**数据包的封装与命令协议相关**

1 sd卡指令数据包

sd卡的指令被封装成48位的数据包，每次传送这48位的数据包。数据包的内容包括起始位、结束位、传输位、命令索引、传输参数和7位CRC校验码。其具体格式分布如下图

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Bit 位置 | 47 | 46 | [45:40] | [39:08] | [07:01] | 00 |
| Bit 宽度 | 1 | 1 | 6 | 32 | 7 | 1 |
| 值 | “0” | “1” | x | x | x | “1” |
| 说明 | Start bit | Transmission  bit | Command  index | Argument | CRC7 | End bit |

其中的命令索引位是[45：40]，里面可以封装各种命令，具体的命令表将在下面给出。不同的命令会对应不同的回应(respond)，回应有三种(R1，R2，R3)格式，在命令表中的选项会给出。

2 sd卡命令索引表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CMD | 简略语 | 指令说明 | SPI 模式 | |
| 自变量 | 回应 |
| 0 | GO\_IDLE\_STATE | 这是使card初始化到Idle状态的指令.CS信号设在Low的状态时,接到本指令后,card 将转换到SPI模式. | None | R1 |
| 1 | SEND\_OP\_COND | 接到本指令后,card将做R3回应(含有OCR 数据).根据OCR值,可以得知card能工作电压范围.OCR数据最高值位的1bit是用来确认card内部处理是否结束(Ready/Busy轮询). | None | R1 |
| 2 | ALL\_SEND\_CID | 接到本指令后, 处于Ready状态的card将传送CID数据.在MMC模式下,数据被送到CMD信号,在CID数据的每1bit传送后,CMD信号状态将与该card内部状态相比较,如果不一致,將中止数据传送,card返回到Ready状态.如果相一致,该card 将认为已被选中,然后转换到Identification 状态. |  |  |
| 3 | SET\_RELATIVE\_ADDR | 本指令会为已转换到Identification状态的card分配一个相对card地址(RCA).当RCA分配后,card将转换到Stand-by 状态,对以后的CMD2和CMD3不回应. |  |  |
| 4 | NOP | 这是用来设定DSR(DriveState寄存器)的指令,但是本car不支持DSR. |  |  |
| 7 | SELECT/DESELECT\_CARD | 本指令是用来选择一张card,让它在Stand-by状态和Transfer状态之间转换的指令.如果给card设定已分配到的RCA地址,card将从Stand-by状态转换到Transfer状态,并将回应以后的读取指令及其他指令.如果给card设定RCA以外的地址,card将转换到Stand-by状态。当RCA=0000h时,card将无条件地转换到Stand-by状态. |  |  |
| 9 | SEND\_CSD | 接到本指令后,将传送CSD数据. | None | R1 |
| 10 | SEND\_CID | 接到本指令后,将传送CID数据. | None | R1 |
| 11 | READ\_DAT\_UNTIL\_STOP | 接到本指令后,将从设定的地址传送P2ROM数据,直到接到指令CMD12为止. |  |  |
| 12 | STOP \_TRANSMISSION | 本指令强行终止CMD11和CMD18的处理 | None | R1 |
| 13 | SEND\_STATUS | 接到本指令后, 将传送状态寄存器的信息. | None | R2 |
| 15 | GO\_INACTIVE \_STATE | 接到本指令后,将转换到休止(inactive)状态. |  |  |
| 16 | SET\_BLOCKLEN | 本指令用来设定Block长度.对象是以后的指令CMD17和CMD18. | 31:00]  block  length | R1 |
| 17 | READ\_SINGLE \_BLOCK | 接到本指令后,将从自变量设定的地址传送1个block 长度的数据.(block长度由指令CMD16设定). | [31:00]  data  address | R1 |
| 18 | READ\_MULTIPLE \_BLOCK | 接到本指令后,将从自变量设定的地址连续传送block长度的数据,直到接到指令CMD12为止.(block长度由指令CMD16设定). | [31:00]  data  address | R1 |
| 23 | SET\_BLOCK \_COUNT | 本指令是给紧跟的指令CMD18设定要传送的block 数量. | [31:16]set to 0  [15:00]number  of block | R1 |
| 24 | WRITE\_BLOCK | 接到本指令后,将写1个block 长度的数据到自变量设定的地址.(block长度由指令CMD16设定). | [31:0] data address | R1 |
| 25 | WRITE\_MULTIPLE\_  BLOCK | 接到本指令后,将连续写block长度的数据到自变量设定的地址,直到接到指令CMD12为止.(block长度由指令CMD16设定). | [31:0] data address | R1 |
| 58 | READ\_OCR | 接到本指令后,Card将传送OCR数据. | None | R3 |
| 59 | CRC\_ON\_OFF | 本指令是用来设定CRC 选项为ON或OFF.在SPI模式下,CRC的初始值设定为OFF.[CRC选项[bit=1]表示CRC ON;[CRC选项[bit=0]表示CRC OFF. | [31:01]  stuff bits  [00:00]  CRC  option | R1 |

3 命令回应repond

1)        R1模式

对象指令

CMD0 ： GO\_IDLE\_STATE

CMD1 ： SEND\_OP\_COND

CMD9 ： SEND\_CSD

CMD10：SEND\_CID

CMD12：STOP\_TRANSMISSION

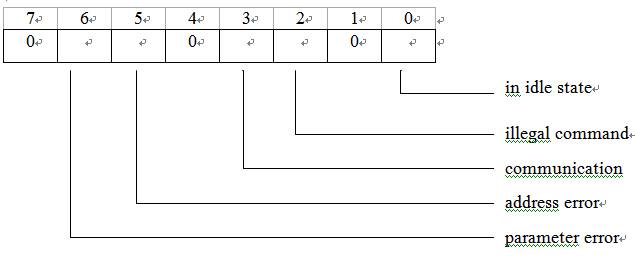
CMD16：SET\_BLOCKLEN

CMD17：READ\_SINGLE\_BLOCK

CMD18：READ\_MULTIPLE\_BLOCK

CMD23：SET\_BLOCK\_COUNT

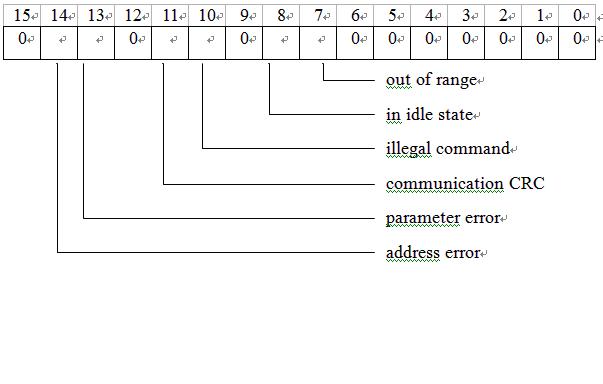
CMD59：CRC\_ON\_OFF



2)        R2模式

对象指令

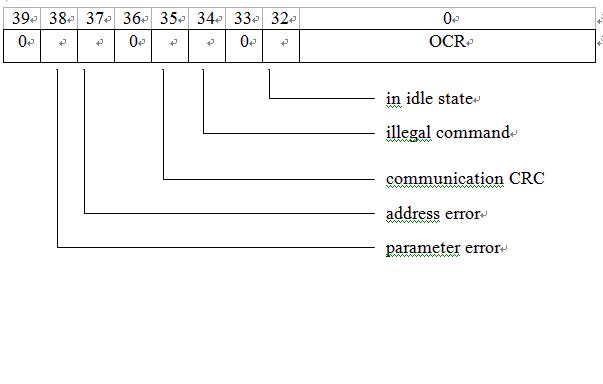
CMD13：SEND\_STATUS



3)        R3模式

对象指令

CMD58：READ\_OCR



4. sd卡的配置信息

4.1卡识别寄存器——CID(card identification register )

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CID-slice | 名称 | 栏目 | Bit宽度 | 值 | OKI 原创 |
| [127:120] | Manufacture ID | MID | 8 | 0x41 | OKI ID, 由MMCA 指定 |
| [119:104] | OEM/Application ID | OID | 16 | 0x0000 | OK设备名 |
| [103:56] | Product name | PNM | 48 | P2016 | 产品名 |
| [55:48] | Product revision | PRV | 8 | 0x10 | 产品修订版(version 1.0) |
| [47:16] | Product serial number | PSN | 32 | 0x00000001 | 产品编号  Default=1, user设定可 |
| [15:8] | Manufacturing data | MDT | 8 | 收到Code 文件的月份 | OKI 收到客户Code 档案的日期,或是写入程序的日期 |
| [7:1] | 7-bit CRC checksum | CRC7 | 7 | CRC | 检验和 |
| [0:0] | - | - | 1 | 1 | 固定是”1” |

4.2卡特性寄存器——CSD(card specific data register)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| CSD-slice | 名称 | 栏目 | Bit 宽度 | 值 | | Type | 备注 |
| Bin. | Hex. |  |  |
| [127:126] | CSD Structure | CSD\_STRUCTURE | 2 | 10 | 0x2 | R | Version 3.1 |
| [125:122] | spec version | SEPC\_VERS | 4 | 0011 | 0x3 | R | Version 3.1 |
| [121:120] | Reserved |  | 2 | X | X | R | 不必理会 |
| [119:112] | datatime1 read access | TAAC | 8 | 00001000 | 0x08 | R | 1ns |
| [111:104] | datatime2 read access | NSAC | 8 | 00000001 | 0x01 | R | 100cycles |
| [103:96] | max rate data transfer | TRAN\_SPEED | 8 | 00101010 | 0x2A | R | 20Mb/s |
| [95:84] | card command class | CCC | 12 | 000000000111 | 0x007 | R | 支持Class0,1,2 |
| [83:80] | max read data block  length | READ\_BL\_LEN | 4 | 1001 | 0x9 | R | 512 字节 |
| 79 | partial blocks for read  allowed | READ\_BL\_PARTIAL | 1 | 1 | 0x1 | R |  |
| 78 | Reserved |  | 1 | X | X | R | 不必理会 |
| 77 | read block misalignment | READ\_BLK\_MISALIGN | 1 | 0 | 0x0 | R | 禁止跨越Block |
| 76 | DSR implemented | DSR\_IMP | 1 | 0 | 0x0 | R | DSR: 不支持 |
| [75:74] | Reserved |  | 2 | X | X | R | 不必理会 |
| [73:62] | device size | C\_CIZE | 12 | 111111111110 | 0xFFE | R |  |
| [61:59] | min. read current | VDD\_R\_CURR\_MIN | 3 | 000 | 0x0 | R | 0.5mA |
| [58:56] | max. read current | VDD\_R\_CURR\_MAX | 3 | 100 | 0x4 | R | 35mA |
| [55:50] | Reserved |  | 6 | X | X | R | 不必理会 |
| [49:47] | device size multiplier | C\_SIZE\_MULT | 3 | 001 | 0x1 | R |  |
| [46:31] | Reserved |  | 16 | X | X | R | 不必理会 |
| [30:29] | manufacture default ECC | DEFAULT\_ECC | 2 | 00 | 0x0 | R | ECC: 不支持 |
| [28:26] | Reserved |  | 3 | X | X | R | 不必理会 |
| [25:22] | Max. write data block  length | WRITE\_BL\_LEN | 4 | 1001 | 0x9 | R |  |
| [21:16] | Reserved |  | 6 | X | X | R | 不必理会 |
| 15 | file format group | FILE\_FORMAT\_GRP | 1 | 0 | 0x0 | R |  |
| 14 | Copy flag (OTP) | COPY | 1 | 0 | 0x0 | R | Default=0 User 设定可 |
| 13 | permanent protection  write | PERM\_WRITE\_ PROTECT | 1 | 1 | 0x1 | R |  |
| 12 | temporary protection   write | TMP\_WRITE\_ PROTECT | 1 | 1 | 0x1 | R |  |
| [11:10] | file format | FILE\_FORMAT | 2 | 00 | 0x0 | R | Default=0 User 设定可 |
| [9:8] | ECC code | ECC | 2 | 00 | 0x0 | R | ECC: 不支持 |
| [7:1] | CRC | CRC | 7 | CRC | CRC | R |  |
| 0 | Reserved |  | 1 | 1 | 0x1 | R |  |