- **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 4-46: AU_SIZE Field

The maximum AU size, depends on the card capacity, is defined in Table 4-47. The card can set any AU size specified in Table 4-46 that is less than or equal to the maximum AU size. The card should set smaller AU size as much as possible.

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

Table 4-47: 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.

- **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 4-48: Erase Size Field

DJI-1019, Part 2
IPR2023-01106

- **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 (6). Refer to 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 4-49: Erase Timeout Field

- **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 4-57 and Equation (6) in 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 4-50: Erase Offset Field

SD Association

- **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 – Fh	Reserved

Table 4-51: UHS_ SPEED_GRADE Field

- **UHS_AU_SIZE**

  This 4-bit field indicates AU Size for 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 in UHS-I or UHS-II bus speed modes.

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 4-52: UHS_AU_SIZE Field

4.11 Memory Array Partitioning

This section is a blank in the Simplified Specification.

SD Association

4.12 Timings

This section is a blank in the Simplified Specification.

SD Association

4.13 Speed Class Specification

The Speed Class Specification classifies card performance by Speed Class number and offers a method to calculate performance. The specification enables the host to support AV applications to perform real time recording to an SD memory card. The following sections describe the Speed Class specification for the card. Refer to the Application Notes for an example of host implementation.
Figure 4-46 shows an overview of the Speed Class Specification. Class 2, 4 and 6 are defined and achieve in Default Speed Mode while Class 10 requires High Speed Mode. Speed Grade 1 is defined in UHS-I and UHS-II modes.

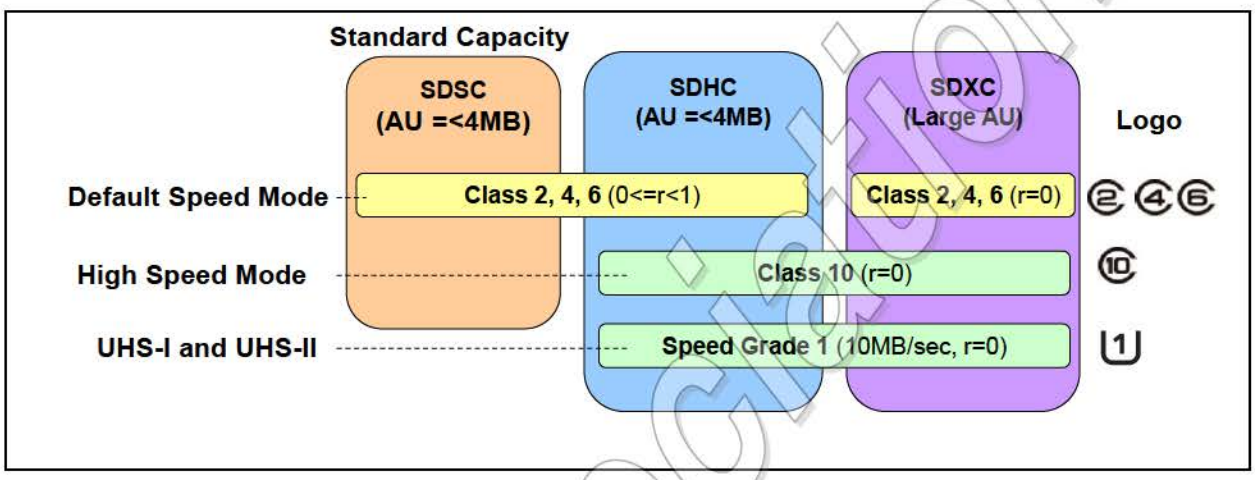


Figure 4-46: Overview of Speed Class Specification

Section 4.13.1 describes Speed Class Specification for SDSC and SDHC.
Section 4.13.2 describes Speed Class Specification for SDXC.
Section 4.13.3 describes Speed Grade Specification for UHS-I and UHS-II Card.

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.

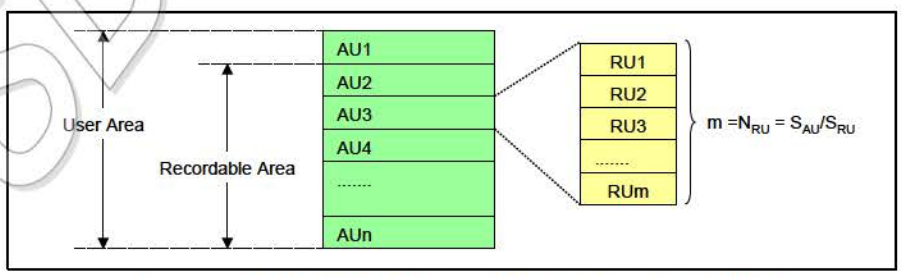


Figure 4-47: Definition of Allocation Unit (AU)

SD Association

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 16KBye. 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).

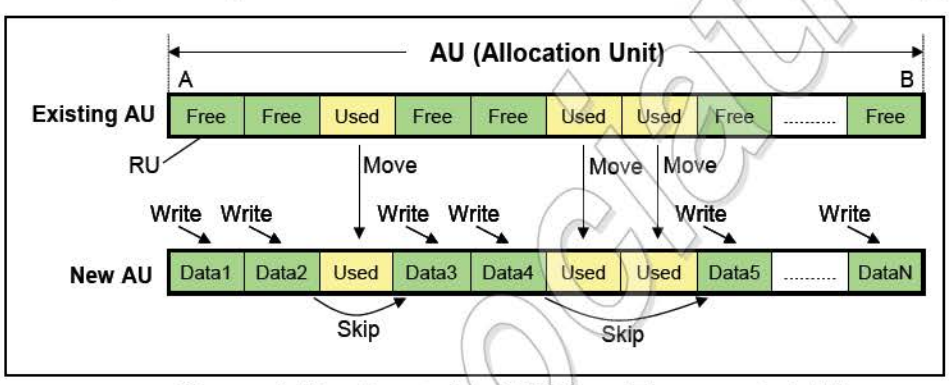


Figure 4-48 : Example of Writing Fragmented AU

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).

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

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 4-57 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.

SD Association

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 4-57 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.

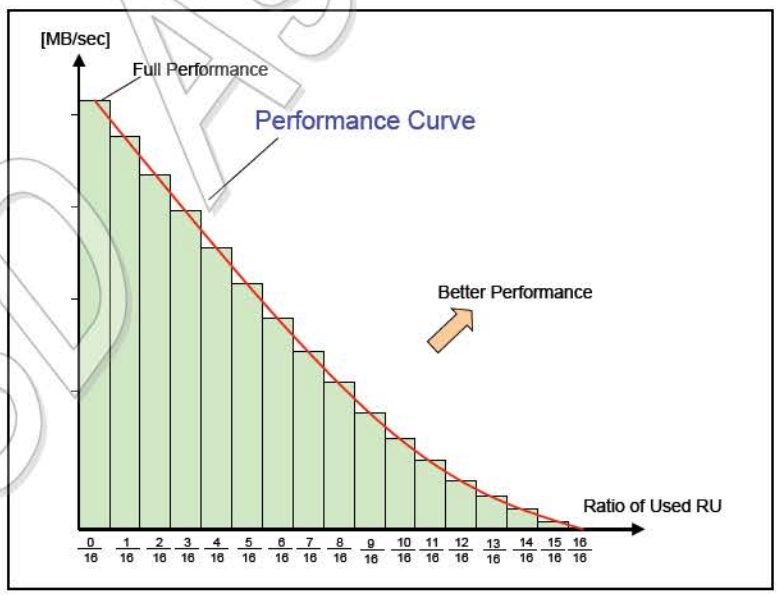


Figure 4-49: Card Performances between 16 RUs

SD Association

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: $$P(r) = \frac{(1-r)PwPm}{rPw + (1-r)Pm} \qquad (0 \le r \le 1)$$ .....(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.

SD Association

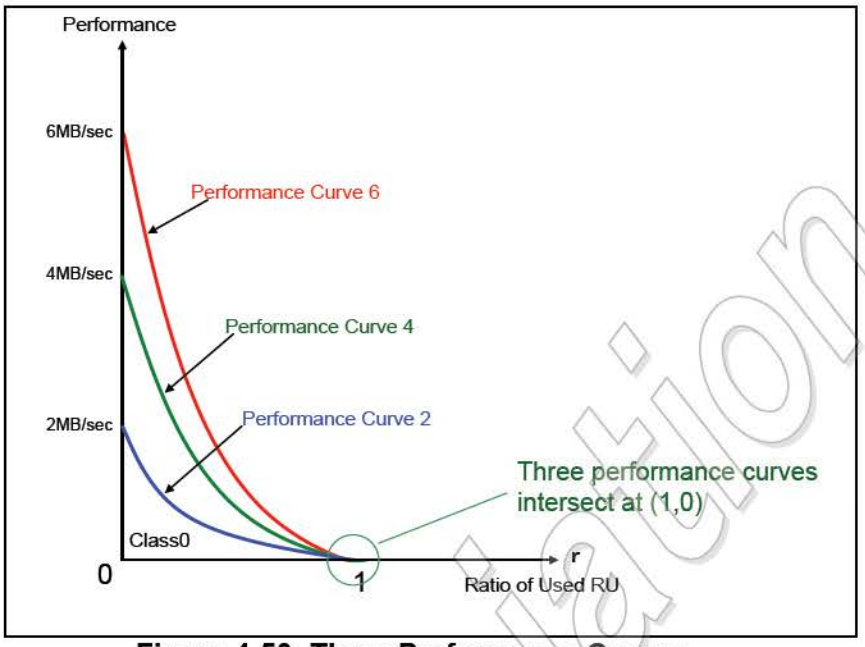


Figure 4-50: Three Performance Curves

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.

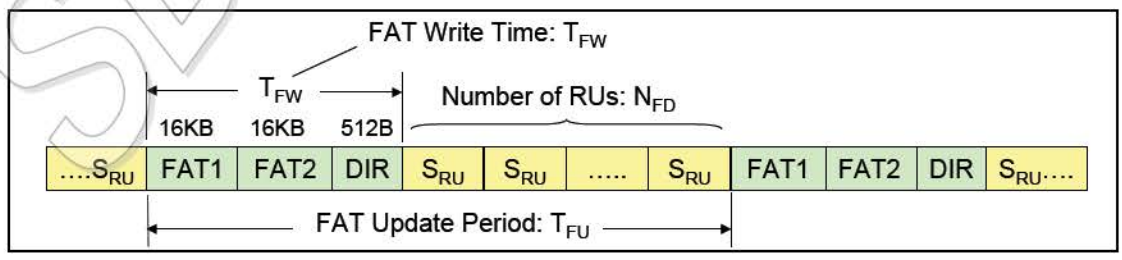


Figure 4-51: Typical Sequence of FAT Update

SD Association

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 the 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 4-56 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 4-56: 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 4-57 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.

SD Association

	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 4-57: 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.

SD Association

4.13.2 Speed Class Specification for SDXC

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

(1) The table of valid AU sizes is updated with five 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.
(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 2TB and the UHS high speed interface require larger AU sizes.
In the case of SDXC 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 4-58. 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~2TB) 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.

SD Association

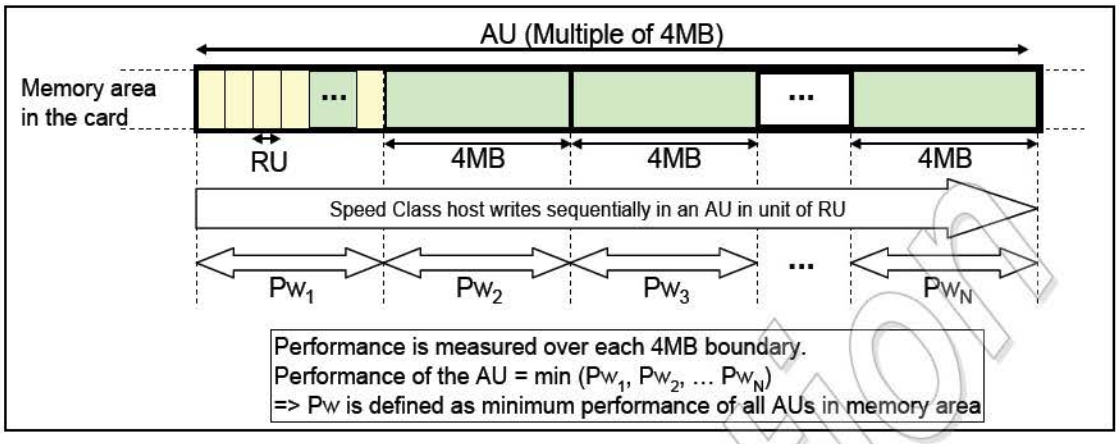


Figure 4-52 : Measurement of Pw (AU size is larger than 4MB)

4.13.2.2.2 Performance Move

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

SD Association

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 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 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

4.13.2.7.1 Measurement Conditions

The measurement conditions of Speed Class for SDXC are defined in Table 4-58. Class 10 card shall meet Class 6 performance under Class 6 condition.

Card Capacity		SDXC
		Over 32GB ~ 2TB
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 4-58 : Measurement Conditions of Speed Class (SDXC)

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 4-59 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 4-59 : Performance Requirements for Each Class (SDXC)

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 File System Specification Version 3.00. This includes complying with the format parameter calculation specified in the Appendix C of the File System Specification Version 3.00.

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 4-54 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.

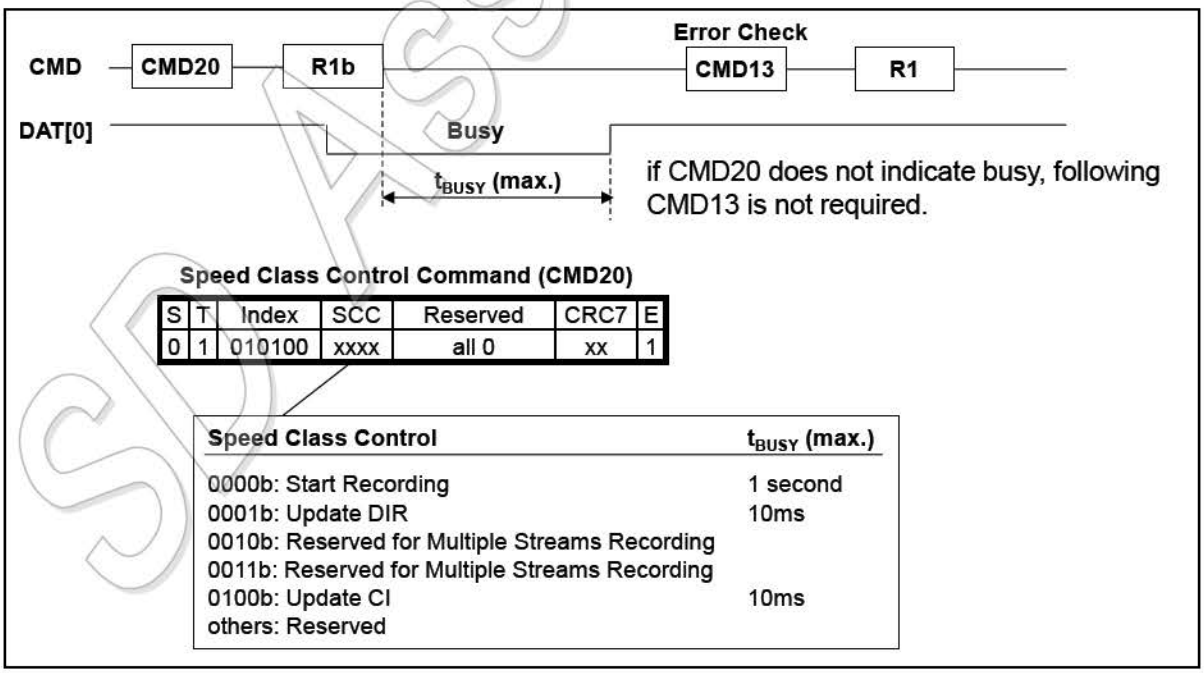


Figure 4-53 : Definition of CMD20

Support of CMD20 is mandatory for SDXC card and optional for SDHC card.
SDXC cards can meet Class performance when data is written after receiving CMD20 indicating the Start Recording function. SDHC cards can meet Class performance without CMD20.

4.13.2.8.1 Definition of Each 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 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 4-55 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 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.

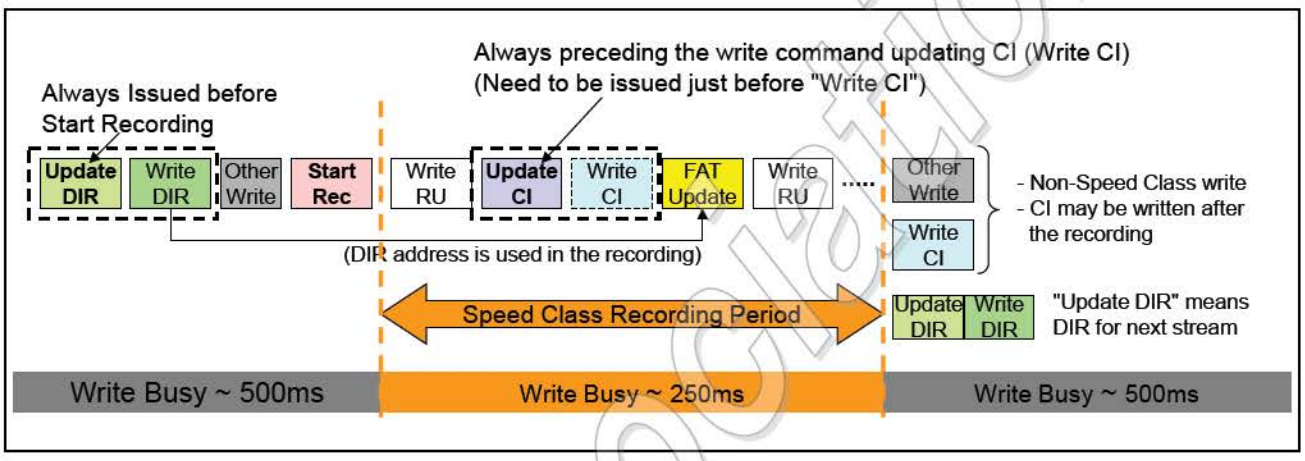


Figure 4-55 : Example of Speed Class Recording

SD Association

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 and SDXC. In case of UHS-II card supports Speed Grade 1, both UHS-I mode and UHS-II mode shall support Speed Grade 1.

(1) Speed Grade 0: Performance is less than 10MB/sec
(2) Speed Grade 1: Performance is 10MB/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. 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. Refer to Section 4.13.2.3.

4.13.3.1.7 UHS-II Parameters

Host may use any N_FCU size to use Speed Grade 1.
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 1 supported Host and Card, support of CMD20 is mandatory for SDXC and optional for SDHC. In order to maintain backward compatibility, Speed Grade 1 SDHC cards shall support Speed Class and Speed Grade 1 without the host indication of CMD20.

SD Association

4.13.3.3 Speed Grade Measurement Conditions

Speed Grades Measurement Conditions are defined in Table 4-60.

Item	Definition
Pw Measurement Conditions for UHS-I	Performance is measured the same as Pw under following conditions. SDCLK=80MHz for SDR and 40MHz for DDR RU=512KB, Power Limit=1.44W (400mA at 3.6V) An AU Size is used described in UHS_AU_SIZE (Up to 64MB). The measurement method of UHS-I performance is the same as SDXC. 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). RCLK=35MHz, Transmission Speed Range=Range A (x15 Multiplier) In FD Mode and Fast Mode. N_FCU=1. The measurement method of UHS-II performance is the same as SDXC. 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 card may be up to 500ms

Table 4-60 : 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 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 to use Speed Grade. 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.

SD Association

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 4-56 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 4-56 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.

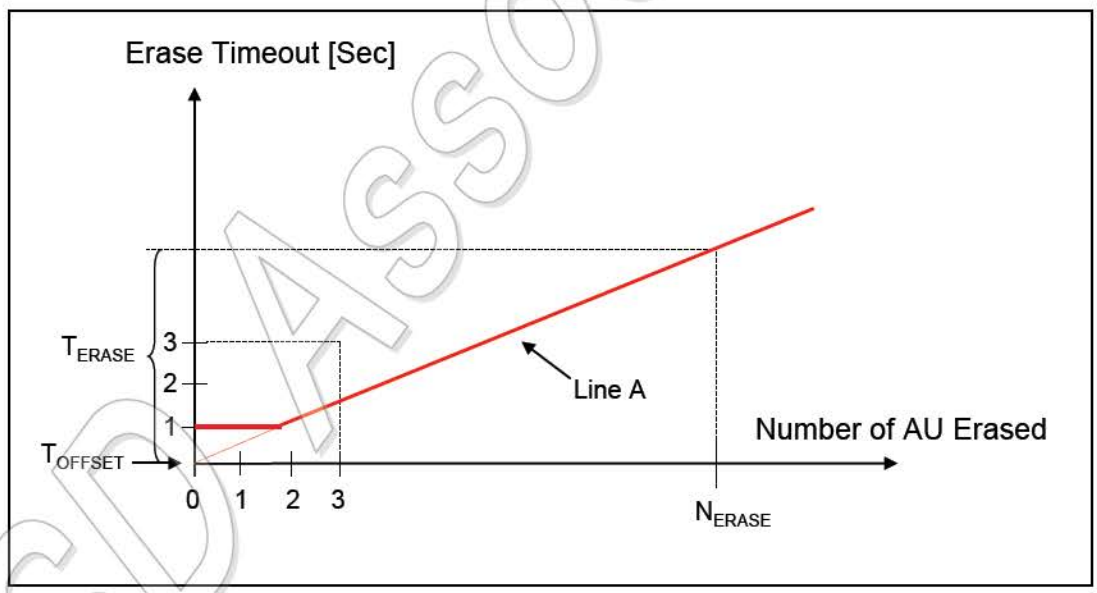


Figure 4-56: Example Erase Characteristics (Case 1 TOFFSET=0)

SD Association

The line B illustrated in Figure 4-57 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.

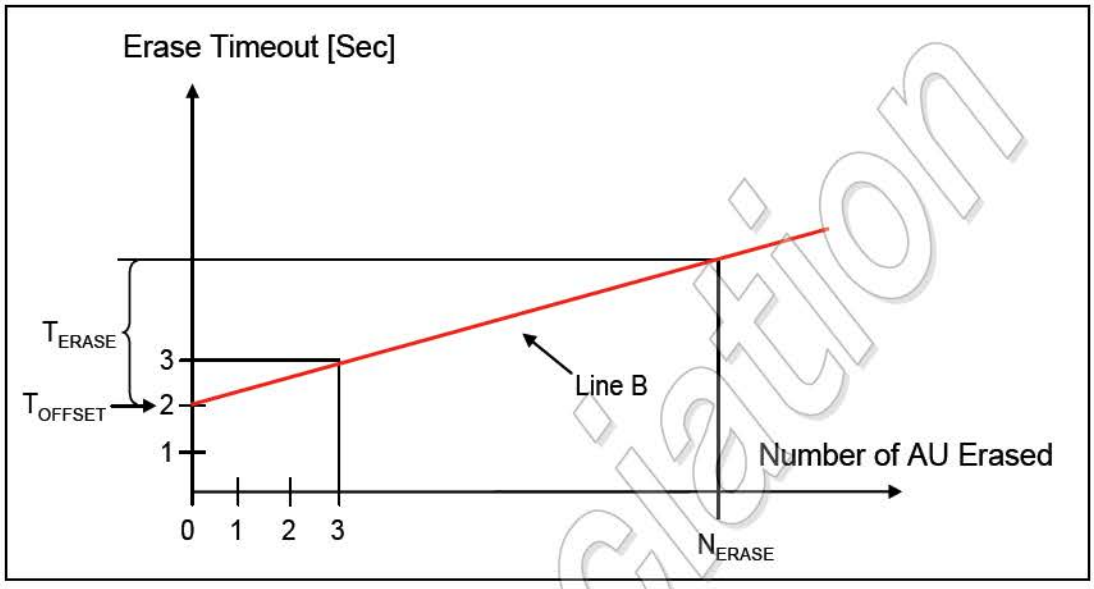


Figure 4-57: Example Erase Characteristics (Case 2 TOFFSET=2)

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 (6).

$$\text{Erase Time-out of X AU} = \frac{T_{ERASE}}{N_{ERASE}} \cdot X + T_{OFFSET} \quad \text{.....(6)}$$

Erase timeout is determined by following steps:
(1) Calculate Equation (6).
(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).

SD Association

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). 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 4-58 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.

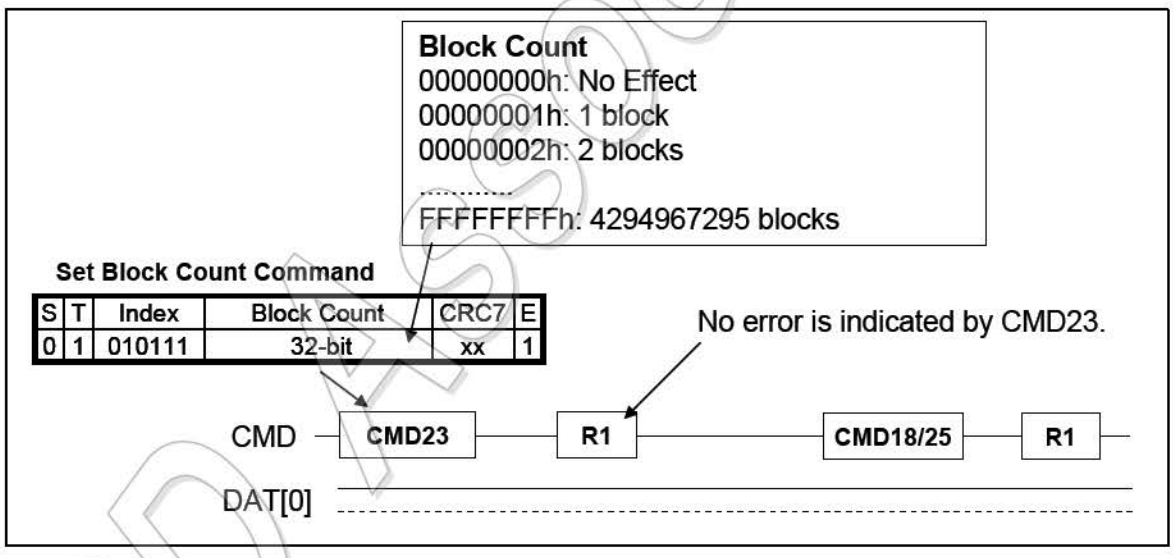


Figure 4-58 : Set Block Count Command

SD Association

©Copyright 2001-2013 SD Group (Panasonic, SanDisk, Toshiba) and SD Card Association

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 Chapter 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 Card.

SD Association

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	**VDD Voltage Window**
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)	
25-28	reserved	
29	UHS-II Card Status	
30	Card Capacity Status (CCS)[1]	
31	Card power up status bit (busy)[2]	

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.

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.

SD Association

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 Panasonic Corporation, SanDisk Corporation and Toshiba 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.

SD Association

- **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 Chapter 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
2-3	reserved	

Table 5-3: CSD Register Structure

SD Association

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]
reserved		2	00b	R/W	[9:8]
CRC	CRC	7	xxxxxxxb	R/W	[7:1]
not used, always'1'	-	1	1b	-	[0:0]

Table 5-4: The CSD Register Fields (CSD Version 1.0)

SD Association

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} 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**

The following table 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.

SD Association

- **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.

SD Association

- **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 Chapter 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

SD Association

- **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.

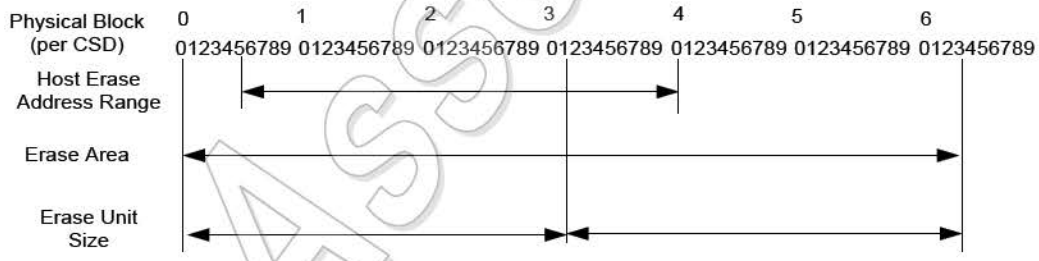


Figure 5-1: ERASE_BLK_EN = 0 Example

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.

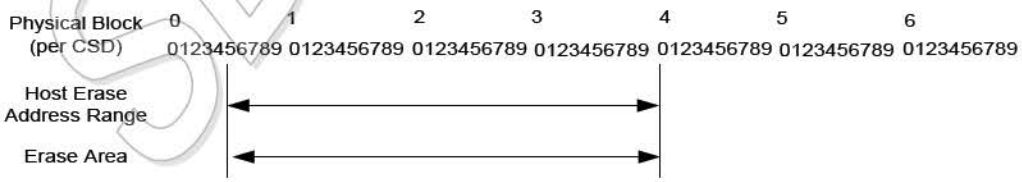


Figure 5-2: ERASE_BLK_EN = 1 Example

SD Association

- **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.

SD Association

- **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.

- **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 Chapter 4.5. The checksum has to be recalculated by the host for any CSD modification. The default corresponds to the initial CSD contents.

SD Association

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	x1x110110101b	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]
reserved	-	2	00b	R	[9:8]
CRC	CRC	7	xxxxxxxb	R/W	[7:1]
not used, always'1'	-	1	1	-	[0:0]

Table 5-16: The CSD Register Fields (CSD Version 2.0)

SD Association

- **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).

SD Association

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).

- **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.

- **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.

SD Association

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 Chapter 0. 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]
Reserved		6	R	[41:36]
Command Support bits	CMD_SUPPORT	4	R	[35:32]
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 1.01-4.00
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 Association

- **SD_SPEC, SD_SPEC3, SD_SPEC4**

The Physical Layer Specification Version is indicated in combination with SD_SPEC, SD_SPEC3 and SD_SPEC4 as described Table 5-19.

SD_SPEC	SD_SPEC3	SD_SPEC4	Physical Layer Specification Version Number
0	0	0	Version 1.0 and 1.01
1	0	0	Version 1.10
2	0	0	Version 2.00
2	1	0	Version 3.0X
2	1	1	Version 4.XX
Others			reserved

(1) Version 2.00 hosts do not recognize SD_SPEC3 and SD_SPEC4.
(2) Version 3.00 hosts do not recognize SD_SPEC4.

Table 5-19: Physical Layer Specification Version

Application Notes:
Hosts recognize Physical Layer Specification Version shall also recognize including future version.
Next version will be defined in SD_SPEC field.

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 and SD_SPEC4=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 and SD_SPEC4=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 and SD_SPEC4=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 and SD_SPEC4=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

SD Association

- Essential conditions to indicate Version 4.XX Card (SD_SPEC=2, SD_SPEC3=1 and SD_SPEC4=1)
  (1) Same as the essential conditions of Version 3.00 device
  (2) Support any of additional functions defined by Version 4.XX:
  Followings 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
  Followings 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

The requirements of supporting commands mentioned above are for the optional commands, the support of which depends on versions (SD_SPEC, SD_SPEC3 and SD_SPEC4). 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

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 it is mandatory for a regular writable SD Memory Card to support Security Protocol. For ROM (Read Only) and OTP (One Time Programmable) types of the SD Memory Card, the security feature is optional.

SD Association

- **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 will be defined by a later version of the Part 3 Security Specification Version 3.00.

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[46:45] is reserved for future extension.

Table 5-22 : Extended Security

- **CMD_SUPPORT**

  Support bit of new commands are defined to Bit 33-32 of SCR.

SCR Bit	Supported Command	Command	CCC	Remark
35	Extension Register Multi-Block	CMD58/59	11	Optional. If CMD58/59 is supported, CMD48/49 shall be supported.
34	Extension Register Single Block	CMD48/49	11	Optional.
33	Set Block Count	CMD23	2, 4	Mandatory for UHS104 card
32	Speed Class Control	CMD20	2, 4	Mandatory for SDXC card

Table 5-23 : Command Support Bits

SD Association

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 Version 4.00 and is applicable from the Physical Version 4.00 products. This function may be supported on SDHC and SDXC Card and is not supported through SPI interface. From the Physical Layer Version 4.10, Multi-Block commands are supported.

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.

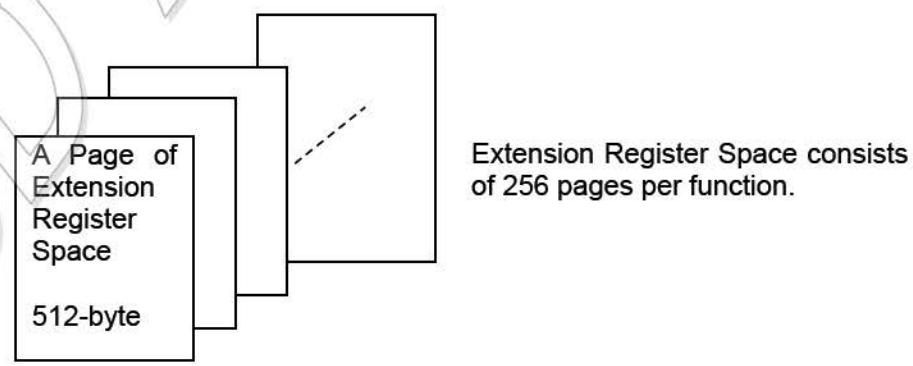


Figure 5-3 : Extension Register Space

SD Association

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.

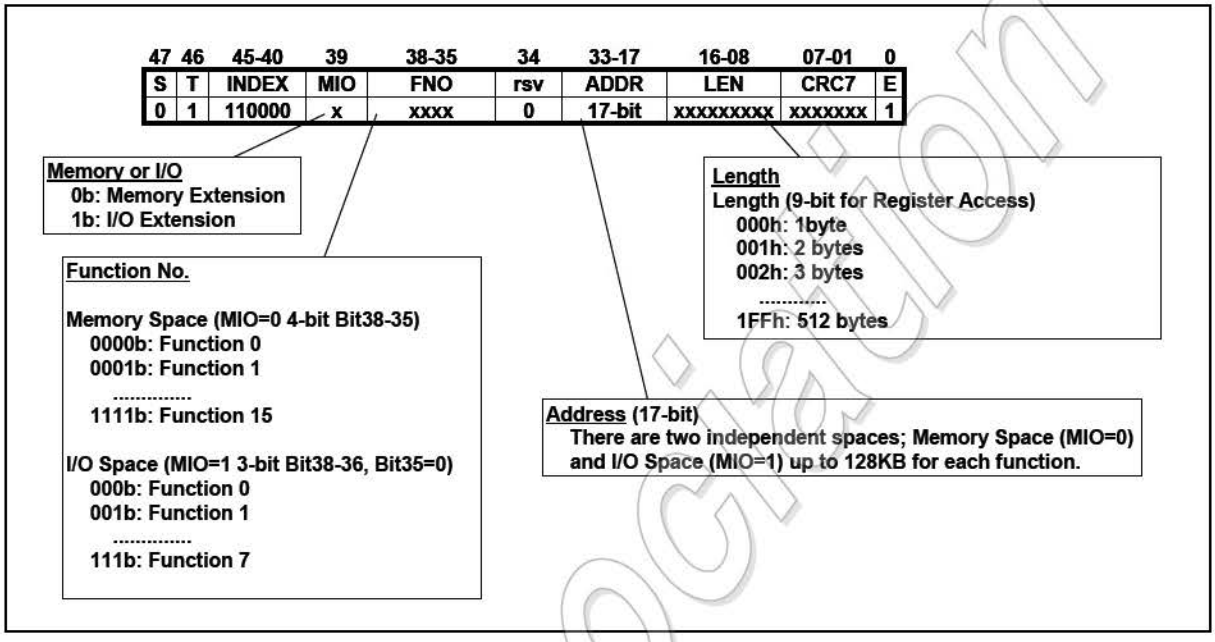


Figure 5-4 : Read Extension Register Single Block Command (CMD48)

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

SD Association

the end bit of CMD48 is 1 second.

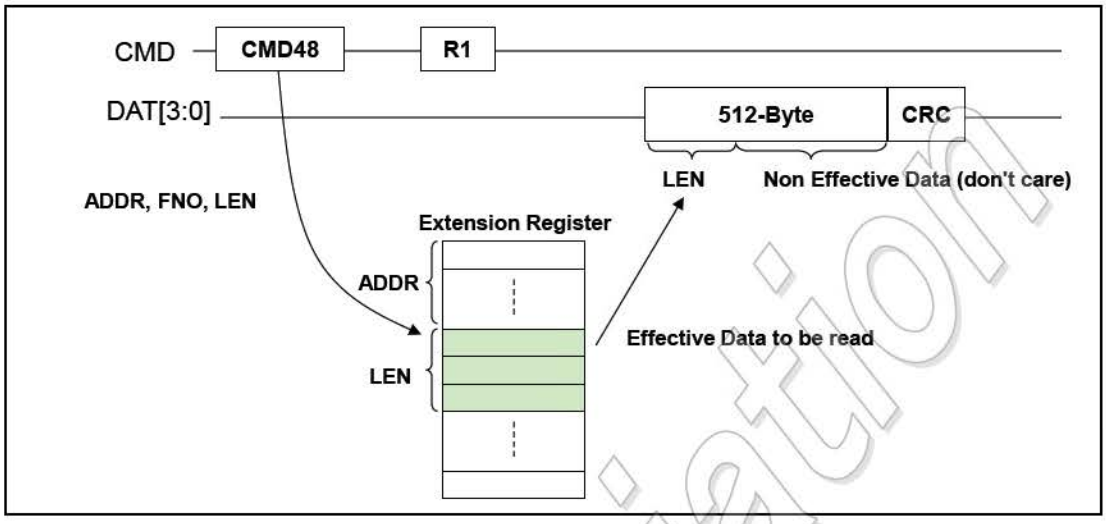


Figure 5-5 : Extension Register Read Operation by CMD48

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.

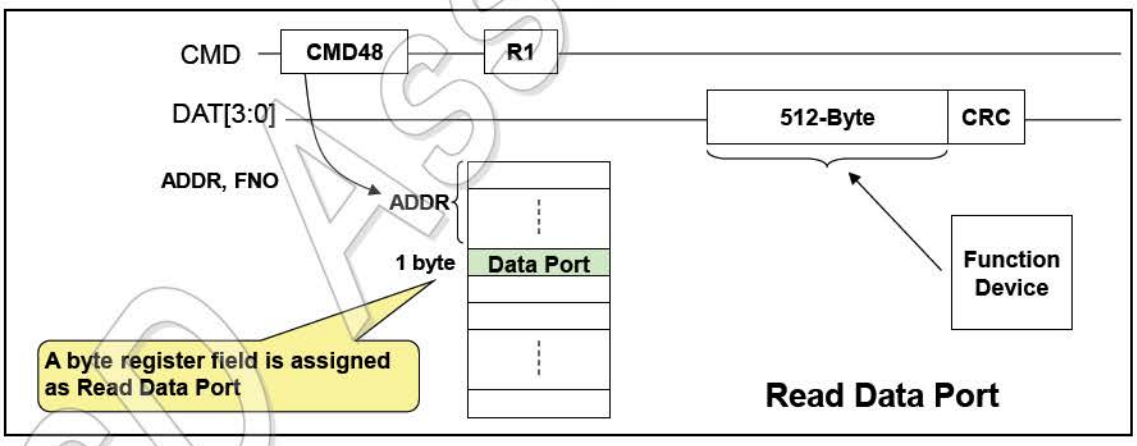


Figure 5-6 : Data Port Read Operation by CMD48

SD Association

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.

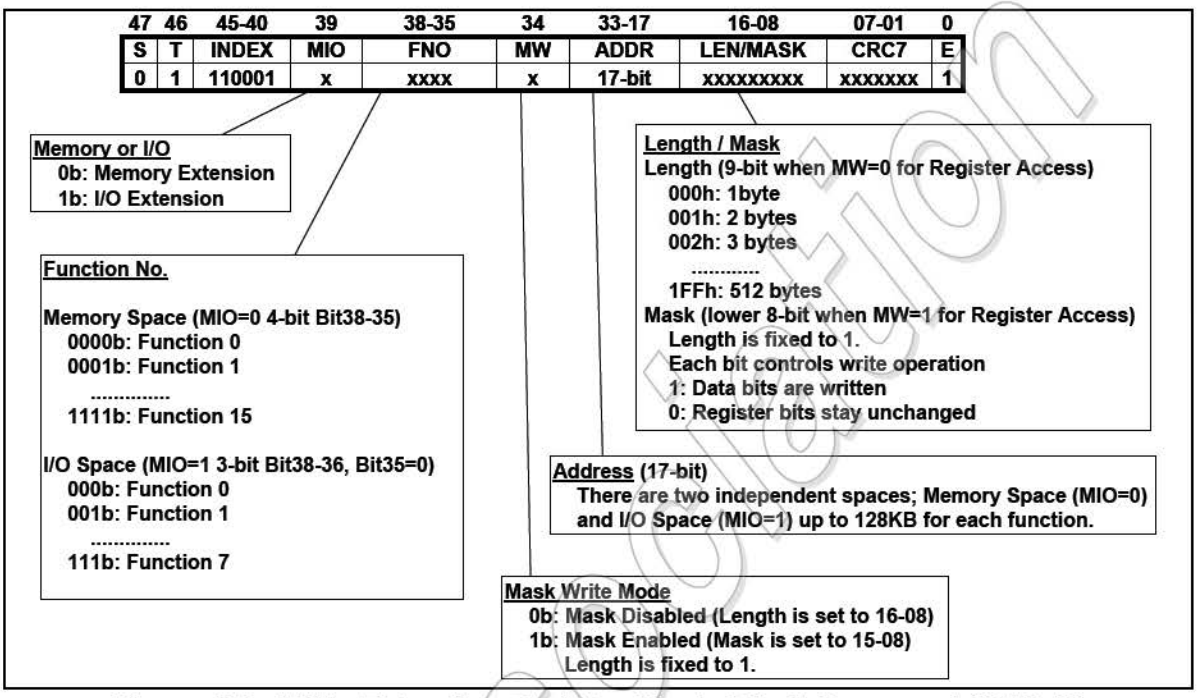


Figure 5-7 : Write Extension Register Single Block Command (CMD49)

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.

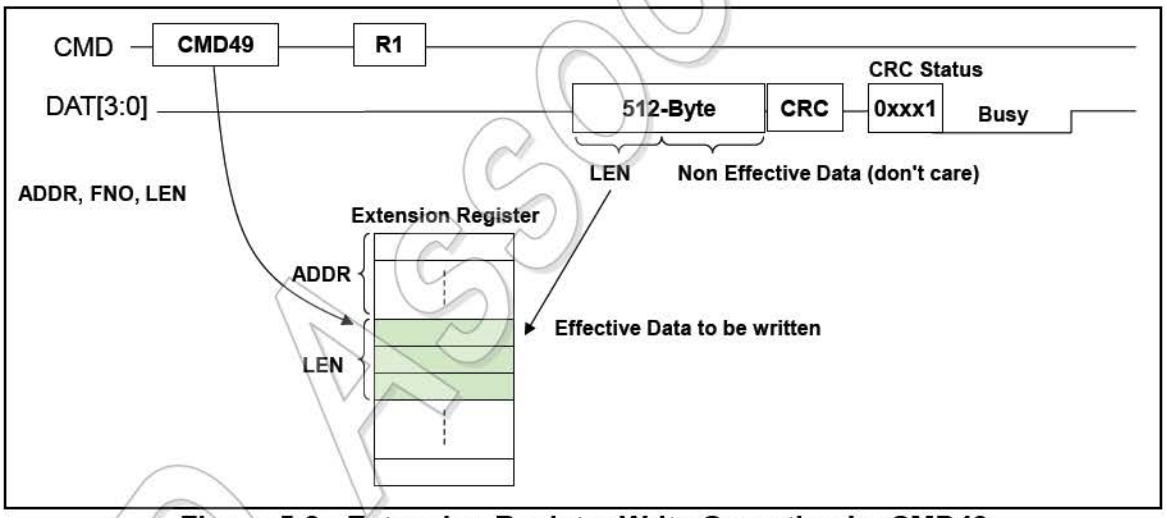


Figure 5-8 : Extension Register Write Operation by CMD49

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.

SD Association

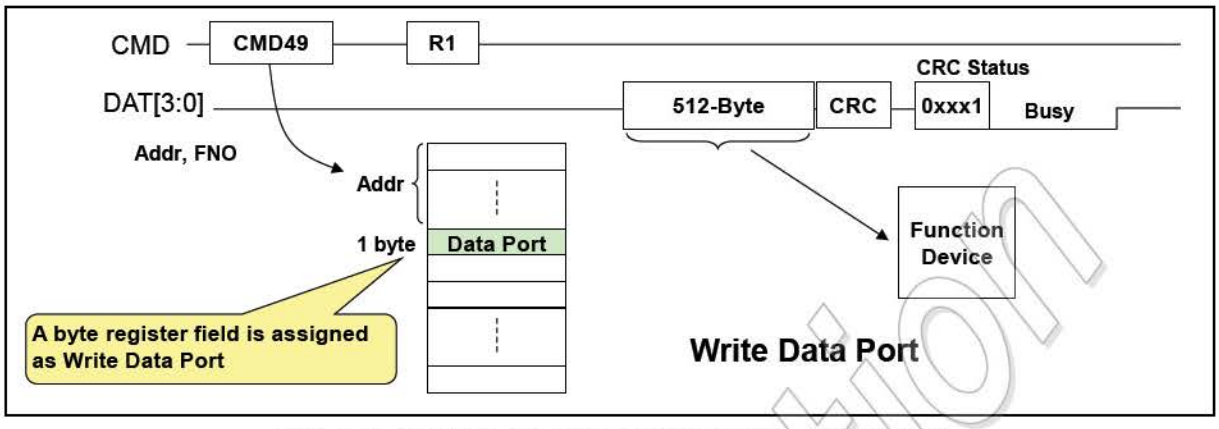


Figure 5-9 : Data Port Write Operation by CMD49

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.

SD Association

©Copyright 2001-2013 SD Group (Panasonic, SanDisk, Toshiba) and SD Card Association

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.

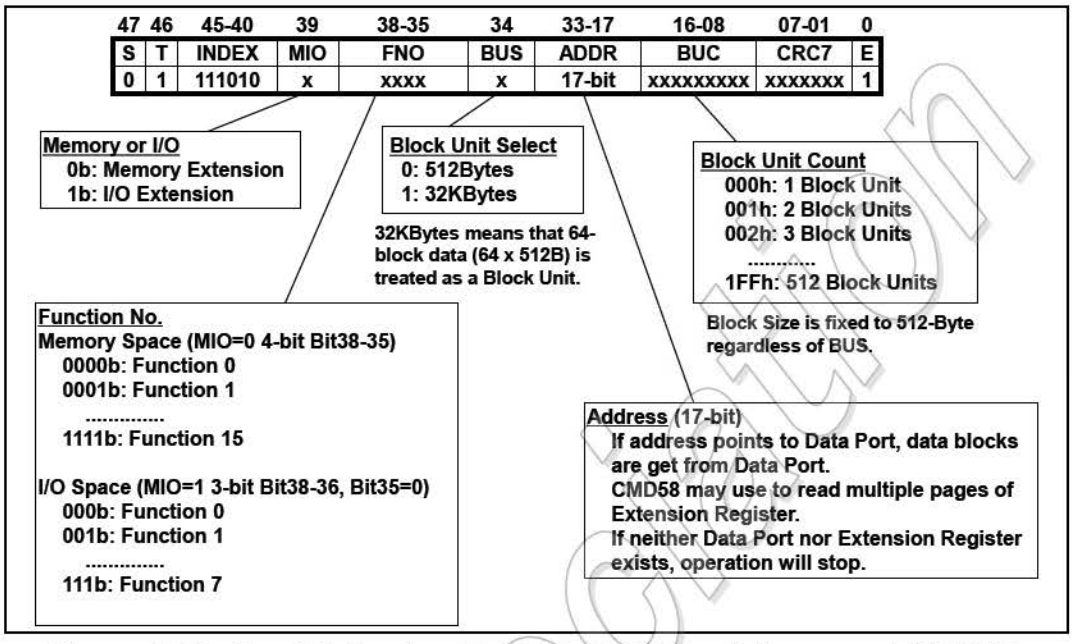


Figure 5-10 : Read Extension Register Multi-Block Command (CMD58)

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).

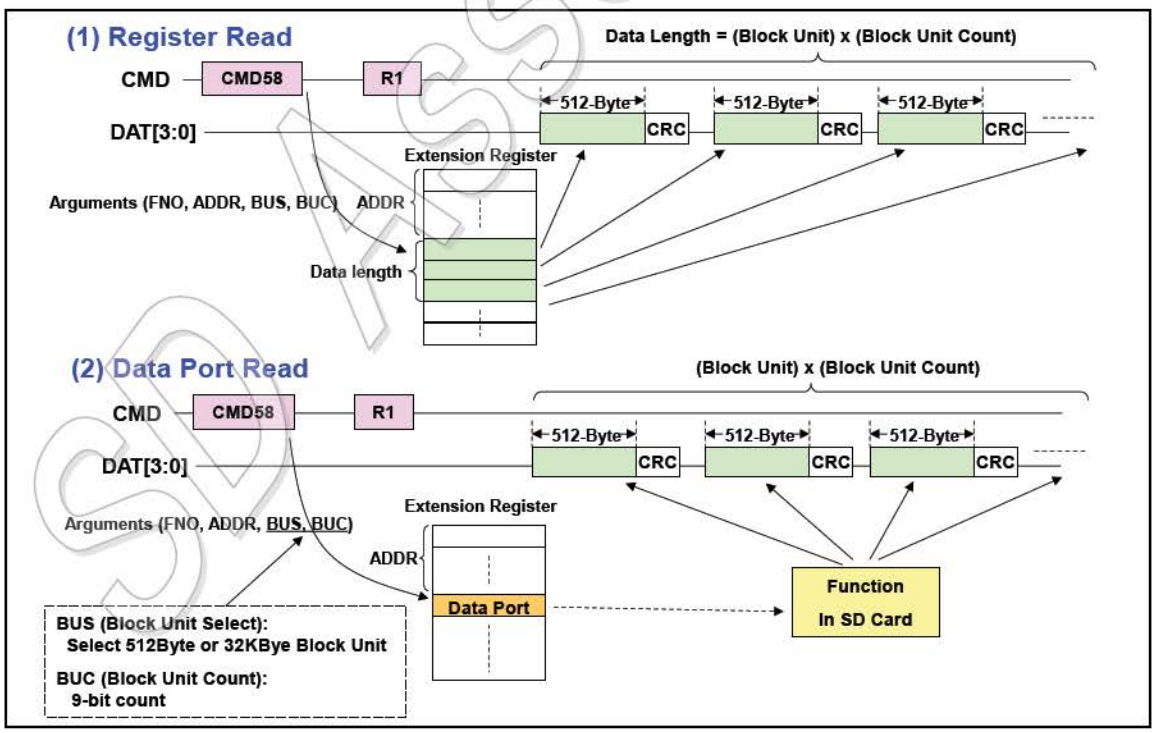


Figure 5-11 : Extension Register and Data Port Read Operation by CMD58

©Copyright 2001-2013 SD Group (Panasonic, SanDisk, Toshiba) and SD Card Association

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.

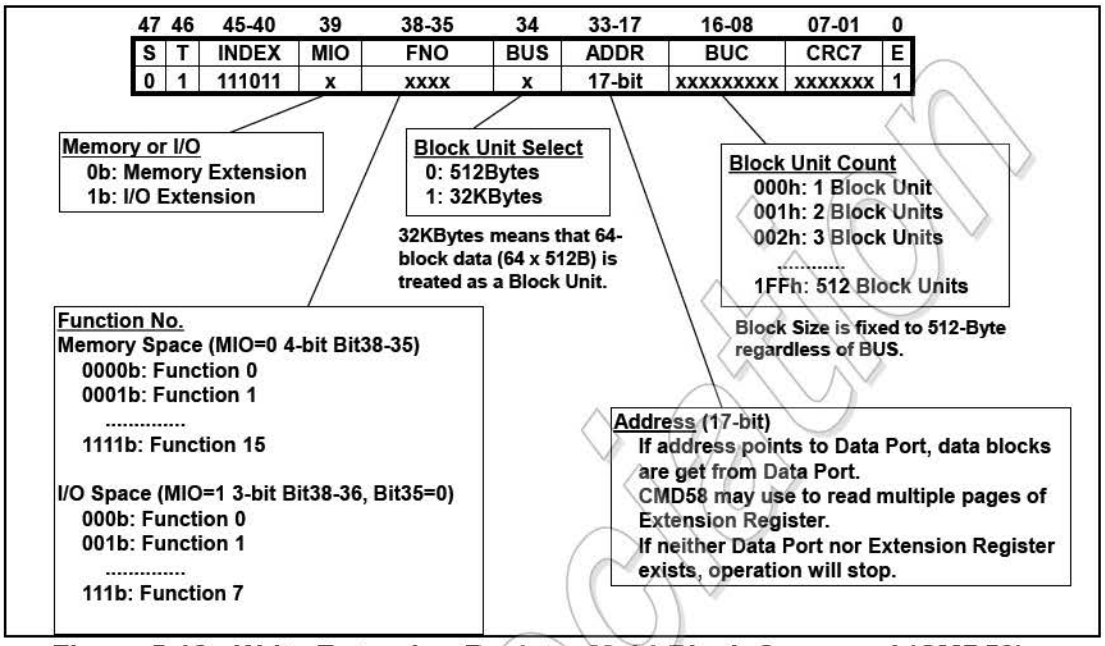


Figure 5-12 : Write Extension Register Multi-Block Command (CMD59)

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).

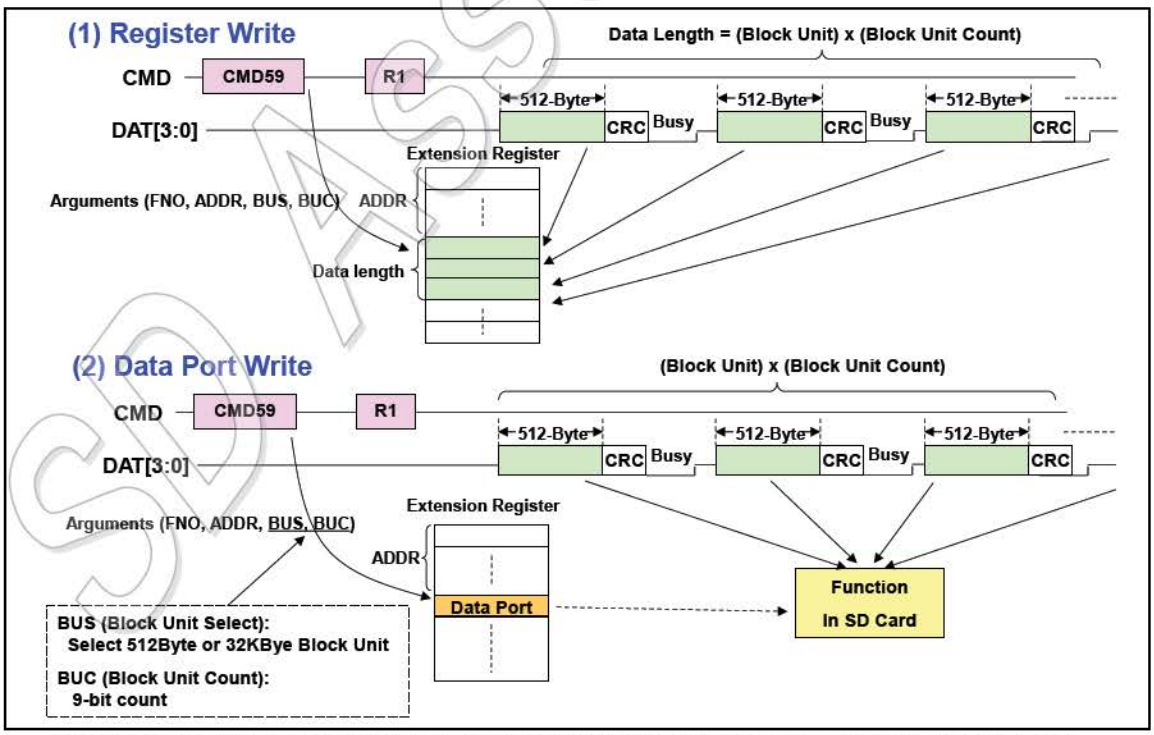


Figure 5-13 : Extension Register and Data Port Write Operation by CMD59

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.

SD Association

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.

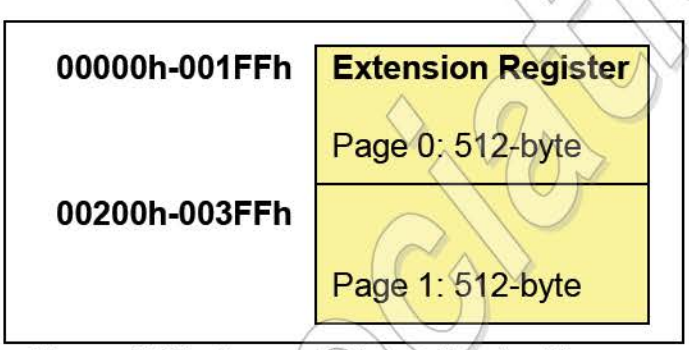
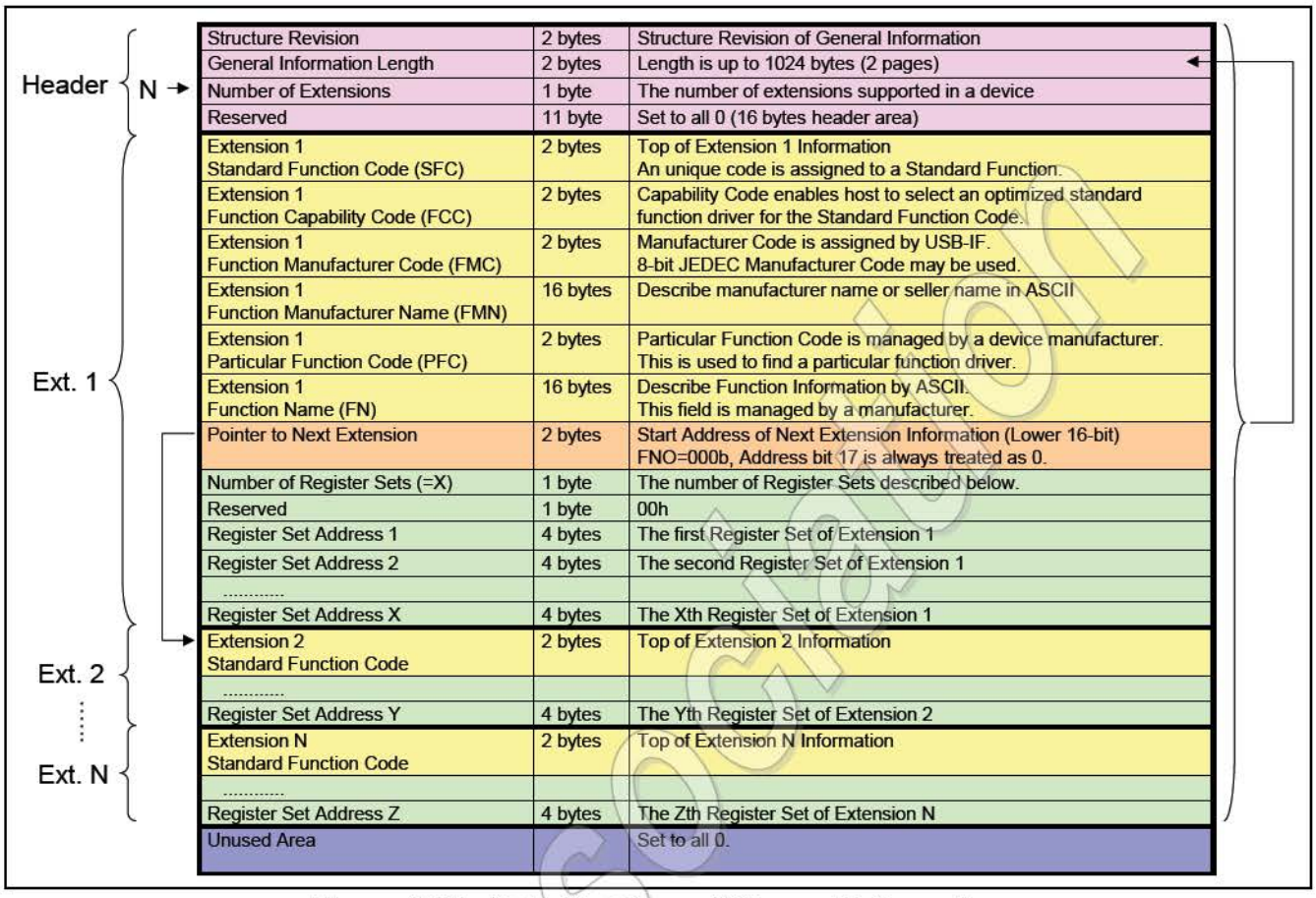


Figure 5-14 : General Information for Memory

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.

SD Association



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.

SD Association

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.

SD Association

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 5-25.

Function Name	Standard Function Code
Power Management Function	0001h

Table 5-25 : 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 5-26 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 5-26 : 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.

SD Association

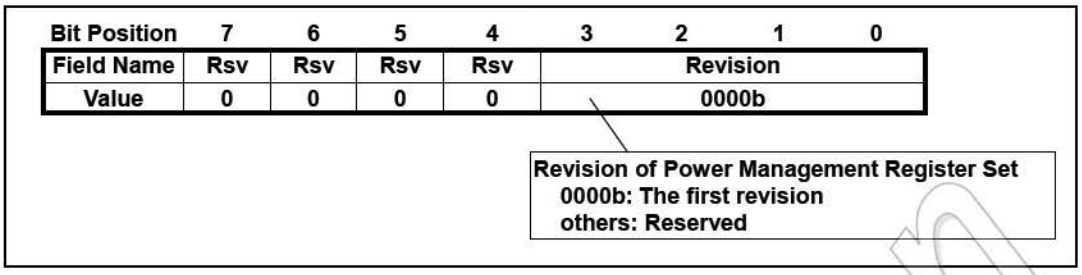


Figure 5-16 : Power Management Revision Register

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.

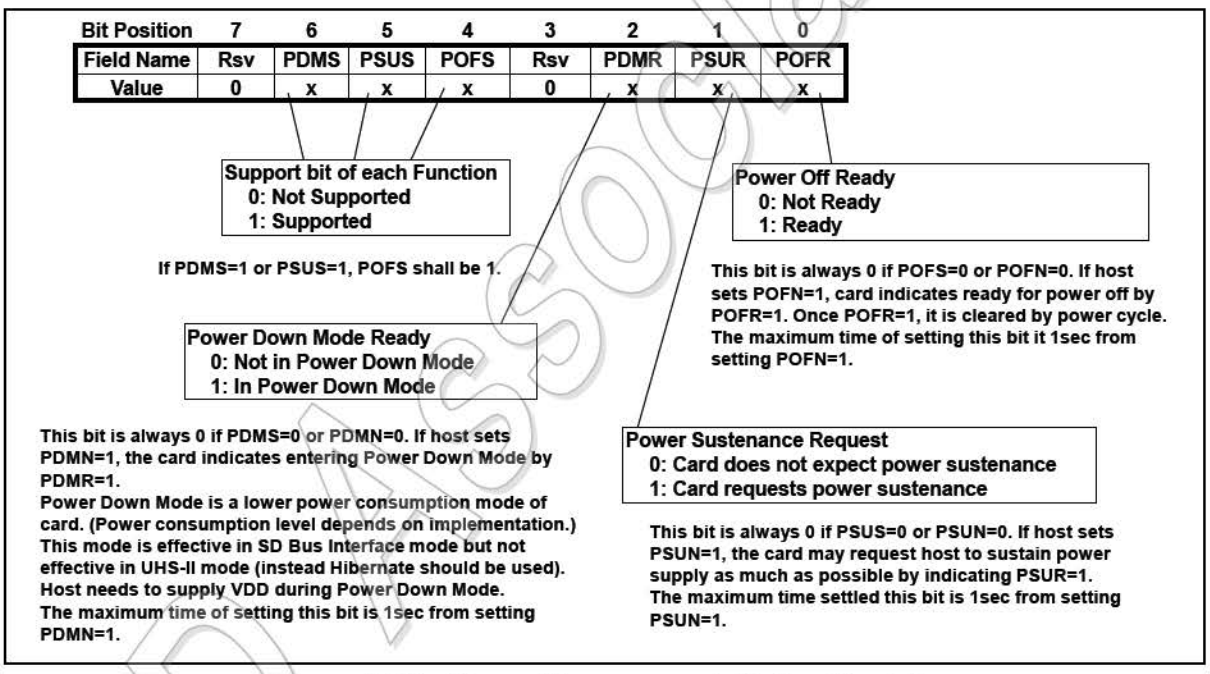


Figure 5-17 : Power Management Status Register

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.

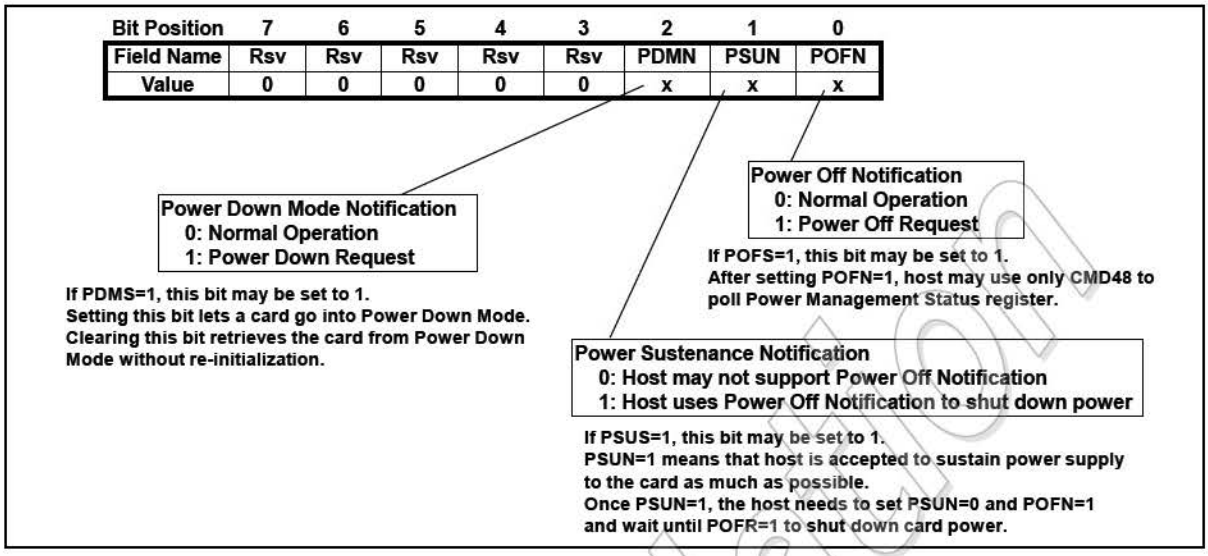


Figure 5-18 : Power Management Setting Register

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.

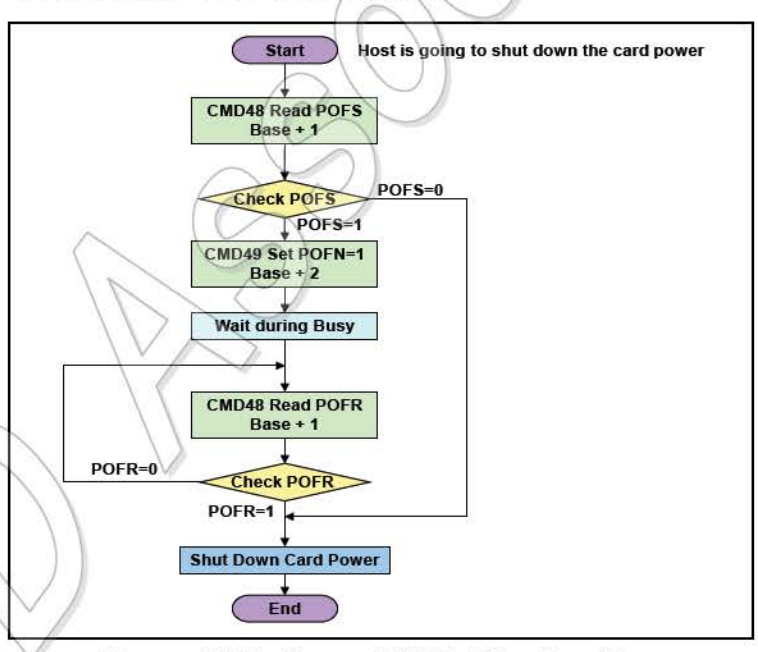


Figure 5-19 : Power Off Notification Flow

- **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

SD Association

(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.

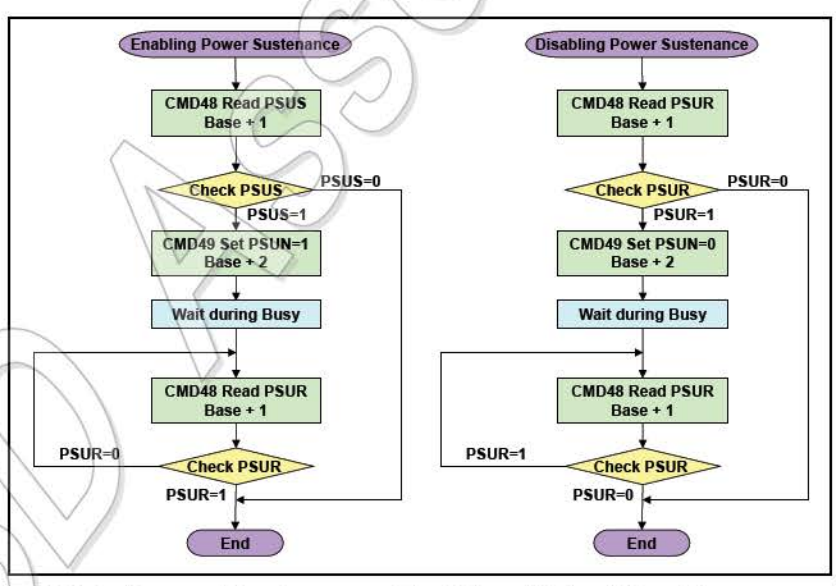


Figure 5-20 : Power Sustenance Enabling / Disabling Sequence Flow

SD Association

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 to 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.

SD Association

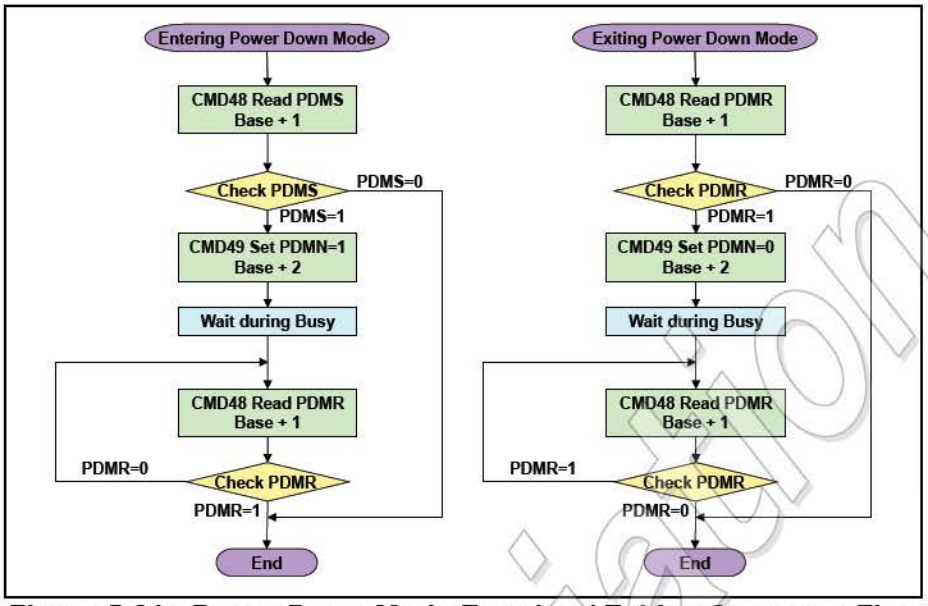


Figure 5-21 : Power Down Mode Entering / Exiting Sequence Flow

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 5-27 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 5-25) 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 5-27 : General Information of Power Management Function

©Copyright 2001-2013 SD Group (Panasonic, SanDisk, Toshiba) and SD Card Association

6. SD Memory Card Hardware Interface

A part of this section is not described.

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.

SD Association

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.

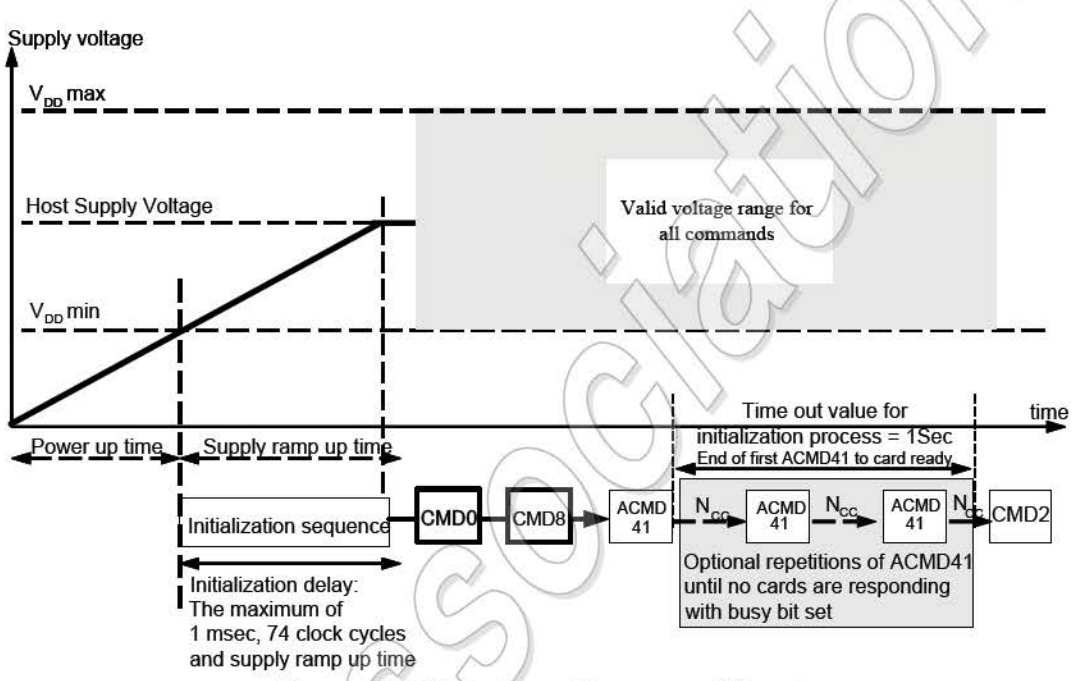


Figure 6-4: Power-up Diagram of Card

- 'Power up time' is defined as voltage rising time from 0 volt to V_{DD} min (refer to 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.

SD Association

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.

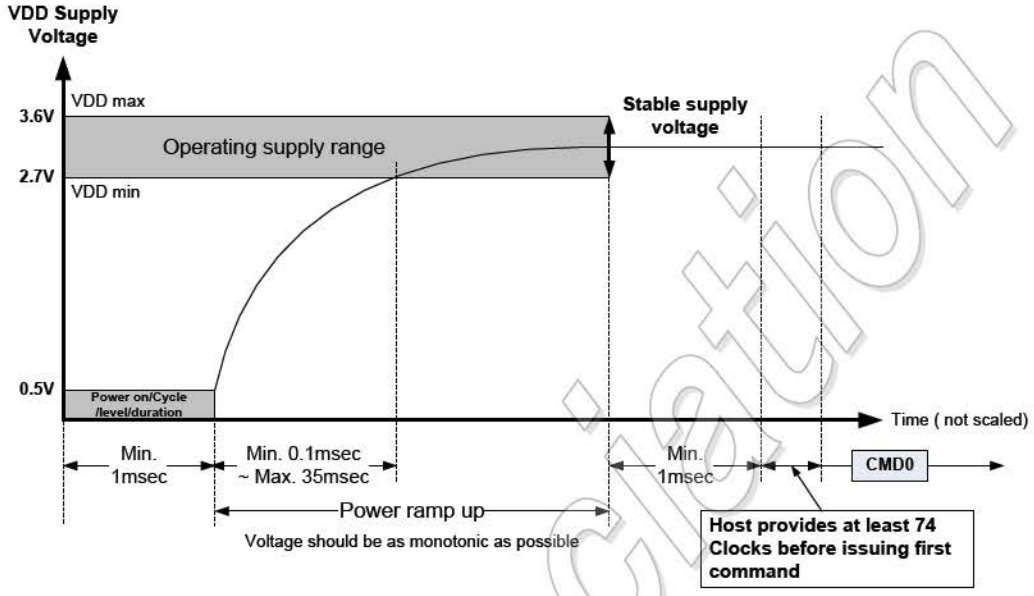


Figure 6-5 : Power Up Diagram of Host

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).

SD Association

©Copyright 2001-2013 SD Group (Panasonic, SanDisk, Toshiba) and SD Card Association

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.

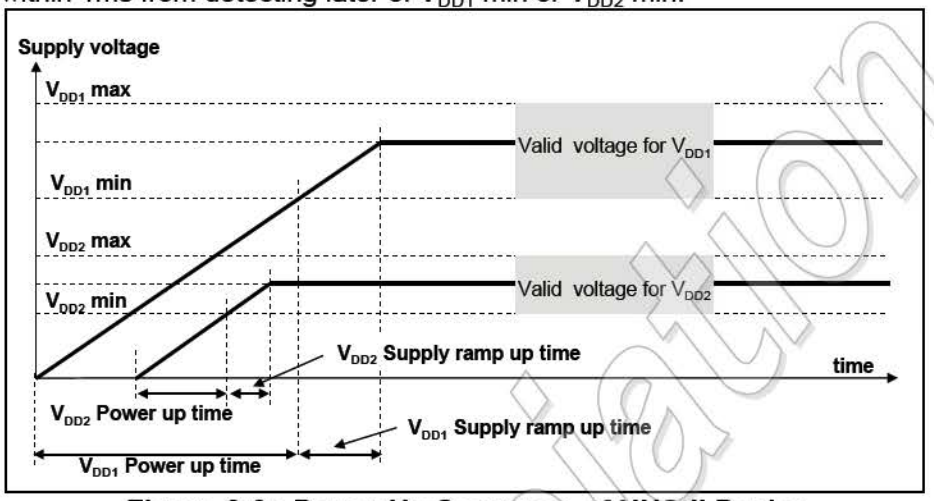


Figure 6-6 : Power Up Sequence of UHS-II Device

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.

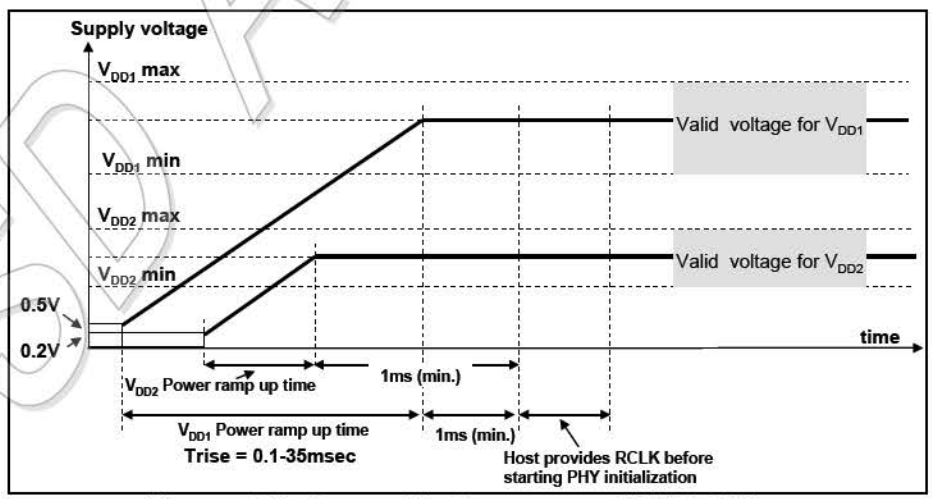


Figure 6-7 : Power Up Sequence of UHS-II Host

SD Association

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

This section is a blank in the Simplified Specification.

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.

SD Association

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, commonly found in Motorola's (and lately a few other vendors') 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.

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 to 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 to 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.

SD Association

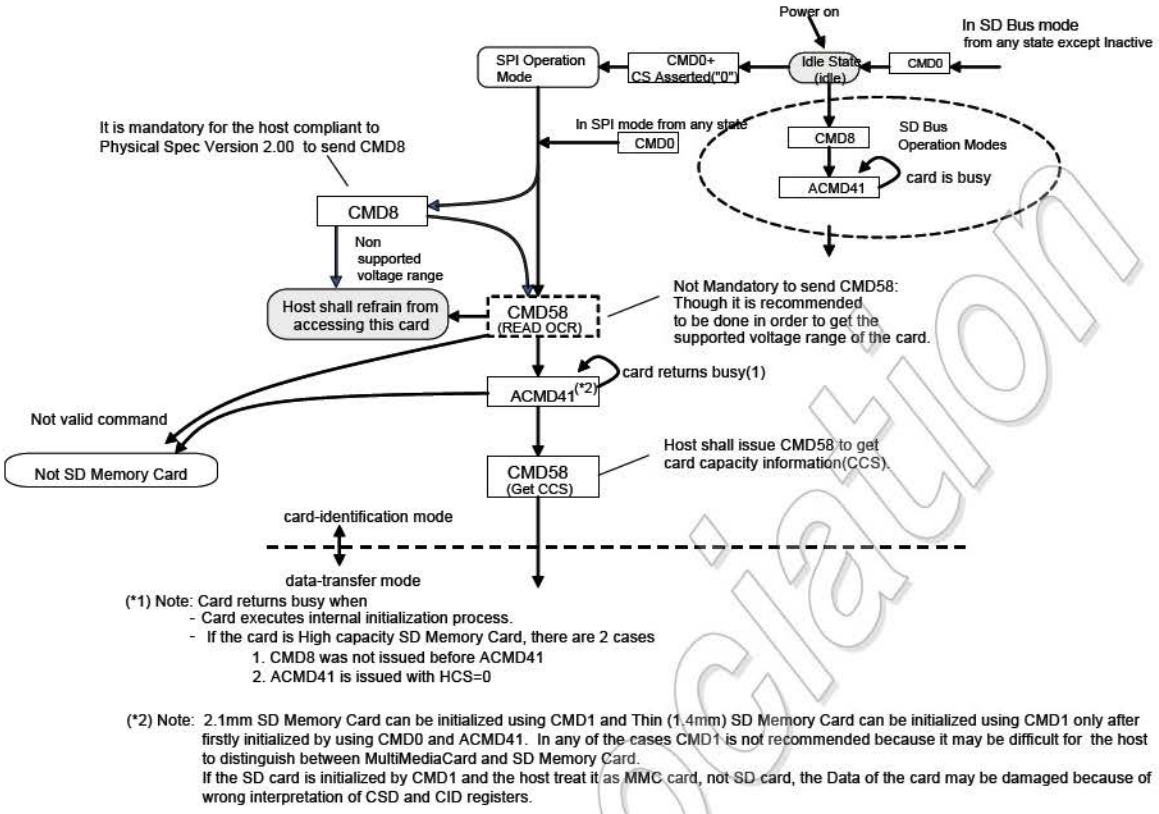


Figure 7-1 : SD Memory Card State Diagram (SPI mode)

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.

SD Association

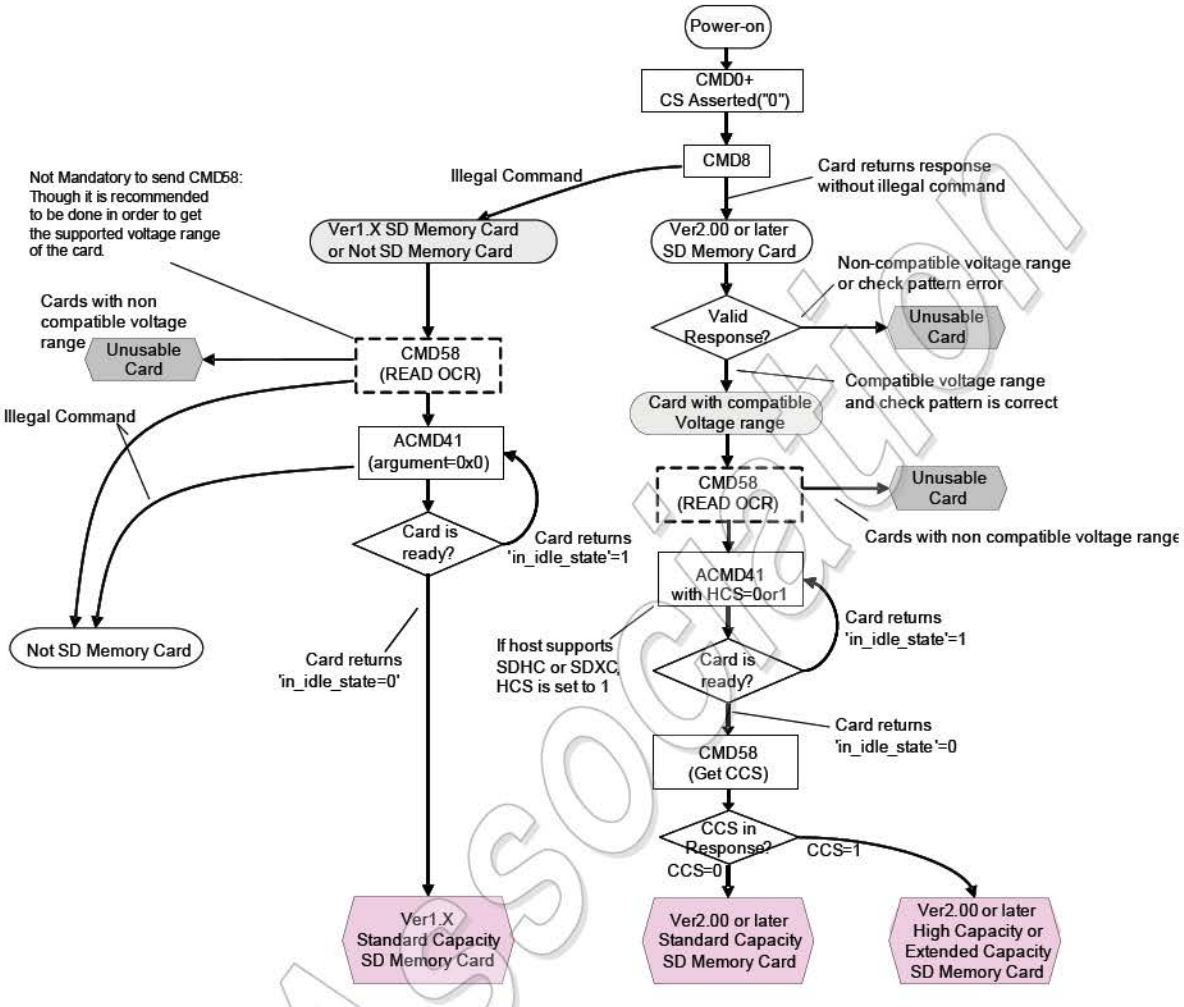


Figure 7-2: SPI Mode Initialization Flow

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 Chapter 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.

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 SDSD. 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.

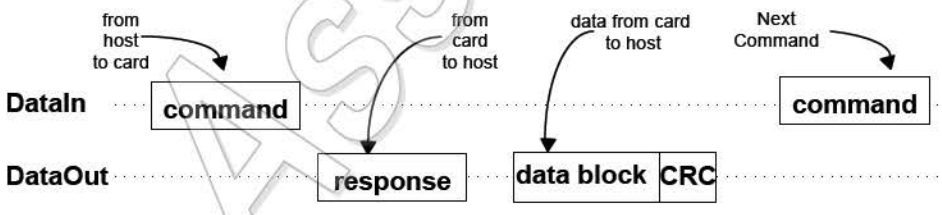


Figure 7-3: Single Block Read Operation

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 to 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.

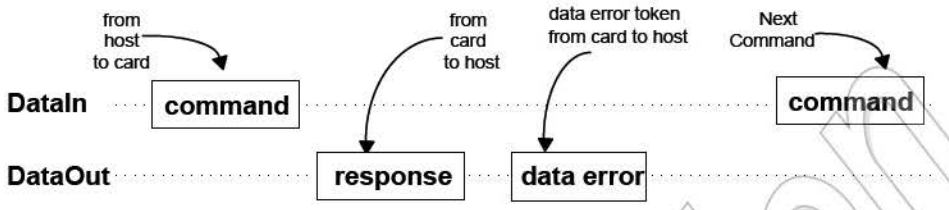


Figure 7-4: Read Operation - Data Error

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).

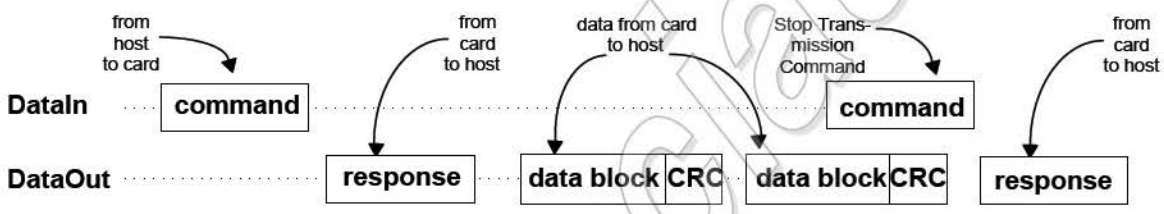


Figure 7-5: Multiple Block Read Operation

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).

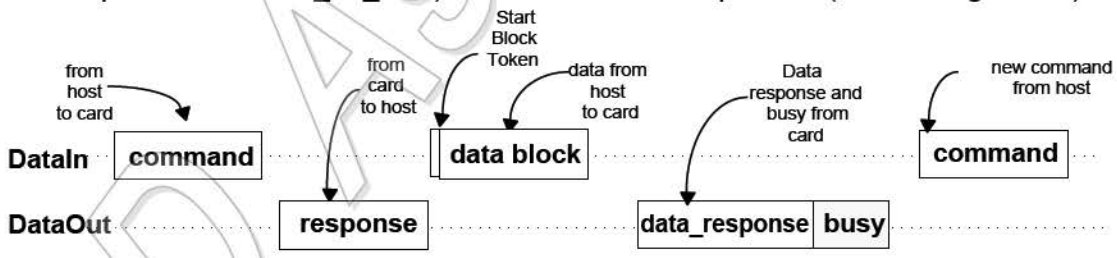


Figure 7-6: Single Block Write Operation

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.

SD Association

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 Chapter 7.3.3.2.

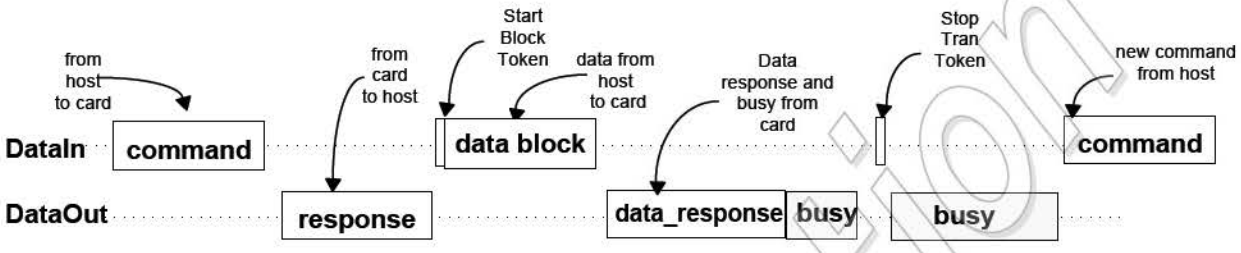


Figure 7-7: Multiple Block Write Operation

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.

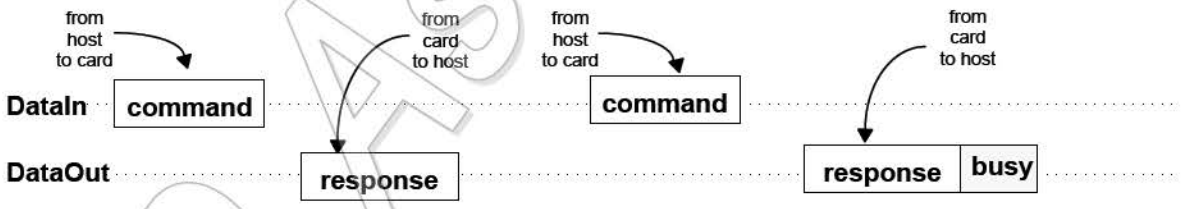


Figure 7-8: 'No data' Operations

SD Association

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 Chapter 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 4-41), 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.

SD Association

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 Chapter 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]					+							
CMD25	Mandatory[1]					+							

SD Association

©Copyright 2001-2013 SD Group (Panasonic, SanDisk, Toshiba) and SD Card Association

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
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

Table 7-2: Command Classes in SPI Mode

SD Association

7.3.1.3 Detailed Command Description

The following table provides a detailed description of the SPI bus commands. The responses are defined in Chapter 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 the "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 Chapter 4.3.10.
CMD7	No				

SD Association

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.

SD Association

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
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 Chapter 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 the "SDIO Card Specification")				

SD Association

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

SD Association

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 4-43
ACMD17	reserved				
ACMD18	yes	--	--	--	Reserved for SD security applications1
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).

(1) Refer to the "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

SD Association

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

SD Association

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.

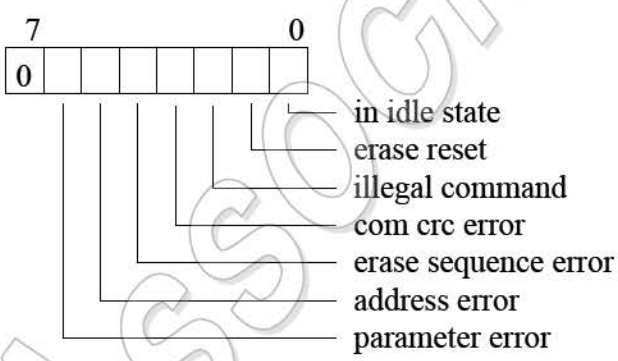


Figure 7-9: R1 Response Format

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.

SD Association

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.

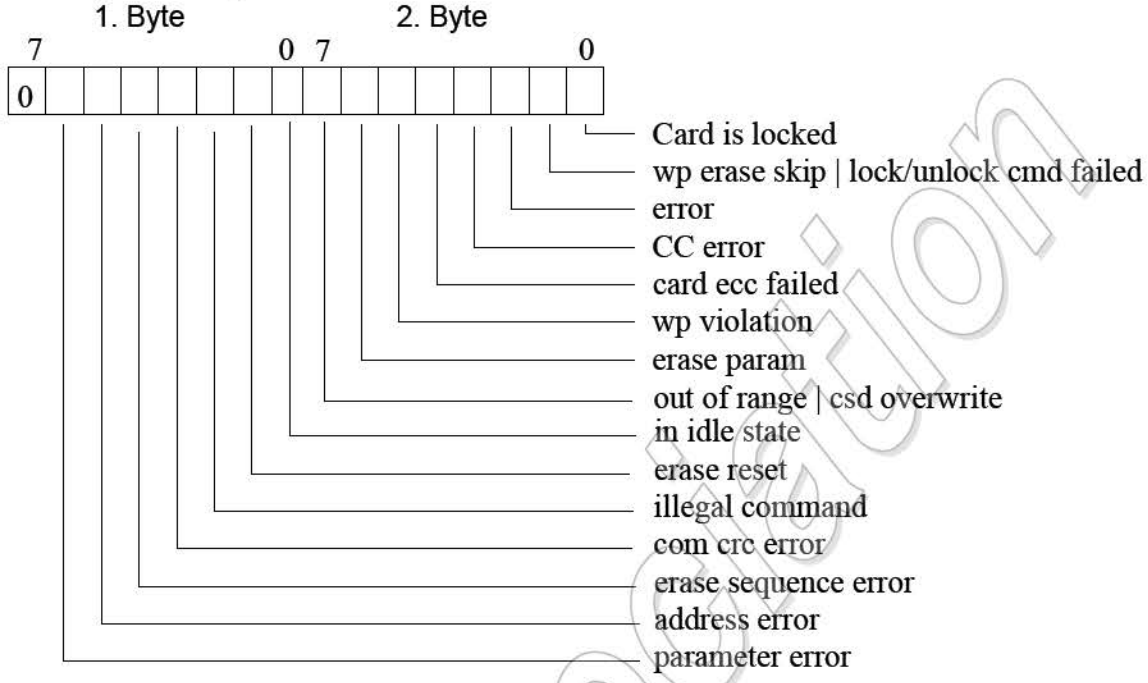


Figure 7-10: R2 Response Format

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.

SD Association

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.

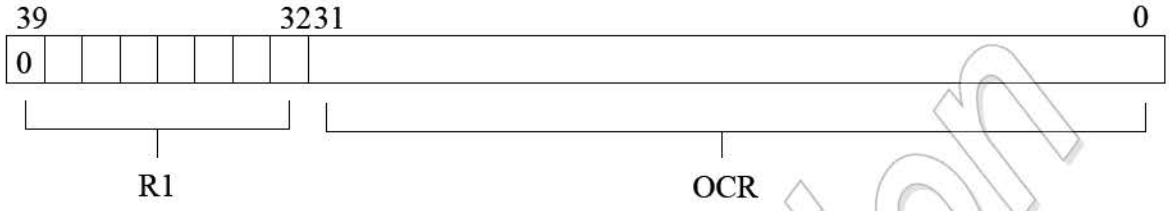


Figure 7-11: R3 Response Format

7.3.2.5 Formats R4 & R5

Those response formats are reserved for I/O mode (refer to the "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).

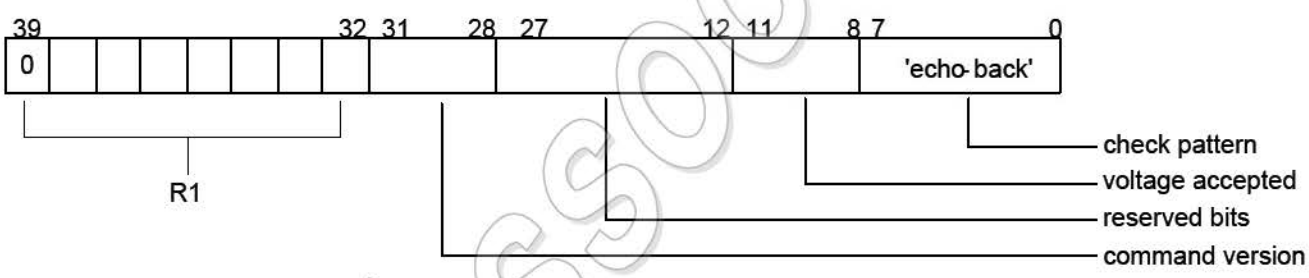


Figure 7-12: R7 Response Format

SD Association

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).

SD Association

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:

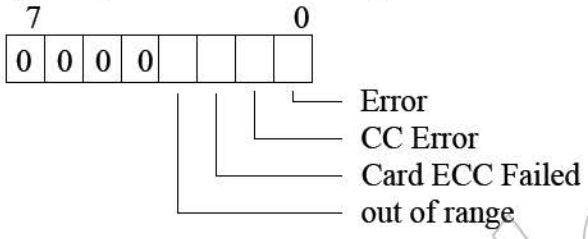


Figure 7-13: Data Error Token

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

SD Association

Identifier	Included in resp	Type[1]	Value	Description	Clear Condition[2]
Error	R2 dataEr	E R X	'0'= no error '1'= error	A general or an unknown error occurred during the operation.	C
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.

SD Association

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.

SD Association

8. Sections Effective to SD I/F Mode and UHS-II Mode

Table 8-1 shows the relation of Sections in this document that is effective to Legacy SD I/F mode and UHS-II mode.

Section	Title	SD I/F	UHS-II
1.	General Description	Yes	Yes
2.	System Features	Yes	Yes
3.	SD Memory Card System Concept	Yes	Yes
3.1	Read-Write Property	Yes	Yes
3.2	Supply Voltage	Yes	Yes
3.3	Card Capacity	Yes	Yes
3.3.1	User Area and Protected Area	Yes	Yes
3.3.2	Card Capacity Classification	Yes	Yes
3.4	Speed Class	Yes	Yes
3.5	Bus Topology	Yes	Yes
3.5.1	SD Bus	Yes	
3.5.2	SPI Bus	Yes	
3.5.3	UHS-II Bus		Yes
3.6	Bus Protocol	Yes	
3.6.1	SD Bus Protocol	Yes	
3.6.2	SPI Bus Protocol	Yes	
3.6.3	UHS-II Bus Protocol		Yes
3.7	SD Memory Card–Pins and Registers	Yes	Yes
3.7.1	SD Bus Pin Assignment	Yes	
3.7.2	UHS-II Pin Assignment		Yes
3.8	ROM Card	Yes	Yes
3.9	Ultra High Speed Phase I (UHS-I) Card	Yes	
3.10	Ultra High Speed Phase II (UHS-II) Card		Yes
4.	SD Memory Card Functional Description	Yes	Yes
4.1	General	Yes	Yes
4.2	Card Identification Mode	Yes	Yes
4.2.1	Card Reset	Yes	Yes
4.2.2	Operating Condition Validation	Yes	
4.2.3	Card Initialization and Identification Process	Yes	
4.2.3.1	Initialization Command (ACMD41)	Yes	Yes
4.2.4	Bus Signal Voltage Switch Sequence	Yes	
4.3	Data Transfer Mode	Yes	
4.3.1	Wide Bus Selection/Deselection	Yes	
4.3.2	2 GByte Card	Yes	
4.3.3	Data Read	Yes	Yes
4.3.4	Data Write	Yes	Yes
4.3.5	Erase	Yes	Yes
4.3.6	Write Protect Management	Yes	
4.3.7	Card Lock/Unlock Operation	Yes	Yes

SD Association

4.3.7.4	Relation Between ACMD6 and Lock/Unlock State	Yes	
4.3.7.5	Commands Accepted for Locked Card	Yes	Yes
4.3.7.6	Two Types of Lock/Unlock Card	Yes	Yes
4.3.8	Content Protection	Yes	Yes
4.3.9	Application-Specific Commands	Yes	
4.3.10	Switch Function Command	Yes	Yes
4.3.11	High-Speed Mode (25 MB/sec interface speed)	Yes	
4.3.12	Command System	Yes	Yes
4.3.13	Send Interface Condition Command (CMD8)	Yes	Yes
4.3.14	Command Functional Difference in Card Capacity Types	Yes	Yes
4.4	Clock Control	Yes	
4.5	Cyclic Redundancy Code (CRC)	Yes	Yes
4.6	Error Conditions	Yes	Yes
4.6.1	CRC and Illegal Command	Yes	
4.6.2	Read, Write and Erase Timeout Conditions	Yes	Yes
4.7	Commands	Yes	Yes
4.7.1	Command Types	Yes	Yes
4.7.2	Command Format	Yes	Yes
4.7.3	Command Classes	Yes	Yes
4.7.4	Detailed Command Description	Yes	Yes
4.7.5	Difference of SD Commands Definition in UHS-II		Yes
4.8	Card State Transition Table	Yes	Yes
4.9	Responses	Yes	Yes
4.10	Two Status Information of SD Memory Card	Yes	Yes
4.11	Memory Array Partitioning	Yes	Yes
4.12	Timings	Yes	
4.13	Speed Class Specification	Yes	Yes
4.14	Erase Timeout Calculation	Yes	Yes
4.15	Set Block Count Command	Yes	
5.	Card Registers	Yes	Yes
5.1	OCR register	Yes	Yes
5.2	CID register	Yes	Yes
5.3	CSD Register	Yes	Yes
5.3.1	CSD_STRUCTURE	Yes	Yes
5.3.2	CSD Register (CSD Version 1.0)	Yes	
5.3.3	CSD Register (CSD Version 2.0)	Yes	Yes
5.4	RCA register	Yes	Yes
5.5	DSR register (Optional)	Yes	Yes
5.6	SCR register	Yes	Yes
5.7	Function Extension Specification	Yes	Yes
5.8	Application Specification on Function Extension	Yes	Yes
6.	SD Memory Card Hardware Interface	Yes	
6.1	Hot Insertion and Removal	Yes	
6.2	Card Detection (Insertion/Removal)	Yes	
6.3	Power Protection (Insertion/Removal)	Yes	Yes

SD Association

6.4.1	Power Up Sequence for SD Bus Interface	Yes	
6.4.2	Power Up Sequence for UHS-II Interface		Yes
6.5	Programmable Card Output Driver (3.3V Single End)	Yes	
6.6	Bus Operating Conditions for 3.3V Signaling	Yes	
6.6.1	Threshold Level for High Voltage Range	Yes	Yes
6.6.2	Peak Voltage and Leakage Current	Yes	
6.6.3	Current Consumption	Yes	Yes
6.6.4	Bus signal line load	Yes	
6.6.5	Bus Signal Levels	Yes	
6.6.6	Bus Timing (Default)	Yes	
6.6.7	Bus Timing (High-Speed Mode)	Yes	
6.7	Driver Strength and Bus Timing for 1.8V Signaling	Yes	
6.8	Electrical Static Discharge (ESD) Requirement	Yes	Yes
7.	SPI Mode	Yes	
8.	Sections Effective to SD I/F Mode and UHS-II Mode	Yes	Yes
A.1	Related Documentation	Yes	Yes
B.1	Terminology	Yes	Yes
B.2	Abbreviations	Yes	Yes
D.1	UHS-I Tuning Procedure	Yes	
C.1	Internal Clock Delay Method	Yes	
E.1	Decoupling Capacitor Connected to Power Line	Yes	Yes
E.2	Decoupling Capacitors	Yes	Yes

Table 8-1 : Sections Effective to SD I/F Mode and UHS-II Mode

SD Association

Appendix A (Normative) : Reference

A.1 Related Documentation

- Part 1 UHS-II Addendum Version 1.00 or later
- Part 1 Standard Size SD Card Mechanical Addendum Version 4.00 or later
- Part 1 miniSD Memory Card Addendum Version 2.01
- Part 1 microSD Memory Card Addendum Version 4.00 or later
- Part 2 File System Specification Version 3.00
- Part 3 Security Specification Version 3.00
- Part H2 Host Implementation Guideline Version 1.00
- Part H2 Speed Class Implementation Guideline Version 1.00

SD Association

Appendix B (Normative) : Special Terms

B.1 Terminology

block	A number of bytes, basic data transfer unit
broadcast	A command sent to all cards on the SD bus
Blocklen	Block Length set by CMD16
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.
Flash	A type of multiple time programmable non-volatile memory
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
Lumped	A signal path between host and card which is considerably small compared to the signal rise time. It is considered as "lumped" system
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
push-pull	A logical interface operation mode, a complementary pair of transistors is used to push the interface level to HIGH or LOW
sector	A number of blocks, basic erase unit
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
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.
Variable Data Window	An overlapped area of valid data window is not available or too small for all Process, Voltage and Temperature variations.

B.2 Abbreviations

ACMD6	Set bus width command
ACMD41	Initialization command
AU	Allocation Unit
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.

SD Association

CMD19	A new command for sending tuning block
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
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
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
LOW, HIGH	Binary interface states with defined assignment to a voltage level
MSB, LSB	The Most Significant Bit or Least Significant Bit
MLCC	Multi-Layer Ceramic Capacitor
MTP	Multiple Time Programmable memory
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.
NSAC	Defines the worst case for the clock rate dependent factor of the data access time
OCR	Operation Conditions Register
OTP	One Time Programmable memory
P_w	Performance of Write
P_m	Performance of Move
P_r	Performance of Read
PDN	Power Delivery Network
RCA	Relative Card Address register
ROM	Read Only Memory
RU	Recording Unit
SDCLK	Clock line of SD bus
S18R	Switching to 1.8V Request in ACMD41 argument
S18A	Switching to 1.8V Accepted in ACMD41 response
SPI	Serial Peripheral Interface
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)
T_{fw}	FAT write time
T_{fr}	FAT read time
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.
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

SD Association

©Copyright 2001-2013 SD Group (Panasonic, SanDisk, Toshiba) and SD Card Association

UHS104	One of UHS-I Card Types supporting SDR104
UHS156	UHS-II Generation 1 Card Type supporting FD156 and HD312 (Optional)
UHS-II I/F	Interface using contact pin numbers 7 to 8 and 10 to 17.
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 (First row)
V_{DD2}	1.8V range power supply for UHS-II Card (Second row)
V_{SS}	Power supply ground
X5R/X7R	Symbol for dielectric material of capacitors

SD Association

Appendix C (Informative) : Examples for Fixed Delay UHS-I Host Design

This section is a blank in the Simplified Specification.

SD Association

©Copyright 2001-2013 SD Group (Panasonic, SanDisk, Toshiba) and SD Card Association

Appendix D : UHS-I Tuning Procedure

This section is a blank in the Simplified Specification.

SD Association

Appendix E :
Host Power Delivery Network (PDN) Design Guide

This section is a blank in the Simplified Specification.

SD Association

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

SD Association