JEDEC STANDARD

Universal Flash Storage (UFS) Card Extension

Version 3.0

JESD220-2B

(Revision of JESD220-2A, January 2018)

NOVEMBER 2020

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.

Special Legal Disclaimer: JEDEC has received information that certain patents or patent applications may be essential to this standard. However, as of the publication date of this standard, no statements regarding an assurance to license such patents or patent applications have been provided. JEDEC does not make any determination as to the validity or relevancy of such patents or patent applications. Anyone making use of the standard assumes all liability resulting from such use. JEDEC and its members disclaim any representation or warranty, express or implied, relating to the standard and its use.

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 2020
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2107

PRICE: Contact JEDEC

UNIVERSAL FLASH STORAGE (UFS) CARD EXTENSION, Version 3.0

CONTENTS

CONTENTS (cont'd)

Figures

Tables

UNIVERSAL FLASH STORAGE (UFS) CARD EXTENSION, Version 3.0

Foreword

This standard has been prepared by JEDEC. The purpose of this standard is to define a UFS card specification. This document will be extension of the UFS Standard, JESD220.

Introduction

The UFS device (embedded/removable) is a universal data storage and communication media. It is designed to cover a wide area of applications as smart phones, VR(virtual reality) device, AR(augmented reality) device, Drone, 3D games, surveillance system, cameras, organizers, PDAs, digital recorders, MP3 players, internet tablets, electronic toys, etc.

UNIVERSAL FLASH STORAGE (UFS) CARD EXTENSION, Version 3.0

(From JEDEC Board Ballot JCB-20-39, formulated under the cognizance of the JC-64.1 Subcommittee on Electrical Specifications and Command Protocols.)

1 Scope

This standard specifies the characteristics of the UFS card electrical interface and the memory device. This document defines the added/modified features in UFS card compared to embedded UFS device. For other common features JESD220D, UFS, Version 3.0, will be referenced.

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 listed. For undated references, the latest edition of the normative document referred to applies.

[MIPI-M-PHY], MIPI Alliance Specification for M-PHY^{SM} Specification, Version 4.1

[MIPI-UniPro], MIPI Alliance Specification for Unified Protocol (UniPro^{SM}), Version 1.8

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

[SAM], SCSI Architecture Model – 5 (SAM–5), Revision 05, 19 May 2010

[SPC], T10 Specification: SCSI Primary Commands – 4 (SPC-4), Revision 27, 11 October 2010

[SBC], T10 Specification: SCSI Block Commands – 3 (SBC–3), Revision 24, 05 August 2010

[UFS], JEDEC JESD220D, Universal Flash Storage (UFS), Version 3.0

[UFS Card MO], JEDEC JEP95, MO-320A, UFS Card Form Factor

3 Terms and Definitions

For the purpose of this standard, the terms and definitions given in the documents included in section 2 “Normative Reference” and the following apply.

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

device: An addressable device on the UFS bus usually a target that contains at least one LUN.

host: An addressable device on the UFS bus which is usually the main CPU that hosts the UFS bus.

3 Terms and definitions (cont'd)

3.1 Acronyms

HCI	Host Controller Interface
UFS	Universal Flash Storage
MIPI	Mobile Industry Processor Interface
PWM	Pulse Width Modulation
RPMB	Replay Protected Memory Block
SBC	SCSI Block Commands
SPC	SCSI Primary Commands
LUN	Logical Unit Number
NA	Not applicable
KB	Kilobyte
eUFS	Embedded Universal Flash Storage

3.2 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.

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.

3.2 Keywords (cont'd)

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.

3.3 Abbreviations

etc. - And so forth (Latin: et cetera)
e.g. - For example (Latin: exempli gratia)
i.e. - That is (Latin: id est)

3.4 Conventions

UFS specification 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 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).

4 Introduction

4.1 Overview

The features of the UFS card are a subset of the features of the JESD220 standard. The UFS card uses the same protocol as the embedded UFS device, but it has the removable card specific requirements like maximum power consumption are added.

4.2 Functional Features

UFS card functional features are a subset of the features of the UFS embedded device. These include:

- Support for MIPI M-PHY PWM-GEAR1. HS-GEAR1, HS-Gear2, HS-Gear3 and HS-GEAR4
- Support for Multiple partitions (LUNs) with partition Management (Note that since RPMB is not supported in UFS Card, RPMB well known logical unit is not supported.)
- Support for Multiple User Data Partition with Enhanced User Data Area options
- Reliable write operation
- Background operations
- Secure operations, Purge and Erase to enhance data security
- Write Protection options, including Permanent and Power-On Write Protection
- Task management operations
- Power management operations
- UFS SCSI Command Set
- Boot Support Operation

5 UFS Card System Architecture

5.1 Overview

The UFS card will use the same protocol as the embedded UFS device. There will not be any change in the overall system architecture of the removable UFS card compared to the embedded card.

5.2 UFS Card Signals

Figure 5.1 shows the conceptual drawing of the UFS card.

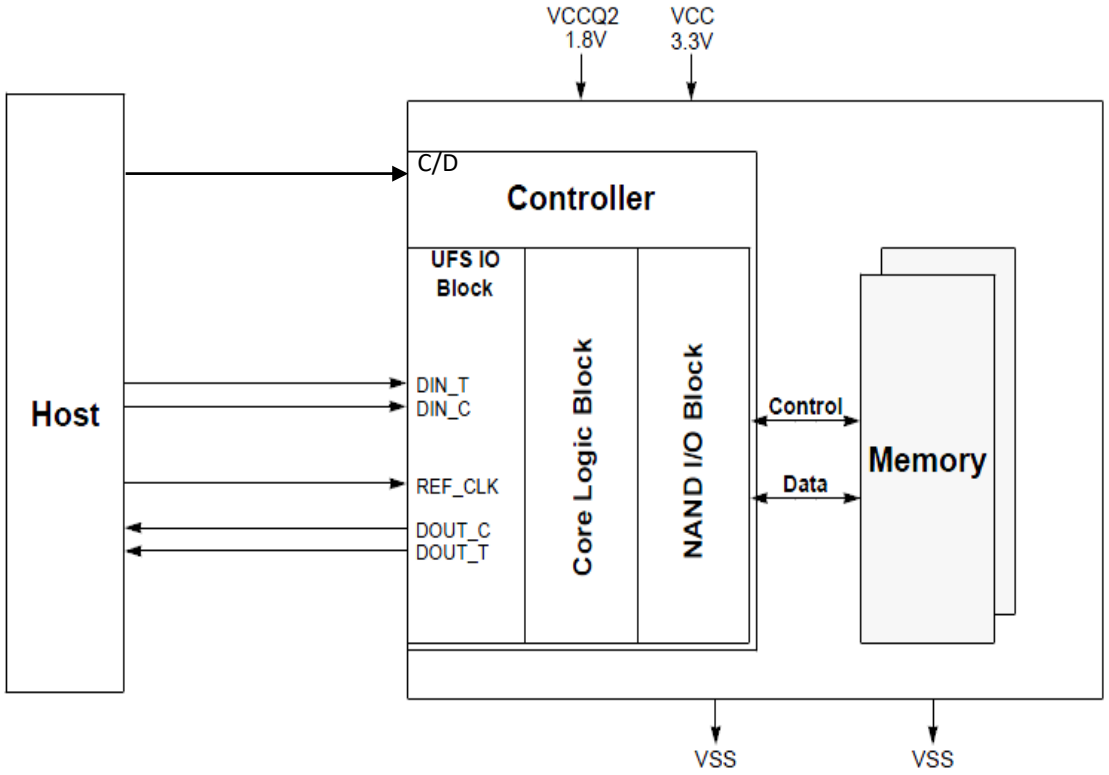


Figure 5.1 — UFS Card Block Diagram

Table 5.1 — Signal Name and Definitions

Name	Type	Description
VCC	Supply (3.3V)	Supply voltage for the memory devices
VCCQ2	Supply (1.8V)	Supply voltage used typically for the PHY interface and the memory controller and any other internal low voltage block
VSS	Supply	Ground
C/D	GND	Card Detection Pin
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 DIN_C	Input	Downstream data lane 0. DIN_T is the positive node of the differential signal.
Differential output signals from the UFS device to the host		
DOUT_T DOUT_C	Output	Upstream data lane 0. DOUT_T is the positive node of the differential signal.

6 UFS Card Design

The UFS card follows the shark design and a simplified pictorial representation is shown in Figure 6.1. Refer to JEP95, MO-320, for more detailed mechanical dimensions of Figure 6.1, Figure 6.2 and Figure 6.3.

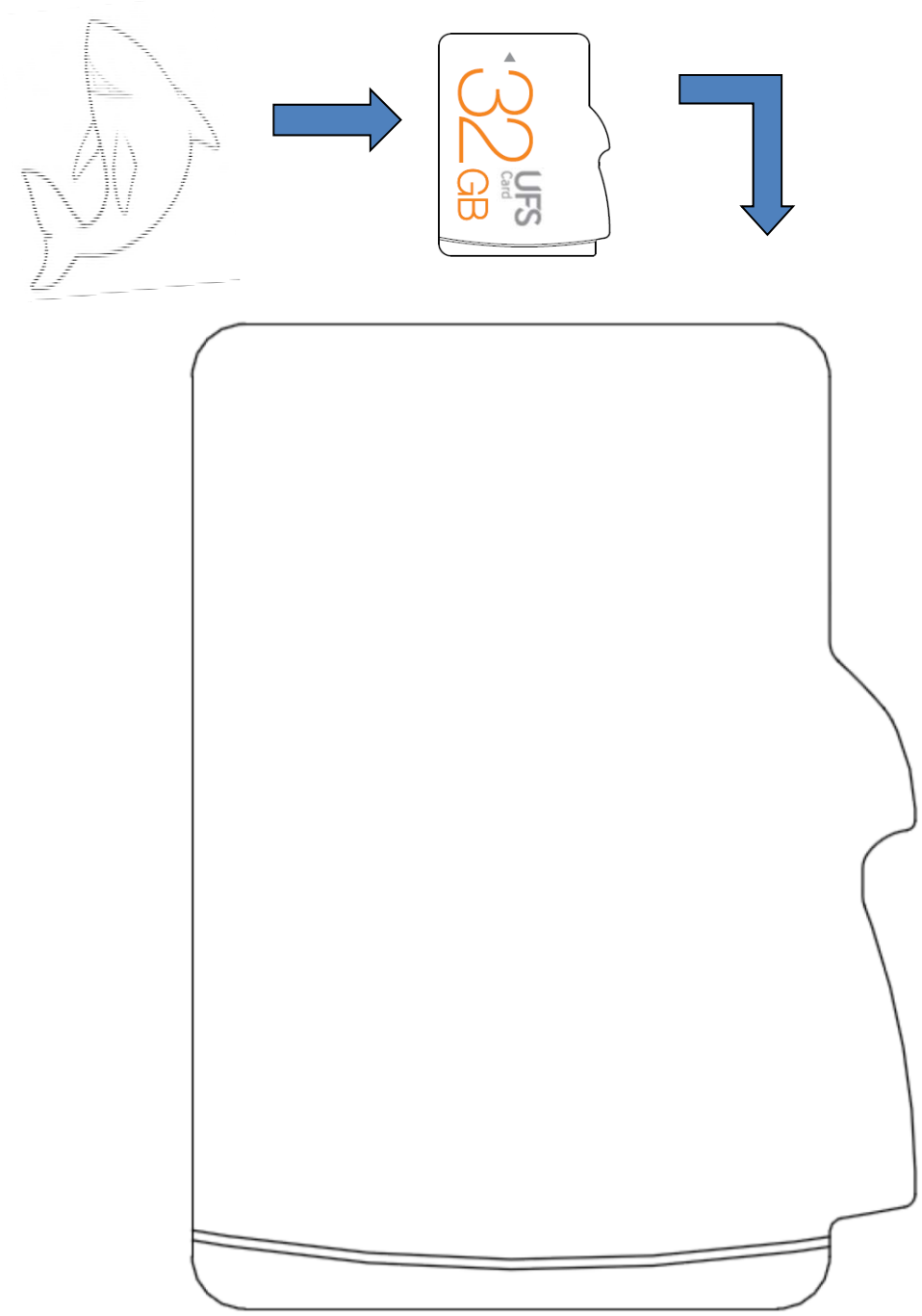
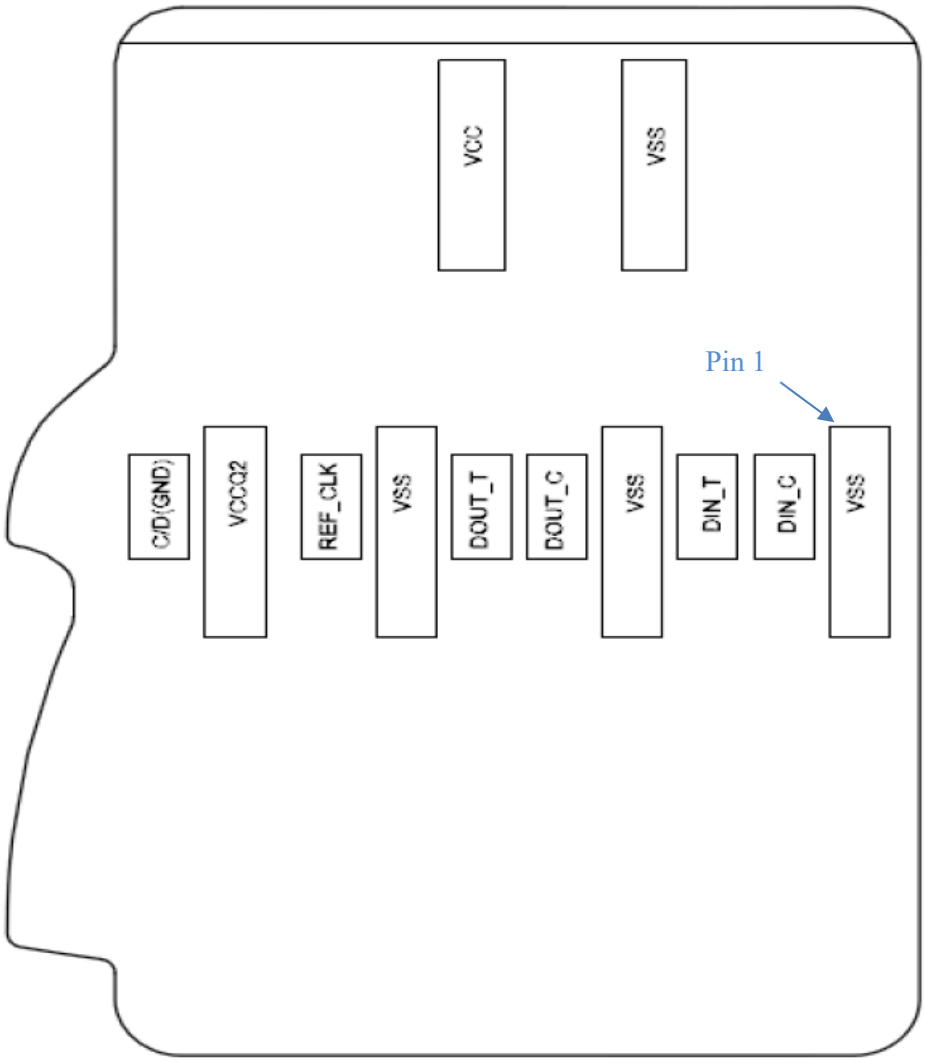


Figure 6.1 — UFS Card Top View

6 UFS Card Design (cont'd)



Pin No.	Signal Name
1	VSS
2	DIN_C
3	DIN_T
4	VSS
5	DOUT_C
6	DOUT_T
7	VSS
8	REF_CLK
9	VCCQ2
10	C/D(GND)
11	VSS
12	VCC

Figure 6.2 — UFS Card Bottom View

6 UFS Card Design (cont'd)

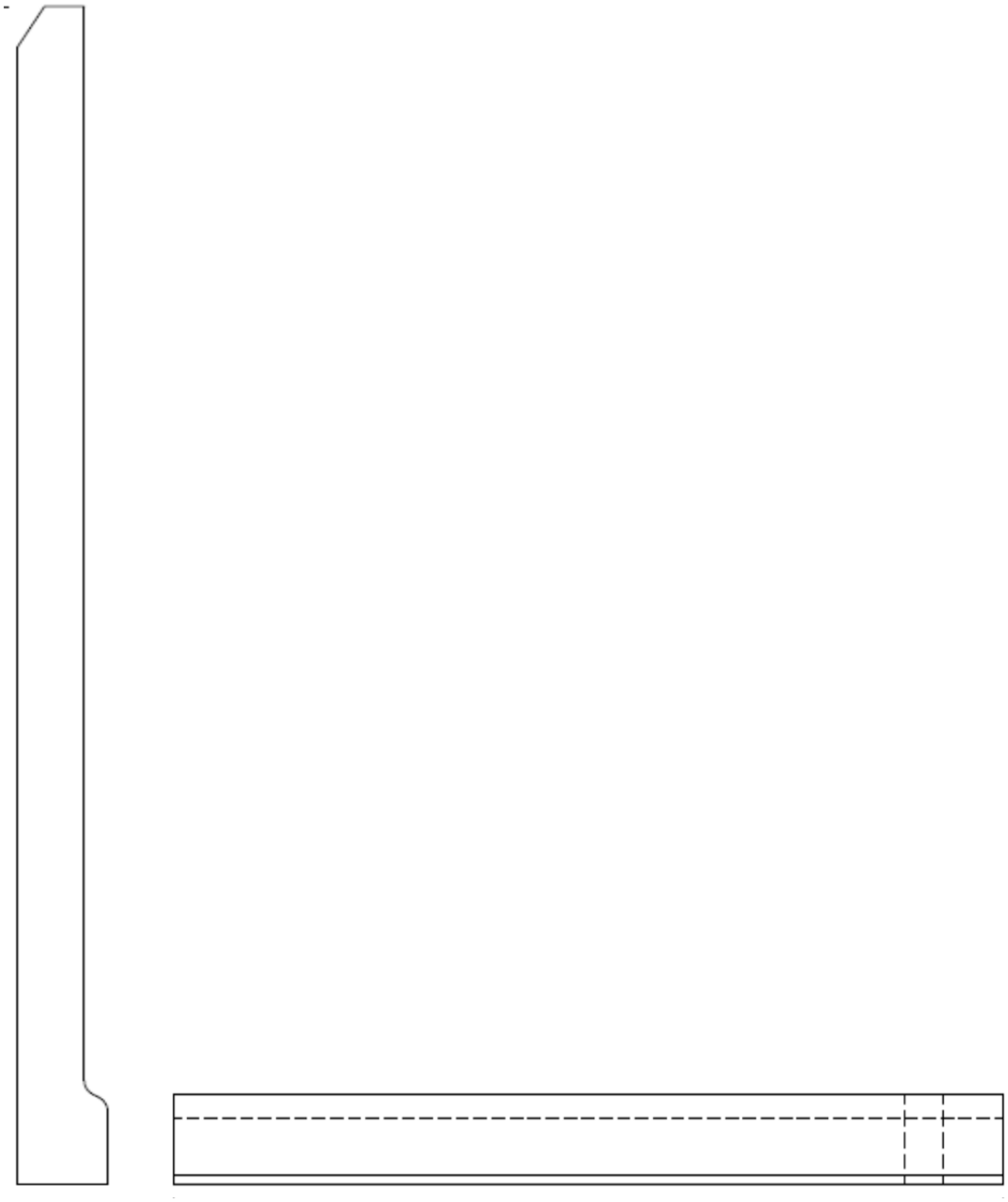


Figure 6.3 — UFS Card Side View

7 Supported Features and Commands of UFS Card

7.1 Supported Features

The embedded UFS and UFS card follow the same protocol. However, because they will be used in different environments, the use cases will differ. The UFS card will support fewer features compared to the embedded UFS. Table 7.1 shows the differences between the embedded UFS and UFS card.

Table 7.1 — Comparison of embedded UFS and UFS Card

Category	Item	eUFS 3.0	UFS Card v3.0
General	Supported PWM Gears	PWM-GEAR1 (Mandatory) PWM-GEAR2, GEAR3, GEAR4, GEAR5, GEAR6, GEAR7 (optional)	PWM-GEAR1 Only (PWM shall be able to be operated without reference clock)
	Supported HS Gears	HS-GEAR1, HS-GEAR2, HS-GEAR3, HS-GEAR4	HS-GEAR1, HS-GEAR2, HS-GEAR3, HS-GEAR4
Interface	Power Supplies	VCCQ (1.2V), VCCQ2 (1.8V), VCC (2.5V or 3.3V)	VCCQ2 (1.8 V), VCC (3.3 V)
	Lanes	Up to 2	Up to 1
	HW Reset	Supported	Not Supported
	Reference Clock	19.2 MHz, 26 MHz(Default), 38.4MHZ.	19.2 MHz, 26MHz (Default),
Functional	bRefClkFreq	Persistent (Default 26 MHz)	Volatile (Default 26 MHz)
	bMaxNumOfRTT	Persistent (Default 2)	Volatile (Default 2)
	bMaxDataInSize	Persistent (Min 4KB)	Volatile (Min 4KB)
	bMaxDataOutSize	Persistent (Min 4KB)	Volatile (Min 4KB)
	bActiveICCLevel	Persistent	Volatile
	RPMB	Supported	Not supported
	Boot Feature	Supported	Supported
	bOutOfOrderDataEn	Write once (Default 0)	Volatile (Default 0)

The UFS card is supposed to be a lite version of the embedded UFS. So support for unnecessary gears shall be removed and the UFS card shall support only PWM-GEAR1, HS-GEAR1, HS-GEAR2, HS-GEAR3 and HS-GEAR4. Similarly, the UFS card shall support up to 1 lane compared to the embedded UFS which supports 2 lanes.

7.1 Supported Features (cont'd)

VCCQ: The VCCQ pin is removed from the UFS card to reduce the pin count. UFS card vendors can use an embedded LDO regularor to achieve a lower voltage that is aligned to their lower voltage circuits from 3.3 V or 1.8 V source. As a power supply, 3.3 V and 1.8 V are mandatory considering the NAND controller, I/O logic. Since 1.2 V can be generated from 3.3 V or 1.8 V, VCCQ pin (1.2 V) is not supported in the UFS card.

HW Reset: In the case of the embedded UFS, the chip cannot be detached from the system PCB, so the HW reset pin is required. In the case of UFS card, the HW reset pin is not supported and the UFS card can be reset by other existing mechanisms, if necessary. TMinimizing the UFS card pin count can reduce the development and testing cost.

Attributes: In the embedded UFS, the access property of bRefClkFreq, bMaxNumOfRTT, bMaxDataInSize, bMaxDataOutSize, bOutOfOrderDataEn and bActiveICCLevel ispersistent. As the embedded UFS chip can not be removed from the host, making these attribute values as persistent avoids repeatly re-initializing these attributes. However, in the case of the UFS card, it can be inserted into a different host which may want to use different values for these attributes. So in the UFS card, these attribute values shall be reset to the default value after every reset or power cycle.

7.2 Supported UFS Card SCSI Commands

The Basic Universal Flash Storage (UFS) SCSI commands are compatible with SCSI Primary Commands - 4 [SPC] and SCSI Block Commands - 3 [SBC].

If enabled (bLUEnable = 01h), each logical unit shall support the commands as defined in Table 7.2.

Table 7.2 — Comparision of Supported UFS SCSI Command Set

Command name	Supported Command	
	eUFS v3.0	UFS Card v3.0
FORMAT UNIT	M	M
INQUIRY	M	M
MODE SELECT (10)	M	M
MODE SENSE (10)	M	M
PRE-FETCH (10)	M	M
PRE-FETCH (16)	O	Not Support
READ (6)	M	M
READ (10)	M	M
READ (16)	O	Not Support
READ BUFFER	M	M
READ CAPACITY (10)	M	M
READ CAPACITY (16)	M	M
REPORT LUNS	M	M
REQUEST SENSE	M	M
SECURITY PROTOCOL IN	M	Not Support
SECURITY PROTOCOL OUT	M	Not Support
SEND DIAGNOSTIC	M	M
START STOP UNIT	M	M
SYNCHRONIZE CACHE (10)	M	M
SYNCHRONIZE CACHE (16)	O	Not Support
TEST UNIT READY	M	M
UNMAP	M	M
VERIFY (10)	M	M
WRITE (6)	M	M
WRITE (10)	M	M
WRITE (16)	O	Not Support
WRITE BUFFER	M	M
NOTE M: mandatory, O: optional		

If device receives a command that is not supported by UFS Card, then the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID COMMAND OPERATION CODE.

8 UFS Card Initialization

The UFS card initialization follows the same sequence as the embedded UFS. But since the UFS Card is a removable device, VCC(3.3 V) and VCCQ2(1.8 V) shall be provided after the UFS card is fully inserted into the card slot. The C/D pin may be used to support card insertion detection (refer to Annex A).

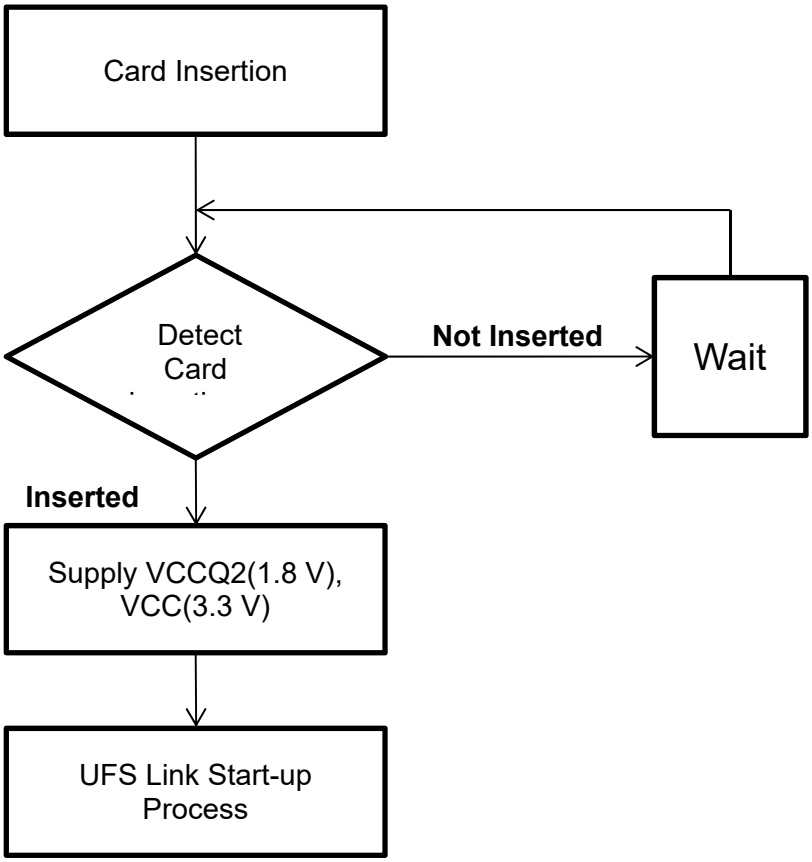


Figure 8.1 — UFS Card Initialization

8.1 Initialization Sequence

Once the reset is done, the host will set the appropriate value for attributes to make the card compatible with the particular host.

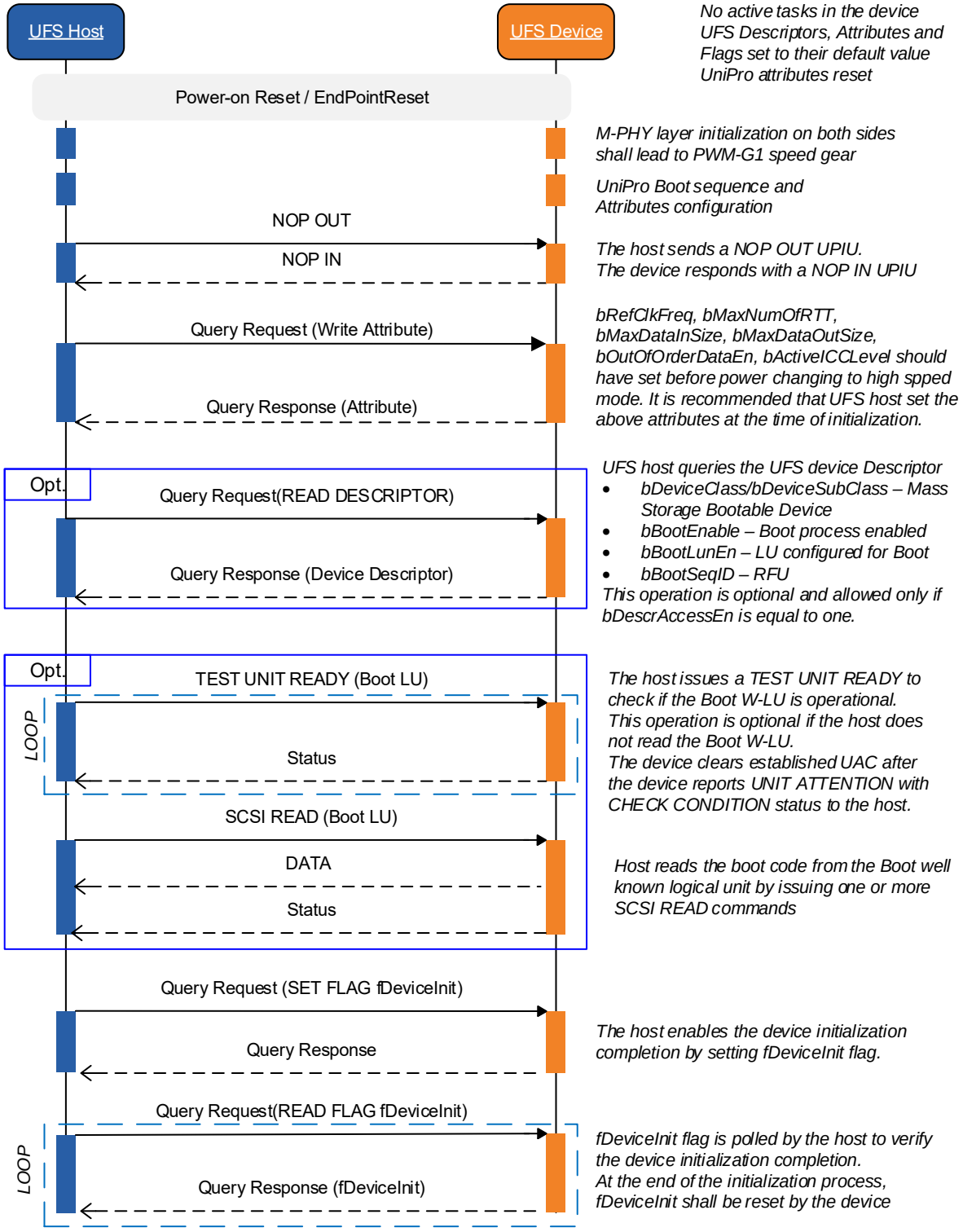


Figure 8.2 — UFS Card Initialization Sequence

8.1 Initialization Sequence (cont'd)

The UFS card initialization is the same as the initialization of the embedded UFS, except the bRefClkFreq, bMaxNumOfRTT, bMaxDataInSize, bMaxDataOutSize,bOutOfOrderDataEn and bActiveICClevel attribute values will be set by the host whenever a reset or power cycle occurs. In the case of the embedded UFS, these attributes will retain the value after the reset since the access property of those attributes are defined as persistent. However in the case of the UFS Card, since the UFS card can be inserted into a different host, the access property of these attribute values shall be volatile and will be reset to their default value whenever a reset or power cycle occurs. Therefore the host has to set the appropriate value for those attributes before changing the mode to high speed mode.

9 Power Consumption

The UFS card of any vendor shall be able to work in any host supporting UFS Card. Therefore, the power level which any UFS card can work with shall be defined for the host to provide the required amount of power. Table 9.1 defines UFS Card RMS current consumption and peak current consumption for operations at HS-GEAR3 and HS-GEAR4.

Table 9.1— UFS Card Maximum Power Consumption (in RMS and Peak)

High-Spee GEARs	RMS (100ms) (1)		Peak (5 us) (2)	
	VCC	VCCQ2	VCC	VCCQ2
HS-GEAR3	300 mA	350 mA	500 mA	400 mA
HS-GEAR4	400 mA	650 mA	700 mA	700 mA

NOTE 1 The measurement for RMS current is the average RMS current consumption at norminal voltages (3.3V,1.8V each) over a period of 100ms. RMS may be used to estimate UFS Card energy consumption.
NOTE 2 Peak current measured at maximum voltages(3.6V, 1.95V each) in 5 us timing window. Peak current is the requirement to be considered for the related with Host PMIC power budget allocation.

10 Descriptors, Attributes and Flags for UFS Card

10.1 Overview

The UFS Card is a removable storage version of the JEDEC UFS specification. Since some features like PSA (Product State Awareness) are not supported by UFS Card, the value of Device Descriptors and Unit Descriptors for PSA shall be ignored. To indicate such UFS features which are not supported or supported in limited scope in UFS Card, this section describes the Device Descriptors, Unit Descriptors, Attributes and Flags which are different from UFS v3.0 specification[UFS]. Except for those described in this section, other Unit Descriptors, Attributes and Flags shall follow as defined in [UFS].

NOTE When values which are defined as ‘not supported’ in this UFS Card specification but which are defined in [UFS] and are written by host’s command, the command shall be succeeded but the value for those ‘not supported’ features shall be ignored.

10.2 Device Descriptor

The Device Descriptor is read only, and some of its parameters can be changed by writing the corresponding parameter of the Configuration Descriptor.

In a QUERY REQUEST UPIU, the Device Descriptor is addressed by setting: DESCRIPTOR IDN = 00h, INDEX = 00h and SELECTOR = 00h.

10.2 Device Descriptor (cont'd)

Table 10.1 — Device Descriptor for UFS Card that differ from [UFS]

DEVICE DESCRIPTOR					
Offset	**Size**	**Name**	**MDV (1)**	**User Conf**	**Description for UFS Card**
04h	1	bDeviceSubClass	Device specific	No	UFS Card Storage Subclass Bits (0/1) specify as follows: Bit 0: Bootable / Non-Bootable Bit 1: Embedded / Removable (shall be 0x1b for UFS Card) Others: Reserved Examples: 02h: Removable Bootable 03h: Removable Non-Bootable
07h	1	bNumberWLU	03h	No	Number of Well known Logical Units. (RPMB Well known Logical Units is not supported for UFS Card.)
0Ch	1	bSecureRemovalType	00h	Yes	Secure Removal Type 00h: information removed by an erase of the physical memory 01h: Reserved 02h: Reserved 03h: information removed using a vendor defined mechanism. Others: Reserved
0Dh	1	bSecurityLU	01h	No	bSecurityLU is not supported. The value of this descriptor shall be ignored.
1Fh	1	bUFSFeaturesSupport	Device specific	No	UFS Features Support This field indicates which features are supported by the device. A feature is supported if the related bit is set to one. bit[0]: Field Firmware Update (FFU) bit[1]: Reserved bit[2]: Device Life Span bit[3]: Refresh Operation bit[4]: TOO_HIGH_TEMPERATURE bit[5]: TOO_LOW_TEMPERATURE bit[6]: Extended Temperature Others: Reserved Bit 0 shall be set to one.
25h	4	dPSAMaxDataSize	Device specific	No	dPSAMaxDataSize is not supported. The value of this descriptor shall be ignored
29h	1	bPSAStateTimeout	Device specific	No	bPSAStateTimeout is not supported. The value of this descriptor shall be ignored

NOTE 1 The column "MDV" (Manufacturer Default Value) specifies parameter values after device manufacturing. Some parameters may be configured by the user writing the Configuration Descriptor.

NOTE 2 "User Conf." column specifies which fields can be configured by the user writing the Configuration Descriptor: "Yes" means that the field can be configured, "No" means that the field is a capability of the device and cannot be changed by the user. The desired value shall be set in the equivalent parameter of the Configuration Descriptor.

10.3 Unit Descriptor

This page describes specific characteristics and capabilities of an individual logical unit, for example geometry of the device and maximum addressable item. There are up to three unit descriptors. In a QUERY REQUEST UPIU, a Unit Descriptor is addressed by setting: DESCRIPTOR IDN = 02h, INDEX = unit index, and SELECTOR = 00h.

Table 10.7 — Unit Descriptor that differ from [UFS]

UNIT DESCRIPTOR					
Offset	**Size**	**Name**	**MDV (1)**	**User Conf. (2)**	**Description for UFS Card**
07h	1	bPSASensitive	Device specific	No	bPSASensitive is not supported. The value of this descriptor shall be ignored.
20h	2	wContextCapabilities	00h	Yes	wContextCapabilities is not supported. The value of this descriptor shall be ignored
22h	1	bLargeUnitGranularity_M1	Device specific	No	bLargeUnitGranularity_M1 is not supported. The value of this descriptor shall be ignored

NOTE 1 The column “MDV” (Manufacturer Default Value) specifies parameters value after device manufacturing. Some fields may be configured by the user writing the Configuration Descriptor.

NOTE 2 “User Conf.” column specifies which fields can be configured by the user writing the Configuration Descriptor: “Yes” means that the field can be configured, “No” means that the field is a capability of the device and cannot be changed by the user. The desired value shall be set in the equivalent parameter of the Configuration Descriptor.

10.4 Configuration Descriptor

The device configuration set by the manufacturer can be modified by writing the Configuration Descriptor. In particular, the Configuration Descriptor allows the user to configure parameters included in the Device Descriptors, and Unit Descriptors. The Configuration Descriptor can be written if bConfigDescrLock attribute value is equal to 00h. If bConfigDescrLock attribute value is 01h the Configuration Descriptor is locked and a write request shall fail.
There is one Configuration Descriptor.

The Configuration Descriptor is addressed by setting DESCRIPTOR IDN = 01h, INDEX = 00h and SELECTOR = 00h in a QUERY REQUEST UPIU, and used to change configurable parameters of Device Descriptor and the three Unit Descriptors (LU 0 to LU 2).

Table 10.2 shows the Configuration Descriptor format with DESCRIPTOR IDN = 01h, INDEX = 00h and SELECTOR = 00h: the lower address space is used for Configuration Descriptor header, and Device Descriptor configurable parameters. Then there are three address spaces for user configurable parameters included in the Unit Descriptors of LU 0 to LU 2.

Parameter offsets are defined based on:

- Offset of the Unit Descriptor 0 configurable parameters within the Configuration Descriptor (B).
- Length of Unit Descriptor configurable parameters (L)

Values for B and L are stored in the Device Descriptor parameters bUD0BaseOffset and bUDConfigPLength respectively which is defined in [UFS].

10.4 Configuration Descriptor (cont'd)

Table 10.2 — Configuration Descriptor Format for UFS Card

Offset	Description for UFS Card
00h … (B-1)h	Configuration Descriptor header, Device Descriptor configurable parameters
(B)h … (B+L-1)h	Unit Descriptor 0 configurable parameters
(B+L)h … (B+2*L-1)h	Unit Descriptor 1 configurable parameters
(B+2*L)h … (B+3*L-1)h	Unit Descriptor 2 configurable parameters
NOTE B is offset of the Unit Descriptor 0configurable parameters within the Configuration Descriptor, L is the total size of the configurable Unit Descriptor parameters.	

Table 10.2 defines Configuration Descriptor header, and Device Descriptor configurable parameters within the Configuration Descriptor with INDEX = 00h.

See Table 10.3, for details about configurable parameters.

Table 10.3 — Configuration Descr. Header and Device Descr. Conf. parameters that differ from [UFS]

Configuration Descriptor Header and Device Descriptor configurable parameters				
Offset	**Size**	**Name**	**MDV (1,2)**	**Description for UFS Card**
02h	1	bConfDescContinue	00h	00h: This value indicates that this is the last Configuration Descriptor in a sequence of write descriptor query requests. Device shall perform internal configuration based on received Configuration Descriptor(s). Others: Reserved.
06h	1	bHighPriorityLUN		bHighPriorityLUN is not supported. The value of this parameter shall be ignored
0Ch	1	bRPMBRegionEnable		RPMB is not supported. The value of this parameter shall be ignored
0Dh	1	bRPMBRegion1Size		RPMB is not supported. The value of this parameter shall be ignored
0Eh	1	bRPMBRegion2Size		RPMB is not supported. The value of this parameter shall be ignored
0Fh	1	bRPMBRegion3Size		RPMB is not supported. The value of this parameter shall be ignored
NOTE 1 The column "MDV" (Manufacturer Default Value) specifies parameter values after device manufacturing. NOTE 2 See Table 10.1 , Device Descriptor for UFS Card, for the default parameters value set by the device manufacturer.				

10.4 Configuration Descriptor (cont'd)

Table 10.4 defines the Unit Descriptors user configurable parameters within the Configuration Descriptor.

Table 10.4— Unit Descriptor configurable parameters for LU for UFS Card

Unit Descriptor configurable parameters			
Offset	**Size**	**Name**	**Description for UFS Card**
00h	1	bLUEnable	Logical Unit Enable
01h	1	bBootLunID	Boot LUN ID
02h	1	bLUWriteProtect	Logical Unit Write Protect
03h	1	bMemoryType	Memory Type
04h	4	dNumAllocUnits	Number of allocation units assigned to the logical unit.
08h	1	bDataReliability	Data Reliability
09h	1	bLogicalBlockSize	Logical Block Size
0Ah	1	bProvisioningType	Provisioning Type
0Bh	2	wContextCapabilities	wContextCapabilities is not supported.
0Dh:0Fh	3	Reserved	

10.5 Flags

A flag is a single Boolean value that represents a TRUE or FALSE, '0' or '1', ON or OFF type of value. A flag can be cleared or reset, set, toggled or read. Flags are useful to enable or disable certain functions or modes or states within the device.

Read access property (read or write only) and write access property (read only, write only, persistent, etc.) are defined for each flag. Table 10.8 describes the supported access properties for flags.

Table 10.8— Flags for UFS Card that differ from [UFS]

FLAGS					
IDN	**Name**	**Type**	**Type**[1]	**Default**	**Description for UFS Card**
08h	fPhyResourceRemoval	Read / Persistent	D	0	fPhyResourceRemoval is not supported. The value of this flag shall be ignored.

NOTE 1 The type "D" identifies a device level attribute, while the type "A" identifies an array of attributes. If Type = "D", the attribute is addressed setting INDEX = 00h and SELECTOR = 00h.

10.6 Attributes

An Attribute is a parameter that represents a specific range of numeric values that can be written or read. For example, the maximum Data In data packet size would be an attribute. Attribute size can be from 1-bit to 32-bit. Attributes of the same type can be organized in arrays, each element of them identified by an index. For example, in case of parameter that is logical unit specific, the LUN would be used as index.

Read access property (read or write only) and write access property (read only, write once, persistent, etc.) are defined for each attribute. Table 10.9 describes the supported access properties for attributes.

Table 10.9 — Attributes for UFS Card that differ from [UFS]

ATTRIBUTES						
IDN	**Name**	**Access Property**	**Size**	**Type[1]** **# Ind.[2]** **# Sel.[3]**	**MDV[4]**	**Description for UFS Card**
09h	dDynCapNeeded	Read only	4 bytes	A Number of LU specified by bMaxNumberLU (LUN) 0	0000 0000h	dDynCapNeeded is not supported. The value of this arribute shall be ignored
0Dh	wExceptionEventControl	Read / Volatile	2 bytes	D	0000h	Exception Event Control This attribute enables the setting of the EVENT_ALERT bit of Device Information field, which is contained in the RESPONSE UPIU. EVENT_ALERT is set to one if at least one exception event occured (wExceptionEventStatus[i]) and the corresponding bit in this attribute is one (wExceptionEventControl[i]). Bit 0: Reserved Bit 1: Reserved Bit 2: URGENT_BKOPS_EN Bit 3: TOO_HIGH_TEMP_EN Bit 4: TOO_LOW_TEMP_EN Bit 5 -15: Reserved
0Eh	wExceptionEventStatus	Read only	2 bytes	D		Each bit represents an exception event. A bit will be set only if the relevant event has occurred (regardless of the wExceptionEventControl status). Bit 0: Reserved Bit 1: Reserved Bit 2: URGENT_BKOPS Bit 3: TOO_HIGH_TEMP Bit 4: TOO_LOW_TEMP Bit 5 -15: Reserved

ATTRIBUTES						
IDN	**Name**	**Access Property**	**Size**	**Type[1]** **# Ind.[2]** **# Sel.[3]**	**MDV[4]**	**Description for UFS Card**
10h	wContextConf	Read / Volatile	2 bytes	A Number of LU specified by bMaxNumberLU (LUN) 15 (ID)	0000h	wContextConf is not supported. The value of this arribute shall be ignored.
15h	bPSAState	Read / Persistent	1 byte	D	Device specific	bPSAState is not supported, The value of this arribute shall be ignored.
16h	dPSADataSize	Read / Persistent	4 bytes	D	00 … 00h	dPSADataSize is not supported. The value of this arribute shall be ignored.

NOTE 1 The type “D” identifies a device level attribute, while the type “A” identifies an array of attributes. If Type = “D”, the attribute is addressed setting INDEX = 00h and SELECTOR = 00h.

NOTE 2 For array of attributes, “# Ind.” specifies the amount of valid values for the INDEX field in QUERY REQUEST/RESPONSE UPIU. If # Ind = 0, the attribute is addressed setting INDEX = 00h.

NOTE 3 For array of attributes, “# Sel.” specifies the amount of valid values for the SELECTOR field in QUERY REQUEST/RESPONSE UPIU. If # Sel = 0, the attribute is addressed setting SELECTOR = 00h

NOTE 4 The column “MDV” (Manufacturer Default Value) specifies attribute values after device manufacturing.

NOTE 5 bRefClkFreq field had “Write once” Access Property up to UFS Card 1.0.

Annex A (informative) Host Guideline for UFS Card Detection

The card detection (C/D) pin can be utilized to detect card insertion, by adding a pull-up resistor to the C/D pin in the host-side C/D pin as shown in Figure A.1. When the UFS card is not inserted, the host-side C/D pin shows a non-zero voltage value. When the UFS card is inserted, the host-side C/D pin shows a zero voltage value because the C/D pin of the device is tied to ground.

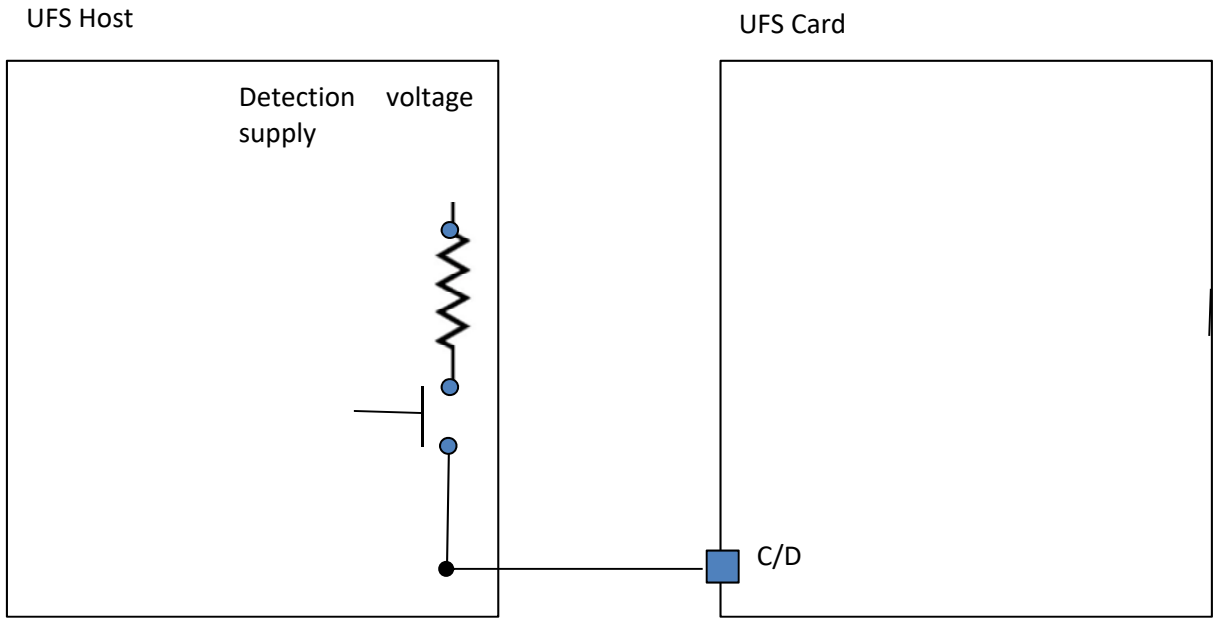


Figure A.1 — Host guideline for UFS Card Detection

Annex B (informative) Difference between Specification Revisions

B.1 Changes between JESD220-2B and its predecessor JESD220-2A (UFS Card v1.1, JAN 2018)

B.1.1 New features or new definitions

The following items were added:

- Added HS-GEAR4 support and its Maximum Power Consumption. See Table 7.1 and Table 9.1
- Added supported SCSI commands. See 7.2 “Supported UFS Card SCSI Commands”
- Added Descriptor, Attribute and Flags for UFS Card. See 10 “Descriptor, Attributes and Flags for UFS Card”

B.1.2 Changes in section 2 “Normative Reference”

- JESD220D, Universal Flash Storage (UFS), Version 3.0
- M-PHY^{SM} specification: from version 3.0 to version 4.1.
- Unified Protocol (UniPro^{SM}) specification: from version 1.6 to version 1.8.

B.1.3 Changes in features already defined in its predecessor UFS Card v1.1

Changes related to features already defined in the previous version of the standard are summarized in the following:

- Boot Feature is supported. See Table 7.1 and Figure 8.2
- RPMB is not supported. See Table 7.1

Several clarifications were added and editorial changes were implemented in addition to what summarized in this annex.

Standard Improvement Form JEDEC _____

The purpose of this form is to provide the Technical Committees of JEDEC with input from the industry regarding usage of the subject standard. Individuals or companies are invited to submit comments to JEDEC. All comments will be collected and dispersed to the appropriate committee(s).

If you can provide input, please complete this form and return to:

JEDEC
Attn: Publications Department
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2107

Fax: 703.907.7583

1. I recommend changes to the following:

☐ Requirement, clause number _____

☐ Test method number _____ Clause number _____

The referenced clause number has proven to be:

☐ Unclear ☐ Too Rigid ☐ In Error

☐ Other _____

2. Recommendations for correction:

3. Other suggestions for document improvement:

Submitted by

Name: _____ Phone: _____

Company: _____ E-mail: _____

Address: _____

City/State/Zip: _____ Date: _____


JEDEC®