SD Specifications
Part 1
Physical Layer
Simplified Specification

Version 9.10

December 1, 2023

Technical Committee
SD Card Association

Revision History

Date	Version	Changes compared to previous issue
April 3, 2006	1.10	Physical Layer Simplified Specification Version 1.10 initial release. (Supplementary Notes Ver1.00 is applied.)
September 25, 2006	2.00	Physical Layer Simplified Specification Version 2.00 (1) High Capacity Memory Card (SDHC) Speed Class 2/4/6
May 18, 2010	3.01	Physical Layer Simplified Specification Version 3.01 (1) Extended Capacity Memory Card (SDXC) (2) Ultra High Speed I (UHS-I) (3) Speed Class 10 (4) UHS Speed Grade 1 Current Limit
January 22, 2013	4.10	Physical Layer Simplified Specification Version 4.10 (1) UHS-II Interface (2) UHS Speed Grade 3 (3) Power Limit Function Extension Specification
August 10, 2016	5.00	Physical Layer Simplified Specification Version 5.00 Video Speed Class VSC4/6/10/30/60/90
April 10, 2017	6.00	Physical Layer Simplified Specification Version 6.00 (1) Discard and FULE to Erase (2) Card Ownership Protection (COP) for Card Lock/Unlock (3) Application Performance Class for A1/A2 (4) Cache (5) Self-Maintenance (6) Command Queuing (7) Simplifed Mechanical Drawings ✓ Simplifed Bus Timings
August 29, 2018	6.00	Revised Disclaimers
March 25, 2020	7.10	Physical Layer Simplified Specification Version 7.10 (1) SD Express (PCIe/NVMe interface added) Full size (2) microSD Express (3) SDUC – Ultra Capacity card (2TB-128TB) (4) CPRM security optional (5) Applied the Physical 7.10-6.00 Supplementary Notes Version 32.00
September 23, 2020	8.00	Physical Layer Simplified Specification Version 8.0 (1) Full Size SD Express cards with following interface support added: a. PCIe 4.0 single lane b. PCIe 4.0 dual lane c. PCIe 3.1 dual lane
August 22, 2022	9.00	Physical Layer Simplified Specification Version 9.0 (1) Boot Functionalities (2) TCG (Trusted Computing Group) security (3) RPMB (Replay Protected Memory Block) (4) Updated Write Protection function including Write Protect Until Power Cycle
December 1, 2023	9.10	Physical Layer Simplified Specification Version 9.1 (1) SD Express Speed Class

Release of SD Simplified Specification/Addendum

The following conditions apply to the release of the SD Simplified Specification/Addendum by the SD Card Association. The Simplified Specification/Addendum is a subset of the complete version of SD Specification/Addendum that is owned by the SD Card Association.

Conditions for publication

Publisher:

SD Card Association
5000 Executive Parkway, Suite 302,
San Ramon, CA 94583 USA
Telephone: +1 (925) 275-6615,
Fax: +1 (925) 275-6691
E-mail: help@sdcard.org

Copyright Holders:

KIOXIA Corporation
Panasonic Connect Co., Ltd.
SanDisk LLC
The SD Card Association

Notes:

The copyright of the previous versions (Version 1.00 and 1.01) and all corrections or non-material changes thereto are owned by SD Group.
The copyright of material changes to the previous versions (Version 1.01) are owned by SD Card Association.

Disclaimers:

This Simplified Specification is made available by the SD Card Association (the "SDA") at https://www.sdcard.org/downloads/pls/index.html (the "Site") and your access to and/or use of this Simplified Specification is subject to the SIMPLIFIED SPECIFICATION TERMS AND CONDITIONS (the "Terms") that are displayed by clicking the "Download" button at https://www.sdcard.org/downloads/pls/index.html.

If you are viewing or have accessed this Simplified Specification via any source, medium, or in any other way other than directly from the Site pursuant your acceptance of the Terms, then your access to, viewing of, and/or use of the Simplified Specification is in violation of the SDA's and its licensors' intellectual property rights. Accordingly, unless obtained directly from the Site pursuant to the Terms, immediately cease and desist all viewing, using, or accessing the Simplified Specification; destroy any copies of the Simplified Specification in your possession, custody or control; and, if you desire access to the Simplified Specification, proceed to the Site to obtain access and use of the Simplified Specification in an authorized manner pursuant to the Terms.

Distribution of the Simplified Specification, other than through the Site, is a violation of the Terms and the intellectual property rights of the SDA and its licensors. The only rights granted in the Simplified Specification are those expressly granted in the Terms. All rights not expressly granted pursuant to your acceptance of the Terms are reserved to the SDA and its licensors. Notice is also hereby provided that notwithstanding any rights granted by the Terms, any implementation of the Simplified Specifications or any portions thereof may require a separate license from the SDA, SD Group, SD-3C, LLC or other third parties.

Conventions Used in This Document

Naming Conventions

- Some terms are capitalized to distinguish their definition from their common English meaning. Words not capitalized have their common English meaning.

Numbers and Number Bases

- Hexadecimal numbers are written with a lower case "h" suffix, e.g., FFFFh and 80h.
- Binary numbers are written with a lower case "b" suffix (e.g., 10b).
- Binary numbers larger than four digits are written with a space dividing each group of four digits, as in 1000 0101 0010b.
- All other numbers are decimal.

Key Words

- May: Indicates flexibility of choice with no implied recommendation or requirement.
- Shall: Indicates a mandatory requirement. Designers shall implement such mandatory requirements to ensure interchangeability and to claim conformance with the specification.
- Should: Indicates a strong recommendation but not a mandatory requirement. Designers should give strong consideration to such recommendations, but there is still a choice in implementation.
- Reserved: Indicates a bit, byte, field, and value that are set aside for future standardization or other usage. A reserved bit, byte, or field shall be set to zero unless specified differently in this specification, or in accordance with a future extension to this or other standards. Hosts and Cards are not required to check reserved bits, bytes, or fields for zero values. Values that are reserved for future standardization shall not be specified by hosts and cards.

Application Notes

Some sections of this document provide guidance to the host implementers as follows:

Application Note: This is an example of an application note.

Table of Contents

Table of Figures

Table of Tables

1. General Description

SD Memory Card is a memory card that is specifically designed to meet the security, capacity, performance, and environment requirements inherent in newly emerging audio and video consumer electronic devices. The SD Memory Card may include a content protection mechanism that complies with the security of the SDMI standard and will be faster and capable of higher Memory capacity. The SD Memory Card security system uses mutual authentication and a "new cipher algorithm" to protect against illegal usage of the card content. A Non-secure access to the user's own content is also available.

SD memory cards may also support a second security system based on commonly used standards, such as ISO-7816, which can be used to interface the SD memory card into public networks and other systems supporting mobile e-commerce and digital signature applications. Moreover, SD memory card can implement TCG (Trusted Computing Group) and RPMB (Replay Protected Memory Block) as another security system, mainly for IoT or mobile/computing usage.

In addition to the SD Memory Card, there is the SD I/O (SDIO) Card. The SDIO Card specification is defined in a separate specification named: "SDIO Card Specification" that can be obtained from the SD Association. The SDIO Specification defines an SD card that may contain interfaces between various I/O units and an SD Host. The SDIO card may contain memory storage capability as well as its I/O functionality. The Memory portion of SDIO card shall be fully compatible to the given Physical Layer Specification. The SDIO card is based on and compatible with the SD Memory card. This compatibility includes mechanical, electrical, power, signalling, and software. The intent of the SD I/O card is to provide high-speed data I/O with low power consumption for mobile electronic devices. A primary goal is that an I/O card inserted into a non-SDIO aware host will cause no physical damage or disruption of that device or its software. In this case, the I/O card should simply be ignored. Once inserted into an SDIO aware host, the detection of the card will be via the normal means described in the given Physical Layer Specification with some extensions that are described in the SDIO Specification.

The basic SD Memory Card communication is based on 9-pin interface (Clock, Command, 4xData and 3xPower lines) designed to operate in at maximum operating frequency of 208 MHz and low voltage range. Additional communication methods, based on differential signaling interface (UHS-II and PCIe®/NVMe™) are specified as optional.The communication protocol is defined as a part of this specification unless specified otherwise (i.e. PCIe and NVMe cases).

The SD Specifications are divided into several documents. The SD Specifications documentation structure is given in Figure 1-1.

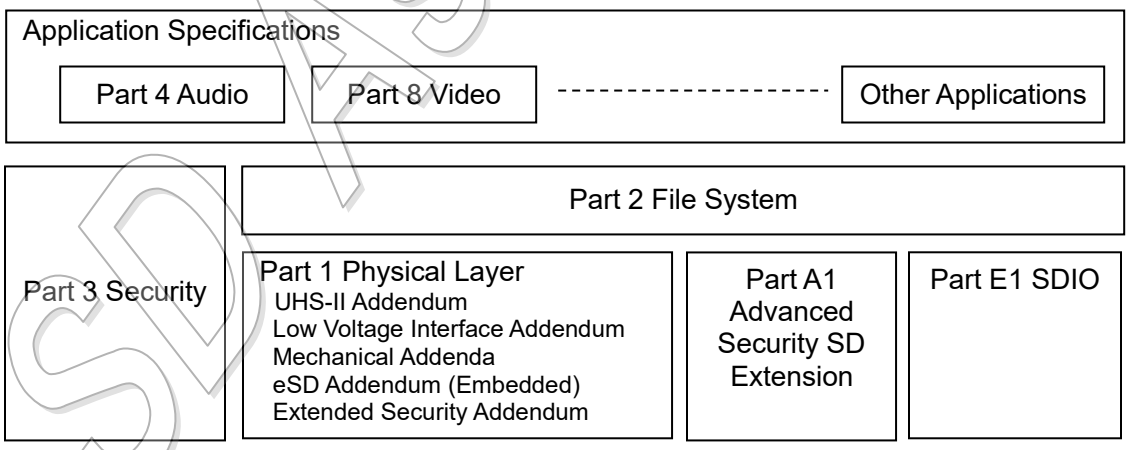


Figure 1-1 : SD Specifications Documentation Structure

- **Audio Specification:**

  This specification, along with other application specifications, describes the specification of a specific application (in this case - Audio Application) and the requirements to implement it.

- **File System Specification:**
  The specification describes the specification of the file format structure of the data saved in the SD Memory Card (in User Area and Protected Area).

- **Security Specification:**
  The specification describes the content protection mechanism and the application-specific commands that support it.

- **Physical Layer Specification (this document):**
  The specification describes the physical interface and the command protocol used by the SD Memory Card. The purpose of the Physical Layer specification is to define the SD Memory Card, its environment, and handling.
  The document is divided into several portions. Chapter 3 gives a general overview of the system concepts. The common SD Memory Card characteristics are described in Chapter 4. As this description defines an overall set of card properties, we recommend using the product documentation in parallel. The card registers are described in Chapter 5.
  Chapter 6 defines the electrical parameters of the SD Memory Card's hardware interface.

  Mechanical Specification described in Chapter 8 in Version 2.00 is moved to the Standard Size Mechanical Addendum.
  There are three mechanical addenda depend on form factors.
  (1) Standard Size Mechanical Addendum
  (2) miniSD Mechanical Addendum
  (3) microSD Mechanical Addendum

  UHS-II Interface Specification is defined by the UHS-II Addendum.
  SD Express Interface Specification is defined by the SD Physical Specification – introduced in Version 7.0.
  The definitions for either UHS-II or SD Express interfaces are independent of a specific form factor unless it is stated as applicable to specific form factor only.
  Un-removable memory device for embedded application is defined by the eSD Addendum.
  TCG security functionality in SD card is defined in the Extended Security Addendum.

  As used in this document, "shall" or "will" denote a mandatory provision of the standard. "Should" denotes a provision that is recommended but is not mandatory. "May" denotes a feature, which may or may not be present–at the option of the implementer–and whose presence does not preclude compliance.

- **Mc-EX Interface Specification:** (This section was added in version 1.10)
  Part A1 of the SD memory card specification (Refer to Figure 1-1) serves as an extension to the SD card Physical Layer Specification and provides all of the definitions required to transfer the Mobile Commerce Extension (Mc-EX) command packets from the Mc-EX host to the Mc-EX enabled SD memory card, and vice versa.

- **SDIO Specification**
  SDIO card and embedded SDIO are specified based on the Physical Layer Specification and modifications and extensions are described in the Part E1 SDIO Specification.

2. System Features

- Targeted for portable and stationary applications
- Capacity of Memory
  (1) Standard Capacity SD Memory Card (SDSC): Up to and including 2 GB
  (2) High Capacity SD Memory Card (SDHC): More than 2GB and up to and including 32GB
  (3) Extended Capacity SD Memory Card (SDXC): More than 32GB and up to and including 2TB
  (4) Ultra Capacity SD Memory Card (SDUC): More than 2TB and up to and including 128TB

- Voltage range:
  High Voltage SD Memory Card – Operating voltage range: 2.7-3.6 V
  UHS-II SD Memory Card – Operating voltage range VDD1: 2.7-3.6 V, VDD2: 1.70-1.95V
  SD Express Memory Card - Operating voltage range VDD1: 2.7-3.6 V, VDD2: 1.70-1.95V and optional VDD3: 1.14-1.30V (operated instead of VDD2 if supported).

- Designed for read-only and read/write cards.
- Bus Speed Mode - using 4 parallel data lines
  (1) Default Speed mode: 3.3V signaling, Frequency up to 25 MHz, up to 12.5 MB/sec
  (2) High Speed mode: 3.3V signaling, Frequency up to 50 MHz, up to 25 MB/sec
  (3) SDR12: UHS-I 1.8V signaling, Frequency up to 25 MHz, up to 12.5MB/sec
  (4) SDR25: UHS-I 1.8V signaling, Frequency up to 50 MHz, up to 25MB/sec
  (5) SDR50: UHS-I 1.8V signaling, Frequency up to 100 MHz, up to 50MB/sec
  (6) SDR104: UHS-I 1.8V signaling, Frequency up to 208 MHz, up to 104MB/sec
  (7) DDR50: UHS-I 1.8V signaling, Frequency up to 50 MHz, sampled on both clock edges, up to 50MB/sec
- Bus Speed Mode - using UHS-II Differential Interface lines
  (1) UHS156: UHS-II RCLK Frequency Range 26MHz - 52MHz, up to 1.56Gbps per lane.
  (2) UHS624: UHS-II RCLK Frequency Range 26MHz- 52MHz, up to 6.24Gbps per lane.
- Bus Speed Mode - using PCIe Differential Interface lines
  (1) PCIe with Gen 3 1 lane - Up to 985MB/s.
  (2) PCIe with Gen 3 2 lanes - Up to 1,969MB/s.
  (3) PCIe with Gen 4 1 lane - Up to 1,969MB/s.
  (4) PCIe with Gen 4 2 lanes - Up to 3,938MB/s.
- Switch function command supports Bus Speed Mode, Command System, Drive Strength, and future functions
- Correction of memory field errors
- Card removal during read operation will never harm the content
- Content Protection Mechanism - Complies with highest security of SDMI standard.
- Password Protection of cards (CMD42 - LOCK_UNLOCK)

- Write Protect feature using mechanical switch
- Built-in write protection features (Permanent Write Protect, Temporary Write Protect and Write Protect Until Power Cycle)
- Card Detection (Insertion/Removal)
- Application specific commands
- Comfortable erase mechanism
- Features of the SD Express card:
  - PCI Express® (PCIe) Gen 3 or Gen 4; 1 lane or 2 lanes
    - Dual simplex point to point serial connection
    - For Gen3x1 bus, two differential I/O (1 RX / 1 TX) of 8Gbps transfer, producing up to 985MB/s for each direction (~1.5% overhead due to 128/130 encoding)
    - For Gen3x2 bus, four differential I/O (2 RX / 2 TX) of 8Gbps transfer, producing up to 1,969MB/s for each direction (~1.5% overhead due to 128/130 encoding)
    - For Gen4x1 bus, two differential I/O (1 RX / 1 TX) of 16Gbps transfer, producing up to 1,969MB/s for each direction (~1.5% overhead due to 128/130 encoding)
    - For Gen4x2 bus, four differential I/O (2 RX / 2 TX) of 16Gbps transfer, producing up to 3,938MB/s for each direction (~1.5% overhead due to 128/130 encoding)
    - Hot plug-in/out support
    - Identified as standard Mass storage controller - NVM Express™ device
  - NVM-Express revision 1.3 or later supported over the PCIe interface
    - Mandatory to support revision 1.3 and optional[3] to support features from revision 1.3a, 1.3b, 1.3c, 1.3d or 1.4.
    - NVMe is light protocol – built for performance
    - All the optional standard features of PCIe and NVMe standards may be implemented by host and/or card if they wish to (up to implementation). Following is a partial list:
      - Host Memory Buffer supported by NVMe
      - NVMe with multi queues and no locking mechanism
  - Legacy SD UHS-I interface is supported for backward compatibility (with limited features as described in Sections 8.1.4, 8.1.5 and 0)

- Protocol attributes of the basic SD communication channel:

SD Memory Card Communication Channel
Six-wire communication channel (clock, command, 4 data lines)
Error-protected data transfer
Single or Multiple block oriented data transfer

- SD Express card supports NVMe(1) protocol (revision 1.3 or later(3)) over the PCIe(2) interface
- SD Memory Card Form-factor
  There are three Part 1 mechanical addenda as follows:
  Standard Size SD Memory Card: Specified in "Part 1 Standard Size SD Card Addendum"
  miniSD Memory Card: Specified in "Part 1 miniSD Card Addendum"
  microSD Memory Card: Specified in "Part 1 microSD Card Addendum"
- Standard Size SD Memory Card thickness is defined as both 2.1 mm (normal) and 1.4 mm (Thin SD Memory Card).
- Boot Functionalities including boot partitions and Fast Boot
- TCG (Trusted Computing Group) security including MBR Shadowing.
- RPMB (Replay Protected Memory Block)

(1) NVM Express™ (word mark) and NVMe™ (word mark) are trademarks of NVM Express, Inc.
NVM Express, Inc. defines the NVM Express standard and specifications.
Contact NVM Express, Inc. for further information URL: http://nvmexpress.org/.
(2) PCI-SIG®, PCIe® and PCI Express® are registered trademarks of PCI-SIG.
The PCI-SIG defines the PCI Express standard and specifications.
Contact the PCI-SIG for further information URL: https://pcisig.com/.
PCI-SIG®, PCIe® and PCI Express® are registered trademarks of PCI-SIG.
(3) It is recommended for SD Express 8.0 hosts to support NVM Express revision 1.3d or 1.4 to avoid any compatibility issues with SD Express cards using earlier revisions.
(4) The TCG defines the TCG security standard and specifications.
Contact the TCG for further information URL: https://www.trustedcomputinggroup.org/.

3. SD Memory Card System Concept

Description here is a blank in the Simplified Specification.

3.1 Read-Write Property

In terms of read-write property, two types of SD Memory Cards are defined:

- Read/Write (RW) cards (Flash, One Time Programmable - OTP, Multiple Time Programmable - MTP). These cards are typically sold as blank (empty) media and are used for mass data storage, end user video, audio or digital image recording
- Read Only Memory (ROM) cards. These cards are manufactured with fixed data content. They are typically used as a distribution media for software, audio, video etc.

3.2 Supply Voltage

In terms of operating supply voltage, two types of SD Memory Cards are defined:

- High Voltage SD Memory Cards that can operate the voltage range of 2.7-3.6 V.
- UHS-II SD Memory Card that can operate the voltage ranges VDD1: 2.7-3.6 V, VDD2: 1.70-1.95V
- Low Voltage Signaling (LVS) Cards that can operate the supply voltage range of 2.7-3.6V, but signaling voltage range of the non-differential interface is 1.70-1.95V.
- SD Express Memory Card that can operate under voltage range VDD1: 2.7-3.6 V, VDD2:1.70-1.95V (mandatory) and optional VDD3: 1.14-1.30V that may be used instead VDD2, if supported.

3.3 Card Capacity

3.3.1 User Area, Protected Area and Boot Partitions

Non CPRM Card means Regular Writeable SD Card (SDSC/SDHC/SDXC/SDUC) that does not support the CPRM security. For Non CPRM Cards, the Protected Area cannot be accessed.
SD Memory Card supporting CPRM has two accessible independent areas: User Area and Protected Area. User Area is main memory area and Protected Area can be accessed by the authentication defined by the Part 3 Security Specification. Card Capacity means the sum of User Area Capacity and Protected Area Capacity, and does not include Boot Partitions. Since SDUC does not support security, the size of Protected Area is set to 0 and it cannot be accessed.

3.3.2 Card Capacity Classification

In terms of card capacity, three types of SD Memory Cards are defined:

- Standard Capacity SD Memory Card (SDSC) supports capacity up to and including 2 G bytes (2^{31} bytes). All versions of the Physical Layer Specifications define the Standard Capacity SD Memory Card.
- High Capacity SD Memory Card (SDHC) supports capacity more than 2 G bytes (2^{31} bytes) up to and including 32 G bytes and is defined from the Physical Layer Specification Version 2.00.
- Extended Capacity SD Memory Card (SDXC) supports more than 32 G bytes (2^{35} bytes) up to and including 2TB.
- Ultra Capacity SD Memory Card (SDUC) supports more than 2 T bytes (2^{41} bytes) up to and including 128TB.

Note:

1. The Part 1 Physical Layer Specification Version 3.00 or later and Part 2 File System Specification Version 3.00 or later allow Standard Capacity SD Memory Cards to have capacity up to and including 2 GB, High Capacity SD Memory Cards to have capacity up to and including 32 GB and Extended Capacity SD Memory Card to have capacity up to 2 TB.
2. Hosts that can access (read and/or write) SD Memory Cards with a capacity greater than 2 GB and up to and including 32 GB, shall also be able to access SD Memory Cards with a capacity of 2 GB or less.

3. Hosts that can access (read and/or write) SD Memory Cards with a capacity greater than 32 GB and up to 2 TB, shall also be able to access SD Memory Cards with a capacity of 32 GB or less.

3.4 Speed Class

Five Speed Classes are defined and indicate minimum performance of the cards

- Class 0 - These class cards do not specify performance. It includes all the legacy cards prior to the Physical Layer Specification Version 2.00, regardless of its performance
- Class 2 is more than or equal to 2 MB/sec performance (Default Speed Mode)
- Class 4 is more than or equal to 4 MB/sec performance (Default Speed Mode)
- Class 6 is more than or equal to 6 MB/sec performance (Default Speed Mode)
- Class 10 is more than or equal to 10 MB/sec performance (High Speed Mode)

Note that the unit of performance [MB/sec] indicates 1000x1000 [Byte/sec] while the unit of data size [MB] indicates 1024x1024 [Byte]. This is because the maximum SD Bus speed is specified by the maximum SD clock frequency (25 [MB/sec] = 25000000 [Byte/sec] at 50 MHz) and data size is based on memory boundary (power of 2).

3.5 Bus Topology

This section is a blank in the Simplified Specification.

3.6 Bus Protocol

3.6.1 SD Bus Protocol

Communication over the SD bus is based on command and data bit streams that are initiated by a start bit and terminated by a stop bit.

- **Command**: a command is a token that starts an operation. A command is sent from the host either to a single card (addressed command) or to all connected cards (broadcast command). A command is transferred serially on the CMD line.
- **Response**: a response is a token that is sent from an addressed card, or (synchronously) from all connected cards, to the host as an answer to a previously received command. A response is transferred serially on the CMD line.
- **Data**: data can be transferred from the card to the host or vice versa. Data is transferred via the data lines.

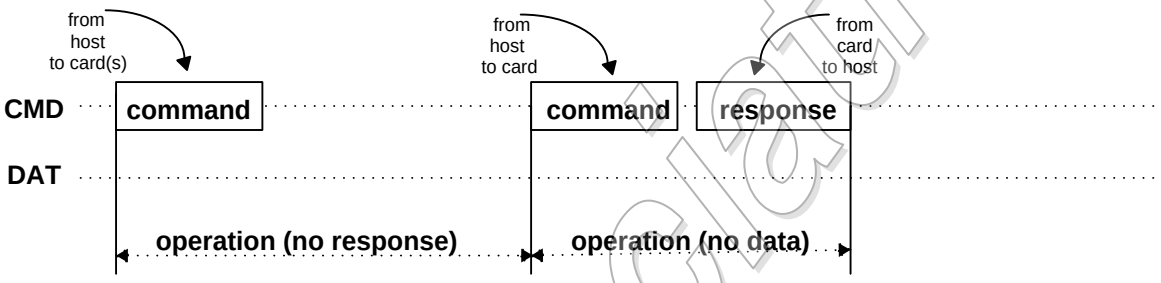


Figure 3-4 : "no response" and "no data" Operations

Card addressing is implemented using a session address, assigned to the card during the initialization phase. The structure of commands, responses and data blocks is described in Chapter 4. The basic transaction on the SD bus is the command/response transaction (refer to Figure 3-4). This type of bus transaction transfers their information directly within the command or response structure. In addition, some operations have a data token.

Data transfers to/from the SD Memory Card are done in blocks. Data blocks are always succeeded by CRC bits. Single and multiple block operations are defined. Note that the Multiple Block operation mode is better for faster write operation. A multiple block transmission is terminated when a stop command follows on the CMD line. Data transfer can be configured by the host to use single or multiple data lines.

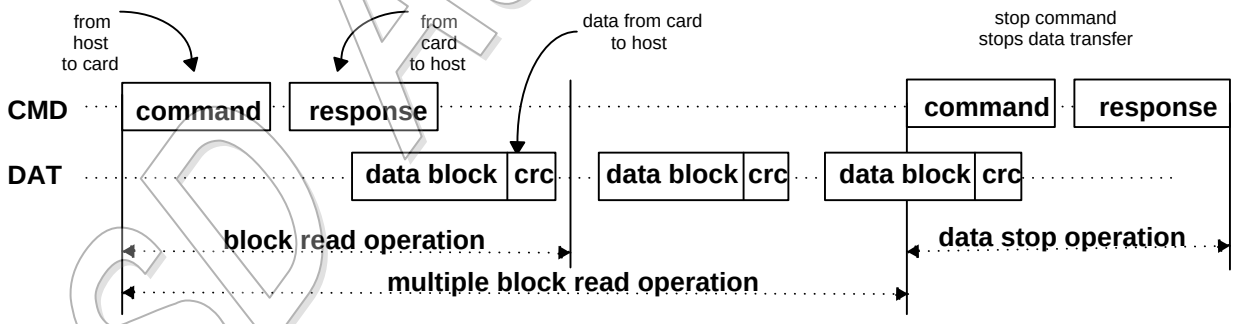


Figure 3-5 : (Multiple) Block Read Operation

The block write operation uses a simple busy signaling of the write operation duration on the DAT0 data line (see Figure 3-6) regardless of the number of data lines used for transferring the data.

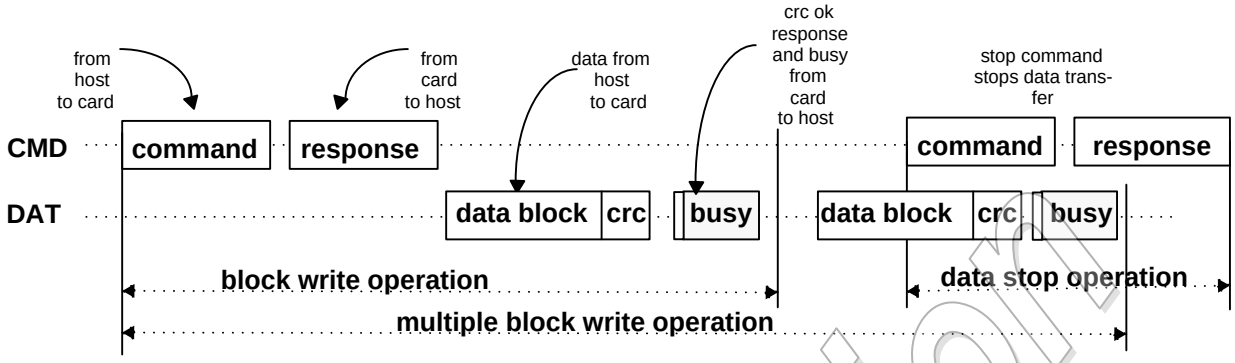


Figure 3-6 : (Multiple) Block Write Operation

Command tokens have the following coding scheme:

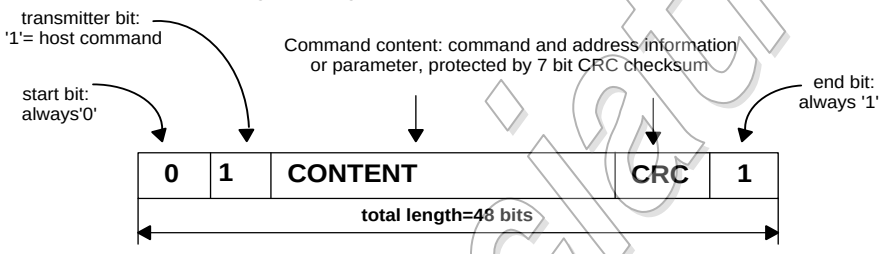


Figure 3-7 : Command Token Format

Each command token is preceded by a start bit (0) and succeeded by an end bit (1). The total length is 48 bits. Each token is protected by CRC bits so that transmission errors can be detected and the operation may be repeated.

Response tokens have one of four coding schemes, depending on their content. The token length is either 48 or 136 bits. The detailed commands and response definition is given in Section 4.7. The CRC protection algorithm for block data is a 16-bit CCITT polynomial. All allowed CRC types are described in Section 4.5.

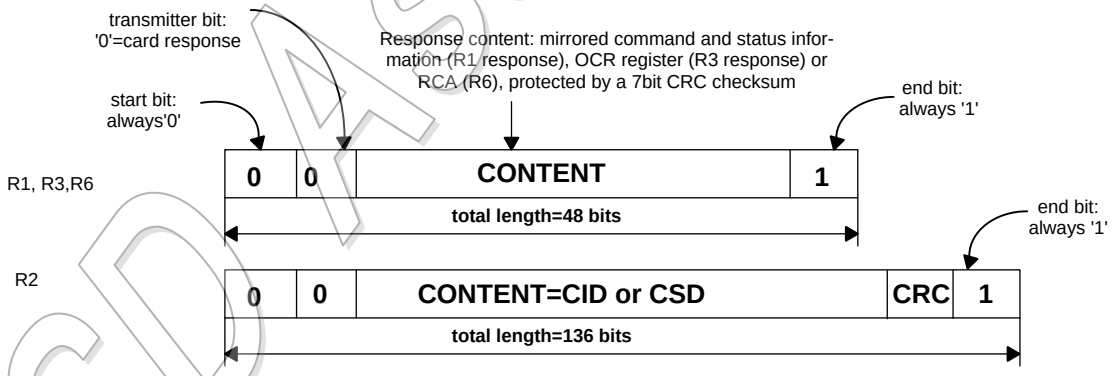


Figure 3-8 : Response Token Format

In the CMD line the Most Significant Bit (MSB) is transmitted first, the Least Significant Bit (LSB) is the last.

When the wide bus option is used, the data is transferred 4 bits at a time (refer to Figure 3-10). Start and end bits, as well as the CRC bits, are transmitted for every one of the DAT lines. CRC bits are calculated and checked for every DAT line individually. The CRC status response and Busy indication will be sent by the card to the host on DAT0 only (DAT1-DAT3 during that period are don't care).

There are two types of Data packet format for the SD card.

(1) Usual data (8-bit width): The usual data (8-bit width) are sent in LSB (Least Significant Byte) first, MSB (Most Significant Byte) last sequence. But in the individual byte, it is MSB (Most Significant Bit) first, LSB (Least Significant Bit) last.

(2) Wide width data (SD Memory Register): The wide width data is shifted from the MSB bit.

1. Data Packet Format for Usual Data (8-bit width)

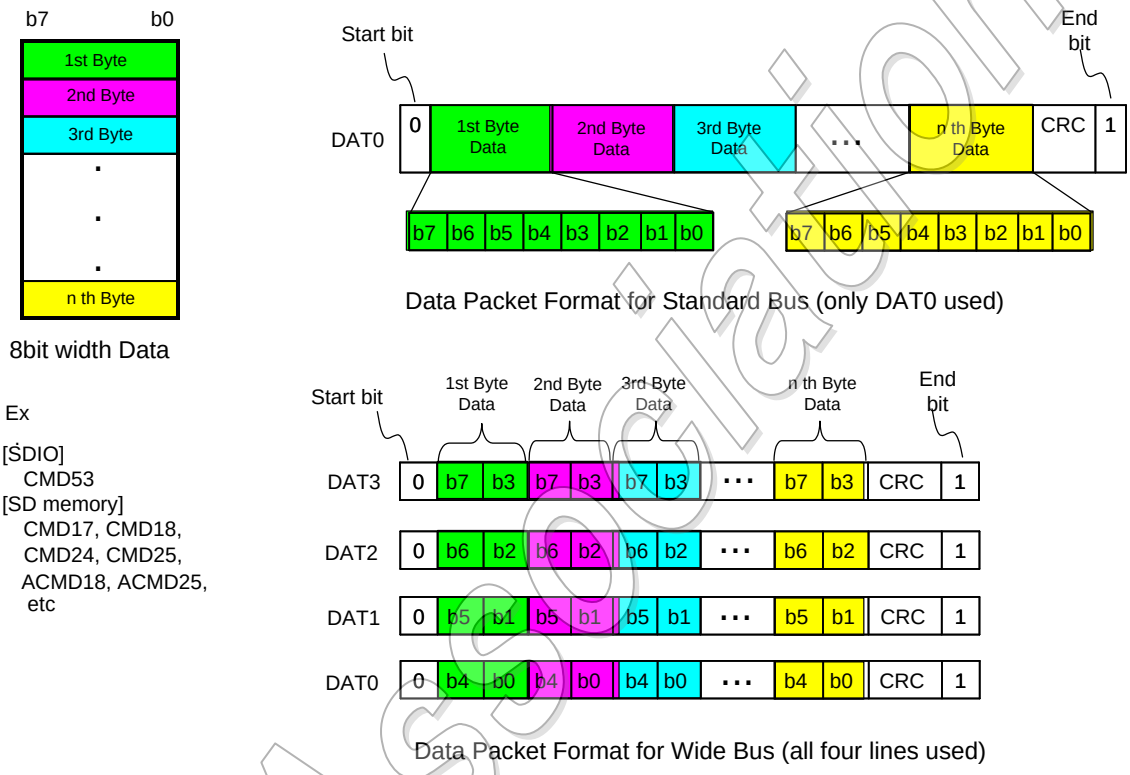


Figure 3-9 : Data Packet Format - Usual Data

2. Data Packet Format for Wide Width Data (Ex. ACMD13)

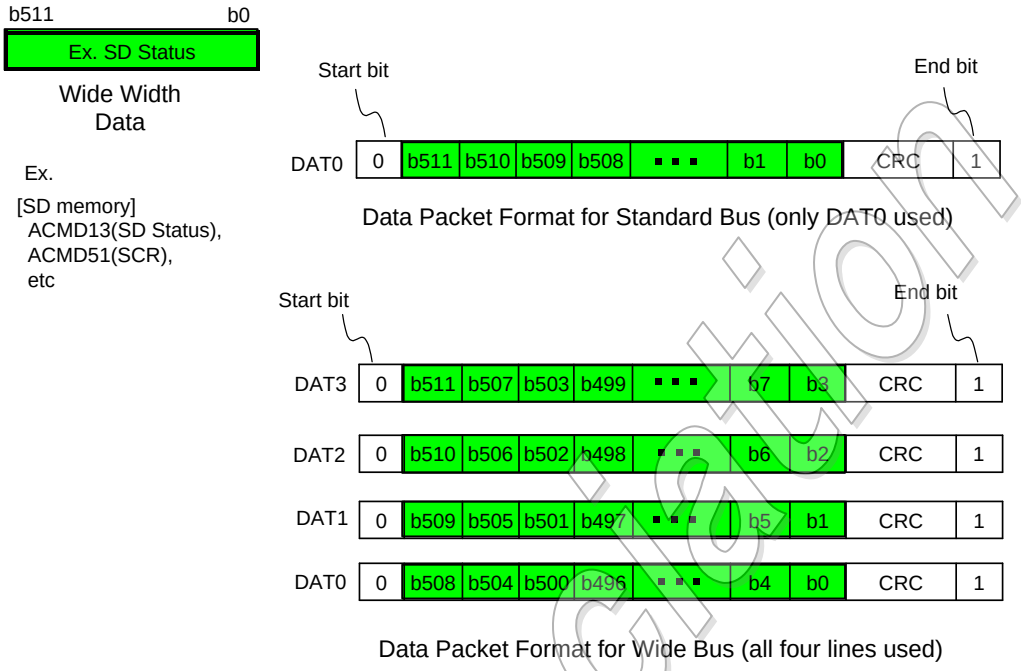


Figure 3-10 : Data Packet Format - Wide Width Data

3.6.2 SPI Bus Protocol

Details of the SPI Bus protocol are described in Chapter 7.

3.6.3 UHS-II Bus Protocol

UHS-II Bus protocol is defined in the UHS-II Addendum.

3.6.4 PCIe/NVMe Bus Protocol

The PCIe and NVMe protocols are defined by the PCIe and NVMe specifications listed in appendix A.2 except for the details captured in this document.

3.7 SD Memory Card–Pins and Registers

3.7.1 SD Bus Pin Assignment

The SD Memory Card has the form factor 24 mm x 32 mm x 2.1 mm or 24 mm x 32 mm x 1.4 mm.

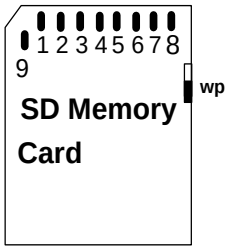


Figure 3-11 : SD Memory Card Shape and Interface (Top View)

Figure 3-11 shows the general shape of Standard Size and interface contacts of the SD Memory Card. The detailed physical dimensions and mechanical description are given in Part 1 Mechanical Addendum. MicroSD and miniSD form factors are available as well. Their physical dimensions, mechanical description as well as pin assignment are given in Part 1 microSD Card Addendum and Part 1 miniSD Card Addendum.

Table 3-1 defines the card contacts:

Pin #	SD Mode			SPI Mode		
	Name	**Type[1]**	**Description**	**Name**	**Type[1]**	**Description**
1	CD/DAT3[2]	I/O/PP[3]	Card Detect/ Data Line [Bit 3]	CS	I[3]	Chip Select (neg true)
2	CMD	I/O/PP	Command/Response	DI	I	Data In
3	VSS1	S	Supply voltage ground	VSS	S	Supply voltage ground
4	VDD	S	Supply voltage	VDD	S	Supply voltage
5	CLK	I	Clock	SCLK	I	Clock
6	VSS2	S	Supply voltage ground	VSS2	S	Supply voltage ground
7	DAT0	I/O/PP	Data Line [Bit 0]	DO	O/PP	Data Out
8	DAT1[4]	I/O/PP	Data Line [Bit 1]	RSV		
9	DAT2[5]	I/O/PP	Data Line [Bit 2]	RSV		

1) S: power supply; I: input; O: output using push-pull drivers; PP: I/O using push-pull drivers;
2) The extended DAT lines (DAT1-DAT3) are input on power up. They start to operate as DAT lines after SET_BUS_WIDTH command. The Host shall keep its own DAT1-DAT3 lines in input mode, as well, while they are not used.
3) At power up this line has a 50KOhm pull up enabled in the card. This resistor serves two functions Card detection and Mode Selection. For Mode Selection, the host can drive the line high or let it be pulled high to select SD mode. If the host wants to select SPI mode it should drive the line low. For Card detection, the host detects that the line is pulled high. This pull-up should be disconnected by the user, during regular data transfer, with SET_CLR_CARD_DETECT (ACMD42) command
4) DAT1 line may be used as Interrupt Output (from the Card) in SDIO mode during all the times that it is not in use for data transfer operations (refer to "SDIO Card Specification" for further details).
5) DAT2 line may be used as Read Wait signal in SDIO mode (refer to "SDIO Card Specification" for further details).

Table 3-1 : SD Memory Card Pad Assignment

Each card has a set of information registers (see also Chapter 5 in the Physical Layer Specification):

Name	Width	Description
CID	128	Card identification number; card individual number for identification (See 5.2). **Mandatory**.
RCA[1]	16	Relative card address; local system address of a card, dynamically suggested by the card and approved by the host during initialization (See 5.4). **Mandatory**.
DSR	16	Driver Stage Register; to configure the card's output drivers (See 5.5). **Optional.**
CSD	128	Card Specific Data; information about the card operation conditions (See 5.3). **Mandatory**
SCR	64	SD Configuration Register; information about the SD Memory Card's Special Features capabilities (See 5.6). **Mandatory**
OCR	32	Operation conditions register (See 5.1). **Mandatory.**
SSR	512	SD Status; information about the card proprietary features (See 4.10.2). **Mandatory**
CSR	32	Card Status; information about the card status (See 4.10.1). **Mandatory**

(1) RCA register is not used (available) in SPI mode

Table 3-2 : SD Memory Card Registers

The host may reset the cards by switching the power supply off and on again. Each card shall have its own power-on detection circuitry that puts the card into a defined state after the power-on. No explicit reset signal is necessary. The cards can also be reset by sending the GO_IDLE (CMD0) command.

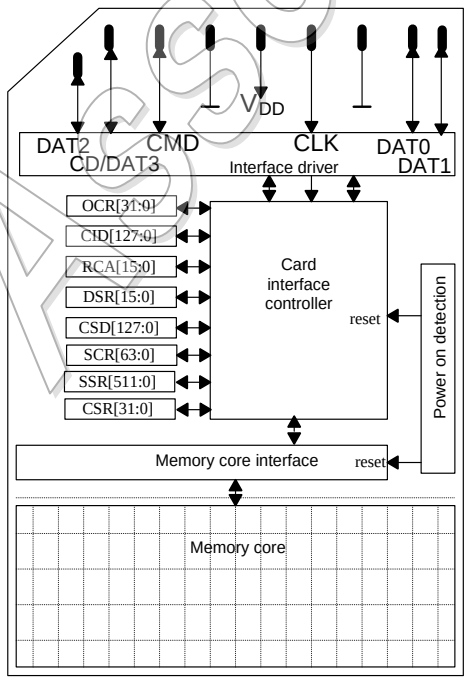


Figure 3-12 : SD Memory Card Architecture

3.7.2 UHS-II Pin Assignment

UHS-II Card shape is the same as SD Cards and UHS-II Interface is assigned to pads on the second row as well as sharing a few of the pads in first row.

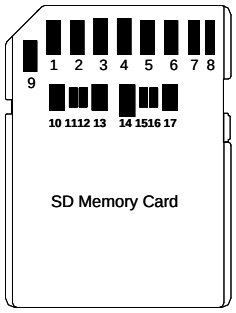


Figure 3-13 : UHS-II Card Shape and Interface (Top View)

Figure 3-13 shows the shape of Standard Size and interface contacts of the UHS-II SD Memory Card. The detailed physical dimensions and mechanical description are given in Part 1 Mechanical Addendum. UHS-II in microSD card form factor is available as well. Its physical dimensions, mechanical description as well as pin assignment are given in Part 1 microSD Card Addendum..
Table 3-3 defines the contacts for UHS-II. SD bus contact Pins 7 and 8 are used for RCLK. The first row contacts in non UHS-II mode are equivalent to Table 3-1. Regarding PHY I/O Type, refer to UHS-II Addendum for more details.

Pin #	Name	Type	Description
4	VDD1	Supply voltage	2.7V to 3.6V
7	RCLK+	Differential Signaling: Input	Clock Input
8	RCLK-	Differential Signaling: Input	Clock Input
10	VSS3	Ground	
11	D0+	Differential Signaling: Input (FD) / Bidirectional (HD)	Input in default
12	D0-	Differential Signaling: Input (FD) / Bidirectional (HD)	Input in default
13	VSS4	Ground	
14	VDD2	Supply Voltage 2	1.70V to 1.95V
15	D1-	Differential Signaling: Output (FD) / Bidirectional (HD)	Output in default
16	D1+	Differential Signaling: Output (FD) / Bidirectional (HD)	Output in default
17	VSS5	Ground	

Table 3-3 : UHS-II Interface Pad Assignment

UHS-II Card shall not drive unused lines of SD I/F in UHS-II mode. (CLK, CMD and DAT[3:2]. DAT[2] may be used as interrupt line in case of UHS-II SDIO Card. In this case, card controls DAT[2] level.)
Host shall not leave these unused lines floating, but keep them at a defined high or low level. How to keep line level is dependent on host implementation. For example, use pull-up resistor or host drives the lines to low level without providing pull-up voltage. As DAT[1:0] are used for providing RCLK, individual line control is required to use pull-up method for CMD and DAT[3:2]. CLK (without pull-up resistor) should be driven to low.
In case of entering hibernate mode, unused lines shall be set to low before turning off VDD1.

3.7.3 1-Lane SD Express Pin Assignment

The SD Express Card shape is the same as SD Cards; the PCIe/NVMe Interface is assigned to pads on the second row as well as sharing some of the pads in the first row.

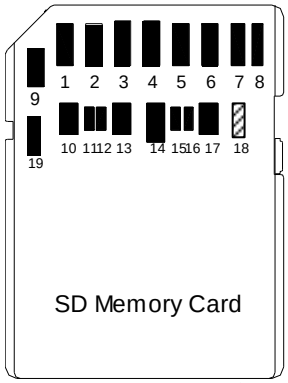


Figure 3.7.3-14 : 1-Lane SD Express Card Shape and Interface (Top View)

Figure 3.7.3-14 shows the shape of 1-Lane SD Express card and its interface contacts for standard size SD card form factor. The detailed physical dimensions and mechanical description are given in Part 1 Mechanical Addendum.
SD Express in microSD card form factor is available as well. Its physical dimensions, mechanical description as well as pin assignment are given in Part 1 microSD Card Addendum.

Table 3.7.3-1 defines the contacts for PCIe interface in case of supporting PCIe 1 lane. SD Express card shall support the basic SD interface allowing operation of the card in non-PCIe mode. The first row contacts is used by the basic SD mode (non-PCIe mode) and are equivalent to Table 3-1. Note that SD bus contact Pins 7 and 8 are used for REFCLK of PCIe interface and contact Pins 9 and 1 are used for CLKREQ# and PERST# respectively.

Pin #	Name	Type	Description
1	PERST#	Input signal (active low)	PE-Reset is a functional reset to the card as defined by the PCIe Mini CEM specification.
4	VDD1	Supply voltage	2.7V to 3.6V
7	REFCLK+	Differential Signaling: Input	Clock Input
8	REFCLK-	Differential Signaling: Input	Clock Input
9	CLKREQ#	I/O (active low, open drain)	Reference clock request signal. Also used by L1 PM substates
10	VSS3	Ground	
11	PCIe TX+	Differential Signaling	Card Input
12	PCIe TX-	Differential Signaling	Card Input
13	VSS4	Ground	
14	VDD2	Supply Voltage 2	1.70V to 1.95V
15	PCIe RX-	Differential Signaling	Card Output
16	PCIe RX+	Differential Signaling	Card Output
17	VSS5	Ground	
18(1)	VDD3	Supply Voltage 3	1.14V to 1.30V (optional)
19(2)	VDD1a	Supply Voltage 4	2.7V to 3.6V

Note: (1) Pin #18 can be implemented when card supports VDD3. VDD3 is reserved for Standard Size SD Express.
(2) Pin #19 is available only when SD Express card is compliant to Part 1 Physical Layer Specification Version 8.00 with PCIe Gen4 (Pin #19 does not exist for SD Express Card compliant to Part 1 Physical Layer Specification Version 7.XX with PCIe Gen3).

Table 3.7.3-1 : 1-Lane SD Express Interface Pad Assignment

SD Express Card shall not drive unused lines of SD I/F in PCIe mode.
Host shall not leave these unused lines floating, but keep them at a defined high or low level. How to keep line level is dependent on host implementation. CLK (without pull-up resistor) should be driven to low.

Pin functionality of the out-of-band signalling of the PCIe interface - CLKREQ# and PERST# is defined in PCIe standard. The existing card detection switch mechanism described in the Mechanical Addendum may be used by hosts as the Card Presence Detect in a similar fashion as PRSNT2# and PRSNT1# lines used for HW hot plug-in/out detection defined in PCIe specifications.

If such a card detection switch is used logic ‘1’ to ‘0’ shall be detected by host when card is inserted into SD PCIe host and transition from ‘0’ to ‘1’ when SD Express card is removed.

3.7.4 2-Lane SD Express Pin Assignment

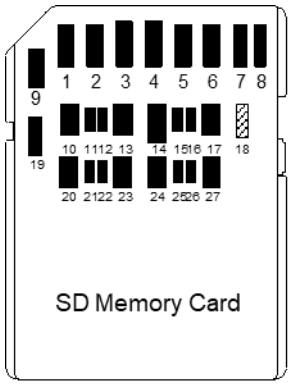


Figure 3.7.4-1 : 2-Lane SD Express Card Shape and Interface (Top View)

Figure 3.7.4-1 shows the shape of 2-Lane SD Express card and its interface contacts for standard size SD card form factor. The detailed physical dimensions and mechanical description are given in Part 1 Mechanical Addendum.

Table 3.7.4-1 defines the contacts for PCIe interface in case of having PCIe 2 lanes. 2-Lane SD Express card also shall support the basic SD interface allowing operation of the card in non-PCIe mode.

Pin #	Name	Type	Description
1	PERST#	Input signal (active low)	PE-Reset is a functional reset to the card as defined by the PCIe Mini CEM specification.
4	VDD1	Supply voltage	2.7V to 3.6V
7	REFCLK+	Differential Signaling: Input	Clock Input
8	REFCLK-	Differential Signaling: Input	Clock Input
9	CLKREQ#	I/O (active low, open drain)	Reference clock request signal. Also used by L1 PM substates
10	VSS3	Ground	
11	PCIe TX0+	Differential Signaling	Card Input; lane 0
12	PCIe TX0-	Differential Signaling	Card Input; lane 0
13	VSS4	Ground	
14	VDD2	Supply Voltage 2	1.70V to 1.95V
15	PCIe RX0-	Differential Signaling	Card Output; lane 0
16	PCIe RX0+	Differential Signaling	Card Output; lane 0
17	VSS5	Ground	
18(1)	VDD3	Supply Voltage 3	1.14V to 1.30V (optional)
19(2)	VDD1a	Supply Voltage 4	2.7V to 3.6V (mandatory)
20	VSS6	Ground	
21	PCIe TX1+	Differential Signaling	Card Input; lane 1
22	PCIe TX1-	Differential Signaling	Card Input; lane 1
23	VSS7	Ground	
24	VDD2a	Supply Voltage 5	1.70V to 1.95V
25	PCIe RX1-	Differential Signaling	Card Output; lane 1
26	PCIe RX1+	Differential Signaling	Card Output; lane 1
27	VSS8	Ground	

Note: (1) Pin #18 can be implemented when card supports VDD3. VDD3 is reserved for Standard Size SD Express.
(2) Pin #19 and #24 are mandatory in 2-Lane SD Express card.

Table 3.7.4-1 : 2-Lane SD Express Interface Pad Assignment

SD I/F treatment in PCIe mode, out-of-band signals functionality and card detection recommendation given in Section 3.7.3 are applicable for 2-Lane SD Express Card.

3.8 ROM Card

ROM Card is defined as read only memory which meets following requirements. A permanent or temporary write protected writable SD memory card does not belong to this category.

3.8.1 Register Setting Requirements

Table 3-4 shows register setting requirements for ROM Card.

Register	Field	Value	Comment
SD Status	SD_CARD_TYPE	0001h	SD ROM Card
CSD	CCC bit 4	0	Class4 block write
	CCC bit 5	0	Class5 erase
	CCC bit 6	0	Class6 write protection
	CCC bit 7	0 or 1	Class7 lock card
	PERM_WRITE_PROT	1	Permanent Write Protect
SCR	SD_SECURITY	0 or 2 or 3 or 4	Security is optional.

Table 3-4 : Register Setting Requirements for ROM Card

3.8.2 Unsupported Commands

The ROM Card shall treat following commands as unsupported and illegal command.

CMD24, CMD25, CMD27, CMD28, CMD29, CMD30, CMD32, CMD33, CMD38

3.8.3 Optional Commands

The ROM Card can support following commands as optional command.

CMD42, security commands

- If CMD42 is not supported, bit 7 of CCC shall be set to 0. CMD42 is treated as illegal command.
- When ROM card supports CMD42, "Unlocking the card" and "Locking the card" functions shall be supported by presetting the password. LOCK_UNLOCK_FAILED is indicated when receiving the other unsupported functions of CMD42.
- If security is not supported, SD_SECURITY shall be set to 0. The security commands are treated as illegal command.
- ROM card does not support write and erase to the protected area. Refer to Part 3 Security Specification about security command support of ROM card.

3.8.4 WP Switch

A full-size ROM card does not have WP Switch. Refer to Figure 3-8 in the Part 1 Standard Size SD Card Mechanical Addendum Ver1.00.

3.9 Ultra High Speed Phase I (UHS-I) Card

UHS-I provides up to 104MB/sec performance on 4-bit SD bus with the single end driver interface. Card form factor is the same and existing connector can be used.

3.9.1 UHS-I Card Operation Modes

- DS - Default Speed up to 25MHz 3.3V signaling
- HS - High Speed up to 50MHz 3.3V signaling
- SDR12 - SDR up to 25MHz 1.8V signaling
- SDR25 - SDR up to 50MHz 1.8V signaling
- SDR50 - SDR up to 100MHz 1.8V signaling
- SDR104 - SDR up to 208MHz 1.8V signaling
- DDR50 - DDR up to 50MHz 1.8V signaling

Note: 1.8V signal timings are different from those of 3.3V.

3.9.2 UHS-I Card Types

UHS-I supports two card Types:

- UHS50
- UHS104

UHS-I is not applied to SDSC card but can be applied to SDHC, SDXC and SDUC card.

Figure 3-14 and Figure 3-15 show UHS-I supported modes.
DDR50 is mandatory for microSD form factor and optional for Standard size SD form factor

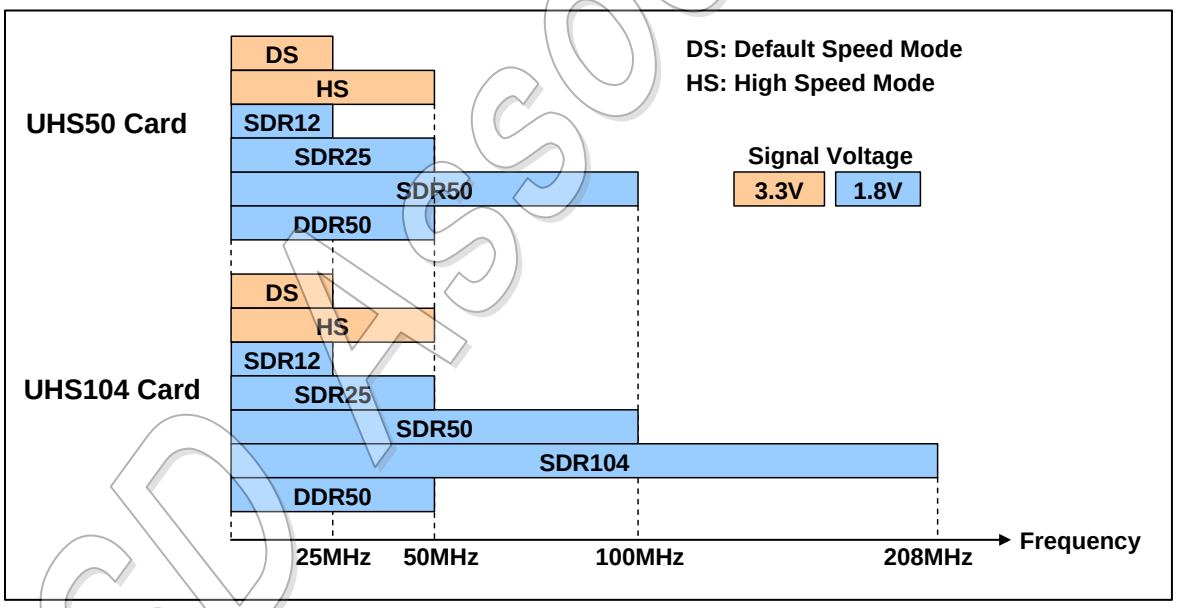


Figure 3-14 : UHS-I Card Type Modes of Operation versus Frequency Range

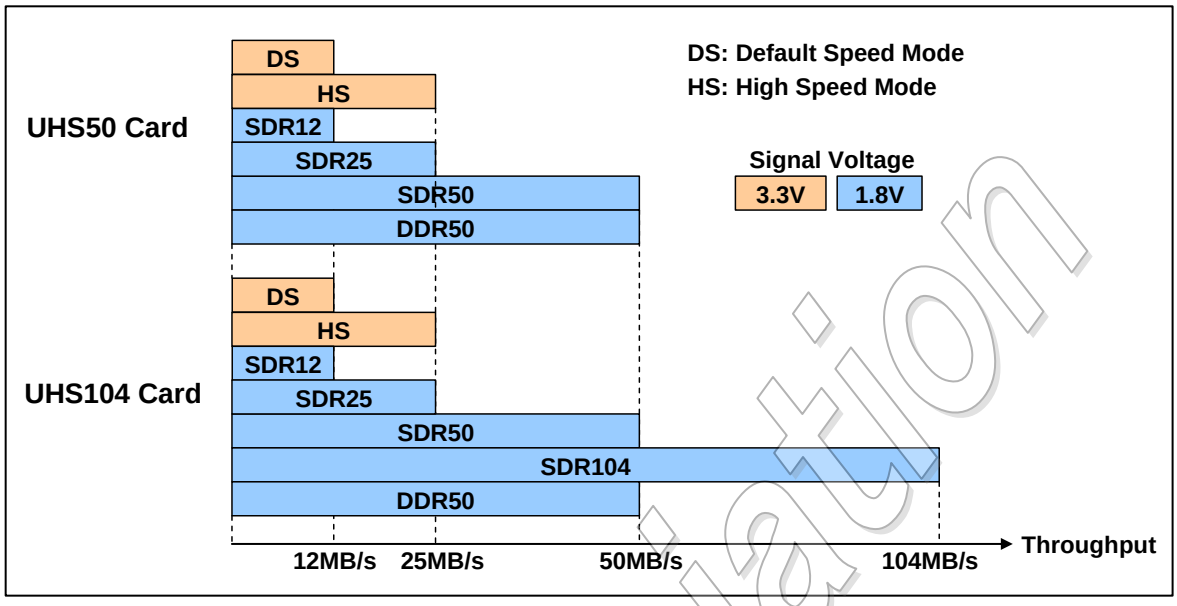


Figure 3-15 : UHS-I Card Type Modes of Operation versus Throughput

3.9.3 UHS-I Host and Card Combination

Host may use SDR50, DDR50 and SDR104 modes with either UHS50 Card or UHS104 Card.
Table 3-5 shows usable UHS performance depends on the combination of host and card. UHS-I for removable card is presumed that one card is connected to a SD bus. Maximum performance of up to 104MB/s is possible only if host supports SDR104 mode and card is UHS104 Card (supports SDR104 mode). If card is a UHS50 Card or if host doesn't support SDR104 mode, performance is limited to 50MB/s (SDR104 mode cannot be used).
Host may use DDR50 mode with UHS50 Card and UHS104 Card in microSD form factors.

Host types:

SDR-FD – SDR signaling, fixed-delay (can't use tuning)
SDR-VD – SDR signaling, variable-delay (can use tuning)
DDR – DDR signaling

Host type / Card type	HOST-SDR-FD (SDR, fixed-delay)	HOST-SDR-VD (SDR, variable-delay)	HOST-DDR (DDR)
UHS50 card microSD	SDR50 ≤ 100MHz	SDR50 ≤ 100MHz + tuning	DDR50 ≤ 50MHz
UHS104 card microSD	SDR50 ≤ 100MHz	SDR104 ≤ 208MHz + tuning	DDR50 ≤ 50MHz
UHS50 card Full-size SD	SDR50 ≤ 100MHz	SDR50 ≤ 100MHz + tuning	Optional
UHS104 card Full-size SD	SDR50 ≤ 100MHz	SDR104 ≤ 208MHz + tuning	Optional

Table 3-5 : UHS-I Host and Card Combinations

3.9.4 UHS-I Bus Speed Modes Selection Sequence

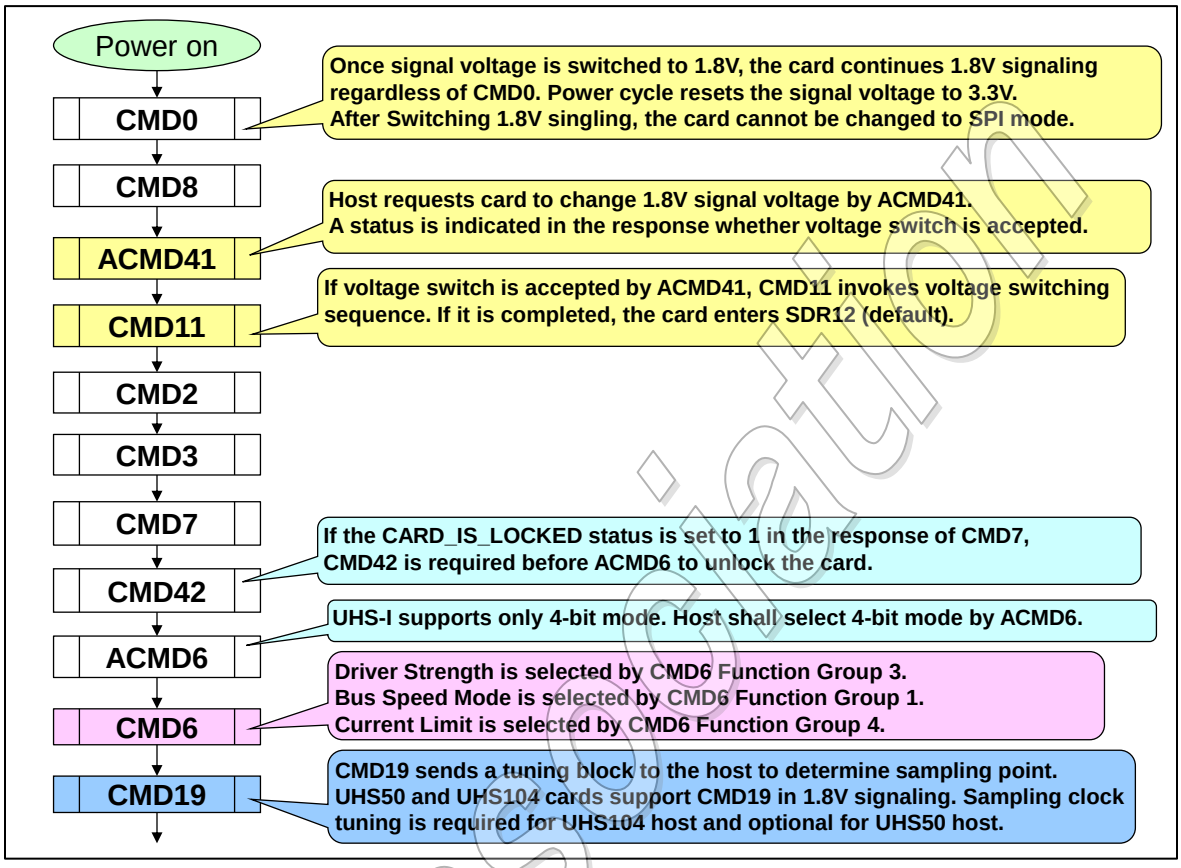


Figure 3-16 : Command Sequence to Use UHS-I

Figure 3-16 shows command sequence to use a UHS-I. After power cycle, card is in 3.3V signaling mode. The first CMD0 selects the bus mode; SD mode or SPI mode. 1.8V signaling mode can be entered only in SD mode. Once the card enters 1.8V signaling mode, the card cannot be switched to SPI mode or 3.3V signaling without power cycle. If the card receives CMD0, card returns to Idle state but still work with SDR12 timing. UHS-I is provided in SD mode but not in SPI mode.

As higher bus speed requires low level signaling, UHS-I adopts 1.8V signaling level for SDR50, DDR50 and SDR104 modes. Still card is supplied with 3.3V by the host and 1.8V signaling level for SDCLK, CMD and DAT[3:0] lines is converted from 3.3V power line. To avoid voltage mismatch between host and card, signaling level is changed by voltage switch sequence at the initialization. The host and card communicate using ACMD41 whether host and card support 1.8V signaling mode. Support of 1.8V signaling both host and card means UHS-I can be used. CMD11 invokes the voltage switch sequence. The card enters UHS-I mode and card input and output timings are changed (SDR12 in default) when the voltage switch sequence is completed successfully. (Refer to Section 4.2.4 for more detail.)

Only 4-bit bus mode is supported in UHS-I except CMD42. If the card is locked, host needs to unlock the card by CMD42 in 1-bit mode and then needs to issue ACMD6 to change 4-bit bus mode. Operating in 1-bit mode is not assured.

Host can choose suitable output driver strength by CMD6 Function Group 3.

Host can choose one of UHS-I modes by CMD6 Function Group 1. Each UHS-I mode is specified by the maximum frequency, sampling edges (rising-only or both) and maximum Power consumption for compatibility with existing cards. Host can choose one of UHS-I mode depending on capability of generating SDCLK frequency and capacity of power supply host supported.

CMD19 can be executed in transfer state of 1.8V signaling mode while the card is unlocked. The other case, CMD19 is treated as illegal command.

3.9.5 UHS-I System Block Diagram

Figure 3-17 shows a typical UHS-I host system that supports removable cards. Host has clock generator which supplies SDCLK to the card. In case of write operation, as clock direction and data direction is the same, write data can be transferred synchronized with SDCLK regardless of transmission line delay. In case of read operation, as clock direction and data direction is opposite, read data host received is delayed by round-trip delay, output delay and latency of host and card. So receiving data is the most critical for the host. Therefore, host needs to have sampling clock generator to receive response, CRC status and read data block.

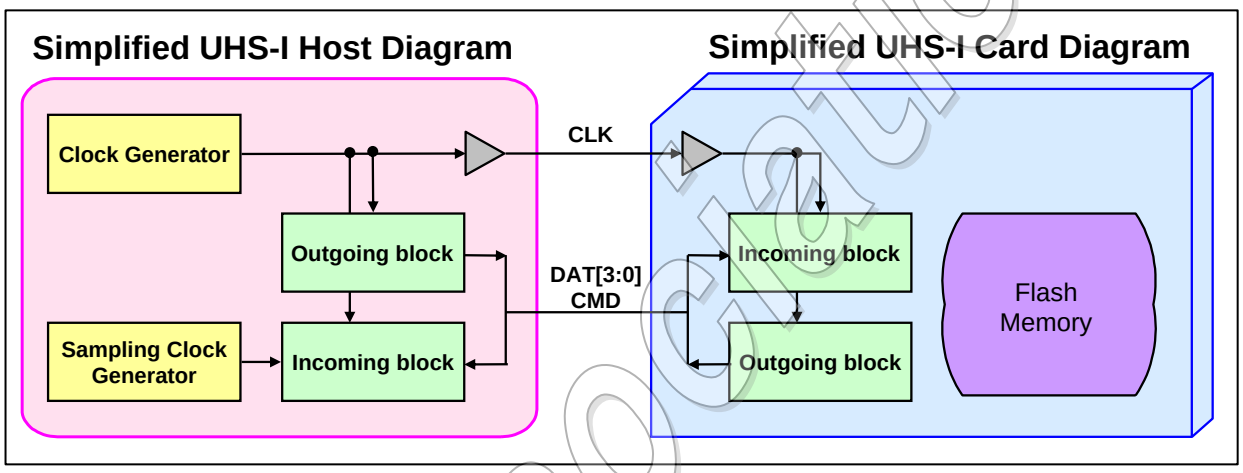


Figure 3-17 : Host and Card Block Diagram

3.9.5.1 Variable Sampling Host

The host use variable sampling clock generator to determine correct sampling point. The host can use predefine tuning block stored in card as an aid for finding sampling operating point. The host can use CMD19 tuning command to read tuning block.
This method is applied to the whole frequency range. In lower frequency less than 25MHz, host needs to access the card without tuning.

3.9.5.2 Fixed Sampling Host

The host uses pre-determined sampling point. This method is available in up to 100MHz. HOST-SDR-FD can make sampling clock by using clock loopback method (Implementation examples of HOST-SDR-FD are shown in 0). UHS50 and UHS104 card shall be compliant to t_{ODLY} (max.) output delay constraint for less than 100MHz frequency range.

3.9.6 Summary of Bus Speed Mode for UHS-I Card

Table 3-6 shows the card requirements regarding Bus Speed modes selected by CMD6 function group 1. The maximum frequency and the maximum power are determined by CMD6.

Bus Speed Mode*1	Max. Bus Speed [MB/s]	Max. Clock Frequency [MHz]	Signal Voltage [V]	Max. Power*2 [W]		
				SDSC*3	SDHC*4	SDXC*5 / SDUC*8
SDR104	104	208	1.8	-	2.88*6	2.88*6
SDR50	50	100	1.8	-	1.44	1.44
DDR50	50	50	1.8	-	1.44	1.44
SDR25	25	50	1.8	-	0.72	0.72
SDR12	12.5	25	1.8	-	0.36	0.36/0.54*7
High Speed	25	50	3.3	0.72	0.72	0.72
Default Speed	12.5	25	3.3	0.36	0.36	0.36/0.54*7

*1: The card supports a UHS-I mode shall support all lower UHS-I modes.
*2: Host may control power by the Power Limit function in CMD6 (Refer to Section 4.3.10.3).
*3: SDSC stands for SD Standard Capacity Memory Card and
*4: SDHC stands for SD High Capacity Memory Card.
*5: SDXC stands for SD Extended Capacity Memory Card.
*6: The actual maximum current may vary from the limit described in this table. It is limited by the Mechanical Addenda in the sections that define the thermal profile of the device and the connector profile.
*7: Host may select either maximum power by XPC in ACMD41 (Refer to Section 4.2.3.1). In SPI mode, XPC is not supported and the power shall be up to 0.36W (100mA at 3.6V on VDD1).
*8: SDUC stands for SD Ultra Capacity Memory Card.

Table 3-6 : Bus Speed Modes of UHS-I Card

DS - Default Speed up to 25MHz 3.3V signaling
HS - High Speed up to 50MHz 3.3V signaling
SDR12 - SDR up to 25MHz 1.8V signaling
SDR25 - SDR up to 50MHz 1.8V signaling
SDR50 - SDR up to 100MHz 1.8V signaling
SDR104 - SDR up to 208MHz 1.8V signaling
DDR50 - DDR up to 50MHz 1.8V signaling

Table 3-7 clarifies option / mandatory of bus speed mode for each card capacity type.

Card Classification		DS	HS	SDR50	SDR104	DDR50
SDSC		M	O	N/A	N/A	N/A
SDHC	Non UHS-I	M	O	N/A	N/A	N/A
	UHS50	M	M	M	N/A	O (Standard SD) M (microSD)
	UHS104	M	M	M	M	O (Standard SD) M (microSD)
SDXC/SDUC	Non UHS-I	M	O	N/A	N/A	N/A
	UHS50	M	M	M	N/A	O (Standard SD) M (microSD)
	UHS104	M	M	M	M	O (Standard SD) M (microSD)

M: Mandatory, O: Optional, N/A: Not Available

Table 3-7 : Bus Speed Mode Option / Mandatory

3.10 Ultra High Speed Phase II (UHS-II) Card

3.10.1 UHS-II Card Operation Modes

SD Bus Interface Modes

- DS - Default Speed up to 25MHz 3.3V signaling
- HS - High Speed up to 50MHz 3.3V signaling
- SDR12 - SDR up to 25MHz 1.8V signaling
- SDR25 - SDR up to 50MHz 1.8V signaling
- SDR50 - SDR up to 100MHz 1.8V signaling
- SDR104 - SDR up to 208MHz 1.8V signaling (Optional)
- DDR50 - DDR up to 50MHz 1.8V signaling (Optional for Standard Size Card)

UHS-II Interface Modes

- FD156 - Full Duplex mode up to 156MB/s at 52MHz in Range B
- HD312 - Half Duplex with 2 Lanes mode up to 312MB/s at 52MHz in Range B (Optional)
- FD312 - Full Duplex mode up to 312MB/s at 52MHz in Range C
- FD624 - Full Duplex mode up to 624MB/s at 52MHz in Range D

3.10.2 UHS-II Card Type

UHS-II supports two card Type

The performance of UHS-II card is indicated based on in Full Duplex mode because HD312 is optional.

- UHS156: UHS-II Card with data rate up to 1.56Gbps in FD156 mode and up to 312Gbps in HD312 mode (Optional).
- UHS624: UHS-II Card with data rate up to 6.24Gbps in FD624 mode

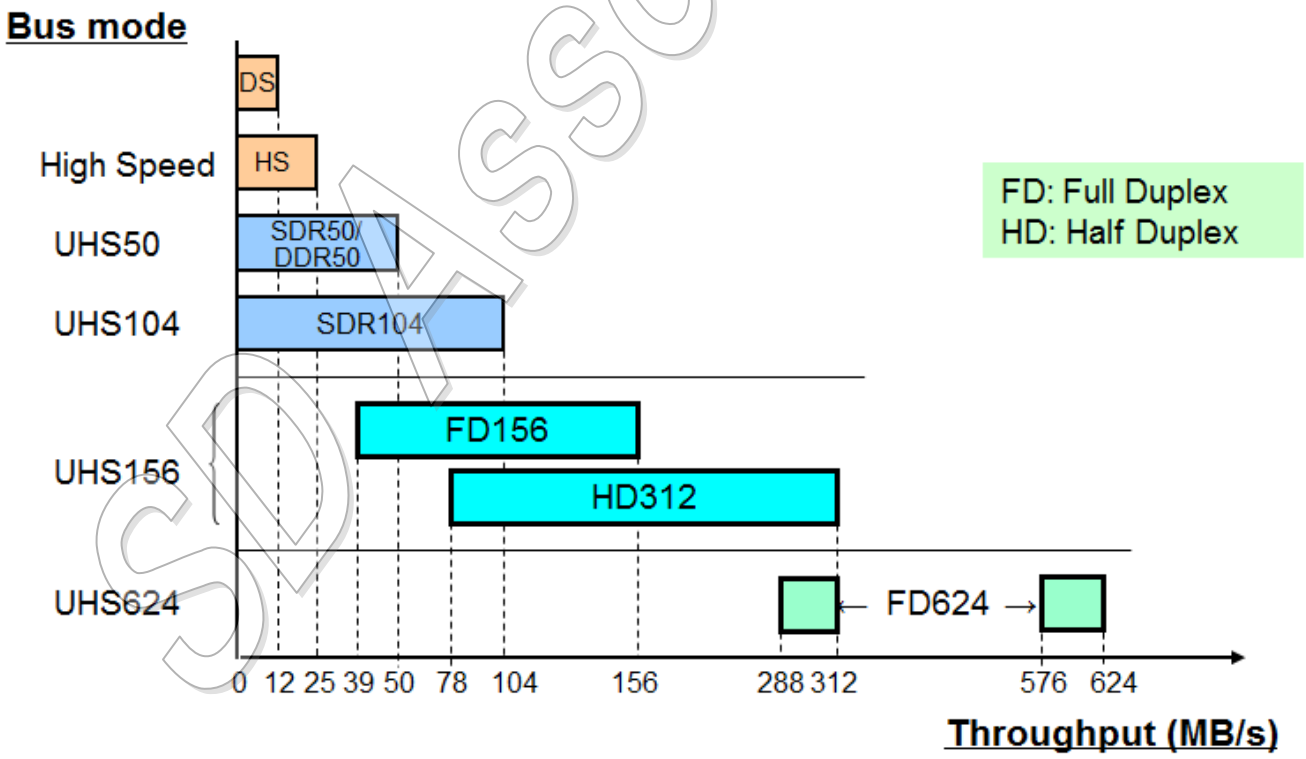


Figure 3-18 : Interface Speed of UHS-II Card (UHS156 and UHS624)

3.10.3 UHS-II Host and Card Combination

Host type / Card type	UHS-II FD156 only	UHS-II HD312 supported	UHS-II FD624 only	UHS-II HD312 and FD624 supported
UHS156 (HD312 not supported)	up to 1.56Gbps	up to 1.56Gbps	up to 1.56Gbps	up to 1.56Gbps
UHS156 (HD312 supported)	up to 1.56Gbps	up to 3.12Gbps	up to 1.56Gbps	up to 3.12Gbps
UHS624 (HD312 not supported)	up to 1.56Gbps	up to 1.56Gbps	up to 6.24Gbps	up to 6.24Gbps
UHS624 (HD312 supported)	up to 1.56Gbps	up to 3.12Gbps	up to 6.24Gbps	up to 6.24Gbps

Note: HD312 cannot be used in Ring Topology

Table 3-8 : UHS-II Host and Card Combinations

3.10.4 UHS-II Interface Selection Sequence

UHS-II supported host shall support Legacy SD Bus Interface (I/F) and UHS-II I/F. Removable UHS-II card slot shall be connected to both I/Fs. Then UHS-II card may be initialized not only in UHS-II mode but also in SD Bus I/F mode.

Figure 3-19 shows how to select UHS-II mode. After power up, SD bus I/F and UHS-II I/F of UHS-II card are enabled. UHS-II supported host provides RCLK and STB.L to D0 lane. Host waits D1 lane to change EIDL to STB.L. If STB.L is detected on D1 lane, host starts UHS-II initialization. If D1 lane is not changed to STB.L by 200us timeout, host should initialize the card in SD Bus I/F mode. As 200us for STB.L detection is defined as per card, host needs to determine total timeout value that depends on UHS-II bus topology of a system.

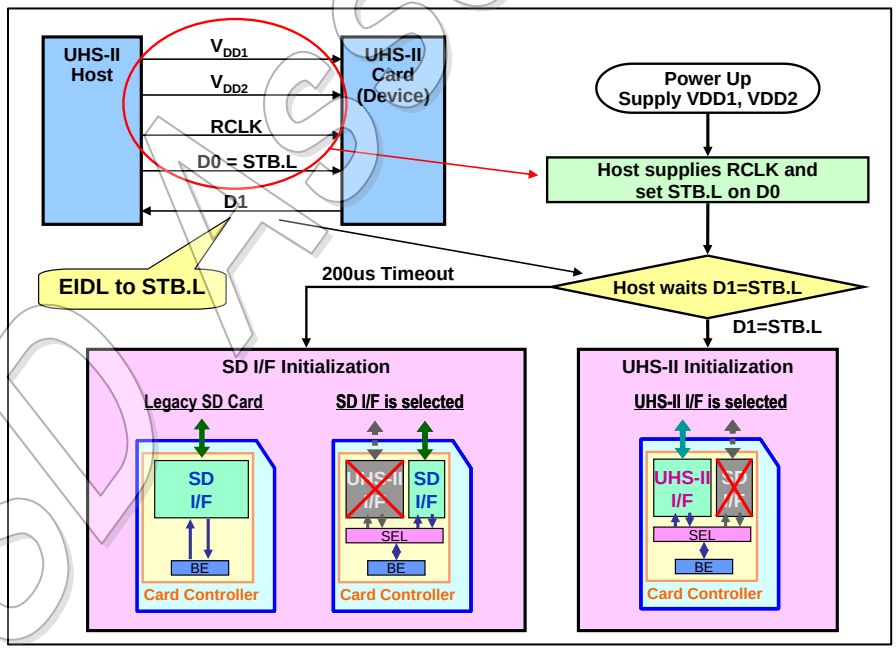


Figure 3-19 : UHS-II Interface Detection

Figure 3-20 shows abstract of UHS-II Initialization sequence. The first step is PHY initialization. PLL is activated and synchronized. Before completing PHY Initialization period, SD Bus I/F is disabled. The second

step is Device Initialization. Backend functions of devices are initialized. The third step is Enumeration. 4-bit unique Device ID is assigned to each device to be able to select one of devices by a Device ID. The fourth step is Configuration. UHS-II register is set to be able to use UHS-II devices in optimized bus sequence. The fifth step is SD-TRAN Initialization. UHS-II emulates SD Commands by the SD-TRAN. SD-TRAN Initialization is equivalent to SD Bus Initialization but host issues SD Commands in UHS-II packets. UHS-II card accepts most SD commands except some specific commands. Refer to Section 4.7.5 about the difference of SD command definition in UHS-II. If CMD0 is received, the UHS-II card re-starts from SD-TRAN initialization.

Figure 3-21 shows SD Bus I/F Initialization sequence for UHS-II Card. UHS-II card shall disable UHS-II Interface before execution of ACMD41 is completed. If VDD2 is provided during SD Interface initialization of UHS-II Card, host shall continue to provide VDD2. VDD2 may be off by power cycle with VDD1 before starting SD Interface initialization. On detecting non UHS-II card, host may turn off VDD2 anytime.

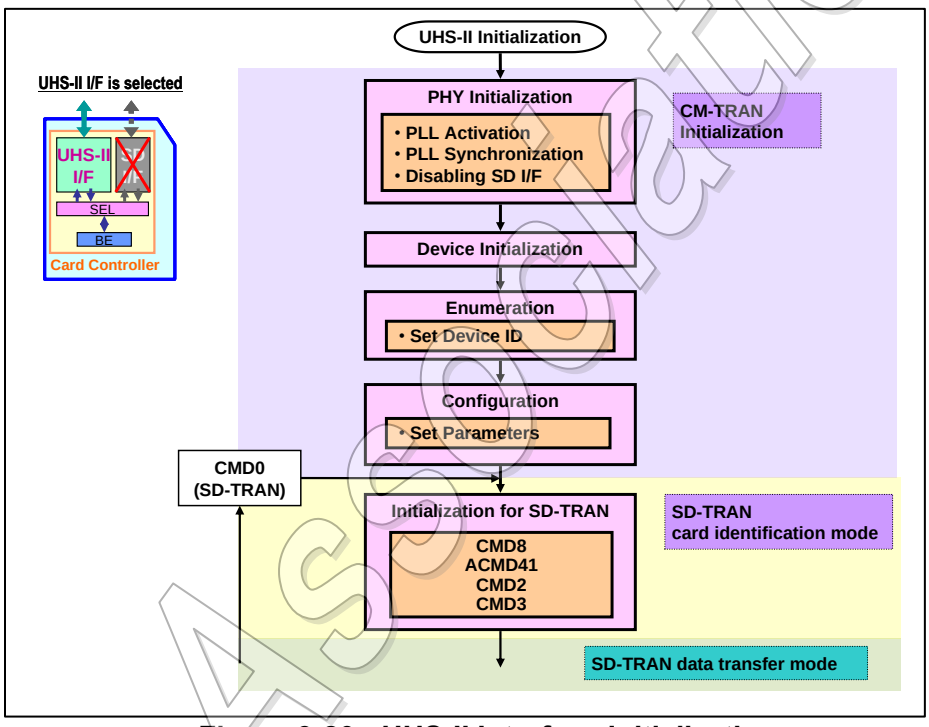


Figure 3-20 : UHS-II Interface Initialization

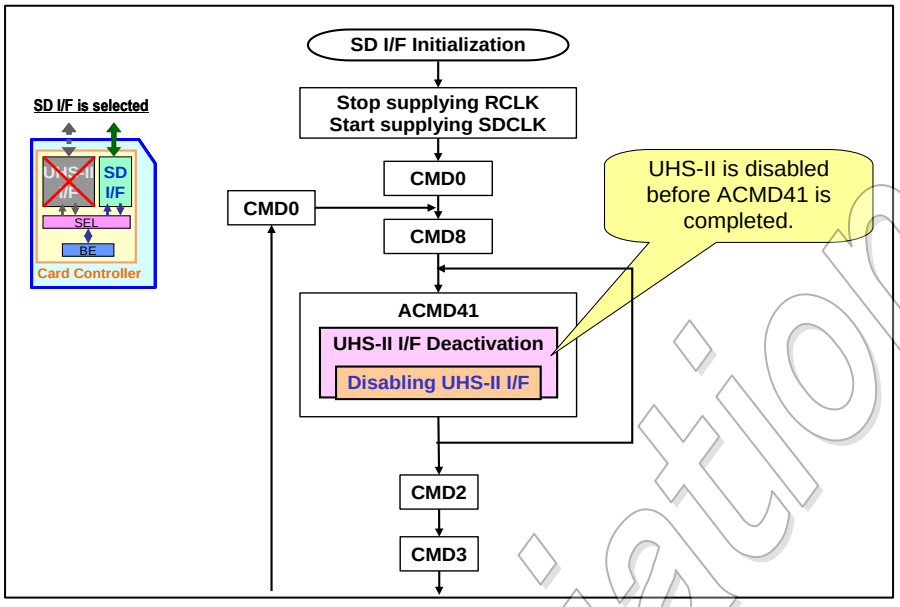


Figure 3-21 : UHS-II Interface Deactivation

3.10.5 Summary of Bus Speed Mode for UHS-II Card

Bus Speed Mode[*1]	Max. Bus Speed [MB/s]	Max. Clock Frequency [MHz]	Signal Voltage [V]	Max. Power[*2] [W]			
				SDSC[*3]	SDHC[*4]	SDXC[*5] / SDUC[*9]	
FD156[*8]	156	52	0.4		1.80[*6]	1.80[*6]	UHS-II I/F
HD312[*8]	312	52	0.4		1.80[*6]	1.80[*6]	
FD624[*8]	624	52	0.4		1.80[*6]	1.80[*6]	
SDR104	104	208	1.8	-	1.80[*6]	1.80[*6]	SD I/F
SDR50	50	100	1.8	-	1.44	1.44	
DDR50	50	50	1.8	-	1.44	1.44	
SDR25	25	50	1.8	-	0.72	0.72	
SDR12	12.5	25	1.8	-	0.36	0.36/0.54[*7]	
High Speed	25	50	3.3	0.72	0.72	0.72	
Default Speed	12.5	25	3.3	0.36	0.36	0.36/0.54[*7]	

*1: The card supports a UHS-I mode shall support all lower UHS-I modes.
*2: Host may control card power by the Power Limit Function of CMD6
*3: SDSC stands for SD Standard Capacity Memory Card.
*4: SDHC stands for SD High Capacity Memory Card.
*5: SDXC stands for SD Extended Capacity Memory Card.
*6: The max. power of removable card is 1.80W and that of embedded device is 2.88W
*7: Host may select either card power by XPC in ACMD41.
In SPI mode, XPC is not supported and the power shall be up to 0.36W (3.6V 100mA).
*8: Minimum power requirement of host in UHS-II mode is 0.72W and it is applied from PHY

Initialization.
*9: SDUC stands for SD Ultra Capacity Memory Card.

Table 3-9 : Bus Speed Modes of UHS-II Card

Table 3-10 shows Card Types and Supported Bus Speed Modes.
It is recommended that UHS-II hosts implement the UHS-I mode (at least SDR50) to provide backward compatibility with UHS-I cards.

Card Types			Bus Modes								
			Non UHS		UHS-I			UHS-II			
								Range B		Range C	Range D
			DS	HS	SDR50	SDR104	DDR50	FD156	HD312	FD312	FD624
SDSC			M	O	N/A	N/A	N/A	N/A	N/A	N/A	N/A
SDHC	Non UHS		M	O	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	UHS-I	UHS50	M	M	M	N/A	(1)	N/A	N/A	N/A	N/A
		UHS104	M	M	M	M	(1)	N/A	N/A	N/A	N/A
	UHS-II	UHS156	M	M	M	O	(1)	M	O	N/A	N/A
	UHS-III	UHS624	M	M	M	O	(1)	M	O	M	M
SDXC/ SDUC	Non UHS		M	O	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	UHS-I	UHS50	M	M	M	N/A	(1)	N/A	N/A	N/A	N/A
		UHS104	M	M	M	M	(1)	N/A	N/A	N/A	N/A
	UHS-II	UHS156	M	M	M	O	(1)	M	O	N/A	N/A
	UHS-III	UHS624	M	M	M	O	(1)	M	O	M	M

Note: Abbreviation: M: Mandatory, Optional, N/A: Not available, (1): O (Standard SD) and M (microSD)

Table 3-10 : Bus Speed Mode Option / Mandatory

3.11 Application Performance Class

The specification defines Application Performance Class that indicates the minimum random and sequential performance of the card under specified pre-conditions (refer to Section 4.16).

- Application Performance Class 1 means:
  - Minimum Card Random performance is Write : 500 IOPS and Read : 1500 IOPS
  - Minimum sustained sequential write performance is 10MB/s
- Application Performance Class 2 means:
  - Minimum Card Random performance is Write : 2000 IOPS and Read : 4000 IOPS using Command Queue (if supported) and Cache Feature
  - Minimum sustained sequential write performance is 10MB/s using Command Queue (if supported) and Cache Feature

Note that unit of random performance [IOPS] indicate number of 4KB sized I/O Operations completed in one second, wherein each transaction was targeted to a random address generated for 256MB address range.

3.12 Cache

Cache feature enables card to improve performance by directing host data into faster memory that may be volatile, which is up to card implementation. Hosts that enable Cache shall be aware that data loss may occur, if the contents in the Cache are not flushed before power off. Refer to Section 4.17 for details of Cache.

3.13 Self Maintenance

Self Maintenance scheme allows the card to performance internal background data management, while the host is aware those operations may be performed by the card and supports them accordingly. Self Maintenance may improve card performance, if enabled. Refer to Section 4.18 for details of Self Maintenance.

3.14 Command Queue

Command Queue (CQ) scheme allows the host to submit commands [tasks] even during data transfer. Voluntary CQ Mode allows card to re-order the tasks and execution of ready task is triggered by host based on the task ready status provided by the card. In Sequential CQ Mode, tasks are executed in order. Refer to Section 4.19 for details of Command Queue scheme.

3.15 LV Interface

Low voltage Interface is to enable card initialization in 1.8V IO mode rather than legacy 3.3V IO mode. Refer to Low Voltage Interface Addendum for additional details. Table 3-11 shows the Card Types and Supported Bus Speed Modes for LV interface.

Card Classification		SPI	DS	HS	SDR12	SDR25	SDR50	SDR104	DDR50	FD156	HD312	FD624
SDHC	LV50	M	M	M	M	M	M	N/A	(1)	N/A	N/A	N/A
	LV104	M	M	M	M	M	M	M	(1)	N/A	N/A	N/A
	LV156	M	M	M	M	M	M	O	(1)	M	O	N/A
	LV624	M	M	M	M	M	M	O	(1)	M	O	M
SDXC/SDUC	LV50	M	M	M	M	M	M	N/A	(1)	N/A	N/A	N/A
	LV104	M	M	M	M	M	M	M	(1)	N/A	N/A	N/A
	LV156	M	M	M	M	M	M	O	(1)	M	O	N/A
	LV624	M	M	M	M	M	M	O	(1)	M	O	M

(1) Optional for Standard SD, Mandatory for microSD cards

Table 3-11 : LV Interface Bus Speed Mode Option / Mandatory

3.16 Higher Bus Speed of UHS-II (UHS-III)

From the Physical Layer Specification Version 6.00, enhanced bus speed mode of UHS-II, i.e. FD624 is newly introduced. Maximum bus speed of FD624 is 624MB/s. Refer to UHS-II Addendum Version 2.00 for more details. "UHS-II" and "UHS-III" denotes card types which are specified by required functions in the UHS-II Addendum. UHS-II Card supports Range A and Range B. UHS-III Card supports Range A to Range D.

For simplicity, use of the term "UHS-II" in common to define extend functions of all card types by the UHS-II Addendum and Versions.

3.17 SD Express Card Type

SD Express card supports the basic SD interface in UHS-I mode as well as PCIe/NVMe interface.

SD Express Card supports either PCIe Gen 3 or Gen 4 interface, with either 1 lane (1 TX, 1 RX) or 2 lanes (2 TX, 2 RX) as defined by PCI-SIG including hot plug-in/removal support. Due to limited space, a limited necessary number of out-of-band signals was adopted in SD Express from the PCI-Standard as described in Section 3.7.3. PCIe Gen 3 supports bit rate of up to 8Gbps, which allows up to 985MB/s (1 lane) or 1,969MB/s (2 lanes) per each direction with 128/130 coding. Additionally, PCIe Gen 4 supports bit rate of up to 16Gbps, which allows up to 1,969MB/s (1 lane) or 3,938MB/s (2 lanes) per each direction. On top of the PCIe channel, the NVM Express (NVMe) protocol is used, as defined by NVMe revision 1.3 (refer to Appendix A.2.2).

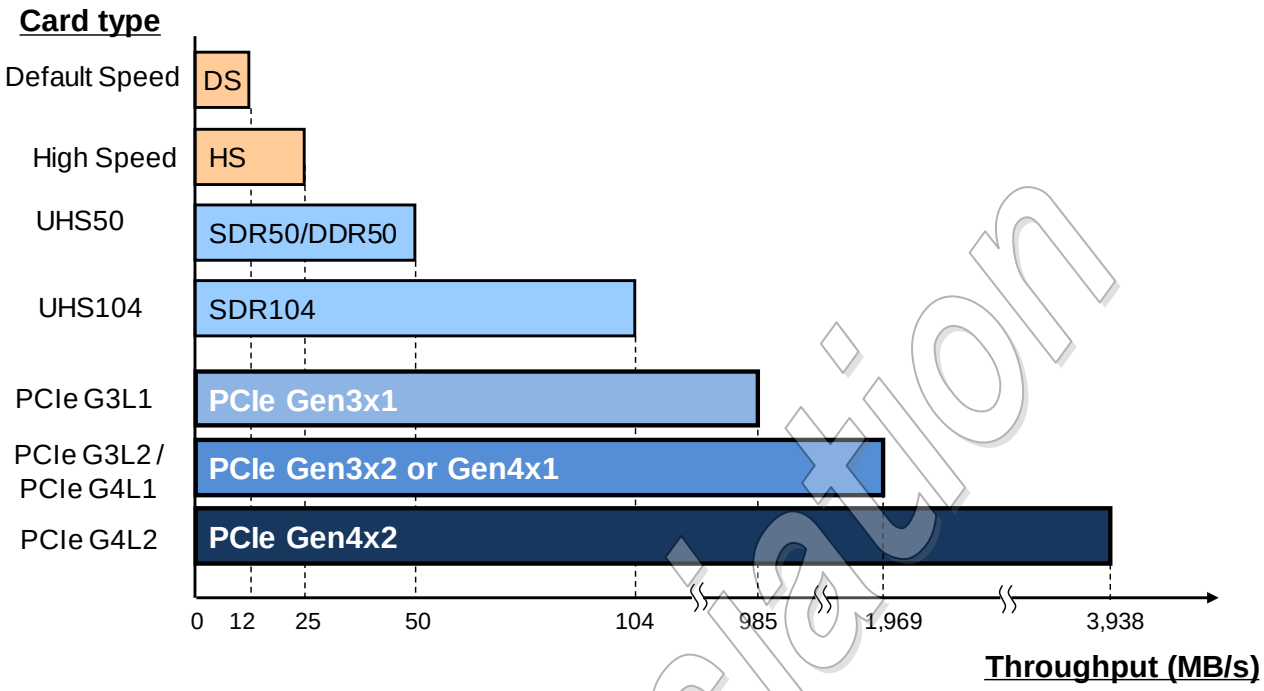


Figure 3-22 : Interface Speed of SD Express Card

SD Express Card may support the following PCIe interface modes.

- Gen3x1 - Using up to PCIe Generation 3 and 1 lane
- Gen3x2 - Using up to PCIe Generation 3 and 2 lanes
- Gen4x1 - Using up to PCIe Generation 4 and 1 lane
- Gen4x2 - Using up to PCIe Generation 4 and 2 lanes

Table 3.17-1 defines SD Express card types and corresponding PCIe interface modes that shall be supported by each card type.

Card Type	PCIe Interface Modes Supported
PCIe G3L1	Gen3x1 (Already introduced at SD Ver7.00)
PCIe G3L2	Gen3x1, Gen3x2
PCIe G4L1	Gen3x1, Gen4x1
PCIe G4L2	Gen3x1, Gen3x2, Gen4x1, Gen4x2

Table 3.17-1 : Card Types of SD Express Cards

3.17.1 SD Express Host and Card Combination

Host type / Card type	SD (any up to UHS50)	SD UHS104	SD-UHS-II	SD Express
SD – up to UHS50	Up to 50MB/s	Up to 50MB/s	Up to 50MB/s (basic SD interface)	Up to 50MB/s (basic SD interface)
SD UHS104	Up to 50MB/s	Up to 104MB/s	Up to 104MB/s (basic SD interface and if host support it)	Up to 104MB/s (basic SD interface and if host support it)
SD-UHS-II	Up to 50MB/s	Up to 104MB/s (if supported by card)	See Table 3-8	Up to 104MB/s (basic SD interface and if host support it)
SD Express	Up to 50MB/s	Up to 104MB/s (if supported by card)	Up to 104MB/s (basic SD interface and if host and card support it)	See Table 3.17.1-1

Table 3-12 : SD Express Host and Card Combinations

Host type / Card type	SD Express Gen3x1	SD Express Gen3x2	SD Express Gen4x1	SD Express Gen4x2
PCIe G3L1	Up to 985MB/s	Up to 985MB/s	Up to 985MB/s	Up to 985MB/s
PCIe G3L2	Up to 985MB/s	Up to 1,969MB/s	Up to 985MB/s	Up to 1,969MB/s
PCIe G4L1	Up to 985MB/s	Up to 985MB/s	Up to 1,969MB/s	Up to 1,969MB/s
PCIe G4L2	Up to 985MB/s	Up to 1,969MB/s	Up to 1,969MB/s	Up to 3,938MB/s

Table 3.17.1-1 : SD Express Host and Card Combinations (Detailed)

3.17.2 SD Express Interface Selection and Initialization Sequence

SD Express card may be initialized in either PCIe mode or SD Bus I/F mode. It is highly recommended to initialize through legacy SD Bus I/F mode allowing the host a simple method to detect card's compatibility and support of PCIe and switch to PCIe mode if supported.

Figure 3-24 shows the process of card detection and initialization from the host side.
Figure 3-23 shows the card's internal states related to SD and PCIe modes of operation.

After VDD1 power up, the card's IOs are in input mode (Pseudo Init State). After VDD2 or VDD3 is powered up after VDD1, the PCIe interface is ready for PCIe link-up operation.

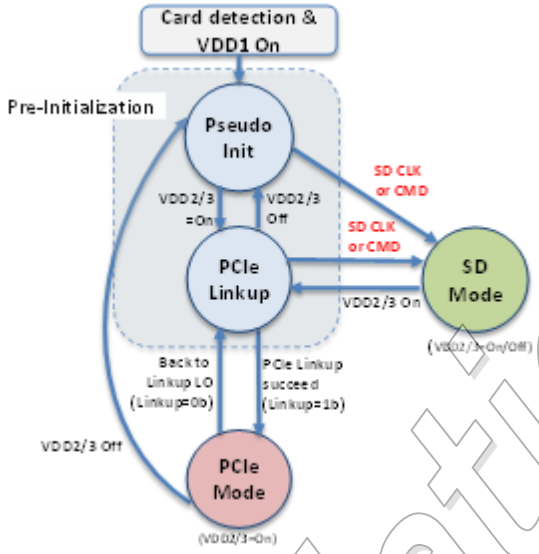


Figure 3-23 : SD Express Interface Detection and Init – Card Internal States

Description of the card's internal states:

- ***Pseudo Init State :***

IOs are in input mode. Moving out from this state happens either if SD CLK detected or CMD (CMD0, CMD8 or ACMD41) is accepted (to SD Mode) or if VDD2 or VDD3 presence detected (to PCIe Linkup state). Usage of SD CLK or CMD (here and after) is up to the card's internal implementation which method to use. In the case of LVS identification, state transition condition shall be based on SD CLK detection only.

- ***PCIe Linkup State :***

IOs are in PCIe IO mode + SD CLK line ready to accept transition or CMD line ready to accept CMD (up to implementation). Moving out from this state happens either if SD CLK detected or CMD (CMD0, CMD8 or ACMD41) is accepted (to SD Mode) or if PCIe Linkup=1b (to PCIe mode) or if transition of VDD2 or VDD3 from On to Off is detected (back to Pseudo Init).

- ***SD Mode State:***

IOs are in SD mode. Moving out from this state is only upon detection of VDD2 or VDD3 transition from Off to On (to PCIe Linkup state). The transition is valid only when SD card is in 'idle' state (as in Figure 4-1). VDD2/3 On is a power on reset to SD Mode.

- ***PCIe Mode State:***

IOs are in PCIe mode (SD CMDs are not supported). Moving out from this state is either if Linkup=0b (back to PCIe Linkup state) or if VDD2 or VDD3 are turned off (back to Pseudo Init State).

- ***Transition notations:***

VDD2/3 on ➔ VDD2 or VDD3 switch from Off to On
VDD2/3 off ➔ VDD2 or VDD3 switch from On to Off
(VDD2 or VDD3 shall remain less than 0.2V for at least 1ms before next step)
VDD2/3=on ➔ VDD2 or VDD3 is in On state
VDD2/3=off ➔ VDD2 or VDD3 is in Off state
SD CLK ➔ Transition from 0 to 1 of SD CLK detected
CMD ➔ CMD0, CMD8, ACMD41 or CMD line driven to Low detection.

(Host shall issue CMD0, CMD8 and ACMD41 command sequence as defined in SD initialization flow (Figure 4-2).
An LVS host shall execute LVS handshake before CMD0.
In addition, Fast Boot takes place if CMD line driven by host to Low for 74 clocks or greater, or if CMD0 with a special argument is issued by host with no other commands issued beforehand (Section 4.21.3)

Following is a description of the interface detection and selection of SD Express card through a typical path. The full description with all the options is described in Figure 3-24.
Upon card insertion detection it is recommended to initialize the SD bus interface and check through CMD8 whether the card supports PCIe mode and whether it supports VDD3. If it does not support PCIe, the host may proceed with the legacy SD Bus I/F mode. If PCIe is supported, the host assures that PERST# is asserted and CLKREQ# is pulled up and then it turns VDD2 on or VDD3 on (if VDD3 is supported by the card and the host). SD CLK shall only be provided when host is entering SD mode.
Host then waits, up to 1ms (T_{PVCRL}) for CLKREQ# line to be asserted (“0”) by the card and, if asserted, host de-asserts PERST# (=“1”) after providing stable REFCLK for at least 100us ($T_{PERST-CLK}$). Host then may proceed with the PCIe/NVMe linkup process and initialization process. In case the PCIe training process fails (CLKREQ# line was not asserted by the card or PCIe linkup failed), the host may access the card through the legacy SD interface.
Host may also initialize the card without going through the legacy SD Bus I/F mode first. However in microSD case it is highly recommended not to turn VDD3 on before SD card is detected to avoid any possible conflict with other similar card types in the market.
Negotiation of Generation and number of lanes between host and card takes place during PCIe Linkup. Default NVMe Power State (PS) value of SD Express Card shall have Maximum Power value of 1.8W. Before completing PCIe Linkup, card shall not consume more than 1.8W even if the card has a capability of higher Power States. If SD Express Card supports power states with Maximum Power (MP) of more than 1.8W, and host wants to utilize a power state above 1.8W, host shall connect VDD1 also to pin#19 allowing the split of the current consumed from VDD1 source between the two pads, pad#19 and pad#4, assuring that the maximum normal current passed through each of the connector’s contacts will not be more than 500mA. After activation of pad#19 host may select any power state above 1.8W.
As described in Section 0, power consumption is limited to 1.8W in PCIe Gen3x1 mode even if the card supports over 1.8W power and host sets MP over 1.8W. However, the host and the card may activate pin#19 and pad#19 respectively in this case.
Host may use either of the following two methods before connecting VDD1 to pad#19:

- The above mentioned method of host reading the cards power state capability and acting upon it.
- Electrical method of detecting pad19 existence as described in Appendix J.

Moreover, host may supply VDD2 and VDD2a at the same time. Refer to Section 6.4.3 for more details.
If the host does not support power states above 1.8W and/or pad#19 does not exist, host shall not activate pin#19.

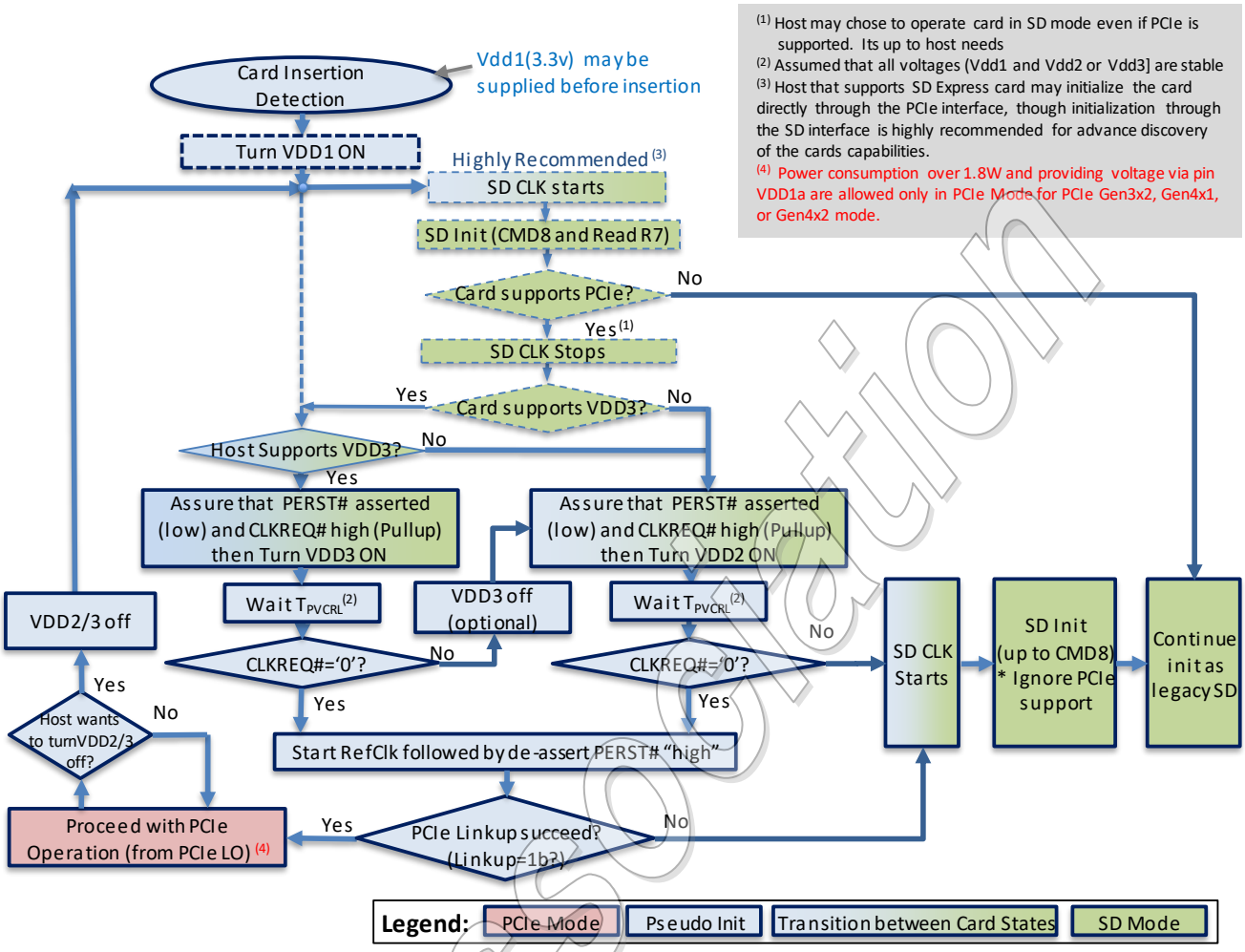


Figure 3-24 : SD Express Interface Detection and Initialization Flow (Host side)

Figure 3-25 provides illustration of the PCIe training process as described in the SD Express Interface Detection flow.
PCIe/NVMe linkup and the rest of the initialization sequence shall be done as defined by PCIe Standard.

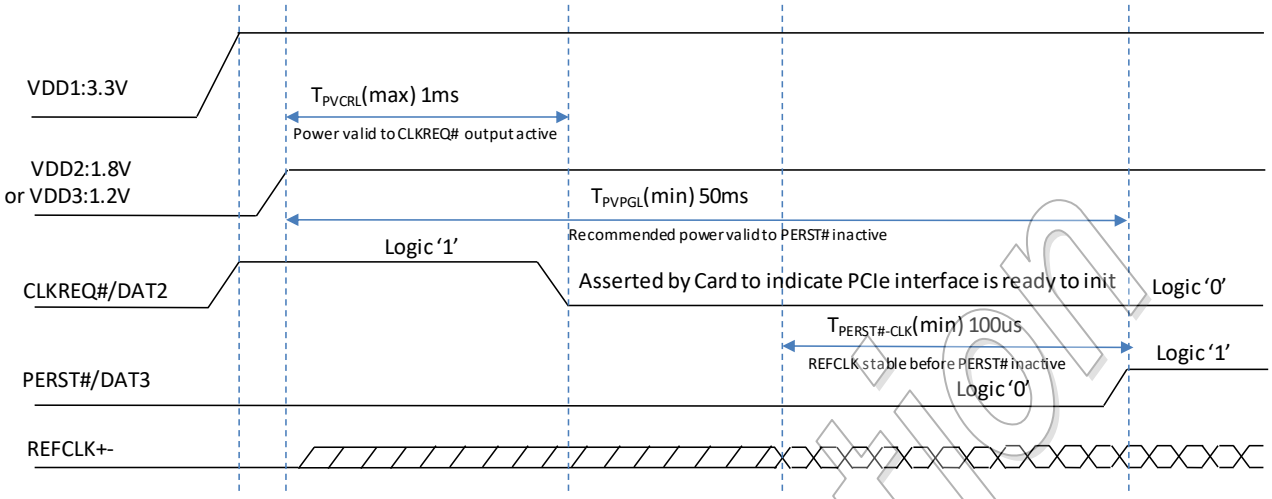


Figure 3-25 : PCIe Training Process

Symbol	Parameter	Min	Max	Units
T_{PVCRL}	Power Valid(1) to CLKREQ# Output active		1	ms
T_{PVPGL}	Power Valid to PERST# Input inactive. This parameter is implementation specific and recommended only.	50		ms
$T_{PERST-CLK}$	REFCLK stable before PERST# inactive	100		us

Note (1): Power Valid is measured at the later rising edge of either VDD1, VDD2 or VDD3

Table 3-13 : Timing of PCIe Interface Training

A detailed description of the SD Express card interface condition is given in Section 8.3.

3.17.3 Summary of Bus Speed Mode for SD Express Card

Bus Speed Mode[*1]	Max. Bus Speed [MB/s]	Max. Clock Frequency [MHz]	Signal Voltage [V]	Max. Power[*2] [W]		
				SDHC[*4]	SDXC[*5] / SDUC[*3]	
PCIe Gen4x2	3,938	100	0.4	4.00	4.00	PCIe I/F
PCIe Gen4x1	1,969	100	0.4	2.80	2.80	
PCIe Gen3x2	1,969	100	0.4	2.80	2.80	
PCIe Gen3x1	985	100	0.4	1.80[*6]	1.80[*6]	
SDR104	104	208	1.8	1.80[*6]	1.80[*6]	SD I/F
SDR50	50	100	1.8	1.44	1.44	
DDR50	50	50	1.8	1.44	1.44	
SDR25	25	50	1.8	0.72	0.72	
SDR12	12.5	25	1.8	0.36	0.36/0.54[*7]	
High Speed	25	50	3.3	0.72	0.72	
Default Speed	12.5	25	3.3	0.36	0.36/0.54[*7]	

*1: The card supporting PCIe mode shall support all lower UHS-I modes except SDR104, which is optional.
*2: Host may control card power by the Power Limit Function of CMD6 when operated in any of non-PCIe modes. In PCIe/NVMe mode the host may control the maximum consumed power through the PCIe power throttling (in static manner during init) or with NVMe power state modes.
*3: SDUC stands for SD Ultra Capacity Memory Card. SDSC is not supported by SD-PCIe Express cards.
*4: SDHC stands for SD High Capacity Memory Card.
*5: SDXC stands for SD Extended Capacity Memory Card.
*6: The max. power of removable card is 1.80W and that of embedded device is 2.88W
*7: Host may select either card power by XPC in ACMD41 when operated in any of non-PCIe modes In SPI mode, XPC is not supported and the power shall be up to 0.36W (3.6V 100mA).
*8: Minimum power requirement of SD Express host in PCIe mode is 1.80W and it is applied from PHY Initialization.

Table 3-14 : Bus Speed Modes of SD Express Card

3.18 Features of Non CPRM Card

Security is optional for writeable SDSC, SDHC and SDXC SD Memory cards. SDUC Card does not support security feature. SD_SECURITY shall be set to 0 for Non CPRM Cards. Non CPRM Cards will not support any security commands except for ACMD44. For all security commands mentioned below, the card shall either set ILLEGAL_COMMAND error bit or ERROR bit in the status register. The choice of the response is left to the card vendor's implementation.

ACMD18, ACMD25, ACMD26, ACMD38, ACMD43, ACMD45, ACMD46, ACMD47, ACMD48, ACMD49 (SDSC only)

3.19 Boot Functionalities

Boot partition and Fast Boot functionalities are defined to support IoT and mobile hosts that store their boot code in SD card. Those hosts need to fetch their boot code promptly after power up with minimal delay.
These functions are not defined for UHS-II mode and cannot be implemented to UHS-II cards.

Refer to Section 4.21 for more details.

3.20 TCG Security

TCG (Trusted Computing Group) provides open standards for secure computing, including enterprise storage, mobile devices and so on. Thousands of vendors offer a variety of TCG-based products, including hardware, applications, and services.
To introduce TCG security while keeping backward compatibility to the existing SD Specifications, following items are defined:

(1) Protocols over SD interface for TCG security
(2) Specifications for realizing MBR Shadowing in SD card

For TCG supported card (Bit45 of SCR is set to 1), TCG is enabled when Enabled column of the SPInfo Table is 1 (True), and disabled otherwise. A card where TCG is enabled is called “TCG enabled card.”

This function is not defined for UHS-II mode and cannot be implemented to UHS-II cards.
Refer to Section 4.22 for more details.

3.21 RPMB

RPMB (Replay Protected Memory Block) is introduced to store data in an authenticated memory area for the purpose of protecting from replay attack, or avoiding unexpected data update regardless of whether it is intentionally or not.

For RPMB supported card (Bit46 of SCR is set to 1), RPMB is enabled when RPMB key is set, and disabled otherwise. A card where RPMB is enabled is called “RPMB enabled card.”

Write Protect Until Power Cycle is introduced in Physical Layer Specification Version 9.00. This feature is controlled by RPMB. When RPMB is enabled, Write Protect Until Power Cycle and Permanent Write Protect are controlled by RPMB.

This function is not defined for UHS-II mode and cannot be implemented to UHS-II cards.
Refer to Section 4.23 for more details.

4. SD Memory Card Functional Description

4.1 General

All communication between host and cards is controlled by the host (master). The host sends commands of two types: broadcast and addressed (point-to-point) commands.

- **Broadcast commands**
  Broadcast commands are intended for all cards. Some of these commands require a response.
- **Addressed (point-to-point) commands**
  The addressed commands are sent to the addressed card and cause a response from this card.
  A general overview of the command flow is shown in Figure 4-1 for card identification mode and in Figure 4-13 for data transfer mode. The commands are listed in the command tables (Table 4-23 to Table 4-33). The dependencies between current state, received command, and following state are listed in Table 4-35. In the following sections, the various card operation modes will be described first. Afterwards, the restrictions for controlling the clock signal are defined. All SD Memory Card commands, along with the corresponding responses, state transitions, error conditions and timings are presented in the succeeding sections.
  Two operation modes are defined for the SD Memory Card system (host and cards):
- **Card identification mode**
  The host will be in card identification mode after reset and while it is looking for new cards on the bus. Cards will be in this mode after reset until the SEND_RCA command (CMD3) is received.
- **Data transfer mode**
  Cards will enter data transfer mode after their RCA is first published. The host will enter data transfer mode after identifying all the cards on the bus.
- **Pre-init mode**
  This mode is introduced to define state transition related to LVS Identification and Fast Boot. When host does not support 3.3V signaling, LVS Identification is necessary before transit to idle state. In addition, to execute Fast Boot, host drives CMD line Low for greater than or equal to 74 clocks or issues CMD0 with a special argument.

Table 4-1 shows the dependencies between operation modes and card states. Each state in the SD Memory Card state diagram (see Figure 4-1) is associated with one operation mode:

Card state	Operation mode
Inactive State	inactive
Idle State	card identification mode
Ready State	
Identification State	
Stand-by State	data transfer mode
Transfer State	
Sending-data State	
Receive-data State	
Programming State	
Disconnect State	

Table 4-1 : Overview of Card States vs. Operation Modes

Refer to Section 4.21.5 for pre-init mode.

4.2 Card Identification Mode

While in card identification mode the host resets all the cards that are in card identification mode, validates operation voltage range, identifies cards and asks them to publish Relative Card Address (RCA). This operation is done to each card separately on its own CMD line. All data communication in the Card Identification Mode uses the command line (CMD) only.

During the card identification process, the card shall operate in the SD clock frequency of the identification clock rate f_{OD} (see Section 6.6.6).

4.2.1 Card Reset

In SD Mode, the command GO_IDLE_STATE (CMD0) is the software reset command and sets each card into *Idle State* regardless of the current card state. Cards in *Inactive State* are not affected by this command.

After power-on by the host, all cards are in *Idle State*, including the cards that have been in *Inactive State* before.

After power-on or CMD0, all cards' CMD lines are in input mode, waiting for start bit of the next command. The cards are initialized with a default relative card address (RCA=0x0000) and with a default driver strength with 400KHz clock frequency. In case of 3.3V signaling, default driver strength is specified by the Driver Stage Register (DSR) if supported and selected highest driving current capability. In case of 1.8V signaling, default driver strength is specified by type B driver.

In UHS-II Mode, RCA is not cleared to 0x0000 by CMD0 and kept its Device ID, which is determined by Enumeration.

4.2.2 Operating Condition Validation

At the start of communication between the host and the card, the host may not know the card supported voltage and the card may not know whether it supports the current supplied voltage. The host issues a reset command (CMD0) with a specified voltage while assuming it may be supported by the card. To verify the voltage, a following new command (CMD8) is defined in the Physical Layer Specification Version 2.00.

SEND_IF_COND (CMD8) is used to verify SD Memory Card interface operating condition. The card checks the validity of operating condition by analyzing the argument of CMD8 and the host checks the validity by analyzing the response of CMD8 (See Section 4.3.13). The supplied voltage is indicated by VHS field in the argument. The card assumes the voltage specified in VHS as the current supplied voltage. Only 1-bit of VHS shall be set to 1 at any given time. Both CRC and check pattern are used for the host to check validity of communication between the host and the card.

If the card can operate on the supplied voltage, the response echoes back the supply voltage and the check pattern that were set in the command argument.

If the card cannot operate on the supplied voltage, it returns no response and stays in idle state. It is mandatory to issue CMD8 prior to first ACMD41 to initialize SDHC, SDXC or SDUC Card (See Figure 4-1). Receipt of CMD8 makes the cards realize that the host supports the Physical Layer Specification Version 2.00 or later and the card can enable new functions.

SD_SEND_OP_COND (ACMD41) is designed to provide SD Memory Card hosts with a mechanism to identify and reject cards which do not match the V_{DD} range desired by the host. This is accomplished by the host sending the required V_{DD} voltage window as the operand of this command (See Section 5.1). Cards which cannot perform data transfer in the specified range shall discard themselves from further bus operations and go into *Inactive State*. The levels in the OCR register shall be defined accordingly (See Section 5.1). Note that ACMD41 is application specific command; therefore APP_CMD (CMD55) shall always precede ACMD41. The RCA to be used for CMD55 in *idle_state* shall be the card's default RCA = 0x0000.

After the host issues a reset command (CMD0) to reset the card, the host shall issue CMD8 prior to ACMD41 to re-initialize the SD Memory card.

Figure 4-1 shows State Diagram of card Identification mode for SD I/F. In case of UHS-II mode, refer to SD-TRAN Section of the UHS-II Addendum. Moreover, in case of executing Fast Boot, also refer to Section 4.21.5.

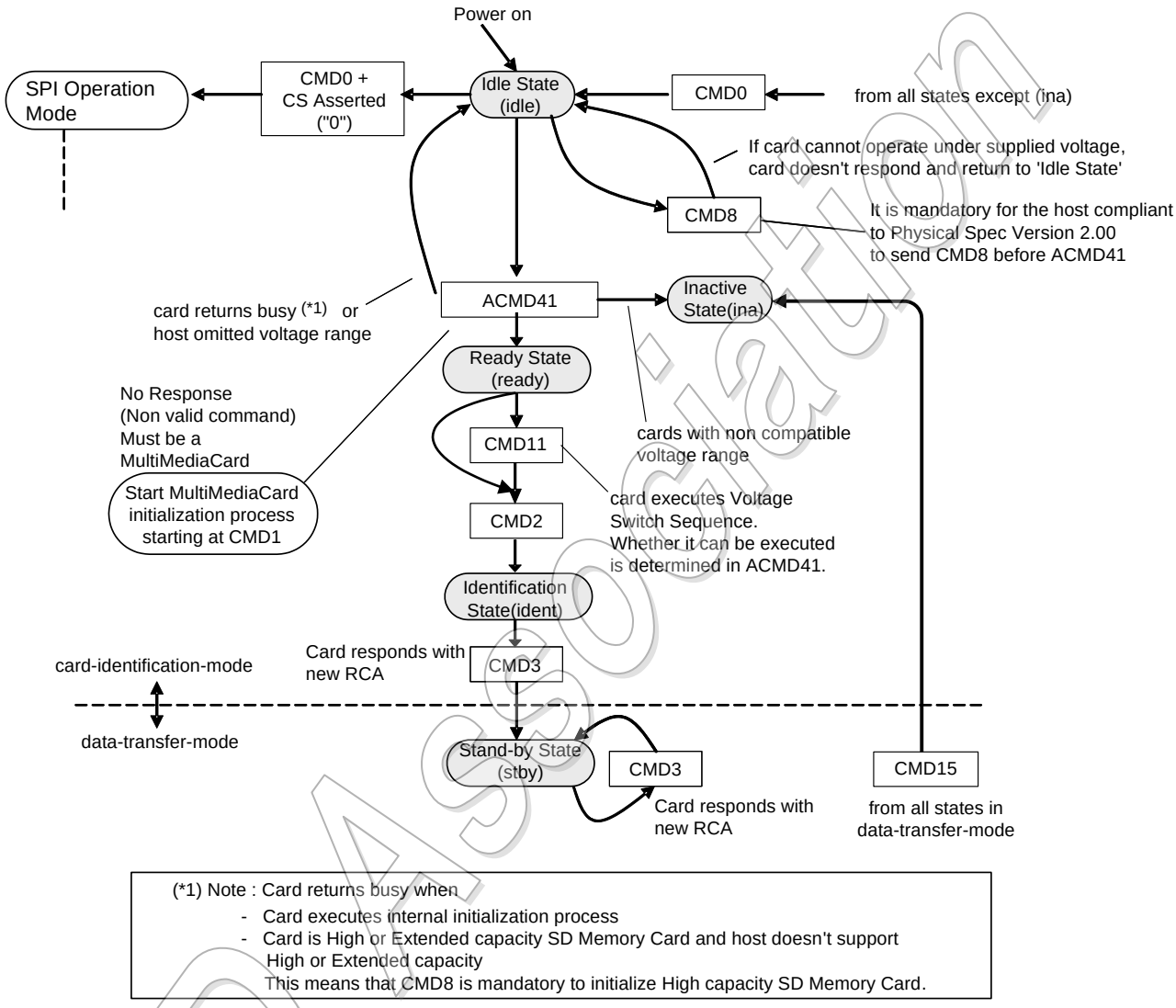


Figure 4-1 : SD Memory Card State Diagram (card identification mode)

By setting the OCR to zero in the argument of ACMD41, the host can query each card and determine the common voltage range before sending out-of-range cards into the *Inactive State* (query mode). This query should be used if the host is able to select a common voltage range or if a notification to the application of non-usable cards in the stack is desired. The card does not start initialization and ignores HCS in the argument (refer to Section 4.2.3) if ACMD41 is issued as a query. Afterwards, the host may choose a voltage for operation and reissue ACMD41 with this condition, sending incompatible cards into the *Inactive State*.

During the initialization procedure, the host is not allowed to change the operating voltage range. Refer to power up sequence as described in Section 6.4.

4.2.3 Card Initialization and Identification Process

After the bus is activated the host starts card initialization and identification process (See Figure 4-2). The initialization process starts with SD_SEND_OP_COND (ACMD41) by setting its operational conditions and the HCS bit in the OCR. The HCS (Host Capacity Support) bit set to 1 indicates that the host supports SDHC or SDXC Card. The HCS (Host Capacity Support) bit set to 0 indicates that the host supports neither SDHC nor SDXC Card.

Receiving of CMD8 expands the ACMD41 function; HCS in the argument and CCS (Card Capacity Status) in the response. HCS is ignored by cards, which didn't respond to CMD8. However the host should set HCS to 0 if the card returns no response to CMD8. Standard Capacity SD Memory Card ignores HCS. If HCS is set to 0, SDHC and SDXC Cards never return ready status (keep busy bit to 0). The busy bit in the OCR is used by the card to inform the host whether initialization of ACMD41 is completed. Setting the busy bit to 0 indicates that the card is still initializing. Setting the busy bit to 1 indicates completion of initialization. Card initialization shall be completed within 1 second from the first ACMD41. The host repeatedly issues ACMD41 for at least 1 second or until the busy bit are set to 1.

The card checks the operational conditions and the HCS bit in the OCR only at the first ACMD41 with setting voltage window in the argument. While repeating ACMD41, the host shall not issue another command except CMD0.

If the card responds to CMD8, the response of ACMD41 includes the CCS field information. CCS is valid when the card returns ready (the busy bit is set to 1). CCS=0 means that the card is SDSC. CCS=1 means that the card is SDHC or SDXC.

The host performs the same initialization sequence to all of the new cards in the system. Incompatible cards are sent into *Inactive State*. The host then issues the command ALL_SEND_CID (CMD2), to each card to get its unique card identification (CID) number. Card that is unidentified (i.e. which is in *Ready State*) sends its CID number as the response (on the CMD line). After the CID was sent by the card it goes into *Identification State*. Thereafter, the host issues CMD3 (SEND_RELATIVE_ADDR) asks the card to publish a new relative card address (RCA), which is shorter than CID and which is used to address the card in the future data transfer mode. Once the RCA is received the card state changes to the *Stand-by State.* At this point, if the host wants to assign another RCA number, it can ask the card to publish a new number by sending another CMD3 command to the card. The last published RCA is the actual RCA number of the card.

The host repeats the identification process, i.e. the cycles with CMD2 and CMD3 for each card in the system.

Initialization of SDXC is identical to SDHC. User area capacity of SDXC card is specified by C_SIZE and it shall be more than or equal to 32GB.

Refer to Section 8.3 for SDUC card initialization.

Figure 4-2 shows Card Initialization and Identification for SD I/F. In case of UHS-II mode, refer to SD-TRAN Section of the UHS-II Addendum.

Application Notes: The host shall set ACMD41 timeout more than 1 second to abort repeat of issuing ACMD41 when the card does not indicate ready. The timeout count starts from the first ACMD41 which is set voltage window in the argument.

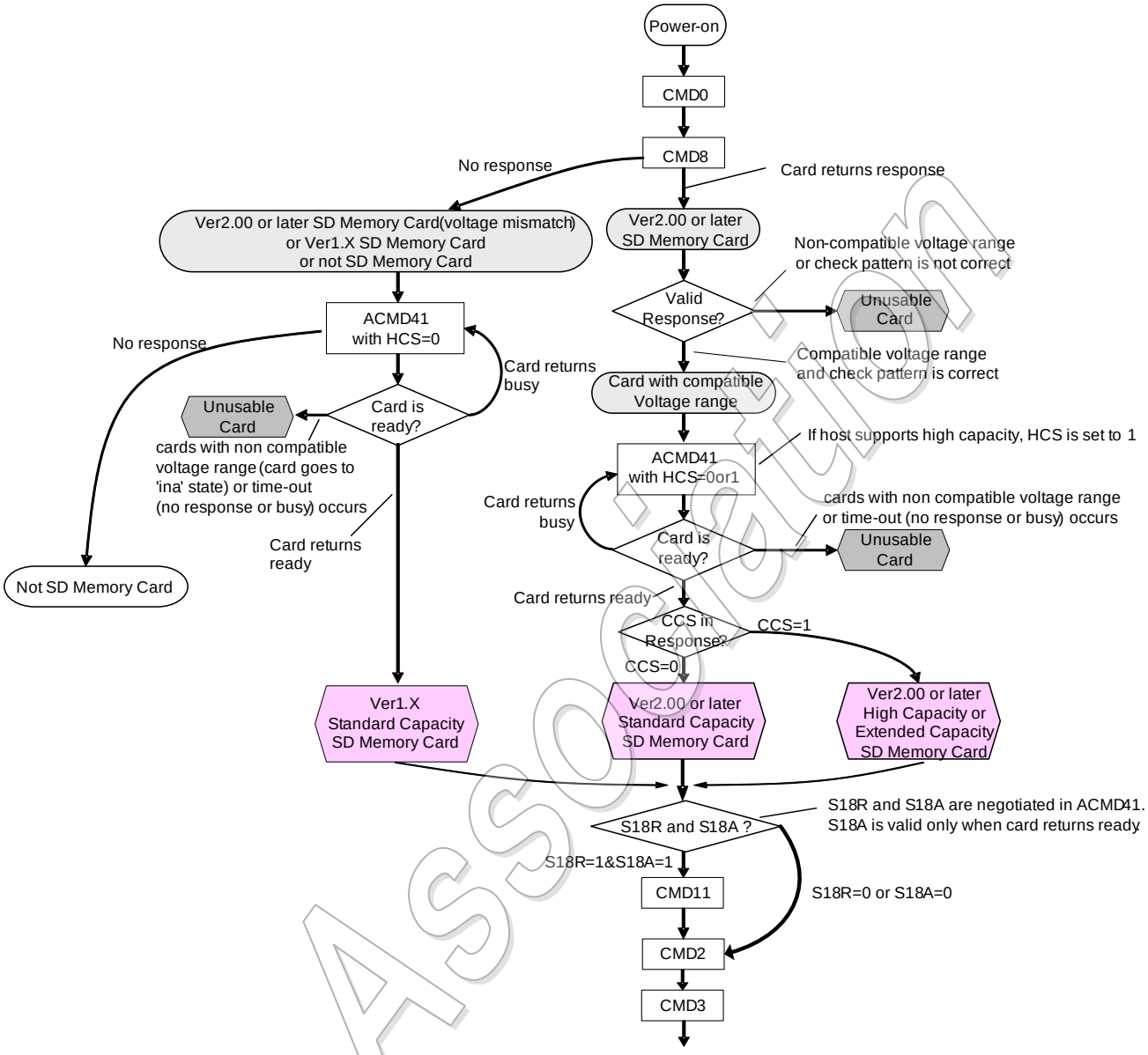


Figure 4-2 : Card Initialization and Identification Flow (SD mode)

4.2.3.1 Initialization Command (ACMD41)

Followings are general rules of the argument of ACMD41:

(1) If the voltage window field (bit 23-0) in the argument is set to zero, it is called "inquiry CMD41" that does not start initialization and is use for getting OCR. The inquiry ACMD41 shall ignore the other field (bit 31-24) in the argument.

(2) If the voltage window field (bit 23-0) in the argument is set to non-zero at the first time, it is called "first ACMD41" that starts initialization. The other field (bit 31-24) in the argument is effective.

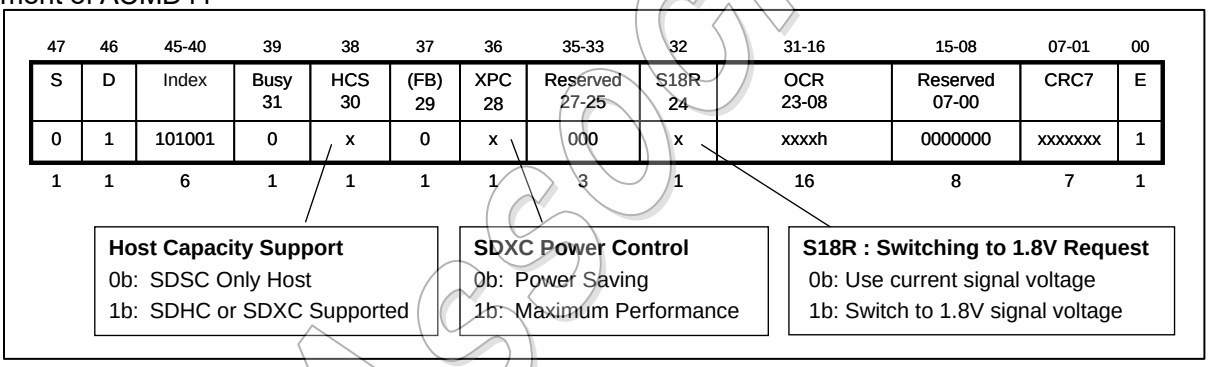
(3) The argument of following ACMD41 shall be the same as that of the first ACMD41.

Figure 4-3 shows argument format and Figure 4-4 shows response format. Two new fields are added to the argument of ACMD41.

If a SDXC Card is initialized with XPC=0, the card is operating at up to 0.36W (100mA at 3.6V on VDD1) in Default Speed or SDR12, and if the card does not support Speed Class, Class 0 is indicated in SD Status. If a SDXC Card is initialized with XPC=1, the card is operating at up to 0.54W (150mA at 3.6V on VDD1) in Default Speed or SDR12, and the card supports Speed Class. Re-initialization is required to change XPC selection.

UHS-I supported host sets S18R=1 in the argument of ACMD41 to request the card to switch 1.8V signaling level. UHS-I card respond with S18A=1 in the response of ACMD41 (if in 3.3V signaling mode) and then host can issue voltage switch command. Once voltage switch is performed, UHS-I card indicates S18A=0 to keep current signal voltage. In UHS-II mode, the card always indicates S18A=0.

(1) Argument of ACMD41



Note: Fast Boot (Bit 29) is reserved for eSD. This bit is not related to the Fast Boot described in Section 4.21.3 of this document.

Figure 4-3 : Argument of ACMD41

(2) Response of ACMD41 (R3)

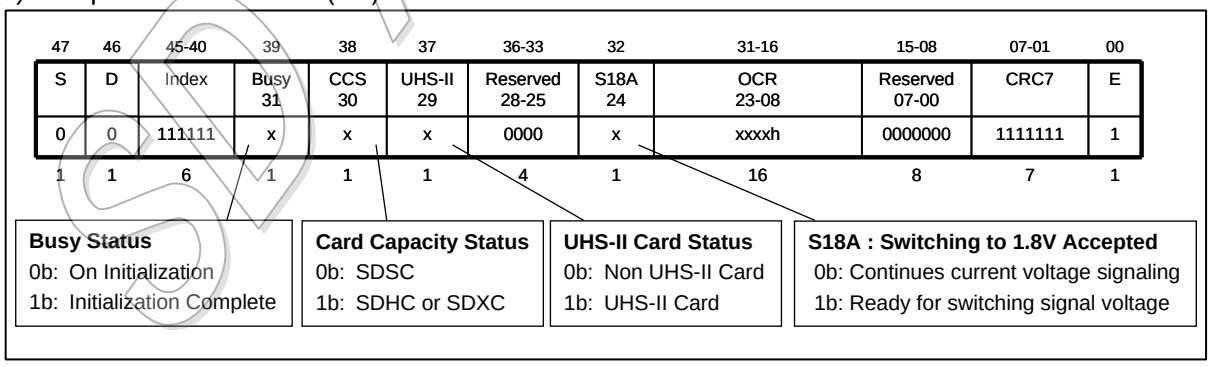


Figure 4-4 : Response of ACMD41

CCS (Bit 30), UHS-II (Bit 29) and S18A (Bit 24) are valid when Busy (Bit 31) is set to 1. For host or card

supporting SDUC, refer to Figure 4-75 in Section 4.20.2.

4.2.4 Bus Signal Voltage Switch Sequence

Refer to Low Voltage Interface Addendum for LV interface Card initialization.
Also refer to Section 4.21.3 for signal voltage switching during Fast Boot.

4.2.4.1 Initialization Sequence for UHS-I

Figure 4-5 shows sequence of commands to perform voltage switch and Figure 4-6 shows initialization flow chart for UHS-I hosts. Red and yellow boxes are new procedure to initialize UHS-I card.

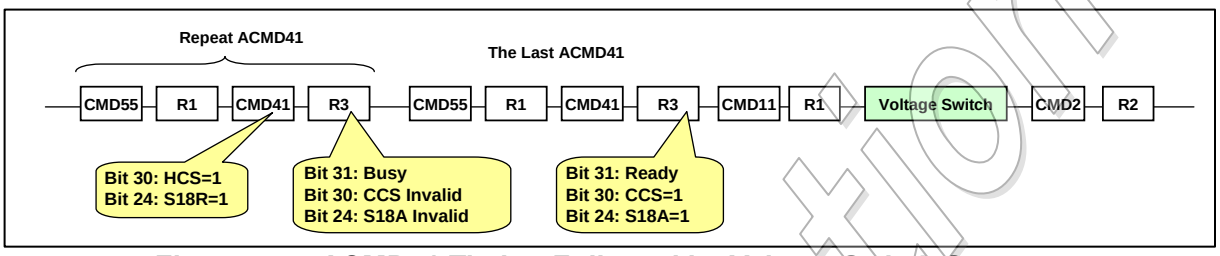


Figure 4-5 : ACMD41 Timing Followed by Voltage Switch Sequence

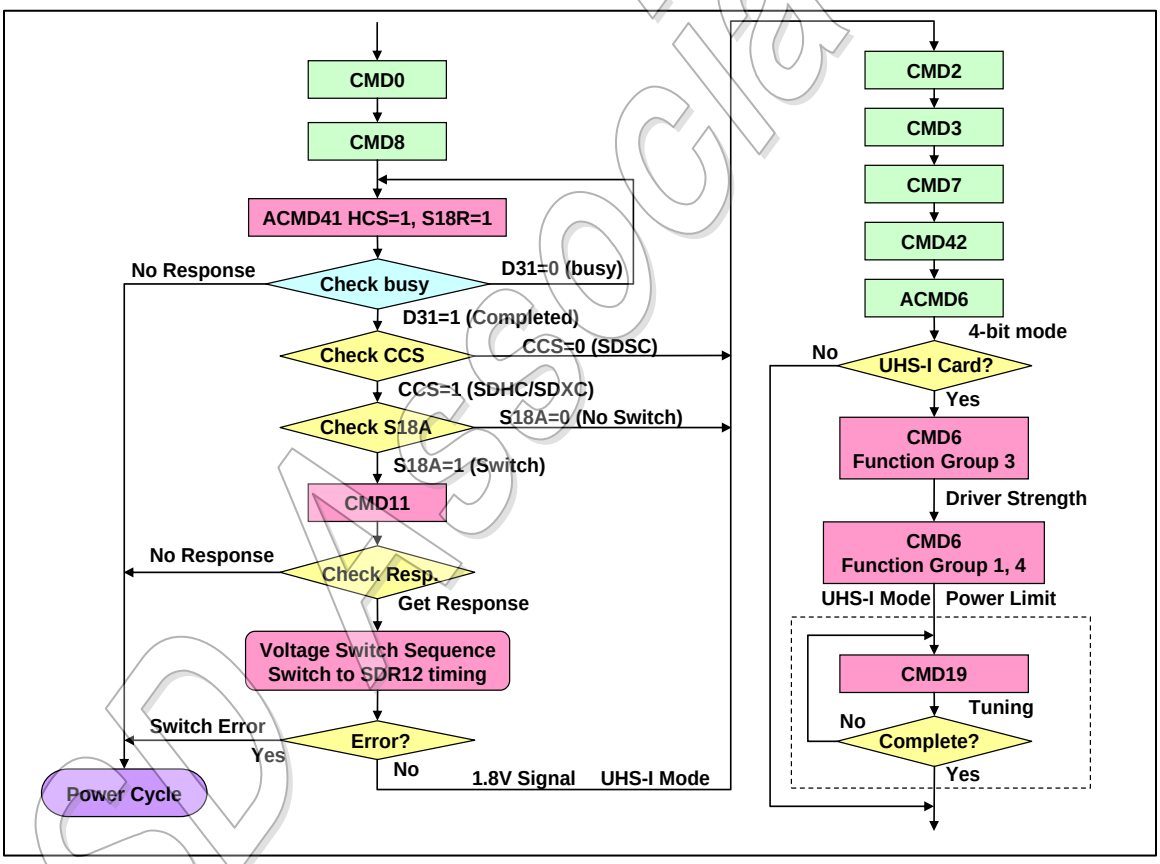


Figure 4-6 : UHS-I Host Initialization Flow Chart

When signaling level is 3.3V, host repeats to issue ACMD41 with HCS=1 and S18R=1 until the response indicates ready. The argument (HCS and S18R) of the first ACMD41 is effective but the all following ACMD41 should be issued with the same argument. If Bit 31 indicates ready, host needs to check CCS and S18A. The card indicates S18A=0, which means that voltage switch is not allowed and the host needs to use current signaling level. S18A=1 means that voltage switch is allowed and host issues CMD11 to invoke voltage switch sequence. By receiving CMD11, the card returns R1 response and start voltage

switch sequence. No response of CMD11 means that S18A was 0 and therefore host should not have sent CMD11. Completion of voltage switch sequence is checked by high level of DAT[3:0]. Any bit of DAT[3:0] can be checked depends on ability of the host.
When entering tran state, CARD_IS_LOCKED status in the R1 response should be checked (it is indicated in the response of CMD7). If the card is locked, CMD42 is required to unlock the card. If the card is unlocked, CMD42 can be skipped.
In case of UHS-I card, appropriate driver strength is selected by CMD6 Function Group 3 and one of UHS-I modes is selected by CMD6 Function Group 1.
In SDR50 and SDR104 modes, if tuning of sampling point is required, CMD19 is repeatedly issued until tuning is completed.

4.2.4.2 Timing to Switch Signal Voltage

This section is a blank in the Simplified Specification.

4.2.4.3 Timing of Voltage Switch Error Detection

This section is a blank in the Simplified Specification.

4.2.4.4 Voltage Switch Command

Figure 4-9 shows Voltage Switch Command (CMD11) definition. CMD11 can be executed in ready state and doesn't change the state. Even if the card is locked, CMD11 can be executed. Returning R1 type response means the card starts voltage switch sequence. If the host detects no response, power cycle should be executed.
There are four cases that the card indicates no response to CMD11.

(1) The card does not support voltage switch.
(2) The card supports voltage switch but ACMD41 is received with S18R=0.
(3) The card receives CMD11 not in ready state.
(4) Signaling level is already switched to 1.8V.

For all above cases, CMD11 is treated as an illegal command.

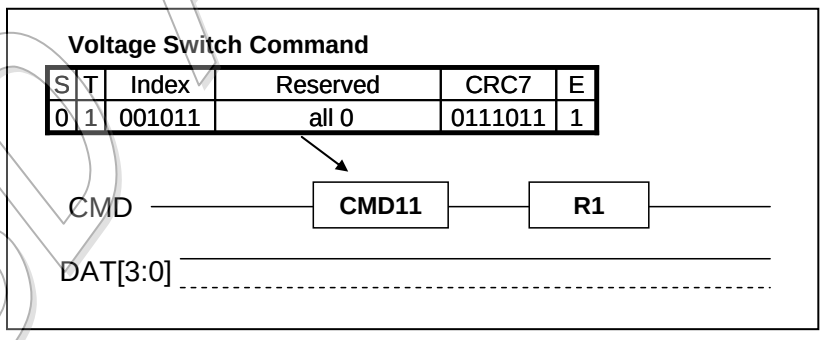


Figure 4-9 : Voltage Switch Command

4.2.4.5 Tuning Command

A known data block ("Tuning block") can be used to tune sampling point for tuning required hosts. The tuning capability of sampling point is mandatory for HOST-SDR-VD and optional for HOST-SDR-FD. This

procedure gives the system optimal timing for each specific host and card combination and compensates for static delays in the timing budget including process, voltage and different PCB loads and skews.

CMD19 is defined for Send Tuning Block Command. R1 type response is defined. CMD19 can be executed in transfer state of 1.8V signaling mode while the card is unlocked. The other case, CMD19 is treated as illegal command. Data block, carried by DAT[3:0], contains a pattern for tuning sampling position to receive data on the CMD and DAT[3:0] line. The block length of CMD19 is fixed and CMD16 is not required.

The tuning command (CMD19) follows the timing of the single block read command as described in Figure 4-10.

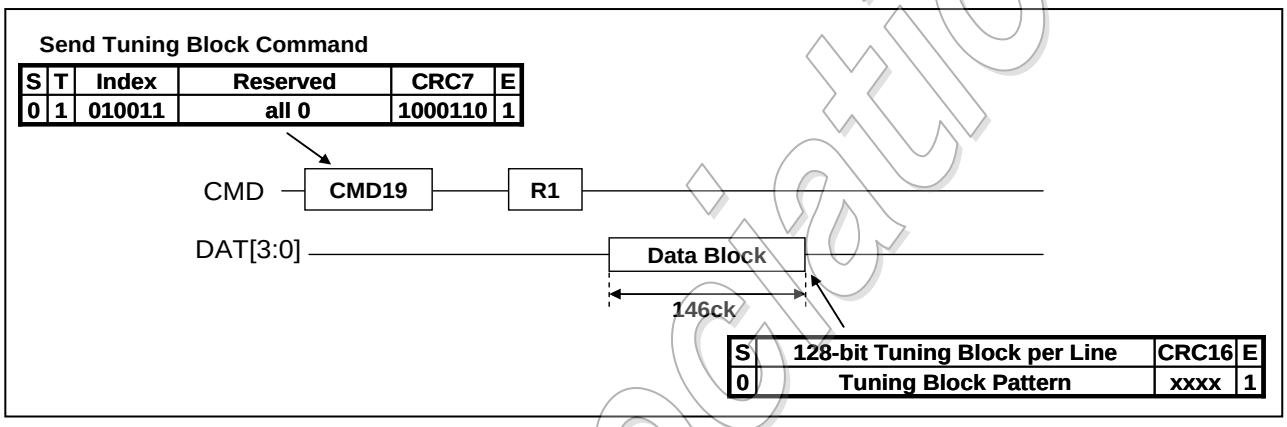


Figure 4-10 : Send Tuning Block Command

This sequence is defined as multiple, consecutive executions of CMD19 that are sent from the host and responded by the card, without any other command mixed between them.
The card shall complete a sequence of 40 times CMD19 executions in no more than 150ms. The tuning process is normally shorter than 40 executions of CMD19, and therefore should be shorter than 150ms. The sequence period definition does not include any host processing time. If host needs time to process CMD19 between executions, the sequence may be longer by this amount of time.

FF0FFF00	FFCCC3CC	C33CCCFF	FEFFFEEF
FFDFFFDD	FFFBFFFB	BFFF7FFF	77F7BDEF
FFF0FFF0	0FFCCC3C	CC33CCCF	FFEFFFEE
FFFDFFFD	DFFFBFFF	BBFFF7FF	F77F7BDE

Table 4-3 : Tuning Block Pattern

This section is a blank in the Simplified Specification.

4.2.4.6 An Example of UHS-I System Block Diagram

This section is a blank in the Simplified Specification.

4.3 Data Transfer Mode

Until the end of Card Identification Mode the host shall remain at f_{OD} frequency because some cards may have operating frequency restrictions during the card identification mode. In Data Transfer Mode the host may operate the card in f_{PP} frequency range (see Section 6.6.6). The host issues SEND_CSD (CMD9) to obtain the Card Specific Data (CSD register), e.g. block length, card storage capacity, etc.

The broadcast command SET_DSR (CMD4) configures the driver stages of all identified cards. It programs their DSR registers corresponding to the application bus layout (length) and the number of cards on the bus and the data transfer frequency. The clock rate is also switched from f_{OD} to f_{PP} at that point. SET_DSR command is an option for the card and the host.

CMD7 is used to select one card and put it into the *Transfer State*. Only one card can be in the *Transfer State* at a given time. If a previously selected card is in the *Transfer State* its connection with the host is released and it will move back to the *Stand-by State*. When CMD7 is issued with the reserved relative card address "0x0000", all cards are put back to *Stand-by State* (Note that it is the responsibility of the Host to reserve the RCA=0 for card de-selection - refer to Table 4-23, CMD7.

SD Memory Card State Diagram in UHS-II is defined by the UHS-II Addendum.

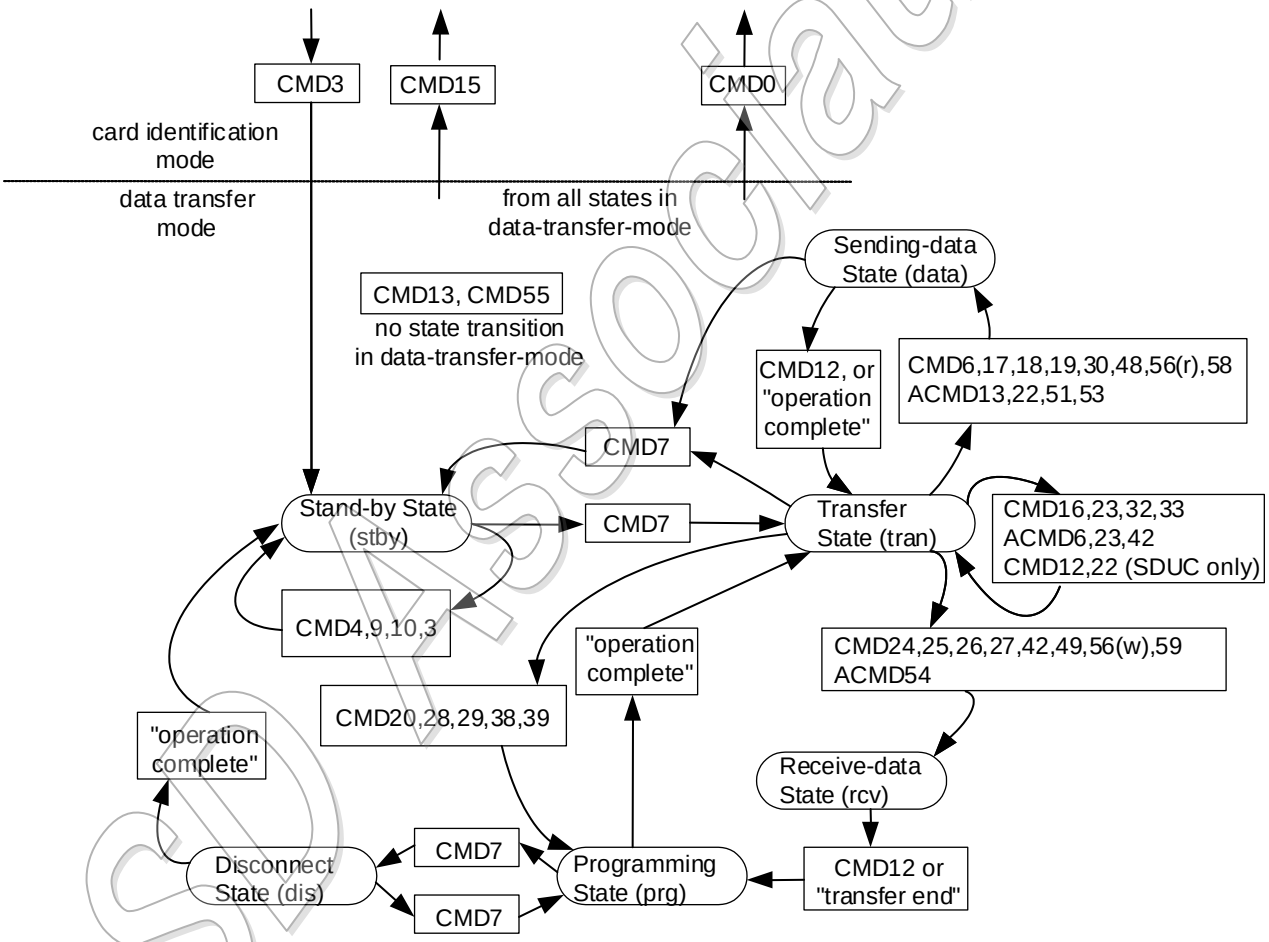


Figure 4-13 : SD Memory Card State Diagram (data transfer mode)

This may be used before identifying new cards without resetting other already registered cards. Cards which already have an RCA do not respond to identification commands (ACMD41, CMD2, see Section 4.2.3) in this state. CMD48, CMD49, CMD58 and CMD59 are also available in UHS-II mode.

Important Note: The card de-selection is done if certain card gets CMD7 with un-matched RCA. That

happens automatically if selection is done to another card and the CMD lines are common. So, in SD Memory Card system it will be the responsibility of the host either to work with common CMD line (after initialization is done) - in that case the card de-selection will be done automatically or if the CMD lines are separate then the host shall be aware to the necessity to de-select cards.
All data communication in the Data Transfer Mode is point-to point between the host and the selected card (using addressed commands). All addressed commands get acknowledged by a response on the CMD line.

The relationship between the various data transfer modes is summarized below.

- All data read commands can be aborted any time by the stop command (CMD12). The data transfer will terminate and the card will return to the *Transfer State.* The read commands are such as block read (CMD17), multiple block read (CMD18), send write protect (CMD30), send SCR (ACMD51) and general command in read mode (CMD56). (Refer to Figure 4-13 for more details.)
- All data write commands can be aborted any time by the stop command (CMD12). The write commands shall be stopped prior to deselecting the card by CMD7. The write commands are such as block write (CMD24 and CMD25), program CSD (CMD27), lock/unlock command (CMD42) and general command in write mode (CMD56). (Refer to Figure 4-13 for more details.)
- As soon as the data transfer is completed, the card will exit the data write state and move either to the *Programming State* (transfer is successful) or *Transfer State* (transfer failed).
- If a block write operation is stopped and the block length and CRC of the last block are valid, the data will be programmed.
- The card may provide buffering for block write. This means that the next block can be sent to the card while the previous is being programmed.
  If all write buffers are full, and as long as the card is in *Programming State* (see SD Memory Card state diagram Figure 4-13), the DAT0 line will be kept low (BUSY).
- There is no buffering option for write CSD, write protection and erase. This means that while the card is busy servicing any one of these commands, no other data transfer commands will be accepted. DAT0 line will be kept low as long as the card is busy and in the *Programming State*. Actually if the CMD and DAT0 lines of the cards are kept separated and the host keep the busy DAT0 line disconnected from the other DAT0 lines (of the other cards) the host may access the other cards while the card is in busy.
- Parameter set commands are *not* allowed while card is programming.
  Parameter set commands are: set block length (CMD16), erase block start (CMD32) and erase block end (CMD33).
- Read commands are *not* allowed while card is programming.
- Moving another card from *Stand-by* to *Transfer State* (using CMD7) will not terminate erase and programming operations. The card will switch to the *Disconnect State* and will release the DAT line.
- A card can be reselected while in the *Disconnect State*, using CMD7. In this case the card will move to the *Programming State* and reactivate the busy indication.
- Resetting a card (using CMD0 or CMD15) will terminate any pending or active programming operation. This may destroy the data contents on the card. It is the host's responsibility to prevent this.
- CMD34-37, CMD50 and CMD57 are reserved for SD command system expansion. State transitions for these commands are defined in each command system specification.

4.3.1 Wide Bus Selection/Deselection

Wide Bus (4 bit bus width) operation mode may be selected/deselected using ACMD6. The default bus width after power up or GO_IDLE (CMD0) is 1 bit bus width.
In order to change the bus width two conditions shall be met:
a) The card is in '*tran state'*.
b) The card is not locked

A locked card will responds to ACMD6 as illegal command.

4.3.2 2 GByte Card

To make 2GByte card, the Maximum Block Length (READ_BL_LEN=WRITE_BL_LEN) shall be set to 1024 bytes. However, the Block Length, set by CMD16, shall be up to 512 bytes to keep consistency with 512 bytes Maximum Block Length cards (Less than or equal 2GByte cards).

4.3.3 Data Read

The DAT bus line level is high by the pull-up when no data is transmitted. A transmitted data block consists of start bits (1 or 4 bits LOW), followed by a continuous data stream. The data stream contains the payload data (and error correction bits if an off-card ECC is used). The data stream ends with end bits (1 or 4 bits HIGH). The data transmission is synchronous to the clock signal. The payload for block oriented data transfer is protected by 1 or 4 bits CRC check sum (See Section 4.5).
The Read operation from SD Memory Card may be interrupted by turning the power off. The SD Memory Card ensures that data is not destroyed during all the conditions except write or erase operations issued by the host even in the event of sudden shut down or removal.
Read command is rejected if BLOCK_LEN_ERROR or ADDRESS_ERROR occurred and no data transfer is performed.

- **Block Read**

Block read is block oriented data transfer. The basic unit of data transfer is a block whose maximum size is always 512 bytes. Smaller blocks whose starting and ending address are entirely contained within 512 bytes boundary may be transmitted.
Block Length set by CMD16 can be set up to 512 bytes regardless of READ_BL_LEN.
A CRC is appended to the end of each block ensuring data transfer integrity. CMD17 (READ_SINGLE_BLOCK) initiates a block read and after completing the transfer, the card returns to the *Transfer State*. CMD18 (READ_MULTIPLE_BLOCK) starts a transfer of several consecutive blocks. Blocks will be continuously transferred until a STOP_TRANSMISSION command (CMD12) is issued. The stop command has an execution delay due to the serial command transmission. The data transfer stops after the end bit of the stop command.
When the last block of user area is read using CMD18, the host should ignore OUT_OF_RANGE error that may occur even the sequence is correct.
If the host uses partial blocks whose accumulated length is not block aligned and block misalignment is not allowed, the card shall detect a block misalignment at the beginning of the first misaligned block, set the ADDRESS_ERROR error bit in the status register, abort transmission and wait in the *Data State* for a stop command.

Table 4-4 defines the card behavior when a partial block accesses is enabled.
If the misaligned block is the first data block of the command (i.e. ADDRESS_ERROR was reported in the actual response to the command), then no data is transferred and the card remains in the TRAN state.

CSD value			Current Blocklen *1	Read CMD Start Address
Max block size READ_BL_LEN	Misalign	Partial		
512Bytes	0 (Disable)	1 (Enable)	1- 512 bytes	Any address is accepted. *2
1kBytes	0 (Disable)	1 (Enable)	1- 512 bytes	Any address is accepted. *2
2kBytes	0 (Disable)	1 (Enable)	1- 512 bytes	Any address is accepted. *2

*1: "Current Blocklen" size is set or changed by CMD16. If value is less than or equal 512 bytes (There are no relations with Misalign and Partial option), it is set with no error.
*2: When the Blocklen size data range crosses 512 bytes block boundary, card outputs the data until the 512 bytes block boundary" and then the data becomes invalid and CRC error also may occur. The card will send "ADDRESS_ERROR" on the next command response. Host should issue CMD12 to recover.

Table 4-4 : Read Command Blocklen

4.3.4 Data Write

The data transfer format is similar to the data read format. For block oriented write data transfer, the CRC check bits are added to each data block. The card performs 1 or 4 bits CRC parity check (See Section 4.5) for each received data block prior to the write operation. By this mechanism, writing of erroneously transferred data can be prevented.
Write command is rejected if BLOCK_LEN_ERROR or ADDRESS_ERROR occurred and no data transfer is performed.

- **Block Write**

During block write (CMD24 - 27, 42, 56(w)) one or more blocks of data are transferred from the host to the card with 1 or 4 bits CRC appended to the end of each block by the host. A card supporting block write shall be required that Block Length set by CMD16 shall be 512 bytes regardless of WRITE_BL_LEN is set to 1k or 2k bytes.
Table 4-5 defines the card behavior when partial block accesses is disabled (WRITE_BL_PARTIAL = 0).

CSD value			Current Blocklen *1	Write CMD Start Address
Max block size WRITE_BL_LEN	Misalign	Partial		
512Bytes	0 (Disable)	0 (Disable)	512 bytes *2	n * 512 bytes *3 (n: Integer)
1kBytes	0 (Disable)	0 (Disable)	512 bytes *2	n * 512 bytes *3 (n: Integer)
2kBytes	0 (Disable)	0 (Disable)	512 bytes *2	n * 512 bytes *3 (n: Integer)

*1: "Current Blocklen" size is set or changed by CMD16. If value is less than 512 bytes (there are no relations with Misalign and Partial option), it is set with no error. And then "Current Blocklen" size is tested when write command execution.
*2: If the current Blocklen is other than this value, the card indicates "BLOCK_LEN_ERROR" on the Write command response.
*3: If start address is other than this value, the card will send "ADDRESS_ERROR" on the Write command response.

Table 4-5 : Write Command Blocklen

If WRITE_BL_PARTIAL is allowed (=1) then smaller blocks, up to resolution of one byte, can be used as well. If the CRC fails, the card shall indicate the failure on the DAT line (see below); the transferred data will be discarded and not be written, and all further transmitted blocks (in multiple block write mode) will be ignored.

Multiple block write command shall be used rather than continuous single write command to make faster write operation.
If the host uses partial blocks whose accumulated length is not block aligned and block misalignment is not allowed (CSD parameter WRITE_BLK_MISALIGN is not set), the card shall detect the block misalignment error and abort programming before the beginning of the first misaligned block. The card shall set the ADDRESS_ERROR error bit in the status register, and while ignoring all further data transfer, wait in the *Receive-data-State* for a stop command.
Note that the first data block is misaligned for write command (i.e. ADDRESS_ERROR is reported in the actual response of the write command), the card remains in tran state and no data is programmed.
The write operation shall also be aborted if the host tries to write over a write protected area. In this case, however, the card shall set the WP_VIOLATION bit.
Programming of the CSD register does not require a previous block length setting. The transferred data is also CRC protected. If a part of the CSD register is stored in ROM, then this unchangeable part shall match the corresponding part of the receive buffer. If this match fails, then the card will report an error and not change any register contents.
Some cards may require long and unpredictable times to write a block of data. After receiving a block of data and completing the CRC check, the card will begin writing and hold the DAT0 line low if its write buffer is full and unable to accept new data from a new WRITE_BLOCK command. The host may poll the status of the card with a SEND_STATUS command (CMD13) at any time, and the card will respond with its status. The status bit READY_FOR_DATA indicates whether the card can accept new data or whether the write process is still in progress). The host may deselect the card by issuing CMD7 (to select a different card) which will displace the card into the *Disconnect State* and release the DAT line without interrupting the write operation. When reselecting the card, it will reactivate busy indication by pulling DAT to low if programming is still in progress and the write buffer is unavailable. Actually, the host may perform simultaneous write operation to several cards with inter-leaving process. The interleaving process can be done by accessing each card separately while other cards are in busy. This process can be done by proper CMD and DAT0-3 line manipulations (disconnection of busy cards).

- **Pre-erased Setting prior to a Multiple Block Write Operation**

Setting a number of write blocks to be pre-erased (ACMD23) will make a following Multiple Block Write operation faster compared to the same operation without preceding ACMD23. The host will use this command to define how many number of write blocks are going to be send in the next write operation. If the host will terminate the write operation (Using stop transmission) before all the data blocks sent to the card the content of the remaining write blocks is undefined(can be either erased or still have the old data). If the host will send more number of write blocks than defined in ACMD23 the card will erase block one by one(as new data is received). This number will be reset to the default (=1) value after Multiple Blocks Write operation.
It is recommended using this command preceding CMD25, some of the cards will be faster for Multiple Write Blocks operation. Note that the host should send ACMD23 just before WRITE command if the host wants to use the pre-erased feature. If not, pre-erase-count might be cleared automatically when another commands (ex: Security Application Commands) are executed.

- **Send Number of Written Blocks**

Systems that use Pipeline mechanism for data buffers management are, in some cases, unable to determine which block was the last to be well written to the flash if an error occurs in the middle of a Multiple Blocks Write operation. The card will respond to ACMD22 with the number of well written blocks.

4.3.5 Erase/Discard/FULE

4.3.5.1 Erase

It is desirable to erase many write blocks simultaneously in order to enhance the data throughput. Identification of these write blocks is accomplished with the ERASE_WR_BLK_START (CMD32), ERASE_WR_BLK_END (CMD33) commands.
The host should adhere to the following command sequence: ERASE_WR_BLK_START, ERASE_WR_BLK_END and ERASE (CMD38). If 00000001h or 000000002h is set in argument of CMD38, ERASE may not be executed if card supports DISCARD or FULE.

If an erase (CMD38) or address setting (CMD32, 33) command is received out of sequence, the card shall set the ERASE_SEQ_ERROR bit in the status register and reset the whole sequence.
If an out of sequence command (except SEND_STATUS) is received, the card shall set the ERASE_RESET status bit in the status register, reset the erase sequence and execute the last command.
If the erase range includes write protected sectors, they shall be left intact and only the non-protected sectors shall be erased. The WP_ERASE_SKIP status bit in the status register shall be set.
The address field in the address setting commands is a write block address in byte units. The card will ignore all LSB's below the WRITE_BL_LEN (see CSD) size.
As described above for block write, the card will indicate that an erase is in progress by holding DAT0 low. The actual erase time may be quite long, and the host may issue CMD7 to deselect the card or perform card disconnection, as described in the Block Write section, above.
The data at the card after an erase operation is either '0' or '1', depends on the card vendor.
The SCR register bit DATA_STAT_AFTER_ERASE (bit 55) defines whether it is '0' or '1'.

4.3.5.2 Discard

The DISCARD operation is similar to the default ERASE operation. Card may de-allocate the discarded blocks partially or completely. The contents of a region where the discard function has been applied shall be treated as ‘don’t care’ by host. After discard operation the previously written data may be partially or fully read by the host, depending on the card implementation.

It is desirable to discard many write blocks simultaneously in order to enhance the data throughput. Identification of these write blocks is accomplished with the ERASE_WR_BLK_START (CMD32), ERASE_WR_BLK_END (CMD33) commands. The host should adhere to the following command sequence: ERASE_WR_BLK_START, ERASE_WR_BLK_END and DISCARD (CMD38) and CMD38 arguments shall be set as 00000001h. The host which supports DISCARD shall check for DISCARD_SUPPORT bit (b313) in SD_STATUS register. If the card does not support DISCARD, the host shall not issue DISCARD command. If the card does not support DISCARD, card shall execute ERASE.
If DISCARD (CMD38) or address setting (CMD32, 33) command is received out of sequence, the card shall set the ERASE_SEQ_ERROR bit in the status register and reset the whole sequence.
If an out of sequence command (except SEND_STATUS) is received, the card shall set the ERASE_RESET status bit in the status register, reset the DISCARD sequence and execute the last command.
If the host provides an out of range address as an argument to CMD32 or CMD33, the card shall indicate OUT_OF_RANGE error in R1 (ERX) for CMD38 or in R1 of next command and reset the whole DISCARD sequence. Host shall check for any errors after DISCARD command.

The DISCARD is supported for SDHC and SDXC cards only and not supported through SPI interface.

As described above for block write, the card shall indicate that DISCARD is in progress by holding DAT0 low. The busy timeout is 250ms per DISCARD command.
The host may issue CMD7 to deselect the card or perform card disconnection, as described in the Block Write section.

On executing power cycle, card forgets pending Discard. Figure 4.3.5-1 shows the flowchart of card behavior for erase command sequence, including DISCARD and FULE operations

Application Note: Host is recommended not to issue DISCARD command to the file system area, since the content of the discarded area shall be 'don't care'

4.3.5.3 Full User Area Logical Erase (FULE)

The FULE operation is similar to the default ERASE operation, except the card shall logically erase the entire user area completely.
It is desirable to logically erase the complete user area. Identification of the full user area is accomplished with the ERASE_WR_BLK_START (CMD32), argument equal to 0, and ERASE_WR_BLK_END (CMD33), argument equal to end of user area, commands.
The host should adhere to the following command sequence: ERASE_WR_BLK_START, ERASE_WR_BLK_END and FULE (CMD38) and CMD38 arguments shall be set as 00000002h

The host that supports FULE shall check for FULE_SUPPORT bit (b312) in SD_STATUS register. If the card does not support FULE, host shall not issue FULE command. If the card does not support FULE, card shall execute ERASE.
If FULE (CMD38) or address setting (CMD32, 33) command is received out of sequence, the card shall set the ERASE_SEQ_ERROR bit in the status register and reset the whole sequence.
If an out of sequence command (except SEND_STATUS) is received, the card shall set the ERASE_RESET status bit in the status register, reset the FULE sequence and execute the last command.
If the host does not provide the full user area using CMD32 and CMD33, then card shall execute ERASE operation with the busy timeout as specified in Section 4.10.2.5 and Section 4.10.2.6. The FULE is supported for SDHC and SDXC cards only and not supported through SPI interface.

As described above for block write, the card shall indicate that FULE is in progress by holding DAT0 low. The busy timeout is 1 second per FULE command.
The host may issue CMD7 to deselect the card or perform card disconnection, as described in the Block Write section.

Figure 4.3.5-1 shows the flowchart of card behavior for erase command sequence, including DISCARD and FULE operations.

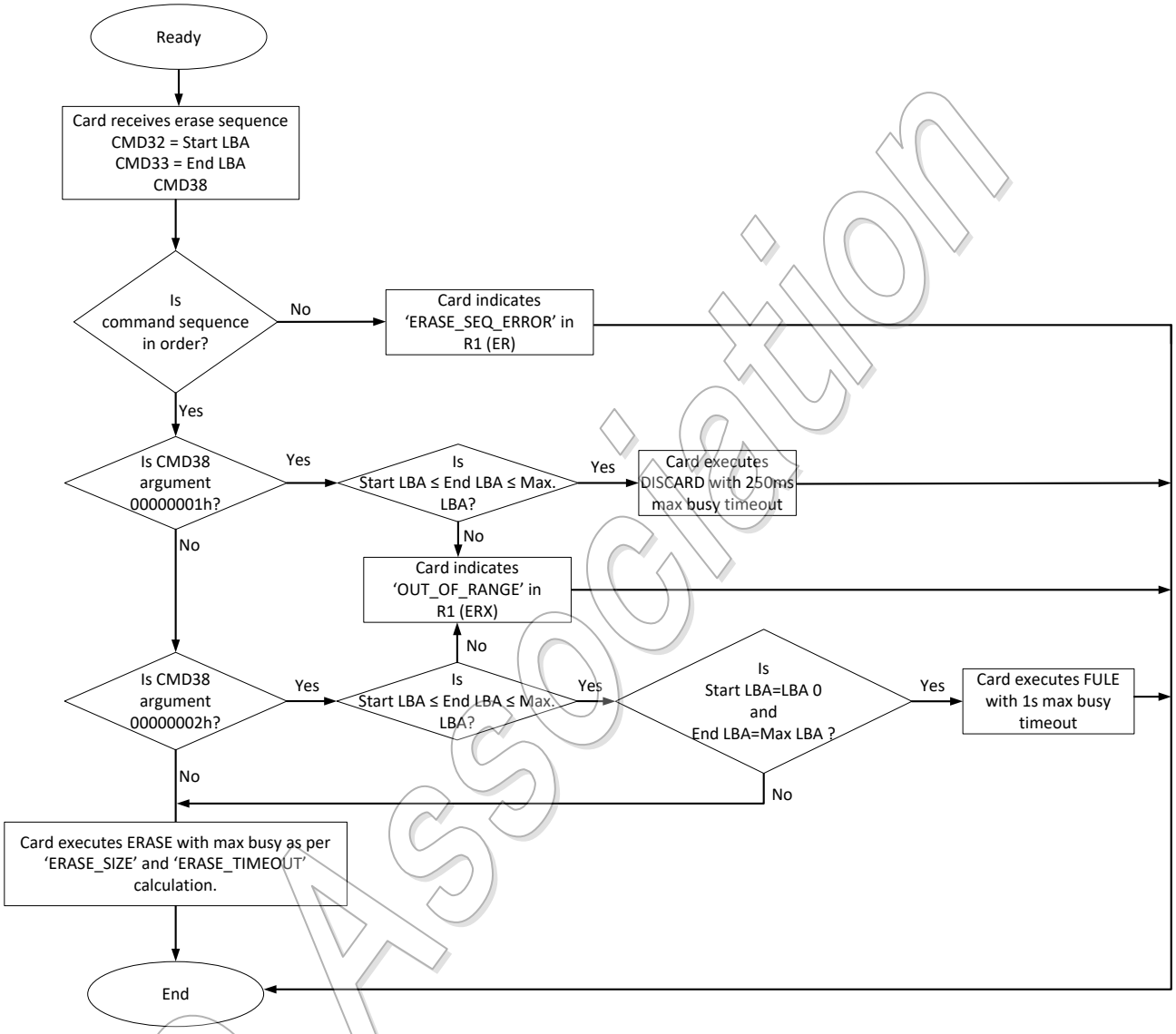


Figure 4.3.5-1 : Flowchart of Card Behavior for Erase Command Sequence

Application Note: If the card does not support DISCARD or FULE,

- The host may proceed with ERASE operation
- The host may proceed to overwrite data

4.3.6 Write Protect Management

Three write protect methods are supported in the SD Memory Card as follows:
- Mechanical write protect switch (Host responsibility only)
- Card internal write protect (Card's responsibility)
- Password protection card lock operation.

- **Mechanical Write Protect Switch**

  A mechanical sliding tablet on the side of the card (refer to Part 1 Standard Size SD Card Mechanical Addendum) will be used by the user to indicate that a given card is write protected or not. If the sliding tablet is positioned in such a way that the window is open it means that the card is write protected. If the window is close the card is not write-protected.
  A proper, matched, switch on the socket side will indicate to the host that the card is write-protected or not. It is the responsibility of the host to protect the card. The position of the write protect switch is unknown to the internal circuitry of the card.

- **Card's Internal Write Protection**

  This section is a blank in the Simplified Specification.

4.3.7 Card Lock/Unlock Operation

4.3.7.1 General

The password protection feature enables the host to lock a card while providing a password, which later will be used for unlocking the card. The password and its size are kept in a 128-bit PWD and 8-bit PWD_LEN registers, respectively. Optionally, Force Erase Password can be stored in additional 128-bit FEP and 8-bit FEP_LEN registers shall be present in case Card Ownership Protection (COP) is supported by the card. These registers are non-volatile so that a power cycle will not erase them.
Locked cards respond to (and execute) all commands in the "basic" command class (class 0), ACMD41, CMD16 and "lock card" command class. Thus, the host is allowed to reset, initialize, select, query for status, etc., but not to access data on the card. If the password was previously set (the value of PWD_LEN is not 0), the card will be locked automatically after power on.
Similar to the existing CSD register write commands, the lock/unlock command is available in "transfer state" only. This means that it does not include an address argument and the card shall be selected before using it.
The card lock/unlock command has the structure and bus transaction type of a regular single block write command. The transferred data block includes all the required information of the command (password setting mode, PWD itself, card lock/unlock etc.).
Content protection is managed by PWD. Physical Card Ownership is managed with COP feature.
The password protection mechanism defined in this section is effective to the User Area (including TCG MBR Table described in Section 4.22.2.1) and the Protected Area (unless otherwise noted such as for force erase). It is not effective to Boot Partitions, RPMB Unit and TCG Tables except TCG MBR Table for both SD and PCIe interfaces (see Figure 4-84).

4.3.7.1.1 Card Ownership Protection

Card Ownership Protection (COP) prevents re-use of COP Card without knowing the protection password PWD. Once Force Erase Password (FEP) is set, it disables “Force Erase” operation, and “Clear PWD” operations. Alternatively, “FEP Force Erase” can be used. As this behavior is different from the Physical Layer Specification Version 5.00 or earlier, COP Unlock is required to operate the card after power up when FEP=set, to ensure only the hosts familiar with COP operations can use such a card.

> Application Note:
> Card Ownership Protection is addressing specific user needs and applications. It is not recommended for support on host devices that are not designed for specific applications that know how to make use of this feature. It is recommended to support this feature only in cards specially designed to work with those designated hosts.

4.3.7.1.2 Special Terms for Lock/Unlock

The following terms and notations are introduced in this section:

- COP: Card Ownership Protection
  – COP feature: set of additional functions to support Card Ownership Protection
  – COP bit: bit4 of CMD42 argument that indicates COP-specific functions vs. PWD functions
- COP Card / Non-COP Card
  – COP Card is a card that supports COP features
  – Non-COP Card is another expression of Type 1, 2 Cards that does not support COP feature
- PWD: Card Lock Password
  – PWD indicates a register to hold a password to lock the card
  – "PWD=0" indicates that PWD is not set and PWD_LEN is set to 0
  – "PWD=set" indicates that a password is set to PWD and the password length is set to PWD_LEN
- FEP: Force Erase Password
  – "FEP" indicates a register to hold a password to protect Force Erase
  – "FEP=0" indicates that FEP is not set and FEP_LEN is set to 0
  – "FEP=set" indicates that a password is set to FEP and the password length is set to FEP_LEN
- Force Erase: is the function of CMD42[08h] (with no password) to erase PWD, PWD_LEN and all memory area. This function is disabled when FEP=set
- FEP Force Erase: is the function of CMD42[18h] (with FEP) to erase PWD, PWD_LEN and all memory area. This function is disabled when FEP=0
- COP Unlock
  – command CMD42 with argument 1Fh (notated as CMD42[1Fh]) that is required to access COP feature functions. Also, if FEP=set, after power on reset or reset COP Unlock is required to enable any other CMD42 functions.
- COP Locked state
  – state of the card after power on reset or reset, when any other argument to CMD42 is not accessible till COP Unlock is performed. Also, the card shall behave as locked card to any other commands.
- Clean COP Card - COP Card with no FEP/PWD set
- Locked COP Card – COP Card with PWD set
- PWD functions - COP Card CMD42 functions defined by Table 4.3.7-1
- COP functions - COP Card CMD42 functions defined by Table 4.3.7-2

4.3.7.1.3 Card Behavior

Figure 4.3.7-1 shows simplified behavior of Non-COP Card. After power on, card is "PWD Locked" if PWD=set or "PWD Unlocked" if PWD=0. Force Erase allows to reset the PWD, while erasing all the User area data.
CMD42[1Fh] is not defined and Non-COP Card returns LOCK_UNLOCK_FAILED error in R1.

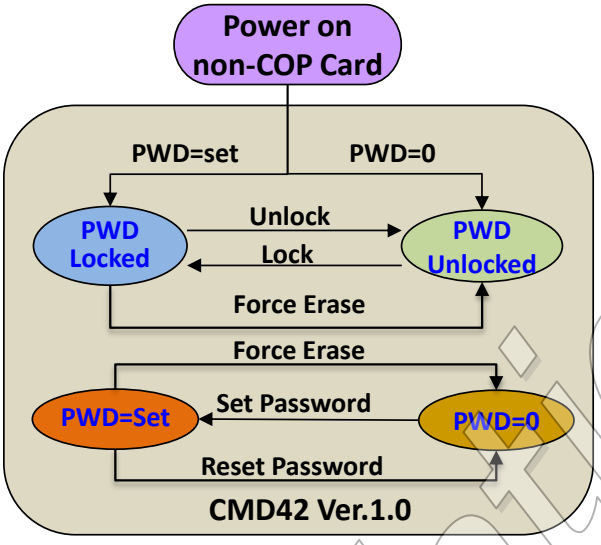


Figure 4.3.7-1 : Simplified Non-COP Card State Diagram

Figure 4.3.7-2 shows simplified behavior of a COP Card. After power on reset or reset, COP Card starts in CMD42 Ver.1.0 Compatible mode:

- If FEP=0, CMD42 functions are compatible with CMD42 Ver.1.0, except the support of additional argument 1Fh (COP Unlock).
- If FEP=1, COP Card is in COP Locked state to prevent access by Non-COP Hosts. COP Card does not execute any CMD42 function and indicates CARD_IS_LOCKED status and LOCK_UNLOCK_FAILED error except CMD42[1Fh]. On receiving CMD42[1Fh] COP Card moves to CMD42 Ver.2.0 mode.

In CMD42 Ver.2.0 mode, "FEP Locked" is disabled and CMD42 functions are extended with functions accessible with COP bit set to '1'. There are three additional features to Ver.1.0:

(1) FEP Management
FEP is managed by Reset FEP, Set FEP and Replace FEP functions.

(2) Force Erase
There are 2 types of Force Erase operation supported based on FEP state:
"Force Erase" (CMD42[08h] with no password) to erase PWD, PWD_LEN and all memory area can be executed as long as FEP=0. It is disabled if FEP=set.
"FEP Force Erase" (CMD42[18h] with password) to erase PWD, PWD_LEN and all memory area which if executed if the password matches FEP. It is enabled if FEP=set and disabled if FEP=0.

(3) Reset PWD inhibit
Once PWD is set, PWD can be replaced, but PWD cannot be reset during FEP=set.

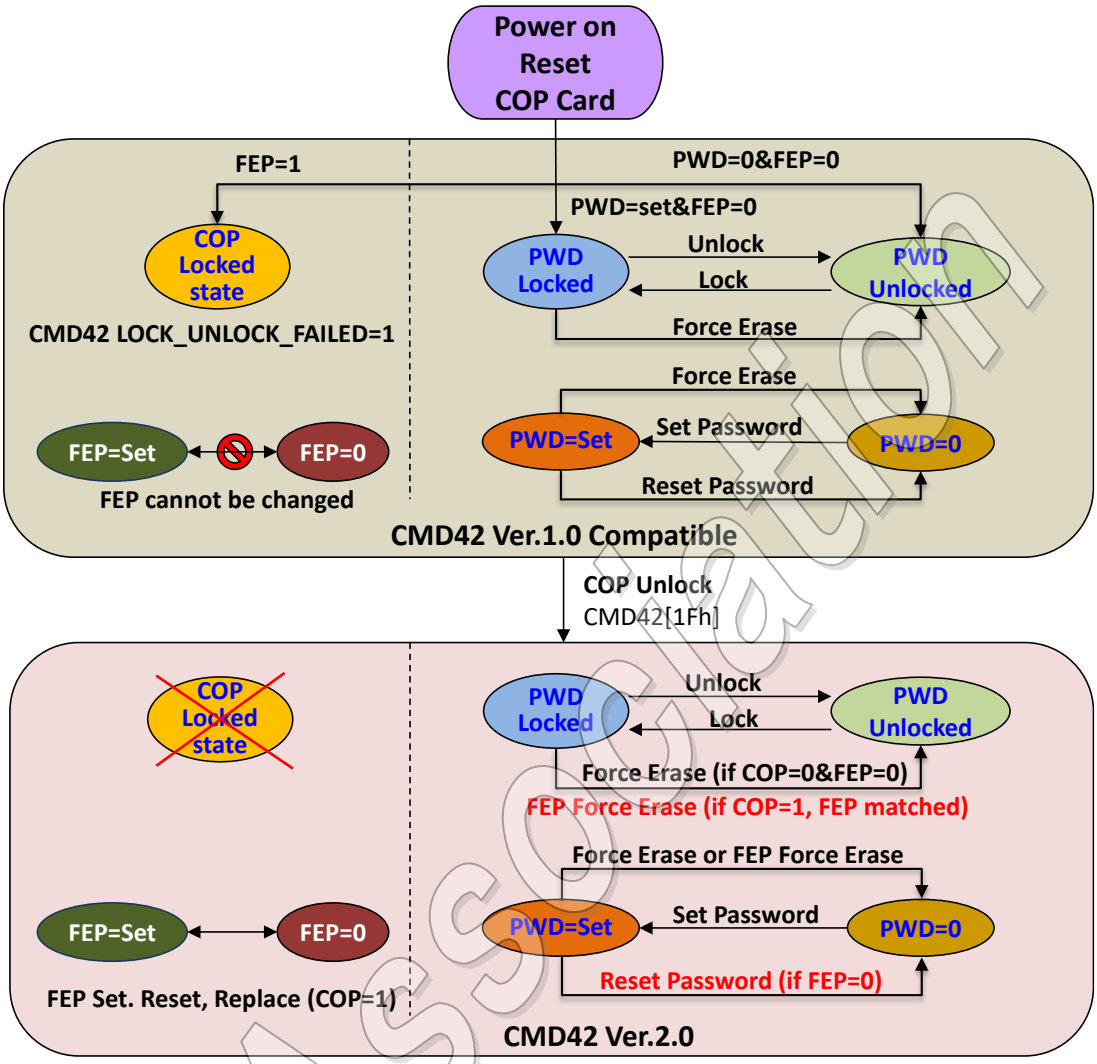


Figure 4.3.7-2 : Simplified COP Card State Diagram

4.3.7.1.4 Lock Card Data Structure

Table 4-6 describes the structure of the command data block. Note that the host compliant to the Physical Layer Specification Version 5.10 or later shall set reserved bits (Bit7-5) to 0 when issuing CMD42.

Byte #	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Reserved (shall be set to 0)			COP	ERASE	LOCK_ UNLOCK	CLR_ PWD	SET_ PWD
1	PWDS_LEN							
2	Password data							
...								
PWDS_LEN + 1								

Table 4-6 : Lock Card Data Structure

- **COP:** When set to 1, indicates Card Ownership Protection feature operations
- **ERASE:** 1 Defines Force Erase Operation. In byte 0, bit 3 will be set to 1 (all other bits shall be 0). All other bytes of this command will be ignored by the card.
- **LOCK/UNLOCK**: 1 = Locks the card. 0 = Unlock the card (note that it is valid to set this bit together with SET_PWD but it is not allowed to set it together with CLR_PWD).
- **CLR_PWD**: 1 = Clears PWD.
- **SET_PWD**: 1 = Set new password to PWD
- **PWDS_LEN**: Defines the following password(s) length (in bytes). In case of a password change, this field includes the total password lengths of old and new passwords. The password length is up to 16 bytes. In case of a password change, the total length of the old password and the new password can be up to 32 bytes.
- **Password data:** In case of setting a new password, it contains the new password. In case of a password change, it contains the old password followed by the new password.

The data block size shall be defined by the host before it sends the card lock/unlock command. The block length shall be set to greater than or equal to the required data structure of the lock/unlock command. In the following explanation, changing block size by CMD16 is not a mandatory requirement for the lock/unlock command. If preset block length is larger than length of required data structure, dummy data shall be sent by the host for unused data area of the block.
Since block length shall always be even in DDR50 mode, the block length for CMD42 shall always be rounded up to an even size. If CMD16 is used prior to CMD42 to set the block length, it shall always specify an even length.

4.3.7.1.5 Command Sequences Common to COP/Non-COP Cards

The following paragraphs define the various lock/unlock command sequences common to COP/Non-COP Cards (items relevant to COP Cards only are marked by square brackets []):

- **Setting PWD**
  - Select a card (CMD7), if not previously selected.
  - Define the block length (CMD16), given by the 8-bit card lock/unlock mode, the 8-bits password size (in bytes), and the number of bytes of the new password. In the case that a password replacement is done, then the block size shall consider that both passwords-the old and the new one-are sent with the command.

- Send the Card Lock/Unlock command with the appropriate data block size on the data line including the 16-bit CRC. The data block shall indicate the mode (SET_PWD), the length (PWDS_LEN) and the password itself. In the case that a password replacement is done, then the length value (PWDS_LEN) shall include both passwords (the old and the new one) and the password data field shall include the old password (currently used) followed by the new password. Note that the card shall handle the calculation of the new password length internally by subtracting the old password length from PWDS_LEN field.
- [In CMD42 ver.2.0 mode, COP Card can execute this command when COP bit=0.]
- In the case that the sent old password is not correct (not equal in size and content), then the LOCK_UNLOCK_FAILED error bit will be set in the status register and the old password does not change. In the case that the sent old password is correct (equal in size and content), then the given new password and its size will be saved in the PWD and PWD_LEN registers, respectively.

Note that the password length register (PWD_LEN) indicates if a password is currently set. When it equals 0, there is no password set. If the value of PWD_LEN is not equal to zero, the card will lock itself after power up. COP Card shall also lock itself after CMD0. Non-COP Card lock status after CMD0 is up to implementation. It is possible to lock the card immediately in the current power session by setting the LOCK/UNLOCK bit (while setting the password) or sending an additional command for card lock.

- **Resetting PWD:**
  - Select a card (CMD7), if not previously selected.
  - Define the block length (CMD16), given by the 8-bit card lock/unlock mode, the 8-bit password size (in bytes), and the number of bytes of the currently used password.
  - Send the card lock/unlock command with the appropriate data block size on the data line including the 16-bit CRC. The data block shall indicate the mode CLR_PWD, the length (PWDS_LEN), and the password itself. If the PWD and PWD_LEN content match the sent password and its size, then the content of the PWD register is cleared and PWD_LEN is set to 0. If the password is not correct, then the LOCK_UNLOCK_FAILED error bit will be set in the status register.
  - [In CMD42 ver.2.0 mode, COP Card can execute this command when COP bit=0.]
  - [If FEP is set, PWD and PWD_LEN registers shall not be affected, and the LOCK_UNLOCK_FAILED error bit shall be set in the status register.]
  - [Resetting PWD shall not affect FEP and FEP_LEN registers.]

- **Locking PWD Unlocked Card:**
  - Select a card (CMD7), if not previously selected.
  - Define the block length (CMD16), given by the 8-bit card lock/unlock mode, the 8-bit password size (in bytes), and the number of bytes of the currently used password.
  - Send the card lock/unlock command with the appropriate data block size on the data line including the 16-bit CRC. The data block shall indicate the mode LOCK, the length (PWDS_LEN) and the password itself.
  - [In CMD42 ver.2.0 mode, COP Card can execute this command when COP bit=0.]

In PWD Unlocked state, if the PWD content is equal to the sent password, then the card will be PWD locked and the card-locked status bit will be set in the status register. If the password is not correct, then the LOCK_UNLOCK_FAILED error bit will be set in the status register.
Note that it is possible to set the password and to lock the card in the same sequence. In such a case, the host shall perform all the required steps for setting the password (as described above) including the bit LOCK set while the new password command is sent. If the password was previously set (PWD_LEN is not 0), then the card will be PWD locked automatically after power on reset.
An attempt to lock a locked card or to lock a card that does not have a password will fail and the

LOCK_UNLOCK_FAILED error bit will be set in the status register, unless it was done during a password definition or change operations.

- **Unlocking PWD Locked Card:**
  - Select a card (CMD7), if not previously selected.
  - [If FEP=set, after power on reset or reset with CMD0, COP Unlock is required (refer to Unlocking COP locked state sequence in Section 4.3.7.1.6)]
  - Define the block length (CMD16), given by the 8-bit card lock/unlock mode, the 8-bit password size (in bytes), and the number of bytes of the currently used password PWD.
  - Send the card lock/unlock command with the appropriate data block size on the data line including the 16-bit CRC. The data block shall indicate the mode UNLOCK, the length (PWDS_LEN) and the password itself.
  - [In CMD42 ver.2.0 mode, COP Card can execute this command when COP bit=0.]

If the PWD content is equal to the sent password, then the card will be unlocked and the card-locked status bit will be cleared in the status register. If the password is not correct, then the LOCK_UNLOCK_FAILED error bit will be set in the status register.
Note that unlocking is done only for the current power session. As long as the PWD [or FEP] is not cleared, the card will be locked automatically on the next power up. Only if PWD=0 [and FEP=0], the card shall be unlocked after power on and CMD0.
An attempt to unlock an unlocked card will fail and LOCK_UNLOCK_FAILED error bit will be set in the status register, unless it was done during PWD definition or change operation.

4.3.7.1.6 COP specific Command Sequences

The following paragraphs define the various lock/unlock command sequences applicable to COP Cards only:

- **Enabling Card Ownership Protection (COP) Function Set**

If FEP=0, after power on reset or reset COP Card shall not execute any COP function of CMD42 other than COP Unlock and shall interpret it as PWD function as defined by Table 4.3.7-1, unless it was done after COP functions are unlocked after the power on reset or reset by the sequence below:

- Select a card (CMD7), if not previously selected.
- Define the block length (CMD16) to 1 byte (8-bit card lock/unlock command). Send the card lock/unlock command with the appropriate data block of one byte on the data line including the 16 bit CRC. The data block shall have COP, ERASE, LOCK_UNLOCK, CLR_PWD, SET_PWD bits set to 1.

If COP Unlock was accepted, then the card shall go to CMD42 ver.2.0 mode. While FEP=0, the only COP function supported is set FEP. “Set PWD” command shall require COP bit to be set to 0, as described in the Section 4.3.7.2.
Before receiving COP Unlock, COP Card is CMD42 Ver.1.0 Compatible mode and CMD42 with COP bit=1 is ignored (except COP Unlock).
Note: For the card not supporting COP, the LOCK_UNLOCK_FAILED error bit shall be set in the status register and the COP enabling request is rejected.

- **Setting Force Erase Password (FEP):**
  - Unlock COP function set.
  - Define the block length (CMD16), given by the 8-bit card lock/unlock mode, the 8-bits password size (in bytes), and the number of bytes of the new password. In the case that a password replacement is done, then the block size shall consider that both passwords-the old and the new one-are sent with the command.

- Send the Card Lock/Unlock command with the appropriate data block size on the data line including the 16-bit CRC. The data block shall indicate the mode (SET_PWD and COP), the length (PWDS_LEN) and the password itself. In the case that a password replacement is done, then the length value (PWDS_LEN) shall include both passwords (the old and the new one) and the password data field shall include the old password (currently used FEP) followed by the new password. Note that the card shall handle the calculation of the new password length internally by subtracting the old password length from PWDS_LEN field.
- In the case that the sent old password is incorrect (not equal in size and content), then the LOCK_UNLOCK_FAILED error bit shall be set in the status register and the old password does not change. In the case that the sent old password is correct (equal in size and content), then the given new password and its size shall be saved in the FEP and FEP_LEN registers, respectively.
- Setting FEP automatically disables “Force Erase” and “Reset PWD” (CLR_PWD) operations. No power cycle is required.

Note that the password length register (FEP_LEN) indicates if a FEP is currently set. When it equals 0, there is no password set. Setting FEP is not affecting current card LOCK/UNLOCK state.

- **Unlocking COP Locked State**

If FEP=1, after power on reset or reset with CMD0, COP Card shall be in locked state (see Section 4.3.7.5) and shall fail any other CMD42 parameter and LOCK_UNLOCK_FAILED error bit shall be set in the status register, unless it was done after COP function set is unlocked after the power on reset or reset by the sequence below:

- Select a card (CMD7), if not previously selected.
- Define the block length (CMD16) to 1 byte (8-bit card lock/unlock command). Send the card lock/unlock command with the appropriate data block of one byte on the data line including the 16 bit CRC. The data block shall have COP, ERASE, LOCK_UNLOCK, CLR_PWD, SET_PWD bits set to 1. This command does not require any password to be included in data block.

If COP Unlock was accepted, then COP Card shall go to CMD42 ver.2.0 mode and processing of other CMD42 operations shall be enabled by the card. PWD commands shall require COP bit to be set to 0, as described in the Section 4.3.7.2.

Note: For the card not supporting COP, the LOCK_UNLOCK_FAILED error bit shall be set in the status register and the COP enabling request is rejected.

- **Resetting FEP:**
  - Unlock COP function set, if not unlocked already.
  - Define the block length (CMD16), given by the 8-bit card lock/unlock mode, the 8-bit password size (in bytes), and the number of bytes of the currently used password.
  - Send the card lock/unlock command with the appropriate data block size on the data line including the 16-bit CRC. The data block shall indicate the mode (CLR_PWD and COP), the length (PWDS_LEN), and the password itself. If the FEP and FEP_LEN content match the sent password and its size, then the content of the FEP register is cleared and FEP_LEN is set to 0. If the password is not correct, then the LOCK_UNLOCK_FAILED error bit shall be set in the status register.
  - If resetting FEP fails, FEP and FEP_LEN registers are maintained, and the LOCK_UNLOCK_FAILED error bit shall be set in the status register.
  - Resetting FEP shall not affect current card LOCK/UNLOCK state.
  - Resetting FEP shall not affect PWD and PWD_LEN registers.
  - Clearing FEP enables “Force Erase” and “Reset PWD” operations.
  - Clearing FEP disables “FEP Force Erase” and other COP operations.

4.3.7.1.7 Card Ownership Protection Flowcharts

Figure 4.3.7-3 presents COP protection setup sequence (left) and FEP Force Erase sequence (right).
Figure 4.3.7-4 presents Lock/Unlock operations sequence on COP Card with FEP set.

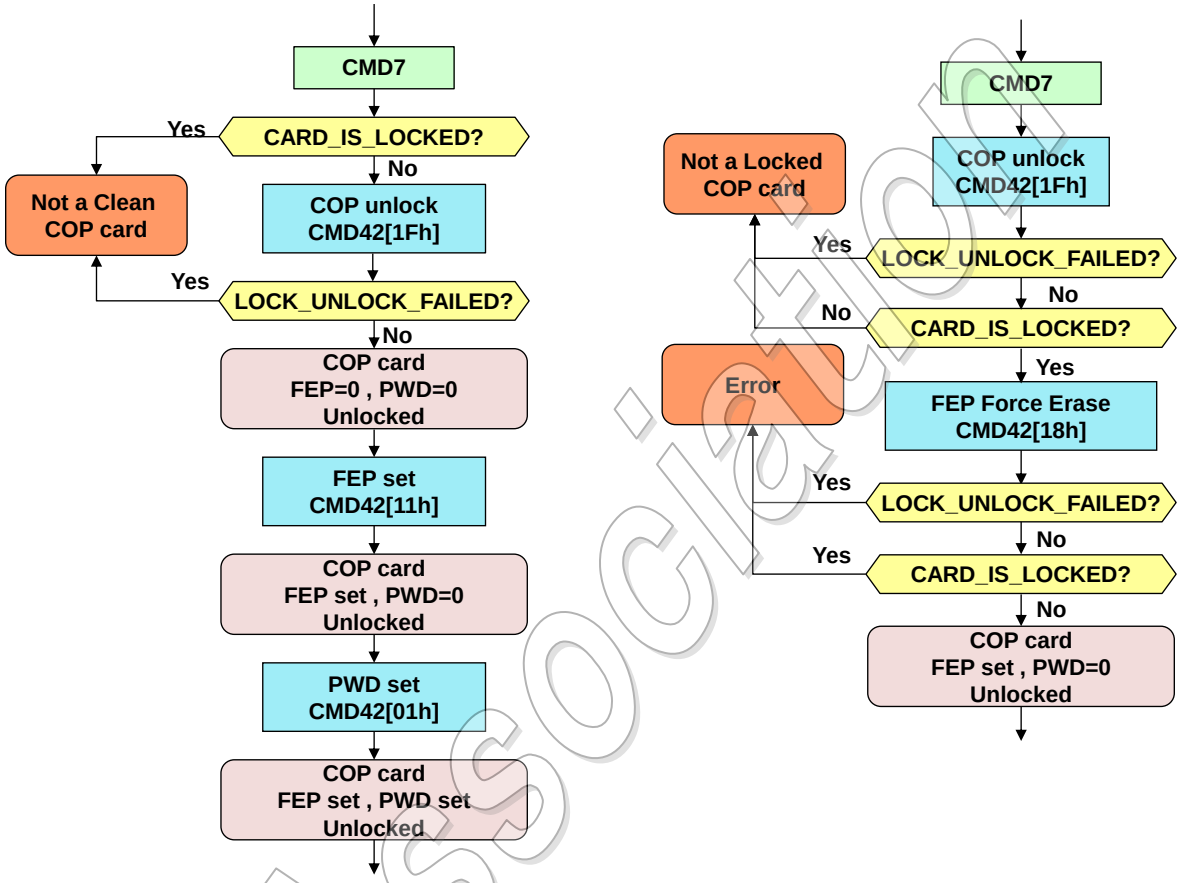


Figure 4.3.7-3 : COP Card Protection Setup (left) and FEP Force Erase (right) Sequences

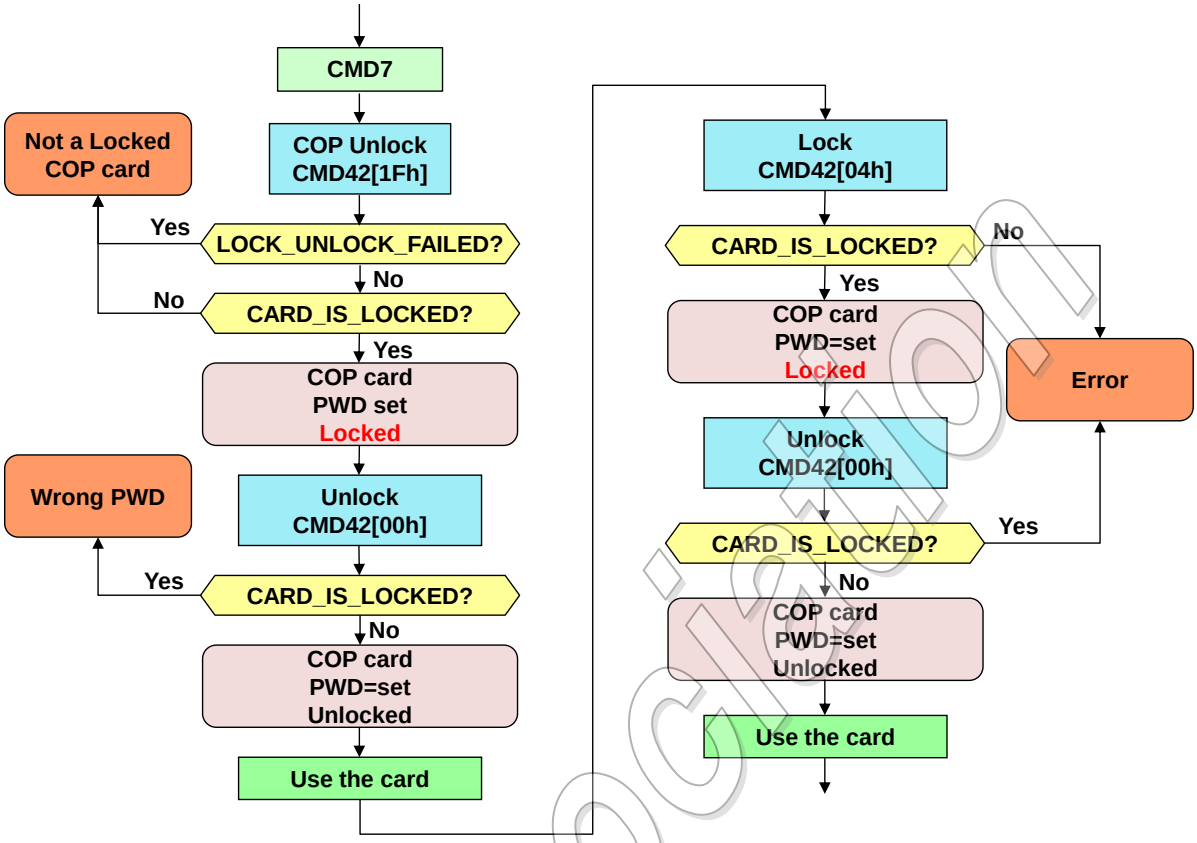


Figure 4.3.7-4 : COP Card Unlock/Lock Sequence

4.3.7.2 Parameter and the Result of CMD42

The block length shall be greater than or equal to the required data structure of CMD42; otherwise, the result of CMD42 is undefined and the card may be in the unexpected locked state. Table 4-7 clarifies the behavior of CMD42 for Non-COP Cards. Table 4.3.7-1 clarifies the behavior of PWD commands for COP Card. Table 4.3.7-2 clarifies the behavior of COP specific functions for COP Card. The reserved bits in the parameter (bit7-5 for COP Card and bit 7-4 for Non-COP Card) of CMD42 shall be don't care. In the case that CMD42 requires the password, it is assumed that the old password and the new password are set correctly; otherwise the card indicates an error regardless of Table 4-7, Table 4.3.7-1 and Table 4.3.7-2. If the password length is 0 or greater than 128 bits, the card indicates an error. If errors occur during execution of CMD42, the LOCK_UNLOCK_FAILED (Bit24 of Card Status) shall be set to 1 regardless of Table 4-7, Table 4.3.7-1 and Table 4.3.7-2. The CARD_IS_LOCKED (Bit25 of Card Status) in the response of CMD42 shall be the same as Current Card State in Table 4-7, Table 4.3.7-1 and Table 4.3.7-2. In the field of Card Status, 0 to 1 means the card changes to Locked and 1 to 0 means the card changes to Unlocked after execution of CMD42. It can be seen in the response of CMD13 after the CMD42. The LOCK_UNLOCK_FAILED (Bit24 of Card Status) as the result of CMD42 can be seen in the response of either CMD42 or the following CMD13.

CMD42 Parameter in the data
Bit3: ERASE
Bit2: LOCK_UNLOCK
Bit1: CLR_PWD
Bit0: SET_PWD

Related bits in the Card Status
Bit25: CARD_IS_LOCKED
Bit24: LOCK_UNLOCK_FAILED

CMD42 Parameter				Current Card State	PWD_LEN and PWD	Result of the Function	Card Status	
Bit3	Bit2	Bit1	Bit0				Bit25	Bit24
After Power On					Exist	The card is locked	1	0
					Cleared	The card is unlocked	0	0
1	0	0	0	Locked	Exist	Force Erase	Table 4-8	
1	0	0	0	Unlocked	Exist	Error	0	1
1	0	0	0	Unlocked	Cleared	Error	0	1
0	1	0	0	Locked	Exist	Error	1	1
0	1	0	0	Unlocked	Exist	Lock the card	0 to 1	0
0	1	0	0	Unlocked	Cleared	Error	0	1
0	1	0	1	Locked	Exist	Replace password and the card is still locked	1	0
0	1	0	1	Unlocked	Exist	Replace password and the card is locked	0 to 1	0
0	1	0	1	Unlocked	Cleared	Set Password and lock the card	0 to 1	0
0	0	1	0	Locked	Exist	Clear PWD_LEN and PWD and the card is unlocked	1 to 0	0
0	0	1	0	Unlocked	Exist	Clear PWD_LEN and PWD	0	0
0	0	1	0	Unlocked	Cleared	Error (Note *4 Refer to Table 4-10)	0	1
0	0	0	1	Locked	Exist	Replace password and the card is unlocked	1 to 0	0
0	0	0	1	Unlocked	Exist	Replace password and the card is unlocked	0	0
0	0	0	1	Unlocked	Cleared	Set password and the card is still unlocked	0	0
0	0	0	0	Locked	Exist	Unlock the card	1 to 0	0
0	0	0	0	Unlocked	Exist	Error	0	1
0	0	0	0	Unlocked	Cleared	Error	0	1
Other combinations				Don't care	Don't care	Error (Note *1 Refer to Table 4-10)	0 or 1	1

Table 4-7 : CMD42 Ver.1.0 Mode (Non-COP Card) Lock Unlock Function

CMD42 Parameter in the data
Bit4: COP
Bit3: ERASE
Bit2: LOCK_UNLOCK
Bit1: CLR_PWD
Bit0: SET_PWD

Related bits in the Card Status
Bit25: CARD_IS_LOCKED
Bit24: LOCK_UNLOCK_FAILED

CMD42 Parameter					Current Card State	PWD	Result of the Function	Card Status	
Bit4	Bit3	Bit2	Bit1	Bit0				Bit25	Bit24
After Power On or Reset						Exist	The card is locked	1	0
						Cleared	The card is unlocked	0	0
						Don't care	The card is locked (If FEP is set (4))	1	0
X(2)	1	0	0	0	Locked	Exist	Force Erase (1)	Table 4-8	
X(2)	1	0	0	0	Unlocked	Exist	Error	0	1
X(2)	1	0	0	0	Unlocked	Cleared	Error	0	1
X(2)	0	1	0	0	Locked	Exist	Error	1	1
X(2)	0	1	0	0	Unlocked	Exist	Lock the card	0 to 1	0
X(2)	0	1	0	0	Unlocked	Cleared	Error	0	1
X(2)	0	1	0	1	Locked	Exist	Replace PWD and the card is still locked	1	0
X(2)	0	1	0	1	Unlocked	Exist	Replace PWD and the card is locked	0 to 1	0
X(2)	0	1	0	1	Unlocked	Cleared	Set PWD and lock the card	0 to 1	0
X(2)	0	0	1	0	Locked	Exist	Clear PWD_LEN and PWD and the card is unlocked (1)	1 to 0	0
X(2)	0	0	1	0	Unlocked	Exist	Clear PWD_LEN and PWD (1)	0	0
X(2)	0	0	1	0	Unlocked	Cleared	Error	0	1
X(3)	0	0	0	1	Locked	Exist	Replace PWD and the card is unlocked	1 to 0	0
X(3)	0	0	0	1	Unlocked	Exist	Replace PWD and the card is unlocked	0	0
X(3)	0	0	0	1	Unlocked	Cleared	Set PWD and the card is still unlocked	0	0
X(2)	0	0	0	0	Locked	Exist	Unlock the card	1 to 0	0
X(2)	0	0	0	0	Unlocked	Exist	Error	0	1
X(2)	0	0	0	0	Unlocked	Cleared	Error	0	1

Table 4.3.7-1 : PWD Related Lock Unlock Function for COP Card

Notes:(1) – Disabled as long as FEP=set
(2) - Bit4 can be set to either 0 or 1 as long as FEP=0. Bit4 shall be set to 0 as long as FEP=set. Functions that shall be accepted with Bit4=1 are defined by Table 4.3.7-2.
(3) - Bit4 shall be set to 0 after CMD42[1Fh].
(4) – If FEP=set, all CMD42 functions are blocked before receiving CMD42[1Fh].
5 - COP Unlock (CMD42[1Fh]) enables CMD42 COP function set. No FEP is required
6 - CMD42[1Fh] followed by CMD42[11h]+FEP sequence is used to set FEP

Application Note:
To replace password, the host should consider following cases. When PWD_LEN and password data exist, the card assumes old and new passwords are set in the data structure. When PWD_LEN and PWD are cleared, the card assumes only new password is set in the data structure. In this case, the host shall not set old password in the data structure; otherwise, unexpected password is set.

CMD42 Parameter					Current Card State	PWD	FEP	Result of the Function	Card Status	
Bit4	Bit3	Bit2	Bit1	Bit0					Bit25	Bit24
1	1	0	0	0	Locked	Exist	Exist	Protected Force Erase (with proper FEP)(5)	1 to 0	0
1	1	0	0	0	Locked	Cleared	Exist	Error	1	1
1	1	0	0	0	Unlocked	Don't care	Exist	Error	0	1
1	0	0	1	0	Locked	Exist	Exist	Clear FEP and the card is still locked(5)	1	0
1	0	0	1	0	Unlocked	Don't care	Exist	Clear FEP and the card is still unlocked(5)	0	0
1	0	0	0	1	Locked	Exist	Don't care	Set/Replace FEP and card is still locked	1	0
1	0	0	0	1	Unlocked	Don't care	Don't care	Set/Replace FEP and the card is still unlocked	0	0
1	1	1	1	1	Unlocked	Don't care	Cleared	Enable Ver.2.0 function set	0	0
1	1	1	1	1	Locked	Exist	Cleared	Enable Ver.2.0 function set	1	0
1	1	1	1	1	Locked(1)	Cleared	Exist	(COP unlock)Enable other CMD42 functions card is unlocked (1)(3)	1 to 0	0
1	1	1	1	1	Locked(1)	Exist	Exist	(COP unlock)Enable other CMD42 functions card is PWD locked (1)(3)(4)	1	0
1	1	1	1	1	Locked(2)	Exist	Exist	Card is still locked (2)(4)	1	1
1	1	1	1	1	Unlocked	Don't care	Exist	Card is still unlocked	0	1

Table 4.3.7-2 : COP Specific Function for COP Card

Notes:

(1) – 1st time after Power On Reset or Reset

(2) – 2nd time or more after Power On Reset or Reset

(3) – "COP Unlock" – CMD42[1Fh] with no FEP is required to access any other CMD42 commands after power on reset or reset, when FEP=set

(4) – If PWD is set, Unlock with PWD - CMD42[00h] is required to unlock the card after "Unlock COP"

(5) – Disabled as long as FEP=0

6 – Commands not covered by Table 4.3.7-1 and Table 4.3.7-2 shall return LOCK_UNLOCK_ERROR with no change to card state.

Application Note:
To replace password, the host should consider following cases. When FEP_LEN and password data exist, the card assumes old and new passwords are set in the data structure. When FEP_LEN and FEP are cleared, the card assumes only new password is set in the data structure.

4.3.7.3 Forcing Erase

4.3.7.3.1 Force Erase

In the case that the user forgot the password (the PWD content) it is possible to erase all the card data content along with the PWD content. This operation is called *Force Erase*. Note that since the password lock is effective only to User Area, data contents stored in other than User Area are not erased by Force Erase. In addition, Force Erase also works for TCG enabled card when it is locked by the password. Note that TCG MBR Table is not erased by the Force Erase.

- Select a card (CMD7), if not previously selected already.
- Define the block length (CMD16) to 1 byte (8-bit card lock/unlock command). Send the card lock/unlock command with the appropriate data block of one byte on the data line including the 16 bit CRC. The data block shall indicate the mode ERASE (the ERASE bit shall be the only bit set).
- If the ERASE bit is not the only bit set in the data field, the LOCK_UNLOCK_FAILED error bit will be set in the status register and the erase request is rejected. If the command was accepted, then ALL THE CARD CONTENT WILL BE ERASED including the PWD and PWD_LEN register content and the locked card will be unlocked.
- An attempt to force erase on an unlocked card will fail and LOCK_UNLOCK_FAILED error bit will be set in the status register.
- [If FEP=set, execution of this function is prevented, PWD and PWD_LEN registers are not cleared, and the LOCK_UNLOCK_FAILED error bit shall be set in the status register.]

4.3.7.3.2 FEP Force Erase

In the case that FEP is set, it is possible to erase all the card data content along with the PWD content only if FEP in known. This operation is called *FEP Force Erase*.

- Unlock COP function set, if not unlocked already.
- Define the block length (CMD16), given by the 8-bit card lock/unlock mode, the 8-bit password size (in bytes), and the number of bytes of the currently used password FEP.
- Send the card lock/unlock command with the appropriate data block size on the data line including the 16-bit CRC. The data block shall indicate the mode ERASE together with COP, the length (PWDS_LEN) and the FEP itself. The rest of the bits shall be set to zero.
- If the ERASE and COP bits are not the only bits set in the data field, the LOCK_UNLOCK_FAILED error bit shall be set in the status register and the erase request is rejected.
- FEP force erase is performed only if password provided matches FEP register. If password is not provided, or incorrect, LOCK_UNLOCK_FAILED error bit shall be set in the status register and the erase request is rejected.
- An attempt to FEP force erase on an unlocked card shall fail and LOCK_UNLOCK_FAILED error bit shall be set in the status register.
- If the command was accepted, then ALL THE CARD CONTENT SHALL BE ERASED including the PWD and PWD_LEN register content and the locked card shall be unlocked.
- Even FEP force erase is executed, FEP and FEP_LEN register content shall remain unchanged.

Note that COP and TCG are exclusive. That is, Force Erase Password cannot be set to TCG enabled card, and TCG cannot be enabled when FEP=set.

4.3.7.3.3 Force Erase Function to the Locked Card

Table 4-8 clarifies the relation between force erase and Write Protection. The force erase does not erase the secure area. The card shall keep its locked state during the erase execution and change to the unlocked state after the erase of all user area is completed. Similarly, the card shall keep Temporary Write Protect, Write Protect Until Power Cycle and Group Write Protection during the erase execution and clear Write Protection after the erase of all user area is completed. In the case of an erase error occurs, the card can continue force erase if the data of error sectors are destroyed.

Write Protections
- PWP: Permanent Write Protect (CSD Bit13)
- TWP: Temporary Write Protect (CSD Bit12)
- WP_UPC: Write Protect Until Power Cycle (CSD Bit9)
- GWP: Group Write Protect (CMD28, CMD29, CMD30)

CMD42 Parameter				PWP	TWP WP_UPC GWP	Result of the Function	Card Status	
Bit3	Bit2	Bit1	Bit0				Bit25	Bit24
1	0	0	0	Yes	don't care	Error (Note *2 Refer to Table 4-10)	1	1
1	0	0	0	No	Yes	Execute force erase and clear Temporary Write Protect, Write Protect Until Power Cycle and Group Write Protect. (Note *3 Refer to Table 4-10)	1 to 0	0
1	0	0	0	No	No	Execute force erase.	1 to 0	0

Table 4-8 : Force Erase Function to the Locked Card (Relation to the Write Protects)

4.3.7.4 Relation Between ACMD6 and Lock/Unlock State

ACMD6 is rejected when the card is locked and bus width can be changed only when the card is unlocked. Table 4-9 shows the relation between ACMD6 and the Lock/Unlock state.

Card State	Bus Mode	Result of the Function
Unlocked	1-bit mode	ACMD6 is accepted
Locked	1-bit mode	ACMD6 is rejected and still in 1-bit mode
Unlocked	4-bit mode	ACMD6 is accepted
Locked	4-bit mode	ACMD6 is rejected and still in 4-bit mode. CMD0 change to 1-bit mode

Table 4-9 : Relation between ACMD6 and the Lock/Unlock State

Application Note:
After power on or reset (in 1-bit mode), if the card is locked, the SD mode host shall issue CMD42 in 1-bit mode. If the card is locked in 4-bit mode, the SD mode host shall issue CMD42 in 4-bit mode.

4.3.7.5 Commands Accepted for Locked Card

4.3.7.5.1 Details of Card Command Operation

Table 4.3.7-4 describes card command operations depending on target and lock status.

Target	User Area Partition[1], Protected Area		Boot Partition	RPMB Unit
Lock status	**Unlocked**	**Locked**	**Unlocked/Locked**	**Unlocked/Locked**
CMD17, 18	Accepted	Treated as an illegal command	Accepted	N/A
CMD19, 22	Accepted if supported	Treated as an illegal command	Accepted if supported	N/A
CMD23	Accepted if supported	Accepted if card supports ACMD53/54	Accepted if supported	Accepted
CMD24, 25 Erase class (5)	Accepted[2]	Treated as an illegal command	[RPMB disabled or after RPMB auth.] Accepted [Otherwise] Treated as an illegal command	N/A
Write protection class (6)	[SDSC] Accepted if supported [Otherwise] Treated as an illegal command	Treated as an illegal command	Treated as an illegal command	N/A

Note 1) User Area and TCG MBR Table
2) TCG MBR Table content modification shall be done using TCG commands following TCG Storage Architecture Core Specification

Table 4.3.7-4 : Card Command Operations Depending on Target and Lock Status

Table 4.3.7-5 describes other card command operations according to lock status.

Lock status	Unlocked	Locked
Basic class (0) Lock card class (7) CMD16 ACMD41, 42	Accepted	Accepted
CMD39	Accepted if supported	Accepted if supported
CMD48, 49, 58, 59	Accepted if supported	Accepted under designated conditions (see Section 4.3.7.5.2)
ACMD53, 54	Accepted if supported	Accepted if supported
Other commands	Accepted if supported	Treated as an illegal command

Table 4.3.7-5 : Other Card Command Operations According to Lock Status

When locked card accepts commands, it shall return response with setting CARD_IS_LOCKED.

4.3.7.5.2 Specific Rules for Extension Commands

As described in Section 4.3.7.5.1, card lock mechanism does not affect access to Boot Partitions and RPMB Unit. For these partitions and unit, associated parameters are defined in the Security and Boot

Register Set of Extension Register as described in Table 5-32.
SD card compliant to Physical Layer Specification Version 8.00 rejects extension commands (CMD48, 49, 58 and 59) when it is locked. However, parameters related to Boot or RPMB should be accessible even card is locked. Thus, the following rules depending on the support of Security and Boot Function (SBF, refer to Section 5.8.3) are introduced when locked card receives extension commands.

Case 1: Card does not support SBF
In this case, card always rejects these extension commands as defined in Physical Layer Specification Version 8.00.

Case 2: Card supports SBF
In this case, card operation varies according to FNO in the extension commands.

Case 2-a) FNO in the extension command corresponds to SBF
Card accepts the command and works as specified by it even card is locked (see Figure 4.3.7-5). Note that R1 response shall indicate CAR_IS_LOCKED in this case.

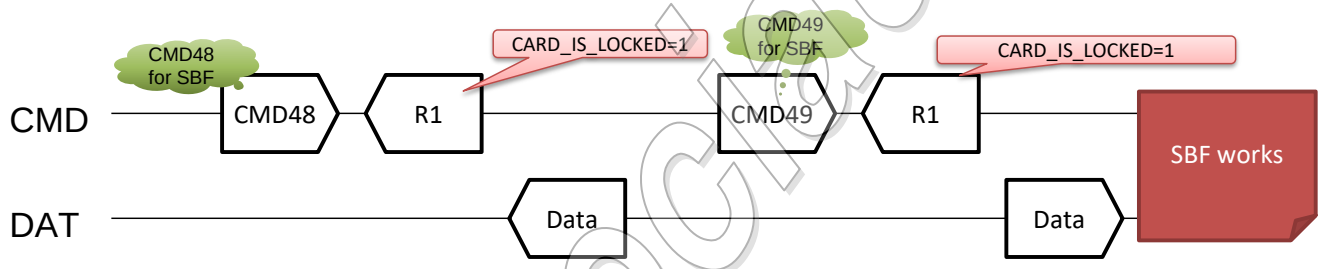


Figure 4.3.7-5 : Locked Card Operation When Supporting SBF (Command Relating to SBF)

Case 2-b) FNO in the extension command does not correspond to SBF
Card accepts the command, but the extension functions do not work. In this case, card does not transmit when it receives CMD48 or CMD58, or keeps busy state more than 1 second after it receives data followed by CMD49 or CMD59. In this time, card indicates OUT_OF_RANGE and CARD_IS_LOCKED in the response of the next command. See Figure 4.3.7-6 for more details.

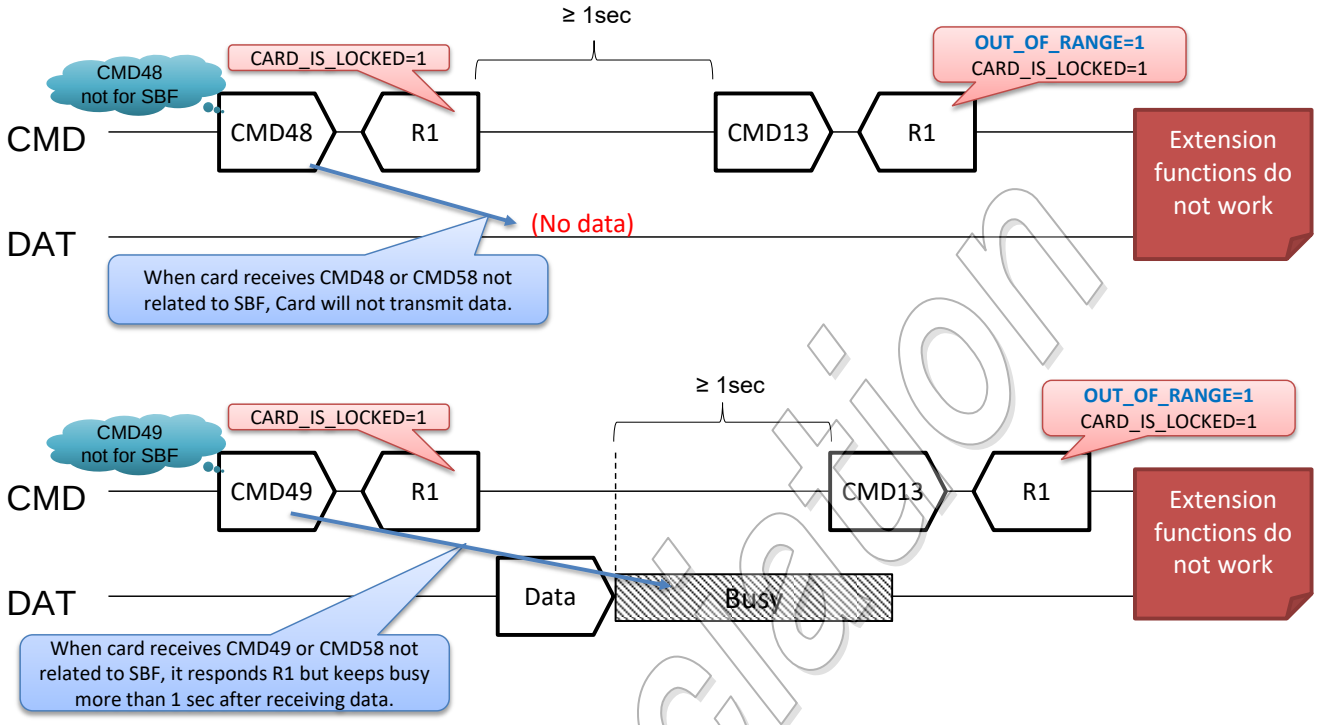


Figure 4.3.7-6 : Locked Card Operation When Supporting SBF (Command Not Relating to SBF)

4.3.7.6 Three Types of Lock/Unlock Card

There are three types of lock/unlock function-supported cards. The Type 1 is the earlier version of SD Memory Card and the Type 2 is defined in the Physical Layer Specification Version 1.10 and higher. The Type 3 (COP Card) is defined in Physical Layer Specification Version 5.10 and higher. Table 4-10 shows the difference between these types of cards. The SD memory cards that support Lock/Unlock and comply with Version 1.01, can take either Type 1 or Type 2. The SD Memory Cards that support Lock/Unlock and comply with Version 1.10 and higher, shall take Type 2. The SD Memory Cards that support Lock/Unlock and comply with Version 5.10 and higher, shall take Type 2 or Type 3.

Notes	Type 1 Card (Earlier Version)	Type 2,3 Card (New Version)
*1 in Table 4-7, Table 4.3.7-1 and Table 4.3.7-2	Treat CMD42 Parameter=0011b as 0001b. Treat CMD42 Parameter=0111b as 0101b. Treat CMD42 Parameter=0110b as 0010b. Results of other combinations are Error.	All results are Error
*2 in Table 4-8	Execute force erase and set Permanent Write Protect. If force erase is completed, the CARD_IS_LOCKED is changed from 1 to 0. A priority is given to force erase from Permanent Write Protect.	The result is Error A priority is given to Permanent Write Protect from force erase.
*3 in Table 4-8	Execute force erase but Temporary Write Protect and Group Write Protect are not cleared. It should be cleared by the host.	Execute force erase and clear Temporary Write Protect, Write Protect Until Power Cycle(1) and Group Write Protect.
*4 in Table 4-7, Table 4.3.7-1 and Table 4.3.7-2	CMD42 Parameter=0010 and CMD42 Parameter=0110 The result is no error. Card status Bit24 will be 0	The result is Error. Card status Bit24 will be 1

Note (1): Write Protect Until Power Cycle is applicable for Physical Layer Specification Version 9.00 or later.

Table 4-10 : Version Difference of Lock/Unlock Functions

Function	Non-COP Card	COP Card
COP Unlock	The result is Error. Card status Bit24 is 1	CMD42 ver.2.0 mode is enabled according to Table 4.3.7-2
Set FEP	PWD is set	FEP is set
Clear FEP	PWD_LEN and PWD is cleared	FEP_LEN and FEP is cleared
PWD clear if FEP=set	PWD_LEN and PWD is cleared	The result is Error. Card status Bit24 is 1
Force Erase if FEP=set	PWD_LEN and PWD is cleared. Card content is erased	The result is Error. Card status Bit24 is 1

Table 4.3.7-3 : Type 3 Lock/Unlock Functions Difference

Application Note:
The host can use both types of cards without checking the difference by taking account of the following points.

(1) The host should not set the parameters of CMD42 that return an error listed in Table 4.3.7-1 and Table 4.3.7-2. (For *1)
(2) The host should not issue a force erase command if the Permanent Write Protect is set to 1, otherwise the Type 1 card can no longer be used even if the user remembers the password. (For *2)
(3) After the force erase, if the Temporary Write Protect is not cleared, the host should clear it. (For *3)
(4) The host should not set FEP on COP Card locked with unknown PWD.

4.3.8 Content Protection

This section is a blank in the Simplified Specification.

4.3.9 Application-Specific Commands

4.3.9.1 Application-Specific Command – APP_CMD (CMD55)

This command, when received by the card, causes the card to interpret the following command as an application-specific command, ACMD. The ACMD provides command extension, has the same structure as that of regular commands and it may have the same CMD number. The card recognizes it as ACMD by the fact that it appears after APP_CMD.
When an ACMD is not defined, the card treats it as regular command. If, as an example, a card has a definition for ACMD13 but not for ACMD7, then, command 13 after APP_CMD is interpreted as the non-regular CMD13 but command 7 after APP_CMD is interpreted as the regular CMD7. In order to use one of the ACMD's, the host should be:

(1) When sending APP_CMD, the response has the APP_CMD bit set signaling to the host that ACMD is now expected.
(2) ACMD55 does not exist. If multiple CMD55 are issued continuously, APP_CMD bit in each response is set to 1. The command issued immediately after the last CMD55 shall be interpreted as ACMD. When more than one command (except CMD55) is issued directly after CMD55, the first command is interpreted as ACMD and the following commands are interpreted as regular commands
(3) If a defined ACMD is send and it is legal, the response has the APP_CMD bit set, indicating that the accepted command is interpreted as ACMD.
(4) If an undefined ACMD is sent and it is legal, the response has the APP_CMD bit cleared, indicating that the accepted command is interpreted as normal CMD.
(5) If a defined or undefined ACMD is sent and it is illegal, then it is handled as an illegal command. Illegal Command Error is indicated in the next R1/R6 response and host should ignore APP_CMD status in the response. Next command is handled as normal command.

Host shall not use undefined ACMDs as regular commands even if the specification defines it.
The following ACMD numbers are reserved for the SD Memory Card proprietary applications and shall not be used by any SD Memory Card manufacturer:
ACMD6, ACMD13, ACMD17-26, ACMD38-49, ACMD51.

In above explanation, commands defined in the detailed command description section are "defined" commands but not defined in the section are "Undefined" commands (Section 4.7.4 or Section 7.3.1.3 depends on bus mode). "Legal" means that a defined or undefined command is accepted at the current state and "Illegal" means that a defined or undefined command is not accepted at the current state.

Exception in ACMD41

- The response of ACMD41 does not have APP_CMD status. Sending the response of CMD41 in idle state means the card is accepted as legal ACMD41.
- As APP_CMD status is defined as "clear by read", APP_CMD status, which is set by ACMD41, may be indicated in the response of next CMD11 or CMD3. However, as ACMD11 and ACMD3 are not defined, it is not necessary to set APP_CMD status.
- Host should ignore APP_CMD status in the response of CMD11 and CMD3.

4.3.9.2 General Command - GEN_CMD (CMD56)

This section is a blank in the Simplified Specification.

4.3.10 Switch Function Command

4.3.10.1 General

Switch function command (CMD6) is used to switch or expand memory card functions. Currently four function groups are defined:

(1) Access Mode:
Selection of SD bus interface speed modes.
(2) Command System:
A specific function can be extended and controlled by a set of shared commands.
(3) Driver Strength
Selection of suitable output driver strength in UHS-I modes depends on host environment.
(4) Power Limit
Selection to limit the maximum power depends on host power supply capability and heat release capability. Current Limit is re-defined as Power Limit to account for the two power supply voltages on UHS II cards.

This was introduced in the Physical Layer Specification Version 1.10. Therefore, cards that are compatible with earlier versions of the spec do not support it. The host should check the "SD_SPEC" field in the SCR register to identify what version of the spec the card complies with before using CMD6. It is also possible to check support of CMD6 by bit10 of CCC in CSD. It is mandatory for an SD memory card of Version 1.10 and higher to support CMD6.

CMD6 is valid under the "Transfer State". Once selected, via the switch command, all functions only return to the default function after a power cycle, CMD6 (Mode 1 operation with Function 0 in each function group) or CMD0. Executing a power cycle or issuing CMD0 will cause the card to reset to the "idle" state and all the functions to switch back to the default function.

As a response to CMD6, the SD Memory Card will send R1 response on the CMD line and 512 bits of status on the DAT lines. From the SD bus transaction point of view, this is a standard single block read transaction and the time out value of this command is 100 ms, the same as in read command. If CRC error occurs on the status data, the host should issue a power cycle.
CMD6 function switching period is within 8 clocks after the end bit of status data. When CMD6 changes the bus behavior (i.e. access mode), the host is allowed to use the new functions (increase/decrease CLK frequency beyond the current max CLK frequency), at least 8 clocks after at the end of the switch command transaction (see Figure 4-14).

In response to CMD0, the switching period is within 8 clocks after the end bit of CMD0. When CMD6 has changed the bus behavior (i.e. access mode) the host is allowed to start the initialization process, at least 8 clocks after at the CMD0.

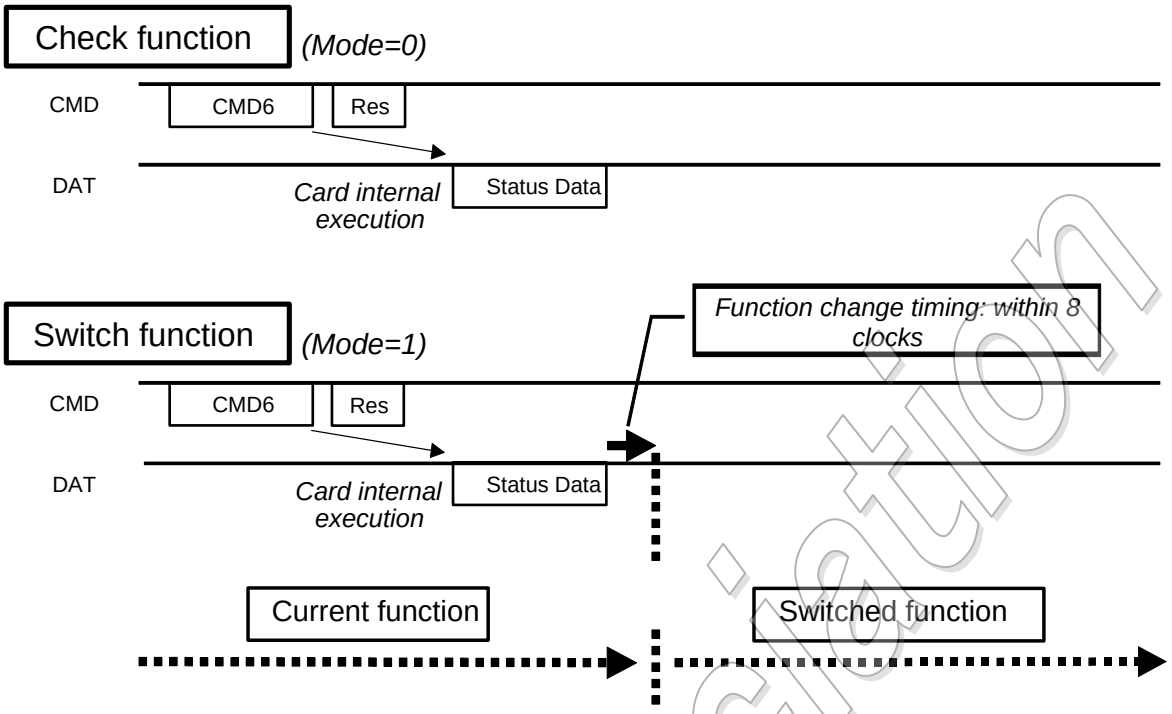


Figure 4-14 : Use of Switch Command

CMD6 supports six function groups, and each function group supports sixteen branches (functions). Only one function can be chosen and active in a given function group. Function 0 in each function group is the default function (compatible with Spec. 1.01).

CMD6 can be used in two modes:

- Mode 0 (Check function) is used to query if the card supports a specific function or functions.
- Mode 1 (set function) is used to switch the functionality of the card.

4.3.10.2 Mode 0 Operation - Check Function

CMD6 mode 0 is used to query which functions the card supports, and to identify the maximum current/power consumption of the card under the selected functions.
Refer to Table 4-32: Switch function commands (class 10) for the argument definition of CMD6.

A query is done by setting the argument field of the command, as follows:

- Set the Mode bit to 0
- Select only one function in each function group. Selection of default function is done by setting the function to 0x0. Select a specific function by using appropriate values from Table 4-11. Selecting 0xF will keep the current function that has been selected for the function group.
- When the function in query is ready, the card returns the inquired function number, if busy, the card returns the current function number (See Table 4-15).

In response to a query, the switch function will return the following 3 statuses (see Table 4-13):

- The functions that are supported by each of the function groups
- The function that the card will switch to in each of the function groups. This value is identical to the provided argument if the host made a valid selection or 0xF if the selected function was invalid.
- Maximum current/power consumption under the selected functions. If one of the selected functions was wrong, the return value will be 0.

4.3.10.3 Mode 1 Operation - Set Function

CMD6 mode 1 is used to switch the functionality of the card.

Switching to a new functionality is done by:

- Setting the Mode bit to 1
- Selecting only one function in each function group. Selection of default function is done by setting the function to 0x0. It is recommended to specify 0xF (no influence) for all selected functions, except for functions that need to be changed. Selecting 0xF will keep the current function for the function group.
- When a function cannot be switched because it is busy, the card returns the current function number (not returns 0xF), the other functions in the other groups may still be switched.

In response to a set function, the switch function will return the following 3 statuses:

- The functions that are supported by each of the function groups
- The function that is the result of the switch command. In case of invalid selection of one function or more, all set values are ignored and no change will be done (identical to the case where the host selects 0xF for all functions groups). The response to an invalid selection of function will be 0xF.
- Maximum current/power consumption under the selected functions. If one of the selected functions was wrong, the return value will be 0.

Arg. Slice	[23:20]	[19:16]	[15:12]	[11:8]	[7:4]	[3:0]
Group No.	**6**	**5**	**4**	**3**	**2**	**1**
Function name	**reserved**	**reserved**	**Power Limit*3**	**Driver Strength**	**Command system**	**Access mode*1**
0x0	Default*2		Default*2 0.72W	Default*2 Type B	Default*2	Default*2 / SDR12
0x1	Reserved	Reserved	1.44W	Type A	For eC	High-Speed / SDR25
0x2	Reserved	Reserved	2.16W	Type C	Reserved	SDR50
0x3	Reserved	Reserved	2.88W	Type D	OTP	SDR104
0x4	Reserved	Reserved	1.80W	Reserved	ASSD	DDR50
0x5	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
0x6	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
0x7	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
0x8	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
0x9	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
0xA	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
0xB	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
0xC	Reserved	Reserved	Reserved	Reserved	(eSD)	Reserved
0xD	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
0xE	Reserved	Reserved	Reserved	Reserved	Vendor specific	Reserved
0xF	No influence					

Note1: Bus Speed Mode is alias of Access Mode.
Note2: "Default" of Function 0 means that a function of just after the card initialized.
Note3: Notes for Power Limit
(1) Function 2 and 3 may be used only for embedded devices.
(2) Allowed power consumption for removable cards shall be up to 1.80W, even though the Power Limit is set to 2.16W or 2.88W.

Table 4-11 : Available Functions of CMD6

Function Group 1 is defined as Bus Speed Mode switch. If the card is initialized in 3.3V signal level, Default Speed and High Speed are assigned to function 0 and 1. Then support bits of function 2 to 4 (SDR50, SDR104 and DDR50) are set to 0. If the card is initialized in 1.8V signal level, SDR and DDR modes are assigned from function 0 to function 4.

Function Group 2 is defined for Command System extension. CMD34-37, CMD50 and CMD57 are reserved for SD command system. OTP and ASSD are added. Refer to Part A1 Advanced Security Extension (McEX), Part 1 OTP Addendum and Part A3 ASSD Core Specification for more detail.

Function Group 3 is defined as driver strength selection for UHS-I modes. This switch is effective in 1.8V signaling mode. Refer to Section 6.7.1 for more detail.

Function Group 4 is defined as Power Limit switch for total card power consumable. This field is set according to host power supply capability.

For SDIO Combo Cards, refer to SDIO specification to set the total card power consumable.

The Power Limit is defined in accordance with the Mechanical Addenda that define thermal requirements as a function of total card power consumption.

This Function Group is used for any bus mode including UHS-I and UHS-II modes. The default value of the Power Limit is 0.72W (minimum setting). In Legacy SD and UHS-I cards that have only VDD1 and not VDD2, this translates to 200mA at 3.6V.
In UHS-I mode, after selecting one of SDR50, SDR104 or DDR50 mode by Function Group 1, host needs to change the Power Limit to enable the card to operate in higher performance. In UHS-II mode, the host needs to change the Power Limit after finishing initialization to enable the card to operate in higher performance. Total card power is the sum of VDD1 and VDD2 power consumption.

In the other cases which may consume power by some function operations, such functions should have a function enable method (it will be defined in each Function Specification). After setting Power Limit, functions should be enabled by the function initialization. In case a function is not provided with enough power, it will fail initialization and the function is disabled by the card. If a function does not have a function enable method, the function shall work in 0.72W.
Power restriction described in Table 3-6, Table 3-9 and Section 6.6.3 are the case of all functions are disabled.

Function 0-F	F0	F1	F2	F3	F4	F5-FE	FF
Non UHS Mode	1	X	X	X	X	0 0	1
UHS Mode	1	1	1	1	1	0 0	1

Note 1: Same values are assigned to all X's: either 0 or 1.
Note 2: For Card compliant to Ver3.0x, all X are equal to 0 in Non UHS Mode and the support bit for F4 is set to 0 in UHS Mode.

For example, in UHS-I card, when the maximum power of a card is 1.08W (300mA at 3.6V on VDD1), the card operates at up to 0.72W (200mA at 3.6V on VDD1) when the Current Limit is set to 0.72W and up to 1.08W when Power Limit is set to 1.44W, 1.80W (and when Function 3 is selected).

Function Group 4 Power Limit	Max. Allowed Power (VDD1 and VDD2)	Max. Current of VDD1	Max. Current of VDD2
Function 0	0.72W	200mA	200mA
Function 1	1.44W	400mA	200mA
Function 2	2.16W	600mA	Note 6
Function 3	2.88W	800mA	Note 6
Function 4	1.80W	400mA	200mA

Note 1: Max. Allowed Power is defined as total of VDD1 and VDD2 power per card at the maximum voltage: 3.6V for VDD1 and 1.95V for VDD2.
Note 2: UHS-I Card is up to 1.44W even if one of Function 2, 3 or 4 is selected. 1.80W is defined for UHS-II mode.
Note 3: Function 2 and 3 cannot be used in UHS-II mode.
Note 4: UHS Host should support power supply capability of 400mA for VDD1 and 200mA for VDD2. VDD1 only Host should set to Function 0 or 1.
Note 5: Host needs to support VDD2 to use UHS-II or NFC supported cards.
Note 6: Max. Current of VDD2 for embedded device will be provided by device vendor.

Table 4-12 : Power Limit and Current Limit of VDD1 and VDD2

Application Notes:
Default setting is 0.72W. In this mode, UHS-I and UHS-II card may not provide the maximum performance. The Speed Grade performance is defined at 1.44W mode. The maximum performance of the card is available when setting of this field covers the maximum current (power) of the card that can be read by CMD6 mode 0.
Host determines setting of this field by the host power supply capability to the card, heat release capability of host and the maximum current of a connector. Refer to Mechanical Addenda for more details.

The followings are requirements in UHS-II mode,

(1) Function Group 1 and 3 are not used.
(2) Support of Function Group 2 is optional.
(3) Support of Function Group 4 is mandatory.

4.3.10.4 Switch Function Status

The switch function status is the returned data block that contains function and current consumption information. The block length is predefined to 512 bits and the use of SET_BLK_LEN command is not necessary. Table 4-13 describes the status data structure.

The status bits of the response contain the information of the function group. Maximum current consumption will be used only for the new function added through this command. In this case, VDD_R_CURR_MIN, VDD_W_CURR_MIN, VDD_R_CURR_MAX and VDD_W_CURR_MAX values in the CSD register provides the current consumption when all card functions are set to the default state and can be used by spec 1.01 compatible hosts.

<table>
<tr><th>Bits</th><th>Description</th><th>Width</th></tr>
<tr><td>511:496</td><td>Maximum Current/Power Consumption
This field indicates total current/power consumption of the card including enabled functions selected by the Function Selection ([399:376] bits) dependent on selected Bus Speed Mode.
The host should check the maximum current consumption by mode 0 operation and verify that it can supply the necessary current/power before executing mode 1 operation.
Especially in UHS-I or UHS-II mode, this field is related to Current Limit / Power Limit setting. CMD6 mode 0 indicates the maximum current/power of a selected bus speed mode regardless of the setting of Current Limit / Power Limit. CMD6 mode 1 indicates the maximum current/power of a selected bus speed mode depending on the setting of Current Limit.

Definition for SD I/F Mode
<table>
<tr><th>Value</th><th>Maximum Current(Power) Consumption at 3.6V</th></tr>
<tr><td>0</td><td>Error</td></tr>
<tr><td>1</td><td>1mA (3.6mW)</td></tr>
<tr><td>2</td><td>2mA (7.2mW)</td></tr>
<tr><td>3</td><td>3mA (10.8mW)</td></tr>
<tr><td>.....</td><td>.....</td></tr>
<tr><td>65,535</td><td>65,535mA (235926mW)</td></tr>
</table>
The voltage to calculate current consumption is defined at 3.6V.
Maximum current consumption is average over 1second.

Definition for UHS-II Mode
For UHS-II Card, this field is defined as Maximum Power Consumption. Total maximum power consumption of V_{DD1} and V_{DD2} is indicated. The definition of Power Consumption in this field is equivalent to Current Consumption at 3.6V. Refer to Current Limit / Power Limit in Section 4.3.10.3 about the relation between power and current of VDD1 and VDD2.
<table>
<tr><th>Value</th><th>Maximum Power Consumption</th></tr>
<tr><td>0</td><td>Error</td></tr>
<tr><td>1</td><td>3.6mW (= 1mA x 3.6V)</td></tr>
<tr><td>2</td><td>7.2mW (= 2mA x 3.6V)</td></tr>
<tr><td>3</td><td>10.8mW (= 3mA x 3.6V)</td></tr>
<tr><td>.....</td><td>.....</td></tr>
<tr><td>65,535</td><td>235926mW</td></tr>
</table>
</td><td>16</td></tr>
</table>

Bits	Description	Width
495:480	**Support Bits of Functions in Function Group 6** Each bit corresponds to a function. Function 15 and 0 are always enabled. If a bit 480+i is set, function i is supported (i = Function 15 to 0)	16
479:464	**Support Bits of Functions in Function Group 5** Each bit corresponds to a function. Function 15 and 0 are always enabled. If a bit 464+i is set, function i is supported (i = Function 15 to 0)	16
463:448	**Support Bits of Functions in Function Group 4** Each bit corresponds to a function. Function 15 and 0 are always enabled. If a bit 448+i is set, function i is supported (i = Function 15 to 0)	16
447:432	**Support Bits of Functions in Function Group 3** Each bit corresponds to a function. Function 15 and 0 are always enabled. If a bit 432+i is set, function i is supported (i = Function 15 to 0)	16
431:416	**Support Bits of Functions in Function Group 2** Each bit corresponds to a function. Function 15 and 0 are always enabled. If a bit 416+i is set, function i is supported (i = Function 15 to 0)	16
415:400	**Support Bits of Functions in Function Group 1** Each bit corresponds to a function. Function 15 and 0 are always enabled. If a bit 400+i is set, function i is supported (i = Function 15 to 0)	16
399:396	**Function Selection of Function Group 6** mode 0 - The function which can be switched in function group 6. mode 1 - The function which is result of the switch command, in function group 6. 0xF shows function set error with the argument.	4
395:392	**Function Selection of Function Group 5** mode 0 - The function which can be switched in function group 5. mode 1 - The function which is result of the switch command, in function group 5. 0xF shows function set error with the argument.	4
391:388	**Function Selection of Function Group 4** mode 0 - The function which can be switched in function group 4. mode 1 - The function which is result of the switch command, in function group 4. 0xF shows function set error with the argument.	4
387:384	**Function Selection of Function Group 3** mode 0 - The function which can be switched in function group 3. mode 1 - The function which is result of the switch command, in function group 3. 0xF shows function set error with the argument.	4
383:380	**Function Selection of Function Group 2** mode 0 - The function which can be switched in function group 2. mode 1 - The function which is result of the switch command, in function group 2. 0xF shows function set error with the argument.	4
379:376	**Function Selection of Function Group 1** mode 0 - The function which can be switched in function group 1. mode 1 - The function which is result of the switch command, in function group 1. 0xF shows function set error with the argument.	4
375:368	**Data Structure Version** 00h – bits 511:376 are defined 01h – bits 511:272 are defined 02h-FFh – reserved	8
367:352	**Reserved for Busy Status of functions in group 6** If bit [i] is set, function [i] is busy. This field can be read in mode 0 and mode 1	16
351:336	**Reserved for Busy Status of functions in group 5** If bit [i] is set, function [i] is busy. This field can be read in mode 0 and mode 1	16

Bits	Description	Width
335:320	**Reserved for Busy Status of functions in group 4** If bit [i] is set, function [i] is busy. This field can be read in mode 0 and mode 1	16
319:304	**Reserved for Busy Status of functions in group 3** If bit [i] is set, function [i] is busy. This field can be read in mode 0 and mode 1	16
303:288	**Busy Status of functions in group 2** If bit [i] is set, function [i] is busy. This field can be read in mode 0 and mode 1	16
287:272	**Reserved for Busy Status of functions in group 1** If bit [i] is set, function [i] is busy. This field can be read in mode 0 and mode 1	16
271:0	Reserved (All '0')	272

Table 4-13 : Status Data Structure

4.3.10.4.1 Busy Status Indication for Functions

Each bit [367-272] shows the busy status of corresponding functions; 0 indicates ready and 1 indicates busy. While the status is busy, the host should not change the corresponding function. Switch command mode 1 can be applied only to ready functions.

If the function failed to be switched in mode 1 operation and returns the current function number in the response, the function is considered busy. The mode 1 operation may affect the behavior of a function. The mode 0 operations should be used to check the busy status of a function because it does not affect its behavior, especially, for function group 2 as defined below.

Function Group 2															
303	302	301	300	299	298	297	296	295	294	293	292	291	290	289	288
0	VS	0	0	0	0	0	0	0	0	0	ASSD	0	0	eC	0

Note: 0: Ready 1: Busy

Figure 4-15 : Busy Status of 'Command System'

4.3.10.4.2 Data Structure Version

Data Structure Version indicates effective bit fields of the Switch Function Status. The cards can set either 00h or 01h. When this field is set to 01, busy status indication is effective.

Data Structure Version	Fields of Status Data Structure
00h	511:376 are defined
01h	511:272 are defined
02h-FFh	Reserved

Table 4-14 : Data Structure Version

4.3.10.4.3 Function Table of Switch Command

Table 4-15, Table 4-16 and Table 4-17 shows possible combinations of the function switch.

"Argument" indicates 4-bit code specified in the argument of switch command (bits 23-0). "**Busy Status**" indicates the function is busy as defined below.

"Status Code" indicates 4-bit code in the Status Data Structure, bits 399-376.

Argument	Busy Status	Status Code	Comment
0	Don't Care	0	Status indicates a default function, which is always supported.
Supported function	Ready	=Arg.	Status indicates that the function specified in the argument is supported and can be switched.
	Busy	Current Selected	Status indicates that the function specified in the argument is supported but cannot be switched because the function is busy.
Not Supported function	Don't Care	Fh	Status indicates that the function specified in the argument is not supported.
Fh	Don't Care	Current Selected	Status indicates current selected function

Table 4-15 : Status Code of Mode 0 to Supported Function Group

Argument	Busy Status	Status Code	Comment
0	Don't Care	0	Default function can always be switched.
Supported function	Ready	=Arg.	Status indicates the same function number as specified in the argument, which means successful function change.
	Busy	Current Selected	Switch function is canceled and status indicates current selected function.
Not Supported function	Don't Care	Fh	If one of the function groups indicates an error code (Fh), switch requests to all switch functions are canceled and the rest of the data in the Status Data Structure should be ignored.
Fh	Don't Care	Current Selected	Status indicates current selected function

Table 4-16 : Status Code of Mode 1 to Supported Function Group

Argument	Busy Status	Status Code	Comment
0	Don't Care	0	Status always indicates 0.
Eh-1h	Don't Care	Fh	Status always indicates Fh.
Fh	Don't Care	0	Status always indicates 0.

Table 4-17 : Status Code of Mode 0 and 1 to Unsupported Function Group

4.3.10.5 Relationship between CMD6 Data and Other Commands

This section is a blank in the Simplified Specification.

4.3.10.6 Switch Function Flow Example

This section is a blank in the Simplified Specification.

4.3.10.7 Example of Checking

This section is a blank in the Simplified Specification.

4.3.10.8 Example of Switching

This section is a blank in the Simplified Specification.

4.3.11 High-Speed Mode (25 MB/sec interface speed)

Although the Rev 1.01 SD memory card supports up to 12.5 MB/sec interface speed, the speed of 25 MB/sec is necessary to support increasing performance needs of the host and because memory size continues to grow.

To achieve the 25 MB/sec interface speed, the clock rate is increased to 50 MHz and CLK/CMD/DAT signal timing and circuit conditions are reconsidered and changed from the Physical Layer Specification Version 1.01.

After power up, the SD memory card is in the default speed mode, and by using Switch Function command (CMD6), the Version 1.10 and higher SD memory card can be placed in High-Speed mode. The High-Speed function is a function in the access mode group (see Table 4-11). Supporting High-Speed mode is optional.

Because it is not possible to control two cards or more in the case that each of them has a different timing mode (Default and High-Speed mode) and in order to satisfy severe timing, the host shall drive only one card. CLK/CMD/DAT signal shall be connected in 1-to-1 between the host and the card.

4.3.12 Command System

SD commands CMD34-37, CMD50, and CMD57 are reserved for SD command system expansion via the switch command. Switching between the various functions of the command system function group, will change the interpretation and associated bus transaction (i.e. command without data transfer, single block read, multiple block write, etc.) of these commands. Supporting Command system is optional

- When the "standard command set" (default function 0x0) is selected, these commands will not be recognized by the card and will be considered as illegal commands (as defined in the Physical Layer Specification Version 1.01)
- When the "vendor specific" (function 0xE) is selected, the behaviors of these commands are vendor specific. They are not defined by this standard and may change for different card vendors.
- When the "mobile e-commerce" (function 0x1) is selected, the behavior of these commands is governed by the SD Specifications Part A1: Mobile Commerce Extension Specification.

When either of these extensions is used, special care should be given to proper selection of the command set function; otherwise, the host command may be interpreted incorrectly.

All other commands of the SD memory card (not reserved for the switch commands) are always available and will be executed as defined in this document regardless of the currently selected commands set.

4.3.13 Send Interface Condition Command (CMD8)

CMD8 (Send Interface Condition Command) is defined to initialize SD Memory Cards compliant to the Physical Layer Specification Version 2.00 or later. CMD8 is valid when the card is in Idle state. This command has two functions.

- Voltage check:
  Checks whether the card can operate on the host supply voltage.
- Enabling expansion of existing command and response:
  Reviving CMD8 enables to expand new functionality to some existing commands by redefining previously reserved bits. ACMD41 was expanded to support initialization of SDHC Card and the expansion is also applied to SDXC and SDUC Card.

Table 4-18 shows the format of CMD8. As described in Low Voltage Interface Addendum, VHS=0010b is set to the CMD8 after LVS Identification, including CA-mode Fast Boot.

Bit position	47	46	[45:40]	[39:22]	21	20	[19:16]	[15:8]	[7:1]	0
Width (bits)	1	1	6	18	1	1	4	8	7	1
Value	'0'	'1'	'001000'	'00000h'	x	x	x	x	x	'1'
Description	Start bit	Transmission bit	Command index	Reserved bits	PCIe 1.2V Support [1]	PCIe Avail-ability [2]	Voltage supplied (VHS)	Check pattern	CRC7	End bit

1) Host asks whether card supports VDD3 (1.2V power rail)
   0b: Not asking 1.2V support
   1b: Asking 1.2V support (VDD3 is supported by host.
   VDD3 shall be used if card supports it, too.)
2) Host asks card's PCIe availability
   0b: Not asking PCIe availability
   1b: Asking PCIe availability (PCIe interface is supported by host.
   PCIe interface shall be used if card supports it, too.)

Voltage Supplied	Value Definition
0000b	Not Defined
0001b	2.7-3.6V
0010b	Reserved for Low Voltage Range
0100b	Reserved
1000b	Reserved
Others	Not Defined

Table 4-18 : Format of CMD8

When the card is in Idle state, the host shall issue CMD8 before ACMD41. In the argument, 'voltage supplied' is set to the host supply voltage and 'check pattern' is set to any 8-bit pattern.
The card checks whether it can operate on the host's supply voltage. The card that accepted the supplied voltage returns R7 response. In the response, the card echoes back both the voltage range and check pattern set in the argument. If the card does not support the host supply voltage, it shall not return response and stays in Idle state. shows the card operation for CMD8.

Refer to Low Voltage Interface Addendum for VHS Value Definition during LV initialization.

A part of this section is not described in the simplified version.

4.3.14 Command Functional Difference in Card Capacity Types

CCS in the response of ACMD41 determines card capacity types: CCS=0 is SDSC and CCS=1 is SDHC or SDXC.

Memory access commands include block read commands (CMD17, CMD18), block write commands (CMD24, CMD25), and block erase commands (CMD32, CMD33).

Following are the functional differences of memory access commands between SDSC and SDHC, SDXC, SDUC:

- Command Argument
  SDHC and SDXC use the 32-bit argument of memory access commands as block address format. Block length is fixed to 512 bytes regardless CMD16,
  SDSC uses the 32-bit argument of memory access commands as byte address format. Block length is determined by CMD16,
  i.e.:
  (a) Argument 0001h is byte address 0001h in the SDSC and 0001h block in SDHC and SDXC
  (b) Argument 0200h is byte address 0200h in the SDSC and 0200h block in SDHC and SDXC
  Refer to Section 4.20.2 for SDUC.
- Partial Access and Misalign Access
  SDHC, SDXC and SDUC disable Partial access and Misalign access (crossing physical block boundary) as the block address is used. Access is only granted based on block addressing.
- Set Block Length
  SDHC, SDXC and SDUC use 512-byte fixed block length for memory access commands regardless of the block length set by CMD16. The setting of the block length does not affect the memory access commands. CMD42 is not classified as a memory access command. The data block size shall be specified by CMD16 and the block length can be set up to 512 bytes. Setting block length larger than 512 bytes sets the BLOCK_LEN_ERROR error bit regardless of the card capacity.
- Write Protected Group
  SDHC, SDXC and SDUC do not support write-protected groups. Issuing CMD28, CMD29 and CMD30 generates the ILLEGAL_COMMAND error.

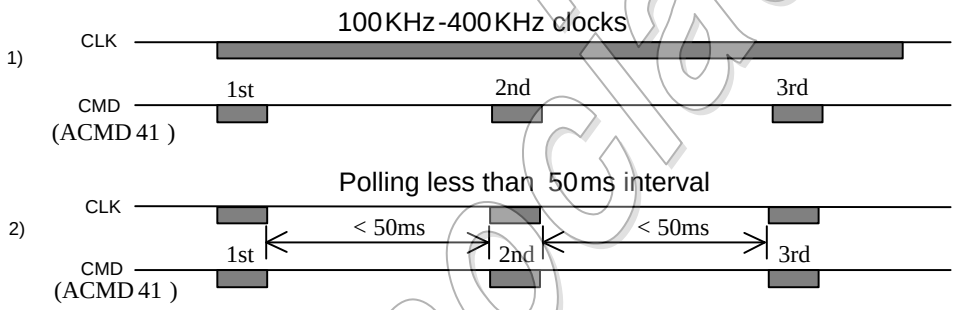
4.4 Clock Control

The SD Memory Card bus clock signal can be used by the host to change the cards to energy saving mode or to control the data flow (to avoid under-run or over-run conditions) on the bus. The host is allowed to lower the clock frequency or shut it down. For example, in the case that a host with 512 Bytes of data buffer would like to transfer data to a card with 1 KByte write blocks. So, to preserve a continuous data transfer, from the card's point of view, the clock to the card shall be stopped after the first 512 Bytes. Then the host will fill its internal buffer with another 512 Bytes. After the second half of the write block is ready in the host, it will continue the data transfer to the card by re-starting the clock supply. In such a way, the card does not recognize any interruptions in the data transfer.

There are a few restrictions the host shall consider:

- The bus frequency can be changed at any time (under the restrictions of maximum data transfer frequency and the identification frequency defined by the specification document).
- An exemption to the above is ACMD41 (SD_APP_OP_COND). After issuing the command ACMD41, the following 1) or 2) procedures shall be done by the host until the card becomes ready.
  1) Issue continuous clock in the frequency range of 100 KHz-400 KHz.

If the host wants to stop the clock, poll busy bit by ACMD41 command at less than 50 ms intervals.



- It is an obvious requirement that the clock shall be running for the card to output data or response tokens. After the last SD Memory Card bus transaction, the host is required, to provide 8 (eight) clock cycles for the card to complete the operation before shutting down the clock. Following is a list of the various bus transactions:
  - •A command with no response. 8 clocks after the host command end bit.
  - •A command with response. 8 clocks after the card response end bit.
  - •A read data transaction. 8 clocks after the end bit of the last data block.
  - •A write data transaction. 8 clocks after the CRC status token.
- The host is allowed to shut down the clock of a "busy" card. The card will complete the programming operation regardless of the host clock. However, the host shall provide a clock edge for the card to turn off its busy signal. Without a clock edge, the card (unless previously disconnected by a deselect command -CMD7) will force the DAT line down forever.

4.5 Cyclic Redundancy Code (CRC)

The CRC is intended to protect SD Memory Card commands, responses, and data transfer against transmission errors on the SD Memory Card bus. One CRC is generated for every command and checked for every response on the CMD line. For data blocks, one CRC per transferred block is generated. The CRC is generated and checked as described in the following.

- **CRC7**

The CRC7 check is used for all commands, for all responses except type R3, and for the CSD and CID registers. The CRC7 is a 7-bit value and is computed as follows:

Generator polynomial: $G(x) = x^7 + x^3 + 1$.

$M(x) = (\text{first bit}) * x^n + (\text{second bit}) * x^{n-1} + ... + (\text{last bit}) * x^0$

$CRC[6...0] = \text{Remainder } [(M(x) * x^7)/G(x)]$

The first bit is the most left bit of the corresponding bit string (of the command, response, CID or CSD). The degree *n* of the polynomial is the number of CRC protected bits decreased by one. The number of bits to be protected is 40 for commands and responses (n = 39), and 120 for the CSD and CID (n = 119).

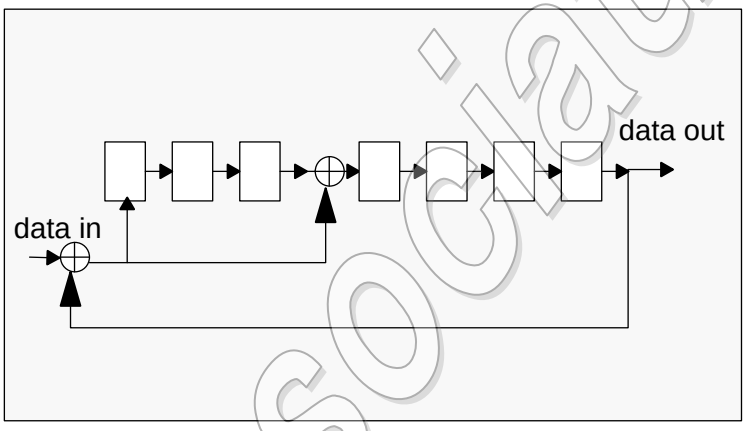


Figure 4-20 : CRC7 Generator/Checker

- **CRC7 Examples**

The CRC section of the command/response is bolded.

CMD0 (Argument=0) --> 01 000000 00000000000000000000000000000000 "**1001010**" 1
CMD17 (Argument=0) --> 01 010001 00000000000000000000000000000000 "**0101010**" 1
Response of CMD17 --> 00 010001 00000000000000000000100100000000 "**0110011**" 1

- **CRC16**

In the case of one DAT line usage, the CRC16 is used for payload protection in block transfer mode. The CRC check sum is a 16-bit value and is computed as follows:

Generator polynomial $G(x) = x^{16} + x^{12} + x^5 + 1$

$M(x) = (\text{first bit}) * x^n + (\text{second bit}) * x^{n-1} + ... + (\text{last bit}) * x^0$

$CRC[15...0] = \text{Remainder } [(M(x) * x^{16})/G(x)]$

The first bit is the first data bit of the corresponding block. The degree *n* of the polynomial denotes the number of bits of the data block decreased by one (e.g. n = 4095 for a block length of 512 bytes). The generator polynomial G(x) is a standard CCITT polynomial. The code has a minimal distance d=4 and is used for a payload length of up to 2048 Bytes (n <= 16383).

The same CRC16 method shall be used in single DAT line mode and in wide bus mode.

In wide bus mode, the CRC16 is done on each line separately.

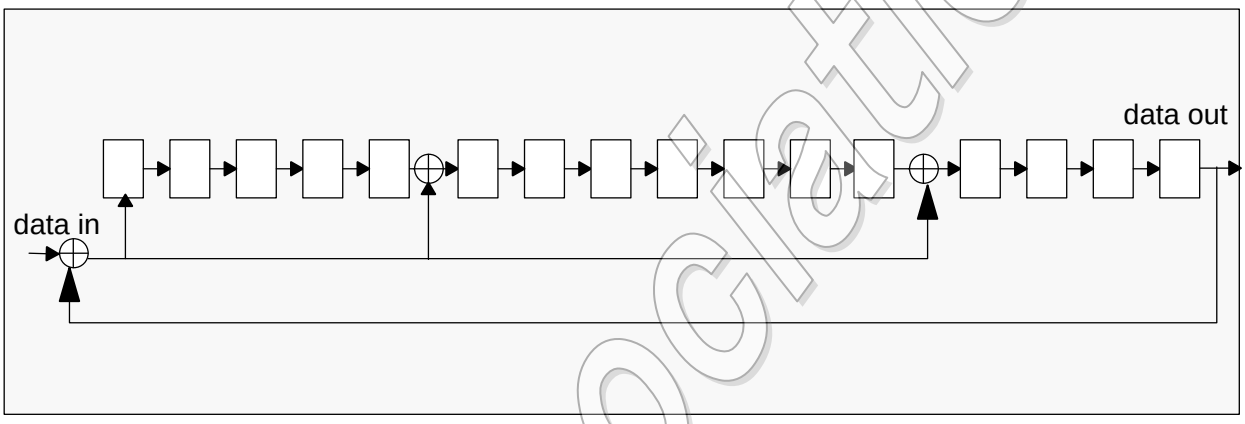


Figure 4-21 : CRC16 Generator/Checker

- **CRC16 Example**

512 bytes with 0xFF data --> CRC16 = 0x7FA1

4.6 Error Conditions

4.6.1 CRC and Illegal Command

All commands are protected by CRC (cyclic redundancy check) bits. If the addressed card's CRC check fails, the card does not respond and the command is not executed. The card does not change its state, and COM_CRC_ERROR bit is set in the status register.

Similarly, if an illegal command has been received, a card shall not change its state, shall not response and shall set the ILLEGAL_COMMAND error bit in the status register. Only the non-erroneous state branches are shown in the state diagrams (see Figure 4-1 and Figure 4-13).

Table 4-35 contains a complete state transition description.

There are different kinds of illegal commands:

- Commands that belong to classes not supported by the card (e.g. write commands in read only cards).
- Commands not allowed in the current state (e.g. CMD2 in Transfer State).
- Commands that are not defined (e.g. CMD5).

4.6.2 Read, Write and Erase Timeout Conditions

A card shall complete the command within the time period defined as follows or give up and return an error message. If the host does not get any response with the given timeout it should assume that the card is not going to respond and try to recover (e.g. reset the card, power cycle, reject, etc.).

4.6.2.1 Read

For a Standard Capacity SD Memory Card, the times after which a timeout condition for read operations occurs are (card independent) **either 100 times longer** than the typical access times for these operations given below **or 100 ms (the lower of the two)**. The read access time is defined as the sum of the two times given by the CSD parameters TAAC and NSAC (see Section 5.3). In the case of a single read operation, these card parameters define the typical delay between the end bit of the read command and the start bit of the data block. In the case of a multiple-read operation, they also define the typical delay between the end bit of a data block and the start bit of next data block.

A High Capacity SD Memory Card and Extended Capacity SD Memory Card indicate TAAC and NSAC as fixed values. The host should use 100 ms timeout (minimum) for single and multiple read operations rather than using TAAC and NSAC.

4.6.2.2 Write

For a Standard Capacity SD Memory Card, the times after which a timeout condition for write operations occurs are (card independent) **either 100 times longer** than the typical program times for these operations given below **or 250 ms (the lower of the two).** The R2W_FACTOR field in the CSD is used to calculate the typical block program time obtained by multiplying the read access time by this factor. It applies to all write commands (e.g. SET(CLR)_WRITE_PROTECT, PROGRAM_CSD and the block write commands). High Capacity SD Memory Card, Extended Capacity SD Memory Card and SDUC Memory Card indicate R2W_FACTOR as a fixed value.

In case of High Capacity SD Memory Card, maximum length of busy is defined as 250ms for all write operation.

While the card should try to maintain that busy indication of write operation does not exceed 250ms in the case of SDXC and SDUC card, if the card is not possible to maintain operations with 250ms busy, the card can indicate write busy up to 500ms including single and multiple block write in the following scenarios:

a) The last busy in any write operation up to 500ms including single and multiple block write.
b) When multiple block write is stopped by CMD12, the busy from the response of CMD12 is up to 500ms.

c) When multiple block write is stopped by CMD23, the busy after the last data block is up to 500ms.
d) Busy indication at block gap in multiple block write is up to 250ms except a following case. When the card executes consecutive two blocks write (2*512Bytes) and it spans across the physical block boundary, the busy after the each block can be indicated up to 500ms.

Especially regardless of the above definition, a speed class writing mode specified by CMD20 shall keep write busy up to 250ms in any case until the end of speed class write is indicated.

There are two types of busies in a multiple block write operation.
(1) Write busy at block gap (without CMD12) is maximum 250ms
(2) Write busy after CMD12 is maximum 250ms (500ms for SDXC and SDUC)

If CMD12 is issued during a multiple block write operation's busy period, the host timeout counter is reset and the 250ms (500ms for SDXC and SDUC) timeout period is measured from the response of CMD12.

In UHS-II mode, data is transferred by the unit of Data Burst. Data Burst consists of one or multiple data blocks and is determined depends on capability of host and card. Busy is not indicated after every block but indicated after every Data Burst. The maximum busy length after Data Burst is defined as 1 second. Refer to UHS-II Addendum about Data Burst for more details.

Application Notes:
The host should use a fixed timeout for write operations rather than using a timeout calculated from the R2W_FACTOR parameter.
It is strongly recommended for hosts to implement more than 500ms timeout value even if the card indicates the 250ms maximum busy length.
Even if the card supports Speed Class, any multiple block write operation may indicate a busy period of up to a maximum of 250ms. The sum of the busy periods over an AU is limited by Speed Class.
In UHS-II mode, refer to UHS-II Addendum about host timeout setting.

4.6.2.3 Erase

If the card supports parameters for erase timeout calculation in the SD Status, the host should use them to determine erase timeout (see Section 4.10.2). If the card does not support these parameters, erase timeout can be estimated by block write delay.
The duration of an erase command can be estimated by the number of write blocks (WRITE_BL) to be erased multiplied by 250 ms.

4.7 Commands

SD Commands applicable to UHS-II are defined in the UHS-II Addendum.

4.7.1 Command Types

There are four kinds of commands defined to control the SD Memory Card:

- Broadcast commands (bc), no response - The broadcast feature is only if all the CMD lines are connected together in the host. If they are separated, then each card will accept it separately in its turn.
- Broadcast commands with response (bcr)
  response from all cards simultaneously - Since there is no Open Drain mode in SD Memory Card, this type of command shall be used only if all the CMD lines are separated - the command will be accepted and responded by every card separately.
- Addressed (point-to-point) commands (ac)
  no data transfer on DAT
- Addressed (point-to-point) data transfer commands (adtc)
  data transfer on DAT

All commands and responses are sent over the CMD line of the SD Memory Card. The command transmission always starts with the left bit of the bit string corresponding to the command codeword.

4.7.2 Command Format

All commands have a fixed code length of 48 bits.

Bit position	47	46	[45:40]	[39:8]	[7:1]	0
Width (bits)	1	1	6	32	7	1
Value	'0'	'1'	x	x	x	'1'
Description	Start bit	Transmission bit	Command index	Argument	CRC7	End bit

Table 4-20 : Command Format

A command always starts with a start bit (always 0), followed by the bit indicating the direction of transmission (host = 1). The next 6 bits indicate the index of the command, this value being interpreted as a binary coded number (between 0 and 63). Some commands need an argument (e.g. an address), which is coded by 32 bits. A value denoted by 'x' in the table above indicates this variable is dependent on the command. All commands are protected by a CRC (see Section 4.5 for the definition of CRC7). Every command codeword is terminated by the end bit (always 1). All commands and their arguments are listed in Table 4-23 to Table 4-33.

4.7.3 Command Classes

The command set of the SD Memory Card system is divided into several classes (See Table 4-21). Each class supports a set of card functionalities.
Table 4-21 determines the setting of CCC from the card supported commands. A CCC bit, which corresponds to a supported command number, is set to 1. A class in CCC includes mandatory commands is always set to 1. Cards with specific functions may need to support some optional commands. For example, Combo Card shall support CMD5.

Class 0, 2, 4, 5 and 8 are mandatory and shall be supported by all SD Memory Cards. Class 7 except CMD40 is mandatory for SDHC, SDXC and SDUC. The other classes are optional. The supported Card Command Classes (CCC) are coded as a parameter in the card specific data (CSD) register of each card, providing the host with information on how to access the card.
If different types of commands are assigned to a Command Class (Class 7, Class 8 and Class 11), which command supported can be determined by referring command support information in SCR register.

Card Command Class (CCC)	0	1	2	3	4	5	6	7	8	9	10	11
	basic	Command Queue	block read	reserved	block write	erase	write protection	lock card	application specific	I/O mode	switch	extension
CMD0	+											
CMD2	+											
CMD3	+											
CMD4	+											
CMD5										+		
CMD6											+	
CMD7	+											
CMD8	+											
CMD9	+											
CMD10	+											
CMD11	+											
CMD12	+											
CMD13	+											
CMD15	+											
CMD16			+		+			+				
CMD17			+									
CMD18			+									
CMD19			+									
CMD20			+		+							
CMD21												+
CMD22			+		+							
CMD23			+		+				+			
CMD24					+							
CMD25					+							
CMD27					+							
CMD28							+					
CMD29							+					
CMD30							+					
CMD32						+						
CMD33						+						
CMD34-37											+	
CMD38						+						
CMD39												+
CMD40								+				
CMD42								+				
CMD43-47		+										
CMD48												+
CMD49												+

Card Command Class (CCC)	0	1	2	3	4	5	6	7	8	9	10	11
	basic	Command Queue	block read	reserved	block write	erase	write protection	lock card	application specific	I/O mode	switch	extension
CMD50											+	
CMD52										+		
CMD53										+		
CMD55									+			
CMD56									+			
CMD57											+	
CMD58												+
CMD59												+
ACMD6									+			
ACMD13									+			
ACMD14									+			
ACMD15									+			
ACMD16									+			
ACMD22									+			
ACMD23									+			
ACMD28									+			
ACMD41									+			
ACMD42									+			
ACMD51									+			
ACMD53									+			
ACMD54									+			

Table 4-21 : Card Command Classes (CCCs) in SD Mode

Commands	Support requirements
CMD0	Mandatory
CMD2	Mandatory
CMD3	Mandatory
CMD4	Mandatory
CMD5	Optional
CMD6	Mandatory for cards version 1.10 and after
CMD7	Mandatory
CMD8	Mandatory for cards version 2.00 and after
CMD9	Mandatory
CMD10	Mandatory
CMD11	Mandatory for cards supporting UHS-I. Optional for cards that do not support UHS-I.
CMD12	Mandatory
CMD13	Mandatory

Commands	Support requirements
CMD15	Mandatory
CMD16	Mandatory
CMD17	Mandatory
CMD18	Mandatory
CMD19	Mandatory for cards supporting UHS-I. Optional for cards that do not support UHS-I.
CMD20	Not supported for SDSC cards. Mandatory for SDHC, SDXC and SDUC cards that support Video Speed Class. Optional for SDHC cards that support: a) Speed Class; or b) UHS Speed Grade, and do not support Video Speed Class. Mandatory for SDXC and SDUC cards that support Speed Class or UHS Speed Grade.
CMD21	Optional
CMD22	Not supported for SDHC and SDXC cards. Mandatory for SDUC cards.
CMD23	Not supported for SDSC cards. Mandatory for SDHC, SDXC and SDUC cards that support UHS104. Optional for SDHC, SDXC and SDUC cards that do not support UHS104. Mandatory for cards supporting ACMD53 or ACMD54 regardless of the conditions above. For SDSC card supporting ACMD53 or ACMD54, CMD23 is effective only before ACMD53 and ACMD54. Otherwise, it is ignored (accepted but does not work).
CMD24	Mandatory for writable types of cards
CMD25	Mandatory for writable types of cards
CMD27	Mandatory for writable types of cards
CMD28	Optional
CMD29	Optional
CMD30	Optional
CMD32	Mandatory for writable types of cards
CMD33	Mandatory for writable types of cards
CMD34-37	Optional for cards version 1.10 and after
CMD38	Mandatory for writable types of cards Discard and FULE support is optional
CMD39	Mandatory for cards supporting boot functionalities
CMD40	Optional
CMD42	Optional for cards version 1.01 and 1.10. Mandatory for cards version 2.00 and after. COP support is optional for CMD42
CMD43-47	Mandatory for cards supporting Command Queue
CMD48	Optional Mandatory for cards supporting “Performance Enhancement Function” (refer to Section 5.8.2) or “Security and Boot Function” (refer to Section 5.8.3)

Commands	Support requirements
CMD49	Optional Mandatory for cards supporting “Performance Enhancement Function” (refer to Section 5.8.2) or “Security and Boot Function” (refer to Section 5.8.3)
CMD50	Optional for cards version 1.10 and after
CMD52	Optional
CMD53	Optional
CMD55	Mandatory
CMD56	Mandatory
CMD57	Optional for cards version 1.10 and after
CMD58	Optional
CMD59	Optional
ACMD6	Mandatory
ACMD13	Mandatory
ACMD14	Optional
ACMD15	Optional
ACMD16	Optional
ACMD22	Mandatory for writable types of cards
ACMD23	Mandatory for writable types of cards
ACMD28	Optional
ACMD41	Mandatory
ACMD42	Mandatory
ACMD51	Mandatory
ACMD53	Optional Mandatory for cards supporting either TCG security or RPMB
ACMD54	Optional Mandatory for cards supporting either TCG security or RPMB

Table 4-22 : Command Support Requirements

4.7.4 Detailed Command Description

The following tables describe in detail all SD Memory Card bus commands. The responses R1-R3, R6 are defined in Section 4.9. The registers CID, CSD and DSR are described in Chapter 5. The card shall ignore stuff bits and reserved bits in an argument.

CMD INDEX	type	argument	resp	abbreviation	command description
CMD0	bc	[31:0] stuff bits or bus mode in Fast Boot	-	GO_IDLE_STATE	Resets all cards to idle state. When card supports boot functionalities and receives this command as the first one in idle state after power up, the argument is regarded as the bus mode in Fast Boot. Refer to Section 4.21.3.3.
CMD1	reserved				
CMD2	bcr	[31:0] stuff bits	R2	ALL_SEND_CID	Asks any card to send the CID numbers on the CMD line (any card that is connected to the host will respond)
CMD3	bcr	[31:0] stuff bits	R6	SEND_RELATIVE_ ADDR	Ask the card to publish a new relative address (RCA)
CMD4	bc	[31:16] DSR [15:0] stuff bits	-	SET_DSR	Programs the DSR of all cards
CMD5	reserved for I/O cards (refer to "SDIO Card Specification")				
CMD7	ac	[31:16] RCA [15:0] stuff bits	R1b (only from the selected card)	SELECT/DESELECT_ CARD	Command toggles a card between the stand-by and transfer states or between the programming and disconnect states. In both cases, the card is selected by its own relative address and gets deselected by any other address; address 0 deselects all. In the case that the RCA equals 0, then the host may do one of the following: - Use other RCA number to perform card de-selection. - Re-send CMD3 to change its RCA number to other than 0 and then use CMD7 with RCA=0 for card de-selection.
CMD8	bcr	[31:12]reserved bits [11:8]supply voltage(VHS) [7:0]check pattern	R7	SEND_IF_COND	Sends SD Memory Card interface condition, which includes host supply voltage information and asks the card whether card supports voltage. Reserved bits shall be set to '0'.
CMD9	ac	[31:16] RCA [15:0] stuff bits	R2	SEND_CSD	Addressed card sends its card-specific data (CSD) on the CMD line.
CMD10	ac	[31:16] RCA [15:0] stuff bits	R2	SEND_CID	Addressed card sends its card identification (CID) on CMD the line.
CMD11	ac	[31:0] reserved bits (all 0)	R1	VOLTAGE_SWITCH	Switch to 1.8V bus signaling level.
CMD12	ac	[31:0] stuff bits	R1b	STOP_ TRANSMISSION	Forces the card to stop transmission

CMD INDEX	type	argument	resp	abbreviation	command description
CMD13	ac	[31:16] RCA [15] Send Task Status Register [14:0] stuff bits	R1	SEND_STATUS/SEND_TASK_STATUS	CQ not enabled: [15] = ‘0’ or ‘1’, Addressed card sends its status register CQ enabled: [15]=’0’, Addressed card sends its status register. [15]=’1’, Addressed card sends task status register.
CMD14	Reserved				
CMD15	ac	[31:16] RCA [15:0] reserved bits	-	GO_INACTIVE_STATE	Sends an addressed card into the *Inactive State*. This command is used when the host explicitly wants to deactivate a card. Reserved bits shall be set to '0'.

Table 4-23 : Basic Commands (class 0)

CMD INDEX	type	argument	resp	abbreviation	command description
CMD16	ac	[31:0] block length	R1	SET_BLOCKLEN	In the case of a Standard Capacity SD Memory Card, this command sets the block length (in bytes) for all following block commands (read, write, lock). Default block length is fixed to 512 Bytes. Set length is valid for memory access commands only if partial block read operation are allowed in CSD. In the case of SDHC, SDXC and SDUC Cards, block length set by CMD16 command does not affect memory read and write commands. Always 512 Bytes fixed block length is used. This command is effective for LOCK_UNLOCK command. In both cases, if block length is set larger than 512Bytes, the card sets the BLOCK_LEN_ERROR bit. In DDR50 mode, data is sampled on both edges of the clock. Therefore, block length shall always be even.
CMD17	adtc	[31:0] data address[2]	R1	READ_SINGLE_BLOCK	In the case of a Standard Capacity SD Memory Card, this command, this command reads a block of the size selected by the SET_BLOCKLEN command. [1] In case of SDHC, SDXC and SDUC Cards, block length is fixed 512 Bytes regardless of the SET_BLOCKLEN command.

CMD INDEX	type	argument	resp	abbreviation	command description
CMD18	adtc	[31:0] data address[2]	R1	READ_MULTIPLE_ BLOCK	Continuously transfers data blocks from card to host until interrupted by a STOP_TRANSMISSION command. Block length is specified the same as READ_SINGLE_BLOCK command.
CMD19	adtc	[31:0] reserved bits (all 0)	R1	SEND_TUNING_ BLOCK	64 bytes tuning pattern is sent for SDR50 and SDR104.
CMD20	ac	[31:28]Speed Class Control [27:0]See command description	R1b	SPEED_CLASS_CON TROL	Speed Class control command. [27:0] Speed Class - 4.13.2.8 [27:0] UHS Speed Grade - See Section 4.13.2.8 [27:0] Video Speed Class - See Section 4.13.4.7
CMD22	ac	[31:6] reserved bits (all 0) [5:0] extended address	R1	ADDRESS_EXTENSI ON	Address extension for SDUC cards. For CMD17, CMD18, CMD24, CMD25, CMD32, CMD33, 6-bit extended address is assigned to upper part of 512B unit sector address. For CMD20 Set Free AU, 6-bit extended address is assigned to upper part of 512KB unit address.
CMD23	ac	[31:0] Block Count	R1	SET_BLOCK_COUNT	Specify block count for CMD18 and CMD25.

1) The data transferred shall not cross a physical block boundary unless READ_BLK_MISALIGN is set in the CSD.
2) SDSC Card (CCS=0) uses byte unit address and SDHC and SDXC Cards (CCS=1) use block unit address (512 Bytes unit).

Table 4-24 : Block-Oriented Read Commands (class 2)

CMD INDEX	type	argument	resp	abbreviation	command description
CMD16	ac	[31:0] block length	R1	SET_BLOCKLEN	See description in Table 4-24
CMD20	ac	[31:28]Speed Class Control [27:0]See command description	R1b	SPEED_CLASS_CONTRO L	Speed Class control command. [27:0] Speed Class - 4.13.2.8 [27:0] UHS Speed Grade - See Section 4.13.2.8 [27:0] Video Speed Class - See Section 4.13.4.7
CMD22	ac	[31:6] reserved bits (all 0) [5:0] extended address	R1	ADDRESS_EXTENSION	Address extension for SDUC cards. For CMD17, CMD18, CMD24, CMD25, CMD32, CMD33, 6-bit extended address is assigned to upper part of 512B unit sector address. For CMD20 Set Free AU, 6-bit extended address is assigned to upper part of 512KB unit address.
CMD23	ac	[31:0] Block Count	R1	SET_BLOCK_COUNT	Specify block count for CMD18 and CMD25.

CMD INDEX	type	argument	resp	abbreviation	command description
CMD24	adtc	[31:0] data address[2]	R1	WRITE_BLOCK	In case of SDSC Card, block length is set by the SET_BLOCKLEN command[1]. In case of SDHC, SDXC and SDUC Cards, block length is fixed 512 Bytes regardless of the SET_BLOCKLEN command.
CMD25	adtc	[31:0] data address[2]	R1	WRITE_MULTIPLE_BLOCK	Continuously writes blocks of data until a STOP_TRANSMISSION follows. Block length is specified the same as WRITE_BLOCK command.
CMD26	Reserved For Manufacturer				
CMD27	adtc	[31:0] stuff bits	R1	PROGRAM_CSD	Programming of the programmable bits of the CSD.

1) The data transferred shall not cross a physical block boundary unless WRITE_BLK_MISALIGN is set in the CSD. In the case that write partial blocks is not supported, then the block length=default block length (given in CSD).
2) SDSC Card (CCS=0) uses byte unit address and SDHC and SDXC Cards (CCS=1) use block unit address (512 bytes unit).

Table 4-25 : Block-Oriented Write Commands (class 4)

CMD INDEX	type	argument	resp	abbreviation	command description
CMD28	ac	[31:0] data address[2]	R1b	SET_WRITE_PROT	If the card has write protection features, this command sets the write protection bit of the addressed group. The properties of write protection are coded in the card specific data (WP_GRP_SIZE). SDHC, SDXC and SDUC Cards do not support this command.
CMD29	ac	[31:0] data address[2]	R1b	CLR_WRITE_PROT	If the card provides write protection features, this command clears the write protection bit of the addressed group. SDHC, SDXC and SDUC Cards do not support this command.
CMD30	adtc	[31:0] write protect data address[2]	R1	SEND_WRITE_ PROT	If the card provides write protection features, this command asks the card to send the status of the write protection bits.[1] SDHC, SDXC and SDUC Cards do not support this command.
CMD31	Reserved				

1) 32 write protection bits (representing 32 write protect groups starting at the specified address) followed by 16 CRC bits are transferred in a payload format via the data line. The last (least significant) bit of the protection bits corresponds to the first addressed group. If the addresses of the last groups are outside the valid range, then the corresponding write protection bits shall be set to 0.
2) Data address is in byte units in a Standard Capacity SD Memory Card.

Table 4-26 : Block Oriented Write Protection Commands (class 6)

CMD INDEX	type	argument	resp	abbreviation	command description
CMD32	ac	[31:0] data address[1]	R1	ERASE_WR_BLK_ START	Sets the address of the first write block to be erased.
CMD33	ac	[31:0] data address[1]	R1	ERASE_WR_BLK_END	Sets the address of the last write block of the continuous range to be erased.
CMD38	ac	[31:0] Erase Function	R1b	ERASE	Erase Function 00000001h = Discard 00000002h = FULE Others = Erase
CMD41	Reserved				

1) SDSC Card (CCS=0) uses byte unit address and SDHC and SDXC Cards (CCS=1) use block unit address (512 bytes unit).
2) CMD40 is moved to Table 4-28 (Class 7).

Table 4-27 : Erase Commands (class 5)

CMD INDEX	type	argument	resp	abbreviation	command description
CMD16	ac	[31:0] block length	R1	SET_BLOCKLEN	See description in Table 4-24
CMD40	adtc	Defined by DPS Spec.	R1	Defined by DPS Spec.	Single block read type. Available even if card is locked.
CMD42	adtc	[31:0] Reserved bits (Set all 0)	R1	LOCK_UNLOCK	Used to set/reset the password or lock/unlock the card. The size of the data block is set by the SET_BLOCK_LEN command. Reserved bits in the argument and in Lock Card Data Structure shall be set to 0.
CMD51	reserved				

Table 4-28 : Lock Card (class 7)

CMD INDEX	type	argument	resp	abbreviation	command description
CMD23	ac	[31:0] Block Count	R1	SET_BLOCK_COUNT	Specify block count for ACMD53 and ACMD54.
CMD55	ac	[31:16] RCA [15:0] stuff bits	R1	APP_CMD	Indicates to the card that the next command is an application specific command rather than a standard command
CMD56	adtc	[31:1] stuff bits. [0]: RD/WR	R1	GEN_CMD	Used either to transfer a data block to the card or to get a data block from the card for general purpose/application specific commands. In case of a SDSC Card, block length is set by the SET_BLOCK_LEN command. In case of SDHC, SDXC and SDUC Cards, block length is fixed to 512 bytes. The host sets RD/WR=1 for reading data from the card and sets to 0 for writing data to the card.
CMD60-63	reserved for manufacturer				

Table 4-29 : Application-Specific Commands (class 8)

All the application-specific commands (given in Table 4-29) are supported if Class 8 is allowed (mandatory in SD Memory Card).

CMD INDEX	type	argument	resp	abbreviation	command description
CMD52-54	Commands for SDIO (refer to "SDIO Card Specification")				

Table 4-30 : I/O Mode Commands (class 9)

All future reserved commands shall have a codeword length of 48 bits, as well as their responses (if there are any).

The following table describes all the application-specific commands supported/reserved by the SD Memory Card. All the following ACMDs shall be preceded with APP_CMD command (CMD55).

ACMD INDEX	type	argument	resp	abbreviation	command description
ACMD1-5	Reserved				
ACMD6	ac	[31:2] stuff bits [1:0]bus width	R1	SET_BUS_WIDTH	Defines the data bus width ('00'=1bit or '10'=4 bits bus) to be used for data transfer. The allowed data bus widths are given in SCR register.
ACMD7-12	Reserved				
ACMD13	adtc	[31:0] stuff bits	R1	SD_STATUS	Send the SD Status. The status fields are given in Table 4-44.
ACMD14-16	Reserved for DPS Specification				
ACMD17	Reserved				
ACMD18	Reserved for SD security applications[1]				

ACMD INDEX	type	argument	resp	abbreviation	command description
ACMD19-21	Reserved				
ACMD22	adtc	[31:0] stuff bits	R1	SEND_NUM_WR_ BLOCKS	Send the number of the written (without errors) write blocks. Responds with 32bit+CRC data block. If WRITE_BL_PARTIAL='0', the unit of ACMD22 is always 512 byte. If WRITE_BL_PARTIAL='1', the unit of ACMD22 is a block length which was used when the write command was executed.
ACMD23	ac	[31:23] stuff bits [22:0]Number of blocks	R1	SET_WR_BLK_ ERASE_COUNT	Set the number of write blocks to be pre-erased before writing (to be used for faster Multiple Block WR command). "1"=default (one wr block)[2]. Not supported by SDUC card.
ACMD24	Reserved				
ACMD25	Reserved for SD security applications[1]				
ACMD26	Reserved for SD security applications[1]				
ACMD27	Shall not use this command				
ACMD28	Reserved for DPS Specification				
ACMD29	Reserved				
ACMD30-35	Reserved for Security Specification				
ACMD36-37	Reserved				
ACMD38	Reserved for SD security applications[1]				
ACMD39-40	Reserved				
ACMD41	bcr	[31]reserved bit [30]HCS(OCR[30]) [29]reserved for eSD [28]XPC [27:25]reserved bits [24]S18R [23:0] V_{DD} Voltage Window(OCR[23:0])	R3	SD_SEND_OP_COND	Sends host capacity support information (HCS) and asks the accessed card to send its operating condition register (OCR) content in the response on the CMD line. HCS is effective when card receives SEND_IF_COND command. Sends request to switch to 1.8V signaling (S18R). Reserved bit shall be set to '0'. CCS bit is assigned to OCR[30]. XPC controls the maximum power in the default speed mode of SDXC and SDUC card. XPC=0 means 0.36W (100mA at 3.6V on VDD1) (max.) but speed class is not supported. XPC=1 means 0.54W (150mA at 3.6V on VDD1) (max.) and speed class is supported.

ACMD INDEX	type	argument	resp	abbreviation	command description
ACMD42	ac	[31:1] stuff bits [0]set_cd	R1	SET_CLR_CARD_ DETECT	Connect[1]/Disconnect[0] the 50 KOhm pull-up resistor on CD/DAT3 (pin 1) of the card.
ACMD43-49	--	--	--	--	Reserved for SD security applications[1]
ACMD51	adtc	[31:0] stuff bits	R1	SEND_SCR	Reads the SD Configuration Register (SCR).
ACMD52	Reserved for Security Specification				
ACMD53	adtc	[31:24] Security Protocol [23:16] Security Protocol Specific 1 (SPSP1) [15:8] Security Protocol Specific 0 (SPSP0) [7:0] SD Security Specific Field (SSSF) (4)	R1	SECURE_RECEIVE	Continuously transfers data blocks from device to host. [3] Bus timing of this command is equivalent to a multi-block read command, but the maximum data access time from the end bit of ACMD53 is 1 second.
ACMD54	adtc	[31:24] Security Protocol [23:16] Security Protocol Specific 1 (SPSP1) [15:8] Security Protocol Specific 0 (SPSP0) [7:0] SD Security Specific Field (SSSF) (4)	R1	SECURE_SEND	Continuously transfers data blocks from host to device. [3] Bus timing of this command is equivalent to a multi-block write command, but the maximum busy length to write by ACMD54 is 1 second.
ACMD55	Not exist				Equivalent to CMD55. Refer to Section 4.3.9.1.
ACMD56-59	Reserved for Security Specification				

1) Refer to "Part3 Security Specification" for a detailed explanation about the SD Security Features
2) Command STOP_TRAN (CMD12) shall be used to stop the transmission in Write Multiple Block whether or not the pre-erase (ACMD23) feature is used.
3) Security Protocol specific 0 and 1 relates to SP Specific 0 and SP Specific 1 of NVMe specification.
   Number of data blocks shall be defined by a preceding CMD23. Data transfer may be interrupted by a STOP_TRANSMISSION command.
   Block size is always 512bytes.
4) Refer to Table 4-96 for definition of SD Security Specific field for security protocol E7h. For all other security protocol, this field is reserved

Table 4-31 : Application Specific Commands used/reserved by SD Memory Card

Table 4-32 was added in version 1.10.

CMD INDEX	type	argument	resp	abbreviation	command description
CMD6	adtc	[31] Mode 0:Check function 1:Switch function [30:24] reserved (All '0') [23:20] reserved for function group 6 (0h or Fh) [19:16] reserved for function group 5 (0h or Fh) [15:12] function group 4 for Power Limit [11:8] function group 3 for Drive Strength [7:4] function group 2 for Command System [3:0] function group 1 for Access Mode	R1	SWITCH_FUNC	Checks switchable function (mode 0) and switch card function (mode 1). See Section 4.3.10.
CMD34	Reserved for each command system set by switch function command (CMD6). Detailed definition is referred to each command system specification.				
CMD35					
CMD36					
CMD37					
CMD50					
CMD57					

Table 4-32 : Switch Function Commands (class 10)

CMD INDEX	type	argument	resp	abbreviation	command description
CMD21	Reserved for DPS Specification				
CMD39	ac	[31:24] Partition ID [23:0] Reserved	R1b	SELECT_CARD_PARTITION	Select an active partition. Refer to Section 4.21.1.
CMD48	adtc	[31]MIO 0:Memory, 1:I/O [30:27] FNO [26] Reserved (=0) [25:9] ADDR [8:0] LEN	R1	READ_EXTR_SINGLE	Single block read type. Refer to Section 5.7.2.1.
CMD49	adtc	[31]MIO 0:Memory, 1:I/O [30:27] FNO [26] MW [25:9] ADDR [8:0] LEN/MASK	R1	WRITE_EXTR_SINGLE	Single block write type. Refer to Section 5.7.2.2.
CMD58	adtc	[31]MIO 0:Memory, 1:I/O [30:27] FNO [26] BUS 0:512B, 1:32KB [25:9] ADDR [8:0] BUC	R1	READ_EXTR_MULTI	Multi-block read type. Refer to Section 5.7.2.4.

CMD INDEX	type	argument	resp	abbreviation	command description
CMD59	adtc	[31]MIO 0:Memory, 1:I/O [30:27] FNO [26] BUS 0:512B, 1:32KB [25:9] ADDR [8:0] BUC	R1	WRITE_EXTR_MULTI	Multi-block write type. Refer to Section 5.7.2.5.

Note: CCC bit 11 is set to 1 when any command of class 11 is supported. Supporting of these commands is indicated in SCR register.

Table 4-33 : Function Extension Commands (class 11)

CMD INDEX	type	argument	resp	abbreviation	command description
CMD43	ac	[31:21] Reserved [20:16]: Task ID [3:0]: Operation Code (Abort tasks etc.)	R1b	Q_MANAGEMENT	Queue Management command [3:0] Operation Code: 0000b – Reserved 0001b – Abort Entire Queue 0010b – Abort Task ID 0011b to 1111b - Reserved
CMD44	ac	[31] Reserved [30] Direction [29:24] Extended Address [23] Priority [22:21] Reserved [20:16] Task ID [15:0] Number of Blocks	R1	Q_TASK_INFO_A	Task Parameters for Command Queuing [30] Direction: ‘1’ – Read ‘0’ – Write [23] Priority : ‘1’ – Priority request ‘0’ – No priority [29:24] upper 6bits ADDR of 38-bit 512B unit address, for SDUC cards
CMD45	ac	[31:0] Start block address	R1	Q_TASK_INFO_B	Task Parameters for Command Queuing
CMD46	adtc	[31:21] Reserved [20:16] Task ID [15:0] Reserved	R1	Q_RD_TASK	Executing Task Read operation
CMD47	adtc	[31:21] Reserved [20:16] Task ID [15:0] Reserved	R1	Q_WR_TASK	Executing Task Write operation

Table 4.7.4-1 : Command Queue Function Commands (class 1)

4.7.5 Difference of SD Commands Definition in UHS-II

Table 4-34 shows the difference of SD commands definition when the card is in UHS-II mode.
SD-TRAN driver of host should manage the difference of SD commands functions. Not supported commands should not issue to UHS-II card. CMD13 shall not be issued during data transfer. Normally, data transfer should be stopped by setting TLEN instead of using CMD12. CMD23 and CMD55 functions are included in UHS-II packet functions.

Command	Description
CMD0	Terminate SD transaction and reset SD-TRAN state.
CMD3	Returns Device ID in the response instead of RCA
CMD4	Illegal
CMD6	Function Group 1 and 3 are not used.
CMD7	Device ID is set to the argument instead of RCA
CMD13	Device operation is up to implementation during data transfer (e.g. CTS).
CMD11	Illegal
CMD12	Normally, TLEN (data length) in UHS-II packet is used to stop data transfer. CMD12 should be used to abort an operation when illegal situation occurs.
CMD15	Illegal
CMD19	Illegal
CMD23	Not Affected. TLEN in UHS-II packet is used to specify data length.
CMD39	Illegal
CMD55	Not Affected. ACMD is set by APP field in UHS-II packet.
ACMD6	Illegal
ACMD42	Illegal
ACMD53	Illegal
ACMD54	Illegal

(1) Not Affected means that the command is not executed in any card state, and response is returned (response type is up to implementation).
(2) Illegal means that card returns response with NACK=1.
(3) As SDHC/SDXC/SDUC Cards do not support CMD28, 29 and 30, these commands are also illegal in UHS-II mode.

Table 4-34 : Difference of SD Commands Definition in UHS-II

4.8 Card State Transition Table

Table 4-35 defines the card state transitions depend on the received command. State name in the table is the next state after the command is executed. "-" indicated that the command is treated as illegal command. In addition, whether a command is executable depends on command class (CCC).

Commands	Current State									
	idle	**ready**	**ident**	**stby**	**tran**	**data**	**rcv**	**prg**	**dis**	**ina**
"Operation Complete"	-	-	-	-	-	tran	-	tran	stby	-
class 0										
CMD0	idle/ boot[4]	idle	idle	idle	idle	idle	idle	idle	idle	-
CMD2	-	ident	-	-	-	-	-	-	-	-
CMD3	-	-	stby	stby	-	-	-	-	-	-
CMD4	-	-	-	stby	-	-	-	-	-	-
CMD7, card is addressed	-	-	-	tran	-	-	-	-	prg	-
CMD7, card is not addressed	-	-	-	stby	stby	stby	-	dis	-	-
CMD8	idle	-	-	-	-	-	-	-	-	-
CMD9	-	-	-	stby	-	-	-	-	-	-
CMD10	-	-	-	stby	-	-	-	-	-	-
CMD11	-	ready	-	-	-	-	-	-	-	-
CMD12	-	-	-	-	tran[3]	tran	prg	-	-	-
CMD13	-	-	-	stby	tran	data	rcv	prg	dis	-
CMD15	-	-	-	ina	ina	ina	ina	ina	ina	-
class 2										
CMD16	-	-	-	-	tran	-	-	-	-	-
CMD17	-	-	-	-	data	-	-	-	-	-
CMD18	-	-	-	-	data	-	-	-	-	-
CMD19	-	-	-	-	data	-	-	-	-	-
CMD20	-	-	-	-	prg	-	-	-	-	-
CMD22[3]	-	-	-	-	tran[3]	-	-	-	-	-
CMD23	-	-	-	-	tran	-	-	-	-	-
class 4										
CMD16	-	-	-	-	tran	-	-	-	-	-
CMD20	-	-	-	-	prg	-	-	-	-	-
CMD22[3]	-	-	-	-	tran[3]	-	-	-	-	-
CMD23	-	-	-	-	tran	-	-	-	-	-
CMD24	-	-	-	-	rcv	-	-	-	-	-
CMD25	-	-	-	-	rcv	-	-	-	-	-
CMD27	-	-	-	-	rcv	-	-	-	-	-
class 6										
CMD28	-	-	-	-	prg	-	-	-	-	-
CMD29	-	-	-	-	prg	-	-	-	-	-
CMD30	-	-	-	-	data	-	-	-	-	-
class 5										
CMD32	-	-	-	-	tran	-	-	-	-	-
CMD33	-	-	-	-	tran	-	-	-	-	-
CMD38	-	-	-	-	prg	-	-	-	-	-

Commands	Current State									
	idle	**ready**	**ident**	**stby**	**tran**	**data**	**rcv**	**prg**	**dis**	**ina**
class 7										
CMD40	-	-	-	-	data	-	-	-	-	-
CMD42	-	-	-	-	rcv	-	-	-	-	-
class 8										
CMD55	idle	-	-	stby	tran	data	rcv	prg	dis	-
CMD56; RD/WR = 0	-	-	-	-	rcv	-	-	-	-	-
CMD56; RD/WR = 1	-	-	-	-	data	-	-	-	-	-
ACMD6	-	-	-	-	tran	-	-	-	-	-
ACMD13	-	-	-	-	data	-	-	-	-	-
ACMD14-16	Refer to DPS Specification									
ACMD22	-	-	-	-	data	-	-	-	-	-
ACMD23	-	-	-	-	tran	-	-	-	-	-
ACMD28	Refer to DPS Specification									
ACMD18,25,26,38, 43,44,45,46,47,48,49	Refer to "Part3 Security Specification" for information about the SD Security Features									
ACMD41, OCR check is OK and card is not busy	ready	-	-	-	-	-	-	-	-	-
ACMD41, OCR check is OK and card is busy[2]	idle	-	-	-	-	-	-	-	-	-
ACMD41, OCR check fails	ina	-	-	-	-	-	-	-	-	-
ACMD41, query mode	idle	-	-	-	-	-	-	-	-	-
ACMD42	-	-	-	-	tran	-	-	-	-	-
ACMD51	-	-	-	-	data	-	-	-	-	-
ACMD53	-	-	-	-	data	-	-	-	-	-
ACMD54	-	-	-	-	rcv	-	-	-	-	-
class 9										
CMD52-CMD54	Refer to "SDIO Card Specification"									
class 10[1]										
CMD6	-	-	-	-	data	-	-	-	-	-
CMD34-37,50,57	Refer to each command system specification									
class 11										
CMD21	Refer to DPS Specification									
CMD39	-	-	-	-	prg	-	-	-	-	-
CMD48	-	-	-	-	data	-	-	-	-	-
CMD49	-	-	-	-	rcv	-	-	-	-	-
CMD58	-	-	-	-	data	-	-	-	-	-
CMD59	-	-	-	-	rcv	-	-	-	-	-
CMD41	reserved									
CMD60...CMD63	reserved for manufacturer									
class 1										
CMD43-CMD47	Illegal command in Non CQ mode. Refer to Table 4-75 for Card state transition in CQ mode									

Note (1): Class 10 commands were defined in Version 1.10.
Note (2): Card returns busy in case of following.
- Card executes internal initialization process
- When HCS in the argument is set to 0 to SDHC or SDXC Card.
Note(3): For SDUC only
Note(4): Transition to boot state takes place when valid argument for CA-mode Fast Boot is set in it. During boot

state, state transition (to idle state) occurs only when CMD0 is issued. Refer to Section 4.21.3.3 for more details.

Table 4-35 : Card State Transition Table

The state transitions of the SD Memory Card application-specific commands are given under Class 8, above.

4.9 Responses

All responses are sent via the command line CMD. The response transmission always starts with the left bit of the bit string corresponding to the response codeword. The code length depends on the response type.

A response always starts with a start bit (always 0), followed by the bit indicating the direction of transmission (card = 0). A value denoted by 'x' in the tables below indicates a variable entry. All responses except the type R3 (see below) are protected by a CRC (see Section 4.5 for the definition of CRC7). Every command codeword is terminated by the end bit (always 1).

There are five types of responses for the SD Memory Card. The SDIO Card supports additional response types named R4 and R5. Refer to SDIO Card Spec for detailed information on the SDIO commands and responses. Their formats are defined as follows:

4.9.1 R1 (normal response command):

Code length is 48 bits. The bits 45:40 indicate the index of the command to be responded to, this value being interpreted as a binary coded number (between 0 and 63). The status of the card is coded in 32 bits. Note that if a data transfer to the card is involved, then a busy signal may appear on the data line after the transmission of each block of data. The host shall check for busy after data block transmission. The card status is described in Section 4.10.

Bit position	47	46	[45:40]	[39:8]	[7:1]	0
Width (bits)	1	1	6	32	7	1
Value	'0'	'0'	x	x	x	'1'
Description	start bit	transmission bit	command index	card status	CRC7	end bit

Table 4-36 : Response R1

4.9.2 R1b

R1b is identical to R1 with an optional busy signal transmitted on the data line. The card may become busy after receiving these commands based on its state prior to the command reception. The Host shall check for busy at the response. Refer to Section 4.12.3 for a detailed description and timing diagrams.

4.9.3 R2 (CID, CSD register)

Code length is 136 bits. The contents of the CID register are sent as a response to the commands CMD2 and CMD10. The contents of the CSD register are sent as a response to CMD9. Only the bits [127...1] of the CID and CSD are transferred, the reserved bit [0] of these registers is replaced by the end bit of the response.

Bit position	135	134	[133:128]	[127:1]	0
Width (bits)	1	1	6	127	1
Value	'0'	'0'	'111111'	x	'1'
Description	start bit	transmission bit	reserved	CID or CSD register incl. internal CRC7	end bit

Table 4-37 : Response R2

4.9.4 R3 (OCR register)

Code length is 48 bits. The contents of the OCR register are sent as a response to ACMD41.

Bit position	47	46	[45:40]	[39:8]	[7:1]	0
Width (bits)	1	1	6	32	7	1
Value	'0'	'0'	'111111'	x	'1111111'	'1'
Description	start bit	transmission bit	reserved	OCR register	reserved	end bit

Table 4-38 : Response R3

4.9.5 R6 (Published RCA response)

Code length is 48 bit. The bits 45:40 indicate the index of the command to be responded to - in that case, it will be '000011' (together with bit 5 in the status bits it means = CMD3). The 16 MSB bits of the argument field are used for the Published RCA number.

Bit position	47	46	[45:40]	[39:8] Argument field		[7:1]	0
Width (bits)	1	1	6	16	16	7	1
Value	'0'	'0'	x	x	x	x	'1'
Description	start bit	transmission bit	command index ('000011')	New published RCA [31:16] of the card	[15:0] card status bits: 23,22,19,12:0 (see Table 4-42)	CRC7	end bit

Table 4-39 : Response R6

4.9.6 R7 (Card interface condition)

Code length is 48 bits. The card supported voltage information of 3.3V range power pin is sent by the response of CMD8. Bits 19-16 indicate the voltage range that the card supports. The card that accepted the supplied voltage returns R7 response. In the response, the card echoes back both the voltage range and check pattern set in the argument.

Bit position	47	46	[45:40]	[39:22]	21	20	[19:16]	[15:8]	[7:1]	0
Width (bits)	1	1	6	18	1	1	4	8	7	1
Value	'0'	'0'	'001000'	'00000h'	x	x	x	x	x	'1'
Description	Start bit	Transmission bit	Command index	Reserved bits	PCIe 1.2V Support 1)	PCIe Response 2)	Voltage accepted	Echo-back of check pattern	CRC 7	End bit

1) Card responds whether it supports VDD3 (1.2V power rail)
0b: Not supporting 1.2V
1b: Supporting 1.2V
2) Card responds PCIe acceptance
0b: Not accepted
1b: Accepted

Table 4-40 : Response R7

Table 4-41 shows the format of 'voltage accepted' in R7.

Voltage accepted	Value Definition
0000b	Not Defined
0001b	2.7-3.6V
0010b	Reserved for Low Voltage Range
0100b	Reserved
1000b	Reserved
Others	Not Defined

Table 4-41 : Voltage Accepted in R7

As described in Low Voltage Interface Addendum, ‘voltage accepted’ in R7 is set to 0010b after LVS Identification, including CA-mode Fast Boot.

4.10 Three Status Information of SD Memory Card

The SD Memory Card supports three status fields as follows:
- '*Card Status*': Error and state information of a executed command, indicated in the response
- *'SD Status*': Extended status field of 512 bits that supports special features of the SD Memory Card and future Application-Specific features.
-*'Task Status'*. Status information of queued tasks in CQ mode (Refer to Section 0), indicated in the response. CMD13 R1 indicates either Card Status or Task Status by selecting b[15] of CMD13 argument. R1 of the other commands indicates only Card Status.

4.10.1 Card Status

The response format R1 shall contain a 32-bit field named *card status* as a response to CMD13 with argument b[15]='0'. This field is intended to transmit the card's status information (which may be stored in a local status register) to the host. If not specified otherwise, the status entries are always related to the previous issued command.
Table 4-42 defines the different entries of the status. Unused reserved bits shall be set to 0. The type and clear condition fields in the table are abbreviated as follows:

- Type:
  •E: Error bit.
  •S: Status bit.
  •R: Detected and set for the actual command response.
  •X: Detected and set during command execution. The host can get the status by issuing a command with R1 response.

- Clear Condition:
  •A: According to the card current status.
  •B: Always related to the previous command. Reception of a valid command will clear it (with a delay of one command).
  •C: Clear by read.

Bits	Identifier	Type	Value	Description	Clear Condition
31	OUT_OF_RANGE	E R X	'0'= no error '1'= error	The command's argument was out of the allowed range for this card.	C
30	ADDRESS_ERROR	E R X	'0'= no error '1'= error	A misaligned address which did not match the block length was used in the command.	C
29	BLOCK_LEN_ERROR	E R X	'0'= no error '1'= error	The transferred block length is not allowed for this card, or the number of transferred bytes does not match the block length.	C
28	ERASE_SEQ_ERROR	E R	'0'= no error '1'= error	An error in the sequence of erase commands occurred.	C
27	ERASE_PARAM	E R X	'0'= no error '1'= error	An invalid selection of write-blocks for erase occurred.	C
26	WP_VIOLATION	E R X	'0'= not protected '1'= protected	Set when the host attempts to write to a protected block or to the temporary write protected card or write protected until power cycle card or permanent write protected card.	C
25	CARD_IS_LOCKED	S X	'0' = card unlocked '1' = card locked	When set, signals that the card is locked by the host	A
24	LOCK_UNLOCK_FAILED	E R X	'0' = no error '1' = error	Set when a sequence or password error has been detected in lock/unlock card command.	C
23	COM_CRC_ERROR	E R	'0'= no error '1'= error	The CRC check of the previous command failed.	B
22	ILLEGAL_COMMAND	E R	'0'= no error '1'= error	Command not legal for the card state	B
21	CARD_ECC_FAILED	E R X	'0'= success '1'= failure	Card internal ECC was applied but failed to correct the data.	C
20	CC_ERROR	E R X	'0'= no error '1'= error	Internal card controller error	C
19	ERROR	E R X	'0'= no error '1'= error	A general or an unknown error occurred during the operation.	C
18	reserved				
17	reserved for DEFERRED_RESPONSE (Refer to eSD Addendum)				
16	CSD_OVERWRITE	E R X	'0'= no error '1'= error	Can be either one of the following errors: - The read only section of the CSD does not match the card content. - An attempt to reverse the copy (set as original) or permanent WP (unprotected) bits was made.	C

Bits	Identifier	Type	Value	Description	Clear Condition
15	WP_ERASE_SKIP	E R X	'0'= not protected '1'= protected	"Set when only partial address space was erased due to existing write protected blocks or the temporary write protected or write protected until power cycle or permanent write protected card was erased.	C
14	CARD_ECC_DISABLED	S X	'0'= enabled '1'= disabled	The command has been executed without using the internal ECC.	A
13	ERASE_RESET	S R	'0'= cleared '1'= set	An erase sequence was cleared before executing because an out of erase sequence command was received	C
12:9	CURRENT_STATE (1)	S X	0 = idle 1 = ready 2 = ident 3 = stby 4 = tran 5 = data 6 = rcv 7 = prg 8 = dis 9-14 = reserved 15 = reserved for I/O mode	The state of the card when receiving the command. If the command execution causes a state change, it will be visible to the host in the response to the next command. The four bits are interpreted as a binary coded number between 0 and 15.	B
8	READY_FOR_DATA	S X	'0'= not ready '1'= ready	Corresponds to buffer empty signaling on the bus	A
7	reserved				
6	FX_EVENT	SX	'0'= No event '1'= Event invoked	Extension Functions may set this bit to get host to deal with events.	A
5	APP_CMD	S R	'0' = Disabled '1' = Enabled	The card will expect ACMD, or an indication that the command has been interpreted as ACMD	C
4	reserved for SD I/O Card				
3	AKE_SEQ_ERROR (SD Memory Card app. spec.)	E R	'0' = no error '1' = error	Error in the sequence of the authentication process	C
2	reserved for application specific commands				
1, 0	reserved for manufacturer test mode				

Note (1): Both p-init and boot states are not reported in CURRENT_STATE hence these states cannot be detected by issuing this command.

Table 4-42 : Card Status

Refer to Section 4.19.3 for ‘CURRENT_STATE’ value definition in CQ mode

For each command responded by R1 response, following table defines the affected bits in the status field.

An 'x' means the error/status bit may be set in the response to the respective command.

CMD Number	Response Format Card Status Bit Number																						
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12:9	8	6	5
3[1]									x	x			x							x			
6[2]	x						x		x	x	x	x	x							x		x	
7					x	x	x	x	x	x	x	x	x			x	x	x	x	x	x	x	
11							x		x	x			x							x			
12	x	x				x	x		x	x	x	x	x					x		x		x	
13	x	x			x	x	x	x	x	x	x	x	x			x	x	x		x	x	x	
16			x		x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	
17	x	x			x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	
18	x	x			x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	
19	x	x			x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	
20	x	x	x		x	x	x	x	x	x	x	x	x			x	x	x	x	x	x	x	
22	x	x	x		x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	
23	x	x	x		x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	
24	x	x	x		x	x	x	x	x	x	x	x	x			x	x	x	x	x	x	x	
25	x	x	x		x	x	x	x	x	x	x	x	x			x	x	x	x	x	x	x	
26					x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	
27					x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	
28	x				x	x	x	x	x	x	x	x	x			x	x	x	x	x			
29	x				x	x	x	x	x	x	x	x	x			x	x	x	x	x			
30	x				x	x	x	x	x	x	x	x	x			x	x	x	x	x			
32	x			x	x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	
33	x			x	x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	
38	x			x	x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	
39	x						x	x	x	x	x	x	x			x	x	x	x	x		x	
40	x	x	x		x	x	x	x	x	x	x	x	x			x	x	x	x	x	x	x	
42					x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	
43						x	x		x	x		x	x					x	x	x		x	
44						x	x		x	x		x						x	x	x		x	
45						x	x		x	x		x						x	x	x		x	
46	x					x	x		x	x	x	x	x					x	x	x		x	
47	x					x	x		x	x		x	x					x	x	x	x	x	
48	x	x	x		x	x	x	x	x	x	x	x	x			x	x	x	x	x	x	x	
49	x	x	x		x	x	x	x	x	x	x	x	x			x	x	x	x	x	x	x	
55					x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	x
56					x	x	x	x	x	x	x	x	x			x	x	x	x	x	x	x	x
58	x	x	x		x	x	x	x	x	x	x	x	x			x	x	x	x	x	x	x	
59	x	x	x		x	x	x	x	x	x	x	x	x			x	x	x	x	x	x	x	
ACMD6	x				x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	x
ACMD13					x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	x
ACMD22					x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	x
ACMD23					x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	x
ACMD42					x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	x
ACMD51					x	x	x	x	x	x	x	x	x			x	x	x	x	x		x	x
ACMD53					x	x	x	x	x	x	x	x	x			x	x	x	x	x	x	x	x
ACMD54					x	x	x	x	x	x	x	x	x			x	x	x	x	x	x	x	x

(1) The response to CMD3 is R6 that includes only bits 23, 22, 19 and 12:9 out of the Card Status
(2) This command was defined in version 1.10
(3) Bit 06 will be set in R1 (R1b) response after entering "tran" state.

Table 4-43 : Card Status Field/Command - Cross Reference

4.10.2 SD Status

The SD Status contains status bits that are related to the SD Memory Card proprietary features and may be used for future application-specific usage. The size of the SD Status is one data block of 512 bit. The content of this register is transmitted to the Host over the DAT bus along with a 16-bit CRC. The SD Status is sent to the host over the DAT bus as a response to ACMD13 (CMD55 followed with CMD13). ACMD13 can be sent to a card only in '*tran_state'* (card is selected). The SD Status structure is described in below. Unused reserved bits shall be set to 0.

The same abbreviation for 'type' and 'clear condition' were used as for the Card Status above.

Bits	Identifier	Type	Value	Description	Clear
511:510	DAT_BUS_WIDTH	S R	'00'= 1 (default) '01'= reserved '10'= 4 bit width '11'= reserved	Shows the currently defined data bus width that was defined by SET_BUS_WIDTH command	A
509	SECURED_MODE	S R	'0'= Not in the mode '1'= In Secured Mode	Shows if the card is in the secured mode of operation or not. (refer to "Part 3 Security Specification").	A
508:502	Reserved for Security Functions (Refer to “Part 3 Security Specification”)				
501:499	reserved				
498:496	SECURE_CMD_STATUS	S R	'000'= Successful Completion '001'= Invalid Field in command '010' = Command sequence error '011' = Access denied '100'-'111'= reserved	Provides card status after SECURE_SEND and SECURE_RECEIVE command **NOTE**: For further information refer to Extended Security Addendum specification	A
495:480	SD_CARD_TYPE	S R	'00xxh'= SD Memory Cards as defined by this document ('x'=don't care). The following cards are currently defined: '0000h'= Regular SD RD/WR Card. '0001h'= SD ROM Card '0002h'=OTP	In the future, the 8 LSBs will be used to define different variations of an SD Memory Card (Each bit will define different SD Types).	A
479:448	SIZE_OF_PROTECTED_AREA	S R	Size of protected area	(See Section 4.10.2.1)	A
447:440	SPEED_CLASS	S R	Speed Class of the card	(See Section 4.10.2.2)	A
439:432	PERFORMANCE_MOVE	S R	Performance of move indicated by 1 [MB/s] step.	(See Section 4.10.2.3)	A
431:428	AU_SIZE	S R	Size of AU	(See Section 4.10.2.4)	A
427:424	reserved				
423:408	ERASE_SIZE	S R	Number of AUs to be erased at a time	(See Section 4.10.2.5)	A

Bits	Identifier	Type	Value	Description	Clear
407:402	ERASE_TIMEOUT	S R	Timeout value for erasing areas specified by UNIT_OF_ERASE_AU	(See Section 4.10.2.6)	A
401:400	ERASE_OFFSET	S R	Fixed offset value added to erase time.	(See Section 4.10.2.7)	A
399:396	UHS_SPEED_GRADE	S R	Speed Grade for UHS mode	(See Section 4.10.2.8)	A
395:392	UHS_AU_SIZE	S R	Size of AU for UHS mode	(See Section 4.10.2.9)	A
391:384	VIDEO_SPEED_CLASS	S R	Video Speed Class value of the card	(See Section 4.10.2.10)	A
383:378	reserved				
377:368	VSC_AU_SIZE	S R	AU size in MB for Video Speed Class	(See Section 4.10.2.11)	A
367:346	SUS_ADDR	S R	Suspension Address	(See Section 4.10.2.12)	A
345:340	reserved				
339:336	APP_PERF_CLASS	S R	Application Performance Class Value of the card	(See Section 4.10.2.13)	A
335:328	PERFORMANCE_ENHANCE	S R	Support for Performance Enhancement functionalities	(See Section 4.10.2.14)	A
327:316	reserved				
315	WP_UPC_SUPPORT (1)	S R	'0'= Not supported '1'= Supported	Write Protect Until Power Cycle Support (See Section 4.23.5)	A
314	BOOT_PARTITION_SUPPORT	S R	'0'= Not supported '1'= Supported	Boot Partition Support (See Section 4.21.1)	A
313	DISCARD_SUPPORT	S R	'0' Not Supported '1' Supported	Discard Support. (See Section 4.3.6)	A
312	FULE_SUPPORT	S R	'0' Not Supported '1' Supported	Full User Area Logical Erase Support. (See Section 4.3.7)	A
311:0	reserved for manufacturer				

Note (1):Write Protect Until Power Cycle may be supported only if RPMB is enabled.

Table 4-44 : SD Status

4.10.2.1 SIZE_OF_PROTECTED_AREA

Setting this field differs between SDSC and SDHC/SDXC.

In case of SDSC Card, the capacity of protected area is calculated as follows:
Protected Area = SIZE_OF_PROTECTED_AREA_* MULT * BLOCK_LEN.
SIZE_OF_PROTECTED_AREA is specified by the unit in MULT*BLOCK_LEN.

In case of SDHC and SDXC Cards, the capacity of protected area is calculated as follows:
Protected Area = SIZE_OF_PROTECTED_AREA
SIZE_OF_PROTECTED_AREA is specified by the unit in byte.

In case of SDUC Cards, the capacity of protected area is zero.

For Non CPRM Cards, SIZE_OF_PROTECTED_AREA shall be set to the minimum value that is defined

in Appendix D.2 through D.5 of the Part 3 Security Specification Version 7.00.

4.10.2.2 SPEED_CLASS

This 8-bit field indicates the Speed Class. Classes lower than indicated by this field are also effective. For example, Class 10 is indicated, host should consider Class 2 to 6 is also effective.

SPEED_CLASS	Value Definition
00h	Class 0
01h	Class 2
02h	Class 4
03h	Class 6
04h	Class 10
05h – FFh	Reserved

Table 4-45 : Speed Class Code Field

Application Note:
If a Class value indicated in SD Status (including reserved value) is larger than that of host supported, the host should read as any Class can be used with the card.

4.10.2.3 PERFORMANCE_MOVE

This 8-bit field indicates Pm and the value can be set by 1 [MB/sec] step. If the card does not move used RUs, Pm should be considered as infinity. Setting to FFh means infinity. The minimum value of Pm is defined by in Table 4-46. Pm is defined for Class 2 to 6 in Default Speed Mode. When host uses Class 10, Pm indicated in SD Status shall be ignored and treated as 0.

PERFORMANCE_MOVE	Value Definition
00h	Sequential Write
01h	1 [MB/sec]
02h	2 [MB/sec]
.....	.....
FEh	254 [MB/sec]
FFh	Infinity

Table 4-46 : Performance Move Field

4.10.2.4 AU_SIZE

This 4-bit field indicates AU Size and the value can be selected from 16 KB.

AU_SIZE	Value Definition
0h	Not Defined
1h	16 KB
2h	32 KB
3h	64 KB
4h	128 KB
5h	256 KB
6h	512 KB
7h	1 MB
8h	2 MB
9h	4 MB
Ah	8 MB
Bh	12 MB
Ch	16 MB
Dh	24 MB
Eh	32 MB
Fh	64 MB

Table 4-47 : AU_SIZE Field

The maximum AU size, depends on the card capacity, is defined in Table 4-48. The card can set any AU size specified in Table 4-47 that is less than or equal to the maximum AU size. The card should set smaller AU size as much as possible.
AU_SIZE can be selected as integer multiple of exFAT cluster size. Regarding cluster size, refer to Appendix C.5.3 of the Part 2 File System Specification Version 7.00.

Card Capacity	up to 64MB	up to 256MB	up to 512MB	up to 32GB	up to 128TB
Maximum AU Size	512 KB	1 MB	2 MB	4 MB	64MB

Table 4-48 : Maximum AU size

Application Notes:
The host should determine host buffer size based on total busy time of 4MB and the card supported class. The host can treat multiple AUs combined as one unit.

4.10.2.5 ERASE_SIZE

This 16-bit field indicates N_{ERASE}. When N_{ERASE} numbers of AUs are erased, the timeout value is specified by ERASE_TIMEOUT (Refer to ERASE_TIMEOUT). The host should determine proper number of AUs to be erased in one operation so that the host can indicate progress of erase operation. If this field is set to 0, the erase timeout calculation is not supported.

ERASE_SIZE	Value Definition
0000h	Erase Time-out Calculation is not supported.
0001h	1 AU
0002	2 AU
0003	3 AU
.....	.....
FFFFh	65535 AU

Table 4-49 : Erase Size Field

4.10.2.6 ERASE_TIMEOUT

This 6-bit field indicates the T_{ERASE} and the value indicates erase timeout from offset when multiple AUs

are erased as specified by ERASE_SIZE. The range of ERASE_TIMEOUT can be defined as up to 63 seconds and the card manufacturer can choose any combination of ERASE_SIZE and ERASE_TIMEOUT depending on the implementation. Once ERASE_TIMEOUT is determined, it determines the ERASE_SIZE. The host can determine timeout for any number of AU erase by the Equation (9). Refer to Section 4.14 for the concept of calculating erase timeout. If ERASE_SIZE field is set to 0, this field shall be set to 0.

ERASE_TIMEOUT	Value Definition
00	Erase Time-out Calculation is not supported.
01	1 [sec]
02	2 [sec]
03	3 [sec]
.....	.....
63	63 [sec]

Table 4-50 : Erase Timeout Field

4.10.2.7 ERASE_OFFSET

This 2-bit field indicates the T_{OFFSET} and one of four values can be selected. The erase offset adjusts the line by moving in parallel on the upper side. Refer to Figure 4-65 and Equation (9) in Section 4.14. This field is meaningless if ERASE_SIZE and ERASE_TIMEOUT fields are set to 0.

ERASE_OFFSET	Value Definition
0h	0 [sec]
1h	1 [sec]
2h	2 [sec]
3h	3 [sec]

Table 4-51 : Erase Offset Field

4.10.2.8 UHS_SPEED_GRADE

This 4-bit field indicates the UHS mode Speed Grade. Reserved values are for future speed grades larger than the highest defined value. Host shall treat reserved values (undefined) as highest grade defined.

UHS_SPEED_GRADE	Value Definition
0h	Less than 10MB/sec
1h	10MB/sec and above
2h	Reserved
3h	30MB/sec and above
4h – Fh	Reserved

Table 4-52 : UHS_SPEED_GRADE Field

4.10.2.9 UHS_AU_SIZE

This 4-bit field indicates AU Size for UHS Speed Grade of UHS-I and UHS-II cards. Card should set smaller value as much as possible. Host shall refer to UHS_AU_SIZE instead of AU_SIZE when the card is operating as UHS Speed Grade in UHS-I or UHS-II bus speed modes.
UHS_AU_SIZE can be selected as integer multiple of exFAT cluster size. Regarding cluster size, refer to Appendix C.5.3 of the Part 2 File System Specification Version 7.00.

UHS_AU_SIZE	Value Definition
0h	Not Defined
1h -6h	Not Used
7h	1 MB
8h	2 MB
9h	4 MB
Ah	8 MB
Bh	12 MB
Ch	16 MB
Dh	24 MB
Eh	32 MB
Fh	64 MB

Table 4-53 : UHS_AU_SIZE Field

4.10.2.10 VIDEO_SPEED_CLASS

Table 4-54 defines an 8-bit field that indicates the Video Speed Class supported by the card. The Video Speed Class defined in the table shall be equal or less than the minimum write performance of the card in Video Speed Class as specified in Section 4.13.4.3.1 (e.g., Video Speed Class 6 supports at least 6MB/s). Any Video Speed Class lower than the specified Video Speed Class value shall also be supported (e.g., if the value in this field is 1Eh, then Video Speed Class 10 and Video Speed Class 6 are supported).

VIDEO_SPEED_CLASS	Value Definition
0 (=00h)	Video Speed Class 0 (Video Speed Class is not supported)
6 (=06h)	Video Speed Class 6
10 (=0Ah)	Video Speed Class 10
30 (=1Eh)	Video Speed Class 30
60 (=3Ch)	Video Speed Class 60
90 (=5Ah)	Video Speed Class 90
Others	Reserved

Table 4-54 : VIDEO_SPEED_CLASS Field

4.10.2.11 VSC_AU_SIZE

Table 4-55 defines a 10-bit field that indicates the AU Size for Video Speed Class recording. It is recommended that cards report the smallest value if multiple values are valid. This field shall be set to one of the specified AU sizes described in Table 4-56 or zero (e.g., if VSC_AU_SIZE is 480MB, 1E0h (=480) is set in this field). Note that the other values which are not specified in Table 4-56 are reserved. Since these reserved values may be assigned by SDA in future, card manufacturers shall not use these reserved values until SDA defines those in future versions of specification.

And SU size is also determined by this field, because a unique SU size is bound for each AU size. For example, if VSC_AU_SIZE is set as 160MB, SU size is 8MB as shown in Table 4-56.

VSC_AU_SIZE	Value Definition
0	Video Speed Class is not supported
1 to 1023 (=3FFh)	AU Size for Video Speed Class of the card in unit of [MB]

Table 4-55 : VSC_AU_SIZE Field

VSC_AU_SIZE	AU Size	SU Size
008h	8MB	8MB
010h	16MB	8MB
015h	21MB	7MB
018h	24MB	8MB
01Bh	27MB	9MB
01Ch	28MB	7MB
01Eh	30MB	10MB
020h	32MB	8MB
024h	36MB	9MB
028h	40MB	8MB
02Ah	42MB	7MB
02Dh	45MB	9MB
030h	48MB	8MB
036h	54MB	9MB
038h	56MB	8MB
03Ch	60MB	10MB
040h	64MB	8MB
048h	72MB	8MB
050h	80MB	8MB
060h	96MB	8MB
070h	112MB	8MB
078h	120MB	8MB
080h	128MB	8MB
090h	144MB	8MB
0A0h	160MB	8MB
0C0h	192MB	8MB
0D8h	216MB	8MB
0E0h	224MB	8MB
0F0h	240MB	8MB
100h	256MB	8MB
120h	288MB	8MB
140h	320MB	8MB
180h	384MB	8MB
1B0h	432MB	8MB
1C0h	448MB	8MB
1E0h	480MB	8MB
200h	512MB	8MB

Table 4-56 : Valid AU Size and SU Size

SDHC cards shall have a minimum of 40 AU blocks (e.g., for a 4GB card, AU sizes over 96MB do not result in 40 AUs). SDXC cards shall have a minimum of 80 AU blocks.
VSC_AU_SIZE can be selected as integer multiple of exFAT cluster size. Regarding cluster size, refer to Appendix C.5.3 of the Part 2 File System Specification Version 7.00.

4.10.2.12 SUS_ADDR

Table 4-57 defines a 22-bit field that indicates a valid suspension address in 512KB units. If this field is set to zero, then this card does not have a valid suspension address. If this field is non-zero, then a valid suspension address is reported by this field.

SUS_ADDR	Value Definition
0	No valid suspension address
Non-zero	Valid suspension address in 512KB units

Table 4-57 : SUS_ADDR Field

A valid suspension address is set by the CMD20 "Suspend AU" command (see Section 4.13.4.7.5).

The suspension address is cleared by any of these events:

a) the host issues a CMD20 "Set Free AU" command (see Section 4.13.4.7.7);
b) the host writes to any location within the suspended AU; and
c) the host erases any location within the suspended AU.

Also refer to Section 8.4.8.2 if card supports PCIe bus.

4.10.2.13 APP_PERF_CLASS

Table 4.10.2-1 defines a 4-bit field that indicates the Application Performance Class supported by the card. The Application Performance Class defined in the table shall be equal or less than the average random performance of the card as specified in Section 4.16.1.

APP_PERF_CLASS	Value Definition
0h	Application Performance Class is not supported
1h	A1, Application Performance Class 1
2h	A2, Application Performance Class 2
3h – Fh	Reserved

Table 4.10.2-1 : APP_PERF_CLASS Field

Application Notes:
For any APP_PERF_CLASS indicated by the card, it is assumed that the lower classes are supported as well. Then Application Performance supported Hosts should assume that higher APP_PERF_CLASS levels may be set in the future. For example, A1 host should interpret APP_PERF_CLASS; not only 1h but also higher value indicates support of Application Class 1.

4.10.2.14 PERFORMANCE_ENHANCE

Table 4.10.2-2 defines an 8-bit field that indicates the performance enhancement functionalities supported by the card.

PERFORMANCE_ENHANCE	Description
SD_STATUS b[335:331]	Command Queue Support. Refer to Table 4.10.2-3
SD_STATUS b[330]	Support for Cache 0 : Not Supported 1: Supported
SD_STATUS b[329]	Support for Host-initiated maintenance 0 : Not Supported 1: Supported
SD_STATUS b[328]	Support for Card-initiated maintenance 0 : Not Supported 1: Supported

Table 4.10.2-2 : PERFORMANCE_ENHANCE Field

SD Status b[335:331], as in, Table 4.10.2-3 indicates the command queue support and the depth required to meet Application Performance Class 2.

Command Queue Support	Value Definition
0h	Command Queue is not supported
1h	Command Queue supported, with queue depth 2
2h	Command Queue supported, with queue depth 3
….	
1Fh	Command Queue supported, with queue depth 32

Table 4.10.2-3 : Command Queue Support Field

4.10.2.15 Notes for SD Status

The following SD Status Identifiers are ignored for Boot Partitions (refer to Section 4.21.1) and User Area before pre-boot authentication (refer to Section 4.22.2),

Identifier	Note
SPEED_CLASS	Speed Class is not supported for a User Area before pre-boot authentication, and Boot Partitions.
PERFORMANCE_MOVE	
AU_SIZE	
ERASE_SIZE	Erase cannot be operated for a User Area before pre-boot authentication, and Boot Partitions.
ERASE_TIMEOUT	
ERASE_OFFSET	
UHS_SPEED_GRADE	UHS Speed Grade is not supported for a User Area before pre-boot authentication, and Boot Partitions.
UHS_AU_SIZE	
VIDEO_SPEED_GRADE	Video Speed Class is not supported for a User Area before pre-boot authentication, and Boot Partitions.
VIDEO_AU_SIZE	
SUS_ADDR	
APP_PERF_CLASS	Application Performance Class is not supported for a User Area before pre-boot authentication, and Boot Partitions.
PERFORMANCE_ENHANCE	
DISCARD_SUPPORT	Discard cannot be operated for a User Area before pre-boot authentication, and Boot Partitions.
FULE_SUPPORT	FULE cannot be operated for a User Area before pre-boot authentication, and Boot Partitions.

Table 4.10.2-4 : SD Status for a User Area before Pre-Boot Authentication and Boot Partitions

In addition, even card supports TCG security, the following identifiers can be set based on the performance when TCG is disabled (refer to Section 3.20):

- SPEED_CLASS
- UHS_SPEED_GRADE
- VIDEO_SPEED_GRADE
- APP_PERF_CLASS

4.10.3 Task Status

The response format R1 shall contain a 32-bit field named *task status* as a response to CMD13 with argument b[15]='1' . This field is intended to transmit the queued task's status information (which may be stored in a local status register) to the host.

Table 4.10.3-1 defines the different entries of the status. Unused reserved bits shall be set to 0. The type and clear condition fields in the table are abbreviated as follows:

Bits	Identifier	Type	Value	Description	Clear Condition
31	STATUS_TASK_31	S	‘0’ – Not Ready ‘1’ - Ready	Ready status of queued task 31	A
30	STATUS_TASK_30	S	‘0’ – Not Ready ‘1’ - Ready	Ready status of queued task 30	A
…					
0	STATUS_TASK_0	S	‘0’ – Not Ready ‘1’ - Ready	Ready status of queued task 0	A

Table 4.10.3-1 : Task Status

4.11 Memory Array Partitioning

This section is a blank in the Simplified Specification .

4.12 Timings

This section is a blank in the Simplified Specification.

4.13 Speed Class Specification

This section provides descriptions of several speed class methods. Each classifies card performance by a speed class number and offers a method to calculate performance. These methods enable the host to support AV applications to perform real time recording to an SD memory card. The following sections describe the speed class methods for the card.

Section 4.13.1 and 4.13.2 provides a description of the Speed Class specification. The Speed Class specification is supported by SDHC, SDXC and SDUC cards. Speed Class 2, Speed Class 4, Speed Class 6, and Speed Class 10 are defined in this method.

Section 4.13.3 provides a description of the UHS Speed Grade specification. The UHS Speed Grade specification is supported by SDHC, SDXC and SDUC cards. UHS Speed Grade 1 and UHS Speed Grade 3 are defined in this method,

Section 4.13.4 provides a description of the Video Speed Class specification. The Video Speed Class specification is supported by SDHC, SDXC and SDUC cards. Video Speed Class 6, Video Speed Class 10, Video Speed Class 30, Video Speed Class 60 and Video Speed Class 90 are defined in this method.

Section 4.13.5 and 8.4 provide a description of the SD Express Speed Class specification. The SD Express Speed Class specification is supported by SDXC and SDUC cards over PCIe bus. In other words, this specification is applied to SD Express card only. SD Express Speed Class 150, SD Express Speed Class 300, SD Express Speed Class 450, and SD Express Speed Class 600 are defined in this method.

In version 4.20 or earlier, implementation of the Speed Class was mandatory for SDHC and SDXC cards and the UHS Speed Grade method was optional for all types of cards. From version 5.00, implementation of Speed Class was changed to optional for all types of cards.

Figure 4-46 specifies the speed classes that are supported by each interface mode.

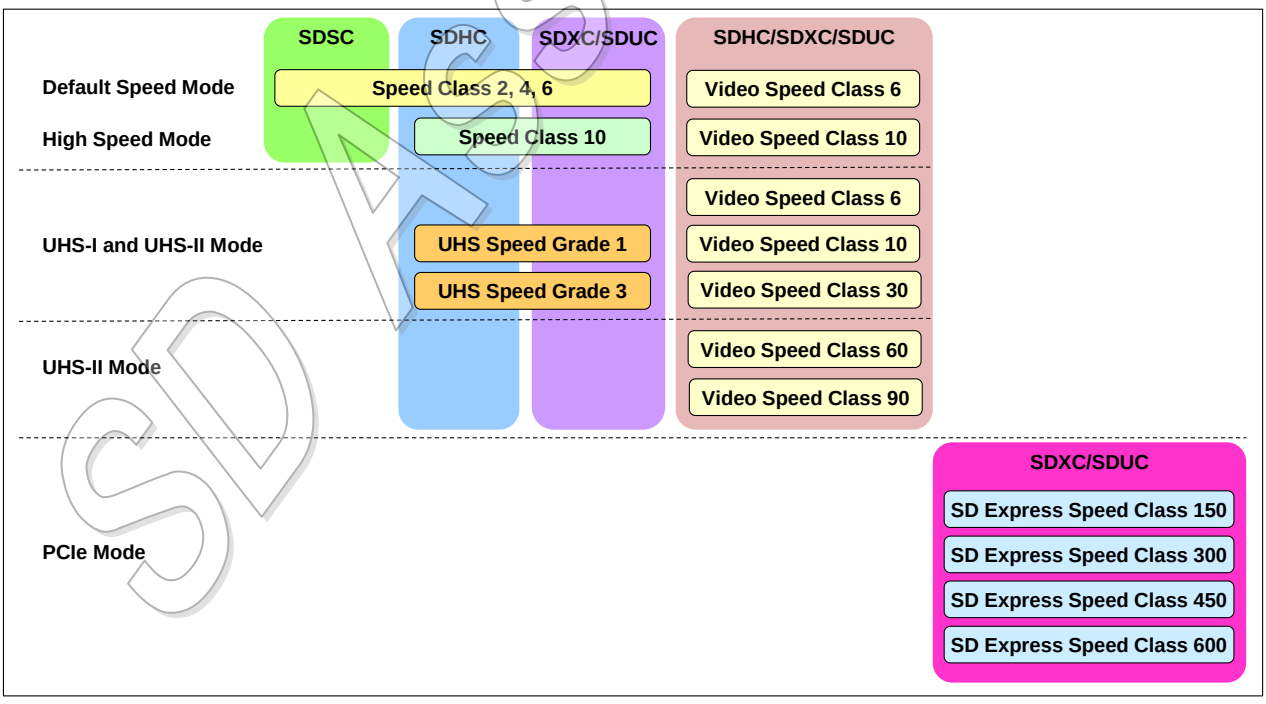


Figure 4-46 : Overview of Speed Class Specification

4.13.1 Speed Class Specification for SDSC and SDHC

4.13.1.1 Allocation Unit (AU)

The User Area is divided into units called "**Allocation Unit (AU)**" (Refer to Figure 4-47). AU is physical boundary in User Area of a card and is not defined by the file system boundary. Each card has its own fixed **AU Size (S_{AU})** and the maximum AU Size is defined depending on the card's capacity. The host should manage data areas with the unit of AU. If the first AUs in the card contain file system information then they should not be used for real time recording. An AV application should start recording from the first complete AU, to which only user data can be recorded. Note that this specification does not apply to the Protected Area.

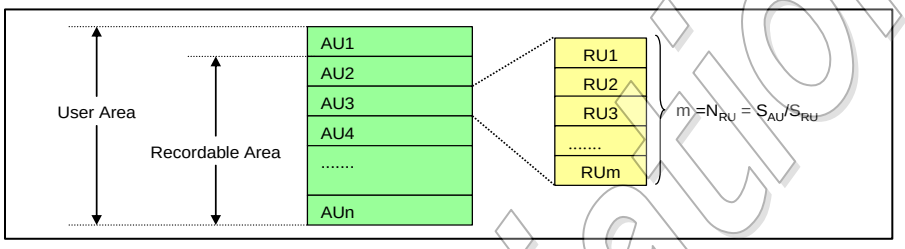


Figure 4-47 : Definition of Allocation Unit (AU)

4.13.1.2 Recording Unit (RU)

Each AU is divided into units called **"Recording Unit (RU)"** (Refer to Figure 4-47). The unit of **RU Size (S_{RU})** is 16KByte. The RU Size is a multiple of 16KByte and shall not span across an AU boundary.
Larger RU size may improve performance. The condition and requirement of the minimum RU Size is defined by Section 4.13.1.8.1. **The number of RUs in an AU (N_{RU})** is calculated from S_{AU}/S_{RU}.

4.13.1.3 Write Performance

Figure 4-48 shows the typical data management of the card when the host writes RUs of an AU. When the host writes to a fragmented AU, the card prepares a new AU by copying the used RUs and writing the new RUs. The location A is at the start of the AU boundary and location B is at the end of the AU boundary. From A to B, the host shall write data to free RUs contiguously and skip used RUs (shall not skip any free RU). The card may indicate busy to the host, so the host can wait, during the time the card controller is writing and moving data. The total write time from A to B can be calculated by summing up the write time of free RUs and the moving time of the used RUs. The number of used RUs (Nu) is available by counting it over one AU and number of free RUs is expressed by (N_{RU} – Nu).

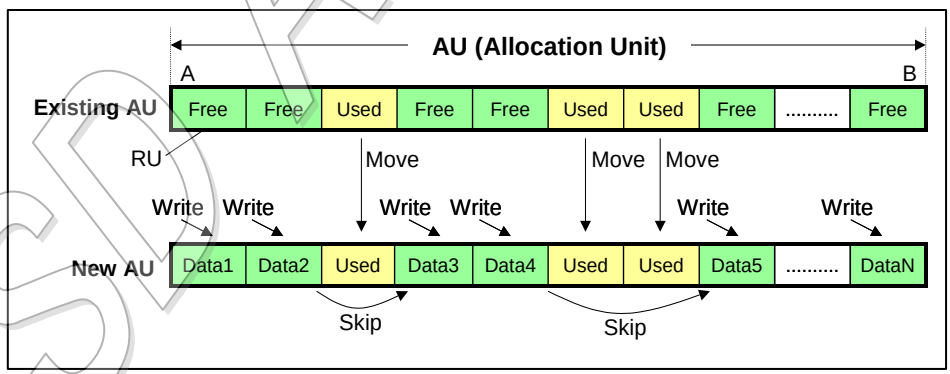


Figure 4-48 : Example of Writing Fragmented AU

The average Performance of a Fragmented AU can be calculated by dividing the number of free RUs by the total execution time. It is expressed by using Performance Wite (Pw) and Performance Move (Pm).

Performance of Fragmented AU: $$P(Nu) = \frac{S_{RU}(N_{RU} - Nu)}{\frac{S_{RU}(N_{RU} - Nu)}{Pw} + \frac{S_{RU}Nu}{Pm}} = \frac{(N_{RU} - Nu)PmPw}{(N_{RU} - Nu)Pm + NuPw} \quad \text{.....(1)}$$

The Performance Write (Pw) is defined as a minimum average write performance over an AU. It is calculated by taking the average of all sequential RU write operations to one complete AU, which is not fragmented. Pw is not influenced even if read operations are inserted between write operations.

The Performance Move (Pm) is defined as a minimum average move performance. It is calculated by taking the average over sequential RU move operations to one complete AU. A move is an internal operation of the card, so SD clock frequency does not affect the time of the move operation. In case the card does not have to move RU, Pm should be considered as infinity (1/Pm = 0). Refer to Table 4-61 for the values defined for each Speed Class.

Note that a Speed Class that supports Class10 shall not use the Pm value stored in the SD Status to calculate performance in any fragmented AU. Class 10 performance is defined only for entirely free AUs.

Application Notes: Performance may increase when larger data is written by one multiple write command. Therefore, the host may use larger RU sizes and transfer multiple RUs with one multiple-write command.

4.13.1.4 Read Performance

Two kinds of read performances are defined. It is possible to insert either type of read operation during write operations. All read operations, regardless of read address shall meet this performance specification.

(1) Read Performance of Stream Data
This is simply called **Read Performance (Pr). Pr** is defined as minimum average random RU read performance. The average is measured over 256 random single RU read operations. Each RU is read by a multiple-read command. Pr shall be greater than or equal to Pw.

(2) FAT and Directory Entry Read Time
T_{FR}(4KB) is defined as the maximum time to read a 4KB FAT and Directory Entry. The **FAT and Directory Entry Read Time** (**S_{FR} [KB]**) is defined using the CEIL function:

FAT Read Time of S_{FR} [KB]: $$T_{FR}(S_{FR}) = \left\lceil \frac{S_{FR}}{4KB} \right\rceil \cdot T_{FR}(4KB) \quad \text{.....(2)}$$

(CEIL function - Convert decimal fraction x to the smallest integer greater than or equal to x.)

Refer to Table 4-61 for the values defined for each Speed Class.

4.13.1.5 Performance Curve Definition

Figure 4-49 shows the write performance bar chart of P(Nu) of equitation (1). An AU consists of 16 RUs in this example. Joining the points of each bar shows the performance curve, which can be determined from the two parameters, Pw and Pm.

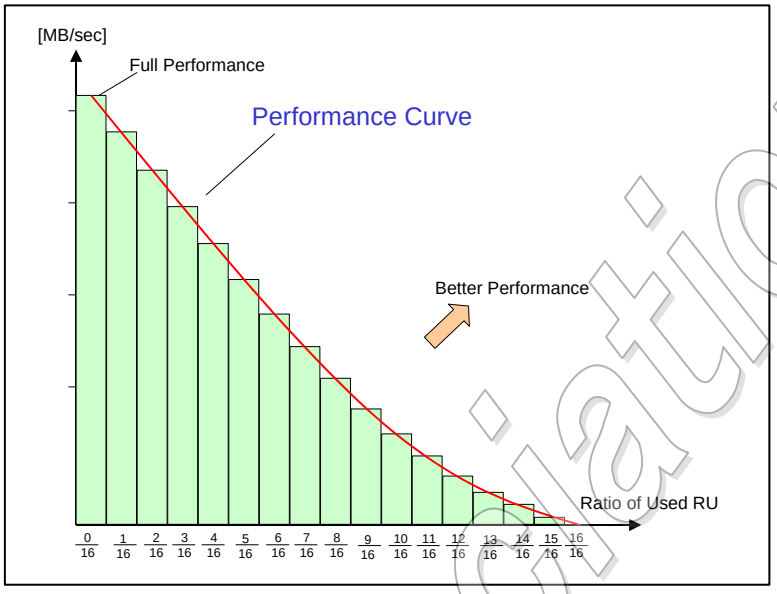


Figure 4-49 : Card Performances between 16 RUs

The ratio of used RU (r) is defined as:

$$r = \frac{Nu}{N_{RU}}, \qquad Nu = rN_{RU}$$

The range of r is 0 to 1. (1 – r) means ratio of free RU, r=0 means all RUs are free. r=1 means all RUs are used and performance indicates zero at this point. By using r, Equation (1) is transformed into Equation (3).

Performance Curve: $$P(r) = \frac{(1-r)PwPm}{rPw + (1-r)Pm} \quad (0 \le r \le 1) \quad \text{.....(3)}$$

P(Nu) in Equation (1) is a discrete function but P(r) is treated as a continuous function.

4.13.1.6 Speed Class Definition

Figure 4-50 shows three performance curves. Pw indicates the performance of r=0 and Pm determines the shape of the curve. All performance curves converge at the point (1, 0). Therefore, there is little difference in performance where r is near to 1. These three curves divide the performance into four speed classes: Class 0, Class 2, Class4 and Class 6. The **Class 0** card provides no guarantee to be compliant to the Speed Class Specification. It does not report performance parameters even if the cards can achieve performance of higher speed classes. Class 0 also covers all legacy SD products prior to the introduction of this specification. The Classes are defined so that an AV application, such as MPEG2 recording, can support an SD card device. The performance of a Speed **Class 2** card shall be higher than performance curve 2. It is defined for standard TV image quality; approximately 2MB/sec performance will be required. The performance of a Speed **Class 4** card shall be higher than performance curve 4. Speed Class 4 is defined for HD video quality; approximately 4MB/sec performance will be required. Higher classes can be added in the future, if required. It is important that the host shall always accept cards which meet

minimum speed class performance.
Note that performance of Class 10 does not conform to the performance curve. Class 10 is supported only in the case r=0.

Speed Class shall be defined as SD Bus interface level performance, though the performance curve is derived from only back-end performance analysis in Section 4.13.1.3. SD clock frequency and RU size are defined as measurement conditions for Speed Classes. Refer to Section 4.13.1.8.

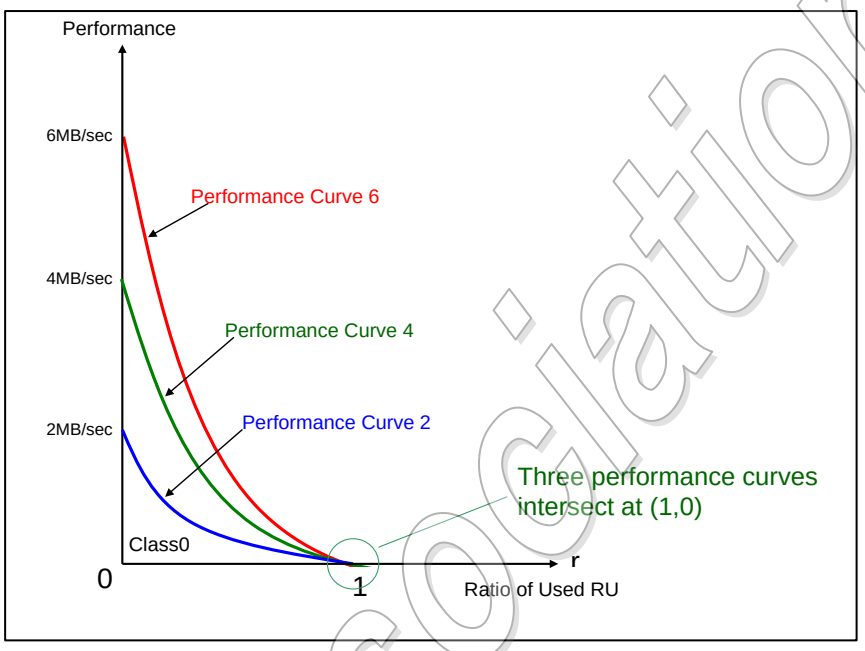


Figure 4-50 : Three Performance Curves

Application Note: For the convenience of legacy card users, the host should try to use the card that has lower performance than expected and attempt to record if necessary. When a mode provides operation only for specific Speed Class cards, one of the other modes should provide operation for lower Speed Class cards including Class 0.

4.13.1.7 Consideration for Inserting FAT Update during Recording

Figure 4-51 shows the typical sequence of the FAT update cycle for real time recording. FAT updates can be inserted between any RU accesses. The FAT update cycle consists of 3 write operations. The FAT1 and FAT2 means two FAT-table writes using one multiple write command for each FAT-table. The FAT table write can start any 512-byte boundary address and any size up to 16 KBytes. Only modified parts of FAT should be written. The DIR stands for directory entry write. A directory entry should be created before recording starts and only the modified parts should be written in the directory entry (512 Bytes). The **FAT Write Time (T_{FW})** is defined as the total time of 3 write operations of FAT update cycle. The host shall take the sequence to be able to calculate degradation of performance by inserting a FAT update cycle. The card requires higher Card Performance (Pc) than Application Performance (Pa) to insert FAT update cycle. It is noted that any order of 3 write operations is allowed for FAT update. Figure 4-51 shows an example order.

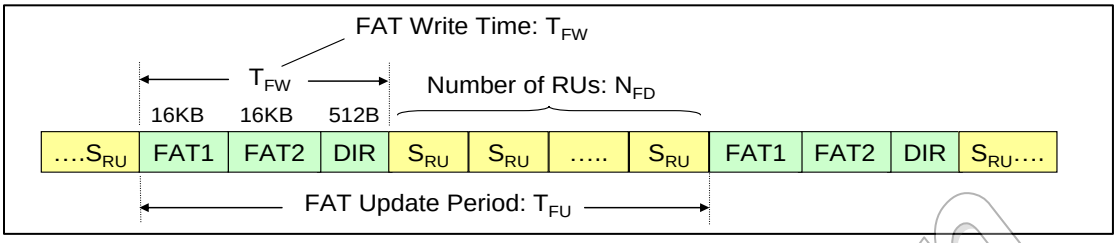


Figure 4-51 : Typical Sequence of FAT Update

4.13.1.7.1 Measurement Condition to determine Average T_{FW}

The equation (4) defines **Average FAT Write Time (T_{FW}(ave.))**, which is the maximum sliding average of 8 times FAT write cycles.

$$\text{Average FAT Write Time: } T_{FW}(ave.) = \frac{\max(\sum_{i=1}^{8} T_{FW}(i))}{8} \quad \text{.....(4)}$$

4.13.1.7.2 Maximum FAT Write Time

During a FAT update, the host cannot write data to the card. Therefore, the host should prepare enough buffers to save the data temporarily. The **Maximum FAT Write Time (T_{FW} (max.))** is one of the factors to determine host buffer size. During 8 times FAT write cycle, occurrence of T_{FW} (max.) should not appear more than once. On the method of Host Buffer Size estimation, refer to Implementation Guideline of the Speed Class Specification.

$$\text{Maximum FAT Write Time: } T_{FW}(\max.) \le 750ms \quad \text{.....(5)}$$

4.13.1.8 Measurement Conditions and Requirements of the Speed Class

4.13.1.8.1 Measurement Conditions

Table 4-61 shows measurement conditions for each Speed Class. The Speed Class 10 card shall support high speed mode. The higher Speed Class may require higher frequency or larger RU size. These values provide margin for host applications running at maximum speed.

Card Capacity		SDSC					SDHC
		~64MB	~256MB	~512MB	~1GB	~2GB	~32GB
AU Size (max.)		512KB	1MB	2MB	4MB		
RU Size	Class2, 4	16KB				32KB	
	Class6	64KB					
	Class 10	Not Supported					512KB

Notes: Class 2 to 6 are measured at 20MHz in Default Speed Mode
Class 10 is measured at 40MHz in High Speed Mode

Table 4-61 : Measurement Condition of Speed Class (SDSC and SDHC)

Application Note:
The minimum performance is measured at 100% usage rate of the SD Bus (No idle time is assumed in accesses). Therefore, writing to the card at intervals decreases performance.
Host needs to use higher frequency clock than that of measurement condition.

4.13.1.8.2 Requirements of the Performance Parameters for Each Speed Class

Table 4-62 identifies the requirement of the parameters for each class under measurement conditions. All conditions of any Class should apply simultaneously. Any cards having a specific Speed Class shall also satisfy the requirements and conditions of lower Classes. For example, Class 6 card shall satisfy Class 4 performance under Class 4 condition. Class 10 card shall satisfy Class 6 performance under the Class 6 conditions and Class 4 performance under the Class 4 conditions (Class 4 always covers Class 2 because of using the same conditions).

Regarding Class 10 Card, as Class 10 mode does not support Pm, the minimum requirement of Pm is more than or equal to 2MB/sec under the Class 4 conditions and 3MB/sec under the Class 6 conditions even if PERFORMANCE_MOVE in SD Status is set to 0.

	Pw min. [MB/sec]	Pm min. [MB/sec]	Pr min. [MB/sec]	T_{FW} (ave.) [ms]	T_{FW} (max.) [ms]	T_{FR} (4KB) max. [ms]
Class 2	2	1	2	100	750	12*1
Class 4	4	2	4	100	750	12*1
Class 6	6	3	6	100	750	12*1
Class 10	10	0*2	10	100	750	12*1

Note 1: T_{FR}(4KB) value is changed in Version 3.00

Note 2: Even Class 10 card, Pm may be used for Class 2 to 6 operations by host. In this case, if Pm=0, host should consider that Pm is half of Pw for Class 2 to 6. If host uses Class 10 mode, Pm indicated in SD Status shall be ignored and treated as 0.

Table 4-62 : Performance Requirements for Each Class (SDSC and SDHC)

4.13.1.8.3 Requirements of SD File System

This specification can be applied only to the SD file system formatted card defined by the File System Specification Version 3.00. This includes complying with the format parameter calculation specified in the Appendix C of the File System Specification Ver3.00.

Furthermore, the Number of Hidden Sectors shall be adopted as minimum number that meets Boundary Unit Recommendation for Data Area.

4.13.1.9 CMD20 Support

SDHC Memory Card may support Speed Class Control Command (CMD20) as optional. Setting SCR bit 32 to 1 indicates support of CMD20 but use of CMD20 is optional for host. Then even if a card supports CMD20, the card shall satisfy Speed Class specification without using CMD20. Refer to Section 4.13.2.8 about CMD20 functions definition.

4.13.2 Speed Class Specification for SDXC/SDUC

Speed Class is defined for SDXC and SDUC. Though the basic concept is similar to Speed Class for SDSC and SDHC, there are several differences. Key features of SDXC/SDUC Speed Class are listed below.

(1) The table of valid AU sizes (Table 4-47) is updated with six values larger than 4MB. When an AU size larger than 4MB is used, performance is measured in each of the 4MB sub-unit.
(2) RU sizes are larger and common for each card capacity range.
(3) Speed Class performance is defined only in the case of sequential writes to an entirely free AU. No move operation is supported.
(4) The FAT Update sequence is based on the exFAT file system for SDXC/SDUC.
(5) Sequence of updating CI (Continuous Information, defined in Part2 File System Specification Version 3.00) during stream recording is defined and supported.
(6) Speed Class Control command (CMD20) is introduced to optimize card operation for Speed Class recording.

4.13.2.1 Speed Class Parameters

4.13.2.1.1 AU

Capacities of up to 128TB and the UHS high speed interface require larger AU sizes.
In the case of SDXC and SDUC the maximum AU size is increased to 64MB.
To record the stream data, a Speed Class host shall manage the memory area in units of an AU and use only completely free AUs (zero fragmentation) to record the data.
Note that all AU sizes larger than 4MB are integer multiples of 4MB and performance is measured over each 4MB sub-unit of an AU.

4.13.2.1.2 RU

The definition of an RU is the same as in SDSC and SDHC. A Speed Class host shall write data in units of an RU.
The RU sizes are defined in Table 4-63. The same RU size is used for Class 2, 4 and 6. For Class 10, the RU size is larger to achieve higher performance. The same RU size is applied across the entire card capacity range (over 32GB~128TB) and varies only for each performance Class.

4.13.2.2 Write Performance

4.13.2.2.1 Measurement of Pw

Pw is measured in the case when data is written in units of RUs, from top to the bottom of an entirely free AU. There are two cases of how to measure Pw.

1. If the AU size is equal to or smaller than 4MB;
   Pw is defined as the minimum average write performance over an AU. This is the same definition as that for SDSC and SDHC.
2. If the AU size is larger than 4MB (Always a multiple of 4MB);
   Pw is defined as the minimum of the average write performance of every 4MB sub-unit in an AU.

Figure 4-52 shows the measurement of an AU's Pw. In the figure, performance of the measured AU is defined as the minimum of Pw1, Pw2,...PwN. The card's Pw is defined as the minimum performance of all AUs in the memory area.

Regardless of its size, an AU is the size of the memory area to which Speed Class host shall write data sequentially from top to the bottom in units of RU.

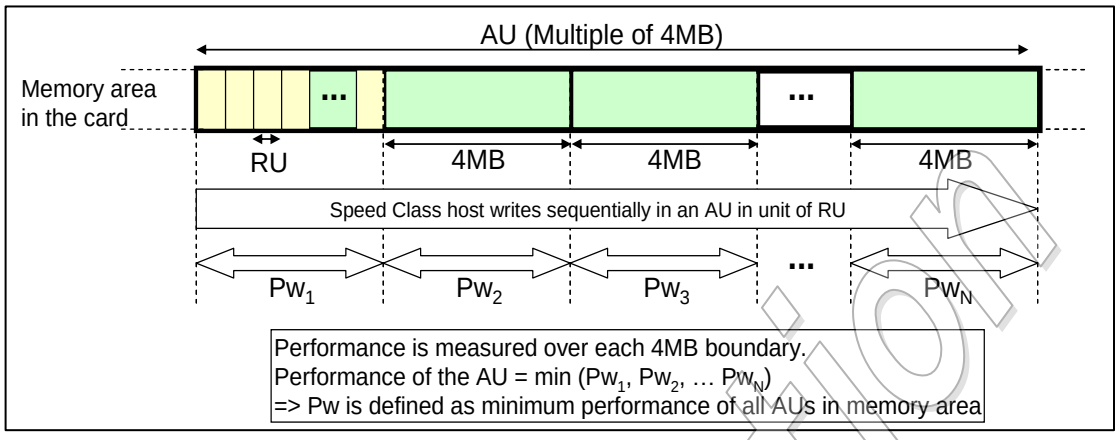


Figure 4-52 : Measurement of Pw (AU size is larger than 4MB)

4.13.2.2.2 Performance Move

In contrast with Speed Class for SDSC and SDHC, Pm is not supported in SDXC and SDUC. This means that Speed Class host shall write data to the entire free AUs so that SDXC and SDUC card can provide Class performance. Pm shall be set to 0 in SD Status of SDXC and SDUC card.

Application Note:
The amount of unwritten memory and the amount of Speed Class recordable memory may be different. Speed Class hosts can maximize the amount of recordable memory by defragmenting and freeing AUs for use when Speed Class performance is required.

4.13.2.3 Read Performance

The read performance of an SDXC and SDUC card has the same definition as that for SDSC and SDHC. Pr is defined as the minimum average random RU read performance over 256 RUs. Pr shall be greater than or equal to Pw.
T_{FR}(4KB) is defined as the maximum time to read 4KB data.

4.13.2.4 FAT Update

The SDXC and SDUC FAT update cycle has almost the same definition as in SDSC and SDHC. The only difference is the data type and its location. In the case of SDXC and SDUC, the FAT update cycle consists of three write operations, FAT, Bitmap and directory entry. The FAT is written starting at any 512-byte boundary address and with any size up to 16Kbytes. Bitmap is written starting at any 512-byte boundary address and with any size from 512-byte to 16Kbytes.
A directory entry should be created before starting recording and the same block should be written by 512-byte single block write (either CMD24 or CMD25).
Average and Maximum FAT Write Time (T_{FW}(ave.) and T_{FW}(max.)) are the same as defined for SDSC and SDHC. Average FAT Write time is defined as the maximum sliding average of 8 FAT update cycles.
Note that a Speed Class host can insert FAT update at any RU boundary. Insertion of FAT update never affects Pw.

4.13.2.5 CI (Continuous Information) Update

CI (Continuous Information) is a new structure used to manage exFAT file fragments. It is newly defined in the Part2 File System Specification Version 3.00. CI may be updated during Speed Class recording. Creating CI is optional for the host. The specific feature of CI update is described below.

- CI can be inserted at any RU boundary. The frequency of CI update depends on the host implementation.
  If a host tries to protect recorded data from any error including power failure, it may update CI frequently.
- A cluster is allocated to store CI and when it is filled, a new cluster is allocated. The first update of CI in a stream recording may be written from any point in an existing cluster. After the existing cluster is filled, a new cluster is allocated and all subsequent updates are written from the beginning to the end before a new cluster is allocated.
- The address to which CI is written may change during a stream recording. The same address may be overwritten several times. When the address is changed, it is always increased sequentially within the cluster. When a CI cluster is filled and a new one is allocated, the CI cluster address is changed randomly. (Any free cluster can be allocated).

The Speed Class specification for SDXC and SDUC defines the CI Update sequence and operation time. The sequence is similar to the directory entry update. CI is always written by a 512-byte single block write (either CMD24 or CMD25) and preceded by CMD20 Update CI command. The CI update operation time is maximum 250ms.
Note that Insertion of CI update never affects Pw.

Application Note: It is recommended that CI be updated after Speed Class recording is finished. In this case, the CI update is performed outside of Speed Class recording and the host does not need to consider it as overhead.

4.13.2.6 Distinction of Data Type

During Speed Class recording, several types of data (Stream data, FAT, Bitmap, directory entry and CI) are written by the host.
To satisfy Class performance, an SDXC and SDUC card needs to distinguish between each type of data in order to treat them properly.
For example, directory entry and CI can be distinguished by their data size (always written by 512B), so the card can store them in separate areas from the stream data.
Locations of the FAT and bitmap are described in Section 4.13.2.7.3.
Since directory entry and CI are written by 512-byte single block write (either CMD24 or CMD25), in user area, they should be distinguished by CMD20.

4.13.2.7 Measurement Conditions and Requirements of the Speed Class for SDXC/SDUC

4.13.2.7.1 Measurement Conditions

The measurement conditions of Speed Class for SDXC and SDUC are defined in Table 4-63. Class 10 card shall meet Class 6 performance under Class 6 condition.

Card Capacity		SDXC/SDUC
		Over 32GB ~ 128TB
AU Size (max.)		64MB
Unit of Performance Measurement		4MB
RU Size	Class2, 4, 6	256KB
	Class 10	512KB

Notes: Class 2 to 6 are measured at 20MHz in Default Speed Mode
Class 10 is measured at 40MHz in High Speed Mode

Table 4-63 : Measurement Conditions of Speed Class (SDXC/SDUC)

Host needs to use higher frequency clock than that of measurement condition.

4.13.2.7.2 Requirements of the Performance Parameters for Each Speed Class

Table 4-64 identifies the requirement of the parameters for each class under measurement conditions.

	Pw min. [MB/sec]	Pm min. [MB/sec]	Pr min. [MB/sec]	T_{FW} (ave.) [ms]	T_{FW} (max.) [ms]	T_{FR} (4KB) max. [ms]
Class 2	2	0	2	100	750	20
Class 4	4	0	4	100	750	20
Class 6	6	0	6	100	750	20
Class 10	10	0	10	100	750	20

Table 4-64 : Performance Requirements for Each Class (SDXC/SDUC)

4.13.2.7.3 Requirements of SD File System

This specification can be applied only to the SD file system formatted card defined by the latest version of File System Specification. This includes complying with the format parameter calculation specified in the Appendix C of the latest version of File System Specification.

Furthermore, the Number of Hidden Sectors shall be adopted as minimum number that meets Boundary Unit Recommendation for Data Area. And in case of exFAT file system, Allocation Bitmap shall be stored in the first 4MB of Cluster Heap.

4.13.2.8 Speed Class Control Command (CMD20)

CMD20 is defined to optimize card operation to support Speed Class recording. Figure 4-53 shows the definition of CMD20 timing and arguments. If any error occurs during the CMD20 busy period, it will be reported in the R1 response of next command. The host may issue CMD13 to check the occurrence of an error.

Speed Class Control (SCC) in the argument controls several functions which assist the card in supporting and meeting Class performance.

The response type of CMD20 is R1b. The maximum busy indication period depends on the function selected by SCC in the argument.

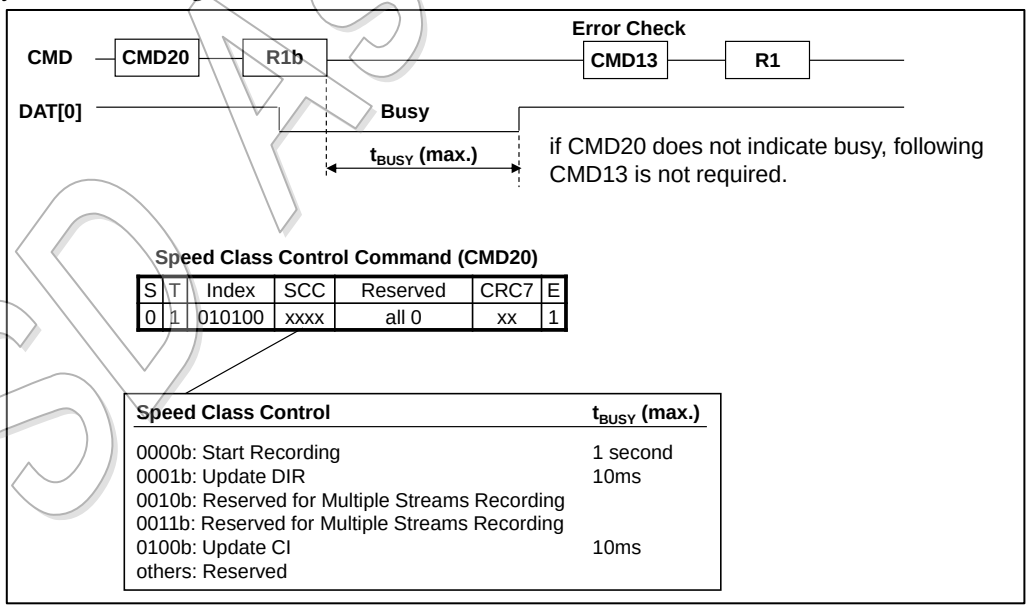


Figure 4-53 : Definition of CMD20

Support of CMD20 is indicated by CMD_SUPPORT of SCR[32].
If an SDXC/SDUC Card supports Speed Class, support of CMD20 is mandatory and host shall use CMD20. SDXC/SDUC Cards can meet Class performance only when host uses CMD20. (cf. In case of SDHC, support of CMD20 is optional for card and host may or may not use CMD20. SDHC cards meet Class performance without CMD20.)

4.13.2.8.1 Definition of Each CMD20 Function

(1) Start Recording

This function indicates that Speed Class host starts stream recording. When the card receives CMD20 Start Recording function, the card indicates busy up to 1 second to prepare recording (Garbage collection, clean-up of internal status, etc.).

(2) Update DIR

This function indicates that following write command shall be a directory entry write. On receiving CMD20 Update DIR function, the card shall recognize and manage the 512-byte area (DIR) specified by the following single block write command (either CMD24 or CMD25) as a location of the directory entry updated during the stream recording. The designated area (DIR) shall be written by single block writes and may be written repeatedly without preceding CMD20 Update DIR. Write to DIR (even without FAT write) counts in one of 8 times FAT write cycles. This function is always needed before Start Recording. If this function is issued during the recording, the card recognizes that the current recording is ended and the following write command is recognized as the directory entry for the next recording. After that, the host shall issue CMD20 Start Recording to start the next Speed Class recording. The busy indication of this function is up to 10ms for this function.

(3) Update CI

This function indicates that the following write command is a write to a CI cluster. When the card receives CMD20 Update CI function during the recording, the card recognizes that following 512-byte single block write (either CMD24 or CMD25) is an update to a CI cluster. Busy indication of CMD20 Update CI is up to 10ms for this function.

4.13.2.8.2 Requirements for Speed Class Host

There are requirements for host to support CMD20

- If host records data to SDXC or SDUC card, it shall support CMD20 to indicate Start Recording.
- The host shall issue CMD20 Update DIR before Start Recording regardless of updating CI during the recording.
- If host updates CI during the recording, it shall issue CMD20 Update CI just before the 512-byte single block write (either CMD24 or CMD25) updating CI.

4.13.2.9 Example of Speed Class Recording Sequence

Figure 4-54 shows example sequence of Speed Class recording.
Even if the write data size after CMD20 Update DIR is wrong (larger than 512-byte), the card accepts data writes without error but Speed Class performance is not maintained.

The host starts Speed Class recording by CMD20 indicating the Start Recording function and exits by either CMD20 indicating the Update DIR function or non-Speed Class write command. During the recording period, allowable write operations are limited to those shown below.

1. Stream data is written by one or more RU
2. FAT Update consists of three write commands, FAT($\leq$ 16KB)+Bitmap($\leq$ 16KB)+DIR(512-byte)
3. CI Update consists of single block write (512-byte by either CMD24 or CMD25 always preceded by CMD20 Update CI)

Even if the write data size after CMD20 Update CI is wrong (larger than 512-byte), the card accepts data writes without error but Speed Class performance is not maintained.

The SDXC or SDUC Speed Class host shall issue CMD20 Update DIR before CMD20 Start Recording so that the card can distinguish DIR and CI properly. Between Update DIR and Start Recording, data read/write/erase commands and secure commands are allowed. In addition, read commands of registers/status are also allowed.
If any other commands are issued, CMD20 Update DIR shall be issued again before CMD20 Start Recording.

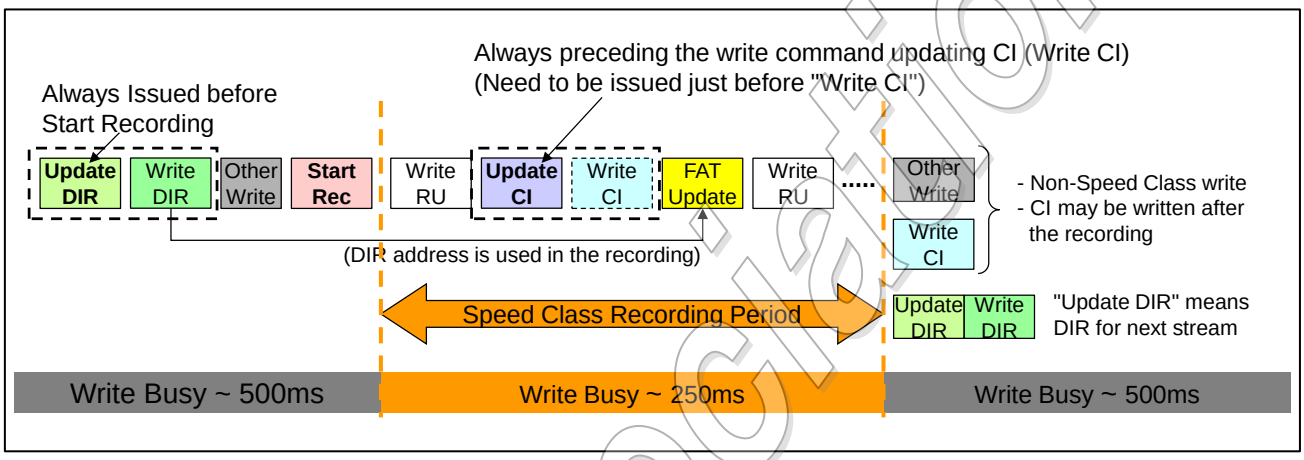


Figure 4-54 : Example of Speed Class Recording Sequence

4.13.3 Speed Grade Specification for UHS-I and UHS-II

4.13.3.1 Speed Grade Parameters

4.13.3.1.1 UHS Speed Grade

Following grades are defined and indicated by UHS_SPEED_GRADE in SD Status. Speed Grade may be supported in SDHC, SDXC and SDUC. Any cards having a specific Speed Grade shall also satisfy the requirements and conditions of lower Speed Grades. In case of UHS-II card supports Speed Grade 1, both UHS-I mode and UHS-II mode shall support Speed Grade 1. Similarly, in case of UHS-II card supports Speed Grade 3, both UHS-I mode and UHS-II mode shall support Speed Grade 1 and 3.

(1) Speed Grade 0: Performance is less than 10MB/sec
(2) Speed Grade 1: Performance is 10MB/sec and above
(3) Speed Grade 3: Performance is 30MB/sec and above

4.13.3.1.2 AU (Allocation Unit)

AU size of Speed Grade is specified by UHS_AU_SIZE in SD Status. The maximum AU size is 64MB. Card should indicate smaller AU Size as much as possible. Refer to Section 4.10.2 for more details.

4.13.3.1.3 RU (Recording Unit)

RU size is 512KB.

4.13.3.1.4 Pw (Write Performance)

The definition of Pw is the same as SDXC and SDUC. Refer to Section 4.13.2.2.

4.13.3.1.5 Pm (Performance Move)

Pm shall be treated as 0 regardless of the register value.

4.13.3.1.6 Pr and T_{FR}(4KB) (Read Performance)

The definition of Pr and T_{FR}(4KB) are the same as SDXC and SDUC. Refer to Section 4.13.2.3.

4.13.3.1.7 UHS-II Parameters

Host may use any N_FCU size in case of using Speed Grade 1 and more than or equal to N_FCU=4 in case of using Speed Grade 3. Use of larger N_FCU is recommended. Speed Grade is not supported in 2L-HD Mode or Low Power Mode.

4.13.3.2 Support of Speed Class Control Command (CMD20)

With regard to Speed Grade supported Host and Card, support of CMD20 is mandatory for SDXC and SDUC; and optional for SDHC. In order to maintain backward compatibility, Speed Grade SDHC cards shall support Speed Class and Speed Grade without the host indication of CMD20.

4.13.3.3 Speed Grade Measurement Conditions

Speed Grades Measurement Conditions are defined in Table 4-65.

Item	Definition
Pw Measurement Conditions for UHS-I	Performance is measured the same as Pw under following conditions. RU=512KB, Power Limit=1.44W (400mA at 3.6V) An AU Size is used described in UHS_AU_SIZE (Up to 64MB). [For Speed Grade 1] SDCLK=80MHz for SDR and 40MHz for DDR [For Speed Grade 3] SDCLK=90MHz for SDR and 45MHz for DDR The measurement method of UHS-I performance is the same as SDXC and SDUC. If AU size is larger than 4MB, performance is measured in each 4MB sub-unit.
Pw Measurement Conditions for UHS-II	Performance is measured the same as Pw under following conditions. RU=512KB, Power Limit=1.44W (400mA at 3.6V) An AU Size is used described in UHS_AU_SIZE (Up to 64MB). [For Speed Grade 1] RCLK=35MHz, Transmission Speed Range=Range A (x15 Multiplier) In FD Mode and Fast Mode. N_FCU=1. [For Speed Grade 3] RCLK=26MHz, Transmission Speed Range=Range B (x30 Multiplier) In FD Mode and Fast Mode. N_FCU=4. The measurement method of UHS-II performance is the same as SDXC and SDUC. Performance is measured in each 4MB sub-unit.
FAT Update / CI Update	Same as existing Speed Class
Write busy	Each write busy shall be less than or equal to 250ms Except DIR write of SDXC and SDUC card may be up to 500ms

Table 4-65 : Speed Grade Measurement Conditions for UHS-I and UHS-II

4.13.3.4 Notes for Preparation Time of UHS-I and UHS-II Card

At the beginning of a recording, the SD Memory Card indicates busy as preparation time to start Speed Class Recording. The maximum preparation time is considered as 1 second. If CMD20 is received, the card shall indicate the preparation time in busy of CMD20 Start Recording. If CMD20 is not received, the card should indicate the preparation time in the busy of write operations, which writes to the first AU. In this case, the card is not necessary to include preparation time of the first AU in Pw and host should estimate 1 second preparation time to write the first AU.

4.13.3.5 Host Operating Frequency

In case of Non UHS-II mode, Host needs to use higher frequency clock than that of measurement condition.

In case of Speed Grade 1 in UHS-II mode, as Speed Grade of card is measured at 35MHz Range A, host needs to set higher bit rate than measurement bit rate. That is when UHS-II is used in Transmission Speed Range A (x15 Multiplier), host needs to provide RCLK more than or equal to 35MHz. When UHS-II card is used in Transmission Speed Range B (x30 Multiplier), any frequency of RCLK range (26MHz - 52MHz) may be used.

In case of Speed Grade 3 in UHS-II mode, as Speed Grade of card is measured at 26MHz Range B, host needs to set higher bit rate than measurement bit rate. That is when UHS-II card is used in Transmission Speed Range B (x30 Multiplier), any frequency of RCLK range (26MHz - 52MHz) may be used. Speed Grade 3 is not supported in Range A (x15 Multiplier).

4.13.4 Video Speed Class Specification

4.13.4.1 AU Use in Video Speed Class

4.13.4.1.1 Video Speed Class Introduction

In Video Speed Class, the AU (see Section 4.13.4.2.2) is used differently than in the other speed class methods. In Video Speed Class recording, the following steps occur:

1. The host issues one or more (up to eight) CMD20 "Update DIR" functions (see Section 4.13.4.7.3) to set up directory entries.
2. The host selects an LBA space on AU boundaries that contains no data to be retained or prepares the space by moving the data to be retained to free the AU (see Section 4.13.1.1). If there is data in this LBA space to be retained, then the host is responsible to relocate this data to other logical areas in the card.
3. The host tells the card to use the selected LBA space by specifying one to eight AUs using the CMD20 "Set Free AU" command (see Section 4.13.4.7.7), which informs the card that the contents of the specified AUs have been released by the host and those AUs may be prepared for efficient sequential writing.
4. The host issues the CMD20 "Start Recording" command (see Section 4.13.4.7.2).
5. The host begins issuing sequential write operations from the lowest LBA in the first active AU in units of RU. The host uses assigned AUs sequentially from the lowest to the highest. If the host writes to non-sequential locations within the assigned AUs, the card may drop out of Video Speed Class recording without warning (i.e., subsequent writes will have non-speed class timing).
6. If the assigned AUs have been completely written, then:
   a) the host may continue Video Speed Class recording by issuing another CMD20 "Set Free AU" command and issuing sequential write operations from the lowest LBA in the first newly-active AU;
   b) issuing CMD20 "Start Recording" is optional after an AU boundary if the previous AU was completely written (i.e., the last RU in this AU was written); and
   c) issuing CMD20 "Start Recording" is required after an AU boundary if the previous AU was not completely written (i.e., the last RU in this AU was not written).

If the card receives CMD20 "Set Free AU" command before completing an active AU (e.g., step 6c) above), then the data stored in the unwritten LBAs in those AUs is undefined.

Use of Speed Class (see Section 4.13.1 and 4.13.2) or UHS Speed Grade (see Section 4.13.3) during Video Speed Class recording results indeterminate card behavior.

As described in step 3 above, AUs become assigned through a CMD20 "Set Free AU" command.
As described in step 6 above, the active assigned AU is released by sequentially writing all its LBAs from its lowest LBA to its highest LBA. If another AU remains assigned, Video Speed Class recording may continue in that AU.

A CMD20 "Suspend AU" command releases all assigned AUs (i.e., only the active AU is suspended) and terminates Video Speed Class recording (i.e., Pw (see Section 4.13.4.3.1) is not maintained). Video Speed Class recording may be resumed by a CMD20 "Resume AU" command (see Section 4.13.4.1.2).

If any violating condition from Table 4-66 occurs while there are assigned AUs:

a) all assigned AUs shall be released;
b) Video Speed Class recording shall be terminated (i.e., Pw (see Section 4.13.4.3.1) is not maintained); and
c) without a suspend address, Video Speed Class recording is not resumable.

Commands	Violating Conditions
CMD0-CMD11	Any CMD0 through CMD11 terminates the Video Speed Class recording.
CMD14-CMD16	Any CMD14 through CMD16 terminates the Video Speed Class recording.
CMD22	Any CMD22 for SDHC and SDXC cards terminates the Video Speed Class recording. CMD22 for SDUC cards is required and it is included in sequential write performance calculation of Video Speed Class recording after Start Recording
CMD24	If CMD24 addresses any non-sequential LBA in an active AU, then the Video Speed Class recording is terminated. Any other CMD24 in an active AU breaks SU timing.
CMD25	If CMD25 addresses any non-sequential LBA in an active AU, then the Video Speed Class recording is terminated. Any CMD25 that is not multiple RU size in an active AU breaks SU timing. Any CMD25 that does not start on an RU boundary in an active AU breaks SU timing.
CMD26-CMD31	Any CMD26 through CMD31 terminates the Video Speed Class recording.
CMD34-37	If any CMD34 through CMD37 addresses any non-sequential LBA in an active AU, then the Video Speed Class recording is terminated. Any CMD34 through CMD37 that is not multiple RU size in an active AU breaks SU timing. Any CMD34 through CMD37 that does not start on an RU boundary in an active AU breaks SU timing.
CMD38	If an LBA area designated by CMD32 and CMD33 includes a part of an active AU, CMD38 terminates the Video Speed Class recording.
CMD39	Any CMD39 terminates the Video Speed Class recording.
CMD41-CMD47	Any CMD41 through CMD47 terminates the Video Speed Class recording.
CMD50	If CMD50 addresses any non-sequential LBA in an active AU, then the Video Speed Class recording is terminated. Any CMD50 that is not multiple RU size in an active AU breaks SU timing. Any CMD50 that does not start on an RU boundary in an active AU breaks SU timing.
CMD51	Any CMD51 terminates the Video Speed Class recording.
CMD56	Any CMD56 terminates the Video Speed Class recording.
CMD57	If CMD57 addresses any non-sequential LBA in an active AU, then the Video Speed Class recording is terminated. Any CMD57 that is not multiple RU size in an active AU breaks SU timing. Any CMD57 that does not start on an RU boundary in an active AU breaks SU timing.
CMD60-63	Any CMD60 through CMD63 terminates the Video Speed Class recording.
ACMD1-ACMD12	Any ACMD1 through ACMD12 terminates the Video Speed Class recording.
ACMD17	Any ACMD17 terminates the Video Speed Class recording.
ACMD19-ACMD21	Any ACMD19 through ACMD21 terminates the Video Speed Class recording.
ACMD24	Any ACMD24 terminates the Video Speed Class recording.
ACMD27	Any ACMD27 terminates the Video Speed Class recording.
ACMD29-ACMD37	Any ACMD29 through ACMD37 terminates the Video Speed Class recording.
ACMD39-42	Any ACMD39 through ACMD42 terminates the Video Speed Class recording.
ACMD50	Any ACMD50 terminates the Video Speed Class recording.
ACMD52-ACMD54	Any ACMD52 through ACMD54 terminates the Video Speed Class recording.
ACMD56-ACMD63	Any ACMD56 through ACMD63 terminates the Video Speed Class recording.

Table 4-66 : Video Speed Class Violating Conditions

4.13.4.1.2 Suspend and Resume in Video Speed Class

In Video Speed Class it is possible to continue using an AU across a power cycle. This involves a save of the current write state of an AU, an optional power cycle the card, and then a resume of the Video Speed Class write operations.

To save the current write location in the active AU the host suspends the AU by issuing the CMD20 "Suspend AU" command (see Section 4.13.4.7.5).

While the card is performing Video Speed Class recording, the results of processing a CMD20 "Suspend AU" command between RU writes are:

a) all assigned DIR slots shall be released;
b) unused assigned AUs shall be released;
c) if the lowest address of the subsequent RU is in an active AU, then the SUS_ADDR field shall be set to the lowest address of the subsequent RU; and
d) if the lowest address of the subsequent RU is not in an active AU, then the SUS_ADDR field shall be set to zero.

While the card is performing Video Speed Class recording, the current RU is partially filled and a CMD20 "Suspended AU" command is issued, then:

a) all assigned DIR slots shall be released;
b) all active AUs shall be released; and
c) the SUS_ADDR field shall be set to zero.

If the card doesn't have active AU (e.g., a CMD20 "Set Free AU" command has not been issued) and the card receives a CMD20 "Suspend AU", then the SUS_ADDR field shall not be changed.

If the SUS_ADDR field is non-zero, the process to resume the Video Speed Class write operations to a suspended AU in a card is:

1) the host shall issue one or more CMD20 "Update DIR" commands;
2) the host shall issue the CMD20 "Resume AU" command;
3) the host shall issue the CMD20 "Start Recording" command; and
4) the host may continue write operations starting at the suspend address.

If the SUS_ADDR field is zero, then the card is not configured to resume and the host should select a new free AU to start Video Speed Class recording (see Section 4.13.4.1.1).

While the AU is suspended, any write to the suspended AU will result in the SUS_ADDR field being cleared to zero (see Section 4.10.2.12).

4.13.4.1.3 Multiple File Recording in Video Speed Class Introduction

In Video Speed Class, an AU may contain multiple files. It is possible to concatenate multiple files in an AU, or to interleave write operations of different files within an AU. Video Speed Class supports up to eight directory locations to be opened at the same time. Each directory location is assigned using the CMD20 "Update DIR" command (with CNT/ID "slot ID") (see Section 4.13.4.7.3) and released for reassignment using the CMD20 "Release DIR" command (with CNT/ID "slot ID") (see Section 4.13.4.7.8).

Figure 4-55 is a video recording with multiple captures of still photo. In this example the host will assign a CNT/ID "slot ID" for the video file and another CNT/ID "slot ID" to be used for each of the capture photos. In this example:

1. the host configures the directory entries for the first two files;
2. begins writing file #1;
3. writes file #2;

4. writes additional data for file #1;
5. reconfigures CNT/ID "slot ID" for file #3;
6. writes file #3;
7. writes additional data for file #1;
8. reconfigures CNT/ID "slot ID" for file #4;
9. writes file #4; and
10. fills the remaining blocks in the AU with data for file #1.

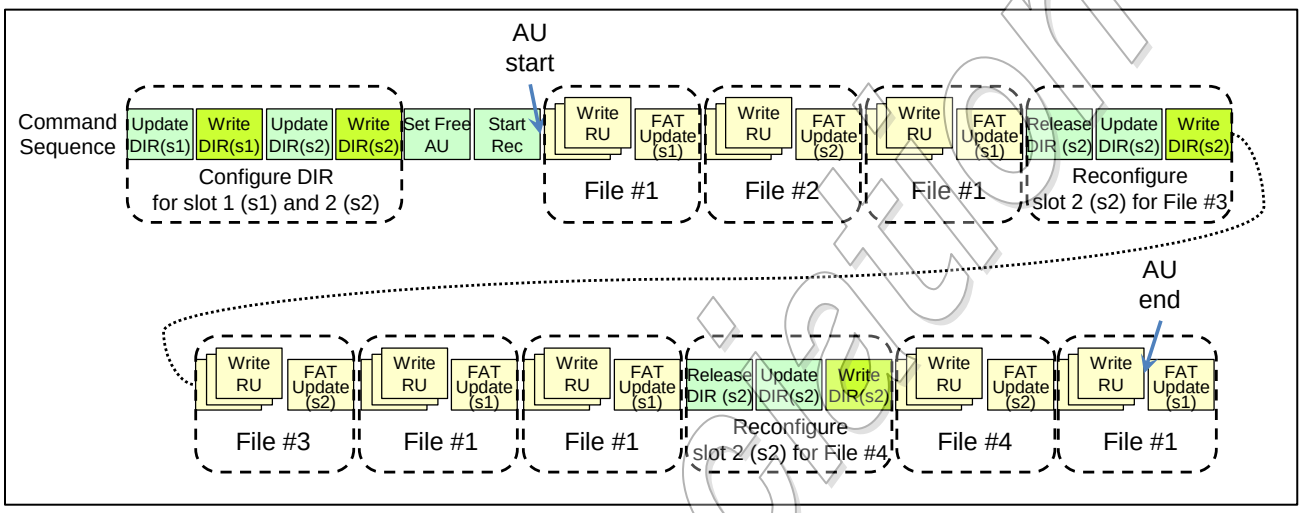


Figure 4-55 : Example of Interleaved Multiple File Write in One AU

The example shown in Figure 4-56 includes the concatenated write of two files. In this example the host will assign a CNT/ID "slot ID" for the first file and reassign the CNT/ID "slot ID" for the second file. In this example:

1. the host configures the directory entries for file #1;
2. completes writing file #1;
3. reconfigures CNT/ID "slot ID" for file #2; and
4. completes writing file #2.

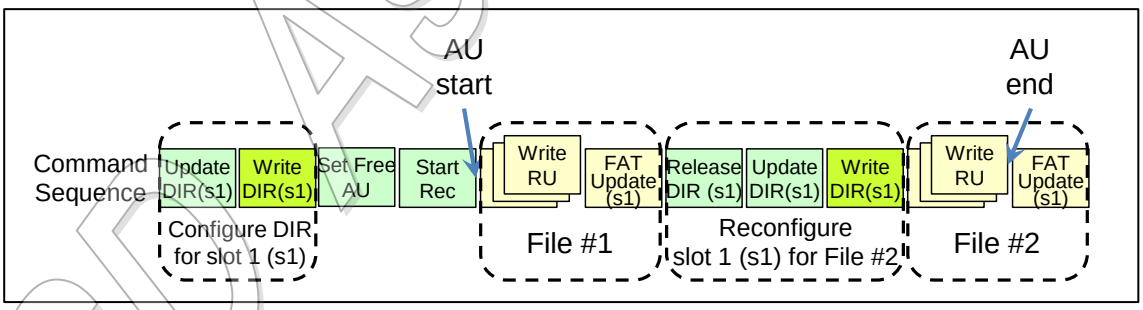


Figure 4-56 : Example of Concatenated File Write in One AU

4.13.4.1.4 Access Rules for Multiple File Recording

When a host writes multiple files in one AU, the following access rules apply:

a) file data of multiple files are recorded sequentially in one AU by the multiple size of RU, where partial RUs are to be padded by the host (the preferred pad value is zero); and
b) if the DIR address or the CI address of the next file differs from the address of the previous file, then the new address may be assigned by using the CMD20 "Release DIR", the CMD20 "Update DIR" or the CMD20 "Update CI" during the recording period.

Examples of these rules are shown in Figure 4-57.

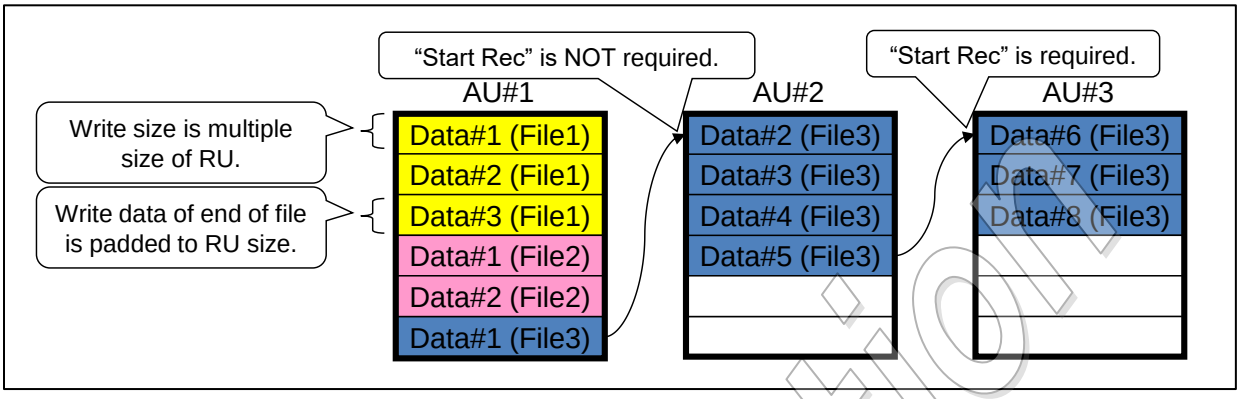


Figure 4-57 : Access Rules for Multiple Files Recording

4.13.4.1.5 CMD25 Boundary Crossovers in Video Speed Class

Cards shall maintain Pw (see Section 4.13.4.3.1) for CMD25, WRITE_MULTIPLE_BLOCK commands that cross SU (see Section 4.13.4.2.4) boundaries within the active AUs.

Cards shall maintain Pw for CMD25, WRITE_MULTIPLE_BLOCK commands that cross AU boundaries within the assigned AUs.

Cards may not maintain Pw (see Section 4.13.4.3.1) for the portion of CMD25, WRITE_MULTIPLE_BLOCK commands that cross the end of the active AUs.

4.13.4.1.6 Read Response and Data in Active AUs

When the card processes a CMD20 "Set Free AU" command, AUs to be selected become active (see Section 4.13.4.7.7). When the CMD20 "Set Free AU" command received, the host has released the content of active AUs. In Video Speed Class recording, these active AUs are filled sequentially.

If the host reads LBAs in the active AUs that have been sequentially written, then the data that was written should be returned.

If the host reads LBAs in the active AUs that have not been written yet, then the data to be returned changes during recording process. The data to be returned is undefined.

4.13.4.2 Video Speed Class Parameters

4.13.4.2.1 Video Speed Class

The Video Speed Class of the card is specified by VIDEO_SPEED_CLASS (see Section 4.10.2.10) in SD Status.

4.13.4.2.2 AU (Allocation Unit)

The AU size for Video Speed Class is specified by VSC_AU_SIZE (see Section 4.10.2.11) in SD Status. The maximum AU size is 512MB. Card should indicate the smallest valid AU size that meets the card design target.

4.13.4.2.3 RU (Recording Unit)

The RU size for Video Speed Class is 512KB.

4.13.4.2.4 SU (Sub Unit)

An AU is composed of multiple SU's. Pw measurement (see Section 4.13.4.3.1) is collected across each SU. The SU size for Video Speed Class is defined by the card's AU size (see Section 4.10.2.11).

4.13.4.2.5 N_FCU

The configured N_FCU is reported in the LINK/TRAN Settings Register and the card's maximum N_FCU is reported in the Device-specific N_FCU in the LINK/TRAN Capabilities Register. Both registers are described in the SD Part1 Physical Layer Specification UHS-II Addendum.

The minimum configured N_FCU required for Video Speed Class is defined in the Video Speed Class Measurement Conditions (see Section 4.13.4.8.2).

4.13.4.3 Video Speed Class Timing

4.13.4.3.1 Pw (Write Performance)

The sum of the time for the sequential write operations needed to write a SU (see Section 4.13.4.2.4) defines the performance measurement period of that SU. The performance measurement period of the first SU in the first AU preceded by a CMD20 "Set Free AU" command includes the busy time of the CMD20 "Set Free AU" command that is issued after CMD20 "Start Recording". See Figure 4-58. If a RU write is halted (e.g., CMD12) on an address that does not complete the RU, then the Pw measurement over the affected SU (i.e. the SU which contains the interrupted RU) is not valid. If the AU writing remains sequential, the timing of next SU is valid.

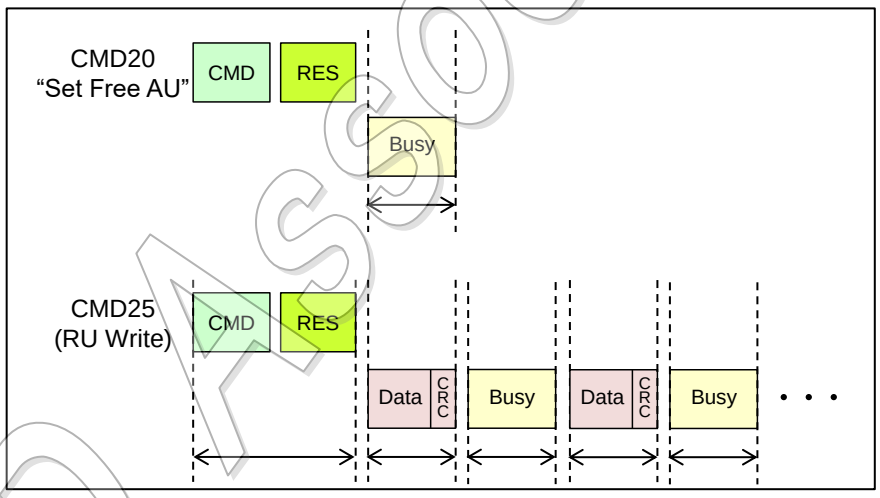


Figure 4-58 : Time Components for Video Speed Class Performance Write Measurement

The time needed to complete any other commands including write commands outside of the SU is not included in the time measurement period of any SU (e.g., CMD20 "Start Recording", CMD20 "Update DIR" operations, CMD20 "Release DIR", CMD20 "Update CI" operations, CMD20 "Suspend AU", CMD20 "Resume AU", CMD17, CMD18, CMD23).

The measured performance write time period (Pw time) of a card is the largest performance time measurement period of each of the SUs that are included in all AUs except the AUs in the card that contain file system information. The measured Pw of SU is the SU size in MB (see Section 4.13.4.2.4) divided by Pw_time in seconds. The value reported by VIDEO_SPEED_CLASS (see Section 4.10.2.10) is equal or less than the measured Pw.

The card shall satisfy the above measured Pw regardless of the frequency of FAT Update cycles during Video Speed Class write. The minimum FAT Update frequency is zero. The card stops performing Video Speed Class recording, if receives any of the commands in Table 4-66.

Figure 4-59 shows the measurement of an AU's Pw.

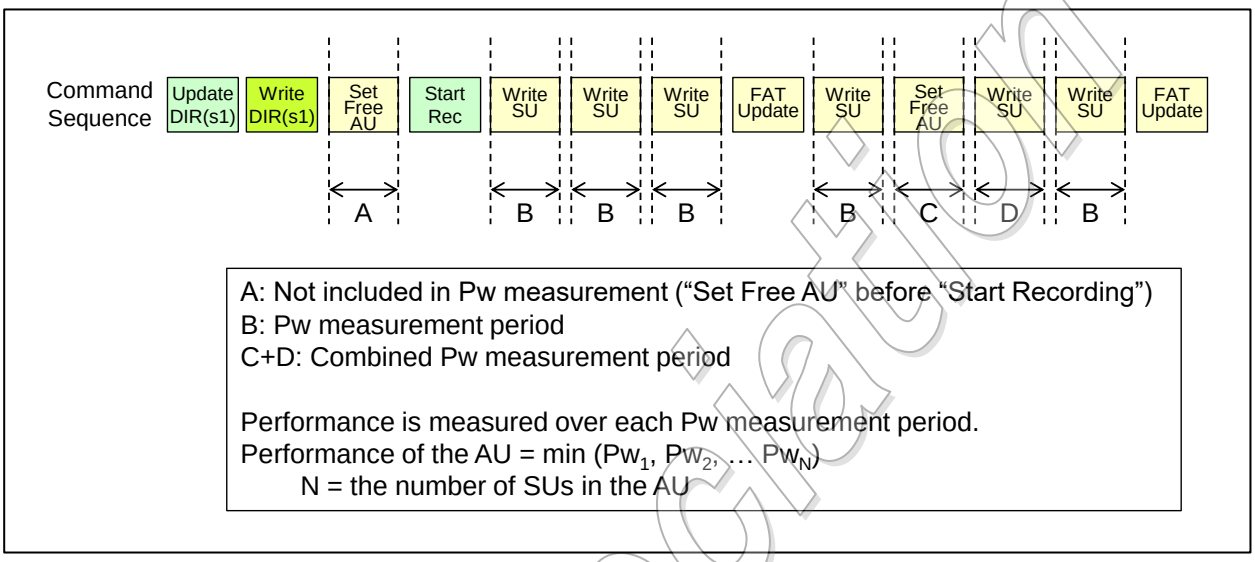


Figure 4-59 : Measurement of Pw

4.13.4.3.2 Pm (Performance Move)

Pm doesn't apply to Video Speed Class.

4.13.4.3.3 Pr and T_{FR}(4KB) (Read Performance)

The read performance period of an SU is the sum of the time used for the sequential read operations (see Figure 4-60) that are needed to read the entire SU (see Section 4.13.4.2.4), given:

a) the SU is aligned within an AU (see Section 4.13.4.2.2);
b) the SU is in an AU that does not contain file system information as defined in: SD Specifications, Part 2, File System Specification;
c) the sequential read commands have a data size equal to an RU (see Section 4.13.4.2.3); and
d) the sequential read commands are not interleaved with any other commands.

The Pr of a given SU is the SU's size in MB divided by read performance period of that SU in seconds.

The Pr of the card is the smallest Pr for all the SUs in the card that are in AUs that do not contain file system information as defined in: SD Specifications, Part 2, File System Specification.

If Video Speed Class recording is supported, then the card's Pr shall be greater than, or equal to the card's Pw (see Section 4.13.4.3.1).

T_{FR}(4KB) is defined as the maximum time to read a 4KB FAT and Directory Entry (see Figure 4-60), given:

a) the read command addresses a 4KB extent that contains file system information as defined in: SD Specifications, Part 2, File System Specification; and
b) the read command is a single read command with a data size of 4KB.

The T_{FR}(4KB) of the card is the largest all T_{FR}(4KB) reads.

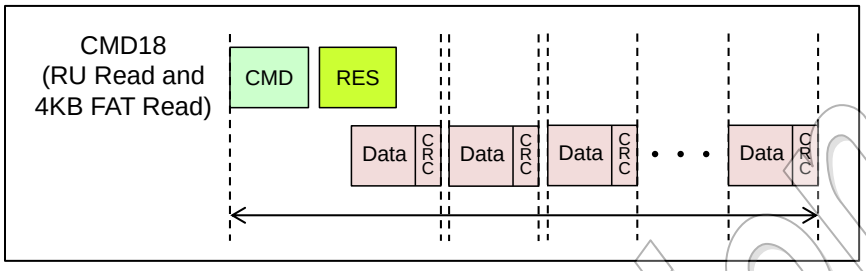


Figure 4-60 : Read Performance Period for Pr and T_{FR}(4KB)

T_{FR}(4KB) of the card shall be less than or equal to 20ms.

4.13.4.3.4 Busy Times for CMD20 Operations

The busy time for CMD20 operations and whether the busy times are included in Pw measurement is defined in Table 4-67.

No.	CMD20 Function	tBUSY	Applies to VSC Pw
1	Start Recording	1 second	No
2	Update DIR	10ms	No
3	Associated Write Command to (2)	250ms (SDHC) / 500ms (SDXC and SDUC)	No
4	Update CI	10ms	No
5	Associated Write Command to (4)	250ms	No
6	Suspend AU	250ms	No
7	Resume AU	250ms	No
8	Set Free AU	250ms	Yes
9	Release DIR	50ms	No

Table 4-67 : Busy Time for CMD20 Operations

4.13.4.4 SD Interface Mode Requirements for Video Speed Classes

The requirements for lower Video Speed Class support (see Section 4.10.2.10) and the SD interface mode requirements (see Section 3.10.5) interact. Two examples of interacting requirements on a card supports Video Speed Class 90 are:

a) the card supports Video Speed Class 6 and its required SD interface mode (e.g., High Speed mode); and
b) the card supports Video Speed Class on all SD interface modes that the card supports (e.g., if UHS-I SDR104 is supported, then all supported Video Speed Classes are supported on UHS-I SDR104).

Table 4-68 defines the SD interface modes that shall be supported for each Video Speed Class. This table also defines the SD interface modes that shall be supported if the combination of a Video Speed Class and a SD interface mode is supported.

Note: DS and SDR12 interface modes are not applicable to Video Speed Class specification.

Video Speed Class	HS	UHS-I				UHS-II	
		SDR25	SDR50	DDR50	SDR104	FD	HD
VSC6	M	MiS	MiS	MiS	MiS	MiS	MiS
VSC10	M	MiS	MiS	MiS	MiS	MiS	MiS
VSC30			M	MiS	MiS	MiS	MiS
VSC60						M	MiS
VSC90						M	MiS

"M": Video Speed Class support is mandatory.
"MiS": Video Speed Class support is mandatory if the SD interface mode is supported.

Table 4-68 : SD Interface Mode Requirements for Video Speed Classes

4.13.4.5 Requirements of SD File System for Video Speed Class

See Section 4.13.1.8.3 and 4.13.2.7.3.

4.13.4.6 FAT Updates in Video Speed Class

4.13.4.6.1 Multiple Files in an AU

The FAT update cycle is defined in Section 4.13.1.7. In Video Speed Class, up to eight directory entries (see Section 4.13.4.1.3) may be supported to allow simultaneous access of up to eight files, resulting in up to eight FAT update cycles between any RU accesses. The Average FAT Write Time and Maximum FAT Write Time for Video Speed Class Specification are described below.

4.13.4.6.2 Average FAT Write Time

Equation (6) defines Average FAT Write Time (T_{FW}(ave.)) of Video Speed Class for Single Slot mode, which has compatibility to that of Speed Class. Equation (7) defines Average FAT Write Time (T_{FW}(ave.)) of Video Speed Class for Multi-Slot mode.

Average FAT Write Time:

$$T_{FW}(ave.) = \frac{\sum_{i=1}^{8} T_{FW}(i)}{8} \le 100ms(\max.) \quad \text{for Slot 1 only} \quad \text{.....(6)}$$

$$T_{FW}(ave.) = \frac{\sum_{i=1}^{8} T_{FW}(i)}{8} \le 150ms(\max.) \quad \text{for multiple slots} \quad \text{.....(7)}$$

Single Slot is default mode. Card keeps Single Slot mode or switches to Multi-Slot mode depends on assignment of DIR Slots on receiving CMD20 "Start of Recording" command:

(1) when no DIR Slot has been assigned, card is in Single Slot mode (default)
If any of DIR Slots is assigned, another "Start Recording" command is required

(2) when only DIR Slot 1 has been assigned, card is in Single Slot mode
DIR Slot 1 can be re-assigned after "Start Recording" command
If any of DIR Slot 2 to Slot 8 is assigned, another "Start Recording" command changes to Multi-Slot mode

(3) when card has been assigned any of DIR Slot 2 to Slot 8, card is in Multi-Slot mode
Any of DIR Slots can be re-assigned after "Start Recording" command

Host needs to prepare buffer to save streaming data while FAT update is performed. For example, if

host budgets time for 8 FAT updates cycles, 800ms (=8x100ms) host buffer is required for single slot mode and 1.2seconds (=8x150ms) host buffer is required for Multi-Slot mode regardless of the number of slots.

4.13.4.6.3 Maximum FAT Write Time

Equation (8) limits the Maximum FAT Write Time (T_{FW} (max.)) of Video Speed Class Specification.

Maximum FAT Write Time: $T_{FW}(\max.) \leq 750ms$ .....(8)

At most one T_{FW}(max.) may be indicated during any 8 FAT update cycles to satisfy sliding average requirement by managing DIR slots.

4.13.4.6.4 Example Sequence of FAT Cycle

T_{FW}(ave.) is calculated as the sliding average of 8 FAT updates cycles among all DIR slots (see Section 4.13.4.6.2). Host may perform any slot combination of FAT Update.

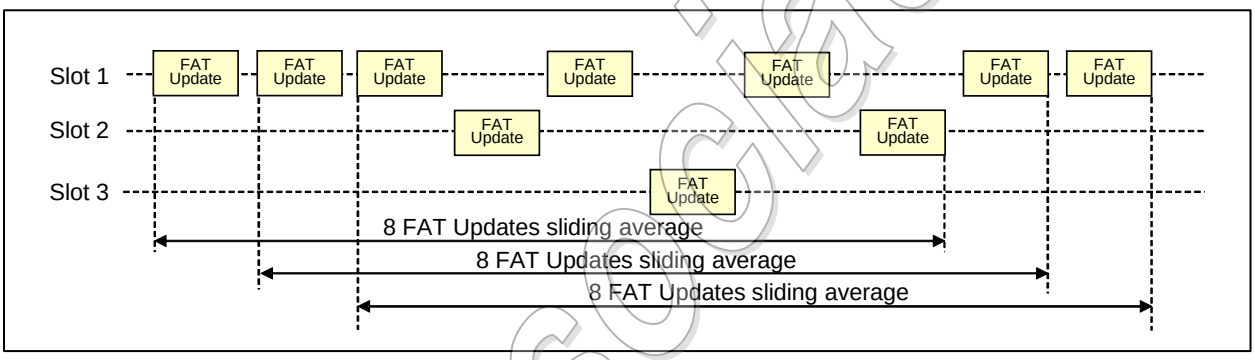


Figure 4-61 : Example Sequence of FAT Cycle

4.13.4.7 CMD20 in Video Speed Class

4.13.4.7.1 CMD20 in Video Speed Class Overview

CMD20 for Video Speed Class has several functions that were not defined in earlier speed classes. If a card supports Video Speed Class, support of CMD20 is mandatory regardless of capacity. Host shall use CMD20 to perform Video Speed Class recording.

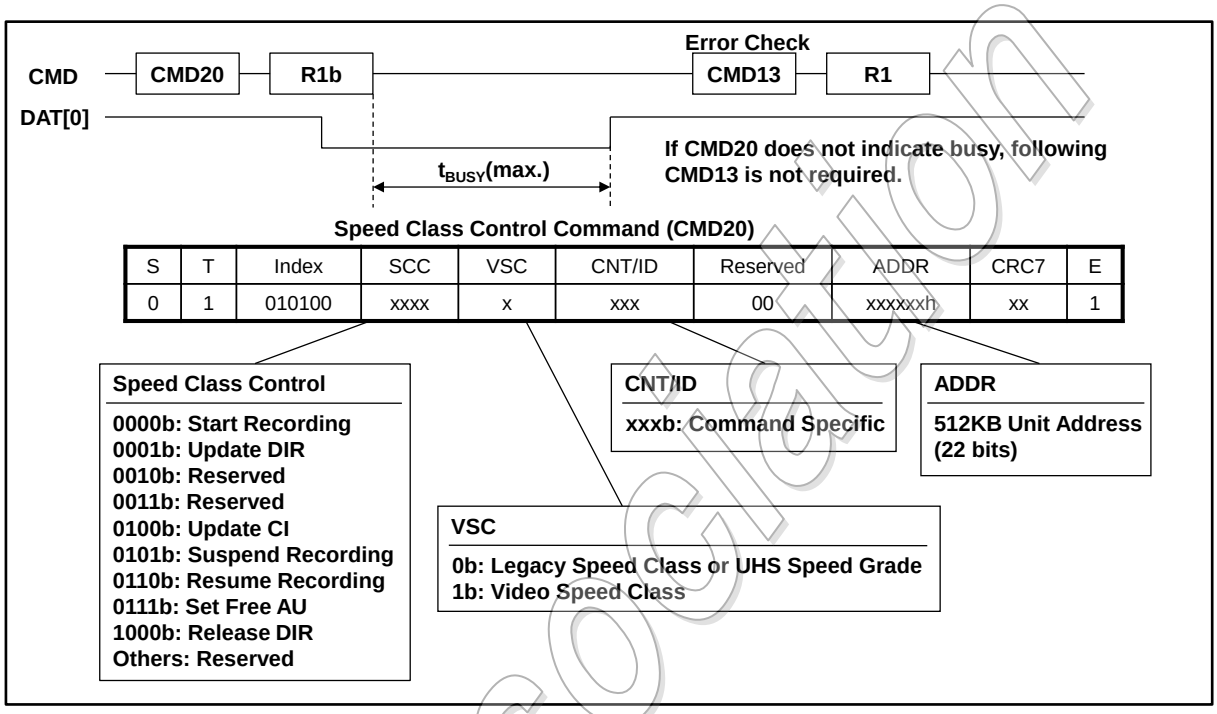


Figure 4-62 : Definition of CMD20 (Video Speed Class)

4.13.4.7.2 Start Recording

4.13.4.7.2.1 Start Recording Description

A value of 0000b in the SCC field (see Section 4.13.4.7.1) of CMD20 is "Start Recording". Video Speed Class uses this CMD20 function differently than Speed Class and UHS Speed Grade.

In Video Speed Class, this function indicates that host intends to starts stream recording (see Section 4.13.4.2.1). This function requests the card to prepare recording (Garbage collection, clean-up of internal status, etc.).

4.13.4.7.2.2 Start Recording Parameters

For CMD20 "Start Recording":

a) VSC is set to 1b;
b) CNT/ID is reserved; and
c) ADDR is reserved.

4.13.4.7.2.3 Start Recording Busy Timing

The maximum tBUSY for CMD20 "Start Recording" is 1 second (see Section 4.13.4.3.4).

4.13.4.7.2.4 Start Recording Error Conditions

CMD20 "Start Recording" has no error conditions.

4.13.4.7.3 Update DIR

4.13.4.7.3.1 Update DIR Description

A value of 0001b in the SCC field (see Section 4.13.4.7.1) of CMD20 is "Update DIR". Video Speed Class has added parameters and functionality to this CMD20 function that are not used in Speed Class and UHS Speed Grade.

This function indicates that following write command shall be a directory entry write. On receiving CMD20 "Update DIR", the card shall recognize and manage the 512-byte area (DIR) specified by the Slot ID in the CNT/ID field. The address of the following single block write command (either CMD24 or CMD25) is the location of the directory entry specified by the Slot ID in the CNT/ID field. The designated area (DIR) shall be written by single block writes and may be written repeatedly without preceding CMD20 "Update DIR". Subsequent write commands to this location apply to FAT update timings as defined in Section 4.13.4.6.

Cards shall support eight concurrently active Slot IDs.

The assignment of any directory entry write areas in the card shall not persist over power cycles or after processing a CMD20 "Suspend AU" command.

This function may be issued before Start Recording and during Video Speed Class writing.

Application Note:
(1) VSC application should use the minimum number of Directory Entries as much as possible so that 8 DIR slots can be efficiently utilized.
(2) VSC application should consider that DIR locations are not distributed. If all DIRs belong to an AU, release of DIRs will generate a write back to the AU. However, if DIRs are distributed to multiple AUs, release of DIRs will generate inefficient write backs to the all AUs in which DIRs have been assigned.

4.13.4.7.3.2 Update DIR Parameters

For CMD20 "Update DIR":

a) VSC is set to 1b;
b) CNT/ID shall contain the slot ID to be assigned (e.g., 000b is the first slot); and
c) ADDR is reserved.

4.13.4.7.3.3 Update DIR Busy Timing

The maximum tBUSY for CMD20 "Update DIR" is 10ms (see Section 4.13.4.3.4).

The maximum tBUSY for the write command associated with CMD20 "Update DIR" is 250ms (SDHC) / 500ms (SDXC/SDUC) (see Section 4.13.4.3.4).

4.13.4.7.3.4 Update DIR Error Conditions

If the card receives CMD20 "Update DIR" to a slot ID that has been assigned and not released (CMD20 "Release DIR"), then the card indicates ERROR in R1 (ERX).

If the card does not receive one or more CMD20 "Update DIR" in the sequence before CMD20 "Start Recording", then the card may not enter Video Speed Class recording without warning.

And if host changes DIR address that has already been stored in a specific slot to other DIR address without power cycle, host shall release the registration by issuing CMD20 "Release DIR" and then register other DIR address by issuing CMD20 "Update DIR" with the same "SLOT ID".

4.13.4.7.4 Update CI

4.13.4.7.4.1 Update CI Description

A value of 0100b in the SCC field (see Section 4.13.4.7.1) of CMD20 is "Update CI". The use of this function in Video Speed Class is described in Section 4.13.4.1.1.

This function indicates that the following write command is a write to a CI cluster. When the card receives CMD20 Update CI function during the recording, the card recognizes that following 512-byte single block write (either CMD24 or CMD25) is an update to a CI cluster.

4.13.4.7.4.2 Update CI Parameters

For CMD20 "Update DIR":

a) VSC is set to 1b;
b) CNT/ID is reserved; and
c) ADDR is reserved.

4.13.4.7.4.3 Update CI Busy Timing

The maximum tBUSY for CMD20 "Update CI" is 10ms (see Section 4.13.4.3.4).

The maximum tBUSY for the write command associated with CMD20 "Update CI" is 250ms (SDHC/SDXC/SDUC) (see Section 4.13.4.3.4).

4.13.4.7.4.4 Update CI Error Conditions

CMD20 "Update CI" has no error conditions.

4.13.4.7.5 Suspend AU

4.13.4.7.5.1 Suspend AU Description

A value of 0101b in the SCC field (see Section 4.13.4.7.1) of CMD20 is "Suspend AU". This function is not used in Speed Class recording or in UHS Speed Grade recording.

This function indicates that Video Speed Class host intends to suspend Video Speed Class recording. On receiving CMD20 "Suspend AU", the card shall recognize that Video Speed Class recording is suspended, and store the lowest address in the next RU to be recorded as the suspension address. This suspension address is available in the SUS_ADDR field (see Section 4.10.2.12).

After suspension, the card leaves the recording period and it doesn't ensure performance based on Video Speed Class Specification. To restart Video Speed Class recording, the host shall follow the resume procedure (see Section 4.13.4.1.2).

And after suspension, if host alters any data in the target AU which includes the suspension address, the suspension address is cleared by the card (see Section 4.10.2.12).

If CMD20 "Suspend AU" is issued under the conditions that the suspension address has already been registered in the card, the suspension address is not changed.

If CMD20 "Suspend AU" command is issued to a partially filled RU (see Section 4.13.4.1.2), then:

a) any assigned DIR slot are released;
b) any active AUs are released; and
c) the SUS_ADDR field is cleared to zero.

Application Note: If CMD20 "Suspend AU" is not issued at SU boundary, the host should estimate the write time of the incomplete SU as write time of one complete SU.

4.13.4.7.5.2 Suspend AU Parameters

For CMD20 "Suspend AU":

a) VSC is set to 1b;
b) CNT/ID is reserved; and
c) ADDR is reserved.

4.13.4.7.5.3 Suspend AU Busy Timing

The maximum tBUSY for CMD20 "Suspend AU" is 250ms (see Section 4.13.4.3.4).

4.13.4.7.5.4 Suspend AU Error Conditions

CMD20 "Suspend AU" has no error conditions.

Application Note: If the card did not suspend for any reason, then the only warning to the host is that the SUS_ADDR field (see Section 4.10.2.12) is set to zero.

4.13.4.7.6 Resume AU

4.13.4.7.6.1 Resume AU Description

A value of 0110b in the SCC field (see Section 4.13.4.7.1) of CMD20 is "Resume AU". This function is not used in Speed Class recording or in UHS Speed Grade recording.

This function indicates that Video Speed Class host intends to resume stream recording. Video Speed Class hosts issue this command at any time after suspension. See Section 4.13.4.1.2 for a description of the procedure to suspend and resume.

Application Note: After CMD20 "Resume AU" is issued and before CMD20 "Start Recording" is issued, any write command to the AU to be resumed clears the SUS_ADDR field so that the card will not resume Video Speed Class recording when requested.

Application Note: If CMD20 "Resume AU" is not issued at SU boundary, the host should estimate the write time of the incomplete SU as write time of one complete SU.

4.13.4.7.6.2 Resume AU Parameters

For CMD20 "Resume AU":

a) VSC is set to 1b;
b) CNT/ID is reserved; and
c) ADDR is reserved.

4.13.4.7.6.3 Resume AU Busy Timing

The maximum tBUSY for CMD20 "Resume AU" is 250ms (see Section 4.13.4.3.4).

4.13.4.7.6.4 Resume AU Error Conditions

If the card receives CMD20 "Resume AU" command and the SUS_ADDR field (see Section 4.10.2.12) is zero, then the card indicates ERROR in R1 (ERX).

4.13.4.7.7 Set Free AU

4.13.4.7.7.1 Set Free AU Description

A value of 0111b in the SCC field (see Section 4.13.4.7.1) of CMD20 is "Set Free AU". This function is not used in Speed Class recording or in UHS Speed Grade recording.

This function indicates that Video Speed Class host assigns one or more sequential write AUs for Video Speed Class recording. See Section 4.13.4.1 for a description of the Video Speed Class recording flow.

After issuing this command, data stored in the active AUs may not be preserved and the card may return indeterminate data until subsequent write commands succeed.

If more than one AU is bundled in this command, then the AUs are assigned are sequential, starting at the AU specified in the ADDR field (see Section 4.13.4.7.7.2).

If the card receives a CMD20 "Set Free AU" command before the current assigned AUs are filled, then
a) the contents of the unwritten portion of the previous bundled AUs are undefined; and
b) the contents of the newly assigned AUs may not be preserved (e.g. the data is undefined).

The read response for LBAs that are assigned by this command is specified in Section 4.13.4.1.6.

Figure 4-63 illustrates an example of the relationship between assigned AUs and the active AU. The figure illustrates the assigned AUs and the active AU after a sequence where:
1. A CMD20 "Set Free AU" assigned four AUs
2. The first two AUs have been filled by Video Speed Class recording
3. The third AU of the original bundle is to be used for the next Video Speed Class recording (i.e., the active AU)
4. The third AU and fourth AU of the original bundle are still reserved for Video Speed Class recording (i.e., the assigned AUs)

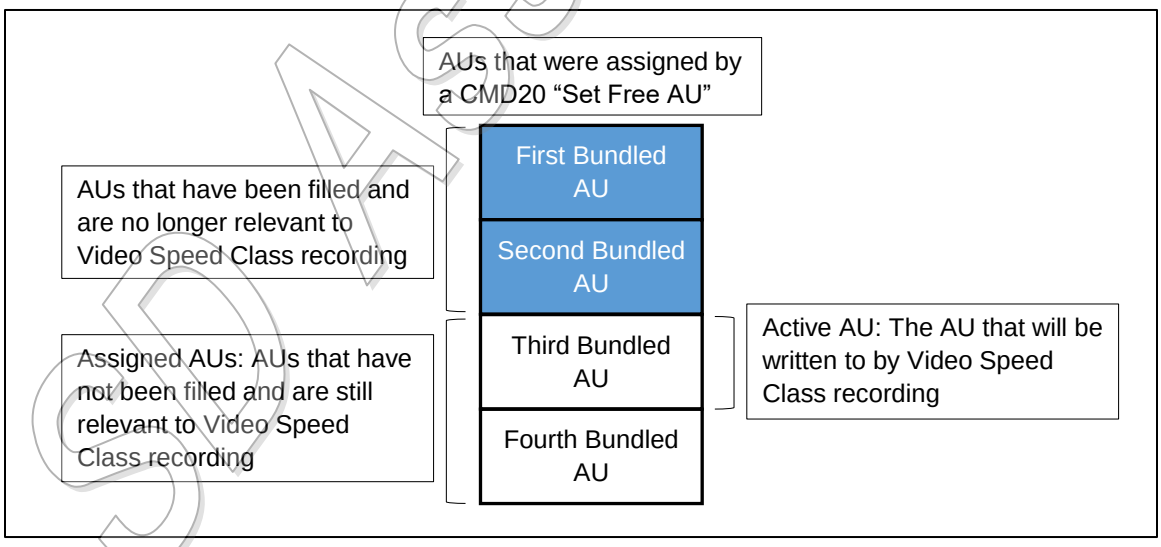


Figure 4-63 : Example of the Relationship between Assigned AUs and the Active AU

4.13.4.7.7.2 Set Free AU Parameters

For CMD20 "Set Free AU":

a) VSC is set to 1b;
b) CNT/ID contains the count of the number of bundled AUs to be assigned (e.g., 000b is one AU, 111b is eight AUs); and
c) ADDR contains the lowest address of the first bundled AU where the address is in 512KB units (e.g., the first usable free AU may be AU 3).

Cards shall support all CNT/ID field count values.

4.13.4.7.7.3 Set Free AU Busy Timing

The maximum tBUSY for CMD20 "Set Free AU" is 250ms (see Section 4.13.4.3.4).
The busy time of CMD20 "Set Free AU" is included in Pw calculations (see Section 4.13.4.3.1).

4.13.4.7.7.4 Set Free AU Error Conditions

If the card receives CMD20 "Set Free AU" command with non-AU aligned address, then the card indicates ERROR in R1 (ERX).

Application Note: If the card receives CMD20 "Set Free AU" command with the ADDR field set in the file system area (see Section 4.13.4.5), then the Video Speed Class performance may not be achieved.

4.13.4.7.8 Release DIR

4.13.4.7.8.1 Release DIR Description

A value of 1000b in the SCC field (see Section 4.13.4.7.1) of CMD20 is "Release DIR". This function is not used in Speed Class recording or in UHS Speed Grade recording.

This function indicates that the DIR address stored in the specified slot shall be cleared. This function used before reassigning the directory entry assigned to a slot ID as shown in Section 4.13.4.1.3.

4.13.4.7.8.2 Release DIR Parameters

For CMD20 "Release DIR":

a) VSC is set to 1b;
b) CNT/ID shall contain the slot ID to be released (e.g., 000b is the first slot); and
c) ADDR is reserved.

4.13.4.7.8.3 Release DIR Busy Timing

The maximum tBUSY for CMD20 "Release DIR" is 50ms (see Section 4.13.4.3.4).

4.13.4.7.8.4 Release DIR Error Conditions

CMD20 "Release DIR" has no error conditions.

4.13.4.8 Video Speed Class Measurement Conditions

4.13.4.8.1 Video Speed Class Measurement Conditions Overview

Video Speed Class Measurement Conditions are defined in Table 4-69.

Item	Definition
AU Size	See Section 4.13.4.2.2
RU Size	See Section 4.13.4.2.3
SU Size	See Section 4.13.4.2.4 and Section 4.10.2.11
Clock Condition	See Section 4.13.4.8.2
Power Limit	See Section 4.13.4.8.3
Pw Measurement Method	See Section 4.13.4.3.1
FAT Update Timing	See Section 4.13.4.6
Update CI Timing	See Section 4.13.4.7.4.3
Update DIR Timing	See Section 4.13.4.7.3.3

Table 4-69 : Video Speed Class Measurement Conditions

4.13.4.8.2 Clock Condition

Clock condition to measure each Video Speed Class value is shown in Table 4-70. Video Speed Class is not supported in Low Power Mode of UHS-II.

VSC Value	DS [MHz]	HS [MHz]	UHS-I					UHS-II FD			UHS-II HD		
			SDR12 [MHz]	SDR25 [MHz]	DDR50 [MHz]	SDR50 [MHz]	SDR104 [MHz]	RCLK [MHz]	PLL Range[2]	N_FCU	RCLK [MHz]	PLL Range[2]	N_FCU
VSC6[1]	N/A	20	N/A	20	40	80	80	26	A	1	26	A	1
								26	B		26	B	
VSC10[1]	N/A	40	N/A	40	40	80	80	35	A	1	35	A	1
								26	B		26	B	
VSC30	N/A	N/A	N/A	N/A	45	90	90	26	B	4	26	B	4
VSC60	N/A	N/A	N/A	N/A	N/A	N/A	N/A	38	B	4	38	B	4
VSC90	N/A	N/A	N/A	N/A	N/A	N/A	N/A	47	B	16	47	B	16

Note1: Video Speed Class 6 and Video Speed Class 10 shall support both conditions A and B of PLL Range shown in this table.

Note2: "PLL Range" means "Transmission Speed Range".

Table 4-70 : Clock Condition for Video Speed Class

4.13.4.8.3 Power Limit

The following table shows the Power Limit for each interface when measuring Video Speed Class performance. Host should supply this power supply level or more to ensure Video Speed Class performance.

For a given interface, the power requirements of the card will be lower for lower speeds.

Interface	Power Limit [W]	Video Speed Class
HS	0.72	6, 10
UHS-I	0.72	6, 10 (SDR25)
	1.44	6, 10, 30 (DDR50, SDR50 and SDR104)
UHS-II	1.44	6, 10, 30
	1.80	6, 10, 30, 60, 90

Table 4-71 : Power Limit per Interface for Video Speed Class

4.13.4.9 Host Operating Frequency

Card performance is specified with a fixed clock frequency for each Video Speed Class mode and Bus Speed mode, see Table 4-70. To ensure a level of performance, a host has to use the specified frequency or higher for Video Speed Class recording.

In case of UHS-II mode, use of Range B covers all Video Speed Class modes but the RCLK frequency is different for each Video Speed Class mode. Video Speed Class 6, Video Speed Class 10 and Video Speed Class 30 can be used in RCLK range 26MHz to 52MHz, Video Speed Class 60 can be used in 38MHz to 52MHz and Video Speed Class 90 can be used in 47MHz to 52MHz. For example, if the measurement condition defines RCLK=38MHz in Transmission Speed Range B (x30 Multiplier), the host needs to use 38MHz to 52MHz with Transmission Speed Range B (x30 Multiplier).

4.13.5 SD Express Speed Class Specification

SD Express Speed Class specification is introduced to perform real time recording over PCIe bus of SD Express card. Refer to Section 8.4 for its detailed specification.

4.14 Erase Timeout Calculation

This chapter provides the guideline for long erase and a method to calculate erase timeout value.

4.14.1 Erase Unit

The Speed Class Specification defines a new management unit of AU (Allocation Unit). Erase timeout calculation is defined as the basis of AU. SD memory card supports block erase but it takes more time to erase blocks, which are part of AU (partial erase AU). In this case, the host should add 250 ms to the result of timeout calculated on AU basis. When the start and end blocks are in the same partially erase AU, 500ms should be added.

4.14.2 Case Analysis of Erase Time Characteristics

Figure 4-64 shows an example of erase characteristics, number of AU erased versus erase time. Erase time is derived from erasing specified numbers of AUs by one erase command. Assuming that Erase is performed on AU basis and its erase characteristics can be approximated to a linear line. The line A illustrated in Figure 4-64 is an example characteristic.

The red line indicates the erase timeout value the host should use. The timeout value can be determined by line A. If the erase timeout is less than 1 second the host should use 1 second as timeout. If the timeout is bigger than 1 second the host should use the value determined by Line A.

Register parameters N_{ERASE}, T_{ERASE} and T_{OFFSET} define the shape of the line. T_{ERASE} indicates timeout for erasing N_{ERASE} AUs from T_{OFFSET}. T_{ERASE} and N_{ERASE} determine the slope of the line. T_{OFFSET} adjusts the line by moving in parallel on the upper side. The card manufacturer shall determine these parameters so that the line is always greater than the erase time of any AUs. Actual erase time shall be always less than erase timeout and the slope of the line shall be less than 3 second per AU.

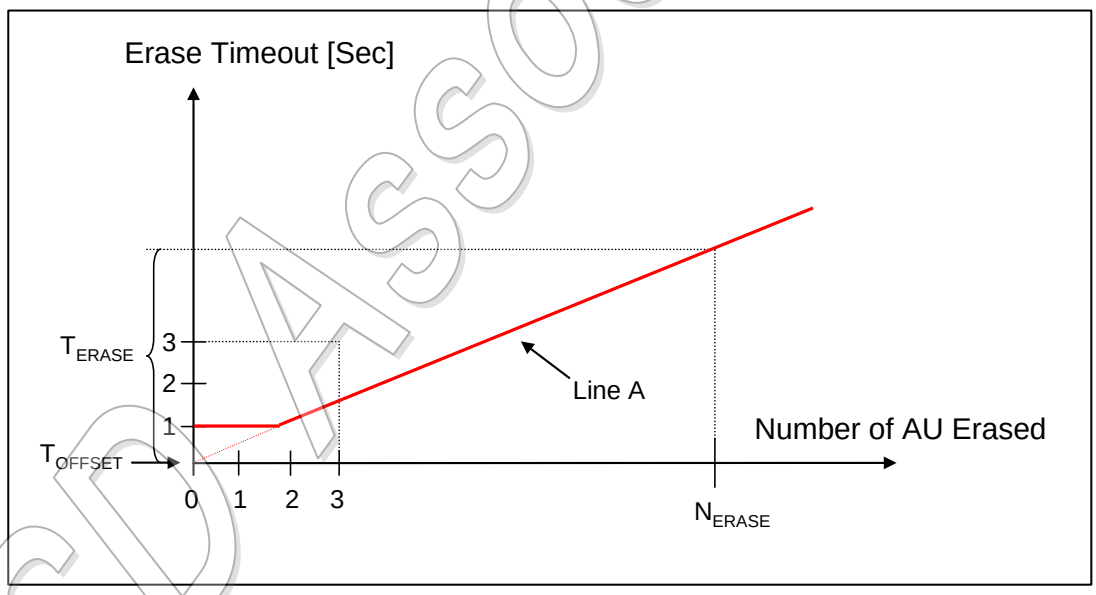


Figure 4-64 : Example Erase Characteristics (Case 1 TOFFSET=0)

The line B illustrated in Figure 4-65 shows another example of erase characteristics. The red line indicates the erase timeout value that the host should use. Since the time-out is bigger than 1 second, the red line and line B are equivalent.
Erase time of an AU shall be less than 3 second. T_{OFFSET} is mainly used to adjust erase timeout of an AU.

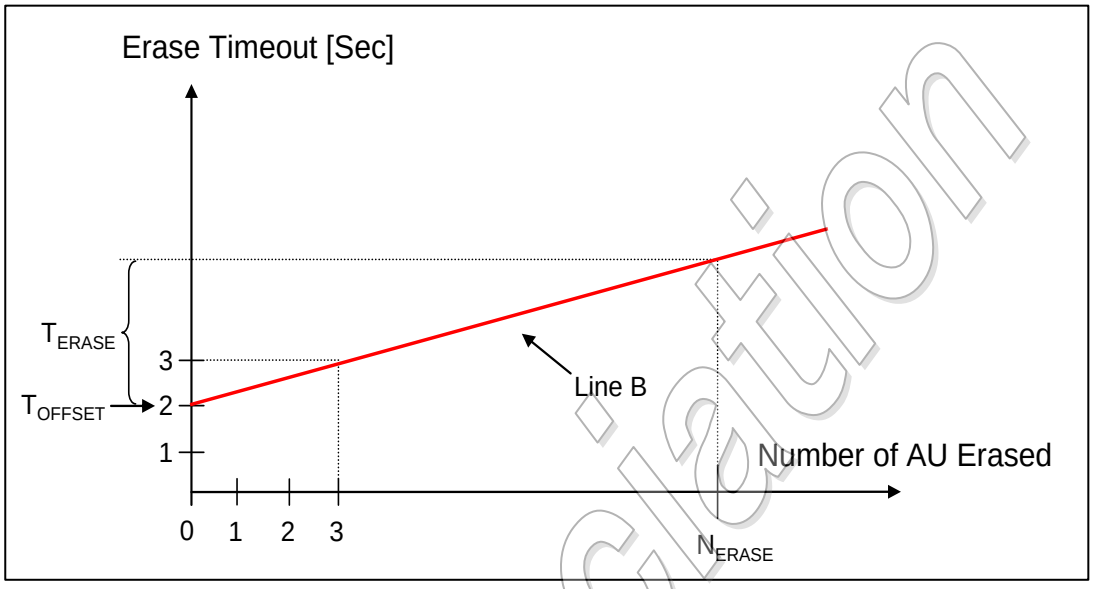


Figure 4-65 : Example Erase Characteristics (Case 2 TOFFSET=2)

4.14.3 Method for Erase Large Areas

The calculated erase timeout for multiple AUs might be too large compared with the actual erase time. The calculation of erase timeout is not accurate because calculated timeout includes a margin. A margin per AU accumulates and the result of calculating the timeout for large number of AUs will include large margins. Such calculations would be meaningless because the range of margin might be in order of minutes. Therefore, a small number of AUs should be erased at one time. This enables the host to calculate smaller timeout with fewer errors.

Application Note: When a large area is erased, the host should divide it into small areas at the AU boundary and continuously erase the small areas using a small area erase timeout. It may take a long time to erase a large area, so the host should inform the user about the erase progress, otherwise the user might abort the execution of the erase.

4.14.4 Calculation of Erase Timeout Value Using the Parameter Registers

Erase Timeout of X AU can be calculated by Equation (9).

$$\text{Erase Time-out of X AU} = \frac{T_{ERASE}}{N_{ERASE}} \cdot X + T_{OFFSET} \quad \text{.....(9)}$$

Erase timeout is determined by following steps:
(1) Calculate Equation (9).
(2) If the result of (1) is less than 1 second, the timeout is set to 1 second.
(3) 250 ms should be added to the result of (2) for each partial erase AU. When the start and end blocks are in partially erase AUs, add 500 ms to the result of (2).

4.15 Set Block Count Command

CMD12 has been used to stop multiple-block Read / Write operation. However, CMD12 is timing dependent and it is difficult to control timing to issue CMD12 at exact timing. As UHS104 card has large delay variation between clock and data, CMD23 is useful for the host to stop multiple read / write operation instead of CMD12. Host is not necessary to control timing of CMD12. This command is applicable to always 512-byte block length read/write operation and then SDSC card does not support this command. Support of CMD23 is mandatory for UHS104 card.

Support of CMD23 is defined in SCR. The response type of CMD23 is R1 and busy is not indicated.
CMD23 is accepted in transfer state and effective to the multiple-block read/write command (CMD18 or CMD25) just behind CMD23. If another command follows CMD23, set block count is canceled (including CMD13). For SDUC, CMD13 does not cancel the operation sequence (see Section 4.20.3 for more details). If command CRC error occurs, the card does not return R1 response for CMD23. In this case, Set block count is not valid and retry of CMD23 is required. If multiple CMD23 are issued, the last one is valid.
Figure 4-66 shows the definition of CMD23. If block count in the argument is set to 0, CMD23 has no effect. The block count value set by CMD23 is not checked by the card and then CMD23 does not indicate any error in the response (A previous command error is indicated in the response of CMD23). If illegal block count is set, out of range error will be indicated during read/write operation (For example, data transfer is stopped at user area boundary). Host needs to issue CMD12 if any error is detected in the CMD18 and CMD25 operations. If a CMD25 is aborted and the amount of data transferred is less than the amount of data indicated by the preceding CMD23, then the area specified by CMD23 that is unwritten may contain undefined data. If the amount of data transferred is greater than the amount of data indicated by the preceding CMD23, then the extra data is not written.

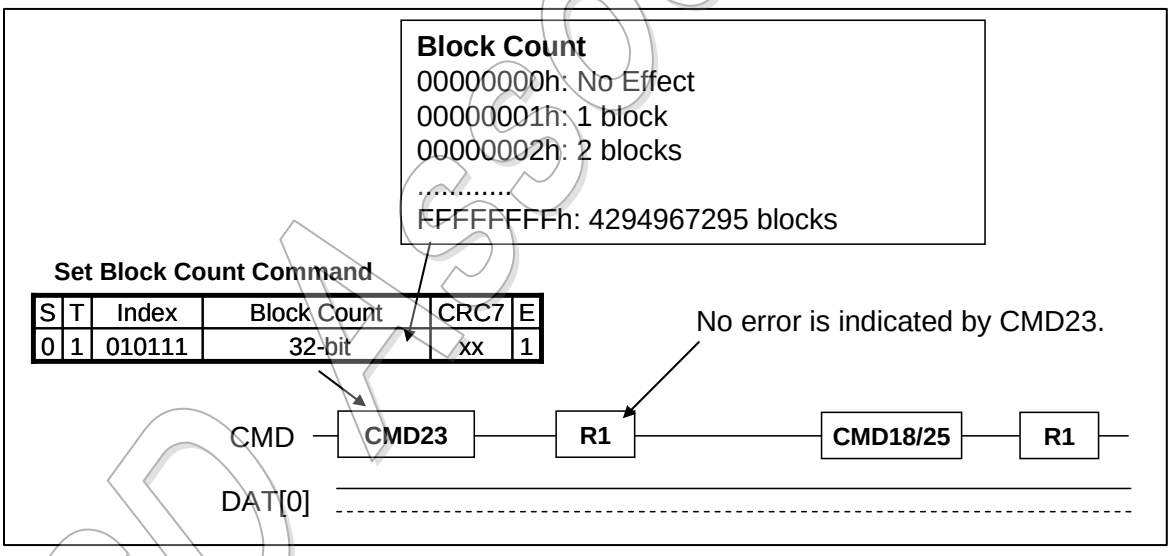


Figure 4-66 : Set Block Count Command

ACMD53 (SECURE_RECEIVE) and ACMD54 (SECURE_SEND) requires CMD23 prior to these commands to designate the number of transmission blocks. So if card supports ACMD53/54, it always shall support CMD23. ACMD53 and ACMD54 that are not preceded by CMD23 shall be considered as Illegal commands.
Note that there are some additional rules for SDSC card supporting ACMD53/54 as follows:

- CMD23 affects only ACMD53/54 that follow after it.
- The size of data transmission block is always 512 bytes regardless of BLOCK_LEN value.

- CMD23 does not work (or is ignored) for any other commands except ACMD53/54.

For example, when SDSC card receives CMD23, CMD18 and ACMD53 in this order, the card operation is as follows:

- CMD23 will be accepted.
- CMD18 will be executed (based on the BLOCK_LEN specified by CMD16). CMD23 setting will be discarded.
- ACMD53 will fail as it is not preceded by CMD23

.

4.16 Application Performance Specification

The Application Performance specification classifies card random performance levels under specific conditions. The specification defines a level of performance that satisfies both random and sequential access requirements as defined below. The level of performance may be used to indicate a sufficient performance levels for specific products.

4.16.1 Application Performance Classes

4.16.1.1 Application Performance Class 1

Application Performance Class 1 assures the following performance levels:

- Random Performance under the defined conditions as defined in Section 4.16.2:
  Read : 1500 IOPS
  Write : 500 IOPS
- Sustained Sequential Write performance:
  Minimum of 10MB/s write performance while writing any contiguous
  1GB range using aligned 4MB chunks, overwriting a fully sequentially written card.

The preconditions and test method of the given levels provided in the following sections.

4.16.1.2 Application Performance Class 2

Application Performance Class 2 assures the following performance levels with the following conditions:

- Command Queue and Cache shall be enabled as below:
  1. SD Express Card or SDUC Card:
     If Command Queue is supported, Command Queue shall be enabled and used (as support of Command Queue is optional) along with Cache enabled.
  2. Other type of Card:
     Command Queue shall be enabled and used (as support of Command Queue is mandatory) along with Cache enabled.
- Random Performance under the conditions as defined in Section 4.16.2:
  Read : 4000 IOPS
  Write : 2000 IOPS
- Sustained Sequential Write performance:
  Minimum of 10MB/s write performance while writing any contiguous 1GB range using aligned 4MB chunks, overwriting a fully sequentially written card.

Application Performance Class 2 supported card also shall support Application Performance Class 1 under the condition of disabling Command Queue and Cache.

> Application Note:
> For SD Express cards and SDUC cards, Command Queue feature is optional to meet A2. If Card indicates support for CQ and A2, then host can achieve A2 performance with CQ enabled. If Card indicates support of A2 only without the support of CQ, then host can achieve A2 performance without CQ scheme.

Card shall indicate the queue depth required to meet Application Performance Class 2 in both Performance Enhancement Function Register Byte[6], refer to Table 5-30, and SD_STATUS(bit[335:331]), refer to Table 4.10.2-2. Card that meets Application Performance Class 2 shall support LV signaling, refer to Low Voltage Interface Addendum.

The preconditions and test method of the given levels provided in the following sections

4.16.2 Application Performance Class Measurement Conditions

The Application Performance Class measurement conditions include the clock condition, power limit, precondition state of the card and the performance parameters

4.16.2.1 Clock Condition

Clock condition to measure random and sustained sequential performance is shown in Table 4-72. Application Performance is not supported in Low Power Mode of UHS-II.

Application Performance Class	DS [MHz]	HS [MHz]	UHS-I					UHS-II FD			UHS-II HD		
			SDR12 [MHz]	SDR25 [MHz]	DDR50 [MHz]	SDR50 [MHz]	SDR104 [MHz]	RCLK [MHz]	PLL Range[2]	N_FCU	RCLK [MHz]	PLL Range[2]	N_FCU
A1/A2	N/A	NA	N/A	NA	40	80	80	35	A	1	35	A	1
								26	B		26	B	

Table 4-72 : Clock Condition for Application Performance Class

4.16.2.2 Power Limit

The following table shows the Power Limit for each interface when measuring Application Performance. Host should supply this power supply level or more to ensure card Application Performance.
For a given interface, the power requirements of the card shall be lower for lower speeds.

Interface	Power Limit [W]	Application Performance Class
UHS-I	1.44	A1, A2 (DDR50, SDR50 and SDR104)
UHS-II	1.80	A1, A2

Table 4-73 : Power Limit per Interface for Application Performance Class

4.16.2.3 Host Operating Frequency

Application Performance is measured with a fixed clock frequency for each Bus Speed mode, see Table 4-72. To ensure a level of Application Performance, a host has to use the specified frequency or higher for access.

4.16.2.4 Application Performance Measurement Conditions Overview

The Application Performance measurement conditions include the precondition state of the card and the Application performance test. .

4.16.2.4.1 Precondition

The card under measurement shall be preconditioned with the following steps

1. Full user area is erased by CMD38 FULE as described in Section 4.3.5.3 (If FULE is not supported, card shall execute legacy erase).
2. 75% of user area shall be written sequentially using CMD25 with data size of 4MB, aligned with 4MB boundary starting LBA 0.
3. On already written region as in step 2, sequential write 256MB area starting from a randomly selected 4KB aligned LBA.

The preconditioned 256MB region shall be noted as ‘Random Performance Test Unit’ (RPTU).

For Application Performance Class 2, if Command Queue is supported, Sequential CQ mode shall be used to carry out the precondition. Step 2 and 3 as mentioned above shall be written sequentially using class 1 commands, as below, in increasing order of Task ID.

- Task Submission : CMD44/CMD45 with Block count = 8192 [4MB size] and increasing start block

address per Task ID.
- Task Execution : CMD47 with increasing order of Task ID

4.16.2.4.2 Random Performance Test

Perform random writes / reads by targeting 4KB sized IO, aligned with 4KB unit, to the already preconditioned 256MB region, RPTU. Write and Read test shall be carried out for 10minutes each, over RPTU. The same RPTU shall be used to measure the random write and random read performance.

For Application Performance Class 2, Cache shall be enabled and, if Command Queue is supported, Voluntary CQ mode shall be used to carry out random performance test. Average random read/write performance of the card shall be measured as average number of 4KB sized task carried out per second over RPTU.

- Task Submission : CMD44+CMD45 with Block count = 8 [4KB size]
- Task Execution : CMD46/CMD47 with Task ID which was marked 'ready' by card
  - CMD46 used during random read performance test
  - CMD47 used during random write performance test

4.16.2.5 Sustained Sequential Write Performance Measurement Conditions Overview

The Sustained Sequential Write performance measurement conditions include the precondition state of the card and the sequential performance test.

4.16.2.5.1 Precondition

The card under measurement shall be preconditioned with the following steps

1. Full user area is erased by CMD38 FULE as described in Section 4.3.5.3 (If FULE is not supported, card shall execute legacy erase).
2. 100% of user area shall written sequentially using CMD25 with data size of 4MB, aligned with 4MB boundary

For Application Performance Class 2, if Command Queue is supported, Sequential CQ mode shall be used to carry out the precondition. 100% of user capacity shall be written sequentially using class 1 commands, as below, in increasing order of Task ID.

- Task Submission : CMD44/CMD45 with Block count = 8192 [4MB size] and increasing start block address per Task ID.
- Task Execution : CMD47 with increasing order of Task ID

4.16.2.5.2 Sequential Write Performance Test

Sequential write up to 1GB (1024MB) using CMD25 with data size of 4MB, aligned with 4MB boundary, to the already preconditioned card.

For Application Performance class 2, if Command Queue is supported, Sequential CQ mode shall be used to carry out sequential write performance test. Sequential write up to 1GB (1024MB) shall be carried out using Class 1 commands, as below, in increasing order of Task ID.

- Task Submission : CMD44/CMD45 with Block count = 8192 [4MB size] and increasing start block address per Task ID
- Task Execution : CMD47 with increasing order of Task ID

4.16.3 Application Performance Class Parameters

4.16.3.1 Performance of Random Write, (PRw)

Random write performance, PRw is measured as average number of 4KB sized writes performed per second for a duration of 10minutes over RPTU and the unit is IOPS.

For example, PRw 500 IOPS denotes, on an average 2000KB of data is transferred to card in one second.

4.16.3.2 Performance of Random Read, (PRr)

Random read performance, PRr is measured as average number of 4KB sized reads performed per second for a duration of 10minutes over RPTU and the unit is IOPS.

For example, PRr 1500 IOPS denotes, on an average 6000KB of data is read from card in one second.

4.16.3.3 Performance of Sustained Sequential Write, (PSSw)

Sustained sequential write performance, PSSw is measured as the average write performance of the card measured over 1GB contiguous region using CMD25 with data size of 4MB.

4.17 Cache

Card shall indicate Cache feature support in the Performance Enhancement Function Register Byte[4], refer to Table 5-30 and 'PERFORMANCE_ENHANCE' of SD Status(b[330]), refer to Table 4.10.2-2. The Cache shall be transparent to the host and host does not have direct access to the contents in the Cache. It is up to card implementation to decide which of host data goes into Cache.

Host shall exercise 'Flush Cache', Performance Enhancement Function Register Byte[261] bit[0]='1' as in Table 5-30, at least once before power off. Otherwise data kept in Cache may be lost. Any power loss occurring during 'Flush Cache' operation may end in losing host data.

Card shall not take more than 1 second to complete 'Flush Cache' operation. Card shall indicate 'busy' using DAT0 line during 'Flush Cache' operation. Card shall ensure that the Cache size is as much required to complete 'Flush Cache' operation within 1 second.

Card shall support 'Power off Notification' when supporting Cache and on reception of 'Power off Notification', host data stored in Cache shall be flushed to non-volatile memory.

Below flow chart illustrates checking for Cache support, enable Cache and power off notification

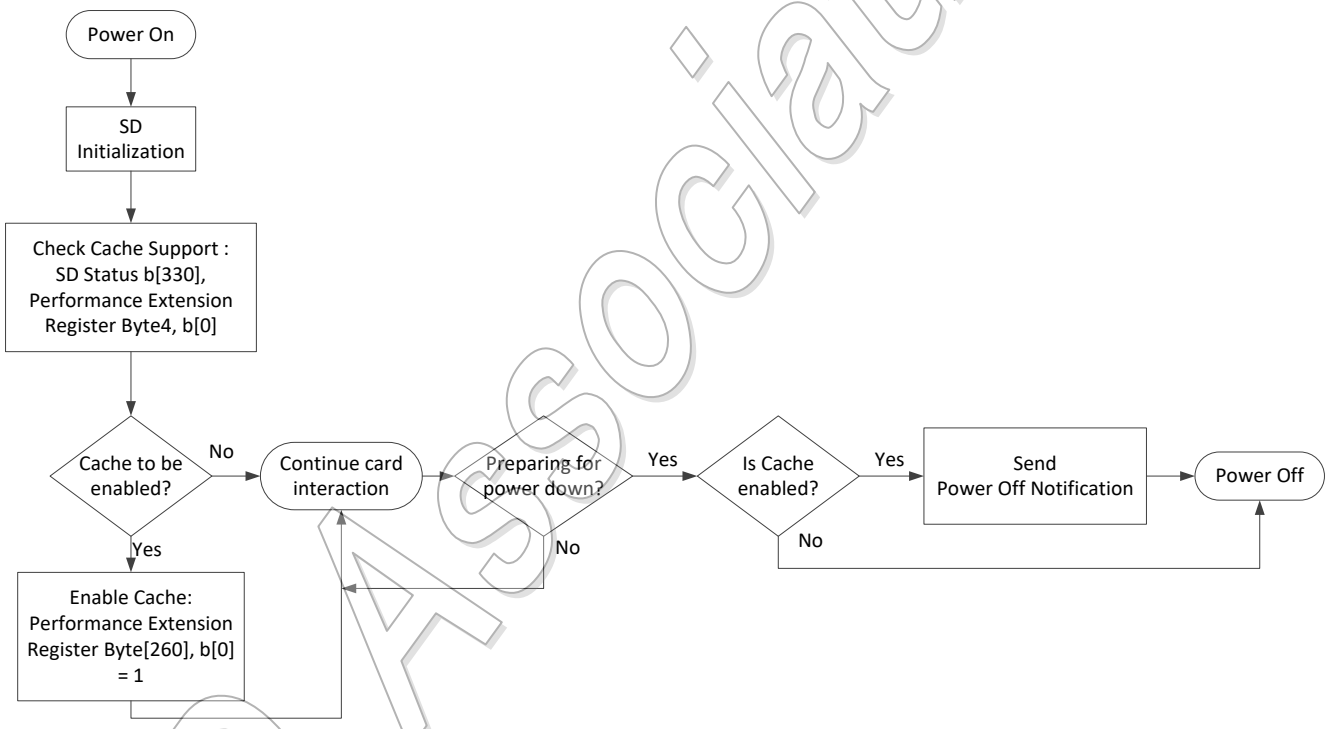


Figure 4-67 : Flowchart for Checking Cache

4.18 Self Maintenance

Card may carryout internal maintenance using any of the two methods as follows, when enabled by the host.

- Card initiated maintenance
- Host initiated maintenance

4.18.1 Card Initiated Maintenance

Card shall indicate support for card initiated maintenance in 'PERFORMANCE_ENHANCE' of SD_STATUS (b[328]), refer to Table 4.10.2-2 and Performance Enhancement Function Register Byte[2], refer to Table 5-30. Host shall enable card initiated maintenance and 'Power Sustenance'(refer to Section 5.8.1.4) for allowing card to carryout internal maintenance operations. Once card initiated maintenance is enabled by host, card may carryout internal maintenance operations when the SD bus is idle.
Card shall receive and execute any command within the time limit defined by specification.
Host that enable card initiated maintenance shall ensure to send 'Power Off Notification' as described in Section 5.8.1.3 before turning off power to card, so that card shall gracefully complete any active internal maintenance operation within 1 second.

Below flowchart illustrates the checking for card initiated maintenance, enabling and disabling card initiated maintenance

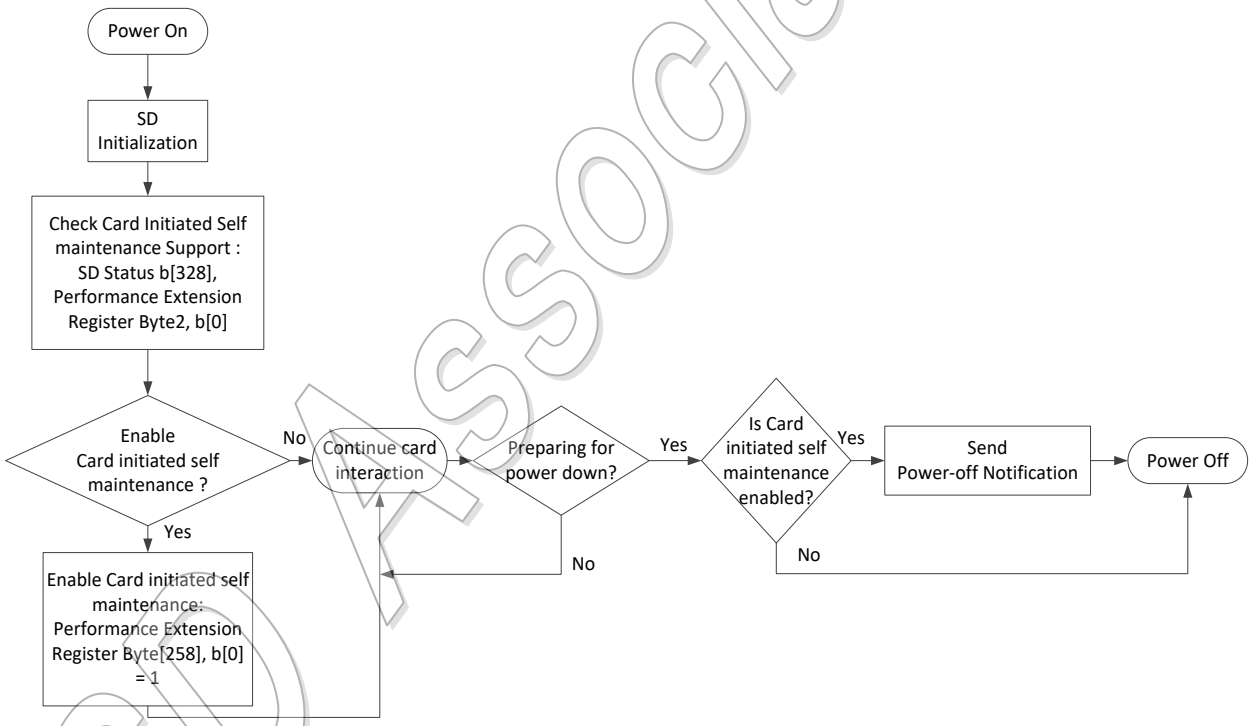


Figure 4-68 : Flowchart for Card Initiated Maintenance

4.18.2 Host Initiated Maintenance

Card shall indicate support for card initiated maintenance in 'PERFORMANCE_ENHANCE' of SD_STATUS (b[329]), refer to Table 4.10.2-2 and Performance Enhancement Function Register Byte[2], refer to Table 5-30. If card that support host initiated maintenance wants to indicate the urgency level, it shall indicate it in Performance Enhancement Function Register Byte[3], refer to Table 5-30, for carrying out internal maintenance operation. Card shall choose any one of the four urgency level whenever required. Based on urgency level host may instruct the card to carryout internal maintenance operation

up to 1 second. If the card needs additional time to carryout internal maintenance, card shall keep or change the urgency level. Card shall clear the urgency level once the internal maintenance has been completed.
Card may utilize the FX_EVENT to indicate the host that maintenance has to be carried out. FX_EVENT support is indicated in Performance Enhancement Function Register Byte[1], refer to Table 5-30. Host may enable FX_EVENT method to receive urgency level indication from the card using Performance Enhancement Function Register Byte[257], refer to Table 5-30.

Below flowchart illustrates the checking for host initiated maintenance, FX_EVENT and feature enable methods.

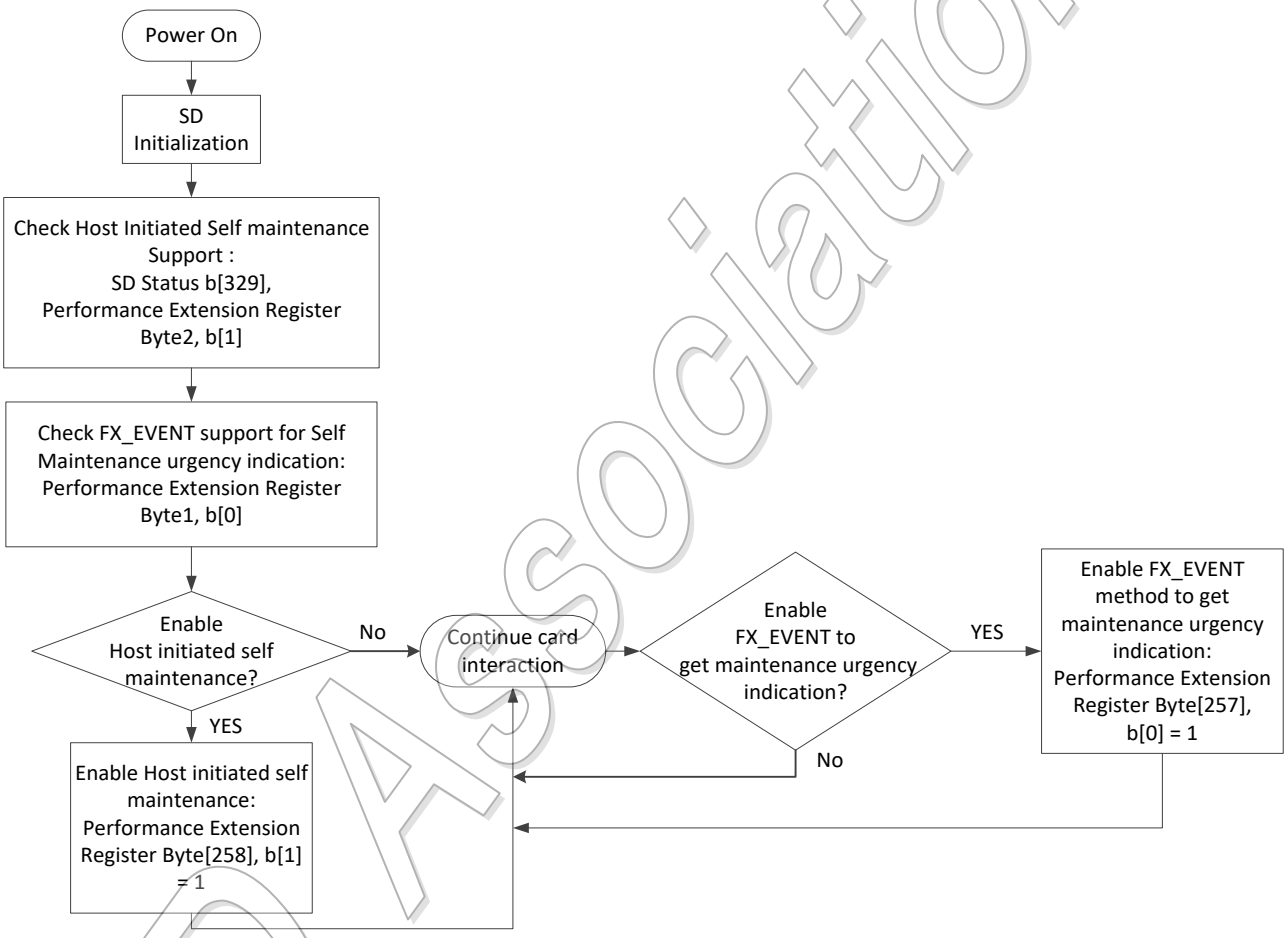


Figure 4-69 : Flowchart for Checking Host Initiated Maintenance

4.19 Command Queue

Command Queue separates bus transaction in tasks assignment phase and data transfer phase. Multiple of memory read/write tasks can be assigned. Card shall indicate the support for Command Queue and the depth of queue in Performance Enhancement Function Register Byte[6], refer to Table 5-30 and in 'PERFORMANCE_ENHANCE' of SD_STATUS b[335:331],.refer to Table 4.10.2-3. The following sections define the different phases in command queue. Card supporting command queue shall support 'Power Off Notification'. Command Queue scheme is enabled by host using Performance Enhancement Function Register Byte[262] b[0], refer to Table 5-30.

4.19.1 Command Queue Mode

Command Queue mode, herein referred by CQ mode, is selected by host using Performance Enhancement Function Register Byte[262] b[1], refer to Table 5-30. Cache Feature (Refer to Section 4.17) should be enabled before enabling Command Queue .
Two Command Queue modes are defined in this specification:

- Voluntary CQ Mode
- Sequential CQ Mode

Application Performance Class 2, refer to Section 4.16.1.2, is guaranteed only in Voluntary CQ Mode with Cache Feature enabled.

When command queue is not enabled and Class 1 commands(refer to Table 4.7.4-1) are received card shall not respond and indicate 'Illegal Command' in next R1.

4.19.1.1 Voluntary CQ Mode

Host shall set Voluntary CQ mode by setting Performance Enhancement Function Register Byte[262] b[1] = '0', refer to Table 5-30. In Voluntary CQ mode, task reordering is up to card implementation and Host shall ensure the data integrity by submitting tasks in proper order to avoid any data loss. 'Priority' field of CMD44 shall be supported by card in Voluntary CQ mode, however the readiness of a task submitted with 'Priority' may not be guaranteed by card.

4.19.1.2 Sequential CQ Mode

Host shall set Sequential CQ mode by setting Performance Enhancement Function Register Byte[262] b[1] = '1', refer to Table 5-30. In Sequential CQ mode, tasks shall be executed in increasing Task ID order. 'Priority' field of CMD44 shall be ineffective.
The following conditions are to be followed during Task assignment

- Queue shall be empty before assigning Tasks [Host may abort entire queue of tasks queued earlier]
- Tasks shall always be assigned at once from Task ID 0 in sequential order before execution
- Card shall prepare tasks from Task ID 0 in sequential order
- Host shall not abort a single pending task by CMD43 when tasks have been assigned

The following conditions are to be followed during Task execution

- Host shall execute tasks from Task ID 0 in sequential order without polling for ready state in Task Status Register, refer to Section 4.10.3.
- Host shall not abort a single pending task by CMD43 when tasks have been assigned

If the above conditions are not followed in Sequential CQ Mode, the behavior of the card is undefined. Cards should indicate an error if illegal sequence is generated by the host. CMD46 or CMD47 may be illegal during Task execution. Host shall abort all queued tasks when there are errors in Sequential CQ Mode.

4.19.2 Command Support in CQ Mode

Table 4-74 lists the command support in CQ mode is enabled and disabled [non CQ mode]

CQ Mode	Commands	Behavior
Disabled	Class 1 commands [CMD43,44,45,46,47]	Illegal
	Other Commands	Legal
Enabled	CMD0, CMD12, CMD13	Legal
	Class 1 commands [CMD43,44,45,46,47]	Legal
	CMD22(1) CMD32, CMD33, CMD38[Discard]	Legal
	Extension Function [CMD48/49/58/59]	Legal
	CMD23(2), ACMD53, 54	Legal
	CMD6, CMD19, CMD39(3)	Legal
	Others	Illegal

Note (1): CMD22 is available for SDUC card only.
Note (2): CMD23 is effective to ACMD53/ACMD54 only.
Note (3): When CMD39 is issued during CQ mode, application performance is not guaranteed.

Table 4-74 : Command Support list in CQ Mode

4.19.3 CURRENT_STATE for CQ mode

Table 4-75 lists the CURRENT_STATE values of R1 Response for CQ mode.

CURRENT_STATE Value	CQ Disabled	CQ Enabled
0	idle	reserved
1	ready	reserved
2	ident	reserved
3	stby	reserved
4	tran	CQ Tran
5	data	Rdt
6	rcv	Wdt
7	prg	Bsy
8	dis	reserved
9-14	reserved	reserved
15	reserved for I/O mode	reserved

Table 4-75 : CURRENT_STATE for CQ Mode

4.19.4 Card State Machine in CQ Mode

Figure 4-70 illustrates the state machine transition of card in CQ mode. The card state transition is described in Table 4.19.4-1.

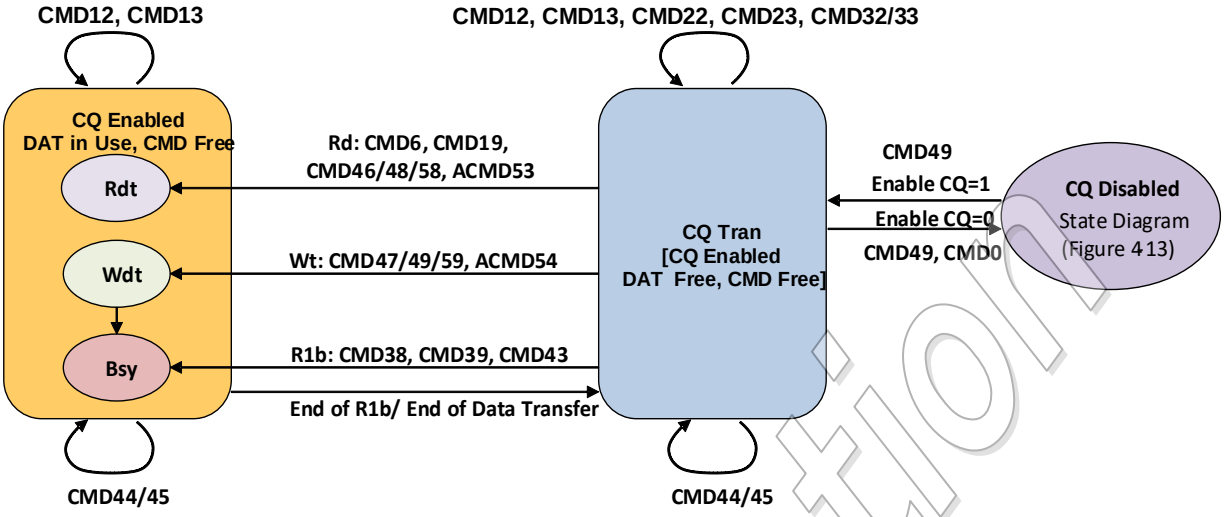


Figure 4-70 : State Diagram in CQ Mode

Commands	Current State			
	CQ tran	Rdt	Wdt	Bsy
"Operation Complete"	-	CQ tran	-	CQ tran
CMD0	idle*	idle*	idle*	idle*
CMD6	Rdt	-	-	-
CMD12	CQ tran	CQ tran	Bsy	-
CMD13	CQ tran	Rdt	Wdt	Bsy
CMD19	Rdt	-	-	-
CMD22	CQ tran	-	-	-
CMD23	CQ tran	-	-	-
CMD32	CQ tran	-	-	-
CMD33	CQ tran	-	-	-
CMD38	Bsy	-	-	-
CMD39	Bsy	-	-	-
CMD48	Rdt	-	-	-
CMD49	Wdt	-	-	-
CMD49 ("Enable CQ"=0)	tran**	-	-	-
CMD58	Rdt	-	-	-
CMD59	Wdt	-	-	-
CMD43	Bsy	-	-	-
CMD44	CQ tran	Rdt	Wdt	Bsy***
CMD45	CQ tran	Rdt	Wdt	Bsy***
CMD46	Rdt	-	-	-
CMD47	Wdt	-	-	-
ACMD53	Rdt	-	-	-
ACMD54	Wdt	-	-	-

* non-CQ modes
** After CMD49 completed, card returns to tran state.
The state transition is "CQ tran -> Wdt -> Bsy -> CQ tran -> tran".
*** Host is recommended not to assign another task when card is in 'Bsy' due to CMD43. Otherwise, card's behavior may be unexpected.

Table 4.19.4-1: Card State Transition in CQ Mode

4.19.5 Task Submission

A data transfer task shall be assigned to command queue by the host using CMD44 followed by CMD45. CMD44 arguments contain direction of data, Priority, Task ID and the total number of blocks to be transferred. CMD45 argument contains the start block address for the Task ID given in CMD44. On successful reception of Task submission[CMD44+CMD45], card shall respond with R1 response for each command and shall queue the task. When a valid task is received, card shall clear the corresponding error bit in Task Error Status of Performance Enhancement Function Register Bytes[15:8], refer to Table 5-30.

If CMD44 is not successful due to the following, card shall not respond and indicate 'Illegal Command' in the next R1.

- Task ID already exists
- Task ID is larger than supported Queue Depth
- Number of blocks equals to 0

If CMD44 is not successful for other reasons, card shall indicate appropriate errors during the Task Execution phase. If CMD45 is not immediate after a valid CMD44, card shall not respond and indicate 'Illegal Command' in next R1.
A task is assigned when valid CMD44 is followed by valid CMD45.
If another command follows CMD44, the CMD44 is invalidated (including CMD13).If multiple CMD44 are issued, the last one is valid.

Host may add a task to queue while the DAT lines are idle or during active data transfer or during indication of BUSY by card. In Voluntary CQ mode, host may add any number of task to queue, up to supported queue depth, as long as unique Task IDs are available. In Sequential mode, host shall ensure that tasks are queued in increasing order of Task ID.
Before disabling CQ, host shall make sure the queue is empty, or card will discard all tasks in queue.

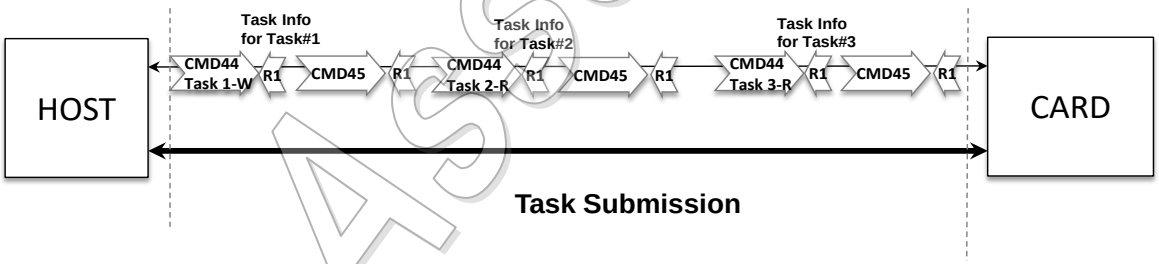


Figure 4-71 : Illustration of Task Submission

4.19.6 Queued Task Status Check

To understand which of the queued tasks are ready, Host shall read the Task status register by sending CMD13 SEND_STATUS – Send Task Status extension command in Voluntary CQ mode. Card shall indicate the ready status by setting '1' in the corresponding Task ID bit of Task Status Register, refer to Section 4.10.3.

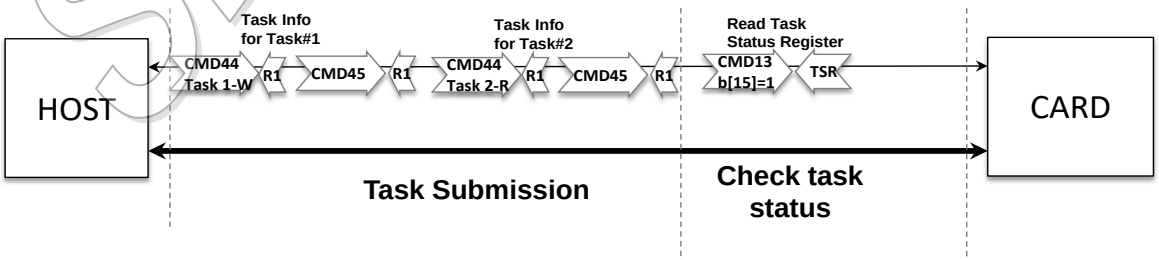


Figure 4-72 : Illustration of Reading Task Ready Status

Card shall ensure at least one task to be ready within 250ms of successful task submission after R1 of CMD45. However host shall not expect additional queued tasks to be set as 'ready' by repeatedly reading Task status. In Sequential CQ mode, host may not need to check the Task status for execution.

4.19.7 Execution of Task

Host shall execute queued task using CMD46 and CMD47. CMD46 for Read Task and CMD47 for Write Task .Host shall issue CMD46/CMD47 for the tasks which are marked as ready by card in Voluntary CQ mode. In Sequential CQ mode, host shall issue CMD46/47 with task ID in increasing order irrespective of Task ID's ready status. On successful reception of CMD46 card shall be ready to start the data transmission up to the block numbers, which is specified for the Task ID earlier during task submission, within 100ms. On successful reception of CMD47 card shall be ready to receive the data from host up to the block numbers specified for the Task ID earlier during task submission. The write timeout per block shall be 250ms (500ms for SDXC and SDUC). Once the task is completed by CMD46/CMD47/CMD12, the information carried by CMD44/CMD45 shall be cleared and the Task ID can be reused. Also, the execution of a completed task ID is illegal.

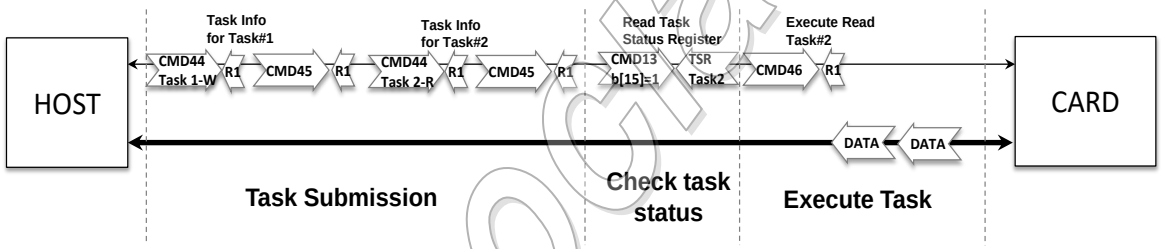


Figure 4-73 : Illustration of Executing Read Task

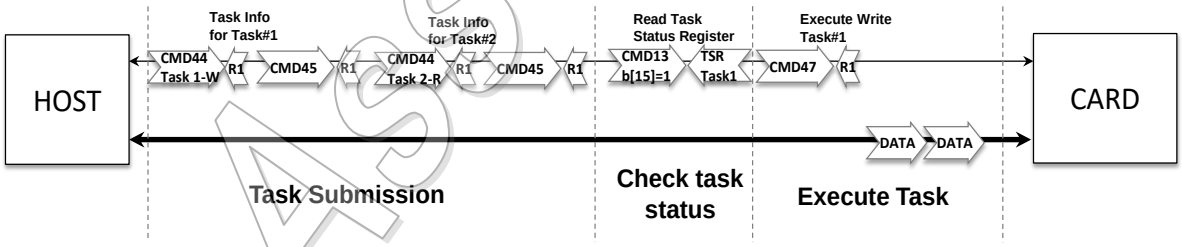


Figure 4-74 : Illustration of Executing Write Task

Card shall not respond for CMD46/CMD47 for the following conditions and indicate 'Illegal Command'

- Task ID not in queue
- Wrong data direction
- Task ID not ready in Voluntary CQ Mode
- Task ID not in sequential order for Sequential CQ Mode

Memory errors caused by an execution of CMD46/47 may be set in CMD46/47 R1 or any R1 after the CMD46/47 by Card status bit19.
Because CQ is not supported on SDSC card, data size is always 512-byte. Block length defined by CMD16 before enabling CQ is ignored and BLOCK_LEN_ERROR will not be indicated in R1 response of CMD46/CMD47. After disabling CQ, the previous definition of block length is still effective for CMD42.

Any error caused due to task parameters, submitted using CMD44+CMD45, or due to card internal operations shall be indicated with the corresponding error [ERX] in R1 response, for example

OUT_OF_RANGE, WP_VIOLATION, CARD_ECC_FAILED, ERROR, etc. The error Task ID shall be indicated in Task Error Status of Performance Enhancement Function Register Bytes[15:8], refer to Table 5-30.

4.19.8 Task Management

Host shall send CMD43 – Q_MANAGEMENT to abort either any of the queued task or the entire queue using operation code in the argument, refer to Table 4.7.4-1. For an invalid operation code, card shall indicate 'ERROR' [ERX] in R1 Response. If host sends CMD43 when queue is empty or task ID is not in queue, card shall not indicate any error.
Host is recommended not to assign another task until CMD43 is completed. Otherwise, the behavior of the next CMD44 may be unexpected.

4.19.9 CQ Commands Error Responses

Table 4.19.9-1 summarizes error responses of CMD43-47, as defined in Sections 4.19.5-4.19.8.

Command	Description	Response	Error Handling
CMD44	Task ID already exists. Task ID is larger than supported Queue Depth. Number of blocks equals to 0.	No response	Illegal Command
CMD44	If CMD44 is not successful for other reasons, card shall indicate appropriate errors during the Task Execution phase.	No error indication	Error in CMD46/CMD47
CMD45	CMD45 is not immediate after a valid CMD44	No response	Illegal Command
CMD46 / CMD47	Task ID is not in queue. Wrong data direction. Task ID not ready in Voluntary CQ Mode. Task ID not in sequential order for Sequential CQ mode.	No response	Illegal Command
CMD46 / CMD47	Error due to task parameters and submitted using CMD44+CMD45 Card internal operations error (e.g. OUT_OF_RANGE, WP_VIOLATION, CARD_ECC_ERROR, ERROR, etc.)	Error in Response	Type 'R' and 'X' error in CMD46/CMD47
CMD43	For invalid operation code, card shall indicate 'ERROR' [ERX] in R1 Response.	Error in Response	Type 'R' and Type 'X' are both valid. It depends on implementation.
CMD43	Queue is empty. Task ID is not in queue.	Regular response (R1b)	No error indication

Table 4.19.9-1: Error Case Handling of CMD43-47

4.20 Over 2TB Extension

4.20.1 Overview

Memory access commands for cards with capacity up to 2TB use 32-bit argument as block address (2^{32} (32-bit address) * 2^{9} (512B block) = 2^{41} Bytes = 2TB). This section defines memory space expansion over 2TB. It is assumed that the expansion can be achieved by only Host Driver modification without Host Controller hardware modification on most of the hosts. The expansion includes Video Speed Class support. SDUC capacity range is unrelated to the interface modes (UHS-I, UHS-II, PCIe) supported by the card.

SD Express SDUC Card can access user area in both SD mode and PCIe mode. User area capacity and data in it shall be equivalent between SD mode and PCIe mode. It is noted that there is no compatibility to over 2TB Card due to address expansion and then over 2TB extension is treated as a new function.

Following is a brief outline of required extension or modification for SDUC Card.

(1) Data Protection from Address Mismatch
Memory access with address length mismatch would cause data corruption. To avoid data corruption, mutual recognition mechanism between host and card is defined in ACMD41 that is similar to HCS and CCS for recognizing SDHC/SDXC. ACMD41 initialization can be completed when both the host and the card support over 2TB.

(2) Extension of User Area Capacity Indication
CSD Version 2.0 can indicate up to 2TB capacity by 22-bit C_SIZE of 512KB unit.
CSD Version 3.0 defines 28-bit C_SIZE (by utilizing 6-bit reserved field) of 512KB unit for SDUC Card that can extend memory space up to 128TB. This capacity limitation comes from use of 6-bit reserved field for C_SIZE.

(3) Extension of Memory Addressing
CMD22 is defined to expand 6-bit upper address. 38-bit block address enables access to 128TB memory space (matching maximum capacity of C_SIZE). CMD22 shall be issued before memory access commands such as CMD17, CMD18, CMD24, CMD25, CMD32, CMD33 and CMD20.

(4) Extension/Modification for Subsidiary Commands
ACMD22 is used to verify a number of successfully written blocks after a write operation. In case of up to 2TB Card, ACMD22 returns the number via short data block. ACMD22 of SDUC Card returns 64-bit data + CRC16.
SDUC Card does not support ACMD23 Number of Write Blocks Pre-erased.

(5) Extension for Video Speed Class
CMD22 precedes memory access commands and CMD22 execution time is counted to performance measurement.
SUS_ADDR in SD Status is expanded to 28-bit of 512KB unit by using 6-bit reserved field.
CMD20 "Set Free AU" is a special function that has 22-bit ADDR in the argument to specify bundled AU location. CMD22 preceding CMD20 expands ADDR to 28-bit.
In case of up to 2TB Card, CMD20 Set Free AU uses 22-bit 512KB unit address. In case of SDUC Card, CMD22 precedes CMD20 Set Free AU to provide 6-bit upper address of 28-bit 512KB unit address.

(6) Extension for Command Queue
CMD44 is modified to include 6-bit upper address by utilizing reserved bits.

Over 2TB Extension specifications are also applied to UHS-II bus.

4.20.2 Over 2TB Support Recognition

To avoid data corruption via address space mismatch, mutual recognition mechanism is implemented via ACMD41 initialization.

Figure 4-75 shows extension of ACMD41 for Over 2TB. HO2T is added to bit35 of ACMD41 argument and CO2T is added to bit35 of the response type R3. Mutual recognition is achieved via setting HCS/HO2T combination by a host and setting CCS/CO2T combination by a card. Host sets own capacity support ability to HCS/HO2T and SDUC Card completes ACMD41 execution only when HCS=1b and HO2T=1b. At the completion of ACMD41, R3 response is returned with indicating CCS=1b and CO2T=1b. Otherwise, SDUC card does not complete ACMD41 and the initialization is aborted by host timeout.

Argument of ACMD41

47	46	45-40	39	38	37	36	35	34-33	32	31-16	15-08	07-01	00
S	D	Index	Busy 31	HCS 30	(FB) 29	XPC 28	HO2T 27	Reserved 26-25	S18R 24	OCR 23-08	Reserved 07-00	CRC7	E
0	1	101001	0	x	0	x	x	00	x	xxxxh	0000000	xxxxxxx	1
1	1	6	1	1	1	1	1	2	1	16	8	7	1

Host Capacity Support (HCS,HO2T)
00b: SDSC Supported Host
10b: SDHC/SDXC Supported Host
11b: Over 2TB Supported Host
01b: Not used

Response of ACMD41

47	46	45-40	39	38	37	36	35	34-33	32	31-16	15-08	07-01	00
S	D	Index	Busy 31	CCS 30	UHS-II 29	Rsv 28	CO2T 27	Reserved 26-25	S18A 24	OCR 23-08	Reserved 07-00	CRC7	E
0	0	111111	x	x	x	0	x	00	x	xxxxh	0000000	1111111	1
1	1	6	1	1	1	1	1	2	1	16	8	7	1

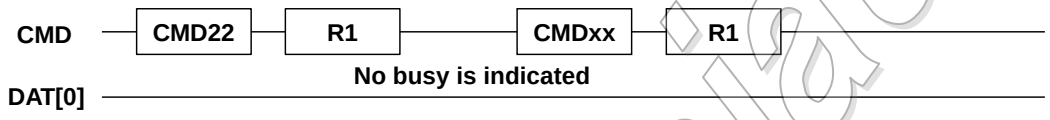
Card Capacity (CCS,CO2T) when Busy=0b
00b: SDSC Card
10b: SDHC/SDXC Card
11b: Over 2TB Card
01b: Not used

Figure 4-75 : Extension of ACMD41 for Over 2TB

4.20.3 Extension of Memory Addressing

Figure 4-76 shows definition of CMD22 Address Extension Command. Lower 6-bit of the argument is used to extend upper address. There are two types of address formats.
One is a block unit address (512-byte) that is used by memory commands (CMD17, CMD18, CMD24, CMD25, CMD32, CMD33). 38-bit block address is required to access 128TB memory space and it is configured by upper 6-bit from CMD22 and lower 32-bit from the memory command.
The other is a 512KB unit address that is used by CMD20 "Set Free AU" to designate AU location. 28-bit 512KB unit address can designate AU location in 128TB memory space and it is configured by upper 6-bit from CMD22 and lower 22-bit from CMD20.

The response type of CMD22 is R1 and CMD22 does not indicate busy. Address range is checked when a memory command is received.
Issuing CMD13 after CMD22 and CMD23 has no influence on the sequence. CMD12 resets current command sequence. Card receiving any other command not defined above (CMD12, CMD13, CMD17, CMD18, CMD23, CMD24, CMD25, CMD32, CMD33) or with wrong parameters (CMD20, which is not “Set Free AU”) shall indicate ADDRESS_ERROR in R1 response.



Address Extension Command (CMD22)

S	T	Index	Reserved	ADDR	CRC7	E
0	1	010110	all 0	xxxxxxb	xx	1

Extended Address (6-bit)

- **for Memory Commands (CMD17, CMD18, CMD24, CMD25, CMD32, CMD33)**
  - **6-bit ADDR is assigned to upper of 38-bit 512B unit address**
  - **Lower 32-bit address is provided by a memory access command**
- **for CMD20 Set New AU**
  - **6-bit ADDR is assigned to upper of 28-bit 512KB unit address**
  - **Lower 22-bit address is provided by CMD20 Set Free AU**

Figure 4-76 : CMD22 Address Extension Command

Figure 4-77 shows memory read/write command sequences. CMD22 shall always precede a memory read/write command. CMD23 shall precede CMD22.

Regarding to data transfer using a multi-block memory read/write command, setting data transfer length by using CMD23 is recommended rather than stopping by CMD12 with the expectation that card may be able to manage memory operation more efficiently by knowing data transfer length. Still stop by CMD12 is useful for data transfers with length more than the maximum data length supported by CMD23.
A multi-block data transfer is performed by incrementing 38-bit address with continuous data block access and it means a multi-block data transfer can span across 2TB boundary.
Any error related to address and length are checked at memory read/write command (CMD17/18/24/25) and ADDRESS_ERROR or OUT_OF_RANGE is indicated either R1 of the memory read/write command or R1 of the next command (ex., CMD12 or CMD13). CMD22 and CMD23 do not indicate any own error (they can still indicate error from a previous command).

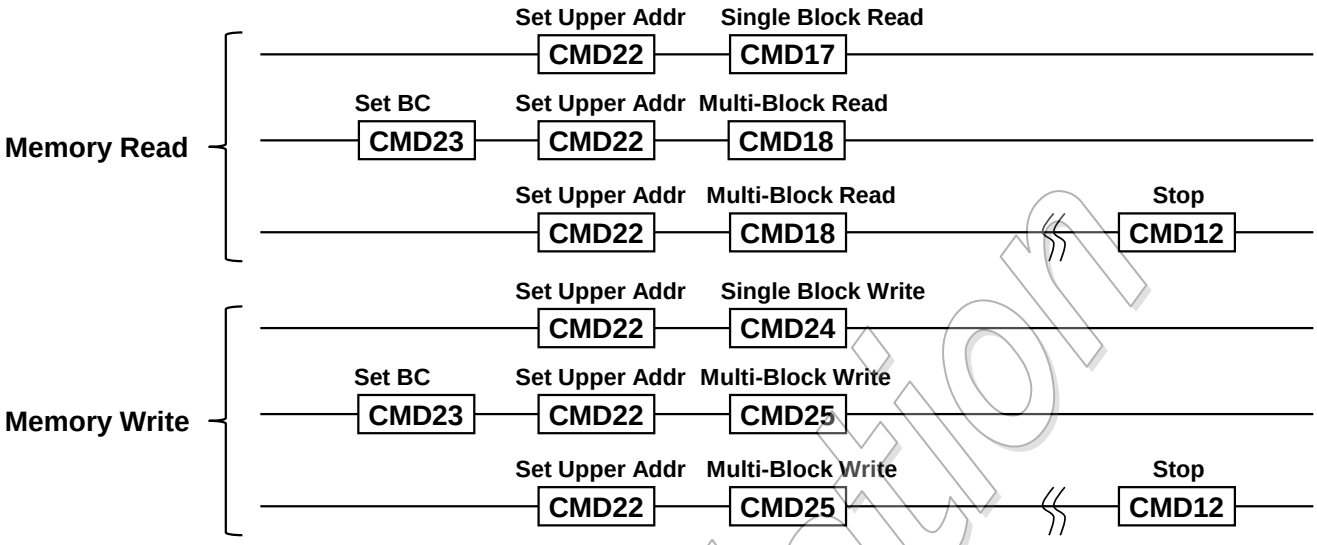


Figure 4-77 : Memory Command Sequences

Figure 4-78 shows an erase command sequence. CMD22 shall precede CMD32 and CMD33 to configure 38-bit erase start address and 38-bit erase end address. Any error related to address is checked at CMD38 and ADDRESS_ERROR is indicated either in R1 of CMD38 or in R1 of the next command (ex., CMD13). If any error occurs, CMD38 shall not perform erase operation (long busy is not indicated).

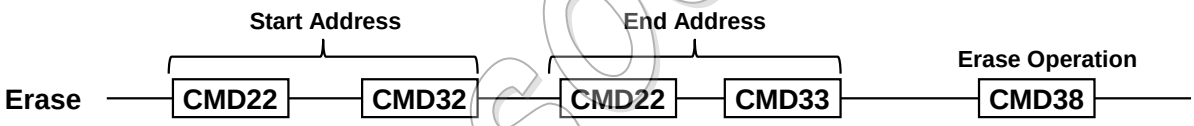


Figure 4-78 : Erase Command Sequences

4.20.4 Extension/Modification for Subsidiary Commands

4.20.4.1 Number of Written Block Command (ACMD22)

ACMD22 is used to verify previous write operation. SDUC Card returns a data block (64-bit data + CRC16). The 64-bit data shown in Figure 4-79 indicates the number of well written blocks by a previous write operation.

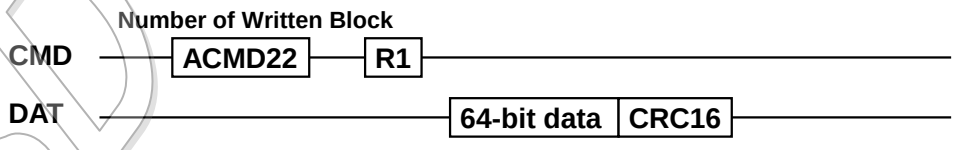


Figure 4-79 : ACMD22 Data Block for SDUC Card

4.20.4.2 ACMD23 Number of Write Blocks Pre-erased

SDUC Card does not support ACMD23.

There are two options of ACMD23 behavior:

1) SDUC Card does not respond to ACMD23 and indicates Illegal Command error in next R1
2) SDUC Card responds to ACMD23 but card shall not execute ACMD23 and no error is indicated to R1 of ACMD23 (previous command errors are indicated R1 of CMD55)

4.20.5 Extension for Video Speed Class

SDUC Card may support Speed Class, UHS Speed Grade and Video Speed Class (VSC) in SD mode or UHS-II mode as optional.

4.20.5.1 CMD20 "Start Recording" Command Sequence

Figure 4-80 shows CMD20 "Start Recording" Command Sequence of VSC. Use of CMD23 is recommended to perform sequential write. It is useful for card to verify alignment of RU Size, SU Size and AU Size.

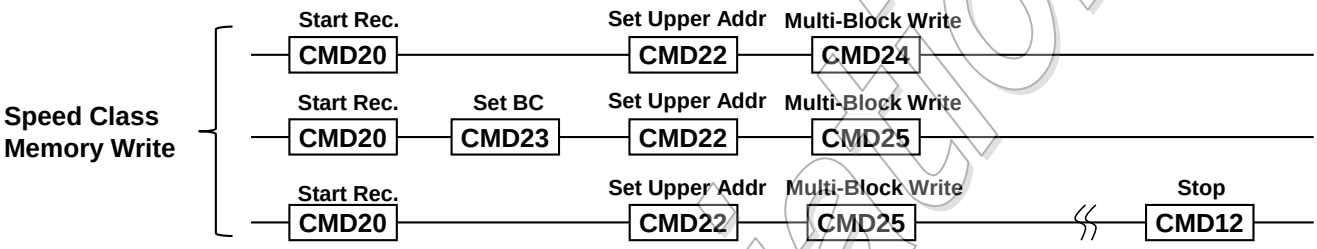


Figure 4-80 : CMD20 "Start Recording" Command Sequence

4.20.5.2 FAT Update Command Sequence

Figure 4-81 shows a typical FAT Update Command Sequence. Any order of three writes is allowed.

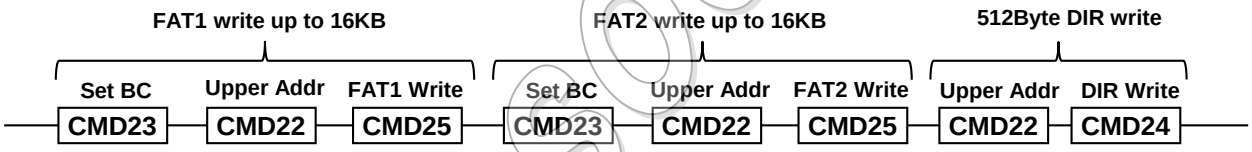


Figure 4-81 : FAT Update Command Sequence

4.20.5.3 CMD20 "DIR Update" Command Sequence

Figure 4-82 shows CMD20 "Update DIR" command sequence.

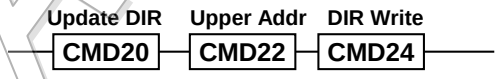


Figure 4-82 : CMD20 "Update DIR" Command Sequence

4.20.5.4 CMD20 "Set Free AU" Command Sequence

Figure 4-83 shows CMD20 "Set Free AU" Command Sequence. This sequence is different from other CMD20 sequences. CMD22 shall precede CMD20 "Set Free AU" to expand 28-bit ADDR of 512KB unit. 28-bit ADDR can designate the First Bundled AU in 128TB memory space. Following AUs (Second Bundled AU, Third Bundled AU, etc.) are designated by incrementing 28-bit ADDR and it means that Bundled AU may across 2TB boundary.

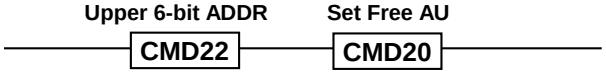


Figure 4-83 : CMD20 "Set Free AU"

4.20.5.5 Suspend Address Extension

Table 4-76 shows a part of SD Status around SUS_ADDR field. It is noted that 28-bit SUS_ADDR files are not contiguous and there is a boundary between Bit345 and Bit346. Bit346 is LSB of SUS_ADDR and Bit345 is MSB of SUS_ADDR.

Bits	Identifier	Type	Value	Description	Clear
377:368	VSC_AU_SIZE	S R	AU size in MB for Video Speed Class	(See Section 4.10.2.11)	A
367:346	SUS_ADDR [21:0]	S R	Suspension Address	(See Section 4.10.2.12)	A
345:340	SUS_ADDR [27:22]	S R	Suspension Address	(See Section 4.10.2.12)	A
339:336	APP_PERF_CLASS	S R	Application Performance Class Value of the card	(See Section 4.10.2.13)	A

Table 4-76 : SUS_ADDR Extension in SD Status

4.20.6 Command Sequence Reset for Error Recovery

If any error occurs, command sequence would be out of synchronization between host and card. SDUC Card utilizes CMD12 to reset command sequence. The behavior of CMD12 for SDUC Card is different from up to 2TB Card as CMD12 is not defined in tran state (CMD12 is illegal in tran) for up to 2TB Cards. Table 4-77 shows a part of Card State Transition Table that modifies a transition condition of CMD12 from tran to tran. This means that SDUC Card accept CMD12 in tran state and returns R1 response.

Commands	Current State									
	idle	ready	ident	stby	tran	data	rcv	prg	dis	ina
"Operation Complete"	-	-	-	-	-	tran	-	tran	stby	-
class 0										
CMD11	-	ready	-	-	-	-	-	-	-	-
CMD12	-	-	-	-	tran	tran	prg	-	-	-
CMD13	-	-	-	stby	tran	data	rcv	prg	dis	-

Table 4-77 : Modification of CMD12

By this modification, CMD12 can be used to reset command sequence. SDUC Card treats a command after CMD12 as the first command of a new command sequence. As it is conventionally done, CMD12 can be used for error recovery that makes a card to go back to tran state. Then this modification has compatibility to a legacy Host Driver processing.

4.21 Boot Functionalities

4.21.1 Boot Partition and Partition Selection

To protect boot code from intentional or unintentional modification by regular write/erase commands, it is stored apart from the conventional User Area in a dedicated Boot Partition.

Description here is a blank in the Simplified Specification.

When card supports boot functionalities, it shall have exactly two Boot Partitions called “Boot Partition 0” and “Boot Partition 1”. Capacity of these Boot Partitions shall be identical and can be from 128KB to 32MB

for each. Refer to Section 5.8.3 for the setting of the Extension Register related to Boot Partitions.
SD card can have exactly one User Area Partition.

Table 4-78 shows an assignment of Partition ID for each partition.

Partition ID	Partition Name
00h	Boot Partition 0
01h	Boot Partition 1
F0h	User Area Partition
Others	Reserved

Table 4-78 : Partition ID

To switch a target partition, host issues CMD39 (SELECT_CARD_PARTITION) described in Table 4-33. The response of CMD39 is R1b, and its maximum busy time is 1 second. When BOOT_PARTITION_SUPPORT=1, card shall support CMD39. CMD39 does not work in SPI mode.
After completing the card identification mode, User Area Partition is automatically (i.e. without CMD39) selected.
When card receives CMD39 designating undefined Partition ID in its argument, card does not respond and indicates OUT_OF_RANGE in Card Status.

Application Note:
Partition ID for User Area is defined as 0 in eSD Addendum. But in the Physical Layer Specification, different value F0h is assigned to User Area Partition.
Moreover, similar command for selecting partitions is defined as CMD43 in eSD Addendum. But since CMD43 is already assigned to other function in the Physical Layer Specification, partition selection is realized by other command (CMD39) in SD card.

4.21.2 Basic Access to Boot Partitions

This section describes how to access the Boot Partitions. Also refer to Section 4.21.4 for Boot Partition Protection.

4.21.2.1 Access to Boot Partitions by SD Interface

After power up, host executes process known as card identification mode (refer to Section 4.2). When finishing the card identification mode, the selected partition is User Area Partition as described in Section 4.21.1.
Then, host issues to CMD7 to make card state ‘tran’ and CMD39 to select either one of Boot Partition 0 or 1.
When Boot Partition 0 or 1 is selected, host is able to read data from these partitions by CMD17 or CMD18. On the other hand, if host intends to write data to or erase the selected Boot Partition, following sequence is required including RPMB authentication:

(1) If Boot Partition Protection is enabled, host unlocks the target Boot Partition by updating the corresponding bit of Boot Partition Lock through the RPMB authentication (refer to Table 4-83). It is done by “Authenticated Device Configuration Block write request” defined in RPMB sequence.
(2) After the target Boot Partition is unlocked, host writes a boot code by CMD24 or CMD25. Host may remove the current boot code by a series of CMD32, CMD33 and CMD38 at this time. When card receives CMD24, CMD25, CMD32, CMD33 or CMD38 while the target Boot Partition is locked, card shall handle them as illegal commands.
(3) After completing the boot code update, host may change Active Boot Partition in Extension Register by CMD49 (refer to Table 5-32). It is accepted only when either bit 0 or bit 1 of Boot Partition Lock

indicates '0' (refer to Table 4-83).
(4) Host changes the bit of target Boot Partition Lock to 1 to protect malicious boot code update, and switches to the User Area Partition by CMD39.

Access methods to Boot Partitions are the same as those to User Area. For example, when host accesses Boot Partitions of SDUC card, it shall use CMD22 the same way as when accessing User Area (refer to Section 4.20.3). Also refer to Section 4.22.2 for TCG supported SDUC card.

4.21.2.2 Access to Boot Partitions by PCIe Interface

Following description is a brief sequence for reading boot code from a Boot Partition by PCIe/NVMe interface:
(1) Host obtains which Boot Partition is active (BPINFO.ABPID) and the size of the Boot Partition (BPINFO.BPSZ).
(2) Host initializes the address (BPMBL.BMBBA) into the memory buffer where the contents should be copied, if necessary.
(3) Host initiates the transfer of data from a Boot Partition by writing to the Boot Partition Read Select (BPRSEL). It includes setting the Boot Partition identifier (BPRSEL.BPID), size of Boot Partition Read Size (BPRSEL.BPRSZ) and Boot Partition Read Offset (BPRSEL.BPROF).
(4) Host waits to completely transfer the requested portion of the Boot Partition, indicated in the status field (BPINFO.BRS).

Following description is a brief sequence for writing boot code to a Boot Partition by PCIe/NVMe interface:

(1) Host issues a Firmware Image Download command to download the contents of the Boot Partition.
(2) Host unlocks the target Boot Partition by updating RPMB Device Configuration Block.
(3) Host submits a Firmware Commit command with a Commit Action of 110b which specifies that the downloaded image replaces the contents of the Boot Partition specified in the Boot Partition ID field.
(4) Host locks Boot Partition access to prevent further modification.

Refer to NVMe specification for more details:

4.21.3 Fast Boot

4.21.3.1 Overview of Fast Boot

Fast Boot realizes that boot data in SD card is transmitted immediately after power up with minimum operations. Fast Boot is available over SD interface including DS, HS and UHS-I modes (UHS-I mode is either SDR12, SDR25, SDR50, SDR104 or DDR50). Note that it is not applicable to SPI mode, UHS-II (UHS-III) mode and PCIe mode. In addition, when the bus mode of Fast Boot is either SDR50 or SDR104, forty tuning blocks are transmitted from the card prior to the boot code in order that host can determine appropriate sampling point (i.e. execute tuning). The tuning block is same as that defined in Table 4-3 Figure 4-11.
There are two methods called CV-mode (standing for CMD line Voltage) and CA-mode (standing for CMD0 Argument) for Fast Boot in SD card.
By CV-mode, host starts Fast Boot by driving CMD line low for greater than or equal to 74 clocks after power up. The bus mode is determined by the value of "Preprogrammed bus mode for Fast Boot" in the Extension Register in card side, and host is expected to operate the card using the same bus mode.
By CA-mode, on the other hand, host issues CMD0 with a special argument for executing Fast Boot. The bus mode is selected according to the argument in CMD0. In addition, if the selected bus mode is based on 1.8V signaling mode (in other words, UHS-I mode), host shall execute the LVS Identification Sequence described in Part 1 Low Voltage Interface Addendum.

4.21.3.2 CV-mode Fast Boot

This section is a blank in the Simplified Specification.

4.21.3.3 CA-mode Fast Boot

This section explains details of CA-mode Fast Boot sequences. In CA-mode, the bus mode for Fast Boot is specified by the argument of CMD0 defined in Table 4-79. Note that these special arguments of CMD0 are only valid in 'idle' state. If card receives these arguments in other than 'idle' state, it regards as 00000000h.

Argument	Description	Notes
00000000h	(Handled as conventional CMD0)	
F1F1F1F1h	DS mode	When card receives this argument in 1.8V signaling mode, the argument is regarded as 00000000h.
F2F2F2F2h	HS mode	(Same as the above)
F8F8F8F8h	Preprogrammed mode stored in "Preprogrammed bus mode for Fast Boot" of the Extension Register	When card receives this argument if the current signaling voltage does not match to that corresponding to the bus mode in the register, the argument is regarded as 00000000h.
F9F9F9F9h	SDR12 mode	When card receives this argument in 3.3V signaling mode, the argument is regarded as 00000000h.
FAFAFAFAh	SDR25 mode	(Same as the above)
FBFBFBFBh	SDR50 mode	(Same as the above)
FCFCFCFCh	SDR104 mode	(Same as the above)
FDFDFDFDh	DDR50 mode	(Same as the above)
Others	Reserved	The argument is regarded as 00000000h.

Table 4-79 : Special Arguments of CMD0 for Bus Mode in CA-mode Fast Boot

When card receives CMD0 with a valid special argument, that is, the argument is not equal to and is not regarded as 00000000h, card shall change data bus mode from 1 bit to 4 bits. In case of other arguments, the data bus width is changed to 1 bit as described in Section 4.3.1.

Description here is a blank in the Simplified Specification.

4.21.3.4 Modification of Tuning Blocks Transmission during Fast Boot

Before fetching boot code data by Fast Boot over SDR104 or SDR50, host should execute tuning to determine the appropriate sampling point as described in Section 4.2.4.5. When SDR104 or SDR50 is selected, card continuously transmits tuning blocks 40 times with block gaps prior to boot code data as described in Figure 4-92.

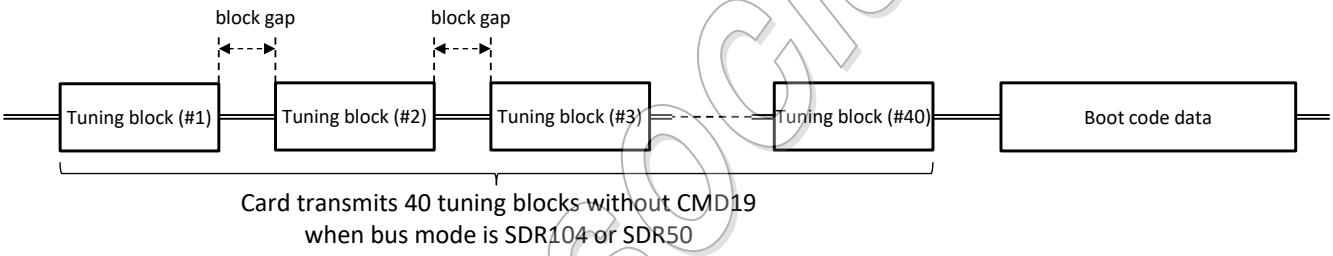


Figure 4-92 : Transmission of Tuning Blocks during Fast Boot

Different from the UHS-I initialization in Figure 4-6, host needs to detect exactly 40 tuning blocks without issuing CMD19. If host drops one or more tuning blocks, host may regard a part of actual boot code data as a tuning block. This means host fails to obtain proper boot code as a result.

Description here is a blank in the Simplified Specification.

4.21.4 Boot Partition Protection Using RPMB

SD card supporting Boot Partitions shall also support RPMB. Such card requires Boot Partition Protection which is configured by using RPMB (refer to Section 4.23). Figure 4-95 shows an overview of Boot Partition Protection.

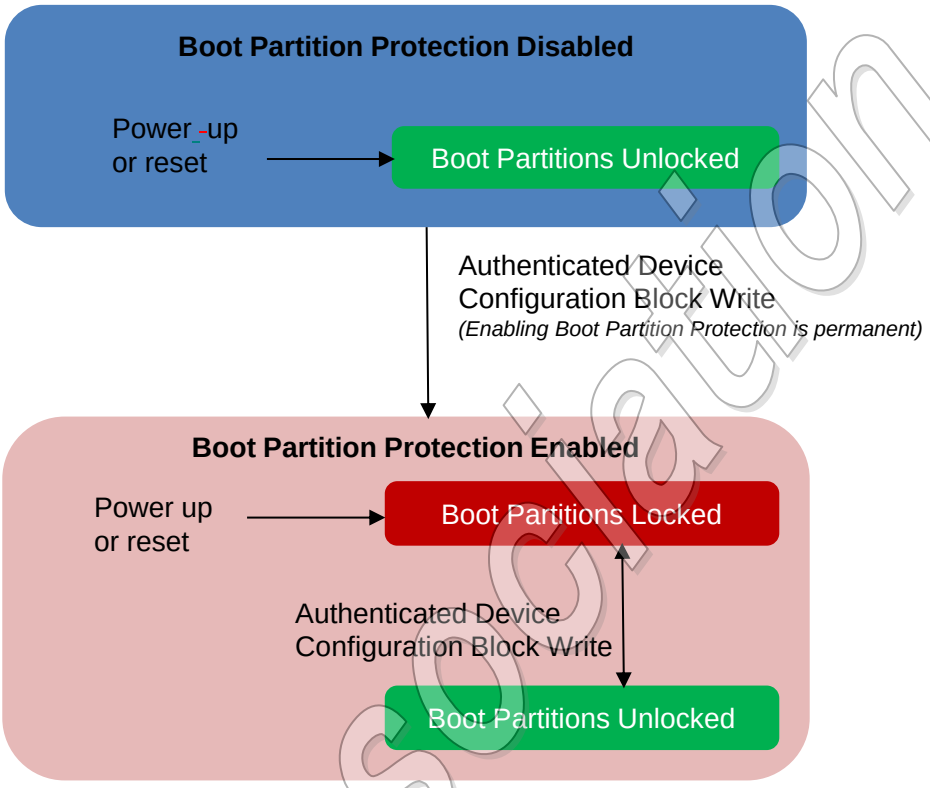


Figure 4-95 : Boot Partition Lock/Unlock State

The default state of boot partition is the “unlocked” state. In this state, host software may read and write to Boot Partitions without RPMB authentication. Active Boot Partition register can be modified without RPMB authentication. All Boot Partitions remain unlocked until Boot Partition Protection is enabled by host. Host enables Boot Partition Protection by setting the Boot Partition Protection Enable bit in the RPMB Device Configuration Block data structure (refer to Section 4.23.1.1). Once Boot Partition Protection is enabled, the controller shall reject Authenticated Device Configuration Block Writes that disable Boot Partition Protection (i.e., enabling Boot Partition Protection is permanent). Once Boot Partition Protection is enabled, Boot Partitions can be modified only after unlocking the Boot Partition using RPMB. After activating Boot Partition Protection, the default state for all Boot Partitions is the “Locked” state. In this state, host software may read a Boot Partition. In this state, the controller rejects attempts to write or erase to a Boot Partition using CMD24 or CMD25. Each Boot Partition may be locked or unlocked independently using the corresponding bit in the Device Configuration Block data structure.

4.21.5 Pre-init mode

Considering LVS Identification and Fast Boot, pre-init mode is introduced.
Table 4-81 shows card states and operation modes including pre-init mode In this mode, p-init state and boot state are introduced as a card state. These states are not defined in CURRENT_STATE of Card Status. Note that pre-init mode is applicable only to SD interface.

Card state	Operation mode
Inactive State	inactive
P-init State	pre-init mode
Boot State	
Idle State	card identification mode
Ready State	
Identification State	
Stand-by State	data transfer mode
Transfer State	
Sending-data State	
Receive-data State	
Programming State	
Disconnect State	

Table 4-81 : Overview of Card States vs. Operation Modes Considering LVS and Fast Boot

Figure 4-96 illustrates a state diagram of pre-init mode.

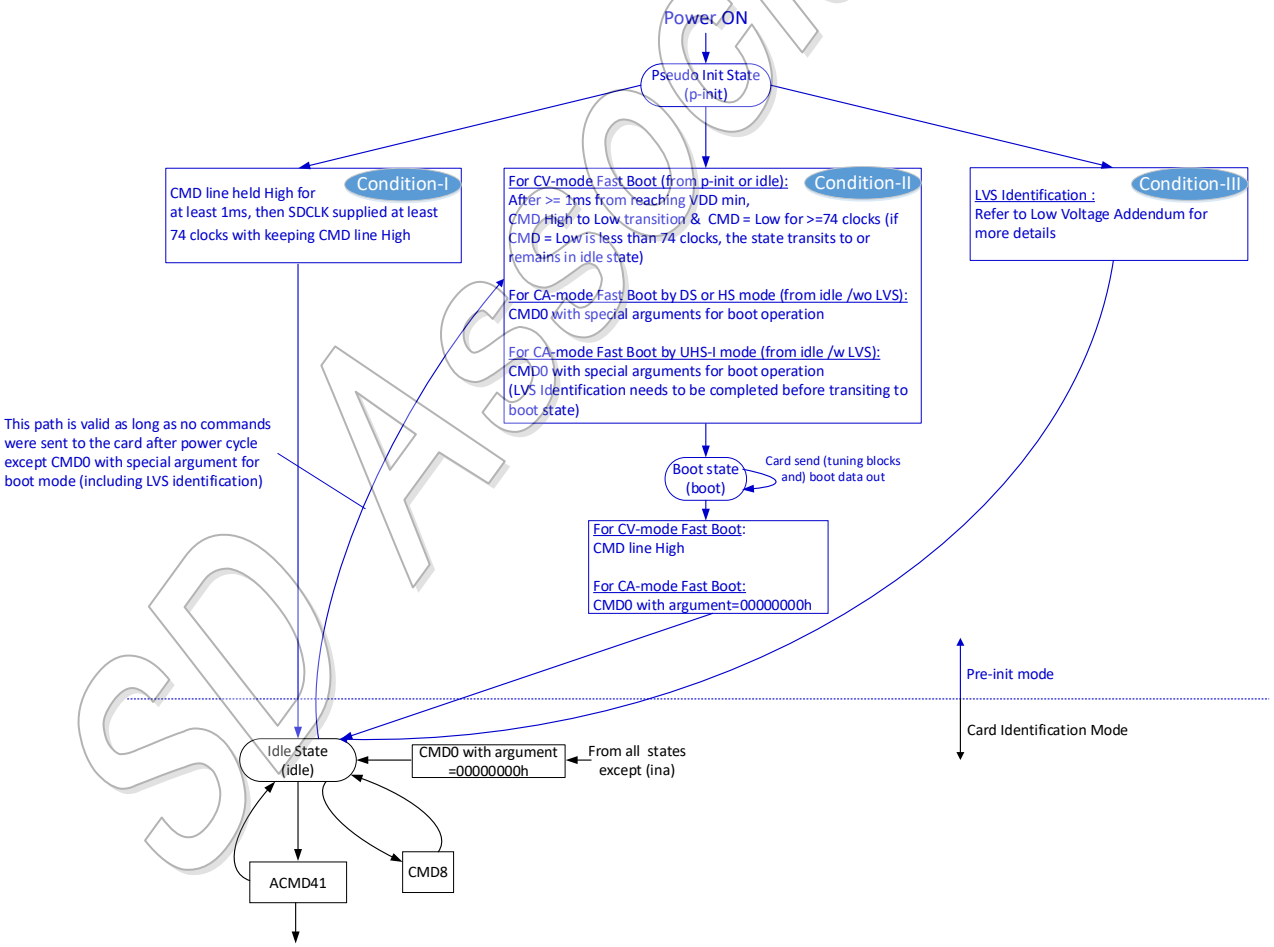


Figure 4-96 : State Diagram (Pre-init Mode)

The state ‘p-init’ is introduced for indicating state just after power up and branching to either one of Condition-I, Condition-II or Condition-III. Host cannot issue any commands in p-init state.
Condition-I indicates the conventional transition to idle that card can receive commands. This condition is based on the power-up diagram described in Figure 6-4.
Condition-II indicates ones for starting Fast Boot by either CV-mode or CA-mode and transiting to boot state.
Condition-III indicates LVS Identification to utilize Low Voltage Interface (without Fast Boot) or to perform CA-mode Fast Boot over UHS-I mode.
When host intends to initiate CV-mode Fast Boot, the host shall drive CMD line low for more than or equal to 74 clocks to satisfy Condition-II.
Depending on the timing of host operation, the state may transit from ‘p-init’ to ‘idle’ because Condition-I is satisfied first, as described in Figure 4-88.
When host intends to execute CA-mode Fast Boot over UHS-I mode, it needs performing LVS Identification described in Condition-III and issuing CMD0 with a special argument. To keep consistency with the existing specification, state transits to the idle state after completing LVS Identification. So transition from ‘idle’ to ‘boot’ is also applied to CA-mode Fast Boot. In addition, CV-mode Fast Boot is not allowed after executing LVS Identification.
When transiting to boot state, partition is automatically switched to the boot partition indicated by “Active Boot Partition” in Extension Register. Additionally, User Area Partition is automatically selected after completing card identification mode as described in Section 4.21.1.
In boot state, card transmits tuning blocks (limited when the bus mode is either SDR50 or SDR104) and boot code data. After finishing boot code data transmission, host drives CMD line high (CV-mode) or issues CMD0 with argument=00000000h (CA-mode) to transit to idle state. These operations can be executed before or on the way of tuning blocks or boot code data transmission. In this case, Fast Boot is suspended and state transits to ‘idle’.
In addition, when card receives a command except CMD0 in boot state, card treats it as an illegal command.

4.22 TCG Security

4.22.1 Protocols over SD interface for TCG Security

To realize TCG security protocols over SD interface including commands or payload transfer, ACMD53 and ACMD54 are introduced. ACMD53 and ACMD54 correspond to IF-RECV and IF-SEND respectively, which are defined in TCG Storage Architecture Core Specification.
Also refer to Part 1 Extended Security Addendum and TCG Storage Interface Interactions Specification (SIIS) for details of TCG security protocols over SD interface.
Note that when host accesses TCG supported SD Express card over PCIe mode, refer to NVM Express Specification. Once TCG is enabled in SD Express card, card access requires TCG security protocol for both SD interface and PCIe interface.

4.22.2 MBR Shadowing for SD Card

As described in Part 1 Extended Security Addendum, SD card with TCG security shall support MBR Shadowing. Basic specifications of MBR shadowing are referred to TCG Storage Architecture Core Specification.
Here are specific rules of MBR Shadowing for SD card.

4.22.2.1 TCG MBR Table

TCG MBR Table is introduced to store code to be processed after power cycle, including pre-boot authentication program. It is identical to “MBR Table” defined in the TCG Storage Architecture Core Specification. The size of TCG MBR Table is fixed to 128MB in SD card. TCG MBR Table content is returned by the card when User Area LBAs from 00000000h to 0003FFFFh are read and is accessible

only when MBR Shadowing is enabled and pre-boot authentication is not completed. TCG MBR Table content is not affected by Force Erase operations of Section 4.3.7.3.

4.22.2.2 MBRControl Table

MBR Shadowing is controlled by MBRControl Table (see Table 4-82). This table can be accessed by IF-RECV command (ACMD53) or IF-SEND command (ACMD54).

SD Cards supporting MBR Shadowing shall reset successful pre-boot authentication after power cycle only. Thus, MBRDoneOnReset shall always include a set of single value 0 for SD card, and cannot be over-written by host. Refer to Table 4-82 and TCG Storage Architecture Core Specification for more details.

Column Number	Column Name	Column Type	Description
00h	UID	uid	Unique identifier of this table defined in “TCG Storage Architecture Core Specification”.
01h	Enable	boolean	When MBR shadowing is enabled, this column is set to 1 (True). Otherwise, set to 0 (False).
02h	Done	boolean	When pre-boot authentication is not succeeded, this column is set to 0 (False). Otherwise, it is set to 1 (True). Note that this column is automatically set to 0 (False) when power cycle occurs.
03h	MBRDoneOnReset	reset_type	It is preset to {0} (a set including value 0 only as an element, and indicating power cycle only) for SD card. This means “Done” is set to 0 (False) automatically only when power cycle occurs.

Table 4-82 : MBRControl Table

4.22.2.3 Pre-boot Authentication Sequence over SD Interface

Figure 4-97 shows a sequence diagram of pre-boot authentication over SD interface. In this sequence, “MBRControl Enable” is supposed to be 1b and host recognizes that.

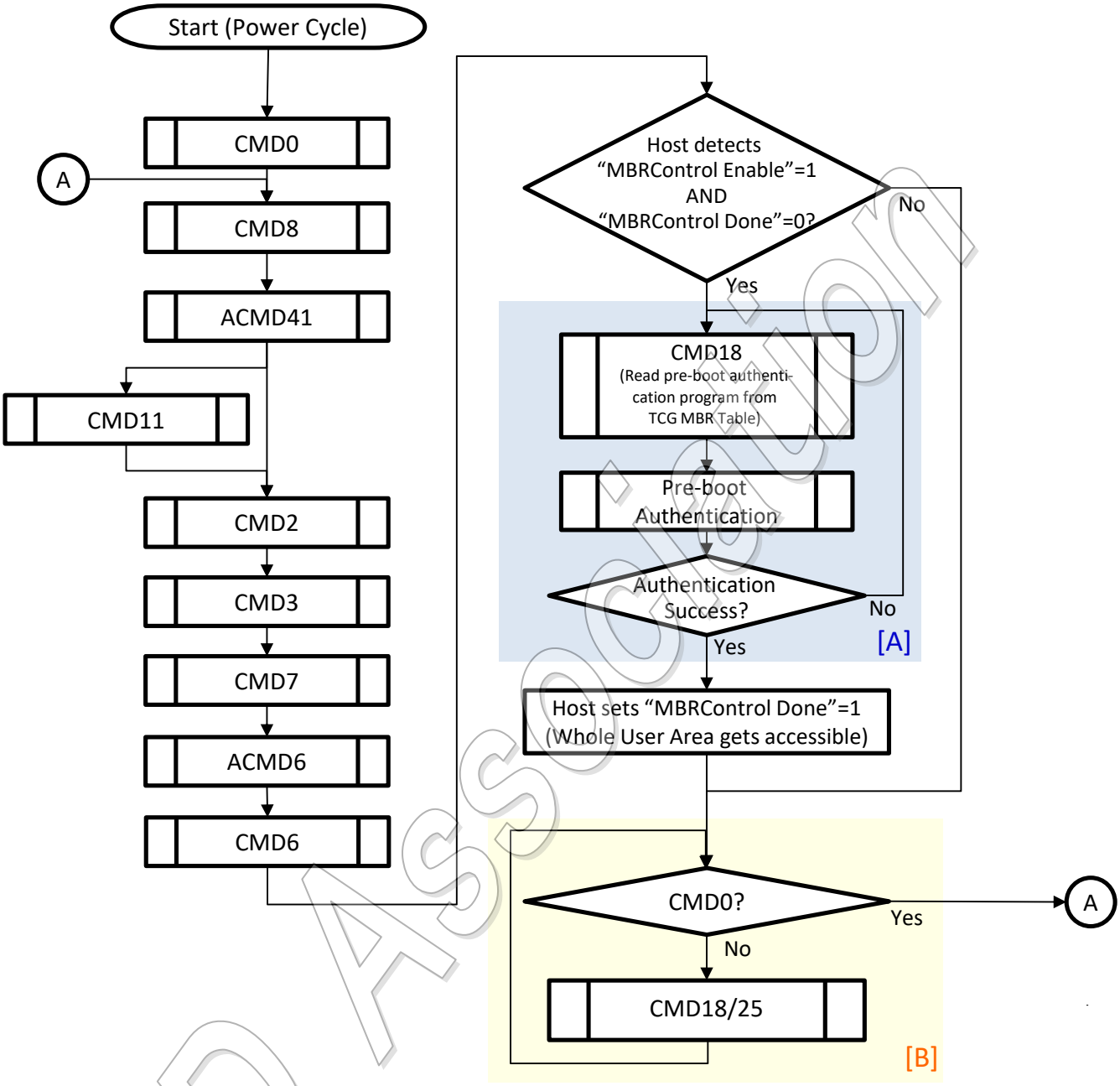


Figure 4-97 : Pre-boot Authentication Sequence over SD interface

At power cycle, "MBRControl Done" is reset to 0b since "MBRControl MBRDoneOnReset" is set to {0}. This means status of pre-boot authentication is reset by the power cycle.
When host issues ACMD41, host may set S18R=1 in its argument to check card's UHS-I availability. If card responds S18A=1 in R3, host may issue CMD11 to use UHS-I bus after ACMD41 operation.

After issuing CMD2, CMD3, CMD7 and ACMD6 to make data transmission over 4 bits mode available, host issues CMD6 to switch a bus mode including UHS-I.
If host detects “MBRControl Done” is 0, it cannot obtain proper data from User Area because card supports TCG security and the pre-boot authentication is not completed yet. In this case, host reads pre-boot authentication program from the TCG MBR Table and executes pre-boot authentication. If the authentication succeeds, host sets the value of “MBRControl Done” to 1b. After that, host can read/write data from/to the User Area as a normal manner. Host is not necessary to issue CMD0 after completing pre-boot authentication. Yet, if host issues CMD0 at this time, host needs card identification again from CMD8. However, pre-boot authentication is not required to access the User Area because “MBRControl Done” is not reset and kept as 1b.

4.23 Replay Protected Memory Block

4.23.1 Introduction

The Replay Protected Memory Block (RPMB) provides a mechanism for the system to store data to a specific memory area in an authenticated and replay protected manner. This is provided by first programming authentication key information to the SD card that is used as shared secret. The system is not yet authenticated in this phase, therefore the authentication key programming should be done in a secure environment. (e.g., as a part of the manufacturing process). The authentication key is utilized to sign read and write accesses made to the replay protected memory area with Message Authentication Code (MAC). Use of Random Number (nonce) generation and write count register provide additional protection against replay of messages where messages could be recorded and played back later by an attacker.

RPMB feature may be supported through SD interface. SD Express Card supporting RPMB feature shall support it through both SD and PCIe interface. The usage of RPMB through PCIe interface is done over NVMe protocol.

The specific RPMB memory area is referred the RPMB unit.
Description here is a blank in the Simplified Specification.

Note that the term RPMB Target and RPMB Unit are in fact the same and they are both used hereafter interchangeably:

a. RPMB Unit is not part of the User area (or Namespace in case of NVMe).
b. The Force Erase operation, Password Lock/Unlock and Write Protection is not applicable to RPMB Unit.
c. At most one RPMB target may be supported.
d. RPMB feature may be supported through PCIe interface in SD Express card using Security Send and Security Receive commands as defined in NVMe specification.
e. RPMB feature may be supported through SD interface using SECURE_SEND and SECURE_RECEIVE commands.
f. RPMB feature discovery may be done as following:
    i. Through the SD Interface – reading EX_SECURITY field in SCR register (refer to Section 5.6) and Security and Boot Register Set (refer to Table 5-32).
    ii. Through the PCIe/NVMe interface – using the Identify Controller Data Structure as defined in NVMe Specification.

Following subsections describe RPMB implementation over SD interface in detail including RPMB Device

Configuration Block Data Structure, RPMB data frame and RPMB operations. One can refer to NVMe specification for RPMB implementation over PCIe interface.

4.23.1.1 RPMB Device Configuration Block

Table 4-83 defines the RPMB Device Configuration Block Data structure - the non-volatile contents are stored within the SD card for RPMB unit / RPMB target. Device configuration block data structure field are defined in Security and Boot Register Set (refer to Table 5-32).

Bytes	Component Name	Description
00	Boot partition Protection Enable	This field indicates if Boot Partition Protection is enabled Bits 7:1 are reserved Bit 0: A value of '1' indicates Boot Partition Protection is enabled. A value of '0' indicates Boot Partition Protection is disabled or not supported. Once enabled, the SD card shall prevent disabling Boot Partition Protection
01	Boot Partition Lock	This filed indicates the current status of the Boot Partition Lock/Protection using RPMB. This Field shall be cleared to 0h unless Boot Partition Protection is enabled (refer to Section 4.21.4) Bits 7:2 are reserved. Bit 1: A value of '1' indicates Boot Partition 1 is locked or protected using RPMB A value of '0' indicates Boot Partition 1 is unlocked or not protected using RPMB. Bit 0: A value of '1' indicates Boot Partition 0 is locked or protected using RPMB. A value of '0' indicates Boot Partition 0 is unlocked or not protected using RPMB
02	User Area Write Protection Authentication Control	This field specifies whether the SD Card processes or aborts modification of Permanent Write Protect and Write Protect Until Power Cycle. Bits 1:0 of this field shall be cleared to 00b after a power cycle. Bits 7:2 are reserved. Bit 1: A value '0' indicates that the controller shall fail the command which attempts to set the Permanent Write Protect. A value '1' indicates that the controller shall process the command which attempts to set the Permanent Write Protect. Bit 0: A value '0' indicates that the controller shall fail the command which attempts to modify Write Protect Until Power Cycle. A value '1' indicates that the controller shall process the command which modify Write Protect Until Power Cycle.
511:03	--	Reserved

Table 4-83 : RPMB Device Configuration Block Data Structure

4.23.1.2 RPMB Contents

Table 4-84 defines the non-volatile contents stored within the SD card for RPMB target.

Content	Type	Size	Description
Authentication key	Write Once, Not erasable or readable	Size is dependent on authentication method reported in Security and Boot Register Set (Refer to Table 5-32)	Authentication key which is used to authenticate accesses when MAC is calculated
Write Counter	Read Only	4 bytes	Counter value for the total amount of successful authenticated data write requests made by the host. The initial value of this register after manufacture is 00000000h. The value is incremented by one automatically by the controller with each successful programming access. The value is not resettable. After the counter has reached maximum value of FFFFFFFFh, the controller shall no longer increment to prevent overflow.
RPMB Data Area	Readable and Writable, not erasable	Size is reported in Security and Boot Register Set (128KB minimum, 32MB maximum. Refer to Table 5-32)	Data that is able to be read and written only via successfully authenticated read/write access. This data can be overwritten by host but can never be erased.

Table 4-84 : RPMB Contents

4.23.1.3 RPMB Data Frame

Each RPMB Data Frame is 256 bytes in size plus the size of the Data Field, and is organized as shown in Table 4-85. RPMB uses sector size of 512 bytes. The RPMB sector size is independent and not related to the logical block size used for the user area / Namespace.

Bytes	Component Name	Description
222-N:00	Stuff Bytes	Padding for the frame. Values in this field are not part of the MAC calculation. The size is 223 bytes minus the size of the Authentication Key(N)
222:222-(N-1)	Authentication Key or Message Authentication Code(MAC)	Size is dependent authentication method reported in Security and Boot Register Set (refer to Table 5-32) (e.g., SHA-256 key is 32 bytes (refer to RFC 6234))
223	RPMB Target	Indicates Request/Response for RPMB target. If the value in this field is not equal to the SD Security Specific Field (SSSF) in the SECURE_SEND and SECURE_RECEIVE command, then the SD card shall return an error of Invalid Field in Command for the SECURE_SEND and SECURE_RECEIVE command
239:224	Nonce	Random number generated by the host for the requests and copied to the response by the RPMB target
243:240	Write Counter	Total amount of successfully authenticated data write requests.
247:244	Address	Starting address of data to be programmed to or read from the RPMB.

251:248	Sector count	Number of sectors (512 bytes) requested to be read or written.
253:252	Result	Defined in Table 4-87. **Note**: The Result field is not needed for Request.
255:254	Request/Response Message	Defined in Table 4-86.
(M-1)+256:256	Data (Optional)	Data to be written or read by signed access where M = 512 * Sector count

Table 4-85 : RPMB Data Frame

SECURE_SEND and SECURE_RECEIVE commands are used to encapsulate and deliver data packets of any security protocol between the host and SD card without interpreting, dis-assembling or re-assembling the data packets for delivery. SECURE_SEND and SECURE_RECEIVE commands used for RPMB access are populated with the RPMB Data Frame(s) defined in Table 4-85. The controller shall not return successful completion of a SECURE_SEND or SECURE_RECEIVE command for RPMB access until the requested RPMB Request/Response Message Type indicated is completed. The Security Protocol used for RPMB is defined in Section 4.23.4.

4.23.1.4 RPMB Request and Response Message Types

RPMB Data frame supports different types of Request/Response messages between the host and RPMB target. Request and Response Message types are indicated by 2bytes long Request/Response message field in RPMB Data frame. All message types supported are defined in Table 4-86. Request message types are sent from the host to the SD card. Response message types are sent to the host from SD card.

Request/Response Message Types		Description	Requires Data	RPMB Frame Length (bytes)
0001h	Authentication Key programming request	The host is attempting to program the Authentication Key for the RPMB Target to the controller	No	256
0002h	Reading of the Write Counter value request	The host is requesting to read the current Write Counter value from the RPMB target	No	256
0003h	Authenticated data write request	The host is attempting to write data to the RPMB target	Yes	M + 256
0004h	Authenticated data read request	The host is attempting to read data from the RPMB target	No	256
0005h	Result read Request	The host is attempting to read the result code for any of the other Message Types	No	256
0006h	Authenticated Device Configuration Block Write request	The host is attempting to write Device configuration block (DCB) to the RPMB target.	Yes	512 + 256

0007h	Authenticate Device Configuration Block read request	The host is attempting to read Device Configuration Block (DCB) from the RPMB target.	No	256
0100h	Authentication key programming response	Returned as a result of the host requesting a Result read request Message Type after programming the Authentication key	No	256
0200h	Reading of the Write Counter value response	Returned as a result of the host requesting a Result read request Message type after requesting the Write Counter value	No	256
0300h	Authenticated data write response	Returned as a result of the host requesting a Result read request Message Type after attempting to write data to RPMB target	No	256
0400h	Authenticated data read response	Returned as a result of the host requesting a Result read request Message Type after attempting to read data from RPMB target	Yes	M + 256
0500h	Reserved			
0600h	Authenticated Device Configuration data write response	Returned as a result of the host requesting the Result read Message Type after attempting to write a Device Configuration Block to an RPMB target	No	256
0700h	Authenticated Device Configuration Data Read response	Returned as a result of the host requesting a Result read request Message Type after attempting to read DCB from RPMB target	Yes	512 + 256

Table 4-86 : RPMB Request and Response Message Types

4.23.1.5 RPMB Operation Result

RPMB Data frame has 2 bytes long Result field (refer to Table 4-85), which will contain RPMB Operation Result for Response message types. The structure and possible values of RPMB Operation Result are defined in Table 4-87.

Bits	Description
15:08	Reserved
07	**Write Counter Status:** Indicates if the Write Counter has expired (i.e., reached its maximum value). A value of ‘1’ indicates that the Write Counter has expired. A value of ‘0’ indicates a valid Write Counter.

<table>
<tr><td>06:00</td><td>Operation Status: Indicates the operation status. Valid operation status values are listed below
<table>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>00h</td><td>Operation successful</td></tr>
<tr><td>01h</td><td>General Failure</td></tr>
<tr><td>02h</td><td>Authentication Failure (MAC comparison not matching, MAC calculation failure</td></tr>
<tr><td>03h</td><td>Counter failure (counters are not matching in comparison , counter incrementing failure)</td></tr>
<tr><td>04h</td><td>Address failure (address out of range , wrong address alignment)</td></tr>
<tr><td>05h</td><td>Write failure (data/counter/result write failure)</td></tr>
<tr><td>06h</td><td>Read failure (data/counter/result read failure)</td></tr>
<tr><td>07h</td><td>Authentication Key not yet programmed. This value is the only valid Result value until the Authentication Key has been programmed. Once the key is programmed, the Result value shall no longer be used</td></tr>
<tr><td>08h</td><td>Invalid RPMB Device Configuration Block – This may be used when the target is not 0h</td></tr>
<tr><td>09h to 7Fh</td><td>Reserved</td></tr>
</table></td></tr>
</table>

Table 4-87 : RPMB Operation Result

4.23.2 Authentication Method

SD card with RPMB shall support one of the Authentication Method as indicated in the Security and Boot Register Set (refer to Table 5-32).

Description here is a blank in the Simplified Specification.

4.23.3 RPMB Operations

The host sends a Request Message type to the SD card to request an operation by the SD Card or to deliver data to be written into the RPMB memory block. To deliver a Request Message Type, the host uses the SECURE_SEND command. If the data to be delivered to the SD card is more than reported in Security and Boot Register Set (refer to Table 5-32), the host sends multiple SECURE_SEND commands to transfer the entire data.

The host sends request using a Response Message Type to the controller to read the result of a previous operation request, to read the Write Counter, or to read data from RPMB memory block. To deliver a Response Message Type, the host uses the SECURE_RECEIVE command. If the data to be read from the SD card is more than reported in Security and Boot Register Set (refer to Table 5-32), the host sends multiple SECURE_RECEIVE commands to transfer the entire data.

Figure 4-99 shows the SEURE_SEND and SECURE_RECEIVE operations of RPMB request and response message without RPMB Data frames - containing 256 bytes RPMB header and remaining 256 bytes padded with zero.

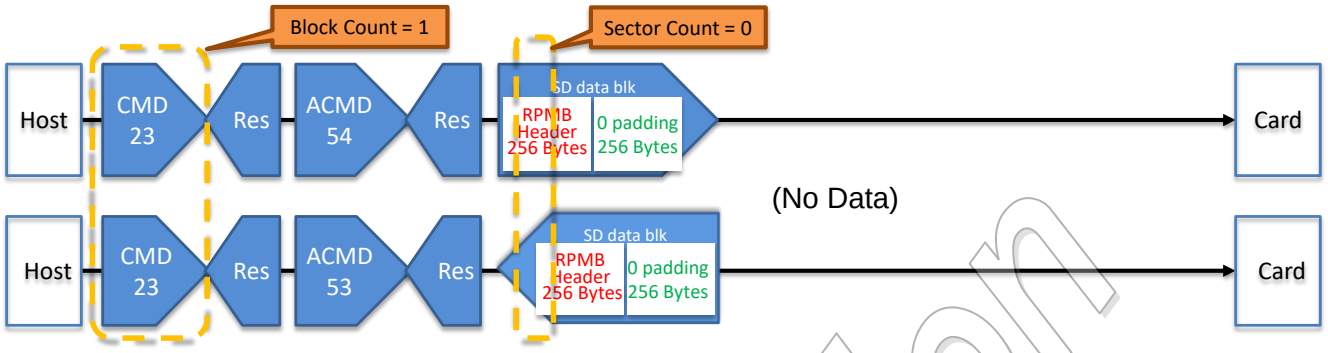


Figure 4-99 : RPMB Messages Transfer without Data

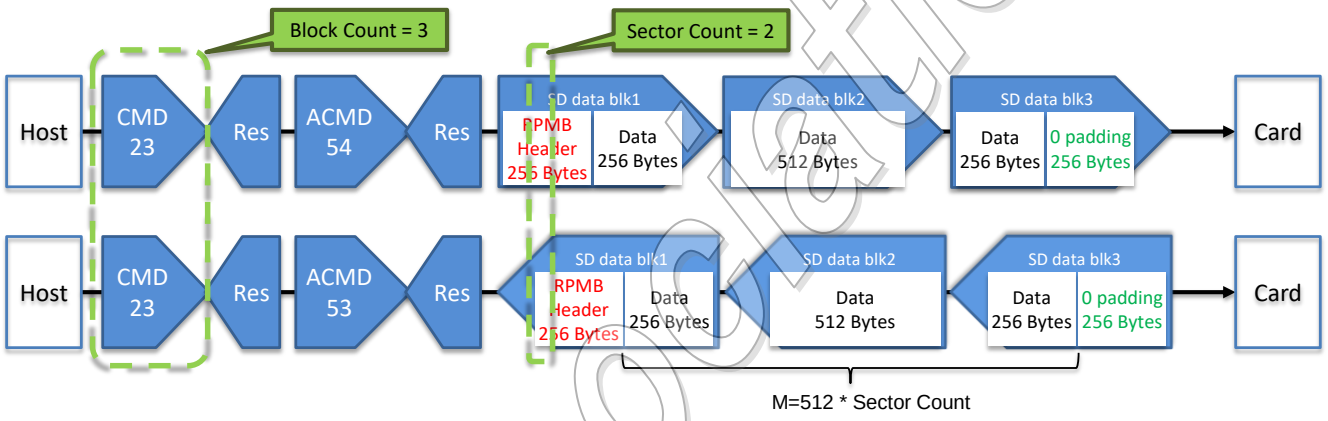


Figure 4-100 : RPMB Messages Transfer with Data

Figure 4-100 shows an example of SECURE_SEND and SECURE_RECEIVE operations with RPMB request and response messages associated with RPMB data frames of two sectors - containing 256 bytes RPMB header and 1024 bytes of data and 256 bytes of 0 padding. CMD23 will be send before SECURE_SEND and SECURE_RECEIVE command. The Block count in CMD23 will be one more than the sector count in RPMB header because the Sector Count in the RPMB header reflects the total number of sectors requested to be written or read while the Block Count in CMD23 reflects the total number of physical blocks send with the ACMD53 or ACMD54.

4.23.3.1 Authentication Key Programming

Authentication key programming is initiated by a SECURE_SEND command to program the Authentication Key to the RPMB target, followed by a subsequent SECURE_SEND command to request the result, and lastly, the host issues a SECURE_RECEIVE command to retrieve the result. RPMB authentication key data flow is shown in Table 4-89.

Command	Bytes in Command	Field Name	Value	Objective
SECURE_SEND 1	**Data populated by the host and sent to the controller**			Send Authentication Key to be Programmed to the SD card
	222-*N*:00	Stuff Bytes	0…00h	
	222:222-(*N*-1)	MAC/Key	Key to be programmed	
	223	RPMB Target	RPMB target to access (00h)	
	239:224	Nonce	0…00h	

	243:240	Write Counter	00000000h	
	247:244	Address	00000000h	
	251:248	Sector Count	00000000h	
	253:252	Result	0000h	
	255:254	Request/Response	0001h (*Request*)	
SECURE_SEND 2	**Data populated by the host and sent to the controller**			Request Result of Key Programming
	222-*N*:00	Stuff Bytes	0…00h	
	222:222-(*N*-1)	MAC/Key	0…00h	
	223	RPMB Target	RPMB target to access (00h)	
	239:224	Nonce	0…00h	
	243:240	Write Counter	00000000h	
	247:244	Address	00000000h	
	251:248	Sector Count	00000000h	
	253:252	Result	0000h	
	255:254	Request/Response	0005h (*Request*)	
SECURE_RECEIVE 1	**Data populated by the controller and returned to the host**			Retrieve the Key Programming Result
	222-*N*:00	Stuff Bytes	0…00h	
	222:222-(*N*-1)	MAC/Key	0…00h	
	223	RPMB Target	RPMB target response was sent from (00h)	
	239:224	Nonce	0…00h	
	243:240	Write Counter	00000000h	
	247:244	Address	00000000h	
	251:248	Sector Count	00000000h	
	253:252	Result	Result Code	
	255:254	Request/Response	0100h (Response)	

Table 4-89 : RPMB – Authentication Key Data Flow

4.23.3.2 Read Write Counter Value

The Read Write Counter sequence is initiated by a SECURE_SEND command to request the Write Counter value, followed by a SECURE_RECEIVE command to retrieve the Write Counter result. RPMB Read Write counter value flow is shown in Table 4-90.

Command	Bytes in Command	Field Name	Value	Objective
SECURE_SEND 1	**Data populated by the host and sent to the controller**			Request Write Counter Read
	222-*N*:00	Stuff Bytes	0…00h	
	222:222-(*N*-1)	MAC/Key	0…00h	
	223	RPMB Target	RPMB target to access (00h)	
	239:224	Nonce	Nonce generated by the host	
	243:240	Write Counter	00000000h	
	247:244	Address	00000000h	
	251:248	Sector Count	00000000h	
	253:252	Result	0000h	
	255:254	Request/Response	0002h (*Request*)	
SECURE_RECEIVE 1	**Data populated by the controller and returned to the host**			Retrieve Write Counter Read Result
	222-*N*:00	Stuff Bytes	0…00h	
	222:222-(*N*-1)	MAC/Key	MAC generated by the controller	
	223	RPMB Target	RPMB target response was sent from (00h)	
	239:224	Nonce	Copy of the Nonce generated by the host	
	243:240	Write Counter	Current Write Counter value	
	247:244	Address	00000000h	
	251:248	Sector Count	00000000h	
	253:252	Result	Result Code	
	255:254	Request/Response	0200h (Response)	

Table 4-90 : RPMB – Read Write Counter Value Flow

4.23.3.3 Authenticated Data Write

The Authenticated Data Write is initiated by a SECURE_SEND command. The RPMB Data Frame delivered from the host to the SD card included the Request Message Type = 0003h, Nonce generated by the host, Sector count, Address, Write Counter, Data and MAC.

When the SD card receives this RPMB Data Frame, that SD card first checks whether the Write Counter has expired. If the Write Counter has expired, then that SD card sets the result to 0085h (write failure, write counter expired) and no data is written to the RPMB data area.

After checking the Write Counter is not expired, the Address is checked. If there is an error in the Address (e.g., out of range), then the result is set to 0004h (address failure) and no data is written to the RPMB data area.

After checking the address is valid, the SD card calculates the MAC (refer to Section 4.23.1) and compares this with the MAC in the request. If the MAC in the request and the calculated MAC are different, then the controller sets the result to 0002h (authentication failure) and no data is written to the RPMB data area.

If the MAC in the request and the calculated MAC are equal, then the SD card compares the Write Counter in the request with the Write Counter stored in the SD card. If the counters are different, then the SD card sets the result to 0003h (counter failure) and no data is written to RPMB data area.

If the MAC and Write Counter comparisons are successful, then the write request is authenticated. The Data from the request is written to the Address indicated in the request and the Write Counter is incremented by one.
If the write fails, then the returned result is 0005h (write failure). If another error occurs during the write procedure, then the returned result is 0001h (general failure).

The success of programming the data should be checked by the host by reading the result register of the RPMB:

1. The host initiates the Authenticated Data Write Verification process by issuing a SECURE_SEND command with delivery of a RPMB data frame containing the Request Message type = 0005h;
2. The SD card returns a successful completion of the SECURE_SEND command when the verification result is read for retrieval;
3. The host should then retrieve the verification result by issuing a SECURE_RECEIVE command; and
4. The controller returns a successful completion of the SECURE_RECEIVE command and returns the RPMB data frame containing the Response Message Type = 0300h, copy of the Nonce received with Write request, the incremented counter value, the data address, the MAC and result of the data programming operation.

Command	Bytes in Command	Field Name	Value	Objective
SECURE_SEND 1	**Data populated by the host and sent to the controller**			Program data request
	222-*N*:00	Stuff Bytes	0…00h	
	222:222-(*N*-1)	MAC/Key	MAC generated by the host	
	223	RPMB Target	RPMB target to access (00h)	
	239:224	Nonce	Nonce generated by the host	
	243:240	Write Counter	Current Write Counter Value	
	247:244	Address	Address in RPMB	
	251:248	Sector Count	Number of 512B blocks	
	253:252	Result	0000h	
	255:254	Request/Response	0003h (*Request*)	
	(M-1)+256:256	Data	Data to be written	

SECURE_SEND 2	**Data populated by the host and sent to the controller**			Request Result of data Programming
	222-*N*:00	Stuff Bytes	0…00h	
	222:222-(*N*-1)	MAC/Key	0…00h	
	223	RPMB Target	RPMB target to access (00h)	
	239:224	Nonce	0…00h	
	243:240	Write Counter	00000000h	
	247:244	Address	00000000h	
	251:248	Sector Count	00000000h	
	253:252	Result	0000h	
	255:254	Request/Response	0005h (*Request*)	
SECURE_RECEIVE 1	**Data populated by the controller and returned to the host**			Retrieve Result from data Programming
	222-*N*:00	Stuff Bytes	0…00h	
	222:222-(*N*-1)	MAC/Key	MAC generated by the controller	
	223	RPMB Target	RPMB target response was sent from (00h)	
	239:224	Nonce	Copy of the Nonce received with the Write request	
	243:240	Write Counter	Incremented Write Counter Value	
	247:244	Address	Address in RPMB	
	251:248	Sector Count	00000000h	
	253:252	Result	Result Code	
	255:254	Request/Response	0300h (Response)	

Table 4-91 : RPMB – Authentication Data Write Flow

4.23.3.4 Authenticated Data Read

The Authenticated Data Read sequence is initiated by a SECURE_SEND command. The RPMB data frame delivered from the host to the SD card includes the Request Message Type =0004h, Nonce, Address, and the Sector count.

When the SD card receives this RPMB Data Frame, that controller first check the address. If there is an error in the Address, then the result is set to 0004h (address failure) and the data read is not valid.

When the host receives a successful completion of the SECURE_SEND command from the controller, that host should send a SECURE_RECEIVE command to the SD card to retrieve the data. The controller returns an RPMB Data Frame with Response Message Type (0400h), the Sector Count, a copy of the Nonce received in the request, the Address, the Data, the controller calculated MAC, and the Result.
Note: It is the responsibility of the host to verify the MAC returned on an Authenticated Data Read Request.

Command	Bytes in Command	Field Name	Value	Objective
SECURE_SEND 1	**Data populated by the host and sent to the controller**			Read Data request
	222-*N*:00	Stuff Bytes	0…00h	
	222:222-(*N*-1)	MAC/Key	0…00h	
	223	RPMB Target	RPMB target to access (00h)	
	239:224	Nonce	Nonce generated by the host	
	243:240	Write Counter	00000000h	
	247:244	Address	Address in RPMB	
	251:248	Sector Count	Number of 512B blocks	
	253:252	Result	0000h	
	255:254	Request/Response	0004h (*Request*)	
SECURE_RECEIVE 1	**Data populated by the controller and returned to the host**			Retrieve result and data from read request
	222-*N*:00	Stuff Bytes	0…00h	
	222:222-(*N*-1)	MAC/Key	MAC generated by the controller	
	223	RPMB Target	RPMB target response was sent from (00h)	
	239:224	Nonce	Copy of the Nonce generated by the host	
	243:240	Write Counter	00000000h	
	247:244	Address	Address in RPMB	
	251:248	Sector Count	Number of 512B blocks	
	253:252	Result	Result Code	
	255:254	Request/Response	0400h (Response)	
	(M-1)+256:256	Data	Data read from RPMB unit/RPMB target	

Table 4-92 : RPMB – Authentication Data Read Flow

4.23.3.5 Authenticated Device Configuration Block Write

The Authenticated Device configuration block Write is initiated by a SECURE_SEND command. The RPMB Data Frame delivered from the host to the SD Card includes the Request Message type = 0006h, Nonce generated by host, Sector Count = 01h, MAC, Write Counter set to the current Write Counter value, and the RPMB Device configuration Block Data structure (refer to Table 4-93). All other fields are cleared to 0h.

If the Write Counter has expired, then that controller sets the result to 0085h (write failure, write counter expired) and no data is written to the Device Configuration Block.

The SD card calculate the MAC of Request Type, Block Count, Write Counter, Address and Data, and compares this with the MAC in the request. If the MAC in the request and the calculated MAC are different, then the SD card sets the result to 0002h (authentication failure) and no data is written to the RPMB Device Configuration Block.

If the Data from RPMB Device configuration Block attempts to disable Boot Partition Protection, then the

SD card sets the result to 0008h (Invalid RPMB Device Configuration Block) and no data is written to the RPMB Device configuration Block.
If the MAC in the requested and the calculated MAC are equal, then the write request is authenticated. The Data from the request is written to the RPMB Device Configuration Block.
If any error occurs during the write procedure, then the returned result is 0001h (general failure).
The controller returns successful completion for SECURE_SEND command when the Authenticated Data Write operation is completed regardless of whether the Authenticated Device Configuration Block Write was successful or not.

When the host receives a successful completion of the SECURE_SEND command from the SD card, that host should send a SECURE_RECEIVE command to the controller to retrieve the data. The controller returns an RPMB Data Frame with Response Message Type (0600h), the incremented counter value, copy of the Nonce received with the Write request, the MAC, and the Result. All other fields are cleared to 0h.

The Write counter for the Device Configuration Block is independent of the Write Counter for RPMB unit/ RPMB target 0. Authenticated Device Configuration Block Writes do not affect the Write Counter for RPMB target 0 / RPMB unit since the data is not part of RPMB data area. The current value of Write Counter for the Device Configuration Block may be read using an Authenticated Device Configuration Block Read (refer to Section 4.23.3.6).

Command	Bytes in Command	Field Name	Value	Objective
SECURE_SEND 1	**Data populated by the host and sent to the controller**			Request Device Configuration Block Write
	222-*N*:00	Stuff Bytes	0…00h	
	222:222-(*N*-1)	MAC/Key	MAC generated by the host	
	223	RPMB Target	RPMB target to access (00h)	
	239:224	Nonce	Nonce generated by the host	
	243:240	Write Counter	Current Write Counter Value	
	247:244	Address	00000000h	
	251:248	Sector Count	00000001h	
	253:252	Result	0000h	
	255:254	Request/Response	0006h (*Request*)	
	767:256	Data	RPMB Device Configuration Block data structure	
SECURE_SEND 2	**Data populated by the host and sent to the controller**			Request result for data programming
	222-*N*:00	Stuff Bytes	0…00h	
	222:222-(*N*-1)	MAC/Key	0…00h	
	223	RPMB Target	RPMB target to access (00h)	
	239:224	Nonce	0...00h	
	243:240	Write Counter	00000000h	
	247:244	Address	00000000h	
	251:248	Sector Count	00000000h	
	253:252	Result	0000h	
	255:254	Request/Response	0005h (*Request*)	

SECURE_RECEIVE 1	**Data populated by the controller and returned to the host**			Retrieve Device Configuration Block Write Result
	222-*N*:00	Stuff Bytes	0…00h	
	222:222-(*N*-1)	MAC/Key	MAC generated by the controller	
	223	RPMB Target	RPMB target response was sent from (00h)	
	239:224	Nonce	Copy of the Nonce received with the Write request	
	243:240	Write Counter	Incremented Write Counter Value	
	247:244	Address	00000000h	
	251:248	Sector Count	00000000h	
	253:252	Result	Result Code	
	255:254	Request/Response	0600h (Response)	

Table 4-93 : RPMB – Authenticated Device Configuration Block Write Flow

4.23.3.6 Authenticated Device Configuration Block Read

The Authenticated Device Configuration Block Read sequence is initiated by a SECURE_SEND command. The RPMB data frame delivered from the host to the SD card includes the Nonce, Request Message Type = 0007h and Sector count =01h. All other fields are cleared to 0h.

When the host receives a successful completion of the SECURE_SEND command from the SD card, that host should send a SECURE_RECEIVE command to the SD card to retrieve the data. The SD card returns an RPMB Data Frame with Response Message Type (0700h), the Sector Count =01h, a copy of Nonce received in the request, the RPMB Device Configuration Block Data Structure (Table 4-83), the MAC, the Write Counter set the current Write Counter value, and the Result. All other fields are cleared to 0h

The Write Counter for the Device Configuration Block is independent of the Write Counter for RPMB target. The controller returns the Device Configuration Block Write Counter as shown in Table 4-94.

Command	Bytes in Command	Field Name	Value	Objective
SECURE_SEND 1	**Data populated by the host and sent to the controller**			Request Device Configuration Block Read
	222-*N*:00	Stuff Bytes	0…00h	
	222:222-(*N*-1)	MAC/Key	0…00h	
	223	RPMB Target	RPMB target to access (00h)	
	239:224	Nonce	Nonce generated by the host	
	243:240	Write Counter	00000000h	
	247:244	Address	00000000h	
	251:248	Sector Count	00000001h	
	253:252	Result	0000h	
	255:254	Request/Response	0007h (*Request*)	

SECURE_RECEIVE 1	**Data populated by the controller and returned to the host**			Retrieve Device Configuration Block Read Result
	222-*N*:00	Stuff Bytes	0…00h	
	222:222-(*N*-1)	MAC/Key	MAC generated by the controller	
	223	RPMB Target	RPMB target response was sent from (00h)	
	239:224	Nonce	Copy of the Nonce generated by the host	
	243:240	Write Counter	Current Write Counter Value	
	247:244	Address	00000000h	
	251:248	Sector Count	00000000h	
	253:252	Result	Result Code	
	255:254	Request/Response	0700h (Response)	
	(M-1)+256:256	Data	RPMB Device Configuration Block data structure	

Table 4-94 : RPMB – Authenticated Device Configuration Block Read Flow

4.23.4 Security Protocol Type and Security Protocol Specific

The security protocol code used in the argument fields in SECURE_SEND and SECURE_RECEIVE are as defined for SCSI commands SECURITY PROTOCOL IN and SECURITY PROTOCOL OUT respectively in INCITS 513-2015 SPC-4 (T10) and INCITS SPC-6 (T10) documents. A unique security protocol ID is assigned by T10 for each application. Refer to SFSC whose reference is described in A.2.4 and Table 4-31 for the details of Security Protocol Specific 0 and Security Protocol Specific 1.

4.23.4.1 Security Protocol 00h

A SECURE_RECEIVE command with security protocol field cleared to 00h shall return information about the security protocols applications supported by the card out of the Security Protocol Field list defined by INCITS 513-2015 SPC-4 (T10). This command is used in security discovery process and is not assigned with SECURE_SEND command. Refer to SFSC whose reference is described in A.2.4 and Table 4-31 for the details of security protocol 00h and Security Protocol specific field.

4.23.4.2 Security Protocol EAh

Security Protocol EAh is assigned for NVMe interface use (refer to Security Protocol Field in SECURITY PROTOCOL OUT command table in INCITS 513-2015 SPC-4 (T10). This Security Protocol code may be used while operating SD Express card through its PCIe/NVMe interface. The specific usage type is defined by the Security Protocol Specific Field in Table 4-95.

Security Protocol Specific (SPSP0/SPSP1) Value	Description	NVMe Security Specific Field (NSSF) Definition
0001h	Replay Protected Memory Block	RPMB Target
0002h – FFFFh	Reserved	Reserved

Note: There is a single RPMB target in SD card. Therefore the value of RPMB target in the NVMe Security Specific Field shall be 00h.

Table 4-95 : Security Protocol EAh – Security Protocol Specific Values

4.23.4.3 Security Protocol E7h

Security Protocol E7h is assigned for SD Extended Security use in INCITS SPC-6 (T10). This Security Protocol Code may be used while operating any SD card through its SD interface that supports SECURE_RECEIVE and SECURE_SEND commands. The specific usage type is defined by the Security Protocol Specific Field in Table 4-96.

Security Protocol Specific (SPSP0/SPSP1) Value	Description	SD Security Specific Field (SSSF) Definition
0001h	Replay Protected Memory Block	RPMB Target
0002h – FFFFh	Reserved	Reserved

Note: There is a single RPMB target in SD card. Therefore the value of RPMB target in the SD Security Specific Field shall be 00h.

Table 4-96 : Security Protocol E7h – Security Protocol Specific Values

4.23.5 User Area Write Protection States and Authentication Control by RPMB

User Area Write Protection is an optional configurable controller capability that enables the host to control the write protection state of User Area or to determine the write protection state of User Area. Write Protection state control is defined in CSD register (refer to Table 5-4, Table 5-16 and Table 5.3.4-1) and capability bit for Write Protect Until Power Cycle is defined in SD Status register (refer to Table 4-44).

4.23.5.1 User Area Write Protection States

Table 4-97 defines write protection states, that may be supported by card. All protection states persist across power cycles and controller level resets (refer to Section 4.2.1) except Write Protect Until Power Cycle, which transitions to the No Write Protect state on the occurrence of a power cycle. Note that Permanent Write Protect and Temporary Write Protect are mandatory features for SD interface cards while Write Protect Until Power Cycle is optional and it may be supported only if RPMB is enabled.

State	Description	Persistent Across	
		Power Cycles	Controller Level Resets
No Write Protect	The User Area is not Write Protected	Yes	Yes
Temporary Write Protect (TWP)	The User Area is Write Protected	Yes	Yes
Write Protect Until Power Cycle (WP_UPC)	The User Area is Write Protected until the next power cycle	No	Yes
Permanent Write Protect (PWP)	The Namespace is Permanently Write Protected	Yes	Yes

Table 4-97 : User Area Write Protection States

Figure 4-101 defines the transition between Write Protection states. All Write Protection state transitions are based on commands unless specified otherwise.

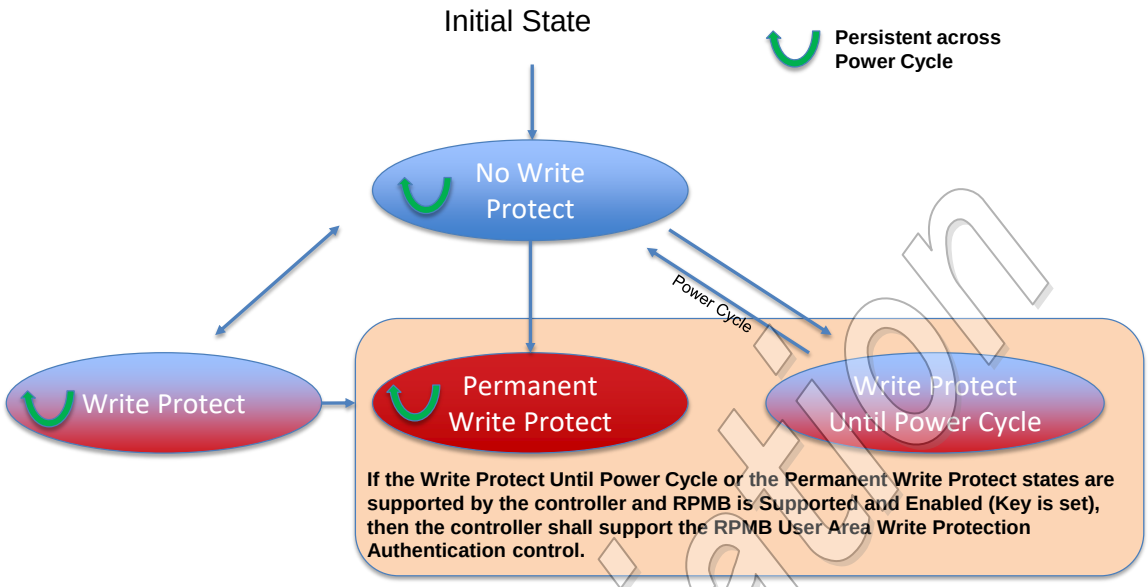


Figure 4-101 : User Area Write Protection State Machine

If RPMB is supported by the card then Write Protect Until Power Cycle (WP_UPC) and Permanent Write Protect (PWP) states are subject to the User Area Write Protection Authentication Control mechanism, which determines whether the controller processes or aborts commands which cause a transition into either of these two states (refer to Table 4-83).
The results of using User Area Write Protection in combination with an external write protection system (e.g., TCG Storage Interface Interactions Specification) are outside the scope of this specification.
Host software may check the current User Area Write Protection state of a User Area using CSD Register bits (refer to Table 5-4, Table 5-16 and Table 5.3.4-1)

- For SD Interface
  - If RPMB is enabled, then PWP and WP_UPC shall be controlled via RPMB User Area Write Protection Authentication Control.
  - If RPMB is disabled, then PWP shall NOT be controlled via RPMB User Area Write Protection Authentication Control.
- For SD Express card
  - If RPMB is supported by the card, it shall be supported by both interfaces – PCIe/NVMe and SD.
  - NVMe interface of SD Express card may support PWP or WP_UPC over PCIe interface only if the RPMB is supported and control them via “Namespace Write Protection Authentication Control” (as defined in NVMe specification ver 1.4). If RPMB is disabled, then PWP, WP_UPC shall not be supported by PCIe/NVMe interface.
  - If NVMe interface of SD Express card is not supporting PWP or WP_UPC as well as RPMB is not supported by the card, then PWP and TWP features of SD interface are applicable without RPMB and User Area Write Protection Authentication Control.

4.23.5.2 User Area Write Protection Configuration

A host can configure or determine the Write Protection state using CSD register. User Area write protection states are given below based on the CSD register bits set for Temporary Write Protect, Permanent Write Protect and Write Protect Until Power Cycle along with NVMe mapping.
For further information about SD vs NVMe behavior, refer to Section 8.1.7.

CSD Register			SD Definition	NVMe Definition
PERM_WRITE_PROTECT [13:13]	TMP_WRITE_PROTECT [12:12]	WP_UPC [9:9]		
0	0	0	No Write Protect	No Write Protect
0	0	1	Write Protect Until Power Cycle	Write Protect Until Power Cycle
0	1	Don't care	Temporary Write Protect	Write Protect
1	Don't care	Don't care	Permanent Write Protect	Permanent Write Protect

Note: WP_UPC is cleared to 0 after power cycle.

Table 4-98 : Write Protection States definition and NVMe Mapping

If RPMB is enabled:

- If host attempts to modify the Write Protect Until Power Cycle state or the Permanent Write Protect state without User Area Write Protection Authentication, then the command shall fail. In this case, card shall set the WP_VIOLATION bit.
- If a host attempts to modify the Write Protect Until Power Cycle state and bit 0 of the Write Protection Authentication Control field in RPMB Device Configuration Block Data (Table 4-83 and Table 5-32) is cleared to '0', then the command shall fail. In this case, card shall set the WP_VIOLATION bit.
- If a host attempts to set the Permanent Write Protect state and bit 1 of the Write Protection Authentication Control field in RPMB Device Configuration Block Data (Table 4-83 and Table 5-32) is cleared to '0', then the command shall fail. In this case, card shall set the WP_VIOLATION bit.

Card shall respect any attempt to modify the Temporary Write Protection state without doing RPMB User Area Write Protection Authentication.

5. Card Registers

Six registers are defined within the card interface: OCR, CID, CSD, RCA, DSR and SCR. These can be accessed only by corresponding commands (see Section 4.7). The OCR, CID, CSD and SCR registers carry the card/content specific information, while the RCA and DSR registers are configuration registers storing actual configuration parameters.
In order to enable future extension, the card shall return 0 in the reserved bits of the registers.

5.1 OCR register

The 32-bit operation conditions register stores the V_{DD} voltage profile of the non UHS-II card and V_{DD1} voltage profile of the UHS-II card. Additionally, this register includes status information bits. One status bit is set if the card power up procedure has been finished. This register includes another status bit indicating the card capacity status after set power up status bit. The OCR register shall be implemented by the cards.
The 32-bit operation conditions register stores the V_{DD} voltage profile of the card. Bit 7 of OCR is newly defined for Dual Voltage Card and set to 0 in default. If a Dual Voltage Card does not receive CMD8, OCR bit 7 in the response indicates 0, and the Dual Voltage Card which received CMD8, sets this bit to 1.
Additionally, this register includes 2 more status information bits.
Bit 31 - Card power up status bit, this status bit is set if the card power up procedure has been finished.
Bit 30 - Card Capacity Status bit, 0 indicates that the card is SDSC. 1 indicates that the card is SDHC or SDXC. The Card Capacity Status bit is valid after the card power up procedure is completed and the card power up status bit is set to 1. The Host shall read this status bit to identify SDSC Card or SDHC/SDXC/SDUC Card.

OCR bit position	OCR Fields Definition
0-3	reserved
4	reserved
5	reserved
6	reserved
7	Reserved for Low Voltage Range
8	reserved
9	reserved
10	reserved
11	reserved
12	reserved
13	reserved
14	reserved
15	2.7-2.8
16	2.8-2.9
17	2.9-3.0
18	3.0-3.1
19	3.1-3.2
20	3.2-3.3
21	3.3-3.4
22	3.4-3.5
23	3.5-3.6
24[3]	Switching to 1.8V Accepted (S18A)
25-26	reserved
27[4]	Over 2TB support Status (CO2T)
28	reserved
29	UHS-II Card Status
30	Card Capacity Status (CCS)[1]
31	Card power up status bit (busy)[2]

VDD Voltage Window

1) This bit is valid only when the card power up status bit is set.
2) This bit is set to LOW if the card has not finished the power up routine.
3) Only UHS-I card supports this bit.
4) Only SDUC card supports this bit.

Table 5-1 : OCR Register Definition

The supported voltage range is coded as shown in Table 5-1. A voltage range is not supported if the corresponding bit value is set to LOW. As long as the card is busy, the corresponding bit (31) is set to LOW.

VDD Voltage Window of OCR indicates V_{DD1} voltage range in case of UHS-II Card.
UHS-II Card Status bit is added in Bit 29 to indicate whether the card supports UHS-II Interface. Non UHS-II Card sets Bit 29 to 0 and UHS-II Card sets Bit 29 to 1. This bit is not affected by whether VDD2 is supplied or not.

5.2 CID register

The Card IDentification (CID) register is 128 bits wide. It contains the card identification information used during the card identification phase. Every individual Read/Write (RW) card shall have a unique identification number. The structure of the CID register is defined in the following paragraphs:

Name	Field	Width	CID-slice
Manufacturer ID	MID	8	[127:120]
OEM/Application ID	OID	16	[119:104]
Product name	PNM	40	[103:64]
Product revision	PRV	8	[63:56]
Product serial number	PSN	32	[55:24]
reserved	--	4	[23:20]
Manufacturing date	MDT	12	[19:8]
CRC7 checksum	CRC	7	[7:1]
not used, always 1	-	1	[0:0]

Table 5-2 : The CID Fields

- **MID**

An 8-bit binary number that identifies the card manufacturer. The MID number is controlled, defined, and allocated to a SD Memory Card manufacturer by the SD-3C, LLC. This procedure is established to ensure uniqueness of the CID register.

- **OID**

A 2-character ASCII string that identifies the card OEM and/or the card contents (when used as a distribution media either on ROM or FLASH cards). The OID number is controlled, defined, and allocated to a SD Memory Card manufacturer by the SD-3C, LLC. This procedure is established to ensure uniqueness of the CID register.

Note: SD-3C, LLC licenses companies that wish to manufacture and/or sell SD Memory Cards, including but not limited to flash memory, ROM, OTP, RAM, and SDIO Combo Cards.
SD-3C, LLC is a limited liability company established by Kioxia Corporation, Panasonic Holdings Corporation, and SanDisk Corporation.

- **PNM**

The product name is a string, 5-character ASCII string.

- **PRV**

The product revision is composed of two Binary Coded Decimal (BCD) digits, four bits each, representing an "n.m" revision number. The "n" is the most significant nibble and "m" is the least significant nibble.
As an example, the PRV binary value field for product revision "6.2" will be: 0110 0010b

- **PSN**

The Serial Number is 32 bits of binary number.

- **MDT**

  The manufacturing date is composed of two hexadecimal digits, one is 8 bits representing the year(y) and the other is 4 bits representing the month (m).
  The "m" field [11:8] is the month code. 1 = January.
  The "y" field [19:12] is the year code. 0 = 2000.
  As an example, the binary value of the Date field for production date "April 2001" will be:
  00000001 0100.

- **CRC**

  CRC7 checksum (7 bits). This is the checksum of the CID contents computed as described in Section 4.5.

5.3 CSD Register

The Card-Specific Data register provides information regarding access to the card contents. The CSD defines the data format, error correction type, maximum data access time, whether the DSR register can be used, etc. The programmable part of the register (entries marked by W or E, see below) can be changed by CMD27. The types of the entries in the table below are coded as follows: R = readable, W(1) = writable once, W = multiple writable.

5.3.1 CSD_STRUCTURE

Field structures of the CSD register are different depend on the Physical Layer Specification Version and Card Capacity.
The CSD_STRUCTURE field in the CSD register indicates its structure version.
Table 5-3 shows the version number of the related CSD structure.

CSD_STRUCTURE	CSD structure version	Card Capacity
0	CSD Version 1.0	Standard Capacity
1	CSD Version 2.0	High Capacity and Extended Capacity
2	CSD Version 3.0	Ultra Capacity (SDUC)
3	reserved	

Table 5-3 : CSD Register Structure

5.3.2 CSD Register (CSD Version 1.0)

Name	Field	Width	Value	Cell Type	CSD-slice
CSD structure	CSD_STRUCTURE	2	00b	R	[127:126]
reserved	-	6	00 0000b	R	[125:120]
data read access-time-1	TAAC	8	xxh	R	[119:112]
data read access-time-2 in CLK cycles (NSAC*100)	NSAC	8	xxh	R	[111:104]
max. data transfer rate	TRAN_SPEED	8	32h or 5Ah	R	[103:96]
card command classes	CCC	12	01x110110101b	R	[95:84]
max. read data block length	READ_BL_LEN	4	xh	R	[83:80]
partial blocks for read allowed	READ_BL_PARTIAL	1	1b	R	[79:79]
write block misalignment	WRITE_BLK_MISALIGN	1	xb	R	[78:78]
read block misalignment	READ_BLK_MISALIGN	1	xb	R	[77:77]
DSR implemented	DSR_IMP	1	xb	R	[76:76]
reserved	-	2	00b	R	[75:74]
device size	C_SIZE	12	xxxh	R	[73:62]
max. read current @VDD min	VDD_R_CURR_MIN	3	xxxb	R	[61:59]
max. read current @VDD max	VDD_R_CURR_MAX	3	xxxb	R	[58:56]
max. write current @VDD min	VDD_W_CURR_MIN	3	xxxb	R	[55:53]
max. write current @VDD max	VDD_W_CURR_MAX	3	xxxb	R	[52:50]
device size multiplier	C_SIZE_MULT	3	xxxb	R	[49:47]
erase single block enable	ERASE_BLK_EN	1	xb	R	[46:46]
erase sector size	SECTOR_SIZE	7	xxxxxxxb	R	[45:39]
write protect group size	WP_GRP_SIZE	7	xxxxxxxb	R	[38:32]
write protect group enable	WP_GRP_ENABLE	1	xb	R	[31:31]
reserved (Do not use)		2	00b	R	[30:29]
write speed factor	R2W_FACTOR	3	xxxb	R	[28:26]
max. write data block length	WRITE_BL_LEN	4	xxxxb	R	[25:22]
partial blocks for write allowed	WRITE_BL_PARTIAL	1	xb	R	[21:21]
reserved	-	5	00000b	R	[20:16]
File format group	FILE_FORMAT_GRP	1	xb	R/W(1)	[15:15]
copy flag	COPY	1	xb	R/W(1)	[14:14]
permanent write protection	PERM_WRITE_PROTECT	1	xb	R/W(1)	[13:13]
temporary write protection	TMP_WRITE_PROTECT	1	xb	R/W	[12:12]
File format	FILE_FORMAT	2	xxb	R/W(1)	[11:10]
write protection until power cycle*	WP_UPC	1	xb**	R/W**	[9:9]
reserved		1	0b	R	[8:8]
CRC	CRC	7	xxxxxxxb	R/W	[7:1]
not used, always'1'	-	1	1b	-	[0:0]

* Register “write protection until power cycle” is applicable for Physical Layer Specification Version 9.00 or later.
** If WP_UPC_SUPPORT in SD Status indicates 0, the cell type is R and value of this bit is always 0b.

Table 5-4 : The CSD Register Fields (CSD Version 1.0)

The following sections describe the CSD fields and the relevant data types. If not explicitly defined otherwise, all bit strings are interpreted as binary coded numbers starting with the left bit first.

- **TAAC**

Defines the asynchronous part of the data access time.

TAAC bit position	code
2:0	time unit 0=1ns, 1=10ns, 2=100ns, 3=1µs, 4=10µs, 5=100µs, 6=1ms, 7=10ms
6:3	time value 0=reserved, 1=1.0, 2=1.2, 3=1.3, 4=1.5, 5=2.0, 6=2.5, 7=3.0, 8=3.5, 9=4.0, A=4.5, B=5.0, C=5.5, D=6.0, E=7.0, F=8.0
7	reserved

Table 5-5 : TAAC Access Time Definition

- **NSAC**

Defines the worst case for the clock-dependent factor of the data access time. The unit for NSAC is 100 clock cycles. Therefore, the maximal value for the clock-dependent part of the data access time is 25.5 k clock cycles.
The total access time N_{AC} as expressed in the Table 4-59 is the sum of TAAC and NSAC. It should be computed by the host for the actual clock rate. The read access time should be interpreted as a typical delay for the first data bit of a data block or stream.

- **TRAN_SPEED**

Table 5-6 defines the maximum data transfer rate per one data line - TRAN_SPEED:

TRAN_SPEED bit	code
2:0	transfer rate unit 0=100kbit/s, 1=1Mbit/s, 2=10Mbit/s, 3=100Mbit/s, 4... 7=reserved
6:3	time value 0=reserved, 1=1.0, 2=1.2, 3=1.3, 4=1.5, 5=2.0, 6=2.5, 7=3.0, 8=3.5, 9=4.0, A=4.5, B=5.0, C=5.5, D=6.0, E=7.0, F=8.0
7	reserved

Table 5-6 : Maximum Data Transfer Rate Definition

Note that for current SD Memory Cards, this field shall be always 0_0110_010b (032h) which is equal to 25 MHz - the mandatory maximum operating frequency of SD Memory Card.
In High-Speed mode, this field shall be always 0_1011_010b (05Ah) which is equal to 50 MHz, and when the timing mode returns to the default by CMD6 or CMD0 command, its value will be 032h.

- **CCC**

The SD Memory Card command set is divided into subsets (command classes). The card command class register CCC defines which command classes are supported by this card. A value of 1 in a CCC bit means that the corresponding command class is supported. For command class definitions, refer to Table 4-21.

CCC bit	Supported card command class
0	class 0
1	class 1
.....	
11	class 11

Table 5-7 : Supported Card Command Classes

- **READ_BL_LEN**

The maximum read data block length is computed as $2^{READ\_BL\_LEN}$. The maximum block length might therefore be in the range 512...2048 bytes (Refer to 4.3.3 for details). Note that in an SD Memory Card the WRITE_BL_LEN is always equal to READ_BL_LEN

READ_BL_LEN	Block length
0-8	reserved
9	2^9 = 512 Bytes
10	2^{10} = 1024 Bytes
11	2^{11} = 2048 Bytes
12-15	reserved

Table 5-8 : Data Block Length

- **READ_BL_PARTIAL (always = 1 in SD Memory Card)**

Partial Block Read is always allowed in an SD Memory Card. It means that smaller blocks can be used as well. The minimum block size will be one byte.

- **WRITE_BLK_MISALIGN**

Defines if the data block to be written by one command can be spread over more than one physical block of the memory device. The size of the memory block is defined in WRITE_BL_LEN.
WRITE_BLK_MISALIGN=0 signals that crossing physical block boundaries is invalid.
WRITE_BLK_MISALIGN=1 signals that crossing physical block boundaries is allowed.

- **READ_BLK_MISALIGN**

Defines if the data block to be read by one command can be spread over more than one physical block of the memory device. The size of the memory block is defined in READ_BL_LEN.
READ_BLK_MISALIGN=0 signals that crossing physical block boundaries is invalid.
READ_BLK_MISALIGN=1 signals that crossing physical block boundaries is allowed.

- **DSR_IMP**

Defines if the configurable driver stage is integrated on the card. If set, a driver stage register (DSR) shall be implemented (also see Section 5.5).

DSR_IMP	DSR type
0	no DSR implemented
1	DSR implemented

Table 5-9 : DSR Implementation Code Table

- **C_SIZE**

This parameter is used to compute the user's data card capacity (not include the security protected area). The memory capacity of the card is computed from the entries C_SIZE, C_SIZE_MULT and READ_BL_LEN as follows:

memory capacity = BLOCKNR * BLOCK_LEN

Where

BLOCKNR = (C_SIZE+1) * MULT

MULT = $2^{C\_SIZE\_MULT+2}$ (C_SIZE_MULT < 8)

BLOCK_LEN = $2^{READ\_BL\_LEN}$, (READ_BL_LEN < 12)

To indicate 2 GByte card, BLOCK_LEN shall be 1024 bytes.
Therefore, the maximal capacity that can be coded is 4096*512*1024 = 2 G bytes.
Example: A 32 Mbyte card with BLOCK_LEN = 512 can be coded by C_SIZE_MULT = 3 and C_SIZE = 2000.
The Maximum Data Area size of Standard Capacity SD Card is 4,153,344 sectors (2028MB).

- **VDD_R_CURR_MIN, VDD_W_CURR_MIN**

The maximum values for read and write currents at the minimal power supply V_{DD} are coded as follows:

VDD_R_CURR_MIN VDD_W_CURR_MIN	Code for Current Consumption @ VDD
2:0	0=0.5mA; 1=1mA; 2=5mA; 3=10mA; 4=25mA; 5=35mA; 6=60mA; 7=100mA

Table 5-10 : $V_{DD, min}$ Current Consumption

- **VDD_R_CURR_MAX, VDD_W_CURR_MAX**

The maximum values for read and write currents at the maximal power supply V_{DD} are coded as follows:

VDD_R_CURR_MAX VDD_W_CURR_MAX	Code for Current Consumption @ VDD
2:0	0=1mA; 1=5mA; 2=10mA; 3=25mA; 4=35mA; 5=45mA; 6=80mA; 7=200mA

Table 5-11 : $V_{DD, max}$ Current Consumption

- **C_SIZE_MULT**

This parameter is used for coding a factor MULT for computing the total device size (see 'C_SIZE'). The factor MULT is defined as $2^{C\_SIZE\_MULT+2}$.

C_SIZE_MULT	MULT
0	$2^2 = 4$
1	$2^3 = 8$
2	$2^4 = 16$
3	$2^5 = 32$
4	$2^6 = 64$
5	$2^7 = 128$
6	$2^8 = 256$
7	$2^9 = 512$

Table 5-12 : Multiply Factor for the Device Size

- **ERASE_BLK_EN**

The ERASE_BLK_EN defines the granularity of the unit size of the data to be erased. The erase operation can erase either one or multiple units of 512 bytes or one or multiple units (or sectors) of SECTOR_SIZE (see definition below).

If ERASE_BLK_EN=0, the host can erase one or multiple units of SECTOR_SIZE. The erase will start from the beginning of the sector that contains the start address to the end of the sector that contains the end address. For example, if SECTOR_SIZE=31 and the host sets the Erase Start Address to 5 and the Erase End Address to 40, the physical blocks from 0 to 63 will be erased as shown in Figure 5-1.

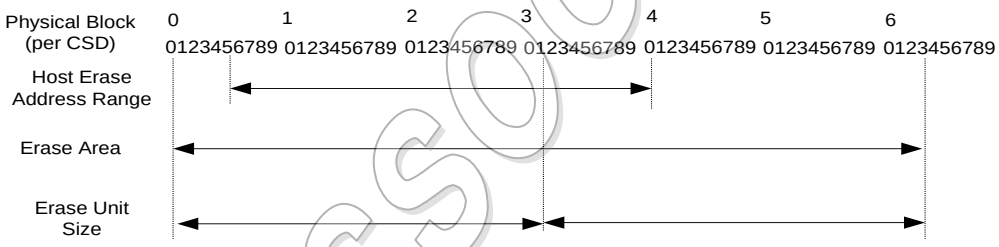


Figure 5-1 : ERASE_BLK_EN = 0 Example

If ERASE_BLK_EN=1 the host can erase one or multiple units of 512 bytes. All blocks that contain data from start address to end address are erased. For example, if the host sets the Erase Start Address to 5 and the Erase End Address to 40, the physical blocks from 5 to 40 will be erased as shown in Figure 5-2.

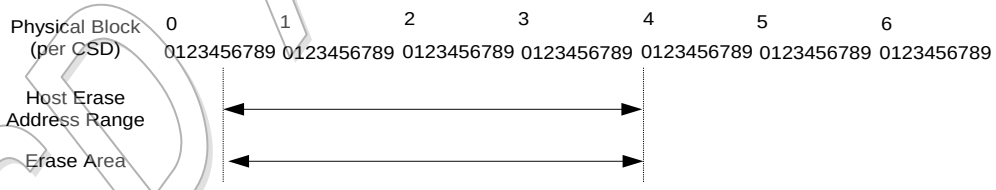


Figure 5-2 : ERASE_BLK_EN = 1 Example

- **SECTOR_SIZE**

The size of an erasable sector. The content of this register is a 7-bit binary coded value, defining the

number of write blocks (see WRITE_BL_LEN). The actual size is computed by increasing this number by one. A value of zero means one write block, 127 means 128 write blocks.

- **WP_GRP_SIZE**

The size of a write protected group. The content of this register is a 7-bit binary coded value, defining the number of erase sectors (see SECTOR_SIZE). The actual size is computed by increasing this number by one. A value of zero means one erase sector, 127 means 128 erase sectors.

- **WP_GRP_ENABLE**

A value of 0 means no group write protection possible.

- **R2W_FACTOR**

Defines the typical block program time as a multiple of the read access time. The following table defines the field format.

R2W_FACTOR	Multiples of read access time
0	1
1	2 (write half as fast as read)
2	4
3	8
4	16
5	32
6,7	reserved

Table 5-13 : R2W_FACTOR

- **WRITE_BL_LEN**

The maximum write data block length is computed as $2^{WRITE\_BL\_LEN}$. The maximum block length might therefore be in the range from 512 to 2048 bytes. Write Block Length of 512 bytes is always supported. Note that in the SD Memory Card, the WRITE_BL_LEN is always equal to READ_BL_LEN.

WRITE_BL_LEN	Block Length
0-8	reserved
9	2^9 = 512 bytes
10	2^{10} = 1024 Bytes
11	2^{11} = 2048 Bytes
12-15	reserved

Table 5-14 : Data Block Length

- **WRITE_BL_PARTIAL**

Defines whether partial block sizes can be used in block write commands.
WRITE_BL_PARTIAL=0 means that only the WRITE_BL_LEN block size and its partial derivatives, in resolution of units of 512 bytes, can be used for block oriented data write.
WRITE_BL_PARTIAL=1 means that smaller blocks can be used as well. The minimum block size is one byte.

- **FILE_FORMAT_GRP**
  Indicates the selected group of file formats. This field is read-only for ROM. The usage of this field is shown in Table 5-15 (Refer to FILE_FORMAT).

- **COPY**
  Defines whether the contents is original (=0) or has been copied (=1). Setting this bit to 1 indicates that the card content is a copy. The COPY bit is a one time programmable bit except ROM card.

- **PERM_WRITE_PROTECT**
  Permanently protects the entire card content against overwriting or erasing (all write and erase commands for this card are permanently disabled). The default value is 0, i.e. not permanently write protected.

- **TMP_WRITE_PROTECT**
  Temporarily protects the entire card content from being overwritten or erased (all write and erase commands for this card are temporarily disabled). This bit can be set and reset. The default value is 0, i.e. not write protected.

- **WP_UPC**
  Write protect until power cycle protects the entire card content from being overwritten or erased (all write and erase commands for this card are temporarily disabled). This bit can be set (Write Protect) and reset (No Write Protect). A power cycle will reset this bit. The default value is 0, i.e., No Write Protect. If WP_UPC_SUPPORT in SD Status indicates 0, this bit cannot be set to 1 and always indicates 0.

- **FILE_FORMAT**
  Indicates the file format on the card. This field is read-only for ROM. The following formats are defined:

FILE_FORMAT_GRP	FILE_FORMAT	Type
0	0	Hard disk-like file system with partition table
0	1	DOS FAT (floppy-like) with boot sector only (no partition table)
0	2	Universal File Format
0	3	Others/Unknown
1	0, 1, 2, 3	Reserved

Table 5-15 : File Formats

A more detailed description is given in the File System Specification.

- **CRC**
  The CRC field carries the check sum for the CSD contents. It is computed according to Section 4.5. The checksum has to be recalculated by the host for any CSD modification. The default corresponds to the initial CSD contents.

5.3.3 CSD Register (CSD Version 2.0)

Table 5-16 shows Definition of the CSD Version 2.0 for the High Capacity SD Memory Card and Extended Capacity SD Memory Card.
The following sections describe the CSD fields and the relevant data types for SDHC and SDXC Cards.

CSD Version 2.0 is applied to SDHC and SDXC Cards. The field name in parenthesis is set to fixed value and indicates that the host is not necessary to refer these fields. The fixed values enables host, which refers to these fields, to keep compatibility to CSD Version 1.0. The Cell Type field is coded as follows: R = readable, W(1) = writable once, W = multiple writable.

Name	Field	Width	Value	Cell Type	CSD-slice
CSD structure	CSD_STRUCTURE	2	01b	R	[127:126]
reserved	-	6	00 0000b	R	[125:120]
data read access-time	(TAAC)	8	0Eh	R	[119:112]
data read access-time in CLK cycles (NSAC*100)	(NSAC)	8	00h	R	[111:104]
max. data transfer rate	(TRAN_SPEED)	8	32h, 5Ah, 0Bh or 2Bh	R	[103:96]
card command classes	CCC	12	x1x1101101x1b	R	[95:84]
max. read data block length	(READ_BL_LEN)	4	9	R	[83:80]
partial blocks for read allowed	(READ_BL_PARTIAL)	1	0	R	[79:79]
write block misalignment	(WRITE_BLK_MISALIGN)	1	0	R	[78:78]
read block misalignment	(READ_BLK_MISALIGN)	1	0	R	[77:77]
DSR implemented	DSR_IMP	1	x	R	[76:76]
reserved	-	6	00 0000b	R	[75:70]
device size	C_SIZE	22	xxxxxxh	R	[69:48]
reserved	-	1	0	R	[47:47]
erase single block enable	(ERASE_BLK_EN)	1	1	R	[46:46]
erase sector size	(SECTOR_SIZE)	7	7Fh	R	[45:39]
write protect group size	(WP_GRP_SIZE)	7	0000000b	R	[38:32]
write protect group enable	(WP_GRP_ENABLE)	1	0	R	[31:31]
reserved		2	00b	R	[30:29]
write speed factor	(R2W_FACTOR)	3	010b	R	[28:26]
max. write data block length	(WRITE_BL_LEN)	4	9	R	[25:22]
partial blocks for write allowed	(WRITE_BL_PARTIAL)	1	0	R	[21:21]
reserved	-	5	00000b	R	[20:16]
File format group	(FILE_FORMAT_GRP)	1	0	R	[15:15]
copy flag	COPY	1	x	R/W(1)	[14:14]
permanent write protection	PERM_WRITE_PROTECT	1	x	R/W(1)	[13:13]
temporary write protection	TMP_WRITE_PROTECT	1	x	R/W	[12:12]
File format	(FILE_FORMAT)	2	00b	R	[11:10]
write protection until power cycle*	WP_UPC	1	xb**	R/W**	[9:9]
reserved	-	1	0b	R	[8:8]
CRC	CRC	7	xxxxxxxb	R/W	[7:1]
not used, always'1'	-	1	1	-	[0:0]

* Register “write protection until power cycle” is applicable for Physical Layer Specification Version 9.00 or later.

** If WP_UPC_SUPPORT in SD Status indicates 0, the cell type is R and value of this bit is always 0b.

Table 5-16 : The CSD Register Fields (CSD Version 2.0)

- **TAAC**
  This field is fixed to 0Eh, which indicates 1 ms. The host should not use TAAC, NSAC, and R2W_FACTOR to calculate timeout and should uses fixed timeout values for read and write operations (See 4.6.2).

- **NSAC**
  This field is fixed to 00h. NSAC should not be used to calculate time-out values.

- **TRAN_SPEED**
  TRAN_SPEED is variable depends on bus speed mode of SD Interface. Definition of this field is same as in CSD Version1.0 in case of Default and High Speed mode. This field shall be set to 0Bh (100Mbit/sec) in both SDR50 and DDR50 mode, and shall be set to 2Bh (200Mbit/sec) in SDR104 mode. When CMD0 is received, this field is reset to 32h.
  UHS-II mode is not related to this field.

- **CCC**
  Definition of this field is same as in CSD Version1.0.

- **READ_BL_LEN**
  This field is fixed to 9h, which indicates READ_BL_LEN=512 Byte.

- **READ_BL_PARTIAL**
  This field is fixed to 0, which indicates partial block read is inhibited and only unit of block access is allowed.

- **WRITE_BLK_MISALIGN**
  This field is fixed to 0, which indicates that write access crossing physical block boundaries is always disabled in SDHC and SDXC Cards.

- **READ_BLK_MISALIGN**
  This field is fixed to 0, which indicates that read access crossing physical block boundaries is always disabled in SDHC and SDXC Cards.

- **DSR_IMP**
  Definition of this field is same as in CSD Version1.0.

- **C_SIZE**
  This field is expanded to 22 bits and can indicate up to 2 TBytes (It is the same as the maximum memory space specified by a 32-bit block address.)
  This parameter is used to calculate the user data area capacity in the SD memory card (not include the protected area). The user data area capacity is calculated from C_SIZE as follows:

  memory capacity = (C_SIZE+1) * 512KByte

  The Minimum user area size of SDHC Card is 4,211,712 sectors (2GB + 8.5MB).

The Minimum value of C_SIZE for SDHC in CSD Version 2.0 is 001010h (4112).
The maximum user area size of SDHC Card is (32GB - 80MB)
The maximum value of C_SIZE for SDHC in CSD Version 2.0 is 00FF5Fh (65375).
The Minimum user area size of SDXC Card is 67,108,864 sectors (32GB).
The Minimum value of C_SIZE for SDXC in CSD Version 2.0 is 00FFFFh (65535).
The maximum user area size of SDXC Card is 4,294,705,152 sectors (2TB - 128MB).
The maximum value of C_SIZE for SDXC in CSD Version 2.0 is 3FFEFFh (4194047).

- **ERASE_BLK_EN**
  This field is fixed to 1, which means the host can erase one or multiple units of 512 bytes.

- **SECTOR_SIZE**
  This field is fixed to 7Fh, which indicates 64 KBytes. This value is not related to erase operation. SDHC and SDXC Cards indicate memory boundary by AU size and this field should not be used.

- **WP_GRP_SIZE**
  This field is fixed to 00h. SDHC and SDXC Cards do not support write protected groups.

- **WP_GRP_ENABLE**
  This field is fixed to 0. SDHC and SDXC Cards do not support write protected groups.

- **R2W_FACTOR**
  This field is fixed to 2h, which indicates 4 multiples. Write timeout can be calculated by multiplying the read access time and R2W_FACTOR. Refer to Section 4.6.2 about write time.

- **WRITE_BL_LEN**
  This field is fixed to 9h, which indicates WRITE_BL_LEN=512 Byte.

- **WRITE_BL_PARTIAL**
  This field is fixed to 0, which indicates partial block read is inhibited and only unit of block access is allowed.

- **FILE_FORMAT_GRP**
  This field is set to 0. Host should not use this field.

- **COPY**
  Definition of this field is same as in CSD Version1.0.

- **PERM_WRITE_PROTECT**
  Definition of this field is same as in CSD Version1.0.

- **TMP_WRITE_PROTECT**
  Definition of this field is same as in CSD Version1.0.

- **WP_UPC**
  Definition of this field is same as in CSD Version1.0.

- **FILE_FORMAT**
  This field is set to 0. Host should not use this field.

- **CRC**
  Definition of this field is same as in CSD Version1.0.

5.3.4 CSD Register (CSD Version 3.0)

Table 5.3.4-1 shows Definition of the CSD Version 3.0 for the Ultra Capacity SD Memory Card. The following sections describe the CSD fields and the relevant data types for SDUC Card.

CSD Version 3.0 is applied to SDUC Card. The field name in parenthesis is set to fixed value and indicates that the host is not necessary to refer these fields. The fixed values enables host, which refers to these fields, to keep compatibility to CSD Version 1.0 or Version 2.0. The Cell Type field is coded as follows: R = readable, W(1) = writable once, W = multiple writable.

Name	Field	Width	Value	Cell Type	CSD-slice
CSD structure	CSD_STRUCTURE	2	10b	R	[127:126]
reserved	-	6	00 0000b	R	[125:120]
data read access-time	(TAAC)	8	0Eh	R	[119:112]
data read access-time in CLK cycles (NSAC*100)	(NSAC)	8	00h	R	[111:104]
max. data transfer rate	(TRAN_SPEED)	8	32h, 5Ah, 0Bh or 2Bh	R	[103:96]
card command classes	CCC	12	x1x1101101x1b	R	[95:84]
max. read data block length	(READ_BL_LEN)	4	9	R	[83:80]
partial blocks for read allowed	(READ_BL_PARTIAL)	1	0	R	[79:79]
write block misalignment	(WRITE_BLK_MISALIGN)	1	0	R	[78:78]
read block misalignment	(READ_BLK_MISALIGN)	1	0	R	[77:77]
DSR implemented	DSR_IMP	1	X	R	[76:76]
device size (user area size)	C_SIZE	28	xxxxxxxh	R	[75:48]
reserved	-	1	0	R	[47:47]
erase single block enable	(ERASE_BLK_EN)	1	1	R	[46:46]
erase sector size	(SECTOR_SIZE)	7	7Fh	R	[45:39]
write protect group size	(WP_GRP_SIZE)	7	0000000b	R	[38:32]
write protect group enable	(WP_GRP_ENABLE)	1	0	R	[31:31]
reserved		2	00b	R	[30:29]
write speed factor	(R2W_FACTOR)	3	010b	R	[28:26]
max. write data block length	(WRITE_BL_LEN)	4	9	R	[25:22]
partial blocks for write allowed	(WRITE_BL_PARTIAL)	1	0	R	[21:21]
reserved	-	5	00000b	R	[20:16]
File format group	(FILE_FORMAT_GRP)	1	0	R	[15:15]
copy flag	COPY	1	x	R/W(1)	[14:14]
permanent write protection	PERM_WRITE_PROTECT	1	x	R/W(1)	[13:13]
temporary write protection	TMP_WRITE_PROTECT	1	x	R/W	[12:12]
File format	(FILE_FORMAT)	2	00b	R	[11:10]
write protection until power cycle*	WP_UPC	1	xb**	R/W**	[9:9]
reserved	-	1	0b	R	[8:8]
CRC	CRC	7	xxxxxxxb	R/W	[7:1]
not used, always'1'	-	1	1	-	[0:0]

* Register “write protection until power cycle” is applicable for Physical Layer Specification Version 9.00 or later.
** If WP_UPC_SUPPORT in SD Status indicates 0, the cell type is R and value of this bit is always 0b.

Table 5.3.4-1 : The CSD Register Fields (CSD Version 3.0)

- **TAAC**
  Definition of this field is same as in CSD Version2.0.

- **NSAC**
  Definition of this field is same as in CSD Version2.0.

- **TRAN_SPEED**
  Definition of this field is same as in CSD Version2.0.

- **CCC**
  Definition of this field is same as in CSD Version1.0.

- **READ_BL_LEN**
  Definition of this field is same as in CSD Version2.0.

- **READ_BL_PARTIAL**
  Definition of this field is same as in CSD Version2.0.

- **WRITE_BLK_MISALIGN**
  Definition of this field is same as in CSD Version2.0.

- **READ_BLK_MISALIGN**
  Definition of this field is same as in CSD Version2.0.

- **DSR_IMP**
  Definition of this field is same as in CSD Version2.0.

- **C_SIZE**
  This field is expanded to 28 bits and can indicate up to 128 TBytes.
  This parameter is used to calculate the user data area capacity in the SD memory card (note that size of the protected area is zero for SDUC card). The user data area capacity is calculated from C_SIZE as follows:

  memory capacity = (C_SIZE+1) * 512KByte

  The Minimum user area size of SDUC Card is 4,294,968,320 sectors (2TB+0.5MB).
  The Minimum value of C_SIZE for SDUC in CSD Version 3.0 is 0400000h (4194304).
  The Maximum user area size of SDUC Card is 274,877,906,944 sectors (128TB).
  The Maximum value of C_SIZE for SDUC in CSD Version 3.0 is FFFFFFFh (268435455).

- **ERASE_BLK_EN**
  Definition of this field is same as in CSD Version2.0.

- **SECTOR_SIZE**
  This field is fixed to 7Fh, which indicates 64 KBytes. This value is not related to erase operation. SDUC

Card indicates memory boundary by AU size and this field should not be used.

- **WP_GRP_SIZE**
  This field is fixed to 00h. SDUC Card does not support write protected groups.

- **WP_GRP_ENABLE**
  This field is fixed to 0. SDUC Card does not support write protected groups.

- **R2W_FACTOR**
  Definition of this field is same as in CSD Version2.0.

- **WRITE_BL_LEN**
  Definition of this field is same as in CSD Version2.0.

- **WRITE_BL_PARTIAL**
  Definition of this field is same as in CSD Version2.0.

- **FILE_FORMAT_GRP**
  Definition of this field is same as in CSD Version2.0.

- **COPY**
  Definition of this field is same as in CSD Version1.0.

- **PERM_WRITE_PROTECT**
  Definition of this field is same as in CSD Version1.0.

- **TMP_WRITE_PROTECT**
  Definition of this field is same as in CSD Version1.0.

- **WP_UPC**
  Definition of this field is same as in CSD Version1.0.

- **FILE_FORMAT**
  Definition of this field is same as in CSD Version2.0.

- **CRC**
  Definition of this field is same as in CSD Version1.0.

5.4 RCA register

The writable 16-bit relative card address register carries the card address that is published by the card during the card identification. This address is used for the addressed host-card communication after the card identification procedure. The default value of the RCA register is 0x0000. The value 0x0000 is reserved to set all cards into the *Stand-by State* with CMD7.

In UHS-II mode, Node ID is used as RCA. Refer to SD-TRAN Section of UHS-II Addendum for more details.

5.5 DSR register (Optional)

The 16-bit driver stage register is described in detail in Section 6.5. It can be optionally used to improve the bus performance for extended operating conditions (depending on parameters like bus length, transfer rate or number of cards). The CSD register carries the information about the DSR register usage. The default value of the DSR register is 0x404.

5.6 SCR register

In addition to the CSD register, there is another configuration register named SD CARD Configuration Register (SCR). SCR provides information on the SD Memory Card's special features that were configured into the given card. The size of SCR register is 64 bits. This register shall be set in the factory by the SD Memory Card manufacturer.

The following table describes the SCR register content.

Description	Field	Width	Cell Type	SCR Slice
SCR Structure	SCR_STRUCTURE	4	R	[63:60]
SD Memory Card - Spec. Version	SD_SPEC	4	R	[59:56]
data_status_after erases	DATA_STAT_AFTER_ERASE	1	R	[55:55]
CPRM Security Support	SD_SECURITY	3	R	[54:52]
DAT Bus widths supported	SD_BUS_WIDTHS	4	R	[51:48]
Spec. Version 3.00 or higher	SD_SPEC3	1	R	[47]
Extended Security Support	EX_SECURITY	4	R	[46:43]
Spec. Version 4.00 or higher	SD_SPEC4	1	R	[42]
Spec. Version 5.00 or higher	SD_SPECX	4	R	[41:38]
Reserved		1	R	[37]
Command Support bits	CMD_SUPPORT	5	R	[36:32]
reserved for manufacturer usage	-	32	R	[31:0]

Table 5-17 : The SCR Fields

SCR_STRUCTURE	SCR Structure Version	SD Physical Layer Specification Version
0	SCR version 1.0	Version 1.01 until current version
1-15	reserved	

Note: SD_SPEC is used to indicate SCR Structure Version instead of this field.

Table 5-18 : SCR Register Structure Version

- **SD_SPEC, SD_SPEC3, SD_SPEC4, SD_SPECX**

The Physical Layer Specification Version is indicated in combination with SD_SPEC, SD_SPEC3, SD_SPEC4 and SD_SPECX as described Table 5-19.

SD_SPEC	SD_SPEC3	SD_SPEC4	SD_SPECX	Physical Layer Specification Version Number
0	0	0	0	Version 1.0 and 1.01
1	0	0	0	Version 1.10
2	0	0	0	Version 2.00
2	1	0	0	Version 3.0X
2	1	1	0	Version 4.XX
2	1	0 or 1	1	Version 5.XX
2	1	0 or 1	2	Version 6.XX
2	1	0 or 1	3	Version 7.XX
2	1	0 or 1	4	Version 8.XX
2	1	0 or 1	5	Version 9.XX
Others				reserved

(1) Version 2.00 hosts do not recognize SD_SPEC3, SD_SPEC4 and SD_SPECX.
(2) Version 3.00 hosts do not recognize SD_SPEC4 and SD_SPECX.
(3) Version 4.00 hosts do not recognize SD_SPECX.
(4) Future versions over Version 5.00 may use SD_SPECX (ex. SD_SPECX=1 (Version 5.XX), SD_SPECX=2 (Version 6.XX), SD_SPECX=3 (Version 7.XX), SD_SPECX=4 (Version 8.XX), SD_SPECX=5 (Version 9.XX), etc.).

Table 5-19 : Physical Layer Specification Version

SD_SPEC, SD_SPEC3, SD_SPEC4, SD_SPECX fields are informative and hence may not provide any definitive information for host. Host should additionally check relevant support bits before using corresponding functions.

The card manufacturer determines SD_SPEC value by conditions indicated below. All conditions shall be satisfied for each version. The other combination of conditions is not allowed.

Essential conditions to indicate Version 1.01 Card (SD_SPEC=0, SD_SPEC3=0, SD_SPEC4=0 and SD_SPECX=0)
(1) The card does not support CMD6
(2) The card does not support CMD8
(3) User area capacity shall be up to 2GB

Essential conditions to indicate Version 1.10 Card (SD_SPEC=1, SD_SPEC3=0, SD_SPEC4=0 and SD_SPECX=0)
(1) The card shall support CMD6
(2) The card does not support CMD8
(3) User area capacity shall be up to 2GB

Essential conditions to indicate Version 2.00 Card (SD_SPEC=2, SD_SPEC3=0, SD_SPEC4=0 and SD_SPECX=0)
(1) The card shall support CMD6
(2) The card shall support CMD8
(3) The card shall support CMD42
(4) User area capacity shall be up to 2GB (SDSC) or 32GB (SDHC)
(5) Speed Class shall be supported (SDHC)

Essential conditions to indicate Version 3.00 Card (SD_SPEC=2, SD_SPEC3=1, SD_SPEC4=0 and SD_SPECX=0)

(1) The card shall support CMD6
(2) The card shall support CMD8
(3) The card shall support CMD42
(4) User area capacity shall be up to 2GB (SDSC) or 32GB (SDHC)
User area capacity shall be more than or equal to 32GB and up to 2TB (SDXC)
(5) Speed Class shall be supported (SDHC or SDXC)

Optional conditions to indicate Version 3.00 Card
A card supports any of following functions shall satisfy essential conditions of Version 3.00 Card
(1) Speed Class supported under the conditions defined in Version 3.00
(2) UHS-I supported card
(3) CMD23 supported card

Essential conditions to indicate Version 4.XX Card (SD_SPEC=2, SD_SPEC3=1, SD_SPEC4=1 and SD_SPECX=0)
(1) Same as the essential conditions of Version 3.00 device
(2) Support any of additional functions defined by Version 4.XX:
Following functions (a) to (c) are defined by Version 4.00.
(a) Support of CMD48 and CMD49
(b) Support of UHS-II mode
(c) Support of DPS
Following functions (d) to (f) are defined by Version 4.10.
(d) Support of CMD58 and CMD59
(e) Support of Power Management Functions
(f) Support of Speed Grade 1 for UHS-II mode
Following function (g) is defined by Version 4.20.
(g) Support of Speed Grade 3

Essential conditions to indicate Version 5.XX Card (SD_SPEC=2, SD_SPEC3=1, SD_SPEC4=0 or 1 and SD_SPECX=1. If at least one function of Version 4.XX is supported, SD_SPEC4=1. Otherwise, SD_SPEC4=0.)
(1) Same as the essential conditions of Version 3.00 device
(2) Support any of additional functions defined by Version 5.XX:
Following function (a) is defined by Version 5.00
(a) Support of Video Speed Class
Following functions (b) to (d) are defined by Version 5.10.
(b) Support of Application Performance Class 1
(c) Support of Discard and FULE
(d) Support of CMD42 COP Option

(3) Speed Class support is treated as optional (SDHC or SDXC)

Essential conditions to indicate Version 6.XX Card (SD_SPEC=2, SD_SPEC3=1, SD_SPEC4=0 or 1 and SD_SPECX=2. If at least one function of Version 4.XX is supported, SD_SPEC4=1. Otherwise, SD_SPEC4=0.)
(1) Same as the essential conditions of Version 3.00 device
(2) Support any of additional functions defined by Version 6.XX:
Following functions (a) to (f) are defined by Version 6.00.
(a) Support of Application Performance Class 2 [shall support (b), (c) and (d) as well]
(b) Support of Command Queue
(c) Support of Cache
(d) Support of LVS
(e) Support of Self Maintenance

(f) Support of Range C and Range D for UHS-II

Following function (g) is defined by version 6.10

(g) CPRM support is treated as optional for regular writeable SDSC, SDHC and SDXC cards

(3) Speed Class support is treated as optional (SDHC or SDXC)

Essential conditions to indicate Version 7.XX Card (SD_SPEC=2, SD_SPEC3=1, SD_SPEC4=0 or 1 and SD_SPECX=3. If at least one function of Version 4.XX is supported, SD_SPEC4=1. Otherwise, SD_SPEC4=0.)

(1) Same as the essential conditions of Version 3.00 device

(2) Support any of additional functions defined by Version 7.XX:

Following functions (a) and (b) are defined by Version 7.00:

(a) SD Express card with PCIe Gen3x1 interface

(b) Ultra capacity (SDUC) card

Following function is defined by Version 7.10:

(c) microSD support of SD Express with PCIe Gen3x1 interface.

If A2 is implemented for either card (a), (b) or (c) the card shall support Cache and LVS (Command Queuing is optional).

Essential conditions to indicate Version 8.XX Card (SD_SPEC=2, SD_SPEC3=1, SD_SPEC4=0 or 1 and SD_SPECX=4. If at least one function of Version 4.XX is supported, SD_SPEC4=1. Otherwise, SD_SPEC4=0.)

(1) Same as the essential conditions of Version 3.00 device

(2) Support any of additional functions defined by Version 8.XX:

Following function (a) is defined by Version 8.00:

(a) SD Express card supporting either PCIe Gen4 or PCIe 2 lanes

Essential conditions to indicate Version 9.XX Card (SD_SPEC=2, SD_SPEC3=1, SD_SPEC4=0 or 1 and SD_SPECX=5. If at least one function of Version 4.XX is supported, SD_SPEC4=1. Otherwise, SD_SPEC4=0.)

(1) Same as the essential conditions of Version 3.00 device

(2) Support any of additional functions defined by Version 9.XX:

Following functions (a) to (c) are defined by Version 9.00:

(a) Support of TCG security

(b) Support of RPMB

(c) Support of Boot Functionalities and RPMB

Note that when Boot Functionalities are supported, Fast Boot (for SD interface) shall be also supported.

Following function (d) is defined by Version 9.10:

(d) SD Express Speed Class

Application Notes: Note that Version 5.XX specification onwards the Speed Class 2 support of the card is modified from mandatory to optional while it was treated as mandatory in Version 2.00, 3.00 and 4.XX.

The requirements of supporting commands mentioned above are for the optional commands, the support of which depends on versions (SD_SPEC, SD_SPEC3, SD_SPEC4 and SD_SPECX). Refer to Table 4-21 (and Notes below the table) about the mandatory and optional commands in the card.

- **DATA_STAT_AFTER_ERASE**
  Defines the data status after erase, whether it is 0 or 1 (the status is card vendor dependent).

- **SD_SECURITY**
  This field indicates CPRM Security Specification Version for each capacity card. The definition of Protected Area is different in each capacity card.

SD_SECURITY	CPRM Security Version
0	No Security
1	Not Used
2	SDSC Card (Security Version 1.01)
3	SDHC Card (Security Version 2.00)
4	SDXC Card (Security Version 3.xx)
5 - 7	Reserved

Table 5-20 : CPRM Security Version

If Security is supported, the basic rule of setting this field:
SDSC Card sets this field to 2 (Version 1.01)
SDHC Card sets this field to 3 (Version 2.00).
SDXC Card sets this field to 4 (Version 3.xx).
Note that security is optional for writeable SDSC, SDHC and SDXC SD Memory cards and ROM and OTP cards. SDUC card does not support security feature. SD_SECURITY shall be set to 0.

- **SD_BUS_WIDTHS**
  Describes all the DAT bus widths that are supported by this card.

SD_BUS_WIDTHS	Supported Bus Widths
Bit 0	1 bit (DAT0)
Bit 1	reserved
Bit 2	4 bit (DAT0-3)
Bit 3	reserved

Table 5-21 : SD Memory Card Supported Bus Widths

Since the SD Memory Card shall support at least the two bus modes 1-bit or 4-bit width, then any SD Card shall set at least bits 0 and 2 (SD_BUS_WIDTH="0101").

- **EX_SECURITY**
  This field indicates Extended Security which is defined by the Part A4 Data Protection System Specification Version 1.00 or other extend security specifications.

EX_SECURITY	Extended Security
0000b	Extended Security is not supported.
Others	Extended Security is supported. SCR[44:43] is defined by the Part A4 Data Protection System Specification. SCR[45] is used for TCG security (see Table 5.6-1). SCR[46] is used for RPMB (see Table 5.6-1).

Table 5-22 : Extended Security

Table 5.6-1 shows support bits for extended securities (RPMB and TCG security) introduced from the Physical Layer Specification Version 9.00.
SCR Bit 46 indicates support bit of RPMB. If the card supports RPMB, RPMB_SUPPORT=1, otherwise, RPMB_SUPPORT=0.
SCR Bit 45 indicates support bit of TCG security. If the card supports the Security Subsystem Class as defined in Extended Security Addendum, TCG_SUPPORT=1. Otherwise, TCG_SUPPORT=0.

Description	Field	Width	Cell Type	SCR Slice
RPMB Support	RPMB_SUPPORT	1	R	[46]
TCG Support	TCG_SUPPORT	1	R	[45]

Table 5.6-1 : Detailed Field Definition of EX_SECURITY

- **CMD_SUPPORT**

Support bit of new commands are defined to Bit 36-32 of SCR.

SCR Bit	Supported Command	Command	CCC	Remark
36	Secure Receive / Secure Send	ACMD53/54	8	Optional. If TCG security or RPMB are supported, then ACMD53/54 shall be supported.
35	Extension Register Multi-Block	CMD58/59	11	Optional. If CMD58/59 is supported, then CMD48/49 shall be supported.
34	Extension Register Single Block	CMD48/49	11	Optional. If RPMB is supported (including the case of Boot Functionalities support), then CMD48/49 shall be supported.
33	Set Block Count	CMD23	2, 4	Mandatory for UHS104 card. If ACMD53/54 is supported, then CMD23 shall be supported.
32	Speed Class Control	CMD20	2, 4	Mandatory for SDXC and SDUC card if Speed Class, UHS Speed Grade or Video Speed Class is supported by the card.

Table 5-23 : Command Support Bits

5.7 Function Extension Specification

There are demands for adding extension functions in the SD Memory Card. With regard to CMD6, it is not suitable for active control but suitable for selecting one of functions at initialization. Then new function extension method suitable for active control is introduced.
Extension Register is introduced to control extension functions. The Extension Register space is independent to Memory Space accessed by CMD17/18 and CMD24/25. Commands CMD48/49 and CMD58/59 are defined to access Extension Register Space.

CMD48: Read Extension Register Single Block Command
CMD49: Write Extension Register Single Block Command
CMD58: Read Extension Register Multi-Block Command
CMD59: Write Extension Register Multi-Block Command

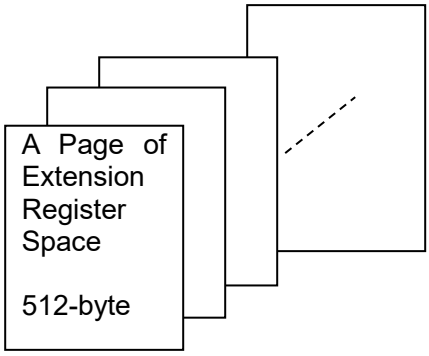
There are two types of Extension Register Spaces; Memory Extension Register Space and I/O Extension Register Space. I/O Extension Register Space is equivalent to the space accessed by CMD52 and CMD53 as defined by the SDIO Specification.
As most host systems can perform 512byte block basis read/write operation, CMD48/49 are defined as 512 bytes fixed block length commands. "Data Port" can be defined in the Extension Register space to perform data transfer between host and a function device. Location of Data Port is defined by each function specification. CMD49 has specific features to perform bit operation without "Read Modify Write". Multiple blocks data transfer is supported by CMD58/59 to increase performance.
This function extension method also defines "General Information" to realize Plug & Play in a host system. General Information includes information to find function driver.

Host shall set reserved bits in a defined writable Extension Register to 0. Even if reserved bits were set to 1, Device is not necessary to set reserved bits to 1.

Function Extension Specification is newly added to the Physical Layer Specification Version 4.00 and is applicable from the Physical Layer Specification Version 4.00 products. This function may be supported on SDHC, SDXC and SDUC Card and is not supported through SPI interface. From the Physical Layer Version 4.10, Multi-Block commands are supported.
From the Physical Layer Specification Version 9.00, the Function Extension realized by CMD48/49/58/59 is also available in SDSC cards to support Boot Functionalities and RPMB. Except these two functions, SDSC card cannot support Function Extension such as Power Management Function. In addition, data block size transmitted by CMD48/49/58/59 is always 512 bytes regardless of BLOCK_LEN value or sector size.

5.7.1 Extension Register Space

Figure 5-3 shows Extension Register Space. The space is divided into 256 pages per a Function Number. Each page is 512 byte length of fixed address boundary. Function Number is 1 to 15 in case of Memory Space and 1 to 7 in case of I/O Space. CMD48/49 cannot access data across page boundary.
Embodiment of Extension Register Set is defined anywhere in the Extension Register Space by a Function Specification. Extension Register Set should be treated as a different type of register from that shown in Table 3-2.



Extension Register Space consists of 256 pages per function.

Figure 5-3 : Extension Register Space

5.7.2 Extension Register Commands

5.7.2.1 Extension Register Read Command (Single Block)

Figure 5-4 shows definition of Read Extension Register Single Block Command (CMD48). Bus timing of this command is equivalent to a single block read command (CMD17). The response type is R1. Data block length is fixed to 512-byte length.

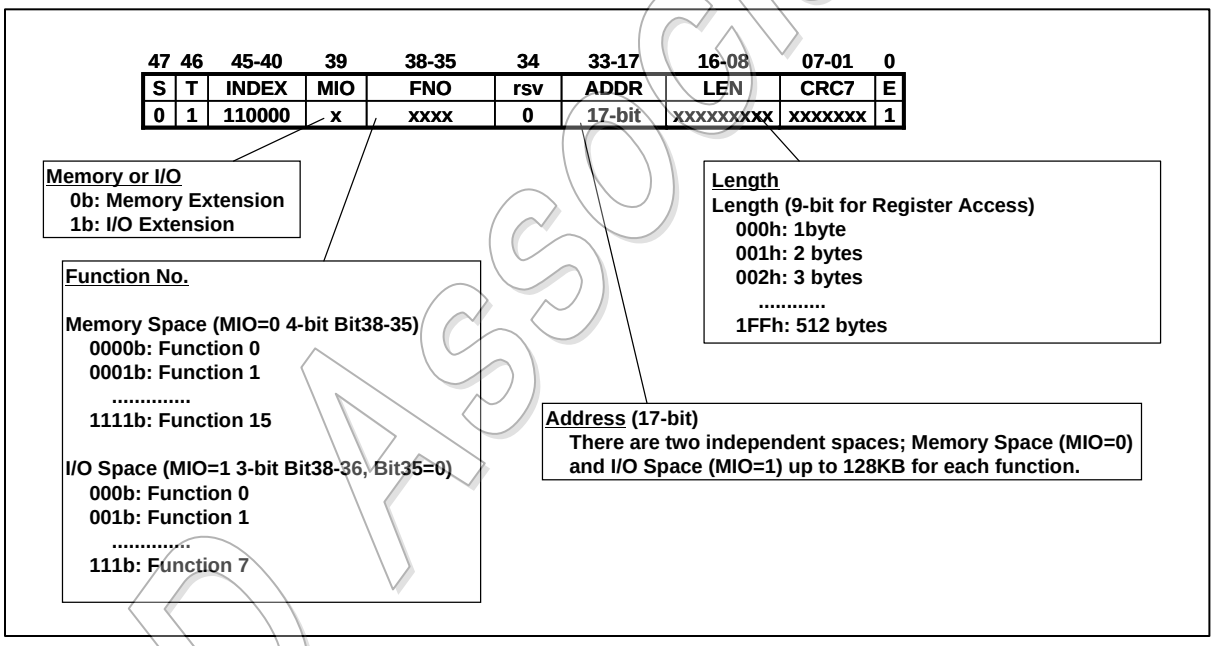


Figure 5-4 : Read Extension Register Single Block Command (CMD48)

Fields in the argument of CMD48:

MIO: Selection of Memory Space or I/O Space (0: Memory Space, 1: I/O Space)
There are two independent spaces; Memory Extension Register Space and I/O Extension Register Space. Which space accessed is selected by this bit. I/O Extension Register Space is equivalent to CMD52/53 SDIO space. (Refer to Part E1 SDIO Specification and Part E7 iSDIO Specification.)

FNO: Function Number
Unique Function Number is assigned to each function. FNO helps host to distinguish functions

and enables card to check sequence of commands. Up to 15 functions may be assigned for memory and up to 7 functions may be assigned for I/O.

ADDR: Address of Extension Register Space

Lower 9-bit is used as offset address in a page and upper 8-bit is used as page number. Up to 256 pages is addressable.

LEN: Effective Length of a Page

On accessing Data Port, card ignores this field but host shall set this field to 000h. If access is not to Data Port, this field is used to set effective data length (byte unit) in a page.

Figure 5-5 shows read timing of Extension Register. The register area for read is specified by ADDR and LEN and shall be in a page boundary. The register data is set from the top of 512-byte data block and the rest is filled with dummy data. If data length is specified over a page boundary (512 bytes), surplus data in next page is not read and not set in the data block. The maximum data access time from the end bit of CMD48 is 1 second. Data zero shall be read when reading from unassigned register area which is in the register sets assigned by the General Information.

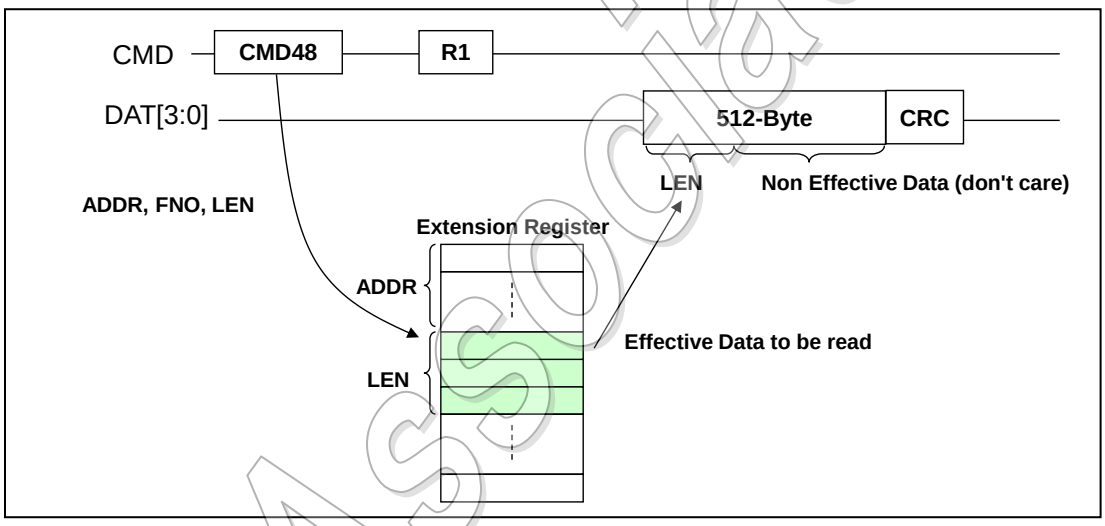


Figure 5-5 : Extension Register Read Operation by CMD48

Extension Register can be used to interface a function device. Use of Data Port is useful to perform data transfer between host and the device. Figure 5-6 shows read timing of Data Port. Data Port can be defined anywhere in Extension Register space by implementation of the card. If FNO and ADDR are matched to the Data Port location (1 byte), a data of the function device can be read. Data from the device is set in the data block of CMD48. Data length is fixed 512 bytes. The maximum data access time of Data Port from the end bit of CMD48 is 1 second.

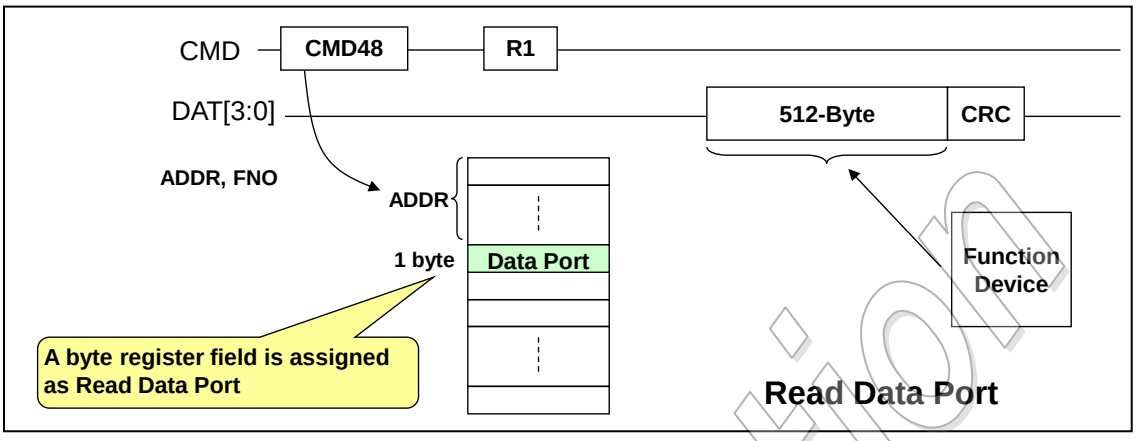


Figure 5-6 : Data Port Read Operation by CMD48

5.7.2.2 Extension Register Write Command (Single Block)

Figure 5-7 shows definition of Write Extension Register Single Block Command (CMD49). Bus timing of this command is equivalent to a single block write command (CMD24). The response type is R1. Data block length is fixed to 512-byte length.

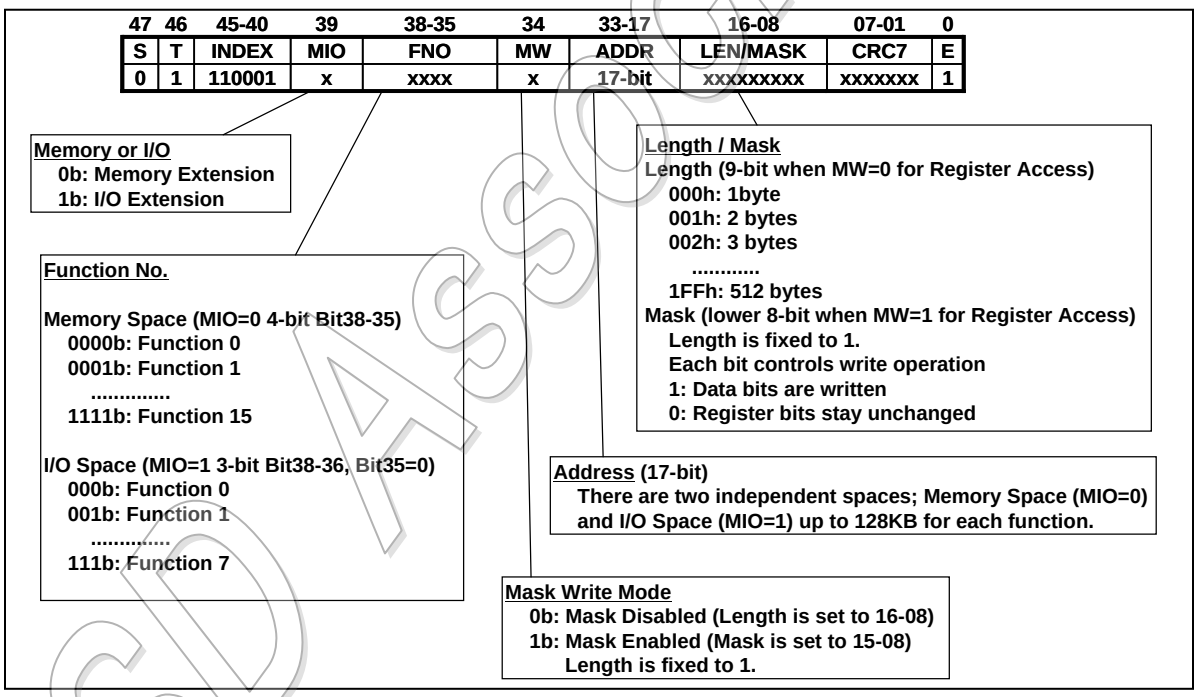


Figure 5-7 : Write Extension Register Single Block Command (CMD49)

Fields in the argument of CMD49:

MIO: Selection of Memory Space or I/O Space (0: Memory Space, 1: I/O Space)
There are two independent spaces; Memory Extension Register Space and I/O Extension Register Space. Which space accessed is selected by this bit.

FNO: Function Number
Unique Function Number is assigned to each function. FNO helps host to distinguish functions

and enables card to check sequence of commands. Up to 15 functions may be assigned for memory and up to 7 functions may be assigned for I/O.

MW: Mask Write Operation (0: Mask is disabled, 1: Mask is enabled)

On accessing Data Port, card ignores this field but host shall set this bit to 0. If access is not to Data Port, this bit determines the meaning of LEN/MASK field.
If this bit is set to 0, LEN/MASK field is used to specify data length.
If this bit is set to 1, LEN/MASK field is used to specify mask data and data length is considered as 1 byte. Destination in Extension Register is specified by FNO and ADDR. The first byte in the data block is used for a write data. Bit operation between destination and write data is dependent on mask data. If a mask bit is set to 0, a bit of the destination correspondent stays unchanged. If a mask bit is set to 1, a bit of write data correspondent is written to the destination.

ADDR: Address of Extension Register Space

Lower 9-bit is used as offset address in a page and upper 8-bit is used as page number. Up to 256 pages is addressable.

LEN/MASK: Effective Length of a Page

On accessing Data Port, card ignores this field but host shall set this field to 000h. If access is not to Data Port, meaning of this field is dependent on MW.
If MW=0, this field is used for setting effective data length (byte unit) in a page. Effective data is started from top of data block and only effective data specified by this length is written to Extension Register and the rest of block data is discarded and not written. This feature is useful to avoid read modify write operation.
If MW=1, this field is used for setting mask data (1 byte) and perform mask write operation.

Figure 5-8 shows write timing of Extension Register. The register area for write is specified by ADDR and LEN, and shall be in a page boundary. The other register area is not affected by this write operation. The write data is set from the top of 512-byte data block and the rest is filled with dummy data. The write data is written to Extension Register area specified by FNO and ADDR. If the register area is specified to across page boundary (512 bytes), data of next page is not written.
Mask write operation is supported to enable bit operation with mask data (MW=1). Data length is always 1 byte. Host is not necessary to perform read modify write operation by using mask operation. A byte write data is set in the top of 512-byte data block. 8-bit mask in the argument is corresponded to the byte write data. Setting Mask to 1 writes a write data bit to the register. Setting Mask to 0 prevents to change a register bit. Busy is indicated until CMD49 operation is completed. The maximum busy length to write Extension Register is 1 second.

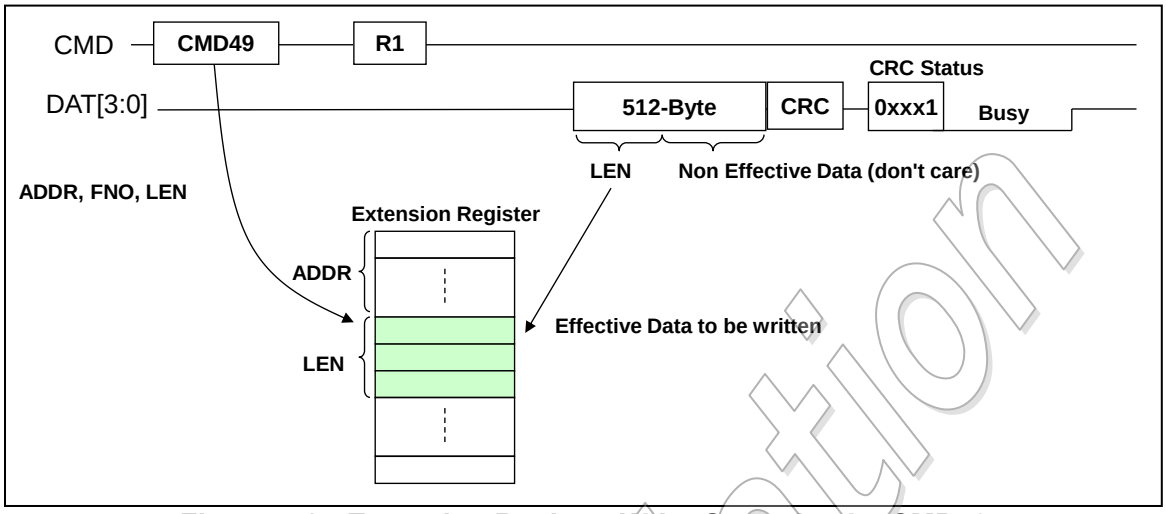


Figure 5-8 : Extension Register Write Operation by CMD49

Figure 5-9 shows write timing of Data Port. Extension Register is used to interface a function device. If FNO and ADDR are matched to the Data Port location (1 byte), a 512-byte data block is sent to the function device. The data is not written to Extension Register and Extension Register is not affected by this Data Port write operation. The maximum busy length to write Data Port is 1 second.

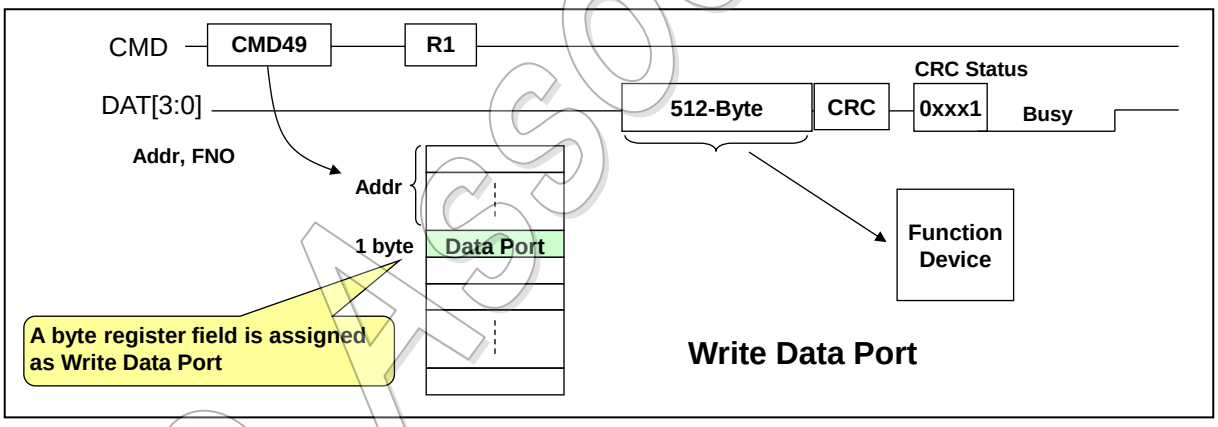


Figure 5-9 : Data Port Write Operation by CMD49

5.7.2.3 Multiple Block Data Transfer

Multi-block data transfer can be supported by using CMD58/59. CMD58/59 are optional and SCR bit 35 is assigned as the support bit of CMD58/59. If CMD58/59 are supported, CMD48/49 shall be supported.

(1) Accessible Location

CMD58/59 are mainly used to transfer multiple pages of data through Data Port and also may be used to access multiple pages of Extension Register. All 17-bit Address in the argument is effective to recognize Data Port location. If the address is not Data Port, the address of CMD58/59 should indicate top of page boundary of Extension Register (Card masks Lower 9-bit address to 0). If a target address becomes illegal, operation will stop. For example, if data transfer encounters a boundary between Data Port and Non Data Port, or function boundary, the card may terminate the data transfer.

(2) Total Data Length

As there is not enough space in the argument to specify total data length, concept of "**Block Unit**" is introduced. Total data length is determined by (Block Unit) x (Block Unit Count).

Two types of Block Units can be selected by **Block Unit Select** (BUS bit-34 in the argument) either 512Bytes (BUS=0) for smaller data transfer or 32KBytes (BUS=1) for larger data transfer. 32KBytes is adopted for considering a cluster size of FAT file system. 32KByte Block Unit means that 64 blocks (64 x 512Bytes) is considered as a data transfer unit. **Block Unit Count** is assigned to Bit 16-08 in the argument. Total Data length is calculated as follows:

BUS=0: 512 Byte (Block Unit) x Block Unit Count (BUC)

BUS=1: 32KByte (Block Unit) x Block Unit Count (BUC)

Block Length is fixed to 512 bytes regardless of Block Unit.

(3) Busy Length and Access Time

Maximum busy length for each busy period and maximum data access time for each read data block are defined as 1 second. Card should follow Function Specification if it defines shorter Busy Length and Access Time.

(4) Abort Operation

CMD12 is used to force the card to go back to "tran" state. Host may issue CMD12 on detecting timeout.

5.7.2.4 Extension Register Read Command (Multi-Block)

CMD58 is assigned as Read Multi-Block Command as shown in Figure 5-10.

BUS and BUC are explained in Section 5.7.2.3 and the definition of other fields is equivalent to CMD48 in Section 5.7.2.1.

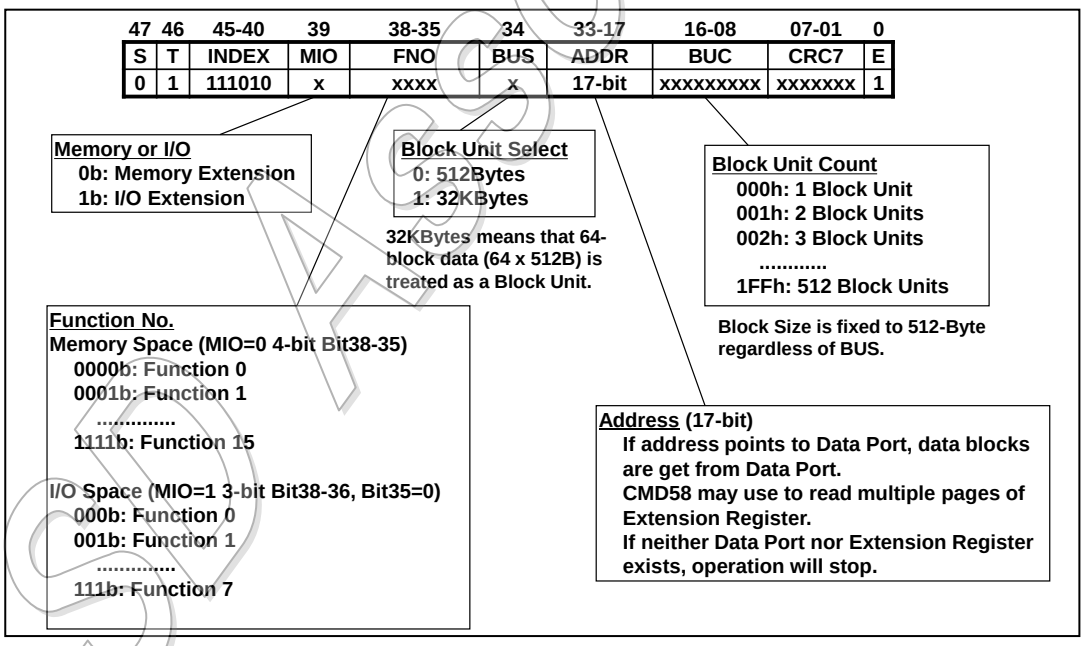


Figure 5-10 : Read Extension Register Multi-Block Command (CMD58)

Figure 5-11 shows CMD58 read timing from Extension Register and Data Port. Bus timing of this command is equivalent to a multi-block read command (CMD18). Data zero shall be read when reading from unassigned register area which is in the register sets assigned by the General Information.

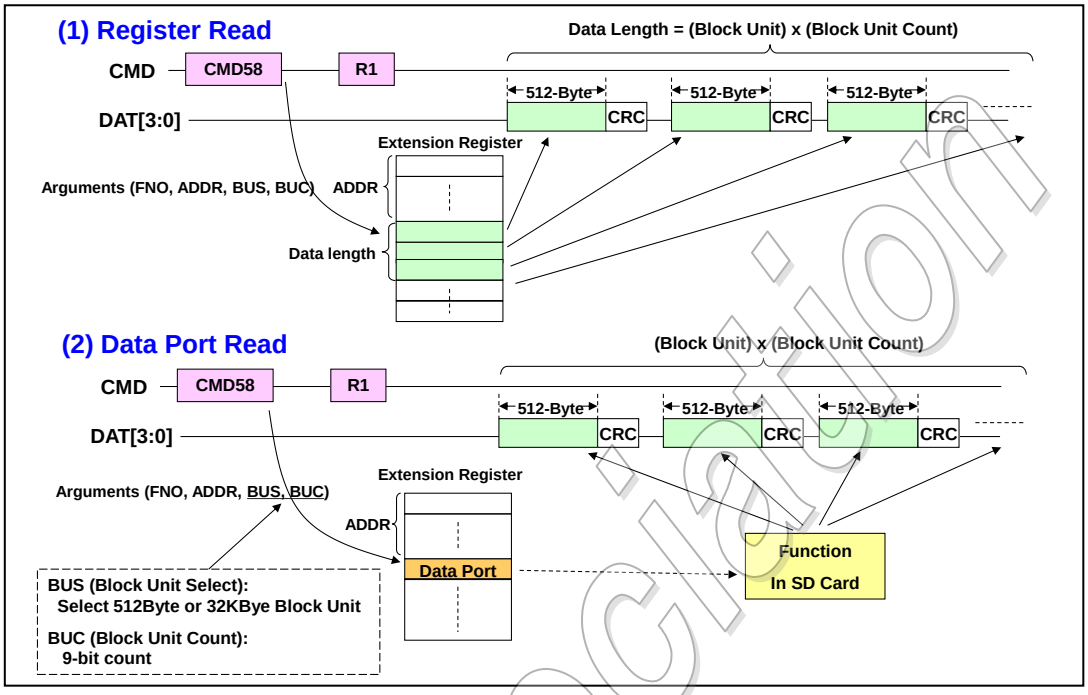


Figure 5-11 : Extension Register and Data Port Read Operation by CMD58

5.7.2.5 Extension Register Write Command (Multi-Block)

CMD59 is assigned for Write Multi-Block Command as shown in Figure 5-12.
BUS and BUC are explained in Section 5.7.2.3 and the definition of other fields is equivalent to CMD49 in Section 5.7.2.2.

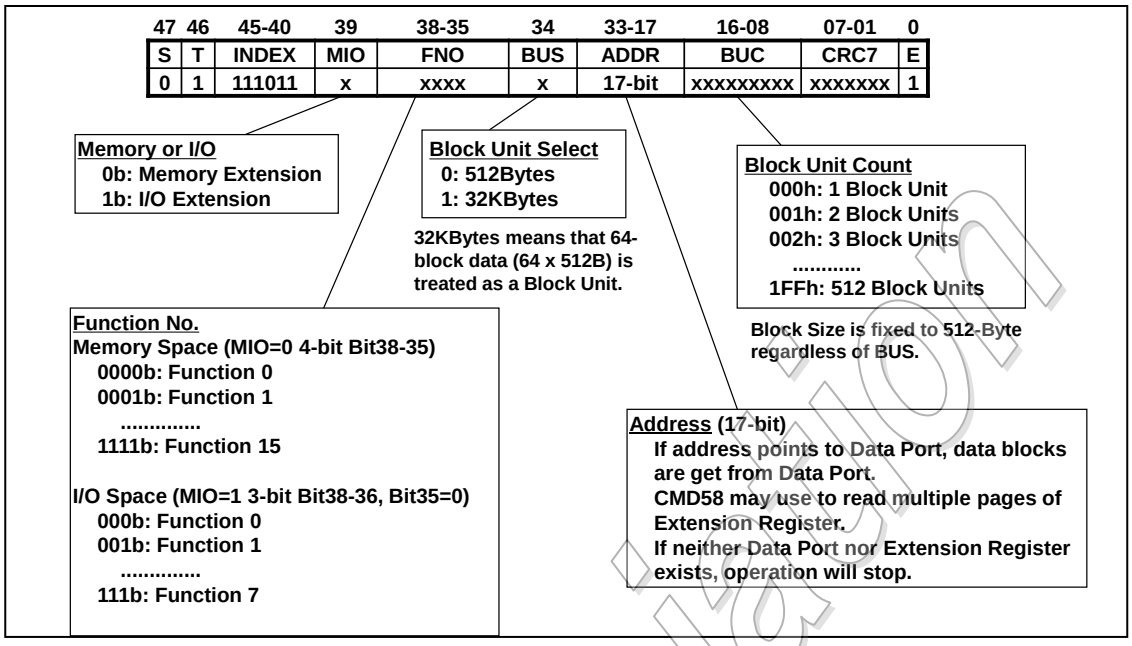


Figure 5-12 : Write Extension Register Multi-Block Command (CMD59)

Figure 5-13 shows CMD59 write timing to Extension Register and Data Port. Bus timing of this command is equivalent to a multi-block write command (CMD25).

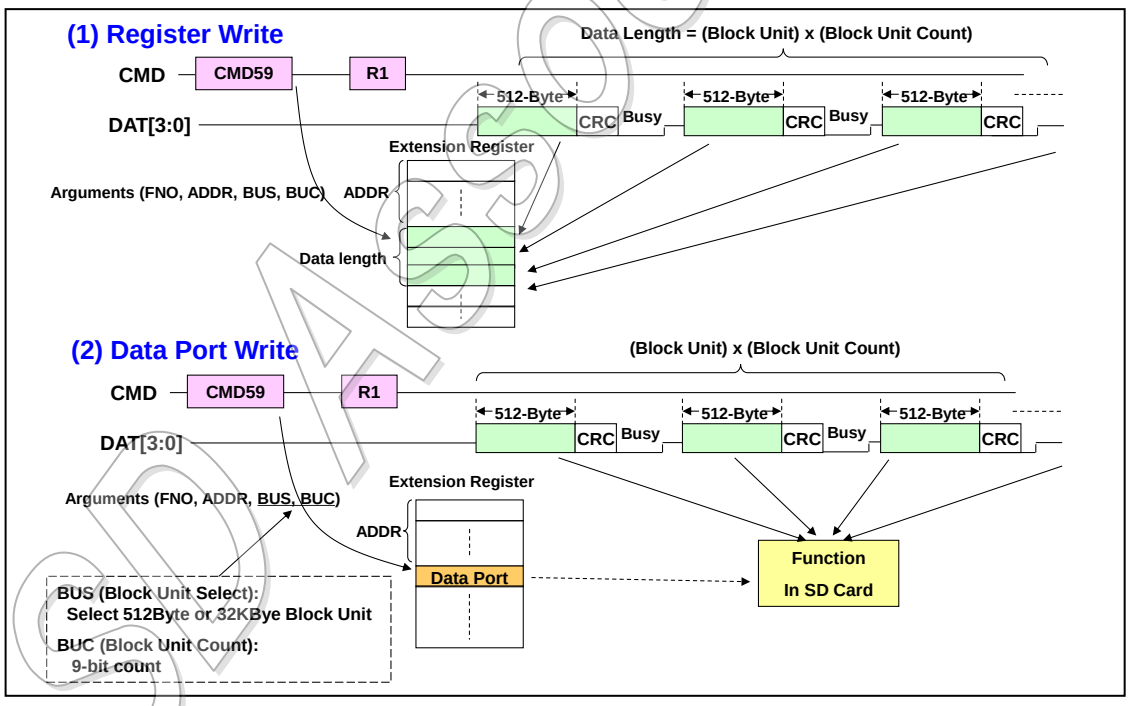


Figure 5-13 : Extension Register and Data Port Write Operation by CMD59

5.7.2.6 Error Status Indication

On detecting Illegal Command Error or Command CRC Error, the card indicates the error to the next R1 response. Backend errors of Extension Functions (ex. network error of I/O function, etc.) shall not be indicated in R1 (backend errors are reserved for memory) and shall be handled by using Error Status

register, which is defined by each function specification. Interface errors (ex. illegal argument of CMD48/49/58/59) may be indicated to R1 or Error Status register.

5.7.3 General Information

Extended Function is supposed to be controlled by a Function Driver on a host system. Host Driver is responsible to find and load the pre-installed Function Driver to use a function on a SD card. Host System can use any function by this method. General Information is defined to realize Plug & Play in Host System. There are two types of Function Drivers. One is Standard Function Driver for Controlling Standard Register Set that will be provided by OS vendor or Host System vendor. The other is Particular Function Driver for Controlling Specific Register Set that is provided by the vendor who made a Function.

Figure 5-14 shows location and size of General Information for Memory (MIO=0). General Information is located from page 0 of FNO=0 and the length of General Information is variable up to 1024 bytes (up to page 1) depends on the number of functions supported on a card. The length (in byte) is indicated in a field of General Information.

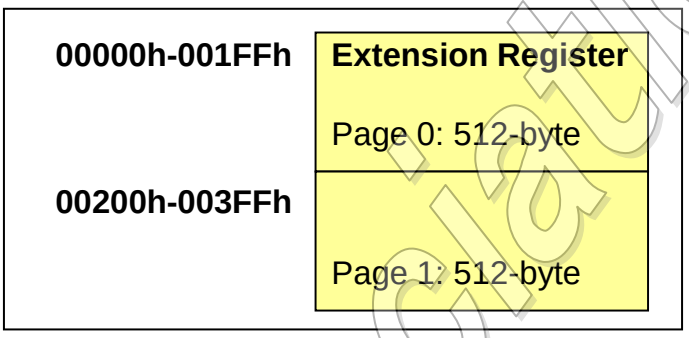
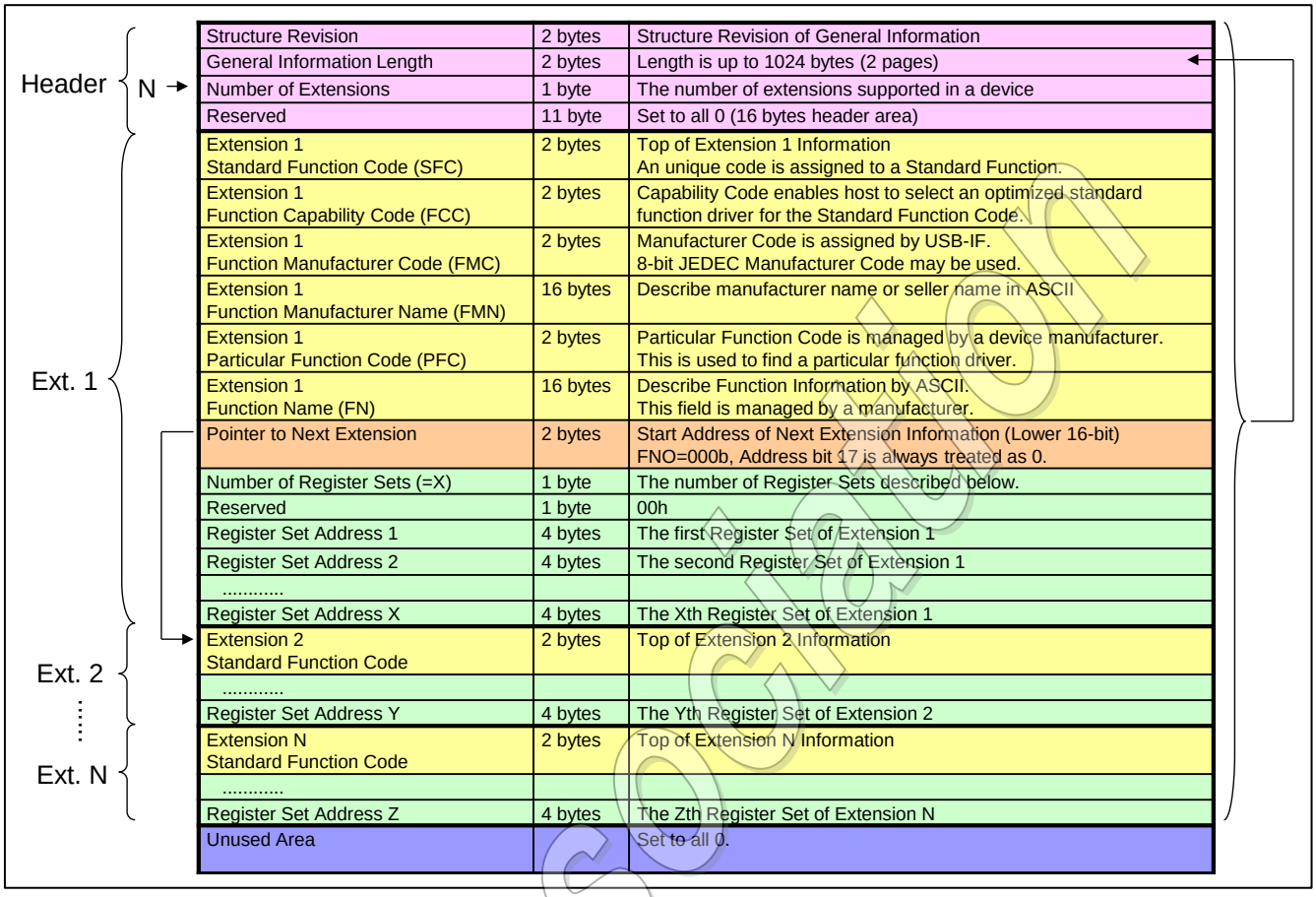


Figure 5-14 : General Information for Memory

Location of General Information for I/O is defined by Part E7 iSDIO Specification Version 1.00. Format of General Information is same as memory.

Figure 5-15 shows Data Structure of General Information. All fields are defined as little endian format. Least significant byte is indicated in lower byte.
The first 16 bytes are assigned as header of General Information. After that multiple of Extensions are described. An Extension is specified by from Standard Function Code field to the last Extension Address field.



Structure Revision	2 bytes	Structure Revision of General Information
General Information Length	2 bytes	Length is up to 1024 bytes (2 pages)
Number of Extensions	1 byte	The number of extensions supported in a device
Reserved	11 byte	Set to all 0 (16 bytes header area)
Extension 1 Standard Function Code (SFC)	2 bytes	Top of Extension 1 Information An unique code is assigned to a Standard Function.
Extension 1 Function Capability Code (FCC)	2 bytes	Capability Code enables host to select an optimized standard function driver for the Standard Function Code.
Extension 1 Function Manufacturer Code (FMC)	2 bytes	Manufacturer Code is assigned by USB-IF. 8-bit JEDEC Manufacturer Code may be used.
Extension 1 Function Manufacturer Name (FMN)	16 bytes	Describe manufacturer name or seller name in ASCII
Extension 1 Particular Function Code (PFC)	2 bytes	Particular Function Code is managed by a device manufacturer. This is used to find a particular function driver.
Extension 1 Function Name (FN)	16 bytes	Describe Function Information by ASCII. This field is managed by a manufacturer.
Pointer to Next Extension	2 bytes	Start Address of Next Extension Information (Lower 16-bit) FNO=000b, Address bit 17 is always treated as 0.
Number of Register Sets (=X)	1 byte	The number of Register Sets described below.
Reserved	1 byte	00h
Register Set Address 1	4 bytes	The first Register Set of Extension 1
Register Set Address 2	4 bytes	The second Register Set of Extension 1
.....		
Register Set Address X	4 bytes	The Xth Register Set of Extension 1
Extension 2 Standard Function Code	2 bytes	Top of Extension 2 Information
.....		
Register Set Address Y	4 bytes	The Yth Register Set of Extension 2
Extension N Standard Function Code	2 bytes	Top of Extension N Information
.....		
Register Set Address Z	4 bytes	The Zth Register Set of Extension N
Unused Area		Set to all 0.

Figure 5-15 : Data Structure of General Information

5.7.3.1 Common Header Fields

5.7.3.1.1 Structure Revision (2-byte)

This field indicates **Structure Revision** of General Information. If a new field is added and structure is modified, a value of this field is incremented. A new field is added after the last field of a function. Driver shall calculate next function location by using Pointer to Next Extension. 0000h is indicated in this version.

5.7.3.1.2 General Information Length (2-byte)

This field indicates the length of General Information in byte unit. If the length is over 512 bytes, two pages are used to describe General Information. The maximum length is defined as 1024 bytes.

5.7.3.1.3 Number of Extensions (1-byte)

This field indicates the number of extension functions supported on a card. In case of Memory, the number of function is 1 to 15. In case of I/O, the number of function is 1 to 7.

5.7.3.2 Function Fields per Function

5.7.3.2.1 Standard Function Code (SFC 2-byte)

This field is used to find a Standard Function Driver. Non Standard Function sets this field to 0000h.

Code assignment is independent between memory and SDIO.

5.7.3.2.2 Function Capability Code (FCC 2-byte)

Host Driver uses this field to select one of Standard Function Drivers when different types of Standard Function Drivers exist to a **Standard Function Code**. Function Specification may define options in this field to distinguish driver types. How to use this field is up to host system implementation. Setting 0000h to this field means that Host Driver selects a Standard Function Driver by a **Standard Function Code** without using this field.

5.7.3.2.3 Function Manufacturer Code (FMC 2-byte)

This field indicates a manufacturer code and is used to find a Particular Function Driver. Setting of this field to 0000h means that a defined standard function is not dependent on vendors.
The FMC field identifies the SDIO Card's manufacturer. A two-byte code (2nd byte as non-zero) is assigned by the USB Implementers Forum (USB-IF). The code with 2nd byte zero is reserved for manufacturers who have an eight-bit JEDEC manufacturer code assigned by JEDEC Publication 106. Manufacturers may use their eight-bit JEDEC manufacturer code as the 1st byte of FMC. For example, if a JEDEC manufacturer code is 89h, its FMC is 0089h. USB-IF has all responsibility for managing manufacturer code including 2 byte codes which are assigned by the former PCMCIA organization. Two byte manufacturer codes assigned by USB-IF and by PCMCIA can be used for FMC. If a manufacturer does not currently have a FMC assigned, they should request assignment of a new manufacture code from USB-IF by mailing to admin@usb.org.

5.7.3.2.4 Function Manufacturer Name (FMN 16-byte)

This field is used to describe manufacturer name or seller name by ASCII character that is correspondent to Function Manufacturer Code. Unused area is filled with 00h. Common Name of an organization may be used.

5.7.3.2.5 Particular Function Code (PFC 2-byte)

This field is used to find a Particular Function Driver and definition of code is managed by a vendor described in FMC and FMN. If Particular Function Driver is not supported, this filed is set to 0000h.

5.7.3.2.6 Function Name (FN 16-byte)

This field is used to describe Function Description by ASCII character. Unused area is filled with 00h.

5.7.3.2.7 Pointer to Next Extension (2-byte)

This field indicates offset address of next function location from the top of General Information. Lower 10-bit is used to indicate offset and upper 6-bit shall be set to 0.
Driver shall calculate location of next function information by this field because a new field may be added at the end of function information. Then the current last field may not be the last one in future. The last function sets this field to 0000h. The last function number N is set to **Number of Extensions** field.

5.7.3.2.8 Number of Register Sets (1-byte)

Multiple of register set can be assigned per a function. This field indicates number of register sets in the function specified by FNO. A list of Extension Register Set Address follows after this field. The first register set is supposed to be started from top of a function space.

5.7.3.2.9 Extension Register Set Address (4-byte for each)

This field indicates FNO and start address of a register set. Assignment of this 4-byte field is shown in

Table 5-24.

31-22	21-18	17	16-00
10-bit	4-bit	1-bit	17-bit
Reserved (0000000000b)	FNO	0	Start Address of a Register Set

Bit 21-18: 4-bit FNO for Memory Space
Bit 21-19: 3-bit FNO for I/O Space. (Bit 18=0).

Table 5-24 : Field Definition of Extension Register Set Address

Refer to Appendix F.1 about how to identify Function Driver.

5.7.4 Revision Management

A function should assign Revision Register in the Function Extension Register Set to indicate the function revision. Functionality of the function may be extended by increasing the function revision. Higher revision shall include all functionality of lower revision to keep compatibility. Function Driver has driver revision which is correspond to the function revision. Any revision of function shall be able to be used with any revision combination of function driver. By reading the Revision Register, Function Driver knows usable functionality which is determined by lower revision of either function or driver.

5.7.5 Event Indication Method

5.7.5.1 FX_EVENT (Bit06 of Card Status)

An event indication method is introduced from Version 4.20. Card may use Bit06 of R1 (R1b) response of any SD command to indicate event generation. The Bit06 is named FX_EVENT (Function Extension Event). By using this method, SD Memory Card can notify host of event generation to get a driver to deal with the event. Support of host is required to detect events. This method is able to reduce frequency of polling considerably rather than each Function Driver executes polling. An example implementation of Host
Event Detection is described in Appendix F.2 .

5.7.5.2 Function Extension Event (FXE) Register Set

As multiple of functions can be supported by using Function Extension feature, card needs to provide information which function generates an event. For this purpose, "Function Extension Event (FXE) Register Set" is placed just after the location of General Information (Space of FNO=0). Table 5-25 shows the format of the FXE Register Set.

Offset	Register Name	Symbol	Type		
0	Page Revision = 01h	FXE_REV	Read only	Register	Standard
1	Number of Functions using FX_EVENT	FXE_NUM	Read only	Register	Standard
2	Function Event Flag (7-0)	FXE_FEFL	Read only	Register	Standard
3	Function Event Flag (15-8)	FXE_FEFH	Read only	Register	Standard

Table 5-25 : Function Extension Event Register

- **FXE_REV (Offset 0: Page Revision)**
  This revision will increase if FXE Register Set is modified. The register set shall be backward compatible and host needs to accept larger revision setting even currently not defined.
  00h: FXE Register Set is not supported.
  01h: FXE Register is supported.

02h-FFh: Reserved.

- **FXE_NUM (Offset 1: Number of Functions using FX_EVENT)**
  Lower 4-bit indicates the number of functions using FX_EVENT. Upper 4-bit is set to 0000b as reserved field.
  00h: There is no function to use FX_EVENT. Another method may be used to detect events.
  01h: A function uses FX_EVENT even card has multiple functions. Only one bit of FXE_FEFx may be set to 1 as fixed value to simplify card implementation (changing the bit is not required). The bit number of FXE_FEFx set to 1 represents which function number uses FX_EVENT.
  02h-0Fh: Multiple of functions will use FX_EVENT. The card needs to set and clear FXE_FEFx.

- **FXE_FEFx (Offset 2, 3: Function Event Flag)**
  FXE_FEFx represents two 8-bit registers FXE_FEFL and FXE_FEFH. Content of this register is valid when FX_EVENT is set to 1. Each bit is correspondent to a function number (FNO 0 to 15) which generates events. If FXE_NUM=01h, this register may be set to a fixed value so that only one bit of this field is set to 1. If FXE_NUM>01h, each of this register bit is set and cleared by "Event Status Register" (refer to next section) defined in a function number of Extension Register Space.
  Bit=1: A function of correspondent function number generates an event
  Bit=0: A function of correspondent function number does not generate any event

Table 5-26 shows summary of setting combination of FXE Register Set.

FXE Register Fields			R1 Bit 06	Note
FXE_REV	FXE_NUM	FXE_FEFx	FX_EVENT	
00h	00h	Not used	Not supported	There is no function using FX_EVENT
>= 01h	01h	Fixed value	Supported	A unique function uses FX_EVENT
	> 01h	Set / Clear	Supported	Multi-functions use FX_EVENT

Table 5-26 : Setting Combination of FXE Register Set

5.7.5.3 Event Status Register

A Function that has capability to generate events need to assign "Event Status Register" in an Extension Register Space of a correspondent function number. This register will indicate details of event information so that Function Driver can solve the events and card will use it to control set/clear of FXE_FEFx register. If this register indicates generation of any event, correspondent bit of FXE_FEFx is set to 1. The bit is cleared when Function Driver deals with the events. If there are no events unsolved, correspondent bit of FXE_FEFx is cleared. If a new event is generated, status in this register and a bit of FXE_FEFx will be set again. FXE_NUM=01h may be treated as exceptional by setting fixed value in FXE_FEFx to simplify card implementation. Once FX_EVENT is set to 1, host should repeat event service at some interval until all events are solved.

5.7.5.4 FX_EVENT Enable

Each Function needs to have "FX_EVENT Enable" to enable event indication on Bit06 of R1. Setting of FX_EVENT is disabled in default. "FX_EVENT Enable" control bit will be assigned to each function's Extension Register Space and it is enabled during a function initialization.

5.8 Application Specification on Function Extension

All application specifications defined in this section are assigned to Memory Extension Register Space. Assignment of Standard Function Code for memory is managed by Table 5-27.

Function Name	Standard Function Code
Power Management Function	0001h
Performance Enhancement Function	0002h
Security and Boot Function	0003h

Table 5-27 : Standard Function Code Assignment Table

5.8.1 Power Management Function

5.8.1.1 Abstract of Power Management Function

This Section defines three power management functions on Extension Register.

(1) Features of Power Off Notification

Power Off Notification is defined to ensure higher reliability and safety for data stored in the SD Memory Card. Host should use Power Off Notification (if card support it) before shut down the card power to avoid problems likely occurs by sudden power off.

(2) Features of Power Sustenance

Card may request host to sustain card power as much as possible to improve card performance. If host accept power sustenance, the card may operate even while host does not use the card. Once host accepts power sustenance, host should keep card power as much as possible until host system power down.

(3) Features of Power Down Mode

Host may let the card to go into lower power consumption mode while host does not intend to use the card (Power consumption level is dependent on implementation.). The card can be used again without re-initialization by retrieving from Power Down Mode (in tran state).

5.8.1.2 Extension Register Set for Power Management

Table 5-28 shows Power Management Register Set consists of 3 bytes registers. Start address of this register is indicated by General Information.

Offset	Register Name	Type		
0	Power Management Revision	Read only	Register	Standard
1	Power Management Status Register	Read only	Register	Standard
2	Power Management Setting Register	Read/Write	Register	Standard

Table 5-28 : Power Management Register Set

Figure 5-16 shows Power Management Revision register. Lower 4-bit is assigned to revision of Power Management Register Set. The first version is indicated to 0000b. Power Management Functions may be updated with keeping former versions compatibility. Revision will be incremented for every update. A driver for controlling Power Management register compares card register revision and own driver revision to determine which functions can be used. This register is read only type and is not affected to a write operation.

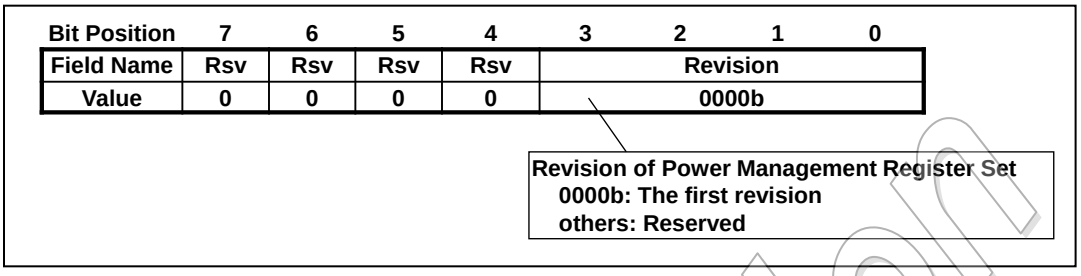


Figure 5-16 : Power Management Revision Register

Figure 5-17 shows Power Management Status register. Three supports bits of Power Manage Functions are defined in the upper 4-bit field. Setting a support bit to 1 means a function is supported by card. Card supporting Power Management Function shall support at least Power Off Notification. Support of Power Sustenance and Power Down Mode is optional.
Three status bits of Power Manage Functions are defined in the lower 4-bit field. There statuses will be changed in response to the setting of Power Management Setting register.

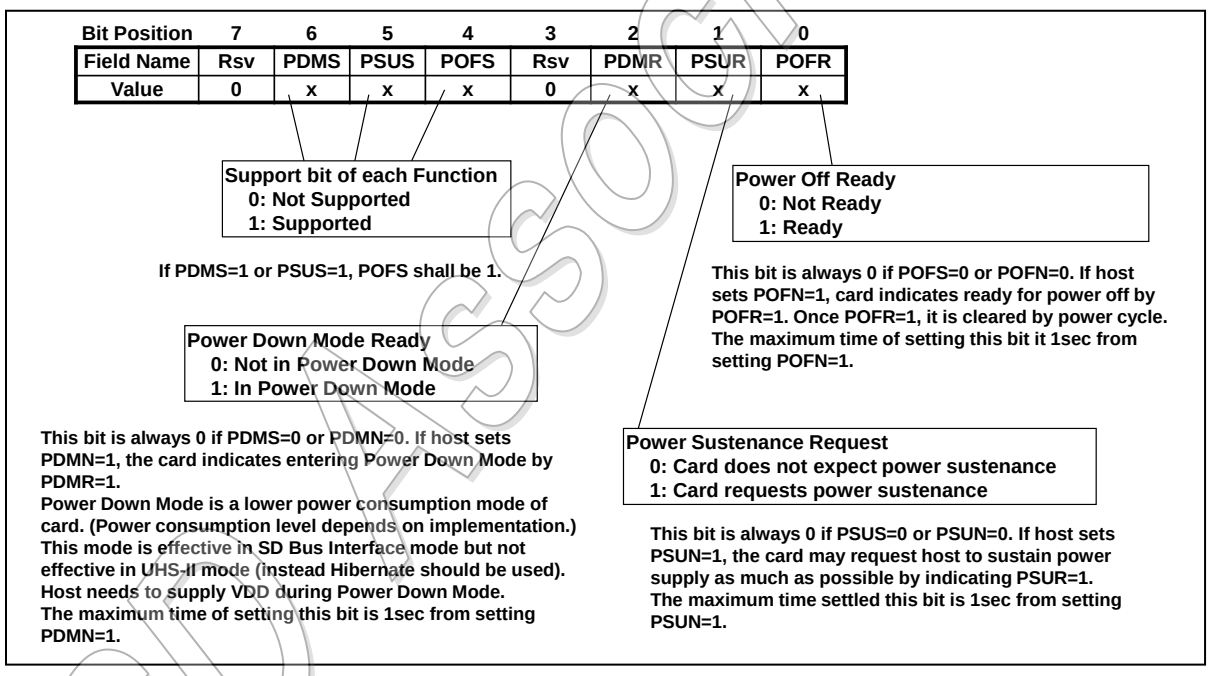


Figure 5-17 : Power Management Status Register

Figure 5-18 shows Power Management Setting register. Three notification bits of Power Manage Functions are defined in the lower 4-bit field. Host may set each notification bit if correspondent support bit is set to 1. Card indicates response of notification to the Power Management Status register. Host requests card to shut down card power by setting POFN=1 and to enter Power Down Mode by setting PDMN=1. Power Sustenance allows card to consume power to increase memory access performance while host does not use the card. Then on accepting Power Sustenance by setting PSUN=1, host should maintain card power as much as possible until host system power down. This register is read only type

and is not affected to a write operation. This register is read/write type.

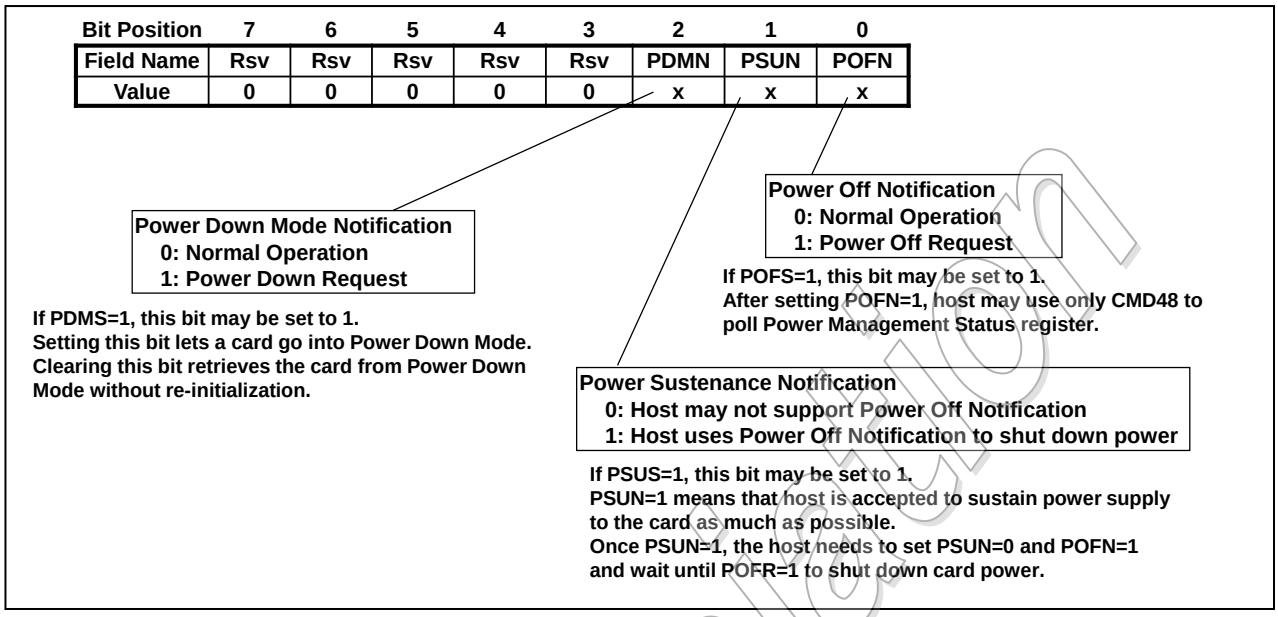


Figure 5-18 : Power Management Setting Register

5.8.1.3 Power Off Notification

Figure 5-19 shows flow chart to control Power Off Notification Sequence. Card should set POFR to 1 within 1 second of CMD49 busy period. If POFR cannot be set to 1 within busy period, card shall set POFR to 1 within 1 second from end of busy of CMD49.

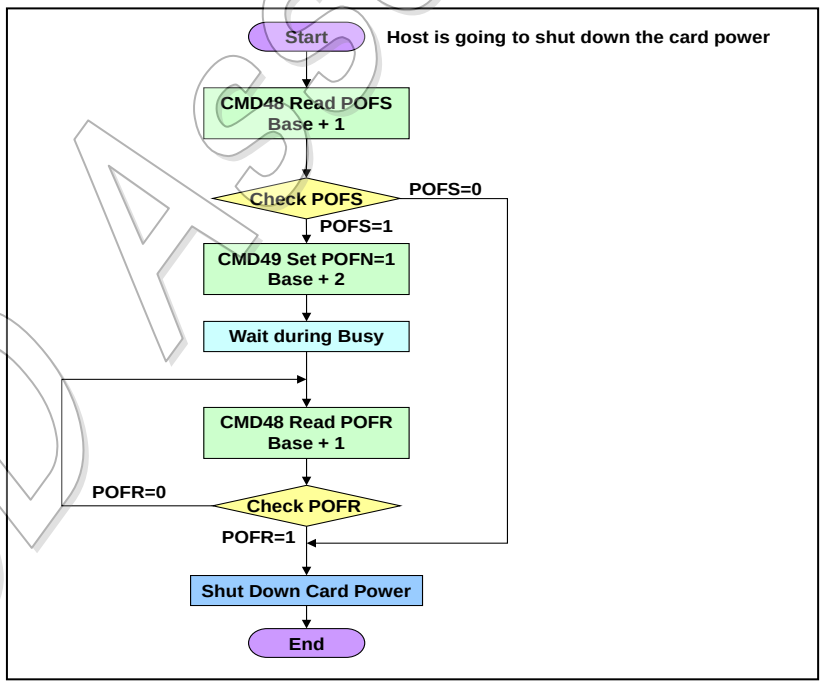


Figure 5-19 : Power Off Notification Flow

- **Power Off Notification Sequence**
  (1) Host checks support bit of Power Off Notification. If POFS=1, host may use Power Off Notification.
  (2) Just before host would like to shut down card power, host requests Power Off Notification to the card by setting POFN=1.
  (3) Host waits until busy of CMD49 is released.
  (4) Host polls Power Management Status register until the card indicates ready to power off (POFR=1).
  (5) If POFR=1, host shuts down the card power

  Once POFN is set to 1, power cycle and re-initialization are required to use the card again.

5.8.1.4 Power Sustenance

Figure 5-20 shows flow chart to control Power Sustenance. If host set PSUN to 1, card should set PSUR to 1 within 1 second of CMD49 busy period. If PSUR cannot be set to 1 within busy period, card shall set PSUR to 1 within 1 second from end of busy of CMD49. If host set PSUN to 0, card should set PSUR to 0 within 1 second of CMD49 busy period. If PSUR cannot be set to 0 within busy period, card shall set PSUR to 0 within 1 second from end of busy of CMD49.

- **Enabling Power Sustenance**
  (1) Host checks support bit of Power Sustenance. If PSUS=1, host may use Power Sustenance.
  (2) Host sets PSUN to 1.
  (3) Host waits until busy of CMD49 is released
  (4) Host polls Power Management Status register until the card indicates PSUR=1.

  Power Off Notification shall be used to shut down the card power while Power Sustenance is enabled.

- **Disabling Power Sustenance**
  (1) Host sets PSUN=0 in the Power Management Setting Register if PSUR=1.
  (2) Host waits until busy of CMD49 is released
  (3) Host polls Power Management Status register until the card indicates PSUR=0.

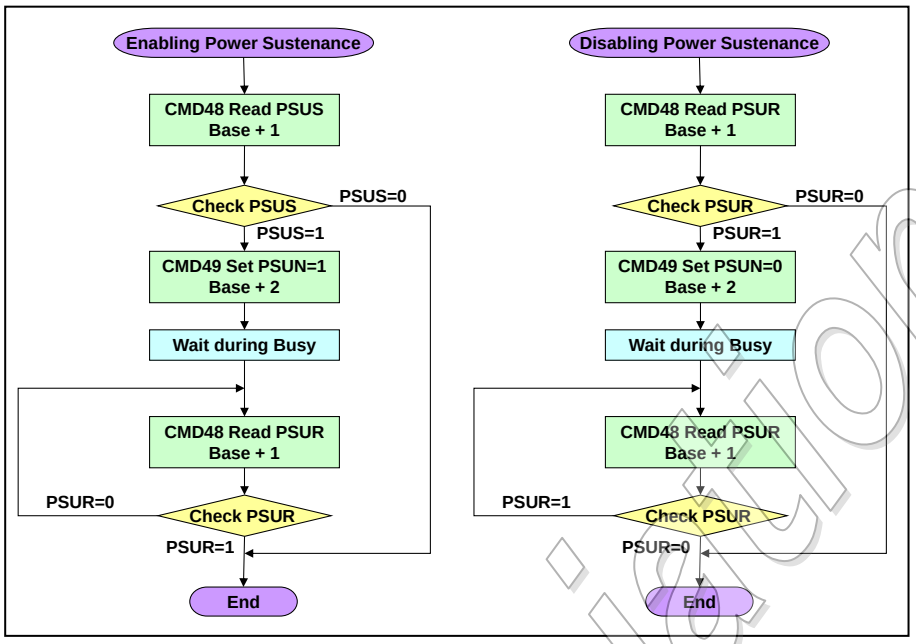


Figure 5-20 : Power Sustenance Enabling / Disabling Sequence Flow

Application Notes for Host designers

1) Once enabling Power Sustenance, Host shall use Power Off Notification sequence to shut down card power. Card power supply control by monitoring open/close the slot cover is one of possible safe solutions while in Power Sustenance to avoid sudden card removal.
2) In addition, card-internal program or erase procedures may continue during Power Sustenance mode even if card is locked by CMD42. If Host is to shut down card power in this situation, Host shall unlock the card at first and execute Power Off Notification sequence as mentioned above.
3) While Power Sustenance is enabled, card may consume power for internal operations, even without Host. So Host who would like to save power while card is unused, should disable Power Sustenance mode.
4) During Power Sustenance Mode, card may access memory internally.
5) Assume power consumption during Power Sustenance mode is equivalent to that of normal command operation. Setting of Power Limit through CMD6, is effective even in Power Sustenance mode.
6) By enabling Power Sustenance mode, total performance of Card may be noticeably improved. The actual level of improvement is not specified by the physical spec, and depends on the card's implementation.

5.8.1.5 Power Down Mode

This function can be used in SD mode only. In UHS-II mode, Hibernate mode should be used. Always PDMR=0 in UHS-II mode. In Power Down Mode, card executes at least CMD48/49 for exiting from Power Down Mode and CMD0. If host set PDMN to 1, card should set PDMR to 1 within 1 second of CMD49 busy period. If PDMR cannot be set to 1 within busy period, card shall set PDMR to 1 within 1 second from end of busy of CMD49. If host set PDMN to 0, card should set PDMR to 0 within 1 second of CMD49 busy period. If PDMR cannot be set to 0 within busy period, card shall set PDMR to 0 within 1 second from end of busy of CMD49. The card states retrieved from Power Down Mode is same as that before

entering Power Down Mode.

- **Entering Power Down Mode**
  (1) Host checks support bit of Power Down Mode. If PDMS=1, host may use Power Down Mode.
  (2) Host sets PDMN to 1.
  (3) Host waits until busy of CMD49 is released.
  (4) Host polls Power Management Status register until the card indicates PDMR=1.
  (5) Host keeps providing VDD. SD Clock should be stopped.

- **Exiting Power Down Mode**
  (1) Host starts to provide SD Clock.
  (2) Host sets PDMN=0 in the Power Management Setting Register if PDMR=1.
  (3) Host waits until busy of CMD49 is released.
  (4) Host polls Power Management Status register until the card indicates PDMR=0.

- **Aborting Power Down Mode**
  (1) If CMD0 is received, the card exits Power Down Mode and enters idle state.
  (2) Initialization is required to use the card.

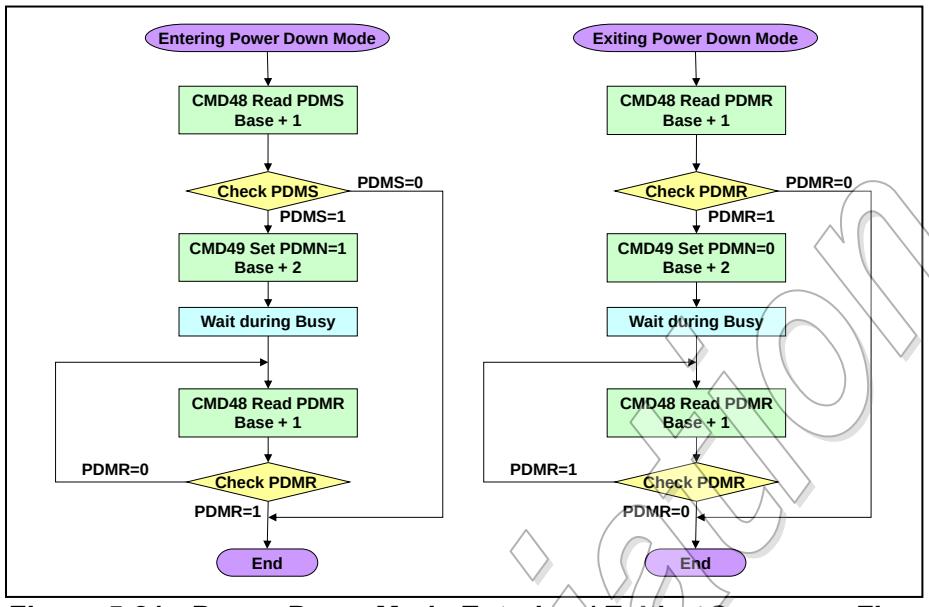


Figure 5-21 : Power Down Mode Entering / Exiting Sequence Flow

Application Notes for Host designers
1) The power consumption level during Power Down Mode is not specified and it varies dependent on Card products.
2) Host designers should consider that recovery time from Power Down Mode is up to 1 second.

5.8.1.6 General Information of Power Management Function

Table 5-29 shows an example of General Information for only Power Management Function is assigned to Extension 1 ("Number of Extended Functions" is set to 1 and "Pointer to Next Extension" is set to 0000h). Standard Code 0001h for Power Management Function (Defined by Table 5-27) is set to SFC. FCC is not used (set to 0000h). FMC, FMN and PFC shall be set to 0 because this function is manufacturer unrelated. FN is set to "PMF" (ASCII code) and filled with 0 for the rest of this field.

000h	Structure Revision	2 bytes	0000h	The first revision of General Information
002h	General Information Length	2 bytes	0040h	Data Length of this information
004h	Number of Extensions	1 byte	01h	Only Power Management Function is implemented
005h-0Fh	Reserved	11 byte	all 0	Set to all 0 (16 bytes header area)
010h	Extension 1 Standard Function Code (SFC)	2 bytes	0001h	Code of Power Management Function
012h	Extension 1 Function Capability Code (FCC)	2 bytes	0000h	FCC is not used
014h	Extension 1 Function Manufacturer Code (FMC)	2 bytes	0000h	Manufacturer unrelated
016h	Extension 1 Function Manufacturer Name (FMN)	16 bytes	all 0	Manufacturer unrelated
026h	Extension 1 Particular Function Code (PFC)	2 bytes	0000h	Manufacturer unrelated
028h	Extension 1 Function Name (FN)	16 bytes	"PMF"	ASCII + all 0 "PMF" = 00000000_00000000_00000000_00464D50h
038h	Pointer to Next Extension	2 bytes	0000h	End of General Information
03Ah	Number of Register Sets (=X)	1 byte	01h	One Register Set
03Bh	Reserved	1 byte	00h	
03Ch	Register Set Address 1	4 bytes	0004_0000h	Top Address of FNO=1
040h-1FFh	Unused Area		all 0	

Table 5-29 : General Information of Power Management Function

5.8.2 Performance Enhancement Function

5.8.2.1 Abstract of Performance Enhancement Function

This section defines the performance enhancement functions on extension register. This register will be utilized by host to understand the feature supported, enable / disable the features, etc.

5.8.2.2 Extension Register Set for Performance Enhancement Function

Table 5-30 shows the register set for performance enhancement function. The register set is divided into two sections. First 256 Bytes contain the support information of features and next 256 Bytes contain the registers to enable/disable feature and indicate the status of the feature.

Byte Offset	Name	R/W	Description
0	Performance Enhancement Function Revision	R	00h: Revision 1
1	Bit[0]: FX_EVENT Support Bit[7:1]: Reserved	R	0: Not supported, 1: Supported
2	Bit[0]: Card Initiated Maintenance Support Bit[1]: Host Initiated Maintenance Support Bit[7:2]: Reserved	R	0: Not supported, 1: Supported 0: Not supported, 1: Supported
3	Bit[1:0]: Card Maintenance Urgency Bit[7:2]: Reserved	R	00b: None, 01b: Mild, 10b: Middle, 11b: Urgent
4	Bit[0]: Cache Support Bit[7:1]: Reserved	R	0: Not supported, 1: Supported
5	Reserved	Reserved	
6	CQ Support and Depth	R	00h: CQ is not supported 01h: CQ is supported with queue depth = 2 (ID 0,1) 02h: CQ is supported with queue depth = 3 (ID

<table>
<tr><th>Byte Offset</th><th>Name</th><th>R/W</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td>0,1,2)
.....
1Fh: CQ is supported with queue depth = 32 (ID 0~31)
20h - FFh: Reserved</td></tr>
<tr><td>7</td><td>Reserved</td><td>Reserved</td><td></td></tr>
<tr><td>8-15</td><td>Task Error Status

Byte15 bit[7:6] Error status for Queued Task ID 31
Byte15 bit[5:4] Error status for Queued Task ID 30
..
Byte8 bit[1:0] Error status for Queued Task ID 0</td><td>R</td><td><table><tr><th></th><th>Error Status</th></tr><tr><td>00</td><td>No Error</td></tr><tr><td>01</td><td>Error</td></tr><tr><td>10</td><td>Reserved</td></tr><tr><td>11</td><td>Reserved</td></tr></table></td></tr>
<tr><td>16-256</td><td>Reserved</td><td>Reserved</td><td></td></tr>
<tr><td>257</td><td>Bit[0]: FX_EVENT Enable
Bit[7:1]: Reserved</td><td>R/W</td><td>0: Disable(d), 1: Enable(d)</td></tr>
<tr><td rowspan="3">258</td><td>Bit[0]: Card Initiated Maintenance Enable</td><td rowspan="3">R/W</td><td>0: Disable(d), 1: Enable(d)</td></tr>
<tr><td>Bit[1]: Host Initiated Maintenance Enable</td><td>0: Disable(d), 1: Enable(d)</td></tr>
<tr><td>Bit[7:2]: Reserved</td><td></td></tr>
<tr><td rowspan="2">259</td><td>Bit[0]: Start Host-Initiated Maintenance</td><td rowspan="2">R/W</td><td>Setting bit[0]=1 starts host-initiated maintenance.
Setting bit[0]=0 by card indicates the break of maintenance.
Card shall break maintenance within 1 second.</td></tr>
<tr><td>Bit[7:1]: Reserved</td><td></td></tr>
<tr><td rowspan="2">260</td><td>Bit[0]: Cache Enable</td><td rowspan="2">R/W</td><td>0: Disable(d), 1: Enable(d)</td></tr>
<tr><td>Bit[7:1]: Reserved</td><td></td></tr>
<tr><td rowspan="2">261</td><td>Bit[0]: Flush Cache</td><td rowspan="2">R/W</td><td>Setting bit[0]=1 starts Cache flush.
Setting bit[0]=0 by card indicates the end of Cache flush.
Card shall complete Cache flush within 1 second.</td></tr>
<tr><td>Bit[7:1]: Reserved</td><td></td></tr>
<tr><td rowspan="3">262</td><td>Bit[0]: Enable CQ</td><td rowspan="3">R/W</td><td>0: Disable(d), 1: Enable(d)</td></tr>
<tr><td>Bit[1]: CQ Mode</td><td>0: Voluntary mode, 1: Sequential Mode</td></tr>
<tr><td>Bit[7:2]: Reserved</td><td></td></tr>
<tr><td>263-511</td><td>Reserved</td><td>Reserved</td><td></td></tr>
</table>

Table 5-30 : Performance Enhancement Register Set

5.8.2.3 General Information of Performance Enhancement Function

Table 5-31 shows an example of General Information for Performance Enhancement Function assigned to Extension 2 ("Number of Extended Functions" is set to 2 and "Pointer to Next Extension" is set to 0040h). Standard Code 0002h for Performance Enhancement Function (Defined by Table 5-30) is set to SFC. FCC is not used (set to 0000h). FMC, FMN and PFC shall be set to 0 because this function is manufacturer unrelated. FN is set to "PEF" (ASCII code) and filled with 0 for the rest of this field.

All performance enhancement shall be disabled after receiving ‘Power Off Notification’, including CQ,

Cache and Self Maintenance.

Address	Name	Size (bytes)	Value	Comments
000h	Structure revision	2	0000h	The first revision of General Information
002h	General Information Length	2	**0070h**	Data Length of this information
004h	Number of extensions	1	02h	Both PMF and PEF are implemented
005h-0Fh	reserved	11	All 0	Pad with 0s (16 bytes header area)
010h-37h	PMF Header	40		Used by Power Management Function
038h	PMF Pointer to Next Extension	2	**0040h**	Pointer to Performance Enhancement Function
03Ah-03Fh	PMF Register sets	6		Used by Power Management Function
040h	Extention2 Standard Function Code (SFC)	2	0002h	Code of Performance Enhancement Function
042h	Extention2 Function Capability Code (FCC)	2	0000h	FCC not used
044h	Extention2 Function Manufacture Code (FMC)	2	0000h	Manufacturer Unrelated
046h	Extention2 Function Manufacture Name (FMN)	16	All 0	Manufacturer Unrelated
056h	Extention2 Particular Function Code (PFC)	2	0000h	Manufacturer Unrelated
058h	Extention2 Function Name (FN)	16	"PEF"	ASCII + all 0 "PEF"=00000000_00000000_00000000_00464550h
068h	Pointer to Next Extension	2	0000h	End of General Information
06Ah	Number of Register Sets	1	01h	Single Register set of 512 bytes
06Bh	Reserved	1	00h	
06Ch	Register Set Address	4	0008_0000h	FNO=2, Start Address = 0h
070h-1FFh	Unused Area		All 0	

Table 5-31 : General Information of Performance Enhancement Function

5.8.3 Security and Boot Function

5.8.3.1 Abstract of Security and Boot Function

This section defines the security (RPMB) and boot functions on extension register. This register will be utilized by host to understand the feature supported, enable / disable the features, etc.

5.8.3.2 Extension Register Set for Security and Boot Function

Table 5-32 shows the register set for security and boot function.

Byte Offset	Name	R/W	Description
0	Security and Boot Function Revision	R	00h: Revision 1
1-7	Reserved	Reserved	
8	Bit[2:0]: Number of RPMB Units (1)	R	000b: RPMB is not supported 001b: Card has one RPMB Unit Others: Reserved
	Bit[5:3]: RPMB Authentication Method		000b: HMAC SHA-256 (refer to RFC 6234) Others: Reserved
	Bit[7:6]: Reserved		
9	Reserved	Reserved	

Byte Offset	Name	R/W	Description
10	RPMB Total Size	R	One RPMB unit size indicated in 128KB units and 0's base (00h means 128KB, FFh means 32,768KB). If Number of RPMB Units = 000b, this field is ignored.
11	RPMB Access Size	R	Max number of data that may be read or written per RPMB access by ACMD53 or 54, in 512B units and 0's base (00h means 512B, FFh means 131,072B). If Number of RPMB Units = 000b, this field is ignored.
12-15	Reserved	Reserved	
16	Bit[0]: Boot Partition Protection Enable	R	Indicating value of Bit 0 of Boot partition Protection Enable in RPMB Device Configuration Block Data Structure (see Table 4-83).
	Bit[7:1]: Reserved		
17	Bit[0]: Partition Lock for Boot Partition 0 (2)	R	Indicating value of Bit 0 of Boot Partition Lock in RPMB Device Configuration Block Data Structure (see Table 4-83).
	Bit[1] Partition Lock for Boot Partition 1 (2)		Indicating value of Bit 1 of Boot Partition Lock in RPMB Device Configuration Block Data Structure (see Table 4-83).
	Bit[7:2]: Reserved		
18	Bit[0]: WP_UPC Authentication Control	R	Indicating value of Bit 0 of User Area Write Protection Authentication Control in RPMB Device Configuration Block Data Structure (see Table 4-83).
	Bit[1] PWP Authentication Control		Indicating value of Bit 1 of User Area Write Protection Authentication Control in RPMB Device Configuration Block Data Structure (see Table 4-83).
	Bit[7:2]: Reserved		
19-23	Reserved	Reserved	
24	Boot Partition Size (3), (4)	R	Boot partition size indicated in 128KB units and 0's base (00h means 128KB, FFh means 32,768KB).
25-255	Reserved	Reserved	
256	Bit[0] Active Boot Partition (4), (5)	R/W	0b: Boot Partition 0 is active 1b: Boot Partition 1 is active.
	Bit[7:1]: Reserved		
257-259	Reserved	Reserved	
260	Bit[3:0]: Preprogrammed bus mode for Fast Boot (4), (5)	R/W	1h: DS mode, 2h: HS mode 9h: SDR12 mode, Ah: SDR25 mode, Bh: SDR50 mode, Ch: SDR104 mode, Dh: DDR50 mode Others: Reserved
	Bit[7:4]: Reserved		
261-511	Reserved	Reserved	

Note (1) The number of RPMB Unit is up to 1 in SD Specification, including SD Express.
(2) RPMB authentication is required for unlocking Boot Partitions.
(3) Size of Boot Partition 0 and 1 shall be the same.
(4) If BOOT_PARTITION_SUPPORT in SD Status is 0b (not supported), this field is ignored.
(5) When boot partition protection is enabled and Host tries to update this field by CMD49 or 59, RPMB authentication is required beforehand.

Table 5-32 : Security and Boot Register Set

5.8.3.3 General Information of Security and Boot Function

Table 5-33 shows an example of General Information for Security and Boot Function assigned to Extension 3. Standard Code 0003h for Security and Boot Function (defined by Table 5-27) is set to SFC. FCC is not used (set to 0000h). FMC, FMN and PFC shall be set to 0 because this function is manufacturer unrelated. FN is set to "SBF" (ASCII code) and filled with 0 for the rest of this field.

000h	Structure Revision	2 bytes	0000h	The first revision of General Information
002h	General Information Length	2 bytes	00A0h	Data Length of this information
004h	Number of Extensions	1 byte	03h	PMF, PEF and SBF are implemented (example)
005h-0Fh	Reserved	11 byte	all 0	Set to all 0 (16 bytes header area)
010h-037h	PMF Header	40 bytes		Used by Power Management Function
038h-039h	PMF Pointer to Next Extension	2 bytes	0040h	Pointer to Performance Enhancement Function
03Ah-03Fh	PMF Register Sets	6 bytes		
040h-067h	PEF Header	40 bytes		Used by Performance Enhancement Function
068h-069h	PEF Pointer to Next Extension	2 bytes	0070h	Pointer to Security and Boot Function
06Ah-06Fh	PEF Register Sets	6 bytes		
070h	Extension 3 Standard Function Code (SFC)	2 bytes	0003h	Code of Security and Boot Function
072h	Extension 3 Function Capability Code (FCC)	2 bytes	0000h	FCC is not used
074h	Extension 3 Function Manufacturer Code (FMC)	2 bytes	0000h	Manufacturer unrelated
076h	Extension 3 Function Manufacturer Name (FMN)	16 bytes	all 0	Manufacturer unrelated
086h	Extension 3 Particular Function Code (PFC)	2 bytes	0000h	Manufacturer unrelated
088h	Extension 3 Function Name (FN)	16 bytes	“SBF"	ASCII + all 0 “SBF" = 00000000_00000000_00000000_00464253h
098h	Pointer to Next Extension	2 bytes	0000h	End of General Information
09Ah	Number of Register Sets	1 byte	01h	One Register Set
09Bh	Reserved	1 byte	00h	
09Ch	Register Set Address	4 bytes	000C_0000h	FNO=3, Start Address = 0h
0A0h-1FFh	Unused Area		all 0	

Table 5-33 : General Information of Security and Boot Function

6. SD Memory Card Hardware Interface

This section is a blank in the Simplified Specification.

6.1 Hot Insertion and Removal

This section is a blank in the Simplified Specification.

6.2 Card Detection (Insertion/Removal)

This section is a blank in the Simplified Specification.

6.3 Power Protection (Insertion/Removal)

This section is a blank in the Simplified Specification.

6.4 Power Scheme

The power scheme of the SD Memory Card bus is handled locally in each SD Memory Card and in the host.

6.4.1 Power Up Sequence for SD Bus Interface

6.4.1.1 Power Up Time of Card

A device shall be ready to accept the first command within 1ms from detecting VDD min.
Device may use up to 74 clocks for preparation before receiving the first command.

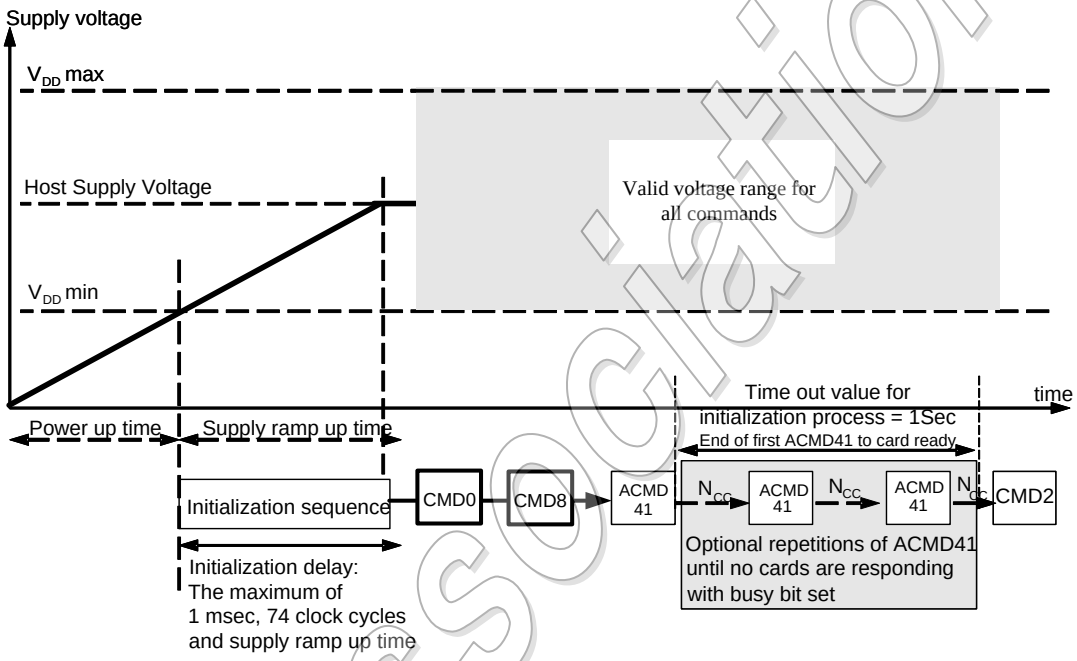


Figure 6-4 : Power-up Diagram of Card

'Power up time' is defined as voltage rising time from 0 volt to V_{DD} min (refer to Section 6.6) and depends on application parameters such as the maximum number of SD Cards, the bus length and the characteristic of the power supply unit.
'Supply ramp up time' provides the time that the power is built up to the operating level (Host Supply Voltage) and the time to wait until the SD card can accept the first command,
The host shall supply power to the card so that the voltage is reached to Vdd_min within 250ms and start to supply at least 74 SD clocks to the SD card with keeping CMD line to high. In case of SPI mode, CS shall be held to high during 74 clock cycles.
After power up (including hot insertion, i.e. inserting a card when the bus is operating) the SD Card enters the *idle state*. In case of SD host, CMD0 is not necessary. In case of SPI host, CMD0 shall be the first command to send the card to SPI mode.
CMD8 is newly added in the Physical Layer Specification Version 2.00 to support multiple voltage ranges and used to check whether the card supports supplied voltage. The version 2.00 or later host shall issue CMD8 and verify voltage before card initialization. The host that does not support CMD8 shall supply high voltage range.
ACMD41 is a synchronization command used to negotiate the operation voltage range and to poll the cards until they are out of their power-up sequence. In case the host system connects multiple cards, the host shall check that all cards satisfy the supplied voltage. Otherwise, the host should select one of the cards and initialize.

6.4.1.2 Power Up Time of Host

Reset level is not described in Figure 6-4 of the Physical Layer Specification Version 2.00. Change of Figure 6-5 is applied to Figure 6-4 of the Physical Layer Specification.
Host needs to keep power line level less than 0.5V and more than 1ms before power ramp up.

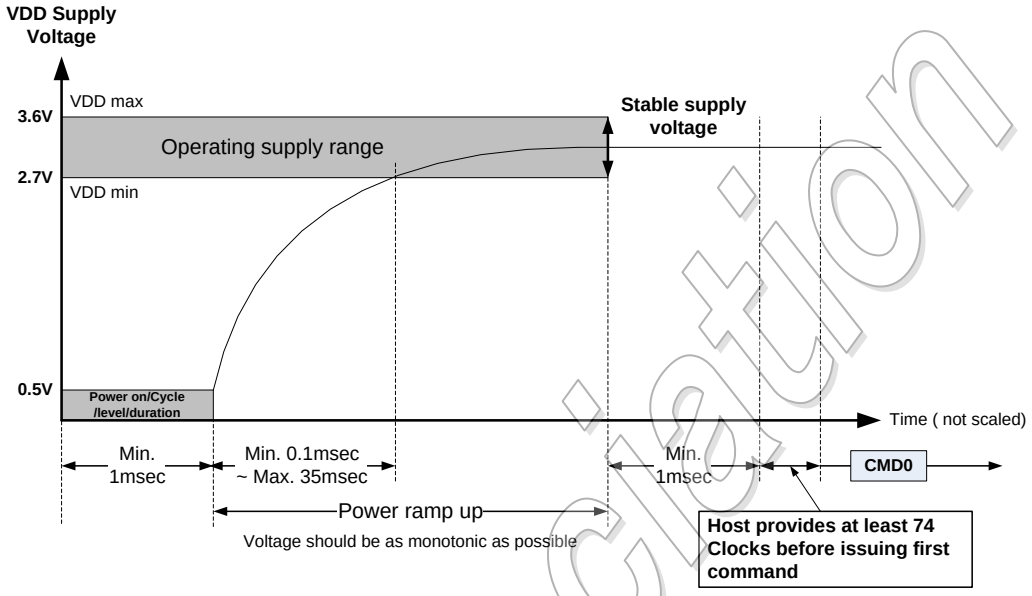


Figure 6-5 : Power Up Diagram of Host

6.4.1.3 Power On or Power Cycle

Followings are requirements for Power on and Power cycle to assure a reliable SD Card hard reset.

(1) Voltage level shall be below 0.5V
(2) Duration shall be at least 1ms.

6.4.1.4 Power Supply Ramp Up

The power ramp up time is defined from 0.5V threshold level up to the operating supply voltage which is stable between VDD(min.) and VDD(max.) and host can supply SDCLK.

Followings are recommendation of Power ramp up:

(1) Voltage of power ramp up should be monotonic as much as possible.
(2) The minimum ramp up time should be 0.1ms.
(3) The maximum ramp up time should be 35ms for 2.7-3.6V power supply.
(4) Host shall wait until VDD is stable.
(5) After 1ms VDD stable time, host provides at least 74 clocks before issuing the first command.

6.4.1.5 Power Down and Power Cycle

When the host shuts down the power, the card V_{DD} shall be lowered to less than 0.5Volt for a minimum period of 1ms. During power down, DAT, CMD, and CLK should be disconnected or driven to logical 0 by the host to avoid a situation that the operating current is drawn through the signal lines.

If the host needs to change the operating voltage, a power cycle is required. Power cycle means the power is turned off and supplied again. Power cycle is also needed for accessing cards that are already in *Inactive State.* To create a power cycle the host shall follow the power down description before power up the card (i.e. the card V_{DD} shall be once lowered to less than 0.5Volt for a minimum period of 1ms).

6.4.2 Power Up Sequence for UHS-II Interface

6.4.2.1 Power Up Sequence of UHS-II Card

Figure 6-6 shows power up sequence for UHS-II Card. Either power up order of V_{DD1} and V_{DD2} should be expected and card power up is dependent on later one. UHS-II Card shall be ready to start PHY Initialization within 1ms from detecting later of V_{DD1} min or V_{DD2} min.

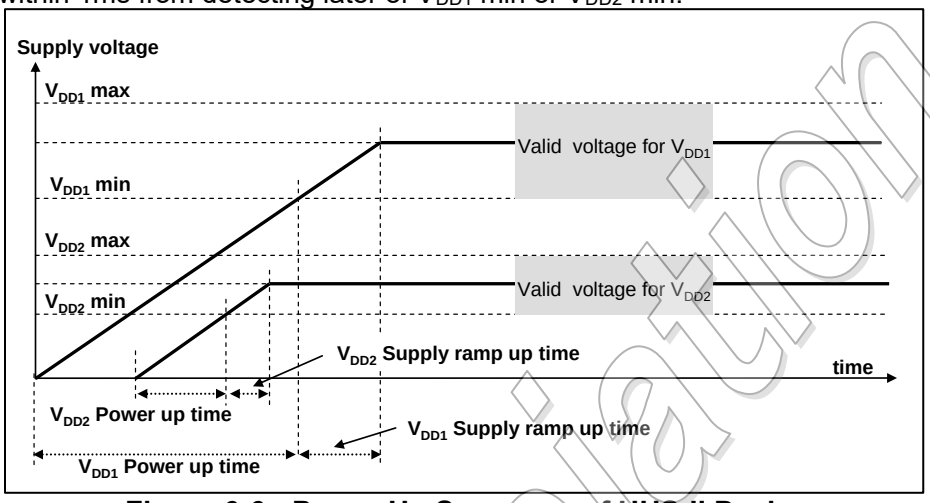


Figure 6-6 : Power Up Sequence of UHS-II Device

6.4.2.2 Power Up Sequence of UHS-II Host

Figure 6-7 shows power up sequence for UHS-II Host. Followings are host requirements.

- Power up and ramp up of V_{DD1} and V_{DD2} should be monotonic.
- Either power up order is allowed for V_{DD1} and V_{DD2}.
- Trise shall be 0.1-35ms.
- Host shall wait until both V_{DD1} and V_{DD2} are stable.
- After additional 1ms stable time from both V_{DD1} and V_{DD2} are stable, host starts to provide RCLK and then starts PHY Initialization.
- Once V_{DD2} is supplied, host needs to supply V_{DD2} until power cycle.
- When power cycle is executed, keep V_{DD1} less than 0.5V and V_{DD2} less than 0.2V at least 1ms before starting power up.

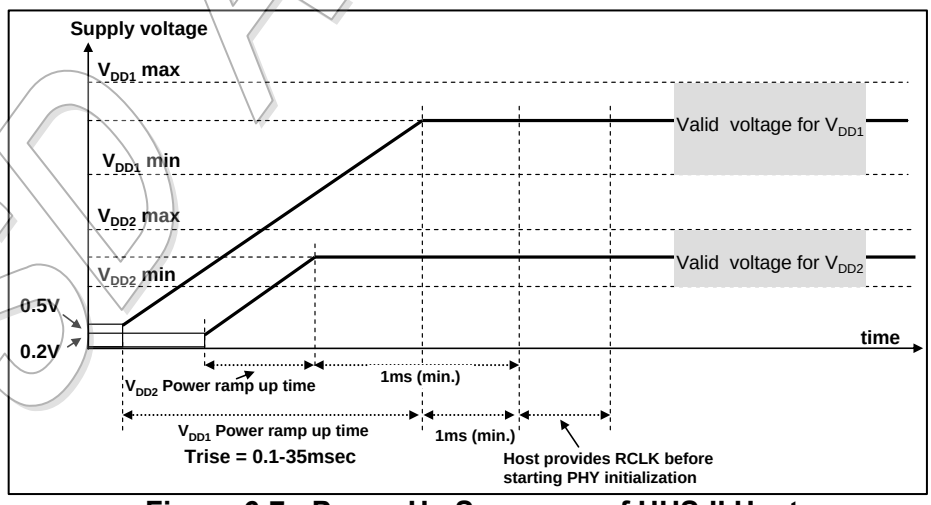


Figure 6-7 : Power Up Sequence of UHS-II Host

6.4.3 Power Up Sequence for PCIe Interface

6.4.3.1 Power Up Sequence of SD Express Card

SD Express card shall support VDD2 (1.8V).
In addition, card and host may support VDD3 (1.2V). VDD3 if supported will be used instead of VDD2 (Supply of VDD2 and VDD3 at the same time is not supported by the card). Following section provides description of VDD3 however all the information is the same for either VDD2 or VDD3.

Figure 6.4.3-1 shows one possible power up sequence for SD Express Card. VDD1 shall be supplied first, followed by VDD2 or VDD3 in case that SD Express card presence was detected.

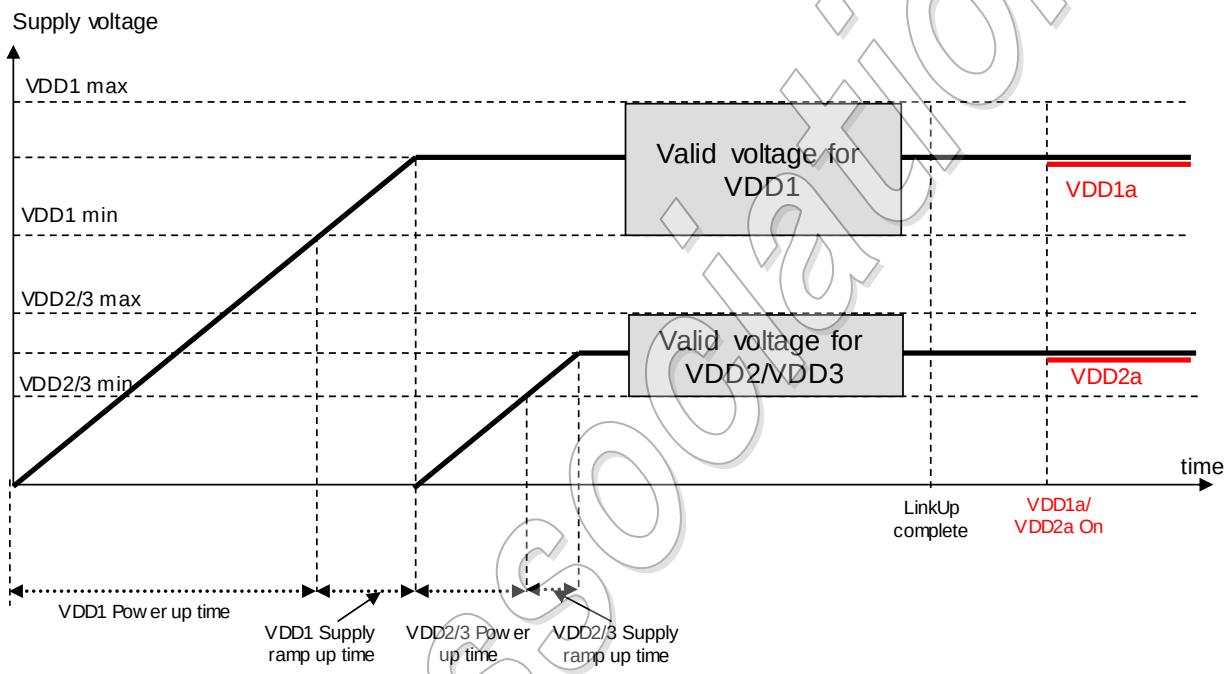


Figure 6.4.3-1: Power Up Sequence of SD Express Card

VDD1a is expected to be active when PCIe bus mode is either Gen3x2, Gen4x1, or Gen4x2. Moreover, VDD2a is expected to be active when PCIe bus mode is Gen3x2, or Gen4x2.
VDD1a and VDD2a can be supplied after PCIe LinkUp at any time before switching to power mode with max power consumption above 1.8W. Either power up order of VDD1a and VDD2a should be expected.

Figure 6.4.3-2 shows another power up sequence for SD Express Card. VDD1 shall be supplied first, followed by VDD2 or VDD3 in case that SD Express card presence was detected.

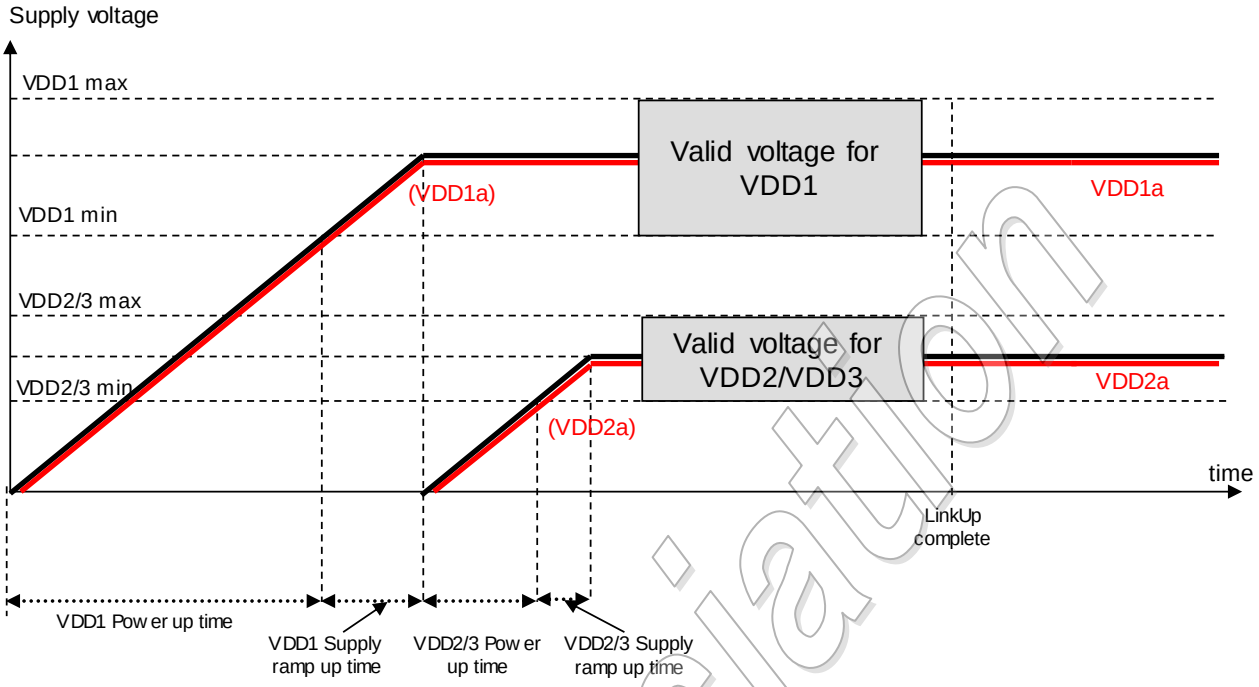


Figure 6.4.3-2 : Power Up Sequence of SD Express Card (Another Case)

VDD1 and VDD1a may be supplied at the same time if host detects the existence of VDD1a pad. Refer to Appendix J for VDD1a pad detection.
VDD2 and VDD2a may be supplied simultaneously.

6.4.3.2 Power Up Sequence of SD Express Host

Figure 6.4.3-3 and Figure 6.4.3-4 show different example if power up sequence allowed for SD Express Host. Followings are host requirements.

- Power up and ramp up of VDD1 and VDD2/VDD3 should be monotonic.
- VDD1 shall be supplied first, followed by VDD2/VDD3 in case that SD Express card presence was detected.
- Host is recommended to turn VDD3 on only if SD Express type card presence is detected.
- VDD1 and VDD2 may be turned on regardless of SD Express card presence detection. Host supporting multiple rows connector shall ensure VDD1a and VDD2a are not supplied until the card is fully inserted.
- Trise shall be 0.1-35ms.
- Host shall wait until both VDD1 and VDD2/VDD3 are stable before initiating the PCIe training process.
- After completing PCIe LinkUp, host may supply VDD1a and VDD2a any time before switching to power mode with max power consumption above 1.8W. VDD1a shall be provided before setting PCIe bus mode to either Gen3x2, Gen4x1, or Gen4x2. Moreover, VDD2a shall be provided before setting PCIe bus mode to Gen3x2, or Gen4x2. Power up order of VDD1a and VDD2a is arbitrary.
- Refer to Section 3.17.2 for further details on the PCIe training during the initialization process.
- When power cycle is executed, keep VDD1/VDD1a below 0.5V and VDD2/VDD2a/VDD3 less than 0.2V for at least 1ms before starting power up.
- It is recommended to turn off powers upon card removal detection.

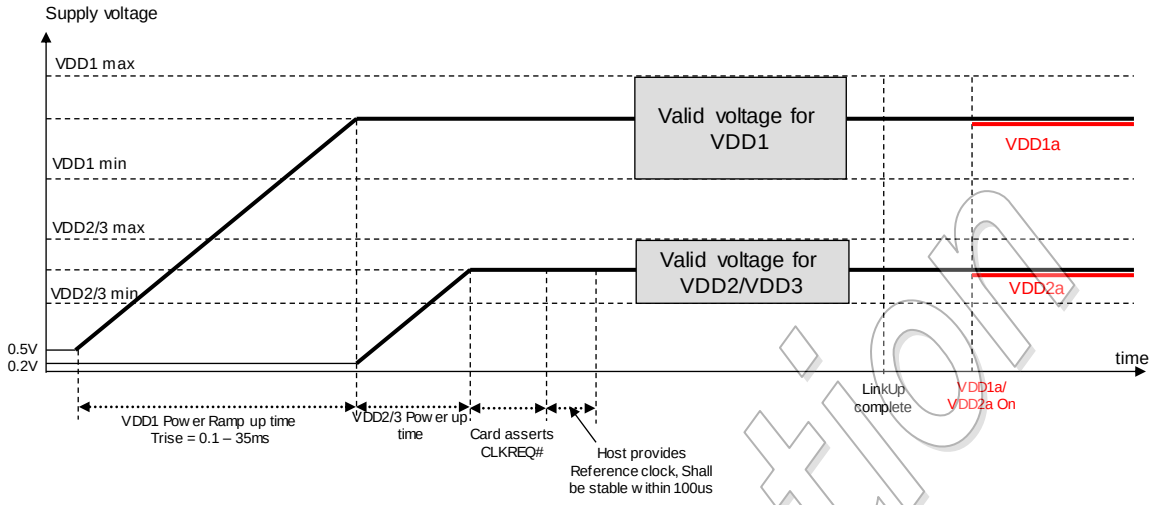


Figure 6.4.3-3 : Power Up Sequence of SD Express Host

Figure 6.4.3-4 shows another option of power up sequence for SD Express Host. In this option, host supplies VDD1 and VDD1a at the same time if host detects the existence of VDD1a pad. Moreover, host may supply VDD2 and VDD2a simultaneously.
Refer to Appendix J for VDD1a pad detection.

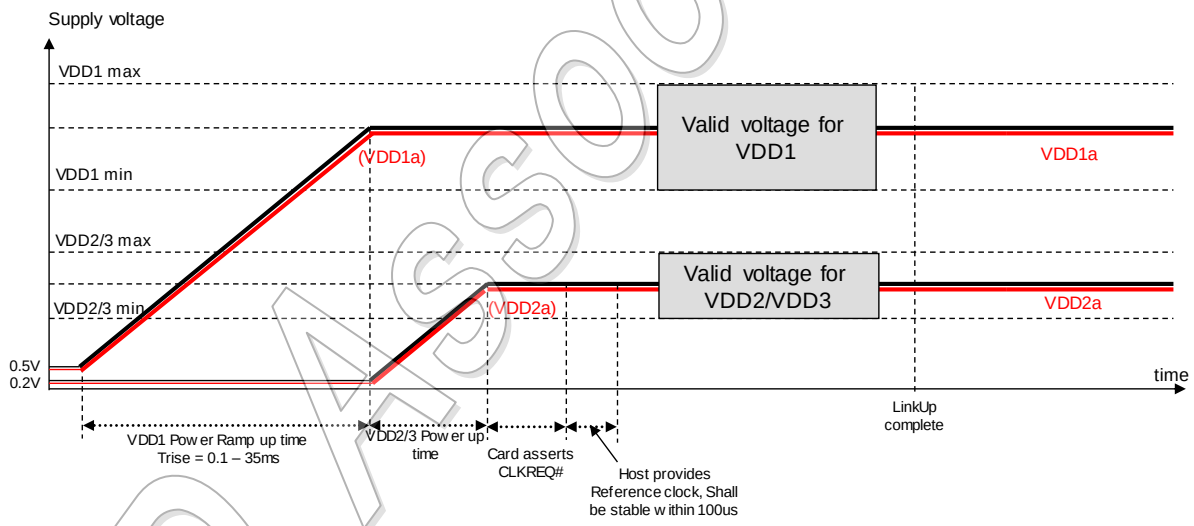


Figure 6.4.3-4 : Power Up Sequence of SD Express Host (Another Option)

6.5 Programmable Card Output Driver (3.3V Single End) (Optional)

This section is a blank in the Simplified Specification.

6.6 Bus Operating Conditions for 3.3V Signaling

6.6.1 Threshold Level for High Voltage Range

This section is a blank in the Simplified Specification.

6.6.2 Peak Voltage and Leakage Current

This section is a blank in the Simplified Specification.

6.6.3 Power Consumption

This section is a blank in the Simplified Specification.

6.6.4 Bus signal line load

This section is a blank in the Simplified Specification.

6.6.5 Bus Signal Levels

This section is a blank in the Simplified Specification.

6.6.6 Bus Timing (Default)

Description here is a blank in the Simplified Specification.

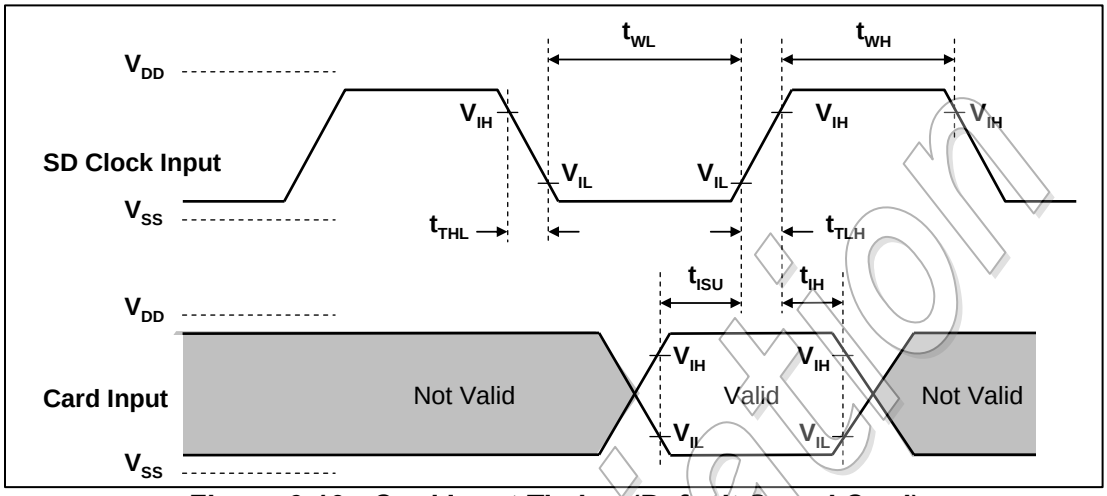


Figure 6-10 : Card Input Timing (Default Speed Card)

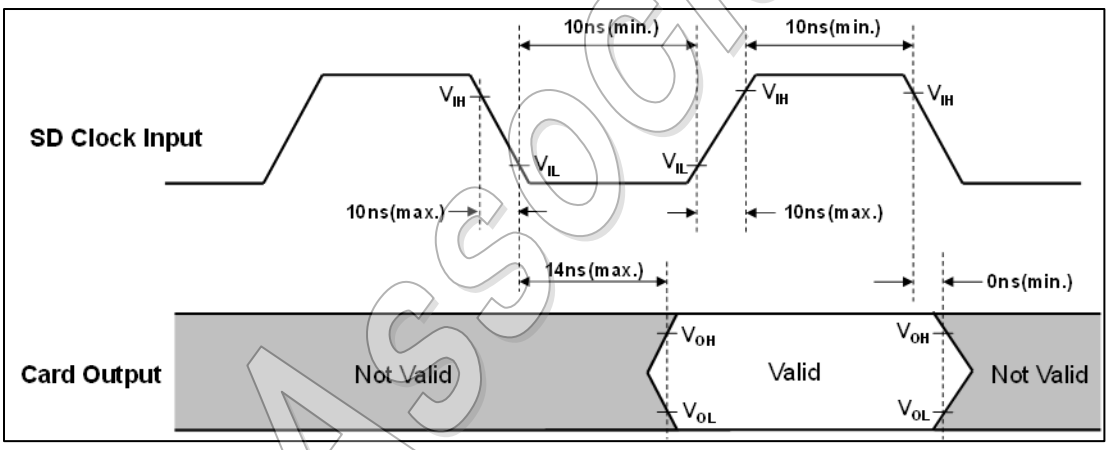


Figure 6-11 : Card Output Timing (Default Speed Mode)

6.6.7 Bus Timing (High-Speed Mode)

Description here is a blank in the Simplified Specification.

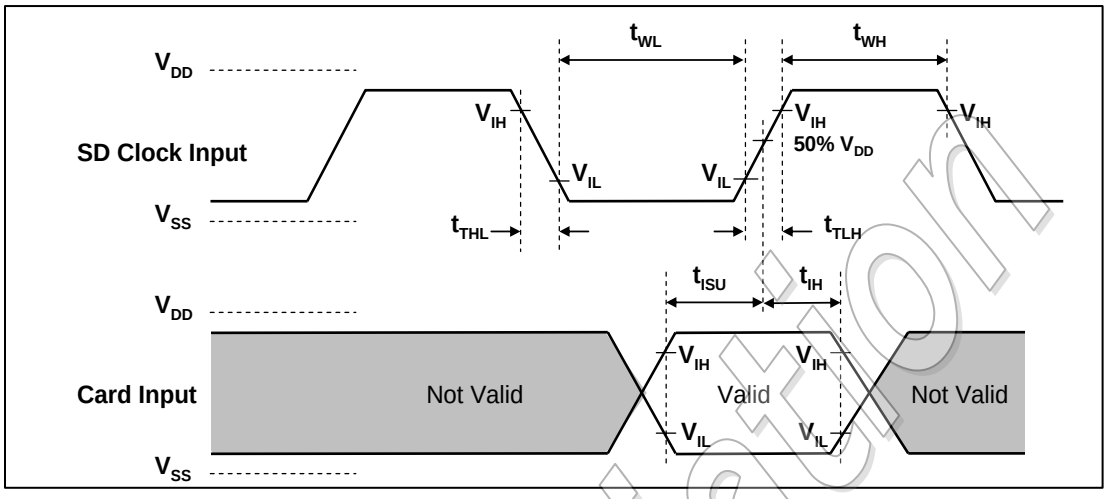


Figure 6-12 : Card Input Timing (High Speed Card)

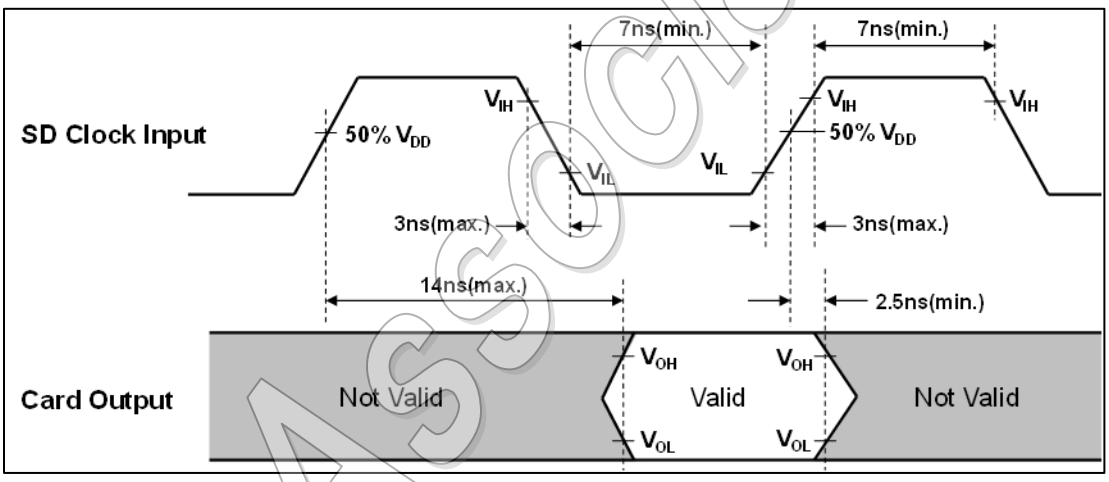


Figure 6-13 : Card Output Timing (High Speed Mode)

6.7 Driver Strength and Bus Timing for 1.8V Signaling

This section is a blank in the Simplified Specification.

6.8 Electrical Static Discharge (ESD) Requirement

This section is a blank in the Simplified Specification.

7. SPI Mode

7.1 Introduction

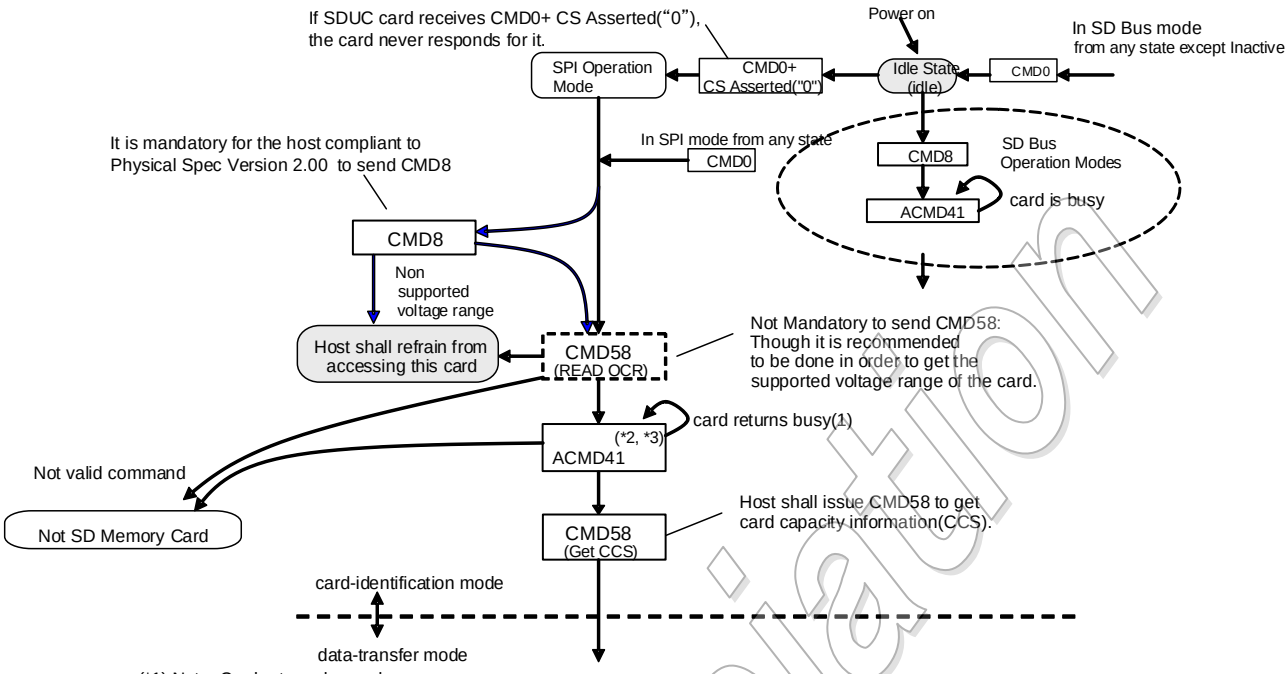
The SPI mode consists of a secondary communication protocol that is offered by Flash-based SD Memory Cards. This mode is a subset of the SD Memory Card protocol, designed to communicate with a SPI channel, found in some microcontrollers. The interface is selected during the first reset command after power up (CMD0) and cannot be changed once the part is powered on.
The SPI standard defines the physical link only, and not the complete data transfer protocol. The SD Memory Card SPI implementation uses a subset of the SD Memory Card protocol and command set. The advantage of the SPI mode is the capability of using an off-the-shelf host, hence reducing the design-in effort to minimum. The disadvantage is the loss of performance of the SPI mode versus SD mode (e.g. Single data line and hardware CS signal per card).

The commands and functions in SD mode defined after the Version 2.00 are not supported in SPI mode. The card may respond to the commands and functions even if the card is in SPI mode but host should not use them in SPI mode.

SPI Mode is not supported by SDUC cards.

7.2 SPI Bus Protocol

While the SD Memory Card channel is based on command and data bit streams that are initiated by a start bit and terminated by a stop bit, the SPI channel is byte oriented. Every command or data block is built of 8-bit bytes and is byte aligned with the CS signal (i.e. the length is a multiple of 8 clock cycles).
The card starts to count SPI bus clock cycle at the assertion of the CS signal. Every command or data token shall be aligned with 8-clock cycle boundary.
Similar to the SD Memory Card protocol, the SPI messages consist of command, response and data-block tokens. All communication between host and cards is controlled by the host (master). The host starts every bus transaction by asserting the CS signal low.
The selected card always responds to the command as opposed to the SD mode.
When the card encounters a data retrieval problem in a read operation, it will respond with an error response (which replaces the expected data block) rather than by a timeout as in the SD mode.
Additionally, every data block sent to the card during write operations will be responded with a data response token.
In the case of a Standard Capacity Memory Card, a data block can be as big as one card write block and as small as a single byte. Partial block read/write operations are enabled by card options specified in the CSD register.
In case of SDHC and SDXC Cards, block length is fixed to 512 bytes. The block length set by CMD16 is only used for CMD42 and not used for memory data transfer. So, partial block read/write operations are also disabled. Furthermore, Write Protected commands (CMD28, CMD29 and CMD30) are not supported.



(*1) Note: Card returns busy when
- Card executes internal initialization process.
- If the card is High capacity SD Memory Card, there are 2 cases
  1. CMD8 was not issued before ACMD41
  2. ACMD41 is issued with HCS=0

(*2) Note: 2.1mm SD Memory Card can be initialized using CMD1 and Thin (1.4mm) SD Memory Card can be initialized using CMD1 only after firstly initialized by using CMD0 and ACMD41. In any of the cases CMD1 is not recommended because it may be difficult for the host to distinguish between MultiMediaCard and SD Memory Card.
If the SD card is initialized by CMD1 and the host treat it as MMC card, not SD card, the Data of the card may be damaged because of wrong interpretation of CSD and CID registers.

(*3) Note: SDUC card, card can stay at busy status and does not reply ready to host during ACMD41 to let host know SDUC card cannot use SPI mode.

Figure 7-1 : SD Memory Card State Diagram (SPI mode)

7.2.1 Mode Selection and Initialization

The SD Card is powered up in the SD mode. It will enter SPI mode if the CS signal is asserted (negative) during the reception of the reset command (CMD0). If the card recognizes that the SD mode is required it will not respond to the command and remain in the SD mode. If SPI mode is required, the card will switch to SPI and respond with the SPI mode R1 response.

The only way to return to the SD mode is by entering the power cycle. In SPI mode, the SD Card protocol state machine in SD mode is not observed. All the SD Card commands supported in SPI mode are always available.

Figure 7-2 shows the initialization sequence of SPI mode.

SEND_IF_COND (CMD8) is used to verify SD Memory Card interface operating condition. The argument format of CMD8 is the same as defined in SD mode and the response format of CMD8 is defined in Section 7.3.2.6. The card checks the validity of operating condition by analyzing the argument of CMD8 and the host checks the validity by analyzing the response of CMD8. The supplied voltage is indicated by VHS field in the argument. The card assumes the voltage specified in VHS as the current supplied voltage. Only 1-bit of VHS shall be set to 1 at any given time. Check pattern is used for the host to check validity of communication between the host and the card.

If the card indicates an illegal command, the card is legacy and does not support CMD8. If the card supports CMD8 and can operate on the supplied voltage, the response echoes back the supply voltage and the check pattern that were set in the command argument.

If VCA in the response is set to 0, the card cannot operate on the supplied voltage. If check pattern is not

matched, CMD8 communication is not valid. In this case, it is recommended to retry CMD8 sequence.

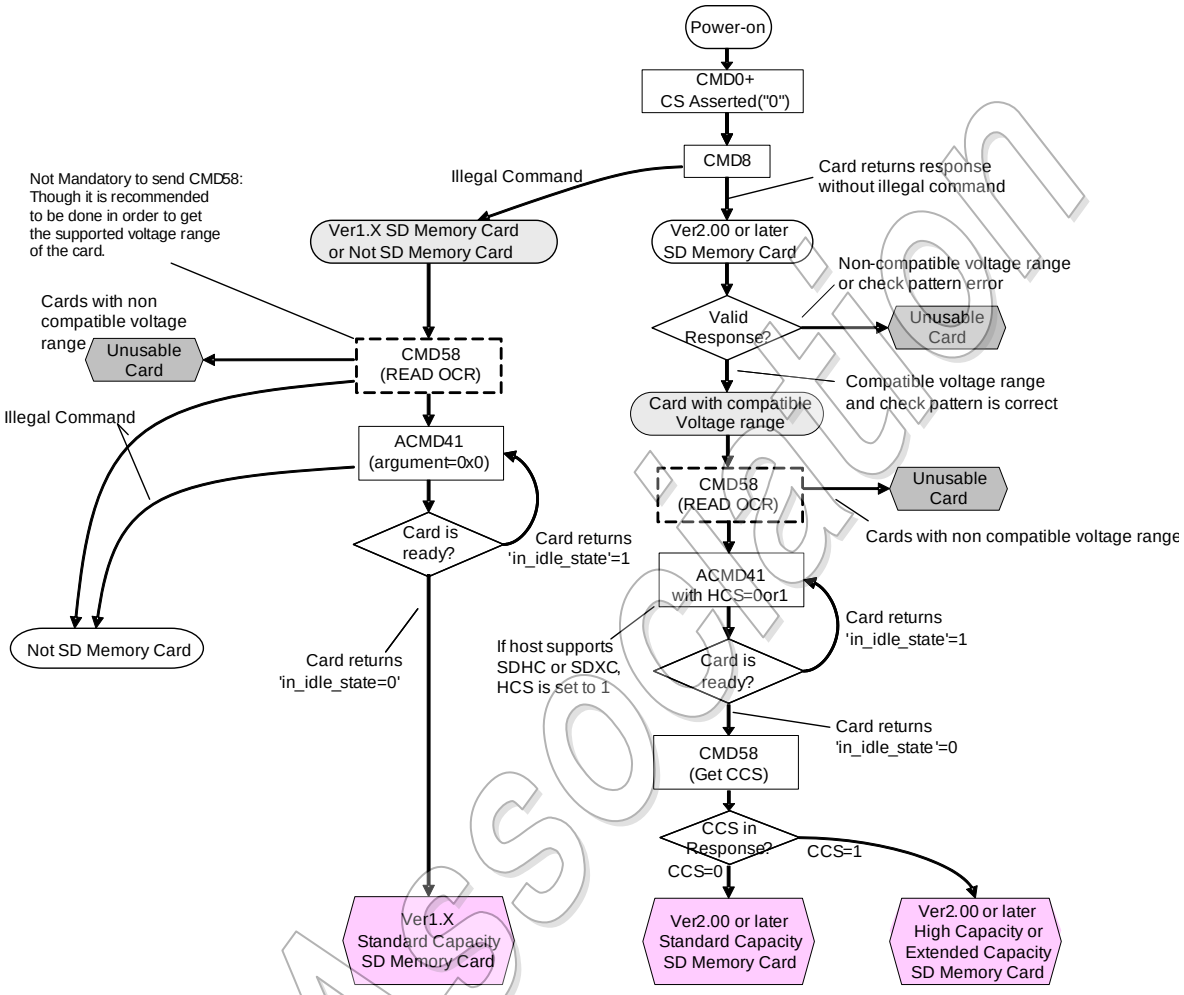


Figure 7-2 : SPI Mode Initialization Flow

READ_OCR (CMD58) is designed to provide SD Memory Card hosts with a mechanism to identify cards that do not match the V_{DD} range desired by the host. If the host does not accept voltage range, it shall not proceed with further initialization sequence. The levels in the OCR register shall be defined accordingly (See Section 5.1).

SD_SEND_OP_COND (ACMD41) is used to start initialization and to check if the card has completed initialization. It is mandatory to issue CMD8 prior to the first ACMD41. Receiving of CMD8 expands the CMD58 and ACMD41 function; HCS (High Capacity Support) in the argument of ACMD41 and CCS (Card Capacity Status) in the response of CMD58. HCS is ignored by the card, which didn't accept CMD8. Standard Capacity SD Memory Card ignores HCS. The "in idle state" bit in the R1 response of ACMD41 is used by the card to inform the host if initialization of ACMD41 is completed. Setting this bit to "1" indicates that the card is still initializing. Setting this bit to "0" indicates completion of initialization. The host repeatedly issues ACMD41 until this bit is set to "0". The card checks the HCS bit in the OCR only at the first ACMD41. While repeating ACMD41, the host shall not issue another command except CMD0. For SDUC card, card may indicate that the card is still initializing in the R3 response of ACMD41 continuously to let host know SDUC card cannot use SPI mode.

After initialization is completed, the host should get CCS information in the response of CMD58. CCS is

valid when the card accepted CMD8 and after the completion of initialization. CCS=0 means that the card is SDSC. CCS=1 means that the card is SDHC or SDXC.

7.2.2 Bus Transfer Protection

Every SD Card command transferred on the bus is protected by CRC bits. In SPI mode, the SD Memory Card offers a CRC ON mode which enables systems built with reliable data links to exclude the hardware or firmware required for implementing the CRC generation and verification functions.
In the CRC OFF mode, the CRC bits of the command are defined as 'don't care' for the transmitter and ignored by the receiver.
The SPI interface is initialized in the CRC OFF mode in default. However, the RESET command (CMD0) that is used to switch the card to SPI mode, is received by the card while in SD mode and, therefore, shall have a valid CRC field.
Since CMD0 has no arguments, the content of all the fields, including the CRC field, are constants and need not be calculated in run time. A valid reset command is:

0x40, 0x0, 0x0, 0x0, 0x0, 0x95

After the card is put into SPI mode, CRC check for all commands including CMD0 will be done according to CMD59 setting.
The host can turn the CRC option on and off using the CRC_ON_OFF command (CMD59). Host should enable CRC verification before issuing ACMD41.
The CMD8 CRC verification is always enabled. The Host shall set correct CRC in the argument of CMD8. If CRC error is detected, card returns CRC error in R1 response regardless of command index.

7.2.3 Data Read

The SPI mode supports single block read and Multiple Block read operations (CMD17 or CMD18 in the SD Memory Card protocol). Upon reception of a valid read command the card will respond with a response token followed by a data token (Refer to Figure 7-3). In case of Standard Capacity Card, the size in the data token is determined by the block length set by SET_BLOCKLEN (CMD16). In case of SDHC and SDXC Cards, block length is fixed to 512 Bytes regardless of the block length set by CMD16.

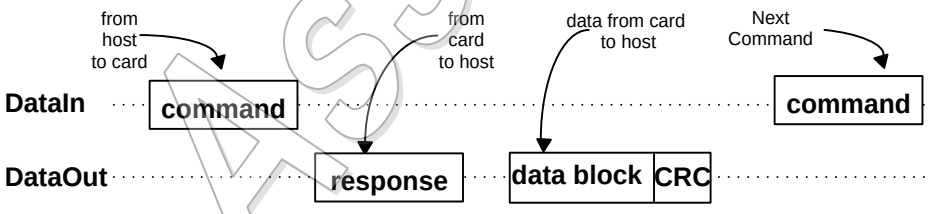


Figure 7-3 : Single Block Read Operation

A valid data block is suffixed with a 16-bit CRC generated by the standard CCITT polynomial $x^{16}+x^{12}+x^5+1$. The maximum block length is given by 512 Bytes regardless of READ_BL_LEN, defined in the CSD. If partial block access is enabled in Standard Capacity Card (i.e. the CSD parameter READ_BL_PARTIAL equals 1), the block length can be any number between 1 and 512 Bytes. The start address can be any byte address in the valid address range of the card. Every block, however, shall be contained in a single physical card sector.
If partial block access is disabled, only 512-Byte data length is supported.
SDHC and SDXC Cards only support 512-byte block length. The start address shall be aligned with the block boundary.

In the case of a data retrieval error, the card will not transmit any data. Instead, a special data error token will be sent to the host. Figure 7-4 shows a data read operation that terminated with an error token rather than a data block.

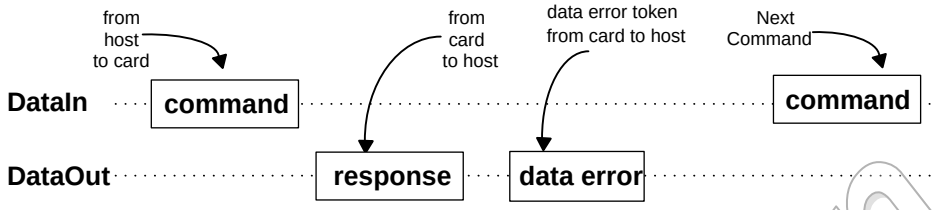


Figure 7-4 : Read Operation - Data Error

In the case of a multiple block read operation every transferred block has its suffix of 16-bit CRC. Stop transmission command (CMD12) will actually stop the data transfer operation (the same as in SD Memory Card operation mode).

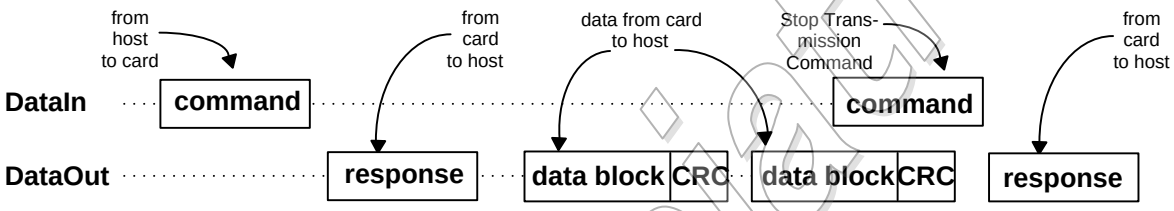


Figure 7-5 : Multiple Block Read Operation

7.2.4 Data Write

The SPI mode supports single block and multiple block write commands. Upon reception of a valid write command (CMD24 or CMD25 in the SD Memory Card protocol), the card will respond with a response token and will wait for a data block to be sent from the host. CRC suffix, block length and start address restrictions are (with the exception of the CSD parameter WRITE_BL_PARTIAL controlling the partial block write option and WRITE_BL_LEN) identical to the read operation (Refer to Figure 7-6).

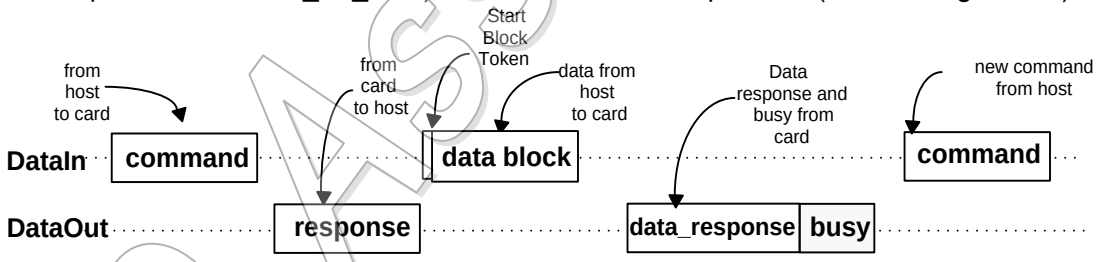


Figure 7-6 : Single Block Write Operation

Every data block has a prefix of 'Start Block' token (one byte).
After a data block has been received, the card will respond with a data-response token. If the data block has been received without errors, it will be programmed. As long as the card is busy programming, a continuous stream of busy tokens will be sent to the host (effectively holding the DataOut line low).
Once the programming operation is completed, the host should check the results of the programming using the SEND_STATUS command (CMD13). Some errors (e.g. address out of range, write protect violation etc.) are detected during programming only. The only validation check performed on the data block, and communicated to the host via the data-response token, is the CRC and general Write Error indication.
In a Multiple Block write operation, the stop transmission will be done by sending 'Stop Tran' token instead of 'Start Block' token at the beginning of the next block. In case of Write Error indication (on the data

response) the host shall use SEND_NUM_WR_BLOCKS (ACMD22) in order to get the number of well written write blocks. The data tokens description is given in Section 7.3.3.2.

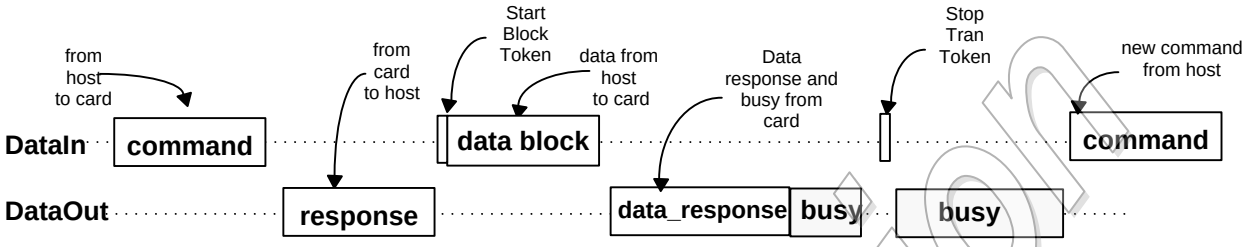


Figure 7-7 : Multiple Block Write Operation

While the card is busy, resetting the CS signal will not terminate the programming process. The card will release the DataOut line (tri-state) and continue with programming. If the card is reselected before the programming is finished, the DataOut line will be forced back to low and all commands will be rejected. Resetting a card (using CMD0 for SD memory card) will terminate any pending or active programming operation. This may destroy the data formats on the card. It is in the responsibility of the host to prevent this for occurring.

7.2.5 Erase & Write Protect Management

The erase and write protect management procedures in the SPI mode are identical to those of the SD mode. While the card is erasing or changing the write protection bits of the predefined sector list, it will be in a busy state and hold the DataOut line low. Figure 7-8 illustrates a 'no data' bus transaction with and without busy signaling.

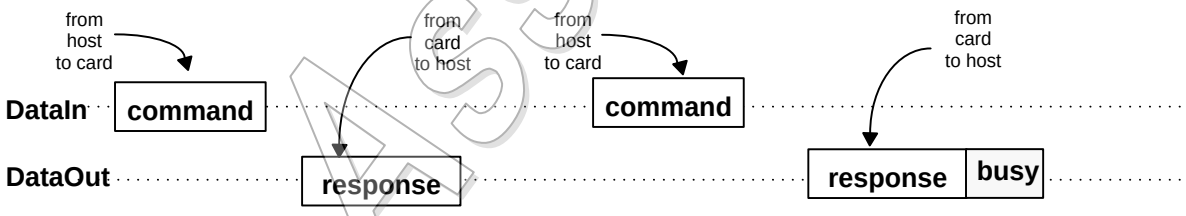


Figure 7-8 : 'No data' Operations

7.2.6 Read CID/CSD Registers

Unlike the SD Memory Card protocol (where the register contents is sent as a command response), reading the contents of the CSD and CID registers in SPI mode is a simple read-block transaction. The card will respond with a standard response token (Refer to Figure 7-3) followed by a data block of 16 bytes suffixed with a 16-bit CRC.

The data timeout for the CSD command cannot be set to the cards TAAC since this value is stored in the card's CSD. Therefore, the standard response timeout value (N_{CR}) is used for read latency of the CSD register.

7.2.7 Reset Sequence

The SD Memory Card requires a defined reset sequence. The card enters an idle state after power on reset or reset command (CMD0 for SD memory card). In this state, the only valid host commands are CMD8 (SEND_IF_COND), ACMD41 (SD_SEND_OP_COND), CMD58 (READ_OCR) and CMD59 (CRC_ON_OFF).

For the Thick (2.1 mm) SD Memory Card - CMD1 (SEND_OP_COND) is also valid - this means that in SPI mode, CMD1 and ACMD41 have the same behaviors, but the usage of ACMD41 is preferable since it allows easy distinction between an SD Memory Card and a MultiMediaCard. **For the Thin (1.4 mm) Standard Size SD Memory Card, CMD1 (SEND_OP_COND) is an illegal command during the initialization that is done after power on. After Power On, once the card has accepted valid ACMD41, it will be able to also accept CMD1 even if used after re-initializing (CMD0) the card.** It was defined in such way in order to be able to distinguish between a Thin SD Memory Card and a MultiMediaCard (that supports CMD1 as well).

7.2.8 Error Conditions

Unlike the SD Memory Card protocol, in the SPI mode, the card will always respond to a command. The response indicates acceptance or rejection of the command. A command may be rejected in any one of the following cases:

- It is sent while the card is in read operation (except CMD12 which is legal).
- It is sent while the card is in Busy.
- Card is locked and it is other than Class 0 or 7 commands.
- It is not supported (illegal opcode).
- CRC check failed.
- It contains an illegal operand.
- It was out of sequence during an erase sequence.

Note that in case the host sends command while the card sends data in read operation then the response with an illegal command indication may disturb the data transfer.

7.2.9 Memory Array Partitioning

Same as SD mode.

7.2.10 Card Lock/Unlock

Usage of card lock and unlock commands in SPI mode is identical to SD mode. In both cases, the command is responded to with an R1b response type. After the busy signal clears, the host should obtain the result of the operation by issuing a SEND_STATUS command (CMD13). Refer to Section 4.3.7 for details.

7.2.11 Application Specific Commands

Identical to SD mode with the exception of the APP_CMD status bit (Refer to Table 4-42), which is not available in SPI.

7.2.12 Content Protection Command

All the special Content Protection ACMDs and security functionality related to the CPRM is the same as SD mode.

7.2.13 Switch Function Command

Same as for SD mode with two exceptions:

- The command is valid under the "not idle state".
- The switching period is within 8 clocks after the end bit of the R1 response of CMD0.

7.2.14 High Speed Mode

Same as SD mode.

7.2.15 Speed Class Specification

As opposed to SD mode, the card cannot guarantee its Speed Class. In SPI mode, host shall treat the card as Class 0 no matter what Class is indicated in SD Status.

7.2.16 Boot Functionalities

Boot functionalities including Fast Boot are not supported in SPI mode.

7.2.17 TCG Security

When TCG is enabled, host can read data from TCG MBR Table by SPI mode before pre-boot authentication. However, since host cannot execute pre-boot authentication over SPI-mode, host cannot access the User Area properly in this mode.

7.2.18 RPMB

Host cannot access to RPMB Unit and execute RPMB authentication in SPI mode.

7.3 SPI Mode Transaction Packets

7.3.1 Command Tokens

7.3.1.1 Command Format

All the SD Memory Card commands are 6 bytes long. The command transmission always starts with the left most bit of the bit string corresponding to the command codeword. All commands are protected by a CRC (see Section 4.5). The commands and arguments are listed in Table 7-3.

Bit position	47	46	[45:40]	[39:8]	[7:1]	0
Width (bits)	1	1	6	32	7	1
Value	'0'	'1'	x	x	x	'1'
Description	start bit	transmission bit	command index	argument	CRC7	end bit

Table 7-1 : Command Format

7.3.1.2 Command Classes

As in SD mode, the SPI commands are divided into several classes (See Table 7-2). Each class supports a set of card functions. A SD Memory Card will support the same set of optional command classes in both communication modes (there is only one command class table in the CSD register). The available command classes, and the supported command for a specific class, however, are different in the SD Memory Card and the SPI communication mode.

Note that except for the classes that are not supported in SPI mode (class 1, 3 and 9), the mandatory required classes for the SD mode are the same for the SPI mode.

CMD58/59 in SD mode are different from those in SPI mode.

Card Command Class (CCC)		**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
Supported commands	**class description**	**basic**	**reserved**	**block read**	**reserved**	**block write**	**erase**	**write protection**	**lock card**	**application specific**	**I/O mode**	**switch**	**reserved**
CMD0	Mandatory	+											
CMD1	Mandatory	+											
CMD5	Optional										+		
CMD6[2]	Mandatory											+	
CMD8[3]	Mandatory	+											
CMD9	Mandatory	+											
CMD10	Mandatory	+											
CMD12	Mandatory	+											
CMD13	Mandatory	+											
CMD16	Mandatory			+		+			+				
CMD17	Mandatory			+									
CMD18	Mandatory			+									
CMD24	Mandatory[1]					+							

Card Command Class (CCC)		0	1	2	3	4	5	6	7	8	9	10	11
Supported commands	**class description**	**basic**	**reserved**	**block read**	**reserved**	**block write**	**erase**	**write protection**	**lock card**	**application specific**	**I/O mode**	**switch**	**reserved**
CMD25	Mandatory[1]					+							
CMD27	Mandatory[1]					+							
CMD28	Optional							+					
CMD29	Optional							+					
CMD30	Optional							+					
CMD32	Mandatory[1]						+						
CMD33	Mandatory[1]						+						
CMD34-37[2]	Optional											+	
CMD38	Mandatory[1]						+						
CMD42[4]	(Note 4)								+				
CMD50[2]	Optional											+	
CMD52	Optional										+		
CMD53	Optional										+		
CMD55	Mandatory									+			
CMD56	Mandatory									+			
CMD57[2]	Optional											+	
CMD58	Mandatory	+											
CMD59	Mandatory	+											
ACMD13	Mandatory									+			
ACMD22	Mandatory[1]									+			
ACMD23	Mandatory[1]									+			
ACMD41	Mandatory									+			
ACMD42	Mandatory									+			
ACMD51	Mandatory									+			

Note (1): The commands related write and erase are mandatory only for the Writable types of Cards.
Note (2): This command was defined in spec version 1.10
Note (3): This command is newly defined in version 2.00
Note (4): This command is optional in Version 1.01 and 1.10 and mandatory from Version 2.00. COP support is optional for CMD42

Table 7-2 : Command Classes in SPI Mode

7.3.1.3 Detailed Command Description

The following table provides a detailed description of the SPI bus commands. The responses are defined in Section 7.3.2. Table 7-3 lists all SD Memory Card commands. A "yes" in the SPI mode column indicates that the command is supported in SPI mode. With these restrictions, the command class description in the CSD is still valid. If a command does not require an argument, the value of this field should be set to zero. The reserved commands are reserved in SD mode as well.

The binary code of a command is defined by the mnemonic symbol. As an example, the content of the **command index** field is (binary) '000000' for CMD0 and '100111' for CMD39.

The card shall ignore stuff bits and reserved bits in an argument.

CMD INDEX	SPI Mode	Argument	Resp	Abbreviation	Command Description
CMD0	Yes	[31:0] stuff bits	R1	GO_IDLE_STATE	Resets the SD Memory Card
CMD1	Yes[1]	[31]Reserved bit [30]HCS [29:0]Reserved bits	R1	SEND_OP_ COND	Sends host capacity support information and activates the card's initialization process. HCS is effective when card receives SEND_IF_COND command. Reserved bits shall be set to '0'.
CMD2	No				
CMD3	No				
CMD4	No				
CMD5	Reserved for I/O Mode (refer to "SDIO Card Specification")				
CMD6[8]	Yes	[31] Mode 0:Check function 1:Switch function [30:24] reserved (All '0') [23:20] reserved for function group 6 (All '0' or 0xF) [19:16] reserved for function group 5 (All '0' or 0xF) [15:12] reserved for function group 4 (All '0' or 0xF) [11:8] reserved for function group 3 (All '0' or 0xF) [7:4] function group 2 for command system [3:0] function group 1 for access mode	R1	SWITCH_FUNC	Checks switchable function (mode 0) and switches card function (mode 1). See Section 4.3.10.
CMD7	No				

CMD INDEX	SPI Mode	Argument	Resp	Abbreviation	Command Description
CMD8[9]	Yes	[31:12]Reserved bits [11:8]supply voltage(VHS) [7:0]check pattern	R7	SEND_IF_COND	Sends SD Memory Card interface condition that includes host supply voltage information and asks the accessed card whether card can operate in supplied voltage range. Reserved bits shall be set to '0'.
CMD9	Yes	[31:0] stuff bits	R1	SEND_CSD	Asks the selected card to send its card-specific data (CSD)
CMD10	Yes	[31:0] stuff bits	R1	SEND_CID	Asks the selected card to send its card identification (CID)
CMD11	No				
CMD12	Yes	[31:0] stuff bits	R1b[5]	STOP_ TRANSMISSION	Forces the card to stop transmission in Multiple Block Read Operation
CMD13	Yes	[31:0] stuff bits	R2	SEND_STATUS	Asks the selected card to send its status register.
CMD14	reserved				
CMD15	No				
CMD16	Yes	[31:0] block length	R1	SET_BLOCKLEN	In case of SDSC Card, block length is set by this command. In case of SDHC and SDXC Cards, block length of the memory access commands are fixed to 512 bytes. The length of LOCK_UNLOCK command is set by this command regardless of card capacity.
CMD17	Yes	[31:0] data address[10]	R1	READ_SINGLE_ BLOCK	Reads a block of the size selected by the SET_BLOCKLEN command.[3]
CMD18	Yes	[31:0] data address[10]	R1	READ_MULTIPL E_BLOCK	Continuously transfers data blocks from card to host until interrupted by a STOP_TRANSMISSION command.
CMD19	reserved				
CMD20	No				
CMD21... CMD23	reserved				
CMD24	Yes	[31:0] data address[10]	R1	WRITE_BLOCK	Writes a block of the size selected by the SET_BLOCKLEN command.[4]
CMD25	Yes	[31:0] data address[10]	R1	WRITE_MULTIPL E_BLOCK	Continuously writes blocks of data until 'Stop Tran' token is sent (instead 'Start Block').
CMD26	No				
CMD27	Yes	[31:0] stuff bits	R1	PROGRAM_CSD	Programming of the programmable bits of the CSD.

CMD INDEX	SPI Mode	Argument	Resp	Abbreviation	Command Description
CMD28	Yes	[31:0] data address	R1b[5]	SET_WRITE_ PROT	If the card has write protection features, this command sets the write protection bit of the addressed group. The properties of write protection are coded in the card specific data (WP_GRP_SIZE). SDHC and SDXC Cards do not support this command.
CMD29	Yes	[31:0] data address	R1b[5]	CLR_WRITE_ PROT	If the card has write protection features, this command clears the write protection bit of the addressed group. SDHC and SDXC Cards do not support this command.
CMD30	Yes	[31:0] write protect data address	R1	SEND_WRITE_ PROT	If the card has write protection features, this command asks the card to send the status of the write protection bits.[6] SDHC and SDXC Cards do not support this command.
CMD31	reserved				
CMD32	Yes	[31:0] data address[10]	R1	ERASE_WR_BL K_START_ADDR	Sets the address of the first write block to be erased.
CMD33	Yes	[31:0] data address[10]	R1	ERASE_WR_BL K_END_ADDR	Sets the address of the last write block of the continuous range to be erased.
CMD34-37[8]	Reserved for each command system set by switch function command (CMD6). Refer to each command system specification for more detail.				
CMD38	Yes	[31:0] stuff bits	R1b[5]	ERASE	Erases all previously selected write blocks. FULE and DISCARD are not supported through SPI interface.
CMD39	No				
CMD40	No				
CMD41	Reserved				
CMD42	Yes	[31:0] Reserved bits (Set all 0)	R1	LOCK_UNLOCK	Used to Set/Reset the Password or lock/unlock the card. A transferred data block includes all the command details - refer to Section 4.3.7. The size of the Data Block is defined with SET_BLOCK_LEN command. Reserved bits in the argument and in Lock Card Data Structure shall be set to 0.
CMD43-49 CMD51	reserved				
CMD50[8]	Reserved for each command system set by switch function command (CMD6). Refer to each command system specification for more detail.				
CMD52-54	Reserved for I/O Mode (refer to "SDIO Card Specification")				

CMD INDEX	SPI Mode	Argument	Resp	Abbreviation	Command Description
CMD55	Yes	[31:0] stuff bits	R1	APP_CMD	Defines to the card that the next command is an application specific command rather than a standard command
CMD56	Yes	[31:1] stuff bits. [0]: RD/WR[7]	R1	GEN_CMD	Used either to transfer a Data Block to the card or to get a Data Block from the card for general purpose/application specific commands. In case of Standard Capacity SD Memory Card, the size of the Data Block shall be defined with SET_BLOCK_LEN command. In case of SDHC and SDXC Cards, block length of this command is fixed to 512-byte.
CMD57[8]	Reserved for each command system set by switch function command (CMD6). Refer to each command system specification for more detail.				
CMD58	Yes	[31:0] stuff bits	R3	READ_OCR	Reads the OCR register of a card. CCS bit is assigned to OCR[30].
CMD59	Yes	[31:1] stuff bits [0:0] CRC option	R1	CRC_ON_OFF	Turns the CRC option on or off. A '1' in the CRC option bit will turn the option on, a '0' will turn it off
CMD60-63	Reserved For Manufacturer				

1. CMD1 is valid command for the Thin (1.4mm) Standard Size SD Memory Card only if used after re-initializing a card (not after power on reset).
2. The default block length is as specified in the CSD.
3. The data transferred shall not cross a physical block boundary unless READ_BLK_MISALIGN is set in the CSD.
4. The data transferred shall not cross a physical block boundary unless WRITE_BLK_MISALIGN is set in the CSD.
5. R1b: R1 response with an optional trailing busy signal
6. 32 write protection bits (representing 32 write protect groups starting at the specified address) followed by 16 CRC bits are transferred in a payload format via the data line. The last (least significant) bit of the protection bits corresponds to the first addressed group. If the addresses of the last groups are outside the valid range, then the corresponding write protection bits shall be set to zero
7. RD/WR_: "1" the Host shall get a block of data from the card.
   "0" the host sends block of data to the card.
8. This command was added in spec version 1.10
9. This command is added in spec version 2.00
10. SDSC Card (CCS=0) uses byte unit address and SDHC and SDXC Cards (CCS=1) use block unit address (512 bytes unit).

Table 7-3 : Commands and Arguments

The following table describes all the application specific commands supported/reserved by the SD Memory Card. All the following commands shall be preceded with APP_CMD (CMD55).

CMD INDEX	SPI Mode	Argument	Resp	Abbreviation	Command Description
ACMD6	No				
ACMD13	yes	[31:0] stuff bits	R2	SD_STATUS	Send the SD Status. The status fields are given in Table 4-44
ACMD17	reserved				
ACMD18	yes	--	--	--	Reserved for SD security applications[1]
ACMD19-ACMD21	reserved				
ACMD22	yes	[31:0] stuff bits	R1	SEND_NUM_WR_BLOCKS	Send the numbers of the well written (without errors) blocks. Responds with 32-bit+CRC data block.
ACMD23	yes	[31:23] stuff bits [22:0]Number of blocks	R1	SET_WR_BLK_ERASE_COUNT	Set the number of write blocks to be pre-erased before writing (to be used for faster Multiple Block WR command). "1"=default (one wr block)(2).
ACMD24	reserved				
ACMD25	yes	--	--	--	Reserved for SD security applications[1]
ACMD26	yes	--	--	--	Reserved for SD security applications[1]
ACMD38	yes	--	--	--	Reserved for SD security applications[1]
ACMD39 - ACMD40	reserved				
ACMD41	Yes	[31]Reserved bit [30]HCS [29:0]Reserved bits	R1	SD_SEND_OP_COND	Sends host capacity support information and activates the card's initialization process. Reserved bits shall be set to '0'
ACMD42	yes	[31:1] stuff bits [0]set_cd	R1	SET_CLR_CARD_DETECT	Connect[1]/Disconnect[0] the 50 KOhm pull-up resistor on CS (pin 1) of the card. The pull-up may be used for card detection.
ACMD43-ACMD49	yes	--	--	--	Reserved for SD security applications[1]
ACMD51	yes	[31:0] stuff bits	R1	SEND_SCR	Reads the SD Configuration Register (SCR).
ACMD53 - ACMD54	No				

(1) Refer to "Part3 Security Specification" for detailed explanation about the SD Security Features
(2) Stop Tran Token shall be used to stop the transmission in Write Multiple Block whether the pre-erase (ACMD23) feature is used or not.

Table 7-4 : Application Specific Commands used/reserved by SD Memory Card - SPI Mode

7.3.1.4 Card Operation for CMD8 in SPI mode

In SPI mode, the card always returns response. Table 7-5 shows the card operation for CMD8.

Command Argument Check					Response of Card *1			
Index	**Reserved**	**VHS**	**Pattern**	**CRC**	**R1**	**Reserved**	**VCA**	**Pattern**
=8	Don't Care	Don't Care	Don't Care	Error	09h	(R1 only)		
Not 8	Don't Care	Don't Care	Don't Care	Don't Care	Depends on command index			
=8	Don't Care	Mismatch *2	Don't Care	Correct	01h	0	0	Echo Back
=8	Don't Care	Match *2	Don't Care	Correct	01h	0	Echo Back	Echo Back

*1: Response indicates the actual response that the card returns. (It does not include errors during transfer response.)
*2: 'Match' means AND of following condition a) and b). 'Mismatch' is other cases.
a) Only 1 bit is set to '1' in VHS.
b) The card supports the host supply voltage.

Table 7-5 : Card Operation for CMD8 in SPI Mode

7.3.2 Responses

There are several types of response tokens. As in SD mode, all are transmitted MSB first.
Multiple bytes responses are defined in SPI mode but the card outputs only first byte (equivalent to R1) when Illegal Command Error or Command CRC Error is indicated in it. In this case, host never reads as the multiple bytes of response.

7.3.2.1 Format R1

This response token is sent by the card after every command with the exception of SEND_STATUS commands. It is one byte long, and the MSB is always set to zero. The other bits are error indications, an error being signaled by a 1. The structure of the R1 format is given in Figure 7-9. The meaning of the flags is defined as following:

In idle state: The card is in idle state and running the initializing process.
Erase reset: An erase sequence was cleared before executing because an out of erase sequence command was received.
Illegal command: An illegal command code was detected.
Communication CRC error: The CRC check of the last command failed.
Erase sequence error: An error in the sequence of erase commands occurred.
Address error: A misaligned address that did not match the block length was used in the command.
Parameter error: The command's argument (e.g. address, block length) was outside the allowed range for this card.

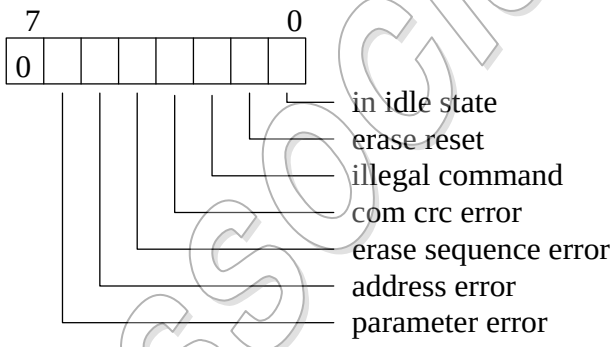


Figure 7-9 : R1 Response Format

7.3.2.2 Format R1b

This response token is identical to the R1 format with the optional addition of the busy signal. The busy signal token can be any number of bytes. A zero value indicates card is busy. A non-zero value indicates the card is ready for the next command.

7.3.2.3 Format R2

This response token is two bytes long and sent as a response to the SEND_STATUS command. The format is given in Figure 7-10.

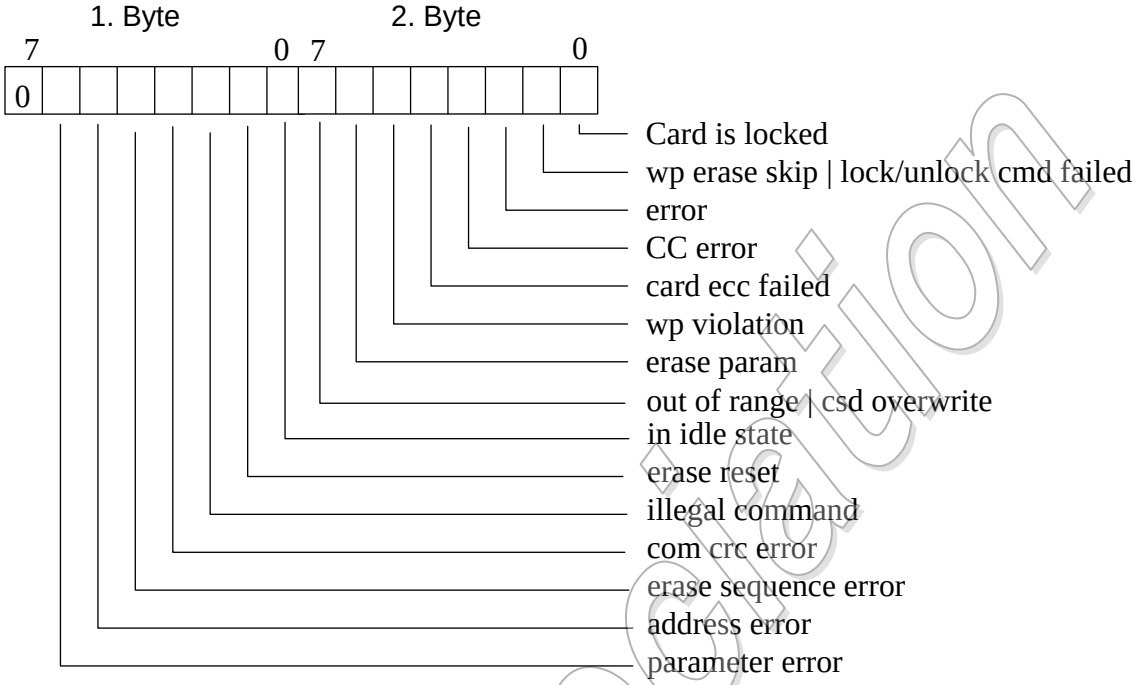


Figure 7-10 : R2 Response Format

The first byte is identical to the response R1. The content of the second byte is described in the following:

Erase param: An invalid selection for erase, sectors or groups.

Write protect violation: The command tried to write a write-protected block.

Card ECC failed: Card internal ECC was applied but failed to correct the data.

CC error: Internal card controller error.

Error: A general or an unknown error occurred during the operation.

Write protect erase skip | lock/unlock command failed:

This status bit has two functions overloaded. It is set when the host attempts to erase a write-protected sector or makes a sequence or password errors during card lock/unlock operation.

Card is locked: Set when the card is locked by the user. Reset when it is unlocked.

7.3.2.4 Format R3

This response token is sent by the card when a READ_OCR command is received. The response length is 5 bytes (see Figure 7-11). The structure of the first (MSB) byte is identical to response type R1. The other four bytes contain the OCR register.

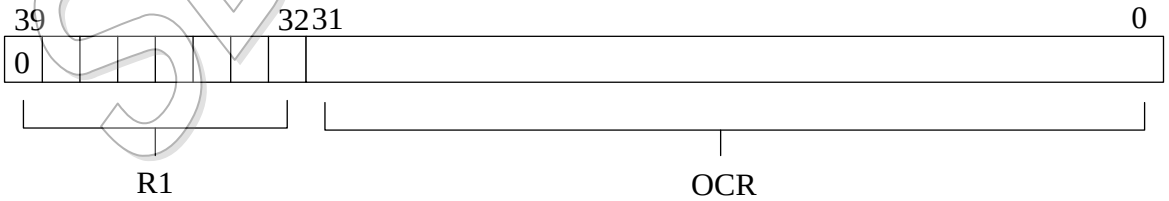


Figure 7-11 : R3 Response Format

7.3.2.5 Formats R4 & R5

Those response formats are reserved for I/O mode (refer to "SDIO Card Specification").

7.3.2.6 Format R7

This response token is sent by the card when a SEND_IF_COND command (CMD8) is received. The response length is 5 bytes. The structure of the first (MSB) byte is identical to response type R1. The other four bytes contain the card operating voltage information and echo back of check pattern in argument and are specified by the same definition as R7 response in SD mode. (Refer to Section 4.9).

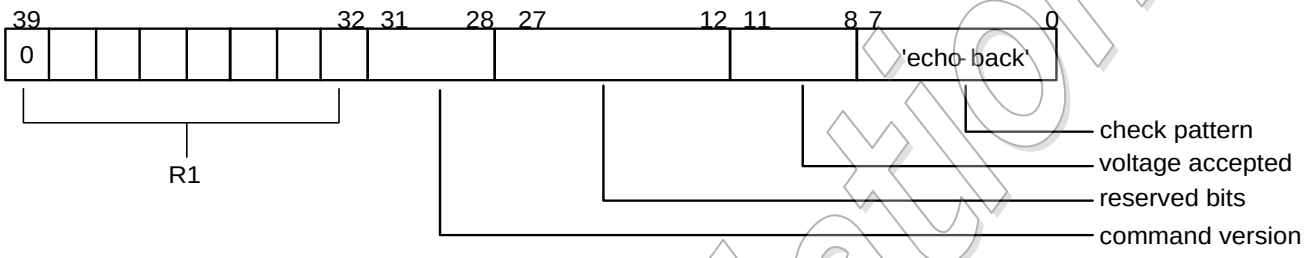


Figure 7-12 : R7 Response Format

7.3.3 Control Tokens

Data block transfer is controlled by some tokens.

7.3.3.1 Data Response Token

Every data block written to the card will be acknowledged by a data response token. It is one byte long and has the following format:

7	6	5	4	3	2	1	0
x	x	x	0	Status			1

The meaning of the status bits is defined as follows:

'010' - Data accepted.
'101' - Data rejected due to a CRC error.
'110' - Data Rejected due to a Write Error

In case of any error (CRC or Write Error) during Write Multiple Block operation, the host shall stop the data transmission using CMD12. In case of a Write Error (response '110'), the host may send CMD13 (SEND_STATUS) in order to get the cause of the write problem. ACMD22 can be used to find the number of well written write blocks.

7.3.3.2 Start Block Tokens and Stop Tran Token

Read and write commands have data transfers associated with them. Data is being transmitted or received via data tokens. All data bytes are transmitted MSB first.
Data tokens are 4 to 515 bytes long and have the following format:
For Single Block Read, Single Block Write and Multiple Block Read:

- First byte: Start Block

7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	0

- Bytes 2-513 (depends on the data block length): User data
- Last two bytes: 16 bit CRC.

For Multiple Block Write operation:

- First byte of each block:
  If data is to be transferred then - Start Block Token

7	6	5	4	3	2	1	0
1	1	1	1	1	1	0	0

If Stop transmission is requested - Stop Tran Token

7	6	5	4	3	2	1	0
1	1	1	1	1	1	0	1

Note that this format is used only for Multiple Block Write. In case of a Multiple Block Read the stop transmission is performed using STOP_TRAN Command (CMD12).

7.3.3.3 Data Error Token

If a read operation fails and the card cannot provide the required data, it will send a data error token instead. This token is one byte long and has the following format:

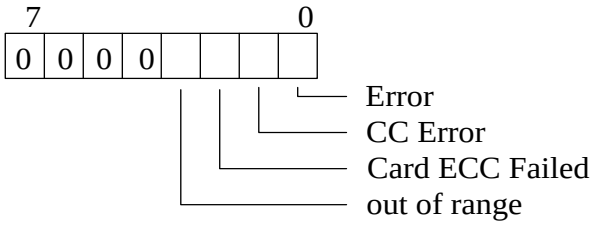


Figure 7-13 : Data Error Token

The 4 least significant bits (LSB) are the same error bits as in response format R2.

7.3.4 Clearing Status Bits

As described in the previous paragraphs, in SPI mode, status bits are reported to the host in three different formats: response R1, response R2, and data error token (the same bits may exist in multiple response types - e.g. Card ECC failed)

As in the SD mode, error bits are cleared when read by the host, regardless of the response format. State indicators are either cleared by reading or are cleared in accordance with the card state.

The following table summarizes the set and clear conditions for the various status bits:

Identifier	Included in resp	Type[1]	Value	Description	Clear Condition[2]
Out of range	R2 DataErr	E R X	'0'= no error '1'= error	The command argument was out of the allowed range for this card.	C
Address error	R1 R2	E R X	'0'= no error '1'= error	A misaligned address which did not match the block length was used in the command.	C
Erase sequence error	R1 R2	E R	'0'= no error '1'= error	An error in the sequence of erase commands occurred.	C
Erase param	R2	E X	'0'= no error '1'= error	An error in the parameters of the erase command sequence	C
Parameter error	R1 R2	E R X	'0'= no error '1'= error	An error in the parameters of the command	C
WP violation	R2	E R X	'0'= not protected '1'= protected	Attempt to program a write protected block.	C
Com CRC error	R1 R2	E R	'0'= no error '1'= error	The CRC check of the command failed.	C
Illegal command	R1 R2	E R	'0'= no error '1'= error	Command not legal for the card state	C
Card ECC failed	R2 DataEr	E X	'0'= success '1'= failure	Card internal ECC was applied but failed to correct the data.	C
CC error	R2 dataEr	E R X	'0'= no error '1'= error	Internal card controller error	C
Error	R2 dataEr	E R X	'0'= no error '1'= error	A general or an unknown error occurred during the operation.	C

Identifier	Included in resp	Type[1]	Value	Description	Clear Condition[2]
CSD_OVERWRITE	R2	E R X	'0'= no error '1'= error	Can be either of the following errors: - The read only section of the CSD does not match the card content. - An attempt to reverse the copy (set as original) or permanent WP (unprotected) bits was made.	C
WP erase skip	R2	S X	'0'= not protected '1'= protected	Only partial address space was erased due to existing write protected blocks.	C
Lock/Unlock cmd failed	R2	X	'0'= no error '1'= error	Sequence or password errors during card lock/unlock operation.	C
Card is locked	R2	S X	'0' = card is not locked '1' = card is locked	Card is locked by a user password.	A
Erase reset	R1 R2	S R	'0'= cleared '1'= set	An erase sequence was cleared before executing because an out of erase sequence command was received	C
In Idle state	R1 R2	S R	0 = Card is ready 1 = Card is in idle state	The card enters the idle state after power up or reset command. It will exit this state and become ready upon completion of its initialization procedures.	A

Table 7-6 : SPI Mode Status Bits

1) Type:
E: Error bit.
S: State bit.
R: Detected and set for the actual command response.
X: Detected and set during command execution. The host can get the status by issuing a command with R1 response.

2) Clear Condition:
A: According to the current state of the card.
C: Clear by read

7.4 Card Registers

In SPI mode, only the RCA register is not accessible. Formats of other registers are identical to the formats in the SD mode.

7.5 SPI Bus Timing Diagrams

This section is a blank in the Simplified Specification

7.6 SPI Electrical Interface

The electrical interface is identical to SD mode with the exception of the programmable card output drivers' option, which is not supported in SPI mode.

7.7 SPI Bus Operating Conditions

Bus operating conditions are identical to SD mode

7.8 Bus Timing

Bus timing is identical to SD mode. The timing of the CS signal is the same as any other card input.

8. PCIe/NVMe Mode in SD Express card

8.1 Functional Description

The functional description of the PCIe interface is defined in PCI-SIG specs (refer to Appendix A.2.1).
The functional description of the NVMe is defined in NVM Express specs (refer to Appendix A.2.2).

8.1.1 PCIe Interface Identification Class

SD Express card in PCIe mode of operation shall be identified as :
Standard Non Volatile Memory subsystem – NVM Express Interface.
In reference to the PCI-SIG standard – “PCI Code and ID Assignment Specification Revision 1.9” the identification shall be : Base Class=01h, Sub Class=08h and Programming Interface = 02h.

8.1.2 Hot Plug-In and Hot Removal

Hot Plug In and Hot Removal shall be supported by card and host.
Host may use the Card detection switch mechanism as SD Express card presence detect.

8.1.3 SD Bus Speed Modes Supported by SD Express card

Following performance modes shall be supported by SD Express card basic SD interface: DS, HS, SDR12, SDR25, DDR50 (only for microSD), SDR50. DDR50 (for full-size SD) and SDR104 are optional.

8.1.4 SD Optional functions for the SD interface in SD Express card

Following functions are defined as optional for the SD interface in SD Express card:

- All types of Speed Classes (SC, UHS SC and VSC)
- Application Performance Classes
- Command Queuing
- Self Maintenance
- Cache
- FULE/Discard
- Power Management
- Function Extension Register
- LVS support
- WP_UPC
- CPRM
- Boot Functionalities
- TCG Security
- RPMB

8.1.5 SD Features NOT supported by SD interface in SD Express card

Following functions shall not be supported by the SD interface in SD Express card in order to eliminate interoperability issues of the same card between SD interface and PCIe/NVMe interface hosts.

- UHS-II interface

8.1.6 SD Features Supported by SD interface in SD Express card but not supported or partially supported through the PCIe interface

Even if the following functions are supported through the SD Express card through the SD interface, they are not supported or partially supported through the PCIe interface:

- Password protection (card lock/unlock):
    - If an SD Express card has been put in a Locked state using the SD-Legacy interface in the device, then the device shall abort any NVMe commands that attempt to read from, write to,

or modify (e.g., namespace create/delete, format, sanitize) the media until the Locked state is cleared by a host using the SD-Legacy Interface. The Operation Denied status code shall be returned for Admin Command Set commands which attempt to read from, write to, or modify the media. The Access Denied status code shall be returned for NVM Command Set commands which attempt to read from, write to, or modify the media.

- Card Internal Write Protect (PWP/TWP) − For cards without NVMe Write Protect features:
  - If an SD Express card has been put in a Temporary Write Protected state or a Permanent Write Protected state using the SD-Legacy interface in the device, then the device shall abort any NVMe command that attempt to write to or modify the media, until the Write Protect condition is cleared by a host using the SD-Legacy Interface. The Operation Denied status code shall be returned for Admin Command Set commands which attempt to write to or modify (e.g., namespace create/delete, format, sanitize) the media. The Access Denied status code shall be returned for NVMe Command Set commands which attempt to write to or modify the media.
- Card Internal Write Protect (PWP/TWP/WP_UPC) − For cards with NVMe Write Protect features:
  - The NVMe features - Write Protect, Permanent Write Protect and Write Protect Until Power Cycle and the SD related features - Temporary Write Protect, Permanent Write Protect and Write Protect Until Power Cycle are expected to behave similarly through both interfaces (SD and PCIe) and shall be respected in the same manner whether a specific write protection state is set through one interface (i.e., NVMe/SD). For example – if card is set to Temporary Write Protect using the SD interface, any write or erase attempt through either the SD interface or the NVMe interface shall be restricted, and related proper error conditions shall be responded by the card as defined by each interface standard.
  - Also Note that for SD Express cards that operates with NVMe mode and supports PWP or WP_UPC, RPMB authentication will be required to modify the states of PWP and WP_UPC through either of the interfaces (as mandated by the NVMe standard)
- CPRM security
  - CPRM commands and functionality may be supported through the SD interface. PCIe interface shall not allow access to protected area of SD card. The capacity reported by NVMe identify command shall be equal to the User Data Area size over SD interface. CPRM commands are not supported over PCIe interface.

8.1.7 Register Mapping of Selected SD Registers into PCIe/NVMe Registers

The initial SD Express card specification is defined to allow standard PCIe/NVMe drivers to be used with SD Express cards in PCIe mode of operation.

Selected, limited SD unique identification information, is mapped into a similar vendor specific fields of the NVMe standard. The purpose of this mapping is:

- Align the basic information between the two interfaces – like Manufacturer, Model, Capacity, Serial Number, SD Express Spec revision
- Allow host to get the same information about the cards unrelated to the interface it is accessed from. Information that may be useful for field track, maintenance etc.

The mapping concept is ASCII to ASCII, binary using Hex mapping (1 byte → 2 ASCIIs, e.g. 01h is mapping to 3031h)

Table 8-1 describes the mapping of the SD interface information into the NVMe identification registers.

Register	SD	NVMe mapping	NVMe Size	Comments
Manufacturer ID	CID/MID (8bit)**	ICDS*/VID	2 bytes	VID (PCI-SIG) and MID (SDA) assigned values shall identify the same manufacturer
OEM ID	CID/OID(2bytes)	ICDS/SSVID	2 bytes	SSVID (PCI-SIG) and OID (SDA) assigned values shall identify the same manufacturer
Product name	CID/PNM (5bytes)	ICDS/MN[0:4]	40 ASCII	MN[7:49] will remain vendor specific

Product revision	CID/PRV (1byte)**	ICDS/FR[0:1]	8 bytes	FR[2:7] will remain vendor specific
Product serial number	CID/PSN (32bit)**	ICDS/SN[0:7]	20 ASCII	
Manufacturing date	CID/MDT(12bit)**	ICDS/SN[8:10]		SN[11:19] shall be zero
device size	CSD/C_SIZE	NCAP		Device Capacity shall match on SD and PCIe interfaces, but each interface will use the appropriate format to report
SD Physical Spec. Version	SCR	ICDS/MN[5:6]		MN[5] major spec revision, MN[6] minor spec revision
Number of RPMB units and Authentication mode	Security and Boot Register Set [Byte Offset 8]	ICDS [Byte Offset 312]	1 byte	3 bits are used to define number of RPMB units. 3 bits are used to define Authentication mode and 2 bits are unused Max number of RPMB units is 1 in SD Express.
RPMB Total Size	Security and Boot Register Set [Byte Offset 10]	ICDS [Byte Offset 314]	1 byte	One RPMB unit size indicated in 128KB units and 0's base (00h means 128KB, FFh means 32,768KB).
RPMB Access Size	Security and Boot Register Set [Byte Offset 11]	ICDS [Byte Offset 315]	1 byte	Max number of data per RPMB access in 512B units and 0's base (00h means 512B, FFh means 131,072B).
Boot Partition Size	Security and Boot Register Set [Byte Offset 24]	BPINFO.BPSZ	14 bits	Boot partition size indicated in 128KB units and 0's base (00h means 128KB, FFh means 32,768KB). Max boot partition size is 32,768KB in SD Express.
Active Boot Partition	Security and Boot Register Set [Byte Offset 256]	BPINFO.ABPID	1 bit	1 bit is used to define the Active Boot Partition. 7 bits are unused.
Boot Partition Protection Enable	Security and Boot Register Set [Byte Offset 16]	RPMB DCBDS [Byte Offset 0]	1 byte	1 bit is used to indicate whether Boot Partition Protection is enabled. 7 bits are unused.
Boot Partition Lock	Security and Boot Register Set [Byte Offset 17]	RPMB DCBDS [Byte Offset 1]	1 byte	2 bits are used to indicate whether each Boot Partition is locked. 6 bits are unused.
User Area / Namespace Write Protection Authentication Control	Security and Boot Register Set [Byte Offset 18]	RPMB DCBDS [Byte Offset 2]	1 byte	1 bit is used to indicate whether PWP can be set. 1 bit is used to indicate whether WP_UPC can to be set. 6 bits are unused.

(*) ICDS = Identify Controller Data Structure
(**) binary using Hex mapping
(***) RPMB DCBDS = RPMB Device Configuration Block Data Structure

Table 8-1 : Mapping of SD Standard Information Into NVMe Identification Registers

8.1.8 Power Limit control of SD Express card

SD Express host shall ensure power rail is fulfilling the Power Rating requirements of SD Express card as defined in Section 0. PCIe Slot Power Limit value assigned by the host shall be greater or equal to 1.8W in case host supports up to PCIe Gen3 x 1 lane, greater or equal to 2.8W in case host supports up

to PCIe Gen3 x 2 lanes or Gen4 x 1 lane, and greater or equal to 4.0W in case host supports up to PCIe Gen4 x 2 lanes.
To Ensure operation with lower power constraints, the host can utilize NVMe dynamic power management, as defined in NVMe spec (refer to Appendix A.2.2).
The number of power states implemented by SD Express card is returned in the Number of Power States Supported (NPSS) field in the Identify Controller data structure. SD Express card shall support at least power states defined in Table 8.1.8-1 according to each card type, and may optionally support up to a total of 32 power states. Power states are contiguously numbered starting with zero such that each subsequent power state consumes less than or equal to the maximum power consumed in the previous state. Thus, power state zero indicates the maximum power that the SD Express card is capable of consuming. The mandatory power states for SD Express Card are shown in Table 8.1.8-1. Note that default power state of the SD Express Card is 1.8W.

Card Type		
PCIe G3L1	**PCIe G3L2 PCIe G4L1**	**PCIe G4L2**
1.8W 1.44W 0.72W	2.8W 2.5W 1.8W 1.44W 0.72W	4.0W 3.2W 2.8W 2.5W 1.8W 1.44W 0.72W

Table 8.1.8-1 : Mandatory Power States for SD Express Card

In addition to the selected power state limitation, the card shall never consume more power than the maximum value of its bus operation mode as defined in Section 6.6.3.
For example:

- If card supports up to Gen4x2 mode but Gen3x1 mode is selected, the card shall never exceed 1.8W power consumption, even if the NVMe power state is set to 4.0W.
- If card supports up to Gen4x2 mode but Gen4x1 mode is selected, the card shall never exceed 2.8W power consumption, even if the NVMe power state is set to 4.0W.

The NVMe power states may be used by the host as part of the thermal control system by limiting the maximum allowed consumed power of a SD Express card. Refer to Mechanical addendum specification for further description of the relation between the maximum card power consumption and the maximum card case temperature.

8.1.9 NVMe Namespace for SD Express Card

SD Express card shall be configured with single namespace with capacity more than 2GB.

> Application Note:
> Based on the use case needs and card's supportability, the card may be reconfigured through its PCIe interface to multiple namespaces, or a single namespace whose capacity is less than or equal to 2GB. For such cards, compatibility to other hosts may be lost.

8.1.10 Notes for SD Express Card Access

The following functions are supported both through the SD interface and the PCIe interface, but their access methods are different:

- Boot Functionalities (also refer to Section 4.21.2)

- When SD Express card supports Boot Functionalities, host has exactly two Boot Partitions accessible through both SD interface and PCIe interface.
- For SD interface, when host reads data from a Boot Partition, it switches to the target Boot Partition by CMD39 and reads data by CMD17 or CMD18. When host writes data to the Boot Partition, it unlocks the target Boot Partition after RPMB authentication and writes data by CMD24 or CMD25.
- For reading data from a Boot Partition by PCIe interface, host writes BPRSEL register to start data transfer from the designated Boot Partition. In case of writing, on the other hand, host unlocks the target Boot Partition after RPMB authentication and submits a Firmware Commit command.
- Fast Boot is available only in SD interface.

- TCG Security
  - In SD Express card, TCG security functions are realized by ACMD53/54 in SD interface. On the other hand, they are realized by Security Send / Security Receive commands in PCIe interface which are defined in NVM Express specification.
- RPMB
  - When SD Express card supports RPMB, card has exactly one RPMB Unit. In SD interface, host accesses RPMB Unit by ACMD53/54. In PCIe interface, host uses Security Send / Security Receive commands defined in NVM Express specification.

8.2 The PCIe Electrical Interface

The electrical definitions of the PCIe interface may be found in PCI-SIG specs (refer to Appendix A.2.1).

8.2.1 SD Express Interface Signals

Table 8-2 describes the SD Express card interface signals

Signal Group	Signal	I/O/S	Description	Voltage
Power	VDD	S	3.3V source	3.3V
	VDD1, VDD1a (2)			
	VDD2, VDD2a (3)	S	1.8V source	1.8V
	VDD3	S	1.2V source (optional by card and by host)	1.2V
	VSS	S		0V
PCIe Interface	PCIeTX+, PCIeTX- PCIeTX0+, PCIeTX0- PCIeTX1+, PCIeTX1- (4)	I	PCIe TX Differential signals defined by the PCI Express Card Electromechanical Specification.	
	PCIeRX+, PCIeRX PCIeRX0+, PCIeRX0- PCIeRX1+, PCIeRX1- (4)	O	PCIe RX Differential signals defined by the PCI Express Card Electromechanical Specification.	
	REFCLK+,REFCLK- (shared with DAT0/DAT1 of SD interface)	I	PCIe Reference Clock signals (100 MHz) defined by the PCI Express Card Electromechanical Specification.	
	PERST# (shared with DAT3 of SD interface)	I	PCIe Reset is a functional reset to the card as defined by the PCI Express Mini Card Electromechanical Specification. Refer to Section 3.17.2 for PCIe training sequence. Active low.	3.3V/1.8V

	CLKREQ# (shared with DAT2 of SD interface)	I/O (1)	PCIe Request is a reference clock request signal and also used for L1 Power Modes substates as defined by the PCI Express Mini Card Electromechanical Specification. Refer to Section 3.17.2 for PCIe training sequence. Clock Request signal is driven by card as active low.	3.3V/1.8V
SD Interface	DAT0-DAT3 (DAT0/1 are shared with REFCLK+/- of PCIe, DAT2 and DAT3 are shared with CLKREQ# and PERST# of PCIe respectively)	I/O	Data lines	3.3V/1.8V
	CMD	I/O	Command/Response	3.3V/1.8V
	CLK	I	SD Clock Input	3.3V/1.8V

Note (1) : The CLKREQ# I/O is an Open Drain gate as defined in PCI-SIG
(2) : G3L2, G4L1 and G4L2 SD Express Cards shall have additional 3.3V pad #19 (VDD1a).
(3) : G3L2 and G4L2 SD Express Cards shall have additional 1.8V pad #24 (VDD2a).
(4) : These are applied to 2-Lane SD Express Card.

Table 8-2 : SD Express Card Interface Signals

8.2.2 Differential Voltage Swing

PCIe Standard allows full swing operation and half swing.
SD Express card shall support full swing operation mode (800mV).

8.2.3 REFCLK Specification

REFCLK shall comply with the REFCLK specification defined in PCI-SIG Base Specification for 8.0GT/s.

Symbol	Description	Limits		Units
		Min	Max	
F_{REFCLK}	Refclk frequency	99.97	100.03	MHz
$T_{REFCLK-RMS-CC}$	RMS Refclk jitter for Independent SSC Separate Refclk architecture		1.0	ps RMS
F_{SSC}	SSC frequency range	30	33	kHz
$T_{SSC-FREQ-DEVIATION}$	SSC deviation		+0.0/-0.5	%
$T_{TRANSPORT-DELAY}$	Tx-Rx transport delay	12		ns

Table 8-3 : Parameters for REFCLK Common Rx Architecture defined for 8.0GT/s

For further details, refer to PCI-SIG specification.

8.2.4 CLKREQ# and PERST# Electrical Definition

The auxiliary signaling of the PCIe interface shall operate at 1.8v or 3.3V logic signaling. Host may use 1.8V or 3.3V logic signaling. Card shall be able to receive both 1.8V and 3.3V signaling. Due to the share of these signals with the legacy SD interface signals, it is the responsibility of the card and the host to operate these signals at the levels defined. Refer to Appendix H.1 for additional description of the IO characteristics.

8.2.5 AC Coupling Capacitors – Placement

AC coupling capacitors for PCIe Gen 3 is defined 176nF to 265nF. The PCIe (M.2) specification defines to mount the coupling capacitors on the TX side, in case of removable card.

Due to limited space in microSD cards this specification defines for PCIe interface of the SD Express card that the coupling capacitors of TX side of SD Express card shall be on the host side as close as possible to the connector. The placement of the AC coupling capacitors shall be up to 12.5mm from the connector's SD contact pads.

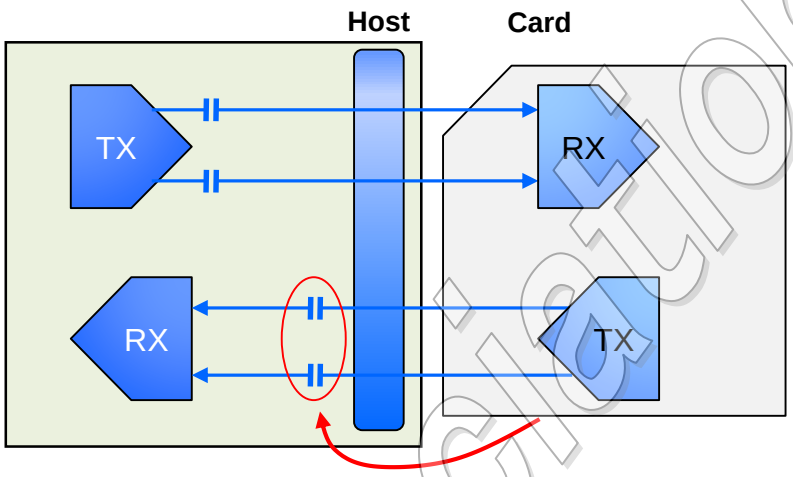


Figure 8-1 : Placement of AC coupling capacitors

8.2.6 PCIe TX/RX Phy protection in Standard Size SD card

This section is a blank in the Simplified Specification

8.2.7 PCIe Power Rating of SD Express card

Table 8-4 below summarizes Power Rating requirements for the SD Express (PCIe Gen3x1) power rail

Power Rail	Voltage Tolerance	Current Consumption limit	
		Peak mA Max Avg @ 25 µs	Normal mA Max Avg @ 1 s
3.3V	2.7-3.6V	600	400
1.8V	1.7-1.95V	600	400
1.2V*	1.14-1.3V	600	400

(*) 1.2V supply is optional
(**) Only one of 1.2V or 1.8V power rails is used

Table 8-4 : Power Rating Table for SD Express (PCIe Gen3x1)

Table 8-5 below summarizes Power Rating requirements for the SD Express (PCIe Gen3x2 or Gen4x1) power rail.

Power Rail	Voltage Tolerance	Total Current Consumption limit	
		Peak mA Max Avg @ 25 µs	Normal mA Max Avg @ 1 s
3.3V	2.7-3.6V	1,000	700
1.8V	1.7-1.95V	600	400
1.2V*	1.14-1.3V	600	400

(*) 1.2V supply is optional
(**) Only one of 1.2V or 1.8V power rails is used
(***) 3.3V power is supplied by both VDD1 and VDD1a pads

(****) 1.8V power is supplied by both VDD2 and VDD2a pads in case of PCIe Gen3x2 mode

Table 8-5 : Power Rating Table for SD Express (PCIe Gen3x2 or Gen4x1)

Table 8-6 below summarizes Power Rating requirements for the SD Express (PCIe Gen4x2) power rail.

Power Rail	Voltage Tolerance	Total Current Consumption limit	
		Peak mA Max Avg @ 25 µs	Normal mA Max Avg @ 1 s
3.3V	2.7-3.6V	1,300	900
1.8V	1.7-1.95V	900	600
1.2V*	1.14-1.3V	900	600

(*) 1.2V supply is optional
(**) Only one of 1.2V or 1.8V power rails is used
(***) 3.3V power is supplied by both VDD1 and VDD1a pads
(****) 1.8V power is supplied by both VDD2 and VDD2a pads

Table 8-6 : Power Rating Table for SD Express Card (PCIe Gen4x2)

8.3 Initialization Process of SD Express Card

Following sections provide detailed description of the SD Express interface signals during the interface detection and initialization process (refer to Section 3.17.2 for the flow chart of the process and the card's states diagram).

8.3.1 Overview

There are two options for SD Express initialization, one is starting with issuing SD commands, the other is without issuing SD commands. SD Express cards shall support both types of initialization. On the other hand, SD Express Host can execute either type of initialization, but starting with issuing SD commands is highly recommended because it can detect card type safely and properly by SD CMD8, and avoid potential interoperability issues as a result.
When host executes Fast Boot, it needs to activate SD interface before initializing PCIe interface, because Fast Boot is available only in SD interface.

8.3.2 SD Express Initialization Starting with Issuing SD Commands

This section is a blank in the Simplified Specification

8.3.3 SD Express Initialization without Issuing SD Commands

This section is a blank in the Simplified Specification

8.3.4 SD Express Initialization without Issuing SD Commands But Unsuccessful

This section is a blank in the Simplified Specification

8.3.5 SD Express Initialization with Fast Boot

This section is a blank in the Simplified Specification

8.4 Detailed Specifications of SD Express Speed Class

8.4.1 Introduction

8.4.1.1 Overview

SD Express Speed Class is defined over PCIe bus of SD Express Card. Since this specification is constructed based on conventional Speed Classes, it is necessary to refer to Section 4.13. In addition, some NVMe specifications are referred for it such as Streams Directive, Dataset Management (abbreviated as DSM), and so on.

SGS and SWS defined in Streams Directive corelate to AU size and RU size respectively (refer to Section 8.4.2.1 for more details). In addition, Speed Class functions operated by CMD20 are mapped to the following 6 commands operated by the DSM command (refer to Section 8.4.6 for more details).

- Start Recording
- Update DIR/CI
- Suspend AU/SGS
- Resume AU/SGS
- Set Free AU/SGS
- Release DIR/CI

Class 150, 300, 450 and 600 are newly introduced for SD Express Speed Class. Class 150 means stream recording speed at 150MB/s or faster is guaranteed. Different from the conventional Speed Classes, up to 8 streams can be recorded simultaneously in real-time to the SD Express card supporting SD Express Speed Class.

Although the following functions are optional in the NVMe spec., the SD Express card shall support all of these functions if it supports SD Express Speed Class.

- Flush command (if a volatile write cache is present in the card)
- Dataset Management
- Streams Directive
- Host Controlled Thermal Management

For explanatory convenience, the following abbreviations are used in this specification though they are not abbreviated in the NVMe specification.

- CMP: Completion
- ICDS: Identify Controller Data Structure
- NCQ: I/O Completion Queue for NVMe
- NSQ: I/O Submission Queue for NVMe
- SID: Stream Identifier

8.4.1.2 Area Assignment

Figure 8-13 illustrates an example of area assignment for the SD Express Speed Class.

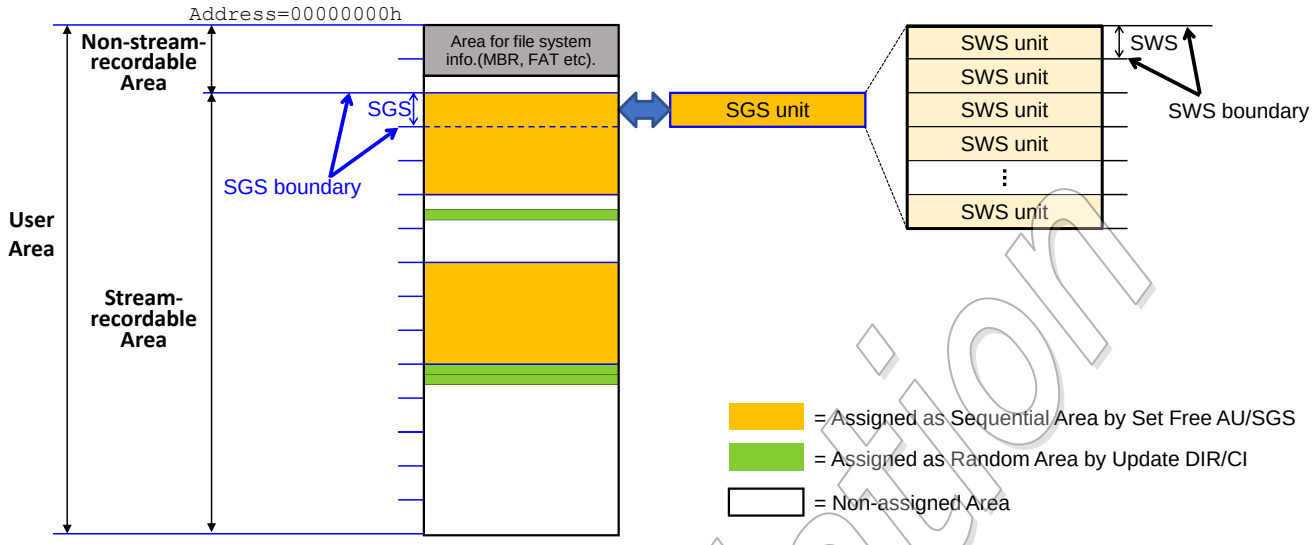


Figure 8-13 : An Example of Area Assignment for SD Express Speed Class

The User Area is divided by an SGS boundary which is a physical boundary of the User Area and whose address is a multiple of SGS (see Section 8.4.2.1.2). The leading part of the User Area starting from Address=00000000h is set aside for recording file system information such as Master Boot Record (MBR), Partition Table or FAT (File Allocation Table). A Stream-recordable Area starts from the next SGS boundary after the end of the area for file system information, and terminates at the end of the User Area. The rest part of the User Area except the Stream-recordable Area is called Non-stream-recordable Area and should not be used for stream recording.

8.4.1.2.1 Sequential Area

The stream data can be recorded in the Sequential Area assigned by “Set Free AU/SGS” command (see Section 8.4.6.6) on the Stream-recordable Area. The start address of each Sequential Area shall be aligned to the SGS boundary and the length of this area shall be a multiple of SGS.
Each Sequential Area consists of one or multiple SGS units, and each SGS unit is divided into the specific number of SWS units as shown in Figure 8-13. Each SWS unit shall be aligned to the SWS boundary (whose address is a multiple of SWS defined in Section 8.4.2.1.1) and its size is identical to SWS.
Let SWC denote Sequential Write Command and define as a Write command for stream recording. In the SWC, the Start LBA (SLBA) shall be aligned to the SWS boundary in the Sequential Area, and the Number of Logical Blocks (NLB) shall be a multiple of SWS except when recording the end part of file, and less than or equal to MDTS (see Section 8.4.2.1.4). During Speed Class Recording, when card receives an SWC whose NLB is not a multiple of SWS, card should pad left out area of the last SWS unit because this means file recording termination.
Refer to Section 8.4.2.1.1 and 8.4.2.1.2 for more details.

8.4.1.2.2 Random Area

During the Speed Class Recording, not only stream data but also its associated data such as DIR or CI need to be written. The size of these data is comparatively small and they are recorded in the Random Area suitable for recording such kind of data. The Random Area is assigned by “Update DIR/CI” command (see Section 8.4.6.3) on the User Area.
Start address of the Random Area shall be aligned to 16KB in the User Area, and its size is fixed to 16KB. The NLB in the Write command for this area will be between 512 bytes to 16KB. The maximum number of Random Area in a card is min(MSL, 8) x 2, where MSL is Max Stream Limit (see Section 8.4.2.1.3). It is recommended to assign Random Areas in the same SGS unit in order to minimize the number of SGS

units for Random Area.
The area assigned to neither Random Area nor Sequential Area is called as Non-assigned Area.

8.4.1.3 Supporting Multiple Streams Access

SD Express Speed Class defines multiple streams access, which allows two or more streams for recording and/or reading simultaneously. The maximum number of recordable streams is indicated in MSL field of the Streams Directive Return Parameters Data Structure (see 8.4.2.1.3). SD Express card supporting SD Express Speed Class shall set 2 or more to its MSL field.
Note that even MSL value is more than 8, host can activate up to 8 recording streams at the same time during the Speed Class Recording State (see Figure 8-17).
Card shall guarantee the sum of accumulated recording speed of valid streams indicated in the List of Open SID for Speed Class (see Section 8.4.10.3.2) is higher than or equal to its supported speed class.
Also refer to Section 8.4.3 for speed class measurement.

8.4.1.4 Basic Sequence of SD Express Speed Class

Figure 8-14 shows an example sequence of SD Express Speed Class. In this example, 2 streams called S1 and S2 are recorded in parallel.

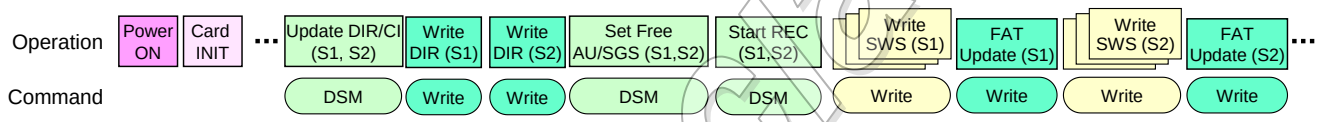


Figure 8-14 : An Example Sequence of SD Express Speed Class

After card initialization, host issues “Update DIR/CI” command and allocates 4 Random Areas for storing DIR and CI for each stream in this example. Then, host writes DIR for S1 and S2 to the area allocated by the “Update DIR/CI” command (see Section 8.4.6.3).
Before starting the stream recording, host issues “Set Free AU/SGS” command (see Section 8.4.6.6) to assign the Sequential Area in the Stream-recordable Area, and then the “Start Recording” command (see Section 8.4.6.2) to notify the start of recording and request its preparation to the card.
After that, host issues one or more Stream Write command (SWC) using NVMe Write command with DTYPE of Dword12 and DSPEC, DSM fields of Dword13 defined for recording S1. Host also needs to issue Random Write commands using NVMe Write command with DTYPE of Dword12 and DSPEC, DSM fields of Dword13 defined for updating the file system management information including FAT, Allocation Bitmap, DIR and CI. This operation is simply called as “FAT Update.” Similarly, writing data for S2 takes place as mentioned above.
Table 8-7 shows essential parameters for NVMe Write commands.

Field	Bit	Description	Stream Write	Random Write
Dword 10 and 11	63:00	Starting LBA (SLBA)	Start Address of stream data	Start Address of random data
Dword 12	31	Limited Retry (LR)	1(1)	1(1)
	30	Force Unit Access (FUA)	0(2)	1 or 0(3)
	29:26	Protection Information Field (PRINFO)	0000b	0000b
	25:24	Reserved	Reserved	Reserved
	23:20	Directive Type (DTYPE)	0001b	0000b
	19:16	Reserved	Reserved	Reserved
	15:00	Number of Logical Blocks (NLB)	A multiple of SWS(4)	Data size
Dword 13	31:16	Directive Specific (DSPEC)	SID	00h
	15:08	Reserved	Reserved	Reserved
	07	Incompressible	0 or 1(5)	0 or 1(5)
	06	Sequential Request	1	0
	05:04	Access Latency	00b	11b
	03:00	Access Frequency	0010b	0101b

Note (1): If LR=0, card does not guarantee Speed Class Recording.
Note (2): Host shall set FUA=0 to assure speed class performance. Otherwise, card may not guarantee the recording speed.
Note (3): Host shall set FUA=1 for writing the file system management information including FAT, Allocation Bitmap, DIR and CI. Host may set either 0 or 1 as FUA to write other data to the Random Area assigned by “Update DIR/CI”.
Note (4): NLB shall be less than or equal to MDTS (refer to Section 8.4.2.1.4). NLB may not be a multiple of SWS when host writes stream data at the end of the file.
Note (5): Card behavior for read/write shall be the same regardless of Incompressible value.

Table 8-7 : Essential Parameters of NVMe Write Commands for SD Express Speed Class

8.4.1.5 Suspend and Resume

Similar to Video Speed Class, Suspend and Resume functions are available in SD Express Speed Class as well. Host can suspend one or multiple stream recordings in the midst of recording to SGS unit by issuing “Suspend AU/SGS” command (see Section 8.4.6.4). Then, host is necessary to execute Release Identifier operated by Directive Send (see also Section 8.4.1.6). Note that the suspension address specified by the host shall be aligned to SWS boundary.

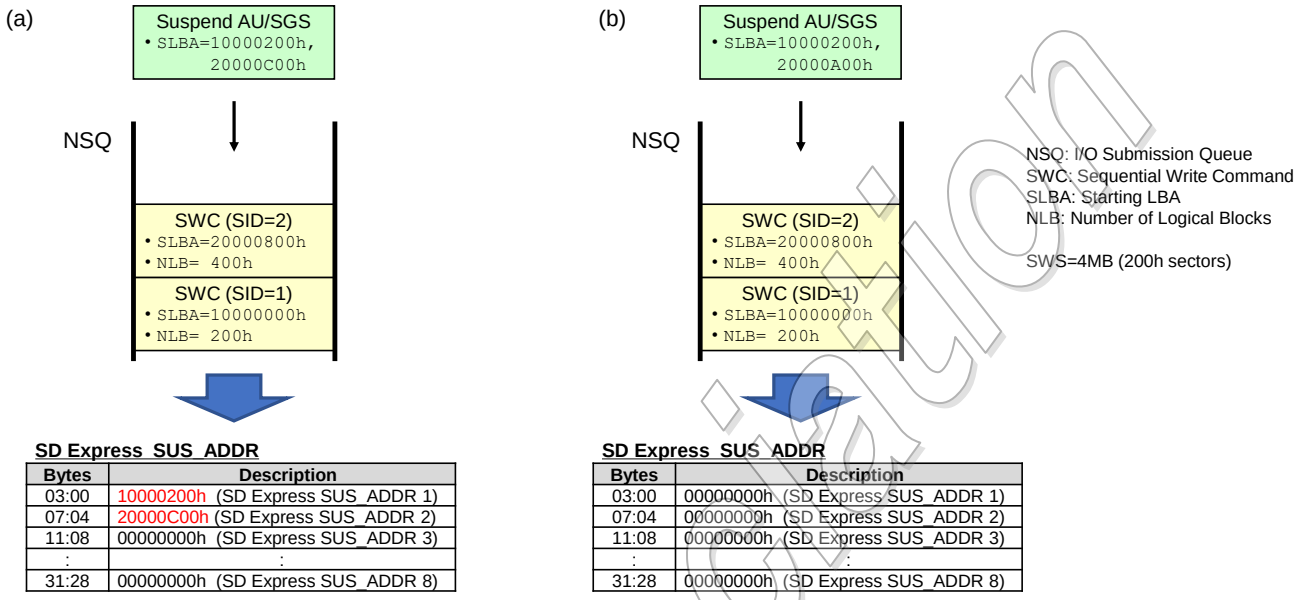
Host can restart the stream recording from the suspension address by the “Resume AU/SGS” command (see Section 8.4.6.5) to utilize the rest part of the SGS units. The suspension addresses are obtained by issuing Get Log Page command (see Section 8.4.10.3.3). New SID is assigned when host issues SWC whose SLBA is included in the Resume AU/SGS.

The rules of “Suspend AU/SGS” command are as follows.

Host can set one or more suspension addresses in the “Suspend AU/SGS” command by calculating “Starting LBA (SLBA) + Number of Logical Blocks (NLB)” in the latest SWC for each stream. Such an address calculated by the formula above is called a valid suspension address. An invalid suspension address means an address not satisfying the formula. Note that when host executes Speed Class Recording for multiple streams, host can suspend either all streams or a part of them.

If the “Suspend AU/SGS” command includes at least one invalid suspension address, card rejects this command, posts its CMP to the NCQ indicating 80h (Conflicting Attributes) in the status value, does not add it to the list of SD Express SUS_ADDR obtained by Get Log Page command, and continues speed class recording as if the command had not been issued. Note that when host specifies an SGS boundary as a suspension address, card handles it similar to invalid suspension address. This is because specifying an SGS boundary as a suspension address is meaningless considering the background that Suspend function is introduced to resume the stream recording from a suspension address “in the middle of” an SGS unit for efficient use of it.

Figure 8-15 shows examples of card operation when the “Suspend AU/SGS” command is received. Suppose SWS is 4MB (200h sectors) and all of SD Express SUS_ADDR indicate zero before issuing the “Suspend AU/SGS” command in these examples.



SD Express_SUS_ADDR

Bytes	Description
03:00	10000200h (SD Express SUS_ADDR 1)
07:04	20000C00h (SD Express SUS_ADDR 2)
11:08	00000000h (SD Express SUS_ADDR 3)
:	:
31:28	00000000h (SD Express SUS_ADDR 8)

SD Express_SUS_ADDR

Bytes	Description
03:00	00000000h (SD Express SUS_ADDR 1)
07:04	00000000h (SD Express SUS_ADDR 2)
11:08	00000000h (SD Express SUS_ADDR 3)
:	:
31:28	00000000h (SD Express SUS_ADDR 8)

Figure 8-15 : Examples of Suspend Operation

In Example (a), two SWCs identified by SID=1 and SID=2 respectively have been posted in the NSQ as shown in the figure. Then, host posts a “Suspend AU/SGS” command to suspend these streams at address 10000200h and 20000C00h. Since SLBA and NLB of the latest SWC for SID1 are 10000000h and 200h respectively, the correct suspension address for it is calculated as 10000000h + 200h = 10000200h. Similarly, that for SID2 is 20000800h + 400h = 20000C00h. In this example, since both SLBAs in the “Suspend AU/SGS” command completely match to the correct suspension addresses, card accepts this suspension command and indicates 10000200h and 20000C00h in the list of SD Express SUS_ADDR as valid suspension addresses.
In Example (b), on the other hand, same SWCs have been posted in the NSQ. Then, host posts a “Suspend AU/SGS” command whose SLBAs are 10000200h and 20000A00h. In this case, the former SLBA (10000200h) matches to the correct suspension address, but the latter one (20000A00h) does not coincide with any suspension addresses (in other words, it is invalid). Therefore, card rejects this suspension command, responds CMP with Conflicting Attributes, and keeps the list of SD Express SUS_ADDR as it is.
Refer to Section 8.4.6.4 and 8.4.6.5 for more details about the “Suspend AU/SGS” and “Resume AU/SGS” commands respectively.

> Application Note:
> Host should specify the suspension address which is aligned to both an SWS boundary and a Cluster boundary (whose address is a multiple of the cluster size).

8.4.1.6 Access Rules for Multiple Streams Recording

The following rules are introduced during Speed Class Recording State for writing one or multiple streams data in real-time.

- Each stream shall be recorded from the top of an SGS unit continuously, and the size of writing data is a multiple of SWS except when recording the end part of a file.

- When the SGS unit is fully recorded, host can continue the stream recording from the top of another SGS unit. These SGS units are not necessary to be continuous.
- Only data for one stream specified by the Stream Identifier (SID) can be stored in one SGS unit except when “Suspend and Resume” took place.
- Host can suspend one or more stream recordings by the “Suspend AU/SGS” command. The suspension addresses shall be the multiple of SWS (see Section 8.4.1.5). This means if host terminates stream file recording at the address other than the SWS boundary, host cannot specify it as a suspension address.
- After issuing “Resume AU/SGS” command, host may start stream recording from one of SLBAs specified by this command.

Figure 8-16 illustrates an example of stream recording along the access rules above.

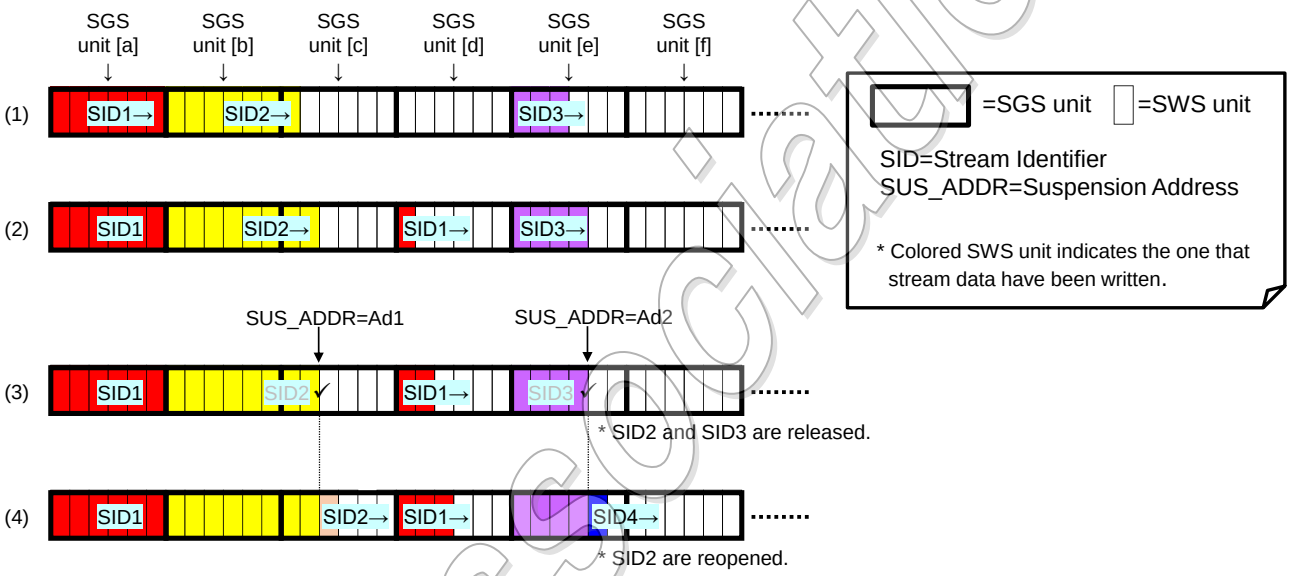


Figure 8-16 : An Example of Multiple Stream Recording

(1) A stream whose Stream Identifier is SID1 is recorded continuously from the top of SGS unit [a]. Similarly, stream data by SID2 and SID3 are being stored in SGS unit [b] and [e] respectively.
(2) As the SGS unit [a] is fully occupied by SID1 stream, host continues to record it from the top of another vacant SGS unit. In this example, since the data has been already recorded in SGS unit [b] subsequent to SGS unit [a], host selects a discrete SGS unit which is fully vacant (SGS unit [d], in this case).
(3) When host issues “Suspend AU/SGS” command to suspend stream recordings specified by SID2 and SID3, they are suspended at the address Ad1 and Ad2 respectively. When host issues Get Log Page for finding SD Express SUS_ADDR, these Ad1 and Ad2 are indicated as suspension addresses. Then, host needs to release SID2 and SID3 by the Release Identifier operation.
(4) After host issues “Resume AU/SGS” command, host gets the suspension addresses by Get Log Page command and starts stream recording by SID2 from address Ad1. Note that the stream operated by SID2 in (4) is irrelevant that by SID2 in (1) through (3) because SID2 was released in (3). Similarly, host may initiate stream recording by SID4 from Address Ad2.

8.4.1.7 State Machine of the SD Express Speed Class

Figure 8-17 shows a state machine of the SD Express Speed Class.

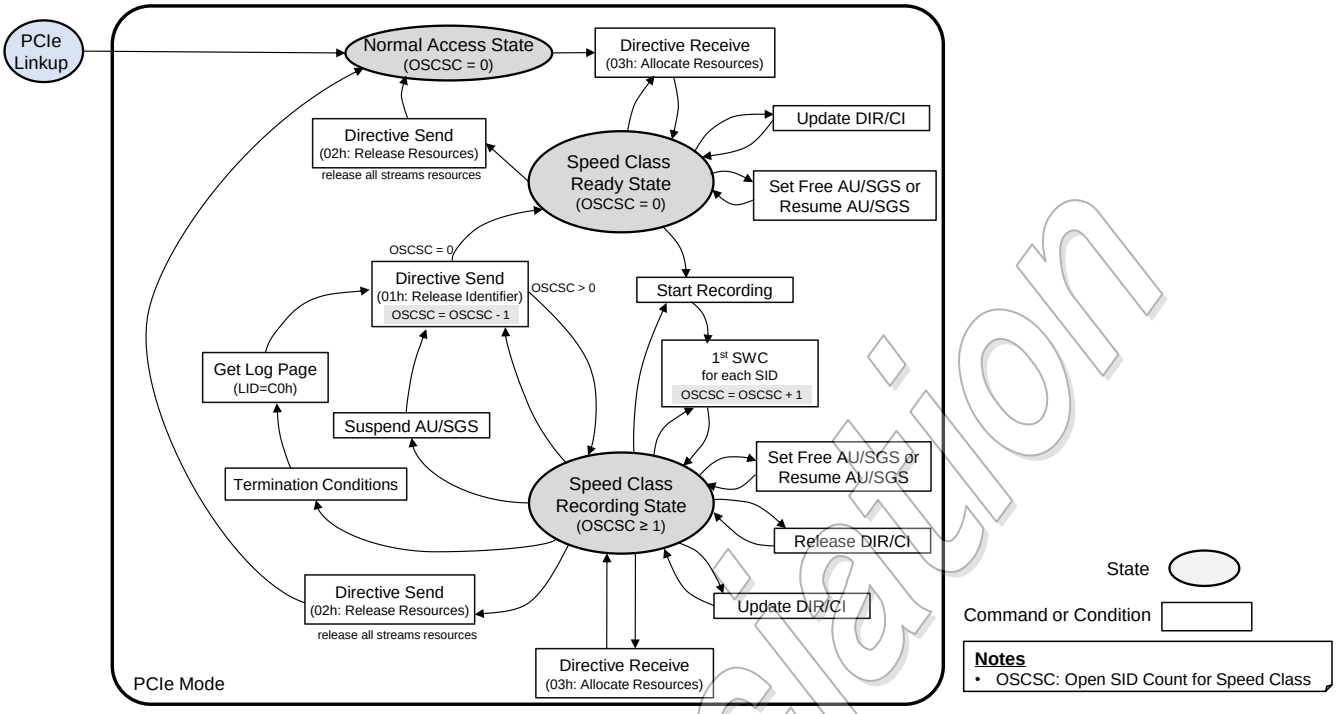


Figure 8-17 : State Machine for SD Express Speed Class

"PCIe Linkup" and "PCIe Mode" in this figure are same as those of Figure 3-23. States for SD Express Speed Class is defined as substates of PCIe Mode.
When transiting from PCIe Linkup to PCIe mode, the state is Normal Access State. It is defined as a state that Speed Class Recording is unavailable because resources for it have not been allocated yet.
When host issues Directive Receive command whose operation name is Allocate Resources, the state transits to Speed Class Ready State. In this state, Speed Class Recording gets available but no streams are recorded based on Speed Class Recording.
After allocating Sequential Areas and Random Areas by "Set Free AU/SGS" and "Update DIR/CI" respectively, host executes "Start Recording" to inform card of that Speed Class Recording will start. Then host issues the SWC indicating a specific SID to start Speed Class Recording designated by the SID. When the SLBA in the first SWC for each SID matches to the one designated by "Start Recording," card registers the associated SID as an open SID for Speed Class (OSID-SC), and increases the Open SID Count for Speed Class (OSCSC) value. As a result, the state transits to Speed Class Recording State and the Speed Class Recording designated by the SID starts successfully.
Note that if the SLBA in the first SWC is not included in the "Start Recording", the recording speed is not guaranteed for the associated SID, and OSCSC is not increased.
If "Start Recording" specifies multiple SLBAs, host can continue to issue the first SWC for some SID to start one stream recording from one of SLBAs specified by the "Start Recording". For example, if the "Start Recording" specifies Ad1, Ad2 and Ad3, host issues the first SWC with SID1 starting from Ad1, that with SID2 from Ad2, and that with SID3 from Ad3 repeatedly.
Since "Start Recording" can specify multiple SLBAs, host can issue an SWC with other SID one by one if its SLBA includes in the "Start Recording" issued previously.
The Allocate Resources operation specifies the number of stream resources. Host does not need to allocate all resources for stream recording supporting by the card when transiting from Normal Access State. Even during Speed Class Ready State or Speed Class Recording State, host may allocate additional stream resources.
When card detects Termination Conditions (refer to Section 8.4.9), host should issue Get Log Page command and get information which OSID-SC gets unavailable. Then, host should issue Directive Send command whose operation name is Release Identifier to release the associated OSID-SC. At that time,

card decreases OSCSC value.
Similarly, when host executes “Suspend AU/SGS,” host shall operate Release Identifier to release the suspended OSID-SC.
Since the Termination Conditions may happen in the multiple streams at the same time, and host can suspend multiple streams by one “Suspend AU/SGS,” host can repeat to issue Directive Send (Release Identifier) commands to release associated multiple OSID-SCs. As a result, if OSCSC gets 0, the state transits to Speed Class Ready State because there are no valid OSID-SCs anymore.
If host issues Directive Send command whose operation name is Release Resources, all OSID-SCs are released and the state directly transits to Normal Access State.
Table 8-8 indicates the overview of each state for SD Express Speed Class. During Speed Class Ready State and Speed Class Recording State, there are some restrictions such as the number of available NSQs.
The performance of Stream Reading is guaranteed during Speed Class Ready State or Speed Class Recording State.

Item	State		
	Normal Access State	**Speed Class Ready State**	**Speed Class Recording State**
Stream Writing	Speed Class Recording is unavailable.	Speed Class Recording is available but no Speed Class Recordings are executed.	At least one Speed Class Recording is executed.
Number of Available NSQs	Up to 64K (based on NVMe spec)	1(1)	1(1)
Command Duration	Max duration is not defined.	Defined in Section 8.4.3	Defined in Section 8.4.3
Stream Recording Rule	N/A	N/A	Compliant to Section 8.4.1.6
Stream Reading	Stream reading is possible but its performance is not guaranteed	Stream read performance is guaranteed.	Stream read performance is guaranteed.

Note (1): When host posts commands to two or more NSQs during Speed Class Ready State or Speed Class Recording State, error does not occur but performance of stream write and/or read is not guaranteed.

Table 8-8 : Overview of States for SD Express Speed Class

8.4.2 SD Express Speed Class Parameters

This section explains data defined by NVMe spec. which are closely related to the SD Express Speed Class. Refer to NVMe specifications for more details about these data and their associated ones.

8.4.2.1 Data Defined in Return Parameters Data Structure of Streams Directive

8.4.2.1.1 SWS

SWS stands for Stream Write Size and means optimal size unit for the stream data recording by a write command. In the SD Express Speed Class, card shall be set either one of 4MB, 5MB or 6MB to this parameter. Host needs to read this parameter from the Return Parameters Data Structure of Streams Directive in the card and to issue a Write command for stream data whose size is a multiple of SWS except at the end of the file (see Section 8.4.1.2.1).
Since SWS indicates a data size, it corresponds to “RU size” of the conventional Speed Class specifications. The term “SWS unit” is introduced for indicating a writing unit for SWS, and corresponds to RU of the conventional specifications.

An SWS boundary means a physical boundary of the User Area whose address is a multiple of SWS.

8.4.2.1.2 SGS

SGS stands for Stream Granularity Size and means optimal size for managing streams. In the SD Express Speed Class, card shall be set either one of values described in Table 8-9.
As well as SWS, SGS can be obtained from the Return Parameters Data Structure of Streams Directive. Since SGS is indicated by a unit of SWS in the data structure, it is a multiple of SWS as a result.
Note that card shall select an SGS value from the integer multiples of cluster size associated with the card capacity. For example, a 32TB card shall set a multiple of 8MB as its SGS value because its cluster size is 8MB (16384 sectors). This means it cannot adopt values indivisible by 8MB such as 150MB, 180MB, and so on. Refer to Appendix C.5.3 in the Part 2 File System Specification.

Available SGS Values (in MB)
120 / 150 / 180 / 210 / 240 / 270 / 300 / 330 / 360 / 390 / 420 / 450 / 480 / 510 / 540 / 570 / 600 / 630 / 660 / 690 / 720 / 750 / 780 / 810 / 840 / 870 / 900 / 930 / 960 / 990 / 1,020 / 1,080 / 1,200 / 1,320 / 1,440 / 1,560 / 1,680 / 1,800 / 1,920 / 2,040 / 2,160 / 2,280 / 2,400 / 2,520 / 2,640 / 2,760 / 2,880 / 3,000 / 3,120 / 3,240 / 3,360 / 3,480 / 3,600 / 3,720 / 3,840 / 3,960 / 4,080

Table 8-9 : Available SGS Values for SD Express Speed Class

Since SGS indicates a data size, it corresponds to “AU size” of the conventional Speed Class specifications. The term “SGS unit” is introduced for indicating a managing unit whose start address is aligned to the SGS boundary and whose size is identical to SGS. SGS unit corresponds to AU of the conventional specifications.

8.4.2.1.3 MSL

MSL stands for Max Streams Limit and indicates the maximum number of concurrently open streams for recording. In the SD Express card supporting SD Express Speed Class, this field shall be set a value greater than or equal to 2. It is allowed to set the value more than 8, but up to 8 streams can be recorded in parallel over the SD Express Speed Class specification.

8.4.2.1.4 MDTS

MDTS stands for Maximum Data Transfer Size. SD Express card supporting SD Express Speed Class shall indicate the value larger than or equal to SWS.

8.4.2.2 Parameters Defined in Identify Controller Data Structure

8.4.2.2.1 VWC

VWC stands for Volatile Write Cache and indicates attributes related a volatile write cache in the card. If Bit 0 of VWC is 1, host should enable VWC by the Set Features command (Feature Identifier 06h) with WCE (Volatile Write Cache Enable) =1.

8.4.2.2.2 PSD

PSD stands for Power State Descriptor and indicates the characteristics of each Power State including MP (Maximum Power). Card can set at most 32 PSDs. In the NVMe spec, each PSD is defined as PSD0, PSD1, ..., and PSD31. SD Express card shall support Power States as described in Section 8.1.8. PSD0 indicates the largest maximum power of all supported PSDs.

8.4.2.2.3 MNTMT and MXTMT

MNTMT stands for Minimum Thermal Management Temperatures and indicates the minimum temperature, in degrees Kelvin, of the specific parameter TMT1 set by Set Features command.
Similarly, MXTMT stands for Maximum Thermal Management Temperatures and indicates the maximum temperature of TMT2.

8.4.2.2.4 WCTEMP

WCTEMP stands for Warning Composite Temperature Threshold and indicates the minimum composite temperature that indicates an overheating condition. Card shall be designed in order that its composite temperature does not exceed WCTEMP for maintaining the Speed Class Recording.

8.4.3 Measurement Conditions for SD Express Speed Class

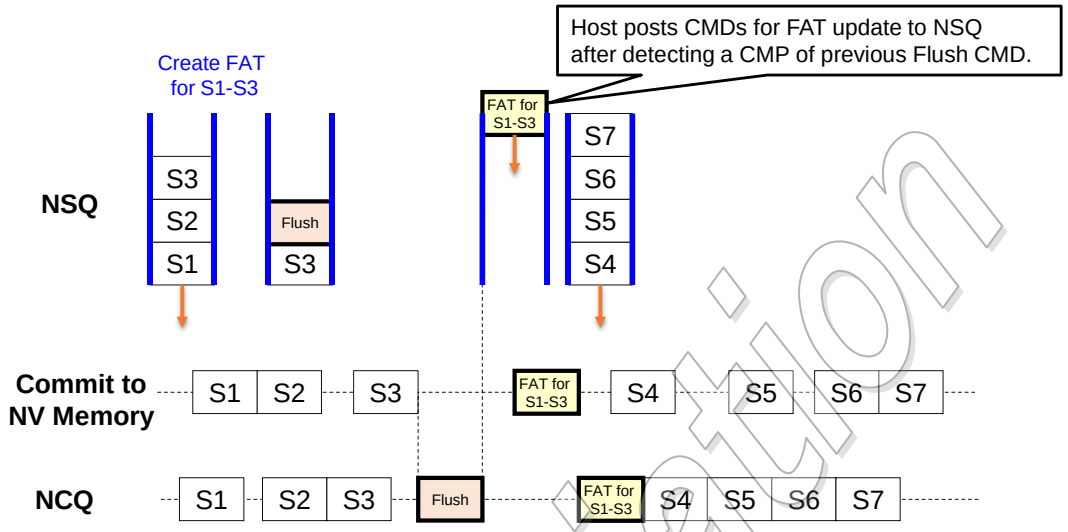
8.4.3.1 Queue Specification

According to the NVMe specification, host can create up to 65,535 NSQs at the same time. However, only one NSQ is available during Speed Class Ready State and Speed Class Recording State. In other words, host can post NVMe commands (Read, Write, DSM and so on) to one specified NSQ at that time. NVMe specification has no ordering restrictions for processing of commands within or across NSQs. Therefore, when volatile write cache is enabled (see Section 8.4.2.2.1), host is recommended to issue Flush command before writing random data such as FAT or DIR if host would like to maintain the consistency between stream data and its file system management information even in shutdown.

Application Note:
To keep consistency among file system management information even in abrupt shutdown, committing random data to non-volatile memory never precedes that of corresponding stream data in the card. If host needs to ensure that consistency, it is required to issue Flush command in advance, and post Random Write commands for file system management information such as FAT to the NSQ after detecting the CMP of the Flush command. Also refer to the following examples.

Example (1)
Figure 8-18 indicates an example of Write command operations that Host posts a command for FAT update to the NSQ after detecting the CMP of the previous Flush command. In this example, host

suspends to issue SWC for S4 and after until FAT update for S1 through S3 completes.



NSQ = I/O Submission, NCQ = I/O Completion Queue,
CMD = Command, CMP = Completion,
NV Memory = Non-volatile Memory
S1, S2, ...: Sequential Write CMD/CMP with order

Figure 8-18 : Write Command Operations for Keeping Consistency in a File (1)

Example (2)
Figure 8-19 indicates another example of Write command operations. Like Example (1), host posts a command for FAT update to the NSQ after detecting the CMP of the previous Flush command. In this example, host continues to write following stream data (S4, S5, and S6) before updating FAT for S1 through S3. In addition, this example shows out-of-order CMP posting to the NCQ.

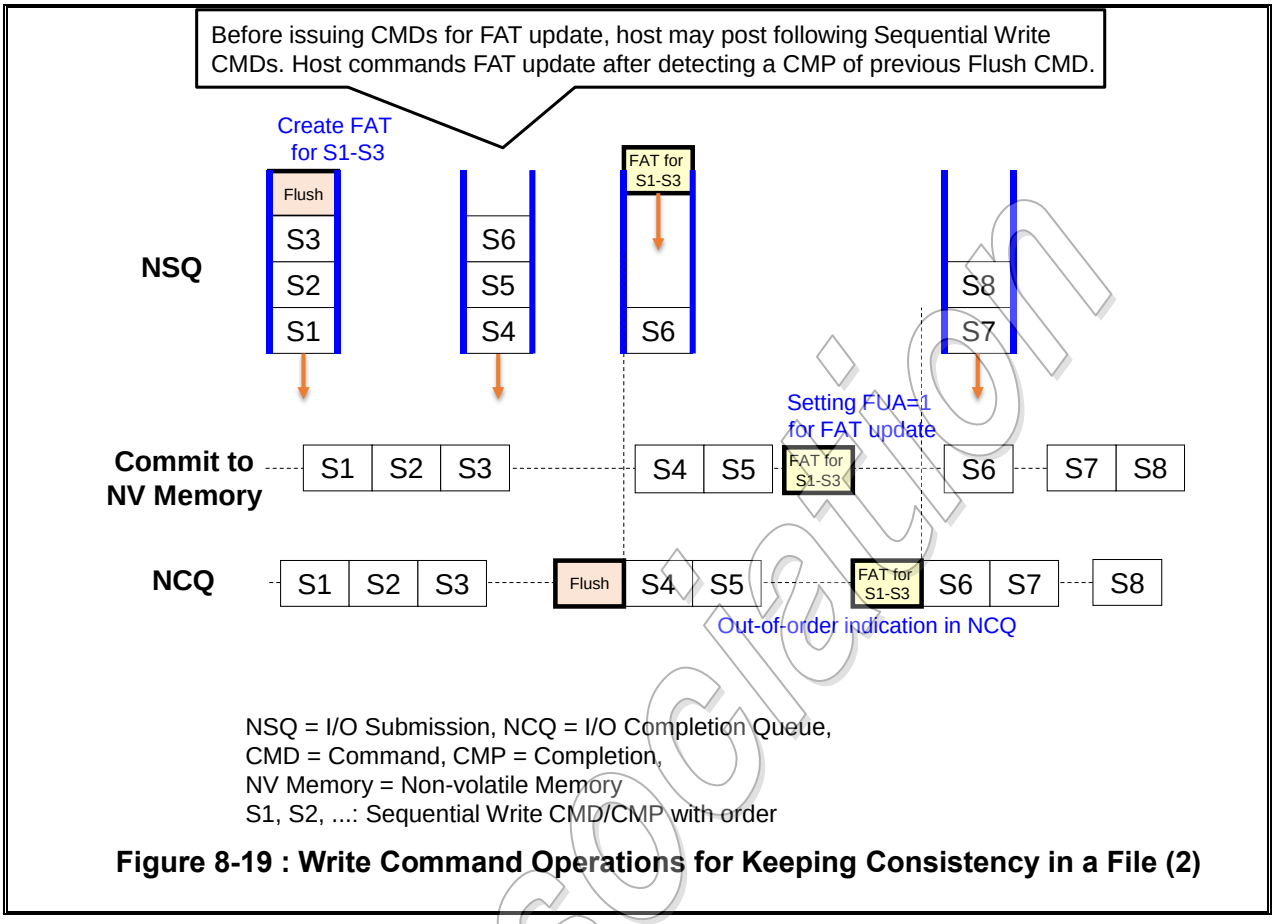


Figure 8-19 : Write Command Operations for Keeping Consistency in a File (2)

8.4.3.2 Command Duration

Command duration is a basic parameter for the performance measurement. In the SD Express Speed Class, the command duration is defined as follows (see also Figure 8-20).

(a) When NSQ is empty, it starts when the target CMD is posted to the NSQ and terminates when its CMP is detected from the NCQ.
(b) Otherwise, it starts when CMP for the previous CMD is detected from the NCQ and ends when the target CMP is detected from the NCQ.

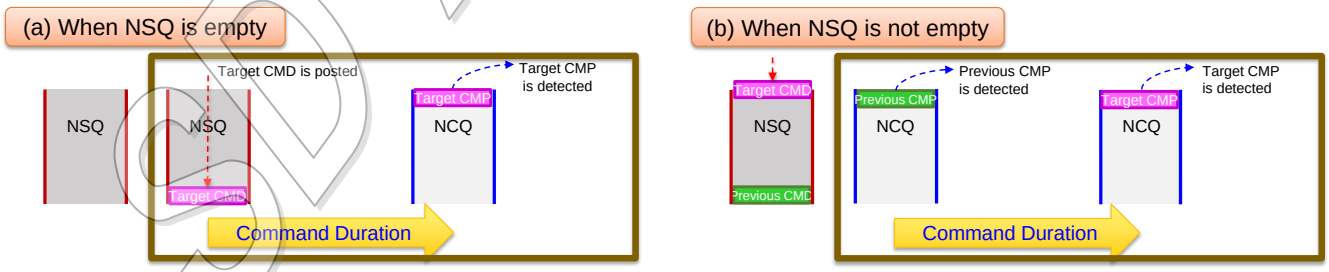


Figure 8-20 : Definition of Command Duration in the SD Express Speed Class

As described later, only case (b) is used for calculating Write Performance (Pw) and Read Performance (Pr) as shown in Figure 8-21.

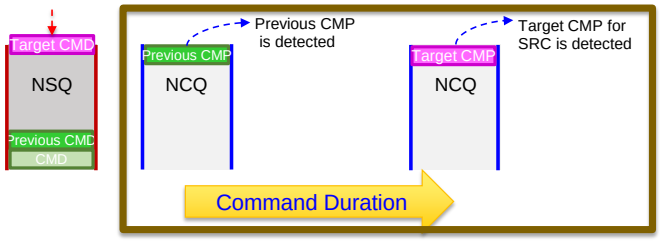


Figure 8-21 : Command Duration Used for Pw and Pr Calculation

8.4.3.3 Performance Measurement for Sequential Write Command

8.4.3.3.1 Measurement of Pw (Write Performance)

Let T'_i be the command duration of the i-th SWC posted when the NSQ is not empty. This means command durations for SWC posted when the NSQ is empty are excluded from the Write Performance calculation.

Write Performance Pw_n is calculated by the following formula:

$$Pw_n = \frac{Ld}{\sum_{i=nd+1}^{(n+1)d} T'_i} \quad (n = 0, 1, \cdots),$$

where L is the NLB in the SWC (data length written by one SWC) and d is a specific natural number dependent on SWS as described in Table 8-10. Note that Ld is always equal to 600MB for calculating Pw_n.

SWS	d
4MB	150
5MB	120
6MB	100

Table 8-10 : Definition of Parameter "d" for Calculating Pw

If all Pw_n measured in MB/s by the formula above are greater than or equal to C, the card satisfies Class C of the SD Express Speed Class (C=150, 300, 450 or 600). In other words, Class C card shall satisfy the following Pw measurement condition for all n.

$$Pw = \min(Pw_n) = \min(Pw_0, Pw_1, Pw_2, \cdots) \geq C$$

The Pw measurement condition shall be satisfied under the indicated class and all lower classes supported by the card. For example, Class 600 card shall satisfy the condition under C=150, 300, 450 and 600. Different from the Video Speed Class, Set Free AU/SGS is always excluded from the Pw measurement (refer to Section 8.4.3.5.2).

The SID in each SWC is arbitrary while measuring Pw. Moreover, considering the actual use case, host may issue the SWC whose NLB is greater than SWS value. Even in this case, Class C card shall satisfy the condition above.

Host shall post the Sequential Write commands in the ascending order of starting LBA in one SGS unit. Card may post these CMPs to NCQ out-of-order.

Application Note: When verifying Class C performance, a tester needs to provide a stream data whose bit rate is higher than C [MB/s], for example, 1.1C or 1.2C [MB/s].

8.4.3.3.2 Maximum Command Duration

The maximum command duration for SWC whose NLB indicates same as one SWS is 250 [ms]. This rule is applied to both cases described as (a) and (b) in Section 8.4.3.2.

8.4.3.4 Performance Measurement for Sequential Read Command

8.4.3.4.1 Measurement of Pr (Read Performance)

Let SRC denote Sequential Read Command and define as a Read command for stream reading. In the SRC, the Start LBA (SLBA) shall be aligned to the SWS boundary, and the Number of Logical Blocks (NLB) shall be a multiple of SWS except at the endpoint of reading.
Similar to SWC, let T'_i be the command duration of the i-th SRC posted when the NSQ is not empty.
Read Performance Pr_m is calculated by the following formula:

$$Pr_m = \frac{Ld}{\sum_{i=md+1}^{(m+1)d} T'_i} \quad (m = 0, 1, \cdots),$$

where the definitions of L and d are same as those in Section 8.4.3.3.1.
If all Pr_m measured in MB/s by the formula above are greater than or equal to C, the card satisfies Class C of the SD Express Speed Class (C=150, 300, 450 or 600). In other words, Class C card shall satisfy the following Pr measurement condition for all m.

$$Pr = \min(Pr_m) = \min(Pr_0, Pr_1, Pr_2, \cdots) \geq C$$

Considering the practical use, Pr shall satisfy the target performance C even NLB is greater than SWS.

8.4.3.4.2 Maximum Command Duration

The maximum command duration for SRC whose NLB indicates same as one SWS is 100 [ms]. This rule is applied to both cases described as (a) and (b) in Section 8.4.3.2.

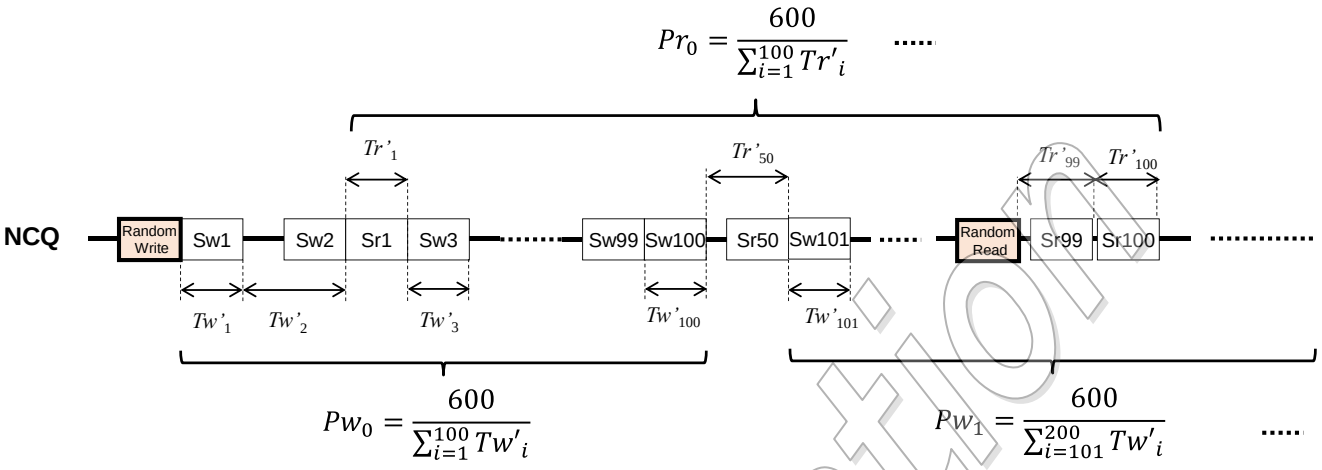
8.4.3.4.3 Note for Sequential Read

Host can set an area as Sequential Read Range by setting SR=1 by the DSM command (see Table 8-12). It is not necessary to set SR=1 before host reads data, but read performance may be improved in case of SR=1.
When host intends to read data from the area recently written, host should post the read command after detecting the CMP of the write command associated with the area to be read.

8.4.3.4.4 In case that Sequential Write and Read Operations Coexist

Even when Sequential Write and Read operations are executed together, their measurement conditions are identical as described in Section 8.4.3.3 and 8.4.3.4 respectively.
Figure 8-22 shows an example of Pw and Pr measurement when SWC and SRC coexist. In this example, value d is equal to 100 since SWS=6MB.
Pw is calculated by detecting Tw'_i (the command duration of the i-th SWC posted when the NSQ is not empty) and applying to the formula described in Section 8.4.3.3.1.
Similarly, Pr is calculated by Tr'_i (the command duration of the i-th SRC posted when the NSQ is not empty) with the formula in Section 8.4.3.4.1.

SWS=6MB, d=100



NCQ = I/O Completion Queue,
Sw1, Sw2, ...: CMP of SWC with order
Sr1, Sr2, ...: CMP of SRC with order
Tw'_1, Tw'_2, ...: Command duration for SWC corresponding to Sw1, Sw2, ...
Tr'_1, Tr'_2, ...: Command duration for SRC corresponding to Sr1, Sr2, ...

Figure 8-22 : Pw and Pr Measurement When Sequential Write and Read Operations Coexist

8.4.3.5 Performance for Random Write Command, Flush Command and DSM Commands

8.4.3.5.1 Conditions of Random Write Command, Flush Command and DSM Commands Other Than Start Recording and Set Free AU/SGS

Let T_i be the command duration of the i-th command including a Write command for updating the file system management information containing FAT, Allocation Bitmap, DIR and CI (up to 16KB), Flush command and DSM commands except “Start Recording” and “Set Free AU/SGS." Different from the calculation of Pw or Pr, T_i is measured for both case (a) and (b) in Section 8.4.3.2.
During the Speed Class Recording state, the maximum T_i (T_{max}) and the average T_i (T_{ave}) (see also Figure 8-23) shall satisfy the following inequalities respectively.

$$T_{max} \leq 100[\text{ms}]$$

$$T_{ave}(n) = \frac{1}{32}\sum_{i=n}^{n+31} T_i \leq 50[\text{ms}] \quad (n = 1,2,\cdots)$$

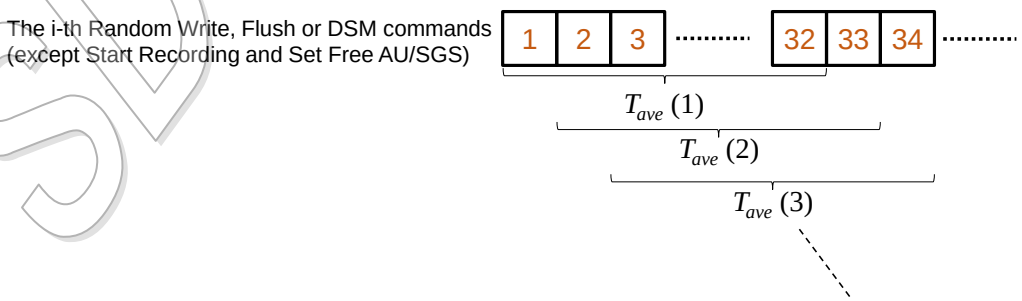


Figure 8-23 : Calculation of T_{ave} (n)

8.4.3.5.2 Conditions of Start Recording and Set Free AU/SGS

The maximum command duration for “Start Recording” command shall be 1 [sec].
The maximum command duration for “Set Free AU/SGS” command is defined as 30 x N [ms], where N is the number of SGS units to be allocated by this command.

8.4.3.6 Performance for Random Read Command

The command duration Ti for a Random Read command (whose NLB is up to 4KB) is defined as described in Figure 8-20.
The maximum command duration for this command is 20 [ms].

8.4.3.7 Frequency of Random Write and Flush

FAT Update is operated by several Random Write commands. In addition, if host would like to maintain the consistency of file system management information even in shutdown, it is recommended for host to issue Flush command before FAT update in case that a volatile write cache is enabled (see Section 8.4.2.2.1 and 8.4.3.1).
In the SD Express Speed Class, the following conditions are applied for Pw measurement.

- Random Write commands can be issued up to 6 x MSL times per 1 second.
  - The data size for these Random Write command is up to 16KB.
- A Flush command can be issued up to MSL times per 1 second.

If MSL=8, for example, host can issue up to 8 Flush commands and up to 48 Random Write commands in 1 second.

8.4.4 PCIe bus Mode Requirements for SD Express Speed Class

Table 8-11 defines the PCIe bus modes that shall be supported for each SD Express Speed Class. If card supports specific SD Express Speed Class, it shall also support all lower Speed Classes. For example, if card supports Class 600, it needs to support Class 150, 300 and 450 additionally.
For each Speed Class, moreover, card shall realize it over all supporting PCIe bus modes. As an example, if card has PCIe Gen4x1 mode and supports Class 300, card shall realize it in PCIe Gen3x1 mode as well.

SD Express Speed Class (if card supporting)	PCIe Express Bus Speed Mode (Interface Mode)			
	Gen3x1	Gen3x2	Gen4x1	Gen4x2
150 (MB/s)	M	MiS	MiS	MiS
300 (MB/s)	M	MiS	MiS	MiS
450 (MB/s)	M	MiS	MiS	MiS
600 (MB/s)	M	MiS	MiS	MiS

Note) "M": This bus mode is mandatory when card supports the corresponding SD Express Speed Class.
"Mis": This bus mode is mandatory if card supports both the corresponding SD Express Speed Class and the bus speed mode.

Table 8-11 : PCIe Bus Mode Requirements for Each SD Express Speed Class

8.4.5 Requirements of SD File System for SD Express Speed Class

Refer to 4.13.2.7.3.

8.4.6 Dataset Management Command for SD Express Speed Class

8.4.6.1 Overview

The Dataset Management (DSM) is used to specify attributes for ranges of logical blocks in the User

Area. In the SD Express Speed Class, it is used to operate some Speed Class functions which are originally defined in the Video Speed Class.
Table 8-12 indicates a configuration of the DSM command. It consists of command parameters and data associated with it. Length of the data depends on the Number of Ranges (NR) in the command parameter. Available values vary according to the type of Speed Class functions as described below.

DSM Command Parameters		
Field	**Bit**	**Description**
Dword 10	31:08	Reserved
	07:00	Number of Ranges (NR)
Dword 11	31:03	Reserved
	02	Attribute – Deallocate (AD)
	01	Attribute – Integral Dataset for Write (IDW)
	00	Attribute – Integral Dataset for Read (IDR)

Data Associated with the DSM Command				
Range	**Byte**	**Bit**		**Description**
Range 0	15:08	-		Starting LBA
	07:04	-		Length in logical blocks
	03	Context Attributes	31:24	Command Access Size
	02:01		23:11	Reserved
			10	WP: Write Prepare
			09	SW: Sequential Write Range
			08	SR: Sequential Read Range
	00		07:06	Reserved
			05:04	AL: Access Latency
			03:00	AF: Access Frequency
Range 1	15:08	-		Starting LBA
	07:04	-		Length in logical blocks
	:	:		:
:				

Table 8-12 : Command Configuration of the DSM Command

8.4.6.2 Start Recording

8.4.6.2.1 Description

This function indicates that host intends to start stream recording (see Section 8.4.1.7). This function requests the card to prepare stream recording (garbage collection, clean-up of internal status, etc.).

8.4.6.2.2 Parameter

Table 8-13 shows parameters for Start Recording.

Setting of DSM Command Parameters			
Field	**Bit**	**Description**	**Value**
Dword 10	31:08	Reserved	0
	07:00	Number of Ranges (NR)	0 to 7
Dword 11	31:03	Reserved	0
	02	Attribute – Deallocate (AD)	0
	01	Attribute – Integral Dataset for Write (IDW)	1
	00	Attribute – Integral Dataset for Read (IDR)	0

Setting of Data Associated with the DSM Command					
Range	**Byte**	**Bit**		**Description**	**Value**
Range 0	15:08	-		Starting LBA	Start Address for the 1st stream (SGS boundary or Resume Address)
	07:04	-		Length in logical blocks	If the Starting LBA of Range 0 is Resume Address, this indicates length from the Resume Address to the end of SGS unit in block unit. Otherwise, this indicates SGS value in block unit.
	03	Context Attributes	31:24	Command Access Size	0
	02:01		23:11	Reserved	0
			10	WP: Write Prepare	1
			09	SW: Sequential Write Range	1
			08	SR: Sequential Read Range	0
	00		07:06	Reserved	0
			05:04	AL: Access Latency	00b
			03:00	AF: Access Frequency	2h
Range 1	15:08	-		Starting LBA	Start Address for the 2nd stream (SGS boundary or Resume Address)
	07:04	-		Length in logical blocks	If the Starting LBA of Range 1 is Resume Address, this indicates length from the Resume Address to the end of SGS unit in block unit. Otherwise, this indicates SGS value in block unit
	:	:		:	:
:					

Table 8-13 : Parameters for Start Recording

NR field in the Start Recording indicates the number of streams that start recording by this command. If Starting LBA indicates Resume Address specified by previous “Resume AU/SGS”, the Length in logical blocks shall indicate the length from the Resume Address to the end of SGS unit. Otherwise, Starting LBA shall be aligned to SGS boundary and Length in logical blocks shall be SGS.

8.4.6.2.3 Maximum Command Duration

The maximum command duration for “Start Recording” command is 1 second (see Section 8.4.3.5.2).

8.4.6.2.4 Error Conditions

At least one of the following conditions is satisfied, card rejects this command and sets 80h (Conflicting Attributes) to the Command Specific Status Values in the corresponding CMP.

- NR is greater than 7.
- Stream resources allocated by the Directive Receive (Allocate Resources) fall short after issuing the command. Suppose host allocates 2 stream resources by the Allocate Resources operation, for instance. At this time, if host issues “Start Recording” with NR=3 or more, card responds error because resources for stream recording are short.
- At least one Start Address does not match to the SGS boundaries or Resume Addresses specified by the previous “Resume AU/SGS.”
- Illegal value is set to at least one of “Length in logical blocks” field.

8.4.6.3 Update DIR/CI

8.4.6.3.1 Description

This function allocates Random Areas for recording a DIR (directory entry) or a CI (continuous information).

8.4.6.3.2 Parameter

Table 8-14 shows parameters for Update DIR/CI.

Setting of DSM Command Parameters			
Field	**Bit**	**Description**	**Value**
Dword 10	31:08	Reserved	0
	07:00	Number of Ranges (NR)	0 to 15
Dword 11	31:03	Reserved	0
	02	Attribute – Deallocate (AD)	0
	01	Attribute – Integral Dataset for Write (IDW)	0
	00	Attribute – Integral Dataset for Read (IDR)	0

Setting of Data Associated with the DSM Command					
Range	**Byte**	**Bit**		**Description**	**Value**
Range 0	15:08	-		Starting LBA	Start Address of Range 0
	07:04	-		Length in logical blocks	32 (16KB in block unit)
	03	Context Attributes	31:24	Command Access Size	8 (4KB in block unit) [Recommended to set to 8]
	02:01		23:11	Reserved	0
			10	WP: Write Prepare	0
			09	SW: Sequential Write Range	0
			08	SR: Sequential Read Range	0
	00		07:06	Reserved	0
			05:04	AL: Access Latency	11b
			03:00	AF: Access Frequency	5h
Range 1	15:08	-		Starting LBA	Start Address of Range 1
	07:04	-		Length in logical blocks	32 (16KB in block unit)
	:	:		:	:
				:	

Table 8-14 : Parameters for Update DIR/CI

The number of random areas to be allocated is specified by NR. Starting LBA shall be aligned to 16KB in the User Area, and Length in logical blocks shall be 16KB. Command Access Size is a reference of the

access size and indicates 8 for SD Express Speed Class.

8.4.6.3.3 Maximum Command Duration

Refer to Section 8.4.3.5.1.

8.4.6.3.4 Error Conditions

At least one of the following conditions is satisfied, card rejects this command and sets 80h (Conflicting Attributes) to the Command Specific Status Values in the corresponding CMP.

- NR is greater than 15.
- The number of allocated Random Areas gets greater than min(MSL x 2, 16) after issuing the command.
- At least one Start Address is not aligned to 16KB boundary.
- Illegal value is set to at least one of “Length in logical blocks” field.
- Range(s) to be assigned by this command has been already defined as the Sequential Area.

8.4.6.4 Suspend AU/SGS

8.4.6.4.1 Description

This function directs suspension of all or a part of stream recording.

8.4.6.4.2 Parameter

Table 8-15 shows parameters for Suspend AU/SGS.

Setting of DSM Command Parameters			
Field	**Bit**	**Description**	**Value**
Dword 10	31:08	Reserved	0
	07:00	Number of Ranges (NR)	0 to 7
Dword 11	31:03	Reserved	0
	02	Attribute – Deallocate (AD)	0
	01	Attribute – Integral Dataset for Write (IDW)	1
	00	Attribute – Integral Dataset for Read (IDR)	0

Setting of Data Associated with the DSM Command					
Range	**Byte**	**Bit**		**Description**	**Value**
Range 0	15:08	-		Starting LBA	Suspension address for the 1st suspended stream
	07:04	-		Length in logical blocks	Length from the suspension address for the 1st suspended stream to the end of SGS
	03	Context Attributes	31:24	Command Access Size	0
	02:01		23:11	Reserved	0
			10	WP: Write Prepare	0
			09	SW: Sequential Write Range	1
			08	SR: Sequential Read Range	0
	00		07:06	Reserved	0
			05:04	AL: Access Latency	00b
			03:00	AF: Access Frequency	0h
Range 1	15:08	-		Starting LBA	Suspension address for the 2nd suspended stream
	07:04	-		Length in logical blocks	Length from the suspension address for the 2nd suspended stream to the end of SGS
	:	:		:	:
:					

Table 8-15 : Parameters for Suspend AU/SGS

NR field in the Suspend AU/SGS indicates the number of streams to be suspended by this command. Starting LBA shall be aligned to SWS boundary and Length in logical block shall be the multiple of SWS. After issuing this command, host shall execute Release Identifier by Directive Send command to release SIDs for suspended streams. Suspension addresses can be obtained by Get Log Page command (see Section 8.4.10.3). After suspension, the areas from each suspension address to the end of its corresponding SGS unit are not regarded as the Sequential Area. If host alters any data in the SGS unit which includes the suspension address, the suspension address is cleared by the card.

8.4.6.4.3 Maximum Command Duration

Refer to Section 8.4.3.5.1.

8.4.6.4.4 Error Conditions

At least one of the following conditions is satisfied, card rejects this command and sets 80h (Conflicting Attributes) to the Command Specific Status Values in the corresponding CMP.

- NR is greater than 7.
- At least one Suspension Address is invalid (see Section 8.4.1.5). Note that the SGS boundary is regarded as invalid address even it is always aligned to the SWS boundary.

- Illegal value is set to at least one of “Length in logical blocks” field.

Application Note: If the card did not suspend at one suspension address for any reason during the Speed Class Recording, then the card indicates a zero in the list of SD Express SUS_ADDR obtained by Get Log Page.

8.4.6.5 Resume AU/SGS

8.4.6.5.1 Description

This function directs resuming of stream recording. Host may issue this command any time after suspension, even the power cycle occurs.

8.4.6.5.2 Parameter

Table 8-16 shows parameters for Resume AU/SGS.

Setting of DSM Command Parameters			
Field	**Bit**	**Description**	**Value**
Dword 10	31:08	Reserved	0
	07:00	Number of Ranges (NR)	0 to 7
Dword 11	31:03	Reserved	0
	02	Attribute – Deallocate (AD)	0
	01	Attribute – Integral Dataset for Write (IDW)	0
	00	Attribute – Integral Dataset for Read (IDR)	0

Setting of Data Associated with the DSM Command					
Range	**Byte**	**Bit**		**Description**	**Value**
Range 0	15:08	-		Starting LBA	Suspension address for the 1st suspended stream
	07:04	-		Length in logical blocks	Length from the suspension address for the 1st suspended stream to the end of SGS
	03	Context Attributes	31:24	Command Access Size	0
	02:01		23:11	Reserved	0
			10	WP: Write Prepare	1
			09	SW: Sequential Write Range	1
			08	SR: Sequential Read Range	0
	00		07:06	Reserved	0
			05:04	AL: Access Latency	00b
			03:00	AF: Access Frequency	0h
Range 1	15:08	-		Starting LBA	Suspension address for the 2nd suspended stream
	07:04	-		Length in logical blocks	Length from the suspension address for the 2nd suspended stream to the end of SGS
	:	:		:	:
:					

Table 8-16 : Parameters for Resume AU/SGS

NR field in the Resume AU/SGS indicates the number of streams to be resumed by this command. Starting LBA shall be aligned to SWS boundary and Length in logical block shall be the multiple of SWS.

Host can issue any time when at least one suspension address is indicated in the list of SD Express SUS_ADDR obtained by Get Log Page (see Section 8.4.10.3.3). When “Resume AU/SGS” command succeeds, suspension addresses specified in the command are removed. Since the area from the suspension address to the end of SGS unit is recovered as a Sequential Area by this command, it is not necessary to issue Set Free AU/SGS command to start a stream recording from the suspension address.

8.4.6.5.3 Maximum Command Duration

Refer to Section 8.4.3.5.1.

8.4.6.5.4 Error Conditions

At least one of the following conditions is satisfied, card rejects this command and sets 80h (Conflicting Attributes) to the Command Specific Status Values in the corresponding CMP.

- NR is greater than 7.
- There are no valid suspension addresses.
- At least one suspension address in the “Starting LBA” field does not match to the one in list of SD Express SUS_ADDR obtained by Get Log Page (see Section 8.4.10.3).
- Illegal value is set to at least one of “Length in logical blocks” field.

8.4.6.6 Set Free AU/SGS

8.4.6.6.1 Description

This function allocates SGS units to record stream data.

8.4.6.6.2 Parameter

Table 8-17 shows parameters for Set Free AU/SGS.

Setting of DSM Command Parameters			
Field	**Bit**	**Description**	**Value**
Dword 10	31:08	Reserved	0
	07:00	Number of Ranges (NR)	0 to 255
Dword 11	31:03	Reserved	0
	02	Attribute – Deallocate (AD)	1
	01	Attribute – Integral Dataset for Write (IDW)	0
	00	Attribute – Integral Dataset for Read (IDR)	0

Setting of Data Associated with the DSM Command					
Range	**Byte**	**Bit**		**Description**	**Value**
Range 0	15:08	-		Starting LBA	Start Address of Range 0
	07:04	-		Length in logical blocks	Length of Range 0 in block unit
	03	Context Attributes	31:24	Command Access Size	0
	02:01		23:11	Reserved	0
			10	WP: Write Prepare	1
			09	SW: Sequential Write Range	1
			08	SR: Sequential Read Range	0
	00		07:06	Reserved	0
			05:04	AL: Access Latency	00b
			03:00	AF: Access Frequency	2h
Range 1	15:08	-		Starting LBA	Start Address of Range 1
	07:04	-		Length in logical blocks	Length of Range 1 in block unit
	:	:		:	:
:					

Table 8-17 : Parameters for Set Free AU/SGS

NR field in the Set Free AU/SGS indicates the number of ranges to be assigned as a Sequential Area by this command. Starting LBA shall be aligned to SGS boundary and Length in logical block shall be the multiple of SGS.
When the “Set Free AU/SGS” command is issued to the SGS unit in the Sequential Area, data in the unit may not be preserved (the data is undefined).

8.4.6.6.3 Maximum Command Duration

The maximum command duration for “Set Free AU/SGS” command is defined as 30 x N [ms], where N is the number of SGS units to be allocated by this command (see Section 8.4.3.5.2).

8.4.6.6.4 Error Conditions

At least one of the following conditions is satisfied, card rejects this command and sets 80h (Conflicting Attributes) to the Command Specific Status Values in the corresponding CMP.

- At least one Start Address is assigned in the Non-stream-recording Area (see Figure 8-13).
- At least one Start Address is not aligned to the SGS boundary.
- Illegal value is set to at least one of “Length in logical blocks” field.

8.4.6.7 Release DIR/CI

8.4.6.7.1 Description

This function releases random areas for recording a DIR (directory entry) or a CI (continuous information).

8.4.6.7.2 Parameter

Table 8-18 shows parameters for Release DIR/CI.

Setting of DSM Command Parameters			
Field	**Bit**	**Description**	**Value**
Dword 10	31:08	Reserved	0
	07:00	Number of Ranges (NR)	0 to 15
Dword 11	31:03	Reserved	0
	02	Attribute – Deallocate (AD)	0
	01	Attribute – Integral Dataset for Write (IDW)	0
	00	Attribute – Integral Dataset for Read (IDR)	0

Setting of Data Associated with the DSM Command					
Range	**Byte**	**Bit**		**Description**	**Value**
Range 0	15:08	-		Starting LBA	Start Address of Range 0
	07:04	-		Length in logical blocks	32 (16KB in block unit)
	03	Context Attributes	31:24	8 (4KB in block unit)	8 (4KB in block unit) [Recommended to set to 8]
	02:01		23:11	Reserved	0
			10	WP: Write Prepare	0
			09	SW: Sequential Write Range	0
			08	SR: Sequential Read Range	0
	00		07:06	Reserved	0
			05:04	AL: Access Latency	00b
			03:00	AF: Access Frequency	0h
Range 1	15:08	-		Starting LBA	Start Address of Range 1
	07:04	-		Length in logical blocks	32 (16KB in block unit)
	:	:		:	:
:					

Table 8-18 : Parameters for Release DIR/CI

The number of random areas to be released is specified by NR. Starting LBA shall be aligned to 16KB in the User Area, and Length in logical blocks shall be 16KB.

8.4.6.7.3 Maximum Command Duration

Refer to Section 8.4.3.5.1.

8.4.6.7.4 Error Conditions

At least one of the following conditions is satisfied, card rejects this command and sets 80h (Conflicting Attributes) to the Command Specific Status Values in the corresponding CMP.

- NR is greater than 15.
- At least one Start Address in the “Start LBA” field does not match to that of the current Random Area.
- Illegal value is set to at least one of “Length in logical blocks” field.

8.4.7 Power and Thermal Management

8.4.7.1 Overview

Available Maximum Powers (or PSDs) for SD Express card depend on card type as described in Section 8.1.8. In addition, NVMe specification has a feature of Host Controlled Thermal Management for controlling card’s performance in order not to rise its composite temperature so rapidly. This card’s

operation is called thermal throttling.
Since the objective of Speed Class Recording is to continue long time sequential write with guaranteeing minimum recording performance, sudden performance degradation by thermal throttling needs to be avoided.
This section describes a method to reduce card power consumption and to maintain the target Speed Class in terms of card's thermal design.

8.4.7.2 Power State Descriptor in the SD Express Card

Figure 8-24 shows an example of Power State Descriptor (PSD) in the ICDS. Each PSD is numbered with respect to Power State and has some elements including Maximum Power (MP). When host specifies a Power State to the card, card is allowed to consume power up to MP corresponding to the specified PS except cases described in Section 8.4.7.5.2.
As described in Section 8.1.8, an SD Express card has some Power States which shall be supported according to the card type. In the example of Figure 8-24, card supports 7 Power States indicated by PSD0 through PSD6, allocated in top-aligned and descending order with respect to MP in them. In addition, PSDs other than PSD2 and PSD4 shall be supported by this card (refer to Table 8.1.8-1).

Available PSD (or PS) shall be assigned in top-aligned and descending order of MP.

PSD (Power State Descriptor)	PS (Power State)	MP (Maximum Power)	...
PSD0	0	2.8W	...
PSD1	1	2.5W	...
PSD2	2	2.2W	...
PSD3	3	1.8W	...
PSD4	4	1.6W	...
PSD5	5	1.44W	...
PSD6	6	0.72W	...
PSD7	7	-	:
:	:	:	...
PSD31	31	-	...

Available PSDs in the card

Mandatory Optional

Figure 8-24 : An Example of PSD Assignment (Card Type: G3L2 or G4L1)

8.4.7.3 Thermal Management for the SD Express Speed Class

NVMe specification defines the following parameters related to the thermal management.
TMT1 and TMT2 indicate thresholds of temperature activating light thermal throttling and heavy thermal throttling respectively. They are set to the card by arguments of Set Features command (Host Controlled Thermal Management). If they are set to zero, the Host Controlled Thermal Management by TMT1/TMT2 is disabled. In addition, WCTEMP (Warning Composite Temperature Threshold) in the ICDS indicates the minimum composite temperature that indicates an overheating condition. Refer to NVMe specification for more details about these parameters.
At least, card shall be designed that its composite temperature does not to exceed WCTEMP under the specified Power State to maintain the Speed Class Recording, as far as host keeps the card case temperature as described in the Application Note below. Moreover, it is important not to fall the recording speed below the target class performance by the thermal throttling. In this sense, TMT2 should be less than or equal to WCTEMP (TMT2=WCTEMP is recommended). Note that the event that the composite temperature gets over WCTEMP may be reported by Thermal Excursion Event when some unexpected error happens due to not efficient heat removable.
To satisfy these conditions, power and thermal management for SD Express Speed Class is introduced as described in Section 8.4.7.4.

Application Note:
The host needs to keep card's surface temperature to the maximum Card Case Temperature or lower defined in Table 3-4 of the Standard Size SD Card Mechanical Addendum or Table 3-8 of the microSD Card Addendum.
The maximum Card Case Temperature depends on the Card Power Consumption in the table which is specified by the Maximum Power (MP) of the selected Power State Descriptor (PSD). When the selected PSD specifies 2.5W, for example, the maximum Card Case Temperature is kept up to 73 deg.C. by the host.
In addition, if the selected MP does not exactly match to the Card Power Consumption in the table, the higher nearest Card Power Consumption shall be referred for it. As an example, if the selected MP is 1.6W, the maximum Card Case Temperature shall be under 75 deg.C (corresponding to 1.8W).
For thermal management, host may use card's composite temperature obtained by SMART / Health Information Log instead of the Card Case Temperature.

8.4.7.4 Power and Thermal Management for SD Express Speed Class

8.4.7.4.1 Managing Maximum Power Consumption during Speed Class Recording

After completing SD Express card initialization, Slot Power Limit (SPL) is set by the host depending on host's power capability. When host starts normal write operation (targeting best effort performance without Speed Class Recording), it will set a PSD by Set Features command, which including the highest MP supported by the card as far as the MP does not exceed the SPL value.
In the SD Express Speed Class, such a higher power is not required. In terms of reducing card power consumption during the Speed Class, SD Express card supporting SD Express Speed Class shall have Speed Class Power State (SCPS) field in the Vendor Specific area of the ICDS. This field is defined according to each PCIe bus mode and target speed class. Refer to Table 8-20 for more details of SCPS. Figure 8-25 shows tables of SCPS and PSD as an example. If host selects PCIe Gen3x1 mode and intends to execute Class 150 Speed Class Recording, host reads corresponding value (84h) from the SCPS table. Then, it refers to the PSD table, detects PSD4 (1.6W), and set it as a Power State during Speed Class Recording by Set Features command. If TMT1 is not zero, the SCPS means a suitable Power State for the Speed Class when the composite temperature is between TMT1 and TMT2 (refer to Section 8.4.7.4.3 and 8.4.7.5.2).

Byte Offset	Value
3088 (Class 150)	**84h (PSD4)**
3089 (Class 300)	83h (PSD3)
3090 (Class 450)	82h (PSD2)
3091 (Class 600)	81h (PSD1)

Bit [7]	Bit [6:5]	Bit [4:0]
Validity (1: Valid)	Reserved	Index of Power State Descriptor

Speed Class Power State (SCPS) for Gen3x1 in ICDS defined by Table 8-20

PSD	MP
PSD0	2.8W
PSD1	2.5W
PSD2	2.2W
PSD3	1.8W
PSD4	**1.6W**
PSD5	1.44W
PSD6	0.72W
PSD7	-
:	:
PSD31	-

Mandatory Optional

Power State Descriptor in ICDS (Card Type: G4L1)

Figure 8-25 : An Example of Speed Class Power State

8.4.7.4.2 Managing Composite Temperature during Speed Class Recording

As described above, card shall be designed that the composite temperature is under WCTEMP by thermal throttling or other ways. However, the appropriate values of TMT1 and TMT2 depend on the target speed class and card structural implementation. So, the SD Express Card supporting SD Express Speed Class shall indicate Speed Class TMT1 and TMT2 for the Speed Class Recording in the Vendor Specific area of the ICDS (they are defined as SCTMT1 and SCTMT2 respectively). These values are defined according to each PCIe bus mode and target speed class. Before starting Speed Class Recording, host reads SCTMT1 and SCTMT2 from the ICDS of the card and sets them as TMT1 and TMT2 respectively by Set Features command.
Card may indicate SCTMT1 and SCTMT2 values by either one of methods below.

(1) If card is designed without referring to TMT1 and TMT2 for keeping Speed Class Recording, card indicates 0 for both SCTMT1 and SCTMT2 fields. In this case, card expects to be set SCTMT1 and SCTMT2 value (both parameters are 0) to TMT1 and TMT2 by the host respectively. If host sets non-zero values to TMT1/2, card may not guarantee the Speed Class Recording.
(2) Otherwise, card indicates appropriate values for TMT1 and TMT2 for each PCIe bus mode and target speed class. In this case, card expects to be set SCTMT1 and SCTMT2 to TMT1 and TMT2 by the host respectively. If host violates this rule, card may not guarantee the Speed Class Recording.

For the second method above, there are two options for card design related to its composite temperature as described in Figure 8-26.

(a) If the card intends to maintain the target speed without any throttlings, card shall be designed in order that its composite temperature never exceeds TMT1 (set as SCTMT1 by the host) during the Speed Class Recording.
(b) If the card enables to maintain the target speed with the light throttling, card shall be designed in order that its composite temperature never exceeds TMT2 (set as SCTMT2 by the host) during the Speed Class Recording.

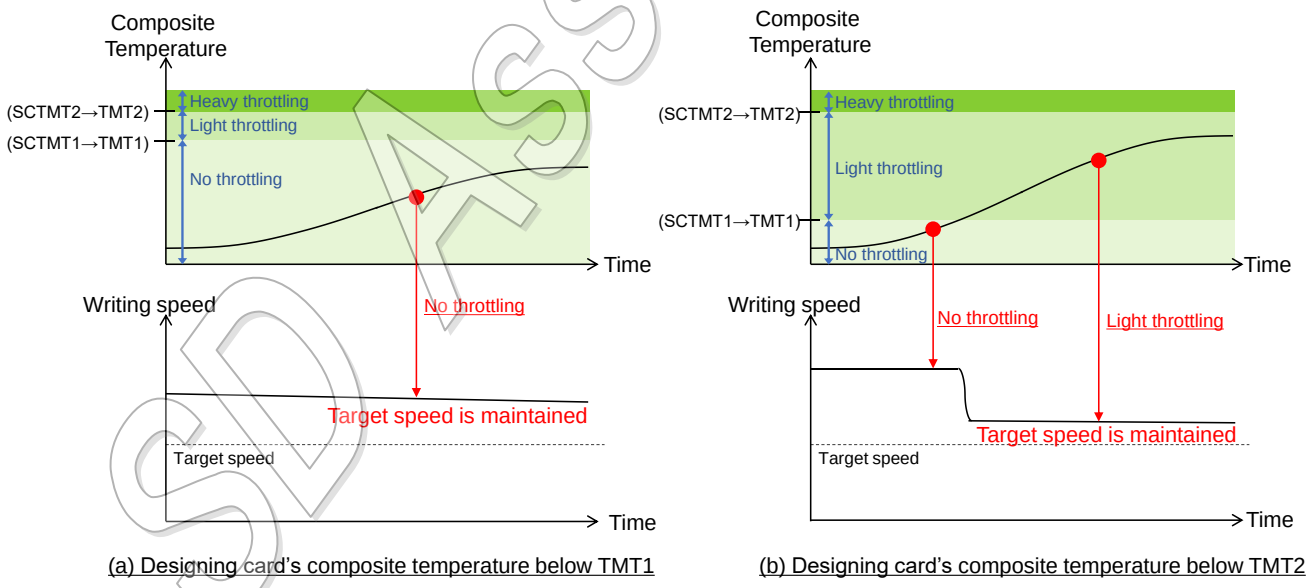


Figure 8-26 : Relationship between Composite Temperature and Writing Speed

Refer to Table 8-20 for more details of SCTMT1 and SCTMT2.

8.4.7.4.3 Speed Class Recording Including Power and Thermal Management

The following steps take place for executing Speed Class Recording including Power and Thermal management. Also refer to Section 8.4.7.5 for the examples.

(1) Host sets Slot Power Limit (SPL) according to its power capability.
(2) If necessary, host reads the default PSD, TMT1 and TMT2 values from the card and stores them before starting Speed Class Recording
(3) Host reads appropriate values of SCPS, SCTMT1 and SCTMT2 according to the selected PCIe bus mode and target speed class (see Table 8-20), and then sets them as PSD, TMT1 and TMT2 respectively by Set Features command.
   a. If zero is set to TMT1 and TMT2, Host Controlled Thermal Management by TMT1/TMT2 is disabled. In this case, card is necessary to control power consumption up to MP specified by the SCPS. In addition, card shall be designed to keep Speed Class Recording by retaining the composite temperature under WCTEMP, as far as host maintains the conditions described in the Application Note of Section 8.4.7.3.
   b. If non-zero value is set to TMT1 and TMT2, card is necessary to control power consumption up to MP specified by the SCPS. Similar to a. above, card shall be designed to keep Speed Class Recording by retaining the composite temperature under WCTEMP (usually same as TMT2=SCTMT2).
      In this case, card may operate on a Power State whose index is less than that of specified SCPS while the composite temperature is under TMT1, as far as the corresponding MP does not exceed SPL. For example, when PSD2 is set to the card according to the SCPS table, card may work at either PSD1 or PSD0 if the composite temperature is lower than TMT1.
(4) On exiting the Speed Class Recording, host may restore the saved PSD, TMT1 and TMT2 values by Set Features command.

Application Note: If the host does not maintain the card's surface temperature described in the Application Note in Section 8.4.7.3, card does not guarantee the speed class performance even host specifies the recommended SCTMT1 and SCTMT2 values to the card.

8.4.7.5 Sample Speed Class Sequences Including Power and Thermal Management

8.4.7.5.1 In Case of SCTMT1/2=0

Figure 8-27 shows an example of SD Express Speed Class sequence when SCTMT1 and SCTMT2 are zero. In this example, host supports PCIe Gen4x1 mode, card type is G4L1 and the target speed class is 150MB/s.

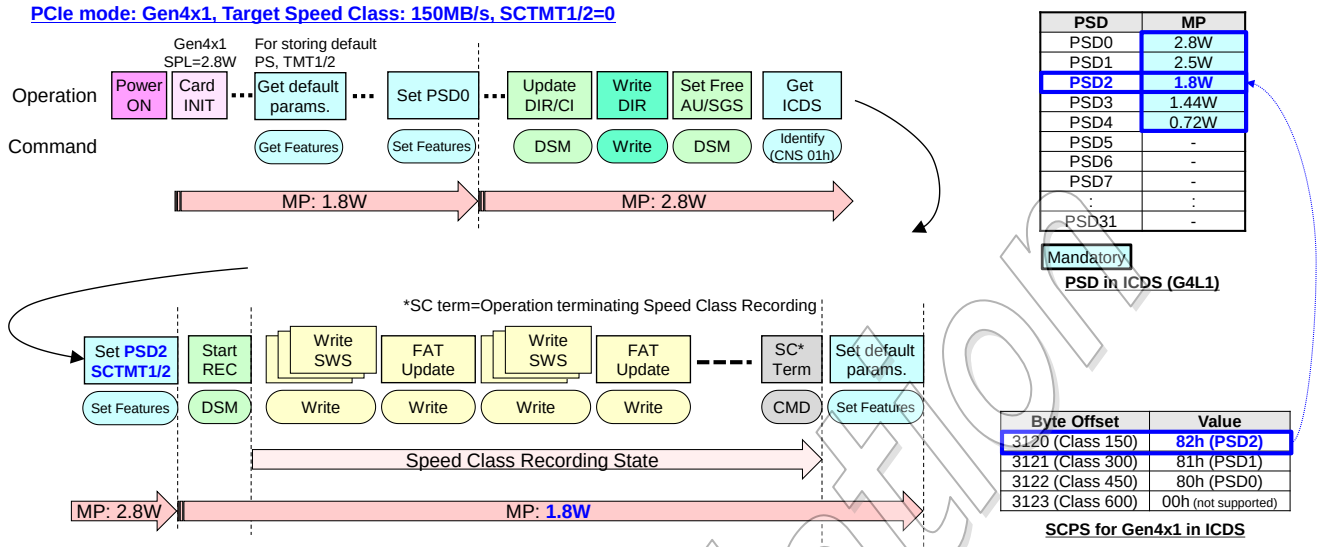


Figure 8-27 : Sample Speed Class Sequence Including Power and Thermal Management (1)

At the card initialization, host selects PCIe Gen4x1 mode and sets SPL=2.8W since host can supply up to 2.8W to the SD Express card. Then, host issues Get Features command to read default settings of PSD, TMT1 and TMT2 for storing these parameters if necessary.
After host initializes the SD Express card, it usually sets Power State whose MP is the highest one as far as MP does not exceed the value of SPL. In this example, since the highest MP (2.8W) is equal to SPL, PSD0 is set by Set Features command.
Before starting Speed Class Recording, host issues Identify command to get Vendor Specific parameters for SD Express Speed Class from ICDS. At this time, host detects PSD2 (1.8W) is suitable Power State Descriptor for this card when the target speed class is 150 over PCIe Gen4x1 mode. In addition, host recognizes both SCTMT1 and SCTMT2 are zero. Then, host issues Set Features command to set these parameters for executing Speed Class recording. Therefore, the MP during Speed Class Recording gets 1.8W. This means card never consumes power more than 1.8W.
After exiting from the Speed Class Recording State, host may set default parameters stored previously.

Figure 8-28 shows another example of SD Express Speed Class sequence when SCTMT1 and SCTMT2 are zero. In this example, host supports only PCIe Gen3x1 mode and can supply up to 1.8W.

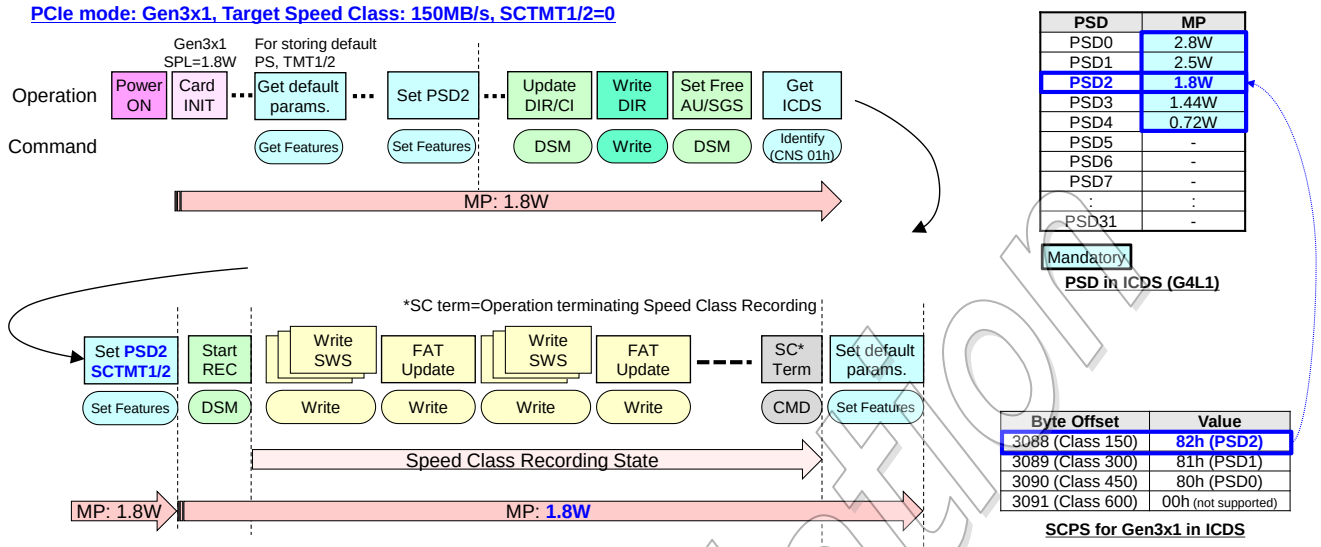


Figure 8-28 : Sample Speed Class Sequence Including Power and Thermal Management (2)

Due to host's capability, host selects PCIe Gen3x1 mode and sets SPL=1.8W. Thus, PSD2 (1.8W) is set in this host and card combination. Since the SCPS for this condition indicates PSD2, host will set PSD2 and the maximum power consumption is still 1.8W during Speed Class Recording.

8.4.7.5.2 In Case of SCTMT1/2≠0

Figure 8-29 shows an example of SD Express Speed Class sequence when SCTMT1 and SCTMT2 are not zero. In this example, host supports PCIe Gen4x1 mode, card type is G4L1 and the target speed class is 150MB/s.

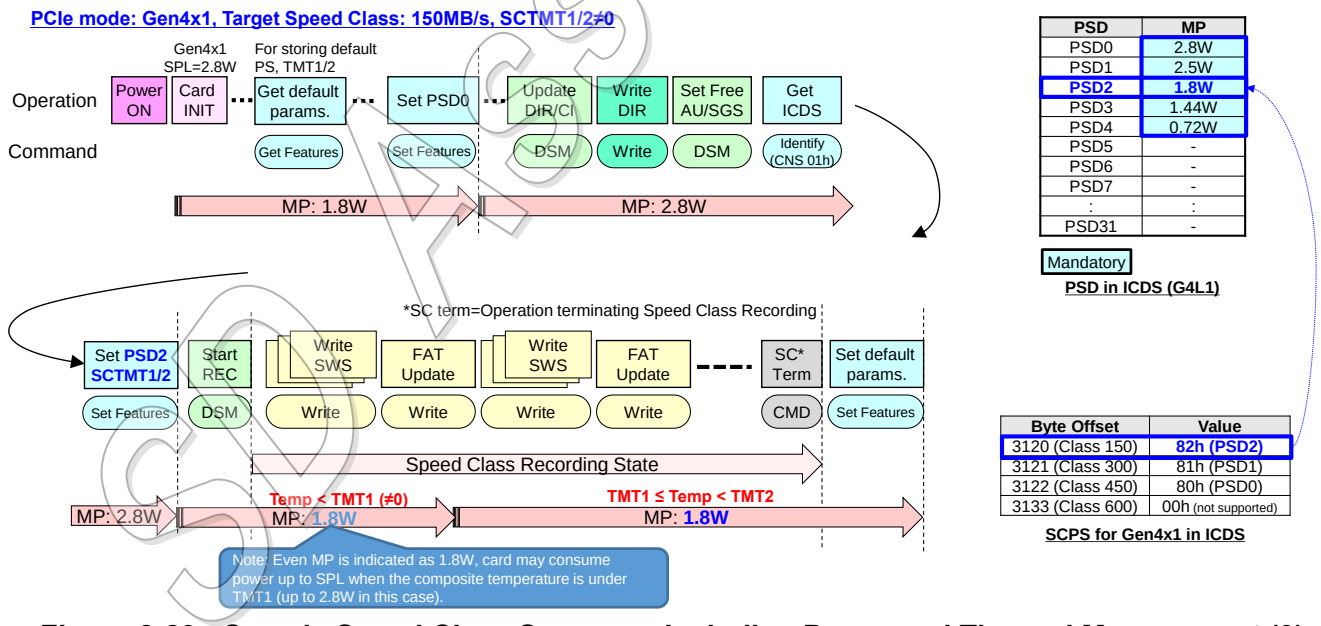


Figure 8-29 : Sample Speed Class Sequence Including Power and Thermal Management (3)

Before issuing Set Features command for setting SCPS (PSD2) and SCTMT1/2 values as Power State and TMT1/2 respectively, host and card operations are same as those of Figure 8-27. Different from Figure 8-27, when the composite temperature is less than TMT1 (same as SCTMT1≠0 set by the host),

card may consume power between the wattage between MP designated by SCPS and SPL. In this example, even host specifies PSD2 (whose MP is 1.8W) with reference to SCPS obtained by the card, card may work up to 2.8W (=SPL).
When the composite temperature gets higher than or equal to TMT1, card can consume power up to 1.8W.

Figure 8-30 shows another example of SD Express Speed Class sequence when SCTMT1 and SCTMT2 are not zero. Similar to Figure 8-28, host supports only PCIe Gen3x1 mode and can supply up to 1.8W. On the other hand, card supports optional Power States (2.2W and 1.6W) as shown in Figure 8-30.

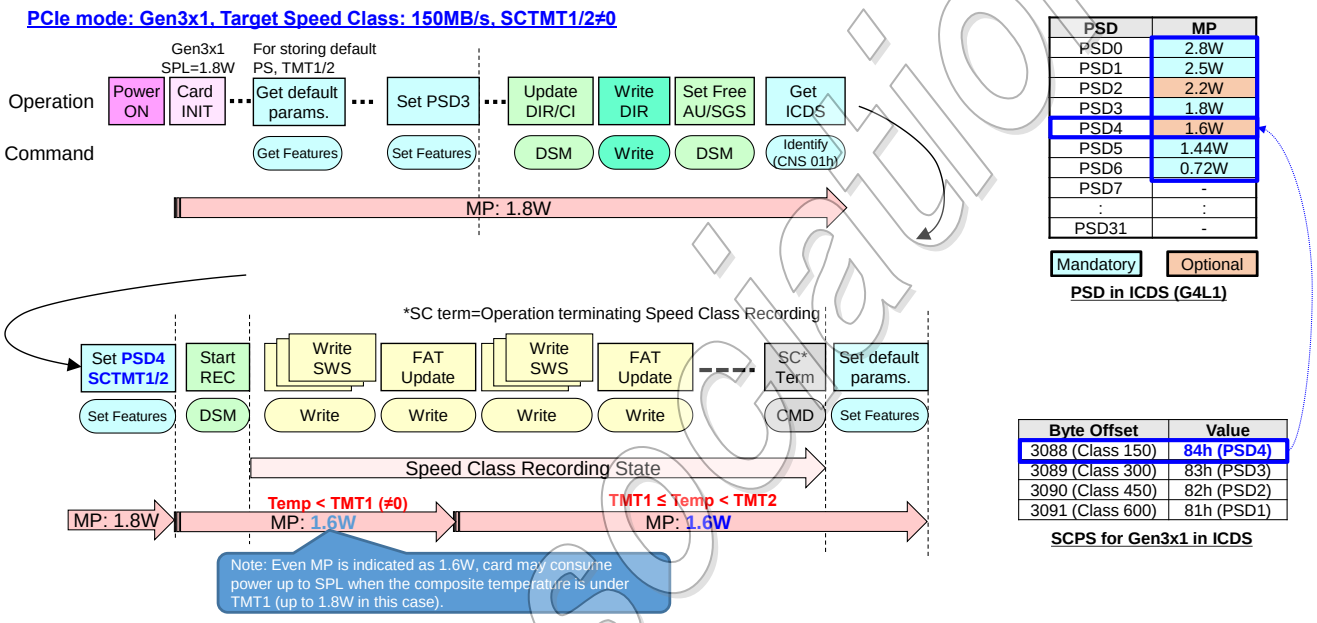


Figure 8-30 : Sample Speed Class Sequence Including Power and Thermal Management (4)

Similar to Figure 8-28, host selects PCIe Gen3x1 mode and sets SPL=1.8W. In this card, PSD3 corresponds to MP=1.8W, host issues Set Features command to set PSD3 after initializing the card.
Before starting Speed Class Recording, host issues Identify command to obtain SCPS (and PSD table) and SCTMT1/2. SCPS indicates PSD4 in this condition, host issues Set Features command to set PSD4 and SCTMT1/2.
Like Figure 8-29, card may operate up to 1.8W (=SPL) while the composite temperature is under TMT1, even specified PSD (PSD4 in this case) indicates its MP is equal to 1.6W. When the temperature gets higher than or equal to TMT1, card shall work up to 1.6W as a power consumption.

8.4.8 Relationship to the Conventional Speed Classes

8.4.8.1 Supporting Speed Classes by SD Express Cards

SD Express Speed Class specification is defined independent to that of Video Speed Class. This means all types of SD Express card described in Table 8-19 are allowed.

Type	Supporting SD Express Speed Class	Supporting Video Speed Class
1	No	No
2	No	Yes
3	Yes	No
4	Yes	Yes

Table 8-19 : SD Express Card Types in Terms of Speed Classes

In addition, supporting the original speed class (C2, C4, C6, and C10) and the UHS speed grade (U1 and U3) are also optional in SD mode for all types of SD Express Card in Table 8-19.

8.4.8.2 Operation Rules When Changing Bus Mode and Power Cycle

The following rules are applied to Type 2, 3, or 4 of the SD Express cards defined in Table 8-19.

8.4.8.2.1 Changing Bus Mode from SD to PCIe

- All registered address information of active AUs (assigned by Set Free AU) shall be cleared.
- All registered address information of DIRs (assigned by Update DIR) shall be cleared.
- The suspension address over SD bus (SUS_ADDR) is remained but inaccessible while the bus mode is PCIe.
- If either write, erase, or Set Free AU/SGS operation is executed to the suspended AU (the AU including SUS_ADDR) through PCIe bus, the remained SUS_ADDR is cleared.

8.4.8.2.2 Changing Bus Mode from PCIe to SD

- All of context attributes specified by DSM commands (both random and sequential) shall be cleared.
- The suspension addresses over PCIe bus (SD Express SUS_ADDR 1 through 8) are remained but inaccessible while the bus mode is SD.
- If either write, erase, discard or Set Free AU operation is executed to the suspended SGS unit (the unit including SD Express SUS_ADDR) through SD bus, the remained SD Express SUS_ADDR is cleared.

8.4.8.2.3 Power Cycle

- All registered address information of active AUs (assigned by Set Free AU) shall be cleared.
- All registered address information of DIR (assigned by Update DIR) shall be cleared.
- All of context attributes specified by DSM commands (both random and sequential) shall be cleared.
- The suspension addresses over both bus mode are remained.
- If either write, erase or Set Free AU/SGS operation is executed to the suspended AU (the AU including SUS_ADDR) through PCIe bus, the remained SUS_ADDR is cleared.
- If either write, erase, discard or Set Free AU operation is executed to the suspended SGS unit (the unit including SD Express SUS_ADDR) through SD bus, the remained SD Express SUS_ADDR is cleared.

8.4.8.3 Recommendation for Setting SGS and VSC_AU_SIZE

For the SD Express Type 4 card, it is preferable that both SGS value and VSC_AU_SIZE are identical. But if it is impossible, it is recommended that SGS value is n times or 1/n times of VSC_AU_SIZE, where n is a natural number, for the efficient use of the User Area accessed by both PCIe bus and SD bus.

8.4.9 Detecting Termination Conditions

When one of following events occur, it is impossible for card to continue Speed Class Recording for specific stream(s) or all of them. Refer to Section 8.4.1.2 for details of Area Assignment.

- Termination conditions for each stream
  (1) Writing a specific stream violates the rules described in Section 8.4.1.6.
  (2) Card internal error occurs for a specific stream.
- Termination conditions for all valid streams
  (1) Card receives a command other than Directive Receive, Directive Send, Identify, Get Log Page, DSM, Flush, Read or Write.
  (2) Write or erase operation is executed in the Non-assigned Area.
  (3) File system management information other than FAT and Allocation Bitmap is written to other area than the Random Area allocated by Update DIR/CI. Stream data are written to other area than the Sequential Area allocated by Set Free AU/SGS.
  (4) Card internal error which is impossible to specify its SID (e.g., error during FAT update) occurs.
  (5) PCIe bus error occurs.

When card detects these events, it reports by Asynchronous Event function. To enable this function, host issues Asynchronous Event Request command before starting Speed Class Recording. When card detects one of termination conditions during Speed Class Recording, it posts CMP of Asynchronous Event Request in the Admin Completion Queue, whose Asynchronous Event Type is 111b (vendor specific) and Log Page Identifier is C0h (Open SID for Speed Class). If host detects these parameters in the Dword0 of the Completion Queue Entry for the Admin Completion Queue, it should issue Get Log Page command with Log Page Identifier = C0h and UUID for the SD Express Speed Class. By this method, host gets SIDs that are no longer valid.
After that, host should operate Release Identifier for the invalid SID.
Refer to Section 8.4.1.7 and 8.4.10.3 for more details.

8.4.10 Vendor Specific Parameters

8.4.10.1 Getting Vendor Specific Parameters

In order that host obtains specific parameters related to SD Express Speed Class, the following two methods are introduced.
The first one is host reads parameters from the Vendor Specific field in the ICDS by Identify command. In this field, static parameters are indicated such as SD Express Speed Class, Speed Class Power State and so on.
The second one is host gets parameters as a log by issuing Get Log Page command. Dynamically changing parameters such as Open SID for Speed Class or SD Express SUS_ADDR can be obtained by this method.
Card supporting SD Express Speed Class shall indicate the following 128-bit value as a UUID in the UUID List defined in the NVMe specification.

UUID for SD Express Speed Class: **724e2db4-8569-406b-8ecc-b3f27f89169e**

When host issues Identify or Get Log Page command to get specific parameters for SD Express Speed Class, host needs to set UUID index specifying the UUID for SD Express Speed Class described above.

8.4.10.2 Parameters Obtained by Identify Command

8.4.10.2.1 Overview

Identify Controller Data Structure includes the Vendor Specific field as described in Figure 8-31.

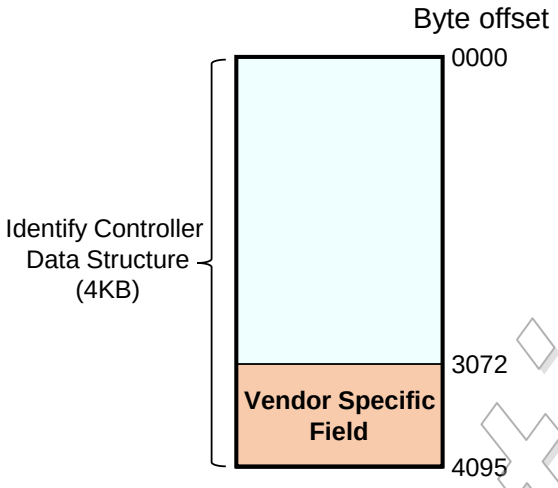


Figure 8-31 : Vendor Specific Field in the ICDS

When host issues Identify command including UUID index designating SD Express Speed Class (see Section 8.4.10.1), host gets parameters defined in Table 8-20. Otherwise, the Vendor Specific field indicates values not related to the SD Express Speed Class.

8.4.10.2.2 Vendor Specific Parameters in ICDS

Table 8-20 indicates vendor specific parameters for SD Express Speed Class defined in the Vendor Specific field of ICDS.

<table>
<tr><th>Bytes</th><th>O/M[1]</th><th>Description</th></tr>
<tr><td>3073:3072</td><td>M</td><td>SD Express Speed Class: This field indicates the SD Express Speed Class supported by the card.
The value of this 2 Bytes field indicates the value of the speed itself guaranteed by the SD Express Speed Class.
Any SD Express Speed Class lower than the specified Class value shall be also supported (e.g., if the card supports SD Express Speed Class 600, then the card also supports SD Express Speed Class 450, 300, and 150).
Higher classes may be added in the future, if required.
Considering the future extension, even an undefined value is indicated in this field, host should not reject this card and regard that card supports its nearest lower class. For example, if card indicates 01F4h (500) in this field, host should recognize that card supports SD Express Speed Class 450.
If card supports TCG security (refer to Section 4.22), this field can be set based on the performance when TCG is disabled.
<table>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>0096h</td><td>SD Express Speed Class 150 (150MB/s)</td></tr>
<tr><td>012Ch</td><td>SD Express Speed Class 300 (300MB/s)</td></tr>
<tr><td>01C2h</td><td>SD Express Speed Class 450 (450MB/s)</td></tr>
<tr><td>0258h</td><td>SD Express Speed Class 600 (600MB/s)</td></tr>
<tr><td>Others</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>3087:3074</td><td></td><td>Reserved</td></tr>
</table>

<table>
<tr>
<td>3103:3088</td>
<td>M</td>
<td>
Speed Class Power State (SCPS) for Gen3x1: This field indicates the Power States that the host should set before starting SD Express Speed Class Recording in PCIe Gen3x1 mode, according to the target speed class. By this operation, host can save power and suppress heat generation.
Each byte field indicates appropriate Power State (0 to 31, refer to Figure 8-24) in Bit [4:0] and its validity (1 means the corresponding speed class is valid or supported, 0 otherwise) in Bit [7] for each speed class. Card shall set ‘1’ to Bit [7] and appropriate Power State for all supported speed classes. Card sets ‘0’ to Bit [7] if it does not support the associated speed class. If card does not support Class 300 or higher, 00h is set to the associated byte field.
Card supporting some Speed Class shall support lower classes as well.
<table>
<tr><th>Bytes</th><th>Bit [7]</th><th>Bits [6:5]</th><th>Bits [4:0]</th></tr>
<tr><td>3088</td><td>Validity of Speed Class 150</td><td rowspan="4">Reserved</td><td>Power State for Gen3x1 Speed Class 150</td></tr>
<tr><td>3089</td><td>Validity of Speed Class 300</td><td>Power State for Gen3x1 Speed Class 300</td></tr>
<tr><td>3090</td><td>Validity of Speed Class 450</td><td>Power State for Gen3x1 Speed Class 450</td></tr>
<tr><td>3091</td><td>Validity of Speed Class 600</td><td>Power State for Gen3x1 Speed Class 600</td></tr>
<tr><td>3092</td><td colspan="3">Reserved</td></tr>
<tr><td>3093</td><td colspan="3">Reserved</td></tr>
<tr><td>:</td><td colspan="3">:</td></tr>
<tr><td>3103</td><td colspan="3">Reserved</td></tr>
</table>
</td>
</tr>
<tr>
<td>3119:3104</td>
<td>MiS</td>
<td>
Speed Class Power State (SCPS) for Gen3x2: This field indicates the Power States that the host should set before starting SD Express Speed Class Recording in PCIe Gen3x2 mode, according to the target speed class.
Setting of each field is same as that of Speed Class Power State for Gen3x1. In addition, if card does not support PCIe Gen3x2 mode, all of this field are filled with 0.
<table>
<tr><th>Bytes</th><th>Bit [7]</th><th>Bits [6:5]</th><th>Bits [4:0]</th></tr>
<tr><td>3104</td><td>Validity of Speed Class 150</td><td rowspan="4">Reserved</td><td>Power State for Gen3x2 Speed Class 150</td></tr>
<tr><td>3105</td><td>Validity of Speed Class 300</td><td>Power State for Gen3x2 Speed Class 300</td></tr>
<tr><td>3106</td><td>Validity of Speed Class 450</td><td>Power State for Gen3x2 Speed Class 450</td></tr>
<tr><td>3107</td><td>Validity of Speed Class 600</td><td>Power State for Gen3x2 Speed Class 600</td></tr>
<tr><td>3108</td><td colspan="3">Reserved</td></tr>
<tr><td>3109</td><td colspan="3">Reserved</td></tr>
<tr><td>:</td><td colspan="3">:</td></tr>
<tr><td>3119</td><td colspan="3">Reserved</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr><td>3135:3120</td><td>MiS</td><td>
Speed Class Power State (SCPS) for Gen4x1: This field indicates the Power States that the host should set before starting SD Express Speed Class Recording in PCIe Gen4x1 mode, according to the target speed class.
Setting of each field is same as that of Speed Class Power State for Gen3x1. In addition, if card does not support PCIe Gen4x1 mode, all of this field are filled with 0.

<table>
<tr><th>Bytes</th><th>Bit [7]</th><th>Bits [6:5]</th><th>Bits [4:0]</th></tr>
<tr><td>3120</td><td>Validity of Speed Class 150</td><td rowspan="4">Reserved</td><td>Power State for Gen4x1 Speed Class 150</td></tr>
<tr><td>3121</td><td>Validity of Speed Class 300</td><td>Power State for Gen4x1 Speed Class 300</td></tr>
<tr><td>3122</td><td>Validity of Speed Class 450</td><td>Power State for Gen4x1 Speed Class 450</td></tr>
<tr><td>3123</td><td>Validity of Speed Class 600</td><td>Power State for Gen4x1 Speed Class 600</td></tr>
<tr><td>3124</td><td colspan="3">Reserved</td></tr>
<tr><td>3125</td><td colspan="3">Reserved</td></tr>
<tr><td>:</td><td colspan="3">:</td></tr>
<tr><td>3135</td><td colspan="3">Reserved</td></tr>
</table>
</td></tr>
<tr><td>3151:3136</td><td>MiS</td><td>
Speed Class Power State (SCPS) for Gen4x2: This field indicates the Power States that the host should set before starting SD Express Speed Class Recording in PCIe Gen4x2 mode, according to the target speed class.
Setting of each field is same as that of Speed Class Power State for Gen3x1. In addition, if card does not support PCIe Gen4x2 mode, all of this field are filled with 0.

<table>
<tr><th>Bytes</th><th>Bit [7]</th><th>Bits [6:5]</th><th>Bits [4:0]</th></tr>
<tr><td>3136</td><td>Validity of Speed Class 150</td><td rowspan="4">Reserved</td><td>Power State for Gen4x2 Speed Class 150</td></tr>
<tr><td>3137</td><td>Validity of Speed Class 300</td><td>Power State for Gen4x2 Speed Class 300</td></tr>
<tr><td>3138</td><td>Validity of Speed Class 450</td><td>Power State for Gen4x2 Speed Class 450</td></tr>
<tr><td>3139</td><td>Validity of Speed Class 600</td><td>Power State for Gen4x2 Speed Class 600</td></tr>
<tr><td>3140</td><td colspan="3">Reserved</td></tr>
<tr><td>3141</td><td colspan="3">Reserved</td></tr>
<tr><td>:</td><td colspan="3">:</td></tr>
<tr><td>3151</td><td colspan="3">Reserved</td></tr>
</table>
</td></tr>
<tr><td>3183:3152</td><td></td><td>Reserved</td></tr>
</table>

<table>
<tr>
<td>3247:3184</td>
<td>M</td>
<td>
Speed Class TMTs (SCTMT1, SCTMT2) for Gen3x1: This field indicates the temperatures, in degrees Kelvin, that the host shall specify as TMT1 and TMT2 when PCIe Gen3x1 mode is selected. SCTMT1 and SCTMT2 indicated in this field correspond to TMT1 and TMT2 specified by Set Features Command and depend on PCIe bus mode and the target speed class. The value of 0000h indicates invalid.
Card supporting some Speed Class shall support lower classes as well. Then, since card supporting Class 300 or higher indicates multiple valid SCTMT1 and SCTMT2, host should select appropriate SCTMT1 and SCTMT2 according to the required performance by the host.
If host does not set these values as TMT1 and TMT2, unexpected thermal throttling will happen and the SD Express Speed Class Recording may not be able to continue.
Since the following relation MNTMT ≤ TMT1 < TMT2 ≤ MXTMT is regulated in the NVMe spec., SCTMT1 and SCTMT2 shall satisfy the following inequality.

MNTMT ≤ SCTMT1 < SCTMT2 ≤ MXTMT

If card does not support Class 300 or higher, 00h is set to the associated byte fields.
<table>
<tr><th>Bytes</th><th>Bits [31:16]</th><th>Bits [15:0]</th></tr>
<tr><td>3187:3184</td><td>SCTMT1 for Speed Class 150 over PCIe Gen3x1</td><td>SCTMT2 for Speed Class 150 over PCIe Gen3x1</td></tr>
<tr><td>3191:3188</td><td>SCTMT1 for Speed Class 300 over PCIe Gen3x1</td><td>SCTMT2 for Speed Class 300 over PCIe Gen3x1</td></tr>
<tr><td>3195:3192</td><td>SCTMT1 for Speed Class 450 over PCIe Gen3x1</td><td>SCTMT2 for Speed Class 450 over PCIe Gen3x1</td></tr>
<tr><td>3199:3196</td><td>SCTMT1 for Speed Class 600 over PCIe Gen3x1</td><td>SCTMT2 for Speed Class 600 over PCIe Gen3x1</td></tr>
<tr><td>3203:3200</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>3207:3204</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>:</td><td>:</td><td>:</td></tr>
<tr><td>3247:3244</td><td>Reserved</td><td>Reserved</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr><td>3311:3248</td><td>MiS</td><td>
Speed Class TMTs (SCTMT1, SCTMT2) for Gen3x2: This field indicates the temperatures, in degrees Kelvin, that the host shall specify as TMT1 and TMT2 when PCIe Gen3x2 mode is selected.
Setting of each field is same as that of Speed Class TMTs (SCTMT1 and SCTMT2) for Gen3x1. In addition, if card does not support PCIe Gen3x2 mode, all of this field are filled with 0.

<table>
<tr><th>Bytes</th><th>Bits [31:16]</th><th>Bits [15:0]</th></tr>
<tr><td>3251:3248</td><td>SCTMT1 for Speed Class 150 over PCIe Gen3x2</td><td>SCTMT2 for Speed Class 150 over PCIe Gen3x2</td></tr>
<tr><td>3255:3252</td><td>SCTMT1 for Speed Class 300 over PCIe Gen3x2</td><td>SCTMT2 for Speed Class 300 over PCIe Gen3x2</td></tr>
<tr><td>3259:3256</td><td>SCTMT1 for Speed Class 450 over PCIe Gen3x2</td><td>SCTMT2 for Speed Class 450 over PCIe Gen3x2</td></tr>
<tr><td>3263:3260</td><td>SCTMT1 for Speed Class 600 over PCIe Gen3x2</td><td>SCTMT2 for Speed Class 600 over PCIe Gen3x2</td></tr>
<tr><td>3267:3264</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>3271:3268</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>:</td><td>:</td><td>:</td></tr>
<tr><td>3311:3308</td><td>Reserved</td><td>Reserved</td></tr>
</table>
</td></tr>
<tr><td>3375:3312</td><td>MiS</td><td>
Speed Class TMTs (SCTMT1, SCTMT2) for Gen4x1: This field indicates the temperatures, in degrees Kelvin, that the host shall specify as TMT1 and TMT2 when PCIe Gen4x1 mode is selected.
Setting of each field is same as that of Speed Class TMTs (SCTMT1 and SCTMT2) for Gen3x1. In addition, if card does not support PCIe Gen4x1 mode, all of this field are filled with 0.

<table>
<tr><th>Bytes</th><th>Bits [31:16]</th><th>Bits [15:0]</th></tr>
<tr><td>3315:3312</td><td>SCTMT1 for Speed Class 150 over PCIe Gen4x1</td><td>SCTMT2 for Speed Class 150 over PCIe Gen4x1</td></tr>
<tr><td>3319:3316</td><td>SCTMT1 for Speed Class 300 over PCIe Gen4x1</td><td>SCTMT2 for Speed Class 300 over PCIe Gen4x1</td></tr>
<tr><td>3323:3320</td><td>SCTMT1 for Speed Class 450 over PCIe Gen4x1</td><td>SCTMT2 for Speed Class 450 over PCIe Gen4x1</td></tr>
<tr><td>3327:3324</td><td>SCTMT1 for Speed Class 600 over PCIe Gen4x1</td><td>SCTMT2 for Speed Class 600 over PCIe Gen4x1</td></tr>
<tr><td>3331:3328</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>3335:3332</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>:</td><td>:</td><td>:</td></tr>
<tr><td>3375:3372</td><td>Reserved</td><td>Reserved</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td>3439:3376</td><td>MiS</td><td>Speed Class TMTs (SCTMT1, SCTMT2) for Gen4x2: This field indicates the temperatures, in degrees Kelvin, that the host shall specify as TMT1 and TMT2 when PCIe Gen4x2 mode is selected.
Setting of each field is same as that of Speed Class TMTs (SCTMT1 and SCTMT2) for Gen3x1. In addition, if card does not support PCIe Gen4x2 mode, all of this field are filled with 0.

<table>
<tr><th>Bytes</th><th>Bits [31:16]</th><th>Bits [15:0]</th></tr>
<tr><td>3379:3376</td><td>SCTMT1 for Speed Class 150 over PCIe Gen4x2</td><td>SCTMT2 for Speed Class 150 over PCIe Gen4x2</td></tr>
<tr><td>3383:3380</td><td>SCTMT1 for Speed Class 300 over PCIe Gen4x2</td><td>SCTMT2 for Speed Class 300 over PCIe Gen4x2</td></tr>
<tr><td>3387:3384</td><td>SCTMT1 for Speed Class 450 over PCIe Gen4x2</td><td>SCTMT2 for Speed Class 450 over PCIe Gen4x2</td></tr>
<tr><td>3391:3388</td><td>SCTMT1 for Speed Class 600 over PCIe Gen4x2</td><td>SCTMT2 for Speed Class 600 over PCIe Gen4x2</td></tr>
<tr><td>3395:3392</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>3399:3396</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>:</td><td>:</td><td>:</td></tr>
<tr><td>3439:3436</td><td>Reserved</td><td>Reserved</td></tr>
</table>
</td></tr>
<tr><td>4095:3440</td><td></td><td>Reserved</td></tr>
</table>

Note 1) O/M definition: M = Mandatory, “MiS”: Mandatory if card supports corresponding PCIe bus mode O/M field is valid when the UUID for SD Express Speed Class (see Section 8.4.10.1) is included in one of UUID List Entry.

Table 8-20 : Vendor Specific Parameters for SD Express Speed Class

8.4.10.3 Parameters Obtained by Get Log Page Command

8.4.10.3.1 Overview

When host issues Get Log Page command with UUID index designating SD Express Speed Class (see Section 8.4.10.1) and a specific Log page Identifier (LID), host gets parameters related to SD Express Speed Class. The contents depend on LID as follows.

- Open SID for Speed Class (LID=C0h)
- SD Express SUS_ADDR (LID=C1h)

If the UUID for SD Express Speed Class is not specified by the UUID index or the LID does not match to the one described above, the obtained log is not related to the SD Express Speed Class.

8.4.10.3.2 Open SID for Speed Class (LID=C0h)

Table 8-21 indicates log information of Open SID for Speed Class (OSID-SC). In addition, Table 8-22 specifies Open SID for Speed Class Data Structure (OSID-SC DS) embedded the OSID-SC information log.

Bytes	Description
01:00	**Open SID Count for Speed Class (OSCSC):** This field indicates the number of SID opened for SD Express Speed Class. Maximum value of OSCSC is min(MSL, 8). Host can start another stream recordings based on Speed Class specification when OSCSC is less than min(MSL, 8). If OSCSC=min(MSL, 8), “Start Recording” fails. In this case, if host detects invalid OSID-SCs from this information and operates Release Identifier for them, host can start stream recording because OSCSC gets less than min(MSL, 8).
07:02	Reserved
15:08	**1st Open SID for Speed Class Data Structure (OSID-SC DS 1):** This field defines the first Open SID for Speed Class Data Structure. OSID-SC DSes are listed in ascending order with respect to the applicable OSID-SC started from “OSID-SC DS 1.” If OSCSC is less than 8, the rest fields of OSID-SC DS are filled by zero. Refer to Table 8-22 for more details.
23:16	**2nd Open SID for Speed Class Data Structure (OSID-SC DS 2):** This field defines the second Open SID for Speed Class Data Structure. Refer to Table 8-22 for more details.
31:24	**3rd Open SID for Speed Class Data Structure (OSID-SC DS 3):** This field defines the third Open SID for Speed Class Data Structure. Refer to Table 8-22 for more details.
39:32	**4th Open SID for Speed Class Data Structure (OSID-SC DS 4):** This field defines the fourth Open SID for Speed Class Data Structure. Refer to Table 8-22 for more details.
47:40	**5th Open SID for Speed Class Data Structure (OSID-SC DS 5):** This field defines the fifth Open SID for Speed Class Data Structure. Refer to Table 8-22 for more details.
55:48	**6th Open SID for Speed Class Data Structure (OSID-SC DS 6):** This field defines the sixth Open SID for Speed Class Data Structure. Refer to Table 8-22 for more details.
63:56	**7th Open SID for Speed Class Data Structure (OSID-SC DS 7):** This field defines the seventh Open SID for Speed Class Data Structure. Refer to Table 8-22 for more details.
71:64	**8th Open SID for Speed Class Data Structure (OSID-SC DS 8):** This field defines the eighth Open SID for Speed Class Data Structure. Refer to Table 8-22 for more details.

Table 8-21 : Open SID for Speed Class (OSID-SC) Information Log

Bytes	Description
01:00	**Open SID for Speed Class (OSID-SC):** This field indicates SID for SD Express Speed Class Recording. When the SLBA in each 1st SWC matches to the one designated by "Start Recording", the associated SID in the SWC is newly indicated in this field. 0000h means SID is not assigned for Speed Class Recording. In this case, other parameters in this OSID-SC DS are undefined. Even this field is non-zero, this OSID-SC may not be used for Speed Class Recording (its validity is indicated by Validity Flag for OSID-SC in the same OSID-SC DS). This field is cleared to zero when Release Identifier for this OSID-SC or Release Resources is executed. The "Open SID for Speed Class" is also indicated in the "Get Status Data Structure" of Streams Directive (a set of "Open SID for Speed Class" is a subset of "Get Status Data Structure").
02	**Validity Flag for OSID-SC:** This field indicates the validity of OSID-SC, that is, whether this OSID-SC is available for Speed Class Recording or not. Bits 7:1 are reserved. If Bit 0 indicates '1', Speed Class Recording for corresponding OSID-SC is valid. If cleared to '0', it is invalid.
03	Reserved
05:04	**Submission Queue ID for OSID-SC:** This indicates the oldest Submission Queue Identifier of the command which makes Speed Class Recording by the corresponding OSID-SC invalid due to the termination condition. If the corresponding Validity Flag is set to '1', this field shall be 0000h. This field is cleared to zero when Release Identifier for the associated OSID-SC or Release Resources is executed.
07:06	**Command ID for OSID-SC:** This indicates the oldest Command Identifier of the command which makes Speed Class Recording by the corresponding OSID-SC invalid due to the termination condition. If the corresponding Validity Flag is set to '1', this field shall be 0000h. This field is cleared to zero when Release Identifier for the associated OSID-SC or Release Resources is executed.

Table 8-22 : Open SID for Speed Class Data Structure (OSID-SC DS)

8.4.10.3.3 SD Express SUS_ADDR (LID=C1h)

Table 8-23 indicates SD Express Speed Class SUS_ADDR (suspension address) log. It is indicated by 512KB units.

Bytes	Description
03:00	**SD Express SUS_ADDR 1:** This field indicates the first suspension address (in a unit of 512KB).
07:04	**SD Express SUS_ADDR 2:** This field indicates the second suspension address.
11:08	**SD Express SUS_ADDR 3:** This field indicates the third suspension address.
15:12	**SD Express SUS_ADDR 4:** This field indicates the fourth suspension address.
19:16	**SD Express SUS_ADDR 5:** This field indicates the fifth suspension address.
23:20	**SD Express SUS_ADDR 6:** This field indicates the sixth suspension address.
27:24	**SD Express SUS_ADDR 7:** This field indicates the seventh suspension address.
31:28	**SD Express SUS_ADDR 8:** This field indicates the eighth suspension address.

Table 8-23 : SD Express SUS_ADDR List Log

A non-zero value indicates a valid suspension address. The value zero means invalid address. Up to 8 valid suspension addresses are listed in ascending order without zero started from "SD Express

SUS_ADDR 1.” If the number of valid suspension addresses is less than 8, the rest of this field is filled by zero.
The suspension address is cleared to zero by either of these events:

(1) The host issues the "Set Free AU/SGS" command to assign an SGS including the suspension address.
(2) The host issues the "Resume AU/SGS" command specifying the suspension address.
(3) The host writes to any location within the suspended SGS unit (that includes suspension address).
(4) The host erases to any location within the suspended SGS unit.

Before issuing "Resume AU/SGS", it is recommended that host gets this list by Get Log Page command.

9. Sections Effective to SD I/F Mode, UHS-II Mode and PCIe Mode

This section is a blank in the Simplified Specification

Appendix A (Normative) : Reference

A.1 Related Documentation

This section is a blank in the Simplified Specification

A.2 Related Documentation From Other Standard Organizations

A.2.1 PCI-SIG :

PCI Express Base Specification Revision 3.1a or later
PCI Code and ID Assignment Specification Revision 1.9
PCI Express M.2 Specification Revision 1.1
PCI Express Card Electromechanical Specification Revision 3.0 or later

PCI-SIG®, PCIe® and PCI Express® are registered trademarks of PCI-SIG.
The PCI-SIG defines the PCI Express standard and specifications.
Contact the PCI-SIG for further information URL: https://pcisig.com/.

A.2.2 NVM Express:

NVM Express Revision 1.3 or later

NVM Express™ (word mark) and NVMe™ (word mark) are trademarks of NVM Express, Inc.
NVM Express, Inc. defines the NVM Express standard and specifications.
Contact NVM Express, Inc. for further information URL: http://nvmexpress.org/.

A.2.3 TCG:

TCG Storage Architecture Core Specification Version 2.01 Revision 1.00 or later
TCG Storage Interface Interactions Specification (SIIS) Version 1.11 (r1.0) or later.

TCG defines TCG standard and specifications.
Contact TCG for further information URL: http://www.trustedcomputinggroup.org/.

A.2.4 RPMB:

INCITS 501-2016, Information technology – Security Features for SCSI Commands (SFSC).
INCITS 513-2015 SPC-4 (T10) and INCITS SPC-6 (T10).
Available from http://webstore.ansi.org

Appendix B (Normative) : Special Terms

B.1 Terminology

active AU	The AU that is currently designated for Video Speed Class recording
assigned AUs	The AUs that have been assigned and not completely written
block	A number of bytes, basic data transfer unit
boot	To start the program that makes host equipment ready
boot code	Set of instructions or software that runs over the host equipment at its start-up
Boot Partition	A specific partition for storing boot code
broadcast	A command sent to all cards on the SD bus
Blocklen	Block Length set by CMD16
Cache	A faster memory in card that may be volatile to store host data temporarily
CA-mode	A Fast Boot starting by issuing CMD0 with special argument
CV-mode	A Fast Boot starting by driving CMD line Low
DIR slot	Card supporting Video Speed Class has eight slots to register locations of DIR write to manage average time of FAT update.
Distributed	A signal path between host and card which has a distributed system effects. As described in transmission line theory.
Extension Register	Register defined by a Function Specification in Extension Register Space.
Extension Register Space	Register Spaces accessible by CMD48/49/58/59 with 17-bit ADDR and FNO. There are two types of spaces: memory space and I/O space.
Fast Boot	A function to obtain boot code faster and earlier. In this document, this means a procedure to get boot code just after power up by minimal operations.
Flash	A type of multiple time programmable non-volatile memory
File System Area	The area comprising Partition table, FAT, Bitmap, Directory Entry, etc.
Fixed Data Window	There is an overlapped area of valid data window for all delay variation.
group	A number of sectors, composite erase and write protect unit
LBA	A Logical Block Address identifies a specific sector
Logical Erase	Erasing the logical address to physical address mapping
Lumped	A signal path between host and card which is considerably small compared to the signal rise time. It is considered as "lumped" system
MBRControl table	A table including parameters to control MBR Shadowing
MBR Shadowing	For TCG supported card, a function to be accessible to TCG MBR Table before pre-boot authentication.
Non CPRM Card	Regular Writeable SD Card (SDSC/SDHC/SDXC/SDUC) that does not support the CPRM security.
Open-drain	A logical interface operation mode. An external resistor or current source is used to pull the interface level to HIGH, the internal transistor pushes it to LOW
payload	Net data
pre-boot authentication	A kind of authentication for accessing the User Area of TCG supported card properly after power up
push-pull	A logical interface operation mode, a complementary pair of transistors is used to push the interface level to HIGH or LOW
SD Express Speed Class	Minimum performance defined in the PCIe mode.
sector	A number of blocks, basic erase unit
Sequential CQ mode	Command Queue mode where tasks are submitted and executed in

	increasing order of task ID from Task 0
Self Maintenance	A method where internal operations are carried out based on Host's enablement
SGS boundary	LBA whose address is the multiple of SGS.
SGS unit	The memory area whose starting address is aligned to the SGS boundary and whose size is same as SGS
Speed Class	Minimum performance defined in Default and High Speed Modes
Speed Grade	Minimum performance defined in UHS-I and UHS-II mode
stuff bit	Filling bits to ensure fixed length frames for commands and responses
suspended AU	An AU that has been addressed by a CMD20 "Suspend AU" command
SWS boundary	LBA whose address is the multiple of SWS.
SWS unit	The memory area whose starting address is aligned to the SWS boundary and whose size is same as SWS
TCG Table	Generic term of tables including basic data structures for TCG security
TCG MBR Table	An area storing code to be processed after power cycle, including pre-boot authentication program. Identical to "MBR Table" defined in the TCG Storage Architecture Core Specification.
three-state driver	A driver stage which has three output driver states: HIGH, LOW and high impedance (which means that the interface does not have any influence on the interface level)
token	Code word representing a command
Tuning	Host adjusts sampling clock by Send Tuning Block Command.
User Area Partition	A partition including User Area
Variable Data Window	An overlapped area of valid data window is not available or too small for all Process, Voltage and Temperature variations.
Video Speed Class	Minimum performance independently defined from Speed Class and Speed Grade
Voluntary CQ mode	Command Queue mode where tasks are submitted in arbitrary order and executed based on ready state indicated by card

B.2 Abbreviations

ACMD6	Set bus width command
ACMD41	Initialization command
AU	Allocation Unit
CDM	Charged Device Model
CID	Card IDentification number register
CLK	clock signal
CMD	command line or SD bus command (if extended CMDXX)
CRC	Cyclic Redundancy Check
CSD	Card Specific Data register
CMD0	Reset command
CMD8	Voltage check command
CMD6	Switch command used for selecting one of UHS-I modes
CMD11	Voltage switch command to change signaling level 3.3V to 1.8V.
CMD19	A new command for sending tuning block
CMP	Completion defined in the NVMe specification
COP	Card Ownership Protection
CQ	Command Queue
DAT or DAT[3:0]	4-bit data line of SD bus
DDR	Double data rate signaling
DDR50	One of UHS modes with double data rate. Up to 50MB/sec at 50MHz
DS	Default Speed Mode

DSM	Dataset Management defined in the NVMe specification
DSR	Driver Stage Register
ECC	Error Correction Code
eSD	Embedded SD Memory Device defined by Part 1 eSD Addendum
ESL	Equivalent Series Inductance
ESR	Equivalent Series Resistance
FD156	UHS-II Full Duplex mode with data transfer rate up to 156MB/s
FD312	UHS-II Full Duplex mode with data transfer rate up to 312MB/s
FD624	UHS-II Full Duplex mode with data transfer rate up to 624MB/s
FEP	Force Erase Password
G3L1	One of SD Express Card Types supporting up to Gen3 and 1 lane
G3L2	One of SD Express Card Types supporting up to Gen3 and 2 lanes
G4L1	One of SD Express Card Types supporting up to Gen4 and 1 lane
G4L2	One of SD Express Card Types supporting up to Gen4 and 2 lanes
HD312	UHS-II Half Duplex with 2 Lanes mode with data transfer rate up to 312MB/s
Host-SDR-FD	One of host types with SDR signaling, fixed-delay (can't use tuning)
Host-SDR-VD	One of host types with SDR signaling, variable-delay (can use tuning)
Host-DDR	One of host types with DDR signaling
HS	High Speed Mode
IF-RECV	An interface command to transmit data from card to host over security protocol
IF-SEND	An interface command to transmit data from host to card over security protocol
IOPS	Input/Output Operations Per Second
ICDS	Identifier Controller Data Structure defined in the NVMe specification
LID	Log Page IDentifier defined in the NVMe specification
LOW, HIGH	Binary interface states with defined assignment to a voltage level
LV50	One of LV card classification type that supports single data rate up to 50MB/sec at 100MHz
LV104	One of LV card classification type that supports single data rate up to 104MB/sec at 208MHz
LV156	One of LV card classification type that supports UHS-II Full Duplex mode with data transfer rate up to 156MB/s. May support Half Duplex mode with data transfer rate up to 312MB/s.
LV624	One of LV card classification type that supports UHS-II Full Duplex mode with data transfer rate up to 624MB/s. May support Half Duplex mode with data transfer rate up to 312MB/s.
MDTS	Maximum Data Transfer Size defined in the NVMe specification
MLCC	Multi-Layer Ceramic Capacitor
MNTMT	Minimum Thermal Management Temperature defined in the NVMe specification
MP	Maximum Power defined in the NVMe specification
MSB, LSB	The Most Significant Bit or Least Significant Bit
MLCC	Multi-Layer Ceramic Capacitor
MTP	Multiple Time Programmable memory
MXTMT	Maximum Thermal Management Temperature defined in the NVMe specification
NCQ	I/O Completion Queue defined in the NVMe specification
N_{ERASE}	The recommended numbers of AUs to be erased in one erase operation.
T_{ERASE}	Timeout value used for erasing multiple AU's as specified by ERASE_SIZE.
T_{OFFSET}	Offset time used for calculating erase timeout.
NLB	Number of Logical Blocks defined in the NVMe specification
NR	Number of Ranges defined in the NVMe specification
NSQ	I/O Submission Queue defined in the NVMe specification
NSAC	Defines the worst case for the clock rate dependent factor of the data access time

OCR	Operation Conditions Register
OSCSC	Open SID Count for Speed Class
OSID-SC	Open SID for Speed Class
OSID-SC DS	Open SID for Speed Class Data Structure
OTP	One Time Programmable memory
P_w	Performance of Write
P_m	Performance of Move
P_r	Performance of Read
PDN	Power Delivery Network
PRw	Performance of Random Write
PRr	Performance of Random Read
PS	Power State defined in the NVMe specification
PSD	Power State Descriptor defined in the NVMe specification
PSSw	Performance of Sustained Sequential Write
RCA	Relative Card Address register
ROM	Read Only Memory
RPMB	Replay Protected Memory Block
RU	Recording Unit
SCPS	Speed Class Power State
SCTMT1	Speed Class TMT1
SCTMT2	Speed Class TMT2
SDCLK	Clock line of SD bus
SGS	Stream Granularity Size defined in the NVMe specification
S18R	Switching to 1.8V Request in ACMD41 argument
S18A	Switching to 1.8V Accepted in ACMD41 response
SID	Stream IDentifier
SIIS	Storage Interface Interactions Specification
SLBA	Starting LBA defined in the NVMe specification
SPI	Serial Peripheral Interface
SPL	Slot Power Limit defined in the PCIe and NVMe specification
SRC	Stream Read Command
SU	Sub Unit
SUS_ADDR	SUSpension ADDRess
SWC	Stream Write Command
SWS	Stream Write Size defined in the NVMe specification
TAAC	Defines the time dependent factor of the data access time
tag	Marker used to select groups or sector to erase
TBD	To Be Determined (in the future)
TCG	Trusted Computing Group
T_{fw}	FAT write time
T_{fr}	FAT read time
TMT1	Thermal Management Temperature 1 defined in the NVMe specification
TMT2	Thermal Management Temperature 2 defined in the NVMe specification
t_{ODLY}	Output Delay from SDCLK under all delay parameters condition.
UHS	Ultra High Speed
UI	Unit Interval is one bit nominal time, SDCLK nominal period.
SD Bus I/F	Interface using contact pin numbers 1 to 9.
SD Express	SD Card that includes PCIe/NVMe interface
SDR	Single data rate signaling
SDR12	One of UHS-I modes with single data rate. Up to 12.5MB/sec at 25MHz
SDR25	One of UHS-I modes with single data rate. Up to 25MB/sec at 50MHz
SDR50	One of UHS-I modes with single data rate. Up to 50MB/sec at 100MHz
SDR104	One of UHS-I modes with single data rate. Up to 104MB/sec at 208MHz

UHS50	One of UHS-I Card Types supporting SDR50
UHS104	One of UHS-I Card Types supporting SDR104
UHS156	UHS-II Generation 1 Card Type supporting FD156 and HD312 (Optional)
UHS624	UHS-II Generation 2 Card Type supporting FD624
UHS-II I/F	Interface using contact pin numbers 7 to 8 and 10 to 17.
UUID	Universally Unique IDentifier
VCA	Card accepted voltage range
VHS	Host supplied voltage range
V_{DD}	+ power supply of non UHS-II Card
V_{DD1}	3.3V range power supply for UHS-II Card and SD Express Card
V_{DD2}	1.8V range power supply for UHS-II Card and SD Express Card
V_{DD3}	1.2V range power supply for SD Express Card (Second row)
VSC	Video Speed Class
V_{SS}	Power supply ground
VWC	Volatile Write Cache defined in the NVMe specification
WCTEMP	Warning Composite TEMPerature threshold defined in the NVMe specification
X5R/X7R	Symbol for dielectric material of capacitors

Appendix C (Informative) : Examples for Fixed Delay UHS-I Host Design

This section is a blank in the Simplified Specification

Appendix D : UHS-I Tuning Procedure

This section is a blank in the Simplified Specification

Appendix E :
Host Power Delivery Network (PDN) Design Guide

This section is a blank in the Simplified Specification

Appendix F :
Application Notes of Extension Function

F.1 Identification of Function Driver

There are two types of function drivers. "Standard Driver" controls a Standard Function, which will be defined by a Function Specification and it will be provided by OS or Host System vendor. "Particular Driver" controls a Particular Function and it will be provided by the function developer.

During a function initialization, Host Driver finds and loads a most suitable function driver installed on Host System referring to 4 codes in the General Information.

SFC	(Standard Function Code)	2 byte
FCC	(Function Capability Code)	2 byte
FMC	(Function Manufacturer Code)	2 byte
PFC	(Particular Function Code)	2 byte

Standard Driver is selected by SFC and FCC. SFC>0 and FCC=0 means that there is a unique function driver for a SFC and Host Driver finds the driver by only SFC. SFC>0 and FCC>0 means that there are multiple of function drivers to a SFC. These Card Drivers will be installed to Host System with "Capability Information" which corresponds to FCC. Host Driver selects one driver of which Capability Information accords with FCC. SFC=0 means that there is no Standard Driver and then Host Driver finds a Particular Driver which accords with FMC and PFC.
Table F-1 shows combination of the codes to identify a function driver. If SFC>0, FMC>0 and PFC>0, the function may use Standard Driver and Particular Driver depends on driver installation to Host System. Use of Particular Driver is higher priority for supporting higher functionality than use of Standard Driver

SFC	FCC	FMC	PFC	Selection of Function Driver
Non-zero	0000h	0000h	0000h	Select a Standard Driver by only SFC
Non-zero	Non-zero	0000h	0000h	Host selects one of Drivers by FCC
0000h	0000h	Non-zero	Non-zero	Select Particular Function Driver by FMC and PFC
Non-zero	any value	Non-zero	Non-zero	Select either Particular or Standard Function Driver Particular Driver has higher priority

Table F-1 : Combination of Codes to Identify a Function Driver

F.2 Concept of Event Detection Method

To use Event Indication Method is defined in Section 5.7.5, host support is required for event detection and function driver management. Host Controller or Host Driver detects FX_EVENT generation and then interrupt hander calls a Function Driver to deal with the event.

F.2.1 Role of Driver Modules

Figure F-1 shows an example configuration of hardware and drivers layers. Some of functions may generate events.

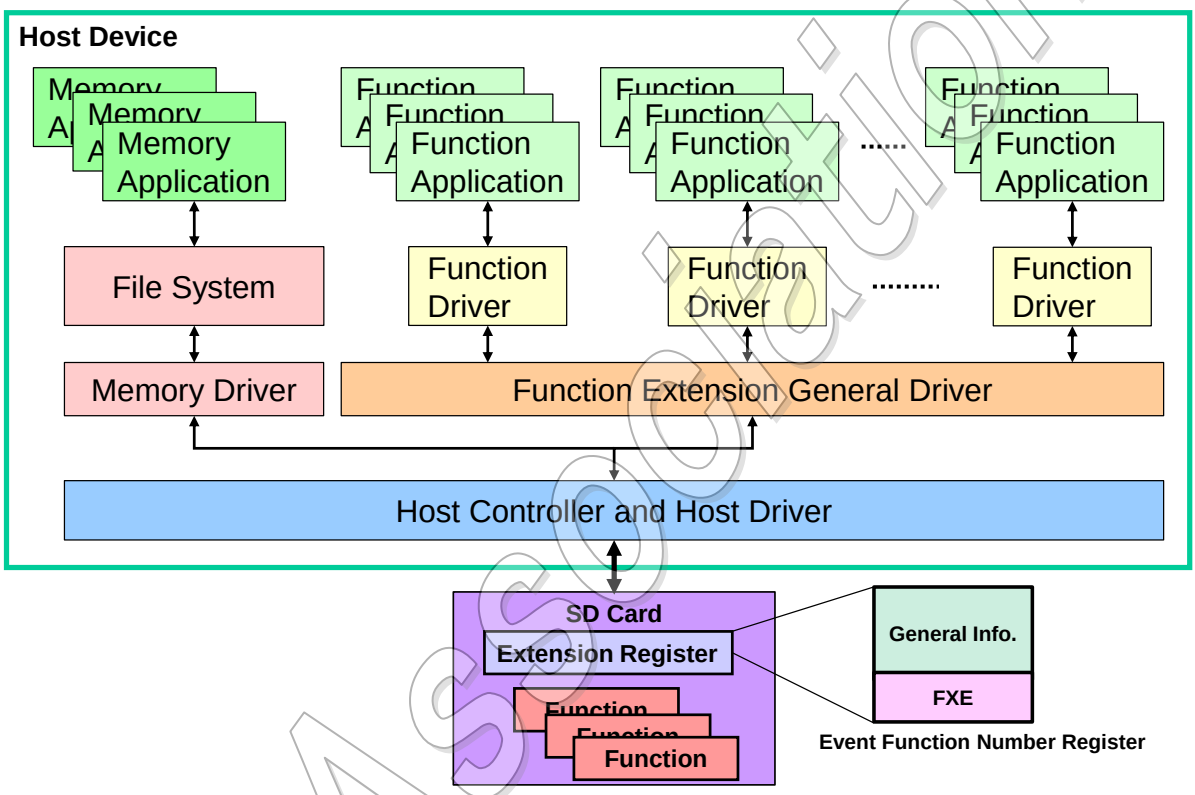


Figure F - 1 :
Figure F-1 : Hardware and Driver Layer of Host and Card

- **Host Controller and Host Driver**
  Host Controller is an interface device between Host System (System Memory) and SD Card that is connected to system bus. Host Driver is software for controlling the Host Controller. Host Driver manages to issue SD Commands to SD Card according to requests from Memory Driver and Function Drivers by accessing Host Controller Registers. Host Controller has capability to generate interrupt to host CPU according to the response of the SD Card. FX_EVENT on R1 is one of interrupt events by Function Devices. On detecting FX_EVENT, Host Driver gets "Function Extesion General Driver" to deal with the event.
- **Memory Driver**
  SD Memory Card Driver to manage memory access to or from SD Memory Card. Indication of FX_EVENT is not supported by this driver.
- **Function Driver**
  This driver knows how to control a function and generates commands sequece to control Extesion Registers according to requests of application. This driver knows how to deal with the evets of the function. Funciton Driver will be provided by card vendor.

- **Function Extesion General Driver (FEGD)**
  During SD Card initialization, this driver finds and loads a Function Driver installed on Host System for controlling a function which is supported by the SD Card. This driver manages communication between multiple of Function Drivers and Host Driver. On detecting FX_EVENT informed by Host Driver, this driver reads FXE Register Set and determines which Function Driver to deal with the event.

F.2.2 Host Implementation to use Event Detection Method

- **Card Initialization**
  When Host Controller detects SD Card, Host Driver will initialize memory portion at first. By detecting Extension Function support on the card, Host Driver tries to find and load Function Drivers installed in the Host System by referring to the General Information. If a Function Driver can be loaded, it is connected to "Function Extension General Driver", which can communicate multiple of function drivers. Host Driver is connected to the "Function Extension General Driver". Standard Driver Interface is assumed to communicate between two driver layers.

- **Event Detection**
  While Host Controller issues SD Commands to the card, host may check event generation by R1 response of any command without issuing extra-commands for polling. While there is no command to be issued to the card, Host Driver inserts CMD13 for polling events at some interval.
  By reading FXE Register Set, Function Extension General Driver determines priority of event handling and get a Function Driver to deal with the event.

Appendix G : Application Notes for Application Performance Class Hosts

G.1 Check for Application Performance Class support

For cases that hosts care about random performance, it is recommended for host to check for their desired Application Performance Class as following:

1. Check what is the Application Performance Class type of card. If it is matching the class required by the host, the card may be approved.

2. In case Application Performance Class is not indicated, the host may perform internal benchmark test and check for the absolute execution time to qualify the card. If it is matching or better than the execution time expected from the corresponding Application Performance Class expected by the host, the card may be approved.

In Android devices that performs card adoption process and the Application Performance Class test is not adopted yet by core Android, it would be highly recommended to add the above two steps before the legacy Android benchmark test process.

The given test method may serve hosts as a simple way for host to filter cards in the field, however Cards that declares Application Performance Class support shall meet the SD standard specification for Application Performance Class conditions as defined in Section 4.16.

Appendix H (Informative) : Application Notes related to SD Express cards

This section is a blank in the Simplified Specification

Appendix I (Informative) : Supply Voltage(s) Generation

This section is a blank in the Simplified Specification

Appendix J (Informative) : Pad 19 Existence Detection

This section is a blank in the Simplified Specification

Appendix K (Informative) : Initialization Sequence Related to Boot Functions, TCG Security and RPMB

In Physical Layer Specification Version 9.00, Boot Functionalities, TCG security and RPMB are introduced. This Appendix shows sample initialization sequences for host to use these functions over SD interface.

K.1 Sample Sequence 1

Figure K-1 shows a sample initialization sequence when host updates a boot code for TCG enabled card without Fast Boot.

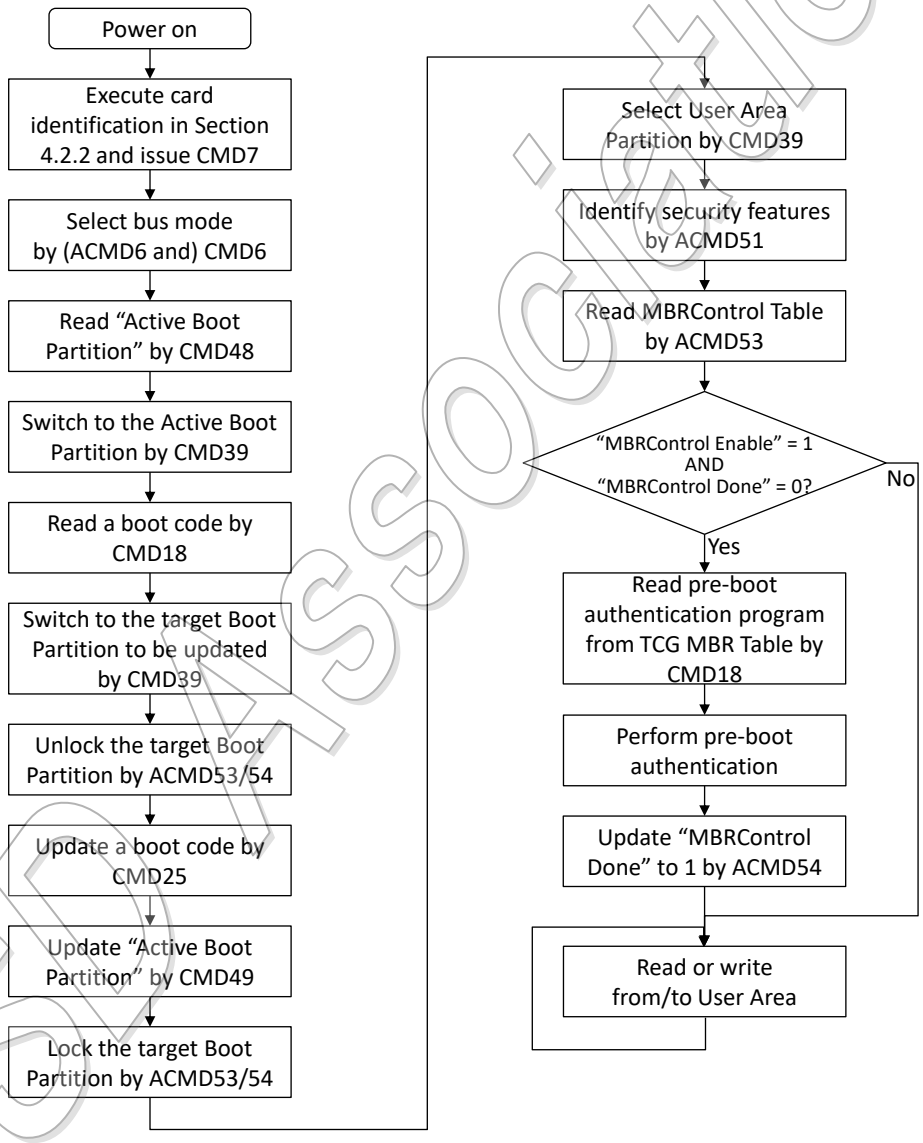


Figure K-1 : Sample Initialization Sequence for Updating a Boot Code and Accessing TCG Enabled Card

K.2 Sample Sequence 2

Figure K-2 shows a sample initialization sequence when host starts up TCG enabled card with Fast Boot.

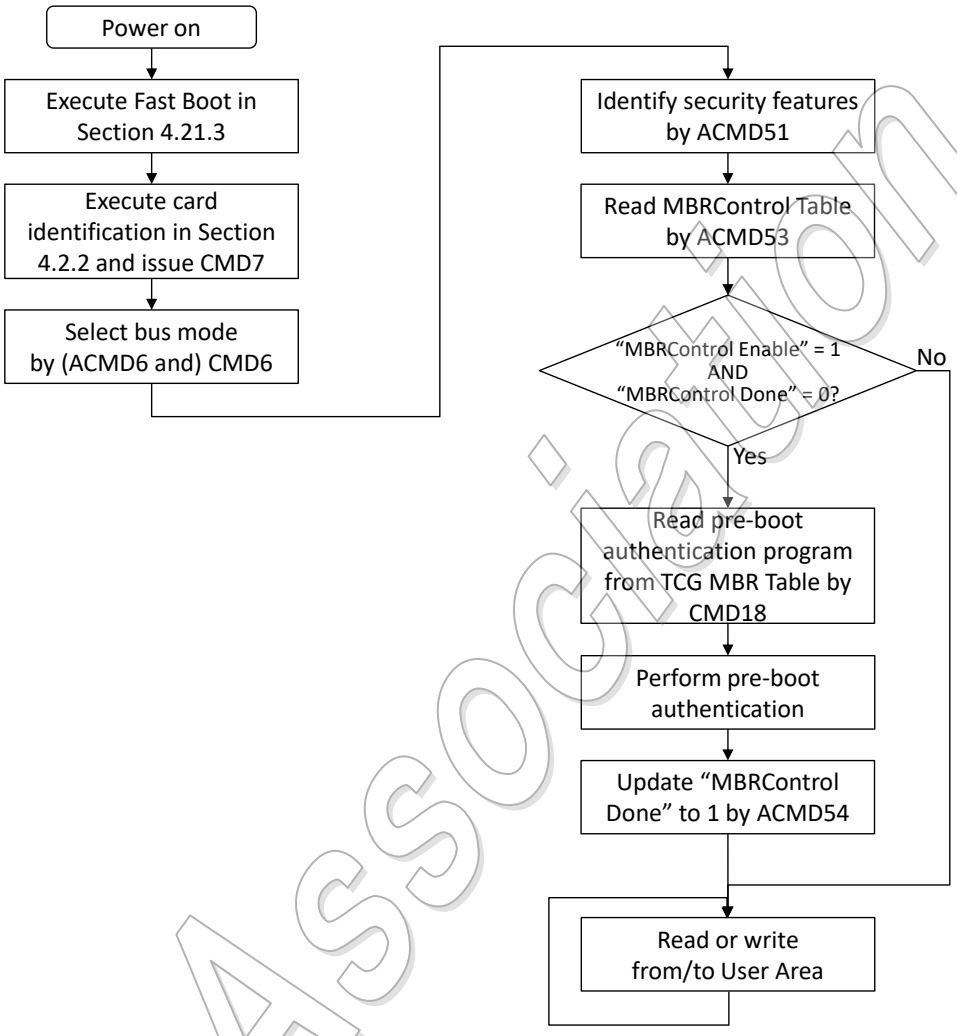


Figure K-2 : Sample Initialization Sequence to for Starting Up TCG Enabled Card with Fast Boot

Appendix L : Simplified Mechanical Drawings

These Simplified Mechanical Dimensions are not described in the original Physical Layer Specification. They are created out of the Mechanical Addenda.

L.1 Standard Size SD Card Simplified Dimensions

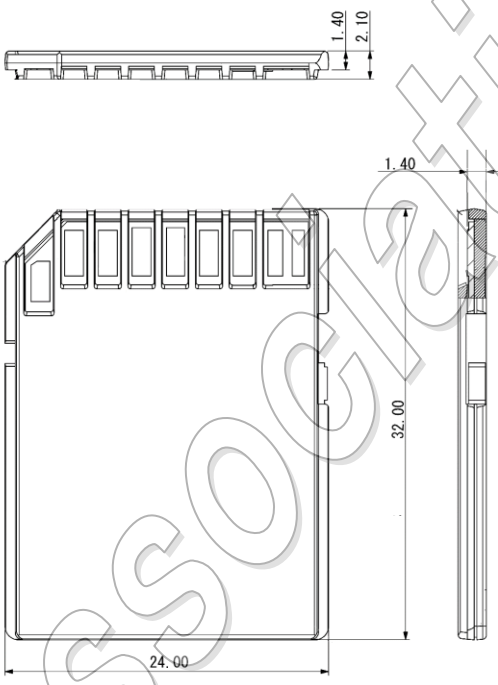


Figure L-1 : Standard Size SD Card Simplified Dimensions

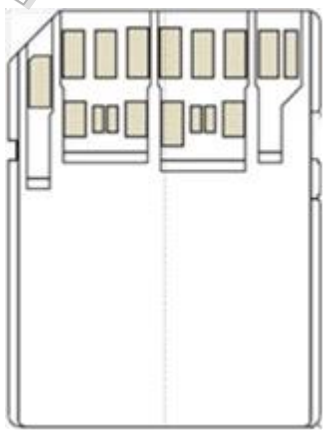


Figure L-2 : Standard Size SD UHS-II and SD Express G3L1 (Gen3 1 lane) Card Simplified Pads side (same dimensions as standard SD Card)

Figure L-3 : Standard SD Express G4L1 (Gen 4 1 lane) Card simplified view of pads side

Figure L-4 : Standard SD Express G3L2 (Gen3 2 lanes) or G4L2 (Gen 4 2 lanes) Card simplified view of pads side

L.2 microSD Card Simplified Dimensions

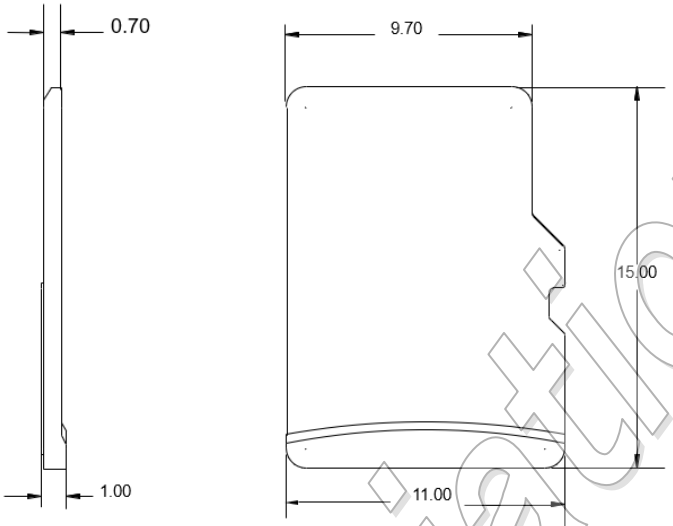


Figure L-5 : microSD Card Simplified Dimensions

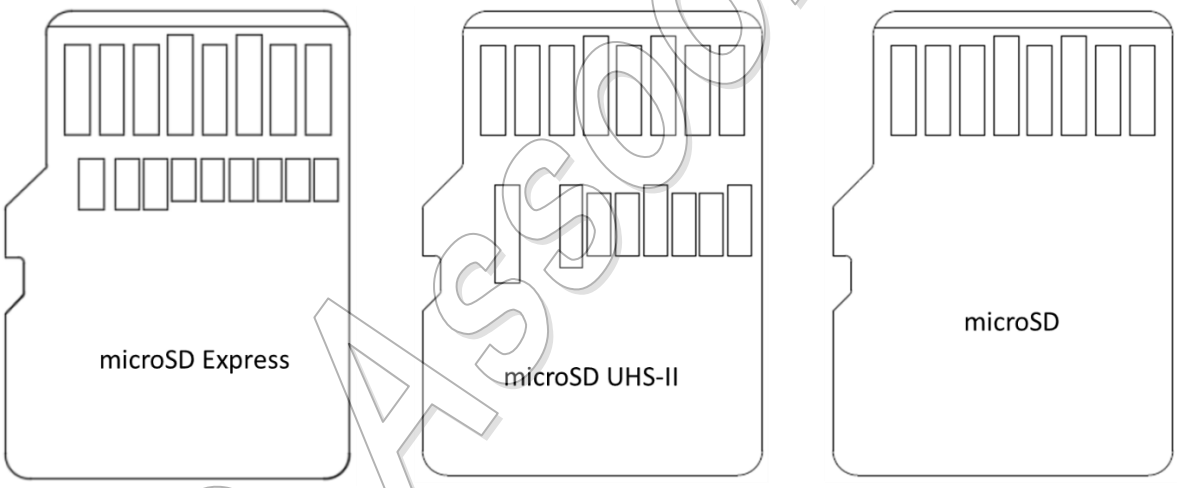


Figure L-6 : microSD Card Simplified Dimensions Pads side