

	<p>多位必须不设置。这两个默认除数值，能通过容限寄存器中“用于 SD 卡的基础时钟频率”计算。</p> <p>(1) 25MHz 除数值</p> <p>(2) 400kHz 除数值根据 SD 物理规范 1.01 版本和 SDIO 卡规范 1.0 版本，最大 SD 时钟频率是 25 MHz，并且绝不超过这个限制。通过下面的公式设置频率：</p> $\text{时钟频率} = (\text{基础时钟}) / \text{除数}$ <p>因此，选择最小可能的除数，这这个除数使时钟频率少于或者等于目标频率。</p> <p>例如：如果容限寄存器中“基础时钟频率对于 SD 卡”值 33MHz，目标频率是 25MHz，选择 01h 除数值将产生 16.5MHz，它是小于或者等于目标值最近的频率。同样，对于接近一个时钟值 400kHz，除数值 40h 产生 258kHz 最佳的时钟值。</p>		
	[7:4]	保留。	0
	[3]	<p>外部时钟状态。</p> <p>写到“SD 时钟启动”在这个寄存器为 1 后，当 SD 时钟输出是稳定的，该位设置为 1。SD 主机驱动器等待发出指令来开始直到该位设置为 1。</p> <p>(ROC)</p> <p>‘1’ = 就绪</p> <p>‘0’ = 没有就绪</p>	0
	[2]	<p>SD 时钟启动。</p> <p>当写该位为 0，主机控制器停止 SDCLK。当该位为 0，“SDCLK 频率选择”可以被改变。接下来主机控制器将保持相同的时钟频率直到 SDCLK 停止 (停止在 SDCLK=0)。如果在当前状态寄存器清除卡插入，该位将被清除。</p> <p>(RW)</p> <p>‘1’ = 有效</p> <p>‘0’ = 无效</p>	0
	[1]	<p>内部时钟稳定。</p> <p>在该寄存器写“内部时钟稳定”到 1 后，当 SD 时钟稳定，该位设置为 1。SD 主机驱动器将等待设置“SD 时钟启动”直到该位设置为 1。</p> <p>注：当要求设置时间的时钟震荡器用 PLL 时，这个很有意义。(ROC)</p> <p>‘1’ = 就绪</p> <p>‘0’ = 没有就绪</p>	0
	[0]	<p>中断时钟启动。</p> <p>当主机驱动器没有使用主机控制器或者主机控制器等待一个唤醒中断，该位设置为 0。主机控制器必须停止它的内部时钟到非常低能量状态。并且，寄存器将能读和写。当该位设置为 1 时，时钟开始震荡。当时钟震荡平稳，在该寄存器中，主机控制器能被设置“内部时钟状态”为 1。该位不影响卡检测。(RW)</p> <p>‘1’ = 震荡</p> <p>‘0’ = 停止</p>	

27.4.16. 超时控制寄存器

主机控制器初始化时，主机驱动器能根据容限寄存器设置数据超时计数器值。

寄存器	地址	读/写	描述	复位值
CMDREG0	0x7C20002E	读/写	超时控制寄存器(0通道)。	0x0
CMDREG1	0x7C30002E	读/写	超时控制寄存器(1通道)。	0x0
CMDREG2	0x7C40002E	读/写	超时控制寄存器(2通道)。	0x0

名称	位	描述	初始状态
	[7:4]	保留。	0
	[3:0]	<p>数据超时计数器值。</p> <p>这个值决定时间间隔，通过数据线超时设定被检测检测。对于指示超时产生的信息因素，参考错误中断寄存器的“数据超时错误”。通过划分基础时钟 TMCLK 值，产生超时时钟频率。当设置这个寄存器时，通过清除“数据超时错误状态启动”（在错误中断状态启动寄存器中）来预防引起不慎超时事件。</p> <p>1111b 保留 1110b TMCLK x 227 1101b TMCLK x 226 0001b TMCLK x 214 0000b TMCLK x 213</p>	0

27.4.17. 软件复位寄存器

写 1 到这个寄存器的每一位，则产生复位脉冲。复位完成后，主机控制器清除每一位。由于完成软件复位要花一些时间，SD 主机驱动器将确认这些位是 0。

寄存器	地址	读/写	描述	复位值
SWRST0	0x7C20002F	读/写	软件复位寄存器(0通道)。	0x0
SWRST1	0x7C30002F	读/写	软件复位寄存器(1通道)。	0x0
SWRST2	0x7C40002F	读/写	软件复位寄存器(2通道)。	0x0

名称	位	描述	初始状态
	[7:3]	保留。	0
	[2]	<p>软件复位对于 DAT 线。</p> <p>只复位部分数据电路。也复位 DMA 电路。(RWAC)</p> <p>下面的寄存器和位通过该位清除：</p> <p>缓冲区数据端口寄存器：缓冲区被读取和初始化</p> <p>当前状态寄存器：缓冲区读启动、缓冲区写启动、读传输有效、写传输有效、DAT 线有效、指令禁止 (DAT)</p> <p>块间隔控制寄存器：继续请求、停止在块间隔请求</p> <p>正常中断状态寄存器：缓冲区读就绪、缓冲区写就绪、DMA 中断、块间隔事件、传输完成。</p> <p>‘1’ = 复位</p> <p>‘0’ = 工作</p>	0
	[1]	<p>软件复位对于 CMD 线。</p> <p>只有部分指令电路复位。(RWAC)</p> <p>下面的寄存器和位通过该位清除：</p> <p>当前状态寄存器</p> <p>指令禁止 (CMD)</p> <p>正常中断状态寄存器</p> <p>指令完成</p> <p>‘1’ = 复位</p> <p>‘0’ = 工作</p>	0
	[0]	<p>软件复位对于所有的。</p> <p>除了卡检测电路，复位影响整个主机控制器。类型 ROC, RW, RWIC, RWAC 寄存器位清除为 0。在初始化期间，主机驱动器设置这位为 1 来复位主机控制器。当容限寄存器有效并且主机驱动器能读它们，主机控制器复位到 0。另外，使用“软件复位对于所有”不影响容限寄存器的这个值。如果该位设置为 1, SD 卡将复位本身，并且必须被主机驱动器初始化。(RWAC)</p> <p>‘1’ = 复位</p> <p>‘0’ = 工作</p>	0

27.4.18. 正常中断状态寄存器

正常中断状态启动影响这个寄存器的读，但是正常中断信号启动就没有这个影响。当正常中断信号启动激活，并且至少一个状态位设置为 1，一个中断产生。除了卡中断和错误中断位外，对于所有的位写 1 到一个位来清除它；写 0 到这些位保持不变。通过写入一个信号寄存器，超过一个状态能被清除。当卡停止中断有效时，卡中断被清除，也就是，卡驱动器为中断提供服务的条件。

寄存器	地址	读/写	描述	复位值
NORINTSTS0	0x7C200030	ROC/RW1C	正常中断状态寄存器(0 通道)。	0x0
NORINTSTS1	0x7C300030	ROC/RW1C	正常中断状态寄存器(1 通道)。	0x0
NORINTSTS2	0x7C400030	ROC/RW1C	正常中断状态寄存器(2 通道)。	0x0

名称	位	描述	初始状态
	[15]	错误中断。 如果错误中断寄存器的任何位被设置，然后将设置这个位。。因此通过首先检查这个位,对于任何错误,主机驱动器能有效测试。这个位是只读。(ROC) '0' = 无错误 '1' = 错误	0
StaFIA3	[14]	FIFO SD 地址指示器中断 3 状态 (RW1C)。 '0' = 发生 '1' = 没有发生 当 SD 时钟 FIFO 地址达到 FIFO 中断地址寄存器 3 的值,该状态位有效。	0
StaFIA2	[13]	FIFO SD 地址指示器中断 2 状态 (RW1C)。 '0' = 发生 '1' = 没有发生 当 SD 时钟 FIFO 地址达到 FIFO 中断地址寄存器 2 的值,该状态位有效。	0
StaFIA1	[12]	FIFO SD 地址指示器中断 1 状态 (RW1C)。 '0' = 发生 '1' = 没有发生 当 SD 时钟 FIFO 地址达到 FIFO 中断地址寄存器 1 的值,该状态位有效。	0
StaFIA0	[11]	FIFO SD 地址指示器中断 0 状态 (RW1C)。 '0' = 发生 '1' = 没有发生 当 SD 时钟 FIFO 地址达到 FIFO 中断地址寄存器 0 的值,该状态位有效。	0
StaWaitInt	[10]	读等待中断状态 (RW1C)。 '0' = 读等待中断发生'1' = 读等待中断没有发生 注: 当检查暂停指令的应答后,如果 BS = 0,自动释放读等待中断状态。	0
StaCCS	[9]	CCS 中断状态 (RW1C)。 指令完成信号中断状态位用于 CE-ATA 模式。 '0' = CCS 中断发生	0

		' 1' = CCS 中断没有发生	
	[8]	<p>卡 中断。</p> <p>将这位写 1 不清除这位。它被清除通过复位 SD 卡中断因子。在 1 位模式中，主机控制器将检测卡中断，没有 SD 时钟来支持唤醒。在 4 位模式中，中断循环期间，卡中断信号被取样，因此在来自 SD 卡的中断信号和到主机系统的中断之间有许多样本延迟。当这个状态已经被设置，并且主机驱动器需要开始中断服务时，为了清除主机控制器中的卡中断状态锁和停止操纵中断信号到主机系统，正常中断状态启动寄存器中卡中断状态启动必须设置为 0。卡中断服务（它必须复位在 SD 卡的中断因子，并且中断信号可能无效）完成后，设置卡中断状态启动为 1 和再次开始取样中断信号。（ROC, RW1C）</p> <p>' 1' =产生卡中断</p> <p>' 0' = 无卡中断</p>	0
	[7]	<p>卡移除。</p> <p>如果当前状态寄存器的卡插入从 1 变到 0，该状态被设置。当主机驱动器写这位为 1 来清除这个状态，当前状态寄存器中的卡插入状态必须被确认。因为当主机驱动器清除该位时卡检测状态可能改变，并且可能不产生中断事件。（RW1C）</p> <p>' 1' = 移除</p> <p>' 0' = 卡状态稳定或抖动</p>	0
	[6]	<p>卡插入。</p> <p>如果当前状态寄存器中卡插入从 0 变为 1，该位被设置。当主机驱动器将该位置 1 来清除这个状态时，当前状态寄存器的卡插入状态必须被确认。因为当主机驱动器清楚该位时卡检测状态可能改变，并且可能不产生中断事件。（RW1C）</p> <p>' 1' = 卡插入</p> <p>' 0' = 卡状态稳定或抖动</p>	0
	[5]	<p>缓冲区读就绪。</p> <p>如果缓冲区读启动从 0 变到 1，该状态被设置。参考当前状态寄存器中的缓冲区读启动。（RW1C）</p> <p>' 1' = 准备读缓冲区</p> <p>' 0' =没有准备读缓冲区</p>	0
	[4]	<p>缓冲区写就绪。</p> <p>如果缓冲区写启动从 0 变到 1，该状态被设置。参考当前状态寄存器中的缓冲区写启动。（RW1C）</p> <p>' 1' = 准备写入缓冲区</p> <p>' 0' =没有准备写入缓冲区</p>	0
	[3]	<p>DMA 中断。</p> <p>如果传输期间主机控制器检测主机 DMA 缓冲区边界，则该状态被设置。参考块大小寄存器中主机 DMA 缓冲区边界。其它 DMA 中断</p>	0

		因素可能在将来添加。传输完成后，其不能产生中断。(RW1C) ‘1’ = DMA 中断产生 ‘0’ = 非 DMA 中断													
	[2]	块间隔事件。 如果块间隔控制寄存器中“停止在块间隔请求”被设置，当读/写传输在块间隔被停止时，该位被设置。如果“停止在块间隔请求”没有被设置为 1，该位不设置为 1。(RW1C) (1) 在读传输的情况下 该位在 DAT 线活动状态的下降沿（当传输被停止在 SD 总线传输时间选择）被设置。为了使用这个功能，读等待必须停止。 (2) 写传输的情况下 该位在写传输活动状态的下降沿（在 SD 总线时间选择获得 CRC 状态后）被设置。 ‘1’ = 传输被停止在块间隔 ‘0’ = 无块间隔事件	0												
	[1]	传输完成。 当读/写传输完成，该位被设置。 (1) 读传输情况下 该位在读传输活动状态下下降沿被设置。有两种产生中断的情况。第一种是数据传输完成。第二种是数据传输被停止在块间隔，和通过设置块间隔控制寄存器中的“停止在块间隔请求”完成数据传输。 (2) 写传输情况下 该位在 DAT 线活动状态的下降沿被设置。有两种产生中断的情况。第一种是最后的数据被写入 SD 卡，并且释放繁忙信号。第二种情况是数据传输被停止在块间隔，和通过设置块间隔控制寄存器中的“停止在块间隔请求”完成数据传输。 下表显示数据传输完成比数据超时错误有更高的优先权。如果两位都设置为 1，认为数据传输完成。 <table border="1" data-bbox="534 1451 1268 1675"> <thead> <tr> <th>传输完成</th> <th>数据超时错误</th> <th>状态含义</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>因其它因素中断</td> </tr> <tr> <td>0</td> <td>1</td> <td>数据传输过程中超时错误</td> </tr> <tr> <td>1</td> <td>-</td> <td>传输完成</td> </tr> </tbody> </table> ‘1’ = 数据传输完成 ‘0’ = 无数据传输完成	传输完成	数据超时错误	状态含义	0	0	因其它因素中断	0	1	数据传输过程中超时错误	1	-	传输完成	0
传输完成	数据超时错误	状态含义													
0	0	因其它因素中断													
0	1	数据传输过程中超时错误													
1	-	传输完成													
	[0]	指令完成。 当获得指令应答的最后一位（除了自动 CMD12）时，该位被设置。参考当前状态寄存器中指令禁止 (CMD)。下表表明指令超时错误比指令完成有更高的优先权。如果两位都设置位 1，则认为应答没有正确接收。(RW1C)	0												

指令完成	指令超时错误	状态的含义	
0	0	被另一个因素中断	
不关心	1	64 SDCLK 周期内不接收应答	
1	0	应答被接收	
‘1’ =指令完成			
‘0’ = 无指令完成			

注：通过轮询和监视 INTREQ 端口，主机驱动器将检查是否中断真的被清除。如果 HCLK 比 SDCLK 更快，该位要花更多的时间被清除。

注：卡中断状态位保持先前的值直到下一个卡中断（一级中断），并且当置 1 的时候能被清除。

27.4.19. 错误中断状态寄存器

通过错误中断状态启动寄存器能启动这个寄存器中的信号定义，但是不能通过错误中断信号启动寄存器。当错误中断信号启动激活，并且至少一个状态设置为 1，中断产生。写 1 清除这位，写 0 保持这位不变。在一个寄存器写，超过一个状态能被清除。

寄存器	地址	读/写	描述	复位值
ERRINTSTS0	0x7C200032	ROC/RW1C	错误中断状态寄存器(0 通道)。	0x0
ERRINTSTS1	0x7C300032	ROC/RW1C	错误中断状态寄存器(1 通道)。	0x0
ERRINTSTS2	0x7C400032	ROC/RW1C	错误中断状态寄存器(2 通道)。	0x0

名称	位	描述	初始状态
	[15:9]	保留。	0
	[8]	自动 CMD12 错误。 当检测（自动 CMD12 错误状态寄存器中的一位）已经从 0 变为 1 时发生。该位设置为 1，不仅错误在自动 CMD12 发生，而且由于先前的指令错误，自动 CMD12 不被执行。 ‘1’ = 错误 ‘0’ = 无错误	0
	[7]	当前限制错误。 不执行此版本。始终为 0	0
	[6]	数据最后位错误。	0

		当检测 0 在最后位位置（使用 DAT 线）或者 CRC 状态的最后位位置时发生。 ‘1’ = 错误 ‘0’ = 无错误	
	[5]	数据 CRC 错误。 检测 CRC 错误时，用 DAT 线传输读取的数据，或者当检测写 CRC 状态有一个其它的值而不是“010”时，数据 CRC 错误发生 ‘1’ = 错误 ‘0’ = 无错误	0
	[4]	数据超时错误。 当检测到下面的一种时发生： (1) 对于 R1b, R5b 类型忙超时 (2) 写入 CRC 状态后忙超时 (3) 写入 CRC 状态超时 (4) 读数据超时 ‘1’ = 超时 ‘0’ = 无错误	0
	[3]	如果在指令响应中一个指令索引错误产生，指令索引错误产生。 ‘1’ = 错误 ‘0’ = 无错误	0
	[2]	指令末位错误。 当检测到指令的最后位时，错误产生。 ‘1’ = 最后位错误产生 ‘0’ = 无错误	0
	[1]	指令 CRC 错误。 在下面两种情况下指令 CRC 错误产生： (1) 如果应答被返回和指令超时错误被设置为 0(表示没有超时)，当在指令应答检测一个 CRC 错误时，该位设置为 1。 (2) 当发出一个命令，通过监测 CMD 线主机控制器检测一个 CMD 线。如果主机控制器驱动 CMD 线到 1 级，但在下一个 SDCLK 缓冲中检测 CMD 线上的 0 级，那么主机控制器将中止命令（停止驱动 CMD 线）和设置此位为 1。命令超时错误也将被设置为 1 用于区别 CMD 线的冲突。 ‘1’ = CRC 错误产生 ‘0’ = 无错误	0
	[0]	指令超时错误。 如果没有应答在 64 SDCLK 周期内被从指令的最后位返回，则错误发生。如果主机控制器测试到一个 CMD 线冲突，在这种情况下，超时错误将被设置，如表 27-34 所示。由于指令将被主机控制器中断，没有等待 64 SDCLK 周期情况下，该位也将被设置。 ‘1’ = 超时 ‘0’ = 无错误	0

指令 CRC 错误和指令超时错误关系如表 27-4 所示。

表 27-4 指令 CRC 错误和指令超时错误关系

指令 CRC 错误	指令超时错误	错误种类
0	0	无错误
0	1	应答超时错误
1	0	应答 CRC 错误
1	1	CMD 线冲突

27.4.20. 正常中断状态启动寄存器

设置 1 来使中断状态启动。

寄存器	地址	读/写	描述	复位值
NORINTSTSENO	0x7C200034	读/写	正常中断状态启动寄存器(0 通道)。	0x0
NORINTSTSEN1	0x7C300034	读/写	正常中断状态启动寄存器(1 通道)。	0x0
NORINTSTSEN2	0x7C400034	读/写	正常中断状态启动寄存器(2 通道)。	0x0

名称	位	描述	初始状态
	[15]	固定到 0。 通过错误中断状态启动寄存器，主机驱动器将控制错误中断。(RO)	0
EnStaFIA3	[14]	FIFO SD 地址指示字中断 3 状态有效。 ‘1’ = 有效 ‘0’ = 屏蔽	0
EnStaFIA2	[13]	FIFO SD 地址指示字中断 2 状态有效。 ‘1’ = 有效 ‘0’ = 屏蔽	0
EnStaFIA1	[12]	FIFO SD 地址指示字中断 1 状态有效。 ‘1’ = 有效 ‘0’ = 屏蔽	0
EnStaFIA0	[11]	FIFO SD 地址指示字中断 0 状态有效。 ‘1’ = 有效 ‘0’ = 屏蔽	0
EnStaRWait	[10]	读等待中断状态有效。 ‘1’ = 有效 ‘0’ = 屏蔽	0
EnStaCCS	[9]	CCS 中断状态有效。	0

		‘1’ = 有效 ‘0’ = 屏蔽	
	[8]	卡中断状态有效。 如果这个位被设置为 0，主机控制器清除中断请求到系统。这个位被设置为 1 时，这个位被清除和恢复，卡中断检测停止。必须重新这个位从卡被清除到阻止无意的中断，在所有中断请求后，维修卡中断前主机控制器必须再次清除卡中断状态启动。 ‘1’ = 有效 ‘0’ = 屏蔽	0
	[7]	卡移动状态有效。 ‘1’ = 有效 ‘0’ = 屏蔽	0
	[6]	卡插入状态有效。 ‘1’ = 有效 ‘0’ = 屏蔽	0
	[5]	缓冲区读就绪信号有效。 ‘1’ = 有效 ‘0’ = 屏蔽	0
	[4]	缓冲区写就绪信号有效。 ‘1’ = 有效 ‘0’ = 屏蔽	0
	[3]	DMA 中断状态有效。 ‘1’ = 有效 ‘0’ = 屏蔽	0
	[2]	块间隔事件状态有效。 ‘1’ = 有效 ‘0’ = 屏蔽	0
	[1]	传输完成状态有效。 ‘1’ = 有效 ‘0’ = 屏蔽	0
	[0]	指令完成状态有效。 ‘1’ = 有效 ‘0’ = 屏蔽	0

27.4.21. 错误中断状态启动寄存器

设置 1 来激活错误中断状态。

寄存器	地址	读/写	描述	复位值
-----	----	-----	----	-----