

JEDEC STANDARD

**Embedded MultiMediaCard(*e*.MMC)
e.MMC/Card Product Standard,
High Capacity, including Reliable
Write, Boot, Sleep Modes, Dual Data
Rate, Multiple Partitions Supports
and Security Enhancement
(MMCA, 4.4)**

JESD84-A44

MARCH 2009

JEDEC SOLID STATE TECHNOLOGY ASSOCIATION



NOTICE

JEDEC standards and publications contain material that has been prepared, reviewed, and approved through the JEDEC Board of Directors level and subsequently reviewed and approved by the JEDEC legal counsel.

JEDEC standards and publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for use by those other than JEDEC members, whether the standard is to be used either domestically or internationally.

JEDEC standards and publications are adopted without regard to whether or not their adoption may involve patents or articles, materials, or processes. By such action JEDEC does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the JEDEC standards or publications.

The information included in JEDEC standards and publications represents a sound approach to product specification and application, principally from the solid state device manufacturer viewpoint. Within the JEDEC organization there are procedures whereby a JEDEC standard or publication may be further processed and ultimately become an ANSI standard.

No claims to be in conformance with this standard may be made unless all requirements stated in the standard are met.

Inquiries, comments, and suggestions relative to the content of this JEDEC standard or publication should be addressed to JEDEC at the address below, or call (703) 907-7559 or www.jedec.org

Published by

©JEDEC Solid State Technology Association 2009

3103 North 10th Street, Suite 240 South

Arlington, VA 22201

This document may be downloaded free of charge; however JEDEC retains the copyright on this material. By downloading this file the individual agrees not to charge for or resell the resulting material.

PRICE: Please refer to the current

Catalog of JEDEC Engineering Standards and Publications online at

<http://www.jedec.org/Catalog/catalog.cfm>

Printed in the U.S.A.

All rights reserved

PLEASE!

DON'T VIOLATE THE LAW!

This document is copyrighted by JEDEC and may not be reproduced without permission.

Organizations may obtain permission to reproduce a limited number of copies through entering into a license agreement. For information, contact:

JEDEC Solid State Technology Association
3103 north 10th Street, Suite 240 South
Arlington, Virginia 22201-2107
or call (703) 907-7559

Embedded MultiMediaCard (e•MMC) e•MMC/Card Product Standard, High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data Rate, Multiple Partitions Supports and Security Enhancement

CONTENTS(*continued*)

1	Scope	1
2	Normative reference	1
3	Terms and definitions	1
3	Terms and definitions (continued).....	2
3	Terms and definitions (continued).....	3
4	General description	5
5	System features	7
6	MultiMediaCard system concept	11
6.1	Higher than a density of 2GB	14
6.2	MMCplus and MMCmobile	14
6.3	Card concept	14
6.3.1	Form factors	17
6.4	Bus concept	17
6.4.1	Bus lines	17
6.4.2	Bus protocol	18
6.5	Controller Concept	23
6.5.1	Application adapter requirements	24
6.5.2	MultiMediaCard adapter architecture	24
7	MultiMediaCard functional description	27
7.1	General	27
7.2	Partition Management	29
7.2.1	General	29
7.2.2	Command restrictions	31
7.2.3	Configure partitions	31
7.2.4	Access partitions	33
7.3	Boot operation mode	33
7.3.1	Card reset to Pre-idle state	33
7.3.2	Boot partition	35
7.3.3	Boot operation	35

Embedded MultiMediaCard (e•MMC) e•MMC/Card Product Standard, High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data Rate, Multiple Partitions Supports and Security Enhancement

CONTENTS(*continued*)

7.3.4	Alternative boot operation	37
7.3.5	Access to boot partition	39
7.3.6	Boot bus width and data access configuration	39
7.3.7	Boot Partition Write Protection	39
7.4	Card identification mode	40
7.4.1	Card reset	40
7.4.2	Operating voltage range validation	40
7.4.3	Access mode validation (higher than 2GB of densities)	42
7.4.4	From busy to ready	42
7.4.5	Card identification process	43
7.5	Interrupt mode	44
7.6	Data transfer mode	45
7.6.1	Command sets and extended settings	47
7.6.2	High-speed mode selection	47
7.6.3	Power class selection	48
7.6.4	Bus testing procedure	48
7.6.5	Bus width selection	50
7.6.6	Data read	50
7.6.7	Data write	52
7.6.8	Erase	55
7.6.9	Secure Erase	57
7.6.10	Secure Trim	58
7.6.11	TRIM	59
7.6.12	Write protect management	60
7.6.13	Card lock/unlock operation	62
7.6.14	Application-specific commands	65
7.6.15	Sleep (CMD5)	66
7.6.16	Replay Protected Memory Block	66
7.6.17	Dual Data Rate mode selection	76
7.6.18	Dual Data Rate mode operation	76
7.7	Clock control	77
7.8	Error conditions	77
7.8.1	CRC and illegal command	77
7.8.2	Time-out conditions	77
7.8.3	Read ahead in stream and multiple block read operation	78
7.9	Minimum performance	78

Embedded MultiMediaCard (e•MMC) e•MMC/Card Product Standard, High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data Rate, Multiple Partitions Supports and Security Enhancement

CONTENTS(*continued*)

7.9.1	Speed class definition	78
7.9.2	Measurement of the performance	79
7.10	Commands	80
7.10.1	Command types	80
7.10.2	Command format	80
7.10.3	Command classes	80
7.10.4	Detailed command description	82
7.11	Card state transition table	88
7.12	Responses	90
7.13	Card status	92
7.14	Memory array partitioning	96
7.15	Timings	97
7.15.1	Command and response	98
7.15.2	Data read	99
7.15.3	Data write	101
7.15.4	Bus test procedure timing	104
7.15.5	Boot operation	104
7.15.6	Alternative boot operation	106
7.15.7	Timing values	107
7.15.8	H/W Reset operation	108
7.15.9	Noise filtering timing for H/W Reset	108
8	Card registers	111
8.1	OCR register	111
8.2	CID register	111
8.3	CSD register	113
8.4	Extended CSD register	124
8.5	RCA register	149
8.6	DSR register	149
9	SPI mode	151
10	Error protection	153
10.1	Error correction codes (ECC)	153
10.2	Cyclic redundancy codes (CRC)	153

Embedded MultiMediaCard (*e*•MMC) *e*•MMC/Card Product Standard, High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data Rate, Multiple Partitions Supports and Security Enhancement

CONTENTS(*continued*)

11	MultiMediaCard mechanical specification	157
12	The MultiMediaCard bus	159
12.1	Hot insertion and removal	159
12.2	Power protection	160
12.3	Power-up	161
12.3.1	<i>e</i> •MMC power-up	162
12.3.2	<i>e</i> •MMC power-up guidelines	163
12.3.3	<i>e</i> •MMC power cycling	164
12.4	Programmable card output driver	165
12.5	Bus operating conditions	167
12.5.1	Power supply: high-voltage MultiMediaCard	167
12.5.2	Power supply: dual-voltage MultiMediaCard	167
12.5.3	Power supply: <i>e</i> •MMC	168
12.5.4	Power supply: <i>e</i> •-MMC	168
12.5.5	Bus signal line load	169
12.6	Bus signal levels	170
12.6.1	Open-drain mode bus signal level	170
12.6.2	Push-pull mode bus signal level—high-voltage MultiMediaCard	170
12.6.3	Push-pull mode bus signal level—dual-voltage MultiMediaCard	170
12.6.4	Push-pull mode bus signal level— <i>e</i> •MMC	171
12.7	Bus timing	172
12.7.1	Card interface timings	173
12.8	Bus timing for DAT signals during 2x data rate operation	175
12.8.1	Dual data rate interface timings	176
13	<i>e</i> •MMC standard compliance	179
14	File formats for the MultiMediaCard	183
Annex A: Application Notes		185
A.1	Power supply decoupling	185
A.2	Payload block length and ECC types handling	185
A.3	Connector	185
A.3.1	General	185
A.3.2	Card insertion and removal	186
A.3.3	Characteristics	186

Embedded MultiMediaCard (e•MMC) e•MMC/Card Product Standard, High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data Rate, Multiple Partitions Supports and Security Enhancement

CONTENTS(*continued*)

A.4	Description of method for storing passwords on the card	187
A.5	MultiMediaCard macro commands	188
A.6	Host interface timing	202
A.7	Handling of passwords	202
A.7.1	Changing the password	202
A.7.2	Removal of the password	203
A.8	High-speed MultiMediaCard bus functions	204
A.8.1	Bus initialization	204
A.8.2	Switching to high-speed mode	205
A.8.3	Changing the data bus width	205
A.9	Erase-unit size selection flow	208
Annex B:	Changes between system specification versions	209
B.1	Version 4.1, the first version of this specification	209
B.2	Changes from version 4.1 to 4.2	209
B.3	Changes from version 4.2 to 4.3	209
B.4	Changes from version 4.3 to 4.4	211

**Embedded MultiMediaCard (e•MMC) e•MMC/Card Product Standard,
High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data
Rate, Multiple Partitions Supports and Security Enhancement**

CONTENTS(*continued*)

This page intentionally left blank.

Embedded MultiMediaCard (eMMC) eMMC/Card Product Standard, High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data Rate, Multiple Partitions Supports and Security Enhancement

CONTENTS(*continued*)

Table 1 —	MultiMediaCard Voltage Modes	7
Table 2 —	MMC System Operational Mode	8
Table 3 —	MultiMediaCard interface pin configuration	15
Table 4 —	MultiMediaCard registers	16
Table 5 —	Bus modes overview	28
Table 6 —	EXT_CSD access mode	47
Table 7 —	Bus testing pattern	48
Table 8 —	1-bit bus testing pattern	49
Table 9 —	4-bit bus testing pattern	49
Table 10 —	8-bit bus testing pattern	49
Table 11 —	Erase command (CMD38) Valid arguments	56
Table 12 —	Erase Command Comparision	56
Table 13 —	Erase Command Argument Definition	57
Table 14 —	Write Protection Hierarchy (when disable bits are clear)	62
Table 15 —	Write Protection Types (when disable bits are clear)	62
Table 16 —	Lock card data structure	63
Table 17 —	RPMB Request/Response Message Types	67
Table 18 —	RPMB Operation Results data structure	68
Table 19 —	RPMB Operation Results	68
Table 20 —	Supported card command classes (0–56)	81
Table 21 —	Basic commands and read-stream command (class 0 and class 1)	82
Table 22 —	Block-oriented read commands (class 2)	83
Table 23 —	Stream write commands (class 3)	83
Table 24 —	Block-oriented write commands (class 4)	84
Table 25 —	Block-oriented write protection commands (class 6)	84
Table 26 —	Erase commands (class 5)	85
Table 27 —	I/O mode commands (class 9)	86
Table 28 —	Lock card commands (class 7)	87
Table 29 —	Application-specific commands (class 8)	87
Table 30 —	Card state transitions	88
Table 31 —	R1 response	90
Table 32 —	R2 response	91
Table 33 —	R3 response	91
Table 34 —	R4 response	91
Table 35 —	R5 response	91
Table 36 —	Card status	92
Table 37 —	Card status field/command—cross reference	95
Table 38 —	Timing parameters	107
Table 39 —	H/W reset timing parameters	108

Embedded MultiMediaCard (e•MMC) e•MMC/Card Product Standard, High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data Rate, Multiple Partitions Supports and Security Enhancement

CONTENTS(*continued*)

Table 40 —	OCR register definitions.....	111
Table 41 —	CID fields	112
Table 42 —	Device types	112
Table 43 —	CSD fields	113
Table 44 —	CSD register structure	115
Table 45 —	System specification version	115
Table 46 —	TAAC access-time definition.....	115
Table 47 —	Maximum bus clock frequency definition	116
Table 48 —	Supported card command classes.....	116
Table 49 —	Data block length	117
Table 50 —	DSR implementation code table.....	118
Table 51 —	V _{DD} (min) current consumption.....	118
Table 52 —	V _{DD} (max) current consumption	119
Table 53 —	Multiplier factor for device size.....	119
Table 54 —	R2W_FACTOR.....	120
Table 55 —	File formats	121
Table 56 —	ECC type	122
Table 57 —	CSD field command classes.....	122
Table 58 —	Extended CSD	124
Table 59 —	Card-supported command sets	127
Table 60 —	Initilaiztion Time out value	127
Table 61 —	TRIM Time out value.....	127
Table 62 —	SEC Feature Support.....	128
Table 63 —	Secure Erase Time out value.....	128
Table 64 —	Secure Trim Time out value.....	129
Table 65 —	Boot information	129
Table 66 —	Boot partition size	130
Table 67 —	Access size	130
Table 68 —	Superpage size.....	130
Table 69 —	Erase-unit size	131
Table 70 —	Erase timeout values.....	131
Table 71 —	Reliable write sector count.....	132
Table 72 —	Write protect group size	132
Table 73 —	S_C_VCC, S_C_VCCQ timeout values	132
Table 74 —	Sleep/awake timeout values	133
Table 75 —	R/W access performance values.....	133
Table 76 —	Power classes.....	135
Table 77 —	Card types.....	136
Table 78 —	CSD register structure	136

Embedded MultiMediaCard (e•MMC) e•MMC/Card Product Standard, High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data Rate, Multiple Partitions Supports and Security Enhancement

CONTENTS(*continued*)

Table 79 —	Extended CSD revisions.....	136
Table 80 —	Standard MMC command set revisions	137
Table 81 —	Power class codes.....	137
Table 82 —	Bus mode values.....	138
Table 83 —	Erased memory content values.....	138
Table 84 —	Boot configuration bytes	139
Table 85 —	Boot config protection.....	140
Table 86 —	Boot bus configuration.....	141
Table 87 —	ERASE_GROUP_DEF	141
Table 88 —	BOOT area write protection.....	142
Table 89 —	User area write protection	143
Table 90 —	FW Update Disable	144
Table 91 —	RPMB Partition Size	144
Table 92 —	H/W reset function	144
Table 93 —	Partitioning Support	145
Table 94 —	Max. Enhanced Area Size	145
Table 95 —	Partitions Attribute	146
Table 96 —	Partition Setting.....	146
Table 97 —	General Purpose Partition Size.....	147
Table 98 —	Enhanced User Data Area Size	148
Table 99 —	Enhanced User Data Start Address	148
Table 100 —	Secure Bad Block management.....	148
Table 101 —	Error correction codes	153
Table 102 —	DSR register content	165
Table 103 —	General operating conditions	167
Table 104 —	Power supply voltage: high-voltage MultiMediaCard	167
Table 105 —	Power supply voltage: dual-voltage MultiMediaCard	167
Table 106 —	e•MMC power supply voltage	168
Table 107 —	e•MMC voltage combinations.....	169
Table 108 —	Capacitance	169
Table 109 —	Open-drain bus signal level.....	170
Table 110 —	Push-pull signal level—high-voltage MultiMediaCard	170
Table 111 —	Push-pull signal level—dual-voltage MultiMediaCard	171
Table 112 —	Push-pull signal level—1.1V-1.3V VCCQ range e•MMC	171
Table 113 —	High-speed card interface timing	173
Table 114 —	Backward-compatible card interface timing	173
Table 115 —	High-speed dual rate interface timing	176
Table 116 —	MultiMediaCard host requirements for card classes.....	179
Table 117 —	New Features List for device type.....	180

**Embedded MultiMediaCard (e•MMC) e•MMC/Card Product Standard,
High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data
Rate, Multiple Partitions Supports and Security Enhancement**

CONTENTS(*continued*)

Table A.1 — Mechanical characteristics	187
Table A.2 — Electrical characteristics.....	187
Table A.3 — Climatic characteristics	187
Table A.4 — Macro commands	188
Table A.5 — Forward-compatible host interface timing	202
Table A.6 — XNOR values	207

Embedded MultiMediaCard (e•MMC) e•MMC/Card Product Standard, High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data Rate, Multiple Partitions Supports and Security Enhancement

CONTENTS(*continued*)

Figure 1 —	Topology of MultiMediaCard systems	11
Figure 2 —	MultiMediaCard system overview	12
Figure 3 —	MultiMediaCard system example	13
Figure 4 —	MultiMediaCard architecture	16
Figure 5 —	MultiMediaCard bus system	17
Figure 6 —	Sequential read operation	18
Figure 7 —	Multiple-block read operation	19
Figure 8 —	Sequential write operation	19
Figure 9 —	(Multiple) Block write operation	19
Figure 10 —	No response” and “no data” operations	20
Figure 11 —	Command token format	20
Figure 12 —	Response token format	20
Figure 13 —	Data packet format for SDR	21
Figure 14 —	Data packet format for DDR	22
Figure 15 —	MultiMediaCard controller scheme	23
Figure 16 —	MultiMediaCard adaptor architecture	24
Figure 17 —	e•MMC memory organization at time zero	29
Figure 18 —	Example of partitions and user data area configuration	30
Figure 19 —	Flow Chart for General Purpose Partitions & Enhanced User Data Area parameter setting	32
Figure 20 —	WP condition transition due to H/W reset assertion	34
Figure 21 —	RST_n signal at the power up period	34
Figure 22 —	Memory partition	35
Figure 23 —	MultiMediaCard state diagram (boot mode)	36
Figure 24 —	MultiMediaCard state diagram (alternative boot mode)	37
Figure 25 —	MultiMediaCard state diagram (boot mode)	38
Figure 26 —	MultiMediaCard state diagram (card identification mode)	41
Figure 27 —	MultiMediaCard state transition diagram, interrupt mode	44
Figure 28 —	MultiMediaCard state diagram (data transfer mode)	45
Figure 29 —	Memory array partitioning	97
Figure 30 —	Identification timing (card identification mode)	98
Figure 31 —	SET_RCA timing (card identification mode)	98
Figure 32 —	Command response timing (data transfer mode)	98
Figure 33 —	R1b response timing	99
Figure 34 —	Timing response end to next command start (data transfer mode)	99
Figure 35 —	Timing of command sequences (all modes)	99
Figure 36 —	Single-block read timing	100
Figure 37 —	Multiple-block read timing	100
Figure 38 —	Stop command timing (CMD12, data transfer mode)	100

Embedded MultiMediaCard (e•MMC) e•MMC/Card Product Standard, High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data Rate, Multiple Partitions Supports and Security Enhancement

CONTENTS(*continued*)

Figure 39 —	Block write command timing.....	101
Figure 40 —	Multiple-block write timing	102
Figure 41 —	Stop transmission during data transfer from the host.....	102
Figure 42 —	Stop transmission during CRC status transfer from the card	102
Figure 43 —	Stop transmission after last data block; card is busy programming.....	103
Figure 44 —	Stop transmission after last data block; card becomes busy	103
Figure 45 —	Bus test procedure timing.....	104
Figure 46 —	Boot operation, termination between consecutive data blocks	104
Figure 47 —	Boot operation, termination during transfer.....	105
Figure 48 —	Bus mode change timing (push-pull to open-drain).....	105
Figure 49 —	Alternative boot operation, termination between consecutive data blocks	106
Figure 50 —	Alternative boot operation, termination during transfer.....	106
Figure 51 —	H/W reset waveform	108
Figure 52 —	Noise filtering timing for H/W reset	108
Figure 53 —	CRC7 generator/checker	154
Figure 54 —	CRC16 generator/checker	155
Figure 55 —	Bus circuitry diagram.....	159
Figure 56 —	Improper power supply	160
Figure 57 —	Shortcut protection	160
Figure 58 —	Power-up diagram	161
Figure 59 —	e•MMC power-up diagram	163
Figure 60 —	The e•MMC power cycle	164
Figure 61 —	MultiMediaCard bus driver.....	166
Figure 62 —	e•MMC internal power diagram.....	168
Figure 63 —	Bus signal levels.....	170
Figure 64 —	Timing diagram: data input/output.....	172
Figure 65 —	Timing diagram: data input/output in dual data rate mode	175
Figure A.1 —	Power supply decoupling	185
Figure A.2 —	Modified MultiMediaCard connector for hot insertion.....	186
Figure A.3 —	Legend for command-sequence flow charts.....	189
Figure A.4 —	SEND_OP_COND command flow chart.....	190
Figure A.5 —	CIM_SINGLE_CARD_ACQ.....	191
Figure A.6 —	CIM_SETUP_CARD	192
Figure A.7 —	CIM_STREAM_READ	193
Figure A.8 —	CIM_READ_BLOCK.....	193
Figure A.9 —	CIM_READ_MBLOCK	194
Figure A.10 —	CIM_WRITE_MBLOCK.....	195
Figure A.11 —	CIM_ERASE_GROUP	196
Figure A.12 —	CIM_SECURE_ERASE.....	197

**Embedded MultiMediaCard (e•MMC) e•MMC/Card Product Standard,
High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data
Rate, Multiple Partitions Supports and Security Enhancement**

CONTENTS(*continued*)

Figure A.13 —CIM_SECURE_TRIM	198
Figure A.14 —CIM_TRIM.....	199
Figure A.15 —CIM_US_PWR_WP.....	200
Figure A.16 —CIM_US_PERM_WP.....	201
Figure A.17 —Bus testing for eight data lines	206
Figure A.18 —Bus testing for four data lines.....	206
Figure A.19 —Bus testing for one data line	206
Figure A.20 —Erase-unit size selection flow	208

**Embedded MultiMediaCard (e•MMC) e•MMC/Card Product Standard,
High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data
Rate, Multiple Partitions Supports and Security Enhancement**

CONTENTS(*continued*)

This page intentionally left blank.

Foreword

This standard has been prepared by JEDEC and the MultiMediaCard Association, hereafter referred to as MMCA.

JEDEC has taken the basic MMCA specification and adopted it for embedded applications, calling it “*e*•MMC.” In addition to the packaging differences, *e*•MMC devices use a reduced-voltage interface.

The purpose of this specification is the definition of the MMC/*e*•MMC Electrical Interface, its environment and handling. It provides guidelines for systems designers. The specification also defines a tool box (a set of macro functions and algorithms) that contributes to reducing design-in costs.

The SPI mode is obsolete in this version.

Introduction

The MMC/*e*•MMC is an universal low cost data storage and communication media. It is designed to cover a wide area of applications as smart phones, cameras, organizers, PDAs, digital recorders, MP3 players, pagers, electronic toys, etc. Targeted features are high mobility and high performance at a low cost price. These features include low power consumption and high data throughput at the memory card interface.

MMC/*e*•MMC communication is based on an advanced 10-signal bus. The communication protocol is defined as a part of this standard and referred to as the MultiMediaCard mode.

To provide for the forecasted migration of CMOS power (V_{DD}) requirements and for compatibility and integrity of MultiMediaCard systems, two types of MultiMediaCards are defined in this standard specification, which differ only in the valid range of system V_{DD} . These two card types are referred to as High Voltage MultiMediaCard and Dual Voltage MultiMediaCard.

Embedded MultiMediaCard (*e*•MMC) *e*•MMC/Card Product Standard, High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data Rate, Multiple Partitions supports, and Security enhancement

(From BoD ballot, JCB-09-03, formulated under the cognizance of the JC-64 committee on Flash Memory Modules)

1 Scope

This document provides a comprehensive definition of the MMC/*e*•MMC Electrical Interface, its environment, and handling. It also provides design guidelines and defines a tool box of macro functions and algorithms intended to reduce design-in costs.

2 Normative reference

The following normative documents contain provisions that, through reference in this text, constitute provisions of this standard. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies.

3 Terms and definitions

For the purposes of this publication, the following abbreviations for common terms apply:

Block a number of bytes, basic data transfer unit

Broadcast a command sent to all cards on the MultiMediaCard bus

NOTE Broadcast occurs only in MultiMediaCard systems supporting versions prior to 4.0. In version 4.0 and later only one card can be present on the bus

CID Card IDentification number register

CLK clock signal

CMD command line or MultiMediaCard bus command (if extended CMDXX)

CRC Cyclic Redundancy Check

CSD Card Specific Data register

3 Terms and definitions (cont'd)

Copies	Copies of erase group(s) or copies of write groups shall be defined as copies of data that are generated by the device controller during internal device controller operations. These can include (but are not limited to) copies generated during error handling, wear-leveling or garbage collection. Copies does not refer to write block data, at a specific address. This overwritten data may still remain in the memory array but is no longer accessible by the host. If this data must be secure trimmed, it is the host application's responsibility to mark this data for secure trim prior to the overwrite event.
DAT	data line
DSR	Driver Stage Register
eMMC	embedded MultiMediaCard
ERASE	Block erase operation which does not require actual physical NAND erase operation
Flash	a type of multiple time programmable non volatile memory
Group	a number of write blocks, composite erase and write protect unit
LOW, HIGH	binary interface states with defined assignment to a voltage level
NSAC	defines the worst case for the clock rate dependent factor of the data access time
MSB, LSB	the Most Significant Bit or Least Significant Bit
OCR	Operation Conditions Register
open-drain	a logical interface operation mode. An external resistor or current source is used to pull the interface level to HIGH, the internal transistor pushes it to LOW
payload	net data
push-pull	a logical interface operation mode, a complementary pair of transistors is used to push the interface level to HIGH or LOW
RCA	Relative Card Address register
Reset	CMD0 with argument of 0x00000000 or 0xF0F0F0F0, H/W reset (or CMD15)
ROM	Read Only Memory
RPMB	Replay Protected Memory Block
Secure Purge	The process of overwriting all the addressable locations within an identified range with a single character and then performing an erase on those same locations. One or multiple write blocks or write protect groups depending on context.
NOTE The definition of secure purge is technology dependent (the definition above assumes NAND flash). Please refer to the http://www.killdisk.com/dod.htm or the following documents for more details. DoD 5220.22M(http://www.dtic.mil/whs/directives/corres/html/522022m.htm) and NIST SP 800-88 (http://csrc.nist.gov/publications/nistpubs/800-88/NISTSP800-88_rev1.pdf)	
stuff bit	filling 0 bits to ensure fixed length frames for commands and responses
SPI	Serial Peripheral Interface
TAAC	defines the time dependent factor of the data access time
three-state driver	a driver stage which has three output driver states: HIGH, LOW and high impedance (which means that the interface does not have any influence on the interface level)
token	code word representing a command
V_{DD}	+ Supply voltage (Card)

3 Terms and definitions (continued)

V_{SS} + Supply voltage ground for Core (BGA)

V_{SS1} + Supply voltage ground (card)

V_{SS2} + Supply voltage ground (card)

V_{CC} + Supply voltage for Core (BGA)

V_{CCQ} + Supply voltage for I/O (BGA)

V_{SSQ} + Supply voltage ground for I/O (BGA)

Write Protectin

- **Permanent** - Write and erase prevention scheme, which once enabled, cannot be reversed
- **Power-on** - Write and erase prevention scheme, which once enabled, cannot be reversed until a power failure event that causes the device to reboot occurs or the device is reset using the reset pin.
- **Temporary** - Write and erase prevention scheme that can be enabled and disabled.

This page intentionally left blank.

4 General description

The MultiMediaCard is an universal low cost data storage and communication media. It is designed to cover a wide area of applications as smart phones, cameras, organizers, PDAs, digital recorders, MP3 players, pagers, electronic toys, etc. Targeted features are high mobility and high performance at a low cost price. These features include low power consumption and high data throughput at the memory card interface.

The MultiMediaCard communication is based on an advanced 13-pin bus. The communication protocol is defined as a part of this standard and referred to as the MultiMediaCard mode.

To provide for the forecasted migration of CMOS power (V_{DD}) requirements and for compatibility and integrity of MultiMediaCard systems, two types of MultiMediaCards are defined in this standard specification, which differ only in the valid range of system V_{DD} . These two card types are referred to as High Voltage MultiMediaCard and Dual Voltage MultiMediaCard.

The purpose of the system specification is the definition of the MultiMediaCard, its environment and handling. It gives guidelines for a system designer. The system specification also defines a tool box (a set of macro functions and algorithms) which contributes to reducing the design-in costs.

The document is split up into several portions. The MultimediaCard Features are described in [Section 5](#).

[Section 6](#) gives a general overview of the system components: card, bus, and host.

The common MultiMediaCard characteristics are described in [Section 7](#). As this description defines an overall set of card properties, you should work with the vendor-specific, product documentation in parallel.

[Section 8](#) describes the card registers.

The SPI mode is removed from this standard.

All error protection techniques employed in this standard are described in [Section 10](#).

[Section 11](#) describes the physical and mechanical properties of the cards and the minimal requirements of the card slots and cartridges.

[Section 12](#) defines the MultiMediaCard bus as a universal communication interface and the electrical parameters of the interface.

The standard compliance criteria for the cards and hosts are described in [Section 13](#).

For achieving high data interchangeability, three basic file formats are defined in [Section 14](#) as valid file formats for the MultiMediaCard

[Annex A](#) contains additional information that is informative in nature and not considered a constituent part of this specification. These Application Notes contain useful hints for the circuit and system designers, helping simplify the design process.

[Annex B](#) lists the major changes between the previous and the current version of this specification.

As used in this document, “shall” or “will” denotes a mandatory provision of the standard. “Should” denotes a provision that is recommended but not mandatory. “May” denotes a feature whose presence does not preclude compliance, that may or may not be present at the option of the implementor.

5 System features

The MultiMediaCard System has a wide variety of system features, whose comprehensive elements serves several purposes, which include:

- Covering a broad category of applications from smart phones and PDAs to digital recorders and toys
- Facilitating the work of designers who seek to develop applications with their own advanced and enhanced features
- Maintaining compatibility and compliance with current electronic, communication, data and error handling standards.

The following list identifies the main features of the MultiMediaCard System, which:

- Is targeted for portable and stationary applications
- Has these System Voltage (V_{DD}) Ranges:

Table 1 — MultiMediaCard Voltage Modes

	High Voltage MultiMediaCard	Dual Voltage MultiMediaCard
Communication	2.7 - 3.6	1.70 - 1.95, 2.7 - 3.6¹
Memory Access	2.7 - 3.6	1.70 - 1.95, 2.7 - 3.6
NOTE 1 V_{DD} range: 1.95V - 2.7V is not supported.		

- Includes MMCplus and MMCmobile definitions
- Is designed for read-only, read/write and I/O cards
- Supports card clock frequencies of 0-20MHz, 0-26MHz or 0-52MHz
- Has a maximum data rate up to 832Mbits/sec.
- Has a defined minimum performance
- Maintains card support for three different data bus width modes: 1-bit (default), 4-bit, and 8-bit
- Includes definition for higher than 2GB of density of memories
- Includes password protection of data
- Supports basic file formats for high data interchangeability
- Includes application specific commands
- Enables correction of memory field errors
- Has built-in write protection features for the boot and user areas, which may be permanent, power-on, or temporary
- Includes a simple erase mechanism
- Maintains full backward compatibility with previous MultiMediaCard systems (1 bit data

bus, multi-card systems)

- Ensures that new hosts retain full compatibility with previous versions of MultiMediaCards (backward compatibility).
- Supports two form factors: Normal size (24mm x 32mm x 1.4mm) and reduced size (24mm x 18mm x 1.4mm)
- Supports multiple command sets
- Includes attributes of the available operation modes:

Table 2 — MMC System Operational Mode

MultiMediaCard Mode
Ten-wire bus (clock, 1 bit command, 8 bit data bus)
Card selection is done through an assigned unique card address to maintain backwards compatibility to prior versions of the specification
One card per MultiMediaCard bus
Easy identification and assignment of session address
Error-protected data transfer
Sequential and Single/Multiple block Read/Write commands

- Provides a possibility for the host to make sudden power failure safe-update operations for the data content.
- Enhanced power saving method by introducing a sleep functionality.
- Introduces Boot Operation Mode to provide a simple boot sequence method.
- Provides a new CID Register setting to recognize either *e*-MMC or a card.
- Obsoletes the SPI Mode.
- Defines I/O voltage (V_{CCQ}) and core voltage (V_{CC}) separately for *e*-MMC.
- Includes *e*-MMC BGA Form Factors:
 - 11.5mm x 13mm x 1.3mm
 - 12mm x 16mm x 1.4mm
 - 12mm x 18mm x 1.4mm
- Defines Erase-unit size and Erase timeout for high-capacity memory.
- Provides access size register indicating one (or multiple) programmable boundary unit(s) of device.
- Obsoletes the Absolute Minimum Performance.
- Introduces *e*-MMC OCR setting and response.
- Defines WP group size for high-capacity devices.
- Introduces Alternate Boot Operation Mode.
- Introduces Secure Erase & Trim to enhance data security.

- Supports Multiple User Data Partition with Enhanced User Data Area options
- Signed access to a Replay Protected Memory Block.
- Introduces dual data rate transfer.
- Introduces high speed boot.
- Enhanced Write Protection with Permanent and Partial protection options.
- Introduces hardware reset signal.

6 MultiMediaCard system concept

The main design goal of the MultiMediaCard system is to provide a very low cost mass storage product, implemented as a 'card' with a simple controlling unit, and a compact, easy-to-implement interface. These requirements lead to a reduction of the functionality of each card to an absolute minimum.

Nevertheless, since the complete MultiMediaCard system has to have the functionality to execute tasks (at least for the high end applications), such as error correction and standard bus connectivity, the system concept is described next. It is based on modularity and the capability of reusing hardware over a large variety of cards.

Figure 1 shows four typical architectures of possible MultiMediaCard systems.

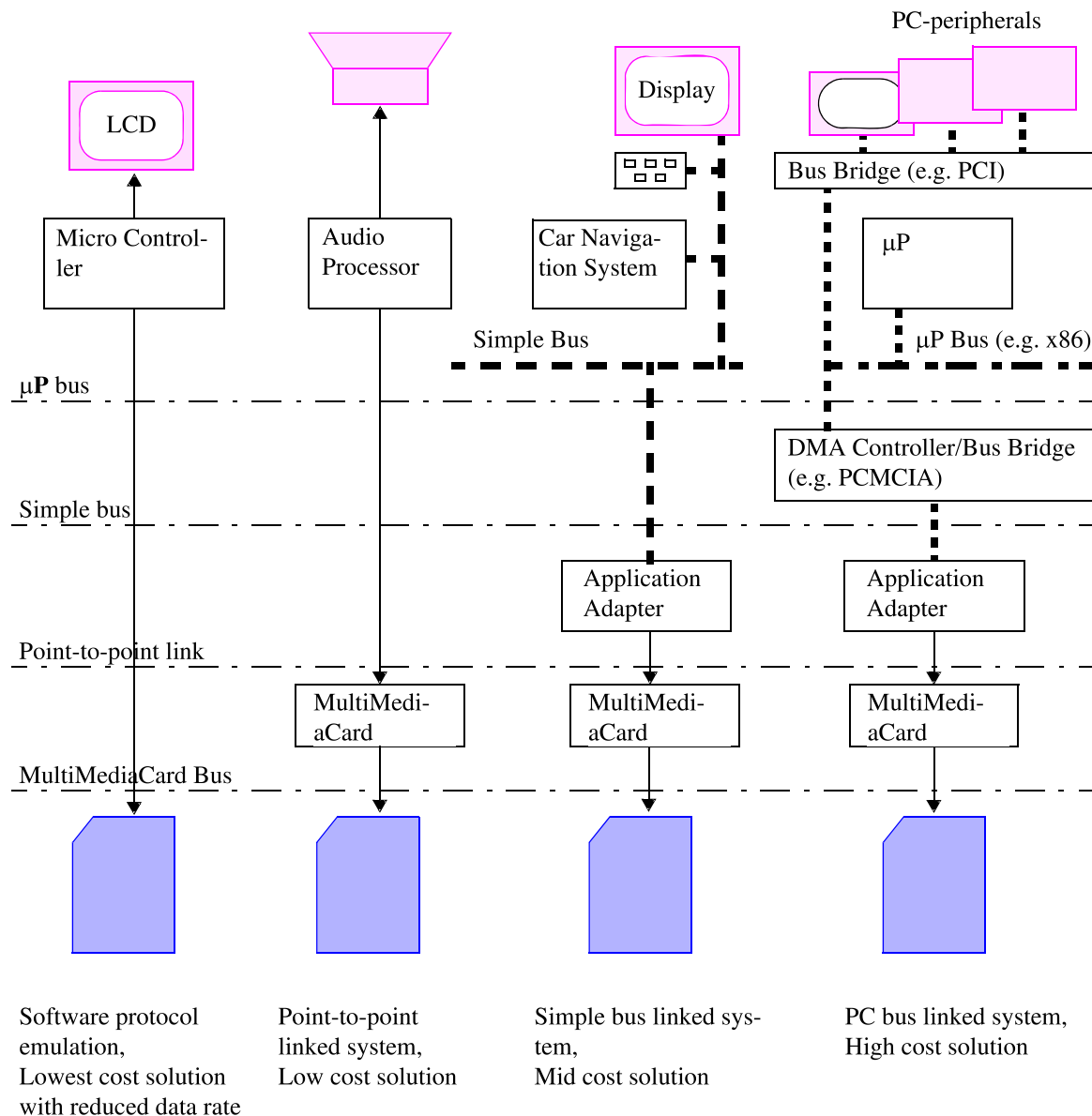


Figure 1 — Topology of MultiMediaCard systems

Four typical types of MultiMediaCard systems can be derived from the diagram shown in [Figure 2](#). The typical systems include:

- Software emulation: reduced data rate, typically 100-300 kbit per second, restricted by the host
- Point to point linkage: full data rate (with additional hardware)
- Simple bus: full data rate, part of a set of addressable units
- PC bus: full data rate, addressable, extended functionality, such as DMA capabilities

In the first variant, the MultiMediaCard bus protocol is emulated in software using up to ten port pins of a microcontroller. This solution requires no additional hardware and is the cheapest system in the list. The other applications extend the features and requirements, step by step, towards a sophisticated PC solution. The various systems, although different in their feature set, have a basic common functionality, as can be seen in Figure 2. This diagram shows a system partitioned into hierarchical layers of abstract ('virtual') components. It describes a logical classification of functions which cover a wide variety of implementations. (See also [Figure 1 on page 11](#).) It does not imply any specific design nor specify rules for implementing parts in hardware or software.

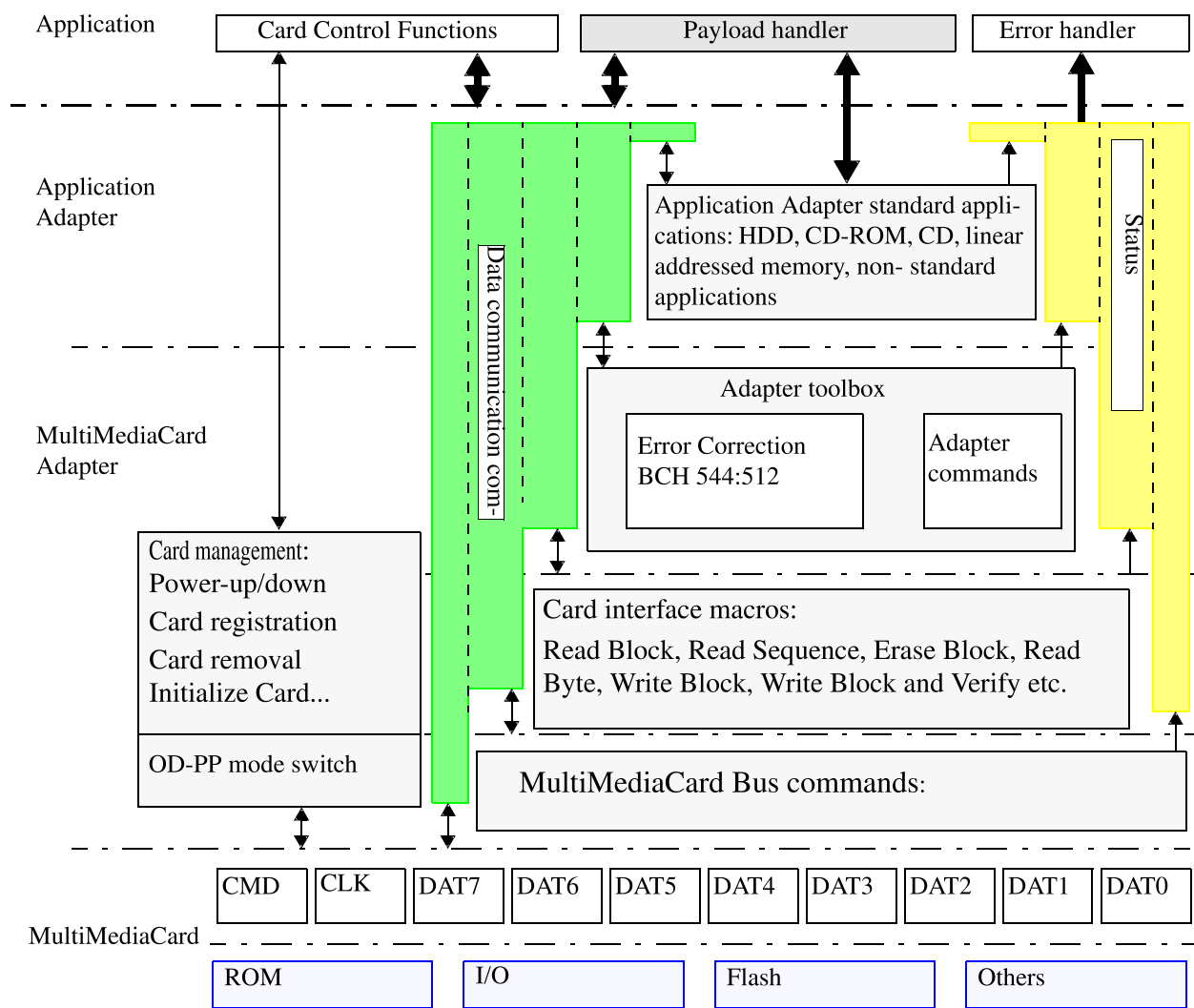


Figure 2 — MultiMediaCard system overview

Figure 3 is a specific design example based on the abstract layer model described in Figure 2 on page 12.

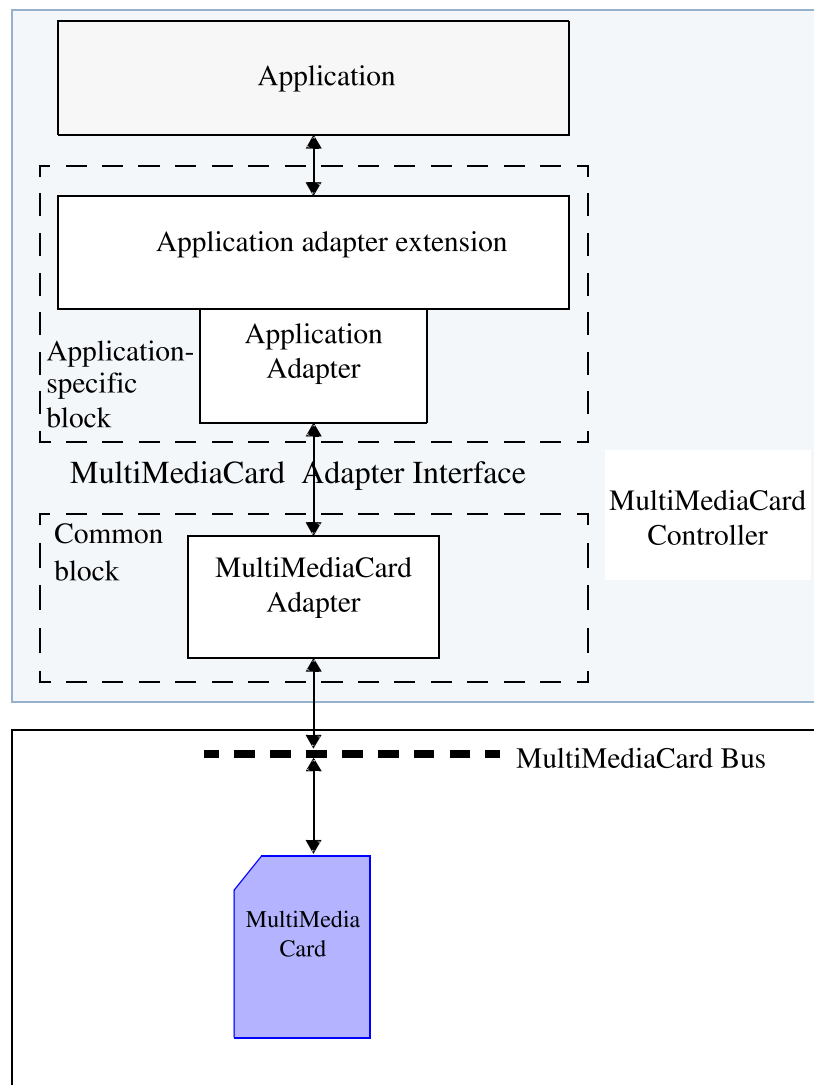


Figure 3 — MultiMediaCard system example

This MultiMediaCard system contains at least two components:

- The MultiMediaCard
- The MultiMediaCard controller

The MultiMediaCard controller is divided into two major blocks. In some implementations like the example shown in Figure 3, the controller may implement the whole application, while in others it may be divided into several physical components which, apart from the application itself, can be identified as:

- 1: Application adapter — the application specific block, for example, a microprocessor or an adapter to a standard bus like USB or ATA
- Performs application oriented tasks, e.g., display controlling or input decoding for hand-held applications

- Typically connected as a bus slave for a standard bus

2: MultiMediaCard adapter — the common block

- Contains all card specific functions, such as initialization and error correction
- Serves as a bus master for the MultiMediaCard bus
- Implements the standard interface to the card.

6.1 Higher than a density of 2GB

The maximum density possible to be implemented according to the versions up to v4.1 of this document was limited in practise to 2GB. This was due to the following reasons:

- Existed 32bit byte-address argument in the command frame (max 4GB could be addressed)
- Existed formula according to which to calculate the density of a card (max 4GB could be indicated)
- Capability of the FAT16 File System to address up to 2GB of address space per one partition

The lowest common nominator, 2GB in this case, will set the limit. The implementation of a higher than 2GB of density of memory will not be backwards compatible with the lower densities. First of all the address argument for higher than 2GB of density of memory is changed to be sector address (512B sectors) instead of byte address. Secondly the density of the card is read from the EXT_CSD register instead of CSD register. And finally the system implementation needs to include a File System capable of handling sector type of addresses.

6.2 MMC*plus* and MMC*mobile*

The specification further defines two card types, MMC*plus* and MMC*mobile*, to describe R/W or ROM cards with specifically defined mandatory features and attributes. Only cards meeting MMC*plus* or MMC*mobile* requirements are eligible to carry the MMC*plus* or MMC*mobile* name and logo.

- MMC*plus* is defined as normal size R/W or ROM cards that supports 2.7-3.6V operation, x1/x4/x8 bus widths.
- MMC*mobile* is defined as reduced size R/W or ROM card that supports 1.70-1.95V and 2.7-3.6V operations, x1/x4/x8 bus widths, minimum of 2.4MB/s read/write performance.

Both implementations are backwards compatible with MMCA System Specification versions 3.xx in max 20MHz clock frequency mode.

6.3 Card concept

The MultiMediaCard transfers data via a configurable number of data bus signals. The communication signals are:

- **CLK**: Each cycle of this signal directs a one bit transfer on the command and either a one bit (1x) or a two bits transfer (2x) on all the data lines. The frequency may vary between zero and the maximum clock frequency.
- **CMD**: This signal is a bidirectional command channel used for card initialization and transfer of commands. The CMD signal has two operation modes: open-drain for initialization mode, and push-pull for fast command transfer. Commands are sent from the MultiMediaCard bus master to the card and responses are sent from the card to the host.
- **DAT0-DAT7**: These are bidirectional data channels. The DAT signals operate in push-pull mode. Only the card or the host is driving these signals at a time. By default, after power up or reset, only DAT0 is

used for data transfer. A wider data bus can be configured for data transfer, using either DAT0-DAT3 or DAT0-DAT7, by the MultiMediaCard controller. The MultiMediaCard includes internal pull-ups for data lines DAT1-DAT7. Immediately after entering the 4-bit mode, the card disconnects the internal pull ups of lines DAT1, DAT2, and DAT3. Correspondingly, immediately after entering to the 8-bit mode the card disconnects the internal pull-ups of lines DAT1–DAT7.

- MultiMediaCards can be grouped into several card classes which differ in the functions they provide (given by the subset of MultiMediaCard system commands):
- Read Only Memory (ROM) cards. These cards are manufactured with a fixed data content. They are typically used as a distribution media for software, audio, video etc.
- Read/Write (RW) cards (Flash, One Time Programmable - OTP, Multiple Time Programmable - MTP). These cards are typically sold as blank (empty) media and are used for mass data storage, end user recording of video, audio or digital images.
- I/O cards. These cards are intended for communication (e.g. modems) and typically will have an additional interface link.

The card is connected directly to the signals of the MultiMediaCard bus. The following table defines the card contacts:

Table 3 — MultiMediaCard interface pin configuration

Name	Type ¹	Description
CLK	I	Clock
DAT0 ²	I/O/PP	Data
DAT1	I/O/PP	Data
DAT2	I/O/PP	Data
DAT3	I/O/PP	Data
DAT4	I/O/PP	Data
DAT5	I/O/PP	Data
DAT6	I/O/PP	Data
DAT7	I/O/PP	Data
CMD	I/O/PP/OD	Command/Response
RST _n	I	Hardware reset
V _{CC}	S	Supply voltage for Core (BGA)
V _{CCQ}	S	Supply voltage for I/O (BGA)
V _{DD}	S	Supply voltage (card)
V _{SS}	S	Supply voltage ground for Core (BGA)
V _{SS1}	S	Supply voltage ground (card)
V _{SS2}	S	Supply voltage ground (card)
V _{SSQ}	S	Supply voltage ground for I/O (BGA)

NOTE 1 I: input; O: output; PP: push-pull; OD: open-drain; NC: Not connected (or logical high); S: power supply.
 NOTE 2 The DAT0–DAT7 lines for read-only cards are output only.

The card initialization uses only the CMD channel and is, therefore, compatible for all cards.

Each card has a set of information registers (see also [Section 8 on page 111](#)):

Table 4 — MultiMediaCard registers

Name	Width (bytes)	Description	Implementation
CID	16	Card IDentification number, a card individual number for identification.	Mandatory
RCA	2	Relative Card Address, is the card system address, dynamically assigned by the host during initialization.	Mandatory
DSR	2	Driver Stage Register, to configure the card’s output drivers.	Optional
CSD	16	Card Specific Data, information about the card operation conditions.	Mandatory
OCR	4	Operation Conditions Register. Used by a special broadcast command to identify the voltage type of the card.	Mandatory
EXT_CSD	512	Extended Card Specific Data. Contains information about the card capabilities and selected modes. Introduced in specification v4.0	Mandatory

The host may reset the card by switching the power supply off and back on. The card shall have its own power-on detection circuitry which puts the card into a defined state after the power-on. For MMC card, no explicit reset signal is necessary. However, for *e*-MMC, there is a reset signal which host can use to reset *e*-MMC device.. The MMC card and *e*-MMC can also be reset by a special command.

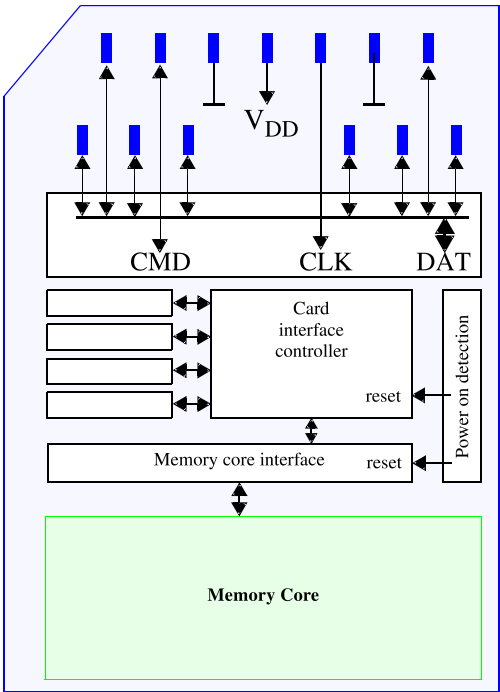


Figure 4 — MultiMediaCard architecture

6.3.1 Form factors

See chapter 11 "MultiMediaCard mechanical specification" for form factor details.

6.4 Bus concept

The MultiMediaCard bus is designed to connect either solid-state mass-storage memory or I/O-devices in a card format to multimedia applications. The bus implementation allows the coverage of application fields from low-cost systems to systems with a fast data transfer rate. It is a single master bus with a single slave. The MultiMediaCard bus master is the bus controller and the slave is either a single mass storage card (with possibly different technologies such as ROM, OTP, Flash etc.) or an I/O-card with its own controlling unit (on card) to perform the data transfer.

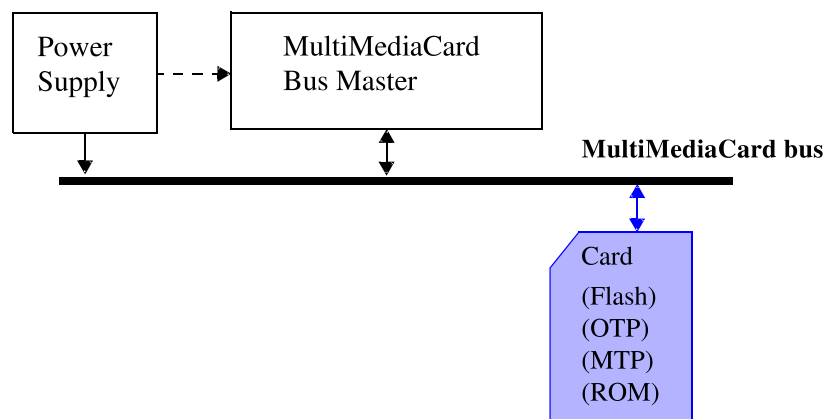


Figure 5 — MultiMediaCard bus system

The MultiMediaCard bus also includes power connections to supply the cards.

The bus communication uses a special protocol (MultiMediaCard bus protocol). The payload data transfer between the host and the card can be bidirectional.

6.4.1 Bus lines

The bus lines can be divided into three groups:

- Power supply: V_{SS1} and V_{SS2} , V_{DD} - used to supply the cards.
 V_{SS} , V_{SSQ} , V_{CC} , and V_{CCQ} - used to supply *e*-MMC.
- Data transfer: CMD, DAT0-DAT7 - used for bidirectional communication.
- Clock: CLK - used to synchronize data transfer across the bus.

The bus line definitions and the corresponding pad numbers are described in [Section 6.3](#).

6.4.2 Bus protocol

After a power-on reset, the host must initialize the card by a special message-based MultiMediaCard bus protocol. Each message is represented by one of the following tokens:

- **command:** a command is a token which starts an operation. A command is sent from the host to a card. A command is transferred serially on the CMD line.
- **response:** a response is a token which is sent from the card to the host as an answer to a previously received command. A response is transferred serially on the CMD line.
- **data:** data can be transferred from the card to the host or vice versa. Data is transferred via the data lines. The number of data lines used for the data transfer can be 1(DAT0), 4(DAT0-DAT3) or 8(DAT0-DAT7).
For each data lines, data can be transferred at the rate of one bit (single data rate) or two bits (dual data rate) per clock cycle.

Card addressing is implemented using a session address, assigned during the initialization phase, by the bus controller to the connected card. A card is identified by its CID number. This method requires the card to have a unique CID number. To ensure uniqueness of CIDs the CID register contains 24 bits (MID and OID fields—see [Section 8 starting on page 111](#)) which are defined by the MMCA/JEDEC. Every card manufacturer is required to apply for a unique MID (and optionally OID) number.

Command, response, and data block structures are described in [Section 7 starting on page 27](#).

MultiMediaCard bus data transfers are composed of these tokens. One data transfer is a *bus operation*. There are different types of operations. Addressed operations always contain a command and a response token. In addition, some operations have a data token, the others transfer their information directly within the command or response structure. In this case no data token is present in an operation. The bits on the DAT0-DAT7 and CMD lines are transferred synchronous to the host clock.

Two types of data transfer commands are defined:

- **Sequential commands¹:** These commands initiate a continuous data stream, they are terminated only when a stop command follows on the CMD line. This mode reduces the command overhead to an absolute minimum.
- **Block-oriented commands:** These commands send a data block succeeded by CRC bits. Both read and write operations allow either single or multiple block transmission. A multiple block transmission is terminated when a stop command follows on the CMD line similarly to the sequential read.

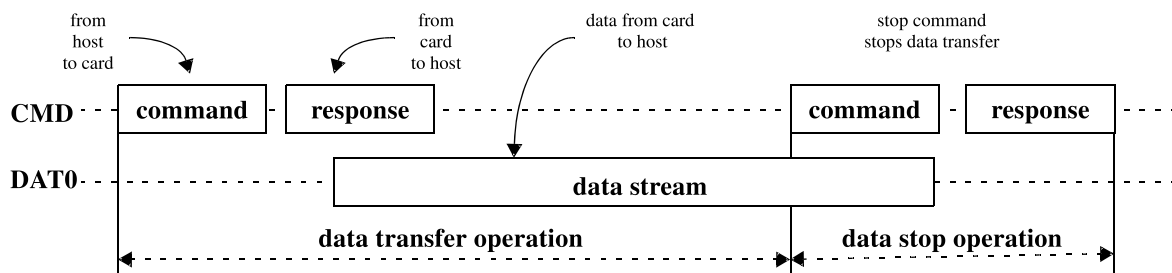


Figure 6 — Sequential read operation

1. Sequential commands are supported only in 1-bit bus mode, to maintain compatibility with previous versions of this specification

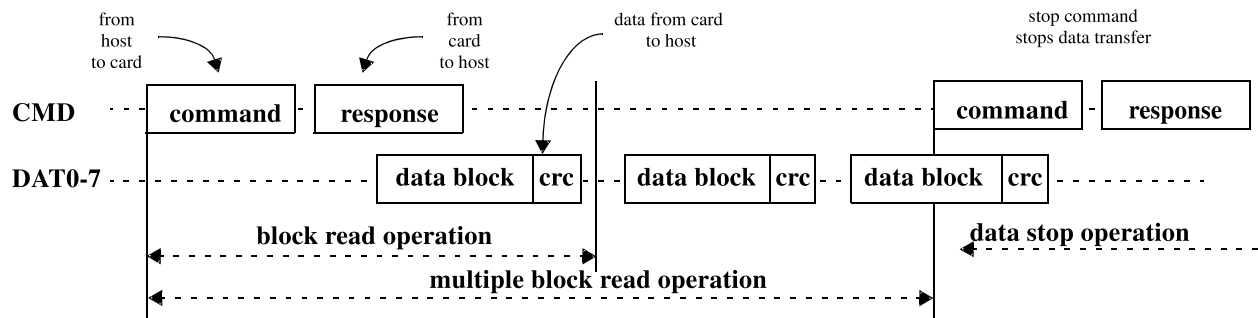


Figure 7 — Multiple-block read operation

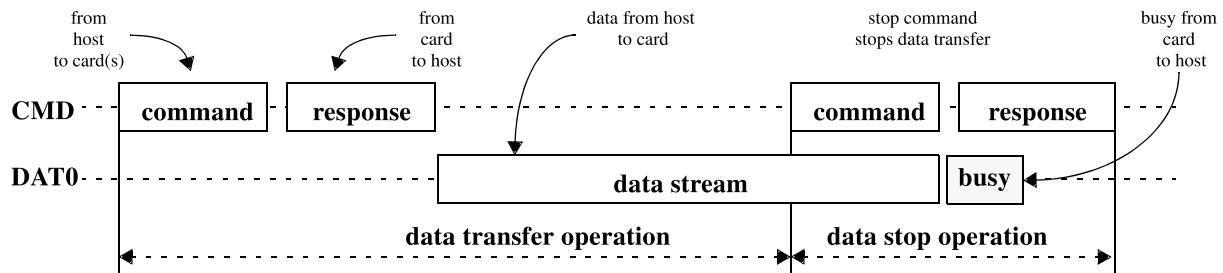


Figure 8 — Sequential write operation

The block write operation uses a simple busy signalling of the write operation duration on the data (DAT0) line. (See [Figure 9](#).)

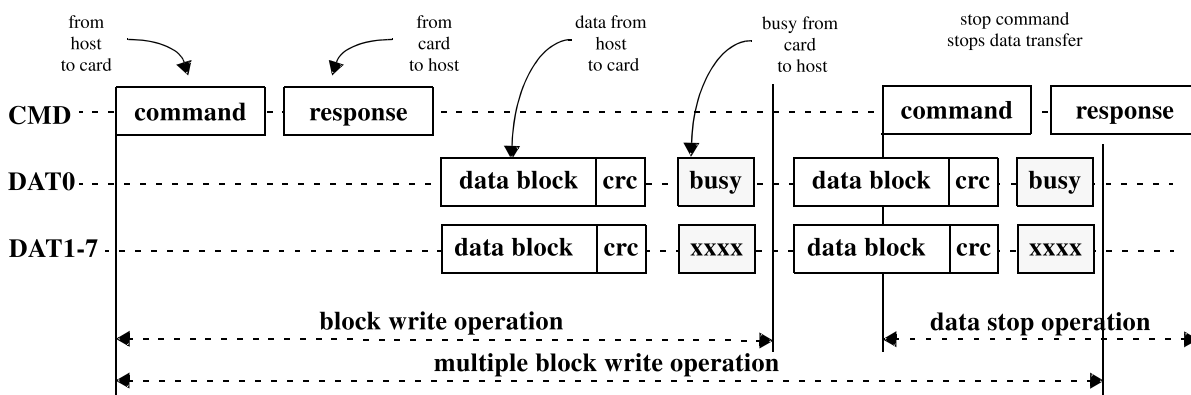


Figure 9 — (Multiple) Block write operation

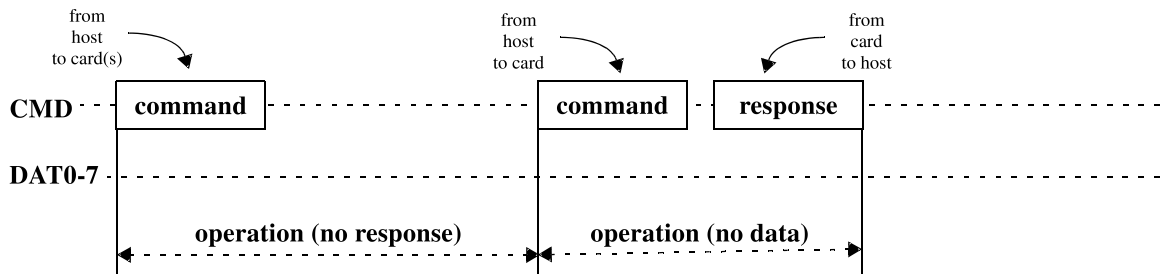


Figure 10 — “No response” and “no data” operations

Command tokens have the following coding scheme:

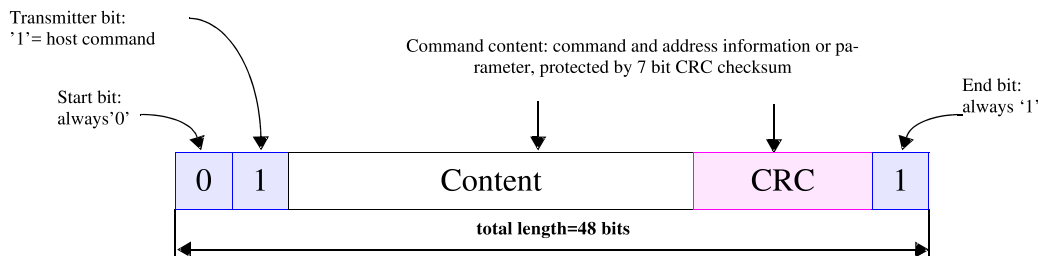


Figure 11 — Command token format

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

Response tokens have five coding schemes depending on their content. The token length is either 48 or 136 bits. The detailed command and response definitions are provided in [Section 7.10 on page 80](#) and [Section 7.12 on page 90](#).

Due to the fact that there is no predefined end in sequential data transfer, no CRC protection is included in this case. The CRC protection algorithm for block data is a 16 bit CCITT polynomial. All used CRC types are described in [Section 10 starting on page 153](#).

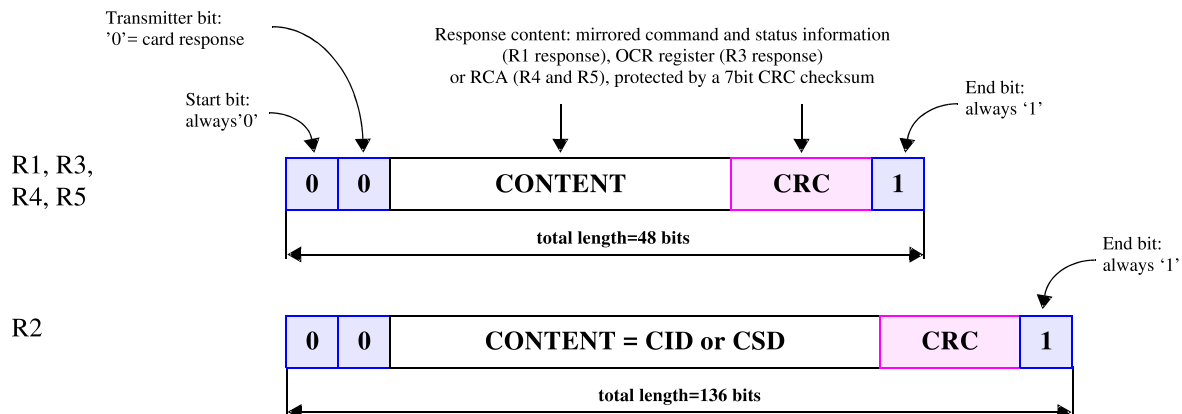


Figure 12 — Response token format

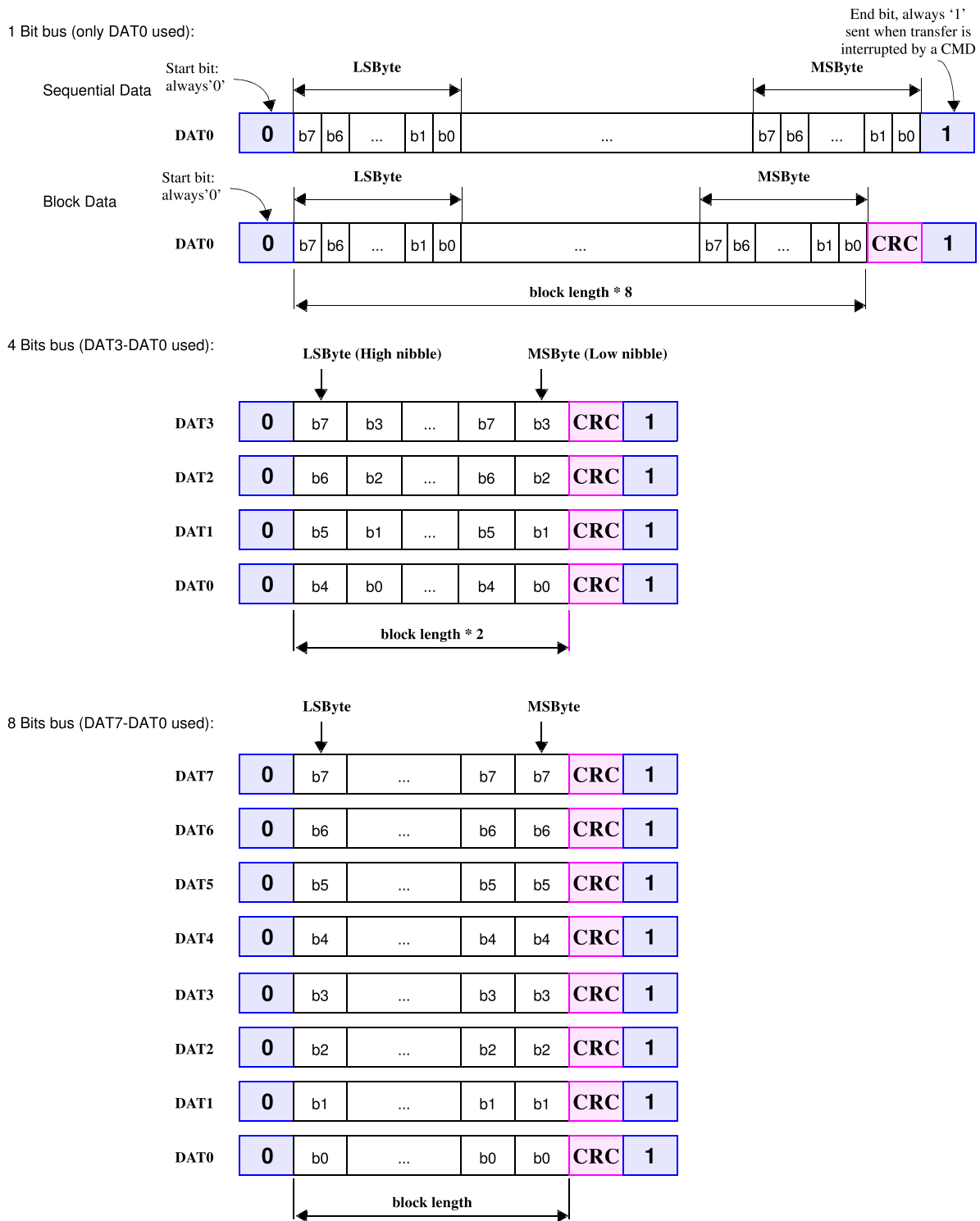
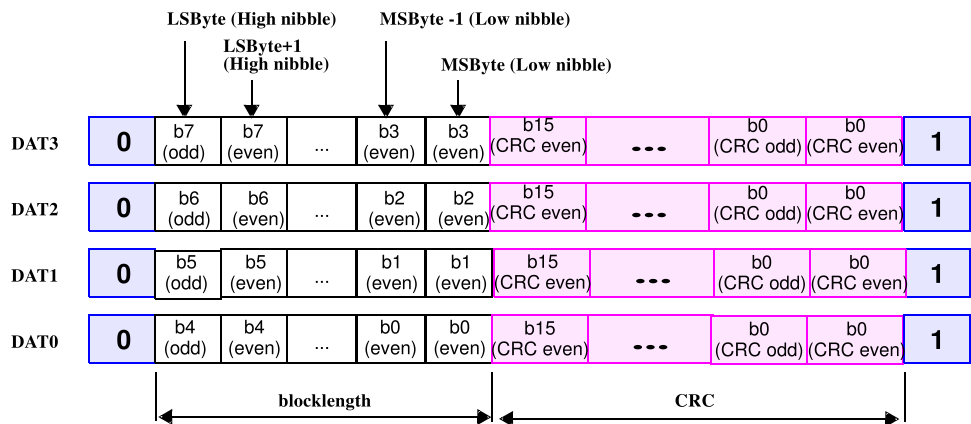
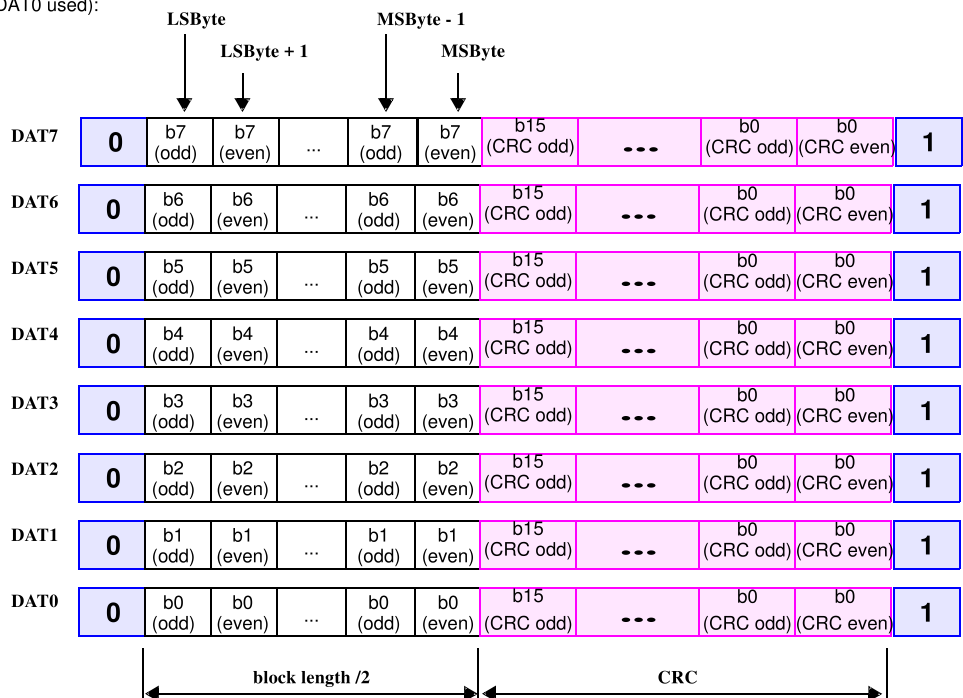


Figure 13 — Data packet format for SDR

4 Bits bus DDR (DAT3-DAT0 used):



8 Bits busDDR (DAT7-DAT0 used):



Notice that bytes DATA and the two CRC16 are interleaved.
Start and stop bits are full cycle.

Figure 14 — Data packet format for DDR

6.5 Controller Concept

The MultiMediaCard is defined as a low cost mass storage product. The shared functions have to be implemented in the MultiMediaCard system. The unit which contains these functions is called the MultiMediaCard controller. The following points are basic requirements for the controller:

- Protocol translation from standard MultiMediaCard bus to application bus
- Data buffering to enable minimal data access latency
- Macros for common complex command sequences

The MultiMediaCard controller is the link between the application and the MultiMediaCard bus with its card. It translates the protocol of the standard MultiMediaCard bus to the application bus. It is divided into two major parts:

- The application adapter: the application oriented part
- The MultiMediaCard adapter: the MultiMediaCard oriented part

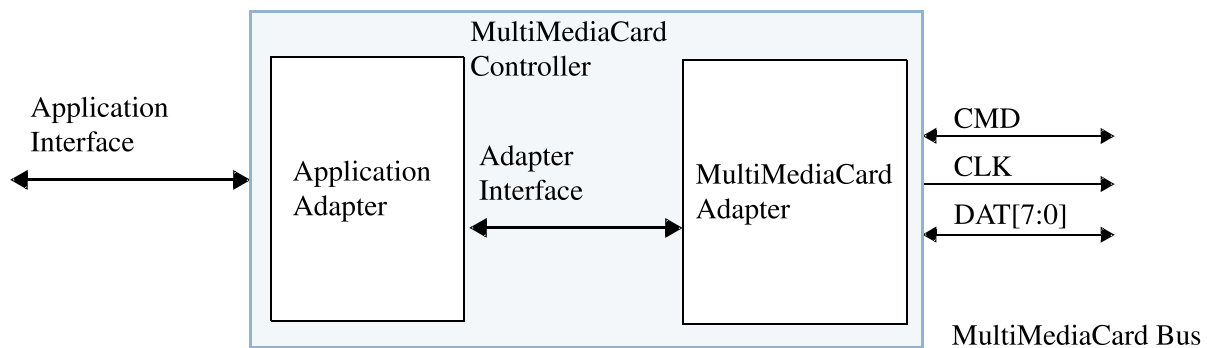


Figure 15 — MultiMediaCard controller scheme

The application adapter consists at least of a bus slave and a bridge into the MultiMediaCard system. It can be extended to become a master on the application bus and support functions like DMA or serve application specific needs. Higher integration will combine the MultiMediaCard controller with the application.

Independently of the type and requirements of the application the MultiMediaCard bus requires a host. This host may be the MultiMediaCard adapter. On the MultiMediaCard bus side it is the only bus master and controls all activity on that bus. On the other side, it is a slave to the application adapter or to the application, respectively. No application specific functions shall be supported here, except for those that are common to most MultiMediaCard systems. It supports all MultiMediaCard bus commands and provides additionally a set of macro commands. The adapter includes error correction capability for non error-free cards. The error correction codes used are defined in [Section 10.1 on page 153](#).

Because the application specific needs and the chosen application interface are out of the scope of this specification, the MultiMediaCard controller defines an internal adapter interface. The two parts communicate across this interface. The adapter interface is directly accessible in low cost (point to point link) systems where the MultiMediaCard controller is reduced to an MultiMediaCard adapter.

6.5.1 Application adapter requirements

The application adapter enhances the MultiMediaCard system in the way that it becomes plug&play in every standard bus environment. Each environment will need its unique application adapter. For some bus systems standard, off the shelf, application adapters exist and can interface with the MultiMediaCard adapter. To reduce the bill of material it is recommended to integrate an existing application adapter with the MultiMediaCard adapter module, to form a MultiMediaCard controller.

The application adapter extension is a functional enhancement of the application adapter from a bus slave to a bus master on the standard application bus. For instance, an extended application adapter can be triggered to perform bidirectional DMA transfers.

6.5.2 MultiMediaCard adapter architecture

The architecture and the functional units described below are not implementation requirements, but general recommendations on the implementation of a MultiMediaCard adapter. The adapter is divided into two major parts:

- The controller: macro unit and power management
- The data path: Adapter interface, ECC unit, read cache, write buffer, CRC unit and MultiMediaCard bus interface

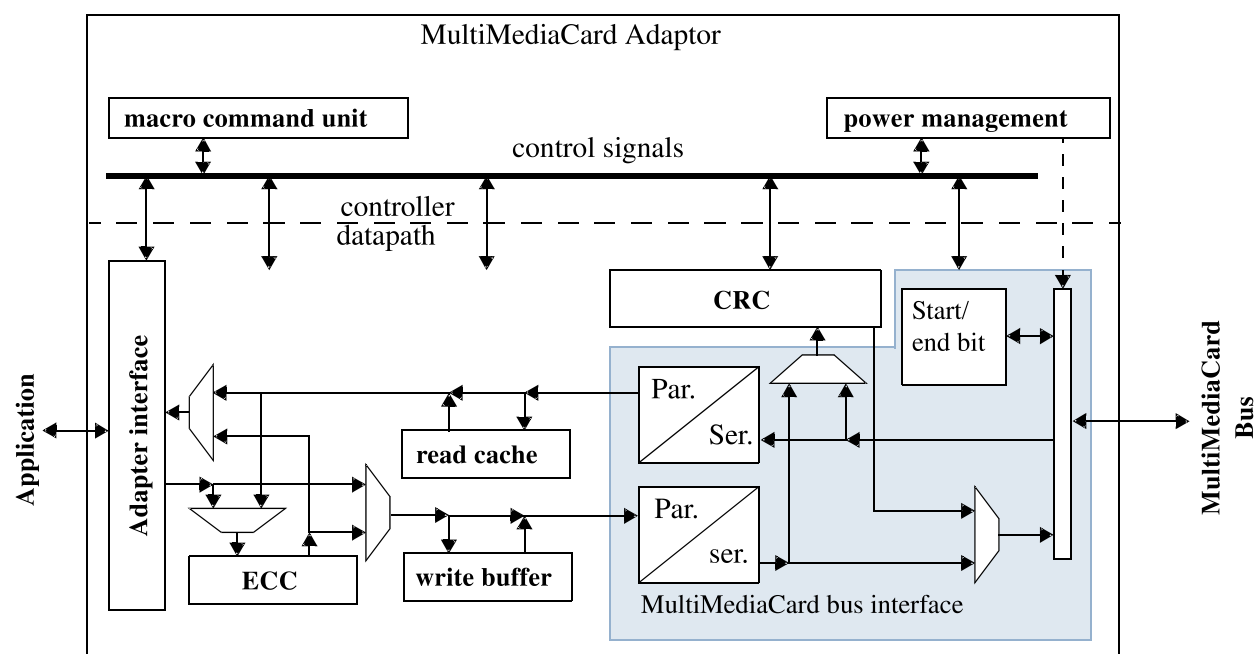


Figure 16 — MultiMediaCard adaptor architecture

The data path units should be implemented in hardware to guarantee the full capabilities of the MultiMediaCard system. The controller part of the adapter can be implemented in hardware or software depending on the application architecture.

The width of the data path should be a byte; the units which are handling data should work on bytes or blocks of bytes. This requirement is derived from the MultiMediaCard bus protocol, which is organized in data blocks. Blocks are multiples of bytes. Thus, the smallest unit of a data access or control unit is a byte.

Commands for the MultiMediaCard bus follow a strict protocol. Each command is encapsulated in a syntactical frame. Each frame contains some special control information like start/end bits and CRC protection. Some commands include stuffing bits to enable simple interpreters to use a fixed frame length. This transport management information should be generated in the MultiMediaCard adapter. These functions are combined in the MultiMediaCard bus interface of the adapter.

The response delays of the MultiMediaCard system may vary; they depend on the type of cards. So the adapter interface must handle asynchronous mode via handshake signals(STB,ACK) or the host has to poll the state (busy/not busy) if no handshake signals are required (synchronous mode). This interface may be a general unit supporting most application protocols or can be tailored to one application.

It is recommended to equip the MultiMediaCard adapter with data buffers for write and read operation. It will, in most cases, improve the system level performance on the application side. The MultiMediaCard bus transports its data with a data rate up to 832 Mbits/sec. This may be slower than a typical applications CPU bus. Enabling the CPU to off load the data to the buffers will free up CPU time for system level tasks, while the MultiMediaCard adapter handles the data transfer to the card.

The access time for random access read operations from a card may be improved by caching a block of data in the read cache. After reading a complete block into the MultiMediaCard adapter cache, repeated accesses to that block can be done very fast. Especially read-modify-write operations can be executed in a very efficient way on a block buffer with the help of the SRAM swapper.

7 MultiMediaCard functional description

In the following sections, the different card operation modes are described first. Thereafter, the restrictions for controlling the clock signal are defined. All MultiMediaCard commands together with the corresponding responses, state transitions, error conditions and timings are presented in the succeeding sections.

7.1 General

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

- Broadcast commands
Broadcast commands are intended for all cards in a MultiMediaCard system². Some of these commands require a response.
- Addressed (point-to-point) commands
The addressed commands are sent to the addressed card and cause a response from this card.

A general overview of the command flow is shown in [Figure 26 on page 41](#) for the card identification mode and in [Figure 28 on page 45](#) for the data transfer mode. The commands are listed in the command tables ([Table 21 on page 82](#) to [Table 29 on page 87](#)). The dependencies between current state, received command and following state are listed in [Table 30 on page 88](#). Five operation modes are defined for the MultiMediaCard system (hosts and cards):

- Boot mode
The card will be in boot mode after power cycle, reception of CMD0 with argument of 0xF0F0F0F0 or (*e*-MMC only) assertion of hardware reset signal.
- Card identification mode
The card will be in card identification mode after boot operation mode is finished or if host and /or card does not support boot operation mode. The card will be in this mode, until the SET_RCA command (CMD3) is received.
- Interrupt mode
Host and card enter and exit interrupt mode simultaneously. In interrupt mode there is no data transfer. The only message allowed is an interrupt service request from the card or the host.
- Data transfer mode
The card will enter data transfer mode once an RCA is assigned to it. The host will enter data transfer mode after identifying the card on the bus.
- Inactive mode
The card will enter inactive mode either card operating voltage range or access mode is not valid. The card can also enter inactive mode with Go_INACTIVE_STATE command (CMD15). The card will reset to *Pre-idle* state with power cycle.

The following table shows the dependencies between bus modes, operation modes and card states. Each state in the MultiMediaCard state diagram (see [Figure 26](#) and [Figure 28](#)) is associated with one bus mode and one operation mode:

2. Broadcast commands are kept for backwards compatibility to previous MultiMediaCard systems, where more than one card was allowed on the bus.

Table 5 — Bus modes overview

Card state	Operation mode	Bus mode
Inactive State	Inactive mode	Open-drain
Pre-Idle State	Boot mode	
Pre-Boot State		
Idle State		
Ready State		
Identification State		
Stand-by State	Data transfer mode	Push-pull
Sleep State		
Transfer State		
Bus-Test State		
Sending-data State		
Receive-data State		
Programming State		
Disconnect State		
Boot State		
Wait-IRQ State	Interrupt mode	Open-drain

7.2 Partition Management

7.2.1 General

The default area of the memory device consists of a User Data Area to store data, two possible boot area partitions for booting (Section 7.3.2 on page 35) and the Replay Protected Memory Block Area Partition (Section 7.6.16 on page 66) to manage data in an authenticated and replay protected manner.

The memory configuration initially consists (before any partitioning operation) of the User Data Area and RPMB Area Partitions and Boot Area Partitions (whose dimensions and technology features are defined by the memory manufacturer).

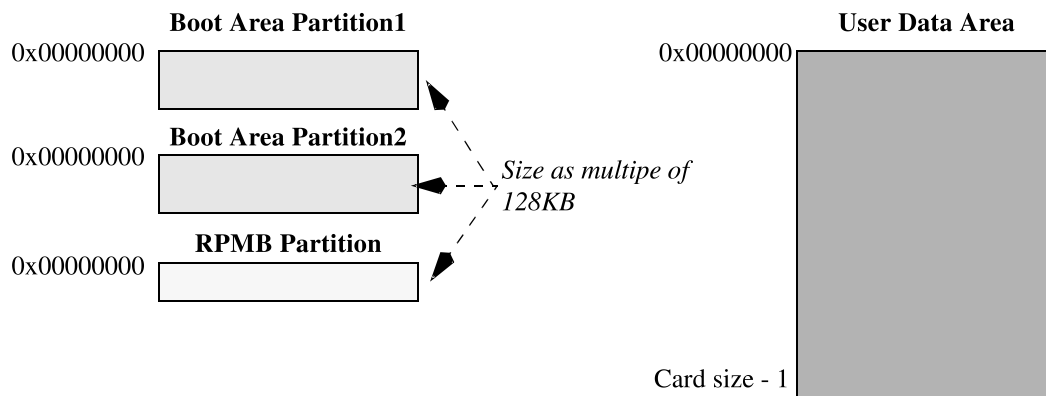


Figure 17 — e-MMC memory organization at time zero

The embedded device offers also the possibility of configuring by the host additional split local memory partitions with independent addressable space starting from logical address 0x00000000 for different usage models.

Therefore memory block Area scan be classified as follows:

- Two Boot Area Partitions, whose size is multiple of 128 KB and from which booting from e-MMC can be performed.
- One RPMB Partition accessed through a trusted mechanism, whose size is defined as multiple of 128 KB.
- Four General Purpose Area Partitions to store sensitive data or for other host usage models and whose size is multiple of a Write Protect Group.

Each of the General Purpose Area Partitions can be implemented with enhanced technological features (such as better reliability*) that distinguish them from the default storage media. If the enhanced storage media feature is supported by the device, boot and RPMB Area Partitions shall be implemented as enhanced storage media by default.

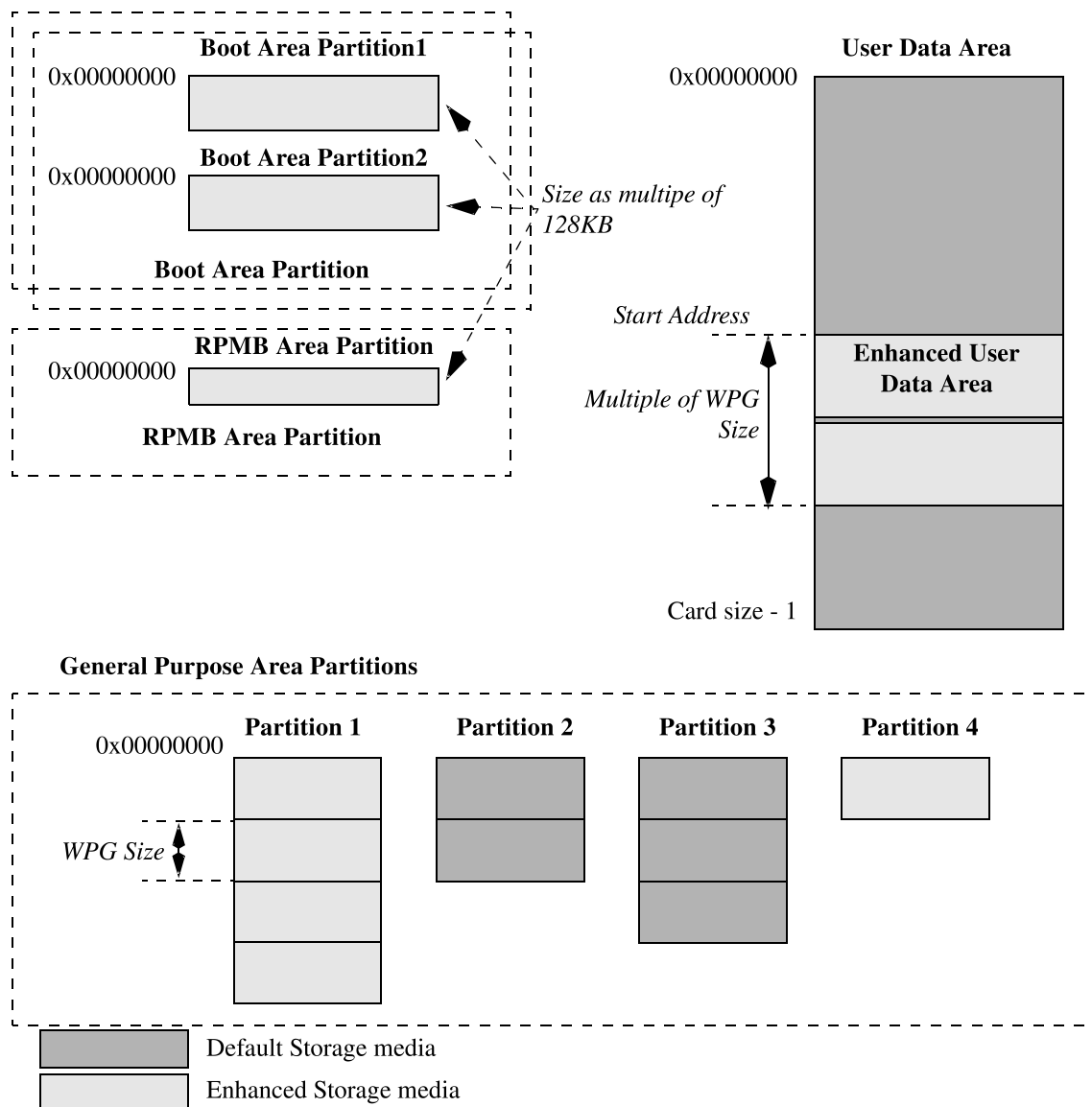
Boot and RPMB Area Partitions' sizes and attributes are defined by the memory manufacturer (read-only), while General Purpose Area Partitions' sizes and attributes can be programmed by the host only once in the device life-cycle (one-time programmable).

Moreover, the host is free to configure one segment in the User Data Area to be implemented as enhanced storage media, and to specify its starting location and size in terms of Write Protect Groups. The attributes of this Enhanced User Data Area can be programmed only once during the device life-cycle (one-time programmable).

* This is cited as an example of an enhanced storage media characteristics, and should not be considered as a necessary definition of enhanced storage media technology. The definition of enhanced storage media should be decided upon by each system manufacturer, and is outside the scope of this specification.

A possible final configuration can be the following one:

Figure 18 — Example of partitions and user data area configuration



General Purpose Partitions and Enhanced User Data Area configuration by the host can have effects on data previously stored (they will be destroyed) and the device initialization time.

In particular, the initialization time after first power cycle subsequent to the configuration can exceed the maximum initialization time defined by the specs since the internal controller could execute operations to set up the configurations stated by the host.

More generally also the following initialization phases can be affected by the new configuration.

Max power up timings shall be specified in the device technical literature.