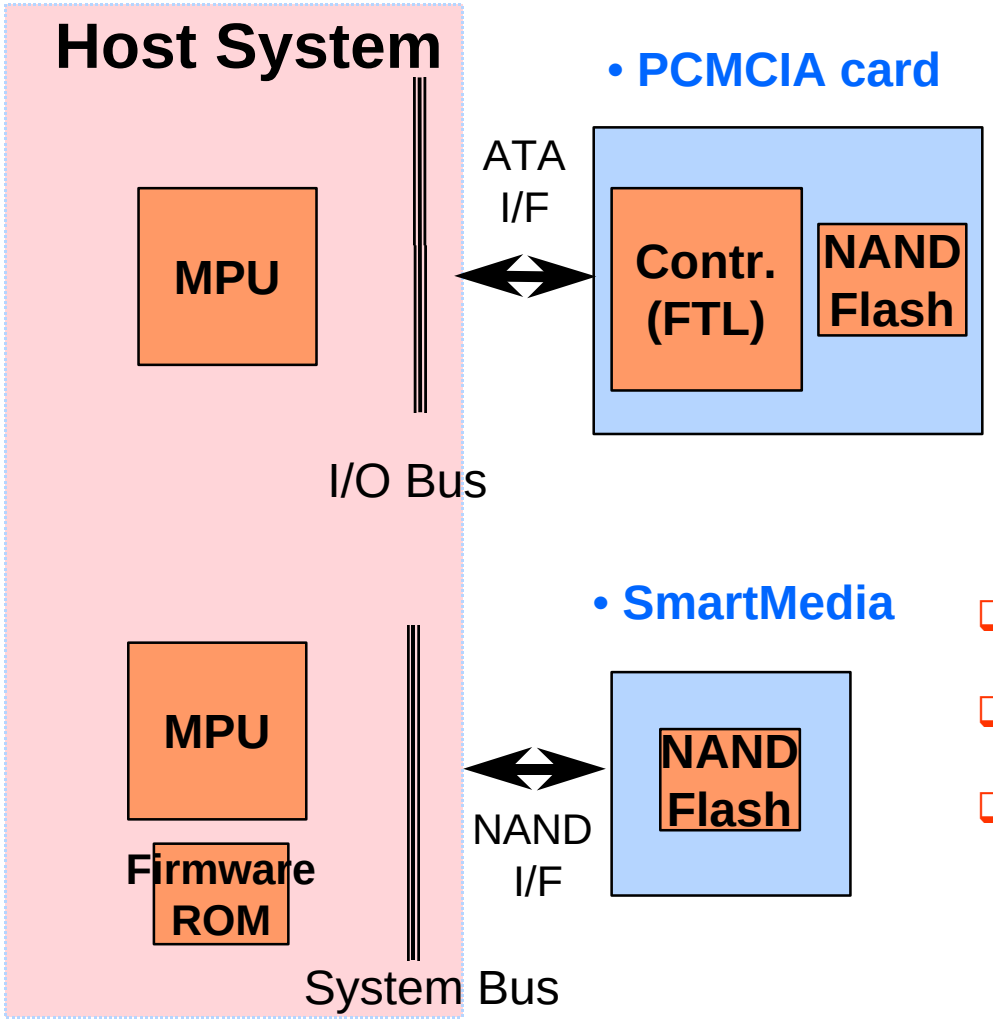
SmartMedia™ Format Introduction
(Software Considerations)

Memory Product &
Technology Division

1999. 07.13

Why Standard File System for SmartMedia?



- High overhead cost for card assembly
- Flash file management by controller-proprietary firmware

- No overhead cost for assembly
- Direct access by host system bus
- Flash file management by host firmware needs to be standardized for compatibility

SmartMedia Specification List

Already standardized up to 128MB SmartMedia !

1. The essential specification for developer.

- SmartMedia Logical Format Specification(Ver 1.11, 99.4)

- SmartMedia Physical Format Specification(Ver 1.20, 99.4)

**** Non-members are not given access to specifications. Anyone who want to get this specifications should be a member of the SSFDC Forum (URL: www.ssfdc.or.jp)**

2. Other useful Specificaton.

- SmartMedia Physical Specification(Ver 1.11, 99.4)

- SmartMedia Application Specification(Ver 1.0,97.9)

- SmartMedia Electronics Specification(Ver 1.20, 98.12)

- SmartMedia LogoMark Interface Specification(Ver 1.00, 97.11)

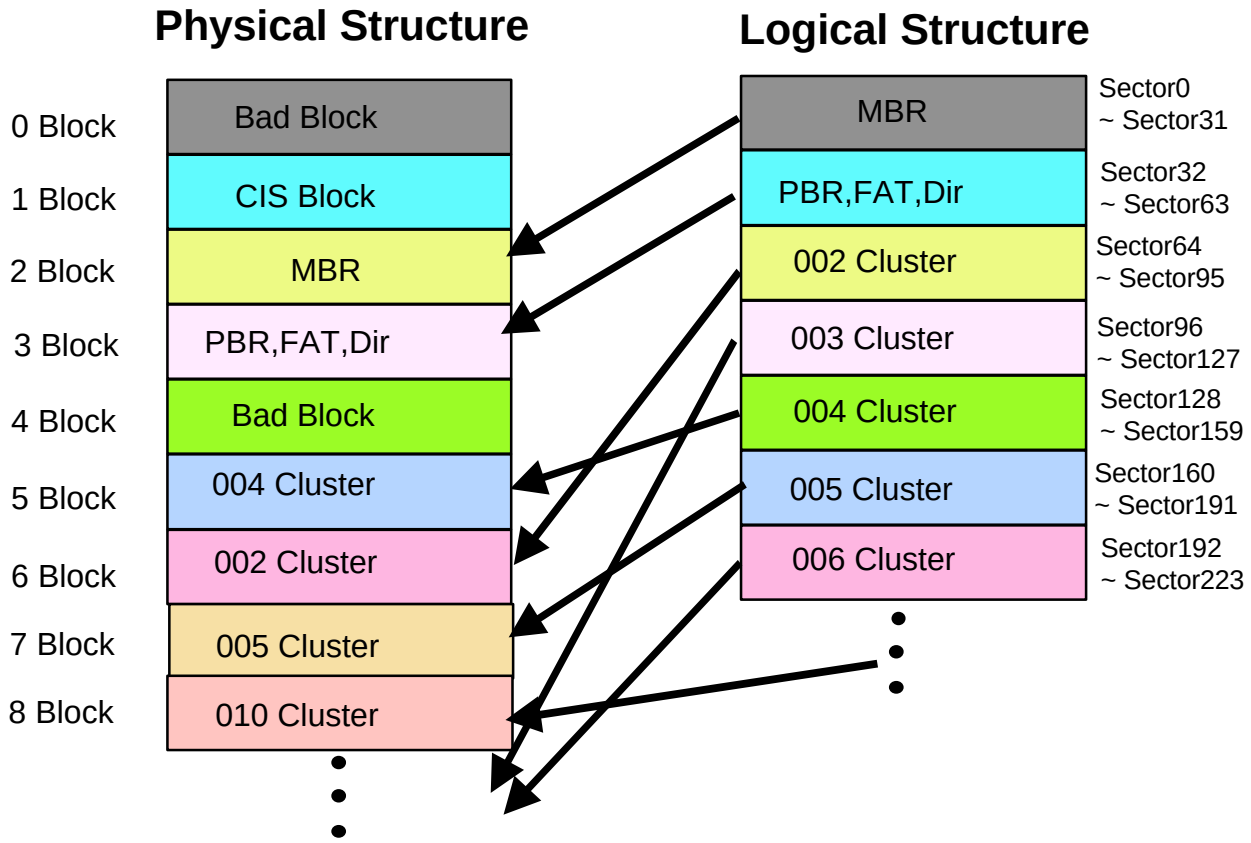
- SmartMedia Voltage,Volume Guideline Specification(Ver 1.00, 98.12)

- SmartMedia Interface Guideline Specification(Ver 1.00,98.12)

- SmartMedia Compatibility Guideline(Ver 1.00,99.4)

How Logical/Physical Structures are interrelated

Irrelevant to physical address, logical structures are pre-defined and gives flexibility in the memory usage



Physical / Logical Format

Application

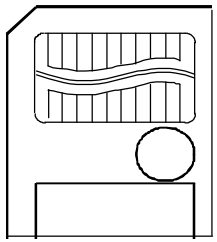
File

FAT management

Logical Sector

Device driver

Physical Address



SmartMedia™

Physical Format Specification.

SmartMedia Physical Structure
- Zone boundary Concept
- Card Information Structure /Identify Drive Information Block
- Data Status Flag
- Block Status Flag
- ECC Area

Logical Format Specification

SmartMedia Logical Structure
- DOS/FAT File System Management
- Logical Sector Concept
- Master Boot Sector
- Partition Boot Sector (ClusterSize,SectorSize Total Cluster,etc)
- FAT1,2
- Root Directory
- Using LBA Mode

What is Physical Format?

2MByte

NAND Structure
264Bytex16Pages x 512Blocks

Block No.000 (CIS)
Block No.001
Block No.002
Block No.003
⋮
Block No.510
Block No.511

Block Structure
1Block=16Page

Page No.00
Page No.01
Page No.02
Page No.03
⋮
Page No.14
Page No.15

Sector Data Structure
1Sector=2Pages

	Even Page	Odd Page
0-255	Data Area-1	Data Area-2
256	Reserved Area	ECC Area-2
257		
258		
259		Block Address-2
260	Data Status Flag	
261	Block Status Flag	ECC Area-1
262	Block Address-1	
263		

4MByte
8MByte
Higher Densities

NAND Structure
528Bytex16,32Pages x 512Blocks

Block No.000 (CIS)
Block No.001
Block No.002
Block No.003
⋮
Block No.510
Block No.511
⋮
Block No.1022
Block No.1023

Block Structure
1Block=16,32Page

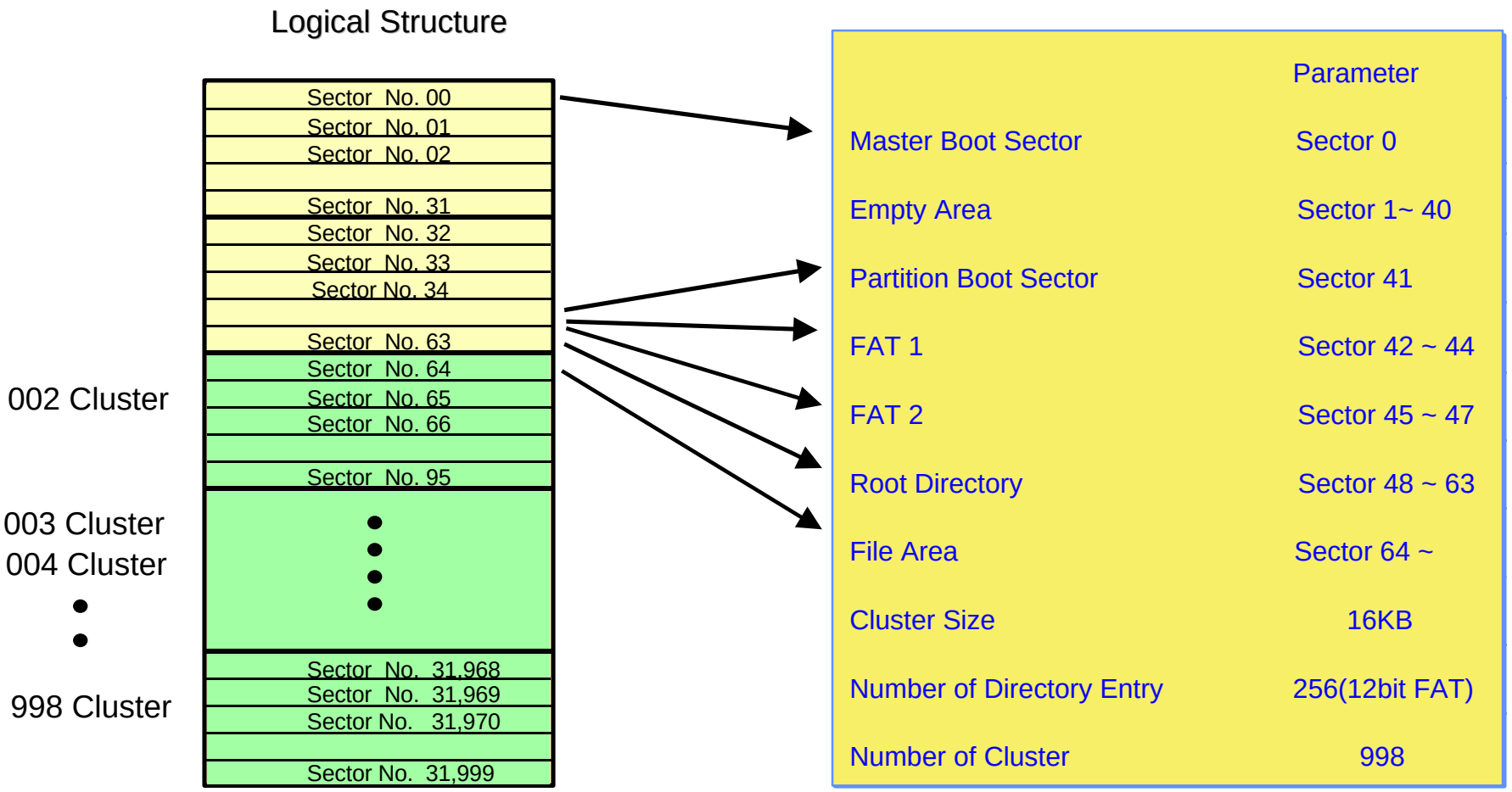
Page No.00
Page No.01
Page No.02
Page No.03
⋮
Page No.14
Page No.15

Sector Data Structure
1Sector=1Page

0-255	Data Area-1
256-511	Data Area-2
512-515	Reserved Area
516	Data Status Flag
517	Block Status Flag
518-519	Block Address-1
520-522	ECC Area-2
523-524	Block Address-2
525-527	ECC Area-1

What is Logical Format?

In case of 16MB SmartMedia



CIS (Card Information System) Area (1 and 2) I

Addr	Data	Contents
00	01h	Tuple ID(CIS TPL_Device)
01	03h	Link to Next Tuple
02	D9h	Device Type : I/O, Rate : 250ns
03	01h	Device Size : 2 K Byte
04	FFh	End of Device ID Tuple
05	18h	Tuple ID(CIS TPL_JEDEC_C)
06	02h	Link to Next Tuple
07	DFh	JEDEC Manufacture ID(PC Card ATA)
08	01h	JEDEC Device ID(VPP not required)
09	20h	Tuple ID(CIS TPL_MANF ID)
0A	04h	Link to Next Tuple
0B	00h	Manufacture Code
0C	00h	Manufacture Code
0D	00h	Manufacture Info.
0E	00h	Manufacture Info.
0F	21h	Tuple ID(CIS TPL_FUNC ID)
10	02h	Link to Next Tuple
11	04h	PL FID_FUNCTION
12	01h	TPL_FID_SYS INIT
13	22h	Tuple ID(CIS TPL_FUNCE)
14	02h	Link to Next Tuple
15	01h	Disk Device Interface Tuple
16	01h	PC Card ATA Interface
17	22h	Tuple ID(CIS TPL_FUNCE)
18	03h	Link to Next Tuple
19	02h	PC Card ATA Extension Tuple
1A	04h	ATA Function Byte1
1B	07h	ATA Function Byte2

Addr	Data	Contents
1C	1Ah	Tuple ID(CIS TPL_CONFIG)
1D	05h	Link to Next Tuple
1E	01h	Field Size Byte
1F	03h	Last Entry in the Card Configuration Table
20	00h	CCR Base Address(Low-order Byte)
21	02h	CCR Base Address(High-order Byte)
22	0Fh	CCR Present Mask
23	1Bh	Tuple ID(CIS TPL_CFTABLE_ENTRY)
24	08h	Link to Next Tuple
25	C0h	Configuration Table Index Byte
26	C0h	Interface Description Field
27	A1h	Feature Selection Byte
28	01h	Power Parameter Selection Byte
29	55h	Power Voltage(5V)
2A	08h	Memory Space(Low-order byte)
2B	00h	Memory Space(High-order byte)
2C	20h	Miscellaneous (ex: CCSR power down)
2D	1Bh	Tuple ID(CIS TPL_CFTABLE_ENTRY)
2E	0Ah	Link to Next Tuple
2F	C1h	Configuration Table Index Byte
30	41h	Interface Description Field
31	99h	Feature Selection Byte
32	01h	Power Parameter Selection Byte
33	55h	Power Voltage(5V)
34	64h	I/O Space Description Byte
35	F0h	Interrupt IRQ Condition Info.
36	FFh	Interrupt IRQs 0 to 7
37	FFh	Interrupt IRQs 8 to 15

CIS (Card Information System) Area (1 and 2) II

Addr	Data	Contents
38	20h	Miscellaneous (ex: CCSR power down)
39	1Bh	Tuple ID [I/O Primary]
3A	0Ch	Link to Next Tuple
3B	82h	Configuration Table Index Byte
3C	41h	Interface Description Field
3D	18h	Feature Selection Byte
3E	EAh	I/O Space Description Byte
3F	61h	I/O Range Description Byte
40	F0h	I/O Address Range(01F0h-01F7h)
41	01h	I/O Address Range(01F0h-01F7h)
42	07h	8 Bytes
43	F6h	I/O Address Range(03F6h-03F7h)
44	03h	I/O Address Range(03F6h-03F7h)
45	01h	2 Bytes
46	EEh	IRQ Condition Info. ($\overline{\text{IRQ14}}$)
47	1Bh	Tuple ID[I/O secondary]
48	0Ch	Link to Next Tuple
49	83h	Configuration Table Index Byte
4A	41h	Interface Description Field
4B	18h	Feature Selection Byte
4C	EAh	I/O Space Description Byte
4D	61h	I/O Range Description Byte
4E	70h	I/O Address Range(0170h-0177h)
4F	01h	I/O Address Range(0170h-0177h)
50	07h	8 Bytes
51	76h	I/O Address Range(0376h-0377h)
52	03h	I/O Address Range(0376h-0377h)
53	01h	2 Bytes

Addr	Data	Contents
54	EEh	IRQ Condition Info. ($\overline{\text{IRQ14}}$)
55	15h	Tuple ID(CIS TPL_VERS_1)
56	14h	Link to Next Tuple
57	05h	Major Version Number[Ver.5]
58	00h	Minor Version Number[Ver.0]
59	20h	Name of Manufacture
5A	20h	Name of Manufacture
5B	20h	Name of Manufacture
5C	20h	Name of Manufacture
5D	20h	Name of Manufacture
5E	20h	Name of Manufacture
5F	20h	Name of Manufacture
60	00h	End of Manufacture Name
61	20h	Name of Product
62	20h	Name of Product
63	20h	Name of Product
64	20h	Name of Product
65	00h	End of Product Name
66	30h	Product Version “0”
67	2Eh	Product Version "."
68	30h	Product Version "0"
69	00h	End of Product Version
6A	FFh	End of Product Info. Tuple
6B	14h	CIS TPL_NO_LINK
6C	00h	Link to Next Tuple
6D	FFh	CIS TPL_END
6E-7F	00h	Null-Tuple

Logical Format Parameter

	1 MB	2 MB	4 MB	8 MB	16 MB	32 MB	64 MB	128 MB
NumCylinder	125	125	250	250	500	500	500	500
NumHead	4	4	4	4	4	8	8	16
NumSector	4	8	8	16	16	16	32	32
SumSector	2,000	4,000	8,000	16,000	32,000	64,000	128,000	256,000
SectorSize	512	512	512	512	512	512	512	512

Master Boot Record (MBR) 1

The Master Boot Record contains the following fields:

Offset	Size(Bytes)	Description
000H	446	Boot code
1BEH	16	Partition Entry
1CEH	16	Partition Entry
1DEH	16	Partition Entry
1EEH	16	Partition Entry
1FEH	2	Signature Word(0x55AA)

Master Boot Record (MBR) 2

Each of the four Partition Entries in the Master Boot Record have the following format:

Offset	Size(Bytes)	Description
00H	1	x86 Default Boot Partition (00H=Not Default, 80H=Default)
01H	1	StartHead-Zero-based(0)head number
02H	1	StartSector-Zero-based(1) sector number. Bits 6 and 7 are high bits of zero-based(0) cylinder number.
03H	1	StartCylinder
04H	1	Partition Type
		00H:Unknown or deleted if NumSectors is zero
		01H:MS-DOS 12-bit BPB/FAT < 16 MB
		04H:MS-DOS 16-bit BPB/FAT < 32 MB
		05H:Extended MS-DOS Partition
		06H:MS-DOS 16-bit BPB/FAT >= 32 MB
05H	1	EndHead-Zero-based(0)head number
06H	1	EndSector-Zero-based(1) sector number. Bits 6 and 7 are high bits of zero-based(0) cylinder number.
07H	1	EndCylinder
08H	4	StartSector(relative to beginning of Extended MS-DOS)
0CH	4	NumSectors

Partition Boot Record (PBR)

The Partition Boot Record contains the following fields

Offset	Size(Bytes)	Description
000H	3	JMP instruction to PBR boot code
003H	8	OEMName and version
00BH	25	BIOS Parameter Block (BPB)
024H	1	DriverNumber(00H=Floppy,80H=Fixed)
025H	1	Reserved, do not use.
026H	1	ExtBootSignature-29H
027H	4	VolumeID or Serial Number
02BH	11	VolumeLabel-ASCII characters. Padded with blanks if less than eleven (11) characters.
036H	8	FileSysType-ASCII Characters identifying file system type. Padded with blanks if less than eight (8) characters. One of The following values:
		Value: FAT12 — Meaning: 12-bit File Allocation Table (FAT)
		Value: FAT16 — Meaning: 16-bit File Allocation Table (FAT)
03EH	448	Boot code
1FEH	2	Signature word - 55AAH

BIOS Parameter Block (BPB)

The BIOS Parameter Block(BPB) contains the following fields:

Offset	Size(Bytes)	Description
000H	2	BytesPerSector-Number of bytes per sector
002H	1	SectorsPerCluster-Number of sectors in a cluster
003H	2	ReservedSectors
005H	1	NumFATs-Number of FAT on the media
006H	2	RootDirEntries-Number of Root Directory entries
008H	2	TotalSectors If Sector is over 65,535, this field is zero and actual number of sectors is in the HugeSectors field.
00AH	1	MedialDByte-Used to quickly identify how the media is formatted. F0H:Various types of media F8H: Hard disk, any size F9H:720 KB 3.5” or 1.2 MB 5.25” FAH:320 KB 5.25” FBH:640 KB 3.5” FCH:180 KB 5.25” FDH:360 KB 5.25” FEH:160 KB 5.25” FFH:320 KB 5.25”
00BH	2	NumFATSectors-Number of sectors in each FAT
00DH	2	SectorsPerTrack-Number of sectors on a track
00FH	2	NumHeads-Number of heads
011H	4	HiddenSectors-Number of hidden sectors
015H	4	HugeSectors-Number of sectors if Total sectors is zero.

Spare Area Information (1 MB, 2 MB)

To manage data in 256-Byte unit, pages are handled in pairs.

Spare Area Configuration (Even+Odd page.16Byte)

Byte No.	Even-numbered page	Odd-numbered page
256	User Data Area	ECC Area-2
257		
258		
259		Block Address Area-2
260	User Status Area	
261	Block Status Area	ECC Area-1
262	Block Address Area-1	
263		

Spare Area Information (4 ~128 MB)

Manage data in 512-Byte unit per page.

Spare Area Configuration (16 Byte)

Byte No.	Contents	Byte No.	Contents
512	User Data Area	520	ECC Area-2
513		521	
514		522	
515		523	Block Address Area-2
516	User Status Area	524	
517	Block Status Area	525	ECC Area-1
518	Block Address Area-1	526	
519		527	

Block Address Area Information

The data in this area indicates address information on the conversion table to be consulted for block-logical-address to physical-address conversion

Block Address Configuration

D7	D6	D5	D4	D3	D2	D1	D0	1, 2 MB SM	4, 8, 16 MB SM
0	0	0	1	0	BA9	BA8	BA7	262 bytes(even) 259 bytes(odd)	518, 523 bytes
BA6	BA5	BA4	BA3	BA2	BA1	BA0	P	263 bytes(even) 260 bytes(odd)	519, 524 bytes

BA9 ~ BA0 : Block Address(values=0 through n,where n = maximum logical block count - 1)

P : Even Parity bit

Block addresses referred to here represent addresses obtained in the form of data segments after logical addresses have been separated by individual erasure blocks.

FAT(File Allocation Table) Content

In case of 12-bit FAT

Offset	Content	Description
00h - 02h	F8h, FFh, FFh	FAT ID (3 Bytes)
03h and after	00h	

In case of 16-bit FAT

Offset	Content	Description
00h - 03h	F8h,FFh,FFh,FFh	FAT ID (4 Bytes)
04h and after	00h	

FAT Content

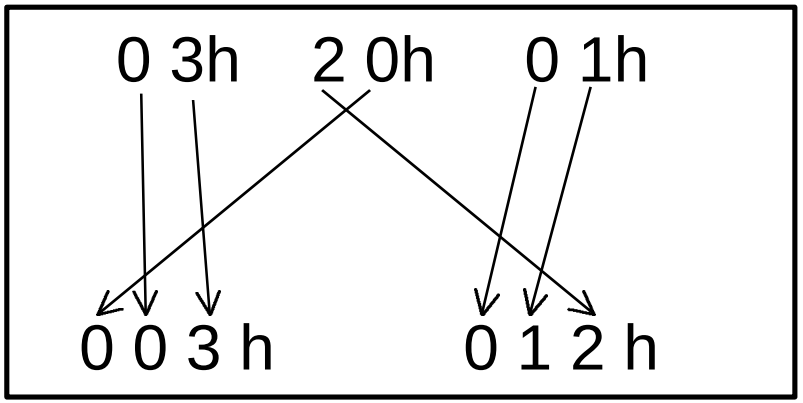
12-bit FAT	16-bit FAT	Description
000h	0000h	Unused Cluster
001h	0001h	Reserved
002h - FEFh	0002h - FFEFh	Next Cluster Number in the chain
FF0h - FF6h	FFF0h - FFF6h	Reserved
FF7h	FFF7h	Defective Cluster
FF8h - FFFh	FFF8h - FFFFh	Last Cluster in the chain

Example of FAT Operation

In case of 12-bit FAT

002h - FEFh (Next Cluster Number in the chain) => About 4000 Cluster

Example



Location	002	003	004	005	006	007	008	009	00A	00B	00C	00D	00E	00F	010	011	012	…
Value	003 h	012h	FFFh	FFFh	008h	123h	009h	FFFh	222h	543h	FFFh	E34h	093h	453h	765h	876h	006h	...

In case of Start Cluster ‘002h’ in file information 32Byte

Cluster chain ==> 002h,003h,012h,006h,008h,009h.

Directory Content

32 Byte Information

In case of SmartMedia, Initialization Values are all zero.

Byte	Content	Initialization Value
0 - 7	File Name	*00h, F6h F6h
8 - 10	Extension	F6h, F6h, F6h
11	Attribute	F6h
12 - 21	Reserved	F6h, F6h F6h
22 -23	Time	F6h, F6h
24 -25	Date	F6h, F6h
26 -27	Start Cluster Number	F6h, F6h
28 - 31	File Size	F6h, F6h, F6h, F6h

***00h : Unused Directory**

E5h : Deleted Directory

2Eh : Sub Directory

Examples

43 4F 4E 46 49 47 20 20 53 59 53 20 00 00 00 00	Config.sys
00 00 00 00 00 00 25 43 AF 20 02 00 9C 03 00 00	

Example of Copy,Del,Mkdir (16 MB)

FAT	f8 ff ff 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	ff ff ff ff ff ff 10 2 ff ff ff 10 2 aa aa 97
Dir	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	ff ff ff ff ff ff 10 4 ff ff ff 10 4 ff ff ff
File	ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff

Initial Format Data

After Copy A.TXT

(Content: ABCDEFGHIJKLMNOPQRTUVWXYZ)

FAT	f8 ff ff ***ff*** 0***f*** 0 0 0 0 0 0 0 0 0 0 0 0 0 0	ff ff ff ff ff ff 10 2 ff ff ff 10 2 aa aa 97
Dir	***41 20 20 20 20 20 20 20 54 58 54 0 ... 97 89 ba 22 2 (32 Bytes)*** 0...	ff ff ff ff ff ff 10 4 ff ff ff 10 4 ab a5 6b
File	***4142 43 44 45 ... 59 5a d a 0 0 0 ...***	ff ff ff ff ff ff 10 7 ff ff ff 10 7 56 aa 67

After md AAA

If new Sub directory is made, File context contains Files.

FAT	f8 ff ff ff ***ff ff*** 0 0 0 0 0 0 0 0 0 0 0 0 0 0	ff ff ff ff ff ff 10 2 ff ff ff 10 2 aa aa 97
Dir	41 20 20 20 20 20 20 20 54 58 54 0 ... 97 89 ba 22 2 .. (32Byte) ***41***	
	41 41 20 20 20 20 20 20 20 20 10 ... 0 97 89 ba 22 3 .. (32 Bytes) 0 0	ff ff ff ff ff ff 10 4 ff ff ff 10 4 ab a5 6b
File1	4142 43 44 45 ... 59 5a d a 0 0 0 .....***(8 KBytes,1 Cluster)***	ff ff ff ff ff ff 10 7 ff ff ff 10 7 56 aa 67
File2	***2e 20 20 20 20 20 20 20 20 20 20 10 0 .. 0 97 89 ba 22 3 0..(32 Bytes)***	
Sub	***2e 2e 20 20 20 20 20 20 20 20 20 10 0.. 0 97 89 ba 22 0 ..(32 Bytes) 0 ..***	ff ff ff ff ff ff 10 8 ff ff ff 10 8 c0 cf 3

Example of Copy,Del,Mkdir (16MB)

FAT	f8 ff ff ff ff ff ***ff 0f*** 0 0 0 0 0 0 0 0 0 0 0 0 ....
Dir	41 20 20 20 20 20 20 20 54 58 54 0 ... 97 89 ba 22 2 .. (32 Bytes) 41 41 41 20 20 20 20 20 20 20 20 10 ... 0 97 89 ba 22 3 .. (32 Bytes) ....
File1	4142 43 44 45 ... 59 5a d a 0 0 0 .....***(8 KBytes,1 Cluster)***
File2 (Sub Dir)	2e 20 20 20 20 20 20 20 20 20 20 10 0 .. 0 97 89 ba 22 3 0..(32 Bytes) 2e 2e 20 20 20 20 20 20 20 20 20 10 0.. 0 97 89 ba 22 0 ..(32 Bytes) ***42 20 20 20 20 20 20 20 54 58 54 0 ... 4d 6e 32 22 4 0 ...*** (32 Bytes) 0 0..
File3	***61 62 63 64 65 66 ... 79 7a d a 0 0 0 .....(8 KBytes,1 Cluster)***

* Spare Area is same as above page.

Copy B.TXT in AAA directory
(Content:abcdefghijklmnopqrstuvwxyz)

FAT	f8 ff ff ***0 f0*** ff ff 0f 0 0 0 0 0 0 0 0 0 0 0 0 ....
Dir	***e5*** 20 20 20 20 20 20 20 54 58 54 0 ... 97 89 ba 22 2 .. (32 Bytes) 41 41 41 20 20 20 20 20 20 20 20 10 ... 0 97 89 ba 22 3 .. (32 Bytes) ....
File1	4142 43 44 45 ... 59 5a d a 0 0 0 .....***(8 KBytes,1 Cluster)***
File2 (Sub Dir)	2e 20 20 20 20 20 20 20 20 20 20 10 0 .. 0 97 89 ba 22 3 0..(32 Bytes) 2e 2e 20 20 20 20 20 20 20 20 20 10 0.. 0 97 89 ba 22 0 ..(32 Bytes) 42 20 20 20 20 20 20 20 54 58 54 0 ... 4d 6e 32 22 4 0 ...Z(32 Bytes)
File3	61 62 63 64 65 66 ... 79 7a d a 0 0 0 .....***(8 KBytes,1 Cluster)***

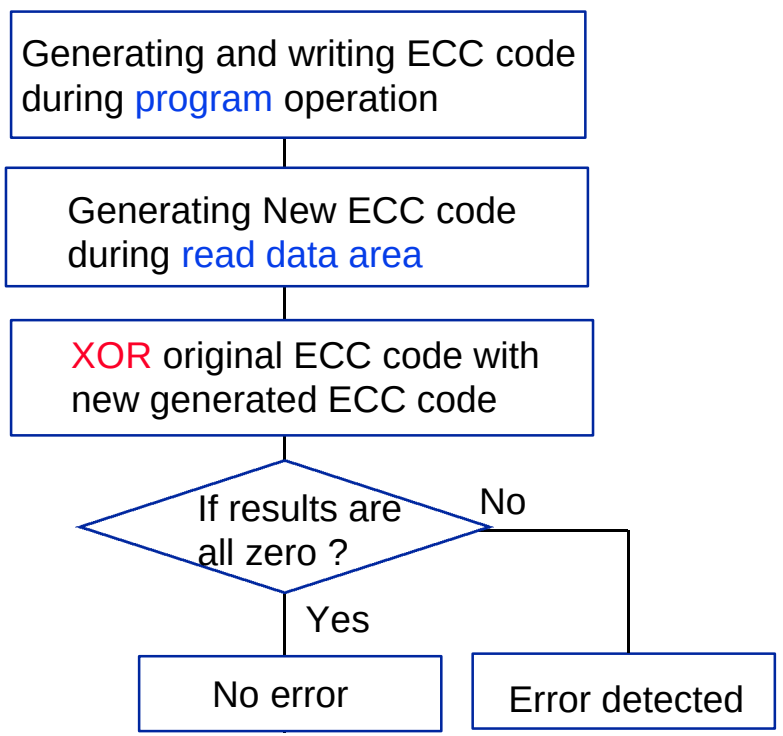
* Spare Area is same as above page.

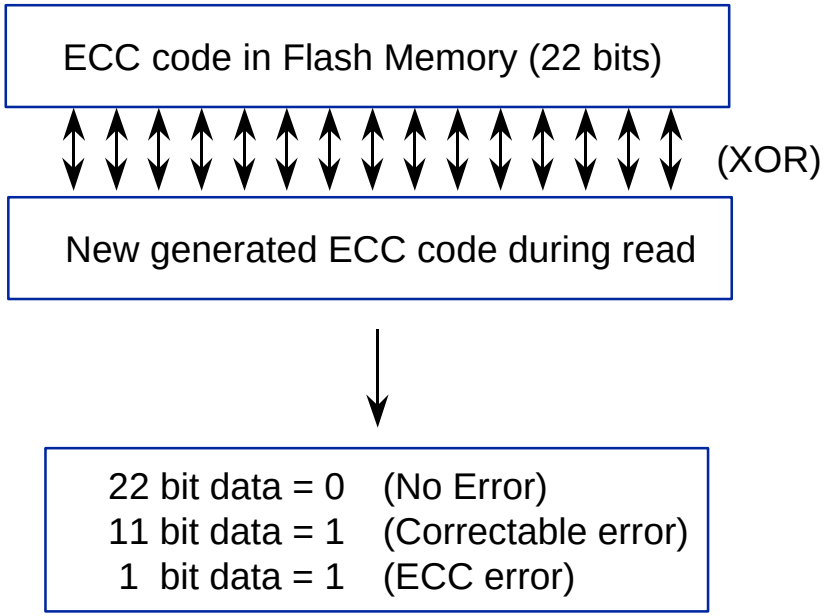
Delete A.TXT in Root directory

ECC Code Generation Method

- **ECC code consists of 3 Bytes per 256 Bytes(Hamming Code ECC Algo)**
  - **Actually 22 bit ECC code per 2048 bits**
  - **22 bit ECC code = 16 bit line parity + 6 bit column parity**

Error Detection Sequence





Considerations for High Density SmartMedia(1)

12bit FAT
Useful Cluster Number = 2^{12} = 4096 (Approximately 4000 Cluster Count Available)

16bit FAT Operation
Useful Cluster Number = 2^{16} = 65536 (Approximately 64000 Cluster Count Available)

12bit FAT Specification Table

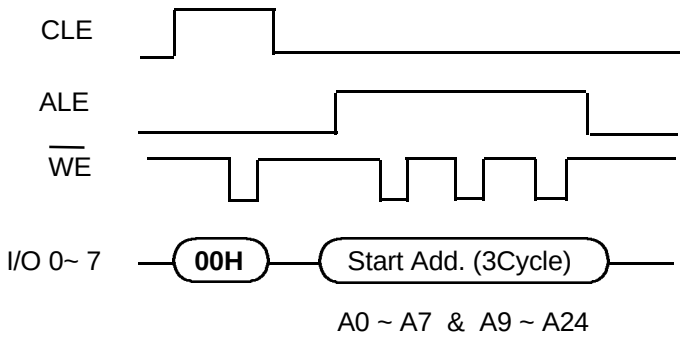
12-bit FAT	16-bit FAT	Description
000h	0000h	Unused Cluster
001h	0001h	Reserved
002h ~ FEFh	0002h ~ FFEFh	Next Cluster Number in the chain
FF0h ~ FF6h	FFF0h ~ FFF6h	Reserved
FF7h	FFF7h	Defective Cluster
FF8h ~ FFFh	FFF8h ~ FFFFh	Last Cluster in the chain

1~ 64MB(250 ~ 4,000 Cluster Chain Needs) : 12bit FAT Operation
128MB(8,000 Cluster Chain Needs) : 16bit FAT Operation

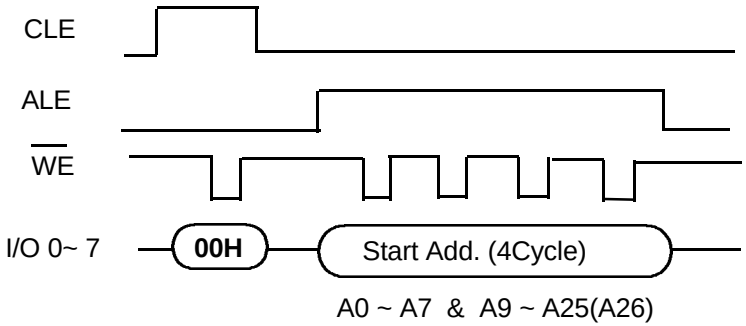
Considerations for High Density SmartMedia(2)

Four address cycles are needed for 64MB and 128MB SmartMedia !

~ 32MB SmartMedia



64MB,128MB SmartMedia



Address Configuration

1st Cycle	CA0 ~ CA7 : column address
2 nd Cycle	PA0 ~ PA7 : page address 1
3 rd Cycle	PA8 ~ PA15 : page address 2
4 th Cycle	PA16 ~ PA23 : page address 3

Model	Valid Page Address	Fixed Low
2MB	PA0 ~ PA12	PA13 ~ PA 15
4MB	PA0 ~ PA12	PA13 ~ PA 15
8MB	PA0 ~ PA13	PA14, PA 15
16MB	PA0 ~ PA14	PA15
32MB	PA0 ~ PA15	-
64MB	PA0 ~ PA16	PA17 ~ PA23
128MB	PA0 ~ PA17	PA18 ~ PA23

Considerations for High Density SmartMedia(3)

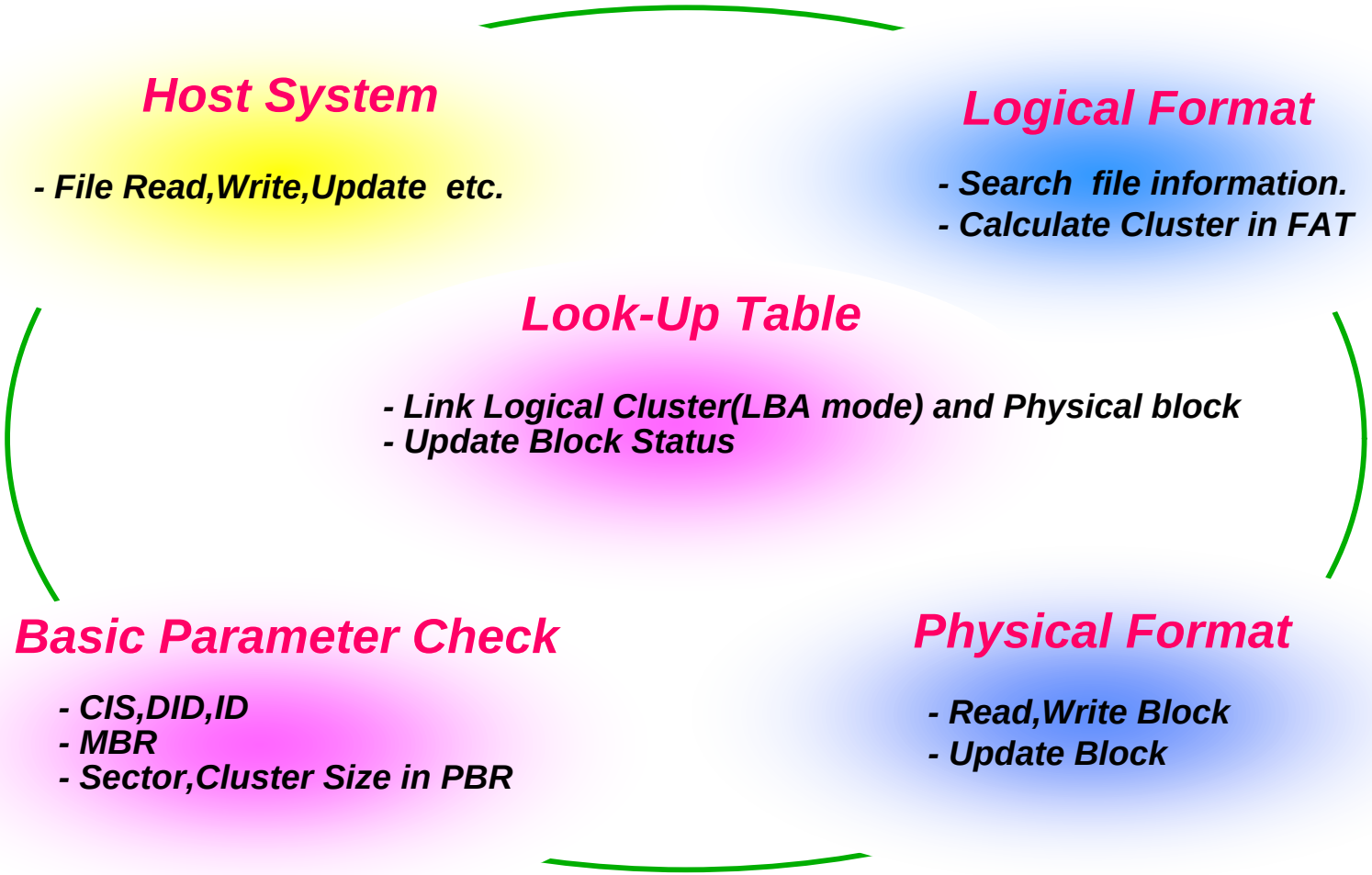
Zone-based block management for 32MB,64MB and 128MB

Zone	Physical Block	Description
0	0	CIS/Identify Drive Information Area
	1 ~ 1023	Data Area (Logical Block : 0 ~ 999)
1	0 ~ 1023	Data Area (Logical Block :1000 ~1999)
:	:	:
Final Zone	0 ~ 1023	Data Area (Logical Block : Zone x 1000 + 999)

* CIS/Identify Drive Information Area ==>Zone 0

Each zone has 1000 data blocks.

Software Functional Blocks



Data Updating Procedure

Updating into empty blocks reduces memory demands and avoid excessive block usage