

## 27.4 SDI 特殊寄存器

本小节主要介绍一下 SDI 的特殊寄存器

### 27.4.1. 配置寄存器类型

配置寄存器域指定下面描述的属性。

寄存器属性	描述
RO	只读寄存器：寄存器位是只读的，并且不能通过软件或者其它设置操作来进行修改。向该位写入将被忽略
ROC	只读状态：这个位被初始化为 0，向该位写入将被忽略
RW 或者 R/W	可读写寄存器：寄存器位可读写，并且能进行其它任何设置或者通过软件清除到所要的状态
RW1C	只读状态，写 1 到清除状态：当读时寄存器位指出状态。一个设置位指出一个状态，通过写 1 可以清除。写 0 到 RW1C 位没有影响
RWAC	读写自动清除寄存器：通过设置这位主机驱动器要求一个主机控制器操作。当操作完成，主机驱动器将自动清除该位。写 0 到 RWAC 位没有影响
HWInit	硬件初始化：通过固件或者硬件装置初始化寄存器位。初始化后读取位，并且写入该位被忽略
Rsvd 或 Reserved	保留。这些位初始化为 0，写入这些位被忽略

### 27.4.2. 系统地址寄存器

该寄存器包含用于 DMA 传输的物理存储地址。

寄存器	地址	读/写	描述	复位值
SYSAD0	0x7C200000	读/写	系统地址寄存器(0通道)	0x0
SYSAD1	0x7C300000	读/写	系统地址寄存器(1通道)	0x0
SYSAD2	0x7C400000	读/写	系统地址寄存器(2通道)	0x0

名称	位	描述	初始状态
SYSAD	[30:0]	DMA 系统地址 这个寄存器包含系统存储地址用于DMA传输。当主机控制器停止	0x00

	<p>DMA传输，寄存器将指向下一个邻近数据位置的系统地址。如果没有处理进行(也就是，处理停止后)，那么它能被访问。在传输过程中，读操作可能返回一个无效值。</p> <p>在开始DMA传输前，主机控制器将初始化这个寄存器。DMA停止后，能从这个寄存器中读取邻近的数据位置的系统地址。</p> <p>DMA传输等待在每个在块大小寄存器中被主机DMA缓冲区边界指定的边界。主机控制寄存器产生DMA中断来请求主机更新寄存器。主机驱动器设置下一个数据位置的系统地址到这个寄存器。</p> <p>当这个寄存器最高的字节(003h)被写入，主机控制器重起DMA传输。当重新启动DMA通过重新启动指令或者通过在块间隔控制寄存器设置继续请求，主机控制器将在被存储在系统地址寄存器中的下一个邻近的地址开始。</p>	
--	--	--

### 27.4.3. 块大小寄存器

该寄存器用来设置一个数据块的字节数。

寄存器	地址	读/写	描述	复位值
BLKSIZE0	0x7C200004	读/写	主机DMA缓冲区边界和传输块大小寄存器(0通道)	0x0
BLKSIZE1	0x7C300004	读/写	主机DMA缓冲区边界和传输块大小寄存器(1通道)	0x0
BLKSIZE2	0x7C400004	读/写	主机DMA缓冲区边界和传输块大小寄存器(2通道)	0x0

名称	位	描述	初始状态
	[15]	保留。	0
	[14:12]	<p>主机DMA缓冲区边界。</p> <p>在虚拟内存系统大的邻近的存储空间不可用。为了进行更长的DMA传输，系统地址寄存器将在DMA传输期间对每个系统存储边界更新。这些位指定在系统内存中邻近的缓冲区的大小。DMA传输将等待被这些区域指定的每一个边界，并且主机控制器产生DMA中断来要求主机驱动器来更新系统地址寄存器。</p> <p>在这些寄存器设置为0的情况下(缓冲区大小为4K字节)，字节地址的低12位指向邻近的缓冲区的数据，高20位的指向系统存储器中缓冲区。当主机控制器检查完地址位11到12位，DMA传输停止。</p> <p>当在容限寄存器里将DMA支持设置为1时，这些位将被支持，并且当在传输模式寄存器里将DMA有效设置为1时，这项功能被启动。</p> <p>000b = 4KB(检测 A11 执行)</p> <p>001b = 8 KB (检测 A12 执行)</p> <p>010b = 16 KB (检测 A13 执行)</p>	0

		011b = 32 KB (检测 A14 执行) 100b = 64 KB (检测 A15 执行) 101b = 128 KB (检测 A16 执行) 110b = 256 KB (检测 A17 执行) 111b = 512 KB (检测 A18 执行)	
	[11:0]	传输块大小。 这个寄存器为CMD17、CMD18、CMD24、CMD25和 CMD53指定数据传输的传输区段大小。值的范围能设置为1到缓冲区的最大值。存储器中，它设置为512个字节。没有处理在执行，它只能被访问（也就是处理关闭后）。在传输期间读操作将返回无效值，写操作将被忽略。 0200h = 512字节 01FFh = 511字节 ... .. 0004h = 4字节 0003h = 3字节 0002h = 2字节 0001h = 1字节 0000h = 无数据传输	0

#### 27.4.4. 块计数寄存器

该寄存器用来设置数据块的数量。

寄存器	地址	读/写	描述	复位值
BLKCNT0	0x7C200006	读/写	为当前传输的块计数( 0通道)	0x0
BLKCNT1	0x7C300006	读/写	为当前传输的块计数( 1通道)	0x0
BLKCNT2	0x7C400006	读/写	为当前传输的块计数( 2通道)	0x0

名称	位	描述	初始状态
	[15:0]	为当前传输进行块计数。 当在传输模式寄存器中将块计数启动设置为1，并且是多区段传输时，这个寄存器启动。主机驱动器将设置为这个寄存器设置一个1到最大区段计数中的一个值。每一块传输后，主机控制器减少这个块计数，当计数减少到0时停止。设置块计数为0表示没有数据块正在传输。 这个寄存器必须在没有传输执行（也就是，传输停止）时访问这个寄存器。在数据传输期间，在这个寄存器读操作将返回一个有效值，忽略写操作。当将传输内容作为一个延缓指令的结果，通过读取这个寄存器的传输的块的数量。当发出一个	0

	指令恢复先前的传输内容，主机驱动器将恢复先前保存的块数。 FFFFh = 65535 块 ... .. 0002h = 2块 0001h = 1 块 0000h = 停止块计数	
--	---	--

### 27.4.5. 变元寄存器

该寄存器包含SD指令变元。

寄存器	地址	读/写	描述	复位值
ARGUMENT0	0x7C200008	读/写	指令变元寄存器(0通道)	0x0
ARGUMENT1	0x7C300008	读/写	指令变元寄存器(1通道)	0x0
ARGUMENT2	0x7C400008	读/写	指令变元寄存器(2通道)	0x0

名称	位	描述	初始状态
ARG	[31:0]	指令变元。在SD存储卡物理层规范中，SD指令变元被指定作为指令格式的位39到8	0

### 27.4.6. 传输模式寄存器

该寄存器用于控制数据传输的操作。在发出用于传输数据的指令前，或者是在发出一个恢复指令前，主机控制驱动器将设置这个寄存器。当数据传输暂停和使用恢复指令恢复它之前，主机驱动器将保存这个寄存器的值。为了防止数据丢失，在数据传输期间，主机控制器将为寄存器执行写保护。当指令禁止(DAT)在当前状态寄存器中设置为1时，写入这个寄存器将被忽略。

寄存器	地址	读/写	描述	复位值
TRNMOD0	0x7C20000C	读/写	传输模式设置寄存器(0通道)	0x0
TRNMOD1	0x7C30000C	读/写	传输模式设置寄存器(1通道)	0x0
TRNMOD2	0x7C40000C	读/写	传输模式设置寄存器(2通道)	0x0

名称	位	描述	初始状态

[15:10]	保留。	0
[9:8]	指令完成信号控制： ‘00’ = No CCS 操作(正常操作，非 CE-ATA 模式) ‘01’ = 读或者写数据传输CCS启动 (仅CE-ATA 模式) ‘10’ = 没有数据传输CCS启动 (仅CE-ATA 模式) ‘11’ = 异常中断完成信号(ACS) 产生(仅CE-ATA 模式)	0
[7:6]	保留。	0
[5]	多/单区段选择。 这位激活多区段DAT线性数据传输。对于其它指令，这位设置为0。 如果这位为0不需要设置块计数寄存器 (参考表6-80，确定传输类型)。 1 = 多区段 0 = 单区段	0
[4]	数据传输流向选择。 这位定义了DAT线数据传输的流向。主机驱动器将这位设置为1，将数据从SD卡传输到SD主机控制器，对于其它指令，它被设置为0。 1 = 读 (从卡到主机) 0 = 写 (从主机到卡)	0
[3]	保留。	0
[2]	自动CMD12启动。 存储器多区段传输要求CMD12来停止传输。当这位设置为1，最后一个区段传输完成时，主机控制器自动发出CMD12。主机驱动器不设置这位发出不要求CMD12来停止数据传输的指令。 1 = 有效 0 = 无效	0
[1]	块计数启动。 该位用来启动块计数寄存器，仅使用于多区段传输。当该位置为0，块计数寄存器无效，这用于执行无限制传输 (参考表6-80)。 1 = 有效 0 = 无效	0
[0]	DMA 启动。 该位启动DMA 功能。当在容限寄存器中“DMA支持”支持DMA作为指示时，DMA启动。 如果不支持DMA，该位是没有意义的，置为0。如果这位设置为1，当主机驱动器写到指令寄存器的高位字节 (00Fh)，将开始DMA操作。 1 = 有效 0 = 无效	0

寄存器设置怎样确定数据的类型，如表27-1所示。

表 27-1 确定传输类型

多/单区段块选择	块计数启动	块计数	功能
0	-	-	单区段传输
1	0	-	无限区段传输
1	1	非0	多区段传输
1	1	0	停止多区段传输

注：对于CE-ATA 存取，指令完成信号失效后，（自动）CMD12必须发送。

### 27.4.7. 命令寄存器

这个寄存器包含SD指令变元。在写入这个寄存器前，主机驱动器将在当前状态寄存器检查指令禁止 (DAT)位和指令禁止 (CMD)位。写入这个寄存器的高位字节引发SD指令产生。主机驱动器将负责写入这个寄存器，因为当指令禁止 (CMD)被设置，主机控制器不提供写保护。

寄存器	地址	读/写	描述	复位值
CMDREG0	0x7C20000E	读/写	命令寄存器(通道0)。	0x0
CMDREG1	0x7C30000E	读/写	命令寄存器(通道1)。	0x0
CMDREG2	0x7C40000E	读/写	命令寄存器(通道2)。	0x0

名称	位	描述	初始状态
	[15:14]	保留。	
	[13:8]	指令索引。 这些位设置为SD存储卡物理层规范中指令格式的第40到45位和SDIO卡规范中指定的指令数 (CMD0-63, ACMD0-63)。	
	[7:6]	指令类型。 有三种特殊指令类型：暂停、恢复和中断。对于其它指令这位设置为00b。 • 暂停指令 如果暂停指令成功，主机控制器将假设SD总线被释放，它能够用DAT线发出下一个指令。主机控制器将不进行读等待处理，并且停止检测写处理的繁忙状况。在4位模式时，将开始中断周期。如果暂停指令失败，主机控制器将保持它的当前状态，主机驱动器将通过在块间隔控制器中设置“继续请求”恢复传输。	

		<ul style="list-style-type: none"> <li>• 恢复指令 主机驱动器通过恢复数据传输通过恢复寄存器在范围000~00Dh。主机控制器将在写传输前检测繁忙情况。</li> <li>• 异常中断指令 当执行读传输时，如果设置这个指令，主机控制器将停止读取缓冲区。当执行写传输的时候，如果设置这个指令，主机控制器将停止控制DAT线。发出异常中断指令后，主机驱动器必须发出一个软件复位。 11b =在CCCR 中，为写 “I/O 异常中断” ，中断CMD12, CMD52 10b = 在CCCR中，为写 “功能选择” ，恢复CMD52 01b = 在CCCR中，为写 “总线暂停”，暂停CMD52 00b = 正常其它指令</li> </ul>	
	[5]	<p>当前数据选择。 该位设置为1，表示数据是当前的，将使用数据线传输。 下面情况将设置为0： (1) 指令只用CMD 线（不包括CMD52）。 (2) 指令没有数据传输，但在DAT[0]线上使用繁忙信号。 (R1b或者R5b ，不包括CMD38) (3) 恢复指令 1 = 当前数据 0 = 无当前数据</p>	
	[4]	<p>指令索引检测启动。 如果这位设置为1，主机控制器将在应答中检测索引域是否有相同的值作为指令索引。如果没有，它将作为一个索引错误被提出。如果这位设置为0，不检测索引域。 1 = 有效 0 = 无效</p>	
	[3]	<p>指令CRC检测启动。 如果该位设置为1，主机控制器将检测应答中的CRC域。如果检测到错误，它将作为一个指令CRC错误被报告。如果该位设置为0，不检测CRC域。CRC域值检测到的位的数值依据应答的长度而改变。 1 =有效 0 = 无效</p>	
	[2]	保留。	
	[1: 0]	<p>应答类型选择： 00 = 无应答 01 = 应答长度 136 10 = 应答长度48 11 = 应答长度 48应答后检测繁忙情况</p>	

如表27-2所示，显示了参数名称和应答类型的关系。

表 27-2 参数名称和应答类型的关系

应答类型	索引检查启动	CRC检查启动	应答类型的名称
00	0	0	没有应答
01	0	1	R2
10	0	0	R3、R4
10	1	1	R1、R6、R5
11	1	1	R1B、R5b

注：在SDIO规范中，应答类型符号R5b没有定义。R5包括R5b在SDIO规范中。规范中定义R5b来指定接收应答后主机控制器检查繁忙情况。例如，通常CMD52被作为R5，但是I/O异常中断将被作为R5b。

注：写“总线暂停”后，CMD52来读BS，指令类型也必须为“暂停”。

### 27.4.8. 应答寄存器

该寄存器用来存储来自SD卡的应答。

寄存器	地址	读/写	描述	复位值
RSPREG0_0	0x7C200010	ROC	应答寄存器0(0通道)。	0x0
RSPREG1_0	0x7C200014	ROC	应答寄存器1(0通道)。	0x0
RSPREG2_0	0x7C200018	ROC	应答寄存器2(0通道)。	0x0
RSPREG3_0	0x7C20001C	ROC	应答寄存器3(0通道)。	0x0

寄存器	地址	读/写	描述	初始状态
RSPREG0_1	0x7C300010	ROC	应答寄存器0(1通道)。	0x0
RSPREG1_1	0x7C300014	ROC	应答寄存器1(1通道)。	0x0
RSPREG2_1	0x7C300018	ROC	应答寄存器2(1通道)。	0x0
RSPREG3_1	0x7C30001C	ROC	应答寄存器3(1通道)。	0x0

寄存器	地址	读/写	描述	初始状态
RSPREG0_2	0x7C400010	ROC	应答寄存器0(2通道)。	0x0

RSPREG1_2	0x7C400014	ROC	应答寄存器1(2通道)。	0x0
RSPREG2_2	0x7C400018	ROC	应答寄存器2(2通道)。	0x0
RSPREG3_2	0x7C40001C	ROC	应答寄存器3(2通道)。	0x0

名称	位	描述	初始状态
	[127:0]	指令应答。下表27-4为每一个应答描述了从SD总线到寄存器的指令映射。在这个表中，在表中 R[]指出在SD总线上传输的应答数据的范围，REP[]指出应答寄存器中位的范围。 128位应答位的顺序： {RSPREG3, RSPREG2, RSPREG1, RSPREG0}	

对于每一种应答类型，应答位的定义如表27-3所示。

表 27-3 每一种应答类型，应答位的定义

应答种类	应答含义	应答区域	应答寄存器
R1, R1b (正常应答)	卡状态	R [39:8]	REP [31:0]
R1b (自动 CMD12 应答)	对于自动CMD12卡状态	R [39:8]	REP [127:96]
R2 (CID, CSD 寄存器)	CID或CSD reg. incl.	R [127:8]	REP [119:0]
R3 (OCR 寄存器)	OCR寄存器用于存储	R [39:8]	REP [31:0]
R4 (OCR寄存器)	OCR寄存器用于I/O等	R [39:8]	REP [31:0]
R5, R5b	SDIO应答	R [39:8]	REP [31:0]
R6 (发表的 RCA 应答)	新发表的RCA[31: 16]等	R [39:8]	REP [31:0]

应答域指出了PHYSICAL LAYER SPECIFICATION1.01版本中定义的“应答”的位置。上表27-33显示大部分48(R[47:0])位的长度应答数据有32位(R[39:8])存储在应答寄存器的REP[31:0]。应答类型R1b(自动CMD12 应答)应答数据R[39:8]存储在应答寄存器的REP[127:96]。长136(R[135:0])的应答的120位应答数据(R[127:8])存储在应答寄存器的REP[119:0]。

为了有效地读应答状态，主机控制器只在应答寄存器中存储部分的应答数据。这能使主机驱动器在32位总线系统的读周期有效地读取32位应答数据。对于部分应答，索引域和CRC被主机控制器(由指令寄存器的指令索引检测启动和指令CRC检测启动位指定)检测，并且如果检测到错误则产生一个错误中断。CRC检测的范围由应答的长度决定。如果应答长度是48，主机控制器将检测R[47:1]，如果应答长度是136，主机控制器将检测R[119:1]。

由于主机控制器通过CMD\_wo\_DAT指令可以让多区段数据DAT线同步传输执行，在应答寄存器的高位(REP[127:96])，主机控制器存储自动CMD12应答。CMD\_wo\_DAT应答存储在REP[31:0]。这使主机控制器避免自动CMD12应答写入过多，反之亦然。

当主机控制器修改应答寄存器中的部分位时，像表27-4所示的那样，它将保护未修改的位。

### 27.4.9. 缓冲区数据端口寄存器

32位数据端口寄存器来访问内部缓冲区。

寄存器	地址	读/写	描述	复位值
BDATA0	0x7C200020	读/写	缓冲区数据 寄存器(0通道)。	0x0
BDATA1	0x7C300020	读/写	缓冲区数据 寄存器(1通道)。	0x0
BDATA2	0x7C400020	读/写	缓冲区数据 寄存器(2通道)。	0x0

名称	位	描述	初始状态
		缓冲区数据。 通过32位数据端口寄存器能访问主机控制器缓冲区。	0

### 27.4.10. 当前状态寄存器

该寄存器包含SD指令变元。

寄存器	地址	读/写	描述	复位值
PRNSTS0	0x7C200024	RO/ROC	当前状态寄存器(通道0)。	0x000A0000
PRNSTS1	0x7C300024	RO/ROC	当前状态寄存器(通道1)。	0x000A0000
PRNSTS2	0x7C400024	RO/ROC	当前状态寄存器(通道2)。	0x000A0000

名称	位	描述	初始状态
	[31:25]	保留。	0
	[24]	CMD线信号电平(RO)。 这个状态用于检测DMD线电平是否从错误中恢复，并且调试。 注：CMD端口被映射到SD0_CMD引脚。	0