

8

DMA

概述

S3C2410A支持位于系统总线和外设总线之间的4个通道的DMA控制器。每一个通道的DMA控制器都能没有约束的实现系统总线或者外设总线之间的数据传输，即每个通道都能处理下面四种情况：

- 1) 源器件和目的器件都在系统总线
- 2) 源器件在系统总线,目的器件在外设总线
- 3) 源器件在外设总线,目的器件在系统总线
- 4) 源器件和目的器件都在外设总线

DMA的主要优点是：可以不通过CPU的干预来实现数据的传输，DMA的运行可以通过软件、内部外设或外部请求引脚信号的请求来初始化。

DMA 请求源

如果在 DCON 寄存器中 H/W DMA 请求模式被选中,每个通道的 DMA 控制器都能够从 4 个 DMA 请求源中选择一个 DMA 请求源。(注意: 如果 S/W 请求模式被选中, 中断请求源的设置没有任何意义) 表 8-1 列出了每一个通道的 4 个 DMA 请求源。

Table 8-1. 每一个通道的DMA请求源

	Source0	Source1	Source2	Source3	Source4
Ch-0	nXDREQ0	UART0	SDI	Timer	USB device EP1
Ch-1	nXDREQ1	UART1	I2SSDI	SPI0	USB device EP2
Ch-2	I2SSDO	I2SSDI	SDI	Timer	USB device EP3
Ch-3	UART2	SDI	SPI1	Timer	USB device EP4

在这里, nXDREQ0和nXDREQ1表示两个外部源(外部设备), I2SSDO和I2SSDI分别表示IIS的发送和接收。

DMA 工作过程

DMA 使用三态 FSM (有限状态机) 进行操作, 以下用三个步骤描述:

State-1. 初始状态, DMA 等待 DMA 请求。若请求到达, 进入状态 2。此阶段, DMA ACK 和 INT REQ 都为 0。

State-2. 在此状态, DMA ACK 变为 1, 计数器的值 (CURR_TC) 从 DCON[19:0]寄存器加载。
注意: DMA ACK 仍然为 1, 直到它随后被清 0。

State-3. 在此状态, 对 DMA 进行原子操作的 sub-FSM (子状态机) 被初始化。sub-FSM (子状态机) 从源地址读取数据然后将数据写入目的地址。在此操作中, 要考虑数据大小和传输的尺寸 (single or burst)。在 Whole service 模式下这种操作重复进行直到计数器 (CURR_TC) 变为 0, 然而在 Single service 模式中只进行一次。当 sub-FSM (子状态机) 完成每一个原子操作, main FSM (主状态机) 对 CURR_TC 减计数。此外, 当 CURR_TC 变为 0 并且中断设置位 DCON[29]为 1 时, main FSM (主状态机) 发出中断请求信号 (INT REQ)。另外, 如果以下条件之一满足, main FSM (主状态机) 清除 DMA ACK 信号。

- 1) 在 Whole service 模式下 CURR_TC 变为 0
- 2) 在 Single service 模式下原子操作完成

注意: 在Single service模式下, main FSM (主状态机) 的三个状态执行然后停止, 再等待另外的DMA REQ。如果DMA REQ到来, 就重复进行这样的三个状态。因此, 每一次原子传输的过程中DMA ACK总是先有效然后再无效。相反, 在Whole service模式下, main FSM (主状态机) 一直在状态3等待, 直到CURR_TC变为0。所以DMA ACK在整个传输过程中有效, 然后当TC为0时无效。

然而, 只要CURR_TC变为0, 中断请求信号 (INT REQ) 总是被发出而与服务模式无关 (Single service模式或是Whole service模式)。

外部 DMA 请求/响应协议

有3种外部DMA请求/响应协议 (Single service Demand, Single service Handshake and Whole service Handshake mode)。每种类型都定义了像DMA请求和DMA响应这些信号是怎样与这些规则相联系的。

基本的 DMA 时序

DMA 服务即在 DMA 操作过程中执行成对的读和写周期。S3C2410A DMA 操作的基本时序如图 8-1 所示。

- 在所有模式下，XnXDREQ 和 XnXDACK 信号的建立时间和延迟时间都相同。
- 如果 XnXDREQ 信号的建立时间满足要求，则在两个周期内实现同步，然后 XnXDACK 信号有效。
- 在 XnXDACK 信号有效后，DMA 请求总线。如果 DMA 得到总线就开始执行 DMA 操作。当 DMA 操作完成后，XnXDACK 信号无效。

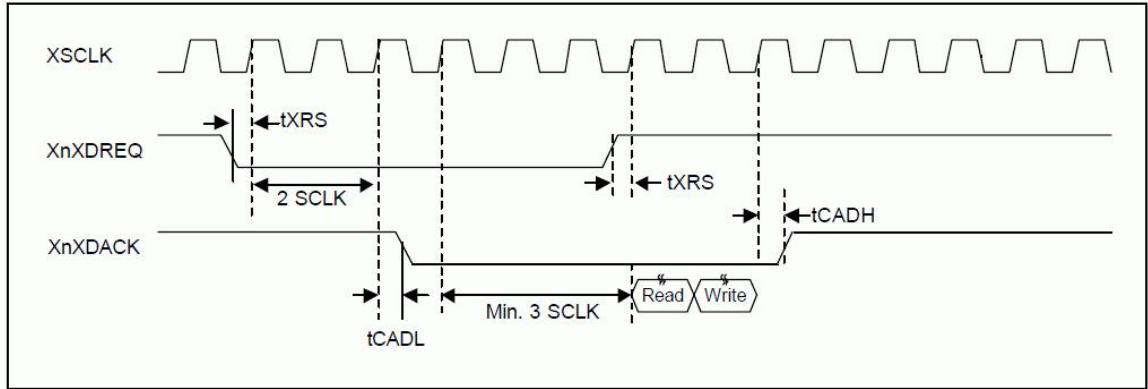


Figure 8-1. Basic DMA Timing Diagram

Table 8-2. DMA Controller Module Signal Timing Constants

($V_{DD} = 1.8V \pm 0.15V$, $2.0V \pm 0.1V$, $T_A = -40$ to $85^\circ C$, $V_{EXT} = 3.3V \pm 0.3V$)

Parameter	Symbol	Min	Typ.	Max	Unit
eXternal Request Setup	t_{XRS}	2	—	6 / 5	ns
aCcess to Ack Delay when Low transition	t_{CADL}	9	—	11 / 10	ns
aCcess to Ack Delay when High transition	t_{CADH}	9	—	11 / 10	ns

请求/握手（Demand/Handshake）模式的比较

请求和握手模式与XnXDREQ和XnXDACK之间的协议相关。两种模式的差别如图8-2所示。

在一次传输的最后（Single或Burst传输），DMA检测XnXDREQ信号的状态。

请求模式（Demand Mode）

- 如果XnXDREQ信号仍然有效，则马上开始下一次的传输。否则等待XnXDREQ信号有效。

握手模式（Handshake Mode）

- 如果XnXDREQ信号无效，DMA在两个周期内使XnXDACK信号无效。否则，一直等待直到XnXDREQ信号无效为止。

警告

只有在XnXDACK信号无效（高电平）之后，XnXDREQ信号才能有效（低电平）。

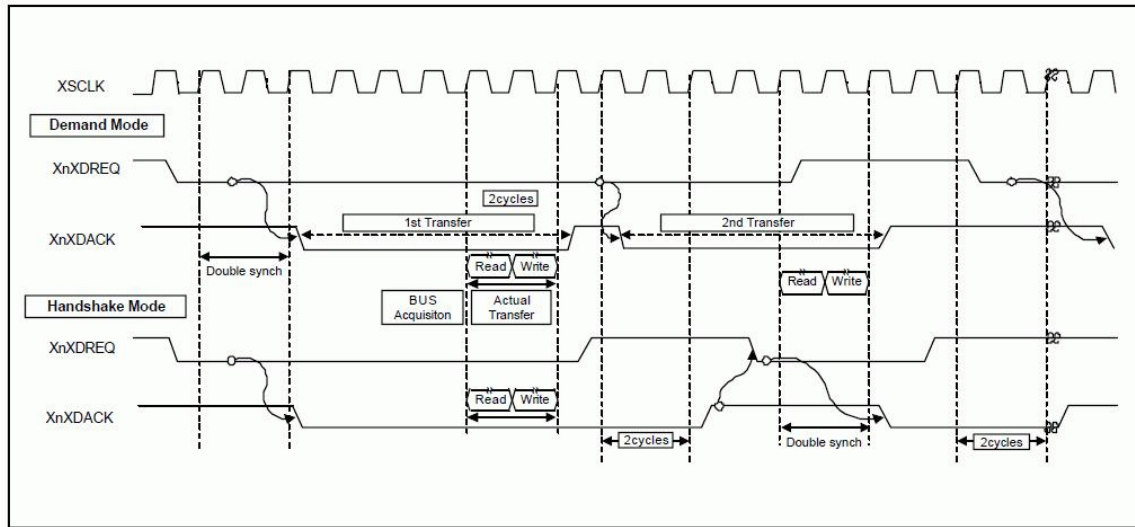


Figure 8-2. Demand/Handshake Mode Comparison

传输尺寸

- 两种不同的传输尺寸：**unit** 和 **Burst 4**
- 在传输这样一块数据时DMA牢牢的占据总线。这样，其它的总线主设备就不能够得到总线使用权。

Burst 4 传输尺寸

在Burst 4传输过程中，4次连续的读和写操作被分别执行。

注意

单位（unit）传输尺寸：一次读和一次写操作被执行。

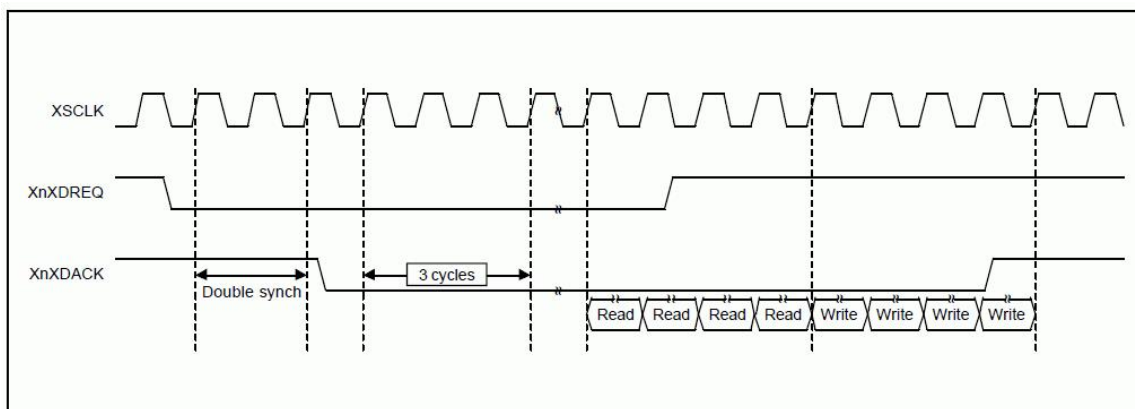


Figure 8-3. Burst 4 Transfer Size

例子

Single service in Demand Mode with Unit Transfer Size

在每一个单元传输过程中XnXDREQ信号都需要有效（Single service mode）。当XnXDREQ信号有效时，操作持续进行（Demand mode），一对读和写（Single transfer size）的操作被执行。

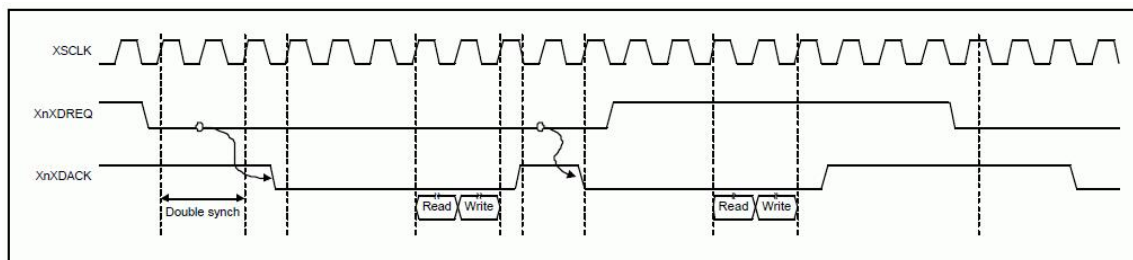


Figure 8-4. Single service in Demand Mode with Unit Transfer Size

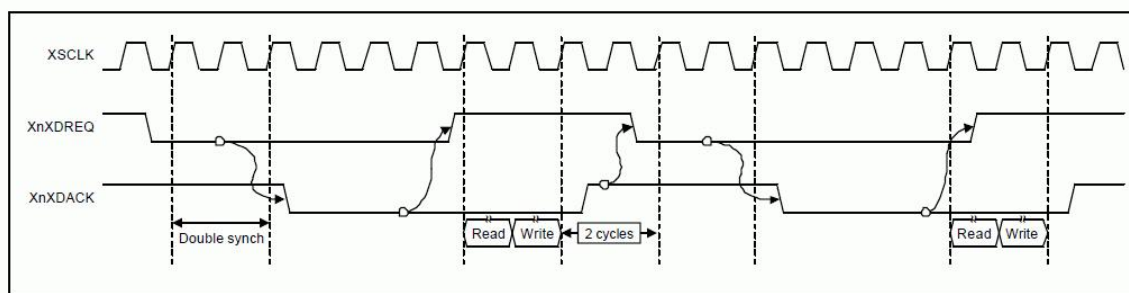
Single service in Handshake Mode with Unit Transfer Size

Figure 8-5. Single service in Handshake Mode with Unit Transfer Size

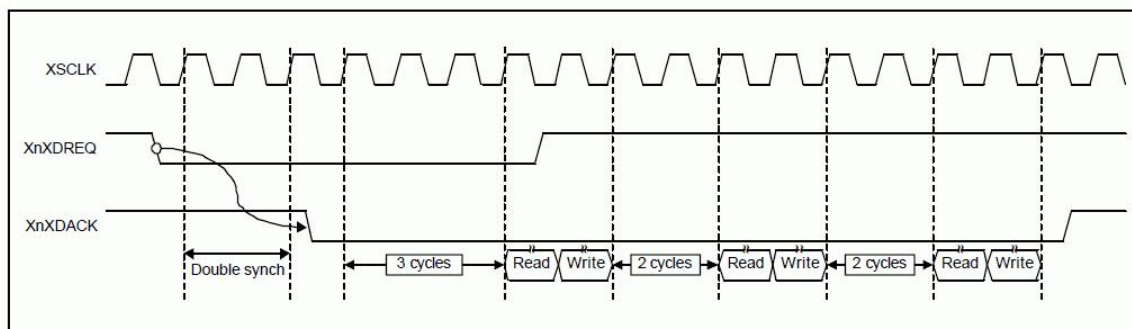
Whole service in Handshake Mode with Unit Transfer Size

Figure 8-6. Whole service in Handshake Mode with Unit Transfer Size

DMA专用寄存器

每一个DMA通道都有9个控制寄存器（4个通道共计36个寄存器）。6个控制寄存器用来控制DMA传输,其它3个监视DMA控制器的状态。这些寄存器的详细情况如下。

DMA初始源地址（DISRC）寄存器

Register	Address	R/W	Description	Reset Value
DISRC0	0x4B000000	R/W	DMA0初始源地址寄存器	0x00000000
DISRC1	0x4B000040	R/W	DMA1初始源地址寄存器	0x00000000
DISRC2	0x4B000080	R/W	DMA2初始源地址寄存器	0x00000000
DISRC3	0x4B0000C0	R/W	DMA3初始源地址寄存器	0x00000000

DISRCn	Bit	Description	Initial State
S_ADDR	[30:0]	要传输的源数据基地址（起始地址）。仅当CURR_SRC为0并且DMA ACK为1时，该值被加载到CURR_SRC。	0x00000000

DMA初始源控制（DISRCC）寄存器

Register	Address	R/W	Description	Reset Value
DISRCC0	0x4B000004	R/W	DMA0初始源控制寄存器	0x00000000
DISRCC1	0x4B000044	R/W	DMA1初始源控制寄存器	0x00000000
DISRCC2	0x4B000084	R/W	DMA2初始源控制寄存器	0x00000000
DISRCC3	0x4B0000C4	R/W	DMA3初始源控制寄存器	0x00000000

DISRCCn	Bit	Description	Initial State
LOC	[1]	Bit 1 用来选择源总线的位置 0: 源数据位于系统总线（AHB） 1: 源数据位于外设总线（APB）	0
INC	[0]	Bit 0 用来选择源地址增量方式 0: 每一次传输完毕，地址值增加相应的数据尺寸 1: 每一次传输完毕，地址值不变 （在burst传输模式下，传输过程中地址值会增加。但是传输完毕后，地址值会被恢复原值）	0

DMA初始目的地址（DIDST）寄存器

Register	Address	R/W	Description	Reset Value
DIDST0	0x4B000008	R/W	DMA0初始目的地址寄存器	0x00000000
DIDST1	0x4B000048	R/W	DMA1初始目的地址寄存器	0x00000000
DIDST2	0x4B000088	R/W	DMA2初始目的地址寄存器	0x00000000
DIDST3	0x4B0000B8	R/W	DMA3初始目的地址寄存器	0x00000000

DIDSTn	Bit	Description	Initial State
D_ADDR	[30:0]	传输目的地的基地址（起始地址）。仅当CURR_DST为0并且DMA ACK为1时，该值被加载到CURR_SRC。	0x00000000

DMA初始目的控制（DIDSTC）寄存器

Register	Address	R/W	Description	Reset Value
DIDSTC0	0x4B00000C	R/W	DMA0初始目的控制寄存器	0x00000000
DIDSTC1	0x4B00004C	R/W	DMA1初始目的控制寄存器	0x00000000
DIDSTC2	0x4B00008C	R/W	DMA2初始目的控制寄存器	0x00000000
DIDSTC3	0x4B0000CC	R/W	DMA3初始目的控制寄存器	0x00000000

DIDSTCn	Bit	Description	Initial State
LOC	[1]	Bit 1 用来选择目的总线的位置 0: 目的数据位于系统总线（AHB） 1: 目的数据位于外设总线（APB）	0
INC	[0]	Bit 0 用来选择目的地址增量方式 0: 每一次传输完毕，地址值增加相应的数据尺寸 1: 每一次传输完毕，地址值不变 （在burst传输模式下，传输过程中地址值会增加。但是传输完毕后，地址值会被恢复原值）	0

DMA控制（DCON）寄存器

Register	Address	R/W	Description	Reset Value
DCON0	0x4B000010	R/W	DMA0控制寄存器	0x00000000
DCON1	0x4B000050	R/W	DMA1控制寄存器	0x00000000
DCON2	0x4B000090	R/W	DMA2控制寄存器	0x00000000
DCON3	0x4B0000D0	R/W	DMA3控制寄存器	0x00000000

DCONn	Bit	Description	Initial State
DMD_HS	[31]	选择请求（Demand）模式或是握手（Handshake）模式 0: 选择请求（Demand）模式 1: 选择握手（Handshake）模式 在这两种模式下，DMA控制器都会在DREQ信号到来时开始数据传输并且使DACK信号有效。这两种模式的差别在于是否等待DACK（译者注：此处应为DREQ，但是三星公司手册上确实为DACK）信号无效。在握手（Handshake）模式下，DMA控制器在开始下一次传输之前要一直等待直到DREQ信号无效。如果DREQ信号无效了，DMA控制器无效DACK信号然后等待下一次的DREQ信号有效。与之相反，在请求（Demand）模式下，DMA控制器不等待DREQ信号无效。如果传输完毕后DREQ信号还是继续有效，DMA控制器只是先无效DACK信号然后又开始新一轮的传输。我们建议对外部DMA请求使用握手（Handshake）模式，以避免不经意的开始新一轮数据传输。	0
SYNC	[30]	选择DREQ/DACK信号的同步方式 0: DREQ和DACK信号与PCLK同步（APB时钟） 1: DREQ和DACK信号与HCLK同步（AHB时钟） 因此，对于连接在AHB总线上的设备该位应该置1，而对于连接在APB总线上的设备该位应该置0。对于连接在外部系统上的设备，该位的设置取决于该外部系统是与AHB还是与APB系统同步。	0
INT	[29]	CURR_TC的中断请求控制 0: 禁止CURR_TC产生中断请求。用户必须检查状态寄存器中的传输计数值（例如：查询模式）。 1: 当所有的传输完成时产生中断请求（例如：CURR_TC变为0）	0
TSZ	[28]	原子传输尺寸选择 （例如：数据传输总是在DMA拥有总线后，释放总线之前执行） 0: 执行单数据传输 1: 执行四数据长的突发传输	0

DMA控制（DCON）寄存器（续表）

DCONn	Bit	Description	Initial State
SERVMODE	[27]	传输模式选择 0: 单服务传输（Single service）模式。每一次原子传输（单数据或四数据长的突发传输）后都要停下来等待下一次DMA请求信号有效。 1: 全服务传输（Whole service）模式。DMA请求到来后，原子传输重复进行直到传输计数值为0。在此模式下，不需要额外的请求信号。 注意：就算是在全服务传输（Whole service）模式下，每一次原子传输后DMA也会释放总线，然后再试图重新获得总线，以保证其它总线主设备能够有机会得到总线使用权。	0
HWSRCSEL	[26:24]	各DMA通道请求源设置 DCON0: 000:nXDREQ0 001:UART0 010:SDI 011:Timer 100:USB device EP1 DCON1: 000:nXDREQ1 001:UART1 010:I2SSDI 011:SPI 100:USB device EP2 DCON2: 000:I2SSDO 001:I2SSDI 010:SDI 011:Timer 100:USB device EP3 DCON3: 000:UART2 001:SDI 010:SPI 011:Timer 100:USB device EP4 这些位控制4选1的多路选择器来为每一个DMA选择请求源。只有DCONn[23]选中H/W请求模式时，这些位的选择才有意义。	00
SWHW_SEL	[23]	DMA源选择方式(软件或硬件)设置 0: 以软件方式产生DMA请求,需要用DMASKTRIG控制寄存器中的SW_TRIG位设置触发。 1: 由位[26:24]提供的DMA源触发DMA操作。	0
RELOAD	[22]	再装载选择 0: 自动再装载，当传输计数值减为0时自动装载DMA初值 1: 不自动再装载，当传输计数值减为0时关闭DMA通道。DMA通道开/关位（DMASKTRIGn[1]）被清零以防止不经意的开始下一次DMA操作。	0
DSZ	[21:20]	传输数据尺寸设置 00: 字节 01: 半字 10: 字 11: 保留	00
TC	[19:0]	传输计数值初值 注意：实际传输字节数用下式计算DSZ x TSZ x TC DSZ代表传输数据尺寸、TSZ代表传输尺寸、TC代表传输计数值初值 仅在CURR_SRC为0并且DMA ACK为1时，该值被加载到CURR_SRC。	00000

DMA状态（DSTAT）寄存器

Register	Address	R/W	Description	Reset Value
DSTAT0	0x4B000014	R	DMA0状态寄存器	0x00000000
DSTAT1	0x4B000054	R	DMA1状态寄存器	0x00000000
DSTAT2	0x4B000094	R	DMA2状态寄存器	0x00000000
DSTAT3	0x4B0000D4	R	DMA3状态寄存器	0x00000000

DSTATn	Bit	Description	Initial State
STAT	[21:20]	DMA控制器状态 00: DMA控制器已经准备好，可以接受下一个DMA请求。 01: DMA控制器忙。	00
CURR_TC	[19:0]	传输计数值 注意：传输计数值被DCONn[19:0]初始化，在每一次原子传输结束后该值减1。	00000

DMA当前源地址（DCSRC）寄存器

Register	Address	R/W	Description	Reset Value
DCSRC0	0x4B000018	R	DMA0当前源地址寄存器	0x00000000
DCSRC1	0x4B000058	R	DMA1当前源地址寄存器	0x00000000
DCSRC2	0x4B000098	R	DMA2当前源地址寄存器	0x00000000
DCSRC3	0x4B0000D8	R	DMA3当前源地址寄存器	0x00000000

DCSRCn	Bit	Description	Initial State
CURR_SRC	[30:0]	DMA _n 当前源地址寄存器	0x00000000

DMA当前目的地址（DCDST）寄存器

Register	Address	R/W	Description	Reset Value
DCDST0	0x4B00001C	R	DMA0当前目的地址寄存器	0x00000000
DCDST1	0x4B00005C	R	DMA1当前目的地址寄存器	0x00000000
DCDST2	0x4B00009C	R	DMA2当前目的地址寄存器	0x00000000
DCDST3	0x4B0000DC	R	DMA3当前目的地址寄存器	0x00000000

DCDSTn	Bit	Description	Initial State
CURR_DST	[30:0]	DMA _n 当前目的地址寄存器	0x00000000

DMA掩码触发寄存器（DMASKTRIG）寄存器

Register	Address	R/W	Description	Reset Value
DMASKTRIG0	0x4B000020	R/W	DMA0掩码触发寄存器	0x000
DMASKTRIG1	0x4B000060	R/W	DMA1掩码触发寄存器	0x000
DMASKTRIG2	0x4B0000A0	R/W	DMA2掩码触发寄存器	0x000
DMASKTRIG3	0x4B0000E0	R/W	DMA3掩码触发寄存器	0x000

DMASKTRIGn	Bit	Description	Initial State
STOP	[2]	停止DMA操作 1: 在当前的原子数据传输完成后尽快停止DMA。如果此时没有正在运行的原子传输，DMA会立即停止。CURR_TC将为0。 注意：由于可能正在进行原子传输，停止操作会需要几个周期。在DMA通道开启/关闭位（DMASKTRIGn[1]）被设为关闭时，表示DMA操作已经实际上停止下来。	0
ON_OFF	[1]	DMA通道开启/关闭位 0: DMA通道关闭。（该通道的DMA请求将被忽略） 1: DMA通道开启，DMA请求将被处理。 如果设置了DCONn[22]位为不自动再装载，或设置了停止位（DMASKTRIGn[2]）位为停止，该位会自动清零以关闭DMA通道。注意：当DCONn[22]位为不自动再装载，在CURR_TC为0时该位清零。如果停止位为1，该位在当前的原子数据传输完成后立刻为0。 注意：在DMA操作过程中该位不应该人为改变（例如：要改变该位，应当仅仅使用DCON[22]或停止位）。	0
SW_TRIG	[0]	实现软件触发DMA请求 1: 发送DMA请求到该控制器。 注意：在软件触发模式被选中（DCONn[23]），以及DMA通道开启/关闭位为1（通道开启）以后软件触发才起作用。当DMA操作开始后，该位自动清零。	0

注意：可以自由的改变DISRC寄存器，DIDST寄存器，和DCON寄存器中TC域的值。这些改变只有在当前的传输完成后才起作用（例如：当CURR_TC为0时）。另外，对其它寄存器和/或其它域的改变会立刻起作用。因此，改变它们时要小心。

S/W Work-Around

DMA自动再装载仅仅只发生在当DMA请求到来，然后DMA计数值为0时。所以，在DMA结束后的中断处理程序中，在为下一次传输设置DMA源地址、目的地址和计数器寄存器之前应该使用下面一段代码。这一段代码将会等待DMA请求到来并且等待先前的自动再装载值已经被加载。

```
while((rDSTATn&0xffff)==0);
```