

HS-MMC Application Note

April 2004

Product Planning & Application Engineering Team

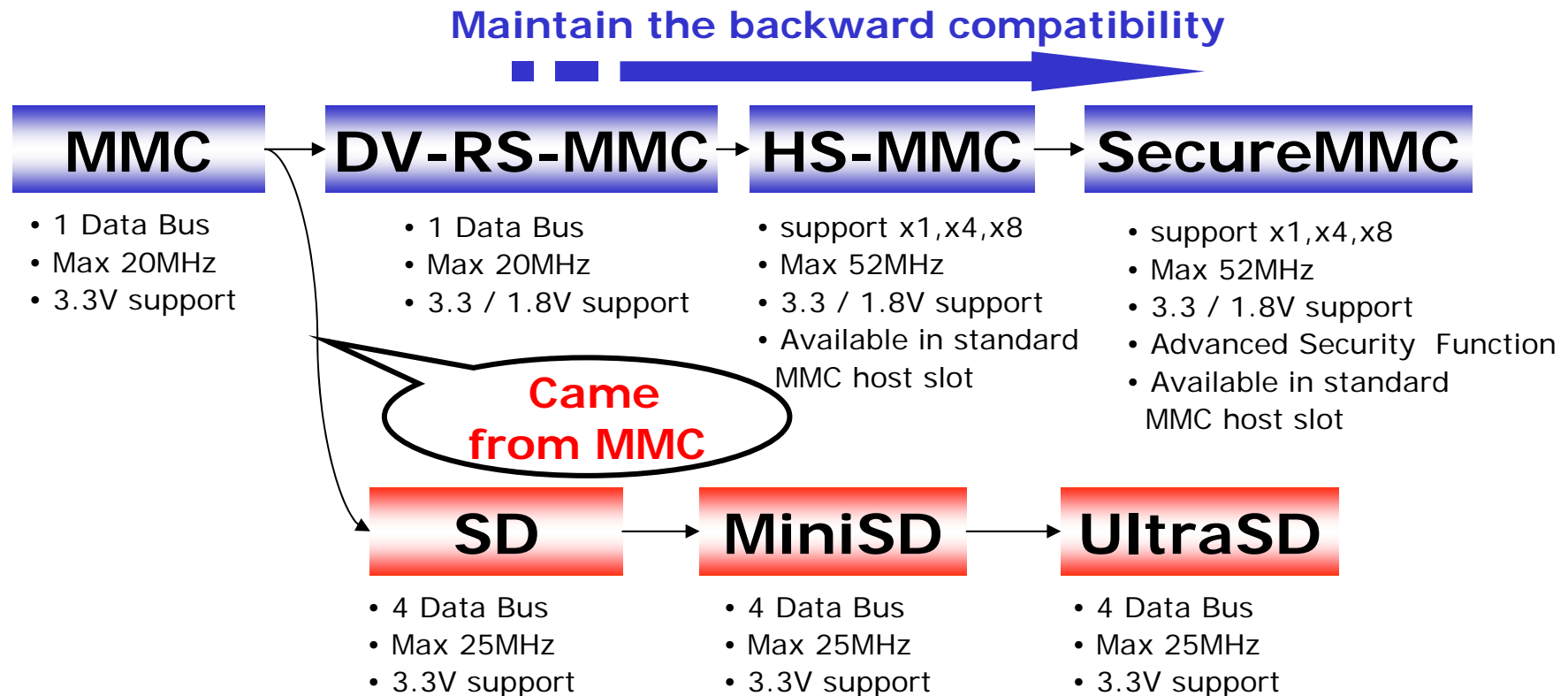
**MEMORY DIVISION
SAMSUNG ELECTRONICS Co., LTD**

Table of Contents

- ❑ History of MMC
- ❑ Why HS-MMC Card?
- ❑ Advantage of MMC
- ❑ General Status
- ❑ Comparison between HS-MMC and MMC
 - Bus Initialization
 - Bus width change & Bus test
 - Frequency change
 - New command set
 - New Register
 - Basic operation comparison
 - Timing Reference
 - Pin out comparison
 - Mechanical spec
- ❑ Conclusion
- ❑ Appendix

History of MMC

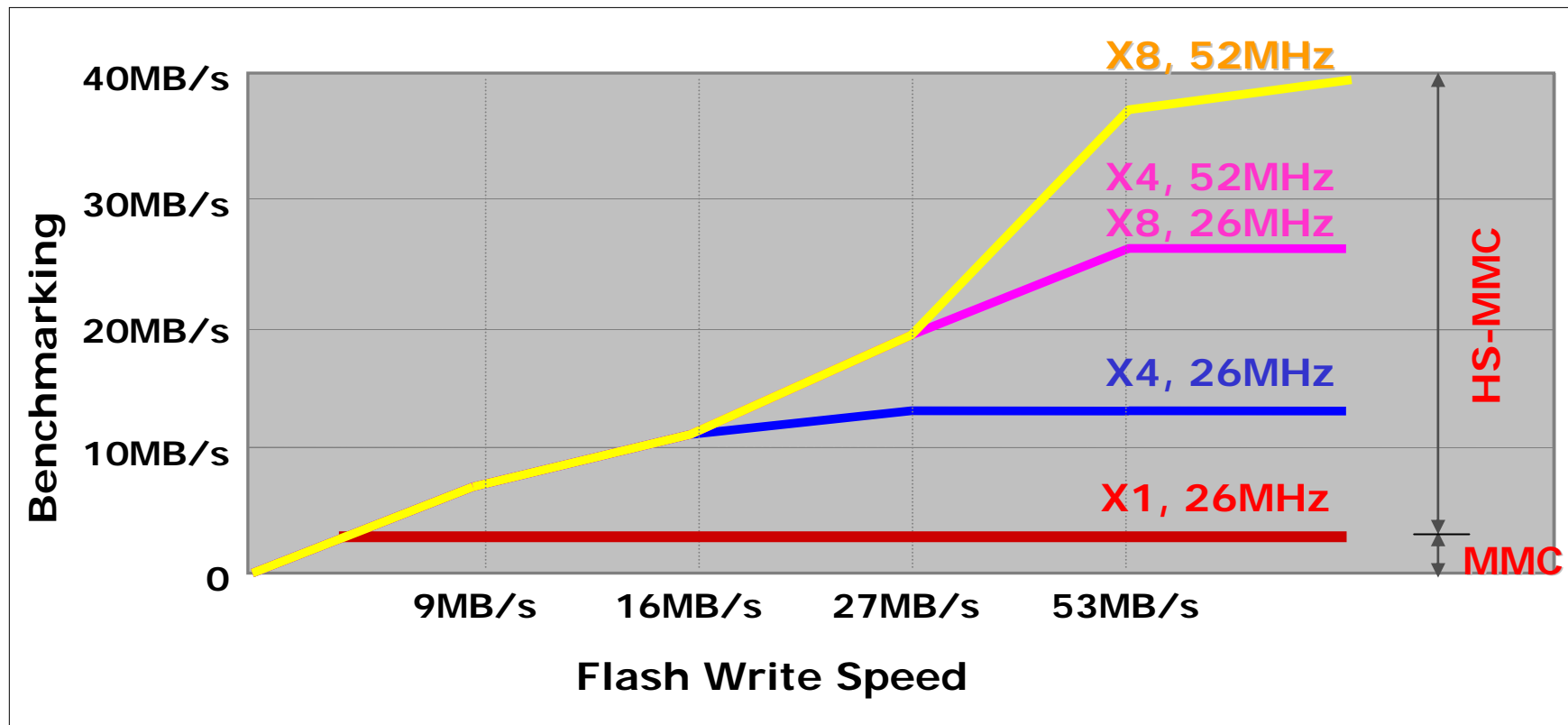
- ❑ SD cards originally came from MMC



- Almost all DSC vendors support both MMC and SD easily

Why HS-MMC Card?

- ❑ MMC Cards do not utilize the Flash Write Speed because of the limitation for data bus & clock frequency.
 - *HS-MMC provides the expandability of performance*



Advantage of MMC

- ❑ MMCA **Open Standard**
- ❑ No License Fee, **No Royalty**, lowest cost
- ❑ Widest choice of top card suppliers with their own controller & flash design & fabrication
- ❑ Over 150 members representing the **leading manufacturers** of host systems, cards, controllers, connectors, etc
- ❑ Best specs (size, **speed, voltage**, security, compliance, etc.)
- ❑ Already adopted by 3 of the top 5 mobile phone suppliers

General Status

- ❑ **MMCA become a Giant in Card industry**
 - Over 150 members representing the leading manufacturers of host systems, cards, controllers, connectors, etc
- ❑ **Main MMC makers are key semiconductor suppliers and can provide solutions**
 - Samsung, Micron, Infineon, ATP, PDC, etc
- ❑ **Major application is now mobile-phone based x1 MMC, and the various application is coming soon because of the open standard advantage**
- ❑ **MMC will be main volume card in next year**
 - 52 million units in 2005 (source : Gartner '03)
- ❑ **HS-MMC Spec finalization Q4'03 done**
- ❑ **High speed first cards available Q2'04 (by Samsung)**

Comparison btw. HS-MMC and MMC

Item		HS-MMC	Comment
Bus Initialization	Power-up Sequence	Same as MMC	See Attach #1
	Register Assignment	Adding Ext. CSR Assignment	See Attach #2
Frequency	Initial	400KHz or Less, Same as MMC	
	Operating	0~20MHz, 26MHz, 52MHz	See Attach #3
Power Class selection		Power Consumption levels	See Attach #4
Changing Data Bus Width & Bus testing pattern		X1, X4, X8	See Attach #5
New Command		CMD6, CMD8, CMD14, CMD19	See Attach #6
New Register		Adding EXT_CSD	See Attach #7
Basic Operation (WR/RD*1)		Compatible as MMC	See Attach #8
Timing Reference		Timing parameters referenced by the clock center	See Attach #9
Pin-out comparison		Adding more data pins	See Attach#10
Mechanical Spec		Compatible with MMC	See Attach#11

*1 :Write/Read

Power-up Sequence (#1)

□ Power-up Sequence for HS-MMC

➤ Same as MMC

1. Apply power to the bus, communication voltage range
2. Set clock to 400KHz, or less
3. Wait for 1ms, then wait for 74 more clock cycles
4. Send CMD0 to reset the bus, keep DAT3 high during this step for MMC mode (If DAT3 line is low, Card converse to SPI mode)
5. Send CMD1, with the intended voltage range in the argument (either High Voltage or Dual Voltage)
6. Receive R3 (Provide the content of the OCR register)
7. If the OCR busy bit is '0', repeat steps 5 and 6
8. From the R3 response argument the host can learn if the card is a high voltage or dual voltage card.
 - ✓ R3 Response : 0x80FF8000 = High Voltage Card
0x80FF8080 = Dual Voltage Card
9. If R3 returned some other value, the card is not compliant (since it should have put itself into *inactive* state, due to voltage incompatibility, and not respond); in such a case the host must power down the bus and start error recovery procedure. (the definition of error recovery procedure is host dependent)

Register Assignment (#2)

□ Register Assignment and Adjustment

- EXT_CSD Register is added.
(See the blue color sentence)

1. Send CMD2 to ask all cards for sending Card-Identification number
2. Receive R2 and get the card's Card-Identification
3. Send CMD3 with a chosen RCA, with value greater than 1
4. Repeat 1 to 3 until there is no response for CMD2
5. Send CMD9 for asking the selected card's Card-Specific Data
6. Receive R2, and get the card's Card-Specific Data from it.
7. Adjust the host parameters, if necessary, according to the information in CSD.

If the SPEC_VERS Field in CSD indicates a version 4.0 or higher, the card is a high speed card and supports SEND_EXT_CSD commands (CMD8) for the card properties and selected modes. The host can change the active command set by issuing the SWICH command (CMD6)

- Properties Segment : Card Capability Information like Card Type
- Mode Segment : High Speed Interface Timing, Bus Width, and others.

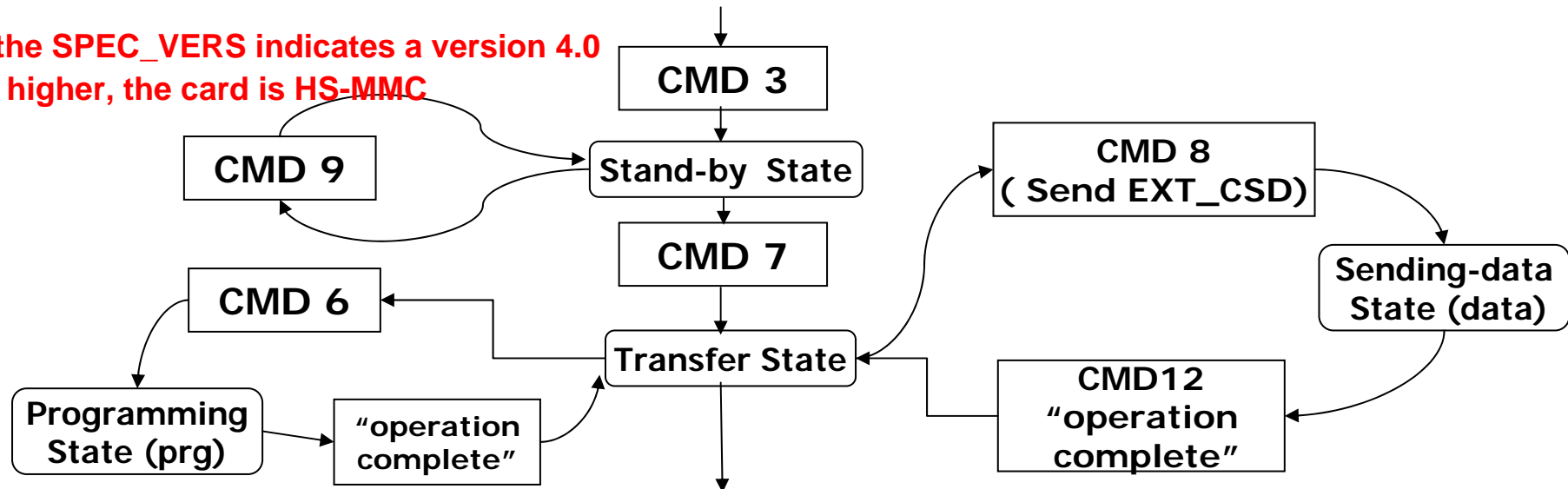
* Regardless of the type of card, the maximum clock frequency that can be set at this point is defined in the TRAN_SPEED field.

Switching to High Speed Mode (#3)

□ High Speed Mode supported by HS-MMC only

1. From EXT_CSD the host can learn the power class of the card, and the card type like the HS-MMC with the certain frequency.
2. Send CMD6, writing 0x1 to the HS_Timing byte of the EXT_CSD to enable the high speed interface timing. The argument 0x03B9_0100 will do it.
 - The card might enter BUSY it is configured for high speed timing
 - After the card comes out of BUSY it is configured for high speed timing
3. Change the clock frequency to the chosen frequency

If the SPEC_VERS indicates a version 4.0 or higher, the card is HS-MMC



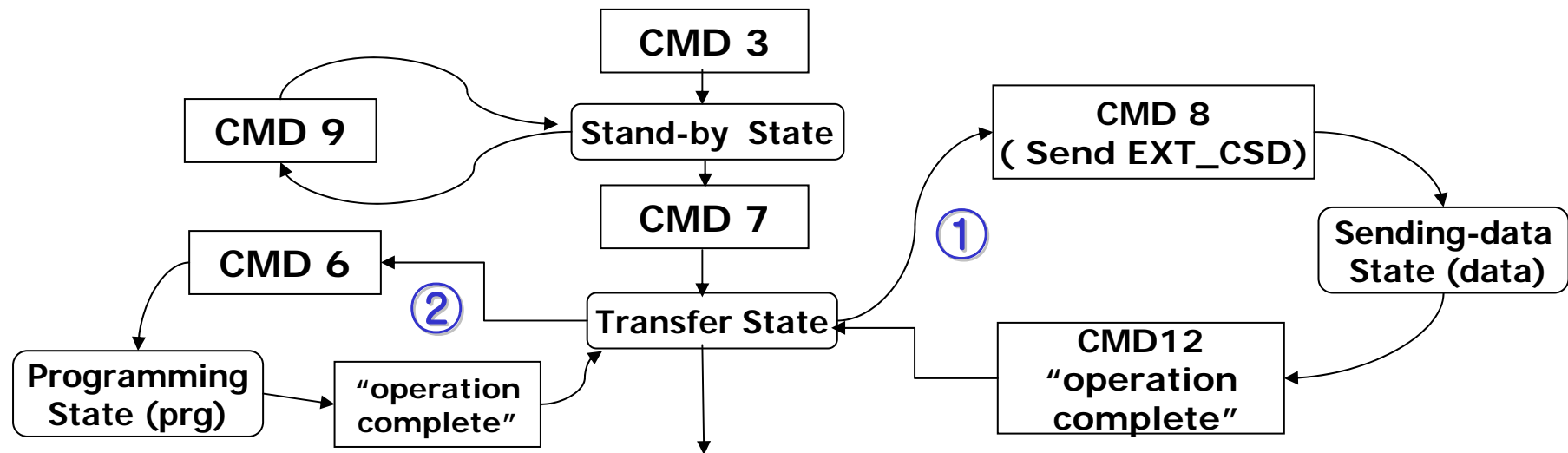
* Enable the high speed mode timing in the card and then select 26MHz or 52MHz through EXT_CSD Register.

Power Class Selection (#4)

□ Power Class Selection for HS-MMC

➤ The host may change the power class of the card.

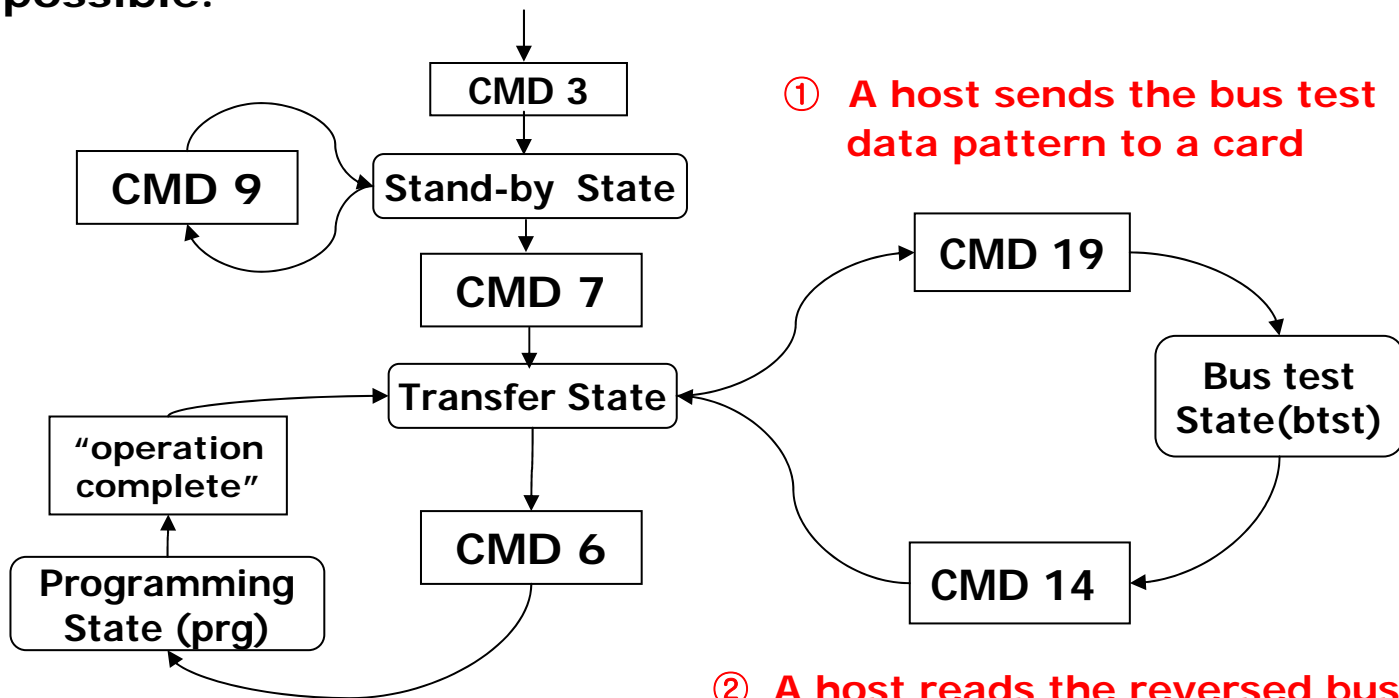
1. The default card power class is class 0 (minimum current consumption class for the card type, either High or Dual voltage card)
2. Host sends CMD8, and determines if it will allow the card to use higher power class.
3. If a power class change is needed, send CMD6 to write POWER_CLASS in EXT_CSD register.



Changing Data Bus Width (#5)

□ Flow Chart of changing data bus width

- If the card's power class allows the host to work on a wider bus, with in the host power budget, the following steps could be possible.



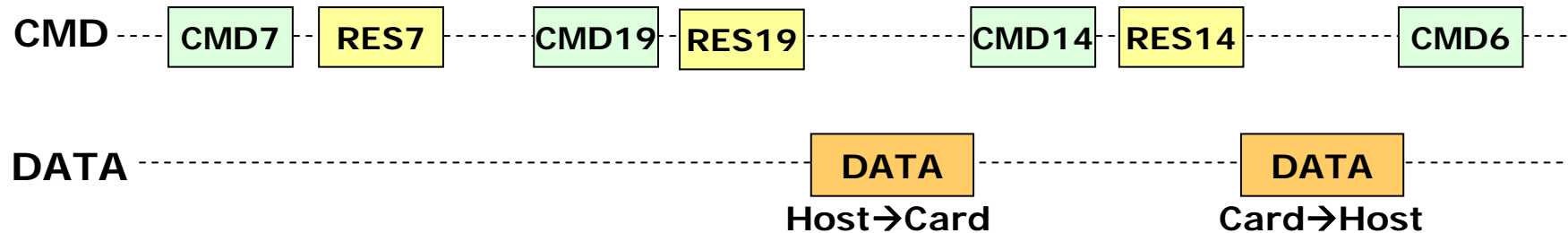
① A host sends the bus test data pattern to a card

② A host reads the reversed bus testing Data pattern from a card

③ Host determine the bus line connection

Changing Data Bus Width (#5)

□ Procedure



1. Card is initialized just like a normal MultiMediaCard
2. The card is controlled to TransferState (by CMD7 with RCA)
3. Host sends a data pattern with a new command (CMD19), card sends a response and latches received data pattern internally.
4. If the host sent another new command (CMD14), then the card would send the reversed pattern of received data.
5. Host determines the bus line connection or the number of the card I/O through CMD6
6. The bus is ready to exchange data using the new width configuration.

Bus Testing Pattern (#5)

- ❑ The card ignores all but the two first bit of the data pattern.
- ❑ The host ignores all but the two first bits of the reverse data pattern.
- ❑ CRC16 checksum is also ignored by the card and host.

Table : 1-bit Bus Testing Pattern

Data line	Data pattern sent by the host	Reserved pattern sent by the card	Notes
DATA0	0 10xxxxxxxxxxxx, [CRC16], 1	0 01 000000, [CRC16], 1	Start bit defines beginning of pattern
DATA1	↓	0,00000000, [CRC16], 1	No data pattern sent
DATA2	Data Pattern	0,00000000, [CRC16], 1	No data pattern sent
DATA3		0,00000000, [CRC16], 1	No data pattern sent
DATA4		0,00000000, [CRC16], 1	No data pattern sent
DATA5		0,00000000, [CRC16], 1	No data pattern sent
DATA6		0,00000000, [CRC16], 1	No data pattern sent
DATA7		0,00000000, [CRC16], 1	No data pattern sent

Bus Testing Pattern (#5)

Table : 4-bit Bus Testing Pattern

Data line	Data pattern sent by the host	Reserved pattern sent by the card	Notes
DATA0	0 10xxxxxxxxxxxx, [CRC16], 1	0 01000000, [CRC16], 1	Start bit defines beginning of pattern
DATA1	0 01xxxxxxxxxxxx, [CRC16], 1	0 10000000, [CRC16], 1	
DATA2	0 10xxxxxxxxxxxx, [CRC16], 1	0 01000000, [CRC16], 1	
DATA3	0 01xxxxxxxxxxxx, [CRC16], 1	0 10000000, [CRC16], 1	
DATA4		0,00000000, [CRC16], 1	No data pattern sent
DATA5		0,00000000, [CRC16], 1	No data pattern sent
DATA6		0,00000000, [CRC16], 1	No data pattern sent
DATA7		0,00000000, [CRC16], 1	No data pattern sent

Table : 8-bit Bus Testing Pattern

Data line	Data pattern sent by the host	Reserved pattern sent by the card	Notes
DATA0	0 10xxxxxxxxxxxx, [CRC16], 1	0 01000000, [CRC16], 1	Start bit defines beginning of pattern
DATA1	0 01xxxxxxxxxxxx, [CRC16], 1	0 10000000, [CRC16], 1	
DATA2	0 10xxxxxxxxxxxx, [CRC16], 1	0 01000000, [CRC16], 1	
DATA3	0 01xxxxxxxxxxxx, [CRC16], 1	0 10000000, [CRC16], 1	
DATA4	0 10xxxxxxxxxxxx, [CRC16], 1	0 01000000, [CRC16], 1	
DATA5	0 01xxxxxxxxxxxx, [CRC16], 1	0 10000000, [CRC16], 1	
DATA6	0 10xxxxxxxxxxxx, [CRC16], 1	0 01000000, [CRC16], 1	
DATA7	0 01xxxxxxxxxxxx, [CRC16], 1	0 10000000, [CRC16], 1	

New Command (#6)

- Only HS-MMC supports CMD6, CMD8, CMD14, CMD19

	Abbreviation	Command Description	Argument
CMD6	SWITCH	Switches the mode of operation of the selected card or modifies the EXT_CSD reg.	[31:26] Set to 0 [25:24] Access [23:16] Index [15:8] Value [7:3] Set to 0 [2:0] Cmd set
CMD8	SEND_EXT_CSD	The card sends its EXT_CSD reg. as a block of data.	[31:0] stuff bits
CMD14	BUSTEST_R	A host reads the reserved bus testing data pattern from a card.	[31:0] stuff bits
CMD19	BUSTEST_W	A host sends the bus test data pattern to a card.	[31:0] stuff bits

New Reg. : EXT_CSD Register (#7)

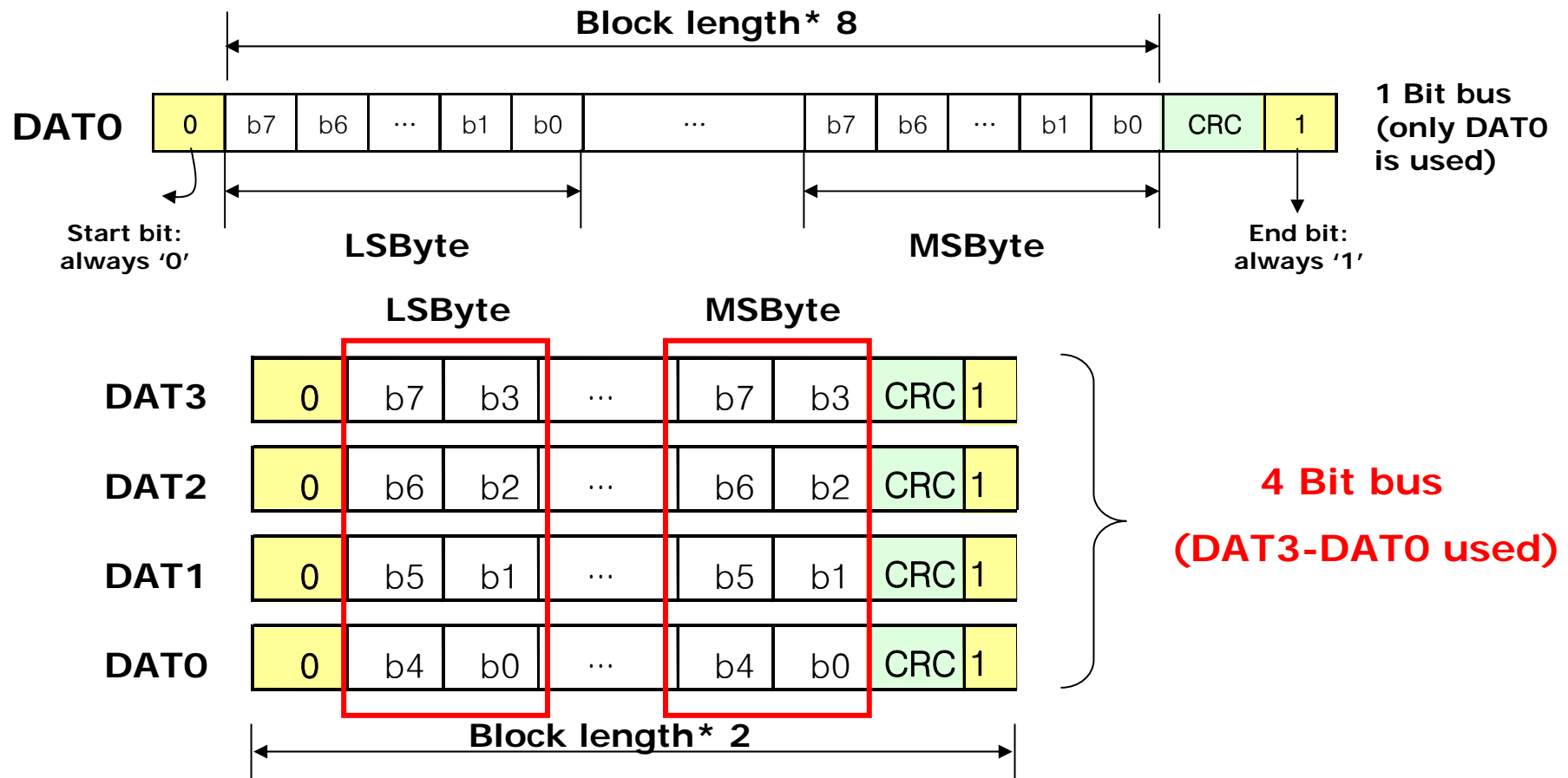
- ❑ Extended CSD Register(512Bytes) is added to HS-MMC
- ❑ Defines the card properties(320Bytes) and selected modes(192Bytes).
- ❑ Modes can be changed by SWITCH command.

Name	Field	Size(Bytes)	Cell Type	CSD-slice (Byte)	Value
Supported command sets	S_CMD_SET	1	R	[504]	0
Power Class for 26MHz@3.6V	PWR_CL_26_360	1	R	[203]	-
Power Class for 52MHz@3.6V	PWR_CL_52_360	1	R	[202]	-
Power Class for 26MHz@1.95 V	PWR_CL_26_195	1	R	[201]	-
Power Class for 52MHz@1.95V	PWR_CL_52_195	1	R	[200]	-
Card Type	CARD_TYPE	1	R	[196]	-
CSD Structure Version	CSD_STRUCTURE	1	R	[194]	3
Extended CSD revision	EXT_CSD_REV	1	R	[192]	0
Command set	CMD_SET	1	R/W	[191]	-
Command set revision	CMD_SET_REV	1	RO	[189]	-
Power Class	POWER_CLASS	1	R/W	[187]	-
High Speed Interface Timing	HS_TIMING	1	R/W	[185]	-
Bus Width Mode	BUS_WIDTH	1	WO	[183]	-

Basic Operation (#8)

□ Basic operation is SAME.

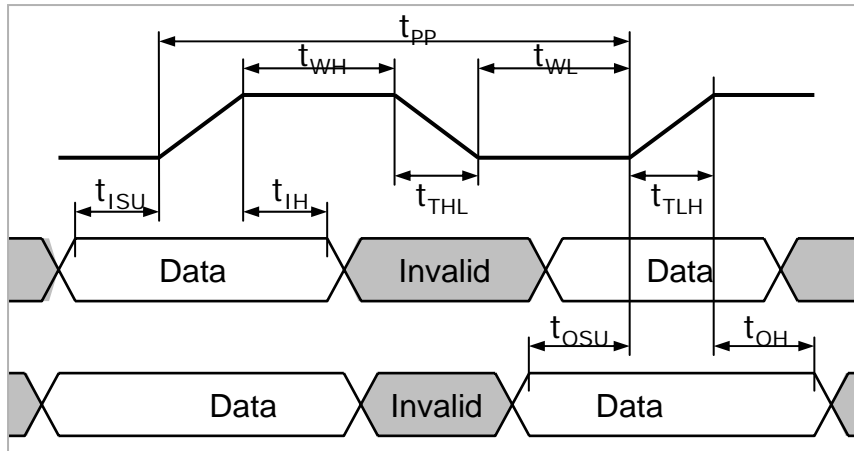
- CRC calculation is used for all bus configuration.
- x4 & x8 is just added CRC for each line separately.
- 8 bit data bus has just additional 4 data line to 4 bit bus.



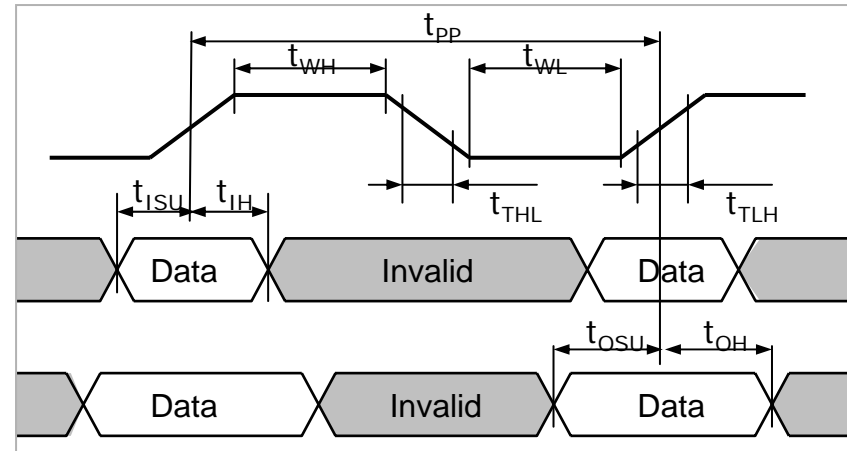
Timing Reference (#9)

- HS-MMC timing parameters referenced by the clock center

MMC 3.3 Version



HS-MMC 4.0 Version



Parameter	
Clock Low Time	t_{WL}
Clock High Time	t_{WH}
Clock Rise Time	t_{TLH}
Clock Fall Time	t_{THL}
Input Setup Time	t_{ISU}
Input Hold Time	t_{IH}
Output Setup Time	t_{OSU}
Output Hold Time	t_{OH}

MMC 3.3 Version	
Min	Max
*50 / 10 ns	
*50 / 10 ns	
	50* / 10 ns
	50* / 10 ns
3 ns	
3 ns	
5 ns	
5 ns	

HS-MMC 4.0 Version	
Min	Max
6.5 ns	
**6.5 ns	
	3 ns
	3 ns
3 ns	
3 ns	
5 ns	
5 ns	

* N=30, ** estimation, Not defined in specification

Pin-out Comparison (#10)

- HS-MMC is fully pin compatible with SD x4 & MMC

MMC

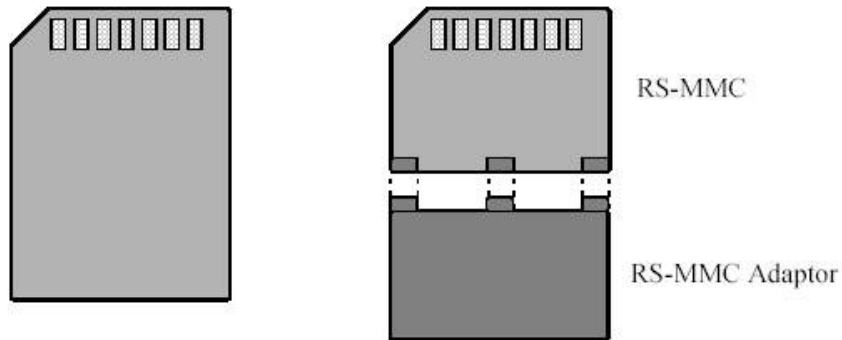


Figure 5: MultiMediaCard Form Factors (Bottom View)

HS-MMC

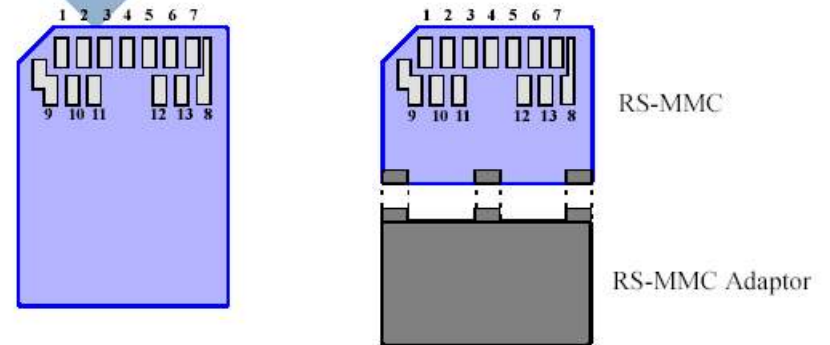
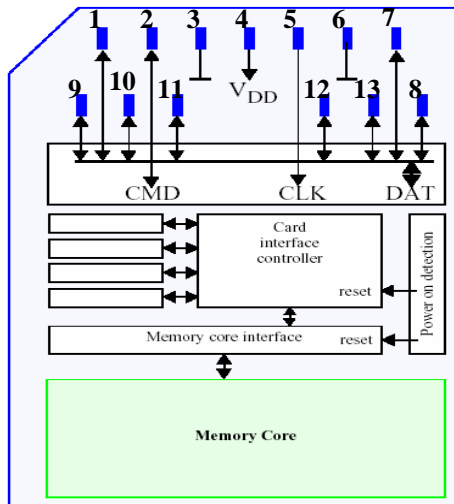


Figure 5: MultiMediaCard Form Factors (Bottom View)

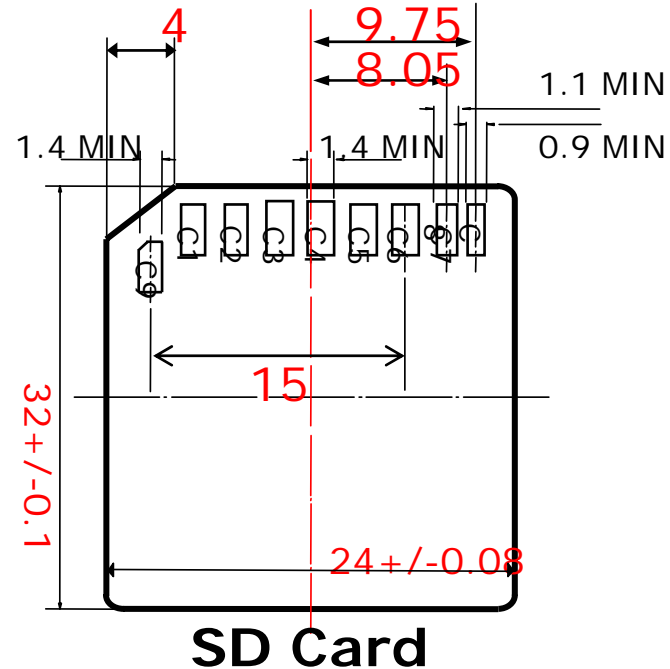
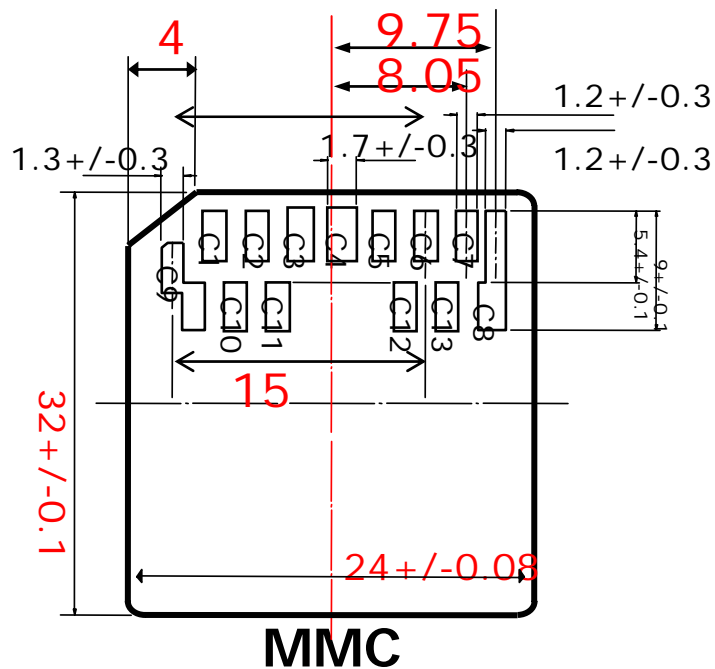


Added for x8

Pin No.	Name	Type	Description
1	DAT3	I/O/PP	DATA
2	CMD	I/O/PP/OD	Command/Response
3	VSS1	S	Supply voltage ground
4	VDD	S	Supply voltage
5	CLK	I	Clock
6	VSS2	S	Supply voltage ground
7	DAT0	I/O/PP	DATA
8	DAT1	I/O/PP	DATA
9	DAT2	I/O/PP	DATA
10	DAT4	I/O/PP	DATA
11	DAT5	I/O/PP	DATA
12	DAT6	I/O/PP	DATA
13	DAT7	I/O/PP	DATA

Mechanical Spec (#11)

- HS-MMC maintains the compatibility with all cards in terms of mechanical spec.



Conclusion

- ❑ **HS-MMC maintains the backward compatibility with MMC**
 - Same basic operation with MMC
 - Add few commands & one register for advanced operation
- ❑ **HS-MMC can provide all advantages as open standard in terms of better availability, cost competitiveness, and others**
- ❑ **Just little modification of firmware, SD x4 host can easily support HS-MMC.**
 - HS-MMC has no more performance degradation by adopting 20MHz
- ❑ **HS-MMC performs superior performance than MMC & SD**
 - X4/x8 with 26MHz/52MHz : ideally 52MB/s (Bus bandwidth)
- ❑ **Performance upgradability & long-term solution**
 - 13MB/s -> 26MB/s ->52MB/s @ Host I/F
- ❑ **Provide the excellent interoperability for all applications**
 - such as Mobile phone, DSC, DVC, PDA, and others

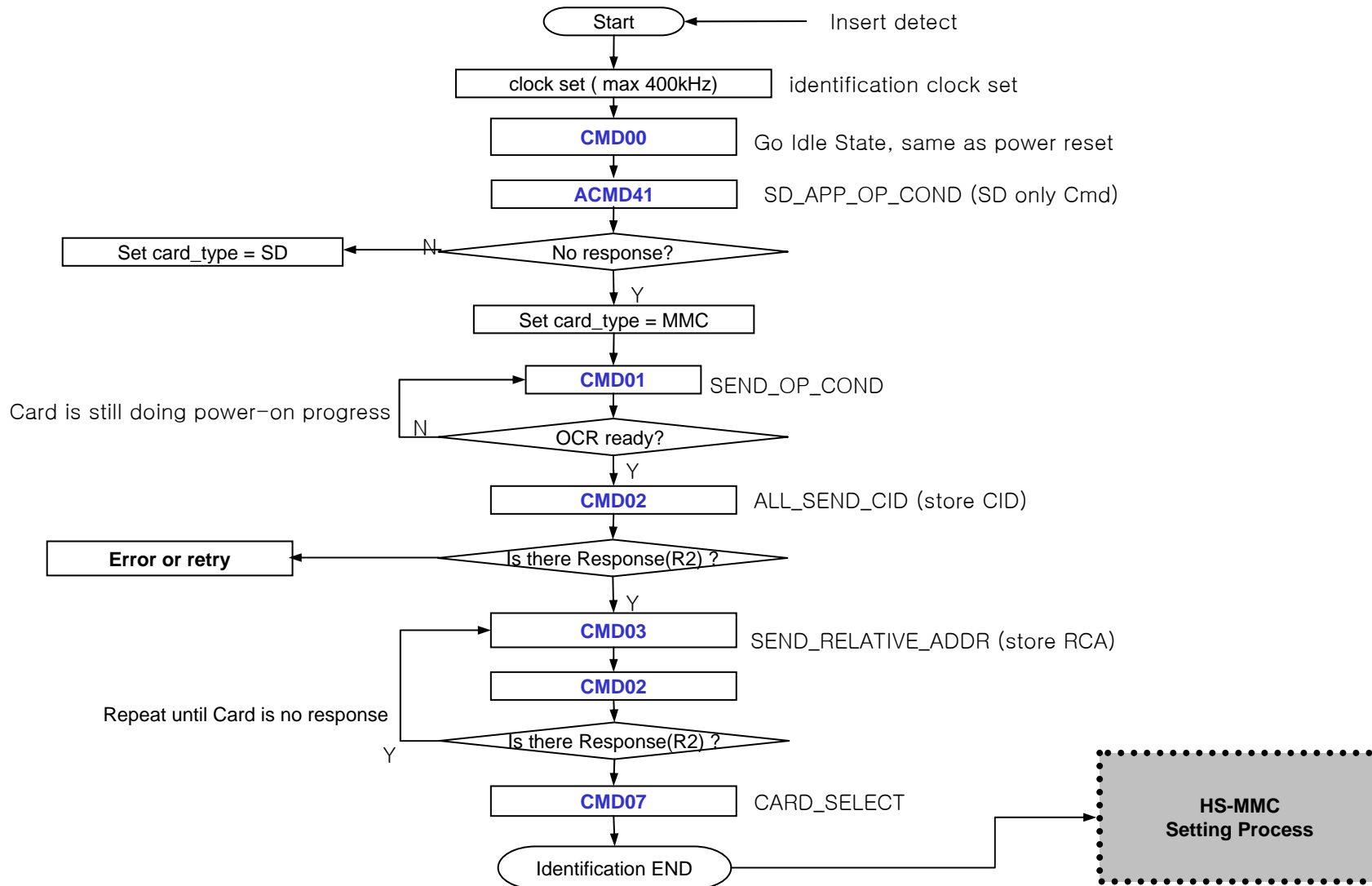
Easy to support HS-MMC by little modification of firmware and can get the higher performance

Appendix

(Host Implementation Guideline)

Card Identification Procedure (1/2)

□ MMC/HS-MMC/SD Identification Process



Card Identification Procedure (2/2)

□ HS-MMC Setting Process

➤ **Precondition : The card state should be transfer state**

