

# **Serial ATA International Organization**

## **Serial ATA Revision 3.2**

**7 August 2013  
Gold Revision**

### **SATA-IO Board Members**

**Dell Computer Corporation  
Hewlett Packard Corporation  
HGST  
Intel Corporation  
Marvell Semiconductor  
PMC-Sierra Inc.  
SanDisk Corporation  
Seagate Technology  
Western Digital Corporation**

The Serial ATA International Organization, Serial ATA Revision 3.2 specification ("Final 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 IMPLEMENTATION 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.

Copyright 2002 to 2013, Serial ATA International Organization. All rights reserved.

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

All product names are trademarks, registered trademarks, or servicemarks of their respective owners.

Serial ATA International Organization contact information

SATA-IO  
3855 SW 153<sup>rd</sup> Drive  
Beaverton, Oregon 97006 USA  
Tel +1 503-619-0572  
Fax +1 503-644-6708  
E-mail [admin@sata-io.org](mailto:admin@sata-io.org)

Click on  
page  
numbers

## **TABLE OF CONTENTS**

1	Revision history .....	27
1.1	Revision 2.5 (ratification date October 27, 2005) .....	27
1.2	Revision 2.6 (ratification date February 15, 2007) .....	27
1.3	Revision 3.0 (ratification date June 6, 2009) .....	27
1.4	Revision 3.1 (ratification date July 18, 2011) .....	28
1.5	Revision 3.2 (ratification date August 7, 2013) .....	29
2	Scope .....	31
3	Normative references .....	33
3.1	Normative references overview .....	33
3.2	Approved references .....	33
3.3	References under development .....	35
3.4	Other references .....	35
3.5	SATA-IO Style Guide .....	36
3.6	Design guides .....	36
4	Definitions, abbreviations, and conventions .....	37
4.1	Terminology .....	37
4.1.1	Definitions and abbreviations .....	37
4.1.2	Symbols and abbreviations .....	50
4.2	Conventions .....	53
4.2.1	Capitalization .....	53
4.2.2	Precedence .....	53
4.2.3	Keywords .....	53
4.2.4	Numbering .....	54
4.2.5	Dimensions .....	54
4.2.6	Signal conventions .....	54
4.2.7	State machine conventions .....	54
4.2.8	Byte, Word, Dword, and Qword Relationships .....	55
5	General overview .....	59
5.1	Connectivity .....	59
5.2	Architecture .....	60
5.3	Usage models .....	61
5.3.1	Usage models scope .....	61
5.3.2	Internal 1 m cabled host to device .....	68
5.3.3	Short backplane to device .....	68
5.3.4	Long backplane to device (obsolete) .....	69
5.3.5	Internal 4-lane cabled disk arrays .....	69
5.3.6	System-to-system interconnects – data center applications (xSATA) .....	70
5.3.7	System-to-system interconnects – external desktop applications (eSATA) .....	72
5.3.8	Proprietary Serial ATA disk arrays .....	73
5.3.9	Serial ATA and SAS .....	73
5.3.10	Potential external SATA incompatibility issues .....	74
5.3.11	Mobile applications .....	74
5.3.12	SATA Universal Storage Module (SATA USM) .....	77
5.3.13	Port Multiplier example applications .....	78
6	Cables and connectors .....	81
6.1	Cables and connectors overview .....	81
6.2	Internal cables and connectors .....	81
6.2.1	Internal single lane description .....	81
6.2.2	Connector locations .....	84
6.2.3	Mating interfaces .....	93
6.2.4	Signal cable receptacle connector .....	97
6.2.5	Signal host plug connector .....	100
6.2.6	Backplane connector .....	103

6.2.7	Power cable receptacle connector .....	106
6.2.8	Internal single lane cable material .....	108
6.2.9	Connector labeling .....	109
6.2.10	Connector and cable assembly requirements and test procedures .....	109
6.2.11	Internal Multilane cables .....	113
6.2.12	Mini SATA Internal Multilane .....	119
6.3	Internal Micro SATA connector for 1.8 inch HDD .....	126
6.3.1	Internal Micro SATA connector for 1.8 inch HDD overview .....	126
6.3.2	Usage model .....	126
6.3.3	General description .....	126
6.3.4	Connector location .....	126
6.3.5	Mating interfaces .....	129
6.4	Internal Slimline cables and connectors .....	135
6.4.1	Internal Slimline cables and connectors overview .....	135
6.4.2	Usage models .....	135
6.4.3	General description .....	136
6.4.4	Connector location and keep out zones .....	137
6.4.5	Mating interfaces .....	143
6.4.6	Backplane connector configuration and blind-mating tolerance .....	154
6.4.7	Connector labeling .....	155
6.4.8	Connector and cable assembly requirements and test procedures .....	155
6.5	Internal LIF-SATA connector for 1.8 inch HDD .....	156
6.5.1	Internal LIF-SATA connector for 1.8 inch HDD overview .....	156
6.5.2	General description .....	156
6.5.3	Connector locations .....	157
6.5.4	Mating interfaces .....	159
6.5.5	Internal LIF-SATA pin signal definition and contact mating sequence .....	162
6.5.6	Housing and contact electrical requirement .....	164
6.6	mSATA connector .....	164
6.6.1	mSATA connector overview .....	164
6.6.2	General description .....	164
6.6.3	Connector location on mSATA host .....	164
6.6.4	Mating interfaces .....	167
6.6.5	mSATA pin signal definition .....	170
6.7	SATA USM connector .....	174
6.7.1	SATA USM connector location .....	174
6.7.2	USM mating interfaces .....	178
6.8	SATA MicroSSD interface .....	182
6.8.1	SATA MicroSSD interface scope .....	182
6.8.2	SATA MicroSSD mechanical specification .....	183
6.8.3	SATA MicroSSD ballout - functional signal definition .....	188
6.9	Internal M.2 connector .....	194
6.9.1	Internal M.2 connector overview .....	194
6.9.2	M.2 mechanical (informative) .....	194
6.9.3	M.2 board connector (informative) .....	194
6.9.4	M.2 keys (informative) .....	199
6.9.5	M.2 sockets (informative) .....	199
6.9.6	M.2 land pattern for top mount connector motherboard (informative) .....	200
6.9.7	M.2 Z-height stack up (informative) .....	200
6.9.8	M.2 board sizes (informative) .....	201
6.9.9	M.2 component placement and board thickness (informative) .....	202
6.9.10	M.2 signal integrity .....	203
6.9.11	M.2 pad and anti-pad recommendations (informative) .....	203
6.9.12	M.2 minimum plane pull back from finger (informative) .....	204
6.9.13	M.2 socket 2 pin definition .....	205
6.9.14	M.2 SSD socket 3 .....	206

HIGH SPEED SERIALIZED AT ATTACHMENT  
 Serial ATA International Organization

6.9.15	M.2 electrical.....	209
6.9.16	M.2 signal definitions, configuration .....	209
6.9.17	M.2 mated connector differential impedance (informative) .....	209
6.10	SATA Express connector .....	210
6.10.1	SATA Express connector overview .....	210
6.10.2	SATA Express connector goals.....	210
6.10.3	SATA Express technical summary .....	210
6.10.4	SATA Express scope.....	212
6.10.5	SATA Express signal list.....	212
6.10.6	SATA Express mechanical .....	215
6.10.7	SATA Express device plug connector .....	216
6.10.8	SATA Express device cable receptacle connector.....	218
6.10.9	SATA Express host receptacle connector .....	221
6.10.10	SATA Express host plug connector.....	223
6.10.11	SATA Express host cable receptacle connector .....	226
6.10.12	Power dongle connector.....	228
6.10.13	SATA Express connector intermateability summary .....	231
6.10.14	SATA Express connector and cable electrical and mechanical requirements ....	231
6.10.15	SATA Express connector and cable assembly signal integrity requirements ....	231
6.10.16	SATA Express connector and cable shielding requirements for EMI.....	232
6.10.17	SATA Express connector and cable assembly DC electrical requirements.....	232
6.10.18	SATA Express connector and cable assembly mechanical and environmental requirements.....	232
6.11	External cables and connectors .....	232
6.11.1	External single lane .....	232
6.11.2	External Serial ATA component general descriptions .....	234
6.11.3	External Multilane .....	242
6.11.4	Mini SATA External Multilane .....	246
6.12	Cable and connector electrical specifications .....	249
6.12.1	Cable and connector electrical specifications overview .....	249
6.12.2	Serial ATA cable electrical requirements.....	250
6.12.3	Cable/connector test methodology.....	251
6.13	Hardware Feature Control (optional).....	259
6.13.1	Behavior.....	259
6.13.2	Electrical requirements specification .....	259
6.13.3	Device activity signal .....	261
6.13.4	Disable Staggered Spinup control .....	263
6.13.5	Micro SATA connector P7 definition (optional).....	264
6.14	Precharge and device presence detection .....	267
6.14.1	Precharge and device presence detection overview.....	267
6.14.2	Device requirements.....	267
6.14.3	Receptacle precharge (informative) .....	267
6.14.4	Presence detection (informative).....	269
7	Phy layer.....	271
7.1	Phy layer introduction .....	271
7.2	Descriptions of Phy electrical specifications.....	271
7.2.1	Terms overview .....	271
7.2.2	List of services .....	272
7.2.3	Low level electronics block diagrams (informative).....	272
7.2.4	Compliance testing .....	279
7.2.5	Link performance.....	280
7.3	SATA Express system electrical requirements .....	280
7.3.1	SATA Express system electrical requirements overview .....	280
7.3.2	SATA Express AC coupling capacitance requirement .....	281
7.3.3	SATA Express interface detect.....	281
7.3.4	PCIe sidebands .....	283

7.3.5	PCIe device power.....	284
7.4	Electrical specifications.....	284
7.4.1	Electrical specifications overview .....	284
7.4.2	Physical layer requirements tables.....	285
7.4.3	Phy layer requirements details .....	305
7.4.4	Loopback .....	320
7.4.5	Test pattern requirements.....	323
7.4.6	Hot plug considerations .....	348
7.4.7	Mated connector pair definition .....	350
7.4.8	Compliance interconnect channels (Gen3i, Gen3u).....	354
7.4.9	Impedance calibration (optional) .....	355
7.5	Jitter .....	356
7.5.1	Jitter overview .....	356
7.5.2	Jitter definition.....	357
7.5.3	Reference clock definition .....	357
7.5.4	Spread spectrum clocking .....	359
7.5.5	Jitter budget.....	361
7.6	Measurements .....	362
7.6.1	Measurements overview.....	362
7.6.2	Test fixtures .....	363
7.6.3	Frame error rate testing .....	371
7.6.4	Measurement of differential voltage amplitudes (Gen1, Gen2).....	374
7.6.5	Measurement of differential voltage amplitudes (Gen3i, Gen3u).....	385
7.6.6	Rise and fall times .....	387
7.6.7	Transmitter amplitude .....	388
7.6.8	Receive amplitude .....	390
7.6.9	Long term frequency accuracy .....	392
7.6.10	Jitter measurements .....	393
7.6.11	Transmit jitter (Gen1i, Gen1m, Gen1u, Gen2i, Gen2m, Gen2u).....	397
7.6.12	Transmit jitter (Gen3i, Gen3u).....	397
7.6.13	Receiver tolerance (Gen1i, Gen1m, Gen1u, Gen2i, Gen2m, Gen2u) .....	399
7.6.14	Receiver tolerance (Gen3i, Gen3u).....	401
7.6.15	Return loss and impedance balance .....	403
7.6.16	SSC profile.....	406
7.6.17	Intra-pair skew .....	407
7.6.18	Sequencing transient voltage .....	409
7.6.19	AC coupling capacitor.....	410
7.6.20	Tx amplitude imbalance.....	410
7.6.21	Tx rise/fall imbalance (obsolete).....	411
7.6.22	Tx AC common mode voltage (Gen2i, Gen2m) .....	411
7.6.23	Tx AC common mode voltage (Gen1u, Gen2u, Gen3i, Gen3u) .....	412
7.6.24	OOB common mode delta .....	412
7.6.25	OOB differential delta .....	412
7.6.26	Squelch detector tests .....	412
7.6.27	OOB signaling tests .....	414
7.6.28	TDR differential impedance (Gen1i, Gen1m, Gen1u) .....	415
7.6.29	TDR single-ended impedance (Gen1i, Gen1m).....	416
7.6.30	DC coupled common mode voltage (Gen1i) .....	417
7.6.31	AC coupled common mode voltage (Gen1i, Gen1m).....	417
7.6.32	Sequencing transient voltage - lab-load (Gen3i, Gen3u) .....	418
7.7	Interface states .....	419
7.7.1	Out-of-Band (OOB) signaling .....	419
7.7.2	Idle bus condition.....	427
7.8	Elasticity buffer management .....	427
8	OOB and phy power states .....	429
8.1	Interface power states .....	429

8.2	Asynchronous signal recovery (optional).....	429
8.2.1	Asynchronous signal recovery overview .....	429
8.2.2	Device Sleep and Asynchronous Signal Recovery .....	430
8.2.3	Unsolicited COMINIT usage (informative).....	430
8.3	OOB and Signature FIS return (informative).....	430
8.4	Power-on sequence state machine .....	430
8.4.1	Power-on sequence state machine overview .....	430
8.4.2	Host Phy initialization state machine .....	431
8.4.3	Device phy initialization state machine .....	436
8.4.4	Speed negotiation .....	440
8.5	DEVSLP signal protocol and timing.....	446
8.5.1	DEVSLP overview .....	446
8.5.2	DEVSLP signal electrical characteristics.....	448
9	Link layer .....	451
9.1	Link layer overview .....	451
9.1.1	Frame transmission .....	451
9.1.2	Frame reception.....	451
9.2	Encoding method.....	451
9.2.1	Encoding method overview .....	451
9.2.2	Notation and conventions .....	451
9.2.3	Character code .....	453
9.2.4	Transmission order summary .....	461
9.2.5	Reception summary.....	462
9.3	Transmission overview .....	464
9.4	Primitives .....	465
9.4.1	Primitives overview .....	465
9.4.2	Primitive disparity.....	465
9.4.3	Primitive handshakes.....	465
9.4.4	Primitive descriptions.....	465
9.4.5	Primitive encoding .....	467
9.4.6	DMAT <sub>P</sub> primitive.....	467
9.4.7	CONT <sub>P</sub> primitive.....	468
9.4.8	ALIGN <sub>P</sub> primitive .....	471
9.4.9	Flow control signaling latency.....	471
9.4.10	Examples of primitive usage (informative).....	474
9.5	CRC and scrambling.....	477
9.5.1	CRC and scrambling overview .....	477
9.5.2	Relationship between scrambling of FIS data and repeated primitives .....	477
9.5.3	Relationship between scrambling and CRC .....	477
9.5.4	Scrambling disable (informative) .....	478
9.6	Link layer state machine .....	478
9.6.1	Terms used in link layer transition tables .....	478
9.6.2	Link idle state machine .....	479
9.6.3	Link transmit state machine .....	482
9.6.4	Link receive state machine .....	490
9.6.5	Link power mode state machine .....	496
10	Transport layer.....	503
10.1	Transport layer overview .....	503
10.2	FIS construction.....	503
10.3	FIS decomposition .....	503
10.4	Frame information structure (FIS) .....	503
10.4.1	Frame information structure (FIS) overview .....	503
10.4.2	Payload content .....	504
10.5	FIS types.....	504
10.5.1	FIS types scope .....	504
10.5.2	FIS type values .....	504

10.5.3	CRC errors on data FISes .....	505
10.5.4	All FIS types.....	505
10.5.5	Register Host to Device FIS .....	506
10.5.6	Register Device to Host FIS .....	508
10.5.7	Set Device Bits - Device to Host FIS .....	510
10.5.8	DMA Activate - Device to Host .....	512
10.5.9	DMA Setup – Device to Host FIS or Host to Device FIS (bidirectional) .....	513
10.5.10	BIST Activate FIS - bidirectional.....	516
10.5.11	PIO Setup – Device to Host FIS .....	519
10.5.12	Data - Host to Device FIS or Device to Host FIS (bidirectional).....	522
10.6	Host transport states.....	524
10.6.1	Host transport states overview .....	524
10.6.2	Host transport idle state machine .....	524
10.6.3	Host transport transmit command FIS state machine .....	527
10.6.4	Host transport transmit control FIS state machine .....	528
10.6.5	Host transport transmit DMA Setup – Device to Host FIS or Host to Device FIS state machine .....	529
10.6.6	Host transport transmit BIST Activate FIS.....	530
10.6.7	Host transport decomposes Register FIS state machine .....	531
10.6.8	Host transport decomposes a Set Device Bits FIS state machine .....	532
10.6.9	Host transport decomposes a DMA Activate FIS state machine.....	533
10.6.10	Host transport decomposes a PIO Setup FIS state machine.....	536
10.6.11	Host transport decomposes a DMA Setup FIS state machine .....	539
10.6.12	Host transport decomposes a BIST Activate FIS state machine.....	540
10.7	Device transport states.....	541
10.7.1	Device transport idle state machine .....	541
10.7.2	Device transport sends Register Device to Host state machine .....	542
10.7.3	Device transport sends Set Device Bits FIS state machine .....	543
10.7.4	Device transport transmit PIO Setup – Device to Host FIS state machine .....	545
10.7.5	Device transport transmit DMA Activate FIS state machine.....	546
10.7.6	Device transport transmit DMA Setup – Device to Host FIS state machine .....	547
10.7.7	Device transport transmit Data – Device to Host FIS state machine .....	547
10.7.8	Device transport transmit BIST Activate FIS state machine.....	549
10.7.9	Device transport decomposes Register Host to Device FIS state machine.....	550
10.7.10	Device transport decomposes Data (Host to Device) FIS state machine .....	551
10.7.11	Device transport decomposes DMA Setup – Host to Device state machine .....	552
10.7.12	Device transport decomposes a BIST Activate FIS state machine.....	553
11	Device command layer protocol .....	555
11.1	Device command layer protocol overview .....	555
11.2	Power-on and COMRESET protocol.....	555
11.3	Device idle protocol .....	558
11.4	Software reset protocol.....	564
11.5	EXECUTE DEVICE DIAGNOSTIC command protocol.....	566
11.6	DEVICE RESET command protocol.....	568
11.7	Non-data command protocol .....	569
11.8	PIO data-in command protocol.....	570
11.9	PIO data-out command protocol.....	572
11.10	DMA data in command protocol .....	573
11.11	DMA data out command protocol .....	574
11.12	PACKET protocol.....	575
11.13	READ DMA QUEUED command protocol.....	582
11.14	WRITE DMA QUEUED command protocol.....	583
11.15	FPDMA QUEUED command protocol .....	585
12	Host command layer protocol.....	593
12.1	FPDMA QUEUED command protocol overview.....	593
12.2	FPDMA QUEUED command protocol.....	593



13	Application layer .....	599
13.1	Parallel ATA emulation .....	599
13.1.1	Parallel ATA emulation overview .....	599
13.1.2	Software reset.....	599
13.1.3	Master-only emulation .....	600
13.1.4	Master/Slave emulation (optional).....	601
13.2	IDENTIFY (PACKET) DEVICE .....	608
13.2.1	IDENTIFY (PACKET) DEVICE overview.....	608
13.2.2	IDENTIFY DEVICE .....	609
13.2.3	IDENTIFY PACKET DEVICE .....	617
13.2.4	Determining support for Serial ATA features.....	623
13.3	SET FEATURES.....	624
13.3.1	SET FEATURES overview .....	624
13.3.2	Enable/disable non-zero offsets in DMA Setup.....	624
13.3.3	Enable/disable DMA Setup FIS auto-activate optimization .....	624
13.3.4	Enable/disable device-initiated interface power state transitions.....	625
13.3.5	Enable/disable guaranteed in-order data delivery.....	625
13.3.6	Enable/disable asynchronous notification .....	625
13.3.7	Enable/disable software settings preservation .....	625
13.3.8	Enable/disable device automatic Partial to Slumber transitions.....	626
13.3.9	Enable Hardware Feature Control.....	626
13.3.10	Enable/disable Device Sleep.....	626
13.3.11	Enable/disable Hybrid Information.....	627
13.4	Device Configuration Overlay (obsolete).....	628
13.5	Software settings preservation (optional).....	629
13.5.1	Software settings preservation overview.....	629
13.5.2	Warm reboot considerations (informative).....	630
13.6	Native Command Queuing (optional).....	631
13.6.1	Native Command Queuing overview .....	631
13.6.2	Native Command Queuing (NCQ) Definition.....	631
13.6.3	Intermixing Non-Native Queued Commands and Native Queued Commands ...	635
13.6.4	READ FPDMA QUEUED.....	637
13.6.5	WRITE FPDMA QUEUED .....	643
13.6.6	NCQ NON-DATA.....	647
13.6.7	RECEIVE FPDMA QUEUED .....	669
13.6.8	SEND FPDMA QUEUED.....	674
13.6.9	First-party DMA HBA support (informative).....	682
13.7	SATA logs.....	683
13.7.1	SATA logs overview.....	683
13.7.2	Log address definitions.....	683
13.7.3	General purpose log directory (00h).....	683
13.7.4	Queued error log (10h) .....	685
13.7.5	NCQ NON-DATA log (12h).....	688
13.7.6	NCQ send and receive log (13h).....	691
13.7.7	Hybrid Information log (14h).....	693
13.7.8	Rebuild Assist log (15h).....	699
13.7.9	Identify Device data log (30h).....	701
13.8	Asynchronous notification (optional).....	717
13.8.1	Asynchronous notification overview .....	717
13.8.2	Set Device Bits FIS Notification bit .....	717
13.8.3	Notification mechanism.....	717
13.8.4	State machine for Asynchronous Notification.....	717
13.8.5	ATAPI notification .....	718
13.9	Phy event counters (optional).....	718
13.9.1	Phy event counters overview.....	718
13.9.2	Counter reset mechanisms.....	719

13.9.3	Counter identifiers.....	719
13.9.4	Phy Event Counters Log (11h) .....	723
13.10	Hardware Feature Control (optional) .....	724
13.11	Staggered spinup (optional) .....	724
13.12	Non-512 byte sector size (informative).....	725
13.13	Defect management (informative) .....	725
13.13.1	Defect management overview .....	725
13.13.2	Typical Serial ATA reliability metrics .....	726
13.13.3	An overview of Serial ATA defect management.....	726
13.13.4	Continuous background defect scanning (CBDS).....	727
13.13.5	Self-monitoring, analysis and reporting technology.....	727
13.14	Enclosure services/management (optional) .....	727
13.14.1	Enclosure services/management overview .....	727
13.14.2	Topology .....	728
13.14.3	Limitations.....	730
13.14.4	Definition.....	730
13.14.5	SES and SAF-TE extensions .....	736
13.14.6	Enclosure services hardware interface.....	742
13.15	HDD activity indication (optional).....	743
13.15.1	HDD activity indication overview .....	743
13.15.2	HDD activity emulation of desktop behavior.....	743
13.15.3	Activity/status indication reference (informative) .....	744
13.16	Port Multiplier discovery and enumeration .....	747
13.16.1	Power-up .....	747
13.16.2	Resets.....	747
13.16.3	Software initialization sequences (informative) .....	748
13.16.4	Port Multiplier discovery and device enumeration (informative).....	749
13.17	Automatic Partial to Slumber transitions.....	750
13.18	Serial ATA Link power management support.....	751
13.19	DHU specific operation (optional).....	751
13.20	Hybrid Information feature (optional).....	751
13.20.1	Hybrid Information feature overview.....	751
13.20.2	Hybrid Information field bits .....	753
13.20.3	Syncing .....	757
13.20.4	Interactions with ATA power management.....	757
13.20.5	Other Hybrid conditions .....	758
13.20.6	Automatic Disable.....	759
13.21	Rebuild Assist (optional).....	759
13.21.1	Rebuild Assist overview.....	759
13.21.2	Enabling Rebuild Assist feature.....	759
13.21.3	Using the Rebuild Assist feature .....	760
13.21.4	Disabling the Rebuild Assist feature.....	761
13.21.5	Testing the Rebuild Assist feature.....	761
14	Host adapter register interface .....	763
14.1	Host adapter register interface overview.....	763
14.2	Status and Control registers .....	763
14.2.1	Status and Control registers overview.....	763
14.2.2	SStatus register .....	764
14.2.3	SError register .....	765
14.2.4	SControl register.....	767
14.2.5	SActive register.....	768
14.2.6	SNotification register (optional) .....	768
15	Error handling .....	771
15.1	Architecture.....	771
15.2	Phy error handling overview .....	772
15.2.1	Error detection .....	772

15.2.2	Error control actions.....	773
15.2.3	Error reporting.....	773
15.3	Link layer error handling overview.....	774
15.3.1	Error detection.....	774
15.3.2	Error control actions.....	774
15.3.3	Error reporting.....	775
15.4	Transport layer error handling.....	775
15.4.1	Transport layer error handling overview.....	775
15.4.2	Error detection.....	776
15.4.3	Error control actions.....	776
15.4.4	Error reporting.....	777
15.5	Application layer error handling.....	778
15.5.1	Application layer error handling overview.....	778
15.5.2	Error detection.....	778
15.5.3	Error control actions.....	778
16	Port Multiplier.....	781
16.1	Introduction.....	781
16.2	Port Multiplier overview.....	781
16.3	Definition.....	782
16.3.1	Addressing mechanism.....	782
16.3.2	Device port requirements.....	782
16.3.3	Policies.....	784
16.4	Port Multiplier registers.....	795
16.4.1	Port Multiplier registers overview.....	795
16.4.2	General Status and Control registers.....	795
16.4.3	Port Status and Control registers.....	805
16.5	Port Multiplier command definitions.....	806
16.5.1	READ PORT MULTIPLIER.....	806
16.5.2	WRITE PORT MULTIPLIER.....	808
16.5.3	Interrupts.....	811
16.6	Controlling PM port value and interface power management.....	811
16.7	Switching types (informative).....	811
16.7.1	Switching types overview (informative).....	811
16.7.2	Command-based switching.....	812
16.7.3	FIS-based switching.....	812
17	Port Selector.....	813
17.1	Port Selector overview.....	813
17.2	Example applications.....	813
17.3	Port Selector introduction.....	814
17.4	Active Port Selection.....	815
17.4.1	Active Port Selection overview.....	815
17.4.2	Protocol-based Port Selection.....	815
17.4.3	Side-band Port Selection.....	819
17.4.4	Behavior during a change of active port.....	819
17.5	Behavior and policies.....	819
17.5.1	Control state machine.....	819
17.5.2	BIST support.....	824
17.5.3	Flow control signaling latency.....	824
17.5.4	Power management.....	824
17.5.5	OOB Phy signals.....	825
17.5.6	Hot plug.....	825
17.5.7	Speed negotiation.....	825
17.5.8	Spread spectrum clocking.....	826
17.6	Power-up and resets.....	826
17.6.1	Power-up.....	826
17.6.2	Resets.....	826

17.7	Host implementation (informative)	826
17.7.1	Software method for protocol-based selection overview	826
17.7.2	Software method for protocol-based selection	827
Appendix A.	Sample code for CRC and scrambling (informative)	829
A.1	CRC calculation	829
A.1.1	CRC overview	829
A.1.2	Maximum frame size	829
A.1.3	Example code for CRC algorithm overview	829
A.1.4	Example code for CRC algorithm	829
A.1.5	Example CRC implementation output	832
A.2	Scrambling calculation	832
A.2.1	Scrambling overview	832
A.2.2	Example code for scrambling algorithm	832
A.2.3	Example scrambler implementation	832
A.2.4	Example scrambler implementation output	835
A.3	Example frame	836
Appendix B.	Command processing overview (informative)	837
B.1	Non-data commands	837
B.2	DMA read by host from device	837
B.3	DMA write by host to device	837
B.4	PIO data read from the device	838
B.5	PIO data write to the device	838
B.6	ATA Tagged Command Queuing DMA read from device	839
B.7	ATA Tagged Command Queuing DMA write to device	840
B.8	ATAPI Packet commands with PIO data in	841
B.9	ATAPI Packet commands with PIO data out	842
B.10	ATAPI Packet commands with DMA data in	842
B.11	ATAPI Packet commands with DMA data out	843
B.12	Odd Word count considerations	844
B.12.1	Odd Word count considerations overview	844
B.12.2	DMA read from target for odd Word count	844
B.12.3	DMA write by host to target for odd Word count	844
B.13	PIO data read from the device	845
B.14	PIO data write to the device	845
B.15	Native Command Queuing examples	846
B.15.1	Native Command Queuing examples overview	846
B.15.2	Queued commands with Out of Order Completion	847
B.15.3	Interrupt aggregation	848
Appendix C.	Device emulation of nIEN with interrupt pending (informative)	851
Appendix D.	I/O controller module (informative)	853
D.1	I/O controller module overview	853
D.2	Supported configurations	854
D.2.1	Supported configurations overview	854
D.2.2	Single I/O controller signals	854
D.2.3	Dual I/O controller signals	855
D.2.4	Further optional features	855
D.3	Optional high speed channel configurations	856
D.4	Optional low speed channel configurations	858
D.5	I/O controller module connectors	859
D.5.1	I/O controller module connectors overview	859
D.5.2	I/O controller module cConnector	859
D.6	I/O controller module connector locations	862
D.7	Pinout listing	865
D.8	Signal descriptions	866
Appendix E.	Jitter formulas without SSC (informative)	873
E.1	Clock to data	873

E.2 Data to data (shown for historical reasons)..... 873

## LIST OF FIGURES

Figure 1 – State machine conventions .....	55
Figure 2 – Byte, Word, Dword, and Qword relationships .....	57
Figure 3 – Parallel ATA device connectivity .....	59
Figure 4 – Serial ATA connectivity .....	60
Figure 5 – Communication layers.....	61
Figure 6 – Internal 1 m cabled host to device application .....	68
Figure 7 – Short backplane to device application.....	69
Figure 8 – Internal 4-lane cabled disk array .....	70
Figure 9 – System-to-system data center interconnects.....	71
Figure 10 – External desktop application .....	72
Figure 11 – SATA disk arrays.....	73
Figure 12 – mSATA application.....	75
Figure 13 – Embedded LIF-SATA application.....	75
Figure 14 – Embedded SATA MicroSSD application .....	76
Figure 15 – Embedded M.2 application.....	77
Figure 16 – SATA USM application.....	77
Figure 17 – Enclosure example using Port Multipliers with Serial ATA as the connection within the rack.....	78
Figure 18 – Enclosure example using Port Multipliers with a different connection within the rack.....	79
Figure 19 – Mobile docking station example using a Port Multiplier .....	80
Figure 20 – Serial ATA connector examples.....	82
Figure 21 – SATA cable / connector connection diagram.....	83
Figure 22 – SATA host / device direct connection diagram .....	83
Figure 23 – Optical device plug connector location on 5.25 inch form factor .....	85
Figure 24 – Non-optical alternate device plug connector location on 5.25 inch form factor .....	86
Figure 25 – Device plug connector location on 3.5 inch side mounted device .....	87
Figure 26 – Device plug connector location on 3.5 inch bottom mounted device.....	88
Figure 27 – Device plug connector location on 2.5 inch side mounted device .....	89
Figure 28 – Device plug connector location on 2.5 inch bottom mounted device.....	90
Figure 29 – Device plug connector location on 1.8 inch side mounted device .....	91
Figure 30 – Device plug connector location on 1.8 inch bottom mounted device.....	92
Figure 31 – Device plug connector keep out zones .....	93
Figure 32 – Device plug connector.....	94
Figure 33 – Device plug connector (additional views).....	95
Figure 34 – Connector pin and feature locations .....	97
Figure 35 – Cable receptacle connector interface dimensions .....	99
Figure 36 – Latching signal cable receptacle (ClickConnect) .....	100
Figure 37 – Host signal plug connector interface dimensions.....	101
Figure 38 – Non-latching connector stack spacing and orientation .....	102
Figure 39 – Latching connector stack spacing and orientation .....	102
Figure 40 – Backplane connector interface dimensions .....	104
Figure 41 – Connector pair blind-mate misalignment tolerance.....	105
Figure 42 – Device-backplane mating configuration .....	106
Figure 43 – Power receptacle connector interface dimensions .....	107
Figure 44 – Latching power cable receptacle.....	108
Figure 45 – Detailed cross-section of an example internal single lane cable .....	109
Figure 46 – Isometric drawings of the internal 2 lane cable and connector .....	115
Figure 47 – Isometric drawings of the internal 4 lane cable and connector .....	115
Figure 48 – 4 lane pin assignments.....	116
Figure 49 – 4 lane to 4 x 1 lanes, fanout implementation .....	117
Figure 50 – 4 lane fanout pin assignments .....	118
Figure 51 – 2 lane fanout pin assignments .....	119
Figure 52 – Isometric drawings for Mini SATA Internal Multilane .....	120

Figure 53 – Mini SATA Internal Multilane connector pin assignments .....	122
Figure 54 – Mini SATA Internal Multilane system, symmetric cable implementation.....	123
Figure 55 – Mini SATA Internal Multilane system, controller based fanout cable implementation .....	124
Figure 56 – Mini SATA Internal Multilane system, backplane based fanout cable implementation .....	125
Figure 57 – Device internal Micro SATA connector location for 1.8 inch HDD .....	127
Figure 58 – Device internal Micro SATA connector location for 1.8 inch HDD .....	128
Figure 59 – Device internal Micro SATA plug connector .....	129
Figure 60 – Internal Micro SATA backplane connector.....	131
Figure 61 – Internal Micro SATA power receptacle connector.....	132
Figure 62 – Internal Micro SATA connector pair blind-mate misalignment capability.....	133
Figure 63 – 7.0 mm Slimline drive connector locations.....	138
Figure 64 – 8.5 mm Slimline drive connector locations (informative).....	139
Figure 65 – 9.5 mm/12.7 mm Slimline drive connector locations.....	140
Figure 66 – 7.0 mm Slimline drive connector location (section A-A).....	141
Figure 67 – 8.5 mm Slimline drive connector location (section A-A).....	141
Figure 68 – 9.5 mm Slimline drive connector location (section A-A).....	142
Figure 69 – 12.7 mm Slimline drive connector location (section A-A).....	142
Figure 70 – 7 mm Slimline device plug connector interface dimensions .....	144
Figure 71 – 7.0 mm Slimline device plug connector interface dimensions section A-A.....	145
Figure 72 – 7.0 mm Slimline device plug connector interface dimensions section B-B.....	145
Figure 73 – 7.0 mm Slimline device plug connector interface dimensions detail D .....	145
Figure 74 – 7.0 mm Slimline device plug connector interface dimensions optional hold down mounting.....	145
Figure 75 – Slimline device plug connector interface dimensions .....	147
Figure 76 – Slimline device plug connector interface dimensions (section A-A) .....	148
Figure 77 – Slimline device plug connector interface dimensions (section B-B) .....	148
Figure 78 – Slimline device plug connector interface dimensions (section C-C) .....	148
Figure 79 – Slimline device plug connector interface dimensions (detail F) .....	148
Figure 80 – Slimline device plug connector interface dimensions (section G-G).....	148
Figure 81 – Slimline device plug connector optional hold down mounting.....	148
Figure 82 – Slimline connector pin and feature locations .....	150
Figure 83 – Slimline power receptacle connector interface dimensions .....	151
Figure 84 – Slimline power receptacle connector option with latch .....	152
Figure 85 – Slimline power receptacle connector option with bump .....	152
Figure 86 – Slimline host receptacle connector interface dimensions .....	153
Figure 87 – Slimline host receptacle connector interface dimensions section C-C .....	153
Figure 88 – Slimline host receptacle connector interface dimensions section X-X.....	154
Figure 89 – Slimline host receptacle connector interface dimensions section Y-Y.....	154
Figure 90 – Slimline connector pair blind-mate misalignment tolerance.....	155
Figure 91 – Internal LIF-SATA connector location for 1.8 inch HDD .....	157
Figure 92 – Internal LIF-SATA connector location for 1.8 inch SSD bulk of single-sided mount type.....	158
Figure 93 – Device internal LIF-SATA embedded type connector.....	160
Figure 94 – Device internal LIF-SATA surface mounting type connector .....	161
Figure 95 – FPC for internal LIF-SATA .....	162
Figure 96 – mSATA card connector .....	165
Figure 97 – mSATA card footprint and keepout.....	166
Figure 98 – mSATA card connector location detail C .....	167
Figure 99 – mSATA card connector location detail W, Y and Z.....	168
Figure 100 – Device mSATA card type internal connector .....	169
Figure 101 – Bi-directional host-side implementation of P51 for compatibility with non-mSATA devices (informative) .....	173
Figure 102 – 14.5 mm SATA USM physical dimensions (see INF-8280) .....	175
Figure 103 – 9 mm SATA USM physical dimensions (see INF-8280) .....	177

Figure 104 – SATA USM vertical receptacle (see INF-8280) .....	179
Figure 105 – SATA USM vertical receptacle continued (see INF-8280).....	180
Figure 106 – SATA USM horizontal receptacle (see INF-8280) .....	181
Figure 107 – SATA USM horizontal receptacle continued (see INF-8280).....	182
Figure 108 – Footprint 1, SATA MicroSSD variant AC, 169 balls (informative).....	184
Figure 109 – Footprint 9, SATA MicroSSD variant AK and CB, 193 balls (informative) .....	185
Figure 110 – Footprint 10, SATA MicroSSD variant AL, 237 balls (informative) .....	186
Figure 111 – Footprint 11, SATA MicroSSD variant AM and CA, 156 balls (informative) .....	187
Figure 112 – Footprint 15, SATA MicroSSD variant AR, DB, and DC, 132 balls (informative) ..	188
Figure 113 – M.2 board connector top details (informative).....	195
Figure 114 – M.2 board connector bottom details (informative).....	196
Figure 115 – M.2 board connector top slot details (informative).....	197
Figure 116 – M.2 board connector bottom slot details (informative).....	198
Figure 117 – M.2 keys (informative).....	199
Figure 118 – M.2 land pattern for mother board (informative) .....	200
Figure 119 – M.2 single sided assembly – S2 profile (informative).....	200
Figure 120 – M.2 double sided assembly – D2 (informative).....	201
Figure 121 – M.2 double sided assembly – D3 (informative).....	201
Figure 122 – M.2 double sided assembly – D5 (informative).....	201
Figure 123 – M.2 board sizes (informative).....	202
Figure 124 – M.2 component profile and keep out zone (informative).....	203
Figure 125 – M.2 mother board pad and void dimensions (informative).....	204
Figure 126 – M.2 pull back (informative).....	204
Figure 127 – Example configurations for client .....	211
Figure 128 – SATA Express pinout for host receptacle connectors.....	213
Figure 129 – Examples of SATA Express connectors .....	215
Figure 130 – SATA Express device plug connector isometric drawing.....	216
Figure 131 – SATA Express device plug connector drawing .....	217
Figure 132 – SATA Express device cable receptacle connector drawing.....	219
Figure 133 – SATA Express host receptacle connector isometric drawing .....	221
Figure 134 – SATA Express host receptacle connector drawing.....	222
Figure 135 – SATA Express host plug connector pinout (isometric view) .....	224
Figure 136 – SATA Express host plug connector isometric drawing .....	225
Figure 137 – SATA Express host plug connector drawing.....	226
Figure 138 – SATA Express host cable receptacle connector isometric drawing.....	227
Figure 139 – SATA Express host cable receptacle connector drawing .....	227
Figure 140 – Example of power dongle usage.....	229
Figure 141 – Example power dongle to be used with the 15 pin SATA power connector .....	229
Figure 142 – Example power pigtail to be used with the 15 pin SATA power connector.....	230
Figure 143 – Usage model for HBA with external cable and single device enclosure.....	233
Figure 144 – Usage model for on-board Serial ATA connector with extension cable to external cable to disk.....	234
Figure 145 – Renderings of External Serial ATA cable receptacle and right angle plug .....	235
Figure 146 – Mechanical dimensions of External Serial ATA cable receptacle assembly.....	236
Figure 147 – Mechanical dimensions of External Serial ATA RA SMT plug.....	237
Figure 148 – Mechanical dimensions of External SATA RA SMT plug – reversed pin out.....	238
Figure 149 – Mechanical dimensions of External Serial ATA RA through-hole .....	239
Figure 150 – Mechanical dimensions of External Serial ATA vertical SMT plug .....	240
Figure 151 – Mechanical dimensions of External Serial ATA vertical through-hole plug.....	241
Figure 152 – External Multilane cable and connector .....	243
Figure 153 – Multilane cable connector blocking key locations .....	244
Figure 154 – Plug/receptacle keying .....	245
Figure 155 – Mini SATA External Multilane system, key features .....	247
Figure 156 – Mini SATA External Multilane system, key slots 7 for m level signals .....	248
Figure 157 – Mini SATA External Multilane connector pin assignments.....	249
Figure 158 – Example activity signal electrical block diagram .....	261



Figure 159 – Example host LED driver circuits .....	262
Figure 160 – Example circuit for disable staggered spinup .....	264
Figure 161 – Example activity signal electrical block diagram .....	265
Figure 162 – Example host LED driver circuits .....	266
Figure 163 – Typical precharge configuration .....	268
Figure 164 – Example presence detection implementation .....	270
Figure 165 – Physical plant overall block diagram (informative).....	273
Figure 166 – Analog Front End (AFE) block diagram .....	276
Figure 167 – Analog Front End (AFE) cabling .....	278
Figure 168 – The simplex link.....	279
Figure 169 – AC coupling capacitor requirements .....	281
Figure 170 – SATA Express interface detect mechanism.....	282
Figure 171 – Interface Detect, shutting down DC path after detection .....	283
Figure 172 – Common mode biasing examples for Gen1i (informative).....	306
Figure 173 – Common mode biasing for Gen1m, Gen1u, Gen2i, Gen2m, Gen2u, Gen3i, and Gen3u.....	307
Figure 174 – Common mode biasing for SATA MicroSSD .....	307
Figure 175 – Differential return loss limits .....	309
Figure 176 – Common mode return loss limits.....	310
Figure 177 – Impedance balance limits.....	311
Figure 178 – Differential return loss limits, Gen3i, Tx and Rx.....	311
Figure 179 – Signal rise and fall times .....	312
Figure 180 – Tx intra-pair skew .....	313
Figure 181 – OOB differential delta (at compliance point with AC coupling) .....	314
Figure 182 – Rx differential input Voltage conditions .....	317
Figure 183 – Rx intra-pair skew.....	318
Figure 184 – Far-End Retimed Loopback .....	321
Figure 185 – Far-End Analog Loopback .....	322
Figure 186 – Near-End Analog Loopback.....	323
Figure 187 – Compliant test patterns .....	324
Figure 188 – Example circuit for common mode transients .....	349
Figure 189 – Mated connector pair.....	350
Figure 190 – Mated connector pair, pin tail detail .....	351
Figure 191 – Mated connector pair for mSATA.....	352
Figure 192 – mSATA connector pin detail.....	353
Figure 193 – M.2 compliance point .....	353
Figure 194 – Compliance Interconnect Channel loss for Gen3i.....	355
Figure 195 – SSC profile example, triangular .....	360
Figure 196 – Spectral fundamental frequency comparison.....	361
Figure 197 – Lab-Load (LL).....	363
Figure 198 – Lab-Load (LL) for mSATA device.....	364
Figure 199 – Lab-Load (LL) for mSATA host .....	364
Figure 200 – Lab-Load (LL) for SATA USM host .....	365
Figure 201 – Lab-Load (LL) for SATA MicroSSD device .....	365
Figure 202 – Lab-Load (LL) for SATA MicroSSD host.....	366
Figure 203 – Lab-Load (LL) for M.2 device .....	366
Figure 204 – Lab-Load (LL) for M.2 host.....	367
Figure 205 – Lab-Sourced Signal (LSS) .....	367
Figure 206 – Lab-Sourced Signal (LSS) for mSATA device .....	368
Figure 207 – Lab-Sourced Signal (LSS) for mSATA host.....	368
Figure 208 – Lab-Sourced Signal (LSS) for SATA USM host.....	369
Figure 209 – Lab-Sourced Signal (LSS) for SATA MicroSSD device .....	369
Figure 210 – Lab-Sourced Signal (LSS) for SATA MicroSSD host.....	370
Figure 211 – Lab-Sourced Signal (LSS) for M.2 device.....	370
Figure 212 – Lab-Sourced Signal (LSS) for M.2 host .....	371
Figure 213 – Differential Voltage amplitude measurement .....	374

Figure 214 – Differential Voltage amplitude measurement pattern example .....	374
Figure 215 – LFTP Pattern on High BW Scope (HBWS) .....	383
Figure 216 – Single ended rise and fall time .....	387
Figure 217 – Transmit amplitude test with Lab-Load (LL).....	389
Figure 218 – Transmit amplitude test with CIC .....	389
Figure 219 – Receiver amplitude test, setting levels .....	390
Figure 220 – Receiver amplitude test.....	390
Figure 221 – Voltage at receiver input .....	391
Figure 222 – Tx long term frequency measurement .....	393
Figure 223 – Receiver model for jitter .....	393
Figure 224 – Jitter at receiver.....	394
Figure 225 – Jitter at receiver, high pass function.....	394
Figure 226 – JTF and CLTF definition.....	395
Figure 227 – Transmitter jitter test (Gen1i, Gen2i).....	397
Figure 228 – Transmitter jitter test at Tx (Gen3i) .....	398
Figure 229 – Transmitter jitter test through the CIC (Gen3i).....	398
Figure 230 – Receiver jitter and CM tolerance test – setting levels .....	400
Figure 231 – Receiver jitter and CM tolerance test .....	400
Figure 232 – Receiver jitter and CM tolerance test – setting RJ level (Gen3i) .....	402
Figure 233 – Receiver jitter and CM tolerance test – setting TJ and CM levels (Gen3i) .....	402
Figure 234 – Receiver jitter and CM tolerance test (Gen3i).....	403
Figure 235 – Return loss test-calibration.....	404
Figure 236 – Return loss test .....	406
Figure 237 – Intra-pair skew test for a transmitter.....	408
Figure 238 – Receiver intra-pair skew test – setting levels.....	408
Figure 239 – Receiver intra-pair skew test.....	408
Figure 240 – Example intra-pair skew test for transmitter (10.8 ps) .....	409
Figure 241 – Tx/Rx sequencing transient Voltage measurement .....	409
Figure 242 – AC coupled capacitance measurement .....	410
Figure 243 – Squelch detector threshold test – setting levels.....	413
Figure 244 – Squelch detector threshold test.....	413
Figure 245 – TDR differential impedance test – setting risetime .....	415
Figure 246 – TDR impedance test.....	416
Figure 247 – TDR single-ended impedance test – setting risetime .....	416
Figure 248 – DC coupled common mode Voltage measurement .....	417
Figure 249 – AC coupled common mode Voltage measurement .....	418
Figure 250 – TDR impedance test.....	418
Figure 251 – Sequencing transient Voltage lab-load (LL).....	419
Figure 252 – OOB signals .....	420
Figure 253 – Transmitter examples.....	421
Figure 254 – Transmitter examples (concluded).....	422
Figure 255 – COMRESET sequence .....	423
Figure 256 – COMINIT sequence.....	424
Figure 257 – OOB signal detector.....	426
Figure 258 – Squelch detector .....	427
Figure 259 – Host Phy initialization state machine.....	432
Figure 260 –Device Phy initialization state machine .....	436
Figure 261 – Power-on sequence .....	441
Figure 262 – Speed negotiation using COMRESET (RSN).....	442
Figure 263 – PHYRDY to Partial – host initiated.....	444
Figure 264 – PHYRDY to Partial – device initiated .....	445
Figure 265 – DEVSLP protocol overview .....	446
Figure 266 – Example DEVSLP electrical block diagram .....	448
Figure 267 – Nomenclature reference.....	452
Figure 268 – Bit ordering and significance .....	462
Figure 269 – Transmission structures .....	464

Figure 270 – Link layer state machine .....	479
Figure 271 – Link layer transmit state machine.....	483
Figure 272 – Link layer receiver state machine.....	490
Figure 273 – Link layer power mode state machine.....	497
Figure 274 – FIS type value assignments .....	505
Figure 275 – Register Host to Device FIS layout .....	506
Figure 276 – Register Device to Host FIS layout .....	508
Figure 277 – Set Device Bits – Device to Host FIS layout .....	510
Figure 278 – DMA Activate – Device to Host FIS layout.....	512
Figure 279 – DMA Setup – Device to Host or Host to Device FIS layout .....	513
Figure 280 – BIST Activate FIS – bidirectional.....	516
Figure 281 – PIO Setup – Device to Host FIS layout .....	519
Figure 282 – Data – Host to Device or Device to Host FIS layout .....	522
Figure 283 – Host transport idle state machine.....	524
Figure 284 – Host transport command state machine .....	527
Figure 285 – Host transport control state machine .....	528
Figure 286 – Host transport DMA setup state machine .....	529
Figure 287 – Host transport transmit BIST state machine .....	530
Figure 288 – Host transport register state machine .....	531
Figure 289 – Host transport set device bits state machine .....	532
Figure 290 – Host transport DMA activate state machine.....	533
Figure 291 – Host transport PIO setup state machine .....	536
Figure 292 – Host transport DMA setup state machine .....	539
Figure 293 – Host transport BIST state machine .....	540
Figure 294 – Device transport idle state machine .....	541
Figure 295 – Host transport register device to host state machine.....	543
Figure 296 – Device transport setup device bits state machine.....	544
Figure 297 – Device transport PIO setup state machine .....	545
Figure 298 – Device transport DMA activate state machine .....	546
Figure 299 – Device transport DMA setup device to host state machine .....	547
Figure 300 – Device transport transmit data device to host state machine .....	548
Figure 301 – Device transport transmit BIST state machine.....	549
Figure 302 – Device transport register host to device state machine .....	550
Figure 303 – Device transport data host to device state machine .....	551
Figure 304 – Device transport DMA setup host to device state machine .....	552
Figure 305 – Device transport BIST state machine.....	553
Figure 306 – Device command layer reset state machine .....	555
Figure 307 – DHR2: Send_good_status, PACKET command feature set not implemented .....	556
Figure 308 – DHR2: Send_good_status, PACKET command feature set implemented .....	556
Figure 309 – DHR3: Send_bad_status, PACKET command feature set not implemented .....	557
Figure 310 – DHR3: Send_bad_status, PACKET command feature set not implemented .....	557
Figure 311 – Device command layer idle state machine.....	558
Figure 312 – Device command layer software reset state machine .....	564
Figure 313 – DSR2: Send_good_status, PACKET command feature set not implemented .....	565
Figure 314 – DSR2: Send_good_status, PACKET command feature set not implemented .....	565
Figure 315 – DSR3: Send_bad_status, PACKET command feature set not implemented .....	566
Figure 316 – DSR3: Send_bad_status, PACKET command feature set implemented .....	566
Figure 317 – Device command layer diagnostic state machine .....	566
Figure 318 – DEDD1: Send_good_status, PACKET command feature set not implemented....	567
Figure 319 – DEDD1: Send_good_status, PACKET command feature set not implemented....	567
Figure 320 – DEDD2: Send_bad_status, PACKET command feature set not implemented.....	568
Figure 321 – DEDD2: Send_bad_status, PACKET command feature set implemented.....	568
Figure 322 – Device command layer reset state machine .....	568
Figure 323 – Device command layer non-data state machine .....	569
Figure 324 – Device command layer non-data state machine .....	569
Figure 325 – Device command layer PIO data-in state machine.....	570

Figure 326 – Device command layer PIO data-out state machine.....	572
Figure 327 – Device command layer DMA data in state machine .....	573
Figure 328 – Device command layer DMA data out state machine .....	574
Figure 329 – Device command layer packet state machine .....	576
Figure 330 – Device command layer read DMA queued state machine.....	582
Figure 331 – Device command layer write DMA queued state machine .....	584
Figure 332 – Device command layer FPDMA queued state machine .....	586
Figure 333 – Host command layer FPDMA queued state machine .....	593
Figure 334 – Host application layer Parallel ATA emulation state machine.....	603
Figure 335 – DMA Setup FIS definition for memory buffer selection .....	633
Figure 336 – READ FPDMA QUEUED command definition .....	637
Figure 337 – Set Device Bits FIS for successful READ FPDMA QUEUED command completion .....	638
Figure 338 – READ FPDMA QUEUED error on command receipt.....	640
Figure 339 – Set Device Bits FIS with error notification, and command completions.....	640
Figure 340 – Set Device Bits FIS aborting all outstanding commands .....	642
Figure 341 – WRITE FPDMA QUEUED command definition .....	643
Figure 342 – Set Device Bits FIS for successful WRITE FPDMA QUEUED command completion .....	644
Figure 343 – WRITE FPDMA QUEUED error on command receipt .....	645
Figure 344 – Set Device Bits FIS with error notification, and command completions.....	646
Figure 345 – NCQ NON-DATA - command definition .....	647
Figure 346 – ABORT NCQ QUEUE – subcommand definition .....	649
Figure 347 – ABORT NCQ QUEUE – successful completion.....	650
Figure 348 – ABORT NCQ QUEUE – error on command receipt.....	651
Figure 349 – ABORT NCQ QUEUE – error during processing .....	651
Figure 350 – DEADLINE HANDLING – subcommand definition .....	653
Figure 351 – DEADLINE HANDLING – successful completion .....	654
Figure 352 – DEADLINE HANDLING – error on command receipt .....	654
Figure 353 – DEADLINE HANDLING – error during processing .....	655
Figure 354 – HYBRID CHANGE BY LBA RANGE – subcommand definition .....	657
Figure 355 – HYBRID CHANGE BY LBA RANGE – successful completion .....	657
Figure 356 – HYBRID CHANGE BY LBA RANGE – error on command receipt .....	658
Figure 357 – HYBRID CHANGE BY LBA RANGE – error during processing.....	659
Figure 358 – HYBRID CONTROL – subcommand definition .....	660
Figure 359 – HYBRID CONTROL – successful completion.....	661
Figure 360 – HYBRID CONTROL – error on command receipt.....	662
Figure 361 – HYBRID CONTROL – error during processing .....	662
Figure 362 – HYBRID DEMOTE BY SIZE – command definition .....	664
Figure 363 – HYBRID DEMOTE BY SIZE – successful completion .....	665
Figure 364 – HYBRID DEMOTE BY SIZE – error on command receipt.....	666
Figure 365 – HYBRID DEMOTE BY SIZE – error during processing .....	666
Figure 366 – SET FEATURES QUEUED subcommand = 05h.....	668
Figure 367 – RECEIVE FPDMA QUEUED command definition .....	669
Figure 368 – Set Device Bits FIS for successful RECEIVE FPDMA QUEUED command completion .....	670
Figure 369 – RECEIVE FPDMA QUEUED error status result values on command receipt.....	671
Figure 370 – Set Device Bits FIS with error notification, and command completions.....	671
Figure 371 – RECEIVE FPDMA QUEUED subcommand = 01h.....	673
Figure 372 – SEND FPDMA QUEUED command definition .....	674
Figure 373 – Set Device Bits FIS for successful SEND FPDMA QUEUED command completion .....	675
Figure 374 – SEND FPDMA QUEUED error status result values on command receipt.....	676
Figure 375 – Set Device Bits FIS with error notification, and command completions.....	677
Figure 376 – SEND FPDMA QUEUED, subcommand = 00h .....	678
Figure 377 – Subcommand specific parameters for the HYBRID EVICT subcommand = 01h ..	680

Figure 378 – Output data from the host for the HYBRID EVICT command.....	680
Figure 379 – SEND FPDMA QUEUED subcommand = 02h .....	681
Figure 380 – Example DMA engine indirection for First-party DMA support .....	682
Figure 381 – Queued Error Log data structure definition.....	686
Figure 382 – NCQ NON-DATA log (12h) data structure definition.....	688
Figure 383 – NCQ SEND and RECEIVE log (13) data structure definition.....	691
Figure 384 – Hybrid Information log data .....	693
Figure 385 – Hybrid Information Header .....	693
Figure 386 – Hybrid Information Descriptor .....	697
Figure 387 – Rebuild Assist log.....	700
Figure 388 – Transitional Energy, Slumber to DevSleep and DevSleep to PHYRDY .....	711
Figure 389 – Transitional Energy, PM0: Active to Off and Off to PM0: Active .....	712
Figure 390 – Transitional Energy, PM0: Active to PM2: Standby and PM2: Standby to PM0: Active.....	715
Figure 391 – Transitional Energy, DevSleep/PM2: Standby to PM0: Active and Off to DHR2:Send_good_status.....	716
Figure 392 – Asynchronous notification state machine.....	718
Figure 393 – Phy Event Counter identifiers.....	720
Figure 394 – Phy event counters log data structure definition .....	723
Figure 395 – Generic enclosure services topology .....	729
Figure 396 – Simplified view of generic topology .....	729
Figure 397 – Enclosure services definition configuration .....	730
Figure 398 – Register signature indicating presence of enclosure services device .....	731
Figure 399 – Register signature for absent enclosure processor .....	731
Figure 400 – Command Block Register fields used in enclosure processor communications....	732
Figure 401 – I <sup>2</sup> C frame for conveying an enclosure services command.....	733
Figure 402 – WRITE SEP command block registers .....	734
Figure 403 – I <sup>2</sup> C transactions corresponding to a WRITE SEP command .....	735
Figure 404 – READ SEP command block registers .....	735
Figure 405 – I <sup>2</sup> C transactions corresponding to READ SEP command.....	736
Figure 406 – IDENTIFY SEP data structure definition .....	737
Figure 407 – SAF-TE write device slot status data structure .....	739
Figure 408 – SES device element data structure.....	739
Figure 409 – Example subsystem .....	740
Figure 410 – SAF-TE device ID field convention .....	741
Figure 411 – SES slot address field convention.....	741
Figure 412 – Activity LED definition for desktop behavior emulation .....	743
Figure 413 – Device activity LEDs with separate wires.....	744
Figure 414 – Device activity LEDs with ribbon cable .....	745
Figure 415 – Device activity LEDs in a storage subsystem .....	746
Figure 416 – Software reset to control port result values.....	748
Figure 417 – Port Multiplier signature .....	749
Figure 418 – Hybrid Information field .....	754
Figure 419 – SStatus register definition .....	764
Figure 420 – SError register definition .....	765
Figure 421 – ERR field definition.....	765
Figure 422 – DIAG field definition.....	766
Figure 423 – SControl register definition.....	767
Figure 424 – SActive register definition.....	768
Figure 425 – SNotification register definition.....	769
Figure 426 – Error handling architecture.....	771
Figure 427 – Port Multiplier overview .....	781
Figure 428 – Port Multiplier collisions state machine .....	786
Figure 429 – Host port hot plug state machine .....	788
Figure 430 – Device port hot plug state machine.....	790
Figure 431 – Register values for an unsupported command .....	792

Figure 432 – READ PORT MULTIPLIER command definition.....	806
Figure 433 – READ PORT MULTIPLIER success status result values .....	807
Figure 434 – READ PORT MULTIPLIER error status result values .....	808
Figure 435 – WRITE PORT MULTIPLIER command definition .....	809
Figure 436 – WRITE PORT MULTIPLIER success status result values .....	810
Figure 437 – WRITE PORT MULTIPLIER error status result values .....	811
Figure 438 – Port Selector overview .....	813
Figure 439 – Example failover application with two hosts.....	814
Figure 440 – Port selection signal based on assertion of COMRESET to assertion of following COMRESET .....	816
Figure 441 – Complete port selection signal consisting of two sequences with requisite inter-reset spacings.....	816
Figure 442 – Host port Phy state machine enhancements .....	818
Figure 443 – Control state machine .....	820
Figure 444 – Phy block diagram.....	821
Figure 445 – Port Selector state machine .....	822
Figure D.1 – Concept summary interconnect structure.....	853
Figure D.2 – An example of signal connections with one I/O controller.....	854
Figure D.3 – Example of signal connections with two I/O modules .....	856
Figure D.4 – High speed channels – configuration 0 .....	857
Figure D.5 – High-speed channels – configuration 1 .....	857
Figure D.6 – Low speed channels .....	858
Figure D.7 – Interconnect channels .....	858
Figure D.8 – I/O controller module connector rendering .....	859
Figure D.9 – Connector pin layout and pin lengths .....	860
Figure D.10 – I/O controller module connector receptacle engineering drawing .....	861
Figure D.11 – Side view of connector .....	861
Figure D.12 – I/O controller module connector locations on 1xWide I/O module .....	863
Figure D.13 – I/O controller module connector locations on 2xWide I/O module .....	864
Figure D.14 – Jitter deviations .....	873
Figure D.15 – Edge to edge timing.....	874

## **LIST OF TABLES**

Table 1 – Usage model descriptions .....	62
Table 2 – Usage model electrical requirements .....	64
Table 3 – SATA Express usage models .....	66
Table 4 – SATA Express usage model electrical requirements .....	67
Table 5 – Standard SATA connector (3.5 inch & 2.5 inch HDD) .....	96
Table 6 – Allowed values for dimension A and B for device-to-backplane mating .....	106
Table 7 – Housing and contact electrical parameters, test procedures, and requirements .....	110
Table 8 – Mechanical test procedures and requirements .....	111
Table 9 – Environmental parameters, test procedures, and requirements .....	112
Table 10 – Additional requirement .....	112
Table 11 – Connector test sequences .....	113
Table 12 – Signal and power Internal Micro SATA plug and nominal mate sequence .....	134
Table 13 – Unique connector mechanical testing procedures and requirements .....	135
Table 14 – Slimline connector location references .....	138
Table 15 – Slimline device plug connector pin definition .....	149
Table 16 – Slimline connector mechanical test procedures and requirements .....	156
Table 17 – Signal and power internal LIF-SATA plug .....	163
Table 18 – Unique connector mechanical testing procedures and requirements .....	164
Table 19 – Signal assignments for mSATA .....	171
Table 20 – 14.5 mm SATA USM physical dimensions (see INF-8280) .....	176
Table 21 – 9 mm SATA USM physical dimensions (see INF-8280) .....	178
Table 22 – Signal assignments for SATA MicroSSD .....	189
Table 23 – M.2 connector, electrical requirements .....	203
Table 24 – M.2 device side signal assignments for key B (1x SATA, 2x PCIe) .....	205
Table 25 – M.2 signal assignments for card keyed for slot M (1x SATA, 1x, 2x, or 4x PCIe) .....	207
Table 26 – M.2 voltage and current requirements .....	209
Table 27 – M.2 config pin settings .....	209
Table 28 – SATA Express signal list summary .....	212
Table 29 – SATA Express pin group table (for host receptacle and device plug connectors) .....	212
Table 30 – SATA Express signal pin list for host receptacle and device plug connectors .....	213
Table 31 – SATA Express power pin list for host receptacle and device plug connectors .....	214
Table 32 – SATA Express host plug connector pin list .....	224
Table 33 – SATA Express cable wire connection .....	225
Table 34 – SATA Express connector intermatability summary .....	231
Table 35 – Multilane pin assignments .....	246
Table 36 – Internal cable / connector measurement parameter and requirements .....	250
Table 37 – External Single Lane cable / connector measurement parameter and requirements .....	251
Table 38 – Limited External Multilane cable / connector measurement parameter and requirements .....	251
Table 39 – Common interconnect measurement procedure methodologies .....	253
Table 40 – Interconnect test methodologies / procedures .....	254
Table 41 – DSS and DAS for various connectors .....	259
Table 42 – Electrical requirements for DHU .....	260
Table 43 – Power segment pin P11 activity signal electrical parameters .....	262
Table 44 – Host activity signal electrical parameters .....	262
Table 45 – Activity signal functional states .....	263
Table 46 – Host staggered spinup control electrical requirements .....	263
Table 47 – Micro SATA connector P7 activity signal electrical parameters .....	265
Table 48 – Host activity signal electrical parameters .....	265
Table 49 – Activity signal functional states .....	267
Table 50 – Comparator voltages for alternate example presence detection circuit .....	270
Table 51 – Logic states of Vsense .....	282
Table 52 – General specifications .....	285

Table 53 – Transmitter specifications.....	287
Table 54 – Transmitted signal requirements .....	290
Table 55 – UHost transmitted signal requirements .....	294
Table 56 – Receiver specifications.....	296
Table 57 – Lab-Sourced Signal (for receiver tolerance testing).....	299
Table 58 – Lab-Sourced Signal (for UHost receiver tolerance testing).....	301
Table 59 – OOB specifications .....	303
Table 60 – Loopback modes .....	320
Table 61 – Low Transition Density Pattern (LTDP) starting with RD-.....	326
Table 62 – Low Transition Density Pattern (LTDP) starting with RD+ .....	327
Table 63 – High Transition Density Pattern (HTDP) starting with RD-.....	328
Table 64 – High Transition Density Pattern (HTDP) starting with RD+.....	329
Table 65 – Low Frequency Spectral Content Pattern (LFSCP) starting with RD- .....	330
Table 66 – Low Frequency Spectral Content Pattern (LFSCP) starting with RD+.....	331
Table 67 – Simultaneous Switching Outputs Pattern (SSOP) starting with RD-.....	332
Table 68 – Simultaneous Switching Outputs Pattern (SSOP) starting with RD+.....	332
Table 69 – Lone Bit Pattern (LBP) starting with RD-.....	333
Table 70 – Lone Bit Pattern (LBP) starting with RD+ .....	333
Table 71 – Composite-Bit Pattern (COMP) starting with RD- .....	335
Table 72 – Composite-Bit Pattern (COMP) starting with RD+ .....	338
Table 73 – Framed Composite Pattern (FCOMP).....	341
Table 74 – Frame Error Rate confidence levels versus sample size .....	372
Table 75 – Bit Error Rate confidence levels versus sample size .....	373
Table 76 – OOB signal times.....	420
Table 77 – ALIGN <sub>P</sub> .....	428
Table 78 – Interface power states .....	429
Table 79 – DevSleep timing parameters .....	446
Table 80 – Device side DEVSLP electrical parameters .....	448
Table 81 – Host side DEVSLP electrical parameters.....	449
Table 82 – Bit designations .....	452
Table 83 – Conversion examples .....	453
Table 84 – 5b/6b coding .....	455
Table 85 – 3b/4b coding .....	455
Table 86 – Encoding examples .....	456
Table 87 – Valid data characters.....	457
Table 88 – Valid control characters.....	461
Table 89 – Single bit error with two character delay .....	463
Table 90 – Single bit error with one character delay .....	463
Table 91 – Description of primitives .....	466
Table 92 – Primitive encoding .....	467
Table 93 – CONTP usage example.....	469
Table 94 – Example of components of a round trip delay.....	473
Table 95 – SRST write from host to device transmission breaking through a device to host data FIS.....	474
Table 96 – Command shadow register block register transmission example .....	475
Table 97 – Data from host to device transmission example .....	476
Table 98 – Simplified shadow register block register numbering.....	504
Table 99 – BIST Activate FIS modes and bit settings .....	517
Table 100 – IDENTIFY DEVICE information.....	609
Table 101 – IDENTIFY PACKET DEVICE information .....	617
Table 102 – Features enable/disable values.....	624
Table 103 – Feature identification values.....	624
Table 104 – Hardware Feature Control pin definitions.....	626
Table 105 – Priority .....	635
Table 106 – Subcommands for NCQ NON-DATA .....	648
Table 107 – Abort Type .....	650



HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

Table 108 – Cache Behavior .....	656
Table 109 – HYBRID DEMOTE BY SIZE – number of logical sectors affected .....	664
Table 110 – Subcommands for RECEIVE FPDMA QUEUED .....	672
Table 111 – Subcommands for SEND FPDMA QUEUED .....	677
Table 112 – Log addresses for Serial ATA .....	683
Table 113 – General purpose log directory values for Serial ATA .....	684
Table 114 – Hybrid Information Enabled .....	694
Table 115 – Hybrid Health .....	694
Table 116 – Supported Options .....	696
Table 117 – Rebuild Assist Enabled .....	700
Table 118 – Serial ATA (page 08h) .....	701
Table 119 – Coded values for negotiated Serial ATA signaling speed .....	708
Table 120 – Transition Time Units .....	711
Table 121 – Power Cycle Time Units .....	713
Table 122 – Power units .....	714
Table 123 – Pins used by Hardware Feature Control .....	724
Table 124 – Current power source interactions .....	757
Table 125 – SCR definition .....	763
Table 126 – SCR definition .....	764
Table 127 – Static information registers .....	796
Table 128 – Status information and control registers .....	798
Table 129 – Features Supported registers .....	802
Table 130 – Features Enabled registers .....	803
Table 131 – Vendor-specific registers .....	804
Table 132 – Phy event counter registers .....	805
Table 133 – Reserved registers .....	805
Table 134 – PSCR definition .....	806
Table 135 – Port selection signal inter-reset timing requirements .....	816
Table A.1 – CRC and scrambler calculation example – PIO write command .....	836
Table B.1 – Queued commands with Out of Order Completion .....	847
Table B.2 – Interrupt aggregation .....	848
Table D.3 – J1, J2, J3 pin assignments .....	865
Table D.4 – J4, J5, J6 pin assignments .....	866



## 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';
  - ab) ECN033 - Definition of Terms;
  - ac) ECN034 - Corrections to Technical Proposal 005;
  - ad) ECN035 - Clarification of Words 76 to 79;
  - ae) ECN036 - LIF-SATA Clarifications;
  - af) ECN037 - Changes made by Technical Integration Work Group; and
  - ag) 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;
  - q) ECN055 - Consistency of Register FIS nomenclature;
- and the following new features and enhancements:
- a) TPR003 - SSC Profile df/dt Excursion Limitation;

- b) TPR019 - HOLD<sub>P</sub>/HOLDA<sub>P</sub> 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 ALIGN<sub>P</sub> 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-3;
- 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.

## **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-3 standard, augmented with Native Command Queuing 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 Gbps, 3.0 Gbps and 6.0 Gbps).

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. At the time of publication, the editions indicated were valid. 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 (ISO, IEC, CEN/CENELEC, ITUT), may be obtained from the international and regional organizations who control them.

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

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

Serial Attached SCSI – 1.1 (SAS-1.1) [ANSI INCITS 417-2006]

Serial Attached SCSI – 2 (SAS-2) [ANSI INCITS 457-2010]

Serial Attached SCSI – 2.1 (SAS-2.1) [ANSI INCITS 478-2011]

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

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

SCSI Enclosure Services - 2 (SES-2) [ANSI INCITS T10/1559-D]

SCSI Primary Commands – 4 (SPC-4) [ISO/IEC 14776-454]

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<sup>2</sup>C-Bus Specification version 2.1. Available from NXP Semiconductors at <http://www.nxp.com>.

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.com>):

- 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 standards published by the Small Form Factor (SFF) Committee are referenced. These standards are available for download through <http://www.sffcommittee.org>:

- a) SFF-8086, Compact Multilane Series: Common Elements;
- b) SFF-8087, Compact Multilane Series: Unshielded;
- c) SFF-8088, Compact Multilane Series: Shielded;
- d) SFF-8111, 1.8" drive form factor (60x70 mm);
- e) SFF-8144, 54 mm x 71 mm Form Factor w/micro SATA Connector;
- f) SFF-8201, Form Factor of 2.5" Disk Drives;
- g) SFF-8301, Form Factor of 3.5" Disk Drives;
- h) SFF-8470, Shielded High Speed Serial Multilane Copper Connector;
- i) SFF-8484, Multilane Internal Serial Attachment Connector;
- j) SFF-8553, Form Factor of 5 1/4" 7 mm Height Optical Drives with SATA Interface;
- k) SFF-8639, Multifunction 12 Gb/s 6x Unshielded Connector; and
- l) SFF-8680 Serial Attachment 12 Gb/s 2x Unshielded Connector.

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;

## HIGH SPEED SERIALIZED AT ATTACHMENT

### Serial ATA International Organization

- 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.org>:

- a) PCI specification, revision 3.0;
- b) PCI Express specification, revision 2.1; and
- c) PCI Express® Card Electromechanical Specification, Revision 2.0. April 11, 2007. 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) ATA/ATAPI-8 Serial Transport (ATA8-AST) [ANSI INCITS T13/1697-D];
- b) ATA/ATAPI-8 Parallel Transport (ATA8-APT) [ANSI INCITS T13/1698-D];
- c) ATA Host Adapter Standards - 2 (HBA-2) [ANSI INCITS T13/2014D];
- d) ATA/ATAPI Command Set - 3 (ACS-3) [ANSI INCITS T13/2161-D]; and
- e) Serial Attached SCSI-3 (SAS-3) [ANSI INCITS T10/2212-D].

The following PCI Express® Specification under development is available from <http://www.pcisig.org>, PCI Express® M.2 Electromechanical Specification, Revision 0.9, Version 3.0 July 31, 2013.

### 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](http://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus).

### **3.5 SATA-IO Style Guide**

Serial ATA Design Guide #004 version 17 dated July 27, 2012 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 Attachment (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<sub>P</sub> sequence**

A continuous stream of consecutive ALIGN<sub>P</sub> 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. 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 9.2.2).

##### **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<sub>P</sub>, 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<sub>P</sub>.

#### **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<sub>P</sub> 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 ABRT bit set to one in the Error register, and 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) BSY & DRQ bits 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 that Z is equal to K (See 9.2.2).

#### **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<sub>P</sub>.

#### **4.1.1.28 data character**

A data character is a character that Z is equal to D (See 9.2.2).

#### **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 9.2 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<sub>p</sub>, a FIS, a CRC calculated over the contents of the FIS, and EOF<sub>p</sub>.

#### **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<sub>P</sub>, CRC, and EOF<sub>P</sub> 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 Gbps. Details as given in 7.2.

#### **4.1.1.54 Gen1i**

The internal electrical specifications at 1.5 Gbps 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 meters, defined at 1.5 Gbps.

#### **4.1.1.56 Gen1u**

The electrical specifications defined at 1.5 Gbps 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 Gbps.

#### **4.1.1.58 Gen2**

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

#### **4.1.1.59 Gen2i**

The internal electrical specifications at 3.0 Gbps 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 Gbps.

#### **4.1.1.61 Gen2u**

The electrical specifications defined at 3.0 Gbps 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 Gbps.

#### **4.1.1.63 Gen3**

Refers to third generation signaling characterized by a speed of 6.0 Gbps.

#### **4.1.1.64 Gen3i**

The internal electrical specifications at 6.0 Gbps with cable lengths up to 1 m.

#### **4.1.1.65 Gen3u**

The electrical specifications defined at 6.0 Gbps 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 pins**

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 native command queuing (NCQ) command is a NCQ command that shall 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<sub>P</sub> 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. 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 unit under test. 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 9.4.7 for using a Linear Feedback Shift Register (LFSR) in scrambling.

#### **4.1.1.85 Logical Block Address (LBA)**

As defined in the ACS-3 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 Non-Streaming command**

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

#### **4.1.1.91 NCQ Streaming command**

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

#### **4.1.1.92 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.

#### **4.1.1.93 OS-aware hot removal**

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

#### **4.1.1.94 Out-of-Band (OOB) signaling**

Out-of-Band (OOB) signaling is a pattern of ALIGN<sub>P</sub> 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).

#### **4.1.1.95 PCI Express® (PCIe)**

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

#### **4.1.1.96 PCI Express® Generation 2 (PCIe Gen2)**

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

#### **4.1.1.97 PCI Express® Generation 3 (PCIe Gen3)**

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

#### **4.1.1.98 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.

#### **4.1.1.99 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.

#### **4.1.1.100 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.

#### **4.1.1.101 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.

#### **4.1.1.102 primary medium**

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

#### **4.1.1.103 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.

#### **4.1.1.104 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.

#### **4.1.1.105 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.

#### **4.1.1.106 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.

#### **4.1.1.107 random jitter (RJ)**

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

#### **4.1.1.108 reset speed negotiation (RSN)**

The use of COMRESET to accomplish speed negotiation.

#### **4.1.1.109 sector**

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

#### **4.1.1.110 Serial ATA Enclosure Management Bridge (SEMB)**

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

#### **4.1.1.111 Shadow Register Block registers**

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

#### **4.1.1.112 Self-Monitoring, Analysis, and Reporting Technology (SMART)**

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

#### **4.1.1.113 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.

#### **4.1.1.114 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 EIA (Electronic Industries Association) or an ASC (Accredited Standards Committee).

#### **4.1.1.115 software reset (SRST)**

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

#### **4.1.1.116 Solid State Drive (SSD)**

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

#### **4.1.1.117 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.

#### **4.1.1.118 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<sup>2</sup>C interface to the Serial ATA Enclosure Management Bridge.

#### **4.1.1.119 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.

#### **4.1.1.120 surprise hot removal**

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



#### **4.1.1.121 sync**

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

#### **4.1.1.122 SYNC Escape**

The condition when SYNC<sub>P</sub> 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.

#### **4.1.1.123 time domain reflectometer (TDR)**

An instrument used to test the impedance of the unit under test.

#### **4.1.1.124 timing interval analyzer (TIA)**

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

#### **4.1.1.125 total jitter (TJ)**

Unless further specified by including the BER level, total jitter (TJ) has a peak to peak value of  $(14 \times RJ) + DJ$ . For Gen3i Tx jitter,  $TJ(10^{-12})$  and  $TJ(10^{-8})$  are specified and measured directly. RJ and DJ are not directly specified.

#### **4.1.1.126 unit interval (UI)**

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

#### **4.1.1.127 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 unit under test (UUT).

#### **4.1.1.128 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.

#### **4.1.1.129 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 to one in the Status register at command completion.

#### **4.1.1.130 vector network analyzer (VNA)**

An instrument used to test the impedance of the unit under test.

#### 4.1.1.131 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.

#### 4.1.1.132 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.

#### 4.1.1.133 xSATA

The System to System Interconnects - Data Center Applications usage model (see 5.3.6).

#### 4.1.1.134 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  $\pm 5\%$  of the waveform voltage. The “zero crossing” is the location of the mean of the waveforms.

### 4.1.2 Symbols and abbreviations

<b>Abbreviation</b>	<b>Meaning</b>
AC	Alternating Current
AFE	Analog Front End
AnyDword	Any Dword (see 9.6.1)
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	bit
B	byte (see 4.1.1.8)
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
dB	Decibel
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

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

<b>Abbreviation</b>	<b>Meaning</b>
DP	Device Present
DSS	Disable Staggered Spinup
DW	Dword (see 4.1.1.39)
Dword	Double Word (see 4.2.8)
EIA	Electronic Industries Alliance
EMI	Electromagnetic Interference
eSATA	eSATA usage model (see 5.3.7)
ESD	Electrostatic Discharge
Fbaud	Frequency baud (see 4.1.1.52)
FER	Frame Error Rate (see 4.1.1.50)
FCOMP	Frame Composite Pattern (see 7.4.5.4.8)
FIFO	First In First Out
FIS	Frame Information Structure (see 4.1.1.51)
FPC	Flexible Printed Circuit (see 5.3.11.6)
FS	Feature Specific
Gbps	Gigabits per second (i.e., 10 <sup>9</sup> bits per second)
GPL	General Purpose Logging (see ACS-3)
GND	Ground
GT/s	Giga Transfers per second (i.e., 10 <sup>9</sup> transfers per second)
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)
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 4.1.1.94)
PCB	Printed Circuit Board
PCIe	Peripheral Component Interconnect Express (see 4.1.1.95)
Phy	Physical
PIO	Programmed Input/Output (see 4.1.1.104)
PLL	Phase Lock Loop
PRD	Physical Region Descriptor (see 4.1.1.100)
Qword	Quad Word (see 4.2.8)

<b>Abbreviation</b>	<b>Meaning</b>
RA	Right Angle
RAID	Redundant Array of Independent Disks
RD+	Running Disparity positive (see 9.2.3.3)
RD-	Running Disparity negative (see 9.2.3.3)
RJ	Random Jitter (see 4.1.1.107)
RSN	Reset Speed Negotiation (see 4.1.1.108)
Rx	Receiver
SAS	Serial Attached SCSI
SATA	Serial ATA
SATA Express	SATA Express usage model (see 6.10)
SCSI	Small Computer System Interface
SDB	Set Device Bits (see 10.5.7)
SEMB	Serial ATA Enclosure Management Bridge (see 4.1.1.110)
SEP	Storage Enclosure Processor (see 4.1.1.118)
SES	SCSI Enclosure Services
SFF	Small Form Factor (see 4.1.1.114)
SMA	SubMiniature version A
SMART	Self-Monitoring, Analysis, and Reporting Technology (see 4.1.1.112)
SMT	Surface Mount Technology
SRIS	Separate RefClk Independent SSC
SRST	Software reset (see 4.1.1.115)
SSC	Spread Spectrum Clocking (see 4.1.1.117)
SSD	Solid State Drive (see 4.1.1.116)
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)
TDR	Time Domain Reflectometry (see 4.1.1.123)
TIA	Time Interval Analyzer (see 4.1.1.124)
TJ	Total Jitter (see 4.1.1.125)
Tx	Transmitter
UHost	Universal Host (see 4.1.1.128)
UI	Unit Interval (see 4.1.1.126)
us	microsecond (i.e., 10 <sup>-6</sup> seconds)
USM	Universal Storage Module (see 5.3.12)
UUT	Unit Under Test (see 4.1.1.127)
VNA	Vector Network Analyzer (see 4.1.1.130)
XOR	Exclusive Logical OR

<b>Symbol</b>	<b>Meaning</b>
^	exclusive logical OR
x	multiplication
/	division
≠	not equal
≤	less than or equal to
±	plus or minus
≈	approximately
+	add
-	subtract
<	less than
=	equal
>	greater than
≥	greater than or equal to

## **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, fields, and acronyms used as signal names are in all uppercase (e.g., IDENTIFY DEVICE). Fields containing only one bit are usually referred to as the "name" bit instead of the "name" field.

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

Primitive names are followed by a "P" subscript (e.g., R\_OK<sub>P</sub>).

### **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 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.

#### **4.2.3.6 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".

#### **4.2.3.7 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.

#### **4.2.3.8 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.

#### **4.2.3.9 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.

#### **4.2.3.10 shall**

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

#### **4.2.3.11 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.

<b>State Designator: State name</b>	Action list <sub>[P   W]</sub>		
1) Transition condition <sup>a</sup>	→	<b>Next state 1</b>	
2) Transition condition <sup>b</sup>	→	<b>Next state 2</b>	
<sup>a</sup> footnote			
<sup>b</sup> 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).

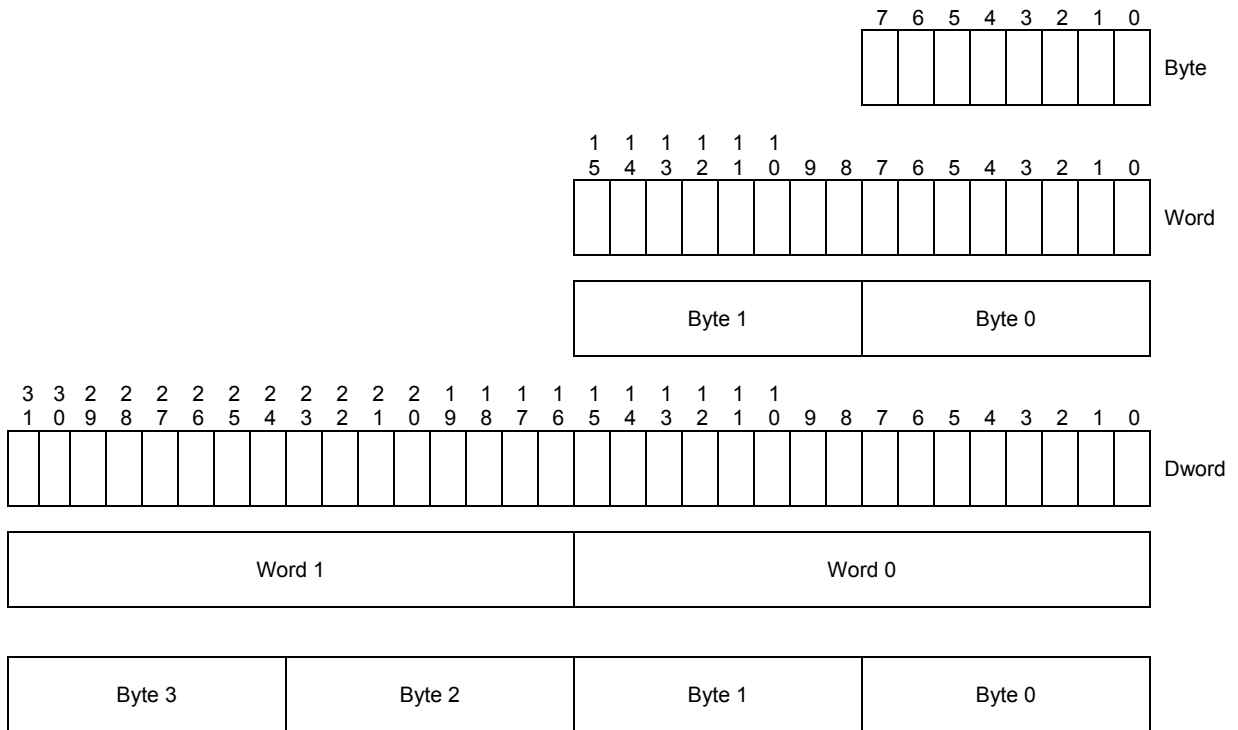
A Dword may be represented as an ordered set of four (4) bytes. The least significant byte is byte 0 and the most significant byte is byte 3. The most significant byte is shown on the left (see Figure 2 part 1).

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).

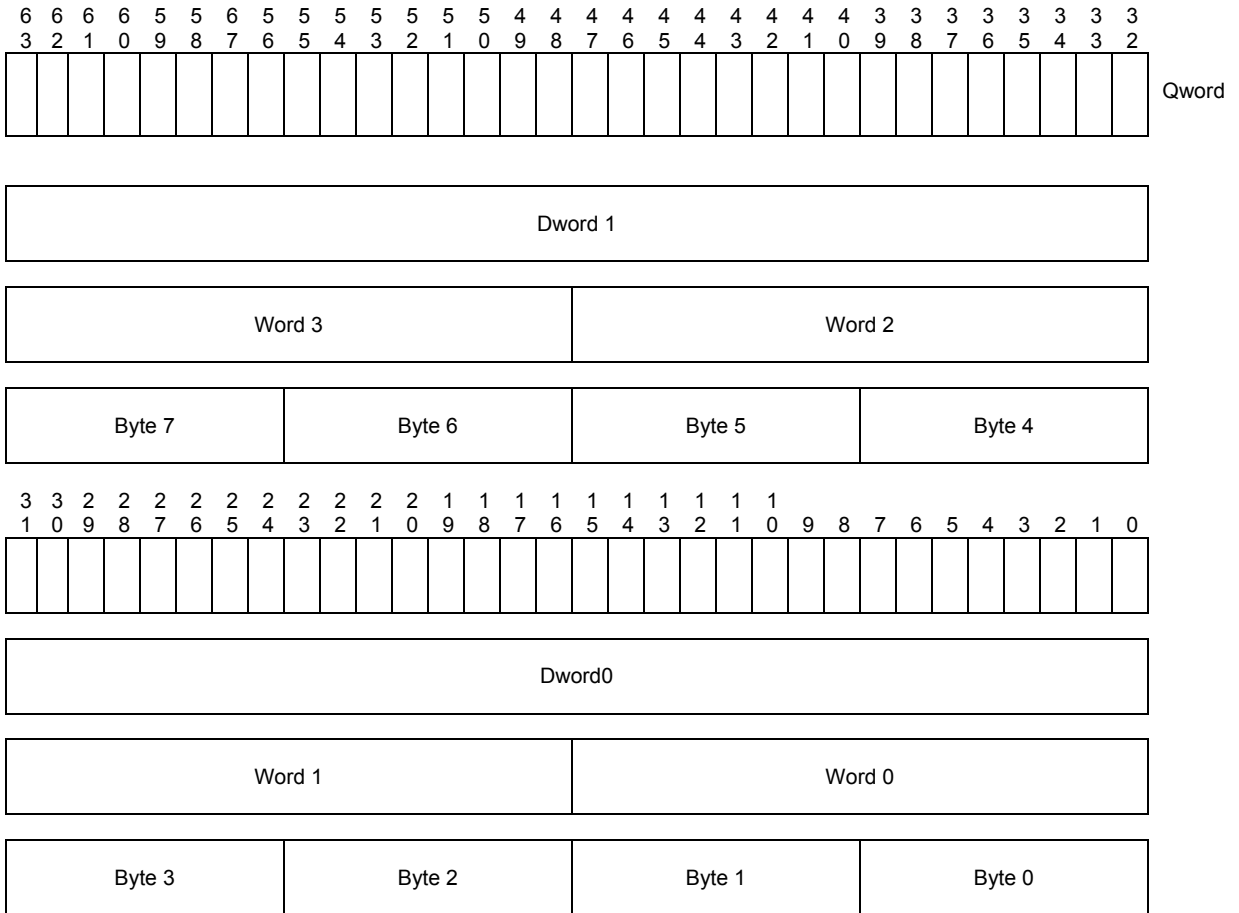
A Qword may be represented as an ordered set of eight (8) bytes. The least significant byte is byte 0 and the most significant byte is byte 7. The most significant byte is shown on the left (see Figure 2 part 2).





**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 replacement for the parallel ATA attachment of mass storage devices. The serial link employed is a high-speed differential layer that utilizes Gigabit technology and 8b/10b encoding.

Figure 3 illustrates how two devices are connected to a parallel ATA host adapter. This method allows up to two devices to be connected to a parallel ATA bus using a Master/Slave communication technique. Each device is connected via a ribbon cable that “daisy chains” the devices.

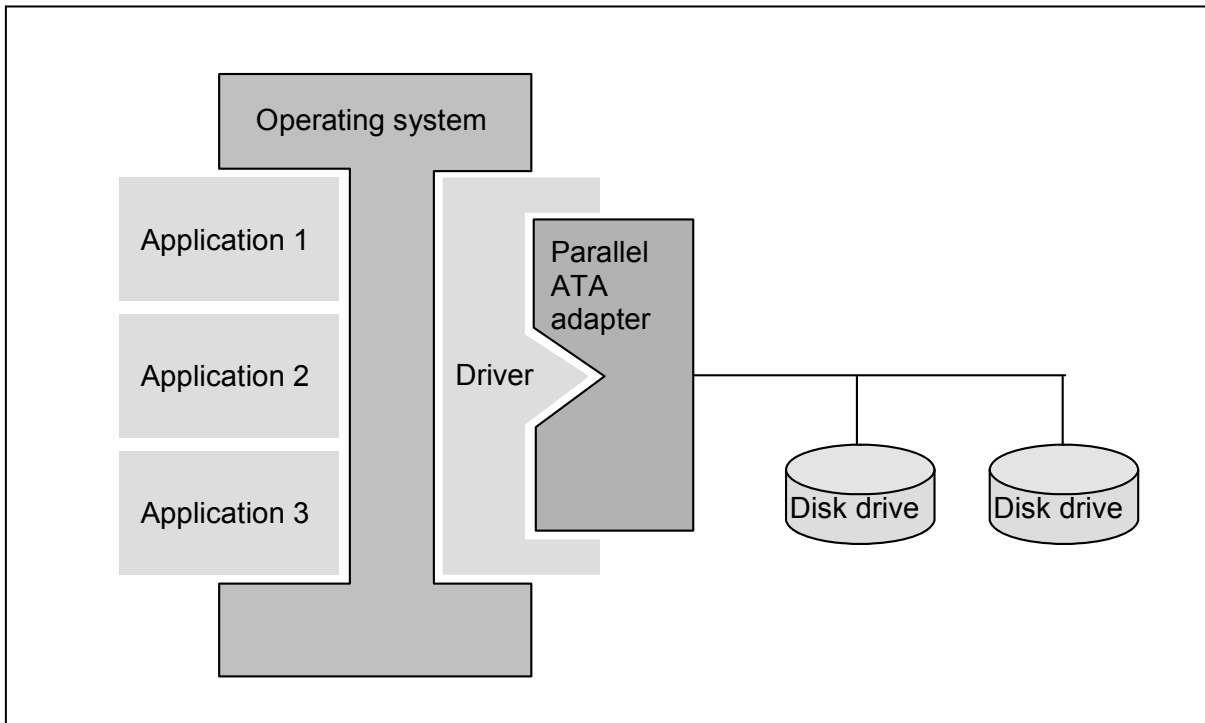
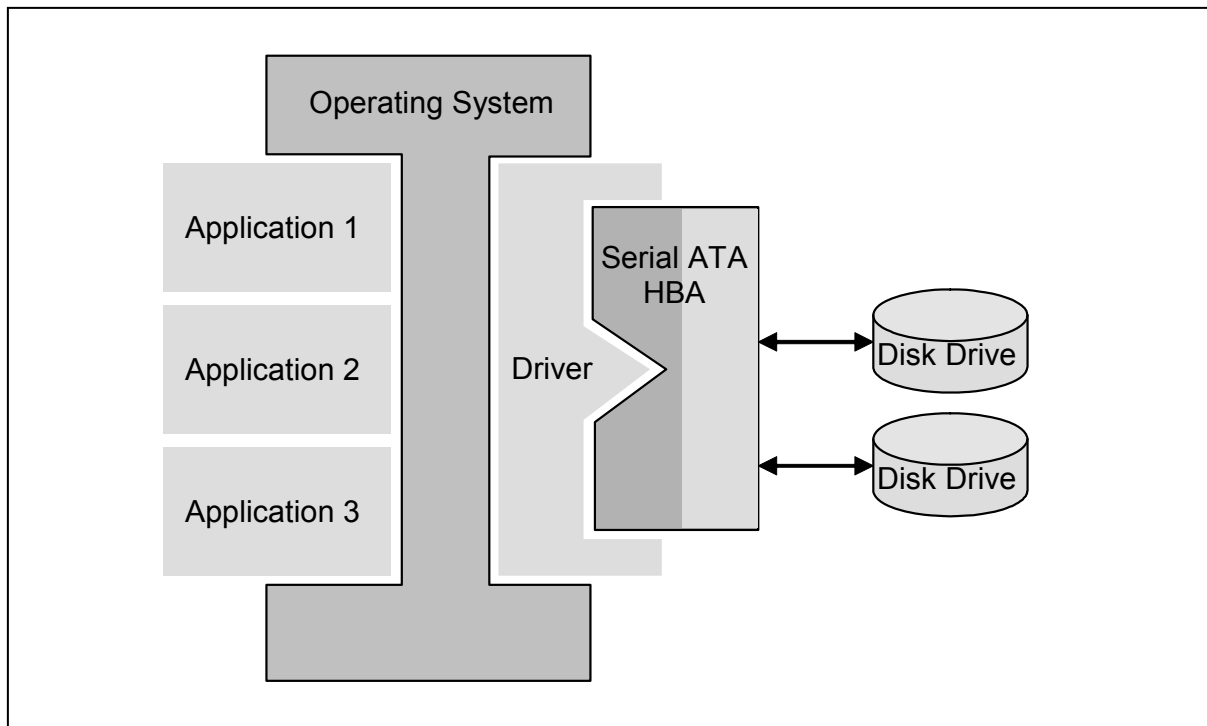


Figure 3 – Parallel ATA device connectivity

Figure 4 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 Figure 3. Host software that is only parallel ATA aware 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 “masters” 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.



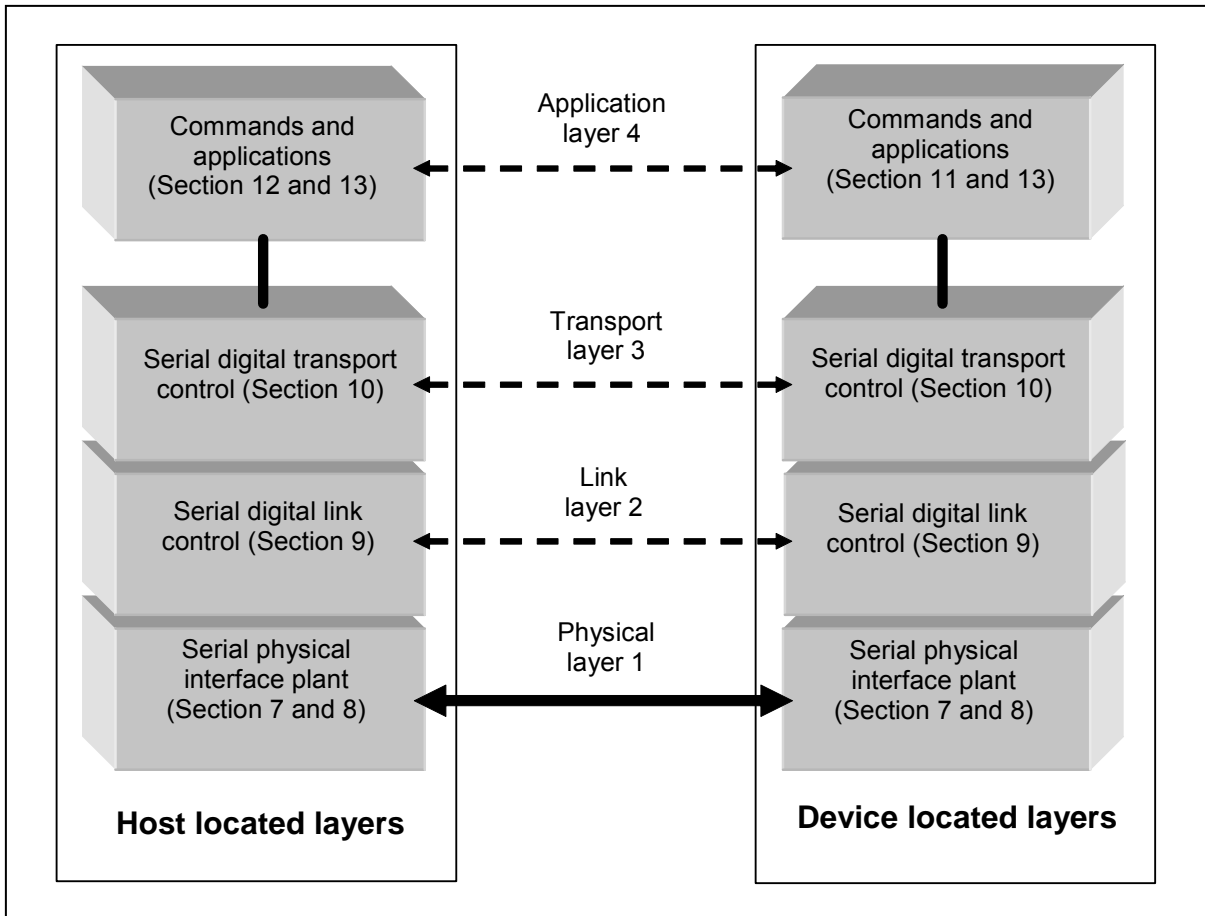
**Figure 4 – Serial ATA connectivity**

## 5.2 Architecture

There are four layers in the Serial ATA architecture:

- a) 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 Physical layer is responsible for transmitting and receiving the encoded information as a serial data stream on the wire.



**Figure 5 – 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., Native Command Queuing), 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
<b>Internal 1 m Cabled Host to Device</b>	5.3.2	Int SL	<= 1 m	Table 36	-6 dB	6.2.5	6.2.3.1	NS
<b>Short Backplane to Device</b>	5.3.3	BP	-	P	P	P	6.2.3.1	R
<b>Internal 4-lane Cabled Disk Arrays</b>	5.3.5	Int ML	<= 1 m	Table 36	-6 dB	6.2.11 or 6.2.12	6.2.3.1	NS
<b>System to System Interconnects – Data Center Applications xSATA</b>	5.3.6	Ext ML	<= 2 m	Table 38	-8 dB	6.11.4 (key 7)	6.11.4 (key 7)	R
<b>System to System Interconnects – External Desktop Applications eSATA</b>	5.3.7	Ext SL	<= 2 m	Table 37	-8 dB	6.11	6.11	R
<p>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</p>								
<p>NOTE - Many of the references in the table are section numbers or notations of clarification that do not require Key values.</p>								

**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
<b>Proprietary Serial ATA Disk Arrays</b>	5.3.8	BP and cable	-	Table 36	-6 dB	6.2.5 or P	6.2.3.1	R
<b>Serial ATA and SAS</b>	5.3.9	BP	-	P	P	SAS	6.2.3.1	R
<b>LIF-SATA</b>	5.3.11.6	P	P	Table 36	-6 dB	6.5.4	6.5.3	NS
<b>mSATA</b>	5.3.11.5	BP	-	P	P	6.6.3	6.6.4.1	NS
<b>SATA USM</b>	5.3.12	P	P	P	P	6.7.2	6.2.3.1	R
<b>SATA MicroSSD</b>	5.3.11.7	P	P	P	P	NS	NS	NS
<b>Embedded M.2</b>	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 Gbps	Gen1m 1.5 Gbps	Gen1u 1.5 Gbps	Gen2i 3.0 Gbps	Gen2m 3.0 Gbps	Gen2u 3.0 Gbps	Gen3i 6.0 Gbps	Gen3u 6.0 Gbps
<b>Internal</b> 1 m cabled host to device	R	NS	NS	FS <sup>a</sup>	NS	NS	FS <sup>a</sup>	NS
<b>Short backplane to device</b>	D (host to provide received signal)	H	NS	D (host to provide received signal)	H	NS	NS	NS
<b>Internal 4-lane cabled disk arrays</b>	R	NS	NS	FS <sup>a</sup>	NS	NS	FS <sup>a</sup>	NS
<b>System to system inter connects – data center applications</b> xSATA	NS	R (key 7)	NS	NS	R (key 7)	NS	NS	NS
<b>System to system inter connects – external desktop applications</b> eSATA	NS	R	NS	NS	FS <sup>a</sup>	NS	NS	NS
Proprietary Serial ATA disk arrays	R	NS	NS	FS <sup>a</sup>	NS	NS	FS <sup>a</sup>	NS
<b>Serial ATA and SAS</b>	D	NS	NS	D	NS	NS	D	NS
<b>LIF-SATA</b>	R	NS	NS	FS <sup>a</sup>	NS	NS	FS <sup>a</sup>	NS
<b>mSATA</b>	R	NS	NS	FS <sup>a</sup>	NS	NS	NS	NS
<b>SATA USM</b>	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.								
<sup>a</sup> 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)**

<b>Characteristic</b>	<b>Gen1i 1.5 Gbps</b>	<b>Gen1m 1.5 Gbps</b>	<b>Gen1u 1.5 Gbps</b>	<b>Gen2i 3.0 Gbps</b>	<b>Gen2m 3.0 Gbps</b>	<b>Gen2u 3.0 Gbps</b>	<b>Gen3i 6.0 Gbps</b>	<b>Gen3u 6.0 Gbps</b>
<b>SATA MicroSSD</b>	D	NS	H	D	NS	H	D	H
<b>Embedded M.2</b>	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.								
<sup>a</sup> Feature specific is intended to indicate that Gen1 is required but higher data rates are optional.								

**Table 3 – SATA Express usage models**

Characteristic	Usage model section number	Host side connector	Device side connector	Cable length	Cable and connector electrical	PCIe lanes	SATA Lanes
<b>SATA Express Internal 1m Cabled Host to Device</b>	6.10	6.10.10	6.10.7	<= 1 m	6.10.15	x2	SL (single lane) muxed with PCIe
<b>SATA Express Direct Connection to Device</b>	6.10	6.10.9	6.10.7	BP	6.10.15	x2	SL (single lane) muxed with PCIe
<b>SATA Express Short Backplane to Device</b>	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 4.1.1.98)							
NOTE - Many of the references in the table are section numbers or notations of clarification that do not require Key values.							
<sup>a</sup> 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. <sup>b</sup> Optional RefClk pins are provided in the SATA Express host receptacle connector, that allows use of the PCIe PHY that requires common RefClk. <sup>c</sup> System OEM is responsible for the management of PCIe devices with or without common RefClk.							

**Table 4 – SATA Express usage model electrical requirements**

<b>Characteristic</b>	<b>PCIe Gen2<sup>a b c</sup></b>	<b>PCIe Gen3<sup>a b c</sup></b>	<b>SATA Gen1i and SATA Gen2i</b>	<b>SATA Gen3i</b>
<b>SATA Express Internal 1m Cabled Host to Device</b>	R	FS <sub>1</sub>	R	FS <sub>1</sub>
<b>SATA Express Direct Connection to Device</b>	R	FS <sub>1</sub>	R	FS <sub>1</sub>
<b>SATA Express Short Backplane to Device</b>	R	FS <sub>1</sub>	R	FS <sub>1</sub>
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.				
<sup>a</sup> 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. <sup>b</sup> Optional RefClk pins are provided in the SATA Express host receptacle connector, that allows use of the PCIe PHY that requires common RefClk. <sup>c</sup> 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, 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 6, operates at 1.5 Gbps, 3.0 Gbps, and 6.0 Gbps. The application may comply with the electrical hot plug specification according to 7.4.5.4.8.

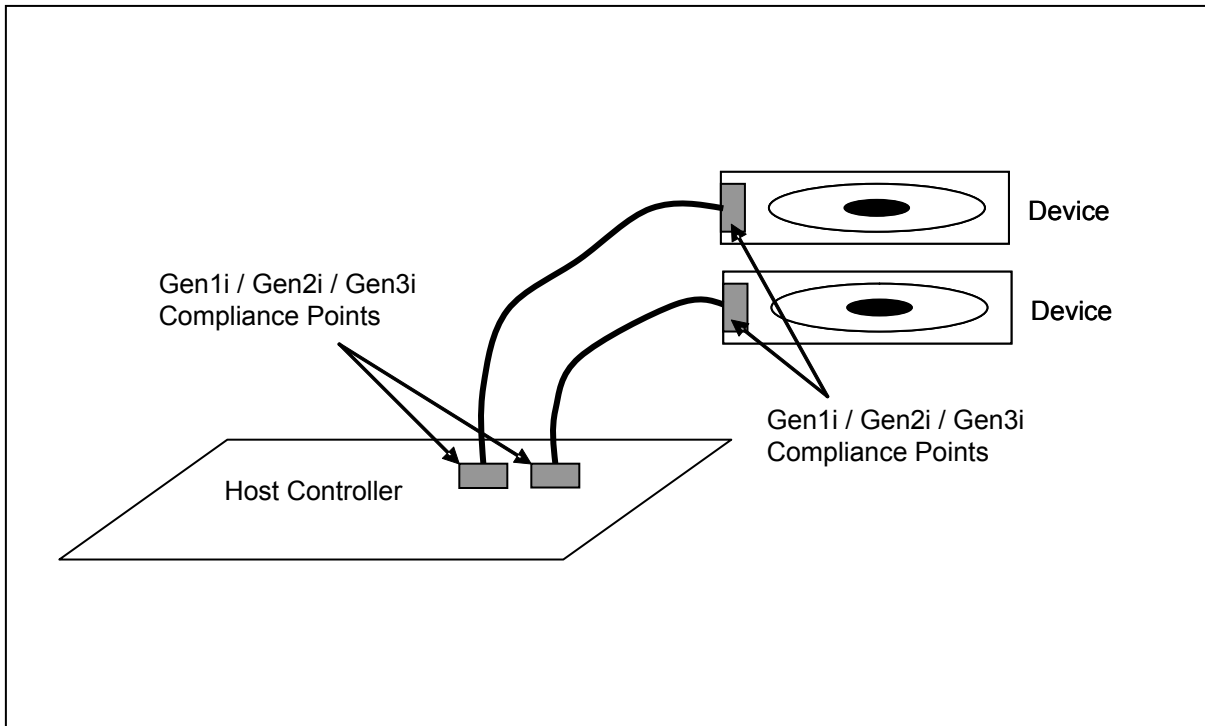


Figure 6 – Internal 1 m cabled host to device application

### 5.3.3 Short backplane to device

In this application, 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 Gbps and 3.0 Gbps where it is anticipated that the backplanes attenuate signals more than the compliant copper cable as defined in clause 6. 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.

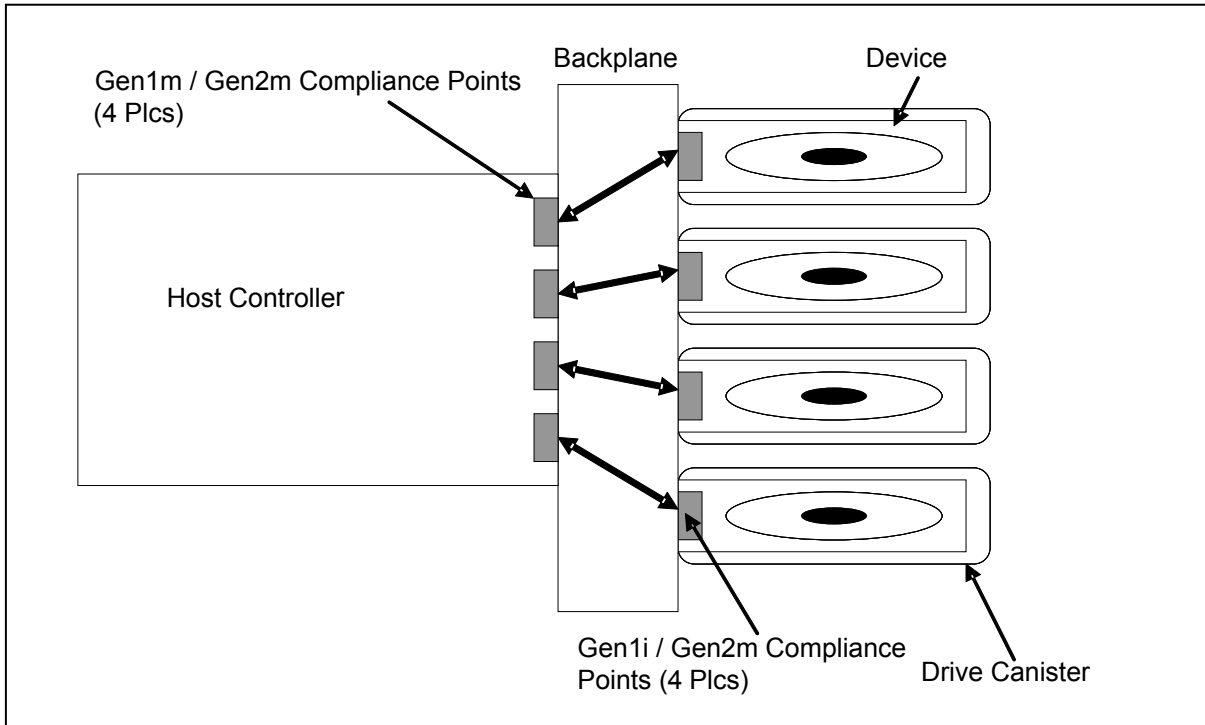


Figure 7 – Short backplane to device application

### 5.3.4 Long backplane to device (obsolete)

Obsolete.

### 5.3.5 Internal 4-lane cabled disk arrays

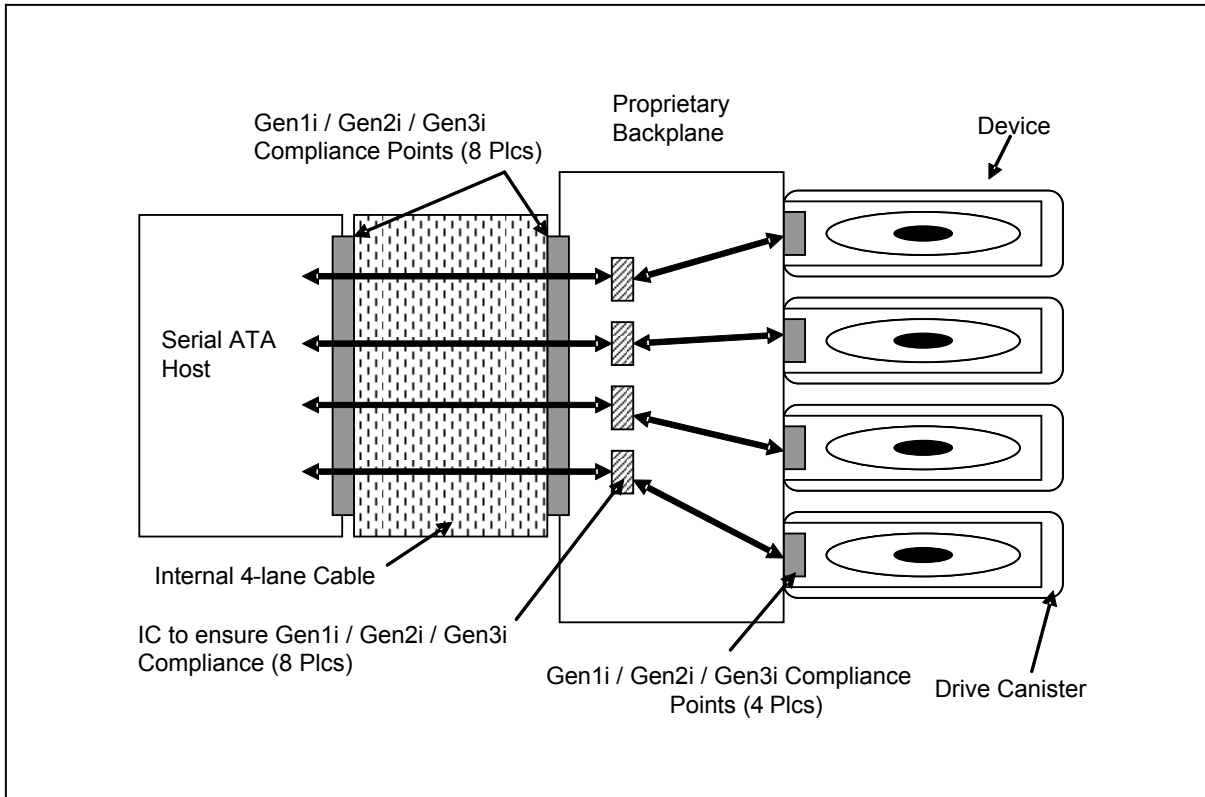
In this application, Gen1i, Gen2i and Gen3i Serial ATA devices are connected to the host controller via the internal 4-lane cable solution. Gen1i, Gen2i and Gen3i specifications shall be met at each end of the 4-lane cable mated interface.

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

- 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
- 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 8 – 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-device 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 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.

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

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

EXAMPLE 1 - If this 1.5 Gbps cable uses Serial ATA specified connectors, the cable shall be 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 9. 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.

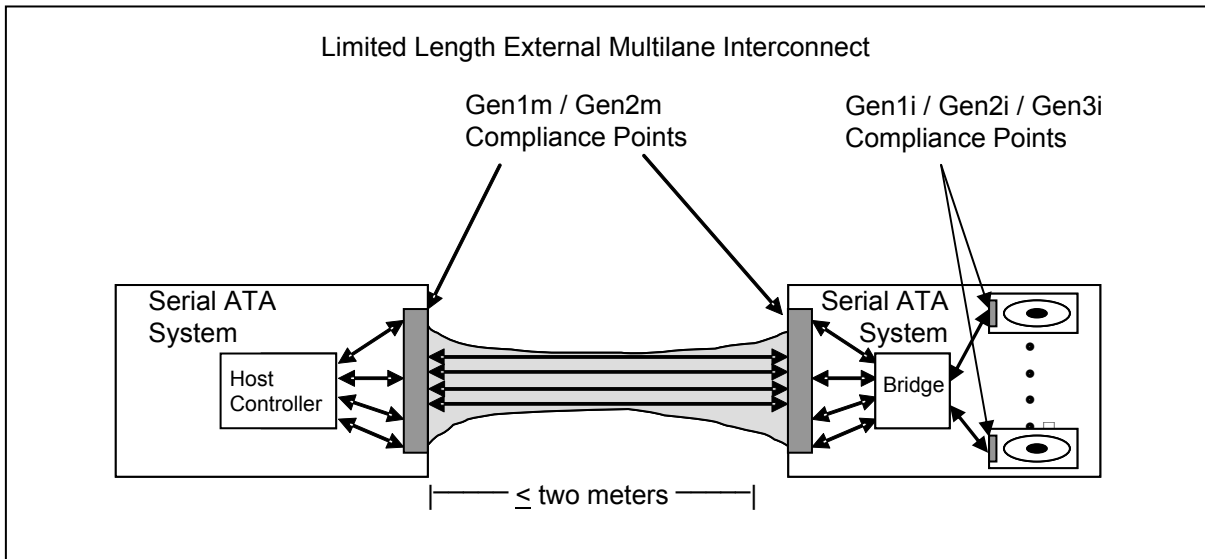


Figure 9 – System-to-system data center interconnects

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

This application 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:

- 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;
- the host system and device enclosure shall comply with Gen1m specifications if operating at Gen1 speeds;
- 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
- 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.

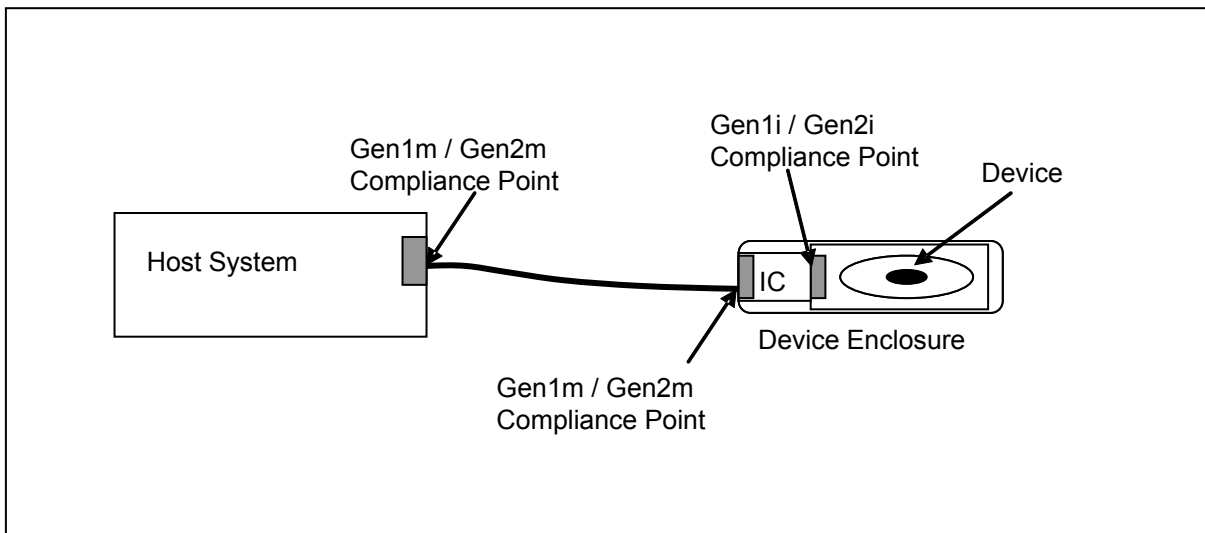


Figure 10 – External desktop application



### 5.3.8 Proprietary Serial ATA disk arrays

In this application, 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.

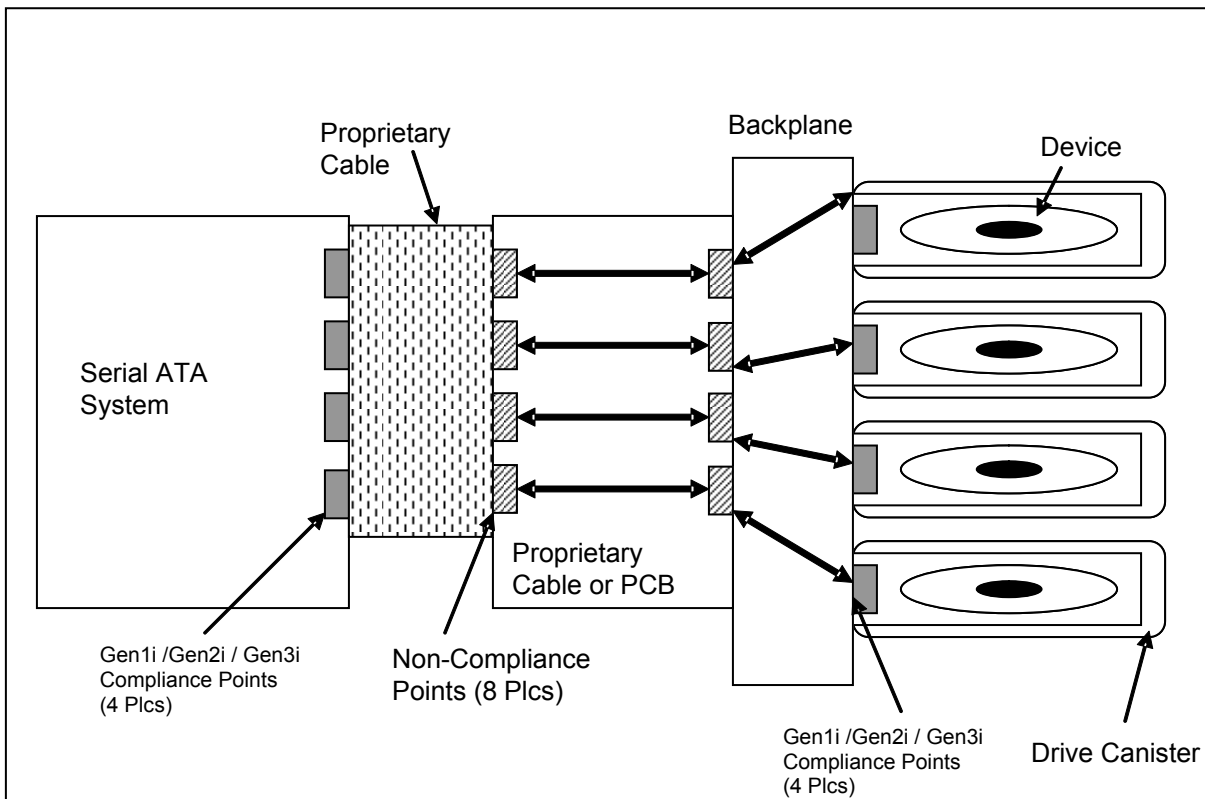


Figure 11 – SATA disk arrays

### 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 has 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 for embedded applications. 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**

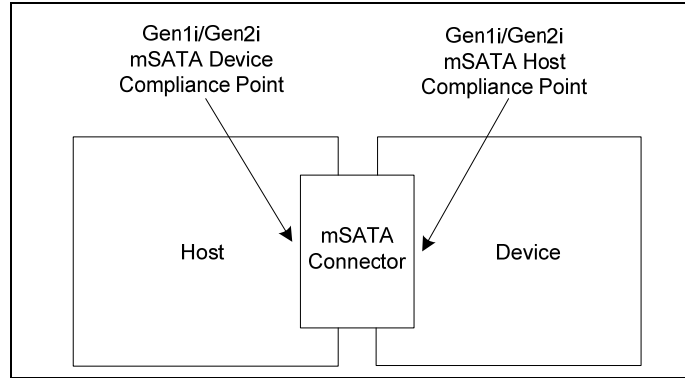
Proprietary docking bay interfaces shall be designed so as to prevent connection with standard SATA components.

#### **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.

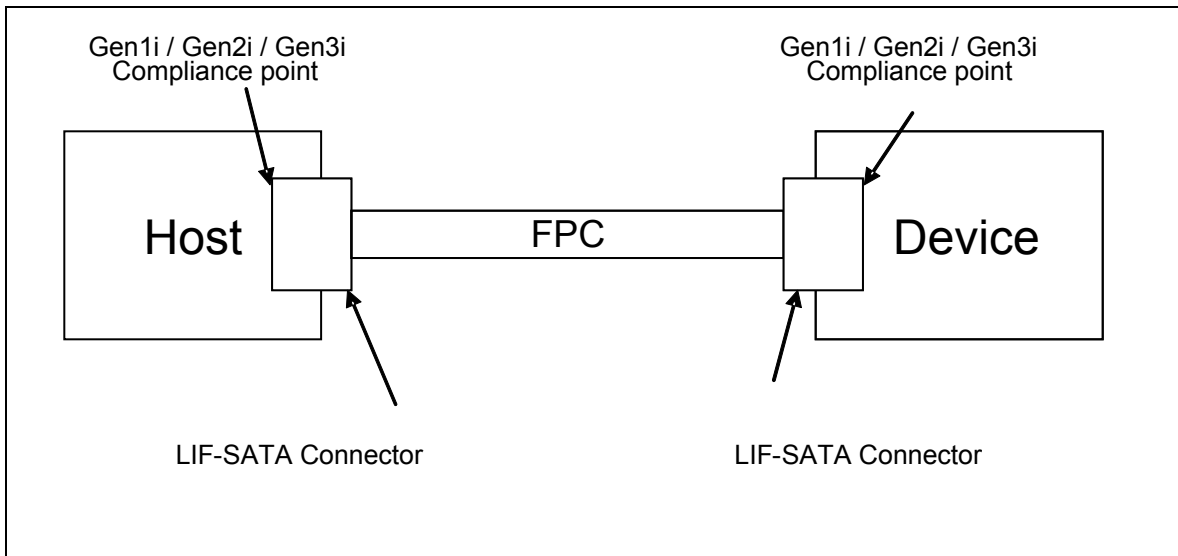
#### **5.3.11.5 mSATA**

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 and is equivalent to the Mobile Applications usage model.



**Figure 12 – mSATA application**

### 5.3.11.6 LIF-SATA



**Figure 13 – Embedded LIF-SATA application**

In this application, 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 13. 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, a Gen1i/2i/3i 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/2i/3i specifications for the device and Gen1u/2u/3u for the host. The signaling at the host controller may exceed the Gen1i/2i/3i transmit maximum providing the Gen1i/2i/3i 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 14.

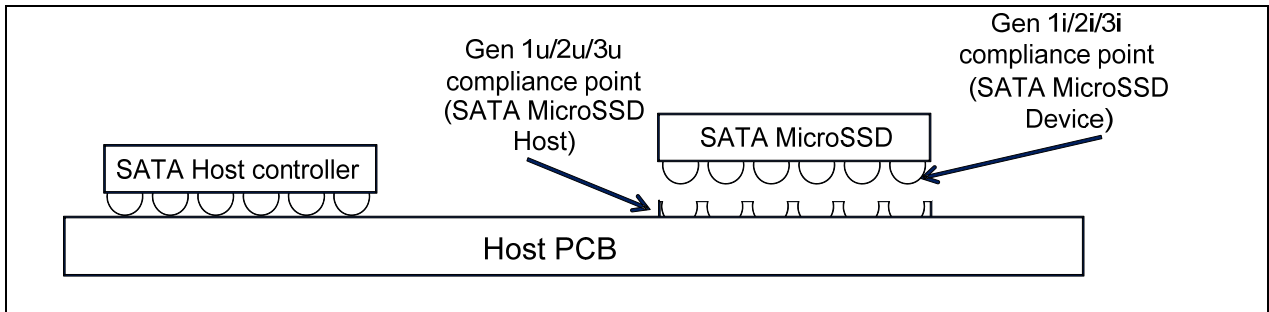


Figure 14 – Embedded SATA MicroSSD application

### 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, Gen 2i, 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 and is equivalent to the Mobile Applications usage model.

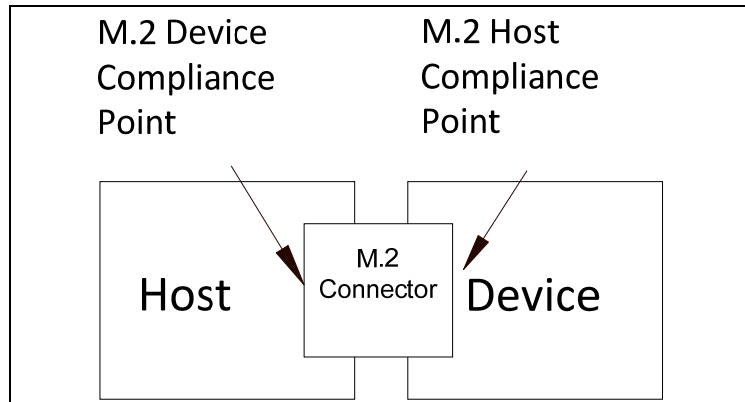


Figure 15 – Embedded M.2 application

### 5.3.12 SATA Universal Storage Module (SATA USM)

In this application, 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 (available at [www.sffcommittee.org](http://www.sffcommittee.org)) 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).

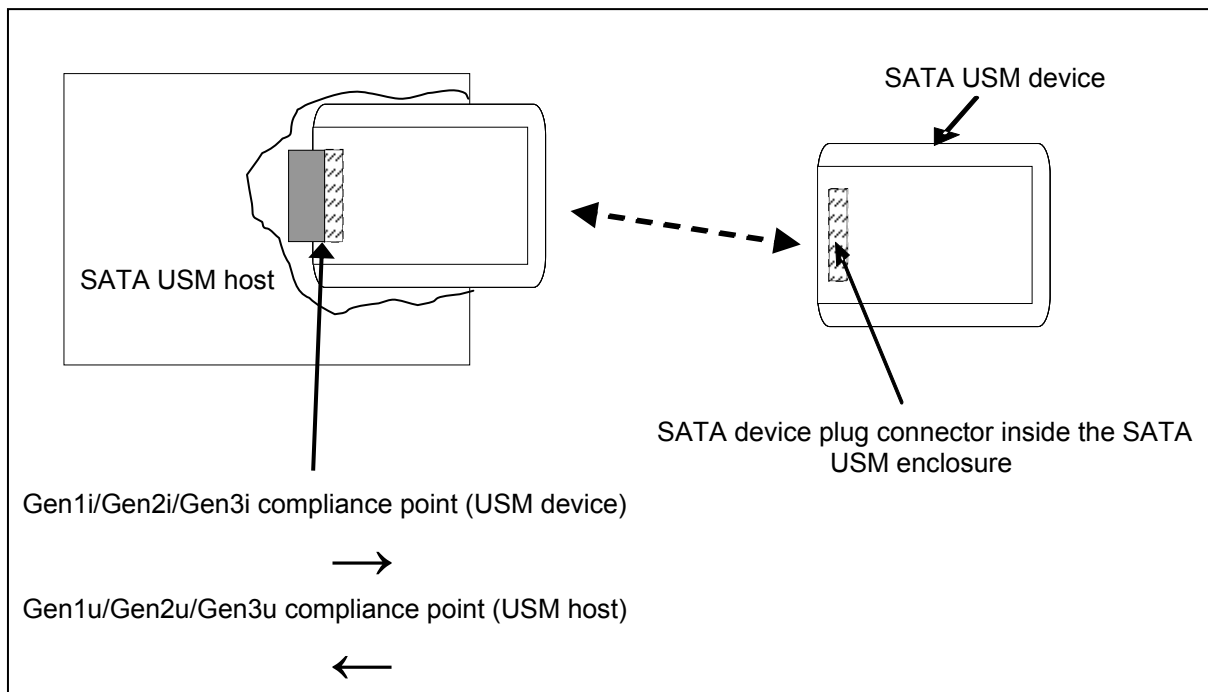
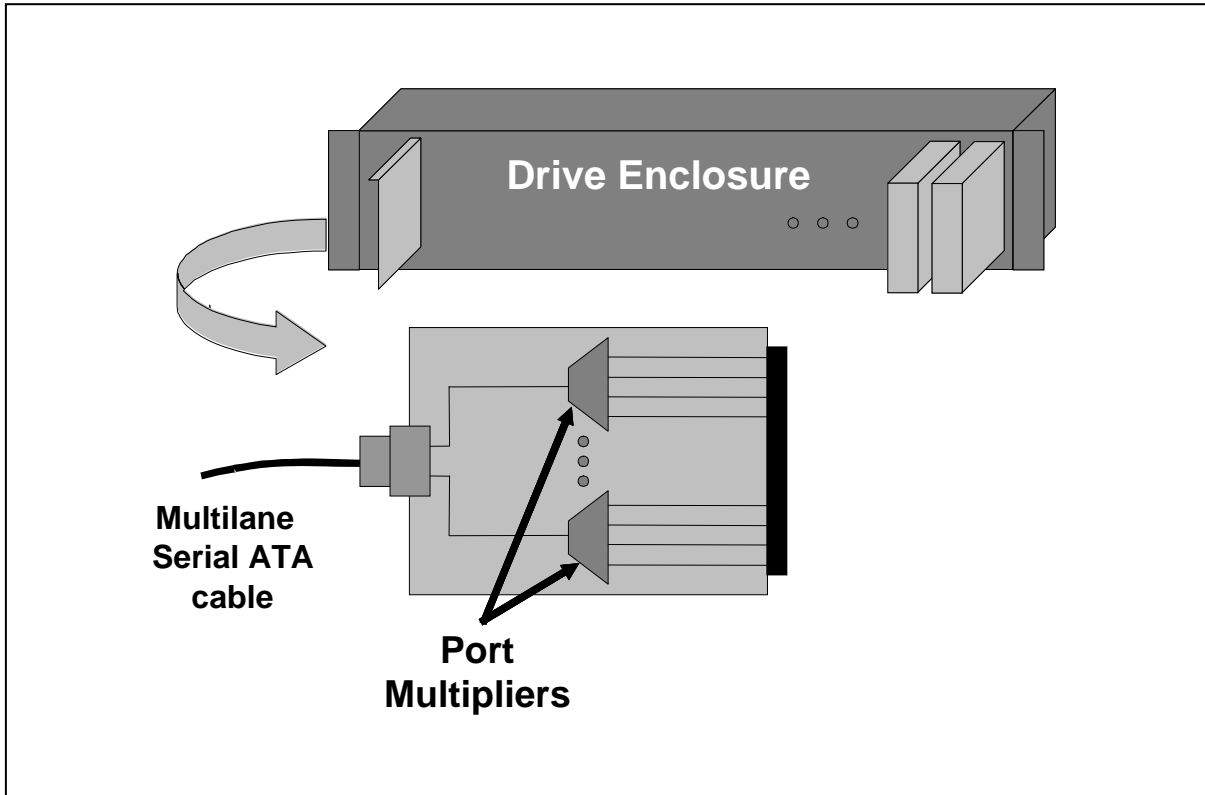


Figure 16 – SATA USM application

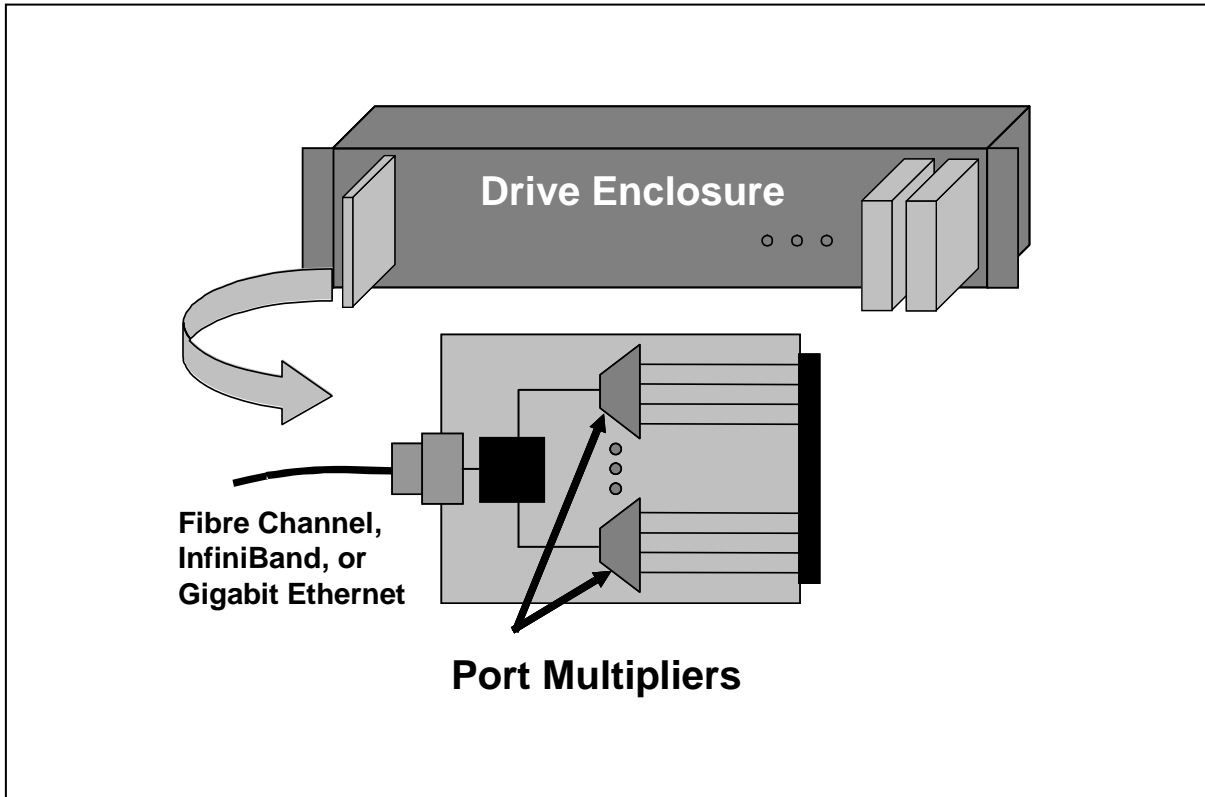
### 5.3.13 Port Multiplier example applications

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 17. 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 17 – Enclosure example using Port Multipliers with Serial ATA as the connection within the rack**

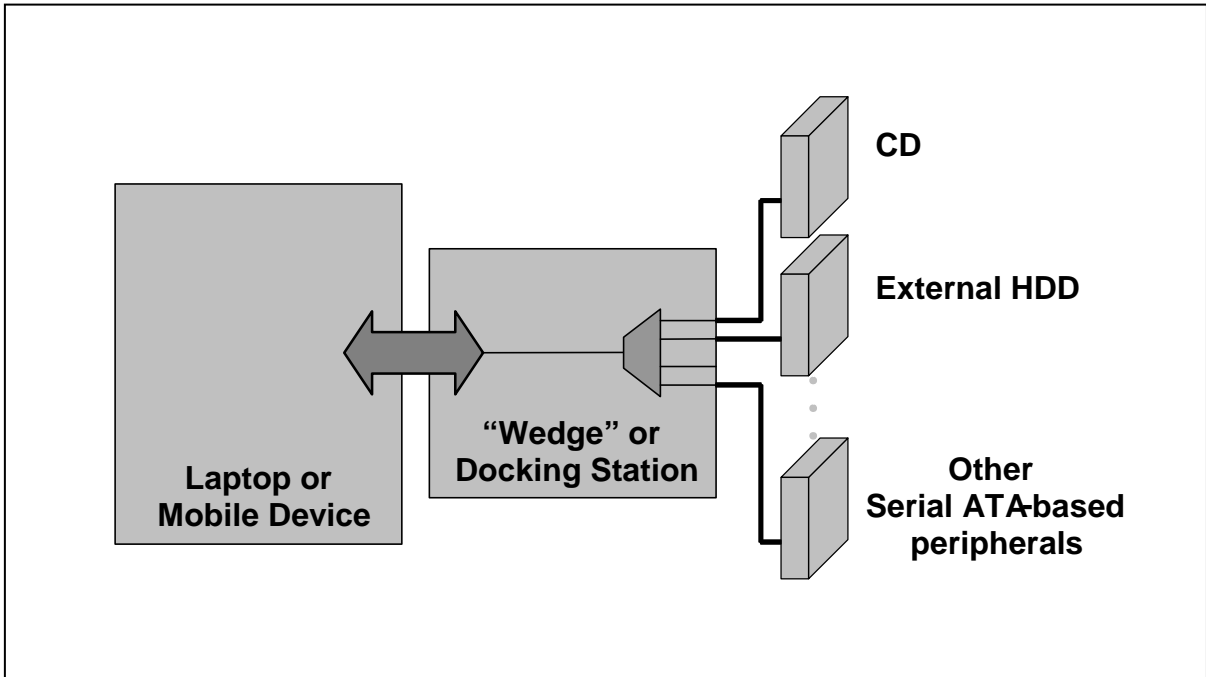
Another example is shown in Figure 18. Fibre Channel, InfiniBand, or Gigabit Ethernet is used as the connection within the rack to the enclosure. Inside the enclosure, a host controller creates two Serial ATA connections from the connection delivered. The enclosure contains eight Serial ATA devices. To create the appropriate number of Serial ATA connections, two 1-to-4 Port Multipliers are used to create eight Serial ATA connections.



**Figure 18 – 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 19 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 19 – 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**

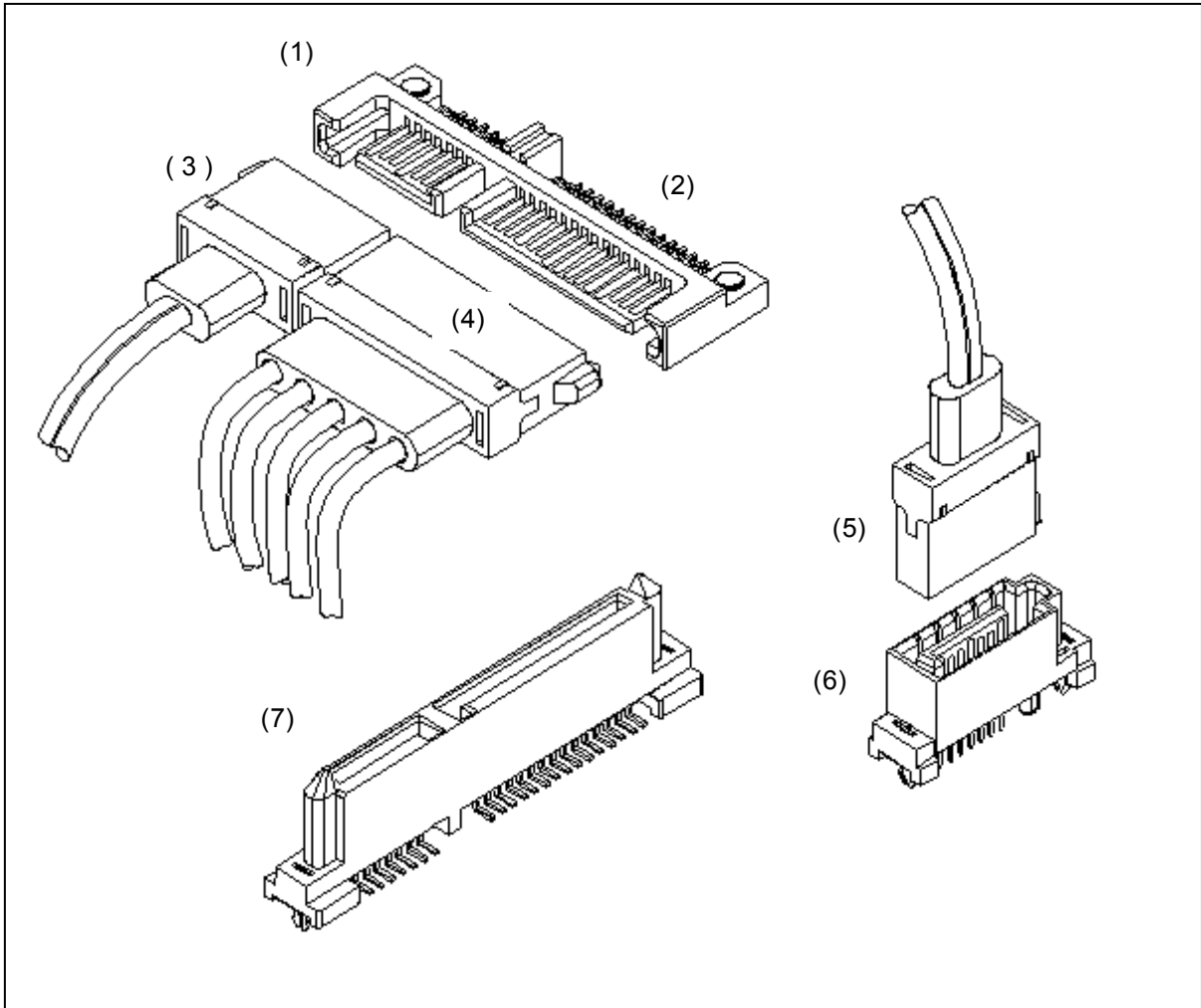
#### **6.2.1 Internal single lane description**

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 20, is inserted directly into a backplane connector, illustrated as (7) in Figure 20. 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 20, mates with the signal cable receptacle connector on one end of the cable, illustrated as (3) in Figure 20. 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 20. 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 20, 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 20. The other end of the power cable is attached to the host as necessary.

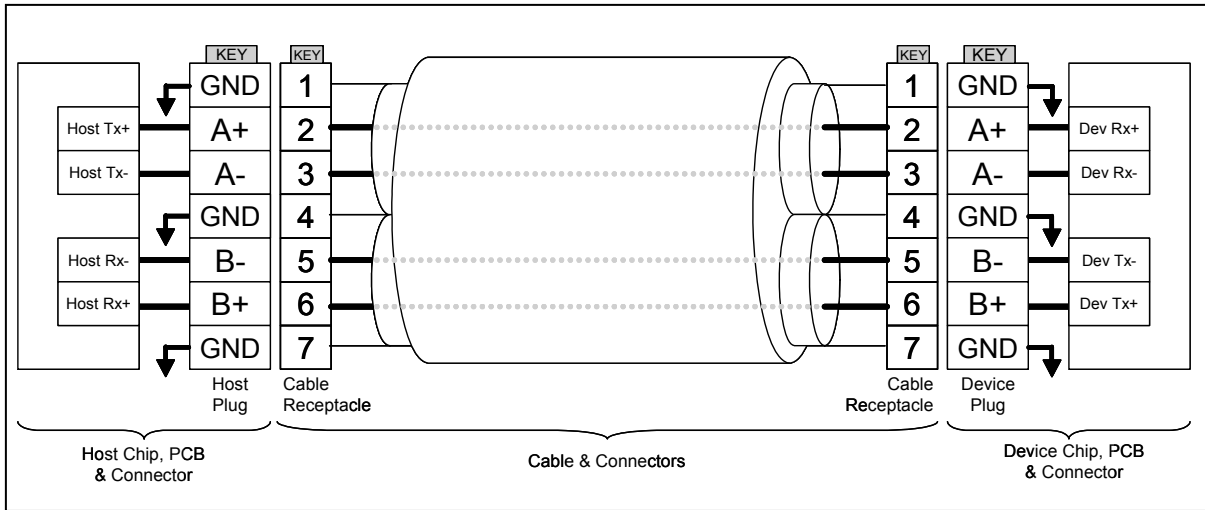


Key:

- 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) & (2)

**Figure 20 – Serial ATA connector examples**

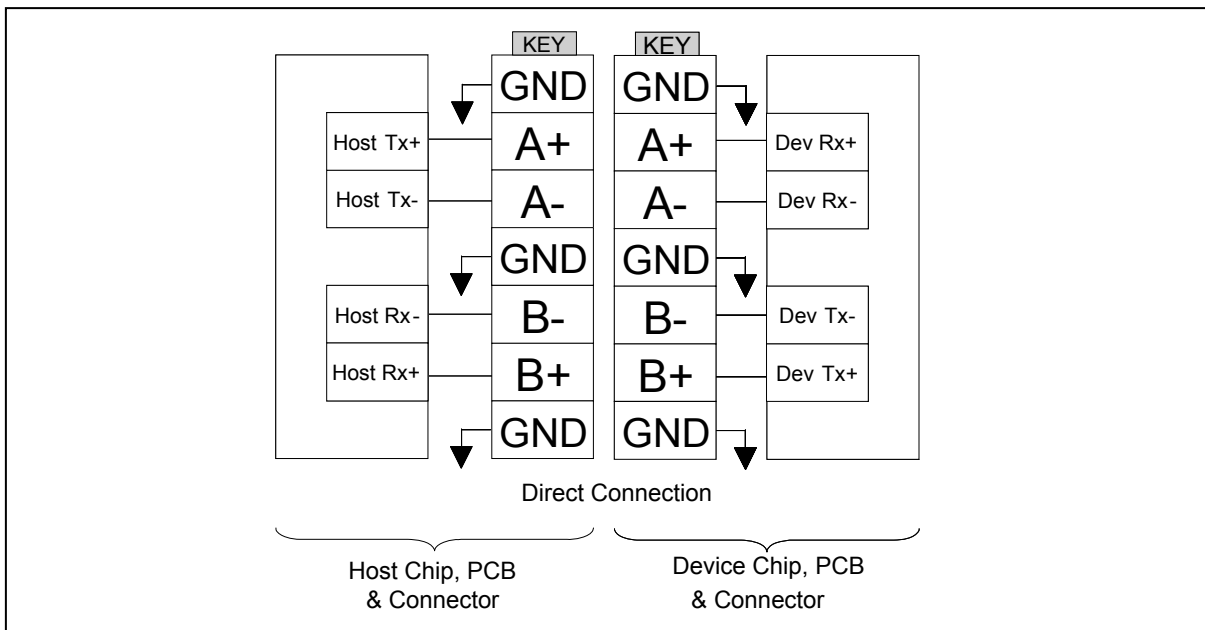
Figure 21 shows a direct cable / connector connection and highlights the signal path of the differential transmitter (Tx) and receiver (Rx) pairs.



**Figure 21 – 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 22 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 22 – 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 23 and Figure 24 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 25 defines the connector location on side mounted 3.5 inch devices. Figure 26 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 27 defines the connector location on side mounted 2.5 inch devices. Figure 28 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 29 defines the Serial ATA connector location on side mounted 1.8 inch devices. Figure 30 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 42 illustrates the fully mated condition of the device to a nominally flush backplane receptacle and Figure 31 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 34.

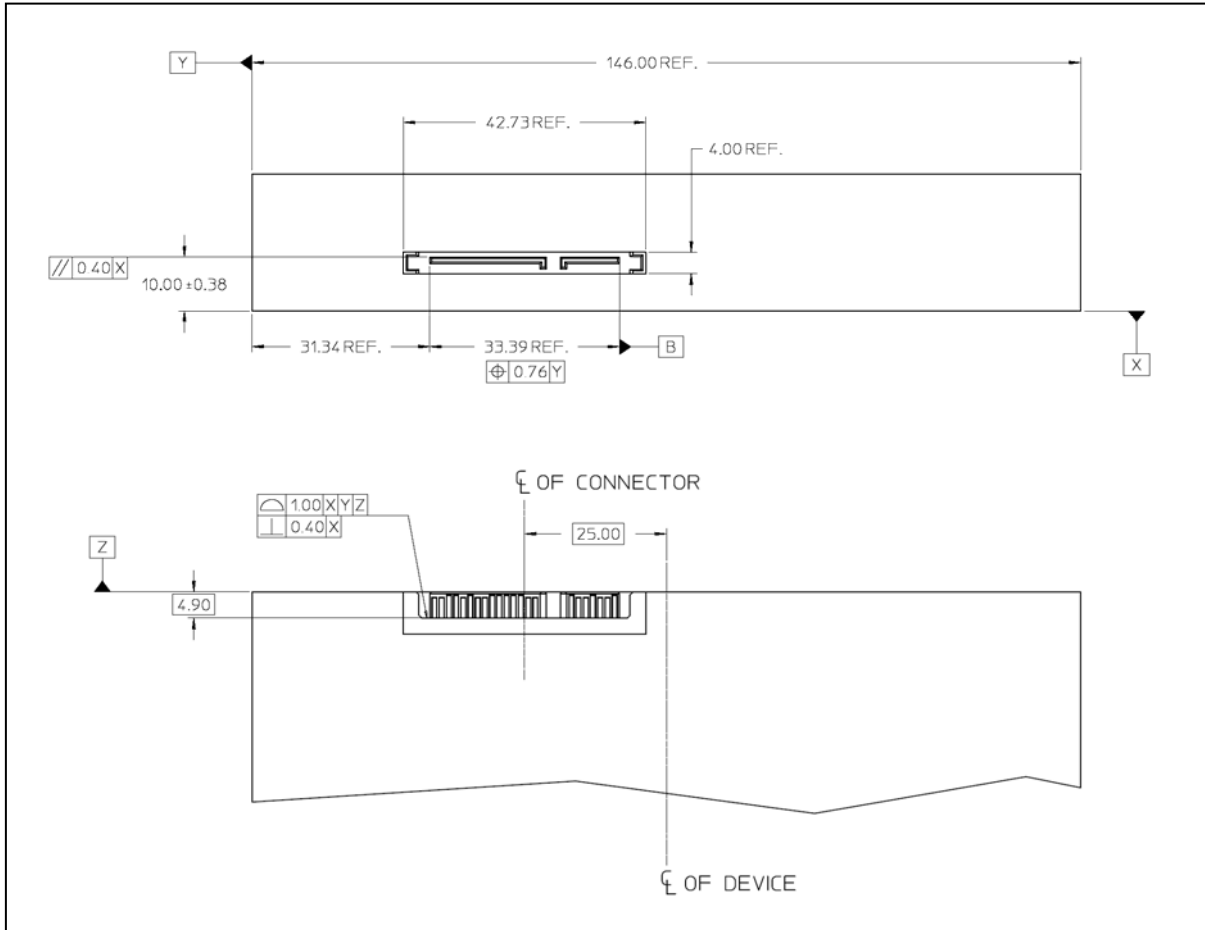


Figure 23 – Optical device plug connector location on 5.25 inch form factor

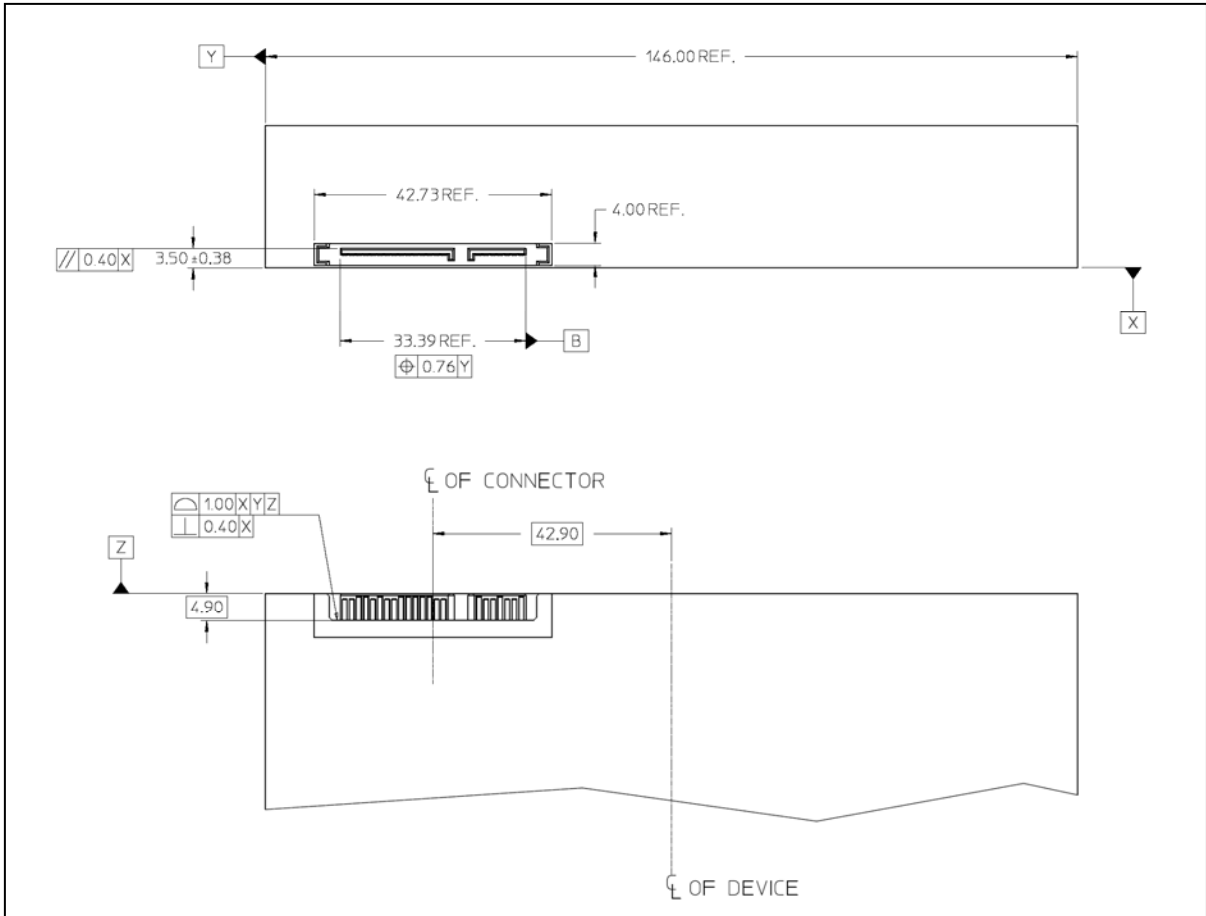


Figure 24 – Non-optical alternate device plug connector location on 5.25 inch form factor

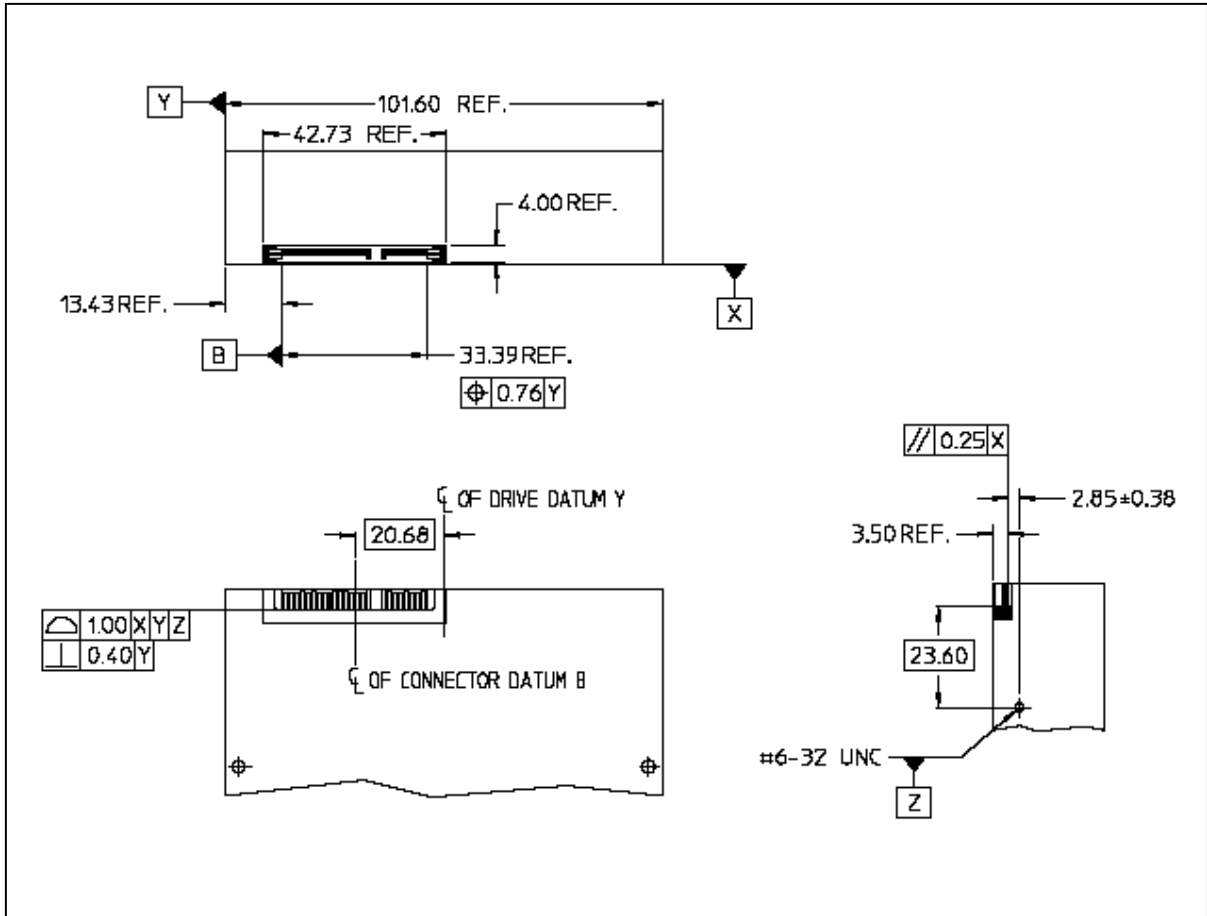


Figure 25 – Device plug connector location on 3.5 inch side mounted device

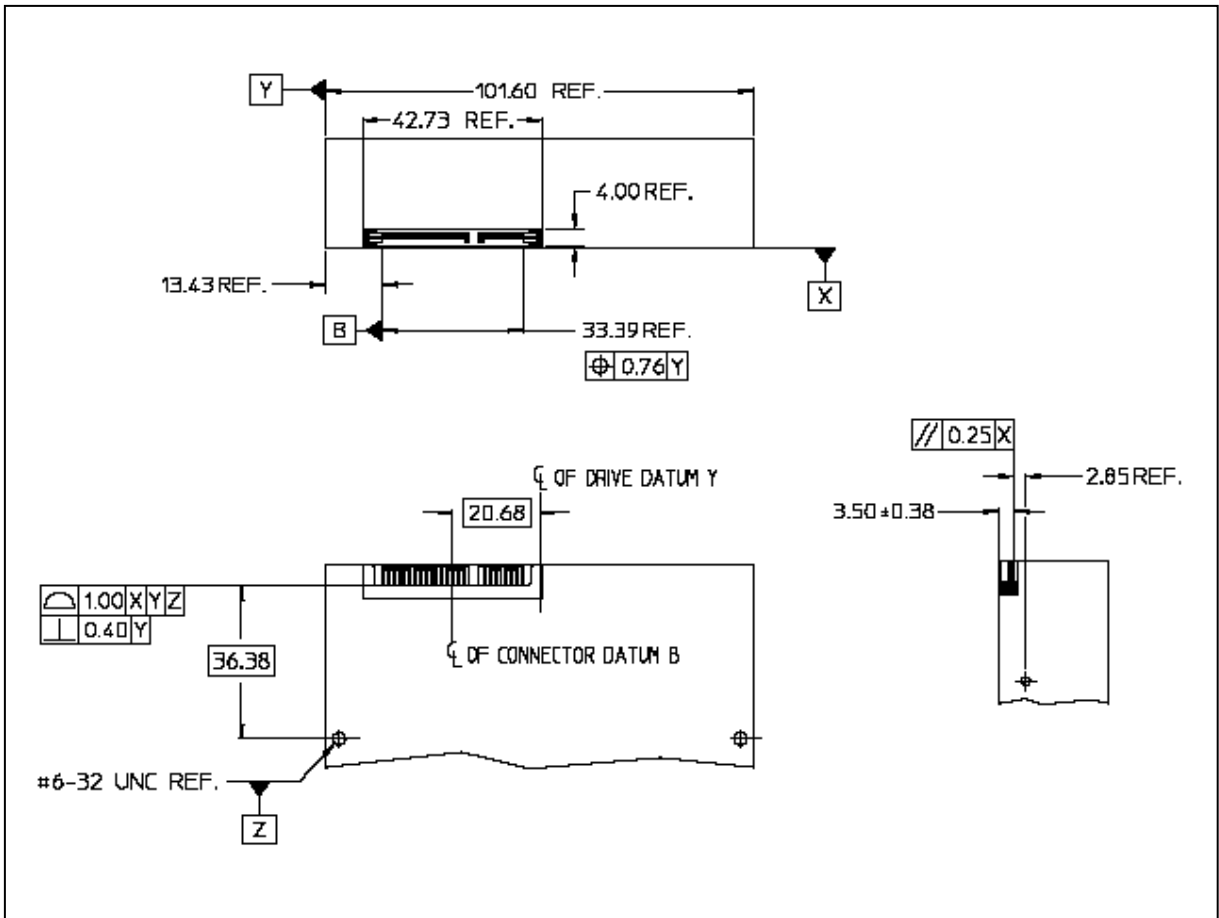


Figure 26 – Device plug connector location on 3.5 inch bottom mounted device



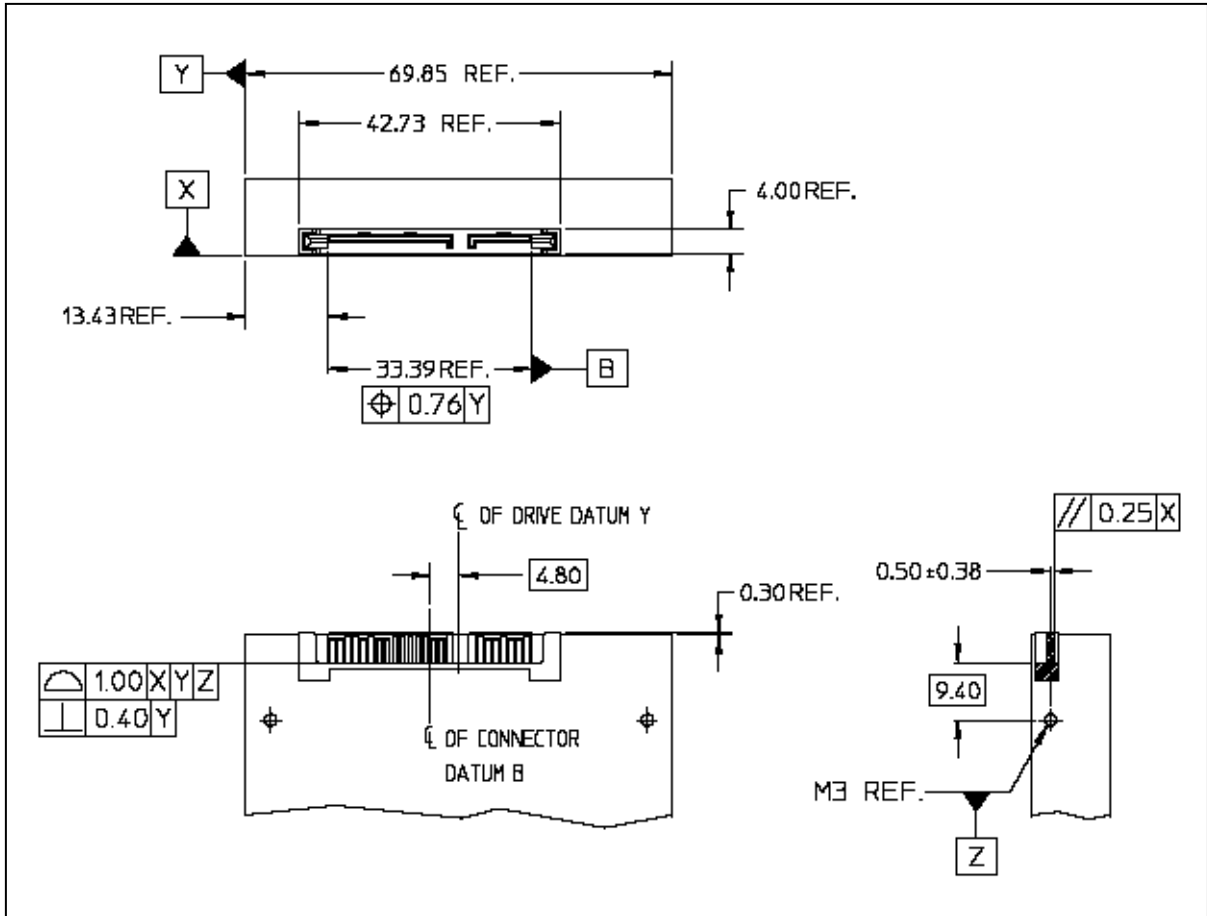


Figure 27 – Device plug connector location on 2.5 inch side mounted device

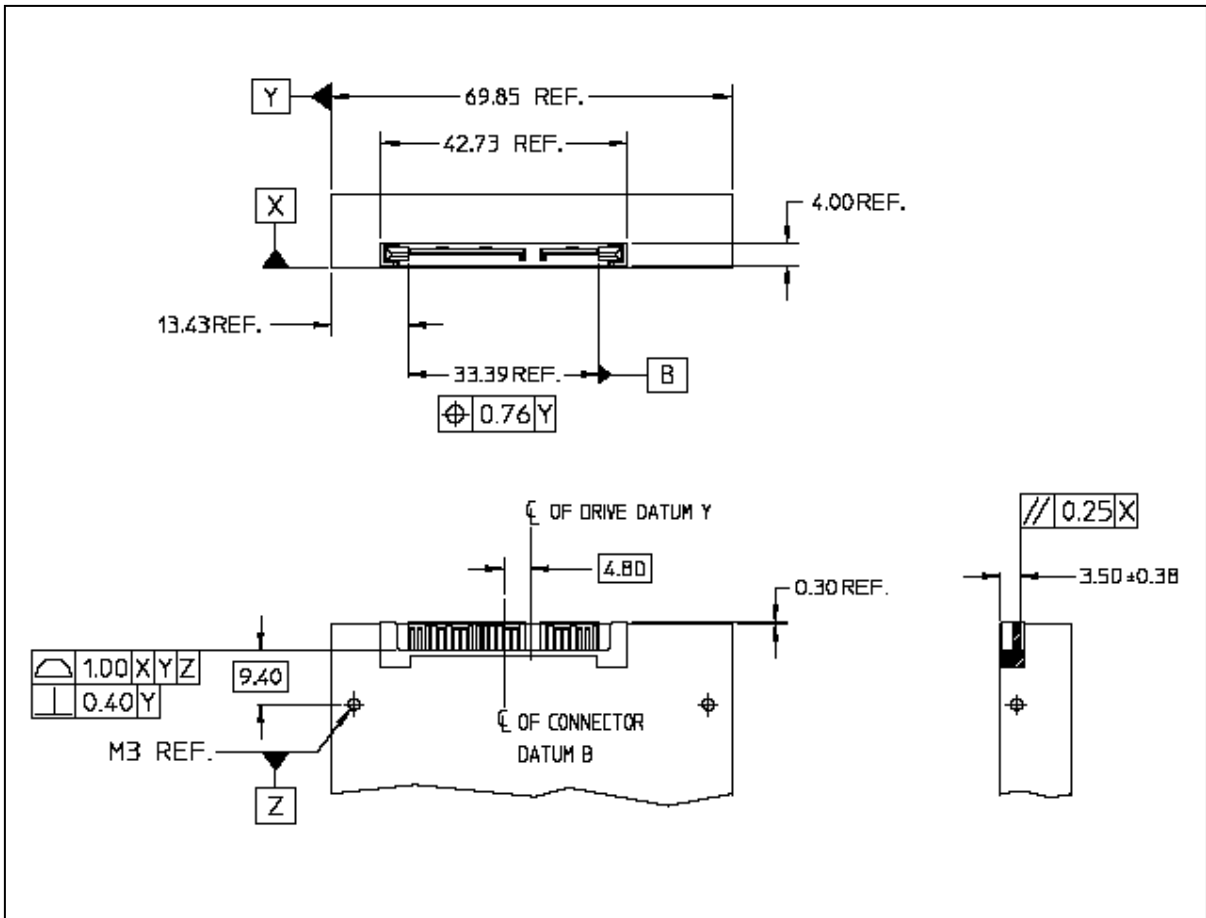


Figure 28 – Device plug connector location on 2.5 inch bottom mounted device

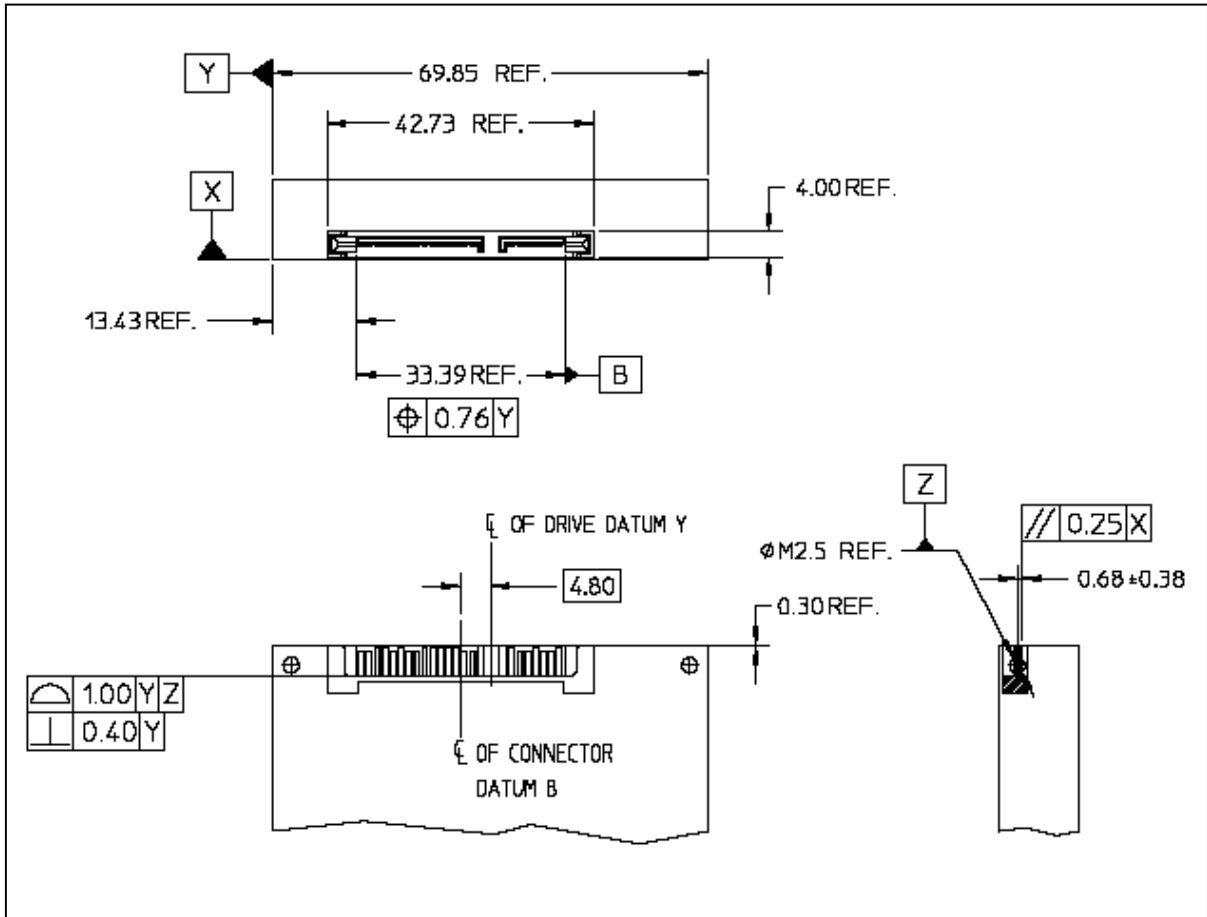


Figure 29 – Device plug connector location on 1.8 inch side mounted device

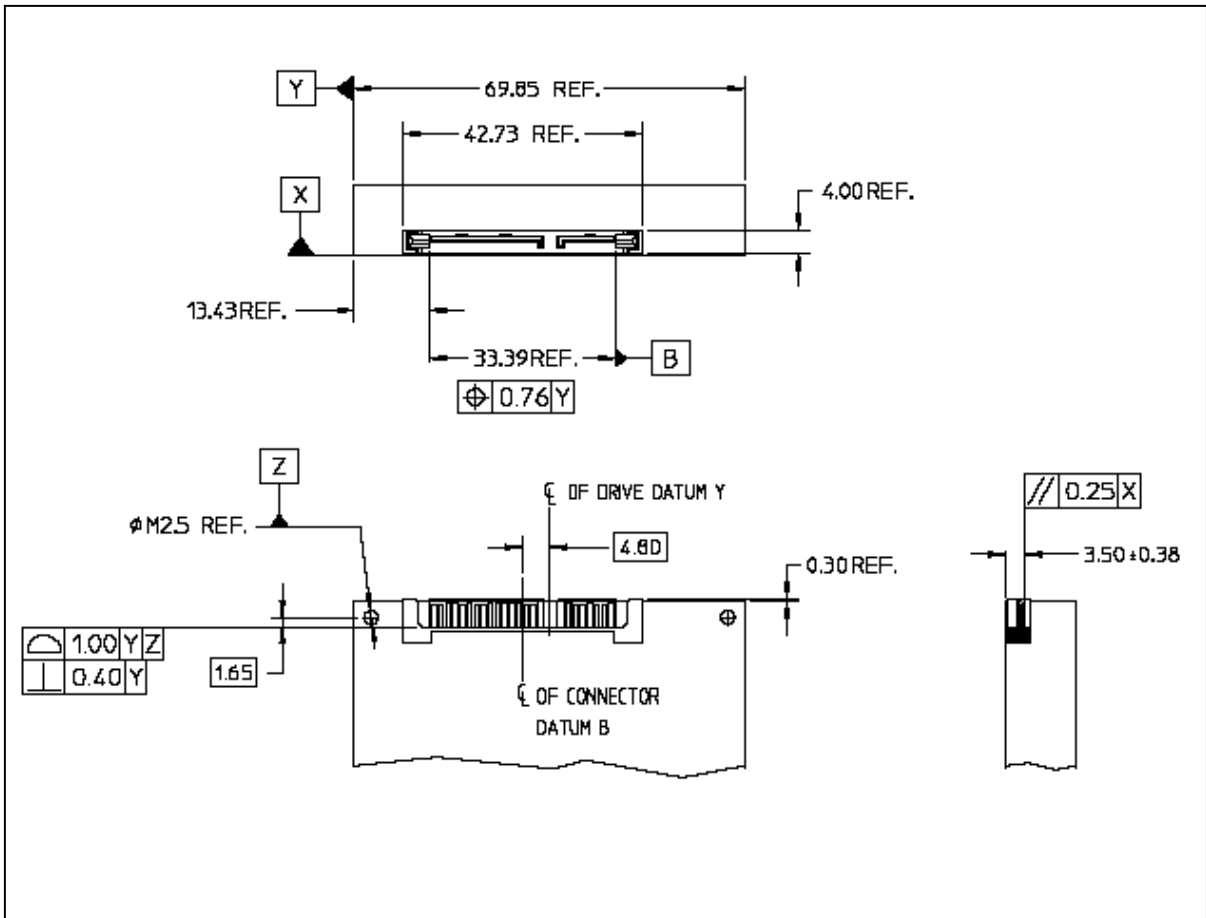
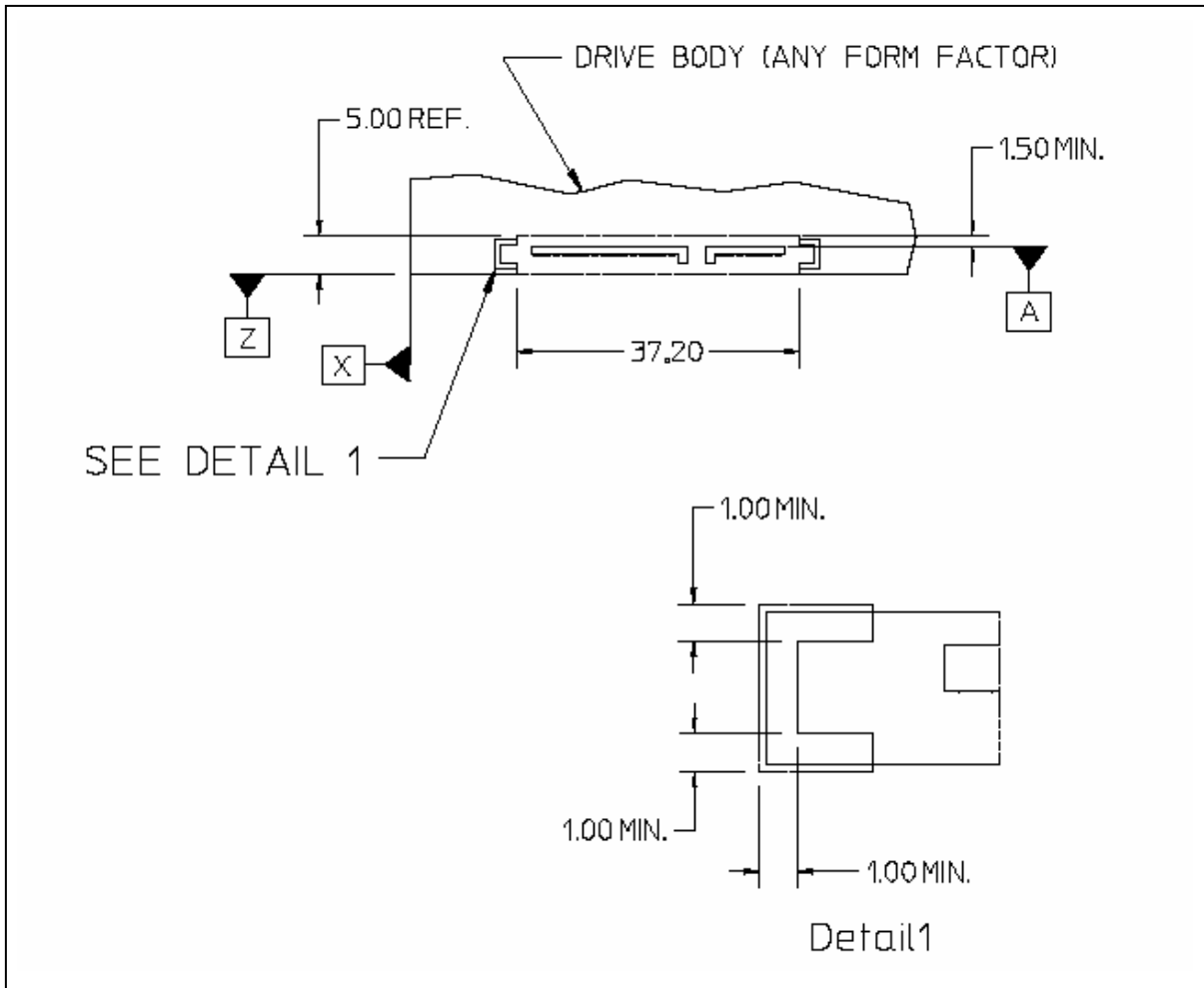


Figure 30 – Device plug connector location on 1.8 inch bottom mounted device



**Figure 31 – Device plug connector keep out zones**

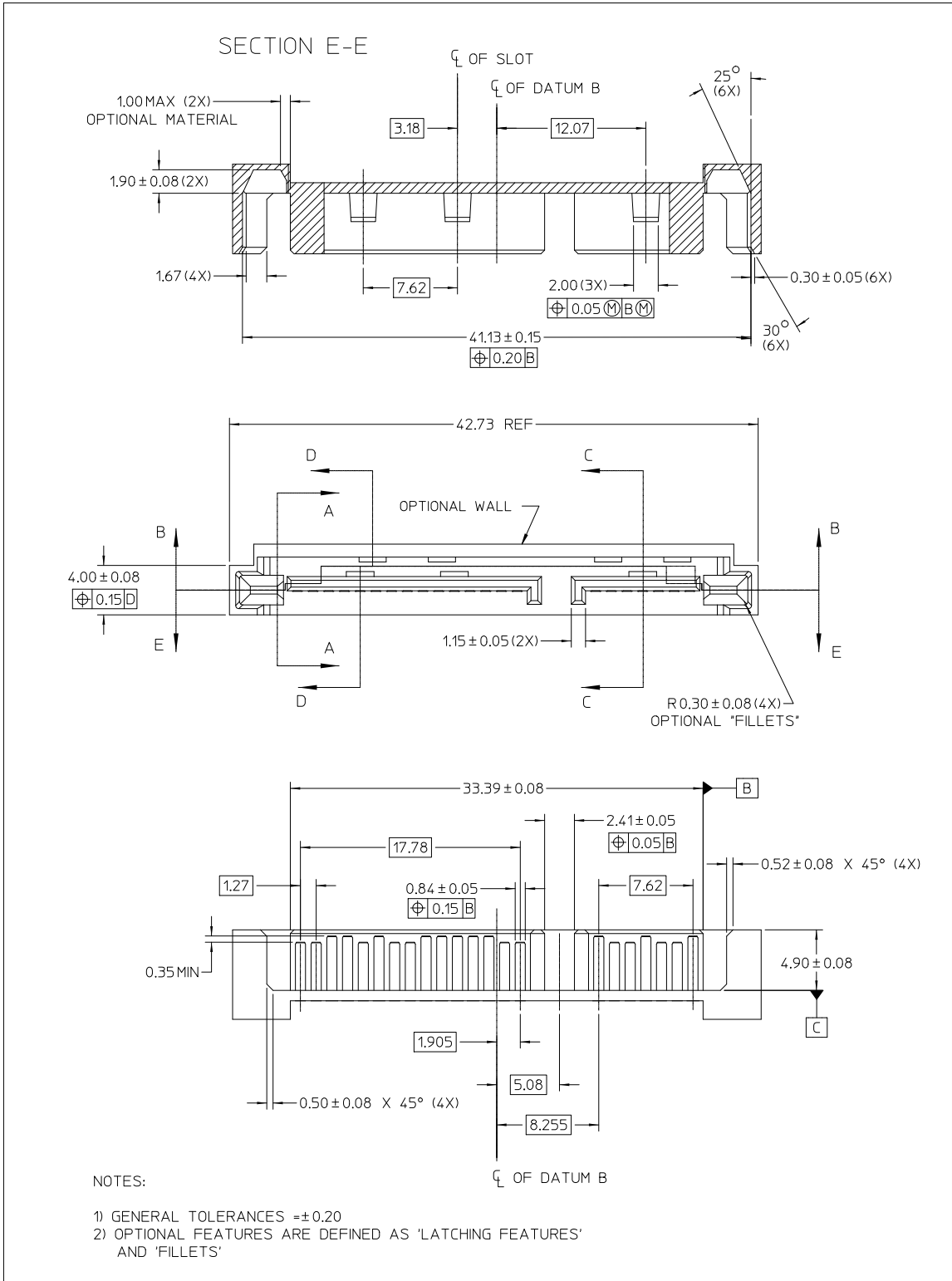
NOTE 9 -The 1.50 mm keepout area above Datum A extends into the form factor to the connector Datum C in Figure 32.

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 32 and Figure 33 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 32 – Device plug connector**

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

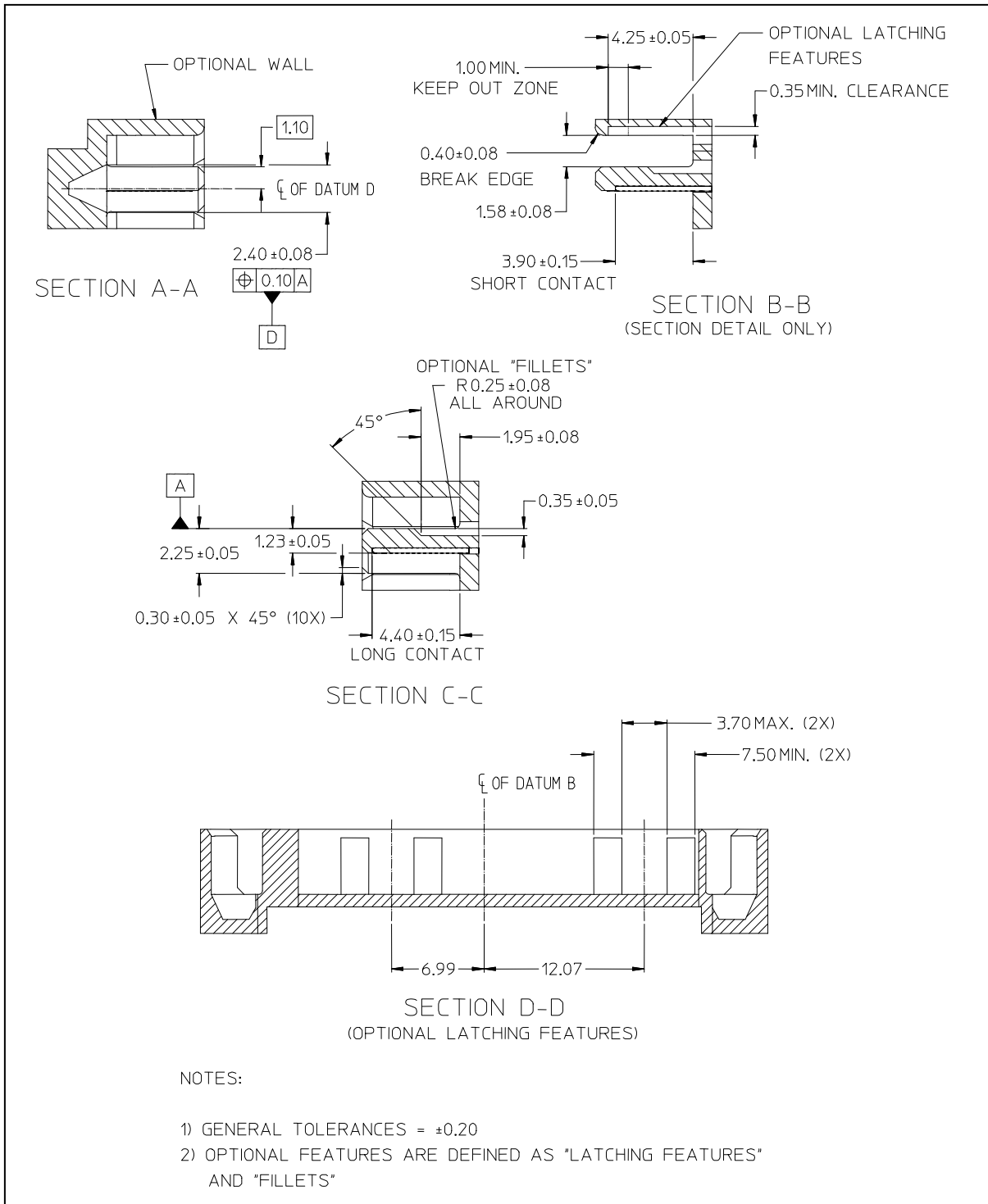


Figure 33 – Device plug connector (additional views)

### 6.2.3.2 Standard SATA connector (3.5 inch & 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.

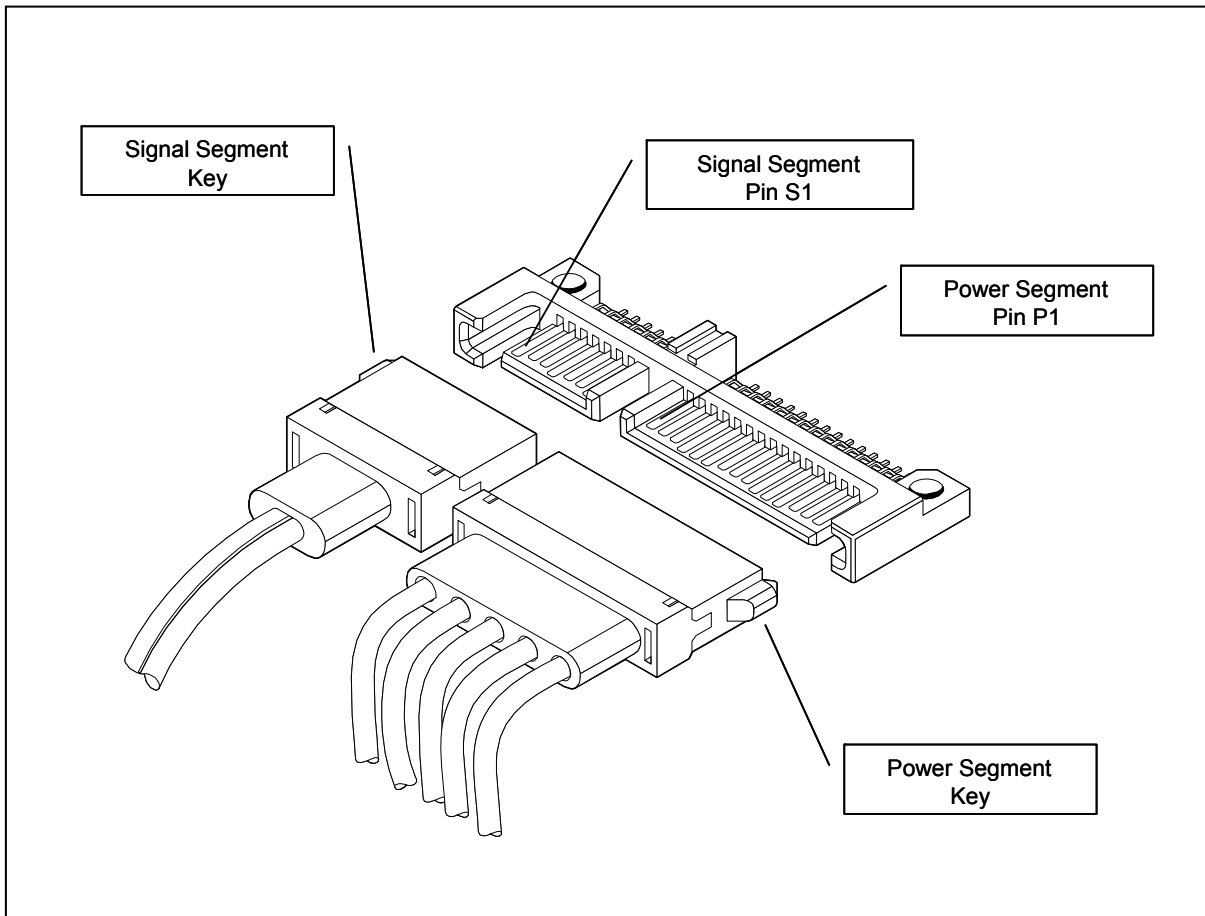
**Table 5 – Standard SATA connector (3.5 inch & 2.5 inch HDD)**

	Name	Type	Description	Cable Usage <sup>b c</sup>	Backplane Usage <sup>c</sup>
Signal Segment Key					
Signal Segment	S1	GND	Ground	1 <sup>st</sup> Mate	2 <sup>nd</sup> Mate
	S2	A+	Differential Signal Pair A	2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
	S3	A-		2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
	S4	GND	Ground	1 <sup>st</sup> Mate	2 <sup>nd</sup> Mate
	S5	B-	Differential Signal Pair B	2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
	S6	B+		2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
	S7	GND	Ground	1 <sup>st</sup> Mate	2 <sup>nd</sup> Mate
Signal Segment "L"					
Central Connector Gap <sup>d</sup>					
Power Segment "L"					
Power Segment	P1	Retired <sup>e f</sup>		2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
	P2	Retired <sup>e f</sup>		2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
	P3	DEVSLP <sup>e</sup>	Enter/Exit DevSleep	1 <sup>st</sup> Mate	2 <sup>nd</sup> Mate
	P4	GND	Ground	1 <sup>st</sup> Mate	1 <sup>st</sup> Mate
	P5	GND	Ground	1 <sup>st</sup> Mate	2 <sup>nd</sup> Mate
	P6	GND	Ground	1 <sup>st</sup> Mate	2 <sup>nd</sup> Mate
	P7	V <sub>5</sub>	5 V Power, Pre-charge	1 <sup>st</sup> Mate	2 <sup>nd</sup> Mate
	P8	V <sub>5</sub>	5 V Power	2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
	P9	V <sub>5</sub>	5 V Power	2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
	P10	GND	Ground	1 <sup>st</sup> Mate	2 <sup>nd</sup> Mate
	P11	DAS/DSS/DHU	Device Activity Signal / Disable Staggered Spinup/ Direct Head Unload / Vendor Specific <sup>a</sup>	2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
	P12	GND	Ground	1 <sup>st</sup> Mate	1 <sup>st</sup> Mate
	P13	V <sub>12</sub>	12 V Power, Pre-charge	1 <sup>st</sup> Mate	2 <sup>nd</sup> Mate
	P14	V <sub>12</sub>	12 V Power	2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
	P15	V <sub>12</sub>	12 V Power	2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
Power Segment Key					
<sup>a</sup> For specific optional usage of pin P11 (see 6.13). <sup>b</sup> Although the mate order is shown, hot plugging is not supported if using the cable connector receptacle. <sup>c</sup> All mate sequences assume zero angular offset between connectors. <sup>d</sup> The signal segment and power segment may be separate. <sup>e</sup> 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. <sup>f</sup> It is recommended to have P1 and P2 connected together for the purpose of legacy functionality. Pin P3 should be a no connect if DEVSLP is not implemented.					



**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 34 – Connector pin and feature locations**

**6.2.4 Signal cable receptacle connector**

Figure 35 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 36 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.

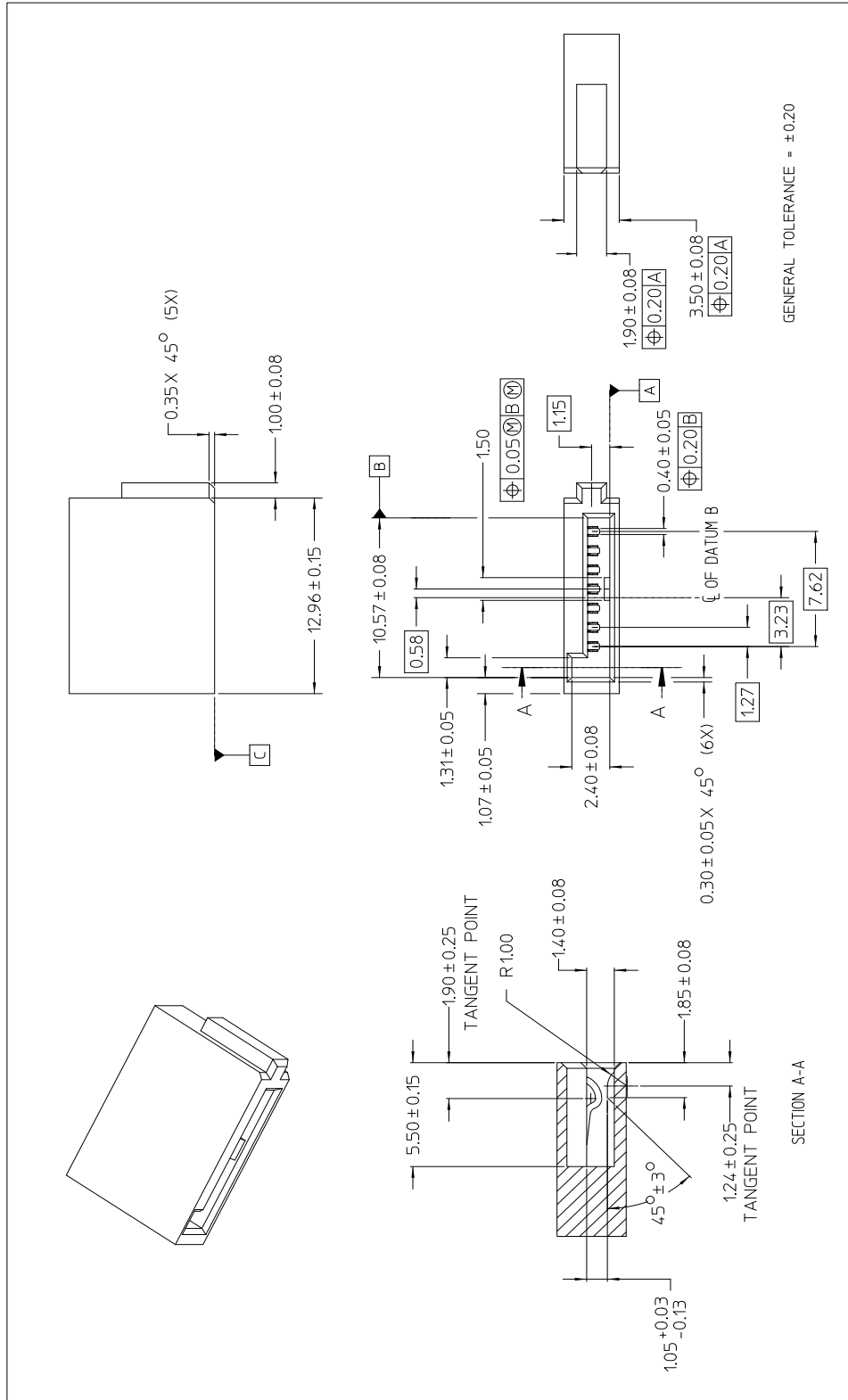
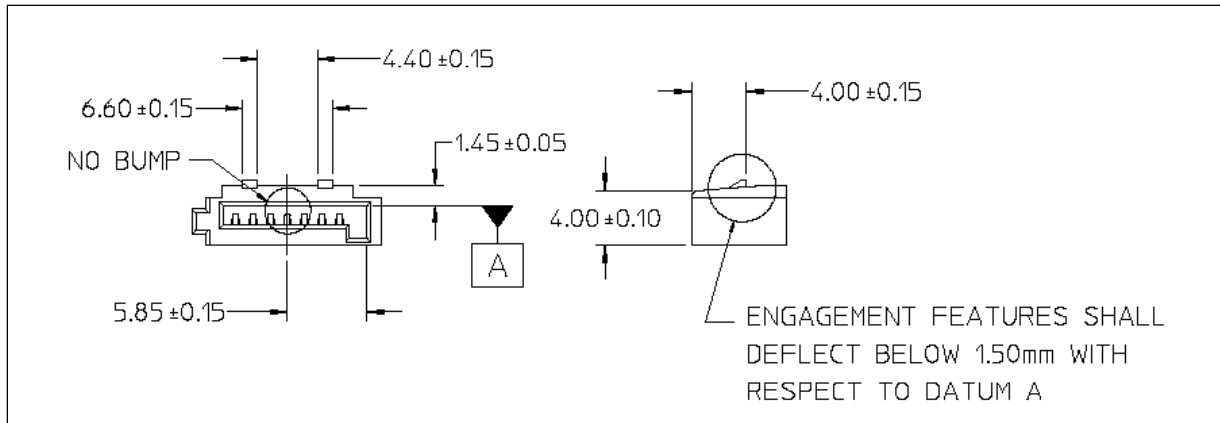


Figure 35 – Cable receptacle connector interface dimensions



**Figure 36 – 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:

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

## 6.2.5 Signal host plug connector

### 6.2.5.1 Host signal plug dimensions

The signal host 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 37 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.

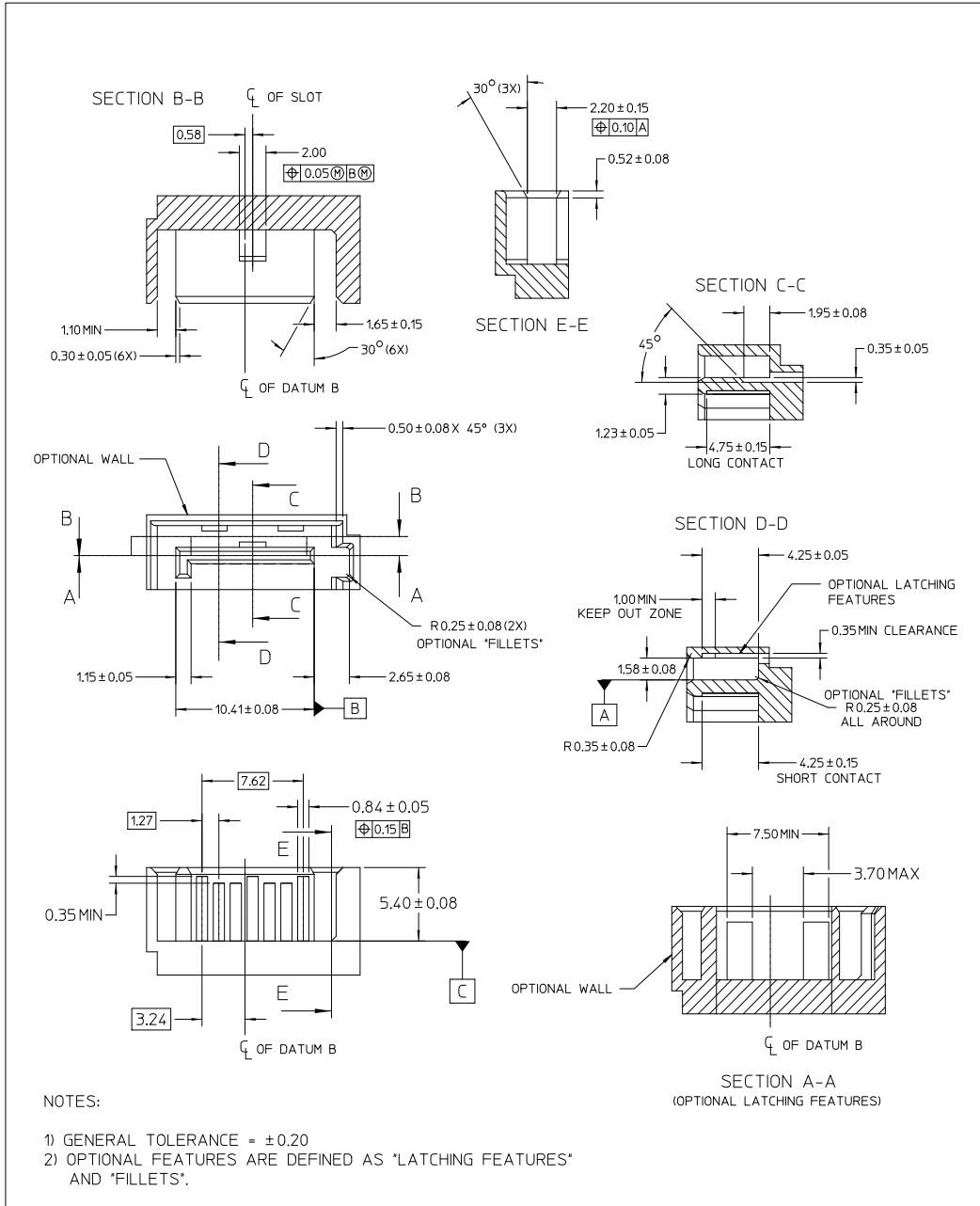


Figure 37 – Host signal plug connector interface dimensions

### 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 38 shows the recommended clearance or spacing. Figure 39 shows the recommended clearance and orientation to allow access for latching cables.

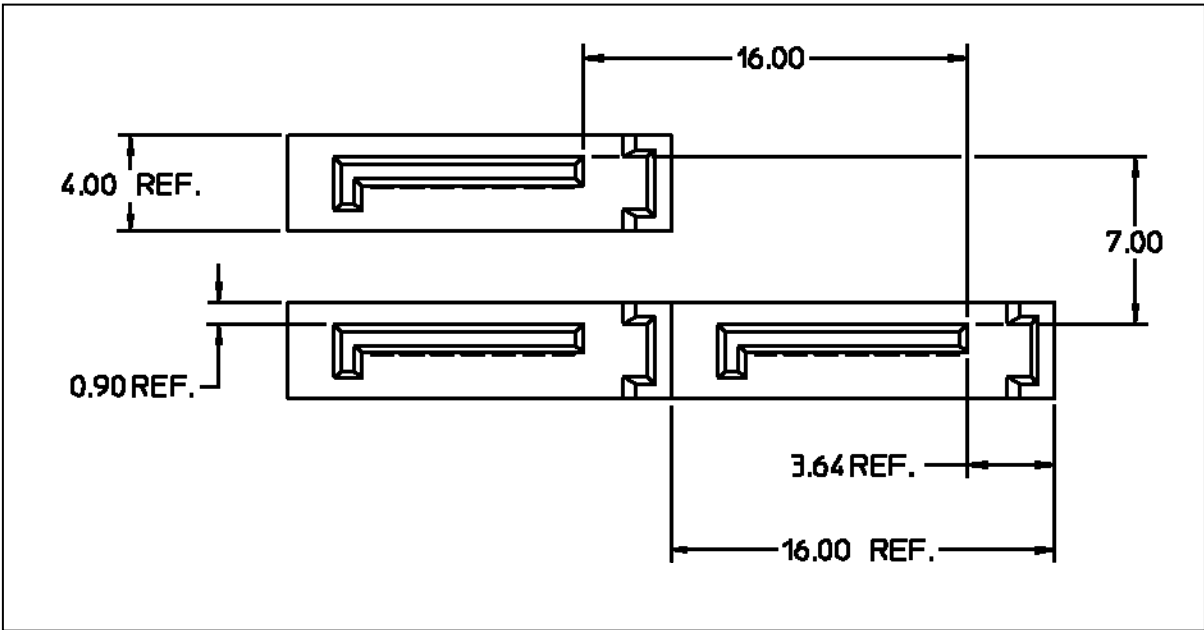


Figure 38 – Non-latching connector stack spacing and orientation

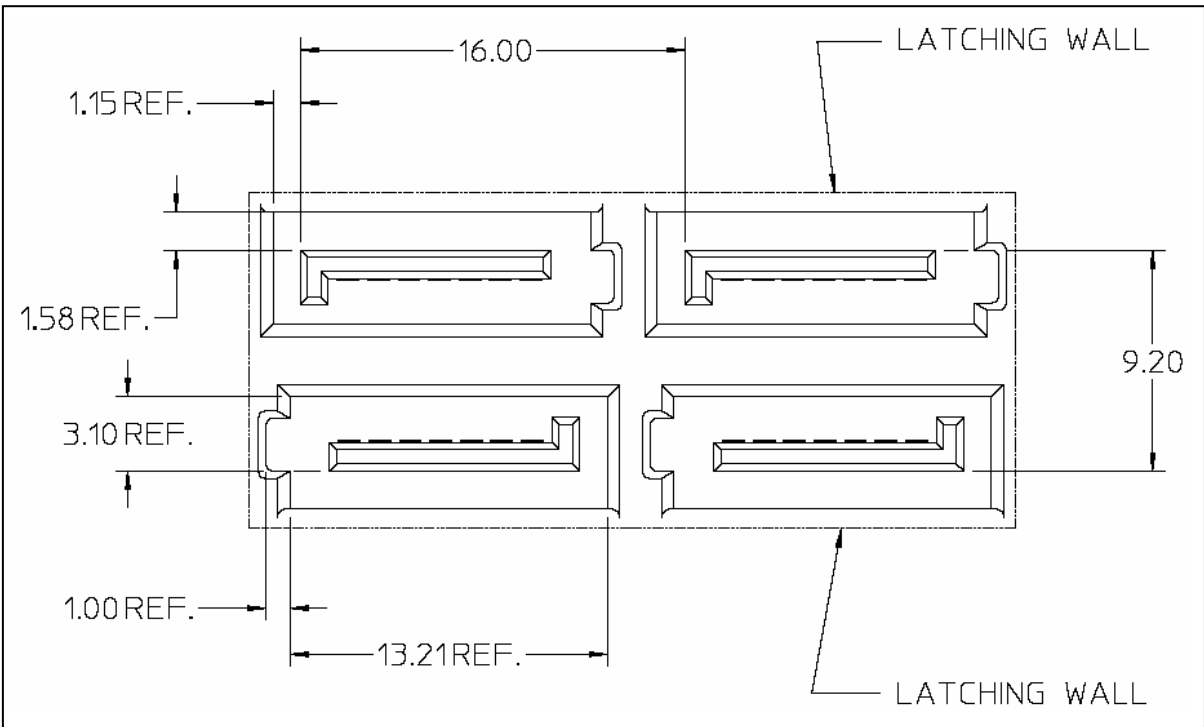


Figure 39 – Latching connector stack spacing and orientation

## **6.2.6 Backplane connector**

### **6.2.6.1 Backplane connector dimensions**

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 40.

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.

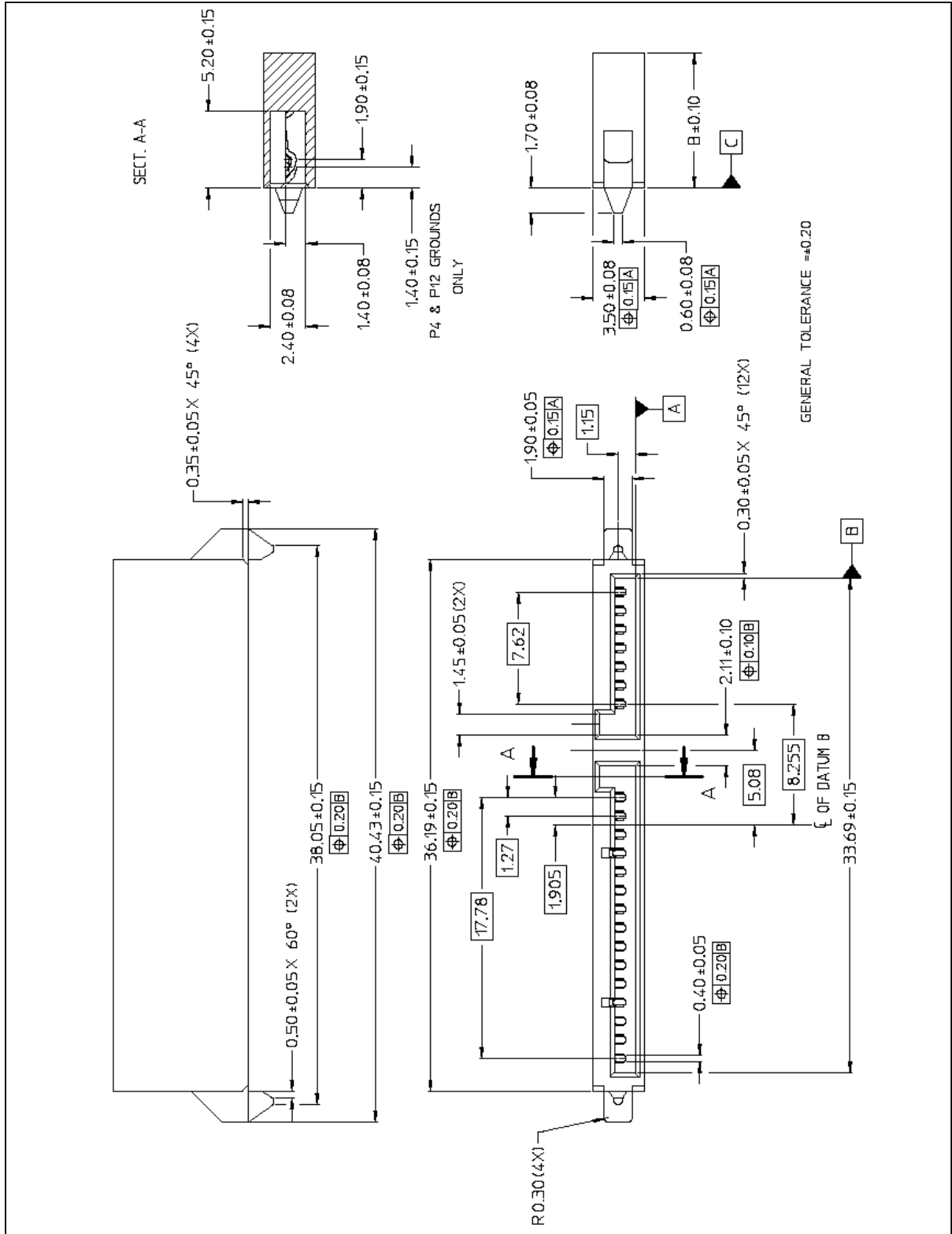
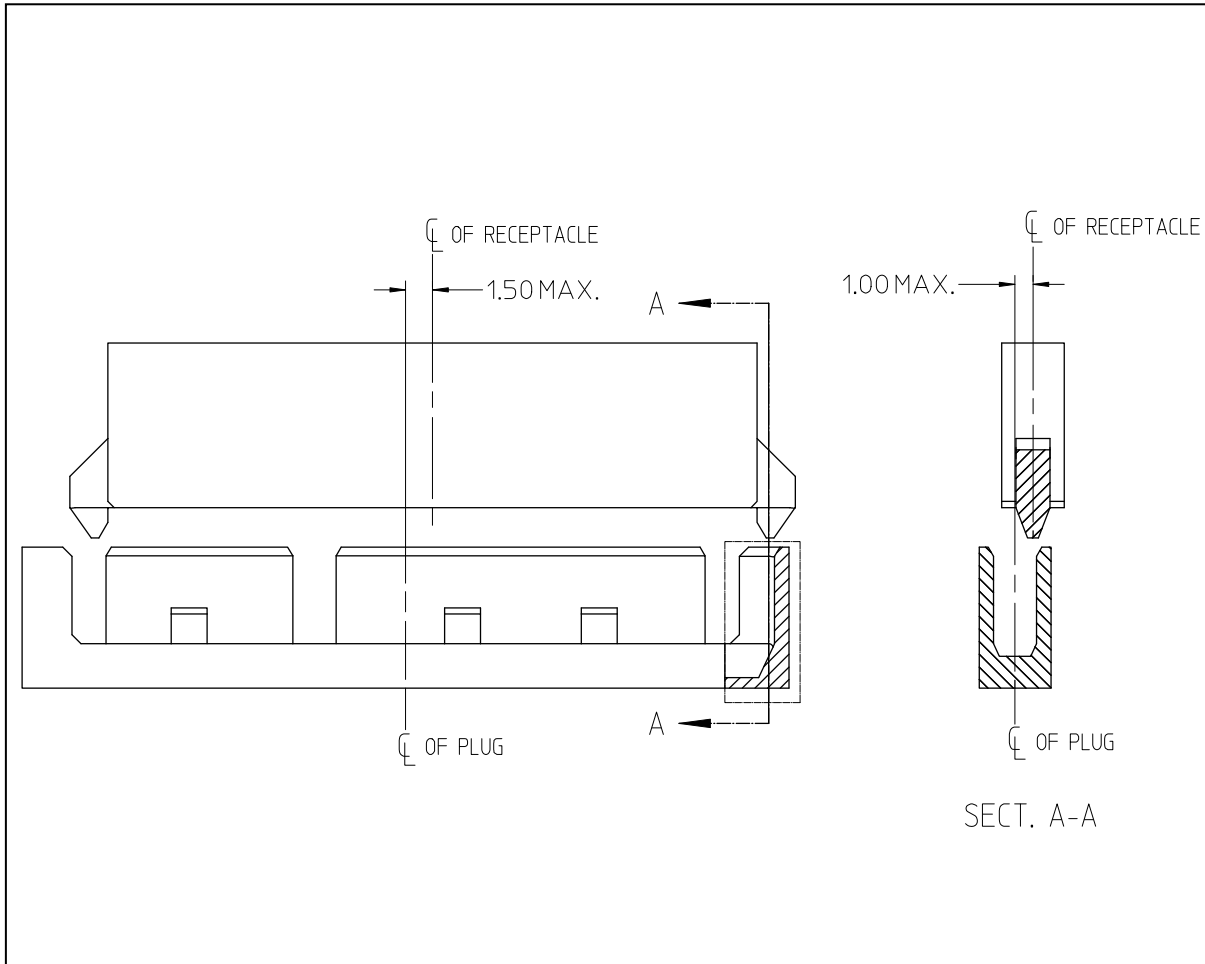


Figure 40 – Backplane connector interface dimensions



### 6.2.6.2 Backplane connector configuration and blind-mating tolerance

The maximum blind-mate misalignment tolerances are  $\pm 1.50$  mm and  $\pm 1.00$  mm, respectively, for two perpendicular axes illustrated in Figure 41. Any skew angle of the plug, with respect to the receptacle, reduces the blind-mate tolerances.

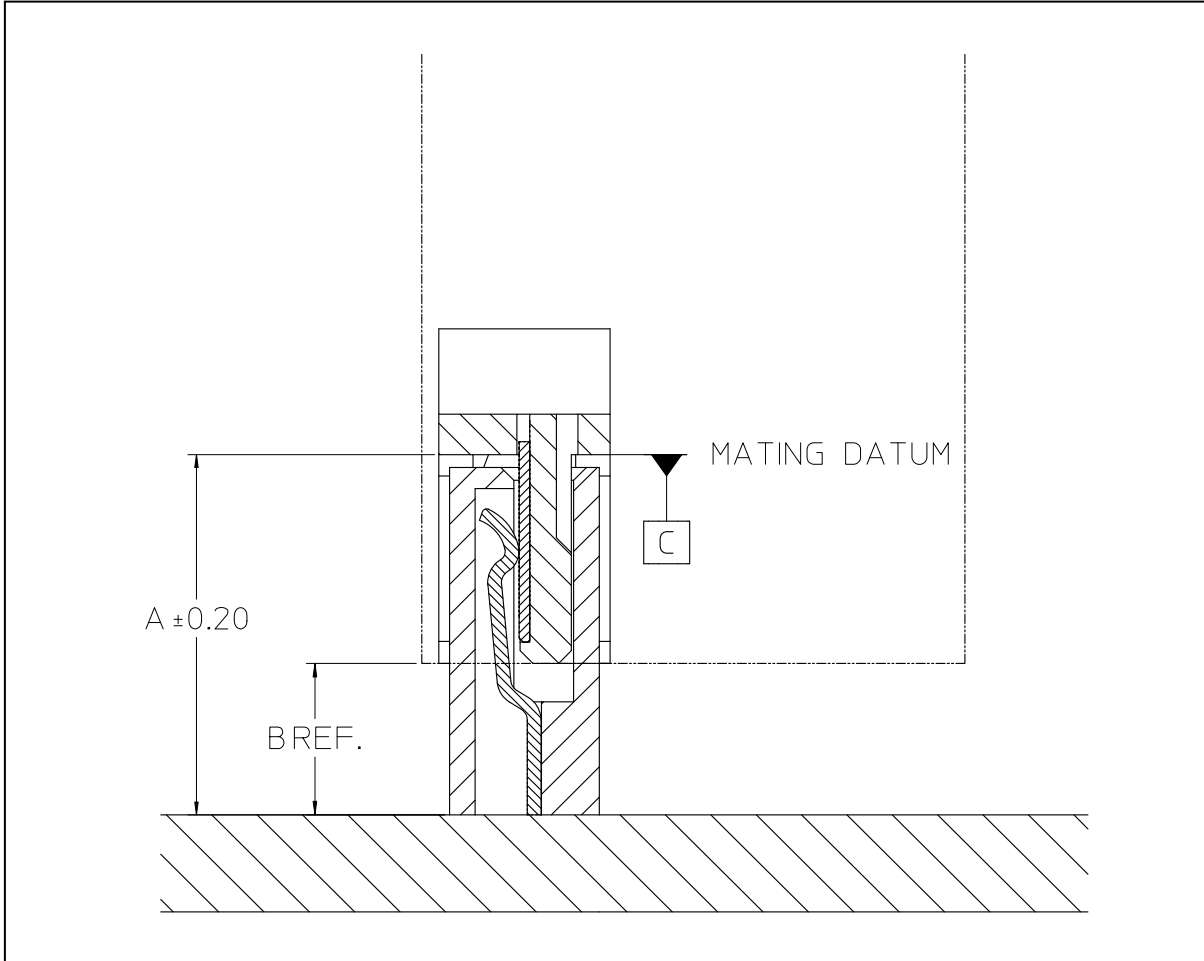


**Figure 41 – Connector pair blind-mate misalignment tolerance**

The device-to-backplane mating configuration is shown in Figure 42. 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 42 – 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 43 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 44 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.

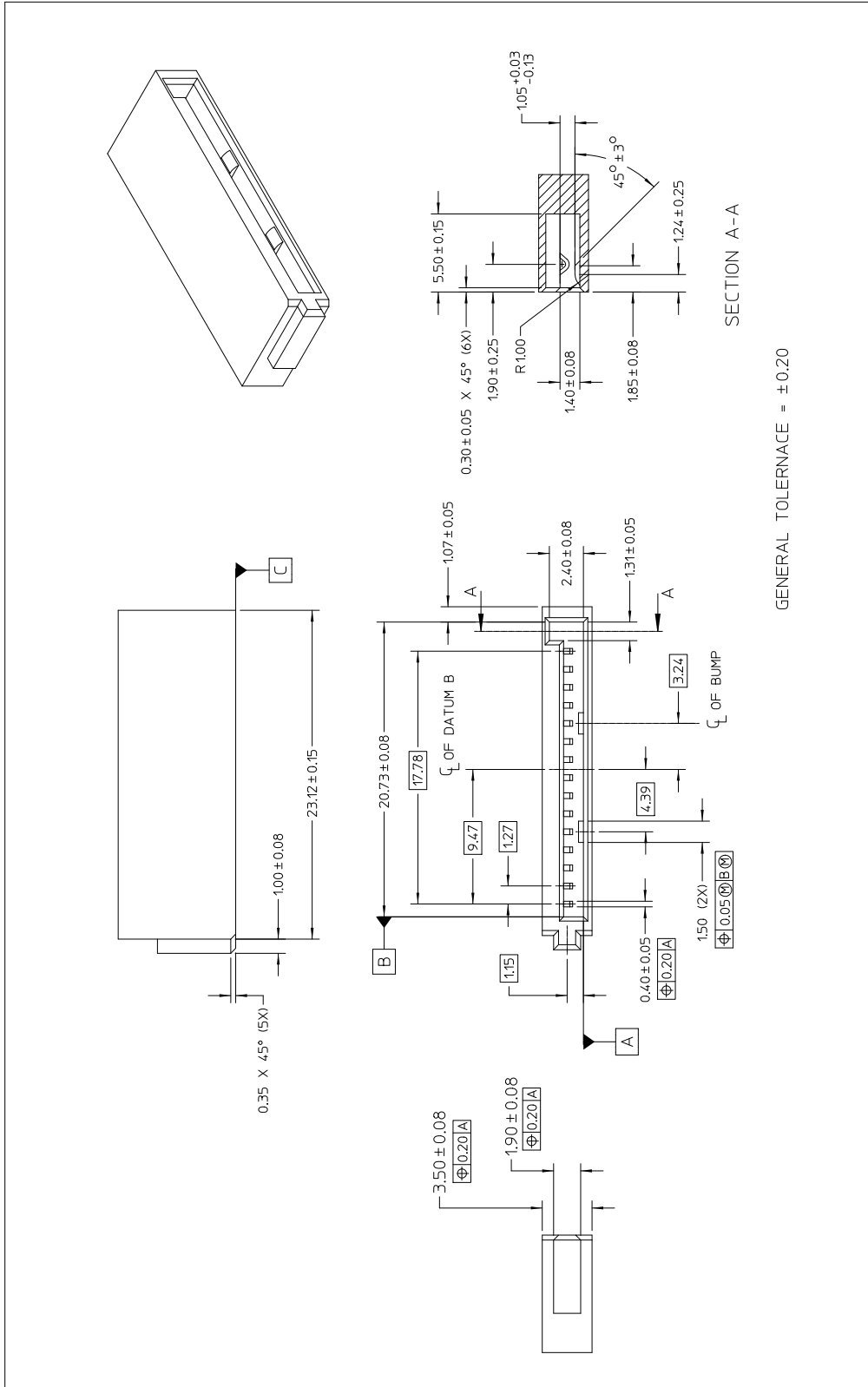
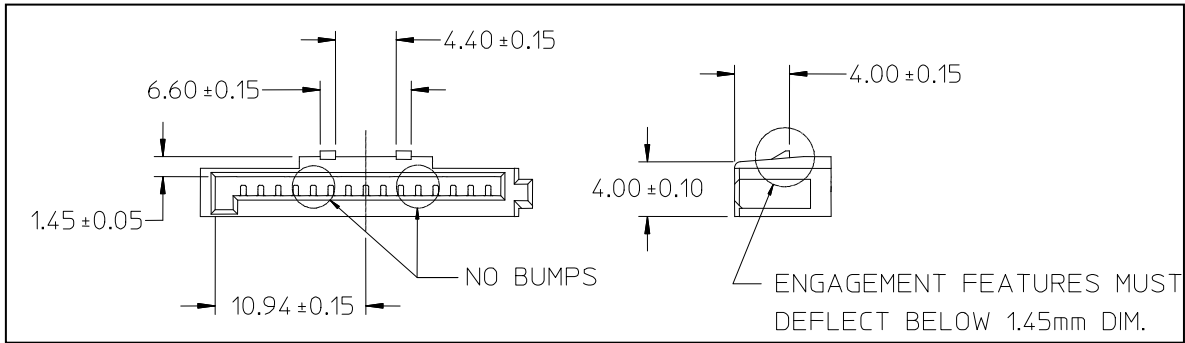


Figure 43 – Power receptacle connector interface dimensions



**Figure 44 – 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.

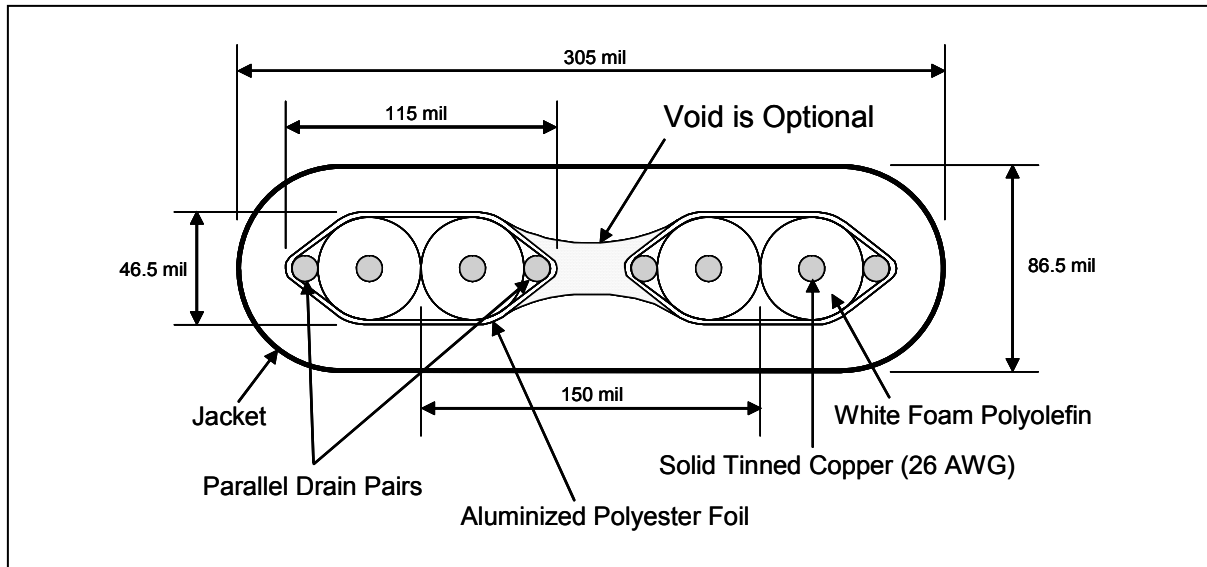
This specification does not specify a standard internal single lane cable. Any cable that meets the electrical requirements given in 6.4 is considered an acceptable internal single lane cable. The connector and cable vendors have the flexibility to choose cable constructions and termination methods based on performance and cost considerations. An example cable construction is given in Figure 45 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 45 for details.



**Figure 45 – 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 mm of Hg to 800 mm of Hg.

If an EIA (Electronic Industry Association) 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 is the connector housing and contact electrical requirements.

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

<b>Parameter</b>	<b>Procedure</b>	<b>Requirement</b>
Insulation resistance	EIA 364-21 After 500 VDC for 1 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 1 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: <ol style="list-style-type: none"> <li>1) mount connector to a test PCB;</li> <li>2) wire 3 adjacent pins in parallel for supply (or the minimum number required by the connector type);</li> <li>3) wire 3 adjacent pins in parallel for return (or the minimum number required by the connector type);</li> <li>4) apply a DC current of 3 times the current rating per contact to the supply pins, returning through the return pins; and</li> <li>5) record temperature rise when thermal equilibrium is reached.</li> </ol>	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**

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 min 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 $x=3.7 \times$ 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 mm/min.	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 mm/min .	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 mm/min.	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 mm/min.	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 mm/min.	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 mm/min .	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/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**

<b>Parameter</b>	<b>Procedure</b>	<b>Requirement</b>
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 <sup>b</sup> .	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. 30 minutes in each of three mutually perpendicular planes <sup>b</sup> .	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 <sup>a</sup> .	
Temperature life	EIA 364-17 Test Condition III Method A. Subject mated connectors to temperature life at +85 °C for 500 hours <sup>a</sup> .	
Thermal shock	EIA 364-32 Test Condition I. Subject mated connectors to 10 cycles between –55 °C and +85 °C <sup>a</sup> .	
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 <sup>a</sup> .	
<sup>a</sup> 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. <sup>b</sup> 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**

<b>Parameter</b>	<b>Procedure</b>	<b>Requirement</b>
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 <sup>a</sup>			2 <sup>a</sup>
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	

<sup>a</sup> 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 / h.

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 host bus adapter to a backplane within the same enclosure or server, or for connecting multiple Serial ATA links from a host bus adapter 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 Gbps and 3.0 Gbps; 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**

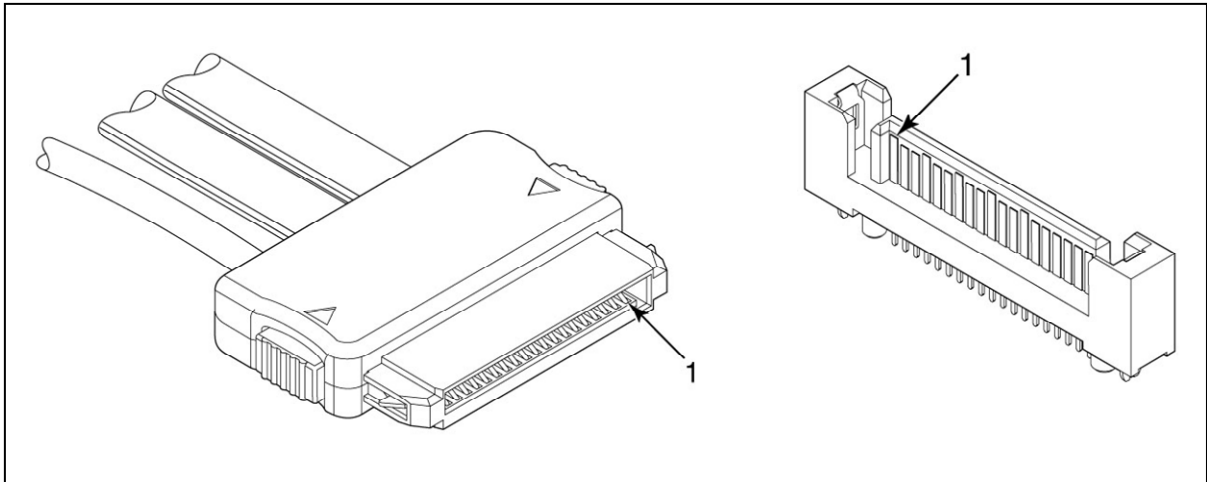
Three components are defined in this section for Multilane applications. Each component has a 2 Lane and a 4 Lane version. Pin assignments are provided at the end of this section.

The three components are:

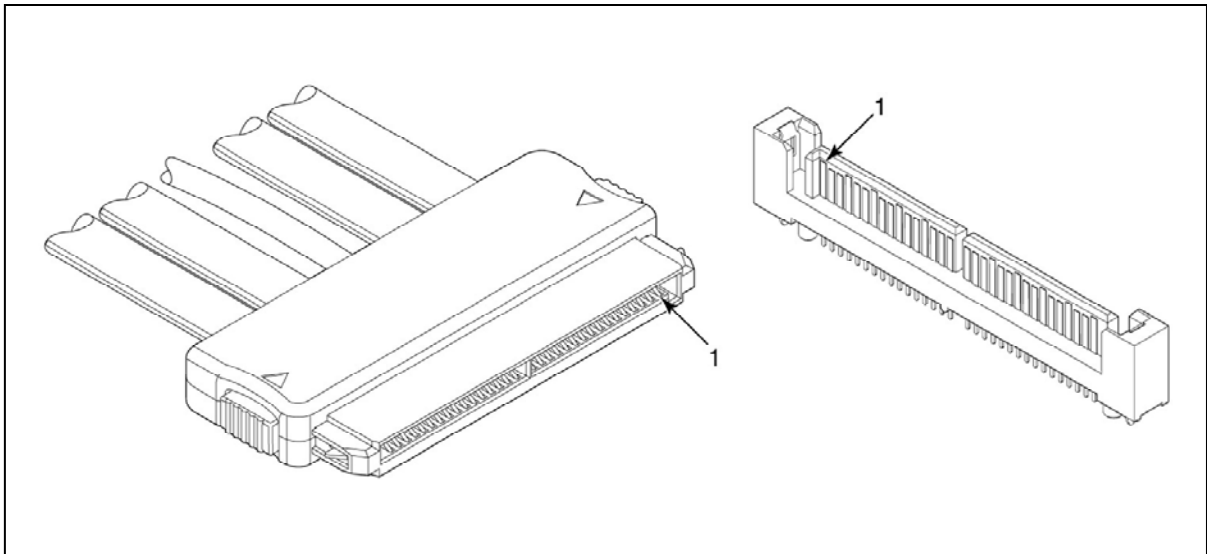
- a) cable Receptacles and Backshells;
- b) vertical Headers; and
- c) right Angle Headers.

### **6.2.11.2.4 Cable receptacles and backshells**

Figure 46 and Figure 47 show isometric drawings of the internal Multilane cables and connectors. Refer to SFF-8484 for dimensions and mechanical details.



**Figure 46 – Isometric drawings of the internal 2 lane cable and connector**



**Figure 47 – Isometric drawings of the internal 4 lane cable and connector**

### 6.2.11.3 4 lane pin assignments

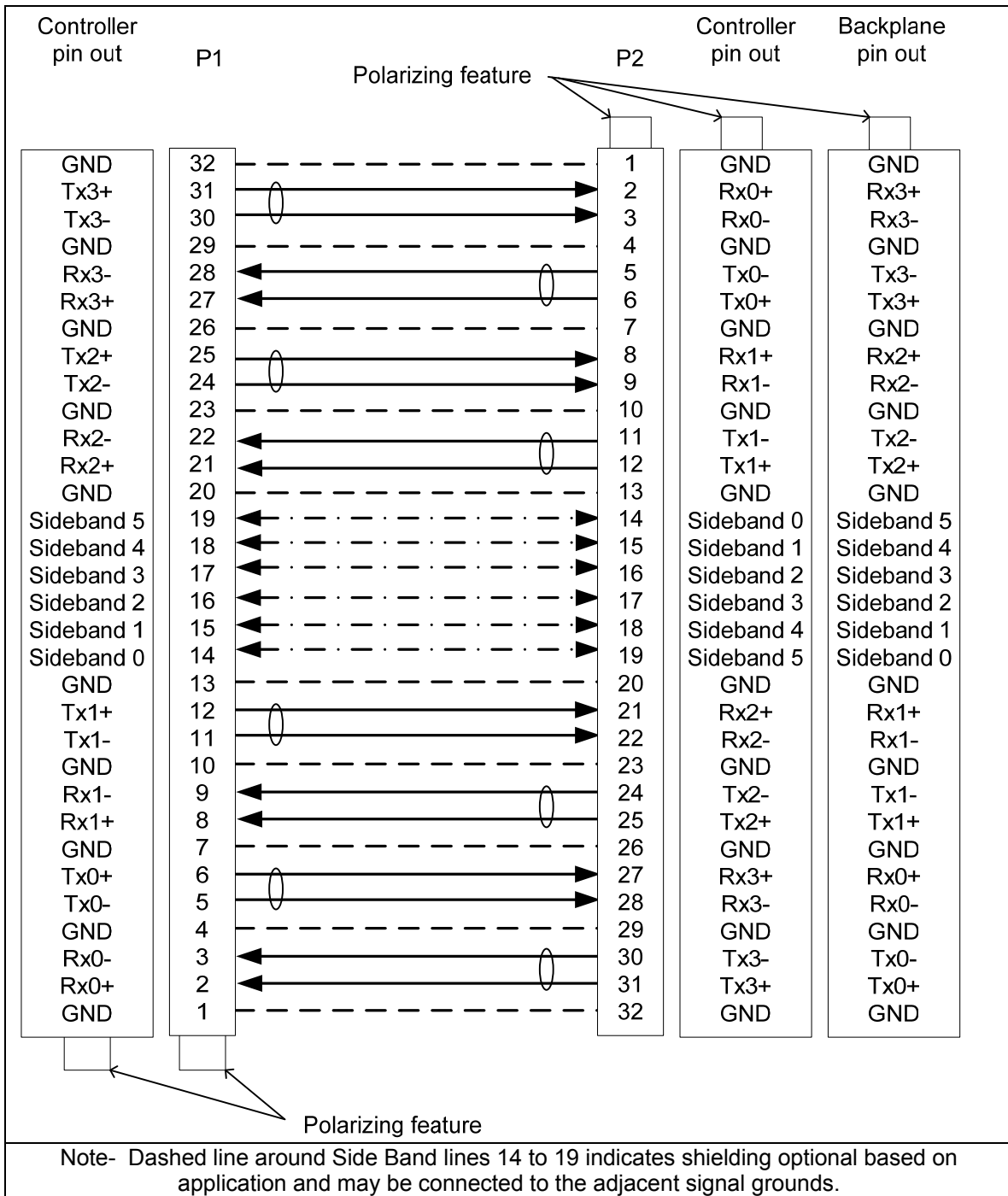
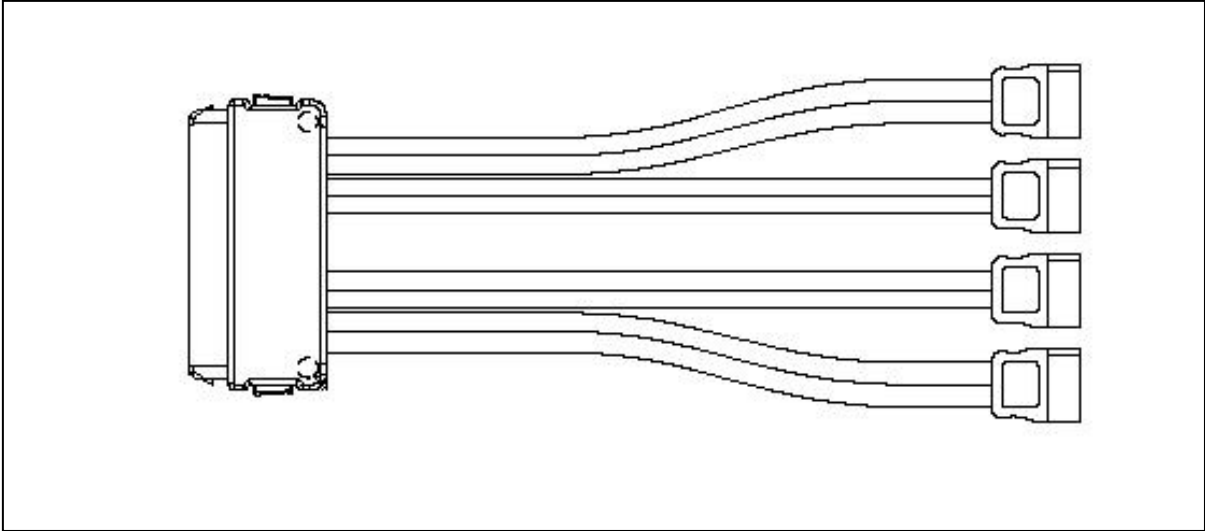


Figure 48 – 4 lane pin assignments



**Figure 49 – 4 lane to 4 x 1 lanes, fanout implementation**

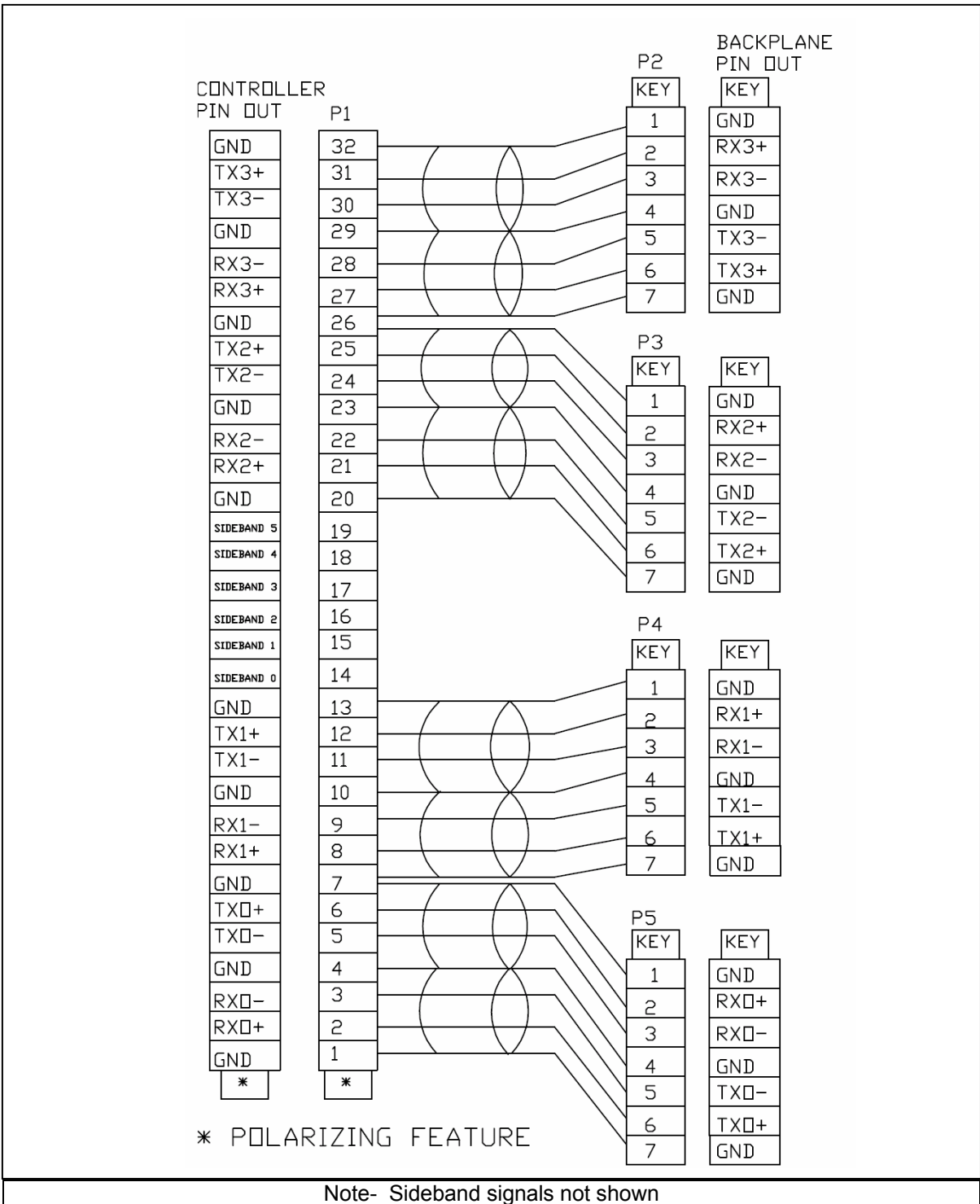


Figure 50 – 4 lane fanout pin assignments

### 6.2.11.4 2 lane pin assignments

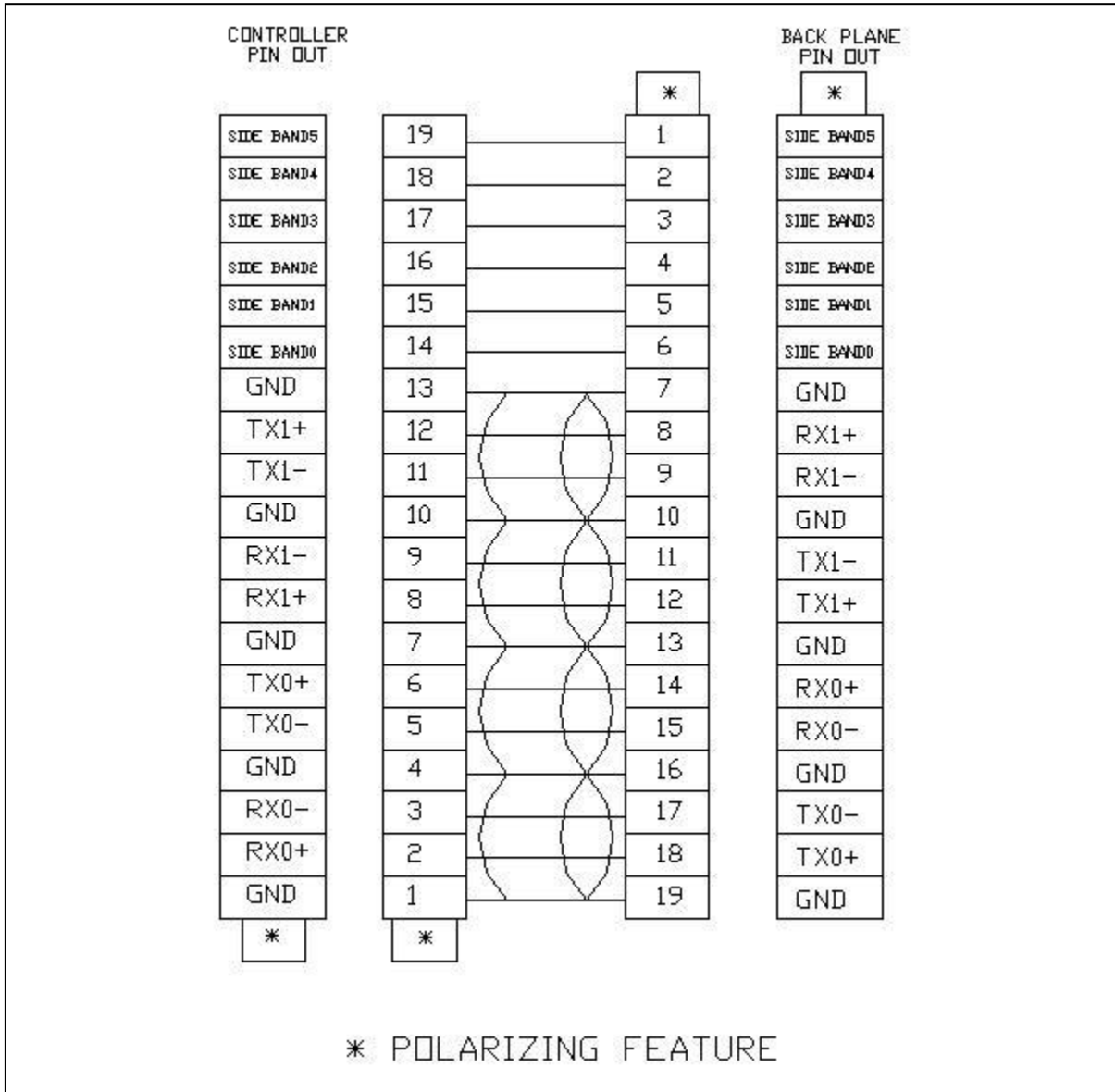


Figure 51 – 2 lane fanout pin assignments

## 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 host bus adapter to a backplane within the same enclosure or server, or for connecting multiple Serial ATA links from a host bus adapter 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 Gbps, 3.0 Gbps and 6.0 Gbps; 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 52 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.

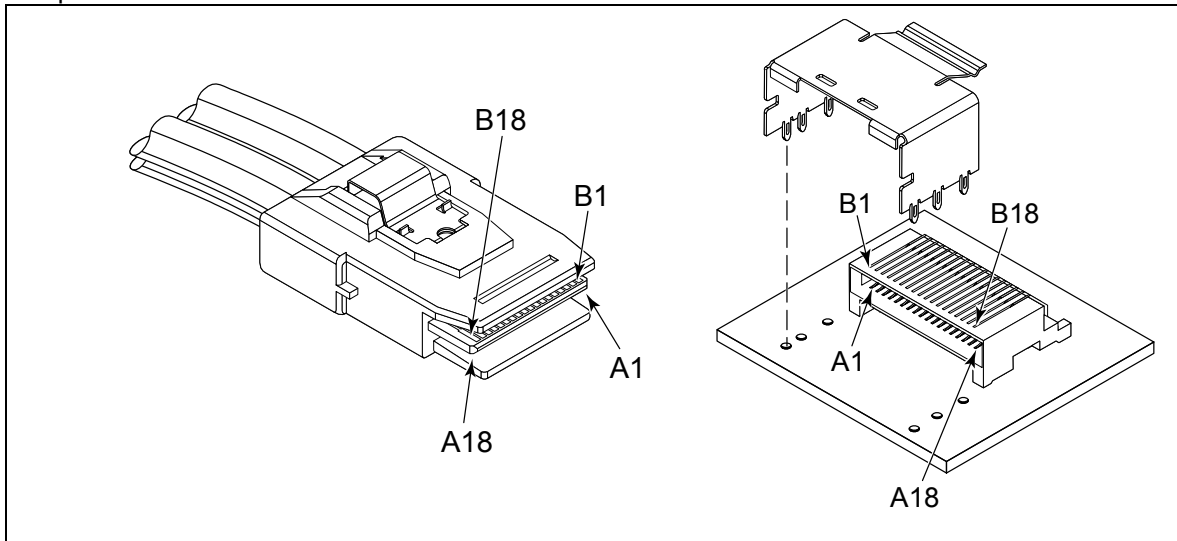


Figure 52 – Isometric drawings for Mini SATA Internal Multilane



### **6.2.12.3 Mini SATA Internal Multilane pin assignments**

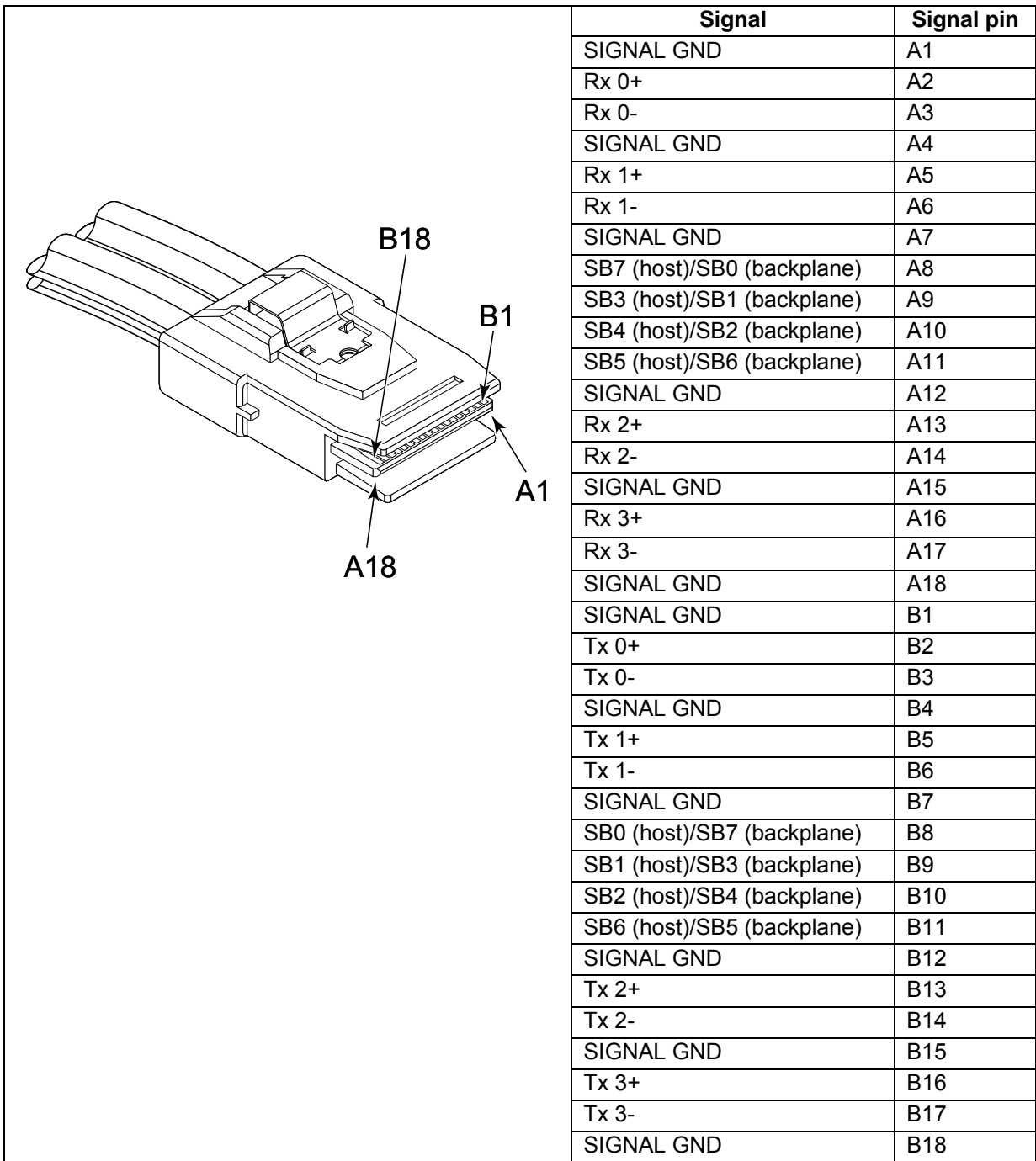
The Mini SATA Internal Multilane connector pin assignments are shown in Figure 53.

Pin assignments for sideband signals are based on the Internal Symmetrical Cable Assembly Implementation shown in Figure 54.

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 55 shows the Controller based fanout cable assembly.

Figure 56 shows the Backplane based fanout cable assembly



**Figure 53 – Mini SATA Internal Multilane connector pin assignments**

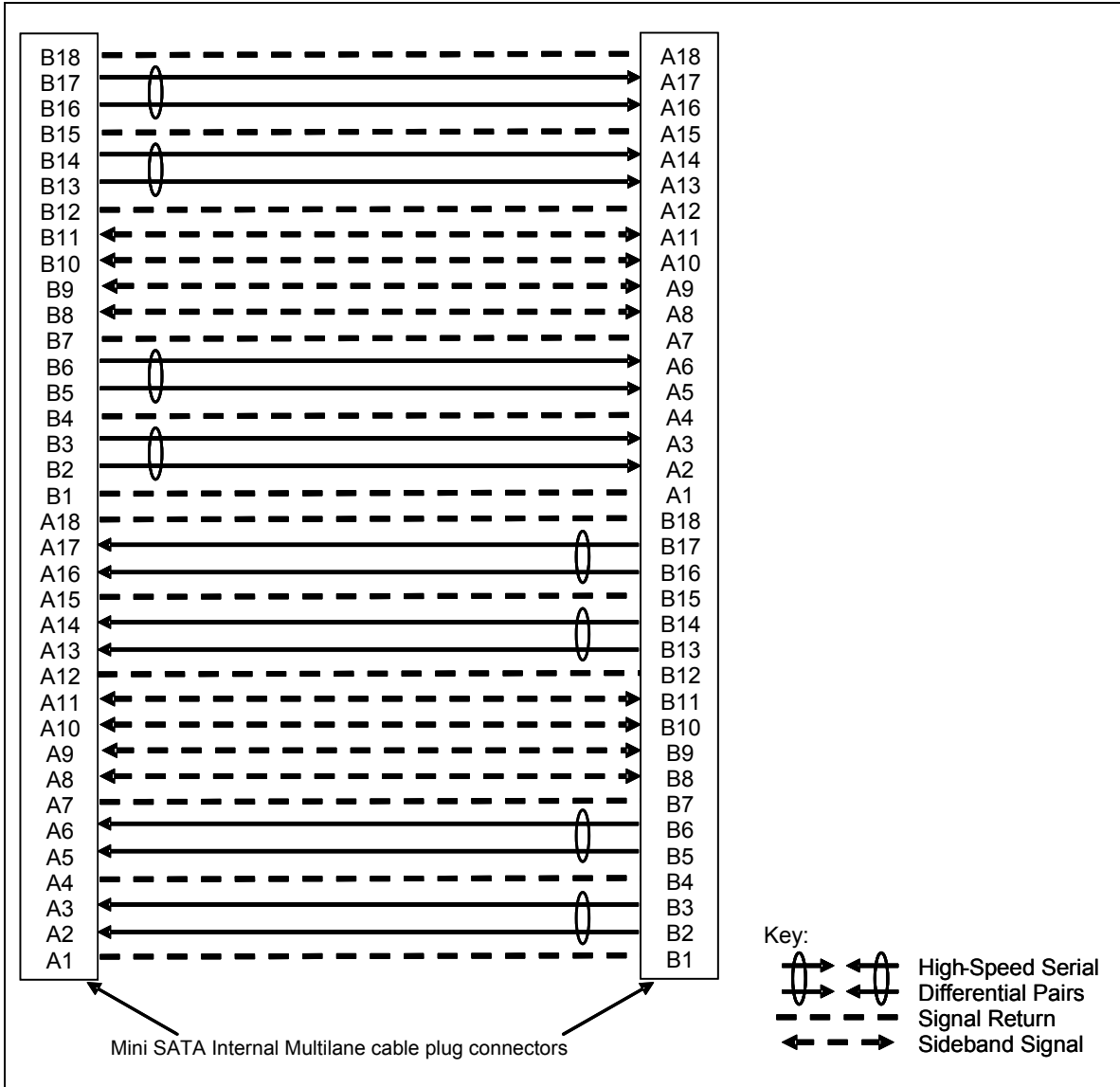
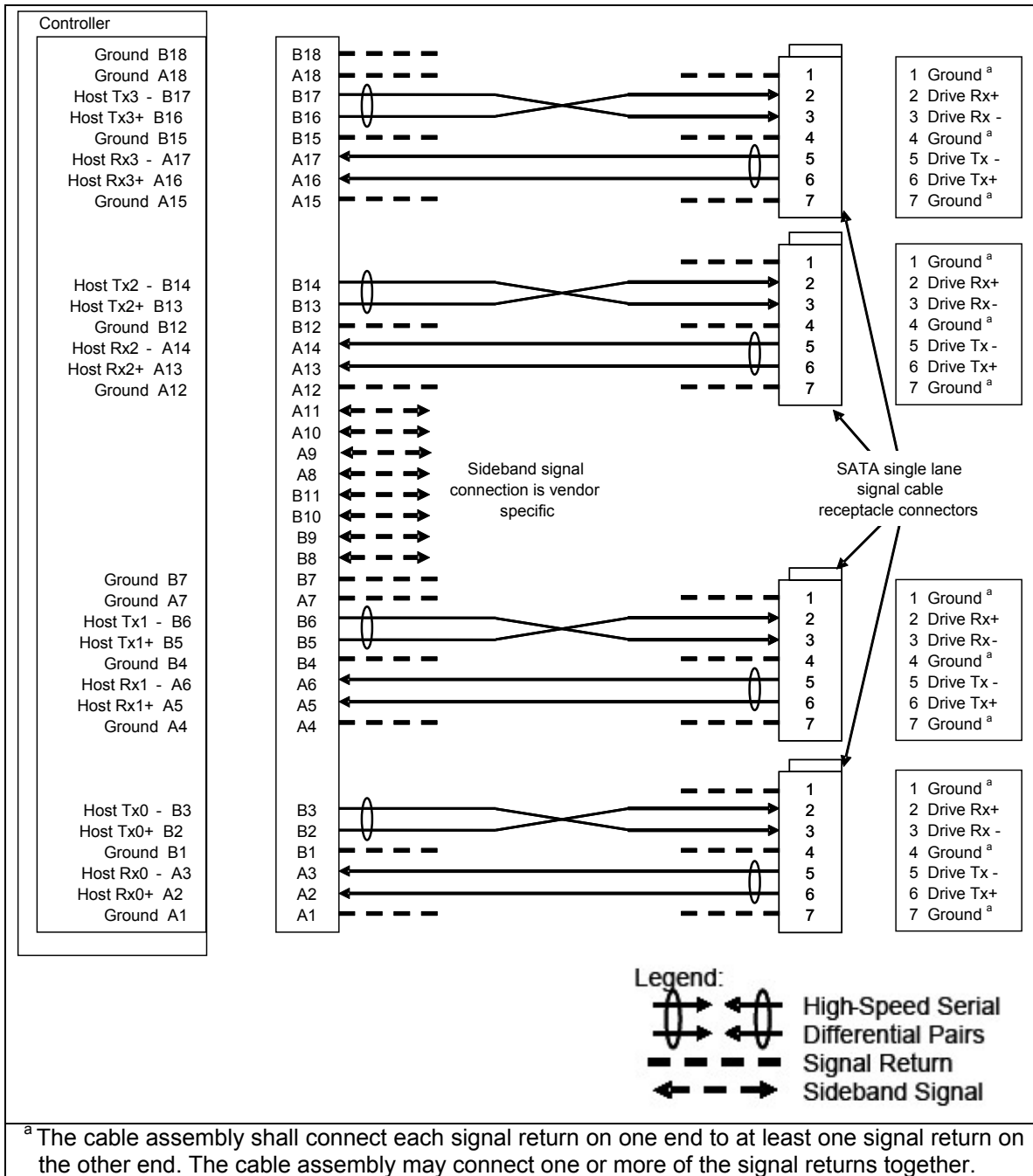
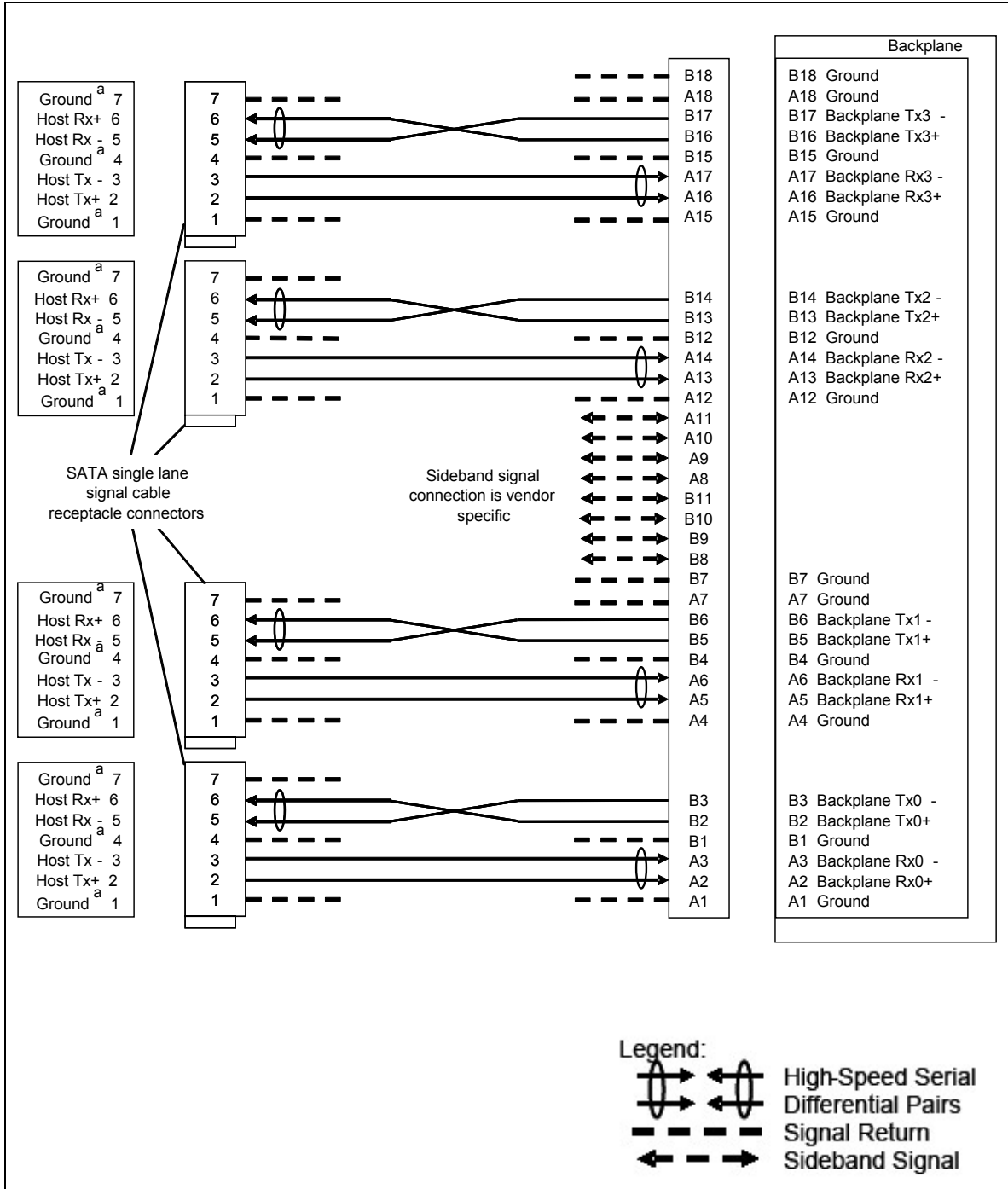


Figure 54 – Mini SATA Internal Multilane system, symmetric cable implementation



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

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization



<sup>a</sup> 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 56 – 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 Gbps), Gen2 (3.0 Gbps), and Gen3 (6.0 Gbps) 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 57 and Figure 58 for reference purposes. See SFF-8144 for form factor definition and connector location.

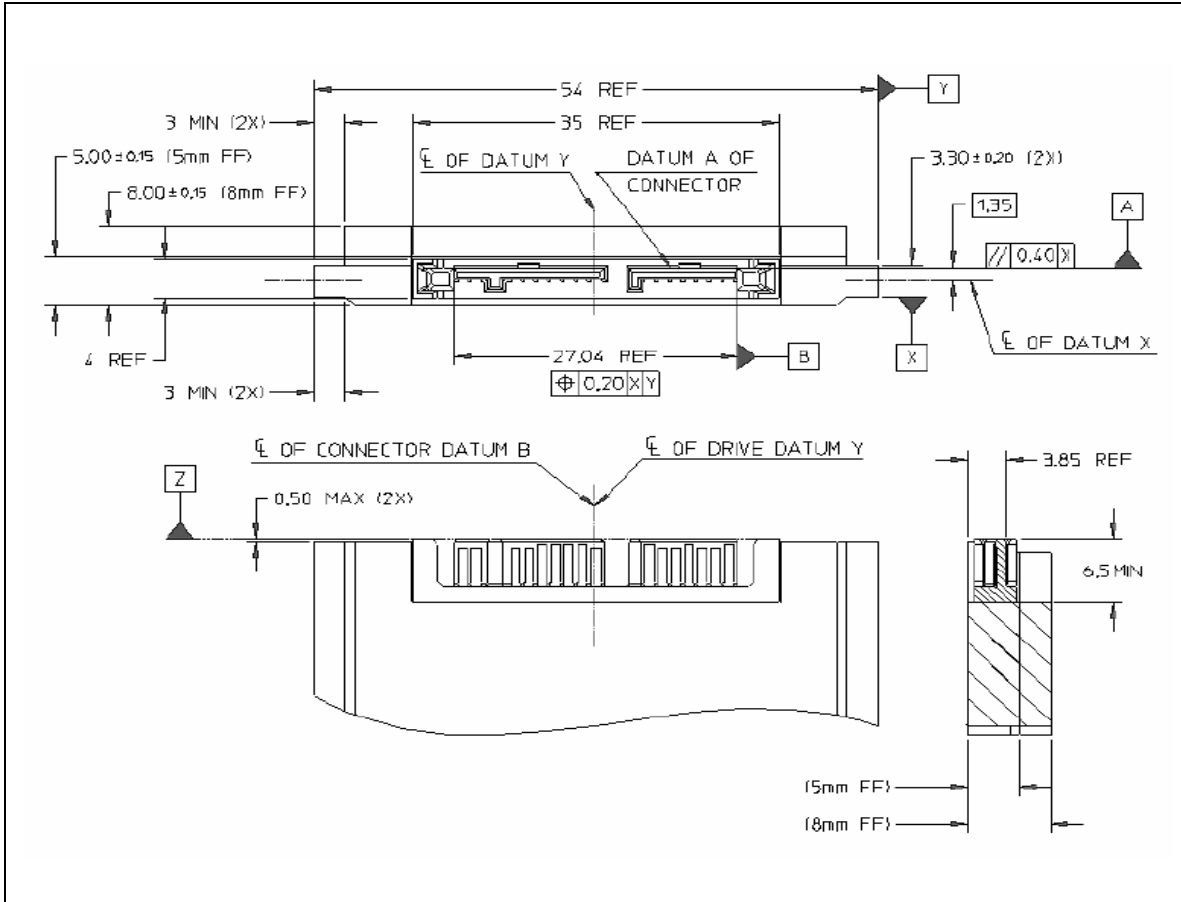
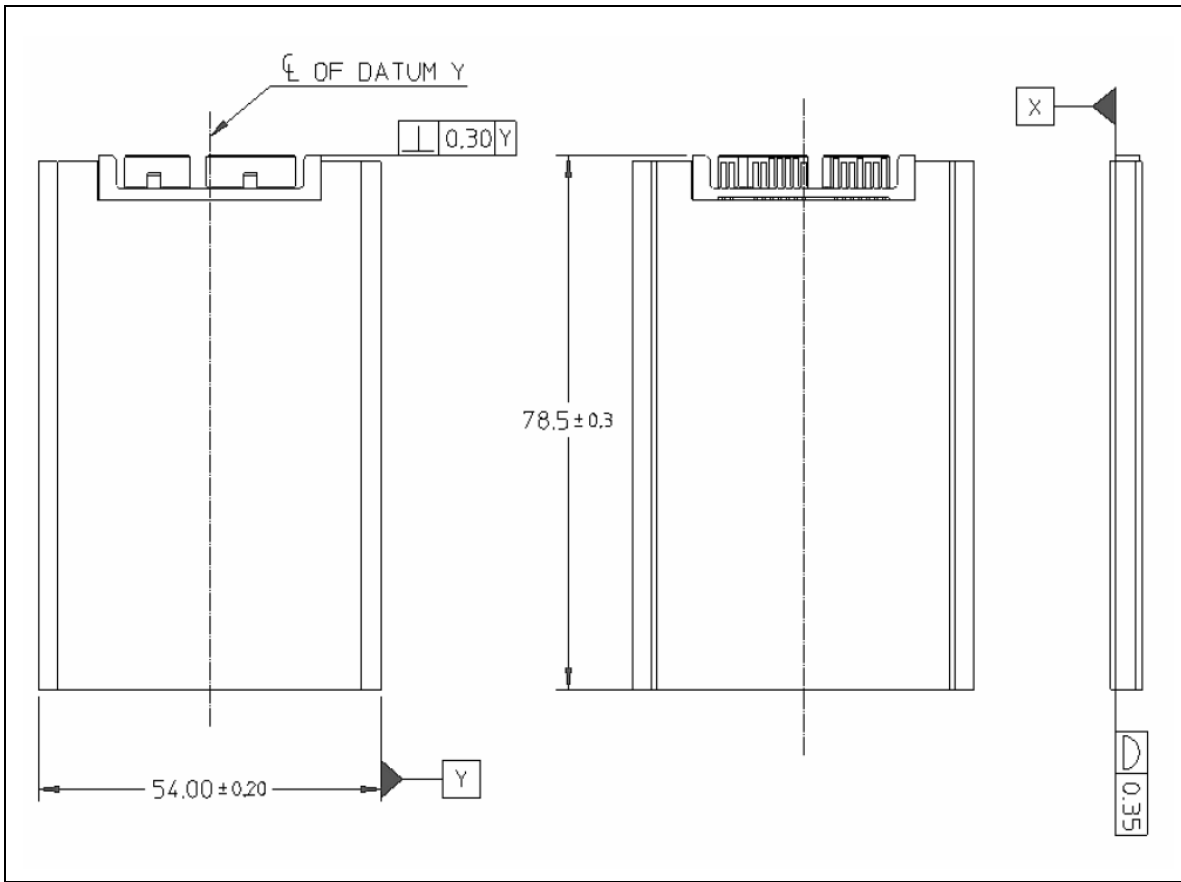


Figure 57 – Device internal Micro SATA connector location for 1.8 inch HDD



**Figure 58 – 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 59 defines the interface dimensions for the internal Micro SATA device plug connector with both signal and power segments.

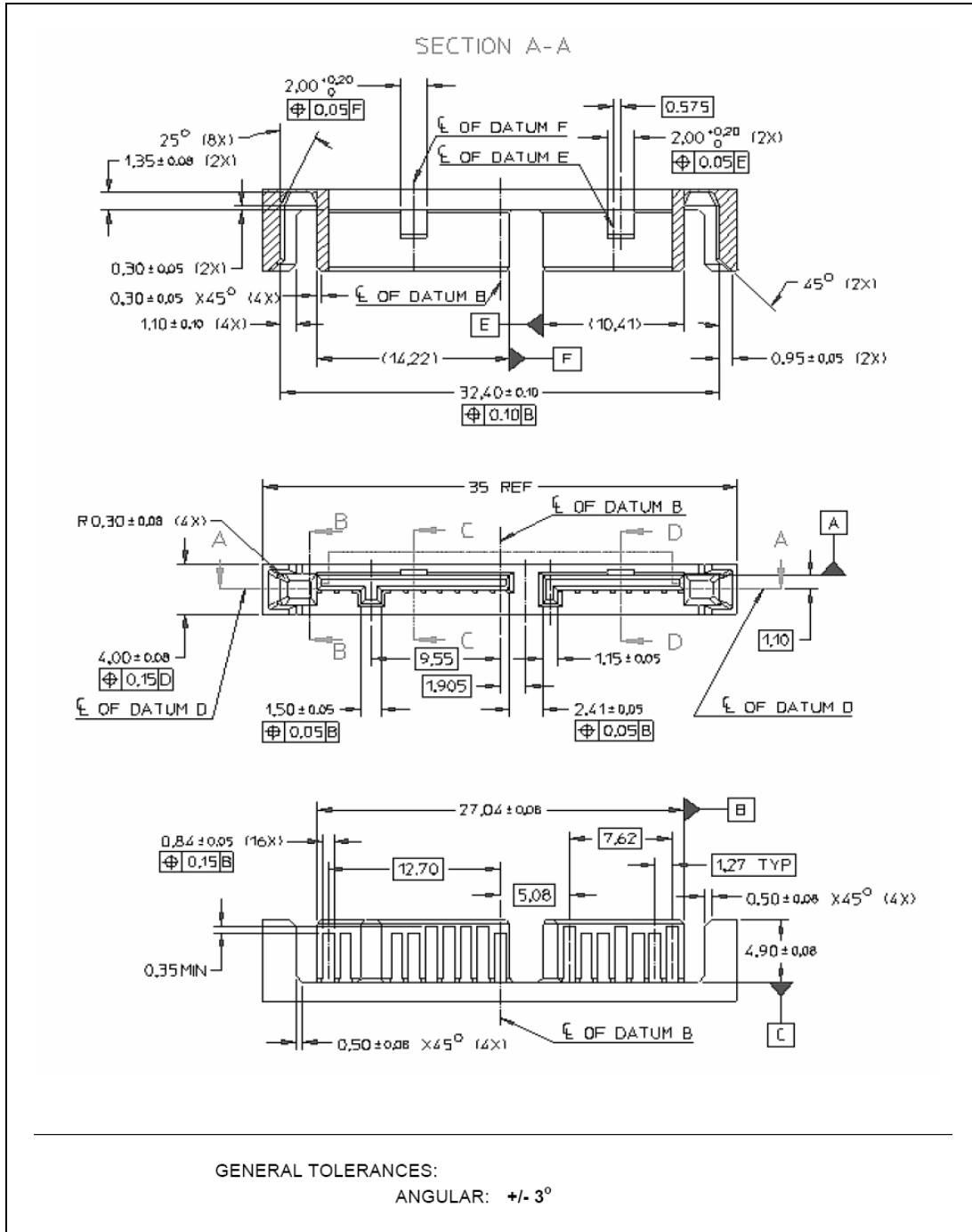


Figure 59 – Device internal Micro SATA plug connector

(part 1 of 2)

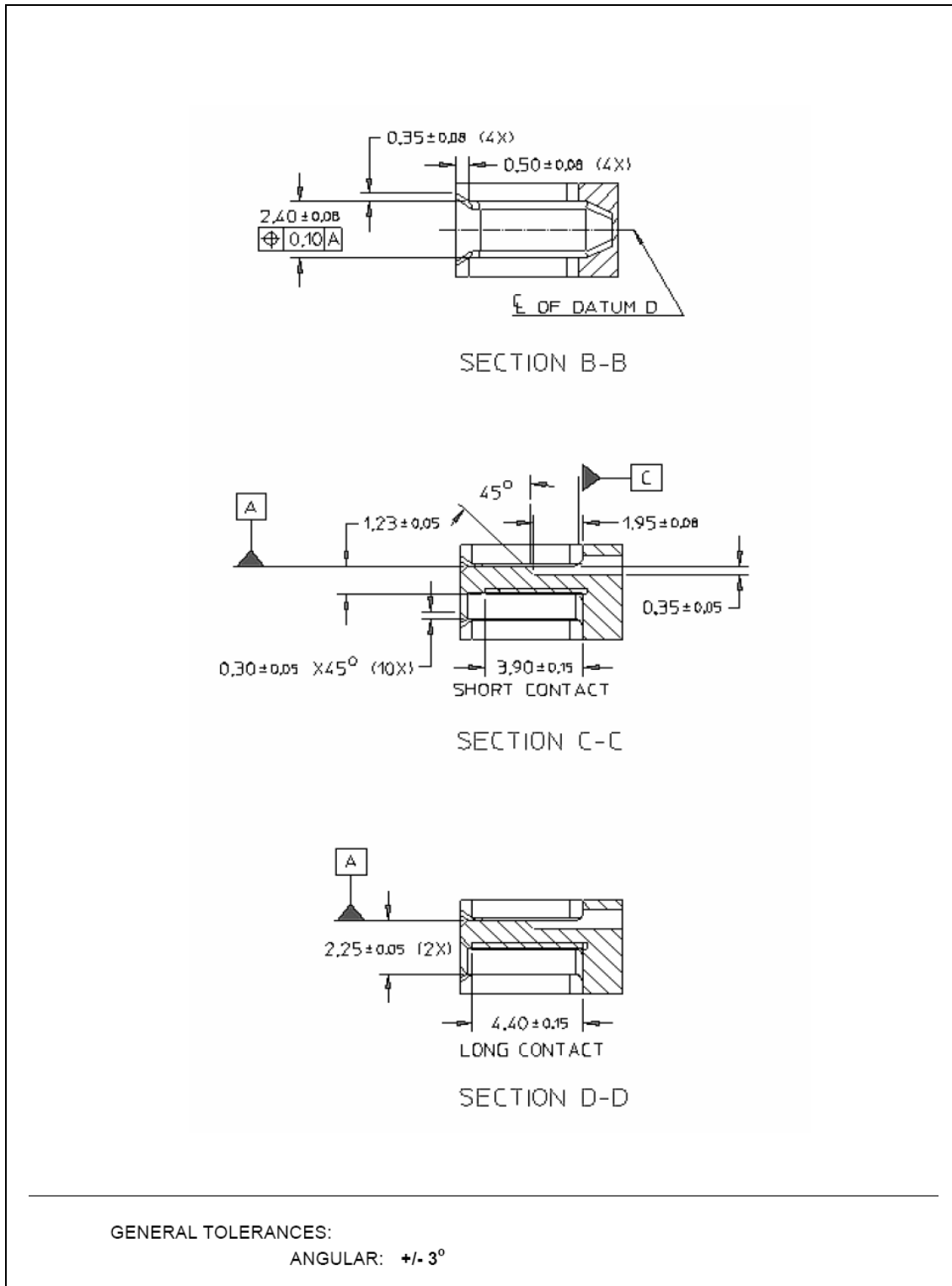


Figure 59 – Device internal Micro SATA plug connector (part 2 of 2)

### 6.3.5.2 Internal Micro SATA backplane connector

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

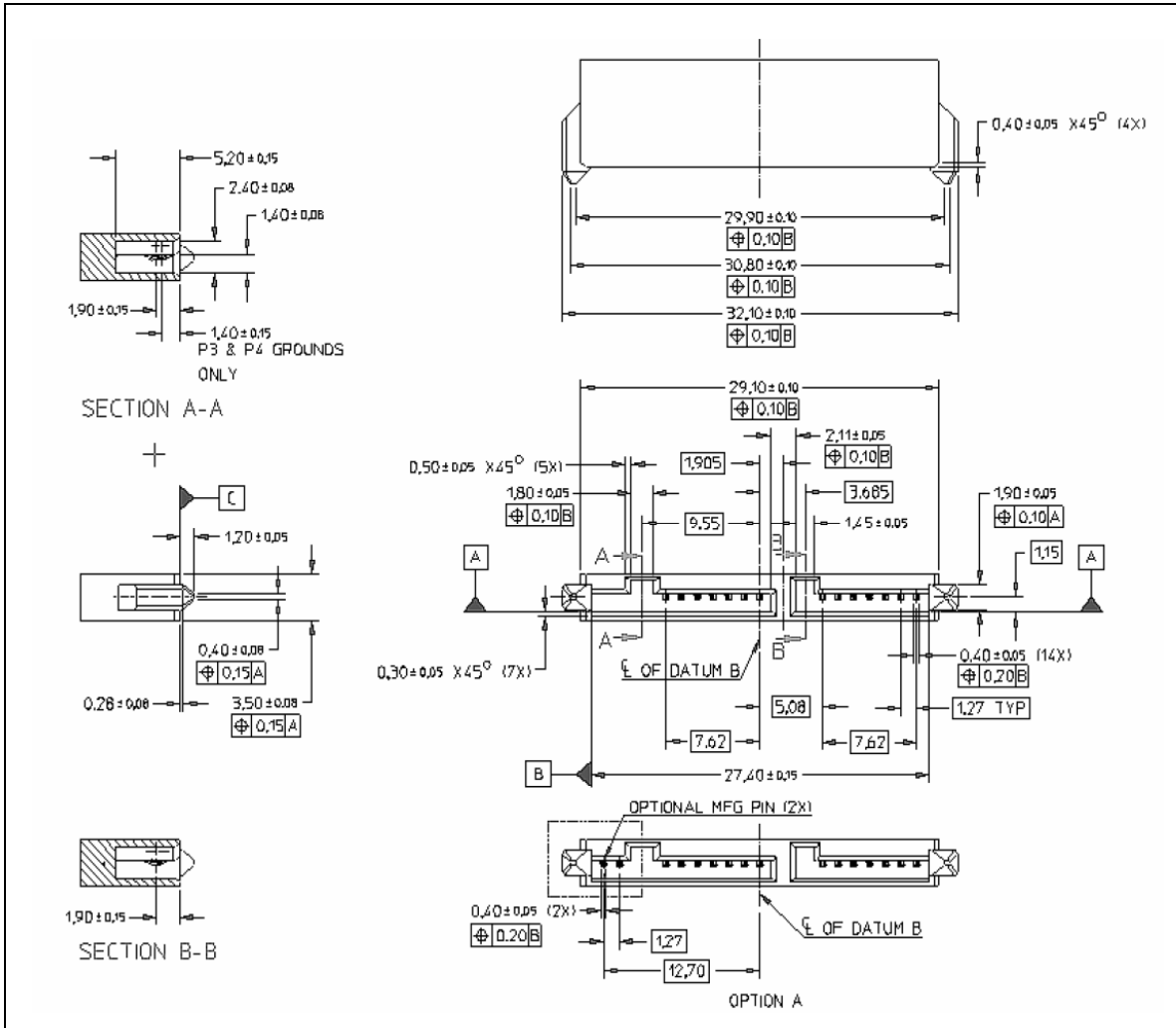


Figure 60 – Internal Micro SATA backplane connector

### 6.3.5.3 Internal Micro SATA power receptacle connector

Figure 61 defines the interface dimensions for the internal Micro SATA power receptacle connector.

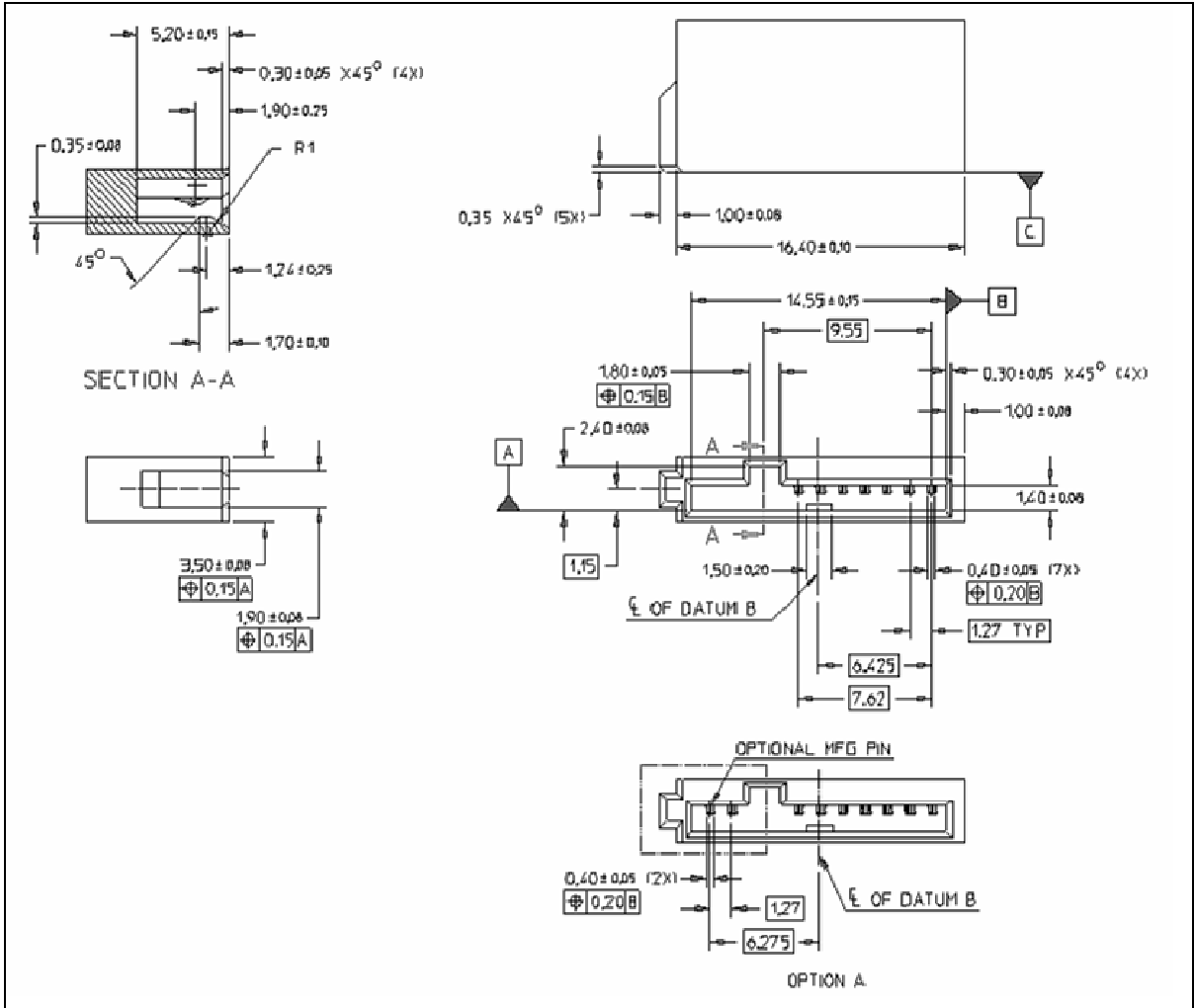


Figure 61 – Internal Micro SATA power receptacle connector

#### 6.3.5.4 Internal Micro SATA connector pair blind-mate misalignment capability

The maximum blind-mate misalignment capabilities are  $\pm 1.50$  mm and  $\pm 1.00$  mm, respectively, for two perpendicular axes illustrated in Figure 62. Any skew angle of the plug, with respect to the receptacle, reduces the blind-mate capabilities.

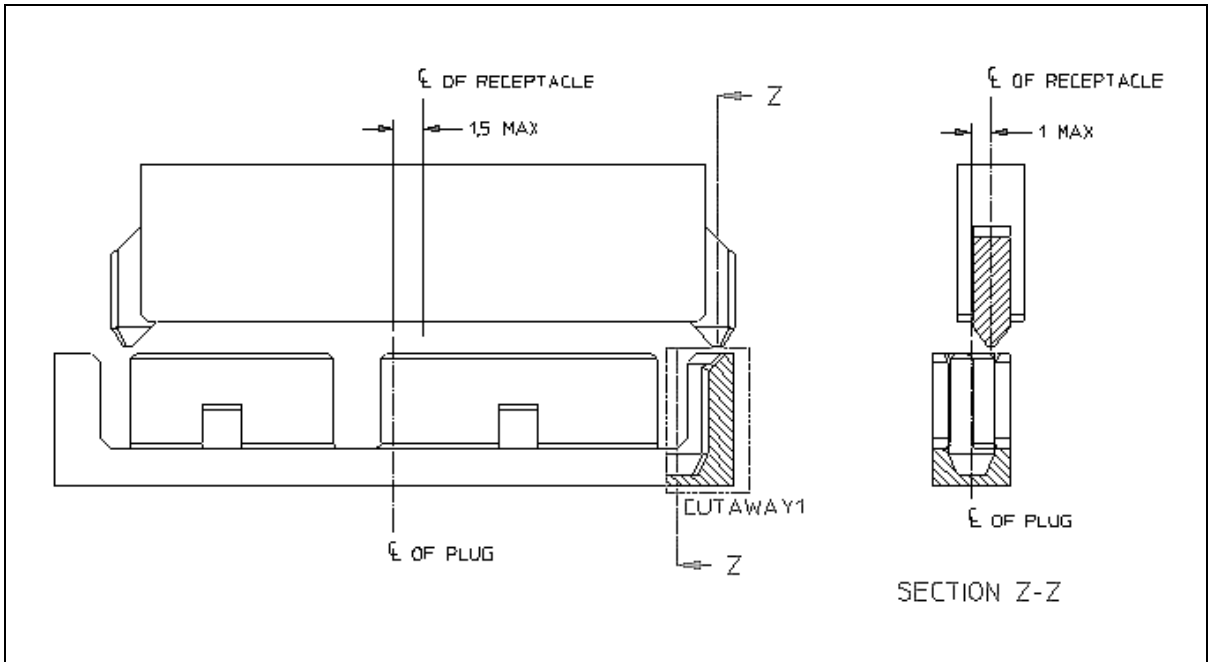


Figure 62 – Internal Micro SATA connector pair blind-mate misalignment capability

#### 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 <sup>a b</sup>	Backplane Usage <sup>b</sup>
Signal Segment	Refer to Table 5.				
	Spacing separate signal and power segments <sup>c</sup>				
Power Segment	P1	V <sub>33</sub>	3.3 V Power	2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
	P2	V <sub>33</sub>	3.3 V Power, Pre-charge	1 <sup>st</sup> Mate	2 <sup>nd</sup> Mate
	P3	GND	Ground	1 <sup>st</sup> Mate	1 <sup>st</sup> Mate
	P4	GND	Ground	1 <sup>st</sup> Mate	1 <sup>st</sup> Mate
	P5	V <sub>5</sub>	5 V Power, Pre-charge <sup>d</sup>	1 <sup>st</sup> Mate	2 <sup>nd</sup> Mate
	P6	V <sub>5</sub>	5 V Power <sup>d</sup>	2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
	P7	DAS/DHU	DAS/Direct Head Unload / Vendor Specific <sup>e</sup>	2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
	Key	Key	Key	N/C	N/C
	P8	Optional	Vendor specific <sup>f</sup>	2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
	P9	Optional	Vendor specific <sup>f</sup>	2 <sup>nd</sup> Mate	3 <sup>rd</sup> Mate
<sup>a</sup> Although the mate order is shown, hot plugging is not supported if using the cable connector receptacle. <sup>b</sup> All mate sequences assume zero angular offset between connectors. <sup>c</sup> The signal segment and power segment may be separate. <sup>d</sup> 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. <sup>e</sup> For specific optional usage of pin P7, see 6.13. <sup>f</sup> 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**

Removal force Backplane connector	EIA-364-13 Measure the force necessary to unmate the connector assemblies at a max. rate of 12.5 mm/min.	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 mm/min.	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 mm/min.	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 Gbps), Gen2 (3.0 Gbps), and Gen3 (6.0 Gbps) 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 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.
- 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 drive 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 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  $\pm 10\%$ . 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 13.2.2.18).

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  $\pm 10\%$  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  $\pm 10\%$ , 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  $\pm 10\%$ , 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  $\pm 10\%$ .

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  $\pm 10\%$ , 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 Device Attention Supported bit in the IDENTIFY PACKET DEVICE, 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**

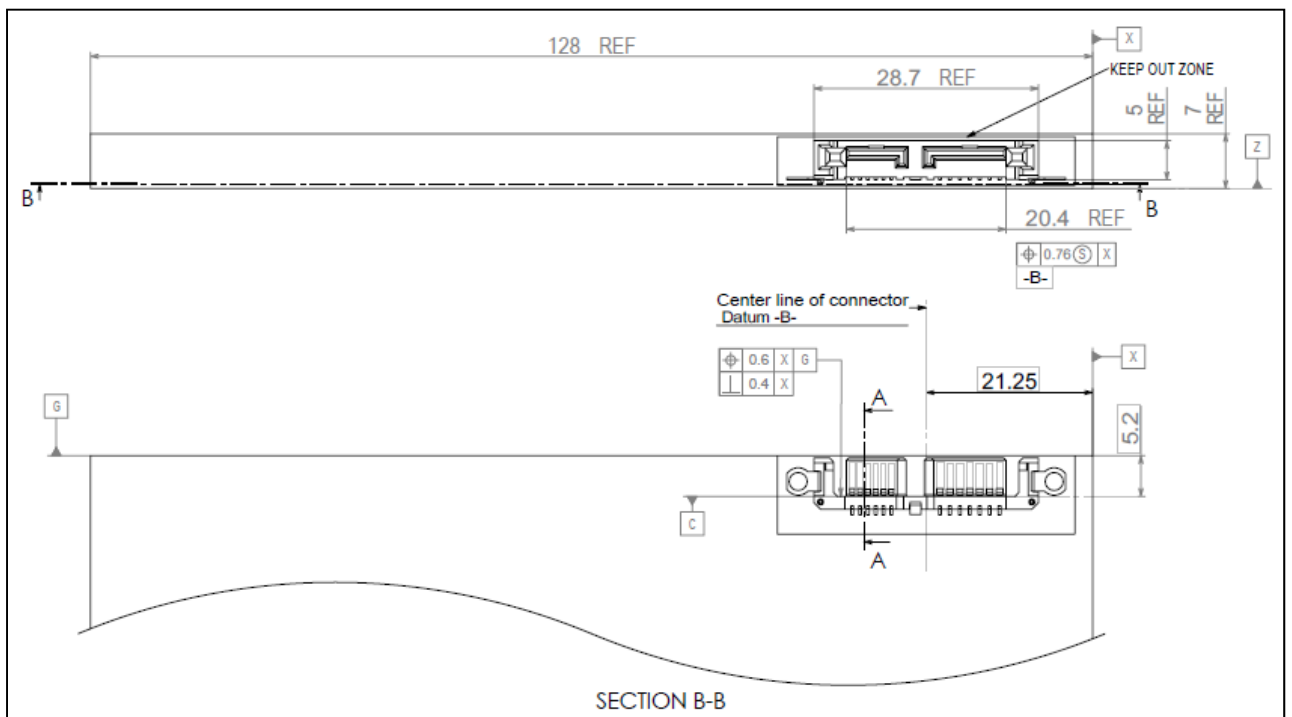
<b>Slimline Drive</b>	<b>Horizontal</b>	<b>Vertical</b>	<b>Depth</b>
7.0 mm Slimline	Drive left edge to connector CL	Drive bottom edge to connector tongue top edge	Drive back edge to connector back wall
8.5 mm Slimline	Drive left edge to connector CL	Drive bottom edge to connector tongue top edge	Drive 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 mm Slimline optical disk drive (ODD) is shown in Figure 63 for reference purposes only. Additional details may be found in SFF-8553.



**Figure 63 – 7.0 mm Slimline drive connector locations**

### 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 64 for reference purposes only. Additional details may be found in SFF-8553.

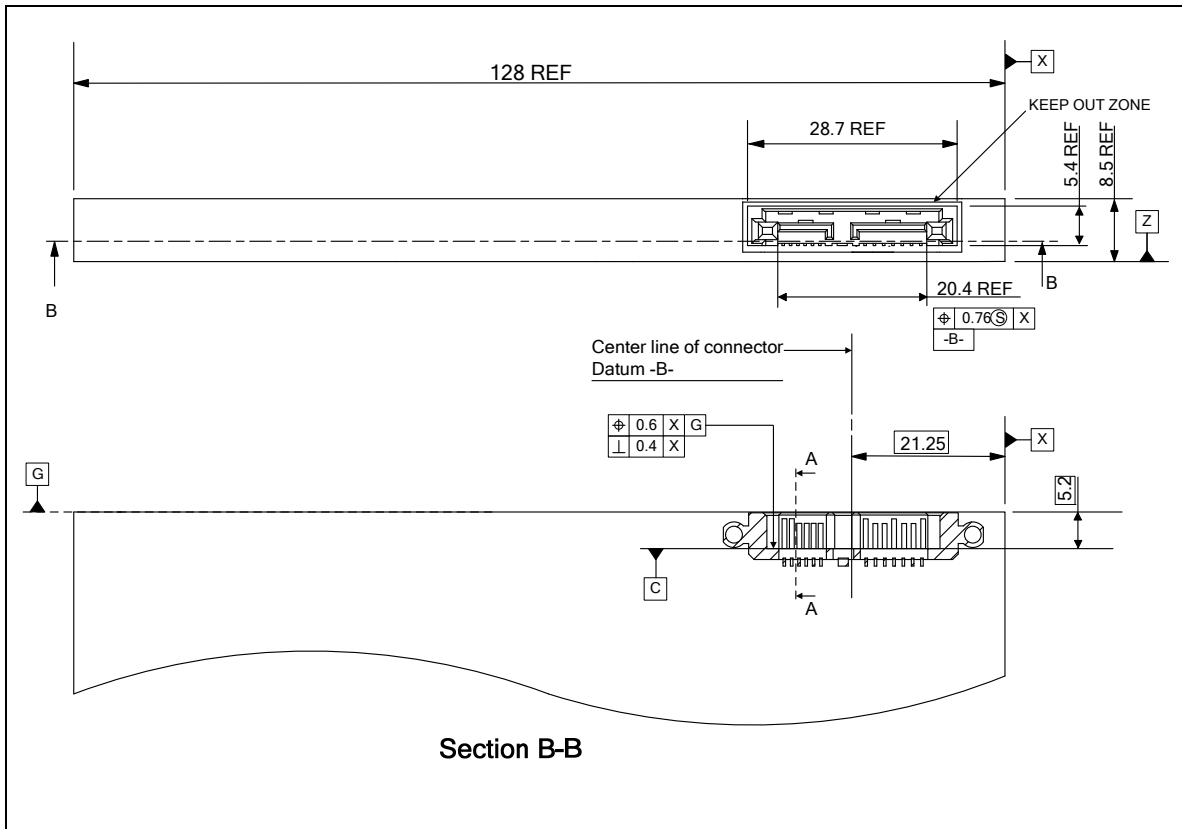


Figure 64 – 8.5 mm Slimline drive connector locations (informative)

### 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 65 for reference purposes only. Additional details may be found in SFF-9553.

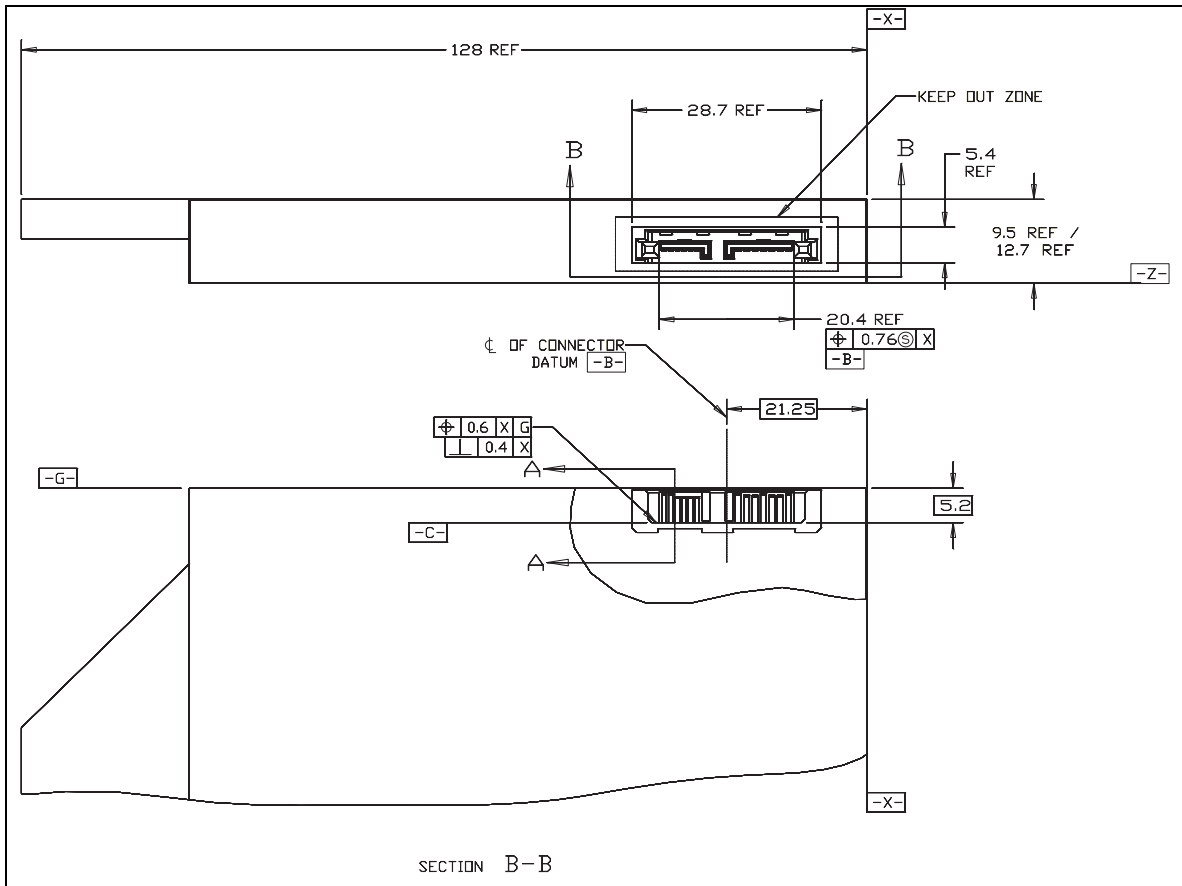


Figure 65 – 9.5 mm/12.7 mm Slimline drive connector locations

**6.4.4.3.4 7.0 mm Slimline Devices**

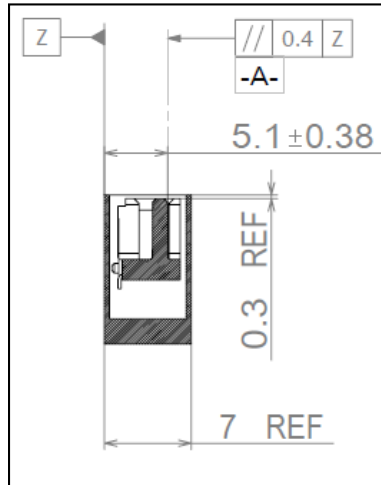


Figure 66 – 7.0 mm Slimline drive connector location (section A-A)

**6.4.4.3.5 8.5 mm Slimline drives**

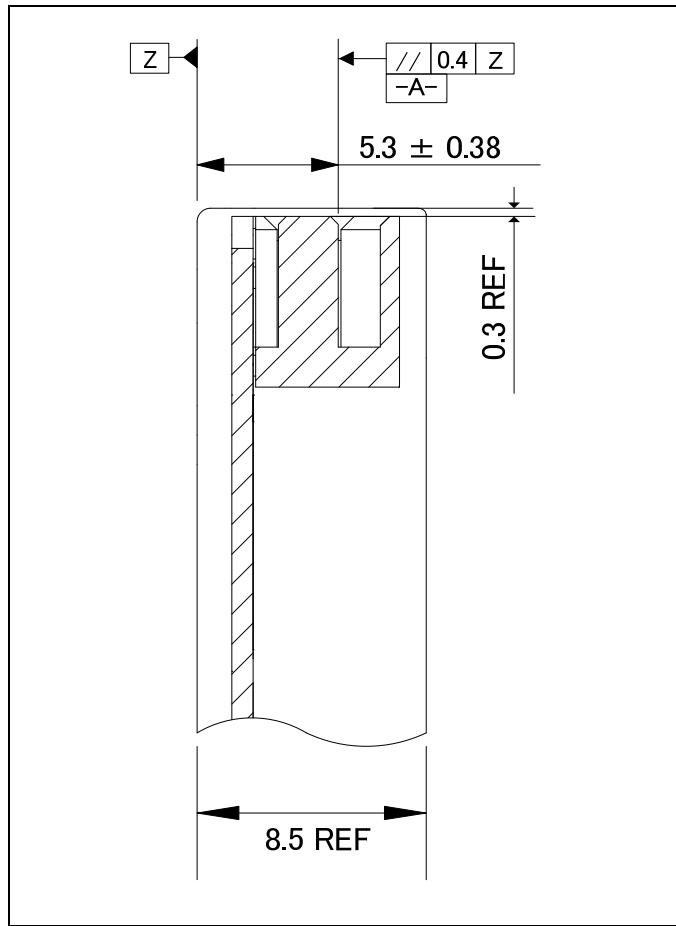


Figure 67 – 8.5 mm Slimline drive connector location (section A-A)

6.4.4.3.6 9.5 mm Slimline drives

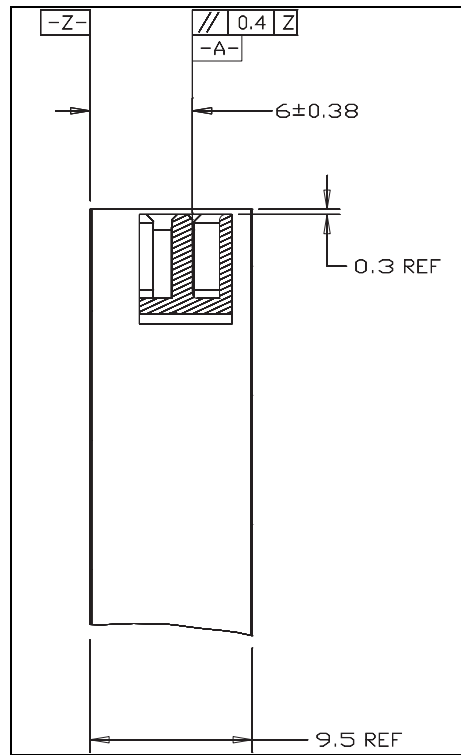


Figure 68 – 9.5 mm Slimline drive connector location (section A-A)

6.4.4.3.7 12.7 mm Slimline drives

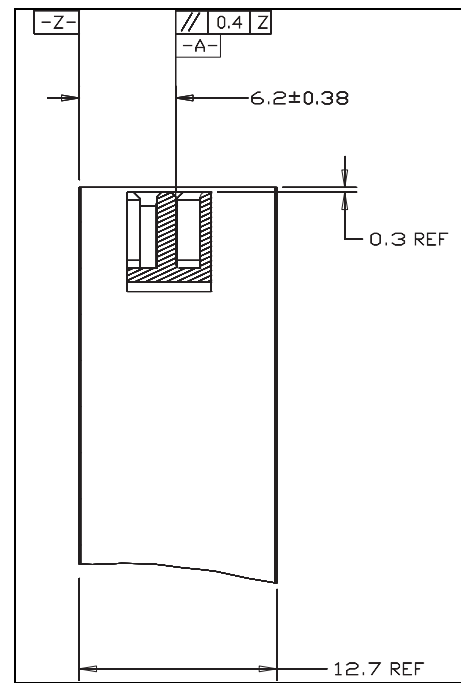


Figure 69 – 12.7 mm Slimline drive connector location (section A-A)

## **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 mm Slimline device plug connector

### 6.4.5.2.1 7 mm Slimline device plug dimensions

This is the connector used in the 7 mm Slimline device. Figure 70 to Figure 74 show the interface dimensions for the device plug connector with both signal and power segments. General tolerances on these drawings are  $\pm 0.20$  mm.

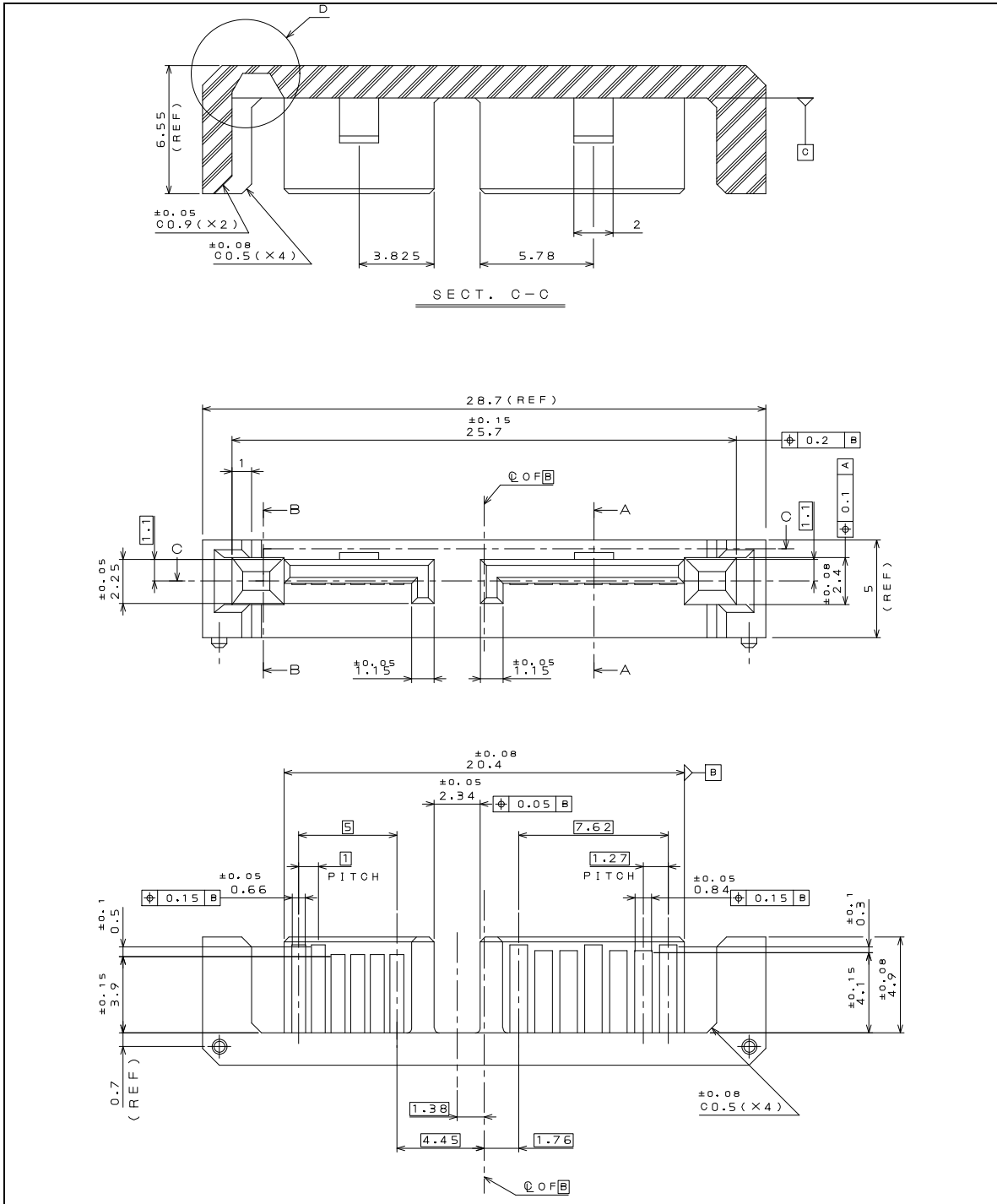
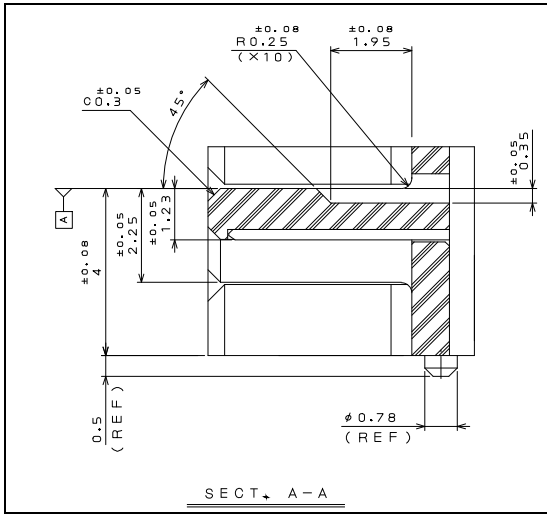
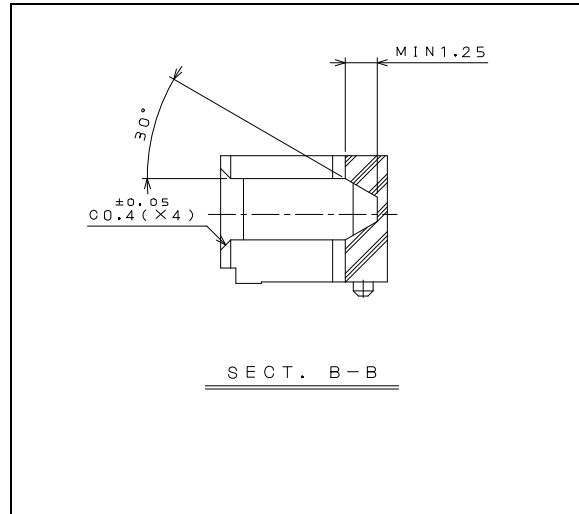


Figure 70 – 7 mm Slimline device plug connector interface dimensions

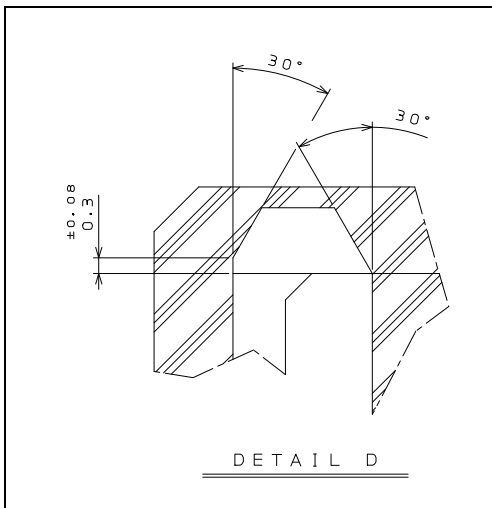




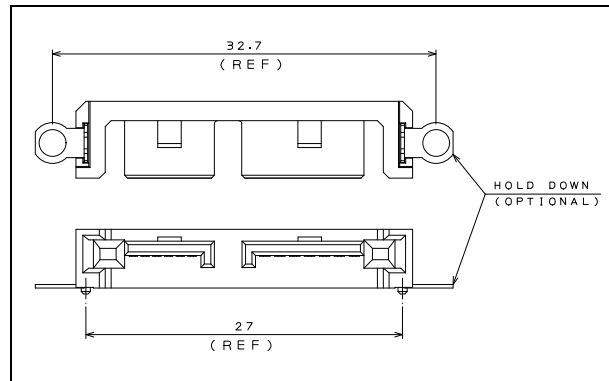
**Figure 71 – 7.0 mm Slimline device plug connector interface dimensions section A-A**



**Figure 72 – 7.0 mm Slimline device plug connector interface dimensions section B-B**



**Figure 73 – 7.0 mm Slimline device plug connector interface dimensions detail D**



**Figure 74 – 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 75 and Table 15 show the interface dimensions for the device plug connector with both signal and power segments. General tolerances on these drawings is  $\pm 0.20$  mm.

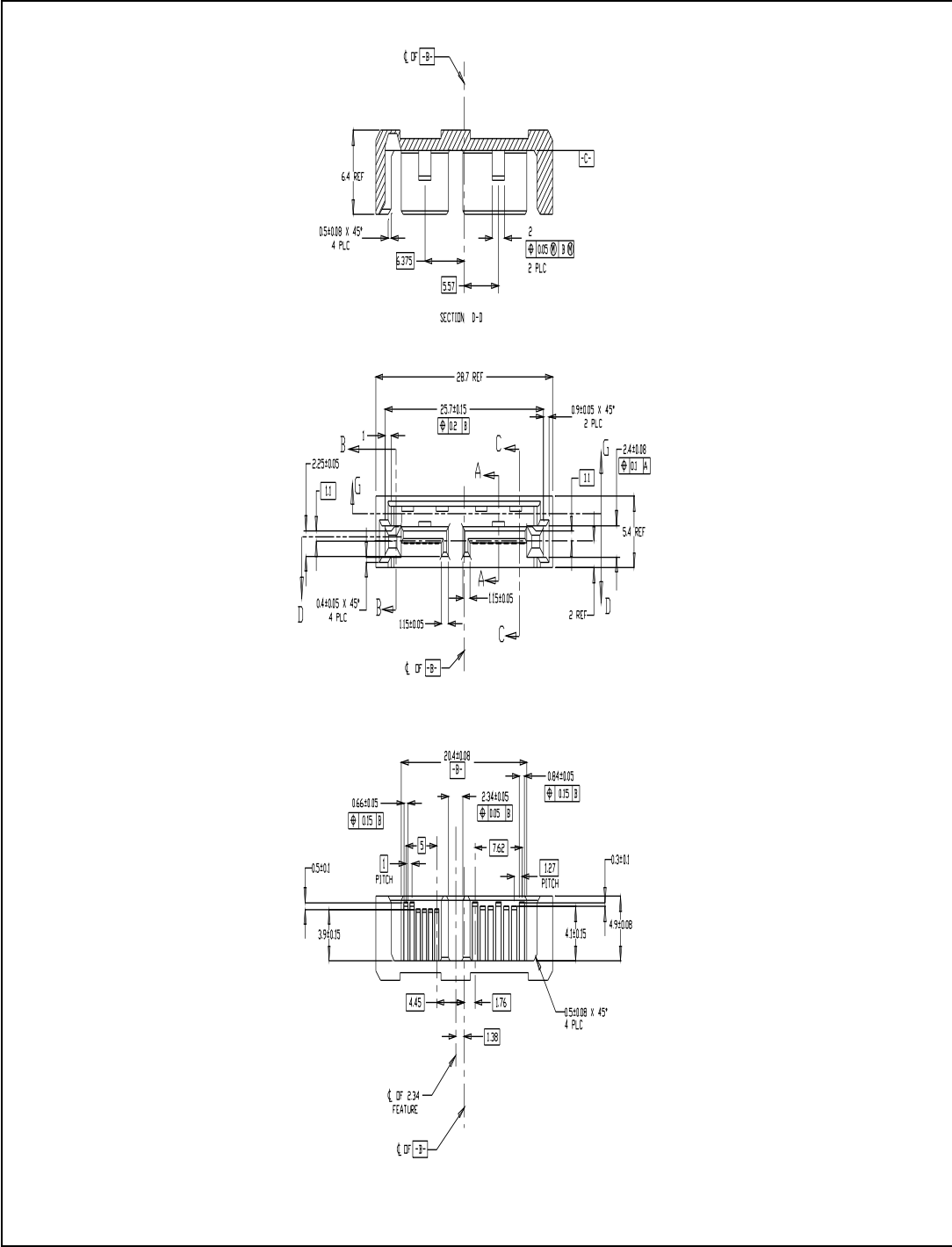
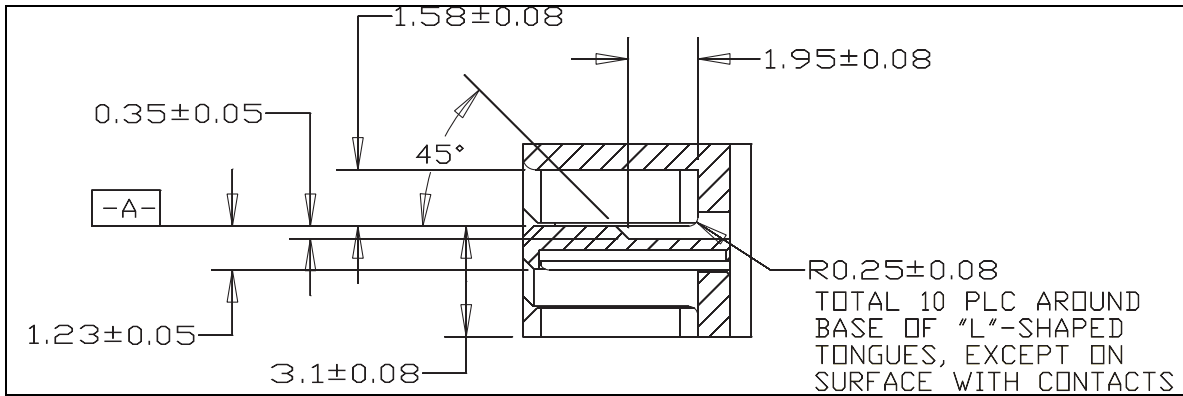
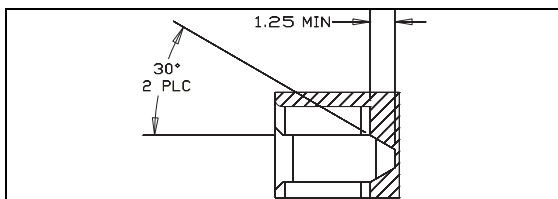


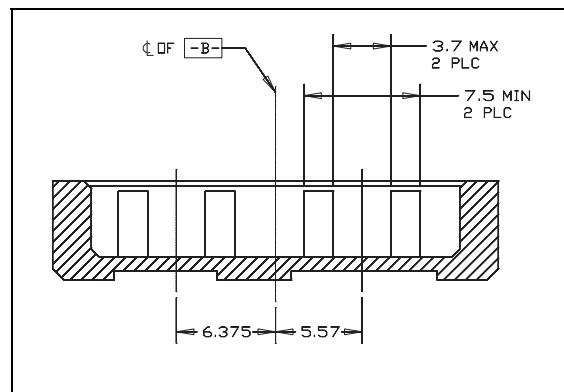
Figure 75 – Slimline device plug connector interface dimensions



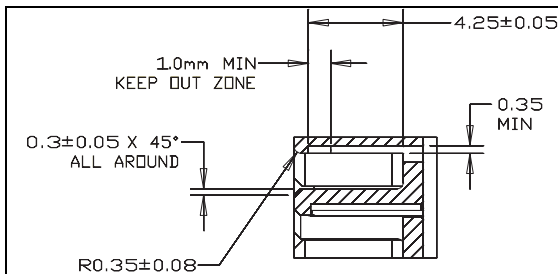
**Figure 76 – Slimline device plug connector interface dimensions (section A-A)**



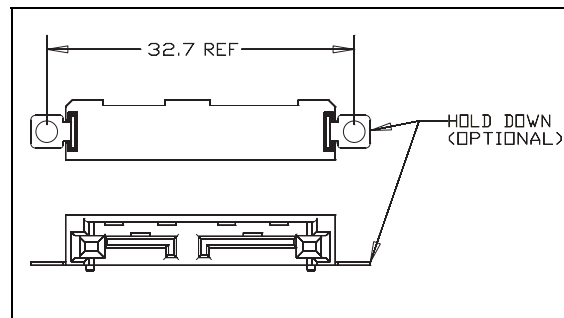
**Figure 77 – Slimline device plug connector interface dimensions (section B-B)**



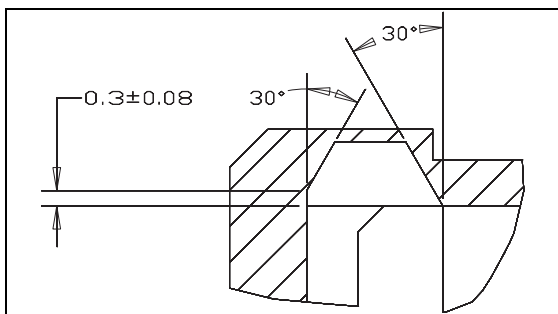
**Figure 80 – Slimline device plug connector interface dimensions (section G-G)**



**Figure 78 – Slimline device plug connector interface dimensions (section C-C)**



**Figure 81 – Slimline device plug connector optional hold down mounting**

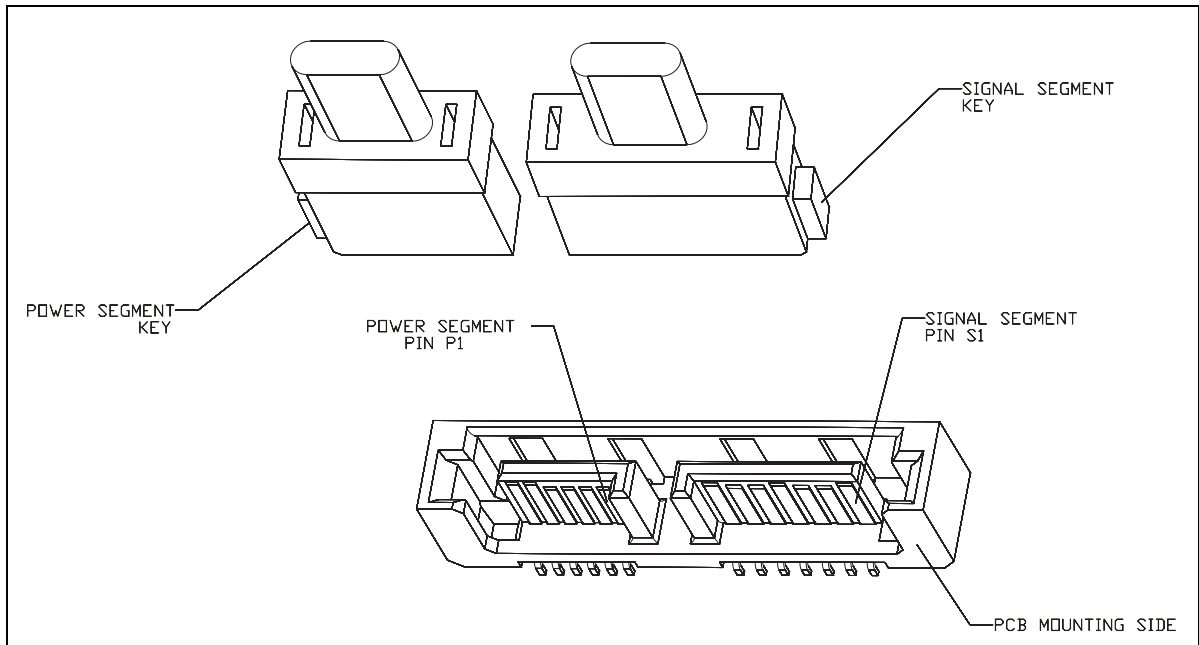


**Figure 79 – Slimline device plug connector interface dimensions (detail F)**

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

**Table 15 – Slimline device plug connector pin definition**

	Name	Type	Description	Cable Usage <sup>a b</sup>	Backplane Usage <sup>b</sup>
Signal Segment	Refer to Table 5.				
	Signal Segment "L"				
	Central Connector Gap				
	Power Segment "L"				
Power Segment	P1	DP	Device Present	3 <sup>rd</sup> mate	3 <sup>rd</sup> mate
	P2	+5 V <sup>e</sup>		2 <sup>nd</sup> mate	2 <sup>nd</sup> mate
	P3	+5 V <sup>e</sup>		2 <sup>nd</sup> mate	2 <sup>nd</sup> mate
	P4	MD/DA	Manufacturing Diagnostic/Device Attention <sup>d</sup>	2 <sup>nd</sup> mate	2 <sup>nd</sup> mate
	P5	Gnd <sup>c</sup>	Ground	1 <sup>st</sup> mate	1 <sup>st</sup> mate
	P6	Gnd <sup>c</sup>	Ground	1 <sup>st</sup> mate	1 <sup>st</sup> mate
Power Segment Key					
<p><sup>a</sup> All pins are in a single row with 1.00 mm (0.039 inch) pitch on the power segment portion.</p> <p><sup>b</sup> 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.</p> <p><sup>c</sup> The connection between the Serial ATA Slimline device signal ground and power ground is vendor specific.</p> <p><sup>d</sup> The DP and MD/DA signals shall be referenced to the power portion ground pins, P5 and P6.</p> <p><sup>e</sup> 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.</p>					



**Figure 82 – 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 83 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.

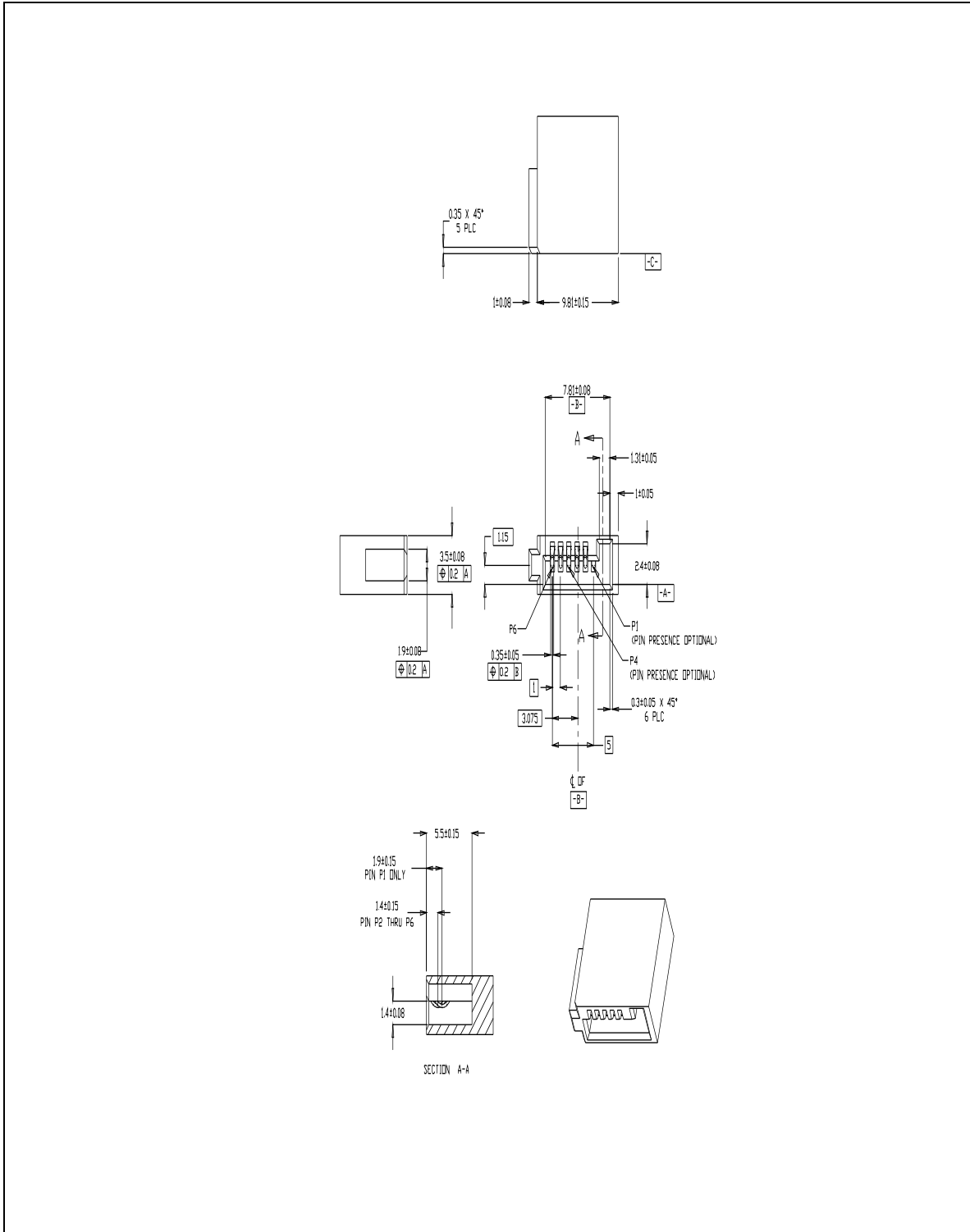
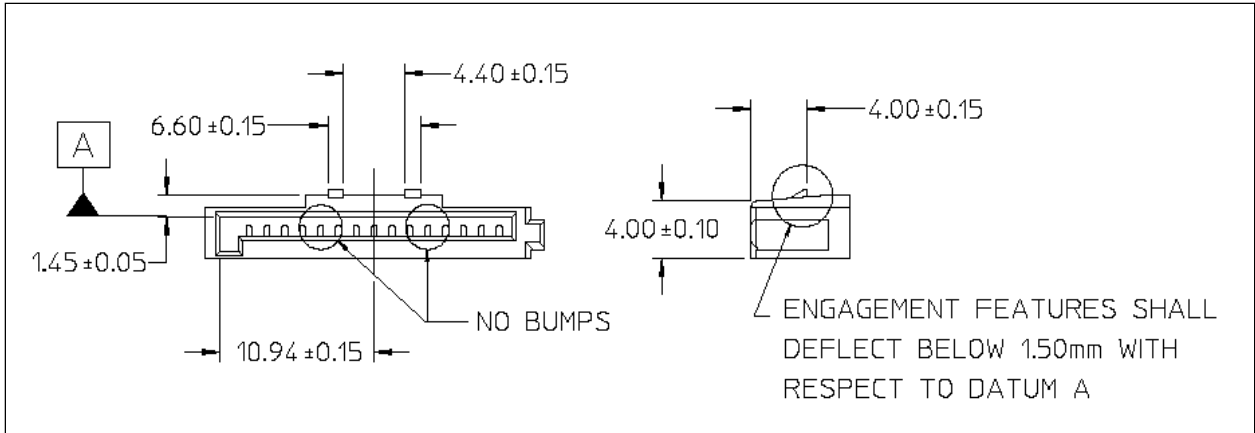
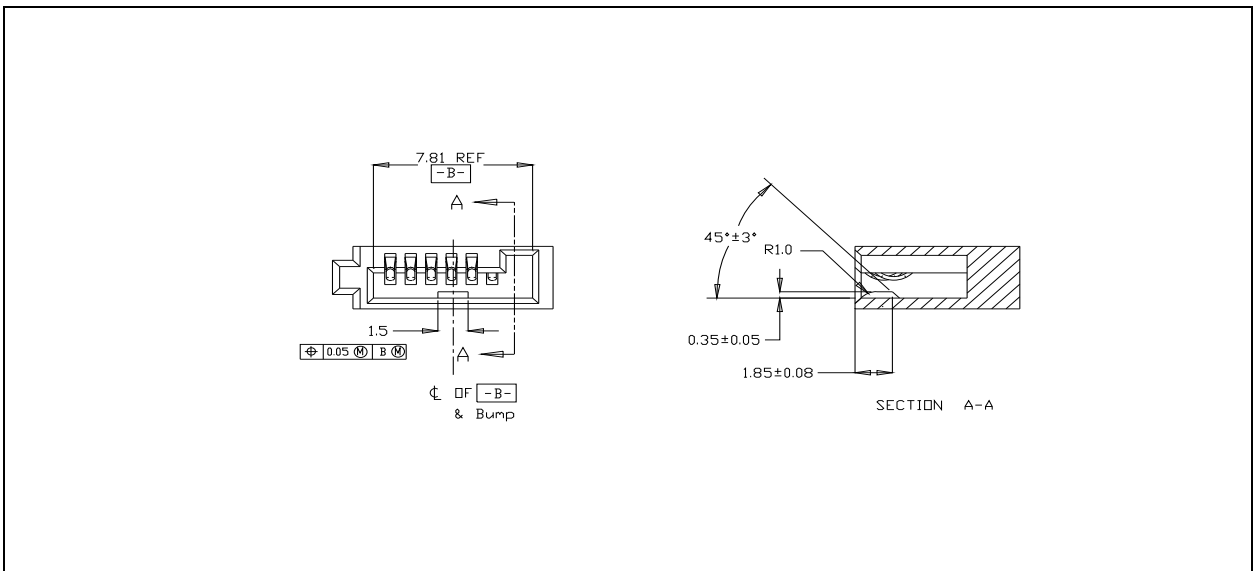


Figure 83 – Slimline power receptacle connector interface dimensions



**Figure 84 – Slimline power receptacle connector option with latch**

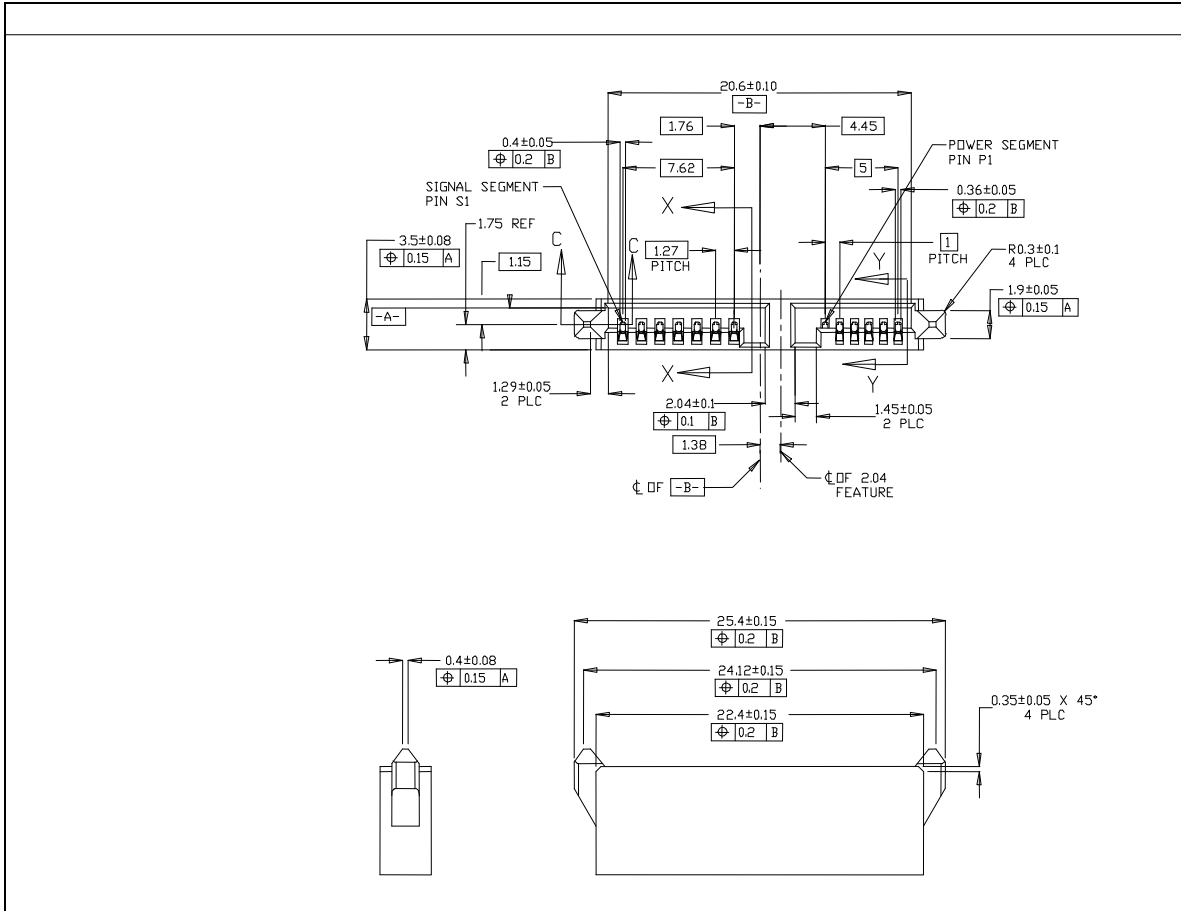


**Figure 85 – Slimline power receptacle connector option with bump**

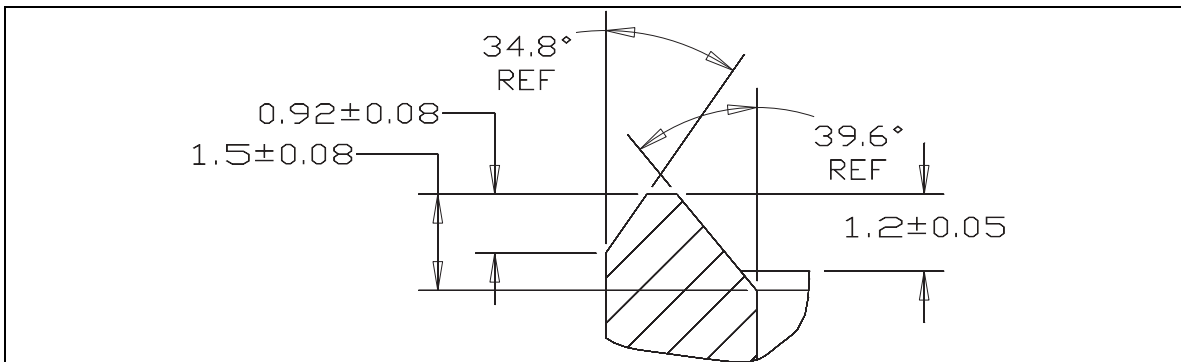


**6.4.5.4 Slimline host receptacle connector**

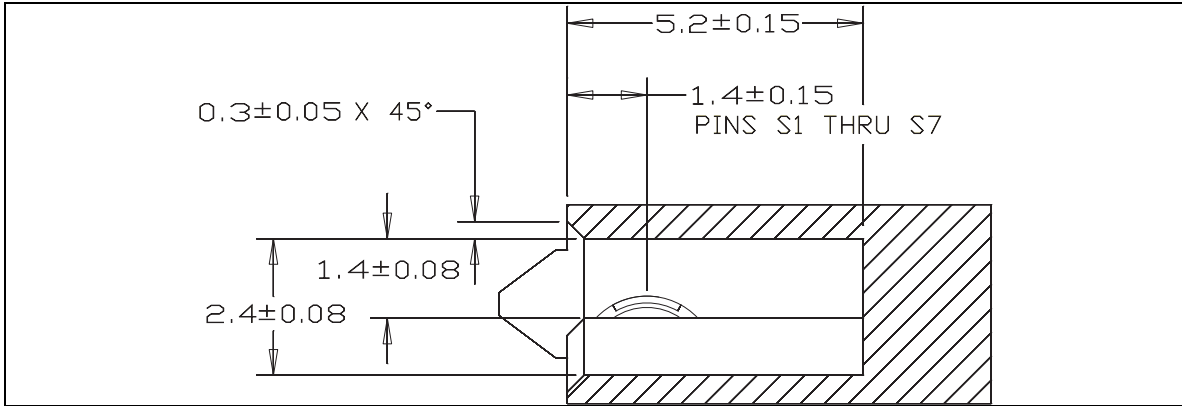
The Slimline host receptacle connector is to be blind-mated directly with the device plug connector. The interface dimensions for the host receptacle connector are shown in Figure 86. An appropriate external retention mechanism independent of the connector is required to keep the host PCB and the device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



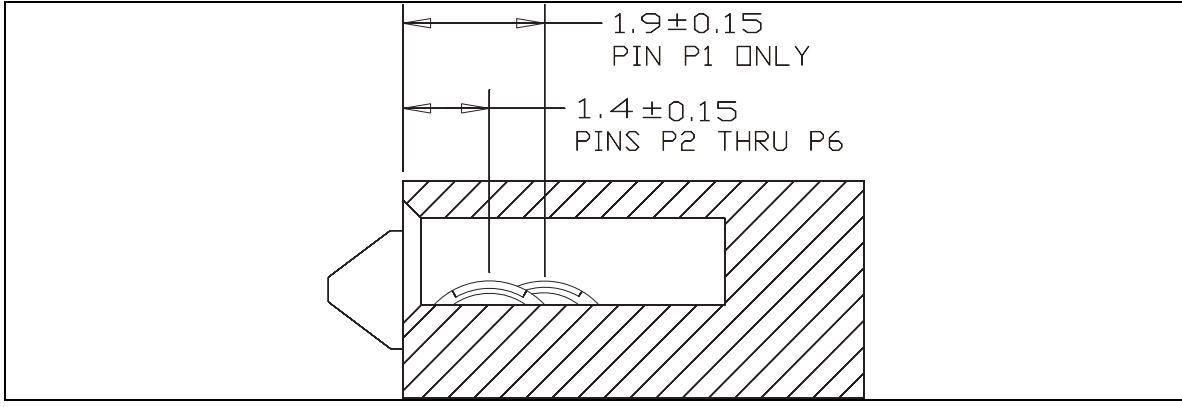
**Figure 86 – Slimline host receptacle connector interface dimensions**



**Figure 87 – Slimline host receptacle connector interface dimensions section C-C**



**Figure 88 – Slimline host receptacle connector interface dimensions section X-X**



**Figure 89 – 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 90. Any skew angle of the plug, with respect to the receptacle, reduces the blind-mate tolerances.

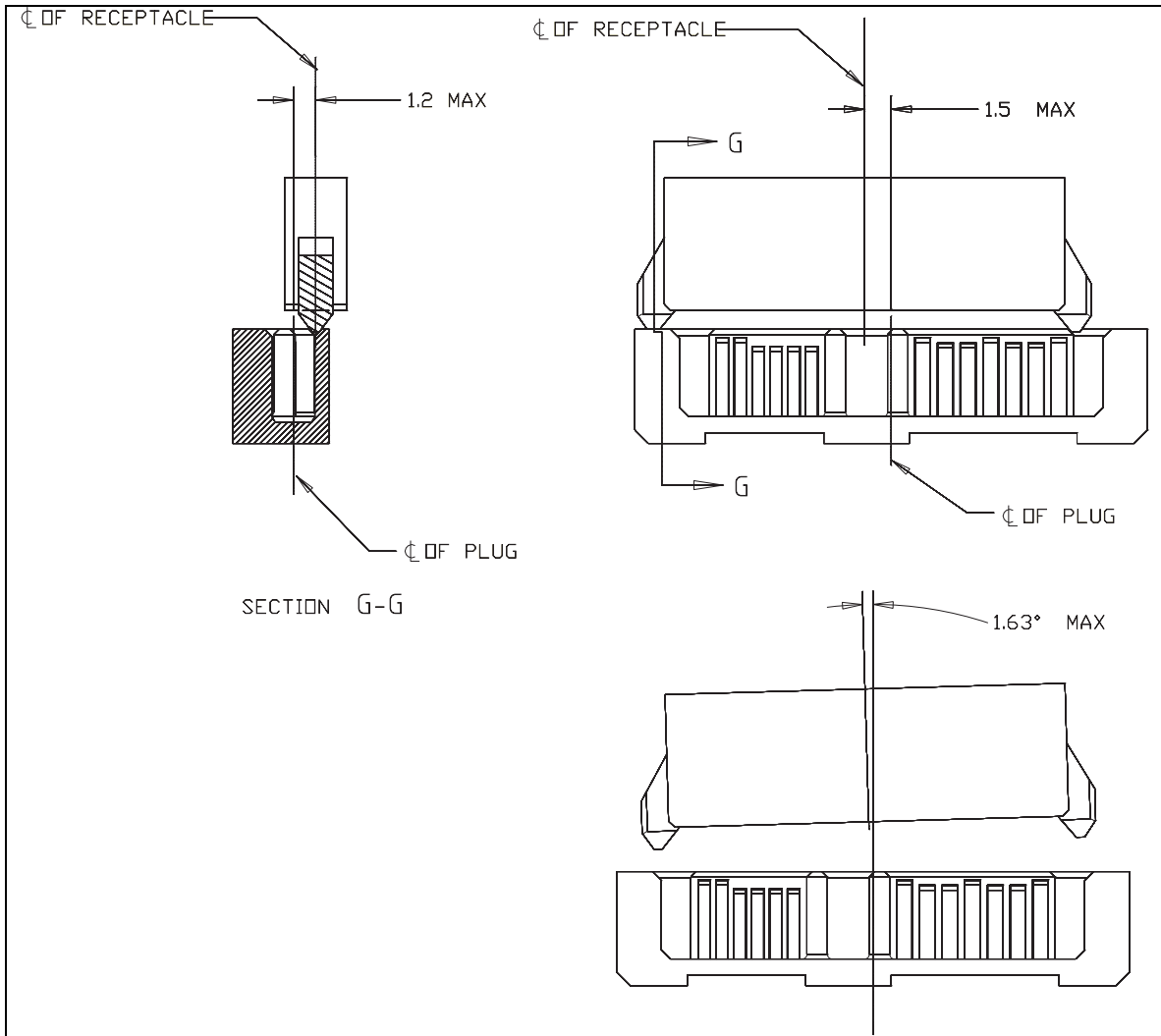


Figure 90 – Slimline connector pair blind-mate misalignment tolerance

#### 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**

Removal force Backplane connector	EIA-364-13 Measure the force necessary to unmate the connector assemblies at a max. rate of 12.5 mm/min.	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 mm/min.	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 mm/min.	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 mm/min.	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 Gbps) and Gen2 (3 Gbps) transfer rates;
- b) support for FPC usage models;
- c) support for 8.0 mm and 5.0 mm slim 1.8 inch Form Factor (FF) 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 Device Activity Signal/Disable Staggered Spin up). 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 91 for reference purposes only. See SFF-8146 for form factor definition and connector location.

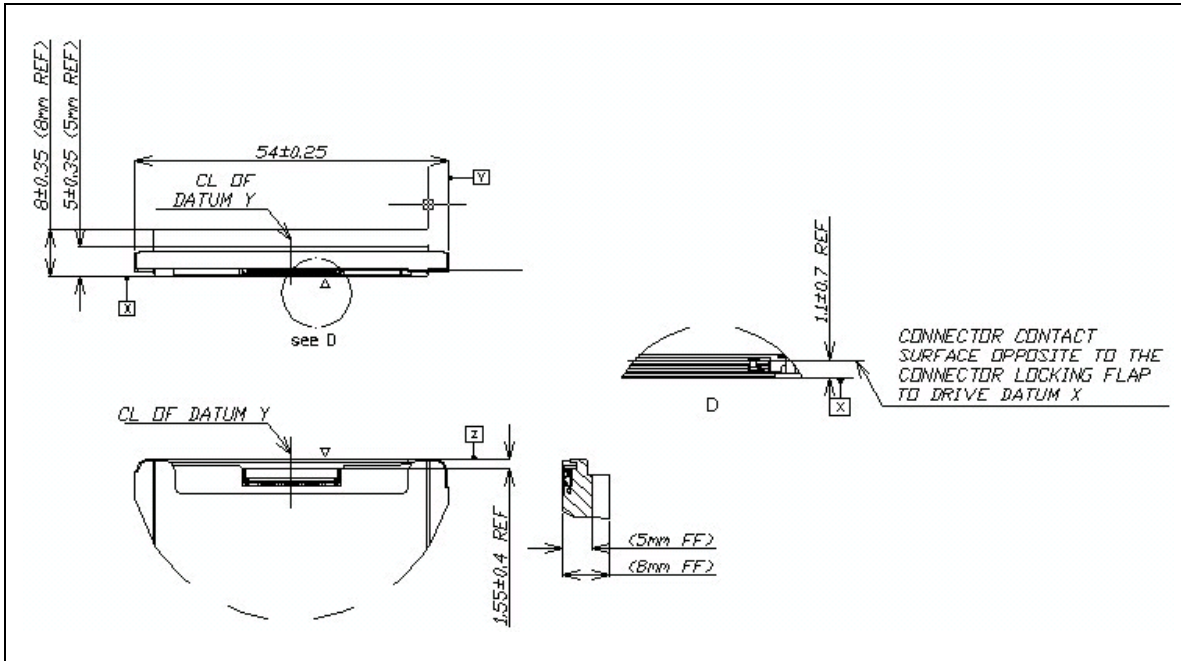


Figure 91 – Internal LIF-SATA connector location for 1.8 inch HDD

### 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 92 for reference purposes only.

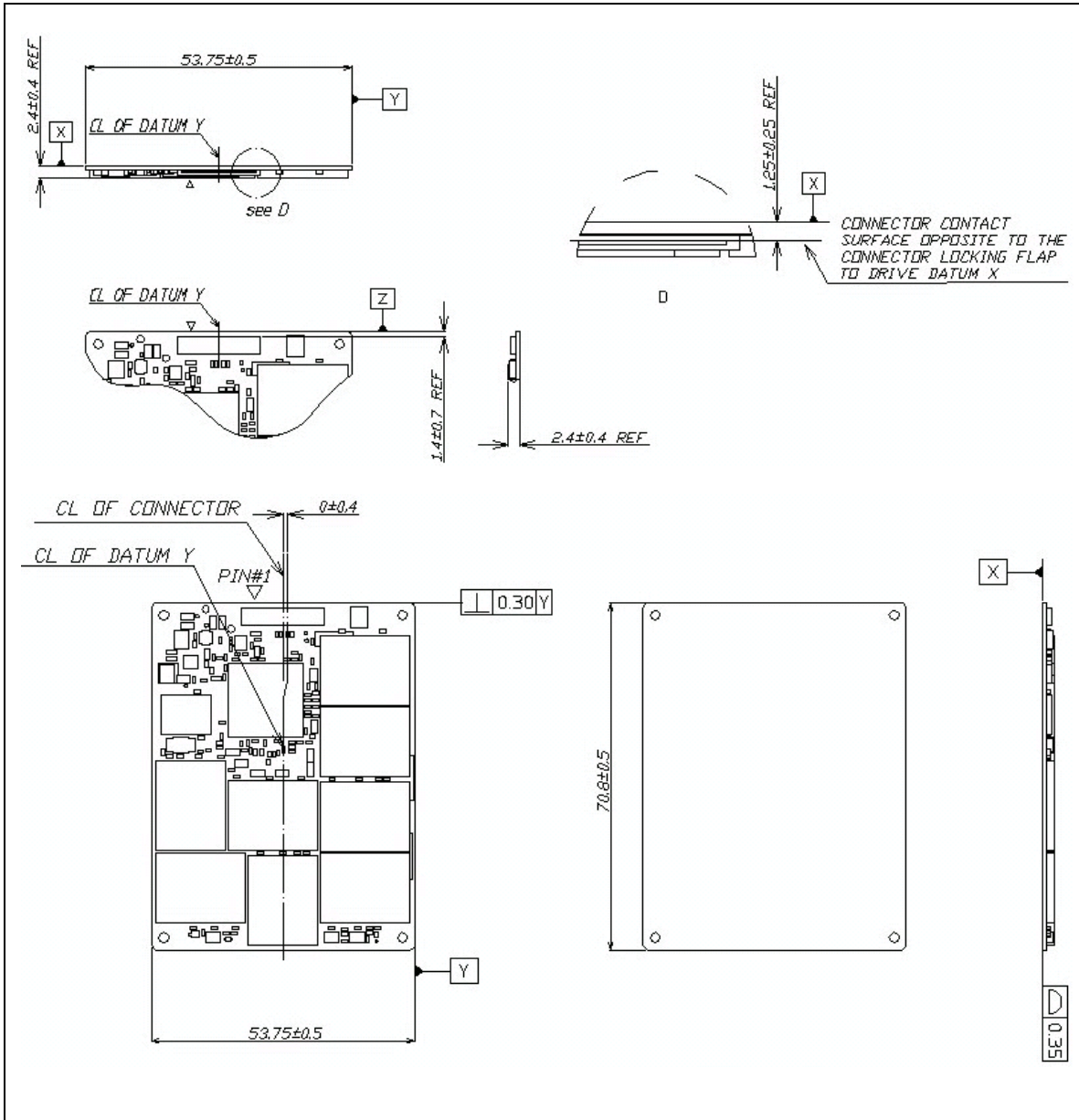


Figure 92 – Internal LIF-SATA connector location for 1.8 inch SSD bulk of single-sided mount type

## **6.5.4 Mating interfaces**

### **6.5.4.1 Device internal LIF-SATA embedded type connector**

Figure 93 defines the interface dimensions for the internal LIF-SATA embedded type connector with both signal and power segments.

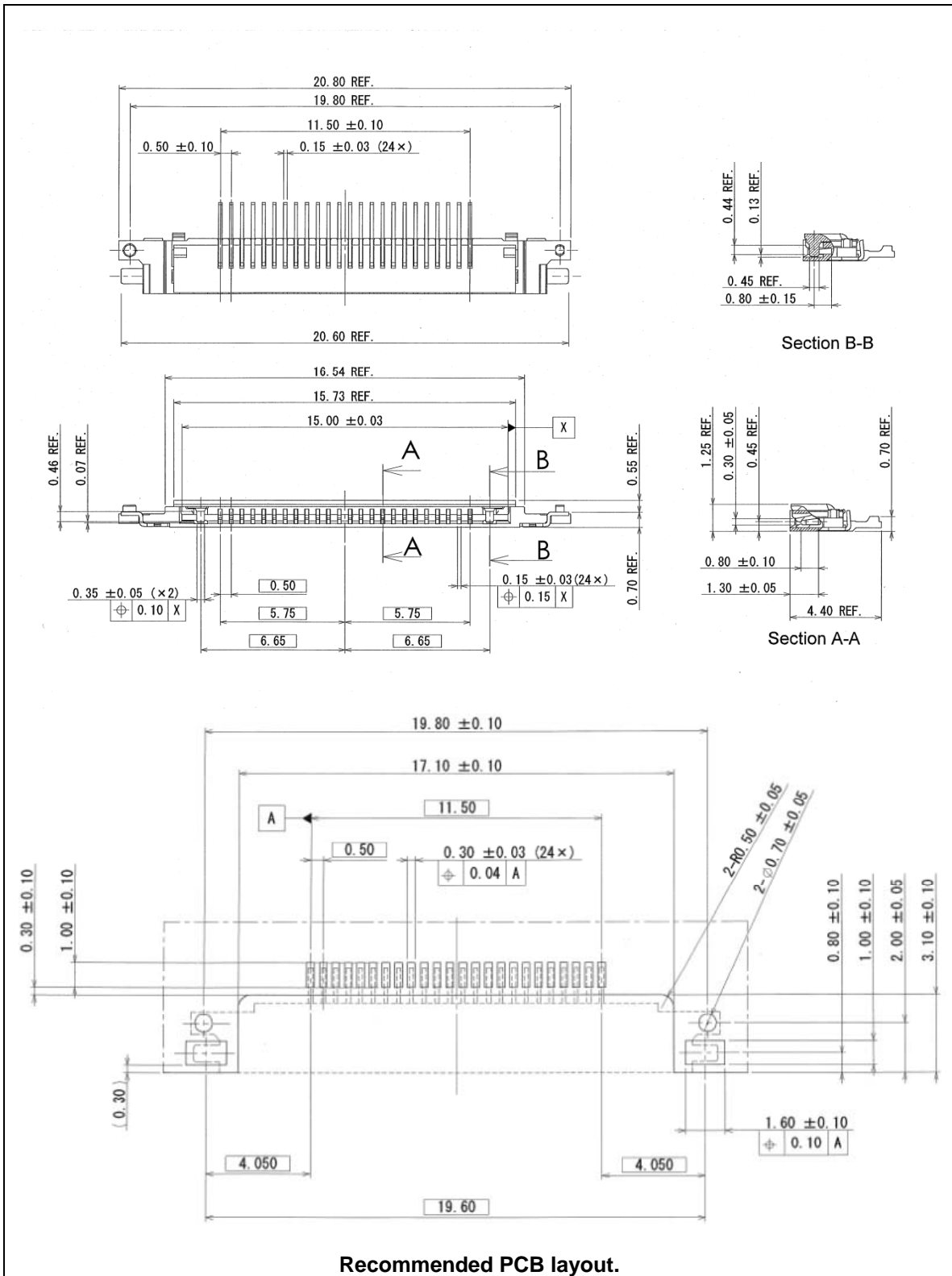
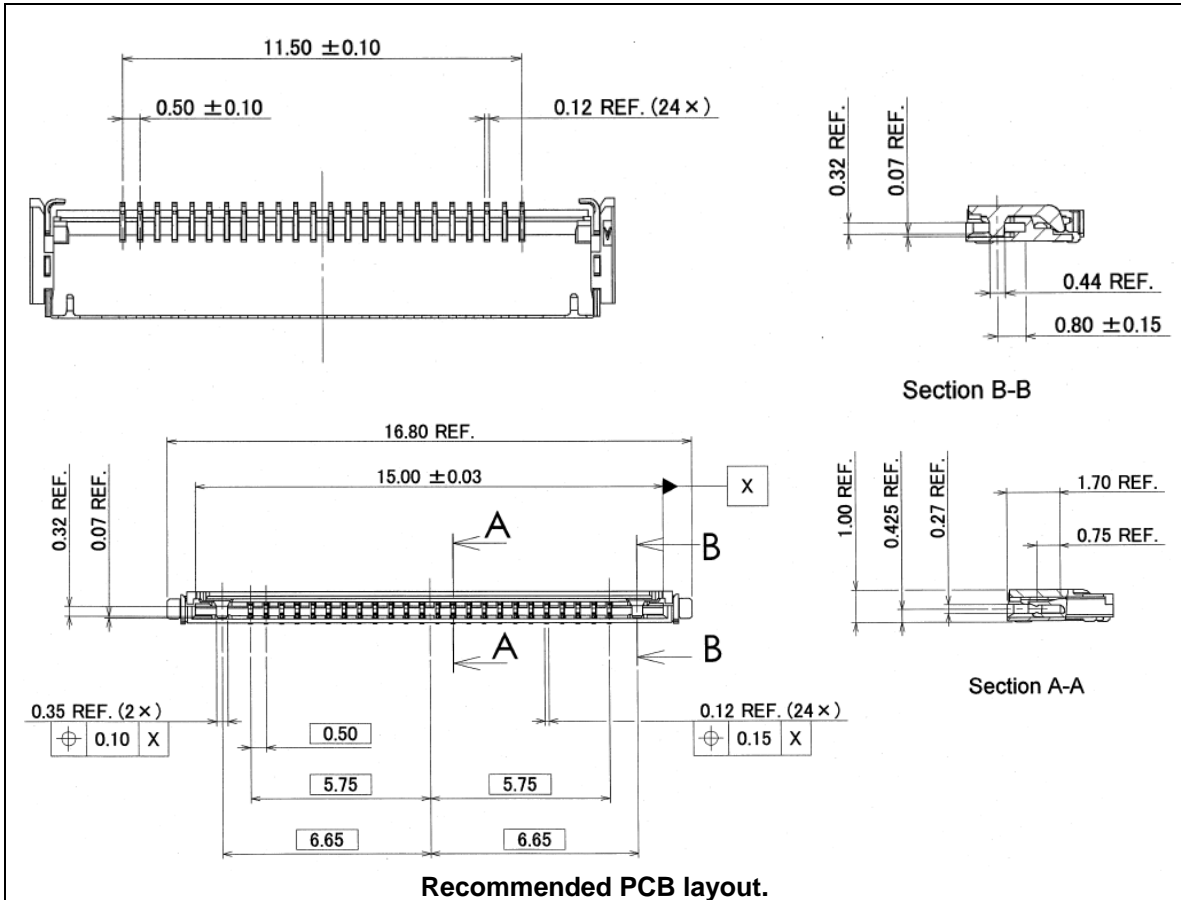


Figure 93 – Device internal LIF-SATA embedded type connector



**6.5.4.2 Device internal LIF-SATA surface mounting type connector**

Figure 94 defines the interface dimensions for the internal LIF-SATA device surface mounting type connector.



**Figure 94 – Device internal LIF-SATA surface mounting type connector**

**6.5.4.3 FPC for internal LIF-SATA**

Figure 95 defines the interface dimensions for the FPC of LIF-SATA.

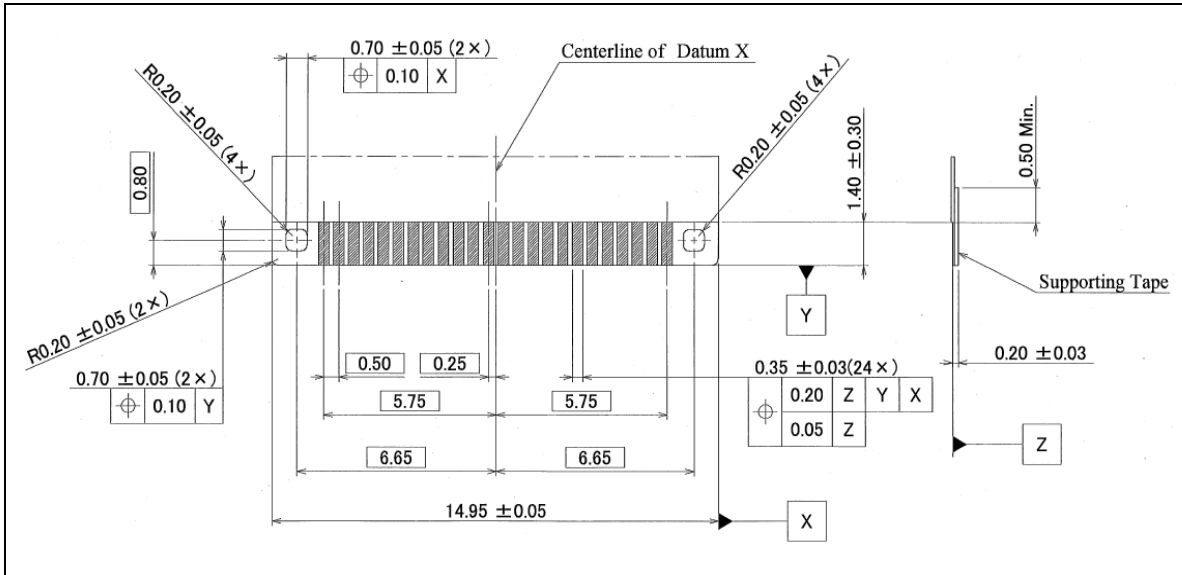


Figure 95 – FPC for internal LIF-SATA

### 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**

<b>Name</b>	<b>Type</b>	<b>Description</b>
P1	GND	Ground
P2	V <sub>33</sub>	3.3 V Power
P3	V <sub>33</sub>	3.3 V Power
P4	GND	Ground
P5	V <sub>5</sub>	5 V Power
P6	V <sub>5</sub>	5 V Power <sup>a</sup>
P7	GND	Ground
P8	DAS/DSS/Vendor Specific	Device Activity Signal/Disable Staggered Spinup/Vendor Specific <sup>b</sup>
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 <sup>d</sup>
P22	Vendor	Vendor Specific - Mfg pin <sup>c</sup>
P23	Vendor	Vendor Specific - Mfg pin <sup>c</sup>
P24	GND	Ground

<sup>a</sup> 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.  
<sup>b</sup> For specific optional usage of pin P8 see 6.13.  
<sup>c</sup> No connect on the host side.  
<sup>d</sup> 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**

Insertion force for FPC	Measure the force necessary to mate the connector assemblies at a max. rate of 12.5 mm/min.	Max 6 N.
Removal force for FPC	Measure the force necessary to unmate the connector assemblies at a max. rate of 12.5 mm/min.	Min 6 N through 10 cycles.
Durability	10 cycles for internal cabled application; Test done at a maximum rate of 200 cycles/h.	No physical damage.
Minimum current	Mount the connector to a test PCB: <ol style="list-style-type: none"> <li>1) wire power pins P2, P3, P5, P6 in parallel for power;</li> <li>2) wire ground pins P1, P4, P7, P9, P10, P13, P16, P17, P24 in parallel for return;</li> <li>3) supply 2 A total DC current to the power pins in parallel, returning from the parallel ground pins; and</li> <li>4) record temperature rise when thermal equilibrium is reached.</li> </ol>	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 Gbps) and Gen2 (3 Gbps) 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 the tables below. 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 Connector location on mSATA host

See JEDEC MO-300 for form factor definition and connector location. The connector location is defined in the drawing below.

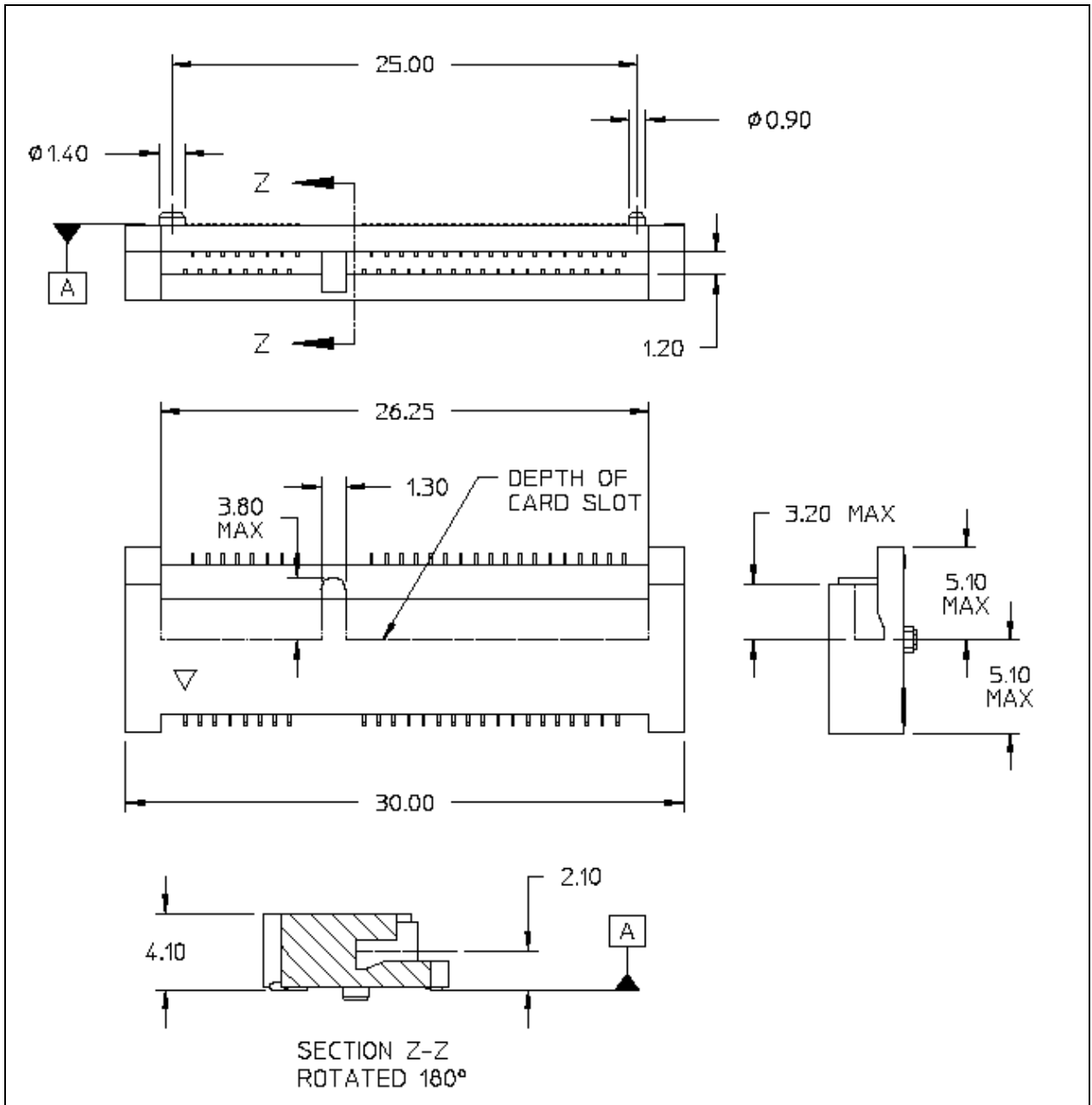


Figure 96 – mSATA card connector

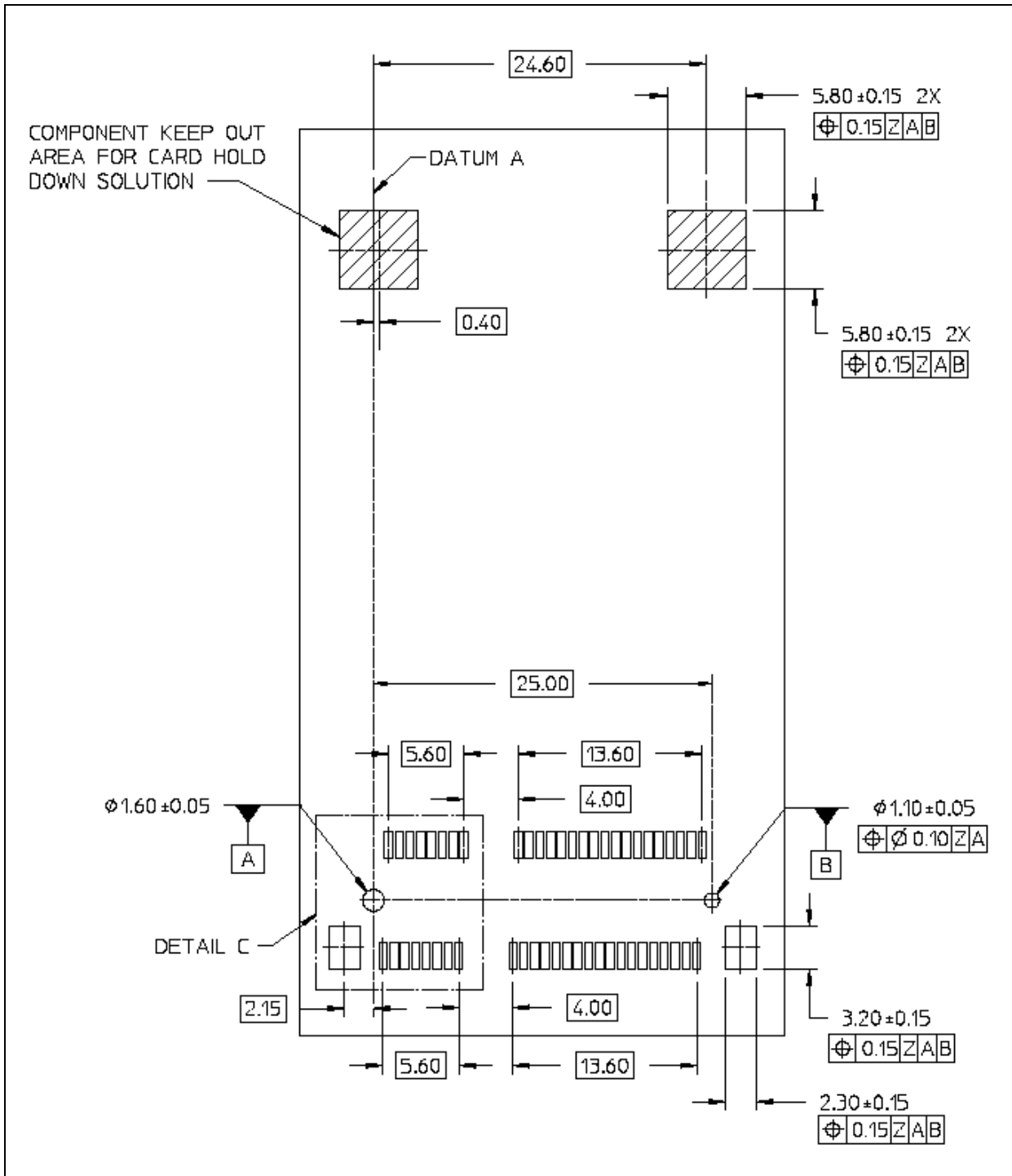


Figure 97 – mSATA card footprint and keepout

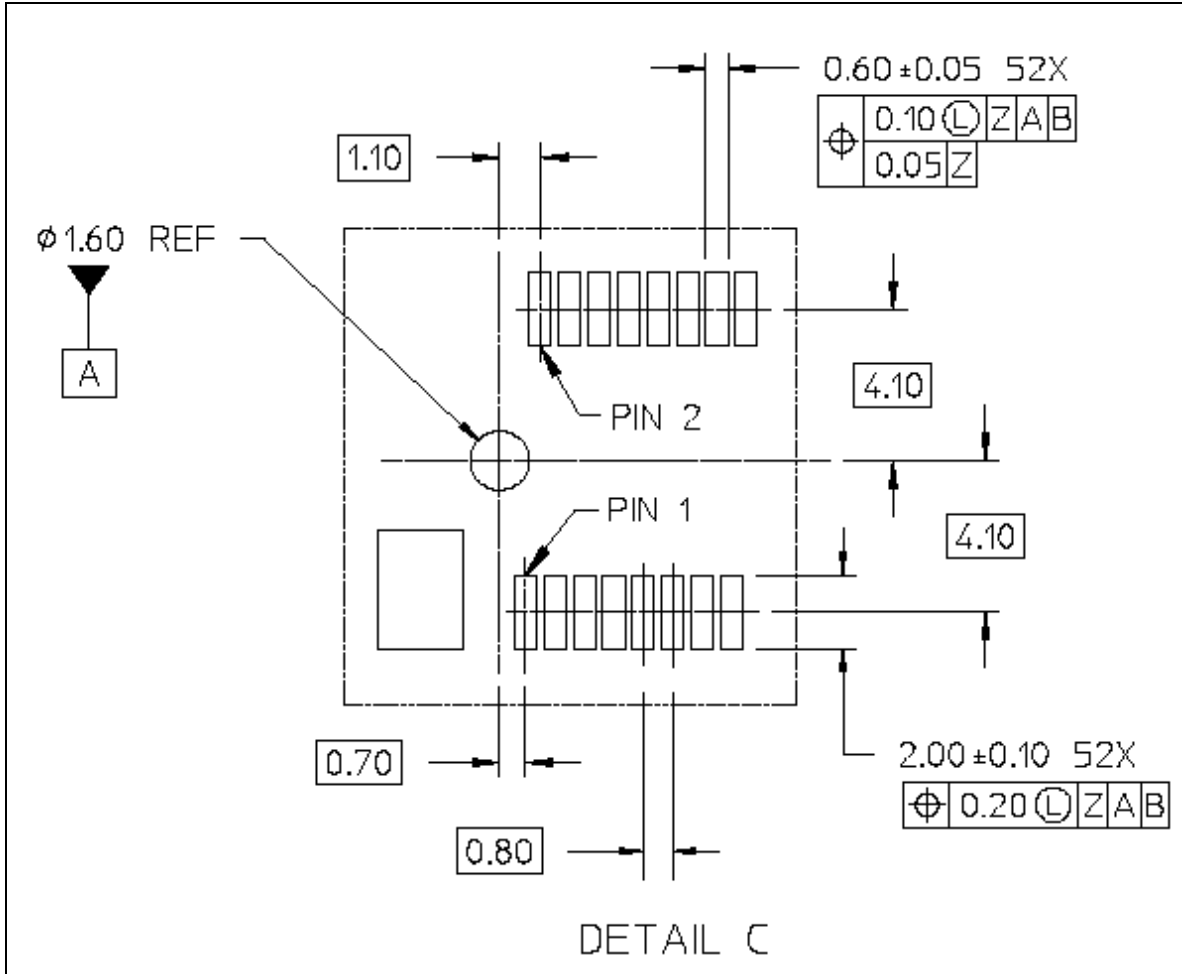


Figure 98 – mSATA card connector location detail C

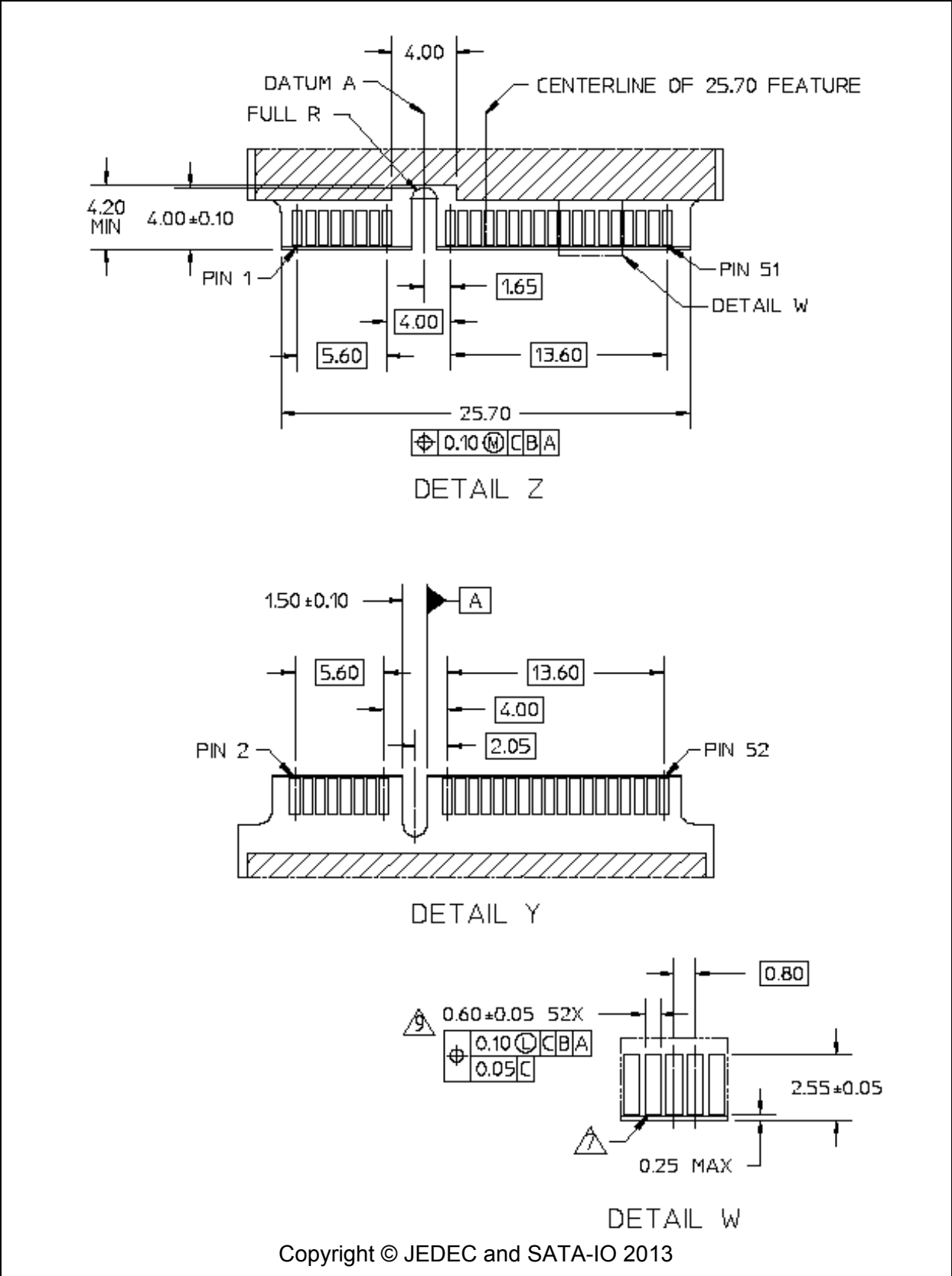
## 6.6.4 Mating interfaces

### 6.6.4.1 Device mSATA card embedded type connector

Figure 100 defines the device mSATA card type internal connector.

### 6.6.4.2 mSATA interface dimensions

Figure 98 defines the interface dimensions for the mSATA connector.



Copyright © JEDEC and SATA-IO 2013

Figure 99 – mSATA card connector location detail W, Y and Z



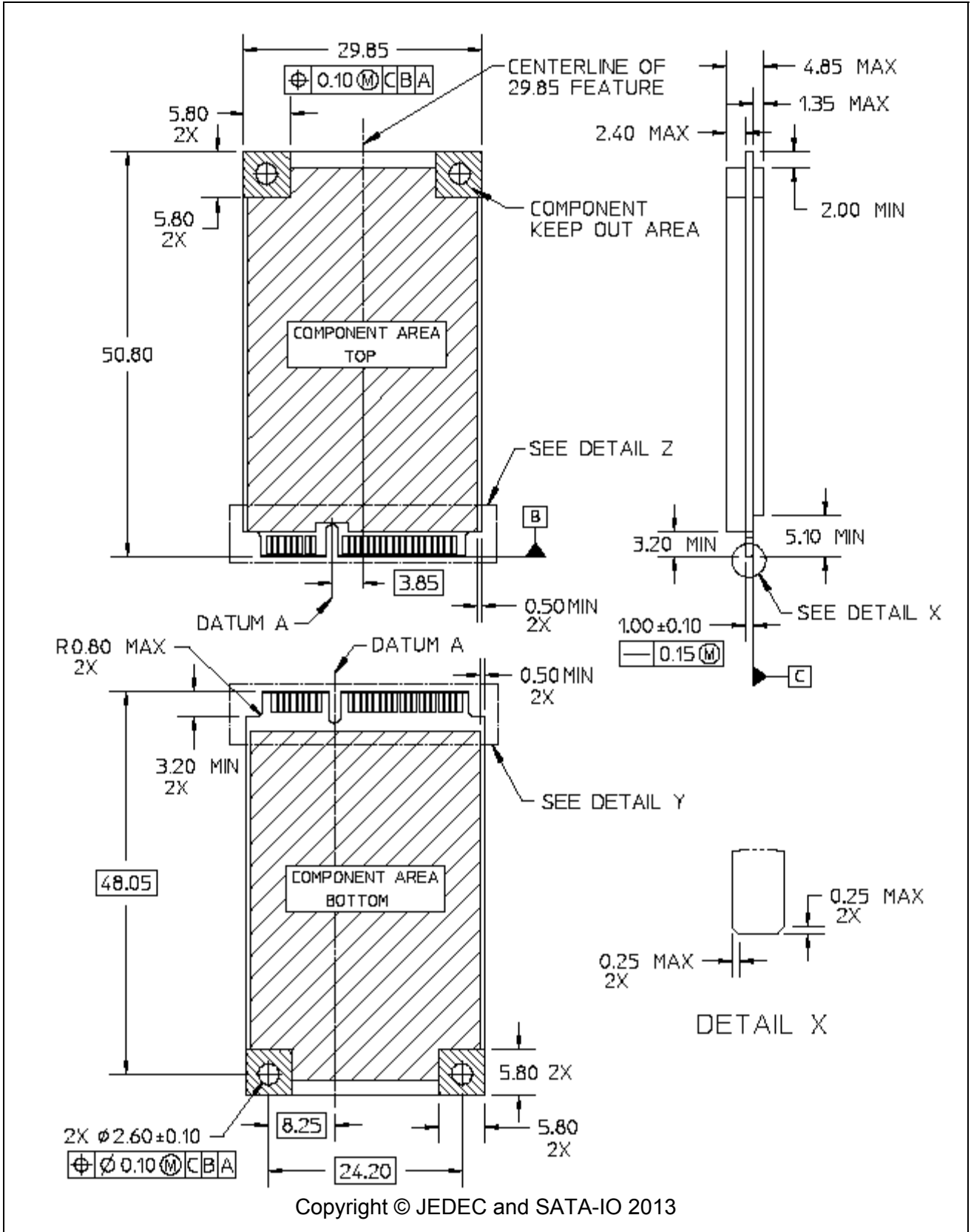


Figure 100 – Device mSATA card type internal connector

### 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 device activity / disable staggered spinup;
- f) 1 pin for presence detection;
- 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 3)

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

<sup>a</sup> Presence detection pin indicates presence of an mSATA device. See Figure 101 for an informative host-side bi-directional implementation for compatibility with non-mSATA devices that may use this pin as an input.

<sup>b</sup> No connect on the host side.

<sup>c</sup> 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.

<sup>d</sup> 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 3)**

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 <sup>c</sup>
P31	-A	Host Transmitter Differential Signal Pair
P32	Two Wire Interface	Two Wire Interface Data <sup>c</sup>
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 <sup>d</sup>
P44	DEVSLP	Enter/Exit DevSleep
P45	Vendor	Vendor Specific / Manufacturing Pin <sup>b</sup>

<sup>a</sup> Presence detection pin indicates presence of an mSATA device. See Figure 101 for an informative host-side bi-directional implementation for compatibility with non-mSATA devices that may use this pin as an input.

<sup>b</sup> No connect on the host side.

<sup>c</sup> 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.

<sup>d</sup> 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 3 of 3)**

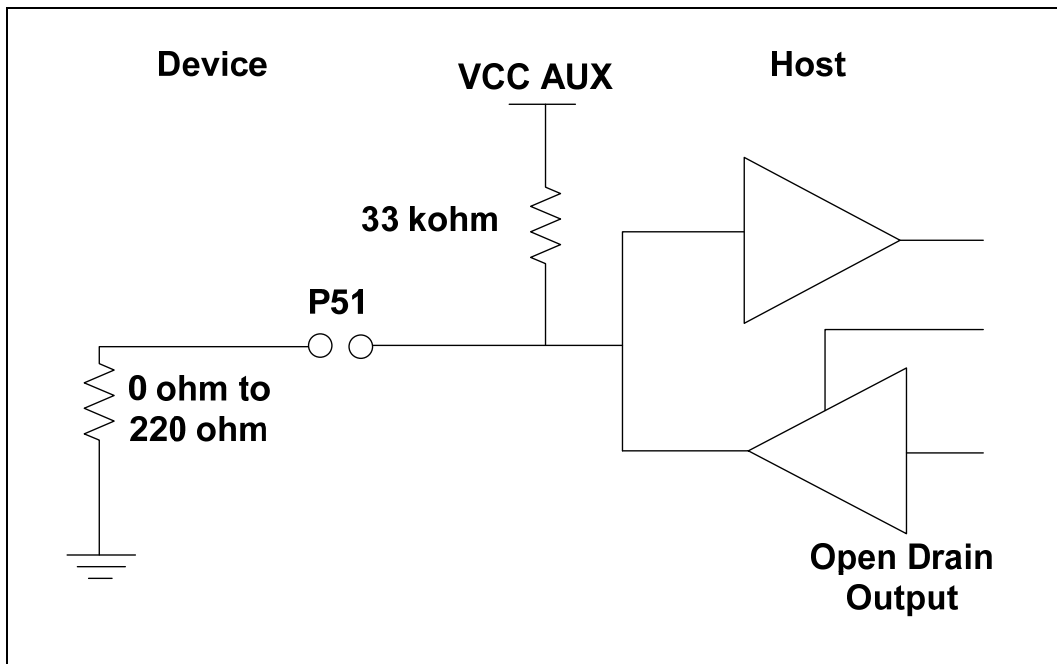
Pin position	Type	Description
P46	Reserved	No Connect
P47	Vendor	Vendor Specific / Manufacturing Pin <sup>b</sup>
P48	+1.5 V	1.5 V Source
P49	DAS/DSS	Device Activity Signal / Disable Staggered Spinup
P50	GND	Ground
P51	Presence Detection	Shall be Connected to GND by a 0 ohm to 220 ohm Resistor on device <sup>a</sup>
P52	+3.3 V	3.3 V Source

<sup>a</sup> Presence detection pin indicates presence of an mSATA device. See Figure 101 for an informative host-side bi-directional implementation for compatibility with non-mSATA devices that may use this pin as an input.

<sup>b</sup> No connect on the host side.

<sup>c</sup> 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.

<sup>d</sup> 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 101 – 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 102 and Table 20 show the connector location on the 14.5 mm SATA USM device. Figure 103 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 (available at <http://www.sffcommittee.org>) for additional details regarding the SATA USM.

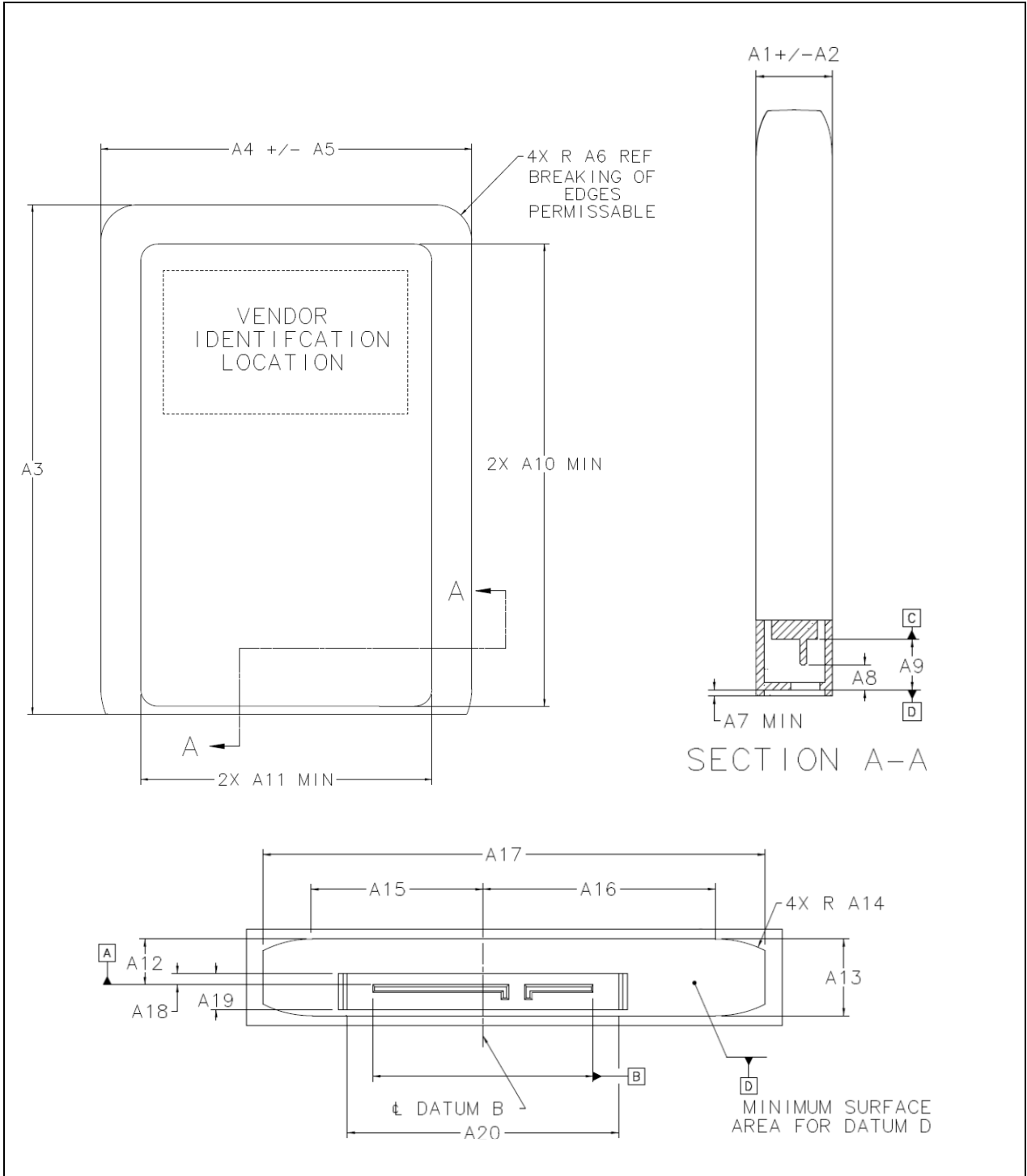


Figure 102 – 14.5 mm SATA USM physical dimensions (see INF-8280)

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

<b>Dimension</b>	<b>mm</b>	<b>inch</b>
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



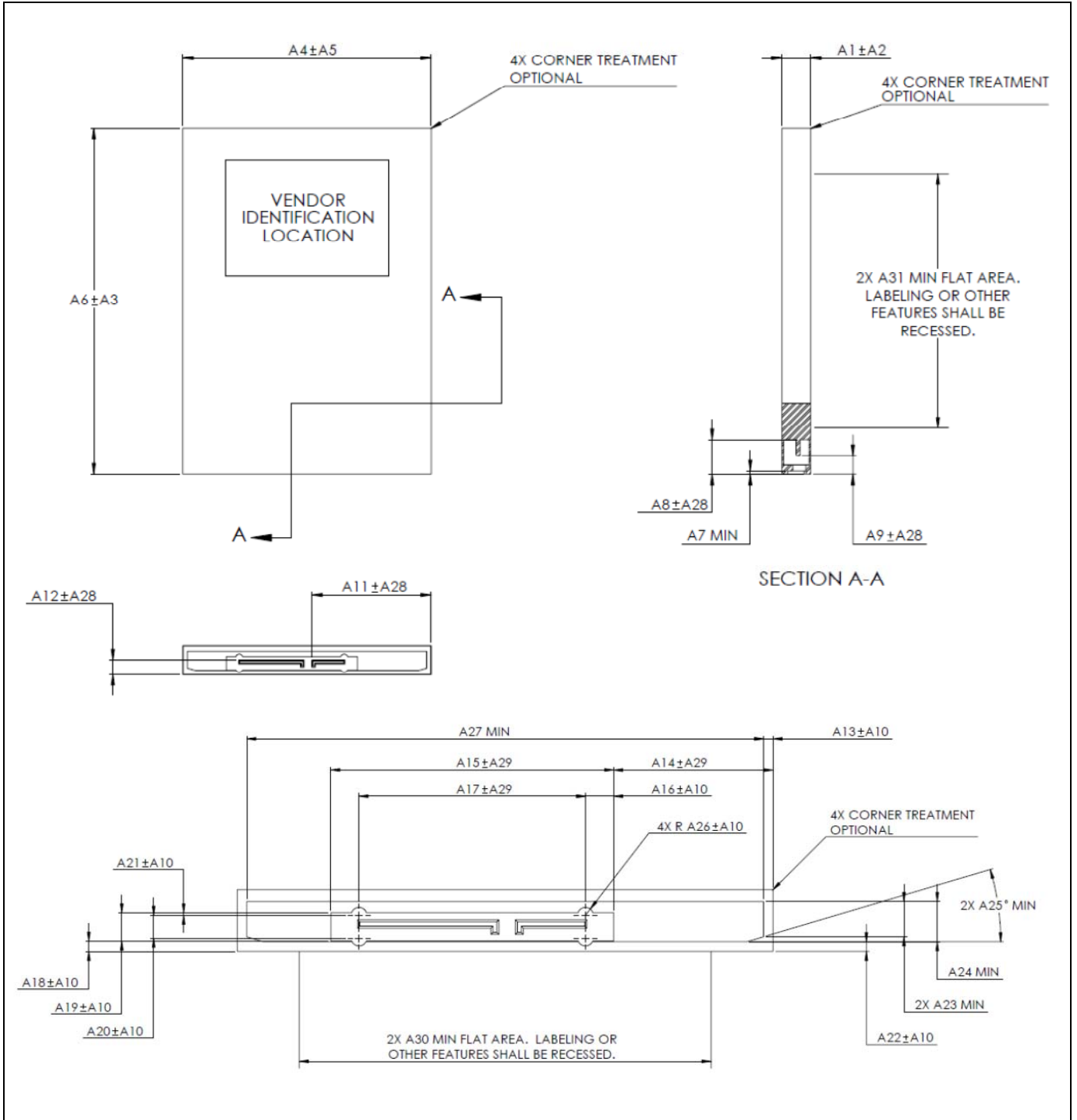


Figure 103 – 9 mm SATA USM physical dimensions (see INF-8280)

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

<b>Dimension</b>	<b>mm</b>	<b>inch</b>
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 104, Figure 105, Figure 106 and Figure 107.

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, it should be noted 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.

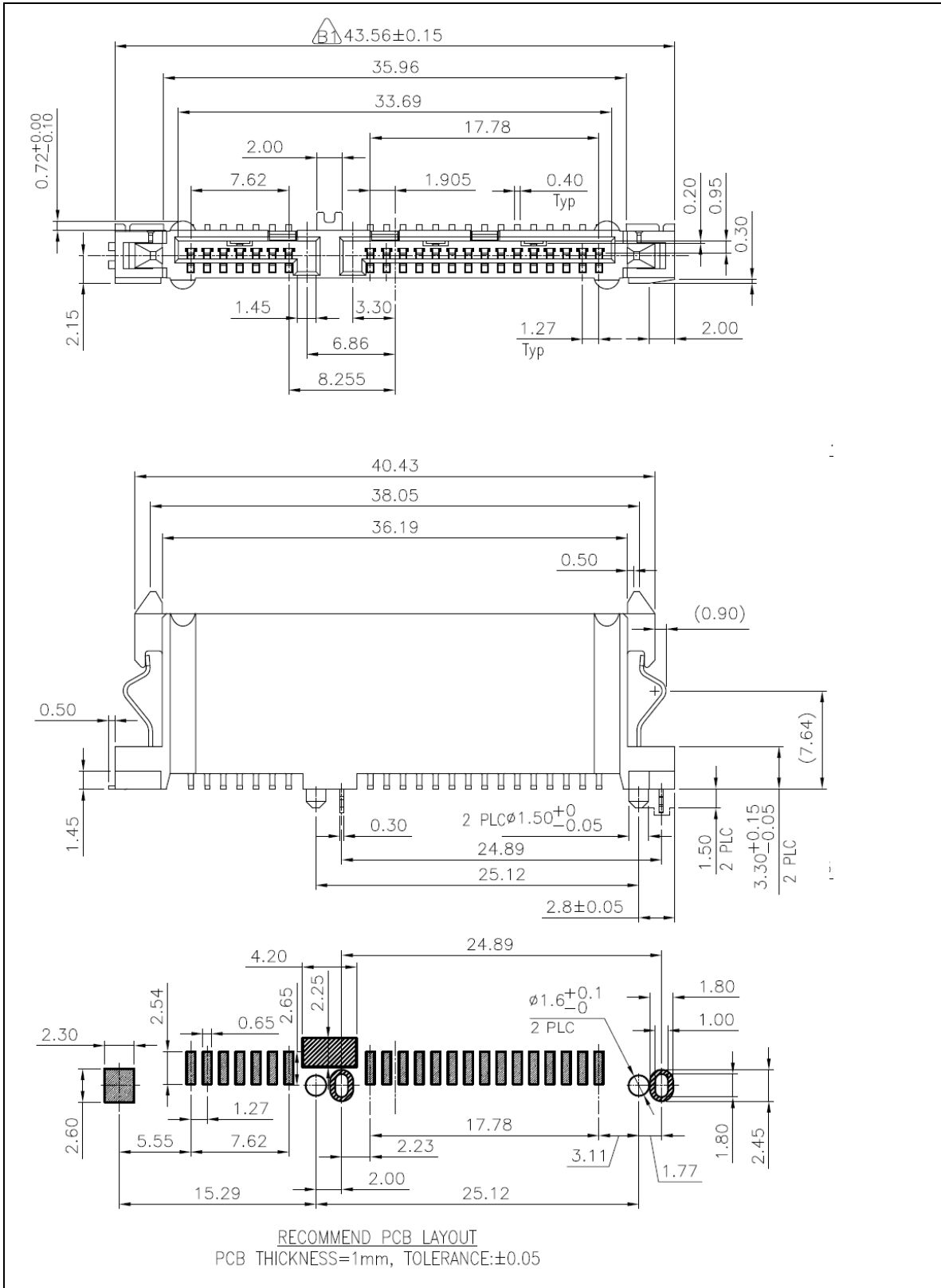


Figure 104 – SATA USM vertical receptacle (see INF-8280)

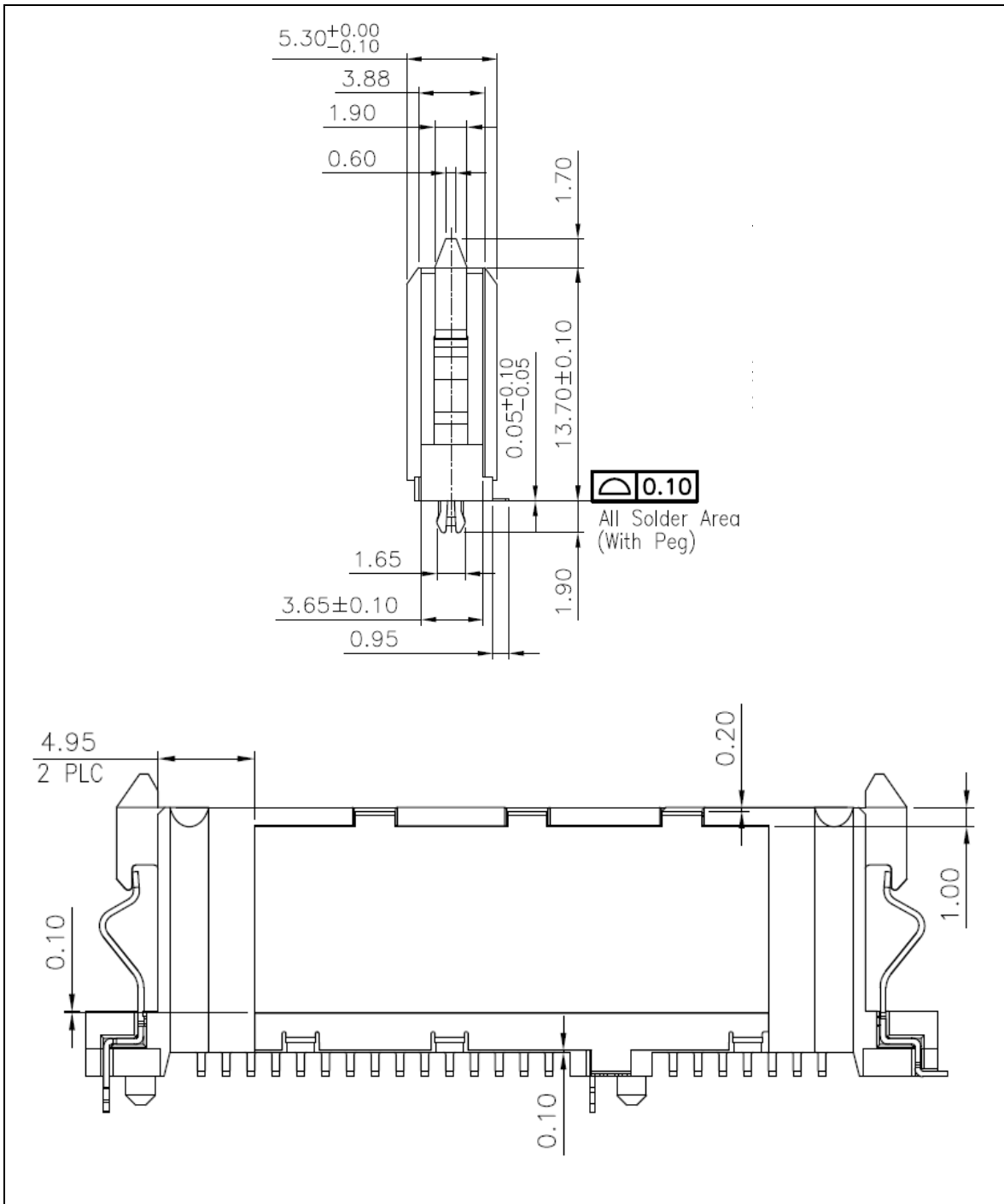


Figure 105 – SATA USM vertical receptacle continued (see INF-8280)

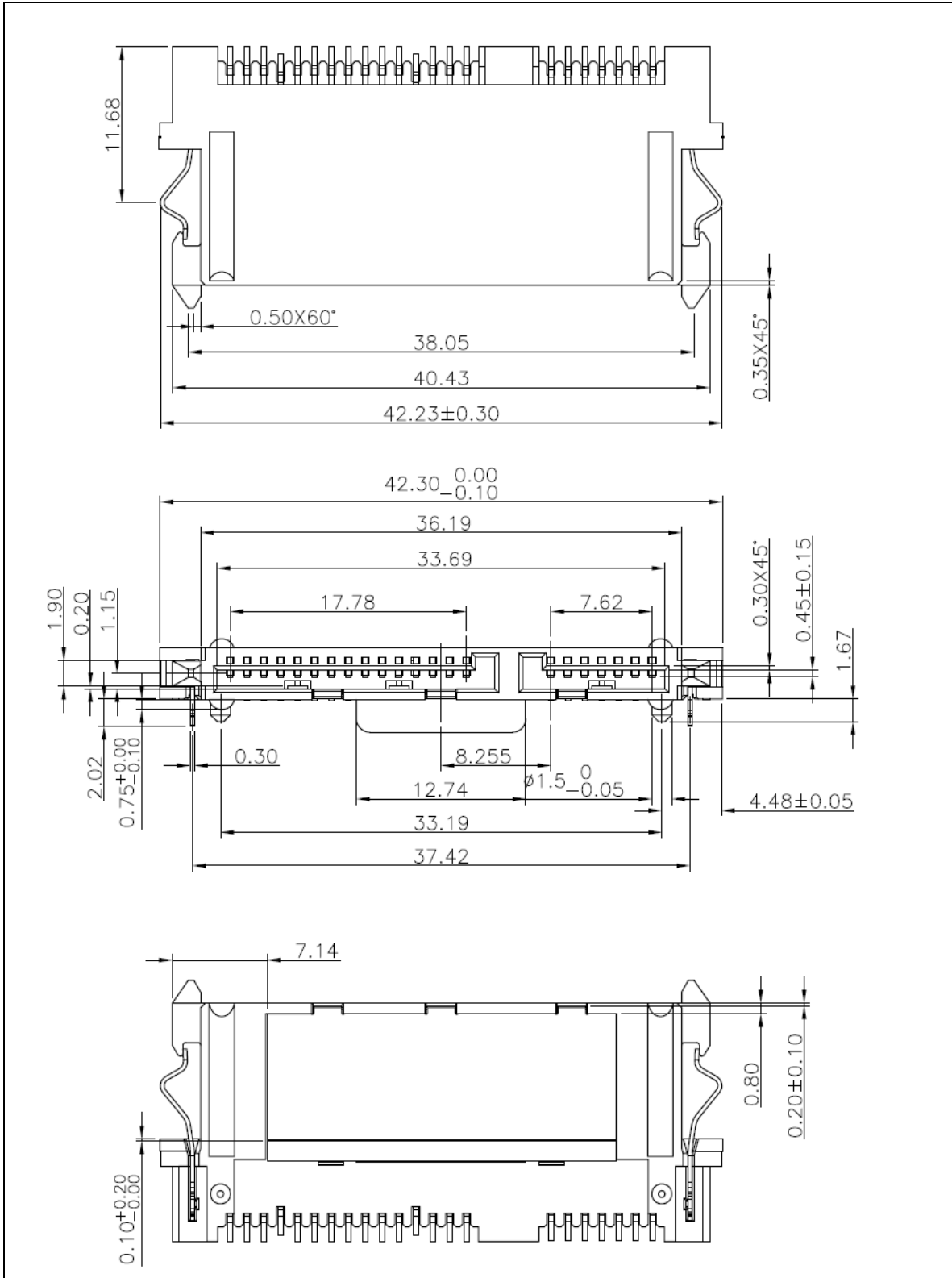


Figure 106 – SATA USM horizontal receptacle (see INF-8280)

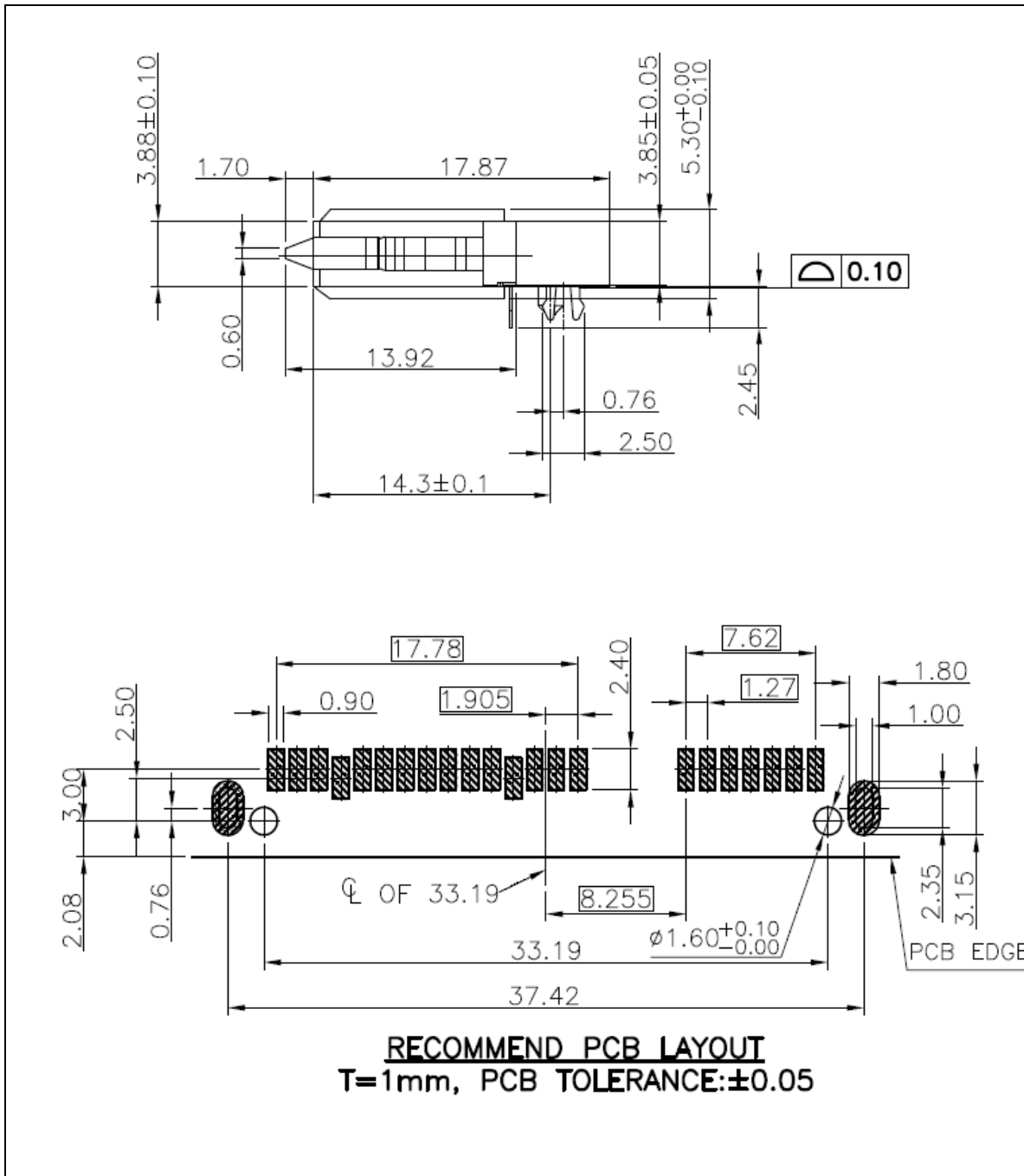


Figure 107 – SATA USM horizontal receptacle continued (see INF-8280)

## 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**

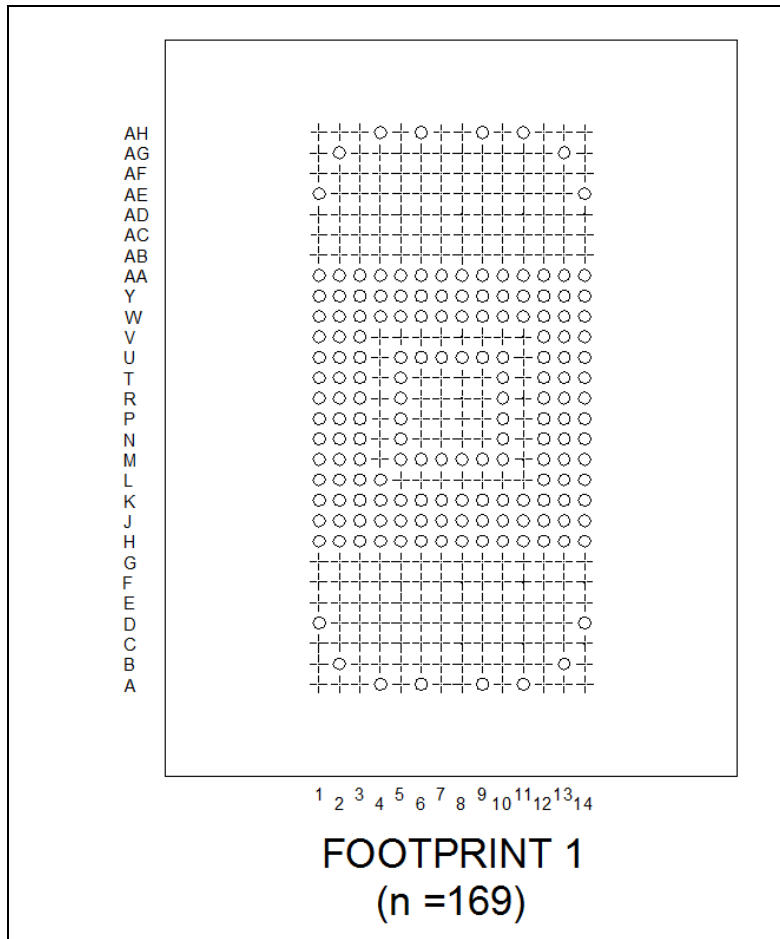
A SATA MicroSSD shall use package variants:

- a) AC;
- b) AM;
- c) AK;
- d) AL;
- e) AR;
- f) CA;
- g) CB;
- h) DB; or
- i) DC,

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 distinct package footprints. The package footprints (bottom view) are shown below. A "+" sign denotes a depopulated ball position. The functional balls (signals or power) for each footprint are the inner 3 or 4 rows on each footprint (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 108 – Footprint 1, SATA MicroSSD variant AC, 169 balls (informative)**



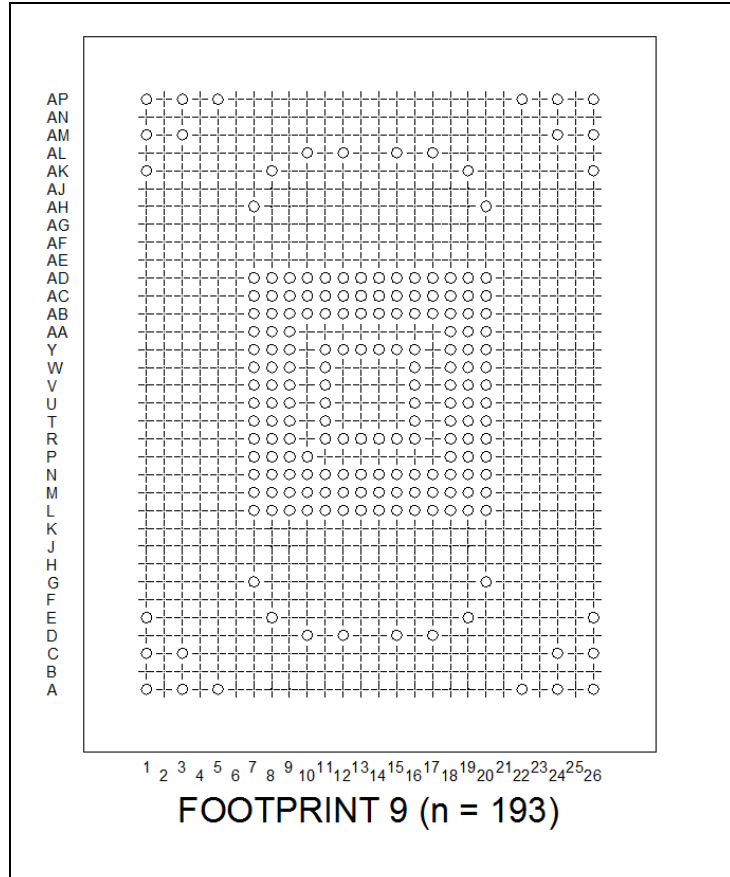
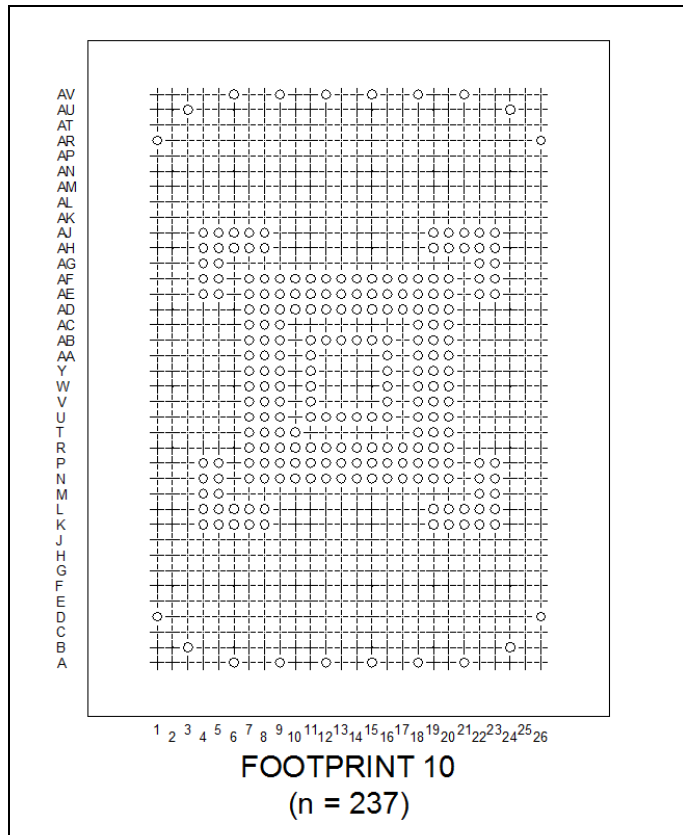


Figure 109 – Footprint 9, SATA MicroSSD variant AK and CB, 193 balls (informative)



**Figure 110 – Footprint 10, SATA MicroSSD variant AL, 237 balls (informative)**

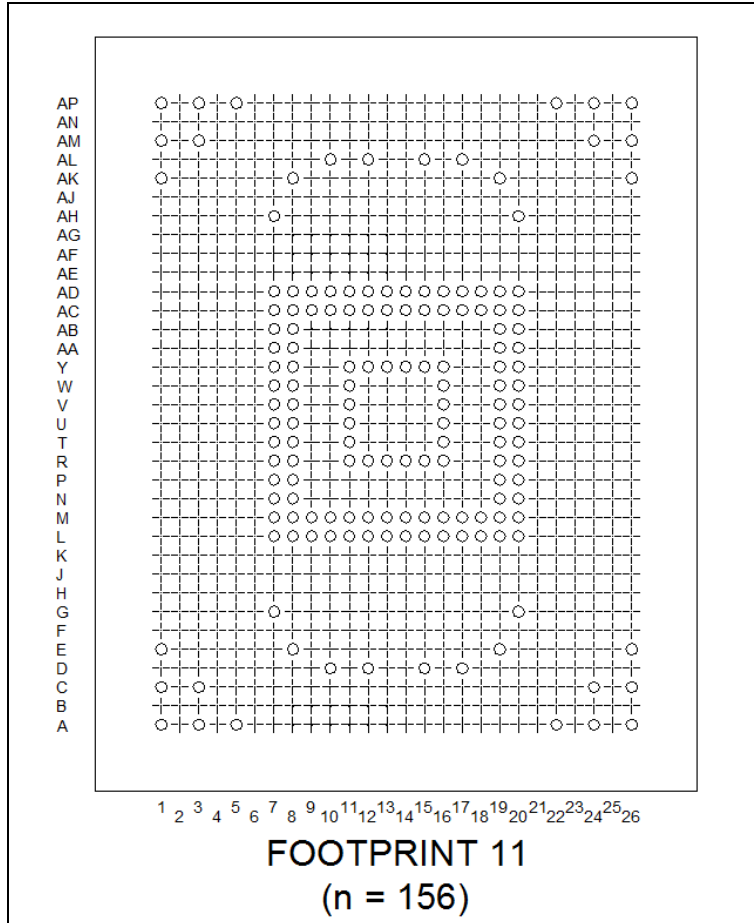
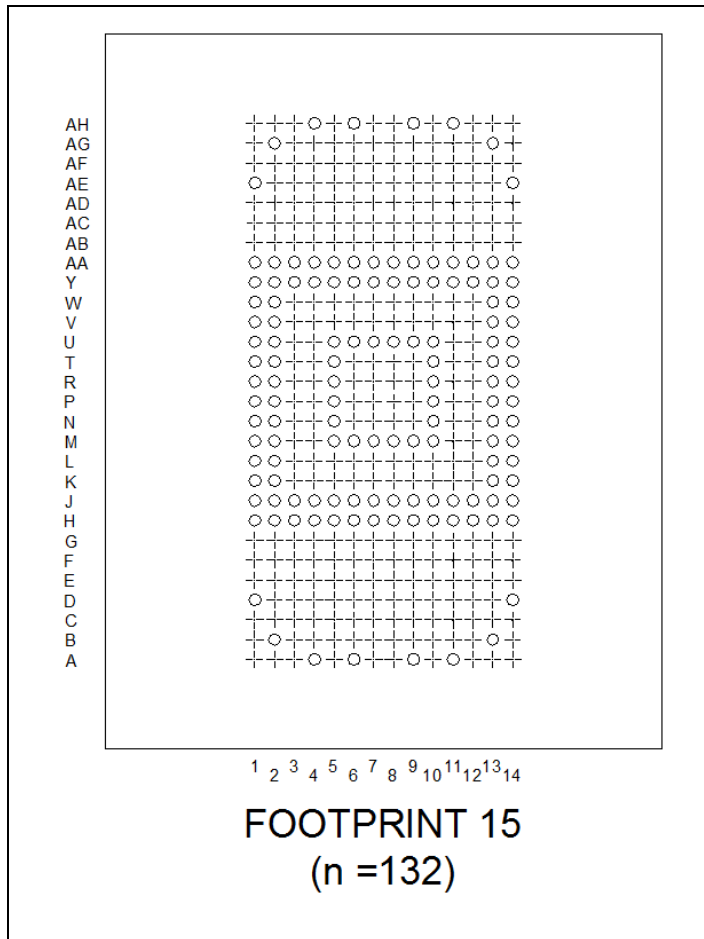


Figure 111 – Footprint 11, SATA MicroSSD variant AM and CA, 156 balls (informative)



**Figure 112 – 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 Section 6.8.2.

Some of the general categories are:

- a) VSP, vendor specific;
- b) Reserved, reserved for future standardization (See 4.2.3.9);
- 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)			
<b>SATA Interface Signals</b>							
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
<b>Optional Signals</b>							
AA10	AD16	AA10	AD16	AF16	SPI_MISO	Input	Master In Slave Out
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	Output	Master Out Slave In
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
<p><sup>a</sup> Optional pin reserved for system clock output to drive crystal or other system requirements. Frequency is system implementation dependent.</p> <p><sup>b</sup> Optional pin reserved for system clock input. Frequency is system implementation dependent.</p> <p><sup>c</sup> Optional pin reserved for Hardware Reset. Implementation is system dependent. For detailed timing information, consult device data sheet.</p>							

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)			
<b>Control Signals</b>							
H3	L9	H3	L9	N9	XTAL_OUT	Output	System Clock output <sup>a</sup>
J4	M10	J4	M10	P10	XTAL_IN	Input	System Clock input <sup>b</sup>
J3	M9	J3	M9	P9	PWR_RESETN	Input	Hardware Reset <sup>c</sup>
<b>Power Supply Signals</b>							
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
<sup>a</sup> Optional pin reserved for system clock output to drive crystal or other system requirements. Frequency is system implementation dependent. <sup>b</sup> Optional pin reserved for system clock input. Frequency is system implementation dependent. <sup>c</sup> 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)			
<b>GND signals</b>							
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
<b>Do Not Use (DNU)</b>							
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)			
<b>Reserved (Continued)</b>							
		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		
<p><sup>a</sup> Optional pin reserved for system clock output to drive crystal or other system requirements. Frequency is system implementation dependent.</p> <p><sup>b</sup> Optional pin reserved for system clock input. Frequency is system implementation dependent.</p> <p><sup>c</sup> Optional pin reserved for Hardware Reset. Implementation is system dependent. For detailed timing information, consult device data sheet.</p>							

## **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 Gbps);
  - B) Gen2 (i.e., 3.0 Gbps); and
  - C) Gen3 (i.e., 6.0 Gbps);
- 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 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. The connector provides for 75 pin locations.

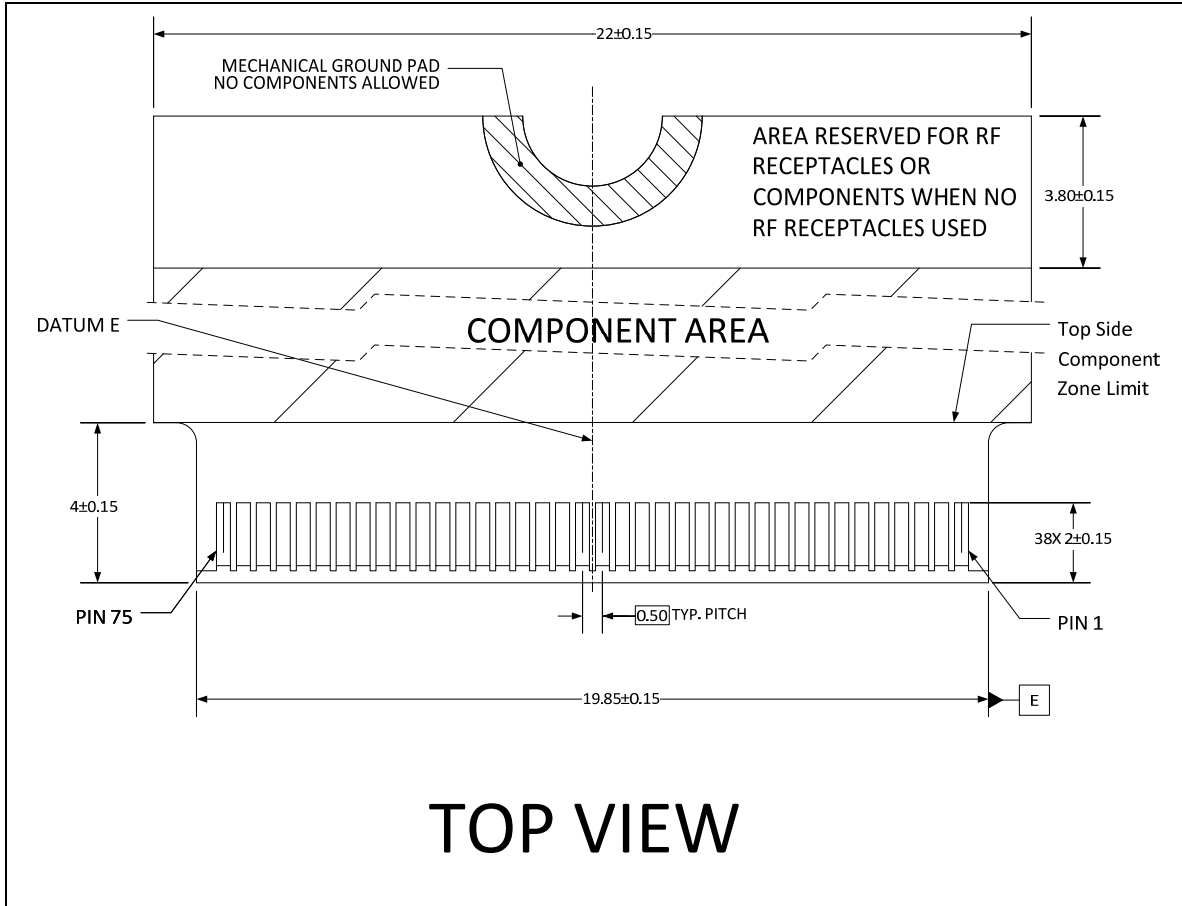


Figure 113 – M.2 board connector top details (informative)

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.

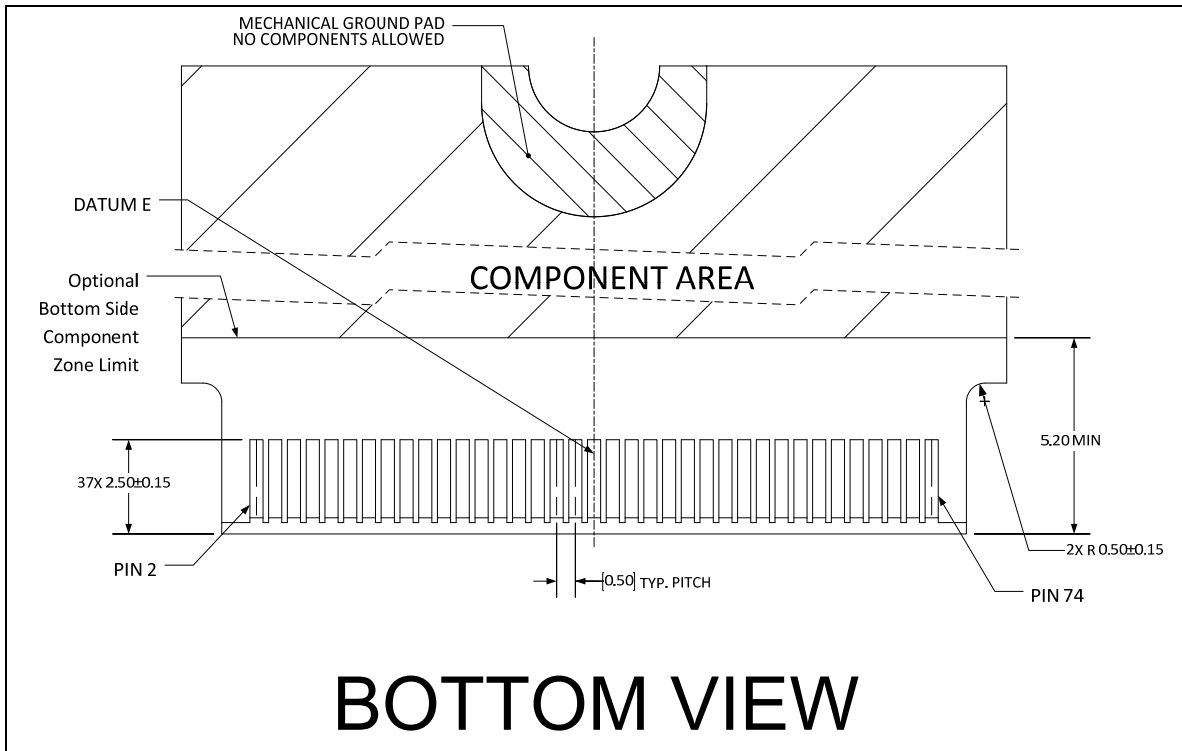


Figure 114 – M.2 board connector bottom details (informative)

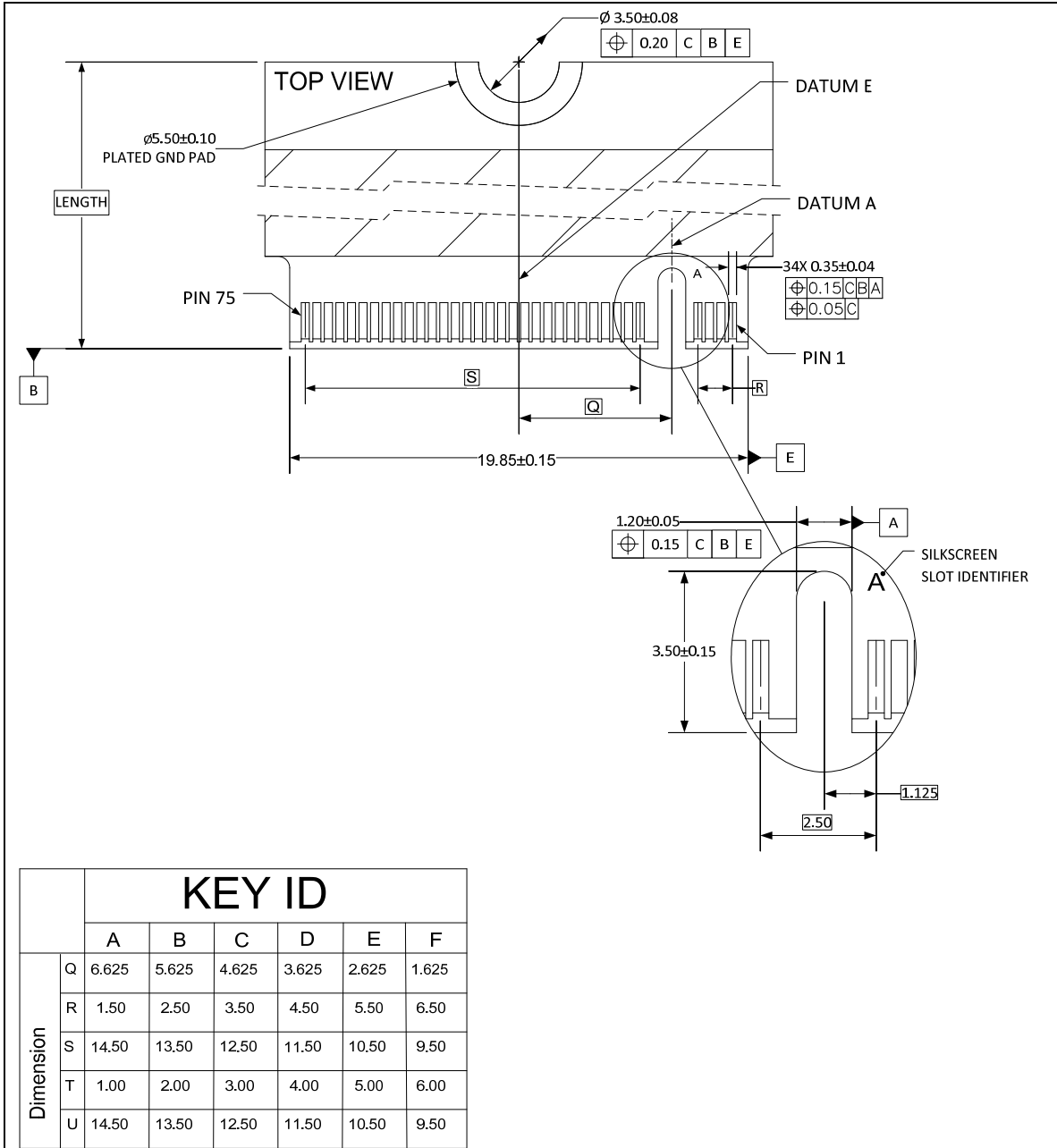


Figure 115 – M.2 board connector top slot details (informative)

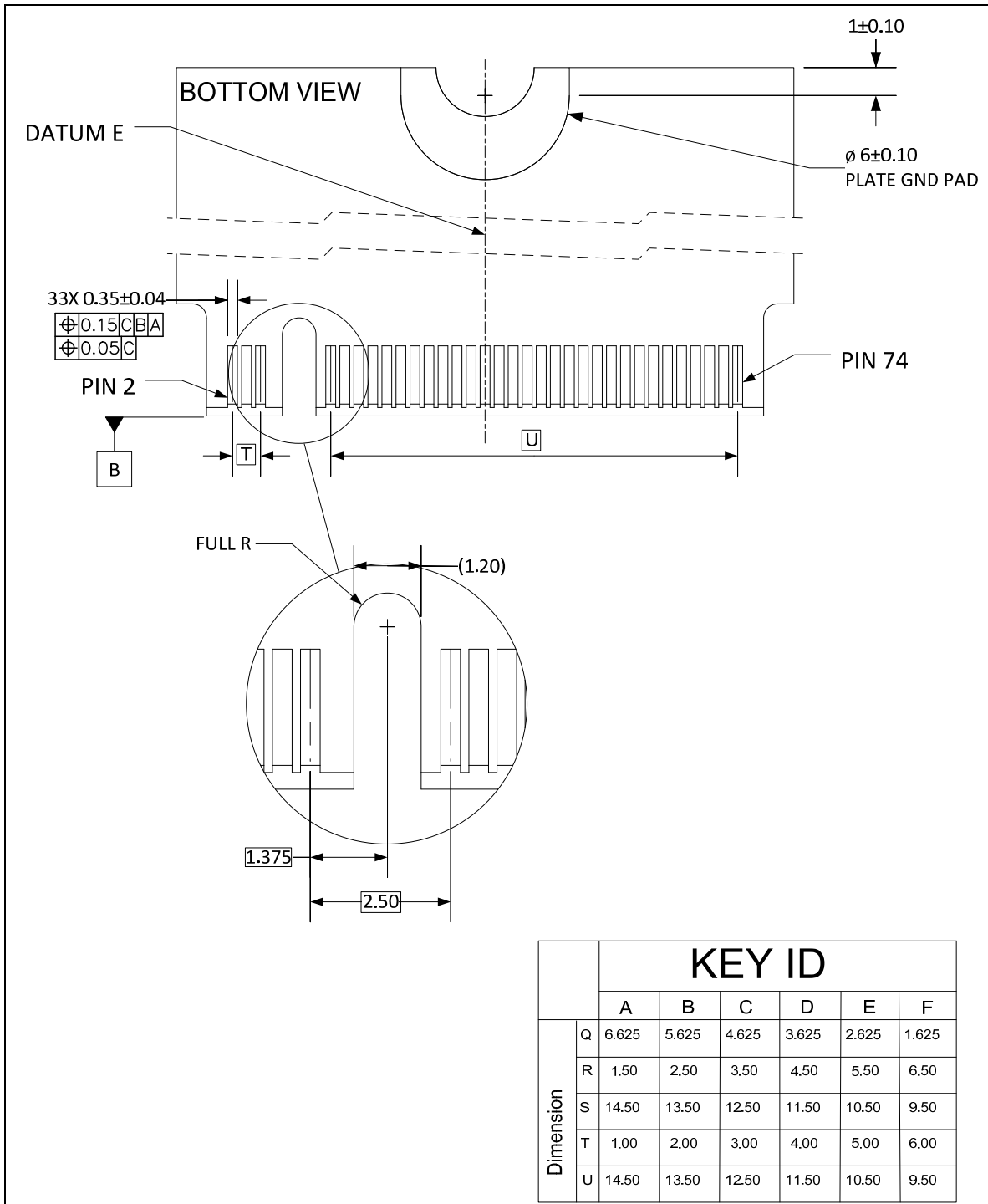


Figure 116 – M.2 board connector bottom slot details (informative)

### 6.9.4 M.2 keys (informative)

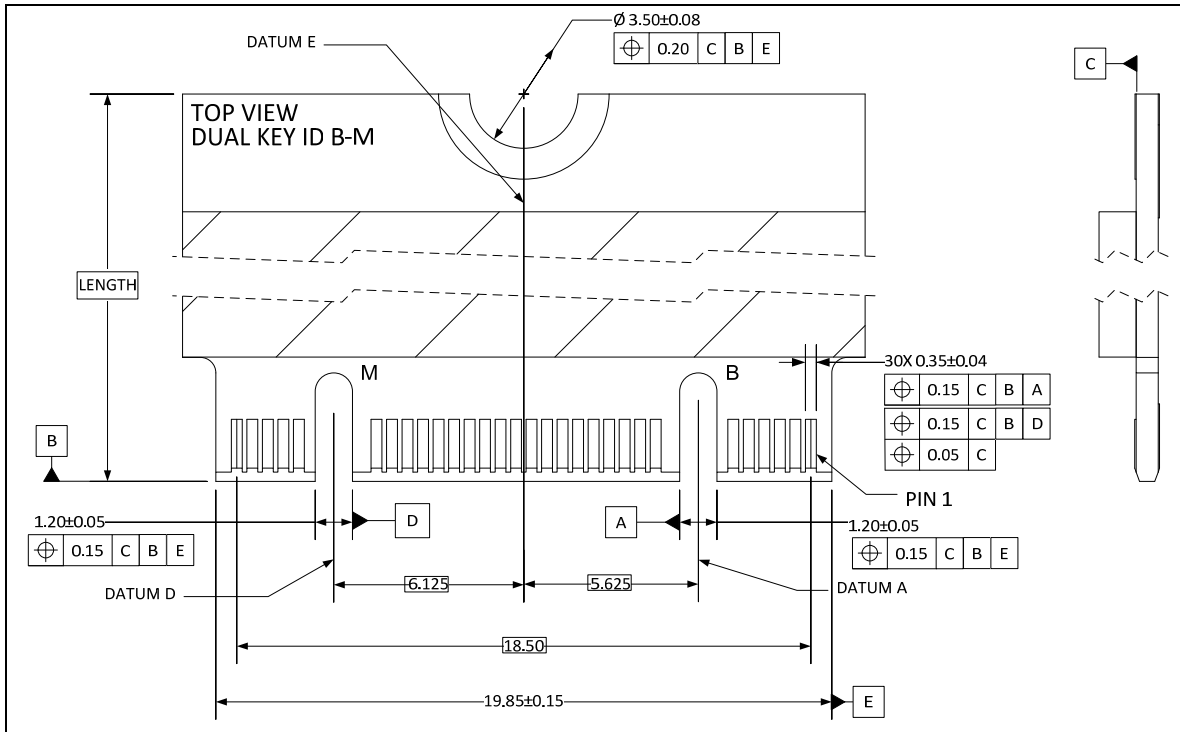


Figure 117 – M.2 keys (informative)

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 employ 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)

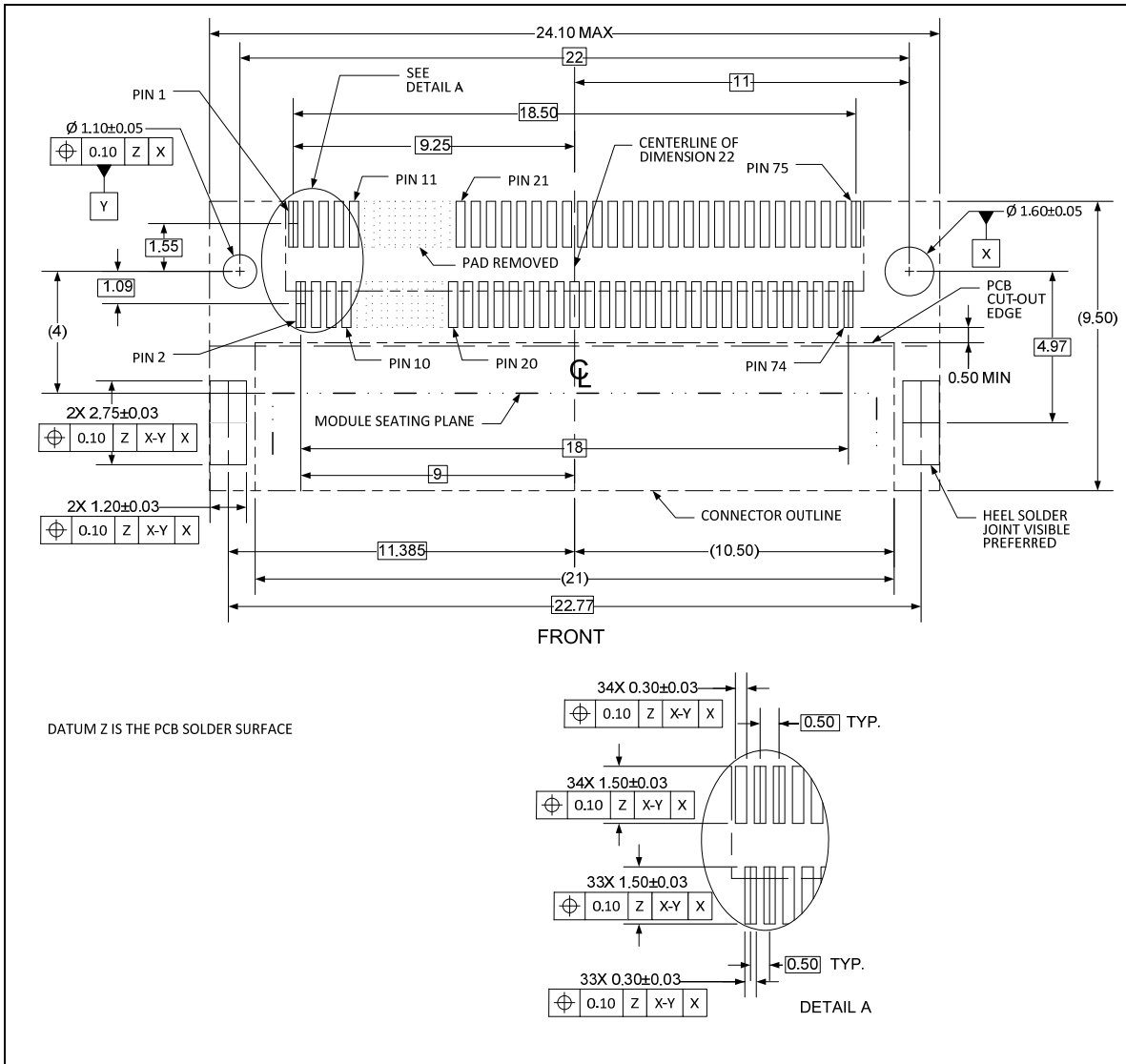


Figure 118 – M.2 land pattern for mother board (informative)

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

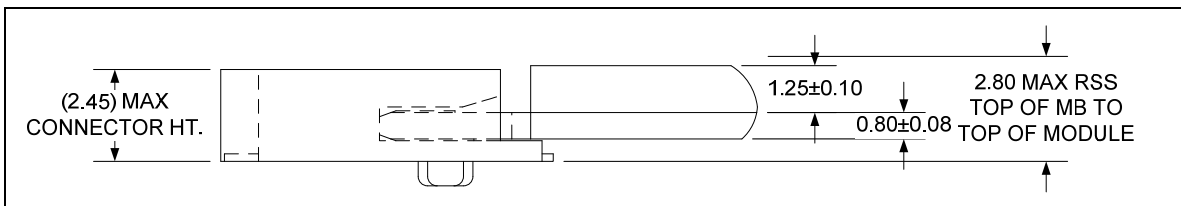


Figure 119 – M.2 single sided assembly – S2 profile (informative)



HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

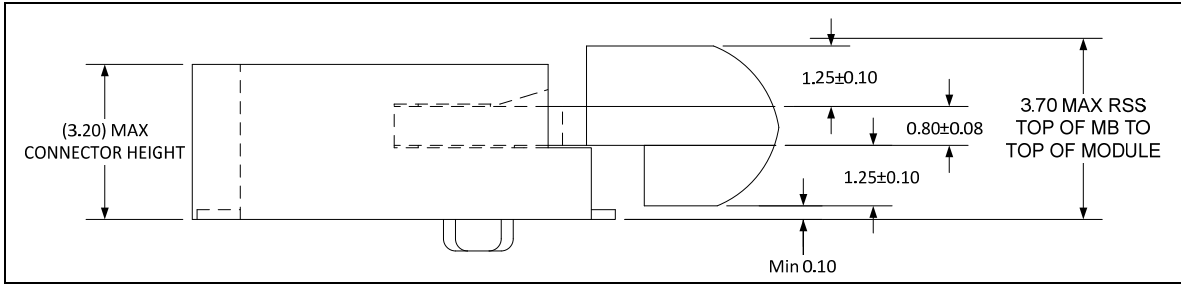


Figure 120 – M.2 double sided assembly – D2 (informative)

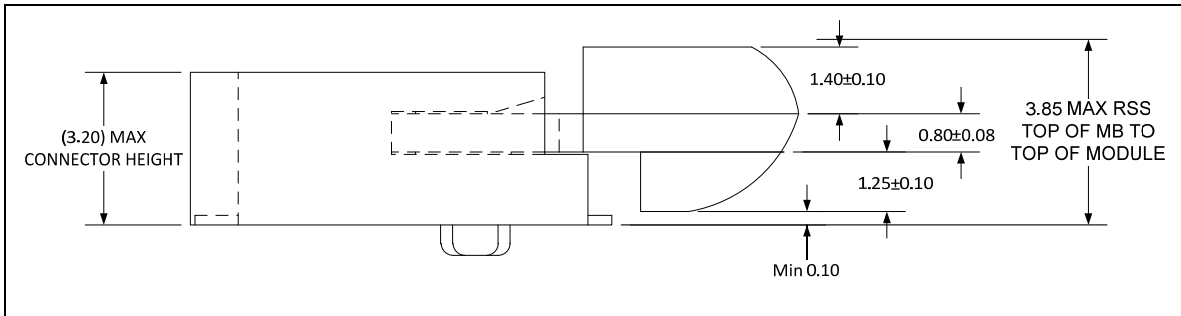


Figure 121 – M.2 double sided assembly – D3 (informative)

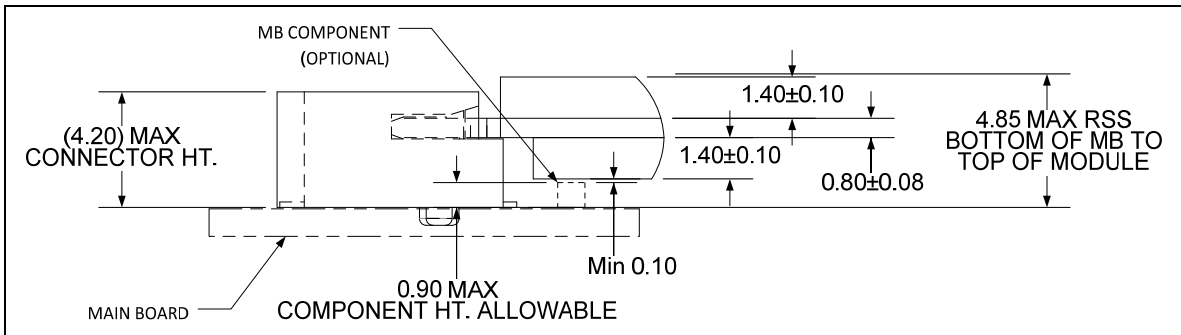


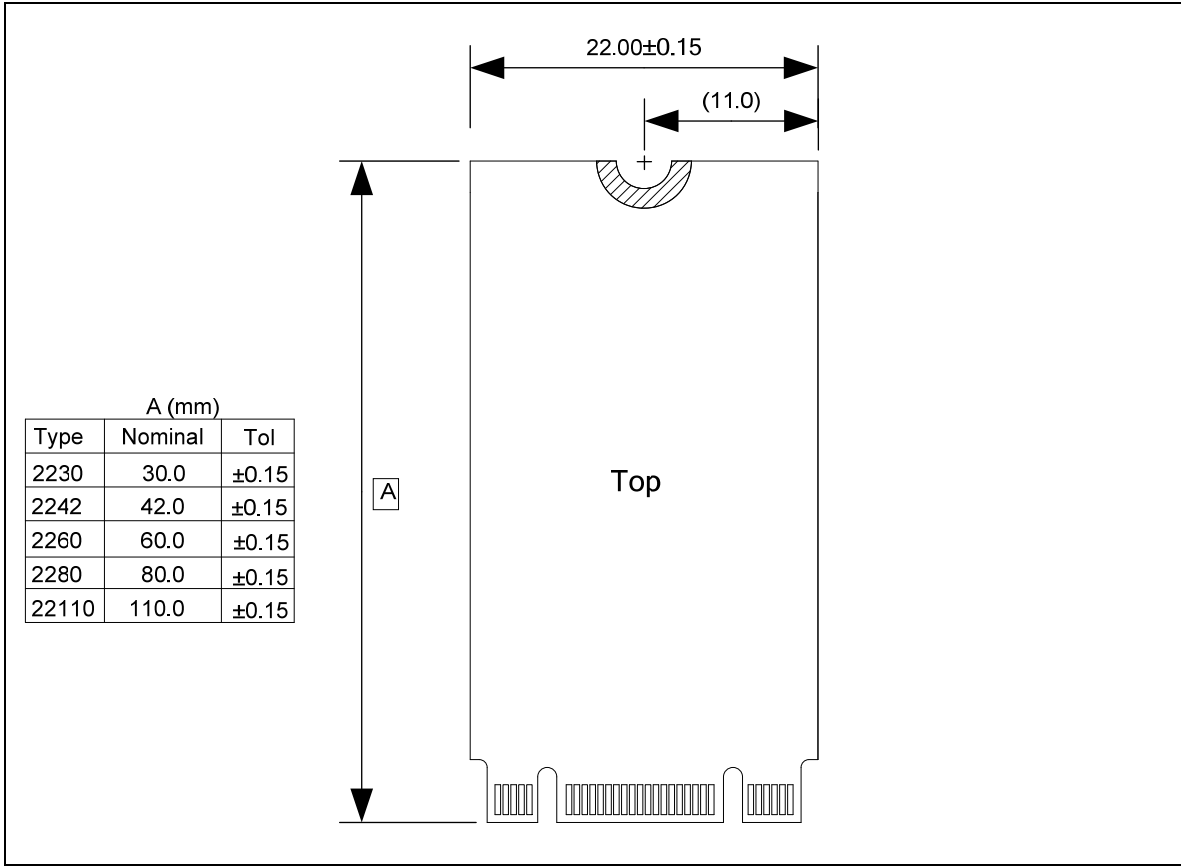
Figure 122 – M.2 double sided assembly – D5 (informative)

**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 123 (e.g., with both notch B and notch M present). All boards are nominally 22 mm wide.



**Figure 123 – 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 below. 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).

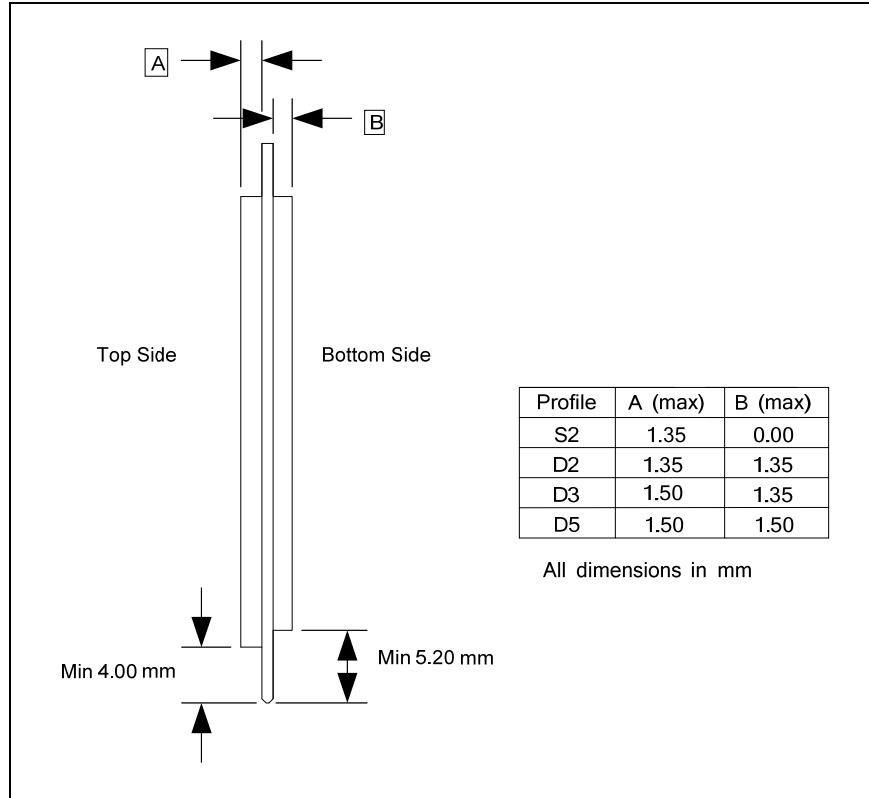


Figure 124 – M.2 component profile and keep out zone (informative)

### 6.9.10 M.2 signal integrity

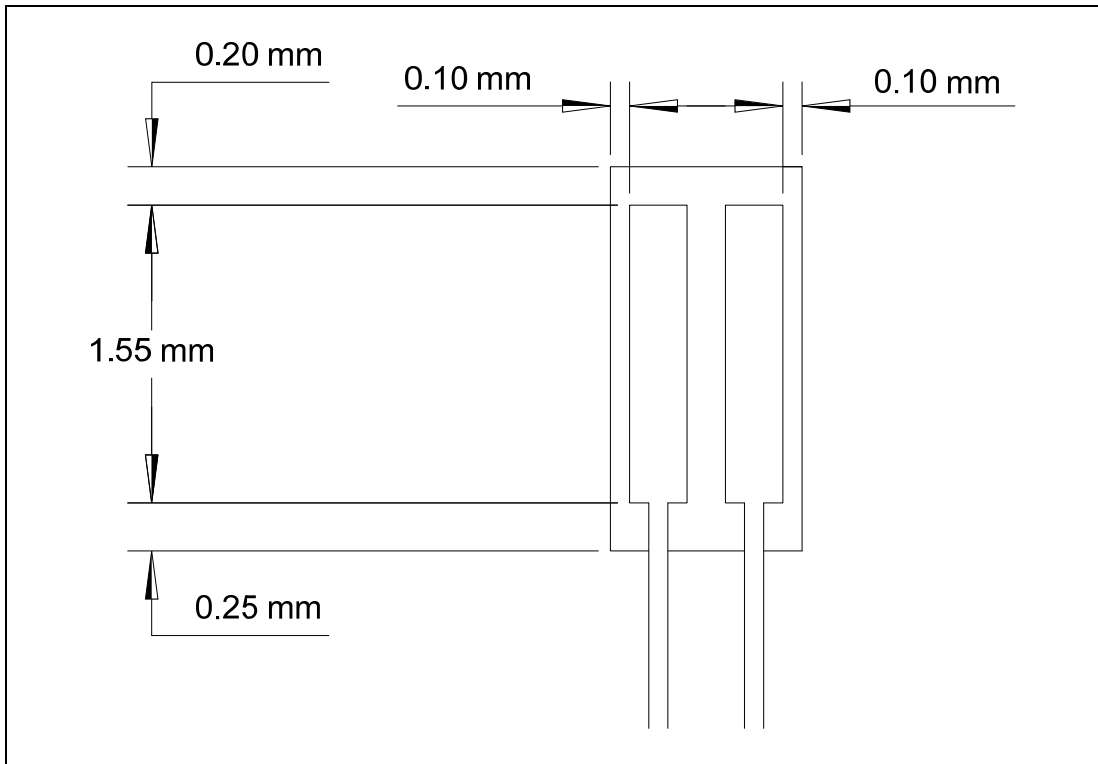
Table 23 – M.2 connector, electrical requirements

Parameter	Requirement
Differential impedance <sup>a</sup>	75 ohm to 95 ohm measured at 50 ps rise time, from the 20 % threshold to the 80 % threshold
Differential insertion loss <sup>a b</sup>	≥ -0.5 dB up to 4 GHz and then ≥ -1 dB up to 8 GHz
Differential Near End Crosstalk <sup>a b c</sup>	≤ -36 dB up to 4 GHz and then ≤ -32 dB up to 8 GHz
Differential Far End Crosstalk <sup>a b c</sup>	≤ -40 dB up to 4 GHz and then ≤ -32 dB up to 8 GHz

<sup>a</sup> Mated connector and module including solder pad and gold finger.  
<sup>b</sup> The result is referenced to 85 ohm differential impedance.  
<sup>c</sup> The crosstalk shall be pair to pair between any two differential pairs.

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

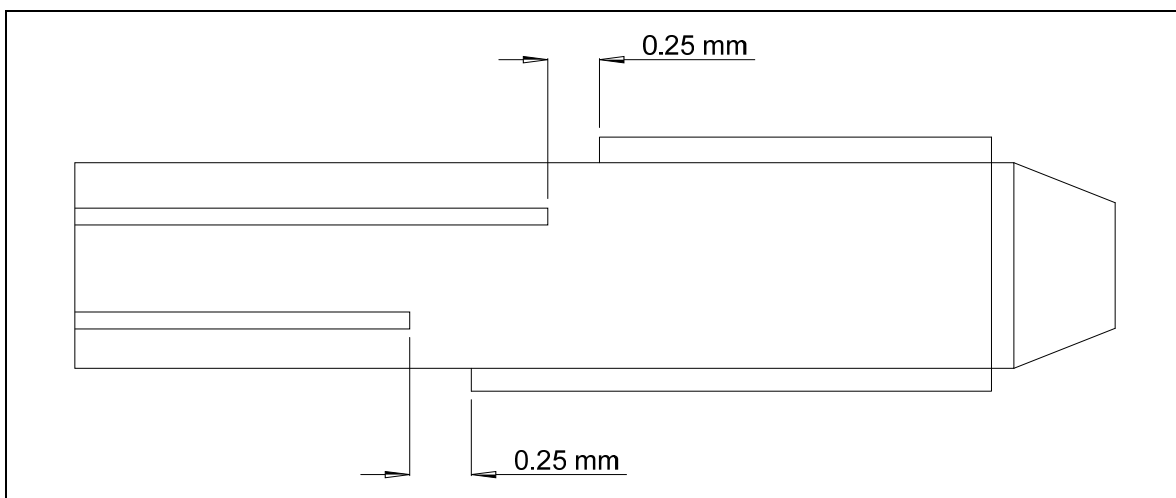
Voiding planes under the pads reduces launch capacitance that improves signal integrity. The values shown in Figure 125 are typical of a board constructed with mainstream commercial grade FR-4 PCB material. Dimension of recommended pad sizes are in previous figures.



**Figure 125 – 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.



**Figure 126 – 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)

<sup>a</sup> 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. <sup>a</sup>
57	GND	Ground
58	MFG2	Manufacturing pin. Use determined by vendor. <sup>a</sup>
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)

<sup>a</sup> 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

<sup>a</sup> 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. <sup>a</sup>
57	GND	Ground
58	MFG2	Manufacturing pin. Use determined by vendor. <sup>a</sup>
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)

<sup>a</sup> 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 <sup>c</sup>	Supply voltage	-	3.135	3.465	V
V <sub>IH</sub>	Input high voltage	-	2.0	3.6	V
V <sub>IL</sub> <sup>d</sup>	Input low voltage	-	-0.5	0.5	V
I <sub>OL</sub> <sup>a</sup>	Output low current for open-drain signals	0.4 V	4	-	mA
I <sub>IN</sub>	Input leakage current	0 V to 3.3 V	-10	+10	µA
I <sub>LKG</sub>	Output leakage current	0 V to 3.3 V	-50	+50	µA
C <sub>IN</sub>	Input pin capacitance	-	-	7	pF
C <sub>OUT</sub>	Output pin capacitance	-	-	30	pF
T <sub>RISE</sub> <sup>b</sup>	Power up time	-	-	10	ms
<sup>a</sup> Not applicable to LED# and DAS/DSS pins. <sup>b</sup> Rise time of 3.3 V supply from 0 V to its minimum voltage under maximum load. <sup>c</sup> Each pin shall be capable of supplying at least 500 mA. <sup>d</sup> 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 ps 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**

### **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 127 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.

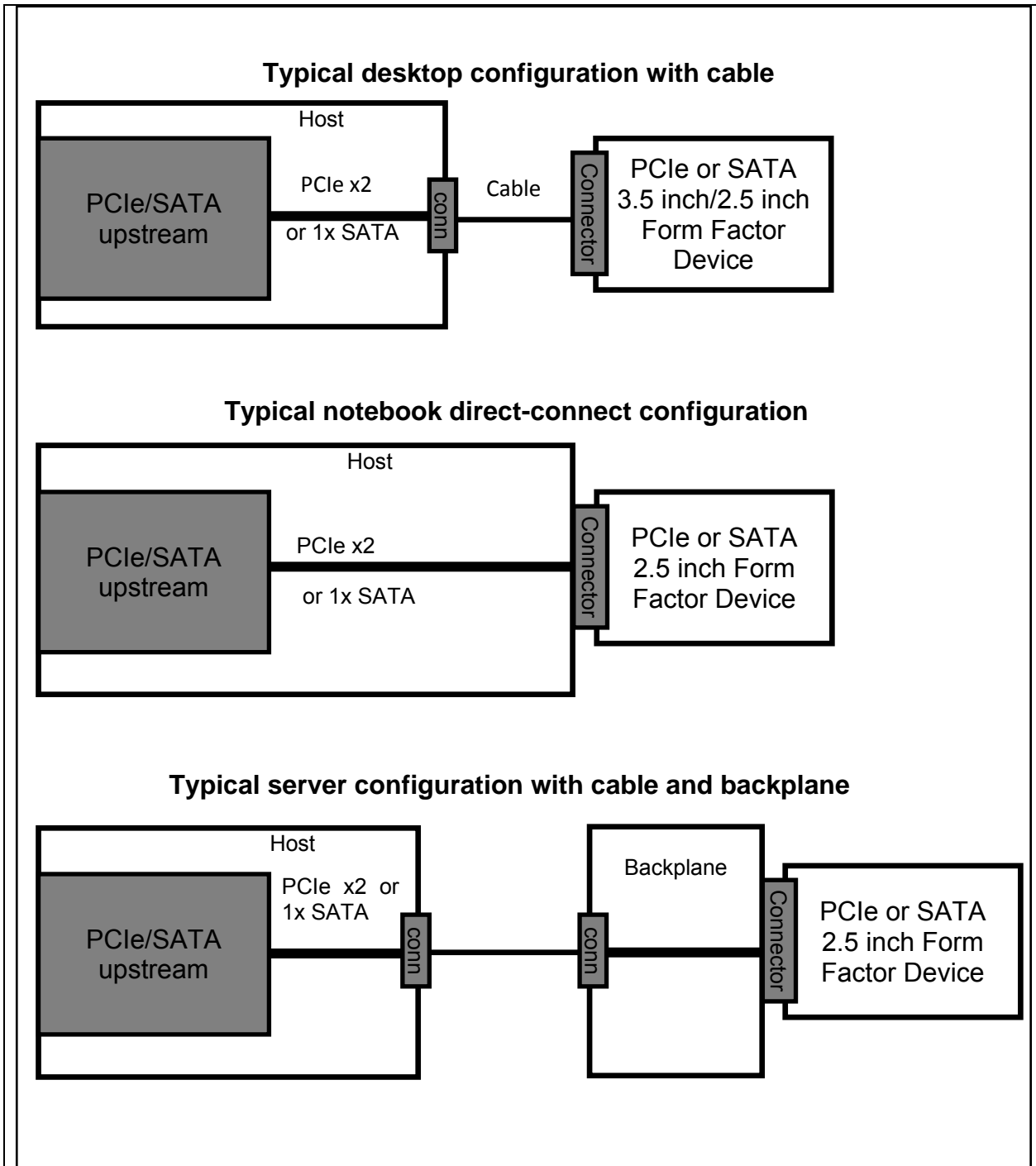


Figure 127 – Example configurations for client

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 128 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

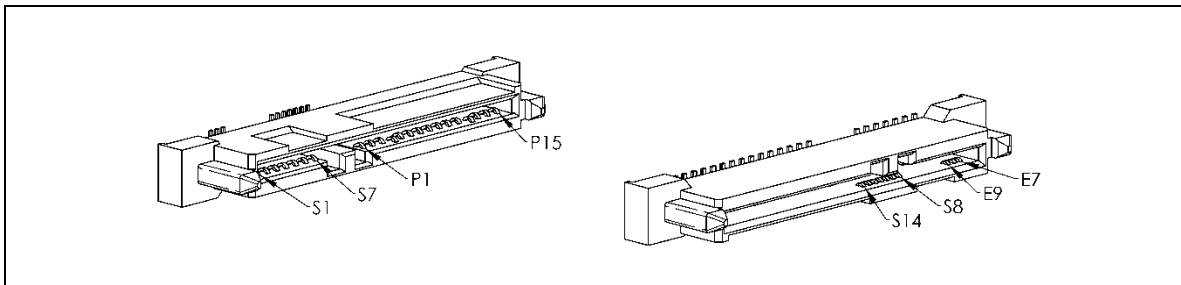
**Table 29 – SATA Express pin group table (for host receptacle and device plug connectors)**

(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 128 – SATA Express pinout for host receptacle connectors**

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

Pin position	Name <sup>a</sup>	Signal Description <sup>a</sup>	Usage	Signal Direction	Mating Sequence <sup>c</sup>
S1	GND	Ground	PCIe/SATA		Second
S2	PETp0/A+	PCIe Lane 0 Host Tx / SATA Tx pair <sup>b</sup>	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 <sup>b</sup>	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

<sup>a</sup> The "Name" and "Signal Description" columns are relative to the host, as defined by each protocol specification.

<sup>b</sup> 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.

<sup>c</sup> 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 <sup>a</sup>	Signal Description <sup>a</sup>	Usage	Signal Direction	Mating Sequence <sup>c</sup>
P1	Reserved <sup>d</sup>	Reserved for future use	PCIe/SATA		Third
P2	PERST# <sup>d</sup>	PCIe reset	PCIe only	Host Output	Second
P3	CLKREQ# / DEVSLP <sup>d</sup>	L1 PM substate / Device sleep	PCIe/SATA	Host Output	Third
P4	IFDet <sup>e</sup>	Interface (PCIe/SATA) detect	PCIe/SATA	Input to Host	First
P5	GND	Ground	PCIe/SATA		Second
P6	GND	Ground	PCIe/SATA		Second
P7	V5 <sup>b</sup>	5 V power, pre-charge	PCIe/SATA	Host Output	Second
P8	V5 <sup>b</sup>		PCIe/SATA	Host Output	Third
P9	V5 <sup>b</sup>		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 <sup>b</sup>	12 V power, pre-charge	PCIe/SATA	Host Output	Second
P14	V12 <sup>b</sup>	12 V power	PCIe/SATA	Host Output	Third
P15	V12 <sup>b</sup>	12 V power	PCIe/SATA	Host Output	Third
E7	RefClk+ <sup>g</sup>	PCIe common RefClk	PCIe only, optional <sup>f</sup>	Host Output	Third
E8	RefClk- <sup>g</sup>				
E9	ClkDet <sup>h</sup>	PCIe RefClk detect	PCIe only, optional	Input to Host	Second

<sup>a</sup> The "Name" and "Signal Description" columns are relative to the host, as defined by each protocol specification.

<sup>b</sup> 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.

<sup>c</sup> 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.

<sup>d</sup> 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.

<sup>e</sup> 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.

<sup>f</sup> 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.

<sup>g</sup> 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.

<sup>h</sup> 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 129:

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

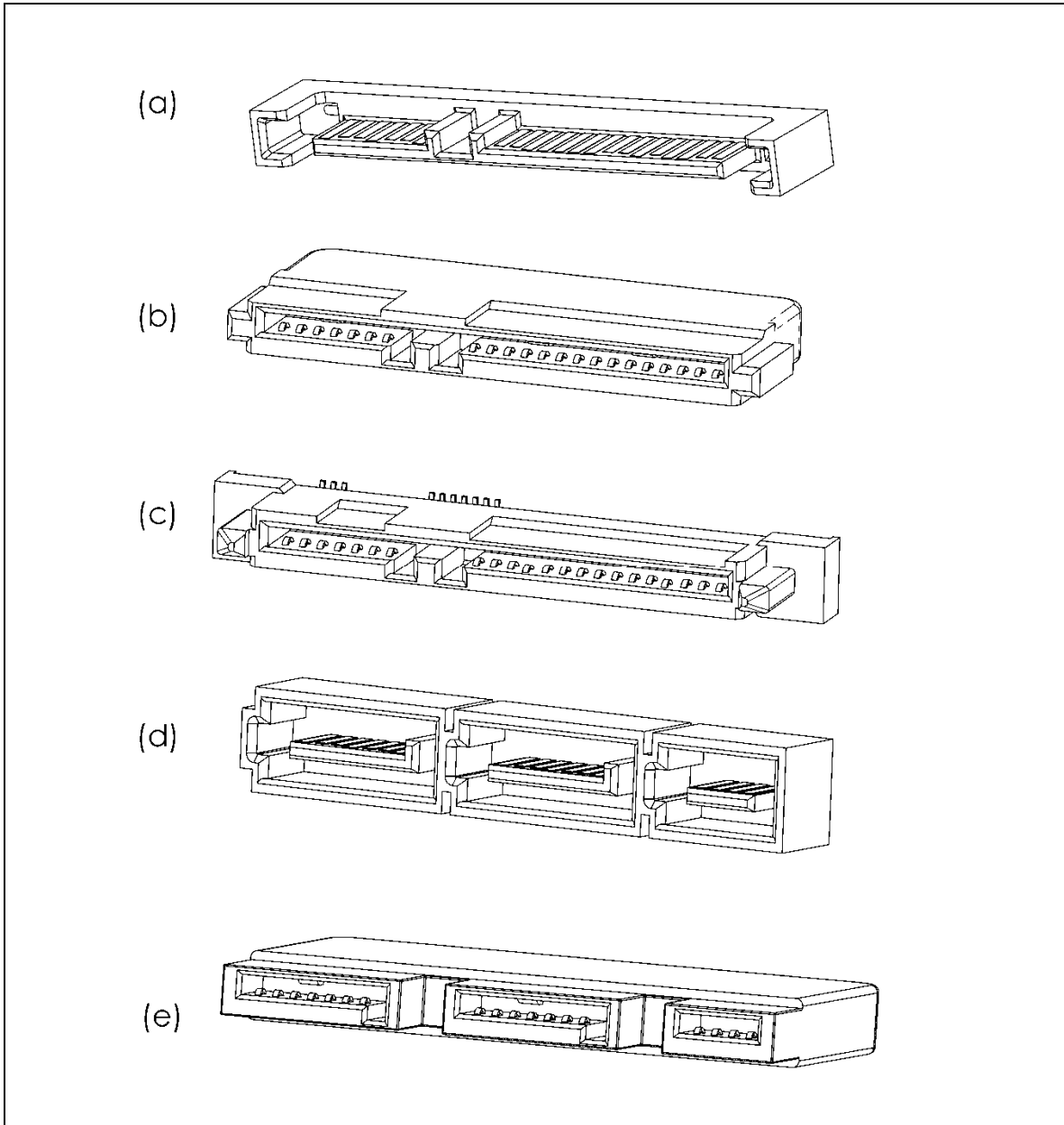


Figure 129 – Examples of SATA Express connectors

### 6.10.7 SATA Express device plug connector

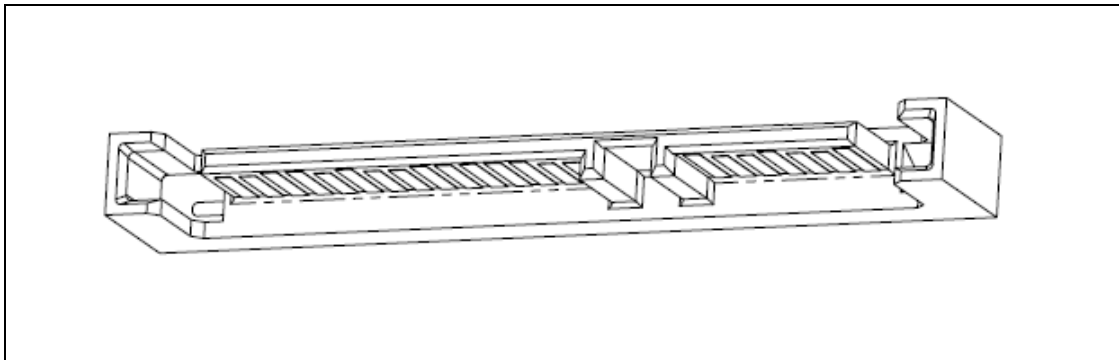
The SATA Express device plug connector is physically similar to the standard SAS device connector defined in SFF-8680. The cable retention feature requirements are unchanged, the keying is different, and optional pins are added (i.e., E7, E8, and E9) for applications requiring RefClk.

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 130 shows an isometric view of the SATA Express device plug connector and Figure 131 defines the SATA Express device plug connector mating interface.



**Figure 130 – SATA Express device plug connector isometric drawing**



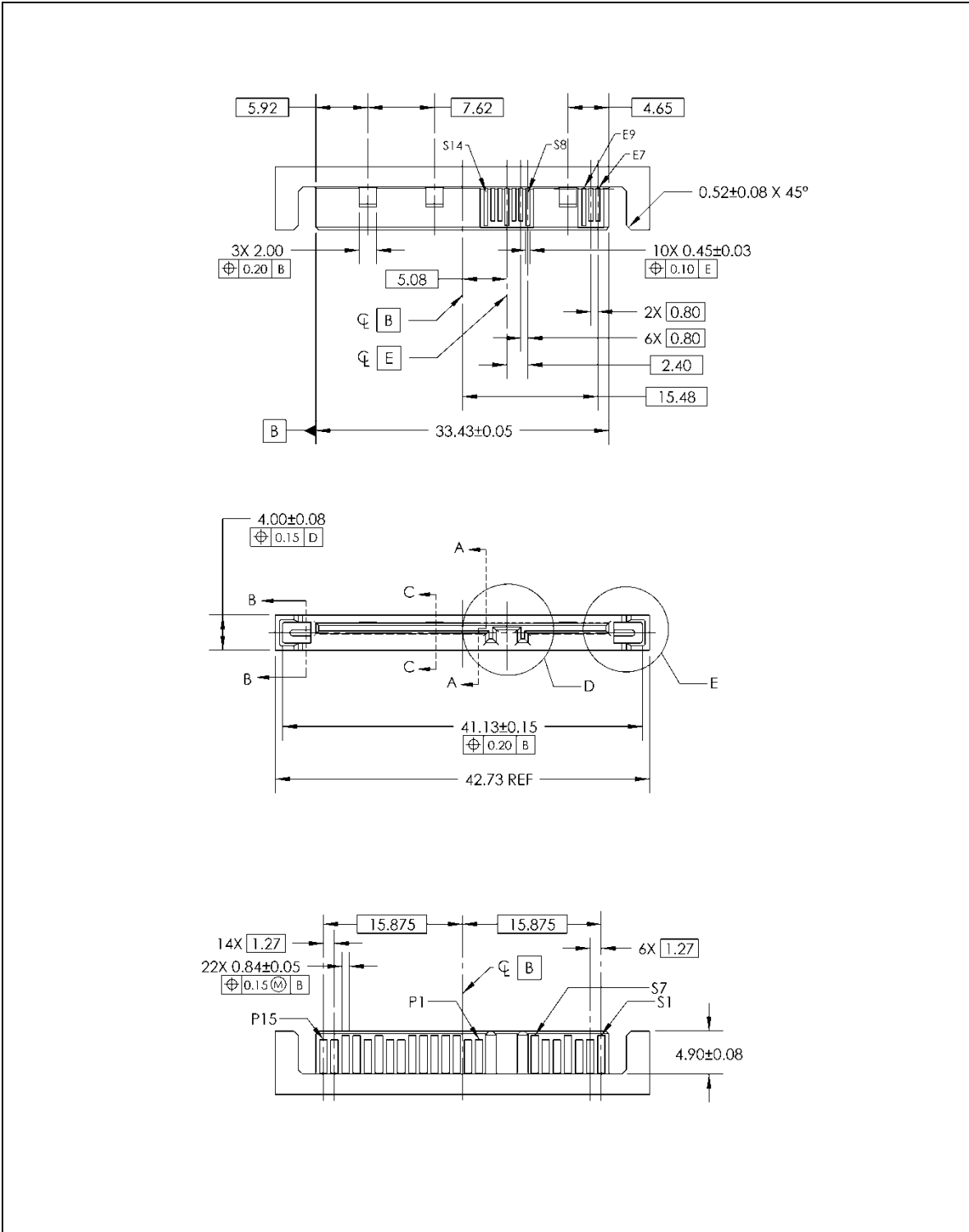


Figure 131 – SATA Express device plug connector drawing

(part 1 of 2)

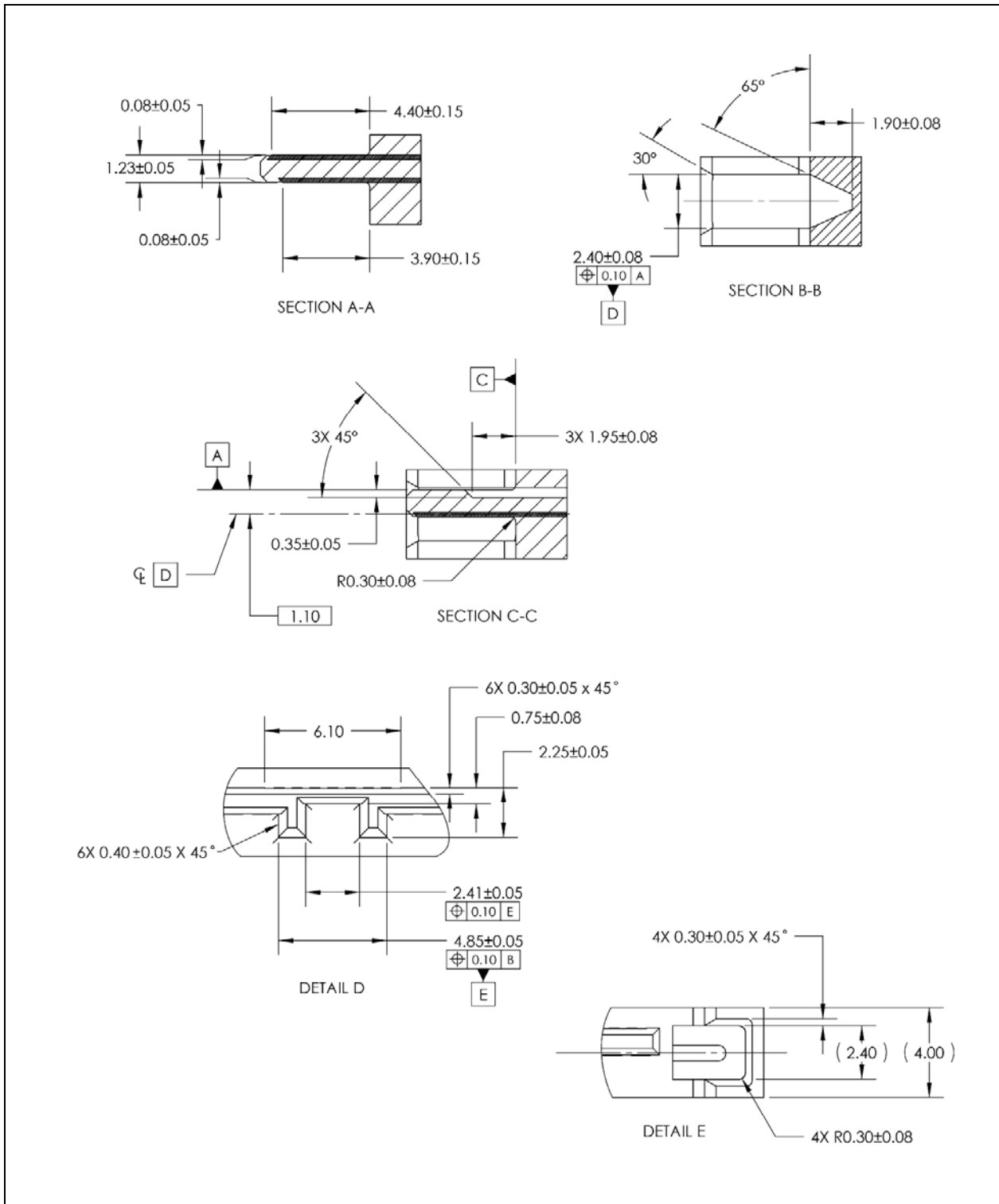


Figure 131 – SATA Express device plug connector drawing (part 2 of 2)

### 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

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

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 132 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.

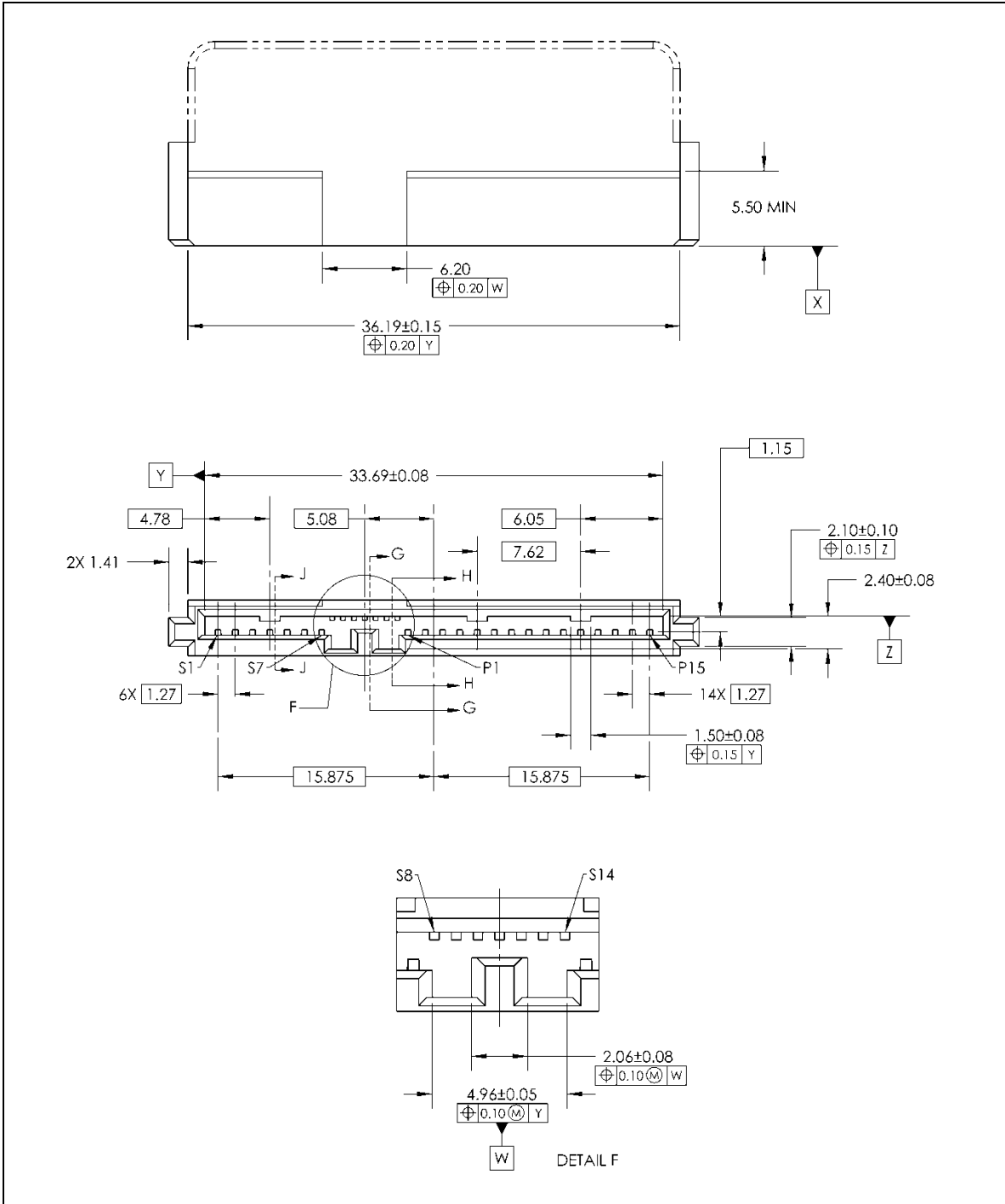


Figure 132 – SATA Express device cable receptacle connector drawing

(part 1 of 2)

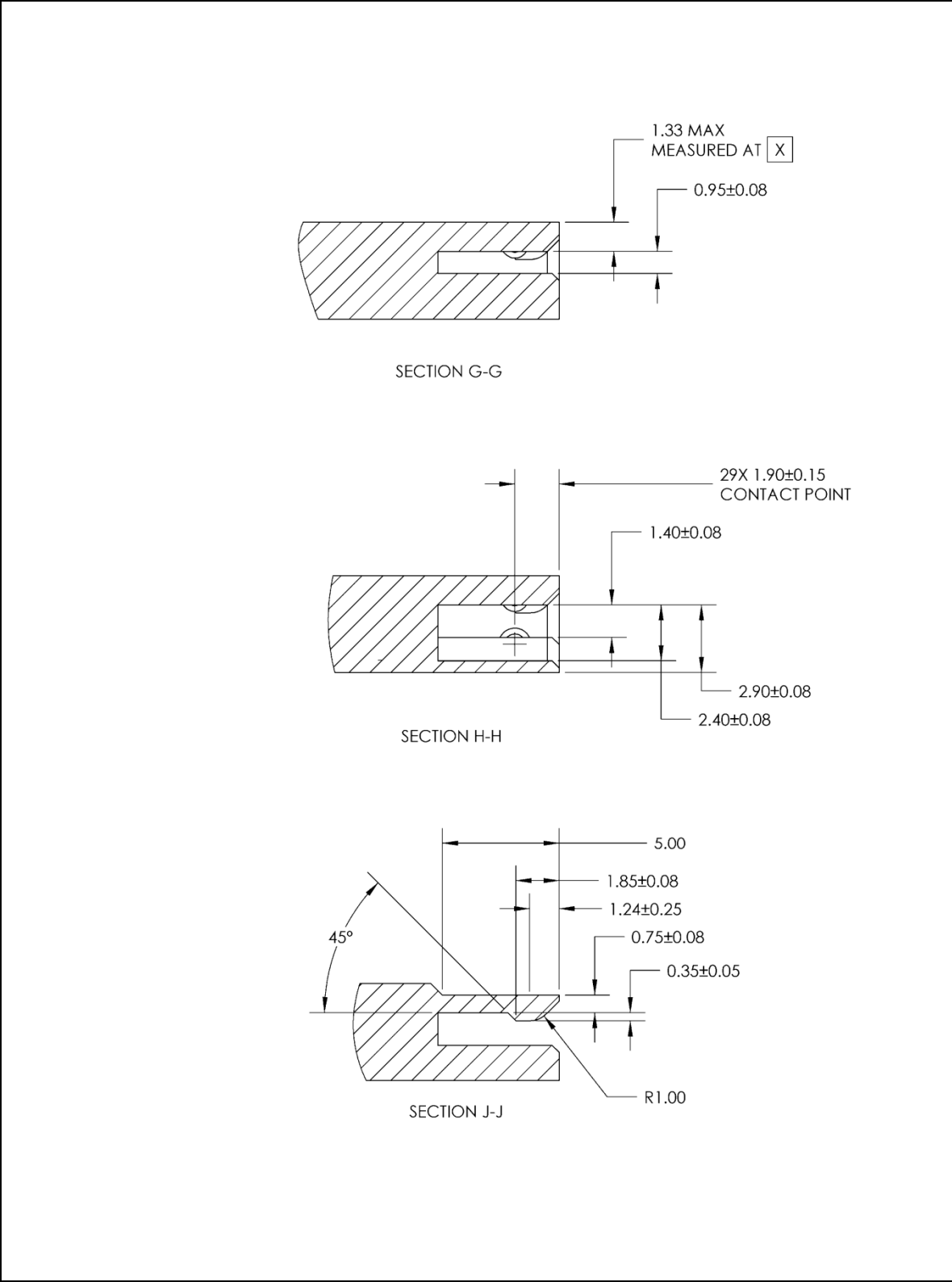


Figure 132 – SATA Express device cable receptacle connector drawing (part 2 of 2)

### 6.10.9 SATA Express host receptacle connector

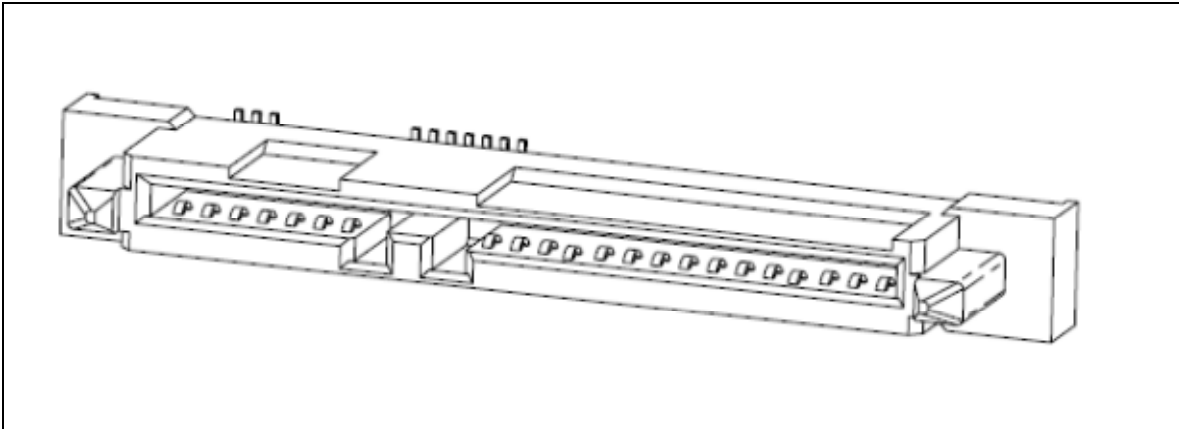
The SATA Express host receptacle connector is identical to SATA Express device cable receptacle connector except for the following differences:

- 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 132).

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

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

Figure 133 shows an isometric view of the SATA Express host receptacle connector and Figure 134 defines the SATA Express host receptacle connector mating interface. Note that Figure 134 does not include all the necessary dimensions or section views to complete the interface definition. Refer to Figure 132 for completeness.



**Figure 133 – SATA Express host receptacle connector isometric drawing**

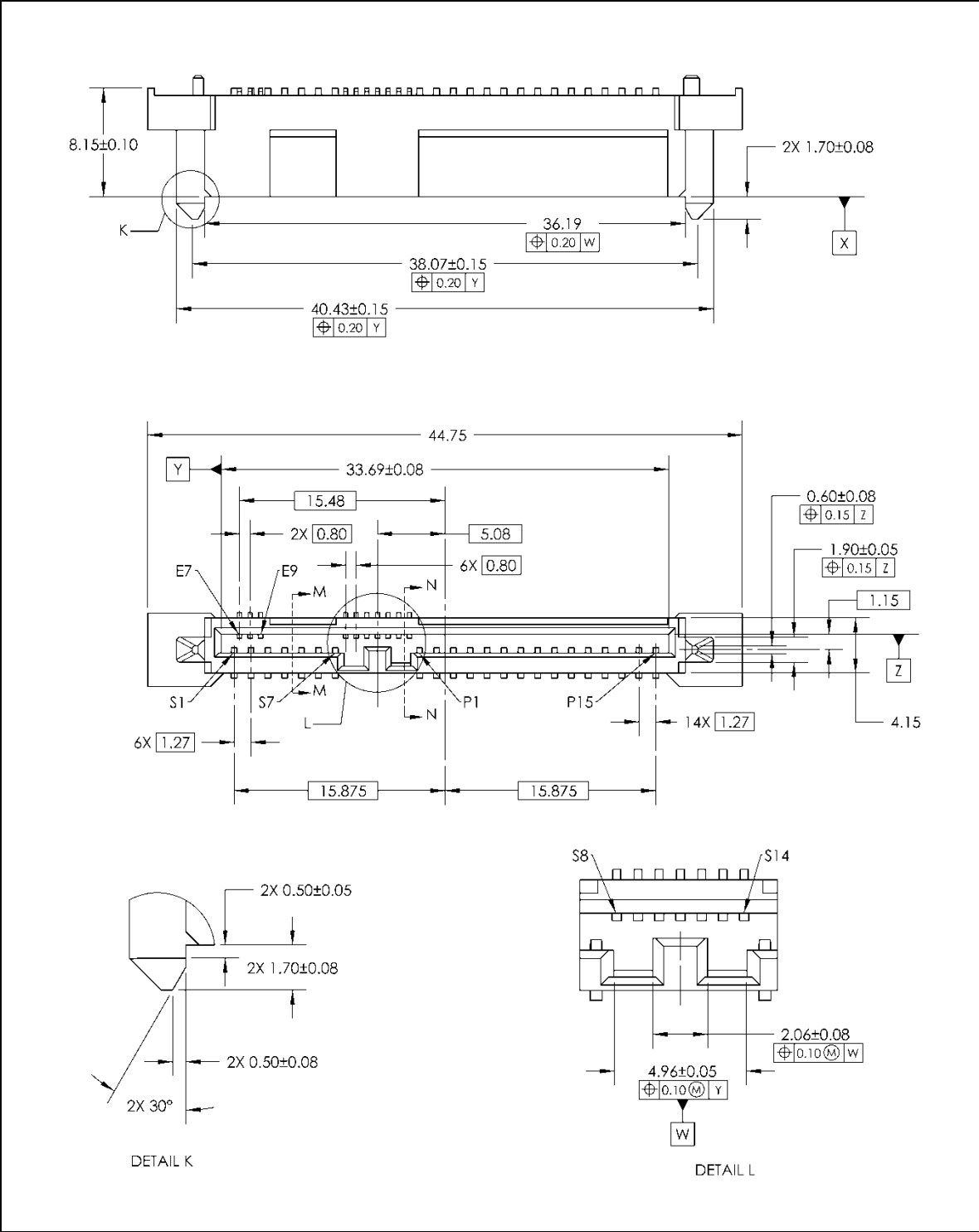
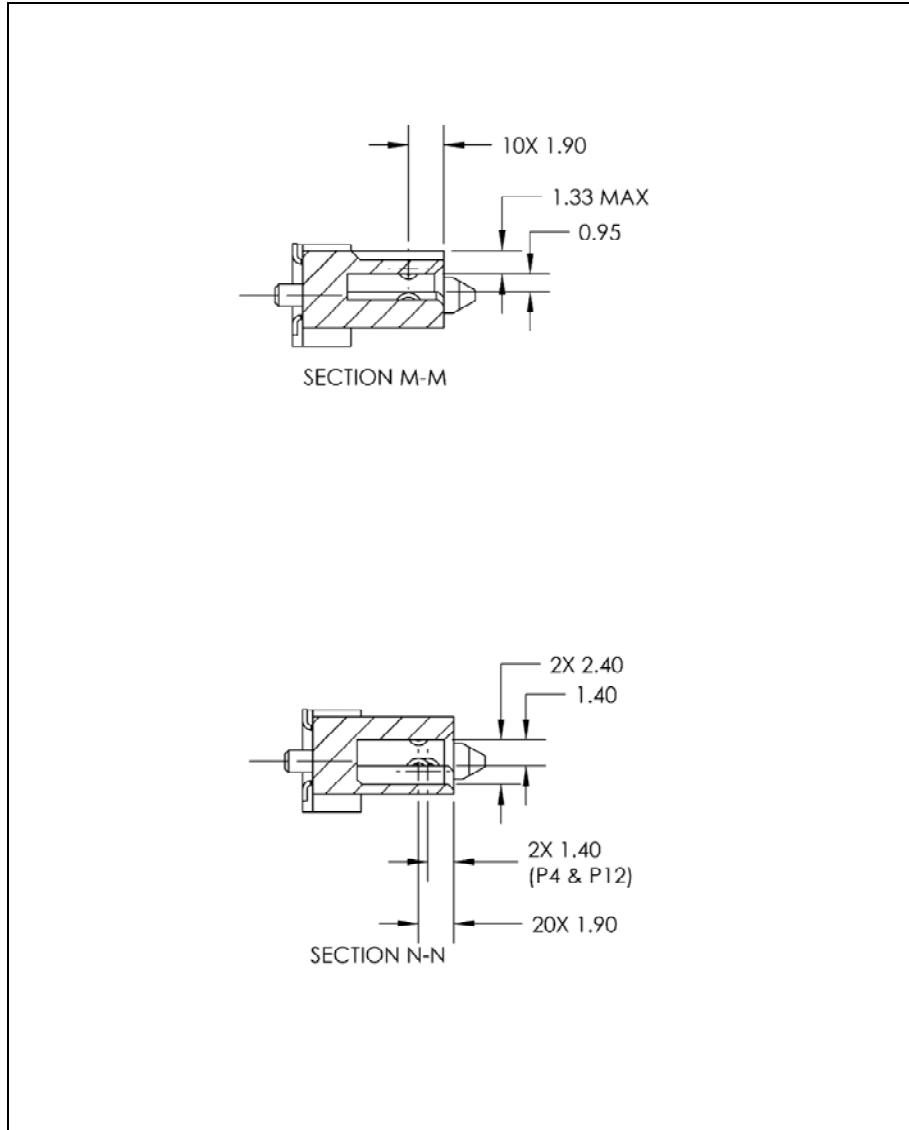


Figure 134 – SATA Express host receptacle connector drawing

(part 1 of 2)



**Figure 134 – 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 135 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 135 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 136 and Figure 137 are the SATA Express host plug connector mechanical drawings.

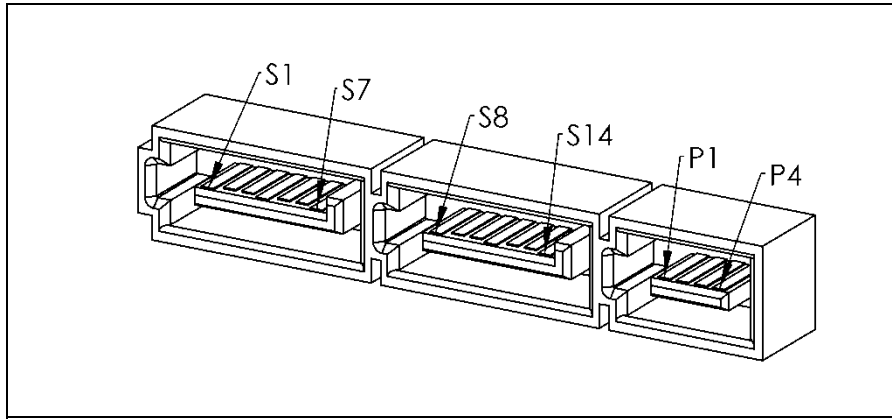


Figure 135 – SATA Express host plug connector pinout (isometric view)

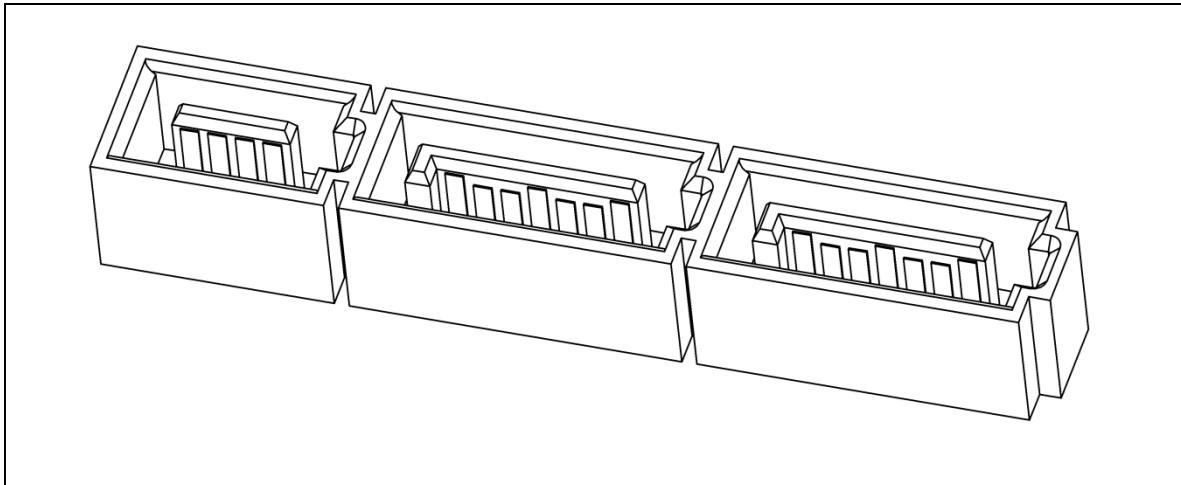
Table 32 – SATA Express host plug connector pin list

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		Second
P2	PERST#	PCIe reset	PCIe only	Host Output	Second
P3	CLKREQ# / DEVSLP	PCIe L1 PM Substate / SATA device sleep	PCIe/SATA		Second
P4	IFDet	Interface detect	PCIe/SATA	Input to Host	First

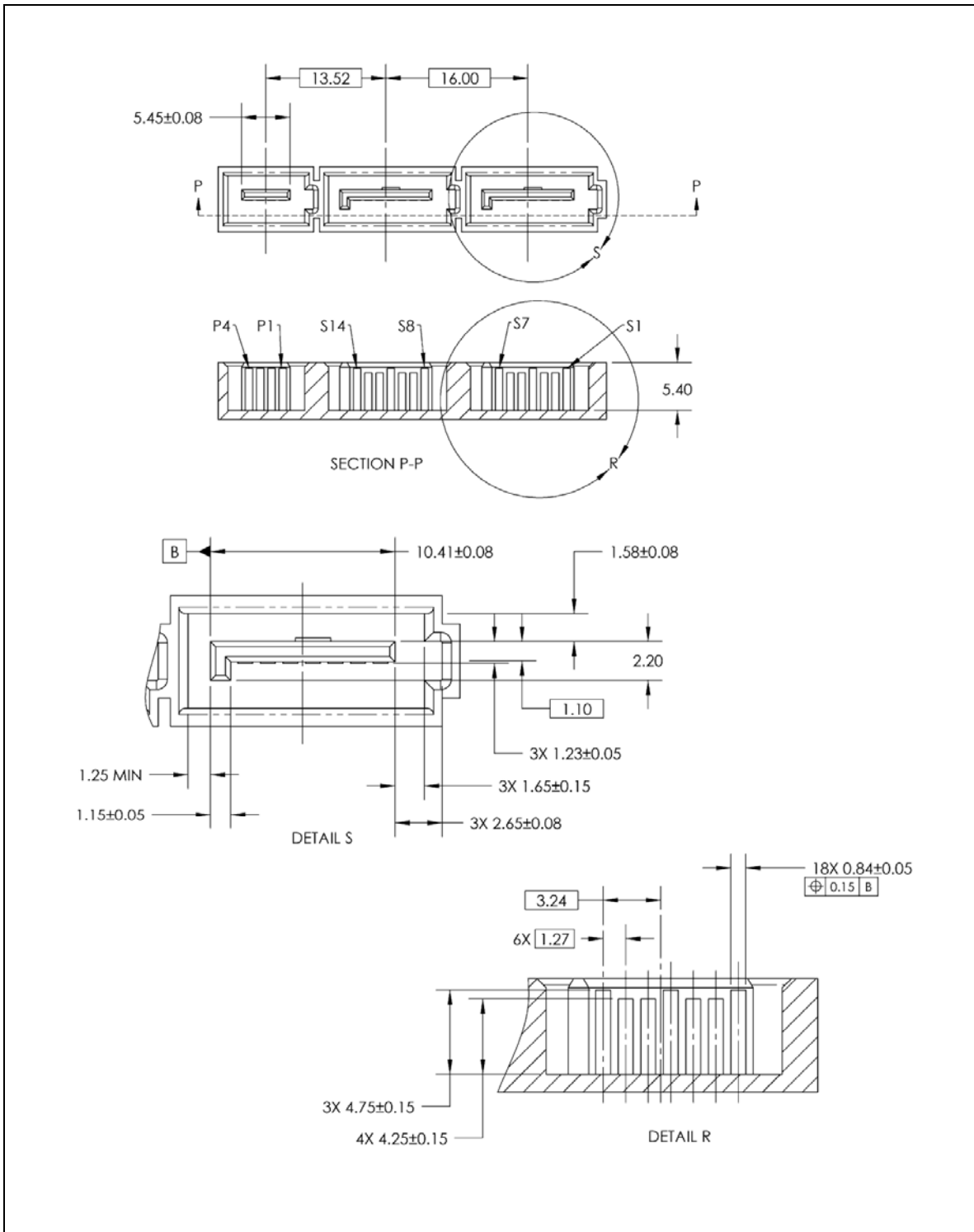


**Table 33 – SATA Express cable wire connection**

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 136 – SATA Express host plug connector isometric drawing**



**Figure 137 – 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 138 shows an isometric view of the SATA Express host cable receptacle

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

connector. Figure 139 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 139:

- 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).

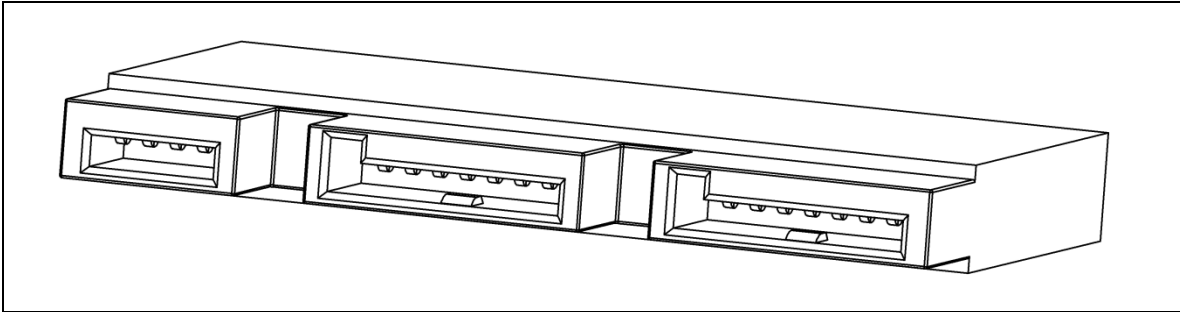


Figure 138 – SATA Express host cable receptacle connector isometric drawing

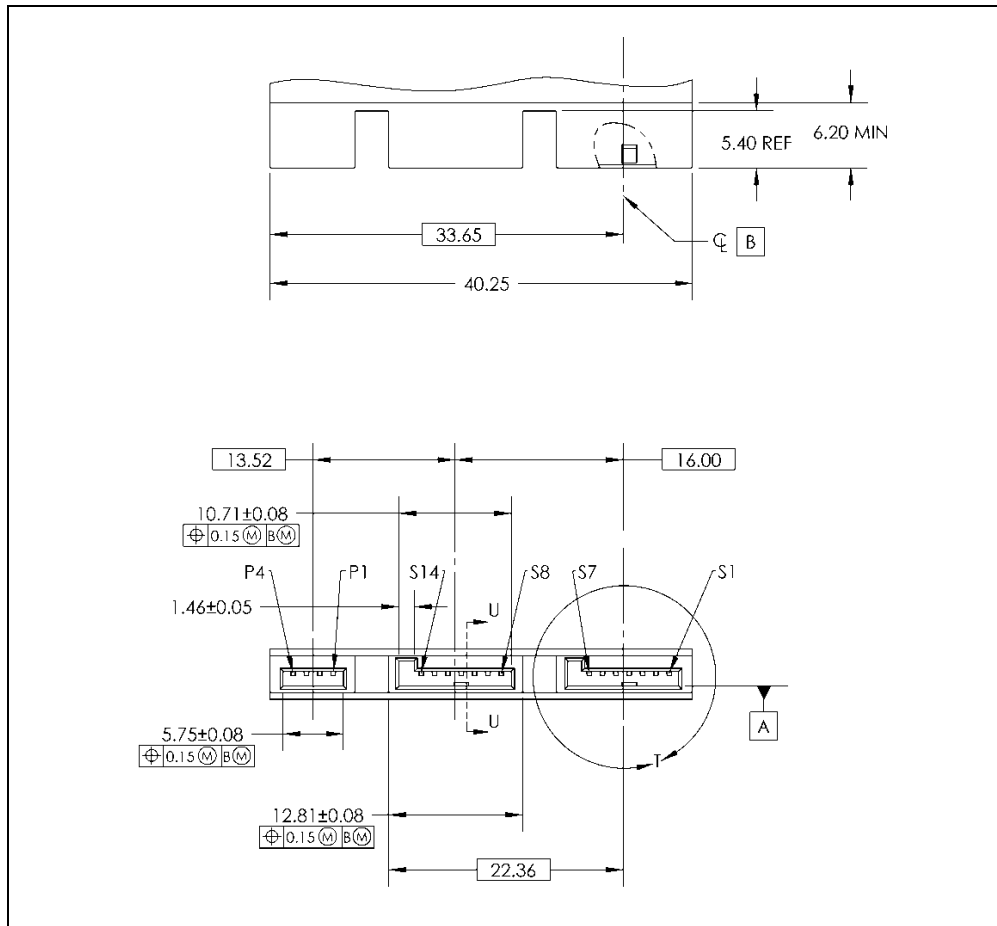
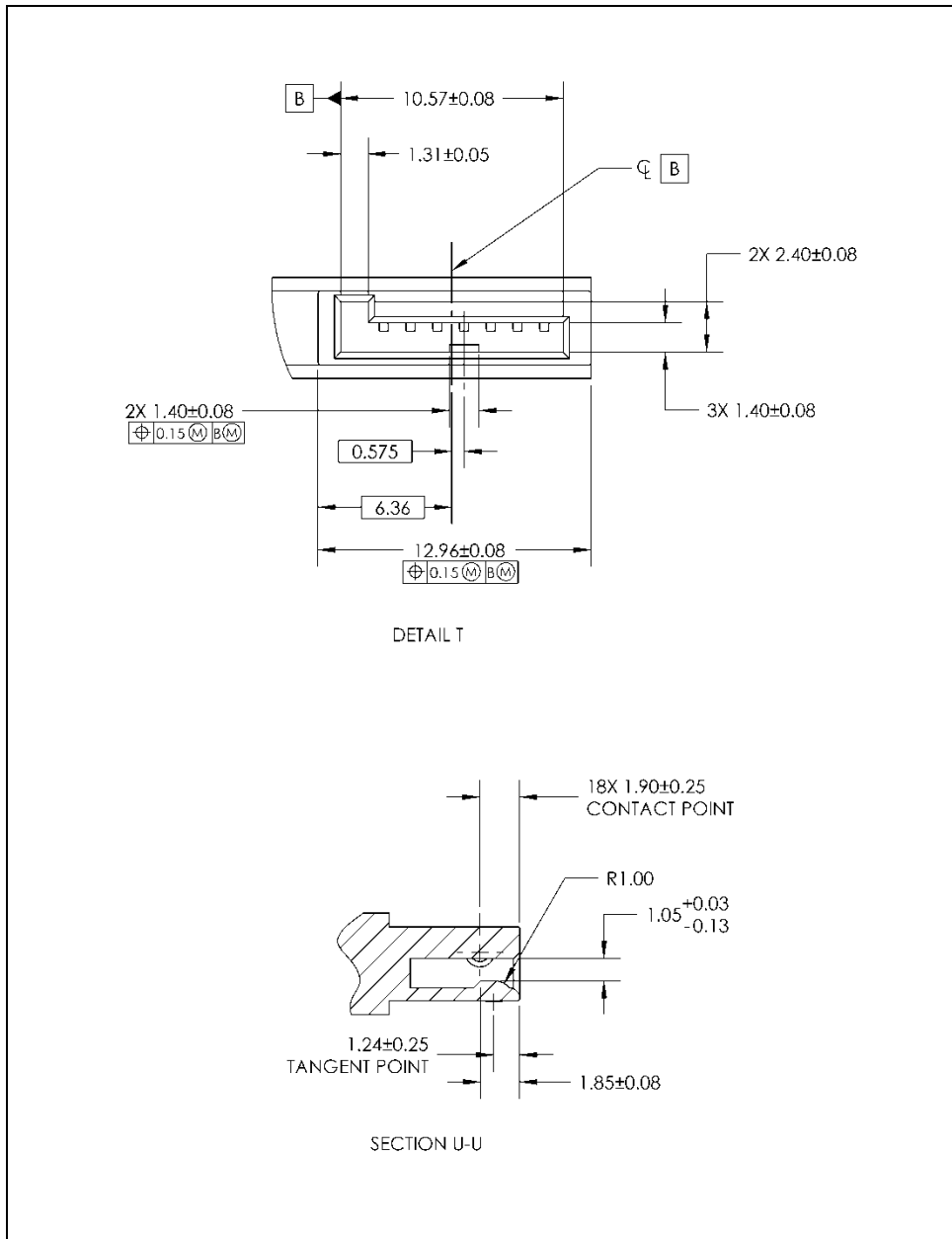


Figure 139 – SATA Express host cable receptacle connector drawing

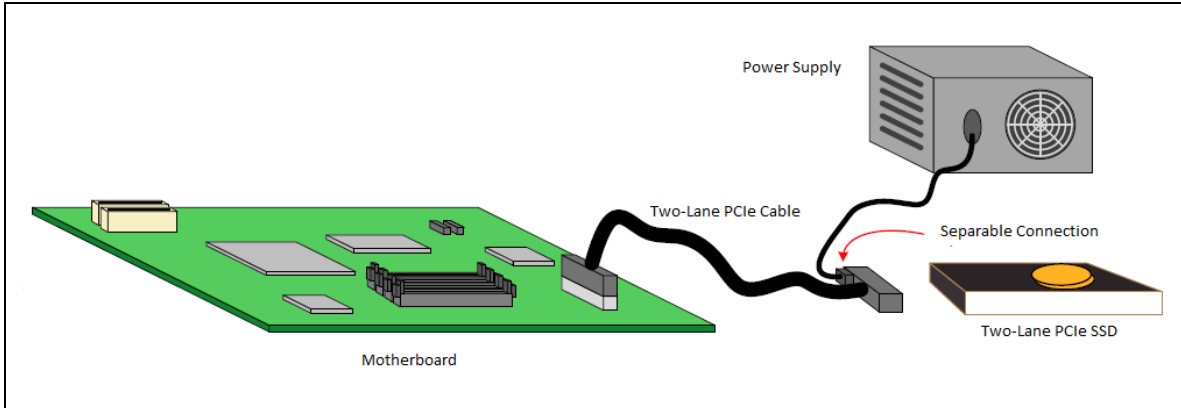
(part 1 of 2)



**Figure 139 – SATA Express host cable receptacle connector drawing (part 2 of 2)**

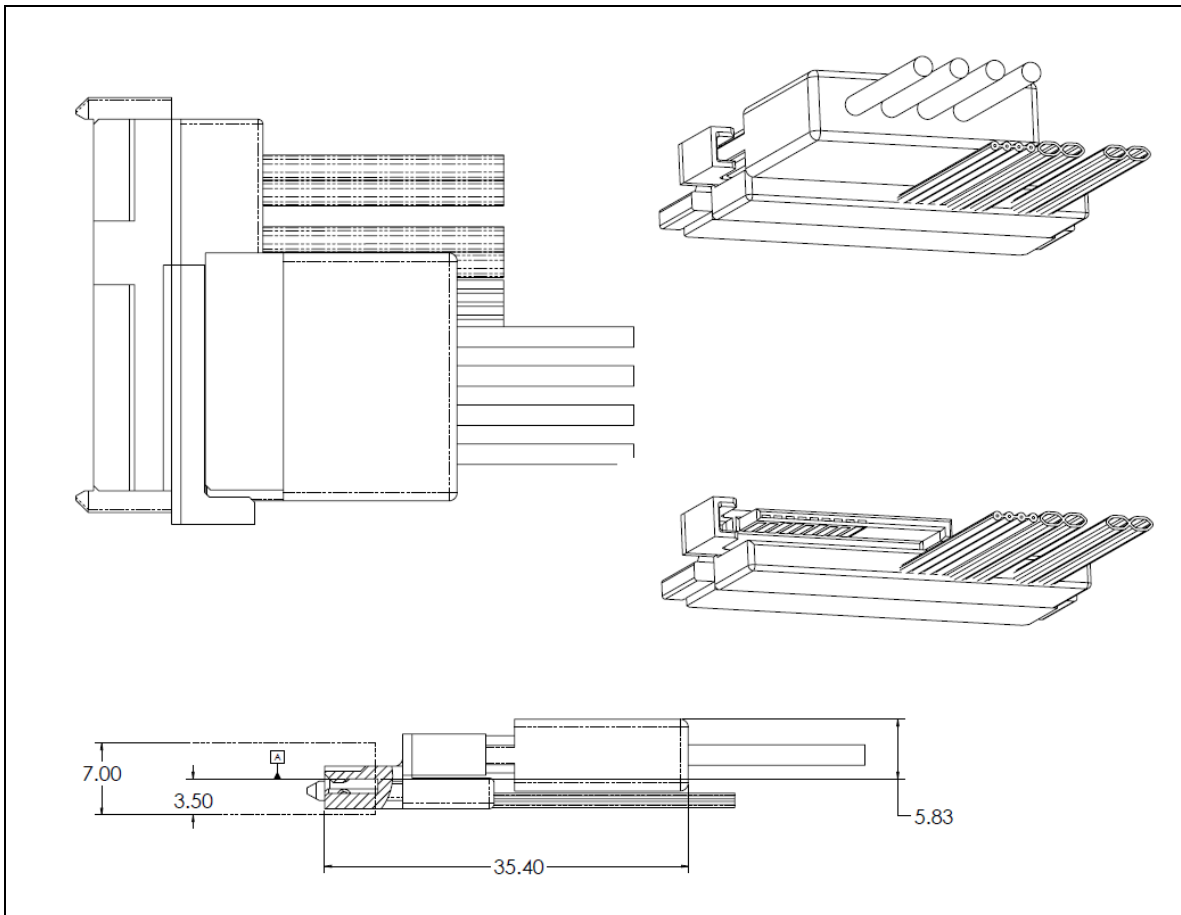
### 6.10.12 Power dongle connector

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 140.



**Figure 140 – Example of power dangle usage**

Implementation of the power dangle on the SATA Express device cable receptacle connector is vendor specific. The example in Figure 141 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 142 provides a pigtail power dangle from the SATA Express device cable receptacle connector. Implementation is not limited to the configurations illustrated in these two examples.



**Figure 141 – Example power dangle to be used with the 15 pin SATA power connector**

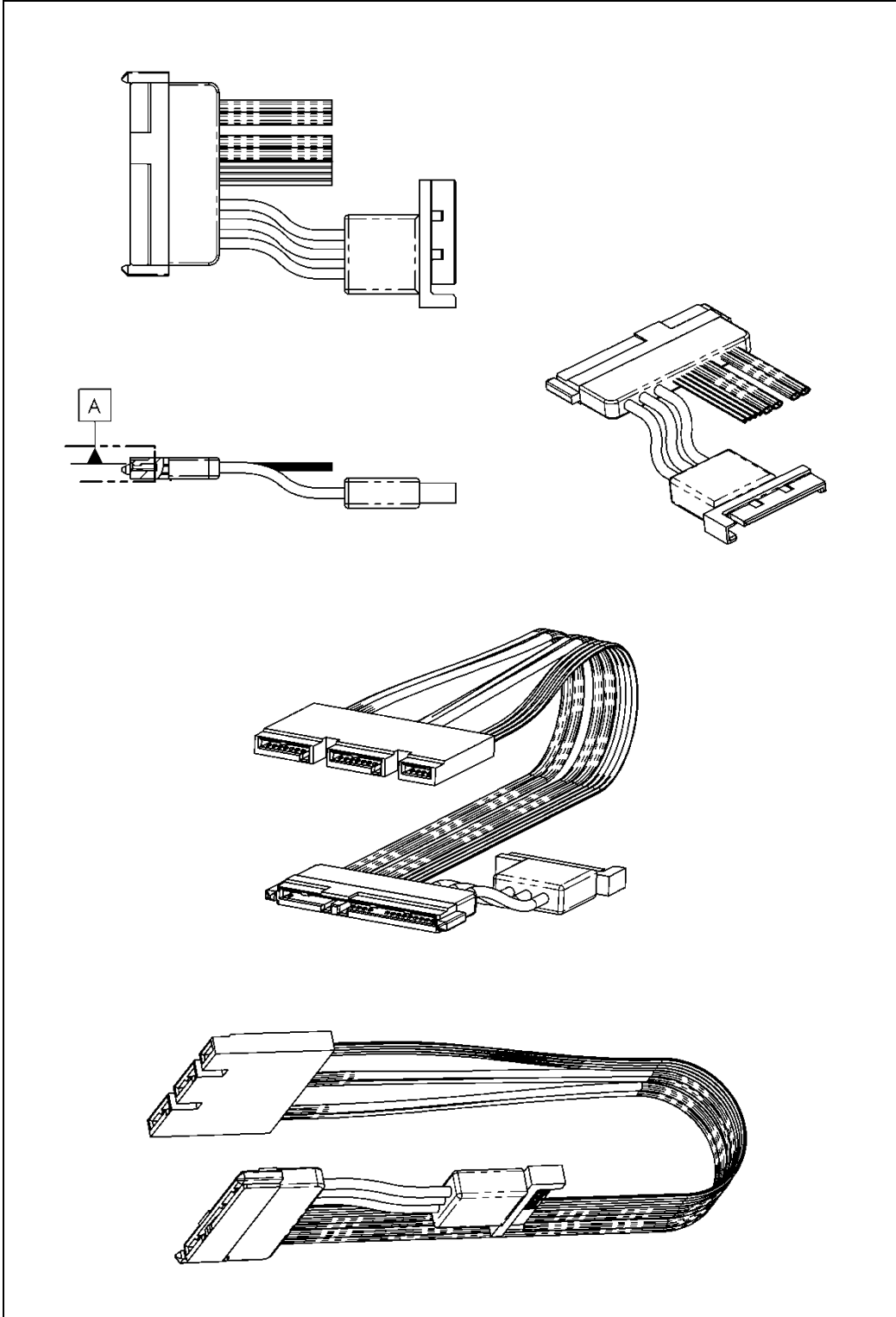


Figure 142 – Example power pigtail to be used with the 15 pin SATA power connector

### 6.10.13 SATA Express connector intermateability summary

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

**Table 34 – SATA Express connector intermateability summary**

Connector	Intermateable with
SATA Express Host Receptacle Connector	SATA Express device plug connector (SATA spec) <sup>a</sup> SATA device plug connector (SATA spec) <sup>a</sup>
SATA Express Device Plug Connector	SATA Express host receptacle connector (SATA spec) <sup>a</sup> SFF-8639 backplane receptacle connector <sup>b</sup> SATA Express device cable receptacle connector (SATA spec) <sup>a</sup> SAS/SAS MultiLink backplane and cable receptacle connector <sup>c</sup>
SATA Express Host Plug Connector	SATA Express host cable receptacle connector (SATA spec) <sup>a</sup> 7-pin SATA single lane cable receptacle connector(s) <sup>a</sup>
SATA Express Host Cable Receptacle Connector	SATA Express host plug connector (SATA spec) <sup>a</sup>
SATA Express Device Cable Receptacle Connector	SATA Express device plug connector (SATA Spec) <sup>a</sup> SATA Device Plug Connector (SATA spec) <sup>a</sup>
Power Dongle (on the backside of the SATA Express device cable receptacle connector)	SATA power cable receptacle connector (SATA spec) <sup>a</sup>
<sup>a</sup> Interface is supported. <sup>b</sup> Interface is supported if muxing of SATA/SAS/PCIe are implemented in SFF-8639 host. <sup>c</sup> 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  $\leq 10$  ps, as measured with a differential Time Domain Transmission (TDT) (risetime = 50 ps, 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 of a through-hole (TH) or SMT pad. The recommendations are provided to minimize connector PCB launch mismatches:

- a) through-hole 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) SMT a small pad that meets design for manufacturability (i.e., DFM) rules should be used. The ground plane immediately underneath the signal pads should be voided to remove excessive parasitic.

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.

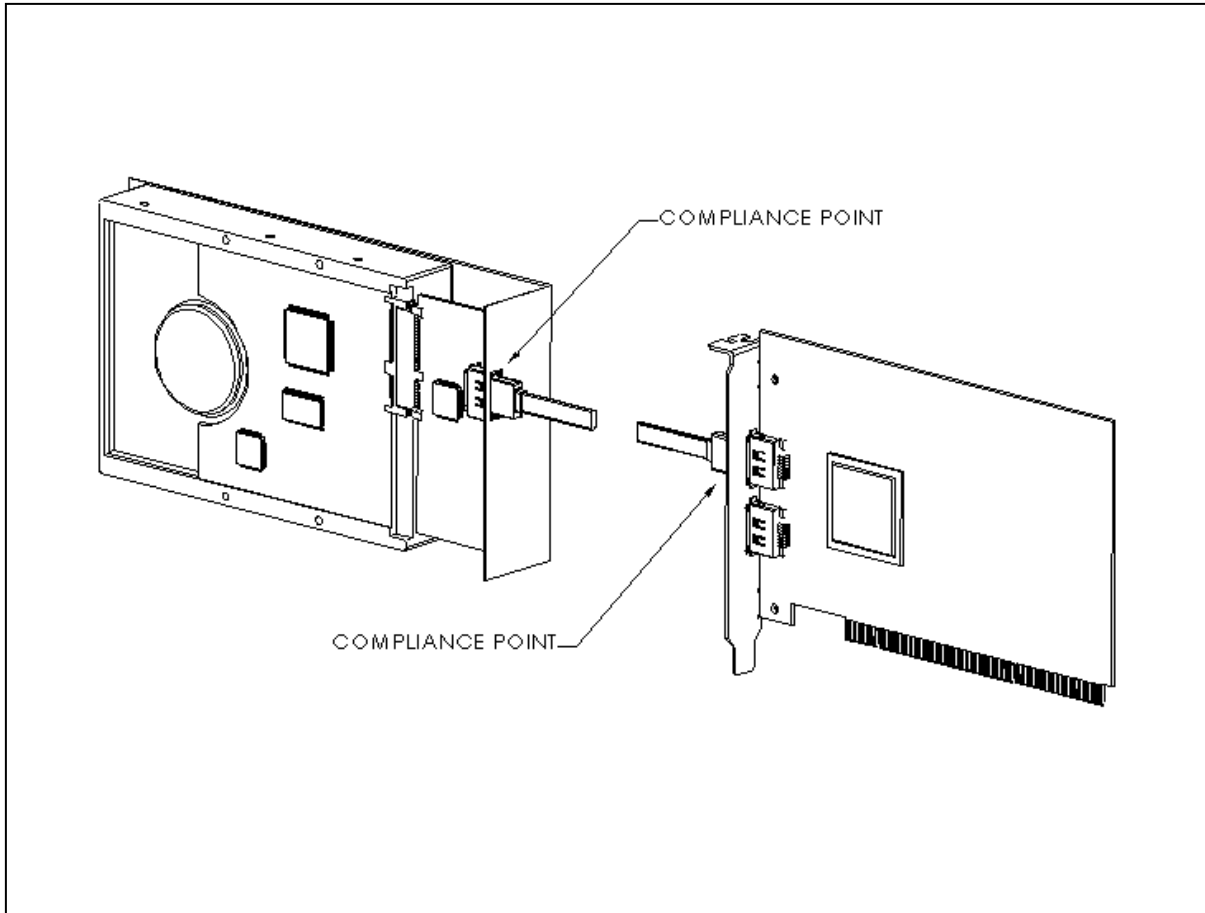
The primary implementation is:

- 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.

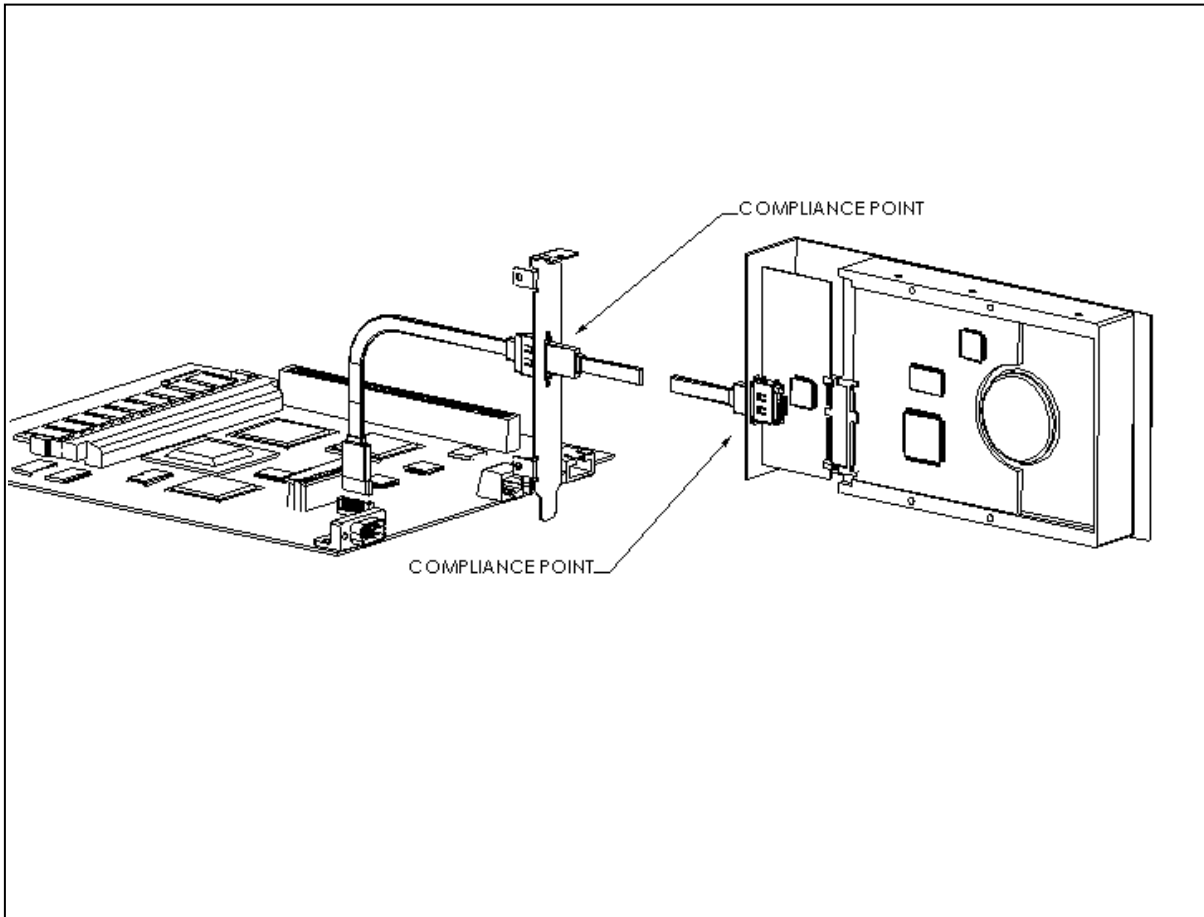
This implementation is shown in Figure 143.



**Figure 143 – 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 187, 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 144, the compliance points remain at the shielded external connectors.



**Figure 144 – 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 for use with shielded (external Serial ATA) cabling;
- b) a fully shielded right angle (RA) PCB mounted SMT plug, (and reversed pin-out version);
- c) a fully shielded RA PCB mounted through-hole plug;

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

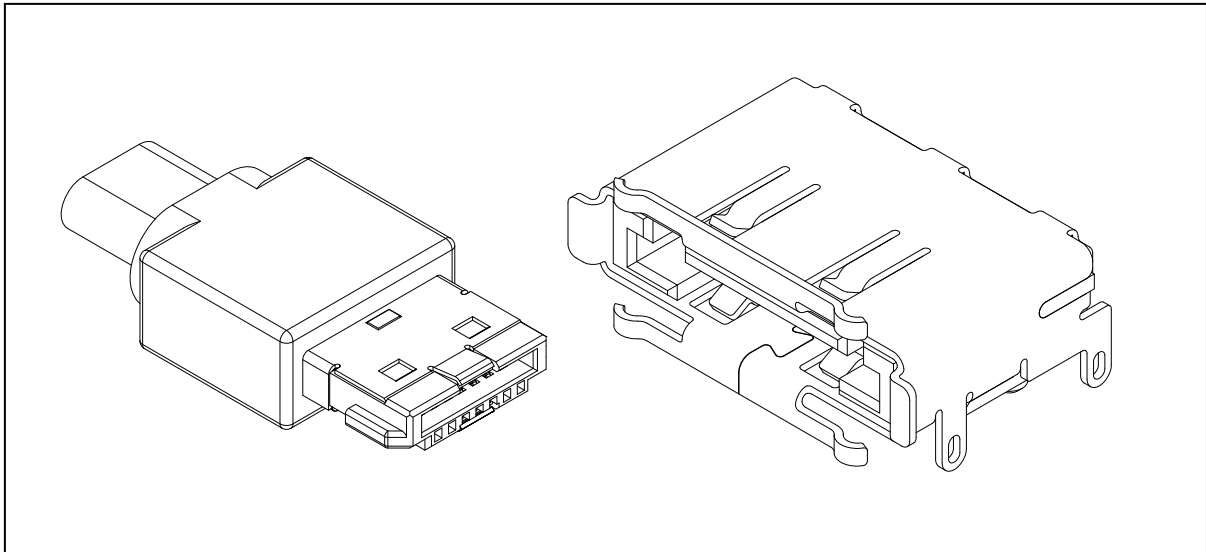
- d) a fully shielded vertical PCB mounted SMT plug; and
- e) a fully shielded vertical PCB mounted through-hole plug.

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 6, 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.



**Figure 145 – Renderings of External Serial ATA cable receptacle and right angle plug**

### 6.11.2.2 External Serial ATA connector mechanical drawings

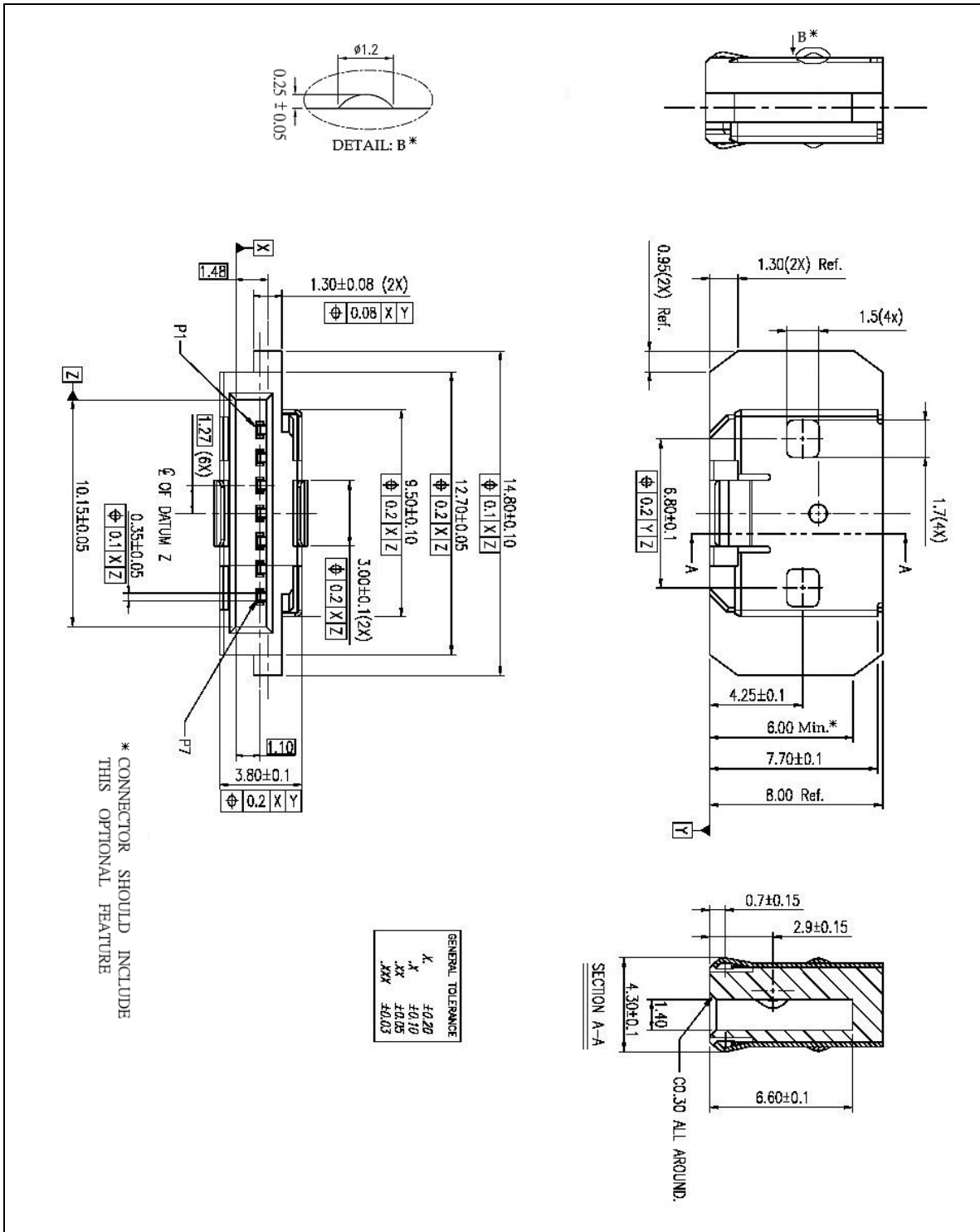


Figure 146 – Mechanical dimensions of External Serial ATA cable receptacle assembly

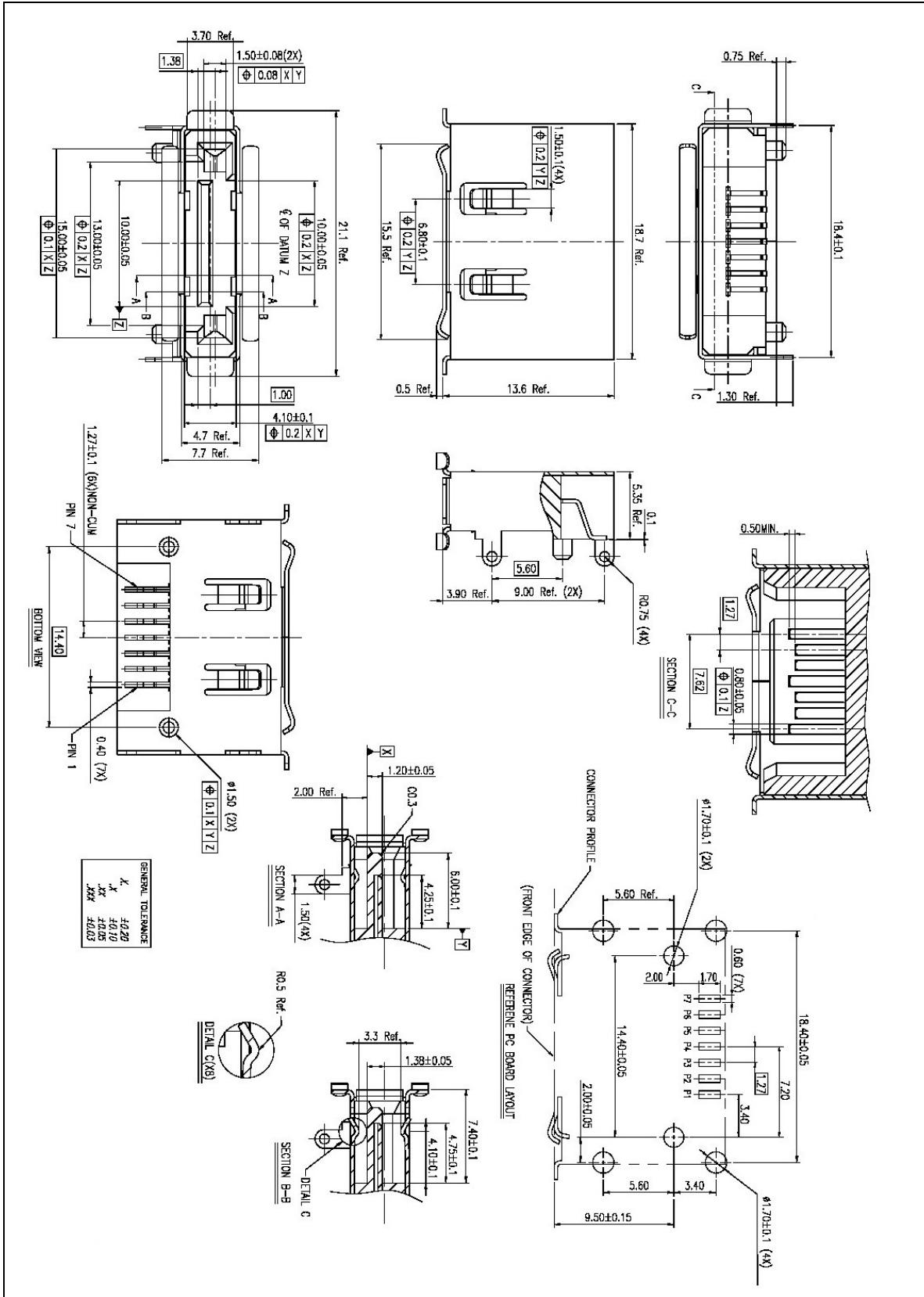


Figure 147 – Mechanical dimensions of External Serial ATA RA SMT plug

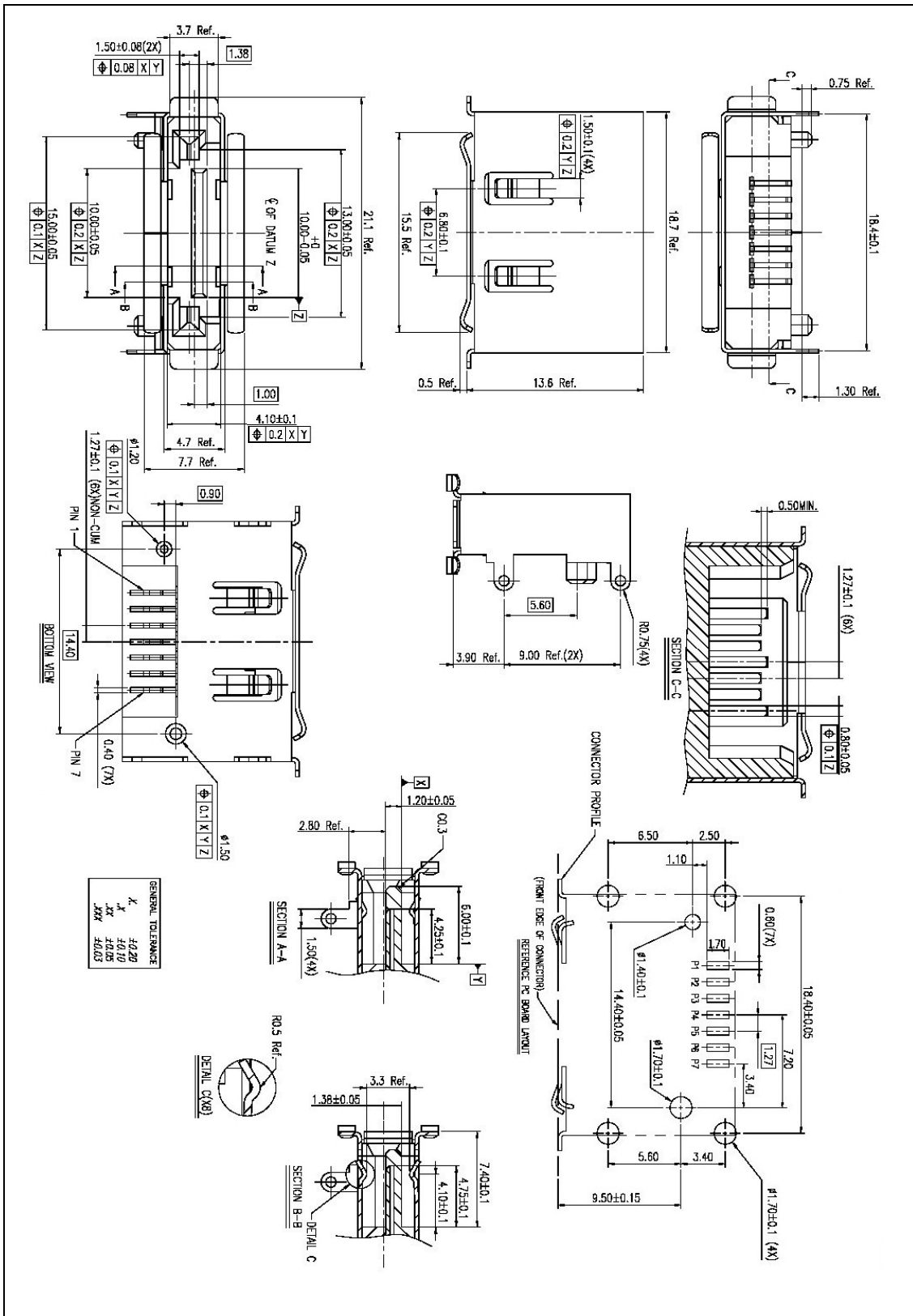


Figure 148 – Mechanical dimensions of External SATA RA SMT plug – reversed pin out

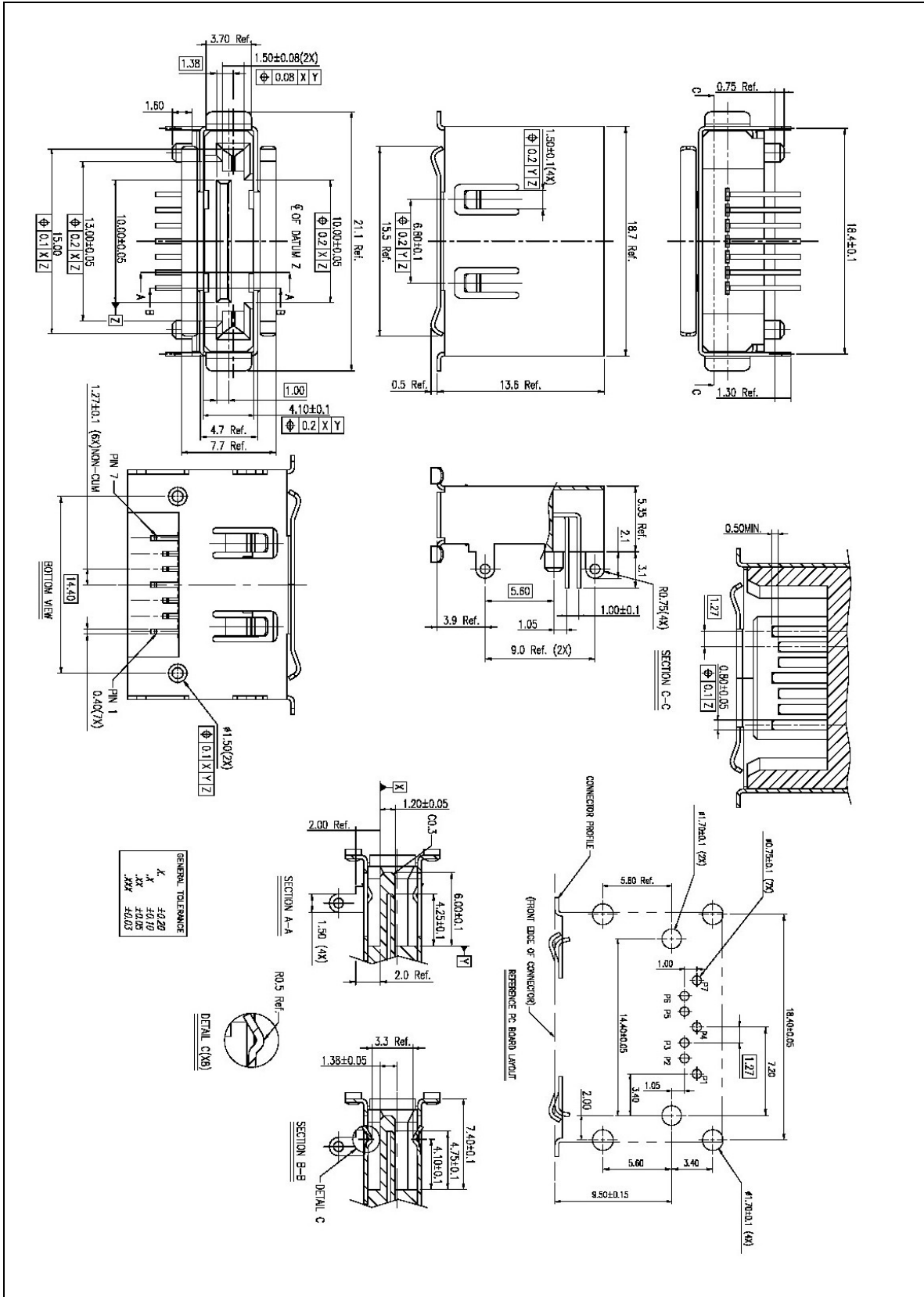


Figure 149 – Mechanical dimensions of External Serial ATA RA through-hole

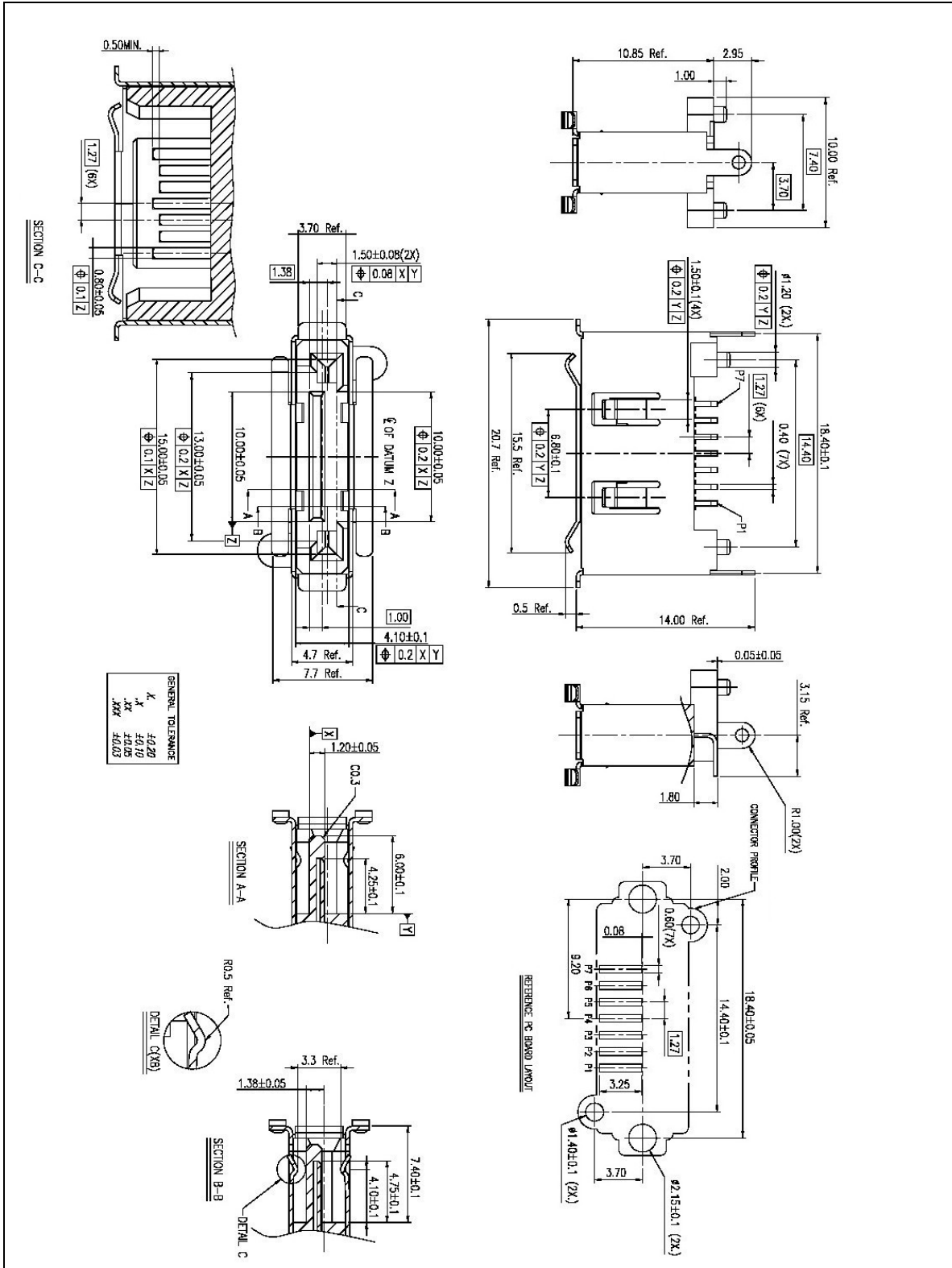


Figure 150 – Mechanical dimensions of External Serial ATA vertical SMT plug



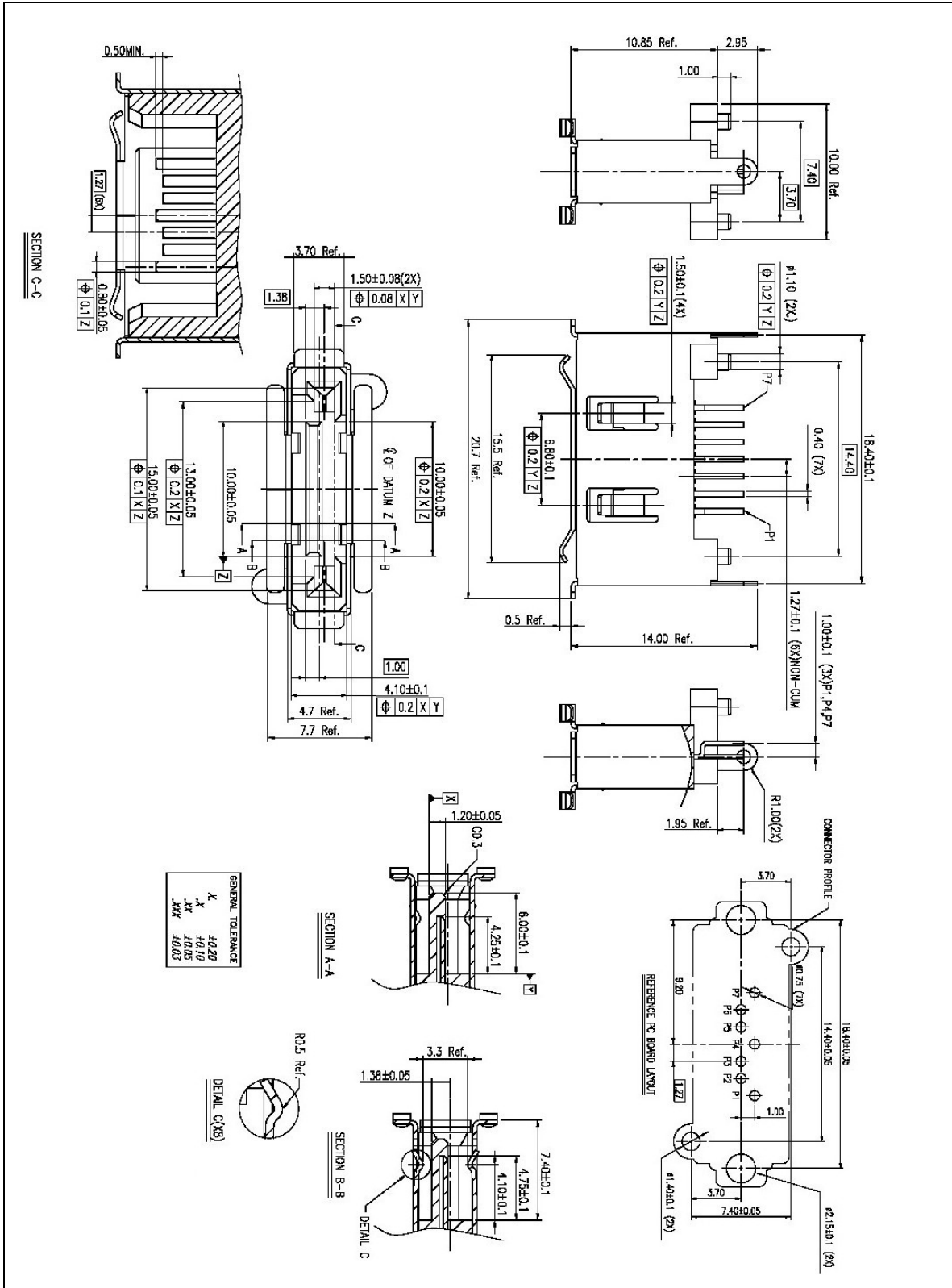


Figure 151 – Mechanical dimensions of External Serial ATA vertical through-hole plug

### **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;
- 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 host bus adapter 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**

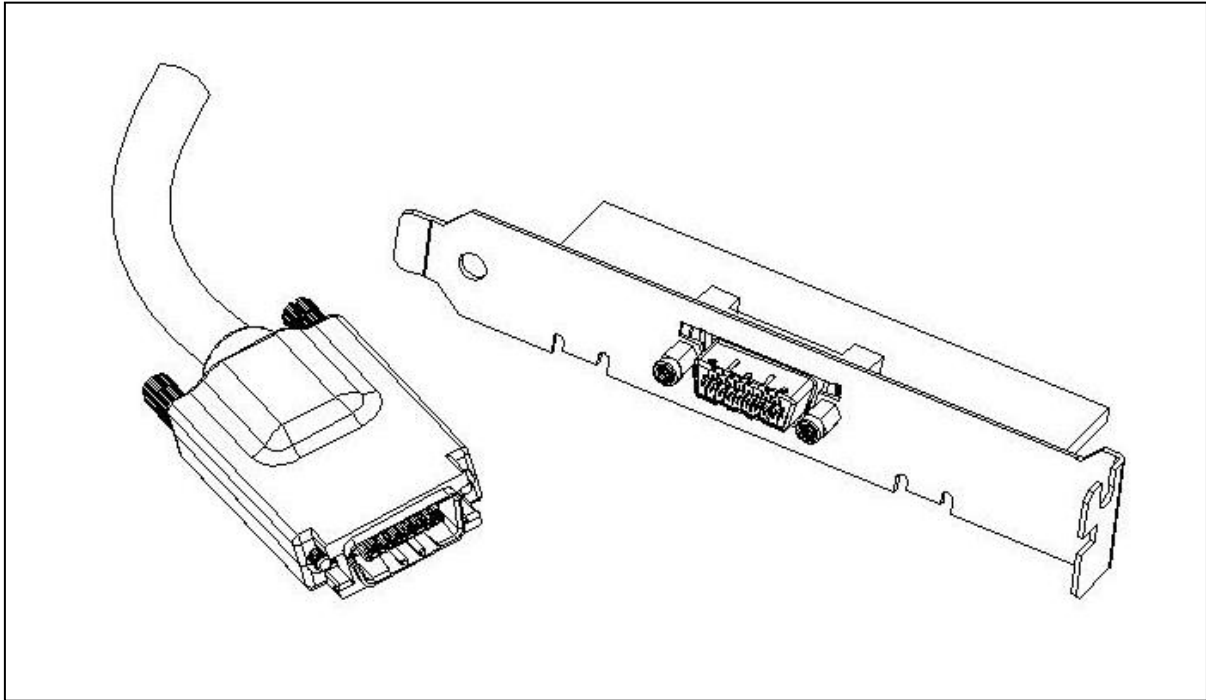
The External Multilane cable/connector shall be used with Gen1m/Gen2m signal levels only. If Gen1m/Gen2m signal levels are used, the cable length is limited to two meters.

#### **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.

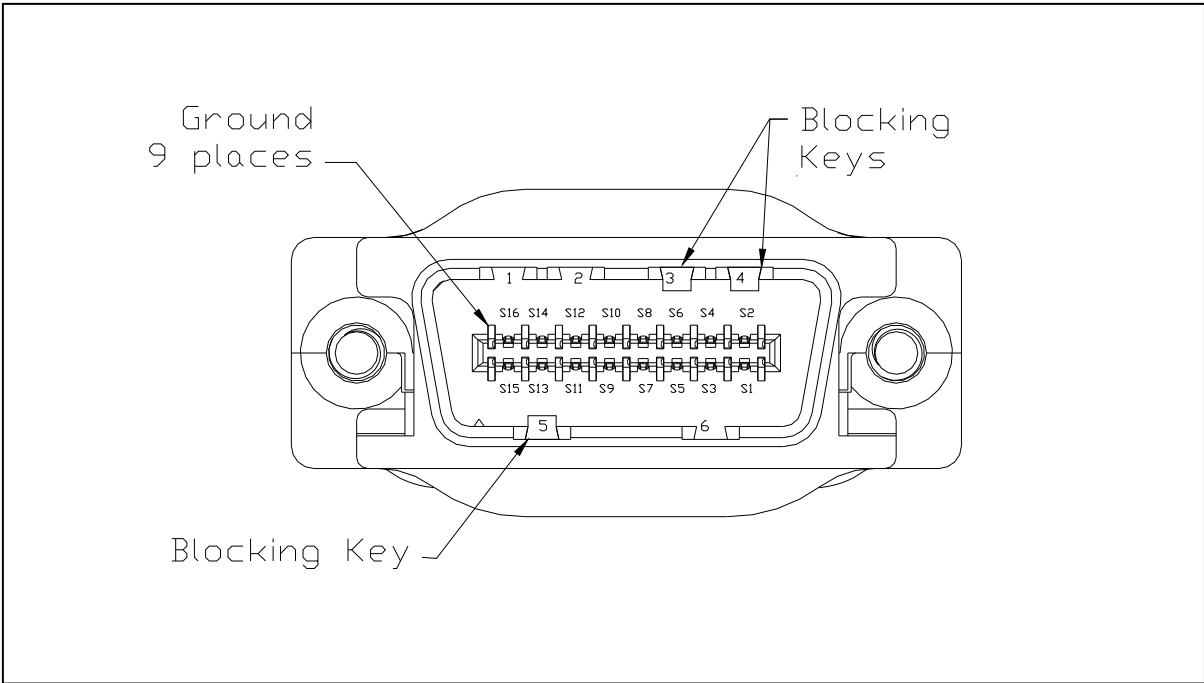


**Figure 152 – External Multilane cable and connector**

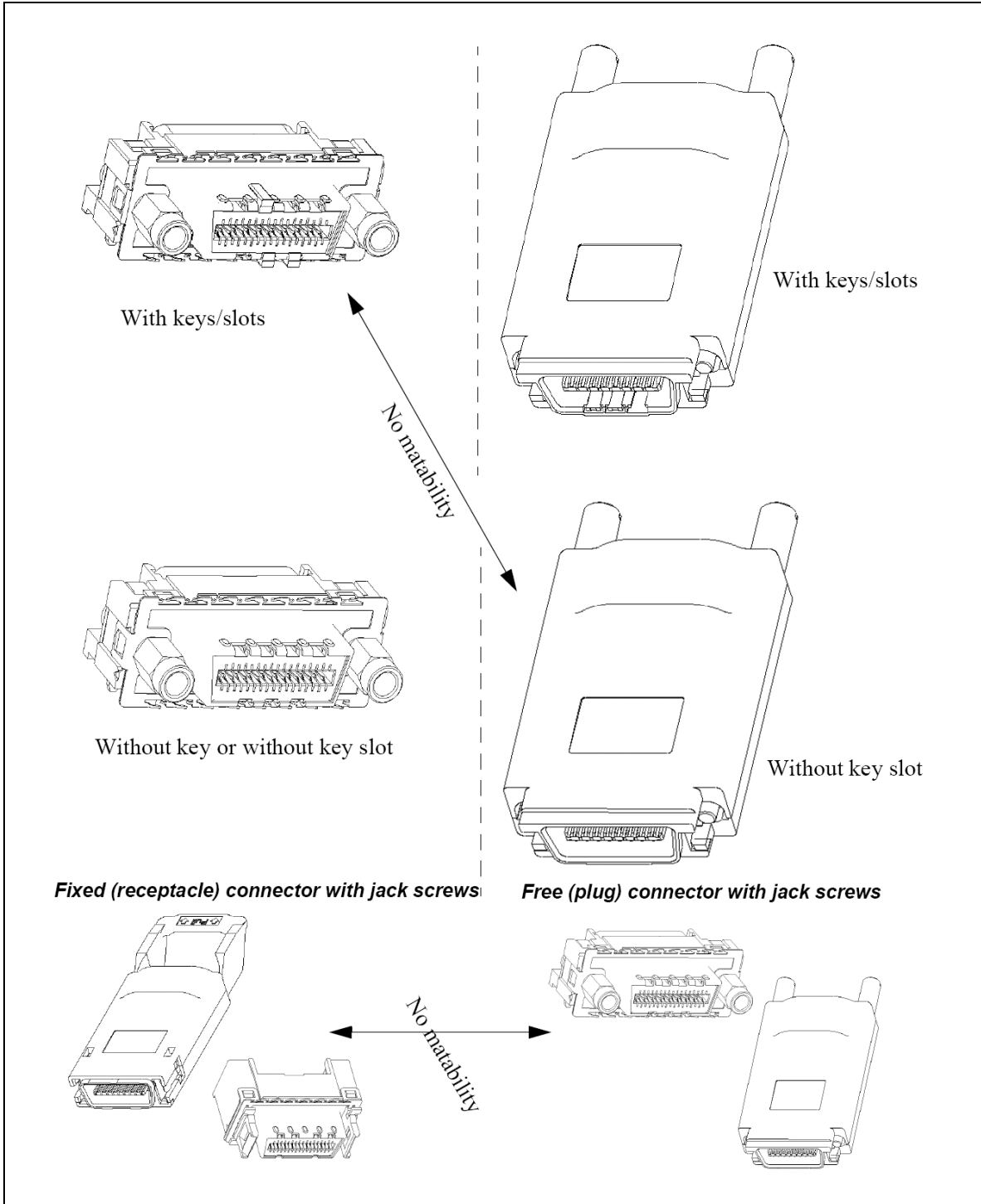
### 6.11.3.3 Keying requirements

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 153.

Optional keying allows connection between Serial ATA HBAs and JBODs, but disallows connection to SAS HBAs or JBODs, 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 153 – Multilane cable connector blocking key locations**



**Figure 154 – Plug/receptacle keying**

### 6.11.3.4 4 lane 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 host bus adapter to an intelligent backplane in an adjacent JBOD (just a bunch of disks) 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 Gbps; 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 155.

Unique keying requirements for x-level signal levels are obsolete. Unique keying requirements for m-level signal levels are shown in Figure 156.

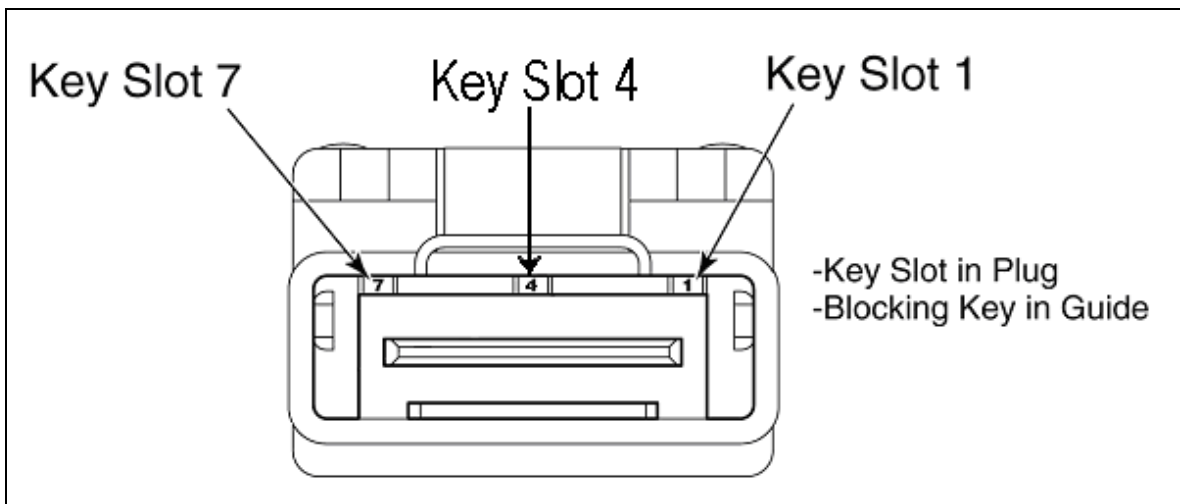
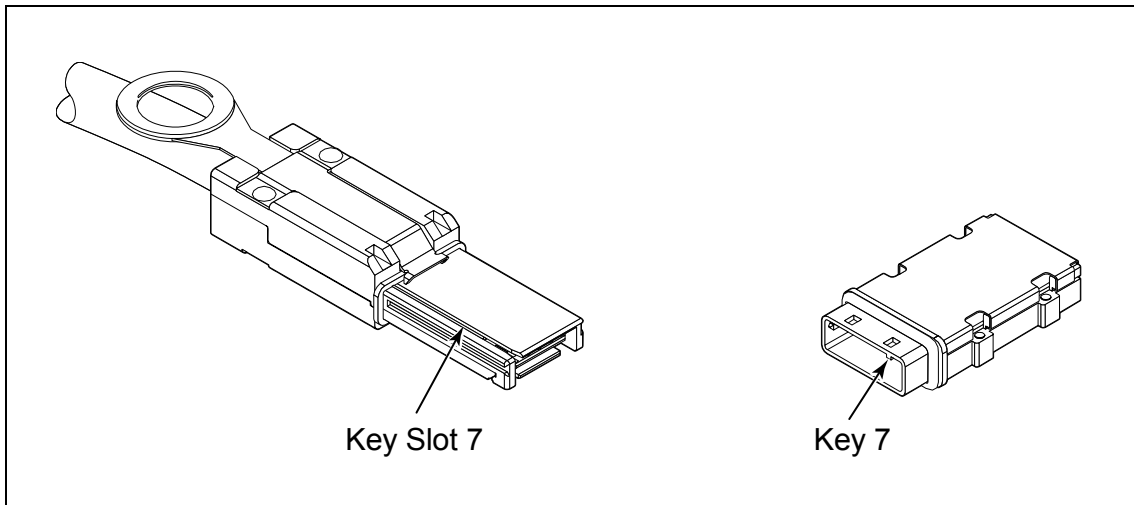


Figure 155 – Mini SATA External Multilane system, key features



**Figure 156 – 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 157.

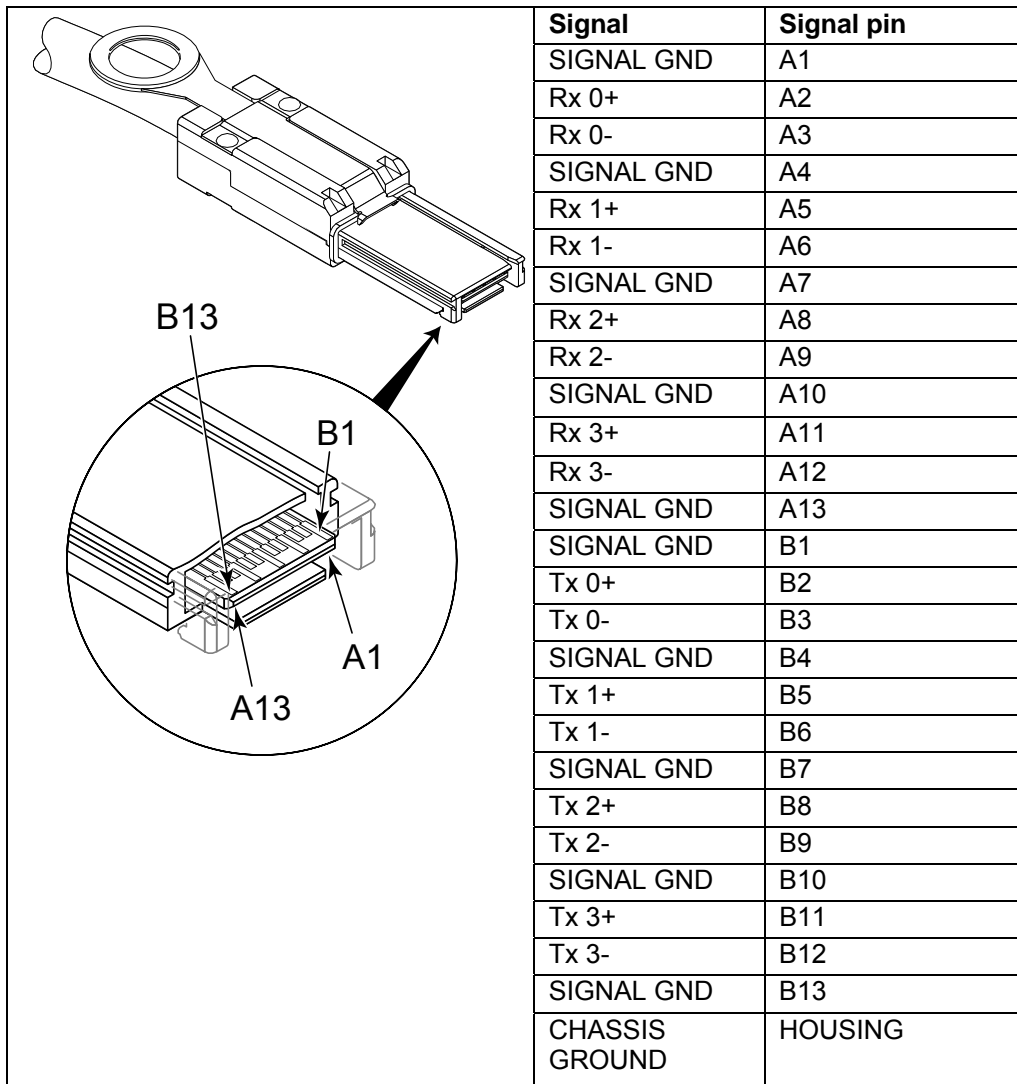


Figure 157 – Mini SATA External Multilane connector pin assignments

## 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 $\pm$ 15 ohm	P1
Cable Absolute Differential Impedance	100 ohm $\pm$ 10 ohm	P2
Cable Pair Matching Impedance	$\pm$ 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 ps	P8
Maximum Inter-Symbol Interference	50 ps	P9
Maximum Intra-Pair Skew	10 ps	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 ps	P8
Maximum Inter-Symbol Interference	50 ps	P9
Maximum Intra-Pair Skew	20 ps	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 ps	P8
Maximum Inter-Symbol Interference	50 ps	P9
Maximum Intra-Pair Skew	20 ps	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 ps 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 Gbps Pattern Source with a minimum rise time of 136 ps, measured from the 20 % threshold to 80 % threshold. The rise time should be as close to 136 ps 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 mm of Hg to 800 mm of Hg.

### **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  $\pm$  5 ohm) over a ground plane (e.g., single ended 50 ohm  $\pm$  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**

<b>C1</b>	<b>Minimizing Skew between V+ and V-, Diff Signals</b>
	<p>The procedure is:</p> <ol style="list-style-type: none"> <li>1) define Differential Channel Stimulus:             <ol style="list-style-type: none"> <li>a) channel 1 and channel 3 positive edge step response (<math>V^+</math>); and</li> <li>b) channel 2 and channel 4 negative edge step response (<math>V^-</math>);</li> </ol> </li> <li>2) differential response, identify the differential signal (<math>V_{diff}</math>) as a scope response, (e.g., use math function to obtain <math>V_{diff} = V^+ - V^-</math> (CH1-CH2 or CH3-CH4)); and</li> <li>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 1ps.</li> </ol>
<b>C2</b>	<b>Establishing a reference plane at the connector</b>
	<p>The procedure is:</p> <ol style="list-style-type: none"> <li>1) follow calibration procedures outlined in the firmware of the oscilloscope;</li> <li>2) select the define reference plane option within the scope firmware to establish a reference plane at the input of the test fixture; and</li> <li>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.</li> </ol>
<b>C3</b>	<b>Establishing the rise time reference trace</b>
	<p>The procedure is:</p> <ol style="list-style-type: none"> <li>1) configure the TDR modules to generate a differential step impulse response and identify the differential rising edge of the trace;</li> <li>2) identify the high and low voltage values of the impulse response;</li> <li>3) identify the 20 % and 80 % voltage levels and verify that the rise time of the step impulse is between 25 ps and 35 ps. There are two methods for adjusting the step impulse response to be within the desired range:             <ol style="list-style-type: none"> <li>a) the system rise time is to be set via equipment filtering techniques. The filter programmed equals <math>\sqrt{t_{r(observable)}^2 - t_{r(stimulus)}^2}</math>; or</li> <li>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;</li> </ol> <p style="text-align: center;">and</p> </li> <li>4) once the correct rise time has been established, verify the rise time using the reference traces on the PCB fixture.</li> </ol>

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)

<b>P1</b>	<p><b>Mated Connector Differential Impedance</b></p> <p>The procedure is:</p> <ol style="list-style-type: none"> <li>1) 1) calibrate the instrument and system using the measurement traces, then follow common procedures:             <ol style="list-style-type: none"> <li>1) C1;</li> <li>2) C2; and</li> <li>3) C3;</li> </ol> </li> <li>2) the instrument rise time shall be set or the results filtered for a minimum of 55 ps to a maximum of 70 ps, measured from the 20 % threshold to 80 % threshold system risetime. The system risetime shall be set as close to 70 ps, measured from the 20 % threshold to 80 % threshold as practical; and</li> <li>3) measure and record the maximum and minimum values of the near end connector differential impedance.</li> </ol>
<b>P2</b>	<p><b>Cable Absolute Differential Impedance</b></p> <p>The procedure is:</p> <ol style="list-style-type: none"> <li>1) calibrate the instrument and system using the measurement traces, then follow common procedures:             <ol style="list-style-type: none"> <li>1) C1;</li> <li>2) C2; and</li> <li>3) C3;</li> </ol> </li> <li>2) the instrument rise time shall be set or the results filtered for a minimum of 55 ps to a maximum of 70 ps, measured from the 20 % threshold to 80 % threshold system risetime. The system risetime shall be set as close to 70 ps, measured from the 20 % threshold to 80 % threshold as practical.</li> <li>3) Measure and record maximum and minimum cable differential impedance values from the TDR trace in the first 500 ps of cable response following any vestige of the connector response.</li> </ol>
<b>P3</b>	<p><b>Cable Pair Matching</b></p> <p>The procedure is:</p> <ol style="list-style-type: none"> <li>1) calibrate the instrument and system using the measurement traces, then follow common procedures:             <ol style="list-style-type: none"> <li>1) C1;</li> <li>2) C2; and</li> <li>3) C3;</li> </ol> </li> <li>2) 2) The instrument rise time shall be set or the results filtered for a minimum of 55 ps to a maximum of 70 ps, measured from the 20 % threshold to 80 % threshold system risetime. The system risetime shall be set as close to 70 ps, measured from the 20 % threshold to 80 % threshold as practical;</li> <li>3) 3) measure and record the single-ended cable impedance of each cable within a pair, (e.g., <math>Z_{L1}</math>, <math>Z_{L2}</math>); and</li> <li>4) measure and record maximum and minimum cable impedance values from the TDR trace in the first 500 ps of cable response following any vestige of the connector response, (e.g., <math>Z_{L1-max}</math>, <math>Z_{L1-min}</math> and <math>Z_{L2-max}</math>, <math>Z_{L2-min}</math>).</li> </ol> <p>The desired parameter equals <math>Z_{max} = Z_{L1-max} - Z_{L2-max}</math> and <math>Z_{min} = Z_{L1-min} - Z_{L2-min}</math>.</p>

**Table 40 – Interconnect test methodologies / procedures (part 2 of 5)**

<b>P4</b>	<b>Common Mode Impedance</b>
	<p>The procedure is:</p> <ol style="list-style-type: none"> <li>1) calibrate the instrument and system using the measurement traces, then follow common procedures:                             <ol style="list-style-type: none"> <li>1) C1;</li> <li>2) C2; and</li> <li>3) C3;</li> </ol> </li> <li>2) the instrument rise time shall be set or the results filtered for a minimum of 55 ps to a maximum of 70 ps, measured from the 20 % threshold to 80 % threshold system risetime. The system risetime shall be set as close to 70 ps, measured from the 20 % threshold to 80 % threshold as practical;</li> <li>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.</li> <li>4) measure the even mode impedance from the TDR trace of the first step generator in the first 500 ps of cable response following any vestige of the connector response;</li> <li>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</li> <li>6) make the same measurement and math calculation of the second step generator.</li> </ol> <p>The Common Mode Impedance for each step generator shall meet the requirement.</p>
<b>P5</b>	<b>Insertion Loss</b>
	<p>The procedure is:</p> <ol style="list-style-type: none"> <li>1) calibrate the instrument and system using the measurement traces, then follow common procedures:                             <ol style="list-style-type: none"> <li>1) C1; and</li> <li>2) C2;</li> </ol> </li> <li>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);</li> <li>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</li> <li>4) the insertion loss of the sample is calculated by <math>IL_{sample} = IL_{system} - IL_{fixture}</math>.</li> </ol>

**Table 40 – Interconnect test methodologies / procedures (part 3 of 5)**

<b>P6</b>	<b>Differential to Differential Crosstalk NEXT</b>
	<p>The procedure is:</p> <ol style="list-style-type: none"> <li>1) calibrate the instrument and system using the measurement traces, then follow common procedures;               <ol style="list-style-type: none"> <li>1) C1;</li> <li>2) C2; and</li> <li>3) C3;</li> </ol> </li> <li>2) terminate the far ends of the reference trace with characteristic impedance loads of 50 ohm;</li> <li>3) measure and record the system and fixturing crosstalk, it is defined as the noise floor, (e.g., <math>V_{noise}</math>) ;</li> <li>4) Terminate the far ends of the device and listen lines with characteristic impedance loads of 50 ohm;</li> <li>5) Connect the source to the device pair and the receiver to the near-end of the listen pair;</li> <li>6) Measure the NEXT over a frequency range of 10 MHz to 4 500 MHz, (e.g., <math>V_{NEXT}</math>); and</li> <li>7) Verify that the sample crosstalk is out of the noise floor, (e.g., <math>V_{NEXT} &gt; V_{noise}</math>).</li> </ol>



**Table 40 – Interconnect test methodologies / procedures (part 4 of 5)**

<b>P7</b>	<b>Multilane (Multi Disturber) Differential Crosstalk ML-CXT</b>
	<p>The procedure is:</p> <ol style="list-style-type: none"> <li>1) calibrate the instrument and system using the measurement traces, then follow common procedures:               <ol style="list-style-type: none"> <li>1) C1,</li> <li>2) C2; and</li> <li>3) C3;</li> </ol> </li> <li>2) terminate the far ends of the reference trace with characteristic impedance loads of 50 ohm;</li> <li>3) measure and record the system and fixturing crosstalk. It is defined as the noise floor, (e.g., <math>V_{noise}</math>);</li> <li>4) terminate the far ends of the device and listen lines with characteristic impedance loads of 50 ohm;</li> <li>5) connect the source to the device pair and the receiver to the near-end of the listen pair;</li> <li>6) measure the ML-CXT over a frequency range of 10 MHz to 4 500 MHz, (e.g., <math>V_{ML-CXT}</math>); and</li> <li>7) verify that the sample crosstalk is out of the noise floor, (e.g., <math>V_{ML-CXT} &gt; V_{noise}</math>).</li> </ol> $8) \quad ML - CXT(f) = -20 \times \log \left( \sum_{A=1}^{7^1} 10^{-V_{ML-CXT}(f)_A/20} \right)$ <p>where:</p> <p><math>ML-CXT(f)</math> is the Multilane Cable assembly Crosstalk at frequency <math>f</math> observed on any given receive lane.</p> <p><math>V_{ML-CXT}(f)_A</math> is the relative crosstalk at frequency <math>f</math> between the receiver/victim and any combination of aggressor <math>A^a</math> that exhibits less than 40 dB of isolation.</p> <p><math>f</math> is the frequency ranging from 10 MHz to 4 500 MHz</p> <p><math>A</math> is the 1 to 7 (receiver/victim to aggressor <math>a</math> pair combinations)</p> <p><sup>a</sup> 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.</p> <p>Note- ML-CXT summation calculations accounts for any aggressor <sup>a</sup> relative to a receiver/victim pair that exhibit less than 40 dB of isolation.</p>

**Table 40 – Interconnect test methodologies / procedures (part 5 of 5)**

<b>P8</b>	<b>Differential Rise Time</b>
	<p>The procedure is:</p> <ol style="list-style-type: none"> <li>1) calibrate the instrument and system using the measurement traces, then follow common procedures:             <ol style="list-style-type: none"> <li>1) C1;</li> <li>2) C2; and</li> <li>3) C3;</li> </ol> </li> <li>2) connect the TDR step impulse response generators to the near end of the signal path under test; and</li> <li>3) record the output rise time at the far end of the signal path under test.</li> </ol>
<b>P9</b>	<b>Inter-Symbol Interference<sup>a</sup></b>
	<p>The procedure is:</p> <ol style="list-style-type: none"> <li>1) connect a differential pattern source at the input of the test fixture;.</li> <li>2) the minimum rise time of the pattern source shall be 136 ps, measured from the 20 % threshold to 80 % threshold, the rise and fall times should be as close to 136 ps as is practical, to minimize the resulting DJ and produce the most accurate results.</li> <li>3) Generate a LBP at 3.0 Gbps through the fixture. The Lone Bit Pattern emphasizes ISI.</li> <li>4) Using a JMD, evaluate the DJ introduced at the end of the cable.</li> </ol>
<b>P10</b>	<b>Intra-Pair Skew</b>
	<p>The procedure is:</p> <ol style="list-style-type: none"> <li>1) calibrate the instrument and system using the measurement traces, then follow common procedures:             <ol style="list-style-type: none"> <li>1) C1;</li> <li>2) C2; and</li> <li>3) C3;</li> </ol> </li> </ol> <p>and</p> <ol style="list-style-type: none"> <li>2) measure the propagation delay of each single ended signal within a pair at the mid point of the voltage swing, (e.g., <math>t_{\text{delay}} = V_{\text{mid}+} - V_{\text{mid}-}</math>)</li> </ol> <p>where <math>V_{\text{mid}} = \frac{V_{\text{high}} - V_{\text{low}}}{2}</math>.</p>
<p><sup>a</sup> 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.</p>	

## 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 operates using the default behavior in accordance with 6.13.1.2. Otherwise the Hardware Feature Control operates 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 connector type as indicated in Table 41. Not all features are defined for all connector types.

**Table 41 – DSS and DAS for various connectors**

Standard Connector (3.5 inch & 2.5 inch)	1.8 inch Micro SATA Connector	LIF-SATA Connector
Pin 11: a) DSS; and b) DAS.	Pin 7: a) DAS	Pin 8: a) DSS; and b) DAS.

The Hardware Feature Control, (i.e., pins P11, P7, or P8 depending upon connector) 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, the Hardware Feature Control 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 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.

A host shall support one feature, either receiving activity indication or disable staggered spinup control. If a host supports receiving activity indication, then the host shall not use pin P11, P7, or P8 depending upon connector to disable staggered spinup. If a host does not support receiving activity indication, then the host may use pin P11, P7, or P8 depending upon connector to disable staggered spinup.

#### 6.13.1.3 Extended behavior

Hardware Feature Control may be used by the device for one of the following:

- a) default use of the Hardware Feature Control (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**

<b>Parameter</b>	<b>Min</b>	<b>Max</b>	<b>Description &amp; Conditions</b>
V <sub>DHUactive</sub>	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 <sub>DInact</sub> leakage currents.
V <sub>DHUnegate</sub>	-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 <sub>DInact</sub> leakage currents.

### 6.13.3 Device activity signal

#### 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 158 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.

NOTE 18 - Note that the host should not rely on a particular resistor pull-up value on the device side, nor should the device rely on particular host resistor values.

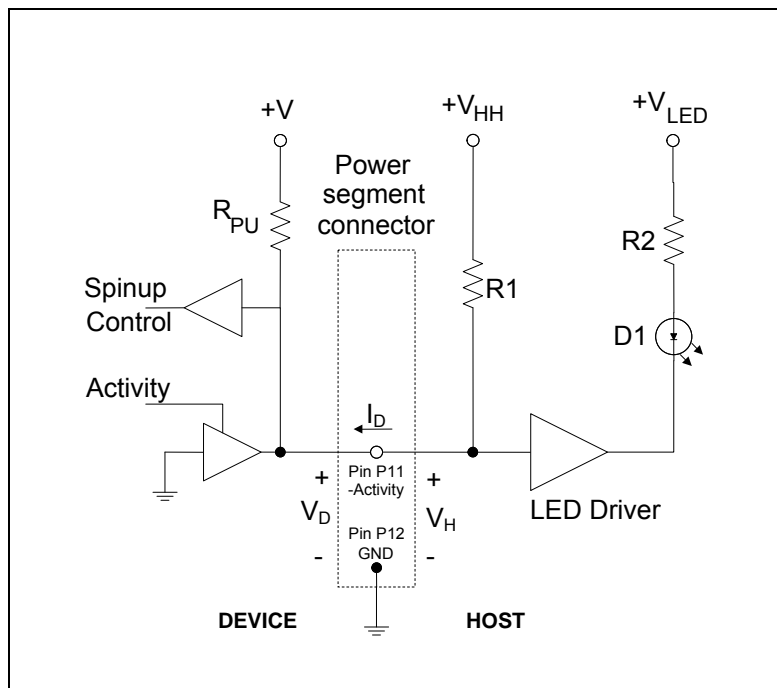


Figure 158 – Example activity signal electrical block diagram

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 activity signal electrical parameters**

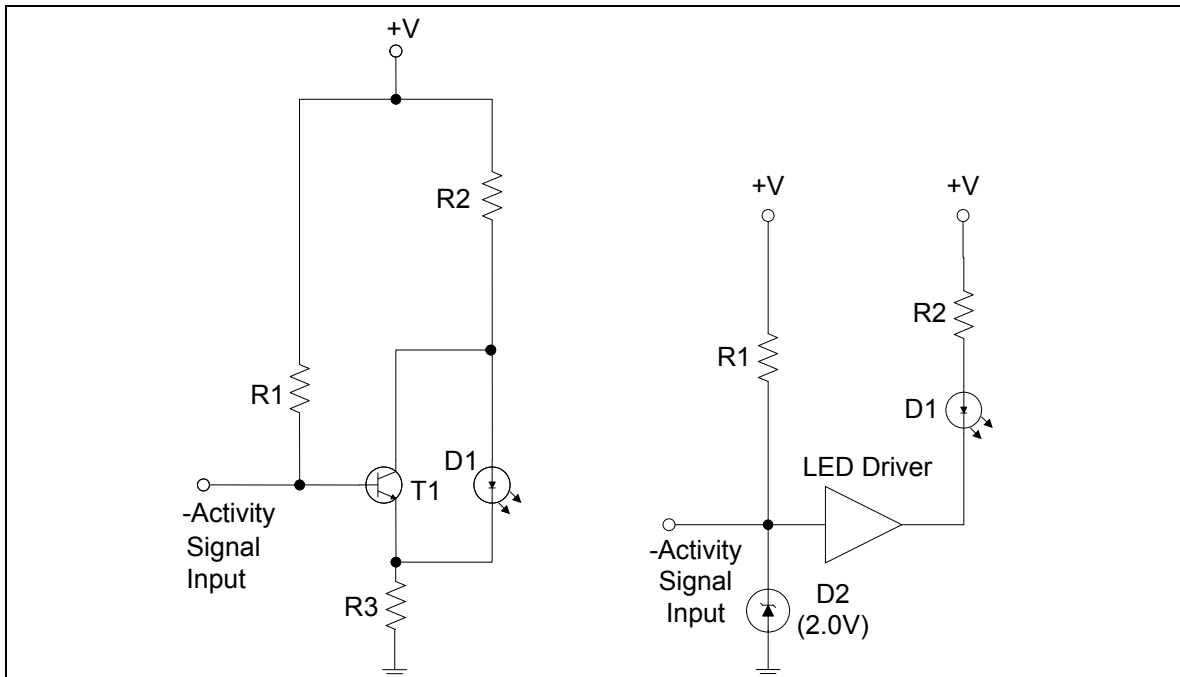
Parameter	Min	Max	Description & Conditions
$V_{DIn}$	-0.5 V	2.1 V	Tolerated input voltage.
$V_{DAct}$	0 mV	225 mV	Device output voltage when driving low under the condition $I_D$ less than or equal to 300 $\mu$ A.
$V_{DIact}$	-0.1 V	3.3 V	Device output voltage when not driving low.
$I_{DIact}$	-10 $\mu$ A	100 $\mu$ A	Device leakage current when not driven.

**Table 44 – Host activity signal electrical parameters**

Parameter	Min	Max	Description & Conditions
$V_{HIin}$	-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 $\mu$ A	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 159 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 159 – 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
Signal asserted (driven low)	Command(s) outstanding <sup>a</sup>
Signal negated (high impedance)	All other conditions

<sup>a</sup> 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.

### 6.13.4 Disable Staggered Spinup control

#### 6.13.4.1 Electrical and functional definition

The staggered spinup feature as defined in 13.10. Devices may optionally provide support to disable staggered spinup through pin P11 of the power segment connector. The disable staggered spinup control is an active asserted low host signal.

Before the device spins up its media, devices that support disable staggered spinup control shall detect whether pin P11 is asserted low by the host. If pin P11 is asserted low the device shall disable staggered spinup and immediately initiate media spinup. If pin P11 is not connected in the host (floating), devices that support disable staggered spinup through pin P11 shall enable staggered spinup. Table 46 defines the electrical signal requirements for the device detection of disable staggered spinup.

**Table 46 – Host staggered spinup control electrical requirements**

Parameter	Min	Max	Description & 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 be static and shall not be changed while power is applied. If the signal is pulled low by the host during the disable staggered spinup detection period, the signal shall remain low. Devices shall disable the activity signal if the host signals disable staggered spinup.

If supported, the device shall sample the disable staggered spinup condition after the time DC power is applied and before PHYRDY is asserted.

### 6.13.4.2 Disable Staggered Spinup circuit (informative)

The host circuit for signaling disable staggered spinup by pulling pin P11 low is vendor specific. Figure 160 illustrates a conceptual host circuit that is able to satisfy the electrical requirements for signaling disable staggered spinup. It is permissible for the host to statically short pin P11 to ground or for the host to actively drive the signal low.

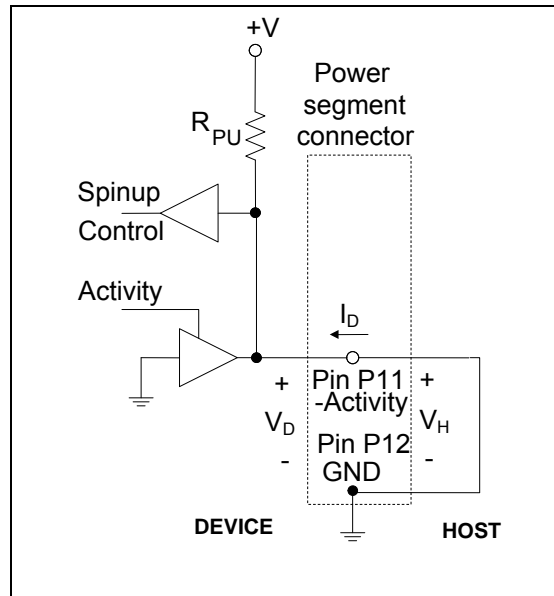


Figure 160 – Example circuit for disable staggered spinup

### 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 (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 161 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 19 - 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.

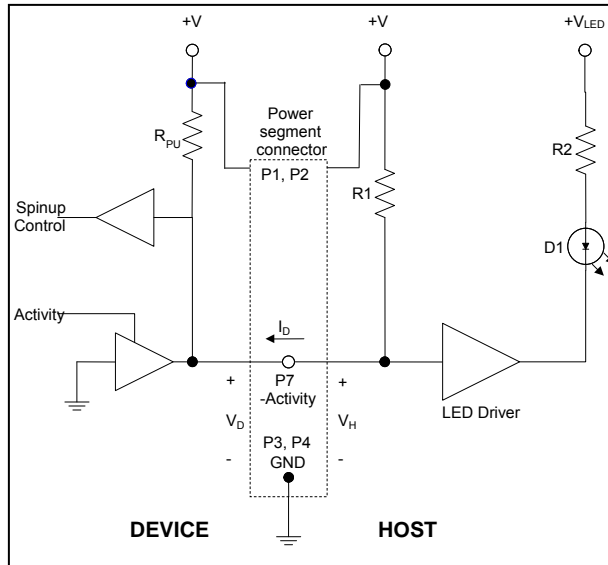


Figure 161 – Example activity signal electrical block diagram

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 activity signal electrical parameters

Parameter	Min	Max	Description & 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 $\mu$ A.
$V_{DIInact}$	-0.1 V	3.3 V	Device output voltage when not driving low.
$I_{DIInact}$	-10 $\mu$ A	100 $\mu$ A	Device leakage current when not driven.

Table 48 – Host activity signal electrical parameters

Parameter	Min	Max	Description & 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 $\mu$ A	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 162 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.

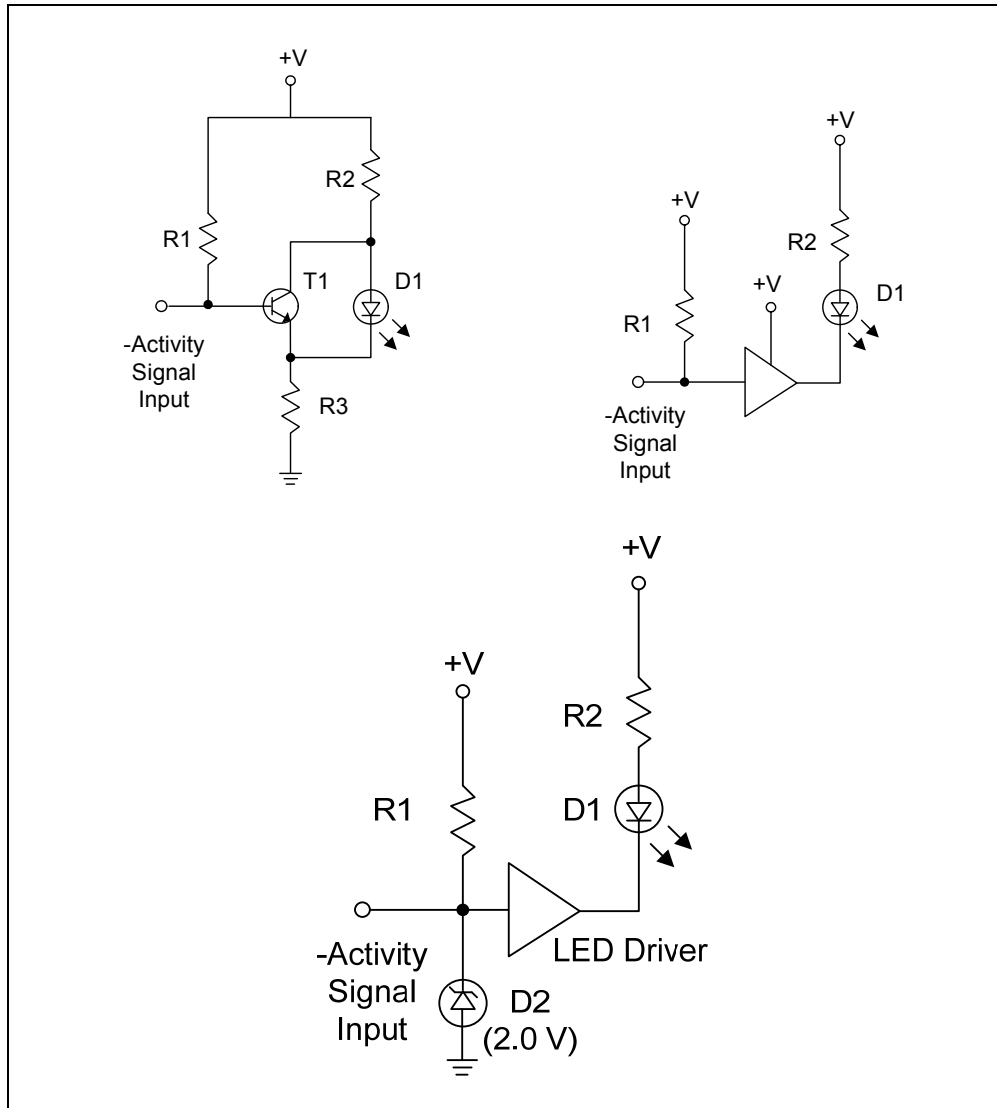


Figure 162 – Example host LED driver circuits

### 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**

<b>State</b>	<b>Condition</b>
Signal asserted (driven low)	Command(s) outstanding <sup>a</sup>
Signal negated (high impedance)	All other conditions

<sup>a</sup> 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**

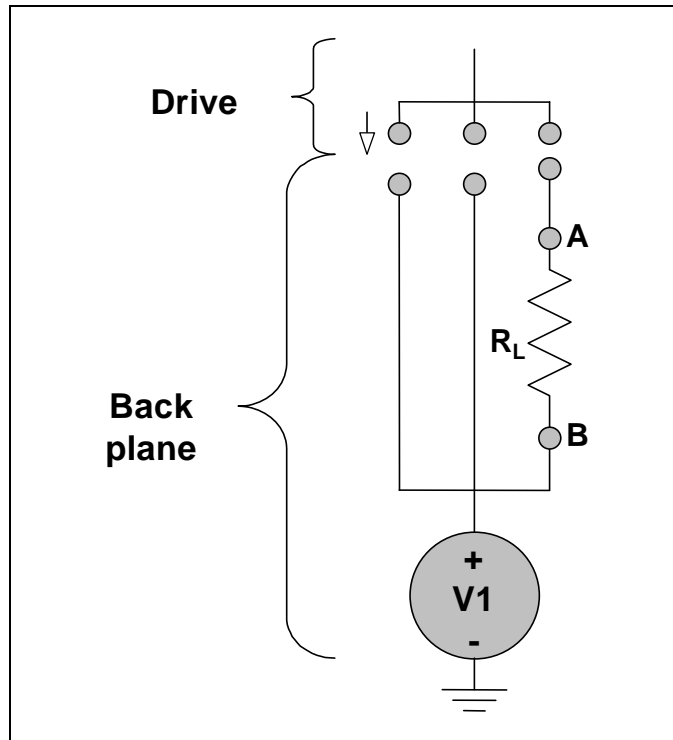
For a storage subsystem, hot plug capability is required as well as the ability to seamlessly handle presence detection in those cases where the storage subsystem may remove power to individual receptacles.

### **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 163 illustrates one hot plug power delivery scheme that utilizes the precharge connector feature.



**Figure 163 – 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:

- Device insertion velocity
- Effective capacitance of the inserted device
- 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  $2.2 \times R \times C$ . Thus:

$$T = 2.2 \times R \times C_{EQ}$$

$$R = \frac{T}{2.2 \times C_{EQ}}$$

For the example charging time of 3 ms and an effective capacitance of 20 uF, the resultant precharge resistor value is approximately:

$$R = \frac{3 \text{ ms}}{2.2 \times 20 \text{ uF}} = 68 \text{ ohm}$$

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 164 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.

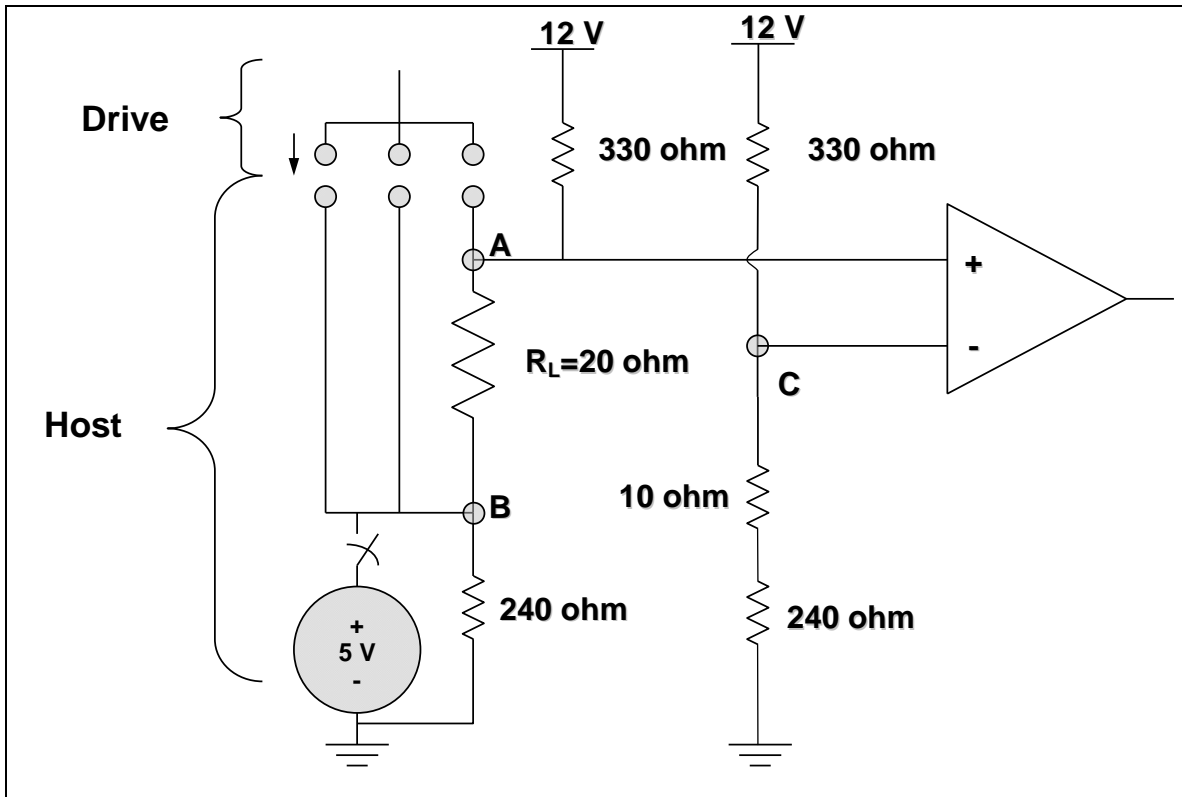


Figure 164 – Example presence detection implementation

Table 50 – Comparator voltages for alternate example presence detection circuit

	Receptacle Powered		Receptacle unpowered	
Device not present	$V_A = 5.40 \text{ V}$	$V_C = 5.17 \text{ V}$	$V_A = 5.29 \text{ V}$	$V_C = 5.17 \text{ V}$
Device present	$V_A = 5.0 \text{ V}$	$V_C = 5.17 \text{ V}$	<sup>a</sup> $V_A = 5.05 \text{ V}$	$V_C = 5.17 \text{ V}$

<sup>a</sup> 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 Gbps 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 Gbps electrical specifications aimed at short 1.5 Gbps 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 Gbps 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 Gbps 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 Gbps 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 Gbps 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 Gbps electrical specifications for internal host to device applications; and
- h) Gen3u Generation 3 electrical specifications defined at 6.0 Gbps 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 Gbps, 3.0 Gbps, or 6.0 Gbps 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 Gbps, 3.0 Gbps, or 6.0 Gbps 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 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

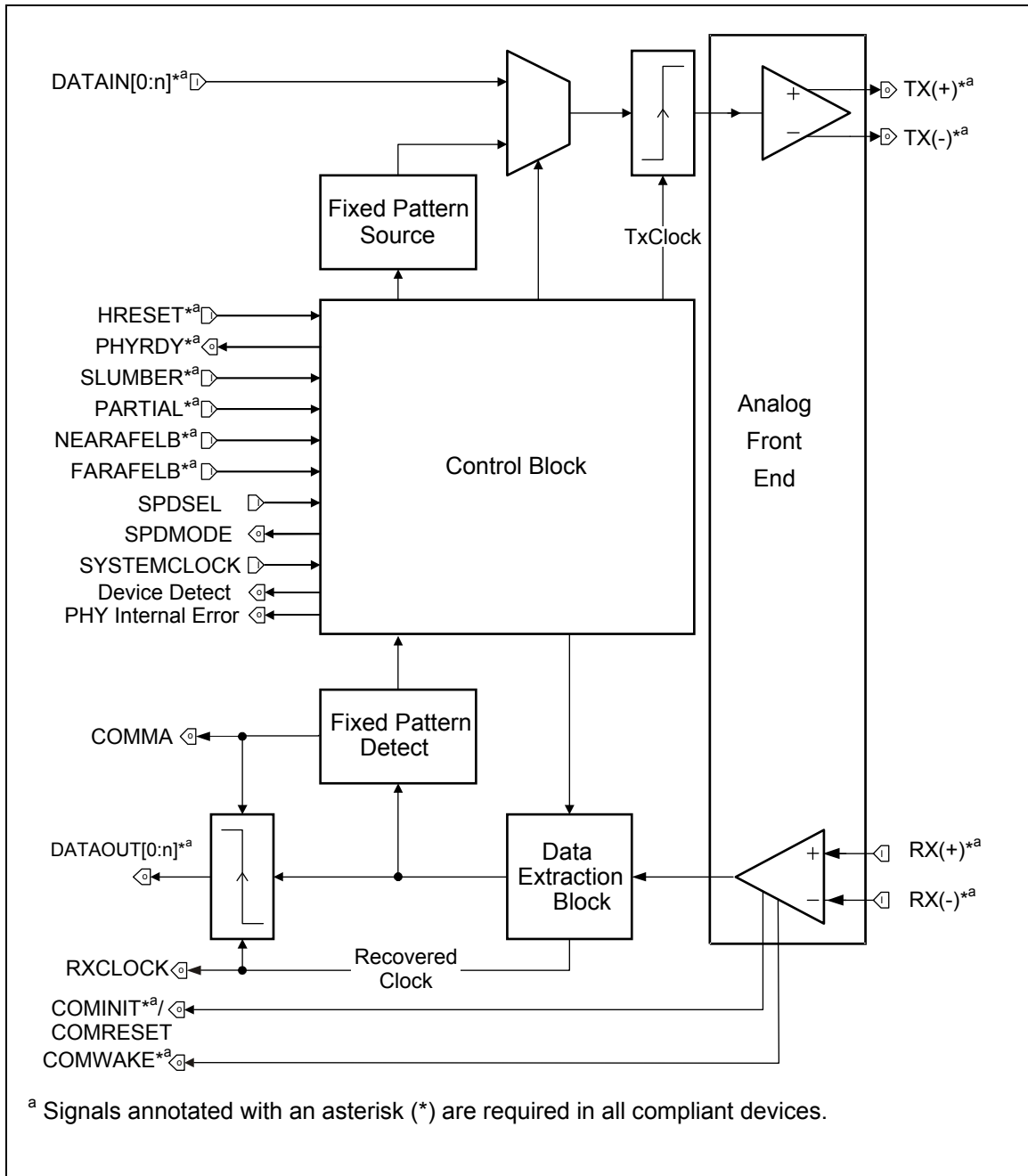


Figure 165 – Physical plant overall block diagram (informative)

### 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 <sub>P</sub> activity.
Fixed pattern detect	This block provides the support circuitry to allow proper processing of the ALIGN <sub>P</sub> 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.

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

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.

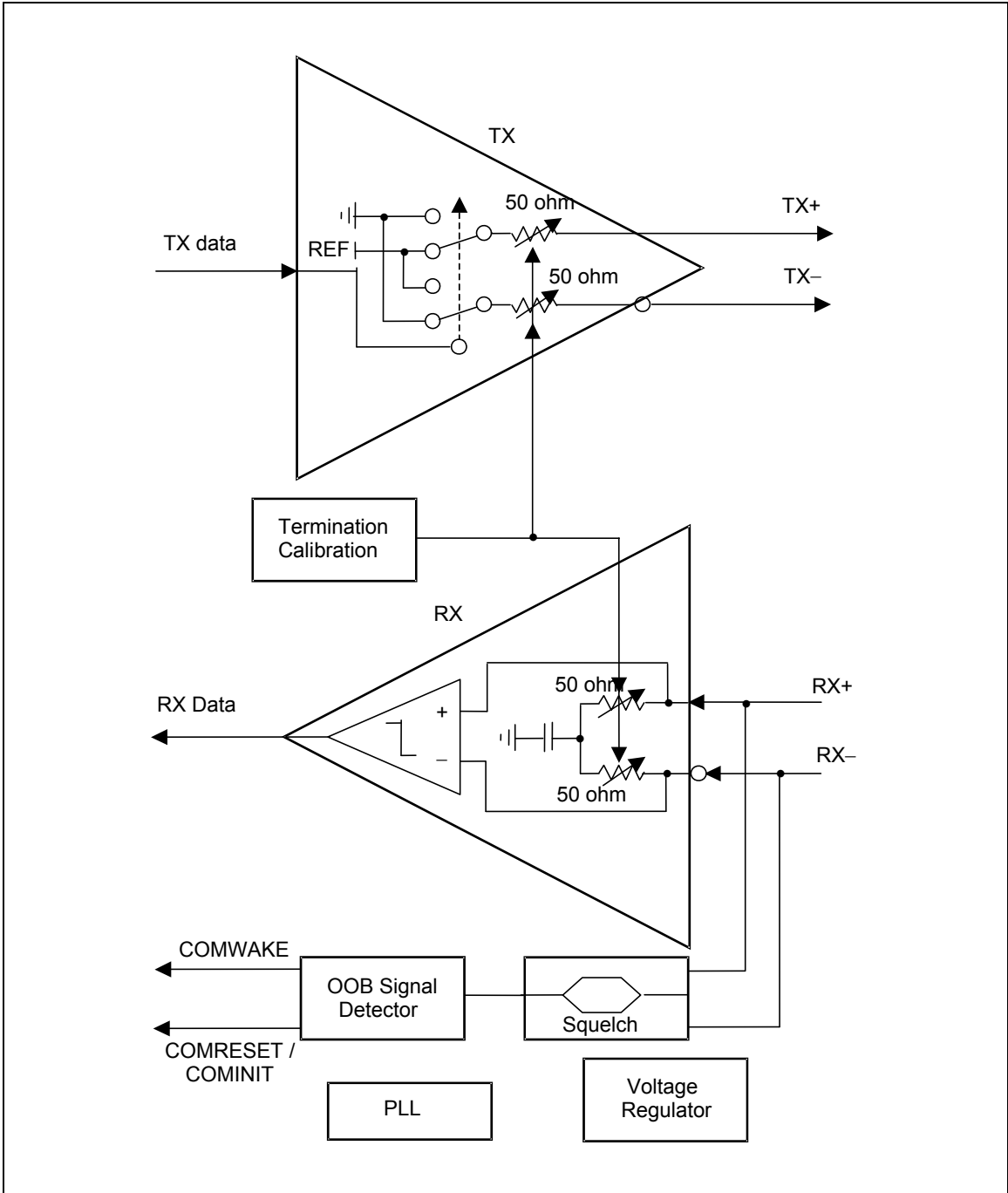
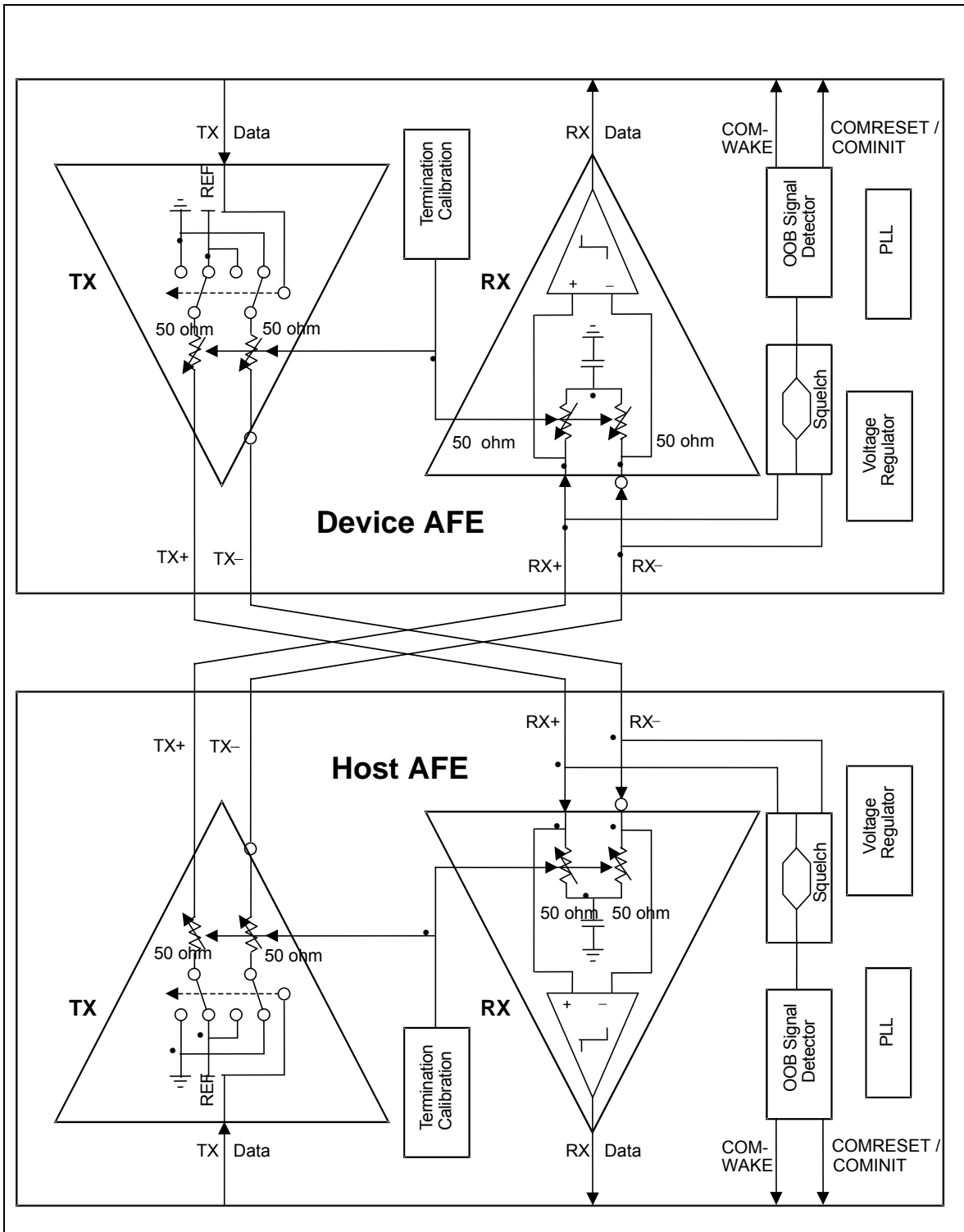


Figure 166 – Analog Front End (AFE) block diagram

#### 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 167 – 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.

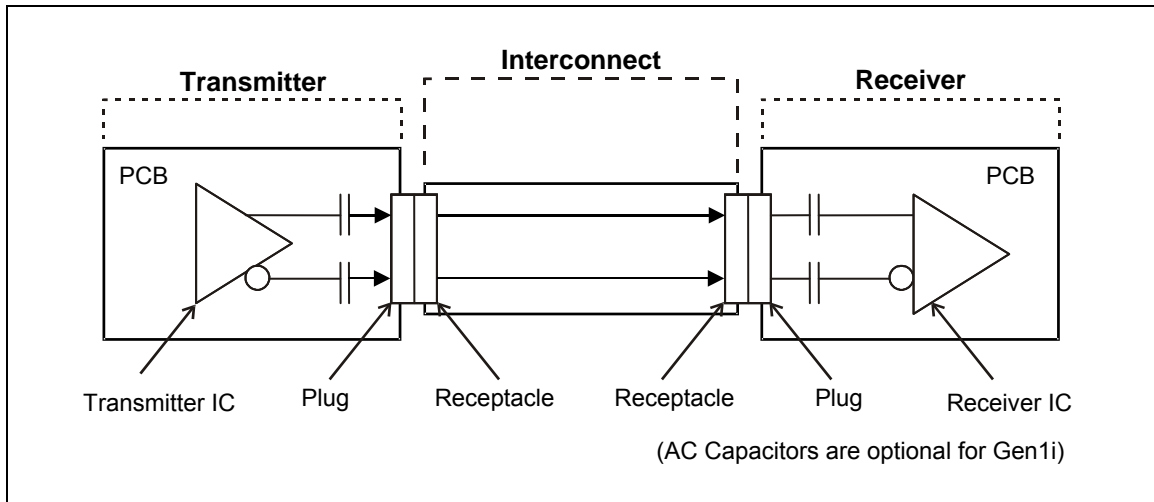
Table 52, Table 53, Table 54, Table 55, Table 56, Table 57, Table 58, and Table 59 detail the electrical requirements for SATA compliance. Each requirement as defined in 7.4.3. Jitter as defined in 7.5. Measurement methods for each specification with details as given in 7.6. See 7.7 discuss Interface 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.

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 168 – 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. 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 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 otherwise specified, all compliance measurements shall be taken through the mated connector pair.

NOTE 20 - 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, 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 (FER) 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

### 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 169; and
- b) the PCIe device shall have AC coupling capacitors on both the Tx and Rx lanes, as shown in Figure 169. The capacitance value shall be 176 nF min to 265 nF max, as defined in the PCIe spec.

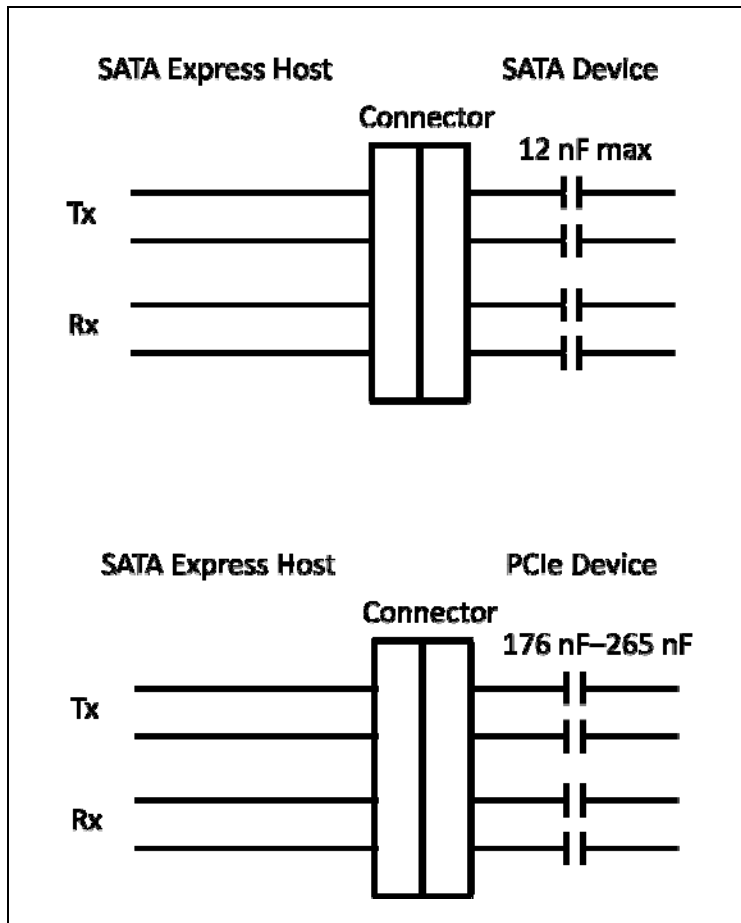


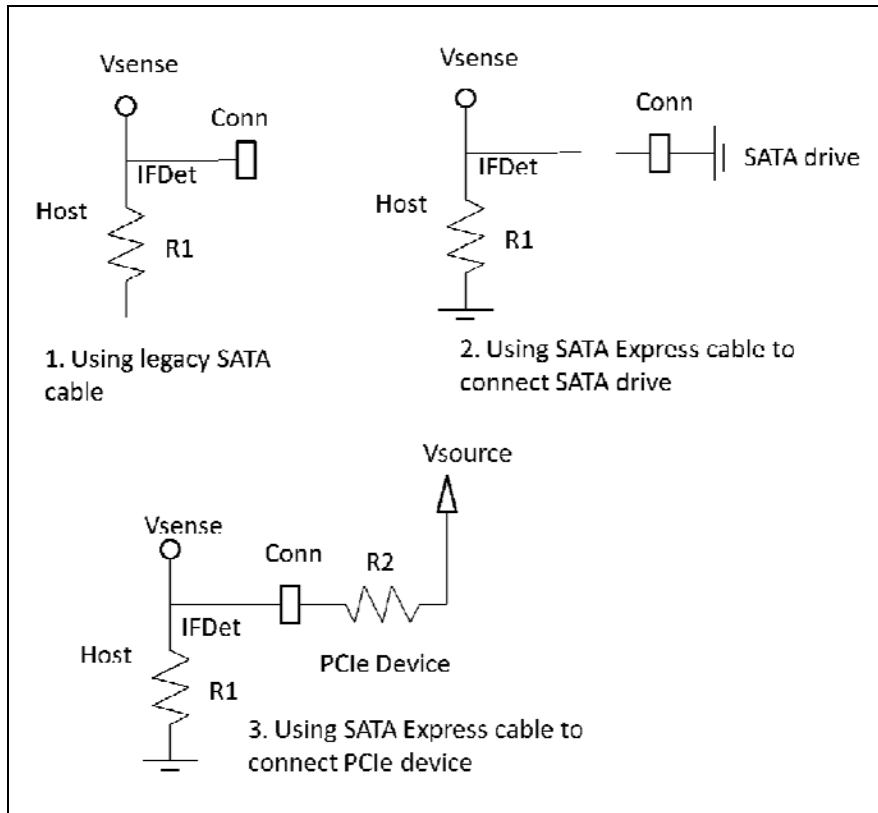
Figure 169 – AC coupling capacitor requirements

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 170 illustrates the interface detect mechanism for the cabled case.



**Figure 170 – 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 170.

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:

- 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  $\pm 40\%$ ;
- the pull-up resistor R2 on the PCIe device of 5 kohm with a relative tolerance of  $\pm 10\%$  shall be used, connecting to the 1.8 V rail with a relative tolerance of  $\pm 5\%$ ; and
- the SATA Express cable shall have a wire to connect the IFDet pins at both ends.

NOTE 21 - 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 171. The host/controller allows  $T_{delay}$  amount of time for  $V_{sense}$  to stabilize. Then it latches in the detected voltage and opens the DC path to ground thus avoiding a constant DC power drain.

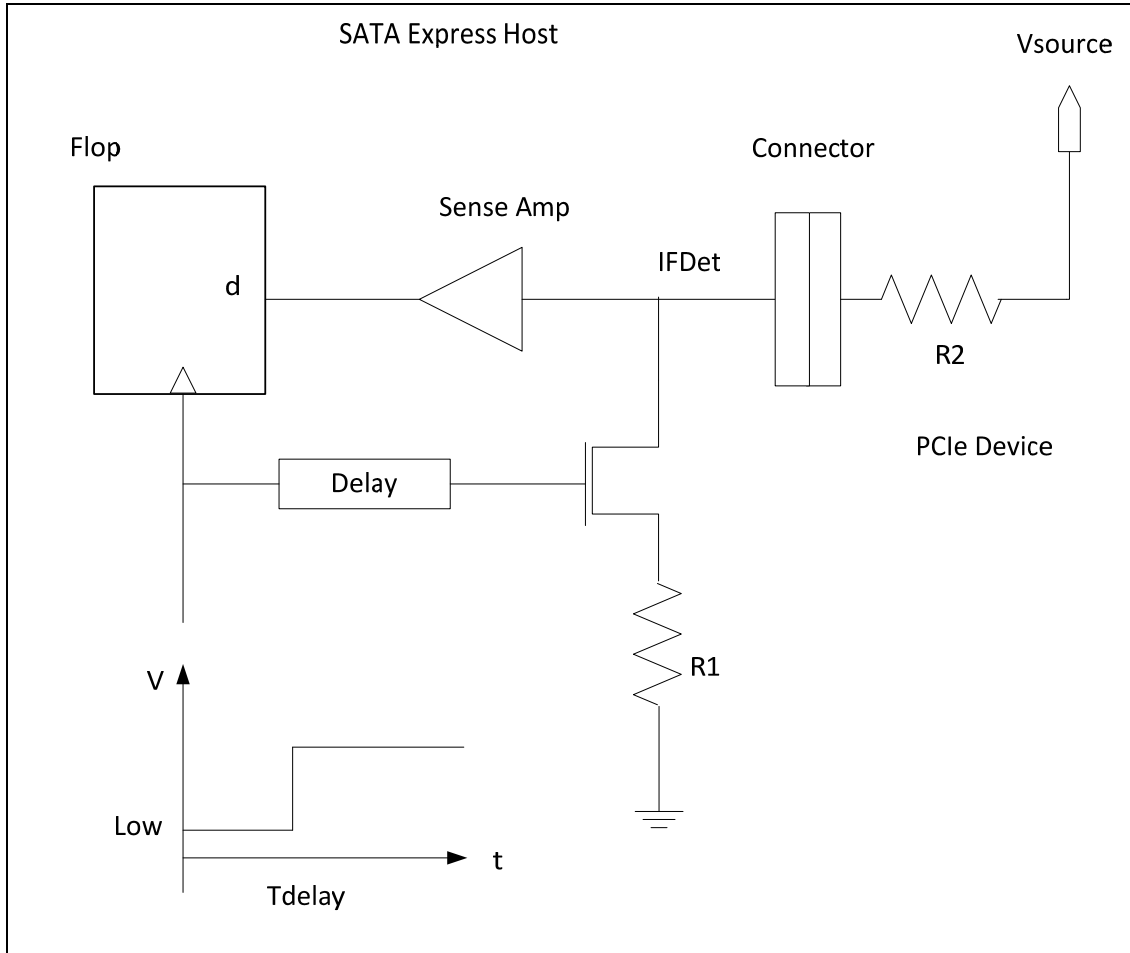


Figure 171 – Interface Detect, shutting down DC path after detection

### 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; 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 Physical 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	Gbps	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	0
T <sub>UI</sub> , Unit Interval	Ps	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 <sub>tol</sub> , 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 <sub>SSC</sub> , Spread-Spectrum Modulation Frequency	kHz	Min	30		30			30			7.4.3.1.6 7.5.4	7.6.16
		Max	33		33			33				
SSC <sub>tol</sub> , Spread-Spectrum Modulation Deviation	ppm of Fbaud	Min	-5 350		-5 350			-5 350			7.4.3.1.7 7.5.4	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 <sub>tol</sub> , Spread-Spectrum Modulation Rate	ppm/us	Max	1 250			1 250			1 250		7.4.3.1.7 7.5.4	7.6.16
V <sub>cm,dc</sub> , 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 <sub>cm,ac coupled</sub> , AC Coupled Common Mode Voltage	mV	Min	0		-	-	-	-	7.4.3.1.9	7.6.31		
		Max	2 000		-	-	-					
Z <sub>diff</sub> , Nominal Differential Impedance	ohm	Nom	100			-			100	7.4.3.1.10	7.6.28	
C <sub>ac coupling</sub> , AC Coupling Capacitance	nF	Max	12			12			12	7.4.3.1.11	7.6.19	
t <sub>settle,cm</sub> , Common Mode Transient Settle Time	ns	Max	10			-			-	7.4.3.3.8	7.4.6.3	
V <sub>trans</sub> , 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 <sub>trans</sub> , Sequencing Transient Voltage LL	V	Min	-	-	-	-	-1.2	7.4.3.1.13	7.6.32
		Max	-	-	-	-	1.2		
Z <sub>diffTx</sub> , Tx Pair Differential Impedance	ohm	Min	85	-	-	-	-	7.4.3.2.2	7.6.28
		Max	115	-	-	-	-		
Z <sub>s-eTx</sub> , 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 <sub>DD11,Tx</sub> , 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 <sub>DD11,Tx</sub> , 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 <sub>CC11,Tx</sub> , 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 <sub>DC11,Tx</sub> , 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 to 6.5 GHz	-		-	-	4		

**Table 54 – Transmitted signal requirements**

(part 1 of 4)

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	7.6.7
		Min	-		-		240		7.6.5 7.6.5.4
		Nom	500		-		-		7.6.7
		Max	600		700		-		7.6.5
		Max	-		-		900		
$V_{diffTxhost}$ , Tx Differential Host Output Voltage	mVppd	Min	400		400		-	7.4.3.3	7.6.7
		Min	-		-		200		7.6.5 7.6.5.4
		Nom	500		-		-		7.6.7
		Max	600		700		-		7.6.5
		Max	-		-		900		
$UI_{VminTx}$ , 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
$t_{20-80Tx}$ , 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

**Table 54 – Transmitted signal requirements (part 2 of 4)**

Parameter	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
$V_{cm,acTx}$ , Tx AC Common Mode Voltage	mVp-p	Max	-		50		-	7.4.3.3.6	7.6.22
$V_{cm,acTx}$ , Tx AC Common Mode Voltage	mVp-p	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
$D_{VcmOOB}$ , 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		-		

Table 54 – Transmitted signal requirements (part 3 of 4)

Parameter	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
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		-	7.5.3	7.6.10
		Max	3.1		3.1		-	7.5.3	7.6.10
Jitter Transfer Function Peaking	dB	Min	0		0		-	7.5.3	7.6.10
		Nom	0		0		-	7.5.3	7.6.10
		Max	3.5		3.5		-	7.5.3	7.6.10
Jitter Transfer Function Low Frequency Attenuation	dB	Min	69		69		-	7.5.3	7.6.10
		Nom	72		72		-	7.5.3	7.6.10
		Max	75		75		-	7.5.3	7.6.10
Jitter Transfer Function Low Frequency 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	7.5.3.4	7.6.10
		Max	-		-		6.2	7.5.3.4	7.6.10
Jitter Transfer Function Peaking (Gen3)	dB	Min	-		-		0	7.5.3.4	7.6.10
		Nom	-		-		0	7.5.3.4	7.6.10
		Max	-		-		3.5	7.5.3.4	7.6.10

**Table 54 – Transmitted signal requirements (part 4 of 4)**

Parameter	Units	Limit	Electrical Specification					Detail Cross-Ref Section	Measurement Cross-Ref Section
			Gen1i	Gen1m	Gen2i	Gen2m	Gen3i		
Jitter Transfer Function Low Frequency Attenuation (Gen3)	dB	Min	-	-	-	-	35.2	7.5.3.4	7.6.10
		Nom	-	-	-	-	38.2	7.5.3.4	7.6.10
		Max	-	-	-	-	41.2	7.5.3.4	7.6.10
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 <sup>-12</sup> ) before and after CIC, Clk-Data JTF Defined	UI	Max	-	-	-	-	0.52	7.4.3.3.13 7.5	7.6.10 7.6.12
TJ (10 <sup>-6</sup> ) before and after CIC, Clk-Data JTF Defined	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 <sup>1</sup>	Units	Limit	Electrical Specification			Detail Cross-Ref Section <sup>2,3</sup>	Measurement Cross-Ref Section <sup>2,3</sup>
			Gen1u	Gen2u	Gen3u <sup>3</sup>		
V <sub>diffTx</sub> , 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
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
V <sub>cm,acTx</sub> , Tx AC Common Mode Voltage	mVp-p	Max	100	100	120	7.4.3.3.6 7.4.3.3.7	7.6.22 7.6.23
V <sub>trans</sub> , Sequencing Transient Voltage LL	V	Min	-	-	-1.2	7.4.3.1.12	7.6.18
		Max	-	-	1.2		
Z <sub>diffTx</sub> , Tx Pair Differential Impedance	ohm	Min	85	-	-	7.4.3.2.2	7.6.28
		Max	115	-	-		

<sup>1</sup> The UHost Tx requirements are defined at the target device attachment point and the UHost channel loss is not separable from the host transmitter.

<sup>2</sup> Referenced detail and measurement sections may indicate a different connection type in the figures than what applies to the specific unit under test. 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.

<sup>3</sup> 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 <sup>1</sup>	Units	Limit	Electrical Specification			Detail Cross-Ref Section <sup>2,3</sup>	Measurement Cross-Ref Section <sup>2,3</sup>
			Gen1u	Gen2u	Gen3u <sup>3</sup>		
Z <sub>s-eTx</sub> , Tx Single-Ended Impedance	ohm	Min	40	-	-	7.4.3.2.3	7.6.29
Amp <sub>bal</sub> , 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 <sup>-12</sup> ) 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 <sup>-6</sup> ) without CIC, Clk-Data JTF Defined	UI	Max	-	-	0.46		

<sup>1</sup> The UHost Tx requirements are defined at the target device attachment point and the UHost channel loss is not separable from the host transmitter.

<sup>2</sup> Referenced detail and measurement sections may indicate a different connection type in the figures than what applies to the specific unit under test. Many figures are based on the Internal 1 meter Cabled Host to Device usage model. Specific connection type may vary, depending on the type UHost under test.

<sup>3</sup> 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	18	-		
		300 MHz to 600 MHz	10	10	14	14	-		
		600 MHz to 1.2 GHz	8	8	10	10	-		
		1.2 GHz to 2.4 GHz	3	3	8	8	-		
		2.4 GHz to 3.0 GHz	1	-	3	3	-		
		3.0 GHz to 5.0 GHz	-	-	1	-	-		



**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 <sub>DD11,Rx</sub> , 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 <sub>CC11,Rx</sub> , 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		
$R_{L_{DC11,Rx}}$ 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		-		7.6.5 7.6.14
		Max	-		-		1 000		
$V_{diffRxhost}$ , 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		-		7.6.5 7.6.14
		Max	-		-		1 000		
$t_{20-80Rx}$ , 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
			-		-		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
			-		-		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 <sub>VminRx</sub> , 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 <sub>skewRx</sub> , Rx Differential Skew	ps	Max	-		50		30	7.4.3.5.5	7.6.17
V <sub>cm,acRx</sub> , Rx AC Common Mode Voltage	mVp-p	Max	100		100		100	7.4.3.5.6	7.6.13
f <sub>cm,acRx</sub> , 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 <sub>BAUD</sub> /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 <sub>BAUD</sub> /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 ps 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 <sup>1</sup>	Measurement Cross-Ref Section <sup>1</sup>
			Gen1u	Gen2u	Gen3u		
V <sub>diffRx</sub> , Rx Differential Input Voltage	mVppd	Min	400	400	-	7.4.3.5.2	7.6.8
		Min	-	-	240		
		Max	600	700	-		
		Max	-	-	1 000		
t <sub>20-80Rx</sub> , 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)		
U <sub>VminRx</sub> , 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 <sub>skewRx</sub> , Rx Differential Skew	ps	Max	20	20	20	7.4.3.5.5	7.4.15
V <sub>cm,acRx</sub> , Rx AC Common Mode Voltage	mVp-p	Max	50	50	50	7.4.3.5.6	7.6.13 7.6.14
f <sub>cm,acRx</sub> , AC Common Mode Frequency	MHz	Min	2	2	2	7.4.3.5.7	7.6.13 7.6.14
		Max	200	200	200		
<p><sup>1</sup> Referenced detail and measurement sections may indicate a different connection type in the figures than what applies to the specific unit under test. Many figures are based on the Internal 1 meter Cabled Host to Device usage model. Specific connection type may vary depending on the type UHost under test.</p> <p><sup>2</sup> 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.</p>							

**Table 58 – Lab-Sourced Signal (for UHost receiver tolerance testing) (part 2 of 2)**

Parameter	Units	Limit	Electrical Specification			Detail Cross-Ref Section <sup>1</sup>	Measurement Cross-Ref Section <sup>1</sup>
			Gen1u	Gen2u	Gen3u		
$Z_{diffRx}$ 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
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 Defined	UI	Max	0.27	0.27	-		
TJ Clk-Data JTF Defined	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 ps 1 sigma)		
1 Referenced detail and measurement sections may indicate a different connection type in the figures than what applies to the specific unit under test. Many figures are based on the Internal 1 meter Cabled Host to Device usage model. Specific connection type may vary depending on the type UHost under test. 2 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 <sub>thresh</sub> , 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 <sub>OOB</sub> , 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	$35 \leq T < 175$					7.4.3.6.8	7.6.27
		Shall detect	$101.3 \leq T \leq 112$						
		Shall not detect	$T < 35$ or $T \geq 175$						
COMINIT/ COMRESET Gap Detection Windows	ns	May detect	$175 \leq T < 525$					7.4.3.6.9	7.6.27
		Shall detect	$304 \leq T \leq 336$						
		Shall not detect	$T < 175$ or $T \geq 525$						





### **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**

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**

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**

The modulation frequency of the Spread Spectrum frequency modulation. See further details of Spread Spectrum as defined in 7.5.4.

##### **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 7.5.4.

##### **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 172 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.

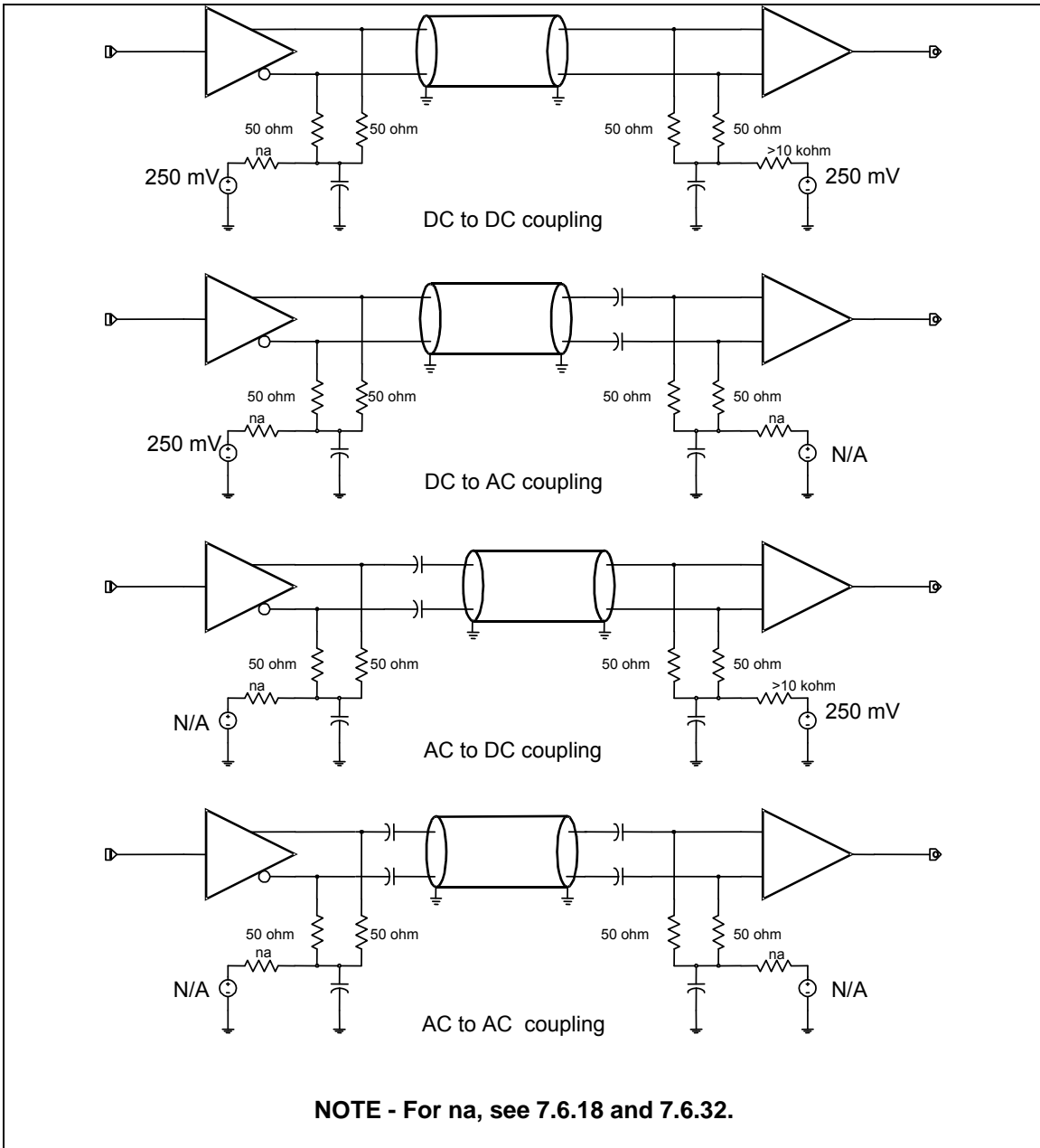
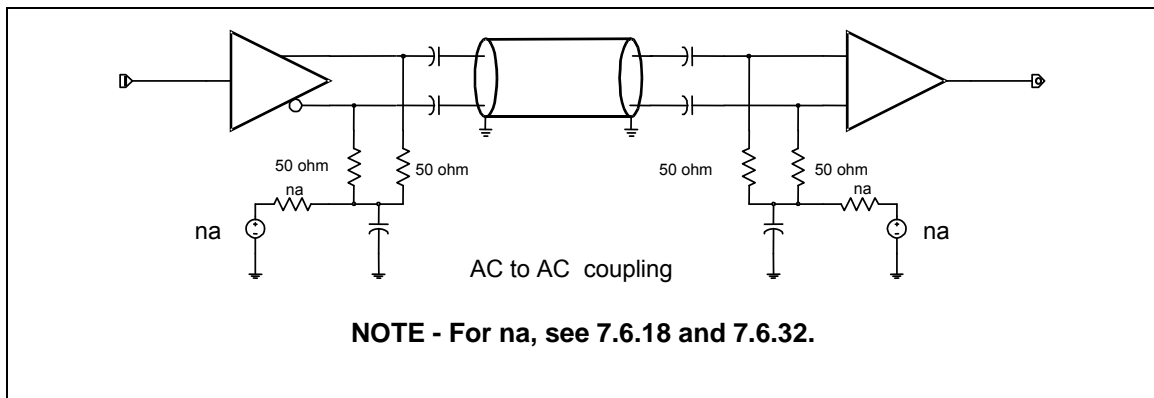


Figure 172 – Common mode biasing examples for Gen1i (informative)

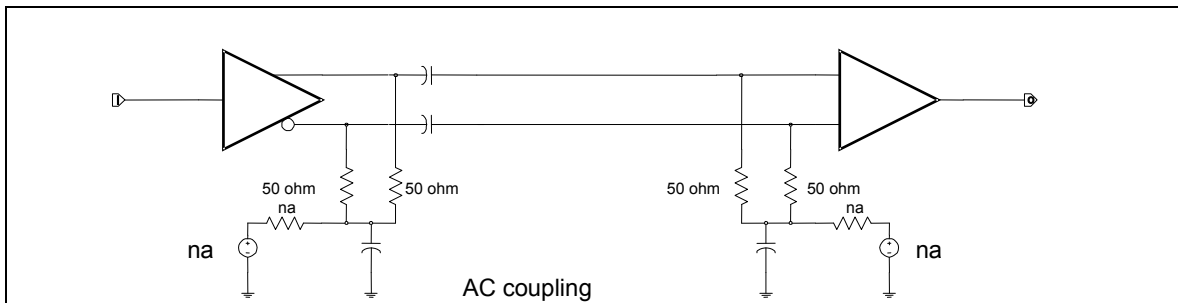
**7.4.3.1.9 AC coupled common mode voltage (Gen1m, Gen1u, Gen2i, Gen2m, Gen2u, Gen3i, Gen3u)**

The SATA interfaces, defined as Gen1i, may be AC or DC coupled as shown in Figure 172. The SATA interfaces defined as Gen1m, Gen1u, Gen2i, Gen2m, Gen2u, Gen3i, and Gen3u shall be AC coupled. Figure 173 shows an example of a fully AC coupled system. The SATA MicroSSD application shall use the configuration shown in Figure 174.

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 conditionsthat 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.



**Figure 173 – Common mode biasing for Gen1m, Gen1u, Gen2i, Gen2m, Gen2u, Gen3i, and Gen3u**



**Figure 174 – 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, and 7.4.3.4.4.

The physical size of the capacitor should be as small as practical to reduce the capacitance to ground. Body sizes larger than 0603 should be avoided, as they are likely to result in a failure of the return loss requirements in 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 complete, 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 ps 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 ps 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. 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.

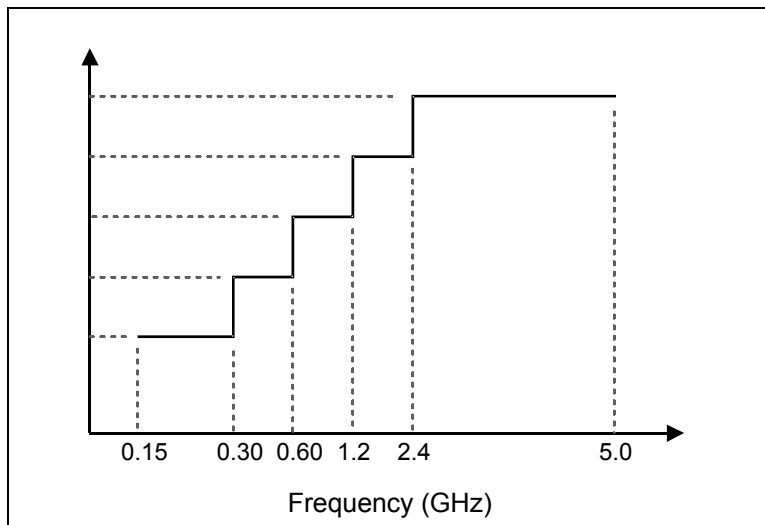


Figure 175 – Differential return loss limits

#### 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.

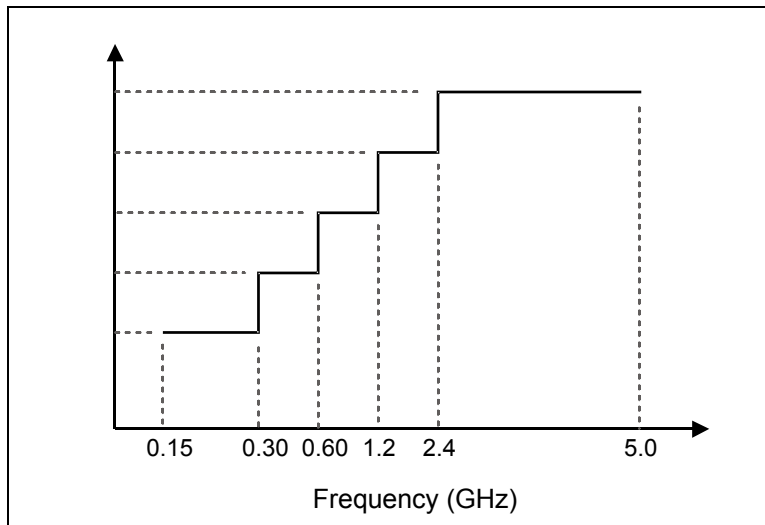


Figure 176 – Common mode return loss limits

#### 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 100 ohm impedance level to differential mode reflected power at a 25 ohm impedance level. The impedance balance is a bound on the coupling between common and differential modes.

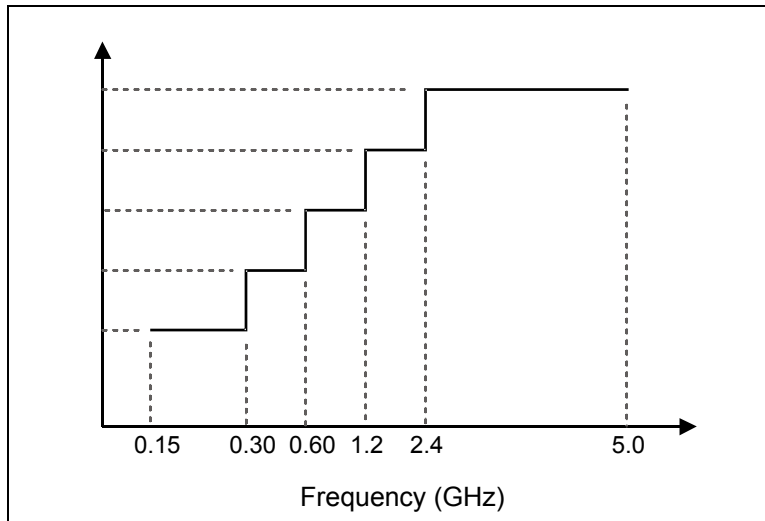


Figure 177 – Impedance balance limits

#### 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 178, 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.

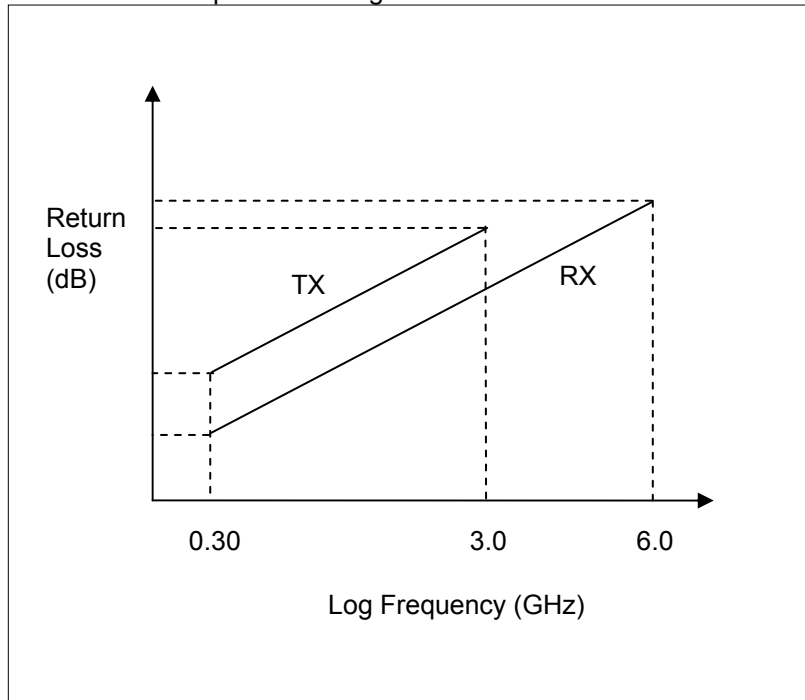


Figure 178 – Differential return loss limits, Gen3i, Tx and Rx

### 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 7.4.

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 179. The rise and fall time requirement  $t_{r/f}$  applies to differential transitions (Tx+ – Tx-), for both normal and OOB signaling.

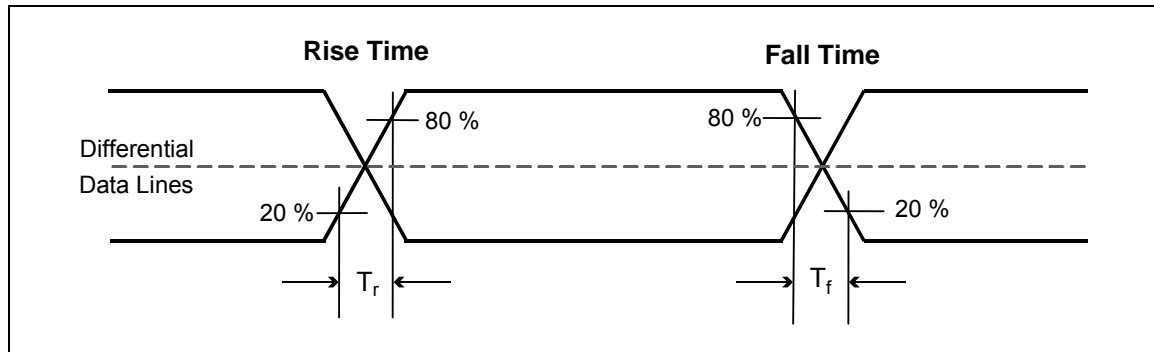


Figure 179 – Signal rise and fall times

#### 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. 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.



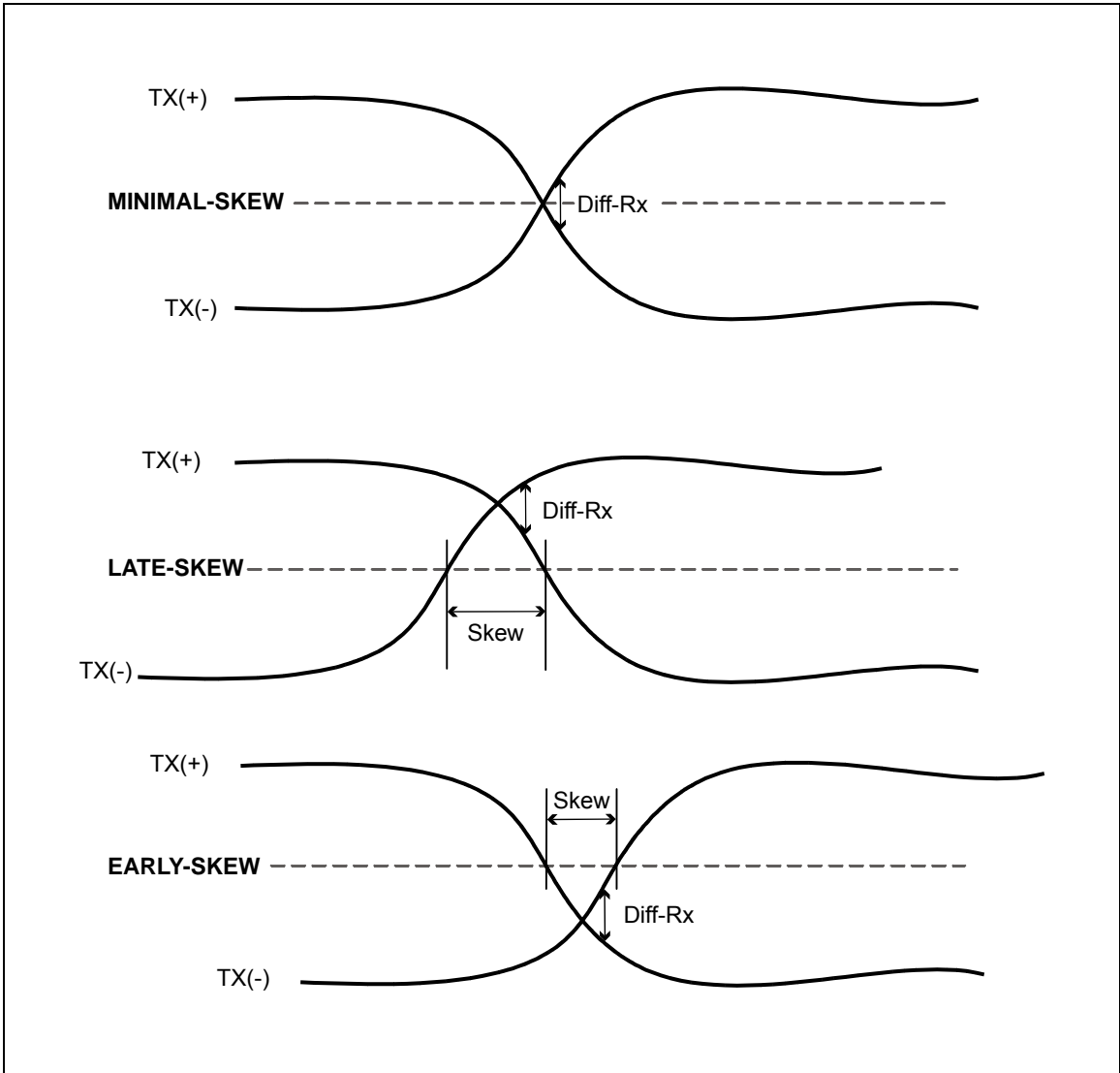


Figure 180 – Tx intra-pair skew

#### 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.

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  $1 / (2 \times 106 \text{ ns}) = 4.7 \text{ MHz}$ . The AC error voltage propagates through the interconnect and causes an offset in the receiver OOB detector.

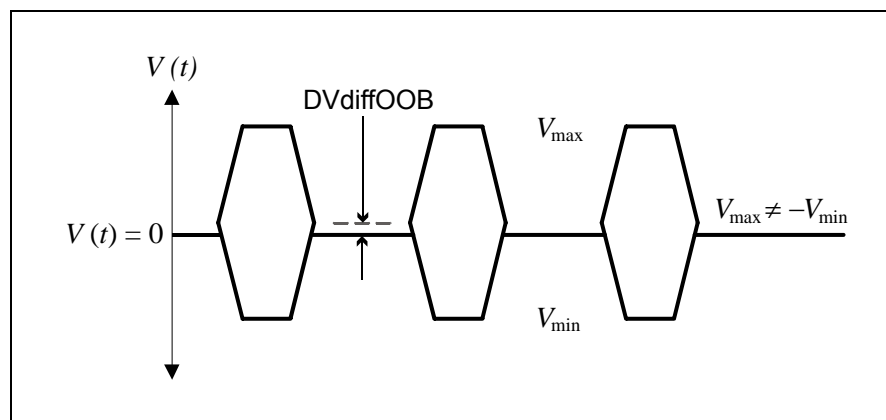


Figure 181 – OOB differential delta (at compliance point with AC coupling)

#### 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:

$$\text{absolute value } (Tx+, \text{rise} - Tx-, \text{fall}) / \text{average}$$

where

$$\text{average} = (Tx+, \text{rise} + Tx-, \text{fall}) / 2$$

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:

$$\text{absolute value } (Tx+, \text{fall} - Tx-, \text{rise}) / \text{average}$$

where

$$\text{average} = (Tx+, \text{fall} + Tx-, \text{rise}) / 2$$

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:

$$\text{absolute value } (Tx+ \text{ amplitude} - Tx- \text{ amplitude}) / \text{average}$$

where

$$\text{average is } (Tx+ \text{ amplitude} + Tx- \text{ amplitude}) / 2$$

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.

### **7.4.3.4 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 ps 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 ps 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**

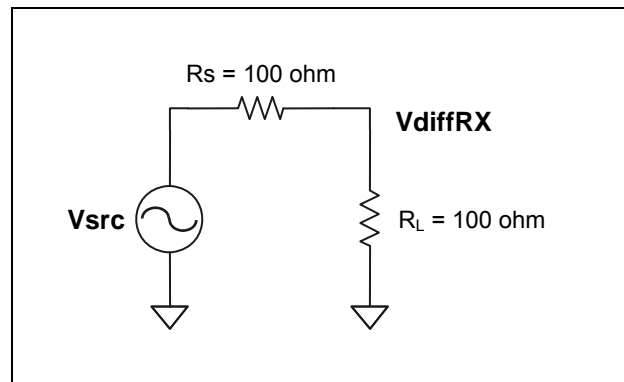
#### **7.4.3.5.1 Receiver tolerance testing scope**

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 182). 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 182 – 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  $t_{20-80Rx}$  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 183 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.

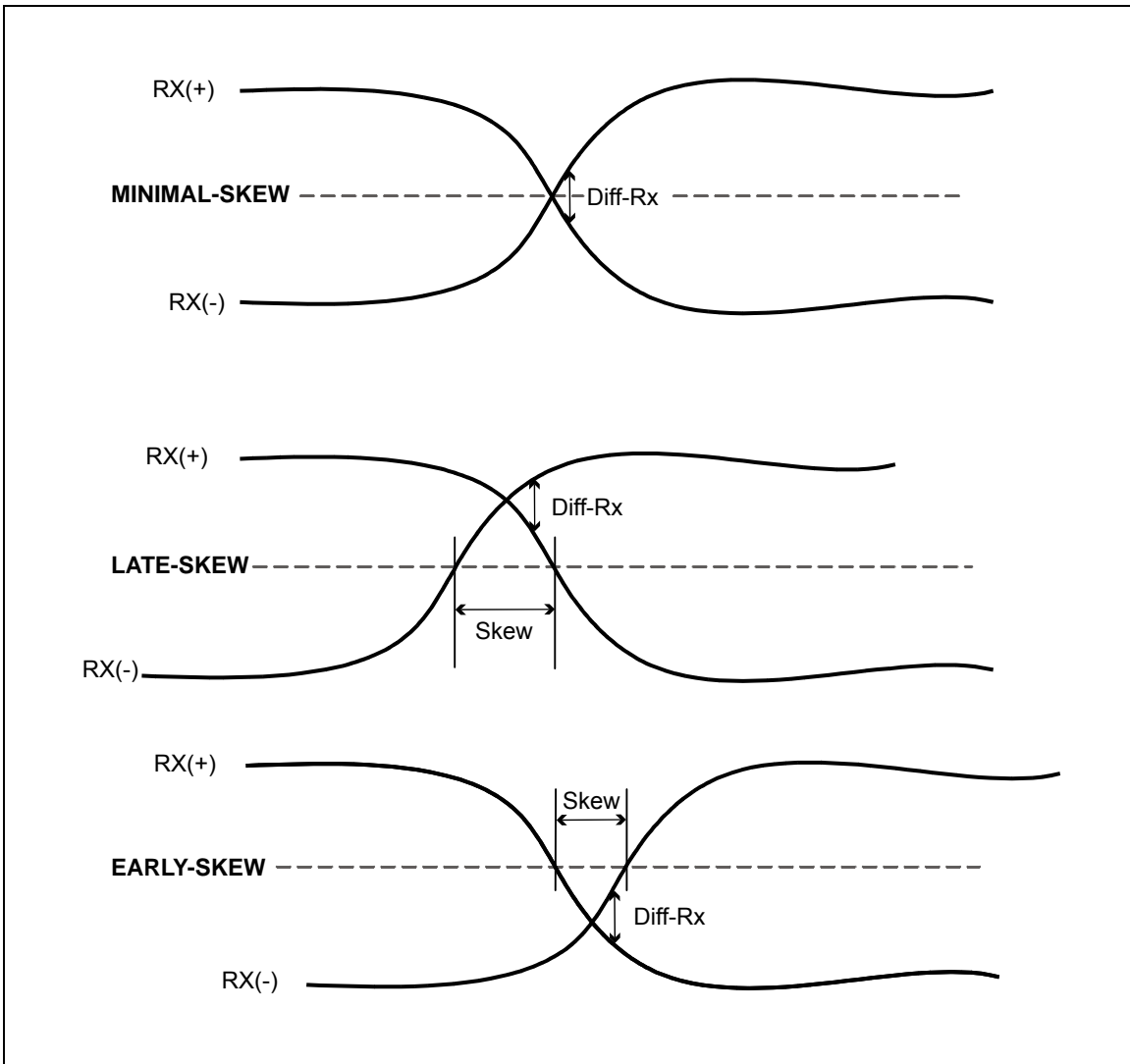


Figure 183 – Rx intra-pair skew

#### 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, whenif subjected to a sinusoidal common mode interfering signal with peak-to-peak voltage  $V_{cmRx,ac}$  defined in Table 57 and Table 58 and swept across the frequency range,  $f_{cm,acRx}$ , defined in Table 57 and Table 58 at a sweep rate period no shorter than 33.33  $\mu s$ .

#### 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 operate is the jitter tolerance specification provided in Table 57 and Table 58 and section according to 7.6.13 describes the measurement 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<sub>p</sub> 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<sub>OOB</sub> periods and then a 0 for 2 UI<sub>OOB</sub> 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.

$V_{\text{diffRx}}$  signals less than the minimum  $V_{\text{thresh}}$  defined in Table 59 shall not be detected as activity. Signal levels greater than the maximum  $V_{\text{thresh}}$  defined in Table 59 shall be detected as activity.

#### **7.4.3.6.4 UI during OOB signaling (UI<sub>OOB</sub>)**

Average data period during OOB burst transmission (at Gen1 speed with a relative tolerance of  $\pm 3\%$  of nominal).

#### **7.4.3.6.5 COMINIT/COMRESET and COMWAKE transmit burst length**

Burst length in ns, 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 ns, 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 ns, 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 (shall 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 through 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**

<b>Loopback modes</b>	<b>Required or Optional</b>
Far-End Retimed	Required
Far-End Analog	Optional
Near-End Analog (Effectively Retimed)	Optional



### 7.4.4.2 Far-end retimed

Figure 184, 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<sub>P</sub> 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.

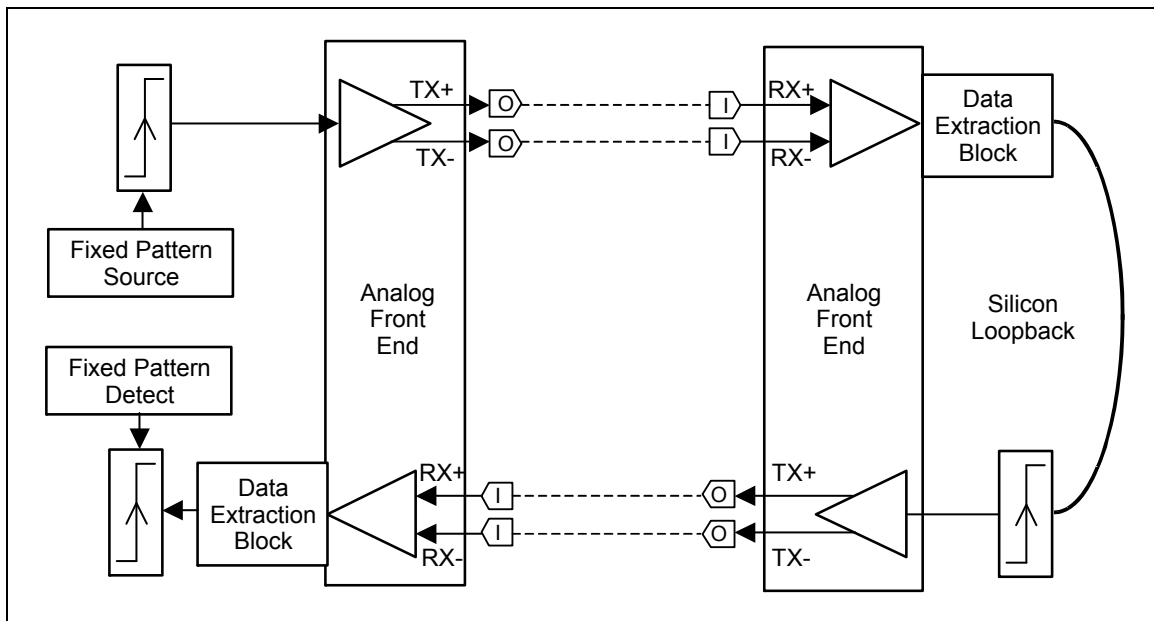


Figure 184 – Far-End Retimed Loopback

### 7.4.4.3 Far-end analog (optional)

Figure 185 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.

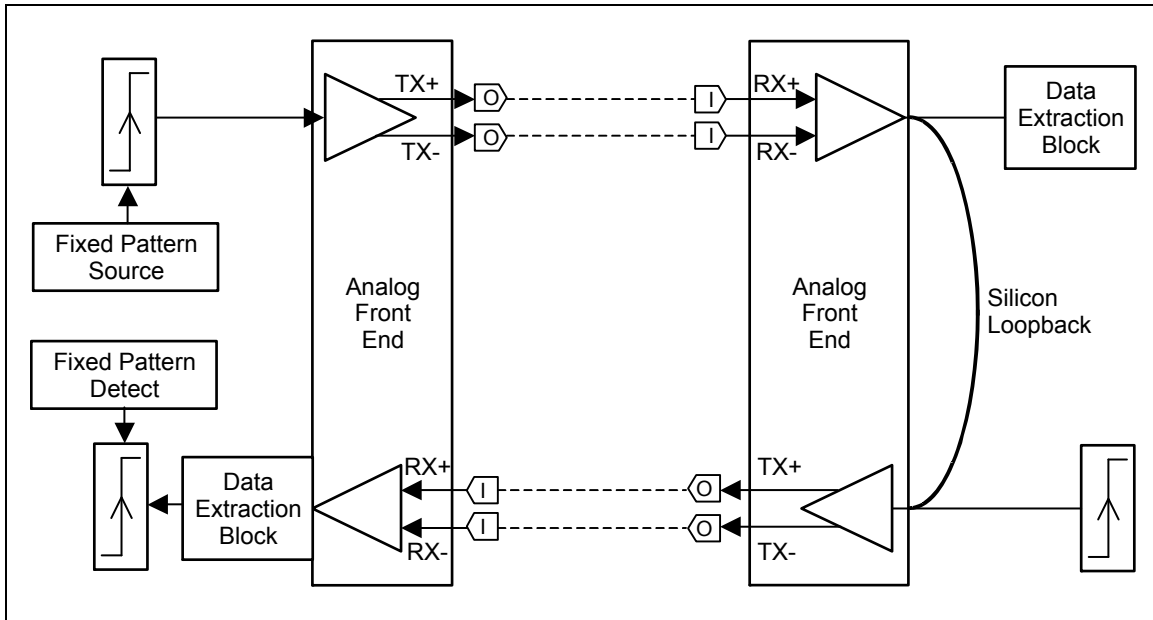


Figure 185 – Far-End Analog Loopback

#### 7.4.4.4 Near-end analog (optional)

Figure 186 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.

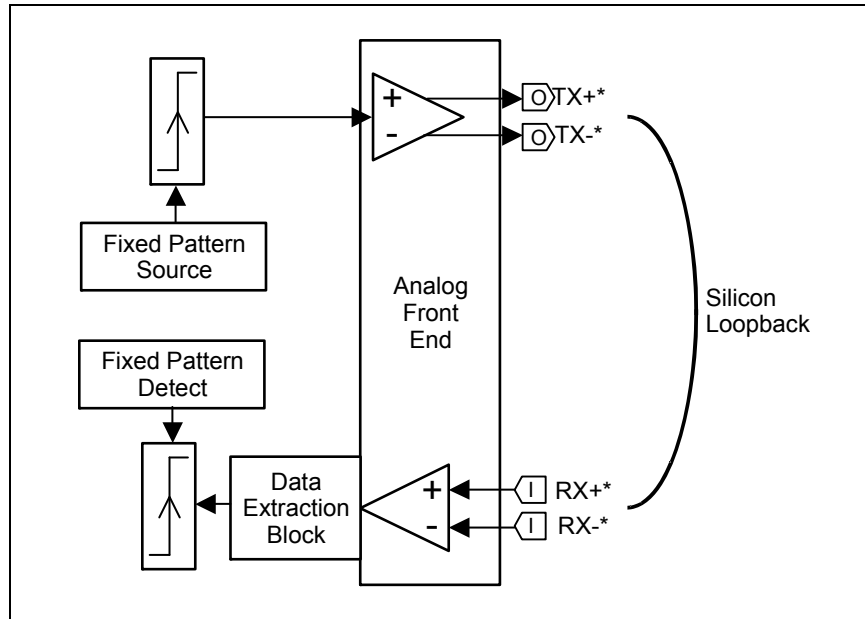


Figure 186 – Near-End Analog Loopback

### 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<sub>P</sub> primitive, the specified pattern as data content, and trailing CRC and EOF<sub>P</sub> primitive. There is no suppression of the two consecutive ALIGN<sub>P</sub> 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<sub>P</sub> leading primitive, the Dword header containing the FIS Type indicating a Data FIS, the specified test pattern, the calculated CRC, and the trailing EOF<sub>P</sub>, as shown in Figure 187. To generate these patterns on the SATA link, scrambling needs to be taken into account.

NOTE 22 - Note that the cited patterns should appear on the wire.

SOF <sub>P</sub>	Data FIS Header	Specified Test Pattern	CRC	EOF <sub>P</sub>
------------------	-----------------	------------------------	-----	------------------

**Figure 187 – 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.

The test patterns illustrated in the following sections are indicated to start with negative running disparity for illustrative purposes only in order to convey the encoded 8b/10b patterns transmitted for each sequence.

### 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	0001 b	1110b	0101b	1100 b	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	0111 b	1000b	1110b	0001 b	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	1100 b	0111b	1000b	1110 b	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)-			+
	0111 b	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)-			+
	1000 b	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)-			+
	1000 b	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)-			+
	0011 b	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 or positive.

**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 $4 \times 512 = 2\,048$ Dwords Short version total $4 \times 32 = 128$ Dwords											



**Table 64 – High Transition Density Pattern (HTDP) starting with RD+**

<b>Transmission Order</b> →												
+	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 $4 \times 512 = 2\,048$ Dwords Short version total $4 \times 32 = 128$ 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 or positive.

**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	
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)+		D11.5(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	
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.											
Long version total $2 + (2 \times 1\,023) = 2\,048$ 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: $2 + (2 \times 63) = 128$ 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	
<p>Above Dword is repeated a total of 1 023 times. Above Dword is repeated a total of 63 times for short version.</p>											
+	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	
<p>Above Dword is repeated a total of 1 023 times. Above Dword is repeated a total of 63 times for short version.</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	
<p>Above Dword is repeated a total of 1 023 times. Above Dword is repeated a total of 63 times for short version.</p>											
-	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	

<p>Long version total: <math>2 + (2 \times 1\,023) = 2\,048</math> 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</p> <p>Short version total: <math>2 + (2 \times 63) = 128</math> Dwords  255 bytes of D11.5, 255 bytes of D20.2</p>
--

### 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 or positive.

**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 $1 \times 2\,048 = 2\,048$ Dwords. Short version total $1 \times 128 = 128$ 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 $1 \times 2\,048 = 2\,048$ Dwords. Short version total $1 \times 128 = 128$ 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 or positive.

**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 $2 \times 1\,024 = 2\,048$ Dwords. Short version total $2 \times 64 = 128$ 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 $2 \times 1\,024 = 2\,048$ Dwords. Short version total $2 \times 64 = 128$ 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 23 - Note that the cited patterns should appear on the wire.

NOTE 24 - 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)

<b>Transmission Order</b> →												
-	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		
Above Dword is repeated a total of 64 times for long version. Above Dword is repeated a total of 4 times for short version.												
-	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		
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 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	1101 b	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.											
<p>Long version total 2 048 Dwords total</p> <p>256 DW SSOP</p> <p>256 DW HTDP (64 DW, 64 DW, 64 DW, 64 DW)</p> <p>512 DW LTDP (1 DW, 1 DW, 509 DW, 1 DW)</p> <p>256 DW LBP ((1 DW, 1 DW) × 128)</p> <p>512 DW LFSCP (255 DW, 1 DW, 255 DW, 1 DW)</p> <p>256 DW HTDP (64 DW, 64 DW, 64 DW, 64 DW)</p> <p>Short version total 128 Dwords total</p> <p>16 DW SSOP</p> <p>16 DW HTDP (4 DW, 4 DW, 4 DW, 4 DW)</p> <p>32 DW LTDP (1 DW, 1 DW, 29 DW, 1 DW)</p> <p>16 DW LBP ((1 DW, 1 DW) × 8)</p> <p>32 DW LFSCP (15 DW, 1 DW, 15 DW, 1 DW)</p> <p>16 DW HTDP (4 DW, 4 DW, 4 DW, 4 DW)</p>											

**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)**

<b>Transmission Order</b> →													
+	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)+			D24.3(78h)+			+
	0011b	0011b	0011b	0011b	0011b	0011b	0011b	0011b	0011b	0011b	0011b	0011b	
	3h	3h	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	0101b	0101b	
	5h	5h	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	1001b	1001b	
	9h	9h	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 25 - 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:

- according to 7.4.5.3 the COMP pattern is framed;
- 2 ALIGN<sub>P</sub> primitives inserted every 256 Dwords; and
- a short Inter Gap region is introduced before and after the SOF<sub>P</sub>/EOF<sub>P</sub> to ensure that if repeated sequentially by a generator the 256 Dword ALIGN<sub>P</sub> 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)

<b>Transmission Order</b> →											
+	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 <sub>P</sub> .											
+	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 <sub>P</sub> 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 <sub>P</sub> 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 <sub>P</sub> with RD+ All together 5 DW X_RDY <sub>P</sub>											
-	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 <sub>P</sub>											
+	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. ALIGN <sub>p</sub>											
-	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. ALIGN <sub>p</sub>											
-	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)**

<b>Transmission Order</b> →												
-	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 <sub>p</sub>												
-	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 <sub>p</sub>												
-	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 <sub>p</sub>											
-	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 <sub>p</sub>											
-	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 <sub>p</sub>												
+	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 <sub>P</sub>												
-	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 <sub>P</sub>												
+	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 <sub>P</sub>												
-	K28.3(7Ch)-		D21.5(B5h)+			D24.2(58h)+			D24.2(58h)-			+
	0011b	1100b	1110b	1010b	1010b	1100b	0001b	0111b	0011b	0101b		
	3h	Ch	Eh	Ah	Ah	Ch	1h	7h	3h	5h		
Above 2 Dwords are repeated a total of 2 times. WTRM <sub>P</sub> (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 <sub>P</sub> (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 <sub>P</sub>											
-	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 <sub>P</sub>											
+	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 <sub>P</sub> 7 DW HFTP Junk data/fill 5 DW X_RDY <sub>P</sub> 1 DW S <sub>OF</sub> <sub>P</sub> 1 DW Data FIS Header 256 DW SSOP with 2 DW ALIGN <sub>P</sub> 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 <sub>P</sub> (1 DW) WTRM <sub>P</sub> (4 DW) SYNC <sub>P</sub> (2 DW) CONT <sub>P</sub> (2 DW) HFTP (214 DW) Junk data/fill ALIGN <sub>P</sub> (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 / 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 / removal of a Device into / 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 removal of a 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. 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 26 - This 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. Regardless of the above definitions, the removal of a device that is still rotating, is not recommended and should be prevented by the system designer.

### 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)

It is a requirement that the Hot Plug Capable SATA component is designed to handle hot plug events; this informative section highlights the maximum transient events encountered during hot plug operations. An example is presented 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  $V_{cm}$  and the minimum single-ended impedance of 42.5 ohm. Hence the worst possible surge current is able to be  $2\text{ V} / 42.5\text{ ohm} = 47\text{ mA}$ . The duration of this current is limited by the time constant,  $C \times R_{tx} \sim 0.5\text{ us}$ . This current should be further reduced by supplying common mode termination at the victim end, assuming ESD diodes do not turn on.

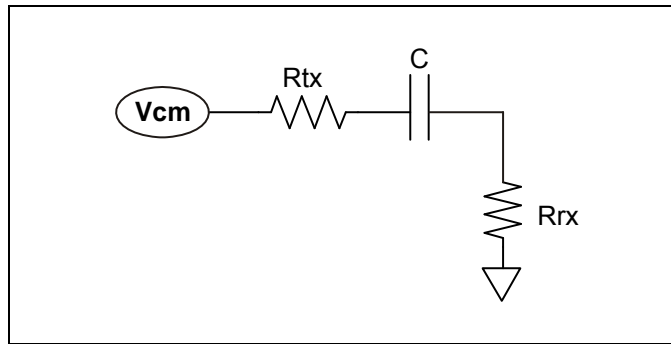


Figure 188 – Example circuit for common mode transients

$$\begin{aligned} 0\text{ V} < V_{cm} < 2\text{ V} \\ 21.25\text{ ohm} < R_{tx} < 40\text{ ohm} \\ 21.25\text{ ohm} < R_{rx} < 40\text{ ohm} \\ C < 0.024\text{ uF (Two } 0.012\text{ uF capacitors in parallel)} \end{aligned}$$

The maximum voltage step that may be transmitted to the "victim" end by a transient at the "aggressor" end is the maximum  $V_{cm}$ . 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)

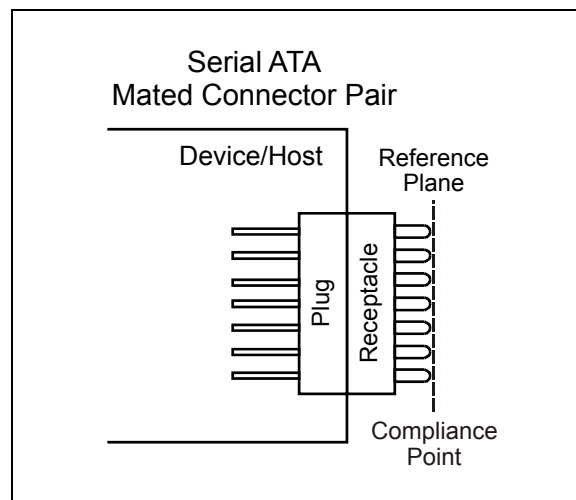
For all Serial ATA backplane systems, the device canister or enclosure should provide sufficient electrical bonding such that electrostatic potential is discharged from the device body ground to the enclosure ground prior to the connector mating. It is strongly advised for all Serial ATA backplane systems, 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

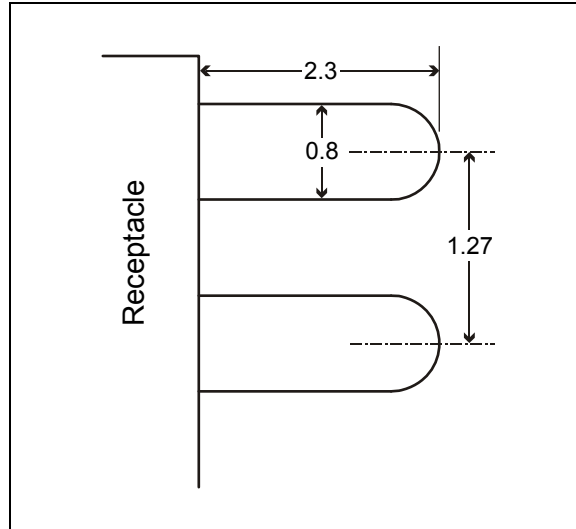
The compliance point for receiver and transmitter is at the device/host I/O including the mated connector pair.

Figure 189 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 190.



**Figure 189 – 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 190 – Mated connector pair, pin tail detail**

#### **7.4.7.2 mSATA connector connection definition**

The compliance points of mSATA are shown in Figure 191. The same concept of compliance point of SATA extends to the mSATA application. The detailed physical pin dimensions are shown in Figure 192.

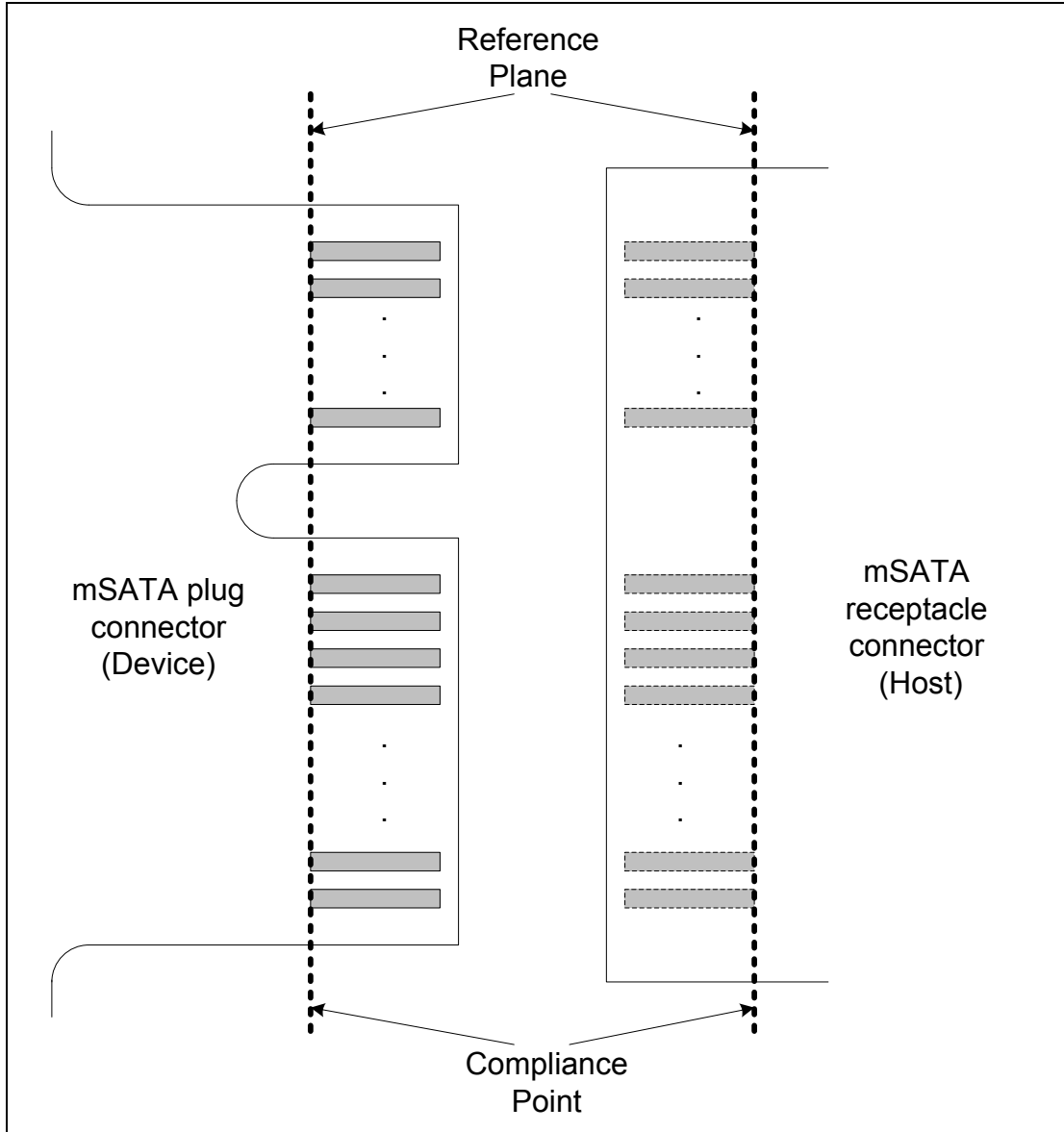


Figure 191 – Mated connector pair for mSATA



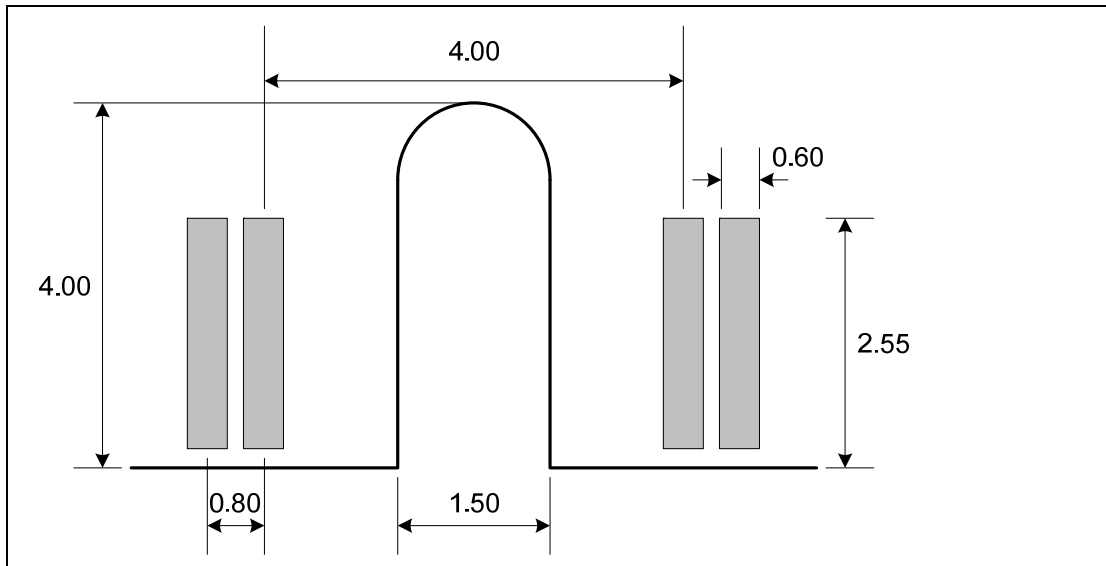


Figure 192 – mSATA connector pin detail

### 7.4.7.3 M.2 connector compliance point

The compliance points of M.2 are shown in Figure 193.

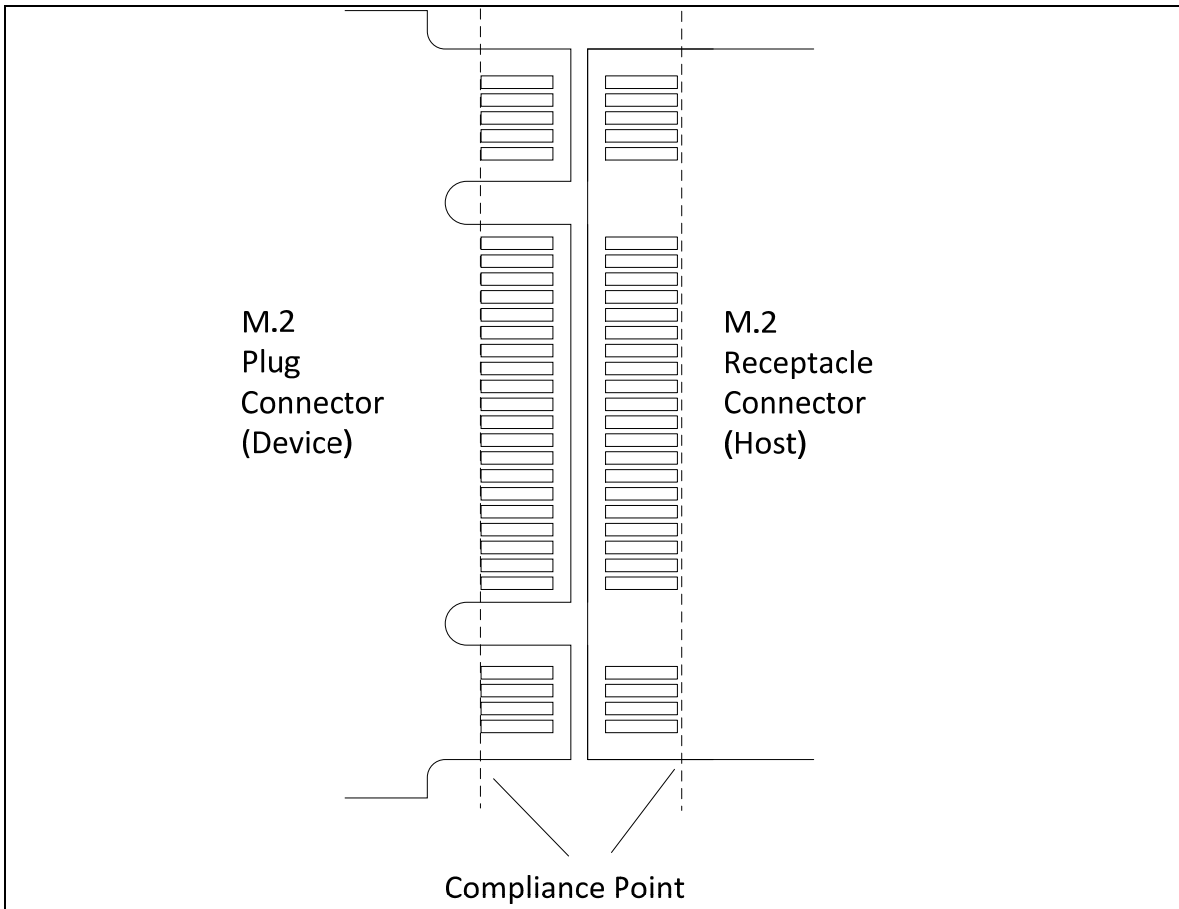


Figure 193 – M.2 compliance point

## 7.4.8 Compliance interconnect channels (Gen3i, Gen3u)

### 7.4.8.1 Compliance interconnect channels 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. 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). Any linear, passive, differential two-port (e.g., a SATA cable) with loss greater than the TCTF at all frequencies and that meets the ISI loss constraint (defined below) is defined to be a CIC (See 7.4.8.2).

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  $V_{diffTx}$  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 27 - 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.

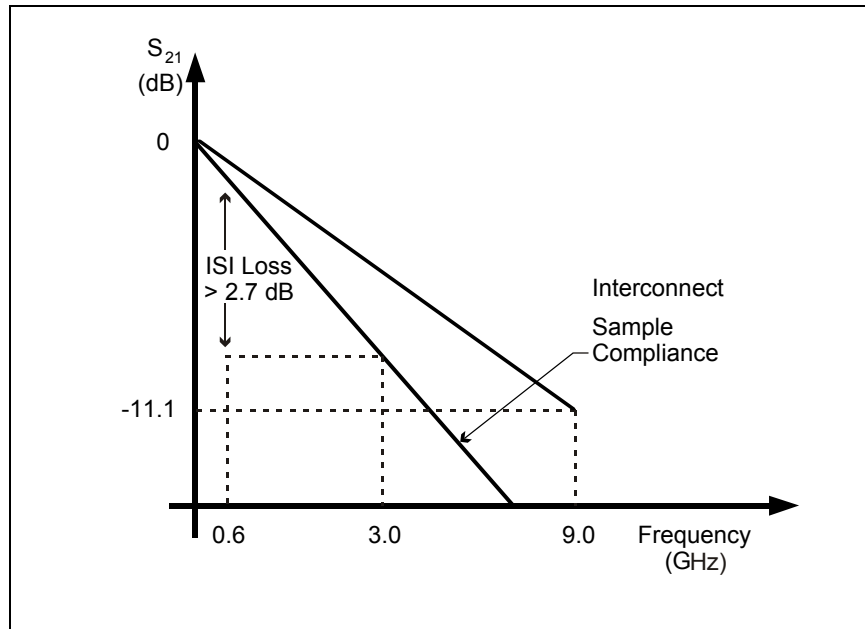
The transmission magnitude response,  $|S_{21}|$ , of the Gen3i TCTF satisfies the following two inequalities

$$|S_{21}| \leq -20 \times \log_{10}(e) \times ((3.0 \times 10^{-6} (f^{0.5})) + (1.1 \times 10^{-10} (f))) \text{ dB}$$

for 50 MHz < f < 9.0 GHz, (f expressed in Hz),

$$|S_{21}| \text{ at 600 MHz} - |S_{21}| \text{ at 3 000 MHz} > 2.7 \text{ dB}$$

NOTE 28 - 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 194 – Compliance Interconnect Channel loss for Gen3i**

The second constraint, termed ISI loss, may be motivated as follows,  $|S_{21}|$  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,  $|S_{21}|$  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:

- the peak-peak voltage (established by the long run lengths); and
- 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

The TCTF defines the worst-case loss exclusive of the two SATA connectors in the path from transmitter to receiver. However, the loss due to these two connectors shall be included in the transmitter characterization. That is, the transmitter shall be tested with the TCTF-defined loss plus two mated SATA connector pairs.

For a CIC implemented with SubMiniature version A (SMA) connectors, it is seen in Figure 218 that the addition of a SATA adapter (plug) following the CIC and driving into a Laboratory Load provides the required total loss of TCTF (embodied in the CIC loss) plus the loss of two SATA connectors.

#### 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 unable 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; 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.

## HIGH SPEED SERIALIZED AT ATTACHMENT

### Serial ATA International Organization

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 JTF (Jitter Transfer Function) 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 as 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  $\pm 10\%$ . 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  $\pm 10\%$ .

### 7.5.3.3 Gen1i, Gen1m, Gen1u, Gen2i, Gen2m, and Gen2u informative comments

Typically a CLTF -3 dB corner frequency of  $f_{\text{BAUD}}/500$  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, should shift 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 JTF (Jitter Transfer Function) 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 as 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  $4.2 \text{ MHz} \pm 2 \text{ MHz}$ . 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  $\pm 1 \%$  Jitter Transfer Function Low Frequency Attenuation Measurement Frequency (Gen3) in Table 54 shall be  $38.2 \text{ dB} \pm 3 \text{ dB}$  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  $\pm 10 \%$ . 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  $\pm 10 \%$ . 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 195. 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:

$$0 < t < \frac{1}{2 \times f_m}$$

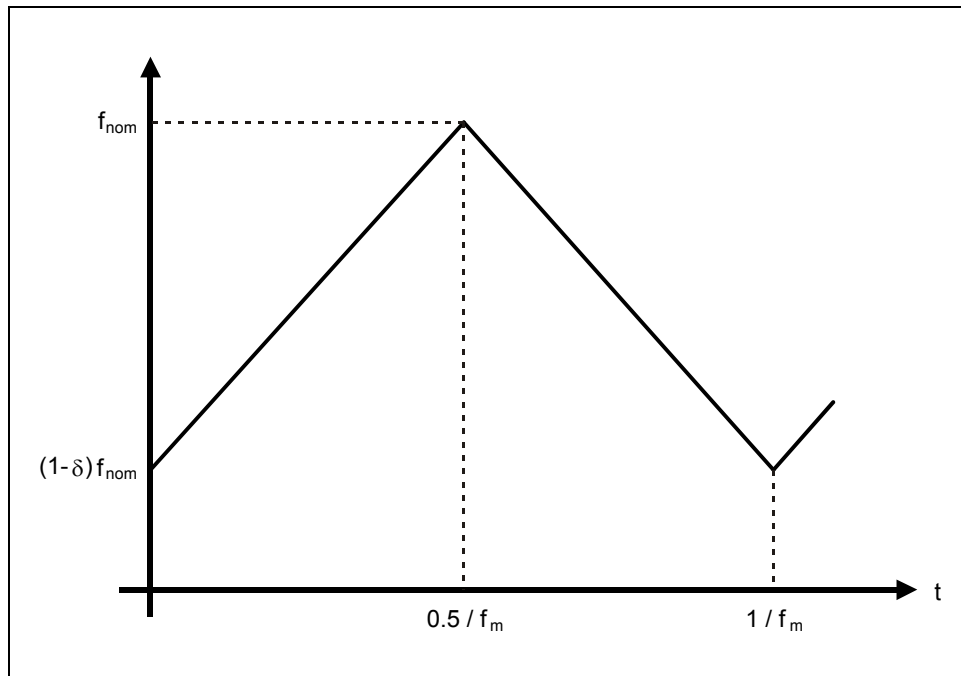
and:

$$f = (1 + \delta) f_{nom} - 2 \times f_m \times \delta \times f_{nom} \times t$$

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,  $\delta$  is the modulation amount, and  $t$  is time.



**Figure 195 – SSC profile example, triangular**

As an example, for triangular modulation, the absolute spread amount at the fundamental frequency is shown in Figure 196, as the width of its spectral distribution.



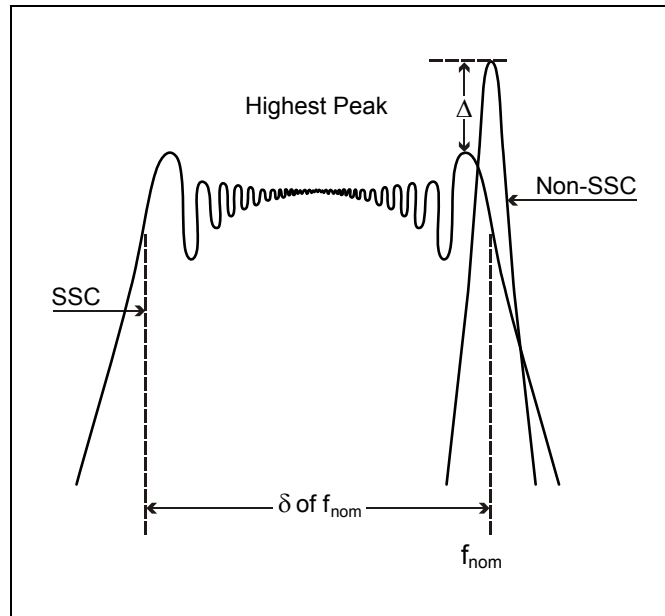


Figure 196 – Spectral fundamental frequency comparison

### 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 ( $RJ_{\sigma}$ ) 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 unit under test 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 unit under test 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 (unit under test). 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 unit under test.

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 unit under test.

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 unit under test that goes beyond specification compliance issues.

Both methods produce measurements of electrical performance, however, the parametric method shall be used for validation of the unit under test 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 unit under test. 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  $\pm$  5 ohm loads. The cables shall be 50 ohm  $\pm$  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 ps 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 197.

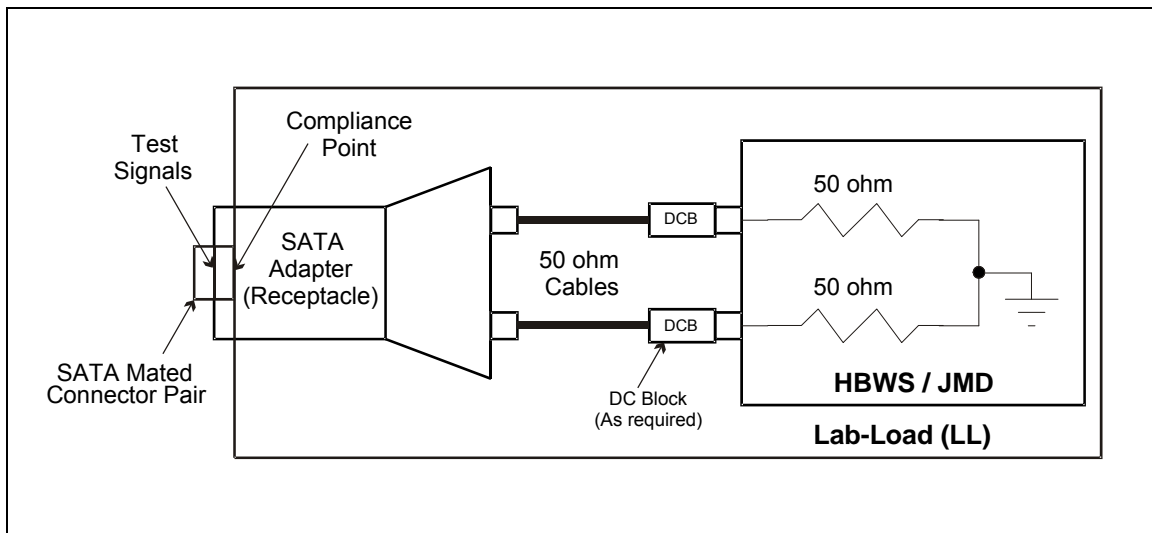
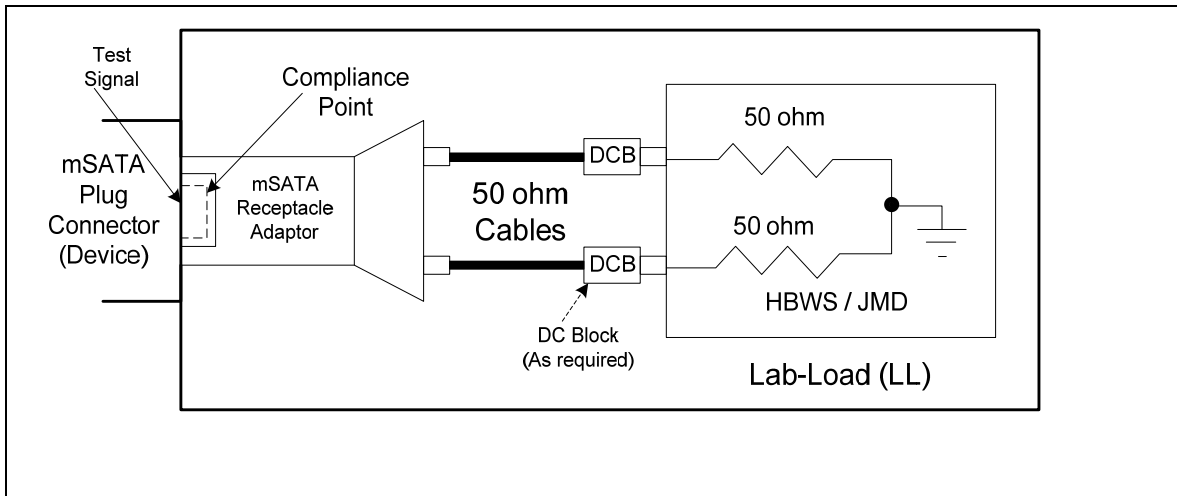


Figure 197 – Lab-Load (LL)

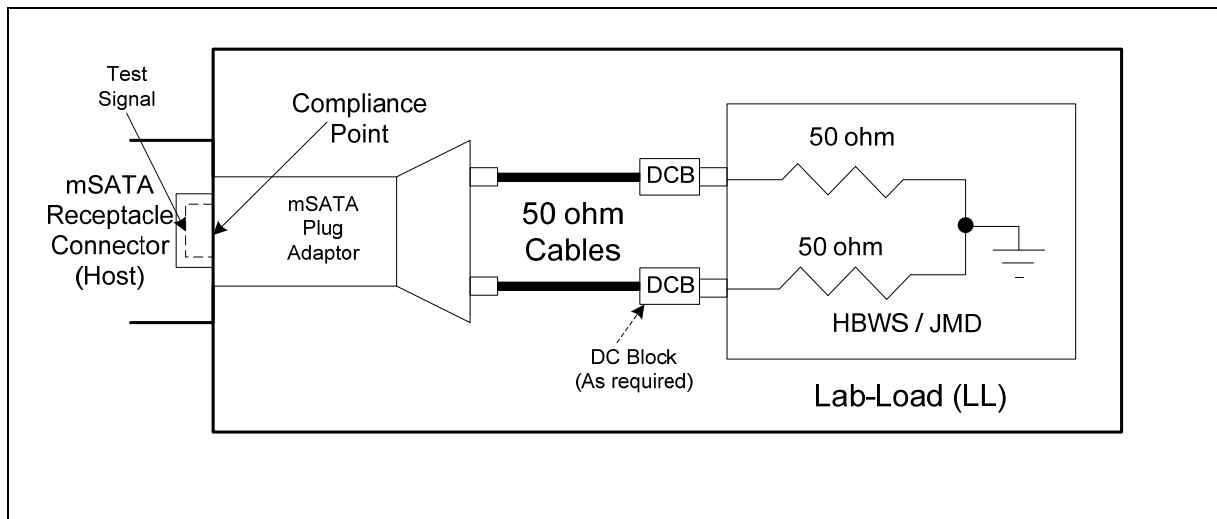
The electrical characteristics of the lab-load shall be greater than the required performance of the parameter being measured such that the effects of the lab-load on the parameter under test may be successfully compensated for, or de-embedded, in the measured data.

#### 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 198 and Figure 199, respectively.



**Figure 198 – Lab-Load (LL) for mSATA device**

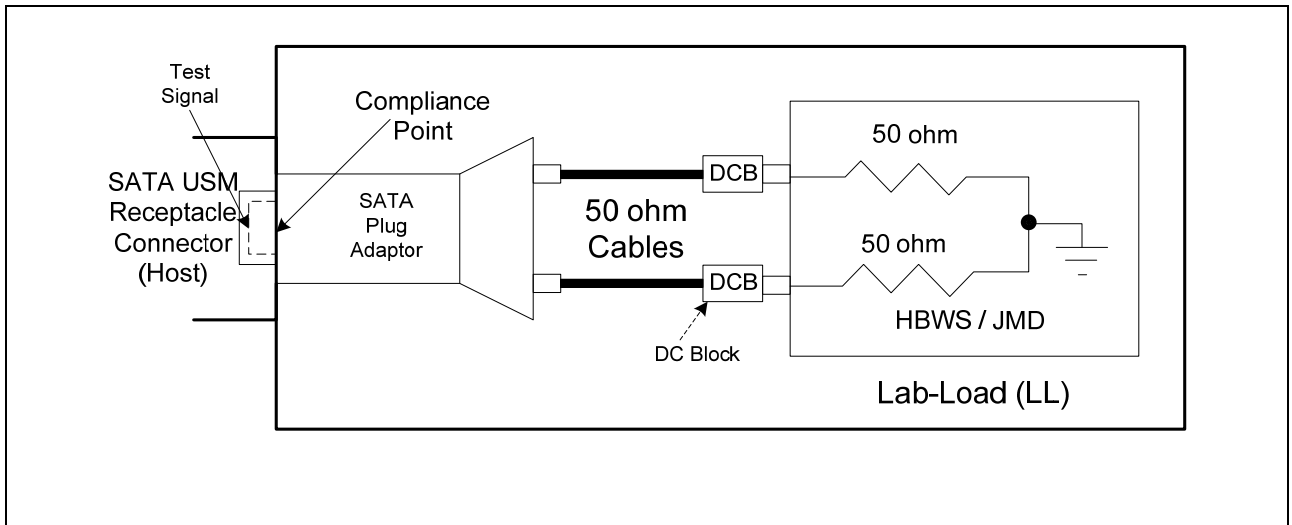


**Figure 199 – Lab-Load (LL) for mSATA host**

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 200. The adaptor may require additional mechanical clearance to mate to the SATA USM host connector. See SFF publication INF-8280 for host connector details.

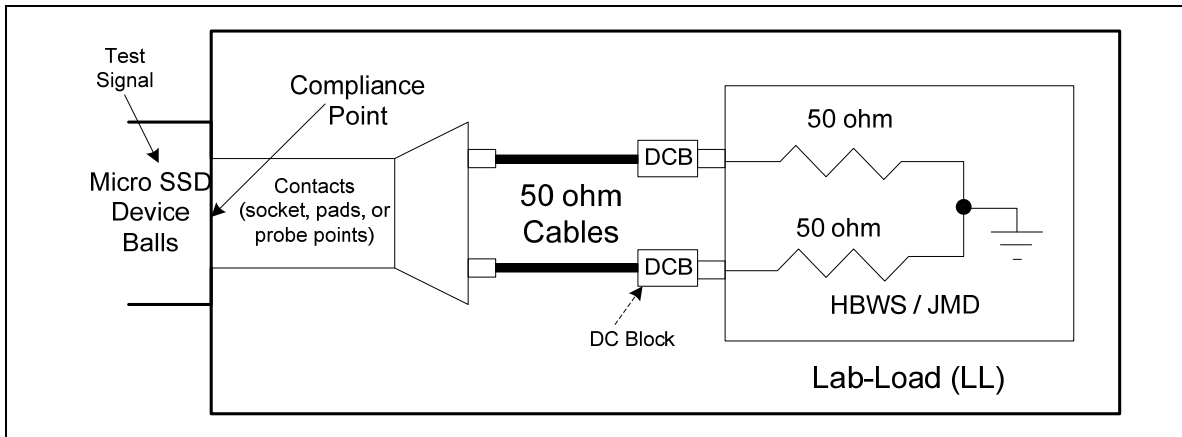


**Figure 200 – Lab-Load (LL) for SATA USM host**

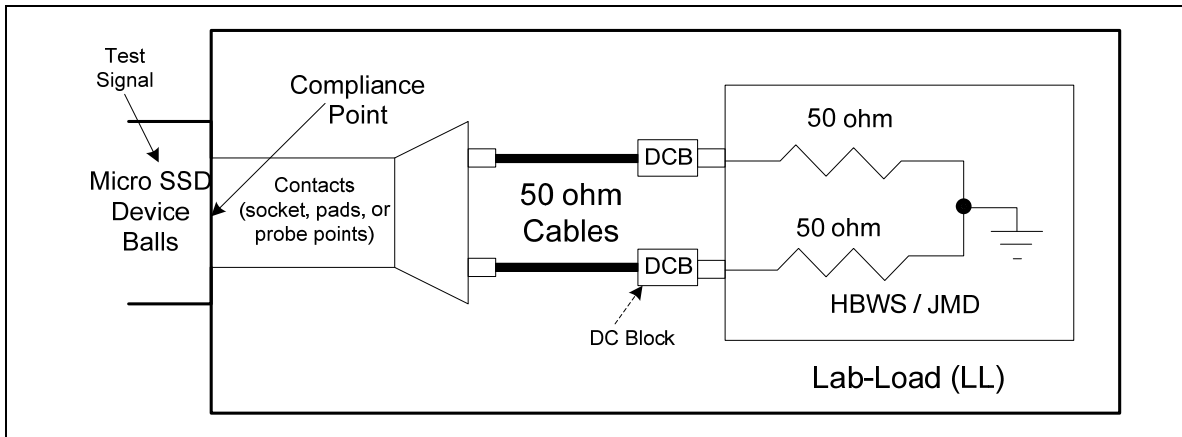
The electrical characteristics of the lab-load shall be greater than the required performance of the parameter being measured such that the lab-load effects on the parameter under test may be successfully compensated for, or de-embedded, in the measured data.

**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 201 and the host shall be mated to an adaptor as shown in Figure 202, respectively. It is not possible to test the SATA MicroSSD host if the SATA MicroSSD Device is soldered in place.



**Figure 201 – Lab-Load (LL) for SATA MicroSSD device**



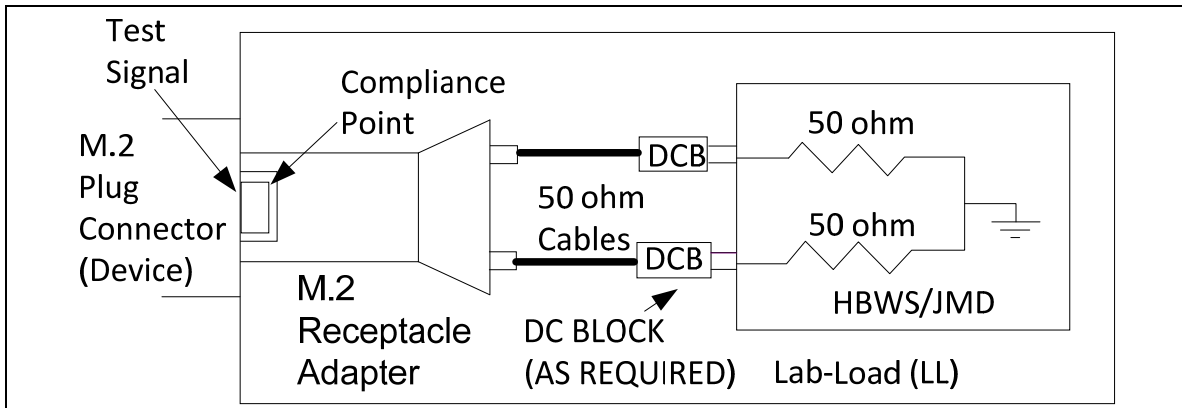
**Figure 202 – Lab-Load (LL) for SATA MicroSSD host**

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.

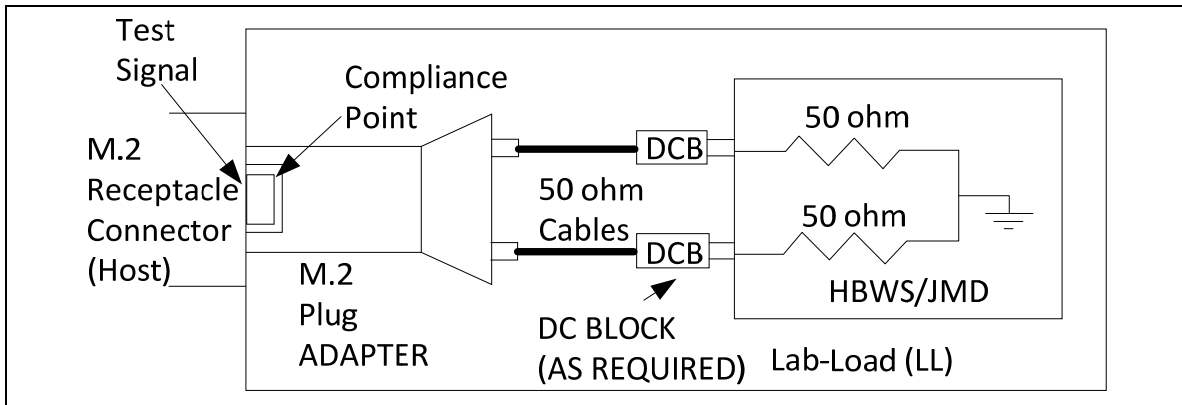
In a test environment, a test fixture and methodology may be used that makes it possible to deembed the signal attenuation effects of the test fixture, and thus measure compliance at the solder balls on the Device and/or the socket/pads on the Host.

**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 203 and Figure 204.



**Figure 203 – Lab-Load (LL) for M.2 device**



**Figure 204 – Lab-Load (LL) for M.2 host**

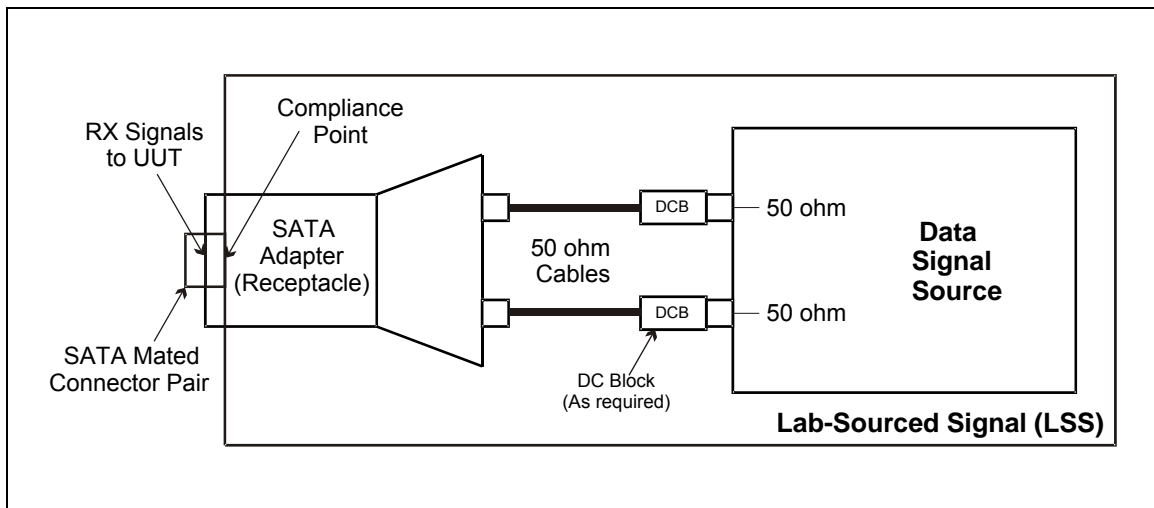
The electrical characteristics of the lab-load shall be greater than the required performance of the parameter being measured such that the lab-load effects of the on the parameter under test may be successfully compensated for, or de-embedded, in the measured data.

### 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 unit under test. 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.



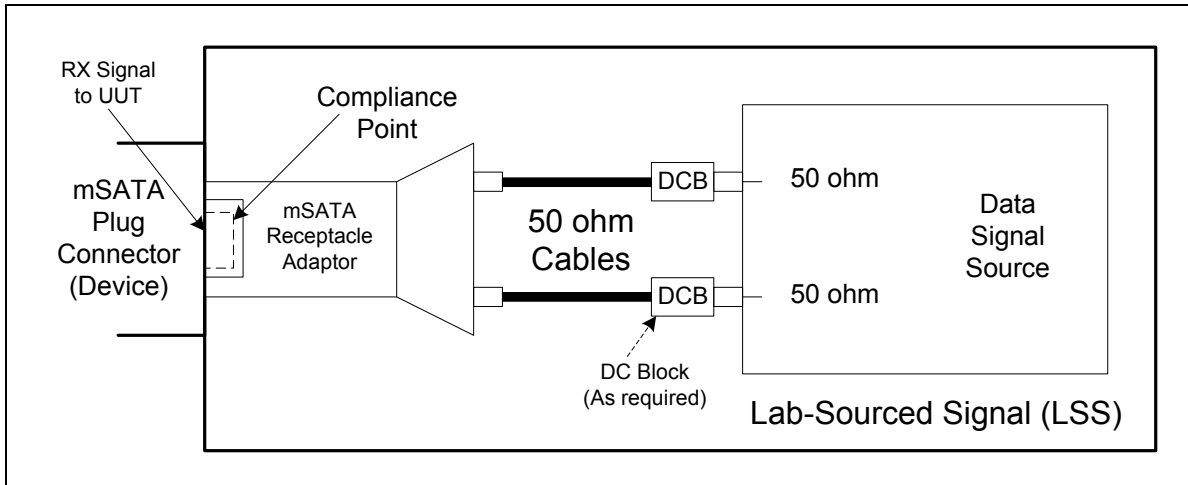
**Figure 205 – Lab-Sourced Signal (LSS)**

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

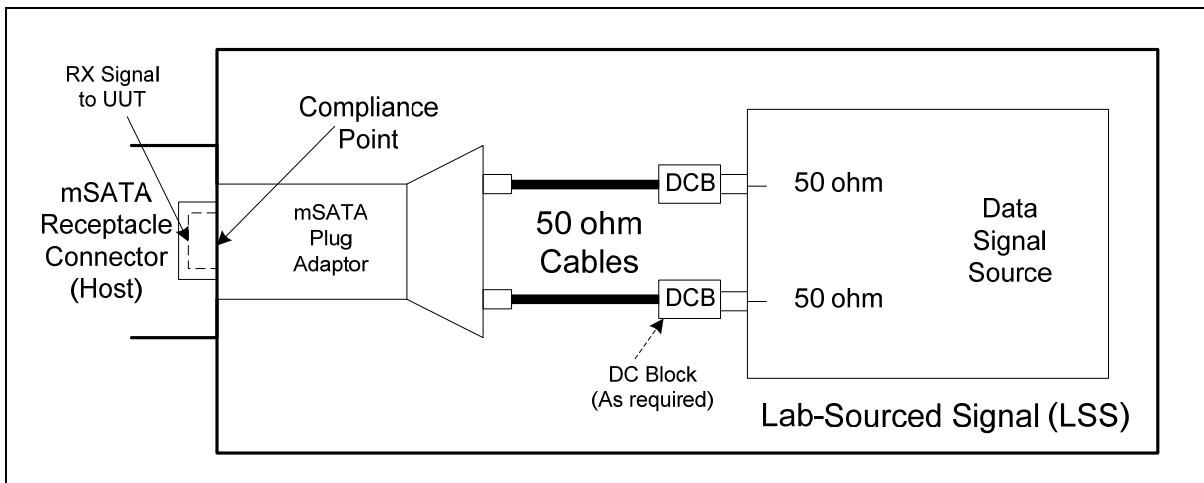
have an individual return loss greater than 20 dB over a bandwidth of 100 MHz to 5.0 GHz, and greater than 10 dB from 5 GHz to 8 GHz. During calibration, the characteristics of the lab-sourced signal shall comply with the specifications of Table 57. This signal is then applied to the Receiver Under Test 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 206 and Figure 207.



**Figure 206 – Lab-Sourced Signal (LSS) for mSATA device**



**Figure 207 – 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 208. The adaptor may require additional mechanical clearance to mate to the SATA USM host connector. See SFF publication INF-8280 for host connector details.

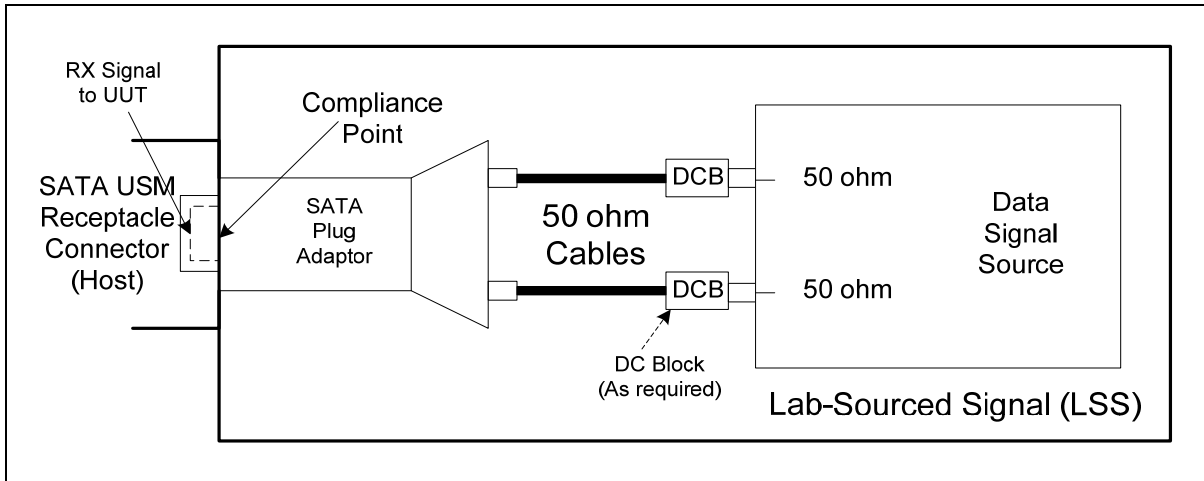


Figure 208 – Lab-Sourced Signal (LSS) for SATA USM host

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 209 and Figure 210, respectively. It is not possible to test the MicroSSD host if the SATA MicroSSD Device is soldered in place.

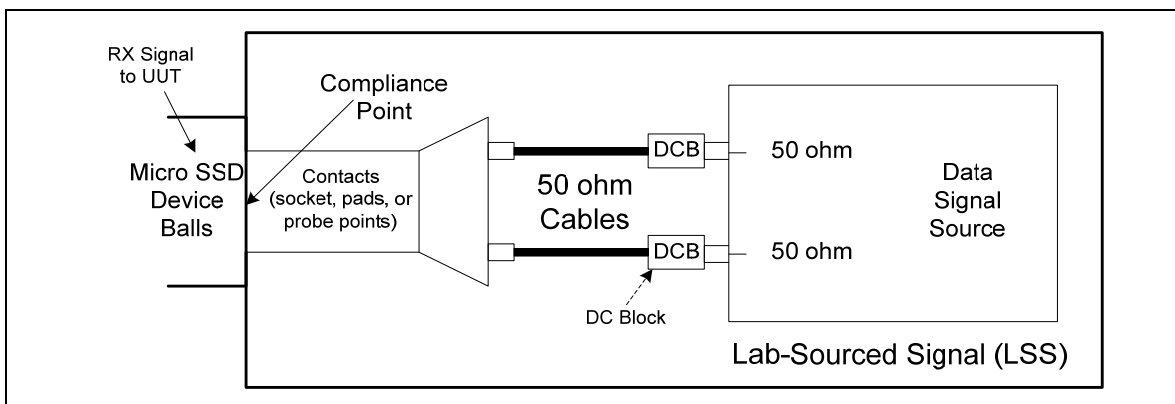
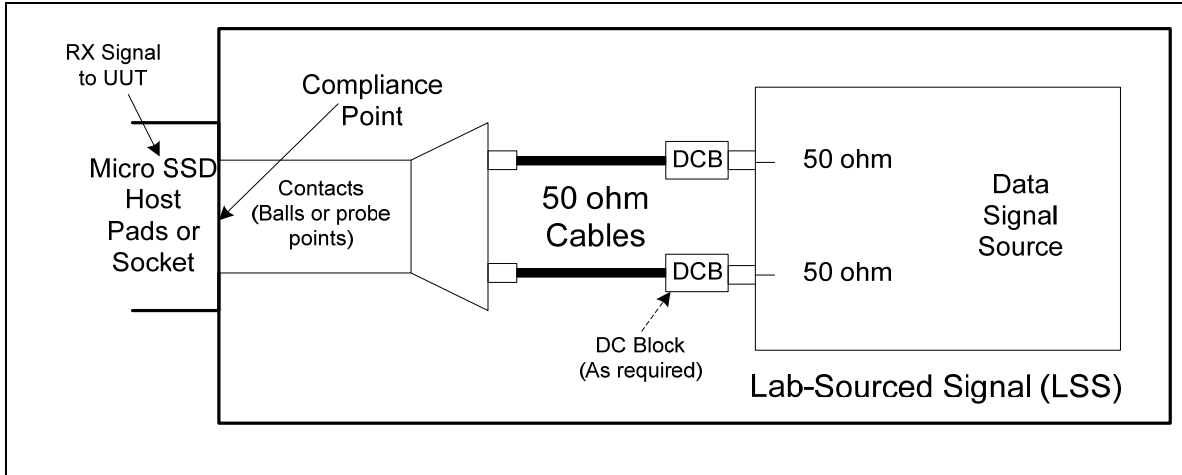


Figure 209 – Lab-Sourced Signal (LSS) for SATA MicroSSD device



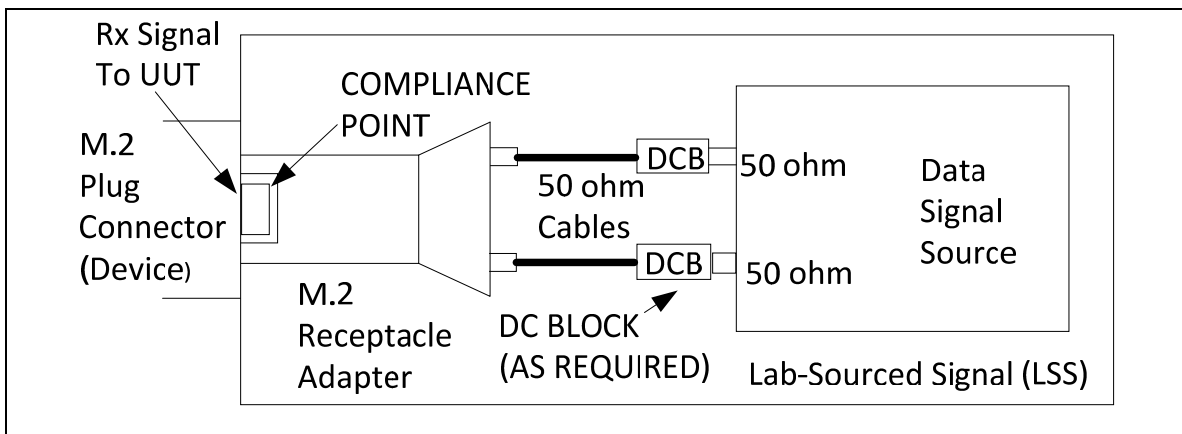
**Figure 210 – Lab-Sourced Signal (LSS) for SATA MicroSSD host**

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 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 test 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 211 and Figure 212.



**Figure 211 – Lab-Sourced Signal (LSS) for M.2 device**

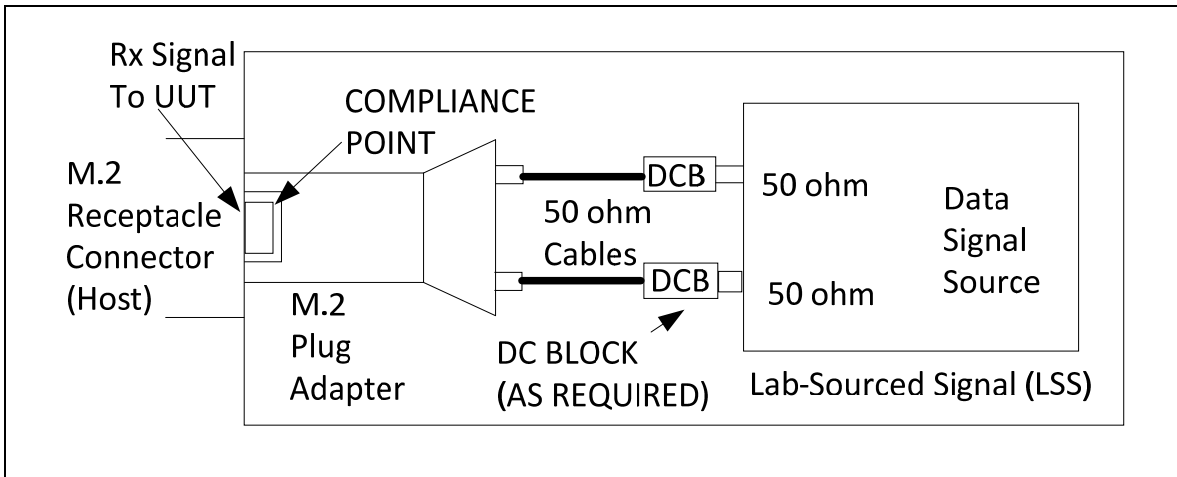


Figure 212 – Lab-Sourced Signal (LSS) for M.2 host

### 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 (Cyclic Redundancy Check) error detection mechanism, and respective reporting to the higher layer levels. Since all frames include a header and CRC field, 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 29 -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  $8.200 \times 10^{-8}$  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 30 - Note that the cited patterns should 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 should 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
$1.22 \times 10^7$	63.21 %	26.42 %	8.03 %	1.90 %	0.37 %	0.06 %	0.01 %	<0.01 %	<0.01 %	<0.01 %	<0.01 %
$1.22 \times 10^8$	>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  $1.22 \times 10^8$  frames are passed, the maximum number of frame errors to measure a Frame Error Rate of  $8.200 \times 10^{-8}$  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 31 - Burst Error Rate measurements shall not be 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 0 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 should 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
$1.00 \times 10^{12}$	63.21 %	26.42 %	8.03 %	1.90 %	0.37 %	0.06 %	0.01 %	<0.01 %	<0.01 %	<0.01 %	<0.01 %
$1.00 \times 10^{13}$	>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 213 shows the connections.

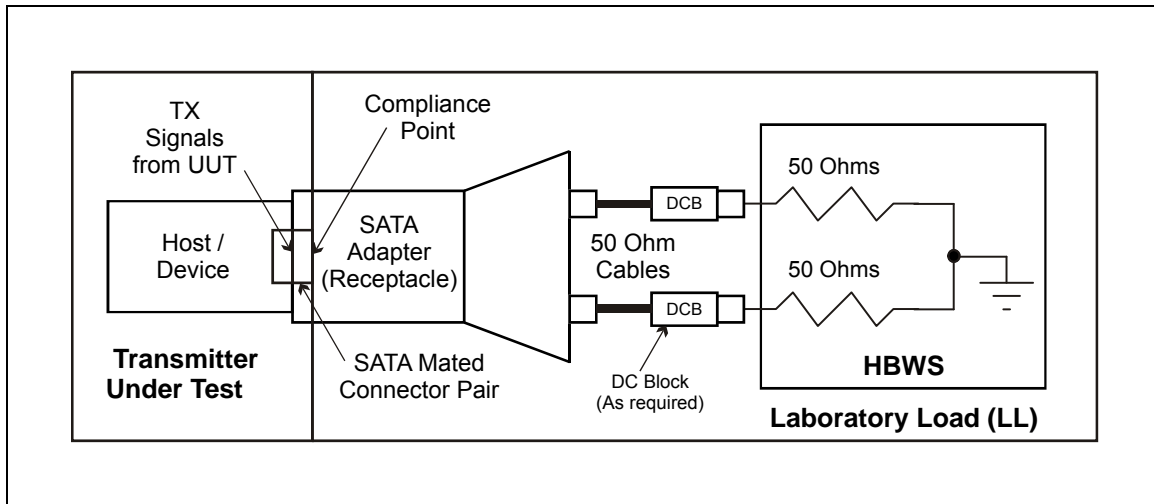


Figure 213 – Differential Voltage amplitude measurement

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 214 illustrates an example of a display on an equivalent time scope.

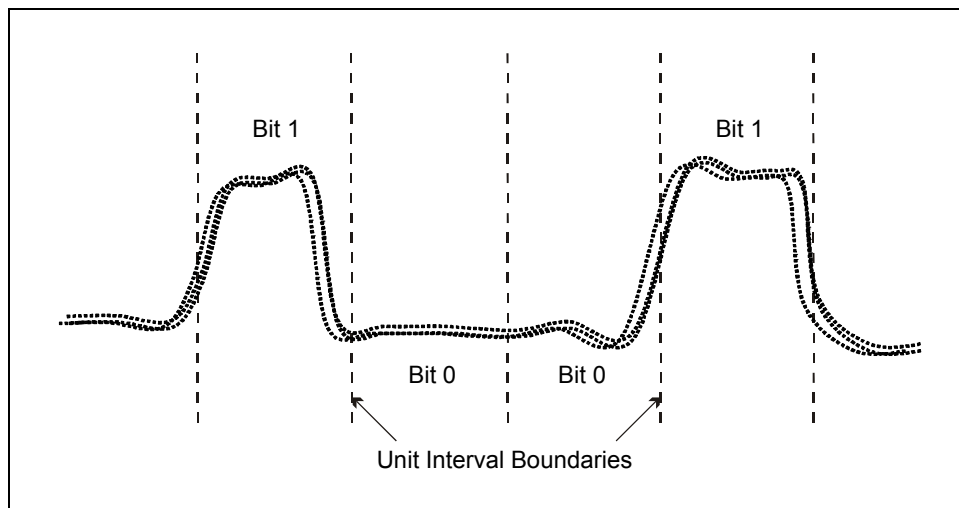


Figure 214 – Differential Voltage amplitude measurement pattern example

## 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 unit interval (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:

$\bar{x}$  = the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode

s = the standard deviation of the voltage samples in the histogram that may also be read from the HBWS

n = 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 unit interval (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:

$\bar{x}$  = the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode

s = the standard deviation of the voltage samples in the histogram that may also be read from the HBWS

n = the number of samples that contribute to the histogram – this may also be read from the HBWS

Compute the following value:

$$LH = \left[ \bar{x} + \frac{1.96 \times s}{\sqrt{n}} \right]$$

**Common Step 3**, transmitting a MFTP pattern, for a unit interval (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:

$$1\,537 \left( s / \bar{x} \right)^2 \leq n$$

where:

$\bar{x}$  = the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode

s = the standard deviation of the voltage samples in the histogram that may also be read from the HBWS

n = the number of samples that contribute to the histogram – this may also be read from the HBWS

Compute the following value:

$$UM = \left[ \bar{x} - \frac{1.96 \times s}{\sqrt{n}} \right]$$

**Common Step 4**, transmitting a MFTP pattern, for a unit interval (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:

$\bar{x}$  = the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode

s = the standard deviation of the voltage samples in the histogram that may also be read from the HBWS

n = 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 unit interval (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:

$\bar{x}$  = the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode

s = the standard deviation of the voltage samples in the histogram that may also be read from the HBWS

n = the number of samples that contribute to the histogram – this may also be read from the HBWS

Compute the following value:

$$A = \left[ \bar{x} - \frac{1.96 \times s}{\sqrt{n}} \right]$$

**Option 1 Step 7**, transmitting a LBP pattern, for a unit interval (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:

$$1.537 \left( \frac{s}{\bar{x}} \right)^2 \leq n$$

where:

$\bar{x}$  = the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode

s = the standard deviation of the voltage samples in the histogram that may also be read from the HBWS

n = the number of samples that contribute to the histogram – this may also be read from the HBWS

Compute the following value:

$$B = \left[ \bar{x} + \frac{1.96 \times s}{\sqrt{n}} \right]$$

**Option 1 Step 8**, from A and B obtained in steps 1 and 2, compute:

$$V_{\text{TestLBP}} = A - B$$

Then take the minimum of  $V_{\text{TestLBP}}$  and the previously computed DHM (from Common Step 5), that is,

$$V_{\text{Test}} = \min (V_{\text{TestLBP}}, \text{DHM})$$

The test for minimum amplitude is passed if:

$$V_{\text{Test}} > V_{\text{diffTx}}(\text{Min})$$

See Table 54 and Table 55 according to 7.4.2 for  $V_{\text{diffTx}}(\text{Min})$ . 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:

$$1.537 \left( \frac{s}{\bar{x}} \right)^2 \leq n$$

where:

$\bar{x}$  = the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode

$s$  = the standard deviation of the voltage samples in the histogram that may also be read from the HBWS

$n$  = the number of samples that contribute to the histogram – this may also be read from the HBWS

Compute the following value:

$$A = \left[ \bar{x} - \frac{1.96 \times s}{\sqrt{n}} \right]$$

**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:

$\bar{x}$  = the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode

s = the standard deviation of the voltage samples in the histogram that may also be read from the HBWS

n = the number of samples that contribute to the histogram – this may also be read from the HBWS

Compute the following value:

$$B = \left[ \bar{x} + \frac{1.96 \times s}{\sqrt{n}} \right]$$

**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:

$\bar{x}$  = the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode

s = the standard deviation of the voltage samples in the histogram that may also be read from the HBWS

n = the number of samples that contribute to the histogram – this may also be read from the HBWS

Call the mean,

$$\bar{x} = C$$

**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:

$$1.537 \left( \frac{s}{\bar{x}} \right)^2 \leq n$$

where:

$\bar{x}$  = the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode

$s$  = the standard deviation of the voltage samples in the histogram that may also be read from the HBWS

$n$  = the number of samples that contribute to the histogram – this may also be read from the HBWS

Call the mean,

$$\bar{x} = D$$

**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:

$$1.537 \left( \frac{s}{\bar{x}} \right)^2 \leq n$$

where:

$\bar{x}$  = the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode

$s$  = the standard deviation of the voltage samples in the histogram that may also be read from the HBWS

$n$  = the number of samples that contribute to the histogram – this may also be read from the HBWS

Call the mean,

$$\bar{x} = E$$

**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:

$$1.537 \left( \frac{s}{\bar{x}} \right)^2 \leq n$$

where:

$\bar{x}$  = the mean of the voltage samples in the histogram that may be read from the HBWS in histogram measurement mode

s = the standard deviation of the voltage samples in the histogram that may also be read from the HBWS

n = the number of samples that contribute to the histogram – this may also be read from the HBWS

Call the mean,

$$\bar{x} = F$$

**Option 2 Step 12**, from A and B obtained in steps 1 and 2, compute:

$$V_{\text{TestAPP}} = (A + C + F) - (B + D + E)$$

Then take the minimum of  $V_{\text{TestAPP}}$  and the previously computed DHM, that is,

$$V_{\text{Test}} = \min(V_{\text{TestAPP}}, \text{DHM})$$

The test for minimum amplitude is passed if:

$$V_{\text{Test}} > V_{\text{diffTx}}(\text{Min})$$

See Table 54 and Table 55, according to 7.4.2 for  $V_{\text{diffTx}}(\text{Min})$ . 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 215 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 through 6 above) are to be made.

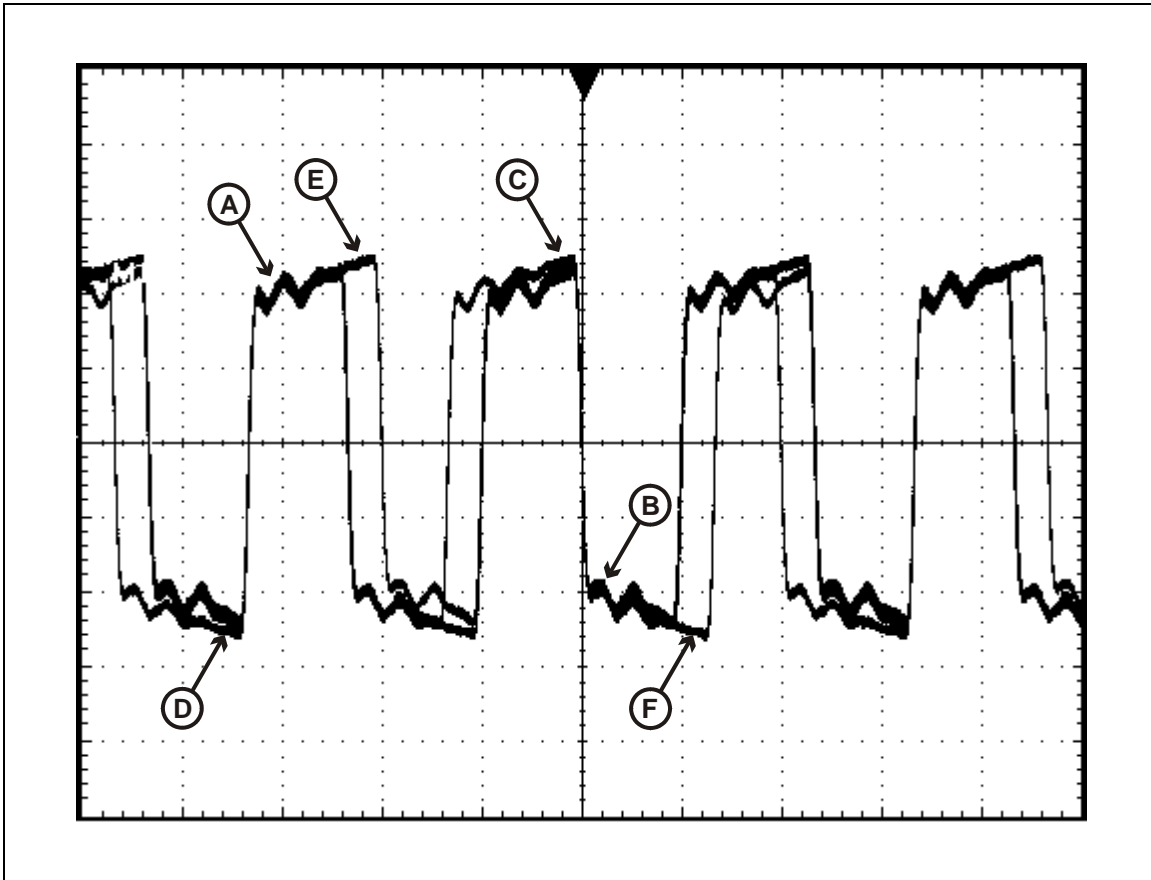


Figure 215 – LFTP Pattern on High BW Scope (HBWS)

### 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  $V_{diffTx}(\text{Max})$ .

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:

$$VL = -\frac{1}{2}V_{DIFFIX}(\max)$$

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 32 - 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 33 -\_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 217 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 217 and Figure 218 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 200 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  $5 \times 10^6$  unit intervals 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  $5 \times 10^6$  unit intervals. 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  $5 \times 10^6$  unit intervals 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 % threshold from the reference levels. 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.

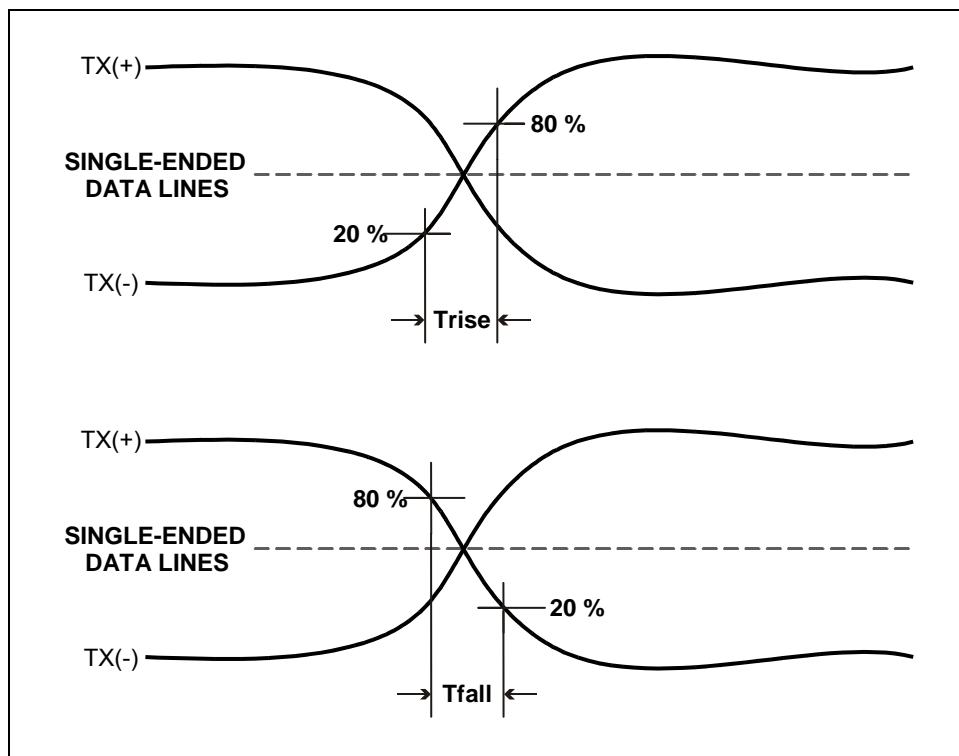


Figure 216 – Single ended rise and fall time

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 % threshold, 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 234), 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 217. 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 unit under test (UUT) at the mated connector into a lab-load (for Gen1i, Gen1m, Gen1u, Gen2i, Gen2m, Gen2u, Gen3i, and Gen3u), or from the unit under test 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 217. The HBWS is measuring a DC voltage and the DC blocks (DCB) shall not be present.

Figure 217 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).

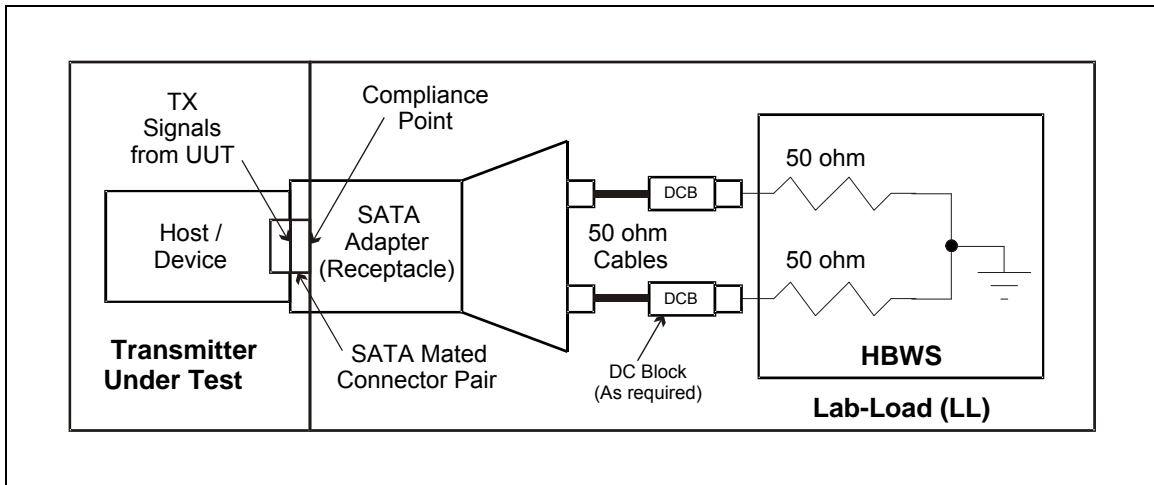


Figure 217 – Transmit amplitude test with Lab-Load (LL)

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 217 and Figure 218 show test setups for measuring transmitter amplitude. See 7.6.7.2 for suggestions on compensating for losses in the test connections.

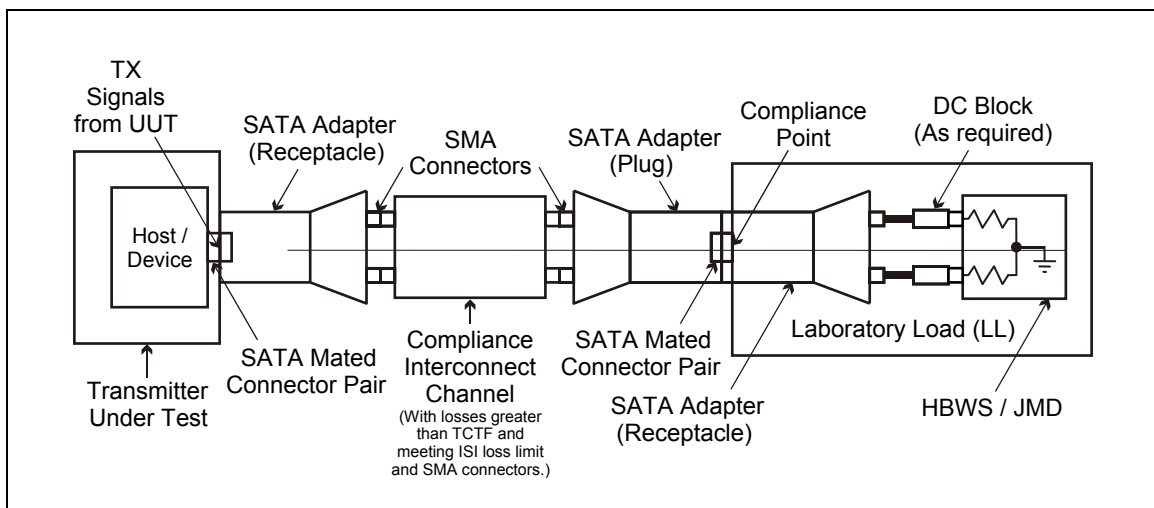
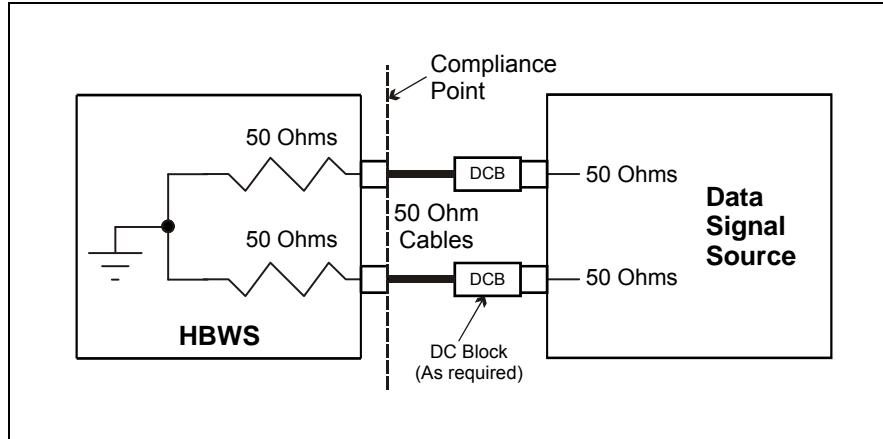


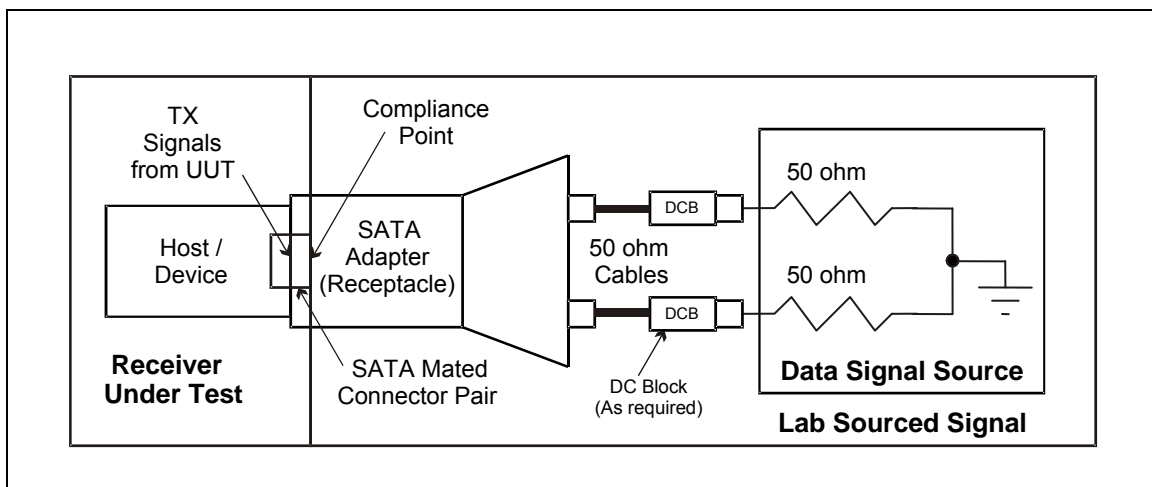
Figure 218 – Transmit amplitude test with CIC

### 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.



**Figure 219 – Receiver amplitude test, setting levels**



**Figure 220 – Receiver amplitude test**

Figure 219 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.

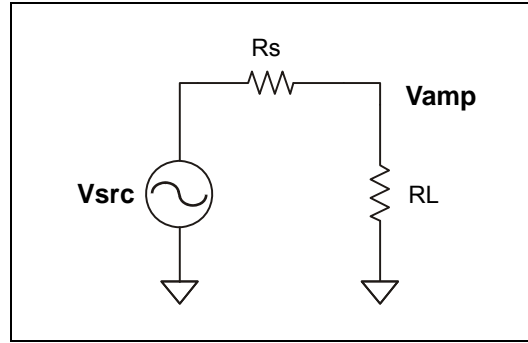


Figure 221 – Voltage at receiver input

$$V_{AMP} = \frac{V_{SRC}}{\left(1 + \frac{R_S}{R_L}\right)} = \frac{V_A \left(1 + \frac{R_S}{R_{LL}}\right)}{\left(1 + \frac{R_S}{R_L}\right)}$$

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.

$$V_{AMP} = \frac{0.7 \left( 1 + \frac{100}{100} \right)}{\left( 1 + \frac{100}{128.8} \right)} = 0.788 \text{ V}$$

The lowest amplitude at the receiver occurs if the receiver input resistance is lowest.

$$V_{AMP} = \frac{0.4 \left( 1 + \frac{100}{100} \right)}{\left( 1 + \frac{100}{77.64} \right)} = 0.349 \text{ V}$$

### 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 222 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.



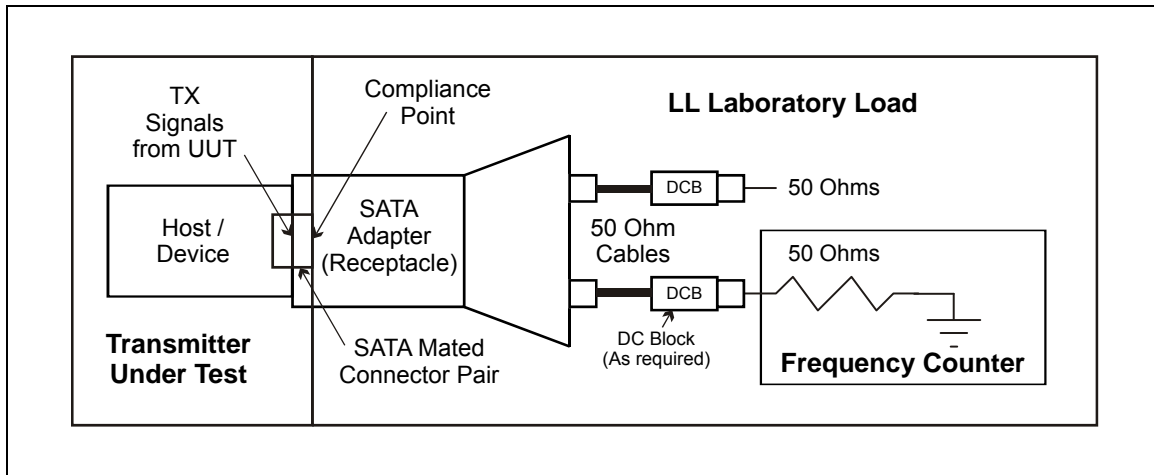


Figure 222 – Tx long term frequency measurement

## 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.

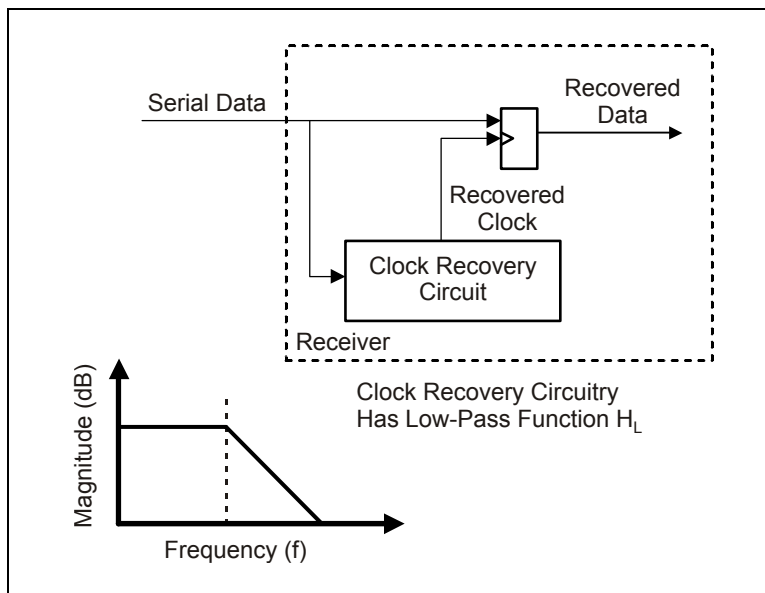
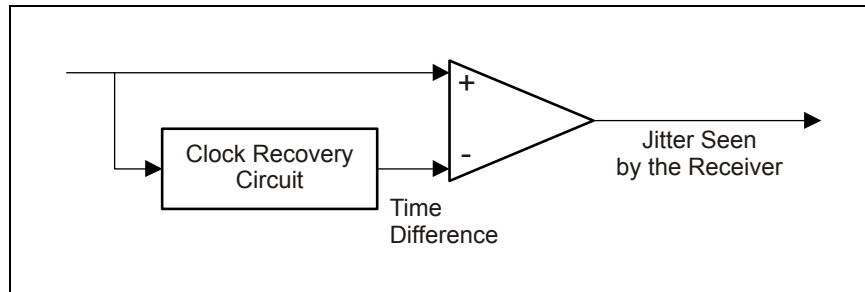
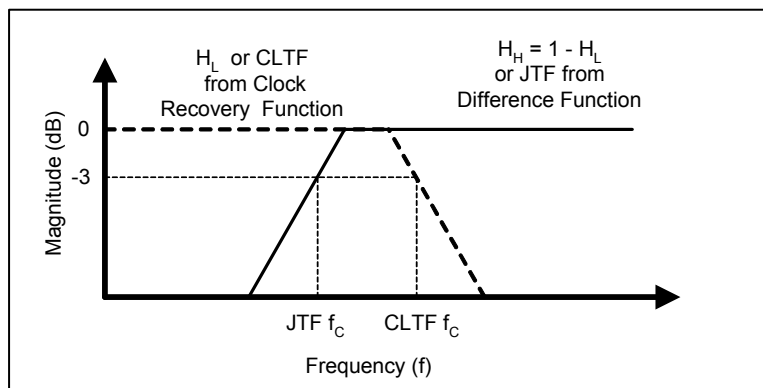


Figure 223 – Receiver model for jitter

The jitter measurement methodology is defined as a clock to data jitter measurement. Figure 223 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 224. The resulting jitter seen by the receiver has a high pass function  $H_H$  shown in Figure 225. This high pass function is the JTF (Jitter Transfer Function) 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 224 – Jitter at receiver**



**Figure 225 – Jitter at receiver, high pass function**

This JTF ( $H_H$  in Figure 225) 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 226 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 rationale 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.

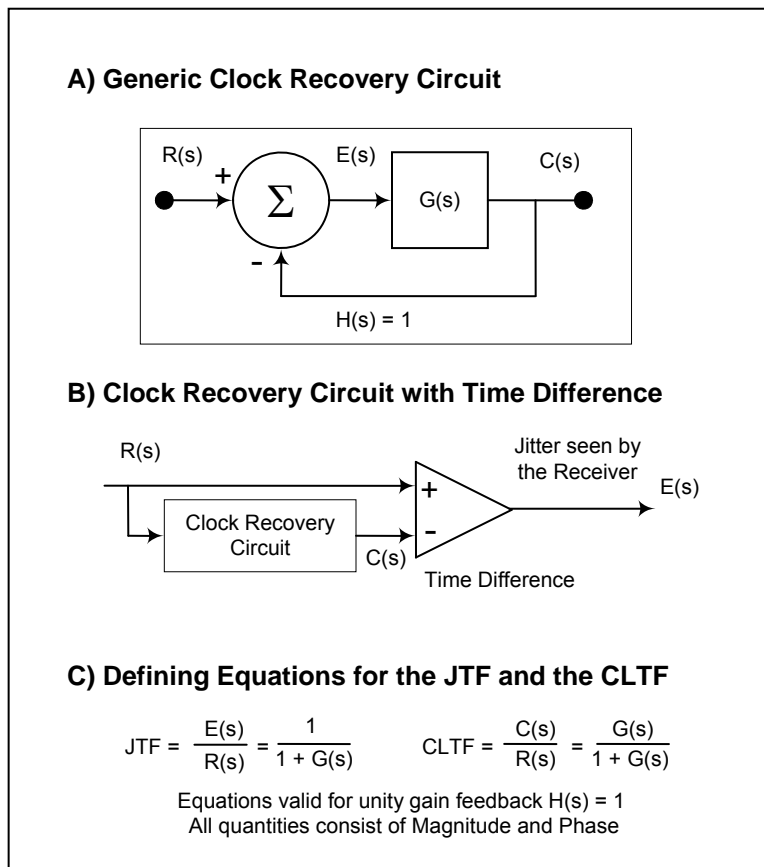


Figure 226 – JTF and CLTF definition

**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.<sup>1</sup> 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

---

<sup>1</sup> "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.

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 unit under test (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 227.

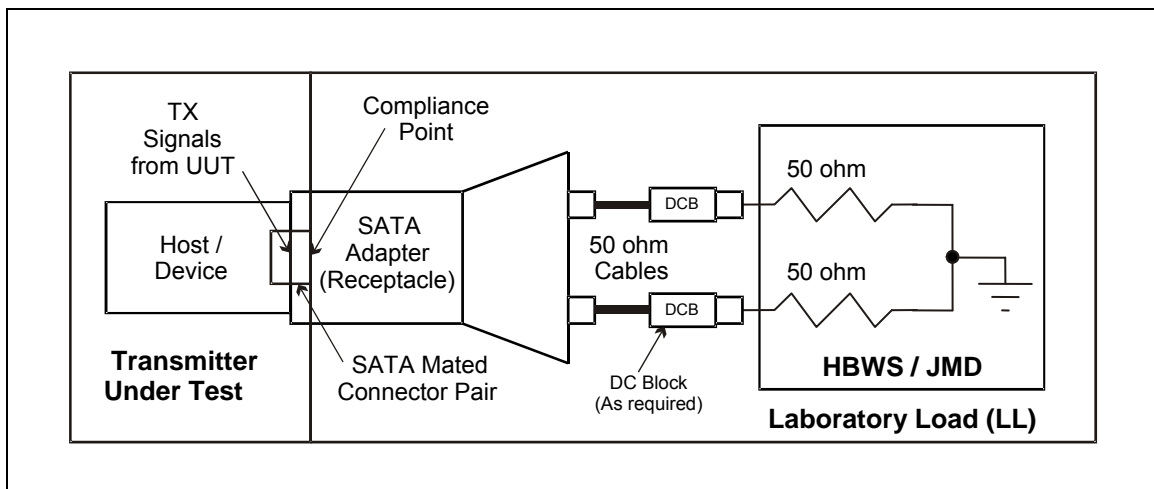


Figure 227 – Transmitter jitter test (Gen1i, Gen2i)

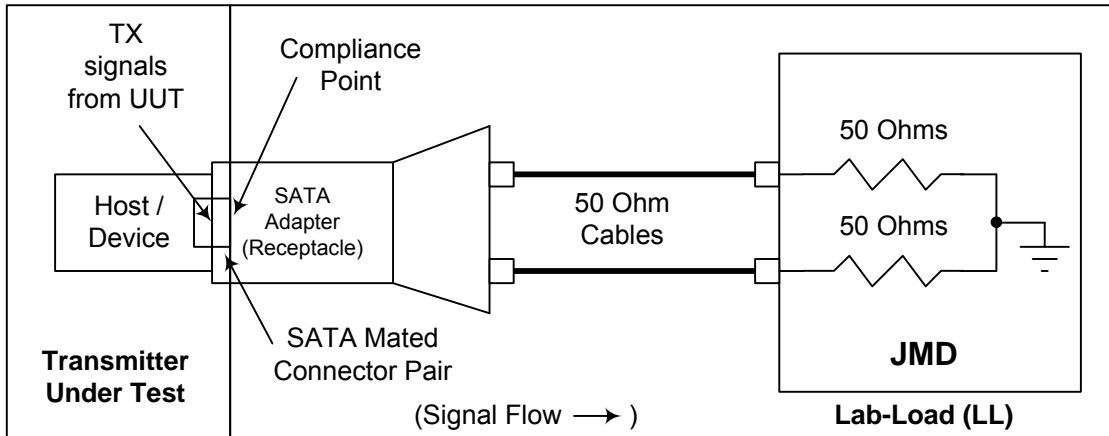
Transmitter jitter is measured into the lab-load, or in conjunction with the Compliance Interconnect Channel; both have very good impedance matching. The jitter in an actual system is higher since load and interconnect mismatch results in reflections and additional data dependent jitter. It is generally 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 unit under test (UUT) at the mated connector into a lab-load, and, for Table 54, from the unit under test 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.

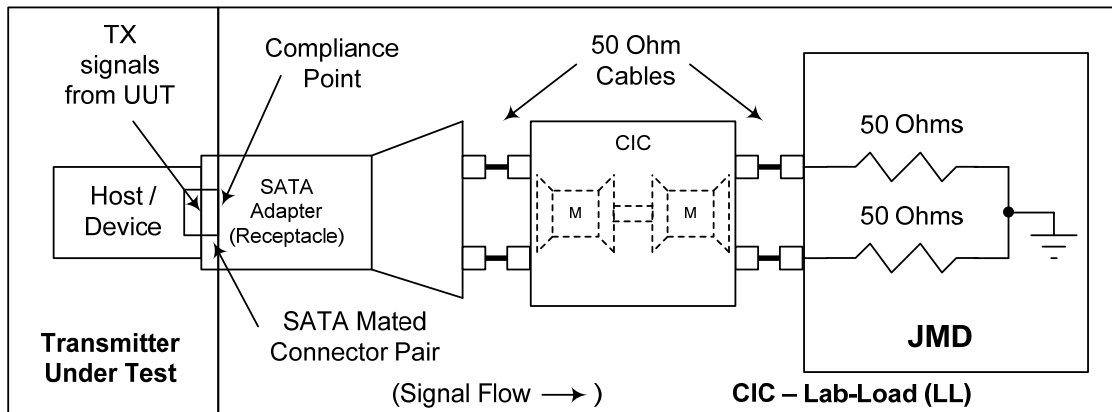
The Total Jitter parameters are measured with each of the specified patterns as defined in 7.4.5.2 and 7.4.5.4.5. The measurement of jitter as defined in 7.6.10.

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 228.



**Figure 228 – 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 229. This measurement does not apply to Table 55 (i.e., Gen3u).



**Figure 229 – Transmitter jitter test through the CIC (Gen3i)**

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 ps, this equates to a  $TJ(10^{-12})$  of 86.667 ps and a RJ 1 sigma value of 2.143 ps.

This calculation is performed using the dual Dirac equation  $TJ(BER) = (2Q_{BER} \times \sigma) + DJ_{\delta\delta}$ , where  $\sigma$  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.

## HIGH SPEED SERIALIZED AT ATTACHMENT Serial ATA International Organization

At a BER level of  $10^{-12}$ ,  $DJ_{\delta\delta} = TJ(10^{-12}) - (14.069 \times \sigma) = 86.667 \text{ ps} - (14.069 \times 2.143 \text{ ps}) = 56.517 \text{ ps}$ .

At a BER level of  $10^{-6}$ ,  $TJ(10^{-6}) = (9.507 \times \sigma) + DJ_{\delta\delta} = (9.507 \times 2.143 \text{ ps}) + 56.517 \text{ ps} = 76.891 \text{ ps}$  pp 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 ps, then the allowable  $DJ_{\delta\delta}$  is able to be 19 % larger than the 56.5 ps derived in the calculation above. If the  $DJ_{\delta\delta}$  was 20.0 ps, then the RJ 1 sigma value is able to be 4.74 ps. 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 Compliance Interconnect Channel. 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 231). 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 230 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 231 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.

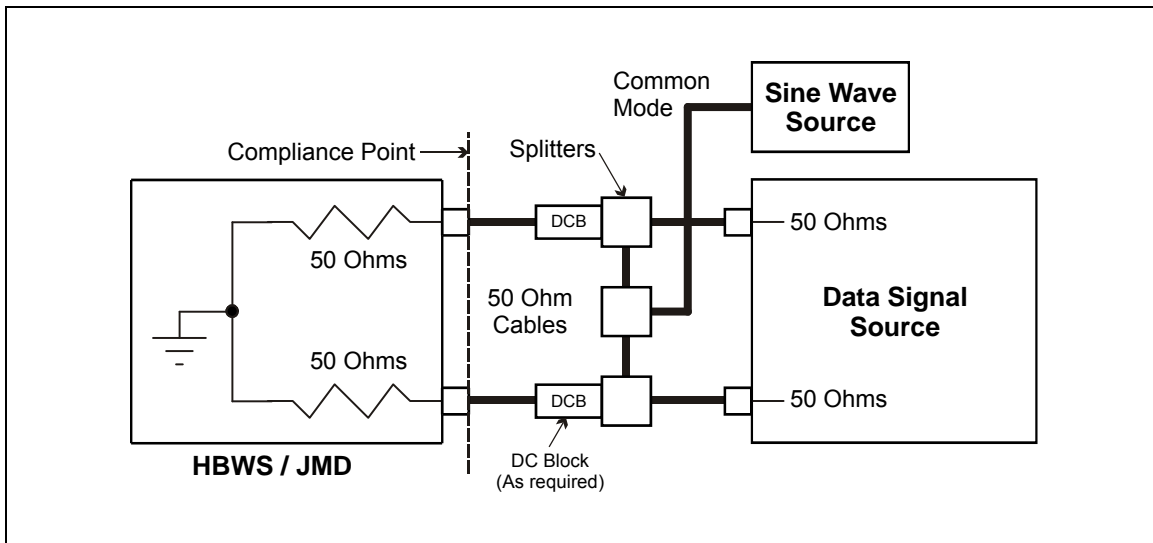


Figure 230 – Receiver jitter and CM tolerance test – setting levels

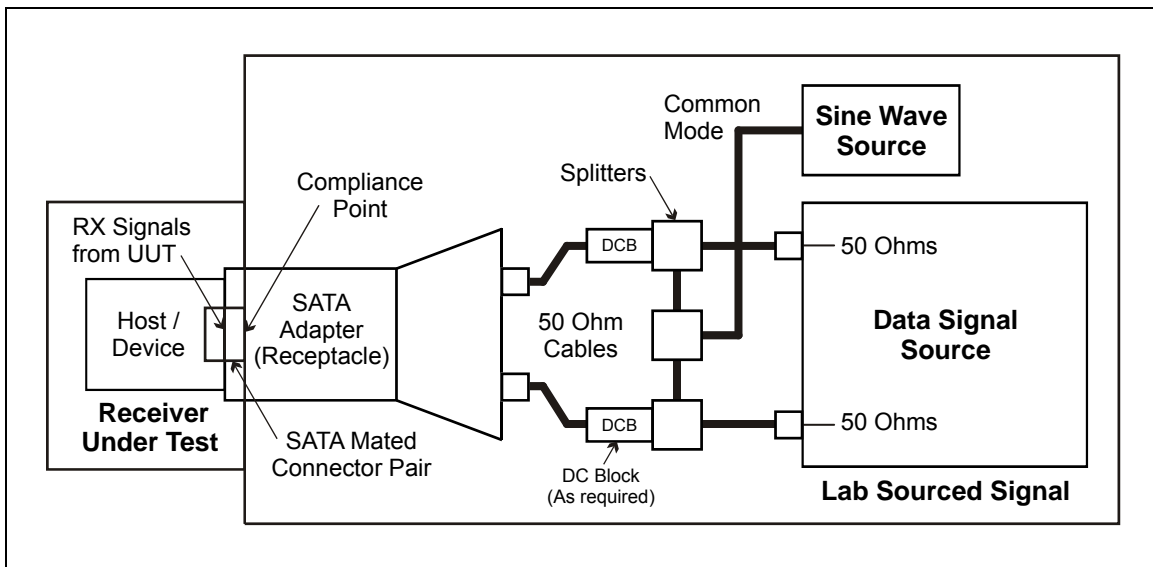


Figure 231 – Receiver jitter and CM tolerance test



#### **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 232 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 233 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 233 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 234 shows the calibrated lab-sourced signal applied to the Gen3i Receiver Under Test. Figure 231 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.

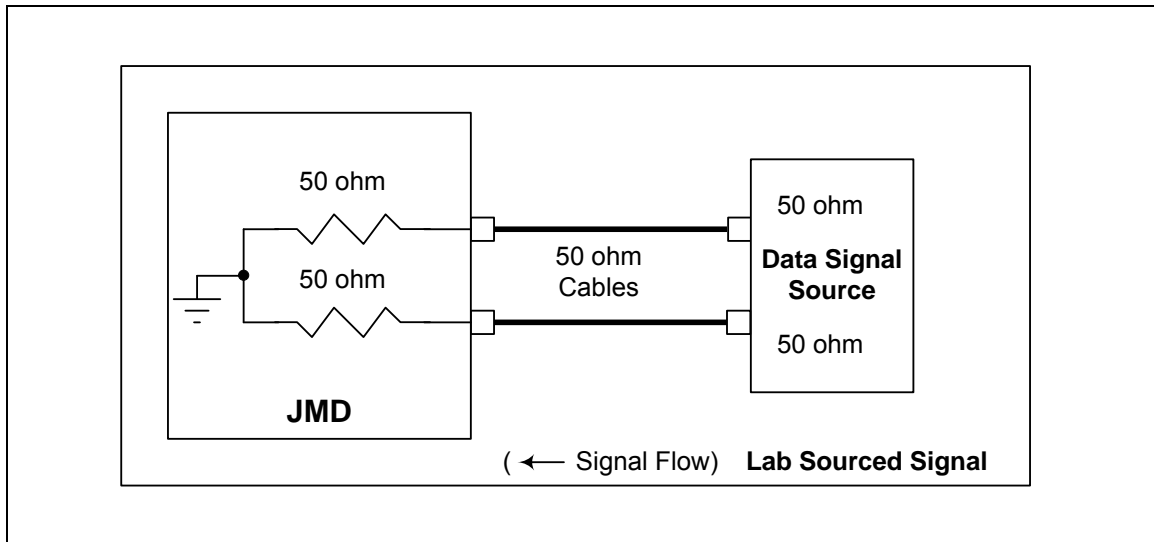


Figure 232 – Receiver jitter and CM tolerance test – setting RJ level (Gen3i)

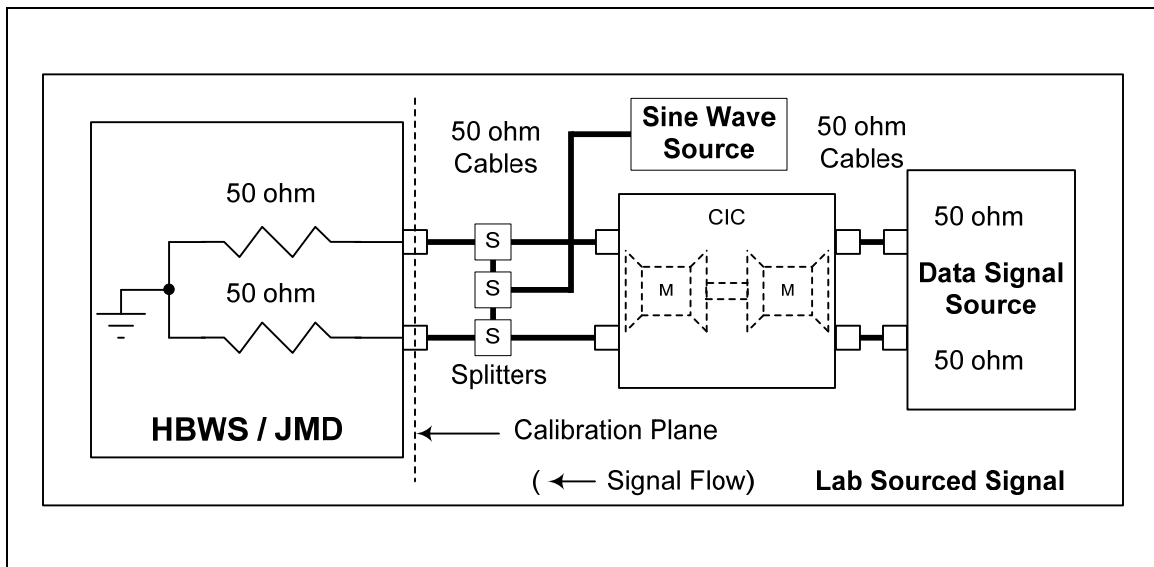


Figure 233 – Receiver jitter and CM tolerance test – setting TJ and CM levels (Gen3i)

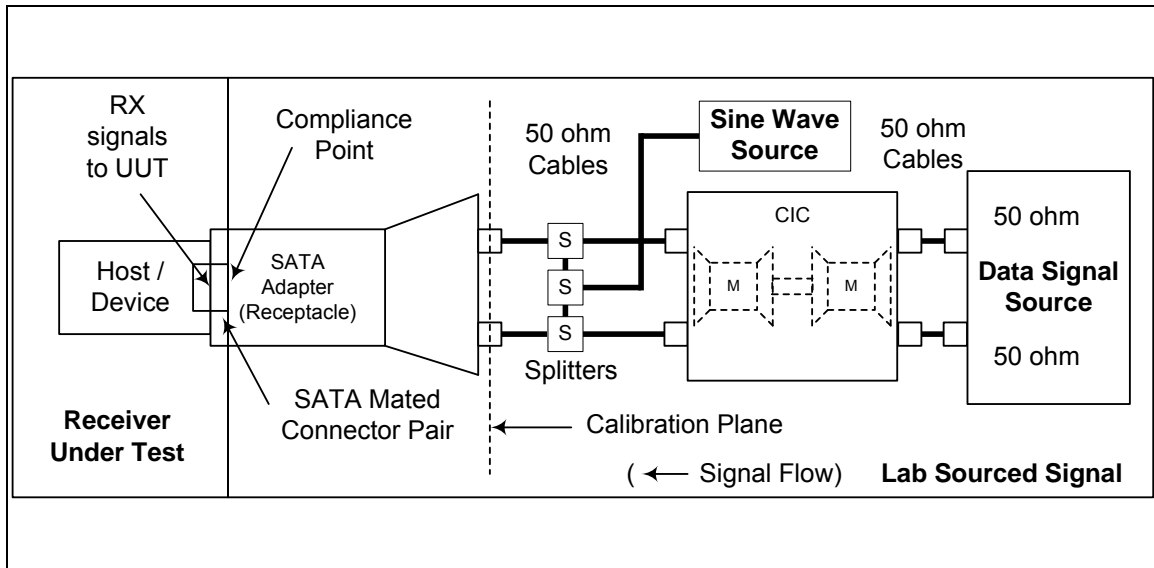


Figure 234 – Receiver jitter and CM tolerance test (Gen3i)

### 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

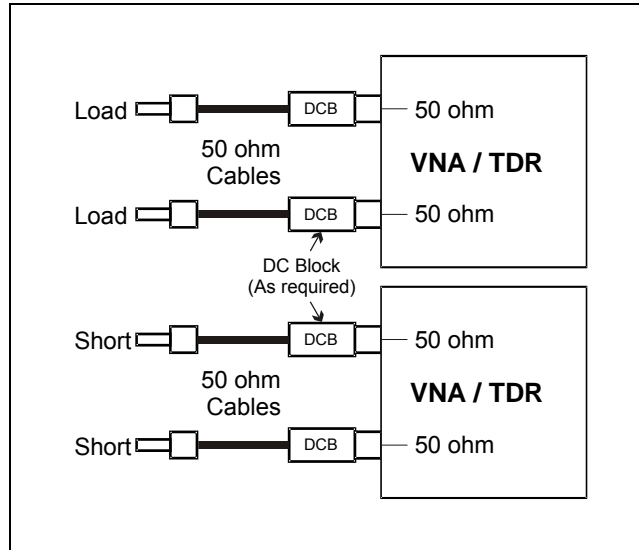
$$v_{cm} = \frac{v_2 + v_1}{2}$$

$$i_{cm} = \frac{i_2 + i_1}{2}$$

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 235 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 235 – 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 236 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 unit under test; they introduce measurement uncertainty.

EXAMPLE - The attenuation loss in the test adapter reduces the reflection from the unit under test 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.

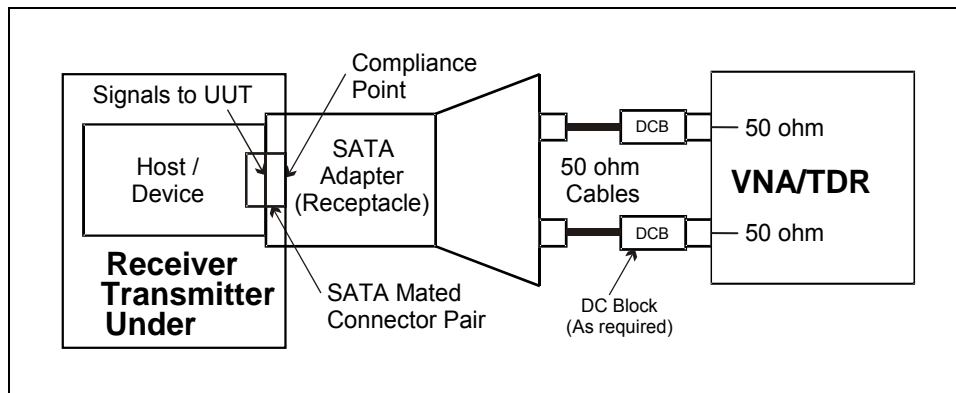
The return loss of the adapter may affect the measured return loss higher or lower as the reflection from the adapter either adds or subtracts from the reflection from the unit under test. A well-matched adapter with 20 dB return loss may affect the measurements of a unit under test with return loss of 5 dB by  $\pm 1$  dB. These effects are most pronounced at higher frequencies.

The adapter affects the measured reflection by the following measurement uncertainty equation

$$s_{11m} = \varepsilon_{00} + \varepsilon_{01}\varepsilon_{10}s_{11a} + \varepsilon_{01}\varepsilon_{10}\varepsilon_{11}(s_{11a})^2$$

where  $\varepsilon_{10}$  and  $\varepsilon_{01}$  are the attenuation loss,  $\varepsilon_{00}$  is the input reflection, and  $\varepsilon_{11}$  is the output reflection of the adapter.  $s_{11m}$  is the measured reflection, and  $s_{11a}$  is the actual reflection from the unit under test. 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 236 – 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 Unit Interval 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/decade.

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  $\pm 3\%$ .

### **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 237 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 240 of a transmitter, Ch5 and Ch6 are the two single-ended signals of the differential pair, the M5-trace is the inverted Ch5 (-Ch5), and M6 is the inverted Ch6 (-Ch6). Vertical cursors are used to measure the time between crossovers as the intra-pair skew.

Receivers shall be tested 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 238. Use the HFTP as the pattern when setting the skew. The skew measurement is performed as described above for the transmitter.

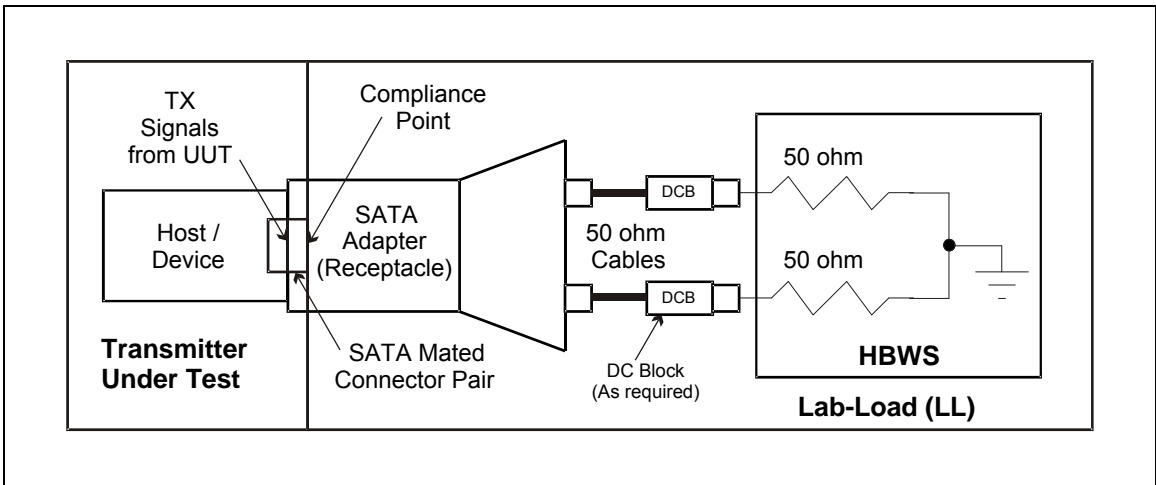


Figure 237 – Intra-pair skew test for a transmitter

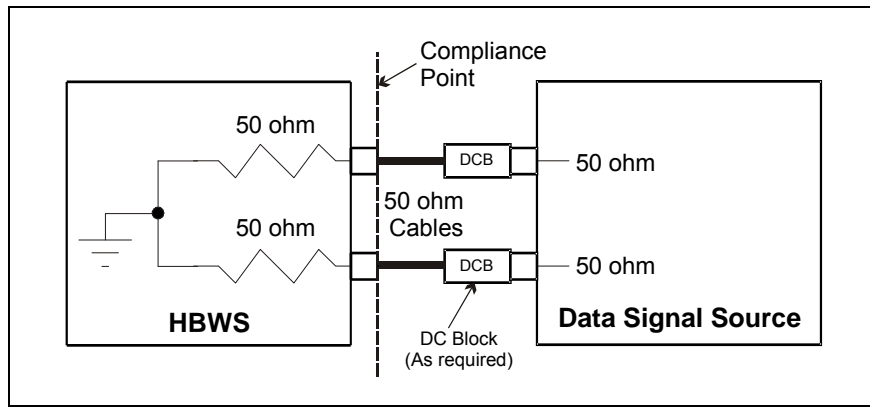


Figure 238 – Receiver intra-pair skew test – setting levels

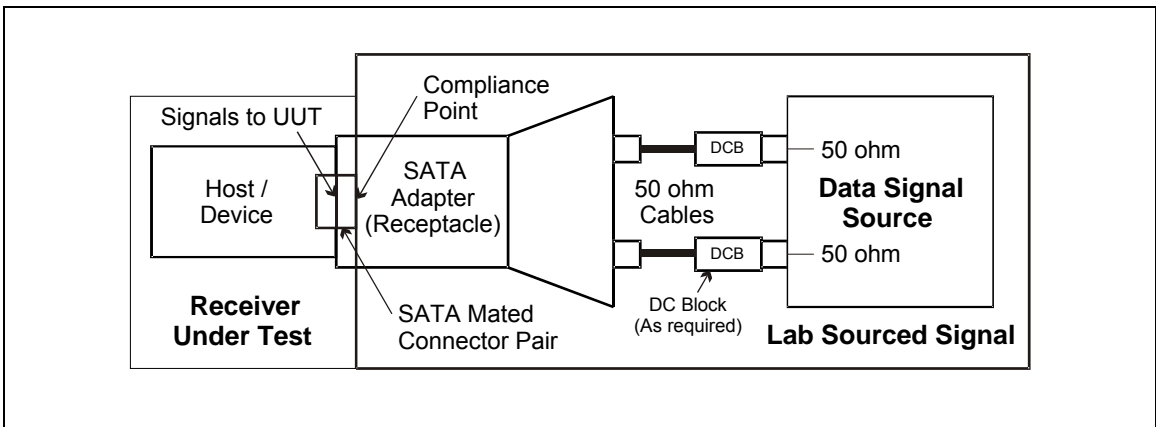


Figure 239 – Receiver intra-pair skew test



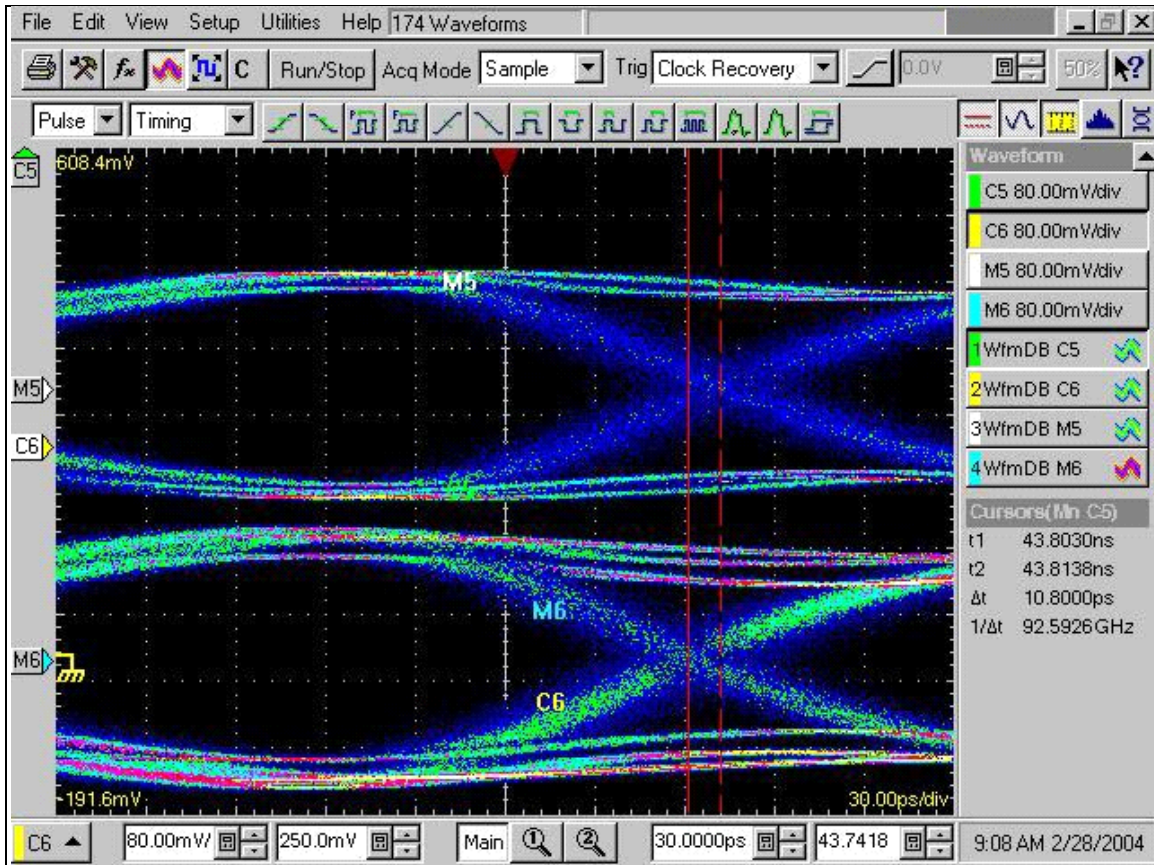


Figure 240 – Example intra-pair skew test for transmitter (10.8 ps)

### 7.6.18 Sequencing transient voltage

Figure 241 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  $V_p$  or  $V_n$ , shall remain in the voltage range specified.

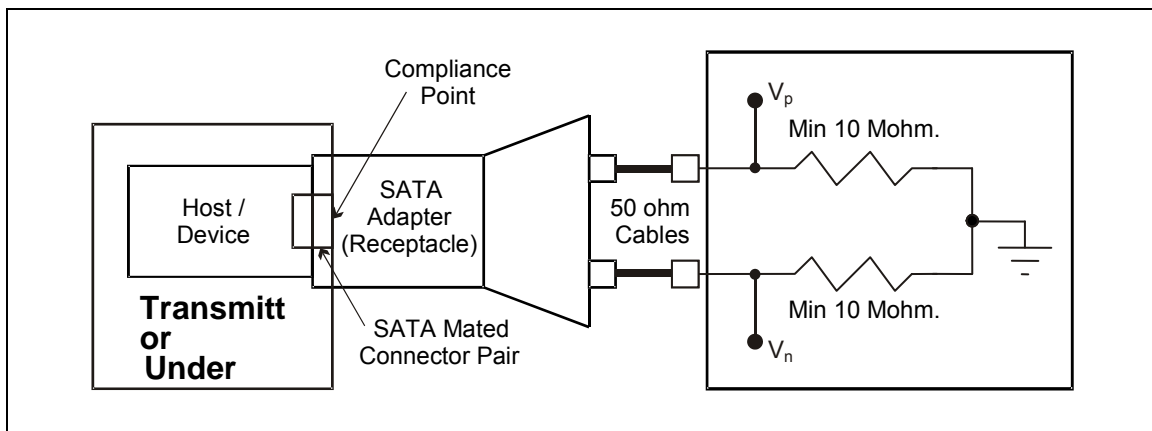


Figure 241 – Tx/Rx sequencing transient Voltage measurement

### 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 unit under test 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 242 shows the connections to each coupling capacitor. Each coupling capacitor shall be lower than the specified maximum.

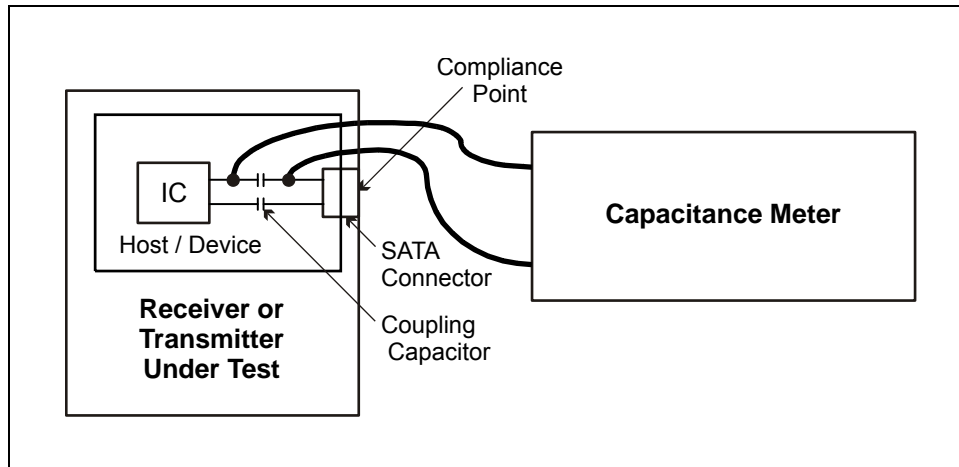


Figure 242 – AC coupled capacitance measurement

### 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 217 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:

$$\frac{|TX+ \text{ amplitude} - TX- \text{ amplitude}|}{\text{average}}$$

where:

$$\text{average} = \frac{\text{TX+ amplitude} + \text{TX- amplitude}}{2}$$

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 217 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:

$$\frac{|\text{TX+, rise} - \text{TX-, fall}|}{\text{average}}$$

where:

$$\text{average} = \frac{\text{TX+, rise} + \text{TX-, fall}}{2}$$

and:

$$\frac{|\text{TX+, fall} - \text{TX-, rise}|}{\text{average}}$$

where:

$$\text{average} = \frac{\text{TX+, fall} + \text{TX-, rise}}{2}$$

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 217). 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 217. 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  $\text{bitrate}/2$  (fundamental), using first order filtering. Separate channels shall be used for Tx+ and Tx-, with the common mode being  $(\text{Tx+} + \text{Tx-}) / 2$ . 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 217 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  $(\text{Tx+} + \text{Tx-})/2$  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 217 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  $\text{Tx+} - \text{Tx-}$  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 the detector shall detect; and
- b) if below the minimum threshold the detector shall not detect.

Figure 243 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 34 - Note the same method is used to calibrate the lab-sourced signal amplitude as defined in 7.6.8.

NOTE 35 - 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).

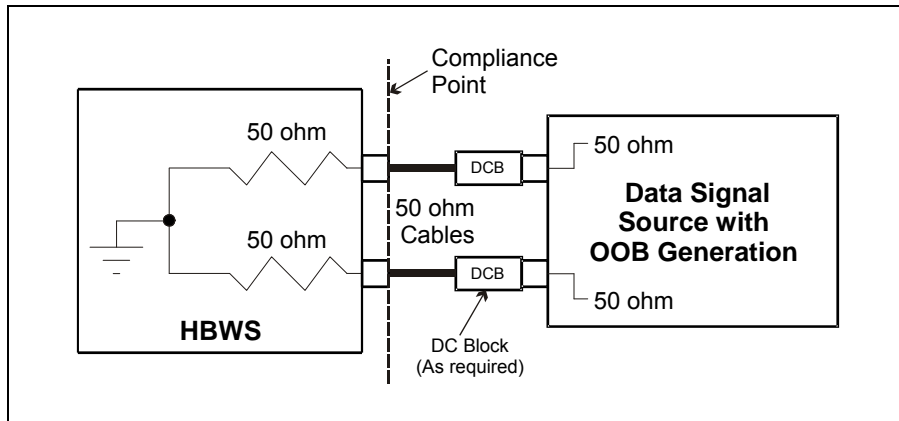


Figure 243 – Squelch detector threshold test – setting levels

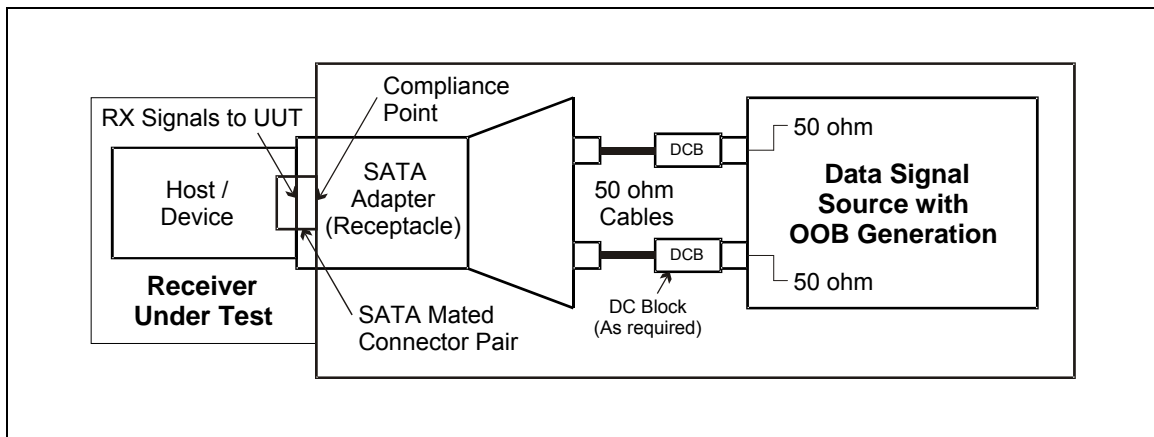


Figure 244 – Squelch detector threshold test

## **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 and Gen2 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. 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 245 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.

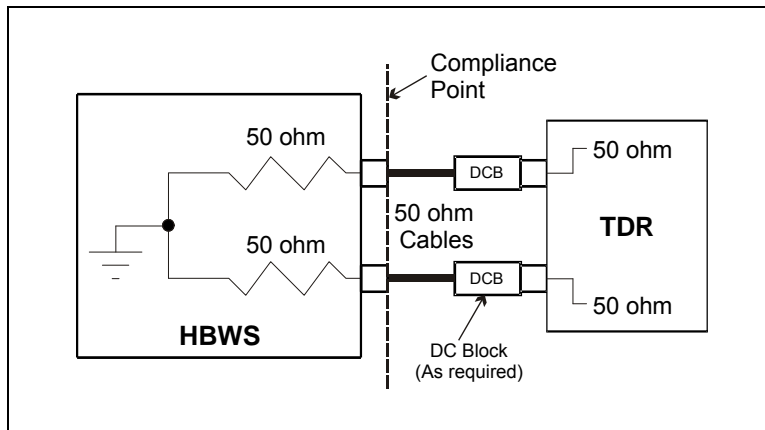


Figure 245 – TDR differential impedance test – setting risetime

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.

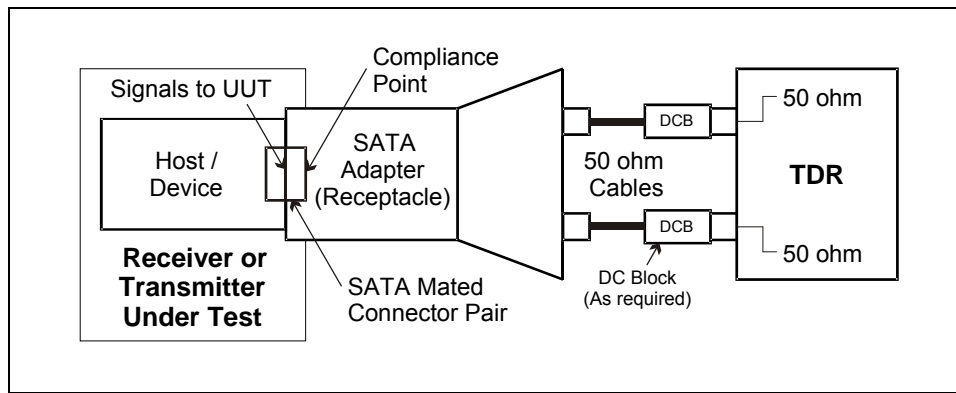


Figure 246 – TDR impedance test

### 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 247 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.

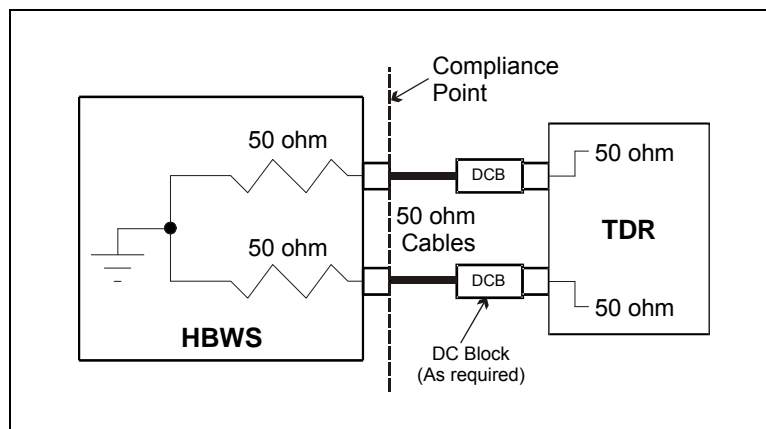


Figure 247 – TDR single-ended impedance test – setting risetime

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 250 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 248 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  $(V_p + V_n)/2$  and this term shall be in the range specified.

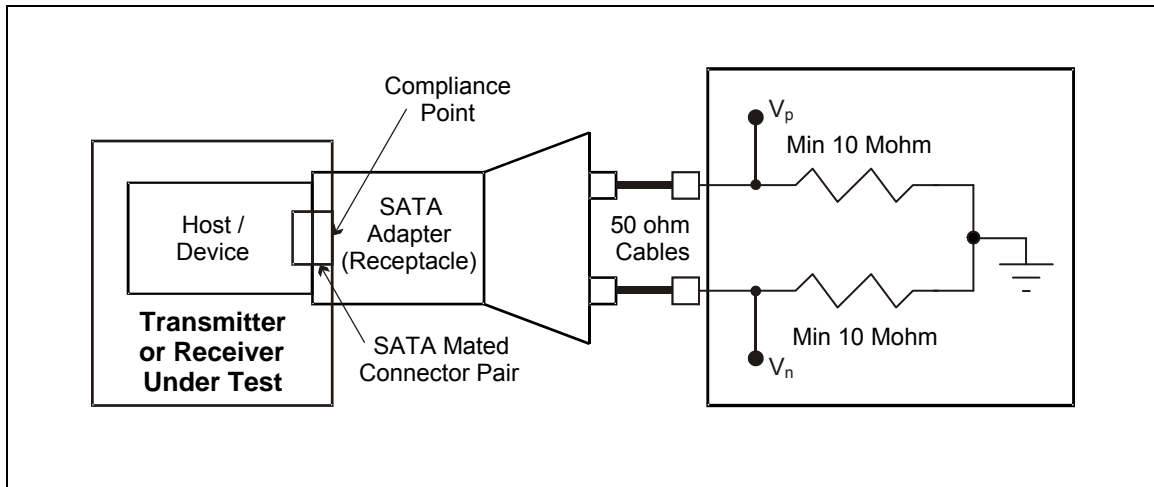


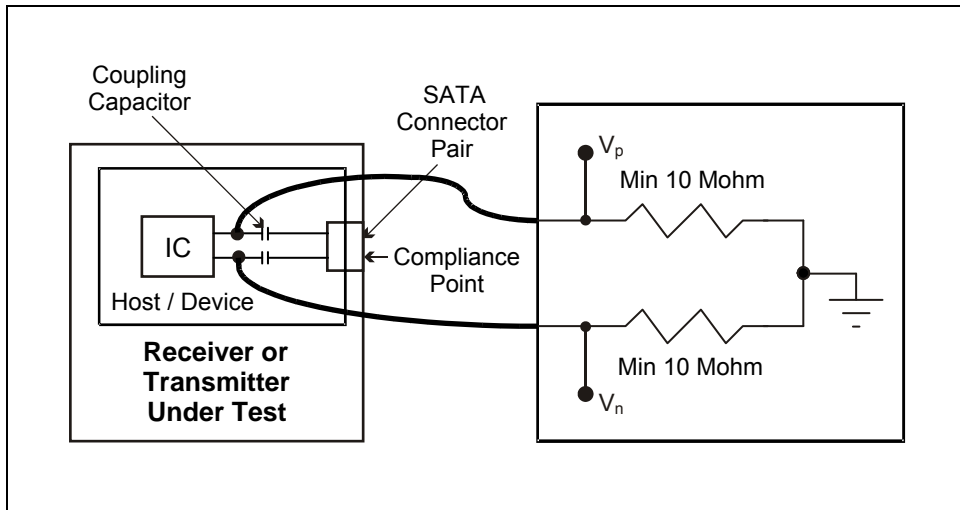
Figure 248 – DC coupled common mode Voltage measurement

### 7.6.31 AC coupled common mode voltage (Gen1i, Gen1m)

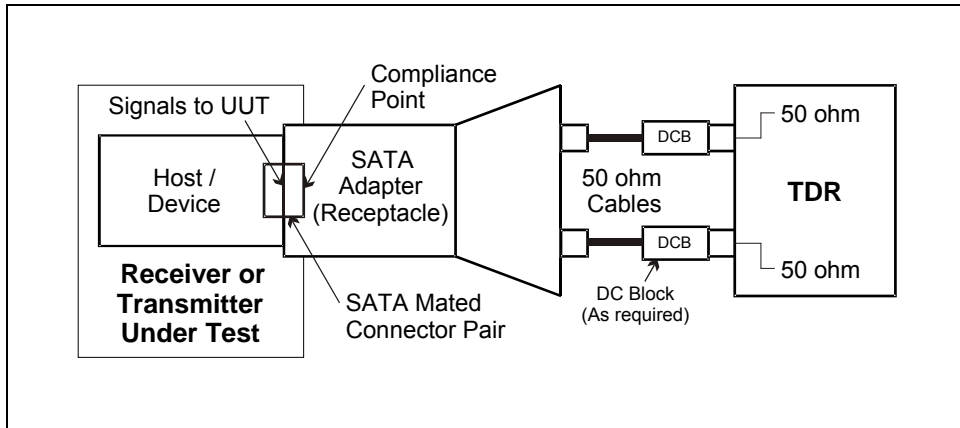
This measurement is only applicable to AC coupled transmitters and receivers. The AC coupled common mode voltage is not directly observable at the SATA connector.

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 249 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  $(V_p + V_n)/2$  and this term shall be in the range specified.



**Figure 249 – AC coupled common mode Voltage measurement**



**Figure 250 – TDR impedance test**

### 7.6.32 Sequencing transient voltage - lab-load (Gen3i, Gen3u)

Figure 251 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  $V_p$  or  $V_n$ , 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.

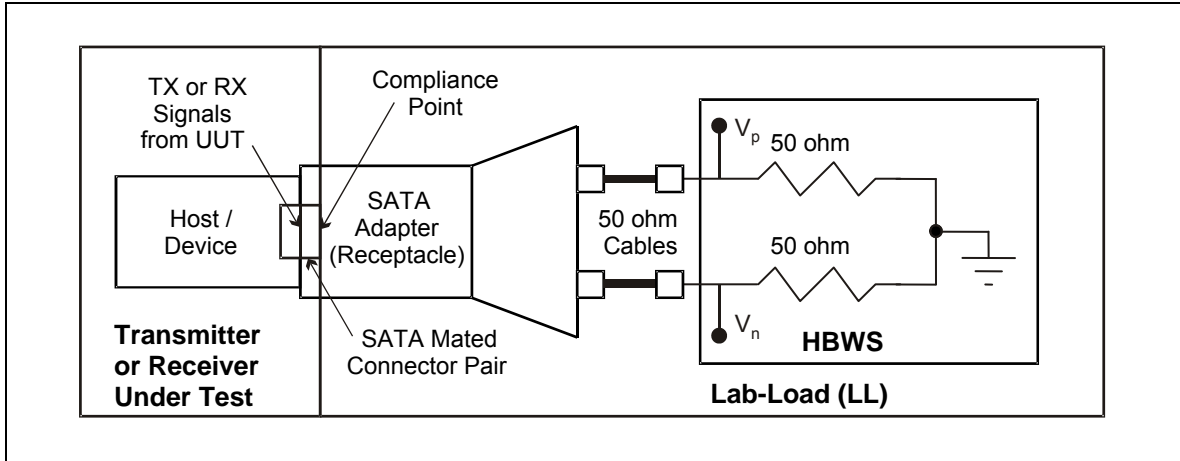


Figure 251 – Sequencing transient Voltage lab-load (LL)

## 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<sub>P</sub> 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 U<sub>IOOB</sub>. 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<sub>P</sub> primitives as legitimate OOB signal content. The alternate OOB sequence defined in this section has different characteristics than the ALIGN<sub>P</sub> 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.

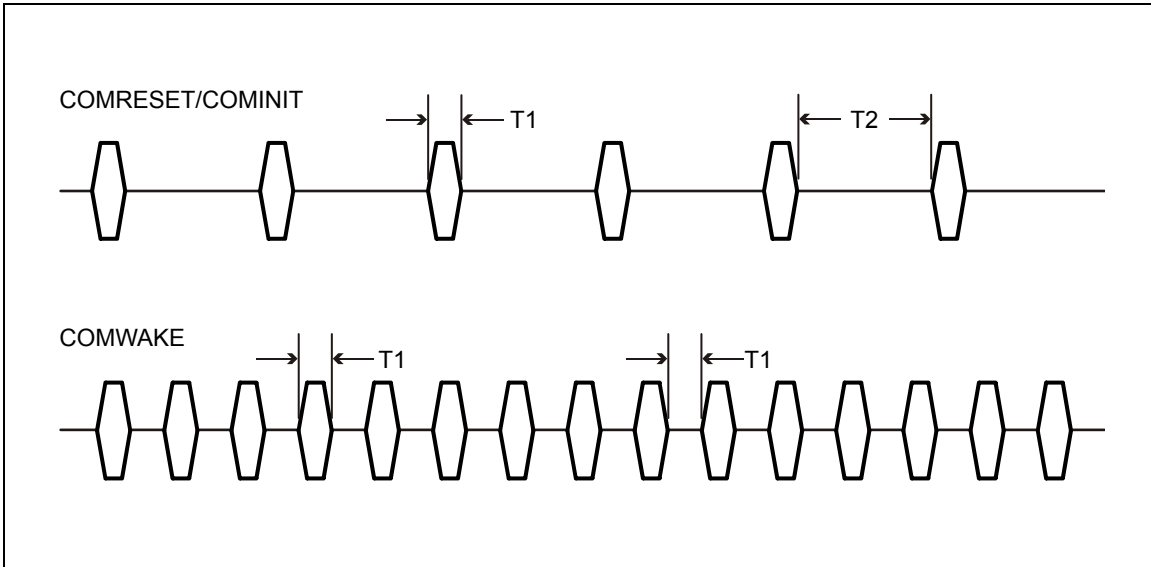


Figure 252 – OOB signals

Table 76 – OOB signal times

Time	Value
T1	160 $U_{IOOB}$ (Nom 106.7 ns)
T2	480 $U_{IOOB}$ (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 ( $V_{cm,ac}$ ) according to 7.4, and shall settle to within 25 mV of the previous state common mode voltage within  $T_{settle,cm}$ , according to 7.4. The following figure shows several transmitter examples, and how the transition to and from the idle state may be implemented.

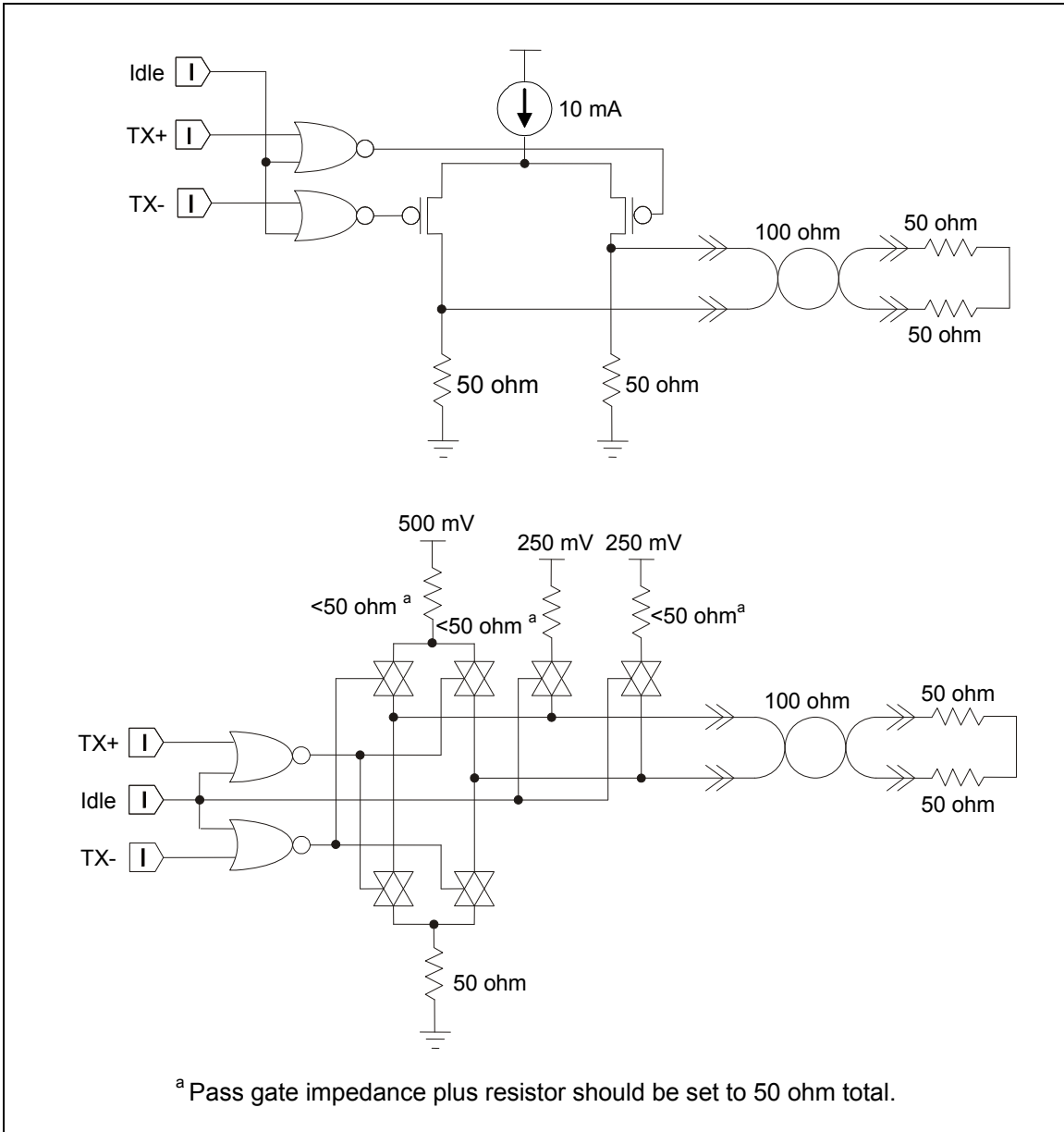
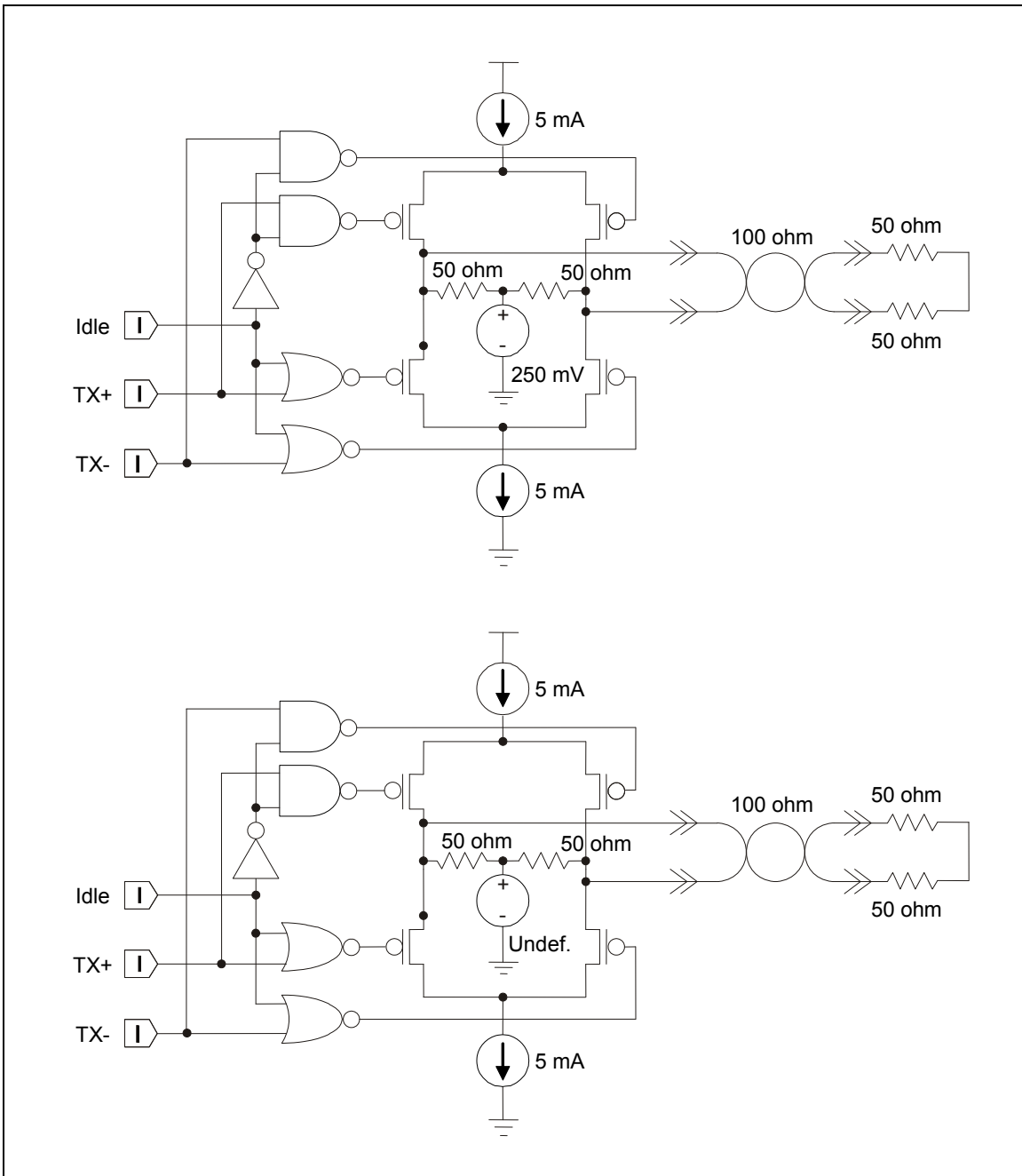


Figure 253 – Transmitter examples



**Figure 254 – Transmitter examples (concluded)**

### 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;

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

- 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 255. The interface shall be held inactive for at least 525 ns after the last burst to ensure far-end detector detects the negation properly.

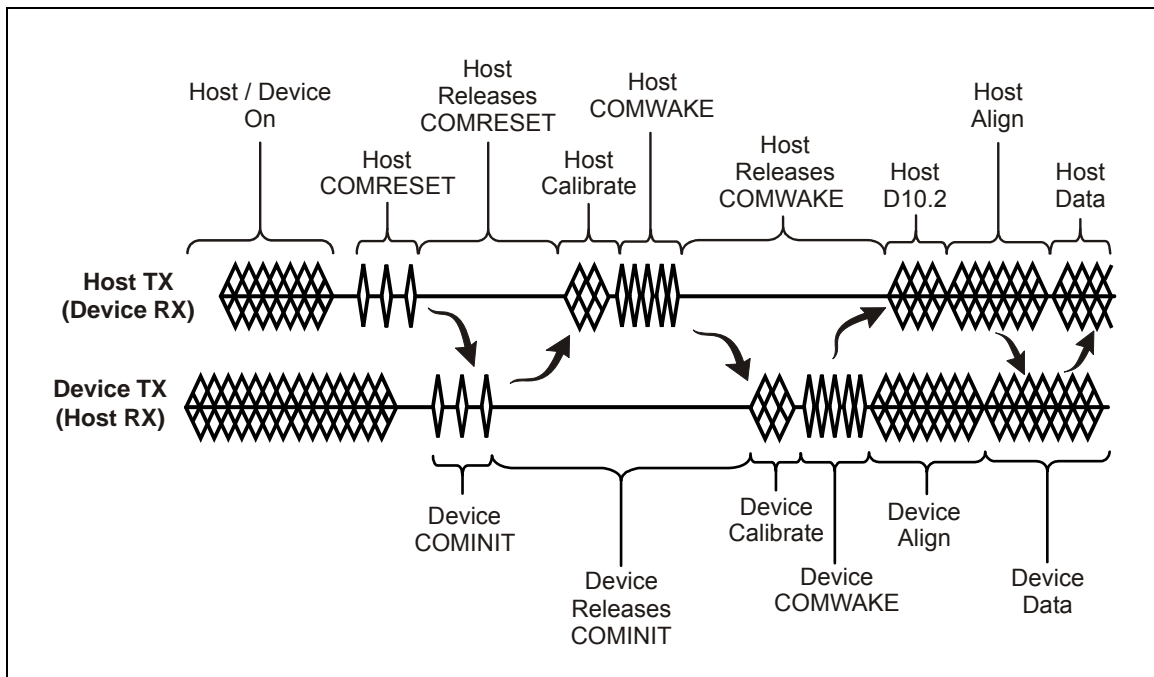


Figure 255 – COMRESET sequence

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<sub>P</sub> sequence starting at the device's highest supported speed. After ALIGN<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> sequence and, if ready, returns the ALIGN<sub>P</sub> 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<sub>P</sub>. This ensures interoperability with multi-generational and synchronous designs. If no ALIGN<sub>P</sub> 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<sub>P</sub> sequence and, if ready, sends SYNC<sub>P</sub> indicating it is ready to start normal operation; and
- 9) upon receipt of three back-to-back non-ALIGN<sub>P</sub> 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 256.

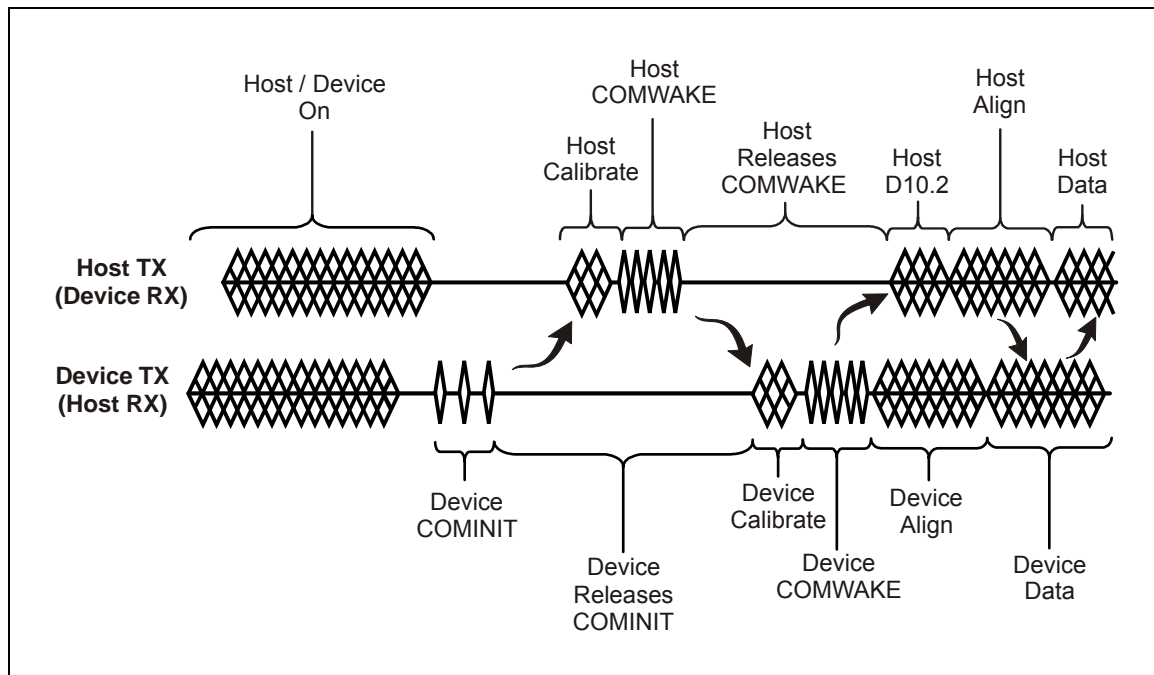


Figure 256 – COMINIT sequence

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<sub>P</sub> sequence starting at the device's highest supported speed. After ALIGN<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> sequence and, if ready, returns the ALIGN<sub>P</sub> 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<sub>P</sub>. This ensures interoperability with multi-generational and synchronous designs. If no ALIGN<sub>P</sub> 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<sub>P</sub> sequence and, if ready, sends SYNC<sub>P</sub> indicating it is ready to start normal operation; and
- 7) Upon receipt of three back-to-back non-ALIGN<sub>P</sub> 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.

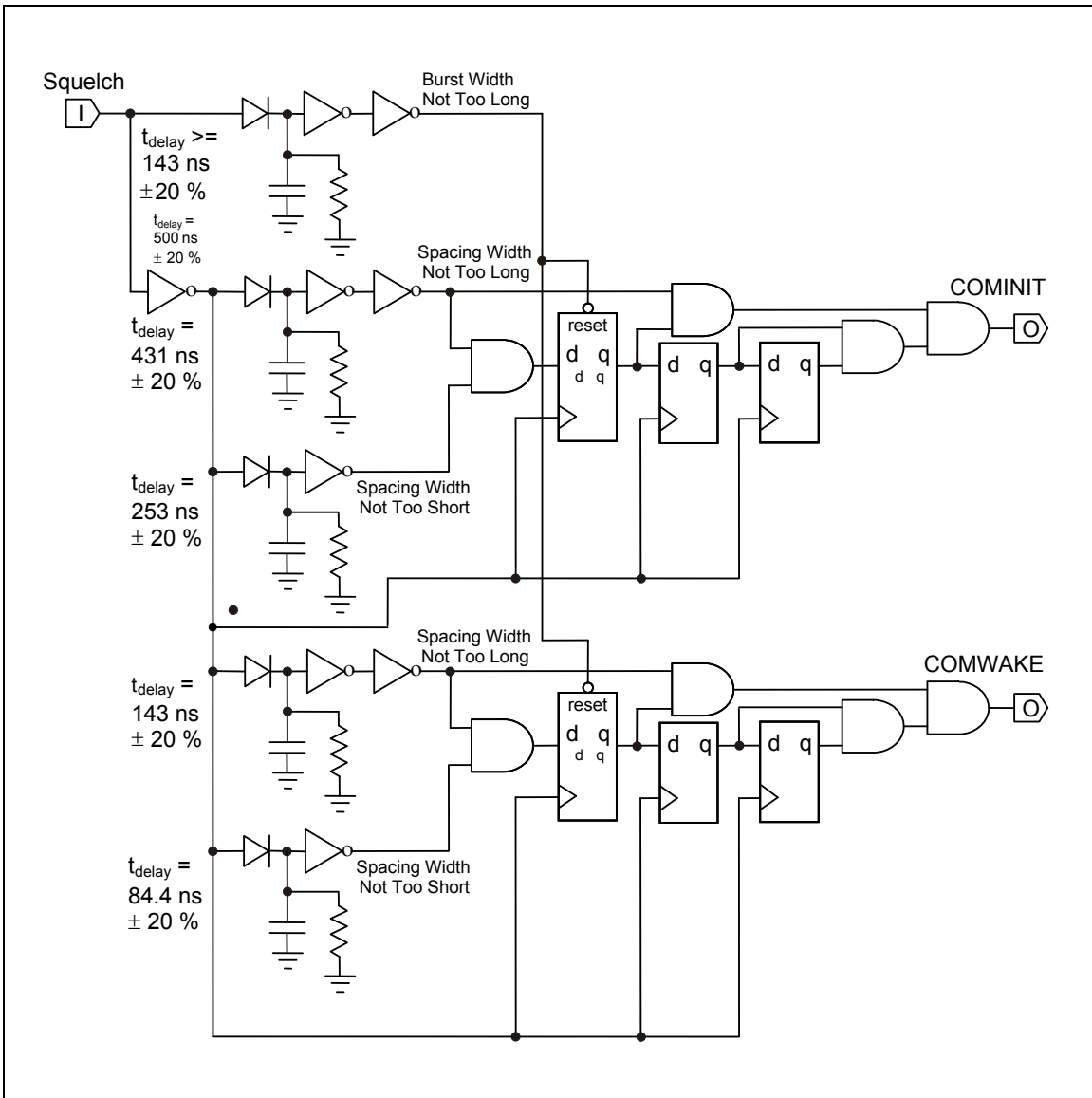
Each burst shall be 160 Gen1 UI long and each inter-burst idle state shall be 160 Gen1 UI long. A COMWAKE detector looks for four consecutive burst with a 106.7 ns spacing (nominal).

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 than 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 257 – 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 258, like the OOB Signal Detector figure shown in Figure 257, 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.

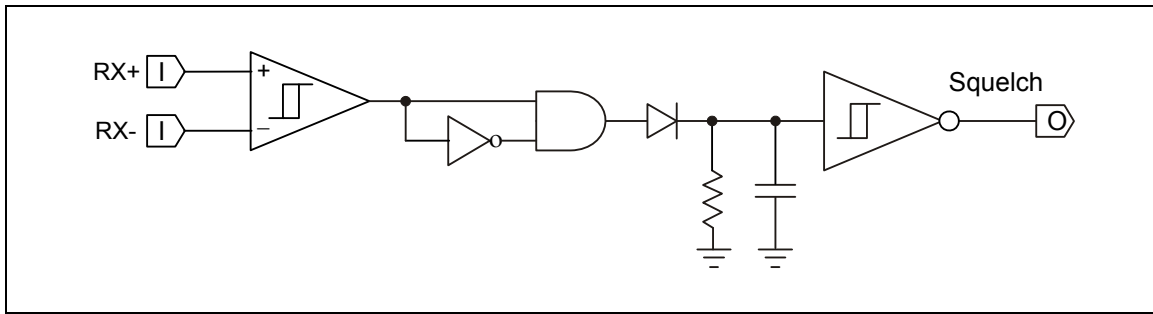


Figure 258 – Squelch detector

### 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. While 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 36 - 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<sub>P</sub> 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<sub>P</sub> primitives, followed by at most 254 non-ALIGN<sub>P</sub> Dwords. The cycle repeats starting with two consecutive ALIGN<sub>P</sub> primitives. The Link may issue more than

a single instance of two consecutive ALIGN<sub>P</sub> primitives but shall not send an odd number of ALIGN<sub>P</sub> primitives (i.e., ALIGN<sub>P</sub> primitives are always sent in pairs) except as noted for retimed loopback ( see 10.5.10).

ALIGN<sub>P</sub> consists of four characters as shown in Table 77.

**Table 77 – ALIGN<sub>P</sub>**

<b>(rd+)</b>	<b>(rd-)</b>	
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 13.7.9.4).

## 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 37 - 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<sub>P</sub> primitives or comma sequences for state transitions, the Phy shall ensure accurate detection of the ALIGN<sub>P</sub> primitives at the compatible signaling speed, with adequate implementation safeguards to ensure that there is no misdetection of ALIGN<sub>P</sub> in the HP6:HR\_AwaitAlign state in light of aliasing effects given the different data rates of ALIGN<sub>P</sub> 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\_AwaitNoCOMINITstate 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 259) 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.

<b>HP1: HR_Reset<sup>a</sup></b>	Transmit COMRESET <sup>b c d</sup> . If asynchronous signal recovery is supported, then clear ResumePending to 0.	
1. Power-on reset and explicit reset request negated.	→	HR_AwaitCOMINIT
2. Power-on reset or explicit reset request asserted.	→	HR_Reset
<sup>a</sup> 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. <sup>b</sup> Shall transmit COMRESET for a minimum of 6 bursts (and a multiple of 6) <sup>c</sup> 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. <sup>d</sup> 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.		

<b>HP2: HR_AwaitCOMINIT</b>	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

<b>HP2B: HR_AwaitNoCOMINIT<sup>a</sup></b>	Interface quiescent.	
1. COMINIT not detected from device.	→	HR_Calibrate
2. COMINIT detected from device.	→	HR_AwaitNoCOMINIT
<sup>a</sup> 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.		

<b>HP3: HR_Calibrate</b>	Perform calibration <sup>a</sup> .	
1. Calibration complete or bypass not implemented.	→	HR_COMWAKE
2. Calibration not complete.	→	HR_Calibrate
<sup>a</sup> Calibration is optional. If bypassed or not implemented, proceed directly to HR_COMWAKE.		

<b>HP4: HR_COMWAKE</b>	Transmit COMWAKE.	
1. COMWAKE not detected from device.	→	HR_AwaitCOMWAKE
2. COMWAKE detected from device.	→	HR_AwaitNoCOMWAKE

**Figure 259 – Host Phy initialization state machine**

(part 1 of 5)



<b>HP5: HR_AwaitCOMWAKE</b>		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
<b>HP5B: HR_AwaitNoCOMWAKE</b>		Interface quiescent	
1.	COMWAKE not detected from device.	→	HR_AwaitAlign
2.	COMWAKE detected from device.	→	HR_AwaitNoCOMWAKE

**Figure 259 –Host Phy initialization state machine (part 2 of 5)**

HP6: HR_AwaitAlign	Host transmits D10.2 characters at lowest supported rate <sup>b e</sup>	
1. If reset speed negotiation is not supported and ALIGN <sub>P</sub> detected from device (at any supported speed) <sup>c</sup> . If reset speed negotiation is supported and ALIGN <sub>P</sub> detected from device at expected speed.	→	HR_AdjustSpeed
2. ALIGN <sub>P</sub> 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 <sub>P</sub> is detected at a lower speed.	→	HR_Reset <sup>a d f</sup>
3. ALIGN <sub>P</sub> 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
<p><sup>a</sup> Host retries the power-on sequence indefinitely unless explicitly turned off by the Application layer.</p> <p><sup>b</sup> 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.</p> <p><sup>c</sup> 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<sub>P</sub>.</p> <p>EXAMPLE - A compliant host may detect the negation of COMWAKE in as little as 112 ns, such a host should 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<sub>P</sub> primitives.</p> <p><sup>d</sup> The Host Phy Initialization state machine may use the transition to HR_Reset as a method of speed negotiation.</p> <p><sup>e</sup> The Device may respond with D10.2 if out of lock (see DR_SendAlign).</p> <p><sup>f</sup> If ALIGN<sub>P</sub> 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)</p>		

HP7: HR_SendAlign	Transmit ALIGN <sub>P</sub> at speed detected	
1. Three back-to-back non-ALIGN <sub>P</sub> primitives <sup>b</sup> detected from device.	→	HR_Ready
2. Three back-to-back non-ALIGN <sub>P</sub> primitives not detected from device.	→	HR_SendAlign <sup>a</sup>
<p><sup>a</sup> Host retries indefinitely unless explicitly turned off by the Application layer</p> <p><sup>b</sup> Non-ALIGN<sub>P</sub> primitives may be detected by the presence of the K28.3 control character in the Byte 0 position.</p>		

**Figure 259 –Host Phy initialization state machine (part 3 of 5)**

<b>HP8: HR_Ready</b>	Transmit Word from Link <sup>a</sup> .	
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 <sup>b</sup> 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 <sup>b</sup> .	→	HR_Reset
<sup>a</sup> 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. <sup>b</sup> The latency that a host elects to initiate an optional signal recovery poll is implementation specific but shall be greater than the ALIGN <sub>P</sub> transmit interval.		

<b>HP9: HR_Partial</b>	Interface quiescent. If asynchronous signal recovery is supported, then set ResumePending = 1.	
1. Partial signal from Link negated and no COMWAKE detected from device <sup>a,b</sup> .	→	HR_COMWAKE
2. Partial signal from Link negated and COMWAKE detected from device <sup>a,b</sup> .	→	HR_AwaitNoCOMWAKE
3. Slumber signal from the Link asserted and host Automatic Partial to Slumber transitions are supported <sup>c</sup> .	→	HR_Slumber
4. Partial signal from Link asserted.	→	HR_Partial
<sup>a</sup> Host Phy shall remember if COMWAKE was detected during Partial to determine if the wakeup request originated from the host or the Phy. <sup>b</sup> 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. <sup>c</sup> 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 259 –Host Phy initialization state machine (part 4 of 5)**

<b>HP10: HR_Slumber</b>	Interface quiescent. If asynchronous signal recovery is supported, then set ResumePending = 1.	
1. Slumber signal from Link negated and no COMWAKE detected from device <sup>a,b</sup> .	→	HR_COMWAKE
2. Slumber signal from Link negated and COMWAKE detected from device <sup>a,b</sup> .	→	HR_AwaitNoCOMWAKE
3. Slumber signal from Link asserted.	→	HR_Slumber
<sup>a</sup> Host Phy shall remember if COMWAKE was detected during Slumber to determine if the wakeup request originated from the host or the Phy. <sup>b</sup> 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 it shall remain in this state.		

<b>HP11: HR_AdjustSpeed</b>	Interface undefined but not quiescent <sup>a</sup>	
1	Transition to appropriate speed completed.	→ HR_SendAlign
2	Transition to appropriate speed not completed.	→ HR_AdjustSpeed
<sup>a</sup> 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 unit interval 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 259 –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 260) 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.

<b>DP1: DR_Reset</b> <sup>a</sup>	Interface quiescent	
1.	COMRESET not detected and power-on reset negated.	→ DR_COMINIT
2.	COMRESET detected or power-on reset asserted.	→ DR_Reset
<sup>a</sup> This state is entered asynchronously any time in response to power-on reset or receipt of a COMRESET signal from the host.		

**Figure 260 –Device Phy initialization state machine**

(part 1 of 4)

<b>DP2: DR_COMINIT</b>		Transmit COMINIT <sup>a b</sup>	
1. Unconditional	→	DR_AwaitCOMWAKE	
<sup>a</sup> COMINIT transmitted for a 6 bursts duration <sup>b</sup> According to 13.1, 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.			
<b>DP3: DR_AwaitCOMWAKE</b>		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	
<b>DP3B: DR_AwaitNoCOMWAKE</b>		Interface quiescent	
1. COMWAKE not detected from host and part of power-on reset sequence <sup>a</sup> .	→	DR_Calibrate	
2. COMWAKE not detected from host and part of Partial/Slumber awake sequence <sup>a</sup> .	→	DR_COMWAKE	
3. COMWAKE detected from host.	→	DR_AwaitNoCOMWAKE	
<sup>a</sup> Device shall remember if it was sent to Partial or Slumber mode for proper wakeup action.			
<b>DP4: DR_Calibrate</b>		Perform calibration <sup>a</sup>	
1. Calibration complete or bypass not implemented.	→	DR_COMWAKE	
2. Calibration not complete.	→	DR_Calibrate	
<sup>a</sup> Calibration is optional. If bypassed or not implemented, proceed directly to DR_COMWAKE.			
<b>DP5: DR_COMWAKE</b>		Transmit COMWAKE	
1. Unconditional	→	DR_SendAlign	

Figure 260 – Device Phy initialization state machine (part 2 of 4)

DP6: DR_SendAlign	Transmit ALIGN <sub>P</sub> <sup>a b c e</sup>	
1. ALIGN <sub>P</sub> detected from host (device locked to incoming data) <sup>d</sup> .	→	DR_Ready
2. ALIGN <sub>P</sub> not detected from host and ALIGN <sub>P</sub> primitives transmitted for 54.6 us (i.e., 2 048 <sup>e</sup> Gen1 ALIGN <sub>P</sub> primitives) at speed other than lowest <sup>f</sup> .	→	DR_ReduceSpeed
3. ALIGN <sub>P</sub> not detected from host and ALIGN <sub>P</sub> primitives transmitted for 54.6 us (i.e., 2 048 <sup>e</sup> Gen1 ALIGN <sub>P</sub> primitives) at lowest speed <sup>f</sup> .	→	DR_Error
4. ALIGN <sub>P</sub> not detected from host and ALIGN <sub>P</sub> primitives transmitted for less than 54.6 us (i.e., 2 048 Gen1 ALIGN <sub>P</sub> primitives).	→	DR_SendAlign
<p><sup>a</sup> If this is part of a recovery from a Slumber or Partial power management state, the device shall send ALIGN<sub>P</sub> at the previously negotiated speed. For all other cases, ALIGN<sub>P</sub> should be sent at the device's fastest supported speed.</p> <p><sup>b</sup> ALIGN<sub>P</sub> primitives should be sent only at valid frequencies (i.e., if PLL not locked, send D10.2).</p> <p><sup>c</sup> After COMWAKE is released as specified in the OOB signaling section, the device shall ensure the interface is active (not quiescent).</p> <p><sup>d</sup> 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<sub>P</sub>. 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<sub>P</sub>.</p> <p><sup>e</sup> 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.</p> <p><sup>f</sup> 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.</p>		

DP7: DR_Ready <sup>a</sup>	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
<p><sup>a</sup> 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.</p>		

**Figure 260 – Device Phy initialization state machine (part 3 of 4)**

<b>DP8: DR_Partial</b>	Interface quiescent	
1. Partial signal from Link negated <sup>a</sup>	→	DR_COMWAKE
2. Partial signal from Link negated and COMWAKE detected from host <sup>a</sup>	→	DR_AwaitNoCOMWAKE
3. Slumber signal from Link asserted and device Automatic Partial to Slumber transitions enabled <sup>b</sup>	→	DR_Slumber
4. Partial signal from Link asserted	→	DR_Partial
<sup>a</sup> 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. <sup>b</sup> 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.		

<b>DP9: DR_Slumber</b>	Interface quiescent	
1. Slumber signal from Link negated <sup>a</sup>	→	DR_COMWAKE
2. Slumber signal from Link negated and COMWAKE detected from host <sup>a</sup>	→	DR_AwaitNoCOMWAKE
3. Slumber signal from Link asserted	→	DR_Slumber
<sup>a</sup> 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 it shall remain in this state.		

<b>DP10: DR_ReduceSpeed</b>	Interface quiescent	
1. Transition to a slower speed complete	→	DR_SendAlign <sup>a</sup>
2. Transition to a slower speed not complete	→	DR_ReduceSpeed
<sup>a</sup> 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.		

<b>DP11: DR_Error</b>	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 260 – 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<sub>P</sub> 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<sub>P</sub> detection/handshake, receivers should fully qualify the entire received ALIGN<sub>P</sub> 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<sub>P</sub> primitives are successfully decoded. Other possible means for ensuring accuracy of the ALIGN<sub>P</sub> 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 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.

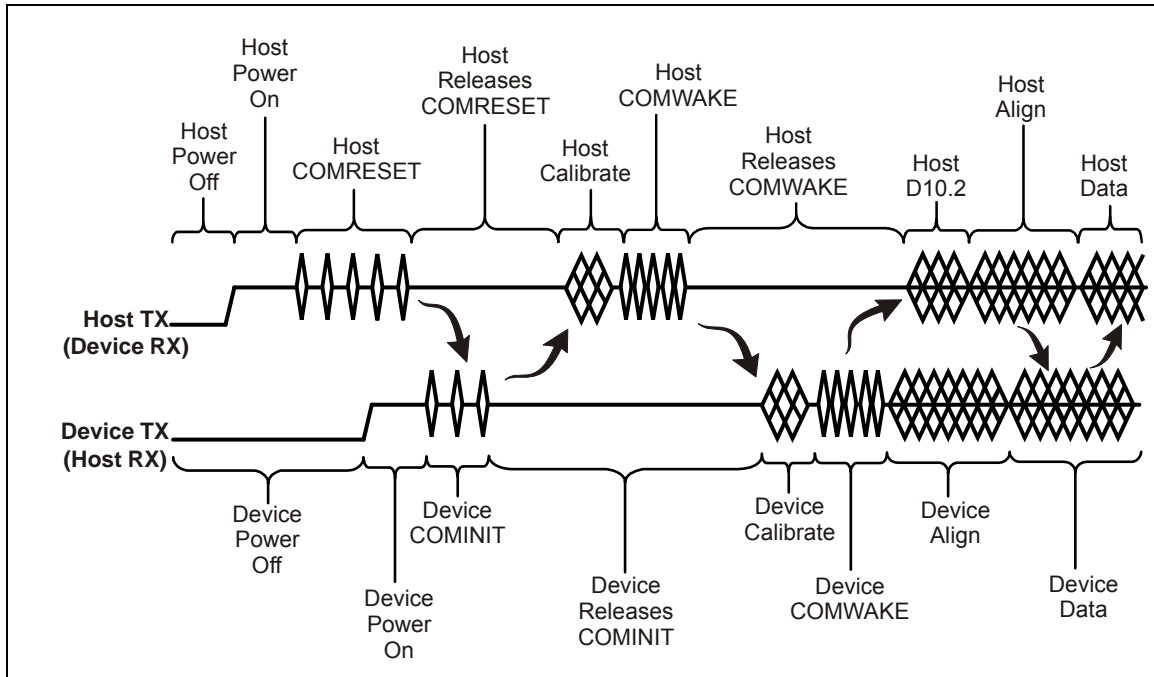
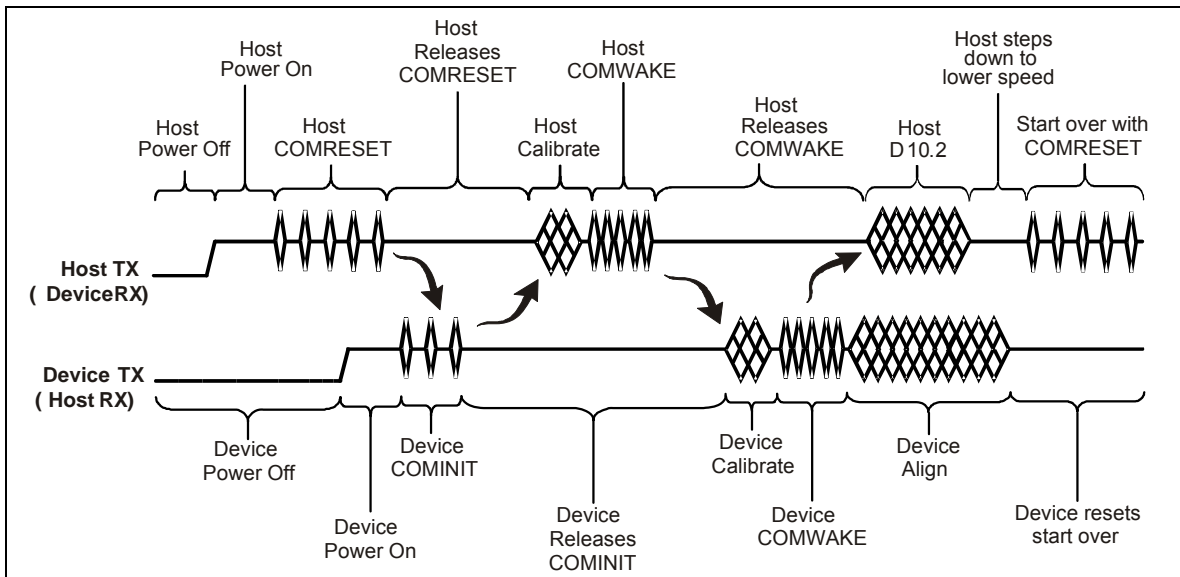


Figure 261 – Power-on sequence

#### 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<sub>P</sub> sequence starting at the device's highest supported speed. After ALIGN<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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 ALIGN<sub>P</sub> sequence and, if ready, returns the ALIGN<sub>P</sub> 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<sub>P</sub>. This insures interoperability with multi-generational and synchronous designs. If no ALIGN<sub>P</sub> 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 ALIGN<sub>P</sub> sequence and, if ready, sends the SYNC<sub>P</sub> primitive indicating it is ready to start normal operation; and
- 10) upon receipt of three back-to-back non-ALIGN<sub>P</sub> primitives, the communication link is established and normal operation may begin.



**Figure 262 – 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 ALIGN<sub>P</sub> sequence starting at the device's highest supported speed. After ALIGN<sub>P</sub> 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 ALIGN<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> sequence and, if ready, returns the ALIGN<sub>P</sub> 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<sub>P</sub>. This ensures interoperability with multi-generational and synchronous designs. If no ALIGN<sub>P</sub> 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<sub>P</sub> Dwords may issue a COMRESET on detection and rate verification of the lower speed ALIGN<sub>P</sub> 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 263;
  - 8) Device locks. The device locks to the ALIGN<sub>P</sub> sequence and, if ready, sends SYNC<sub>P</sub> indicating it is ready to start normal operation; and
  - 9) Upon receipt of three back-to-back non-ALIGN<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> primitives at the speed determined at power-on.

#### 8.4.4.4 PHYRDY to Partial/Slumber

##### 8.4.4.4.1 Host initiated

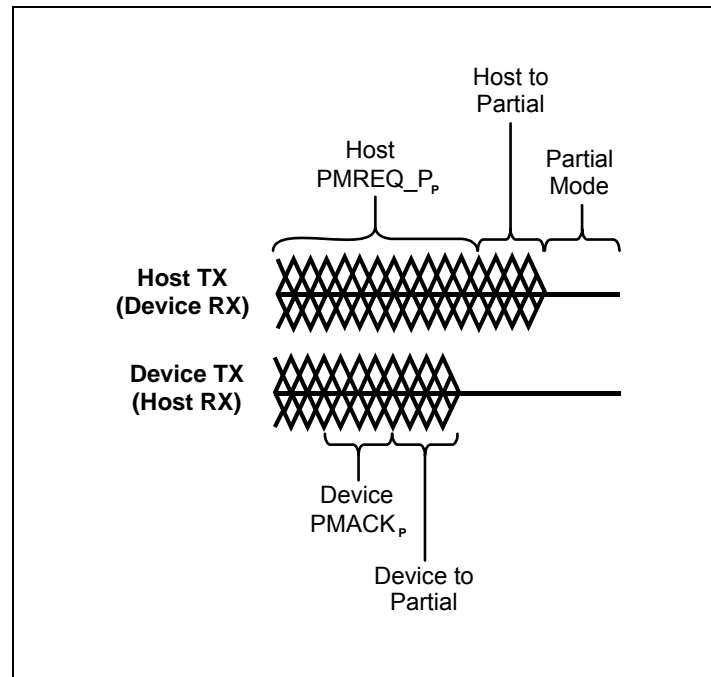


Figure 263 – PHYRDY to Partial – host initiated

NOTE 38 - For Slumber, the same sequence applies except PMREQ\_P<sub>P</sub> is replaced with PMREQ\_S<sub>P</sub> 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<sub>P</sub> primitive and transmits it to the host Phy layer;
- 4) host Phy layer serializes PMREQ\_P<sub>P</sub> primitives and transmits them to device Phy layer;
- 5) device Phy de-serializes PMREQ\_P<sub>P</sub> primitives and transmits them to device Link layer;
- 6) device Link layer decodes PMREQ\_P<sub>P</sub> 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\_P<sub>P</sub> primitive and transmits it four times to device Phy layer;
- 11) device Phy layer transmits between four and sixteen PMACK\_P<sub>P</sub> primitives to host Phy layer;
- 12) device Link layer places device Phy layer in Partial state;
- 13) host Phy layer de-serializes PMACK\_P<sub>P</sub> primitives and transmits them to host Link layer;
- 14) host Link layer decodes PMACK\_P<sub>P</sub> 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

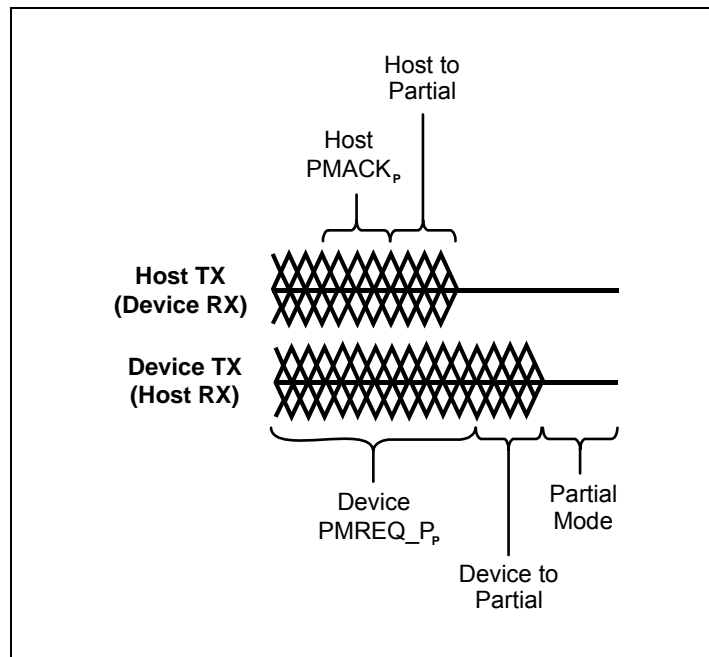


Figure 264 – PHYRDY to Partial – device initiated

#### 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<sub>P</sub> primitive and transmits it to device Phy layer;
- 4) device Phy layer serializes PMREQ\_P<sub>P</sub> primitives and transmits them to host Phy layer;
- 5) host Phy de-serializes PMREQ\_P<sub>P</sub> primitives and transmits them to host Link layer;
- 6) host Link layer decodes PMREQ\_P<sub>P</sub> primitives and transmits request to host Transport layer;
- 7) host Transport layer transmits request to host Application layer;

NOTE 39 - 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<sub>P</sub> and transmits it four times to host Phy layer;
- 11) host Phy layer transmits between four and sixteen PMACK<sub>P</sub> 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<sub>P</sub> primitives and transmits them to device Link layer;
- 15) device Link layer decodes PMACK<sub>P</sub> 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 265 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.

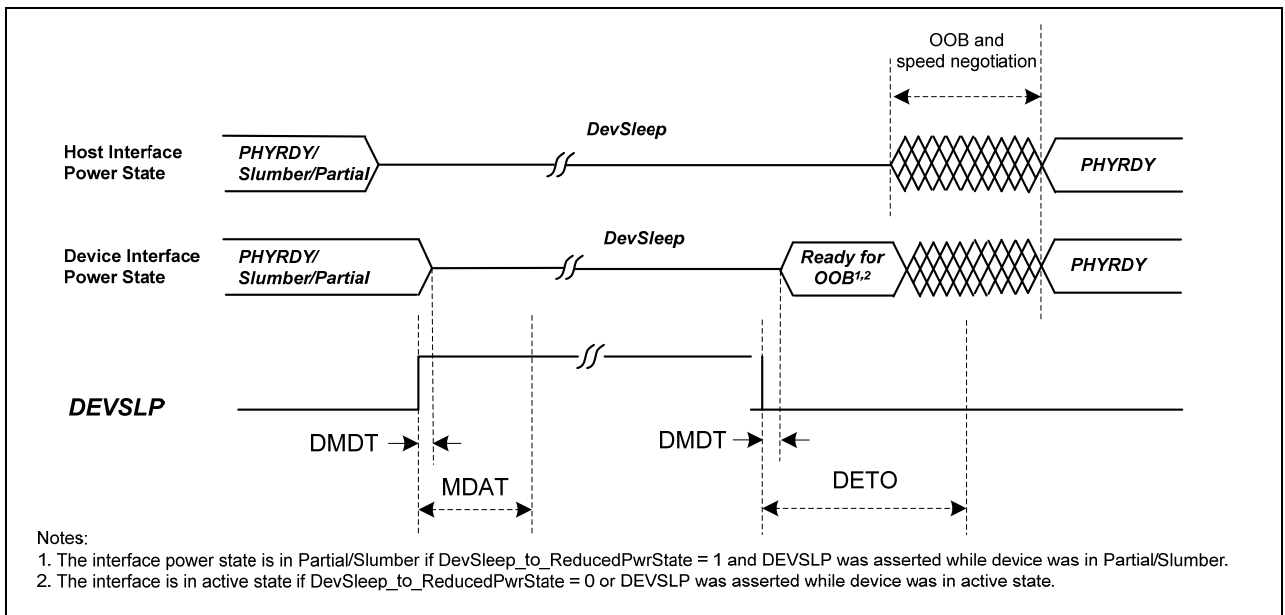


Figure 265 – DEVSLP protocol overview

Table 79 – DevSleep timing parameters

Symbol	Parameter	Value
MDAT	<b>Minimum DEVSLP Assertion Time</b> , minimum time that the host shall assert DEVSLP, once it has been asserted.	10 ms unless otherwise specified in Identify Device Data log.
DMDT	<b>DEVSLP Minimum Detection Time</b> , minimum time the device needs to de-bounce the DEVSLP signal after detecting it has been asserted.	10 us
DETO	<b>DevSleep Exit Timeout</b> , 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.

### 8.5.1.1 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., PLL's, 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 13.7.9).

After negating DEVSLP, the host shall keep it negated until the host Phy has reached the PHYRDY interface power state (see 8.1).

### 8.5.1.2 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.

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., PLL's, 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 13.7.9);

- 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.

### 8.5.2 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 266 is an example of a DEVSLP implementation for illustrative purposes.

NOTE 40 - The host should not rely on particular device resistor values.

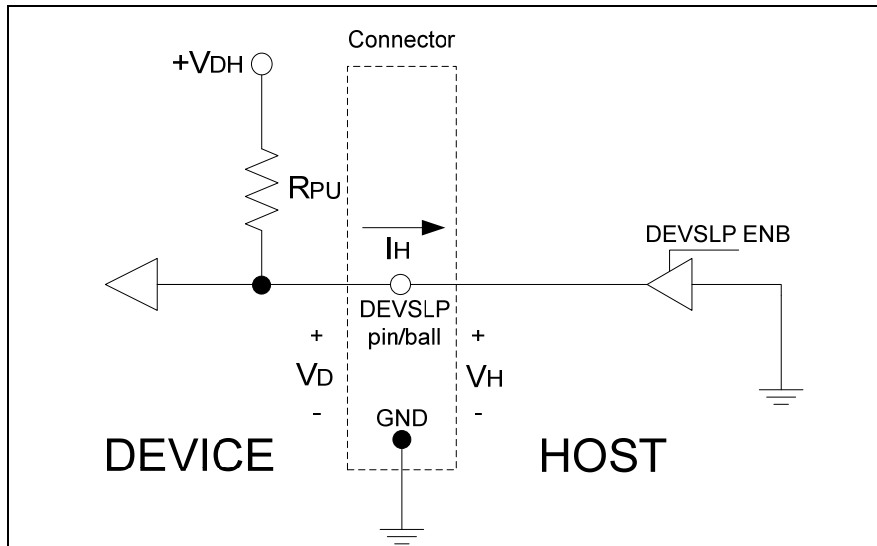


Figure 266 – Example DEVSLP electrical block diagram

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 & 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 & 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



## 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.

#### 9.1.1 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<sub>P</sub>, CRC, EOF<sub>P</sub>, 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.

#### 9.1.2 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<sub>P</sub>, CRC, EOF<sub>P</sub>);
- 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.

## 9.2 Encoding method

### 9.2.1 Encoding method overview

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.

### 9.2.2 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.

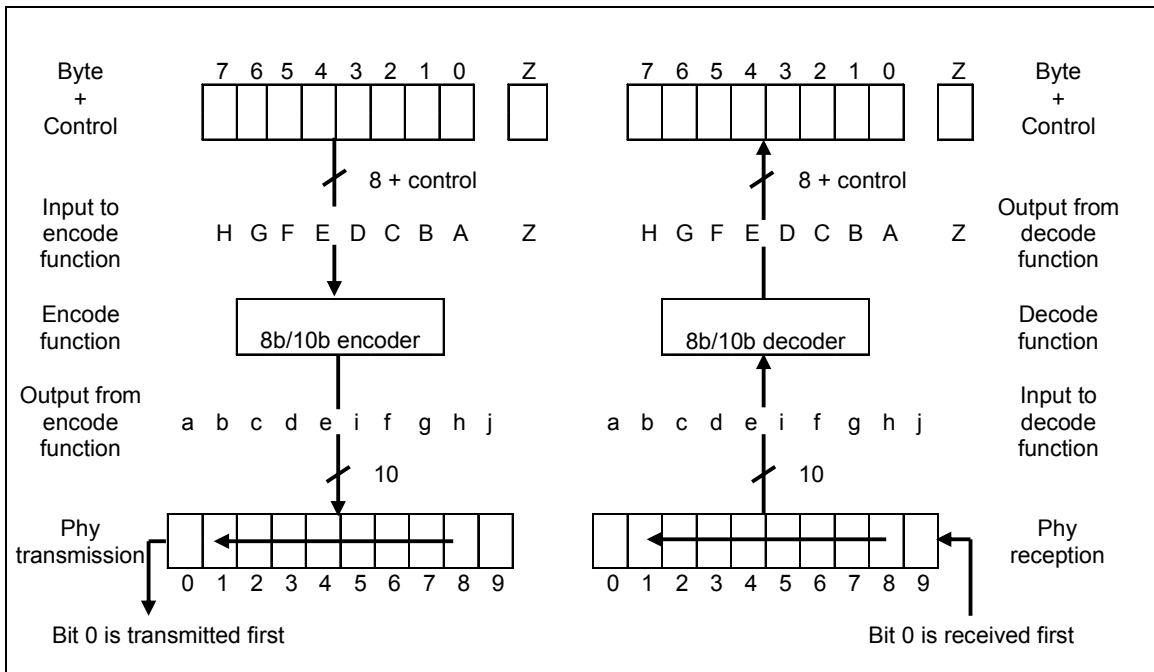
Table 82 illustrates the association between the numbered unencoded bits in a byte, the control variable, and the letter-labeled bits in the encoding scheme.

**Table 82 – 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 83 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 83 – Conversion examples**

Byte Notation	BCh, Control Character			4Ah, Data Character		
Bit notation	<u>7654 3210</u> 1011 1100b	Control variable K		<u>7654 3210</u> 0100 1010b	Control variable D	
Unencoded bit notation	<u>HGF EDCBA</u> 101 11100b	<u>Z</u> K		<u>HGF EDCBA</u> 010 01010b	<u>Z</u> D	
Bit notation reordered to conform with Zxx.y convention	<u>Z</u> K	<u>EDCBA HGF</u> 11100 101b		<u>Z</u> D	<u>EDCBA HGF</u> 01010 010b	
Character name	K	28 .5		D	10 .2	

### 9.2.3 Character code

#### 9.2.3.1 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 Franszek (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.

#### 9.2.3.2 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.

#### 9.2.3.3 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 – 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.

#### **9.2.3.4 Data encoding**

Table 84 and Table 85 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 84 and Table 85 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 84 – 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		D19	10011b	110010b			
D4	00100b	001010b	110101b	D20	10100b	001011b			
D5	00101b	101001b		rd	D21	10101b	101010b		
D6	00110b	011001b			D22	10110b	011010b		
D7	00111b	000111b	111000b	-rd	D23	10111b	000101b	111010b	-rd
D8	01000b	000110b	111001b		D24	11000b	001100b	110011b	rd
D9	01001b	100101b		rd	D25	11001b	100110b		
D10	01010b	010101b			D26	11010b	010110b		
D11	01011b	110100b			D27	11011b	001001b	110110b	-rd
D12	01100b	001101b			D28	11100b	001110b		rd
D13	01101b	101100b		-rd	D29	11101b	010001b	101110b	
D14	01110b	011100b			D30	11110b	100001b	011110b	
D15	01111b	101000b	010111b		D31	11111b	010100b	101011b	

**Table 85 – 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	-rd
Dx.4	100b	0010b	1101b	
Dx.5	101b	1010b		rd
Dx.6	110b	0110b		
Dx.P7	111b	0001b	1110b	-rd
Dx.A7 <sup>a</sup>	111b	1000b	0111b	

<sup>a</sup> A7 replaces P7 if [(rd>0) and (e=i=0)] or [(rd<0) and (e=i=1)]

**9.2.3.5 Encoding examples**

The encoding examples in Table 86 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, D10 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 86 – 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	-

### 9.2.3.6 8b/10b valid encoded characters

#### 9.2.3.6.1 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 87, 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.



**9.2.3.6.2 Data characters**

**Table 87 – 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 87 – Valid data characters (continued)(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 87 – Valid data characters (continued)(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	A Eh	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 87 – Valid data characters (continued)(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

### 9.2.3.6.3 Control characters

**Table 88 – 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 <sub>P</sub>
K28.5	BCh	001111 1010b	110000 0101b	Occurs only at Byte 0 of ALIGN <sub>P</sub>

In Serial ATA only the K28.3 and K28.5 control characters as defined in Table 88 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<sub>P</sub>, while the K28.5 control character is used to prefix ALIGN<sub>P</sub>. 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<sub>P</sub> primitives are of neutral disparity, (i.e., the running disparity at the end of ALIGN<sub>P</sub> is the same as the running disparity at the beginning of ALIGN<sub>P</sub>).

## 9.2.4 Transmission order summary

### 9.2.4.1 Transmission order

#### 9.2.4.2 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 41 - Note that bit “i” is transmitted between bits “e” and “f”, and that bit “j” is transmitted last, and not in alphabetical order.

#### 9.2.4.3 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.

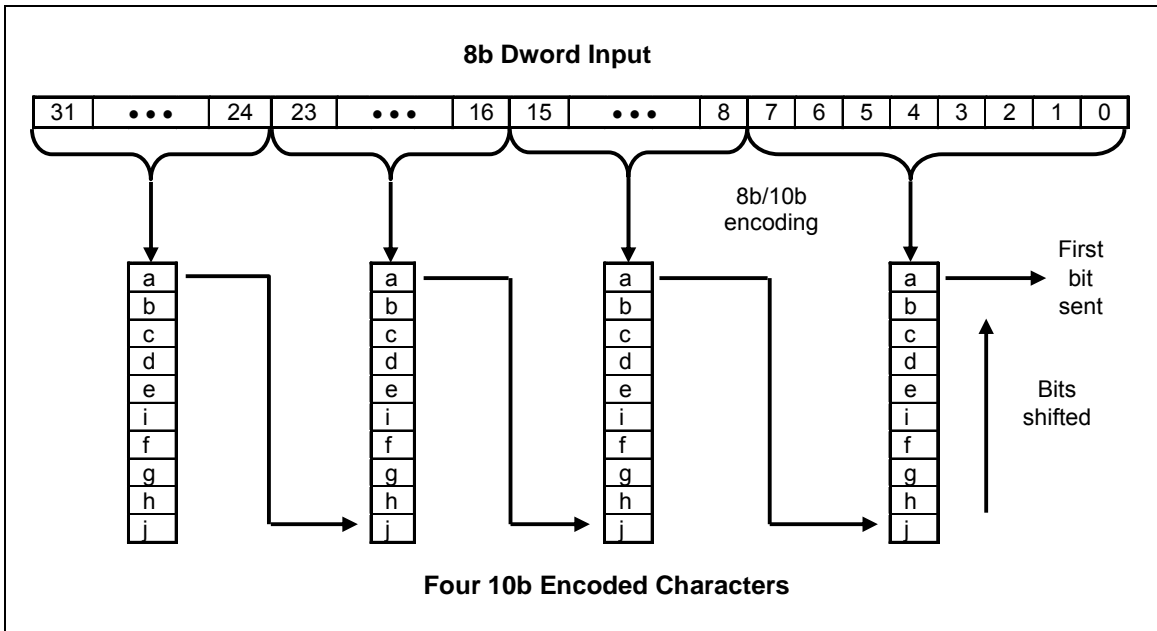


Figure 268 – Bit ordering and significance

#### 9.2.4.4 Dwords within a frame

A frame (see 10.4.1) shall be transmitted sequentially in ascending Dword order starting with the SOF<sub>P</sub> delimiter, followed by the Dwords of the frame contents, followed by the CRC, and ending with the EOF<sub>P</sub> delimiter.

NOTE 42 - 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 shall be preserved.

### 9.2.5 Reception summary

#### 9.2.5.1 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 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.

#### 9.2.5.2 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 89).

EXAMPLE 2 - The second shows a single character of propagation (see Table 89).

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 89 – 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 <sup>a</sup>	+	010101 0101b	+	111010 <sup>b</sup> 1010b	+
Decoded character stream.	-	D21.0	+	D10.2	+	Code violation <sup>b</sup>	+ <sup>c</sup>
<sup>a</sup> Bit error introduced, 1001b changes to 1011b <sup>b</sup> 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. <sup>c</sup> Running disparity shall be computed on the received character regardless of the validity of the encoded character.							

**Table 90 – 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 <sup>a</sup>	+	111010 <sup>b</sup> 0010b	-	111010 1010b	+
Decoded character stream	-	D21.0	+	Code violation <sup>b</sup>	-	D23.5	+ <sup>c</sup>
<sup>a</sup> Bit error introduced, 1001b changes to 1011b <sup>b</sup> 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. <sup>c</sup> Running disparity shall be computed on the received character regardless of the validity of the encoded character.							

### 9.3 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.

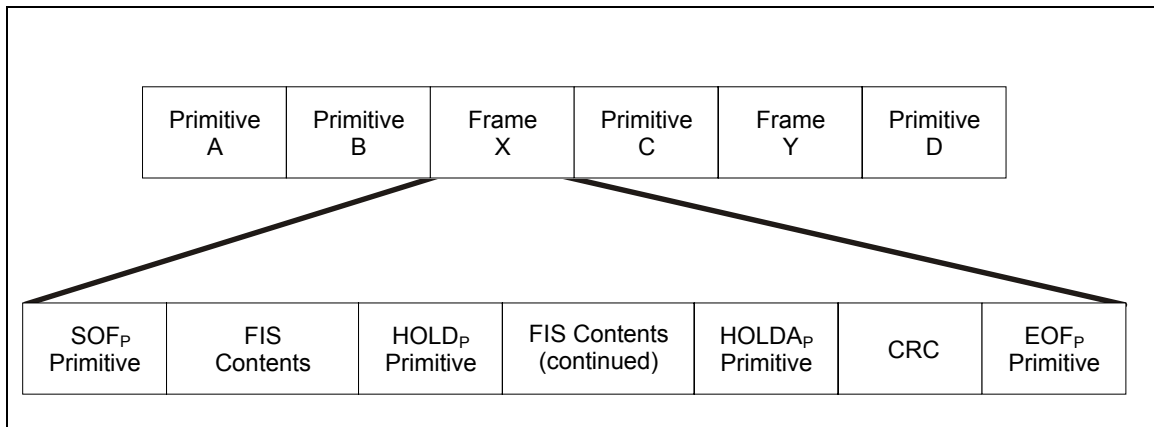


Figure 269 – Transmission structures



## 9.4 Primitives

### 9.4.1 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<sub>P</sub> that begins with the K28.5 control character. ALIGN<sub>P</sub> thus represents the only primitive that contains the comma character. Following the control character, three additional characters are encoded to complete the Dword. Table 91 is a summary of the character combinations that make up each primitive.

### 9.4.2 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<sub>P</sub> 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<sub>P</sub> is the same as the ending disparity of the last character transmitted before ALIGN<sub>P</sub>.

Each primitive is described and the encoding defined in the following sections.

### 9.4.3 Primitive handshakes

Some primitives are transmitted in response to receipt of other primitives to acknowledge receipt.

EXAMPLE 1 - HOLDA<sub>P</sub> is transmitted in response to the receipt of HOLD<sub>P</sub> primitives and R\_OK<sub>P</sub> or R\_ERR<sub>P</sub> is transmitted in response to WTRM<sub>P</sub>.

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<sub>P</sub> primitives but receive six HOLDA<sub>P</sub> 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<sub>P</sub> may be sent).

### 9.4.4 Primitive descriptions

The following table (see Table 91) contains the primitive mnemonics and a brief description of each.

**Table 91 – Description of primitives**

<b>Primitive</b>	<b>Name</b>	<b>Description</b>
ALIGN <sub>P</sub>	Phy layer control.	Upon receipt of an ALIGN <sub>P</sub> , 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 <sub>P</sub> primitives shall be sent (except for retimed loopback).
CONT <sub>P</sub>	Continue repeating previous primitive.	CONT <sub>P</sub> allows long strings of repeated primitives to be eliminated. CONT <sub>P</sub> implies that the previously received primitive be repeated as long as another primitive is not received.
DMAT <sub>P</sub>	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 <sub>P</sub> . The transmitter context is assumed to remain stable after EOF <sub>P</sub> has been sent.
EOF <sub>P</sub>	End of frame.	EOF <sub>P</sub> marks the end of a frame. The previous non-primitive Dword is the CRC for the frame.
HOLD <sub>P</sub>	Hold data transmission.	HOLD <sub>P</sub> is transmitted in place of payload data within a frame while the transmitter does not have the next payload data ready for transmission. HOLD <sub>P</sub> is also transmitted by the receiver while a receiver is not ready to receive additional payload data.
HOLDA <sub>P</sub>	Hold acknowledge.	This primitive is sent while HOLD <sub>P</sub> is received.
PMACK <sub>P</sub>	Power management acknowledge.	Sent in response to a PMREQ_S <sub>P</sub> or PMREQ_P <sub>P</sub> if a receiving node is prepared to enter a power mode state.
PMNAK <sub>P</sub>	Power management denial.	Sent in response to a PMREQ_S <sub>P</sub> or PMREQ_P <sub>P</sub> if a receiving node is not prepared to enter a power mode state or if power management is not supported.
PMREQ_P <sub>P</sub>	Power management request to Partial.	This primitive is sent continuously until PMACK <sub>P</sub> or PMNAK <sub>P</sub> is received. If PMACK <sub>P</sub> is received, the current node (host or device) stops transmitting PMREQ_P <sub>P</sub> and enters the Partial power management state.
PMREQ_S <sub>P</sub>	Power management request to Slumber	This primitive is sent continuously until PMACK <sub>P</sub> or PMNAK <sub>P</sub> is received. If PMACK <sub>P</sub> is received, the current node (host or device) stops transmitting PMREQ_S <sub>P</sub> and enters the Slumber power management state.
R_ERR <sub>P</sub>	Reception error.	Current node (host or device) detected error in received payload.
R_IP <sub>P</sub>	Reception in Progress.	Current node (host or device) is receiving payload.
R_OK <sub>P</sub>	Reception with no error.	Current node (host or device) detected no error in received payload.
R_RDY <sub>P</sub>	Receiver ready.	Current node (host or device) is ready to receive payload.
SOF <sub>P</sub>	Start of frame.	Start of a frame. Payload and CRC follow until EOF <sub>P</sub> .
SYNC <sub>P</sub>	Synchronization	Synchronizing primitive.
WTRM <sub>P</sub>	Wait for frame termination	After transmission of an EOF <sub>P</sub> , the transmitter sends WTRM <sub>P</sub> while waiting for reception status from receiver.
X_RDY <sub>P</sub>	Transmission data ready.	Current node (host or device) has payload ready for transmission.

### 9.4.5 Primitive encoding

Table 92 defines the encoding for each primitive.

**Table 92 – Primitive encoding**

Primitive Name	Byte 3 Contents	Byte 2 Contents	Byte 1 Contents	Byte 0 Contents
ALIGN <sub>P</sub>	D27.3	D10.2	D10.2	K28.5
CONT <sub>P</sub>	D25.4	D25.4	D10.5	K28.3
DMAT <sub>P</sub>	D22.1	D22.1	D21.5	K28.3
EOF <sub>P</sub>	D21.6	D21.6	D21.5	K28.3
HOLD <sub>P</sub>	D21.6	D21.6	D10.5	K28.3
HOLDA <sub>P</sub>	D21.4	D21.4	D10.5	K28.3
PMACK <sub>P</sub>	D21.4	D21.4	D21.4	K28.3
PMNAK <sub>P</sub>	D21.7	D21.7	D21.4	K28.3
PMREQ_P <sub>P</sub>	D23.0	D23.0	D21.5	K28.3
PMREQ_S <sub>P</sub>	D21.3	D21.3	D21.4	K28.3
R_ERR <sub>P</sub>	D22.2	D22.2	D21.5	K28.3
R_IP <sub>P</sub>	D21.2	D21.2	D21.5	K28.3
R_OK <sub>P</sub>	D21.1	D21.1	D21.5	K28.3
R_RDY <sub>P</sub>	D10.2	D10.2	D21.4	K28.3
SOF <sub>P</sub>	D23.1	D23.1	D21.5	K28.3
SYNC <sub>P</sub>	D21.5	D21.5	D21.4	K28.3
WTRM <sub>P</sub>	D24.2	D24.2	D21.5	K28.3
X_RDY <sub>P</sub>	D23.2	D23.2	D21.5	K28.3

### 9.4.6 DMAT<sub>P</sub> primitive

No consistent use of the DMAT<sub>P</sub> facility is defined, and its use may impact software compatibility. Implementations should tolerate reception of DMAT<sub>P</sub> as defined in this specification but should avoid transmission of DMAT<sub>P</sub> in order to minimize potential interaction problems. One valid response to reception of DMAT<sub>P</sub> is to treat the DMAT<sub>P</sub> as R\_IP<sub>P</sub> 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<sub>P</sub>, 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<sub>P</sub>, has to accept all data until receiving an EOF<sub>P</sub> 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<sub>P</sub>) 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<sub>P</sub> signal shall cause the recipient to close the current frame by inserting the CRC and EOF<sub>P</sub>, and return to the idle state.

For host to device data transfers, upon receiving the DMAT<sub>P</sub> signal the host shall terminate the transfer in progress by deactivating its DMA engine and closing the frame with valid CRC and EOF<sub>P</sub>. 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.

## 9.4.7 $CONT_P$ primitive

### 9.4.7.1 $CONT_P$ primitive overview

In order to accommodate EMI reductions, scrambling of data is incorporated in Serial ATA as defined in 9.5. 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 9.4.7.2. 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).

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

The following primitives may be followed by a CONT<sub>P</sub>:

- a) HOLD<sub>P</sub>;
- b) HOLDA<sub>P</sub>;
- c) PMREQ\_P<sub>P</sub>;
- d) PMREQ\_S<sub>P</sub>;
- e) R\_ERR<sub>P</sub>;
- f) R\_IP<sub>P</sub>;
- g) R\_OK<sub>P</sub>;
- h) R\_RDY<sub>P</sub>;
- i) SYNC<sub>P</sub>;
- j) WTRM<sub>P</sub>; and
- k) X\_RDY<sub>P</sub>.

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 9.6.2 and 9.6.5), the use of CONT<sub>P</sub> 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<sub>P</sub> primitives have been transmitted or until receipt of a primitive other than SYNC<sub>P</sub> or ALIGN<sub>P</sub> has been detected.

Table 93 illustrates use of CONT<sub>P</sub> in the transmission of a FIS.

**Table 93 – CONT<sub>P</sub> usage example**

(part 1 of 2)

Transmitter	Receiver
XXXX	XXXX
XXXX	XXXX
X_RDY <sub>P</sub>	XXXX
X_RDY <sub>P</sub>	XXXX
CONT <sub>P</sub>	XXXX
XXXX	XXXX
XXXX	R_RDY <sub>P</sub>
XXXX	R_RDY <sub>P</sub>
XXXX	CONT <sub>P</sub>
SOF <sub>P</sub>	XXXX
DATA (FIS Type)	XXXX
DATA	XXXX
DATA	R_IP <sub>P</sub>
DATA	R_IP <sub>P</sub>
DATA	CONT <sub>P</sub>
HOLD <sub>P</sub>	XXXX
HOLD <sub>P</sub>	XXXX
CONT <sub>P</sub>	XXXX
XXXX	XXXX
XXXX	HOLDA <sub>P</sub>
XXXX	HOLDA <sub>P</sub>

**Table 93 – CONTP usage example (part 2 of 2)**

<b>Transmitter</b>	<b>Receiver</b>
HOLD <sub>P</sub>	CONT <sub>P</sub>
DATA	XXXX
DATA	XXXX
DATA	XXXX
CRC	XXXX
EOF <sub>P</sub>	R_IP <sub>P</sub>
WTRM <sub>P</sub>	R_IP <sub>P</sub>
WTRM <sub>P</sub>	CONT <sub>P</sub>
WTRM <sub>P</sub>	XXXX
CONT <sub>P</sub>	XXXX
XXXX	R_OK <sub>P</sub>
XXXX	R_OK <sub>P</sub>
XXXX	CONT <sub>P</sub>
XXXX	XXXX
SYNC <sub>P</sub>	XXXX
SYNC <sub>P</sub>	XXXX
CONT <sub>P</sub>	XXXX
XXXX	XXXX
XXXX	XXXX
XXXX	SYNC <sub>P</sub>
XXXX	SYNC <sub>P</sub>
XXXX	CONT <sub>P</sub>
XXXX	XXXX
XXXX	XXXX

Key:  
 XXXX = Scrambled data values (non-primitives)  
 DATA = FIS payload data.  
 Subscript p = primitive

**9.4.7.2 Scrambling of data following the CONT<sub>P</sub> primitive**

The data following the CONT<sub>P</sub> shall be the output of an LFSR that implements the same polynomial as is used to scramble FIS contents. That polynomial as defined in 9.5.

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<sub>P</sub> shall be reset to the initial value upon detection of a COMINIT or COMRESET event.

Since the data following CONT<sub>P</sub> is discarded by the Link layer, the value of the LFSR is undefined between CONT<sub>P</sub> primitives (i.e., the LFSR result used for CONT<sub>P</sub> sequence N is not required to be continuous from the last LFSR result of CONT<sub>P</sub> 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 7.7 for additional information on scrambling and repeated primitive suppression.

### 9.4.7.3 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.

### 9.4.8 $ALIGN_P$ 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.

### 9.4.9 Flow control signaling latency

#### 9.4.9.1 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 24 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 94 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 94 – Example of components of a round trip delay**

<b>Receiver Sends HOLD<sub>p</sub></b>	
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 <sub>p</sub> .
1 Dword	Link acts on notification and inserts HOLDA <sub>p</sub> 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
<b>HOLDA<sub>p</sub> on the Cable</b>	

**9.4.9.2 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 Gbps 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.

#### 9.4.10 Examples of primitive usage (informative)

Table 95, Table 96, and Table 97 are examples that illustrate basic primitive usage. They do not show detailed lengthy sequences that invoke the use of  $CONT_P$ .

**Table 95 – 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 96 – Command shadow register block register transmission example**

Host	Device	Description
SYNC <sub>P</sub>	SYNC <sub>P</sub>	Idle condition.
SYNC <sub>P</sub>	SYNC <sub>P</sub>	Idle condition.
X_RDY <sub>P</sub>	SYNC <sub>P</sub>	Host ready to send Shadow register block registers.
X_RDY <sub>P</sub>	SYNC <sub>P</sub>	Device decodes X_RDY <sub>P</sub> .
X_RDY <sub>P</sub>	R_RDY <sub>P</sub>	Device indicates ready to receive.
X_RDY <sub>P</sub>	R_RDY <sub>P</sub>	Host decodes R_RDY <sub>P</sub> .
SOF <sub>P</sub>	R_RDY <sub>P</sub>	Host starts a frame.
DATA 0	R_RDY <sub>P</sub>	Host sends Register Host to Device FIS Dword 0 / device decodes SOF <sub>P</sub> .
DATA 1	R_IP <sub>P</sub>	Host sends Register Host to Device FIS Dword 1 / device stores DATA Dword 0.
...	...	...
DATA n	R_IP <sub>P</sub>	Host sends Register Host to Device FIS Dword n / device stores DATA Dword n-1.
CRC	R_IP <sub>P</sub>	Host sends CRC / device stores DATA Dword n.
EOF <sub>P</sub>	R_IP <sub>P</sub>	Host sends EOF <sub>P</sub> / device stores CRC.
WTRM <sub>P</sub>	R_IP <sub>P</sub>	Device decodes EOF <sub>P</sub> .
WTRM <sub>P</sub>	R_IP <sub>P</sub>	Device computes good CRC and releases TF contents.
WTRM <sub>P</sub>	R_OK <sub>P</sub>	Device sends good end.
WTRM <sub>P</sub>	R_OK <sub>P</sub>	Host decodes R_OK <sub>P</sub> as good results.
SYNC <sub>P</sub>	R_OK <sub>P</sub>	Host releases interface.
SYNC <sub>P</sub>	R_OK <sub>P</sub>	Device decodes release by host - is allowed to release.
SYNC <sub>P</sub>	SYNC <sub>P</sub>	Idle condition.

**Table 97 – Data from host to device transmission example**

<b>Host</b>	<b>Device</b>	<b>Description</b>
SYNC <sub>P</sub>	SYNC <sub>P</sub>	Idle condition.
SYNC <sub>P</sub>	SYNC <sub>P</sub>	Idle condition.
X_RDY <sub>P</sub>	SYNC <sub>P</sub>	Host ready to send Shadow register block registers.
X_RDY <sub>P</sub>	SYNC <sub>P</sub>	Device decodes X_RDY <sub>P</sub> .
X_RDY <sub>P</sub>	R_RDY <sub>P</sub>	Device indicates ready to receive.
X_RDY <sub>P</sub>	R_RDY <sub>P</sub>	Host decodes R_RDY <sub>P</sub> .
SOF <sub>P</sub>	R_RDY <sub>P</sub>	Host starts a frame.
DATA 0	R_RDY <sub>P</sub>	Host sends DATA Dword 0 / device decodes SOF <sub>P</sub> .
DATA 1	R_IP <sub>P</sub>	Host sends DATA Dword 1 / device stores DATA Dword 0.
...	..	...
DATA x	R_IP <sub>P</sub>	Host sends DATA Dword x / device stores DATA Dword (x-1) .
HOLD <sub>P</sub>	R_IP <sub>P</sub>	Host sends HOLD <sub>P</sub> / device stores DATA Dword (x) and decodes HOLD <sub>P</sub> .
HOLD <sub>P</sub>	HOLDA <sub>P</sub>	Device acknowledges HOLD <sub>P</sub> .
HOLD <sub>P</sub>	HOLDA <sub>P</sub>	Host decodes HOLDA <sub>P</sub> – host may release HOLD <sub>P</sub> at any time.
DATA(n-2)	HOLDA <sub>P</sub>	Host sends (n-2)th DATA Dword / device decodes DATA Dword.
DATA(n-1)	R_IP <sub>P</sub>	Host sends (n-1)th data Dword / device stores (n-2)th DATA Dword.
DATA(n)	R_IP <sub>P</sub>	Host sends nth data Dword / device stores (n-1)th DATA Dword.
CRC	R_IP <sub>P</sub>	Host sends CRC / device stores nth DATA Dword.
EOF <sub>P</sub>	R_IP <sub>P</sub>	Host sends EOF <sub>P</sub> / device stores CRC.
WTRM <sub>P</sub>	R_IP <sub>P</sub>	Device decodes EOF <sub>P</sub> .
WTRM <sub>P</sub>	R_IP <sub>P</sub>	Device computes good CRC and releases data contents.
WTRM <sub>P</sub>	R_OK <sub>P</sub>	Device sends good end.
WTRM <sub>P</sub>	R_OK <sub>P</sub>	Host decodes R_OK <sub>P</sub> as good results.
SYNC <sub>P</sub>	R_OK <sub>P</sub>	Host releases interface.
SYNC <sub>P</sub>	R_OK <sub>P</sub>	Device decodes release by host - is allowed to release.
SYNC <sub>P</sub>	SYNC <sub>P</sub>	Idle condition.

## 9.5 CRC and scrambling

### 9.5.1 CRC and scrambling overview

The CRC (Cyclic Redundancy Check) of a frame is a Dword (32 bit) field that shall follow the last Dword of the contents of a FIS and precede EOF<sub>P</sub>. The CRC calculation covers all of the FIS transport data between the SOF<sub>P</sub> and EOF<sub>P</sub> primitives, and excludes any intervening primitives and CONT<sub>P</sub> 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<sub>P</sub> to EOF<sub>P</sub> 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<sub>P</sub> is transmitted. All data Words between the SOF<sub>P</sub> and EOF<sub>P</sub> shall be scrambled, including the CRC.

### 9.5.2 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<sub>P</sub> and EOF<sub>P</sub>), 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<sub>P</sub>, 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<sub>P</sub> primitive). Since it is possible for a repeated primitive stream to occur in the middle of a data frame – multiple HOLD<sub>P</sub>/HOLDA<sub>P</sub> 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.

### 9.5.3 Relationship between scrambling and CRC

The order of application of scrambling shall be as follows. For a Dword of data following SOF<sub>P</sub> 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.

#### **9.5.4 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<sub>p</sub>/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.

## **9.6 Link layer state machine**

### **9.6.1 Terms used in link layer transition tables**

Terms used in link layer transition tables are:

- a) LRESET, link layer COMRESET or COMINIT signal;
- b) PHYRDY<sub>n</sub>, 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.

### 9.6.2 Link idle state machine

The link idle state machine is defined in Figure 270.

L1: L_IDLE <sup>d</sup>		Transmit SYNC <sub>P</sub> .	
1.	Transport layer requests frame transmission and PHYRDY <sup>b</sup> .	→	HL_SendChkRdy or DL_SendChkRdy <sup>a</sup>
2.	Transport layer requests transition to Partial and PHYRDY <sup>b,e</sup> .	→	L_TPMPartial
3.	Transport layer requests transition to Slumber and PHYRDY <sup>b,e</sup> .	→	L_TPMSlumber
4.	X_RDY <sub>P</sub> received from Phy.	→	L_RcvWaitFifo
5.	Phy layer forwards ( PMREQ_P <sub>P</sub> or PMREQ_S <sub>P</sub> ) and power modes are enabled and acceptable.	→	L_PMOff
6.	Phy layer forwards ( PMREQ_P <sub>P</sub> or PMREQ_S <sub>P</sub> ) and power modes are disabled or are unacceptable.	→	L_PMDeny
7.	Phy layer forwards AnyDword other than (X_RDY <sub>P</sub> or PMREQ_P <sub>P</sub> or PMREQ_S <sub>P</sub> ) and no transmit request from Transport layer <sup>b,c</sup> .	→	L_IDLE
8.	PHYRDYn	→	L_NoCommErr
<sup>a</sup> The host Link layer makes a transition to the HL_SendChkRdy state; the device Link layer makes a transition to the DL_SendChkRdy state. <sup>b</sup> This transition is taken even if errors (i.e., 10b/8b decoding errors) are detected. <sup>c</sup> This statement also ignores any unrecognized sequences or commands not defined in this specification. <sup>d</sup> Upon entry to this state from the LS3:L_SendAlign state or the LPM8:L_WakeUp2 state, use of CONT <sub>P</sub> is not allowed until either a minimum of 10 non-ALIGN <sub>P</sub> primitives have been transmitted or until receipt of a primitive other than SYNC <sub>P</sub> or ALIGN <sub>P</sub> has been detected. <sup>e</sup> 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)

<b>L2: L_SyncEscape<sup>a</sup></b>	Transmit SYNC <sub>P</sub> .	
1. AnyDword other than X_RDY <sub>P</sub> or SYNC <sub>P</sub> received from Phy.	→	L_SyncEscape
2. X_RDY <sub>P</sub> or SYNC <sub>P</sub> received from Phy.	→	L_IDLE
3. PHYRDY <sub>n</sub>	→	L_NoCommErr <sup>b</sup>
<sup>a</sup> This state is entered asynchronously from any other Link layer state where the Link layer has transmitted SYNC <sub>P</sub> to escape a FIS transfer, also known as a SYNC Escape. <sup>b</sup> The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		
<b>LS1: L_NoCommErr</b>	Post Phy not ready error to Transport layer.	
1. Unconditional	→	L_NoComm
<b>LS2: L_NoComm</b>	Transmit ALIGN <sub>P</sub> <sup>a</sup> .	
1. PHYRDY <sub>n</sub>	→	L_NoComm
2. PHYRDY	→	L_SendAlign
<sup>a</sup> Also deactivate any signal for Phy layer to abort operation.		
<b>LS3: L_SendAlign</b>	Transmit ALIGN <sub>P</sub> .	
1. PHYRDY <sub>n</sub>	→	L_NoCommErr
2. PHYRDY	→	L_IDLE
<b>LS4: L_RESET<sup>a</sup></b>	Reset Link state to initial conditions.	
1. LRESET Link reset signal asserted.	→	L_RESET
2. LRESET Link reset signal negated.	→	L_NoComm
<sup>a</sup> 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<sub>P</sub> and waits for X\_RDY<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> or PMREQ\_S<sub>P</sub> 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<sub>P</sub> or a PMREQ\_S<sub>P</sub> 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<sub>P</sub> from the Phy layer or does not receive a PMREQ\_x<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> and waits for a SYNC<sub>P</sub> from the Phy layer before proceeding to L\_IDLE. The Link layer also transitions to L\_IDLE if X\_RDY<sub>P</sub> 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<sub>P</sub> or SYNC<sub>P</sub>, the Link layer shall make a transition to the L2: L\_SyncEscape state.

**Transition L2:2**, if the Link layer receives X\_RDY<sub>P</sub> or SYNC<sub>P</sub> 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<sub>p</sub> 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.

### 9.6.3 Link transmit state machine

The link transmitter state machine is defined in Figure 271.

<b>LT1: HL_SendChkRdy</b>	Transmit X_RDY <sub>P</sub> .	
1. R_RDY <sub>P</sub> received from Phy.	→	L_SendSOF
2. X_RDY <sub>P</sub> received from Phy.	→	L_RcvWaitFifo
3. AnyDword other than (R_RDY <sub>P</sub> or X_RDY <sub>P</sub> ) <sup>a</sup> received from Phy layer.	→	HL_SendChkRdy
4. PHYRDY <sub>n</sub>	→	L_NoCommErr <sup>b</sup>
<sup>a</sup> Any received errors (e.g., 10b/8b decoding errors and invalid primitives) are ignored. <sup>b</sup> The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		

<b>LT2: DL_SendChkRdy</b>	Transmit X_RDY <sub>P</sub> .	
1. R_RDY <sub>P</sub> received from Phy.	→	L_SendSOF
2. AnyDword other than R_RDY <sub>P</sub> received from Phy.	→	DL_SendChkRdy
3. PHYRDY <sub>n</sub>	→	L_NoCommErr

<b>LT3: L_SendSOF</b>	Transmit SOF <sub>P</sub>	
1. PHYRDY <sup>a</sup>	→	L_SendData
2. PHYRDY <sub>n</sub>	→	L_NoCommErr <sup>b</sup>
3. SYNC <sub>P</sub> received from Phy.	→	L_IDLE <sup>b</sup>
<sup>a</sup> Any received errors (e.g., 10b/8b decoding errors and invalid primitives) are ignored. <sup>b</sup> 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 <sub>P</sub> or DMAT <sub>P</sub> or SYNC <sub>P</sub> ) received from Phy <sup>a,b</sup> .	→	L_SendData
2. More data to transmit and HOLD <sub>P</sub> received from Phy.	→	L_RcvrHold
3. Data transmit not complete and data not ready to transmit and AnyDword other than SYNC <sub>P</sub> received from Phy.	→	L_SendHold
4. DMAT <sub>P</sub> received from Phy or data transmit complete and AnyDword other than SYNC <sub>P</sub> received from Phy.	→	L_SendCRC <sup>e</sup>
5. SYNC <sub>P</sub> received from Phy.	→	L_IDLE <sup>c</sup>
6. PHYRDYn	→	L_NoCommErr <sup>c</sup>
7. Transport layer indicates request to escape current frame <sup>d</sup> .	→	L_SyncEscape
<p><sup>a</sup> Any received errors (e.g., 10b/8b decoding errors and invalid primitives) are ignored.</p> <p><sup>b</sup> This makes possible a back channel during this time.</p> <p><sup>c</sup> The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.</p> <p><sup>d</sup> If this condition is true, the associated transition has priority over all other transitions exiting this state.</p> <p><sup>e</sup> The DMAT<sub>P</sub> 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<sub>P</sub> is received.</p>		

**Figure 271 – Link layer transmit state machine (part 2 of 4)**

<b>LT5: L_RcvrHold</b>		Transmit HOLD <sub>P</sub> .	
1.	More data to transmit and AnyDword other than (HOLD <sub>P</sub> or SYNC <sub>P</sub> or DMAT <sub>P</sub> ) received from Phy with no DecErr.	→	L_SendData
2.	More data to transmit and HOLD <sub>P</sub> received from Phy or DecErr.	→	L_RcvrHold
3.	More data to transmit and SYNC <sub>P</sub> received from Phy.	→	L_IDLE <sup>a</sup>
4.	More data to transmit and DMAT <sub>P</sub> received from Phy.	→	L_SendCRC <sup>c</sup>
5.	PHYRDYn	→	L_NoCommErr <sup>a</sup>
6.	Transport layer indicates request to escape current frame <sup>b</sup> .	→	L_SyncEscape
7.	SYNC <sub>P</sub> received from Phy.	→	L_IDLE <sup>a</sup>
<sup>a</sup> The Link layer shall notify the Transport layer of the condition and fail the attempted transfer. <sup>b</sup> If this condition is true, the associated transition has priority over all other transitions exiting this state. <sup>c</sup> The DMAT <sub>P</sub> 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 <sub>P</sub> is received.			

<b>LT6: L_SendHold</b>		Transmit HOLD <sub>P</sub> .	
1.	More data ready to transmit and AnyDword other than (HOLD <sub>P</sub> or SYNC <sub>P</sub> ) received from Phy.	→	L_SendData
2.	More data ready to transmit and HOLD <sub>P</sub> received from Phy.	→	L_RcvrHold
3.	Data transmit not complete and data not ready to transmit and AnyDword other than (SYNC <sub>P</sub> or DMAT <sub>P</sub> ) received from Phy.	→	L_SendHold
4.	DMAT <sub>P</sub> received from Phy or data transmit complete and AnyDword other than SYNC <sub>P</sub> received from Phy.	→	L_SendCRC <sup>c</sup>
5.	SYNC <sub>P</sub> received from Phy.	→	L_IDLE <sup>a</sup>
6.	PHYRDYn	→	L_NoCommErr <sup>a</sup>
7.	Transport layer indicates request to escape current frame <sup>b</sup> .	→	L_SyncEscape
<sup>a</sup> The Link layer shall notify the Transport layer of the condition and fail the attempted transfer. <sup>b</sup> If this condition is true, the associated transition has priority over all other transitions exiting this state. <sup>c</sup> The DMAT <sub>P</sub> 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 <sub>P</sub> is received.			

**Figure 271 – Link layer transmit state machine (part 3 of 4)**

<b>LT7: L_SendCRC</b>	Transmit CRC.		
1. PHYRDY and SYNC <sub>P</sub> not received from Phy.	→	L_SendEOF	
2. PHYRDY <sub>n</sub>	→	L_NoCommErr <sup>a</sup>	
3. PHYRDY and SYNC <sub>P</sub> received from Phy.	→	L_IDLE <sup>a</sup>	
<sup>a</sup> The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.			

<b>LT8: L_SendEOF</b>	Transmit EOF <sub>P</sub> .		
1. PHYRDY and SYNC <sub>P</sub> not received from Phy.	→	L_Wait	
2. PHYRDY <sub>n</sub>	→	L_NoCommErr <sup>a</sup>	
3. PHYRDY and SYNC <sub>P</sub> received from Phy.	→	L_IDLE <sup>a</sup>	
<sup>a</sup> The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.			

<b>LT9: L_Wait</b>	Transmit WTRM <sub>P</sub> .		
1. R_OK <sub>P</sub> received from Phy.	→	L_IDLE (good status)	
2. R_ERR <sub>P</sub> received from Phy.	→	L_IDLE (bad status)	
3. SYNC <sub>P</sub> received from Phy.	→	L_IDLE <sup>a</sup>	
4. AnyDword other than (R_OK <sub>P</sub> or R_ERR <sub>P</sub> or SYNC <sub>P</sub> ) received from Phy.	→	L_Wait	
5. PHYRDY <sub>n</sub>	→	L_NoCommErr <sup>a</sup>	
<sup>a</sup> 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<sub>P</sub> and waits for X\_RDY<sub>P</sub> or R\_RDY<sub>P</sub> from the Phy layer.

NOTE 43 - It is possible that both the host and the device simultaneously request frame transmission by transmitting X\_RDY<sub>P</sub>. If the host receives X\_RDY<sub>P</sub> while transmitting X\_RDY<sub>P</sub>, 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> or X\_RDY<sub>P</sub> 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<sub>P</sub> and waits for R\_RDY<sub>P</sub> from the Phy layer.

**Transition LT2:1**, if the device Link layer receives R\_RDY<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> has been received from the Phy layer.

If in this state, the Link layer transmits SOF<sub>P</sub>.

**Transition LT3:1**, if the device Link layer has transmitted SOF<sub>P</sub>, 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub>, DMAT<sub>P</sub>, or SYNC<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> is received.

**Transition LT4:2**, if the device Link layer receives HOLD<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> from the Phy layer, it shall notify the Transport layer and terminate the transmission in progress as defined in 9.4.6 and shall transition to the LT7: L\_SendCRC state.

**Transition LT4:5**, if the Link layer receives SYNC<sub>P</sub> 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<sub>P</sub> has been received from the Phy layer.

If in this state, the Link layer shall transmit HOLD<sub>A</sub><sub>P</sub>.

**Transition LT5:1**, if the Link layer receives any Dword other than a HOLD<sub>P</sub>, SYNC<sub>P</sub>, or a DMAT<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> from the Phy layer, it shall notify the Transport layer and terminate the transmission in progress as defined in 9.4.6 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<sub>P</sub> 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<sub>P</sub> has not been received from the Phy layer.

If in this state, the Link layer shall transmit HOLD<sub>P</sub>.

**Transition LT6:1**, if the Link layer receives any Dword other than a HOLD<sub>P</sub> or SYNC<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> from the Phy layer, it shall notify the Transport layer and terminate the transmission in progress as defined in 9.4.6 and shall transition to the LT7:L\_SendCRC state.

**Transition LT6:5**, if the Link layer receives SYNC<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub>.

**Transition LT8:1**, if EOF<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> has been transmitted.

If in this state, the Link layer shall transmit WTRM<sub>P</sub>.

**Transition LT9:1**, if the Link layer receives R\_OK<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub>, R\_ERR<sub>P</sub>, or SYNC<sub>P</sub> 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.

#### 9.6.4 Link receive state machine

The link receive state machine is defined in Figure 272.

<b>LR1: L_RcvChkRdy</b>	Transmit R_RDY <sub>P</sub> .		
1. X_RDY <sub>P</sub> received from Phy.	→	L_RcvChkRdy	
2. SOF <sub>P</sub> received from Phy.	→	L_RcvData	
3. Any Dword other than (X_RDY <sub>P</sub> or SOF <sub>P</sub> ) received from Phy.	→	L_IDLE	
4. PHYRDYn	→	L_NoCommErr <sup>a</sup>	
<sup>a</sup> The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.			
<b>LR2: L_RcvWaitFifo</b>	Transmit SYNC <sub>P</sub> .		
1. X_RDY <sub>P</sub> received from Phy and FIFO space available.	→	L_RcvChkRdy	
2. X_RDY <sub>P</sub> received from Phy and FIFO space not available.	→	L_RcvWaitFifo	
3. Any Dword other than X_RDY <sub>P</sub> received from Phy.	→	L_IDLE	
4. PHYRDYn	→	L_NoCommErr <sup>a</sup>	
<sup>a</sup> 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)

<b>LR3: L_RcvData</b>	Transmit R_IP <sub>P</sub> or DMAT <sub>P</sub> <sup>a</sup> .	
1. (DatDword received from Phy and FIFO space) or HOLD <sub>P</sub> received from Phy.	→	L_RcvData
2. DatDword received from Phy and insufficient FIFO space.	→	L_Hold
3. HOLD <sub>P</sub> received from Phy.	→	L_RcvHold
4. EOF <sub>P</sub> received from Phy.	→	L_RcvEOF
5. WTRM <sub>P</sub> received from Phy.	→	L_BadEnd
6. SYNC <sub>P</sub> received from Phy.	→	L_IDLE
7. AnyDword other than (HOLD <sub>P</sub> or EOF <sub>P</sub> or HOLD <sub>A</sub> <sub>P</sub> or SYNC <sub>P</sub> or WTRM <sub>P</sub> ) received from Phy.	→	L_RcvData
8. PHYRDY <sub>n</sub>	→	L_NoCommErr <sup>b</sup>
9. Transport layer indicates request to escape current frame.	→	L_SyncEscape
<sup>a</sup> If the Transport layer signals that it wishes to terminate the transfer, DMAT <sub>P</sub> is transmitted in place of R_IP <sub>P</sub> . <sup>b</sup> The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.		

<b>LR4: L_Hold</b>	Transmit HOLD <sub>P</sub> .	
1. FIFO space available and AnyDword other than HOLD <sub>P</sub> or EOF <sub>P</sub> received from Phy.	→	L_RcvData
2. FIFO space available and HOLD <sub>P</sub> received from Phy.	→	L_RcvHold
3. EOF <sub>P</sub> received from Phy.	→	L_RcvEOF
4. No FIFO space available and EOF <sub>P</sub> not received from Phy and SYNC <sub>P</sub> not received from Phy and PHYRDY	→	L_Hold
5. PHYRDY <sub>n</sub>	→	L_NoCommErr <sup>a</sup>
6. SYNC <sub>P</sub> received from Phy.	→	L_IDLE
7. Transport layer indicates request to escape current frame.	→	L_SyncEscape
<sup>a</sup> 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)**

<b>LR5: L_RcvHold</b>	Transmit HOLD <sub>P</sub> or DMAT <sub>P</sub> <sup>a</sup> .		
1. AnyDword other than (HOLD <sub>P</sub> or EOF <sub>P</sub> or SYNC <sub>P</sub> ) received from Phy.	→	L_RcvData	
2. HOLD <sub>P</sub> received from Phy.	→	L_RcvHold	
3. EOF <sub>P</sub> received from Phy.	→	L_RcvEOF	
4. SYNC <sub>P</sub> received from Phy.	→	L_IDLE	
5. PHYRDY <sub>n</sub>	→	L_NoCommErr <sup>b</sup>	
6. Transport layer indicates request to escape current frame.	→	L_SyncEscape	
<sup>a</sup> If the Transport layer signals that it wishes to terminate the transfer, DMAT <sub>P</sub> is transmitted in place of HOLD <sub>P</sub> . <sup>b</sup> The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.			

<b>LR6: L_RcvEOF</b>	Transmit R_IP <sub>P</sub> .		
1. CRC check not complete.	→	L_RcvEOF	
2. CRC good.	→	L_GoodCRC	
3. CRC bad.	→	L_BadEnd	
4. PHYRDY <sub>n</sub>	→	L_NoCommErr <sup>a</sup>	
<sup>a</sup> The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.			

<b>LR7: L_GoodCRC<sup>a</sup></b>	Transmit R_IP <sub>P</sub> .		
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. PHYRDY <sub>n</sub>	→	L_NoCommErr <sup>b</sup>	
5. Transport or Link layer indicated error detected during reception of recognized FIS.	→	L_BadEnd	
6. SYNC <sub>P</sub> received from Phy.	→	L_IDLE	
<sup>a</sup> Upon entering this state for the first time, the Link layer shall notify the Transport layer that the CRC for this frame is valid. <sup>b</sup> The Link layer shall notify the Transport layer of the condition and fail the attempted transfer.			

<b>LR8: L_GoodEnd</b>	Transmit R_OK <sub>P</sub> .		
1. SYNC <sub>P</sub> received from Phy.	→	L_IDLE	
2. AnyDword other than SYNC <sub>P</sub> received from Phy.	→	L_GoodEnd	
3. PHYRDY <sub>n</sub>	→	L_NoCommErr	

**Figure 272 – Link layer receiver state machine (part 3 of 4)**

<b>LR9: L_BadEnd</b>	Transmit R_ERR <sub>P</sub> .		
1. SYNC <sub>P</sub> received from Phy.	→	L_IDLE	
2. AnyDword other than SYNC <sub>P</sub> received from Phy.	→	L_BadEnd	
3. PHYRDY <sub>n</sub>	→	L_NoCommErr	

**Figure 272 – Link layer receiver state machine (part 4 of 4)**

**LR1: L\_RcvChkRdy state**, this state is entered if X\_RDY<sub>P</sub> has been received from the Phy layer.

If in this state, the Link layer shall transmit R\_RDY<sub>P</sub> and wait for SOF<sub>P</sub> 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<sub>P</sub> from the host, a device may hold off responding with R\_RDY<sub>P</sub> 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<sub>P</sub> transmission, but the response may be deferred as mentioned above.

**Transition LR1:1**, if the Link layer receives X\_RDY<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> or SOF<sub>P</sub> 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<sub>P</sub> has been received, and the FIFO is not ready to receive a FIS.

If in this state, the Link layer shall transmit SYNC<sub>P</sub>.

**Transition LR2:1**, if the Link layer receives X\_RDY<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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 44 - 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<sub>P</sub> 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<sub>P</sub> has been received from the Phy layer.

If in this state, the Link layer shall either transmit HOLD<sub>A</sub><sub>P</sub> to signal transmission to proceed if the transmitter becomes ready or transmit DMAT<sub>P</sub> to signal the transmitter to terminate the transmission.

**Transition LR5:1**, if the Link layer receives any Dword other than a HOLD<sub>P</sub> or SYNC<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub>.

**Transition LR8:1**, if the Link layer receives SYNC<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub>.

**Transition LR9:1**, if the Link layer receives SYNC<sub>P</sub> 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<sub>P</sub> 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.

### 9.6.5 Link power mode state machine

The link power mode state machine is defined in Figure 273.



<b>LPM1: L_TPMPartial</b>	Transmit PMREQ_P <sub>p</sub> .	
1. PMACK <sub>p</sub> received from Phy layer.	→	L_ChkPhyRdy
2. X_RDY <sub>p</sub> received from Phy layer.	→	L_RcvWaitFifo <sup>a</sup>
3. SYNC <sub>p</sub> or R_OK <sub>p</sub> received from Phy layer.	→	L_TPMPartial
4. AnyDword other than (PMACK <sub>p</sub> or PMNAK <sub>p</sub> or X_RDY <sub>p</sub> or SYNC <sub>p</sub> or R_OK <sub>p</sub> or PMREQ_P <sub>p</sub> <sup>c</sup> or PMREQ_S <sub>p</sub> <sup>c</sup> ) <sup>a</sup> received from Phy layer.	→	L_IDLE
5. PMREQ_P <sub>p</sub> or PMREQ_S <sub>p</sub> received from Phy layer.	→	L_TPMPartial <sup>c</sup>
6. PHYRDY <sub>n</sub>	→	L_NoCommErr <sup>b</sup>
7. PMNAK <sub>p</sub> received from Phy layer.	→	L_NoPmnaK
<sup>a</sup> 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. <sup>b</sup> This is an unexpected transition and constitutes an error condition. An error condition needs to be sent to the Transport layer as a result. <sup>c</sup> If PMREQ_P <sub>p</sub> or PMREQ_S <sub>p</sub> 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.		

<b>LPM2: L_TPMSlumber</b>	Transmit PMREQ_S <sub>p</sub> .	
1. PMACK <sub>p</sub> received from Phy layer.	→	L_ChkPhyRdy
2. X_RDY <sub>p</sub> received from Phy layer.	→	L_RcvWaitFifo <sup>a</sup>
3. SYNC <sub>p</sub> or R_OK <sub>p</sub> received from Phy layer.	→	L_TPMSlumber
4. AnyDword other than (PMACK <sub>p</sub> or PMNAK <sub>p</sub> or X_RDY <sub>p</sub> or SYNC <sub>p</sub> or R_OK <sub>p</sub> or PMREQ_P <sub>p</sub> <sup>c</sup> or PMREQ_S <sub>p</sub> <sup>c</sup> ) <sup>a</sup> received from Phy layer.	→	L_IDLE
5. PMREQ_P <sub>p</sub> or PMREQ_S <sub>p</sub> received from Phy layer.	→	L_TPMSlumber <sup>c</sup>
6. PHYRDY <sub>n</sub>	→	L_NoCommErr <sup>b</sup>
7. PMNAK <sub>p</sub> received from Phy layer.	→	L_NoPmnaK
<sup>a</sup> 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. <sup>b</sup> This is an unexpected transition and constitutes an error condition. An error condition needs to be sent to the Transport layer as a result. <sup>c</sup> If PMREQ_P <sub>p</sub> or PMREQ_S <sub>p</sub> 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)

<b>LPM3: L_PMOff</b>	Transmit PMACK <sub>P</sub> <sup>a</sup> .	
1. A total of 4 ≤ n ≤ 16 PMACK <sub>P</sub> primitives sent.	→	L_ChkPhyRdy
2. Less than n PMACK <sub>P</sub> primitives sent.	→	L_PMOff
<sup>a</sup> A flag is set according to whether a PMREQ <sub>P</sub> or PMREQ <sub>S</sub> was received from the Phy layer.		
<b>LPM4: L_PMDeny</b>	Transmit PMNAK <sub>P</sub> .	
1. PMREQ <sub>P</sub> or PMREQ <sub>S</sub> received from Phy layer.	→	L_PMDeny
2. AnyDword other than (PMREQ <sub>P</sub> or PMREQ <sub>S</sub> ) received from Phy layer.	→	L_IDLE
3. PHYRDYn	→	L_NoCommErr
<b>LPM5: L_ChkPhyRdy</b>	Assert Partial/Slumber to Phy layer (as appropriate).	
1. PHYRDY	→	L_ChkPhyRdy
2. PHYRDYn	→	L_NoCommPower
<b>LPM6: L_NoCommPower</b>	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
<b>LPM7: L_WakeUp1</b>	Negate both Partial and Slumber.	
1. PHYRDY	→	L_WakeUp2
2. PHYRDYn	→	L_WakeUp1
<b>LPM8: L_WakeUp2</b>	Transmit ALIGN <sub>P</sub> .	
1. PHYRDY	→	L_IDLE
2. PHYRDYn	→	L_NoCommErr
<b>LPM9: L_NoPmnaK</b>	Transmit SYNC <sub>P</sub> .	
1. PMNAK <sub>P</sub> received from Phy layer.	→	L_NoPmnaK
2. AnyDword other than (PMNAK <sub>P</sub> ) 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<sub>P</sub> shall be transmitted. If the Link layer receives PMACK<sub>P</sub> 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 yet processed  $PMREQ\_P_P$  yet 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 yet processed  $PMREQ\_S_P$  yet 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<sub>P</sub> or PMREQ\_S<sub>P</sub> 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<sub>P</sub>, 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<sub>P</sub> or PMREQ\_P<sub>P</sub> was received by the Link layer. The Link layer transmits PMACK<sub>P</sub> for each processing of this state.

**Transition LPM3:1**, if  $4 \leq n \leq 16$  PMACK<sub>P</sub> primitives have been transmitted, a transition shall be made to the L\_ChkPhyRdy state.

**Transition LPM3:2**, if less than  $n$  PMACK<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> for each processing of this state. In this state, the Link layer waits for receipt of any Dword that is not PMNAK<sub>P</sub> from the Phy layer.

**Transition LPM9:1**, if the Link layer receives PMNAK<sub>P</sub>, 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<sub>P</sub> from the Phy layer.

**Transition LPM9:2**, if the Link layer receives any Dword from the Phy layer other than PMNAK<sub>P</sub>, 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 Frame Information 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 98 may be referenced to refresh the reader's memory of a simplified version of the Shadow Register Block organization of an ATA adapter.

**Table 98 – Simplified shadow register block register numbering**

				Register access operation	
CS 0 Active	A2	A1	A0	Read	Write
	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

The following sections detail the types of payloads that are possible. The SOF<sub>P</sub>, EOF<sub>P</sub> and HOLD<sub>P</sub> 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.



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 9.6 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 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., ICRC bit) to one in the Error field. See ACS-3.

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

FIS Type - Set to a value of 27h. Defines the rest of the FIS fields. Defines the length of the FIS as five Dwords.

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.

Command(7:0) - Contains the contents of the Command register of the Shadow Register Block.

Control(7:0) - Contains the contents of the Device Control register of the Shadow Register Block.

LBA(7:0) - Contains the contents of the LBA low register of the Shadow Register Block.

LBA(15:8) - Contains the contents of the LBA mid register of the Shadow Register Block.

LBA(23:16) - Contains the contents of the LBA high register of the Shadow Register Block.

LBA(31:24) - 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(47:40) – Contains the contents of the expanded address field of the Shadow Register Block

Device(7:0) - Contains the contents of the Device register of the Shadow Register Block.

Features(7:0) - Contains the contents of the Features register of the Shadow Register Block.

Features(15:8) – Contains the contents of the expanded address field of the Shadow Register Block

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.

R – Reserved – shall be cleared to zero.

Count(7:0) - Contains the contents of the Sector Count register of the Shadow Register Block.

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

Count(15:8) – Contains the contents of the expanded address field of the Shadow Register Block

ICC(7:0) - Isochronous Command Completion (ICC) contains a value is 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.

Auxiliary(7:0) – Contains parameter values specified on a per command basis.

Auxiliary(15:8) – Contains parameter values specified on a per command basis.

Auxiliary(23:16) – Contains parameter values specified on a per command basis.

Auxiliary(31:24) – 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 45 - 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(7:0), Count(7:0), LBA(7:0), LBA(15:8), LBA(23:16), or Device registers if either BSY bit or DRQ bit is set to one in the Status register. Any write to the Command register if BSY bit or 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 (0)	LBA(47:40)	LBA(39:32)				LBA(31:24)	
3	Reserved (0)	Reserved (0)	Count(15:8)				Count(7:0)	
4	Reserved (0)	Reserved (0)	Reserved (0)				Reserved (0)	

**Figure 276 – Register Device to Host FIS layout**

#### Field Definitions

FIS Type - Set to a value of 34h. Defines the rest of the FIS fields. Defines the length of the FIS as five Dwords.

LBA(7:0) - Contains the new value of the LBA low register of the Shadow Register Block.

LBA(15:8) - Contains the new value of the LBA mid register of the Shadow Register Block.

LBA(23:16) - Contains the new value of the LBA high register of the Shadow Register Block.

LBA(31:24) – 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(47:40) – Contains the contents of the expanded address field of the Shadow Register Block

Device(7:0) - Contains the new value of the Device register of the Shadow Register Block.

Error(7:0) - Contains the new value of the Error register of the Shadow Register Block.

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.

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 set this field to 0h.

R - Reserved, – shall be cleared to zero.

Count(7:0) - Contains the new value of the Sector Count register of the Shadow Register Block.

Count(15:8) – Contains the contents of the expanded address field of the Shadow Register Block

Status(7:0) - Contains the new value of the Status (and Alternate status) register of the Shadow Register Block.

#### **10.5.6.2 Description**

The Register Device to Host FIS 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 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.

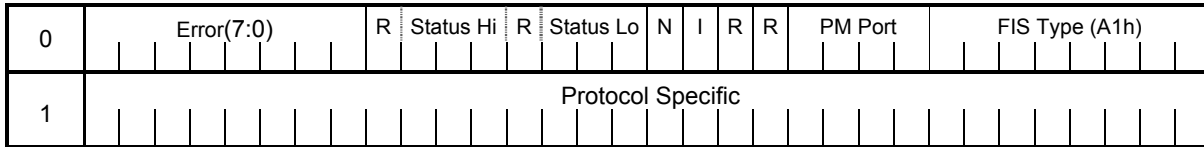
#### **10.5.6.4 Reception**

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 DRQ bit in the shadow Status register are both cleared if 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

- FIS Type** – Set to a value of A1h. Defines the rest of the FIS fields. Defines the length of the FIS as two Dwords.
- 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 zero.  
 If the host is processing native queued commands:  
 a) READ FPDMA QUEUED;  
 b) WRITE FPDMA QUEUED;  
 c) NCQ NON-DATA;  
 d) RECEIVE FPDMA QUEUED; or  
 e) SEND FPDMA QUEUED;  
 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.
- N** – Notification 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).
- Error** – Contains the new value of the Error register of the Shadow Register Block.
- 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 set this field to 0h.
- 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.
- Protocol Specific** – The value of this field is only defined for use with the Native Command Queuing Protocol. Refer to 13.6 for details. This field shall be cleared to zero for any uses other than Native Command Queuing (e.g., Asynchronous Notification).
- R** – Reserved – shall be cleared to zero.

### **10.5.7.2 Description**

The Set Device Bits (SDB) FIS 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

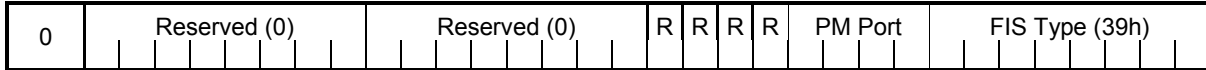


Figure 278 – DMA Activate – Device to Host FIS layout

#### Field Definitions

FIS Type - Set to a value of 39h. Defines the rest of the FIS fields. Defines the length of the FIS as one Dword.

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 set this field to 0h.

R - Reserved – shall be cleared to zero.

### 10.5.8.2 Description

The DMA Activate FIS 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**

0	Reserved (0)	Reserved (0)	A	I	D	R	PM Port	FIS Type (41h)
1	DMA Buffer Identifier Low							
2	DMA Buffer Identifier High							
3	Reserved (0)							
4	DMA Buffer Offset							
5	DMA Transfer Count							
6	Reserved (0)							

**Figure 279 – DMA Setup – Device to Host or Host to Device FIS layout**

Field Definitions

- 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.
- D Direction - 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.
- A Auto-Activate - 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.
- 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 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 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 zero.
- DMA Transfer Count - This is the number of bytes to be read or written. Bit zero shall be zero.
- I Interrupt - 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.
- 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 set this field to 0h for Device to Host transmissions.
- R - Reserved – shall be cleared to zero.

### **10.5.9.2 Description**

The DMA Setup – Device to Host or Host to Device FIS 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, the first Dword of a subsequent Data – Host to Device FIS or Device to Host FIS shall be 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 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 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 46 - 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 host bus adapter shall ensure the First-party DMA Data Phase is uninterrupted. Unless the ERR bit in the shadow Status register is set, 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 (0)	Pattern Definition T   A   S   L   F   P   R   V	R   R   R   R	PM Port	FIS Type (58h)
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

FIS Type - Set to a value of 58h. Defines the rest of the FIS fields.

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 set this field to 0h for Device to Host transmissions.

R - Reserved – shall be cleared to zero.

#### Pattern Definition

F – Far End Analog (i.e., analog front end (AFE)) Loopback (Optional)

L - Far End Retimed Loopback\* Transmitter shall insert additional ALIGN<sub>P</sub> primitives

T - Far end transmit only mode

A - ALIGN<sub>P</sub> Bypass (Do not Transmit ALIGN<sub>P</sub> primitives) (valid only in combination with T Bit)

S - Bypass Scrambling (valid only in combination with T Bit)

P - Primitive bit. (valid only in combination with the T Bit) (Optional)

V - Vendor Specific Test Mode. Causes all other bits to be ignored

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 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 Physical 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<sub>P</sub> primitives after reception of R\_OK<sub>P</sub> 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

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

places its own Application, Transport and Physical 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<sub>P</sub> and has received at least one good SYNC<sub>P</sub>. 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 47 - 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 99 for valid bit settings within a BIST Activate FIS.

**Table 99 – BIST Activate FIS modes and bit settings**

<b>BIST Test Mode</b>	<b>F</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>A</b>	<b>S</b>	<b>V</b>
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 <sub>P</sub> primitives, scrambled data	0	0	1	0	0	0	0
Far End Transmit with ALIGN <sub>P</sub> primitives, unscrambled data	0	0	1	0	0	1	0
Far End Transmit without ALIGN <sub>P</sub> primitives, scrambled data	0	0	1	0	1	0	0
Far End Transmit without ALIGN <sub>P</sub> primitives, unscrambled data	0	0	1	0	1	1	0
Far End Transmit primitives with ALIGN <sub>P</sub> primitives	0	0	1	1	0	na	0
Far End Transmit primitives without ALIGN <sub>P</sub> 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 Loopback Mode 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<sub>P</sub> primitives every 256 Dwords transmitted and, if it requires any ALIGN<sub>P</sub> primitives to be present in the returned data stream, it should insert additional ALIGN<sub>P</sub> primitives in the transmitted stream. The initiator shall transmit additional two consecutive ALIGN<sub>P</sub> primitives in a single burst at the normal interval of every 256 Dwords transmitted (as opposed to inserting two consecutive ALIGN<sub>P</sub> primitives at half the interval).

The loopback device may remove zero, one, or two ALIGN<sub>P</sub> primitives from the received data. It may insert one or more ALIGN<sub>P</sub> primitives if they are directly preceded or followed by the initiator inserted ALIGN<sub>P</sub> primitives (resulting in at least two consecutive ALIGN<sub>P</sub> primitives) or it may insert two or more ALIGN<sub>P</sub> primitives if not preceded or followed by the initiator's ALIGN<sub>P</sub> primitives. One side effect of the loopback retiming is that the returned data stream may have instances of an odd number of ALIGN<sub>P</sub> primitives, however, returned ALIGN<sub>P</sub> primitives are

always in bursts and if the initiator transmitted four consecutive ALIGN<sub>P</sub> primitives, then the returned data stream shall include ALIGN<sub>P</sub> bursts that are no shorter than two ALIGN<sub>P</sub> primitives long (although the length of the ALIGN<sub>P</sub> burst may be odd). The initiator of the retimed loopback mode shall not assume any relationship between the relative position of the ALIGN<sub>P</sub> primitives returned by the loopback device and the relative position of the ALIGN<sub>P</sub> 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 ALIGN<sub>P</sub> 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 9.4. 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**, ALIGN<sub>P</sub> primitive bypass mode. If set to one, no ALIGN<sub>P</sub> primitives are sent. If the A bit is not asserted, ALIGN<sub>P</sub> 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 (0)	LBA(47:40)	LBA(39:32)				LBA(31:24)	
3	E_Status(7:0)	Reserved (0)	Count(15:8)				Count(7:0)	
4	Reserved (0)				Transfer Count			

**Figure 281 – PIO Setup – Device to Host FIS layout**

#### Field Definitions

FIS Type - Set to a value of 5Fh. Defines the rest of the FIS fields. Defines the length of the FIS as five Dwords.

LBA(15:8) - Holds the contents of the LBA(15:8) register of the Command Block.

LBA(39:32) – Contains the contents of the LBA(39:32) field of the Shadow Register Block

LBA(23:16) - Holds the contents of the LBA(23:16) register of the Command Block.

LBA(47:40) – Contains the contents of the LBA(47:40) field of the Shadow Register Block

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.

Device(7:0) - Holds the contents of the Device register of the Command Block.

Status(7:0) - Contains the new value of the Status register of the Command Block for initiation of host data transfer.

Error(7:0) - Contains the new value of the Error register of the Command Block at the conclusion of all subsequent Data to Device frames.

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.

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 set this field to 0h.

R – Reserved – shall be cleared to zero.

Count(7:0) - Holds the contents of the Count(7:0) register of the Command Block.

Count(15:8) – Contains the contents of the Count(15:8) field of the Shadow Register Block

LBA(7:0) - Holds the contents of the LBA(7:0) register of the Command Block.

LBA(31:24) – Contains the contents of the LBA(31:24) field of the Shadow Register Block

E\_Status(7:0) - Contains the new value of the Status register of the Command Block at the conclusion of the subsequent Data FIS.

Transfer Count – Holds 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 zero (even number of bytes transferred).

#### **10.5.11.2 Description**

The PIO Setup – Device to Host FIS 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 (BSY bit is cleared to zero and 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.

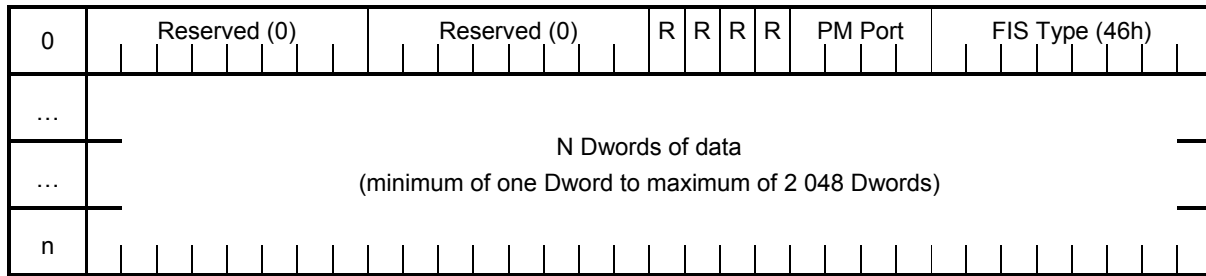


#### **10.5.11.6 Reception of PIO Setup by Host Prior to a Data Transfer from Device to Host**

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

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 48 - 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.

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 set this field to 0h for Device to Host transmissions.

R – Reserved – shall be cleared to zero.

### 10.5.12.2 Description

The Data – Host to Device and the Data – Device to Host FISes 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 shall be 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 <sup>a</sup>
3.	Frame receipt indicated by Link layer	→	HT_ChkTyp <sup>c</sup>
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 <sup>b</sup>
<p><sup>a</sup> 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.</p> <p><sup>b</sup> 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.</p> <p><sup>c</sup> FIS reception shall have priority over all other transitions in this state.</p>			

**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 <sup>a</sup>
8. Data FIS type detected and previous FIS was PIO Setup	→		HT_PIOITrans1 <sup>a</sup>
9. Unrecognized FIS received	→		HT_HostIdle
10. Notification of illegal transition error received from Link layer	→		HT_HostIdle
<sup>a</sup> 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.

**Transition HTI1:1**, 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 283).

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 <sup>a</sup>	→	HT_HostIdle
<sup>a</sup> Upon return to the HT_HostIdle state in response to a detected error, the associated FIS remains pending for transmission			

**Figure 284 – Host transport command 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 <sup>a</sup>	→	HT_HostIdle
<sup>a</sup> Upon return to the HT_HostIdle state in response to a detected error, the associated FIS remains pending for transmission			

**Figure 285 – Host transport control 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 buffer identifier, the host buffer offset, the byte count, and the interrupt flag (see Figure 286).

HTDMASTERUP0: HT_DMMASTERUPFIS	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_DMMASTERUPTrans Status
	2. Notification of illegal transition error received from Link layer	→	HT_HostIdle
	3. Frame receipt indicated by Link layer	→	HT_HostIdle
HTPDMASTERUP1: HT_DMMASTERUPTransStatus	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 <sup>a</sup>	→	HT_HostIdle
<sup>a</sup> Upon return to the HT_HostIdle state in response to a detected error, the associated FIS remains pending for transmission			

Figure 286 – Host transport DMA setup state machine

**HTDMASTERUP0: HT\_DMMASTERUPFIS 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 HTDMASTERUP0:1**, 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 HTDMASTERUP1: HT\_DMMASTERUPTransStatus state.

**Transition HTDMASTERUP0:2**, if the Transport layer receives notification from the Link layer of an illegal state transition, the Transport layer shall make a transition to the HT11: HT\_HostIdle state.

**Transition HTDMASTERUP0:3**, if the Link layer indicates that a FIS is being received, the Transport layer shall make a transition to the HT11:HT\_HostIdle state.

**HTPDMASTERUP1: HT\_DMMASTERUPTransStatus 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 HTDMASTERUP1:1**, if the FIS status has been handled, and no error detected, the Transport layer shall transition to the HT11: HT\_HostIdle state.

**Transition HTDMASTERUP1:2**, if the FIS status has been handled, and an error detected, the Transport layer shall transition to the HT11: HT\_HostIdle state. The associated FIS remains pending for transmission.

### 10.6.6 Host transport transmit BIST Activate FIS

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 Physical 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 Physical 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 <sup>a</sup>
	<sup>a</sup> Re-transmission of the BIST Activate FIS due to errors is not required but allowed.		

**Figure 287 – Host transport transmit BIST 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 register state machine**

**HTR1: HT\_RegFIS state**, this state is entered 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 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, 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 set device bits 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 DMA activate 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 49 - The host should not 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<sub>P</sub>, 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 <sup>a</sup>
	2. SRST asserted, or DEVICE RESET command requested	→	HT_HostIdle
	3. Notification of illegal transition error received from Link layer	→	HT_HostIdle
<sup>a</sup> 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 PIO setup 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 PIO setup 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, the FIS byte count, and 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 50 - 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 51 - The host should not 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 (Dbit 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 DMA setup 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 (Direction is set to one) or the request is a write (Direction is cleared to zero) and Auto-Activate is 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:HT\_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 Physical 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 BIST 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 Physical 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 – Host transport register device to host state machine**

**DTR0: DT\_RegDHFIS state**, this state is entered 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 DTIO: DT\_DeviceIdle state.

**DTR1: DT\_RegTransStatus state**, this state is 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 DTIO: 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_SDBTransSt atus	
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 setup device bits 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 DTIO: 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 DTIO: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_DeviceIdle
	2. Status checked, and error detected.	→	DT_PIOSTUPFIS

**Figure 297 – Device transport PIO setup state machine**

**DTPIOSTUP0: DT\_PIOSTUPFIS state**, this state is entered 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 DMA activate 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 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 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 DMA setup device to host 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 DTDMASTUP0:2**, 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\_DATAIFIS 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<sub>P</sub> 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<sub>P</sub>, 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 Physical layer to enter the appropriate mode for the transmission of the Built-in Test data defined by the FIS. The means that the Transport, Link and Physical layers 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 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 and Phy 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 <sup>a</sup>
<sup>a</sup> 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, then the transition to DTI0: DT\_DeviceIdle is also accommodated. After issuing an Abort, the Transport returns to normal data transfer, and awaits the end of transfer indication from the Link.

**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, Link or Physical layers 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 Physical 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) then, regardless of the device power management mode (e.g., SLEEP, STANDBY) or the current device command layer state, 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 does not implement the PACKET command feature set the register content shall be as defined in Figure 307.

Count(7:0)	01h
LBA(7:0)	01h
LBA(15:8)	00h
LBA(23:16)	00h
Device(7:0)	na
Error(7:0)	01h
Status(7:0)	00h..70h <sup>a</sup>
<sup>a</sup> Setting of bits 6:4 in the Status register are device specific.	

**Figure 307 – DHR2: 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 308.

Count(7:0)	01h
LBA(7:0)	01h
LBA(15:8)	14h
LBA(23:16)	EBh
Device(7:0)	na
Error(7:0)	01h
Status(7:0)	00h

**Figure 308 – DHR2: Send\_good\_status, PACKET command feature set 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 does not implement the PACKET command feature set the register content shall be as defined in Figure 309.

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

Count(7:0)	01h
LBA(7:0)	01h
LBA(15:8)	00h
LBA(23:16)	00h
Device(7:0)	na
Error(7:0)	00h, 02h..7Fh
Status(7:0)	00h..70h <sup>a</sup>
<sup>a</sup> Setting of bits 6:4 in the Status register are device specific.	

**Figure 309 – DHR3: 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 310.

Count(7:0)	01h
LBA(7:0)	01h
LBA(15:8)	14h
LBA(23:16)	EBh
Device(7:0)	na
Error(7:0)	00h, 02h..7Fh
Status(7:0)	00h

**Figure 310 – DHR3: Send\_bad\_status, PACKET command feature set not implemented**

**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 311). States and transitions utilized if Native Command Queuing 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. <sup>b</sup>	→	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. <sup>b</sup>	→	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. <sup>b</sup>	→	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. <sup>b</sup>	→	DFPDMAQ10: SendStatus <sup>a</sup>	
6. FPDMA QUEUED or NCQ NON-DATA command terminated with failure and FIS receipt not indicated. <sup>b</sup>	→	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	
<sup>a</sup> 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. <sup>b</sup> States and transitions utilized if Native Command Queuing 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 <sup>a</sup>	
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	
<sup>a</sup> A Register Device to Host FIS shall not be sent in response to the received Register Host to Device FIS.			

**Figure 311 – Device command layer idle state machine**

(part 1 of 3)

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

DI2: Check_command <sup>a</sup>	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. <sup>b</sup>	→	DDMAQI0: DMA_queued_in	
8. WRITE DMA QUEUED command protocol and no native queued command outstanding. <sup>b</sup>	→	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. <sup>b</sup>	→	DI5: Service_test	
13. FPDMA QUEUED command protocol. <sup>b</sup>	→	DFPDMAQ1: AddCommand_ToQueue	
14. Not READ FPDMA QUEUED and not WRITE FPDMA QUEUED and not NCQ NON-DATA and not RECEIVE FPDMA QUEUED and not SEND FPDMA QUEUED and not DEVICE RESET and native queued command(s) outstanding. <sup>b</sup>	→	DFPDMAQ12: BrokenHost_ClearBusy	
<sup>a</sup> This state shows transitions for all commands. If a device does not implement any particular command, then that transition should not be processed.			
<sup>b</sup> States and transitions utilized if Native Command Queuing 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 <sup>a</sup>	Request transmission of Set Device Bits FIS with SERV set.		
1. FIS transmission complete.	→	DI0: Device_idle	
<sup>a</sup> States and transitions utilized if Native Command Queuing or ATA Tagged Command Queuing commands are implemented.			

Figure 311 – Device command layer idle state machine (part 2 of 3)

DI5: Service_test <sup>a</sup>	Test command to see if Register Device to Host FIS is needed to set DRQ bit and set Tag.	
1. PACKET PIO data-in or PACKET PIO data-out.	→	DI7: Service_decode
2. Other	→	DI6: Service_send_tag
<sup>a</sup> States and transitions utilized if Native Command Queuing or ATA Tagged Command Queuing commands are implemented.		
DI6: Service_send_tag <sup>a</sup>	Request transmission of Register Device to Host FIS with BSY bit cleared to zero, DRQ bit set to one, and appropriate Tag	
1. FIS transmission complete	→	DI7: Service_decode
<sup>a</sup> States and transitions utilized if Native Command Queuing or ATA Tagged Command Queuing commands are implemented.		
DI7: Service_decode <sup>a</sup>	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.	→	DDMAQ11: Send_data
6. WRITE DMA QUEUED.	→	DDMAQO1: Send_DMA_activate
<sup>a</sup> States and transitions utilized if Native Command Queuing or ATA Tagged Command Queuing commands are implemented		

**Figure 311 – 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 D10:5**, if the device has successfully completed a FPDMA QUEUED or a NCQ NON-DATA command, the device shall transition to the DFPDMAQ10: SendStatus state.

\* **Transition D10:6**, 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 D10:7**, 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.

**D11: 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 D11:1**, 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 D11:2**, 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 D10: Device\_idle state.

**Transition D11:3**, 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 D12: Check\_command state.

**Transition D11:4**, 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 D10: Device\_idle state.

**Transition D11:5**, for any other FIS, the device shall transition to the D10: Device\_idle state.

**D12: Check\_command state**, this state is entered if the device recognizes that the received Register Host to Device FIS contains a new command.

NOTE 52 - this state shows transitions for all commands. If a device does not implement any particular command, then transition D12:11 to state D13: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 D12:1**, if the received command is a non-data transfer command, the device shall transition to the DND0: Non-data state.

**Transition D12:2**, if the received command is a PIO data-in command, the device shall transition to the DPIOI0: PIO\_in state.

**Transition D12:3**, if the received command is a PIO data-out command, the device shall transition to the DPIOO0: PIO\_out state.

**Transition D12:4**, 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 an 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 not READ FPDMA QUEUED; and not WRITE FPDMA QUEUED; and not NCQ NON-DATA; and not RECEIVE FPDMA QUEUED; and not SEND FPDMA QUEUED; and not DEVICE RESET; 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-3 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 the 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.

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

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 DRQ 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-3 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 the DI7: 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 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 312).

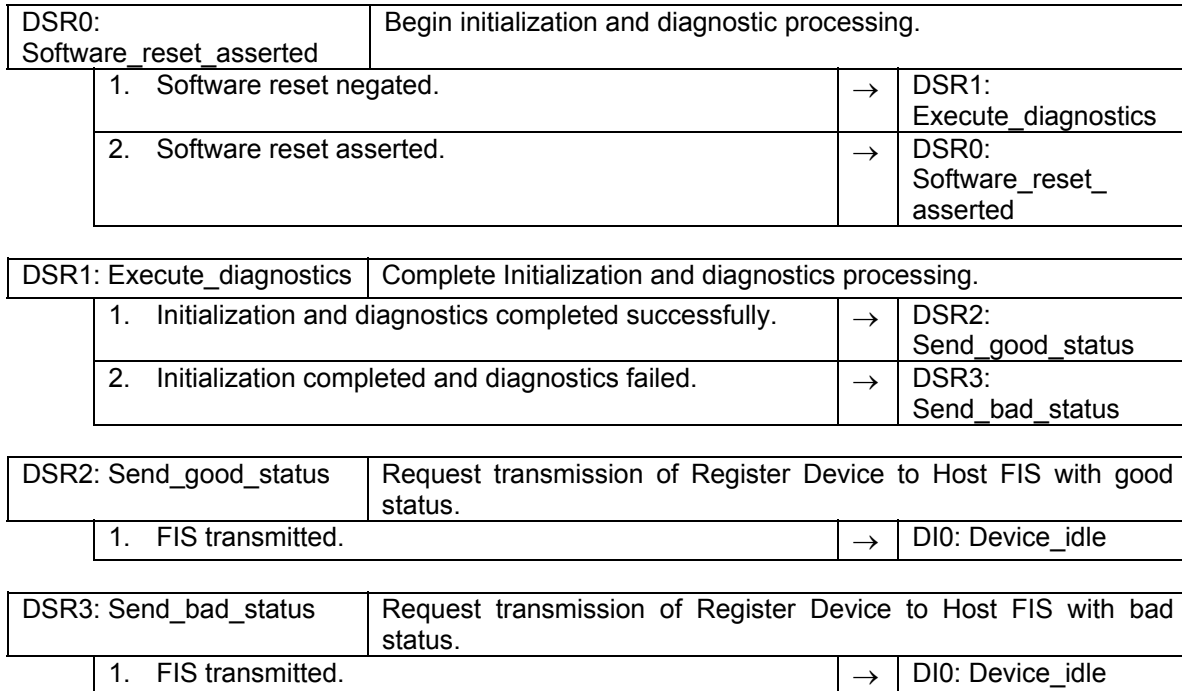


Figure 312 – Device command layer software reset state machine

**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 313.

Count(7:0)	01h
LBA(7:0)	01h
LBA(15:8)	00h
LBA(23:16)	00h
Device(7:0)	na
Error(7:0)	01h
Status(7:0)	00h..70h <sup>a</sup>
<sup>a</sup> Setting of bits 6:4 in the Status register are device specific.	

**Figure 313 – DSR2: 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 314.

Count(7:0)	01h
LBA(7:0)	01h
LBA(15:8)	14h
LBA(23:16)	EBh
Device(7:0)	na
Error(7:0)	01h
Status(7:0)	00h

**Figure 314 – DSR2: Send\_good\_status, PACKET command feature set not 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 315.

Count(7:0)	01h
LBA(7:0)	01h
LBA(15:8)	00h
LBA(23:16)	00h
Device(7:0)	na
Error(7:0)	00h, 02h..7Fh
Status(7:0)	00h..70h <sup>a</sup>
<sup>a</sup> Setting of bits 6:4 in the Status register are device specific.	

**Figure 315 – 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 316.

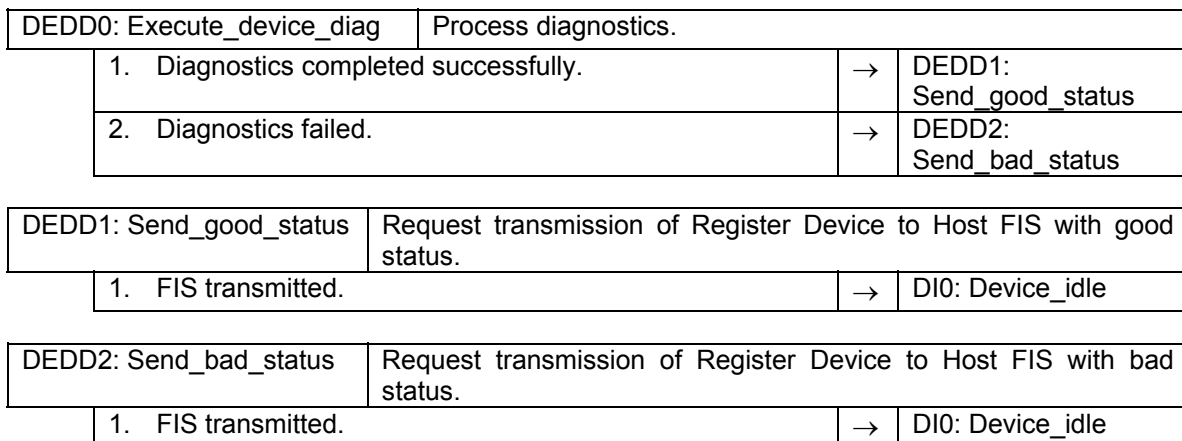
Count(7:0)	01h
LBA(7:0)	01h
LBA(15:8)	14h
LBA(23:16)	EBh
Device(7:0)	na
Error(7:0)	00h, 02h..7Fh
Status(7:0)	00h

**Figure 316 – DSR3: Send\_bad\_status, PACKET command feature set 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 from the beginning (see Figure 317). 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.



**Figure 317 – 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 318.

Count(7:0)	01h
LBA(7:0)	01h
LBA(15:8)	00h
LBA(23:16)	00h
Device(7:0)	na
Error(7:0)	01h
Status(7:0)	00h..70h <sup>a</sup>
<sup>a</sup> Setting of bits 6:4 in the Status register are device specific.	

**Figure 318 – 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 319.

Count(7:0)	01h
LBA(7:0)	01h
LBA(15:8)	14h
LBA(23:16)	EBh
Device(7:0)	na
Error(7:0)	01h
Status(7:0)	00h

**Figure 319 – DEDD1: Send\_good\_status, PACKET command feature set not 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 320.

Count(7:0)	01h
LBA(7:0)	01h
LBA(15:8)	00h
LBA(23:16)	00h
Device(7:0)	na
Error(7:0)	00h, 02h..7Fh
Status(7:0)	00h..70h <sup>a</sup>
<sup>a</sup> Setting of bits 6:4 in the Status register are device specific.	

**Figure 320 – 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 321.

Count(7:0)	01h
LBA(7:0)	01h
LBA(15:8)	14h
LBA(23:16)	EBh
Device(7:0)	na
Error(7:0)	00h, 02h..7Fh
Status(7:0)	00h

**Figure 321 – 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 322). 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 322 – 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 323.

Count(7:0)	01h
LBA(7:0)	01h
LBA(15:8)	14h
LBA(23:16)	EBh
Device(7:0)	na
Error(7:0)	01h
Status(7:0)	00h

**Figure 323 – 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 324). See the NOP command description and the SLEEP command description in ACS-3 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 324 – 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-3 standard and the Interrupt bit set to one.

**Transition DND1:1**, if the FIS has been transmitted, then the device shall transition to the DIO: 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 325).

DPIO10: PIO_in	Prepare a DRQ data block for transfer to the host.		
	1. DRQ data block ready to transfer and no error encountered.	→	DPIO11: Send_PIO_setup
	2. Error encountered during command processing.	→	DPIO13: Error_status
DPIO11: Send_PIO_setup	Request transmission of a PIO Setup FIS to host.		
	1. PIO Setup FIS transmitted.	→	DPIO12: Transmit_data
DPIO12: Transmit_data	Request transmission of a Data FIS to host.		
	1. Data FIS transmitted, no more data transfer required for this command.	→	DIO: Device_idle
	2. Data FIS transmitted, more data transfer required for this command, or 2 048 Dwords transmitted.	→	DPIO10: PIO_in
DPIO13: Error_status	Request transmission of a Register Device to Host FIS.		
	1. FIS transmission complete.	→	DIO: Device_idle

**Figure 325 – Device command layer PIO data-in state machine**

**DPIO10: 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 DPIO10:1**, if the device has a DRQ data block ready to transfer and no error was encountered, the device shall transition to the DPIO11: Send\_PIO\_setup state.

**Transition DPIO10:2**, if the device has encountered an error during command processing, the device shall transition to the DPIO13: Error\_status state.

**DPIO11: 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 BSY bit cleared to zero and DRQ bit set to one and with register content as described in the command description in the ACS-3 standard. The Interrupt bit shall be set. If this is the last DRQ data block requested by the command, the ending status shall have BSY bit cleared to zero and DRQ bit cleared to zero. If this is not the last data block requested by the command, the ending status shall have BSY bit set to one and DRQ bit cleared to zero.

**Transition DPIO11:1**, if the PIO Setup FIS has been transferred, the device shall transition to the DPIO12: 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 DIO: 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-3 standard and the Interrupt bit set to one. In addition to the ACS-3 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 DIO: 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 326).

DPIO00: PIO_out	Prepare to receive DRQ data block transfer from the host.		
1. Ready to receive DRQ data block transfer.		→	DPIO01: Send_PIO_setup
2. All DRQ data blocks received or command aborted due to error.		→	DPIO03: Send_status
DPIO01: Send_PIO_setup	Request transmission of a PIO Setup FIS to host.		
1. PIO Setup FIS transmitted.		→	DPIO02: Receive_data
DPIO02: Receive_data	Receive Data FIS from the Transport layer.		
1. Data FIS received.		→	DPIO00: PIO_out
DPIO03: Send_status	Request transmission of a Register Device to Host FIS.		
1. FIS transmission complete.		→	DI0: Device_idle

**Figure 326 – Device command layer PIO data-out state machine**

**DPIO00: 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 DPIO00:1**, if the device is ready to receive a DRQ data block, the device shall transition to the DPIO01: Send\_PIO\_setup state.

**Transition DPIO00:2**, 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 DPIO03: Send\_status state.

**DPIO01: 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 BSY bit cleared to zero and 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 BSY bit set to one and DRQ bit cleared to zero. The byte count for the DRQ data block shall be indicated.

**Transition DPIO01:1**, if the PIO Setup FIS has been transferred, the device shall transition to the DPIO02: 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-3 standard and the Interrupt bit set to one. In addition to the ACS-3 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 DIO: 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 327).

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.	→	DIO: Device_idle

**Figure 327 – 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.

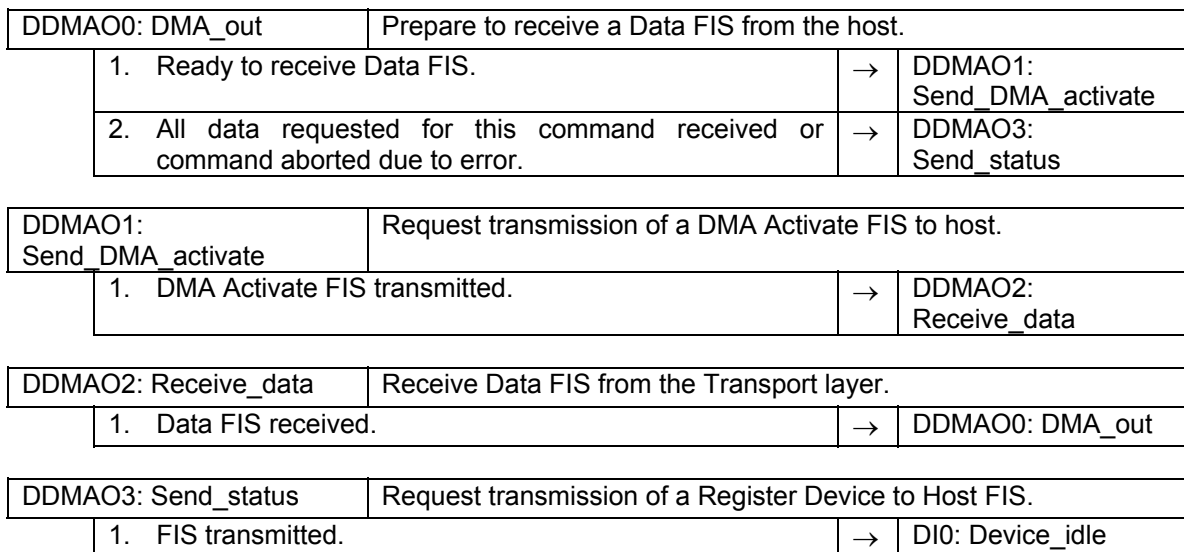
**DDMAI2: Send\_status**, this state is entered if the device has transferred all of the data requested by the command or has encountered an error that causes the command to abort before completing the transfer of the requested data.

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-3 standard and the Interrupt bit set to one.

**Transition DDMAI2:1**, if the FIS has been transmitted, the device shall transition to the DIO: 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 328). A single interrupt is issued at the completion of the successful transfer of all data required by the command.



**Figure 328 – 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-3 standard and the Interrupt bit set to one.

**Transition DDMAO3:1**, if the FIS has been transmitted, the device shall transition to the DIO: Device\_idle state.

## 11.12 PACKET protocol

The device command layer PACKET protocol is defined in Figure 329.

States and transitions only utilized if queuing is implemented have a footnote.

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 block is not ready for immediate transfer <sup>a</sup>	→	DPKT15: Release	
<sup>a</sup> States and transitions 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 block immediately. <sup>a</sup>	→	DPKT15: Release	
<sup>a</sup> States and transitions utilized if queuing is implemented.			

**Figure 329 – Device command layer packet state machine**

(part 1 of 2)



HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

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. <sup>a</sup>	→	DPKT15: Release
<sup>a</sup> States and transitions 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. <sup>a</sup>	→	DPKT15: Release
<sup>a</sup> States and transitions 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 <sup>a</sup>	Request transmission of a Register Device to Host FIS.	
1. FIS transmission complete.	→	DI0: Device_idle
<sup>a</sup> States and transitions utilized if queuing is implemented.		

**Figure 329 – 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 BSY bit cleared to zero and DRQ bit set to one. The Interrupt bit shall be cleared to zero. The ending status shall have BSY bit set to one and 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 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 BSY bit cleared to zero and DRQ bit set to one. The Interrupt bit shall be set to one. The ending status shall have BSY bit set to one and 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 BSY bit cleared to zero and DRQ bit set to one. The Interrupt bit shall be set to one. The ending status shall have BSY bit set to one and 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-3 standard and the Interrupt bit set to one.

**Transition DPKT14:1**, if the FIS has been transmitted, then the device shall transition to the DIO: 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-3 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 DIO: 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 330). 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 330 – 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 DDMAQ1:1**, if the Data FIS has been transferred, the device shall transition to the DDMAQ2: Prepare\_data state. The device command layer shall request a Data FIS size of no more than 2 048 Dwords .

**DDMAQ2: 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 DDMAQ2:1**, if data is ready for the Data FIS, the device shall transition to the DDMAQ1: Send\_data state.

**Transition DDMAQ2:2**, 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 DDMAQ3: Send\_status state.

**DDMAQ3: 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-3 standard and the Interrupt bit set to one.

**Transition DDMAQ3:1**, if the FIS has been transmitted, the device shall transition to the DIO: Device\_idle state.

**DDMAQ4: 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-3 standard, and, if the bus release interrupt has been enabled by a previous Set Features Command, with the Interrupt bit set to one.

**Transition DDMAQ4:1**, if the FIS has been transmitted, then the device shall transition to the DIO: 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 331). 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 331 – 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-3 standard and the Interrupt bit set to one.

**Transition DDMAQO4:1**, if the FIS has been transmitted, then the device shall transition to the DIO: 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-3 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 DIO: Device\_idle state.

## 11.15 FPDMA QUEUED command protocol

The device command layer FPDMA Queued state machine is defined in Figure 332.

This class includes:

- a) READ FPDMA QUEUED;
- b) WRITE FPDMA QUEUED;
- c) NCQ NON-DATA;
- d) RECEIVE FPDMA QUEUED; and
- e) SEND FPDMA QUEUED.

DFPDMAQ1: AddCommandToQueue	Append command to internal device command queue and store TAG value.		
	1. Device successfully en-queued the command	→	DFPDMAQ2: ClearInterfaceBsy
	2. Command malformed	→	DFPDMAQ12: BrokenHost_ ClearBusy
DFPDMAQ2: ClearInterfaceBsy	Transmit Register Device to Host FIS with BSY bit cleared to zero and 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 332 – Device command layer FPDMA queued state machine**

(part 1 of 3)

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

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 <sup>a</sup>	→	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	
<sup>a</sup> 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 <sup>a</sup>	→	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	
<sup>a</sup> 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 ERR bit cleared to zero, Interrupt bit set to one, and bit <b>n</b> in ACT field set to one where <b>n</b> = 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 ERR bit in Status field set to one, Interrupt bit set to one, ATA error code set to one in Error field, and 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 332 – 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 ERR bit in Status field set to one, Interrupt bit set to one, BSY bit cleared to zero, DRQ bit cleared to zero, and 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 <sup>a</sup> .	→	DFPDMAQ15: SendQueue_CleanACKDMA
	3. SRST received	→	DSR0: Software_reset_asserted
	4. Any other command received	→	DFPDMAQ12: BrokenHost_ClearBusy
	<sup>a</sup> See 13.7		
DFPDMAQ14: SendQueue_CleanACK	Discard all commands in the pending device queue. Transmit Set Device Bits FIS with ERR in Status field cleared to zero, Error field set 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 ERR in Status field cleared to zero, Error field set to 00h, ACT field = FFFF FFFFh, and Interrupt bit cleared to zero.		
	1. Set Device Bits FIS transmission complete	→	DDMAI0: DMA_in

**Figure 332 – 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 Native Command Queuing is supported and enabled.

If in this state, the device shall check the TAG 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 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, 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.

**DFPDMAQ2: ClearInterfaceBusy**, this state is entered if the device has appended the command to its internal queue and is ready transmit a Register Device to Host FIS with BSY bit cleared to zero and 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 DIO: 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 Direction 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 Direction 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 Direction 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 DIO: 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 DIO: Device\_idle state.

**Transition DFPDMAQ8:4**, if the device determines that an unrecoverable error has occurred, the device shall transition to the DIO: 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 DIO: 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 DIO: Device\_idle state.

**Transition DFPDMAQ9:4**, if the device determines that an unrecoverable error has occurred, the device shall transition to the DIO: 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 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 DIO: 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 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 the device has received a READ FPDMA QUEUED or WRITE FPDMA QUEUED or NCQ NON-DATA or RECEIVE FPDMA QUEUED or SEND FPDMA QUEUED command with a TAG that already exists in its command queue, or if the received command is not READ FPDMA QUEUED; and not WRITE FPDMA QUEUED; and not NCQ NON-DATA; and not RECEIVE FPDMA QUEUED; and not SEND FPDMA QUEUED; and not DEVICE RESET; and there are native queued command(s) outstanding.

If in this state, the device halts command processing and transmits a Register Device to Host FIS with ERR set to one in the Status field, Interrupt bit set to one, BSY bit cleared to zero, DRQ bit cleared to zero, and ATA error code set to one in the Error field. 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.

**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 and 13.7) 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 ERR bit in the Status field cleared to zero, Error field set 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 ERR bit in the Status field cleared to zero, Error field set 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 Native Command Queuing 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:

- a) READ FPDMA QUEUED;
- b) WRITE FPDMA QUEUED;
- c) NCQ NON-DATA;
- d) RECEIVE FPDMA QUEUED; and
- e) SEND FPDMA QUEUED.

### 12.2 FPDMA QUEUED command protocol

The host command layer FPDMA Queued state machine is defined in Figure 333.

HFPIO: Idle <sup>a</sup>			
1. Free TAG location and command waiting to have TAG assigned	→	HFPDMAQ2: PresetACTBit	
2. Command with assigned TAG awaiting issue and BSY bit cleared to zero and not First-party DMA Data Phase	→	HFPDMAQ3: IssueCommand	
3. Interrupt received from device.	→	HFPDMAQ4: DeviceINT	
4. Default	→	HFPIO: Idle	
<sup>a</sup> 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	→	HFPIO: 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	→	HFPIO: 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)	→	HFPIO: Idle	
2. FIS transmission deferred (command not issued)	→	HFPIO: Idle	

**Figure 333 – 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 ERR bit in SavedStatus value		
	1. ERR bit cleared to zero	→	HFPIO: Idle
	2. ERR 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 DRQ bit set to one and BSY bit cleared to zero <sup>a</sup>		
	1. DRQ bit set to one and BSY bit cleared to zero	→	HFPDMAQ10: RetrieveRequest_ Sense
	2. DRQ cleared to zero or BSY set to one	→	HFPDMAQ9: CleanupACK
	<sup>a</sup> The host may wait for this condition using any means including awaiting an interrupt and checking the DRQ bit and BSY bit status, spinning, or periodic timer.		
HFPDMAQ10: RetrieveRequest_Sense	Receive PIO Data FIS with Queued Error Log contents		
	1. PIO Data FIS reception complete	→	HFPDMAQ11: ErrorFlush

**Figure 333 – Host command layer FPDMA queued state machine (part 2 of 3)**

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	→	HFPDIO: 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 333 – Host command layer FPDMA queued state machine (part 3 of 3)**

**HFPIO: Idle**, if in this state, if queuing is supported and enabled, the Command layer is awaiting a:

- a) READ FPDMA QUEUED;
- b) WRITE FPDMA QUEUED;
- c) NCQ NON-DATA;
- d) RECEIVE FPDMA QUEUED; or
- e) SEND FPDMA QUEUED:
  - 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 HFPIO:1**, if a READ FPDMA QUEUED, WRITE FPDMA QUEUED, NCQ NON-DATA, RECEIVE FPDMA QUEUED, or SEND FPDMA QUEUED 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 HFPIO:2**, if a command with assigned TAG value is awaiting issue to the device and 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 HFPIO:3**, if an interrupt is received from the device, indicating status is available for a previously queued command, it shall transition to the HFPDMAQ4: DeviceINT state.

**Transition HFPIO:4**, if the queuing is supported and enabled, and the Command layer is awaiting a free TAG, a new command, or an interrupt for a previously queued command, it shall transition to the HFPIO: Idle state.

**HFPDMAQ1: AddCommandToQueue**, the Command layer enters this state if it has received a READ FPDMA QUEUED, WRITE FPDMA QUEUED, NCQ NON-DATA, RECEIVE FPDMA QUEUED, or SEND FPDMA QUEUED 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 command to the internal host command queue, it shall transition to the HFPIO: Idle state.

**HFPDMAQ2: PresetACTBit,** if in this state, the Command layer assigns a free TAG value to the previously queued 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 HFPIO: Idle state.

**HFPDMAQ3: IssueCommand,** if in this state, the Command layer attempts to issue a command with preassigned TAG to the device by transmitting a Register Host to Device FIS with the new 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 command as issued and transition to the HFPIO: 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 HFPIO: Idle state. The corresponding 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 commands.

**Transition HFPDMAQ5:1,** if the SActive comparison indicates one or more commands have completed, it shall transition to the HFPDMAQ6: CompleteRequests2 state.

**Transition HFPDMAQ5:2,** if the SActive comparison indicates no 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 HFPIO: 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 and 13.7) 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 HFPDMAQ8: ResetQueue state.

**HFPDMAQ9: CleanupACK,** if in this state, the Command layer tests the Device for DRQ bit set to one and BSY bit cleared to zero in preparation for a PIO data FIS transfer.

**Transition HFPDMAQ9:1,** if DRQ bit is set to one and BSY bit is cleared to zero, it shall transition to the HFPDMAQ10: ReceiveRequestSense state.

**Transition HFPDMAQ9:2,** if DRQ bit is cleared to zero or 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 native queued command tags, and flushes pending native commands from the host command queue with system-specific error condition or re-issue pending queued commands

**Transition HFPDMAQ11:1:** After the error flush actions have been completed, it shall transition to the HFPDMAQ11: 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-3 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

#### 13.1.1 Parallel ATA emulation overview

Emulation of parallel ATA device behavior as perceived by the host BIOS or software driver, is a cooperative effort between the device and the Serial ATA host adapter hardware. The behavior of Command and Control Block registers, PIO and DMA data transfers, resets, and interrupts are all emulated.

The host adapter contains a set of registers that shadow the contents of the traditional device registers, referred to as the Shadow Register Block. All Serial ATA devices behave like Device 0 devices. Devices shall ignore the DEV bit in the Device field of received Register Host to Device FISes, and it is the responsibility of the host adapter to gate transmission of Register Host to Device FISes, as appropriate, based on the value of the DEV bit.

After a reset or power-on, the host bus adapter emulates the behavior of a traditional ATA system during device discovery. Immediately after reset, the host adapter shall place the value 7Fh in its Shadow Status register and shall place the value FFh in all the other Shadow Command Block registers (FFFFh in the Data register). In this state the host bus adapter shall not accept writes to the Shadow Command Block Registers. If the Phy detects presence of an attached device, the host bus adapter shall place the value FFh or 80h in the Shadow Status register, and the host bus adapter shall now allow writes to the Shadow Command Block Registers. If a device is present, the Phy shall take no longer than 10 ms to indicate that it has detected the presence of a device, and has set the BSY bit to one in the Shadow Status register. Placing the value 80h in the Shadow Status register is recommended as it provides the highest level of BIOS compatibility.

NOTE 53 - Note that if the BSY bit is set to one in the Shadow Status register all other bits in that register have indeterminate values.

If the attached device establishes communication with the host bus adapter, it shall send a Register Device to Host FIS, resulting in the Shadow Command Block Registers being updated with values appropriate for the attached device.

BIOS and system software writers should be aware of the 10 ms latency the interface may incur in determining device presence, and either ensure the Shadow Status register is read no sooner than 10 ms after initialization or ensure the Shadow Status register is re-read 10 ms after having read a value of 7Fh in order to positively determine presence of a device.

The host adapter may present a Master-only emulation to host software (i.e., each device is a Device 0), and each Device 0 is accessed at a different set of host bus addresses. The host adapter may optionally present a Master/Slave emulation to host software (i.e., two devices on two separate Serial ATA ports are represented to host software as a Device 0 and a Device 1 accessed at the same set of host bus addresses).

#### 13.1.2 Software reset

According to the ATA/ATAPI-6 standard, issuing a software reset is performed by toggling the SRST bit in the Device Control register. The toggle period is stipulated as being no shorter than 5 us. As a result of the SRST bit changing in the Device Control register, Serial ATA host adapters shall issue at least two Register Host to Device FISes (one with the SRST bit set to one and a subsequent one with the SRST bit cleared to zero). See 10.5.5 for a detailed definition of Register Host to Device FIS. Although host software is required to toggle the SRST bit no faster than 5 us, devices may not rely on the inter-arrival time of received Register Host to Device FISes also meeting this timing. Because of flow control, frame handshaking, and other protocol interlocks, devices may effectively receive the resulting Register Host to Device FISes back-to-back.

Due to flow control, protocol interlocks, power management state, or other transmission latencies, the subsequent Register Host to Device FIS transmission clearing the SRST bit to zero during a software reset may be triggered prior to the previous Register Host to Device FIS transmission having been completed. Host adapters are required to allow host software to toggle the SRST bit with the minimum 5 us timing specified in the ATA/ATAPI-6 standard even if frame transmission latencies result in the first Register Host to Device FIS transmission taking longer than 5 us. Host adapters are required to ensure transmission of the two resulting Register Host to Device FISes to the device regardless of the transmission latency of each individual FIS.

### 13.1.3 Master-only emulation

NOTE 54 - Unlike the remainder of this specification, this section is based on the ATA/ATAPI-5 standard.

A native Serial ATA host adapter behaves the same as if a legacy mode master only device were attached with no slave present. It is the responsibility of the host adapter to properly interact with host software and present the correct behavior for this type of configuration. All Serial ATA devices, therefore, need not be aware of master / slave issues and ignore legacy mode task file information that deal with a secondary device. If the DEV bit in the Device register is set to one, selecting the non-existent Device 1, the host adapter shall respond to register reads and writes as specified for a Device 0 with no Device 1 present, as defined in the ATA/ATAPI-5 standard. This includes not setting the BSY bit to one in the Shadow Status register if Device 1 is selected, as described in the ATA/ATAPI-5 standard. If Device 0 is selected, the host adapter shall process the Serial ATA protocols for managing the Shadow Status register contents as defined in later sections.

If Device 0 is selected and the Command register is written in the Shadow Register Block, the host adapter sets the BSY bit to one in its shadow Status register. The host adapter then transmits a Register Host to Device FIS containing the new register contents. If the Device Control register is written in the Shadow Register Block with a change of state of the SRST bit, the host adapter sets the BSY bit to one in its shadow Status register and transmit a Register Host to Device FIS containing the new register contents. Transmission of register contents if the Device Control register is written with any value that is not a change of state of the SRST bit shall not set the BSY bit to one in the shadow Status register, and transmission of a frame to the device containing new register contents is optional. Similarly, the host adapter sets the BSY bit in its shadow Status register to one if a COMRESET is requested or the SRST bit is set to one in the Device Control register. The expected timing for setting BSY bit to one is thereby preserved.

The device updates the contents of the host adapter Shadow Register Block by transmitting a Register Device to Host FIS. This allows the device to set the proper ending status in the host adapter Shadow Register Block at the completion of a command or control request. Specific support is added to ensure proper timing of the DRQ and BSY bits in the Status register for PIO transfers.

Finally the host adapter cooperates with the device by providing an interrupt pending flag in the host adapter. This flag is set by the host adapter if the device sends a Register Device to Host FIS with the Interrupt bit set to one. The host adapter asserts the interrupt to the host processor any time the interrupt pending flag is set, the DEV bit is cleared to zero in the shadow Device register, and the nIEN bit in the shadow Device Control register is cleared to zero. The host adapter clears the interrupt pending flag any time a COMRESET is requested, the SRST bit in the shadow Device Control register is set to one, the shadow Command register is written with DEV bit cleared to zero, or the Shadow Status register is read with DEV bit cleared to zero. This allows the emulation of the host interrupt and its proper timing.



### **13.1.4 Master/Slave emulation (optional)**

#### **13.1.4.1 Master/Slave emulation overview (optional)**

NOTE 55 - Unlike the remainder of this specification, this section is based on the ATA/ATAPI-5 standard.

All devices behave as if they are Device 0 devices. However, the host adapter may optionally implement emulation of the behavior of a Master/Slave configuration by pairing two Serial ATA ports, and managing their associated shadow registers accordingly, as though they were a Device 0 and Device 1 at the same set of host bus addresses.

A host adapter that emulates Master/Slave behavior shall manage the two sets of shadow registers (one set for each of the two devices) based on the value of the DEV bit in the shadow Device register. Based on the value of the DEV bit, the host adapter shall direct accesses to the Shadow Register Block Registers, and accesses to the Control Block Registers, to the appropriate set of shadow registers in the correct device. It is the responsibility of the host adapter to ensure that communication with one or both of the attached devices is handled properly, and that information gets routed to the devices correctly. Each device shall process any communication with the host adapter as if it is targeted for the device regardless of the value of the DEV bit.

If a host adapter is emulating Master/Slave behavior, and there is no device attached to the cable designated as the Device 1 cable, the host adapter shall emulate Device 0 behavior with no Device 1 present as described in the ATA/ATAPI-5 standard.

#### **13.1.4.2 Software reset**

Host adapters that emulate Master/Slave behavior shall emulate proper behavior for software reset. Based on the Phy initialization status, the host adapter knows whether a device is attached to each of the two ports used in a Master/Slave emulation configuration. Device Control register writes, that have the SRST bit set to one, shall result in the associated Shadow Device Control register being written for each port that a device is attached. The frame transmission protocol for each associated port processed, results in a Register Host to Device FIS being transmitted to each attached device. Similarly, the subsequent write to the Shadow Device Control register that clears the SRST bit to zero shall result in a Register Host to Device FIS being sent to each attached device. The host adapter shall then await a response from each attached device (or timeout), and shall merge the contents of the Error and Status registers for the attached devices, in accordance with the ATA/ATAPI-5 standard, to produce the Error and Status register values visible to host software.

#### **13.1.4.3 EXECUTE DEVICE DIAGNOSTICS**

Host adapters that emulate Master/Slave behavior shall emulate proper behavior for EXECUTE DEVICE DIAGNOSTICS. The host adapter shall detect the EXECUTE DEVICE DIAGNOSTIC command being written to the Command register. Detecting the EXECUTE DEVICE DIAGNOSTICS command shall result in the associated shadow register being written for each port that a device is attached. The frame transmission protocol for each associated port processed, results in a Register Host to Device FIS being transmitted to each attached device. The host adapter shall then await response from each attached device (or timeout), and shall merge the contents of the Error and Status registers for the attached devices, in accordance with the ATA/ATAPI-5 standard to produce the Error and Status register values visible to host software.

#### 13.1.4.4 Restrictions and limitations

Superset capabilities that are unique to Serial ATA and not supported by parallel ATA may not be supported in Master/Slave emulation mode. Such capabilities include but are not limited to support for First-party DMA, hot plug/unplug, interface power management, and superset Status and Control registers. Master/Slave emulation is recommended only in configurations where software written for parallel ATA is used and the number of attached devices exceeds the number of Shadow Command Block register interfaces that software supports.

#### 13.1.4.5 Shadow Command Block Register access restrictions

Host software should take measures to ensure that the access restrictions for the Command Block registers (see ATA/ATAPI-5) are observed if accessing the Shadow Command Block registers in order to avoid indeterminate behavior.

Some of these measures include:

- a) prior to writing the Shadow Command Block registers to issue a new command, host software should check the Shadow Status register to verify that both BSY and DRQ bits are cleared to zero; or
- b) if DRQ bit is set to one if software expects it to be cleared to zero (e.g., if ERR bit is set to one), host software should perform an error recovery action (e.g., issue a software reset to the device) prior to writing the Shadow Command register to issue a new command.

NOTE 56 - Issuing a software reset by setting the SRST bit to one in the Device Control register to one while BSY bit or DRQ bit are set to one is legal behavior as described in the Serial ATA state machines, and completion of a software reset ensures that the device and interface are in a known state.

The ATA/ATAPI-5 standard defines restrictions for writing to the Command Block registers.

It shall not write the Shadow Command register if BSY bit or DRQ bit is set to one except for the DEVICE RESET command. The Serial ATA access restrictions differ from parallel ATA where similar access to the Command register is indeterminate. However, the resultant indeterminate behavior may, in some Serial ATA implementations, not be as benign as in some parallel ATA implementations. This is because in Serial ATA the Shadow Command Block registers are cooperatively managed between the host and the device, and the defined coordination is based on the values of BSY and DRQ bits.

EXAMPLE - An example situation where parallel ATA and Serial ATA behavior may differ is if the device sets both DRQ and ERR bits to one in the (Shadow) Status register in a PIO read operation.

In parallel ATA, many device implementations allow software to issue a new command without transferring the data from the device. In Serial ATA, issuing a new command without transferring the data may lead to a hang condition. In Serial ATA, if DRQ bit is set to one there is an associated Data FIS that is being transferred on the Serial ATA interface. Until the Data FIS transfer is completed by the host reading the associated data from the Shadow Data register, the Data FIS remains on the interface in a HOLD<sub>P</sub>/HOLDA<sub>P</sub> flow-controlled condition. While the Data FIS remains on the interface, a new command is unable to be issued to the device.

#### 13.1.4.6 Parallel ATA interoperability state machines

This state machine defines the protocol of the host adapter to emulate Master only parallel ATA devices as seen from the host BIOS or software driver (see Figure 334). The interrupt pending flag (IPF) is an internal state bit in the host adapter that reflects whether or not the device has an interrupt pending to the host.

HA0: HA_SEL/NOINTRQ	The interrupt signal is not asserted to the host.	
1. COMRESET requested by the host.	→	HA_SEL/NOINTRQ
2. Device Control register written by the host with SRST bit cleared to zero and (nIEN bit set to one or IPF bit set to one).	→	HA_SEL/NOINTRQ
3. Device Control register written by the host with SRST bit cleared to zero and (nIEN bit cleared to zero and IPF bit set to one).	→	HA_SEL/INTRQ
4. Device Control register written by the host with SRST bit set to one.	→	HA_SEL/NOINTRQ
5. Command register written by the host.	→	HA_SEL/NOINTRQ
6. Device register written by the host with DEV bit set to one.	→	HA_NOTSEL
7. Any other register read or write by the host	→	HA_SEL/NOINTRQ
8. Transport layer indicates new register content from the device with the IPF bit cleared to zero.	→	HA_SEL/NOINTRQ
9. Transport layer indicates new register content from the device with the IPF bit set to one and nIEN bit to one.	→	HA_SEL/NOINTRQ
10. Transport layer indicates new register content from the device with the IPF bit set to one and nIEN bit cleared to zero.	→	HA_SEL/INTRQ

HA1: HA_SEL/INTRQ	The interrupt signal is asserted to the host.	
1. COMRESET requested by the host.	→	HA_SEL/NOINTRQ
2. Device Control register written by the host with SRST bit cleared to zero and (nIEN bit set to one or IPF bit cleared to zero).	→	HA_SEL/NOINTRQ
3. Device Control register written by the host with SRST bit cleared to zero and (nIEN bit cleared to zero and IPF bit set to one).	→	HA_SEL/INTRQ
4. Device Control register written by the host with SRST bit set to one.	→	HA_SEL/NOINTRQ
5. Command register written by the host.	→	HA_SEL/NOINTRQ
6. Device register written by the host with DEV bit set to one.	→	HA_NOTSEL
7. Status register read by the host	→	HA_SEL/NOINTRQ
8. Any other register write by the host	→	HA_SEL/INTRQ
9. Any other register read by the host	→	HA_SEL/INTRQ
10. Transport layer indicates new register content from the device.	→	HA_SEL/INTRQ

**Figure 334 – Host application layer Parallel ATA emulation state machine**

(part 1 of 2)

HA2: HA_NOTSEL	The interrupt signal is not asserted to the host.	
1. COMRESET requested by the host.	→	HA_SEL/NOINTRQ
2. Device Control register written by the host with SRST bit cleared to zero.	→	HA_NOTSEL
3. Device Control register written by the host with SRST bit set to one.	→	HA_SEL/NOINTRQ
4. Command register written by the host with command other than EXECUTE DEVICE DIAGNOSTIC.	→	HA_NOTSEL
5. Command register written by the host with command EXECUTE DEVICE DIAGNOSTIC.	→	HA_SEL/NOINTRQ
6. Device register written by the host with DEV bit cleared to zero and (nIEN bit set to one or IPF bit cleared to zero).	→	HA_SEL/NOINTRQ
7. Device register written by the host with DEV bit cleared to zero and (nIEN bit cleared to zero and IPF bit set to one).	→	HA_SEL/INTRQ
8. Any other register write by the host	→	HA_NOTSEL
9. Read of Status or Alternate Status register by the host.	→	HA_NOTSEL
10. Read of any other register by the host.	→	HA_NOTSEL
11. Transport layer indicates new register content from the device.	→	HA_NOTSEL

**Figure 334 – Host application layer Parallel ATA emulation state machine (part 2 of 2)**

**HA0: HA\_SEL/NOINTRQ state**, this state is entered if Device 0 is selected and either IPF bit is cleared to zero or nIEN bit is set to one.

If in this state, the interrupt signal to the host shall not be asserted.

**Transition HA0:1**, if a COMRESET is requested by the host, the host adapter shall set BSY bit to one in the shadow Status register, clear IPF bit to zero, notify the Link layer to have a bus COMRESET asserted, and make a transition to the HA0: HA\_SEL/NOINTRQ state.

**Transition HA0:2**, if the Device Control register is written by the host with SRST bit cleared to zero and either nIEN bit set to one or IPF bit is cleared to zero, the host adapter shall notify the Transport layer to send a Register Host to Device FIS with C bit cleared to zero with the current content of the shadow registers, and make a transition to the HA0: HA\_SEL/NOINTRQ state.

**Transition HA0:3**, if the Device Control register is written by the host with SRST bit cleared to zero, nIEN bit cleared to zero, and IPF bit is set to one, the host adapter shall notify the Transport layer to send a Register Host to Device FIS with C bit cleared to zero with the current content of the shadow registers, and make a transition to the HA1: HA\_SEL/INTRQ state.

**Transition HA0:4**, if the Device Control register is written by the host with SRST bit set to one, the host adapter shall set BSY bit to one in the Shadow Status register, clear IPF bit to zero, notify the Transport layer to send a Register Host to Device FIS with C bit cleared to zero with the current content of the shadow registers, and make a transition to the HA0: HA\_SEL/NOINTRQ state.

**Transition HA0:5**, if the Command register is written by the host, the host adapter shall set BSY bit to one in the Shadow Status register, clear IPF bit to zero, notify the Transport layer to send a

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

Register Host to Device FIS with C bit set to one with the current content of the shadow registers, and make a transition to the HA0: HA\_SEL/NOINTRQ state.

**Transition HA0:6**, if the Device register is written by the host with DEV bit set to one, the host adapter shall make a transition to the HA2: HA\_NOTSEL state.

**Transition HA0:7**, if any register is read or written by the host other than those described above, the host adapter shall make a transition to the HA0: HA\_SEL/NOINTRQ state.

**Transition HA0:8**, if the Transport layer indicates new register content from the device with IPF bit cleared to zero, the host adapter shall place the new register content into the shadow registers and make a transition to the HA0: HA\_SEL/NOINTRQ state.

**Transition HA0:9**, if the Transport layer indicates new register content from the device with IPF bit set to one and nIEN bit set to one, the host adapter shall set IPF bit to one, place the new register content into the shadow registers, and make a transition to the HA0: HA\_SEL/NOINTRQ state.

**Transition HA0:10**, if the Transport layer indicates new register content from the device with IPF bit set to one and nIEN bit is cleared to zero, the host adapter shall set IPF bit to one, place the new register content into the shadow registers, and make a transition to the HA1: HA\_SEL/INTRQ state.

**HA1: HA\_SEL/INTRQ state**, this state is if Device 0 is selected, nIEN bit is cleared to zero, and IPF bit is set to one.

If in this state, the interrupt signal to the host shall be asserted.

**Transition HA1:1**, if a COMRESET is requested by the host, the host adapter shall set BSY bit to one in the Shadow Status register, clear IPF bit to zero, notify the Link layer to have a bus COMRESET asserted, and make a transition to the HA0: HA\_SEL/NOINTRQ state.

**Transition HA1:2**, if the Device Control register is written by the host with SRST bit cleared to zero, with nIEN bit set to one or IPF bit cleared to zero, the host adapter shall notify the Transport layer to send a Register Host to Device FIS with C bit cleared to zero with the current content of the shadow registers, and make a transition to the HA0: HA\_SEL/NOINTRQ state.

**Transition HA1:3**, if the Device Control register is written by the host with SRST bit cleared to zero with nIEN bit cleared to zero and IPF bit set to one, the host adapter shall notify the Transport layer to send a Register Host to Device FIS with C bit cleared to zero with the current content of the shadow registers, and make a transition to the HA1: HA\_SEL/INTRQ state.

**Transition HA1:4**, if the Device Control register is written by the host with SRST bit set to one, the host adapter shall set BSY bit to one in the Shadow Status register, clear IPF bit to zero, notify the Transport layer to send a Register Host to Device FIS with C bit cleared to zero with the current content of the shadow registers, and make a transition to the HA0: HA\_SEL/NOINTRQ state.

**Transition HA1:5**, if the Command register is written by the host, the host adapter shall set BSY bit to one in the Shadow Status register, clear IPF bit to zero, notify the Transport layer to send a Register Host to Device FIS with C bit set to one with the current content of the shadow registers, and make a transition to the HA0: HA\_SEL/NOINTRQ state.

**Transition HA1:6**, if the Device register is written by the host with DEV bit set to one, the host adapter shall make a transition to the HA2: HA\_NOTSEL state.

**Transition HA1:7:** If the Status register is read by the host, the host adapter shall clear IPF bit to zero and make a transition to the HA0: HA\_SEL/NOINTRQ state.

**Transition HA1:8,** if any register is written by the host other than those described above, the host adapter shall make a transition to the HA1: HA\_SEL/INTRQ state.

**Transition HA1:9,** if any register is read by the host other than that described above, the host adapter shall make a transition to the HA1: HA\_SEL/INTRQ state.

**Transition HA1:10,** if the Transport layer indicates new register content from the device, the host adapter shall place the new register content into the shadow registers and make a transition to the HA1: HA\_SEL/INTRQ state.

**HA2: HA\_NOTSEL state,** this state is entered if Device 1 is selected.

If in this state, the interrupt signal to the host shall not be asserted.

**Transition HA2:1,** if a COMRESET is requested by the host, the host adapter shall set BSY bit to one in the Shadow Status register, clear DEV bit to zero in the Shadow Device register, clear IPF bit to zero, notify the Link layer to have a bus COMRESET asserted, and make a transition to the HA0: SEL/NOINTRQ state.

**Transition HA2:2,** if the Device Control register is written by the host with SRST bit cleared to zero, the host adapter shall notify the Transport layer to send a Register Host to Device FIS with C bit cleared to zero with the current content of the shadow registers, and make a transition to the HA2: HA\_NOTSEL state.

**Transition HA2:3,** if the Device Control register is written by the host with SRST bit set to one, the host adapter shall set BSY bit to one in the Shadow Status register, clear DEV bit to zero in the Shadow Device register, clear IPF bit to zero, notify the Transport layer to send a Register Host to Device FIS with C bit cleared to zero with the current content of the shadow registers, and make a transition to the HA0: HA\_SEL/NOINTRQ state.

**Transition HA2:4,** if the Command register is written by the host with any command code other than EXECUTE DEVICE DIAGNOSTIC, the host adapter shall not set BSY bit in the Shadow Status register and shall make a transition to the HA2: HA\_NOTSEL state.

**Transition HA2:5,** if the Command register is written by the host with the EXECUTE DEVICE DIAGNOSTIC command code, the host adapter shall set BSY bit to one in the Shadow Status register, clear DEV bit to zero in the Device register, clear IPF bit to zero, notify the Transport layer to send a Register Host to Device FIS with C bit set to one with the current content of the shadow registers, and make a transition to the HA0: HA\_SEL/INTRQ state.

**Transition HA2:6,** if the Device register is written by the host with DEV bit cleared to zero and either nIEN bit set to one or IPF bit cleared to zero, the host adapter shall make a transition to the HA0: HA\_SEL/NOINTRQ state.

**Transition HA2:7,** if the Device register is written by the host with DEV bit cleared to zero, nIEN bit cleared to zero, and IPF bit set to one, the host adapter shall make a transition to the HA1: HA\_SEL/INTRQ state.

**Transition HA2:8,** if any register is written by the host other than those described above, the host adapter shall make a transition to the HA2: HA\_NOTSEL state.

**Transition HA2:9,** if the Status or Alternate Status register is read by the host, the host shall return register content 00h to the host and make a transition to the HA2: HA\_NOTSEL state.

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

**Transition HA2:10**, if any register is read by the host other than that described above, the host adapter shall return the current shadow register content to the host and make a transition to the HA2: HA\_NOTSEL state.

**Transition HA2:11**, if the Transport layer indicates new register content from the device, the host adapter shall place the new register content into the shadow registers and make a transition to the HA2: HA\_NOTSEL state.

## **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-3 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 100.

**Table 100 – IDENTIFY DEVICE information**

(part 1 of 4)

Word	O/M	F/V	
0..46			Set as indicated in ACS-3
47	M	F R	Multiple Count 15..8 80h 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-3
49	M	F F	Capabilities 15..12 Set as indicated in ACS-3 11 Shall be set to one 10 Shall be set to one 9..0 Set as indicated in ACS-3
50..52			Set as indicated in ACS-3
53	M	R F F F	Field validity 15..3 Reserved 2 1=the fields reported in Word 88 are valid 0=the fields reported in Word 88 are not valid 1 1=the fields reported in Words (70:64) are valid 0=the fields reported in Words (70:64) are not valid 0 Obsolete
54..62			Set as indicated in ACS-3
63	M	F F F	Multiword DMA transfer 15..3 Set as indicated in ACS-3 2 Multiword DMA mode 2 and below are supported 1 Multiword DMA mode 1 and below are supported 0 Multiword DMA mode 0 is supported
64	M	F	PIO transfer modes supported 15..2 Set as indicated in ACS-3 1..0 PIO modes 3 and 4 supported
65	M	F	Minimum Multiword DMA transfer cycle time per Word 15..0 Cycle time in nanoseconds
66	M	F	Manufacturer's recommended Multiword DMA transfer cycle time 15..0 Cycle time in nanoseconds
67	M	F	Minimum PIO transfer cycle time without flow control 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 zero.

**Table 100 – IDENTIFY DEVICE information (part 2 of 4)**

Word	O/M	F/V	
68	M	F	Minimum PIO transfer cycle time with IORDY flow control 15..0 Cycle time in nanoseconds
69..74			Set as indicated in ACS-3
75	O	R F	Queue depth 15..5 Reserved 4..0 Maximum queue depth – 1
76	O	F F F F F F F R F F F F	Serial ATA capabilities 15 Supports READ LOG DMA EXT as equivalent to READ LOG EXT 14 Supports Device Automatic Partial to Slumber transitions 13 Supports Host Automatic Partial to Slumber transitions 12 Supports Native Command Queuing priority information 11 Supports Unload while NCQ commands outstanding 10 Supports Phy event counters 9 Supports receipt of host-initiated interface power management requests 8 Supports Native Command Queuing 7..4 Reserved for future Serial ATA signaling speed grades 3 Supports Serial ATA Gen3 signaling speed (6.0 Gbps) 2 Supports Serial ATA Gen2 signaling speed (3.0 Gbps) 1 Supports Serial ATA Gen1 signaling speed (1.5 Gbps) 0 Shall be cleared to zero
77	O	R F F F F V F	Serial ATA Additional capabilities 15..8 Reserved 7 DevSleep_to_ReducedPwrState 6 Supports RECEIVE FPDMA QUEUED and SEND FPDMA QUEUED commands 5 Supports NCQ NON-DATA Command 4 Supports NCQ Streaming 3..1 Coded value indicating current negotiated Serial ATA signal speed 0 Shall be cleared to zero
<p>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 zero.</p>			

**Table 100 – IDENTIFY DEVICE information (part 3 of 4)**

Word	O/M	F/V	
78	O		Serial ATA features supported
		R	15..12 Reserved
		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
		R	10 Reserved
		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-3
88			15..6 Set as indicated in ACS-3
		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-3
93		V	COMRESET result. The contents of this Word shall be cleared to zero.
94..221			Set as indicated in ACS-3
<p>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 zero.</p>			

**Table 100 – IDENTIFY DEVICE information (part 4 of 4)**

Word	O/M	F/V		
222			Transport Major Revision 0000h or FFFFh = device does not report version	
		F	<b>Bits</b> 15:12	<b>Description</b> Transport Type 0h = Parallel 1h = Serial 2h .. Fh = Reserved
		R	11: 8	Reserved
		F	7	SATA Rev 3.2
		F	6	SATA Rev 3.1
		F	5	SATA Rev 3.0
		F	4	SATA Rev 2.6
		F	3	SATA Rev 2.5
		F	2	SATA II: Extensions
		F	1	SATA 1.0a
		F	0	ATA8-AST
223		F	Transport Minor Revision	
224..255			Set as indicated in ACS-3	
<p>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 zero.</p>				

### 13.2.2.2 Word 0..46

Word 0..46 shall be set as indicated in ACS-3.

### 13.2.2.3 Word 47, multiword PIO transfer

Bits 15:8 of Word 47 shall be set as indicated in ACS-3.

Bits 7 through 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-3.

### 13.2.2.5 Word 49, capabilities

Bits 15:12 of Word 49 shall be set as indicated in ACS-3.

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-3.

#### **13.2.2.6 Words 50..52**

Word 50..52 shall be set as indicated in ACS-3.

#### **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 through 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 through 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 Word 54..62**

Word 54..62 shall be set as indicated in ACS-3.

#### **13.2.2.9 Word 63, Multiword DMA transfer**

Bits 15..3 shall be set as indicated in ACS-3.

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-3.

#### **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**

Word 69..74 shall be set as indicated in ACS-3.

#### **13.2.2.16 Word 75, queue depth**

This Word is as defined in the ACS-3 standard. The Native Command Queuing 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 device reporting a value in this field of 15, corresponding to a maximum of 16

outstanding commands, the host shall never use 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 1..3. 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 through 79 shall be valid. If this field is 0000h or FFFFh the device does not claim compliance with this specification and words 76 through 79 are not valid and shall be ignored.

Bit 15 is a copy of READ LOG DMA EXT AS EQUIVALENT TO READ LOG EXT SUPPORTED (see 13.7.9.2.11).

Bit 14 is a copy of DEVICE AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED (see 13.7.9.2.10).

Bit 13 is a copy of HOST AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED (see 13.7.9.2.9).

Bit 12 is a copy of NCQ PRIORITY INFORMATION SUPPORTED (see 13.7.9.2.8).

Bit 11 is a copy of UNLOAD WHILE NCQ COMMANDS ARE OUTSTANDING SUPPORTED (see 13.7.9.2.7).

Bit 10 is a copy of SATA PHY EVENT COUNTERS LOG SUPPORTED (see 13.7.9.2.6).

Bit 9 is a copy of RECEIPT OF HOST INITIATED POWER MANAGEMENT REQUESTS SUPPORTED (see 13.7.9.2.5).

Bit 8 is a copy of NCQ FEATURE SET SUPPORTED (see 13.7.9.2.4).

Bits (7:4) are reserved.

Bit 3 is a copy of SATA GEN3 SIGNALING SPEED SUPPORTED (see 13.7.9.2.3).

Bit 2 is a copy of SATA GEN2 SIGNALING SPEED SUPPORTED (see 13.7.9.2.2).

Bit 1 is a copy of SATA GEN1 SIGNALING SPEED SUPPORTED (see 13.7.9.2.1).

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 zero indicating the device has no support for Serial ATA additional capabilities.

Bits 15:8 are reserved.

Bit 7 is a copy of DEVSLEEP\_TO\_REDUCEDPWRSTATE CAPABILITY SUPPORTED (see 13.7.9.2.23).

Bit 6 is a copy of SEND AND RECEIVE QUEUED COMMANDS SUPPORTED (see 13.7.9.2.14).

Bit 5 is a copy of NCQ NON-DATA COMMAND SUPPORTED (see 13.7.9.2.13).

Bit 4 is a copy of NCQ STREAMING SUPPORTED (see 13.7.9.2.12).

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

Bits (3:1) are a copy of CURRENT NEGOTIATED SERIAL ATA SIGNAL SPEED (see 13.7.9.3.1).

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 15:12 are reserved.

Bit 11 is a copy of REBUILD ASSIST SUPPORTED (see 13.7.9.2.26)

Bit 10 is a copy of DIPM SSP PRESERVATION SUPPORTED (see 13.7.9.2.25)

Bit 9 is a copy of HYBRID INFORMATION SUPPORTED (see 13.7.9.2.24).

Bit 8 is a copy of DEVICE SLEEP SUPPORTED (see 13.7.9.2.22).

Bit 7 is a copy of NCQ AUTONSENSE SUPPORTED (see 13.7.9.2.21).

Bit 6 is a copy of SOFTWARE SETTINGS PRESERVATION SUPPORTED (see 13.7.9.2.20).

Bit 5 is a copy of HARDWARE FEATURE CONTROL SUPPORTED (see 13.7.9.2.19).

Bit 4 is a copy of IN-ORDER DATA DELIVERY SUPPORTED (see 13.7.9.2.18).

Bit 3 is a copy of DEVICE INITIATED POWER MANAGEMENT SUPPORTED (see 13.7.9.2.17).

Bit 2 is a copy of DMA SETUP AUTO-ACTIVATION SUPPORTED (see 13.7.9.2.16).

Bit 1 is a copy of NON-ZERO BUFFER OFFSETS SUPPORTED (see 13.7.9.2.15).

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 REBUILD ASSIST ENABLED bit (see 13.7.9.3.11).

Bit 10 is reserved.

Bit 9 is a copy of HYBRID INFORMATION ENABLED (see 13.7.9.3.10)

Bit 8 is a copy of DEVICE SLEEP ENABLED (see 13.7.9.3.9).

Bit 7 is a copy of AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS ENABLED (see 13.7.9.3.8).

Bit 6 is a copy of SOFTWARE SETTINGS PRESERVATION ENABLED (see 13.7.9.3.7).

Bit 5 is a copy of HARDWARE FEATURE CONTROL ENABLED (see 13.7.9.3.6).

Bit 4 is a copy of IN-ORDER DATA DELIVERY ENABLED (see 13.7.9.3.5).

Bit 3 is a copy of DEVICE INITIATED POWER MANAGEMENT ENABLED (see 13.7.9.3.4).

Bit 2 is a copy of DMA SETUP FIS AUTO-ACTIVATE ENABLED (see 13.7.9.3.3).

Bit 1 is a copy of NON-ZERO BUFFER OFFSETS ENABLED (see 13.7.9.3.2).

Bit 0 shall be cleared to zero.

#### **13.2.2.21 Words 80..87**

Words 80..87 shall be set as indicated in ACS-3.

#### **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-3.

#### **13.2.2.23 Words 89..92**

Words 89..92 shall be set as indicated in ACS-3.

#### **13.2.2.24 Word 93, hardware configuration test results**

Word 93 shall be set 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-3.

#### **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-3.

#### **13.2.2.28 Words 224..255**

Words 224..255 shall be set as indicated in ACS-3.



### 13.2.3 IDENTIFY PACKET DEVICE

#### 13.2.3.1 IDENTIFY PACKET DEVICE information

The IDENTIFY PACKET DEVICE information structure is defined in Table 101.

**Table 101 – IDENTIFY PACKET DEVICE information**

(part 1 of 3)

Word	O/M	F/V	
0..48			Set as indicated in ACS-3
49	M		Capabilities
		F	15..12 Set as indicated in ACS-3
		F	11 Shall be set to one
		F	10 Shall be set to one
			9..0 Set as indicated in ACS-3
50..52			Set as indicated in ACS-3
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-3
63	M		Multiword DMA transfer
		F	15..3 Set as indicated in ACS-3
		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
		F	15..2 Set as indicated in ACS-3
		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-3
<p>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 zero.</p>			

**Table 101 – IDENTIFY PACKET DEVICE information (part 2 of 3)**

Word	O/M	F/V	
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 Gbps)
		F	2 Supports Serial ATA Gen2 signaling speed (3.0 Gbps)
		F	1 Supports Serial ATA Gen1 signaling speed (1.5 Gbps)
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..7 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-3
<p>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 zero.</p>			

**Table 101 – IDENTIFY PACKET DEVICE information (part 3 of 3)**

Word	O/M	F/V		
88		F	15..6	Set as indicated in ACS-3
		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-3	
93		V	COMRESET result. The contents of this Word shall be cleared to zero.	
94..221			Set as indicated in ACS-3	
222			Transport Major Revision 0000h or FFFFh = device does not report version	
		F	<b>Bits</b> 15:12	<b>Description</b> Transport Type 0h = Parallel 1h = Serial 2h .. Fh = Reserved
		R	11: 8	Reserved
		F	7	SATA Rev 3.2
		F	6	SATA Rev 3.1
		F	5	SATA Rev 3.0
		F	4	SATA Rev 2.6
		F	3	SATA Rev 2.5
		F	2	SATA II: Extensions
		F	1	SATA 1.0a
		F	0	ATA8-AST
223		F	Transport Minor Revision	
224..255			Set as indicated in ACS-3	
<p>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 zero.</p>				

### 13.2.3.2 Word 0..48

Words 0..48 shall be set as indicated in ACS-3.

### 13.2.3.3 Word 49, capabilities

Bits 15:12 of Word 49 shall be set as indicated in ACS-3.

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-3.

#### **13.2.3.4 Words 50..52**

Words 50..52 shall be set as indicated in ACS-3.

#### **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 through 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 through 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-3.

#### **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-3.

#### **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-3.

#### **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-3.

#### **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 zero indicating the device has no support for Serial ATA additional capabilities.

Bit 0 shall be cleared to zero

Bit 1..3 are a coded value to indicate the current Serial ATA phy speed that device is communicating at. Table 119 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 57 - 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.

Bit 6..15 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.

Bit 1..2 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 and 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 either host-initiated interface power management or device-initiated interface power management. 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 and 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.

Bit 7..15 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-3.

#### **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-3.

#### **13.2.3.20 Word 89..92**

Word 89..92 shall be set as indicated in ACS-3.

#### **13.2.3.21 Word 93, hardware configuration test results**

Word 93 shall be set 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-3.

#### **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-3.

#### **13.2.3.25 Words 224..255**

Words 224..255 shall be set as indicated in ACS-3.

#### **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

### 13.3.1 SET FEATURES overview

Devices are informed of host capabilities and have optional features enabled/disabled through the SET FEATURES command defined in the ACS-3 standard. Serial ATA features are controlled using a features value as defined in Table 102.

**Table 102 – Features enable/disable values**

Features(7:0) Value	Description
10h	Enable use of Serial ATA feature
90h	Disable use of Serial ATA feature

Count(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 103.

**Table 103 – Feature identification values**

Count(7:0) Value	Description
00h	Reserved
01h	Non-zero buffer offset in DMA Setup FIS
02h	DMA Setup FIS Auto-Activate optimization
03h	Device-initiated interface power state transitions
04h	Guaranteed In-Order Data Delivery
05h	Asynchronous Notification
06h	Software Settings Preservation
07h	Device Automatic Partial to Slumber transitions
08h	Enable Hardware Feature Control
09h	Enable Device Sleep
0Ah	Enable/Disable Hybrid Information
0Bh..FFh	Reserved for future Serial ATA definition

### 13.3.2 Enable/disable non-zero offsets in DMA Setup

A Count(7:0) value of 01h is used by the host to enable or disable non-zero buffer offsets in the DMA Setup FIS if the device utilizes the First-party DMA mechanism (see 13.6.2.2). By default, non-zero buffer offsets in the DMA Setup FIS are disabled. Enabling non-zero buffer offsets in the DMA Setup FIS 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 non-zero offsets in DMA Setup FISes shall be preserved across software reset. The enable/disable state for non-zero offsets in DMA Setup FISes shall be reset to its default state upon COMRESET.

### 13.3.3 Enable/disable DMA Setup FIS auto-activate optimization

A Count(7:0) value of 02h is used by the host to enable or disable the DMA Setup FIS optimization 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 host bus adapter 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**

A Count(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<sub>P</sub> or PMREQ\_S<sub>P</sub> 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 13.7.9.2.25), 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**

A Count(7:0) value of 04h is used by the host to enable or disable guaranteed in-order data delivery if the device utilizes the First-party DMA mechanism and non-zero buffer offsets in the DMA Setup FIS. By default, guaranteed in-order data delivery is disabled. Enabling guaranteed in-order data delivery 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 guaranteed in-order data delivery shall be preserved across software reset. The enable/disable state for guaranteed in-order data delivery shall be reset to its default state upon COMRESET.

### **13.3.6 Enable/disable asynchronous notification**

A Count(7:0) value of 05h is used by the host to enable or disable asynchronous notification. By default, asynchronous notification is disabled. The host may enable asynchronous notification in order to allow the device to request attention without the host polling.

NOTE 58 - This may be useful to avoid polling for media change events in ATAPI devices.

The enable/disable state for asynchronous notification shall be preserved across software reset. The enable/disable state for asynchronous notification shall be reset to its default state upon COMRESET.

### **13.3.7 Enable/disable software settings preservation**

A Count(7:0) value of 06h is used by the host to enable or disable software settings preservation, 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 software settings preservation shall persist across software reset. The enable/disable state for software settings preservation shall be reset to its default state upon COMRESET. The host may disable software settings preservation 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

A Count(7:0) value of 07h is used by the host to enable or disable Device Automatic Partial to Slumber transitions. By default, if the device supports Device Automatic Partial to Slumber transitions the feature is disabled on power-up. The enable/disable state for Device Automatic Partial to Slumber transitions shall persist across software reset. The enable/disable state for Automatic Partial to Slumber transitions shall be reset to its default state upon COMRESET.

Device Automatic Partial to Slumber transitions shall not be enabled if Device-Initiated Interface Power State transitions are disabled. Attempting to enable Automatic Partial to Slumber transitions while Device-Initiated Interface Power State transitions are 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(7:0) value of 08h is used by the host to enable Hardware Feature Control. See 13.19 for additional information about Hardware Feature Control.

Hardware Feature Control shall be disabled by power-on reset.

LBA(15:0) contains a function identifier (see Table 104).

**Table 104 – Hardware Feature Control pin definitions**

LBA(15:0)	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:

- Current Hardware Feature Control Identifier (see 13.7.9) shall be set to the value in LBA(15:0);
- IDENTIFY DEVICE data Word 79, bit 5 shall be set to one; and
- the behavior of Hardware Feature Control is specified by Table 104.

The device shall return command aborted if:

- IDENTIFY DEVICE data Word 78 bit 5 is cleared to zero;
- the value in LBA(15:0) is not equal to the Supported Hardware Feature Control Identifier (see 13.7.9); or
- the Current Hardware Feature Control Identifier (see 13.7.9) is non-zero.

### 13.3.10 Enable/disable Device Sleep

A Count(7:0) value of 09h is used by the host to enable or disable Device Sleep. If the value in Features(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.

### **13.3.11 Enable/disable Hybrid Information**

#### **13.3.11.1 Enable/disable Hybrid Information 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 13.7.7.2.3) 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 13.7.7.2.3) in the Hybrid Information log to FFh;
- c) increment the Enable Count field (see 13.7.7.2.14) 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 Feature 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) set the Enabled field (see 13.7.7.2.3) 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.

### **13.4 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-3.

## 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 subcommand code 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-3, and was last defined in ATA/ATAPI-5.

NCQ NON-DATA (DEADLINE HANDLING) (see 13.6.6.4), the state of WDNC and RDNC.

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-3).

Security mode state, the security mode state established by Security Mode feature set commands (refer to ACS-3). 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(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 Initiated 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-3.

NOTE 59 - Note that these recommendations do not apply for external SATA devices.

## **13.6 Native Command Queuing (optional)**

### **13.6.1 Native Command Queuing overview**

This section defines a simple and streamlined command queuing model for Serial ATA.

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.

Devices supporting Native Command Queuing shall implement, and report support for, the general purpose logging feature set as defined in ACS-3. In addition, the device shall implement the Queued Error Log.

### **13.6.2 Native Command Queuing (NCQ) 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 through 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. As described in 14.2.5 the SActive register and the access conventions for it.

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 Setup FIS 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 as 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 60 - 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 61 - 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



HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

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 335 (see 10.5.9).

0	Reserved (0)	Reserved (0)	A	I	D	Reserved (0)	FIS Type (41h)
1	0h						TAG
2	0h						
3	Reserved (0)						
4	DMA Buffer Offset						
5	DMA Transfer Count						
6	Reserved (0)						

Figure 335 – DMA Setup FIS definition for memory buffer selection

Field Definitions

FIS Type As defined in 10.5.9.

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).

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 zero.

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 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 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 1. 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. According to 10.5.12.3 for special considerations if the transfer

count is for an odd number of Words. Devices shall not set this field to 0h; a value of 0h for this field is illegal and results in indeterminate behavior.

I Interrupt Native Command Queuing 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.

R/Reserved All reserved fields shall be cleared to zero.

### 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 native command queuing (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 PRIO bit for NCQ commands (i.e., READ FPDMA QUEUED, WRITE FPDMA QUEUED, RECEIVE FPDMA QUEUED, and SEND FPDMA QUEUED). 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.

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 105).

**Table 105 – Priority**

<b>Prio(1:0)</b>	<b>Description</b>
00b	Normal Priority
01b	Isochronous – deadline dependent priority
10b	High priority
11b	Reserved

### **13.6.2.5 Unload**

If using Native Command Queuing 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-3.

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 and 13.7). 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 sequent 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-Native Queued Commands and Native Queued Commands**

#### **13.6.3.1 Intermixing Non-Native Queued Commands and Native Queued Commands scope**

The host shall not issue a non-native queued command while a native queued command is outstanding. Upon receiving a non-native queued command while a native queued command is outstanding, the device shall signal the error condition to the host by transmitting a Register Device to Host FIS with the ERR and ABRT bits 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.

Non-native queued commands include all commands other than:

- a) READ FPDMA QUEUED (see 13.6.4);
- b) WRITE FPDMA QUEUED (see 13.6.5);
- c) NCQ NON-DATA (see 13.6.6);
- d) RECEIVE FPDMA QUEUED (see 13.6.7); and
- e) SEND FPDMA QUEUED (see 13.6.8).

Reception of a command to read the Queued Error Log (see 13.7) after an error has occurred shall cause any outstanding Serial ATA native queued 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 queued commands regardless of any previous errors on a queued command.

In the case that a command to read the Queued Error Log is issued while a native queued 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-native queued command if a native queued command is outstanding. In this case, a subsequent command to read the Queued Error Log is required to recover from the error.

### **13.6.3.2 Intermixing Non-Native Queued Commands and Native Queued Commands overview**

NCQ commands consist of the following:

- a) READ FPDMA QUEUED;
- b) WRITE FPDMA QUEUED;
- c) NCQ NON-DATA;
- d) RECEIVE FPDMA QUEUED; and
- e) 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.

NCQ NON-DATA, RECEIVE FPDMA QUEUED and SEND FPDMA QUEUED contain subcommands.

### 13.6.4 READ FPDMA QUEUED

#### 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 336.

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					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							
Auxiliary(15:8)	Reserved							
Auxiliary(23:16)	Hybrid Information							
Auxiliary(31:24)	Reserved							
Device(7:0)	FUA	1	Res	0	Reserved			
Command(7:0)	60h							

**Figure 336 – READ FPDMA QUEUED command definition**

**TAG** The TAG 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 Priority (PRIO) 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 105).

**ICC** The Isochronous Command Completion (ICC) field is valid if PRIO 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 ICC.

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 ICC Bit 7 is cleared to zero, then:

- a) the time interval is fine-grained;
- b) interval = 10 ms;
- c) time limit = (ICC(6:0) + 1) × 10 ms; and
- d) max fine time = 128 × 10 ms = 1.28 s.

If ICC 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(6:0) + 1) × 0.5 s; and
- d) max coarse time = 128 × 0.5 s = 64 s.

**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.

**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.

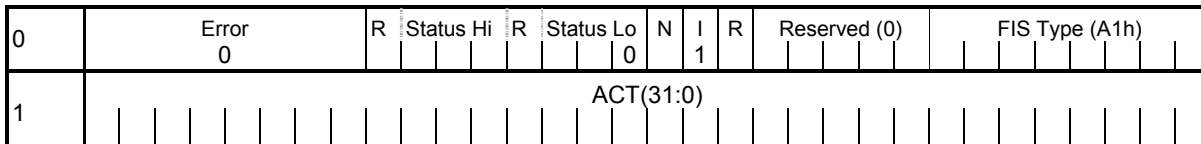
**RARC** If the Rebuild Assist feature is not supported, then the RARC bit shall be ignored.  
 If the Rebuild Assist feature is supported and is disabled, then the RARC bit shall be ignored.  
 If the Rebuild Assist feature is supported and enabled, then the RARC bit specifies that read operations shall be processed as defined in 13.21.

**Others** All other registers have contents consistent with the READ DMA QUEUED EXT command defined in the ACS-3 standard, including the Sector Count(15:0) convention where a value of zero specifies that 65 536 sectors are to be transferred.

### 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 337.



**Figure 337 – Set Device Bits FIS for successful READ FPDMA QUEUED command completion**

**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.

**Error** The Error register shall be cleared to zero.

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

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.

All other fields as defined in section 10.5.7.

NOTE 62 - 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.

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 338).

Field	7	6	5	4	3	2	1	0
Error(7:0)	ERROR							
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 338 – READ FPDMA QUEUED error on command receipt**

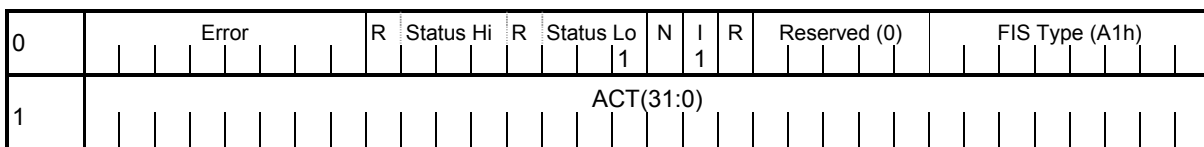
ERROR ATA error code for the failure condition of the failed command  
 BSY 0  
 DRDY 1  
 DF 0  
 DRQ 0  
 ERR 1

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) 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 339) 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 339 – Set Device Bits FIS with error notification, and command completions**

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.

Error The Error register shall contain the ATA error code.

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.



HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

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.



### 13.6.5 WRITE FPDMA QUEUED

#### 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 341.

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					Reserved		
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							
Auxiliary(15:8)	Reserved							
Auxiliary(23:16)	Hybrid Information							
Auxiliary(31:24)	Reserved							
Device(7:0)	FUA	1	0	0	Reserved			
Command(7:0)	61h							

**Figure 341 – WRITE FPDMA QUEUED command definition**

**TAG** The TAG 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 Priority (PRIO) 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 105).

**ICC** The Isochronous Command Completion (ICC) field is valid if PRIO 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 ICC.

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 ICC Bit 7 is cleared to zero, then

- a) the time interval is fine-grained;
- b) interval = 10 ms;
- c) time Limit = (ICC(6:0) + 1) × 10 ms;
- d) max Fine Time = 128 × 10 ms = 1.28 s.

If ICC 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(6:0) + 1) × 0.5 s;
- d) max Coarse Time = 128 × 0.5 s = 64 s.

**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.

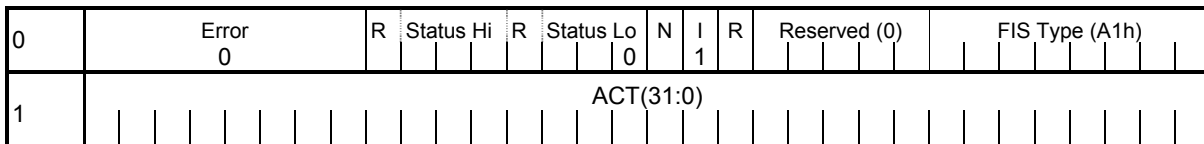
**Hybrid Information** If the Hybrid Information feature is supported (i.e., IDENTIFY DEVICE data Word 78 bit 9 is set to one) and is enabled (i.e., IDENTIFY DEVICE data Word 79 bit 9 is set to one), then the Hybrid Information field shall be processed as defined in 13.20). If the Hybrid Information 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.

**Others** All other registers have contents consistent with the WRITE DMA QUEUED EXT command defined in the ACS-3 standard, including the Sector Count(15:0) convention where a value of zero specifies that 65 536 sectors are to be transferred.

### 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 342.



**Figure 342 – Set Device Bits FIS for successful WRITE FPDMA QUEUED command completion**

**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.

**Error** The Error register 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.

All other fields as defined in 10.5.7.

NOTE 63 - 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.

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 343).

Field	7	6	5	4	3	2	1	0
Error(7:0)	ERROR							
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 343 – WRITE FPDMA QUEUED error on command receipt**

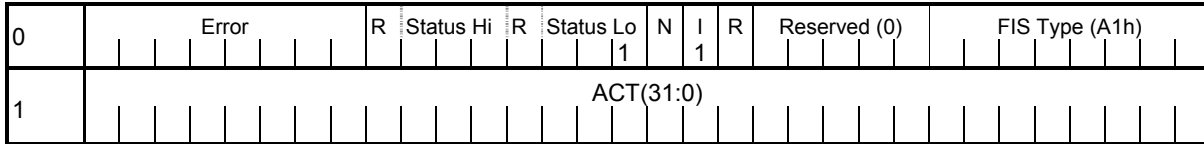
ERROR	ATA error code for the failure condition of the failed command
BSY	0
DRDY	1
DF	0
DRQ	0
ERR	1

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) 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 344) 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 344 – Set Device Bits FIS with error notification, and command completions**

- 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.
  - Error** The Error register shall contain the ATA error code.
  - 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.
- 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) has been read before continuing to abort all outstanding commands. See 13.6.4.4 for more details.

### 13.6.6 NCQ NON-DATA

#### 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 are processed as Immediate NCQ commands (see 4.1.1.73).

The NCQ NON-DATA log (see 13.7.5) 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			
Features(15:8)	Subcommand Specific							
Count(7:0)	TAG				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)	Reserved							
Auxiliary(15:8)	Reserved							
Auxiliary(23:16)	Subcommand Specific							
Auxiliary(31:24)	Reserved							
Device(7:0)	Res	1	Res	0	Reserved			
Command(7:0)	63h							

**Figure 345 – NCQ NON-DATA - command definition**

**TAG** The TAG 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.

**Subcommand** The Subcommand field (see Table 106) 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.

**Subcommand Specific** The Subcommand Specific fields are defined for each subcommand as referenced from Table 106.

NOTE 64 - In previous versions of this specification LBA(7:3) were assigned to Subcommand Specific (TTAG) and LBA(2:0) were reserved. See Table 106 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 106 – Subcommands for NCQ NON-DATA**

<b>Subcommand</b>	<b>Description</b>	<b>Reference</b>
0h	ABORT NCQ QUEUE	13.6.6.3
1h	DEADLINE HANDLING	13.6.6.4
2h	HYBRID DEMOTE BY SIZE	13.6.6.7
3h	HYBRID CHANGE BY LBA RANGE	13.6.6.5
4h	HYBRID CONTROL	13.6.6.6
5h	SET FEATURES	13.6.6.8
6h..Fh	Reserved	



### 13.6.6.3 ABORT NCQ QUEUE subcommand (0h)

#### 13.6.6.3.1 ABORT NCQ QUEUE subcommand definition

A Subcommand set 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 13.7.5).

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 346.

Field	7	6	5	4	3	2	1	0
Features(7:0)	Abort Type				0h			
Features(15:8)	Reserved							
Count(7:0)	TAG				Reserved			
Count(15:8)	Reserved							
LBA(7:0)	TTAG				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 346 – ABORT NCQ QUEUE – subcommand definition**

Abort Type describes the action requested. Table 107 shows the defined abort types. The NCQ NON-DATA Log (see 13.7.5) provides a list of abort types supported by the device (see Table 107).

**TAG** The TAG 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.

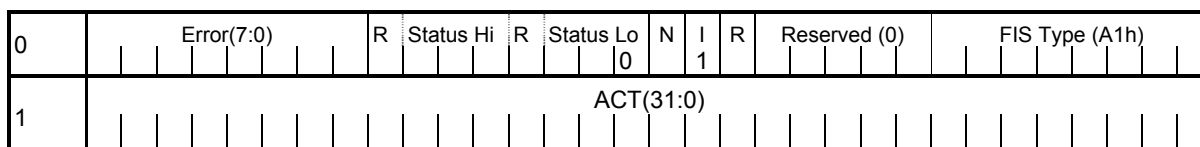
**Table 107 – Abort Type**

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 parameter 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, 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 347 – ABORT NCQ QUEUE – successful completion**

**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.

**Error** The Error register shall contain 00h.

**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.

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 348).

Field	7	6	5	4	3	2	1	0
Error(7:0)	ERROR							
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 348 – ABORT NCQ QUEUE – error on command receipt**

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) 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 349) 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.

0	Error(7:0)							R	Status Hi			R	Status Lo			N	I	R	Reserved (0)				FIS Type (A1h)			
1	ACT(31:0)																									

**Figure 349 – ABORT NCQ QUEUE – error during processing**

- 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.
- Error The Error register shall contain the ATA error code.
- 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.
- 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) 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 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 13.7.5). The format of the command is defined in Figure 350.

Field	7	6	5	4	3	2	1	0
Features(7:0)	Reserved		RDNC	WDNC	1h			
Features(15:8)	Reserved							
Count(7:0)	TAG				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 350 – DEADLINE HANDLING – subcommand definition**

**WDNC** If the WDNC (Write Data Not Continue) bit is cleared to zero, then the device may allow WRITE FPDMA QUEUED command completion times to exceed what the ICC parameter specified. If the WDNC bit is set to one, then the all WRITE FPDMA QUEUED commands shall be completed by the time specified by the ICC timer value, otherwise the device shall return command aborted for all outstanding WRITE FPDMA QUEUED commands with PRIO set to 01b (see 13.6.5).

**RDNC** If the RDNC (Read Data Not Continue) bit is cleared to zero, then the device may allow READ FPDMA QUEUED command completion times to exceed what the ICC parameter specified. If the RDNC bit is set to one, then all READ FPDMA QUEUED commands shall be completed by the time specified by the ICC timer value, otherwise the device shall return command aborted for all outstanding READ FPDMA QUEUED commands with PRIO set to 01b (see 13.6.4).

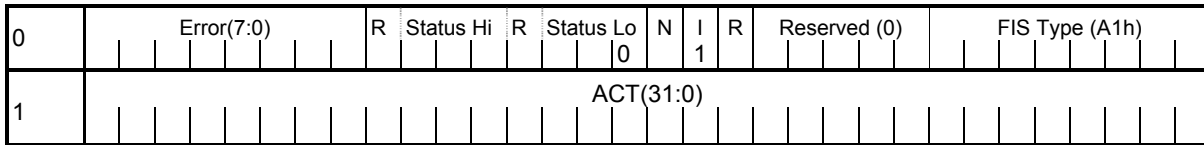
The state of the WDNC and RDNC bits shall be preserved across software resets and COMRESETs (via Software Setting Preservations), and shall not be preserved across power cycles.

**TAG** The TAG 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 command is supported, the device shall return command completed with no error.

If a DEADLINE HANDLING Subcommand command completes successfully, 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 351 – DEADLINE HANDLING – successful completion**

**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.

**Error** The Error register shall contain 00h.

**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.

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 352).

Field	7	6	5	4	3	2	1	0
Error(7:0)	ERROR							
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 352 – DEADLINE HANDLING – error on command receipt**

**ERROR** ATA error code for the failure condition of the failed command

**BSY** 0

**DRDY** 1

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

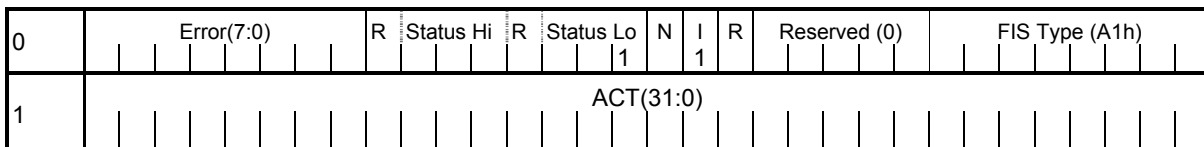
DF        0  
DRQ      0  
ERR      1

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) 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 344). 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 353 – DEADLINE HANDLING – error during processing**

- 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.
  - Error**      The Error register shall contain the ATA error code.
  - 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.
- 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) 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 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 13.7.5.11).

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 13.7.7.2.11), the Hybrid Priority field, and the CB (i.e., Cache Behavior) bit 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 108.

**Table 108 – Cache Behavior**

Hybrid Priority	Max Priority Behavior	Cache Behavior	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 13.7.7.2.11) 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., ASC field and ASCQ 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 65: 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 should 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				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							
Auxiliary(31:24)	Reserved							

**Figure 354 – HYBRID CHANGE BY LBA RANGE – subcommand definition**

**CB** The CB (Cache Behavior) 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 (Cache Behavior) 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 (Cache Behavior) 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.

**Starting LBA** The Starting LBA field specifies the first LBA.

**Sector Count** The Sector Count field specifies the requested number of logical sectors, starting from Starting LBA.

**Hybrid Priority** The Hybrid Priority field is embedded in the Hybrid Information field.

**Hybrid Information** The Hybrid Information field contains the data hints for the specified logical sectors (see 13.20.2).

**TAG** The TAG 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.

**13.6.6.5.2 Success outputs**

If a HYBRID CHANGE BY LBA RANGE subcommand completes without error, 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	1	R	Reserved (0)				FIS Type (A1h)			
1	ACT(31:0)																						

**Figure 355 – HYBRID CHANGE BY LBA RANGE – successful completion**

- 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.
  - Error The Error register shall contain 00h.
  - 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.
- 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 356).

Field	7	6	5	4	3	2	1	0
Error(7:0)	ERROR							
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 356 – HYBRID CHANGE BY LBA RANGE – error on command receipt**

- ERROR ATA error code for the failure condition of the failed command
- BSY 0
- DRDY 1
- DF 0
- DRQ 0
- ERR 1

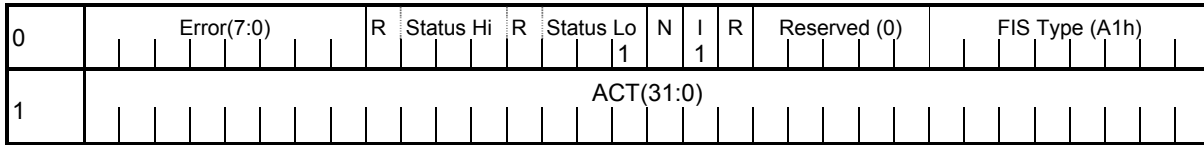
Status bit 4 may be set to one.

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 357) 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 357 – HYBRID CHANGE BY LBA RANGE – error during processing**

**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.

**Error** The Error register shall contain the ATA error code.

**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.

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 provides parameters for the use of the non-volatile caching medium. Support for this subcommand is indicated in the NCQ NON-DATA Log (see 13.7.5.12).

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 Threshold field and the Dirty High Threshold field across all resets and power cycle events. Current values are available in the Hybrid Information log (see 13.7.7).

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				Reserved			
Count(15:8)	Reserved							
LBA(7:0)	Dirty Low Threshold							
LBA(15:8)	Dirty High Threshold							
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 358 – HYBRID CONTROL – subcommand definition**

**Disable Caching Media** If the Disable Caching Media field is cleared to zero, then the device shall process the Dirty Low Threshold field and the Dirty High Threshold field.

If the Disable Caching Media field is set to one and the Hybrid Information feature is enabled, then the device shall:

- 1) ignore the contents of the Dirty Low Threshold field and the Dirty High Threshold field;
- 2) change the value of the Enabled field (see 13.7.7.2.3) 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 13.7.7.2.3) 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 13.7.7.2.3) 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.

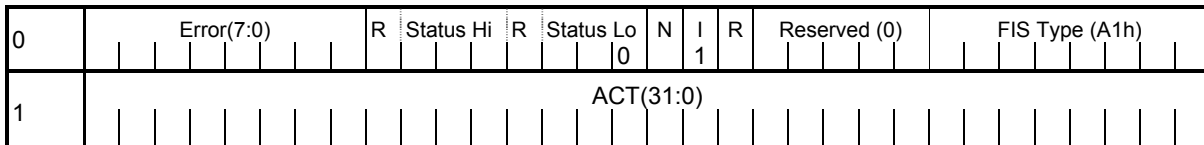
**Dirty Low Threshold** The Dirty Low Threshold represents the threshold for the amount of dirty user logical sectors in the non-volatile caching medium that point syncing operations should stop. The value of the Dirty Low Threshold field divided by 255 specifies a fraction of the total reported NVM 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 13.7.7.2.5).

**Dirty High Threshold** The Dirty High Threshold represents the threshold for the amount of dirty user logical sectors in the non-volatile caching medium that point 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 13.7.7.2.6). See 13.20.3 for additional information on syncing.

**TAG** The TAG 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.

**13.6.6.6.2 Success outputs**

If a HYBRID CONTROL subcommand completes without error, a Set Device Bits FIS shall be sent to the host. This SDB FIS may also indicate other completed commands.



**Figure 359 – HYBRID CONTROL – successful completion**

**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.

**Error** The Error register shall contain 00h.

**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.

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 360).

Field	7	6	5	4	3	2	1	0
Error(7:0)	ERROR							
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 360 – HYBRID CONTROL – error on command receipt**

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 361) 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	N	I	R	Reserved (0)	FIS Type (A1h)
1	ACT(31:0)									

**Figure 361 – HYBRID CONTROL – error during processing**

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

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

successful completion notification for. The device may set more than one bit to one if it is explicitly aggregating successful status returns.

Error The Error register shall contain the ATA error code.

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.

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 is used to change the hybrid priority associated with logical sectors. Support for this subcommand is indicated in the NCQ NON-DATA Log (see 13.7.5.10).

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 109 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 109 – HYBRID DEMOTE BY SIZE – number of logical sectors affected**

Sector Count <sup>a</sup> is less than Actual <sup>b</sup>	Number of Logical Sectors to change to the To Priority <sup>c</sup>
Y	Sector Count <sup>a</sup>
N	Actual <sup>b</sup> number
<sup>a</sup> The value of the Sector Count field in the command parameters. <sup>b</sup> The actual number of logical sectors in the non-volatile caching medium associated with the From priority. <sup>c</sup> The value of the Hybrid Priority field (see 13.7.5.10) 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 13.7.7.2.8) in the Hybrid Information log and the Max Priority Behavior bit (see 13.7.7.2.11) 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				2h			
Features(15:8)	Sector Count(7:0)							
Count(7:0)	TAG				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							
Auxiliary(31:24)	Reserved							

**Figure 362 – HYBRID DEMOTE BY SIZE – command definition**



**From Priority** The FromPriority specifies the Hybrid Priority level corresponding to the data that is to be changed to the hybrid priority specified in the Hybrid Information field.

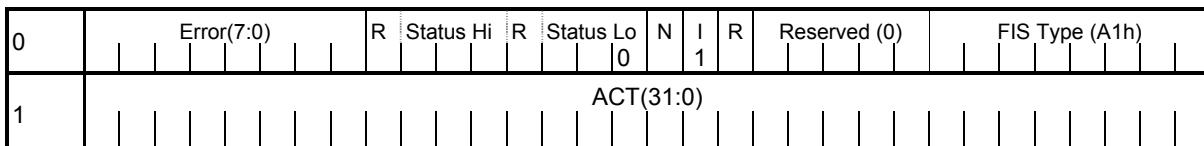
**Sector Count** The Sector Count field specifies the requested number of logical sectors that the change of hybrid priority applies.

**TAG** The TAG 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, then a Set Device Bits FIS shall be sent to the host. This SDB FIS may also indicate other completed commands.



**Figure 363 – HYBRID DEMOTE BY SIZE – successful completion**

**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.

**Error** The Error register shall contain 00h.

**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.

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 364).

Field	7	6	5	4	3	2	1	0
Error(7:0)	ERROR							
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 364 – HYBRID DEMOTE BY SIZE – error on command receipt**

ERROR ATA error code for the failure condition of the failed command

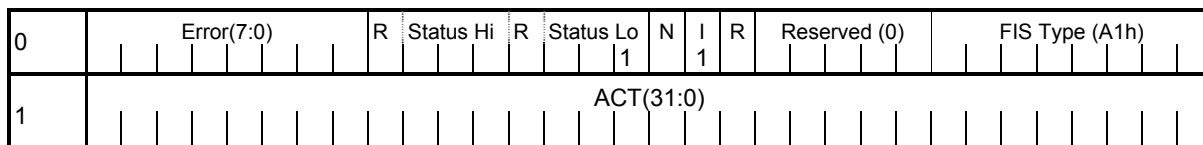
BSY 0  
DRDY 1  
DF 0  
DRQ 0  
ERR 1

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 365) 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 365 – HYBRID DEMOTE BY SIZE – error during processing**

- 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.
- Error The Error register shall contain the ATA error code.
- 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.

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

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 (5h)

#### 13.6.6.8.1 SET FEATURES overview

The SET FEATURES subcommand functionality and behavior is dependent on all requirements of the SET FEATURES command and the IDENTIFY DEVICE command defined in ACS-3.

NOTE 66 – 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

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) field							
Count(7:0)	TAG				Reserved			
Count(15:8)	Contents of the SET FEATURES command Count(7:0) field							
LBA(7:0)	Contents of the SET FEATURES command LBA(7:0) field							
LBA(15:8)	Contents of the SET FEATURES command LBA(15:8) field							
LBA(23:16)	Contents of the SET FEATURES command LBA(23:16) field							
LBA(31:24)	Reserved				Contents of the SET FEATURES command LBA(27:24) field			
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 366 – SET FEATURES QUEUED subcommand = 05h

See ACS-3 for the definition of LBA(27:0) 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

### 13.6.7 RECEIVE FPDMA QUEUED

#### 13.6.7.1 RECEIVE FPDMA QUEUED 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 367.

#### 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				Reserved			
Count(15:8)	PRIO(1:0)		Res	Subcommand				
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)	Auxiliary(7:0)							
Auxiliary(15:8)	Auxiliary(15:8)							
Auxiliary(23:16)	Reserved							
Auxiliary(31:24)	Reserved							
Device(7:0)	Res	1	Res	0	Reserved			
Command(7:0)	65h							

**Figure 367 – RECEIVE FPDMA QUEUED command definition**

**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 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 Priority (PRIO) 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 105).

**Subcommand** See 13.6.7.5.

**LBA** See 13.6.7.5.

**ICC** The Isochronous Command Completion (ICC) field shall be assigned by the host based on the intended deadline associated with the command issued. By default, if deadline is expired, the device shall continue to complete the command as soon as possible.

**Auxiliary** See 13.6.7.5.

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.

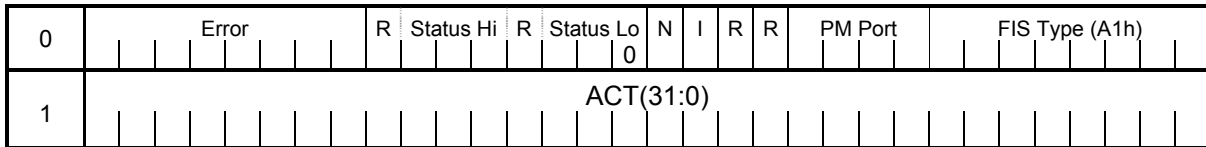
The host shall check the BSY bit in the shadow Status register before attempting to issue a new command in order to determine that the device is ready to receive another command (and

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 BSY 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 368.



**Figure 368 – Set Device Bits FIS for successful RECEIVE FPDMA QUEUED command completion**

- 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.
- Error     The Error register 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.

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 369).

Field	7	6	5	4	3	2	1	0
Error(7:0)	ERROR							
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 369 – RECEIVE FPDMA QUEUED error status result values on command receipt**

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 370) 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.

0	Error	R	Status Hi	R	Status Lo	N	I	R	R	PM Port	FIS Type (A1h)
1	ACT(31:0)										

**Figure 370 – Set Device Bits FIS with error notification, and command completions**

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.

Error The Error register shall contain the ATA error code.

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.

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(12:8) field. The allowed values are defined in Table 110.

**Table 110 – Subcommands for RECEIVE FPDMA QUEUED**

Value	Subcommand
00h	Reserved
01h	READ LOG DMA EXT (see 13.6.7.6)
02h..1Fh	Reserved

### 13.6.7.6 READ LOG DMA EXT

#### 13.6.7.6.1 READ LOG DMA EXT overview

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-3.



### 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) field							
Features(15:8)	Contents of READ LOG DMA EXT Count(15:8) field							
LBA(7:0)	Contents of READ LOG DMA EXT LBA(7:0) field							
LBA(15:8)	Contents of READ LOG DMA EXT LBA(15:8) field							
LBA(23:16)	Contents of READ LOG DMA EXT LBA(23:16) field							
LBA(31:24)	Contents of READ LOG DMA EXT LBA(31:24) field							
LBA(39:32)	Contents of READ LOG DMA EXT LBA(39:32) field							
LBA(47:40)	Contents of READ LOG DMA EXT LBA(47:40) field							
Auxiliary(7:0)	Reserved							
Auxiliary(15:8)	Reserved							
Auxiliary(23:16)	Reserved							
Auxiliary(31:24)	Reserved							

**Figure 371 – RECEIVE FPDMA QUEUED subcommand = 01h**

See ACS-3 for the definition of LBA(47:0) 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

## 13.6.8 SEND FPDMA QUEUED

### 13.6.8.1 SEND FPDMA QUEUED 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 372.

### 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				Reserved			
Count(15:8)	PRIO(1:0)		Res	Subcommand				
LBA(7:0)	LBA(7:0)							
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)	ICC(7:0)							
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 372 – SEND FPDMA QUEUED command definition**

**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 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 Priority (PRIO) 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 105).

**Subcommand** Subcommand specific (see 13.6.8.5)

**LBA** Subcommand specific (see 13.6.8.5)

**ICC** The Isochronous Command Completion (ICC) field shall be assigned by the host based on the intended deadline associated with the command issued. By default, if deadline is expired, the device shall continue to complete the command as soon as possible.

**Auxiliary** Subcommand specific (see 13.6.8.5)

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.

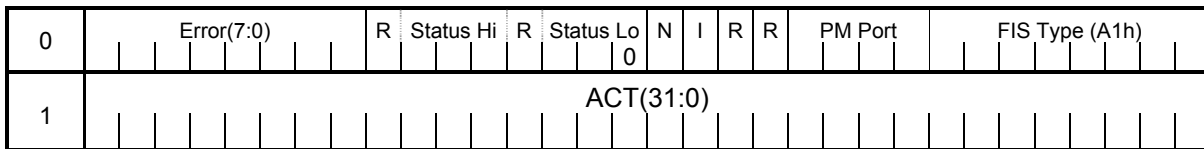
The host shall check the BSY bit in the shadow Status register before attempting to issue a new command in order to determine that the device is ready to receive another command (and determine that the host has write access to the Shadow Register Block Registers).

The device shall not trigger an interrupt in response to having successfully received the command, so the initial status return that clears BSY 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 373.



**Figure 373 – Set Device Bits FIS for successful SEND FPDMA QUEUED command completion**

- 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.
- Error** The Error register 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.

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 374).

Field	7	6	5	4	3	2	1	0
Error(7:0)	ERROR							
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 374 – SEND FPDMA QUEUED error status result values on command receipt

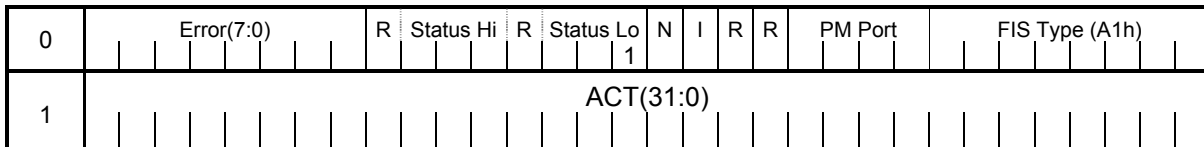
ERROR	ATA error code for the failure condition of the failed command
BSY	0
DRDY	1
DF	0
DRQ	0
ERR	1

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 398) 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 375 – Set Device Bits FIS with error notification, and command completions**

**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.

**Error** The Error register shall contain the ATA error code.

**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.

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 (12:8) field. The allowed values are defined in Table 111. Support for each subcommand is reported in the NCQ Send and Receive log (see 13.7.6).

**Table 111 – Subcommands for SEND FPDMA QUEUED**

Value	Subcommand
00h	DATA SET MANAGEMENT (see 13.6.8.6)
01h	HYBRID EVICT (see 13.6.8.7)
02h	WRITE LOG DMA EXT (see 13.6.8.8)
03h..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-3.

#### 13.6.8.6.2 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							Trim
Auxiliary(15:8)	Reserved							
Auxiliary(23:16)	Reserved							
Auxiliary(31:24)	Reserved							

Figure 376 – SEND FPDMA QUEUED, subcommand = 00h

Trim As defined by the DATA SET MANAGEMENT command in ACS-3.

#### 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-3.

### 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 13.7.6.2).

If the Evict All field (see Figure 377) 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 field 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 377) is greater than the value of the Maximum Eviction Data Blocks field (see 13.7.7.2.16) 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 13.7.7.2.15) 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 67 - 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 377 – Subcommand specific parameters for the HYBRID EVICT subcommand = 01h**

Evict All    The Evict All field 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 378 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 Bits 47:0 Starting LBA
8..15	Qword	LBA Range entry 1 Bits 63:48 Range Length Bits 47:0 Starting LBA
...		...
504..511	Qword	LBA Range entry 63 Bits 63:48 Range Length Bits 47:0 Starting LBA

**Figure 378 – 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 overview**

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-3.

**13.6.8.8.2 Inputs**

Field	7	6	5	4	3	2	1	0
Features(7:0)	Contents of WRITE LOG DMA EXT Count(7:0) field							
Features(15:8)	Contents of WRITE LOG DMA EXT Count(15:8) field							
LBA(7:0)	Contents of WRITE LOG DMA EXT LBA(7:0) field							
LBA(15:8)	Contents of WRITE LOG DMA EXT LBA(15:8) field							
LBA(23:16)	Contents of WRITE LOG DMA EXT LBA(23:16) field							
LBA(31:24)	Contents of WRITE LOG DMA EXT LBA(31:24) field							
LBA(39:32)	Contents of WRITE LOG DMA EXT LBA(39:32) field							
LBA(47:40)	Contents of WRITE LOG DMA EXT LBA(47:40) field							
Auxiliary(7:0)	Reserved							
Auxiliary(15:8)	Reserved							
Auxiliary(23:16)	Reserved							
Auxiliary(31:24)	Reserved							

**Figure 379 – SEND FPDMA QUEUED subcommand = 02h**

See ACS-3 for the definition of LBA(47:0) 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

### 13.6.9 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 Native Command Queuing protocol, the buffer 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 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 380 illustrates these concepts (the figure is intended as illustrative and does not exclude other possible host controller implementations).

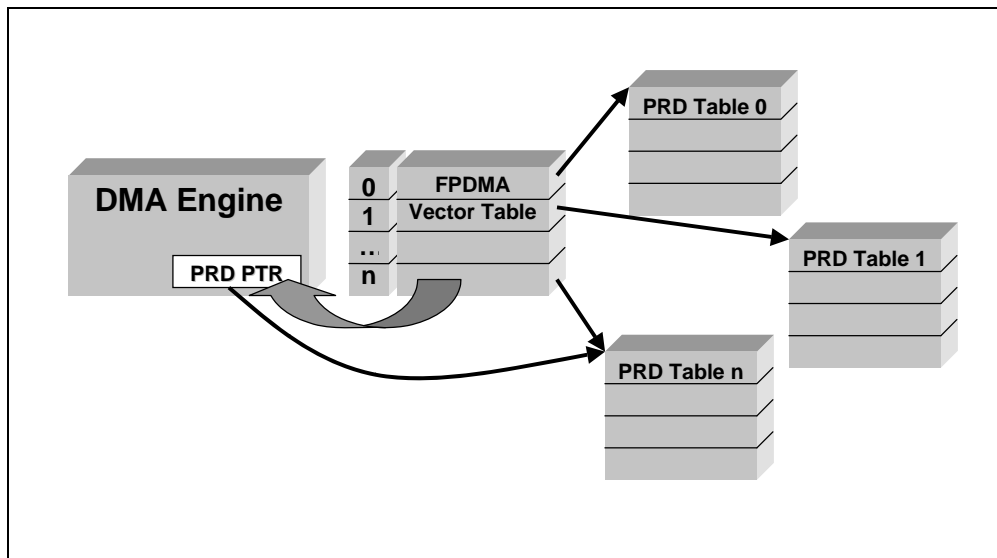


Figure 380 – Example DMA engine indirection for First-party DMA support

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 Native Command Queuing 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. They are all read-only. The READ LOG EXT command in the General Purpose Logging feature set (see ACS-3) are used to read the SATA logs. In some cases the READ LOG DMA EXT command may also be used (see ACS-3).

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 112.

**Table 112 – Log addresses for Serial ATA**

Log Address	Description	Reference
00h .. 0Fh	As defined in the ACS-3 standard	-
10h	NCQ Queued Error log	13.7.4
11h	Phy Event Counters log	13.9.4
12h	NCQ NON-DATA log	13.7.5
13h	NCQ Send and Receive log	13.7.6
14h	Hybrid Information Log	13.7.7
15h	Rebuild Assist log	13.7.8
16h..17h	Reserved	-
18h..FFh	As defined in the ACS-3 standard	-

### 13.7.3 General purpose log directory (00h)

Devices supporting the Queued Error Log (see 13.7) reflect this support in the General Purpose Log Directory (see Table 113) 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 1 page 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 (log 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.

**Table 113 – General purpose log directory values for Serial ATA**

<b>Byte</b>	<b>Log</b>	<b>Value</b>
000h..01Fh	-	As defined in the ACS-3 standard
020h	10h	1 if Native Command Queuing is supported, 0 if Native Command Queuing is not supported
021h	10h	0
022h	11h	1 if Phy Event Counters are supported 0 if Phy Event Counters are not supported
023h	11h	0
024h	12h	1 if NCQ NON-DATA is supported 0 if NCQ NON-DATA 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..02Fh	-	Reserved
030h..1FFh	-	As defined in the ACS-3 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-3.

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 one page in length and is defined in Figure 381.

If the device supports NCQ Autosense (i.e., IDENTIFY DEVICE data Word 78 bit 7 is set to one), then:

- a) the Sense Key field;
  - b) the Additional Sense Code Field; and
  - c) 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 Sense Key field shall be cleared to zero;
- b) the Additional Sense Code field shall be cleared to zero; and
- c) the Additional Sense Code Qualifier field shall be cleared to zero.

Byte	7	6	5	4	3	2	1	0
0	NQ	UNL	R	TAG				
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							
15	Additional Sense Code							
16	Additional Sense Code Qualifier							
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	Reserved							
...								
255								
256								
...	Vendor Specific							
510								
511								

**Figure 381 – Queued Error Log data structure definition**

- TAG** If the NQ bit is cleared to zero, the TAG field contains the TAG corresponding to the queued command that failed.
- 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, Error, and LBA(7:0) fields (bytes 3..5) in the log shall be set as follows:
- Status(7:0), BSY bit shall be cleared to zero and ERR bit shall be set to one;
  - Error(7:0), ABRT bit shall be set to one; and
  - LBA(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.

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

- 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.
- BYTE1..19 An image of a device to host Register Device to Host FIS is embedded in the data structure. The fields correspond to the Shadow Register Block Registers and are encoded with error information consistent with the READ DMA QUEUED EXT or WRITE DMA QUEUED EXT command defined in the ACS-3 standard.
- 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., ABRT bit set to one in the error register and all other bits cleared to zero).

NOTE 68 - 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 is ABORTED COMMAND; and
- c) the Additional Sense Code/Additional Sense Code Qualifier 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 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.

Reserved/R

All reserved fields shall be cleared to zero.

## 13.7.5 NCQ NON-DATA log (12h)

### 13.7.5.1 NCQ NON-DATA log overview

To determine the supported NCQ NON-DATA subcommands and their respective features, host software may read log 12h.

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
0	Subcommand 0h	
	31..5	Reserved
	4	Supports Abort Selected TTAG (see 13.7.5.6)
	3	Supports Abort Non-Streaming (see 13.7.5.5)
	2	Supports Abort Streaming (see 13.7.5.4)
	1	Supports Abort All (see 13.7.5.3)
	0	Supports Abort NCQ (see 13.7.5.2)
1	Subcommand 1h	
	31..3	Reserved
	2	Supports Read Data Not Continue (see 13.7.5.9)
	1	Supports Write Data Not Continue (see 13.7.5.8)
	0	Supports DEADLINE HANDLING (see 13.7.5.7)
2	Subcommand 2h	
	31..1	Reserved
	0	Supports HYBRID DEMOTE BY SIZE (see 13.7.5.10)
3	Subcommand 3h	
	31..1	Reserved
	0	Supports HYBRID CHANGE BY LBA RANGE (see 13.7.5.11)
4	Subcommand 4h	
	31..1	Reserved
	0	Supports HYBRID CONTROL (see 13.7.5.12)
5	Subcommand 5h	
	31..1	Reserved
	0	Supports Set Features (see 13.7.5.13)
...	...	...
15	Subcommand Fh	
	31..1	Reserved
	0	Supports subcommand Fh
16..128	31..0	Reserved

Figure 382 – NCQ NON-DATA log (12h) data structure definition

### 13.7.5.2 Supports Abort NCQ

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.

### 13.7.5.3 Supports Abort All

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.



#### **13.7.5.4 Supports Abort Streaming**

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.

#### **13.7.5.5 Supports Abort Non-Streaming**

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.

#### **13.7.5.6 Supports Abort Selected TTAG**

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.

#### **13.7.5.7 Supports DEADLINE HANDLING**

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.

#### **13.7.5.8 Supports Write Data Not Continue**

If the Supports Write Data Not Continue bit is set to one, then the device supports the WDNC field 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 field of the DEADLINE HANDLING command (see 13.6.6.4).

#### **13.7.5.9 Supports Read Data Not Continue**

If the Supports Read Data Not Continue bit is set to one, then the device supports the RDNC field 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 field of the DEADLINE HANDLING command (see 13.6.6.4).

#### **13.7.5.10 Supports HYBRID DEMOTE BY SIZE**

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.

#### **13.7.5.11 Supports HYBRID CHANGE BY LBA RANGE**

If the Supports HYBRID CHANGE BY LBA RANGE 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.

#### **13.7.5.12 Supports HYBRID CONTROL**

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.

#### **13.7.5.13 Supports Set Features**

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 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 NCQ send and receive log (13h)

#### 13.7.6.1 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.

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
0	Subcommands Supported	
	31..2	Reserved
	1	Supports HYBRID EVICT (see 13.7.6.2)
	0	Supports Data Set Management(see 13.7.6.3)
1	Data Set Management	
	31..1	Reserved
	0	Supports Trim (see 13.7.6.4)
2	Supports Read Log	
	31..1	Reserved
	0	Supports Read Log (see 13.7.6.5)
3	Supports Write Log	
	31..1	Reserved
	0	Supports Write Log (see 13.7.6.6)
4..127		Reserved

Figure 383 – NCQ SEND and RECEIVE log (13) data structure definition

#### 13.7.6.2 Supports HYBRID EVICT

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.

#### 13.7.6.3 Supports Data Set Management

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.

#### 13.7.6.4 Supports Trim

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.

#### 13.7.6.5 Supports Read Log

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.

### **13.7.6.6 Supports Write Log**

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 Hybrid Information log (14h)

#### 13.7.7.1 Hybrid Information log overview

If the Hybrid Information feature is supported (see 13.7.9.2.24), then the Hybrid Information log shall be supported. The Hybrid Information log consists of one page (see Figure 384). 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	13.7.7.2
64	Hybrid Information Descriptor for Hybrid Priority 0	13.7.7.3
...	...	-
	Hybrid Information Descriptor for Maximum Priority (N)	13.7.7.3
64 + (16 × N) to 511	Padding	-

**Figure 384 – Hybrid Information log data**

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.

N The number of Hybrid Information Descriptors reported in the log.

#### 13.7.7.2 Hybrid Information Header

##### 13.7.7.2.1 Hybrid Information Header overview

Figure 385 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	13.7.7.2.2
2	Byte	Enabled	13.7.7.2.3
3	Byte	Hybrid Health	13.7.7.2.4
4	Byte	Dirty Low Threshold	13.7.7.2.5
5	Byte	Dirty High Threshold	13.7.7.2.6
6	Byte	Optimal Write Granularity	13.7.7.2.7
7	Byte	Bits Description 7:4 Reserved 3:0 Maximum Hybrid Priority Level	13.7.7.2.8
8	Byte	Power Condition	13.7.7.2.9
9	Byte	Caching Medium Enabled	13.7.7.2.10
10	Byte	Supported Options	13.7.7.2.11
11		Reserved	-
12..15	Dword	Time Since Enabled	13.7.7.2.12
16..23	Qword	NVM Size	13.7.7.2.13
24..31	Qword	Enable Count	13.7.7.2.14
32..33	Word	Bits Description 7:5 Reserved 4:0 Maximum Eviction Commands	13.7.7.2.15
34..35	Word	Maximum Eviction Data Blocks	13.7.7.2.16
36..63		Reserved	-

**Figure 385 – Hybrid Information Header**

### 13.7.7.2.2 Number of Hybrid Information Descriptors

The Number of Hybrid Information Descriptors field indicates the the number of Hybrid Information descriptors that follow the header.

### 13.7.7.2.3 Enabled

Table 114 indicates the value that the device shall indicate in IDENTIFY DEVICE in each case.

**Table 114 – 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	-

### 13.7.7.2.4 Hybrid Health

#### 13.7.7.2.4.1 Hybrid Health overview

Table 115 describes the values of the Hybrid Health field that contains several indicators of the health of the non-volatile caching medium.

NOTE 69 – If the non-volatile caching medium is healthy, the Hybrid Health field should be zero.

**Table 115 – Hybrid Health**

Bit	Description	Reference
7:4	Reserved	-
3	Data Loss	13.7.7.2.4.2
2	Read Only	13.7.7.2.4.3
1	NVM Size Changed	13.7.7.2.4.4
0	Unuseable	13.7.7.2.4.5

#### 13.7.7.2.4.2 Data Loss

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 13.7.7).

#### 13.7.7.2.4.3 Read Only

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.

#### 13.7.7.2.4.4 NVM Size Changed

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.

#### **13.7.7.2.4.5 Unuseable**

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.

#### **13.7.7.2.5 Dirty Low Threshold**

The Dirty Low Threshold field indicates the threshold for the amount of dirty user logical sectors in the non-volatile caching medium that point syncing operations should stop. For additional information, see the HYBRID CONTROL subcommand (see 13.6.6.6).

#### **13.7.7.2.6 Dirty High Threshold**

The Dirty High Threshold field indicates the threshold for the amount of dirty user logical sectors in the non-volatile caching medium that point syncing operations should begin. For additional information, see the HYBRID CONTROL subcommand (see 13.6.6.6).

#### **13.7.7.2.7 Optimal Write Granularity**

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.

#### **13.7.7.2.8 Maximum Hybrid Priority Level**

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 shall be non-zero.

#### **13.7.7.2.9 Power Condition**

The Power Condition field indicates the current power condition that the CHECK POWER MODE command would report in normal outputs (see ACS-3).

#### **13.7.7.2.10 Caching Medium Enabled**

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 .

#### **13.7.7.2.11 Supported Options**

The Supported Options field (see Table 116) indicates that optional behaviors are supported.

**Table 116 – Supported Options**

Bit	Description
7:2	Reserved
1	Supports Cache Behavior
0	Max Priority Behavior

If the Supports Cache Behavior bit is set to one, then the device supports per command tagging of the HYBRID CHANGE BY LBA RANGE command 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 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.

#### **13.7.7.2.12 Time Since Enabled**

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.

#### **13.7.7.2.13 NVM Size**

The NVM Size field indicates the number of logical sectors that comprise the non-volatile caching medium.

NOTE 70 - The value of the NVM Size field may vary over time because of vendor specific factors.

#### **13.7.7.2.14 Enable Count**

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).

#### **13.7.7.2.15 Maximum Eviction Commands**

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.

#### **13.7.7.2.16 Maximum Eviction Data Blocks**

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).



### 13.7.7.3 Hybrid Information Descriptor

#### 13.7.7.3.1 Hybrid Information Descriptor overview

Figure 386 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	13.7.7.3.2
1	Byte	Consumed NVM Size Fraction	13.7.7.3.3
2	Byte	Consumed Mapping Resources Fraction	13.7.7.3.4
3	Byte	Consumed NVM Size for Dirty Data Fraction	13.7.7.3.5
4	Byte	Consumed Mapping Resources for Dirty Data Fraction	13.7.7.3.6
5..15		Reserved	

Figure 386 – Hybrid Information Descriptor

#### 13.7.7.3.2 Hybrid Priority

The Hybrid Priority field indicates the Hybrid Priority number that this descriptor represents.

#### 13.7.7.3.3 Consumed NVM Size Fraction

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 (see 13.7.7.2.13).

#### 13.7.7.3.4 Consumed Mapping Resources Fraction

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.

#### 13.7.7.3.5 Consumed NVM Size For Dirty Data Fraction

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 (see 13.7.7.2.13).

#### **13.7.7.3.6 Consumed Mapping Resources for Dirty Data Fraction**

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.

### 13.7.8 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.

The Rebuild Assist log shall be accessed using the GPL feature set commands (see ACS-3).

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 field is cleared to zero, then the device shall:

- 1) disable the Rebuild Assist feature;
  - a) clear the Rebuild Assist Enabled field 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 field 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 field 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
1..6	Reserved
7	Physical Element Length (N)
8	(MSB)
	Disabled Physical Element Mask
7+N	(LSB)
8+N	(MSB)
	Disabled Physical Elements
7+(2×N)	(LSB)
8+(2×N)..511	Reserved

**Figure 387 – Rebuild Assist log**

#### Rebuild Assist Enabled

Table 117 describes the use of the Rebuild Assist Enabled field.

**Table 117 – Rebuild Assist Enabled**

Operation	Rebuild Assist Enabled	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 set 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.

**13.7.9 Identify Device data log (30h)**

**13.7.9.1 Serial ATA settings (page 08h)**

The Serial ATA log page (see Table 119) provides information about the Serial ATA Transport.

**Table 118 – Serial ATA (page 08h)**

(part 1 of 3)

Offset	Type	Contents
0..7	Qword	Serial ATA page information header.
		Bit Meaning
		63 Shall be set to one.
		62:24 Reserved
		32:16 Page number. Shall be set to 08h. 15:0 Revision number. Shall be set to 0001h.
8:15	Qword	SATA Capabilities
		Bit Meaning
		63 Shall be set to one.
		62:29 Reserved
		28 DIPM SSP PRESERVATION SUPPORTED (see 13.7.9.2.25)
		27 Reserved
		26 DEVSLEEP TO REDUCEDPWRSTATE CAPABILITY SUPPORTED (see 13.7.9.2.23)
		25 DEVICE SLEEP SUPPORTED (see 13.7.9.2.22)
		24 NCQ AUTOSENSE SUPPORTED (see 13.7.9.2.21)
		23 SOFTWARE SETTINGS PRESERVATION SUPPORTED (see 13.7.9.2.20)
		22 HARDWARE FEATURE CONTROL SUPPORTED (see 13.7.9.2.19)
		21 IN-ORDER DATA DELIVERY SUPPORTED (see 13.7.9.2.18)
		20 DEVICE INITIATED POWER MANAGEMENT SUPPORTED (see 13.7.9.2.17)
		19 DMA SETUP FIS AUTO-ACTIVATE SUPPORTED (see 13.7.9.2.16)
		18 NON-ZERO BUFFER OFFSETS SUPPORTED (see 13.7.9.2.15)
		17 SEND AND RECEIVE QUEUED COMMANDS SUPPORTED (see 13.7.9.2.14)
		16 NCQ NON-DATA COMMAND SUPPORTED (see 13.7.9.2.13)
		15 NCQ STREAMING SUPPORTED (see 13.7.9.2.12)
		14 READ LOG DMA EXT AS EQUIVALENT TO READ LOG EXT SUPPORTED (see 13.7.9.2.11)
		13 DEVICE AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED (see 13.7.9.2.10)
		12 HOST AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED (see 13.7.9.2.9)
		11 NCQ PRIORITY INFORMATION SUPPORTED (see 13.7.9.2.8)
		10 UNLOAD WHILE NCQ COMMANDS ARE OUTSTANDING SUPPORTED (see 13.7.9.2.7)
		9 SATA PHY EVENT COUNTERS LOG SUPPORTED (see 13.7.9.2.6)
		8 RECEIPT OF HOST INITIATED POWER MANAGEMENT REQUESTS SUPPORTED (see 13.7.9.2.5)
		7 NCQ FEATURE SET SUPPORTED (see 13.7.9.2.4)
		6:3 Reserved
		2 SATA GEN3 SIGNALING SPEED SUPPORTED (see 13.7.9.2.3)
1 SATA GEN2 SIGNALING SPEED SUPPORTED (see 13.7.9.2.2)		
0 SATA GEN1 SIGNALING SPEED SUPPORTED (see 13.7.9.2.1)		

**Table 118 – Serial ATA (page 08h) (part 2 of 3)**

Offset	Type	Contents																														
16..23	Qword	<p>Current SATA Settings</p> <table border="0"> <tr> <td>Bit</td> <td>Meaning</td> </tr> <tr> <td>63</td> <td>Shall be set to one.</td> </tr> <tr> <td>62:11</td> <td>Reserved</td> </tr> <tr> <td>10</td> <td>DEVICE SLEEP ENABLED (see 13.7.9.3.9)</td> </tr> <tr> <td>9</td> <td>AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS ENABLED (see 13.7.9.3.8)</td> </tr> <tr> <td>8</td> <td>SOFTWARE SETTINGS PRESERVATION ENABLED (see 13.7.9.3.7)</td> </tr> <tr> <td>7</td> <td>HARDWARE FEATURE CONTROL ENABLED (see 13.7.9.3.6)</td> </tr> <tr> <td>6</td> <td>IN-ORDER DATA DELIVERY ENABLED (see 13.7.9.3.5)</td> </tr> <tr> <td>5</td> <td>DEVICE INITIATED POWER MANAGEMENT ENABLED (see 13.7.9.3.4)</td> </tr> <tr> <td>4</td> <td>DMA SETUP AUTO-ACTIVATION ENABLED (see 13.7.9.3.3)</td> </tr> <tr> <td>3</td> <td>NON-ZERO BUFFER OFFSETS ENABLED (see 13.7.9.3.2)</td> </tr> <tr> <td>2:0</td> <td>CURRENT NEGOTIATED SERIAL ATA SIGNAL SPEED (see 13.7.9.3.1)</td> </tr> </table>	Bit	Meaning	63	Shall be set to one.	62:11	Reserved	10	DEVICE SLEEP ENABLED (see 13.7.9.3.9)	9	AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS ENABLED (see 13.7.9.3.8)	8	SOFTWARE SETTINGS PRESERVATION ENABLED (see 13.7.9.3.7)	7	HARDWARE FEATURE CONTROL ENABLED (see 13.7.9.3.6)	6	IN-ORDER DATA DELIVERY ENABLED (see 13.7.9.3.5)	5	DEVICE INITIATED POWER MANAGEMENT ENABLED (see 13.7.9.3.4)	4	DMA SETUP AUTO-ACTIVATION ENABLED (see 13.7.9.3.3)	3	NON-ZERO BUFFER OFFSETS ENABLED (see 13.7.9.3.2)	2:0	CURRENT NEGOTIATED SERIAL ATA SIGNAL SPEED (see 13.7.9.3.1)						
Bit	Meaning																															
63	Shall be set to one.																															
62:11	Reserved																															
10	DEVICE SLEEP ENABLED (see 13.7.9.3.9)																															
9	AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS ENABLED (see 13.7.9.3.8)																															
8	SOFTWARE SETTINGS PRESERVATION ENABLED (see 13.7.9.3.7)																															
7	HARDWARE FEATURE CONTROL ENABLED (see 13.7.9.3.6)																															
6	IN-ORDER DATA DELIVERY ENABLED (see 13.7.9.3.5)																															
5	DEVICE INITIATED POWER MANAGEMENT ENABLED (see 13.7.9.3.4)																															
4	DMA SETUP AUTO-ACTIVATION ENABLED (see 13.7.9.3.3)																															
3	NON-ZERO BUFFER OFFSETS ENABLED (see 13.7.9.3.2)																															
2:0	CURRENT NEGOTIATED SERIAL ATA SIGNAL SPEED (see 13.7.9.3.1)																															
24..39		Reserved																														
40..41	Word	CURRENT HARDWARE FEATURE CONTROL IDENTIFIER (see 13.7.9.3.12)																														
42..43	Word	SUPPORTED HARDWARE FEATURE CONTROL IDENTIFIER (see 13.7.9.3.13)																														
44..47		Reserved																														
48..55	Qword	<p>DEVSLP TIMING VARIABLES</p> <table border="0"> <tr> <td>Bit</td> <td>Meaning</td> </tr> <tr> <td>63</td> <td>DEVSLP TIMING VARIABLES SUPPORTED (see 13.7.9.4.1)</td> </tr> <tr> <td>62:16</td> <td>Reserved</td> </tr> <tr> <td>15:8</td> <td>DEVSLEEP EXIT TIMEOUT (DETO) (see 13.7.9.4.2)</td> </tr> <tr> <td>7:5</td> <td>Reserved</td> </tr> <tr> <td>4:0</td> <td>MINIMUM DEVSLP ASSERTION TIME (MDAT) (see 13.7.9.4.3)</td> </tr> </table>	Bit	Meaning	63	DEVSLP TIMING VARIABLES SUPPORTED (see 13.7.9.4.1)	62:16	Reserved	15:8	DEVSLEEP EXIT TIMEOUT (DETO) (see 13.7.9.4.2)	7:5	Reserved	4:0	MINIMUM DEVSLP ASSERTION TIME (MDAT) (see 13.7.9.4.3)																		
Bit	Meaning																															
63	DEVSLP TIMING VARIABLES SUPPORTED (see 13.7.9.4.1)																															
62:16	Reserved																															
15:8	DEVSLEEP EXIT TIMEOUT (DETO) (see 13.7.9.4.2)																															
7:5	Reserved																															
4:0	MINIMUM DEVSLP ASSERTION TIME (MDAT) (see 13.7.9.4.3)																															
56..63	Qword	<p>TRANSITIONAL ENERGY REPORTING</p> <table border="0"> <tr> <td>Bit</td> <td>Meaning</td> </tr> <tr> <td>63</td> <td>TER SUPPORTED (see 13.7.9.5.14)</td> </tr> <tr> <td>62:55</td> <td>Reserved</td> </tr> <tr> <td>54:53</td> <td>IN-STATE DEVSLEEP POWER UNIT (see 13.7.9.5.13)</td> </tr> <tr> <td>52:48</td> <td>TYPICAL IN-STATE DEVSLEEP POWER (see 13.7.9.5.12)</td> </tr> <tr> <td>47:46</td> <td>OFF TO GOOD STS LATENCY TIME UNIT (see 13.7.9.5.11)</td> </tr> <tr> <td>45:40</td> <td>OFF TO GOOD STS LATENCY (see 13.7.9.5.10)</td> </tr> <tr> <td>39:38</td> <td>BETWEEN POWER CYCLES TIME UNIT (see 13.7.9.5.9)</td> </tr> <tr> <td>37:24</td> <td>RECOMMENDED TIME BETWEEN POWER CYCLES (see 13.7.9.5.8)</td> </tr> <tr> <td>23:22</td> <td>OFF TO GOOD STS TIME UNIT (see 13.7.9.5.7)</td> </tr> <tr> <td>21:16</td> <td>OFF TO GOOD STS RECOUP COST (see 13.7.9.5.6)</td> </tr> <tr> <td>15:14</td> <td>DEVSLEEP TO PHYRDY TIME UNIT (see 13.7.9.5.5)</td> </tr> <tr> <td>13:8</td> <td>DEVSLEEP TO PHYRDY RECOUP COST (see 13.7.9.5.4)</td> </tr> <tr> <td>7:6</td> <td>SLUMBER TO DEVSLEEP TIME UNIT (see 13.7.9.5.3)</td> </tr> <tr> <td>5:0</td> <td>SLUMBER TO DEVSLEEP RECOUP COST (see 13.7.9.5.2)</td> </tr> </table>	Bit	Meaning	63	TER SUPPORTED (see 13.7.9.5.14)	62:55	Reserved	54:53	IN-STATE DEVSLEEP POWER UNIT (see 13.7.9.5.13)	52:48	TYPICAL IN-STATE DEVSLEEP POWER (see 13.7.9.5.12)	47:46	OFF TO GOOD STS LATENCY TIME UNIT (see 13.7.9.5.11)	45:40	OFF TO GOOD STS LATENCY (see 13.7.9.5.10)	39:38	BETWEEN POWER CYCLES TIME UNIT (see 13.7.9.5.9)	37:24	RECOMMENDED TIME BETWEEN POWER CYCLES (see 13.7.9.5.8)	23:22	OFF TO GOOD STS TIME UNIT (see 13.7.9.5.7)	21:16	OFF TO GOOD STS RECOUP COST (see 13.7.9.5.6)	15:14	DEVSLEEP TO PHYRDY TIME UNIT (see 13.7.9.5.5)	13:8	DEVSLEEP TO PHYRDY RECOUP COST (see 13.7.9.5.4)	7:6	SLUMBER TO DEVSLEEP TIME UNIT (see 13.7.9.5.3)	5:0	SLUMBER TO DEVSLEEP RECOUP COST (see 13.7.9.5.2)
Bit	Meaning																															
63	TER SUPPORTED (see 13.7.9.5.14)																															
62:55	Reserved																															
54:53	IN-STATE DEVSLEEP POWER UNIT (see 13.7.9.5.13)																															
52:48	TYPICAL IN-STATE DEVSLEEP POWER (see 13.7.9.5.12)																															
47:46	OFF TO GOOD STS LATENCY TIME UNIT (see 13.7.9.5.11)																															
45:40	OFF TO GOOD STS LATENCY (see 13.7.9.5.10)																															
39:38	BETWEEN POWER CYCLES TIME UNIT (see 13.7.9.5.9)																															
37:24	RECOMMENDED TIME BETWEEN POWER CYCLES (see 13.7.9.5.8)																															
23:22	OFF TO GOOD STS TIME UNIT (see 13.7.9.5.7)																															
21:16	OFF TO GOOD STS RECOUP COST (see 13.7.9.5.6)																															
15:14	DEVSLEEP TO PHYRDY TIME UNIT (see 13.7.9.5.5)																															
13:8	DEVSLEEP TO PHYRDY RECOUP COST (see 13.7.9.5.4)																															
7:6	SLUMBER TO DEVSLEEP TIME UNIT (see 13.7.9.5.3)																															
5:0	SLUMBER TO DEVSLEEP RECOUP COST (see 13.7.9.5.2)																															

**Table 118 – Serial ATA (page 08h) (part 3 of 3)**

Offset	Type	Contents																														
64:71	Qword	Transitional energy reporting extended																														
		<table border="0"> <tr> <td>Bit</td> <td>Meaning</td> </tr> <tr> <td>63</td> <td>TERE SUPPORTED (see 13.7.9.6.14)</td> </tr> <tr> <td>62:47</td> <td>Reserved</td> </tr> <tr> <td>46:45</td> <td>IN-STATE SLUMBER POWER UNITS (see 13.7.9.6.13)</td> </tr> <tr> <td>44:40</td> <td>TYPICAL IN-STATE SLUMBER POWER (see 13.7.9.6.12)</td> </tr> <tr> <td>39:38</td> <td>PM2 TO PM0 LATENCY TIME UNIT (see 13.7.9.6.11)</td> </tr> <tr> <td>37:32</td> <td>PM2 TO PM0 LATENCY (see 13.7.9.6.10)</td> </tr> <tr> <td>31:30</td> <td>OFF TO GOOD STS RELATIVE TO DEVSLEEP/PM2 TIME UNIT (see 13.7.9.6.9)</td> </tr> <tr> <td>29:24</td> <td>OFF TO GOOD STS RELATIVE TO DEVSLEEP/PM2 RECOUP COST (see 13.7.9.6.8)</td> </tr> <tr> <td>23:22</td> <td>DEVSLEEP/PM2 TO OFF TIME UNIT (see 13.7.9.6.7)</td> </tr> <tr> <td>21:16</td> <td>DEVSLEEP/PM2 TO OFF RECOUP COST (see 13.7.9.6.6)</td> </tr> <tr> <td>15:14</td> <td>PM2 TO PM0 TIME UNIT (see 13.7.9.6.5)</td> </tr> <tr> <td>13:8</td> <td>PM2 TO PM0 RECOUP COST (see 13.7.9.6.4)</td> </tr> <tr> <td>7:6</td> <td>PM0 TO PM2 TIME UNIT (see 13.7.9.6.3)</td> </tr> <tr> <td>5:0</td> <td>PM0 TO PM2 RECOUP COST (see 13.7.9.6.2)</td> </tr> </table>	Bit	Meaning	63	TERE SUPPORTED (see 13.7.9.6.14)	62:47	Reserved	46:45	IN-STATE SLUMBER POWER UNITS (see 13.7.9.6.13)	44:40	TYPICAL IN-STATE SLUMBER POWER (see 13.7.9.6.12)	39:38	PM2 TO PM0 LATENCY TIME UNIT (see 13.7.9.6.11)	37:32	PM2 TO PM0 LATENCY (see 13.7.9.6.10)	31:30	OFF TO GOOD STS RELATIVE TO DEVSLEEP/PM2 TIME UNIT (see 13.7.9.6.9)	29:24	OFF TO GOOD STS RELATIVE TO DEVSLEEP/PM2 RECOUP COST (see 13.7.9.6.8)	23:22	DEVSLEEP/PM2 TO OFF TIME UNIT (see 13.7.9.6.7)	21:16	DEVSLEEP/PM2 TO OFF RECOUP COST (see 13.7.9.6.6)	15:14	PM2 TO PM0 TIME UNIT (see 13.7.9.6.5)	13:8	PM2 TO PM0 RECOUP COST (see 13.7.9.6.4)	7:6	PM0 TO PM2 TIME UNIT (see 13.7.9.6.3)	5:0	PM0 TO PM2 RECOUP COST (see 13.7.9.6.2)
Bit	Meaning																															
63	TERE SUPPORTED (see 13.7.9.6.14)																															
62:47	Reserved																															
46:45	IN-STATE SLUMBER POWER UNITS (see 13.7.9.6.13)																															
44:40	TYPICAL IN-STATE SLUMBER POWER (see 13.7.9.6.12)																															
39:38	PM2 TO PM0 LATENCY TIME UNIT (see 13.7.9.6.11)																															
37:32	PM2 TO PM0 LATENCY (see 13.7.9.6.10)																															
31:30	OFF TO GOOD STS RELATIVE TO DEVSLEEP/PM2 TIME UNIT (see 13.7.9.6.9)																															
29:24	OFF TO GOOD STS RELATIVE TO DEVSLEEP/PM2 RECOUP COST (see 13.7.9.6.8)																															
23:22	DEVSLEEP/PM2 TO OFF TIME UNIT (see 13.7.9.6.7)																															
21:16	DEVSLEEP/PM2 TO OFF RECOUP COST (see 13.7.9.6.6)																															
15:14	PM2 TO PM0 TIME UNIT (see 13.7.9.6.5)																															
13:8	PM2 TO PM0 RECOUP COST (see 13.7.9.6.4)																															
7:6	PM0 TO PM2 TIME UNIT (see 13.7.9.6.3)																															
5:0	PM0 TO PM2 RECOUP COST (see 13.7.9.6.2)																															
72..511		Reserved																														

### 13.7.9.2 SATA capabilities

#### 13.7.9.2.1 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 Gbps.

IDENTIFY DEVICE data Word 77 bit 1 is a copy of this field.

#### 13.7.9.2.2 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 Gbps.

IDENTIFY DEVICE data Word 77 bit 2 is a copy of this field.

#### 13.7.9.2.3 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 Gbps.

IDENTIFY DEVICE data Word 77 bit 3 is a copy of this field.

#### 13.7.9.2.4 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 13.6.6).

IDENTIFY DEVICE data Word 77 bit 8 is a copy of this field.

#### 13.7.9.2.5 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 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 77 bit 9 is a copy of this field.

#### **13.7.9.2.6 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 77 bit 10 is a copy of this field.

#### **13.7.9.2.7 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 77 bit 11 is a copy of this field.

#### **13.7.9.2.8 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 13.7.9.2.4) is set to one.

IDENTIFY DEVICE data Word 77 bit 12 is a copy of this field.

#### **13.7.9.2.9 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. This bit shall only be set to one if the NCQ FEATURE SET SUPPORTED bit (see 13.7.9.2.4) is set to one.

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 77 bit 13 is a copy of this field.

#### **13.7.9.2.10 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. This bit shall only be set to one if the NCQ FEATURE SET SUPPORTED bit (see 13.7.9.2.4) is set to one.



HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

If the DEVICE INITIATED POWER MANAGEMENT SUPPORTED (see 13.7.9.2.17) bit is cleared to zero, then the DEVICE AUTOMATIC PARTIAL TO SLUMBER TRANSITIONS SUPPORTED bit shall be cleared to zero.

IDENTIFY DEVICE data Word 77 bit 14 is a copy of this field.

**13.7.9.2.11 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-3) and the READ LOG EXT command (see ACS-3) 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 13.7.9.2.4) 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 77 bit 15 is a copy of this field.

**13.7.9.2.12 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 13.7.9.2.4) is set to one.

IDENTIFY DEVICE data Word 78 bit 4 is a copy of this field.

**13.7.9.2.13 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 13.7.9.2.4) is set to one.

IDENTIFY DEVICE data Word 78 bit 5 is a copy of this field.

**13.7.9.2.14 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 13.7.9.2.4) is set to one.

IDENTIFY DEVICE data Word 78 bit 6 is a copy of this field.

**13.7.9.2.15 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 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 Buffer Offset field cleared to zero.

IDENTIFY DEVICE data Word 78 bit 1 is a copy of this field.

**13.7.9.2.16 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 field.

#### **13.7.9.2.17 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.

Devices shall support either host-initiated interface power management or device-initiated interface power management. 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 field.

#### **13.7.9.2.18 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 zero.

IDENTIFY DEVICE data Word 78 bit 4 is a copy of this field.

#### **13.7.9.2.19 HARDWARE FEATURE CONTROL SUPPORTED bit**

If the HARDWARE FEATURE CONTROL SUPPORTED bit is set to one, then the device supports Hardware Feature Control (see 6.13). If the HARDWARE FEATURE CONTROL SUPPORTED bit is cleared to zero, then Hardware Feature Control is not supported and the HARDWARE FEATURE CONTROL ENABLED bit shall be cleared to zero.

IDENTIFY DEVICE data Word 78 bit 5 is a copy of this field.

#### **13.7.9.2.20 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 field.

#### **13.7.9.2.21 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 13.7.9.2.4) is set to one.

IDENTIFY DEVICE data Word 78 bit 7 is a copy of this field.

**13.7.9.2.22 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; and
- c) the DEVSLP TIMING VARIABLES SUPPORTED bit (see 13.7.9.4.1) shall be set to one.

If the DEVICE SLEEP SUPPORTED bit is cleared to zero, then:

- a) the device does not support the Device Sleep feature.

IDENTIFY DEVICE data Word 78 bit 8 is a copy of this field.

NOTE 71 – 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.

**13.7.9.2.23 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 field.

**13.7.9.2.24 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 field.

**13.7.9.2.25 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 field.

**13.7.9.2.26 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 field.

### 13.7.9.3 SATA Current Settings

#### 13.7.9.3.1 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 119) that the device is currently communicating.

IDENTIFY DEVICE data Word 77 bits 3:1 is a copy of this field.

**Table 119 – 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 72 - 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.

#### 13.7.9.3.2 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 zero.

IDENTIFY DEVICE data Word 79 bit 1 is a copy of this field.

#### 13.7.9.3.3 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 zero.

IDENTIFY DEVICE data Word 79 bit 2 is a copy of this field.

#### 13.7.9.3.4 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 zero by default.

IDENTIFY DEVICE data Word 79 bit 3 is a copy of this field.

#### **13.7.9.3.5 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 13.7.9.3.2), 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 zero.

IDENTIFY DEVICE data Word 79 bit 4 is a copy of this field.

#### **13.7.9.3.6 HARDWARE FEATURE CONTROL ENABLED bit**

If the HARDWARE FEATURE CONTROL ENABLED bit is set to one, then device support for the Hardware Feature Control feature (see 13.10) is enabled. If the HARDWARE FEATURE CONTROL ENABLED bit is cleared to zero, then Hardware Feature Control is disabled.

IDENTIFY DEVICE data Word 79 bit 5 is a copy of this field.

#### **13.7.9.3.7 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 (see 13.7.9.2.20) 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 is cleared to zero, then the software settings preservation enabled bit shall be zero by default.

IDENTIFY DEVICE data Word 79 bit 6 is a copy of this field.

#### **13.7.9.3.8 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 field.

#### **13.7.9.3.9 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 field.

#### **13.7.9.3.10 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 13.7.9.2.24), then the HYBRID INFORMATION ENABLED bit shall be cleared to zero.

#### **13.7.9.3.11 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 13.7.9.2.26) and the device supports NCQ (see 13.7.9.2.4).

#### **13.7.9.3.12 CURRENT HARDWARE FEATURE CONTROL IDENTIFIER bit**

If the CURRENT HARDWARE FEATURE CONTROL IDENTIFIER bit is non-zero, then Table 104 describes the current Hardware Feature Control behavior. If the CURRENT HARDWARE FEATURE CONTROL IDENTIFIER is cleared to zero, then the current Hardware Feature Control behavior shall be either DSS or DAS.

#### **13.7.9.3.13 SUPPORTED HARDWARE FEATURE CONTROL IDENTIFIER bit**

The SUPPORTED HARDWARE FEATURE CONTROL IDENTIFIER bit (see Table 104) indicates the value that is permitted for the Current Hardware Feature Control Identifier field.

### **13.7.9.4 DEVSLP TIMING VARIABLES**

#### **13.7.9.4.1 DEVSLP TIMING VARIABLES SUPPORTED bit**

If the DEVSLP TIMING VARIABLES SUPPORTED bit is set to one, then the device supports DEVSLP TIMING VARIABLES.

#### **13.7.9.4.2 DEVSLEEP EXIT TIMEOUT field**

The DEVSLEEP EXIT TIMEOUT (DETO) field contains the maximum time, in ms, from when DEVSLP is negated, to when the device shall be ready to detect OOB signals. If the value in DETO is zero, then the host should use 20 ms as the value of DETO. See 8.5 for more information.

#### **13.7.9.4.3 MINIMUM DEVSLP ASSERTION TIME field**

The MINIMUM DEVSLP ASSERTION TIME (MDAT) field contains the minimum time, in ms that the host shall assert DEVSLP, once it has been asserted. If the value in MDAT is zero, then the host should use 10 ms as the value of MDAT. See 8.5 for more information.

### **13.7.9.5 TRANSITIONAL ENERGY REPORTING**

#### **13.7.9.5.1 TRANSITIONAL ENERGY REPORTING overview**

See ACS-3 for descriptions of PM0: Active and PM2: Standby.

The TRANSITIONAL ENERGY REPORTING field is defined as follows:

- c) the TRANSITIONAL ENERGY REPORTING field is valid only under typical device operating parameters (e.g., temperature); and

- d) 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.

**13.7.9.5.2 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 set 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 73 - 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 388) is the energy consumed to calculate the Slumber to DevSleep transitional energy time value (SLUMBER TO DEVSLEEP RECOUP COST field).

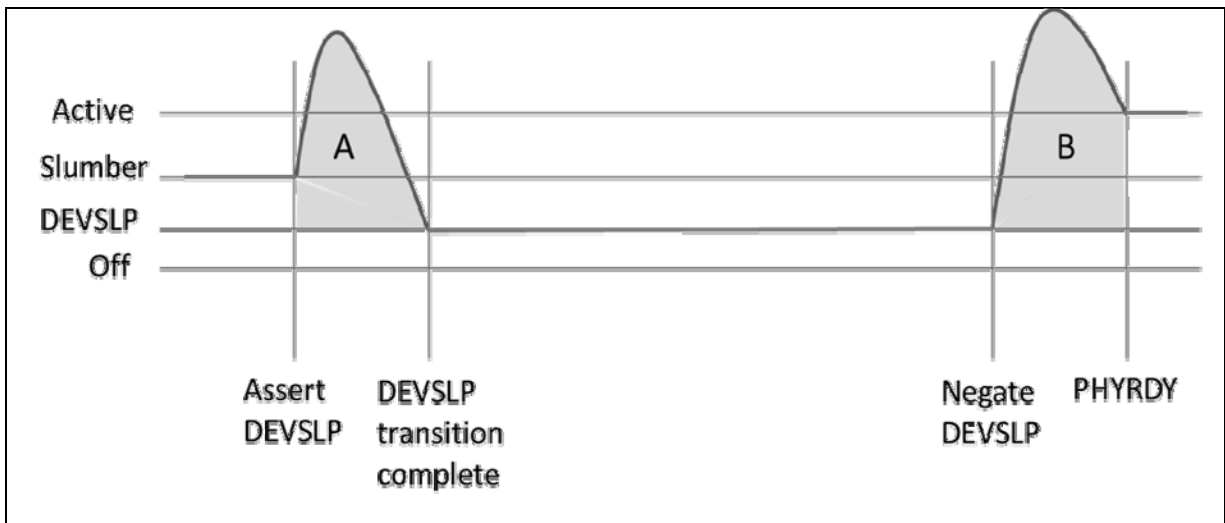


Figure 388 – Transitional Energy, Slumber to DevSleep and DevSleep to PHYRDY

**13.7.9.5.3 SLUMBER TO DEVSLEEP TIME UNIT field**

The SLUMBER TO DEVSLEEP TIME UNIT field indicates the time units as defined in Table 120 represented in the SLUMBER TO DEVSLEEP RECOUP COST field.

Table 120 – Transition Time Units

Value	Time Units
0	1 millisecond units
1	10 millisecond units
2	100 millisecond units
3	1 second units

#### 13.7.9.5.4 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 set 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 74 - 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 388) is the energy consumed to calculate the DEVSLP to PHYRDY transitional energy time value (DEVSLEEP TO PHYRDY RECOUP COST field).

#### 13.7.9.5.5 DEVSLEEP TO PHYRDY TIME UNIT field

The DEVSLEEP TO PHYRDY TIME UNIT field indicates the time units as defined in Table 120 represented in the DEVSLEEP TO PHYRDY RECOUP COST field.

#### 13.7.9.5.6 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 set 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 75 - 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 389) is the energy consumed to calculate the Off to DHR2: Send\_good\_status transitional energy time value (OFF TO GOOD STS RECOUP COST field).

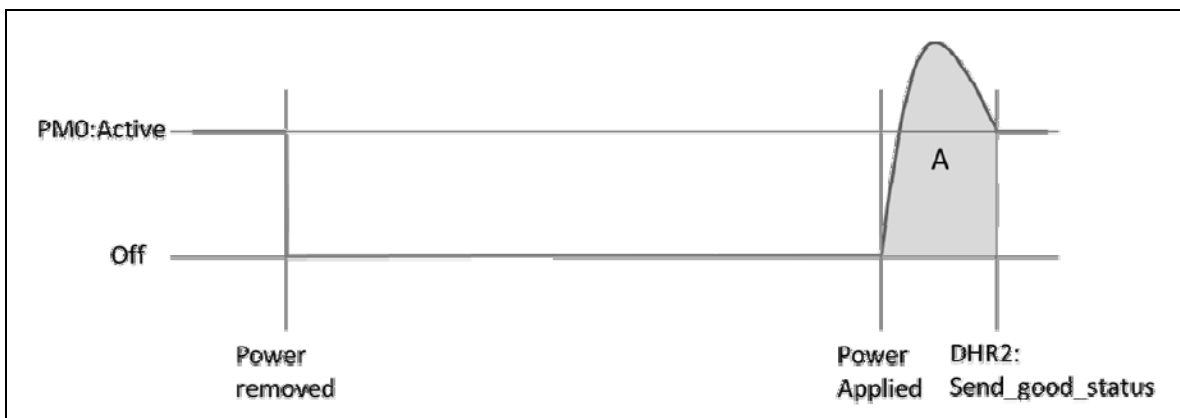


Figure 389 – Transitional Energy, PM0: Active to Off and Off to PM0: Active



**13.7.9.5.7 OFF TO GOOD STS TIME UNIT field**

The OFF TO GOOD STS TIME UNIT field indicates the time units as defined in Table 120 represented in the OFF TO GOOD STS RECOUP COST field.

**13.7.9.5.8 RECOMMENDED TIME BETWEEN POWER CYCLES field**

The RECOMMENDED TIME BETWEEN POWER CYCLES field indicates the recommended number of minimum time units between power cycles.

**13.7.9.5.9 BETWEEN POWER CYCLES TIME UNIT field**

The BETWEEN POWER CYCLES TIME UNIT field indicates the time units as defined in Table 121 represented in the RECOMMENDED TIME BETWEEN POWER CYCLES field.

**Table 121 – Power Cycle Time Units**

Value	Time Units
0	1 millisecond units
1	1 second units
2	10 second units
3	30 second units

**13.7.9.5.10 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.

**13.7.9.5.11 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 120 represented in the OFF TO GOOD STS LATENCY field.

**13.7.9.5.12 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.

**13.7.9.5.13 IN-STATE DEVSLEEP POWER UNIT field**

The IN-STATE DEVSLEEP POWER UNIT field indicates the power units as defined in Table 122 represented in the TYPICAL IN-STATE DEVSLEEP POWER field.

**Table 122 – Power units**

<b>Value</b>	<b>Power Units</b>
0	1 milliwatts units
1	10 milliwatts units
2	100 milliwatts units
3	1 watt units

#### **13.7.9.5.14 TER SUPPORTED bit**

If the TER SUPPORTED bit is set to one, then the device supports TRANSITIONAL ENERGY REPORTING.

#### **13.7.9.6 TRANSITIONAL ENERGY REPORTING EXTENDED**

##### **13.7.9.6.1 TRANSITIONAL ENERGY REPORTING EXTENDED overview**

See ACS-3 for descriptions of PM0: Active and PM2: Standby.

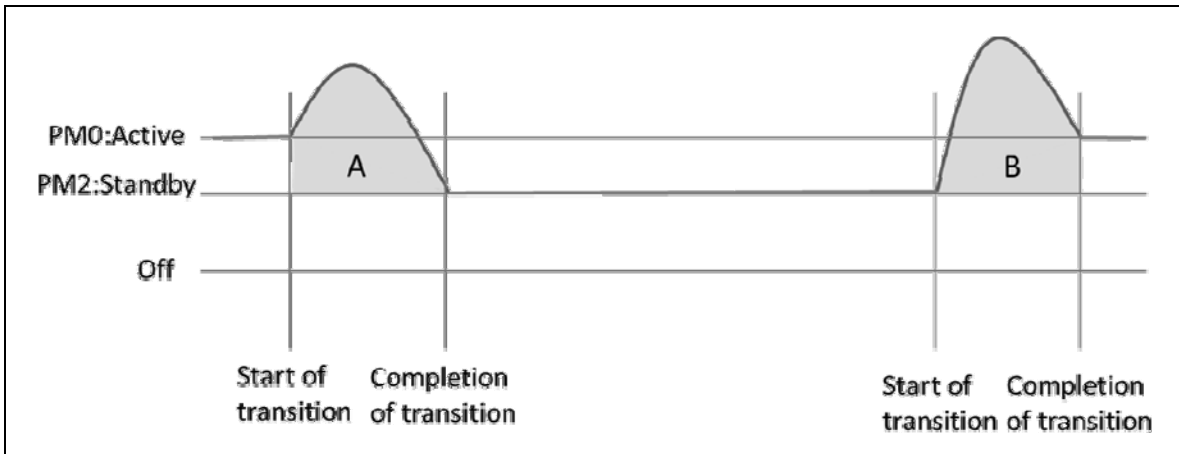
The TRANSITIONAL ENERGY REPORTING EXTENDED field is defined as follows:

- 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.

##### **13.7.9.6.2 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. If the PM0 TO PM2 RECOUP COST field is set 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 76 - The accuracy of the nominal value of the PM0 TO PM2 RECOUP COST field may change over time.



**Figure 390 – Transitional Energy, PM0: Active to PM2: Standby and PM2: Standby to PM0: Active**

#### 13.7.9.6.3 PM0 TO PM2 TIME UNIT field

The PM0 TO PM2 TIME UNIT field indicates the time units as defined in Table 120 represented in the PM0 TO PM2 RECOUP COST field.

#### 13.7.9.6.4 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 set 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 77 - The accuracy of the nominal value of the PM2 TO PM0 RECOUP COST field may change over time.

#### 13.7.9.6.5 PM2 TO PM0 TIME UNIT field

The PM2 TO PM0 TIME UNIT field indicates the time units as defined in Table 120 represented in the PM2 TO PM0 RECOUP COST field.

#### 13.7.9.6.6 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. If the DEVSLEEP/PM2 TO OFF RECOUP COST field is set 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 78 - The accuracy of the nominal value of the DEVSLEEP/PM2 TO OFF RECOUP COST field may change over time.

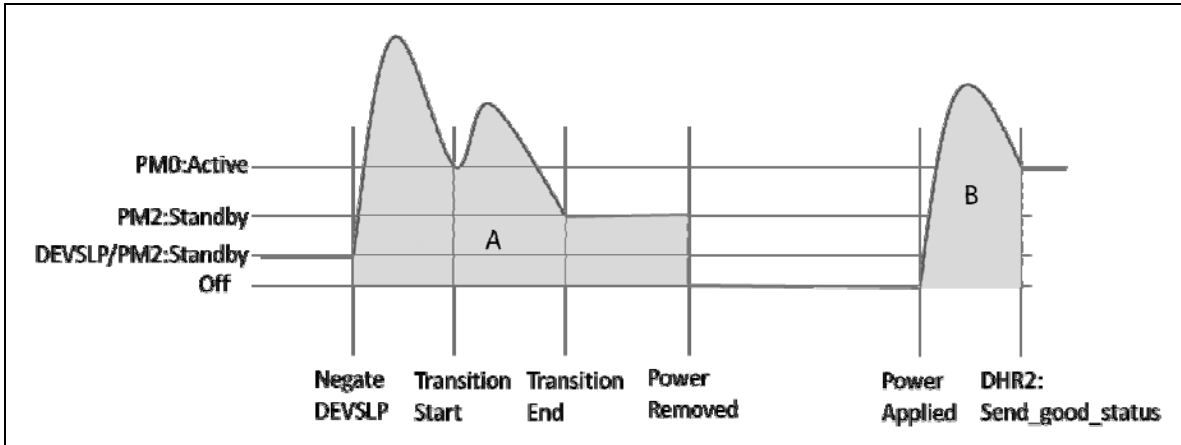


Figure 391 – Transitional Energy, DevSleep/PM2: Standby to PM0: Active and Off to DHR2:Send\_good\_status

#### 13.7.9.6.7 DEVSLEEP/PM2 TO OFF TIME UNIT field

The DEVSLEEP/PM2 TO OFF TIME UNIT field indicates the time units as defined in Table 120 represented in the DEVSLEEP/PM2 TO OFF RECOUP COST field.

#### 13.7.9.6.8 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 set 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.

#### 13.7.9.6.9 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 120 represented in the OFF TO GOOD STS RELATIVE TO DEVSLEEP/PM2 RECOUP COST field.

#### 13.7.9.6.10 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.

#### 13.7.9.6.11 PM2 TO PM0 LATENCY TIME UNIT field

The PM2 TO PM0 LATENCY TIME UNIT field indicates the time units as defined in Table 120 represented in the PM2 TO PM0 LATENCY field.

#### **13.7.9.6.12 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.

#### **13.7.9.6.13 IN-STATE SLUMBER POWER UNITS field**

The IN-STATE SLUMBER POWER UNITS field indicates the power units as defined in Table 122 represented in the TYPICAL IN-STATE SLUMBER POWER field.

#### **13.7.9.6.14 TERE SUPPORTED bit**

If the TERE SUPPORTED bit is set to one, then the device supports TRANSITIONAL ENERGY REPORTING EXTENDED.

### **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 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 with the N bit set, 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 Notification 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 392) defines the required behavior if the device supports Asynchronous Notification. A device shall only send a Set Device Bits FIS with the Notification 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, current Status and Error values, and all Reserved fields cleared to zero. Set NotifyPending to 1.
1. Unconditional	→ DI0: Device_idle

**Figure 392 – 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, current Status and Error field values, 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 393.

NOTE 79 - 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<sub>P</sub> event is recorded; a missing EOF<sub>P</sub> 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-3 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 Features(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 393 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; 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 ICRC error bit set to one in Error register
002h	Optional	R_ERR <sub>P</sub> response for Data FIS
003h	Optional	R_ERR <sub>P</sub> response for Device-to-Host Data FIS
004h	Optional	R_ERR <sub>P</sub> response for Host-to-Device Data FIS
005h	Optional	R_ERR <sub>P</sub> response for non-Data FIS
006h	Optional	R_ERR <sub>P</sub> response for Device-to-Host non-Data FIS
007h	Optional	R_ERR <sub>P</sub> 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 <sub>P</sub> response for Host-to-Device Data FIS due to CRC errors
010h	Optional	R_ERR <sub>P</sub> response for Host-to-Device Data FIS due to non-CRC errors
012h	Optional	R_ERR <sub>P</sub> response for Host-to-Device non-Data FIS due to CRC errors
013h	Optional	R_ERR <sub>P</sub> 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 <sub>P</sub> 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 393 – Phy Event Counter identifiers**

FISes that are terminated due to reception of SYNC<sub>P</sub> primitives before the end of the FIS (a SYNC Escape) are not counted in the R\_ERR<sub>P</sub> 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<sub>p</sub>) and (the number of received Host-to-Device Data FISes that the device responded with R\_ERR<sub>p</sub>). 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<sub>p</sub>.

#### **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<sub>p</sub>. 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<sub>p</sub>) and (the number of received Host-to-Device non-Data FISes that the device responded with R\_ERR<sub>p</sub>). 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<sub>p</sub>. 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<sub>p</sub>. 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<sub>p</sub>.

#### **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<sub>p</sub> 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<sub>p</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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 394.

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							
...	...							
n	Counter n Identifier							
n+1								
n+2	Counter n Value							
n+ Counter n Length								
...	...							
508	Reserved							
509								
510								
511	Data Structure Checksum							

**Figure 394 – Phy event counters log data structure definition**

#### Counter n Identifier

Phy event counter identifier that corresponds to Counter n Value. Specifies the particular event counter that is being reported. The Identifier is 16 bits in length. Valid identifiers are listed in Figure 393.

#### Counter n Value

Value of the Phy event counter that corresponds to Counter n Identifier. The number of significant bits is determined by Counter n Identifier bits 14:12 (see 13.9.3). The length of Counter n Value shall be a multiple of 16 bits. The counter shall stop (and not wrap to zero) after reaching its maximum value.

#### Counter n Length

Size of the Phy event counter as defined by bits 14:12 of Counter n Identifier. The size of the Phy event counter shall be a multiple of 16 bits.

#### Data Structure Checksum

The data structure checksum 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.

**Reserved** All reserved fields shall be cleared to zero

### 13.10 Hardware Feature Control (optional)

In Serial ATA Revision 3.0 and previous specifications, Hardware Feature Control 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).

Table 123 specifies the pins used by Hardware Feature Control for various connectors

**Table 123 – Pins used by Hardware Feature Control**

Standard Connector (3.5 inch & 2.5 inch)	1.8 inch Micro SATA Connector <sup>a</sup>	LIF-SATA Connector
Pin P11: a) DSS; b) DAS; c) DHU; or d) other vendor specific.	Pin P7: a) DAS; b) DHU; or c) other vendor specific.	Pin P8: a) DSS; b) DAS; or c) other vendor specific. Pin P21: a) DHU
<sup>a</sup> DSS is not defined for 1.8 inch Micro SATA Connector.		

If Hardware Feature Control is supported, then:

- a) IDENTIFY DEVICE data Word 78 bit 5 (see 13.2.2.19) shall be set to one;
- b) the SET FEATURES Select Hardware Feature Control subcommand shall be supported (see 13.3.9);
- c) page 08h of the Identify Device Data log (see 13.7.9) 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 in the Identify Device Data log shall be cleared to zero; and
  - C) See 6.13 for requirements 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 in the Identify Device Data log shall be non-zero;
  - C) the Supported Hardware Feature Control Identifier in the Identify Device Data log shall be non-zero; and
  - D) the behavior of the Hardware Feature Control is specified by the SET FEATURES Enable Hardware Feature Control subcommand.

If Hardware Feature Control is 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 Select Hardware Feature Control subcommand shall not be supported (see 13.3.9);
- c) the Supported Hardware Feature Control Identifier (see 13.7.9) in the Identify Device Data log shall be cleared to zero;
- d) the Current Hardware Feature Control Identifier (see 13.7.9) in the Identify Device Data log shall be cleared to zero; and
- e) See 6.13.1.2 for requirements 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

## HIGH SPEED SERIALIZED AT ATTACHMENT

### Serial ATA International Organization

desirable to provide a simple mechanism that the storage subsystem controller(s) may sequence disk device initialization and spin up.

NOTE 80 - 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 spin up 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-3 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 “black-box” 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 “black box” 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-3 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<sup>2</sup>C 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<sup>2</sup>C 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 395 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<sup>2</sup>C in the figure). For an intelligent PCI RAID controller, the SEMB may be firmware running on the RAID processor and associated design-specific I<sup>2</sup>C controller interface logic, while for a Port Multiplier, the SEMB is able to be controller logic that bridges the I<sup>2</sup>C 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).



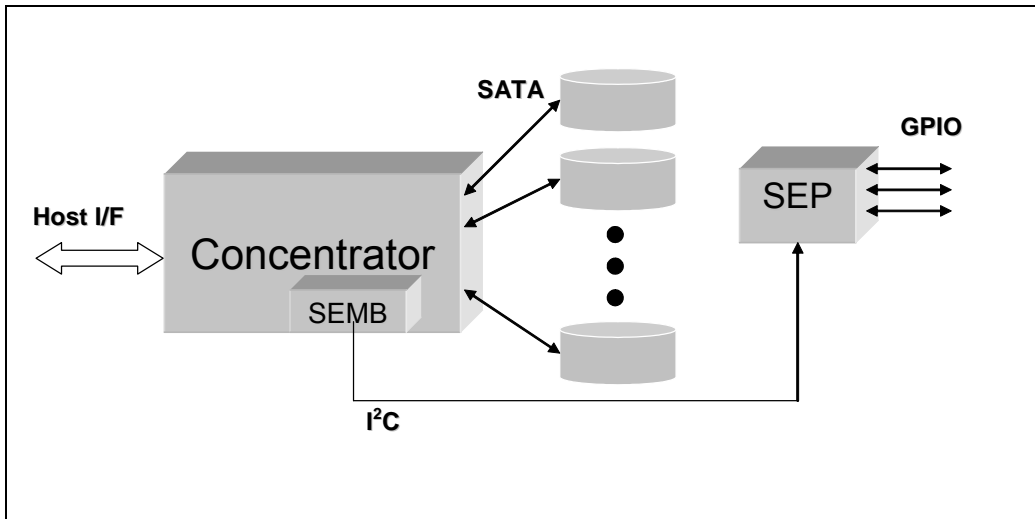


Figure 395 – Generic enclosure services topology

### 13.14.2.2 Definition configuration

The definition configuration is distilled from the generic topology in Figure 395 where the extraneous elements have been removed for the sake of definition clarity. 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.

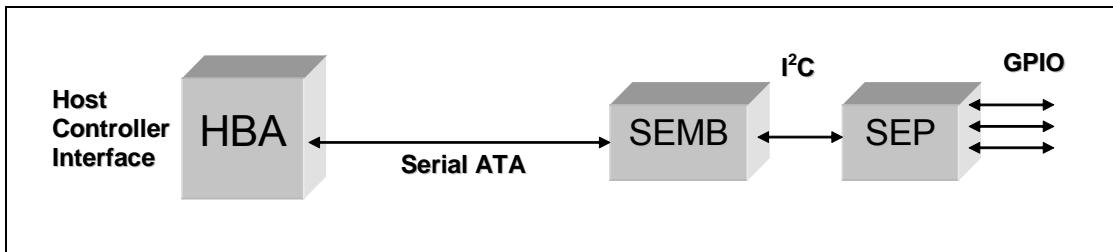
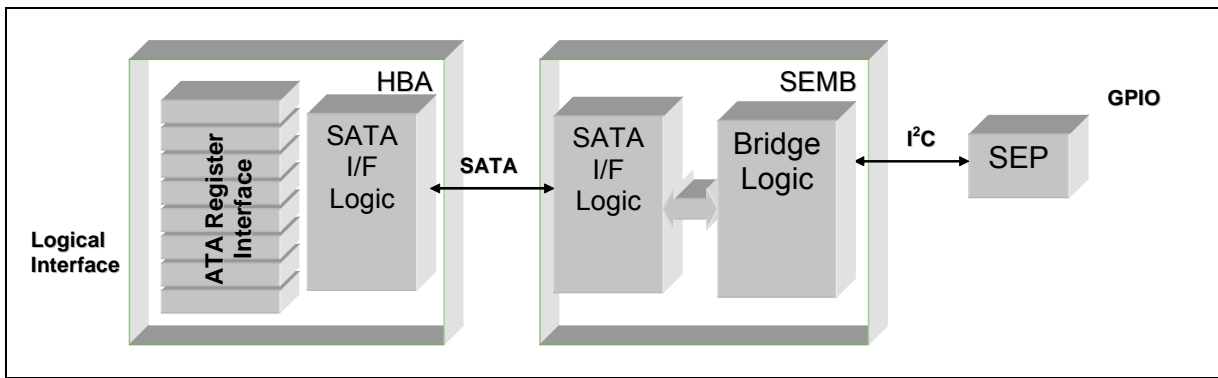


Figure 396 – Simplified view of generic topology

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 397 that presents the enclosure services facility as an ATA device with a Command Block Register interface.



**Figure 397 – 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<sup>2</sup>C 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 interface that 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<sup>2</sup>C 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, 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

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

command structure as the existing ATA protocol has been used. All SEP commands are issued to the SEMB with the SEP\_ATTEN opcode in the Command register and the actual SEP command is passed as a parameter in Features(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 398 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)	3Ch							
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 398 – 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 399 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 399 – 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 I<sup>2</sup>C mapping

#### 13.14.4.3.1 Logical command block registers to I<sup>2</sup>C 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 I<sup>2</sup>C 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 I<sup>2</sup>C interconnect, the SEP command fields are packaged as an I<sup>2</sup>C frame and forwarded to the SEP via the I<sup>2</sup>C 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 I<sup>2</sup>C packet based on the contents of the Command Block Registers or Register Host to Device FIS. The mapping of Command Block Registers to transmitted I<sup>2</sup>C 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 I<sup>2</sup>C packet shall be as indicated in Figure 400.

Field	7	6	5	4	3	2	1	0
Features(7:0)	SEP_CMD							
Features(15:8)	Reserved							
Count(7:0)	LEN							
Count(15:8)	Reserved							
LBA(7:0)	CMD_TYPE							
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 400 – Command Block Register fields used in enclosure processor communications**

SEP\_CMD The SAF-TE or SES command code to be issued in conjunction with the command type specified in CMD\_TYPE.

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

SAF-TE READ BUFFER usage, SEP\_CMD 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, SEP\_CMD 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, SEP\_CMD 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, SEP\_CMD 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, SEP\_CMD is equal to ECh.

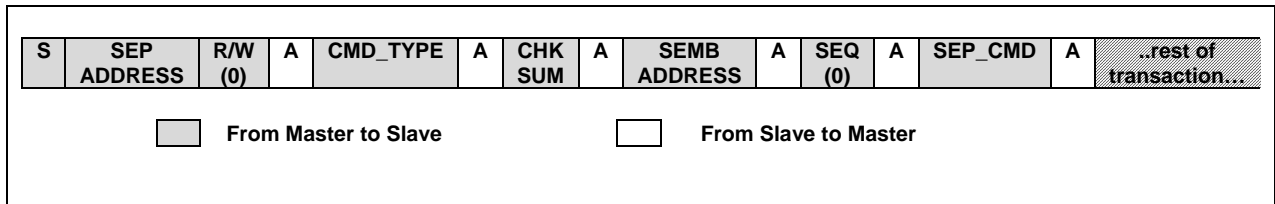
**LEN** The transfer length of the data transfer phase of the command in Dword units. Valid values are 1..255 (yielding a maximum transfer length of 1 020 bytes). Data transfers that are not a multiple of 4 bytes shall be padded by the transmitter with zeros to the next 4 byte (Dword) granularity.

**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<sup>2</sup>C frame for delivering a command is as indicated in Figure 401.



**Figure 401 – I<sup>2</sup>C frame for conveying an enclosure services command**

The SEMB need not support any Command register writes with values other than SEP\_ATTEN. In response to a Command register write of a value other than SEP\_ATTEN, 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<sup>2</sup>C 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<sup>2</sup>C 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 and CMD\_TYPE byte 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.

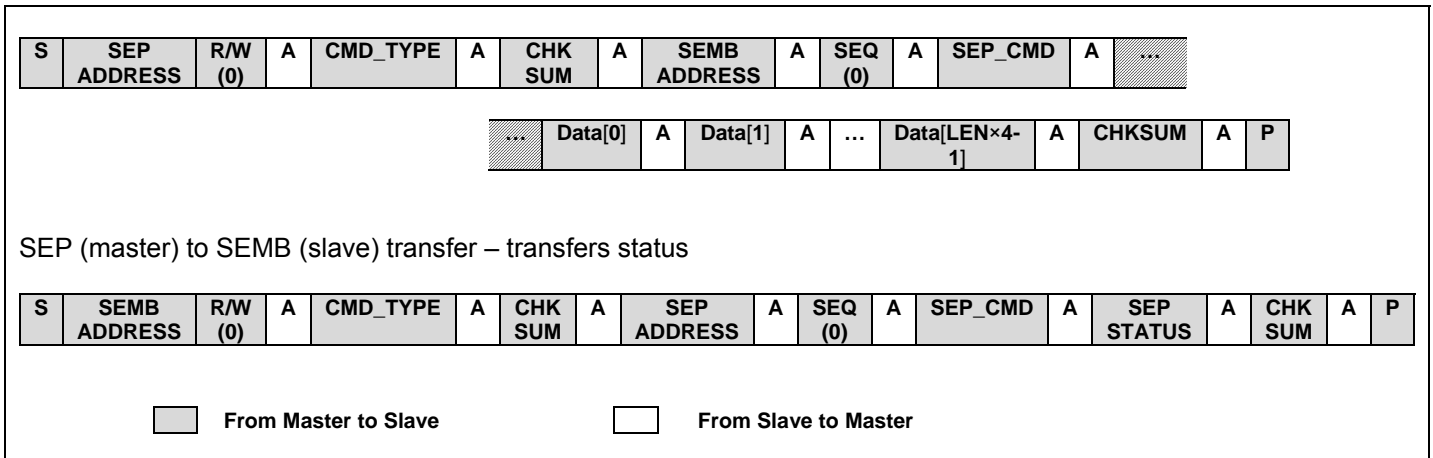
All SEP commands are issued using the SEP\_ATTN command (opcode 67h) in the Command register and the SEP command code in Features(7:0) as illustrated in the Command Block Registers image of Figure 402. The CMD\_TYPE field identifies whether the issued SEP command is a read or a write and whether the command protocol is SAF-TE or SES.

Field	7	6	5	4	3	2	1	0
Features(7:0)	SEP_CMD							
Features(15:8)	Reserved							
Count(7:0)	LEN							
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 402 – 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<sup>2</sup>C interface, the transmitted I<sup>2</sup>C packets shall be of the form indicated in Figure 403.

SEMB (master) to SEP (slave) transfer – transfers both the command and data



**Figure 403 – I<sup>2</sup>C transactions corresponding to a WRITE SEP command**

The I<sup>2</sup>C transport for Serial ATA enclosure service traffic shall be Master to Slave (IPMB) compliant. See the IPMB and I<sup>2</sup>C references for details on Master to Slave 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.

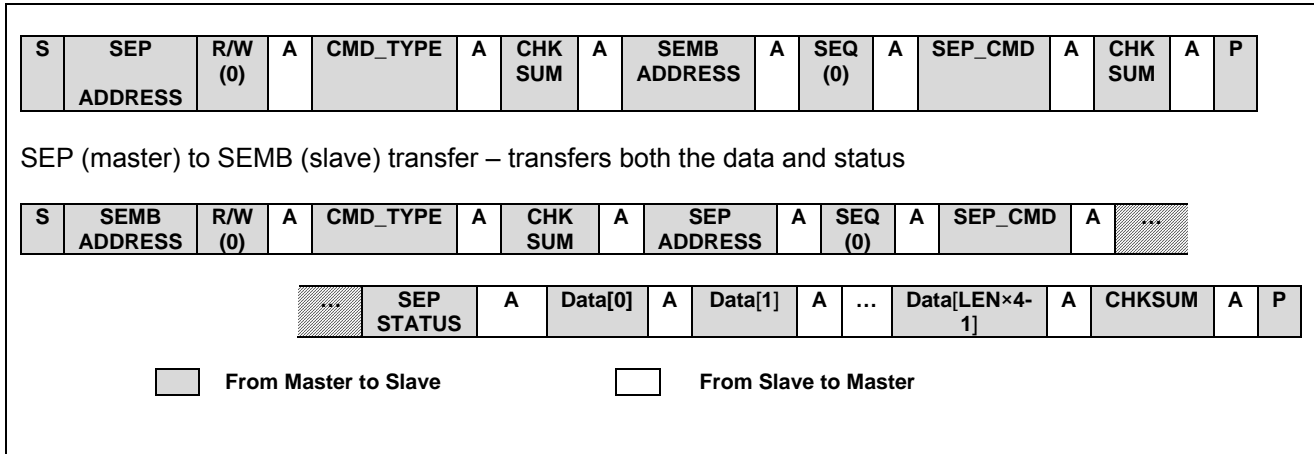
All SEP commands are issued using the SEP\_ATTN command (opcode 67h) in the Command register and the SEP command code in Features(7:0) as illustrated in the Command Block Registers image of Figure 404. The CMD\_TYPE field identifies whether the issued SEP command is a read or a write and whether the command protocol is SAF-TE or SES.

Field	7	6	5	4	3	2	1	0
Features(7:0)	SEP_CMD							
Features(15:8)	Reserved							
Count(7:0)	LEN							
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 404 – 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<sup>2</sup>C interface, the transmitted I<sup>2</sup>C packets shall be of the form indicated in Figure 405.

SEMB (master) to SEP (slave) transfer – transfer the command



**Figure 405 – I<sup>2</sup>C 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\_ATTEN command with the CMD\_TYPE 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 through 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 406 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 larger VENDOR SPECIFIC ENCLOSURE INFORMATION.



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
1	SUB-ENCLOSURE IDENTIFIER
2..9	ENCLOSURE LOGICAL IDENTIFIER
10..17	ENCLOSURE VENDOR IDENTIFICATION
18..33	PRODUCT IDENTIFICATION
34..37	PRODUCT REVISION LEVEL
38	CHANNEL IDENTIFIER
39..42	FIRMWARE REVISION LEVEL
43..48	INTERFACE IDENTIFICATION STRING
49..52	INTERFACE SPECIFICATION REVISION LEVEL
53..63	VENDOR SPECIFIC ENCLOSURE INFORMATION

**Figure 406 – IDENTIFY SEP data structure definition**

**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 NAA identifier, the format that is defined in the SCSI Primary Commands reference. The ENCLOSURE LOGICAL IDENTIFIER 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 and PRODUCT IDENTIFICATION fields, 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 SCSI Primary Commands reference). The ENCLOSURE VENDOR IDENTIFICATION 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 SCSI Primary Commands reference). 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 SCSI Primary Commands reference). The PRODUCT REVISION LEVEL 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 407.

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 407 – SAF-TE write device slot status data structure**

The Serial ATA modification to the definition of the fields is as follows:

- a) Slot d Byte 0 As defined in SAF-TE;
- b) Slot d Byte 1 Modified from definition in SAF-TE;
  - A) Bit 0..1 As defined in SAF-TE;
  - B) 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
  - C) Bit 3..7 As defined in SAF-TE;
 and
- c) 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 408. The data structure is modified with the addition of the device activity control bit (DR\_ACT) 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							
1	Reserved							
2	DR_ACT	DO NOT REMOVE	Reserved		RQST INSERT	RQST REMOVE	RQST IDENT	Res
3	Reserved		RQST FAULT	DEVICE OFF	ENABLE BYP A	ENABLE BYP B	Reserved	

**Figure 408 – 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 409 illustrates a configuration where there is no direct correspondence between host connection and enclosure device slot.

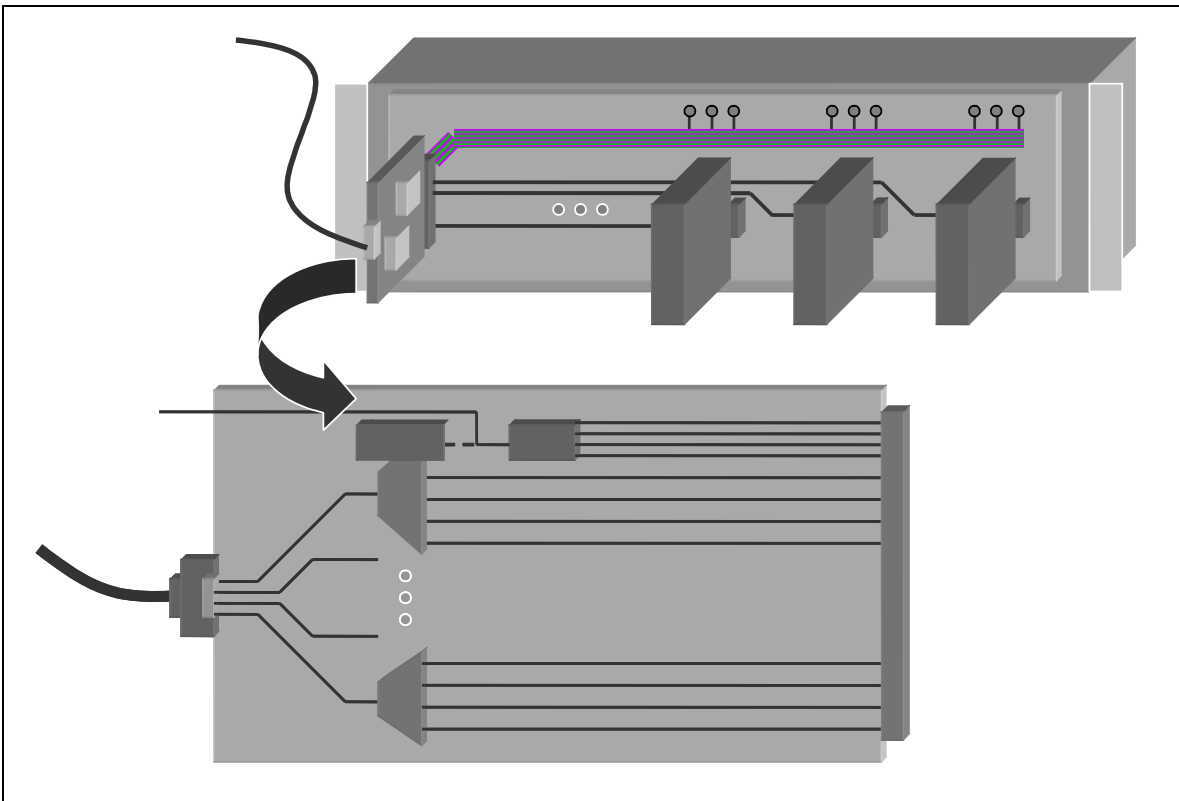


Figure 409 – Example subsystem

##### 13.14.5.3.2 SAF-TE correspondence definition (optional)

Figure 410 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 0.

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				Channel			

**Figure 410 – SAF-TE device ID field convention**

- 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 value is assigned serially starting at zero and is indicated on the host connections packaging accordingly.
- 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.

For the configuration in Figure 409, 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 411 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 411. 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				Channel			

**Figure 411 – SES slot address field convention**

- 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 value is assigned serially starting at zero and is indicated on the host connections packaging accordingly.

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.

For the configuration in Figure 409, 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<sup>2</sup>C. 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<sup>2</sup>C 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<sup>2</sup>C 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<sup>2</sup>C 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<sup>2</sup>C 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<sup>2</sup>C 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<sup>2</sup>C interface with the SEMB and should not share the I<sup>2</sup>C 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<sup>2</sup>C bus attached to a SEMB in a storage subsystem shall be assigned I<sup>2</sup>C address C0h.

NOTE 81 - Note that the LSB of the I<sup>2</sup>C address field is the R/W bit effectively resulting in the I<sup>2</sup>C 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<sup>2</sup>C interface in a storage subsystem shall be assigned consecutively higher I<sup>2</sup>C 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 412.

```
//      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 (MASTER_SLAVE_EMULATION_ENABLED && SLAVE_PRESENT)
{
    if (RESET)
        SLAVE_LED = ON;
}
else
    SLAVE_LED = OFF;

if (REGISTER_FIS_TRANSMITTED_ON_SLAVE_CHANNEL)
    SLAVE_LED = OFF;

LED = ACTIVITY_LED || SLAVE_LED;
```

**Figure 412 – 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 413 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 414 is identical in configuration to that shown in Figure 413, except that it shows the use of a ribbon cable for ease of assembly.

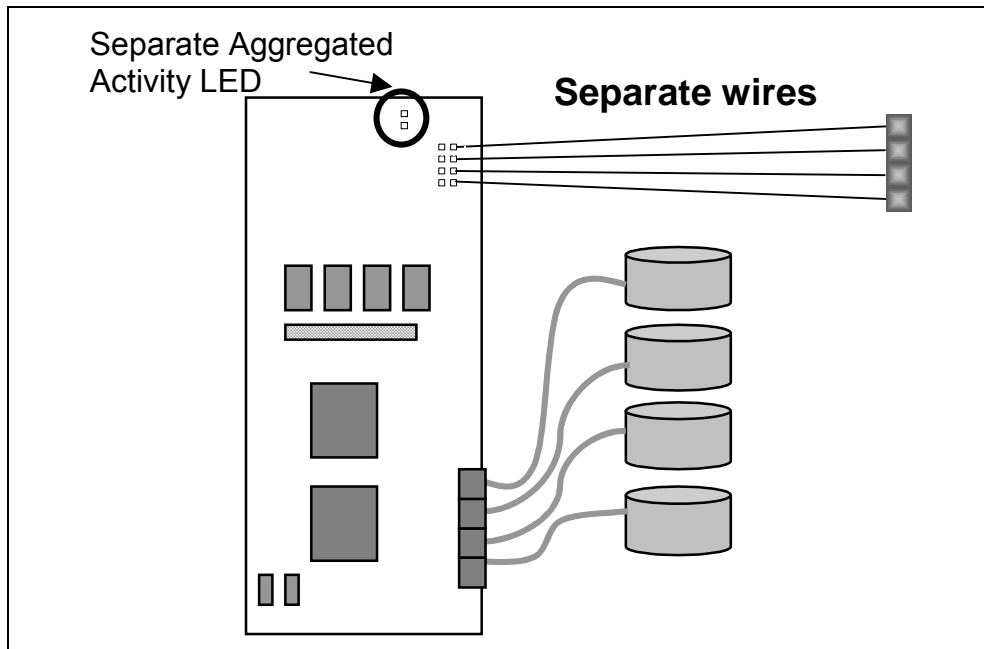
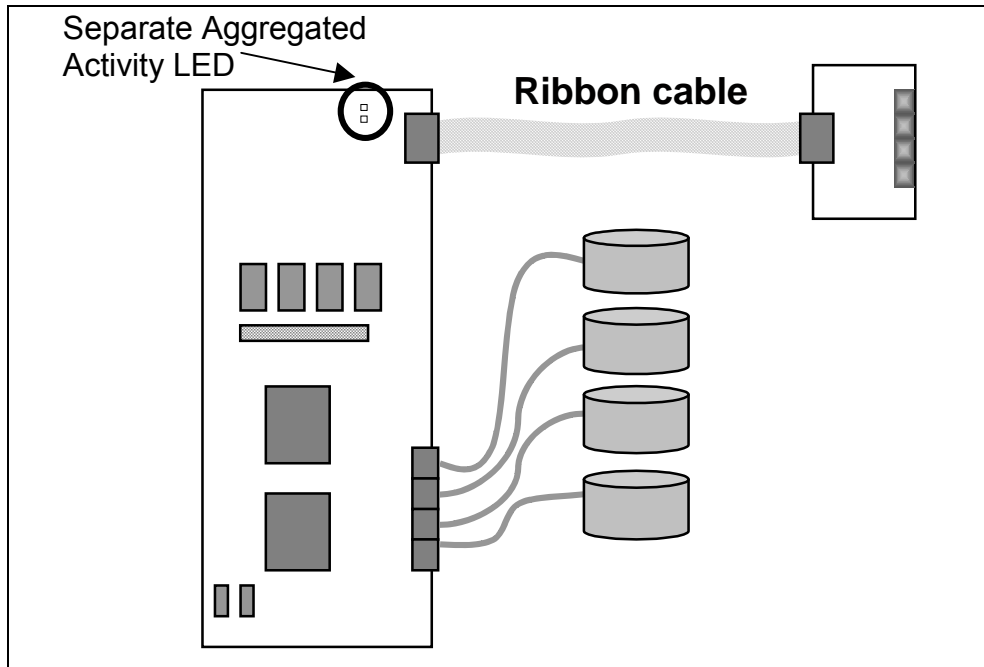


Figure 413 – Device activity LEDs with separate wires



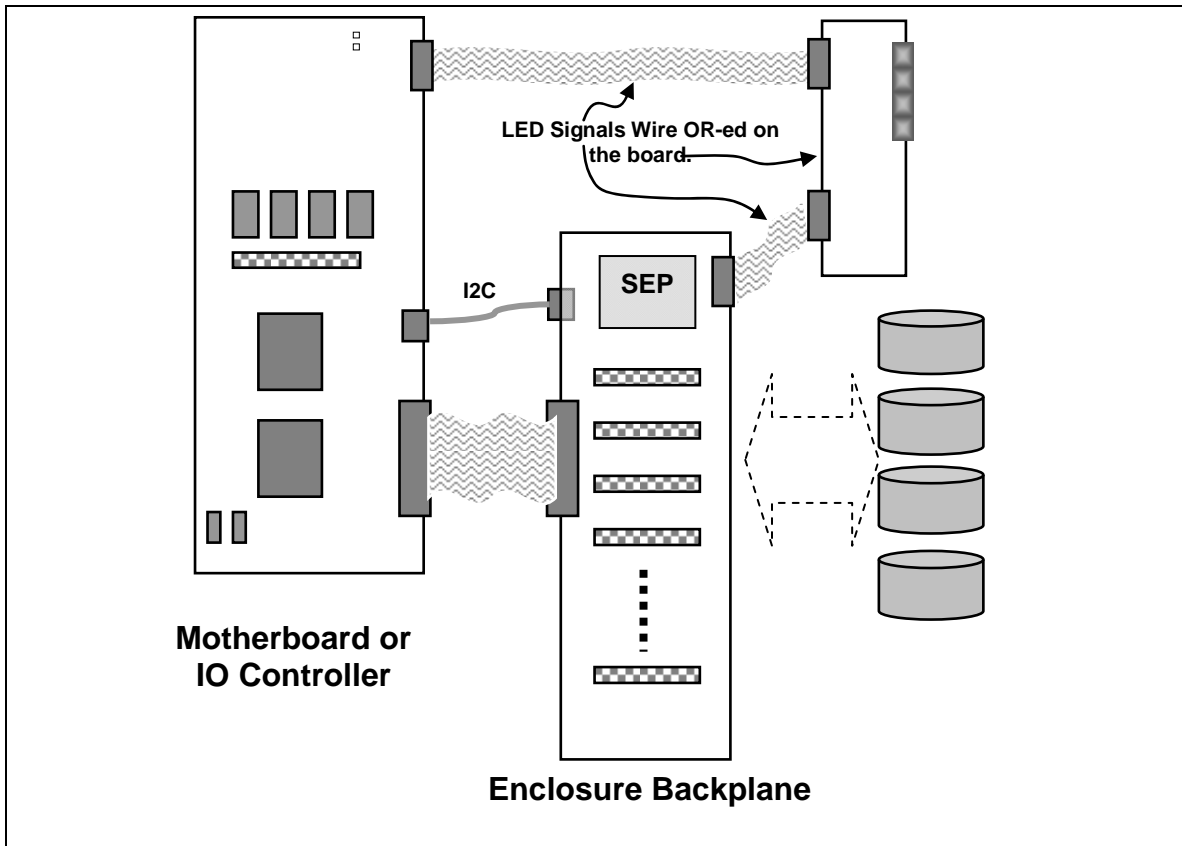


**Figure 414 – Device activity LEDs with ribbon cable**

NOTE 82 - 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 415 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 415 – 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 416.

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 416 – Software reset to control port result values**

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 (see 16.3.3.8.7).

### 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 should 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 417 .

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 417 – 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 or 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  $V_{DHUnegate}$  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 13.7.7.2.11) 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 13.7.7);
  - 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-3);
  - B) the following SET FEATURES subcommands:
    - a) Enable/disable the POWER UP IN STANDBY feature set (see ACS-3); 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 13.7.5);
    - b) the NCQ Send and Receive log (see 13.7.6);
    - c) the Identify Device Data log (see 13.7.9); and
    - d) the Hybrid Information log (see 13.7.7);
- 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-3);
- d) the following commands shall be able to return command completion without error while in the PM2: Standby (see ACS-3) state or in the PM5: PUIS and spinup subcommand not supported state (see ACS-3):



HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

- 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);
    - 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 418 describes the mapping of the Hybrid Information field that is transported in the Auxiliary(23:16) field 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	13.20.2.2
20	Reserved	
21	Hybrid Information Is Valid	
22..23	Reserved	

**Figure 418 – Hybrid Information field**

### **13.20.2.2 Hybrid Priority**

#### **13.20.2.2.1 Hybrid Priority 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 13.7.7.2.8), 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**

##### **13.20.2.2.2.1 Hybrid Priority field is equal to Maximum Hybrid Priority Level 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 13.7.7.2.11) in the Hybrid Information log.

##### **13.20.2.2.2.2 Max Priority Behavior 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., ASC field and ASCQ 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 is cleared to zero and Cache Behavior 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 Cache Behavior 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 is cleared to zero and Cache Behavior 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 Cache Behavior bit is set to one; and
- b) 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 Cache Behavior bit is set to one; and
- d) the disk is spinning,

then:

- a) 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 and is greater than zero**

#### **13.20.2.2.3.1 Cache Behavior 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 Cache Behavior 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 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 Cache Behavior 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 Cache Behavior bit is set to one; and
- c) the disk is spinning,

then:

- a) the device shall behave as if the Cache Behavior bit is cleared to zero (see 13.20.2.2.3.1).

#### **13.20.2.2.3.3 Hybrid priority 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) Dirty Low Threshold (see 13.7.7.2.5); and
- b) Dirty High Threshold (see 13.7.7.2.6).

The device should prioritize the logical sectors contained in the non-volatile caching medium to be synced as follows:

- 1) starting from the Hybrid Priority zero, up to the Maximum Hybrid Priority Level;
- 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., ASC field and ASCQ field) shall be set to LOGICAL UNIT NOT READY – INITIALIZING COMMAND REQUIRED.

If the device indicates a sense key 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-3) with the Hold Power Condition bit cleared to zero, then see Table 124 for interactions with the current power source reported by the device.

**Table 124 – Current power source interactions**

Current Power Source <sup>a</sup>	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.
<sup>a</sup> 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

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 13.7.7.2.4) 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(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

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 13.7.7.2.4) of the Hybrid Information log,

The device may clear the Read Only bit under vendor specific conditions.

### 13.20.5.3 Data Loss

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 13.7.7.2.4) 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(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

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 13.7.7.2.4) 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(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 13.7.7.2.3) 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 field 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 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 field 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 83 - If the host receives sense data with sense key set to MEDIUM ERROR and additional sense code set to UNRECOVERED READ ERROR, then the host should 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 84 - If the host receives sense data with sense key set to ABORTED COMMAND and additional sense code set to MULTIPLE READ ERRORS, then the host should 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, 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 85 - If the host receives sense data with sense key set to ABORTED COMMAND and additional sense code set to MULTIPLE WRITE ERRORS, then the host should 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 13.7.8) with the Rebuild Assist Enabled bit cleared to zero.

### 13.21.5 Testing the Rebuild Assist feature

The Rebuild Assist log (see 13.7.8) 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 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).



## 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 125 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 master/slave emulation the host adapter may implement.

**Table 125 – SCR definition**

				ATA Command Block and Control Block registers			
				Read		Write	
CS 0	A2	A1	A0	Data		Data	
	0	0	0	Error		Features	
	0	0	1	Count(7:0)		Count(7:0)	
	0	1	0	[15:8]	[7:0]	[15:8]	[7:0]
	0	1	1	[31:24]	[7:0]	[31:24]	[7:0]
	1	0	0	[39:32]	[15:8]	[39:32]	[15:8]
	1	0	1	[47:40]	[23:16]	[47:40]	[23:16]
	1	1	0	Device		Device	
CS 1	1	1	1	Status		Command	
	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 126 defines the Serial ATA Status and Control registers.

**Table 126 – 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 419). 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 419 – SStatus register definition**

- DET** The DET 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
- SPD** The SPD 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
- IPM** The IPM 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
- Reserved** All reserved fields shall be cleared to zero.

NOTE 86 - 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 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 87 - The SStatus register's IPM 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 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 (SErrror) 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 420). The register represents all the detected errors accumulated since the last time the SError register was cleared to zero (whether recovered by the interface or 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-up 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 420 – SError register definition

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 421).

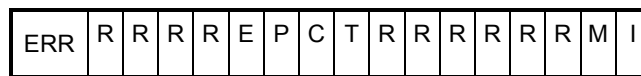


Figure 421 – ERR field definition

- C Non-recovered persistent communication 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.
- E Internal error, the host bus adapter experienced an internal error that caused the operation to fail and may have put the host bus adapter into an error state. Host software should reset the interface before re-trying the operation. If the condition persists, the host bus adapter may suffer from a design issue rendering it incompatible with the attached device.
- I Recovered data integrity 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.

- M Recovered communications 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.
  - P Protocol 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 host bus adapter.
  - R Reserved bit for future use, shall be cleared to zero.
  - T Non-recovered transient 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.
- 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 422).



**Figure 422 – DIAG field definition**

- A Port Selector presence detected, this bit is set to one if COMWAKE is received while the host is in state HP2: HR\_AwaitCOMINIT.
- B 10b/8b 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.
- C CRC 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 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.
- F Unrecognized FIS 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 type field that was not recognized.
- I Phy Internal 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, if set to one, this bit indicates that the PHYRDY signal changed state since the last time this bit was cleared to zero.
- H Handshake error, if set to one, this bit indicates that one or more R\_ERR<sub>P</sub> 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.
- R Reserved bit for future use, Shall be cleared to zero.
- S Link Sequence 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.
- T Transport 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.
- W COMWAKE 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.

- X Exchanged, if set to one this bit indicates that device presence has changed since the last time this bit was cleared to zero. This 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.

#### 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 423). 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.

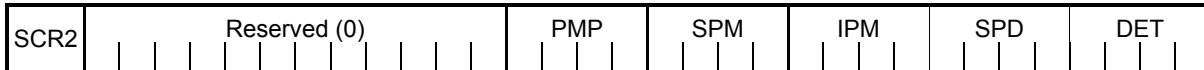


Figure 423 – SControl register definition

- 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 259) 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
  
- 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
  
- 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
  
- 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

**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.

**Reserved** All reserved fields shall be cleared to zero.

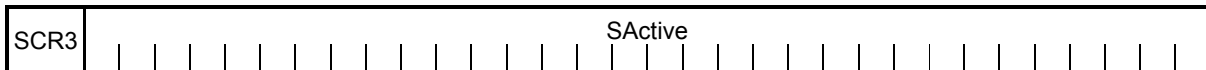
### 14.2.5 SActive register

The SActive register is a 32 bit register that conveys the information returned in the SActive field of the Set Device Bits FIS. 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 424 – SActive register definition**

**SActive** For the Native Command Queuing protocol, the SActive 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 Notification bit set, as defined in 10.5.7. If the host receives a Set Device Bits FIS with the Notification 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 shall set 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.



HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

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 425 – SNotification register definition**

**Notify** The field represents whether a particular device with the corresponding PM Port number has sent a Set Device Bits FIS to the host with the Notification bit set to one.

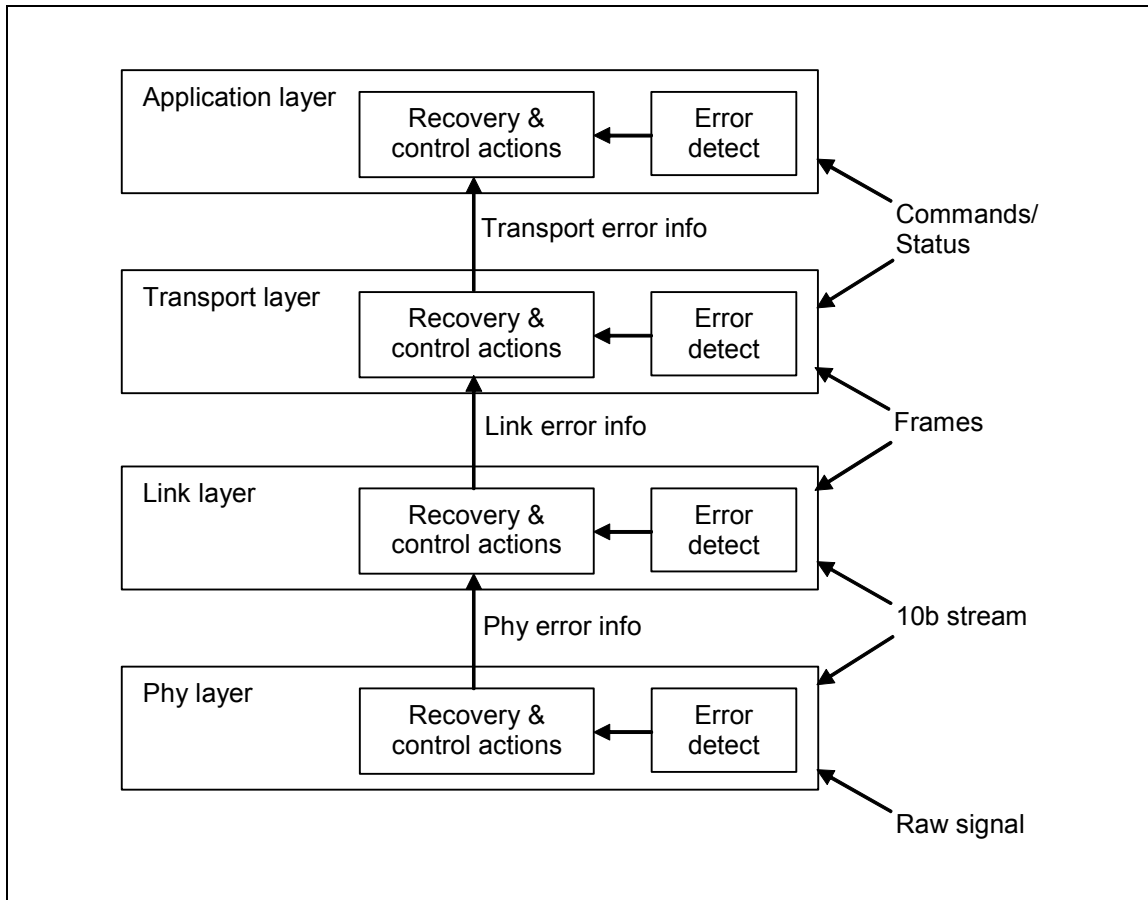
**Reserved** The reserved field shall be cleared to zero.



## 15 Error handling

### 15.1 Architecture

The layered architecture of Serial ATA extends to error handling as well. As indicated in Figure 426, 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 426 – 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 7.7.

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<sub>p</sub> patterns to bring the communications link up between host controller and device. The specific sequences as defined in 7.7.

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 88 - 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" 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 responds to many such error conditions as defined in clause 0. 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 9.6 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 a  $R\_OK_P$  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<sub>P</sub>), 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<sub>P</sub>, 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<sub>P</sub>, 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<sub>P</sub>/R\_OK<sub>P</sub> handshake at the end of each frame. Devices shall reflect any R\_ERR<sub>P</sub> 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; and
- c) protocol errors & 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 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<sub>P</sub> 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<sub>P</sub> 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 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.



## HIGH SPEED SERIALIZED AT ATTACHMENT

### Serial ATA International Organization

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 host bus adapter'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 TYPE value, the Transport layer shall signal the Link layer to negatively acknowledge the frame reception by asserting R\_ERR<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> is not detected two Dwords later (the CRC falls between the last data Dword and EOF<sub>P</sub>), 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<sub>P</sub> 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

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

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 427.

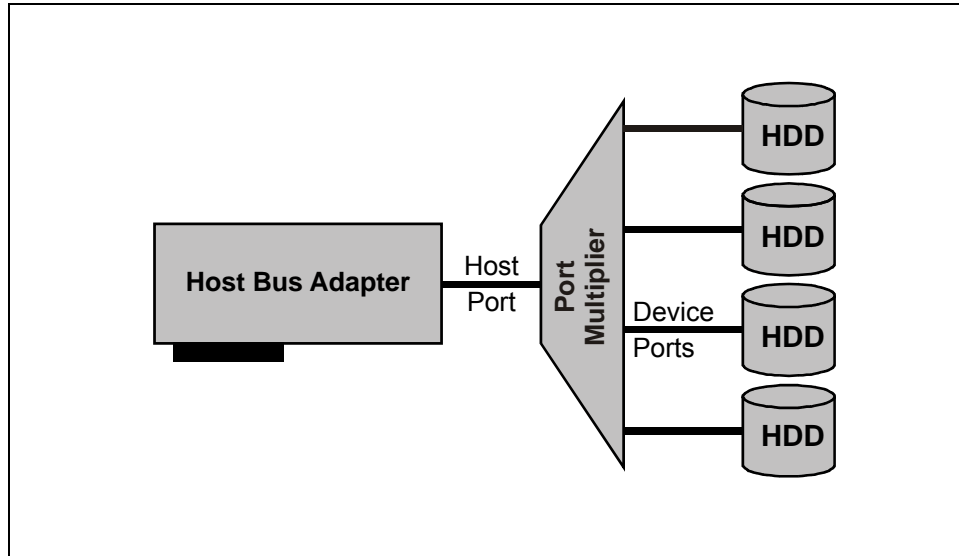


Figure 427 – Port Multiplier overview

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<sub>P</sub> 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<sub>P</sub> to the device. A collision may occur as the Port Multiplier is issuing the X\_RDY<sub>P</sub> to the device if the device has just decided to start a transmission to the host. If the device starts transmitting X\_RDY<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> status to the

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

host until the device has issued an R\_OK<sub>P</sub> for the FIS reception. The R\_OK<sub>P</sub> 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<sub>P</sub> handshake is interlocked. The Port Multiplier shall ensure that the flow control signaling latency requirement as defined in 9.4.9 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<sub>P</sub> 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<sub>P</sub> 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<sub>P</sub> to the host to start the transmission of the FIS to the host shall wait for reception of R\_RDY<sub>P</sub> from the host, shall transmit the FIS from the device to the host, shall wait for reception of R\_OK<sub>P</sub> for the FIS reception from the host, and shall send R\_OK<sub>P</sub> to the device to complete the R\_OK<sub>P</sub> 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<sub>P</sub>/R\_RDY<sub>P</sub> handshake with the host prior to issuing an R\_RDY<sub>P</sub> 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<sub>P</sub> handshake is interlocked. The Port Multiplier shall ensure that the flow control signaling latency requirement as defined in 9.4.9 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<sub>P</sub> or R\_ERR<sub>P</sub> to the initiator of a FIS until the target of the FIS has issued R\_OK<sub>P</sub> or R\_ERR<sub>P</sub> once the end-to-end transmission has commenced.

If the initiator or target of a FIS transmission issues SYNC<sub>P</sub> 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<sub>P</sub> at the same time that the Port Multiplier is issuing X\_RDY<sub>P</sub> to that device. All collisions are treated as an X\_RDY<sub>P</sub>/X\_RDY<sub>P</sub> 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 428).

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 Native Command Queuing 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<sub>P</sub> 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.

<b>PMC1: PmColl_Idle</b>	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
<b>PMC2: PmColl_ChkRecv</b>	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
<b>PMC3: PmColl_Collision</b>	Collision is detected. Finish reception of the FIS from the host and discard the FIS contents.		
	1. WTRM <sub>P</sub> or SYNC <sub>P</sub> not received from host.	→	PmColl_Collision
	2. WTRM <sub>P</sub> received from host.	→	PmColl_FailHostFIS
	3. SYNC <sub>P</sub> received from host.	→	PmColl_Idle
<b>PMC4: PmColl_FailHostFIS</b>	Transmit R_ERR <sub>P</sub> to the host.		
	1. Unconditional	→	PmColl_Idle
<b>PMC5: PmColl_XmitToDP</b>	Transmit X_RDY <sub>P</sub> to the device port X.		
	1. X_RDY <sub>P</sub> or R_RDY <sub>P</sub> not received from device port X.	→	PmColl_XmitToDP
	2. X_RDY <sub>P</sub> received from device port X.	→	PmColl_Collision
	3. R_RDY <sub>P</sub> received from device port X.	→	PmColl_Idle

**Figure 428 – Port Multiplier collisions state machine**

X\_RDY<sub>P</sub>/X\_RDY<sub>P</sub> collisions that occur on the host port shall be handled in accordance with the Link layer state machines as defined in 9.6.

### 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 429) 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.

<b>HPHP1: NoComm</b> <sup>a</sup>	Set Port Multiplier to initial state as defined in 13.16.1 and 13.16.2.2.	
1. Unconditional	→	StartComm
<sup>a</sup> This state is entered upon power-on reset or in response to a received COMRESET.		
<b>HPHP2: StartComm</b>	Transition Phy state machine to state DP1: DR_RESET. Communications retry interval reset to initial value. <sup>a</sup>	
1. Unconditional	→	WaitComm
<sup>a</sup> The communications retry interval shall be greater than or equal to 10 ms.		
<b>HPHP3: WaitComm</b>		
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
<b>HPHP4: EnablePort0</b>	Enable device Port 0 if device Port 0 is currently disabled.	
1. PHYRDY asserted.	→	Port0ComInit
2. PHYRDY not asserted.	→	NoComm
<b>HPHP4b: Port0ComInit</b>		
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 <sup>a</sup>
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
<sup>a</sup> 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 429 – 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 <sup>a</sup>
3. Communications established and FIS not received for control port and COMINIT negated from device Port 0.	→	LegacyCommOK <sup>b</sup>
4. Communications established and FIS not received for control port and COMINIT asserted from device Port 0.	→	Port0ComInitWait
<sup>a</sup> 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. <sup>b</sup> 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 <sup>a</sup>
4. FIS received for control port.	→	CommOK <sup>b</sup>
<sup>a</sup> The X bit in PSCR[1] (SError) for Port 0 shall be set to one prior to making the transition. <sup>b</sup> 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 429 – 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 430) 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.

<b>DPHP1: NoComm</b>			
1. PHYRDY asserted.	→	CommOK	
2. PHYRDY not asserted.	→	NoComm	
<b>DPHP2: CommOK</b>			
1. Communications lost and interface not in power management state.	→	NoComm	
2. Communications established or interface in power management state.	→	CommOK	

**Figure 430 – Device port hot plug state machine**

### 16.3.3.6 Link power management

The Port Multiplier shall support reception of PMREQ\_P<sub>P</sub> and PMREQ\_S<sub>P</sub> 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<sub>P</sub> or PMREQ\_S<sub>P</sub> with PMNAK<sub>P</sub>.

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<sub>P</sub> or PMREQ\_S<sub>P</sub> received on a device port. PMREQ\_P<sub>P</sub> or PMREQ\_S<sub>P</sub> received from a device only affects the link between that device and the Port Multiplier.

If the Port Multiplier receives PMREQ\_P<sub>P</sub> or PMREQ\_S<sub>P</sub> over a device port, the Port Multiplier shall perform the following actions:

- a) the Port Multiplier shall respond to the device with PMACK<sub>P</sub> or PMNAK<sub>P</sub>; and
- b) if the Port Multiplier responds to the device with PMACK<sub>P</sub>:
  - 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<sub>P</sub> or PMREQ\_S<sub>P</sub> received from the host to all active device ports if the Port Multiplier responds to PMREQ\_P<sub>P</sub> or PMREQ\_S<sub>P</sub> with PMACK<sub>P</sub>. In this case, the Port Multiplier shall propagate PMREQ\_P<sub>P</sub> or PMREQ\_S<sub>P</sub> to all device ports that have PHYRDY asserted. If a device responds to PMREQ\_P<sub>P</sub> or PMREQ\_S<sub>P</sub> with PMNAK<sub>P</sub>, 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<sub>P</sub> or PMREQ\_S<sub>P</sub> from the host, the specific actions the Port Multiplier shall take are:

- a) the Port Multiplier shall respond to the host with PMACK<sub>P</sub> or PMNAK<sub>P</sub>; and
- b) if the Port Multiplier responds to the host with PMACK<sub>P</sub>:
  - 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<sub>P</sub> or PMREQ\_S<sub>P</sub> to all device ports that have PHYRDY asserted:

## HIGH SPEED SERIALIZED AT ATTACHMENT

### Serial ATA International Organization

- 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 9.6.3 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<sub>P</sub> or R\_ERR<sub>P</sub> 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 431 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 431 – Register values for an unsupported command

Register Values for an Unsupported Command:

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 0.

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 (see 16.7.2). 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 393. 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). RegNum(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). RegNum(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 and DRQ bits 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.

## HIGH SPEED SERIALIZED AT ATTACHMENT

### Serial ATA International Organization

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 127).

**Table 127 – Static information registers**

Field	O/M	F/V	Description
GSCR[0]	M	F F	Product Identifier 31..16 Device ID allocated by the vendor. 15..0 Vendor ID allocated by the PCI-SIG.
GSCR[1]	M	F F F F F F F	Revision Information 31..16 Reserved 15..8 Revision level of the Port Multiplier. 7..4 Reserved 3 Supports Port Multiplier specification 1.2. 2 Supports Port Multiplier specification 1.1. 1 Supports Port Multiplier specification 1.0. 0 Reserved
GSCR[2]	M	F F	Port Information 31..4 Reserved 3..0 Number of exposed device fan-out ports.
GSCR[3] to GSCR[31]	O	F	Reserved
<p>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.</p>			

**Register 0, Product Identifier**

The register identifies the vendor that produced the Port Multiplier and the specific device identifier.

Bit 15..0 shall be set to the vendor identifier allocated by the PCI-SIG of the vendor that produced the Port Multiplier.

Bit 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.

Bit 7..4 are reserved.

Bit 15..8 identifies the revision level of the Port Multiplier product identified by Register 0. This identifier is allocated by the vendor.

Bit 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.

Bit 3..0 specifies the number of exposed device fan-out ports. A value of zero is invalid. The control port shall not be counted.

Bit 31..4 are reserved.

Register 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 128) contains registers that convey status information and control operation of the Port Multiplier.

**Table 128 – Status information and control registers**

Field	O/M	F/V	Description	
GSCR[32]	M	F	Error Information	
			31..15	Reserved
			14	OR of selectable bits in Port 14 PSCR[1] (SError)
			13	OR of selectable bits in Port 13 PSCR[1] (SError)
			12	OR of selectable bits in Port 12 PSCR[1] (SError)
			11	OR of selectable bits in Port 11 PSCR[1] (SError)
			10	OR of selectable bits in Port 10 PSCR[1] (SError)
			9	OR of selectable bits in Port 9 PSCR[1] (SError)
			8	OR of selectable bits in Port 8 PSCR[1] (SError)
			7	OR of selectable bits in Port 7 PSCR[1] (SError)
			6	OR of selectable bits in Port 6 PSCR[1] (SError)
			5	OR of selectable bits in Port 5 PSCR[1] (SError)
			4	OR of selectable bits in Port 4 PSCR[1] (SError)
			3	OR of selectable bits in Port 3 PSCR[1] (SError)
			2	OR of selectable bits in Port 2 PSCR[1] (SError)
1	OR of selectable bits in Port 1 PSCR[1] (SError)			
0	OR of selectable bits in Port 0 PSCR[1] (SError)			
GSCR[33]	M	V	Error Information Bit Enable	
			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	
<p>Key:</p> <p>O/M = Mandatory/optional requirement.</p> <p>M = Support of the register is mandatory.</p> <p>O = Support of the register is optional.</p> <p>F/V = Fixed/variable content.</p> <p>F = the content of the register is fixed and does not change.</p> <p>V = the contents of the register is variable and may change.</p>				

**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.

Bit 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 1..15 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.



## HIGH SPEED SERIALIZED AT ATTACHMENT

### Serial ATA International Organization

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.

Register 35..63, Reserved

Registers 35..63 are reserved for future Port Multiplier definition.

#### 16.4.2.4 Features Supported

The Features Supported section of the General Status and Control registers (see Table 129) contains registers that convey the optional features that are supported by the Port Multiplier. All Features Supported registers are read-only.

**Table 129 – Features Supported registers**

Field	O/M	F/V	Description
GSCR[64]	M	F	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 <sub>P</sub> 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<sub>P</sub> and PMREQ<sub>S</sub> 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<sub>P</sub> or PMREQ<sub>S</sub> 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 Notification 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.

Bit 31..5 are reserved.

**Register 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 130) 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 130 – Features Enabled registers**

Field	O/M	F/V	Description
GSCR[96]	M	F V V V V	Port Multiplier Revision 1.X Features Enable 31..4 Reserved 3 Asynchronous notification enabled 2 Dynamic SSC transmit is enabled 1 Issuing PMREQ <sub>P</sub> to host is enabled 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<sub>P</sub> and PMREQ<sub>S</sub> 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<sub>P</sub> or PMREQ<sub>S</sub> 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 Notification 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.

Bit 31..4 are reserved.

Register 97..127: 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 131) contains registers that are vendor specific.

**Table 131 – 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.			

Register 128..255: Vendor specific

#### 16.4.2.7 Phy event counters

The Phy event counters section of the General Status and Control registers (see Table 132) 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 132 – 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.			

Register 256..2 303, 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 133).

**Table 133 – 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 134.

**Table 134 – 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 432) 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			
Command(7:0)	E4h							

**Figure 432 – READ PORT MULTIPLIER command definition**

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

- PortNum Set to the port address that has register to be read.
- RegNum Set to number of register to read.
- RS1 Register Specific 1, 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, 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 RegNum.  
All other usage, this bit shall be treated as “na” on all READ PORT MULTIPLIER commands to registers other than Phy event counter registers.

**16.5.1.3 Success outputs**

Upon successful completion, the Port Multiplier registers shall be returned as described in Figure 433.

Field	7	6	5	4	3	2	1	0
Error(7:0)	0							
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 433 – READ PORT MULTIPLIER success status result values**

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 434.

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 434 – READ PORT MULTIPLIER error status result values**

- 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 435) 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			
Command(7:0)	E8h							

**Figure 435 – WRITE PORT MULTIPLIER command definition**

- Value Set to the 64 bit value to write to the register.
- RegNum Set to number of register to write.
- PortNum Set to the port address that has register to be written.
- RS1 Register Specific 1, this bit is register specific.

Phy event counter usage, 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 RegNum 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 and DRQ bits 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, 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 436).

Field	7	6	5	4	3	2	1	0
Error(7:0)	0							
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 436 – WRITE PORT MULTIPLIER success status result values**

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 437.

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 437 – WRITE PORT MULTIPLIER error status result values**

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 438.

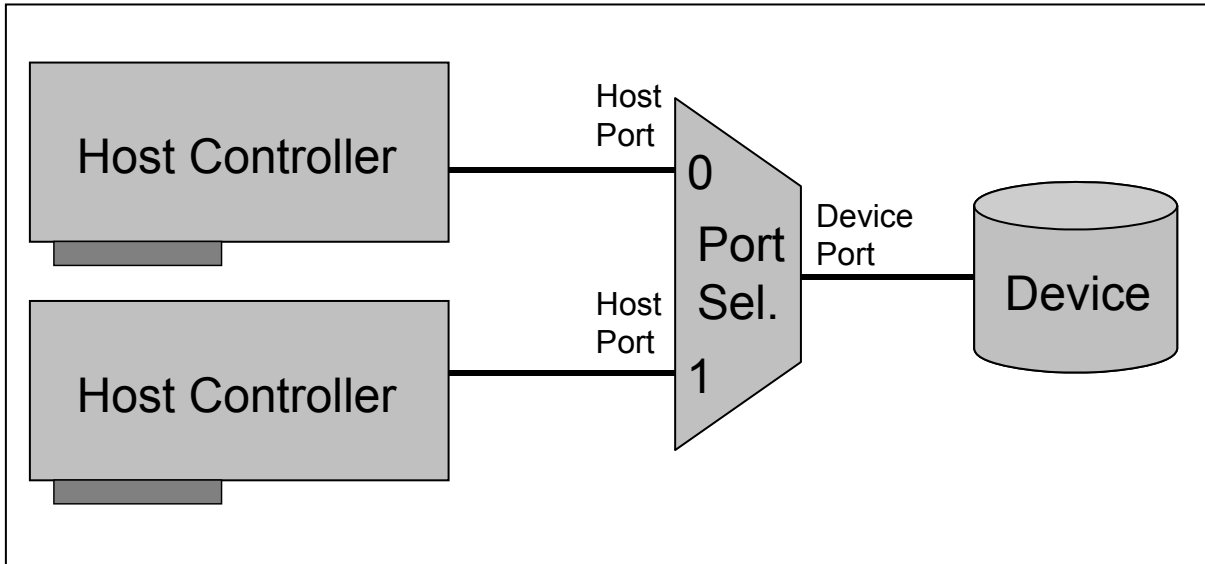
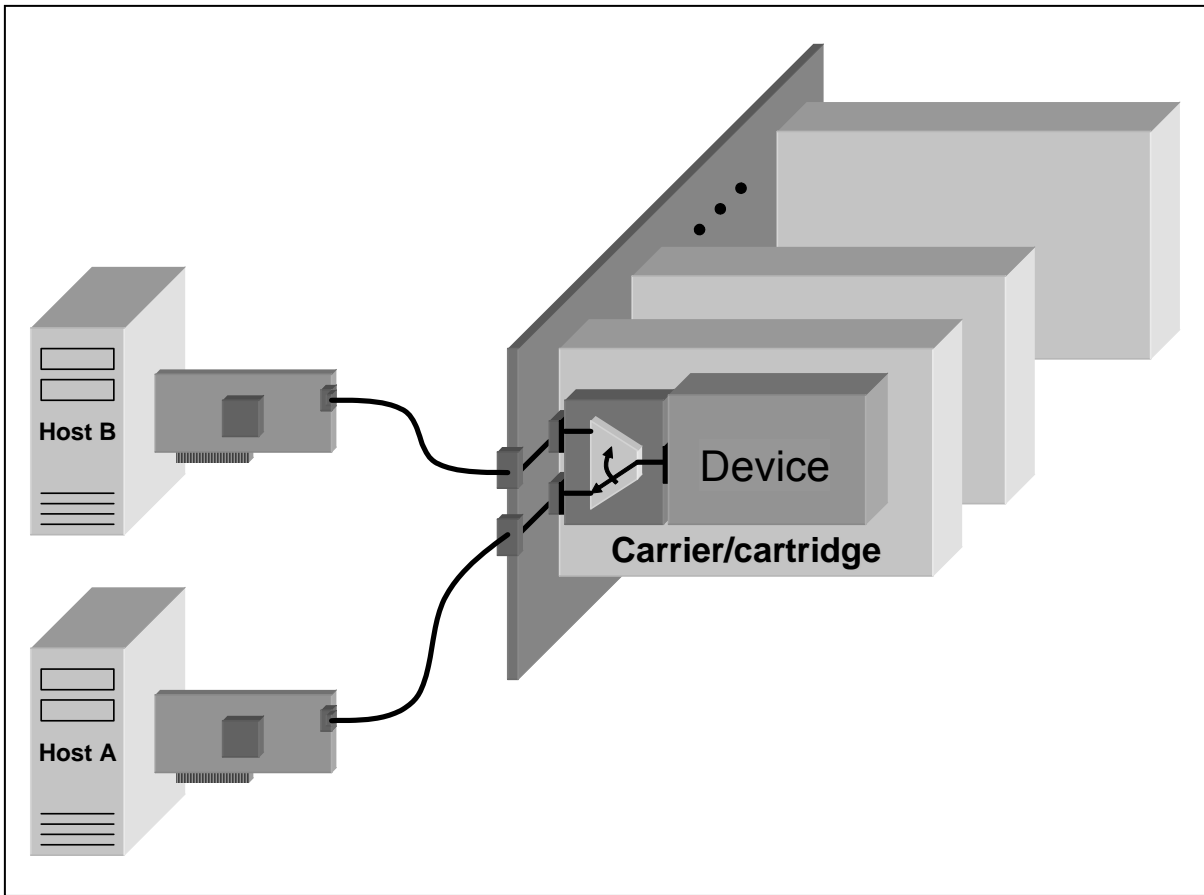


Figure 438 – Port Selector overview

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 439, 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 439 – 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**

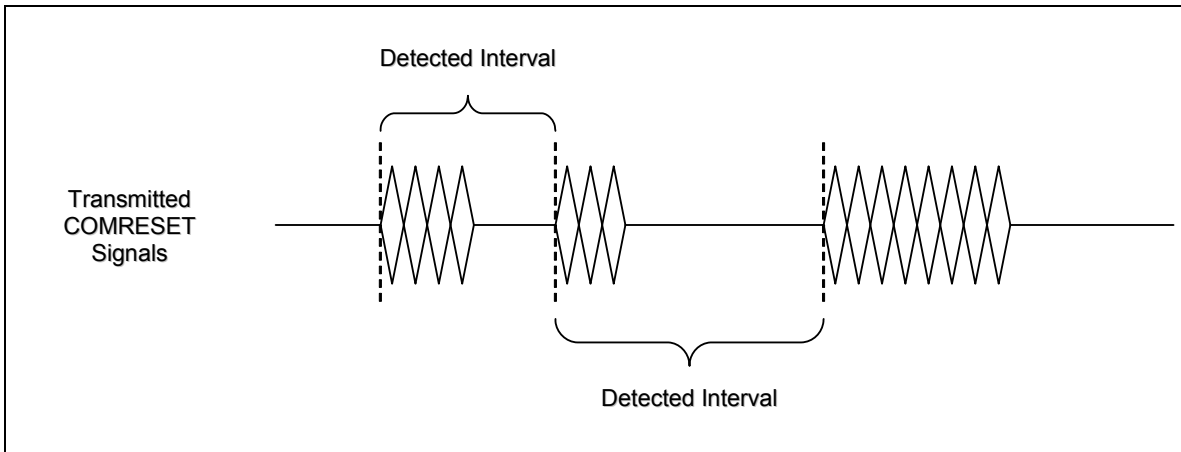
#### **17.4.2.1 Protocol-based Port Selection overview**

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 440, 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 440 – Port selection signal based on assertion of COMRESET to assertion of following COMRESET**

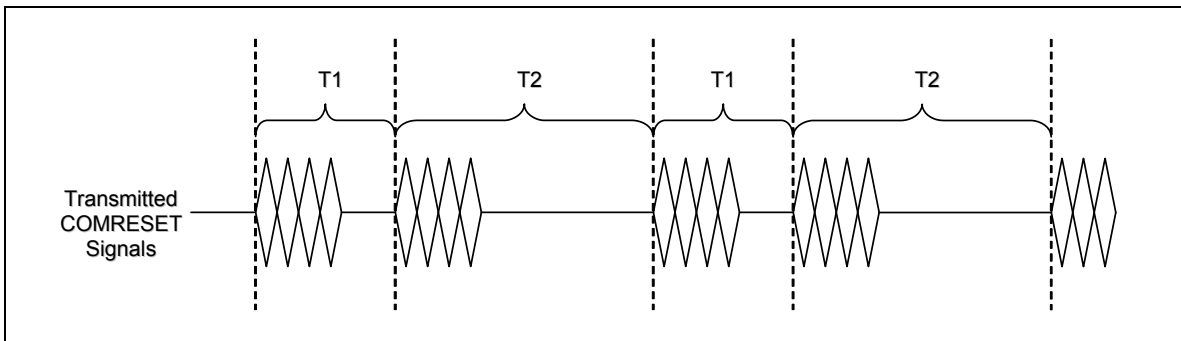
The port selection signal is defined as a series of COMRESET signals with the timing from the assertion of one COMRESET signal to the assertion of the next as defined in Table 135 and illustrated in Figure 441. The Port Selector shall select the port, if inactive, on the negation of COMRESET after receiving 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 135 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 135 – 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 441 – 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 442). A Port Selector shall remain in the DR\_PS\_Wait state if the Phy is offline and presence detection is enabled.

DP1: DR_Reset <sup>a</sup>	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	
<sup>a</sup> 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 442 – 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-pollled 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)**

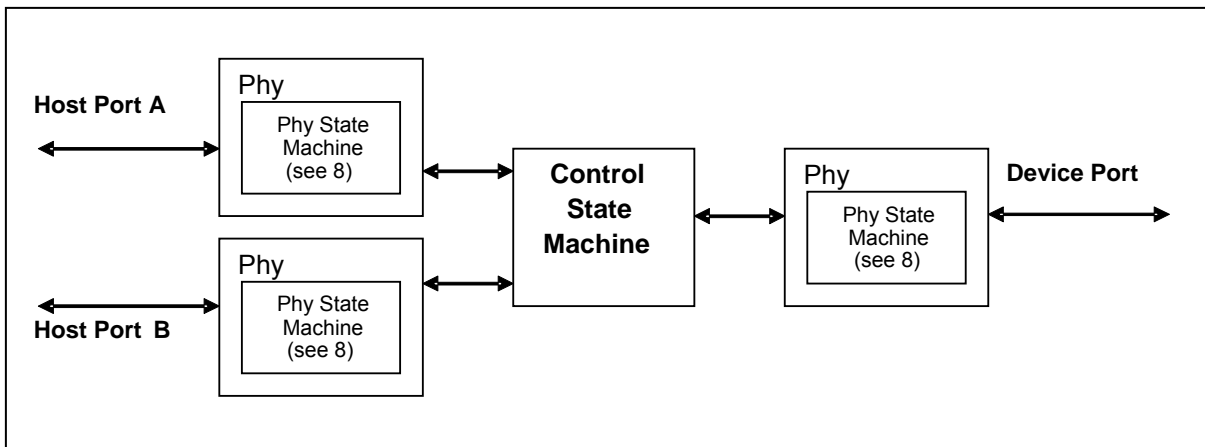
A Port Selector may support an orderly switch to a new active host port. A Port Selector that supports an orderly switch ensures that primitive alignment with the device Phy is maintained during the switch to the new active host port. Maintaining primitive alignment ensures that PHYRDY remains present 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 443. 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.

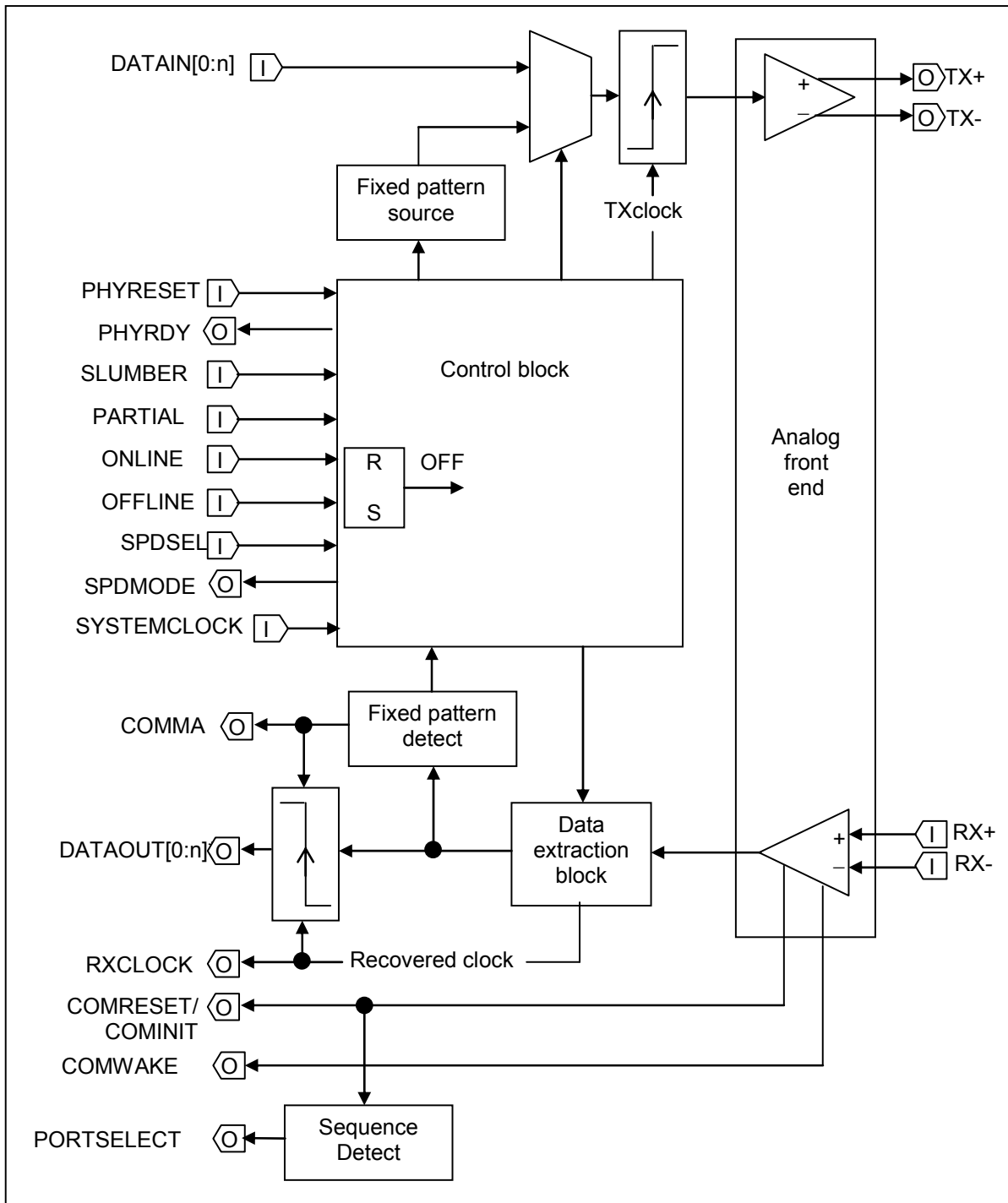


**Figure 443 – Control state machine**

The Phys depicted in Figure 443 are presumed to have the basic capabilities and controls indicated in Figure 444. Figure 444 is a variant of Figure 165.

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 444 – Phy block diagram**

The following state machines specifies the behavior of the Port Selector control logic block (see Figure 445). 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 <sup>a</sup>	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
<sup>a</sup> 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 445 – Port Selector state machine**

(part 1 of 3)

HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization

PS7: WaitForComm			
1. SelHostPort.PHYRDY and D.PHYRDY and no change in PORTSELECT signal	→	CheckSpeeds	
2. (!SelHostPort.PHYRDY or !D.PHYRDY) and timeout <sup>a</sup> not exceeded and no change in PORTSELECT signal	→	WaitForComm	
3. (!SelHostPort.PHYRDY or !D.PHYRDY) and timeout <sup>a</sup> exceeded and no change in PORTSELECT signal	→	PowerManageCheck	
4. PORTSELECT signal received for non-selected host port	→	ChangePort	
<sup>a</sup> The timeout is vendor specific but shall be larger than 1 760 us			
PS8: CheckSpeeds <sup>a</sup>			
1. SelHostPort.SPDMODE = D.SPDMODE	→	Online	
2. SelHostPort.SPDMODE > D.SPDMODE	→	SetDeviceSpeed	
3. SelHostPort.SPDMODE < D.SPDMODE	→	SetHostSpeed	
<sup>a</sup> 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 <sup>a</sup>		Assert SelHostPort.ONLINE. Assert D.ONLINE. Assert SelHostPort.PHYRESET	
1. Unconditional	→	WaitForComm	
<sup>a</sup> This state is entered unconditionally upon receipt of D.COMINIT if SelHostPort != none and the Port Selector is not in state ResetDevice.			

Figure 445 – Port Selector state machine (part 2 of 3)

PS15: ResetDevice <sup>a</sup>	Assert SelHostPort.ONLINE. Assert D.ONLINE. Assert D.PHYRESET.	
1. Unconditional	→	WaitForComm
<sup>a</sup> 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 445 – 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 9.4.9. 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 to 4h;
- 2) set Phy to reset state by writing SControl.DET 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 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 to 4h;
- 7) wait 1.975 ms to satisfy T1 timing as specified in Table 135;
- 8) repeat steps 2 to 6;
- 9) wait 7.975 ms to satisfy T2 timing as specified in Table 135;
- 10) repeat steps 2 to 6;
- 11) wait 1.975 ms to satisfy T1 timing as specified in Table 135;
- 12) repeat steps 2 to 6;
- 13) wait 7.975 ms to satisfy T2 timing as specified in Table 135;
- 14) set Phy to reset state by writing SControl.DET 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 to 0h;
- 17) wait up to 10 ms for SStatus.DET = 3h; and
- 18) if SStatus.DET != 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<sub>P</sub>. 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**



# HIGH SPEED SERIALIZED AT ATTACHMENT

## Serial ATA International Organization

```

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];
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





```

}

/* 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[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[14] ^ now[10] ^ now[9] ^ now[8] ^ 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(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 <sub>P</sub> , primitives not scrambled, scrambler reset
0030 8027h	C2D2 768Dh	C2E2 F6AAh	11E3 53FDh	Register Host to Device FIS, Command = 30, Cbit 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 <sub>P</sub> , primitives not scrambled

NOTE 89 - All values in hexadecimal, shown 31:0.

The transmitted Dwords for this Register Host to Device FIS, prior to 8b/10b encoding, are:

SOF<sub>P</sub>  
 C2E2 F6AAh  
 FE05 F60Fh  
 A508 436Ch  
 3452 D356h  
 8A55 9502h  
 8A85 4174h  
 EOF<sub>P</sub>

## **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 DRQ bit getting set to one and 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 DRQ 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 DRQ being cleared to zero and the ending status reported; and
- 8) if there are more data blocks to be transferred, the ending status indicates 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 DRQ bit getting set to one and 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 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 DRQ being cleared to zero, and closes the frame with a CRC and EOF<sub>p</sub>. 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 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 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., 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;
- 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., 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);
- 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 BSY bit getting cleared to zero and DRQ bit getting set to one;
- 5) as a result of BSY bit being cleared to zero and 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 DRQ bit being cleared to zero and BSY bit being set to one, and closes the frame with a CRC and EOF<sub>P</sub>. 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 DRQ bit getting set to one and 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 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 DRQ bit being cleared to zero and BSY bit being set to one;
- 13) the device transmits final ending status by sending a Register Device to Host FIS with 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 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 BSY bit being cleared to zero and DRQ bit being set to one;
- 5) as a result of BSY bit being cleared to zero and 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 DRQ bit being cleared to zero and BSY bit being set to one, and closes the frame with a CRC and EOF<sub>P</sub>. 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 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 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 DRQ bit being cleared to zero, BSY bit being set to one, and closes the frame with a CRC and EOF<sub>P</sub>;
- 13) the device transmits final ending status by sending a Register Device to Host FIS with 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 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 BSY bit getting cleared to zero and DRQ bit getting set to one;
- 6) as a result of BSY bit getting cleared to zero and 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 DRQ bit being cleared to zero and BSY bit being set to one, and closes the frame with a CRC and EOF<sub>P</sub>. 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 BSY bit getting cleared to zero and DRQ bit getting set to one;
- 6) as a result of BSY bit getting cleared to zero and 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 DRQ bit being cleared to zero and BSY bit being set to one, and closes the frame with a CRC and EOF<sub>P</sub>. 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 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 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 Native Command Queuing examples**

### **B.15.1 Native Command Queuing examples overview**

The following is an overview of macro operations and their sequencing for two typical Native Command Queuing 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 BSY bit to zero by transmitting a Register Device to Host FIS.
Host issues Read Command Tag=5 if 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 BSY bit to zero.
	Device sends DMA Setup FIS, DMA buffer ID=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 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 ID=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 BSY bit to zero by transmitting a Register Device to Host FIS.
Host issues Read Command Tag=5 (if 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 BSY bit to zero.
	Device sends DMA Setup FIS, DMA buffer ID=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 001 00 000b), 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 ID=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 000 0 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). 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. This function is typically used in devices that support Device 0 and Device 1 operation (see 13.1.4), and it is also required for Overlapped operation.

In this specification, the implementation of Device 0 / Device 1 emulation is performed exclusively by the host (see 13.1.4).

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:

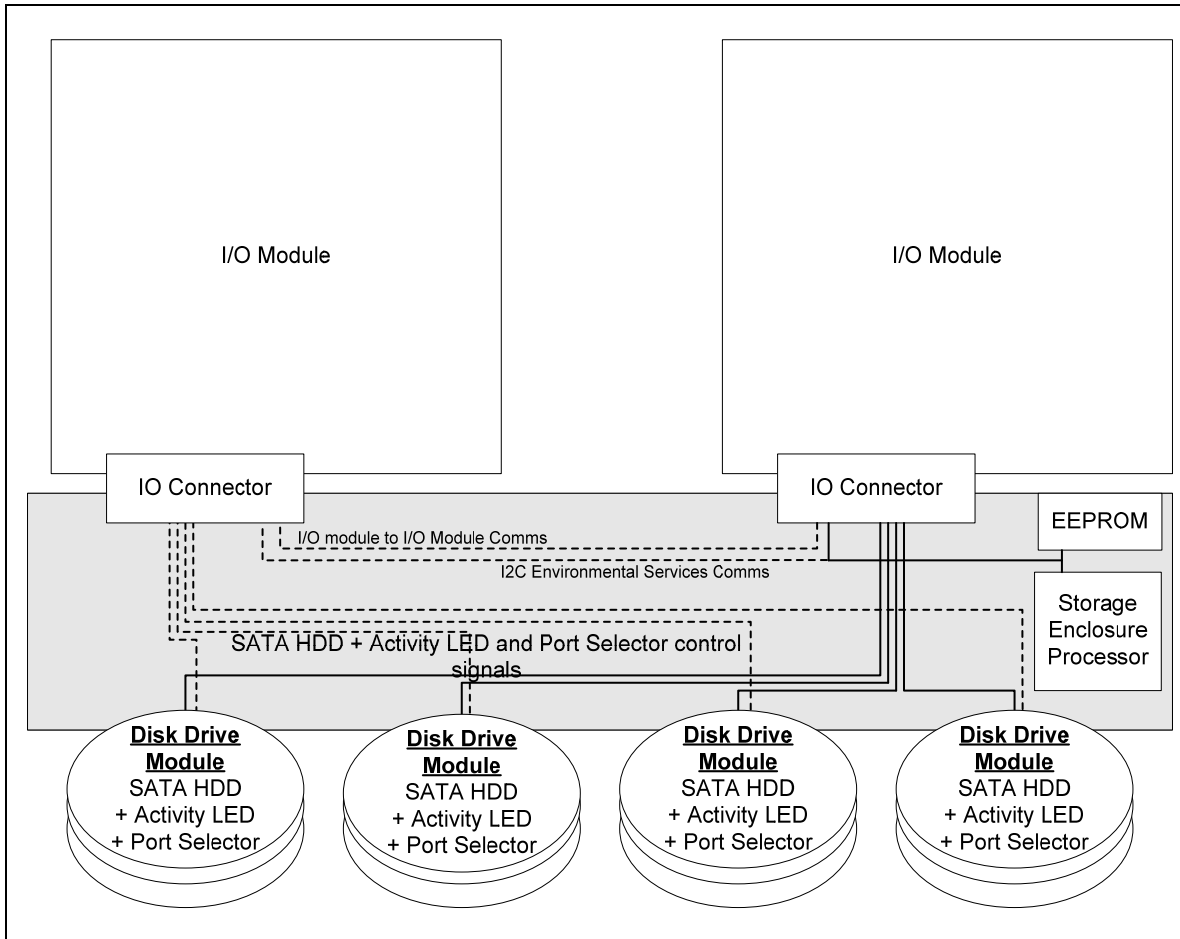
- 1) the Interrupt bit is the interrupt pending flag for Serial ATA devices;
- 2) the behavior of the Interrupt bit may be modified by the state of the nIEN bit in devices implemented to prior versions of this specification. Such devices may change the behavior of the Interrupt bit based on the current 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**

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:

- power connection between the module and the backplane;
- data connection between the module and the backplane;
- HDD activity LED (see 6.13.1.1), optional;
- 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:

- Single I/O Controller System; or
- 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:

- Serial ATA interface;
- I/O controller to enclosure processor (SEP) communication;
- Disk activity LED signals;
- I/O controller power and ground; and
- I/O controller identification and control (RESET).

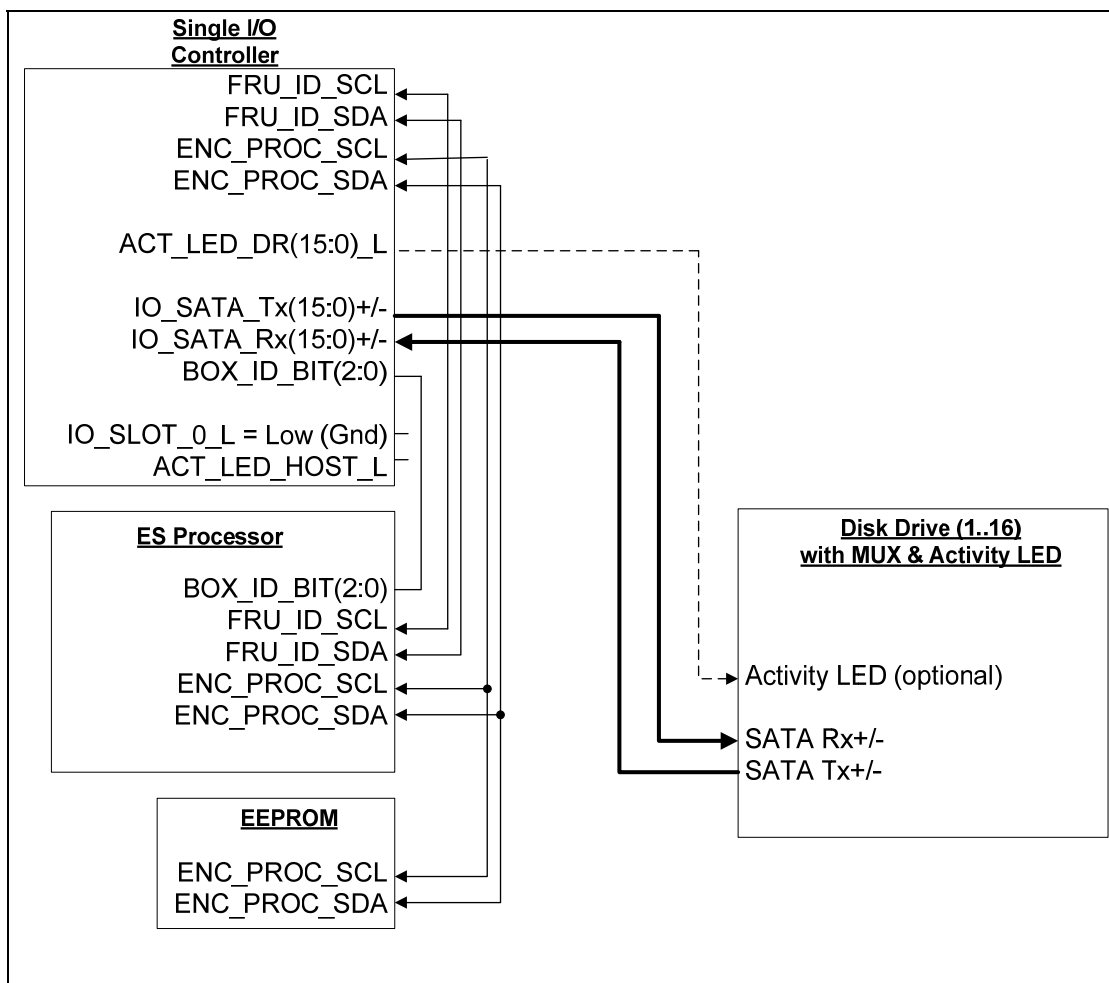


Figure D.2 – An example of signal connections with one I/O controller

### **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**

Further 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.

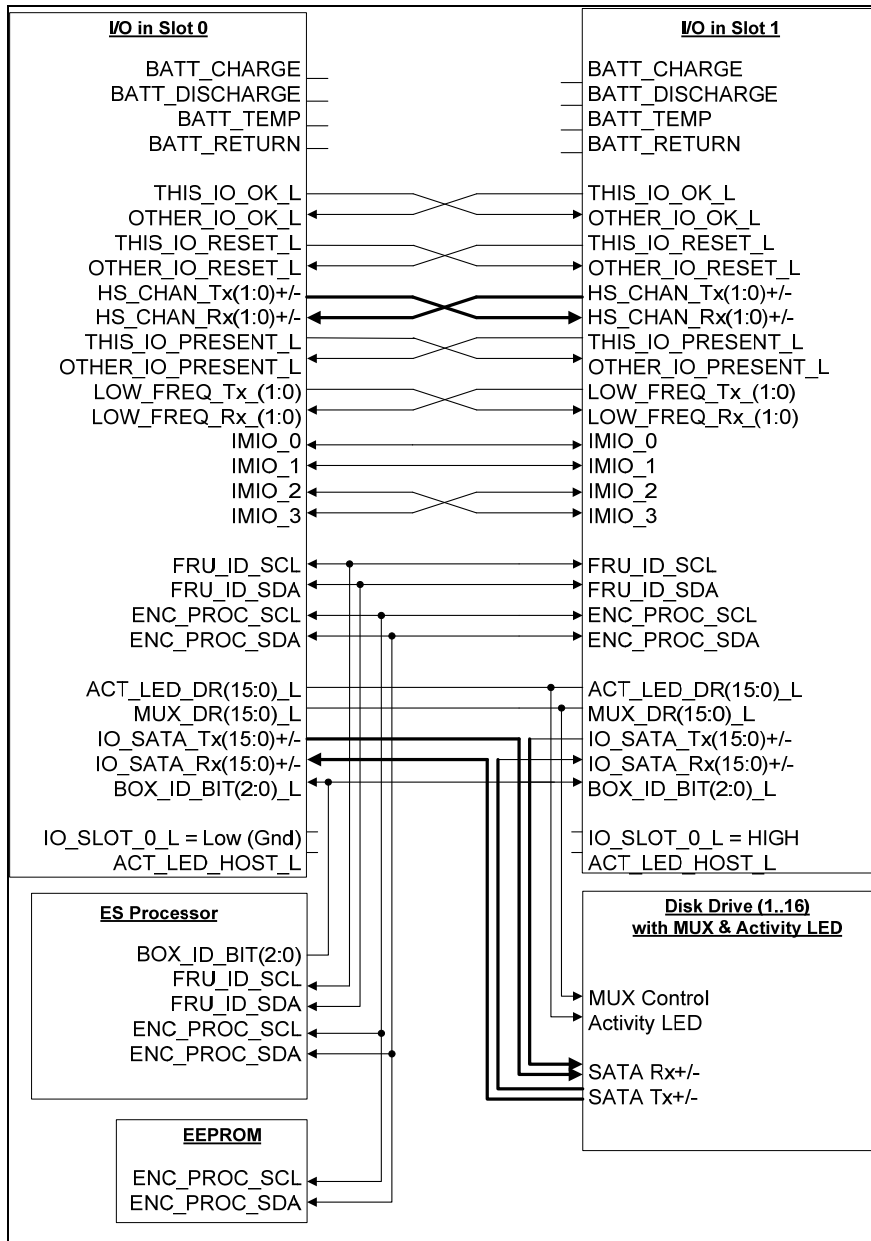


Figure D.3 – Example of signal connections with two I/O modules

### D.3 Optional high speed channel configurations

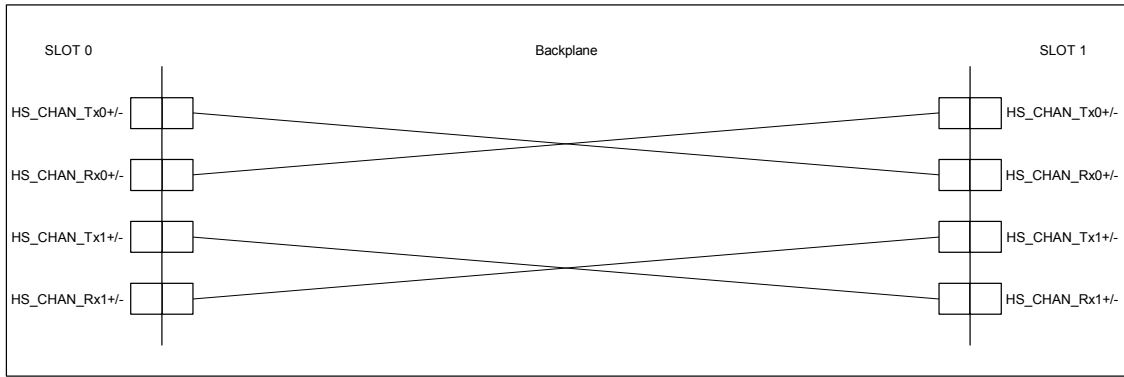
The optional High speed communications channels may be used for high speed differential communications between the two I/O controllers, or for I/O controller to host communications (e.g., inclusion into a Fibre Channel loop). It is recommended that these signals should be 100 ohm differential characteristic impedance.

Actual usage is open to the user definition however the normative backplane routing should be either:

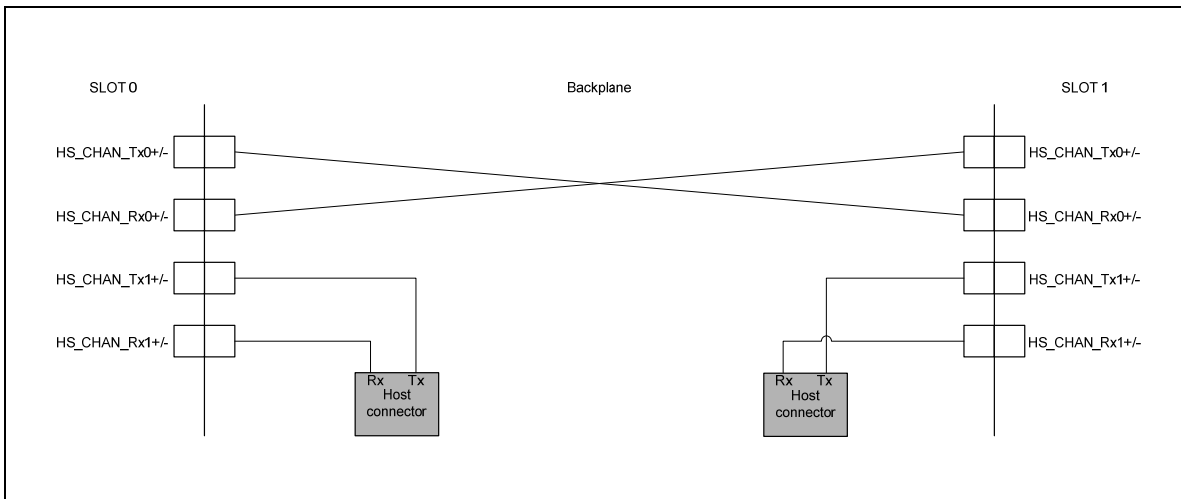
- Configuration 0, Both channels link I/O controller to I/O controller; and
- Configuration 1, One channel links the two I/O Controllers, and the other is router to a host connector.



HIGH SPEED SERIALIZED AT ATTACHMENT  
Serial ATA International Organization



**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.

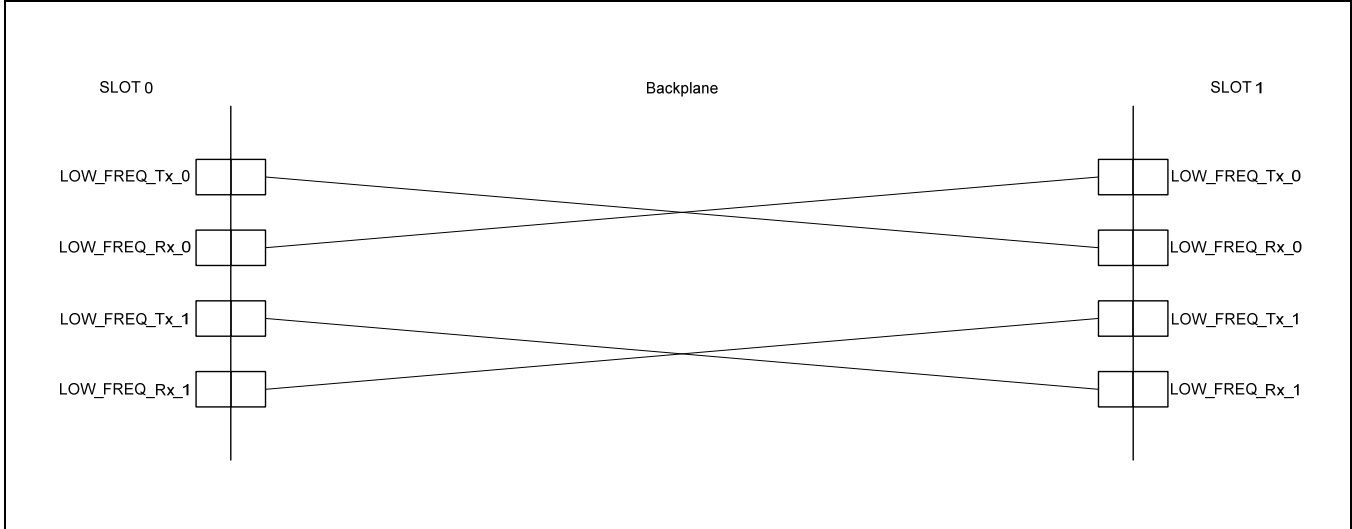


Figure D.6 – Low speed channels

There are 4 end-to-end signals that may be used for intermodule I/O communications.

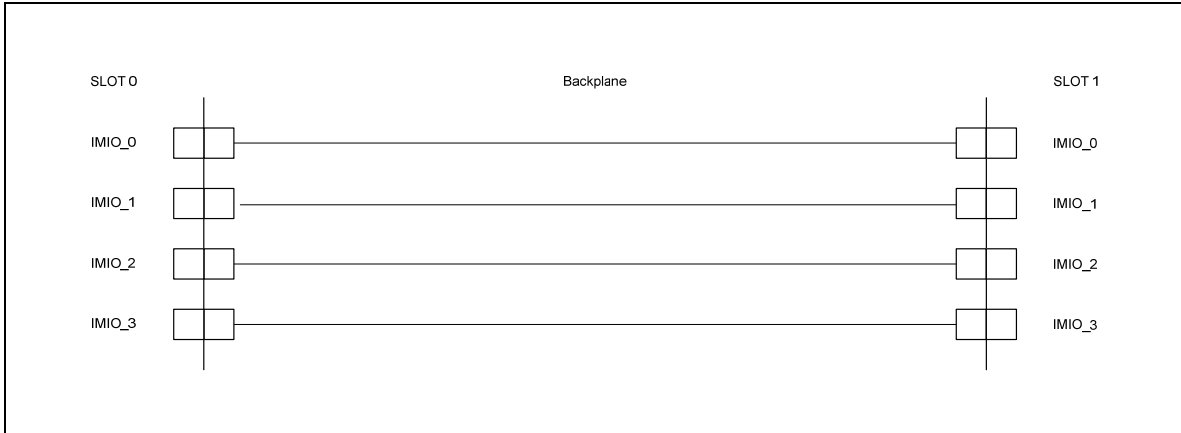


Figure D.7 – Interconnect channels

## 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 Metal 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 cConnector

The interface connector that shall be used on the I/O controller is based on a Berg-FCI High Speed Metal 5x6 30-pin right angle female receptacle part number 52057-102. This connector is used for the 6 high-speed connectors.

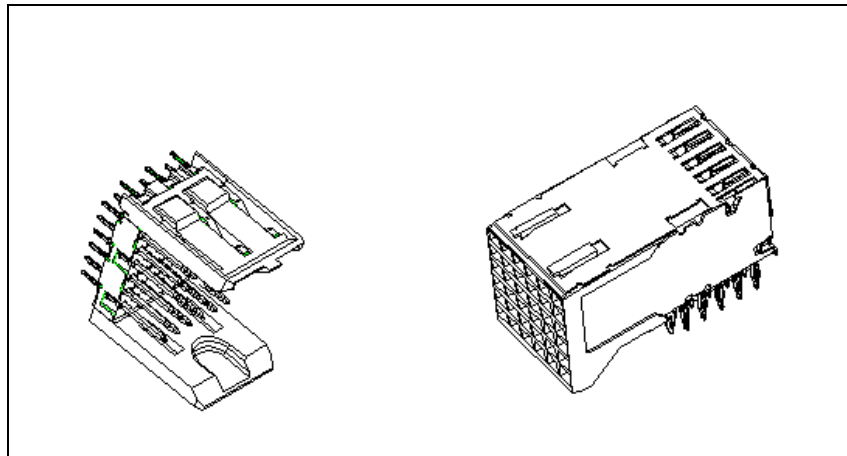


Figure D.8 – I/O controller module connector rendering

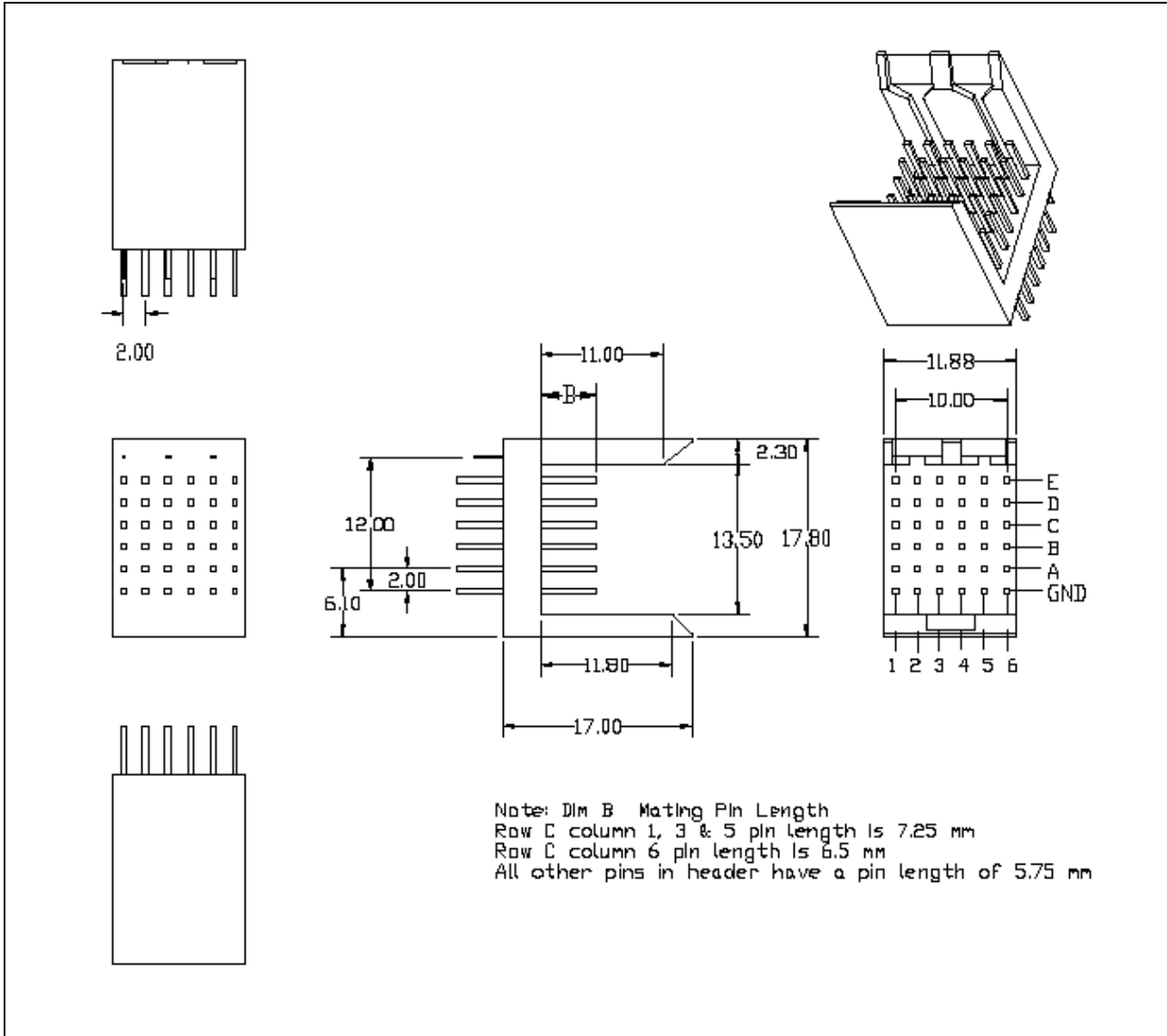


Figure D.9 – Connector pin layout and pin lengths

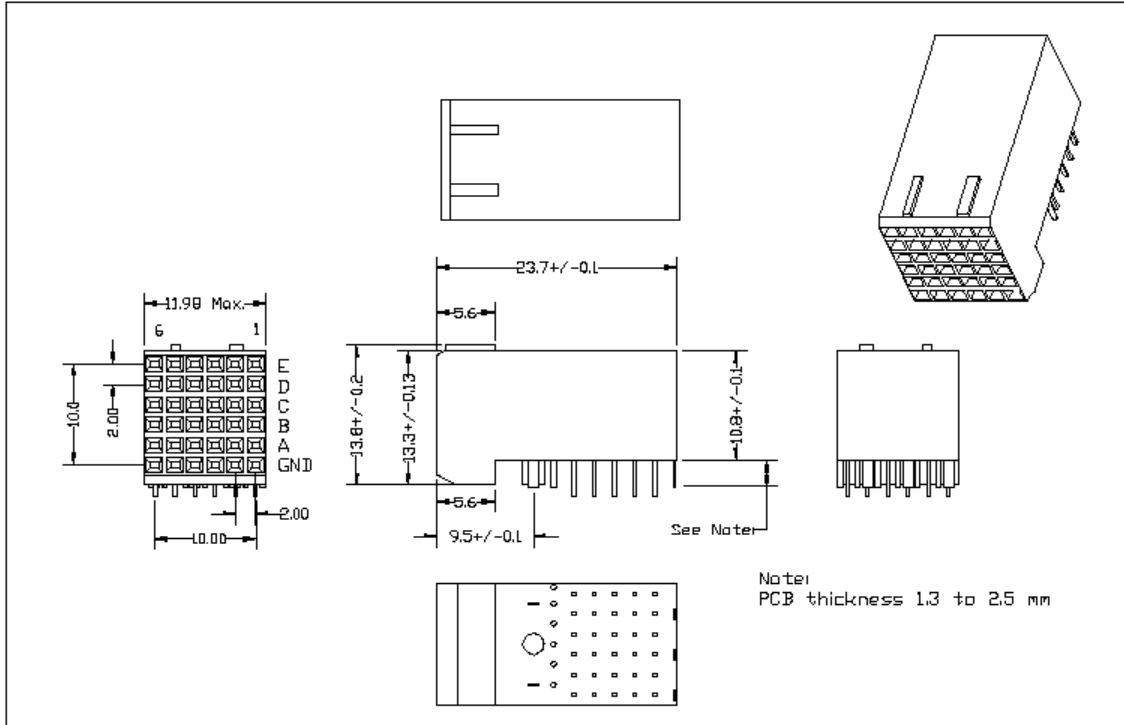


Figure D.10 – I/O controller module connector receptacle engineering drawing

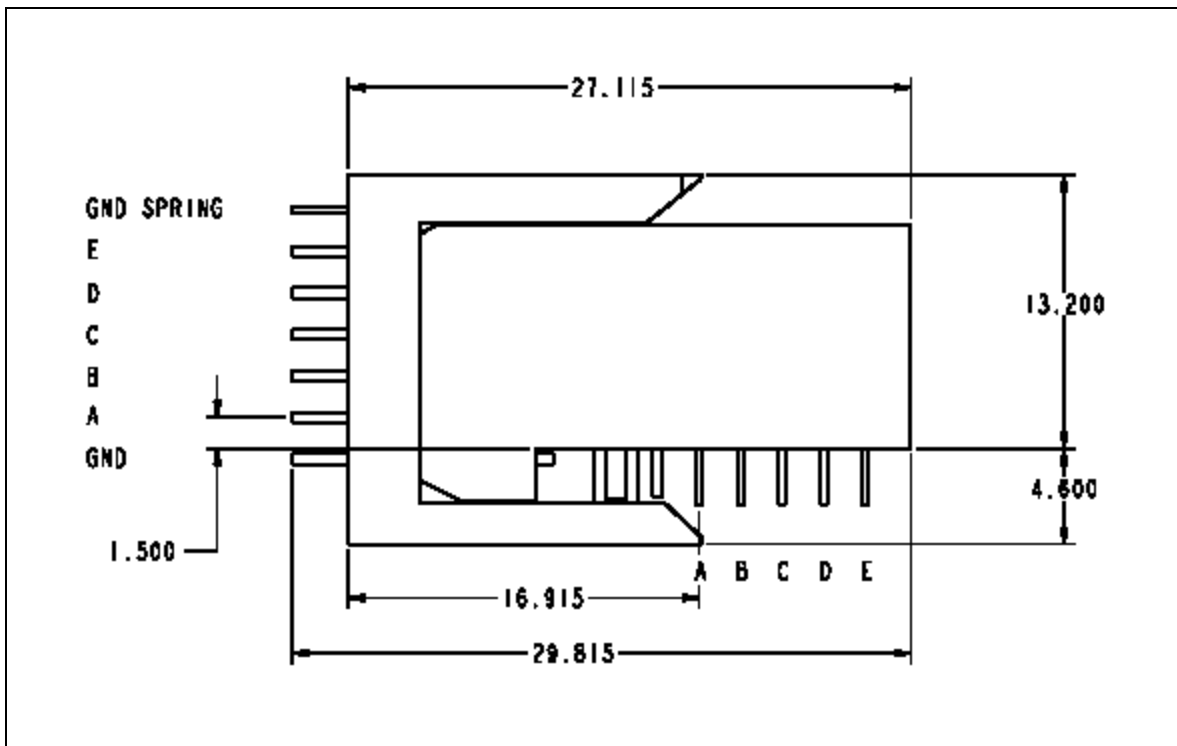


Figure D.11 – Side view of connector

## **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.

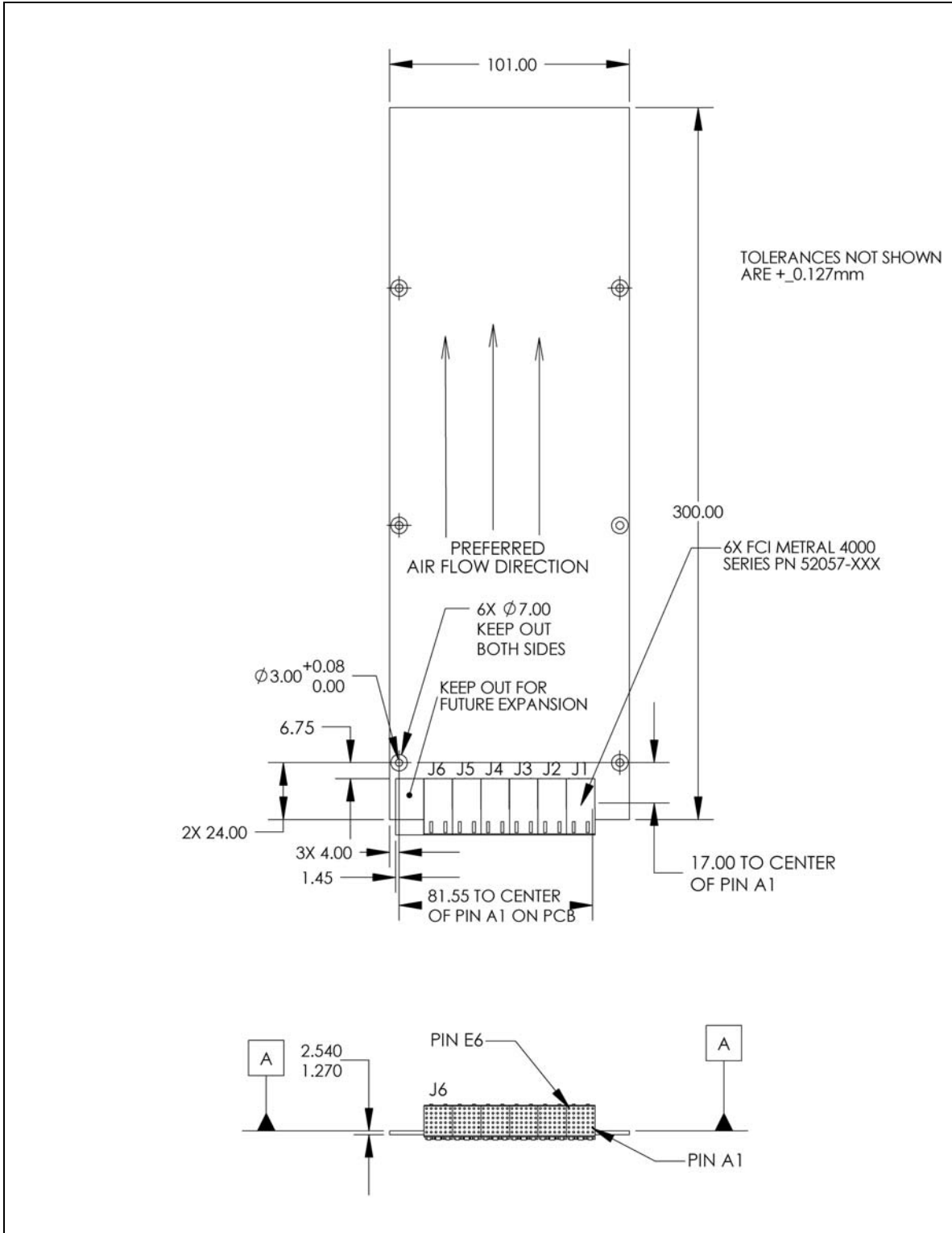


Figure D.12 – I/O controller module connector locations on 1xWide I/O module

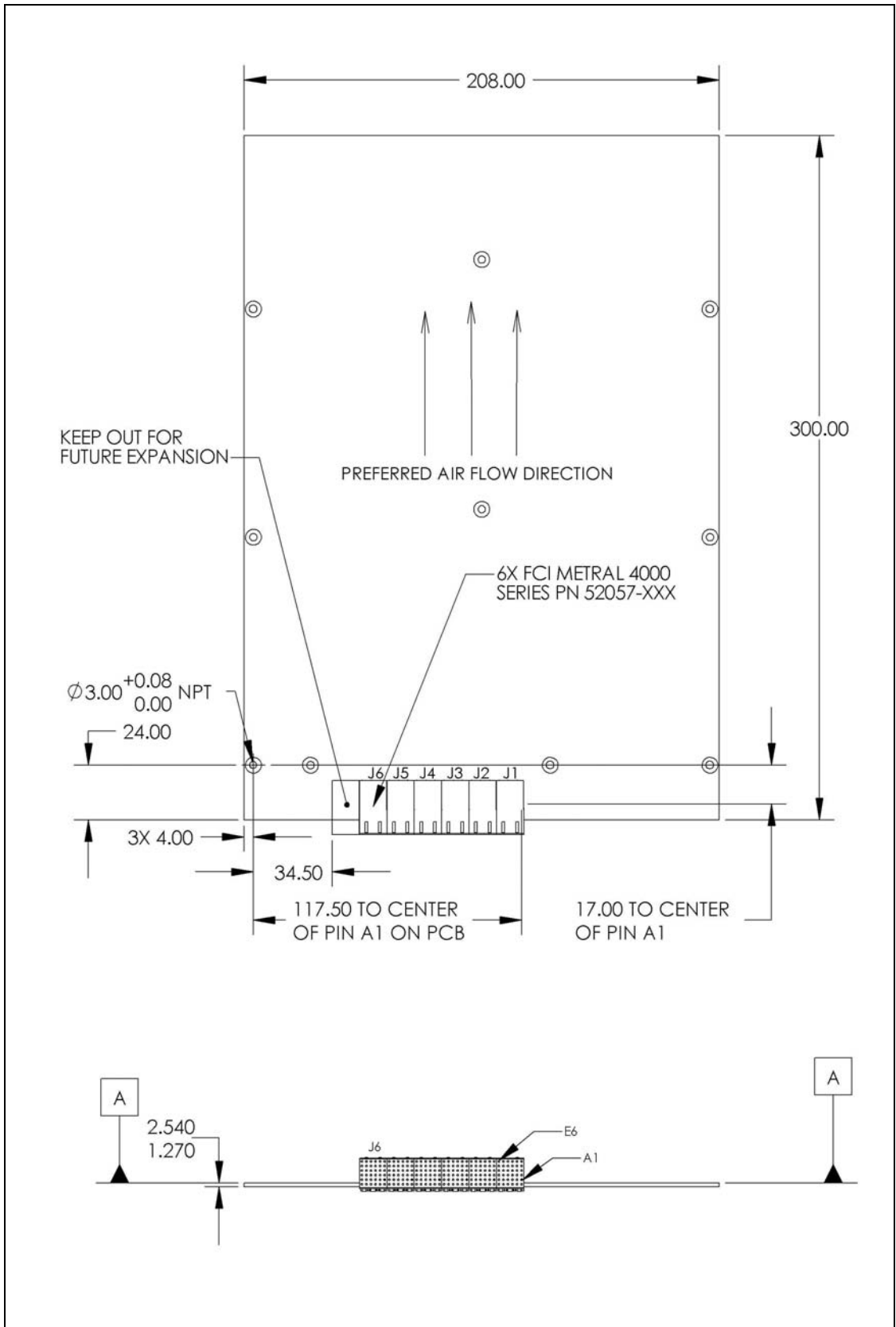


Figure D.13 – I/O controller module connector locations on 2xWide I/O module



## 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 90 - 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 6 that 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

$$t_c(i) = iT$$

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

$$t_D(i) = iT + \varepsilon_i$$

where the  $\varepsilon_i$  are time deviations from ideal, the clock to edge jitter. Over time, these perturbations are constrained by

$$\lim_{N \rightarrow \infty} \sum_{i=0}^{N-1} \frac{\varepsilon_i}{N} = 0$$

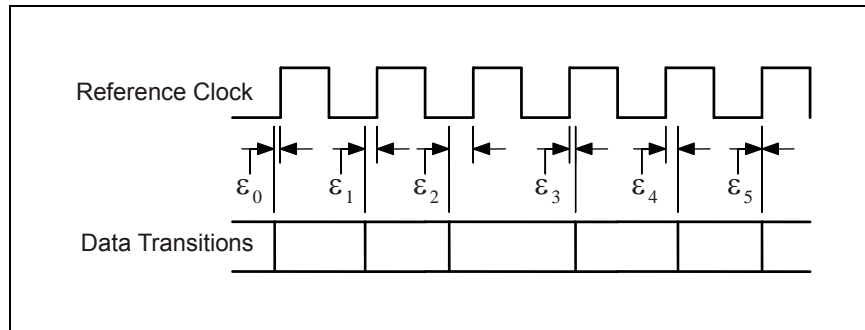
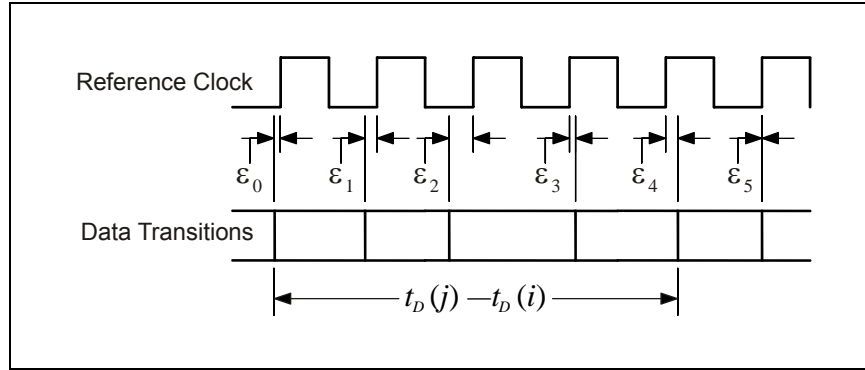


Figure D.14 – Jitter deviations

### E.2 Data to data (shown for historical reasons)

The time difference between data transitions is shown in Figure D.15 and given by

$$t_D(j) - t_D(i) = (j - i)T + \varepsilon_j - \varepsilon_i$$



**Figure D.15 – 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

$$t_D(i+k) - t_D(i) = kT + \varepsilon_j - \varepsilon_i$$

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  $k$ .

$$t_J(k) = \max [t_D(i+k) - t_D(i)]_{\forall i} - \min [t_D(i+k) - t_D(i)]_{\forall i}$$

$$t_J(k) = \max [kT + \varepsilon_{i+k} - \varepsilon_i]_{\forall i} - \min [kT + \varepsilon_{i+k} - \varepsilon_i]_{\forall i}$$

NOTE 91 - 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

$$t_J(k) = \max [\varepsilon_{i+k} - \varepsilon_i]_{\forall i} - \min [\varepsilon_{i+k} - \varepsilon_i]_{\forall i}$$

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.