable and may change depending on the state of the device or the commands processed by the device.
R = the content of the bit, field, or Word is reserved and shall be cleared to zero.

Table 101 – IDENTIFY DEVICE information (part 3 of 4)

Word	O/M	F/V	Description
78	O		Serial ATA features supported
		R	15:13 Reserved
		F	12 Supports Power Disable feature
		F	11 Supports Rebuild Assist
		F	10 Supports Device Initiated Interface Power Management Software Settings Preservation
		F	9 Supports Hybrid Information
		F	8 Supports Device Sleep
		F	7 Supports NCQ Autosense
		F	6 Supports software settings preservation
		F	5 Supports Hardware Feature Control
		F	4 Supports in-order data delivery
		F	3 Supports initiating interface power management
		F	2 Supports DMA Setup Auto-Activate optimization
		F	1 Supports non-zero buffer offsets in DMA Setup FIS
		F	0 Shall be cleared to zero
79	O		Serial ATA features enabled
		R	15:12 Reserved
		V	11 Rebuild Assist enabled
		V	10 Power Disable feature enabled
		V	9 Hybrid Information feature is enabled
		V	8 Device Sleep enabled
		V	7 Device Automatic Partial to Slumber transitions enabled
		V	6 Software settings preservation enabled
		V	5 Hardware Feature Control enabled
		V	4 In-order data delivery enabled
		V	3 Device initiating interface power management enabled
		V	2 DMA Setup Auto-Activate optimization enabled
		V	1 Non-zero buffer offsets in DMA Setup FIS enabled
		F	0 Shall be cleared to zero
80..87			Set as indicated in ACS-4
88			15:6 Set as indicated in ACS-4
		F	5 Ultra DMA mode 5 and below are supported
		F	4 Ultra DMA mode 4 and below are supported
		F	3 Ultra DMA mode 3 and below are supported
		F	2 Ultra DMA mode 2 and below are supported
		F	1 Ultra DMA mode 1 and below are supported
		F	0 Ultra DMA mode 0 is supported
89..92			Set as indicated in ACS-4
93		V	COMRESET result. The contents of this Word shall be cleared to zero.
94..221			Set as indicated in ACS-4

Key:
M = Support of the Word is mandatory.
O = Support of the Word is optional.
F = the content of the bit, field, or Word is fixed and does not change. For removable media devices, these values may change if media is removed or changed.
V = the contents of the bit, field, or Word is variable and may change depending on the state of the device or the commands processed by the device.
R = the content of the bit, field, or Word is reserved and shall be cleared to zero.

Table 101 – IDENTIFY DEVICE information (part 4 of 4)

Word	O/M	F/V	Description		
222			Transport Major Revision 0000h or FFFFh = device does not report version		
			Bits	**Description**	
		F	15:12	Transport Type 0h = Parallel 1h = Serial 2h..Dh Reserved Eh = Obsolete Fh = Reserved	
				Serial	**PCIe**
		R	11	Reserved	Reserved
		F	10	SATA Rev 3.5	Reserved
		F	9	SATA Rev 3.4	Reserved
		F	8	SATA Rev 3.3	Reserved
		F	7	SATA Rev 3.2	Reserved
		F	6	SATA Rev 3.1	Reserved
		F	5	SATA Rev 3.0	Reserved
		F	4	SATA Rev 2.6	Reserved
		F	3	SATA Rev 2.5	Reserved
		F	2	SATA II: Extensions	Reserved
			1	SATA 1.0a	Reserved
			0	ATA8-AST	Reserved
223		F	Transport Minor Revision		
224..255			Set as indicated in ACS-4		

Key:
M = Support of the Word is mandatory.
O = Support of the Word is optional.
F = the content of the bit, field, or Word is fixed and does not change. For removable media devices, these values may change if media is removed or changed.
V = the contents of the bit, field, or Word is variable and may change depending on the state of the device or the commands processed by the device.
R = the content of the bit, field, or Word is reserved and shall be cleared to zero.

13.2.2.2 Words 0..46

Word 0..46 shall be set as indicated in ACS-4.

13.2.2.3 Word 47, multiword PIO transfer

Bits 15:8 of Word 47 shall be set as indicated in ACS-4.

Bits 7:0 are used to indicate the maximum number of sectors that shall be transferred per interrupt on READ/WRITE MULTIPLE commands. This field shall be set to 16 or less (see 10.5.11.2).

13.2.2.4 Word 48

Word 48 shall be set as indicated in ACS-4.

13.2.2.5 Word 49, capabilities

Bits 15:12 of Word 49 shall be set as indicated in ACS-4.

Bit 11 of Word 49 is used to determine whether a device supports IORDY. This bit shall be set to one, indicating the device supports IORDY operation.

Bit 10 of Word 49 is used to indicate a device's ability to enable or disable the use of IORDY. This bit shall be set to one, indicating the device supports the disabling of IORDY. Disabling and enabling of IORDY is accomplished using the SET FEATURES command.

Bits 9:0 of Word 49 shall be set as indicated in ACS-4.

13.2.2.6 Words 50..52

Words 50..52 shall be set as indicated in ACS-4.

13.2.2.7 Word 53, field validity

Bit 0 shall be set to one. Bit 1 of Word 53 shall be set to one, the values reported in Words 64..70 are valid. Any device that supports PIO mode 3 or above, or supports Multiword DMA mode 1 or above, shall set bit 1 of Word 53 to one and support the fields contained in Words 64..70. Bit 2 of Word 53 shall be set to one indicating the device supports Ultra DMA and the values reported in Word 88 are valid. Bits 15:3 are reserved.

13.2.2.8 Words 54..62

Words 54..62 shall be set as indicated in ACS-4.

13.2.2.9 Word 63, Multiword DMA transfer

Bits 15:3 shall be set as indicated in ACS-4.

Bits 2:0 of Word 63 shall be set to one indicating that the device supports Multiword DMA modes 0, 1, and 2.

13.2.2.10 Word 64, PIO transfer modes supported

Bits 1:0 of Word 64 shall be set to one indicating that the device supports PIO modes 3 and 4.

Bits 15:2 shall be set as indicated in ACS-4.

13.2.2.11 Word 65, minimum Multiword DMA transfer cycle time per Word

Shall be set to indicate 120 ns.

13.2.2.12 Word 66, device recommended Multiword DMA cycle time

Shall be set to indicate 120 ns.

13.2.2.13 Word 67, minimum PIO transfer cycle time without flow control

Shall be set to indicate 120 ns.

13.2.2.14 Word 68, minimum PIO transfer cycle time with IORDY

Shall be set to indicate 120 ns.

13.2.2.15 Words 69..74

Words 69..74 shall be set as indicated in ACS-4.

13.2.2.16 Word 75, queue depth

This Word is as defined in the ACS-4 standard. The NCQ protocol supports at most 32 queued commands. With Native Command Queueing, the host shall issue only unique tag values for queued commands that have a value less than or equal to the value reflected in this field (e.g., for a device reporting a value in this field of 15, corresponding to a maximum of 16 outstanding commands, the host never uses a tag value greater than 15 if issuing Native Command Queueing commands).

13.2.2.17 Word 76, Serial ATA capabilities

If not 0000h or FFFFh, the device claims compliance with this specification and supports the signaling speed indicated in bits 3:1. Since Serial ATA supports generational compatibility, multiple bits may be set. Bit 0 is reserved (thus a Serial ATA device has at least one bit cleared to zero in this field and at least one bit set providing clear differentiation). If this field is not 0000h or FFFFh, Words 77..79 shall be valid. If this field is 0000h or FFFFh the device does not claim compliance with this specification and words 76..79 are not valid and shall be ignored.

Bit 15 is a copy of the READ LOG DMA EXT AS EQUIVALENT TO READ LOG EXT SUPPORTED bit (see 13.7.11.2.11).

Bit 14 is a copy of the DEVICE AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED bit (see 13.7.11.2.10).

Bit 13 is a copy of the HOST AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED bit (see 13.7.11.2.9).

Bit 12 is a copy of the NCQ PRIORITY INFORMATION SUPPORTED bit (see 13.7.11.2.8).

Bit 11 is a copy of the UNLOAD WHILE NCQ COMMANDS ARE OUTSTANDING SUPPORTED bit (see 13.7.11.2.7).

Bit 10 is a copy of the SATA PHY EVENT COUNTERS LOG SUPPORTED bit (see 13.7.11.2.6).

Bit 9 is a copy of the RECEIPT OF HOST INITIATED POWER MANAGEMENT REQUESTS SUPPORTED bit (see 13.7.11.2.5).

Bit 8 is a copy of the NCQ FEATURE SET SUPPORTED bit (see 13.7.11.2.4).

Bits (7..4) are reserved.

Bit 3 is a copy of the SATA GEN3 SIGNALING SPEED SUPPORTED bit (see 13.7.11.2.3).

Bit 2 is a copy of the SATA GEN2 SIGNALING SPEED SUPPORTED bit (see 13.7.11.2.2).

Bit 1 is a copy of the SATA GEN1 SIGNALING SPEED SUPPORTED bit (see 13.7.11.2.1).

Bit 0 shall be cleared to zero.

13.2.2.18 Word 77, Serial ATA additional capabilities

Word 77 reports additional optional capabilities supported by the device. Support for this Word is optional and if not supported, the Word shall be cleared to zero indicating the device has no support for Serial ATA additional capabilities.

Bits 15:10 are reserved.

Bit 9 is a copy of the OUT OF BAND MANAGEMENT INTERFACE SUPPORTED bit (see 13.7.11.2.29).

Bit 8 is a copy of the POWER DISABLE FEATURE ALWAYS ENABLED bit (see 13.7.11.2.16).

Bit 7 is a copy of the DEVSLEEP_TO_REDUCEDPWRSTATE CAPABILITY SUPPORTED bit (see 13.7.11.2.15).

Bit 6 is a copy of the SEND AND RECEIVE QUEUED COMMANDS SUPPORTED bit (see 13.7.11.2.14).

Bit 5 is a copy of the NCQ NON-DATA COMMAND SUPPORTED bit (see 13.7.11.2.13).

Bit 4 is a copy of the NCQ STREAMING SUPPORTED bit (see 13.7.11.2.12).

Bits (3:1) are a copy of the CURRENT NEGOTIATED SERIAL ATA SIGNAL SPEED field (see 13.7.11.3.1).

Bit 0 shall be cleared to zero.

13.2.2.19 Word 78, Serial ATA features supported

Word 78 reports the optional features supported by the device. If this Word is cleared to zero, then the Serial ATA features indicated by this Word are not supported.

Bits 15:13 are reserved.

Bit 12 is a copy of the POWER DISABLE FEATURE SUPPORTED bit (see 13.7.11.2.28).

Bit 11 is a copy of the REBUILD ASSIST SUPPORTED bit (see 13.7.11.2.27).

Bit 10 is a copy of the DIPM SSP PRESERVATION SUPPORTED bit (see 13.7.11.2.26).

Bit 9 is a copy of the HYBRID INFORMATION SUPPORTED bit (see 13.7.11.2.25).

Bit 8 is a copy of the DEVICE SLEEP SUPPORTED bit (see 13.7.11.2.24).

Bit 7 is a copy of the NCQ AUTOSENSE SUPPORTED bit (see 13.7.11.2.23).

Bit 6 is a copy of the SOFTWARE SETTINGS PRESERVATION SUPPORTED bit (see 13.7.11.2.22).

Bit 5 is a copy of the HARDWARE FEATURE CONTROL SUPPORTED bit (see 13.7.11.2.21).

Bit 4 is a copy of the IN-ORDER DATA DELIVERY SUPPORTED bit (see 13.7.11.2.20).

Bit 3 is a copy of the DEVICE INITIATED POWER MANAGEMENT SUPPORTED bit (see 13.7.11.2.19).

Bit 2 is a copy of the DMA SETUP AUTO-ACTIVATION SUPPORTED bit (see 13.7.11.2.18).

Bit 1 is a copy of the NON-ZERO BUFFER OFFSETS SUPPORTED bit (see 13.7.11.2.17).

Bit 0 shall be cleared to zero.

13.2.2.20 Word 79, Serial ATA features enabled

Word 79 reports that optional features supported by the device are enabled. This Word shall be supported if optional Word 78 is supported and shall not be supported if optional Word 78 is not supported.

Bits 15:12 are reserved.

Bit 11 is a copy of the REBUILD ASSIST ENABLED bit (see 13.7.11.3.11).

Bit 10 is a copy of the POWER DISABLE FEATURE ENABLED bit (see 13.7.11.3.10).

Bit 9 is a copy of the HYBRID INFORMATION ENABLED bit (see 13.7.11.3.12).

Bit 8 is a copy of the DEVICE SLEEP ENABLED bit (see 13.7.11.3.9).

Bit 7 is a copy of the AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS ENABLED bit (see 13.7.11.3.8).

Bit 6 is a copy of the SOFTWARE SETTINGS PRESERVATION ENABLED bit (see 13.7.11.3.7).

Bit 5 is a copy of the HARDWARE FEATURE CONTROL ENABLED bit (see 13.7.11.3.6).

Bit 4 is a copy of the IN-ORDER DATA DELIVERY ENABLED bit (see 13.7.11.3.5).

Bit 3 is a copy of the DEVICE INITIATED POWER MANAGEMENT ENABLED bit (see 13.7.11.3.4).

Bit 2 is a copy of the DMA SETUP FIS AUTO-ACTIVATE ENABLED bit (see 13.7.11.3.3).

Bit 1 is a copy of the NON-ZERO BUFFER OFFSETS ENABLED bit (see 13.7.11.3.2).

Bit 0 shall be cleared to zero.

13.2.2.21 Words 80..87

Words 80..87 shall be set as indicated in ACS-4.

13.2.2.22 Word 88, Ultra DMA modes

Bits 5:0 of Word 88 shall be set to one indicating that the device supports Ultra DMA modes 0, 1, 2, 3, 4, and 5. Bits 15:5 shall be set as indicated in ACS-4.

13.2.2.23 Words 89..92

Words 89..92 shall be set as indicated in ACS-4.

13.2.2.24 Word 93, hardware configuration test results

Word 93 shall be cleared to 0000h indicating that the Word is not supported.

13.2.2.25 Words 94..221

Words 94..221 shall be set as indicated in ACS-4.

13.2.2.26 Word 222, transport Major Revision

Bits 15:12 shall be set to 1h.

13.2.2.27 Word 223, transport Minor Revision

Word 223 shall be set as indicated in ACS-4.

13.2.2.28 Words 224..255

Words 224..255 shall be set as indicated in ACS-4.

13.2.3 IDENTIFY PACKET DEVICE

13.2.3.1 IDENTIFY PACKET DEVICE information

The IDENTIFY PACKET DEVICE information structure is defined in Table 102.

Table 102 – IDENTIFY PACKET DEVICE information (part 1 of 3)

Word	O/M	F/V	Description
0..48			Set as indicated in ACS-4
49	M		Capabilities
			15:12 Set as indicated in ACS-4
		F	11 Shall be set to one
		F	10 Shall be set to one
			9:0 Set as indicated in ACS-4
50..52			Set as indicated in ACS-4
53	M		Field validity
		R	15:3 Reserved
		F	2 1=the fields reported in Word 88 are valid
			0=the fields reported in Word 88 are not valid
		F	1 1=the fields reported in Words (70..64) are valid
			0=the fields reported in Words (70..64) are not valid
		F	0 Obsolete
54..62			Set as indicated in ACS-4
63	M		Multiword DMA transfer
			15:3 Set as indicated in ACS-4
		F	2 Multiword DMA mode 2 and below are supported
		F	1 Multiword DMA mode 1 and below are supported
		F	0 Multiword DMA mode 0 is supported
64	M		PIO transfer modes supported
			15:2 Set as indicated in ACS-4
		F	1:0 PIO modes 3 and 4 supported
65	M		Minimum Multiword DMA transfer cycle time per Word
		F	15:0 Cycle time in nanoseconds
66	M		Manufacturer's recommended Multiword DMA transfer cycle time
		F	15:0 Cycle time in nanoseconds
67	M		Minimum PIO transfer cycle time without flow control
		F	15:0 Cycle time in nanoseconds
68	M		Minimum PIO transfer cycle time with IORDY flow control
		F	15:0 Cycle time in nanoseconds
69..75			Set as indicated in ACS-4

Key:
M = Support of the Word is mandatory.
O = Support of the Word is optional.
F = the content of the bit, field, or Word is fixed and does not change. For removable media devices, these values may change if media is removed or changed.
V = the contents of the bit, field, or Word is variable and may change depending on the state of the device or the commands processed by the device.
R = the content of the bit, field, or Word is reserved and shall be cleared to zero.

Table 102 – IDENTIFY PACKET DEVICE information (part 2 of 3)

Word	O/M	F/V		Description
76	O			Serial ATA capabilities
		R	15	Reserved
		F	14	Supports Device Automatic Partial to Slumber transitions
		F	13	Supports Host Automatic Partial to Slumber transitions
		R	12:11	Reserved
		F	10	Supports Phy event counters
		F	9	Supports receipt of host-initiated interface power management requests
		R	8	Reserved
		R	7:4	Reserved for future Serial ATA signaling speed grades
		F	3	Supports Serial ATA Gen3 signaling speed (6.0 Gbit/s)
		F	2	Supports Serial ATA Gen2 signaling speed (3.0 Gbit/s)
		F	1	Supports Serial ATA Gen1 signaling speed (1.5 Gbit/s)
		F	0	Shall be cleared to zero
77	O			Serial ATA Additional capabilities
		R	15:6	Reserved
		F	5	Supports host environment detect
		F	4	Supports Device Attention on Slimline connected device
		V	3:1	Coded value indicating current negotiated Serial ATA signal speed
		F	0	Shall be cleared to zero
78	O			Serial ATA features supported
		R	15:7	Reserved
		F	6	Supports software settings preservation
		F	5	Supports asynchronous notification
		R	4	Reserved
		F	3	1 = Device supports initiating interface power management
		R	2:1	Reserved
		F	0	Shall be cleared to zero
79	O			Serial ATA features enabled
		R	15:8	Reserved
		V	7	Device Automatic Partial to Slumber transitions enabled
		V	6	Software settings preservation enabled
		V	5	Asynchronous notification enabled
		R	4	Reserved
		V	3	Device initiating interface power management enabled
		R	2:1	Reserved
		F	0	Shall be cleared to zero
80..87				Set as indicated in ACS-4

Key:
M = Support of the Word is mandatory.
O = Support of the Word is optional.
F = the content of the bit, field, or Word is fixed and does not change. For removable media devices, these values may change if media is removed or changed.
V = the contents of the bit, field, or Word is variable and may change depending on the state of the device or the commands processed by the device.
R = the content of the bit, field, or Word is reserved and shall be cleared to zero.

Table 102 – IDENTIFY PACKET DEVICE information (part 3 of 3)

Word	O/M	F/V	Description
88			15:6 Set as indicated in ACS-4
		F	5 Ultra DMA mode 5 and below are supported
		F	4 Ultra DMA mode 4 and below are supported
		F	3 Ultra DMA mode 3 and below are supported
		F	2 Ultra DMA mode 2 and below are supported
		F	1 Ultra DMA mode 1 and below are supported
		F	0 Ultra DMA mode 0 is supported
89..92			Set as indicated in ACS-4
93		V	COMRESET result. The contents of this Word shall be cleared to zero.
94..221			Set as indicated in ACS-4
222			Transport Major Revision 0000h or FFFFh = device does not report version **Bits** **Description** 15:12 Transport Type 0h = Parallel 1h = Serial 2h..Fh = Reserved **Serial** 11 Reserved 10 SATA Rev 3.5 9 SATA Rev 3.4 8 SATA Rev 3.3 7 SATA Rev 3.2 6 SATA Rev 3.1 5 SATA Rev 3.0 4 SATA Rev 2.6 3 SATA Rev 2.5 2 SATA II: Extensions 1 SATA 1.0a 0 ATA8-AST
		F	F: 15:12
		R	R: 11
		F	F: 10..2
223		F	Transport Minor Revision
224..255			Set as indicated in ACS-4

Key:
M = Support of the Word is mandatory.
O = Support of the Word is optional.
F = the content of the bit, field, or Word is fixed and does not change. For removable media devices, these values may change if media is removed or changed.
V = the contents of the bit, field, or Word is variable and may change depending on the state of the device or the commands processed by the device.
R = the content of the bit, field, or Word is reserved and shall be cleared to zero.

13.2.3.2 Words 0..48

Words 0..48 shall be set as indicated in ACS-4.

13.2.3.3 Word 49, capabilities

Bits 15:12 of Word 49 shall be set as indicated in ACS-4.

Bit 11 of Word 49 is used to determine whether a device supports IORDY. This bit shall be set to one, indicating the device supports IORDY operation.

Bit 10 of Word 49 is used to indicate a device's ability to enable or disable the use of IORDY. This bit shall be set to one, indicating the device supports the disabling of IORDY. Disabling and enabling of IORDY is accomplished using the SET FEATURES command.

Bits 9:0 of Word 49 shall be set as indicated in ACS-4.

13.2.3.4 Words 50..52

Words 50..52 shall be set as indicated in ACS-4.

13.2.3.5 Word 53, field validity

Bit 0 shall be set to one. Bit 1 of Word 53 shall be set to one, the values reported in Words 64..70 are valid. Any device that supports PIO mode 3 or above, or supports Multiword DMA mode 1 or above, shall set bit 1 of Word 53 to one and support the fields contained in Words 64..70. Bit 2 of Word 53 shall be set to one indicating the device supports Ultra DMA and the values reported in Word 88 are valid. Bits 15:3 are reserved.

13.2.3.6 Words 54..62

Words 54..62 shall be set as indicated in ACS-4.

13.2.3.7 Word 63, Multiword DMA transfer

If bit 15 of Word 62 is set to one, then bit 2 of Word 63 shall be cleared to zero. If bit 15 of Word 62 is cleared to zero, then bit 2 of Word 63 shall be set to one.

If bit 15 of Word 62 is set to one, then bit 1 of Word 63 shall be cleared to zero. If bit 15 of Word 62 is cleared to zero, then bit 1 of Word 63 shall be set to one.

If bit 15 of Word 62 is set to one, then bit 0 of Word 63 shall be cleared to zero. If bit 15 of Word 62 is cleared to zero, then bit 0 of Word 63 shall be set to one.

Bits 15:3 shall be set as indicated in ACS-4.

13.2.3.8 Word 64, PIO transfer modes supported

Bits 1:0 of Word 64 shall be set to one indicating that the device supports PIO modes 3 and 4.

Bits 15:2 shall be set as indicated in ACS-4.

13.2.3.9 Word 65, minimum Multiword DMA transfer cycle time per word

Shall be set to indicate 120 ns.

13.2.3.10 Word 66, device recommended Multiword DMA cycle time

Shall be set to indicate 120 ns.

13.2.3.11 Word 67, minimum PIO transfer cycle time without flow control

Shall be set to indicate 120 ns.

13.2.3.12 Word 68, minimum PIO transfer cycle time with IORDY

Shall be set to indicate 120 ns.

13.2.3.13 Words 69..75

Words 69..75 shall be set as indicated in ACS-4.

13.2.3.14 Word 76, Serial ATA capabilities

Word 76 shall have the content described for IDENTIFY DEVICE data Word 76 bits 0..7, 9, 10, 13, and 14, except for reserved bits 8, 11, 12, and 15.

13.2.3.15 Word 77, Serial ATA additional capabilities

Support for this Word is optional and if not supported, the Word shall be cleared to zero indicating the device has no support for Serial ATA additional capabilities.

Bit 0 shall be cleared to zero.

Bits 3:1 are a coded value to indicate the current Serial ATA Phy speed that device is communicating at. Table 121 defines these values.

Bit 4 if set to one, indicates that the device supports Device Attention capability in the Slimline connector. Devices that do not use the Slimline connector shall clear this bit to zero.

Bit 5 if set to one, indicates that the device supports the ability to detect whether or not the device is in a manufacturing test or PC application environment.

> NOTE 51 – In the case of system configurations that have more than one Phy link in the data path (e.g., port multiplier), the indicated speed is only relevant for the link between the device Phy and its immediate host Phy. It is possible for each link in the data path to negotiate a different Serial ATA signaling speed.

Bits 15:6 are reserved.

13.2.3.16 Word 78, Serial ATA features supported

Word 78 reports the optional features supported by the device. If this Word is cleared to zero, then the Serial ATA features indicated by this Word are not supported.

Bit 0 shall be cleared to zero.

Bits 2:1 are reserved.

Bit 3 indicates whether the device supports initiating power management requests to the host. If set to one, the device supports initiating interface power management requests. If cleared to zero, the device does not support initiating power management requests. A device may support reception of power management requests initiated by the host as described in the definition of bit 9 of Word 76 without supporting initiating such power management requests as indicated by this bit. Devices shall support host-initiated interface power management, device-initiated interface power management, or both. Devices shall set this bit to one if Word 76 bit 9 is cleared to zero.

Bit 4 is reserved.

Bit 5 indicates whether the device supports asynchronous notification to indicate to the host that attention is required. If set to one, the device supports initiating notification events. If cleared to zero, the device does not support initiating notification events.

EXAMPLE – An example of an event that the device may need attention for includes a media change.

Asynchronous device notification as defined in 13.8.2.

Bit 6 indicates whether the device supports software settings preservation as defined in 13.5. If set to one, the device supports software settings preservation across COMRESET. If cleared to zero, the device clears all software settings if a COMRESET occurs.

Bits 15:7 are reserved.

13.2.3.17 Word 79, Serial ATA features enabled

Word 79 bit 0 shall have the content described for IDENTIFY DEVICE data Word 79 bit 0 (see 13.2.2.20).

Word 79 bit 1 is reserved.

Word 79 bit 2 is reserved.

Word 79 bit 3 shall have the content described for IDENTIFY DEVICE data Word 79 bit 3 (see 13.2.2.20).

Word 79 bit 4 is reserved.

Word 79 bit 5 indicates whether device support for asynchronous notification to indicate to the host that attention is required is enabled. If set to one, the device may initiate notification events. If cleared to zero, the device shall not initiate notification events. This field shall be cleared to zero by default.

EXAMPLE – An example of an event that the device may need attention for includes a media change.

Asynchronous notification as defined in 13.8.

Word 79 bit 6 shall have the content described for IDENTIFY DEVICE data Word 79 bit 6 (see 13.2.2.20).

Word 79 bit 7 shall have the content described for IDENTIFY DEVICE data Word 79 bit 7 (see 13.2.2.20).

Word 79 bits 15:8 are reserved.

13.2.3.18 Words 80..87

Words 80..87 shall be set as indicated in ACS-4.

13.2.3.19 Word 88, Ultra DMA modes

Bits 5:0 of Word 88 shall be set to one indicating that the device supports Ultra DMA modes 0, 1, 2, 3, 4, and 5. Bits 15:5 shall be set as indicated in ACS-4.

13.2.3.20 Words 89..92

Words 89..92 shall be set as indicated in ACS-4.

13.2.3.21 Word 93, hardware configuration test results

Word 93 shall be cleared to 0000h indicating that the Word is not supported.

13.2.3.22 Words 94..221

Words 94..221 shall be set as indicated in ACS-4.

13.2.3.23 Word 222, transport major revision

Bits 15:12 shall be set to 1h.

13.2.3.24 Word 223, transport minor revision

Word 223 shall be set as indicated in ACS-4.

13.2.3.25 Words 224.255

Words 224..255 shall be set as indicated in ACS-4.

13.2.4 Determining support for Serial ATA features

Software should verify a device's Serial ATA capabilities by reading the relevant bits in Words 76..79 of the IDENTIFY (PACKET) DEVICE data. A device claims compliance with this specification by setting IDENTIFY (PACKET) DEVICE Word 76 appropriately.

Although Serial ATA was first introduced in the ATA/ATAPI specification material in the ATA/ATAPI-7 revision, it is unable to assume that if Word 80 (Major version number) of the IDENTIFY (PACKET) DEVICE data is read with support of ATA/ATAPI-7 or later that the device supports Serial ATA or any specific Serial ATA features.

13.3 SET FEATURES command

13.3.1 SET FEATURES command overview

Devices are informed of host capabilities and have optional features enabled/disabled through the SET FEATURES command defined in the ACS-4 standard. Serial ATA features are controlled using a features value as defined in Table 103.

Table 103 – Features enable/disable values

FEATURES field (7:0) Value	Description
10h	Enable use of Serial ATA feature
90h	Disable use of Serial ATA feature

The COUNT field (7:0) contains the specific Serial ATA feature to enable or disable. The specific Serial ATA features that SET FEATURES is applicable are defined in Table 104.

Table 104 – Feature identification values

COUNT field (7:0) Value	Description
00h	Reserved
01h	Non-zero buffer offset in DMA Setup FIS feature
02h	DMA Setup FIS Auto-Activate optimization feature
03h	Device-initiated interface power state transitions feature
04h	Guaranteed In-Order Data Delivery feature
05h	Asynchronous Notification feature
06h	Software Settings Preservation feature
07h	Device Automatic Partial to Slumber transitions feature
08h	Enable Hardware Feature Control feature
09h	Enable Device Sleep feature
0Ah	Enable/Disable Hybrid Information feature
0Bh	Enable/Disable Power Disable feature
0Ch..FFh	Reserved for future Serial ATA definition

13.3.2 Enable/disable Non-zero offsets in DMA Setup FIS feature

A COUNT field (7:0) value of 01h is used by the host to enable or disable the Non-zero buffer offsets in the DMA Setup FIS feature if the device utilizes the First-party DMA mechanism (see 13.6.2.2). By default, the Non-zero buffer offsets in the DMA Setup FIS feature is disabled. Enabling the Non-zero buffer offsets in the DMA Setup FIS feature is useful for performing out of order data delivery within commands, (e.g., delivering the last half of the data before the first half of the data, or to support segmentation of large First-party DMA operations into multiple data phases). The enable/disable state for the Non-zero offsets in DMA Setup FIS feature shall be preserved across software reset. The enable/disable state for the Non-zero offsets in DMA Setup FIS feature shall be reset to its default state upon COMRESET.

13.3.3 Enable/disable DMA Setup FIS Auto-Activate optimization feature

A COUNT field (7:0) value of 02h is used by the host to enable or disable the DMA Setup FIS Auto-Activate optimization feature for automatically activating transfer of the first Host-to-Device Data FIS following a DMA Setup FIS with a host-to-device transfer direction. For transfers from the host to the device, First-party DMA transfers require a sequence of DMA Setup FIS followed by a DMA Activate FIS to initiate the transfer. The Auto-Activate optimization allows the DMA Setup FIS operation to imply immediate activation thereby eliminating the need for the additional separate

DMA Activate FIS to start the transfer. Enabling the optimization notifies the device that the HBA implementation allows the DMA Setup FIS to include the Auto-Activate bit to trigger immediate transfer following receipt and processing of the DMA Setup FIS. By default, the optimization is disabled (see 10.5.9.4.2). The enable/disable state for the auto-activate optimization shall be preserved across software reset. The enable/disable state for the auto-activate optimization shall be reset to its default state upon COMRESET.

13.3.4 Enable/disable Device-initiated interface power state transitions feature

A COUNT field (7:0) value of 03h is used by the host to enable or disable device initiation of interface power state transitions. By default, the device is not permitted to attempt interface power state transitions by issuing PMREQ_P$_P$ or PMREQ_S$_P$ to the host. The host may enable device initiation of such interface power state transitions for such cases where it may be desirable for the device to attempt initiating such transitions. The enable/disable state for device initiated power management shall persist across software reset. The enable/disable state shall be reset to its default disabled state upon COMRESET only if the device does not support Device Initiated Interface Power Management Software Settings Preservation (see 13.2.2.19). If Software Settings Preservation is enabled and Device Initiated Interface Power Management Software Settings Preservation is supported (see 13.7.11.2.26), then the enable/disable state shall persist across a COMRESET.

If device initiated interface power management is enabled, the device shall not attempt to initiate an interface power state transition between reset and the delivery of the device reset signature.

13.3.5 Enable/disable Guaranteed In-Order Data Delivery feature

A COUNT field (7:0) value of 04h is used by the host to enable or disable the Guaranteed In-Order Data Delivery feature if the device utilizes the First-party DMA mechanism and non-zero buffer offsets in the DMA Setup FIS. By default, the Guaranteed In-Order Data Delivery feature is disabled. Enabling the Guaranteed In-Order Data Delivery feature is useful for segmenting large I/O processes into multiple atomic data phases using non-zero buffer offsets in the DMA Setup FIS, while minimizing the complexity that may be imposed on the host with out-of-order data delivery. The enable/disable state for the Guaranteed In-Order Data Delivery feature shall be preserved across software reset. The enable/disable state for the Guaranteed In-Order Data Delivery feature shall be reset to its default state upon COMRESET.

13.3.6 Enable/disable Asynchronous Notification feature

A COUNT field (7:0) value of 05h is used by the host to enable or disable the Asynchronous Notification feature. By default, the Asynchronous Notification feature is disabled. The host may enable the Asynchronous Notification feature in order to allow the device to request attention without the host polling.

NOTE 52 - This may be useful to avoid polling for media change events in ATAPI devices.

The enable/disable state for the Asynchronous Notification feature shall be preserved across software reset. The enable/disable state for the Asynchronous Notification feature shall be reset to its default state upon COMRESET.

13.3.7 Enable/disable Software Settings Preservation feature

A COUNT field (7:0) value of 06h is used by the host to enable or disable the Software Settings Preservation feature, as defined in 13.5. By default, if the device supports software settings preservation, the feature is enabled on power-up. The enable/disable state for the Software Settings Preservation feature shall persist across software reset. The enable/disable state for the Software Settings Preservation feature shall be reset to its default state upon COMRESET. The host may disable the Software Settings Preservation feature in order to not preserve software settings across COMRESET (and make COMRESET equivalent to hardware reset in Parallel ATA).

13.3.8 Enable/disable Device Automatic Partial to Slumber transitions feature

A COUNT field (7:0) value of 07h is used by the host to enable or disable the Device Automatic Partial to Slumber transitions feature. By default, if the device supports the Device Automatic Partial to Slumber transitions feature, the feature is disabled on power-up. The enable/disable state for the Device Automatic Partial to Slumber transitions feature shall persist across software reset. The enable/disable state for the Device Automatic Partial to Slumber transitions feature shall be reset to its default state upon COMRESET.

The Device Automatic Partial to Slumber transitions feature shall not be enabled if the Device-Initiated Interface Power State transitions feature is disabled. Attempting to enable the Device Automatic Partial to Slumber transitions feature while the Device-Initiated Interface Power State transitions feature is disabled shall result in the device aborting the SET FEATURES command. Attempting to disable Device Automatic Partial to Slumber transitions if it is already disabled shall have no effect and the device shall return successful completion of the SET FEATURES command.

13.3.9 Enable Hardware Feature Control

A COUNT field (7:0) value of 08h is used by the host to enable the extended uses of the Hardware Feature Control pin(s). See 13.10 for additional information about Hardware Feature Control.

The extended uses of the Hardware Feature Control pin(s) shall be disabled by power-on reset.

Table 105 defines function identifiers used to enable specific extended uses of the Hardware Feature Control pin(s) in the LBA field (15:0) of the Enable Hardware Feature Control command.

Table 105 – Extended Uses of the Hardware Feature Control pin(s)

Function Identifier	Description	Preserved Across Software Reset	Preserved Across COMRESET
0000h	Reserved	na	na
0001h	Direct Head Unload (DHU) (see 13.19)	Y	Yes, regardless of SSP setting.
0002h to EFFFh	Reserved	na	na
F000h to FFFFh	Vendor specific	Vendor specific	Vendor specific

On successful completion of this command:

a) the CURRENT HARDWARE FEATURE CONTROL IDENTIFIER field (see 13.7.11) shall be set to the value in the LBA field (15:0);
b) the HARDWARE FEATURE CONTROL ENABLED bit (see 13.7.11.3.6) shall be set to one; and
c) the behavior of Hardware Feature Control pin(s) is specified by Table 105.

The device shall return command aborted if:

a) the HARDWARE FEATURE CONTROL SUPPORTED bit (see 13.7.11.2.21) is cleared to zero;
b) the value in the LBA field (15:0) is not equal to the SUPPORTED HARDWARE FEATURE CONTROL IDENTIFIER field (see 13.7.11.3.14); or
c) the CURRENT HARDWARE FEATURE CONTROL IDENTIFIER field (see 13.7.11.3.13) is non-zero.

13.3.10 Enable/disable Device Sleep feature

A COUNT field (7:0) value of 09h is used by the host to enable or disable the Device Sleep feature. If the value in FEATURES field (7:0) is set to 10h, then the device shall set IDENTIFY DEVICE data Word 79, bit 8, to one. If the value in Features (7:0) is set to 90h, then the device shall clear IDENTIFY DEVICE data Word 79, bit 8, to zero. As a result of processing a power on reset, the Device Sleep feature shall be disabled.

If:
a) the host attempts to enable or disable the Device Sleep feature; and
b) the Device Sleep feature is not supported (i.e., IDENTIFY DEVICE data Word 78 bit 8 is cleared to zero),

then the device shall return command aborted.

If the host attempts to enable the Device Sleep feature and the Power Disable feature is enabled (i.e., IDENTIFY DEVICE data Word 79 bit 10 is set to one), then the device shall return command aborted.

The host should ensure the Power Disable feature is disabled before enabling the Device Sleep feature.

13.3.11 Enable/disable Hybrid Information feature

13.3.11.1 Enable/disable Hybrid Information feature overview

See 13.20 for additional information about the Hybrid Information feature.

The Enable/Disable Hybrid Information subcommand:
a) enables the Hybrid Information feature and the non-volatile caching medium; or
b) disables the Hybrid Information feature and leaves the non-volatile caching medium in a vendor specific state.

The device shall return command aborted if the Hybrid Information feature is not supported.

If the Hybrid Information feature is enabled, then it shall remain enabled across all resets (e.g., power cycles), except as specified in 13.20.6.

13.3.11.2 Enable Hybrid Information subcommand

The Enable Hybrid Information subcommand enables the Hybrid Information feature.

If the Hybrid Information feature is currently enabled (i.e., the ENABLED field (see 13.7.8.2.3) in the Hybrid Information log is set to FFh), then the device shall return command aborted.

If the Hybrid Information feature is currently disabled, then the device shall:
a) enable the Hybrid Information feature (i.e., set IDENTIFY DEVICE data Word 79 bit 9 to one);
b) set the ENABLED field (see 13.7.8.2.3) in the Hybrid Information log to FFh;
c) increment the ENABLE COUNT field (see 13.7.8.2.14) by one in the Hybrid Information log; and
d) enable the use of the non-volatile caching medium.

13.3.11.3 Disable Hybrid Information subcommand

The Disable Hybrid Information subcommand disables the Hybrid Information feature.

If the Hybrid Information feature is currently enabled (i.e., the ENABLED field in the Hybrid Information log is set to FFh), then the device shall:
a) disable the Hybrid Information feature (i.e., clear IDENTIFY DEVICE data Word 79 bit 9 to zero);
b) clear the ENABLED field (see 13.7.8.2.3) in the Hybrid Information log to 00h; and
c) change the Hybrid Priority for all logical sectors in the non-volatile caching medium to zero.

If the Hybrid Information feature is currently disabled, then the device should return command completed with no error.

13.3.12 Enable/Disable Power Disable feature

A COUNT field (7:0) value of 0Bh is used by the host to enable or disable the Power Disable feature (see 8.6).

If the POWER DISABLE FEATURE SUPPORTED bit (see 13.7.11.3.10) is cleared to zero, then the device shall return command aborted.

If the host specified that the Power Disable feature is to be:

a) enabled and the Device Sleep feature is enabled (i.e., IDENTIFY DEVICE data Word 79 bit 8 is set to one); or
b) disabled and the POWER DISABLE FEATURE ALWAYS ENABLED bit (see 13.7.11.2.16) is set to one,

then the device shall return command aborted.

If the host specified that the Power Disable feature is to be:

a) disabled and the POWER DISABLE FEATURE ENABLED bit (see 13.7.11.3.10) is cleared to zero;
b) enabled and the POWER DISABLE FEATURE ENABLED bit is set to one; or
c) enabled and the power disable feature always enabled bit is set to one,

then the device shall return command completion with no error.

If the host specified that the Power Disable feature is to be enabled and the POWER DISABLE FEATURE ENABLED bit is cleared to zero, then the device shall:

1) enable the Power Disable feature;
2) set the POWER DISABLE FEATURE ENABLED bit to one; and
3) return command completion with no error.

The host should ensure the Device Sleep feature is disabled before enabling the Power Disable feature.

13.4 Device Configuration Overlay (obsolete)

For information on the Device Configuration Overlay feature set and commands, see SATA Revision 3.1. The feature set is obsolete in ACS-4.

13.5 Software settings preservation (optional)

13.5.1 Software settings preservation overview

If a device is enumerated, software configures the device using SET FEATURES and other commands. These software settings are often preserved across software reset but not necessarily across COMRESET. In Parallel ATA, only commanded hardware resets may occur, thus legacy mode software only reprograms settings that are cleared to zero for the particular type of reset it has issued. In Serial ATA, COMRESET is equivalent to hardware reset and a non-commanded COMRESET may occur if there is an asynchronous loss of signal. Since COMRESET is equivalent to hardware reset, in the case of an asynchronous loss of signal some software settings may be lost without legacy mode software knowledge. In order to avoid losing important software settings without legacy mode driver knowledge, the software settings preservation ensures that the value of important software settings is maintained across a COMRESET. Software settings preservation may be enabled or disabled using SET FEATURES with a feature identification value of 06h (see 13.3.7). If a device supports software settings preservation, the feature shall be enabled by default.

The software settings that shall be preserved across COMRESET are listed below. The device is only required to preserve the indicated software setting if it supports the particular feature/command the setting is associated with.

INITIALIZE DEVICE PARAMETERS, device settings established with the INITIALIZE DEVICE PARAMETERS command. This command is obsolete in ACS-4, and was last defined in ATA/ATAPI-5.

NCQ NON-DATA (DEADLINE HANDLING) (see 13.6.6.4), the state of WDNC bit and RDNC bit.

Power Management Feature Set Standby Timer, the Standby timer used in the Power Management feature set.

Read/Write Stream Error log, the Read Stream Error log and Write Stream Error Logs (accessed using the General Purpose Logging (GPL) feature set commands).

SANITIZE FREEZE LOCK EXT, the Frozen state (see ACS-4).

Security mode state, the security mode state established by Security Mode feature set commands (see ACS-4). The device shall not transition to a different security mode state based on a COMRESET. SECURITY FREEZE LOCK, the Frozen mode setting established by the SECURITY FREEZE LOCK command.

SECURITY UNLOCK, the unlock counter that is decremented as part of a failed SECURITY UNLOCK command attempt.

SET FEATURES (Advanced Power Management Enable/Disable), the advanced power management enable/disable setting established by the SET FEATURES command with subcommand code of 05h or 85h. The advanced power management level established in the COUNT field (7:0) register if advanced power management is enabled (SET FEATURES subcommand code 05h) shall also be preserved.

SET FEATURES (Read Look-Ahead), the read look-ahead enable/disable setting established by the SET FEATURES command with subcommand code of 55h or AAh.

SET FEATURES (Release Interrupt), the release interrupt enable/disable setting established by the SET FEATURES command with a subcommand code of 5Dh or DDh.

SET FEATURES (SERVICE Interrupt), the SERVICE interrupt enable/disable setting established by the SET FEATURES command with a subcommand code of 5Eh or DEh.

SET FEATURES (Set Transfer Mode), PIO, Multiword, and ultra direct memory access (UDMA) transfer mode settings established by the SET FEATURES command with subcommand code of 03h.

SET FEATURES (Reverting to Defaults), the reverting to power-on defaults enable/disable setting established by the SET FEATURES command with a subcommand code of CCh or 66h.

SET FEATURES (Write Cache Enable/Disable), the write cache enable/disable setting established by the SET FEATURES command with subcommand code of 02h or 82h.

SET FEATURES (Device Sleep), Device Sleep enable/disable setting established by the SET FEATURES command (see 13.3.10).

SET FEATURES (Device Initiated Interface Power Management), the Device Intiated Interface Power Management enable/disable setting (i.e., IDENTIFY DEVICE data Word 79 bit 3) established by the SET FEATURES command (see 13.3.4).

SET MAX ADDRESS (EXT), the maximum logical block address (LBA) specified in SET MAX ADDRESS or SET MAX ADDRESS EXT.

SET MULTIPLE MODE, the block size established with the SET MULTIPLE MODE command.

Write-Read-Verify feature set, the contents of IDENTIFY DEVICE data Word 120 bit 1, Words 210..211, and Word 220 bits (7:0). The device shall not return to its Write-Read-Verify factory default setting after processing a COMRESET.

13.5.2 Warm reboot considerations (informative)

During a system reboot, the security settings maintained by software settings preservation may cause an error condition. Some system implementations choose to reboot the system by sending a COMRESET to the device. If the device has software settings preservation enabled, the security settings remain in the Unlock / Frozen State (SEC6). If in the Unlock / Frozen State (SEC6), system software sending a SECURITY UNLOCK command with the password is aborted by the device. System software should implement recommendations in this section to avoid the password entered by the user being aborted.

It is recommended that system software not prompt for the user password during a warm reboot. If system software detects that the device is in the SEC5 state during the warm reboot, the SECURITY FREEZE command may be issued by system software to enter the Unlock / Frozen State (SEC6).

The SEC states and SECURITY commands are described in ACS-4.

NOTE 53 – Note that these recommendations do not apply for external SATA devices.

13.6 Native Command Queuing (NCQ) feature set (optional)

13.6.1 NCQ feature set overview

This section defines a simple and streamlined command queuing model for Serial ATA.

Devices that support the NCQ feature set shall:

a) report support for the NCQ feature set (i.e., IDENTIFY DEVICE data Word 76 bit 8 is set to one);
b) report support for the general purpose logging feature set (i.e., IDENTIFY DEVICE data Word 84 bit 5 is set to one); and
c) implement the Queued Error log (see 13.7.4).

The following commands are mandatory for devices that implement the NCQ feature set:

a) READ FPDMA QUEUED; and
b) WRITE FPDMA QUEUED.

The following commands are optional for devices that implement the NCQ feature set:

a) NCQ NON-DATA;
b) RECEIVE FPDMA QUEUED; and
c) SEND FPDMA QUEUED.

NCQ NON-DATA is the only NCQ command that is performed with no data transfer.

READ FPDMA QUEUED and WRITE FPDMA QUEUED commands have transfer sizes of logical sector size multiples.

RECEIVE FPDMA QUEUED and SEND FPDMA QUEUED commands have transfer sizes of 512 byte multiples.

READ FPDMA QUEUED and RECEIVE FPDMA QUEUED commands transfer data from the device to the host.

WRITE FPDMA QUEUED and SEND FPDMA QUEUED commands transfer data from the host to the device.

The native queuing definition utilizes the reserved 32 bit field in the Set Device Bits FIS to convey the pending status for each of up to 32 outstanding commands. The BSY bit in the Status register conveys only the device's readiness to receive another command, and does not convey the completion status of queued commands. Upon receipt of a new command, the device clears the BSY bit to zero before proceeding to process received commands. The 32 protocol specific bits in the Set Device Bits FIS are handled as a 32-element array of active command bits (referred to as ACT bits), one for each possible outstanding command, and the array is bit significant such that bit "n" in the array corresponds to the pending status of the command with tag "n."

Data returned by the device (or transferred to the device) for queued commands use the First Party DMA mechanism to cause the host controller to select the appropriate destination/source memory buffer for the transfer. The memory handle used for the buffer selection is the same as the tag associated with the command. For traditional desktop host controllers, the handle may be used to index into a vector of pointers to pre-constructed scatter/gather lists (often referred to as physical region descriptor tables or simply Physical Region Descriptor (PRD) tables) in order to establish the proper context in the host's DMA engine. The First-party DMA Data Phase is defined as the period from reception of a DMA Setup FIS until either the associated transfer count is exhausted or the ERR bit in the shadow Status register is set. During this period the host may not issue new commands to the device nor may the device signal new command completions to the host.

Status is returned by updating the 32-element bit array in the Set Device Bits FIS for successful completions. For failed commands, the device halts processing commands allowing host software

or controller firmware to intervene and resolve the source of the failure, by using the general purpose logging feature set, before processing is again explicitly restarted.

13.6.2 Native Command Queuing (NCQ) feature set definition

13.6.2.1 Command issue mechanism

The Serial ATA transmission protocol is sensitive to the state of the BSY bit in the Shadow Status register that provides write protection to the shared Shadow Command Block registers. Since the Shadow Command Block registers may be safely written only if the BSY bit is cleared to zero, the BSY bit conventions defined in the Transport layer shall be adhered to, and issuing a new command shall only be attempted if the BSY bit is cleared to zero. If the BSY bit in the Shadow Status register is cleared to zero, another command may be issued to the device.

The state of the BSY bit in the Shadow Status register shall be checked prior to attempting to issue a new queued command. If the BSY bit is set to one, issuing the next command shall be deferred until the BSY bit is cleared to zero. It is desirable to minimize such command issue deferrals, so devices should clear the BSY bit to zero in a timely manner. Host controllers may have internal designs that mitigate the need for host software to block on the state of the BSY bit.

The native queuing commands include a tag value that identifies the command. The tag value is in the range 0 to 31 inclusive, and is conveyed in the Register Host to Device FIS if the command is issued. For devices that report a value less than 31 in their IDENTIFY DEVICE data Word 75, the host shall issue only unique tag values that are less than or equal to the value reported.

Upon issuing a new native queued command, the bit in the SActive register corresponding to the tag value of the command being issued shall be set to one by the HBA prior to the command being transmitted to the device. See SActive register and associated access conventions as defined in 14.2.5.

Upon accepting the command, the device shall clear the BSY bit to zero if it is prepared to receive another command by transmitting a Register Device to Host FIS with the BSY bit cleared to zero in the Status field of the FIS, and the Interrupt bit cleared to zero.

13.6.2.2 Data delivery mechanism

The First-party DMA mechanism is used by the device to transmit (or receive) data for an arbitrary queued command. The command's tag value shall also be the DMA Buffer Identifier used to uniquely identify the source/destination memory buffer for the transfer.

The DMA Setup FIS is used by the device to select the proper transfer buffer prior to each data transfer. Only a single DMA Setup FIS is required at the beginning of each transfer and if the transfer spans multiple Data FISes a new DMA Setup FIS is not required before each Data FIS. Serial ATA host controller hardware shall account for the DMA Buffer Identifier being a value between 0 and 31 and the host controller shall select the proper transfer buffer based on such an index.

For data transfers from the host to the device, an optimization to the First-party DMA mechanism is included to eliminate one transaction by allowing the requested data to immediately be transmitted to the device following such a request without the need for a subsequent DMA Activate FIS for starting the flow of data. This optimization to the First-party DMA mechanism is defined in 10.5.9.4.2.

If non-zero buffer offsets in the DMA Setup FIS are not enabled (see 13.3.2) or not supported (see 13.2.2), the data transfer for a command shall be satisfied to completion following a DMA Setup FIS before data transfer for a different command may be started. Host controllers are not required to preserve DMA engine context upon receipt of a new DMA Setup FIS, and if non-zero buffer

offsets are not enabled or not supported, a device is unable to resume data transfer for a previously abandoned context at the point where it left off.

If the host controller hardware supports non-zero buffer offsets in the DMA Setup FIS and use of non-zero offsets is enabled, and if guaranteed in-order data delivery is either not supported by the device (see 13.2.2) or is disabled (see 13.3.5), the device may return (or receive) data for a given command out of order (i.e., returning data for the last half of the command first). In this case the device may also interleave partial data delivery for multiple commands provided the device keeps track of the appropriate buffer offsets.

> NOTE 54 – An example of interleaving partial data delivery for multiple command is data for the first half of command 0 may be delivered followed by data for the first half of command 1 followed by the remaining data for command 0.

By default use of non-zero buffer offsets is disabled. See 13.3.2 for information on enabling non-zero buffer offsets for the DMA Setup FIS.

If the host controller hardware supports non-zero buffer offsets in the DMA Setup FIS and use of non-zero offsets is enabled, and if the device supports guaranteed in-order data delivery and guaranteed in-order data delivery is enabled, then the device may use multiple DMA Setup FISes to satisfy a particular I/O process. If multiple DMA Setup FISes are used, then the data shall be delivered in-order, starting at the first LBA. In this case the device may not interleave partial data delivery for either individual or multiple commands.

> NOTE 55 – Data for the first half of a command may be delivered using one DMA Setup FIS and one or more subsequent Data FISes, followed by the remaining data for that command, delivered using a second DMA Setup FIS and one or more subsequent Data FISes.

Non-zero buffer offsets are used as in the more general out-of-order data delivery case described above. By default use of guaranteed in-order data delivery is disabled.

For selecting the memory buffer for data transfers, the DMA Setup FIS is issued by the device. The DMA Setup FIS fields are defined in Figure 336 (see 10.5.9).

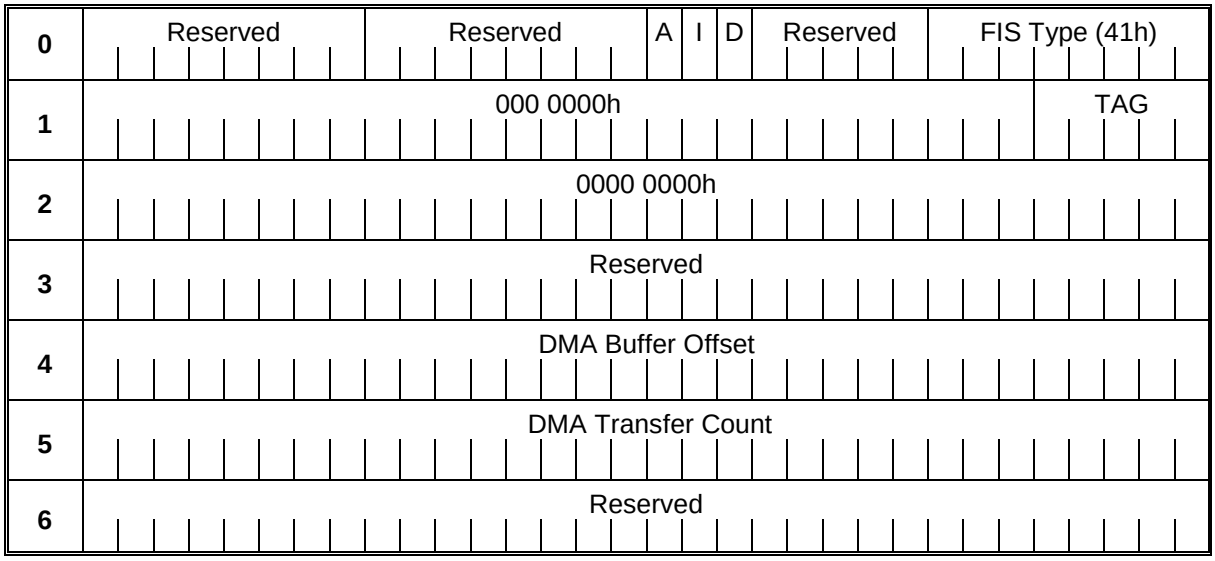


Figure 336 – DMA Setup FIS definition for memory buffer selection

Field Definitions

A — As defined in 10.5.9, including additional details according to 10.5.9.4.2. For DMA Setup with transfer direction from device to host, this bit shall be cleared to zero.

I — Interrupt, NCQ does not make use of an interrupt following the data transfer phase (after the transfer count is exhausted). The Interrupt bit shall be cleared to zero.

D — As defined in 10.5.9. Since the DMA Setup FIS is only issued by the device for the queuing model defined here, the value in the field is defined as 1 = device to host transfer (write to host memory), 0 = host to device transfer (read from host memory).

FIS Type — As defined in 10.5.9.

TAG — This field is used to identify the DMA buffer region in host memory to select for the data transfer. The low order 5 bits of the DMA Buffer Identifier Low field shall be set to the TAG field value corresponding to the command TAG that data is being transferred. The remaining bits of the DMA Buffer Identifier Low/High shall be cleared to zero. The 64 bit DMA Buffer Identifier field defined in the DMA Setup FIS according to 10.5.9 is used to convey a TAG value that occupies the five least-significant bits of the field.

DMA Buffer Offset
As defined in section 10.5.9. The device may specify/indicate a non-zero value in this field only if the host indicates support for it through the SET FEATURES mechanism as defined in section 13.3. Data is transferred to/from sequentially increasing logical addresses starting at the specified offset in the specified buffer.

DMA Transfer Count
As defined in 10.5.9. The value shall accurately reflect the length of the data transfer to follow. See 10.5.12.3 for special considerations if the transfer count is for an odd number of Words. Devices shall not clear this field to 0h. A value of 0h for this field is illegal and results in indeterminate behavior.

13.6.2.3 Status return mechanism

For maximum efficiency, the status return mechanism is not interlocked (does not include a handshake) while at the same time ensuring no status notifications are lost or overwritten (i.e., status notifications are race-free). The status return mechanism relies on an array of ACT bits – one ACT bit to convey the active status for each of the 32 possible outstanding commands, resulting in a 32 bit ACT status field. The 32 bit reserved field in the Set Device Bits FIS as defined in 10.5.7 is defined as the SActive field and is used to convey command completion information for updating the ACT bit array. The zero bit position in the 32 bit field corresponds to the ACT bit for the command with tag value of zero. Host software shall check the SActive register (containing the ACT bit array) if checking status in order to determine that command(s) have completed since the last time the host processed a command completion. It is possible for multiple commands to indicate completion by the time the host checks the status due to the software latencies in the host (i.e., by the time the host responds to one completion notification, another command may also have completed). Only successfully completed commands indicate their status using this mechanism – failed commands use an additional mechanism described in 13.6.4.3.1 and 13.6.5.3.2 to convey error information as well as the affected command tag. The Queued Error log is used to convey additional queued command error information as outlined in 13.7.4 and 13.7.

13.6.2.4 Priority

Host knowledge of I/O priority may be transmitted to the device as part of the command. There are two priority values for NCQ commands, normal and high. If the host marks an NCQ command as high priority, the host is requesting a better quality of service for that command than commands issued with normal priority.

The classes are forms of soft priority. The device may choose to complete a normal priority command before an outstanding high priority command, although preference should be given to the high priority commands.

EXAMPLE – One example where a normal priority command may be completed before a high priority command is if the normal priority command is a cache hit, whereas the high priority command requires access of the device media.

The priority class is specified in the Priority (PRIO) field for READ FPDMA QUEUED commands, WRITE FPDMA QUEUED commands, RECEIVE FPDMA QUEUED commands, SEND FPDMA QUEUED commands, and DURABLE/ORDERED WRITE NOTIFICATION subcommands. This bit may specify either the normal priority or high priority value. If a command is marked by the host as high priority, the device should attempt to provide better quality of service for the command. It is not required that devices process all high priority requests before satisfying normal priority requests.

For the READ FPDMA QUEUED command (see 13.6.4) and the WRITE FPDMA QUEUED command (see 13.6.5), processing of the PRIO field may be affected by the Command Duration Limits feature set (see ACS-5).

The device should complete high priority requests in a more timely fashion than normal and isochronous requests. The device should complete isochronous request prior to its associated deadline.

The device should complete isochronous request prior to its associated deadline (see Table 106).

Table 106 – Priority

PRIO field (1:0)	**Description**
00b	Normal Priority
01b	Isochronous – deadline dependent priority
10b	High priority
11b	Reserved

13.6.2.5 Unload

If using NCQ in a laptop environment, the host needs to be able to park the head of a device with rotating media due to excessive movement (e.g., the laptop being dropped). This section defines a mechanism that the host may use to park the heads if NCQ commands are outstanding in the device. The typical time for completion of the unload operation is defined in ATA/ATAPI-7 clause 6.20.10.

If NCQ commands are outstanding, the device is able to accept the IDLE IMMEDIATE command with the Unload Feature as defined in ACS-4.

Upon reception of this command with the Unload Feature specified, the device shall:

1) unload/park the heads, if any;
2) immediately; and
3) respond to the host with a Register Device to Host FIS with the ERR bit set to one in the Status register since this is a non-queued command.

If the host receives the error indication, it should proceed to read the Queued Error log (see 13.7.4). In the Queued Error log, the device shall indicate whether the error was due to receiving an UNLOAD and whether the UNLOAD was processed. The device shall not load the heads to the media if reading the Queued Error log.

The Queued Error log indicates whether the device has accepted the Unload and is in the process of processing the command. To get a definitive indication of Unload completion (and success), the IDLE IMMEDIATE command with the Unload Feature needs to be issued again after the Queued Error log has been read. After the Queued Error log has been read, there are no NCQ commands outstanding and the NCQ error is cleared to zero. A subsequent IDLE IMMEDIATE command with the Unload Feature shall be processed normally and a successful status shall be returned if the unload process completes successfully.

There may be a delay in issuing the IDLE IMMEDIATE command with the Unload feature to the device if the device is currently performing a data transfer for a previously issued NCQ command. If the device happens to be processing extensive data error recovery procedures, this delay may be longer than acceptable. However, this same issue may occur if a non-queued data command is outstanding and the device is performing error recovery procedures.

13.6.3 Intermixing Non-NCQ commands and NCQ commands

Non-NCQ commands are all commands other than NCQ commands.

The host shall not issue a non-NCQ command while an NCQ command is outstanding. Upon receiving a non-NCQ command while an NCQ command is outstanding, the device shall signal the error condition to the host by transmitting a Register Device to Host FIS with the ERR bit and ABRT bit set to one and the BSY bit cleared to zero in the Status field of the FIS and halt command processing as defined in 13.6.4.4 except as noted below.

Reception of a non-NCQ command to read the Queued Error log (see 13.7.4) after an error has occurred shall cause any outstanding NCQ commands to be aborted, and the device shall perform necessary state cleanup to return to a state with no commands pending. The device shall clear all bits in the SActive register by transmitting a Set Device Bits FIS to the host with all the bits in the SActive field set to one (i.e., FFFF FFFFh). After reading the Queued Error log, the device shall be prepared to process subsequently issued NCQ commands regardless of any previous errors on an NCQ command.

In the case that a non-NCQ command to read the Queued Error log is issued while an NCQ command is outstanding and no error was previously reported by the device, then the device shall signal an error condition. The receipt of this command if no error is outstanding shall be handled as any other non-NCQ command if a native queued command is outstanding. In this case, a subsequent non-NCQ command to read the Queued Error log is required to recover from the error.

13.6.4 READ FPDMA QUEUED command

13.6.4.1 READ FPDMA QUEUED command definition

Queued native read commands use this command. The command supports LBA mode only and uses 48 bit addressing only. The format of the command is defined in Figure 337.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	SECTOR COUNT(7:0)							
FEATURES(15:8)	SECTOR COUNT(15:8)							
COUNT(7:0)	TAG(4:0)					Reserved		RARC
COUNT(15:8)	PRIO(1:0)		Reserved					
LBA(7:0)	LBA(7:0)							
LBA(15:8)	LBA(15:8)							
LBA(23:16)	LBA(23:16)							
LBA(31:24)	LBA(31:24)							
LBA(39:32)	LBA(39:32)							
LBA(47:40)	LBA(47:40)							
ICC(7:0)	ICC(7:0)							
AUXILIARY(7:0)	Reserved					COMMAND DURATION LIMITS INDEX		
AUXILIARY(15:8)	Reserved							
AUXILIARY(23:16)	HYBRID INFORMATION(7:0)							
AUXILIARY(31:24)	Reserved							
DEVICE(7:0)	FUA	1	Res	0	Reserved			
COMMAND(7:0)	**60h**							

Figure 337 – READ FPDMA QUEUED command definition

Field Definitions

SECTOR COUNT

The SECTOR COUNT field is set to the number of logical sectors to be transferred. A value of 0000h indicates that 65 536 logical sectors are to be transferred.

TAG The TAG field value shall be assigned by host software to be different from all other TAG values corresponding to outstanding commands. The assigned TAG value shall not exceed the value specified in IDENTIFY DEVICE data Word 75.

RARC If the Rebuild Assist feature is not supported, then the RARC bit shall be ignored.

If the Rebuild Assist feature is supported and is disabled, then the RARC bit shall be ignored.

If the Rebuild Assist feature is supported and enabled, then the RARC bit specifies that read operations shall be processed as defined in 13.21.

PRIO If the Command Duration Limits feature set (see ACS-5) is not supported or not enabled, then the PRIO field value is assigned by the host based on the priority of the command issued. The device should complete high priority requests in a more timely fashion than normal and isochronous requests. The device should complete isochronous request prior to its associated deadline (see Table 106). If the Command Duration Limits feature set is supported and enabled, then the device shall ignore the PRIO field.

LBA The LBA field is set to the Logical Block Address of the first logical sector to be transferred.

ICC

The ICC field is valid if the PRIO field is set to a value of 01b. It is assigned by the host based on the intended deadline associated with the command issued. If a deadline has expired, the device shall continue to complete the command as soon as possible. This behavior may be modified by the host if the device supports the NCQ NON-DATA command (see 13.6.6) and supports the DEADLINE HANDLING subcommand (see 13.6.6.4). This subcommand allows the host to set whether the device shall abort (or continue processing) commands that have exceeded the time set in the ICC field.

There are several parameters encoded in the ICC field:

a) fine or coarse timing;
b) interval; and
c) the max time.

The Interval indicates the time units of the Time Limit parameter.

If the ICC field bit 7 is cleared to zero, then:

a) the time interval is fine-grained;
b) interval = 10 ms;
c) time limit = (ICC field (6:0) + 1) × 10 ms; and
d) max fine time = 128 × 10 ms = 1.28 s.

If the ICC field bit 7 is set to one (coarse encoding), then:

a) the time interval is coarse-grained;
b) interval = 0.5 s;
c) time limit = (ICC field (6:0) + 1) × 0.5 s; and
d) max coarse time = 128 × 0.5 s = 64 s.

COMMAND DURATION LIMITS INDEX

If the Command Duration Limits feature set is supported and enabled, then usage of the COMMAND DURATION LIMITS INDEX field is defined by ACS-5. If the Command Duration Limits feature set (see ACS-5) is not supported or is not enabled, then the device shall ignore the COMMAND DURATION LIMITS INDEX field.

HYBRID INFORMATION

If the Hybrid Information feature is supported (i.e., IDENTIFY DEVICE data Word 78 bit 9 is set to one) and is enabled (i.e., IDENTIFY DEVICE data Word 79 bit 9 is set to one), then the HYBRID INFORMATION field shall be processed as defined in 13.20. If the Hybrid Information feature is not supported (i.e., IDENTIFY DEVICE data Word 78 bit 9 is cleared to zero) or is disabled (i.e., IDENTIFY DEVICE data Word 79 bit 9 is cleared to zero), then the device shall ignore the HYBRID INFORMATION field.

FUA

If set to one forces the requested data to be retrieved from non-volatile storage medium regardless of whether the storage device holds the requested information in its volatile cache. If the device holds a modified copy of the requested data as a result of having cached writes, the modified data is first written to the medium before being retrieved from the storage medium as part of this operation. If cleared to zero, the requested data may be retrieved either from the device's non-volatile storage medium or from volatile cache.

Res

Reserved, shall be cleared to zero.

13.6.4.2 Success outputs

Upon successful completion of one or more outstanding commands, the device shall transmit a Set Device Bits FIS with the Interrupt bit set to one and one or more bits set to one in the ACT field corresponding to the bit position for each command TAG that has completed since the last status notification was transmitted.

The ACT field occupies the last 32 bits of the Set Device Bits FIS as defined in Figure 338.

0	Error(7:0) 00h								R	Status Hi							R	Status Lo 0							N	I 1	R	Reserved																	FIS Type (A1h)							
1	ACT(31:0)																																	

Figure 338 – Set Device Bits FIS for successful READ FPDMA QUEUED command completion

Field Definitions

Error The Error register shall be cleared to zero.

R Reserved, shall be cleared to zero.

Status As defined in section 10.5.7. The ERR bit shall be cleared to zero indicating successful command completion. Bit 4 may be set to one.

I Interrupt bit, the interrupt bit shall be set to one.

ACT The ACT field of the Set Device Bits FIS communicates successful completion notification for each of up to 32 queued commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating successful completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns.

All other fields as defined in section 10.5.7.

NOTE 56 - Devices are recommended to be aware that if choosing to aggregate status to the point where many of the outstanding commands have actually completed successfully without notification to the host, that an error may cause the final completion status of those commands to be failure.

A device should be selective if using status aggregation for outstanding queued commands to ensure the host is made aware of successful completion for outstanding commands so that an error does not force a high number of unnecessary command retries.

13.6.4.3 Error outputs

13.6.4.3.1 Upon receipt of a command

If the device has received a command that has not yet been acknowledged by clearing the BSY bit to zero and an error is encountered, the device shall transmit a Register Device to Host FIS (see Figure 339).

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	ERROR(7:0)							
COUNT(7:0)	na							
COUNT(15:8)	na							
LBA(7:0)	na							
LBA(15:8)	na							
LBA(23:16)	na							
LBA(31:24)	na							
LBA(39:32)	na							
LBA(47:40)	na							
DEVICE(7:0)	na							
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	na	na	ERR

Figure 339 – READ FPDMA QUEUED error on command receipt

Field Definitions

ERROR ATA error code for the failure condition of the failed command.

BSY 0

DRDY 1

DF 0

DRQ 0

ERR 1

Status bit 4 may be set to one.

Following transmission of the Register Device to Host FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands.

13.6.4.3.2 During processing of a command

If all commands have been acknowledged by clearing the BSY bit to zero and an error condition is detected, the device shall transmit a Set Device Bits FIS (see Figure 340) to the host. All outstanding commands at the time of an error shall be aborted as part of the error response and may be re-issued as appropriate by the host. For any commands that have not completed successfully or have resulted in error, the device shall clear the corresponding ACT bits to zero in the Set Device Bits FIS.

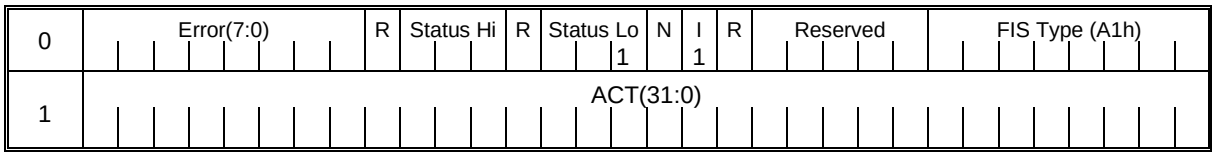


Figure 340 – Set Device Bits FIS with error notification and command completions

Field Definitions

Error The Error register shall contain the ATA error code.

R Reserved, shall be cleared to zero.

Status As defined in section 10.5.7. The ERR bit shall be set to one indicating an NCQ error has occurred. Status bit 4 may be set to one.

I Interrupt bit, the interrupt bit shall be set to one.

ACT The ACT field of the Set Device Bits FIS communicates successful completion notification for each of up to 32 queued commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating successful completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns.

All other fields as defined in section 10.5.7.

Only the registers that are updated as part of the Set Device Bits FIS are modified if the device signals an error condition and if the BSY bit in the Shadow Status register is cleared to zero, leaving the other Shadow Register Block Registers unchanged. If the device signals an error condition and if the BSY bit in the shadow Status register is set to one, then the device clears the BSY bit to zero with a Register Device to Host FIS that updates all registers in the Shadow Register Block, but the corresponding error information for the command is still retrieved by reading the Queued Error log.

Following transmission of the Set Device Bits FIS, the device shall stop processing any outstanding or new commands until the Queued Error log has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.4.4 Queue abort

Following transmission of the Register Device to Host FIS or Set Device Bits FIS in response to an NCQ error condition, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) is read using the GPL feature set.

If a command to read the Queued Error log is received, the device shall perform any necessary cleanup before returning detailed error information for the last failed command including the tag value for the failed command as described in 13.7.

In response to a received command to read the Queued Error log the device shall transmit a Set Device Bits FIS (see Figure 341) to the host with all the bits in the ACT field set to one. This policy avoids the host inadvertently completing a failed command with successful status. The exception to this policy is if the host reads the Queued Error log for information that is not directly tied to a specific error reported by the device. In the case where a device receives a command to read the Queued Error log that is not in direct response to an error reported by the device as well as no queued commands being outstanding, it is not required that a Set Device Bits FIS is delivered in response as it is not a necessity to abort any commands at that time.

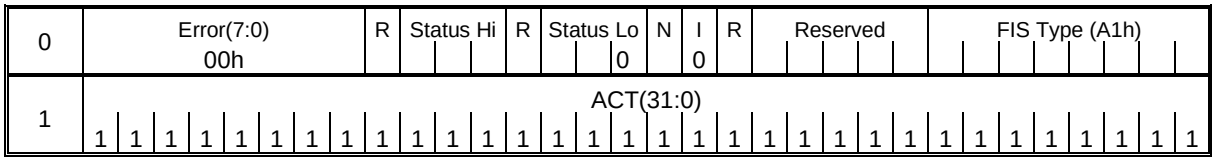


Figure 341 – Set Device Bits FIS aborting all outstanding commands

Field Definitions

Error The Error register shall be cleared to zero.

R Reserved, shall be cleared to zero.

Status As defined in 10.5.7. The ERR bit shall be cleared to zero indicating clean up of all previously outstanding commands. Bit 4 may be set to one.

I Interrupt bit, the interrupt bit shall be cleared to zero.

ACT The entire ACT field shall be set to one as an indication that all outstanding commands are being aborted.

All other fields as defined in 10.5.7.

If an error is indicated, the host shall treat any outstanding commands that do not have their corresponding SActive register bit cleared to zero as failed.

13.6.5 WRITE FPDMA QUEUED command

13.6.5.1 WRITE FPDMA QUEUED command definition

Queued native write commands make use of a command. The format of the command is defined in Figure 342.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	SECTOR COUNT(7:0)							
FEATURES(15:8)	SECTOR COUNT(15:8)							
COUNT(7:0)	TAG(4:0)					Reserved		
COUNT(15:8)	PRIO(1:0)		GROUP ID(5:0)					
LBA(7:0)	LBA(7:0)							
LBA(15:8)	LBA(15:8)							
LBA(23:16)	LBA(23:16)							
LBA(31:24)	LBA(31:24)							
LBA(39:32)	LBA(39:32)							
LBA(47:40)	LBA(47:40)							
ICC(7:0)	ICC(7:0)							
AUXILIARY(7:0)	Reserved					COMMAND DURATION LIMITS INDEX		
AUXILIARY(15:8)	Reserved							
AUXILIARY(23:16)	HYBRID INFORMATION(7:0)							
AUXILIARY(31:24)	Reserved							
DEVICE(7:0)	FUA	1	0	0	Reserved			
COMMAND(7:0)	**61h**							

Figure 342 – WRITE FPDMA QUEUED command definition

Field Definitions

SECTOR COUNT

The SECTOR COUNT field is set to the number of logical sectors to be transferred. A value of 0000h indicates that 65 536 logical sectors are to be transferred.

TAG The TAG field value shall be assigned by host software to be different from all other TAG values corresponding to outstanding commands. The assigned TAG value shall not exceed the value specified in IDENTIFY DEVICE data Word 75.

PRIO If the Command Duration Limits feature set (see ACS-5) is not supported or not enabled, then the PRIO field value is assigned by the host based on the priority of the command issued. The device should complete high priority requests in a more timely fashion than normal and isochronous requests. The device should complete isochronous request prior to its associated deadline (see Table 106). If the Command Duration Limits feature set is supported and enabled, then the device shall ignore the PRIO field.

GROUP ID If the SUPPORTS DURABLE/ORDERED WRITE NOTIFICATION bit is set to one, the data transferred by this WRITE FPDMA QUEUED command is associated with the Group ID, while the data is in the device's write cache.

LBA The LBA field is set to the Logical Block Address of the first logical sector to be transferred.

ICC The ICC field is valid if the PRIO field is set to a value of 01b. It is assigned by the host based on the intended deadline associated with the command issued. If a deadline has expired, the device shall continue to complete the command as soon as possible. This behavior may be modified by the host if the device

supports the NCQ NON-DATA command (see 13.6.6) and supports the DEADLINE HANDLING subcommand (see 13.6.6.4). This subcommand allows the host to set whether the device shall abort (or continue processing) commands that have exceeded the time set in the ICC field.

There are several parameters encoded in the ICC field:

a) fine or coarse timing;
b) interval; and
c) the Max Time.

The Interval indicates the time units of the Time Limit parameter.

If the ICC field bit 7 is cleared to zero, then

a) the time interval is fine-grained;
b) interval = 10 ms;
c) time Limit = (ICC field (6:0) + 1) × 10 ms; and
d) max Fine Time = 128 × 10 ms = 1.28 s.

If the ICC field bit 7 is set to one (coarse encoding), then:

a) the time interval is coarse-grained;
b) interval = 0.5 s;
c) time Limit = (ICC field (6:0) + 1) × 0.5 s; and
d) max Coarse Time = 128 × 0.5 s = 64 s.

COMMAND DURATION LIMITS INDEX

If the Command Duration Limits feature set is supported and enabled, then usage of the COMMAND DURATION LIMITS INDEX field is defined by ACS-5. If the Command Duration Limits feature set (see ACS-5) is not supported or is not enabled, then the device shall ignore the COMMAND DURATION LIMITS INDEX field.

HYBRID INFORMATION

If the Hybrid Information feature is supported (i.e., IDENTIFY DEVICE data Word 78 bit 9 is set to one) and is enabled (i.e., IDENTIFY DEVICE data Word 79 bit 9 is set to one), then the HYBRID INFORMATION field shall be processed as defined in 13.20). If the Hybrid Information feature is not supported (i.e., IDENTIFY DEVICE data Word 78 bit 9 is cleared to zero) or is disabled (i.e., IDENTIFY DEVICE data Word 79 bit 9 is cleared to zero), then the device shall ignore the HYBRID INFORMATION field.

FUA If set to one, forces the requested data to be written to non-volatile storage medium before completion status is indicated. If cleared to zero, the device may indicate completion status before the requested data is committed to the non-volatile storage medium.

13.6.5.2 Success outputs

Upon successful completion of one or more outstanding commands, the device shall transmit a Set Device Bits FIS with the Interrupt bit set to one and one or more bits set to one in the ACT field corresponding to the bit position for each command TAG that has completed since the last status notification was transmitted.

The ACT field occupies the last 32 bits of the Set Device Bits FIS as defined in Figure 343.

0	Error(7:0) 00h	R	Status Hi	R	Status Lo 0	N	I 1	R	Reserved	FIS Type (A1h)
1	ACT(31:0)									

Figure 343 – Set Device Bits FIS for successful WRITE FPDMA QUEUED command completion

Field Definitions

Error   The Error register shall be cleared to zero.

R   Reserved, shall be cleared to zero.

Status   As defined in 10.5.7. The ERR bit shall be cleared to zero indicating successful command completion. Bit 4 may be set to one.

I   Interrupt bit, the interrupt bit shall be set to one.

ACT   The ACT field of the Set Device Bits FIS communicates successful completion notification for each of up to 32 queued commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating successful completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns.

All other fields as defined in 10.5.7.

NOTE 57 - Devices are recommended to be aware that if choosing to aggregate status to the point where many of the outstanding commands have actually completed successfully without notification to the host, that an error may cause the final completion status of those commands to be failure.

A device should be selective if using status aggregation for outstanding queued commands to ensure the host is made aware of successful completion for outstanding commands so that an error does not force a high number of unnecessary command retries.

13.6.5.3 Error outputs

13.6.5.3.1 Upon receipt of a command

If the device has received a command that has not yet been acknowledged by clearing the BSY bit to zero and an error is encountered, the device shall transmit a Register Device to Host FIS (see Figure 344).

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	ERROR(7:0)							
COUNT(7:0)	na							
COUNT(15:8)	na							
LBA(7:0)	na							
LBA(15:8)	na							
LBA(23:16)	na							
LBA(31:24)	na							
LBA(39:32)	na							
LBA(47:40)	na							
DEVICE(7:0)	na							
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	na	na	ERR

Figure 344 – WRITE FPDMA QUEUED error on command receipt

Field Definitions

ERROR ATA error code for the failure condition of the failed command.

BSY 0

DRDY 1

DF 0

DRQ 0

ERR 1

Status bit 4 may be set to one.

Following transmission of the Register Device to Host FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.5.3.2 During processing of a command

If all commands have been acknowledged by clearing the BSY bit to zero and an error condition is detected, the device shall transmit a Set Device Bits FIS (see Figure 345) to the host. All outstanding commands at the time of an error shall be aborted as part of the error response and may be re-issued as appropriate by the host. For any commands that have not completed successfully or have resulted in error, the device shall clear the corresponding ACT bits to zero in the Set Device Bits FIS.

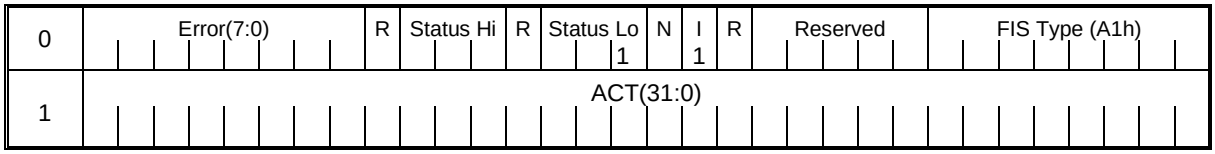


Figure 345 – Set Device Bits FIS with error notification and command completions

Field Definitions

Error The Error register shall contain the ATA error code.

R Reserved, shall be cleared to zero.

Status As defined in 10.5.7. The ERR bit shall be set to one indicating an NCQ error has occurred. Status bit 4 may be set to one.

I Interrupt bit, the interrupt bit shall be set to one.

ACT The ACT field of the Set Device Bits FIS communicates successful completion notification for each of up to 32 queued commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating successful completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns.

All other fields as defined in 10.5.7.

Only the registers that are updated as part of the Set Device Bits FIS are modified if the device signals an error condition and if the BSY bit in the shadow Status register is cleared to zero, leaving the other Shadow Register Block Registers unchanged. If the device signals an error condition and

if the BSY bit in the shadow Status register is set to one, then the device clears the BSY bit to zero with a Register Device to Host FIS that updates all registers in the Shadow Register Block.

Following transmission of the Set Device Bits FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.6 NCQ NON-DATA command

13.6.6.1 NCQ NON-DATA command definition

The NCQ NON-DATA command transports queued subcommands to the device that do not require any data transfer.

Some NCQ NON-DATA subcommands (see ACS-5) are processed as immediate NCQ commands (see 4.1.1.73). Some NCQ NON-DATA subcommands are processed as ordered NCQ commands (see 4.1.1.94).

The NCQ Non-Data log (see 13.7.6) indicates that subcommands are supported.

If NCQ is disabled and an NCQ NON-DATA command is issued to the device, then the device shall abort the command with the ERR bit set to one in the Status register and the ABRT bit set to one in the Error register. This command shall not be implemented for devices that implement the PACKET feature set. The queuing behavior of the device depends on the subcommand specified.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	Subcommand Specific				SUBCOMMAND(3:0)			
FEATURES(15:8)	Subcommand Specific							
COUNT(7:0)	TAG(4:0)					Reserved		
COUNT(15:8)	Subcommand Specific							
LBA(7:0)	Subcommand Specific							
LBA(15:8)	Subcommand Specific							
LBA(23:16)	Subcommand Specific							
LBA(31:24)	Subcommand Specific							
LBA(39:32)	Subcommand Specific							
LBA(47:40)	Subcommand Specific							
ICC(7:0)	Reserved							
AUXILIARY(7:0)	Subcommand Specific							
AUXILIARY(15:8)	Subcommand Specific							
AUXILIARY(23:16)	Subcommand Specific							
AUXILIARY(31:24)	Subcommand Specific							
DEVICE(7:0)	Res	1	Res	0	Reserved			
COMMAND(7:0)	**63h**							

Figure 346 – NCQ NON-DATA - command definition

Field Definitions

Subcommand Specific
: The Subcommand Specific fields are defined for each subcommand as referenced from Table 107.

SUBCOMMAND
: The SUBCOMMAND field (see Table 107) defines the subcommands that are valid.

  If an invalid subcommand is specified, then the device shall abort the command with the ERR bit set to one in the Status register, the ABRT bit set to one in the Error register, and shall cause all outstanding commands to be aborted.

TAG
: The TAG field value shall be assigned by host software to be different from all other TAG values corresponding to outstanding commands. The assigned TAG value shall not exceed the value specified in IDENTIFY DEVICE data Word 75.

Res
: Reserved, shall be cleared to zero.

NOTE 58 - In previous versions of this specification the LBA field (7:3) were assigned to Subcommand Specific (TTAG) field and the LBA field (2:0) were reserved. See Table 107 and the associated subcommand definitions for Subcommand Specific (TTAG) bit mapping.

The error and normal returns for this command are subcommand specific.

13.6.6.2 NCQ NON-DATA subcommands

Table 107 – Subcommands for NCQ NON-DATA

SUBCOMMAND field	**Description**	**Reference**
0h	ABORT NCQ QUEUE subcommand	13.6.6.3
1h	DEADLINE HANDLING subcommand	13.6.6.4
2h	HYBRID DEMOTE BY SIZE subcommand	13.6.6.7
3h	HYBRID CHANGE BY LBA RANGE subcommand	13.6.6.5
4h	HYBRID CONTROL subcommand	13.6.6.6
5h	SET FEATURES subcommand	13.6.6.8
6h	ZERO EXT subcommand	13.6.6.9
7h	ZAC MANAGEMENT OUT subcommand	13.6.6.10
8h	DURABLE/ORDERED WRITE NOTIFICATION subcommand	13.6.6.11
9h..Fh	Reserved	

13.6.6.3 ABORT NCQ QUEUE subcommand (0h)

13.6.6.3.1 ABORT NCQ QUEUE subcommand definition

A SUBCOMMAND field cleared to 0h specifies the ABORT NCQ QUEUE subcommand. The ABORT NCQ QUEUE subcommand is an immediate NCQ command. Support for this subcommand is indicated in the NCQ Non-Data log (see 13.7.6).

The ABORT NCQ QUEUE subcommand shall affect only those NCQ commands that the device has indicated command acceptance before accepting this NCQ NON-DATA command.

The format of the command is defined in Figure 347.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	ABORT TYPE(3:0)				0h			
FEATURES(15:8)	Reserved							
COUNT(7:0)	TAG(4:0)					Reserved		
COUNT(15:8)	Reserved							
LBA(7:0)	TTAG(4:0)					Reserved		
LBA(15:8)	Reserved							
LBA(23:16)	Reserved							
LBA(31:24)	Reserved							
LBA(39:32)	Reserved							
LBA(47:40)	Reserved							
AUXILIARY(7:0)	Reserved							
AUXILIARY(15:8)	Reserved							
AUXILIARY(23:16)	Reserved							
AUXILIARY(31:24)	Reserved							

Figure 347 – ABORT NCQ QUEUE – subcommand definition

Field Definitions

ABORT TYPE Describes the action requested. Table 108 shows the defined abort types. The NCQ Non-Data log (see 13.7.6) provides a list of abort types supported by the device (see Table 108).

TAG The TAG field value shall be assigned by host software to be different from all other TAG values corresponding to outstanding commands. TAG shall not exceed the value specified in IDENTIFY DEVICE data Word 75.

TTAG The TTAG field contains the value of the TAG of the outstanding command that is requested to be aborted. The TTAG value is only valid if the ABORT TYPE field is set to 3h (Abort Selected). TTAG shall not exceed the value specified in IDENTIFY DEVICE data Word 75.

Table 108 – Abort Type

Abort Type	Abort Type	Description
0h	Abort All	The device shall attempt to abort all outstanding NCQ commands.
1h	Abort Streaming	The device shall attempt to abort all outstanding NCQ Streaming commands. All non-streaming NCQ commands shall be unaffected.
2h	Abort Non-Streaming	The device shall attempt to abort all outstanding NCQ Non-Streaming commands. All NCQ Streaming commands shall be unaffected.
3h	Abort Selected	The device shall attempt to abort the outstanding NCQ command associated with the tag represented in TTAG field.
4h..Fh		Reserved

13.6.6.3.2 Success outputs

If a supported Abort Type is specified, then the device shall indicate success, even if the command results in no commands being aborted.

If an ABORT NCQ QUEUE command completes successfully (see Figure 348), a Set Device Bits FIS shall be sent to the host to complete the Abort subcommand and commands that were aborted as a consequence of the Abort subcommand by setting the ACT bits for those commands to one. This SDB FIS may also indicate other completed commands.

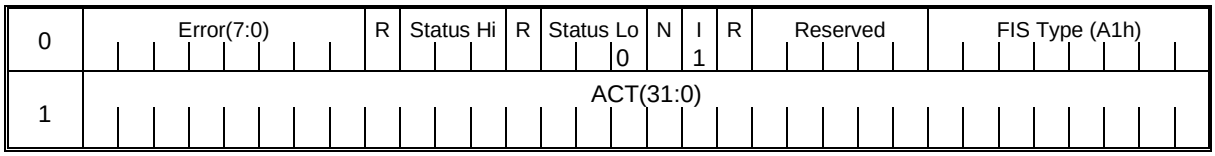


Figure 348 – ABORT NCQ QUEUE – successful completion

Field Definitions

Error The Error register shall contain 00h.

R Reserved, shall be cleared to zero.

Status As defined in 10.5.7. The ERR bit shall be cleared to zero indicating successful command completion. Bit 4 may be set to one.

I Interrupt bit, the interrupt bit shall be set to one.

ACT The ACT field of the Set Device Bits FIS communicates completion notification for each of up to 32 commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns. The device shall set the appropriate bit to one for each queued command that has been aborted, and shall set to one the bit associated with the TAG value for the ABORT NCQ QUEUE command.

All other fields as defined in 10.5.7.

13.6.6.3.3 Error outputs

13.6.6.3.3.1 Upon receipt of a command

If the value of the TTAG field equals the value of the TAG field, or if an unsupported Abort type parameter is specified, the device shall return command aborted.

If the device has received a command that has not yet been acknowledged by clearing the BSY bit to zero and an error is encountered, the device shall transmit a Register Device to Host FIS (see Figure 349).

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	ERROR(7:0)							
COUNT(7:0)	na							
COUNT(15:8)	na							
LBA(7:0)	na							
LBA(15:8)	na							
LBA(23:16)	na							
LBA(31:24)	na							
LBA(39:32)	na							
LBA(47:40)	na							
DEVICE(7:0)	na							
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	na	na	ERR

Figure 349 – ABORT NCQ QUEUE – error on command receipt

Field Definitions

ERROR ATA error code for the failure condition of the failed command.

BSY 0

DRDY 1

DF 0

DRQ 0

ERR 1

Status bit 4 may be set to one.

Following transmission of the Register Device to Host FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.6.3.3.2 During processing of a command

If all commands have been acknowledged by clearing the BSY bit to zero and an error condition is detected, the device shall transmit a Set Device Bits FIS (see Figure 350) to the host. All outstanding commands at the time of an error are aborted as part of the error response and may be re-issued as appropriate by the host. For any commands that have not completed successfully or have resulted in error, the device shall clear the corresponding ACT bits to zero in the Set Device Bits FIS.

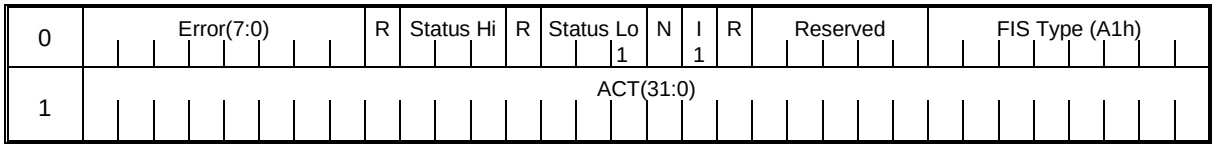


Figure 350 – ABORT NCQ QUEUE – error during processing

Field Definitions

Error The Error register shall contain the ATA error code.

R Reserved, shall be cleared to zero.

Status As defined in 10.5.7. The ERR bit shall be set to one indicating an NCQ error has occurred. Status bit 4 may be set to one.

I Interrupt bit, the interrupt bit shall be set to one.

ACT The ACT field of the Set Device Bits FIS communicates successful completion notification for each of up to 32 queued commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating successful completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns.

All other fields as defined in 10.5.7.

Only the registers that are updated as part of the Set Device Bits FIS are modified if the device signals an error condition and if the BSY bit in the shadow Status register is cleared to zero, leaving the other Shadow Register Block Registers unchanged. If the device signals an error condition and if the BSY bit in the shadow Status register is set to one, then the device clears the BSY bit to zero with a Register Device to Host FIS that updates all registers in the Shadow Register Block.

Following transmission of the Set Device Bits FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.6.4 DEADLINE HANDLING subcommand (1h)

13.6.6.4.1 DEADLINE HANDLING subcommand definition

A SUBCOMMAND field set to 1h specifies the DEADLINE HANDLING subcommand. This subcommand controls how NCQ Streaming commands are processed by the device. Support for this subcommand is indicated in the NCQ Non-Data log (see 13.7.6). The format of the command is defined in Figure 351.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	Reserved		RDNC	WDNC	1h			
FEATURES(15:8)	Reserved							
COUNT(7:0)	TAG(4:0)					Reserved		
COUNT(15:8)	Reserved							
LBA(7:0)	Reserved							
LBA(15:8)	Reserved							
LBA(23:16)	Reserved							
LBA(31:24)	Reserved							
LBA(39:32)	Reserved							
LBA(47:40)	Reserved							
AUXILIARY(7:0)	Reserved							
AUXILIARY(15:8)	Reserved							
AUXILIARY(23:16)	Reserved							
AUXILIARY(31:24)	Reserved							

Figure 351 – DEADLINE HANDLING – subcommand definition

Field Definitions

RDNC

If the Read Data Not Continue (RDNC) bit is cleared to zero, then the device may allow a READ FPDMA QUEUED command completion time to exceed the value specified by the ICC field in that READ FPDMA QUEUED command. If the RDNC bit is set to one, then all READ FPDMA QUEUED commands with the PRIO field set to 01b (see 13.6.2.4) shall:

a) be completed by the time specified by the ICC field timer value in that READ FPDMA QUEUED command; or

b) return command aborted for all outstanding commands.

The state of the RDNC bit shall be preserved across software resets and COMRESETs (via Software Setting Preservations), and shall not be preserved across power cycles.

The device shall ignore the state of the RDNC bit for READ FPDMA QUEUED commands where the PRIO field is not set to 01b.

WDNC

If the Write Data Not Continue (WDNC) bit is cleared to zero, then the device may allow a WRITE FPDMA QUEUED command completion time to exceed the value specified by the ICC field in that WRITE FPDMA QUEUED command. If the WDNC bit is set to one, then all WRITE FPDMA QUEUED commands with the PRIO field set to 01b shall:

a) be completed by the time specified by the ICC field timer value in that WRITE FPDMA QUEUED command; or

b) return command aborted for all outstanding commands.

The state of the WDNC bit shall be preserved across software resets and COMRESETs (via Software Setting Preservations), and shall not be preserved across power cycles.

The device shall ignore the state of the WDNC bit for WRITE FPDMA QUEUED commands where the PRIO field is not set to 01b.

TAG The TAG field value shall be assigned by host software to be different from all other TAG values corresponding to outstanding commands. TAG shall not exceed the value specified in IDENTIFY DEVICE data Word 75.

13.6.6.4.2 Success outputs

If this DEADLINE HANDLING subcommand is supported, the device shall return command completed with no error.

If a DEADLINE HANDLING subcommand completes successfully (see Figure 352), a Set Device Bits FIS shall be sent to the host to complete the DEADLINE HANDLING subcommand. This SDB FIS may also indicate other completed commands.

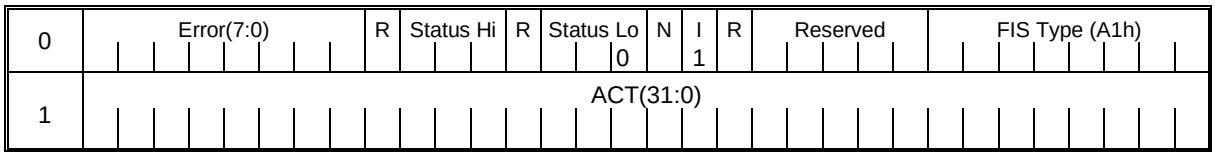


Figure 352 – DEADLINE HANDLING – successful completion

Field Definitions

Error The Error register shall contain 00h.

R Reserved, shall be cleared to zero.

Status As defined in 10.5.7. The ERR bit shall be cleared to zero indicating successful command completion. Bit 4 may be set to one.

I Interrupt bit, the interrupt bit shall be set to one.

ACT The ACT field of the Set Device Bits FIS communicates completion notification for each of up to 32 commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns. The device shall set to one the bit associated with the TAG value for the DEADLINE HANDLING command.

All other fields as defined in 10.5.7.

13.6.6.4.3 Error outputs

13.6.6.4.3.1 Upon receipt of a command

If the device has received a command that has not yet been acknowledged by clearing the BSY bit to zero and an error is encountered, the device shall transmit a Register Device to Host FIS (see Figure 353).

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	ERROR(7:0)							
COUNT(7:0)	na							
COUNT(15:8)	na							
LBA(7:0)	na							
LBA(15:8)	na							
LBA(23:16)	na							
LBA(31:24)	na							
LBA(39:32)	na							
LBA(47:40)	na							
DEVICE(7:0)	na							
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	na	na	ERR

Figure 353 – DEADLINE HANDLING – error on command receipt

Field Definitions

ERROR ATA error code for the failure condition of the failed command.

BSY 0

DRDY 1

DF 0

DRQ 0

ERR 1

Status bit 4 may be set to one.

Following transmission of the Register Device to Host FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.6.4.3.2 During processing of a command

If all commands have been acknowledged by clearing the BSY bit to zero and an error condition is detected, the device shall transmit a Set Device Bits FIS (see Figure 345). All outstanding commands at the time of an error are aborted as part of the error response and may be re-issued as appropriate by the host. For any commands that have not completed successfully or have resulted in error, the device shall clear the corresponding ACT bits to zero in the Set Device Bits FIS (see Figure 354).

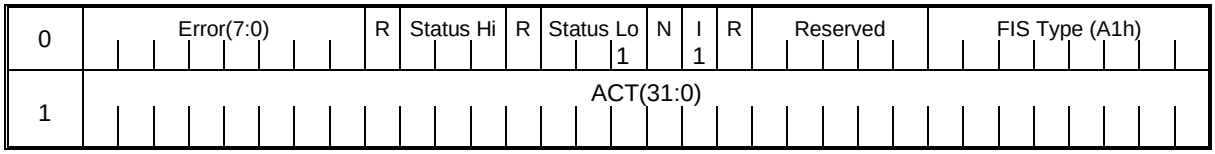


Figure 354 – DEADLINE HANDLING – error during processing

Field Definitions

Error The Error register shall contain the ATA error code.

R Reserved, shall be cleared to zero.

Status — As defined in 10.5.7. The ERR bit shall be set to one indicating an NCQ error has occurred. Status bit 4 may be set to one.

I — Interrupt bit, the interrupt bit shall be set to one.

ACT — The ACT field of the Set Device Bits FIS communicates successful completion notification for each of up to 32 queued commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating successful completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns.

All other fields as defined in 10.5.7.

Only the registers that are updated as part of the Set Device Bits FIS are modified if the device signals an error condition and if the BSY bit in the shadow Status register is cleared to zero, leaving the other Shadow Register Block Registers unchanged. If the device signals an error condition and if the BSY bit in the shadow Status register is set to one, then the device clears the BSY bit to zero with a Register Device to Host FIS that updates all registers in the Shadow Register Block.

Following transmission of the Set Device Bits FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.6.5 HYBRID CHANGE BY LBA RANGE subcommand (3h)

13.6.6.5.1 HYBRID CHANGE BY LBA RANGE subcommand definition

The HYBRID CHANGE BY LBA RANGE subcommand (see Figure 355) is used to change the hybrid information associated with a specified range of logical sectors. Support for this subcommand is indicated in the NCQ Non-Data log (see 13.7.6.11).

The device shall set the hybrid priority associated with a number of logical sectors starting at the LBA specified by the STARTING LBA field, regardless of what hybrid priority is associated with the selected logical sectors. The requested new hybrid priority may be any valid hybrid priority.

The Sector Count specifies the number of logical sectors that the device should change to the hybrid priority value specified in the HYBRID PRIORITY field in the HYBRID INFORMATION field.

The values of the MAX PRIORITY BEHAVIOR bit (see 13.7.8.2.11), the HYBRID PRIORITY field, and the CB bit (i.e., Cache Behavior) shall control the movement of all of the selected logical sectors that are not already in the non-volatile caching medium into the non-volatile caching medium as defined in Table 109.

Table 109 – Cache Behavior (CB) bit

Hybrid Priority	MAX PRIORITY BEHAVIOR bit	Cache Behavior (CB) bit	Description
Maximum Priority	1	na	See 13.20.2.2.2.2
	0	0	See 13.20.2.2.2.3
	0	1	See 13.20.2.2.2.4
Less than Maximum Priority but greater than zero	na	0	See 13.20.2.2.3.1
	na	1	Device shall not copy any of the selected logical sectors that are not already in the non-volatile caching medium into the non-volatile caching medium if such movement causes the device to spinup.
0	na	na	See 13.20.2.2.3.3

If:

a) the HYBRID PRIORITY field (see 13.20.2.2) is set to the Maximum Priority;
b) the MAX PRIORITY BEHAVIOR bit (see 13.7.8.2.11) is set to one; and
c) the non-volatile caching medium does not have mapping resources,

then:

a) the device shall return command aborted; and
b) in the Queued Error log, the device shall:
    A) set the SENSE KEY field to ABORTED COMMAND; and
    B) set the additional sense code (i.e., the ADDITIONAL SENSE CODE field and the ADDITIONAL SENSE CODE QUALIFIER field) to INSUFFICIENT RESOURCES.

If any of the selected logical sectors are already in the non-volatile caching medium associated with other HYBRID INFORMATION field values, then the new values shall replace the previous values.

The device shall complete the requested changes and move the data, if needed, before returning command completed.

NOTE 59: The device may take 30 s or more to complete the command if a large LBA range is specified. To minimize system response issues, large LBA ranges are recommended to be broken up into multiple smaller operations.

If the value of the HYBRID PRIORITY field is cleared to zero, then the device may evict the selected logical sectors from the non-volatile caching medium.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	Reserved			CB	3h			
FEATURES(15:8)	SECTOR COUNT(7:0)							
COUNT(7:0)	TAG(4:0)					Reserved		
COUNT(15:8)	SECTOR COUNT(15:8)							
LBA(7:0)	STARTING LBA(7:0)							
LBA(15:8)	STARTING LBA(15:8)							
LBA(23:16)	STARTING LBA(23:16)							
LBA(31:24)	STARTING LBA(31:24)							
LBA(39:32)	STARTING LBA(39:32)							
LBA(47:40)	STARTING LBA(47:40)							
AUXILIARY(7:0)	Reserved							
AUXILIARY(15:8)	Reserved							
AUXILIARY(23:16)	HYBRID INFORMATION(7:0)							
AUXILIARY(31:24)	Reserved							

Figure 355 – HYBRID CHANGE BY LBA RANGE – subcommand definition

Field Definitions

CB
: The Cache Behavior (CB) bit specifies the movement or non-movement of logical sectors into the non-volatile caching medium if the HYBRID INFORMATION field specifies a hybrid priority level other than the Max Priority. If the CB bit is set to one, then the device shall not copy any of the selected logical sectors that are not already in the non-volatile caching medium into the non-volatile caching medium if such movement would cause spinup of the device. If the CB bit is cleared to zero and the HYBRID PRIORITY field is non-zero, then the device should copy any of the selected logical sectors that are not already in the non-volatile caching medium into the non-volatile caching medium.

SECTOR COUNT
: The SECTOR COUNT field specifies the requested number of logical sectors, starting from Starting LBA.

TAG
: The TAG field value shall be assigned by host software to be different from all other TAG values corresponding to outstanding commands. TAG shall not exceed the value specified in IDENTIFY DEVICE data Word 75.

STARTING LBA
: The STARTING LBA field specifies the first LBA.

HYBRID INFORMATION
: The HYBRID INFORMATION field contains the data hints for the specified logical sectors (see 13.20.2).

HYBRID PRIORITY
: The HYBRID PRIORITY field is embedded in the HYBRID INFORMATION field.

13.6.6.5.2 Success outputs

If a HYBRID CHANGE BY LBA RANGE subcommand completes without error (see Figure 356), then a Set Device Bits FIS shall be sent to the host. This SDB FIS may also indicate other completed commands.

0	Error(7:0)	R	Status Hi	R	Status Lo 0	N	I 1	R	Reserved	FIS Type (A1h)
1	ACT(31:0)									

Figure 356 – HYBRID CHANGE BY LBA RANGE – successful completion

Field Definitions

- Error — The Error register shall contain 00h.
- R — Reserved, shall be cleared to zero.
- Status — As defined in 10.5.7. The ERR bit shall be cleared to zero indicating successful command completion. Bit 4 may be set to one.
- I — Interrupt bit, the interrupt bit shall be set to one.
- ACT — The ACT field of the Set Device Bits FIS communicates completion notification for each of up to 32 commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns. The device shall set to one the bit associated with the TAG value for the HYBRID CHANGE BY LBA RANGE subcommand.

All other fields as defined in 10.5.7.

13.6.6.5.3 Error outputs

13.6.6.5.3.1 Upon receipt of a command

If the device has received a command that has not yet been acknowledged by clearing the BSY bit to zero and an error is encountered, the device shall transmit a Register Device to Host FIS (see Figure 357).

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	ERROR(7:0)							
COUNT(7:0)	na							
COUNT(15:8)	na							
LBA(7:0)	na							
LBA(15:8)	na							
LBA(23:16)	na							
LBA(31:24)	na							
LBA(39:32)	na							
LBA(47:40)	na							
DEVICE (7:0)	na							
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	na	na	ERR

Figure 357 – HYBRID CHANGE BY LBA RANGE – error on command receipt

Field Definitions

- ERROR — ATA error code for the failure condition of the failed command.
- BSY — 0
- DRDY — 1
- DF — 0

DRQ	0
ERR	1

Status bit 4 may be set to one.

Following transmission of the Register Device to Host FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.6.5.3.2 During processing of a command

If all commands have been acknowledged by clearing the BSY bit to zero and an error condition is detected, the device shall transmit a Set Device Bits FIS (see Figure 358) to the host. All outstanding commands at the time of an error are aborted as part of the error response and may be re-issued as appropriate by the host. For any commands that have not completed or have completed with error, the device shall clear the corresponding ACT bits to zero in the Set Device Bits FIS.

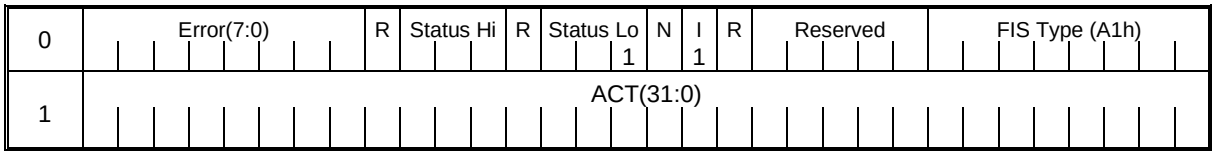


Figure 358 – HYBRID CHANGE BY LBA RANGE – error during processing

Field Definitions

Error	The Error register shall contain the ATA error code.
R	Reserved, shall be cleared to zero.
Status	As defined in 10.5.7. The ERR bit shall be set to one indicating an NCQ error has occurred. Status bit 4 may be set to one.
I	Interrupt bit, the interrupt bit shall be set to one.
ACT	The ACT field of the Set Device Bits FIS communicates successful completion notification for each of up to 32 queued commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating successful completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns.

All other fields as defined in 10.5.7.

Only the registers that are updated as part of the Set Device Bits FIS are modified if the device signals an error condition and if the BSY bit in the shadow Status register is cleared to zero, leaving the other Shadow Register Block Registers unchanged. If the device signals an error condition when the BSY bit in the shadow Status register is set to one, then the device clears the BSY bit to zero with a Register Device to Host FIS that updates all registers in the Shadow Register Block.

Following transmission of the Set Device Bits FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.6.6 HYBRID CONTROL subcommand (4h)

13.6.6.6.1 HYBRID CONTROL subcommand definition

The HYBRID CONTROL subcommand (see Figure 359) provides parameters for the use of the non-volatile caching medium. Support for this subcommand is indicated in the NCQ Non-Data log (see 13.7.6.12).

In contrast, the SET FEATURES Enable/Disable Hybrid Information subcommand (see 13.3.11) provides a mechanism to enable or disable the Hybrid Information feature.

If the command completes without error, then the device shall preserve the values of the Dirty Low Threhold field and the DIRTY HIGH THRESHOLD field across all resets and power cycle events. Current values are available in the Hybrid Information log (see 13.7.8).

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	DISABLE CACHING MEDIUM	Reserved			4h			
FEATURES(15:8)	Reserved							
COUNT(7:0)	TAG(4:0)					Reserved		
COUNT(15:8)	Reserved							
LBA(7:0)	DIRTY LOW THRESHOLD(7:0)							
LBA(15:8)	DIRTY HIGH THRESHOLD(7:0)							
LBA(23:16)	Reserved							
LBA(31:24)	Reserved							
LBA(39:32)	Reserved							
LBA(47:40)	Reserved							
AUXILIARY(7:0)	Reserved							
AUXILIARY(15:8)	Reserved							
AUXILIARY(23:16)	Reserved							
AUXILIARY(31:24)	Reserved							

Figure 359 – HYBRID CONTROL – subcommand definition

Field Definitions

DISABLE CACHING MEDIA

If the DISABLE CACHING MEDIA bit is cleared to zero, then the device shall process the Dirty Low Threhold field and the DIRTY HIGH THRESHOLD field.

If the DISABLE CACHING MEDIA bit is set to one and the Hybrid Information feature is enabled, then the device shall:

1) ignore the contents of the Dirty Low Threhold field and the DIRTY HIGH THRESHOLD field;
2) change the value of the ENABLED field (see 13.7.8.2.3) of the Hybrid Information log to 80h (i.e., Hybrid Information Disable In Process);
3) sync all dirty data in the non-volatile caching medium to the primary medium;
4) evict all data from the non-volatile caching medium;
5) disable the Hybrid Information feature (see 13.3.11);
6) change the value of the ENABLED field (see 13.7.8.2.3) of the Hybrid Information log to 00h (i.e., Hybrid Information Disabled); and
7) disable the use of the non-volatile caching medium for storing user data until the device processes a SET FEATURES Enable Hybrid Information subcommand.

If the device processes a reset or a power cycle while the value of the ENABLED field (see 13.7.8.2.3) of the Hybrid Information log is 80h (i.e., Hybrid Information Disable In Process), then the device shall change the value of the ENABLED field to FFh (i.e., Hybrid Information Enabled).

If the Disable Caching Medium bit is set to one, then the device may report command completion before making the requested changes. The host should check the Hybrid Information log to determine if the requested changes have been completed.

TAG

The TAG field value shall be assigned by host software to be different from all other TAG values corresponding to outstanding commands. TAG shall not exceed the value specified in IDENTIFY DEVICE data Word 75.

DIRTY LOW THRESHOLD

The DIRTY LOW THRESHOLD field represents the threshold for the amount of dirty user logical sectors in the non-volatile caching medium that syncing operations should stop. The value of the DIRTY LOW THRESHOLD field divided by 255 specifies a fraction of the total reported NVM Size of the non-volatile caching medium that contains dirty logical sectors. The device shall preserve this setting over all power cycles and all resets. See 13.20.3 for additional information on syncing. The current value is reported in the Hybrid Information log (see 13.7.8.2.5).

DIRTY HIGH THRESHOLD

The DIRTY HIGH THRESHOLD field represents the threshold for the amount of dirty user logical sectors in the non-volatile caching medium that syncing operations should begin. The value of the DIRTY HIGH THRESHOLD field divided by 255 specifies a fraction of the total reported NVM Size of the non-volatile caching medium that contains dirty data. The device shall preserve this setting over all power cycles and all resets. The current value is reported in the Hybrid Information log (see 13.7.8.2.6). See 13.20.3 for additional information on syncing.

13.6.6.6.2 Success outputs

If a HYBRID CONTROL subcommand completes without error (see Figure 360), a Set Device Bits FIS shall be sent to the host. This SDB FIS may also indicate other completed commands.

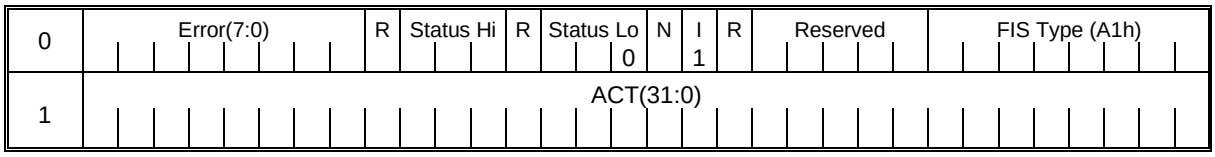


Figure 360 – HYBRID CONTROL – successful completion

Field Definitions

Error — The Error register shall contain 00h.

R — Reserved, shall be cleared to zero.

Status — As defined in 10.5.7. The ERR bit shall be cleared to zero indicating successful command completion. Bit 4 may be set to one.

I — Interrupt bit, the interrupt bit shall be set to one.

ACT — The ACT field of the Set Device Bits FIS communicates completion notification for each of up to 32 commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating completion notification for. The device may set more than one bit to one if it is explicitly aggregating

successful status returns. The device shall set to one the bit associated with the TAG value for the HYBRID CONTROL subcommand.

All other fields as defined in 10.5.7.

13.6.6.6.3 Error outputs

13.6.6.6.3.1 Upon receipt of a command

If the device has received a command that has not yet been acknowledged by clearing the BSY bit to zero and an error is encountered, the device shall transmit a Register Device to Host FIS (see Figure 361).

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	ERROR(7:0)							
COUNT(7:0)	na							
COUNT(15:8)	na							
LBA(7:0)	na							
LBA(15:8)	na							
LBA(23:16)	na							
LBA(31:24)	na							
LBA(39:32)	na							
LBA(47:40)	na							
DEVICE(7:0)	na							
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	na	na	ERR

Figure 361 – HYBRID CONTROL – error on command receipt

Field Definitions

ERROR ATA error code for the failure condition of the failed command.

BSY 0

DRDY 1

DF 0

DRQ 0

ERR 1

Status bit 4 may be set to one.

Following transmission of the Register Device to Host FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.6.6.3.2 During processing of a command

If all commands have been acknowledged by clearing the BSY bit to zero and an error condition is detected, the device shall transmit a Set Device Bits FIS (see Figure 362) to the host. All outstanding commands at the time of an error are aborted as part of the error response and may be re-issued as appropriate by the host. For any commands that have not completed or have completed with error, the device shall clear the corresponding ACT bits to zero in the Set Device Bits FIS.

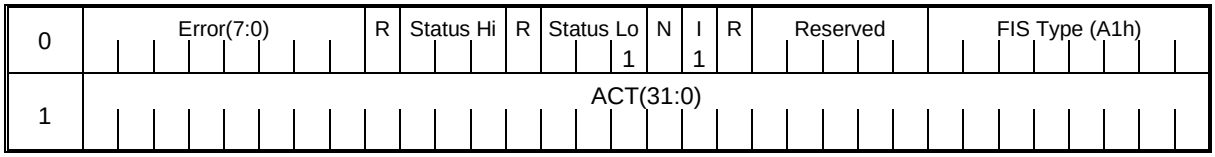


Figure 362 – HYBRID CONTROL – error during processing

Field Definitions

Error The Error register shall contain the ATA error code.

R Reserved, shall be cleared to zero.

Status As defined in 10.5.7. The ERR bit shall be set to one indicating an NCQ error has occurred. Status bit 4 may be set to one.

I Interrupt bit, the interrupt bit shall be set to one.

ACT The ACT field of the Set Device Bits FIS communicates successful completion notification for each of up to 32 queued commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating successful completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns.

All other fields as defined in 10.5.7.

Only the registers that are updated as part of the Set Device Bits FIS are modified if the device signals an error condition and if the BSY bit in the shadow Status register is cleared to zero, leaving the other Shadow Register Block Registers unchanged. If the device signals an error condition when the BSY bit in the shadow Status register is set to one, then the device clears the BSY bit to zero with a Register Device to Host FIS that updates all registers in the Shadow Register Block.

Following transmission of the Set Device Bits FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.6.7 HYBRID DEMOTE BY SIZE subcommand (2h)

13.6.6.7.1 HYBRID DEMOTE BY SIZE subcommand definition

The HYBRID DEMOTE BY SIZE subcommand (see Figure 363) is used to change the hybrid priority associated with logical sectors. Support for this subcommand is indicated in the NCQ Non-Data log (see 13.7.6.10).

The device changes the Hybrid Priority of logical sectors in the non-volatile caching medium from the value specified in the PRIORITY field (see 13.20.2.2) in the HYBRID INFORMATION field to the Hybrid Priority value specified in the HYBRID PRIORITY field (see 13.20.2.2).

Table 110 specifies the number of logical sectors that should be demoted by the HYBRID DEMOTE BY SIZE. The number of logical sectors demoted may be less than what is specified by the host.

Table 110 – HYBRID DEMOTE BY SIZE – number of logical sectors affected

Sector Count [a] is less than Actual [b]	Number of Logical Sectors to change to the To Priority [c]
Y	Sector Count [a]
N	Actual [b] number

[a] The value of the SECTOR COUNT field in the command parameters.
[b] The actual number of logical sectors in the non-volatile caching medium associated with the FROMPRIORITY field.
[c] The value of the HYBRID PRIORITY field (see 13.7.6.10) within the HYBRID INFORMATION field (see 13.20.2).

The device selects the logical sectors that are to be changed based on vendor specific criteria (e.g., age and usage).

The device shall return command aborted if:

a) the value of the FROMPRIORITY field is less than or equal to the value of the HYBRID PRIORITY field; or
b) the value of the FROMPRIORITY field is equal to the MAXIMUM HYBRID PRIORITY LEVEL field (see 13.7.8.2.8) in the Hybrid Information log and the MAX PRIORITY BEHAVIOR bit (see 13.7.8.2.11) is set to one.

The device shall complete the operation before returning command complete.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	FROMPRIORITY(3:0)				2h			
FEATURES(15:8)	SECTOR COUNT(7:0)							
COUNT(7:0)	TAG(4:0)					Reserved		
COUNT(15:8)	SECTOR COUNT(15:8)							
LBA(7:0)	SECTOR COUNT(23:16)							
LBA(15:8)	SECTOR COUNT(31:24)							
LBA(23:16)	Reserved							
LBA(31:24)	Reserved							
LBA(39:32)	Reserved							
LBA(47:40)	Reserved							
AUXILIARY(7:0)	Reserved							
AUXILIARY(15:8)	Reserved							
AUXILIARY(23:16)	HYBRID INFORMATION(7:0)							
AUXILIARY(31:24)	Reserved							

Figure 363 – HYBRID DEMOTE BY SIZE – command definition

Field Definitions

FROMPRIORITY

The FROMPRIORITY field specifies the Hybird Priority level corresponding to the data that is to be changed to the hybirid priority specified in the HYBRID INFORMATION field.

SECTOR COUNT

The SECTOR COUNT field specifies the requested number of logical sectors that the change of hybrid priority applies.

TAG

The TAG field value shall be assigned by host software to be different from all other TAG values corresponding to outstanding commands. TAG shall not exceed the value specified in IDENTIFY DEVICE data Word 75.

HYBRID INFORMATION

The HYBRID INFORMATION field contains the data hints for the specified logical sectors (see 13.20.2).

13.6.6.7.2 Success outputs

If a HYBRID DEMOTE BY SIZE subcommand completes without error (see Figure 364), then a Set Device Bits FIS shall be sent to the host. This SDB FIS may also indicate other completed commands.

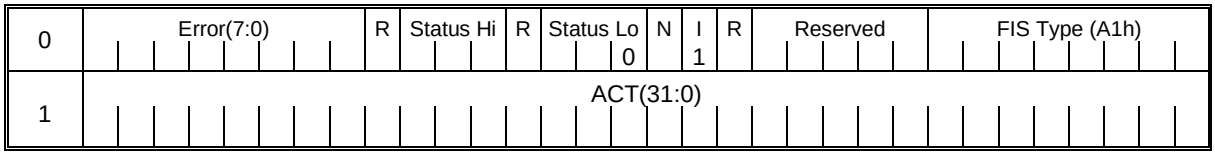


Figure 364 – HYBRID DEMOTE BY SIZE – successful completion

Field Definitions

Error — The Error register shall contain 00h.

R — Reserved, shall be cleared to zero.

Status — As defined in 10.5.7. The ERR bit shall be cleared to zero indicating successful command completion. Bit 4 may be set to one.

I — Interrupt bit, the interrupt bit shall be set to one.

ACT — The ACT field of the Set Device Bits FIS communicates completion notification for each of up to 32 commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns. The device shall set to one the bit associated with the TAG value for the HYBRID DEMOTE BY SIZE subcommand.

All other fields as defined in 10.5.7.

13.6.6.7.3 Error outputs

13.6.6.7.3.1 Upon receipt of a command

If the device has received a command that has not yet been acknowledged by clearing the BSY bit to zero and an error is encountered, the device shall transmit a Register Device to Host FIS (see Figure 365).

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	ERROR(7:0)							
COUNT(7:0)	na							
COUNT(15:8)	na							
LBA(7:0)	na							
LBA(15:8)	na							
LBA(23:16)	na							
LBA(31:24)	na							
LBA(39:32)	na							
LBA(47:40)	na							
DEVICE(7:0)	na							
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	na	na	ERR

Figure 365 – HYBRID DEMOTE BY SIZE – error on command receipt

Field Definitions

ERROR ATA error code for the failure condition of the failed command.

BSY 0

DRDY 1

DF 0

DRQ 0

ERR 1

Status bit 4 may be set to one.

Following transmission of the Register Device to Host FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.6.7.3.2 During processing of a command

If all commands have been acknowledged by clearing the BSY bit to zero and an error condition is detected, the device shall transmit a Set Device Bits FIS (see Figure 366) to the host. All outstanding commands at the time of an error are aborted as part of the error response and may be re-issued as appropriate by the host. For any commands that have not completed or have completed with error, the device shall clear the corresponding ACT bits to zero in the Set Device Bits FIS.

0	Error(7:0)	R	Status Hi	R	Status Lo 1	N	I 1	R	Reserved	FIS Type (A1h)
1	ACT(31:0)									

Figure 366 – HYBRID DEMOTE BY SIZE – error during processing

Field Definitions

Error The Error register shall contain the ATA error code.

R Reserved, shall be cleared to zero.

Status As defined in 10.5.7. The ERR bit shall be set to one indicating an NCQ error has occurred. Status bit 4 may be set to one.

I Interrupt bit, the interrupt bit shall be set to one.

ACT The ACT field of the Set Device Bits FIS communicates successful completion notification for each of up to 32 queued commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating successful completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns.

All other fields as defined in 10.5.7.

Only the registers that are updated as part of the Set Device Bits FIS are modified if the device signals an error condition and if the BSY bit in the shadow Status register is cleared to zero, leaving the other Shadow Register Block Registers unchanged. If the device signals an error condition when the BSY bit in the shadow Status register is set to one, then the device clears the BSY bit to zero with a Register Device to Host FIS that updates all registers in the Shadow Register Block.

Following transmission of the Set Device Bits FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.6.8 SET FEATURES subcommand (5h)

13.6.6.8.1 SET FEATURES subcommand overview

The SET FEATURES subcommand (see Figure 367) functionality and behavior is dependent on all requirements of the SET FEATURES command and the IDENTIFY DEVICE command defined in ACS-4.

NOTE 60 – Some SET FEATURES subcommands may return command aborted if they are issued at a time when the device is unable to process them (e.g., changing the non-zero buffer offsets setting while commands are in the queue).

13.6.6.8.2 Inputs

The format of the command is defined in Figure 367.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	Reserved				5h			
FEATURES(15:8)	Contents of the SET FEATURES command FEATURES(7:0)							
COUNT(7:0)	TAG(4:0)					Reserved		
COUNT(15:8)	Contents of the SET FEATURES command COUNT(7:0)							
LBA(7:0)	Contents of the SET FEATURES command LBA(7:0)							
LBA(15:8)	Contents of the SET FEATURES command LBA(15:8)							
LBA(23:16)	Contents of the SET FEATURES command LBA(23:16)							
LBA(31:24)	Reserved				Contents of the SET FEATURES command LBA(27:24)			
LBA(39:32)	Reserved							
LBA(47:40)	Reserved							
AUXILIARY(7:0)	Reserved							
AUXILIARY(15:8)	Reserved							
AUXILIARY(23:16)	Reserved							
AUXILIARY(31:24)	Reserved							

Figure 367 – SET FEATURES subcommand = 05h

See ACS-4 for the definition of the SET FEATURES command.

13.6.6.8.3 Success outputs

See 13.6.8.3.

13.6.6.8.4 Error outputs

See 13.6.8.4.

13.6.6.9 ZERO EXT subcommand (6h)

13.6.6.9.1 ZERO EXT subcommand overview

The ZERO EXT subcommand functionality and behavior is defined in ACS-4.

13.6.6.9.2 Inputs

The format of the command is defined in Figure 368.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	Reserved				6h			
FEATURES(15:8)	Contents of the ZERO EXT command COUNT(7:0)							
COUNT(7:0)	TAG(4:0)					Reserved		
COUNT(15:8)	Contents of the ZERO EXT command COUNT(15:8)							
LBA(7:0)	Contents of the ZERO EXT command LBA(7:0)							
LBA(15:8)	Contents of the ZERO EXT command LBA(15:8)							
LBA(23:16)	Contents of the ZERO EXT command LBA(23:16)							
LBA(31:24)	Contents of the ZERO EXT command LBA(31:24)							
LBA(39:32)	Contents of the ZERO EXT command LBA(39:32)							
LBA(47:40)	Contents of the ZERO EXT command LBA(47:40)							
AUXILIARY(7:0)	Contents of the ZERO EXT command FEATURE(7:0)							
AUXILIARY(15:8)	Contents of the ZERO EXT command FEATURE(15:8)							
AUXILIARY(23:16)	Reserved							
AUXILIARY(31:24)	Reserved							

Figure 368 – ZERO EXT subcommand = 6h

See ACS-4 for the definition of the ZERO EXT command.

13.6.6.9.3 Success outputs

See 13.6.8.3.

13.6.6.9.4 Error outputs

See 13.6.8.4.

13.6.6.10 ZAC MANAGEMENT OUT subcommand (7h)

13.6.6.10.1 ZAC MANAGEMENT OUT subcommand overview

The ZAC MANAGEMENT OUT subcommand functionality and behavior is defined in ACS-4.

Some ZAC MANAGEMENT OUT subcommands (see ZAC-2) are processed as ordered NCQ commands (see 4.1.1.94).

13.6.6.10.2 Inputs

The format of the command is defined in Figure 369.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	Reserved				7h			
FEATURES(15:8)	Contents of the ZAC MANAGEMENT OUT command COUNT(7:0)							
LBA(7:0)	Contents of the ZAC MANAGEMENT OUT command LBA(7:0)							
LBA(15:8)	Contents of the ZAC MANAGEMENT OUT command LBA(15:8)							
LBA(23:16)	Contents of the ZAC MANAGEMENT OUT command LBA(23:16)							
LBA(31:24)	Contents of the ZAC MANAGEMENT OUT command LBA(31:24)							
LBA(39:32)	Contents of the ZAC MANAGEMENT OUT command LBA(39:32)							
LBA(47:40)	Contents of the ZAC MANAGEMENT OUT command LBA(47:40)							
AUXILIARY(7:0)	Contents of the ZAC MANAGEMENT OUT command FEATURE(7:0)							
AUXILIARY(15:8)	Contents of the ZAC MANAGEMENT OUT command FEATURE(15:8)							
AUXILIARY(23:16)	Reserved							
AUXILIARY(31:24)	Reserved							

Figure 369 – ZAC MANAGEMENT OUT subcommand = 7h

See ZAC for the definition of the ZAC MANAGEMENT OUT command.

13.6.6.10.3 Success outputs

See 13.6.8.3.

13.6.6.10.4 Error outputs

See 13.6.8.4.

13.6.6.11 DURABLE/ORDERED WRITE NOTIFICATION subcommand (8h)

13.6.6.11.1 DURABLE/ORDERED WRITE NOTIFICATION subcommand overview

The DURABLE/ORDERED WRITE NOTIFICATION subcommand provides all Group IDs to be used when processing the DURABLE/ORDERED WRITE NOTIFICATION subcommand. Support for this subcommand is indicated in the NCQ NON-DATA log (see 13.7.6).

If a DURABLE/ORDERED WRITE NOTIFICATION subcommand is marked by the host as high priority, the device should attempt to provide better quality of service for the command than for other NCQ commands that are not marked by the host as high priority.

WRITE FPDMA QUEUED commands outstanding at the same time as a DURABLE/ORDERED WRITE NOTIFICATION subcommand may have data associated with any of the Group IDs provided by this command and may add data associated with those Group IDs to the volatile cache and may delay the completion of this command. If the device indicates command acceptance of multiple DURABLE/ORDERED WRITE NOTIFICATION subcommands with the same values in the GROUP ID MASK field, then the device shall complete all DURABLE/ORDERED WRITE NOTIFICATION subcommands with the same values in the GROUP ID MASK field concurrently.

NOTE 61 - This command may take more than 30 seconds to complete..

13.6.6.11.2 Inputs

The format of the command is defined in Figure 370.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	D/OW	PRIO		R	8h			
FEATURES(15:8)	GROUP ID MASK (55:48)							
COUNT(7:0)	TAG (4:0)					Reserved		
COUNT(15:8)	GROUP ID MASK (63:56)							
LBA(7:0)	GROUP ID MASK (7:0)							
LBA(15:8)	GROUP ID MASK (15:8)							
LBA(23:16)	GROUP ID MASK (23:16)							
LBA(31:24)	GROUP ID MASK (31:24)							
LBA(39:32)	GROUP ID MASK (39:32)							
LBA(47:40)	GROUP ID MASK (47:40)							
ICC(7:0)	Reserved							
AUXILIARY(7:0)	Reserved							
AUXILIARY(15:8)	Reserved							
AUXILIARY(23:16)	Reserved							
AUXILIARY(31:24)	Reserved							

Figure 370 – DURABLE/ORDERED WRITE NOTIFICATION subcommand=8h

Field Definitions

D/OW

If the SUPPORTS D/OW bit (see 13.7.6.17) is cleared to zero or the D/OW bit is cleared to zero, the device shall not indicate successful completion until all data received and stored in the device's cache associated with any of the Group IDs provided by this command has been flushed to the non-volatile media. If the volatile cache is disabled or no volatile cache is present, the device shall indicate command completion without error.

If the SUPPORTS D/OW bit is set one and the D/OW bit is set to one, successful completion indicates that all data received and stored in the device's write cache associated with any of the Group IDs provided by this command shall be flushed to the non-volatile media before all data not yet received and stored in the device's write cache associated with any of the Group IDs provided by this command. If the

volatile write cache is disabled or no volatile write cache is present, the device shall indicate command completion without error.

PRIO The PRIO field value is assigned by the host based on the priority of the command issued. The device should complete high priority requests in a more timely fashion than normal priority and isochronous requests. If the PRIO field value is isochronous, then the device shall complete this command with command aborted (see Table 106).

R Reserved.

GROUP ID MASK
The GROUP ID MASK field of the DURABLE/ORDERED WRITE NOTIFICATION subcommand communicates the data targeted in the device's write cache. Data associated with a GROUP ID was specified during the WRITE FPDMA QUEUED command. The GROUP ID MASK field is bit-significant with each bit corresponding to a GROUP ID , where bit 0 corresponds to GROUP ID 0 and bit 63 corresponds to GROUP ID 63. More than one bit may be set to one.

All other fields as defined in 13.6.6.1.

13.6.6.11.3 Success outputs

If a DURABLE/ORDERED WRITE NOTIFICATION subcommand completes without error (see Figure 364), then a Set Device Bits FIS shall be sent to the host. This SDB FIS may also indicate other completed commands.

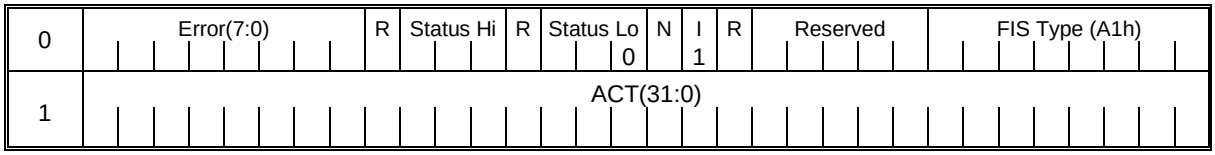


Figure 371 – DURABLE/ORDERED WRITE NOTIFICATION – successful completion

Field Definitions

Error The Error register shall contain 00h.

R Reserved, shall be cleared to zero.

Status As defined in 10.5.7. The ERR bit shall be cleared to zero indicating successful command completion. Bit 4 may be set to one.

I Interrupt bit, the interrupt bit shall be set to one.

ACT The ACT field of the Set Device Bits FIS communicates completion notification for each of up to 32 commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns. The device shall set to one the bit associated with the TAG value for the HYBRID DEMOTE BY SIZE subcommand.

All other fields as defined in 10.5.7.

13.6.6.11.4 Error outputs

13.6.6.11.4.1 Upon receipt of a command

If the device has received a command that has not yet been acknowledged by clearing the BSY bit to zero and an error is encountered, the device shall transmit a Register Device to Host FIS (see Figure 365).

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	ERROR(7:0)							
COUNT(7:0)	na							
COUNT(15:8)	na							
LBA(7:0)	na							
LBA(15:8)	na							
LBA(23:16)	na							
LBA(31:24)	na							
LBA(39:32)	na							
LBA(47:40)	na							
DEVICE(7:0)	na							
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	na	na	ERR

Figure 372 – DURABLE/ORDERED WRITE NOTIFICATION – error on command receipt

Field Definitions

ERROR ATA error code for the failure condition of the failed command.

BSY 0

DRDY 1

DF 0

DRQ 0

ERR 1

Status bit 4 may be set to one.

Following transmission of the Register Device to Host FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.6.11.4.2 During processing of a command

If all commands have been acknowledged by clearing the BSY bit to zero and an error condition is detected, the device shall transmit a Set Device Bits FIS (see Figure 366) to the host. All outstanding commands at the time of an error are aborted as part of the error response and may be re-issued as appropriate by the host. For any commands that have not completed or have completed with error, the device shall clear the corresponding ACT bits to zero in the Set Device Bits FIS.

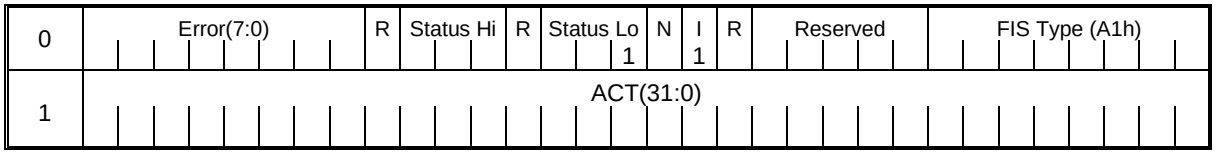


Figure 373 – DURABLE/ORDERED WRITE NOTIFICATION – error during processing

Field Definitions

Error The Error register shall contain the ATA error code.

R Reserved, shall be cleared to zero.

Status As defined in 10.5.7. The ERR bit shall be set to one indicating an NCQ error has occurred. Status bit 4 may be set to one.

I Interrupt bit, the interrupt bit shall be set to one.

ACT The ACT field of the Set Device Bits FIS communicates successful completion notification for each of up to 32 queued commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating successful completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns.

All other fields as defined in 10.5.7.

Only the registers that are updated as part of the Set Device Bits FIS are modified if the device signals an error condition and if the BSY bit in the shadow Status register is cleared to zero, leaving the other Shadow Register Block Registers unchanged. If the device signals an error condition when the BSY bit in the shadow Status register is set to one, then the device clears the BSY bit to zero with a Register Device to Host FIS that updates all registers in the Shadow Register Block.

Following transmission of the Set Device Bits FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.7 RECEIVE FPDMA QUEUED command and subcommand

13.6.7.1 RECEIVE FPDMA QUEUED command definition

The 512 Byte Block DMA IN subcommands make use of this transport command. The RECEIVE FPDMA QUEUED command supports LBA mode only and uses 48 bit addressing only. The format of the command is defined in Figure 374.

Some RECEIVE FPDMA QUEUED subcommands are processed as sequential NCQ commands (see 4.1.1.117). Some RECEIVE FPDMA QUEUED subcommands (see ACS-5) are processed as ordered NCQ commands (see 4.1.1.94).

13.6.7.2 Inputs

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	SECTOR COUNT(7:0)							
FEATURES(15:8)	SECTOR COUNT(15:8)							
COUNT(7:0)	TAG(4:0)					Reserved		
COUNT(15:8)	PRIO(1:0)		Res	SUBCOMMAND(4:0)				
LBA(7:0)	Subcommand Specific							
LBA(15:8)	Subcommand Specific							
LBA(23:16)	Subcommand Specific							
LBA(31:24)	Subcommand Specific							
LBA(39:32)	Subcommand Specific							
LBA(47:40)	Subcommand Specific							
ICC(7:0)	Retired							
AUXILIARY(7:0)	Subcommand Specific							
AUXILIARY(15:8)	Subcommand Specific							
AUXILIARY(23:16)	Subcommand Specific							
AUXILIARY(31:24)	Subcommand Specific							
DEVICE(7:0)	Res	1	Res	0	Reserved			
COMMAND(7:0)	**65h**							

Figure 374 – RECEIVE FPDMA QUEUED command definition

Field Definitions

SECTOR COUNT
: The number of 512 byte blocks to be transferred, 0000h indicates that 65 536 512 byte blocks are to be transferred.

TAG
: The TAG field value shall be assigned by host software to be different from all other TAG values corresponding to outstanding commands. The assigned TAG value shall not exceed the value specified in IDENTIFY DEVICE data Word 75.

PRIO
: The PRIO field value shall be assigned by the host based on the priority of the command issued. The device shall make a best effort to complete High priority requests in a more timely fashion than Normal and Isochronous priority requests. The device shall make a best effort to complete each Isochronous request prior to its associated deadline (see Table 106).

Res
: Reserved, shall be cleared to zero.

SUBCOMMAND
: The SUBCOMMAND field (see Table 111) defines the subcommands that are valid (see 13.6.7.5).

Subcommand Specific

The Subcommand Specific fields are defined for each subcommand as referenced from Table 111.

Upon accepting the command, the device shall clear the BSY bit by transmitting a Register Device to Host FIS to the host with the BSY bit cleared to zero in the Status field of the FIS. The ability for the device to quickly clear the BSY bit allows the host to issue another queued command without blocking on this bit.

The host shall check the BSY bit in the shadow Status register before attempting to issue a new command in order to determine that the device is ready to receive another command (and determine that the host has write access to the Shadow Register Block Registers). The device shall not trigger an interrupt in response to having successfully received the command, so the initial status return that clears the BSY bit shall not have an interrupt associated with it.

13.6.7.3 Success outputs

Upon successful completion of one or more outstanding commands, the device shall transmit a Set Device Bits FIS with the Interrupt bit set to one and one or more bits set to one in the ACT field corresponding to the bit position for each command TAG that has completed since the last status notification was transmitted.

The ACT field occupies the last 32 bits of the Set Device Bits FIS as defined in Figure 375.

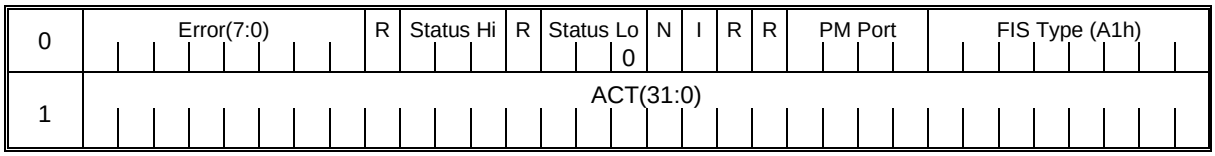


Figure 375 – Set Device Bits FIS for successful RECEIVE FPDMA QUEUED command completion

Field Definitions

Error — The Error register shall be cleared to zero.

R — Reserved, shall be cleared to zero.

Status — As defined in 10.5.7. The ERR bit shall be cleared to zero indicating successful command completion. Bit 4 may be set to one.

I — Interrupt bit, the interrupt bit shall be set to one.

ACT — The ACT field of the Set Device Bits FIS communicates successful completion notification for each of up to 32 queued commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating successful completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns.

All other fields as defined in 10.5.7.

Devices should be aware that if choosing to aggregate status to the point where many of the outstanding commands have actually completed successfully without notification to the host, that an error may cause the final completion status of those commands to be failure. The device should be selective if using status aggregation for outstanding queued commands to ensure the host is made aware of successful completion for outstanding commands so that an error does not force a high number of unnecessary command retries.

13.6.7.4 Error outputs

13.6.7.4.1 Upon Receipt of a Command

If the device has received a command that has not yet been acknowledged by clearing the BSY bit to zero and an error is encountered, the device shall transmit a Register Device to Host FIS (see Figure 376).

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	ERROR(7:0)							
COUNT(7:0)	na							
COUNT(15:8)	na							
LBA(7:0)	na							
LBA(15:8)	na							
LBA(23:16)	na							
LBA(31:24)	na							
LBA(39:32)	na							
LBA(47:40)	na							
DEVICE(7:0)	na							
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	na	na	ERR

Figure 376 – RECEIVE FPDMA QUEUED error status result values on command receipt

Field Definitions

ERROR ATA error code for the failure condition of the failed command.

BSY 0

DRDY 1

DF 0

DRQ 0

ERR 1

Status bit 4 may be set to one.

Following transmission of the Register Device to Host FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.7.4.2 During processing of a command

If all commands have been acknowledged by clearing the BSY bit to zero and an error condition is detected, the device shall transmit a Set Device Bits FIS (see Figure 377) to the host. All outstanding commands at the time of an error shall be aborted as part of the error response and may be re-issued as appropriate by the host. For any commands that have not completed successfully or have resulted in error, the device shall clear the corresponding ACT bits to zero in the Set Device Bits FIS.

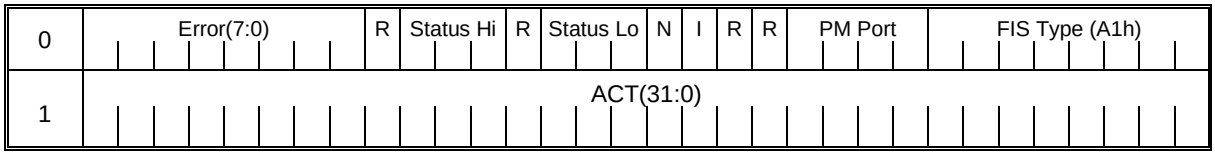


Figure 377 – Set Device Bits FIS with error notification and command completions

Field Definitions

Error The Error register shall contain the ATA error code.

R Reserved, shall be cleared to zero.

Status As defined in 10.5.7. The ERR bit shall be set to one indicating an NCQ error has occurred. Status bit 4 may be set to one.

I Interrupt bit, the interrupt bit shall be set to one.

ACT The ACT field of the Set Device Bits FIS communicates successful completion notification for each of up to 32 queued commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating successful completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns.

All other fields as defined in 10.5.7.

Only the registers that are updated as part of the Set Device Bits FIS are modified if the device signals an error condition and if the BSY bit in the shadow Status register is cleared to zero, leaving the other Shadow Register Block Registers unchanged. If the device signals an error condition and if the BSY bit in the shadow Status register is set to one, then the device clears the BSY bit to zero with a Register Device to Host FIS that updates all registers in the Shadow Register Block.

Following transmission of the Set Device Bits FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.7.5 RECEIVE FPDMA QUEUED subcommands

Subcommands for the RECEIVE FPDMA QUEUED commands are contained within the COUNT field (12:8). The allowed values are defined in Table 111.

Table 111 – Subcommands for RECEIVE FPDMA QUEUED

Value	Description	Reference
00h	Reserved	
01h	READ LOG DMA EXT subcommand	13.6.7.6
02h	ZAC MANAGEMENT IN subcommand	13.6.7.7
03h..1Fh	Reserved	

13.6.7.6 READ LOG DMA EXT subcommand (01h)

13.6.7.6.1 READ LOG DMA EXT subcommand overview

The READ LOG DMA EXT subcommand (see Figure 378) is a sequential NCQ command (see 4.1.1.117). The READ LOG DMA EXT subcommand functionality and behavior is dependent on all requirements of the READ LOG DMA EXT command and the IDENTIFY DEVICE command defined in ACS-4.

13.6.7.6.2 Inputs

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	Contents of READ LOG DMA EXT COUNT(7:0)							
FEATURES(15:8)	Contents of READ LOG DMA EXT COUNT(15:8)							
LBA(7:0)	Contents of READ LOG DMA EXT LBA(7:0)							
LBA(15:8)	Contents of READ LOG DMA EXT LBA(15:8)							
LBA(23:16)	Contents of READ LOG DMA EXT LBA(23:16)							
LBA(31:24)	Contents of READ LOG DMA EXT LBA(31:24)							
LBA(39:32)	Contents of READ LOG DMA EXT LBA(39:32)							
LBA(47:40)	Contents of READ LOG DMA EXT LBA(47:40)							
AUXILIARY(7:0)	Contents of READ LOG DMA EXT FEATURES(7:0)							
AUXILIARY(15:8)	Contents of READ LOG DMA EXT FEATURES(15:8)							
AUXILIARY(23:16)	Reserved							
AUXILIARY(31:24)	Reserved							

Figure 378 – RECEIVE FPDMA QUEUED subcommand = 01h

See ACS-4 for the definition of the READ LOG DMA EXT command.

13.6.7.6.3 Success outputs

See 13.6.7.3.

13.6.7.6.4 Error outputs

See 13.6.7.4.

13.6.7.7 ZAC MANAGEMENT IN subcommand (02h)

13.6.7.7.1 ZAC MANAGEMENT IN subcommand overview

The ZAC MANAGEMENT IN subcommand functionality and behavior is defined in ACS-4.

Some ZAC MANAGEMENT IN subcommands (see ZAC-2) are processed as ordered NCQ commands (see 4.1.1.94).

13.6.7.7.2 Inputs

The format of the command is defined in Figure 379.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	Contents of the ZAC MANAGEMENT IN command COUNT(7:0)							
FEATURES(15:8)	Contents of the ZAC MANAGEMENT IN command COUNT(15:8)							
LBA(7:0)	Contents of the ZAC MANAGEMENT IN command LBA(7:0)							
LBA(15:8)	Contents of the ZAC MANAGEMENT IN command LBA(15:8)							
LBA(23:16)	Contents of the ZAC MANAGEMENT IN command LBA(23:16)							
LBA(31:24)	Contents of the ZAC MANAGEMENT IN command LBA(31:24)							
LBA(39:32)	Contents of the ZAC MANAGEMENT IN command LBA(39:32)							
LBA(47:40)	Contents of the ZAC MANAGEMENT IN command LBA(47:40)							
AUXILIARY(7:0)	Contents of the ZAC MANAGEMENT IN command FEATURE(7:0)							
AUXILIARY(15:8)	Contents of the ZAC MANAGEMENT IN command FEATURE(15:8)							
AUXILIARY(23:16)	Reserved							
AUXILIARY(31:24)	Reserved							

Figure 379 – ZAC MANAGMENT IN subcommand = 02h

See ZAC for the definitions of the ZAC MANAGEMENT IN command.

13.6.7.7.3 Success outputs

See 13.6.8.3.

13.6.7.7.4 Error outputs

See 13.6.8.4.

13.6.8 SEND FPDMA QUEUED command and subcommand

13.6.8.1 SEND FPDMA QUEUED command definition

The 512 Byte Block DMA OUT subcommands make use of this transport command. The SEND FPDMA QUEUED command supports LBA mode only and uses 48 bit addressing only. The format of the command is defined in Figure 380.

Some SEND FPDMA QUEUED subcommands are processed as sequential NCQ commands (see 4.1.1.117). Some SEND FPDMA QUEUED subcommands (see ACS-5) are processed as ordered NCQ commands (see 4.1.1.94).

13.6.8.2 Inputs

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	SECTOR COUNT(7:0)							
FEATURES(15:8)	SECTOR COUNT(15:8)							
COUNT(7:0)	TAG(4:0)					Reserved		
COUNT(15:8)	PRIO(1:0)		Res	SUBCOMMAND(4:0)				
LBA(7:0)	Subcommand Specific							
LBA(15:8)	Subcommand Specific							
LBA(23:16)	Subcommand Specific							
LBA(31:24)	Subcommand Specific							
LBA(39:32)	Subcommand Specific							
LBA(47:40)	Subcommand Specific							
ICC(7:0)	Retired							
AUXILIARY(7:0)	Subcommand Specific							
AUXILIARY(15:8)	Subcommand Specific							
AUXILIARY(23:16)	Subcommand Specific							
AUXILIARY(31:24)	Subcommand Specific							
DEVICE(7:0)	Res	1	Res	0	Reserved			
COMMAND(7:0)	**64h**							

Figure 380 – SEND FPDMA QUEUED command definition

Field Definitions

SECTOR COUNT
: The number of 512 byte blocks to be transferred, 0000h indicates that 65 536 512 byte blocks are to be transferred.

TAG
: The TAG field value shall be assigned by host software to be different from all other TAG values corresponding to outstanding commands. The assigned TAG value shall not exceed the value specified in IDENTIFY DEVICE data Word 75.

PRIO
: The PRIO field value shall be assigned by the host based on the priority of the command issued. The device shall make a best effort to complete High priority requests in a more timely fashion than Normal and Isochronous priority requests. The device shall make a best effort to complete each Isochronous request prior to its associated deadline (see Table 106).

Res
: Reserved, shall be cleared to zero.

SUBCOMMAND
: The SUBCOMMAND field (see Table 112) defines the subcommands that are valid (see 13.6.8.5).

Subcommand Specific

The Subcommand Specific fields are defined for each subcommand as referenced in Table 112.

Upon accepting the command, the device shall clear the BSY bit by transmitting a Register Device to Host FIS to the host with the BSY bit cleared to zero in the Status field of the FIS. The ability for the device to quickly clear the BSY bit allows the host to issue another queued command without blocking on this bit.

The host shall check the BSY bit in the shadow Status register before attempting to issue a new command in order to determine that the device is ready to receive another command (and determine that the host has write access to the Shadow Register Block Registers).

The device shall not trigger an interrupt in response to having successfully received the command, so the initial status return that clears the BSY bit shall not have an interrupt associated with it.

13.6.8.3 Success outputs

Upon successful completion of one or more outstanding commands, the device shall transmit a Set Device Bits FIS with one or more bits set to one in the ACT field corresponding to the bit position for each command TAG that has completed since the last status notification was transmitted.

The ACT field occupies the last 32 bits of the Set Device Bits FIS as defined in Figure 381.

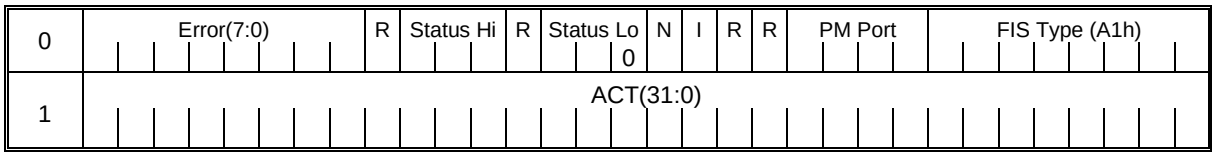


Figure 381 – Set Device Bits FIS for successful SEND FPDMA QUEUED command completion

Field Definitions

Error — The Error register shall be cleared to zero.

R — Reserved, shall be cleared to zero.

Status — As defined in 10.5.7. The ERR bit shall be cleared to zero indicating successful command completion. Bit 4 may be set to one.

I — Interrupt bit, the interrupt bit shall be set to one.

ACT — The ACT field of the Set Device Bits FIS communicates successful completion notification for each of up to 32 queued commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating successful completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns.

All other fields as defined in 10.5.7.

Devices should be aware that if choosing to aggregate status to the point where many of the outstanding commands have actually completed successfully without notification to the host, that an error may cause the final completion status of those commands to be failure. The device should be selective if using status aggregation for outstanding queued commands to ensure the host is made aware of successful completion for outstanding commands so that an error does not force a high number of unnecessary command retries.

13.6.8.4 Error outputs

13.6.8.4.1 Error outputs status

If the device has received a command that has not yet been acknowledged by clearing the BSY bit to zero and an error is encountered, the device shall transmit a Register Device to Host FIS (see Figure 382).

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	ERROR(7:0)							
COUNT(7:0)	na							
COUNT(15:8)	na							
LBA(7:0)	na							
LBA(15:8)	na							
LBA(23:16)	na							
LBA(31:24)	na							
LBA(39:32)	na							
LBA(47:40)	na							
DEVICE(7:0)	na							
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	na	na	ERR

Figure 382 – SEND FPDMA QUEUED error status result values on command receipt

Field Definitions

ERROR ATA error code for the failure condition of the failed command.

BSY 0

DRDY 1

DF 0

DRQ 0

ERR 1

Status bit 4 may be set to one.

Following transmission of the Register Device to Host FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

13.6.8.4.2 During processing of a command

If all commands have been acknowledged by clearing the BSY bit to zero and an error condition is detected, the device shall transmit a Set Device Bits FIS (see Figure 383) to the host. All outstanding commands at the time of an error shall be aborted as part of the error response and may be re-issued as appropriate by the host. For any commands that have not completed successfully or have resulted in error, the device shall clear the corresponding ACT bits to zero in the Set Device Bits FIS.

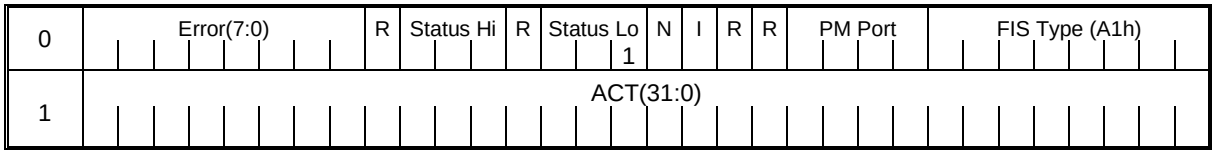


Figure 383 – Set Device Bits FIS with error notification and command completions

Field Definitions

Error The Error register shall contain the ATA error code.

R Reserved, shall be cleared to zero.

Status As defined in 10.5.7. The ERR bit shall be set to one indicating an NCQ error has occurred. Status bit 4 may be set to one.

I Interrupt bit, the interrupt bit shall be set to one.

ACT The ACT field of the Set Device Bits FIS communicates successful completion notification for each of up to 32 queued commands. The field is bit-significant and the device sets bit positions to one for each command tag it is indicating successful completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns.

All other fields as defined in 10.5.7.

Only the registers that are updated as part of the Set Device Bits FIS are modified if the device signals an error condition and if the BSY bit in the shadow Status register is cleared to zero, leaving the other Shadow Register Block Registers unchanged. If the device signals an error condition and if the BSY bit in the shadow Status register is set to one, then the device clears the BSY bit to zero with a Register Device to Host FIS that updates all registers in the Shadow Register Block.

Following transmission of the Set Device Bits FIS, the device shall stop processing any outstanding or new commands until the Queued Error log (see 13.7.4) has been, then abort all outstanding commands. See 13.6.4.4 for more details.

13.6.8.5 SEND FPDMA QUEUED subcommands

Subcommands for the SEND FPDMA QUEUED commands are contained within the COUNT field (12:8). The allowed values are defined in Table 112. Support for each subcommand is reported in the NCQ Send and Receive log (see 13.7.7).

Table 112 – Subcommands for SEND FPDMA QUEUED

Value	Description	Reference
00h	DATA SET MANAGEMENT subcommand	13.6.8.6
01h	HYBRID EVICT subcommand	13.6.8.7
02h	WRITE LOG DMA EXT subcommand	13.6.8.8
03h	ZAC MANAGEMENT OUT subcommand	13.6.8.9
04h	DATA SET MANAGEMENT XL subcommand	13.6.8.10
05h..1Fh	Reserved	

13.6.8.6 DATA SET MANAGEMENT subcommand (00h)

13.6.8.6.1 DATA SET MANAGEMENT subcommand definition

The DATA SET MANAGEMENT subcommand functionality and behavior is dependent on all requirements of the DATA SET MANAGEMENT command and the IDENTIFY DEVICE command defined in ACS-4.

13.6.8.6.2 Inputs

The format of the DATA SET MANAGEMENT subcommand is defined in Figure 384.

Field	7	6	5	4	3	2	1	0
LBA(7:0)	DSM SPECIFIC(7:0)							
LBA(15:8)	DSM SPECIFIC(15:8)							
LBA(23:16)	DSM SPECIFIC(23:16)							
LBA(31:24)	DSM SPECIFIC(31:24)							
LBA(39:32)	DSM SPECIFIC(39:32)							
LBA(47:40)	DSM SPECIFIC(47:40)							
AUXILIARY(7:0)	Reserved							TRIM
AUXILIARY(15:8)	DSM FUNCTION(7:0)							
AUXILIARY(23:16)	Reserved							
AUXILIARY(31:24)	Reserved							

Figure 384 – SEND FPDMA QUEUED subcommand = 00h

Field Definitions

DSM SPECIFIC
As defined by the DATA SET MANAGEMENT command in ACS-4.

TRIM As defined by the DATA SET MANAGEMENT command in ACS-4.

DSM FUNCTION
As defined by the DATA SET MANAGEMENT command in ACS-4.

13.6.8.6.3 Success outputs

See 13.6.8.3.

13.6.8.6.4 Error outputs

See 13.6.8.4.

13.6.8.6.5 Output from the host to the device data structure

As defined in the DATA SET MANAGEMENT command in ACS-4.

13.6.8.7 HYBRID EVICT subcommand (01h)

13.6.8.7.1 HYBRID EVICT subcommand definition

The HYBRID EVICT subcommand evicts data from the non-volatile caching medium to the primary medium. Support for this subcommand is indicated in the NCQ Send and Receive log (see 13.7.7.2).

If the EVICT ALL bit (see Figure 385) is cleared to zero, then for each LBA range specified in the data transferred from the host (see 13.6.8.7.5), the device:

a) shall sync all dirty data in the non-volatile caching medium;
b) may evict the specified LBA range in the non-volatile caching medium; and
c) may free vendor specific resources in the non-volatile caching medium related to the specified LBA range.

If the EVICT ALL bit is set to one, then:

a) the device shall ignore all data transferred from the host (see 13.6.8.7.5); and
b) for all data in the non-volatile caching medium, the device shall:
    A) sync all dirty data in the non-volatile caching medium;
    B) evict all user data in the non-volatile caching medium; and
    C) free vendor specific resources in the non-volatile caching medium.

If the value of the SECTOR COUNT field (see Figure 385) is greater than the value of the MAXIMUM EVICTION DATA BLOCKS field (see 13.7.8.2.16) of the Hybrid Information log, then the device shall return command aborted.

The device may limit the number of HYBRID EVICT commands that are in the queue at the same time. The device shall return command aborted if:

a) a new HYBRID EVICT command is accepted; and
b) the number of HYBRID EVICT commands in the queue was previously equal to the value of the MAXIMUM EVICTION COMMANDS field (see 13.7.8.2.15) of the Hybrid Information log.

If the device processes any reset while processing a HYBRID EVICT command, then the resulting condition of the non-volatile caching medium is indeterminate.

The device should return command completion with no error (see 13.6.8.7.3) after all selected LBA ranges have been successfully evicted.

NOTE 62 - This command may take longer to complete than a typical maximum timeout.

See the SEND FPDMA QUEUED command (see 13.6.8) for the contents of the Count field.

13.6.8.7.2 Inputs

Field	7	6	5	4	3	2	1	0
LBA(7:0)	Reserved							
LBA(15:8)	Reserved							
LBA(23:16)	Reserved							
LBA(31:24)	Reserved							
LBA(39:32)	Reserved							
LBA(47:40)	Reserved							
AUXILIARY(7:0)	Reserved							EVICT ALL
AUXILIARY(15:8)	Reserved							
AUXILIARY(23:16)	Reserved							
AUXILIARY(31:24)	Reserved							

Figure 385 – Subcommand specific parameters for the HYBRID EVICT subcommand = 01h

Field Definition

EVICT ALL The EVICT ALL bit specifies that all of the data in the non-volatile caching medium is to be evicted.

13.6.8.7.3 Success outputs

See 13.6.8.3.

13.6.8.7.4 Error outputs

See 13.6.8.4.

13.6.8.7.5 Output from the host to the device data structure

Figure 386 describes the format for all 512-byte data blocks transferred from the host to the device for the HYBRID EVICT subcommand, containing up to 64 LBA Range entries each. There may be more than one 512-byte data block transferred. The LBA Range entries shall be sorted in order of increasing Starting LBA. If the value of the RANGE LENGTH field of an LBA Range entry is cleared to zero, then the device shall ignore the LBA Range entry and all following LBA Range entries.

For any LBA range, if the Starting LBA plus the Range Length is greater than the maximum LBA, then the device:

a) shall return command aborted; and
b) may evict LBA ranges that are valid.

Byte	Type	Description
0..7	Qword	**LBA Range entry 0** Bits 63:48 RANGE LENGTH field Bits 47:0 STARTING LBA field
8..15	Qword	**LBA Range entry 1** Bits 63:48 RANGE LENGTH field Bits 47:0 STARTING LBA field
…		…
504..511	Qword	**LBA Range entry 63** Bits 63:48 RANGE LENGTH field Bits 47:0 STARTING LBA field

Figure 386 – Output data from the host for the HYBRID EVICT command

13.6.8.8 WRITE LOG DMA EXT subcommand (02h)

13.6.8.8.1 WRITE LOG DMA EXT subcommand definition

The WRITE LOG DMA EXT subcommand is a sequential NCQ command (see 4.1.1.117). The WRITE LOG DMA EXT subcommand functionality and behavior is dependent on all requirements of the WRITE LOG DMA EXT subcommand and the IDENTIFY DEVICE command defined in ACS-4.

13.6.8.8.2 Inputs

The format of the SEND FPDMA QUEUED subcommand is defined in Figure 387.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	Contents of WRITE LOG DMA EXT COUNT(7:0)							
FEATURES(15:8)	Contents of WRITE LOG DMA EXT COUNT(15:8)							
LBA(7:0)	Contents of WRITE LOG DMA EXT LBA(7:0)							
LBA(15:8)	Contents of WRITE LOG DMA EXT LBA(15:8)							
LBA(23:16)	Contents of WRITE LOG DMA EXT LBA(23:16)							
LBA(31:24)	Contents of WRITE LOG DMA EXT LBA(31:24)							
LBA(39:32)	Contents of WRITE LOG DMA EXT LBA(39:32)							
LBA(47:40)	Contents of WRITE LOG DMA EXT LBA(47:40)							
AUXILIARY(7:0)	Reserved							
AUXILIARY(15:8)	Reserved							
AUXILIARY(23:16)	Reserved							
AUXILIARY(31:24)	Reserved							

Figure 387 – SEND FPDMA QUEUED subcommand = 02h

See ACS-4 for the definition of the WRITE LOG DMA EXT subcommand.

13.6.8.8.3 Success outputs

See 13.6.8.3.

13.6.8.8.4 Error outputs

See 13.6.8.4.

13.6.8.9 ZAC MANAGEMENT OUT subcommand (03h)

13.6.8.9.1 ZAC MANAGEMENT OUT subcommand definition

The ZAC MANAGEMENT OUT subcommand functionality and behavior is defined in ACS-4.

Some ZAC MANAGEMENT OUT subcommands (see ZAC-2) are processed as ordered NCQ commands (see 4.1.1.94).

13.6.8.9.2 Inputs

The format of the command is defined in Figure 388.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	Contents of the ZAC MANAGEMENT OUT command COUNT(7:0)							
FEATURES(15:8)	Contents of the ZAC MANAGEMENT OUT command COUNT(15:8)							
LBA(7:0)	Contents of the ZAC MANAGEMENT OUT command LBA(7:0)							
LBA(15:8)	Contents of the ZAC MANAGEMENT OUT command LBA(15:8)							
LBA(23:16)	Contents of the ZAC MANAGEMENT OUT command LBA(23:16)							
LBA(31:24)	Contents of the ZAC MANAGEMENT OUT command LBA(31:24)							
LBA(39:32)	Contents of the ZAC MANAGEMENT OUT command LBA(39:32)							
LBA(47:40)	Contents of the ZAC MANAGEMENT OUT command LBA(47:40)							
AUXILIARY(7:0)	Contents of the ZAC MANAGEMENT OUT command FEATURE(7:0)							
AUXILIARY(15:8)	Contents of the ZAC MANAGEMENT OUT command FEATURE(15:8)							
AUXILIARY(23:16)	Reserved							
AUXILIARY(31:24)	Reserved							

Figure 388 – ZAC MANAGEMENT OUT subcommand = 03h

See ZAC for the definitions of the ZAC MANAGEMENT OUT command.

13.6.8.9.3 Success outputs

See 13.6.8.3.

13.6.8.9.4 Error outputs

See 13.6.8.4.

13.6.8.10 DATA SET MANAGEMENT XL subcommand (04h)

13.6.8.10.1 DATA SET MANAGEMENT XL subcommand definition

The DATA SET MANAGEMENT XL subcommand functionality and behavior is dependent on all requirements of the DATA SET MANAGEMENT XL command and the IDENTIFY DEVICE command defined in ACS-4.

13.6.8.10.2 Inputs

Field	7	6	5	4	3	2	1	0
LBA(7:0)	DSM SPECIFIC(7:0)							
LBA(15:8)	DSM SPECIFIC(15:8)							
LBA(23:16)	DSM SPECIFIC(23:16)							
LBA(31:24)	DSM SPECIFIC(31:24)							
LBA(39:32)	DSM SPECIFIC(39:32)							
LBA(47:40)	DSM SPECIFIC(47:40)							
AUXILIARY(7:0)	Reserved							TRIM
AUXILIARY(15:8)	DSM FUNCTION(7:0)							
AUXILIARY(23:16)	Reserved							
AUXILIARY(31:24)	Reserved							

Figure 389 – DATA SET MANAGEMENT XL subcommand = 04h

Field Definition

DSM SPECIFIC
: As defined by the DATA SET MANAGEMENT XL command in ACS-4.

TRIM
: As defined by the DATA SET MANAGEMENT XL command in ACS-4.

DSM FUNCTION
: As defined by the DATA SET MANAGEMENT XL command in ACS-4.

13.6.8.10.3 Success outputs

See 13.6.8.3.

13.6.8.10.4 Error outputs

See 13.6.8.4.

13.6.8.10.5 Output from the host to the device data structure

As defined in the DATA SET MANAGEMENT XL command in ACS-4.

13.6.9 First-party DMA HBA support (informative)

The Serial ATA native queuing model utilizes the First-party DMA mechanism to allow the device to select the appropriate host memory buffer to transfer data to or from. The First-party DMA mechanism ensures memory protection in order to avoid a rogue or errant device indiscriminately accessing host memory. This is accomplished in Serial ATA by having the device only refer to memory buffers by a DMA Buffer Identifier, rather than through the use of physical memory addresses.

For the NCQ protocol, the identifier used for selecting memory buffers is the same as the unique tag value used to identify the corresponding command. The tags have a value in the range 0 to 31 inclusive and correspond to the tag values assigned by the host at the time commands are issued to the device.

For mainstream desktop host controllers, upon receipt of a DMA Setup FIS the DMA Buffer Identifier may be used by the host as an index into a vector of pointers to pre-constructed PRD tables (physical region descriptor tables, also commonly referred to as scatter/gather lists) that correspond to the memory buffers for the various outstanding queued commands. The pointer in the vector table at the appropriate index may be transferred into the DMA engine as the base pointer for the active PRD table, effectively causing the DMA engine to select the corresponding memory buffer for subsequent data transfers. This allows minimal change to the existing host DMA architecture and provides a streamlined and efficient buffer selection mechanism.

For such an implementation, host software is able to be responsible for pre-constructing corresponding PRD tables and updating the vector table entry prior to issuing a new native queued command. Figure 390 illustrates these concepts (the figure is intended as illustrative and does not exclude other possible host controller implementations).

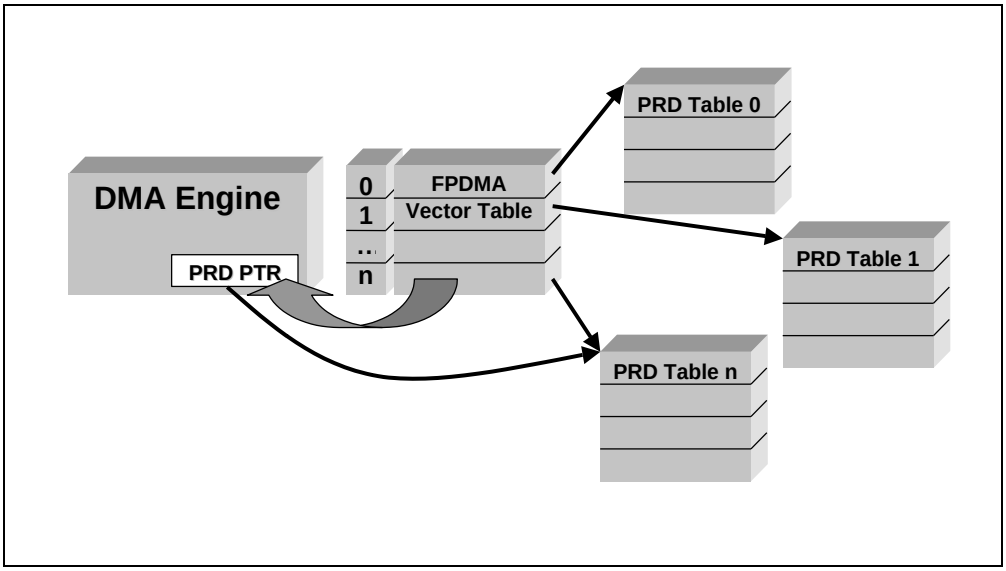


Figure 390 – Example DMA engine indirection for First-party DMA support

This illustrative host controller implementation for supporting First-party DMA has a known shortcoming in that handling non-zero buffer offsets for First-party DMA accesses is cumbersome since the entries in the pre-constructed PRD tables do not necessarily have uniform lengths. For the NCQ model, there is no requirement for non-zero buffer offset support, however, if out of order data delivery within commands is desired (i.e., data for a given command is return by delivering the last half of the data first followed by the first half of the data), support for non-zero buffer offsets is required. See 13.3 for information on non-zero buffer offsets.

13.7 SATA logs

13.7.1 SATA logs overview

There are several log “files” available in a SATA device. Whether a log is read-only or is able to be written is indicated in Table 113. The READ LOG EXT command and the READ LOG DMA EXT command, if supported, in the General Purpose Logging feature set (see ACS-4) are used to read the SATA logs. In some cases the READ LOG DMA EXT command may also be used (see ACS-4). If the log is writeable, then the WRITE LOG EXT command and the WRITE LOG DMA EXT command, if supported, in the General Purpose Logging feature set (see ACS-4) are used to write the SATA logs.

Each log has an “address” that it is referenced. Each log contains zero or more “pages” of data. Each “page” contains 512 bytes of data.

13.7.2 Log address definitions

The log addresses assigned for Serial ATA are defined in Table 113.

Table 113 – Log addresses for Serial ATA

Log Address	Description	R/W	Reference
00h..0Fh	As defined in the ACS-4 standard	-	-
10h	NCQ Queued Error log	RO	13.7.4
11h	Phy Event Counters log	RO	13.9.4
12h	NCQ NON-DATA log	RO	13.7.6
13h	NCQ Send and Receive log	RO	13.7.7
14h	Hybrid Information log	RO	13.7.8
15h	Rebuild Assist log	RW	13.7.9
16h	Out Of Band Management Control log	RW	13.7.10
17h	Reserved	-	-
18h..FFh	As defined in the ACS-4 standard	-	-
Key: RO = Log is read only R/W = Log is read or written			

13.7.3 General purpose log directory (00h)

Devices supporting the Queued Error log (see 13.7.4) reflect this support in the General Purpose Log Directory (see Table 114) log (00h) by having the value 1 at offset 020h and the value 0 at offset 021h of that log to indicate existence of a log at address 10h of at least 1 page in length. For a ZAC device (see 11.2), this log is 2 pages in length.

Devices supporting the Phy Event Counters log reflect this support in the General Purpose Log Directory (00h) by having the value 1 at offset 022h and the value 0 at offset 023h of that log to indicate existence of a log at address 11h of 1 page in length.

Devices supporting the NCQ Non-Data log reflect this support in the General Purpose Log Directory (00h) by having the value 1 at offset 024h and the value 0 at offset 025h of that log to indicate existence of a log at address 12h of 1 page in length.

Devices supporting the NCQ Send and Receive log reflect this support in the General Purpose Log Directory (00h) by having the value 1 at offset 026h and the value 0 at offset 027h of that log to indicate existence of a log at address 13h of 1 page in length.

Devices supporting the Hybrid Information log reflect this support in the General Purpose Log Directory (00h) by having the value 1 at offset 028h and the value 0 at offset 029h of that log to indicate existence of a log at address 14h of 1 page in length.

Devices supporting the Rebuild Assist log reflect this support in the General Purpose Log Directory (00h) by having the value 1 at offset 02Ah and the value 0 at offset 02Bh of that log to indicate existence of a log at address 15h of 1 page in length.

Devices supporting the Out Of Band Management Control log reflect this support in the General Purpose Log Directory (00h) by having the value 1 at offset 02Ch and the value 0 at offset 02Dh of that log to indicate existence of a log at address 16h of 1 page in length.

Table 114 – General purpose log directory values for Serial ATA

Byte	Log	Value
000h..01Fh	-	As defined in the ACS-4 standard
020h	10h	1 if NCQ log is supported, 0 if NCQ log is not supported
021h	10h	0
022h	11h	1 if Phy Event Counters log are supported 0 if Phy Event Counters log are not supported
023h	11h	0
024h	12h	1 if NCQ Non-Data log is supported 0 if NCQ Non-Data log is not supported
025h	12h	0
026h	13h	1 if NCQ Send and Receive log is supported 0 if NCQ Send and Receive log is not supported
027h	13h	0
028h	14h	1 if Hybrid Information log is supported 0 if Hybrid Information log is not supported
029h	14h	0
02Ah	15h	1 if Rebuild Assist log is supported 0 if Rebuild Assist log is not supported
02Bh	15h	0
02Ch	16h	1 if Out Of Band Management log is supported 0 if Out Of Band Management log is not supported
02Dh	16h	0
02Eh..02Fh	-	Reserved
030h..1FFh	-	As defined in the ACS-4 standard

13.7.4 Queued Error Log (10h)

The error-handling scheme for native queued commands halts processing of commands after the host is notified of an error on a native queued command. This allows host software to intervene and take appropriate action to resolve the error and avoids the potential for inconsistency due to data dependencies in the outstanding commands. The host explicitly restarts command processing by issuing a specific command to the device that results in the device aborting all remaining outstanding commands. Because the shadow Status and Error registers are not sufficiently large to contain both information about the error condition and the tag identifying the erring queued command, an additional log has been added in order for the host to be able to retrieve additional information for erring queued commands.

The GPL feature set is defined in ACS-4.

If IDENTIFY DEVICE data Word 76 bit 15 is set to one, the Queued Error log may be read using either of the READ LOG EXT or READ LOG DMA EXT commands.

If IDENTIFY DEVICE data Word 76 bit 15 is cleared to zero, the Queued Error log shall be read using the READ LOG EXT command. An attempt to read the Queued Error log using the READ LOG DMA EXT command shall be aborted and the state of the device shall not change.

Reading the Queued Error log (10h) has the additional side effect as defined in 13.6.3 of aborting any outstanding queued commands and returns a device that has halted due to a queued command error to a state where it has no commands outstanding and is again ready to accept commands (e.g., after completion of a command to read the log the device returns to state DI0:Device_idle state as defined in 11.3). The Queued Error log contains extended command error information.

The Queued Error log reflects the error information for the first recorded NCQ command with error until such time as another NCQ error is encountered after reading the Queued Error log. The contents of the Queued Error log are indeterminate after a software reset or a COMRESET.

Devices supporting the native queued capability shall support the Queued Error log. The Queued Error log is two pages in length and is defined in Figure 391.

If the device supports NCQ Autosense (i.e., IDENTIFY DEVICE data Word 78 bit 7 is set to one), then:

a) the DER bit shall be set as defined in ACS-4;
b) the SENSE KEY field shall be set to values defined in SPC-4;
c) the ADDITIONAL SENSE CODE field shall be set to values defined in SPC-4; and
d) the ADDITIONAL SENSE CODE QUALIFIER field shall be set to values defined in SPC-4.

If the device does not support NCQ Autosense (i.e., IDENTIFY DEVICE data Word 78 bit 7 is cleared to zero), then:

a) the DER bit shall be cleared to zero;
b) the SENSE KEY field shall be cleared to zero;
c) the ADDITIONAL SENSE CODE field shall be cleared to zero; and
d) the ADDITIONAL SENSE CODE QUALIFIER field shall be cleared to zero.

Byte	7	6	5	4	3	2	1	0
0	NQ	UNL	DER	TAG(4:0)				
1	Reserved							
2	STATUS(7:0)							
3	ERROR(7:0)							
4	LBA(7:0)							
5	LBA(15:8)							
6	LBA(23:16)							
7	DEVICE(7:0)							
8	LBA(31:24)							
9	LBA(39:32)							
10	LBA(47:40)							
11	Reserved							
12	COUNT(7:0)							
13	COUNT(15:8)							
14	SENSE KEY(7:0)							
15	ADDITIONAL SENSE CODE(7:0)							
16	ADDITIONAL SENSE CODE QUALIFIER(7:0)							
17	FINAL LBA IN ERROR(7:0)							
18	FINAL LBA IN ERROR(15:8)							
19	FINAL LBA IN ERROR(23:16)							
20	FINAL LBA IN ERROR(31:24)							
21	FINAL LBA IN ERROR(39:32)							
22	FINAL LBA IN ERROR(47:40)							
23 … 255	Reserved							
256 … 510	Vendor Specific							
511	DATA STRUCTURE CHECKSUM(7:0)							
512	WRITE POINTER VALID(7:0)							
513	WRITE POINTER VALID(15:8)							
514	WRITE POINTER VALID(23:16)							
515	WRITE POINTER VALID(31:24)							
516	WRITE POINTER [0] (7:0)							
517	WRITE POINTER [0] (15:8)							
518	WRITE POINTER [0] (23:16)							
519	WRITE POINTER [0] (31:24)							
520	WRITE POINTER [0] (39:32)							
521	WRITE POINTER [0] (47:40)							
...	...							
702	WRITE POINTER [31] (7:0)							
703	WRITE POINTER [31] (15:8)							
704	WRITE POINTER [31] (23:16)							
705	WRITE POINTER [31] (31:24)							
706	WRITE POINTER [31] (39:32)							
707	WRITE POINTER [31] (47:40)							
708 ... 1 022	Reserved							
1 023	WRITE POINTER CHECKSUM(7:0)							

Figure 391 – Queued Error log data structure definition

Field Definitions

NQ — If set to one, indicates that the error condition was a result of a non-queued command having been issued and that the TAG field is therefore not valid. If cleared to zero, indicates that the TAG field is valid and that the error condition applies to a queued command.

UNL — If set to one, indicates that the error condition was a result of receiving an IDLE IMMEDIATE command with the Unload Feature specified. If cleared to zero, the reason for the error was not due to reception of an IDLE IMMEDIATE command with the Unload Feature specified. If the last command received was an Unload Immediate, then the device shall not load the heads to the media if reading the Queued Error log.

If set to one, the NQ bit shall also be set to one to indicate the failure was due to reception of a non-queued command.

If set to one, the value of the STATUS field (7:0), ERROR field (7:0), and LBA field (7:0) (bytes 3..5) in the log shall be set as follows:

a) STATUS field (7:0), the BSY bit shall be cleared to zero and the ERR bit shall be set to one;

b) ERROR field (7:0), the ABRT bit shall be set to one; and

c) LBA field (7:0), shall be set to C4h if the unload is being processed or has completed successfully. Shall be set to 4Ch if the unload was not accepted or has failed.

DER — The DER bit indicates whether the SENSE KEY field, ADDITIONAL SENSE CODE field, and ADDITIONAL SENSE CODE QUALIFIER field describe current information or a deferred error as described in ACS-4.

TAG — If the NQ bit is cleared to zero, the TAG field contains the TAG corresponding to the queued command that failed.

BYTE1..13 — A copy of bytes 1..13 of the Register Device to Host FIS is embedded in the data structure.

ERROR — The value corresponding to the ATA ERROR register value for the command that failed. The command-specific error condition of invalid tag value shall be handled as an invalid command parameter and shall be reported as such (i.e., the ABRT bit set to one in the error register and all other bits cleared to zero).

NOTE 63 - Note that the value returned in the ERROR field of the data structure is separate from the value returned in the Error shadow register if the initial error condition is signaled.

The Error shadow register value is used for the purpose of signaling a queued command error, while the value in the ERROR field of the data structure provides specific information about the error condition that the specific queued command encountered.

SENSE KEY — See SPC-4.

ADDITIONAL SENSE CODE — See SPC-4.

ADDITIONAL SENSE CODE QUALIFIER — See SPC-4.

FINAL LBA IN ERROR — If:

a) the command in error is READ FPDMA QUEUED or WRITE FPDMA QUEUED;

b) the SENSE KEY field is ABORTED COMMAND; and
c) the ADDITIONAL SENSE CODE field /the ADDITIONAL SENSE CODE QUALIFIER field is MULTIPLE READ ERRORS or MULTIPLE WRITE ERRORS,

then the FINAL LBA IN ERROR field shall contain the LBA of the last logical sector in a sequence of contiguous unrecovered logical sector. Otherwise, the FINAL LBA IN ERROR field shall be cleared to zero.

Vendor Specific

Allocated for vendor specific use.

DATA STRUCTURE CHECKSUM

The DATA STRUCTURE CHECKSUM field is the 2's complement of the sum of the first 511 bytes in the data structure. Each byte shall be added with unsigned arithmetic and overflow shall be ignored. The sum of all 512 bytes of the data structure is zero if the checksum is correct.

WRITE POINTER VALID

If Write Pointer[n] contains a valid write pointer corresponding to the command with tag n, then bit n is set to one. If the WRITE POINTER [n] field does not contain a valid write pointer, then bit n is cleared to zero. The WRITE POINTER VALID field is cleared to zero unless the device is in state DFPDMAQ13: WaitforClear. For any outstanding media access commands, the device shall set the corresponding bits to one in the WRITE POINTER VALID field.

WRITE POINTER [n]

The current value of the write pointer in the zone accessed by the operation with tag n.

WRITE POINTER CHECKSUM

The write pointer checksum is the 2's complement of the sum of bytes 512..1 022 in the data structure. Each byte shall be added with unsigned arithmetic and overflow shall be ignored. This sum of bytes 512..1 023 is zero if the checksum is correct.

13.7.5 Phy Event Counters log (11h)

See 13.9.4

13.7.6 NCQ Non-Data log (12h)

13.7.6.1 NCQ Non-Data log overview

To determine the supported NCQ NON-DATA subcommands and their respective features, host software may read log 12h (see Figure 392).

This log shall be supported if the NCQ NON-DATA command is supported (i.e., IDENTIFY DEVICE data Word 77 bit 5 is set to one).

Dword	Bits	Description	Reference
0	**Subcommand 0h**		
	31:5	Reserved	
	4	SUPPORTS ABORT SELECTED TTAG bit	13.7.6.6
	3	SUPPORTS ABORT NON-STREAMING bit	13.7.6.5
	2	SUPPORTS ABORT STREAMING bit	13.7.6.4
	1	SUPPORTS ABORT ALL bit	13.7.6.3
	0	SUPPORTS ABORT NCQ bit	13.7.6.2
1	**Subcommand 1h**		
	31:3	Reserved	
	2	SUPPORTS READ DATA NOT CONTINUE bit	13.7.6.9
	1	SUPPORTS WRITE DATA NOT CONTINUE bit	13.7.6.8
	0	SUPPORTS DEADLINE HANDLING bit	13.7.6.7
2	**Subcommand 2h**		
	31:1	Reserved	
	0	SUPPORTS HYBRID DEMOTE BY SIZE bit	13.7.6.10
3	**Subcommand 3h**		
	31:1	Reserved	
	0	SUPPORTS HYBRID CHANGE BY LBA RANGE bit	13.7.6.11
4	**Subcommand 4h**		
	31:1	Reserved	
	0	SUPPORTS HYBRID CONTROL bit	13.7.6.12
5	**Subcommand 5h**		
	31:1	Reserved	
	0	SUPPORTS SET FEATURES bit	13.7.6.13
6	**Subcommand 6h**		
	31:1	Reserved	
	0	SUPPORTS ZERO EXT bit	13.7.6.14
7	**Subcommand 7h**		
	31:1	Reserved	
	0	SUPPORTS ZAC MANAGEMENT OUT bit	13.7.6.15
8	**Subcommand 8h**		
	31:2	Reserved	
	1	SUPPORTS D/OW bit	13.7.6.17
	0	SUPPORTS DURABLE/ORDERED WRITE NOTIFICATION bit	13.7.6.16
…	…	…	
15	**Subcommand Fh**		
	31:1	Reserved	
	0	Supports subcommand Fh	
16..127	31:0	Reserved	

Figure 392 – NCQ Non-Data log (12h) data structure definition

13.7.6.2 SUPPORTS ABORT NCQ bit

If the SUPPORTS ABORT NCQ bit is set to one, then the device supports the ABORT NCQ QUEUE command (see 13.6.6.3). If the SUPPORTS ABORT NCQ bit is cleared to zero, then the device does not support the ABORT NCQ QUEUE command.

13.7.6.3 SUPPORTS ABORT ALL bit

If the SUPPORTS ABORT ALL bit is set to one, then the device supports the ABORT NCQ QUEUE command (see 13.6.6.3) with the value of 0h in the ABORT TYPE field. If the SUPPORTS ABORT ALL bit is cleared to zero, then the device does not support the ABORT NCQ QUEUE command with the value of 0h in the ABORT TYPE field.

13.7.6.4 SUPPORTS ABORT STREAMING bit

If the SUPPORTS ABORT STREAMING bit is set to one, then the ABORT NCQ QUEUE command (see 13.6.6.3) supports the value of 1h in the ABORT TYPE field. If the SUPPORTS ABORT ALL bit is cleared to zero, then the ABORT NCQ QUEUE command does not support the value of 1h in the ABORT TYPE field.

13.7.6.5 SUPPORTS ABORT NON-STREAMING bit

If the SUPPORTS ABORT NON-STREAMING bit is set to one, then the device supports the ABORT NCQ QUEUE command (see 13.6.6.3) with the value of 2h in the ABORT TYPE field. If the SUPPORTS ABORT ALL bit is cleared to zero, then the device does not support the ABORT NCQ QUEUE command with the value of 2h in the ABORT TYPE field.

13.7.6.6 SUPPORTS ABORT SELECTED TTAG bit

If the SUPPORTS ABORT SELECTED TTAG bit is set to one, then the device supports the ABORT NCQ QUEUE command (see 13.6.6.3) with the value of 3h in the ABORT TYPE field. If the SUPPORTS ABORT ALL bit is cleared to zero, then the device does not support the ABORT NCQ QUEUE command with the value of 3h in the ABORT TYPE field.

13.7.6.7 SUPPORTS DEADLINE HANDLING bit

If the SUPPORTS DEADLINE HANDLING bit is set to one, then the device supports the device supports the DEADLINE HANDLING command (see 13.6.6.4). If the SUPPORTS DEADLINE HANDLING bit is cleared to zero, then the device does not support the DEADLINE HANDLING command.

13.7.6.8 SUPPORTS WRITE DATA NOT CONTINUE bit

If the SUPPORTS WRITE DATA NOT CONTINUE bit is set to one, then the device supports the WDNC bit of the DEADLINE HANDLING command (see 13.6.6.4). If the SUPPORTS WRITE DATA NOT CONTINUE bit is cleared to zero, then the device does not support the WDNC bit of the DEADLINE HANDLING command (see 13.6.6.4).

13.7.6.9 SUPPORTS READ DATA NOT CONTINUE bit

If the SUPPORTS READ DATA NOT CONTINUE bit is set to one, then the device supports the RDNC bit of the DEADLINE HANDLING command (see 13.6.6.4). If the SUPPORTS READ DATA NOT CONTINUE bit is cleared to zero, then the device does not support the RDNC bit of the DEADLINE HANDLING command (see 13.6.6.4).

13.7.6.10 SUPPORTS HYBRID DEMOTE BY SIZE bit

If the SUPPORTS HYBRID DEMOTE BY SIZE bit is set to one, then the device supports the HYBRID DEMOTE BY SIZE subcommand (see 13.6.6.7). If the SUPPORTS HYBRID DEMOTE BY SIZE bit is cleared to zero, then the device does not support the HYBRID DEMOTE BY SIZE subcommand.

13.7.6.11 SUPPORTS HYBRID CHANGE BY LBA RANGE bit

If the SUPPORTS HYBRID CHANGE BY LBA RANGE bit is set to one, then the device supports the HYBRID CHANGE BY LBA RANGE subcommand (see 13.6.6.5). If the SUPPORTS HYBRID CHANGE BY LBA RANGE bit is cleared to zero, then the device does not support the HYBRID CHANGE BY LBA RANGE subcommand.

13.7.6.12 SUPPORTS HYBRID CONTROL bit

If the SUPPORTS HYBRID CONTROL bit is set to one, then the device supports the HYBRID CONTROL subcommand (see 13.6.6.6). If the SUPPORTS HYBRID CONTROL bit is cleared to zero, then the device does not support the HYBRID CONTROL subcommand.

13.7.6.13 SUPPORTS SET FEATURES bit

If the SUPPORTS SET FEATURES bit is set to one, then the device supports the SET FEATURES subcommand (see 13.6.6.8) of the NCQ NON-DATA command. If the SUPPORTS SET FEATURES bit is cleared to zero, then the device does not support the SET FEATURES subcommand of the NCQ NON-DATA command.

13.7.6.14 SUPPORTS ZERO EXT bit

If the SUPPORTS ZERO EXT bit is set to one, then the device supports the ZERO EXT subcommand (see 13.6.6.9) of the NCQ NON-DATA command. If the SUPPORTS ZERO EXT bit is cleared to zero, then the device does not support the ZERO EXT subcommand of the NCQ NON-DATA command.

13.7.6.15 SUPPORTS ZAC MANAGEMENT OUT bit

If the SUPPORTS ZAC MANAGEMENT OUT bit is set to one, then the device supports the ZAC MANAGEMENT OUT subcommand (see 13.6.6.10) of the NCQ NON-DATA command. If the SUPPORTS ZAC MANAGEMENT OUT bit is cleared to zero, then the device does not support the ZAC MANAGEMENT OUT subcommand of the NCQ NON-DATA command.

13.7.6.16 SUPPORTS DURABLE/ORDERED WRITE NOTIFICATION bit

If the SUPPORTS DURABLE/ORDERED WRITE NOTIFICATION bit is set to one, then the device supports the DURABLE/ORDERED WRITE NOTIFICATION subcommand (see 13.6.6.11.1). If the SUPPORTS DURABLE/ORDERED WRITE NOTIFICATION bit is cleared to zero, then the device does not support the DURABLE/ORDERED WRITE NOTIFICATION subcommand.

13.7.6.17 SUPPORTS D/OW bit

If the SUPPORTS D/OW bit is set to one, then the device supports the D/OW bit of the DURABLE/ORDERED WRITE NOTIFICATION subcommand. If the SUPPORTS D/OW bit is cleared to zero, then the device does not support the D/OW bit of the DURABLE/ORDERED WRITE NOTIFICATION subcommand.

13.7.7 NCQ Send and Receive log (13h)

13.7.7.1 NCQ Send and Receive log overview

To determine the supported SEND FPDMA QUEUED and RECEIVE FPDMA QUEUED subcommands and their respective features, host software may read log 13h (see Figure 393).

This log shall be supported if the SEND FPDMA QUEUED and RECEIVE FPDMA QUEUED command is supported (i.e., IDENTIFY DEVICE data Word 77 bit 6 is set to one).

Dword	Bits	Description	Reference
0	**Subcommands Supported**		
	31:3	Reserved	
	2	SUPPORTS DATA SET MANAGEMENT XL bit	13.7.7.2
	1	SUPPORTS HYBRID EVICT bit	13.7.7.3
	0	SUPPORTS DATA SET MANAGEMENT bit	13.7.7.4
1	**Data Set Management**		
	31:1	Reserved	
	0	SUPPORTS TRIM bit	13.7.7.5
2	**Supports Read Log**		
	31:3	Reserved	
	2	SUPPORTS READ LOG FEATURES FIELD ENCAPSULATION bit	13.7.7.6
	1	SUPPORTS SEQUENTIAL READ LOG bit	13.7.7.7
	0	SUPPORTS READ LOG bit	13.7.7.8
3	**Supports Write Log**		
	31:2	Reserved	
	1	SUPPORTS SEQUENTIAL WRITE LOG bit	13.7.7.9
	0	SUPPORTS WRITE LOG bit	13.7.7.10
4	**Supports ZAC Management**		
	31:2	Reserved	
	1	SUPPORTS ZAC MANAGEMENT IN bit	13.7.7.11
	0	SUPPORTS ZAC MANAGEMENT OUT bit	13.7.7.12
5..127		Reserved	

Figure 393 – NCQ Send and Receive log (13h) data structure definition

13.7.7.2 SUPPORTS DATA SET MANAGEMENT XL bit

If the SUPPORTS DATA SET MANAGEMENT XL bit is set to one, then the device supports the DATA SET MANAGEMENT XL subcommand (see 13.6.8.10) of the SEND FPDMA QUEUED command. If the SUPPORTS DATA SET MANAGEMENT XL bit is cleared to zero, then the device does not support the DATA SET MANAGEMENT XL subcommand of the SEND FPDMA QUEUED command.

13.7.7.3 SUPPORTS HYBRID EVICT bit

If the SUPPORTS HYBRID EVICT bit is set to one, then the device supports the HYBRID EVICT subcommand (see 13.6.8.7) of the SEND FPDMA QUEUED command. If the SUPPORTS HYBRID EVICT bit is cleared to zero, then the device does not support the HYBRID EVICT subcommand of the SEND FPDMA QUEUED command.

13.7.7.4 SUPPORTS DATA SET MANAGEMENT bit

If the SUPPORTS DATA SET MANAGEMENT bit is set to one, then the device supports the Data Set Management subcommand (see 13.6.8.6) of the SEND FPDMA QUEUED command. If the SUPPORTS DATA SET MANAGEMENT bit is cleared to zero, then the device does not support the Data Set Management subcommand of the SEND FPDMA QUEUED command.

13.7.7.5 SUPPORTS TRIM bit

If the SUPPORTS TRIM bit is set to one, then the device supports the Trim attribute of the Data Set Management subcommand (see 13.6.8.6) of the SEND FPDMA QUEUED command. If the SUPPORTS DATA SET MANAGEMENT bit is cleared to zero, then the device does not support the Trim attribute of the Data Set Management subcommand of the SEND FPDMA QUEUED command.

13.7.7.6 SUPPORTS READ LOG FEATURES FIELD ENCAPSULATION bit

If the SUPPORTS READ LOG FEATURES FIELD ENCAPSULATION bit is set to one, then the device supports copying the READ LOG DMA EXT Features field to the RECEIVE FPDMA QUEUED inputs (see 13.6.7.6.2). If the SUPPORTS READ LOG FEATURES FIELD ENCAPSULATION bit is cleared to zero, then the device does not copy the READ LOG DMA EXT Features field to the RECEIVE FPDMA QUEUED inputs as defined in a previous revision of this specification (i.e., SATA revision 3.2).

If the SUPPORTS READ LOG FEATURES FIELD ENCAPSULATION bit is set to one, then the SUPPORTS READ LOG bit (see 13.7.7.8) shall be set to one.

13.7.7.7 SUPPORTS SEQUENTIAL READ LOG bit

If the SUPPORTS SEQUENTIAL READ LOG bit is set to one, then the device supports the READ LOG DMA EXT subcommand (see 13.6.7.6) of the RECEIVE FPDMA QUEUED command as a sequential NCQ command. If the SUPPORTS SEQUENTIAL READ LOG bit is cleared to zero, then the device does not support the READ LOG DMA EXT subcommand of the RECEIVE FPDMA QUEUED command as a sequential NCQ command.

If the SUPPORTS SEQUENTIAL READ LOG bit is set to one, then the SUPPORTS READ LOG bit (see 13.7.7.8) shall be set to one.

13.7.7.8 SUPPORTS READ LOG bit

If the SUPPORTS READ LOG bit is set to one, then the device supports the READ LOG DMA EXT subcommand (see 13.6.7.6) of the RECEIVE FPDMA QUEUED command. If the SUPPORTS READ LOG bit is cleared to zero, then the device does not support the READ LOG DMA EXT subcommand of the RECEIVE FPDMA QUEUED command.

13.7.7.9 SUPPORTS SEQUENTIAL WRITE LOG bit

If the SUPPORTS SEQUENTIAL WRITE LOG bit is set to one, then the device supports the WRITE LOG DMA EXT subcommand (see 13.6.8.8) of the SEND FPDMA QUEUED command as a sequential NCQ command. If the SUPPORTS SEQUENTIAL WRITE LOG bit is cleared to zero, then the device does not support the WRITE LOG DMA EXT subcommand of the SEND FPDMA QUEUED command as a sequential NCQ command.

If the SUPPORTS SEQUENTIAL WRITE LOG bit is set to one, then the SUPPORTS WRITE LOG bit (see 13.7.7.10) shall be set to one.

13.7.7.10 SUPPORTS WRITE LOG bit

If the SUPPORTS WRITE LOG bit is set to one, then the device supports the WRITE LOG DMA EXT subcommand (see 13.6.8.8) of the SEND FPDMA QUEUED command. If the SUPPORTS WRITE LOG bit is cleared to zero, then the device does not support the WRITE LOG DMA EXT subcommand of the SEND FPDMA QUEUED command.

13.7.7.11 SUPPORTS ZAC MANAGEMENT IN bit

If the SUPPORTS ZAC MANAGEMENT IN bit is set to one, then the device supports the ZAC MANAGEMENT IN subcommand (see 13.6.7.7) of the RECEIVE FPDMA QUEUED command. If

the SUPPORTS ZAC MANAGEMENT IN bit is cleared to zero, then the device does not support the ZAC MANAGEMENT IN subcommand of the RECEIVE FPDMA QUEUED command.

13.7.7.12 SUPPORTS ZAC MANAGEMENT OUT bit

If the SUPPORTS ZAC MANAGEMENT OUT bit is set to one, then the device supports the ZAC MANAGEMENT OUT subcommand (see 13.6.8.9) of the SEND FPDMA QUEUED command. If the SUPPORTS ZAC MANAGEMENT OUT bit is cleared to zero, then the device does not support the ZAC MANAGEMENT OUT subcommand of the SEND FPDMA QUEUED command.

13.7.8 Hybrid Information log (14h)

13.7.8.1 Hybrid Information log overview

If the Hybrid Information feature is supported (see 13.7.11.2.25), then the Hybrid Information log shall be supported. The Hybrid Information log consists of one page (see Figure 394). The log is read-only. Reading the log shall not cause the device to change power management state.

Byte	Description	Reference
0	HYBRID INFORMATION HEADER field	13.7.8.2
64	HYBRID INFORMATION DESCRIPTOR FOR HYBRID PRIORITY 0 field	13.7.8.3
...	...	-
	HYBRID INFORMATION DESCRIPTOR FOR MAXIMUM PRIORITY (N) field	13.7.8.3
64 + (16 × N) to 511	PADDING field	-

Figure 394 – Hybrid Information log data

Field Definitions

N The number of Hybrid Information Descriptors reported in the log.

Padding Data transfer lengths shall be non-zero multiples of 512 bytes. Pad bytes shall be appended as needed to meet this requirement. Pad bytes shall have a value of 00h.

13.7.8.2 Hybrid Information Header

13.7.8.2.1 Hybrid Information Header overview

Figure 395 describes the Hybrid Information Header that contains summary information for the hybrid device.

Byte	Type	Description	Reference
0..1	Word	**Bits Description** 15:4 Reserved 3:0 NUMBER OF HYBRID INFORMATION DESCRIPTORS field	 13.7.8.2.2
2	Byte	ENABLED field	13.7.8.2.3
3	Byte	HYBRID HEALTH field	13.7.8.2.4
4	Byte	DIRTY LOW THRESHOLD field	13.7.8.2.5
5	Byte	DIRTY HIGH THRESHOLD field	13.7.8.2.6
6	Byte	OPTIMAL WRITE GRANULARITY field	13.7.8.2.7
7	Byte	**Bits Description** 7:4 Reserved 3:0 MAXIMUM HYBRID PRIORITY LEVEL field	 13.7.8.2.8
8	Byte	POWER CONDITION field	13.7.8.2.9
9	Byte	CACHING MEDIUM ENABLED field	13.7.8.2.10
10	Byte	SUPPORTED OPTIONS field	13.7.8.2.11
11		Reserved	-
12.15	Dword	TIME SINCE ENABLED field	13.7.8.2.12
16..23	Qword	NVM SIZE field	13.7.8.2.13
24..31	Qword	ENABLE COUNT field	13.7.8.2.14
32..33	Word	**Bits Description** 7:5 Reserved 4:0 MAXIMUM EVICTION COMMANDS field	 13.7.8.2.15
34..35	Word	MAXIMUM EVICTION DATA BLOCKS field	13.7.8.2.16
36..63		Reserved	-

Figure 395 – Hybrid Information Header

13.7.8.2.2 NUMBER OF HYBRID INFORMATION DESCRIPTORS field

The NUMBER OF HYBRID INFORMATION DESCRIPTORS field indicates the the number of Hybrid Information descriptors that follow the header.

13.7.8.2.3 ENABLED field

Table 115 indicates the value that the device shall indicate in IDENTIFY DEVICE in each case.

Table 115 – Hybrid Information Enabled

Value	Description	IDENTIFY DEVICE data Word 79 bit 9
00h	Hybrid Information Disabled	0
80h	Hybrid Information Disable In Process	0
FFh	Hybrid Information Enabled	1
All other values	Reserved	-

13.7.8.2.4 HYBRID HEALTH field

13.7.8.2.4.1 HYBRID HEALTH field overview

Table 116 describes the values of the HYBRID HEALTH field that contains several indicators of the health of the non-volatile caching medium.

NOTE 64 – If the non-volatile caching medium is healthy, the HYBRID HEALTH field is recommended to be zero.

Table 116 – HYBRID HEALTH field

Bit	Description	Reference
7:4	Reserved	-
3	DATA LOSS bit	13.7.8.2.4.2
2	READ ONLY bit	13.7.8.2.4.3
1	NVM SIZE CHANGED bit	13.7.8.2.4.4
0	UNUSEABLE bit	13.7.8.2.4.5

13.7.8.2.4.2 DATA LOSS bit

If the DATA LOSS bit is set to one, then some of the data in the non-volatile caching medium has become inaccessible since the Hybrid Information log was last read (see 13.20.5.3).

If the DATA LOSS bit is cleared to zero, then no data loss has been detected in the non-volatile caching medium since the Hybrid Information log was last read (see 13.7.8).

13.7.8.2.4.3 READ ONLY bit

If the READ ONLY bit is set to one, then the non-volatile caching medium is read only (see 13.20.5.2).

If the READ ONLY bit is cleared to zero, then the non-volatile caching medium may be read or written.

13.7.8.2.4.4 NVM SIZE CHANGED bit

If the NVM SIZE CHANGED bit is set to one, then the device has changed the NVM Size of the non-volatile caching medium since the Hybrid Information log was last read (see 13.20.5.1).

If the NVM SIZE CHANGED bit is cleared to zero, then the device has not changed the NVM Size of the non-volatile caching medium since the Hybrid Information log was last read.

13.7.8.2.4.5 UNUSEABLE bit

If the UNUSEABLE bit is set to one, then the non-volatile caching medium is no longer useable (see 13.20.5.4).

If the UNUSEABLE bit is cleared to zero, then the non-volatile caching medium is useable.

13.7.8.2.5 DIRTY LOW THRESHOLD field

The DIRTY LOW THRESHOLD field indicates the threshold for the amount of dirty user logical sectors in the non-volatile caching medium that syncing operations should stop. For additional information, see the HYBRID CONTROL subcommand (see 13.6.6.6).

13.7.8.2.6 DIRTY HIGH THRESHOLD field

The DIRTY HIGH THRESHOLD field indicates the threshold for the amount of dirty user logical sectors in the non-volatile caching medium that syncing operations should begin. For additional information, see the HYBRID CONTROL subcommand (see 13.6.6.6).

13.7.8.2.7 OPTIMAL WRITE GRANULARITY field

The OPTIMAL WRITE GRANULARITY field indicates the optimal number of logical sectors for the host to write to the non-volatile caching medium, expressed as a power of 2. If the field contains FFh, then the optimal write granularity is not indicated.

Example: 0 indicates 2^0 = 1 logical sector, 1 indicates 2^1 = 2 logical sectors, 2 indicates 2^2 = 4 logical sectors.

13.7.8.2.8 MAXIMUM HYBRID PRIORITY LEVEL field

The MAXIMUM HYBRID PRIORITY LEVEL field indicates the maximum supported value of the HYBRID PRIORITY field (see 13.20.2.2). The MAXIMUM HYBRID PRIORITY LEVEL field shall be non-zero.

13.7.8.2.9 POWER CONDITION field

The POWER CONDITION field indicates the current power condition that the CHECK POWER MODE command would report in normal outputs (see ACS-4).

13.7.8.2.10 CACHING MEDIUM ENABLED field

The CACHING MEDIUM ENABLED field indicates whether or not the non-volatile caching medium is usable by the host or the device (see 13.6.6.6).

If the CACHING MEDIUM ENABLED field is set to FFh, then the non-volatile caching medium is enabled for use by the host and the device.

If the CACHING MEDIUM ENABLED field is cleared to 00h, then the non-volatile caching medium shall not be used by either the host or the device .

13.7.8.2.11 SUPPORTED OPTIONS field

The SUPPORTED OPTIONS field (see Table 117) indicates that optional behaviors are supported.

Table 117 – SUPPORTED OPTIONS field

Bit	Description
7:2	Reserved
1	SUPPORTS CACHE BEHAVIOR bit
0	MAX PRIORITY BEHAVIOR bit

If the SUPPORTS CACHE BEHAVIOR bit is set to one, then the device supports per command tagging of the HYBRID CHANGE BY LBA RANGE subcommand to control the movement of logical sectors into the non-volatile caching medium if a command specifies a hybrid priority level other than the Max Priority (see 13.6.6.5.1).

If the SUPPORTS CACHE BEHAVIOR bit is cleared to zero, then the device shall ignore the setting of the CACHE BEHAVIOR bit in the HYBRID CHANGE BY LBA RANGE subcommand (see 13.6.6.5) and the subcommand shall be processed as if the CACHE BEHAVIOR bit were cleared to zero.

If the MAX PRIORITY BEHAVIOR bit is set to one (see 13.20.2.2.2), then the device shall:

a) insert logical sectors into the non-volatile caching medium if a command specifies the Maximum Hybrid Priority Level;
b) abort any command that specifies the Maximum Hybrid Priority Level if there are not enough available logical sectors in the non-volatile caching medium;
c) support the HYBRID EVICT subcommand (see 13.6.8.7); and
d) support the HYBRID CHANGE BY LBA RANGE subcommand (see 13.6.6.5).

If the MAX PRIORITY BEHAVIOR bit is cleared to zero (see 13.20.2.2.2), then the device should insert logical sectors into the non-volatile caching medium if a command specifies the Maximum Hybrid Priority Level.

13.7.8.2.12 TIME SINCE ENABLED field

The TIME SINCE ENABLED field indicates the number of power-on hours since the Hybrid Information feature was enabled. This is an unsigned integer. This field shall be cleared to zero each time the Hybrid Information feature is disabled.

13.7.8.2.13 NVM SIZE field

The NVM SIZE field indicates the number of logical sectors that comprise the non-volatile caching medium.

NOTE 65 - The value of the NVM SIZE field may vary over time because of vendor specific factors.

13.7.8.2.14 ENABLE COUNT field

The ENABLE COUNT field contains an unsigned integer that is incremented by one each time the device successfully enables the Hybrid Information feature (see 13.3.11).

13.7.8.2.15 MAXIMUM EVICTION COMMANDS field

The MAXIMUM EVICTION COMMANDS field indicates the maximum number of HYBRID EVICT commands (see 13.6.8.7) that the device supports in the command queue at the same time. A value of zero indicates that the device does not limit the number of HYBRID EVICT commands in the queue.

13.7.8.2.16 MAXIMUM EVICTION DATA BLOCKS field

The MAXIMUM EVICTION DATA BLOCKS field limits the maximum number of data blocks that may be specified in a single HYBRID EVICT subcommand (see 13.6.8.7).

13.7.8.3 Hybrid Information Descriptor

13.7.8.3.1 Hybrid Information Descriptor overview

Figure 396 describes the Hybrid Information Descriptor. There shall be one Hybrid Information Descriptor returned for each supported Hybrid Priority value, in order of increasing Hybrid Priority Level.

Byte	Type	Description	Reference
0	Byte	HYBRID PRIORITY field	13.7.8.3.2
1	Byte	CONSUMED NVM SIZE FRACTION field	13.7.8.3.3
2	Byte	CONSUMED MAPPING RESOURCES FRACTION field	13.7.8.3.4
3	Byte	CONSUMED NVM SIZE FOR DIRTY DATA FRACTION field	13.7.8.3.5
4	Byte	CONSUMED MAPPING RESOURCES FOR DIRTY DATA FRACTION field	13.7.8.3.6
5..15		Reserved	

Figure 396 – Hybrid Information Descriptor

13.7.8.3.2 HYBRID PRIORITY field

The HYBRID PRIORITY field indicates the Hybrid Priority number that this descriptor represents.

13.7.8.3.3 CONSUMED NVM SIZE FRACTION field

The value of the CONSUMED NVM SIZE FRACTION field, when divided by 255, indicates the fraction of the NVM Size for this Hybrid Priority's logical sectors that is currently consumed (i.e., used). The

value is an unsigned integer from 00h to FFh. The value 00h indicates that no NVM Size is currently consumed. The value FFh indicates that all of the NVM Size is currently consumed.

$$\text{Consumed Capacity Fraction} = \frac{A \times 255}{B}$$

where:

A is current number of logical sectors associated with this hybrid priority; and
B is NVM SIZE field value (see 13.7.8.2.13).

13.7.8.3.4 CONSUMED MAPPING RESOURCES FRACTION field

The value of the CONSUMED MAPPING RESOURCES FRACTION field divided by 255 indicates the fraction of the mapping resources for this Hybrid Priority's logical sectors in the non-volatile caching medium that are currently consumed. The value is an unsigned integer from 00h to FFh. The value 00h indicates that no mapping resources are currently consumed. The value FFh indicates that all of the mapping resources are currently consumed.

13.7.8.3.5 CONSUMED NVM SIZE FOR DIRTY DATA FRACTION field

The value of the CONSUMED NVM SIZE FOR DIRTY DATA FRACTION field divided by 255 indicates the fraction of the maximum NVM Size for this Hybrid Priority's data that is currently marked as dirty data. The value is an unsigned integer from 00h to FFh. The value 00h indicates that no NVM Size is currently consumed. The value FFh indicates that all of the NVM Size is currently consumed.

$$\text{Consumed NVM Size for Dirty Data Fraction} = \frac{A \times 255}{B}$$

where:

A is current NVM Size consumed by dirty data associated with this Hybrid Priority level; and
B is NVM SIZE field value (see 13.7.8.2.13).

13.7.8.3.6 CONSUMED MAPPING RESOURCES FOR DIRTY DATA FRACTION field

The value of the CONSUMED MAPPING RESOURCES FOR DIRTY DATA FRACTION field divided by 255 indicates the fraction of the mapping resources for this Hybrid Priority's data in the non-volatile caching medium that are currently consumed for mapping dirty data. The value is an unsigned integer from 00h to FFh. The value 00h indicates that no mapping resources are currently consumed that relate to dirty data. The value FFh indicates that all of the mapping resources are currently consumed that relate to dirty data.

13.7.9 Rebuild Assist log (15h)

If the device supports the Rebuild Assist feature (i.e., IDENTIFY DEVICE data Word 78 bit 11 is set to one), then the Rebuild Assist log shall be supported (see Figure 397).

The Rebuild Assist log shall be accessed using the GPL feature set commands (see ACS-4).

If the Rebuild Assist log is not supported and the host:

a) reads the Rebuild Assist log; or
b) writes the Rebuild Assist log,

then the device shall return command aborted.

The Rebuild Assist log provides information about the Rebuild Assist feature (see 13.21).

If the host writes to the Rebuild Assist log, the device supports the Rebuild Assist feature, and the REBUILD ASSIST ENABLED bit is cleared to zero, then the device shall:

1) disable the Rebuild Assist feature;
    a) clear the rebuild assist enabled bit to zero;
    b) set the PHYSICAL ELEMENT LENGTH field to its default value;
    c) set the DISABLED PHYSICAL ELEMENT MASK field to a vendor specific value; and
    d) clear the disabled physical elements field to zero;
2) ignore all other data from the host; and
3) return command completion with no error.

If the host writes to the Rebuild Assist log and the device supports the Rebuild Assist feature and the REBUILD ASSIST ENABLED bit is set to one, then:

1) if:
    a) the device is unable to enable the Rebuild Assist feature;
    b) the host sets the PHYSICAL ELEMENT LENGTH field to a value other than the value returned when reading the Rebuild Assist Log;
    c) the host attempts to set any bits to one in the DISABLED PHYSICAL ELEMENTS field that are cleared to zero in the DISABLED PHYSICAL ELEMENT MASK field; or
    d) the host attempts to set all bits to one in the DISABLED PHYSICAL ELEMENTS field that are set to one in the DISABLED PHYSICAL ELEMENT MASK field (i.e., attempt to disable all physical elements),

    then the device shall return command aborted and shall not process steps 2, 3, and 4;
2) the device shall enable the Rebuild Assist feature (see 13.21.2);
3) if the device successfully enabled the Rebuild Assist feature, then the device shall logically OR the DISABLED PHYSICAL ELEMENTS field with any prior DISABLED PHYSICAL ELEMENTS field that the device was using (e.g., the host may add bits but shall not clear bits in the field) and save the new value of the DISABLED PHYSICAL ELEMENTS field; and
4) the device shall set IDENTIFY DEVICE data Word 79 bit 11 to one (i.e., Rebuild Assist feature enabled).

If the host reads from the Rebuild Assist log and Rebuild Assist feature is supported, then:

a) if the Rebuild Assist feature is disabled, then the device shall return the supported value for the PHYSICAL ELEMENT LENGTH field, a vendor specific value for the DISABLED PHYSICAL ELEMENT MASK field, and all other fields cleared to zero; and
b) if the Rebuild Assist feature is enabled, then the device:
    A) shall set the REBUILD ASSIST ENABLED bit to one;
    B) shall set the PHYSICAL ELEMENT LENGTH field to its supported value;
    C) may set additional bits in the DISABLED PHYSICAL ELEMENTS field; and
    D) shall not clear any DISABLED PHYSICAL ELEMENTS bits that were previously set by the host.

Byte	Description
0	**Flag Bits** Bits 7:1 Reserved 0 REBUILD ASSIST ENABLED bit
1..6	Reserved
7	PHYSICAL ELEMENT LENGTH field (N)
8	(MSB) DISABLED PHYSICAL ELEMENT MASK field
7+N	(LSB)
8+N	(MSB) DISABLED PHYSICAL ELEMENTS field
7+(2×N)	(LSB)
8+(2×N)..511	Reserved

Figure 397 – Rebuild Assist log

Field Definitions

REBUILD ASSIST ENABLED

Table 118 describes the use of the REBUILD ASSIST ENABLED bit.

Table 118 – REBUILD ASSIST ENABLED bit

Operation	REBUILD ASSIST ENABLED bit	Description
read log	1	the Rebuild Assist feature is enabled
read log	0	the Rebuild Assist feature is disabled
write log	1	request to enable the Rebuild Assist feature
write log	0	request to disable the Rebuild Assist feature

PHYSICAL ELEMENT LENGTH

The PHYSICAL ELEMENT LENGTH field indicates the number of bytes in the DISABLED PHYSICAL ELEMENT MASK field and the number of bytes in the DISABLED PHYSICAL ELEMENTS field.

The device shall ignore any attempt by the host to change the value of this field when writing to the Rebuild Assist log.

DISABLED PHYSICAL ELEMENT MASK

The DISABLED PHYSICAL ELEMENT MASK field indicates that bits in the DISABLED PHYSICAL ELEMENTS field are supported.

The device shall ignore any attempt by the host to change the value of this field when writing to the Rebuild Assist log.

DISABLED PHYSICAL ELEMENTS

The DISABLED PHYSICAL ELEMENTS field specifies if physical elements shall be disabled. Each bit that is set to one in the DISABLED PHYSICAL ELEMENTS field specifies that LBAs associated with this physical element shall respond to read commands and write commands as if the associated LBAs have predicted errors (see 13.21). Each bit that is cleared to zero in the DISABLED PHYSICAL ELEMENTS field specifies that LBAs associated with this physical element shall respond to read commands and write commands as if the associated LBAs do not have predicted errors.

13.7.10 Out Of Band Management log (16h)

13.7.10.1 Out Of Band Management log overview

The Out Of Band Management Control log is one page in length that contains parameters that control the reporting of Out Of Band attributes as defined in SFF-8609 over the Out Of Band management interface (see 13.22). If the OUT OF BAND MANAGEMENT INTERFACE SUPPORTED bit (see 13.7.11.2.29) is set to one, then the Out Of Band Management Control log shall be supported.

The log is readable and writeable.

The persistence of the content of the Out Of Band Management Control log across resets is specified by the VOLATILE bit in the log.

Byte	Description
0..2	Reserved
3	Bits 7:4 Reserved 3:0 NUMBER OF VALID DESCRIPTORS field (N)
4	Bits 7 REPORTING ENABLED bit 6 VOLATILE bit 5:0 Reserved
5	Reserved
6..7	PROTOCOL REVISION CODE field
8..39	1st attribute control descriptor (see Figure 399)
40..71	2nd attribute control descriptor
…	
8+ (32*(N-1)) .. 7+(32*N)	Nth attribute control descriptor

Figure 398 – Out Of Band Management Control log

The NUMBER OF VALID DESCRIPTORS field specifies the number of valid attribute control descriptors contained in the log.

The REPORTING ENABLED bit set to one specifies that the device shall enable the transfer of attribute information over the Out Of Band Management interface (see SFF-8609), based on fields in the attribute control descriptors in this log. The REPORTING ENABLED bit cleared to zero specifies that the device shall not transfer any information over the Out Of Band Management interface. If the CURRENT HARDWARE FEATURE CONTROL IDENTIFIER field (see 13.7.11) is or becomes non-zero, then this bit shall be cleared to zero and shall not be changeable by writing to this log (i.e., writing to the REPORTING ENABLED bit is ignored).

If the REPORTING ENABLED bit is changed from one to zero by writing to this log, then the device transfers the stopping transmission (see SFF-8609) as described in 13.22.1.

If the REPORTING ENABLED bit is changed from zero to one by writing to this log, then the device transfers the protocol revision code packet (see SFF-8609) as described in 13.22.1.

The VOLATILE bit specifies if the contents of the Out Of Band Management Control log page is persistent across a hardware reset and power on reset. If the VOLATILE bit is set to one, then the contents of the log page shall:

a) not persist across a hardware reset and power on reset; and
b) after that reset:
    A) be set to the contents of the log page the last time this log was written with the VOLATILE bit cleared to zero; or
    B) be set to the manufacturer default log page values, if the log page has never been written with the VOLATILE bit cleared to zero.

If the VOLATILE bit is cleared to zero, then the contents of the Out Of Band Management Control log page shall persist across any resets.

The PROTOCOL REVISION CODE field specifies the revision of SFF-8609 specification implemented. The SFF-8609 revision code consists of two numeric values separated by a period (e.g., SFF-8609 Revision 1.2). The PROTOCOL REVISION CODE field is encoded such that the first byte (i.e., most significant byte) of this field contains the numerical value of the revision that precedes the period and the second byte (i.e., least significant byte) of this field contains the numerical value of the revision that follows the period (e.g., SFF-8609 Revision 1.2 is encoded as 0102h). This field shall not be changeable by writing to this log (i.e., writing to the PROTOCOL REVISION CODE field is ignored).

The attribute control descriptors contains a list of descriptors that controls if the attribute identified by the descriptor identifier is transferred over the Out Of Band Management interface and other related control settings.

Figure 399 defines the format of the attribute control descriptor.

Byte	Description
0	Bits 7:4 Reserved 3:0 DESCRIPTOR IDENTIFIER field
1	Reserved
2..31	Attribute control descriptor specific

Figure 399 – Attribute control descriptor format

The DESCRIPTOR IDENTIFIER field specifies the attribute associated with this descriptor. Figure 400 defines the values of the DESCRIPTOR IDENTIFIER field.

Value [a]	Description	Reference
0	Temperature attribute control	Figure 401
All Others	Restricted for SFF-8609	
[a] The code values are the same as the Data Code values in the Data Type Definition (see SFF-8609).		

Figure 400 – DESCRIPTOR IDENTIFIER field

The attribute control descriptor specific parameters contain parameters that control the reporting of the specific descriptor type (i.e., attribute) based on the descriptor identifier.

13.7.10.2 Temperature attribute control descriptor format

Figure 401 defines the format of the Temperature attribute control descriptor.

Byte	Description
0	Bits 7:4 Reserved 3:0 DESCRIPTOR IDENTIFIER field
1..3	Reserved
4	Bits 7:1 Reserved 0 TEMPERATURE REPORTING ENABLED bit
5	REPORTING INTERVAL field
6	MINIMUM REPORTING INTERVAL field
7	Bits 7:4 CHANGE UP field 3:0 CHANGE DOWN field
8	Bits 7:2 Reserved 1:0 TEST MODE field
9	Reserved
10	TEST MODE TEMPERATURE field
11..31	Reserved

Figure 401 – Temperature attribute control descriptor format

The TEMPERATURE REPORTING ENABLED bit set to one specifies that reporting of this attribute is enabled over the Out Of Band Management interface. The TEMPERATURE REPORTING ENABLED bit cleared to zero specifies that reporting of this attribute is not enabled.

The REPORTING INTERVAL field specifies the interval in seconds (i.e., how often) that the device should transfer this attribute over the Out Of Band Management interface as described in this subclause. The interval is from the start of the transfer of this attribute over the Out Of Band Management interface to the start of the next transfer of this attribute. If the device processes a General Purpose Logging feature set command that sets the REPORTING INTERVAL field to zero, then the device shall:

a) return command aborted; and
b) set sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN PARAMETER LIST, if the Sense Data Reporting feature set is supported and enabled (see ACS-4).

If the OUT OF BAND TEMPERATURE CHANGE REPORTING SUPPORTED bit (see 13.7.11.2.29) is set to one, then the MINIMUM REPORTING INTERVAL field specifies the minimum time in seconds that shall elapse between the start of the transfer of this attribute over the Out Of Band Management interface and the start of the next transfer of this attribute, without regard for whether the transfer of this attribute is the result of a value in the REPORTING INTERVAL field, the CHANGE UP field, or the CHANGE DOWN field.

If the OUT OF BAND TEMPERATURE CHANGE REPORTING SUPPORTED bit is set to one and the device processes a General Purpose Logging feature set command that set the MINIMUM REPORTING INTERVAL field to a value greater than or equal to REPORTING INTERVAL field, then the device shall:

a) return command aborted; and
b) set sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN PARAMETER LIST, if the Sense Data Reporting feature set is supported and enabled (see ACS-4).

If the OUT OF BAND TEMPERATURE CHANGE REPORTING SUPPORTED bit is set to one and the CHANGE UP field is:

a) cleared to zero, then no amount of increase in temperature shall result in the device transferring this attribute, except as specified by the REPORTING INTERVAL field; or
b) set to a non-zero value, then an increase in temperature from the last time the attribute was transferred that is greater than or equal to the number of degrees Celsius specified in the CHANGE UP field shall result in the device transferring this attribute as described in this section.

If the OUT OF BAND TEMPERATURE CHANGE REPORTING SUPPORTED bit is set to one and the CHANGE DOWN field is:

a) cleared to zero, then no amount of decrease in temperature shall result in the device transferring this attribute, except as specified by the REPORTING INTERVAL field; or
b) set to a non-zero value, then an decrease in temperature from the last time the attribute was transferred that is greater than or equal to the number of degrees Celsius specified in the CHANGE DOWN field shall result in the device transferring this attribute as described in this section.

The device shall transfer this attribute over the Out Of Band Management interface, if:

a) the TEMPERATURE REPORTING ENABLED bit is set to one;
b) the interval since the most recent transfer of this attribute is greater than or equal to the minimum reporting interval, if non-zero; and
c) at least one of the following conditions is met:
    A) the interval since the most recent transfer of this attribute is greater than or equal to the reporting interval;
    B) a temperature increase has occurred that is greater than or equal to the non-zero value in the CHANGE UP field; or
    C) a temperature decrease has occurred that is greater than or equal to the non-zero value in the CHANGE DOWN field.

If the OUT OF BAND TEMPERATURE CHANGE REPORTING SUPPORTED bit is set to one and the device processes a General Purpose Logging feature set command that:

a) clears the MINIMUM REPORTING INTERVALfield to zero; and
b) sets:
    A) the CHANGE UP field to a non-zero value; or
    B) the CHANGE DOWN field to a non-zero value,

then the device shall:

a) return command aborted; and
b) set sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN PARAMETER LIST, if the Sense Data Reporting feature set is supported and enabled (see ACS-4).

The TEST MODE field enables a test mode for this attribute that allows for simulating temperature conditions as described in Table 119.

If the TEST MODE field is set to a non-zero value and the device processes General Purpose Logging feature set command that changes any of the changeable fields in this descriptor, then the device should restart the test mode as if the TEST MODE field was just changed to another non-zero value (e.g., if a test mode sequence is in progress, and the device processes a General Purpose Logging feature set command that changes REPORTING INTERVAL field, then the device should restart the test mode sequence using the new reporting interval value.

Table 119 - TEST MODE field

Value	Description [a]
00b	Test mode is disabled and the device should transfer the actual temperature of the Serial ATA device over the Out Of Band Management interface based on other fields in this descriptor.
01b	A test mode is enabled and the device should transfer a sequence of incrementing temperature values expressed in two's complement over the Out Of Band Management interface. If the device is transferring the attribute information, then the device shall start at the temperature specified in the TEST MODE TEMPERATURE field and increment the reported temperature by one every reporting interval (i.e., based on the REPORTING INTERVAL field). Once the temperature being reported reaches 7Fh (i.e., 127 degrees Celsius), the device shall stop incrementing the reported temperature value and continue to report this value until: a) the test mode is disabled (i.e., TEST MODE field is cleared to 00b); b) the test mode is changed to a different test mode; or c) one of the following occurs: A. software reset; B. power-on reset; or C. a hard reset.
10b	A test mode is enabled and the device should transfer a sequence of decrementing temperature values expressed in two's complement over the Out Of Band Management interface. If the device is transferring the attribute information, then the device shall start at the temperature specified in the TEST MODE TEMPERATURE field and decrement the reported temperature by one every reporting interval (i.e., based on the REPORTING INTERVAL field). Once the temperature being reported reaches 80h (i.e., -128 degrees Celsius), the device shall stop decrementing the reported temperature value and continue to report this value until: a) the test mode is disabled (i.e., TEST MODE field is cleared to 00b); b) the test mode is changed to a different test mode; or c) one of the following occurs: A. software reset; B. power-on reset; or C. a hard reset.
11b	A test mode is enabled and the device should transfer the temperature specified in the TEST MODE TEMPERATURE field over the Out Of Band Management interface for every reporting interval (i.e., based on the REPORTING INTERVAL field).

[a] If a test mode is enabled (i.e., TEST MODE field is set to non-zero value), and the device stopped transferring this attribute information as part of a change to the standby mode (see ACS-4) as described in 13.22.1, then as a result of the device returning to the active mode (see ACS-4) or idle mode (see ACS-4) the device should continue transferring this attribute information from where it left off in the test mode (e.g., if the device was transferring a sequence of incrementing temperatures as part of the TEST MODE field being set to 01b and 75 degrees Celsius was the last temperature transferred prior to a changing to a standby mode, then after changing back to an active mode or idle mode, the device should resume transferring the temperature continuing at 76 degrees Celsius).

The TEST MODE TEMPERATURE field specifies the temperature in degrees Celsius in two's complement notation used in test modes specified by the TEST MODE field. This field is ignored if the TEST MODE field is cleared to 00b.

13.7.11 Identify Device Data log (30h)

13.7.11.1 Serial ATA settings (page 08h)

The Serial ATA log page (see Table 120) provides information about the Serial ATA Transport.

Table 120 – Serial ATA (page 08h) (part 1 of 3)

Offset	Type	Contents	Reference
0..7	Qword	**Serial ATA Page Information Header**	
		Bit **Meaning** 63 Shall be set to one. 62:33 Reserved 32:16 PAGE NUMBER field. Shall be set to 08h. 15:0 REVISION NUMBER field. Shall be set to 0001h.	
8..15	Qword	**SATA Capabilities**	
		Bit **Meaning**	
		63 Shall be set to one.	
		62:34 Reserved	
		33 OUT OF BAND TEMPERATURE CHANGE REPORTING SUPPORTED bit	13.7.11.2.30
		32 OUT OF BAND MANAGEMENT INTERFACE SUPPORTED bit	13.7.11.2.29
		31 POWER DISABLE FEATURE ALWAYS ENABLED bit	13.7.11.2.16
		30 POWER DISABLE FEATURE SUPPORTED bit	13.7.11.3.10
		29 REBUILD ASSIST SUPPORTED bit	13.7.11.2.27
		28 DIPM SSP PRESERVATION SUPPORTED bit	13.7.11.2.26
		27 HYBRID INFORMATION SUPPORTED bit	13.7.11.2.25
		26 DEVSLEEP TO REDUCEDPWRSTATE CAPABILITY SUPPORTED bit	13.7.11.2.15
		25 DEVICE SLEEP SUPPORTED bit	13.7.11.2.24
		24 NCQ AUTOSENSE SUPPORTED bit	13.7.11.2.23
		23 SOFTWARE SETTINGS PRESERVATION SUPPORTED bit	13.7.11.2.22
		22 HARDWARE FEATURE CONTROL SUPPORTED bit	13.7.11.2.21
		21 IN-ORDER DATA DELIVERY SUPPORTED bit	13.7.11.2.20
		20 DEVICE INITIATED POWER MANAGEMENT SUPPORTED bit	13.7.11.2.19
		19 DMA SETUP AUTO-ACTIVATION SUPPORTED bit	13.7.11.2.18
		18 NON-ZERO BUFFER OFFSETS SUPPORTED bit	13.7.11.2.17
		17 SEND AND RECEIVE QUEUED COMMANDS SUPPORTED bit	13.7.11.2.14
		16 NCQ NON-DATA COMMAND SUPPORTED bit	13.7.11.2.13
		15 NCQ STREAMING SUPPORTED bit	13.7.11.2.12
		14 READ LOG DMA EXT AS EQUIVALENT TO READ LOG EXT SUPPORTED bit	13.7.11.2.11
		13 DEVICE AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED bit	13.7.11.2.10
		12 HOST AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED bit	13.7.11.2.9
		11 NCQ PRIORITY INFORMATION SUPPORTED bit	13.7.11.2.8
		10 UNLOAD WHILE NCQ COMMANDS ARE OUTSTANDING SUPPORTED bit	13.7.11.2.7
		9 SATA PHY EVENT COUNTERS LOG SUPPORTED bit	13.7.11.2.6
		8 RECEIPT OF HOST INITIATED POWER MANAGEMENT REQUESTS SUPPORTED bit	13.7.11.2.5
		7 NCQ FEATURE SET SUPPORTED bit	13.7.11.2.4
		6:3 Reserved	
		2 SATA GEN3 SIGNALING SPEED SUPPORTED bit	13.7.11.2.3
		1 SATA GEN2 SIGNALING SPEED SUPPORTED bit	13.7.11.2.2
		0 SATA GEN1 SIGNALING SPEED SUPPORTED bit	13.7.11.2.1

Table 120 – Serial ATA (page 08h) (part 2 of 3)

Offset	Type	Contents	Reference
16..23	Qword	**Current SATA Settings**	
		Bit **Meaning**	
		63 Shall be set to one.	
		62:14 Reserved	
		13 HYBRID INFORMATION ENABLED bit	13.7.11.3.12
		12 REBUILD ASSIST ENABLED bit	13.7.11.3.11
		11 POWER DISABLE FEATURE ENABLED bit	13.7.11.3.10
		10 DEVICE SLEEP ENABLED bit	13.7.11.3.9
		9 AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS ENABLED bit	13.7.11.3.8
		8 SOFTWARE SETTINGS PRESERVATION ENABLED bit	13.7.11.3.7
		7 HARDWARE FEATURE CONTROL ENABLED bit	13.7.11.3.6
		6 IN-ORDER DATA DELIVERY ENABLED bit	13.7.11.3.5
		5 DEVICE INITIATED POWER MANAGEMENT ENABLED bit	13.7.11.3.4
		4 DMA SETUP AUTO-ACTIVATION ENABLED bit	13.7.11.3.3
		3 NON-ZERO BUFFER OFFSETS ENABLED bit	13.7.11.3.2
		2:0 CURRENT NEGOTIATED SERIAL ATA SIGNAL SPEED field	13.7.11.3.1
24..39		Reserved	
40..41	Word	CURRENT HARDWARE FEATURE CONTROL IDENTIFIER field	13.7.11.3.13
42..43	Word	SUPPORTED HARDWARE FEATURE CONTROL IDENTIFIER field	13.7.11.3.14
44..47		Reserved	
48..55	Qword	**DEVSLP Timing Variables**	
		Bit **Meaning**	
		63 DEVSLP TIMING VARIABLES SUPPORTED bit	13.7.11.4.1
		62:16 Reserved	
		15:8 DevSleep Exit Timeout (DETO) field	13.7.11.4.2
		7:5 Reserved	
		4:0 Minimum DEVSLP Assertion Time (MDAT) field	13.7.11.4.3

Table 120 – Serial ATA (page 08h) (part 3 of 3)

Offset	Type	Contents		Reference
56..63	Qword	**Transitional Energy Reporting**		
		Bit	**Meaning**	
		63	TER SUPPORTED bit	13.7.11.5.14
		62:55	Reserved	
		54:53	IN-STATE DEVSLEEP POWER UNIT field	13.7.11.5.13
		52:48	TYPICAL IN-STATE DEVSLEEP POWER field	13.7.11.5.12
		47:46	OFF TO GOOD STS LATENCY TIME UNIT field	13.7.11.5.11
		45:40	OFF TO GOOD STS LATENCY field	13.7.11.5.10
		39:38	BETWEEN POWER CYCLES TIME UNIT field	13.7.11.5.9
		37:24	RECOMMENDED TIME BETWEEN POWER CYCLES field	13.7.11.5.8
		23:22	OFF TO GOOD STS TIME UNIT field	13.7.11.5.7
		21:16	OFF TO GOOD STS RECOUP COST field	13.7.11.5.6
		15:14	DEVSLEEP TO PHYRDY TIME UNIT field	13.7.11.5.5
		13:8	DEVSLEEP TO PHYRDY RECOUP COST field	13.7.11.5.4
		7:6	SLUMBER TO DEVSLEEP TIME UNIT field	13.7.11.5.3
		5:0	SLUMBER TO DEVSLEEP RECOUP COST field	13.7.11.5.2
64..71	Qword	**Transitional Energy Reporting Extended**		
		Bit	**Meaning**	
		63	TERE SUPPORTED bit	13.7.11.6.14
		62:47	Reserved	
		46:45	IN-STATE SLUMBER POWER UNITS field	13.7.11.6.13
		44:40	TYPICAL IN-STATE SLUMBER POWER field	13.7.11.6.12
		39:38	PM2 TO PM0 LATENCY TIME UNIT field	13.7.11.6.11
		37:32	PM2 TO PM0 LATENCY field	13.7.11.6.10
		31:30	OFF TO GOOD STS RELATIVE TO DEVSLEEP/PM2 TIME UNIT field	13.7.11.6.9
		29:24	OFF TO GOOD STS RELATIVE TO DEVSLEEP/PM2 RECOUP COST field	13.7.11.6.8
		23:22	DEVSLEEP/PM2 TO OFF TIME UNIT field	13.7.11.6.7
		21:16	DEVSLEEP/PM2 TO OFF RECOUP COST field	13.7.11.6.6
		15:14	PM2 TO PM0 TIME UNIT field	13.7.11.6.5
		13:8	PM2 TO PM0 RECOUP COST field	13.7.11.6.4
		7:6	PM0 TO PM2 TIME UNIT field	13.7.11.6.3
		5:0	PM0 TO PM2 RECOUP COST field	13.7.11.6.2
72..511		Reserved		

13.7.11.2 SATA Capabilities

13.7.11.2.1 SATA GEN1 SIGNALING SPEED SUPPORTED bit

If the SATA GEN1 SIGNALING SPEED SUPPORTED bit is set to one, then the device supports the Gen1 signaling rate of 1.5 Gbit/s.

IDENTIFY DEVICE data Word 76 bit 1 is a copy of this bit.

13.7.11.2.2 SATA GEN2 SIGNALING SPEED SUPPORTED bit

If the SATA GEN2 SIGNALING SPEED SUPPORTED bit is set to one, then the device supports the Gen2 signaling rate of 3.0 Gbit/s.

IDENTIFY DEVICE data Word 76 bit 2 is a copy of this bit.

13.7.11.2.3 SATA GEN3 SIGNALING SPEED SUPPORTED bit

If the SATA GEN3 SIGNALING SPEED SUPPORTED bit is set to one, then the device supports the Gen3 signaling rate of 6.0 Gbit/s.

IDENTIFY DEVICE data Word 76 bit 3 is a copy of this bit.

13.7.11.2.4 NCQ FEATURE SET SUPPORTED bit

If the NCQ FEATURE SET SUPPORTED bit is set to one, then the device supports the NCQ feature set (see 13.6).

IDENTIFY DEVICE data Word 76 bit 8 is a copy of this bit.

13.7.11.2.5 RECEIPT OF HOST INITIATED POWER MANAGEMENT REQUESTS SUPPORTED bit

If the RECEIPT OF HOST INITIATED POWER MANAGEMENT REQUESTS SUPPORTED bit is set to one, then the device supports Partial and Slumber interface power management states (see 8.4.2) when initiated by the host.

If the DEVICE INITIATED POWER MANAGEMENT SUPPORTED bit is cleared to zero, then the RECEIPT OF HOST INITIATED POWER MANAGEMENT REQUESTS SUPPORTED bit shall be set to one.

IDENTIFY DEVICE data Word 76 bit 9 is a copy of this bit.

13.7.11.2.6 SATA PHY EVENT COUNTERS LOG SUPPORTED bit

If the SATA PHY EVENT COUNTERS LOG SUPPORTED bit is set to one, then the device supports the SATA Phy Event Counters log (see 13.9.4).

IDENTIFY DEVICE data Word 76 bit 10 is a copy of this bit.

13.7.11.2.7 UNLOAD WHILE NCQ COMMANDS ARE OUTSTANDING SUPPORTED bit

If the UNLOAD WHILE NCQ COMMANDS ARE OUTSTANDING SUPPORTED bit is set to one, then the device supports moving the heads to a safe position upon reception of the IDLE IMMEDIATE command with the Unload Feature specified while NCQ commands are outstanding. This bit shall only be set to one if the NCQ FEATURE SET SUPPORTED bit is set to one.

IDENTIFY DEVICE data Word 76 bit 11 is a copy of this bit.

13.7.11.2.8 NCQ PRIORITY INFORMATION SUPPORTED bit

If the NCQ PRIORITY INFORMATION SUPPORTED bit is set to one, then the device supports the PRIORITY field (see 13.6.4.1) in the READ FPDMA QUEUED command and WRITE FPDMA QUEUED command and optimization based on this information. This bit shall only be set to one if the NCQ FEATURE SET SUPPORTED bit (see 13.7.11.2.4) is set to one.

IDENTIFY DEVICE data Word 76 bit 12 is a copy of this bit.

13.7.11.2.9 HOST AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED bit

If the HOST AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED bit is set to one, then the device supports host automatic partial to slumber transitions. If the RECEIPT OF HOST INITIATED POWER MANAGEMENT REQUESTS SUPPORTED bit is cleared to zero, then the HOST AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED bit shall be cleared to zero.

The device shall tolerate a Partial exit latency up to the max Slumber exit latency. This allows the host to asynchronously transition from Partial to Slumber.

If the RECEIPT OF HOST INITIATED POWER MANAGEMENT REQUESTS SUPPORTED bit is cleared to zero, then the HOST AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED bit shall be cleared to zero.

IDENTIFY DEVICE data Word 76 bit 13 is a copy of this bit.

13.7.11.2.10 DEVICE AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED bit

If the DEVICE AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED bit is set to one, then the device supports device automatic partial to slumber transitions and may asynchronously transition from Partial to Slumber when enabled. If the DEVICE AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED bit is cleared to zero (i.e., device initiating interface power management is not supported), then the DEVICE AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED bit shall be cleared to zero.

If the DEVICE INITIATED POWER MANAGEMENT SUPPORTED (see 13.7.11.2.19) bit is cleared to zero, then the DEVICE AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED bit shall be cleared to zero.

IDENTIFY DEVICE data Word 76 bit 14 is a copy of this bit.

13.7.11.2.11 READ LOG DMA EXT AS EQUIVALENT TO READ LOG EXT SUPPORTED bit

If the READ LOG DMA EXT AS EQUIVALENT TO READ LOG EXT SUPPORTED bit is set to one, then the READ LOG DMA EXT command (see ACS-4) and the READ LOG EXT command (see ACS-4) may be used in all cases with identical results. If the GPL DMA SUPPORTED bit is cleared to zero, this bit shall be cleared to zero. This bit shall only be set to one if the NCQ FEATURE SET SUPPORTED bit (see 13.7.11.2.4) is set to one.

If the READ LOG DMA EXT AS EQUIVALENT TO READ LOG EXT SUPPORTED bit is cleared to zero and the device indicates command acceptance for a READ LOG DMA EXT command to read the Queued Error log (see 13.7.4) or the Phy Event Counters log (see 13.9.4), then the device shall return command aborted.

IDENTIFY DEVICE data Word 76 bit 15 is a copy of this bit.

13.7.11.2.12 NCQ STREAMING SUPPORTED bit

If the NCQ STREAMING SUPPORTED bit is set to one, then the device supports NCQ Streaming. This bit shall only be set to one if the NCQ FEATURE SET SUPPORTED bit (see 13.7.11.2.4) is set to one.

IDENTIFY DEVICE data Word 77 bit 4 is a copy of this bit.

13.7.11.2.13 NCQ NON-DATA COMMAND SUPPORTED bit

If the NCQ NON-DATA COMMAND SUPPORTED bit is set to one, then the device supports the NCQ NON-DATA command. This bit shall only be set to one if the NCQ FEATURE SET SUPPORTED bit (see 13.7.11.2.4) is set to one.

IDENTIFY DEVICE data Word 77 bit 5 is a copy of this bit.

13.7.11.2.14 SEND AND RECEIVE QUEUED COMMANDS SUPPORTED bit

If the SEND AND RECEIVE QUEUED COMMANDS SUPPORTED bit is set to one, then the device supports the RECEIVE FPDMA QUEUED command and the SEND PDMA QUEUED command. This bit shall only be set to one if the NCQ FEATURE SET SUPPORTED bit (see 13.7.11.2.4) is set to one.

IDENTIFY DEVICE data Word 77 bit 6 is a copy of this bit.

13.7.11.2.15 DEVSLEEP_TO_REDUCEDPWRSTATE CAPABILITY SUPPORTED bit

If the DEVSLEEP_TO_REDUCEDPWRSTATE CAPABILITY SUPPORTED bit is set to one, then:

a) the device supports maintaining whether it was in Partial or Slumber after detection of assertion, and subsequent detection of negation, of DEVSLP; and
b) the DEVICE SLEEP SUPPORTED bit shall be set to one.

If the DEVSLEEP_TO_REDUCEDPWRSTATE CAPABILITY SUPPORTED bit is cleared to zero, then the device does not support remembering whether it was in Partial or Slumber after detection of assertion, and subsequent detection of negation, of DEVSLP.

IDENTIFY DEVICE data Word 77 bit 7 is a copy of this bit.

13.7.11.2.16 POWER DISABLE FEATURE ALWAYS ENABLED bit

If the POWER DISABLE FEATURE ALWAYS ENABLED bit is set to one, then:

a) the Power Disable feature is always enabled (see 8.6);
b) the DEVICE SLEEP SUPPORTED bit (see 13.7.11.2.24) shall be cleared to zero;
c) the POWER DISABLE FEATURE ENABLED bit (see 13.7.11.3.10) shall be set to one; and
d) the value of the POWER DISABLE FEATURE ALWAYS ENABLED bit and the value of the POWER DISABLE FEATURE ENABLED bit shall persist across all resets.

If the POWER DISABLE FEATURE ALWAYS ENABLED bit is cleared to zero and the POWER DISABLE FEATURE SUPPORTED bit (see 13.7.11.2.28) is set to one, then the Power Disable feature may be enabled using the SET FEATURES Enable/Disable Power Disable Feature subcommand (see 13.3.12).

IDENTIFY DEVICE data Word 77 bit 8 is a copy of this bit.

13.7.11.2.17 NON-ZERO BUFFER OFFSETS SUPPORTED bit

If the NON-ZERO BUFFER OFFSETS SUPPORTED bit is set to one, then the device supports transmission and reception of DMA Setup FISes with a non-zero value in the DMA Buffer Offset field of the FIS. If the NON-ZERO BUFFER OFFSETS SUPPORTED bit is cleared to zero, then the device supports transmission and reception of the DMA Setup FIS only with the DMA Buffer Offset field cleared to zero.

IDENTIFY DEVICE data Word 78 bit 1 is a copy of this bit.

13.7.11.2.18 DMA SETUP AUTO-ACTIVATION SUPPORTED bit

If the DMA SETUP AUTO-ACTIVATION SUPPORTED bit is set to one, then the device supports the use of the DMA Setup FIS Auto-Activate optimization as described in 10.5.9.4.2. When set to one the device supports use of the Auto-Activate optimization. If the DMA SETUP AUTO-ACTIVATION SUPPORTED bit is cleared to zero, then the device does not support the Auto-Activate optimization.

IDENTIFY DEVICE data Word 78 bit 2 is a copy of this bit.

13.7.11.2.19 DEVICE INITIATED POWER MANAGEMENT SUPPORTED bit

If the DEVICE INITIATED POWER MANAGEMENT SUPPORTED bit is set to one the device supports device initiated power management (DIPM) requests. If the DEVICE INITIATED POWER MANAGEMENT SUPPORTED bit is cleared to zero, the device does not support device initiated power management requests.

If the RECEIPT OF HOST INITIATED POWER MANAGEMENT REQUESTS SUPPORTED bit is cleared to zero, then the DEVICE INITIATED POWER MANAGEMENT SUPPORTED bit shall be set to one.

IDENTIFY DEVICE data Word 78 bit 3 is a copy of this bit.

13.7.11.2.20 IN-ORDER DATA DELIVERY SUPPORTED bit

If the IN-ORDER DATA DELIVERY SUPPORTED bit is set to one, the device supports guaranteed in-order data delivery when non-zero buffer offsets are used for commands in the NCQ feature set. If the IN-ORDER DATA DELIVERY SUPPORTED bit is set to one, then the device guarantees in-order data delivery for READ FPDMA QUEUED or WRITE FPDMA QUEUED commands when non-zero buffer offsets are used with multiple DMA Setup FIS. Target data is delivered in order, starting with the first LBA through command completion. If the IN-ORDER DATA DELIVERY SUPPORTED bit is cleared to zero, then the device does not guarantee in-order data delivery when non-zero buffer offsets are enabled. In this case, data may be interleaved both within a command and across multiple commands. By default this field shall be cleared to zero.

IDENTIFY DEVICE data Word 78 bit 4 is a copy of this bit.

13.7.11.2.21 HARDWARE FEATURE CONTROL SUPPORTED bit

If the HARDWARE FEATURE CONTROL SUPPORTED bit is set to one, then the device supports the extended uses of the Hardware Feature Control pin(s) (see 13.10). If the HARDWARE FEATURE CONTROL SUPPORTED bit is cleared to zero, then the extended uses of the Hardware Feature Control pin(s) are not supported and the HARDWARE FEATURE CONTROL ENABLED bit is cleared to zero (see 13.10).

IDENTIFY DEVICE data Word 78 bit 5 is a copy of this bit.

13.7.11.2.22 SOFTWARE SETTINGS PRESERVATION SUPPORTED bit

If the SOFTWARE SETTINGS PRESERVATION SUPPORTED bit is set to one, then the device supports the Software Settings Preservation (SSP) feature set (see 13.5).

IDENTIFY DEVICE data Word 78 bit 6 is a copy of this bit.

13.7.11.2.23 NCQ AUTOSENSE SUPPORTED bit

If the NCQ AUTOSENSE SUPPORTED bit is set to one, then the device supports NCQ Autosense (see B.15). This bit shall only be set to one if the NCQ FEATURE SET SUPPORTED bit (see 13.7.11.2.4) is set to one.

IDENTIFY DEVICE data Word 78 bit 7 is a copy of this bit.

13.7.11.2.24 DEVICE SLEEP SUPPORTED bit

If the DEVICE SLEEP SUPPORTED bit is set to one, then:

a) the device supports the Device Sleep feature;
b) the device shall support the Identify Device Data log;
c) the DEVSLP TIMING VARIABLES SUPPORTED bit (see 13.7.11.4.1) shall be set to one; and
d) the POWER DISABLE FEATURE ALWAYS ENABLED bit (see 13.7.11.2.16) shall be cleared to zero.

If the DEVICE SLEEP SUPPORTED bit is cleared to zero, then the device does not support the Device Sleep feature.

IDENTIFY DEVICE data Word 78 bit 8 is a copy of this bit.

> NOTE 66 – If the DEVICE SLEEP SUPPORTED bit is cleared to zero, then the host ignores the DEVSLEEP TO REDUCEDPWRSTATE CAPABILITY SUPPORTED bit and the DEVSLP TIMING VARIABLES SUPPORTED bit.

13.7.11.2.25 HYBRID INFORMATION SUPPORTED bit

If the HYBRID INFORMATION SUPPORTED bit is set to one, then the device supports the hybrid information feature (see 13.20). If the device does not support the hybrid information feature, then the HYBRID INFORMATION SUPPORTED bit shall be cleared to zero.

IDENTIFY DEVICE data Word 78 bit 9 is a copy of this bit.

13.7.11.2.26 DIPM SSP PRESERVATION SUPPORTED bit

If the DIPM SSP PRESERVATION SUPPORTED bit is set to one, then the device supports persistence of the Device Initiated Interface Power Management enable/disable setting via Software Settings Preservation.

IDENTIFY DEVICE data Word 78 bit 10 is a copy of this bit.

13.7.11.2.27 REBUILD ASSIST SUPPORTED bit

If the REBUILD ASSIST SUPPORTED bit is set to one, then the device supports the Rebuild Assist feature (see 13.21). This bit shall only be set to one if the device supports NCQ as shown in bit 8 of Word 76. The host may determine if the Rebuild Assist feature is enabled or disabled by reading the Rebuild Assist log or by reading IDENTIFY DEVICE data Word 79 bit 11.

IDENTIFY DEVICE data Word 78 bit 11 is a copy of this bit.

13.7.11.2.28 POWER DISABLE FEATURE SUPPORTED bit

If the POWER DISABLE FEATURE SUPPORTED bit is set to one, then the device supports Power Disable (see 8.6).

If the POWER DISABLE FEATURE SUPPORTED bit is cleared to zero, then:

a) the device does not support the Power Disable feature;
b) the POWER DISABLE FEATURE ALWAYS ENABLED bit (see 13.7.11.2.16) shall be cleared to zero; and
c) the POWER DISABLE FEATURE ENABLED bit (see 13.7.11.3.10) shall be cleared to zero.

IDENTIFY DEVICE data Word 78 bit 12 is a copy of this bit.

13.7.11.2.29 OUT OF BAND MANAGEMENT INTERFACE SUPPORTED bit

If the OUT OF BAND MANAGEMENT INTERFACE SUPPORTED bit is set to one, then the device supports the Out Of Band Management interface (see 13.22).

If the OUT OF BAND MANAGEMENT INTERFACE SUPPORTED bit is cleared to zero, then the device does not supports the Out Of Band Management interface.

IDENTIFY DEVICE data Word 77 bit 9 is a copy of this bit.

13.7.11.2.30 OUT OF BAND TEMPERATURE CHANGE REPORTING SUPPORTED BIT

If the OUT OF BAND TEMPERATURE CHANGE REPORTING bit is set to one, then the device supports the MINIMUM REPORTING INTERVAL field, the CHANGE UP field, and the CHANGE DOWN field in the temperature attribute control descriptor (see 13.7.10.2).

If the OUT OF BAND TEMPERATURE CHANGE REPORTING bit is cleared to zero, then the MINIMUM REPORTING INTERVAL field, the CHANGE UP field, and the CHANGE DOWN field in the temperature attribute control descriptor are reserved.

13.7.11.3 SATA Current Settings

13.7.11.3.1 CURRENT NEGOTIATED SERIAL ATA SIGNAL SPEED field

The CURRENT NEGOTIATED SERIAL ATA SIGNAL SPEED field is a coded value that indicates the Serial ATA Phy speed (see Table 121) that the device is currently communicating.

IDENTIFY DEVICE data Word 77 bits 3:1 is a copy of this field.

Table 121 – Coded values for negotiated Serial ATA signaling speed

Coded Values	Description
000b	Reporting of current signaling speed is not supported
001b	Current signaling speed is Gen1
010b	Current signaling speed is Gen2
011b	Current signaling speed is Gen3
All other values	Reserved

NOTE 67 - In the case of system configurations that have more than one Phy link in the data path (e.g., port multiplier), the indicated speed is only relevant for the link between the device Phy and its immediate host Phy. It is possible for each link in the data path to negotiate a different Serial ATA signaling speed.

13.7.11.3.2 NON-ZERO BUFFER OFFSETS ENABLED bit

If the NON-ZERO BUFFER OFFSETS ENABLED bit is set to one, then device transmission of DMA Setup FISes with a non-zero value in the Buffer Offset field of the FIS is enabled.

If the NON-ZERO BUFFER OFFSETS ENABLED bit is cleared to zero, then the device is permitted to transmit DMA Setup FIS only with the Buffer Offset field cleared to zero.

By default this field shall be cleared to zero.

IDENTIFY DEVICE data Word 79 bit 1 is a copy of this bit.

13.7.11.3.3 DMA SETUP AUTO-ACTIVATION ENABLED bit

If the DMA SETUP AUTO-ACTIVATION ENABLED bit is set to one, then the device may utilize the DMA Setup FIS Auto-Activate optimization as described in 10.5.9.4.2.

If the DMA SETUP AUTO-ACTIVATION ENABLED bit is cleared to zero, then the device shall not utilize the Auto-Activate optimization.

By default, this field shall be cleared to zero.

IDENTIFY DEVICE data Word 79 bit 2 is a copy of this bit.

13.7.11.3.4 DEVICE INITIATED POWER MANAGEMENT ENABLED bit

If the DEVICE INITIATED POWER MANAGEMENT ENABLED bit is set to one, then the device may initiate power management transition requests.

If the DEVICE INITIATED POWER MANAGEMENT ENABLED bit is cleared to zero, then the device shall not initiate interface power management requests to the host.

This field shall be cleared to zero by default.

IDENTIFY DEVICE data Word 79 bit 3 is a copy of this bit.

13.7.11.3.5 IN-ORDER DATA DELIVERY ENABLED bit

If the IN-ORDER DATA DELIVERY ENABLED bit is set to one and NON-ZERO BUFFER OFFSETS ENABLED bit is set to one (see 13.7.11.3.2), then the device may satisfy a READ FPDMA QUEUED or WRITE FPDMA QUEUED command by transmitting multiple DMA Setup FISes with non-zero buffer offset values where appropriate, provided that the target data is delivered in order, starting with the first LBA through command completion.

If the IN-ORDER DATA DELIVERY ENABLED bit is cleared to zero, then the device may interleave data both in a command and across multiple commands using non-zero buffer offsets if the NON-ZERO BUFFER OFFSETS ENABLED bit is set to one.

By default this field shall be cleared to zero.

IDENTIFY DEVICE data Word 79 bit 4 is a copy of this bit.

13.7.11.3.6 HARDWARE FEATURE CONTROL ENABLED bit

If the HARDWARE FEATURE CONTROL ENABLED bit is set to one, then device support for the extended uses of the Hardware Feature Control pin(s) (see 13.10) are enabled. If the HARDWARE FEATURE CONTROL ENABLED bit is cleared to zero, then:

a) the extended uses of the Hardware Feature Control pin(s) are disabled; and
b) the default uses of the Hardware Feature Control pin(s) are not affected (see 13.10).

IDENTIFY DEVICE data Word 79 bit 5 is a copy of this bit.

13.7.11.3.7 SOFTWARE SETTINGS PRESERVATION ENABLED bit

If the SOFTWARE SETTINGS PRESERVATION ENABLED bit is set to one, then the SSP feature set is enabled and the device shall preserve specified software settings across COMRESET (see 13.5).

If the SOFTWARE SETTINGS PRESERVATION ENABLED bit is cleared to zero, then the SSP feature set is disabled and the device shall clear specified software settings if a COMRESET occurs (see 13.5).

If the SOFTWARE SETTINGS PRESERVATION SUPPORTED bit (see 13.7.11.2.22) is set to one, then SOFTWARE SETTINGS PRESERVATION ENABLED bit shall be set to one after a power on reset has been processed. If the SOFTWARE SETTINGS PRESERVATION SUPPORTED bit is cleared to zero, then the SOFTWARE SETTINGS PRESERVATION ENABLED bit shall be cleared to zero by default.

IDENTIFY DEVICE data Word 79 bit 6 is a copy of this bit.

13.7.11.3.8 AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS ENABLED bit

If the AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS ENABLED bit is set to one, then the device may asynchronously transition from Partial to Slumber. If the AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS ENABLED bit is cleared to zero, then shall not asynchronously transition from Partial to Slumber.

If the DEVICE INITIATED POWER MANAGEMENT ENABLED bit is cleared to zero, then the AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS ENABLED bit shall be cleared to zero.

IDENTIFY DEVICE data Word 79 bit 7 is a copy of this bit.

13.7.11.3.9 DEVICE SLEEP ENABLED bit

If the DEVICE SLEEP ENABLED bit is set to one, then the Device Sleep feature (see 13.3.10) is enabled. If the DEVICE SLEEP ENABLED bit is cleared to zero, then the Device Sleep feature is disabled.

IDENTIFY DEVICE data Word 79 bit 8 is a copy of this bit.

13.7.11.3.10 POWER DISABLE FEATURE ENABLED BIT

If the POWER DISABLE FEATURE ENABLED bit is set to one, then the Power Disable feature is enabled (see 8.6).

If the POWER DISABLE FEATURE ENABLED bit is cleared to zero, then the Power Disable feature is disabled.

If the POWER DISABLE FEATURE ALWAYS ENABLED bit is cleared to zero, after processing:

a) a power-on reset, the value of the POWER DISABLE FEATURE ENABLED bit shall be cleared to zero;
b) a hardware reset, the value of the POWER DISABLE FEATURE ENABLED bit shall not be changed; and
c) a software reset, the value of the POWER DISABLE FEATURE ENABLED bit shall not be changed.

If the host attempts to enable the Power Disable feature and the Device Sleep feature is enabled (i.e., IDENTIFY DEVICE data Word 79 bit 8 is set to one), then the device shall return command aborted.

IDENTIFY DEVICE data Word 79 bit 10 is a copy of this bit.

13.7.11.3.11 REBUILD ASSIST ENABLED bit

If the REBUILD ASSIST ENABLED bit is set to one, then the Rebuild Assist feature (see 13.21) is enabled. This bit shall only be set to one if the device supports the Rebuild Assist feature (see 13.7.11.2.27) and the device supports NCQ (see 13.7.11.2.4).

IDENTIFY DEVICE data Word 79 bit 11 is a copy of this bit.

13.7.11.3.12 HYBRID INFORMATION ENABLED bit

If the HYBRID INFORMATION ENABLED bit is set to one, then the Hybrid Information feature is enabled. If the device does not support the Hybrid Information feature (see 13.7.11.2.25), then the HYBRID INFORMATION ENABLED bit shall be cleared to zero.

IDENTIFY DEVICE data Word 79 bit 9 is a copy of this bit.

13.7.11.3.13 CURRENT HARDWARE FEATURE CONTROL IDENTIFIER FIELD

If the CURRENT HARDWARE FEATURE CONTROL IDENTIFIER field is non-zero, then Table 105 describes the current Hardware Feature Control behavior. If the CURRENT HARDWARE FEATURE CONTROL IDENTIFIER field is cleared to zero, then the current Hardware Feature Control behavior shall be DSS, DAS, or neither.

13.7.11.3.14 SUPPORTED HARDWARE FEATURE CONTROL IDENTIFIER FIELD

The SUPPORTED HARDWARE FEATURE CONTROL IDENTIFIER field (see Table 105) indicates the value that is permitted for the CURRENT HARDWARE FEATURE CONTROL IDENTIFIER field.

13.7.11.4 DEVSLP Timing Variables

13.7.11.4.1 DEVSLP TIMING VARIABLES SUPPORTED bit

If the DEVSLP TIMING VARIABLES SUPPORTED bit is set to one, then the device supports DEVSLP Timing Variables Qword.

13.7.11.4.2 DEVSLEEP EXIT TIMEOUT (DETO) field

The DevSleep Exit Timeout (DETO) field contains the maximum time, in milliseconds, from when DEVSLP is negated, to when the device shall be ready to detect OOB signals. If the value in the DETO field is zero, then the host should use 20 ms as the value of DETO. See 8.5 for more information.

13.7.11.4.3 MINIMUM DEVSLP ASSERTION TIME (MDAT) field

The Minimum DEVSLP Assertion Time (MDAT) field contains the minimum time, in milliseconds that the host shall assert DEVSLP, once it has been asserted. If the value in the MDAT field is zero, then the host should use 10 ms as the value of MDAT. See 8.5 for more information.

13.7.11.5 Transitional Energy Reporting

13.7.11.5.1 TRANSITIONAL ENERGY REPORTING FIELD overview

See ACS-4 for descriptions of PM0: Active and PM2: Standby.

The TRANSITIONAL ENERGY REPORTING field is defined as follows:

a) the TRANSITIONAL ENERGY REPORTING field is valid only under typical device operating parameters (e.g., temperature); and
b) the TRANSITIONAL ENERGY REPORTING field only relates to the transition to the requested power states, described in this subclause, immediately preceded by 32, 1 MB random write requests.

13.7.11.5.2 SLUMBER TO DEVSLEEP RECOUP COST field

The SLUMBER TO DEVSLEEP RECOUP COST field indicates the nominal number of time units the device needs to remain in a DevSleep interface power state in order to recoup the energy consumed by transitioning to the DevSleep interface power state from the Slumber interface power state, relative to operating in the Slumber interface power state. If the SLUMBER TO DEVSLEEP RECOUP COST field is cleared to zero, then the device does not support reporting the Slumber to DevSleep transitional energy. The amount of energy consumed during the transition to the DevSleep interface power state is measured from assertion of the DEVSLP sideband signal until the device completes transition to the DevSleep interface power state. If the SLUMBER TO DEVSLEEP RECOUP COST field is set to all ones, then transitioning to the DevSleep interface power state from the Slumber interface power state has no power advantage.

NOTE 68 - The accuracy of the nominal value of the SLUMBER TO DEVSLEEP RECOUP COST field may change over time.

The area under the curve, above the DEVSLP interface power state line, between assertion of the DEVSLP sideband signal and the transition to the DevSleep interface power state (see shaded area A in Figure 402) is the energy consumed to calculate the Slumber to DevSleep transitional energy time value (SLUMBER TO DEVSLEEP RECOUP COST field).

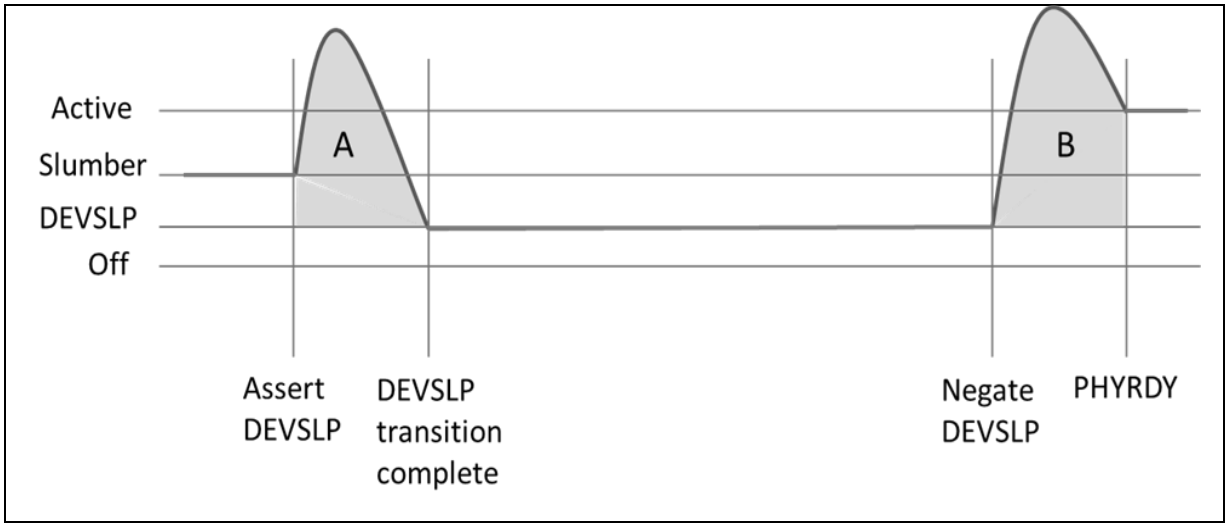


Figure 402 – Transitional Energy, Slumber to DevSleep, and DevSleep to PHYRDY

13.7.11.5.3 SLUMBER TO DEVSLEEP TIME UNIT field

The SLUMBER TO DEVSLEEP TIME UNIT field indicates the time units as defined in Table 122 represented in the SLUMBER TO DEVSLEEP RECOUP COST field.

Table 122 – Transition Time Units

Value	Time Units
0	1 millisecond units
1	10 millisecond units
2	100 millisecond units
3	1 second units

13.7.11.5.4 DEVSLEEP TO PHYRDY RECOUP COST field

The DEVSLEEP TO PHYRDY RECOUP COST field indicates the nominal recommended number of time units the device needs to remain in a DevSleep interface power state in order to recoup the energy consumed by transitioning to PHYRDY from the DevSleep interface power state, relative to operating in the PHYRDY state. If the DEVSLEEP TO PHYRDY RECOUP COST field is cleared to zero, then the device does not support reporting the DevSleep to PHYRDY transitional energy. The amount of energy consumed during the transition to the PHYRDY is measured from negation of the DEVSLP sideband signal until PHYRDY.

NOTE 69 - The accuracy of the nominal value of the DEVSLEEP TO PHYRDY RECOUP COST field may change over time.

The area under the curve and above the DEVSLP interface state line, between negation of the DEVSLP sideband signal and PHYRDY (see shaded area B in Figure 402) is the energy consumed to calculate the DEVSLP to PHYRDY transitional energy time value (DEVSLEEP TO PHYRDY RECOUP COST field).

13.7.11.5.5 DEVSLEEP TO PHYRDY TIME UNIT field

The DEVSLEEP TO PHYRDY TIME UNIT field indicates the time units as defined in Table 122 represented in the DEVSLEEP TO PHYRDY RECOUP COST field.

13.7.11.5.6 OFF TO GOOD STS RECOUP COST field

The OFF TO GOOD STS RECOUP COST field indicates the nominal number of time units the device needs to remain in the Power Off condition in order to recoup the energy consumed between DHR2: Send_good_status (see 11.2) from Off, relative to operating in the PM0: Active state. If the OFF TO GOOD STS RECOUP COST field is cleared to zero, then the device does not support reporting Off to DHR2: Send_good_status transitional energy. The amount of energy consumed during the transition is measured from Power On until DHR2: Send_good_status.

NOTE 70 - The accuracy of the nominal value of the OFF TO GOOD STS RECOUP COST field may change over time.

The area under the curve and above the Off line between Power On and DHR2: Send_good_status (see shaded area A in Figure 403) is the energy consumed to calculate the Off to DHR2: Send_good_status transitional energy time value (OFF TO GOOD STS RECOUP COST field).

Figure 403 – Transitional Energy, PM0: Active to Off, and Off to PM0: Active

13.7.11.5.7 OFF TO GOOD STS TIME UNIT field

The OFF TO GOOD STS TIME UNIT field indicates the time units as defined in Table 122 represented in the OFF TO GOOD STS RECOUP COST field.

13.7.11.5.8 RECOMMENDED TIME BETWEEN POWER CYCLES field

The RECOMMENDED TIME BETWEEN POWER CYCLES field indicates the recommended number of minimum time units between power cycles.

13.7.11.5.9 BETWEEN POWER CYCLES TIME UNIT field

The BETWEEN POWER CYCLES TIME UNIT field indicates the time units as defined in Table 123 represented in the RECOMMENDED TIME BETWEEN POWER CYCLES field.

Table 123 – Power Cycle Time Units

Value	Time Units
0	**1 millisecond units**
1	**1 second units**
2	**10 second units**
3	**30 second units**

13.7.11.5.10 OFF TO GOOD STS LATENCY field

The OFF TO GOOD STS LATENCY field indicates the typical number of time units to transition from Power Off to DHR2: Send_good_status.

13.7.11.5.11 OFF TO GOOD STS LATENCY TIME UNIT field

The OFF TO GOOD STS LATENCY TIME UNIT field indicates the time units as defined in Table 122 represented in the OFF TO GOOD STS LATENCY field.

13.7.11.5.12 TYPICAL IN-STATE DEVSLEEP POWER field

The TYPICAL IN-STATE DEVSLEEP POWER field indicates the typical number of in-state power units consumed while the device is in the DevSleep interface power state.

13.7.11.5.13 IN-STATE DEVSLEEP POWER UNIT field

The IN-STATE DEVSLEEP POWER UNIT field indicates the power units as defined in Table 124 represented in the TYPICAL IN-STATE DEVSLEEP POWER field.

Table 124 – Power units

Value	Power Units
0	1 milliwatts units
1	10 milliwatts units
2	100 milliwatts units
3	1 watt units

13.7.11.5.14 TER SUPPORTED bit

If the TER SUPPORTED bit is set to one, then the device supports Transitional Energy Reporting.

13.7.11.6 TRANSITIONAL ENERGY REPORTING EXTENDED

13.7.11.6.1 TRANSITIONAL ENERGY REPORTING EXTENDED FIELD overview

See ACS-4 for descriptions of PM0: Active and PM2: Standby.

The TRANSITIONAL ENERGY REPORTING EXTENDED field is defined as follows:

a) the TRANSITIONAL ENERGY REPORTING EXTENDED field is valid only under typical device operating parameters (e.g., temperature); and
b) the TRANSITIONAL ENERGY REPORTING EXTENDED field only relates to the transition to the requested power states, described in this subclause, immediately preceded by 32, 1 MB random write requests.

13.7.11.6.2 PM0 TO PM2 RECOUP COST field

The PM0 TO PM2 RECOUP COST field indicates the nominal number of time units the device needs to remain in the PM2: Standby state in order to recoup the energy consumed by transitioning to the PM2: Standby state from the PM0: Active state, relative to operating in the PM0: Active state (see Figure 404). If the PM0 TO PM2 RECOUP COST field is cleared to zero, then the device does not support reporting the PM0: Active to PM2: Standby transitional energy. The amount of energy consumed during the transition to the PM2:Active state is measured from issue of the STANDBY IMMEDIATE command until the device completes transition to the PM2: Standby state. If the PM0 TO PM2 RECOUP COST field is set to all ones, then transitioning to the PM2: Standby interface power state from the PM0: Active interface power state has no power advantage.

NOTE 71 - The accuracy of the nominal value of the PM0 TO PM2 RECOUP COST field may change over time.

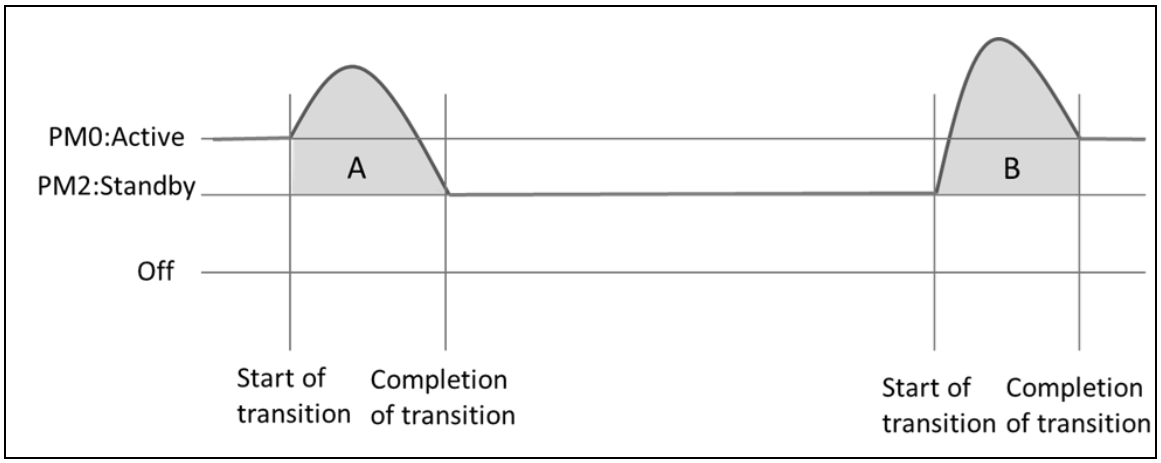


Figure 404 – Transitional Energy, PM0: Active to PM2: Standby, and PM2: Standby to PM0: Active

13.7.11.6.3 PM0 TO PM2 TIME UNIT field

The PM0 TO PM2 TIME UNIT field indicates the time units as defined in Table 122 represented in the PM0 TO PM2 RECOUP COST field.

13.7.11.6.4 PM2 TO PM0 RECOUP COST field

The PM2 TO PM0 RECOUP COST field indicates the nominal recommended number of time units the device needs to remain in the PM2: Standby state in order to recoup the energy consumed by transitioning to the PM0: Active state from the PM2: Standby state, relative to operating in the PM0: Active state. If the PM2 TO PM0 RECOUP COST field is cleared to zero, then the device does not support reporting the PM2: Standby to PM0: Active transitional energy. The amount of energy consumed during the transition to the PM0: Active is measured from any command issued that causes the device to exit PM2:Active until PM0: Active.

NOTE 72 - The accuracy of the nominal value of the PM2 TO PM0 RECOUP COST field may change over time.

13.7.11.6.5 PM2 TO PM0 TIME UNIT field

The PM2 TO PM0 TIME UNIT field indicates the time units as defined in Table 122 represented in the PM2 TO PM0 RECOUP COST field.

13.7.11.6.6 DEVSLEEP/PM2 TO OFF RECOUP COST field

The DEVSLEEP/PM2 TO OFF RECOUP COST field indicates the nominal recommended number of time units the device needs to remain in the Power Off condition in order to recoup the energy consumed by transitioning to the Power Off state from the DevSleep interface power state/PM2: Standby state, relative to operating in the DevSleep Interface Power state/PM2: Standby state (see Figure 405). If the DEVSLEEP/PM2 TO OFF RECOUP COST field is cleared to zero, then the device does not support reporting the DevSleep/PM2: Standby to Power Off transitional energy. The amount of energy consumed during the transition to the Power Off condition is measured from the exit of the DevSleep Interface Power State, transitioning to PM0: Active, transitioning to PM2: Standby, and then transitioning to Power Off. If the DEVSLEEP/PM2 TO OFF RECOUP COST field is set to all ones,

then transitioning to the Power Off state from the DevSleep interface power state/PM2: Standby state has no power advantage.

NOTE 73 - The accuracy of the nominal value of the DEVSLEEP/PM2 TO OFF RECOUP COST field may change over time.

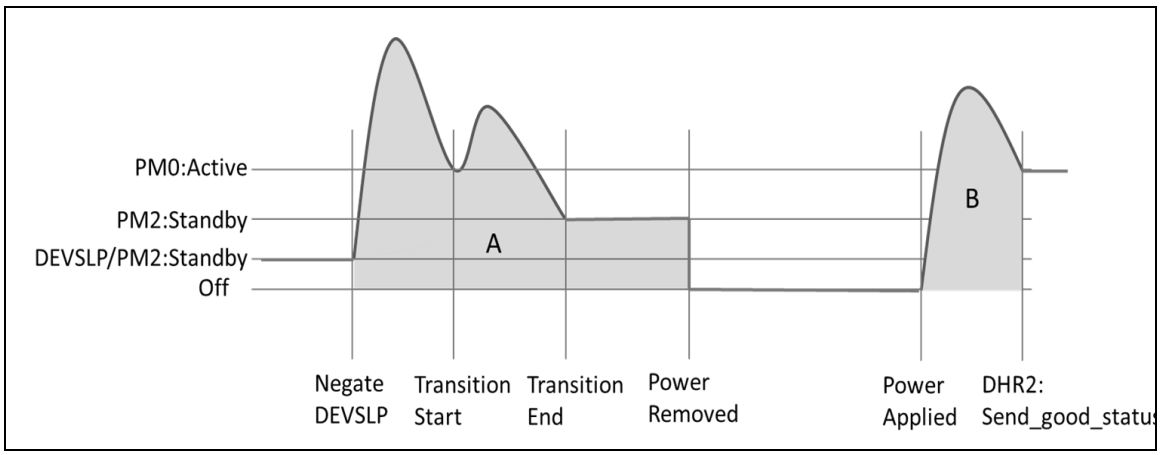


Figure 405 – Transitional Energy, DevSleep/PM2: Standby to PM0: Active, and Off to DHR2:Send_good_status

13.7.11.6.7 DEVSLEEP/PM2 TO OFF TIME UNIT field

The DEVSLEEP/PM2 TO OFF TIME UNIT field indicates the time units as defined in Table 122 represented in the DEVSLEEP/PM2 TO OFF RECOUP COST field.

13.7.11.6.8 OFF TO GOOD STS RELATIVE TO DEVSLEEP/PM2 RECOUP COST field

The OFF TO GOOD STS RELATIVE TO DEVSLEEP/PM2 RECOUP COST field indicates the nominal recommended number of time units the device needs to remain in the Power Off condition in order to recoup the energy consumed by transitioning to the PM0: Active state from the Off state, relative to operating in the DevSleep Interface Power state/PM2: Standby state. If the OFF TO GOOD STS RELATIVE TO DEVSLEEP/PM2 RECOUP COST field is cleared to zero, then the device does not support reporting the recoup cost of exiting Power Off to DHR2: Send_good_status relative to DevSleep/PM2: Standby. The amount of energy consumed during the transition to the PM0: Active state is measured from Power On, transitioning to PM0: Active, and exiting DHR2: Send_good_status, relative to operating in the DevSleep/PM2: Standby state.

13.7.11.6.9 OFF TO GOOD STS RELATIVE TO DEVSLEEP/PM2 TIME UNIT field

The OFF TO GOOD STS RELATIVE TO DEVSLEEP/PM2 TIME UNIT field indicates the time units as defined in Table 122 represented in the OFF TO GOOD STS RELATIVE TO DEVSLEEP/PM2 RECOUP COST field.

13.7.11.6.10 PM2 TO PM0 LATENCY field

The PM2 TO PM0 LATENCY field indicates the typical number time units to transition from PM2: Standby to PM0: Active.

13.7.11.6.11 PM2 TO PM0 LATENCY TIME UNIT field

The PM2 TO PM0 LATENCY TIME UNIT field indicates the time units as defined in Table 122 represented in the PM2 TO PM0 LATENCY field.

13.7.11.6.12 TYPICAL IN-STATE SLUMBER POWER field

The TYPICAL IN-STATE SLUMBER POWER field indicates the typical number of in-state power units consumed while the device is in the Slumber interface power state.

13.7.11.6.13 IN-STATE SLUMBER POWER UNITS field

The IN-STATE SLUMBER POWER UNITS field indicates the power units as defined in Table 124 represented in the TYPICAL IN-STATE SLUMBER POWER field.

13.7.11.6.14 TERE SUPPORTED bit

If the TERE SUPPORTED bit is set to one, then the device supports Transitional Energy Reporting Extended Qword.

13.8 Asynchronous notification (optional)

13.8.1 Asynchronous notification overview

Asynchronous notification is a mechanism for a device to send a notification to the host that the device requires attention.

EXAMPLE - A few examples of how this mechanism is able to be used include indicating media has been inserted in an ATAPI device or indicating that a hot plug event has occurred on a Port Multiplier port.

This definition does not list all events that cause a device to generate an asynchronous notification. The mechanism that the host uses to determine the event and the action that is required is outside the scope of this specification, refer to the command set specification for the specific device for more information.

13.8.2 Set Device Bits FIS Notification (N) bit

The Set Device Bits FIS Notification (N) bit is used by devices to notify the host that attention is needed. The N bit is set to one by a device if the device needs attention, otherwise it is cleared to zero.

By default, a device shall not set the N bit to one in the Set Device Bits FIS. The device shall have the Asynchronous Notification feature enabled by the host before the device may set the N bit to one in the Set Device Bits FIS. After receiving a Set Device Bits FIS with the N bit set to one, it is the responsibility of the host to interrogate the device and determine what type of action is needed.

13.8.3 Notification mechanism

To indicate that the device needs attention, the device shall issue a Set Devices Bits FIS to the host with the Interrupt (I) bit set to one and the Notification (N) bit set to one. The Error, Status Hi, and Status Lo fields shall accurately reflect the current values of the corresponding register fields in the device.

Reception of a Set Device Bits FIS with the N bit set to one may be reflected to the host using the SNotification register, as defined in 14.2.6. A host may support Asynchronous Notification without supporting the SNotification register. The requirement for a host to support Asynchronous Notification is that it may generate an interrupt if the Set Device Bits FIS is received. The SNotification register enables software to be sure that the interrupt was due to a notification event.

13.8.4 State machine for Asynchronous Notification

The following state machine (see Figure 406) defines the required behavior if the device supports Asynchronous Notification. A device shall only send a Set Device Bits FIS with the N bit set if it is in a state that may explicitly send a Set Device Bits FIS to the host as described by the Command

layer state machines. The state machine is entered from the Device_idle state as defined in 11.3. A device that supports asynchronous notification is also required to support the enhancements for Asynchronous Notification support in the Device_idle and Check_command states as defined in 11.3.

The state machine utilizes an internal variable called NotifyPending. The NotifyPending variable indicates that an asynchronous notification has been sent to the host. If NotifyPending is cleared to zero and an event occurs that requires attention, an asynchronous notification may be sent to the host. If NotifyPending is set to one and an event occurs that requires attention, an asynchronous notification shall not be sent to the host since the host has not yet acknowledged reception of the last asynchronous notification received. The host acknowledges reception of an asynchronous notification by sending a Register Host to Device FIS.

AN0: Notify_host	Transmit Set Device Bits FIS to host that has I bit set to one, N bit set to one, Status and Error values and all Reserved fields cleared to zero. Set NotifyPending to one.	
1. Unconditional	→	DI0: Device_idle

Figure 406 – Asynchronous notification state machine

AN0: Notify_host, this state is entered if the asynchronous notification feature is enabled and an asynchronous notification needs to be sent to the host.

If in this state, the device shall issue a Set Device Bits FIS to the host that has the Interrupt (I) bit set to one, the Notification (N) bit set to one, the current STATUS field cleared to zero, the ERROR field cleared to zero, and all Reserved fields cleared to zero. The internal variable NotifyPending shall be set to one to indicate that a notification has been sent to the host that has not yet been acknowledged by the host through the reception of a Register Host to Device FIS.

Transition AN0:1, the device shall unconditionally transition to the state DI0: Device_idle state.

13.8.5 ATAPI notification

13.8.5.1 ATAPI notification overview

An ATAPI device shall indicate whether it supports asynchronous notification in Word 78 of IDENTIFY PACKET DEVICE, according to 13.2.3. The feature is enabled by using SET FEATURES, as defined in 13.3.6.

13.8.5.2 Event example (informative)

An example of an event that may cause the device to generate an asynchronous notification to the host to request attention is the Media Change Event. The Media Change Event occurs if an ATAPI device has detected a change in device state – either media has been inserted or removed.

13.9 Phy event counters (optional)

13.9.1 Phy event counters overview

Phy event counters are an optional feature to obtain more information about Phy level events that occur on the interface. This information may aid designers and integrators in testing and evaluating the quality of the interface. A device indicates whether it supports the Phy event counter feature in IDENTIFY (PACKET) DEVICE Word 76, bit 10. The host determines the current values of Phy event counters by reading the Phy Event Counters log (see 13.9.4). The counter values shall not

be retained across power cycles. The counter values shall be preserved across COMRESET and software resets.

The counters defined are grouped into three basic categories:

a) those that count events that occur during Data FIS transfers;
b) those that count events that occur during non-Data FIS transfers; and
c) events that are unrelated to FIS transfers.

Counters related to events that occur during FIS transfers may count events related to host-to-device FIS transfers, device-to-host FIS transfers, or bi-directional FIS transfers. A counter that records bi-directional events is not required to be the sum of the counters that record the same events that occur on device-to-host FIS transfers and host-to-device FIS transfers.

Implementations that support Phy event counters shall implement all mandatory counters, and may support any of the optional counters as shown in Figure 407.

NOTE 74 - Note that some counters may increment differently based on the speed that non-Data FIS retries are performed by the host and device.

Implementations may record CRC and non-CRC error events differently.

EXAMPLE 1 - There is a strong likelihood that a disparity error may cause a CRC error. Thus, the disparity error may cause both the event counter that records non-CRC events and the event counter that records CRC events to be incremented for the same event.

EXAMPLE 2 - Another example implementation difference is how a missing EOF_P event is recorded; a missing EOF_P may imply a bad CRC even though the CRC on the FIS may be correct.

These examples illustrate that some Phy event counters are sensitive to the implementation of the counters themselves, and thus these implementation sensitive counters are unable to be used as an absolute measure of interface quality between different implementations.

Devices supporting Phy Event Counters shall implement, and report support for, the general purpose logging feature set as defined in the ACS-4 standard. In addition, the device shall implement the Phy Event Counters Log.

13.9.2 Counter reset mechanisms

There are two mechanisms that the host may explicitly cause the Phy counters to be reset. The first mechanism is to issue a BIST Activate FIS to the device. Upon reception of a BIST Activate FIS the device shall reset all Phy event counters to their reset value. In addition, if the host reads the Phy Event Counters log and bit 0 in the FEATURES field (7:0) is set to one, the device shall return the current counter values for the command and then reset all Phy event counter values.

13.9.3 Counter identifiers

13.9.3.1 Counter identifiers overview

Each counter begins with a 16 bit identifier. Figure 407 defines the counter value for each identifier. Any unused counter slots in the log should have a counter identifier value of 0h. Optional counters that are not implemented shall not be returned in the Phy Event Counter Log. A value of 0 returned for a counter means that there have been no instances of that particular event. There is no required ordering for event counters within the log (i.e., the order is arbitrary and selected by the device vendor).

Bits 14:12 of the counter identifier convey the number of significant bits that counter uses. All counter values consume a multiple of 16 bits.

The valid values for bits 14:12 and the corresponding counter sizes are:

a) 1h 16 bit counter;
b) 2h 32 bit counter;
c) 3h 48 bit counter; and
d) 4h 64 bit counter.

Any counter that has an identifier with bit 15 set to one is vendor specific. This creates a vendor specific range of counter identifiers from 8000h to FFFFh. Vendor specific counters shall observe the number of significant bits 14:12 as defined above.

Identifier (Bits 11:0)	Mandatory/ Optional	Description
000h	Mandatory	No counter value; marks end of counters in the log
001h	Mandatory	Command failed and the ICRC bit set to one in Error register
002h	Optional	R_ERR$_P$ response for Data FIS
003h	Optional	R_ERR$_P$ response for Device-to-Host Data FIS
004h	Optional	R_ERR$_P$ response for Host-to-Device Data FIS
005h	Optional	R_ERR$_P$ response for non-Data FIS
006h	Optional	R_ERR$_P$ response for Device-to-Host non-Data FIS
007h	Optional	R_ERR$_P$ response for Host-to-Device non-Data FIS
008h	Optional	Device-to-Host non-Data FIS retries
009h	Optional	Transitions from drive PHYRDY to drive PHYRDYn
00Ah	Mandatory	Register Device to Host FISes sent due to a COMRESET (see 11.2)
00Bh	Optional	CRC errors within a host-to-device FIS
00Dh	Optional	Non-CRC errors within a host-to-device FIS
00Fh	Optional	R_ERR$_P$ response for Host-to-Device Data FIS due to CRC errors
010h	Optional	R_ERR$_P$ response for Host-to-Device Data FIS due to non-CRC errors
012h	Optional	R_ERR$_P$ response for Host-to-Device non-Data FIS due to CRC errors
013h	Optional	R_ERR$_P$ response for Host-to-Device non-Data FIS due to non-CRC errors
C00h	Optional	(Port Multiplier) Host-to-Device non-Data FIS R_ERR$_P$ ending status due to collision
C01h	Optional	(Port Multiplier) Signature Register – Device-to-Host FISes
C02h	Optional	(Port Multiplier) Corrupt CRC propagation of Device-to-Host FISes

Figure 407 – Phy Event Counter identifiers

FISes that are terminated due to reception of SYNC$_P$ primitives before the end of the FIS (a SYNC Escape) are not counted in the R_ERR$_P$ ending status counters.

13.9.3.2 Counter definitions

The counter definitions in this section specify the events that a particular counter identifier represents.

13.9.3.2.1 Identifier 000h

There is no counter associated with identifier 000h. A counter identifier of 000h indicates that there are no additional counters in the log.

13.9.3.2.2 Identifier 001h

The counter with identifier 001h returns the number of commands that returned an ending status with the ERR bit set to one in the Status register and the ICRC bit set to one in the Error register.

13.9.3.2.3 Identifier 002h

The counter with identifier 002h returns the sum of (the number of transmitted Device-to-Host Data FISes that the host responded with R_ERR$_P$) and (the number of received Host-to-Device Data FISes that the device responded with R_ERR$_P$). The count returned for identifier 002h is not required to be equal to the sum of the counters with identifiers 003h and 004h.

13.9.3.2.4 Identifier 003h

The counter with identifier 003h returns the number of transmitted Device-to-Host Data FISes that the host responded with R_ERR$_P$.

13.9.3.2.5 Identifier 004h

The counter with identifier 004h returns the number of received Host-to-Device Data FISes that the device responded with R_ERR$_P$. The count returned for identifier 004h is not required to be equal to the sum of the counters with identifiers 00Fh and 010h.

13.9.3.2.6 Identifier 005h

The counter with identifier 005h returns the sum of (the number of transmitted Device-to-Host non-Data FISes that the host responded with R_ERR$_P$) and (the number of received Host-to-Device non-Data FISes that the device responded with R_ERR$_P$). Retries of non-Data FISes are included in this count.

13.9.3.2.7 Identifier 006h

The counter with identifier 006h returns the number of transmitted Device-to-Host non-Data FISes that the host responded with R_ERR$_P$. Retries of non-Data FISes are included in this count.

13.9.3.2.8 Identifier 007h

The counter with identifier 007h returns the number of received Host-to-Device non-Data FISes that the device responded with R_ERR$_P$. Retries of non-Data FISes are included in this count.

13.9.3.2.9 Identifier 008h

The counter with identifier 008h returns the number of transmitted Device-to-Host non-Data FISes that were retried after the host responded with R_ERR$_P$.

13.9.3.2.10 Identifier 009h

The counter with identifier 009h returns the number of times the device transitioned into the PHYRDYn state from the PHYRDY state, including but not limited to asynchronous signal events, power management events, and COMRESET events. If interface power management is enabled, then this counter may be incremented due to interface power management transitions.

13.9.3.2.11 Identifier 00Ah

The counter with identifier 00Ah returns the number of transmitted Register Device to Host FISes with the device reset signature in response to a COMRESET that were successfully followed by an R_OK$_P$ from the host.

13.9.3.2.12 Identifier 00Bh

The counter with identifier 00Bh returns the number of received Host-to-Device FISes of all types (Data and non-Data) that the device responded with R_ERR$_P$ due to CRC error. The count returned for identifier 00Bh is not required to be equal to the sum of the counters with identifiers 00Fh and 012h.

13.9.3.2.13 Identifier 00Dh

The counter with identifier 00Dh returns the number of received Host-to-Device FISes of all types (Data and non-Data) that the devices responded with R_ERR$_P$ for reasons other than CRC error. The count returned for identifier 00Dh is not required to be equal to the sum of the counters with identifiers 010h and 013h.

13.9.3.2.14 Identifier 00Fh

The counter with identifier 00Fh returns the number of received Host-to-Device Data FISes that the device responded with R_ERR$_P$ due to CRC error.

13.9.3.2.15 Identifier 010h

The counter with identifier 010h returns the number of received Host-to-Device Data FISes that the device responded with R_ERR$_P$ for reasons other than CRC error.

13.9.3.2.16 Identifier 012h

The counter with identifier 012h returns the number of received Host-to-Device non-Data FISes that the device responded with R_ERR$_P$ due to CRC error.

13.9.3.2.17 Identifier 013h

The counter with identifier 013h returns the number of received Host-to-Device non-Data FISes that the device responded with R_ERR$_P$ for reasons other than CRC error.

13.9.4 Phy Event Counters log (11h)

The Phy Event Counters log is one page (i.e., 512 bytes) in length. The first Dword of the log contains information that applies to the rest of the log. Software should continue to process counters until a counter identifier with value 0h is found or the entire log has been read. A counter identifier with value 0h indicates that the log contains no more counter values past that point. Log 11h is defined in Figure 408.

If IDENTIFY DEVICE data Word 76 bit 15 is set to one, the Phy Event Counters may be read using either of the READ LOG EXT or READ LOG DMA EXT commands.

If IDENTIFY DEVICE data Word 76 bit 15 is cleared to zero, the Queued Error log shall be read using the READ LOG EXT command. An attempt to read the Phy Event Counters log using the READ LOG DMA EXT command shall be aborted and the state of the device shall not change.

Byte	7	6	5	4	3	2	1	0
0	Reserved							
1	Reserved							
2	Reserved							
3	Reserved							
4	COUNTER 0 IDENTIFIER							
5								
6	COUNTER 0 VALUE							
5 + COUNTER 0 LENGTH								
…	…							
n	COUNTER z IDENTIFIER							
n + 1								
n + 2	COUNTER z VALUE							
n + 1 + COUNTER z LENGTH								
…	…							
508	Reserved							
509								
510								
511	DATA STRUCTURE CHECKSUM(7:0)							

Figure 408 – Phy event counters log data structure definition

Field Definitions

COUNTER z IDENTIFIER

Phy event counter identifier that corresponds to the COUNTER z VALUE field. Specifies the particular event counter that is being reported. The Identifier is 16 bits in length. Valid identifiers are listed in Figure 407.

COUNTER z VALUE

Value of the Phy event counter that corresponds to the COUNTER z IDENTIFIER field. The number of significant bits is determined by the COUNTER z IDENTIFIER field, bits 14:12 (see 13.9.3). The length of the COUNTER z VALUE field shall be a multiple of 16 bits. The counter shall stop (and not wrap to zero) after reaching its maximum value.

COUNTER z LENGTH

Size of the Phy event counter as defined by bits 14:12 of the COUNTER z IDENTIFIER field. The size of the Phy event counter shall be a multiple of 16 bits.

DATA STRUCTURE CHECKSUM

The DATA STRUCTURE CHECKSUM field value is the 2's complement of the sum of the first 511 bytes in the data structure. Each byte shall be added with unsigned arithmetic and overflow shall be ignored. The sum of all 512 bytes of the data structure is zero if the checksum is correct.

13.10 Hardware Feature Control (optional)

In Serial ATA Revision 3.0 and previous specifications, a Hardware Feature Control pin is defined only for these uses:

a) Disable Staggered Spinup (i.e., DSS) (see 6.13.1.2 and 13.11); and
b) Activity indication LED (i.e., DAS) (see 6.13.1.2 and 13.15).

This specification defines additional uses for the Hardware Feature Control pin(s) (see 6.13). Table 125 specifies the pins used by Hardware Feature Control for various connectors.

Table 125 – Pin(s) used by Hardware Feature Control

Standard Connector (3.5 inch and 2.5 inch)	1.8 inch Micro SATA Connector [a]	LIF-SATA Connector
Pin P11: a) DSS; b) DAS [b]; c) DHU [b]; or d) other vendor specific.	Pin P7: a) DAS [b]; b) DHU [b]; or c) other vendor specific.	Pin P8: a) DSS; b) DAS [b]; c) DHU [b]; or d) other vendor specific. Pin P21: a) DHU.
[a] DSS is not defined for 1.8 inch Micro SATA Connector. [b] Concurrent support of both DAS and DHU on the same pin is not permitted.		

If the extended uses of the Hardware Feature Control pin(s) are supported, then:

a) IDENTIFY DEVICE data Word 78 bit 5 (see 13.2.2.19) shall be set to one;
b) the SET FEATURES Enable Hardware Feature Control subcommand shall be supported (see 13.3.9);
c) page 08h of the Identify Device Data log (see 13.7.11) shall be supported;
d) on processing a power on reset, then:
   A) IDENTIFY DEVICE data Word 79 bit 5 (see 13.2.2.20) shall be cleared to zero;
   B) the CURRENT HARDWARE FEATURE CONTROL IDENTIFIER field (see 13.7.11.3.13) in the Identify Device Data log shall be cleared to zero; and
   C) see 6.13.1.2 for requirements of the default uses of the Hardware Feature Control pin(s);

and

e) after processing a SET FEATURES Enable Hardware Feature Control subcommand with no error, then:
   A) IDENTIFY DEVICE data Word 79 bit 5 (see 13.2.2.20) shall be set to one;
   B) the CURRENT HARDWARE FEATURE CONTROL IDENTIFIER field (see 13.7.11.3.13) in the Identify Device Data log shall be non-zero;
   C) the SUPPORTED HARDWARE FEATURE CONTROL IDENTIFIER field (see 13.7.11.3.14) in the Identify Device Data log shall be non-zero; and
   D) the behavior of the Hardware Feature Control pin(s) are specified by the SET FEATURES Enable Hardware Feature Control subcommand.

If the extended uses of the Hardware Feature Control pin(s) are not supported, then:

a) IDENTIFY DEVICE data Word 79 bit 5 (see 13.2.2.20) shall be cleared to zero;
b) the SET FEATURES Enable Hardware Feature Control subcommand shall not be supported (see 13.3.9);
c) the SUPPORTED HARDWARE FEATURE CONTROL IDENTIFIER field (see 13.7.11.3.14) in the Identify Device Data log shall be cleared to zero;
d) the CURRENT HARDWARE FEATURE CONTROL IDENTIFIER field (see 13.7.11.3.13) in the Identify Device Data log shall be cleared to zero; and
e) see 6.13.1.2 for requirements of the default uses of the Hardware Feature Control pin(s).

13.11 Staggered spinup (optional)

Storage subsystems that include numerous Serial ATA hard disk drives are presented with power system design issues related to the current load presented during system power-up. It is desirable to provide a simple mechanism that the storage subsystem controller(s) may sequence disk device initialization and spinup.

> NOTE 75 - Note that Serial ATA disk drive vendors may not always provide the capability to parse or process ATA commands prior to spinning up a device and completing device initialization, therefore this mechanism may not rely on the ATA protocol.

In order to accommodate staggered spinup of an array of disk drives in an enclosure, disk drives shall not spinup until after successful Phy initialization (i.e., after the Phy enters the DP7:DR_Ready state). Any of a number of methods may be used by the disk drive to defer spinup prior to Phy initialization and to maintain correct interface status during device initialization.

Storage subsystem controllers may employ a variety of methods to sequence Phy initialization across their plurality of Serial ATA ports, including but not limited to staged release of chip-level resets of host side Serial ATA transceivers, or embedded advanced power management logic.

System implementations should comprehend the various scenarios that may require power management, and the corresponding Phy initialization sequences.

EXAMPLE - Upon power up of a populated storage subsystem, Phy communication is initiated with a COMRESET signal generated by the host side transceiver.

The use of the term “host side transceiver” here refers to the Serial ATA interface located on the storage subsystem controllers. This contrasts with sequences associated with hot plugging of a Serial ATA disk drive into an operational storage subsystem, wherein COMINIT signals generated by disk side transceivers initiate Phy communications. In both of these cases, COMRESET or COMINIT signals are followed by exchange of COMWAKE signals. It is the successful entry into the DP7:DR_Ready state that gates disk drive spinup.

The device shall not use the staggered spinup mechanism to cause the device to spinup in cases where the device was spun down using an ATA command. See ACS-4 for more information.

13.12 Non-512 byte sector size (informative)

The Serial ATA interface has no inherent sector size dependency and there is nothing in this specification that precludes sector sizes other than 512 bytes.

Regardless of the physical sector size of storage devices, the maximum Data FIS payload length is 2 048 Dword as defined in 10.5.12. This may imply that either the number of physical sectors that are encompassed by a single Data FIS is reduced for devices with larger sector sizes or that Data FISes convey a non-integer number of sectors of data being transferred.

13.13 Defect management (informative)

13.13.1 Defect management overview

Storage subsystems that have been based on SCSI disk drives have evolved processes to address disk drive failure and mitigate effects of disk defects.

EXAMPLE - SCSI commands (e.g., READ DEFECT DATA (37h) and REASSIGN BLOCKS (07h)) have been used to allow storage administrators or applications to proactively address problematic sectors on a disk surface.

Serial ATA devices leverage the economies of the desktop device market, and as a consequence inherit both the design philosophy and implementation of desktop devices. From a design philosophy perspective, desktop devices have never yielded the low-level of control, (e.g., reassignment of logical sectors that is commonplace in enterprise-class devices). Instead, desktop devices have been positioned as a opaque data repository. This approach has many benefits, (e.g., elimination of defect management) as a design task for the computer (O/S, file system, I/O, device driver) designer, at the expense in some cases of deterministic device performance and awareness of defect management activity in general.

As opaque data providers, Serial ATA device vendors assume the bulk of the responsibility for defect management and data availability. This section provides a high-level overview of approaches that Serial ATA device vendors employ in these areas, and of the tools that are available to system designers, storage management application designers, and system administrators to maximize storage and data dependability.

Subsystem designers make no provisions for reassignment of blocks; rather reassignment of defective blocks is performed automatically if indicated by desktop-class devices. Additionally, disk device manufacturers provide a spectrum of tools and embedded features that help prevent occurrence of non-recoverable read errors, that help detect disk device degradation that may cause catastrophic loss of data, and that help administrators diagnose and isolate the cause of error events.

13.13.2 Typical Serial ATA reliability metrics

Various methods are employed by disk manufacturers to recover bad bits in a block of data. First and foremost, extensive error correcting codes (ECC) are used “on-the-fly” to detect and correct errors without impacting device performance. Second, various read-retry schemes may be used if ECC fails to correct an error on the fly. Only after retry processes are exhausted are errors posted to the host.

Design recommendation, accommodate inherent non-recoverable read error rate through RAID schemes appropriate for the target market’s reliability needs.

13.13.3 An overview of Serial ATA defect management

Defects that cause non-recoverable read errors come in two distinct flavors, temporary and permanent. Various schemes are employed by device manufacturers to determine the nature of a defect. If a defect is determined to be of a permanent nature, device firmware re-maps the LBA to a predefined spare sector, and marks the defective sector as such. Subsequent accesses to the re-mapped LBA are directed to the spare sector in a fashion transparent to the host. As part of the re-mapping process, the device preserves the error state of the remapped block until such time as it is written with new data.

The number of spare sectors configured in a disk drive is generally unique across different device vendors and device models and reflects tradeoffs between device capacity and expected defect rates. Generally, spares are associated with defined allocation groups in a disk drive. If the number of spare sectors becomes exhausted over time, subsequent permanent defects result in sectors that are unable to be re-mapped. The affected Logical Block Addresses (LBAs) are then recognized as bad by the host operating system or disk utilities (e.g., scandisk), and are subsequently not used.

Design recommendation, if a host or a Serial ATA storage subsystem encounters a non-recoverable read error, that error is managed by the disk drive. The disk drive is responsible for performing extensive read retry processes prior to communicating an error condition, exercising internal device diagnostics to determine the nature of the error, and re-map the physical sector if required. Subsequent accesses to that logical address are directed to a known good region on the disk.

Care should be taken to assure that spares are not exhausted if write caching is enabled. If this care is not taken, there is a possibility that a write operation may not actually complete successfully and return the appropriate error condition to the host in a timely fashion. Refer to the information on Self-Monitoring, Analysis, and Reporting Technology (SMART) according to 13.13.5 for monitoring spares status.

In especially critical data applications, specific queries of disk drive logs may be performed to determine whether the error event is a random event, or if there exists some degrading condition that warrants additional system or administrator action.

In storage subsystems where known good data may be recovered from redundant sources (e.g., in a RAID subsystem), it is recommended that known good data be written back to any LBA that a read error is reported. This behavior ensures that the disk drive has an opportunity to remedy the error condition and write known good data to known good blocks on disk, whether those be remapped blocks resulting from a permanent error condition, or the same blocks that may have been affected by an error of a temporary nature. Subsequent read accesses are assured of being satisfied without error.

13.13.4 Continuous background defect scanning (CBDS)

Serial ATA device vendors may employ schemes called continuous background defect scanning (CBDS), where during idle periods, firmware routines are used to scan sectors on the disk to look for and correct defects. CBDS therefore processes as a background task. The effect of CBDS is to reduce the occurrence of non-recoverable read errors by proactively “cleaning” defects from good sectors or copying data from suspect regions on disk to spare sectors. CBDS is described in more detail in the SMART specification that also provides a means for enabling/disabling the capability.

Design recommendation, consider CBDS requirements if evaluating system power-saving schemes that may power-off disk drives during apparent periods of inactivity.

13.13.5 Self-monitoring, analysis and reporting technology

Some disk failures are predictable. Such failure mechanisms are characterized by degradation over time, and in some cases may be effectively monitored by the disk drive and logged for periodic reporting to storage managers or management utilities.

The ACS-4 standard describes SMART command support in detail. Individual device vendors provide unique SMART capabilities on their devices and documentation on those capabilities so that host or controller developers may effectively use predictive failure information.

Design recommendation, use capabilities provided through ATA SMART commands to predict disk failure if possible. Predictive knowledge may be used to swap in spare disk drives in RAID configurations, to trigger data backup or protection routines, or to schedule storage subsystem maintenance.

13.14 Enclosure services/management (optional)

13.14.1 Enclosure services/management overview

A means for providing support for industry-standard SCSI Accessed Fault-Tolerant Enclosures (SAF-TE) and SCSI Enclosure Services (SES) enclosure services is provided in order to improve the functionality of Serial ATA storage subsystems. No modification to Serial ATA devices contained within an enclosure is required.

A storage enclosure processor (SEP) is a device that interfaces with various sensors and indicators within a storage enclosure subsystem. A Serial ATA Enclosure Management Bridge (SEMB) is a device that allows a Serial ATA host controller or Port Multiplier to communicate with a SEP, acting as a bridge between the host and the enclosure processor. This section defines a standard interface for software to communicate with SEMB devices via the ATA Command Block Registers. It also defines how SEMB devices communicate with SEP devices using the I^2C IPMI protocol. To allow software to communicate effectively with the SEMB, the SEMB is viewed as another Serial ATA device connected to the storage subsystem. The SEMB may be implemented within a Serial ATA host controller, within a Port Multiplier, or as a hardware bridge between a host and the SEP.

In this specification, the host (the Serial ATA RAID controller or HBA) may use either the SAF-TE or SES command protocol to communicate control/status with the SEP. These protocols provide the necessary features and have the advantage of being well known and widely implemented, and should therefore minimize the impact on RAID controller firmware and host management software. The implementation requires that Serial ATA host controllers or Port Multipliers that support enclosure management have an I^2C interface to communicate with the SEP device.

This specification also addresses the need to support a generic and standardized interface that allows application software from different vendors to communicate with the management device. This is implemented by having the appropriate commands be sent/received using the ATA Command Block Register set using the READ SEP/WRITE SEP commands associated with this interface. The ATA Command Block Register interface allows for consistency with the other devices in the subsystem as well as being simple to use and well understood.

13.14.2 Topology

13.14.2.1 Topology overview

The enclosure services support mechanisms should support the configurations and topologies expected for storage subsystems that require such enclosure services (e.g., external storage enclosures). Figure 409 illustrates a generic configuration.

The concentrator is controller logic that bridges a host interface to one or more Serial ATA devices. Typically a concentrator is able to be a RAID controller or a Port Multiplier, but it may be a simple HBA or part of an integrated multi-function chipset. Because a concentrator may be embodied differently depending on implementation, the generic term is used in order to avoid implying any particular implementation.

The host interface is the interface that the host communicates with the concentrator. For RAID controllers plugged into a PCI slot, the host interface is able to be PCI, while for external RAID controllers in a storage subsystem, the host interface may be Fibre Channel, InfiniBand Architecture, Ethernet (iSCSI), or any of a number of different external subsystem interconnects.

The SEMB is logic that bridges enclosure management data from a host interface to an enclosure management bus (indicated as I^2C in Figure 409). For an intelligent PCI RAID controller, the SEMB may be firmware running on the RAID processor and associated design-specific I^2C controller interface logic, while for a Port Multiplier, the SEMB is able to be controller logic that bridges the I^2C interface (and transactions) to Serial ATA via a logical ATA Command Block Register interface.

The SEP interfaces with the various sensors and indicators in the enclosure (e.g., temperature sensors, fan tachometers, and indicator LEDs).

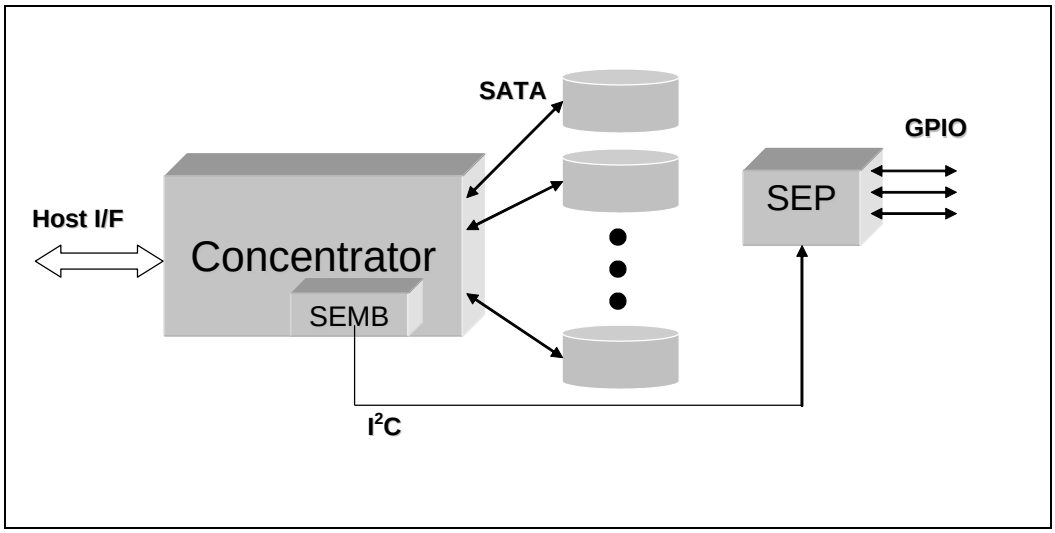


Figure 409 – Generic enclosure services topology

13.14.2.2 Definition configuration

The definition configuration is distilled from the generic topology in Figure 409 where the extraneous elements have been removed for the sake of definition clarity (see Figure 410). For the definition configuration, the host interface is selected as Serial ATA since this results in the maximum number of interfaces/elements being exposed for definition. In practice, the various elements may be integrated into another subsystem element (for instance, the SEMB may be integrated with the RAID controller or Port Multiplier). For the sake of definition, the elements are shown separately and the solution is presented as if it were a Serial ATA target device. In such a configuration, the SEP is being presented as merely another exposed Serial ATA device. This configuration is most analogous to existing SCSI enclosure services schemes where the enclosure services device is implemented using a SCSI target ID. For this configuration the enclosure services bridge and associated SEP are exposed as a single-purpose ATA device.

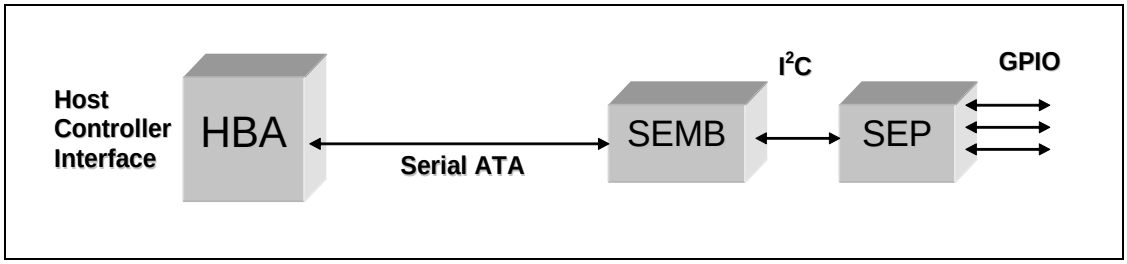


Figure 410 – Simplified view of generic topology

Since the HBA in the previous figure is a standard Serial ATA HBA and is therefore fixed, the elements being defined in this section may be further reduced to the definition configuration in Figure 411 that presents the enclosure services facility as an ATA device with a Command Block Register interface.

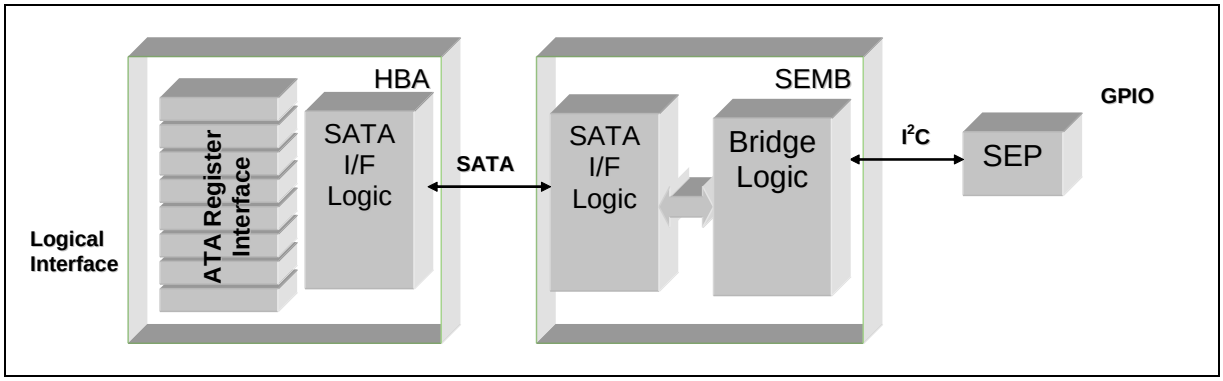


Figure 411 – Enclosure services definition configuration

For implementations where the SEP is not provided packaged with its front-end SEMB or equivalent (i.e., the SEP and SEMB are provided by different vendors), it is recommended that the interconnect between these two elements be I^2C and the command protocol for this interconnect as defined in 13.14.4.3. For implementations where the SEP is provided by the same supplier as the SEMB, the interconnect between these elements may be vendor-specific (and may be embedded for those cases where the two elements are integrated).

Similarly, for implementations where the SEMB is packaged with its front-end HBA or equivalent, the interconnect between these elements may be vendor specific and is not required to be Serial ATA (in some configurations, the SEMB may be integrated into the HBA in that case the interconnect between these two logical elements is embedded).

13.14.3 Limitations

In order to accommodate a range of possible implementations, the interfaces and elements defined in this section represent more than is able to typically be utilized by any one design/implementation. Only those interfaces that are exposed and interconnect ingredients from different vendors are expected to utilize the corresponding definitions and specifications described in this section.

EXAMPLE - An intelligent RAID controller may have the SEMB functionality implemented as firmware running on the embedded RAID processor, and such a solution be unable to expose the SEMB front-end interfacethat is able to be vendor-specific (i.e., internal to the RAID card and managed by the vendor-supplied firmware).

However, such a solution probably is able to expose the I^2C interface if it interconnects with another vendor's SEP, and is able to therefore need to comply with the specification for the communications between the SEMB and the SEP.

13.14.4 Definition

13.14.4.1 Definition overview

This specification supports the same enclosure management command/status as the SAF-TE protocol (plus addendums). Since SAF-TE is widely used in current SCSI applications, it is a natural path to try to reuse as much of the data format as possible. This specification also supports the SES enclosure management command/status as defined in the SCSI-3 Enclosure Services Command Set specification. The requirements given in clause 3 contain specific references.

For the case of the Serial ATA Register set to one, a subset of the existing ATA task file is used for creating the commands to send/receive data from the SEP device. Where possible, the same command structure as the existing ATA protocol has been used. All SEP commands are issued to the SEMB with the SEP_ATTN opcode in the Command register and the actual SEP command is

passed as a parameter in the FEATURES field (7:0). As defined in 13.14.4.4 and 13.14.4.5 the host-to-SEP and SEP-to-host command protocols.

13.14.4.2 Discovery

Following a COMRESET or software reset, the SEMB shall place the unique SEMB-specific signature as identified in Figure 412 into the logical Command Block Registers if the SEMB detects the presence of an attached SEP. The signature shall be available in the logical Command Block Registers no later than 3 s after a reset operation (whether power-on reset, COMRESET, or soft reset). For implementations where the SEMB is separate from the HBA, the Command Block Register signature shall be conveyed over the Serial ATA interconnect using a Register Device to Host FIS.

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	00h							
COUNT(7:0)	01h							
COUNT(15:8)	00h							
LBA(7:0)	01h							
LBA(15:8)	3Ch							
LBA(23:16)	C3h							
LBA(31:24)	00h							
LBA(39:32)	00h							
LBA(47:40)	00h							
DEVICE(7:0)	na							
STATUS(7:0)	BSY	DRDY	DF	DSC	DRQ	0	0	ERR

Status = 50h

Figure 412 – Register signature indicating presence of enclosure services device

If the SEMB does not detect the presence of an attached SEP, then the SEMB shall place the signature as identified in Figure 413 into the logical Command Block Registers in order to convey to the host that there is no device present at the logical interface, and the SEMB shall not respond to any subsequent Command Block Register accesses or issued commands until the next COMRESET or software reset. For implementations where the SEMB is separate from the HBA, the Command Block Register signature is conveyed over the Serial ATA interconnect using a Register Device to Host FIS.

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	FFh							
COUNT(7:0)	FFh							
COUNT(15:8)	FFh							
LBA(7:0)	FFh							
LBA(15:8)	FFh							
LBA(23:16)	FFh							
LBA(31:24)	FFh							
LBA(39:32)	FFh							
LBA(47:40)	FFh							
DEVICE(7:0)	FFh							
STATUS(7:0)	0	1	1	1	1	1	1	1

Status = 7Fh

Figure 413 – Register signature for absent enclosure processor

In addition to the device reset signature, SEPs shall support the IDENTIFY SEP command as defined in 13.14.5.1 in order to allow host software to determine its capabilities and to identify whether it supports the SAF-TE or SES command set.

13.14.4.3 Logical command block registers to I²C mapping

13.14.4.3.1 Logical command block registers to I²C mapping overview

The SEMB provides the mapping and translation from transactions between the SEP and SEMB, and the transactions presented to the host via the logical Command Block Register interface.

13.14.4.3.2 Command delivery

Commands are delivered to the SEP by the SEMB via the Command Block Register interface as a result of the Command or Device Control register being written. If the SEMB logical register interface is closely coupled to the host/HBA or in response to receipt of a Register Host to Device FIS (i.e., if the SEMB logical register interface is at the far end of a Serial ATA physical interconnect).

Commands are delivered to the SEP over I²C by the SEMB that extracts the SEP command and command type fields from the Command Block Registers (or received Register Host to Device FIS) and forwards the fields to the SEP. If the SEP is discrete and connected via an I²C interconnect, the SEP command fields are packaged as an I²C frame and forwarded to the SEP via the I²C interconnect. The SEMB logical Command Block Registers observes the same conventions as Serial ATA for handling of the BSY bit. The BSY bit is set by the interface/SEMB in response to the Command register being written, and the SEP later clears this bit to indicate command completion/status.

For issuing commands, the SEMB shall generate and transmit an I²C packet based on the contents of the Command Block Registers or Register Host to Device FIS. The mapping of Command Block Registers to transmitted I²C packet shall only be done for commands written using the SEP_ATTN opcode, and the SEMB need not respond to any other opcode in the Command register. The Command Block Registers mapping to I²C packet shall be as indicated in Figure 414.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	SEP_CMD(7:0)							
FEATURES(15:8)	Reserved							
COUNT(7:0)	LEN(7:0)							
COUNT(15:8)	Reserved							
LBA(7:0)	CMD_TYPE(7:0)							
LBA(15:8)	Reserved							
LBA(23:16)	Reserved							
LBA(31:24)	Reserved							
LBA(39:32)	Reserved							
LBA(47:40)	Reserved							
DEVICE(7:0)	Reserved			0	Reserved			
COMMAND(7:0)	SEP_ATTN (67h)							

Figure 414 – Command Block Register fields used in enclosure processor communications

Field Definitions

SEP_CMD The SAF-TE or SES command code to be issued in conjunction with the command type specified in the CMD_TYPE field.

SAF-TE READ BUFFER usage, the SEP_CMD field is equivalent to the BUFFER ID field of the SCSI READ BUFFER command. See 3.1 in the SAF-TE specification reference for the command codes and their functions.

SAF-TE WRITE BUFFER usage, the SEP_CMD field is equivalent OPERATION CODE field transferred in the parameter data of the SCSI WRITE BUFFER command. See 3.2 in the SAF-TE specification reference for the command codes and their functions.

SES RECEIVE DIAGNOSTIC RESULTS usage, the SEP_CMD field is equivalent to the PAGE CODE field of the SCSI RECEIVE DIAGNOSTIC RESULTS command. See 6.1 of the SES specification reference for the command codes and their functions.

SES SEND DIAGNOSTIC usage, the SEP_CMD field is equivalent to the PAGE CODE field transferred in the parameter data of the SCSI SEND DIAGNOSTIC command. See 6.1 of the SES specification reference for the command codes and their functions.

IDENTIFY SEP usage, the SEP_CMD field is equal to ECh.

LEN — The transfer length of the data transfer phase of the command in Dword units. Valid values are 1..255 (yielding a maximum transfer length of 1 020 bytes). Data transfers that are not a multiple of 4 bytes shall be padded by the transmitter with zeros to the next 4 byte (Dword) granularity.

CMD_TYPE — Flag indicating whether the issued SEP command is a SAF-TE command code or a SES command code and whether the data transfer direction is from SEP-to-host or host-to-SEP. The encoding of the field is as follows:

a) 00h SAF-TE command code with SEP-to-host data transfer, including IDENTIFY SEP;
b) 80h SAF-TE command code with host-to-SEP data transfer;
c) 02h SES command code with SEP-to-host data transfer, including IDENTIFY SEP; and
d) 82h SES command code with host-to-SEP data transfer.

All other values reserved

The resulting I^2C frame for delivering a command is as indicated in Figure 415.

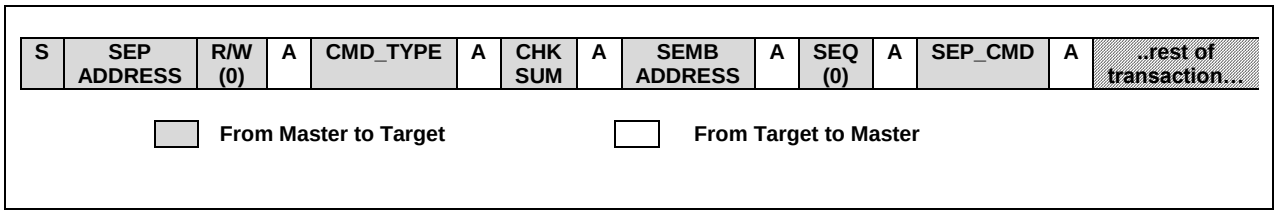


Figure 415 – I^2C frame for conveying an enclosure services command

The SEMB need not support any Command register writes with values other than SEP_ATTN. In response to a Command register write of a value other than SEP_ATTN, the SEMB shall set the ERR bit and clear the BSY bit in the Status register. In response to such illegal host behavior, the SEMB shall not generate any I^2C traffic for that illegal command. The SEMB shall support Device Control register writes where the state of the SRST bit changes, but shall take no action in response to Device Control register writes where SRST does not change state.

The SEP need not support both SAF-TE and SES command protocols. In response to a SEP command issues with a protocol not supported by the SEP, the SEP shall return with error status as defined in 13.14.4.3.3.

13.14.4.3.3 Status mechanism

The SEMB indicates command completion to the host by clearing the BSY bit to zero in the Status register and by triggering an interrupt. SEP status returns to the SEMB consist only of the Status byte and no other Command Block Registers are used to convey status. If the SEP has encountered some error condition or does not support the issued SEP command, then the ERR bit in the Status register is set to one.

If the SEP is communicating to the SEMB over an I^2C interconnect, then the status byte is included at the end of the transactions as indicated in the SEP read and write definitions that follow. Upon transferring the status value into the Status register, the SEMB shall clear the BSY bit to zero and signal an interrupt. If the SEMB is connected to the host via a Serial ATA interconnect, then the ending status shall be collected in a Register Device to Host FIS and transmitted to the host. In response to the read and write SEP commands, the SEP_CMD field and the CMD_TYPE field values are returned.

Upon successful completion of a command, the status value in the Status register shall be 50h. Upon an error condition, the status value shall be 51h.

13.14.4.4 Host-to-SEP data commands

All host-to-SEP data transfers transfer a data payload with length as indicated in the LEN field for the command. The SAF-TE and SES references define the commands and functions supported as well as the format of the transferred data structures.

All SEP commands are issued using the SEP_ATTN command (opcode 67h) in the Command register and the SEP command code in FEATURES field (7:0) as illustrated in the Command Block Registers image of Figure 416. The CMD_TYPE field identifies whether the issued SEP command is a read or a write and whether the command protocol is SAF-TE or SES.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	SEP_CMD(7:0)							
FEATURES(15:8)	Reserved							
COUNT(7:0)	LEN(7:0)							
COUNT(15:8)	Reserved							
LBA(7:0)	CMD_TYPE(80h or 82h)							
LBA(15:8)	Reserved							
LBA(23:16)	Reserved							
LBA(31:24)	Reserved							
LBA(39:32)	Reserved							
LBA(47:40)	Reserved							
DEVICE(7:0)	Reserved			0	Reserved			
COMMAND(7:0)	SEP_ATTN (67h)							

Figure 416 – WRITE SEP command block registers

Host-to-SEP data transfer commands shall be followed by a data transfer from the host to complete the SEP command delivery. If the command is delivered to the SEP over an I^2C interface, the transmitted I^2C packets shall be of the form indicated in Figure 417.

SEMB to SEP transfer – transfers both the command and data

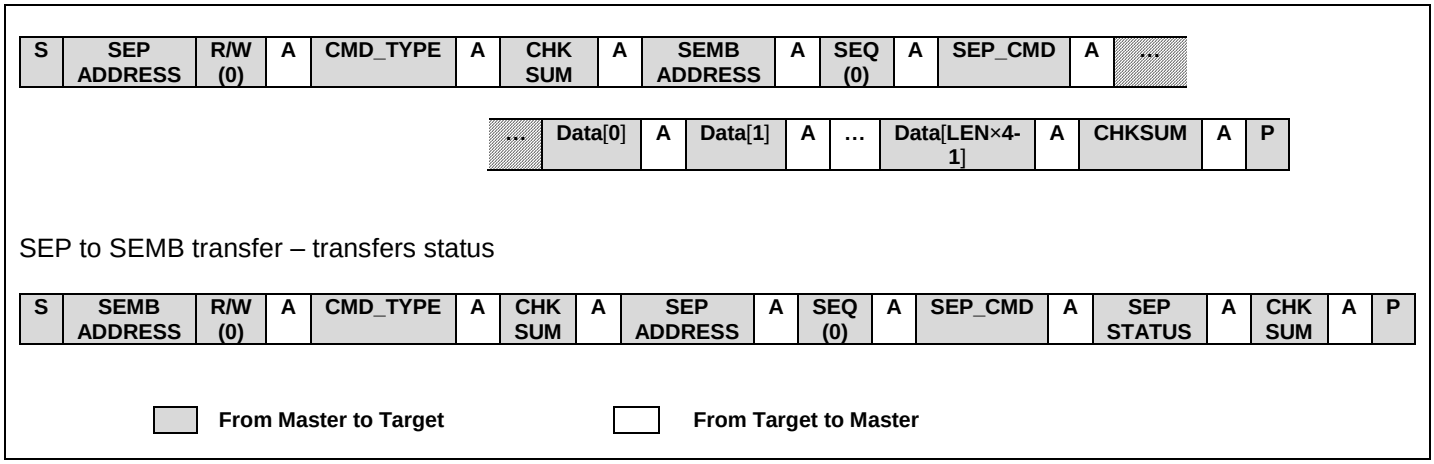


Figure 417 – I²C transactions corresponding to a WRITE SEP command

The I²C transport for Serial ATA enclosure service traffic shall be Master to Target (IPMB) compliant. See the IPMB and I²C references for details on Master to Target transport conventions for IPMB.

The host shall use the DMA protocol for transferring data from the host to the SEMB, and if the interface between the SEMB and the host is a Serial ATA interface, the SEMB shall trigger the DMA data transfer by transmitting a DMA Activate FIS to the host.

13.14.4.5 SEP-to-Host data commands

All SEP-to-Host data transfers transfer a data payload. The LEN field for the command indicates the length of the data payload being transferred. The SAF-TE and SES references define the commands and functions supported as well as the format of the transferred data structures. Both SAF-TE and SES SEPs shall support the IDENTIFY SEP command that has the same format for both command sets.

All SEP commands are issued using the SEP_ATTN command (opcode 67h) in the Command register and the SEP command code in the FEATURES field (7:0) as illustrated in the Command Block Registers image of Figure 418. The CMD_TYPE field identifies whether the issued SEP command is a read or a write and whether the command protocol is SAF-TE or SES.

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	SEP_CMD(7:0)							
FEATURES(15:8)	Reserved							
COUNT(7:0)	LEN(7:0)							
COUNT(15:8)	Reserved							
LBA(7:0)	CMD_TYPE (00h or 02h)							
LBA(15:8)	Reserved							
LBA(23:16)	Reserved							
LBA(31:24)	Reserved							
LBA(39:32)	Reserved							
LBA(47:40)	Reserved							
DEVICE(7:0)	Reserved			0	Reserved			
COMMAND(7:0)	SEP_ATTN (67h)							

Figure 418 – READ SEP command block registers

SEP-to-host data transfer commands shall be followed by a subsequent data transfer from the device to the host to complete the SEP command. If the command is delivered to the SEP over an I^2C interface, the transmitted I^2C packets shall be of the form indicated in Figure 419.

SEMB to SEP transfer – transfer the command

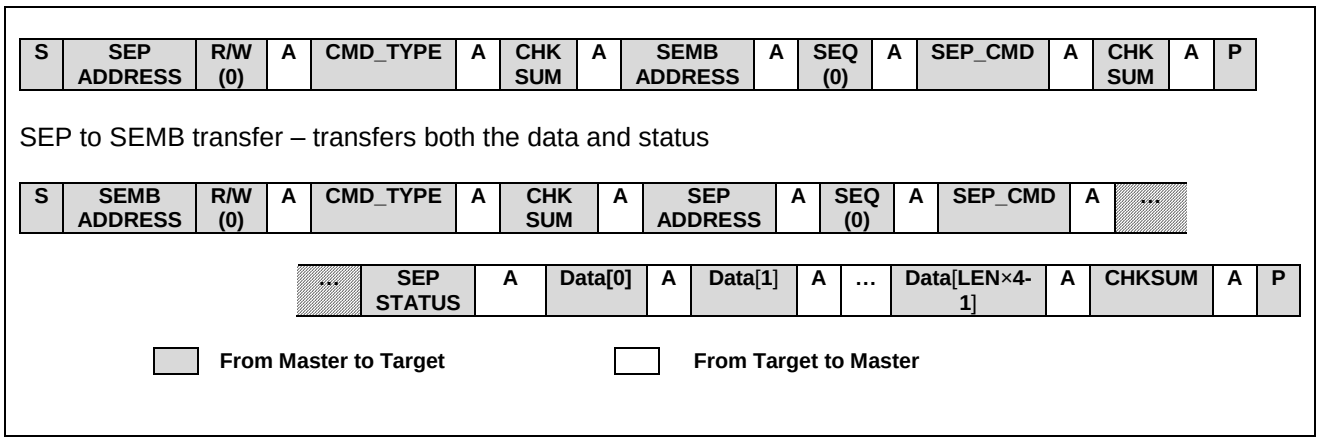


Figure 419 – I^2C transactions corresponding to READ SEP command

The host shall use the DMA protocol for transferring data from the SEMB to the host.

13.14.5 SES and SAF-TE extensions

13.14.5.1 IDENTIFY SEP command (ECh)

13.14.5.1.1 IDENTIFY SEP command definition (ECh)

Both SAF-TE and SES SEPs shall support the IDENTIFY SEP command as defined here. The IDENTIFY SEP command is a SEP-to-host command code used as the SEP command argument for the SEP_ATTN command with the CMD_TYPE field value set to either 00h or 02h depending on the command protocol being used (see 13.14.4.3.2). The command returns a data structure that describes the capabilities and attributes of the attached SEP. The IDENTIFY SEP command requests that the SEP return enclosure specific information (not device or environmental information). The command is roughly analogous to a SCSI INQUIRY command.

For parameters defined as a string of American Standard Code for Information Interchange (ASCII) characters, the ASCII data fields shall contain only graphic codes (i.e., code values 20h to 7Eh) and all strings shall be padded with space characters to the full width of the field. For the string ‘Copyright’, the character ‘C’ is the first byte, the character ‘o’ is the second byte, etc.

13.14.5.1.2 IDENTIFY SEP data structure

Figure 420 describes the data structure that is returned by the SEP in response to the IDENTIFY_SEP command. All reserved fields shall be cleared to zero. By setting the LEN field of the issued Read SEP command, the amount of returned data may be controlled (i.e., the transfer time may be reduced by not transferring the reserved and vendor specific bytes at the end of the data structure).

The IDENTIFY SEP data structure is normally 64 bytes long and may be extended in order to support a larger VENDOR SPECIFIC ENCLOSURE INFORMATION field.

The data structure does not include a list of elements in an enclosure.

This data block provides enclosure descriptor information and parameters.

Bytes	Field name
0	ENCLOSURE DESCRIPTOR LENGTH field
1	SUB-ENCLOSURE IDENTIFIER field
2..9	ENCLOSURE LOGICAL IDENTIFIER field
10..17	ENCLOSURE VENDOR IDENTIFICATION field
18..33	PRODUCT IDENTIFICATION field
34..37	PRODUCT REVISION LEVEL field
38	CHANNEL IDENTIFIER field
39..42	FIRMWARE REVISION LEVEL field
43..48	INTERFACE IDENTIFICATION STRING field
49..52	INTERFACE SPECIFICATION REVISION LEVEL field
53..63	VENDOR SPECIFIC ENCLOSURE INFORMATION field

Figure 420 – IDENTIFY SEP data structure definition

Field Definitions

ENCLOSURE DESCRIPTOR LENGTH

The ENCLOSURE DESCRIPTOR LENGTH field specifies the number of valid bytes contained in the IDENTIFY SEP data structure.

SUB-ENCLOSURE IDENTIFIER

As defined in the SES reference. Unless sub-enclosures are defined this field shall be cleared to zero.

ENCLOSURE LOGICAL IDENTIFIER

The ENCLOSURE LOGICAL IDENTIFIER field contains a unique logical identifier for the subenclosure. It shall use an 8 byte Network Address Authority (NAA) identifier, the format that is defined in SPC-4. The ENCLOSURE LOGICAL IDENTIFIER field is unique to the enclosure and may be different from the worldwide name of the device providing the enclosure services. The combination of this field, along with the ENCLOSURE VENDOR IDENTIFICATION field and the PRODUCT IDENTIFICATION field, uniquely identifies any SEP unit from any manufacturer. The worldwide name should have an NAA field of 5h, indicating that Institute of Electrical and Electronics Engineers (IEEE) is the naming authority.

ENCLOSURE VENDOR IDENTIFICATION

The ENCLOSURE VENDOR IDENTIFICATION field shall contain the identification string for the vendor of the enclosure in the same format as specified for the VENDOR IDENTIFICATION field of the standard SCSI INQUIRY data (see SPC-4). The ENCLOSURE VENDOR IDENTIFICATION field may be different from the vendor identification of the device providing the enclosure services.

PRODUCT IDENTIFICATION

The PRODUCT IDENTIFICATION field shall contain the product identification string for the enclosure in the same format as specified for the PRODUCT IDENTIFICATION field of the standard SCSI INQUIRY data (see SPC-4). The PRODUCT IDENTIFICATION field may be different from the product identification of the device providing the enclosure services.

PRODUCT REVISION LEVEL

The PRODUCT REVISION LEVEL field shall contain the product revision level string for the enclosure in the same format as specified for the PRODUCT REVISION LEVEL field of the standard SCSI INQUIRY data (see SPC-4). The PRODUCT REVISION LEVEL field may be different from the product revision level of the device providing the enclosure services.

CHANNEL IDENTIFIER

The CHANNEL IDENTIFIER field is used to distinguish between separate HBA channels supported by a single enclosure (i.e., through multiple/redundant host connections). The value in this field is unique for each channel. This field is optional and if not used shall be cleared to zero.

FIRMWARE REVISION LEVEL

The FIRMWARE REVISION LEVEL field is a 4 byte ASCII string that identifies the current firmware revision of the SEP device.

INTERFACE IDENTIFICATION STRING

The INTERFACE IDENTIFICATION STRING field is a 6 byte field that holds the constant ASCII string ‘SAF-TE’. If the command is issued with the SAF-TE protocol bit set to one in the CMD_TYPE field and the SEP supports this protocol or ‘S-E-S’, if the command is issued with the SES protocol bit set to one in the CMD_TYPE field and the SEP supports this protocol. This serves to identify that the enclosure is compliant with the command protocol indicated by the CMD_TYPE field used in issuing the IDENTIFY SEP command.

INTERFACE SPECIFICATION REVISION LEVEL

The INTERFACE SPECIFICATION REVISION LEVEL field is a 4 byte field that holds an ASCII string of the format ‘x.xx’ that identifies the revision of the Interface Specification that this SEP device claims compliance. ASCII string data is stored with the most significant (leftmost) character stored at the lowest byte offset of the field.

VENDOR SPECIFIC ENCLOSURE INFORMATION

The VENDOR SPECIFIC ENCLOSURE INFORMATION field is available for vendor specific definition and use.

13.14.5.2 Activity LED control

13.14.5.2.1 Activity LED control overview

The SES and SAF-TE protocols provide commands that are used to inform the SEP device of the state of each of its associated slots and the devices potentially inserted. This information is used to drive the enclosure status signals (LEDs, LCD, audible alarm, etc.) to some meaningful state, or to force the SEP to respond with a preprogrammed response as required; depending on the vendor's implementation.

Since Serial ATA devices do not necessarily provide an activity indication, extensions to the SAF-TE and SES facilities as defined in 13.14.5.2.2 and 13.14.5.2.3 to accommodate a means that an enclosure processor may be used to provide operator activity indication.

13.14.5.2.2 SAF-TE - Write Device Slot Status modification

The length of the valid data for the SAF-TE Write Device Slot Status depends on the number of device slots (d) on this channel. There are three bytes of data for each device slot on the channel and the associated data structure is defined in Figure 421.

Bit/Byte	7	6	5	4	3	2	1	0
0	SLOT 0 BYTE 0							
1	SLOT 0 BYTE 1							
2	SLOT 0 BYTE 2							
…								
(d×3)-3	SLOT d-1 BYTE 0							
(d×3)-2	SLOT d-1 BYTE 1							
(d×3)-1	SLOT d-1 BYTE 2							
d×3	Vendor Specific							
…63								

Figure 421 – SAF-TE write device slot status data structure

Field Definitions

SLOT d BYTE 0

As defined in SAF-TE.

SLOT d BYTE 1

Modified from definition in SAF-TE:

1. Bits 1:0 As defined in SAF-TE;
2. Bit 2 (DR_ACT) Set to one by the host to indicate device activity if issuing commands to the device associated with this slot. (Defined as Reserved in the SAF-TE reference); and
3. Bits 7:3 As defined in SAF-TE.

SLOT d BYTE 2

As defined in SAF-TE.

If no flags are set to one in any byte for a device slot this is a NO CHANGE FROM CURRENT STATE indication. This allows a host to change the state of one particular device slot without having to be aware of the current state of all device slots. Setting one or more flags requires that all flags

be written to the correct binary value. Thus, changes should be preceded by a read of the corresponding device slot status and the Write Device Slot Status implemented as a “read-modify-write” operation.

13.14.5.2.3 SES device element definition modification

The format of the CONTROL INFORMATION field for a device element type in the enclosure control page is defined in Figure 422. The data structure is modified with the addition of the device activity control (DR_ACT) bit to bit 7 of byte 2 (this bit is defined as Reserved in the SES reference).

Bits Bytes	7	6	5	4	3	2	1	0
0	COMMON CONTROL field (7:0)							
1	Reserved							
2	DR_ACT bit	DO NOT REMOVE bit	Reserved		RQST INSERT bit	RQST REMOVE bit	RQST IDENT bit	Res
3	Reserved		RQST FAULT bit	DEVICE OFF bit	ENABLE BYP A bit	ENABLE BYP B bit	Reserved	

Figure 422 – SES device element data structure

13.14.5.2.4 Activity indication behavior and operation

The host sets the DR_ACT bit to one to set the external DRIVE ACTIVITY LED to “on”. In response, the SEP shall blink the associated LED at a vendor-specific rate. The LED shall blink for a duration of approximately 0.5 s after that this bit shall be automatically cleared to zero by the SEP (and the LED stops blinking).

13.14.5.3 Slot–to–port correspondence

13.14.5.3.1 Slot–to–port correspondence overview

For storage subsystems that do not have a direct one-to-one correspondence between host connection and device slot, a correspondence convention is required in order to ensure the host has a means for accurately controlling the proper enclosure slot for a particular storage device. Configurations that do not have direct correspondence include those that have intervening elements between the host and the device that compromise direct correspondence (e.g., if Serial ATA Port Multipliers were used).

Figure 423 illustrates a configuration where there is no direct correspondence between host connection and enclosure device slot.

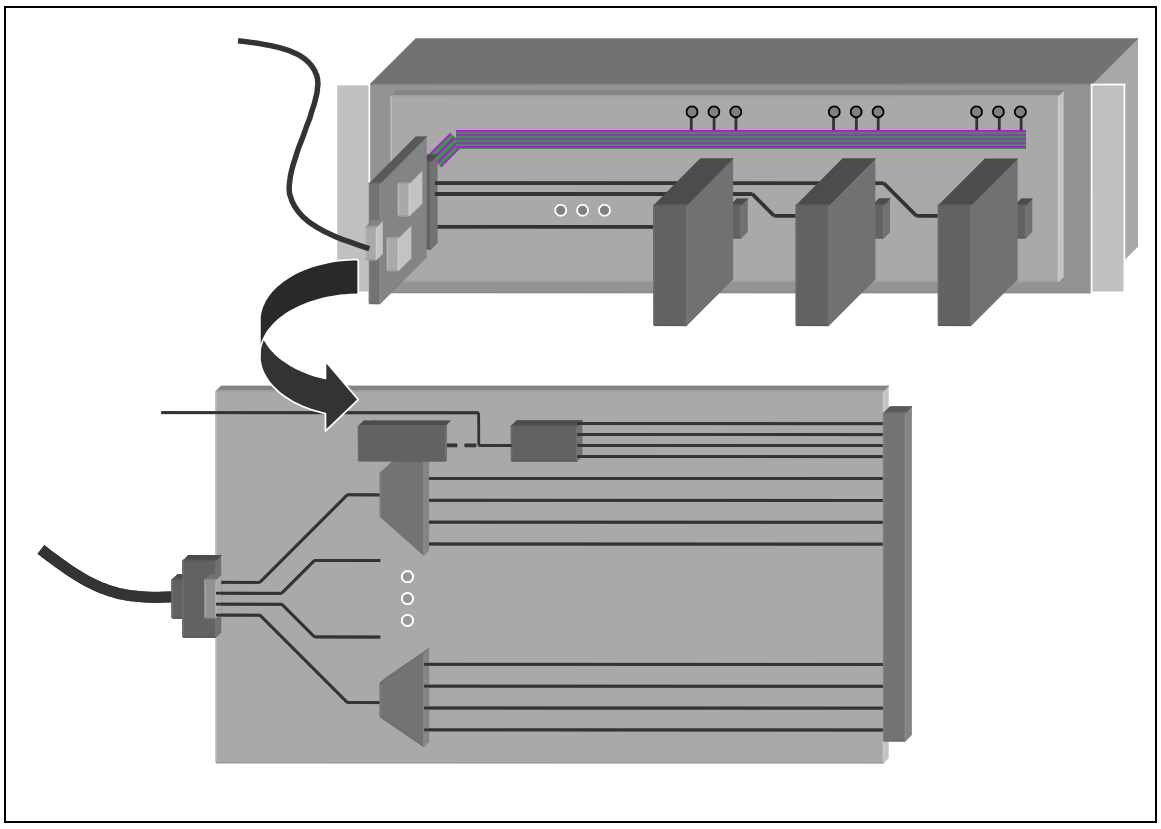


Figure 423 – Example subsystem

13.14.5.3.2 SAF-TE correspondence definition (optional)

Figure 424 defines the SAF-TE SCSI ID field convention for establishing correspondence between host connections and enclosure slots. SEPs that support SAF-TE may optionally adopt this convention. If an enclosure complies with this correspondence definition, it shall set bit 0 of byte 7 in the data returned for the Read Enclosure Configuration command to 1. If an enclosure does not comply with this correspondence definition, it shall clear bit 0 of byte 7 in the data returned for the Read Enclosure Configuration command to zero.

This correspondence convention, if implemented, shall be used for the SCSI ID in the Read Enclosure Status command where the SCSI ID for each device slot is returned to the host. The correspondence shall also be used by the host for the SCSI ID specified in the Set SCSI ID command. An empty or unused slot shall have a SCSI ID of FFh; a SCSI ID of FFh shall not be used to identify an active device slot.

Bits **Field**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
SCSI ID	PORT(3:0)				CHANNEL(3:0)			

Figure 424 – SAF-TE DEVICE ID field convention

Field Definitions

PORT The PORT field corresponds to a device connection. For a given enclosure slot, it indicates that port of an intervening multi-port controller the slot is associated

with. In the instance where the intervening controller is a Port Multiplier, the PORT field indicates that port of the Port Multiplier the slot is connected to and corresponds to the PM Port field used in the FIS to address the device. In the absence of an intervening multi-port controller, this field is zero.

CHANNEL The CHANNEL field corresponds to a host connection. For a given enclosure slot, it indicates that host connection the slot is associated with. In the instance where there is an intervening Port Multiplier, the CHANNEL field indicates that host channel the upstream port of the associated Port Multiplier is connected to. The CHANNEL field value is assigned serially starting at zero and is indicated on the host connections packaging accordingly.

For the configuration in Figure 423, the SCSI ID value corresponding to Slot 0 in the enclosure is possible to be 33h, while the SCSI ID value corresponding to Slot 3 in the enclosure is possible to be 03h.

13.14.5.3.3 SES correspondence definition (optional)

Figure 425 defines the SES SLOT ADDRESS field convention for establishing correspondence between host connections and enclosure slots. SEPs that support SES may optionally adopt this convention. The method for determining whether an enclosure complies with this correspondence definition is vendor specific or as defined by the SES-2 specification.

The SEP shall use this convention, if implemented, in the STATUS INFORMATION field for a device element type (element type 01h) in the enclosure status page. The first byte in this field for an element of device type is the Slot Address and shall be set as specified in Figure 425. The enclosure status page is read with the Receive Diagnostic Results command.

Bits **Field**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Slot Address	PORT(3:0)				CHANNEL(3:0)			

Figure 425 – SES SLOT ADDRESS field convention

Field Definitions

PORT The PORT field corresponds to a device connection. For a given enclosure slot, it indicates that port of an intervening multi-port controller the slot is associated with. In the instance where the intervening controller is a Port Multiplier, the PORT field indicates that port of the Port Multiplier the slot is connected to and corresponds to the PM Port field used in the FIS to address the device. In the absence of an intervening multi-port controller, this field is zero.

CHANNEL The CHANNEL field corresponds to a host connection. For a given enclosure slot, it indicates that host connection the slot is associated with. In the instance where there is an intervening Port Multiplier, the CHANNEL field indicates that host channel the upstream port of the associated Port Multiplier is connected to. The CHANNEL field value is assigned serially starting at zero and is indicated on the host connections packaging accordingly.

For the configuration in Figure 423, the Slot Address value corresponding to Slot 0 in the enclosure is possible to be 33h, while the Slot Address value corresponding to Slot 3 in the enclosure is possible to be 03h.

13.14.6 Enclosure services hardware interface

13.14.6.1 Enclosure services hardware interface overview

For implementations where the enclosure/backplane is not bundled with the storage subsystem controller, it is recommended that the out of band enclosure services interface used by both the enclosure and the controller be I^2C. As defined in 13.14.6.2 the connector and cable interconnect between the enclosure/backplane and the storage subsystem controller.

13.14.6.2 I^2C cable/connector definition

Implementations where the storage subsystem controller is not directly connected to the enclosure/backplane require an interconnect between the backplane and the controller for the I^2C enclosure services bus.

EXAMPLE - A self-contained system that uses a PCI-based Serial ATA RAID controller and houses a backplane with front-panel accessible devices, requires a means that the I^2C interface originating on the PCI controller is connected to the backplane that interfaces with the various storage devices and operator indicators (LEDs).

Products that utilize I^2C for enclosure services and desire interoperability/interchangeability with others' products shall use Molex Part # 22-43-6030 or equivalent as the bus connector. This connector has the same footprint/dimensions as the IPMB connector but its color is white instead of yellow as used by the actual IPMB connector.

13.14.6.3 SEP discovery and enumeration

For implementations where the SEP is an I^2C device and is not provided as part of a complete storage solution (i.e., is intended to communicate with a SEMB provided by another vendor), the SEP should have a dedicated I^2C interface with the SEMB and should not share the I^2C interface with enclosure sensors and indicators.

In order to enable the SEMB to efficiently discover and enumerate an attached SEP, the first SEP on an I^2C bus attached to a SEMB in a storage subsystem shall be assigned I^2C address C0h.

NOTE 76 - Note that the LSB of the I^2C ADDRESS field is the R/W bit effectively resulting in the I^2C ADDRESS field being 7 bits in length.

A SEMB directly supports only a single SEP, so use of more than one SEP attached to the same SEMB in a storage subsystem is vendor specific. Any additional SEP(s) on the same I^2C interface in a storage subsystem shall be assigned consecutively higher I^2C addresses.

13.15 HDD activity indication (optional)

13.15.1 HDD activity indication overview

Operator notification/indication of storage device status and activity may be driven by the host through the enclosure services facilities as defined in 13.14 or through other host-driven means.

Operator notification/indication of storage device status and activity may also be driven through an intelligent processor (e.g., an IO Processor). Reference the SAF-TE Write Device Slot Status command in the SAF-TE reference as defined in 13.14.5.2.2 for methods to support a SEP controlling HDD Activity Indication. As this reference suggests, these implementations may be vendor specific.

13.15.2 HDD activity emulation of desktop behavior

13.15.2.1 HDD activity emulation of desktop behavior overview

If the host controller optionally implements the desktop activity LED functionality, with the desired behavior to be compatible with current parallel ATA solutions, the host controller shall generate such a signal with the behavior defined in Figure 426.

```
//          POR - Power On Reset
//          HRESET - Hardware RESET
RESET = POR || HRESET;


if ((BSY || SActive) && DEVICE_TYPE != ATAPI)       // How !ATAPI determined is implementation specific
          ACTIVITY_LED = ON;
else
          ACTIVITY_LED = OFF;


if (DEV0_DEV1_EMULATION_ENABLED && DEV1_PRESENT)
{
          if (RESET)
                    DEV1_LED = ON;
}
else
          DEV1_LED = OFF;


If (REGISTER_FIS_TRANSMITTED_ON_DEV1_CHANNEL)
          DEV1_LED = OFF;


LED = ACTIVITY_LED || DEV1_LED;
```

Figure 426 – Activity LED definition for desktop behavior emulation

In the LED behavioral logic, the means that an implementation may determine that the attached device type is ATAPI (see logic expression DEVICE_TYPE != ATAPI in logic behavioral definition) is implementation specific, and may include detection of device type based on reset signature, detection of the command opcodes issued to the device (i.e., ATAPI devices have command issued using the ATAPI command codes of A0h and A1h), or through other means. The required behavior is that activity to ATAPI devices not generate LED activity indication.

For implementations that have multiple Serial ATA channels, the controller should provide an aggregate activity signal (i.e., the wired-OR of the individual activity signals from each channel).

13.15.2.2 Desktop HDD activity signal electrical requirements

For implementations that provide an activity signal in accordance with 13.15.2, the signal shall be active low and shall be of an open collector/drain design. The voltage and current requirements/capabilities of the signal is vendor-specific.

13.15.3 Activity/status indication reference (informative)

Serial ATA controller devices may include discrete physical pins for the purpose of connecting device activity LEDs. Such controllers may provide one device activity pin per port or an aggregate activity signal. They may be included so that the host (or IO Processor) software need not supply

a device activity status, or so that a SEP need not be burdened with blinking a LED if writes/reads to/from a device are occurring. These signals allow device activity LEDs without need for changes to this specification. It is likely that the methods outlined in this section may become obsolete as enclosure management facilities for Serial ATA mature and become readily available.

Figure 427 shows an example configuration for implementing device activity LEDs within an enclosure. In this example, it is likely that the solution uses the standard 0.1 inch headers that are common and widely available. Depending upon the number of devices, the standard header is possible to vary in size, but generally require two pins per device activity LED in the configuration. Thus, a 4-port solution with 4-device status LEDs requires a 2x4 pin standard header.

Figure 428 is identical in configuration to that shown in Figure 427, except that it shows the use of a ribbon cable for ease of assembly.

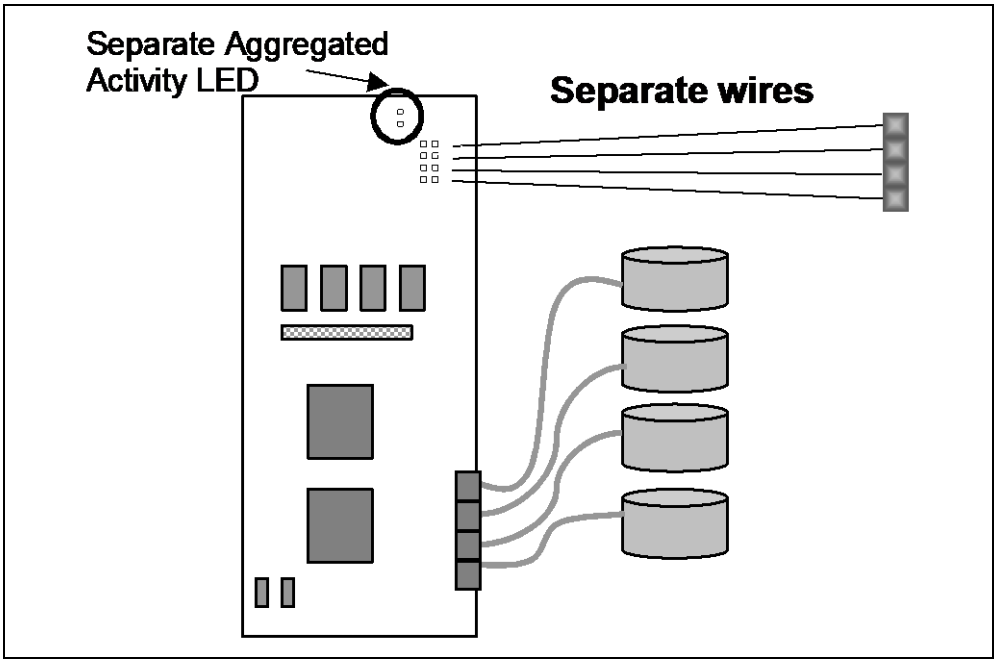


Figure 427 – Device activity LEDs with separate wires

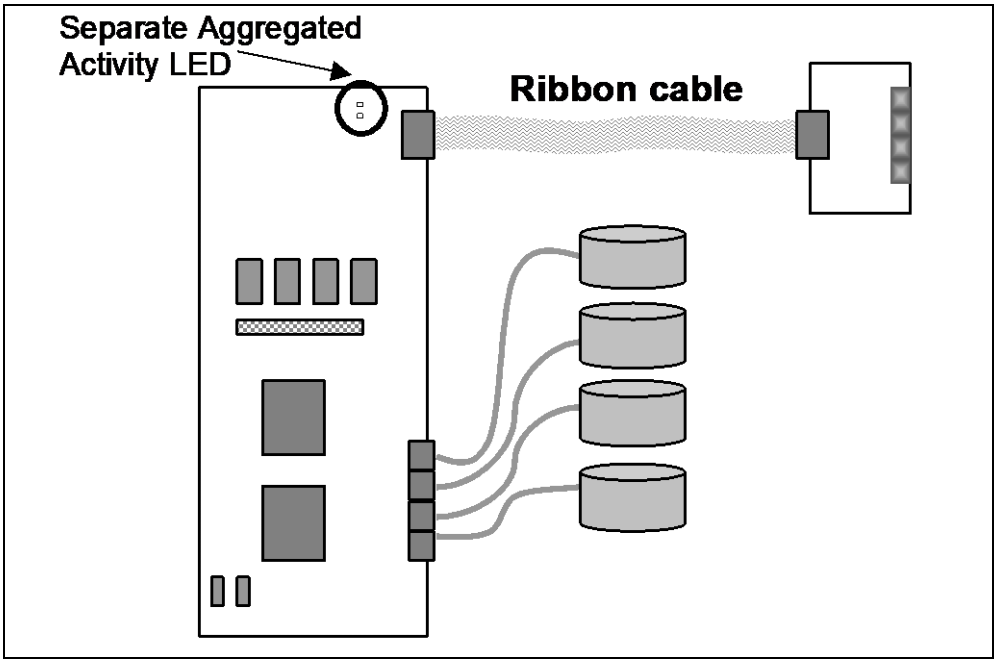


Figure 428 – Device activity LEDs with ribbon cable

NOTE 77 - Note also the activity outputs have been designed to support wired-OR configurations. They may then support a configuration where all the outputs are connected together for the purpose of generating an aggregate activity indication.

Taking advantage of the wired-OR functionality, Figure 429 shows a more complex configuration that is possible to support additional needs that may be required by an integrated system. In this example, the devices plug straight into a backplane. The device activity LEDs are placed on a separate, small mezzanine card that there is a possibility to wire-OR the LED activity signals from both the Serial ATA controller on the I/O Controller and the SEP that in this example is located on the backplane. This allows normal device activity to be indicated as described in the preceding paragraphs, but supports the concept of the SEP generating distinguishable visual patterns to the LEDs for other purposes.

EXAMPLE - An LED is able to be placed into a steady state ON by the SEP to indicate a failed card or a card that is targeted for hot plug.

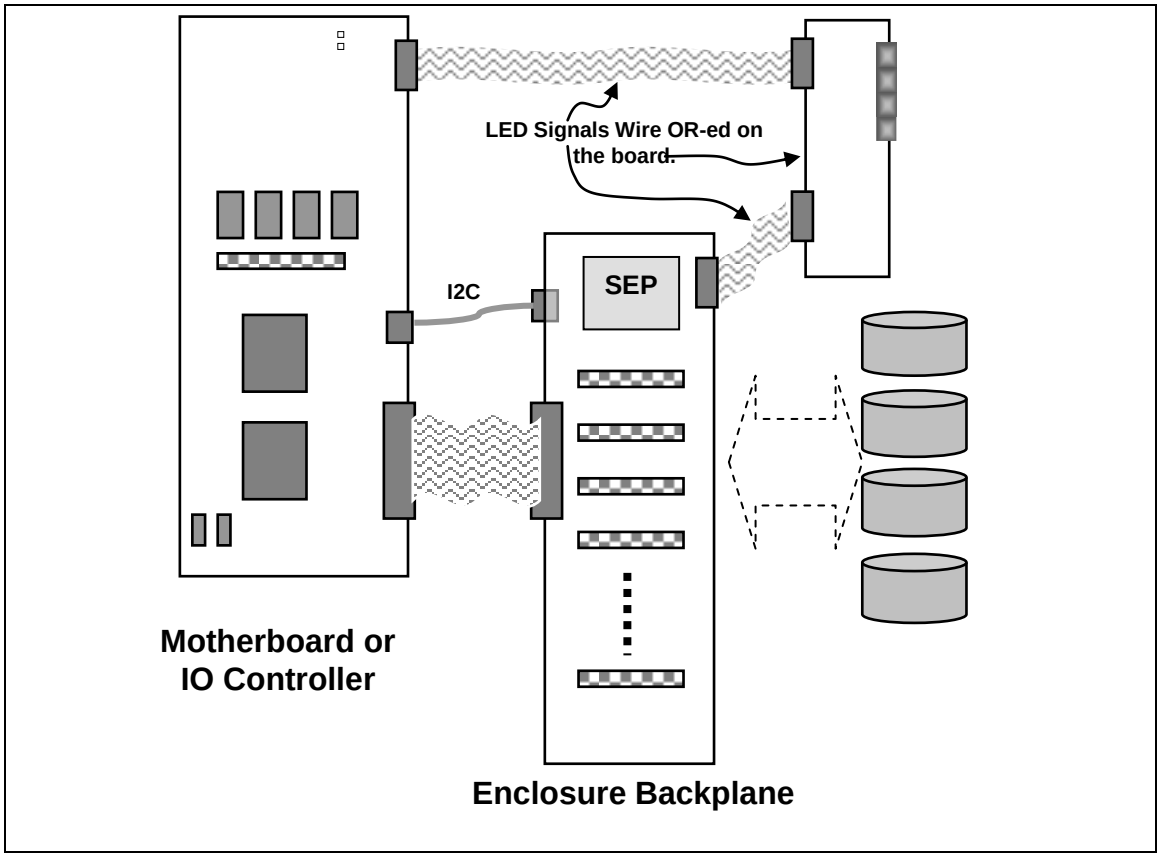


Figure 429 – Device activity LEDs in a storage subsystem

Designers should be aware of some of the limitations of the solutions described in this section. If a solution is using a Port Multiplier, (i.e., then the LED may only be useful for general front end port activity), and may not identify a specific device on a specific Port Multiplier port that an operator may be needing to take an action upon (e.g., device failed). Another consideration is that the signal generated by the Serial ATA controller may change states based upon certain states, and these state transitions may be too rapid to be readily detected by the operator.

EXAMPLE - The LED pin may be negated if the BSY bit is set to one or if the transport state machine is active.

Thus, in certain configurations where many short packets are being transmitted/received, the device may be active if the LED appears to be off if no additional methods are implemented to prevent this case from occurring.

13.16 Port Multiplier discovery and enumeration

13.16.1 Power-up

On power-up, the Port Multiplier shall enter state HPHP1: NoComm in the hot plug state machine for the host port, as defined in 16.3.3.5.2.
Upon entering this state, the Port Multiplier shall:

1) clear any internal state and reset all parts of the Port Multiplier hardware;
2) place the reset values in all Port Multiplier registers, including port specific registers. The reset values shall disable all device ports.

After performing this sequence, the Port Multiplier shall proceed with the actions described in the hot plug state machine for the host port.

If the Port Multiplier receives a COMRESET from the host before issuing the initial COMINIT signal to the host, the Port Multiplier shall immediately perform the actions as defined in 13.16.2.2 and cease performing the power-up sequence listed above.

13.16.2 Resets

13.16.2.1 Resets overview

There are three mechanisms to reset a device:

a) COMRESET;
b) software reset; and
c) the DEVICE RESET command for PACKET devices.

To reset a Port Multiplier, the host shall issue a COMRESET. A Port Multiplier does not reset in response to a software reset or a DEVICE RESET command. The specific actions that a Port Multiplier takes in response to each reset type is detailed in the following sections.

13.16.2.2 COMRESET

If the Port Multiplier receives a COMRESET over the host port, the Port Multiplier shall enter state HPHP1: NoComm in the hot plug state machine for the host port, as defined in 16.3.3.5.2.
Upon entering this state, the Port Multiplier shall:

1) clear any internal state and reset all parts of the Port Multiplier hardware;
2) place the reset values in all Port Multiplier registers, including port specific registers. The reset values shall disable all device ports.

After performing this sequence, the Port Multiplier shall proceed with the actions described in the hot plug state machine for the host port.

13.16.2.3 Software reset

If the host issues a software reset to the control port, two Register Host to Device FISes are sent to the control port as a result. In the first Register Host to Device FIS, the SRST bit in the Device Control register is set to one. In the second Register Host to Device FIS, the SRST bit in the Device Control register is cleared to zero.

Upon receiving the Register Host to Device FIS with the SRST bit asserted, the Port Multiplier shall wait for the Register Host to Device FIS that has the SRST bit cleared to zero before issuing a Register Device to Host FIS with the Port Multiplier signature to the host. The Port Multiplier's behavior shall be consistent with the Software reset protocol as defined in 11.4. The values to be placed in the Register Device to Host FIS are listed in Figure 430.

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	00h							
COUNT(7:0)	01h							
COUNT(15:8)	00h							
LBA(7:0)	01h							
LBA(15:8)	00h							
LBA(23:16)	69h							
LBA(31:24)	00h							
LBA(39:32)	96h							
LBA(47:40)	00h							
DEVICE(7:0)	na							
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	0	0	ERR

Figure 430 – Software reset to control port result values

Field Definitions

BSY 0

DRDY 1

DF 0

DRQ 0

ERR 0

The Port Multiplier shall take no reset actions based on the reception of a software reset. The only action that a Port Multiplier shall take is to respond with a Register Device to Host FIS that includes the Port Multiplier signature. To cause a general Port Multiplier reset, the COMRESET mechanism is used.

13.16.2.4 Device reset

A DEVICE RESET command issued to the control port shall be treated as an unsupported command by the Port Multiplier (see16.3.3.8.7).

13.16.3 Software initialization sequences (informative)

13.16.3.1 Software initialization sequences scope (informative)

This section details the sequences that host software should take to initialize a Port Multiplier device.

13.16.3.2 Port Multiplier aware software

Port Multiplier aware software checks the host's SStatus register to determine if a device is connected to the port. If a device is connected to the port, the host then issues a software reset to the control port. If the Port Multiplier signature is returned, then a Port Multiplier is attached to the port. Then the host proceeds with enumeration of devices on Port Multiplier ports as defined in 13.16.4.2.

Port Multiplier aware software shall not require a device to be present on device Port 0 in order to determine if a Port Multiplier is present.

13.16.3.3 Non-port multiplier aware software

Non-Port Multiplier aware software waits for the signature of the attached device to be returned to the host. If a device is present on device Port 0, the device connected to device Port 0 returns a Register Device to Host FIS that contains its signature. If a device is not present on device Port 0, non-Port Multiplier aware software times out waiting for the signature to be returned and assumes that a device failure has occurred. If fast boot is a requirement, the system should have a Port Multiplier aware BIOS and Port Multiplier aware OS driver.

If non-Port Multiplier aware software is loaded, all device ports other than device Port 0 are disabled. The host shall only receive FISes from the device attached to device Port 0.

13.16.3.4 Boot devices connected to Port Multiplier

System designers should only connect multiple boot devices to a Port Multiplier if the BIOS is Port Multiplier aware.

EXAMPLE - In a system that contains three bootable devices (hard drive, CD-ROM, and DVD) these devices are recommended to only be attached to the Port Multiplier if the BIOS is Port Multiplier aware or the user is unable to boot off of the devices that are not connected to device Port 0.

13.16.4 Port Multiplier discovery and device enumeration (informative)

13.16.4.1 Port Multiplier discovery

13.16.4.1.1 Port Multiplier signature

To determine if a Port Multiplier is present, the host performs the following procedure. The host determines if communication is established on the host's Serial ATA port by checking the host's SStatus register. If a device is present, the host issues a software reset with the PM Port field set to the control port. The host checks the signature value returned and if it corresponds to the Port Multiplier Signature, the host knows that a Port Multiplier is present. If the signature value does not correspond to a Port Multiplier, the host may proceed with the normal initialization sequence for that device type. The host shall not rely on a device being attached to device Port 0 to determine that a Port Multiplier is present.

If a Port Multiplier receives a software reset to the control port, the Port Multiplier shall issue a Register Device to Host FIS according to the procedure as defined in 13.16.2.3. The signature value contained in the Register Device to Host FIS is shown in Figure 431 .

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	na							
COUNT(7:0)	01h							
COUNT(15:8)	na							
LBA(7:0)	01h							
LBA(15:8)	69h							
LBA(23:16)	96h							
LBA(31:24)	na							
LBA(39:32)	na							
LBA(47:40)	na							
DEVICE(7:0)	na							
STATUS(7:0)	na							

Figure 431 – Port Multiplier signature

13.16.4.1.2 Considerations if Port Multiplier not present

Directly attached devices may not be prepared to receive a software reset from the host prior to transmission of the Signature FIS.

It is recommended that host software not issue software reset prior to successful reception of the Signature FIS by the host, unless the host is Port Multiplier aware.

13.16.4.2 Device enumeration

After discovering a Port Multiplier, the host enumerates all devices connected to the Port Multiplier. The host reads GSCR[2], as defined in 16.4.2.2, to determine the number of device ports on the Port Multiplier.

For each device port on the Port Multiplier, the host performs the following procedure to enumerate a device connected to that port:

1) the host enables the device port. The host enables a device port by setting the DET field appropriately in the device port's PSCR[2] (SControl) register, as defined in 14.2.4. The host uses the WRITE PORT MULTIPLIER command to write PSCR[2] (SControl) for the device port to be enabled;
2) the host should allow for communication to be established and device presence to be detected after enabling a device port. According to 8.4.2 that describes the host Phy initialization sequence;
3) the host reads PSCR[0] (SStatus) for the device port using the READ PORT MULTIPLIER command. If PSCR[0] (SStatus) indicates that a device is present, the host queries PSCR[1] (SError) for the device port and clears the X bit indicating device presence has changed; and
4) the signature generated by the device as a consequence of the initial COMRESET to the device port may be discarded if the host does not support context switching because the BSY bit may be clear if the Register Device to Host FIS is received by the host. Therefore, to determine the signature of the attached device, the host should issue a software reset to the device port. If a valid signature is returned for a recognized device, the host may then proceed with normal initialization for that device type.

Cascading Port Multipliers shall not be supported.

13.17 Automatic Partial to Slumber transitions

It is possible that the host and device may be independently aware of conditions where they may transition to the Slumber state to save power without impacting performance (e.g., idle, seeking).

To allow for increased power savings, Automatic Partial to Slumber Transitions defines a capability that allows the Phy to transition to Slumber from Partial without first entering Active.

The following are requirements of the Automatic Partial to Slumber Transitions capability:

a) Host Automatic Partial to Slumber shall only be used if the device reports host Automatic Partial to Slumber support (i.e., IDENTIFY DEVICE Word 76 bit 13); and
b) Device Automatic Partial to Slumber shall only be used if the host supports:
    A) an increased Partial exit latency (up to the max Slumber latency); and
    B) has enabled device Automatic Partial to Slumber transitions (via Set Features) on a device that has claimed device support (i.e., IDENTIFY DEVICE Word 76 bit 14).

Transitioning to Slumber from Partial does not require a COMWAKE followed by an Active state transition, and the respective attached Phy should not be aware of the transition. Since the respective Phy is in Partial it shall tolerate an increased Partial exit latency because the other Phy may be in Slumber.

13.18 Serial ATA Link power management support

Devices shall support host-initiated interface power management, device-initiated interface power management, or both.

13.19 DHU specific operation (optional)

The optional DHU feature of Hardware Feature Control (see 13.10) provides a method for the host to cause a device that has movable read/write heads to move them to a safe position.

DHU is an active high signal driven by the host. See 6.13.2 for electrical requirements for DHU.

If $V_{DHUactive}$ condition (see 6.13.2) is met, then:

1) the device shall stop read look-ahead if that operation is in process;
2) the device shall stop writing cached data to the media if that operation is in process;
3) if the device has a write buffer, then the device shall retain data in the write buffer; and
4) the device shall put itself in a state that minimizes or prevents damage due to a high-G event, (e.g., if a device implements unloading its head(s) onto a ramp, then the device shall retract the head(s) onto the ramp, and if a device implements parking its head(s) in a landing zone on the media, then the device shall park its head(s) in the landing zone).

If $VDHU_{negate}$ condition (see 6.13.2) is met, then the device shall perform normal operations.

13.20 Hybrid Information feature (optional)

13.20.1 Hybrid Information feature overview

A Solid State Hybrid Device (SSHD) contains both a primary medium (e.g., rotating magnetic) and a non-volatile caching medium (e.g., flash memory).

The Hybrid Information feature allows the host to provide information (e.g., hints) to the device that the device uses for various purposes (e.g., to decide that medium to save the user data to for optimal performance in retrieval).

SSHDs may determine data to cache based on observed accesses of LBAs and length of incoming requests. However, SSHDs do not possess host information to make the most optimal caching decisions (e.g., file type associated with the LBAs in the request). The Hybrid Information feature allows the host to be involved in making caching decisions to aid in optimizing cache utilization.

This feature provides a method for the host to indicate the caching priority of incoming requests to the device, and feedback to the host on how much non-volatile caching medium has been consumed at the various caching priority levels.

The use of the maximum hybrid priority is determined by the MAX PRIORITY BEHAVIOR bit (see 13.7.8.2.11) in the Hybrid Information log.

If the MAX PRIORITY BEHAVIOR bit is set to one, then the highest caching priority passed to the device instructs the SSHD that this data shall be placed and remain in the non-volatile caching medium until explicitly evicted by the host.

If the MAX PRIORITY BEHAVIOR bit is cleared to zero, then the highest caching priority passed to the device instructs the SSHD that this data should:

a) be placed in non-volatile caching medium; and
b) remain in the non-volatile caching medium.

Intermediate caching priority levels inform the device of the importance of the data being placed in the non-volatile caching medium, but makes no requirement for the device to place the logical sectors in the non-volatile caching medium (e.g., the host only imparts a relative caching level in relation to other requests). The device should make the best decision possible based on the caching priority provided by the host and other device knowledge (e.g., rotational position optimizations).

Having multiple caching priorities allows for the host to group data by importance. Data that is required to be in the non-volatile caching medium is passed with the highest priority level. The intermediate priorities may be used for data that would provide value to the user if present in the non-volatile caching medium, but is not required to meet power budgets or responsiveness criterion (e.g., medium files, application data, etc).

In order to maintain a given responsiveness for insertions into the non-volatile caching medium, the host may specify high and low dirty thresholds to ensure that the device does not consume too much internal bandwidth syncing logical sectors between the non-volatile caching medium and the primary medium, while still leaving room to absorb new writes.

The hints are attached to some commands sent by the host, included in the Register Host to Device FIS (see 10.5.5 and 13.20.2).

The host may evict logical sectors from the non-volatile caching medium using the HYBRID EVICT subcommand (see 13.6.8.7).

If the Hybrid Information feature is not supported (i.e., IDENTIFY DEVICE data Word 78 bit 9 is cleared to zero), then:

a) the device shall ignore hints as described in this subclause for any command; and
b) the device shall not support:
    A) the SET FEATURES Enable/Disable Hybrid Information subcommand (see 13.3.11);
    B) the Hybrid Information log (see 13.7.8);
    C) the HYBRID EVICT subcommand (see 13.6.8.7);
    D) the HYBRID DEMOTE BY SIZE subcommand (see 13.6.6.7);
    E) the HYBRID CHANGE BY LBA RANGE subcommand (see 13.6.6.5); and
    F) the HYBRID CONTROL subcommand (see 13.6.6.6).

If the Hybrid Information feature is supported (i.e., IDENTIFY DEVICE data Word 78 bit 9 is set to one), then:

a) the device shall support:
    A. the following features:
        a. NCQ Autosense (i.e., IDENTIFY DEVICE data Word 78 bit 7 shall be set to one); and
        b. POWER UP IN STANDBY (see ACS-4);
    B. the following SET FEATURES subcommands:
        a. Enable/disable the POWER UP IN STANDBY feature set (see ACS-4); and
        b. Enable/Disable Hybrid Information (see 13.3.11);
    C. the HYBRID DEMOTE BY SIZE subcommand (see 13.6.6.7);
    D. the HYBRID CONTROL subcommand (see 13.6.6.6); and
    E. the following logs:
        a. the NCQ Non-Data log (see 13.7.6);
        b. the NCQ Send and Receive log (see 13.7.7);
        c. the Identify Device Data log (see 13.7.11); and
        d. the Hybrid Information log (see 13.7.8);
b) the device should support the Device Sleep feature (i.e., IDENTIFY DEVICE data Word 78 bit 8 should be set to one);
c) the device shall not support the SET FEATURES POWER UP IN STANDBY feature set device spinup subcommand (see ACS-4);
d) the following commands shall be able to return command completion without error while in the PM2: Standby (see ACS-4) state or in the PM5: PUIS and spinup subcommand not supported state (see ACS-4):
    A. all read commands, if the requested logical sectors are in the non-volatile caching medium;
    B. all write commands, if the device stores all of the data for the command in the non-volatile caching medium;
    C. commands to read the following logs:
        a. Identify Device Data log;
        b. Hybrid Information log;
        c. Power Conditions log;
        d. Queued Error log;
        e. NCQ Non-Data log; and
        f. NCQ Send and Receive log;
    D. IDENTIFY DEVICE command;
    E. CHECK POWER MODE command;
    F. SMART RETURN STATUS command; and
    G. SECURITY UNLOCK command, if supported;

e) if the Hybrid Information feature is enabled (i.e., IDENTIFY DEVICE data Word 79 bit 9 is set to one), then:
    A. the device shall process the HYBRID INFORMATION field as described in this subclause for:
        a. the READ FPDMA QUEUED command (see 13.6.4);
        b. the WRITE FPDMA QUEUED command (see 13.6.5);
        c. the HYBRID DEMOTE BY SIZE subcommand (see 13.6.6.7);
        d. the HYBRID CHANGE BY LBA RANGE subcommand (see 13.6.6.5);
        e. the HYBRID CONTROL subcommand (see 13.6.6.6); and
        f. these non-NCQ commands:
            A. the READ DMA EXT command;
            B. the WRITE DMA EXT command; and
            C. the WRITE DMA FUA EXT command;

    and
    B. for all other commands, the device shall ignore the HYBRID INFORMATION field as described in this subclause;

and

f) if the Hybrid Information feature is disabled (i.e., IDENTIFY DEVICE data Word 79 bit 9 is cleared to zero), then the device shall ignore the HYBRID INFORMATION field as described in this subclause.

13.20.2 HYBRID INFORMATION field bits

13.20.2.1 HYBRID INFORMATION field bits overview

Figure 432 describes the mapping of the HYBRID INFORMATION field that is transported in the AUXILIARY field (23:16) of the Register Host to Device FIS (see 10.5.5).

If the Hybrid Information feature is not supported, then the device shall ignore the HYBRID PRIORITY field.

If:

a) the Hybrid Information feature is supported;
b) the Hybrid Information feature is enabled; and
c) the hybrid information is valid bit is set to one,

then the HYBRID PRIORITY field is valid.

If:

a) the Hybrid Information feature is supported; and
b) the Hybrid Information feature is not enabled or the HYBRID INFORMATION IS VALID bit is cleared to zero,

then the device shall ignore the HYBRID PRIORITY field.

AUXILIARY Field Bit	Description	Reference
16..19	HYBRID PRIORITY field	13.20.2.2
20	Reserved	
21	HYBRID INFORMATION IS VALID bit	
22..23	Reserved	

Figure 432 – HYBRID INFORMATION field

13.20.2.2 HYBRID PRIORITY field

13.20.2.2.1 HYBRID PRIORITY field overview

The HYBRID PRIORITY field indicates the hybrid priority to be associated with the logical sectors if processing a command that the HYBRID INFORMATION fields are valid (see 13.20.2).

If a command specifies a Hybrid Priority value that is greater than the MAXIMUM HYBRID PRIORITY LEVEL field (see 13.7.8.2.8), then the device shall return command aborted.

If the device processes a read command, then for each logical sector the device should read from the medium with the shortest latency that contains the most recent copy of the requested logical sectors.

13.20.2.2.2 HYBRID PRIORITY field is equal to MAXIMUM HYBRID PRIORITY LEVEL field

13.20.2.2.2.1 HYBRID PRIORITY field is equal to MAXIMUM HYBRID PRIORITY LEVEL field overview

The behavior of the highest permitted Hybrid Priority value (i.e., Maximum Hybrid Priority Level) is specified by the value of the MAX PRIORITY BEHAVIOR bit of the SUPPORTED OPTIONS field (see 13.7.8.2.11) in the Hybrid Information log.

13.20.2.2.2.2 MAX PRIORITY BEHAVIOR bit set to one

If:

a) the device processes a read command or a write command, with the HYBRID INFORMATION IS VALID bit set to one and the Hybrid Priority value set equal to the Maximum Hybrid Priority Level; and
b) the MAX PRIORITY BEHAVIOR bit is set to one,

then:

a) the device shall not evict logical sectors associated with the Maximum Hybrid Priority Level from the non-volatile caching medium in order to insert the logical sectors specified by the command;
b) the device may evict logical sectors associated with a lower hybrid priority from the non-volatile caching medium in order to insert the logical sectors specified by the command;
c) if:
    A) the logical sectors specified by the command are not currently in the non-volatile caching medium; and
    B) the device does not have capacity available for all of the requested logical sectors in the non-volatile caching medium,

    then the device shall:
    A) return command aborted; and
    B) in the Queued Error log:
        a) the SENSE KEY field shall be set to COMMAND ABORTED; and
        b) the additional sense code (i.e., the ADDITIONAL SENSE CODE field and the ADDITIONAL SENSE CODE QUALIFIER field) set to INSUFFICIENT RESOURCES;
d) copy the requested logical sectors to the non-volatile caching medium if the logical sectors specified by the command are not currently in the non-volatile caching medium and the non-volatile caching medium has capacity available for all of the requested logical sectors; and
e) associate the logical sectors specified by the command with the Hybrid Priority value supplied by the command and return command complete with no error.

13.20.2.2.2.3 MAX PRIORITY BEHAVIOR bit is cleared to zero and CACHE BEHAVIOR (CB) bit is cleared to zero

If:

a) the device processes a read command or a write command, with the HYBRID INFORMATION IS VALID bit set to one and the Hybrid Priority value set to the Maximum Hybrid Priority Level;
b) the MAX PRIORITY BEHAVIOR bit is cleared to zero; and
c) the CB bit is cleared to zero,

then:

a) the device may evict any logical sectors from the non-volatile caching medium in order to insert the logical sectors specified by the command;
b) the device should put the requested logical sectors in the non-volatile caching medium; and
c) associate the logical sectors that were put in the non-volatile caching medium with the Hybrid Priority value specified by the command and return command complete with no error.

13.20.2.2.2.4 MAX PRIORITY BEHAVIOR bit is cleared to zero and CACHE BEHAVIOR (CB) bit is set to one

13.20.2.2.2.4.1 Disk is spun down

If:

a) the device processes a read command or a write command, with the HYBRID INFORMATION IS VALID bit set to one and the Hybrid Priority value set to the Maximum Hybrid Priority Level;

b) the MAX PRIORITY BEHAVIOR bit is cleared to zero;
c) the CB bit is set to one; and
d) the disk is spun down,

then the device shall:

a) not put the requested logical sectors in the non-volatile caching medium;
b) not associate the logical sectors with the Hybrid Priority value specified by the command; and
c) return command complete with no error.

13.20.2.2.2.4.2 Disk is spinning

If:

a) the device processes a read command or a write command, with the HYBRID INFORMATION IS VALID bit set to one and the Hybrid Priority value set to the Maximum Hybrid Priority Level;
b) the MAX PRIORITY BEHAVIOR bit is cleared to zero;
c) the CB bit is set to one; and
d) the disk is spinning,

then the device shall behave as defined in 13.20.2.2.2.3.

13.20.2.2.3 HYBRID PRIORITY field is less than MAXIMUM HYBRID PRIORITY LEVEL field and is greater than zero

13.20.2.2.3.1 CACHE BEHAVIOR (CB) bit is cleared to zero

If the device processes a read command or a write command, with the non-zero Hybrid Priority value set less than the Maximum Hybrid Priority Level, and the CB bit is cleared to zero, then:

1) the device may evict logical sectors from the non-volatile caching medium in order to insert the logical sectors specified by the command with Hybrid Priority value if the logical sectors in the non-volatile caching medium:
    A) have a lower Hybrid Priority; or
    B) have the same Hybrid Priority but is older;
2) if the logical sectors specified by the command are not currently in the non-volatile caching medium, then the device:
    A) should put the requested logical sectors to the non-volatile caching medium; and
    B) may put the requested logical sectors to the non-volatile caching medium after returning command completion;
3) the device should associate the logical sectors specified by the command with the Hybrid Priority value supplied by the command; and
4) the device shall return command complete with no error.

13.20.2.2.3.2 CACHE BEHAVIOR (CB) bit is set to one

13.20.2.2.3.2.1 Disk is spun down

If:

a) the device processes a read command or a write command, with the HYBRID INFORMATION IS VALID bit set to one and the Hybrid Priority value set less than the Maximum Hybrid Priority Level;
b) the CB bit is set to one; and
c) the disk is spun down,

then the device shall:

a) not put the requested logical sectors in the non-volatile caching medium;
b) not associate the logical sectors with the Hybrid Priority value specified by the command; and
c) return command complete with no error.

13.20.2.2.3.2.2 Disk is spinning

If:

a) the device processes a read command or a write command, with the HYBRID INFORMATION IS VALID set to one and the Hybrid Priority value set less than the Maximum Hybrid Priority Level;
b) the CB bit is set to one; and
c) the disk is spinning,

then the device shall behave as if the CB bit is cleared to zero (see 13.20.2.2.3.1).

13.20.2.2.3.3 HYBRID PRIORITY field is zero

If the device processes a read command or a write command, with the Hybrid Priority value cleared to zero, then:

a) no Hybrid Priority preference is specified; and
b) the device should write to primary medium.

13.20.3 Syncing

The Hybrid Information log indicates several parameters related to syncing:

a) the DIRTY LOW THRESHOLD field (see 13.7.8.2.5); and
b) the DIRTY HIGH THRESHOLD field (see 13.7.8.2.6).

The device should prioritize the logical sectors contained in the non-volatile caching medium to be synched as follows:

1) starting from the Hybrid Priority zero, up to the Maximum Hybrid Priority Level; and
2) within each Hybrid Priority, sync the least recently used logical sectors first and then the most recently used logical sectors.

13.20.4 Interactions with ATA power management

If :

a) The Hybrid Information feature is enabled;
b) the device processes a read command or a write command; and
c) the requested logical sectors are not accessible in the current power condition,

then:

a) the device shall return command aborted; and
b) if the failing command is READ FPDMA QUEUED or WRITE FPDMA QUEUED, then the device shall set the following in the Queued Error log (see 13.7.4):
    A) the SENSE KEY field shall be set to NOT READY; and
    B) the additional sense code (i.e., the ADDITIONAL SENSE CODE field and the ADDITIONAL SENSE CODE QUALIFIER field) shall be set to LOGICAL UNIT NOT READY – INITIALIZING COMMAND REQUIRED.

If the device indicates a SENSE KEY field of NOT READY and an additional sense code of LOGICAL UNIT NOT READY – INITIALIZING COMMAND REQUIRED, then the device should be explicitly spun up before the host reissues the command.

If the power condition was entered as a result of processing the EPC Go To Power Condition command (see ACS-4) with the HOLD POWER CONDITION bit cleared to zero, then see Table 126 for interactions with the current power source reported by the device.

Table 126 – Current power source interactions

Current Power Source [a]	Description
0h	vendor specific
1h	The device should not go to a power condition that consumes more power to sync logical sectors.
2h	Syncing operations should not adversely affect performance. The device should be more aggressive about syncing than when on battery. If the device changed power condition to process syncing, then the device should return to the previous power condition on completion of the syncing operation.
[a] The CURRENT POWER SOURCE field is indicated in the Identify Device Data log, page 04h.	

13.20.5 Other Hybrid conditions

13.20.5.1 NVM SIZE CHANGED bit

The device may reduce the NVM Size of the non-volatile caching medium. If the device reduces the NVM Size of the non-volatile caching medium, then

a) the device shall set the NVM SIZE CHANGED bit to one in the HYBRID HEALTH field (see 13.7.8.2.4) of the Hybrid Information log; and

b) if the Hybrid Information feature is enabled and the device processes a SMART RETURN STATUS command, then the device shall:
   A) set the value of LBA field (23:8) to 2CF4h (i.e., the device has detected a threshold exceeded condition); and
   B) return command complete with no error.

If the host reads the Hybrid Information log, then the device shall clear the NVM SIZE CHANGED bit after returning the log data to the host.

13.20.5.2 READ ONLY bit

The device may change the non-volatile caching medium to read-only access. If the device changes the non-volatile caching medium to read-only access, then the device shall set the READ ONLY bit to one in the HYBRID HEALTH field (see 13.7.8.2.4) of the Hybrid Information log.

The device may clear the READ ONLY bit under vendor specific conditions.

13.20.5.3 DATA LOSS bit

If the device encounters conditions such that some logical sectors in the non-volatile caching medium is no longer accessible, then:

a) the device shall set the DATA LOSS bit to one in the HYBRID HEALTH field (see 13.7.8.2.4) of the Hybrid Information log; and
b) if the Hybrid Information feature is enabled and the device processes a SMART RETURN STATUS command, then the device shall:
   A) set the value of LBA field (23:8) to 2CF4h (i.e., the device has detected a threshold exceeded condition); and
   B) return command complete with no error.

If the host reads the Hybrid Information log, then the device shall clear the DATA LOSS bit after returning the log data to the host.

13.20.5.4 UNUSEABLE bit

If the device encounters conditions such that the non-volatile caching medium has become unuseable, then:

a) the device shall set the Unuseable bit to one in the HYBRID HEALTH field (see 13.7.8.2.4) of the Hybrid Information log;
b) the device shall disable the Hybrid Information feature (i.e., IDENTIFY DEVICE data Word 79 bit 9 cleared to zero);
c) the device shall remove indication of support for the Hybrid Information feature (i.e., IDENTIFY DEVICE data Word 79 bit 9 cleared to zero); and
d) if the device processes a SMART RETURN STATUS command, then the device shall:
   A) set the value of LBA field (23:8) to 2CF4h (i.e., the device has detected a threshold exceeded condition); and
   B) return command complete with no error.

The device may clear the Unuseable bit under vendor specific conditions.

13.20.6 Automatic Disable

If:

a) the Hybrid Information feature is currently enabled; and
b) the device has not processed any command to read the Hybrid Information log for 25 consecutive power cycles,

then the device shall:

1) change the Hybrid Priority for all logical sectors in the non-volatile caching medium to zero;
2) clear the ENABLED field (see 13.7.8.2.3) in the Hybrid Information log to zero; and

3) disable the Hybrid Information feature (i.e., clear IDENTIFY DEVICE data Word 79 bit 9 to zero).

13.21 Rebuild Assist (optional)

13.21.1 Rebuild Assist overview

The Rebuild Assist mode provides a method for a host controlling the rebuild process to determine that logical sectors on the failed device are unreadable without having to read every LBA to determine the unreadable logical sectors (i.e., the read command is terminated with an error and the failed LBA is reported in the sense data). The storage array controller then may reconstruct the failed logical sectors. The remaining logical sectors may be copied to the replacement device.

Enabling the Rebuild Assist feature:

a) may cause the device to initiate a self test to identify the scope of failures, if any;
b) modifies read command recovery behavior based on the setting of the RARC bit (see 13.6.4); and
c) allows the NCQ Error log to indicate the location of multiple failing LBAs on both READ FPDMA QUEUED commands or WRITE FPDMA QUEUED commands.

If the device processes a power cycle, then the Rebuild Assist feature shall be disabled. All other resets shall not affect the Rebuild Assist feature.

If the Rebuild Assist feature is supported, then the device shall also support the NCQ Autosense feature.

Self-test operations performed while Rebuild Assist mode is enabled may result in detection of failed physical elements.

A predicted unrecovered error is an unrecovered error that is the result of an attempt to access an LBA associated with a failed physical element.

An unpredicted unrecovered error is an unrecovered error that is the result of accessing an LBA that is not associated with a failed physical element.

13.21.2 Enabling Rebuild Assist feature

If the host writes to the Rebuild Assist log and sets the REBUILD ASSIST ENABLED bit to one, then:

a) the device may initiate a self test of the physical elements contained within the device and should disable any physical elements that are not functioning correctly;
b) the device shall initialize the DISABLED PHYSICAL ELEMENTS field from the results of the self-test;
c) the device shall minimize device-initiated background activities; and
d) the device shall enable the Rebuild Assist feature.

The host may verify that Rebuild Assist feature is enabled by reading the Rebuild Assist log, and then examining the data returned and verifying that the REBUILD ASSIST ENABLED bit is set to one.

13.21.3 Using the Rebuild Assist feature

13.21.3.1 Using the Rebuild Assist feature overview

If the Rebuild Assist feature is enabled, then the host should issue sequential READ FPDMA QUEUED commands to extract the available data from the device. If a READ FPDMA QUEUED command does not detect an unrecovered error, then the command should complete without error.

The Rebuild Assist feature allows reporting of an unrecovered read error or an unrecovered write error that is either predicted (i.e., a predicted unrecovered error) or unpredicted (i.e., an unpredicted unrecovered error).

If a device processes a READ FPDMA QUEUED command with the RARC bit set to one, then Rebuild Assist feature shall not affect processing of the READ FPDMA QUEUED command.

13.21.3.2 Unpredicted Unrecovered Read Error

If the device processes a READ FPDMA QUEUED command with the RARC bit cleared to zero and detects an unpredicted unrecovered error, then the device:

a) performs limited read recovery that is vendor specific;
b) transfers the data for all recovered logical sectors, if any, from the Starting LBA of the failed READ FPDMA QUEUED command up to the unrecovered logical sector;
c) shall terminate the READ FPDMA QUEUED command with an error, with the following information recorded in the Queued Error log (see 13.7.4):
    A. the SENSE KEY field shall be set to MEDIUM ERROR;
    B. the ADDITIONAL SENSE CODE field and the ADDITIONAL SENSE CODE QUALIFIER field shall be set to UNRECOVERED READ ERROR; and
    C. the LBA field shall be set to the LBA of the first unrecovered logical sector;
    and
d) may use this failure in a vendor specific manner to predict other logical sectors that may be unrecovered.

NOTE 78 - If the host receives sense data with the SENSE KEY field set to MEDIUM ERROR and additional sense code set to UNRECOVERED READ ERROR, then the host is recommended to issue the next read command with the Starting LBA set to the contents of the FINAL LBA IN ERROR field plus one.

13.21.3.3 Predicted Unrecovered Read Error

If the device processes a READ FPDMA QUEUED command with the RARC bit cleared to zero and detects a predicted unrecovered error, then the device:

a) performs limited read recovery that is vendor specific;
b) transfers the data for all recovered logical sectors, if any, from the Starting LBA of the failed READ FPDMA QUEUED command up to the first unrecovered logical sector; and
c) shall terminate the READ FPDMA QUEUED command with an error, with the following information recorded in the Queued Error log (see 13.7.4):
    A) the SENSE KEY field shall be set to ABORTED COMMAND;
    B) the ADDITIONAL SENSE CODE field and the ADDITIONAL SENSE CODE QUALIFIER field shall be set to MULTIPLE READ ERRORS;
    C) the LBA field shall be set to the LBA of the first unrecovered logical sector; and
    D) the FINAL LBA IN ERROR field shall be set to the LBA of the last predicted unrecovered logical sector in a sequence of contiguous unrecovered logical sectors that started with the first LBA in error.

NOTE 79 - If the host receives sense data with the SENSE KEY field set to ABORTED COMMAND and additional sense code set to MULTIPLE READ ERRORS, then the host is recommended to issue the next read command with the Starting LBA set to the contents of the FINAL LBA IN ERROR field plus one.

13.21.3.4 Unpredicted unrecovered write error

If the device encounters an unpredicted unrecovered error on a write command that is not the WRITE FPDMA QUEUED command, then the device shall terminate the command with an error.

If the device encounters an unpredicted unrecovered error on a WRITE FPDMA QUEUED command, then the device shall terminate the command with an error, with the following information recorded in the Queued Error log (see 13.7.4):

a) the SENSE KEY field shall be set to MEDIUM ERROR;
b) the ADDITIONAL SENSE CODE field and the ADDITIONAL SENSE CODE QUALIFIER field shall be set to WRITE ERROR; and
c) the LBA field shall be set to the LBA of the first unrecovered logical sector.

13.21.3.5 Predicted unrecovered write error

If the device encounters a predicted unrecovered error on a write command that is not the WRITE FPDMA QUEUED command, then the device shall terminate the command with an error.

If the device encounters a predicted unrecovered error on a WRITE FPDMA QUEUED command, then the device shall terminate the command with an error, with the following information recorded in the Queued Error log (see 13.7.4):

a) the SENSE KEY field shall be set to ABORTED COMMAND;
b) the ADDITIONAL SENSE CODE field and the ADDITIONAL SENSE CODE QUALIFIER field shall be set to MULTIPLE WRITE ERRORS;
c) the LBA field shall be set to the LBA of the first unrecovered logical sector; and
d) the FINAL LBA IN ERROR field shall be set to the LBA of the last predicted unrecovered logical sector in a sequence of contiguous unrecovered logical sectors that started with the first LBA in error.

NOTE 80 - If the host receives sense data with the SENSE KEY field set to ABORTED COMMAND and additional sense code set to MULTIPLE WRITE ERRORS, then the host is recommended to issue the next write command with the Starting LBA set to the contents of the FINAL LBA IN ERROR field plus one.

13.21.4 Disabling the Rebuild Assist feature

The Rebuild Assist feature shall be disabled if:

a) the device processes a power cycle; or
b) the device processes a command to write to the Rebuild Assist log (see 13.7.9) with the REBUILD ASSIST ENABLED bit cleared to zero.

13.21.5 Testing the Rebuild Assist feature

The Rebuild Assist log (see 13.7.9) provides a method to test the host's rebuild process.

A device is put into a simulated failing condition by writing to the Rebuild Assist log with the REBUILD ASSIST ENABLED bit set to one and the DISABLED PHYSICAL ELEMENTS field with one or more bits set to one. The host may write to the Rebuild Assist log more than once to simulate additional failing physical elements.

Each bit in the DISABLED PHYSICAL ELEMENTS field represents a physical element that is associated with a group of LBAs that are treated as predicted unrecovered read errors and predicted unrecovered write errors. The correlation of bits in the DISABLED PHYSICAL ELEMENTS field to LBAs in the device is vendor specific.

To end this test, disable the Rebuild Assist feature (see 13.21.4).

13.22 Out Of Band Management (optional)

13.22.1 Out Of Band Management Interface Overview

The Out Of Band Management interface provides the ability to report attribute information about the Serial ATA device using the methods defined in SFF-8609. The attribute information is defined in the Out Of Band Management Control log (see 13.7.10) and corresponds to the Data Code values in the Data Type Definition in SFF-8609. Support for theOut Of Band Management interface is indicated by the OUT OF BAND MANAGEMENT INTERFACE SUPPORTED bit (see 13.7.11.2.29). The Out Of Band Management interface is enabled if the the REPORTING ENABLED bit is set to one in Out Of Band Management Control log.

The behavior of the Out Of Band Management interface when the device is in the DevSleep interface power state (see 8.1) is vendor specific.

The attribute information is transferred as described in SFF-8609. Fields in the Out Of Band Management Control log specify and control the frequency at which attribute information is transferred. The device transfers the enabled attribute information over the Out Of Band Management interface.

For each enabled attribute, the device should transfer the attribute information based on the reporting interval and other fields in the attribute control descriptor.

If

a) multiple attributes are enabled; and
b) multiple attributes should be transferred at the same time based on the reporting interval of the enabled attributes,

then the device shall choose the attribute to be transferred next based on a vendor specific algorithm.

If:

1) the REPORTING ENABLED bit (see Figure 398) is cleared to zero (i.e., the Out Of Band Management interface is disabled); and
2) the device processes a General Purpose Logging feature set command that causes the REPORTING ENABLED bit to change from zero to one (i.e., enabling the Out Of Band Management interface),

then device shall transfer the protocol revision code packet (see SFF-8609) five times at a one second interval, before transferring any enabled attribute information.

If the Out Of Band Management interface is enabled (i.e., the REPORTING ENABLED bit is set to one), then after a:

a) power-on reset;
b) hardware reset; or
c) download microcode activation (see ACS-4),

the device shall transfer the protocol revision code packet five times at a one second interval, before transferring any enabled attribute information.

If the REPORTING ENABLED bit is set to one and the device is going to the:

a) standby mode (see ACS-4); or
b) sleep mode (see ACS-4),

then the device should transfer the stopping transmission packet (see SFF-8609), twice with no more than one second from the start of the first packet transfer to the start of the second, prior to entry into this mode.

If the stopping transmission packet was transferred as part of entry into the standby mode or sleep mode, then the device should stop transferring the attribute information over the Out Of Band management interface until the device goes back to the:

a) active mode (see ACS-4); or
b) idle mode (see ACS-4).

If the device processes a General Purpose Logging feature set command that causes:

a) the REPORTING ENABLED bit (see Figure 398) to change from one to zero; or
b) the REPORTING ENABLED bit is set to one and the attribute enable bit (e.g., TEMPERATURE REPORTING ENABLED bit (see Figure 401)) in all attribute control descriptors (see Figure 399) to become or remain zero when the REPORTING ENABLED bit is set to one,

then the device should transfer the stopping transmission packet twice with no more than one second from the start of the first packet transfer to the start of the second, prior to stopping transmission of the attribute information over the Out Of Band Management interface.

14 Host adapter register interface

14.1 Host adapter register interface overview

Serial ATA host adapters include an additional block of registers mapped separately and independently from the ATA Command Block Registers for reporting additional status and error information and to allow control of capabilities unique to Serial ATA. These additional registers, referred to as the Serial ATA Status and Control Registers (SCR's) are organized as 16 contiguous 32 bit registers. The base address and mapping scheme for these registers is defined by the specific host adapter implementation.

EXAMPLE -PCI controller implementations may map the SCR's using the PCI mapping capabilities.

Table 127 illustrates the overall organization of the Serial ATA register interface including both the ATA Command Block Registers and the Status and Control registers. Legacy mode software does not make use of the Serial ATA Status and Control registers. The Serial ATA Status and Control register are associated with the serial interface and are independent of any device 0/device 1 emulation the host adapter may implement.

Table 127 – SCR definition

ATA Command Block and Control Block registers

	A2	A1	A0	Read	Write
CS 0	0	0	0	Data	Data
	0	0	1	Error	Features
	0	1	0	Count(7:0) [15:8] [7:0]	Count(7:0) [15:8] [7:0]
	0	1	1	LBA [31:24] [7:0]	LBA [31:24] [7:0]
	1	0	0	LBA [39:32] [15:8]	LBA [39:32] [15:8]
	1	0	1	LBA [47:40] [23:16]	LBA [47:40] [23:16]
	1	1	0	Device	Device
	1	1	1	Status	Command
CS 1	1	1	0	Alternate Status	Device Control

Serial ATA Status and Control registers

SATA register	0	SATA Status/Control
SATA register	1	SATA Status/Control
…	…	SATA Status/Control
SATA register	14	SATA Status/Control
SATA register	15	SATA Status/Control

14.2 Status and Control registers

14.2.1 Status and Control registers overview

Serial ATA provides an additional block of registers to control the interface and to retrieve interface state information. There are 16 contiguous registers allocated that the first five are defined and the remaining 11 are reserved for future definition. Table 128 defines the Serial ATA Status and Control registers.

Table 128 – SCR definition

SCR[0]	SStatus register
SCR[1]	SError register
SCR[2]	SControl register
SCR[3]	SActive register
SCR[4]	SNotification register
SCR[5]	Reserved
…	…
SCR[15]	Reserved

14.2.2 SStatus register

The Serial ATA interface Status (SStatus) register is a 32 bit read-only register that conveys the current state of the interface and host adapter (see Figure 433). The register conveys the interface state at the time it is read and is updated continuously and asynchronously by the host adapter. Writes to the register have no effect.

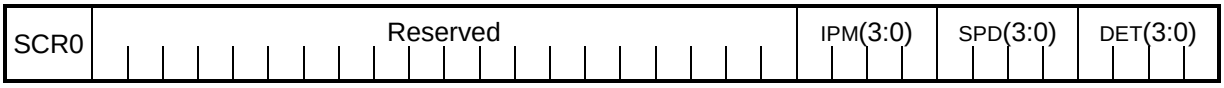


Figure 433 – SStatus register definition

Field Definitions

IPM The IPM field value indicates the current interface power management state.
0000b Device not present or communication not established.
0001b Interface in active state.
0010b Interface in Partial power management state.
0110b Interface in Slumber power management state.
1000b Interface in DevSleep power management state.
All other values reserved.

SPD The SPD field value indicates the negotiated interface communication speed established.
0000b No negotiated speed (device not present or communication not established).
0001b Gen1 communication speed negotiated.
0010b Gen2 communication speed negotiated.
0011b Gen3 communication speed negotiated.
All other values reserved.

DET The DET field value indicates the interface device detection and Phy state.
0000b No device detected and Phy communication not established.
0001b Device presence detected but Phy communication not established.
0011b Device presence detected and Phy communication established.
0100b Phy in offline mode as a result of the interface being disabled or running in a BIST loopback mode.
All other values reserved.

NOTE 81 - The interface needs to be in the active state for the interface device detection value (DET field) to be accurate.

If the interface is in the Partial, Slumber, or DevSleep state no communication between the host and target is established resulting in a DET field value corresponding to no device present or no communication established. As a result the insertion or removal of a device may not be accurately detected under all conditions (e.g., if the interface is quiescent) as a result of being in the Partial,

Slumber, or DevSleep state. This field alone may therefore be insufficient to satisfy all the requirements for device attach or detach detection during all possible interface states.

NOTE 82 - The SStatus register's IPM field value may not correctly represent the current interface low power state (Partial or Slumber) if the host, device, or both are enabled to support Automatic Partial to Slumber transitions.

The IPM field value is guaranteed to indicate that the interface has entered a low power state, however, it may not represent the interface low power state of the host or device currently. See 13.17 for further information regarding Automatic Partial to Slumber transitions.

14.2.3 SError register

The Serial ATA interface Error (SError) register is a 32 bit register that conveys supplemental Interface error information to complement the error information available in the Shadow Register Block Error register (see Figure 434). The register represents all the detected errors accumulated since the last time the SError register was cleared to zero (whether recovered by the interface of not). Set bits in the error register are explicitly cleared to zero by a write operation to the SError register, or a reset operation. On power-on reset all SError register bits are cleared to zero. The value written to clear set error bits shall have 1's encoded in the bit positions corresponding to the bits that are to be cleared to zero. Host software should clear the Interface SError register at appropriate checkpoints in order to best isolate error conditions and the commands they impact.

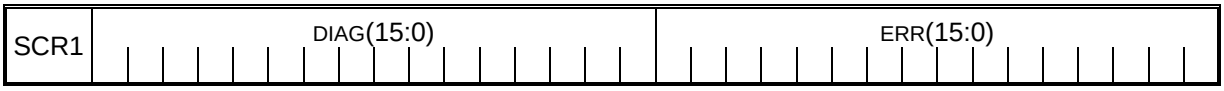


Figure 434 – SError register definition

Field Definitions

DIAG — The DIAG field contains diagnostic error information for use by diagnostic software in validating correct operation or isolating failure modes. The field is bit significant as defined in the following figure (see Figure 436).

ERR — The ERR field contains error information for use by host software in determining the appropriate response to the error condition. The field is bit significant as defined in the following figure (see Figure 435).

ERR	R	R	R	R	E	P	C	T	R	R	R	R	R	R	M	I

Figure 435 – ERR field definition

Field Definitions

R — Reserved (R) bit, shall be cleared to zero.

E — Internal Error (E) bit, the HBA experienced an internal error that caused the operation to fail and may have put the HBA into an error state. Host software should reset the interface before re-trying the operation. If the condition persists, the HBA may suffer from a design issue rendering it incompatible with the attached device.

P — Protocol (P) bit error, a violation of the Serial ATA protocol was detected. This may arise from invalid or poorly formed FISes being received, from invalid state

transitions, or from other causes. Host software should reset the interface and retry the corresponding operation. If such an error persists, the attached device may have a design issue rendering it incompatible with the HBA.

C — Non-recovered persistent Communication (C) bit or data integrity error. A communication error that was not recovered occurred (i.e., expected to be persistent). Since the error condition is expected to be persistent the operation need not be retried by host software. Persistent communications errors may arise from faulty interconnect with the device, from a device that has been removed or has failed, or a number of other causes.

T — Non-recovered Transient (T) bit data integrity error, a data integrity error occurred that was not recovered by the interface. Since the error condition is not expected to be persistent the operation should be retried by host software.

M — Recovered coMmunications (M) bit error, communications between the device and host was temporarily lost but was re-established. This may arise from a device temporarily being removed, from a temporary loss of Phy synchronization, or from other causes and may be derived from the PHYRDYn signal between the Phy and Link layers. No action is required by the host software since the operation ultimately succeeded, however, host software may elect to track such recovered errors in order to gauge overall communications integrity and potentially step down the negotiated communication speed.

I — Recovered data Integrity (I) bit error, a data integrity error occurred that was recovered by the interface through a retry operation or other recovery action. This may arise from a noise burst in the transmission, a voltage supply variation, or from other causes. No action is required by host software since the operation ultimately succeeded, however, host software may elect to track such recovered errors in order to gauge overall communications integrity and potentially step down the negotiated communication speed.

Figure 436 – DIAG field definition

Field Definitions

R — Reserved (R) field, shall be cleared to zero.

A — Port Selector presence detected, the A bit is set to one if COMWAKE is received while the host is in state HP2: HR_AwaitCOMINIT.

X — EXchanged (X) bit, if set to one, indicates that device presence has changed since the last time this bit was cleared to zero. The means that the implementation determines that the device presence has changed is vendor specific. This bit may be set to one anytime a Phy reset initialization sequence occurs as determined by reception of the COMINIT signal whether in response to a new device being inserted, in response to a COMRESET having been issued, or in response to power-up.

F — Unrecognized FIS (F) bit type, if set to one, this bit indicates that since the bit was last cleared to zero, one or more FISes were received by the Transport layer with good CRC, but had a FIS Type field that was not recognized.

T — Transport (T) bit state transition error, if set to one, this bit indicates that an error has occurred in the transition from one state to another within the Transport layer since the last time this bit was cleared to zero.

S — Link Sequence (S) bit error, if set to one, this bit indicates that one or more Link state machine error conditions was encountered since the last time this bit was cleared to zero. The Link layer state machine defines the conditions under that the Link layer detects an erroneous transition.

H — Handshake (H) bit error, if set to one, this bit indicates that one or more R_ERR$_P$ handshake response was received in response to frame transmission. Such errors may be the result of a CRC error detected by the recipient, a disparity or 10b/8b decoding error, or other error condition leading to a negative handshake on a transmitted frame.

C — CRC (C) bit error, if set to one, this bit indicates that one or more CRC errors occurred with the Link layer since the bit was last cleared to zero.

D — Disparity (D) bit error, if set to one, this bit indicates that incorrect disparity was detected one or more times since the last time the bit was cleared to zero.

B — 10b/8b (B) bit Decode error, if set to a one, this bit indicates that one or more 10b/8b decoding errors occurred since the bit was last cleared to zero.

W — COMWAKE (W) bit Detected, if set to one, this bit indicates that a COMWAKE signal was detected by the Phy since the last time this bit was cleared to zero.

I — Phy Internal (I) bit error, if set to one, this bit indicates that the Phy detected some internal error since the last time this bit was cleared to zero.

N — PHYRDY chaNge (N) bit, if set to one, this bit indicates that the PHYRDY signal changed state since the last time this bit was cleared to zero.

14.2.4 SControl register

The Serial ATA interface Control (SControl) register is a 32 bit read-write register that provides the interface that software controls Serial ATA interface capabilities (see Figure 437). Writes to the SControl register result in an action being taken by the host adapter or interface. Reads from the register return the last value written to it.

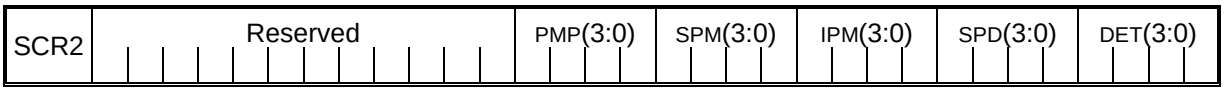


Figure 437 – SControl register definition

Field Definitions

PMP — The Port Multiplier Port (PMP) field represents the 4 bit value to be placed in the PM Port field of all transmitted FISes. This field is 0000b upon power-up. This field is optional and an HBA implementation may choose to ignore this field if the FIS to be transmitted is constructed via an alternative method.

SPM — The Select Power Management (SPM) field is used to select a power management state. A non-zero value written to this field shall cause the power management state specified to be initiated. A value written to this field is not stored for later retrieval. This field shall be read as 0000b.
0000b No power management state transition requested.
0001b Transition to the Partial power management state initiated.
0010b Transition to the Slumber power management state initiated.
0100b Transition to the active power management state initiated.
All other values reserved.

IPM — The IPM field represents the enabled interface power management states that may be invoked via the Serial ATA interface power management capabilities.

0000b No interface power management state restrictions.
0001b Transitions to the Partial power management state disabled.
0010b Transitions to the Slumber power management state disabled.
0011b Transitions to both the Partial and Slumber power management states disabled.
0100b Transitions to the DevSleep power management state are disabled.
0101b Transitions to the Partial and DevSleep power management states are disabled.
0110b Transitions to the Slumber and DevSleep power management states are disabled.
0111b Transitions to the Partial, Slumber, and DevSleep power management states are disabled.
All other values reserved.

SPD The SPD field represents the highest allowed communication speed the interface is allowed to negotiate if interface communication speed is established.
0000b No speed negotiation restrictions.
0001b Limit speed negotiation to a speed not greater than Gen1 communication speed.
0010b Limit speed negotiation to a speed not greater than Gen2 communication speed.
0011b Limit speed negotiation to a speed not greater than Gen3 communication speed.
All other values reserved.

DET The DET field controls the host adapter device detection and interface initialization.
0000b No device detection or initialization action requested.
0001b Perform interface communication initialization sequence to establish communication. This is functionally equivalent to a hard reset and results in the interface being reset and communications reinitialized. Upon a write to the SControl register that sets the DET field to 0001b, the host interface shall transition to the HP1: HR_Reset state (see Figure 258) and shall remain in that state until the DET field is set to a value other than 0001b by a subsequent write to the SControl register.
0100b Disable the Serial ATA interface and put Phy in offline mode.
All other values reserved.

14.2.5 SActive register

The SActive register is a 32 bit register that conveys the information returned in the SActive field of the Set Device Bits FIS (see Figure 438). If NCQ is not supported, then the SActive register does not need to be implemented.

The host may set bits in the SActive register by a write operation to the SActive register. The value written to set bits shall have ones encoded in the bit positions corresponding to the bits that are to be set. Bits in the SActive register are not cleared as a result of a register write operation by the host, and host software is unable to directly clear bits in the SActive register.

Set bits in the SActive register are cleared to zero as a result of data returned by the device in the SActive field of the Set Device Bits FIS. The value returned in the SActive field of the Set Device Bits FIS shall have ones encoded in the bit positions corresponding to the bits that are to be cleared to zero in the SActive register. The device is unable to set bits in the SActive register.

The host controller shall clear all bits in the SActive register to zero upon issuing a COMRESET signal or as a result of issuing a software reset by transmitting a Register Host to Device FIS with the C bit cleared to zero with the SRST bit set to one.

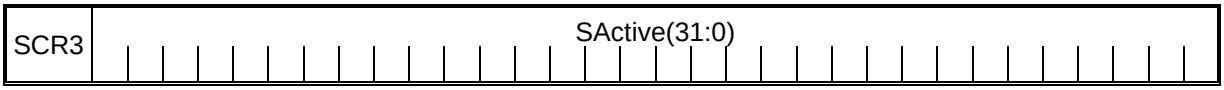


Figure 438 – SActive register definition

Field Definition

Sactive For the NCQ protocol, the SActive field value represents the set of outstanding queued commands that have not completed successfully yet (see 13.6).

14.2.6 SNotification register (optional)

The Serial ATA interface Notification (SNotification) register is a 32 bit register that conveys the devices that have sent the host a Set Device Bits FIS with the N bit set to one, as defined in 10.5.7 (see Figure 439). If the host receives a Set Device Bits FIS with the N bit set to one, the host shall set the bit in the SNotification register corresponding to the value of the PM Port field in the received FIS.

EXAMPLE - If the PM Port field is set to 7, then the host sets bit 7 in the SNotification register to one.

After setting the bit in the SNotification register, the host shall generate an interrupt if the Interrupt bit is set to one in the FIS and interrupts are enabled.

Set bits in the SNotification register are explicitly cleared to zero by a write operation to the SNotification register, or a power-on reset operation. The register is not cleared due to a COMRESET, software is responsible for clearing the register as appropriate. The value written to clear set bits shall have ones encoded in the bit positions corresponding to the bits that are to be cleared to zero.

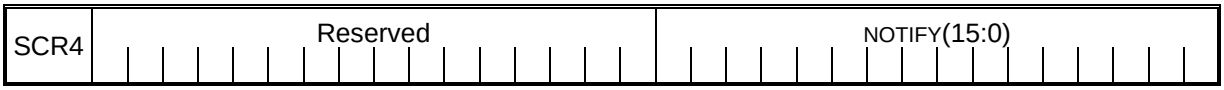


Figure 439 – SNotification register definition

Field Definition

NOTIFY The NOTIFY field represents whether a particular device with the corresponding PM Port number has sent a Set Device Bits FIS to the host with the N bit set to one.

15 Error handling

15.1 Architecture

The layered architecture of Serial ATA extends to error handling as well. As indicated in Figure 440, each layer in the Serial ATA stack has as inputs error indication from the next lower layer (except for the Phy layer that has no lower layer associated with it), data from the next lower layer in the stack, and data from the next higher layer in the stack. Each layer has its local error detection capability to identify errors specific to that layer based on the received data from the lower and higher layers. Each layer performs local recovery and control actions and may forward error information to the next higher layer in the stack.

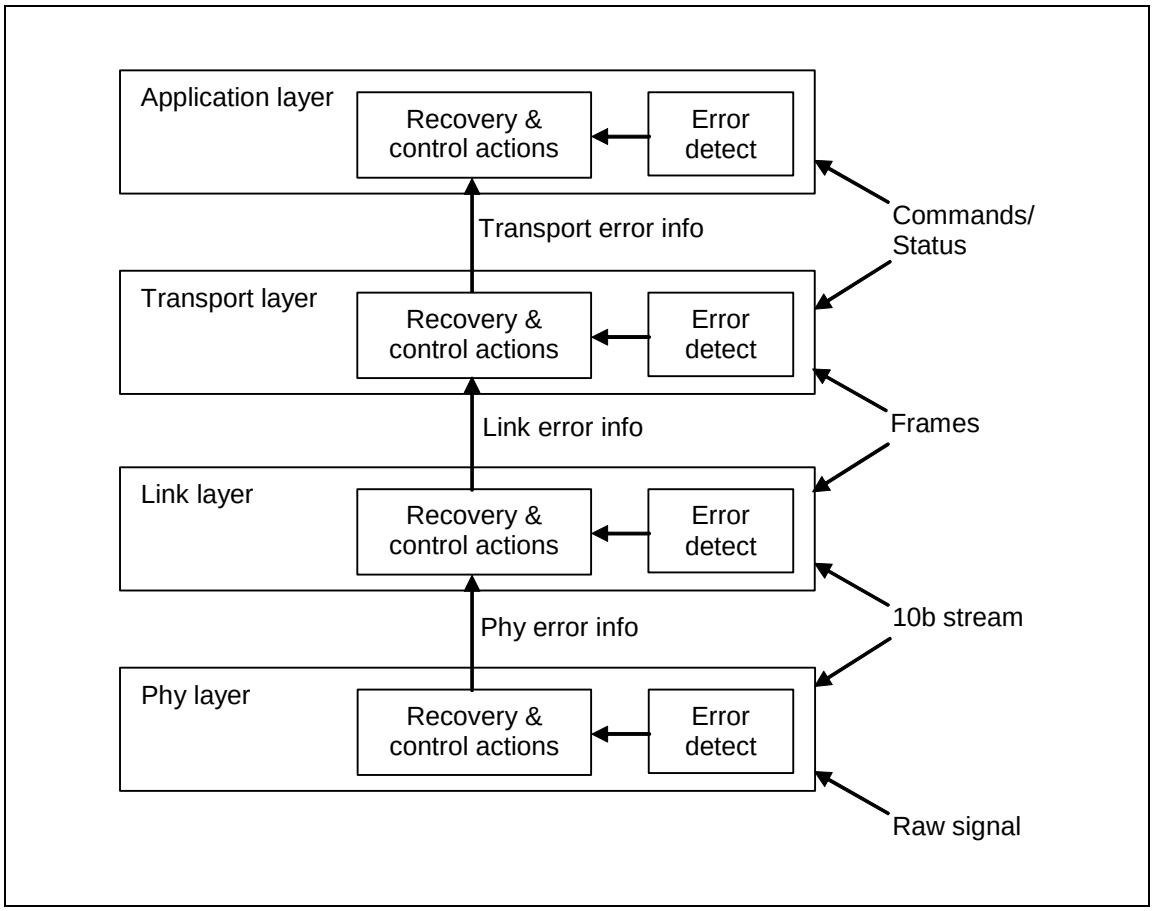


Figure 440 – Error handling architecture

Error responses are generally classified into four categories:

a) Freeze;
b) Abort;
c) Retry; and
d) Track/ignore.

The error handling responses described in this section are not comprehensive and are included to cover specific known error scenarios as well as to illustrate typical error control and recovery actions. This section is therefore descriptive and supplemental to the error reporting interface as defined in 14.2.3 and implementations may vary in their internal error recovery and control actions.

For the most severe error conditions that state has been critically perturbed in a way that it is not recoverable, the appropriate error response is to freeze and rely on a reset or similar operation to restore all necessary state to return to normal operation.

For error conditions that are expected to be persistent, the appropriate error response is to abort and fail the attempted operation. Such failures usually imply notification up the stack in order to inform host software of the condition.

For error conditions that are transient and not expected to persist, the appropriate response is to retry the failed operation. Only failed operations that have not perturbed system state may be retried. Such retries may either be handled directly by the recovery and error control actions in the relevant layer or may be handled by host software in response to error information being conveyed to it.

Non-critical recoverable conditions may either be tracked or ignored. Such conditions include those that were recovered through a retry or other recovery operation at a lower layer in the stack. Tracking such errors may often be beneficial in identifying a marginally operating component or other imminent failure.

15.2 Phy error handling overview

15.2.1 Error detection

There are three primary categories of error that the Phy layer detects internally:

a) no device present;
b) OOB signaling sequence failure; and
c) Phy internal error (loss of synchronization of communications link).

A no device present condition results from a physical disconnection in the media between the host controller and device, whether intermittent or persistent. The host controller shall detect device presence as part of the interface reset sequence as defined in 7.6.33.3.

An OOB signaling failure condition arises if the sequence of OOB signaling events is unable to be completed, prior to the PHYRDY signal being asserted. OOB signaling sequences are required if emerging from a power management Partial or Slumber state, from a loopback BIST test state, or from initial power-up. OOB signaling sequences are used to achieve a specifically ordered exchange of COMRESET, COMINIT, COMWAKE, and $ALIGN_P$ patterns to bring the communications link up between host controller and device. The specific sequences as defined in 7.6.33.3.

A Phy Internal Error may arise from a number of conditions, whether it is caused by the characteristics of the input signal, or an internal error unique to the implementation, it always results in the loss of the synchronization of the communications link.

Fixed local receive PLL frequency architectures (oversampling, tracking/non-tracking) are sensitive to input data speed frequency variations from the nominal expected speed, thus they usually have elasticity buffers. Elasticity buffers are used to accommodate the difference between input data speed frequency, and the local receive PLL frequency. Overrun/underrun conditions may occur if the Tx difference in frequency is too high/low with respect to the Rx local PLL frequency and may result in a Phy Internal Error.

Voltage controlled oscillator (VCO) based PLL clock recovery architectures are also sensitive to input data speed frequency variations, high frequency jitter, and may have trouble achieving lock, which may result in a Phy Internal Error.

A number of state machine, impedance compensation, and serializer/deserializer (SerDes) circuits make up a typical Serial ATA interface Phy, and the various types of error conditions may be

grouped together to make up a "Phy Internal Error". These errors are usually specific to each implementation.

15.2.2 Error control actions

15.2.2.1 No device present

Due to the nature of the physical interface, it is possible for the Phy to determine that a device is attached to the cable at various times. As a direct result, the Phy is responsible for detecting presence of an attached device and this presence shall be reported in the SStatus register such that host software should respond appropriately.

During the interface initialization sequence, an internal “Device Detect” state bit shall be cleared to zero in the host controller if the COMRESET signal is issued. The “Device Detect” state bit shall be set to one in the host controller if the host controller detects a COMINIT signal from the attached device. The “Device Detect” state bit may be set to one in the host controller if the host controller detects a COMWAKE signal while recovering from the Partial or Slumber state. The “Device Detect” state bit corresponds to the device presence detect information in the SStatus register as defined in 14.2.2.

> NOTE 83 - Note that device presence and communications established are separately reported in the SStatus register in order to encompass situations that an attached device is detected by the Phy, but the Phy is unable to establish communications with it.

15.2.2.2 OOB signaling sequence failure

The Phy does not have any timeout conditions for the interface OOB reset signaling sequence as defined in 8.1. If a device is present, the Phy shall detect device presence within 10 ms of a power-on reset (i.e., COMINIT shall be returned within 10 ms of an issued COMRESET). If a device is not present, the Phy is not required to time-out and may remain in the reset state indefinitely until host software intervenes. Upon successful completion of the interface initialization sequence, the Phy shall be ready, active, and synchronized, and the SStatus register bits shall reflect this as defined in 14.2.2.

15.2.2.3 Phy internal error

As defined in 15.2.1, there are several potential sources of errors categorized as “Phy Internal Errors.” In order to accommodate a range of implementations without making the software error handling approach implementation dependent, all the different potential sources of internal Phy errors are combined for the purpose of reporting the condition in the SError register as defined in 14.2.3. This requirement does not preclude each vendor from implementing their own level of error diagnostic bits, but those bits shall reside in vendor specific register locations.

Phy internal errors shall result in the Phy becoming not ready (the PHYRDY signal being negated) and the corresponding SStatus register and SError register bits shall be updated as defined in 14.2.

The PHYRDY change (N) bit, as defined in the SError register, shall be updated as defined in 14.2.3.

15.2.3 Error reporting

Phy errors are generally reported to the Link layer in addition to being reflected in the SStatus register and SError register as defined in 14.2.

15.3 Link layer error handling overview

15.3.1 Error detection

There are two primary categories of errors that the Link layer detects internally are:

a) invalid state transitions; and
b) data integrity errors.

Invalid state transition errors may arise from a number of sources and the Link layer responses to many such error conditions as defined in clause 9. Data integrity errors generally arise from noise in the physical interconnect.

15.3.2 Error control actions

15.3.2.1 Error control actions overview

Errors detected by the Link layer during a transmission are generally handled by accumulating the errors until the end of the transmission and reflecting the reception error condition in the final R_ERR$_P$ /R_OK$_P$ handshake. Specific scenarios are listed in the following sections.

15.3.2.2 Invalid state transitions

Invalid state transitions are generally handled through the return to a known state (i.e., where the Link state machines as defined in 9.7 shows the responses for invalid state transition attempts). Returning to a known state is generally achieved through one of two recovery paths depending on the state of the system.

If the invalid state transitions are attempted during the transmission of a frame (after the receipt of an SOF$_P$), the Link layer shall signal negative acknowledgement (R_ERR$_P$) to the transmitting agent.

If the invalid state transition is not during a frame transmission, the Link shall go directly to the idle state, and await the next operation.

The following paragraphs outline the requirements during specific state transition scenarios, and their respective Link error control actions.

Following reception of one or more consecutive X_RDY$_P$ at the receiver interface, if the next control character received is not SOF$_P$, the Link layer shall notify the Transport layer of the condition and transition to the idle state.

Following transmission of X_RDY$_P$, if there is no returned R_RDY$_P$ received, no Link layer recovery action shall be attempted. The higher-level layers should eventually time out, and reset the interface.

On receipt of an unexpected SOF$_P$, if the receiving interface had not yet signaled readiness to receive data with R_RDY$_P$, that receiving interface shall remain in the idle state issuing SYNC$_P$ primitives until the transmitting interface terminates the transmission and also returns to the idle state.

If the transmitter closes a frame with EOF$_P$ and WTRM$_P$, and receives neither an R_OK$_P$ nor an R_ERR$_P$ within a predetermined timeout, no Link layer recovery action shall be attempted. The higher-level layers should eventually time out, and reset the interface.

If the transmitter signals EOF$_P$, and a primitive other than SYNC$_P$, R_OK$_P$, or R_ERR$_P$ is received, the Link layer shall persistently continue to await reception of a proper terminating primitive.

Data integrity errors are generally handled by signaling the Transport layer in order to potentially trigger a transmission retry operation, or to convey failed status information to the host software.

In order to return to a known state, data integrity errors are usually signaled via the frame acknowledgement handshake, at the end of a frame transmission, before returning to the idle state.

The following paragraphs outline the requirements during specific data integrity error scenarios, and the respective Link error control actions.

On detection of a CRC error at the end of receiving a frame (at EOF_P), the Link layer shall notify the Transport layer that the received frame contains a CRC error. Furthermore, the Link layer shall issue the negative acknowledgement, R_ERR_P, as the frame status handshake, and shall return to the idle state.

On detection of a disparity error or other 8b/10b coding violation during the receipt of a frame, the Link layer shall retain this error information, and at the close of the received frame the Link layer shall provide the negative acknowledgement, R_ERR_P, as the frame handshake, and shall notify the Transport layer of the error.

The control actions are essentially the same for coding violations as for CRC errors.

15.3.3 Error reporting

Link layer error conditions are reported to the Transport layer via a private interface between the Link layer and Transport layer. Additionally, Link layer errors are reported in the SError register as defined in 14.2.3.

15.4 Transport layer error handling

15.4.1 Transport layer error handling overview

The Transport layer is the highest level layer in the Serial ATA interface. The Transport layer communicates errors to the software or performs local error recovery, and initiates control actions (e.g., retrying a class of FIS transmissions).

The Transport layer informs the Link layer of detected errors so that the Link layer reflects Transport errors in the R_ERR_P/R_OK_P handshake at the end of each frame. Devices shall reflect any R_ERR_P frame handshakes in the command ending status reflected in the transmitted Register Device to Host FIS that conveys the operation ending status. The Transport layer also reflects any encountered error information in the SError register.

The Transport layer may retry any FIS transmission, provided the system state has not changed as a result of the corresponding failure, and may retry any number of times. For scenarios where repeated retry operations persistently fail, host software should eventually time out the corresponding command and perform recovery operations.

15.4.2 Error detection

In addition to the error information passed to it by the Link layer, the Transport layer internally detects the following categories of errors:

a) internal errors;
b) frame errors;
c) protocol errors; and
d) state errors.

There are several kinds of internal errors to the Transport layer, including overflow/underflow of the various speed matching FIFOs. Internal errors are generally handled by failing the corresponding transaction and returning to a state equivalent to a failed transaction (e. g., the state that results from a bad CRC).

The Transport layer detects several kinds of frame errors including reception of frames with incorrect CRC, reception of frames with invalid FIS Type field, and reception of ill-formed frames (e.g., a register frames that are not the correct length). Frame errors are generally handled by failing the corresponding transaction and returning to a state equivalent to a failed transaction (e.g., the state that results from a bad CRC).

Protocol and state transition errors often stem from ill-behaved devices not following the proper Serial ATA protocol, and include errors (e.g., the PIO count value not matching the number of data characters subsequently transferred) and errors in the sequence of events.

Protocol and state transition errors are generally handled by failing the corresponding transaction and returning to a state equivalent to a failed transaction (e.g., the state that results from a frame being received with a bad CRC).

15.4.3 Error control actions

15.4.3.1 Internal errors

Internal errors are normally handled by failing the corresponding transaction and either re-trying the transaction or notifying host software of the failure condition in order to ultimately generate a host software retry response. The following are specific internal error scenarios and their corresponding Transport layer error control actions.

If the receive FIFO overflows, the Transport layer shall signal frame reception negative acknowledgement, by signaling the Link layer to return R_ERR_P during the frame acknowledgement handshake. Subsequent actions are equivalent to a frame reception with erroneous CRC.

If the transmit FIFO underruns, the Transport layer shall close the transmitting frame with an EOF_P and CRC value that is forced to be incorrect in order to ensure the receiver of the corrupted frame also processes appropriate error control actions.

15.4.3.2 Frame errors

Frame errors generally are handled in one of two ways depending on whether the error is expected to be transient or persistent and whether system state has been perturbed. For error conditions expected to be transient (e.g., a CRC error), and that the system state has not been perturbed, the Transport layer may retry the corresponding transaction any number of times until ultimately a host timeout and software reset, or other error recovery attempt is made. For error conditions that are not a result of a transient error condition (e.g., an invalid FIS Type field in a received FIS), the error response is generally to fail the transaction and report the failure.

The following are specific frame error scenarios and their corresponding Transport layer error control actions.

The transmitter of a negatively acknowledged frame may retry the FIS transmission provided the system state has not been perturbed. Frame types that may be retransmitted are:

a) Register Host to Device FIS;
b) Register Device to Host FIS;
c) DMA Activate – Device to Host;
d) DMA Setup – Device to Host;
e) PIO Setup – Device to Host;
f) Set Device Bits – Device to Host; and
g) BIST Activate – Host to Device or Device to Host.

Because data transmission FISes result in a change in the HBA's internal state, either through the DMA controller changing its state or through a change in the remaining PIO repetition count, data transmission FISes should never be retried.

The Transport layer is not required to retry those failed FIS transmissions that do not change system state, but the Transport layer may attempt retry any number of times. For conditions that are not addressed through retries (e.g., persistent errors), host software should eventually time out the transaction and reset the interface.

If the Transport layer detects reception of a FIS with unrecognized FIS Type value, the Transport layer shall signal the Link layer to negatively acknowledge the frame reception by asserting R_ERR$_P$ during the frame acknowledgement handshake.

If the Transport layer detects reception of a malformed frame (e.g., a frame with incorrect length), the Transport layer shall signal the Link layer to negatively acknowledge the frame reception by transmitting R_ERR$_P$ during the frame acknowledgement handshake.

15.4.3.3 Protocol and state transition errors

Protocol and state errors stem from ill-behaved devices not following the Serial ATA protocol. Such errors are generally handled by failing the corresponding transactions and returning to a known state. Since such errors are not caused by an environmental transient, no attempt to retry such failed operations should be made. The following are specific frame error scenarios and their corresponding Transport layer error control actions.

If the PIO transfer count expires and EOF$_P$ is not detected two Dwords later (the CRC falls between the last data Dword and EOF$_P$), then the transfer count stipulated in the PIO Setup FIS did not match the size of the subsequent data payload. For this data-payload/transfer-count mismatch, the Transport layer shall signal the Link layer to negatively acknowledge frame reception by transmitting R_ERR$_P$ during the frame acknowledgement handshake.

15.4.4 Error reporting

The Transport layer reports errors to host software via the Serial ATA Status and Control registers. Devices communicate Transport layer error information to host software via transmitting a Register Device to Host FIS to update the ATA Shadow Register Block Status and Error register values.

All Transport layer error conditions that are not handled/recovered by the Transport layer shall set the error bit in the Shadow Register Block Status register, and update the value in the Error register through transmission of an appropriate Register Device to Host FIS.

Host Transport layer error conditions shall result in the status and error values in the SStatus and SError registers being updated with values corresponding to the error condition and shall result in the Link layer being notified to negatively acknowledge the offending FIS during the frame acknowledgement handshake.

15.5 Application layer error handling

15.5.1 Application layer error handling overview

The Application layer error handling is in part defined by behavior of software written for Parallel ATA. Superset error reporting capabilities are supported by the Transport layer through the Status and Control registers, and software should take advantage of those error reporting capabilities to improve error handling for Serial ATA.

15.5.2 Error detection

There are three overall error detection mechanisms that software identifies and responds to Serial ATA errors:

a) bad status in the Command Block Status register;
b) bad status in the SError register; and

c) command failed to complete (i.e., timeout).

Conditions that return bad status in the Command Block Status register, but that no Serial ATA interface error information is available, correspond to the error conditions specified in the ATA standard. Such error conditions and responses are defined in the ATA standard and there is no unique handling of those in Serial ATA. Errors in this category include command errors, (e.g., attempts to read from an LBA past the end of the disk), as well as device-specific failures (e.g., data not readable from the given sector number). These failures are not related to the Serial ATA interface, and thus no Serial ATA specific interface status information is available for these error conditions. Only the status information returned by the device is available for identifying the source of the problem, plus any available SMART data that may apply.

Transport layer error conditions, whether recovered or not, are reflected in the SStatus and SError registers as defined in 14.2.3. The host Transport layer is responsible for reflecting error information in the SStatus and SError registers, while the device Transport layer is responsible for reflecting unrecovered errors in the Shadow Register Block Status and Error registers through transmission of appropriate Register Device to Host FISes.

Commands that fail to complete are detected by host driver software through a timeout mechanism. Generally such timeouts result in no status or error information for the command being conveyed to host software and software may not be able to determine the source or cause of such errors.

15.5.3 Error control actions

Conditions that return bad status in the Shadow Register Block Status register but that no interface error information in the SError register is available shall be handled as defined in the ATA standard.

Conditions that return interface error information in the SError register are handled through four basic responses:

a) Freeze;
b) Abort/Fail;
c) Retry (possible after reset); and
d) Track/ignore.

Error conditions that result in catastrophic system perturbation (i.e., not recoverable) should result in the system halting. Serial ATA does not define any explicit halting conditions, however, software may be able to infer such conditions.

Error conditions that are not expected to be transient and that are not expected to succeed with subsequent attempts should result in the affected command being aborted and failed. Failure of such commands should be reported to higher software layers for handling. Scenarios that this response is appropriate include attempts to communicate with a device that is not attached, and failure of the interface to successfully negotiate communications with an attached device.

Error conditions that are expected to be transient should result in the affected command being retried. Such commands may either be retried directly or may be retried after an interface or device reset, depending on the particular error value reported in the SError register. Scenarios that this response is appropriate include noise events resulting in CRC errors, 8b/10b code violations, or disparity errors.

Conditions that are recoverable and when no explicit error handling is required may be tracked or ignored. Tracking such errors allows subsequent fault isolation for marginal components and accommodates possible recovery operations. Scenarios that this response is appropriate include tracking the number of Phy synchronization losses in order to identify a potential cable fault or to accommodate an explicit reduction in the negotiated communications speed.

16 Port Multiplier

16.1 Introduction

A Port Multiplier is a mechanism for one active host connection to communicate with multiple devices. A Port Multiplier may be thought of as a simple multiplexer where one active host connection is multiplexed to multiple device connections, as shown in Figure 441.

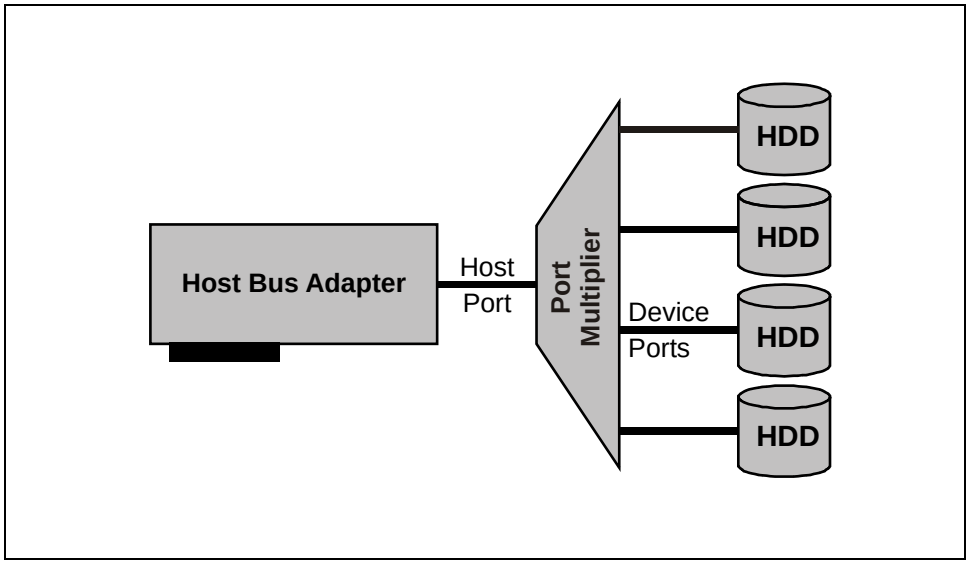


Figure 441 – Port Multiplier overview

Only one active host connection to the Port Multiplier is supported. The Port Multiplier is an extensible design that supports up to 15 endpoint device connections and utilizes the full bandwidth of the host connection. A Port Multiplier shall not be connected to another Port Multiplier (i.e., no cascading).

16.2 Port Multiplier overview

The Port Multiplier uses four bits, known as the PM Port field, in all FIS Types to route FISes between the host and the appropriate device. Using the PM Port field, the Port Multiplier routes FISes to up to 15 Serial ATA devices from one active host. The PM Port field is filled in by the host on a host-to-device FIS with the port address of the device to route the FIS to. For a device-to-host FIS, the PM Port field is filled in by the Port Multiplier with the port address of the device that is transmitting the FIS. Device port addresses start at zero and are numbered sequentially higher until the last device port address has been defined. The control port, port address Fh, is used for control and status communication with the Port Multiplier itself.

In order to utilize all devices connected to a Port Multiplier, the host needs to have a mechanism to set the PM Port field in all transmitted FISes.

The Port Multiplier maintains a set of general purpose registers and also maintains the Serial ATA superset Status and Control registers for each device port. The control port supports two commands, READ PORT MULTIPLIER and WRITE PORT MULTIPLIER, that are used to read and write these registers.

Some additional Port Multiplier features include:

a) supports booting with legacy mode software on device Port 0;
b) supports staggered spinup; and
c) supports hot plug.

16.3 Definition

16.3.1 Addressing mechanism

The Port Multiplier uses four bits, known as the PM Port field, in all FIS Types to route FISes between the host and the appropriate device. Using the PM Port field, the Port Multiplier routes FISes to up to 15 Serial ATA devices from one active host. The PM Port field is filled in by the host on a host-to-device FIS with the port address of the device to route the FIS to. For a device-to-host FIS, the PM Port field is filled in by the Port Multiplier with the port address of the device that is transmitting the FIS.

The PM Port field directly follows the FIS Type field in all FISes, according to 10.5.

16.3.2 Device port requirements

16.3.2.1 Device port requirements overview

A device port is a port that may be used to connect a device to the Port Multiplier. The Port Multiplier may support up to 15 device ports. The device port addresses shall start at zero and shall be numbered sequentially until all device ports have port addresses. A valid device port shall have a port address that has a value less than the total number of device ports supported by the Port Multiplier.

16.3.2.2 Transmission from host to device

To transmit a FIS to a device connected to a Port Multiplier, the host shall set the PM Port field in the FIS to the device's port address. Then the host shall start transmitting the FIS to the Port Multiplier in accordance with the Transport, Link, and Phy state machines.

If a Port Multiplier receives a FIS over the host port, the Port Multiplier shall check the PM Port field in the FIS to determine the port address that the FIS should be transmitted over. If the FIS is destined for the control port, the Port Multiplier shall receive the FIS and perform the command or operation requested. If the FIS is destined for a device port, the Port Multiplier shall perform the following procedure:

1) the Port Multiplier shall determine if the device port is valid. If the device port is not valid, then the Port Multiplier shall issue a SYNC Escape to the host and terminate reception of the FIS (see 16.3.3.8.4);
2) the Port Multiplier shall determine if the X bit in the DIAG field of the device port's PSCR[1] (SError) register is set to one. If set to one, the Port Multiplier shall issue a SYNC Escape to the host and terminate reception of the FIS (see 16.3.3.8.3);
3) the Port Multiplier shall determine if a collision has occurred (i.e., a collision is if a reception is already in progress from the device that the host wants to transmit to). If a collision has occurred, then the Port Multiplier shall finish receiving the FIS from the host and shall issue R_ERR_P to the host as the ending status. The Port Multiplier shall follow the procedures in accordance with 16.3.3.2 to clear the collision condition;
4) the Port Multiplier shall initiate the transfer with the device by issuing X_RDY_P to the device. A collision may occur as the Port Multiplier is issuing the X_RDY_P to the device if the device has just decided to start a transmission to the host. If the device starts transmitting X_RDY_P to the Port Multiplier, a collision has occurred. If a collision has occurred, then the Port Multiplier shall finish receiving the FIS from the host, shall issue R_ERR_P to the host as the ending status. The Port Multiplier shall follow the procedures in accordance with 16.3.3.2 to clear the collision condition; and
5) after the device issues R_RDY_P to the Port Multiplier, the Port Multiplier shall transmit the FIS from the host to the device. The Port Multiplier shall not send R_OK_P status to the

host until the device has issued an R_OK$_P$ for the FIS reception. The R_OK$_P$ status handshake shall be interlocked from the device to the host (see 16.3.3.1).

If an error is detected during any part of the FIS transfer, the Port Multiplier shall ensure that the error condition is propagated to the host and the device.

The transfer between the host and Port Multiplier is handled separately from the transfer between the Port Multiplier and device; only the end of frame R_OK$_P$ handshake is interlocked. The Port Multiplier shall ensure that the flow control signaling latency requirement as defined in 9.5.9 is met for all FIS transfers on a per link basis.
Specifically, the Port Multiplier shall ensure that the flow control signaling latency is met between:

1) the host port and the host it is connected to; and
2) each device port and the device that it is connected to.

If there is not an error detected during the FIS transfer, the Port Multiplier shall not alter the FIS transmitted to the device. The Port Multiplier is not required to check or recalculate the CRC.

16.3.2.3 Transmission from device to host

To transmit a FIS to the host, the device shall proceed with the transmission in accordance with the Transport, Link, and Phy state machines. The device behavior is the same whether it is connected directly to the host or is connected to the host via a Port Multiplier.

If a device wants to transmit a FIS to the host, the Port Multiplier shall perform the following procedure:

1) after receiving X_RDY$_P$ from the device, the Port Multiplier shall determine if the X bit is set to one in the DIAG field of the device port's PSCR[1] (SError) register. The Port Multiplier shall not issue R_RDY$_P$ to the device until the X bit in the DIAG field is cleared to zero;
2) the Port Multiplier shall receive the FIS from the device and shall fill in the PM Port field with the port address of the transmitting device. The Port Multiplier shall transmit the modified FIS to the host with a recalculated CRC. The Port Multiplier shall check the CRC received from the device. If the CRC from the device is invalid the Port Multiplier shall corrupt the CRC sent to the host to ensure that the error condition is propagated. According to 16.3.3.8.5 the Port Multiplier corrupts the CRC in this error case; and
3) the Port Multiplier shall issue X_RDY$_P$ to the host to start the transmission of the FIS to the host, shall wait for reception of R_RDY$_P$ from the host, shall transmit the FIS from the device to the host, shall wait for reception of R_OK$_P$ for the FIS reception from the host, and shall send R_OK$_P$ to the device to complete the R_OK$_P$ status handshake interlock between the device and the host (see 16.3.3.1).

If an error is detected during any part of the FIS transfer, the Port Multiplier shall ensure that the error condition is propagated to the host and the device.

The Port Multiplier may wait for an X_RDY$_P$/R_RDY$_P$ handshake with the host prior to issuing an R_RDY$_P$ to the device to minimize buffering. The transfer between the device and Port Multiplier is handled separately from the transfer between the Port Multiplier and the host; only the end of frame R_OK$_P$ handshake is interlocked. The Port Multiplier shall ensure that the flow control signaling latency requirement as defined in 9.5.9 is met for all FIS transfers on a per link basis.

Specifically, the Port Multiplier shall ensure that the flow control signaling latency is met between:

a) the host port and the host it is connected to; and
b) each device port and the device that it is connected to.

16.3.3 Policies

16.3.3.1 FIS delivery

16.3.3.1.1 FIS delivery overview

The end of frame handshake shall be interlocked between the host and the device. Specifically, the Port Multiplier shall not issue an R_OK$_P$ to the initiator of a FIS before the target of a FIS has issued an R_OK$_P$. The Port Multiplier shall propagate R_OK$_P$ and R_ERR$_P$ from the target of the FIS to the initiator of a FIS.

If a transmission fails before the R_OK$_P$ handshake is delivered to the initiator, the Port Multiplier is responsible for propagating the error condition. Specifically, the Port Multiplier shall propagate SYNC$_P$ primitives received during a FIS transmission end-to-end to ensure that any error condition encountered in the middle of a FIS is propagated. Refer to the requirements given in 16.3.3.8.5 and 16.3.3.8.6 on the appropriate actions to take if CRC calculation errors or possible data corruption occurs in a FIS transmission.

If there is a FIS transfer ongoing and the link between the Port Multiplier and the active device becomes inoperable, then the Port Multiplier should issue SYNC$_P$ primitives to the host until the host responds with SYNC$_P$ in order to fail the transfer. Failing the transfer upon detecting an inoperable link allows the host to proceed with recovery actions immediately, thereby eliminating latency associated with a timeout.

16.3.3.1.2 Port priority

The Port Multiplier shall ensure that an enabled and active device port is not starved. The specific priority algorithm used is implementation specific.

The control port shall have priority over all device transfers. While a command is outstanding to the control port, no device transmissions shall be started by the Port Multiplier until the command outstanding to the control port is completed.

16.3.3.1.3 FIS delivery mechanisms (informative)

This section provides an informative reference for one method that a Port Multiplier may use to satisfy the FIS Delivery policies outlined.

Starting a FIS Transmission, if a device on a Port Multiplier:

1) asserts X_RDY$_P$;
2) the Port Multiplier has selected that device for transmission next; and
3) the host port is not busy, then the Port Multiplier shall:
    1) issue X_RDY$_P$ to the host;
    2) wait for the host to respond with R_RDY$_P$; and
    3) after the host issues R_RDY$_P$, the Port Multiplier issues R_RDY$_P$ to the device,

then the transmission to the host may proceed.

If the host asserts X_RDY$_P$ to the Port Multiplier and the Port Multiplier does not have X_RDY$_P$ asserted to the host, the Port Multiplier responds with R_RDY$_P$ to the host. The Port Multiplier receives the first Dword of the FIS payload from the host. If the Port Multiplier Port specified is a device port that is enabled on the Port Multiplier, the Port Multiplier issues an X_RDY$_P$ over the device port specified and proceeds to transmit the entire FIS to the device. If a collision occurs during this process, the Port Multiplier shall follow the procedures as defined in 16.3.3.2.

Status Propagation, if there is an on-going FIS transmission between the host and a device, the Port Multiplier only issues R_OK$_P$, R_ERR$_P$, and SYNC$_P$ if it has first received that primitive from the host or device, unless a collision occurs or an invalid port is specified.

The Port Multiplier shall not convey R_OK$_P$ or R_ERR$_P$ to the initiator of a FIS until the target of the FIS has issued R_OK$_P$ or R_ERR$_P$ once the end-to-end transmission has commenced.

If the initiator or target of a FIS transmission issues SYNC$_P$ during a FIS transfer, this primitive shall be propagated in order to ensure that the error condition is propagated to either end.

16.3.3.2 Collisions

A collision is if the Port Multiplier has already started a reception from the device that the host wants to transmit to. A collision also occurs if the device issues X_RDY$_P$ at the same time that the Port Multiplier is issuing X_RDY$_P$ to that device. All collisions are treated as an X_RDY$_P$/X_RDY$_P$ collision; in accordance with the Link layer state machine, the host loses all such collisions and should retransmit its FIS at a later time (see Figure 442).

A collision only occurs if the host is trying to issue another native queued command to a device that has native queued commands outstanding. The NCQ protocol guarantees that the host is never transmitting a Data FIS if the collision occurs. This means that the Port Multiplier may safely issue an error to the host and that the host should retry the failed FIS transmission at a later time.

If the Port Multiplier detects a collision, the Port Multiplier shall finish reception of the FIS from the host and shall issue an R_ERR$_P$ to the host for the end of frame handshake. The Port Multiplier shall discard the FIS received from the host. The host should attempt to retry the FIS transfer that failed.

PMC1: PmColl_Idle	Perform normal operation. Wait for FIS reception from host.	
1. Reception started from host for device port X.	→	PmColl_ChkRecv
2. Reception not started from host for device port X.	→	PmColl_Idle

PMC2: PmColl_ChkRecv	Determine if a FIS reception is in progress on device port X.	
1. Reception from device port X in progress.	→	PmColl_Collision
2. Reception from device port X not in progress.	→	PmColl_XmitToDP

PMC3: PmColl_Collision	Collision is detected. Finish reception of the FIS from the host and discard the FIS contents.	
1. $WTRM_P$ or $SYNC_P$ not received from host.	→	PmColl_Collision
2. $WTRM_P$ received from host.	→	PmColl_FailHostFIS
3. $SYNC_P$ received from host.	→	PmColl_Idle

PMC4: PmColl_FailHostFIS	Transmit R_ERR_P to the host.	
1. Unconditional	→	PmColl_Idle

PMC5: PmColl_XmitToDP	Transmit X_RDY_P to the device port X.	
1. X_RDY_P or R_RDY_P not received from device port X.	→	PmColl_XmitToDP
2. X_RDY_P received from device port X.	→	PmColl_Collision
3. R_RDY_P received from device port X.	→	PmColl_Idle

Figure 442 – Port Multiplier collisions state machine

X_RDY_P/X_RDY_P collisions that occur on the host port shall be handled in accordance with the Link layer state machines as defined in 9.7.

16.3.3.3 Booting with software that is not Port Multiplier aware

Booting is accommodated off of the first port of the Port Multiplier, device Port 0, without any special software or hardware support. An HBA that does not support attachment of a Port Multiplier shall work with the device on device Port 0.

If a system requires fast boot capability, it should ensure that both the BIOS and OS driver software are Port Multiplier aware. If software that is not Port Multiplier aware is used in the presence of a Port Multiplier and there is no device present on device Port 0, that software detects a device as present but never receives a Register Device to Host FIS with the device signature. Waiting for the device signature may cause a BIOS that is not Port Multiplier aware to not meet fast boot timing requirements.

16.3.3.4 Staggered spinup support

The Port Multiplier shall disable all device ports on power-up or upon receiving a COMRESET signal over the host port. This feature allows the host to control if each device spins up. Refer to

the requirements given in 13.16.4.2 on how to enumerate a device, including devices that may be spun down because the port is disabled.

16.3.3.5 Hot Plug events

16.3.3.5.1 Hot Plug events overview

Port Multiplier handling of hot plug events is defined by the hot plug state machines for the host port and device port, as defined in 16.3.3.5.2 and 16.3.3.5.3 respectively.

Upon receiving a COMRESET signal from the host, the Port Multiplier shall perform an internal reset. As part of the internal reset, the Port Multiplier shall update the Serial ATA superset Status and Control registers for each device port. A more detailed description of COMRESET handling as defined in 13.16.2.2.

Upon receiving a COMINIT signal from a device, the Port Multiplier shall update the Serial ATA superset Status and Control registers for that port as defined in 13.17. The Port Multiplier shall set the X bit in the DIAG field of the device port's PSCR[1] (SError) register to mark that device presence has changed as defined in 14.2.3. If the Port Multiplier has not received a FIS for the control port after the last COMRESET and an unsolicited COMINIT signal was received over device Port 0, the Port Multiplier shall propagate the COMINIT signal to the host as specified in the hot plug state machine for the host port as defined in 16.3.3.5.2. For all other cases, the COMINIT signal shall not be propagated to the host.

If the X bit in the DIAG field of the PSCR[1] (SError) register is set for a device port, the Port Multiplier shall disallow FIS transfers with that port until the X bit has been cleared to zero. If the Port Multiplier is disallowing FIS transfers with a device port, the Port Multiplier shall ensure FISes the device is attempting to transmit are not dropped.

It is recommended that host software frequently query GSCR[32], as defined in 16.4.2.3, to determine if there has been a device presence change on a device port.

16.3.3.5.2 Hot Plug state machine for host port

Cabled hot plug of the Port Multiplier host port shall be supported since Port Multipliers may not share the same power supply as the host. Therefore the Port Multiplier shall periodically poll for host presence by sending periodic COMINIT signals to the host after transitioning to the HPHP1: NoComm state.

The state machine (see Figure 443) enables device Port 0 after communication has been established over the host port and also clears the X bit in the DIAG field of the PSCR[1] (SError) register for device Port 0 after it is set if operating with software that is not Port Multiplier aware. In addition, the state machine shall propagate unsolicited COMINIT signals received from device Port 0 to the host if no control port register accesses have yet occurred. These accommodations allow device Port 0 to work with host software that is not Port Multiplier aware.

If operating with Port Multiplier aware software, the state machine treats device Port 0 exactly like every other device port.

HPHP1: NoComm [a]	Set Port Multiplier to initial state as defined in 13.16.1 and 13.16.2.2.	
1. Unconditional	→	StartComm
[a] This state is entered upon power-on reset or in response to a received COMRESET.		

HPHP2: StartComm	Transition Phy state machine to state DP1: DR_RESET. Communications retry interval reset to initial value. [a]	
1. Unconditional	→	WaitComm
[a] The communications retry interval shall be greater than or equal to 10 ms.		

HPHP3: WaitComm		
1. PHYRDY asserted.	→	EnablePort0
2. PHYRDY not asserted and communications retry interval not expired.	→	WaitComm
3. PHYRDY not asserted and communications retry interval expired.	→	NoComm

HPHP4: EnablePort0	Enable device Port 0 if device Port 0 is currently disabled.	
1. PHYRDY asserted.	→	Port0ComInit
2. PHYRDY not asserted.	→	NoComm

HPHP4b: Port0ComInit		
1. Communications lost and interface not in power management state (i.e., unplug) and FIS not received for control port.	→	NoComm
2. FIS received for control port.	→	CommOK [a]
3. Communications established and FIS not received for control port and (COMINIT received from device Port 0 or ≥ 10 ms elapsed since entry into HPHP4b: Port0ComInit state).	→	Port0ComInitWait
4. Communications established and FIS not received for control port and COMINIT not received from device Port 0 and < 10 ms elapsed since entry into HPHP4b: Port0ComInit state.	→	Port0ComInit
[a] If bit 0 of the DET field in PSCR[0] (SStatus) for Port 0 is set to one, then the X bit in the DIAG field of PSCR[1] (SError) for Port 0 shall be set prior to making the transition.		

Figure 443 – Host port hot plug state machine (part 1 of 2)

HPHP4c: Port0ComInitWait		
1. Communications lost and interface not in power management state (i.e., unplug) and FIS not received for control port.	→	NoComm
2. FIS received for control port.	→	CommOK [a]
3. Communications established and FIS not received for control port and COMINIT negated from device Port 0.	→	LegacyCommOK [b]
4. Communications established and FIS not received for control port and COMINIT asserted from device Port 0.	→	Port0ComInitWait
[a] If bit 0 of the DET field in PSCR[0] (SStatus) for Port 0 is set to one, then the X bit in the DIAG field of PSCR[1] (SError) for Port 0 shall be set prior to making the transition. [b] The X bit in PSCR[1] (SError) for device Port 0 shall be cleared to zero prior to making the transition.		

HPHP5: LegacyCommOK		
1. Communications lost and interface not in power management state (i.e., unplug) and FIS not received for control port.	→	NoComm
2. Communications established or interface in power management state and COMINIT not received on device Port 0 and FIS not received for control port.	→	LegacyCommOK
3. Communications established or interface in power management state and COMINIT received on device Port 0 and FIS not received for control port.	→	StartComm [a]
4. FIS received for control port.	→	CommOK [b]
[a] The X bit in PSCR[1] (SError) for Port 0 shall be set to one prior to making the transition. [b] If bit 0 of the DET field in PSCR[0] (SStatus) for Port 0 is set to one, then the X bit in the DIAG field of PSCR[1] (SError) for Port 0 shall be set to one prior to making the transition.		

HPHP6: CommOK		
1. Communications lost and interface not in power management state (i.e., unplug).	→	NoComm
2. Communications established or interface in power management state.	→	CommOK

Figure 443 – Host port hot plug state machine (part 2 of 2)

16.3.3.5.3 Hot plug state machine for device port

The device port hot plug behavior (see Figure 444) is exactly the same as the behavior for a host controller supporting device hot plug directly. There is no change to the device and there is no change to the usage model.

DPHP1: NoComm		
1. PHYRDY asserted.	→	CommOK
2. PHYRDY not asserted.	→	NoComm

DPHP2: CommOK		
1. Communications lost and interface not in power management state.	→	NoComm
2. Communications established or interface in power management state.	→	CommOK

Figure 444 – Device port hot plug state machine

16.3.3.6 Link power management

The Port Multiplier shall support reception of PMREQ_P$_P$ and PMREQ_S$_P$ from the host and from attached devices, in accordance with the Link layer state machine. If the Port Multiplier does not support the power management state requested, the Port Multiplier shall respond to the PMREQ_P$_P$ or PMREQ_S$_P$ with PMNAK$_P$.

Before the Port Multiplier delivers a FIS, the Port Multiplier shall check the state of the link and issue a COMWAKE signal if the link is in Partial or Slumber state. The Port Multiplier shall accurately reflect the current state of each device port link in the port specific registers (specifically PSCR[0] (SStatus) for each port) as defined in 16.4.3.

The Port Multiplier shall not propagate PMREQ_P$_P$ or PMREQ_S$_P$ received on a device port. PMREQ_P$_P$ or PMREQ_S$_P$ received from a device only affects the link between that device and the Port Multiplier.

If the Port Multiplier receives PMREQ_P$_P$ or PMREQ_S$_P$ over a device port, the Port Multiplier shall perform the following actions:

a) the Port Multiplier shall respond to the device with PMACK$_P$ or PMNAK$_P$; and
b) if the Port Multiplier responds to the device with PMACK$_P$:
   A) the Port Multiplier shall transition the link with that device to the power state specified; and
   B) the Port Multiplier shall update the port specific registers for that device port to reflect the current power management state of that link.

The Port Multiplier shall propagate PMREQ_P$_P$ or PMREQ_S$_P$ received from the host to all active device ports if the Port Multiplier responds to PMREQ_P$_P$ or PMREQ_S$_P$ with PMACK$_P$. In this case, the Port Multiplier shall propagate PMREQ_P$_P$ or PMREQ_S$_P$ to all device ports that have PHYRDY asserted. If a device responds to PMREQ_P$_P$ or PMREQ_S$_P$ with PMNAK$_P$, this event shall only affect the link with that device. The host may interrogate the Port Multiplier port specific registers to determine that device ports are in a power managed state.

If the Port Multiplier receives PMREQ_P$_P$ or PMREQ_S$_P$ from the host, the specific actions the Port Multiplier shall take are:

a) the Port Multiplier shall respond to the host with PMACK$_P$ or PMNAK$_P$; and
b) if the Port Multiplier responds to the host with PMACK$_P$:
   A) the Port Multiplier shall transition the link with the host to the power state specified; and
   B) the Port Multiplier shall issue PMREQ_P$_P$ or PMREQ_S$_P$ to all device ports that have PHYRDY asserted:

a) if a device responds with PMACK$_P$, the Port Multiplier shall transition the link with that device to the power state specified and shall update the Port Multiplier port specific registers for that port; and
b) if a device responds with PMNAK$_P$, the Port Multiplier shall not take any action with that device port.

The Port Multiplier shall wake device links on an as-needed basis. A COMWAKE signal from the host is not propagated to the devices. The Port Multiplier shall issue a COMWAKE signal to a device if a FIS needs to be delivered to that device.

The Port Multiplier may issue a PMREQ_P$_P$ or PMREQ_S$_P$ to the host if all device ports are in the Partial state, Slumber state, or are disabled.

16.3.3.7 Reducing context switching complexity

It may complicate some host controller designs if traffic from another device is received in the middle of certain FIS sequences.

EXAMPLE - If a host receives a DMA Activate FIS from one device, it may be awkward for the host to receive a FIS from another device before it is able to issue the Data FIS to the first device.

The Port Multiplier shall provide the host with the opportunity to transmit before initiating any pending device transmissions. The Port Multiplier shall not assert X_RDY$_P$ to the host until the Port Multiplier has received at least two consecutive SYNC$_P$ primitives from the host (regardless of whether the host or Port Multiplier was the transmitter of the preceding FIS). In the case of a collision, the Port Multiplier shall ignore this requirement and perform the actions according to 16.3.3.2.

If the host wants to transmit prior to receiving another FIS, the host should issue one SYNC$_P$ between the end of the last FIS transmission and the start of the next FIS transmission. One possible implementation is to transition to the host Link layer state HL_SendChkRdy as defined in 9.7.3 immediately, regardless of whether the host is ready to proceed with the next transmission. The host then only transition out of HL_SendChkRdy state if the host is ready to proceed with the transmission and R_RDY$_P$ is received.

16.3.3.8 Error handling and recovery

16.3.3.8.1 Error handling and recovery overview

The host is responsible for handling error conditions in the same way it handles errors if connected directly to a device. The host is responsible for detecting commands that do not finish and performing error recovery procedures as needed. The exact host software error recovery procedures are implementation specific.

The Port Multiplier is not responsible for performing any error recovery procedures. The Port Multiplier shall return R_ERR$_P$ for certain error conditions as described by the Link layer state machine. The Port Multiplier shall update GSCR[32], as defined in 16.4.2.3, and PSCR[1] (SError) for the device port that experiences an error.

16.3.3.8.2 Command timeout

If a command times out, the host may check PSCR[1] (SError), as defined in 16.4.3, to determine if there has been an interface error condition on the port that had the error. If there has been a device presence change on that port as indicated by the X bit in the DIAG field of PSCR[1] (SError) for the port, the host should re-enumerate the device on that port. The host software error recovery mechanism after a command timeout is implementation specific.

16.3.3.8.3 Disabled device port

If the host transmits a FIS to a device port that is disabled or that has FIS transfers disallowed due to the X bit being set to one in the DIAG field of PSCR[1] (SError) for that port, the Port Multiplier shall not perform an R_OK$_P$ or R_ERR$_P$ handshake at the end of FIS reception and shall instead terminate the FIS reception by issuing SYNC Escape to the host. The host is responsible for detecting that the command did not finish and performing error recovery procedures, including clearing the X bit in the DIAG field of the PSCR[1] register as necessary.

16.3.3.8.4 Invalid device port address

An invalid device port address is a device port address that has a value greater than or equal to the number of device ports that the Port Multiplier supports. If the host specifies an invalid device port address as part of a FIS transmission, the Port Multiplier shall issue a SYNC Escape and terminate reception of the FIS. The host is responsible for detecting that the command did not finish and performing normal error recovery procedures.

16.3.3.8.5 Invalid CRC for device initiated transfer

On a device initiated transfer, the Port Multiplier shall recalculate the CRC since it modifies the first Dword of the FIS. The Port Multiplier shall check the original CRC sent by the device. If the original CRC is invalid, the Port Multiplier shall invert the recalculated CRC to ensure that the CRC error is propagated to the host. The inversion may be done by XORing the recalculated CRC with FFFF FFFFh. The Port Multiplier shall also update the error information in PSCR[1] (SError) for the device port that experienced the error.

16.3.3.8.6 Data corruption

If the Port Multiplier encounters a 10b/8b decoding error or any other error that is able to affect the integrity of the data passed between the transmitter and receiver of a FIS, the Port Multiplier shall ensure that the error is propagated to the receiver. The Port Multiplier may propagate the error by corrupting the CRC for the FIS. The Port Multiplier shall also update the error information in PSCR[1] (SError) for the device port that experienced the error.

16.3.3.8.7 Unsupported command received on control port

If an unsupported command is received on the control port, the Port Multiplier shall respond with a Register Device to Host FIS that has the values shown in Figure 445 for the Status and Error registers.

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	Reserved					ABRT	Reserved	
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	0	0	ERR

Figure 445 – Register values for an unsupported command

Field Definitions

ABRT	1
BSY	0
DRDY	1
DF	0
DRQ	0
ERR	1

16.3.3.9 BIST support

A Port Multiplier may optionally support BIST. A Port Multiplier that supports BIST shall only support BIST in a point-to-point manner. A Port Multiplier that supports BIST shall not propagate a BIST Activate FIS received on one port over another port. The host determines that a Port Multiplier supports BIST by checking GSCR[64], as defined in 16.4.2.4.

To enter BIST mode over the host connection with a Port Multiplier, the host shall issue a BIST Activate FIS to the Port Multiplier control port. The host shall not issue a BIST Activate FIS to a device port.

To enter BIST mode over a device connection, the device shall issue a BIST Activate FIS to the Port Multiplier. The Port Multiplier shall intercept the BIST Activate FIS and enter BIST mode. Upon entering BIST mode with the device, the Port Multiplier shall update the PSCR[0] (SStatus) register for that port to reflect that the link has entered BIST mode, as defined in 10.5.9.

Initiation of BIST by a Port Multiplier is vendor specific.

16.3.3.10 Asynchronous Notification

16.3.3.10.1 Asynchronous Notification overview

A Port Multiplier may optionally support asynchronous notification as defined in 13.8.2. If asynchronous notification is enabled, a Port Multiplier shall send an asynchronous notification to the host if a bit transitions from zero to one in GSCR[32] of the Global Status and Control registers. The Port Multiplier may send one notification for multiple zero to one bit transitions. The asynchronous notification shall only be sent if there is no command currently outstanding to the control port. If there is a command outstanding to the control port if an asynchronous notification needs to be sent, the Port Multiplier shall first complete the command and then send the asynchronous notification. The Port Multiplier shall set the PM Port field in the Set Device Bits FIS to the control port to indicate that the Port Multiplier itself needs attention.

Support for the asynchronous notification feature is indicated in GSCR[64] and it is enabled using GSCR[96].

16.3.3.10.2 Command-based switching (informative)

Host designs should give careful consideration to support of asynchronous notification in command-based switching designs (see16.7.2). If a command-based switching HBA has no explicit accommodation for asynchronous notification, then the host should not enable asynchronous notification on the control port or on any attached device.

16.3.3.11 Phy event counters

16.3.3.11.1 Phy event counters overview

A Port Multiplier may optionally support the Phy event counters feature (see 13.9). If Phy event counters is enabled, a Port Multiplier shall store supported counter information for all of the ports that are enabled, including the host port. It is not required that the same list of Phy event counters be implemented on every port.

Support for the Phy event counters feature is indicated in GSCR[64] and it is enabled using bit 0 in GSCR[34]. The counter values shall not be retained across power cycles. The counter values shall be preserved across COMRESET and software resets that occur on any port.

16.3.3.11.2 Counter identifiers

A Port Multiplier may support any of the counter identifiers described in Figure 407. Support for some counters may not be logical for all ports due to the Port Multiplier architecture.

EXAMPLE - The counters with identifiers 001h and 00Ah.

For the Port Multiplier implementation, all counters other than 000h are optional. A counter identifier value of 000h shall indicate the end of the Phy event counter list implemented in the GSCR or PSCR registers. The counter with identifier 000h shall have no counter value.

All counter values consume a multiple of 16 bits, with a maximum of 64 bits. Each counter is allocated a single register location. The register location contains both the identifier for the counter implemented along with the value of the Phy event counter.

16.3.3.11.3 Reading counter values

Initially, the host may obtain the mapping of counters implemented along with their sizes by submitting reads starting at GSCR[256] (or PSCR[256]) to obtain the identifiers for the counters on a per port basis. Once the identifier of 000h is reached, this signifies the end of the list of counters implemented. The format of this read is a READ PORT MULTIPLIER command with the RS1 bit of the Device register cleared to zero. The value in the PORTNUM field determines that set of counters is to be accessed. Device port counter information for ports 0h to Eh is retrieved by using port numbers 0h to Eh respectively. Host port counter information is retrieved by using the control port number (Fh). The REGNUM field (15:0) shall be set to the specific register location to be read. The output of the READ PORT MULTIPLIER command contains the identifier of the counter. If Phy event counters are enabled, a Port Multiplier shall return identifier 000h as the first counter for a port that no Phy event counters are implemented.

To read the counter value itself, the READ PORT MULTIPLIER command is sent with the RS1 bit of the Device register set to one. The value in the PORTNUM field determines that set of counters are to be accessed. Device port counter information for ports 0h to Eh is retrieved by using port numbers 0h to Eh respectively. Host port counter information is retrieved by using the control port number (Fh). The REGNUM field (15:0) shall be set to the specific register location to be read. The output of the READ PORT MULTIPLIER command contains the value of the counter, up to 64 bits in length. If the counter value read is less than 64 bits in length, the value returned by the Port Multiplier shall have the upper bits padded with zeroes.

All counters are one-extended up to a 16 bit multiple (i.e., 16, 32, 48, 64 bit), once the maximum counter value has been reached. The counter shall stop (and not wrap to zero) after reaching its maximum value.

Upon any read to Phy event counter register space that is at or beyond the identifier 000h location, the Port Multiplier shall return error status with the ERR bit set to one, and the BSY bit and the DRQ bit cleared to zero in the Status field of the FIS. The ABRT bit shall also be set to one in the ERROR field.

16.3.3.11.4 Counter reset mechanisms

There are three mechanisms the host may use to explicitly cause the Phy event counters to be reset. The first mechanism uses the WRITE PORT MULTIPLIER command to clear counters on an individual counter basis. The PORTNUM field determines that set of counters are to be accessed. Device port counter information for ports 0h to Eh is written by using port numbers 0h to Eh respectively. Host port counter information is written by using the control port number (Fh). The REGNUM field shall be set to the register location (counter) value to be written. The Value field within the command contains the value to be written into the register, in this case all zeroes.

The second mechanism allows for a counter to be reset following a read to that counter register. If the RS2 bit in the Device register is set to one for a read to any counter, the Port Multiplier shall return the current counter value for the command and then reset that value upon successful transmission of the Register Device to Host FIS. If retries are required, upon unsuccessful transmission of the FIS it is possible that the counter value may be changed before the re-transmission of the counter value.

The third mechanism is a global reset function by writing appropriate bits in GSCR[34] to reset all Phy event counters for specific ports. A host may reset all Phy event counters by writing FFFFh to bits 31:16 of GSCR[34].

16.4 Port Multiplier registers

16.4.1 Port Multiplier registers overview

The Port Multiplier registers are accessed using READ PORT MULTIPLIER and WRITE PORT MULTIPLIER commands issued to the control port.

16.4.2 General Status and Control registers

16.4.2.1 General Status and Control registers overview

The control port address is specified in the PORTNUM field of the READ PORT MULTIPLIER and WRITE PORT MULTIPLIER commands as defined in 16.5.1 and 16.5.2 in order to read/write the General Status and Control registers.

16.4.2.2 Static Configuration Information

The Static Configuration Information section of the General Status and Control registers contains registers that are static throughout the operation of the Port Multiplier. These registers are read-only (see Table 129).

Table 129 – Static information registers

Field	O/M	F/V	Description
GSCR[0]	M		**Product Identifier**
		F	31:16 Device ID allocated by the vendor.
		F	15:0 Vendor ID allocated by the PCI-SIG.
GSCR[1]	M		**Revision Information**
		F	31:16 Reserved
		F	15:8 Revision level of the Port Multiplier.
		F	7:4 Reserved
		F	3 Supports Port Multiplier specification 1.2.
		F	2 Supports Port Multiplier specification 1.1.
		F	1 Supports Port Multiplier specification 1.0.
		F	0 Reserved
GSCR[2]	M		**Port Information**
		F	31:4 Reserved
		F	3:0 Number of exposed device fan-out ports.
GSCR[3] to GSCR[31]	O	F	Reserved

Key:
O/M = Mandatory/optional requirement.
M = Support of the register is mandatory.
O = Support of the register is optional.
F/V = Fixed/variable content.
F = the content of the register is fixed and does not change.
V = the contents of the register is variable and may change.

Register 0, Product Identifier

The register identifies the vendor that produced the Port Multiplier and the specific device identifier.

Bits 15:0 shall be set to the vendor identifier allocated by the PCI-SIG of the vendor that produced the Port Multiplier.

Bits 31:16 shall be set to a device identifier allocated by the vendor.

Register 1, Revision Information

The register specifies the specification revision that the Port Multiplier supports; the Port Multiplier may support multiple specification revisions. The register also specifies the revision level of the specific Port Multiplier product identified by Register 0.

Bit 0 is reserved.

Bit 1 if set to one, indicates that the Port Multiplier supports Port Multiplier specification version 1.0.

Bit 2 if set to one, indicates that the Port Multiplier supports Port Multiplier specification version 1.1.

Bit 3 if set to one, indicates that the Port Multiplier supports Port Multiplier specification version 1.2.

Bits 7:4 are reserved.

Bits 15:8 identifies the revision level of the Port Multiplier product identified by Register 0. This identifier is allocated by the vendor.

Bits 31:16 are reserved.

Register 2, Port Information

The register specifies information about the ports that the Port Multiplier contains, including the number of exposed device fan-out ports. The number of exposed device ports is the number of device ports that are physically connected and available for use on the product.

Bits 3:0 specifies the number of exposed device fan-out ports. A value of zero is invalid. The control port shall not be counted.

Bits 31:4 are reserved.

Registers 3..31, reserved

Registers 3..31 are reserved for future Port Multiplier definition.

16.4.2.3 Status Information and control

The Status Information section of the General Status and Control registers (see Table 130) contains registers that convey status information and control operation of the Port Multiplier.

Table 130 – Status information and control registers

Field	O/M	F/V	Description
GSCR[32]	M		**Error Information**
		F	31:15 Reserved
		V	14 OR of selectable bits in Port 14 PSCR[1] (SError)
		V	13 OR of selectable bits in Port 13 PSCR[1] (SError)
		V	12 OR of selectable bits in Port 12 PSCR[1] (SError)
		V	11 OR of selectable bits in Port 11 PSCR[1] (SError)
		V	10 OR of selectable bits in Port 10 PSCR[1] (SError)
		V	9 OR of selectable bits in Port 9 PSCR[1] (SError)
		V	8 OR of selectable bits in Port 8 PSCR[1] (SError)
		V	7 OR of selectable bits in Port 7 PSCR[1] (SError)
		V	6 OR of selectable bits in Port 6 PSCR[1] (SError)
		V	5 OR of selectable bits in Port 5 PSCR[1] (SError)
		V	4 OR of selectable bits in Port 4 PSCR[1] (SError)
		V	3 OR of selectable bits in Port 3 PSCR[1] (SError)
		V	2 OR of selectable bits in Port 2 PSCR[1] (SError)
		V	1 OR of selectable bits in Port 1 PSCR[1] (SError)
		V	0 OR of selectable bits in Port 0 PSCR[1] (SError)
GSCR[33]	M		**Error Information Bit Enable**
		V	31:0 If set, bit is enabled for use in GSCR[32]
GSCR[34]	O	V	**Phy Event Counter Control**
			31 Host port global counter reset
			30 Port 14 global counter reset
			29 Port 13 global counter reset
			28 Port 12 global counter reset
			27 Port 11 global counter reset
			26 Port 10 global counter reset
			25 Port 9 global counter reset
			24 Port 8 global counter reset
			23 Port 7 global counter reset
			22 Port 6 global counter reset
			21 Port 5 global counter reset
			20 Port 4 global counter reset
			19 Port 3 global counter reset
			18 Port 2 global counter reset
			17 Port 1 global counter reset
			16 Port 0 global counter reset
			15:1 Reserved
			0 Phy event counters enabled
GSCR[35] to GSCR[63]	O	F	Reserved

Key:
O/M = Mandatory/optional requirement.
M = Support of the register is mandatory.
O = Support of the register is optional.
F/V = Fixed/variable content.
F = the content of the register is fixed and does not change.
V = the contents of the register is variable and may change.

Register 32, Error Information

This register reflects whether specific bits have been set to one in any of the device port's PSCR[1] (SError) register. A bit set to one may reflect an error or that device presence has changed. The host selects the device port's PSCR[1] (SError) bits to reflect using GSCR[33]. The Port Multiplier algorithm for updating the register contents is,

```
for (n=0; n<NumPorts; n++)
{
      if (Port[n].PSCR[1] & GSCR[33]) == 0)
            GSCR[32].Bit[n]=0
      else
            GSCR[32].Bit[n]=1
}
```

This register is read-only. The specific device port's PSCR[1] (SError) register shall be written in order to clear values in the affected PSCR[1] (SError) register and by reflection in this register. Refer to the requirements in 14.2.3 for the definition of the SError register.

Bit 0 shall be set to the logically OR-ed value of selected bits in Port 0 PSCR[1] (SError). The bits to be OR-ed are selected using GSCR[33].

Bit 1 shall be set to the logically OR-ed value of selected bits in Port 1 PSCR[1] (SError). The bits to be OR-ed are selected using GSCR[33]. If Port 1 is not implemented by the Port Multiplier, this bit shall be cleared to zero.

Bit 2 shall be set to the logically OR-ed value of selected bits in Port 2 PSCR[1] (SError). The bits to be OR-ed are selected using GSCR[33]. If Port 2 is not implemented by the Port Multiplier, this bit shall be cleared to zero.

Bit 3 shall be set to the logically OR-ed value of selected bits in Port 3 PSCR[1] (SError). The bits to be OR-ed are selected using GSCR[33]. If Port 3 is not implemented by the Port Multiplier, this bit shall be cleared to zero.

Bit 4 shall be set to the logically OR-ed value of selected bits in Port 4 PSCR[1] (SError). The bits to be OR-ed are selected using GSCR[33]. If Port 4 is not implemented by the Port Multiplier, this bit shall be cleared to zero.

Bit 5 shall be set to the logically OR-ed value of selected bits in Port 5 PSCR[1] (SError). The bits to be OR-ed are selected using GSCR[33]. If Port 5 is not implemented by the Port Multiplier, this bit shall be cleared to zero.

Bit 6 shall be set to the logically OR-ed value of selected bits in Port 6 PSCR[1] (SError). The bits to be OR-ed are selected using GSCR[33]. If Port 6 is not implemented by the Port Multiplier, this bit shall be cleared to zero.

Bit 7 shall be set to the logically OR-ed value of selected bits in Port 7 PSCR[1] (SError). The bits to be OR-ed are selected using GSCR[33]. If Port 7 is not implemented by the Port Multiplier, this bit shall be cleared to zero.

Bit 8 shall be set to the logically OR-ed value of selected bits in Port 8 PSCR[1] (SError). The bits to be OR-ed are selected using GSCR[33]. If Port 8 is not implemented by the Port Multiplier, this bit shall be cleared to zero.

Bit 9 shall be set to the logically OR-ed value of selected bits in Port 9 PSCR[1] (SError). The bits to be OR-ed are selected using GSCR[33]. If Port 9 is not implemented by the Port Multiplier, this bit shall be cleared to zero.

Bit 10 shall be set to the logically OR-ed value of selected bits in Port 10 PSCR[1] (SError). The bits to be OR-ed are selected using GSCR[33]. If Port 10 is not implemented by the Port Multiplier, this bit shall be cleared to zero.

Bit 11 shall be set to the logically OR-ed value of selected bits in Port 11 PSCR[1] (SError). The bits to be OR-ed are selected using GSCR[33]. If Port 11 is not implemented by the Port Multiplier, this bit shall be cleared to zero.

Bit 12 shall be set to the logically OR-ed value of selected bits in Port 12 PSCR[1] (SError). The bits to be OR-ed are selected using GSCR[33]. If Port 12 is not implemented by the Port Multiplier, this bit shall be cleared to zero.

Bit 13 shall be set to the logically OR-ed value of selected bits in Port 13 PSCR[1] (SError). The bits to be OR-ed are selected using GSCR[33]. If Port 13 is not implemented by the Port Multiplier, this bit shall be cleared to zero.

Bit 14 shall be set to the logically OR-ed value of selected bits in Port 14 PSCR[1] (SError). The bits to be OR-ed are selected using GSCR[33]. If Port 14 is not implemented by the Port Multiplier, this bit shall be cleared to zero.

Bits 31:15 are reserved.

Register 33, Error Information Bit Enable

This register selects/enables bits to be used for the OR operation in GSCR[32]. If a bit is set to one, that bit shall be reflected for each device port in GSCR[32]. This is a global enable and is not device port specific. The default value of this register shall be 0400 FFFFh; that corresponds to all bits in the ERR field and the DIAG field's X bit being OR-ed together for each device port in GSCR[32].

Register 34, Phy Event Counter Control

Support for this register is mandatory if Phy event counters is supported by the Port Multiplier as indicated by bit 4 in GSCR[64]. This register is used for Phy event counter control mechanisms.

Bit 0 if set to one, indicates that the Port Multiplier supports Phy event counters and that the feature is currently enabled. If the Port Multiplier supports Phy event counters, it shall support counters for all ports implemented on the Port Multiplier, including the host port. It is not required for the same list of Phy event counters be implemented on every port. If this bit is cleared to zero, then the current values within the counters shall be retained and counting shall stop. This bit shall be cleared to zero on power-up. This bit shall not be affected by a COMRESET or software reset that has occurred on any port. The values within the Phy event counters are not affected by this bit.

Bits 15:1 are reserved.

Bit 16 if set to one, shall result in an immediate reset of all Phy event counters associated with Port 0. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Bit 17 if set to one, shall result in an immediate reset of all Phy event counters associated with Port 1. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Bit 18 if set to one, shall result in an immediate reset of all Phy event counters associated with Port 2. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Bit 19 if set to one, shall result in an immediate reset of all Phy event counters associated with Port 3. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Bit 20 if set to one, shall result in an immediate reset of all Phy event counters associated with Port 4. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Bit 21 if set to one, shall result in an immediate reset of all Phy event counters associated with Port 5. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Bit 22 if set to one, shall result in an immediate reset of all Phy event counters associated with Port 6. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Bit 23 if set to one, shall result in an immediate reset of all Phy event counters associated with Port 7. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Bit 24 if set to one, shall result in an immediate reset of all Phy event counters associated with Port 8. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Bit 25 if set to one, shall result in an immediate reset of all Phy event counters associated with Port 9. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Bit 26 if set to one, shall result in an immediate reset of all Phy event counters associated with Port 10. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Bit 27 if set to one, shall result in an immediate reset of all Phy event counters associated with Port 11. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Bit 28 if set to one, shall result in an immediate reset of all Phy event counters associated with Port 12. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Bit 29 if set to one, shall result in an immediate reset of all Phy event counters associated with Port 13. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Bit 30 if set to one, shall result in an immediate reset of all Phy event counters associated with Port 14. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Bit 31 if set to one, shall result in an immediate reset of all Phy event counters associated with the host port. Once the reset of all counters is complete, this bit shall be cleared to zero automatically by the Port Multiplier. If cleared to zero, no action is taken.

Registers 35..63, Reserved

Registers 35..63 are reserved.

16.4.2.4 Features Supported

The Features Supported section of the General Status and Control registers (see Table 131) contains registers that convey the optional features that are supported by the Port Multiplier. All Features Supported registers are read-only.

Table 131 – Features Supported registers

Field	O/M	F/V	Description
GSCR[64]	M		**Port Multiplier Features Support**
		F	31:5 Reserved
		F	4 Supports Phy event counters
		F	3 Supports asynchronous notification
		F	2 Supports dynamic SSC transmit enable
		F	1 Supports issuing $PMREQ_P$ to host
		F	0 Supports BIST
GSCR[65] to GSCR[95]	O	F	Reserved

Key:
O/M = Mandatory/optional requirement.
M = Support of the register is mandatory.
O = Support of the register is optional.
F/V = Fixed/variable content.
F = the content of the register is fixed and does not change.
V = the contents of the register is variable and may change.

Register 64, Port Multiplier Features Support

This register specifies the optional Port Multiplier features that the Port Multiplier supports.

Bit 0 if set to one indicates that the Port Multiplier supports BIST according to 16.3.3.9. If this bit is cleared to zero the Port Multiplier does not support reception of the BIST Activate FIS.

Bit 1 if set to one indicates that the Port Multiplier supports issuing $PMREQ\_P_P$ and $PMREQ\_S_P$ requests to the host if all device ports are disabled or in a Partial/Slumber state. If this bit is cleared to zero, the Port Multiplier does not support issuing $PMREQ\_P_P$ or $PMREQ\_S_P$ requests to the host.

Bit 2 if set to one indicates that the Port Multiplier supports dynamically enabling and disabling Spread Spectrum Clocking transmit. If this bit is cleared to zero, the Port Multiplier does not support dynamically enabling and disabling Spread Spectrum Clocking transmit.

Bit 3 if set to one indicates that the Port Multiplier supports asynchronous notification. If the Port Multiplier supports asynchronous notification, it shall be capable of sending a Set Device Bits FIS to the host with the Interrupt bit set to one and the N bit set to one if a bit in GSCR[32] transitions from zero to one. If this bit is cleared to zero then the Port Multiplier does not support asynchronous notification.

Bit 4 if set to one indicates that the Port Multiplier supports Phy event counters, and GSCR[34] shall be implemented. If this bit is cleared to zero, the Port Multiplier does not support Phy event counters.

Bits 31:5 are reserved.

Registers 65..95: Reserved

Registers 65..95 are reserved for future Port Multiplier definition.

16.4.2.5 Features Enabled

The Features Enabled section of the General Status and Control registers (see Table 132) contains registers that allow optional features to be enabled. The Features Enabled registers have a one-

to-one correspondence with the Features Supported registers as defined in 16.4.2.4. All Features Enabled registers are read/write.

Table 132 – Features Enabled registers

Field	O/M	F/V	Description
GSCR[96]	M		**Port Multiplier Revision 1.X Features Enable**
		F	31:4 Reserved
		V	3 Asynchronous notification enabled
		V	2 Dynamic SSC transmit is enabled
		V	1 Issuing PMREQ$_P$ to host is enabled
		V	0 BIST support is enabled
GSCR[97] to GSCR[127]	O	F	Reserved

Key:
O/M = Mandatory/optional requirement.
M = Support of the register is mandatory.
O = Support of the register is optional.
F/V = Fixed/variable content.
F = the content of the register is fixed and does not change.
V = the contents of the register is variable and may change.

Register 96, Port Multiplier Revision 1.X Features Enable

This register controls whether the optional Port Multiplier specification revision 1.X features that the Port Multiplier supports are enabled.

Bit 0 if set to one indicates that the Port Multiplier supports BIST and that BIST support is enabled. If this bit is cleared to zero reception of the BIST Activate FIS is not enabled. If the Port Multiplier supports BIST, this bit shall be set to one on power-up or after a reset, otherwise it shall be cleared to zero on power-up and reset.

Bit 1 if set to one indicates that the Port Multiplier supports issuing PMREQ_P$_P$ and PMREQ_S$_P$ requests to the host if all device ports are disabled or in a Partial/Slumber state and the feature is enabled. If this bit is cleared to zero, the Port Multiplier shall not issue PMREQ_P$_P$ or PMREQ_S$_P$ requests to the host. This bit shall be cleared to zero on power-up and reset.

Bit 2 if set to one indicates that the Port Multiplier supports dynamically enabling and disabling Spread Spectrum Clocking transmit and that the feature is currently enabled. If this bit is cleared to zero and the feature is supported as specified in GSCR[64], the Port Multiplier shall not use Spread Spectrum Clocking transmit. If this feature is supported as specified in GSCR[64], the bit shall be set to one on power-up and reset. If this feature is not supported as specified in GSCR[64], then the bit shall be cleared to zero on power-up and reset.

Bit 3 if set to one indicates that the Port Multiplier supports asynchronous notification and that the feature is currently enabled. If the Port Multiplier supports asynchronous notification, it shall be capable of sending a Set Device Bits FIS to the host with the Interrupt bit set to one and the N bit set to one if a bit in GSCR[32] transitions from zero to one. If this bit is cleared to zero and the corresponding bit in GSCR[64] is set to one, then asynchronous notification is not enabled. This bit shall be cleared to zero on power-up and reset.

Bits 31:4 are reserved.

Registers 97..127, are reserved.

Registers 97..127 are reserved for future Port Multiplier definition.

16.4.2.6 Vendor specific

The vendor-specific section of the General Status and Control registers (see Table 133) contains registers that are vendor specific.

Table 133 – Vendor-specific registers

Field	O/M	F/V	Description
GSCR[128] to GSCR[255]	O	X	Vendor specific

Key:
O/M = Mandatory/optional requirement.
M = Support of the register is mandatory.
O = Support of the register is optional.
F/V = Fixed/variable content.
F = the content of the register is fixed and does not change.
V = the contents of the register is variable and may change.
X = the content of the register may be fixed or variable.

Registers 128..255, are vendor specific.

16.4.2.7 Phy event counters

The Phy event counters section of the General Status and Control registers (see Table 134) contains registers that store the data for each of the Phy event counters supported by the Port Multiplier for the host port. The Phy event counters registers contain both the identifier and counter values. All Phy event counter registers are read/write.

A value of zero returned for a counter means that there have been no instances of that particular event.

Table 134 – Phy event counter registers

Field	O/M	F/V	Description
GSCR[256] to GSCR[2 303]	O	V	Phy event counter registers.

Key:
O/M = Mandatory/optional requirement.
M = Support of the register is mandatory.
O = Support of the register is optional.
F/V = Fixed/variable content
F = the content of the register is fixed and does not change.
V = the contents of the register is variable and may change.

Registers 256..2 303, are used for Phy event counters. Phy event counters are not allocated specific registers. Each register contains both the identifier and value for the counter implemented.

16.4.2.8 Reserved

The section of the General Status and Control registers starting at address 2 304 are reserved (see Table 135).

Table 135 – Reserved registers

Field	O/M	F/V	Description
GSCR[2 304] to GSCR[65 535]	O	F	Reserved

Key:
O/M = Mandatory/optional requirement.
M = Support of the register is mandatory.
O = Support of the register is optional.
F/V = Fixed/variable content.
F = the content of the register is fixed and does not change.
V = the contents of the register is variable and may change.

16.4.3 Port Status and Control registers

16.4.3.1 Port Status and Control registers overview

The Port Multiplier shall maintain a set of Port Status and Control registers (PSCRs) for each device port that it supports. These registers are port specific and contain the Serial ATA superset Status and Control registers, along with the Phy event counters information for each port. The host specifies the device port to read or write registers for in the PORTNUM field of the READ PORT MULTIPLIER command or WRITE PORT MULTIPLIER command as defined in 16.5.1 and 16.5.2. The registers are defined in Table 136.

Table 136 – PSCR definition

Field	Definition
PSCR[0]	SStatus register (see 14.2.2)
PSCR[1]	SError register (see 14.2.3)
PSCR[2]	SControl register (see 14.2.4)
PSCR[3]	SActive register (not implemented)
PSCR[4]	SNotification register (not implemented)
PSCR[5] to PSCR[255]	Reserved
PSCR[256] to PSCR[2 303]	Phy event counter registers
PSCR[2 304] to PSCR[65 535]	Reserved

The reset value for the DET field of PSCR[2] (SControl) register shall be 4h (Phy disabled).

16.4.3.2 Phy event counters

The Phy event counter information for each of the device ports within a Port Multiplier is contained in the Port Status and Control registers starting at PSCR[256]. The Phy event counters registers contain both the identifier and counter values. All Phy event counter registers are read/write.

A value of zero returned for a counter means that there have been no instances of that particular event.

16.5 Port Multiplier command definitions

16.5.1 READ PORT MULTIPLIER

16.5.1.1 READ PORT MULTIPLIER overview

The READ PORT MULTIPLIER command (see Figure 446) is used to read a register on a Port Multiplier. The READ PORT MULTIPLIER command shall be issued to the control port.

16.5.1.2 Inputs

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	REGNUM(7:0)							
FEATURES(15:8)	REGNUM(15:8)							
COUNT(7:0)	Reserved							
COUNT(15:8)	Reserved							
LBA(7:0)	Reserved							
LBA(31:24)	Reserved							
LBA(15:8)	Reserved							
LBA(39:32)	Reserved							
LBA(23:16)	Reserved							
LBA(47:40)	Reserved							
DEVICE(7:0)	na	RS1	RS2	na	PORTNUM(3:0)			
COMMAND(7:0)	**E4h**							

Figure 446 – READ PORT MULTIPLIER command definition

Field Definitions

REGNUM Set to number of register to read.

RS1 Register Specific 1 (RS1) bit, this bit is register specific.
Phy event counter usage, used to determine access to Phy event counter identifier or value. If cleared to zero upon a read to a Phy event counter register, the Port Multiplier shall return the Phy event counter identifier for that register. If set to one, the Port Multiplier shall return the Phy event counter value for that register, up to 64 bits in length.
All other usage, this bit shall be treated as na on all READ PORT MULTIPLIER commands to registers other than Phy event counter registers.

RS2 Register Specific 2 (RS2) bit, this bit is register specific.
Phy event counter usage, used in reads to Phy event counter values. If set to one upon a read to a Phy event counter register, the Port Multiplier shall return the value of the counter, followed by a reset of the counter value for the register supplied in the REGNUM field.
All other usage, this bit shall be treated as na on all READ PORT MULTIPLIER commands to registers other than Phy event counter registers.

PORTNUM Set to the port address that has register to be read.

16.5.1.3 Success outputs

Upon successful completion, the Port Multiplier registers shall be returned as described in Figure 447.

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	00h							
COUNT(7:0)	VALUE(7:0)							
COUNT(15:8)	VALUE(39:32)							
LBA(7:0)	VALUE(15:8)							
LBA(31:24)	VALUE(47:40)							
LBA(15:8)	VALUE(23:16)							
LBA(39:32)	VALUE(55:48)							
LBA(23:16)	VALUE(31:24)							
LBA(47:40)	VALUE(63:56)							
DEVICE(7:0)	Reserved							
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	0	0	ERR

Figure 447 – READ PORT MULTIPLIER success status result values

Field Definitions

VALUE Set to the 64 bit value read from the register.

BSY 0

DRDY 1

DF 0

DRQ 0

ERR 0

16.5.1.4 Error outputs

Upon encountering an error, the Port Multiplier shall set the Status register and the Error register as described in Figure 448.

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	Reserved					ABRT	REG	PORT
COUNT(7:0)	Reserved							
COUNT(15:8)	Reserved							
LBA(7:0)	Reserved							
LBA(31:24)	Reserved							
LBA(15:8)	Reserved							
LBA(39:32)	Reserved							
LBA(23:16)	Reserved							
LBA(47:40)	Reserved							
DEVICE(7:0)	Reserved							
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	0	0	ERR

Figure 448 – READ PORT MULTIPLIER error status result values

Field Definitions

ABRT 0

REG Set to one if the register specified is invalid.

PORT Set to one if the port specified is invalid.

BSY 0

DRDY 1

DF 0

DRQ 0

ERR 1

16.5.2 WRITE PORT MULTIPLIER

16.5.2.1 WRITE PORT MULTIPLIER overview

The WRITE PORT MULTIPLIER command (see Figure 449) is used to write a register on a Port Multiplier. The WRITE PORT MULTIPLIER command shall be issued to the control port.

16.5.2.2 Inputs

Field	7	6	5	4	3	2	1	0
FEATURES(7:0)	REGNUM(7:0)							
FEATURES(15:8)	REGNUM(15:8)							
COUNT(7:0)	VALUE(7:0)							
COUNT(15:8)	VALUE(39:32)							
LBA(7:0)	VALUE(15:8)							
LBA(31:24)	VALUE(47:40)							
LBA(15:8)	VALUE(23:16)							
LBA(39:32)	VALUE(55:48)							
LBA(23:16)	VALUE(31:24)							
LBA(47:40)	VALUE(63:56)							
DEVICE(7:0)	na	RS1	na	na	PORTNUM(3:0)			
COMMAND(7:0)	**E8h**							

Figure 449 – WRITE PORT MULTIPLIER command definition

Field Definitions

REGNUM Set to number of register to write.

VALUE Set to the 64 bit value to write to the register.

RS1 Register Specific 1 (RS1) bit, this bit is register specific.

PORTNUM Set to the port address that has register to be written.

Phy event counter usage of the RS1 bit, the WRITE PORT MULTIPLIER command may not be used to write to a Phy event counters' identifier value. If this bit is set to one during a write to a Phy event counter register, the counter addressed by the REGNUM field shall be written with Value. If cleared to zero during a write to a Phy event counter register, the Port Multiplier shall return error status with the ERR bit set to one, and the BSY bit and the DRQ bit cleared to zero in the Status field of the FIS. The ABRT bit shall also be set to one in the ERROR field.

All other usage of the RS1 bit, this bit shall be treated as na on all READ PORT MULTIPLIER commands to registers other than Phy event counter registers.

16.5.2.3 Success outputs

Upon successful completion, the Port Multiplier shall write the specified value to the specified register (see Figure 450).

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	00h							
COUNT(7:0)	Reserved							
COUNT(15:8)	Reserved							
LBA(7:0)	Reserved							
LBA(31:24)	Reserved							
LBA(15:8)	Reserved							
LBA(39:32)	Reserved							
LBA(23:16)	Reserved							
LBA(47:40)	Reserved							
DEVICE(7:0)	Reserved							
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	0	0	ERR

Figure 450 – WRITE PORT MULTIPLIER success status result values

Field Definitions

BSY 0

DRDY 1

DF 0

DRQ 0

ERR 0

16.5.2.4 Error outputs

Upon encountering an error, the Port Multiplier shall set the Status register and the Error register as described for Figure 451.

Field	7	6	5	4	3	2	1	0
ERROR(7:0)	Reserved					ABRT	REG	PORT
COUNT(7:0)	Reserved							
COUNT(15:8)	Reserved							
LBA(7:0)	Reserved							
LBA(31:24)	Reserved							
LBA(15:8)	Reserved							
LBA(39:32)	Reserved							
LBA(23:16)	Reserved							
LBA(47:40)	Reserved							
DEVICE(7:0)	Reserved							
STATUS(7:0)	BSY	DRDY	DF	na	DRQ	0	0	ERR

Figure 451 – WRITE PORT MULTIPLIER error status result values

Field Definitions

ABRT 0

REG Set to one if the register specified is invalid.

PORT Set to one if the port specified is invalid.

BSY 0

DRDY 1

DF 0

DRQ 0

ERR 1

16.5.3 Interrupts

The Port Multiplier shall generate an interrupt in the status response to a command for the control port by setting the Interrupt bit to one in the Register Device to Host FIS. The Port Multiplier shall not generate an interrupt in response to a software reset for the control port.

16.6 Controlling PM Port value and interface power management

The host controller needs to provide a means that software may set the PM Port field in all transmitted FISes. This capability may be exposed to software by supporting the PMP field in the SControl register as defined in 14.2.4. In addition a means that software may cause a specific device port to transition to a low power management state needs to be provided. This capability may be exposed to software by supporting the SPM field in the SControl register as defined in 14.2.4.

16.7 Switching types (informative)

16.7.1 Switching types overview (informative)

The host may use two different switching types depending on the capabilities of the host controller. If the host controller supports hardware context switching based on the value of the PM Port field in a received FIS, then the host may have commands outstanding to multiple devices at the same time. This switching type is called FIS-based switching (see 16.7.3). If the host controller does not

support hardware context switching based on the value of the PM Port field in a received FIS, then the host shall have commands outstanding to one device at any point in time. This switching type is called command-based switching (see 16.7.2). The Port Multiplier's operation is the same regardless of the switching type used by the host.

16.7.2 Command-based switching

Host controllers that do not support hardware context switching utilize a switching type called command-based switching. To use command-based switching, the host controller has commands outstanding to only one device at any point in time. By only issuing commands to one device at a time, the result is that the Port Multiplier only delivers FISes from that device.

A host controller may support command-based switching by implementing the Port Multiplier Port (PMP) field in the SControl register as detailed in 14.2.4. In order to use this mechanism, host software sets the PMP field appropriately before issuing a command to a device connected to the Port Multiplier. If host software had completed the commands with a particular device port, it modifies the PMP field before issuing commands to any other device port. The PMP field shall be set to the control port if host software issues commands to the Port Multiplier itself (e.g., READ PORT MULTIPLIER or WRITE PORT MULTIPLIER).

16.7.3 FIS-based switching

16.7.3.1 FIS-based switching overview

Host controllers that support hardware context switching may utilize a switching type called FIS-based switching. FIS-based switching allows the host controller to have commands outstanding to multiple devices at any point in time. If commands are outstanding to multiple devices, the Port Multiplier may deliver FISes from any device with commands outstanding.

16.7.3.2 Host controller requirements

To support FIS-based switching, the host controller shall have context switching support. The host controller needs to provide a means for exposing a programming interface for up to 16 devices on a single port. The context switching support needs to comprehend not only Control Block register context, but also DMA engine context and the SActive register. The host controller needs to be able to update context for a particular device even if the programming interface for that device is not currently selected by host software.

The host controller needs to fill in the PM Port field in hardware assisted FIS transmissions.

EXAMPLE - If the host controller receives a DMA Activate from a device, it needs to construct a Data FIS with the PM Port field set to the value in the received DMA Activate FIS.

Refer to the requirements given in 16.3.3.7 for a mechanism to reduce the complexity of host context switching.

17 Port Selector

17.1 Port Selector overview

A Port Selector allows two different host ports to connect to the same device in order to create a redundant path to that device. In combination with RAID, the Port Selector allows system providers to build fully redundant solutions. The upstream ports of a Port Selector may also be attached to a Port Multiplier to provide redundancy in a more complex topology. A Port Selector may be thought of as a simple multiplexer as shown in Figure 452.

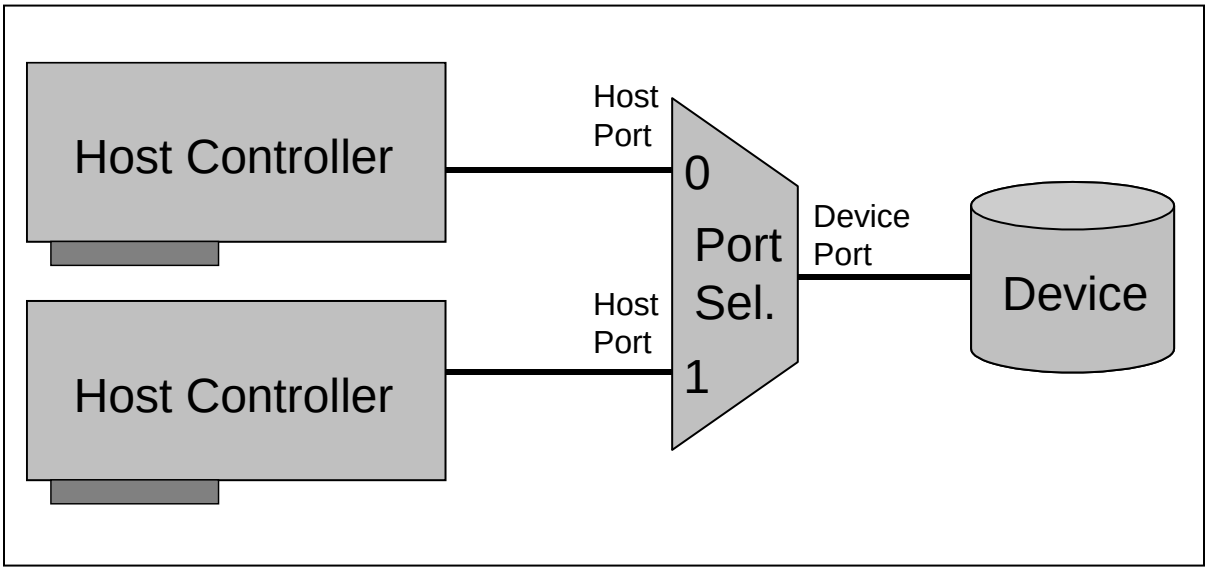


Figure 452 – Port Selector overview

Exactly two host connections are provided by a Port Selector. Only one of the two host ports is active at a time – this specification does not define mechanisms for active/active solutions. Cascading Port Selectors to each other is not supported.

17.2 Example applications

One example application of a Port Selector, as shown in Figure 453, is to provide a means for redundant access to a device. This configuration, along with RAID, allows a system with no single point of failure to be built. Typically the Port Selector is possible to be packaged in the hard drive carrier to create a single serviceable unit in case the hard drive failed. The total system is possible to consist of two hosts each connected to a RAID array where each device in the system had a Port Selector attached that was connected to each host. One host is able to be considered the live host and the other host may be the spare. In this configuration, the live host has access to all of the devices and the spare host may take over access to the devices if the live host has a failure.

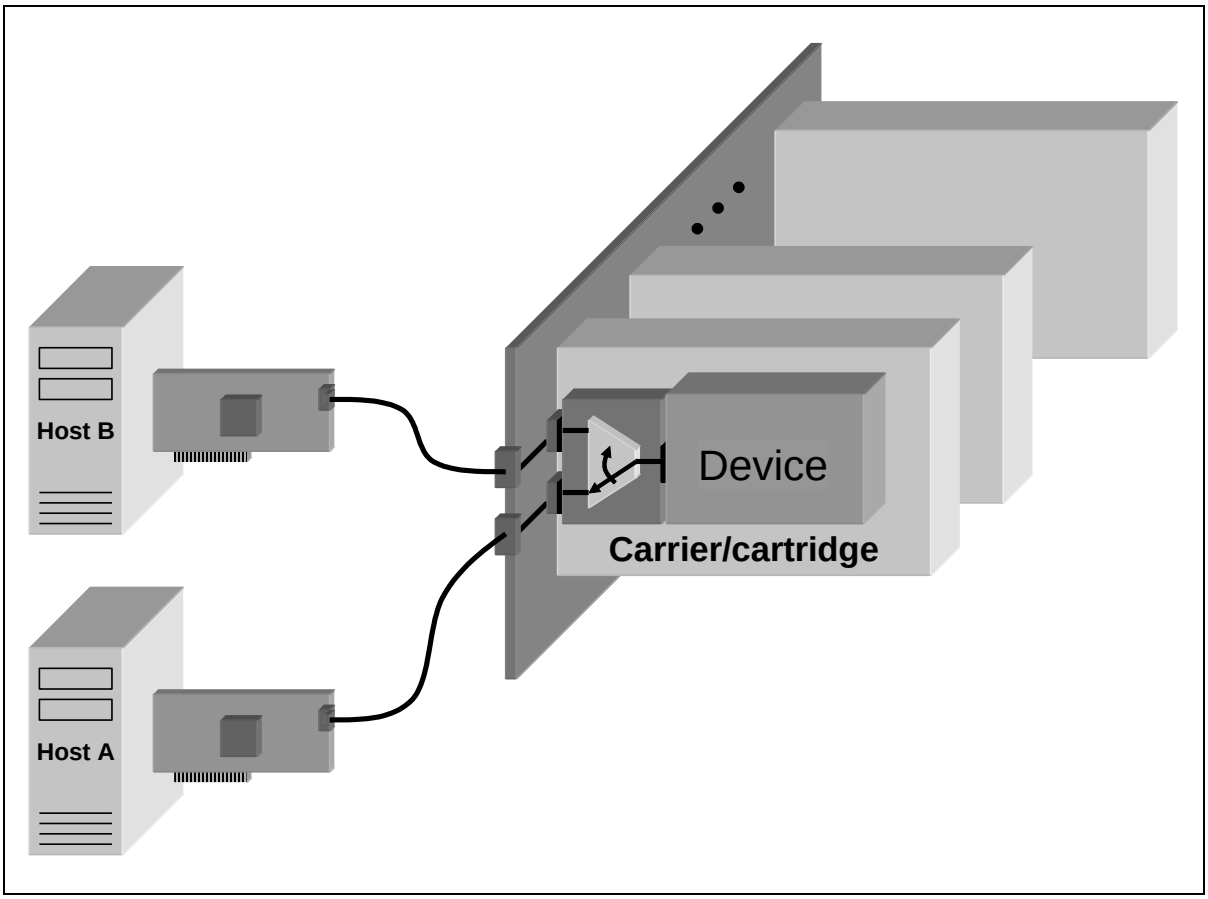


Figure 453 – Example failover application with two hosts

17.3 Port Selector introduction

Port Selector is a mechanism that allows two different host ports to connect to the same device in order to create a redundant path to that device. Only one host connection to the device is active at a time. Effective use of a Port Selector requires coordinated access to the device between the two host ports. The host(s) shall coordinate to determine which host port should be in control of the device at any given point in time. Definition of the coordination mechanism or protocol is beyond the scope of this specification.

Once the host(s) determines which host port should be in control of the device, the host to be made active takes control of the device by selecting its host port to be active. The active host selects a port to be active by using either a protocol-based or side-band port selection mechanism. A side-band port selection mechanism may be as simple as a hardware select line (i.e., pulled high to activate one host port and low to activate the other). The side-band port selection mechanism is outside the scope of this specification. A protocol-based port selection mechanism uses the Serial ATA protocol to cause a switch of active port. This specification defines a protocol-based port selection mechanism that uses a particular Morse coding of COMRESET signals to cause a switch of active host port. A Port Selector shall only support one selection mechanism at any point in time. The externally visible behavior of a Port Selector is the same regardless of whether a protocol-based or side-band port selection mechanism is used.

A Port Selector that supports protocol-based port selection is detected in the signal path if the optional presence detection feature is supported by the Port Selector and the host has an enhanced

SError register that latches this event. The detection mechanism for a Port Selector that supports side-band port selection is outside the scope of this specification.

17.4 Active Port Selection

17.4.1 Active Port Selection overview

The Port Selector has a single active host port at a time. The Port Selector shall support one of two mechanisms (i.e., protocol-based port selection (see 17.4.2) or side-band port selection (see 17.4.3)) for determining which of two host ports is active.

Whether a protocol-based or side-band port selection mechanism is used, the Port Selector shall exhibit the behavior defined within this specification.

After selection of a new active port, the device is in an unknown state. The device may have active commands outstanding from the previous active host that need to be flushed. After an active port switch has been performed the active host should issue a COMRESET to the device to ensure that the device is in a known state.

17.4.2 Protocol-based Port Selection

17.4.2.1 Protocol-based Port Selection overview

Protocol-based port selection is an active port selection mechanism that uses a sequence of Serial ATA OOB Phy signals to select the active host port. A Port Selector that supports protocol-based port selection shall have no active host port selected upon power-up. The first COMRESET or COMWAKE received over a host port shall select that host port as active. The host may then issue explicit switch signals to change the active host port.

Reception of the protocol-based port selection signal on the inactive host port causes the Port Selector to deselect the currently active host port and select the host port that the selection signal is received. The protocol-based port selection signal is defined such that it is generated using the Status and Control registers and the protocol-based port selection signal is received and decoded without the need for the Port Selector to include a full Link or Transport layer (i.e., direct Phy detection of the signal).

17.4.2.2 Port selection signal definition

The port selection signal is based on a pattern of COMRESET OOB signals transmitted from the host to the Port Selector. As illustrated in Figure 454, the Port Selector shall qualify only the timing from the assertion of a COMRESET signal to the assertion of the next COMRESET signal in detecting the port selection signal.

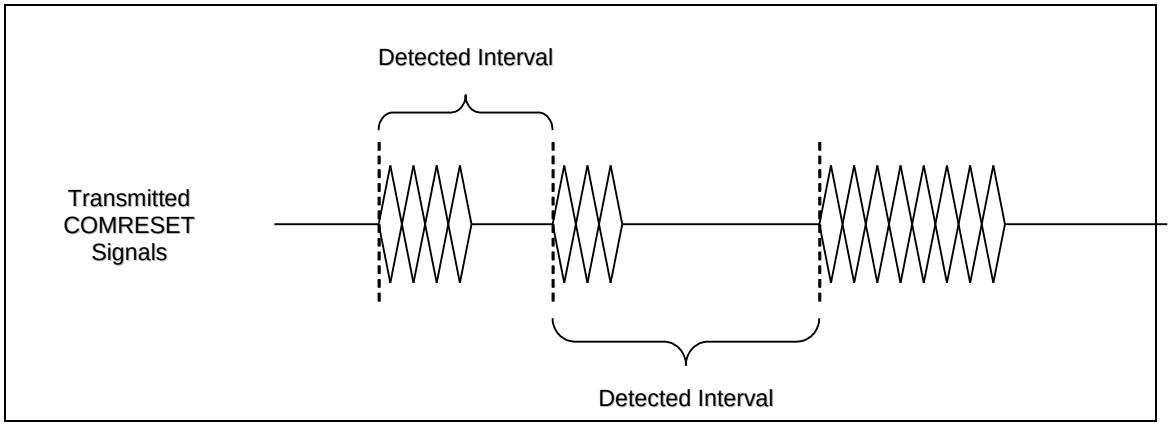


Figure 454 – Port selection signal based on assertion of COMRESET to assertion of following COMRESET

The port selection signal is defined as a series of COMRESET signals with the timing from the assertion of one COMRESET signal to the assertion of the next as defined in Table 137 and illustrated in Figure 455. The Port Selector shall select the port, if inactive, on the negation of COMRESET after receiving two complete back-to-back sequences with specified inter-burst spacing over that port (i.e., two sequences of two COMRESET intervals comprising a total of five COMRESET bursts with four inter-burst delays). Specifically, after receiving a valid port selection signal, the Port Selector shall not select that port to be active until the entire fifth COMRESET burst is no longer detected. The Port Selector is only required to recognize the port selection signal over an inactive port. Reception of COMRESET signals over an active port is propagated to the device without any action taken by the Port Selector, even if the COMRESET signals constitute a port selection signal. This may result in multiple device resets.

The timings detailed in Table 137 shall be independent of the signaling speed used on the link.

EXAMPLE - The inter-reset timings are the same for links using Gen1 or Gen2 speeds.

Table 137 – Port selection signal inter-reset timing requirements

	Nom	Min	Max	Units	Comments
T1	2.0	1.6	2.4	ms	Inter-reset assertion delay for first event of the selection sequence
T2	8.0	7.6	8.4	ms	Inter-reset assertion delay for the second event of the selection sequence

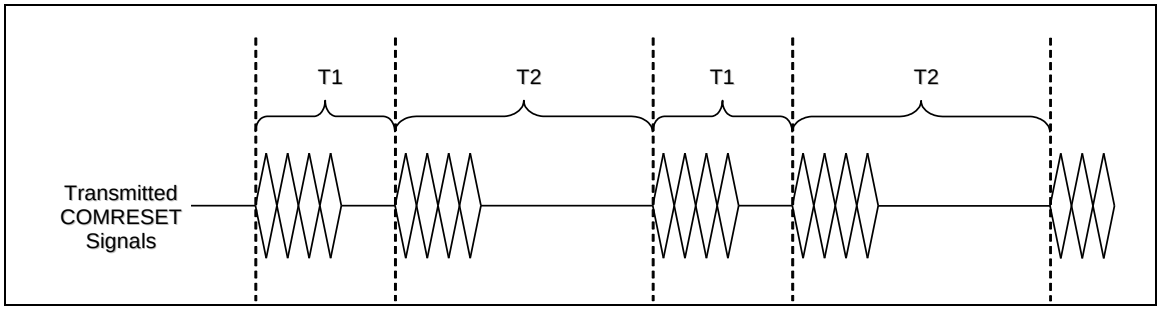


Figure 455 – Complete port selection signal consisting of two sequences with requisite inter-reset spacings

The interpretation and detection of the COMRESET signal by the Port Selector is in accordance with the Phy layer electrical specifications, (i.e., the COMRESET signal is detected upon receipt of the fourth burst that complies with the COMRESET signal timing definition). The inter-reset timings referred to here for the port selection signal are from the detection of a valid COMRESET signal to the next detection of such a signal, and are not related to the bursts that comprise the COMRESET signal itself.

17.4.2.3 Presence detection

17.4.2.3.1 Presence detection overview

Presence detection is the ability for a host to detect that a Port Selector is present on a port. If a Port Selector supports presence detection capabilities, a host shall be able to determine whether a Port Selector is connected to a host port. The host may determine this regardless of whether the Port Selector port that it is connected to is the active or inactive link.

Presence detection capabilities are defined for Port Selectors utilizing protocol-based port selection only. Systems utilizing side-band port selection shall be preconfigured for side-band port selection and therefore those systems may also be preconfigured to support presence detection. Presence detection is optional for protocol-based Port Selectors.

17.4.2.3.2 Host port Phy state machine enhancements

If presence detection is supported, the Port Selector host port Phy state machine as defined in the 8.4.3 shall be modified as shown (see Figure 456). A Port Selector shall remain in the DR_PS_Wait state if the Phy is offline and presence detection is enabled.

DP1: DR_Reset [a]	Interface quiescent	
1. COMRESET not detected and power-on reset negated and presence detection not enabled and Phy not offline	→	DR_COMINIT
2. COMRESET not detected and power-on reset negated and presence detection enabled	→	DR_PS_Presence
3. COMRESET detected or power-on reset asserted	→	DR_Reset
[a] This state is entered asynchronously any time in response to power-on reset or receipt of a COMRESET signal from the host.		

DP12: DR_PS_Presence	Transmit COMWAKE	
1. Phy online	→	DR_COMINIT
2. Phy offline	→	DR_PS_Wait

DP13: DR_PS_Wait	Interface quiescent	
1. Phy online	→	DR_COMINIT
2. Phy offline	→	DR_PS_Wait

Figure 456 – Host port Phy state machine enhancements

17.4.2.3.3 Host Phy initialization state machine impact (informative)

A host connected to a Port Selector performing presence detection receives a COMWAKE signal while in the HP2: HR_AwaitCOMINIT state of the Phy state machine as defined in 8.4.2. This state is insensitive to receiving a COMWAKE and only performs a transition to a new state if COMINIT is received. Host designers should ensure that their implementations are insensitive to receiving a COMWAKE in this state. If the host design is capable of latching the reception of COMWAKE while in this state, the host should expose that reception using the SError register enhancement as detailed in 17.4.2.3.4.

17.4.2.3.4 SError register enhancement for presence detection

The Serial ATA interface error register (SError) as defined in 14.2.3 includes indications for various events that may have occurred on the interface (e.g., a change in the PHYRDY state, detection of disparity, errors). In order to facilitate a means for notifying host software that a Port Selector presence detection signal was received, the A bit in the DIAG field of the SError register is set to one if COMWAKE is received while the host is in state HP2: HR_AwaitCOMINIT.

17.4.2.4 Host transmission considerations (informative)

In order to ensure the port selection signal is reliably conveyed to the Port Selector, the host should account for any other interface activity that may interfere with the transmitted COMRESET port selection sequence.

EXAMPLE - If the host periodically issues a COMRESET signal as part of a hardware-polled device presence detection mechanism, a periodic COMRESET signal is able to occur during the port selection signaling sequence, thereby corrupting the port selection sequence.

In order to avoid such interactions, the host may elect to continually transmit the port selection sequence while monitoring the Phy status in the associated superset Status and Control register. If the port selection signal is recognized by the Port Selector and has taken effect, then the host detects a change in the PHYRDY status since the associated port is activated and communications with it are established. It is recommended that the host check the PHYRDY signal immediately

before issuing each COMRESET burst in the protocol-based selection signal and only issue the next COMRESET burst if PHYRDY is not asserted.

17.4.3 Side-band Port Selection

The active host port may be selected by a side-band mechanism. Side-band port selection uses a mechanism outside of the Serial ATA protocol for selecting which host port is active.

EXAMPLE - One example of a side-band port selection mechanism is a hardware select line.

The side-band selection mechanism used is outside the scope of this specification. A Port Selector that supports side-band port selection shall exhibit the behavior defined within this specification.

17.4.4 Behavior during a change of active port

17.4.4.1 Behavior during a change of active port overview

During a change of active port, the previous host connection is broken and all internal state other than the active host port is initialized before the connection with the new active host is made. If a new active host port is selected, the Port Selector shall perform the following procedure:

1) the Port Selector shall stop transmitting and enter the quiescent power condition on the previously active host port Phy (i.e., now the inactive host port);
2) the Port Selector shall initialize all internal state other than the state of the selection bit for the active host port;
3) the Port Selector shall enter the active power condition on the new active host port; and
4) the Port Selector shall allow OOB and normal traffic to proceed between the new active host port and the device.

17.4.4.2 Device state after a change of active port (informative)

A Port Selector may support an orderly switch to a new active host port. A Port Selector that supports an orderly switch ensures that primitive alignment with the device Phy is maintained during the switch to the new active host port. Maintaining primitive alignment ensures that PHYRDY remains present between the Port Selector and the device throughout the switch to the new active host port.

After selection of a new active port, the device may be in an unknown state. The device may have active commands outstanding from the previously active host port that need to be aborted. The new active host should issue a COMRESET to the device in order to return the device to a known state.

17.5 Behavior and policies

17.5.1 Control state machine

The Port Selector Control state machine is based on a model of a Port Selector consisting of three Serial ATA Phys interconnected and controlled by an overall control logic block as depicted in Figure 457. For convenience, the three ports of a Port Selector are abbreviated “A,” “B,” and “D” corresponding to Host Port A, Host Port B, and Device Port respectively.

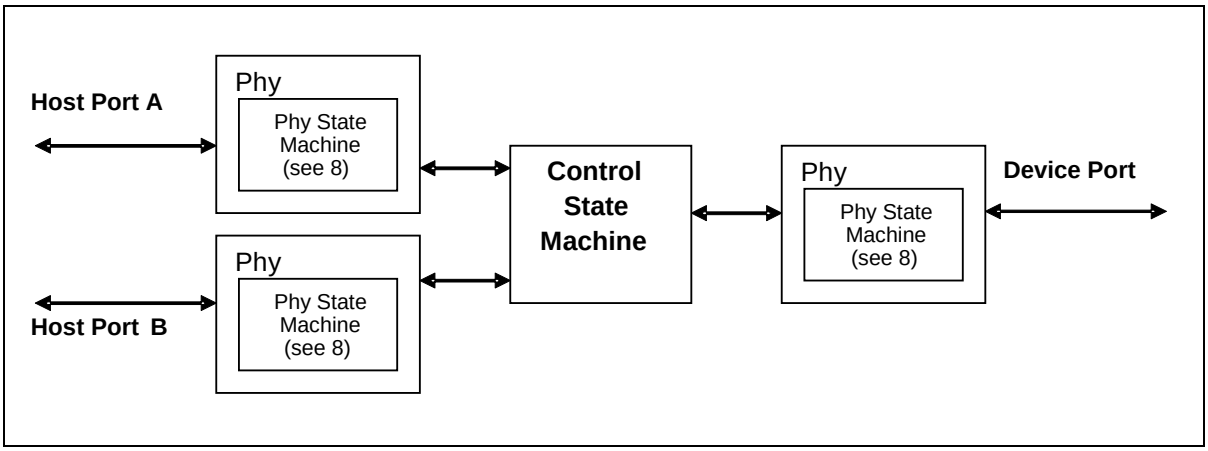


Figure 457 – Control state machine

The Phys depicted in Figure 457 are presumed to have the basic capabilities and controls indicated in Figure 458. Figure 458 is a variant of Figure 161.

The differences are:

a) loopback controls were removed since those controls were not relevant in this state machine;
b) explicit ONLINE and OFFLINE signals were added, including a register latch for these signals so that the signals do not need to be asserted in every state. The SControl register specifies a mechanism for the host to put the Phy in offline mode. Therefore, it is reasonable to expect that the Phy has signals ONLINE and OFFLINE that may be utilized; and
c) a PORTSELECT signal was added. A Port Selector using protocol-based port selection shall set the PORTSELECT signal to the output of the Sequence Detect block. The Sequence Detect block is asserted if the protocol-based selection signal is received, otherwise the signal is negated. A Port Selector using side-band port selection shall set the PORTSELECT signal if a change in active port is requested.

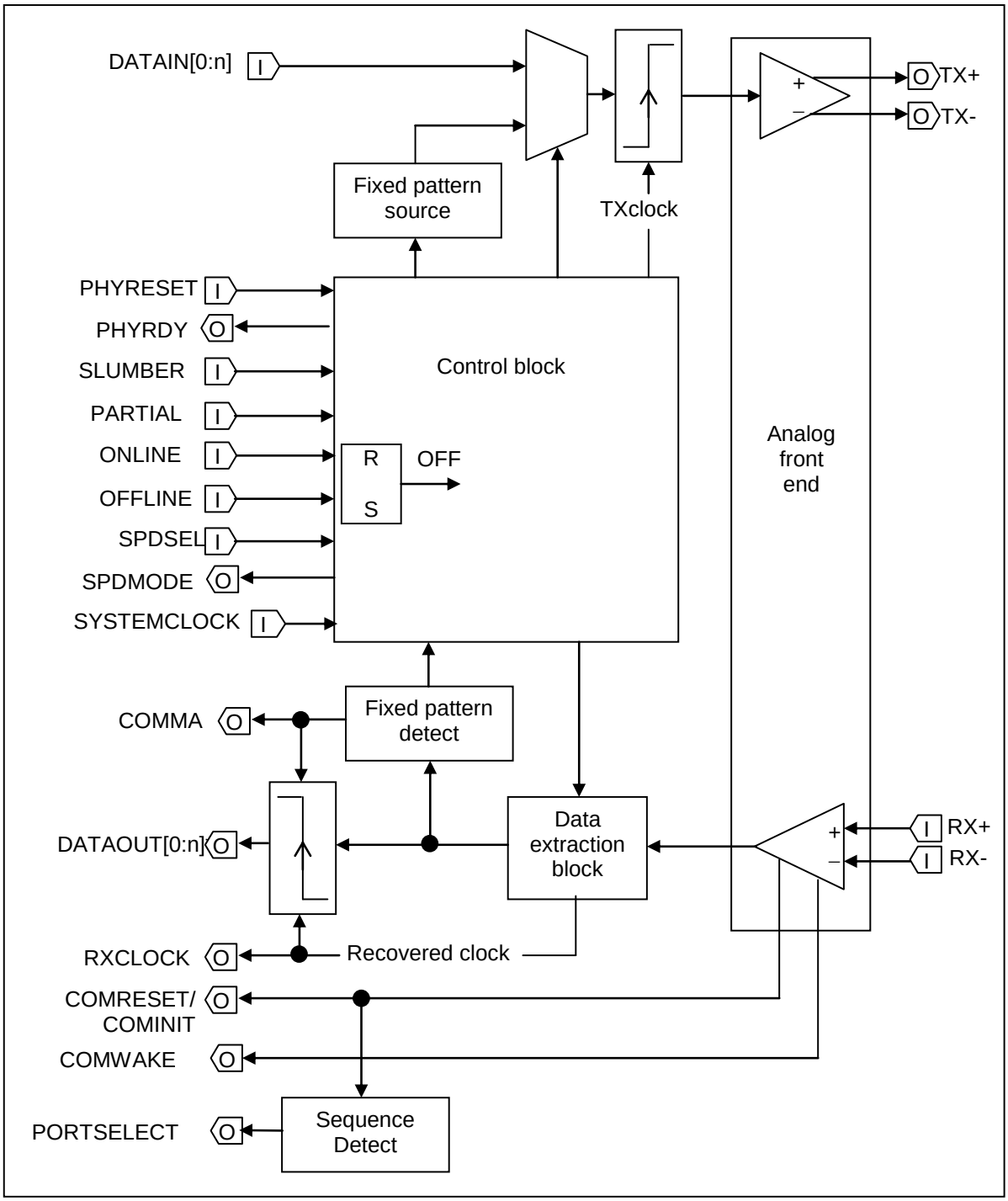


Figure 458 – Phy block diagram

The following state machines specifies the behavior of the Port Selector control logic block (see Figure 459). The Port Selector shall have the externally visible behavior described by this state machine.

Reception of a COMRESET signal from the selected host port shall unconditionally force the control state machine to transition to state PS15: ResetDevice. Reception of a COMINIT signal from the device shall unconditionally force the control state machine to transition to state PS14: ResetHost

if not in the ResetDevice state. For the sake of brevity, this implied transition has been omitted from most states.

PS1: PORReset [a]	Set internal PS state to initial conditions including MaxNegSpeed=Max and SelHostPort=none. Assert A.OFFLINE. Assert B.OFFLINE. Assert D.OFFLINE.	
1. Power-on reset condition and explicit reset request negated and Mode = SideBand	→	SetHostPortSideBand
2. Power-on reset condition and explicit reset request negated and Mode = ProtocolBased	→	AwaitHostSelection
3. Power-on reset or explicit reset request asserted	→	PORReset
[a] This state is entered asynchronously any time in response to power-on reset or an explicit reset request. An explicit reset request is a reset line/button on the Port Selector itself, not a COMRESET signal from a host.		

PS2: SetHostPortSideBand	Set SelHostPort based on value of side-band selection signal.	
1. COMRESET received from SelHostPort		ResetDevice
2. COMINIT received from D and COMRESET not received from SelHostPort		ResetHost
3. COMRESET not received from SelHostPort and COMINIT not received from D	→	SetHostPortSideBand

PS3: ComInitPropToBoth	Assert A.ONLINE. Assert A.PHYRESET. Assert B.ONLINE. Assert B.PHYRESET.	
1. Unconditional	→	AwaitHostSelection

PS4: AwaitHostSelection		
1. COMINIT received from D and (COMRESET or COMWAKE) not received from (A or B)	→	ComInitPropToBoth
2. (COMRESET or COMWAKE) received from A	→	SelectA
3. (COMRESET or COMWAKE) not received from A and (COMRESET or COMWAKE) received from B	→	SelectB
4. No OOB signal detected	→	AwaitHostSelection

PS5: SelectA	Assert B.OFFLINE. Set SelHostPort=A. Assert SelHostPort.ONLINE. Assert D.ONLINE. Assert SelHostPort.PHYRESET. Assert D.PHYRESET.	
1. Unconditional	→	WaitforComm

PS6: SelectB	Assert A.OFFLINE. Set SelHostPort=B. Assert SelHostPort.ONLINE. Assert D.ONLINE. Assert SelHostPort.PHYRESET. Assert D.PHYRESET.	
1. Unconditional	→	WaitforComm

Figure 459 – Port Selector state machine (part 1 of 3)

PS7: WaitForComm		
1. SelHostPort.PHYRDY and D.PHYRDY and no change in PORTSELECT signal	→	CheckSpeeds
2. (!SelHostPort.PHYRDY or !D.PHYRDY) and timeout [a] not exceeded and no change in PORTSELECT signal	→	WaitForComm
3. (!SelHostPort.PHYRDY or !D.PHYRDY) and timeout [a] exceeded and no change in PORTSELECT signal	→	PowerManageCheck
4. PORTSELECT signal received for non-selected host port	→	ChangePort
[a] The timeout is vendor specific but shall be larger than 1 760 us		

PS8: CheckSpeeds [a]		
1. SelHostPort.SPDMODE = D.SPDMODE	→	Online
2. SelHostPort.SPDMODE > D.SPDMODE	→	SetDeviceSpeed
3. SelHostPort.SPDMODE < D.SPDMODE	→	SetHostSpeed
[a] A larger value for the SPDMODE signal shall indicate a higher speed than a smaller value for SPDMODE.		

PS9: Online	Transfer DATA received on D to SelHostPort. Transfer DATA received on SelHostPort to D.	
1. PORTSELECT signal received for non-selected host port	→	ChangePort
2. SelHostPort.PHYRDY negated or D.PHYRDY negated	→	PowerManageCheck

PS10: SetDeviceSpeed	Set MaxNegSpeed=D.SPDMODE	
1. Unconditional	→	ReComm

PS11: SetHostSpeed	Set MaxNegSpeed=SelHostPort.SPDMODE	
1. Unconditional	→	ReComm

PS12: ReComm	Assert D.PHYRESET	
1. Unconditional	→	WaitForComm

PS13: ChangePort	Assert SelHostPort.OFFLINE. Assert !SelHostPort.ONLINE. Set SelHostPort=!SelHostPort. Set MaxNegSpeed=Max.	
1. Unconditional	→	WaitForComm

PS14: ResetHost [a]	Assert SelHostPort.ONLINE. Assert D.ONLINE. Assert SelHostPort.PHYRESET	
1. Unconditional	→	WaitForComm
[a] This state is entered unconditionally upon receipt of D.COMINIT if SelHostPort != none and the Port Selector is not in state ResetDevice.		

Figure 459 – Port Selector state machine (part 2 of 3)

PS15: ResetDevice [a]	Assert SelHostPort.ONLINE. Assert D.ONLINE. Assert D.PHYRESET.		
1. Unconditional		→	WaitForComm
[a] This state is entered unconditionally upon receipt of SelHostPort.COMRESET if SelHostPort != none.			

PS16: PowerManageCheck			
1. Port Selector determined that low power state entered is SLUMBER		→	PowerManageSlumber
2. Port Selector has not determined that low power state entered is SLUMBER		→	PowerManagePartial

PS17: PowerManagePartial	Assert SelHostPort.Partial. Assert D.Partial.		
1. SelHostPort.COMWAKE detected or D.COMWAKE detected		→	WaitForComm
2. PORTSELECT signal received for non-active host port		→	ChangePort
3. COMWAKE not received and no change in PORTSELECT signal		→	PowerManagePartial

PS18: PowerManageSlumber	Assert SelHostPort.Slumber. Assert D.Slumber.		
1. SelHostPort.COMWAKE detected or D.COMWAKE detected		→	WaitForComm
2. PORTSELECT signal received for non-active host port		→	ChangePort
3. COMWAKE not received and no change in PORTSELECT signal		→	PowerManageSlumber

Figure 459 – Port Selector state machine (part 3 of 3)

17.5.2 BIST support

A Port Selector is not required to support the BIST Activate FIS. The resultant behavior of sending a BIST Activate FIS through a Port Selector is undefined.

17.5.3 Flow control signaling latency

The Port Selector shall satisfy the flow control signaling latency as defined in 9.5.9. The Port Selector shall ensure that the flow control signaling latency is met on a per link basis.

Specifically, the Port Selector shall ensure that the flow control signaling latency is met between:

a) the Port Selector active host port and the host it is connected to; and
b) the Port Selector device port and the device it is connected to.

The Port Selector shall not reduce the flow control signaling latency budget of the active host it is connected to or the device it is connected to.

17.5.4 Power management

17.5.4.1 Power management overview

The Port Selector shall maintain the active host port across power management events and only allow an active host port change after receiving a valid port selection signal.

The Phy on the inactive host port shall be in the quiescent power condition. Upon detecting that the PHYRDY signal is not present for the active host port or device port, the Port Selector shall place that Phy in a quiescent power condition.

If the PHYRDY signal is not present between the device and the Port Selector, the Phy connected to the active host port shall enter the quiescent power condition and squelch the Phy transmitter. If the PHYRDY signal is not present between the active host and the Port Selector, the Phy connected to the device shall enter the quiescent power condition and squelch the Phy transmitter. During these periods while PHYRDY is not present, OOB signals shall still be propagated between the active host and the device to ensure that communication is established.

If the Port Selector is able to determine that the active host and device negotiated a Slumber power management transition, the Port Selector may recover from the quiescent power condition in the time defined by the Slumber power state. If the Port Selector is not able to determine the power state entered by the host and device, the Port Selector shall recover from the quiescent power condition in the time defined by the Partial power state.

17.5.4.2 Wakeup budget

The wakeup budget out of Partial or Slumber may increase if a Port Selector is connected to a device. If the active host Phy comes out of low power condition, the Port Selector active host Phy may wakeup before causing the Port Selector device Phy to wakeup that in turn wakes the device. The host shall allow the device at least 20 us to wakeup from the Partial power management state.

17.5.5 OOB Phy signals

The Port Selector shall propagate COMRESET received from the active host to the device as specified in the Control State Machine as defined in 17.5.1. The Port Selector shall propagate COMINIT received from the device to the active host port as specified in the Control State Machine as defined in 17.5.1. If no active host port is selected, the Port Selector shall propagate COMINIT received from the device to the active host port as specified in the Control State Machine as defined in 17.5.1. The Port Selector is allowed to delay delivery of propagated OOB signals.

The Port Selector shall not respond to COMRESET signals received over the inactive host port. The inactive host port Phy shall remain in the quiescent power condition if COMRESET is received over the inactive host port.

17.5.6 Hot plug

The Port Selector shall only generate a COMINIT over a host port if a COMINIT signal is received from the device or as part of an active speed negotiation as specified in the Control State Machine as defined in 17.5.1. If a device connected to a Port Selector is hot plugged, the device issues a COMINIT sequence as part of its normal power-up sequence in accordance with the Phy layer state machine. The Port Selector shall propagate the COMINIT over the active host port (or both host ports if both host ports are inactive). If the host detects the COMINIT signal, the host then interrogates the port to determine whether a device is attached.

If a device connected to a Port Selector is hot unplugged, the Port Selector shall squelch the transmitter for the active host port as defined in 17.5.4. The active host should then determine that the PHYRDY signal is no longer present and therefore that a device is no longer present.

17.5.7 Speed negotiation

Speed is negotiated on a per link basis. Specifically, the Port Selector shall negotiate speed between:

a) the Port Selector active host port and the host that it is connected; and
b) the Port Selector device port and the device that it is connected.

The Port Selector starts speed negotiation at the highest speed that it supports. The Port Selector then negotiates speed on each link to the appropriate supported speed. After negotiating speed on the active host link and on the device link, the Port Selector shall check whether the two speeds match. If the speeds do not match, the Port Selector limits the maximum speed it supports to the lower of the two speeds negotiated. Then the Port Selector forces speed to be renegotiated to reach a common speed.

17.5.8 Spread Spectrum Clocking

The Port Selector shall support Spread Spectrum Clocking receive on all of its ports. The Port Selector may support spread spectrum transmit. It is recommended that a configuration jumper be used to enable/disable Spread Spectrum Clocking if it is settable. There is no means within the Serial ATA protocol provided to enable/disable Spread Spectrum Clocking if it is statically configurable.

If Spread Spectrum Clocking is used, the spreading domain between the host and the Port Selector is not required to be the same as the spreading domain between device and the Port Selector. The signals passing through a Port Selector may be re-spread.

17.6 Power-up and resets

17.6.1 Power-up

17.6.1.1 Power-up overview

Upon power-up, the Port Selector shall reset all internal state, including the active host port. This causes no active host port to be selected if protocol-based port selection is used.

17.6.1.2 Presence detection of Port Selector

For protocol-based port selection, presence detection may be performed using the optional mechanism according to 17.4.2.3. Presence detection for Port Selectors implementing side-band port selection is outside the scope of this specification.

17.6.2 Resets

17.6.2.1 COMRESET

If COMRESET is received over the active host port the Port Selector shall reset all internal state, the active host port shall remain unchanged, the maximum speed shall remain unchanged, and the COMRESET signal shall be propagated to the device. The Port Selector shall take no reset action upon receiving a COMRESET signal over the inactive host port except as defined in 17.4.2 if there is no active host port selected after power-up.

17.6.2.2 Software reset and DEVICE RESET

The Port Selector shall not reset in response to receiving a Software reset or the DEVICE RESET command.

17.7 Host implementation (informative)

17.7.1 Software method for protocol-based selection overview

The preferred software method for producing a protocol-based port selection signal is detailed in this section. Software for HBAs that implement the SControl and SStatus registers may use this method to create the protocol-based port selection signal.

17.7.2 Software method for protocol-based selection

If the Phy is left on for long periods during the generation of the sequence, a hardware based COMRESET polling algorithm may interfere and corrupt the sequence. This method tries to minimize any impact of host based COMRESET polling algorithms by leaving the Phy online for very short sequences during the creation of the signal. The procedure outlined is appropriate for any HBA design that has a COMRESET polling interval greater than 25 us.

The procedure is:

1) set Phy to offline by writing SControl.DET field to 4h;
2) set Phy to reset state by writing SControl.DET field to 1h;
3) wait 5 us to allow charging time for DC coupled Phy designs;
4) set Phy to online state by writing SControl.DET field to 0h;
5) wait 20 us to allow COMRESET burst to be transmitted to the device;
6) set Phy to offline by writing Scontrol.DET field to 4h;
7) wait 1.975 ms to satisfy T1 timing as specified in Table 137;
8) repeat steps 2 to 6;
9) wait 7.975 ms to satisfy T2 timing as specified in Table 137;
10) repeat steps 2 to 6;
11) wait 1.975 ms to satisfy T1 timing as specified in Table 137;
12) repeat steps 2 to 6;
13) wait 7.975 ms to satisfy T2 timing as specified in Table 137;
14) set Phy to reset state by writing SControl.DET field to 1h;
15) wait 5 us to allow charging time for DC coupled Phy designs;
16) set Phy to online state by writing SControl.DET field to 0h;
17) wait up to 10 ms for SStatus.DET field = 3h; and
18) if SStatus.DET field != 3h, go to step 1 to restart the process.

The procedure is also outlined in pseudocode on the following page.

```
//
// Continue to perform this procedure until SStatus.DET == 3.
//
while (SStatus.DET != 3)
{
SControl.DET = 4;          // Turn off Phy

//
// Mimic out the COMRESET bursts at the appropriate T1/T2 timing intervals.
//
for (i = 0; i < 4; i++)
{
       SControl.DET = 1;          // Place HBA in reset state
       Sleep(5);                  // Wait for Phy to charge, in microseconds
       SControl.DET = 0;          // Issue COMRESET
       Sleep(20);                 // Wait for COMRESET to be sent
       SControl.DET = 4;          // Turn off Phy

       if ((i == 0) || (i == 2))
{
Sleep(1975);        // Wait T1 time minus time already waited
       }
       else if ((i == 1) || (i == 3))
       {
Sleep(7975);        // Wait T2 time minus time already waited
       }
}

//
// Issue final COMRESET of the burst.
//
SControl.DET = 1;          // Place HBA in reset state
Sleep(5);                        // Wait for Phy to charge, in microseconds
SControl.DET = 0;          // Issue COMRESET


//
// Wait up to 10 milliseconds for PHYRDY.
//
for (i = 0; i < 10000; i++)
{
       if (SStatus.DET == 3)
       {
break;              // Stop procedure if SStatus.DET == 3
}
       Sleep(1);                  // Wait 1 microsecond
}
```

Appendix A. Sample code for CRC and scrambling (informative)

A.1 CRC calculation

A.1.1 CRC overview

The following section provides an informative implementation of the CRC polynomial. The example is intended as an aid in verifying a HDL implementation of the algorithm.

A.1.2 Maximum frame size

The 32 bit CRC used by Serial ATA may be shown to provide detection of two 10 bit errors up to a maximum frame size of 16 384 bytes. This provides for future expansion of FISes to a maximum of 64 bytes of fixed overhead while still permitting a maximum user data payload of 8 192 bytes.

A.1.3 Example code for CRC algorithm overview

The following code, written in C, illustrates an implementation of the CRC algorithm. A Register Host to Device FIS containing a PIO write command is used as example input. The CRC calculated is the check Dword appended to a transmitted serial stream immediately preceding EOF_P. The code displays both the resulting command FIS and the intermediate CRC polynomial values. To compile the code with the GNU tool chain, use the following command.

```
'gcc –o crc.exe crc.c'
```

This code is supplied for illustrative purposes only.

A.1.4 Example code for CRC algorithm

```
/****************************************************************************/
/*                                                                          */
/* crc.c                                                                    */
/*                                                                          */
/* This sample code reads standard in for a sequence of 32 bit values       */
/* formatted in hexadecimal with a leading "0x" (e.g. 0xDEADBEEF).  The     */
/* code calculates the Serial ATA CRC for the input data stream.  The       */
/* generator polynomial used is:                                            */
/*         32   26   23   22   16   12   11   10   8   7   5   4   2        */
/* G(x) = x  + x  + x  + x  + x  + x  + x  + x  + x + x + x + x + x + x + 1 */
/*                                                                          */
/* This sample code uses a parallel implementation of the CRC calculation   */
/* circuit that is suitable for implementation in hardware.  A block        */
/* diagram of the circuit being emulated is shown below.                    */
/*                                                                          */
/*                      +---+          +---+          +---+                 */
/*   Data_In --------->|   |          |   |          | R |                 */
/*                      | + |--------->| * |--------->| e |----+            */
/*               +----->|   |          |   |          | g |    |            */
/*               |      +---+          +---+          +---+    |            */
/*               |                                             |            */
/*               |                                             |            */
/*               +---------------------------------------------+            */
/*                                                                          */
/* The CRC value is initialized to 0x52325032 as defined in the Serial ATA  */
/* specification.                                                           */
/*                                                                          */
/****************************************************************************/

#include <stdlib.h>
#include <stdio.h>

main(argc,argv)
int argc;
char *argv[];
{
   int               i,
                     data_count;
   unsigned int      crc,
                     data_in;
   unsigned char     crc_bit[32],
                     new_bit[32];


   crc = 0x52325032;
   data_count = 0;

   while (scanf(" 0x%8x", &data_in) == 1)  {
      data_count++;
      /* Add the data_in value to the current value of the CRC held in the  */
      /* "register".  The addition is performed modulo two (XOR).           */
      crc ^= data_in;
      /* Expand the value of the CRC held in the register to 32 individual  */
      /* bits for easy manipulation.                                        */
      for (i = 0; i < 32; ++i)  {
```

```
   crc_bit[i] = (crc >> i) & 0x01;
}

/* The following 32 assignments perform the function of the box       */
/* labeled "*" in the block diagram above.  The new_bit array is a    */
/* temporary holding place for the new CRC value being calculated.    */
/* Note that there are lots of shared terms in the assignments below. */
new_bit[31] = crc_bit[31] ^ crc_bit[30] ^ crc_bit[29] ^ crc_bit[28] ^ crc_bit[27] ^ crc_bit[25] ^ crc_bit[24] ^
              crc_bit[23] ^ crc_bit[15] ^ crc_bit[11] ^ crc_bit[9]  ^ crc_bit[8]  ^ crc_bit[5];
new_bit[30] = crc_bit[30] ^ crc_bit[29] ^ crc_bit[28] ^ crc_bit[27] ^ crc_bit[26] ^ crc_bit[24] ^ crc_bit[23] ^
              crc_bit[22] ^ crc_bit[14] ^ crc_bit[10] ^ crc_bit[8]  ^ crc_bit[7]  ^ crc_bit[4];
new_bit[29] = crc_bit[31] ^ crc_bit[29] ^ crc_bit[28] ^ crc_bit[27] ^ crc_bit[26] ^ crc_bit[25] ^ crc_bit[23] ^
              crc_bit[22] ^ crc_bit[21] ^ crc_bit[13] ^ crc_bit[9]  ^ crc_bit[7]  ^ crc_bit[6]  ^ crc_bit[3];
new_bit[28] = crc_bit[30] ^ crc_bit[28] ^ crc_bit[27] ^ crc_bit[26] ^ crc_bit[25] ^ crc_bit[24] ^ crc_bit[22] ^
              crc_bit[21] ^ crc_bit[20] ^ crc_bit[12] ^ crc_bit[8]  ^ crc_bit[6]  ^ crc_bit[5]  ^ crc_bit[2];
new_bit[27] = crc_bit[29] ^ crc_bit[27] ^ crc_bit[26] ^ crc_bit[25] ^ crc_bit[24] ^ crc_bit[23] ^ crc_bit[21] ^
              crc_bit[20] ^ crc_bit[19] ^ crc_bit[11] ^ crc_bit[7]  ^ crc_bit[5]  ^ crc_bit[4]  ^ crc_bit[1];
new_bit[26] = crc_bit[31] ^ crc_bit[28] ^ crc_bit[26] ^ crc_bit[25] ^ crc_bit[24] ^ crc_bit[23] ^ crc_bit[22] ^
              crc_bit[20] ^ crc_bit[19] ^ crc_bit[18] ^ crc_bit[10] ^ crc_bit[6]  ^ crc_bit[4]  ^ crc_bit[3]  ^
              crc_bit[0];
new_bit[25] = crc_bit[31] ^ crc_bit[29] ^ crc_bit[28] ^ crc_bit[22] ^ crc_bit[21] ^ crc_bit[19] ^ crc_bit[18] ^
              crc_bit[17] ^ crc_bit[15] ^ crc_bit[11] ^ crc_bit[8]  ^ crc_bit[3]  ^ crc_bit[2];
new_bit[24] = crc_bit[30] ^ crc_bit[28] ^ crc_bit[27] ^ crc_bit[21] ^ crc_bit[20] ^ crc_bit[18] ^ crc_bit[17] ^
              crc_bit[16] ^ crc_bit[14] ^ crc_bit[10] ^ crc_bit[7]  ^ crc_bit[2]  ^ crc_bit[1];
new_bit[23] = crc_bit[31] ^ crc_bit[29] ^ crc_bit[27] ^ crc_bit[26] ^ crc_bit[20] ^ crc_bit[19] ^ crc_bit[17] ^
              crc_bit[16] ^ crc_bit[15] ^ crc_bit[13] ^ crc_bit[9]  ^ crc_bit[6]  ^ crc_bit[1]  ^ crc_bit[0];
new_bit[22] = crc_bit[31] ^ crc_bit[29] ^ crc_bit[27] ^ crc_bit[26] ^ crc_bit[24] ^ crc_bit[23] ^ crc_bit[19] ^
              crc_bit[18] ^ crc_bit[16] ^ crc_bit[14] ^ crc_bit[12] ^ crc_bit[11] ^ crc_bit[9]  ^ crc_bit[0];
new_bit[21] = crc_bit[31] ^ crc_bit[29] ^ crc_bit[27] ^ crc_bit[26] ^ crc_bit[24] ^ crc_bit[22] ^ crc_bit[18] ^
              crc_bit[17] ^ crc_bit[13] ^ crc_bit[10] ^ crc_bit[9]  ^ crc_bit[5];
new_bit[20] = crc_bit[30] ^ crc_bit[28] ^ crc_bit[26] ^ crc_bit[25] ^ crc_bit[23] ^ crc_bit[21] ^ crc_bit[17] ^
              crc_bit[16] ^ crc_bit[12] ^ crc_bit[9]  ^ crc_bit[8]  ^ crc_bit[4];
new_bit[19] = crc_bit[29] ^ crc_bit[27] ^ crc_bit[25] ^ crc_bit[24] ^ crc_bit[22] ^ crc_bit[20] ^ crc_bit[16] ^
              crc_bit[15] ^ crc_bit[11] ^ crc_bit[8]  ^ crc_bit[7]  ^ crc_bit[3];
new_bit[18] = crc_bit[31] ^ crc_bit[28] ^ crc_bit[26] ^ crc_bit[24] ^ crc_bit[23] ^ crc_bit[21] ^ crc_bit[19] ^
              crc_bit[15] ^ crc_bit[14] ^ crc_bit[10] ^ crc_bit[7]  ^ crc_bit[6]  ^ crc_bit[2];
new_bit[17] = crc_bit[31] ^ crc_bit[30] ^ crc_bit[27] ^ crc_bit[25] ^ crc_bit[23] ^ crc_bit[22] ^ crc_bit[20] ^
              crc_bit[18] ^ crc_bit[14] ^ crc_bit[13] ^ crc_bit[9]  ^ crc_bit[6]  ^ crc_bit[5]  ^ crc_bit[1];
new_bit[16] = crc_bit[30] ^ crc_bit[29] ^ crc_bit[26] ^ crc_bit[24] ^ crc_bit[22] ^ crc_bit[21] ^ crc_bit[19] ^
              crc_bit[17] ^ crc_bit[13] ^ crc_bit[12] ^ crc_bit[8]  ^ crc_bit[5]  ^ crc_bit[4]  ^ crc_bit[0];
new_bit[15] = crc_bit[30] ^ crc_bit[27] ^ crc_bit[24] ^ crc_bit[21] ^ crc_bit[20] ^ crc_bit[18] ^ crc_bit[16] ^
              crc_bit[15] ^ crc_bit[12] ^ crc_bit[9]  ^ crc_bit[8]  ^ crc_bit[7]  ^ crc_bit[5]  ^ crc_bit[4]  ^
              crc_bit[3];
new_bit[14] = crc_bit[29] ^ crc_bit[26] ^ crc_bit[23] ^ crc_bit[20] ^ crc_bit[19] ^ crc_bit[17] ^ crc_bit[15] ^
              crc_bit[14] ^ crc_bit[11] ^ crc_bit[8]  ^ crc_bit[7]  ^ crc_bit[6]  ^ crc_bit[4]  ^ crc_bit[3]  ^
              crc_bit[2];
new_bit[13] = crc_bit[31] ^ crc_bit[28] ^ crc_bit[25] ^ crc_bit[22] ^ crc_bit[19] ^ crc_bit[18] ^ crc_bit[16] ^
              crc_bit[14] ^ crc_bit[13] ^ crc_bit[10] ^ crc_bit[7]  ^ crc_bit[6]  ^ crc_bit[5]  ^ crc_bit[3]  ^
              crc_bit[2]  ^ crc_bit[1];
new_bit[12] = crc_bit[31] ^ crc_bit[30] ^ crc_bit[27] ^ crc_bit[24] ^ crc_bit[21] ^ crc_bit[18] ^ crc_bit[17] ^
              crc_bit[15] ^ crc_bit[13] ^ crc_bit[12] ^ crc_bit[9]  ^ crc_bit[6]  ^ crc_bit[5]  ^ crc_bit[4]  ^
              crc_bit[2]  ^ crc_bit[1]  ^ crc_bit[0];
new_bit[11] = crc_bit[31] ^ crc_bit[28] ^ crc_bit[27] ^ crc_bit[26] ^ crc_bit[25] ^ crc_bit[24] ^ crc_bit[20] ^
              crc_bit[17] ^ crc_bit[16] ^ crc_bit[15] ^ crc_bit[14] ^ crc_bit[12] ^ crc_bit[9]  ^ crc_bit[4]  ^
              crc_bit[3]  ^ crc_bit[1]  ^ crc_bit[0];
new_bit[10] = crc_bit[31] ^ crc_bit[29] ^ crc_bit[28] ^ crc_bit[26] ^ crc_bit[19] ^ crc_bit[16] ^ crc_bit[14] ^
              crc_bit[13] ^ crc_bit[9]  ^ crc_bit[5]  ^ crc_bit[3]  ^ crc_bit[2]  ^ crc_bit[0];
new_bit[9]  = crc_bit[29] ^ crc_bit[24] ^ crc_bit[23] ^ crc_bit[18] ^ crc_bit[13] ^ crc_bit[12] ^ crc_bit[11] ^
              crc_bit[9]  ^ crc_bit[5]  ^ crc_bit[4]  ^ crc_bit[2]  ^ crc_bit[1];
new_bit[8]  = crc_bit[31] ^ crc_bit[28] ^ crc_bit[23] ^ crc_bit[22] ^ crc_bit[17] ^ crc_bit[12] ^ crc_bit[11] ^
              crc_bit[10] ^ crc_bit[8]  ^ crc_bit[4]  ^ crc_bit[3]  ^ crc_bit[1]  ^ crc_bit[0];
new_bit[7]  = crc_bit[29] ^ crc_bit[28] ^ crc_bit[25] ^ crc_bit[24] ^ crc_bit[23] ^ crc_bit[22] ^ crc_bit[21] ^
```

```
                    crc_bit[16] ^ crc_bit[15] ^ crc_bit[10] ^ crc_bit[8]  ^ crc_bit[7]  ^ crc_bit[5]  ^ crc_bit[3]  ^
                    crc_bit[2]  ^ crc_bit[0];
      new_bit[6]  = crc_bit[30] ^ crc_bit[29] ^ crc_bit[25] ^ crc_bit[22] ^ crc_bit[21] ^ crc_bit[20] ^ crc_bit[14] ^
                    crc_bit[11] ^ crc_bit[8]  ^ crc_bit[7]  ^ crc_bit[6]  ^ crc_bit[5]  ^ crc_bit[4]  ^ crc_bit[2]  ^
                    crc_bit[1];
      new_bit[5]  = crc_bit[29] ^ crc_bit[28] ^ crc_bit[24] ^ crc_bit[21] ^ crc_bit[20] ^ crc_bit[19] ^ crc_bit[13] ^
                    crc_bit[10] ^ crc_bit[7]  ^ crc_bit[6]  ^ crc_bit[5]  ^ crc_bit[4]  ^ crc_bit[3]  ^ crc_bit[1]  ^
                    crc_bit[0];
      new_bit[4]  = crc_bit[31] ^ crc_bit[30] ^ crc_bit[29] ^ crc_bit[25] ^ crc_bit[24] ^ crc_bit[20] ^ crc_bit[19] ^
                    crc_bit[18] ^ crc_bit[15] ^ crc_bit[12] ^ crc_bit[11] ^ crc_bit[8]  ^ crc_bit[6]  ^ crc_bit[4]  ^
                    crc_bit[3]  ^ crc_bit[2]  ^ crc_bit[0];
      new_bit[3]  = crc_bit[31] ^ crc_bit[27] ^ crc_bit[25] ^ crc_bit[19] ^ crc_bit[18] ^ crc_bit[17] ^ crc_bit[15] ^
                    crc_bit[14] ^ crc_bit[10] ^ crc_bit[9]  ^ crc_bit[8]  ^ crc_bit[7]  ^ crc_bit[3]  ^ crc_bit[2]  ^
                    crc_bit[1];
      new_bit[2]  = crc_bit[31] ^ crc_bit[30] ^ crc_bit[26] ^ crc_bit[24] ^ crc_bit[18] ^ crc_bit[17] ^ crc_bit[16] ^
                    crc_bit[14] ^ crc_bit[13] ^ crc_bit[9]  ^ crc_bit[8]  ^ crc_bit[7]  ^ crc_bit[6]  ^ crc_bit[2]  ^
                    crc_bit[1]  ^ crc_bit[0];
      new_bit[1]  = crc_bit[28] ^ crc_bit[27] ^ crc_bit[24] ^ crc_bit[17] ^ crc_bit[16] ^ crc_bit[13] ^ crc_bit[12] ^
                    crc_bit[11] ^ crc_bit[9]  ^ crc_bit[7]  ^ crc_bit[6]  ^ crc_bit[1]  ^ crc_bit[0];
      new_bit[0]  = crc_bit[31] ^ crc_bit[30] ^ crc_bit[29] ^ crc_bit[28] ^ crc_bit[26] ^ crc_bit[25] ^ crc_bit[24] ^
                    crc_bit[16] ^ crc_bit[12] ^ crc_bit[10] ^ crc_bit[9]  ^ crc_bit[6]  ^ crc_bit[0];


      /* The new CRC value has been calculated as individual bits in the    */
      /* new_bit array.  Re-assembled it into a 32 bit value and "clock" it */
      /* into the "register".                                               */
      crc = 0;
      for (i = 31; i >= 0; --i)  {
         crc = crc << 1;
         crc |= new_bit[i];
      }
      printf("Running CRC value is 0x%08X\n", crc);
   }

   printf("\n\nThe total number of data words processed was %d\n", data_count);
   printf("The CRC is 0x%08X\n\n", crc);

   return 0;
}
```

A.1.5 Example CRC implementation output

The following is the sample data used as input for the example stored in file sample:

```
0x00308027
0xE1234567
0x00000000
0x00000002
0x00000000
```

Processing the command ./crc < sample yields the following output:

```
Running CRC value is 0x11E353FD
Running CRC value is 0x0F656DA7
Running CRC value is 0x3D14369C
Running CRC value is 0x92D0D681
Running CRC value is 0x319FFF6F


The total number of data words processed was 5
The CRC is 0x319FFF6F
```

A.2 Scrambling calculation

A.2.1 Scrambling overview

The following section provides an informative implementation of the scrambling polynomial. The example is intended as an aid in verifying a HDL implementation of the algorithm.

A.2.2 Example code for scrambling algorithm

The following code, written in C, illustrates an implementation of the scrambling algorithm. A Register Host to Device FIS containing a PIO write command is used as example input. The code displays both the resulting scrambled data and the raw output of the scrambler. To compile the code with the GNU tool chain, use the following command.

'gcc –o scramble.exe scramble.c'

This code is supplied for illustrative purposes only.

A.2.3 Example scrambler implementation

```
/*****************************************************************************/
/*                                                                           */
/* scramble.c                                                                */
/*                                                                           */
/* This sample code generates the entire sequence of 65535 Dwords produced   */
/* by the scrambler defined in the Serial ATA specification.  The            */
/* specification calls for an LFSR to generate a string of bits that are     */
/* then packaged into 32 bit Dwords to be XORed with the data Dwords.  The   */
/* generator polynomial specified is:                                        */
/*                         16   15   13   4                                  */
/*                 G(x) = x  + x  + x  + x + 1                               */
/*                                                                           */
/* Parallelized versions of the scrambler are initialized to a value         */
/* derived from the initialization value of 0xFFFF defined in the            */
/* specification.  This implementation is initialized to 0xF0F6.  Other      */
/* parallel implementations have different initial values.  The              */
/* important point is that the first Dword output of any implementation      */
/* needs to equal 0xC2D2768D.                                                */
/*                                                                           */
/* This code does not represent an elegant solution for a C implementation,  */
/* but it does demonstrate a method of generating the sequence that can be   */
/* easily implemented in hardware.  A block diagram of the circuit emulated  */
/* by this code is shown below.                                              */
/*                                                                           */
/*         +-----------------------------------+                             */
/*         |                                   |                             */
/*         |                                   |                             */
/*         |     +---+               +---+     |                             */
/*         |     | R |               | * |     |                             */
/*         +---->| e |---------+---->| M |----+----> Output(31 downto 16)   */
/*               | g |         |     | 1 |                                   */
/*               +---+         |     +---+                                   */
/*                             |                                             */
/*                             |     +---+                                   */
/*                             |     | * |                                   */
/*                             +---->| M |---------> Output(15 downto 0)     */
/*                                   | 2 |                                   */
/*                                   +---+                                   */
/*                                                                           */
/* The register shown in the block diagram is a 16 bit register.  The two    */
/* boxes, *M1 and *M2, each represent a multiply by a 16 by 16 binary        */
/* matrix.  A 16 by 16 matrix times a 16 bit vector yields a 16 bit vector.  */
/* The two vectors are the two halves of the 32 bit scrambler value.  The    */
/* upper half of the scrambler value is stored back into the context         */
/* register to be used to generate the next value in the scrambler           */
/* sequence.                                                                 */
/*                                                                           */
/*****************************************************************************/

#include <stdlib.h>
#include <stdio.h>


main(argc,argv)
int argc;
char *argv[];
```

```
{
   int               i, j;
   unsigned short    context;        /* The 16 bit register that holds the context or state */
   unsigned long     scrambler;      /* The 32 bit output of the circuit    */
   unsigned char     now[16];        /* The individual bits of context      */
   unsigned char     next[32];       /* The computed bits of scrambler      */



   /* Parallelized versions of the scrambler are initialized to a value     */
   /* derived from the initialization value of 0xFFFF defined in the        */
   /* specification.  This implementation is initialized to 0xF0F6.  Other  */
   /* parallel implementations have different initial values.  The          */
   /* important point is that the first Dword output of any implementation  */
   /* needs to equal 0xC2D2768D.                                            */
   context = 0xF0F6;


   for (i = 0; i < 65535; ++i)  {
      /* Split the register contents (the variable context) up into its     */
      /* individual bits for easy handling.                                 */
      for (j = 0; j < 16; ++j)  {
         now[j] = (context >> j) & 0x01;
      }

      /* The following 16 assignments implement the matrix multiplication   */
      /* performed by the box labeled *M1.                                  */
      /* Notice that there are lots of shared terms in these assignments.   */
      next[31] = now[12] ^ now[10] ^ now[7]  ^ now[3]  ^ now[1]  ^ now[0];
      next[30] = now[15] ^ now[14] ^ now[12] ^ now[11] ^ now[9]  ^ now[6]  ^ now[3]  ^ now[2]  ^ now[0];
      next[29] = now[15] ^ now[13] ^ now[12] ^ now[11] ^ now[10] ^ now[8]  ^ now[5]  ^ now[3]  ^ now[2]  ^ now[1];
      next[28] = now[14] ^ now[12] ^ now[11] ^ now[10] ^ now[9]  ^ now[7]  ^ now[4]  ^ now[2]  ^ now[1]  ^ now[0];
      next[27] = now[15] ^ now[14] ^ now[13] ^ now[12] ^ now[11] ^ now[10] ^ now[9]  ^ now[8]  ^ now[6]  ^ now[1]  ^ now[0];
      next[26] = now[15] ^ now[13] ^ now[11] ^ now[10] ^ now[9]  ^ now[8]  ^ now[7]  ^ now[5]  ^ now[3]  ^ now[0];
      next[25] = now[15] ^ now[10] ^ now[9]  ^ now[8]  ^ now[7]  ^ now[6]  ^ now[4]  ^ now[3]  ^ now[2];
      next[24] = now[14] ^ now[9]  ^ now[8]  ^ now[7]  ^ now[6]  ^ now[5]  ^ now[3]  ^ now[2]  ^ now[1];
      next[23] = now[13] ^ now[8]  ^ now[7]  ^ now[6]  ^ now[5]  ^ now[4]  ^ now[2]  ^ now[1]  ^ now[0];
      next[22] = now[15] ^ now[14] ^ now[7]  ^ now[6]  ^ now[5]  ^ now[4]  ^ now[1]  ^ now[0];
      next[21] = now[15] ^ now[13] ^ now[12] ^ now[6]  ^ now[5]  ^ now[4]  ^ now[0];
      next[20] = now[15] ^ now[11] ^ now[5]  ^ now[4];
      next[19] = now[14] ^ now[10] ^ now[4]  ^ now[3];
      next[18] = now[13] ^ now[9]  ^ now[3]  ^ now[2];
      next[17] = now[12] ^ now[8]  ^ now[2]  ^ now[1];
      next[16] = now[11] ^ now[7]  ^ now[1]  ^ now[0];

      /* The following 16 assignments implement the matrix multiplication   */
      /* performed by the box labeled *M2.                                  */
      next[15] = now[15] ^ now[14] ^ now[12] ^ now[10] ^ now[6]  ^ now[3]  ^ now[0];
      next[14] = now[15] ^ now[13] ^ now[12] ^ now[11] ^ now[9]  ^ now[5]  ^ now[3]  ^ now[2];
      next[13] = now[14] ^ now[12] ^ now[11] ^ now[10] ^ now[8]  ^ now[4]  ^ now[2]  ^ now[1];
      next[12] = now[13] ^ now[11] ^ now[10] ^ now[9]  ^ now[7]  ^ now[3]  ^ now[1]  ^ now[0];
      next[11] = now[15] ^ now[14] ^ now[10] ^ now[9]  ^ now[8]  ^ now[6]  ^ now[3]  ^ now[2]  ^ now[0];
      next[10] = now[15] ^ now[13] ^ now[12] ^ now[9]  ^ now[8]  ^ now[7]  ^ now[5]  ^ now[3]  ^ now[2]  ^ now[1];
      next[9]  = now[14] ^ now[12] ^ now[11] ^ now[8]  ^ now[7]  ^ now[6]  ^ now[4]  ^ now[2]  ^ now[1]  ^ now[0];
      next[8]  = now[15] ^ now[14] ^ now[13] ^ now[12] ^ now[11] ^ now[10] ^ now[7]  ^ now[6]  ^ now[5]  ^ now[1]  ^ now[0];
      next[7]  = now[15] ^ now[13] ^ now[11] ^ now[10] ^ now[9]  ^ now[6]  ^ now[5]  ^ now[4]  ^ now[3]  ^ now[0];
      next[6]  = now[15] ^ now[10] ^ now[9]  ^ now[8]  ^ now[5]  ^ now[4]  ^ now[2];
      next[5]  = now[14] ^ now[9]  ^ now[8]  ^ now[7]  ^ now[4]  ^ now[3]  ^ now[1];
      next[4]  = now[13] ^ now[8]  ^ now[7]  ^ now[6]  ^ now[3]  ^ now[2]  ^ now[0];
      next[3]  = now[15] ^ now[14] ^ now[7]  ^ now[6]  ^ now[5]  ^ now[3]  ^ now[2]  ^ now[1];
      next[2]  = now[14] ^ now[13] ^ now[6]  ^ now[5]  ^ now[4]  ^ now[2]  ^ now[1]  ^ now[0];
```

```
      next[1]  = now[15] ^ now[14] ^ now[13] ^ now[5]  ^ now[4]  ^ now[1]  ^ now[0];
      next[0]  = now[15] ^ now[13] ^ now[4]  ^ now[0];

      /* The 32 bits of the output have been generated in the "next" array. */
      /* Reassemble the bits into a 32 bit Dword.                           */
      scrambler = 0;
      for (j = 31; j >= 0; --j)  {
         scrambler = scrambler << 1;
         scrambler |= next[j];
      }
      /* The upper half of the scrambler output is stored backed into the   */
      /* register as the saved context for the next cycle.                  */
      context = scrambler >> 16;
      printf("0x%08X\n", scrambler);

   }

   return 0;

}
```

A.2.4 Example scrambler implementation output

The following lists the first 32 results generated by the scrambler and the C sample code listed above.

```
0xC2D2768D
0x1F26B368
0xA508436C
0x3452D354
0x8A559502
0xBB1ABE1B
0xFA56B73D
0x53F60B1B
0xF0809C41
0x747FC34A
0xBE865291
0x7A6FA7B6
0x3163E6D6
0xF036FE0C
0x1EF3EA29
0xEB342694
0x53853B17
0xE94ADC4D
0x5D200E88
0x6901EDD0
0xFA9E38DE
0x68DB4B07
0x450A437B
0x960DD708
0x3F35E698
0xFE7698A5
0xC80EF715
0x666090AF
0xFAF0D5CB
0x2B82009F
0x0E317491
0x76F46A1E
```

A.3 Example frame

Table A.1 shows the steps and values used in the transmission of a simple FIS.

For LBA = 1 234 567 and COUNT field (7:0) = 2

Table A.1 – CRC and scrambler calculation example – PIO write command

FIS Data (hex)	Scrambler Value (hex)	Scrambled Data (hex)	Accumulated CRC Value (hex)	Comments
3737 B57Ch	na	3737 B57Ch	na	SOF_P, primitives not scrambled, scrambler reset
0030 8027h	C2D2 768Dh	C2E2 F6AAh	11E3 53FDh	Register Host to Device FIS, Command = 30, C bit set
E123 4567h	1F26 B368h	FE05 F60Fh	0F65 6DA7h	LBA 1 234 567
0000 0000h	A508 436Ch	A508 436Ch	3D14 369Ch	Extended LBA = 0
0000 0002h	3452 D354h	3452 D356h	92D0 D681h	Device Control register = 0, 2 sectors
0000 0000h	8A55 9502h	8A55 9502h	319F FF6Fh	reserved = 0
319F FF6Fh	BB1A BE1Bh	8A85 4174h	na	CRC
D5D5 B57Ch	na	D5D5 B57Ch	na	EOF_P, primitives not scrambled

NOTE 84 - All values in hexadecimal, shown 31:0.

The transmitted Dwords for this Register Host to Device FIS, prior to 8b/10b encoding, are:

SOF_P
C2E2 F6AAh
FE05 F60Fh
A508 436Ch
3452 D356h
8A55 9502h
8A85 4174h
EOF_P

Appendix B. Command processing overview (informative)

B.1 NON-DATA commands

If the Command register is written by the BIOS or software driver, the host adapter sets the BSY bit to one in the shadow Status register and transmits a Command frame to the device.

If command actions are complete, the device transmits a Register Device to Host FIS to set ending content of the shadow registers.

B.2 DMA read by host from device

DMA read by host from device:

1) prior to the command being issued to the device, the host driver software programs the host adapter's DMA controller with the memory address pointer(s) and the transfer direction, and arms the DMA controller (enables the "run" flag);
2) the host driver software issues the command to the device by writing the Shadow Register Block Registers (command register last);
3) in response to the command Shadow Register Block Register being written, the host adapter sets the BSY bit to one in the Shadow Status register and transmits a Register Host to Device FIS with the Shadow Register Block contents;
4) if the device has processed the command and is ready, it transmits the read data to the host in the form of one or more Data FISes. This transfer proceeds in response to flow control signals/readiness;
5) the host adapter recognizes that the incoming frame is a Data FIS and the DMA controller is programmed, and directs the incoming data to the host adapter's DMA controller that forwards the incoming data to the appropriate host memory locations; and
6) upon completion of the transfer, the device transmits a Register Device to Host FIS to indicate ending status for the command, clearing the BSY bit to zero in the Status register, and if the interrupt flag is set to one in the header an interrupt is asserted to the host.

In some error conditions there may be no Data FIS transmitted prior to the Register Device to Host FIS being transmitted with the status information. If so, the host software driver shall abort the setup of the DMA controller.

B.3 DMA write by host to device

DMA write by host to device:

1) prior to the command being issued to the device, the host driver software programs the host adapter's DMA controller with the memory address pointer(s) and the transfer direction, and arms the DMA controller (enables the "run" flag). As a result the DMA controller becomes armed but remains paused pending a signal from the device to proceed with the data transfer;
2) the host driver software issues the command to the device by writing the Shadow Register Block Registers (command register last);
3) in response to the command Shadow Register Block Register being written, the host adapter sets the BSY bit in the Shadow Status register and transmits a Register Host to Device FIS to the device with the Shadow Register Block contents;
4) if the device is ready to receive the data from the host, the device transmits a DMA Activate FIS to the host that activates the armed DMA controller. The DMA controller transmits the write data to the device in the form of one or more Data FIS. If more than one data FIS is required to complete the overall data transfer request, a DMA Activate FIS shall be sent prior to each and every one of the subsequent Data FISes. The amount of data transmitted to the device is determined by the transfer count programmed into the host adapter's DMA controller by the host driver software during the command setup phase; and

5) upon completion of the transfer, the device transmits a Register Device to Host FIS to indicate ending status for the command, clearing the BSY bit to zero in the Status register, and if the interrupt flag is set to one in the header an interrupt is asserted to the host.

In some error conditions the device may signal an ending status by transmitting a register frame to the host without having transmitted a DMA Activate FIS. In such cases the host driver software should abort and clean up the DMA controller.

B.4 PIO data read from the device

PIO data read from the device:

1) the host driver software issues a PIO read command to the device by writing the Shadow Register Block Registers (command register last);
2) in response to the command register being written, the host adapter sets the BSY bit to one in the Shadow Status register and transmits a Register Host to Device FIS with the Shadow Register Block contents;
3) if the device has processed the command and is ready to begin transferring data to the host, it first transmits a PIO Setup FIS to the host. Upon receiving the PIO Setup FIS, the host adapter holds the FIS contents in a temporary holding buffer;
4) the device follows the PIO Setup FIS with a Data – Device to Host FIS. Upon receiving the Data FIS while holding the PIO Setup FIS, the host adapter transfers the register contents from the PIO Setup FIS into the shadow registers including the initial status value, resulting in the DRQ bit getting set to one and the BSY bit getting cleared to zero in the Status register. Also, if the Interrupt bit is set to one, an interrupt is generated to the host;
5) the host controller receives the incoming data that is part of the Data FIS into a speed matching FIFO that is conceptually attached to the Shadow Register data Block Register;
6) as a result of the issued interrupt and the DRQ bit being set to one in the Status register, host software does a REP INSW on the data register and pulls data from the head of the speed matching FIFO while the serial link is adding data to the tail of the FIFO. The flow control scheme handles data throttling to avoid underflow/overflow of the receive speed matching FIFO that feeds the data Shadow Register Block Register;
7) if the number of Words read by host software from the Data shadow register reaches the value indicated in the PIO Setup FIS, the host transfers the ending status value from the earlier PIO Setup FIS into the Shadow Status register resulting in the DRQ bit being cleared to zero and the ending status reported; and
8) if there are more data blocks to be transferred, the ending status indicates the BSY bit being set to one, and the process repeats from the device sending the PIO Setup FIS to the host.

B.5 PIO data write to the device

PIO data write to the device:

1) the host driver software issues a PIO write command to the device by writing the Shadow Register Block Registers (command register last);
2) in response to the command register being written, the host adapter sets the BSY bit to one in the Shadow Status register and transmits a Register Host to Device FIS with the Shadow Register Block contents;
3) if the device is ready to receive the PIO write data, it transmits a PIO Setup FIS to the host to indicate that the target is ready to receive PIO data and the number of Words of data that are to be transferred;
4) in response to a PIO Setup FIS with the D bit indicating a write to the device, the host transfers the beginning Status register contents from the PIO Setup FIS into the Shadow Status register, resulting in the DRQ bit getting set to one and the BSY bit cleared to zero. Also, if the Interrupt bit is set to one, an interrupt is generated to the host;
5) as a result of the DRQ bit being set to one in the Shadow Status register, the host driver software starts a REP OUTSW to the data register;

6) the data written to the data register is placed in an outbound speed matching FIFO and is transmitted to the device as a Data – Host to Device FIS. The REP OUTSW pushes data onto the tail of the FIFO and the serial link pulls data from the head. The flow control scheme handles data throttling to avoid underflow of the transmit FIFO;
7) if the number of Words indicated in the PIO Setup FIS have been written to the transmit FIFO, the host controller transfers the final status value indicated in the PIO Setup frame into the shadow Status register resulting in the DRQ bit being cleared to zero, and closes the frame with a CRC and EOFP. If additional sectors of data are to be transferred, the ending status value transferred to the Shadow Status register is possible to have the BSY bit set to one and the state is the same as immediately after the command was first issued to the device;
8) if there are more data blocks to be transferred, the ending status indicates the BSY bit being set to one, and the process repeats from the device sending the PIO Setup FIS to the host;
9) if the number of sectors indicated in the Sector Count register have been transferred, the device shall send a Register Device to Host FIS with the command complete interrupt and the BSY bit cleared to zero; and
10) in the case of a write error, the device may, on any sector boundary include error status and a command complete interrupt in the PIO setup FIS, and there is no need to send the Register Device to Host FIS.

B.6 ATA Tagged Command Queuing DMA read from device

ATA Tagged Command Queuing DMA read from device:

1) prior to the command being issued to the device, the host driver software programs the host side DMA controller with the memory address pointer(s) and the transfer direction, and arms the DMA controller (enables the “run” flag);
2) the host driver software issues the command to the device by writing the Shadow Register Block Registers (command register last);
3) in response to the command Shadow Register Block Register being written, the host adapter sets the BSY bit to one in the Shadow Status register and transmits a Register Host to Device FIS with the Shadow Register Block contents;
4) if the device has queued the command and wishes to release the bus, it transmits a Register Device to Host FIS resulting in the BSY bit being cleared to zero and the REL bit being set to one in the Status register;
5) if the device is ready to complete the transfer for the queued command, it transmits a Set Device Bits FIS to the host resulting in the SERV bit being set to one in the Status register. If no other command is active (i.e., the BSY bit set to one), then an interrupt is also generated;
6) in response to the service request, the host software deactivates the DMA controller if activated and issues a SERVICE command to the device by writing the Shadow Register Block Registers, resulting in the BSY bit getting set to one and a Register Host to Device FIS being transmitted;
7) in response to the SERVICE request, the device transmits a Register Device to Host FIS conveying the TAG value to the host and clearing the BSY bit to zero and setting the DRQ bit to one;
8) if the DRQ bit is set to one, the host software reads the TAG value from the Shadow Register Block and restores the DMA controller context appropriate for the command that is completing;
9) the device transmits the read data to the host in the form of one or more Data FIS. This transfer proceeds in response to flow control signals/readiness. Any DMA data arriving before the DMA controller has its context restored backs up into the inbound speed-matching FIFO until the FIFO is filled and thereafter is flow controlled to throttle the incoming data until the DMA controller has its context restored by host software; and
10) the host controller recognizes that the incoming packet is a Data FIS and the DMA controller is programmed, and directs the incoming data to the host controller’s DMA controller that forwards the incoming data to the appropriate host memory locations;

11) upon completion of the transfer, the target transmits a Register Device to Host FIS to indicate ending status for the command, clearing the BSY bit to zero in the Status register, and if the interrupt flag is set to one in the header an interrupt is asserted to the host.

B.7 ATA Tagged Command Queuing DMA write to device

ATA Tagged Command Queuing DMA write to device:

1) prior to the command being issued to the device, the host driver software programs the host side DMA controller with the memory address pointer(s) and the transfer direction, and arms the DMA controller (enables the “run” flag);
2) the host driver software issues the command to the device by writing the Shadow Register Block Registers (command register last);
3) in response to the Shadow Register command Block Register being written, the host adapter sets the BSY bit to one in the Shadow Status register and transmits a Register Host to Device FIS with the Shadow Register Block contents;
4) if the device has queued the command and wishes to release the bus, it transmits a Register Device to Host FIS resulting in the BSY bit being cleared to zero and the REL bit being set to one in the Status register;
5) if the device is ready to complete the transfer for the queued command, it transmits a Set Device Bits FIS to the host resulting in the SERV bit being set to one in the Status register. If no other command is active (i.e., the BSY bit set to one), then an interrupt is also generated;
6) in response to the service request, the host software deactivates the DMA controller (if activated) and issues a SERVICE command to the device by writing the Shadow Register Block Registers, resulting in the BSY bit getting set to one and a Register Host to Device FIS being transmitted;
7) in response to the SERVICE request, the device transmits a Register Device to Host FIS conveying the TAG value to the host and clearing the BSY bit to zero and setting the DRQ bit to one;
8) if the DRQ bit is set to one, the host software reads the TAG value from the Shadow Register Block and restores the DMA controller context appropriate for the command that is completing;
9) if the device is ready to receive the data from the host, the device transmits a DMA Activate FIS to the host that activates the armed DMA controller. The DMA controller transmits the write data to the device in the form of one or more Data – Host to Device FISes. If more than one data FIS is required to complete the overall data transfer request, a DMA Activate FIS shall be sent prior to each and every one of the subsequent Data FISes. The amount of data transmitted to the device is determined by the transfer count programmed into the host’s DMA controller by the host driver software during the command setup phase. If the DMA Activate FIS arrives at the host prior to host software restoring the DMA context, the DMA Activate FIS results in the DMA controller starting the transfer as soon as the host software completes programming it (i.e., the controller is already activated, and the transfer starts as soon as the context is restored); and
10) upon completion of the transfer, the target transmits a Register Host to Device FIS to indicate ending status for the command, clearing the BSY bit to zero in the Status register, and if the Interrupt bit is set to one an interrupt is asserted to the host.

B.8 ATAPI Packet commands with PIO data in

ATAPI Packet commands with PIO data in:

1) the host driver software issues a PACKET command to the device by writing the Shadow Register Block Registers (command register last);
2) in response to the command register being written, the host adapter sets the BSY bit to one in the Shadow Status register and transmits a Register Host to Device FIS with the Shadow Register Block contents;

3) if the device is ready to receive the ATAPI command packet, it transmits a PIO Setup – Device to Host FIS to the host to indicate that the target is ready to receive PIO data and the number of Words of data that are to be transferred;
4) the host transfers the beginning Status register contents from the PIO Setup FIS into the Shadow Status register, resulting in the BSY bit getting cleared to zero and the DRQ bit getting set to one;
5) as a result of the BSY bit being cleared to zero and the DRQ bit being set to one in the Shadow Status register, the host driver software writes the command packet to the Shadow Register Block data register;
6) the data written to the data register is placed in an outbound speed matching FIFO and is transmitted to the device as a Data – Host to Device FIS. The writes to the data register push data onto the tail of the FIFO and the serial link pulls data from the head. The flow control scheme handles data throttling to avoid underflow of the transmit FIFO;
7) if the number of Words indicated in the PIO Setup FIS have been written to the transmit FIFO, the host controller transfers the final status value indicated in the PIO setup frame into the Shadow Status register resulting in the DRQ bit being cleared to zero and the BSY bit being set to one, and closes the frame with a CRC and EOFP. This completes the transmission of the command packet to the device;
8) if the device has processed the command and is ready to begin transferring data to the host, it first transmits a PIO Setup – Device to Host FIS to the host. Upon receiving the PIO Setup – Device to Host FIS, the host adapter holds the FIS contents in a temporary holding buffer;
9) the device follows the PIO Setup – Device to Host FIS with a Data – Device to Host FIS. Upon receiving the Data FIS while holding the PIO Setup FIS context, the host adapter transfers the register contents from the PIO Setup FIS into the shadow registers including the initial status value, resulting in the DRQ bit getting set to one and the BSY bit getting cleared to zero in the Status register. Also, if the Interrupt bit is set to one, an interrupt is generated to the host;
10) the host controller receives the incoming data that is part of the Data FIS into a speed matching FIFO that is conceptually attached to the data Shadow Register Block Register;
11) as a result of the issued interrupt and the DRQ bit being set to one in the Status register, host software reads the byte count and does a REP INSW on the data register to pull data from the head of the speed matching FIFO while the serial link is adding data to the tail of the FIFO. The flow control scheme handles data throttling to avoid underflow/overflow of the receive speed matching FIFO that feeds the data Shadow Register Block Register;
12) if the number of Words received in the Data FIS reaches the value indicated in the PIO Setup FIS and the host FIFO is empty, the host transfers the ending status value from the earlier PIO Setup into the Shadow Status register resulting in the DRQ bit being cleared to zero and the BSY bit being set to one;
13) the device transmits final ending status by sending a Register Device to Host FIS with the BSY bit cleared to zero and the ending status for the command and the Interrupt bit set to one; and
14) the host detects an incoming register frame that contains the BSY bit cleared to zero and ending status for the command and the Interrupt bit set to one and places the frame content into the shadow registers to complete the command.

B.9 ATAPI Packet commands with PIO data out

ATAPI Packet commands with PIO data out:

1) the host driver software issues a PACKET command to the device by writing the Shadow Register Block Registers (command register last);
2) in response to the command register being written, the host adapter sets the BSY bit to one in the Shadow Status register and transmits a Register Host to Device FIS with the Shadow Register Block contents;

3) if the device is ready to receive the ATAPI command packet, it transmits a PIO Setup – Device to Host FIS to the host to indicate that the target is ready to receive PIO data and the number of Words of data that are to be transferred;
4) the host transfers the beginning Status register contents from the PIO Setup – Device to Host FIS into the shadow Status register, resulting in the BSY bit being cleared to zero and the DRQ bit being set to one;
5) as a result of the BSY bit being cleared to zero and the DRQ bit being set to one in the Shadow Status register, the host driver software writes the command packet to the Shadow Register Block data register;
6) the data written to the data register is placed in an outbound speed matching FIFO and is transmitted to the device as a Data – Host to Device FIS. The writes to the data register push data onto the tail of the FIFO and the serial link pulls data from the head. The flow control scheme handles data throttling to avoid underflow of the transmit FIFO;
7) if the number of Words indicated in the PIO Setup FIS have been written to the transmit FIFO, the host controller transfers the final status value indicated in the PIO Setup frame into the Shadow Status register resulting in the DRQ bit being cleared to zero and the BSY bit being set to one, and closes the frame with a CRC and EOFP. This completes the transmission of the command packet to the device;
8) if the device has processed the command and is ready to receive the PIO write data, it transmits a PIO Setup – Device to Host FIS to the host to indicate that the target is ready to receive PIO data and the number of Words of data that are to be transferred;
9) in response to a PIO Setup FIS with the D bit cleared to zero indicating a write to the device, the host transfers the beginning Status register contents from the PIO Setup FIS into the Shadow Status register, resulting in the DRQ bit getting set to one. Also, if the Interrupt bit is set to one, an interrupt is generated to the host;
10) as a result of the DRQ bit being set to one in the Shadow Status register, the host driver software reads the byte count and starts a REP OUTSW to the data register;
11) the data written to the data register is placed in an outbound speed matching FIFO and is transmitted to the device as a Data – Host to Device FIS. The REP OUTSW pushes data onto the tail of the FIFO and the serial link pulls data from the head. The flow control scheme handles data throttling to avoid underflow of the transmit FIFO;
12) if the number of Words indicated in the PIO Setup FIS have been written to the transmit FIFO, the host controller transfers the final status value indicated in the PIO Setup FIS into the Shadow Status register resulting in the DRQ bit being cleared to zero, the BSY bit being set to one, and closes the frame with a CRC and EOFP;
13) the device transmits final ending status by sending a Register Device to Host FIS with the BSY bit cleared to zero and the ending status for the command and the Interrupt bit set to one; and
14) the host detects an incoming Register Device to Host FIS that contains the BSY bit cleared to zero and ending status for the command and the Interrupt bit set to one and places the frame content into the shadow registers to complete the command.

B.10 ATAPI Packet commands with DMA data in

ATAPI Packet commands with DMA data in:

1) prior to the command being issued to the device, the host driver software programs the host side DMA controller with the memory address pointer(s) and the transfer direction, and arms the DMA controller (enables the “run” flag);
2) the host driver software issues a PACKET command to the device by writing the Shadow Register Block Registers (command register last);
3) in response to the command register being written, the host adapter sets the BSY bit to one in the Shadow Status register and transmits a Register Host to Device FIS with the Shadow Register Block contents;
4) if the device is ready to receive the ATAPI command packet, it transmits a PIO Setup – Device to Host FIS to the host to indicate that the target is ready to receive PIO data and the number of Words of data that are to be transferred;

5) the host transfers the beginning Status register contents from the PIO Setup FIS into the Shadow Status register, resulting in the BSY bit getting cleared to zero and the DRQ bit getting set to one;
6) as a result of the BSY bit getting cleared to zero and the DRQ bit being set to one in the Shadow Status register, the host driver software writes the command packet to the Shadow Register Block data register;
7) the data written to the data register is placed in an outbound speed matching FIFO and is transmitted to the device as a Data – Host to Device FIS. The writes to the data register push data onto the tail of the FIFO and the serial link pulls data from the head. The flow control scheme handles data throttling to avoid underflow of the transmit FIFO;
8) if the number of Words indicated in the PIO Setup FIS have been written to the transmit FIFO, the host controller transfers the final status value indicated in the PIO setup frame into the Shadow Status register resulting in the DRQ bit being cleared to zero and the BSY bit being set to one, and closes the frame with a CRC and EOFP. This completes the transmission of the command packet to the device;
9) if the device has processed the command and is ready, it transmits the read data to the host in the form of a single Data FIS. This transfer proceeds in response to flow control signals/readiness;
10) the host controller recognizes that the incoming packet is a Data FIS and the DMA controller is programmed, and directs the incoming data to the host controller's DMA controller that forwards the incoming data to the appropriate host memory locations; and
11) upon completion of the transfer, the target transmits a Register Device to Host FIS to indicate ending status for the command, clearing the BSY bit to zero in the Status register, and if the Interrupt bit is set to one an interrupt is asserted to the host.

B.11 ATAPI Packet commands with DMA data out

ATAPI Packet commands with DMA data out:

1) prior to the command being issued to the device, the host driver software programs the host side DMA controller with the memory address pointer(s) and the transfer direction, and arms the DMA controller (enables the "run" flag). As a result the DMA controller becomes armed but remains paused pending a signal from the device to proceed with the data transfer;
2) the host driver software issues a PACKET command to the device by writing the Shadow Register Block Registers (command register last);
3) in response to the command register being written, the host adapter sets the BSY bit to one in the Shadow Status register and transmits a Register Host to Device FIS with the Shadow Register Block contents;
4) if the device is ready to receive the ATAPI command packet, it transmits a PIO Setup – Device to Host FIS to the host to indicate that the target is ready to receive PIO data and the number of Words of data that are to be transferred;
5) the host transfers the beginning Status register contents from the PIO Setup FIS into the Shadow Status register, resulting in the BSY bit getting cleared to zero and the DRQ bit getting set to one;
6) as a result of the BSY bit getting cleared to zero and the DRQ bit being set to one in the Shadow Status register, the host driver software writes the command packet to the Shadow Register Block data register;
7) the data written to the data register is placed in an outbound speed matching FIFO and is transmitted to the device as a Data – Host to Device FIS. The writes to the data register push data onto the tail of the FIFO and the serial link pulls data from the head. The flow control scheme handles data throttling to avoid underflow of the transmit FIFO;
8) if the number of Words indicated in the PIO Setup FIS have been written to the transmit FIFO, the host controller transfers the final status value indicated in the PIO Setup frame into the shadow Status register resulting in the DRQ bit being cleared to zero and the BSY bit being set to one, and closes the frame with a CRC and EOFP. This completes the transmission of the command packet to the device;

9) if the device is ready to receive the data from the host, the device transmits a DMA Activate FIS to the host that activates the armed DMA controller. The DMA controller transmits the write data to the device in the form of one or more Data FIS. The transfer proceeds in response to flow control signals/readiness. The amount of data transmitted to the device is determined by the transfer count programmed into the host's DMA engine by the host driver software during the command setup phase; and
10) upon completion of the transfer, the device transmits a Register Device to Host FIS to indicate ending status for the command, clearing the BSY bit to zero in the Status register, and if the Interrupt bit is set to one an interrupt is asserted to the host.

B.12 Odd Word count considerations

B.12.1 Odd Word count considerations overview

This section outlines special considerations required to accommodate data transfers of an odd number of 16 bit Word quantities. The considerations are separately outlined for each of the data transaction types. No accommodation in Serial ATA is made for the transfer of an odd number of 8 bit byte quantities.

B.12.2 DMA read from target for odd Word count

DMA read from tartget for odd Word count:

1) prior to the command being issued to the device, the host driver software programs the host side DMA controller with the memory address pointer(s) and the transfer direction, and arms the DMA controller (enables the "run" flag). The count for the DMA transfer is an odd number of Word (16 bit) quantities;
2) if the device has processed the corresponding command and is ready to transmit the data to the host, it does so in the form of one or more data FIS. Because the transfer count is odd, the last 32 bit Dword transmitted to the host has the high order 16 bits padded with zeroes. The CRC value transmitted at the end of the FIS is computed over the entire FIS including any pad bytes in the final transmitted Dword;
3) the host controller receives the incoming data and the DMA controller directs the received data from the receive FIFO to the appropriate host memory locations. The DMA controller has a transfer granularity of a 16 bit Word (consistent with the ATA/ATAPI Host Adapters standard); and
4) upon receiving the final 32 bit Dword of receive data, the DMA controller transfers the first half (low order 16 bits) to the corresponding final memory location at that point the DMA engine's transfer count is exhausted. The DMA controller drops the high-order 16 bits of the final received Dword since it represents data received beyond the end of the requested DMA transfer. The dropped 16 high order bits corresponds with the 16 bits of transmission pad inserted by the sender.

B.12.3 DMA write by host to target for odd Word count

DMA write by host to target for odd Word count:

1) prior to the command being issued to the device, the host driver software programs the host side DMA controller with the memory address pointer(s) and the transfer direction, and arms the DMA controller (enables the "run" flag). The count for the DMA transfer is an odd number of Word (16 bit) quantities;
2) if the device has processed the corresponding command and is ready to receive the data from the host, it signals readiness with a DMA Activate FIS;
3) upon receiving the DMA Activate signal, the host transmits the data to the device in the form of one or more data FIS. Because the transfer count is odd, the DMA controller completes its data transfer from host memory to the transmit FIFO after filling only the low order 16 bits of the last Dword in the FIFO, leaving the upper 16 bits zeroed. This padded

final Dword is transmitted as the final Dword in the data frame. The CRC value transmitted at the end of the FIS is computed over the entire FIS including any pad bytes in the final transmitted Dword; and

4) having awareness of the command set and having decoded the current command, the device that receives the transmitted data has knowledge of the expected data transfer length. Upon receiving the data from the host, the device removes the 16 bit pad data in the upper 16 bits of the final 32 bit Dword of received data.

B.13 PIO data read from the device

PIO data read from the device:

1) in response to decoding and processing a PIO read command with a transfer count for an odd number of 16 bit Words, the device transmits the corresponding data to the host in the form of a single Data FIS. The device pads the upper 16 bits of the final 32 bit Dword of the last transmitted FIS in order to close the FIS. The CRC value transmitted at the end of the FIS is computed over the entire FIS including any pad bytes in the final transmitted Dword;
2) host driver software responsible for retrieving the PIO data is aware of the number of Words of data it expects to retrieve from the Shadow Command Block Register Data register and performs a REP INSW operation for an odd number of repetitions; and
3) upon exhaustion of the REP INSW operation by the host driver software, the receive FIFO that interfaces with the data register has one 16 bit Word of received data remaining in it that corresponds to the pad that the device included at the end of the transmitted frame. This remaining Word of data left in the data register FIFO is flushed upon the next write of the Shadow Command Block Register Command register or upon the receipt of the next data FIS from the device.

B.14 PIO data write to the device

PIO data write to the device:

1) in response to decoding and processing a PIO write command with a transfer count for an odd number of 16 bit Words, the device transmits a PIO Setup FIS to the host indicating it is ready to receive the PIO data and indicating the transfer count. The conveyed transfer count is for an odd number of 16 bit Word quantities;
2) host driver software responsible for transmitting the PIO data is aware of the number of Words of data it needs to write to the Shadow Command Block Register Data register and performs a REP OUTSW operation for an odd number of repetitions;
3) after the final write by the software driver to the Shadow Command Block Register Data register, the transfer count indicated in the PIO Setup packet is exhausted that signals the host controller to close the FIS. Since the transfer count was odd, the upper 16 bits of the final 32 bit Dword of data to transmit remains zeroed (pad) and the host controller closes the FIS after transmitting this final padded Dword. The CRC value transmitted at the end of the FIS is computed over the entire FIS including any pad bytes in the final transmitted Dword; and
4) having awareness of the command set and having decoded the current command, the device that receives the transmitted data has knowledge of the expected data transfer length. Upon receiving the data from the host, the device removes the 16 bit pad data in the upper 16 bits of the final 32 bit Dword of received data.

B.15 NCQ examples

B.15.1 NCQ examples overview

The following is an overview of macro operations and their sequencing for two typical NCQ scenarios. These illustrative sequences presume a host controller DMA implementation equivalent to current mainstream desktop implementations (hence references to PRD tables and other data structures typically referenced for such implementations) and these illustrative sequences are not intended to exclude other possible host controller implementations.

B.15.2 Queued commands with Out of Order Completion

Table B.1 shows a set of queued commands with Out of Order Completion.

Table B.1 – Queued commands with Out of Order Completion (part 1 of 2)

HOST Actions	DEVICE Actions
Host issues Read Command Tag = 0 by presetting bit 0 in the SActive register by writing the value 0000 0001h (i.e., 0000 0000 0000 0000 0000 0000 0000 0001b) to it and transmitting a Register Host to Device FIS.	
	Device clears the BSY bit to zero by transmitting a Register Device to Host FIS.
Host issues Read Command Tag = 5 if the BSY bit not yet cleared to zero, host needs to wait by presetting bit 5 in the SActive register by writing the value 0000 0020h (i.e., 0000 0000 0000 0000 0000 0000 0010 0000b) to it and transmitting a Register Host to Device FIS. The resultant SActive register value is 21h.	
	Device clears the BSY bit to zero.
	Device sends DMA Setup FIS, DMA Buffer Identifier = 5 (in this example the second issued command is serviced first).
Host loads PRD pointer into DMA engine corresponding to buffer 5.	
	Device sends data for command corresponding to TAG = 5.
Host DMA engine directs incoming data into buffer 5.	
	Device sends Set Device Bits FIS with Interrupt bit set to one and with SActive value of 0000 0020h (i.e., 0000 0000 0000 0000 0000 0000 001 0 0000b), indicating that TAG = 5 has finished.
Host receives Set Device Bits FIS with SActive field value of 20h and Interrupt bit set to one. Results in bit 5 in SActive register getting cleared to zero yielding a value of 01h in the SActive shadow register and interrupt getting triggered.	
	Device sends DMA Setup FIS, DMA Buffer Identifier = 0
Host loads PRD pointer into DMA engine corresponding to buffer 0. Host software processes the received interrupt. Reads SActive shadow register and determines that bit 5 is negated and retires command with TAG = 5	
	Device sends data for command corresponding to TAG = 0
Host DMA engine directs incoming data into buffer 0	

Table B.1 – Queued Commands with Out of Order Completion (part 2 of 2)

HOST Actions	DEVICE Actions
	Device sends Set Device Bits FIS with Interrupt bit set to one and with SActive value of 0000 0001h (i.e., 0000 0000 0000 0000 0000 0000 0000 0001b), indicating that TAG = 0 has finished.
Host receives Set Device Bits FIS with SActive field value of 01h and Interrupt bit set to one. Results in bit 0 in SActive register getting cleared to zero yielding a value of 00h in SActive shadow register and interrupt getting triggered.	
	Device idle
Host software processes the received interrupt. Reads SActive shadow register and determines that bit 0 is negated and retires command with TAG = 0. Host idle.	

B.15.3 Interrupt aggregation

Table B.2 shows interrupt aggregation.

Table B.2 – Interrupt aggregation (part 1 of 2)

HOST Actions	DEVICE Actions
Host issues Read Command Tag = 0 by presetting bit 0 in the SActive register by writing the value 0000 0001h (i.e., 0000 0000 0000 0000 0000 0000 0000 0001b) to it and transmitting a Register Host to Device FIS.	
	Device clears the BSY bit to zero by transmitting a Register Device to Host FIS.
Host issues Read Command Tag = 5 (if the BSY bit not yet cleared to zero, host needs to wait) by presetting bit 5 in the SActive register by writing the value 00000020h (i.e., 0000 0000 0000 0000 0000 0000 0010 0000b) to it and transmitting a Register Host to Device FIS. The resultant SActive register value is 21h.	
	Device clears the BSY bit to zero.
	Device sends DMA Setup FIS, DMA Buffer Identifier = 5 (in this example the second issued command is serviced first).
Host loads PRD pointer into DMA engine corresponding to buffer 5.	
	Device sends data for command corresponding to TAG = 5.
Host DMA engine directs incoming data into buffer 5.	

Table B.2 – Interrupt aggregation (part 2 of 2)

HOST Actions	DEVICE Actions
	Device sends Set Device Bits FIS with Interrupt bit set to one and with SActive value of 0000 0020h ((i.e., 0000 0000 0000 0000 0000 0000 0010 0000b), indicating that TAG = 5 has finished.
Host receives Set Device Bits FIS with SActive field value of 20h and Interrupt bit set to one. Results in bit 5 in SActive register getting cleared to zero yielding a value of 01h in the SActive shadow register and interrupt getting triggered.	
	Device sends DMA Setup FIS, DMA Buffer Identifier = 0
Host loads PRD pointer into DMA engine corresponding to buffer 0	
	Device sends data for command corresponding to TAG = 0
Host DMA engine directs incoming data into buffer 0 Host is busy and is slow to respond to and clear the received interrupt. Host interrupt response time is slow relative to device completion speed for this example.	
	Device sends Set Device Bits FIS with Interrupt bit set to one and with SActive value of 00000001h (i.e., 0000 0000 0000 0000 0000 0000 0000 0001b), indicating that TAG = 0 has finished.
Host receives Set Device Bits FIS with SActive field value of 01h and Interrupt bit set to one. Results in bit 0 in SActive register getting cleared to zero yielding value of 00h in SActive shadow register and interrupt getting triggered. If the host had not already reset the pending interrupt from the completion of TAG = 5, no new interrupt is triggered. If the previous interrupt has already been reset, then a new interrupt is triggered that is the previous example details as given in B.15.2	
Host had a long latency, but is now processing the interrupt and has reset the interrupt pending flag. Host is now doing command completion processing. The second interrupt issued by the device got aggregated because the first interrupt did not get reset soon enough. Host reads SActive shadow register and sees that TAG = 0 and TAG = 1 commands have both completed (neither have their SActive bit set to one). Host processes command completions and retires both commands.	
	Device idle

Appendix C. Device emulation of nIEN with interrupt pending (informative)

This specification defines the Interrupt bit in Register Device to Host FISes as the interrupt pending state of the device, and it is not modified by the state of nIEN bit in received Register Host to Device FISes. In this specification, devices ignore the nIEN bit in received Register Host to Device FISes and always perform as if nIEN bit is cleared to zero (see 10.5.5 and 10.5.6).

Some devices implemented to prior Serial ATA specification revisions used the nIEN bit of the Register Host to Device FIS as a pre-condition to setting the Interrupt bit to one of the Register Device to Host FIS and Set Device Bits FIS. The purpose of using nIEN bit to enable the Interrupt bit was to emulate the operation of the parallel implementation of ATA. In the parallel implementation, if the nIEN bit is cleared to zero, the device is enabled for the INTRQ line to the host. If the nIEN bit is set to one, the INTRQ line is put into the high impedance state by the device.

One serious side effect of device emulation of nIEN is the possibility of lost interrupts. In the parallel implementation, a host may disable interrupts, and upon re-enabling interrupts (by clearing nIEN to zero) see the INTRQ line again asserted. If a serial device is performing I bit masking based on the state of nIEN bit, a Register Device to Host FIS may be received with Interrupt bit cleared to zero (since it is masked by the nIEN bit). The device, however, may have an interrupt pending at that time. If the host writes the Device Control register with nIEN bit cleared to zero, it should not see the pending interrupt reflected by the assertion of INTRQ as in the parallel case. There is no way for the device to "re-send" the Interrupt bit set to one condition to the host. In this instance, the host has to resort to a polling operation to resume the operation.

The system designer should be aware of the following:

1) the Interrupt bit is the interrupt pending flag for Serial ATA devices;
2) the behavior of the Interrupt bit may be modified by the state of the nIEN bit in devices implemented to prior versions of this specification. Such devices may change the behavior of the Interrupt bit based on the current state of the nIEN bit, as last written by a Register Host to Device FIS with C bit cleared to zero or C bit set to one. Some devices do not observe the nIEN bit if C bit is set to one. Devices that do not observe the nIEN bit if C bit is set to one have compatibility issues with hosts that only transmit a Register Host to Device FIS with the C bit cleared to zero if the SRST bit changes state;
3) there is no defined behavior for a device if nIEN bit changes and the modification of the behavior of the Interrupt bit by nIEN is vendor specific;
4) the host should clear the nIEN bit to zero in all Register Host to Device FISes. This results in the highest compatibility with devices that use the nIEN bit as a pre-condition to setting the interrupt pending flag; and
5) if a device supports nIEN emulation, and the nIEN bit is set to one by the host, the host driver should accommodate the case of no interrupt generation if nIEN bit is cleared to zero and the device has a pending interrupt.

Appendix D. I/O controller module (informative)

D.1 I/O controller module overview

The purpose of this optional feature is to specify the IO Connectors (shown with heavy outline below) that provide interconnection for power, data and enclosure services between I/O controller(s) and a Serial ATA disk backplane. This specification defines interoperability between I/O modules from one manufacturer that work with backplanes from a different manufacturer. This interoperability includes electrical, mechanical and enclosure management connections. Compliant I/O controllers and Serial ATA backplanes shall adhere to the specified connector placement and connector pinout.

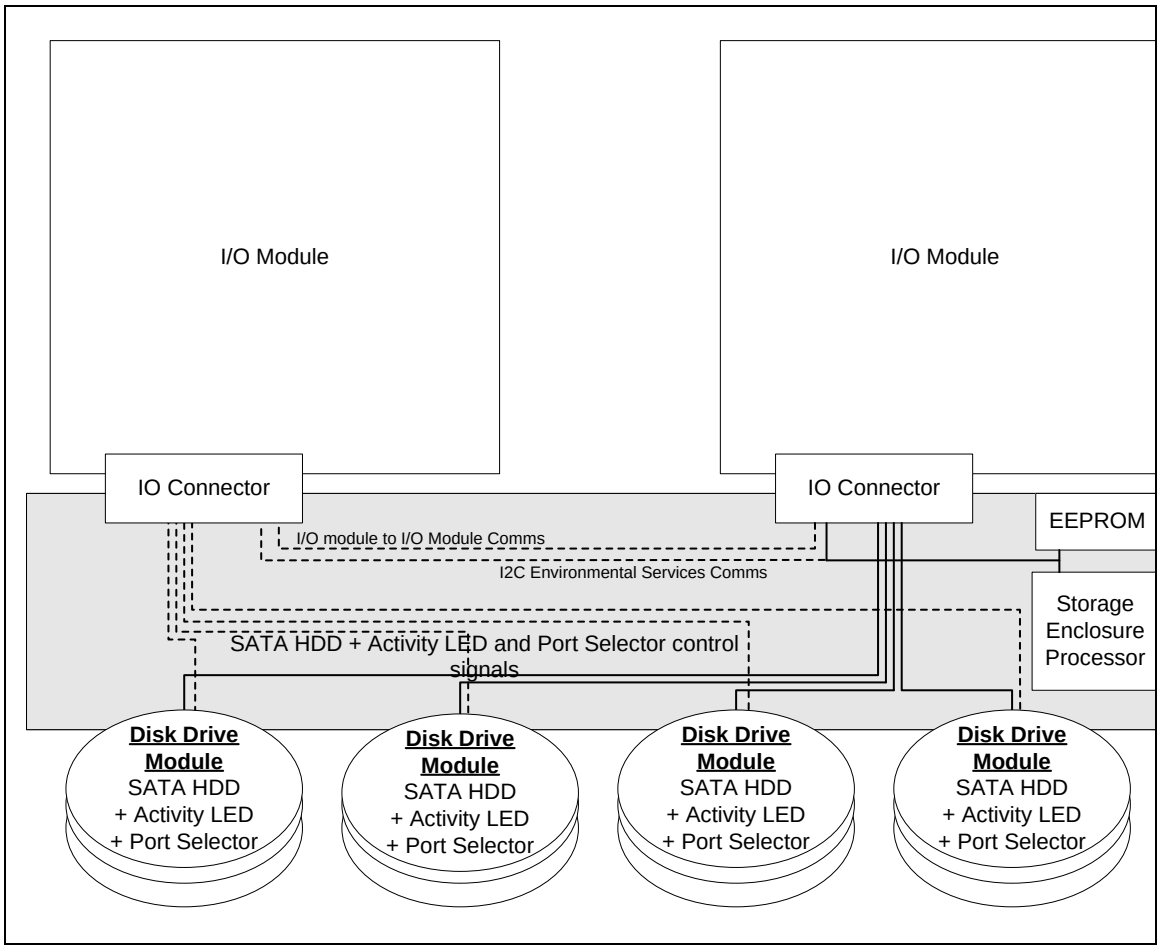


Figure D.1 – Concept summary interconnect structure

2. The I/O connector(s), shown in Figure D.1 with the heavy outline, are the central items being specified. The I/O Module in dotted line format represents a secondary and optional I/O module. The shaded area represents the backplane.

The Disk Drive Module is envisioned as a standard Serial ATA HDD within a carrier, including:

a) power connection between the module and the backplane;
b) data connection between the module and the backplane;
c) HDD activity LED (see 6.13.1.1), optional; and
d) Port Selector with a secondary connection to the backplane, optional.

D.2 Supported configurations

D.2.1 Supported configurations overview

Two configuration goals are supported by the connector specification:

a) Single I/O Controller System; or
b) Dual I/O Controller System with redundancy features (optional).

Each implementation has mandatory and optional features supported through the use of the available signals.

D.2.2 Single I/O controller signals

Signal I/O controller signals:

a) Serial ATA interface;
b) I/O controller to enclosure processor (SEP) communication;
c) disk activity LED signals;
d) I/O controller power and ground; and
e) I/O controller identification and control (RESET).

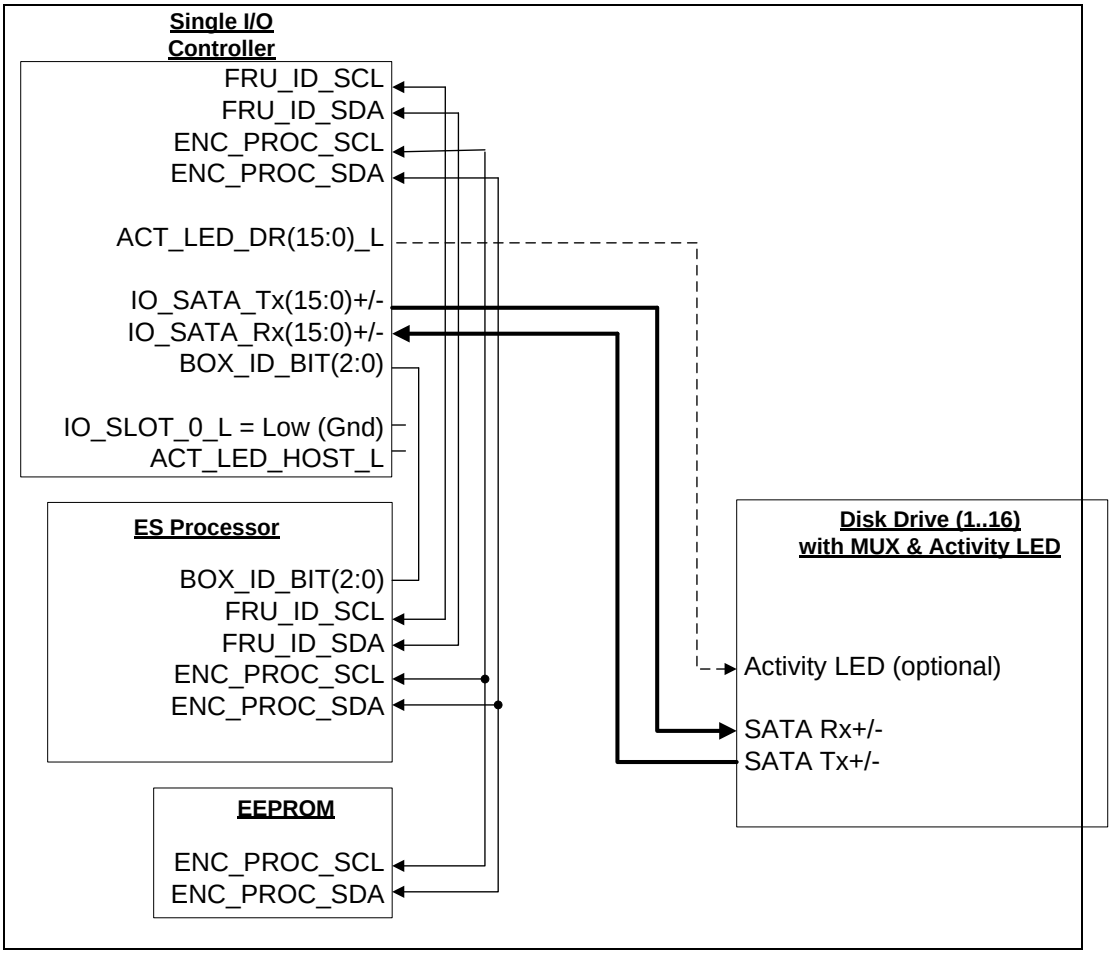


Figure D.2 – An example of signal connections with one I/O controller

D.2.3 Dual I/O controller signals

As for single I/O controller and in addition:

a) multiplexer control to provide dual access for I/O controller failover capabilities; and
b) I/O controller to I/O controller communication.

D.2.4 Further optional features

Futher optional features are:

a) RAID battery backup support; and
b) additional low and high speed communications for optional board-to-board communications. As this specification is not intended to define board-to-board for boards from different manufacturers; the actual implementation of signaling and protocol is left to the discretion of the I/O controller manufacturer.

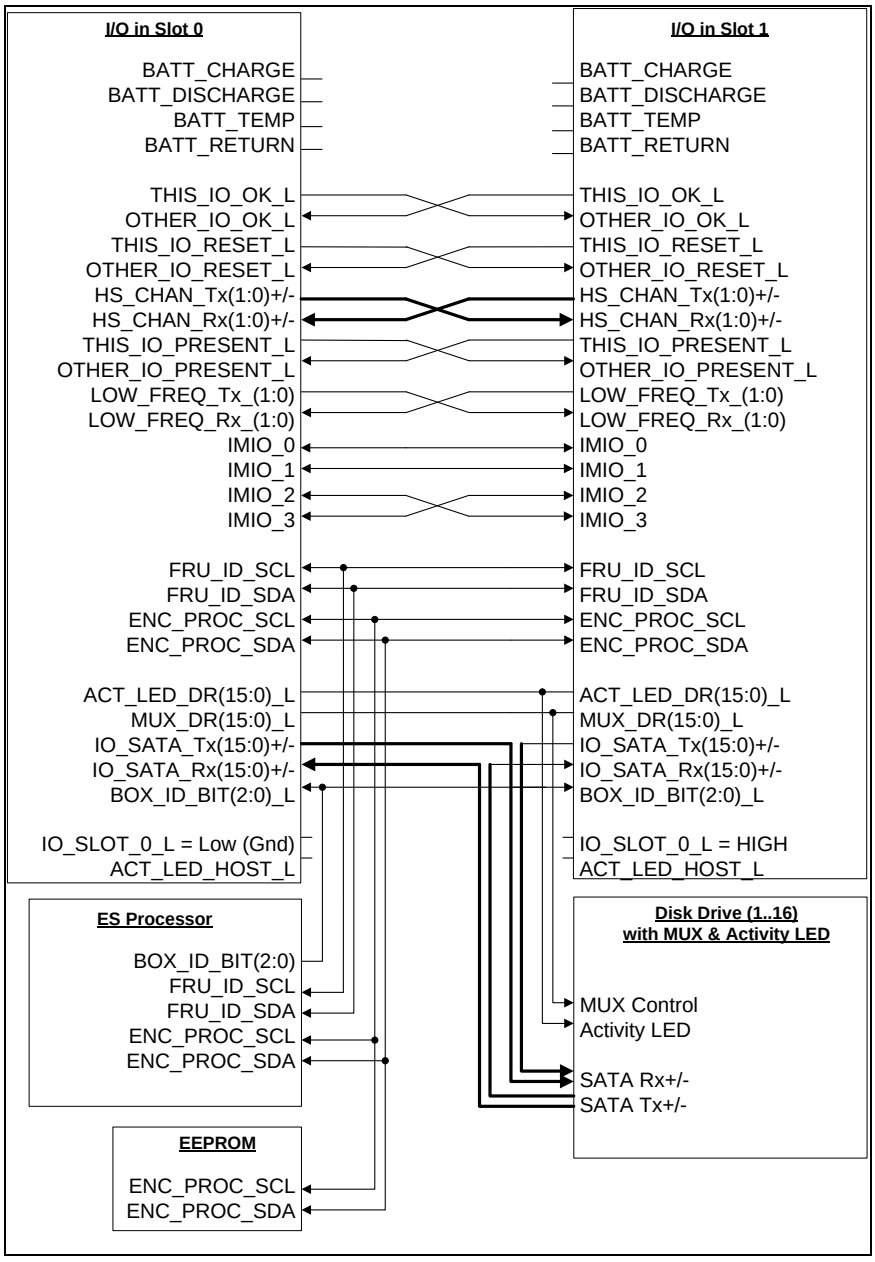


Figure D.3 – Example of signal connections with two I/O modules

D.3 Optional high speed channel configurations

The optional High speed communications channels may be used for high speed differential communications between the two I/O controllers, or for I/O controller to host communications (e.g., inclusion into a Fibre Channel loop). It is recommended that these signals should be 100 ohm differential characteristic impedance.

Actual usage is open to the user definition however the normative backplane routing should be either:

a) Configuration 0, both channels link I/O controller to I/O controller; and
b) Configuration 1, one channel links the two I/O Controllers, and the other is router to a host connector.

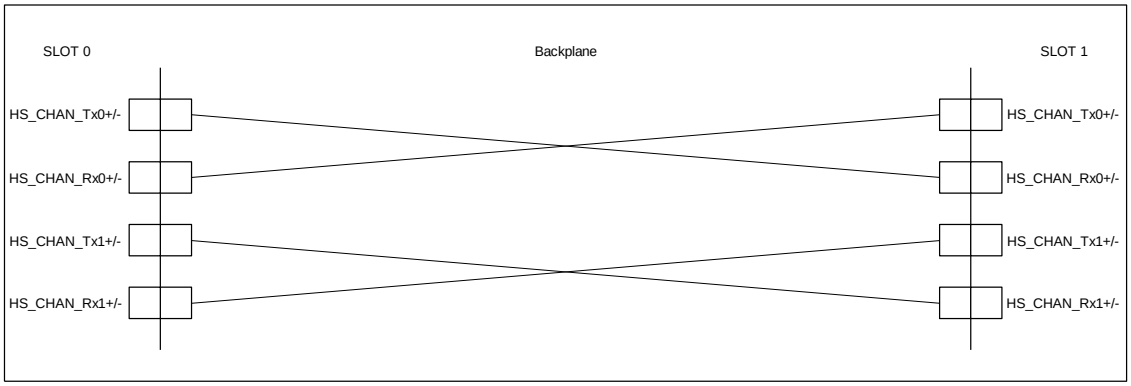


Figure D.4 – High speed channels – configuration 0

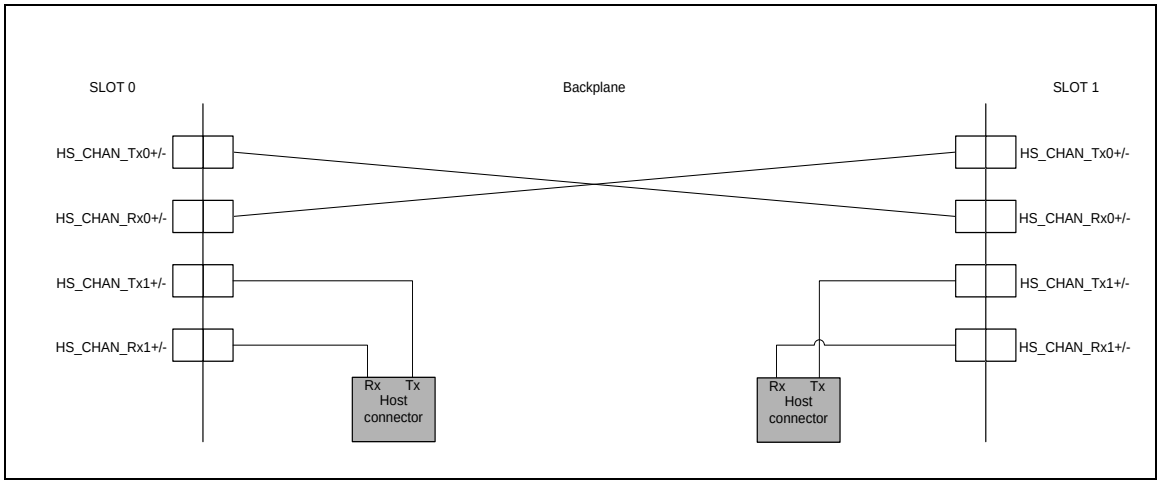


Figure D.5 – High-speed channels – configuration 1

D.4 Optional low speed channel configurations

The optional Low speed communications channels may be used for end-to-end I/O controller to I/O controller communications. There is a pair transmit and receive signals for low frequency (< 500 kHz) I/O controller to I/O controller communications.

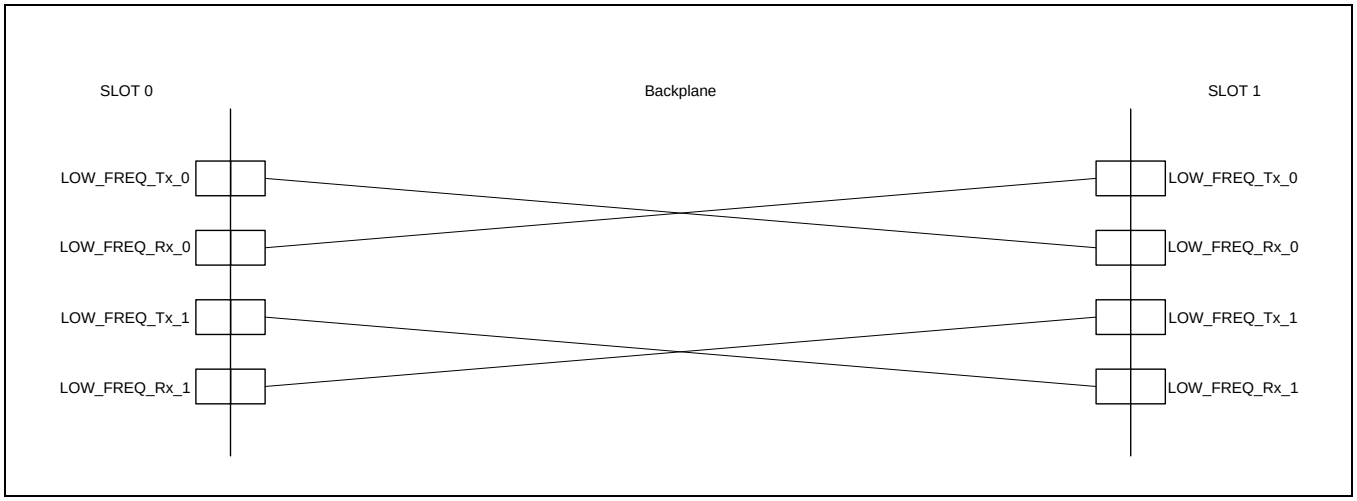


Figure D.6 – Low speed channels

There are 4 end-to-end signals that may be used for intermodule I/O communications.

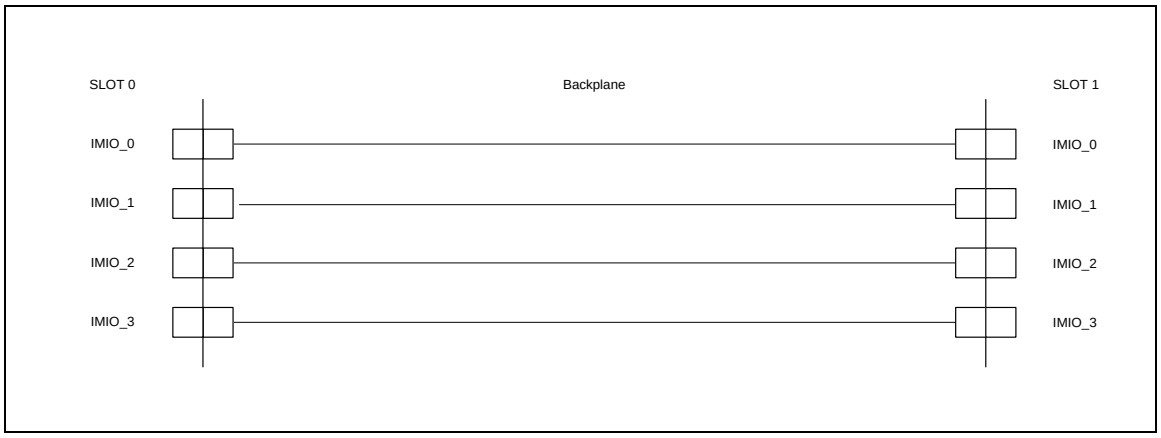


Figure D.7 – Interconnect channels

D.5 I/O controller module connectors

D.5.1 I/O controller module connectors overview

The interface connector that shall be used on the backplane is based on a Berg-FCI High Speed Metral® 5x6 30-pin male header connector part number 59566-1001 or equivalent. Equivalent parts from ITT Cannon: on the backplane 5-row 4000 male CBC20HS4000-030WXP5-5yy-x-VR, and on the I/O controller 5-row 4000 std-female CBC20HS4000-030FDP5-500-x-VR.

D.5.2 I/O controller module connector

The interface connector that shall be used on the I/O controller is based on a Berg-FCI High Speed Metral 5x6 30-pin right angle female receptacle part number 52057-102. This connector is used for the 6 high-speed connectors.

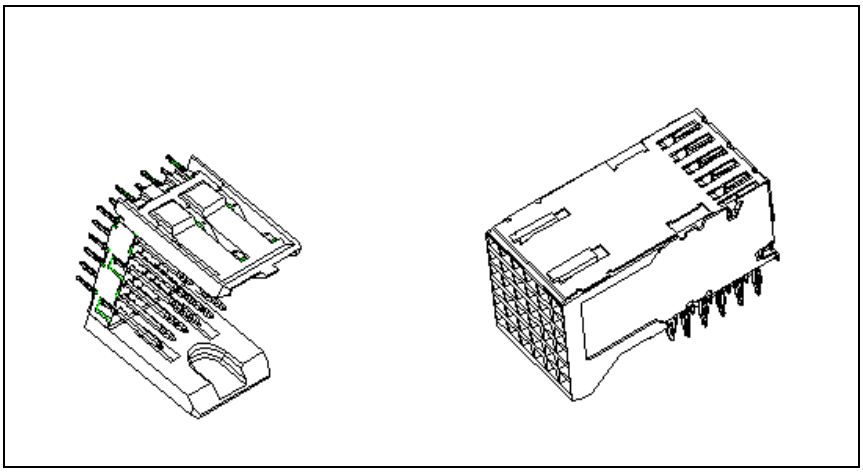


Figure D.8 – I/O controller module connector rendering

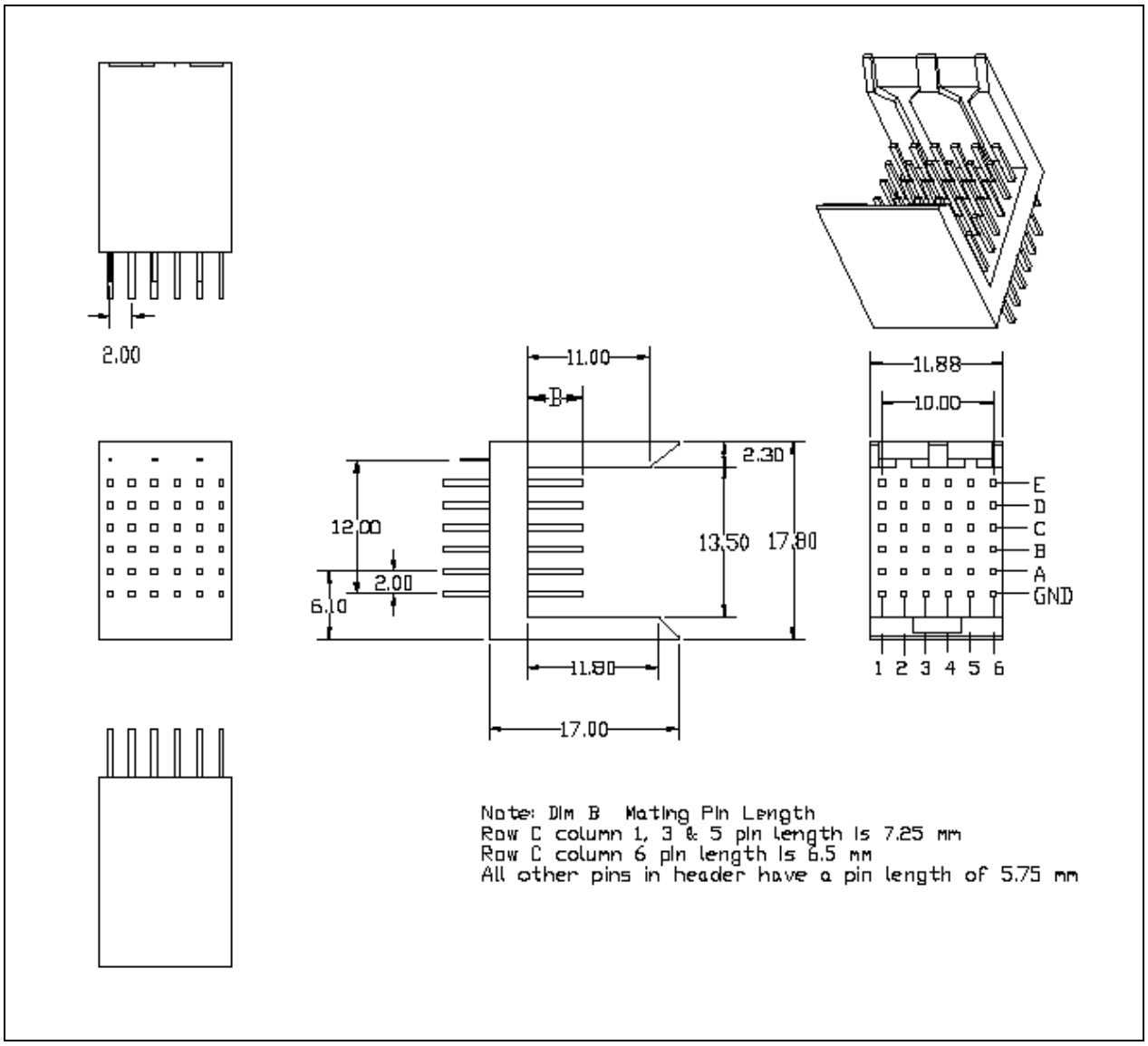


Figure D.9 – Connector pin layout and pin lengths

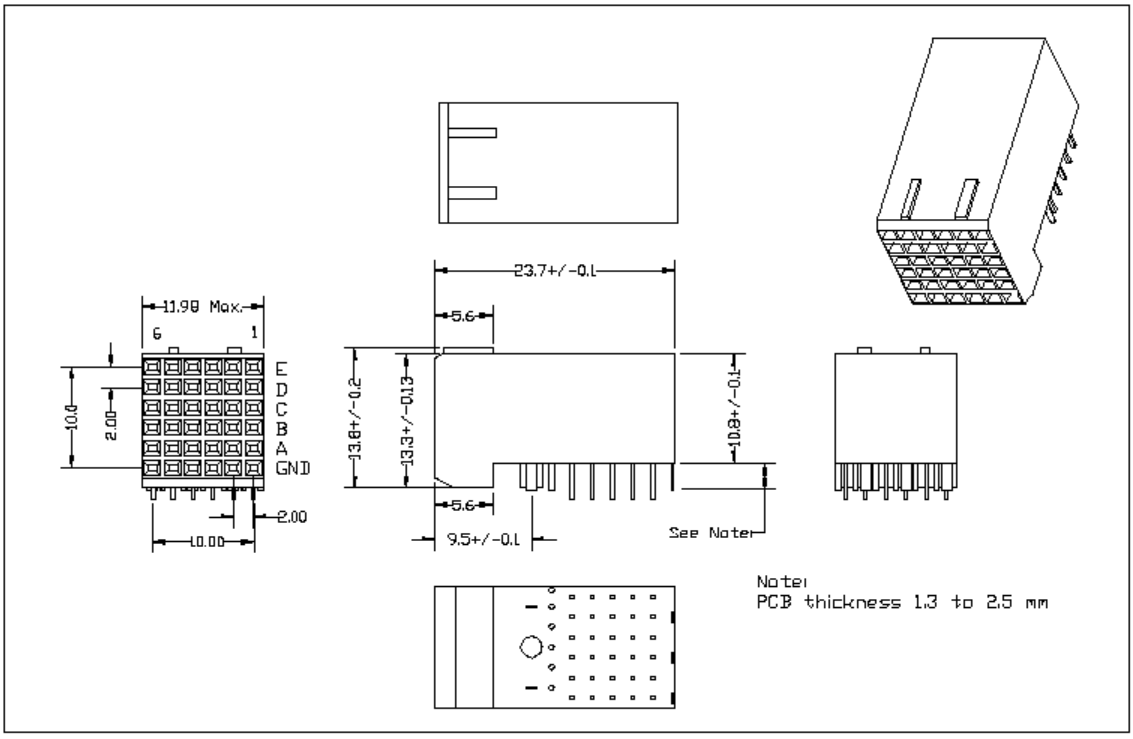


Figure D.10 – I/O controller module connector receptacle engineering drawing

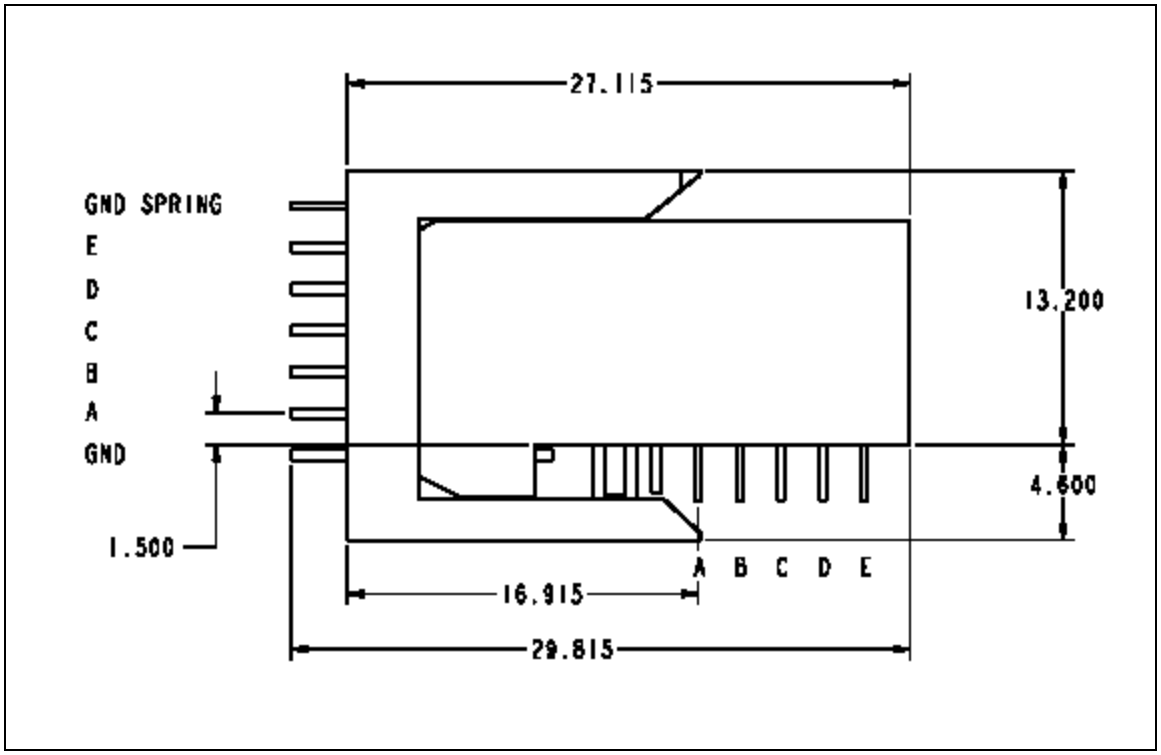


Figure D.11 – Side view of connector

D.6 I/O controller module connector locations

This section defines the mating connector locations and connector alignment between I/O modules from one manufacturer and backplanes from a different manufacturer. This is an optional feature of this specification; however I/O controllers and Serial ATA backplanes in support of this industry normal Serial ATA I/O Controller-to-Backplane interface definition shall adhere to this specified connector placement, and connector pinout.

The width and length dimensions of the I/O Module shown in Figure D.12 and Figure D.13 are presented as examples and not mandatory requirements.

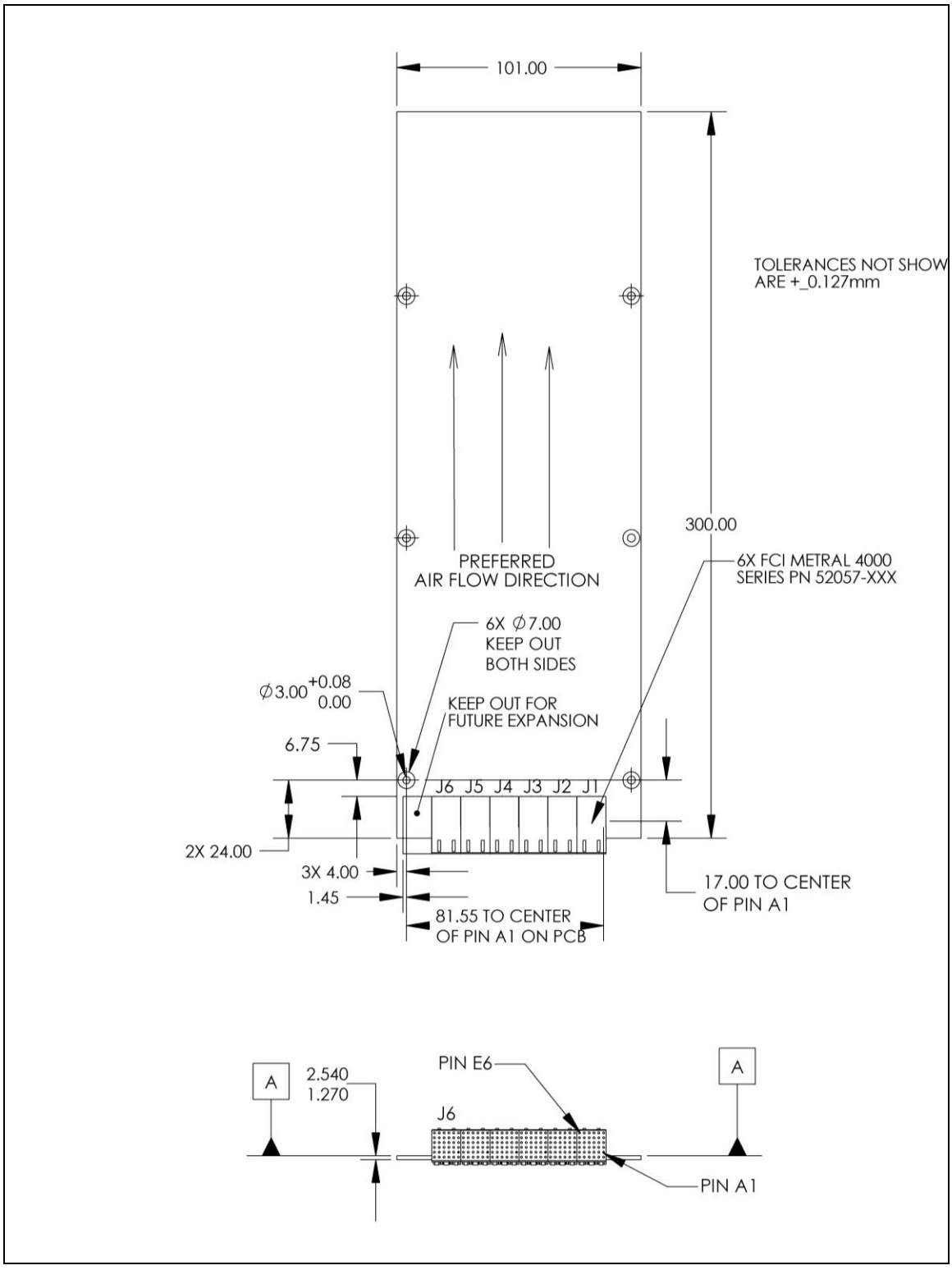


Figure D.12 – I/O controller module connector locations on 1xWide I/O module

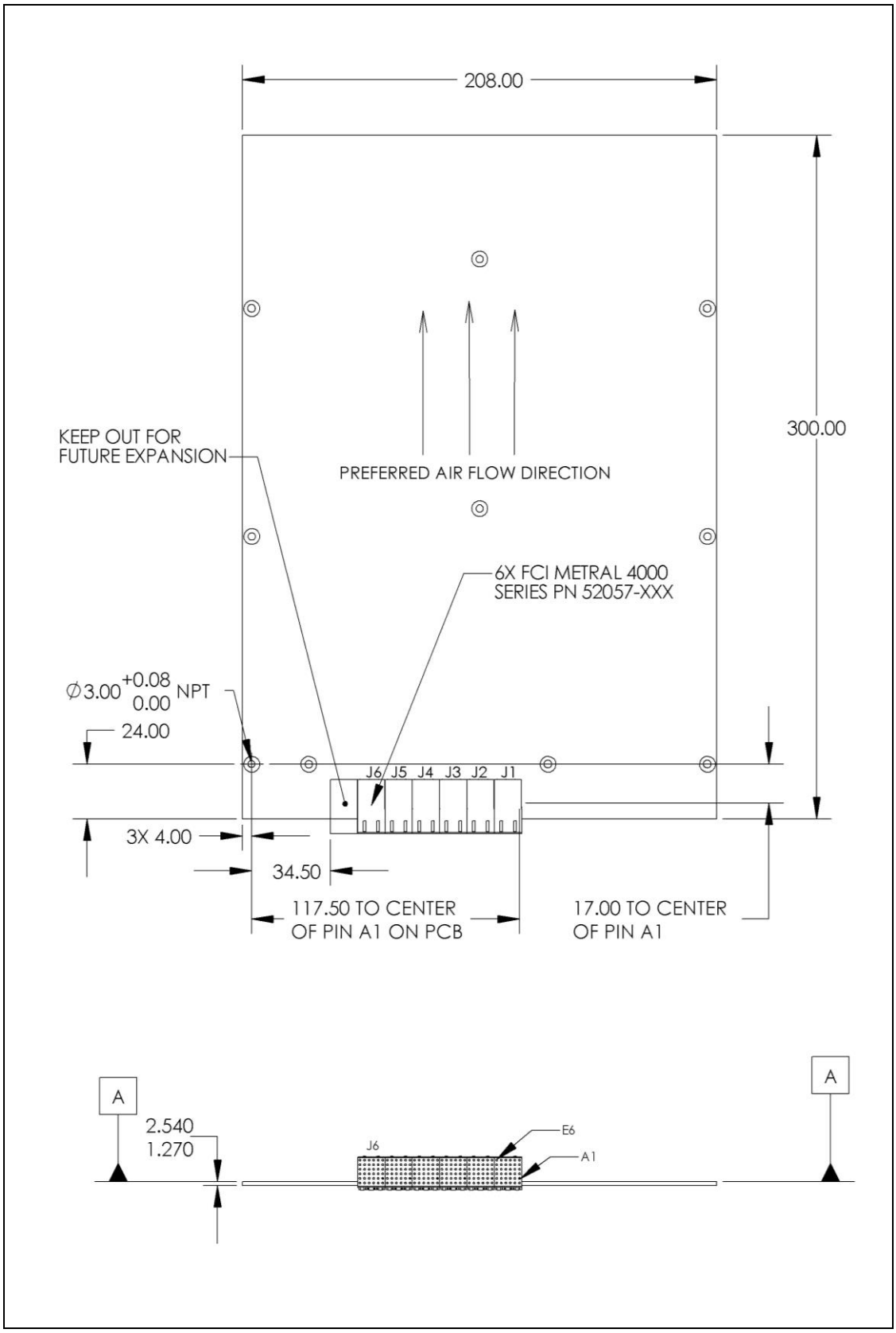


Figure D.13 – I/O controller module connector locations on 2xWide I/O module

D.7 Pinout listing

These tables (see Table D.3 and Table D.4) show the pin listing for the Metral 4000 connector system.

NOTE 85 - Note that there are grounded shields between each connector row and signal that are not shown in this matrix to aid clarity.

Table D.3 – J1, J2, J3 pin assignments

J1	Row E	Row D	Row C	Row B	Row A	Ground
1	IO_SATA_RX0+	IO_SATA_RX0-	Ground	IO_SATA_TX0+	IO_SATA_TX0-	Shield
2	ACT_LED_DR0_L	MUX_DR0_L	5 V	5 V	IO_SLOT_0_L	Shield
3	IO_SATA_RX1+	IO_SATA_RX1-	Ground	IO_SATA_TX1+	IO_SATA_TX1-	Shield
4	ACT_LED_DR1_L	MUX_DR1_L	5 V	5 V	OTHER_IO_PRESENT_L	Shield
5	IO_SATA_RX2+	IO_SATA_RX2-	Ground	IO_SATA_TX2+	IO_SATA_TX2-	Shield
6	ACT_LED_DR2_L	MUX_DR2_L	5 V PRECHARGE	ENC_PROC_SDA	ENC_PROC_SCL	Shield
J2	Row E	Row D	Row C	Row B	Row A	Ground
1	IO_SATA_RX3+	IO_SATA_RX3-	Ground	IO_SATA_TX3+	IO_SATA_TX3-	Shield
2	ACT_LED_DR3_L	MUX_DR3_L	5 V	5 V	BOX_ID_BIT_0	Shield
3	IO_SATA_RX4+	IO_SATA_RX4-	Ground	IO_SATA_TX4+	IO_SATA_TX4-	Shield
4	ACT_LED_DR4_L	MUX_DR4_L	5 V	5 V	BOX_ID_BIT_1	Shield
5	IO_SATA_RX5+	IO_SATA_RX5-	Ground	IO_SATA_TX5+	IO_SATA_TX5-	Shield
6	ACT_LED_DR5_L	MUX_DR5_L	5 V PRECHARGE	FRU_ID_SDA	FRU_ID_SCL	Shield
J3	Row E	Row D	Row C	Row B	Row A	Ground
1	IO_SATA_RX6+	IO_SATA_RX6-	Ground	IO_SATA_TX6+	IO_SATA_TX6-	Shield
2	ACT_LED_DR6_L	MUX_DR6_L	5 V	5 V	BATT_CHARGE	Shield
3	IO_SATA_RX7+	IO_SATA_RX7-	Ground	IO_SATA_TX7+	IO_SATA_TX7-	Shield
4	ACT_LED_DR7_L	MUX_DR7_L	5 V	THIS_IO_PRESENT_L	BATT_DISCHARGE	Shield
5	IO_SATA_RX8+	IO_SATA_RX8-	Ground	IO_SATA_TX8+	IO_SATA_TX8-	Shield
6	ACT_LED_DR8_L	MUX_DR8_L	12 V PRECHARGE	12 V	12 V	Shield

Table D.4 – J4, J5, J6 pin assignments

J4	Row E	Row D	Row C	Row B	Row A	Ground
1	IO_SATA_RX9+	IO_SATA_RX9-	Ground	IO_SATA_TX9+	IO_SATA_TX9-	Shield
2	ACT_LED_DR9_L	MUX_DR9_L	5 V	12 V	BATT_RETURN	Shield
3	IO_SATA_RX10+	IO_SATA_RX10-	Ground	IO_SATA_TX10+	IO_SATA_TX10-	Shield
4	ACT_LED_DR10_L	MUX_DR10_L	5 V	THIS_IO_OK_L	BATT_TEMP	Shield
5	IO_SATA_RX11+	IO_SATA_RX11-	Ground	IO_SATA_TX11+	IO_SATA_TX11-	Shield
6	ACT_LED_DR11_L	MUX_DR11_L	Reserved (3_3 V PRECHARGE)	Reserved (3_3 V)	BOX_ID_BIT_2	Shield
J5	Row E	Row D	Row C	Row B	Row A	Ground
1	IO_SATA_RX12+	IO_SATA_RX12-	Ground	IO_SATA_TX12+	IO_SATA_TX12-	Shield
2	ACT_LED_DR12_L	MUX_DR12_L	5 V	LOW_FREQ_RX_2	LOW_FREQ_TX_2	Shield
3	IO_SATA_RX13+	IO_SATA_RX13-	Ground	IO_SATA_TX13+	IO_SATA_TX13-	Shield
4	ACT_LED_DR13_L	MUX_DR13_L	5 V	ACT_LED_HOST0	ACT_LED_HOST1	Shield
5	IO_SATA_RX14+	IO_SATA_RX14-	Ground	IO_SATA_TX14+	IO_SATA_TX14-	Shield
6	ACT_LED_DR14_L	MUX_DR14_L	Reserved (3_3 V PRECHARGE)	Reserved	Reserved	Shield
J6	Row E	Row D	Row C	Row B	Row A	Ground
1	IO_SATA_RX15+	IO_SATA_RX15-	Ground	IO_SATA_TX15+	IO_SATA_TX15-	Shield
2	ACT_LED_DR15_L	MUX_DR15_L	5 V	5 V	OTHER_IO_RESET_L	Shield
3	HS_CHAN_TX(0)+	HS_CHAN_TX(0)-	Ground	HS_CHAN_RX(0)+	HS_CHAN_RX(0)-	Shield
4	LOW_FREQ_RX_1	LOW_FREQ_TX_1	5 V	IMIO_0	IMIO_1	Shield
5	HS_CHAN_TX(1)+	HS_CHAN_TX(1)-	Ground	HS_CHAN_RX(1)+	HS_CHAN_RX(1)-	Shield
6	OTHER_IO_OK_L	THIS_IO_RESET_L	5 V PRECHARGE	IMIO_2	IMIO_3	Shield

NOTES:

1 Connector as seen from side 1 of backplane. (i.e., Disk Side)

2 Long pins shaded black - Length 7.25 mm

3 Medium pins shaded grey - Length 6.50 mm

4 All other pins short - Length 5.75 mm

D.8 Signal descriptions

ACT_LED_DR(15:0)_L

Signal is used to drive the activity LED associated with each disk drive. This is an active low signal, (i.e., the LED is on if the signal is low). The LEDs are pulled up to +5 VDC. The LED control circuitry shall be capable of sinking 15 mA at 0.4 V steady-state.

This signal should be driven by an open-drain output in order to prevent damage should two I/O modules inadvertently attempt to drive the signal at the same time. During RESET the I/O controller output should be set to a high impedance state.

The number of activity LED signals shall match the number of SATA ports supported by the controller. It is not mandatory to support 16 ports, any number between 1 and 16 SATA ports may be supported by this configuration.

MUX_DR(15:0)_L

Signal is used to control the Serial ATA path through a front-end multiplexer to a disk drive if implemented. If this signal is low, the multiplexer selects the Serial ATA path to the I/O controller in the option slot 0 identified with IO_Slot_0 connected to GND. This is a standard 3.3 V transistor transistor logic (TTL) level signal. This signal should be driven by an open-drain output in order to prevent damage should two I/O modules inadvertently attempt to drive the signal at the same time.

Serial ATA is a point-to-point, controller to disk drive, single initiator interface. Current Serial ATA devices do not support dual Serial ATA ports, therefore, no native capability exists to implement I/O controller failover (Active-Active or Active-Passive). An alternative method to provide a dual I/O controller access to the same disk drive is to implement a front-end multiplexer between the I/O controller and disk drive. This multiplexer allows only one I/O controller to own the complete access to the disk drive and implement an Active-Passive failover.

This signal shall not be implemented in single I/O controller enclosures, as all Serial ATA signals shall be directly connected to the single slot. The I/O controller needs to ensure it is capable of operating in a single I/O controller configuration without the presence of these signals.

The MUX_DRx_L controls on the I/O controller shall be capable of being reset to an OPEN or high impedance state using the I/O controller RESET signal. This is to allow the failover I/O controller to have direct RESET capability over the MUX_DRx_L controls in the event of the I/O controller failure.

The number of activity MUX_DR signals shall match the number of SATA ports supported by the controller. It is not mandatory to support 16 ports, any number between 1 and 16 SATA ports may be supported by this specification.

IO_SATA_Tx(15:0)±

Differential SATA signal pair that originates at the I/O controller (Tx) and is received by the SATA disk drive (Rx). It is not mandatory to support 16 ports, any number between 1 and 16 SATA ports may be supported by this specification.

These signals shall be 100 ohm differential characteristic impedance as per the nominal differential impedance given in Table 52.

IO_SATA_Rx(15:0)±

Differential SATA signal pair that originates at the SATA disk drive (Tx) and is received by the I/O controller (Rx). It is not mandatory to support 16 ports, any number between 1 and 16 SATA ports may be supported by this specification.

These signals shall be 100 ohm differential characteristic impedance. As per the nominal differential impedance given in Table 52.

IO_Slot_0_L

Active low signal that identifies the I/O controller location, 0 or 1. This signal shall be pulled low (GND) on the backplane for IO Slot 0 and high (3.3 VDC through a 10 kohm resistor) for IO Slot 1.

In a single I/O controller enclosure, this signal shall be connected to GND. In a dual controller configuration, the I/O controller should sense this line to determine in which slot it is located.

Ground

Signal and power ground of the I/O module.

5 V PRECHARGE

+5 VDC power that is available on the extended pins. This is used for pre-charging the I/O module.

The enclosure shall provide for a current limit of 4.5 A peak on each 5 V pre-charge pin (R = 1.1 ohm).

5 V

+5 VDC power that is available on standard length pins.

The enclosure shall be capable to supplying 10 A of 5 V current per I/O module.

3V3 PRECHARGE

+3.3 VDC power that is available on the extended pin. This is used for pre-charging the I/O controller 3.3 V circuitry. The enclosure shall provide for a current limit of 0.75 A on each 3.3 V pre-charge pin (R = 4.4 ohm).

3V3

+3.3 VDC power that is available on standard length pins.

The enclosure shall be capable of supplying 0.75 A of 3.3 V current per I/O module.

12 V PRECHARGE

+12 VDC power that is available on the extended pins. This is used for pre-charging the 12 V circuitry in the I/O Option slot module.

The enclosure shall be capable of supplying 2.4 A peak on each 12 V pre-charge pin (R = 5 ohm).

12 V

+12 VDC power that is available on standard length pins.

The enclosure shall be capable of supplying 1.0 A of current per I/O module.

FRU_ID_SCL

Clock signal of the FRU Identification 2-wire, serial bus. The backplane has a 1.1 kohm resistor pulled up to 3.3 V on this signal. The devices on this bus shall have open collector outputs. This 2-wire serial bus is used by the enclosure processor to gather information from all the FRUs located within the enclosure.

FRU_ID_SDA

Data signal of the FRU Identification 2-wire, serial bus. The backplane has a 1.1 kohm resistor pulled up to 3.3 V on this signal. The devices on this bus shall have open collector outputs. This 2-wire serial bus is used by the enclosure processor to gather information from all the FRUs located within the enclosure.

ENC_PROC_SCL

Clock signal of the SES processor 2-wire, serial bus. The SES processor has a 1.1 kohm resistor pulled up to 3.3 V on this signal. The devices on this bus shall have open collector outputs. This 2-wire serial bus is used by the I/O controller to talk to the enclosure processor.

ENC_PROC_SDA

Data signal of the SES processor 2-wire, serial bus. The SES processor has a 1.1 kohm resistor pulled up to 3.3 V on this signal. The devices on this bus shall have open collector outputs. This 2-wire serial bus is used by the I/O controller to talk to the enclosure processor.

HS_CHAN_Tx(1:0)

Differential transmit pair that may connect the I/O controller to either the other I/O controller or an I/O connector receive pair. Details of the uses of these signals are provided in Figure D.4 and Figure D.5 . It is recommended that these signals be 100 ohm differential characteristic impedance.

HS_CHAN_Rx(1:0)

Differential receive pair that may connect the I/O controller to either the other I/O controller or an I/O connector transmit pair. Details of the uses of these signals are provided in Figure D.4 and Figure D.5 . It is recommended that these signals be 100 ohm differential characteristic impedance.

ACT_LED_HOST_L

Host link activity LED control signal from the I/O controller for front panel mounted LED. Each port shall have a separate LED for the intended use of indication which controller port is active. Exact functionality may be defined by the vendor.

BATT_CHARGE

Signal is used to charge a battery back-up unit (BBU). The backplane shall support a maximum of 2 A on this trace.

BATT_DISCHARGE

Signal is used to discharge a BBU. The backplane supports a maximum of 2 A on this trace.

BATT_TEMP

Signal is used to report an analog voltage level that corresponds to a temperature level within a BBU. The backplane supports a maximum of 100 mA on this trace.

BATT_RETURN

Signal is used as the return (ground) path for a BBU. The backplane supports a maximum of 2 A on this trace.

THIS_IO_OK_L

Active low signal is used to determine if the I/O Option slot module is performing within specification. This signal is driven by the I/O controller. The I/O Option Slot module shall provide a pull-up for this line if it is inactive. The backplane should support a maximum of 100 mA on this trace.

OTHER_IO_OK_L

Active low signal is used to determine if the other I/O Option slot module in a dual controller configuration is performing within specification. This signal is an input to the I/O controller. The backplane should support a maximum of 100 mA on this trace.

OTHER_IO_RESET_L

Active low output signal that is used to reset the other I/O Option slot module in a dual controller system, located in the opposite slot. The backplane shall provide a pull-up for this line if it is inactive. This signal is cross-wired with the THIS_IO_RESET_L signal on dual controller backplanes. The backplane should support a maximum of 100 mA on this trace.

THIS_IO_RESET_L

Active low input signal that is used to reset the I/O Option slot module located in this slot. This signal is cross-wired with the OTHER_IO_RESET_L signal on dual controller backplanes. The backplane should support a maximum of 100 mA on this trace.

LOW_FREQ_RX_(1:0)

Signal line used for intermodule, 2-wire communications. This signal is cross-wired with the LOW_FREQ_TX signal on dual controller backplanes as shown in Figure D.6 . This is a low-speed signal line. Maximum frequency is 500 kHz. The actual implementation of this signal shall be decided by the I/O controller vendor. The backplane should support a maximum of 100 mA on this trace.

LOW_FREQ_TX_(1:0)

Signal line used for intermodule, 2-wire communications. This signal is cross-wired with the LOW_FREQ_RX signal on dual controller backplanes as shown in Figure D.6 . This is a low-speed signal line. Maximum frequency is 500 kHz. The actual implementation of this signal shall be decided by the I/O controller vendor. The backplane should support a maximum of 100 mA on this trace.

OTHER_IO_PRESENT_L

Active low signal is used to denote the presence of an option slot module in the opposite I/O option slot module. The I/O controller shall provide a pull-up for this line if it is inactive. This signal is cross-wired with the THIS_IO_PRESENT_L signal on the dual backplane only. This is a low-speed signal line. The actual implementation of this signal shall be decided by the I/O controller vendor. The backplane should support a maximum of 100 mA on this trace.

THIS_IO_PRESENT_L

Active low signal is used to denote the presence of an option slot module in the local I/O option slot module. The I/O controller shall provide a pull-up for this line if it is inactive. This signal is cross-wired with the OTHER_IO_PRESENT_L signal on the dual backplane only. This is a low-speed signal line. The actual implementation of this signal shall be decided by the I/O controller vendor. The backplane should support a maximum of 100 mA on this trace.

IMIO_(3:0)

These are direct connecting signals used for intermodule communication. The backplane should support a maximum of 100 mA on these traces. See Figure D.7 for details.

BOX_ID_BIT_(2:0)

Signal is used to determine the ID of the enclosure that it is installed.

Reserved (2 pins)

Two undefined pins J5 Row A pin 6, and J5 Row B pin 6that are not available for use at this time. No signals should be connected to these signals on either the backplane or the I/O Option Slot Module.

Appendix E. Jitter formulas without SSC (informative)

E.1 Clock to data

Consider the times if the clock edges occur. A perfect clock has edges that occur at multiples of a given period T. Associate an integer index with each clock edge. The times of ideal clock edges is expressed by

$$t_C(i) = i\,T$$

Data transitions always occur on a clock edge. Ideal data transitions occur at the same times as clock edges. In real systems, the data transitions do not occur at ideal times. The time error from ideal of the data transitions is called the "clock to edge jitter". This is expressed by

$$t_D(i) = i\,T + \varepsilon_i$$

where the ε_i are time deviations from ideal, the clock to edge jitter. Over time, these perturbations are constrained by

$$\lim_{N\to\infty} \sum_{i=0}^{N-1} \frac{\varepsilon_i}{N} = 0$$

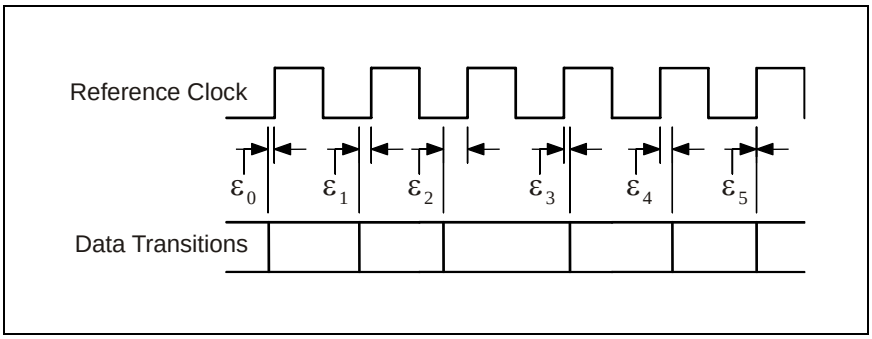


Figure E.1 – Jitter deviations

E.2 Data to data (shown for historical reasons)

The time difference between data transitions is shown in Figure E.2 and given by

$$t_D(j) - t_D(i) = \left(j - i\right)T + \varepsilon_j - \varepsilon_i$$

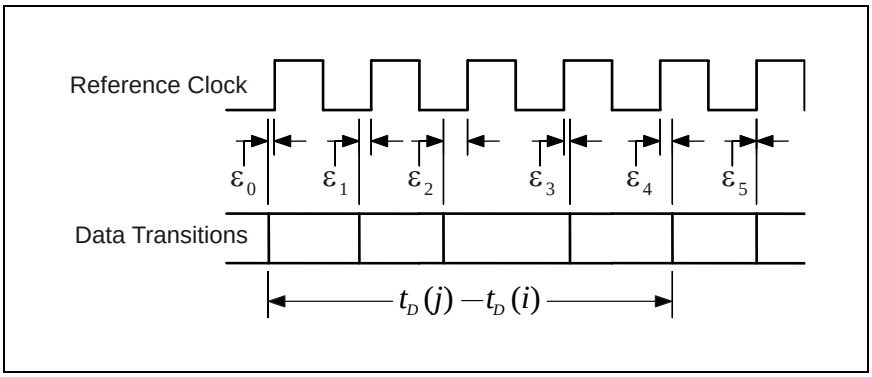


Figure E.2 – Edge to edge timing

A specification shall be invariant to if the measurement is taken so introduce a new quantity $k = (j - i)$ for the spacing between bits. The time difference between data transitions is

$$t_D(i+k) - t_D(i) = k\,T + \varepsilon_j - \varepsilon_i$$

Define a new quantity for the limits of this time difference – this is the jitter definition for Gen1i and Gen1m. The maximum and minimum is taken over all bit positions i that makes the jitter a function of only the bit spacing k.

$$t_J(k) = \max\left[t_D(i+k) - t_D(i)\right]_{\forall i} - \min\left[t_D(i+k) - t_D(i)\right]_{\forall i}$$

$$t_J(k) = \max\left[k\,T + \varepsilon_{i+k} - \varepsilon_i\right]_{\forall i} - \min\left[k\,T + \varepsilon_{i+k} - \varepsilon_i\right]_{\forall i}$$

NOTE 86 - Note that kT is a constant for each k, and is present in both the maximum and minimum terms.

Since the difference is taken, the terms cancel giving

$$t_J(k) = \max\left[\varepsilon_{i+k} - \varepsilon_i\right]_{\forall i} - \min\left[\varepsilon_{i+k} - \varepsilon_i\right]_{\forall i}$$

A distinct advantage of this jitter definition is the absence of dependence on the bit time T. This is not true if the clock is frequency modulated as in Spread Spectrum Clocking (SSC).

For a given separation of data transitions expressed in clock cycles, the maximum peak to peak deviation of the data transition spacing is the jitter. It is a function of the transition separation only, not the position of any particular transition. This jitter definition expresses the extreme separations of data transition times.