SD and SDIO

1 The Introduction of SDIO

The SD standard offers great flexibility, include the ability to use the SD slot for more than memory cards. SDIO card is an interface that extends the functionality of devices by using a standard SD card slot to give devices new capabilities. A partial list of new capabilities incudes:

- GPS
- Camera
- Wi-Fi
- Ethernet
- Barcode readers
- Bluetooth

The SDIO (SD Input/Outpu) card is based on and compatible with the SD memory card. This compatibility incudes mechanical, electrical, power, signaling and software. The intent of the SDIO card is to provide high-speed data I/O with low power consumption for mobile electronic deivices.

1.1 SDIO Features

- Targeted for partable and stationary applications
- Minimal or no modification to SD Physical bus is required
- Minimal change to memory driver software
- Extended physical form faactor available for specialized applications
- Plug and play (PnP) support
- Multi-function support including multiple I/O and combined I/O and memory
- Up to 7 I/O functions plus one memory supported on one card.
- Allows card to interrupt host.
- Operational Voltage range: 2.7-3.6V
- Application Specifications for Standard SDIO Functions.
- Multiple Form Factos:
    - Full-Size SDIO
    - miniSDIO

2 SDIO Signaling Definition

2.1 SDIO Card Types

- Full-Speed card:
- supports SPI, 1-bit SD and the 4-bit SD transfer modes at the full clock range of 0-25MHz.
- Over 100Mb/second (10MB/Sec)
- Low-Speed SDIO card:
- Requires only the SPI and 1-bit SD transfer modes. 4-bit support is optional.
- Support a full clock range of 0-400 KHz

2.2 SDIO Card modes

2.2.1 SPI (Card mandatory support)

In this mode pin 8, which is undefined for memory, is used as the interrupt pin.

2.2.2 1-bit SD Data Transfer Mode (Card Mandatory Support)

In this mode, data is transfered on the DAT[0] pin only.

2.2.3 4-bit SD Data Transfer Mode (Mandatory for Hight-Speed Cards, Optional for Low-Speed)

In this mode, data is transferred on all 4 data pins (DAT[3:0]). The 4-bit SD mode provides the hightest data transfer possible, up to 100Mb/sec.

2.3 Signal Pins

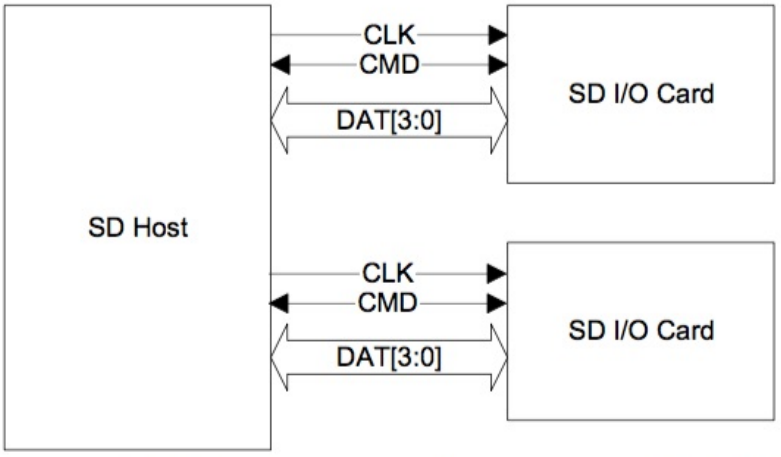


Figure 2-1 Signal connection to two 4-bit SDIO cards

3 SD Bus Protocol

3.0.1 SD Bus

Communication over the SD bus is based on command and data bit streams that are initiated by a start bit and terminated by a stop bit.

- Command : a command is a token that starts an operation. A command is sent from the host either to a single card (addressed command) or to all connected cards (broadcast command). A command is transferred serially on the CMD line.
- Response : a response is a token that is sent from an adressed card, or (synchronously) from all connected cards, to the hosts 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 ro vice versa. Data is transfrerred via the data lines.

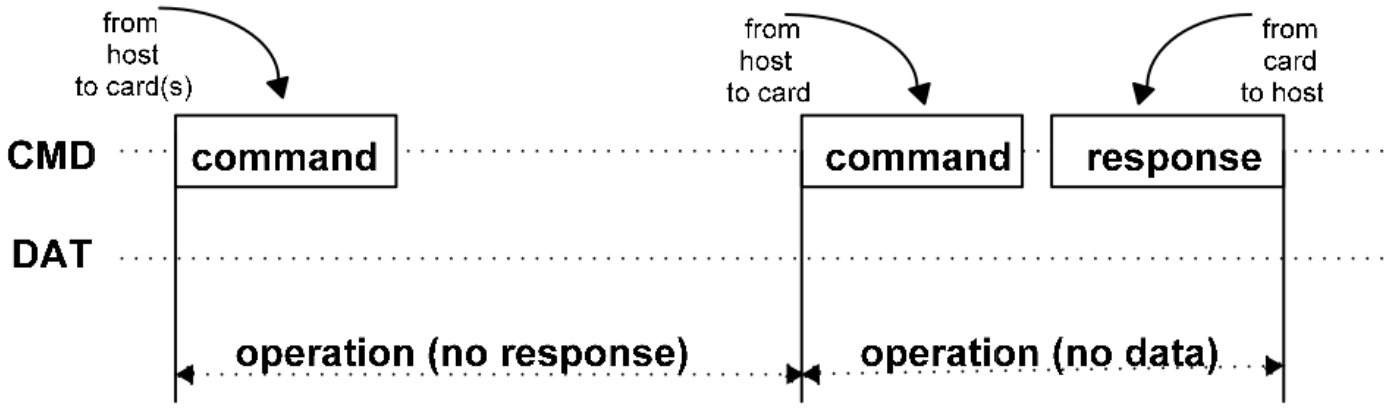


Figure 3-2: “no response” and “no data” Operations

Card addressing is implemented using as session address, assigned to the card druing the initialization phase.
Data transfer to/from the SD Memory Card are done in blocks. Data blocks are always succeeded by CRC bits. Single and multiple block operations are defined. Note that the Multiple Block operation mode is better for faster write operation. A multiple block transmission is terminated when a stop command follows on the CMD line. Data transfer can be configured by the host to use single or multiple data lines.

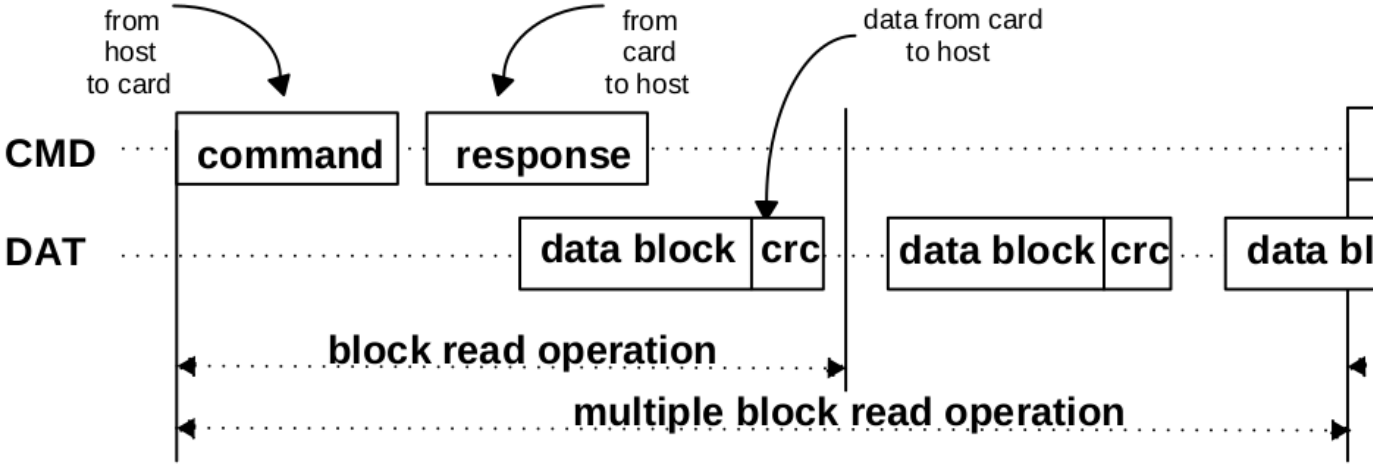


Figure 3-3: (Multiple) Block Read Operation

The block write operation uses a simple busy signaling of the write operation duration on the DAT0 data line regardless of the number of data lines used for transferring the data.

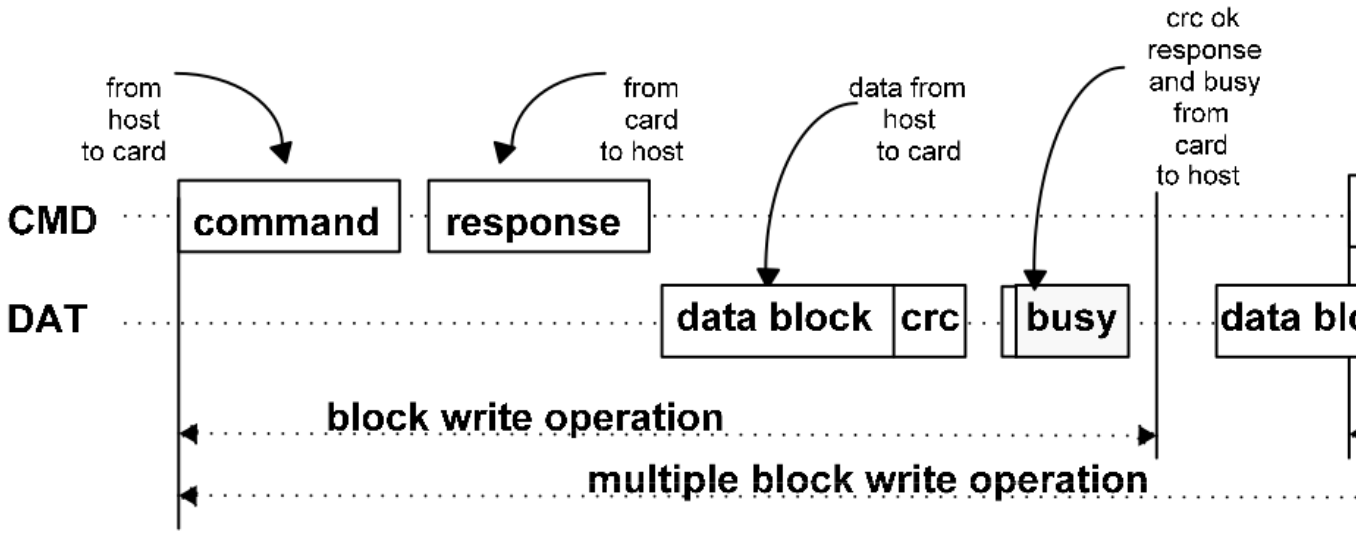


Figure 3-4: (Multiple) Block Write Operation

Command tokens have the following coding scheme:

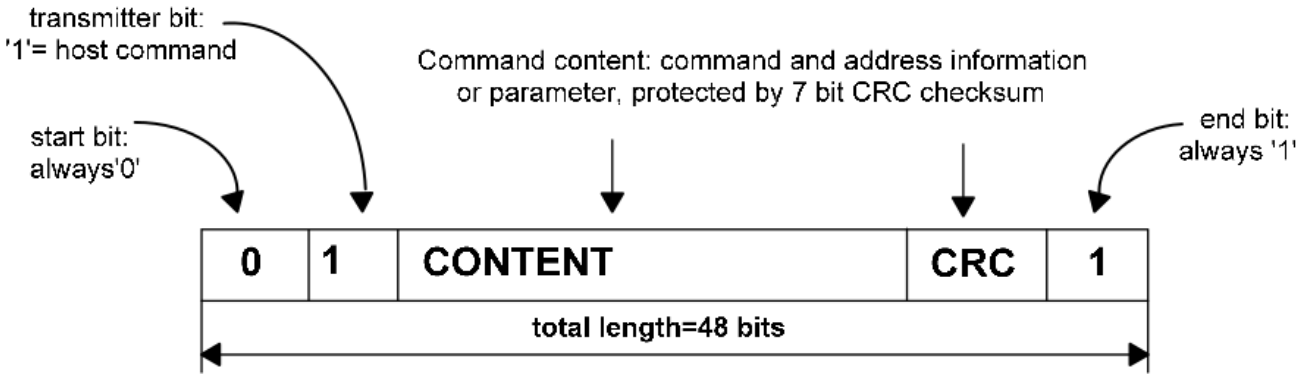


Figure 3-5: Command Token Format

Each command token is precedes by a start bit (0) and succeeded by and 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.

Resopnse tokens have one of four coding schemes, depending on their content. The token length is either 48 or 136 bits. The CRC protection algorithm for block data is a 16-bit CCITT polynomial.

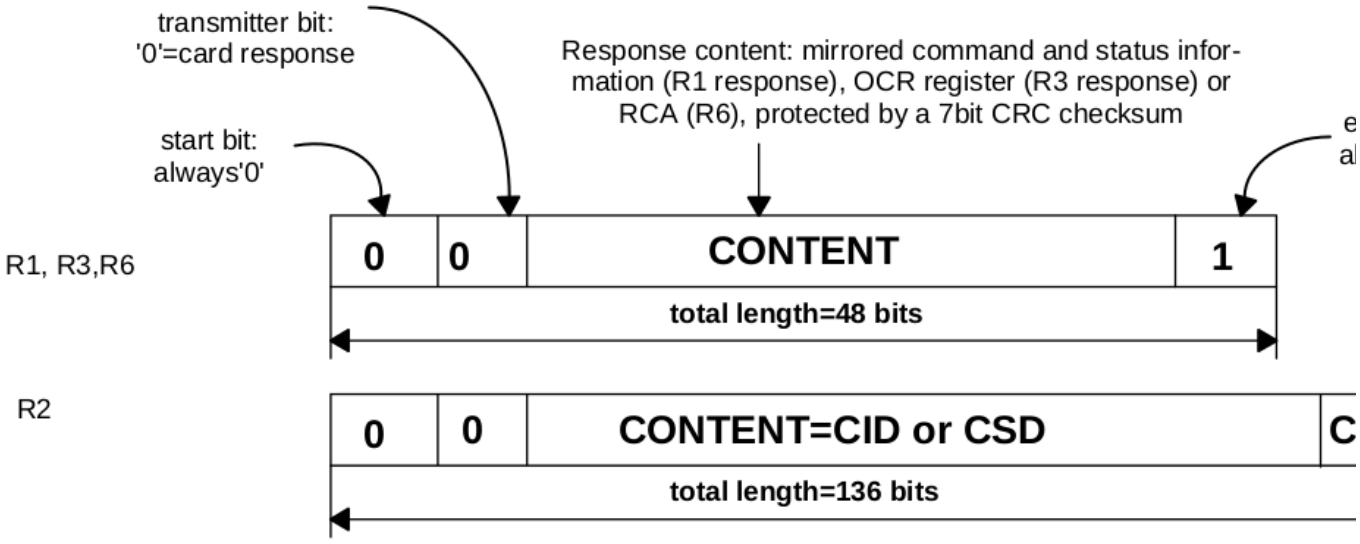


Figure 3-6: Response Token Format

In the CMD line the Most Significant Bit (MSB) is transmitted first, the Least Significant Bit (LSB) is the last. When the wide bus option is used, the data is transferred 4 bits at a time. Start and end bits, as well as the CRC bits, are transmitted for every one of the DAT lines. CRC bits are calculated and checked for every DAT line individually. The CRC status response and Busy indication will be sent by the card to the host on DAT0 only.

There are two types of Data packet format for the SD card. (1) Usual data (8-bit width): The usual data (8-bt width) are sent in LSB (Least Significant Byte) first, MSB (Most Significant Byte) last sequence. But in the individual byte, it is MSB (Most Significant Bit) first, LSB (Least Significant Bit) last. (2) Wide width data (SD Memory Register): The wide width data is shifted from the MSB bit.

- Data Packet Format for Usual Data(8-bit width)

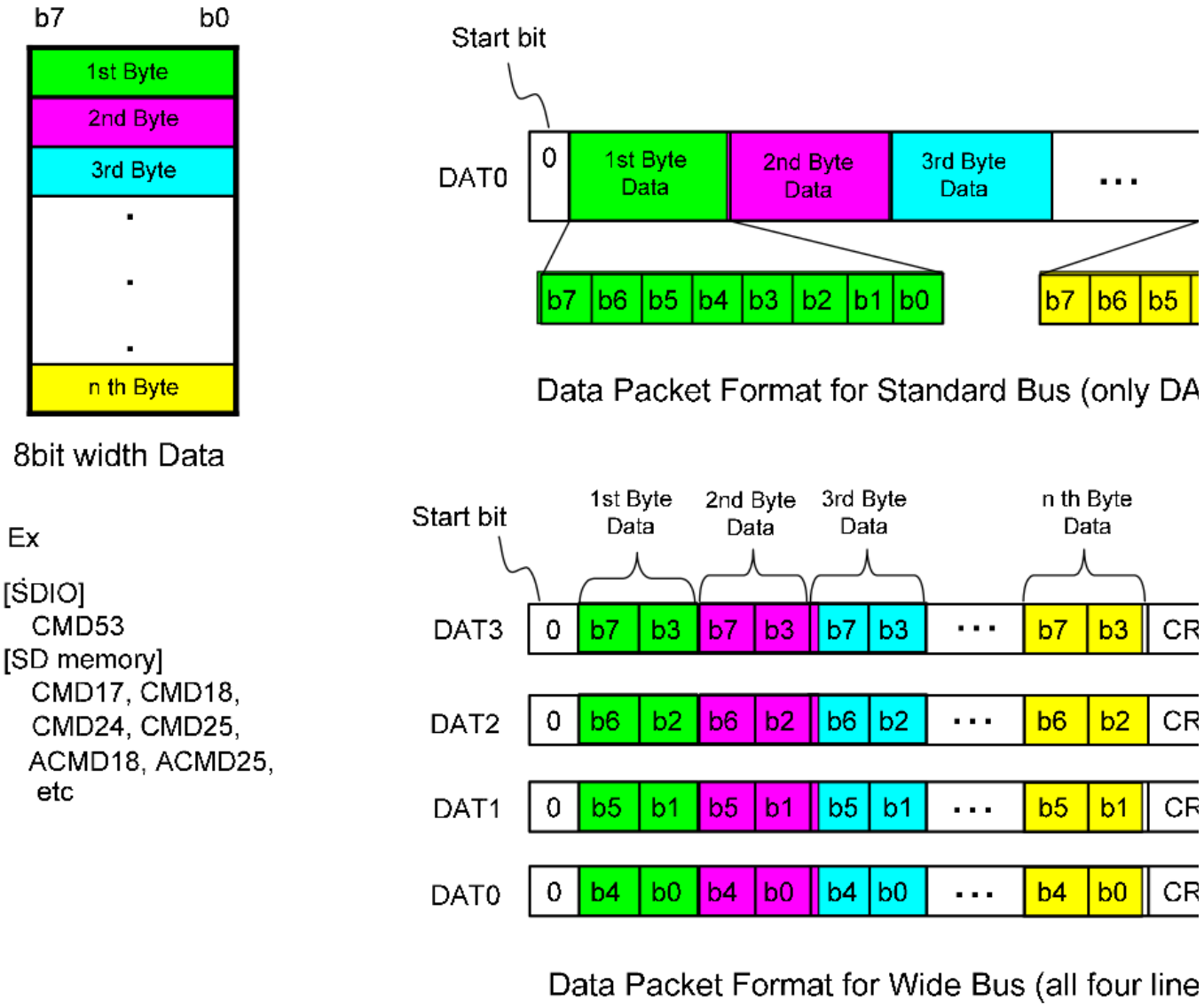


Figure 3-7: Data Packet Format - Usual Data

- Data Packet Format for Wide Width Data (Ex. ACMD13)

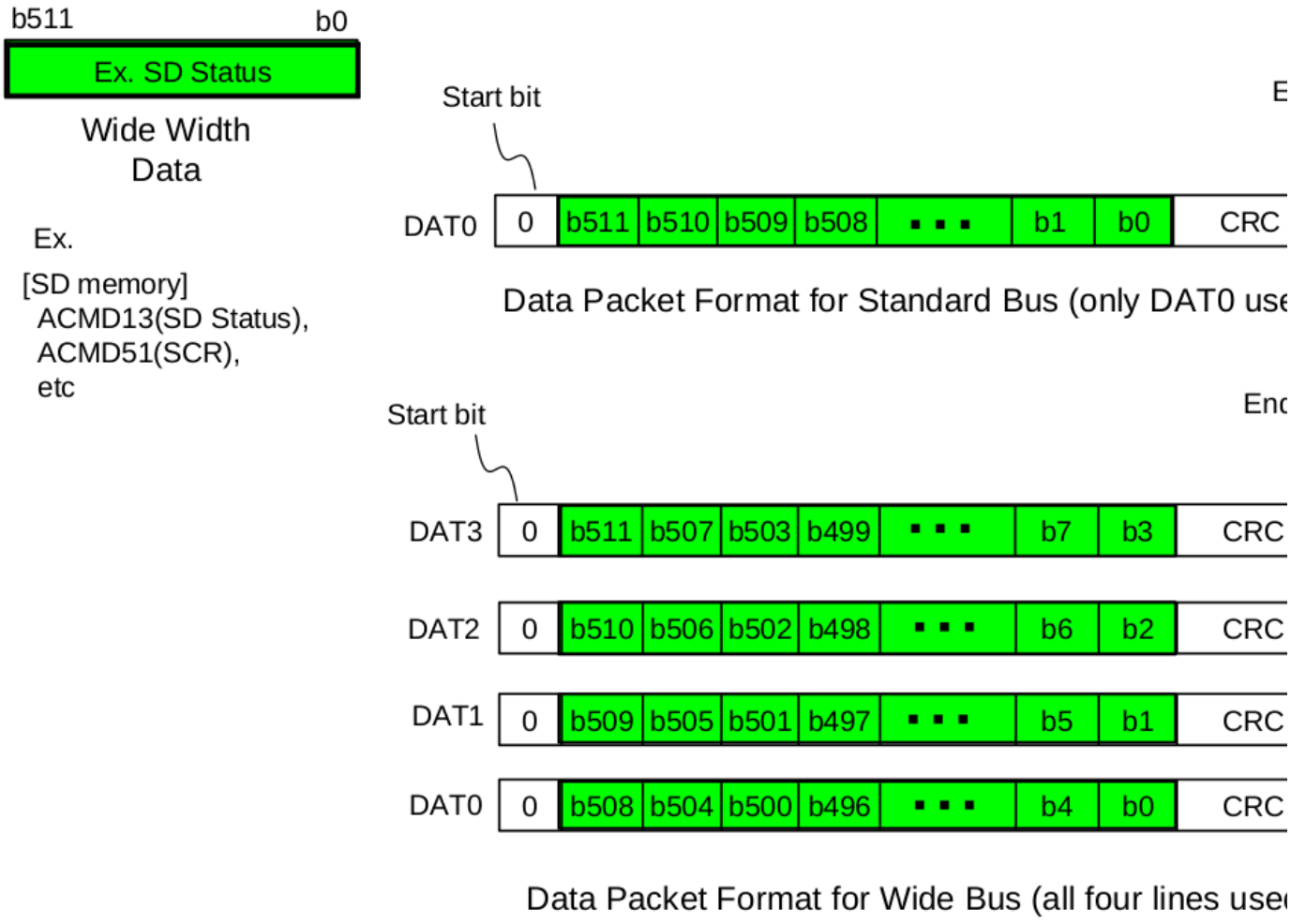


Figure 3-8: Data Packet Format - Wide Width Data

3.1 SD Memory Card Functional Description

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

- Broadcast commands

Broadcast commands are intended for all cards. 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.

- Card identification mode

The host will be in card identification mode after reset and while it is looking for new cards on the bus. Cards will be in this mode after reset until the $SEND_{RCA}$ command (CMD3) is received.

- Data transfer mode

Cards will enter data transfer mode after their RCA is first published. The host will enter data transfer mode after identifying all the cards on the bus.

The following table shows the dependencies between operation modes and card states.

Card state	Operation mode
Inactive State	inactive
Idle State	card identification mode
Ready State	
Identification State	
Stand-by State	data transfer mode
Transfer State	
Sending-data State	
Receive-data State	
Programming State	
Disconnect State	

Table 4-1: Overview of Card States vs. Operation Modes

3.2 Card Identification Mode

While in card identification mode the host resets all the cards that are in card identification mode, validates operation voltage range, identifies cards and asks them to publish Relative Card Address(RCA). This operation is done to each ard separately on its own CMD line. All data communication in the Card Identification Mode uses the command line (CMD) only. During the card identification process, the card shall operate in the SD clock frequency of the identificaiton clock rate.

3.2.1 Card Reset

The command $GO_{IDLESTATE}$(CMD0) is the software reset command and sets each card into *Idle State* regardless of the current card state.

3.2.2 Operating Condition Validation

The host issues a reset command (CMD0) with a specified voltage while assuming it may be supported by the card. To verify the voltage, a following new command (CMD8) is defined in the Physical Layer Specification Version 2.00.
$SEND_{IFCOND}$ (CMD8) is used to verify SD Memory Card interface operating condition. The card checks the validity of operating condition by analyzing the argument of CMD8 and the host checks the validity by analyzing the reponse of CMD8. The supplied voltage is indicated by VHS filed in the argument. The card assumes the voltage specified in VHS as the current supplied voltage. Only 1-bit of VHS shall be set to 1 at any given time. Both CRC and check pattern are used for the host to check validity of communication between the host and the card. If the card cannot operate on the supplied voltage, it returns no response and stays in idle state. It is mandatory to issue CMD8 prior to first ACMD41 for initialization of High Capacity SD Memory Card.

$SD_{SENDOPCOND}$ (ACMD41) is designed to provide SD Memory Card hosts with a mechanism to identify and reject cards which do not match the V_{DD} range desired by the host. Cards which cannot perform data transfer in the specified range shall discard themselves from further bus operations and go into *Inactive StTATE*. Note that ACMD41 is application specific command, therefore APP_{CMD} (CMD55) shall always precede ACMD41.

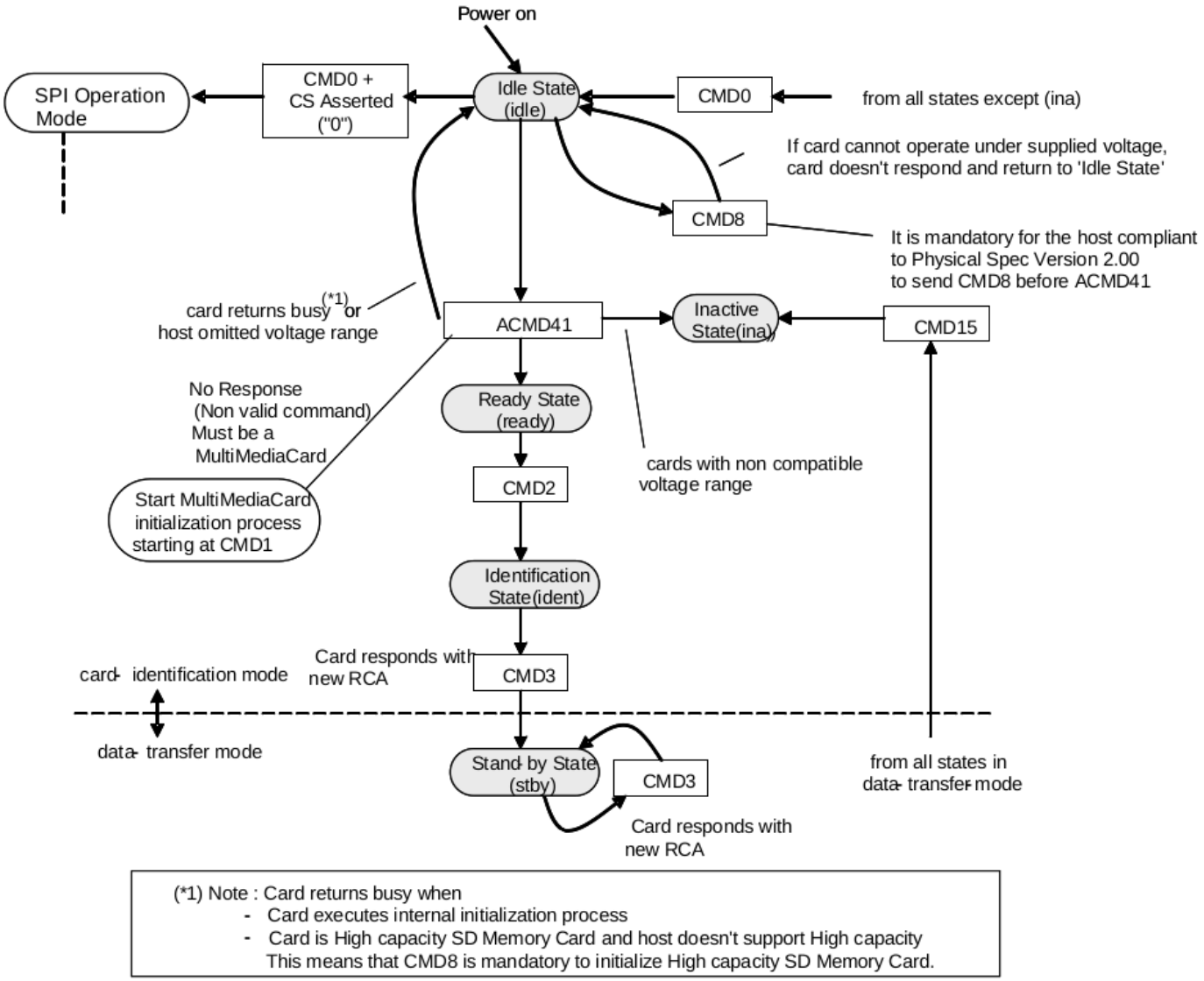


Figure 4-1: SD Memory Card State Diagram (card identification mode)

3.3 Card Initialization and Identification

The initialization process starts with $SD_{SENDOPCOND}$ (ACMD41) by setting its operational conditions and the HCS bit in the OCR. The HCS (Host Capacity Support) bit set to 1 indicates that the host supports High Capacity SD Memeory card. The HCS (Host Capacity Support) bit set to 0 indicates that the host does not support Hight Capacity SD Memory card.

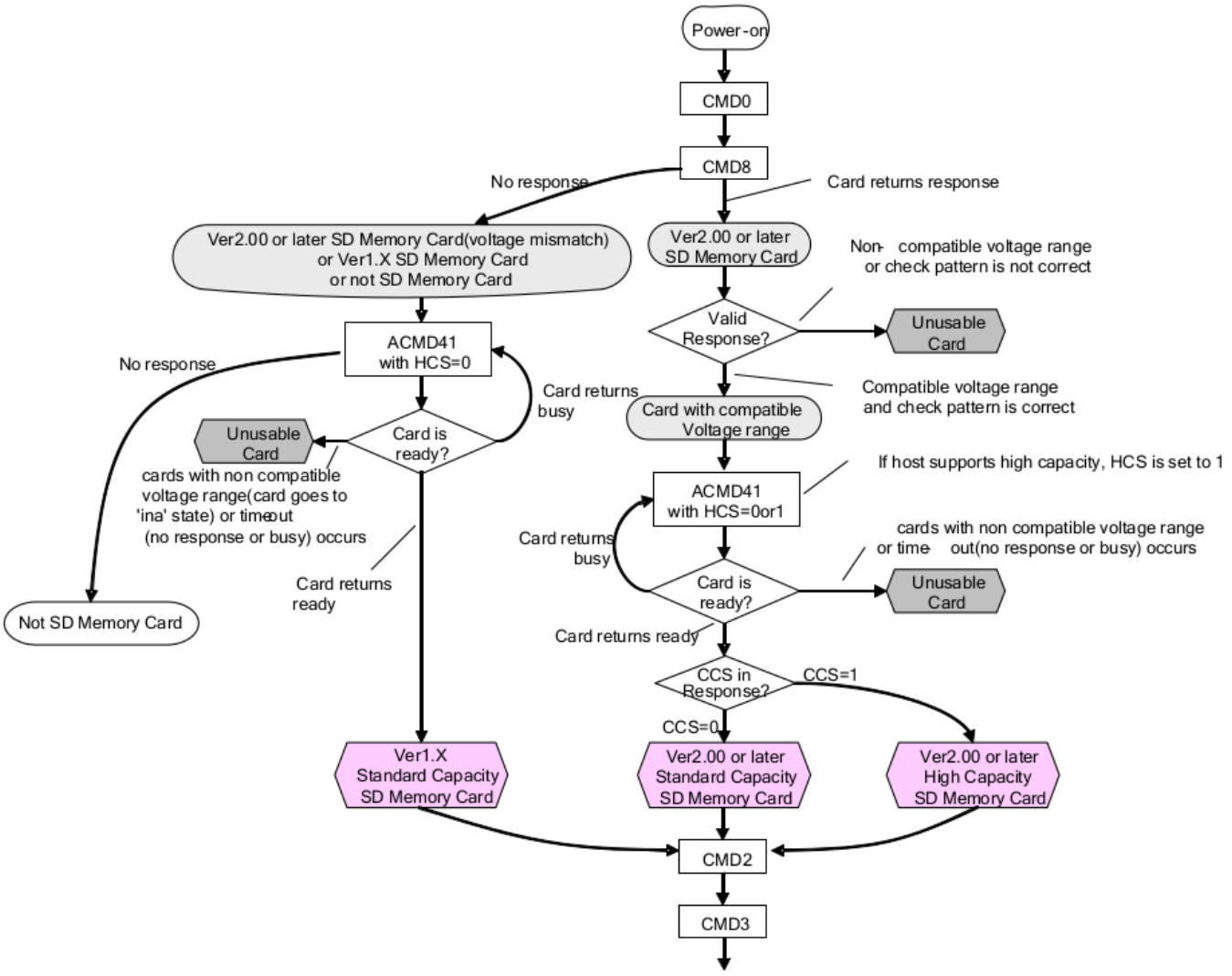


Figure 4-2: Card Initialization and Identification Flow (SD mode)

3.4 Data Transfer Mode

Until the end of Card Identificaton Mode the host shall remain at f_{OD} freequency because some cards may have operating frequency restrictions during the card identification mode. In Data Transfer Mode the host may operate the card in f_{PP} frequency range. The host issues $SEND_{CSD}$(CMD9) to obtain the Card Specific Data(CSD register), e.g. block length, card storage capacity, etc. The broadcast command SET_{DSR} (CMD4) configures the driver stages of all identified cards. It programs their DSR registers corresponding to the application bus layout and the number of cards on the bus and the data transfer frequency. The clock rate is also switched from f_{OD} to f_{PP} at that point. SET_{DSR} command is an option for the card and the host.

CMD7 is used to select one card and put it into the Transfer State. Only one card can be in the Transfer State at a given time. If a previously selected card is in the Transfer State its connection with the host is released and it will move back to the *Stand-by State*. When CMD7 is issued with the reserved relative card address "0x0000", all cards are put back to *Stand-by State*.

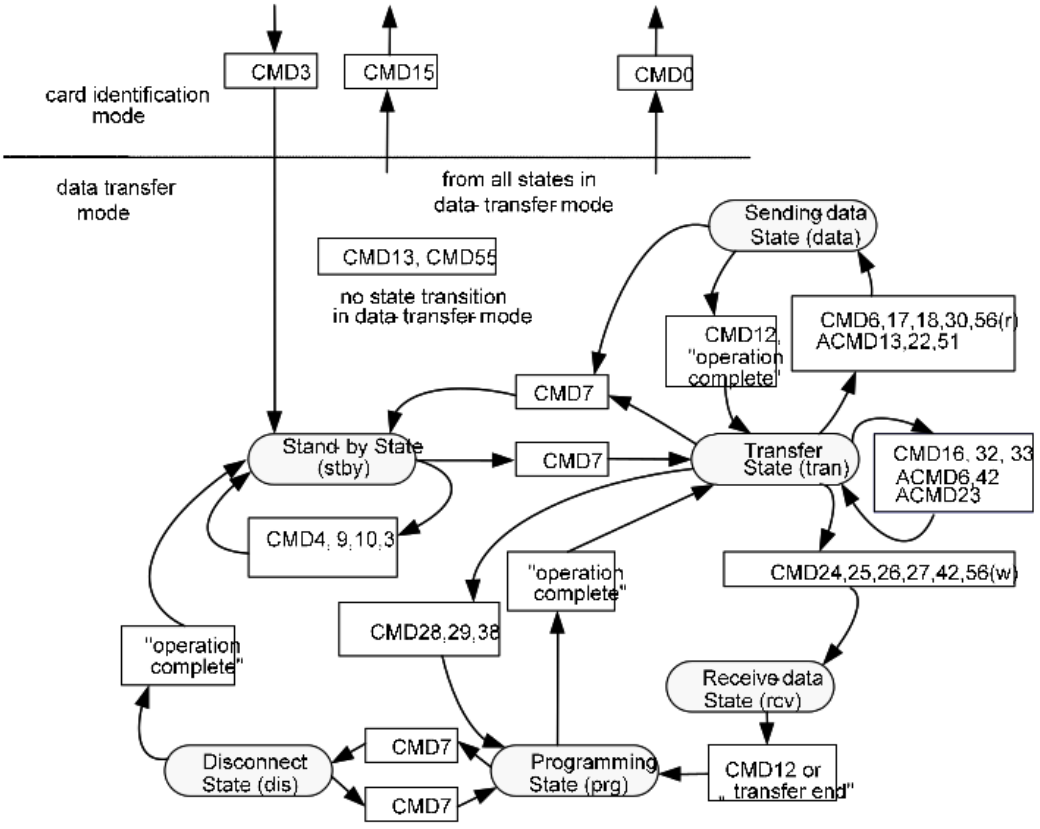


Figure 4-3: SD Memory Card State Diagram (data transfer mode)

The relationship between the various data transfer modes is summarized below.

- All data read commands can be aborted any time by the stop command (CMD12). The data transfer

will terminate and the card will return to the *Transfer State*. The read command are: block read (CMD17), multiple block read (CMD18), send write protect (CMD30), send scr (ACMD51) and general command in read mode (CMD56).

- All data write commands can be aborted any time by the stop command (CMD12). The write commands shall be stopped prior to deselecting the card by CMD7. The write commands are: block write (CMD24 and CMD25), program CSD (CMD27), lock/unlock command (CMD42) and general command in write mode (CMD56).
- As soon as the data transfer is completed, the card will exit the data write state and move either to the *Programming State* (transfer is successful) or *Transfer State* (transfer failed).
- If a block write operation is stopped and the block length and CRC of the last block are valid, the data will be programmmed.
- The card may provide buffering for block write. This means that the next block can be sent to the card while the previous is being programmed.

If all write buffers are full, and as long as the card is in *Programming State*, the DAT0 line will be kept low (BUSY).

- There is no buffering option for write CSD, write protection and erase. This means that while the card is busy servicing any one of these commands, no other data transfer commands will be accepted.

DAT0 line will be kept low as long as the card is busy and in the *Programming State*. Actually if the CMD and DAT0 lines of the cards are kept separated and the host keep the busy DAT0 line disconnected from the other DAT0 lines(of the other cards) the host may access the other cards while the card is in busy.

- Parameter set commands are *not* allowed while card is programming.

Parmeter set commands are: set block length (CMD16), erase block start (CMD32) and erase block end (CMD33).

- Read commands are not allowed while card is programming.
- Moving another card from *Stand-by* to *Transfer State* (using CMD7) will not terminate erase and

programming operations. The card will switch to the *Disconnect State* and will release the DAT line.

- A card can be reslected while in the *Disconnect State*, using CMD7. In this case the card will move to the *Programming State* and reactivate the busy indication.
- Resetting a card (using CMD0 or CMD15) will terminate any pending or active programming operation. This may destroy the data contents on the card. It is the host's

responsibility to prevent this.

- CMD34-37, CMD50 and CMD57 are reserved for SD command system expansion. State transitions for these commands are defined in each command system specification.

3.5 Commands

3.5.1 Command Format

All commands have a fixed code length of 48 bits, needing a transmission time of 1.92 us @ 25 MHz and 0.96 us @ 50 MHz.

Bit position	47	46	[45:40]	[39:8]	[7:1]	0
Width (bits)	1	1	6	32	7	1
Value	'0'	'1'	x	x	x	'1'
Description	start bit	transmission bit	command index	argument	CRC7	end bit

Table 4-16: Command Format

3.5.2 Detailed Command Description

CMD INDEX	type	argument	resp	abbreviation	command description
CMD0	bc	[31:0] stuff bits	-	GO_IDLE_STATE	Resets all cards to idle state
CMD1	reserved				
CMD2	bcr	[31:0] stuff bits	R2	ALL_SEND_CID	Asks any card to send the CID numbers on the CMD line (any card that is connected to the host will respond)
CMD3	bcr	[31:0] stuff bits	R6	SEND_RELATIVE_ ADDR	Ask the card to publish a new relative address (RCA)
CMD4	bc	[31:16] DSR [15:0] stuff bits	-	SET_DSR	Programs the DSR of all cards
CMD5	reserved for I/O cards (refer to the "SDIO Card Specification")				
CMD7	ac	[31:16] RCA [15:0] stuff bits	R1b (only from the selected card)	SELECT/DESELECT _CARD	Command toggles a card between the stand-by and transfer states or between the programming and disconnect states. In both cases, the card is selected by its own relative address and gets deselected by any other address; address 0 deselects all. In the case that the RCA equals 0, then the host may do one of the following: - Use other RCA number to perform card de-selection. - Re-send CMD3 to change its RCA number to other than 0 and then use CMD7 with RCA=0 for card de-selection.
CMD8	bcr	[31:12]reserved bits [11:8]supply voltage(VHS) [7:0]check pattern	R7	SEND_IF_COND	Sends SD Memory Card interface condition, which includes host supply voltage information and asks the card whether card supports voltage. Reserved bits shall be set to '0'.
CMD9	ac	[31:16] RCA [15:0] stuff bits	R2	SEND_CSD	Addressed card sends its card-specific data (CSD) on the CMD line.
CMD10	ac	[31:16] RCA [15:0] stuff bits	R2	SEND_CID	Addressed card sends its card identi-fication (CID) on CMD the line.
CMD11	reserved				
CMD12	ac	[31:0] stuff bits	R1b	STOP_ TRANSMISSION	Forces the card to stop transmission
CMD13	ac	[31:16] RCA [15:0] stuff bits	R1	SEND_STATUS	Addressed card sends its status register.
CMD14	reserved				

CMD INDEX	type	argument	resp	abbreviation	command description
CMD15	ac	[31:16] RCA [15:0] reserved bits	-	GO_INACTIVE_ STATE	Sends an addressed card into the *Inactive State*. This command is used when the host explicitly wants to deactivate a card. Reserved bits shall be set to '0'.

Table 4-18: Basic Commands (class 0)

CMD INDEX	type	argument	resp	abbreviation	command description
CMD16	ac	[31:0] block length	R1	SET_BLOCKLEN	In the case of a Standard Capacity SD Memory Card, this command sets the block length (in bytes) for all following block commands (read, write, lock). Default block length is fixed to 512 Bytes. Set length is valid for memory access commands only if partial block read operation are allowed in CSD. In the case of a High Capacity SD Memory Card, block length set by CMD16 command does not affect the memory read and write commands. Always 512 Bytes fixed block length is used. This command is effective for LOCK_UNLOCK command. In both cases, if block length is set larger than 512Bytes, the card sets the BLOCK_LEN_ERROR bit.
CMD17	adtc	[31:0] data address[2]	R1	READ_SINGLE_ BLOCK	In the case of a Standard Capacity SD Memory Card, this command, this command reads a block of the size selected by the SET_BLOCKLEN command. [1] In the case of a High Capacity Card, block length is fixed 512 Bytes regardless of the SET_BLOCKLEN command.
CMD18	adtc	[31:0] data address[2]	R1	READ_MULTIPLE_ BLOCK	Continuously transfers data blocks from card to host until interrupted by a STOP_TRANSMISSION command. Block length is specified the same as READ_SINGLE_BLOCK command.
CMD19 ... CMD23	reserved				

1) The data transferred shall not cross a physical block boundary unless READ_BLK_MISALIGN is set in the CSD.
2) Data address is in byte units in a Standard Capacity SD Memory Card and in block (512 Byte) units in a High Capacity SD Memory Card.

Table 4-19: Block-Oriented Read Commands (class 2)

CMD INDEX	type	argument	resp	abbreviation	command description
CMD16	ac	[31:0] block length	R1	SET_BLOCKLEN	See description in Table 4-19
CMD24	adtc	[31:0] data address[2]	R1	WRITE_BLOCK	In the case of a Standard Capacity SD Memory Card, this command writes a block of the size selected by the SET_BLOCKLEN command. [1] In the case of a High Capacity Card, block length is fixed 512 Bytes regardless of the SET_BLOCKLEN command.
CMD25	adtc	[31:0] data address[2]	R1	WRITE_MULTIPLE _BLOCK	Continuously writes blocks of data until a STOP_TRANSMISSION follows. Block length is specified the same as WRITE_BLOCK command.
CMD26	Reserved For Manufacturer				
CMD27	adtc	[31:0] stuff bits	R1	PROGRAM_CSD	Programming of the programmable bits of the CSD.

1) The data transferred shall not cross a physical block boundary unless WRITE_BLK_MISALIGN is set in the CSD. In the case that write partial blocks is not supported, then the block length=default block length (given in CSD).
2) Data address is in byte units in a Standard Capacity SD Memory Card and in block (512 Byte) units in a High Capacity SD Memory Card.

Table 4-20: Block-Oriented Write Commands (class 4)

3.6 Responses

All responses are sent via the command line CMD. The response transmission always starts with the left bit of the bit string corresponding to the response codeword. The code length depends on the response type.

There are five types of responses for the SD Memory Card. The SDIO Card supports additional response types named R4 and R5.

3.6.1 R1 (normal response command):

Code length is 48 bits. The bits 45:40 indicate the index of the command to be responded to, this value being interpreted as a binary coded number (between 0 and 63). The status of the card is coded in 32 bits.

Bit position	47	46	[45:40]	[39:8]
Width (bits)	1	1	6	32
Value	'0'	'0'	x	x
Description	start bit	transmission bit	command index	card status

Table 4-29: Response R1

3.6.2 R1b

R1b is identical to R1 with an optional busy signal transmitted on the data line. The card may become busy after receiving these commands based on its state prior to the command reception. The Host shall check for busy at the response.

3.6.3 R2 (CID, CSD register)

Code length is 136 bits. The contents of the CID register are sent as a response to the commands CMD2 and CMD10. The contents of the CSD register are sent as a response to CMD9. Only the bits [127...1] of the CID and CSD are transferred, the reserved bit [1] of these regisers is replaced by the end bit of the response.

Bit position	135	134	[133:128]	[1
Width (bits)	1	1	6	
Value	‘0’	‘0’	‘111111’	
Description	start bit	transmission bit	reserved	CID or CS interr

Table 4-30: Response R2

3.6.4 R3 (OCR register)

Code length is 48 bits. The contents of the OCR register are sent as a response to ACMD41.

Bit position	47	46	[45:40]	[39:8]
Width (bits)	1	1	6	32
Value	‘0’	‘0’	‘111111’	x
Description	start bit	transmission bit	reserved	OCR regis

Table 4-31: Response R3

3.6.5 R6 (Published RCA response)

Code length is 48 bit. The bits 45:40 indicate the index of the Command to be responded to - in that case, it will be '000011'(together with bit 5 in the status bits it means = CMD3). The 16 MSB bits of the argument field are used for the Published RCA number.

Bit position	47	46	[45:40]	[39:8] Argument fie	
Width (bits)	1	1	6	16	
Value	'0'	'0'	x	x	
Description	start bit	transmission bit	command index ('000011')	New published RCA [31:16] of the card	[15:0] status 23,22, (see T

Table 4-32: Response R6

3.7 R7 (Card interface condition)

Code length is 48 bits. The card support voltage information is sent by the response of CMD8. Bits 19–16 indicate the voltage range that the card supports. The card that accepted the supplied voltage returns R7 response. In the response, the card echoes back both the voltage range and check pattern set in the argument.

Bit position	47	46	[45:40]	[39:20]	[19:16]	[15
Width (bits)	1	1	6	20	4	8
Value	'0'	'0'	'001000'	'00000h'	x	x
Description	start bit	transmission bit	command index	reserved bits	voltage accepted	echo- of ch patt

Table 4-33: Response R7

4 SDIO Card Initializatio

After reset or power-up, all I/O functions on the card are disabled and the I/O portion of the card shall not execute any operation except CMD5 or CMD0 with CS=low. If there is SD memory installed on the card, that memory shall respond normally to all normal mandatory memory commands.

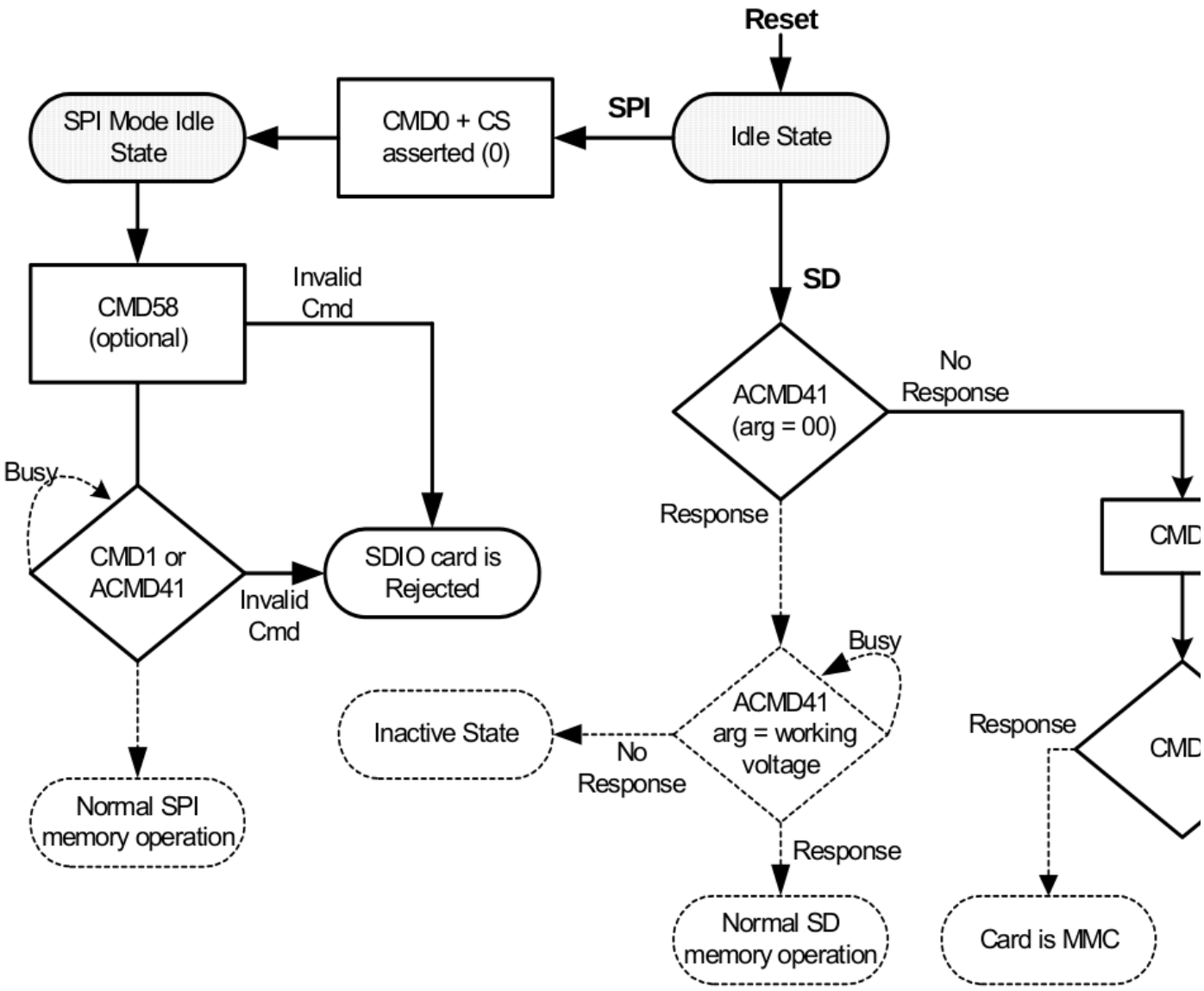
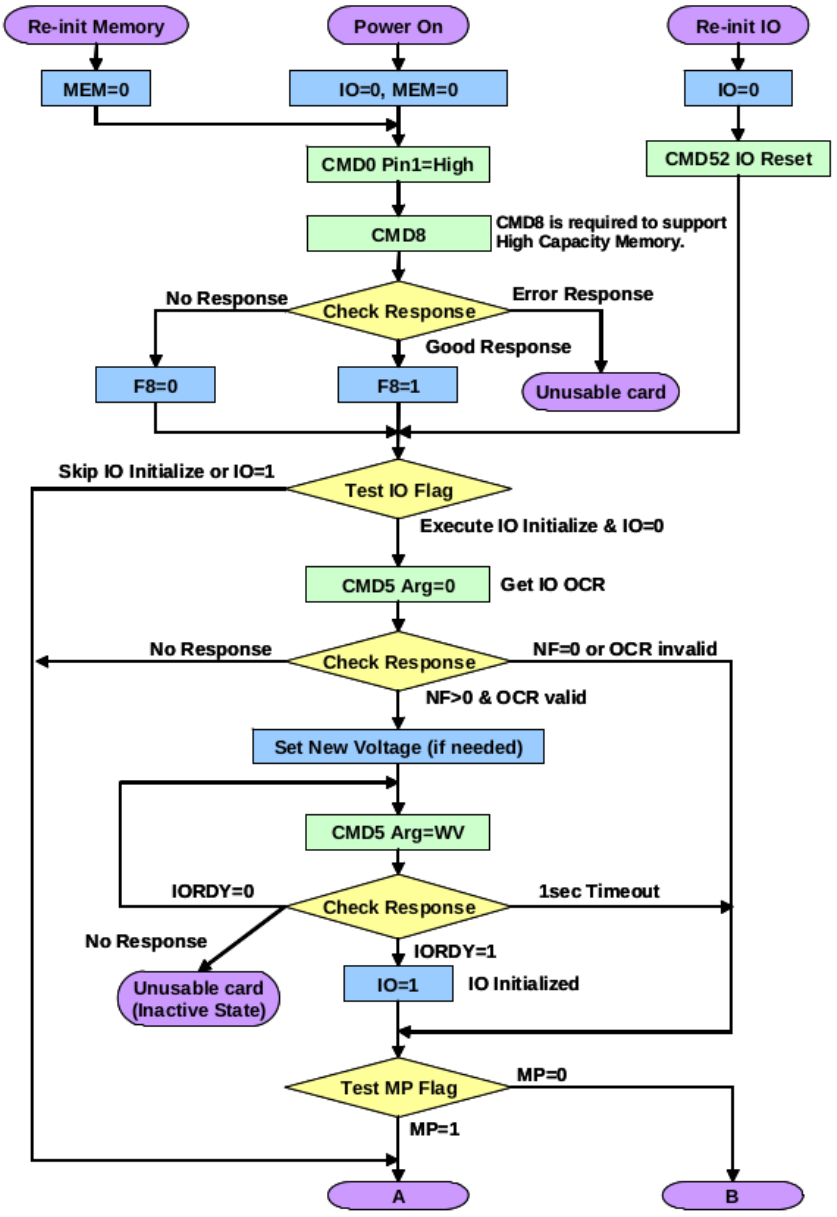


Figure 3-1 SDIO response to non-I/O aware initialization

An SDIO aware host shall send CMD5 arg=0 as part of the initialization sequence after either Power On or a CMD52 with write to I/O Reset. Sending CMD5 arg=0 that has not been preceded by one of these two reset conditions shall not result int either the host or card entering the initialization sequence.


Re-init Memory
Power On
Re-init IO
MEM=0
IO=0, MEM=0
IO=0
CMD0 Pin1=High
CMD52 IO Reset
CMD8
CMD8 is required to support High Capacity Memory.
No Response
Check Response
Error Response
Good Response
F8=0
F8=1
Unusable card
Skip IO Initialize or IO=1
Test IO Flag
Execute IO Initialize & IO=0
CMD5 Arg=0
Get IO OCR
No Response
Check Response
NF=0 or OCR invalid
NF>0 & OCR valid
Set New Voltage (if needed)
CMD5 Arg=WV
IORDY=0
Check Response
1sec Timeout
No Response
IORDY=1
Unusable card (Inactive State)
IO=1
IO Initialized
Test MP Flag
MP=0
MP=1
A
B

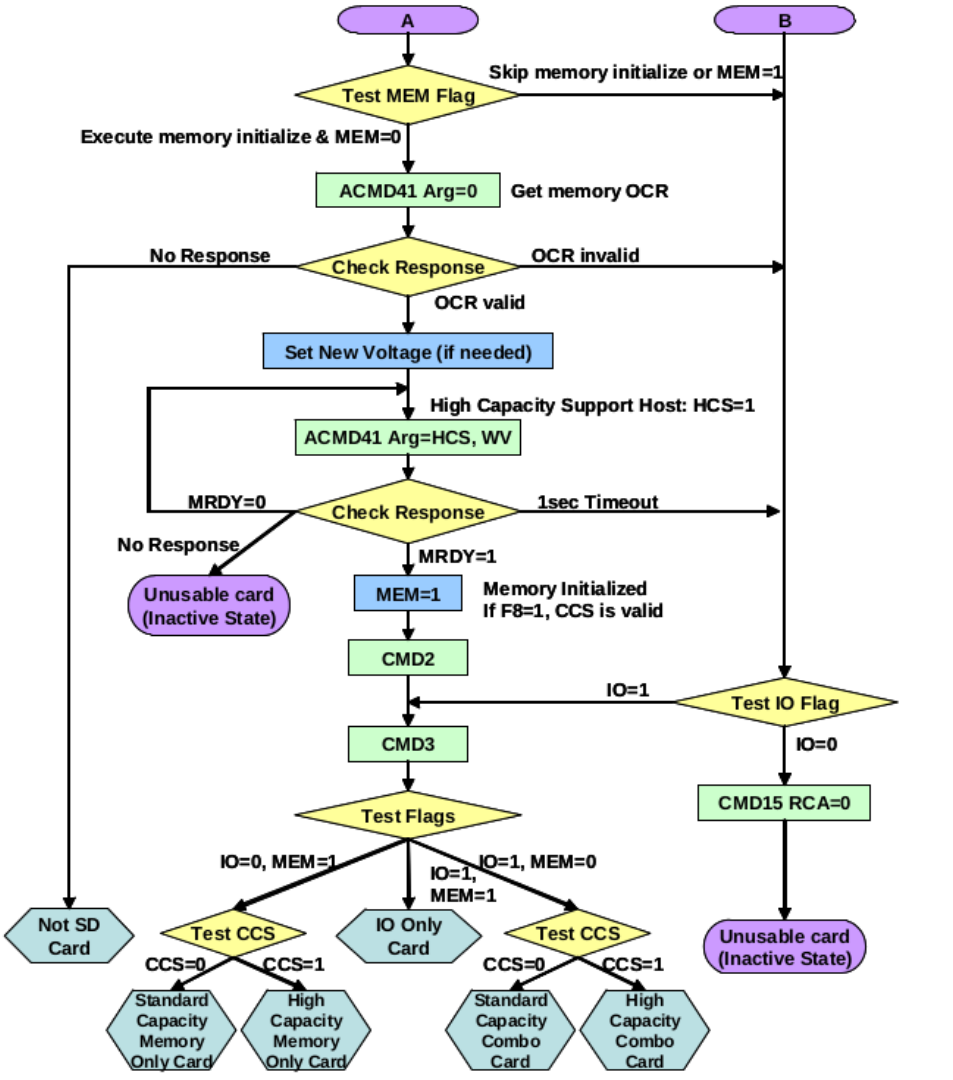


Figure 3-2 Card initialization flow in SD mode (SDIO aware host)

4.1 The $IO_{SENDOPCOND}$ Command (CMD5)

Figure 3-4 shows the format of the $IO_{SENDOPCOND}$ command (CMD5). The function of CMD5 for SDIO cards is similar to the operation of ACMD41 for SD memory cards. It is used to inquire about the voltage range needed by the I/O card. The normal response to CMD5 is R4 in either SD or SPI format. The R4 response in SD mode is shown in Figure 3-5 and the SPI version is shown in Figure 3-6.

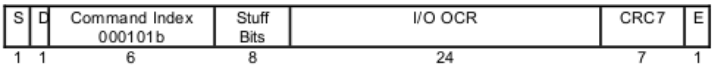


Figure 3-4 IO_SEND_OP_COND Command (CMD5)

The IO_SEND_OP_COND Command contains the following fields:

S(tart bit):	Start bit. Always 0
D(irection):	Direction. Always1 indicates transfer from host to card.
Command Index:	Identifies the CMD5 command with a value of 000101b
Stuff Bits:	Not used, shall be set to 0.
I/O OCR:	Operation Conditions Register. The supported minimum and maximum values for VDD. The layout of the OCR is shown in Table 3-1. See section 4.10.1 for additional information.
CRC7:	7 bits of CRC data
E(nd bit):	End bit, always 1

I/O OCR bit position	VDD Voltage Window (in Volts)
0-3	Reserved
4	Reserved
5	Reserved
6	Reserved
7	Reserved
8	2.0-2.1
9	2.1-2.2
10	2.2-2.3
11	2.3-2.4
12	2.4-2.5
13	2.5-2.6
14	2.6-2.7
15	2.7-2.8
16	2.8-2.9
17	2.9-3.0
18	3.0-3.1
19	3.1-3.2
20	3.2-3.3
21	3.3-3.4
22	3.4-3.5
23	3.5-3.6

Table 3-1 OCR Values for CMD5

4.2 The $IO_{SNEDOPCOND}$ Response (R4)

An SDIO card receiving CMD5 shall respond with a SDIO unique response, R4. the format of R4 for both the SD and SPI modes is:

S	D	Reserved	C	Number of I/O functions	Memory Present	Stuff Bits	I/O OCR	Reserved	E
1	1	6	1	3	1	3	24	7	1

Figure 3-5 Response R4 in SD mode

Modified R1	C	Number of I/O functions	Memory Present	Stuff Bits	I/O OCR
8	1	3	1	3	24

Figure 3-6 Response R4 in SPI mode

The Response, R4 contains the following data:

S(tart bit):	Start bit. Always 0
D(irection):	Direction. Always 0. Indicates transfer from card to host.
Reserved:	Bits reserved for future use. These bits shall be set to 1.
C:	Set to 1 if Card is ready to operate after initialization
I/O OCR:	Operation Conditions Register. The supported minimum and maximum values for VDD. The layout of the OCR is shown in Table 3-1. See section 4.10.1 for additional information.
Memory Present:	Set to 1 if the card also contains SD memory. Set to 0 if the card is I/O only.
Number of I/O Functions:	Indicates the total number of I/O functions supported by this card. The range is 0-7. Note that the common area present on all I/O cards at Function 0 is not included in this count. The I/O functions shall be implemented sequentially beginning at function 1.
Modified R1:	The SPI R1 response byte as described in the SD Physical Specification modified for I/O as follows:

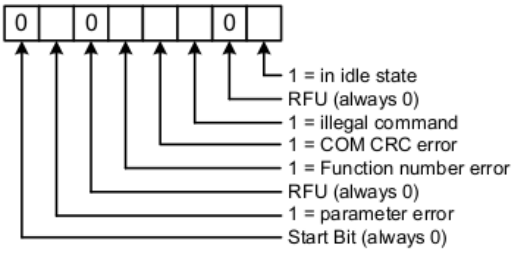


Figure 3-7 Modified R1 Response

5 Differences with SD Memory Specification

5.1 Unsupported SD Memory Commands

Several commands required for SD Memory cards are not supported by either SDIO-only cards or the I/O partion of Combo cards. Some of these commands have no use in SDIO cards such as Erase commands and thus are not supported in SDIO. In addition, there are several commands ofr SD memory cards that have different commands when used with the SDIO section of a card.

SD Memory Command	SDIO Command	Comment
CMD0	CMD52 (write to I/O reset in CCCR)	The reset command (CMD0) is only used for memory or the memory portion of Combo cards. In order to reset an I/O only card or the I/O portion of a combo card, use CMD52 to write a 1 to the RES bit in the CCCR (bit 3 of register 6). Note that in the SD mode, CMD0 is only used to indicate entry into SPI mode and shall be supported. An I/O only card or the I/O portion of a combo card is **not** reset with CMD0
CMD12	CMD52 (write to I/O abort)	In order to abort the block transfer of data, SD memory use CMD12. In order to abort an I/O transaction, use CMD52 to write to the abort register in the CCCR (bits 2:0 of register 6) See 4.8 for details.
CMD16	CMD52 (write to I/O Block Length)	CMD16 sets the block length for SD memory. In order to set the block length for each I/O function, use CMD52 to write the block length in the FBR
CMD2	NONE	The CID register does not exist in an SDIO only card
CMD4	NONE	The DSR register does not exist in an SDIO only card
CMD9	NONE	The CSD register does not exist in an SDIO only card
CMD10	NONE	The CID register does not exist in an SDIO only card
CMD13	NONE	An SDIO only card or the I/O portion of a combo card does not support the same SEND_STATUS (CMD13) protocol the SD memory uses. See 4.10.8.
ACMD6	CMD52 (write to Bus_Width [1:0] in CCCR)	SET_BUS_WIDTH is handled by a write to the CCCR. See 4.4 for details.
ACMD13	NONE	The SD Status register does not exist in an SDIO only card
ACMD41	CMD5	SDIO cards and hosts use the IO_SEND_OP_COND Command (CMD5). See 3.2
ACMD42	CMD52	In the SD mode, the pull-up resistor on DAT[3] is controlled by writing to the CD Disable bit in the CCCR. For Combo Cards, this resistor is enabled unless both the memory and the I/O control registers are set to disable the resistor. For details, see section 4.6
ACMD51	NONE	The SCR register does not exist in an SDIO only card
CMD17,	CMD53	I/O block operations use CMD53, rather than memory block

SD Memory Command	SDIO Command	Comment
CMD18, CMD24, CMD25		read/write commands.

Table 4-1 Unsupported SD Memory Commands

5.2 Modified R6 Response

The normal response to CMD3 by a memory card is R6.

Bit position	47	46	[45:40]	[39:8] Argument field		[7:1]	0
Width (bits)	1	1	6	16	16	7	1
Value	'0'	'0'	X	X	X	X	'1'
Description	Start bit	Direction bit	Command index ('000011')	New published RCA [31:16] of the card	[15:0] *Card status* (see Table 4-3)	CRC7	end bit

Table 4-2 R6 response to CMD3

Bits	Identifier	Type	Value	Description	Clear Condition
15	COM_CRC_ERROR	E R	'0'= no error '1'= error	The CRC check of the previous command failed	B
14	ILLEGAL_COMMAND	E R	'0'= no error '1'= error	Command not legal for the card state	B
13	ERROR	E R X	'0'= no error '1'= error	A general or an unknown error occurred during the operation	C
12: 0	***Undefined. Should read as 0 for SDIO only cards. Host should ignore these bits.***				

Note: Please refer to sections 7.3.4 of the SD Physical Specification for explanation of the entries in the *Type* and *Clear Condition* columns.

Table 4-3 SDIO R6 Status Bits

5.3 New I/O Read/Write Commands

5.3.1 $IO_{RWDIRECT}$ command (CMD52)

The $IO_{RW_{DIRECT}}$ is the simplest means to access a single register within the total 128K of register space in any I/O function, including the common I/O area (CIA). This command reads or wirtes 1 byte using only 1 command/response pair. A common use is to initialize registers or monitor status values for I/O functions. This command is the fastest means to read or write single I/O registers, as it requires only a sigle command/response pari.

S	D	Command Index 110100b	R/W flag	Function Number	RAW flag	Stuff	Register Address	Stuff	Write Data or Stuff Bits	CRC7	E
1	1	6	1	3	1	1	17	1	8	7	1

Figure 5-1 IO_RW_DIRECT Command

The IO_RW_DIRECT Command contains the following fields:

S(tart bit):	Start bit. Always 0
D(irection):	Direction. Always1 indicates transfer host to card.
Command Index:	Identifies the “IO_RW_DIRECT” command with a value of 110100b
R/W Flag:	This bit determines the direction of the I/O operation. If this bit is 0, this command shall read data from the SDIO card at the address specified by the Function Number and the Register Address to the host. The data byte is returned in the response, R5. If this bit is set to 1, the command shall write the bytes in the Write Data field to the I/O location addressed by the Function Number and the Register Address. If the RAW flag is 0, then the data in the register that was written shall be read and that value returned in the response.
RAW Flag:	The Read after Write flag. If this bit is set to 1 and the R/W flag is set to 1, then the command shall read the value of the register after the write. This is useful to allow writing to a control register and reading the status at the same address. If this bit is cleared, the value returned in the R5 response shall be the same as the write data in the command. If this bit is set, the data field of the R5 response shall contain the value read from the addressed register after the write operation.
Function Number:	The number of the function within the I/O card you wish to read or write. Note that function 0 selects the common I/O area (CIA).
Register Address:	This is the address of the byte of data inside of the selected function to read or write. There are 17 bits of address available so the register is located within the first 128K (131,072) addresses of that function.
Write Data/Stuff Bits:	For a direct write command (R/W=1), this is the byte that is written to the selected address. For a direct read (R/W=0), this field is not used and shall be set to 0.
CRC7:	7 bits of CRC data
E(nd bit):	End bit, always 1

5.3.2 $IO_{RWDIRECT}$ Response (R5)

The SDIO card's response to CMD52 shall be in one of two formats. If the communication between the card and host is in the 1-bit or 4-bit SD mode.

1. CDM52 Response (SD modes)

S	D	Command Index 110100b	Stuff	Response Flags Bit 7----------0	Read or Write Data	CRC7	E
1	1	6	16	8	8	7	1

Figure 5-2 R5 IO_RW_DIRECT Response (SD modes)

The IO_RW_DIRECT response (R5) contains the following fields:

S(tart bit): Start bit. Always 0
D(irection): Direction. 0 indicates transfer card to host (Response)
Command Index: Identifies the "IO_RW_DIRECT" command with a value of 110100b
Stuff Bits: Not used, shall be set to 0
Response Flags: 8 Bits of flag data indicating the status of the SDIO card. Table 5-1 shows the format of these flag bits.
Read or Write Data: For an I/O write (R/W=1) with the RAW Flag set (RAW=1) this field shall contain the value read from the addressed register after the write of the data contained in the command. Note that in this case, the read-back data may not be the same as the data written to the register, depending on the design of the hardware. For an I/O write with the RAW bit=0, the SDIO function shall not do a read after write operation, and the data in this field shall be identical to the data byte in the write command. For an I/O read (R/W=0), the actual value read from that I/O location is returned in this field.
CRC7: 7 bits of CRC data
E(nd bit): End bit, always 1

Bits	Identifier	Type	Value	Description	Clear Condition
7	COM_CRC_ERROR	E R	'0'= no error '1'= error	The CRC check of the previous command failed.	B
6	ILLEGAL_COMMAND	E R	'0'= no error '1'= error	Command not legal for the card State.	B

Bits	Identifier	Type	Value	Description	Clear Condition
5-4	IO_CURRENT_STATE	S	00=DIS 01=CMD 02=TRN 03=RFU	**DIS=Disabled:** Initialize, Standby and Inactive States (card not selected) **CMD=DAT lines free:** 1. Command waiting (No transaction suspended) 2. Command waiting (All CMD53 transactions suspended) 3. Executing CMD52 in CMD State **TRN=Transfer:** Command executing with data transfer using DAT[0] or DAT[3:0] lines	B
3	ERROR	E R E R X	'0'= no error '1'= error	A general or an unknown error occurred during the operation. Type "E R" shall be used for CMD52 Type "E R X" shall be used for CMD53	C
2	RFU	--	Fixed at 0	Reserved for Future Use	C
1	FUNCTION_NUMBER	E R	'0'= no error '1'= error	An invalid function number was requested	C
0	OUT_OF_RANGE	E R	'0'= no error '1'= error	The command's argument was out of the allowed range for this card.	C

Table 5-1 Flag data for IO_RW_DIRECT SD Response

2. R5, $IO_{RWDIRECT}$ Response (SPI mode)

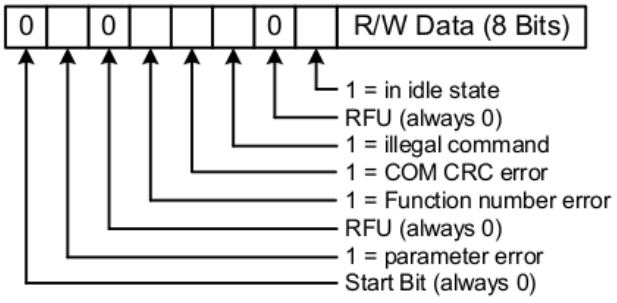


Figure 5-3 IO_RW_DIRECT Response in SPI Mode

5.3.3 $IO_{RWEXTENDED}$ Command (CMD53)

In order to read and write multiple I/O registers with a single command, a new command, $IO_{RWEXTENDED}$ is defined. This command allows the reading or writing of a large number of I/O registers with a single command.

S	D	Command Index 110101b	R/W flag	Function Number	Block Mode	OP Code	Register Address	Byte/Block Count	CRC 7	E
1	1	6	1	3	1	1	17	9	7	1

Figure 5-4 IO_RW_EXTENDED Command

The IO_RW_EXTENDED Command contains the following fields:

S(tart bit): Start bit. Always 0

D(irection): Direction. Always1 indicates transfer host to card.

Command Index: Identifies the "IO_RW_EXTENDED" command with a value of 110101b

R/W Flag: This bit determines the direction of the I/O operation. If this bit is 0, this command reads data from the SDIO card at the address specified by the Function Number and the Register Address to the host. The read data shall be returned on the DAT[x] lines. If this bit is set to 1, the command shall write the bytes from the DAT[x] lines to the I/O location addressed by the Function Number and the Register Address.

Function Number: The number of the function within the I/O card you wish to read or write. Note that function 0x00 selects the common I/O area (CIA).

Block Mode (Optional) this bit, if set to 1, indicates that the read or write operation shall be performed on a block basis, rather than the normal byte basis. If this bit is set, the Byte/Block count value shall contain the number of blocks to be read/written. The block size for functions 1-7 is set by writing the block size to the I/O block size register in the FBR (See Table 6-3 and Table 6-4). The block size for function 0 is set by writing to the FN0 Block Size register in the CCCR. Card and host support of the block I/O mode is optional. The host can determine if a card supports block I/O by reading the *Card supports MBIO* bit (SMB) in the CCCR (see Table 6-2). The block size used when Block Mode = 1 and the maximum byte count per command used when Block Mode = 0 can be read from the CIS in the tuple TPLFE_MAX_BLK_SIZE (see 16.7.4) on a per-function basis.

OP code Defines the read/write operation as described in Table 5-2

OP code	Command operation
0	Multi byte R/W to fixed address
1	Multi byte R/W to incrementing address

Table 5-2 IO_RW_ EXTENDED command Op Code Definition

- OP Code 0 is used to read or write multiple bytes of data to/from a single I/O register address. This command is useful when I/O data is transferred using a FIFO inside of the I/O card. In this case, multiple bytes of data are transferred to/from a single register address. For this operation, the address of the register is set into the Register Address field. Data is transferred on the DAT[0] or DAT[3:0] lines as defined for SD memory cards.
- OP Code 1 is used to read or write multiple bytes of data to/from an I/O register

address that increment by 1 after each operation. This command is used when large amounts of I/O data exist within the I/O card in a RAM like data buffer. In this operation, the start address is loaded into the Register Address field. The first operation occurs at that address within the I/O card. The next operation shall occur at address+1 with the address incrementing by 1 until the operation has completed. As with OP Code 0, the number of bytes is set in the Byte Count field of the command.

Register Address: Start Address of I/O register to read or write. Range is [0x1FFFF:0]

Byte/Block Count If the command is operating on bytes (Block Mode = 0), this field contains the number of bytes to read or write. A value of 0x000 shall cause 512 bytes to be read or written.

Count Value	0x000	0x001	0x002	---	0x1FF
Bytes Transferred	512	1	2		511
Block Transferred	∞	1	2		511

Table 5-3 Byte Count Values

If the command is in block mode (Block Mode=1), the Block Count field specifies the number of Data Blocks to be transferred following this command. A value of 0x000 indicates that the count set to infinite. In this case, the I/O blocks shall be transferred until the operation is aborted by writing to the I/O abort function select bits (ASx) in the CCCR (see Table 6-1 and Table 6-2). Table 5-3 shows the relationship between the value in the command and the actual number of bytes transferred.

CRC7: 7 bits of CRC data

E(nd bit): End bit, always 1

The response from the SDIO card to CMD53 shall be R5 (the same as CMD52) as defined in 5.2. For CMD53, the 8-bit data field shall be stuff bits and shall be read as 0x00. Also, the ERROR response bit shall be type "E R X" (see Table 5-1).

6 SDIO Card Internal Operation

6.1 Overview

Each SDIO card may have from 1 to 7 functions plus one memory function built into it.

6.2 Interrupts

All SDIO hosts should support hardware interrupts.

6.3 SDIO Fixed Internal Map

The SDIO card has a fixed internal register space and a function unique area. The fixed area contains information about the card and certain mandatory and optional registers in fixed locations. The fixed loactions allow any host to obtain information about the card and perform simple operations such as enable in a common manner. The function unique area is a per-function area, which is defined either by the Application Specifications for Standard SDIO functions or by the verdor for non-standard functions.

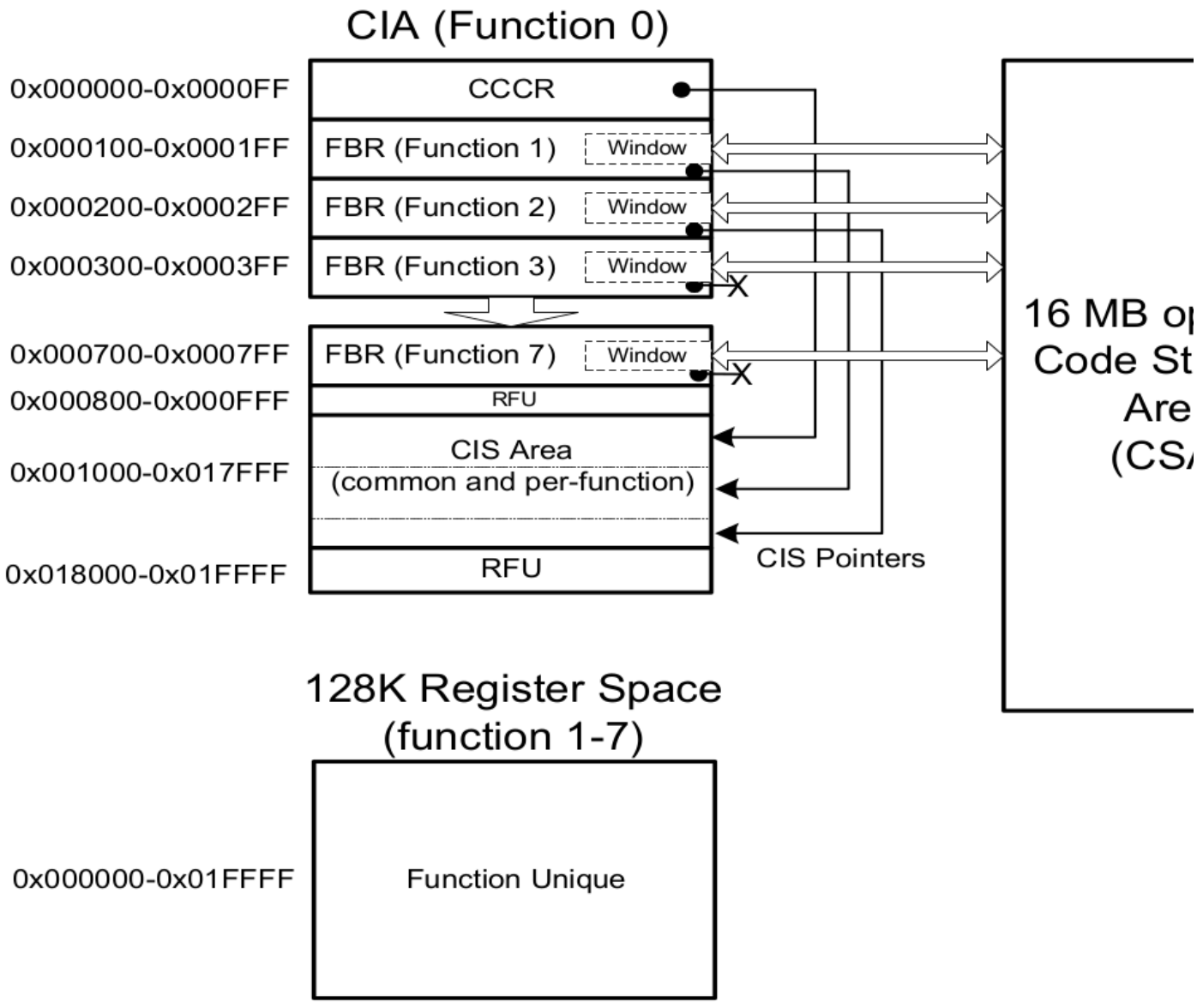


Figure 6-1 SDIO Internal Map

6.4 Common I/O Area (CIA)

The Common I/O Area (CIA) shall be implemented on all SDIO cards. The CIA is accessed by the host via I/O reads and writes to function 0. The registers within the CIA are provided to enable/disable the operation of the I/O function(s), control the generation of interrupts and optionally load softwre to support the I/O functions. The registers in the CIA also provide information about the function(s) ablilities and requirements. There are three distinct register structes supported within the CIA. They are:

1. Card Common Control Register (CCCR)
2. Function Basic Registers (FBR)
3. Card Information Structure (CIS)

Footnotes:

1

DEFINITION NOT FOUND.

Author: Yannik Li

Created: 2016-01-03 日 21:00

Emacs 24.5.1 (Org mode 8.2.10)

Validate