## SDIO协议

### 1 介绍

#### 1.1 概述

SDIO卡是基于SD存储卡设计的，与SD存储卡兼容。SDIO卡是为移动电子设备提供低功耗的高速数据I/O。插入到非SDIO感知主机的SDIO卡将不会对该主机或其软件造成物理损坏或中断。在主机正常初始化和查询卡的过程中，卡将自己标识为SDIO卡。然后，主机软件获取卡片信息并确定该卡的I/O功能是否可以激活。如果该卡是可以被激活的，则允许进行上电并启动内置I/O功能。

#### 1.2 SDIO协议特性

* 针对便携式和固定式应用；
* 需要对SD物理总线进行最小或不进行修改；
* 对内存驱动软件进行最小更改；
* 支持即插即用；
* 多功能支持；
* 允许SDIO卡中断主机；
* 工作电压范围在2.7到3.6V；

### 2 SDIO信号定义

#### 2.1 SDIO卡类型

SDIO卡分为高速SDIO卡和低速SDIO卡。在0-25MHz的时钟范围内，高速卡支持SPI、1位SD和4位SD传输模式。高速SDIO卡的数据传输速率超过100 Mb。低速SDIO卡只需要SPI和1位SD传输模式。4位支持是可选的。低速SDIO卡应支持0- 400KHZ的全时钟范围。低速卡是用最少的硬件支持低速I/O功能。

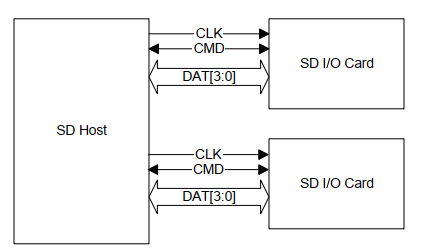
#### 2.2 SDIO卡模式

* SPI模式：在这种模式下，引脚8是内存未定义的，被用作中断引脚。所有其他引脚和协议都与SD卡物理规范相同；
* 1位SD卡模式：此模式与SD内存定义的1位数据模式相同。在此模式下，数据仅在DAT[0]引脚上传输。在这种模式下，引脚8是内存未定义的，被用作中断引脚。所有其他引脚和协议都与SD内存相同；
* 4位SD卡模式：此模式与SD内存定义的4位数据模式相同。在这种模式下，数据在所有4个数据引脚上传输(DAT[3:0])。在这种模式下，中断引脚不能独占使用，因为被用作数据传输线。如果需要中断功能，则需要一个特殊的时序来提供中断；

#### 2.3 SDIO主机模式

如果支持SDIO的主机支持SD传输模式，建议同时支持1位和4位传输模式。虽然仅支持4位传输模式的SDIO主机是可能的，但与低速SDIO卡的性能不高。

#### 2.4 信号引脚



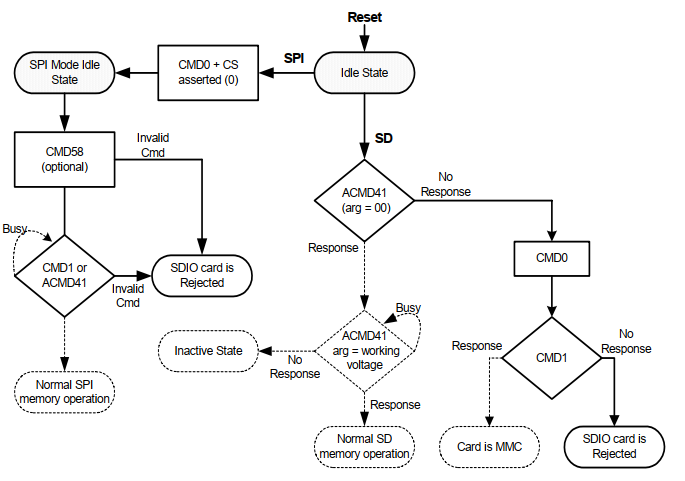
### 3 SDIO卡初始化

#### 3.1 初始化差异

SDIO协议要求是SDIO卡在插入时不能导致非I/O感知主机失败。为了防止I/O功能在非I/O感知主机上运行，需要对SD卡识别模式流程图进行更改。新增命令(IO\_SEND\_OP\_COND, CMD5)，用于I/O感知主机初始化SDIO时替换ACMD41。

复位或上电后，卡上的所有I/O功能将被禁用，卡的I/O部分将不执行除CMD5或CMD0以外的任何操作。如果卡上安装了SD内存，该内存将正常响应所有正常的内存命令。

仅I/O卡不响应ACMD41，因此作为MMC卡出现。仅I/O卡也不能响应用于初始化MMC卡的CMD1并且作为无响应卡。然后主机放弃并禁用这张卡。主机没有收到来自I/O卡的响应并强制其进入非活动状态。



**图3-1-1 SDIO对非I/O感知初始化的响应**

感知SDIO的主机在CMD55/ACMD41对之前发送CMD5，因此将在CMD55/ACMD41对中接收有效的OCRR4响应CMD5并继续初始化卡。

如果卡的I/O部分没有收到CMD5，则I/O部分保持不活动状态并且不响应除CMD5以外的任何命令。如果卡上没有安装内存，则该卡不会响应任何内存命令。然后将卡移除并插入到不支持SDIO的主机中。该主机将不启用I/O功能，因此将显示为仅内存卡。如果主机具有I/O感知，将向卡发送CMD5，卡将响应R4。主机读取R4值并知道可用I/O功能的数量以及是否存在任何SD内存。

在主机初始化卡的I/O部分之后，读取卡的公共信息区(CIA)。这是通过发出一个读取命令来完成的，从I/O功能0的地址0x0处的字节开始。CIA包含卡通用控制寄存器(CCCR)和功能基本寄存器(FBR)。CIA中还包括指向卡的通用卡片信息结构(CIS)和每个单独功能CIS的指针。CIS包括电源、功能、制造商和主机需要的其他信息，以确定I/O功能是否适合上电。如果主机确定该卡应被激活，则CCCR区域中的寄存器启用该卡和每个单独的功能。此时，I/O卡的所有功能都完全可用。此外，主机还可以根据不同的功能控制功耗和启用以及禁用中断。

#### 3.2 CMD5命令

CMD5对SDIO卡的作用类似于ACMD41对SD存储卡的作用。查询I/O卡所需的电压范围。CMD5的正常响应是R4，可以是SD格式，也可以是SPI格式。

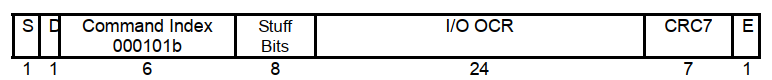
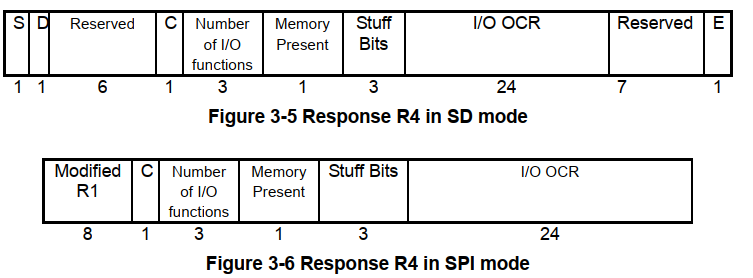


图3-2-1 CMD5命令格式

接收CMD5的SDIO卡应该响应SDIO唯一的响应R4。



当SDIO卡接收到CMD5，该卡的I/O部分就可以正常响应所有进一步的命令。这个I/O使能功能在I/O卡内应该保持设置直到复位，电源周期或写入I/O复位的CMD52被卡接收。对于仅内存卡，正确的响应是内存存在位为1和IO功能数目为0。

#### 3.3 组合卡初始化注意事项

主机在初始化组合卡(SDIO和SD内存在同一张卡上)时必须注意到一些特殊情况。这是因为组合卡的实现使用两个独立的控制器(内存和I/O)并共享相同的总线。为了防止SDIO和SD内存控制器之间的冲突，主机必须检测并正确配置组合卡的两个控制器。这是由于两个控制器对复位(硬复位或软复位)的不同响应引起的。这种考虑仅适用于1位SD和4位SD模式。在SPI模式下，卡的选择和取消选择是使用硬件CS线而不是RCA来完成的。

当主机重新初始化I/O和内存控制器时，强烈建议主机在执行任何其他操作之前执行电源复位或向两个控制器发出复位命令。如果主机选择使用复位命令，应该先发出CMD52(I/O复位)，这是因为在CMD0之后不能发出CMD52。复位后，主机需要重新初始化I/O控制器和内存控制器。

如果主机打算只使用组合卡的SDIO或内存部分，强烈建议在进行任何其他操作之前，主机电源复位或向两个控制器发出复位命令。如果主机选择使用复位命令，应该先发出CMD52(I/O复位)，这是因为在CMD0之后它不能发出CMD52。重置完成后，主机会重新初始化I/O控制器和内存控制器。

当主机重新初始化组合卡时，主机可以发出的可接受的命令受到限制直到I/O控制器进入命令状态并且内存控制器进入传输状态。当主机发出这些被禁止的命令时，组合卡可能无法正常工作。

当接收到CMD59时，组合卡应在卡的SDIO和内存部分同步CRC启用。如果主机使用CMD59启用CRC，随后重新初始化I/O或内存控制器，该控制器的CRC将默认关闭，主机应发出CMD59以重新启用CRC。当收到CMD59后，组合卡返回R1响应而仅SDIO卡返回修改后的R1响应。

### 4 与SD内存差异

#### 4.1 差异

* 不支持部分SD内存命令：仅SDIO卡或组合卡的I/O部分都不支持SD存储卡所需的几个命令。其中一些命令在SDIO卡中没有使用。此外，SD存储卡的几个命令在与存储卡的SDIO部分一起使用时具有不同的命令；
* 复位SDIO卡：重置命令(CMD0)仅用于内存或组合卡的内存部分。为了复位仅I/O卡或组合卡的I/O部分，使用CMD52将1写入CCCR的RES位(寄存器6的第3位)；

#### 4.2 总线宽度

对于SD存储卡，使用ACMD6设置SD模式的总线宽度。SDIO卡使用CMD52写入CCCR来选择总线宽度。在组合卡的情况下，两种选择方法都存在。

如果只激活组合卡的I/O功能，则只需写入CCCR即可更改总线宽度模式。如果只有内存是活动的，那么ACMD6就是改变总线宽度所需要的。如果I/O和内存都是活动的，那么CCCR和ACMD6都需要。在组合卡中，存储器和I/O控制器都应设置为相同的总线宽度。

如果组合卡支持锁以及解锁操作，则无法更改被锁定卡的总线宽度并且返回非法命令错误。主机在改变总线宽度前需要用CMD42解锁卡。这也意味着在管理锁定卡之前，主机不应该在初始化期间更改总线宽度。

#### 4.3 卡检测电阻

SD内存和I/O卡在DAT[3]上使用一个上拉电阻来检测卡的插入。在SD存储器和SDIO之间启用以及禁用这个电阻是不同的。SD内存使用ACMD42来控制这个电阻，而SDIO使用CMD52来写CCCR。在组合卡的情况下，两个控制方法都存在并应由主机管理。对于组合卡，电阻只有在存储器和I/O控制寄存器都使能电阻时才启用。上电后，主机应该使用ACMD42到内存控制器或CCCR写到SDIO控制器禁用电阻。上电后，两个位置默认为电阻启用。在I/O复位后，I/O电阻使能不会改变。

#### 4.4 传输块大小

SDIO卡可以多字节或可选的块格式传输数据，而SD存储卡则固定的块传输模式。SD卡将数据传输的块大小限制为2的幂，除非使用部分读写。为了适应I/O功能的各种块大小，SDIO允许从1字节到2048字节。SDIO卡功能可以在CIS中定义一个小于上述最大值的最大块大小或字节数。

#### 4.5 终止传输

与SD存储卡通信的主机使用CMD12向存储卡传输读或写数据。对于SDIO卡，CMD12终止位被CCCR中的ASx位取代。通常终止用于停止无限块传输(块计数为0)。如果要传输确定数量的块，建议主机发出具有正确块计数的块命令，而不是使用无限计数并在正确的时间终止数据。

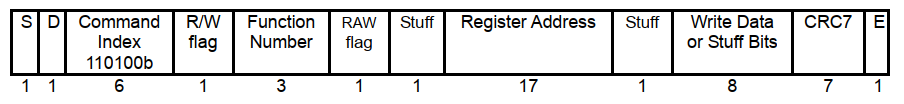
在I/O扩展读操作期间，主机可以随时通过向CCCR写入来发出I/O中止。数据传输在I/O 终止命令结束位之后停止2个时钟周期。

在I/O扩展写操作期间，主机可以在数据块之间的任何时间通过向CCCR写入来发出I/O中止。在这种情况下，最终的块传输(包括卡的CRC响应)应该已经完成。这要求I/O中止命令的结束位应该在对最后一个数据块的CRC响应的结束位之前最多出现两个时钟。主机不能在写块的中间终止。在I/O终止被发送到卡后，卡发出忙信号(将DAT[0]设置为低电平)，直到完成最后一个数据块的传输。在繁忙期间，主机可以通过写入CCCR BR位来释放总线。

### 5 新IO读写命令

#### 5.1 CMD52命令

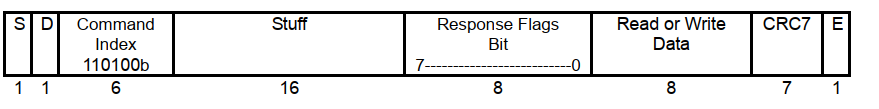
CMD52是访问任何I/O功能中总共128K寄存器空间内的单个寄存器的最简单方法，包括公共I/O区(CIA)。该命令只使用一个命令以及响应对读写1个字节。一个常见的用途是初始化寄存器或查询I/O功能的状态值。该命令是读取或写入单个I/O寄存器的最快方法。



#### 5.2 R5响应

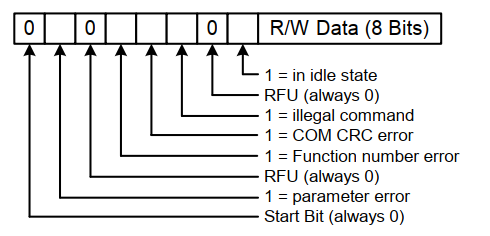
##### 5.2.1 CMD52响应(SD模式)

如果操作是读命令，则读取的数据以8位值返回并且返回15位状态信息。



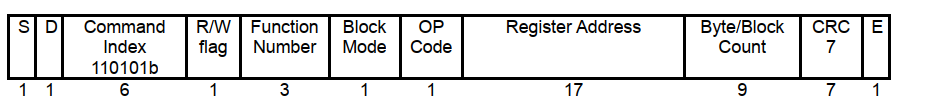
##### 5.2.2 R5响应(SPI模式)

如果操作是读命令，则读取的数据以8位值返回并且在一个SPI R1响应字节中返回8位状态信息。



#### 5.3 CMD53命令

为了用一个命令读写多个I/O寄存器，定义了一个新命令CMD53。该命令包含在I/O命令中。该命令允许使用单个命令读取或写入大量I/O寄存器。



### 6 SDIO内部卡操作

#### 6.1 概述

I/O访问与内存的不同之处在于，寄存器可以单独和直接地读写，而不需要FAT文件结构或块的概念。这些寄存器允许访问I/O数据，控制I/O功能或向主机传输I/O数据。SD内存依赖于固定块大小的概念，其命令读取或写入这些固定大小块的倍数。I/O可能有固定的块长度，也可能没有，读大小可能不同于写大小。I/O操作可以基于字节或块大小计数。

每个SDIO卡可能有1到7个功能，外加一个内置的存储功能。功能是一个独立的I/O设备。I/O功能可能完全相同，也可能完全不同。所有的I/O功能都被组织成一个寄存器集合。每个I/O功能最多可以有(1<<17)个寄存器。这些寄存器和位可以是只读(RO)、只写(WO)或读/写(R/W)。这些寄存器的宽度可以是8、16或32位。所有的寻址都基于字节访问。

仅SDIO卡中的所有寄存器和组合卡的SDIO部分应在1秒内完成读写数据传输。这个超时值与请求的数据在DAT[x]行上从主机传输到/从主机传输的时间有关，而不是命令和响应之间的时间。这个等待时间由卡向主机发出信号，使用busy进行写操作或延迟读操作的起始位。主机可以使用1秒作为无响应位置的超时值。

#### 6.2 中断

所有SDIO主机都应该支持硬件中断。如果一台主机不支持中断，可能在使用对中断条件快速响应的SDIO卡时遇到问题。SDIO或组合卡中的每个功能都可以根据需要实现中断。在SDIO功能上使用的中断是由电平触发的类型。电平触发意味着任何功能都可以在任何时候发出中断信号，但是一旦发出中断信号就不会释放中断，直到中断被处理。

功能应该继续发出中断信号，直到主机响应并清除中断。由于多个中断可能同时处于活动状态，因此主机负责确定中断源并根据需要处理。这是在SDIO函数上通过使用两个比特来完成的，中断启用和中断挂起。每个可能产生中断的功能都有一个中断使能位。此外，SDIO卡有一个控制所有功能的主中断使能。如果设置了该功能的使能和卡的主使能，则中断将仅向SD总线发出信号。

#### 6.3 挂起和恢复

在多功能SDIO或组合卡中，有多个设备(I/O和内存)共享SD总线。为了允许在多个设备之间共享对主机的访问，SDIO和组合卡可以实现挂起和恢复。如果卡支持挂起和恢复，主机可以暂时停止向一个功能或内存的数据传输操作(挂起)，释放总线以便向另一个功能或内存进行更高优先级的传输。一旦这个高优先级的传输完成，原始传输将从停止的地方重新开始(恢复)。

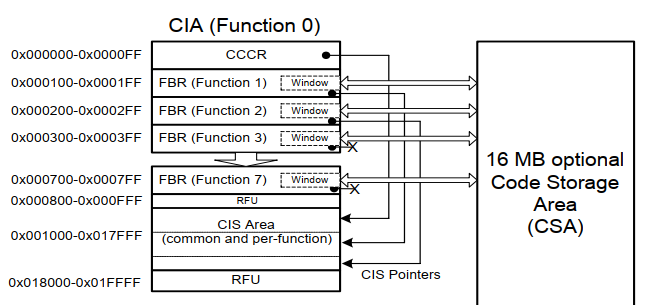
#### 6.4 读等待

根据SD物理规范构建的主机设备应控制SDCLK，以便在主机无法接受更多数据时，停止从执行多次读取命令的卡中读取数据块输出。在主机停止SDCLK期间，不能发出CMD52。这个限制导致了一个问题，即按照SD物理规范构建的主机设备不能在多个读周期中执行I/O命令。

为了消除这个限制，SDIO增加了读等待控制，使主机能够在多个读周期中发出CMD52。读等待使用DAT[2]进行，允许主机向卡发出信号，暂时停止卡发送读数据。此功能对于SDIO或组合卡是可选的。但是，如果SDIO或组合卡支持读等待，则所有功能和任何内存都应支持读等待。任何支持挂起和恢复的卡也应该支持读等待。

#### 6.5 SDIO内部映射

SDIO卡有一个固定的内部寄存器空间和一个功能唯一区。固定区域包含有关卡的信息以及固定位置的某些寄存器。所述固定位置允许任何主机获取有关所述卡的信息并执行简单操作。功能唯一区域是单个功能区域，由应用程序定义。



通用I/O区(CIA)应在所有SDIO卡上实现。主机通过I/O读写功能0来访问CIA。CIA内部的寄存器用于启用和禁用I/O功能，控制中断的生成和可选地加载软件来支持I/O功能。CIA中的注册表还提供有关功能、能力和需求的信息。

卡通用控制寄存器允许在每个卡(主)和每个功能的基础上对I/O卡的启用和中断进行快速检查和控制。CCCR中的位混合为读/写和只读。如果SDIO卡上没有提供这7种功能中的任何一种，则未使用的功能对应的位都为只读，读为0。所有为将来使用保留的位都是只读的并且返回值为0。上电或复位后，所有可写位被置为0。即使在初始化I/O功能被禁用之后，也可以访问CCCR。

除了CCCR之外，每个支持的I/O功能都有一个256字节的区域，用于允许主机快速确定每个功能的要求，为每个功能启用电源选择并启用软件加载。

卡片信息结构提供了关于卡片和各个功能的更完整的信息。CIS是读取卡中存在的所有I/O功能信息的公共区域。所有支持I/O的卡都应该有一个通用的CIS，每个功能都有一个CIS。CIS是通过对一个固定区域的读取来访问的。公共区域和每个功能都有一个指针，指向该内存空间内其CIS的起始地址。

#### 6.6 多功能SDIO卡

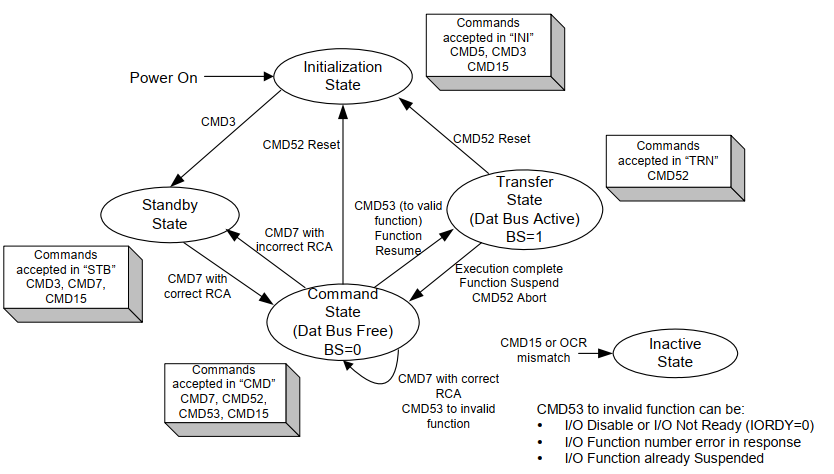
多功能SDIO卡应为卡上的每个功能提供一套单独的配置寄存器。多功能SDIO卡应使用卡上所有功能通用的CIS和卡上每个功能专用的单独功能CIS的组合。通用CIS描述了卡上所有功能的通用特性。每个特定于功能的CIS描述了特定于SDIO卡上特定功能的特性。

CMD5响应表示功能的总数。主机应根据CMD5响应遍历CIS条目。

R5响应的ERROR状态标志是“E R X”类型并且可以指示前面命令中的错误。由于主机软件需要一种方法来确定哪个功能检测到错误，因此多功能SDIO卡在随后向同一功能发出的命令中应只返回R5错误状态标志。

主机通过写入16位功能I/O块来设置功能的多个块传输的块大小。当块模式设置为1时，主机不能使用CMD53写入该寄存器。如果卡在执行CMD53之前检测到无效的块大小且块模式设置为1，则在当前响应中指示错误，不进行数据传输。

#### 6.7 总线状态



### 7 嵌入式I/O代码存储区

#### 7.1 概述

为了支持SDIO卡的即插即用，卡中包含的每个功能可能需要包含一块用于存储驱动程序或应用程序的内存。此外，由于同一个SDIO卡可以在多个不同的主机平台上使用，因此每个功能可能需要几个不同版本的代码。一种选择是将这些程序存储在组合卡的标准SD内存部分。另一种标准访问方式是加载可选的代码存储区(CSA)中包含的代码。CSA是一个单独的16MB内存区域，使用CSA地址指针和FBR寄存器中包含的CSA窗口寄存器来访问。请注意，每个功能可能都有自己的CSA来支持。

#### 7.2 访问CSA

为了让主机访问功能的CSA，首先应该确定该功能是否支持CSA。主机读取地址为0x00n00的FBR寄存器，其中n是功能号(0x1到0x7)。如果第6位=1，则该功能支持CSA，主机通过第7位写入1来使能访问。下一步是主机加载24位地址以开始读写。这是通过将24位写入寄存器0x00n0C到0x00n0E来实现，其中n是功能号(0x1到0x7)。一旦开始地址被写入，就可以通过访问寄存器0x00n0F (CSA数据窗口寄存器)来读写数据。如果需要读写的字节超过1个，可以执行扩展I/O命令(字节或块)。地址指针在每次访问窗口寄存器时自动递增，因此访问的是CSA内的顺序地址。操作完成后，下一次操作的地址将保存在24位地址寄存器中供主机读取。

#### 7.3 CSA数据格式

存储在CSA中的数据应采用FAT12/FAT16格式。使用CSA存储不同主机类型的程序或数据需要SDIO卡制造商以主机可以识别的文件格式加载程序和数据。

### 8 SDIO中断

#### 8.1 概述

为了允许SDIO卡中断主机，在SD接口的引脚上添加了一个中断功能。8号引脚在4位SD模式下工作时用作DAT[1]，用于向主机发出卡的中断信号。中断的使用对于每个卡或卡内的功能都是可选的。SDIO中断是由电平触发，也就是说，中断信号应该保持活动(低)直到被主机处理或者由于中断期结束而取消)。一旦主机处理了中断，就会通过一些功能唯一的I/O操作被清除。所有主机应在所有数据线DAT[3:0]上提供上拉电阻。

#### 8.2 中断时序

在SPI和1位SD模式下，8号引脚专用于中断功能。在SPI和SD 1位模式中，对中断没有时间限制。在SPI或1位SD模式中的卡通过设置引脚8为低电平随时向主机发出中断信号。主机检测这个挂起的中断并负责清除中断。如果SDIO卡在SPI模式下工作，如果卡未被选中，则可能不会触发来自该卡的中断(CS为0电平)。

由于IRQ和DAT[1]在4位SD模式下共用8号引脚，因此中断只能在特定时间内由卡发送并被主机识别。引脚8上的低电平应被识别为中断的时间定义为中断周期。在所有其他时间，主机中断控制器应忽略引脚8上的电平。

由于SDIO卡使用电平触发中断，主机应该通过I/O读或写某些功能唯一区域来清除挂起的中断。在一些主机实现中，将CMD52发送到卡是由主机适配器硬件处理的，而主机CPU可以执行其他操作。如果中断清除的时间不受控制，这种情况可能允许已经被处理的中断重新中断主机。为了防止这种情况，任何实现中断的SDIO卡都应该遵循一些必要的定时，在写入清除中断的功能唯一区域后，从DAT[1]行删除中断。中断的清除可以由函数唯一方法中的I/O写入引起，也可以由函数唯一I/O读取引起。使用I/O读取清除中断的一个例子是这样一个函数，其中数据寄存器的读取可以自动清除数据就绪中断。