SD Specifications

Part 2

File System Specification

Version 2.00 Draft

October 28, 2005

SD Group
Matsushita Electric Industrial Co., Ltd. (MEI)
SanDisk Corporation
Toshiba Corporation

Technical Committee
SD Card Association

CONFIDENTIAL

Draft Only Page

This Page is used to describe draft version and information limited to the draft. This page will be removed when the Specification is released for voting.

Draft Revision History

Date	Version	Changes compared to previous issue
September 5, 2005	0.50	Initial release to FSTG
September 28, 2005	0.90	- LFN option for FAT12/16 is added - Timestamp option for FAT12/16 is added - Alignment method of FAT32 is modified - Copyright statement is modified - Host requirements are added
October 3, 2005	0.91	- Host requirements are modified
October 5, 2005	0.92	- Format Parameter Computations is modified
October 7, 2005	0.93	- The definitions of unit are added in tables and figures
October 13, 2005	0.94	- Copyright statement is modified - FAT32 Comparison Table between Microsoft FAT Specification and SD Specification is added in appendix
October 27, 2005	0.95	- Copyright statement is modified - Host requirements are modified - High Capacity SD Memory Card definition is modified - Typo fixes and some clarification notes
October 28, 2005	0.96	- Host requirements are modified

Revision History

Date	Version	Changes compared to previous issue
March 22, 2000	1.00	Base version initial release.
April 15, 2001	1.01	- One notation is added - The description of System ID field is modified - The description of Reserved field in the Extended FDC Descriptor is modified - The description of Reserved field in the Directory Entry is modified - Some computations are added in Annex B - Typo fixes and some clarification notes
October 28, 2005	2.00	- File System Specification V1.01 Supplemental Note is merged into base specification - New file system definition for High Capacity SD Memory Card, whose capacity is over 2GB and up to 32GB, is added - LFN option for FAT12/16 is added - Timestamp option for FAT12/16 is added - Typo fixes and some clarification notes

Intentionally Blank

This area is reserved for IP Declaration, which will be added later.

Conditions for publication

Publisher:

SD Card Association
2400 Camino Ramon, Suite 375
San Ramon, CA 94583 USA
Telephone: +1 (925) 275-6615
Fax: +1 (925) 886-4870
E-mail: office@sdcard.org

Copyright Holders:

SD Group
- Matsushita Electric Industrial Co., Ltd. (MEI)
- SanDisk Corporation (SanDisk)
- Toshiba Corporation (Toshiba)

SD Card Association

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

Confidentiality:

The contents of this document are deemed confidential information of the SD-3C LLC and/or the SD Card Association (the "Disclosers"). As such, the contents and your right to use the contents are subject to the confidentiality obligations stated in the written agreement you entered into with the Disclosers which entitled you to receive this document, such as a Non-Disclosure Agreement, the License Agreement for SDA Memory Card Specifications (also known as "LAMS"), the SD Host/Ancillary Product License Agreement (also known as "HALA") or the IP Policy.

Exemption:

The information contained herein is presented only as a standard specification for SD Card and SD Host products. No responsibility is assumed by SD Card Association for any damages, any infringements of patents or other right of the third parties, which may result from its use. No license is granted by implication or otherwise under any patent or rights of SD Group, SD Card Association or others.

Conventions Used in This Document

Naming Conventions

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

Numbers and Number Bases

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

Key Words

- May: Indicates flexibility of choice with no implied recommendation or requirement.
- Shall: Indicates a mandatory requirement. Designers shall implement such mandatory requirements to ensure interchangeability and to claim conformance with the specification.
- Should: Indicates a strong recommendation but not a mandatory requirement. Designers should give strong consideration to such recommendations, but there is still a choice in implementation.

Application Notes

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

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

Table of Contents

Table of Figures

Table of Tables

1. Overview

This part specifies the file system of SD Memory Card (Secure Digital Memory Card). It manages the data recorded in SD Memory Card as files, and it enables the data exchange between the hosts supporting SD Memory Card. The card to which the file system in this specification is applied shall comply with "Part1. PHYSICAL LAYER SPECIFICATION" of SD Memory Card specifications.

NOTE: The High Capacity SD Memory Card covered by this specification shall comply with "Part1. PHYSICAL LAYER SPECIFICATION Version 2.00" or the later version. And the High Capacity SD Memory Card supports capacity greater than 2GB and is limited by this version of the specification to 32GB.

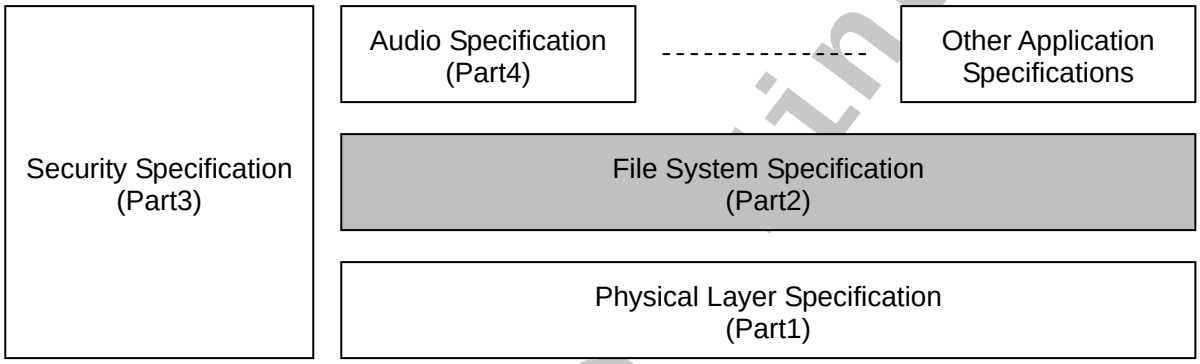


Figure 1-1 : SD Memory Card Document Structure

The SD Memory Card has two recordable areas. One is Data Area that user can access without mutual authentication. The other is Protected Area that user can access after mutual authentication. These two areas have file system independently with each other. This part specifies the file system for Data Area. And the one for Protected Area is specified in "Part3. SECURITY SPECIFICATION" of SD Memory Card specifications.

This specification specifies some file systems. The type of the file system to be used shall be uniquely decided with Card Capacity as follows. Here, Card Capacity means the total size of Data Area and Protected Area size.

Card Capacity	File System Type for Data Area
~2048MB	FAT12 / FAT16
Over 2048MB ~32768MB	FAT32

Table 1-1 : File System Type

That is, all Standard Capacity SD Memory Cards whose capacity is 2048MB or less shall use FAT12 / FAT16 file system, and never use FAT32 file system. And on the contrary, all High Capacity SD Memory Cards shall use FAT32 file system, and never use FAT12 / FAT16 file system. This includes the prohibition of partial format of SD Memory Card. For example, 8GB SD Memory Card should not be formatted as 2GB card with FAT12 / FAT16 file system. In this case, whole area of 8GB should be formatted with FAT32 file system.

This specification consists of several chapters. Chapter 2 describes the main features of this file

system. Chapter 3 specifies FAT12 / FAT16 file system. Chapter 4 specifies FAT32 file system. Appendix follows these chapters and it describes some normative references, definitions, and recommendations.

NOTE:

1. **The Part 1 Physical Layer Specification Version 2.00 and Part 2 File System Specification Version 2.00 allow Standard Capacity SD Memory Cards to have capacity up to and including 2GB and High Capacity SD Memory Cards to have capacity up to and including 32GB. SD Memory Cards with a capacity greater than 32GB will be available with updated versions of Part 1 and Part 2 Specifications.**
2. **Hosts that can access (read and/or write) SD Memory Cards with a capacity greater than 2GB and up to and including 32GB, shall also be able to access SD Memory Cards with a capacity of 2GB or less.**

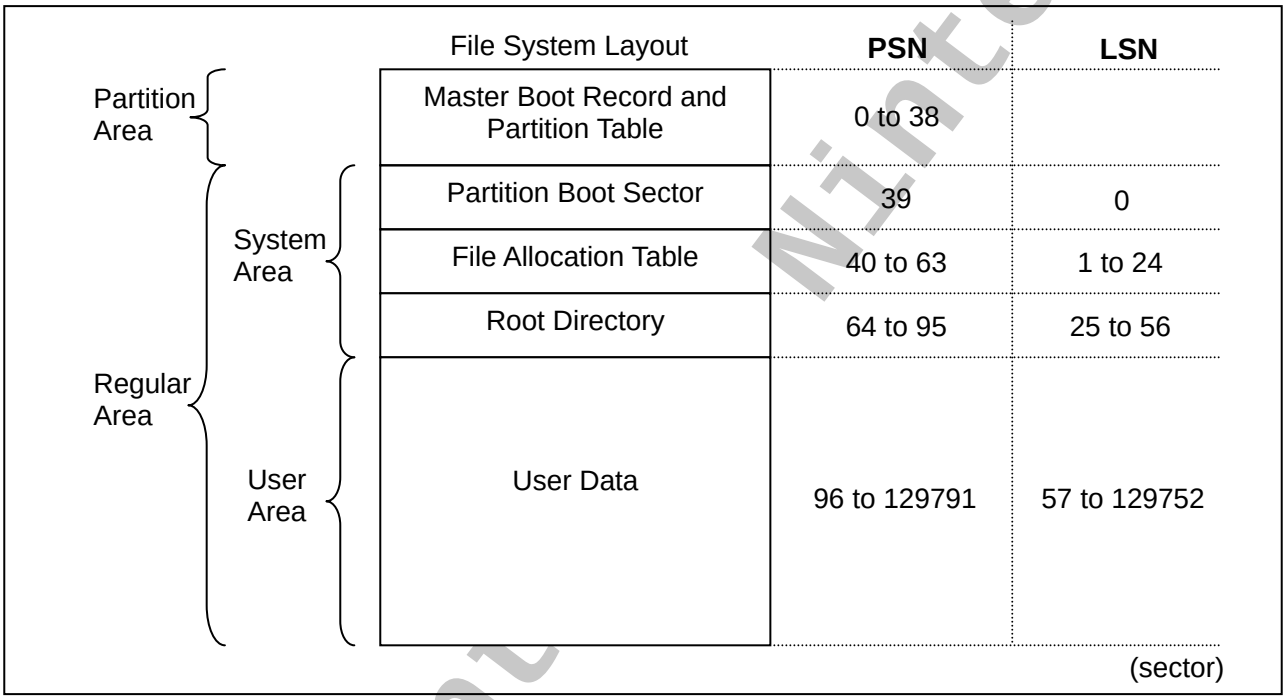
2. Features

- **Compatibility with FAT File System**
  - **Standard Capacity SD Memory Card complies with ISO/IEC 9293 (FAT12 / FAT16).**
  - **High Capacity SD Memory Card complies with FAT32 File System.**
- **CHS parameter recommendation**
- **Cluster size recommendation**
- **Boundary Unit recommendation for User Data area alignment**
- **Long File Name support (Optional)**

3. FAT12/FAT16 File System

3.1. Volume Structure

The volume structure of the Standard Capacity SD Memory Card, whose capacity is 2GB or less, is specified in this section. It defines the logical structure of the Data Area. For the identification of the Data Area as a partition, the first sector has Master Boot Record and Partition Table. And the Standard Capacity SD Memory Card uses the FAT file system (ISO/IEC 9293) and supports both FAT12 and FAT16 as the file system type.



PSN : Physical Sector Number

LSN : Logical Sector Number

Figure 3-1 : Example of Volume Structure for Data Area

3.1.1. Arrangement of the Data Area

3.1.1.1. Physical Address

Each sector shall be identified by a Physical Address comprising the parameters of SD Memory Card's own.

3.1.1.2. Physical Sector Number

Each sector on a volume shall be identified by a Physical Sector Number. There shall be a one-to-one correspondence between Physical Address and Physical Sector Number. The Physical Sector Numbers shall be assigned in an ascending sequence, beginning with 0.

3.1.1.3. Logical Sector Number

Each sector on a partition shall be identified by a Logical Sector Number. The first sector of the partition shall be assigned 0 as Logical Sector Number. There shall be a one-to-one correspondence between Physical Sector Number.

3.1.1.4. Partition Area and Regular Area

The space on Data Area shall be divided into two parts: Partition Area and Regular Area. And the Regular Area shall be divided into System Area and User Area.

The Partition Area shall occupy sectors with the Physical Sector Numbers 0 to *NOM-1*, where *NOM* is the number of sectors in the Master Boot Record and Partition Table.

The Regular Area is a partition of the volume, and divided into System Area and User Area.

The System Area shall occupy sectors with the Physical Sector Numbers *NOM* to *NOM+SSA*-1, where *SSA* is the number of sectors in the System Area. The System Area shall contain Descriptors that specify the recording format of the Regular Area. No part of any file shall be contained in the System Area.

The User Area shall occupy sectors with the Physical Sector Numbers starting with *NOM+SSA*. The User Area shall contain files and directories, and be recorded user data.

3.1.2. Arrangement of the User Area

3.1.2.1. Clusters

The User Area shall be organized into units of allocation called clusters. Each cluster shall consist of the same number of sectors. Each cluster shall be identified by a unique Cluster Number. Cluster Numbers shall be assigned integer number starting with 2.

3.1.2.2. Status of Clusters

A status shall be assigned to each cluster, and shall be one of the following:

- allocated to a file
  The cluster is already allocated.

- available for allocation
  The cluster is prepared for allocate.

- defective
  The cluster is defective. This cluster cannot be allocated.

The status of each cluster shall be identified according to ISO/IEC 9293.

3.1.3. Arrangement of the Partition Area

The first sector of the Data Area has a Master Boot Record that includes executable codes and Partition Table that includes the information to identify the partition.

BP	Length (bytes)	Field Name	Contents
0	446	Master Boot Record	Not Restricted
446	16	Partition Table (partition1)	Refer to Table 3-2
462	16	Partition Table (partition2)	All 00h
478	16	Partition Table (partition3)	All 00h
494	16	Partition Table (partition4)	All 00h
510	2	Signature Word	55h, AAh

Table 3-1 : Master Boot Record and Partition Table

(BP 0 to 445) Master Boot Record
The content of this field is not specified by this specification.

(BP 446 to 461) Partition Table (partition1)
This field shall specify the information of first partition in the volume. This partition means Regular Area that user can access without mutual authentication. It shall be recorded according to Table 3-2.

(BP 462 to 477) Partition Table (partition2)
This field shall be recorded as ZEROs, as a volume shall consist of single Regular Area.

(BP 478 to 493) Partition Table (partition3)
This field shall be recorded as ZEROs, as a volume shall consist of single Regular Area.

(BP 494 to 509) Partition Table (partition4)
This field shall be recorded as ZEROs, as a volume shall consist of single Regular Area.

(BP 510 and 511) Signature Word
This field shall be recorded as 55h (BP 510) and AAh (BP 511).

BP	Length (bytes)	Field Name	Contents
0	1	Boot Indicator	00h or 80h
1	1	Starting Head	Numeric Value
2	2	Starting Sector / Starting Cylinder	Numeric Value
4	1	System ID	01h or 04h or 06h
5	1	Ending Head	Numeric Value
6	2	Ending Sector / Ending Cylinder	Numeric Value
8	4	Relative Sector	Numeric Value
12	4	Total Sector	Numeric Value

Table 3-2 : Partition Table

(BP 0) Boot Indicator
This field shall be recorded as 80h if SD Memory Card is used for boot. Otherwise, this field shall be recorded as 00h.

(BP 1) Starting Head
This field shall specify the starting head of the partition.

(BP 2 and 3) Starting Sector / Starting Cylinder
This field shall specify the starting sector and cylinder of the partition. 6 bits (Bit 0 to Bit 5 in BP 2) in this field shall be used for starting sector. 10 bits (Bit 6 and Bit 7 in BP 2, Bit 0 to Bit 7 in BP 3) in this field shall be used for starting cylinder.

(BP 4) System ID
This field shall be determined only by partition length regardless of file system types (FAT12/FAT16). It shall be recorded as 01h if the partition size is less than 32680 sectors. And it shall be recorded as 04h if the one is less than 65536 sectors. Otherwise, it shall be recorded as 06h.

(BP 5) Ending Head
This field shall specify the ending head of the partition.

(BP 6 and 7) Ending Sector / Ending Cylinder
This field shall specify the ending sector and cylinder of the partition. 6 bits (Bit 0 to Bit 5 in BP 6) in this field shall be used for ending sector. 10 bits (Bit 6 and Bit 7 in BP 6, Bit 0 to Bit 7 in BP 7) in this field shall be used for ending cylinder.

(BP 8 to 11) Relative Sector
This field shall specify the number of sectors existing before the starting sector of this partition.

(BP 12 to 15) Total Sector
This field shall specify the number of sectors on the partition.

3.1.4. Arrangement of the System Area

3.1.4.1. System Area

The System Area shall contain the Partition Boot Sector, the Root Directory and the File Allocation Table (FAT) recorded twice.

3.1.4.2. Partition Boot Sector

The first sector of the System Area shall contain the Partition Boot Sector including the FDC Descriptor. The FDC Descriptor shall contain the parameters for the partition.

3.1.4.3. File Allocation Table (FAT)

The FAT shall contain a Format Identifier and some entries, each of which indicates cluster of the User Area. These entries shall be numbered consecutively starting with 2 and the Entry Number shall be equal to the Cluster Number of the corresponding cluster.

Each entry in the FAT shall indicate the status of the corresponding cluster. The FAT entries shall be used to identify the set of clusters that are allocated to each file.

3.1.4.4. Root Directory

The Root Directory shall be recorded in the System Area following the second occurrence of the FAT.

3.2. File Structure

3.2.1. Partition Boot Sector

There is a Partition Boot Sector at the head of a partition and it contains an FDC Descriptor or an Extended FDC Descriptor. The FDC Descriptor and the Extended FDC Descriptor are compliant to ISO/IEC 9293. The Extended FDC is used for the default.

BP	Length (bytes)	Field Name	Contents
0	3	Jump Command	bytes
3	8	Creating System Identifier	a-characters
11	2	Sector Size	Numeric Value
13	1	Sectors per Cluster	Numeric Value
14	2	Reserved Sector Count	Numeric Value
16	1	Number of FATs	Numeric Value
17	2	Number of Root-directory Entries	Numeric Value
19	2	Total Sectors	Numeric Value
21	1	Medium Identifier	F8h
22	2	Sectors per FAT	Numeric Value
24	2	Sectors per Track	Numeric Value
26	2	Number of Sides	Numeric Value
28	2	(Reserved for future standardization)	0000h
30	480	(Reserved for system use)	Not Restricted
510	2	Signature Word	55h, AAh

Table 3-3 : FDC Descriptor

BP	Length (bytes)	Field Name	Contents
0	3	Jump Command	bytes
3	8	Creating System Identifier	a-characters
11	2	Sector Size	Numeric Value
13	1	Sectors per Cluster	Numeric Value
14	2	Reserved Sector Count	Numeric Value
16	1	Number of FATs	Numeric Value
17	2	Number of Root-directory Entries	Numeric Value
19	2	Total Sectors	Numeric Value
21	1	Medium Identifier	F8h
22	2	Sectors per FAT	Numeric Value
24	2	Sectors per Track	Numeric Value
26	2	Number of Sides	Numeric Value
28	4	Number of Hidden Sectors	Numeric Value
32	4	Total Sectors	Numeric Value
36	1	Physical Disk Number	80h
37	1	Reserved	00h
38	1	Extended Boot Record Signature	29h
39	4	Volume ID Number	Numeric Value
43	11	Volume Label	d-characters
54	8	File System Type	d-characters
62	448	(Reserved for system use)	Not Restricted
510	2	Signature Word	55h, AAh

Table 3-4 : Extended FDC Descriptor

(BP 0 to 2) Jump Command
This field shall specify the jump command to the boot program. It shall be recorded as EBh (BP 0), XXh (BP 1) and 90h (BP 2), or E9h (BP 0), XXh (BP 1) and XXh (BP 2). XXh means that the value is not specified in this specification.

(BP 3 to 10) Creating System Identifier
This field shall specify identification for the system. This field shall be recorded using a-characters and according to ISO/IEC 9293 9.

(BP 11 and 12) Sector Size
This field shall specify the size of a sector in bytes. It shall be recorded as the number 512.

(BP 13) Sectors per Cluster
This field shall specify the number of sectors per cluster. It shall be recorded the following number: 1, 2, 4, 8, 16, 32 or 64. The Cluster Size shall be determined taking the erase block size of physical layer into account. This field should be recorded complying with the recommendation of this specification. Detailed explanations for the recommendation are described in the later section.

(BP 14 and 15) Reserved Sector Count
This field shall specify the number of sectors reserved for system use. It shall be recorded as the number 1.

(BP 16) Number of FATs

This field shall specify the number of FATs. It shall be recorded as the number 2.

(BP 17 and 18) Number of Root-directory Entries
This field shall specify the number of entries in the Root Directory. It shall be recorded as the number 512.

(BP 19 and 20) Total Sectors
This field shall specify the number of sectors on the partition. It shall be recorded according to ISO/IEC 9293 9.

(BP 21) Medium Identifier
This field shall be recorded as F8h for this specification.

(BP 22 and 23) Sectors per FAT
This field shall specify the number of sectors that shall be occupied by each FAT. It shall be recorded according to ISO/IEC 9293 9.

(BP 24 and 25) Sectors per Track
This field shall specify the number of sectors in each track. This parameter depends on the SD Memory Card's parameter. It shall be recorded according to ISO/IEC 9293 9. This field should be recorded complying with the recommendation of this specification. Detailed explanations for the recommendation are described in the later section.

(BP 26 and 27) Number of Sides
This field shall specify the number of sides that can be recorded. This parameter depends on the SD Memory Card's parameter. It shall be recorded according to ISO/IEC 9293 9. This field should be recorded complying with the recommendation of this specification. Detailed explanations for the recommendation are described in the later section.

(BP 28 and 29) Field reserved for future standardization
This field shall be reserved for future standardization. It shall contain only ZEROs.

(BP 30 to 509) Field reserved for system use
This field shall be reserved for system use. It shall not be specified in this specification.

(BP 510 and 511) Signature Word
This field shall be recorded as 55h (BP 510) and AAh (BP 511).

(Extended FDC Descriptor BP 28 to 31) Number of Hidden Sectors
This field shall specify the number of sectors existing before the starting sector of this partition.

(Extended FDC Descriptor BP 32 to 35) Total Sectors
This field shall specify the number of sectors on the partition if the field in BP 19 and 20 is recorded as ZEROs. It shall be recorded according to ISO/IEC 9293 9.

(Extended FDC Descriptor BP 36) Physical Disk Number
This field shall specify the BIOS physical disk number. This field shall be recorded as 80h.

(Extended FDC Descriptor BP 37) Reserved
This field shall be reserved for future standardization. It shall be recorded as ZEROs. It shall be recorded according to ISO/IEC 9293 9. However, since a value other than 00h may be set on other devices, 00h shall not be expected at the time of operation.

(Extended FDC Descriptor BP 38) Extended Boot Record Signature
This field shall be used to identify the descriptor type in the Extended FDC Descriptor when either BP 19 or BP 20 is not recorded as ZEROs. This field shall be recorded as 29h.

(Extended FDC Descriptor BP 39 to 42) Volume ID Number
This field shall specify the volume identification number. It shall be recorded according to ISO/IEC 9293 9.

(Extended FDC Descriptor BP 43 to 53) Volume Label
This field shall specify the volume label. It shall be recorded according to ISO/IEC 9293 9.

(Extended FDC Descriptor BP 54 to 61) File System Type
This field shall specify the type of the file system. It shall be recorded according to ISO/IEC 9293 9.

(Extended FDC Descriptor BP 62 to 509) Field reserved for system use
This field shall be reserved for system use. It shall not be specified in this specification.

(Extended FDC Descriptor BP 510 and 511) Signature Word
This field shall be recorded as 55h (BP 510) and AAh (BP 511).

3.2.2. File Allocation Table

The File Allocation Table supports both the 12-bit FAT and the 16-bit FAT. The FAT structure is compliant to ISO/IEC 9293. The FAT type shall be determined by the number of clusters that depends on the parameter from the physical layer. If the cluster number is less than 4085, FAT12 shall be used. Otherwise, FAT16 shall be used. The first byte of the FAT shall specify the format identifier and be recorded F8h. In case of FAT12, the byte 2 and 3 shall be recorded as FFh each. In case of FAT16, the byte 2, 3 and 4 shall be recorded as FFh each. The sectors of the FAT may include unused area, because the number of clusters shall determine the FAT size in byte. This unused area shall be recorded as ZEROs.

FAT Entry Value		Contents
FAT12	FAT16	
000h	0000h	Indicates that the corresponding cluster is not in use and may be allocated to a file or a directory.
002h to MAX	0002h to MAX	Indicates that the corresponding cluster is already allocated. The value of the entry is the cluster number of the next cluster following this corresponding cluster. Max shall be the Maximum Cluster Number.
MAX+1 to FF6h	MAX+1 to FFF6h	Shall be reserved for future standardization and shall not be used.
FF7h	FFF7h	Indicates that the corresponding cluster has a defective cluster.
FF8h to FFFh	FFF8h to FFFFh	The corresponding cluster is already allocated, and it is the final cluster of the file.

Table 3-5 : FAT Entry Value

3.2.3. File Directories

3.2.3.1. Characteristics

A Directory is a Descriptor that shall contain a set of Directory entries each of which identifies a file, a Volume Label, another Directory or is unused. A Directory can contain the 65536 Directory entries.
The Short File Name entry storing file name as 8.3 format shall be supported as mandatory. And the character code permitted by the ISO/IEC 9293 can be used for this type of Directory entry.

Moreover, hosts can also support Long File Name (LFN) entry as optional. If host doesn't support Long File Name, it may ignore Long File Name entries, and refers to only the file name of 8.3 format that is stored with the LFN.

NOTE: The Long File Name feature is described in Appendix C.3.

3.2.3.2. Directory Entry Types

Directory entries shall contain descriptive information about the files recorded on the partition. There are some types of these entries as below:

- File Entry
  A File Entry shall specify information of a file.

- Volume Label Entry
  A Volume Label Entry shall specify the volume label of the partition.

- Sub-directory Pointer Entry
  A Sub-directory Pointer Entry shall specify information of a directory.

- Sub-directory Identifier Entry
  A Sub-directory Identifier Entry shall identify a file as a Sub-directory.

- Sub-directory Parent Pointer Entry
  A Sub-directory Parent Pointer Entry shall specify information of its parent directory.

- Not-currently-in-use Entry
  A Not-currently-in-use Entry shall specify the entry is not used and able to allocate.

- Never-used Entry
  A Never-used Entry shall specify the end of the directory. It shall not appear before any other type of Directory entry.

These entries shall be recorded according to ISO/IEC 9293 11.

3.2.3.3. General Definition of Directory Entry Fields

The following table indicates the structure of the Directory entry field.

BP	Length (bytes)	Field Name	Contents
0	8	Name	Depends on entry type
8	3	Name Extension	d-characters
11	1	Attributes	8 bits
12	10	Reserved Field	bytes
22	2	Time Recorded	Numeric Value
24	2	Date Recorded	Numeric Value
26	2	Starting Cluster Number	Numeric Value
28	4	File Length	Numeric Value

Table 3-6 : Directory Entry Field

(BP 0 to 7) Name
The content and the description of this field shall depend on the entry type. It shall be recorded according to ISO/IEC 9293 11.

(BP 8 to 10) Name Extension
The content and the description of this field shall depend on the entry type. The content of this field shall be d-characters. It shall be recorded according to ISO/IEC 9293 11.

(BP 11) Attributes
This field shall specify the attributes of the entry. It shall be recorded according to ISO/IEC 9293 11.

(BP 12 to 21) Reserved Field
The content of this field shall depend on the entry type. It should be recorded as ZEROs by default. However, some timestamps (Created Time Tenth, Created Time, Created Date, and Last Access Date) can be stored in this field as optional. Detailed explanations of these timestamps are described in the section “4.2.4.2. Directory Entry Fields”. If the host doesn’t support these timestamps, 00h shall not be expected at the time of operation.

(BP 22 and 23) Time Recorded
This field shall contain a 16-bit integer representing a time. It shall be recorded according to ISO/IEC 9293 11.

(BP 24 and 25) Date Recorded
This field shall contain a 16-bit integer representing a date. It shall be recorded according to ISO/IEC 9293 11.

(BP 26 and 27) Starting Cluster Number
The content of this field shall depend on the entry type. It shall be recorded according to ISO/IEC 9293 11.

(BP 28 to 31) File Length
The content of this field shall depend on the entry type. It shall be recorded according to ISO/IEC 9293 11.

3.2.3.4. User Area

The User Area shall be organized into clusters. Each cluster has a Cluster Number respectively. The first cluster in the User Area is corresponding to Cluster Number 2.

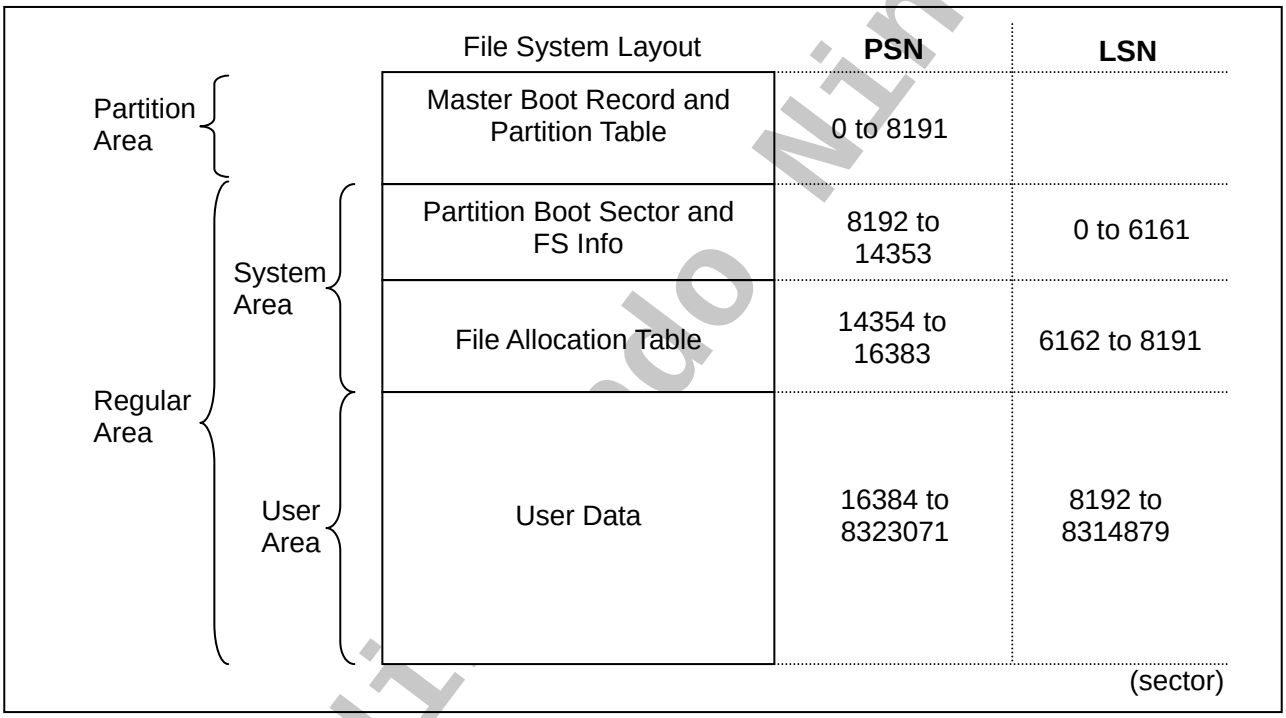
Although it is available to read/write by the sector, it is necessary to transact reading/writing with the unit whose minimum size is the same as that of the recommended reading/writing at the physical layer. Other than that, there are no special restrictions for the SD Memory Card file system.

4. FAT32 File System

4.1. Volume Structure

The volume structure of the High Capacity SD Memory Card is specified in this section. For the identification of the Data Area as a partition, Master Boot Record and Partition Table exist in the first sector of the card as well as the Standard Capacity SD Memory Card. The FAT32 file system shall be applied to the High Capacity SD Memory Card.

NOTE: FAT32 Specification described in this section includes portions of Microsoft FAT Specification, the copyright of which is owned by Microsoft but licensed to SD Card Association.



PSN : Physical Sector Number

LSN : Logical Sector Number

Figure 4-1 : Example of Volume Structure for Data Area

4.1.1. Arrangement of the Data Area

The arrangement of the Data Area for High Capacity SD Memory Card is the same one with the arrangement for the Standard Capacity SD Memory Card.
That is, this type of the card has the concept of Physical Address, Physical Sector Number and Logical Sector Number, and Data Area is divided into the Partition Area and the Regular Area too.

The detailed explanation is described in the section “3.1.1. Arrangement of the Data Area”.

4.1.2. Arrangement of the User Area

The arrangement of the User Area for High Capacity SD Memory Card is the same one with the arrangement for the Standard Capacity SD Memory Card.

That is, the User Area shall be organized into units of Cluster, and there are three statuses for Cluster.

The detailed explanation is described in the section “3.1.2. Arrangement of the User Area”.

4.1.3. Arrangement of the Partition Area

The first sector of the Data Area has a Master Boot Record that includes executable codes and Partition Table that includes the information to identify the partition as well as the Standard Capacity SD Memory Card.

BP	Length (bytes)	Field Name	Contents
0	446	Master Boot Record	Not Restricted
446	16	Partition Table (partition1)	Refer to Table 4-2
462	16	Partition Table (partition2)	All 00h
478	16	Partition Table (partition3)	All 00h
494	16	Partition Table (partition4)	All 00h
510	2	Signature Word	55h, AAh

Table 4-1 : Master Boot Record and Partition Table

(BP 0 to 445) Master Boot Record
The content of this field is not specified by this specification.

(BP 446 to 461) Partition Table (partition1)
This field shall specify the information of first partition in the volume. This partition means Regular Area that user can access without mutual authentication. It shall be recorded according to Table 4-2.

(BP 462 to 477) Partition Table (partition2)
This field shall be recorded as ZEROs, as a volume shall consist of single Regular Area.

(BP 478 to 493) Partition Table (partition3)
This field shall be recorded as ZEROs, as a volume shall consist of single Regular Area.

(BP 494 to 509) Partition Table (partition4)
This field shall be recorded as ZEROs, as a volume shall consist of single Regular Area.

(BP 510 and 511) Signature Word
This field shall be recorded as 55h (BP 510) and AAh (BP 511).

BP	Length (bytes)	Field Name	Contents
0	1	Boot Indicator	00h or 80h
1	1	Starting Head	Numeric Value
2	2	Starting Sector / Starting Cylinder	Numeric Value
4	1	System ID	0Bh or 0Ch
5	1	Ending Head	Numeric Value
6	2	Ending Sector / Ending Cylinder	Numeric Value
8	4	Relative Sector	Numeric Value
12	4	Total Sector	Numeric Value

Table 4-2 : Partition Table

(BP 0) Boot Indicator
This field shall be recorded as 80h if SD Memory Card is used for boot. Otherwise, this field shall be recorded as 00h.

(BP 1) Starting Head
This field shall specify the starting head of the partition. If the starting location of the partition exceeds 8032.5MB (8,422,686,720Bytes), this field shall be recorded as FEh.

(BP 2 and 3) Starting Sector / Starting Cylinder
This field shall specify the starting sector and cylinder of the partition. 6 bits (Bit 0 to Bit 5 in BP 2) in this field shall be used for starting sector. 10 bits (Bit 6 and Bit 7 in BP 2, Bit 0 to Bit 7 in BP 3) in this field shall be used for starting cylinder. If the starting location of the partition exceeds 8032.5MB (8,422,686,720Bytes), this field shall be recorded as FFFFh.

(BP 4) System ID
This field shall be recorded as 0Bh if the ending location of the partition is less than 8032.5MB (8,422,686,720Bytes). Otherwise, this field shall be recorded as 0Ch.

(BP 5) Ending Head
This field shall specify the ending head of the partition. If the ending location of the partition exceeds 8032.5MB (8,422,686,720Bytes), this field shall be recorded as FEh.

(BP 6 and 7) Ending Sector / Ending Cylinder
This field shall specify the ending sector and cylinder of the partition. 6 bits (Bit 0 to Bit 5 in BP 6) in this field shall be used for ending sector. 10 bits (Bit 6 and Bit 7 in BP 6, Bit 0 to Bit 7 in BP 7) in this field shall be used for ending cylinder. If the ending location of the partition exceeds 8032.5MB (8,422,686,720Bytes), this field shall be recorded as FFFFh.

(BP 8 to 11) Relative Sector
This field shall specify the number of sectors existing before the starting sector of this partition. The starting sector of the partition should be aligned to the Boundary Unit specified in Appendix C.2. Detailed explanations for the alignment are described in the later section.

(BP 12 to 15) Total Sector
This field shall specify the number of sectors on the partition.

4.1.4. Arrangement of the System Area

4.1.4.1. System Area

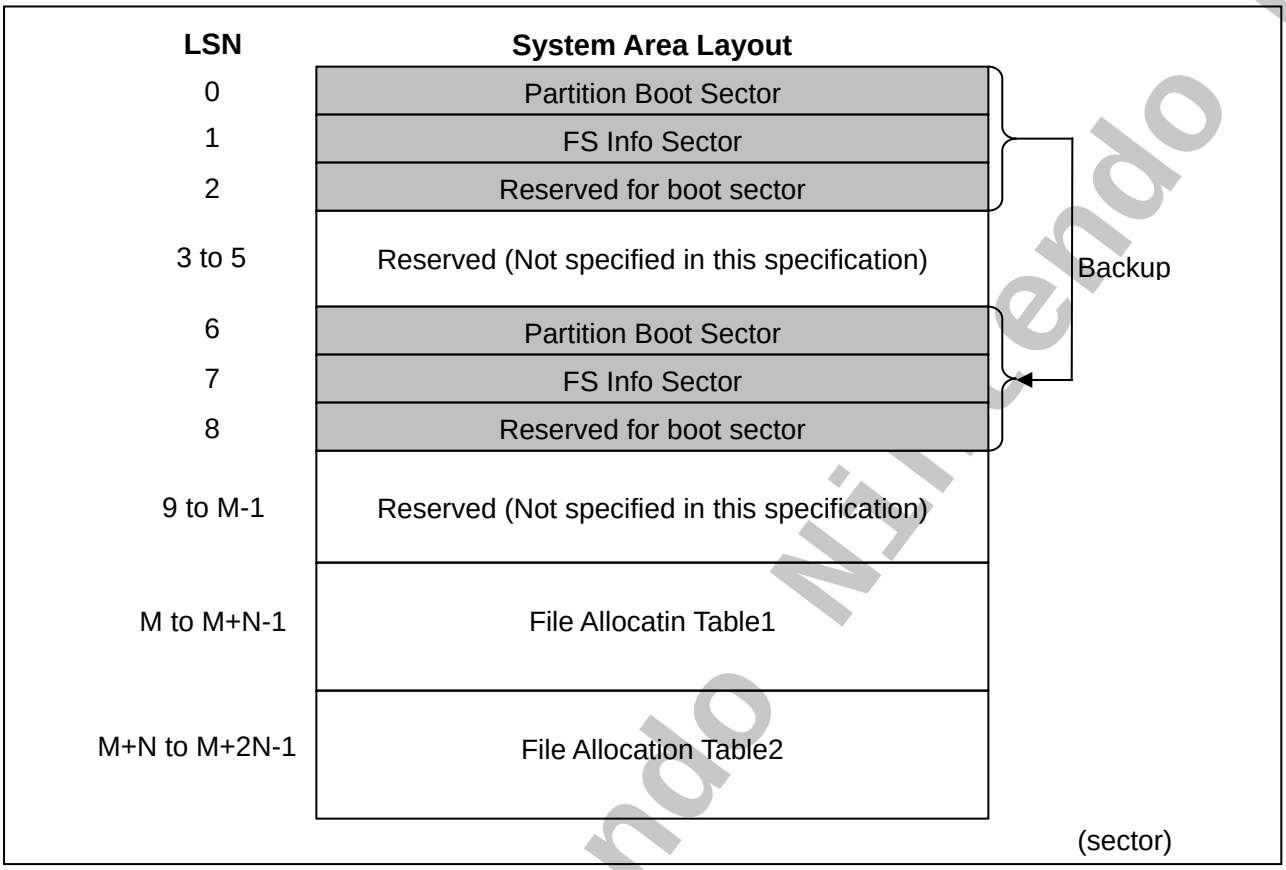
The System Area shall contain the Partition Boot Sector, the FS Info Sector and the File Allocation Table (FAT). These sectors shall be recorded twice on the card for the sake of backup. FAT32 has no reserved area for Root Directory. It shall be located in the User Data Area like the other directories.

4.1.4.2. Partition Boot Sector and FS Info Sector

The first sector of the System Area shall contain the Partition Boot Sector. It shall contain the parameters for the partition like cluster size, partition size, etc.
The second sector of the System Area shall contain the FS Info Sector. It shall contain the additional information like free cluster count, next free cluster, etc.
The third sector of the System Area shall be reserved for boot sectors. This sector has the signature word. That is, 55h shall be recorded in BP 510, and AAh shall be recorded in BP 511. The other bytes in this third sector shall be reserved for system use and they shall not be specified in this specification.

The size of this area shall be recorded in the Reserved Sector Count field in the Partition Boot Sector. In this specification, the size should be used for the alignment of User Data area. For more details, see Appendix C.2.

All 3 sectors described above shall be recorded twice on the card. They shall be recorded in the Logical Sector Number (LSN) 0 to 2 and 6 to 8 as follows.



M : The number of reserved sector count
N : The number of sectors per FAT

Figure 4-2 : System Area Layout

4.1.4.3. File Allocation Table (FAT)

The FAT shall contain a Format Identifier and some entries, each of which indicates cluster of the User Area. These entries shall be numbered consecutively starting with 2 and the Entry Number shall be equal to the Cluster Number of the corresponding cluster.

Each entry in the FAT shall indicate the status of the corresponding cluster. The FAT entries shall be used to identify the set of clusters that are allocated to each file.

4.2. File Structure

4.2.1. Partition Boot Sector

There is a Partition Boot Sector at the head of a partition. For FAT32 file system, some fields are extended from Extended FDC Descriptor of ISO/IEC 9293. Partition Boot Sector shall be recorded twice as described before.

BP	Length (bytes)	Field Name	Contents
0	3	Jump Command	bytes
3	8	Creating System Identifier	bytes
11	2	Sector Size	Numeric Value
13	1	Sectors per Cluster	Numeric Value
14	2	Reserved Sector Count	Numeric Value
16	1	Number of FATs	Numeric Value
17	2	Number of Root-directory Entries	Numeric Value
19	2	Total Sectors	Numeric Value
21	1	Medium Identifier	F8h
22	2	Sectors per FAT	Numeric Value
24	2	Sectors per Track	Numeric Value
26	2	Number of Sides	Numeric Value
28	4	Number of Hidden Sectors	Numeric Value
32	4	Total Sectors	Numeric Value
36	4	Sectors per FAT for FAT32	Numeric Value
40	2	Extension Flag	Numeric Value
42	2	FS Version	0000h
44	4	Root Cluster	Numeric Value
48	2	FS Info	Numeric Value
50	2	Backup Boot Sector	Numeric Value
52	12	Reserved	All 00h
64	1	Physical Disc Number	80h
65	1	Reserved	00h
66	1	Extended Boot Record Signature	29h
67	4	Volume ID Number	Numeric Value
71	11	Volume Label	bytes
82	8	File System Type	bytes
90	420	(Reserved for system use)	Not Restricted
510	2	Signature Word	55h, AAh

Table 4-3 : Partition Boot Sector

(BP 0 to 2) Jump Command
This field shall specify the jump command to the boot program. It shall be recorded as EBh (BP 0), XXh (BP 1) and 90h (BP 2), or E9h (BP 0), XXh (BP 1) and XXh (BP 2). XXh means that the value is not specified in this specification.

(BP 3 to 10) Creating System Identifier
This field shall specify identification for the system.

(BP 11 and 12) Sector Size
This field shall specify the size of a sector in bytes. It shall be recorded as the number 512.

(BP 13) Sectors per Cluster
This field shall specify the number of sectors per cluster. It shall be recorded the following number: 1, 2, 4, 8, 16, 32 or 64. The Cluster Size shall be determined taking the erase block size of physical layer into account. This field should be recorded complying with the recommendation of this specification. Detailed explanations for the recommendation are described in the later section.

(BP 14 and 15) Reserved Sector Count
This field shall specify the number of sectors reserved for system use. This field should be used for the alignment of User Data area. Detailed explanations for the alignment are described in the later section.

(BP 16) Number of FATs
This field shall specify the number of FATs. It shall be recorded as the number 2.

(BP 17 and 18) Number of Root-directory Entries
This field shall specify the number of entries in the Root Directory for FAT12 / FAT16 file system. FAT32 file system shall not use this field and it shall be recorded as the number 0.

(BP 19 and 20) Total Sectors
This field shall specify the number of sectors on the partition whose size is 65535 sectors or less. FAT32 file system shall not use this field and it shall be recorded as the number 0.

(BP 21) Medium Identifier
This field shall be recorded as F8h for this specification.

(BP 22 and 23) Sectors per FAT
This field shall specify the number of sectors that shall be occupied by each FAT for FAT12 / FAT16 file system. FAT32 file system shall not use this field and it shall be recorded as the number 0.

(BP 24 and 25) Sectors per Track
This field shall specify the number of sectors in each track. This parameter depends on the SD Memory Card's parameter. This field should be recorded complying with the recommendation of this specification. Detailed explanations for the recommendation are described in the later section.

(BP 26 and 27) Number of Sides
This field shall specify the number of sides that can be recorded. This parameter depends on the SD Memory Card's parameter. This field should be recorded complying with the recommendation of this specification. Detailed explanations for the recommendation are described in the later section.

(BP 28 to 31) Number of Hidden Sectors
This field shall specify the number of sectors existing before the starting sector of this partition. The starting sector of the partition should be aligned to the Boundary Unit specified in Appendix C.2. Detailed explanations for the alignment are described in the later section.

(BP 32 to 35) Total Sectors
This field shall specify the number of sectors on the partition if the field in BP 19 and 20 is recorded as ZEROs. FAT32 file system shall not use BP 19 and 20. Therefore, this field shall not be ZEROs.

(BP 36 to 39) Sectors per FAT for FAT32
This field shall specify the number of sectors that shall be occupied by each FAT for FAT32 file system.

(BP 40 and 41) Extension Flag
This field shall specify the status of FAT mirroring.

Bits 0 to 3 : These bits shall specify zero-based number of active FAT. These bits are valid only if mirroring is disabled.
Bits 4 to 6 : Reserved.
Bits 7 : 0 means the FAT is mirrored at runtime into all FATs.
1 means only one FAT is active; it is the one referenced in bits 0 to 3.
Bits 8 to 15 : Reserved.

(BP 42 and 43) FS Version
This field shall specify the version number of FAT32 file system. High byte is major revision number. Low byte is minor revision number. This field shall be recorded as 0000h which shows the version 0:0.

(BP 44 to 47) Root Cluster
This field shall specify the cluster number of the first cluster of the root directory. This value should be 2 or the first usable (not defective) cluster available thereafter.

(BP 48 and 49) FS Info
This field shall specify the sector number of the area where FS Info Sector structure is recorded in. Here, the sector number means the offset from the head sector of the System Area. That is, the sector number 0 means the location of the head sector of the System Area.
This field shall be recorded as the sector number 1. It shows FS Info Sector shall be located in the second sector of the System Area.

(BP 50 and 51) Backup Boot Sector
This field shall specify the starting sector number of the area where the copy of the boot sectors is recorded in. Here, the sector number means the offset from the head sector of the System Area. That is, the sector number 0 means the location of the head sector of the System Area.
This field shall be recorded as the sector number 6. It shows the copy of the boot sectors shall be recorded in the area starting the sixth sector of the System Area.

(BP 52 to 63) Reserved
This field shall be reserved for future standardization. It shall be recorded as ZEROs. However, since a value other than 00h may be set on other devices, 00h shall not be expected at the time of operation.

(BP 64) Physical Disk Number
This field shall specify the BIOS physical disk number. This field shall be recorded as 80h.

(BP 65) Reserved
This field shall be reserved for future standardization. It shall be recorded as ZEROs. However, since a value other than 00h may be set on other devices, 00h shall not be expected at the time of operation.

(BP 66) Extended Boot Record Signature
This field shall be used to identify this structure. This field shall be recorded as 29h.

(BP 67 to 70) Volume ID Number
This field shall specify the volume identification number.

(BP 71 to 81) Volume Label
This field shall specify the volume label.

(BP 82 to 89) File System Type
This field shall specify the type of the file system. This field shall be recorded as “FAT32 ”. 20h shall be recorded in the last 3 bytes.

(BP 90 to 509) Field reserved for system use
This field shall be reserved for system use. It shall not be specified in this specification.

(BP 510 and 511) Signature Word
This field shall be recorded as 55h (BP 510) and AAh (BP 511).

4.2.2. FS Info Sector

There is a FS Info Sector in the next sector of the Partition Boot Sector. This structure doesn't exist on the FAT12 / FAT16 volume. FS Info Sector shall be recorded twice as described before.

BP	Length (bytes)	Field Name	Contents
0	4	Lead Signature	52h, 52h, 61h, 41h
4	480	Reserved	All 00h
484	4	Structure Signature	72h, 72h, 41h, 61h
488	4	Free Cluster Count	Numeric Value
492	4	Next Free Cluster	Numeric Value
496	12	Reserved	All 00h
508	4	Trail Signature	00h, 00h, 55h, AAh

Table 4-4 : FS Info Sector

(BP 0 to 3) Lead Signature
This field shall be used to validate the beginning of the FS Info Sector. It shall be recorded as 52h (BP 0), 52h (BP 1), 61h (BP 2) and 41h (BP 3).

(BP 4 to 483) Reserved
This field shall be reserved for future standardization. It shall be recorded as ZEROs. However, since a value other than 00h may be set on other devices, 00h shall not be expected at the time of operation.

(BP 484 to 487) Structure Signature
This field shall be used as an additional signature validating the integrity of the FS Info Sector. It shall be recorded as 72h (BP 484), 72h (BP 485), 41h (BP 486) and 61h (BP 487).

(BP 488 to 491) Free Cluster Count
This field shall contain the last known free cluster count on the volume. The value FFFFFFFFh indicates the free count is not known. The contents of this field shall be validated at volume mount (and subsequently maintained in memory by the file system driver implementation). It is recommended that this field contain an accurate count of the number of free clusters at volume dismount (during controlled dismount/shutdown).

(BP 492 to 495) Next Free Cluster
This field shall contain the cluster number of the first available (free) cluster on the volume. The value FFFFFFFFh indicates that there exists no information about the first available (free) cluster. The contents of this field shall be validated at volume mount. It is recommended that this field be appropriately updated at volume dismount (during controlled dismount/shutdown).

(BP 496 to 507) Reserved
This field shall be reserved for future standardization. It shall be recorded as ZEROs. However, since a value other than 00h may be set on other devices, 00h shall not be expected at the time of operation.

(BP 508 to 511) Trail Signature
This field shall be used to validate the integrity of the data in the FS Info Sector. It shall be recorded as 00h (BP 508), 00h (BP 509), 55h (BP 510) and AAh (BP 511).

4.2.3. File Allocation Table

The File Allocation Table consists of 32-bit FAT Entries. Each FAT Entry value is described in the following table. The high 4 bits of a FAT32 FAT Entry are reserved. All FAT implementations shall preserve the current value of the high 4 bits when modifying any FAT32 FAT entry except during volume initialization (formatting) when the entire FAT table shall be set to 0. And no FAT32 volume should ever be configured containing cluster numbers available for allocation >= FFFFFF7h.

FAT Entry Value (FAT32)	Contents
0000000h	Indicates that the corresponding cluster is not in use and may be allocated to a file or a directory.
0000002h to MAX	Indicates that the corresponding cluster is already allocated. The value of the entry is the cluster number of the next cluster following this corresponding cluster. Max shall be the Maximum Cluster Number.
MAX+1 to FFFFFF6h	Shall be reserved for future standardization and shall not be used.
FFFFFF7h	Indicates that the corresponding cluster has a defective cluster.
FFFFFF8h to FFFFFFFh	The corresponding cluster is already allocated, and it is the final cluster of the file. This entry value is called EOC mark (End Of Clusterchain).

Table 4-5 : FAT Entry Value

The first two entries (8 bytes) in the FAT are reserved. These 8 bytes of the FAT shall be recorded as F8h (BP 0), FFh (BP 1), FFh (BP 2), 0Fh (BP 3), FFh (BP 4), FFh (BP 5), FFh (BP 6) and 0Fh (BP 7). However, the file system driver may use the specific two bits of BP 7 for dirty volume flags as follows.

Bit 3 (Clean Shutdown Bit) :

If bit is 1, the volume is “clean”. The volume can be mounted for access.

If bit is 0, the volume is “dirty” indicating that a FAT file system driver was unable to dismount the volume properly (during a prior mount operation). The volume contents should be scanned for any damage to file system metadata.

Bit 2 (Hard Error Bit) :

If bit is 1, no disk read/write errors were encountered.

If bit is 0, the file system driver implementation encountered a disk I/O error on the volume the last time it was mounted, which is an indicator that some sectors may have gone bad. The volume contents should be scanned with a disk repair utility that does surface analysis on it looking for new bad sectors.

The sectors of the FAT may include unused area, because the number of clusters shall determine the FAT size in byte. This unused area shall be recorded as ZEROs.

4.2.4. File Directories

4.2.4.1. Characteristics

A Directory is a Descriptor that shall contain a set of Directory entries each of which identifies a file, a Volume Label, another Directory or is unused. A Directory can contain the 65536 Directory entries.
The format of the file name for the Directory entries shall support 8.3 format as mandatory. And hosts can also support Long File Name (LFN) as optional. If host doesn't support Long File Name, it may ignore Long File Name entries, and refers to only the file name of 8.3 format that is stored with the LFN.

NOTE: The Long File Name feature is described in Appendix C.3.

As described in the previous section, on FAT12 and FAT16 volumes, the root directory shall immediately follow the last file allocation table. And the size is a fixed size in sectors calculated from the Number of Root-directory Entries value stored in Partition Boot Sector.

On volumes formatted FAT32, the root directory can be of variable size. The location of the first cluster of the root directory on the FAT32 volume is stored in the Root Cluster field of Partition Boot Sector. Only the root directory can contain a Volume Label Entry. There is no name for the root directory (on most operating system implementations, the implied name "\" is used). Further, the root directory does not have any associated date/time stamps. Lastly, the root directory does not contain either the dot and dotdot entries.

4.2.4.2. Directory Entry Fields

The following table indicates the structure of the Directory entry field.

BP	Length (bytes)	Field Name	Contents
0	11	Name and Name Extension	bytes
11	1	Attributes	8 bits
12	1	Reserved for NT	byte
13	1	Created Time Tenth	Numeric Value
14	2	Created Time	Numeric Value
16	2	Created Date	Numeric Value
18	2	Last Access Date	Numeric Value
20	2	Starting Cluster Number High	Numeric Value
22	2	Time Recorded	Numeric Value
24	2	Date Recorded	Numeric Value
26	2	Starting Cluster Number Low	Numeric Value
28	4	File Length	Numeric Value

Table 4-6 : Directory Entry Field

(BP 0 to 10) Name and Name Extension
This field shall contain the 11 character ("short") name of the file or sub-directory described by the corresponding entry containing the field. This field is logically comprised of two components:

- The 8-character man part of the name
- The 3-character extension

Each of the above two components are "trailing space padded" if required (using value: 20h).

Note the following:

1. An implied '.' character separates the main part of the name from the extension. The "." separator character is never stored in this field.
2. If the first byte in this field is E5h, it indicates the directory entry is free (available). However, E5h is a valid KANJI lead byte value for the character set used in Japan. For KANJI character set based names, the value 05h is stored in this byte - if required - to represent E5h. If the FAT file system implementation reads this byte as 05h and if the character set used is KANJI, it shall perform the appropriate substitution in memory prior to returning the name to the application.
3. If the first byte in this field is 00h, it also indicates the directory entry is free (available). However, 00h set in the first byte in this field is an additional indicator that all directory entries following the current free entry are also free.
4. The first byte in this field cannot equal 20h (in other words, names cannot start with a space character).
5. All names in the directory shall be unique.

Restrictions on characters comprising the name

- Lower case characters are not allowed
- Illegal values for characters in the name are as follows:
  - Values less than 20h (except for the special case of 05h in the first byte in this field described earlier)
  - 22h, 2Ah, 2Bh, 2Ch, 2Eh, 2Fh, 3Ah, 3Bh, 3Ch, 3Dh, 3Eh, 3Fh, 5Bh, 5Ch, 5Dh, and 7Ch

(BP 11) Attributes
This field shall specify the attributes of the entry.

Name	Value	Content
ATTR_READ_ONLY	01h	The file cannot be modified – all modification requests shall fail with an appropriate error code value.
ATTR_HIDDEN	02h	The corresponding file or sub-directory shall not be listed unless a request is issued by the user/application explicitly requesting inclusion of "hidden files".
ATTR_SYSTEM	04h	The corresponding file is tagged as a component of the operating system. It shall not be listed unless a request is issued by the user/application explicitly requesting inclusion of "system files".
ATTR_VOLUME_ID	08h	The corresponding entry contains the volume label. Starting Cluster Number High field and Starting Cluster Number Low field shall always be 0 for the corresponding entry (representing the volume label) since no clusters can be allocated for this entry. Only the root directory can contain one entry with this attribute. No sub-directory shall contain an entry of this type. Entries representing long file names are exceptions to these rules.
ATTR_DIRECTORY	10h	The corresponding entry represents a directory (a child or sub-directory to the containing directory). File Length field for the corresponding entry shall always be 0 (even though clusters may have been allocated for the directory).
ATTR_ARCHIVE	20h	This attribute shall be set when the file is created, renamed, or modified. The presence of this attribute indicates that properties of the associated file have been modified. Backup utilities can utilize this information to determine the set of files that need to be backed up to ensure protection in case of media and other failure conditions.

Table 4-7 : Attributes

The upper two bits of the attribute byte are reserved and should always be set to 0 when a file is created and never modified or looked at after that.

When a new directory is created, the file system implementation shall ensure the following:

- The ATTR_DIRECTORY bit shall set in its Attributes field
- The File Length field shall be set to 0
- At least one cluster shall be allocated – the contents of the Starting Cluster Number Low field and

Starting Cluster Number High field shall refer to the first allocated cluster number

- If only a single cluster is allocated, the corresponding file allocation table entry shall indicate end-of-file condition
- The contents of the allocated cluster(s) shall be initialized to 0
- Except for the root directory, each directory shall contain the following two entries at the beginning of the directory:

  - The first directory entry shall have a directory name set to "."

    This dot entry refers to the current directory. Rules listed above for the Attributes field and File Length field shall be followed. Since the dot entry refers to the current directory (the one containing the dot entry), the contents of the Starting Cluster Number Low field and Starting Cluster Number High field shall be the same as that of the current directory. All date and time fields shall be set to the same value as that for the containing directory.

  - The second directory entry shall have the directory name set to ".."

    This dotdot entry refers to the parent of the current directory. Rules listed above for the Attributes field and File Length field shall be followed. Since the dotdot entry refers to the parent of the current directory (the one containing the dotdot entry), the contents of the Starting Cluster Number Low field and Starting Cluster Number High field shall be the same as that of the parent of the current directory. If the parent of the current directory is the root directory, the Starting Cluster Number Low field and Starting Cluster Number High field contents shall be set to 0. All date and time fields shall be set to the same value as that for the containing directory.

(BP 12) Reserved for NT
This field shall be reserved. This field shall be set to 0.

(BP 13) Created Time Tenth
This field shall specify the component of the file creation time. This field actually contains a count of tenths of a second. The valid value range is 0-199 inclusive. If this field is not supported, it should be set to 0 on file create and ignored on other file operations.

(BP 14 and 15) Created Time
This field shall specify the file creation time. The detailed format is described in the following section. If this field is not supported, it should be set to 0 on file create and ignored on other file operations.

(BP 16 and 17) Created Date
This field shall specify the file creation date. The detailed format is described in the following section. If this field is not supported, it should be set to 0 on file create and ignored on other file operations.

(BP 18 and 19) Last Access Date
This field shall specify the last access date. Last access is defined as a read or write operation performed on the file/directory described by this entry. This field shall be updated on file modification (write operation) and the date value shall be equal to Date Recorded field. The detailed format is described in the following section. If this field is not supported, it should be set to 0 on file create and ignored on other file operations.

(BP 20 and 21) Starting Cluster Number High
This field shall specify the high word of first data cluster number for file/directory described by this entry.

(BP 22 and 23) Time Recorded

This field shall specify the last modification (write) time. This field shall be equal to Created Time field at file creation. The detailed format is described in the following section. This field shall be supported by all hosts.

(BP 24 and 25) Date Recorded
This field shall specify the last modification (write) date. This field shall be equal to Created Date field at file creation. The detailed format is described in the following section. This field shall be supported by all hosts.

(BP 26 and 27) Starting Cluster Number Low
This field shall specify the low word of first data cluster number for file/directory described by this entry.

(BP 28 to 31) File Length
This field shall specify the 32-bit quantity containing size in bytes of file/directory described by this entry.

4.2.4.3. Date and Time Formats

The date and time formats in the directory entry are as follows.

(Date Format)
Contents of the date related field in the directory entry shall be in the following format:

- Bit positions 0 through 4 represent the day of the month (valid range: 1..31 inclusive)
- Bit positions 5 through 8 represent the month of the year (1 = January, 12 = December, valid range: 1..12 inclusive)
- Bit positions 9 through 15 are the count of years from 1980 (valid range: 0..127 inclusive allowing representation of years 1980 through 2107)

(Time Format)
Directory entry timestamps are 16-bit values with a granularity of 2 seconds. Contents of the timerelated field in the directory entry shall be in the following format:

- Bit positions 0 through 4 contain elapsed seconds – as a count of 2-second increments (valid range: 0..29 inclusive allowing representation of 0 through 58 seconds)
- Bit positions 5 through 10 represent number of minutes (valid range: 0..59 inclusive)
- Bit positions 11 through 15 represent hours (valid range: 0..23 inclusive)

The valid time range is from Midnight 00:00:00 to 23:59:58.

4.2.4.4. Difference from the FAT12 / FAT16 File Directories

File directories of FAT32 file system are similar to FAT12 / FAT16 file system (ISO/IEC 9293), however there are some differences. The main differences between them are as follows:

- For FAT32, the root directory isn't located in the special location, and it can be of variable size and is a cluster chain, just like any other directory is.
- The following fields in the directory entry are added : Reserved for NT, Created Time Tenth, Created Time, Created Date, Last Access Date, and Starting Cluster Number High

4.2.4.5. User Area

The User Area shall consist of some clusters. Each cluster has a Cluster Number respectively. The first cluster in the User Area is corresponding to Cluster Number 2.
Although it is available to read/write by the sector, it is recommended to transact reading/writing with the unit whose minimum size is the same as that of the recommended reading/writing at the physical layer. Other than that, there are no special restrictions for the SD Memory Card file system.

Appendix A (Normative)

A.1 Reference

This specification refers the following documents.

1) ISO/IEC 646:1991
Information technology – ISO 7-bit code character set for information interchange

2) ISO/IEC 9293:1994
Information technology – Volume and file structure of disk cartridges for information interchange

3) Microsoft FAT Specification: August 30 2005

Appendix B (Informative)

B.1 Abbreviations and Special Terms

BP	Byte Position within a certain field, starting with 0 from the first byte of the field
byte	a string of binary digits operated upon as a unit
defective sector	a sector that cannot be read or write
descriptor	a recorded structure containing information about the volume or a file
FAT	File Allocation Table
FDC	Flexible Disk Cartridge
file	a named collection of information
sector / block	a unit of data that can be accessed independently of other units on the SD Memory Card
partition	an extent of sectors within a volume
user	a person or other entity that causes the invocation of the services provided by an implementation
volume	a sector address space as specified in the relevant standard for recording
ZERO	represents a single bit with the value 0

B.2 Arithmetic Notation

ip(*x*)	The notation shall mean the integer part of *x*.
ceil(*x*)	The notation shall mean the minimum integer that is greater than *x*.
rem(*x*,*y*)	The notation shall mean the reminder of the integer division of *x* by *y.*

B.3 Character Strings

A value for a sequence of bytes for FAT12 / FAT16 file system may be specified by a quoted sequence of characters, encoded according to the ISO/IEC 646 standard.

B.4 Data Types

B.4.1 Numerical Values in One-byte Fields

A numerical value in a one-byte field shall be recorded as an 8-bit number in one-byte field.

B.4.2 Numerical Values in Two-byte Fields

A numerical value in a two-byte field shall be recorded in the little endian representation. It shall be recorded according to ISO/IEC 9293.

B.4.3 Numerical Values in Four-byte Fields

A numerical value in a four-byte field shall be recorded in the little endian representation. It shall be recorded according to ISO/IEC 9293.

B.4.4 Pairs of 12-bit Integers

A pair of 12-bit numbers shall be recorded in three-byte field according to ISO/IEC 9293.

Appendix C (Informative)

C.1 Appendix for FAT12/FAT16 File System

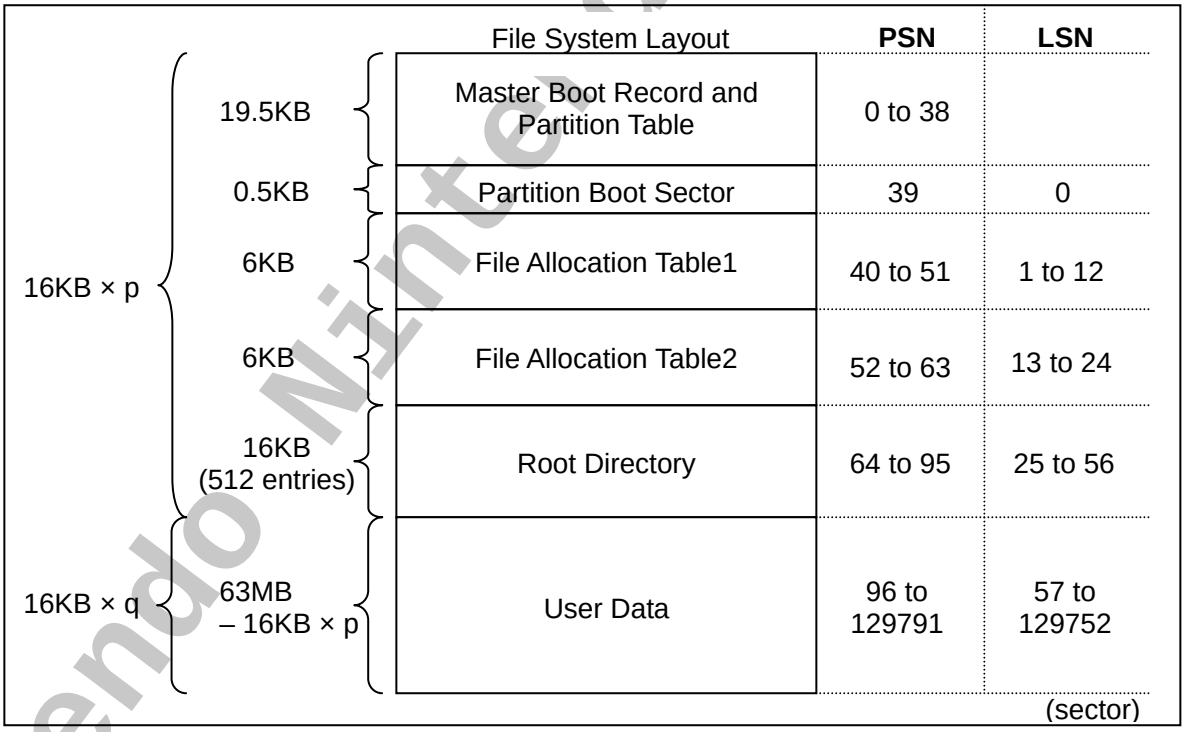
C.1.1 File System Layout

Reading/writing of the User Area should be done with the unit whose minimum size is the same as the recommended size of reading/writing at physical layer. Therefore, Cluster Size should be determined considering the recommended size. And the starting sector of the User Area should be to the boundary of the recommended size.

The structure of the file system should be implemented as follows.

1. The combined size of Master Boot Record, Partition Table, Partition Boot Sector, File Allocation Table and Root Directory is a multiple size of the Boundary Unit. Boundary Unit is the logical value determined taking the recommended size at physical layer into account. The concrete value is described in the later section.
2. The number of the sectors before Partition Boot Sector adjusts the above size.
3. Master Boot Record and Partition Boot Sector belong to different Boundary Unit.
4. The first sector of the Master Boot Record and the first sector of the User Data are always placed on the boundary of Boundary Unit.

The following is an example of a partition when the size of Boundary Unit is 16KB.



p,q : a natural number, PSN : Physical Sector Number, LSN : Logical Sector Number

Figure A-1 : Example of File System Layout

C.1.2 CHS Recommendation

The following table shows the recommendation for CHS parameter. In this table, Card Capacity means the total size of Data Area and Protected Area.

Card Capacity	Number of Heads	Sectors per Track
~2MB	2	16
~16MB	2	32
~32MB	4	32
~128MB	8	32
~256MB	16	32
~504MB	16	63
~1008MB	32	63
~2016MB	64	63
~2048MB	128	63

Table A-1 : CHS Recommendation

Starting Head, Starting Sector, Starting Cylinder, Ending Head, Ending Sector, and Ending Cylinder shall be calculated with the above CHS parameter, Total Sector, and Relative Sector as described below.

$$\text{Starting Head} = ip\{\frac{rem(\text{Relative Sector}, \text{Number of Heads} \times \text{Sectors per Track})}{\text{Sectors per Track}}\}$$

$$\text{Starting Sector} = rem(\text{Relative Sector}, \text{Sectors per Track}) + 1$$

$$\text{Starting Cylinder} = ip(\frac{\text{Relative Sector}}{\text{Number of Heads} \times \text{Sectors per Track}})$$

$$\text{Ending Head} = ip\{\frac{rem(\text{Relative Sector} + \text{Total Sector} - 1, \text{Number of Heads} \times \text{Sectors per Track})}{\text{Sectors per Track}}\}$$

$$\text{Ending Sector} = rem(\text{Relative Sector} + \text{Total Sector} - 1, \text{Sectors per Track}) + 1$$

$$\text{Ending Cylinder} = ip(\frac{\text{Relative Sector} + \text{Total Sector} - 1}{\text{Number of Heads} \times \text{Sectors per Track}})$$

The following is an example of a Partition Table when the size of the Data Area 63MB.

BP	Length (bytes)	Field Name	Contents
0	1	Boot Indicator	00h
1	1	Starting Head	1
2	2	Starting Sector / Starting Cylinder	8 / 0
4	1	System ID	06h
5	1	Ending Head	7
6	2	Ending Sector / Ending Cylinder	32 / 506
8	4	Relative Sector	39
12	4	Total Sector	129753

Table A-2 : Example of Partition Table

C.1.3 Sectors per Cluster and Boundary Unit Recommendation for Data Area

The following table shows the recommendation for Sectors per Cluster and Boundary Unit of Data Area. Number of sectors before the starting sector of User Data is multiple size of Boundary Unit. In this table, Card Capacity means the total size of Data Area and Protected Area.

Card Capacity	Sectors per Cluster (sectors)	Boundary Unit (sectors)
~8MB	16	16
~64MB	32	32
~256MB	32	64
~1024MB	32	128
~2048MB	64	128

Table A-3 : Sectors per Cluster and Boundary Unit Recommendation (Data Area)

Maximum Data Area size and an example of format parameters are shown in the following table. The format parameters in this table are examples. They should be calculated with some steps described in the following section.

Card Capacity	Sectors per Cluster (sectors)	Max Data Area size (sectors)	An example of format parameters (These values vary depending on Data Area size)				
			Clusters	FAT Sec (sectors)	Hidden (sectors)	FAT bits	User Data Offset (sectors)
~4MB	16	8032	498	2	27	12	64
~8MB	16	16224	1010	3	25	12	64
~16MB	32	32448	1011	3	57	12	96
~32MB	32	64896	2025	6	51	12	96
~64MB	32	129792	4053	12	39	12	96
~128MB	32	259584	8106	32	95	16	192
~256MB	32	519168	16216	64	95	16	256
~512MB	32	1038336	32432	127	225	16	512
~1024MB	32	2076672	64872	254	227	16	768
~2048MB	64	4153344	64884	254	227	16	768

Table A-4 : Maximum Data Area size and format parameters

However,

Card Capacity … SD Memory Card Capacity.
Sectors per Cluster … Number of sectors per cluster. This parameter is defined from Card Capacity.
Max Data Area size … Maximum number of sectors for Data Area.
Clusters … Number of clusters in User Data. This parameter varies with the Data Area size.
FAT Sec … Number of sectors per FAT. This parameter varies with the Data Area size.
Hidden … Number of sectors existing before Partition Boot Sector. This parameter varies with the Data Area size.
FAT bits … If the area is formatted with FAT12, FAT bits is 12. And if the area is formatted with FAT16, FAT bits is 16. This parameter varies with the Data Area size.
User Data Offset … Number of sectors existing before the starting sector of User Data.

Sectors per Cluster is defined from Card Capacity. Clusters, FAT Sec, Hidden, FAT bits, and User Data Offset vary with the Data Area size (Use the parameters in Table A-3 for calculation).

C.1.4 Format Parameter Computations

In this section, format parameter computations are proposed. Data Area should be formatted by the following steps.

1. Sectors per Cluster (SC) is determined from the area size.

2. Number of Root-directory Entries (RDE) is 512.

3. Sector Size (SS) is 512.

4. Reserved Sector Count (RSC) is 1.

5. Total Sectors (TS) is the number of all sectors of the area.

6. FAT bits (12[FAT12], 16[FAT16]) is determined by SC and TS.

7. Sectors per FAT (SF) is computed as following:

$$SF = ceil\{ \frac{TS/SC \times FAT\ bits}{SS \times 8} \}$$

8. Number of sectors in the system area (SSA) is computed as following:

$$SSA = RSC + 2 \times SF + ceil(\frac{32 \times RDE}{SS})$$

9. Number of Sectors in Master Boot Record (NOM) is computed as following:

$$NOM + SSA = BU \times n$$

Here, n means the minimum natural number satisfying above expression. And BU means the Boundary Unit.

10. If NOM isn't equal to BU, NOM is added BU.

11. Maximum Cluster Number (MAX) is computed as following:

$$MAX = ip(\frac{TS - NOM - SSA}{SC}) + 1$$

12. Sectors per FAT (SF') is recalculated as following:

$$SF' = ceil\{ \frac{[2 + (MAX - 1)] \times FAT\ bits}{SS \times 8} \}$$

In this formula, 'MAX-1' means the number of clusters. And '2+(MAX-1)' means the number of FAT entries including two signature entries.

13. If SF' is greater than SF, NOM is added BU. And recalculate from step 11.

14. If SF' isn't equal to SF, SF' is used as SF. And recalculate from step 8.

15. If SF' is equal to SF, parameter computing is complete.

Example of a SD Memory Card including 63MB Data Area:
- TS = 129792 Sectors
- SC = 32 Sectors
- RDE = 512 Entries
- SS = 512 B
- RSC = 1 Sectors
- FAT bits = 12 [FAT12]
- SF = 12 Sectors
- SSA = 57 Sectors
- NOM = 39 Sectors
- MAX = 4054

Example of Extended FDC Descriptor for the above example:

BP	Length (bytes)	Field Name	Contents
0	3	Jump Command	EBh, 00h, 90h
3	8	Creating System Identifier	"SYSTEMID"
11	2	Sector Size	512
13	1	Sectors per Cluster	32
14	2	Reserved Sector Count	1
16	1	Number of FATs	2
17	2	Number of Root-directory Entries	512
19	2	Total Sectors	0
21	1	Medium Identifier	F8h
22	2	Sectors per FAT	12
24	2	Sectors per Track	32
26	2	Number of Sides	8
28	4	Number of Hidden Sectors	39
32	4	Total Sectors	129753
36	1	Physical Disk Number	80h
37	1	Reserved	00h
38	1	Extended Boot Record Signature	29h
39	4	Volume ID Number	01234567h
43	11	Volume Label	"VOLUME1 "
54	8	File System Type	"FAT12 "
62	448	(Reserved for system use)	Not Restricted
510	2	Signature Word	55h, AAh

Table A-5 : Extended FDC Descriptor

C.2 Appendix for FAT32 File System

C.2.1 File System Layout

NOTE: Recommendation of FAT32 described in this section includes portions of Microsoft FAT Specification, the copyright of which is owned by Microsoft but licensed to SD Card Association.

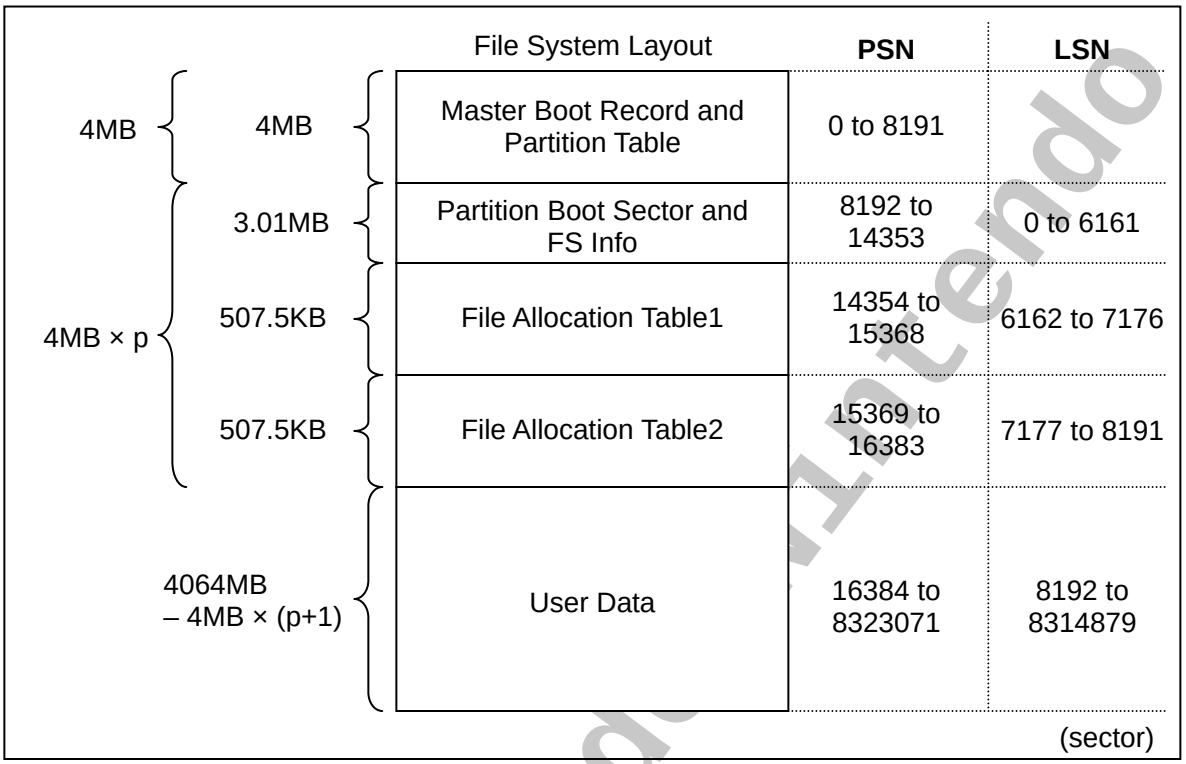
FAT32 file system for SD Memory Card has the similar recommendation with FAT12 / FAT16 file system described in the previous section. The differences with FAT12 / FAT16 recommendation are as follows.

- The file system layout is modified in order to comply with FAT32 structure.
- The adjustment area is changed from the reserved sectors existing before Partition Boot Sector to the reserved sectors existing before FAT.

The structure of the file system should be implemented as follows.

1. The number of the sectors before Partition Boot Sector is the same as the Boundary Unit size. Boundary Unit is the logical value determined taking the recommended size at physical layer into account. The concrete value is described in the later section.
2. The number of the sectors included in the area between Partition Boot Sector and the end sector of FAT2 is a multiple size of the Boundary Unit. Here, this size shall be set to minimum complying with the format parameter computations described in the later section.
3. The first sector of the Master Boot Record, the first Partition Boot Sector, and the first sector of the User Data are always placed on the boundary of Boundary Unit.

The following is an example of a partition when the size of Boundary Unit is 4MB.



p : a natural number, PSN : Physical Sector Number, LSN : Logical Sector Number

Figure A-2 : Example of File System Layout

C.2.2 CHS Recommendation

The recommendation for CHS parameters of over 2GB SD Memory Card is as follows. In this table, Card Capacity means the total size of Data Area and Protected Area.

Card Capacity	Number of Heads	Sectors per Track
Over 2048 ~ 4032MB	128	63
~32768MB	255	63

Table A-6 : CHS Recommendation

Starting Head, Starting Sector, Starting Cylinder, Ending Head, Ending Sector, and Ending Cylinder shall be calculated by the same computations for the FAT12 / FAT16 described in C.1.2. However, these parameters have the upper limit and they can represent up to 8032.5MB (8,422,686,720Bytes). Therefore, these parameters shall set to maximum values in case that the corresponding location exceeds 8032.5MB.

C.2.3 Sectors per Cluster and Boundary Unit Recommendation for Data Area

The recommendation for Sectors per Cluster and Boundary Unit of Data Area of over 2GB SD Memory Card is as follows. In this table, Card Capacity means the total size of Data Area and Protected Area.

Card Capacity	Sectors per Cluster (sectors)	Boundary Unit (sectors)
Over 2048 ~ 32768MB	64	8192

Table A-7 : Sectors per Cluster and Boundary Unit Recommendation (Data Area)

Maximum Data Area size and an example of format parameters are shown in following table. The format parameters in this table are examples. They should be calculated with some steps described in the following section.

Card Capacity	Sectors per Cluster (sectors)	Max Data Area size (sectors)	An example of format parameters (These values vary depending on Data Area size)					
			Clusters	FAT Sec (sectors)	Hidden (sectors)	Reserved Sector Count (sectors)	FAT bits	User Data Offset (sectors)
Over 2048MB ~ 4096MB	64	8323072	129792	1015	8192	6162	32	16384
~8192MB	64	16678912	260352	2035	8192	4122	32	16384
~16384MB	64	33423360	521984	4079	8192	34	32	16384
~32768MB	64	66945024	1045632	8170	8192	44	32	24576

Table A-8 : Maximum Data Area size and format parameters

However,

Card Capacity … SD Memory Card Capacity.
Sectors per Cluster … Number of sectors per cluster. This parameter is defined from Card Capacity.
Max Data Area size … Maximum number of sectors for Data Area.
Clusters … Number of clusters in User Data. This parameter varies with the Data Area size.
FAT Sec … Number of sectors per FAT. This parameter varies with the Data Area size.
Hidden … Number of sectors existing before Partition Boot Sector.
Reserved Sector Count … Number of reserved sector count. This parameter varies with the Data Area size.
FAT bits … FAT bits is 32, because the area shall be formatted with FAT32.
User Data Offset … Number of sectors existing before the starting sector of User Data.

Sectors per Cluster is defined from Card Capacity. Clusters, FAT Sec, Hidden, Reserved Sector Count, FAT bits, and User Data Offset vary with the Data Area size (Use the parameters in Table A-7 for calculation).

C.2.4 Format Parameter Computations

In this section, format parameter computations for over 2GB SD Memory Card are proposed. The basic policy is the same as the FAT12 / FAT16 computations described in the previous section. However, it is slightly modified in order to comply with FAT32 structure.

1. Sectors per Cluster (SC) is 64.

2. Sector Size (SS) is 512.

3. Total Sectors (TS) is the number of all sectors of the area.

4. FAT bits is 32 [FAT32].

5. Sectors per FAT (SF) is computed as following:

$$SF = ceil\{ \frac{TS/SC \times FAT\ bits}{SS \times 8} \}$$

6. Number of Sectors in Master Boot Record (NOM) is computed as following:

$$NOM = BU$$

Here, BU means the Boundary Unit.

7. Reserved Sector Count (RSC) is computed as following:

$$RSC = BU \times n - 2 \times SF$$

Here, n means the minimum natural number satisfying above expression.

8. If RSC is smaller than 9, RSC is added BU.

9. Number of sectors in the system area (SSA) is computed as following:

$$SSA = RSC + 2 \times SF$$

10. Maximum Cluster Number (MAX) is computed as following:

$$MAX = ip(\frac{TS - NOM - SSA}{SC}) + 1$$

11. Sectors per FAT (SF') is recalculated as following:

$$SF' = ceil\{ \frac{[2 + (MAX - 1)] \times FAT\ bits}{SS \times 8} \}$$

In this formula, 'MAX-1' means the number of clusters. And '2+(MAX-1)' means the number of FAT entries including two signature entries.

12. If SF' is greater than SF, SSA and RSC are added BU. And recalculate from step 10.

13. If SF' isn't equal to SF, 'SF-1' is used as new SF. And recalculate from step 7.

14. If SF' is equal to SF, parameter computing is complete.

Example of a SD Memory Card including 4066MB Data Area:
- TS = 8323072 Sectors
- SC = 64 Sectors
- SS = 512 B
- RSC = 6162 Sectors
- FAT bits = 32 [FAT32]
- SF = 1015 Sectors
- SSA = 8192 Sectors
- NOM = 8192 Sectors
- MAX = 129793

Example of Partition Boot Sector and FS Info Sector for the above example:

BP	Length (bytes)	Field Name	Contents
0	3	Jump Command	EBh, 00h, 90h
3	8	Creating System Identifier	"SYSTEMID"
11	2	Sector Size	512
13	1	Sectors per Cluster	64
14	2	Reserved Sector Count	6162
16	1	Number of FATs	2
17	2	Number of Root-directory Entries	0
19	2	Total Sectors	0
21	1	Medium Identifier	F8h
22	2	Sectors per FAT	0
24	2	Sectors per Track	63
26	2	Number of Sides	255
28	4	Number of Hidden Sectors	8192
32	4	Total Sectors	8314880
36	4	Sectors per FAT for FAT32	1015
40	2	Extension Flag	00h
42	2	FS Version	0000h
44	4	Root Cluster	2
48	2	FS Info	1
50	2	Backup Boot Sector	6
52	12	Reserved	All 00h
64	1	Physical Disk Number	80h
65	1	Reserved	00h
66	1	Extended Boot Record Signature	29h
67	4	Volume ID Number	01234567h
71	11	Volume Label	"VOLUME1 "
82	8	File System Type	"FAT32 "
90	420	(Reserved for system use)	Not Restricted
510	2	Signature Word	55h, AAh

Table A-9 : Partition Boot Sector

BP	Length (bytes)	Field Name	Contents
0	4	Lead Signature	52h, 52h, 64h, 41h
4	480	Reserved	All 00h
484	4	Structure Signature	72h, 72h, 41h, 61h
488	4	Free Cluster Count	FFFFFFFFh
492	4	Next Free Cluster	FFFFFFFFh
496	12	Reserved	All 00h
508	4	Trail Signature	00h, 00h, 55h, AAh

Table A-10 : FS Info Sector

C.3 Long File Name Implementation (optional)

C.3.1 LFN in SD Memory Card File System

For all FAT12/FAT16/FAT32 file system of SD Memory Card, the file name format shall support 8.3 format as mandatory. Moreover, hosts can also support Long File Name (LFN) as optional. This section describes Long File Name format and usages.

NOTE: Long File Name described in this section is OPTIONAL. The Long File Name specification described in this section includes portions of Microsoft FAT Specification, the copyright of which is owned by Microsoft but licensed to SD Card Association.

C.3.2 FAT Long Directory Entries

The Name and Name Extension field in the directory entry only allows for a 11 character file / sub-directory name comprised of a main part (maximum length of 8 characters) and an extension (maximum length of 3 characters). The contents of this field are also known as the “short name” and the corresponding directory entry is also known as the short name directory entry. Applications and users typically prefer creating longer (more descriptive) names for files/sub-directories. This section describes how such long file names can be stored on media.

A long file name for a target file or sub-directory is stored in a set (one or more) of additional directory entries associated with the short name directory entry describing the target file or sub-directory. This set of additional directory entries (also known as the long name directory entries) shall immediately precede the corresponding short name directory entry and is, therefore, physically contiguous with the short name directory entry.

NOTE: The sequence of long name directory entries is stored in reverse order (last entry in the set is stored first, followed by entry n-1, followed by entry n-2, and so on, until entry 1).

A long name directory entry structure is described in the table below:

BP	Length (bytes)	Field Name	Description
0	1	LDIR_Ord	The order of this entry in the sequence of long name directory entries (each containing components of the long file name) associated with the corresponding short name directory entry. The contents of this field shall be masked with 40h (LAST_LONG_ENTRY) for the last long directory name entry in the set. Therefore, each sequence of long name directory entries begins with the contents of this field masked with LAST_LONG_ENTRY.
1	10	LDIR_Name1	Contains characters 1 through 5 constituting a portion of the long name.
11	1	LDIR_Attr	Attributes – shall be set to ATTR_LONG_NAME defined as below: ATTR_LONG_NAME = (ATTR_READ_ONLY | ATTR_HIDDEN | ATTR_SYSTEM | ATTR_VOLUME_ID) NOTE: A mask to determine whether a directory entry is part of the set of a long name directory entries is defined below: #define ATTR_LONG_NAME_MASK = (ATTR_READ_ONLY | ATTR_HIDDEN | ATTR_SYSTEM | ATTR_VOLUME_ID | ATTR_DIRECTORY | ATTR_ARCHIVE)
12	1	LDIR_Type	Shall be set to 0.
13	1	LDIR_Chksum	Checksum of name in the associated short name directory entry at the end of the long name directory entry set.
14	12	LDIR_Name2	Contains characters 6 through 11 constituting a portion of the long name.
26	2	LDIR_FstClusLO	Shall be set to 0.
28	4	LDIR_Name3	Contains characters 12 and 13 constituting a portion of the long name.

Table A-11 : FAT Long Directory Entry Structure

The below illustrates how long name directory entries are stored:

Entry	Ordinal
Nth long name directory entry	LAST_LONG_ENTRY (40h) | N
… Additional long name directory entries	…
1st long name directory entry	1
Short Entry Associated with Preceding Long Entries	(not applicable)

Table A-12 : Sequence of Long Directory Entries

C.3.3 Ordinal Number Generation

The below rules shall be followed in storing ordinal numbers for each long name directory entry in a set of such entries (associated with a short name directory entry):

The first member of a set has an LDIR_Ord value of 1.
The LDIR_Ord value for each subsequent entry shall contain a monotonically increasing value.
The Nth (last) member of the set shall contain a value of (N | LAST_LONG_ENTRY)

If any member of the set of long name directory entries does not follow the rules above, the set is considered corrupt.

C.3.4 Checksum Generation

An 8-bit checksum is computed on the name contained in the short name directory entry at the time the short and long name directory entries are created. All 11 characters of the name in the short name entry are used in the checksum calculation. The check sum is placed in every long name directory entry in the LDIR_Chksum field. If any of the check sums in the set of long name entries associated with the appropriate short name directory entry, do not agree with the computed checksum of the name contained in the short name directory entry, the set of long name directory entries is considered corrupt.

The below algorithm describes the logic used to generate the check sum value:

```
//------------------------------------------------------------------------
//      ChkSum()
//      Returns an unsigned byte checksum computed on an unsigned byte
//      array.  The array must be 11 bytes long and is assumed to contain
//      a name stored in the format of a MS-DOS directory entry.
//      Passed:  pFcbName    Pointer to an unsigned byte array assumed to be
//                          11 bytes long.
//      Returns: Sum         An 8-bit unsigned checksum of the array pointed
//                           to by pFcbName.
//------------------------------------------------------------------------
unsigned char ChkSum (unsigned char *pFcbName)
{
        short FcbNameLen;
        unsigned char Sum;

        Sum = 0;
        for (FcbNameLen=11; FcbNameLen!=0; FcbNameLen--) {
                // NOTE: The operation is an unsigned char rotate right
                Sum = ((Sum & 1) ? 0x80 : 0) + (Sum >> 1) + *pFcbName++;
        }
        return (Sum);
}
```

C.3.5 Example illustrating persistence of a long name

The following example is provided to illustrate how a long name is stored across several long name

directory entries. Names are also NULL terminated and padded with FFFFh characters in order to detect corruption of long name fields. A name that fits exactly in a set of long name directory entries (i.e. is an integer multiple of 13) is not NULL terminated and not padded with FFFFh.

Name: "The quick brown.fox"

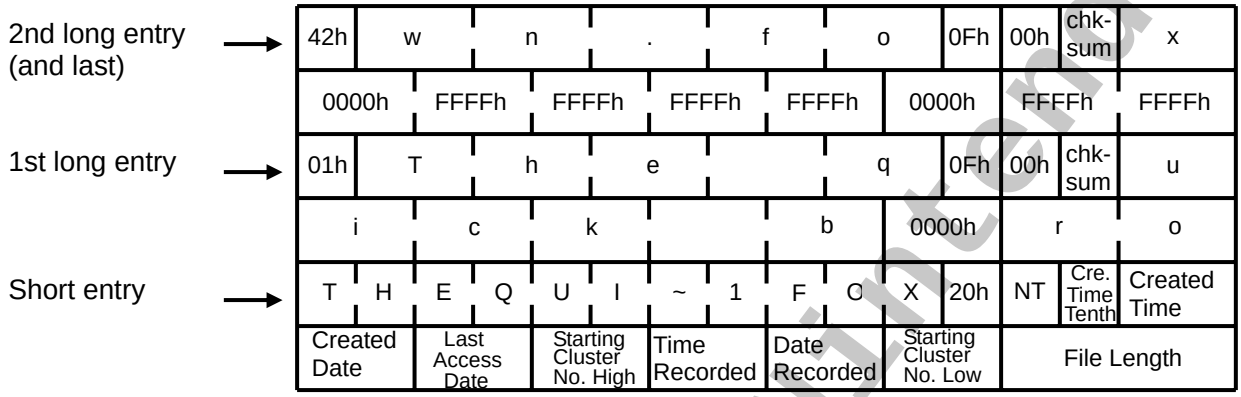


Figure A-3 : Example of Long Directory Entries

C.3.5.1 Name Limits and Character Set for Long File Names

Long file names are limited to 255 characters, not including the trailing NULL. The characters may be any combination of those defined for short file names with the addition of the period (“.”) character used multiple times within the long name. A space is also a valid character in a long file name as it always has been for a short file name. The following six special characters are allowed in a long file name (they are not legal in a short file name):

+ , ; = [ ]

Embedded spaces within a long file name are allowed. Leading and trailing spaces in a long file name are ignored.

Leading and embedded periods are allowed in a name and are stored in the long file name. Trailing periods are ignored.

Long file names are stored in long directory entries in UNICODE. UNICODE characters are 16-bit characters. It is not be possible to store UNICODE in short name directory entries since the names stored there are 8-bit characters or DBCS characters.

Long file names passed to the file system are not converted to upper case and their original case value is preserved. UNICODE solves the case mapping problem prevalent in some OEM code pages by always providing a translation for lower case characters to a single, unique upper case character

C.3.6 Rules governing name creation and matching

The union of all long and short file names is defined as the namespace of objects contained in the volume.

For such a namespace, the following rules shall be observed:

- Any name within a specific directory, whether it is a short name or a long name, can occur only once (difference in case is ignored and such names shall be considered as conflicting)

- When a character on the media (whether stored in the OEM character set or in UNICODE) cannot be translated into the appropriate character in the OEM or ANSI code page, it is always translated to the "_" (underscore) character – the character is not modified on the media. The “_” character is the same in all OEM code pages and ANSI.

C.4 Differences from Microsoft FAT Specification

FAT32 file system described in this specification complies with Microsoft FAT Specification. However, some parameters are limited for optimization. The main differences are as follows.

Item / Field		FAT32 of Microsoft FAT Specification	FAT32 of SD Specification
Support Capacity		Over 32.5MB	Over 2GB and up to 32GB
Master Boot Record and Partition Table		Not specified	Specified in the same way as Ver1.01
Boot Sector	Sector Size	512, 1024, 2048 or 4096 bytes	512 bytes
	Cluster Size	1, 2, 4, 8, 16, 32, 64, or 128 sectors	1, 2, 4, 8, 16, 32, or 64 sectors ('64' is strongly recommended for SD Memory Card which has capacity more than 2GB and up to 32GB.)
	Number of FATs	1 or 2 (Recommendation is 2)	2
	Medium Identifier	The legal values for this field are F0h, F9h, FAh, FBh, FCh, FEh, and FFh. F8h is the standard value for "fixed" (non-removable) media. For removable media, F0h is frequently used.	F8h
	Sectors per Track	Not specified	The recommended values are specified in Appendix.
	Number of Sides	Not specified	The recommended values are specified in Appendix.
	FS Info	Usually 1.	1
	Backup Boot Sector	0 or 6	6
	Physical Disc Number	00h or 80h	80h
	Extended Boot Signature	Set value to 29h if either of the following two fields are non-zero.	29h
FAT Entry Value		FFFFFF8h to FFFFFFEh is reserved and should not be used. It may be interpreted as an allocated cluster and the final cluster in the file.	FFFFFF8h to FFFFFFEh show the corresponding cluster is already allocated, and it is the final cluster of the file.
File System Layout		The method of User Data area offset alignment is not specified.	The method of User Data area offset alignment is specified in Appendix.

Table A-13 : Comparison Table for FAT32 file systems