

JEDEC STANDARD

Universal Flash Storage (UFS)

Version 4.0

JESD220F

(Revision of JESD220E, January 2020)

AUGUST 2022

JEDEC SOLID STATE TECHNOLOGY ASSOCIATION



NOTICE

JEDEC standards and publications contain material that has been prepared, reviewed, and approved through the JEDEC Board of Directors level and subsequently reviewed and approved by the JEDEC legal counsel.

JEDEC standards and publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for use by those other than JEDEC members, whether the standard is to be used either domestically or internationally.

JEDEC standards and publications are adopted without regard to whether or not their adoption may involve patents or articles, materials, or processes. By such action JEDEC does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the JEDEC standards or publications.

The information included in JEDEC standards and publications represents a sound approach to product specification and application, principally from the solid state device manufacturer viewpoint. Within the JEDEC organization there are procedures whereby a JEDEC standard or publication may be further processed and ultimately become an ANSI standard.

No claims to be in conformance with this standard may be made unless all requirements stated in the standard are met.

Inquiries, comments, and suggestions relative to the content of this JEDEC standard or publication should be addressed to JEDEC at the address below, or refer to www.jedec.org under Standards and Documents for alternative contact information.

Published by
©JEDEC Solid State Technology Association 2022
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2107

JEDEC retains the copyright on this material. By downloading this file the individual agrees not to charge for or resell the resulting material.

PRICE: Contact JEDEC

Printed in the U.S.A.
All rights reserved

PLEASE!

DON'T VIOLATE
THE
LAW!

This document is copyrighted by JEDEC and may not be
reproduced without permission.

For information, contact:

JEDEC Solid State Technology Association
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2107

or refer to www.jedec.org under Standards-Documents/Copyright Information.

UNIVERSAL FLASH STORAGE (UFS), VERSION 3.1**Contents**

	Page
1	1
2	1
3	2
3.1	3
3.2	4
3.3	5
3.4	5
4	6
4.1	6
4.2	7
4.3	7
5	8
5.1	8
5.1.1	8
5.1.2	8
5.1.3	9
5.1.4	10
5.1.5	10
5.1.6	10
5.1.7	10
5.1.8	10
5.2	11
5.3	11
5.4	12
5.4.1	12
5.4.2	12
5.4.3	13
5.5	13
5.5.1	14
5.6	18
5.7	19
6	19
6.1	19
6.2	22
6.3	22
6.4	23
6.4.1	27
6.4.2	28
6.5	29
6.6	30
6.7	31
6.8	32
7	33
7.1	33
7.1.1	33

7.1.2	Hardware Reset	34
7.1.3	EndPointReset	35
7.1.4	Logical Unit Reset	36
7.1.5	Host UniPro Warm Reset	36
7.1.6	Summary of Resets and Device Behavior	37
7.2	Power up ramp	38
7.3	Power off ramp	39
7.4	UFS Device Power Modes and LU Power Condition	40
7.4.1	Device Power Modes	40
7.4.2	Power Management Command: START STOP UNIT	50
7.4.3	Power Mode Control	52
7.4.4	Logical Unit Power Condition	54
8	UFS UIC Layer: MIPI M-PHY	55
8.1	Termination	55
8.2	Drive Levels	55
8.3	PHY State machine	55
8.4	HS Burst	56
8.4.1	HS Prepare Length Control	56
8.4.2	HS Sync Length Control	56
8.5	PWM Burst	56
8.5.1	LS Prepare Length Control	56
8.6	Adapt	56
8.7	UFS PHY Attributes	57
8.8	Electrical characteristics	58
8.8.1	Transmitter Characteristics	58
8.8.2	Receiver Characteristics	58
9	UFS UIC Layer: MIPI Unipro	59
9.1	Overview	59
9.2	Architectural Model	59
9.3	UniPro/UFS Transport Protocol Interface (Data Plane)	60
9.3.1	Flow control	60
9.3.2	Object sizes	60
9.4	UniPro/UFS Control Interface (Control Plane)	61
9.5	UniPro/UFS Transport Protocol Address Mapping	62
9.6	Options and Tunable Parameters of UniPro	63
9.6.1	UniPro PHY Adapter	63
9.6.2	UniPro Data Link Layer	64
9.6.3	UniPro Network Layer	64
9.6.4	UniPro Transport Layer	65
9.6.5	UniPro Device Management Entity Transport Layer	65
9.6.6	UniPro Attributes	66
10	UFS Transport Protocol (UTP) Layer	67
10.1	Overview	67
10.2	UTP and UniPro Specific Overview	68
10.2.1	Phases	68
10.2.2	Data Pacing	68
10.2.3	UniPro	68
10.3	UFS Transport Protocol Transactions Overview	69
10.4	Service Delivery Subsystem	69
10.5	UPIU Transactions	69

10.6	General UFS Protocol Information Unit Format	71
10.6.1	Overview	72
10.6.2	Basic Header Format	72
10.7	UFS Protocol Information Units	80
10.7.1	COMMAND UPIU	80
10.7.2	RESPONSE UPIU	84
10.7.3	DATA OUT UPIU	93
10.7.4	DATA IN UPIU	97
10.7.5	READY TO TRANSFER UPIU	103
10.7.6	TASK MANAGEMENT REQUEST UPIU	109
10.7.7	TASK MANAGEMENT RESPONSE UPIU	112
10.7.8	QUERY REQUEST UPIU	115
10.7.9	QUERY RESPONSE UPIU	128
10.7.10	REJECT UPIU	140
10.7.11	NOP OUT UPIU	143
10.7.12	NOP IN UPIU	145
10.7.13	Data out transfer rules	147
10.7.14	Overview - Data Out of Order Transfer	151
10.8	Logical Units	153
10.8.1	UFS SCSI Domain	153
10.8.2	UFS Logical Unit Definition	153
10.8.3	Well Known Logical Unit Definition	154
10.8.4	Logical Unit Addressing	154
10.8.5	Well Known Logical Unit Defined in UFS	155
10.8.6	Translation of 8-bit UFS LUN to 64-bit SCSI LUN Address	156
10.8.7	SCSI Write Command	157
10.8.8	SCSI Read Command	158
10.8.9	Unit Attention Condition	159
10.9	Application Layer and Device Manager Transport Protocol Services	160
10.9.1	UFS Initiator Port and Target Port Attributes	160
10.9.2	Execute Command procedure call transport protocol services	161
10.9.3	SCSI Command transport protocol service	162
10.9.4	SCSI Command Received transport protocol	163
10.9.5	Send Command Complete transport protocol service	164
10.9.6	Command Complete Received transport protocol service	165
10.9.7	Data transfer SCSI transport protocol services	166
10.9.8	Task Management Function procedure calls	170
10.9.9	Query Function transport protocol services	176
11	UFS Application (UAP) Layer – SCSI Commands	179
11.1	Universal Flash Storage Command Layer (UCL) Introduction	179
11.1.1	The Command Descriptor Block (CDB)	179
11.2	Universal Flash Storage Native Commands (UNC)	179
11.3	Universal Flash Storage SCSI Commands	180
11.3.1	General information about SCSI commands in UFS	181
11.3.2	INQUIRY Command	181
11.3.3	MODE SELECT (10) Command	184
11.3.4	MODE SENSE (10) Command	186
11.3.5	READ (6) Command	189
11.3.6	READ (10) Command	190
11.3.7	READ (16) Command	192
11.3.8	READ CAPACITY (10) Command	194

11.3.9	READ CAPACITY (16) Command	197
11.3.10	START STOP UNIT Command	200
11.3.11	TEST UNIT READY Command	201
11.3.12	REPORT LUNS Command	202
11.3.13	VERIFY (10) Command	207
11.3.14	WRITE (6) Command	209
11.3.15	WRITE (10) Command	211
11.3.16	WRITE (16) Command	214
11.3.17	REQUEST SENSE Command	217
11.3.18	FORMAT UNIT Command	219
11.3.19	PRE-FETCH (10) Command	221
11.3.20	PRE-FETCH (16) Command	224
11.3.21	SECURITY PROTOCOL IN Command	225
11.3.22	SECURITY PROTOCOL OUT Command	226
11.3.23	SEND DIAGNOSTIC Command	228
11.3.24	SYNCHRONIZE CACHE (10) Command	230
11.3.25	SYNCHRONIZE CACHE (16) Command	233
11.3.26	UNMAP Command	234
11.3.27	READ BUFFER Command	238
11.3.28	WRITE BUFFER Command	244
11.3.29	BARRIER Command	248
11.4	Mode Pages	249
11.4.1	Mode Page Overview	249
11.4.2	UFS Supported Pages	254
11.5	Vital product data parameters	261
11.5.1	Overview	261
11.5.2	VPD page format	261
11.5.3	Supported VPD Pages VPD page	262
11.5.4	Mode Page Policy VPD page	263
12	UFS Security	265
12.1	UFS Security Feature Support Requirements	265
12.2	Secure Mode	265
12.2.1	Description	265
12.2.2	Requirements	266
12.2.3	Implementation	268
12.3	Device Data Protection	273
12.3.1	Description and Requirements	273
12.3.2	Implementation	273
12.4	RPMB	274
12.4.1	Introduction	274
12.4.2	RPMB Well Known Logical Unit Description	274
12.4.3	Requirements	275
12.4.4	Implementation in Normal RPMB Mode	289
12.4.5	Implementation in Advanced RPMB Mode	290
12.4.6	SECURITY PROTOCOL IN/OUT Commands	291
12.4.7	RPMB Operations	297
12.5	Malware Protection	339
13	UFS Functional Descriptions	340
13.1	UFS Boot	340
13.1.1	Introduction	340

13.1.2	Boot Configuration	340
13.1.3	Initialization and boot code download process	343
13.1.4	Initialization process without boot code download	346
13.1.5	Boot Logical Unit Operations	347
13.1.6	Configurability	347
13.1.7	Security	348
13.2	Logical Unit Management	348
13.2.1	Introduction	348
13.2.2	Logical Unit features	348
13.2.3	Logical Unit Configuration	351
13.3	Logical Block Provisioning	357
13.3.1	Overview	357
13.3.2	Full Provisioning	357
13.3.3	Thin Provisioning	357
13.4	Host Device Interaction	358
13.4.1	Overview	358
13.4.2	Applicable Devices	358
13.4.3	Command Queue: Inter-LU Priority	358
13.4.4	Background Operations Mode	359
13.4.5	Power Off Notification	361
13.4.6	Dynamic Device Capacity	362
13.4.7	Data Reliability	366
13.4.8	Real-Time Clock Information	367
13.4.9	Timestamp Information	368
13.4.10	Context Management	368
13.4.11	System Data Tag Mechanism	372
13.4.12	Exception Events Mechanism	373
13.4.13	Queue Depth Definition	374
13.4.14	Device Life Span Mode	375
13.4.15	Refresh Operation	376
13.4.16	Temperature Event Notification	379
13.4.17	Performance Throttling Event Notification	379
13.4.18	WriteBooster	380
13.4.19	BARRIER COMMAND	384
13.5	UFS Cache	387
13.6	Production State Awareness (PSA)	388
13.6.1	Introduction	388
13.6.2	PSA flow	388
14	UFS Descriptors, Flags And Attributes	391
14.1	UFS Descriptors	391
14.1.1	Descriptor Types	392
14.1.2	Descriptor Indexing	392
14.1.3	Accessing Descriptors and Device Configuration	392
14.1.4	Descriptor Definitions	396
14.2	Flags	432
14.3	Attributes	436
15	UFS Mechanical Standard	450
Annex A	(informative) Dynamic Capacity Host Implementation Example	451
A.1	Overview	451
A.2	Method Outline	451
Annex B	(informative) Reference Clock Measurement Procedure	452

B.1	Random RMS Jitter Measurement Procedure	452
B.2	Deterministic Jitter Measurement Procedure	453
B.3	Equipment Requirements	454
B.4	Test Setup	454
Annex C (informative) Reference Clock Detection in HS-LSS : An Implementation Example		455
C.1	Overview	455
C.2	Method Outline	455
Annex D (informative) Board Design Guideline		456
D.1	Overview	456
D.2	Allowed Max Impedance & Typical Cap for System Board PCB	457
D.3	System Board PCB Noise Margin Recommendation	459
Annex E (informative) Differences Between Specification Revisions		460
E.1	Changes between JESD220F and its predecessor JESD220E (January 2020)	460
E.1.1	NEW features or new definitions	460
E.1.2	Changes in section 2 "Normative Reference"	460
E.1.3	Changes in features already defined in UFS 3.1	461
E.2	Changes between JESD220E and its predecessor JESD220D (January 2018)	462
E.2.1	New features or new definitions	462
E.2.2	Changes in section 2 "Normative Reference"	462
E.2.3	Changes in features already defined in UFS 3.0	462
E.3	Changes between JESD220D and its predecessor JESD220C (March 2016)	463
E.3.1	New features or new definitions	463
E.3.2	Changes in section 2 "Normative Reference"	463
E.3.3	Changes in features already defined in UFS 2.1	463
E.4	Changes between JESD220C and its predecessor JESD220B (September 2013)	464
E.4.1	New features or new definitions	464
E.4.2	Changes in section 2 "Normative Reference"	464
E.4.3	Changes in features already defined in UFS 2.0	464
E.5	Changes between JESD220B and its predecessor JESD220A (June 2012)	467
E.5.1	New features or new definitions	467
E.5.2	Changes in section 2 "Normative Reference"	467
E.5.3	Changes in features already defined in UFS 1.1	467

Figures

Figure 5.1 — UFS Top Level Architecture	8
Figure 5.2 — Usage of UDM_SAP	9
Figure 5.3 — Usage of UIO_SAP	9
Figure 5.4 — UFS System Model	11
Figure 5.5 — SCSI Domain Class Diagram	15
Figure 5.6 — UFS Domain Class Diagram	16
Figure 6.1 — Conceptual UFS Device Block Diagram.....	19
Figure 6.2 — Supply voltage power up timings.....	23
Figure 6.3 — bRefClkGatingWaitTime	26
Figure 6.4 — Clock input levels, rise time, and fall time	27
Figure 6.5 — Test Load Impedance	28
Figure 6.6 — Output driver and Input receiver levels	28
Figure 6.7 — Clock output levels, rise time and fall time	29
Figure 6.8 — DC and AC voltage Range for VCC/VCCQ/VCCQ2	32
Figure 7.1 — Power-on Reset	33
Figure 7.2 — Hardware Reset	34
Figure 7.3 — Reset AC timings	34
Figure 7.4 — EndPointReset	35
Figure 7.5 — Logical Unit Reset.....	36
Figure 7.6 — Power up ramps.....	38
Figure 7.7 — Power off ramps	39
Figure 7.8 — Power Mode State Machine	45
Figure 8.1 — Simplified example for I/O termination	55
Figure 9.1 — UniPro internal layering view (a) and UniPro Black Box view (b).....	59
Figure 9.2 — Physical lane connections.....	64
Figure 10.1 — EHS Entry Format	78
Figure 10.2 — Data out transfer example.....	96
Figure 10.3 — Data in transfer example.....	101
Figure 10.4 — DATA IN UPIU with retransmission sequence example	102
Figure 10.5 — READY TO TRANSFER UPIU sequence example	107
Figure 10.6 — READY TO TRANSFER UPIU with retransmission sequence example	108
Figure 10.7 — Example for data out transfer rule 1	147
Figure 10.8 — Example for Data Transfer Count mismatch.....	148
Figure 10.9 — Example for data out transfer rule 2	149
Figure 10.10 — Example for data out transfer rule 3	150
Figure 10.11 — Example for Data Out of Order Transfer - 1	151
Figure 10.12 — Example for Data Out of Order Transfer - 2	152
Figure 10.13 — UFS SCSI domain	153
Figure 10.14 — Logical Unit Addressing	154
Figure 10.15 — SCSI Write	157
Figure 10.16 — SCSI Read	158
Figure 10.17 — Command without Data Phase	161
Figure 10.18 — Command + Read Data Phase 1/2	167
Figure 10.19 — Command + Read Data Phase 2/2	167
Figure 10.20 — Command + Write Data Phase 1/2	169
Figure 10.21 — Command + Write Data Phase 2/2	169
Figure 10.22 — Task Management Function	173
Figure 10.23 — UFS Query Function	176
Figure 11.1 — UFS Command Layer.....	179
Figure 12.1 — Purge operation state machine.....	270
Figure 12.2 — Advanced RPMB Message Structure in EHS Field	290
Figure 12.3 — Request type message delivery.....	298
Figure 12.4 — Response Type Message delivery	300
Figure 12.5 — Authentication Key Programming Flow.....	302

Figure 12.6 — Read Counter Value Flow	303
Figure 12.7 — Authenticated Data Write Flow	306
Figure 12.8 — Authenticated Data Read Flow	308
Figure 12.9 — Authenticated Secure Write Protect Configuration Block Write Flow	311
Figure 12.10 — Authenticated Secure Write Protect Configuration Block Read Flow	313
Figure 12.11 — RPMB Purge Enable Flow	315
Figure 12.12 — RPMB Purge Status Read Flow	316
Figure 12.13 — Authentication Key Programming Flow (in Advanced RPMB Mode)	318
Figure 12.14 — Read Counter Value Flow (in Advanced RPMB Mode)	320
Figure 12.15 — Authenticated Data Write Flow (in Advanced RPMB Mode)	323
Figure 12.16 — Authenticated Data Read Flow (in Advanced RPMB Mode)	326
Figure 12.17 — Authenticated Secure Write Protect Configuration Block Write Flow (in Advanced RPMB Mode)	330
Figure 12.18 — Authenticated Secure Write Protect Configuration Block Read Flow (in Advanced RPMB Mode)	333
Figure 12.19 — RPMB Purge Enable Flow (in Advanced RPMB Mode)	336
Figure 12.20 — RPMB Purge Status Read Flow (in Advanced RPMB Mode)	339
Figure 13.1 — UFS System Diagram	340
Figure 13.2 — Example of UFS Device Memory Organization for Boot	342
Figure 13.3 — Device Initialization and Boot Procedure Sequence Diagram	345
Figure 13.4 — Example of UFS Device Memory Organization	349
Figure 13.5 — Physical Memory Resource State Machine	365
Figure 13.6 — Example of data status after a power failure during reliable write operation	366
Figure 13.7 — Concept of WriteBooster feature	380
Figure 13.8 — Example of “LU dedicated buffer” mode configuration	381
Figure 13.9 — Example of “shared buffer” mode configuration	382
Figure 13.10 — Example of Barrier operation	384
Figure 13.11 — Example of the Barrier between commands with simple property	385
Figure 13.12 — Example of the Barrier command between commands with different properties	385
Figure 13.13 — Example of the Barrier command between commands with high priorities	385
Figure 13.14 — Barrier operation scope (LU level)	386
Figure 13.15 — PSA flow	389
Figure 13.16 — PSA state machine	390
Figure 14.1 — Descriptor Organization	392
Figure 14.2 — Read Request Descriptor	394
Figure 14.3 — Write Request Descriptor	395
Figure B.1 — Test Setup	454
Figure C.1 — REF CLK frequency detection in HS-LSS	455
Figure D.1 — Z(f) measuring and observation point	457
Figure D.2 — Zprofile Z(f) of the system at the UFS package solder ball (without UFS component)	458
Figure D.3 — A simplified Z(f) system electrical model and frequency response of the behavioral PDN electrical load model without the UFS component per voltage domain per channel	458

Contents (cont'd)

Tables

Table 5.1 — UFS Signals	12
Table 5.2 — ManufacturerID and DeviceClass Attributes	13
Table 6.1 — Signal Name and Definitions	20
Table 6.2 — Electrostatic Discharge Sensitivity Characteristics	21
Table 6.3 — Reset Signal and LSS Signal Electrical Parameters	22
Table 6.4 — UFS Power Supply Parameters	22
Table 6.4 — Reference Clock parameters	24
Table 6.6 — Reference Clock Frequency and HS Operation	25
Table 6.7 — HS-BURST Rates	27
Table 6.8 — Host controller reference clock parameters	28
Table 6.9 — Charge pump capacitors description	29
Table 6.10 — Charge pump related ball names	30
Table 6.11 — ZQ Calibration Resistors Description	30
Table 6.12 — Absolute maximum DC ratings and Operating Conditions	31
Table 6.13 — AC and DC Voltage operating conditions	32
Table 7.1 — Reset timing parameters	34
Table 7.2 — Reset States	37
Table 7.3 — UniPro Attributes, UFS Attributes and UFS Flags reset	37
Table 7.4 — Allowed SCSI commands and UPIU for each Power Mode	48
Table 7.5 — Device Well Known Logical Unit Responses to SSU command	49
Table 7.6 — Device Well Known Logical Unit Responses to commands other than SSU	50
Table 7.7 — Pollable Sense Data for each Power Modes	50
Table 7.8 — START STOP UNIT command	50
Table 7.9 — START STOP UNIT fields	51
Table 7.10 — Attribute for Power Mode Control	52
Table 7.11 — Device Descriptor parameters	52
Table 7.12 — Power Parameters Descriptor fields	53
Table 7.13 — Format for Power Parameter element	53
Table 7.14 — Logical Unit Response to SCSI command	54
Table 8.1 — UFS PHY M-TX Capability Attributes ⁽⁰⁾	57
Table 8.2 — UFS PHY M-RX Capability Attributes ⁽⁰⁾	58
Table 9.1 — UFS Initiator and Target Port Identifiers	63
Table 9.2 — UniPro Attribute	66
Table 10.1 — UPIU Transaction Codes	69
Table 10.2 — UPIU Transaction Code Definitions	70
Table 10.3 — General format of the UFS Protocol Information Unit	71
Table 10.4 — Basic Header Format	72
Table 10.5 — Transaction Type Format	72
Table 10.6 — UPIU Flags	74
Table 10.7 — Task Attribute definition	74
Table 10.8 — UTP Response Values	75
Table 10.9 — UPIU associated to a single task	75
Table 10.10 — Initiator ID Composition	76
Table 10.11 — Command Set Type	76
Table 10.12 — EHS Entry Format	78
Table 10.13 — COMMAND UPIU	80
Table 10.14 — Flags definition for COMMAND UPIU	81
Table 10.15 — RESPONSE UPIU	84
Table 10.16 — Flags definition for RESPONSE UPIU	85
Table 10.17 — SCSI Status Values	86
Table 10.18 — Flags and Residual Count Relationship	88
Table 10.19 — SCSI fixed format sense data	90

Table 10.20 — Sense Key	92
Table 10.21 — DATA OUT UPIU	93
Table 10.22 — Flags definition for DATA OUT UPIU	94
Table 10.23 — DATA IN UPIU	97
Table 10.24 — Flags definition for DATA OUT UPIU	98
Table 10.25 — READY TO TRANSFER UPIU	103
Table 10.26 — Flags definition for DATA OUT UPIU	104
Table 10.27 — Task Management Request UPIU	109
Table 10.28 — Task Management Function values	110
Table 10.29 — Task Management Input Parameters	111
Table 10.30 — Task Management Response UPIU	112
Table 10.31 — Task Management Output Parameters	113
Table 10.32 — Task Management Service Response	114
Table 10.33 — QUERY REQUEST UPIU	115
Table 10.34 — Query Function field values	117
Table 10.35 — Transaction specific fields	118
Table 10.36 — Query Function opcode values	118
Table 10.37 — Read descriptor	119
Table 10.38 — Write Descriptor	120
Table 10.39 — Read Attribute	121
Table 10.40 — Write Attribute	122
Table 10.41 — Read Flag	123
Table 10.42 — Set Flag	124
Table 10.43 — Clear Flag	125
Table 10.44 — Toggle Flag	126
Table 10.45 — NOP	127
Table 10.46 — QUERY RESPONSE	128
Table 10.47 — Query Response Code	129
Table 10.48 — Transaction Specific Fields	130
Table 10.49 — Read Descriptor	131
Table 10.50 — Write Descriptor	132
Table 10.51 — Read Attribute Response Data Format	133
Table 10.52 — Write Attribute	134
Table 10.53 — Read Flag Response Data Format	135
Table 10.54 — Set Flag	136
Table 10.55 — Clear Flag	137
Table 10.56 — Toggle Flag	138
Table 10.57 — NOP	139
Table 10.58 — Reject UPIU	140
Table 10.59 — Basic Header Status Description	141
Table 10.60 — E2E Status Definition	142
Table 10.61 — NOP OUT UPIU	143
Table 10.62 — NOP IN UPIU	145
Table 10.63 — Parameters related to data out transfer rules	150
Table 10.64 — Well known logical unit commands	155
Table 10.65 — Examples of logical unit representation format	156
Table 10.66 — Events for UAC establishment	159
Table 10.67 — Commands for exceptional behavior on UAC	159
Table 10.68 — UFS Initiator Port and Target Port Attributes	160
Table 10.69 — Send SCSI Command transport protocol service	162
Table 10.70 — SCSI Command Received transport protocol	163
Table 10.71 — Send Command Complete transport protocol service	164
Table 10.72 — Command Complete Received transport protocol service	165
Table 10.73 — Send Data-In transport protocol service	166
Table 10.74 — Data-In Delivered transport protocol service	166

Table 10.75 — Receive Data-Out transport protocol service	168
Table 10.76 — Data-Out Received transport protocol service	168
Table 10.77 — Task Management Function procedure calls	170
Table 10.78 — SCSI transport protocol service responses.....	170
Table 10.79 — Send Task Management Request SCSI transport protocol service request.....	174
Table 10.80 — Task Management Request Received SCSI transport protocol service indication	174
Table 10.81 — Task Management Function Executed SCSI transport protocol service response	174
Table 10.82 — Received Task Management Function Executed SCSI transport protocol service confirmation.....	175
Table 10.83 — Send Query Request UFS transport protocol service.....	176
Table 10.84 — Query Request Received UFS transport protocol service indication	177
Table 10.85 — Query Function Executed UFS transport protocol service response.....	177
Table 10.86 — Received Query Function Executed UFS transport protocol service confirmation	178
Table 11.1 — UFS SCSI Command Set.....	180
Table 11.2 — INQUIRY command.....	181
Table 11.3 — Standard INQUIRY data format	182
Table 11.4 — Standard INQUIRY Response Data	183
Table 11.5 — MODE SELECT (10) command.....	184
Table 11.6 — Mode Select Command Parameters	185
Table 11.7 — MODE SENSE (10) command	187
Table 11.8 — Mode Sense Command Parameters	187
Table 11.9 — Page Control Function	188
Table 11.10 — READ (6) command.....	189
Table 11.11 — READ (10) command	190
Table 11.12 — READ (16) command	192
Table 11.13 — READ CAPACITY (10) command	194
Table 11.14 — Read Capacity (10) Parameter Data.....	195
Table 11.15 — READ CAPACITY (16) command	197
Table 11.16 — Read Capacity (16) Parameter Data.....	198
Table 11.17 — START STOP UNIT command.....	200
Table 11.18 — TEST UNIT READY command.....	201
Table 11.19 — REPORT LUNS command.....	202
Table 11.20 — Report LUNS Command Parameters.....	203
Table 11.21 — SELECT REPORT field	203
Table 11.22 — Report LUNS Parameter Data Format.....	204
Table 11.23 — Single level LUN structure using peripheral device addressing method	204
Table 11.24 — Well Known Logical Unit Extended Addressing Format	205
Table 11.25 — Format of LUN field in UPIU.....	206
Table 11.26 — Well known logical unit numbers	206
Table 11.27 — VERIFY (10) command	207
Table 11.28 — Verify Command Parameters.....	208
Table 11.29 — WRITE (6) command	209
Table 11.30 — WRITE (10) command	211
Table 11.31 — WRITE (16) command	214
Table 11.32 — REQUEST SENSE command	217
Table 11.33 — FORMAT UNIT command	219
Table 11.34 — Format Unit Command Parameters	220
Table 11.35 — PRE_FETCH command.....	221
Table 11.36 — PRE-FETCH Command Parameters.....	222
Table 11.37 — PRE-FETCH (16) command.....	224
Table 11.38 — SECURITY PROTOCOL IN command.....	225
Table 11.39 — SECURITY PROTOCOL OUT command.....	226
Table 11.40 — SEND DIAGNOSTIC command.....	228
Table 11.41 — Send Diagnostic Parameters	228
Table 11.42 — SYNCHRONIZE CACHE (10) command	230
Table 11.43 — Synchronize Cache Command Parameters	231
Table 11.44 — SYNCHRONIZE CACHE (16) Command Descriptor Block	233

Table 11.45 — UNMAP command	234
Table 11.46 — UNMAP parameter list	235
Table 11.47 — UNMAP block descriptor	236
Table 11.48 — READ BUFFER command.....	238
Table 11.49 — Read Buffer Command Mode Field Values.....	239
Table 11.50 — Buffer ID Field for Error History Mode	240
Table 11.51 — Error history directory	241
Table 11.52 — EHS_SOURCE field.....	242
Table 11.53 — Error history directory entry	242
Table 11.54 — WRITE BUFFER command.....	244
Table 11.55 — Write Buffer Command Parameters.....	245
Table 11.56 — Write Buffer Command Mode Field Values.....	245
Table 11.57 — BARRIER command	248
Table 11.58 — Mode page code usage.....	249
Table 11.59 — UFS Mode parameter list.....	250
Table 11.60 — UFS Mode parameter header (10).....	250
Table 11.61 — Mode Parameter Header Detail.....	251
Table 11.62 — Page_0 mode page format.....	252
Table 11.63 — Page 0 Format parameters.....	252
Table 11.64 — Sub_page mode page format.....	253
Table 11.65 — Subpage Format parameters	253
Table 11.66 — UFS Supported Pages	254
Table 11.67 — Control Mode Page default value	255
Table 11.68 — Control Mode Page Parameters	256
Table 11.69 — Read-Write Error Recovery Mode Page default value	257
Table 11.70 — Read-Write Error Recovery Parameters	258
Table 11.71 — Caching Mode Page default value	259
Table 11.72 — Caching Mode Page Parameters	260
Table 11.73 — VPD page format	261
Table 11.74 — Supported VPD Pages VPD page	262
Table 11.75 — Mode Page Policy VPD page.....	263
Table 11.76 — Mode page policy descriptor.....	263
Table 11.77 — MODE PAGE POLICY field	264
Table 12.1 — Secure Write Protect Configuration Block for Normal RPMB	277
Table 12.2 — Secure Write Protect Configuration Block for Advanced RPMB.....	278
Table 12.3 — Secure Write Protect Entry	279
Table 12.4 — Write Protect Type field	280
Table 12.5 — RPMB Purge Response Packet Format (legacy RPMB mode).....	281
Table 12.6 — RPMB Purge Response Packet Format (Advanced RPMB mode)	281
Table 12.7 — RPMB Message Components	284
Table 12.8 — Request Message Types.....	285
Table 12.9 — Response Message Types	286
Table 12.10 — Result data structure	287
Table 12.11 — Result code definition	288
Table 12.12 — RPMB Message Data Frame.....	289
Table 12.13 — Advanced RPMB Meta Information.....	291
Table 12.14 — CDB format of SECURITY PROTOCOL IN/OUT commands	292
Table 12.15 — SECURITY PROTOCOL SPECIFIC field for protocol ECh.....	293
Table 12.16 — Security Protocol specific field values for protocol 00h	294
Table 12.17 — Supported security protocols SECURITY PROTOCOL IN parameter data	295
Table 12.18 — UFS Supported security protocols SECURITY PROTOCOL IN parameter data	295
Table 12.19 — Certificate data SECURITY PROTOCOL IN parameter data.....	296
Table 12.20 — UFS certificate data SECURITY PROTOCOL IN parameter data	296
Table 12.21 — Expected Data Transfer Length value for Request Type Messages.....	297
Table 12.22 — Expected Data Transfer Length value for Response Type Messages	299

Table 13.1 — bBootLunEn Attribute	341
Table 13.2 — Valid UPIUs and SCSI Commands for Each Initialization Phase	346
Table 13.3 — Logical unit configurable parameters	352
Table 13.4 — Parameter for controlling logical unit data reliability	367
Table 14.1 — Descriptor identification values	391
Table 14.2 — Generic Descriptor Format	396
Table 14.3 — Logical Unit Descriptor Format.....	396
Table 14.4 — Device Descriptor	397
Table 14.5 — wManufacturerID definition	405
Table 14.6 — Configuration Descriptor Format (INDEX = 00h)	406
Table 14.7 — Configuration Descriptor Format (INDEX = 01h)	406
Table 14.8 — Configuration Descriptor Format (INDEX = 02h)	406
Table 14.9 — Configuration Descriptor Format (INDEX = 03h)	407
Table 14.10 — Configuration Descr. Header and Device Descr. Conf. parameters (INDEX = 00h)	408
Table 14.11 — Configuration Descr. Header with INDEX = 01h/02h/03h	409
Table 14.12 — Unit Descriptor configurable parameters.....	410
Table 14.13 — Geometry Descriptor	411
Table 14.14 — Unit Descriptor	419
Table 14.15 — RPMB Unit Descriptor	423
Table 14.16 — Power Parameters Descriptor.....	425
Table 14.17 — Interconnect Descriptor	426
Table 14.18 — Manufacturer Name String	426
Table 14.19 — Product Name String.....	427
Table 14.20 — OEM_ID String	427
Table 14.21 — Serial Number String Descriptor	428
Table 14.22 — Product Revision Level String	428
Table 14.23 — Device Health Descriptor	429
Table 14.24 — Vendor Specific Descriptor	431
Table 14.25 — Flags access properties	432
Table 14.26 — Flags	433
Table 14.27 — Attributes access properties	436
Table 14.28 — Attributes	437
Table B.1 — Maximum Random RMS Jitter Amount	452
Table D.1 — Allowed Max Impedance & Typical Capacitor at System Board Design.....	457
Table D.2 — Recommended Max PMIC & PCB Noise margin	459
Table D.3 — (alternative) 2.55 VCC to allow more PMIC & PCB Noise margin	459

Foreword

This standard has been prepared by JEDEC. The purpose of this standard is definition of a UFS Universal Flash Storage electrical interface and a UFS memory device. This standard defines a unique UFS feature set and includes the feature set of eMMC standard as a subset. This standard references several other standard specifications by MIPI (M-PHY and UniPro specifications) and INCITS T10 (SBC, SPC and SAM draft standards) organizations.

Introduction

The UFS electrical interface is a universal serial communication bus which can be utilized for different types of applications. It's based on MIPI M-PHY specification as physical layer for optimized performance and power. The UFS architectural model references the INCITS T10 SAM model for ease of adoption.

The UFS device is a universal data storage and communication media. It is designed to cover a wide area of applications as smart phones, cameras, organizers, PDAs, digital recorders, MP3 players, internet tablets, electronic toys, etc.

This page intentionally left blank.

UNIVERSAL FLASH STORAGE (UFS), VERSION 4.0

(From JEDEC Board Ballot JCB-22-26, formulated under the cognizance of the JC-64.1 Subcommittee on Electrical Specifications and Command Protocols, Item # 139.55)

1 Scope

This standard specifies the characteristics of the UFS electrical interface and the memory device. Such characteristics include (among others) low power consumption, high data throughput, low electromagnetic interference and optimization for mass memory subsystem efficiency. The UFS electrical interface is based on an advanced differential interface by MIPI M-PHY specification which together with the MIPI UniPro specification forms the interconnect of the UFS interface. The architectural model is referencing the INCITS T10 (SCSI) SAM standard and the command protocol is based on INCITS T10 (SCSI) SPC and SBC standards.

2 Normative Reference

The following normative documents contain provisions that, through reference in this text, constitute provisions of this standard. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated. For undated references, the latest edition of the normative document referred to applies.

[MIPI-M-PHY], *MIPI Alliance Specification for M-PHY[®], Version 5.0, 23 June 2021*

[MIPI-UniPro], *MIPI Alliance Specification for Unified Protocol (UniPro[®]), Version 2.0, In Press*

[MIPI-DDB], *MIPI Alliance Specification for Device Descriptor Block (DDB), Version 1.0*

[MIPI-AP], *Application Note for UniPro[®] v1.8*

[SAM], *INCITS T10 draft standard: SCSI Architecture Model – 5 (SAM–5), Revision 05, 19 May 2010*

[SPC], *INCITS T10 draft standard: SCSI Primary Commands – 4 (SPC-4), Revision 27, 11 October 2010*

[SBC], *INCITS T10 draft standard: SCSI Block Commands – 3 (SBC–3), Revision 24, 05 August 2010*

[JESD8-12A], *1.2 V +/- 0.1V (Normal Range) and 0.8 - 1.3 V (Wide Range) Power Supply Voltage and Interface Standard for Nonterminated Digital Integrated Circuits*

[HBM-MM], *JEDEC Recommended ESD Target Levels for HBM/MM Qualification, JEP155A.01, March 2012*

[CDM], *JEDEC Recommended ESD-CDM Target Levels, JEP157, October 2009*

[HMAC-SHA], *Eastlake, D. and T. Hansen, US Secure Hash Algorithms (SHA and HMAC-SHA), RFC 4634, July 2006.*

[JEP106], *Standard Manufacturer's identification code, JEP106*

[JESD21C], *Multichip Packages (MCP) and Discrete eMMC, e2MMC, and UFS*

[JESD220-3], *UFS Host Performance Booster (HPB) Extension Specification*

[JESD220-4], *UFS File Based Optimization (FBO) Extension Specification*

3 Terms and Definitions

For the purposes of this standard, the terms and definitions given in the document included in clause 2, Normative Reference, and the following apply.

Application Client: An entity that is the source of SCSI commands and task management function requests in the host.

Byte: An 8-bit data value with most significant bit labeled as bit 7 and least significant bit as bit 0.

Command Descriptor Block: The structure used to communicate commands from an application client to a device server. A CDB may have a fixed length of up to 16 bytes or a variable length of between 12 and 260 bytes.

Device ID: The bus address of a UFS device.

Device Server: An entity in the device that processes SCSI commands and task management functions.

Doubleword: A 32-bit data value with most significant bit labeled as bit 31 and least significant bit as bit 0.

Dword: 32-bit data value, a Doubleword.

Gigabyte: 1,073,741,824 or 2^{30} bytes.

Host: An entity or a device with the characteristics of a primary computing device that includes one or more SCSI initiator devices.

Initiator device: Within a transaction, the originator of a SCSI command request message to a target device.

Kilobyte: 1024 or 2^{10} bytes.

Logical Unit: A logical unit is an internal entity of a bus device that performs a certain function or addresses a particular space or configuration within a bus device.

Logical Unit Number: A numeric value that identifies a logical unit within a device

Megabyte: 1,048,576 or 2^{20} bytes.

Quadword: A 64-bit data value with most significant bit labeled as bit 63 and least significant bit as 0.

Segment: A specified number of sequentially addressed bytes representing a data structure or section of a data structure.

Segment ID: A 16-bit value that represents an index into a table or an address of a segment descriptor or simply an absolute value that is an element of an absolute address

SCSI Request Block: A data packet that contains a multi-byte SCSI command and additional contextual information needed to carry out the command operation. A SCSI Request Block is built by the host and is targeted at a particular bus device.

Target device: Within a transaction, the recipient of a SCSI command request message from an initiator device.

Task: A task is a SCSI command which includes all transactions to complete all data transfers and a status response that will satisfy the requirements of the requested services of the command.

Transaction: A UFS primitive action which results in transmission of serial data packets between a target device and initiator device.

3 Terms and Definitions (cont'd)

Terabyte: 1.099.511.627.776 or 2^{40} bytes.

UFS Protocol Information Unit: Information transfer (communication) between a UFS host and device is done through messages which are called UFS Protocol Information Units. These messages are UFS defined data structures that contain a number of sequentially addressed bytes arranged as various information fields.

Unit: A bus device

Unit Attention: A condition of a bus device utilizing the SCSI protocol where it needs to be serviced before it can continue processing requests and responses.

Word: A 16-bit data value with most significant bit labeled as bit 15 and least significant bit as bit 0.

3.1 Acronyms

CDB	Command Descriptor Block
CPort	A CPort is a Service Access Point on the UniPro Transport Layer (L4) within a Device that is used for Connection-oriented data transmission
DMA	Direct Memory Access
DSC	Digital Still Camera
FFU	Field Firmware Update
GB	Gigabyte
HCI	Host Controller Interface
HPB	UFS Host Performance Booster
KB	Kilobyte
LBA	Logical Block Address
LUN	Logical Unit Number
MB	Megabyte
MIPI	Mobile Industry Processor Interface
MP3	MPEG-2 Audio Layer 3
NA	Not applicable
NU	Not used
PDU	Protocol Data Unit
PLL	Phase-Locked Loop
PMP	Portable media player
PSA	Production State Awareness
PWM	Pulse Width Modulation
RFU	Reserved for future use
RPMB	Replay Protected Memory Block

3.1 Acronyms (cont'd)

SBC	SCSI Block Commands
SID	Segment ID
SDU	Service Data Unit
SPC	SCSI Primary Commands
TB	Terabyte
T_PDU	MIPI Unipro Protocol Data Unit
T_SDU	MIPI Unipro protocol Service Data Unit
UFS	Universal Flash Storage
UMPC	Ultra-Mobile PC
UniPro	Unified Protocol
UPIU	UFS Protocol Information Unit
UTP	UFS Transport Protocol

3.2 Conventions

This standard follows some conventions used in SCSI documents since it adopts several SCSI standards.

A binary number is represented in this standard by any sequence of digits consisting of only the Western-Arabic numerals 0 and 1 immediately followed by a lower-case b (e.g., 0101b). Spaces may be included in binary number representations to increase readability or delineate field boundaries (e.g., 0 0101 1010b).

A hexadecimal number is represented in this standard by any sequence of digits consisting of only the Western-Arabic numerals 0 through 9 and/or the upper-case English letters A through F immediately followed by a lower-case h (e.g., FA23h). Spaces may be included in hexadecimal number representations to increase readability or delineate field boundaries (e.g., B FD8C FA23h).

A decimal number is represented in this standard by any sequence of digits consisting of only the Western-Arabic numerals 0 through 9 not immediately followed by a lower-case b or lower-case h (e.g., 25).

A range of numeric values is represented in this standard in the form "a to z", where a is the first value included in the range, all values between a and z are included in the range, and z is the last value included in the range (e.g., the representation "0h to 3h" includes the values 0h, 1h, 2h, and 3h).

When the value of the bit or field is not relevant, x or xx appears in place of a specific value.

The first letter of the name of a Flag is a lower-case f (e.g., fMyFlag).

The first letter of the name of a parameter included in a Descriptor or the first letter of the name of an Attribute is:

- a lower-case b if the parameter or the Attribute size is one byte (e.g., bMyParameter),
- a lower-case w if the parameter or the Attribute size is two bytes (e.g., wMyParameter),
- a lower-case d if the parameter or the Attribute size is four bytes (e.g., dMyParameter),
- a lower-case q if the parameter or the Attribute size is eight bytes (e.g., qMyParameter).

3.3 Keywords

Several keywords are used to differentiate levels of requirements and options, as follow:

Can - A keyword used for statements of possibility and capability, whether material, physical, or causal (*can* equals *is able to*).

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.

Ignored - A keyword that describes bits, bytes, quadlets, or fields whose values are not checked by the recipient.

Mandatory - A keyword that indicates items required to be implemented as defined by this standard.

May - A keyword that indicates a course of action permissible within the limits of the standard (*may* equals *is permitted*).

Must - The use of the word *must* is deprecated and shall not be used when stating mandatory requirements; *must* is used only to describe unavoidable situations.

Obsolete - A keyword indicating that an item was defined in prior standards but has been removed from this standard.

Optional - A keyword that describes features which are not required to be implemented by this standard. However, if any optional feature defined by the standard is implemented, it shall be implemented as defined by the standard.

Reserved - A keyword used to describe objects—bits, bytes, and fields—or the code values assigned to these objects in cases where either the object or the code value is set aside for future standardization. Usage and interpretation may be specified by future extensions to this or other standards. A reserved object shall be zeroed or, upon development of a future standard, set to a value specified by such a standard. The recipient of a reserved object shall not check its value. The recipient of a defined object shall check its value and reject reserved code values.

Shall - A keyword that indicates a mandatory requirement strictly to be followed in order to conform to the standard and from which no deviation is permitted (*shall* equals *is required to*). Designers are required to implement all such mandatory requirements to assure interoperability with other products conforming to this standard.

Should - A keyword used to indicate that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required; or that (in the negative form) a certain course of action is deprecated but not prohibited (*should* equals *is recommended that*).

Will - The use of the word *will* is deprecated and shall not be used when stating mandatory requirements; *will* is only used in statements of fact.

3.4 Abbreviations

etc. - And so forth (Latin: et cetera)

e.g. - For example (Latin: exempli gratia)

i.e. - That is (Latin: id est)

4 Introduction

Universal Flash Storage (UFS) is a simple, high performance, mass storage device with a serial interface. It is primarily for use in mobile systems, between host processing and mass storage memory devices. The following is a summary of the UFS device features.

4.1 General Features

- Target performance
 - High speed GEARs ⁽¹⁾
 - Support for GEAR1 is mandatory
 - Support for GEAR2 is mandatory
 - Support for GEAR3 is mandatory
 - Support for GEAR4 is mandatory
 - Support for GEAR5 is mandatory
- Target host applications
 - Mobile phone, UMPC, DSC, PMP, MP3 and any other applications that require mass storage, bootable mass storage, and external card
- Target device types
 - External card
 - Embedded device
 - Mass storage and bootable mass storage
 - Future expansion of device class types
 - I/O devices, camera, wireless, ... , etc.
- Topology
 - One device per UFS port.
- UFS Layering
 - UFS Command Set Layer (UCS)
 - Simplified SCSI command set based on SBC and SPC. UFS will not modify these SBC and SPC Compliant commands. Option for defining UFS Native command and future extension exist.
 - UFS Transport Protocol Layer (UTP)
 - JEDEC to define the supported protocol layer, i.e., UTP for SCSI. This does not exclude the support of other protocol in UFS Transport Protocol Layer.
 - UFS Interconnect Layer (UIC)
 - MIPI UniPro[®] [MIPI-UniPro] is adopted for data link layer
 - MIPI M-PHY[®] [MIPI-M-PHY] is adopted for physical layer

NOTE 1 See 6.4.1 for details.

4.2 Interface Features

- Three power supplies
 - VCCQ power supply: 1.2 V (nominal)
 - VCCQ2 power supply: 1.8 V (nominal)
 - VCC power supply: 2.5 V (nominal)
- Signaling as defined by [MIPI-M-PHY]
 - 400 or 620 mVp (not terminated)
 - 200 or 310 mVp (terminated)
- 8b10b line coding, as defined by MIPI M-PHY
- High reliability – BER under 10^{-12}
- Two signaling schemes
 - Low-speed mode with PWM signaling scheme
 - High-Speed burst mode
- Multiple gears defined for High-Speed mode
- Only Gear 1 supported in Low Speed mode
- Adapt (M-PHY[®] versions 4.1 and above)
 - M-RX equalizer training to adapt to the channel characteristic

4.3 Functional Features

UFS functional features are NAND management features. These include

- Similar functional features as eMMC
- Boot Operation Mode
- Multiple logical units with configurable characteristics
- Replay Protected Memory Block (RPMB)
- Reliable write operation
- Background operations
- Secure operations, Purge and Erase to enhance data security
- Write Protection options, including Permanent and Power-On Write Protection
- Signed access to a Replay Protected Memory Block
- HW Reset Signal
- Task management operations
- Power management operations

5 UFS Architecture Overview

5.1 UFS Top Level Architecture

Figure 5.1 shows the Universal Flash Storage (UFS) top level architecture.

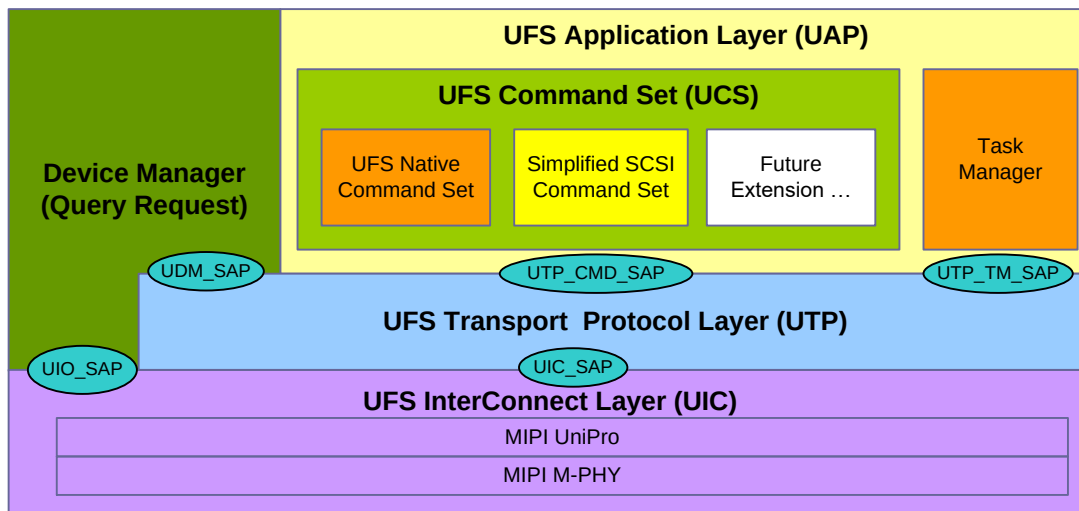


Figure 5.1 — UFS Top Level Architecture

UFS communication is a layered communication architecture. It is based on SCSI SAM architectural model [SAM].

5.1.1 Application Layer

The application layer consists of the UFS command set (UCS), the device manager and the Task Manager. The UCS will handle the normal commands like read, write, and so on. UFS may support multiple command sets. UFS is designed to be protocol agnostic. The command set for this version UFS standard is based on SCSI command set. In particular, a simplified SCSI command set was selected for UFS. UFS Native command set can be supported when it is needed to extend the UFS functionalities.

The Task Manager handles commands meant for command queue control. The Device Manager will provide device level control like Query Request and lower level link-layer control.

5.1.2 UFS Device Manager

The device manager has the following two responsibilities:

- Handling device level operations.
- Managing device level configurations.

Device level operations include functions such as device power management, settings related to data transfer, background operations enabling, and other device specific operations.

Device level configuration is managed by the device manager by maintaining and storing a set of descriptors. The device manager handles commands like query request which allow to modify or retrieve configuration information of the device.

5.1.3 Service Access Points

As seen from the diagram the device manager interacts with lower layers using the following two service access points:

- UDM_SAP
- UIO_SAP

UDM_SAP is the service access point exposed by the UTP for the device manager to allow handling of device level operations and configurations. For example the handling of query request for descriptors would be done using this service access point. Figure 5.2 depicts the usage of the service access point.

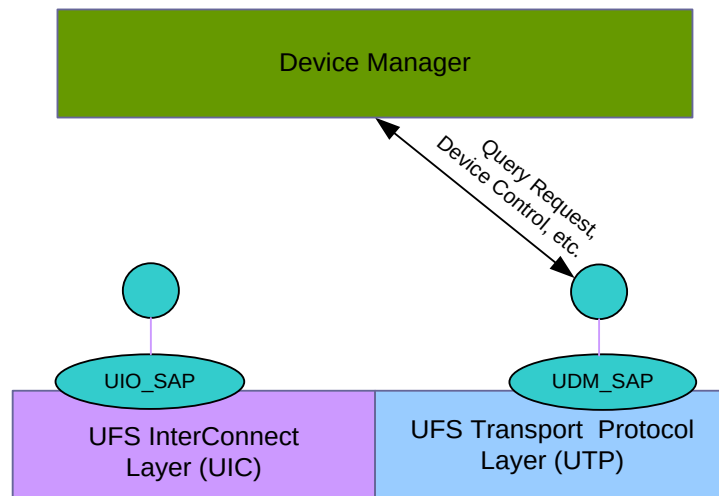


Figure 5.2 — Usage of UDM_SAP

UIO_SAP is the service access point exposed by the UIC layer for the device manager to trigger the reset of the UIC layer and to transfer requests and responses related to UIC management functions. Figure 5.3 depicts the usage of the service access point.

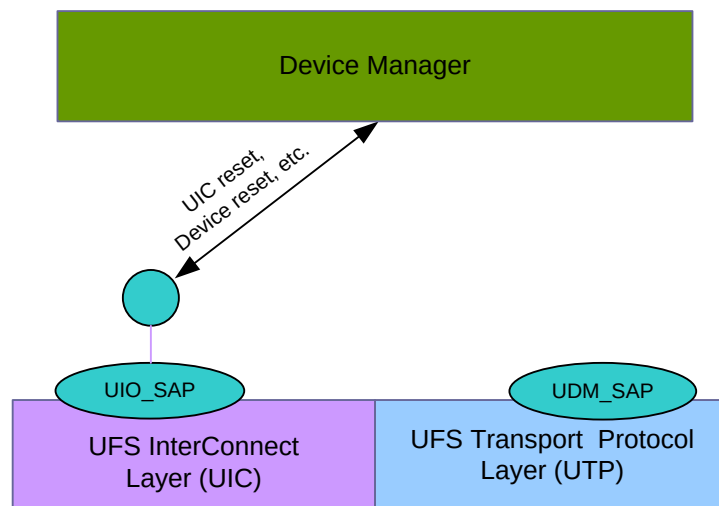


Figure 5.3 — Usage of UIO_SAP

5.1.4 UIO_SAP

UIO_SAP is the service access point exposed by the UIC layer. In UniPro, UIO_SAP corresponds to DME_SAP. The DME_SAP provides service primitives including one for resetting the entire UniPro protocol stack and one for UFS device reset, etc.

- DME_RESET : It is used when the UniPro stack has to be reset.
- DME_ENDPOINTRESET: It is used when UFS host wants the UFS device to perform a reset.

For the detailed internal mechanism, refer the UniPro specification [MIPI-UniPro] released by MIPI (MIPI is Mobile Industry Processor Interface).

5.1.5 UDM_SAP

UDM_SAP is the service access point exposed by the UTP layer to the Device Manager for UFS device level functions. UDM_SAP corresponds to the Query Request and Query Response functions defined by the UFS UTP layer.

For further details refer to the following subclauses: 10.9.9, Query Function transport protocol services, 10.7.8, QUERY REQUEST UPIU, and 10.7.9, QUERY RESPONSE UPIU.

5.1.6 UFS Transport Protocol Layer

The UFS Transport Protocol (UTP) layer provides services for the higher layer . UPIU is “UFS Protocol Information Unit” which is exchanged between UTP layers of UFS host and UFS device. For example, if host side UTP receives the request from application layer or Device Manager, UTP generates a UPIU for that request and transports the generated UPIU to the peer UTP in UFS device side. The UTP layer provides the following three access points.

- 1) UFS Device Manager Service Access Point (UDM_SAP) to perform the device level management like descriptor access.
- 2) UTP Command Service Access Point (UTP_CMD_SAP) to transport commands.
- 3) UTP Task Management Service Access Point (UTP_TM_SAP) to transport task-management function like “abort task” function.

5.1.7 UFS Interconnect Layer

The lowest layer is UFS Interconnect Layer (UIC) which handles connection between UFS host and UFS device. UIC consists of MIPI UniPro and MIPI M-PHY. The UIC provides two service access points to upper layer. The UIC Service Access Point (UIC_SAP) to transport UPIU between UFS host and UFS device. The UIC_SAP corresponds to T_SAP in UniPro. The UIC IO control Service Access Point (UIO_SAP) to manage UIC. The UIO_SAP corresponds to DME_SAP in UniPro.

5.1.8 UFS Topology

This version of the standard assumes that only one device is connected to a UFS port. Other topologies may be defined in future versions of the standard.

5.2 UFS System Model

Figure 5.4 shows an example of UFS system. It shows how a UFS host is connected to a UFS device, the position of UFS host controller and its related UFS HCI interface.

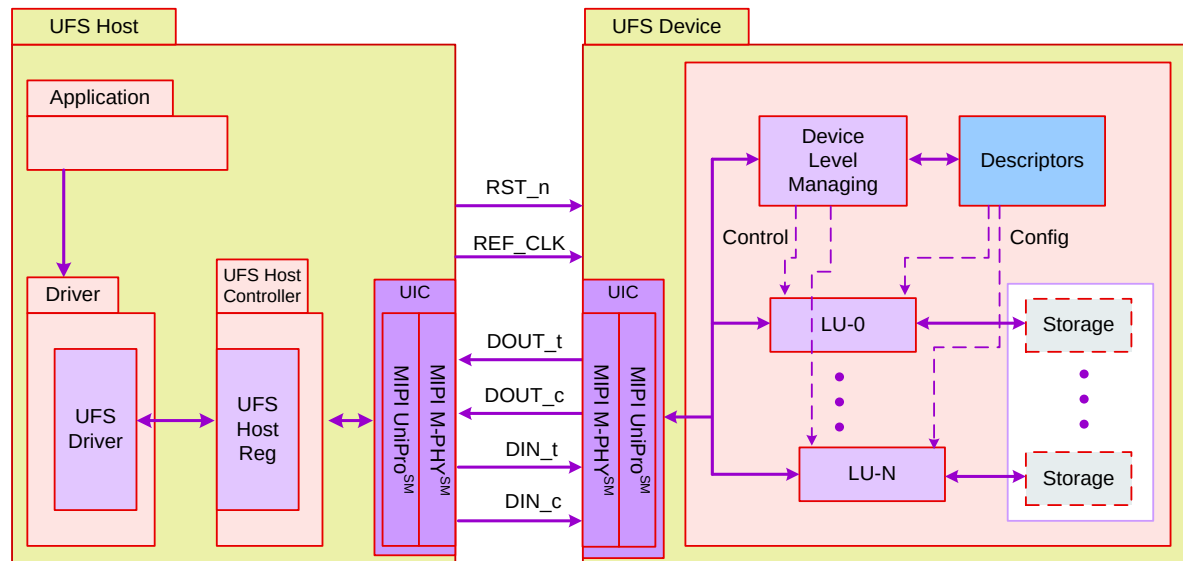


Figure 5.4 — UFS System Model

The UFS host consists of the application which wishes to communicate with the UFS device. It communicates with the device using the UFS driver. The UFS driver is meant for managing the UFS host controller through the UFS HCI (UFS Host Controller Interface). The UFS HCI is basically a set of registers exposed by the host controller.

Figure 5.4 also indicates the UFS interface between the UFS host and the UFS device. The UFS Interconnect (UIC) Layer consists of MIPI UniPro and MIPI M-PHY. The physical layer M-PHY is differential, dual simplex PHY that includes TX and RX pairs.

Potential UFS devices can be memory card (full size and micro size), embedded bootable mass storage devices, IO devices, etc. A UFS device is comprised of multiple logical units, a device manager and descriptors. The device manager performs device level functions like power management while the logical unit performs functions like read, write etc. The descriptors are meant for storage of configuration related information.

5.3 System Boot and Enumeration

The system boot from a bootable UFS device will initiate after power up when the UFS InterConnect Layer (MIPI M-PHY and UniPro) has completed its boot sequence. The boot code can be read from the appropriate boot logical unit, or as desired, boot ROM code can re-configure MIPI M-PHY and UniPro to appropriate setting before reading the boot code.

Multiple boot logical units may be available in a UFS device. However, only one boot logical unit will be active at power-up. Appropriate descriptors are available to configure the boot process.

During boot, accesses to boot logical unit are supported via SCSI commands.

5.4 UFS Interconnect (UIC) Layer

UFS interconnect layer is composed by MIPI UniPro, which provides basic transfer capabilities to the upper layer (UTP), and MIPI M-PHY, adopted as UFS physical layer.

5.4.1 UFS Physical Layer Signals

The UFS physical layer defines the physical portion of the UFS interface that connects UFS device and UFS host. This is based on MIPI M-PHY specification. UFS interface can support multiple lanes in each direction. Each lane consists of a differential pair. Basic configuration is based on one transmit lane and one receive lane.

Optionally, a UFS device may support two downstream lanes and two upstream lanes. An equal number of downstream and upstream lanes shall be provided in each link.

Table 5.1 summarizes the signals required for a UFS device. Only the single lane, per direction, per link, configuration is shown. See clause 6, UFS Electrical: Clock, Reset, Signals And Supplies, and clause 8, UFS UIC Layer: MIPI M-PHY, for full details about UFS signals.

Table 5.1 — UFS Signals

Name	Type	Description
REF_CLK	Input	Reference clock Relatively low speed clock common to all UFS devices in the chain, used as a reference for the PLL in each device.
DIN_t DIN_c	Input	Downstream lane input Differential input true and complement signal pair.
DOUT_t DOUT_c	Output	Upstream lane output Differential output true and complement signal pair.
RST_n	Input	Reset UFS Device hardware reset signal

5.4.2 MIPI UniPro

In UFS, UniPro is responsible for management of the link, including the PHY.

NOTE Device management is outside the scope of the interconnect layer and is the responsibility of the upper layers.

The basic interface to the interconnect layer is UniPro definition of a CPort. CPort is used for all data transfer as well as all control and configuration messages. In general, multiple CPorts can be supported on a device and the number of CPorts is implementation dependent.

Traffic sent over UniPro link can be classified as TC0 or TC1 traffic class with TC1 as higher priority traffic class. This version of UFS standard only uses a single CPort and TC0 traffic class.

UFS takes advantage of the basic types of UniPro services. These include data transfer service, and config/control/status service.

For more details, please refer to clause 9, UIC layer: MIPI UniPro, and MIPI UniPro specification [MIPI-UniPro].

5.4 UFS Interconnect (UIC) Layer (cont'd)

5.4.3 MIPI UniPro Related Attributes

In general the UniPro related attributes, values and use of them are defined in the MIPI UniPro specification. The attributes may be generic for all UniPro applications and thus out of scope of this document. Following attributes are defined in this standard specifically for UFS application as indicated in Table 5.2.

Table 5.2 — ManufacturerID and DeviceClass Attributes

Attribute	AttributeID ⁽¹⁾	Value	Description
DME_DDBL1_ManufacturerID	0x5003		MIPI manufacturer ID. MIPI MID shall be used in this Attribute also for UFS applications. The ID can be requested from MIPI.
DME_DDBL1_DeviceClass	0x5002	Memory = 0x02 Host = 0x03	UniPro DeviceClass ID for UFS application
NOTE 1 Reference MIPI Alliance Specification for Device Descriptor Block [MIPI-DDB]			

5.5 UFS Transport Protocol (UTP) Layer

As mentioned previously, the Transport Layer is responsible for encapsulating the protocol into the appropriate frame structure for the interconnect layer. UFS is protocol agnostic and thus any protocol will need the appropriate translation layer. For this version of UFS standard, this is UTP (UFS Transport Protocol) layer.

In this version of the standard, all accesses are supported only through SCSI, however additional API/service/extension may be added in future versions to introduce new features or address specific requirements.

A design feature of UTP is to provide a flexible packet architecture that will assist the UFS controller in directing the encapsulated command, data and status packets into and out of system memory. The intention is to allow the rapid transmittal of data between the host system memory and the UFS device with minimal host processor intervention. Once the data structures are set up in host memory and the target device is notified, the entire commanded transaction can take place between the UFS device and the host memory. The means by which the UFS controller transfers data into and out of host memory is via a hardware and/or firmware mechanism that is beyond the scope of this document. See the UFS controller standard for further information.

A second feature of the UTP design is that once a device receives a command request notification from the host, the device will control the pacing and state transitions needed to satisfy the data transfers and status completion of the request. The idea here is that the device knows its internal condition and state and when and how to best transfer the data that makes up the request. It is not necessary for the host system or controller to continuously poll the device for “ready” status or for the host to estimate when to start a packet transfer. The device will start the bus transactions when it determines its conditions and status are optimal. This approach cuts down on the firmware and logic needed within the host to communicate with a device. It also affords the maximum possible throughput with the minimum number of bus transactions needed to complete the operation.

5.5 UFS Transport Protocol (UTP) Layer (cont'd)

5.5.1 Architectural Model

The SCSI Architecture Model [SAM] is used as the general architectural model for UTP. The SAM architecture is a client-server model or more commonly a request-response architecture.

5.5.1.1 Client-Server Model

A client-server transaction is represented as a procedure call with inputs supplied by the application client on the Initiator device. The procedure call is processed by the server and returns outputs and a procedure call status. Client-server relationships are not symmetrical. A client only originates requests for service. A server only responds to such requests.

Initiator device and Target device are mapped into UFS physical network devices. An Initiator device may request processing of a command or a task management function through a request directed to the Target device. Device service requests are used to request the processing of commands and task manager requests are used to request the processing of task management functions.

Target device is a UFS device. A UFS device will contain one or more Logical Units. A Logical Unit is an independent processing entity within the device.

An Initiator request is directed to a single Logical Unit within a device. A Logical Unit will receive and process the client command or request. Each Logical Unit has an address within the Target device called a Logical Unit Number (LUN).

Communication between the Initiator device and Target device is divided into a series of messages. These messages are formatted into UFS Protocol Information Units (UPIU) as defined within this standard. There are a number of different UPIU types defined. All UPIU structures contain a common header area at the beginning of the data structure (lowest address). The remaining fields of the structure vary according to the type of UPIU.

A Task is a command or sequence of actions that perform a requested service. A Logical Unit contains a task queue that will support the processing of one or more Tasks. The Task queue is managed by the Logical Unit. A unique Task Tag is generated by the Initiator device when building the Task. This Task Tag is used by the Target device and the Initiator device to distinguish between multiple Tasks. All transactions and sequences associated with a particular Task will contain that Task Tag in the transaction associated data structures.

Command structures consist of Command Descriptor Blocks (CDB) that contain a command opcode and related parameters, flags and attributes. The description of the CDB content and structure are defined in detail in the [SAM], [SBC] and [SPC] INCITS T10 draft standards.

5.5.1.1 Client-Server Model (cont'd)

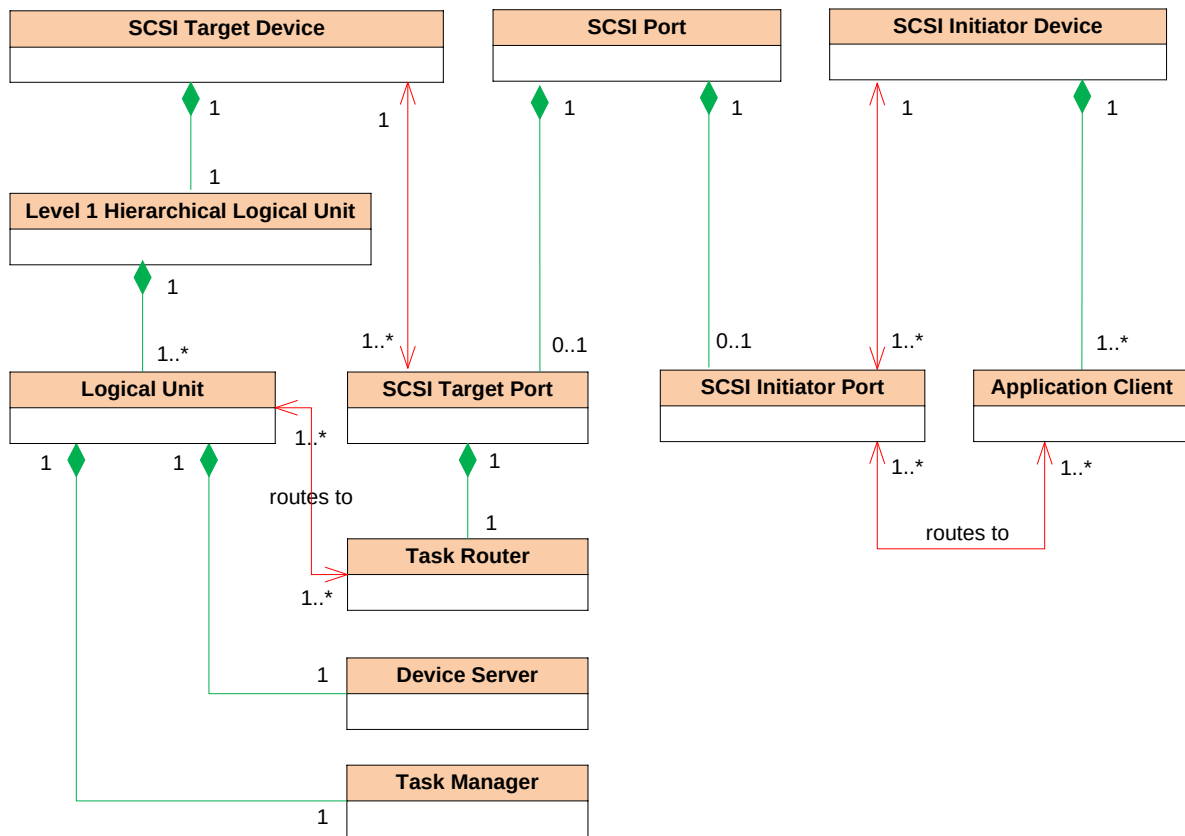


Figure 5.5 — SCSI Domain Class Diagram

5.5.1.1 Client-Server Model (cont'd)

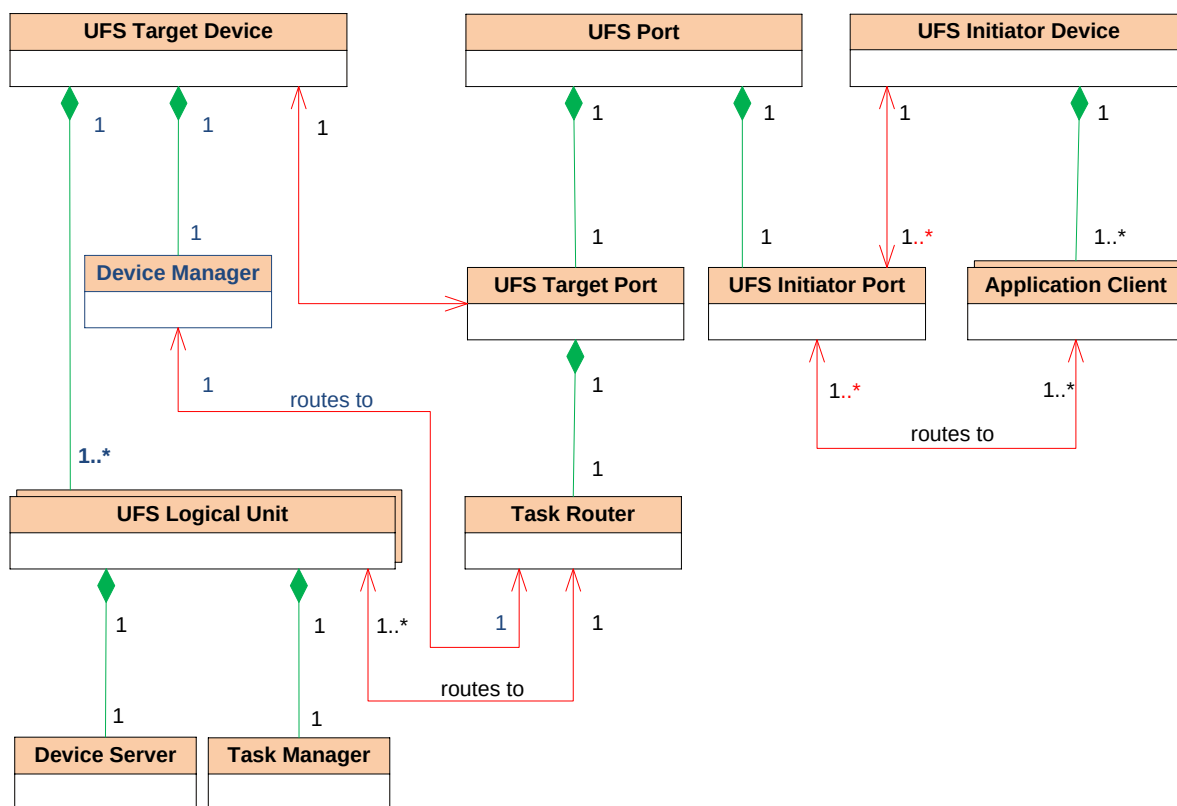


Figure 5.6 — UFS Domain Class Diagram

5.5.1.2 CDB, Status, Task Management

UTP adopts Command Descriptor Block (CDB) format for commands, device status data hierarchy and reporting method, and task management functions of outstanding commands, described in [SAM]. Regardless of the command protocol to be delivered by UTP, SCSI CDB, Status and Task Management Functions should be adopted uniformly in UFS devices.

5.5.1.3 Nexus

The nexus represents the relationship among the Initiator, Target, Logical Unit and Command (Task)

- Nexus notation: I_T_L_Q nexus; where I = Initiator, T = Target, L = Logical Unit, Q = Command

There shall be at least one initiator device in the UFS definition. There shall only one target device, the UFS device. There shall be one or more logical units in a UFS device. The command identifier (i.e., the Q in an I_T_L_Q nexus) is assigned by the Initiator device to uniquely identify one command in the context of a particular I_T_L nexus, allowing more than one command to be outstanding for the I_T_L nexus at the same time.

An overlapped command occurs when a task manager or a task router detects the use of a duplicate I_T_L_Q nexus in a command before that I_T_L_Q nexus completes its command lifetime (see [SAM]).

Concurrent overlapped commands are not allowed in UFS. Each command shall have a unique Task Tag. The UFS device is not required to detect overlapped commands.

5.5.1.4 SCSI Command Model

All command requests originate from application clients in an Initiator device. An application Client requests the processing of a command with the following procedure call:

- Service Response = Execute Command (IN (I_T_L_Q Nexus, CDB, Task Attribute, [Data-In Buffer Size], [Data-Out Buffer], [Data-Out Buffer Size], [CRN], [Command Priority]), OUT ([Data-In Buffer], [Sense Data], [Sense Data Length], Status, [Status Qualifier]))

Parameter fields in the UTP Command, Response, Ready-to-Transfer, Data-Out, Data-In UPIU headers contain the requisite information for the input and output arguments of the Execute Command procedure call in compliance with [SAM].

5.5.1.5 SCSI Task Management Functions

An application client requests the processing of a task management function with the following procedure call:

- Service Response = Function name (IN (Nexus), OUT ([Additional Response Information]))

Parameters fields in the UTP Task Management Request and UTP Task Management Response headers contain the requisite information for the input and output arguments of the Task Management Function procedure call in compliance with [SAM].

5.6 UFS Application and Command Layer

UFS interface is designed to be protocol agnostic interface. However, as mentioned previously, SCSI has been selected as the baseline protocol layer for this standard. Descriptors are available to identify and select the appropriate protocol for UFS interface.

The primary functions of the Command Set Layer are to establish a method of data exchange between the UFS host and UFS device and to provide fundamental device management capability. SCSI SBC and SPC commands are the baseline for UFS. UFS will not modify the SBC and SPC Compliant commands. The goal is to maximize re-use and leverage of the software codebase available on platforms (PC, netbook, MID) that are already supporting SCSI.

Options are available to define UFS Native commands and extension as needed.

UFS SCSI command set includes:

1. SBC compliant commands [SBC]:
 - FORMAT UNIT
 - READ (6) and READ (10)
 - READ CAPACITY (10)
 - REQUEST SENSE
 - SEND DIAGNOSTIC
 - UNMAP
 - WRITE (6) and WRITE (10)
2. SPC compliant commands [SPC]:
 - INQUIRY
 - REPORT LUNS
 - READ BUFFER
 - TEST UNIT READY
 - WRITE BUFFER
 - SECURITY PROTOCOL IN and SECURITY PROTOCOL OUT
3. SCSI operational commands for UFS applications and compatible with existing SCSI driver
 - MODE SELECT (10) and MODE SENSE (10)
 - PRE-FETCH (10)
 - START STOP UNIT
 - SYNCHRONIZE CACHE (10)
 - VERIFY (10)
4. Value-added optional commands for UFS:
 - READ (16), WRITE(16), PRE-FETCH (16), SYNCHRONIZE CACHE (16), and READ CAPACITY(16).

NOTE These commands support logical units with larger capacities having an 8-byte LBA field.

Refer to clause 11, UFS Application (UAP) Layer – SCSI Commands, for more details.

5.7 Mechanical

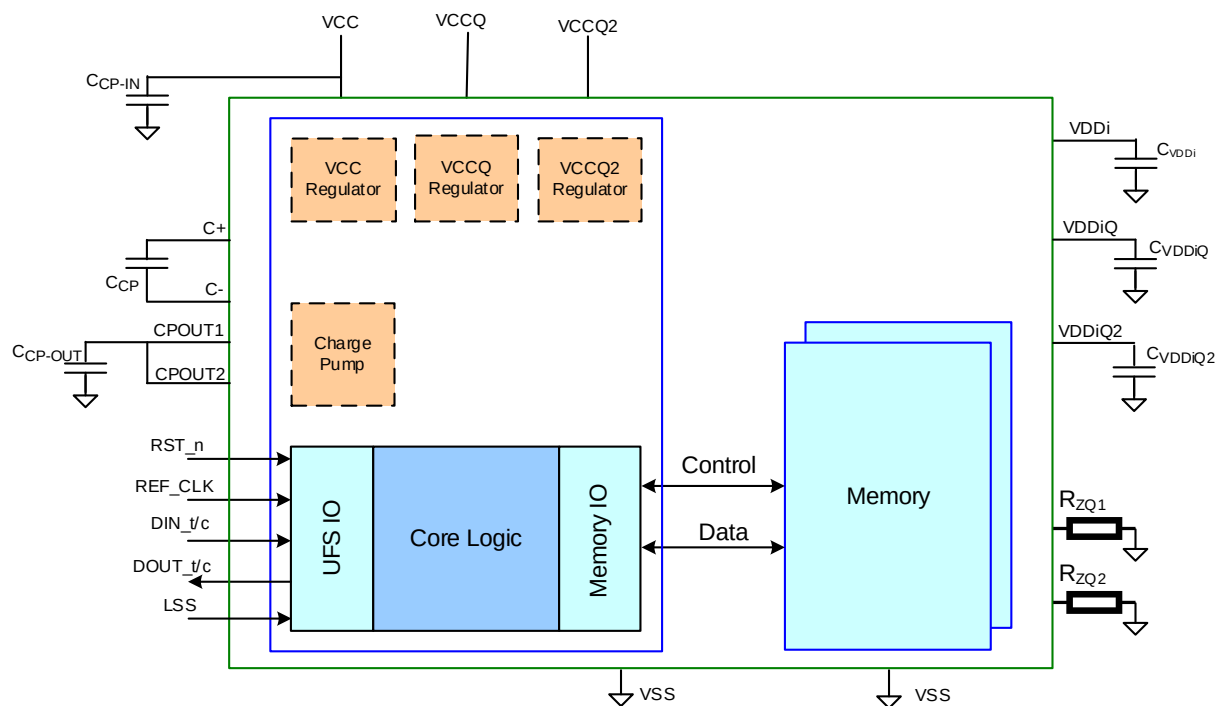
Packaging and requirements for UFS embedded device should adhere to the following guidelines if possible

- Reset and data transfer pins should be located in the second (PoP) or third row (MCP) in from the side of the package to prevent access.

6 UFS Electrical: Clock, Reset, Signals And Supplies

6.1 UFS Signals

Figure 6.1 represents a conceptual drawing of UFS device. Utilization of internal regulators and connection of those to different parts of the sub-system may differ per implementation.



NOTE 1 The memory core power supply may be connected to VCC power supply ball, or the VCC regulator output, while it is connected to the charge pump output if VCC = 1.8 V and the memory requires 2.5 V core power supply.

NOTE 2 The memory IO may consume power from any power supply: VCC, VCCQ or VCCQ2.

NOTE 3 CCP-IN, C_{CP} and C_{CP-OUT} may be required only when internal charge pump is used.

Figure 6.1 — Conceptual UFS Device Block Diagram

6.1 UFS Signals (cont'd)

Table 6.1 — Signal Name and Definitions

Name	Type	Description
VCC	Supply	Supply voltage for the memory devices
VCCQ	Supply	Supply voltage used typically for the memory controller and optionally for the PHY interface, the memory IO, and any other internal very low voltage block
VCCQ2	Supply	Supply voltage used typically for the PHY interface and the memory controller and any other internal low voltage block
VDDiQ ⁽¹⁾	Input	Input terminal to provided bypass capacitor for VCCQ internal regulator
VDDiQ2 ⁽¹⁾	Input	Input terminal to provide bypass capacitor for VCCQ2 internal regulator
VDDi ⁽¹⁾	Input	Input terminal to provide bypass capacitor for VCC internal regulator
VSS	Supply	Ground
RST_n	Input	Input hardware reset signal. This is an active low signal
REF_CLK	Input	Input reference clock. When not active, this signal should be pull-down or driven low by the host SoC.
Differential input signals into UFS device from the host		
DIN_t or DIN0_t ⁽²⁾ DIN_c or DIN0_c ⁽²⁾	Input	Downstream data lane 0. DIN_t is the positive node of the differential signal.
DIN1_t ⁽²⁾ , DIN1_c ⁽²⁾	Input	Downstream data lane 1.
Differential output signals from the UFS device to the host		
DOUT_t or DOUT0_t ⁽³⁾ DOUT_c or DOUT0_c ⁽³⁾	Output	Upstream data lane 0. DOUT_t is the positive node of the differential signal.
DOUT1_t ⁽³⁾ , DOUT1_c ⁽³⁾	Output	Upstream data lane 1.
C+	Input	Optional charge pump capacitor, positive terminal. For more information, please refer to 6.5
C-	Input	Optional charge pump capacitor, negative terminal. For more information, please refer to 6.5
CPOUT1, CPOUT2	Input	Optional Charge pump output capacitor terminal. For more information, please refer to 6.5

6.1 UFS Signals (cont'd)

Table 6.2 — Signal Name and Definitions (cont'd)

Name	Type	Description
LSS ⁽⁴⁾	Input	Input Link Startup Sequence (LSS) mode. 0: low speed link startup sequence (LS-LSS) 1: high speed link startup sequence (HS-LSS) This pin may be directly driven by the host or connected to external pull-up or pull-down according to the selected Link Startup Sequence mode. This value is expected to be fixed before RST_n is high to determine the Link Startup mode.
RzQ1/RzQ2	Input	External Calibration Resistor (optional). Refer to datasheet to determine if resistors are needed.
NOTE 1 If there is no internal regulator requiring output capacitor then VDDi pins should be internally connected as follows: VDDi to VCC, VDDiQ to VCCQ, and VDDiQ2 to VCCQ2. NOTE 2 DIN0_t/_c and DIN1_t/_c apply if the device has two downstream lanes. NOTE 3 DOUT0_t/_c and DOUT1_t/_c apply if the device has two upstream lanes. NOTE 4 In this version of the standard, HS LSS mode support is optional, LS LSS mode support is mandatory. Refer to the vendor datasheet to determine if HS-LSS mode is supported or not. Based on the HS LSS supportability, system vendor could configure the LSS pin for the UFS system to be initialized in LS LSS mode or HS LSS mode.		

It is recommended to apply [HBM-MM] and [CDM] to all signals described in Table 6.1. See Table 6.2 for ESD specification. This standard does not require use of the Machine Model for ESD qualification.

Table 6.3 — Electrostatic Discharge Sensitivity Characteristics

Parameter	Symbol	Min	Max	Unit	Notes
Human body model (HBM)	ESD _{HBM}	1000	-	V	1
Charged-device model (CDM)	ESD _{CDM}	250	-	V	2
NOTE 1 Refer to ESDA/JEDEC Joint Standard JS-001-2017 for HBM measurement procedures NOTE 2 Refer to ESDA/JEDEC Joint Standard JS-002-2018 for CDM measurement procedures					

6.2 Reset Signal and LSS Signal

To meet the requirements of the JEDEC Standard [JESD8-12A], the RST_n and LSS signal voltages shall be within the specified ranges for VCCQ in Table 6.3.

Table 6.4 — Reset Signal and LSS Signal Electrical Parameters

Parameter	Symbol	Min	Max	Unit	Notes
Input HIGH voltage	VIH	0.65*VCCQ	VCCQ+0.3	V	For VCCQ as defined in Table 6.5
Input LOW voltage	VIL	VSS-0.3	0.35*VCCQ	V	For VCCQ as defined in Table 6.5
Input Capacitance	Cin		10	pF	
Input leakage Current	I _{lkg}		10	μA	

6.3 Power Supplies

Table 6.5 — UFS Power Supply Parameters

Parameter	Symbol	Min	Max	Unit	Notes
VCC DC operating range	VCC	2.4	2.7	V	3
VCCQ DC operating range	VCCQ	1.14	1.26	V	1, 3
VCCQ2 DC operating range	VCCQ2	1.70	1.95	V	3
Supply Voltage power up timing for 2.5 V	tPRUH		35	ms	2
Supply Voltage power up timing for 1.8 V	tPRUL		25	ms	2
Supply Voltage power up timing for 1.2 V	tPRUV		20	ms	2
VCC internal regulator capacitor	C _{VDDi}	1		μF	
VCCQ internal regulator capacitor	C _{VDDiQ}	1		μF	
VCCQ2 internal regulator capacitor	C _{VDDiQ2}	1		μF	
NOTE 1 See [JESD8-12A]. NOTE 2 Power up timing starts when the supply voltage crosses 300 mV and ends when it reaches the minimum operating value. NOTE 3 Depending on the vendor, valid power configuration may be defined in each UFS device vendor's data sheet. Refer to the vendor datasheet for the detail.					

6.3 Power Supplies (cont'd)

Figure 6.2 shows t_{PRUH} , t_{PRUL} and t_{PRUV} timings.

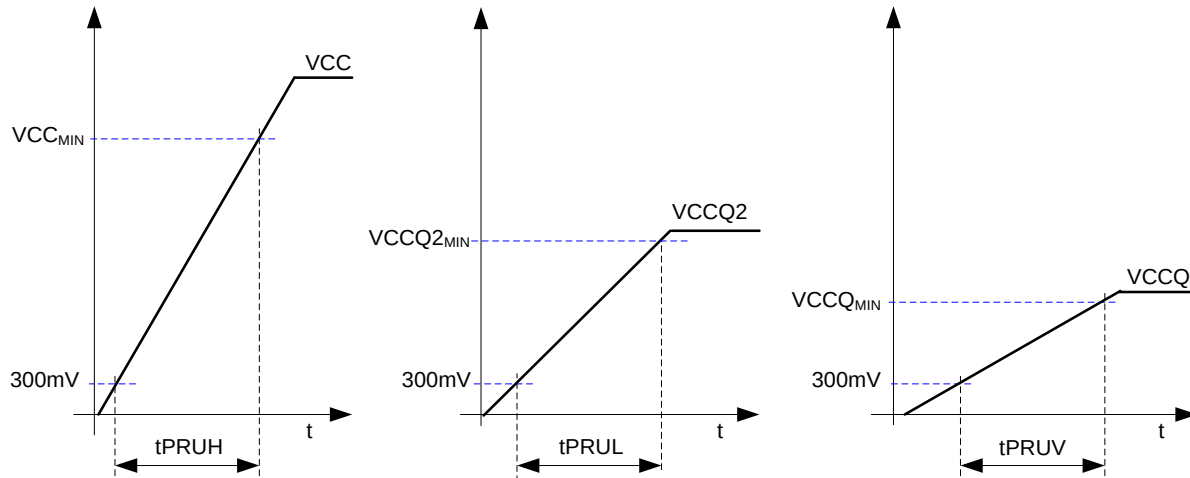


Figure 6.2 — Supply voltage power up timings

6.4 Reference Clock

The M-PHY specification defines the reference clock optional for the State Machine Type I [MIPI M-PHY]. As the PWM signaling is self-clocked the reference clock is not required for the data latching. Therefore, UFS devices shall be able to operate without reference clock in LS-MODE (SLEEP and PWM-BURST).

Still existence of the reference clock may be utilized to enable lower BER and faster HS-MODE PLL/DLL locking. Thus a UFS device shall implement a square wave single ended reference clock input and it requires the presence of a reference clock with the characteristics described in this section when operating in HS-MODE (STALL and HS-BURST). In order to avoid potential race conditions, it is recommended that such reference clock is already present when requesting a power mode change into Fast_Mode or FastAuto_Mode.

6.4 Reference Clock (cont'd)

Table 6.6 — Reference Clock parameters

Parameter	Symbol	Nominal		Unit	Notes
Frequency	f_{ref}	19.2, 26.0, 38.4, 52.0		MHz	1
Parameter	Symbol	Min	Max	Unit	Notes
Frequency Error	f_{ERROR}	-150	+150	ppm	
Input High Voltage	V_{IH}	$0.65 * V_{\text{CCQ}}$		V	2
Input Low Voltage	V_{IL}		$0.35 * V_{\text{CCQ}}$	V	2
Input Clock Rise Time	t_{RISE}		2	ns	3
Input Clock Fall Time	t_{FALL}		2	ns	3
Duty Cycle	t_{DC}	45	55	%	4
Random Jitter ($f_{\text{ref}} = 19.2$)	RJ_{RMS}		5.9	ps	5
Random Jitter ($f_{\text{ref}} = 26.0$)			4.6		
Random Jitter ($f_{\text{ref}} = 38.4$)			3.5		
Random Jitter ($f_{\text{ref}} = 52.0$)			2.8		
Deterministic Jitter	DJ_{dd}		15	ps	6
Input Impedance	RL_{RX}	100		k Ω	7
	CL_{RX}		5	pF	

NOTE 1 HS-BURST rates A and B are achieved with integer multipliers of f_{ref} .

NOTE 2 Figure 6.4 shows the input levels $V_{\text{IL,MAX}}$ to $V_{\text{IH,MIN}}$.

NOTE 3 Clock rise time and clock fall time shall be measured from 20% to 80% of the window defined by $V_{\text{IL,MAX}}$ to $V_{\text{IH,MIN}}$, see Figure 6.4.

NOTE 4 Clock duty cycle shall be measured at the crossings of the REF_CLK signal with the midpoint V_{MID} , defined as: $V_{\text{MID}} = (V_{\text{IL,MAX}} + V_{\text{IH,MIN}}) / 2$, see Figure 6.4.

NOTE 5 The RJ_{RMS} magnitudes are based on the phase noise parameters of previous versions of this standard and replace them. RJ_{RMS} is computed with the integration range from 50 kHz to the Nyquist frequency of the reference clock.

NOTE 6 Reference Clock frequencies of 19.2 MHz and 26.0 MHz cannot be used for HS-G5 and above, because the corresponding RJ_{RMS} values are beyond the allowable limit for HS-G5 operation. The frequency of the reference clock shall be 38.4 MHz or 52.0 MHz for HS-G5 and above operation.

NOTE 7 RL_{RX} and CL_{RX} include Rx package and Rx input impedance.

6.4 Reference Clock (cont'd)

bRefClkFreq attribute indicates to the device the frequency of the REF_CLK signal in LS LSS mode, and its default value corresponds to 52.0 MHz.

Table 6.6 provides an overview of the relationship between reference clock frequencies and HS operation.

Table 6.7 — Reference Clock Frequency and HS Operation

Reference Clock Frequency [MHz]	HS-G1	HS-G2	HS-G3	HS-G4	HS-G5
19.2	Y	Y	Y	Y	N
26.0	Y	Y	Y	Y	N
38.4	Y	Y	Y	Y	Y
52.0	Y	Y	Y	Y	Y

bRefClkFreq attribute can be written only if both sub-links are in LS-MODE. This attribute indicates the REF_CLK frequency. In HS-LSS mode, the UFS device shall detect the frequency of the REF_CLK signal.

A stable REF_CLK signal is expected when RST_n signal goes high. REF_CLK is not expected to change in HS-Mode. See Annex C for details.

The reference clock is not required and it may be turned off when both SUB-LINKs have reached and are operating in one of the following M-PHY states:

- LS-MODE (SLEEP or PWM-BURST state)
- HIBERN8 state

If power mode change from HS-MODE to LS-MODE or HIBERN8 is initiated by UFS Host, then it shall be ensured that at least the minimum time duration defined by bRefClkGatingWaitTime has elapsed before turning off the reference clock, see

Figure 6.3.

6.4 Reference Clock (cont'd)

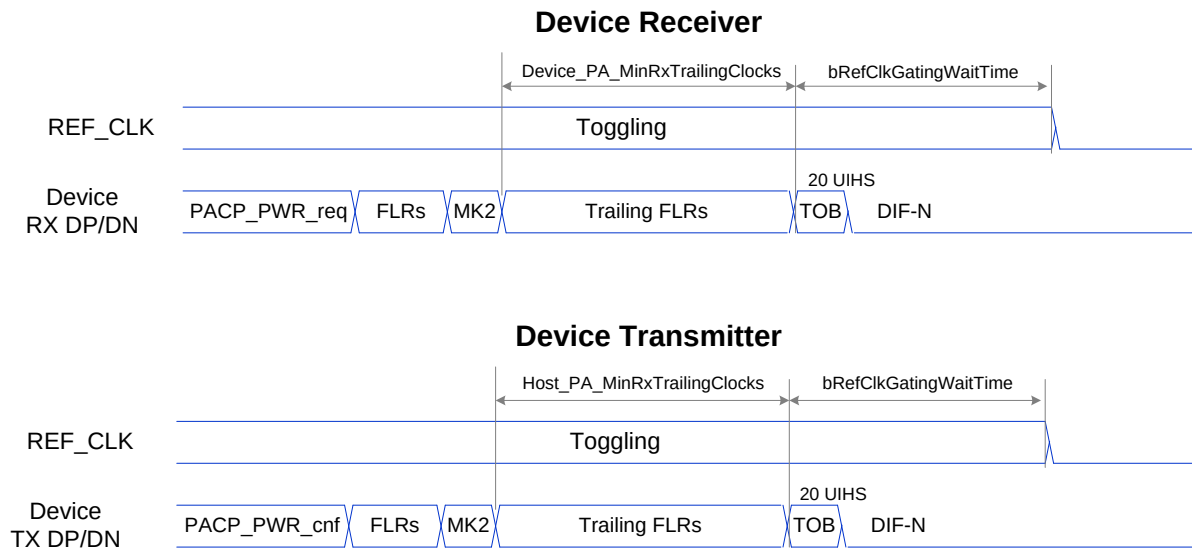


Figure 6.3 — bRefClkGatingWaitTime

UFS Host may start a timer when DME_POWERMODE.ind is received for HS-MODE to LS-MODE transition or DME_HIBERNATE_ENTER.ind is received for HS-MODE to HIBERN8 transition. In addition to bRefClkGatingWaitTime, Device PA_MinRxTrailingClocks and Host PA_MinRxTrailingClocks should be considered to determine when the reference clock may be stopped. In case of a transition from HS-MODE to HIBERN8 to reach the UFS-DeepSleep power mode, this constraint for turning off the reference clock shall be applied as well.

Unipro layer defines re-initialization process using PA_INIT mechanism. See [MIPI-Unipro] for details. During this process both the sub-links may briefly enter LS-MODE before returning to HS-MODE. The reference clock shall not be gated during the entire PA_INIT procedure.

The reference clock shall be turned ON and stably running before initiation of the state transition to STALL from a SLEEP or HIBERN8 state. Reference Clock shall not be gated in HS modes.

6.4 Reference Clock (cont'd)

Figure 6.4 shows clock rise time and fall time measurements.

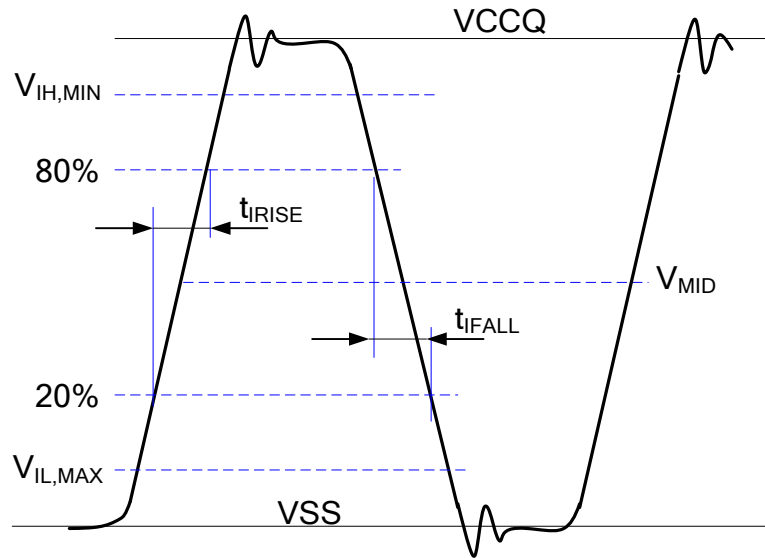


Figure 6.4 — Clock input levels, rise time, and fall time

6.4.1 HS Gear Rates

Table 6.8 defines the data rate values for the two rate series with respect REF_CLK frequency value (f_{ref}).

Table 6.8 — HS-BURST Rates

HS-GEAR	Rate A-series	Rate B-series		Unit
	f_{ref}	f_{ref}		
	19.2 / 26.0 / 38.4 / 52.0	19.2 / 38.4	26.0 / 52.0	MHz
HS-GEAR1	1248 ⁽²⁾	1459.2	1456.0	Mbps
HS-GEAR2	2496	2918.4	2912.0	Mbps
HS-GEAR3	4992	5836.8	5824.0	Mbps
HS-GEAR4	9984	11673.6	11648.0	Mbps
HS-GEAR5	19968	23347.2	23296.0	Mbps
NOTE 1 “Mbps” indicates 1000000 bit per sec.				
NOTE 2 1248 Mbps with $f_{ref} = 38.4$ MHz may be obtained using a prescaler. $f_{ref} * M / P$, $M = 65$ (PLL multiplier), $P = 2$ (Prescaler).				

6.4 Reference Clock (cont'd)

6.4.2 Host Controller requirements for reference clock generation

Table 6.9 — Host controller reference clock parameters

Parameter	Symbol	Min	Max	Unit	Notes
DC Output High Voltage	V_{OH}	$0.75 * V_{CCQ}$		V	1, 2
DC Output Low Voltage	V_{OL}		$0.25 * V_{CCQ}$	V	1, 2
Output Clock Rise Time	t_{ORISE}		2	ns	3
Output Clock Fall Time	t_{OFALL}		2	ns	3
Test Load Impedance	R_{LTest}	100		k Ω	1, 4, 5
	C_{LTest}	20		pF	1, 4, 5

NOTE 1 Output load resistive and capacitance component are defined as 20 pF shunted by 100 k Ω .

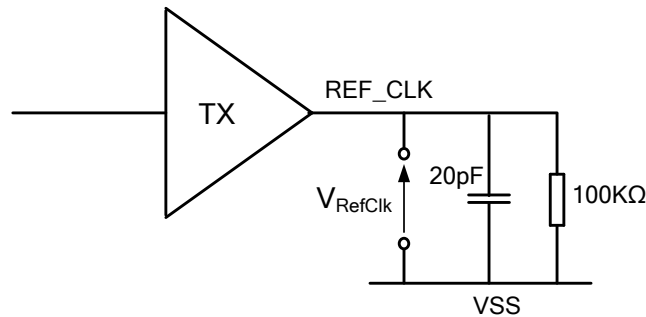


Figure 6.5 — Test Load Impedance

NOTE 2 Figure 6.6 shows Output driver and Input receiver levels. REF_CLK driver AC voltage (e.g., ring back) shall be kept inside Output Voltage limit.

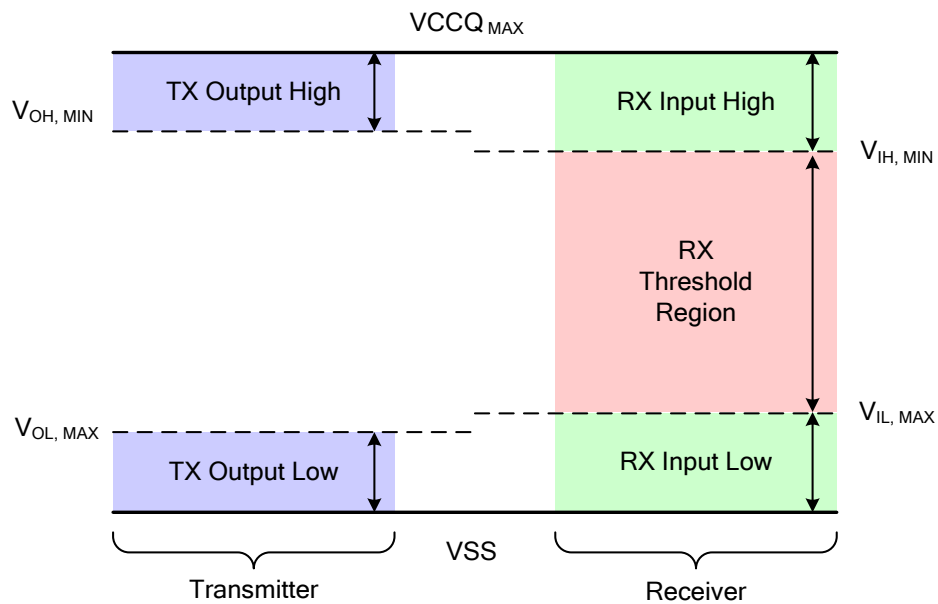


Figure 6.6 — Output driver and Input receiver levels

6.4.2 Host Controller requirements for reference clock generation (cont'd)

Table 6.9 — Host controller reference clock parameters (cont'd)

NOTE 3 Clock rise time and clock fall time shall be measured from 20% to 80% of the window defined by $V_{OL,MAX}$ and $V_{OH,MIN}$.

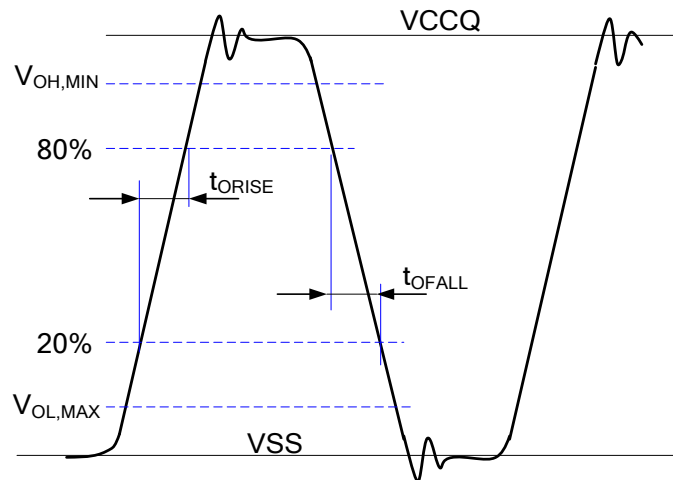


Figure 6.7 — Clock output levels, rise time and fall time

NOTE 4 Including transmitter output, Tx package, interconnect, Rx package and Rx input impedance

NOTE 5 The Test Load Impedance is placed at the output of the driver, with the shortest interconnect.

6.5 External Charge Pump Capacitors (Optional)

In order to produce memory devices that can accommodate a low voltage core supply ($VCC=1.8\text{ V}$) an internal charge pump circuit may be required.

Charge pump circuit requires extra-sized passive components. An optional usage of external charge pump capacitors is provided.

Figure 6.1 shows the electrical connections required in case of charge pump implementation that uses external capacitors.

Table 6.9 provides description of the capacitors to be used.

Table 6.10 — Charge pump capacitors description

Capacitor Name	Min	Typ	Max	Description
C_{CP-IN}	TBD	4.7uF	TBD	When charge pump is used, this capacitor is used as the charge pump input bypass capacitor. When charge pump is not used this capacitor is used as a bypass capacitor for the memory.
C_{CP-OUT}	TBD	4.7uF	TBD	Charge pump output bypass capacitor
C_{CP}	TBD	0.22uF	TBD	Charge pump flying capacitor

The charge pump capacitors are optional for UFS devices. Table 6.11 specifies name and description of the balls for connecting external charge pump capacitors.

6.5 External Charge Pump Capacitors (Optional) (cont'd)

Table 6.11 — Charge pump related ball names

Ball Name	Description
C+	C _{CP} capacitor's positive terminal
C-	C _{CP} capacitor's negative terminal
CPOUT1, CPOUT2	Charge pump output capacitor (2 balls) ¹
NOTE 1 Two CPOUT balls are required to reduce inductance, improve ripple and transient response	
NOTE 2 The given capacitors shall be placed close to the memory device to minimize the inductance. As a guideline for package design, it is recommended to place the CP related balls close to each other and close to the edge of the package.	

6.6 ZQ Calibration Resistor (optional)

In order to produce memory devices that can accommodate high performance, ZQ calibration resistors may be required.

A ZQ calibration resistor requires a passive component. An optional usage of external ZQ calibration resistors is provided.

Figure 6.1 shows an example of the electrical connections needed if the ZQ calibration implementation uses external resistors.

Table 6.11 provides a description of the resistors to be used.

Table 6.12 — ZQ Calibration Resistors Description

Resistor Name	Type	Nominal	Accuracy	Description
R _{ZQ1}	Input	Vendor specific	1%	Optional resistor for device internal calibration
R _{ZQ2}	Input	300ohm	1%	Optional resistor for NAND interface's calibration