## SD卡物理层协议

### SD卡系统

#### 1.1 SD卡读写属性

**SD卡分类：**

* 读写(RW)卡(一次性可编程OTP，多次可编程MTP)：这些卡通常作为空白(空)媒体出售，用于大量数据存储，最终用户视频、音频或数字图像记录。这些卡片是用固定的数据内容制造的。它们通常被用作软件、音频、视频等的分发媒介；
* 只读存储器(ROM)卡：这些卡片是固定的数据内容；通常被用作软件、音频、视频等的分发媒介；

工作电源电压方面：

* 高压SD存储卡，可以在2.7-3.6 V电压范围内工作；
* 支持工作电压范围为2.7 ~ 3.6V或1.70 ~ 1.95V的UHS-II SD存储卡；
* 支持工作电压范围为2.7 ~ 3.6V或1.70 ~ 1.95V的LVS卡，支持工作电压范围为的LVS卡；

#### 1.2 卡的容量

SD存储卡有两个可访问的独立区域：用户区和受保护区域。用户区域是主存区域，保护区域可以通过第三部分安全规范定义的身份验证访问。卡容量指用户区域容量和保护区域容量之和。

卡容量：

* 标准容量SD存储卡(SDSC)支持的容量不超过2GB；
* 高容量SD存储卡(SDHC)支持容量大于2GB，最大32GB；
* 扩展容量SD存储卡(SDXC)支持32GB，最大2TB；

SD卡速度等级：

* 第0类卡不指定性能；包括物理层规范版本2.00之前的所有传统卡，无论其性能如何；
* 第2类大于或等于2MB/秒的性能(默认速度模式)；
* 第4类大于等于4mb /秒 (默认速度模式)；
* 第6类大于等于6MB/秒(默认速度模式)；
* 第10类大于等于10MB /秒(高速模式)；

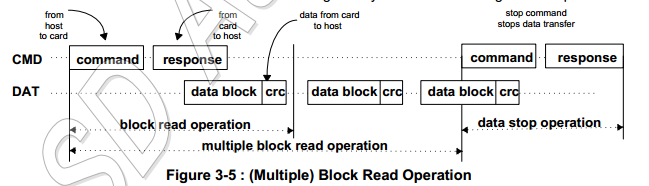
#### 1.3 总线协议

SD总线上的通信基于命令和数据比特流，这些比特流由起始位发起，由停止位终止。

* 命令：命令是启动一个操作的令牌。命令从主机发送到单个卡(寻址命令)或发送到所有连接的卡(广播命令)。命令在CMD行上串行传输；
* 响应：响应是一个令牌，从一个地址卡发送，或者(同步地)从所有连接的卡发送给主机，作为对先前收到的命令的回答。响应在CMD行上串行传输；
* 数据：数据可以从卡传输到主机，反之亦然。数据通过数据线传输；

卡寻址是使用会话地址实现的，会话地址在初始化阶段分配给卡。SD总线上的基本事务是命令响应事务。这种类型的总线事务直接在命令或响应结构中传输信息。此外，一些操作具有数据令牌。

数据传输到或从SD存储卡是在块完成。数据块总是由CRC位校验。定义了单个和多个块操作。注意，多块操作模式更适合于更快的写操作。当CMD行上跟随一个停止命令时，多块传输被终止。数据传输可以由主机配置为使用单个或多个数据线。



每个命令令牌前有一个起始位(0)，后有一个结束位(1)，总长度为48位。每个令牌都由CRC位保护，这样可以检测到传输错误，并可以重复操作。

响应令牌有四种编码方案之一，具体取决于内容。令牌长度为48位或136位。块数据的CRC保护算法是一个16位的CCITT多项式。

在CMD行中，首先传输最高有效位(MSB)，最后传输最低有效位(LSB)。

当使用宽总线选项时，数据每次传输4位。开始和结束位以及CRC位，在每一条DAT线路传输。每个数据行单独计算和检查CRC位。CRC状态响应和忙指示将由卡发送到主机上的DAT0 (DAT1-DAT3在此期间被忽略)。

#### 1.4 SD引脚

* CD(DAT3)：SD卡模式主要作为片选和数据线；SPI模式作为片选线；
* CMD：SD卡模式作为命令和响应；而SPI模式作为数据输入；
* VSS：地面电源电压；
* VDD：电源电压；
* CLK：时钟信号线；
* VSS2：地面电源电压；
* DAT0：SD卡模式作为数据线；SPI模式作为数据输出；
* DAT1：SD卡模式作为数据线；SPI模式保留；
* DAT2：SD卡模式作为数据线；SPI模式保留；

主机可以通过关闭和重新打开电源来复位卡。每个卡都有自己的上电检测电路，使卡在上电后进入规定的状态。不需要显式的复位信号。卡也可以通过发送GO\_IDLE (CMD0)命令来重置。

### 2 SD卡功能

#### 2.1 概述

主机和卡之间的所有通信都由主机控制。主机发送两种类型的命令：广播和寻址(点对点)命令。

* 广播命令：广播命令适用于所有卡。其中一些命令需要响应；
* 寻址(点对点)命令：寻址命令被发送到寻址卡，并引起该卡的响应；

两种操作模式(主机和卡)：

* 卡识别模式：主机在复位后和在总线上寻找新卡时将处于卡识别模式。在收到SEND\_RCA命令(CMD3)之前，复位后卡将处于此模式；
* 数据传输模式：在RCA首次发布后，卡将进入数据传输模式。主机识别出总线上的所有卡后，进入数据传输模式；

#### 2.2 卡识别模式

在卡识别模式下，主机重置所有处于卡识别模式的卡，验证操作电压范围，识别卡并要求发布相对卡地址(RCA)。该操作在每个卡各自的CMD行上分别完成。卡识别模式下的所有数据通信仅使用命令行(CMD)。

在卡识别过程中，卡必须在识别时钟频率fOD的SD时钟频率下工作。

##### 2.2.1 卡重启

在SD模式下，GO\_IDLE\_STATE(CMD0)命令是软件复位命令，无论当前卡的状态如何，都将每个卡设置为空闲状态。处于非活动状态的卡不受此命令的影响。

主机上电后，所有卡的状态为空闲，包括之前为非活跃状态的卡。上电或CMD0后，所有卡的CMD都处于输入模式，等待下一个命令的起始位。这些卡初始化为默认的相对卡地址(RCA=0x0000)，并具有400KHz时钟频率的默认驱动程序强度。在3.3V信号的情况下，如果支持并选择最高驱动电流能力，则默认驱动强度由驱动级寄存器(DSR)指定。

##### 2.2.2 运行情况检测

在主机和卡开始通信时，主机可能不知道卡支持的电压，卡也可能不知道自己是否支持当前供电的电压。主机发出一个带有指定电压的重置命令(CMD0)，同时假定卡可能支持。

SEND\_IF\_COND (CMD8)用于校验SD卡接口的工作状态。卡通过分析CMD8的参数来检查运行状态的有效性，主机通过分析CMD8的响应来检查有效性。供电电压由参数中的VHS字段表示。卡假定VHS中规定的电压为当前供电电压。在任何给定时间，VHS中只有1位被设置为1。主机使用CRC和校验模式来检查主机和卡之间通信的有效性。

如果卡可以在提供的电压上操作，则响应回显在命令参数中设置的提供电压和检查模式。如果卡不能在提供的电压下工作，不返回响应并保持在空闲状态。在初始化SDHC或SDXC卡时，必须在第一个ACMD41之前发布CMD8。

SD\_SEND\_OP\_COND (ACMD41)旨在为SD存储卡主机提供一种机制，以识别和拒绝与主机所需的VDD范围不匹配的卡。这是通过主机发送所需的VDD电压窗口作为该命令的操作数来完成的。

不能在规定范围内进行数据传输的卡应从进一步的总线操作中丢弃自己，并进入非活动状态。OCR寄存器中的电平应相应地定义。注意，ACMD41是特定于应用程序的命令；因此APP\_CMD (CMD55)总是在ACMD41之前。空闲状态中CMD55使用的RCA应该是卡的默认RCA。

主机发出复位命令(CMD0)复位卡后，主机应在ACMD41之前发出CMD8重新初始化SD存储卡。

通过在ACMD41参数中设置OCR为零，主机可以查询每个卡，并确定公共电压范围，然后将超出范围的卡发送到非活跃状态。如果主机能够选择一个公共电压范围，或者如果需要通知堆栈中不可用的卡的应用程序，则应使用此查询。如果ACMD41作为查询发出，则卡不会开始初始化并忽略参数中的HCS。之后，主机可以选择一个电压进行操作，并在此条件下重新发布ACMD41，使不兼容的卡进入非活跃状态。

在初始化过程中，不允许主机改变工作电压范围。

#### 2.3 数据传输模式

在卡识别模式结束之前，主机应保持在fOD频率，因为在卡识别模式中，某些卡可能有工作频率限制。在数据传输模式下，主机可以在fPP频率范围内操作卡。主机发出SEND\_CSD (CMD9)来获取卡片特定数据(CSD寄存器)，例如块长度，卡片存储容量等。

广播命令SET\_DSR(CMD4)配置所有标识卡的驱动阶段。根据应用总线布局(长度)和总线上卡的数量以及数据传输频率对DSR寄存器进行编程。时钟速率也从fOD切换到fPP。SET\_DSR命令是卡和主机的选项。

CMD7用于选择一张卡并将其置于传输状态。在给定的时间内，只能有一张卡处于转移状态。如果先前选择的卡处于转移状态，则与主机的连接将被释放，并且将移回备用状态。当CMD7被授予保留的相对卡地址0时，所有卡都回到待机状态。

这可以在识别新卡之前使用，而无需重置其他已注册的卡。在这种状态下，已经有RCA的卡不会响应识别命令。CMD48、CMD49、CMD58和CMD59也可在UHS-II模式下使用。

* 所有的数据读取命令都可以在任何时候通过停止命令(CMD12)终止。数据传输将终止，卡将返回到传输状态。读命令有块读(CMD17)、多块读(CMD18)、发送写保护(CMD30)、发送SCR (ACMD51)和读模式通用命令(CMD56)；
* 所有的数据写命令都可以通过停止命令(CMD12)随时终止。在CMD7取消选择卡之前，应该停止写命令。写命令包括块写命令(CMD24和CMD25)、程序写命令(CMD27)、锁/解锁命令(CMD42)和写模式通用命令(CMD56)；
* 数据传输完成后，卡将退出数据写入状态，并进入编程状态(传输成功)或传输状态(传输失败)；
* 如果停止写块操作，并且最后一块的块长度和CRC有效，则将对数据进行编程；
* 该卡可以为块写入提供缓冲。这意味着下一个数据块可以在前一个数据块被编程时被发送到卡片上。如果所有写缓冲区都满了，并且只要卡处于编程状态，DAT0线将保持低(BUSY)；
* 写CSD、写保护和擦除没有缓冲选项。这意味着，当卡忙于服务这些命令中的任何一个时，将不接受其他数据传输命令。DAT0线将保持低，只要卡是忙的，并在编程状态。实际上，如果卡的CMD和DAT0线保持分开，主机保持忙DAT0线与其他DAT0线断开连接(其他卡的)，主机可以在卡忙时访问其他卡；
* 当卡正在编程时，不允许参数设置命令；参数设置命令有：设置块长度(CMD16)、擦除块开始(CMD32)和擦除块结束(CMD33)；
* 当卡正在编程时，不允许读取命令；
* 将另一张卡从待机状态移动到传输状态(使用CMD7)不会终止擦除和编程操作。卡将切换到断开状态，并将释放DAT线；
* 在断开状态下，可以使用CMD7重新选择卡。在这种情况下，卡将移动到编程状态并重新激活忙指示；
* 复位卡(使用CMD0或CMD15)将终止任何未决或正在进行的编程操作。这可能会破坏卡上的数据内容。主人有责任防止这种情况发生；
* CMD34-37、CMD50、CMD57预留用于SD指挥系统扩展。这些命令的状态转换在每个命令系统规范中定义；

##### 2.3.1 总线位宽选择

宽总线(4位总线宽度)操作模式可以使用ACMD6选择/取消选择。上电或GO\_IDLE (CMD0)后的默认总线宽度为1位总线宽度。为了改变总线宽度，需要卡处于传输状态并且卡未被锁定，被锁定的卡将响应ACMD6为非法命令。

要制作2G卡，最大块长度(READ\_BL\_LEN=WRITE\_BL\_LEN)应该设置为1024字节。然而，CMD16设置的块长度应不超过512字节，以保持与512字节最大块长度卡的一致性。

##### 2.3.2 数据读取

当没有数据传输时，通过上拉使数据总线电平高。传输的数据块由起始位(低1位或低4位)和连续数据流组成。数据流包含有效载荷数据(如果使用离卡ECC，则包含纠错位)。数据流以结束位(1或4位高电平)结束。数据传输与时钟信号同步。面向块的数据传输的有效载荷由1位或4位CRC校验和保护。

关闭电源可能会中断SD存储卡的读取操作。SD存储卡确保在主机发出的除写或擦除操作外的所有情况下，即使在突然关闭或移除的情况下，数据也不会被破坏。

如果发生BLOCK\_LEN\_ERROR或ADDRESS\_ERROR，并且没有进行数据传输，则拒绝读取命令。

块读取是面向块的数据传输。数据传输的基本单位是一个块，其最大大小始终为512字节。较小的块，其起始和结束地址完全包含在512字节边界内，可以传输。CMD16设置的块长度可以设置为512字节，与READ\_BL\_LEN无关。

CRC被附加到每个块的末尾，以确保数据传输的完整性。CMD17 (READ\_SINGLE\_BLOCK)发起一个块读取，完成传输后，卡返回到传输状态。CMD18 (READ\_MULTIPLE\_BLOCK)开始传输几个连续的块。

块将持续传输，直到发出STOP\_TRANSMISSION命令(CMD12)。由于串行命令传输，停止命令有一个执行延迟。数据传输在停止命令结束位之后停止。当使用CMD18读取用户区的最后一块时，即使顺序正确，主机也应该忽略可能发生的OUT\_OF\_RANGE错误。

如果主机使用的部分块的累积长度不是块对齐的，并且不允许块错位，卡将在第一个错位块的开始处检测到块错位，在状态寄存器中设置ADDRESS\_ERROR错误位，中止传输，并在数据状态中等待停止命令。

##### 2.3.3 数据写入

数据传输格式与数据读取格式类似。对于面向块的写数据传输，CRC校验位被添加到每个数据块中。在写操作之前，卡对每个接收到的数据块执行1或4位CRC奇偶校验。通过这种机制，可以防止写入错误传输的数据。

如果发生BLOCK\_LEN\_ERROR或ADDRESS\_ERROR，并且没有进行数据传输，则拒绝写命令。

在块写入(CMD24)期间，一个或多个数据块从主机传输到卡，主机在每个块的末尾附加1或4位CRC。无论WRITE\_BL\_LEN设置为1k还是2k字节，支持块写的卡都要求CMD16设置的块长度为512字节。

如果WRITE\_BL\_PARTIAL是允许的，那么更小的块，最多一个字节的分辨率，也可以使用。如果CRC失败，卡应在DAT线上指示失败；传输的数据将被丢弃而不被写入，所有后续传输的块(多块写入模式)将被丢弃。为了提高写操作的速度，建议使用多个块写命令，而不是连续的单个写命令。

如果主机使用的部分块的累计长度没有块对齐，并且不允许块错位(CSD参数WRITE\_BLK\_MISALIGN未设置)，则卡应在第一个错位块开始之前检测到块错位错误并中止传输。卡应该在状态寄存器中设置ADDRESS\_ERROR错误位，同时忽略所有进一步的数据传输，在接收数据状态中等待停止命令。

注意，第一个数据块对于写命令是不对齐的(即在写命令的实际响应中报告ADDRESS\_ERROR)，卡保持在传说状态，没有数据被编程。如果主机试图在写保护区域上写，写操作也会被终止。然而，在这种情况下，卡应该设置WP\_VIOLATION位。

CSD寄存器的编程不需要先前的块长度设置。传输的数据也受CRC保护。如果CSD寄存器的一部分存储在ROM中，那么这个不可改变的部分应与接收缓冲区的相应部分相匹配。如果匹配失败，那么卡将报告一个错误，并且不会更改任何寄存器内容。

有些卡可能需要很长且不可预测的时间来写入数据块。在接收到数据块并完成CRC检查后，如果写缓冲区已满并且无法接受来自新WRITE\_BLOCK命令的新数据，则卡将开始写入并保持DAT0低电平。主机可以在任何时候用SEND\_STATUS命令(CMD13)轮询卡片的状态，卡片将用状态进行响应。状态位READY\_FOR\_DATA表示卡是否可以接受新数据，或者写过程是否仍在进行中。主机可以通过发出CMD7(选择不同的卡)来取消卡的选择，这将使卡进入断开状态并释放数据线而不中断写操作。当重新选择卡时，如果编程仍在进行中并且写缓冲区不可用，将通过将DAT拉到低来重新激活忙指示。实际上，主机可以同时对多张卡进行写操作，并有互留过程。交错过程可以通过在其他卡忙时单独访问每个卡来完成。这个过程可以通过适当的CMD和DAT0-3线操作(断开忙卡)来完成。

设置预擦除的写块数量(ACMD23)将使后续的多块写操作比不使用ACMD23的相同操作更快。主机将使用这个命令来定义在下一次写操作中要发送多少个写块。如果主机将在所有数据块发送到卡之前终止写操作(使用停止传输)，则剩余写块的内容是未定义的(可以被擦除或仍然有旧数据)。

如果主机发送的写块数量超过ACMD23中定义的数量，卡将逐个擦除块(当接收到新数据时)。多块写操作完成后，该数值将复位为默认值。

建议在CMD25之前使用此命令，有些卡对于多写块操作会更快。注意，如果主机想要使用预擦除特性，则应该在WRITE命令之前发送ACMD23。否则，在执行其他命令时，可能会自动清除预擦去。

使用管道机制进行数据缓冲区管理的系统，在某些情况下，如果在多块写入操作中发生错误，则无法确定哪个块是最后一个被写入闪存的。卡将用写入良好的块的数量响应ACMD22。

##### 2.3.4 擦除和填充

为了提高数据吞吐量，需要同时擦除多个写块。这些写块的识别是通过ERASE\_WR\_BLK\_START (CMD32)、ERASE\_WR\_BLK\_END (CMD33)命令完成的。

如果在CMD38参数中设置了0x1h或0x2h，则如果卡支持丢弃或填充，则可能不会执行擦除。如果收到顺序错误的擦除(CMD38)或地址设置(CMD32, 33)命令，卡将在状态寄存器中设置ERASE\_SEQ\_ERROR位，并重置整个序列。

如果收到乱序命令(SEND\_STATUS除外)，卡将在状态寄存器中设置ERASE\_RESET状态位，重置擦除顺序，执行最后一条命令。如果擦除范围包括写保护扇区，则保留写保护扇区，只擦除非保护扇区。应该设置状态寄存器中的WP\_ERASE\_SKIP状态位。

地址设置命令中的地址字段是以字节为单位的写块地址。卡将忽略WRITE\_BL\_LEN(参见CSD)大小以下的所有LSB。

丢弃操作类似于缺省的擦除操作。卡可以部分或全部取消分配丢弃的块。已应用丢弃函数的区域的内容将被主机视为忽略。在丢弃操作之后，先前写入的数据可以由主机部分或全部读取，这取决于卡的实现。为了提高数据吞吐量，需要同时丢弃多个写块。

这些写块的识别是通过ERASE\_WR\_BLK\_START (CMD32)、ERASE\_WR\_BLK\_END (CMD33)命令完成的。支持丢弃的主机应该检查SD\_STATUS寄存器中的DISCARD\_SUPPORT位(b313)。如果卡不支持丢弃，主机不应该发出丢弃命令。如果卡不支持丢弃，卡将执行ERASE。

如果收到的丢弃 (CMD38)或地址设置(CMD32, 33)命令顺序错了，卡将在状态寄存器中设置ERASE\_SEQ\_ERROR位，并复位整个序列。

如果收到乱序命令(SEND\_STATUS除外)，卡将在状态寄存器中设置ERASE\_RESET状态位，复位DISCARD顺序，执行最后一条命令。

如果主机提供超出范围的地址作为CMD32或CMD33的参数，卡将在R1 (ERX)中指示CMD38或下一条命令的R1中的OUT\_OF\_RANGE错误，并重置整个丢弃序列。在执行DISCARD命令后，主机将检查是否有错误。

FULE操作类似于默认的擦除操作，除了卡必须在逻辑上完全擦除整个用户区域。需要在逻辑上擦除整个用户区。完整用户区域的识别是通过参数等于0的ERASE\_WR\_BLK\_START (CMD32)和参数等于用户区域结束的ERASE\_WR\_BLK\_END (CMD33)命令完成的。

支持填充的主机应检查SD\_STATUS寄存器中是否有FULE\_SUPPORT位。如果卡不支持FULE，主机将不会发出FULE命令。如果卡不支持填充，卡将执行ERASE。

如果收到的填充(CMD38)或地址 (CMD32, 33)命令顺序错了，卡将在状态寄存器中设置ERASE\_SEQ\_ERROR位，并复位整个序列。如果收到乱序命令(SEND\_STATUS除外)，卡将在状态寄存器中设置ERASE\_RESET状态位，复位填充序列，执行最后一条命令。

##### 2.3.5 卡写保护管理

SD存储卡支持三种写保护方式：机械写保护开关(仅主机负责)；卡内部写保护(卡负责)；密码保护卡锁操作。

用户将使用卡侧面的机械滑动板来指示给定卡是否受写保护。如果滑动平板被放置在这样一种方式，窗口是打开的，这意味着卡片是写保护。如果窗口关闭，则卡不受写保护。插座侧的一个合适的、匹配的开关将向主机指示卡是否为写保护。主机有责任保护卡的安全。卡的内部电路不知道写保护开关的位置。

##### 2.3.6 卡锁和解锁操作

密码保护功能使主机能够在提供密码的同时锁定卡片，稍后将用于解锁卡片。密码及其大小分别保存在128位PWD和8位PWD\_LEN寄存器中。可选地，强制擦除密码可以存储在额外的128位FEP中，如果卡支持卡所有权保护(COP)，则应存在8位FEP\_LEN寄存器。这些寄存器是非易失性的，因此电源周期不会擦除它们。

锁定卡响应并执行命令类(0类)、ACMD41、CMD16和锁定卡命令类中的所有命令。因此，允许主机进行复位、初始化、选择、查询状态等操作，但不允许访问卡上的数据。如果之前设置过密码(PWD\_LEN的值不为0)，上电后卡会自动锁定。

与现有的CSD寄存器写命令类似，锁命令只能在传输状态下使用。这意味着不包括地址参数，卡必须在使用前进行选择。

锁命令具有普通单块写命令的结构和总线事务类型。传输的数据块包括命令所需的所有信息(密码设置模式、PWD本身、卡锁/解锁等)。内容保护由PWD管理。物理卡所有权是用COP特性管理的。

#### 2.4 时钟控制

主机可以使用SD卡总线时钟信号将卡切换到节能模式或控制总线上的数据流。允许主机降低时钟频率或关闭主机。因此，为了保持数据的连续传输，从卡的角度来看，到卡的时钟应该在第一个512字节之后停止。然后，主机将用另一个512字节填充其内部缓冲区。在主机中写块的后半部分准备好后，将通过重新启动时钟供应继续向卡传输数据。这样，卡就不能识别数据传输中的任何中断。

主机需要考虑以下几点限制：

* 总线频率可以随时更改(在最大数据传输频率和规范文件定义的识别频率的限制下)；

#### 2.5 CRC校验码

CRC旨在保护SD存储卡命令、响应和数据传输免受SD存储卡总线上传输错误的影响。为每个命令生成一个CRC，并检查CMD行上的每个响应。对于数据块，每个传输的块生成一个CRC。

#### 2.6 错误条件

所有命令都由CRC(循环冗余校验)位保护。如果地址卡的CRC检查失败，则该卡不响应，命令不执行。卡不改变状态，状态寄存器设置COM\_CRC\_ERROR位。同样，如果接收到非法命令，卡将不改变其状态，不响应，并在状态寄存器中设置ILLEGAL\_COMMAND错误位。

非法命令有以下几种：

* 属于卡不支持的类的命令(例如只读卡中的写命令)；
* 当前状态下不允许的命令(例如CMD2处于传输状态)；
* 未定义的命令；

对于标准容量SD存储卡，读取操作的超时条件发生的时间(与卡无关)要么比下面给出的这些操作的访问时间长100倍，要么长100毫秒(两者中较低者)。读访问时间定义为CSD参数TAAC和NSAC给出的两个时间之和。在单个读取操作的情况下，这些卡参数定义了读取命令的结束位和数据块的开始位之间的典型延迟。在多读操作的情况下，还定义了数据块的结束位和下一个数据块的开始位之间的典型延迟。

对于标准容量SD存储卡，发生写操作的超时条件的时间(与卡无关)要么比下面给出的这些操作的典型程序时间长100倍，要么长250毫秒(两者中较低者)。CSD中的R2W\_FACTOR字段用于计算通过将读访问时间乘以该因子获得的典型块程序时间。适用于所有写命令(例如SET(CLR)\_WRITE\_PROTECT, PROGRAM\_CSD和块写命令)。

如果卡在SD状态中支持擦除超时计算参数，主机应该使用这些参数来确定擦除超时。如果卡不支持这些参数，擦除超时可以通过块写延迟来估计。

一个擦除命令的持续时间可以用需要擦除的写块数(WRITE\_BL)乘以250ms来估计。

#### 2.7 命令

控制SD存储卡的命令定义有四种：

* 广播命令(bc)，无响应；广播功能仅在主机中所有CMD行连接在一起时才具有。如果是分开的，那么每一张牌将轮流单独接受它；
* 所有卡同时响应的广播命令(bcr)：由于SD存储卡中没有开漏模式，因此只有在所有CMD行分开时才使用这种类型的命令；该命令将被每个卡单独接受和响应；
* 寻址(点对点)命令(ac)：无数据传输；
* 寻址(点对点)数据传输命令(adtc)：数据传输数据所有命令和响应都通过SD存储卡的CMD行发送。命令传输总是从与命令码字对应的位串的左位开始；

命令总是以起始位(总是0)开始，然后是指示传输方向的位。接下来的6位表示命令的索引，该值被解释为二进制编码数(0到63之间)。一些命令需要一个参数(例如地址)，由32位编码。所有的命令都受到CRC的保护)。每个命令码字以结束位(总是1)结束。

SD卡系统的命令集分为几类。每个类都支持一组卡片功能。根据卡支持的命令来确定CCC的设置。一个CCC位对应一个支持的命令号，设置为1。CCC中包含强制命令的类总是被设置为1。具有特定功能的卡可能需要支持一些可选命令。

等级0、2、4、5和8是强制性的，所有SD存储卡都应支持。除CMD40外，SDHC和SDXC是强制性的。其他类是可选的。支持的卡命令类(CCC)被编码为每个卡的卡特定数据(CSD)寄存器中的参数，为主机提供如何访问卡的信息。

如果将不同类型的命令分配给一个命令类(第7类、第8类和第11类)，则可以通过参考SCR寄存器中的命令支持信息来确定支持哪个命令。

#### 2.8 响应

所有响应都通过命令行CMD发送。响应传输总是从响应码字对应的位串的左位开始。代码长度取决于响应类型。

响应总是以一个起始位(总是0)开始，后面是指示传输方向的位(card = 0)。除了R3类型(见下文)之外的所有响应都受CRC保护。每个命令码字以结束位(总是1)结束。

对于SD存储卡有五种类型的响应。SDIO卡支持名为R4和R5的其他响应类型。

* R1响应：码长为48位。第45到40位表示要响应的命令的索引，该值被解释为二进制编码数(介于0和63之间)。卡的状态用32位编码。请注意，如果涉及到卡的数据传输，那么在传输每个数据块后，数据线上可能出现忙音信号。数据块传输后，主机检查是否忙。
* R1b：与R1相同，在数据线上传输一个可选的忙音信号。根据卡在接收命令之前的状态，在接收到这些命令后，卡可能会变得忙碌。主机应检查响应是否繁忙；
* R2：码长为136位。CID寄存器的内容作为对CMD2和CMD10命令的响应发送。CSD寄存器的内容作为对CMD9的响应发送；
* R7：码长为48位。卡支持的3.3V范围电源引脚电压信息通过CMD8的响应发送。19 ~ 16位表示卡支持的电压范围。接受供电电压的卡返回R7响应。在响应中，卡回显参数中设置的电压范围和校验模式；

#### 2.9 SD卡状态

SD存储卡支持以下三种状态字段：

* 卡状态：执行命令的错误和状态信息，在响应中显示；
* SD状态：扩展状态字段，512位，支持SD存储卡的特殊功能；
* 任务状态：CQ模式下排队任务的状态信息，在响应中表示。CMD13 R1通过选择CMD13参数b[15]来表示卡状态或任务状态；

### 3 SPI模式

#### 3.1 介绍

SPI模式由基于flash的SD存储卡提供的二级通信协议组成。这种模式是SD存储卡协议的一个子集，设计用于与SPI通道通信。接口是在上电后的第一个复位命令(CMD0)中选择的，设备上电后不能更改。

SPI标准只定义了物理链路，而不是完整的数据传输协议。SD存储卡SPI实现使用SD存储卡协议和命令集的一个子集。SPI模式的优点是能够使用现成的主机，从而将设计工作量降至最低。缺点是SPI模式相对于SD模式的性能损失。

#### 3.2 SPI总线协议

虽然SD存储卡通道是基于命令和数据位流，由开始位发起，由停止位终止，SPI通道是面向字节的。每个命令或数据块都是由8位字节组成的，并且与CS信号对齐(即长度是8个时钟周期的倍数)。卡开始计数SPI总线时钟周期在CS信号的断言。每个命令或数据令牌应与8时钟周期边界对齐。与SD存储卡协议类似，SPI消息由命令、响应和数据锁令牌组成。主机和卡之间的所有通信都由主机控制。主机通过断言CS信号低来启动每个总线事务。

与SD模式相反，所选卡总是响应命令。当卡在读取操作中遇到数据检索问题时，将响应一个错误响应(替换预期的数据块)，而不是像SD模式那样响应一个超时。此外，在写操作期间发送到卡的每个数据块都将使用数据响应令牌进行响应。

在标准容量存储卡的情况下，一个数据块可以像一个卡写块一样大，也可以像单个字节一样小。部分块读写操作由CSD寄存器中指定的卡选项启用。对于SDHC和SDXC卡，块长度固定为512字节。CMD16设置的块长度仅用于CMD42，不用于内存数据传输。因此，部分块读写操作也被禁用。另外，不支持写保护命令(CMD28、CMD29和CMD30)。

##### 3.2.1 模式选择

SD卡以SD模式上电。如果在接收复位命令(CMD0)期间CS信号被断言(低电平)，将进入SPI模式。如果卡识别到需要SD模式，将不响应命令并保持在SD模式。如果需要SPI模式，卡将切换到SPI并响应SPI模式R1响应。

返回SD模式的唯一方法是进入电源循环。在SPI模式下，不观察SD模式下的SD卡协议状态机。所有在SPI模式下支持的SD卡命令总是可用的。

SEND\_IF\_COND (CMD8)用于校验SD卡接口的工作状态。CMD8的参数格式与SD模式下定义的格式相同。卡通过分析CMD8的参数来检测运行状态的有效性，主机通过分析CMD8的响应来检测运行状态的有效性。供电电压由参数中的VHS字段表示。卡假定VHS中规定的电压为当前供电电压。在任何给定时间，VHS中只有1位被设置为1。检查模式用于主机检查主机与卡之间通信的有效性。

如果该卡显示非法命令，则该卡是遗留的卡，不支持CMD8。如果卡支持CMD8并且可以在供电电压上操作，则响应将返回在命令参数中设置的供电电压和检查模式。如果响应中的VCA设置为0，则卡不能在供电电压下工作。如果校验模式不匹配，CMD8通信无效。

READ\_OCR (CMD58)旨在为SD存储卡主机提供一种机制，以识别与主机所需的VDD范围不匹配的存储卡。如果主机不接受电压范围，则不应进行进一步的初始化顺序。OCR寄存器中的电平应相应地定义。

SD\_SEND\_OP\_COND (ACMD41)用于启动初始化并检查卡是否已完成初始化。必须在第一个ACMD41之前发布CMD8。CMD8的接收扩展了CMD58和ACMD41的功能；ACMD41论证中的HCS和CMD58响应中的CCS。卡忽略了HCS，不接受CMD8。

标准容量SD存储卡忽略HCS。ACMD41的R1响应中的在空闲状态位被卡用来通知主机是否完成了ACMD41的初始化。将此位设置为1表示卡仍在初始化。将该位设置为0表示初始化完成。主机重复发出ACMD41，直到该位被设置为0。卡只在第一个ACMD41检查OCR中的HCS位。在重复ACMD41的同时，主机不能发出除CMD0以外的其他命令。

初始化完成后，主机应从CMD58的响应中获取CCS信息。当卡接受CMD8并完成初始化后，CCS才有效。CCS为0表示该卡为SDSD卡。CCS为1表示该卡为SDHC或SDXC。

##### 3.2.2 总线数据传输

总线上传输的每个SD卡命令都受到CRC位的保护。在SPI模式下，SD存储卡提供了CRC打开模式，使具有可靠数据链路的系统能够排除实现CRC生成和验证功能所需的硬件或固件。

在CRC关闭模式下，命令的CRC位被定义为不关心发送端，而被接收端忽略。缺省情况下，SPI接口初始化为CRC关闭模式。然而，用于将卡切换到SPI模式的复位命令(CMD0)在SD模式下被卡接收，因此应该有一个有效的CRC字段。

由于CMD0没有参数，所以所有字段(包括CRC字段)的内容都是常量，不需要在运行时计算。卡进入SPI模式后，将根据CMD59设置对包括CMD0在内的所有命令进行CRC检查。

主机可以使用CRC\_ON\_OFF命令(CMD59)打开和关闭CRC选项。主机应在发出ACMD41之前启用CRC验证。始终启用CMD8 CRC校验。主机应在CMD8参数中设置正确的CRC。如果检测到CRC错误，无论命令索引如何，卡都会在R1响应中返回CRC错误。

##### 3.2.3 读取数据

SPI模式支持单块读操作和多块读操作(CMD17或CMD18)。在接收到有效的读取命令后，卡将使用响应令牌和数据令牌进行响应。在标准容量卡的情况下，数据令牌的大小由SET\_BLOCKLEN (CMD16)设置的块长度决定。对于SDHC和SDXC卡，无论CMD16设置的块长度如何，块长度都固定为512字节。

最大块长度为512字节，与CSD中定义的READ\_BL\_LEN无关。如果在标准容量卡中启用部分块访问(即CSD参数READ\_BL\_PARTIAL等于1)，块长度可以是1到512字节之间的任何数字。起始地址可以是卡有效地址范围内的任意字节地址。但是，每个块应包含在单个物理卡扇区中。如果禁用部分块访问，则只支持512字节的数据长度。SDHC和SDXC卡只支持512字节的块长度。起始地址应与块边界对齐。

在数据检索错误的情况下，卡将不传输任何数据。相反，一个特殊的数据错误令牌将被发送到主机。

在多块读取操作的情况下，每个传输的块都有其16位CRC后缀。停止传输命令(CMD12)实际上将停止数据传输操作(与SD存储卡操作模式相同)。

##### 3.2.4 写入数据

SPI模式支持单块和多块写命令。在接收到有效的写命令(SD存储卡协议中的CMD24或CMD25)后，卡将使用响应令牌进行响应，并等待主机发送数据块。CRC后缀，块长度和起始地址限制(CSD参数WRITE\_BL\_PARTIAL控制部分块写选项和WRITE\_BL\_LEN除外)与读操作相同。

每个数据块都有一个开始块标记的前缀(一个字节)。在接收到数据块后，卡将使用数据响应令牌进行响应。如果接收到的数据块没有错误，将对其进行编程。只要卡在忙着编程，一个连续的忙令牌流将被发送到主机(有效地保持低数据输出线)。

一旦编程操作完成，主机应该使用SEND\_STATUS命令(CMD13)检查编程的结果。一些错误(例如地址超出范围，写保护违反等)仅在编程期间检测到。在数据块上执行并通过数据响应令牌与主机通信的唯一验证检查是CRC和一般写错误指示。

在多块写操作中，停止传输将通过在下一个块的开始处发送停止传输令牌而不是开始块令牌来完成。如果出现写错误指示(在数据响应上)，主机将使用SEND\_NUM\_WR\_BLOCKS (ACMD22)来获取写好的写块的数量。

当卡忙时，重置CS信号不会终止编程过程。卡将释放数据输出线并继续编程。如果在编程完成之前重新选择卡，则数据线线将被强制返回低电平，并且所有命令将被拒绝。

重置卡(使用CMD0作为SD存储卡)将终止任何挂起的或正在进行的编程操作。这可能会破坏卡上的数据格式。

SPI模式下的擦除和写保护管理过程与SD模式下的完全相同。当卡擦除或更改预定义扇区列表的写保护位时，将处于忙状态并使数据线保持低电平。

与SD存储卡协议(其中寄存器内容作为命令响应发送)不同，在SPI模式中读取CSD和CID寄存器的内容是一个简单的读块事务。该卡将响应一个标准的响应令牌，后跟一个16字节的数据块，以16位CRC为后缀。CSD命令的数据超时不能设置为卡的TAAC，因为该值存储在卡的CSD中。因此，CSD寄存器的读延迟使用标准响应超时值(NCR)。

SD存储卡需要一个已定义的复位顺序。上电复位或执行复位命令(SD存储卡为CMD0)后，卡进入空闲状态。在这种状态下，唯一有效的主机命令是CMD8 (SEND\_IF\_COND)， ACMD41 (SD\_SEND\_OP\_COND)， CMD58 (READ\_OCR)和CMD59 (CRC\_ON\_OFF)。