

第八章

直接存储器存取

概述

s3c2440 支持在系统总线和外围总线之间的 4 路 DMA 控制器。每一路 DMA 控制器可以自由的执行数据在系统总线和（或）外围总线之间的动作。换句话说，每一路可以在下列列 4 种情况下运行：

- 1.源设备和目标设备都在系统总线上
- 2.源设备在系统总线上，目标设备在外围总线上
- 3.源设备在外围总线上，目标数据在系统总线上
- 4.源设备和目标设备都在外围总线上

DMA 的主要优点是它可以不通过 CPU 直接调用数据。DMA 工执行可以软件开始，也可以从内部的外设请求或者外部的引脚请求开始。

DMA 请求源

每一路 DMA 控制器都可以从 4 种开始请求中选择一路，如果 DCON 寄存器选择 H/W DAM 请求模式（注意，选择 S/W 时这个 DMA 请求源没意义）。表 8--1 列出了 4 种请求和每一路的关系：

表 8--1.DMA 开始请求和每一路关系

	Source0	Source1	Source2	Source3	Source4	Source5	Source6
Ch-0	nXDREQ0	UART0	SDI	Timer	USB device EP1	I2SSDO	PCMIN
Ch-1	nXDREQ1	UART1	I2SSDI	SPI0	USB device EP2	PCMOUT	SDI
Ch-2	I2SSDO	I2SSDI	SDI	Timer	USB device EP3	PCMIN	MICIN
Ch-3	UART2	SDI	SPI1	Timer	USB device EP4	MICIN	PCMOUT

这里 nXDREQ0 和 nXDREQ1 表示两个外部源（External Devices），I2SSDO 和 I2SSDI 分别表示 IIS 的传输和接收。

DMA 工作状态

DMA 应用 3 态 FSM（有限状态机），它可以描述为以下 3 步：

- State--1. 作为起始状态，DMA 等待 DMA 请求，一旦请求到来马上转到 State--2，在这个状态，DMA ACK 和 INT REQ 是 0.
- State--2. 在这个状态，DMA ACK 变为 1，计数器（CURR_TC）从 DCON[19:0]装载。DMAACK 保持 1，直到它被清 0。
- State--3. 在这个状态，控制 DMA 原子操作的子 FSM 被初始化了。子 FSM 从源地址读取数据后把它写入目标地址。在这个操作中，数据宽度和传输大小（单次或突发）应给与考虑。在完整服务模式中，这种操作不断重复直到计数器

(CURR_TC) 变为 0。然而在单一模式中只进行一次，当子 FSM 完成每个原子操作时，主 FSM 对 CURR_TC 倒计时。另外，当 CURR_TC 为 0 和中断设置 DCON[29] 为 1 时，主 FSM 发出 INT REQ 信号（中断请求信号）。另外，如果发生下列情况之一，DMA ACK 将被清零。

- 1) 在整体服务模式中 CURR_TC 变为 0
- 2) 在单一模式中原子操作完成

注：在单一服务模式中，主 FSM 的三个状态进行然后停止，等待另外的 DMA REQ。如果 DMA REQ 到来，重复三种状态。因此，DMA ACK 被使能然后被取消后进行每一个原子操作。相反的，在整体服务模式中，主 FSM 等待在 state-3 直到 CURR_TC 变为 0。所以，在 TC 到达 0 时，DMA ACK 在传输区间被使能然后被取消。

总之，INT REQ 被使能只由 CURR_TC 是否为 0 决定，与服务模式无关（单一服务模式或整体服务模式）。

外部 DMA 请求/应答 (DREQ/DACK) 协议

有三种类型的外部 DMA 请求/应答协议（单一需求模式、单一同步交换模式和整体同步交换模式）。每一种类型定义这些协议涉及到什么样的 DMA 请求和应答信号。

基本 DMA 时序图

DMA 服务意味着在 DMA 操作中执行一对读、写循环，它可以考试一个 DMA 操作。图 8--1 显示了 S3C2440 DMA 操作中的继电器

- XnXDREQ 和 XnXDACK 中的建立时间 (setup) 和延时时间 (delay) 在所有模式下都是相同的
- 如果 XnXDREQ 完成遇上建立时间，那么它将同步加强为原来的两倍，这是 XnXDACK 被声明了
- XnXDACK 声明以后，DMA 请求总线，如果到达总线则执行其操作，DMA 操作完成后运行 XnXDACK

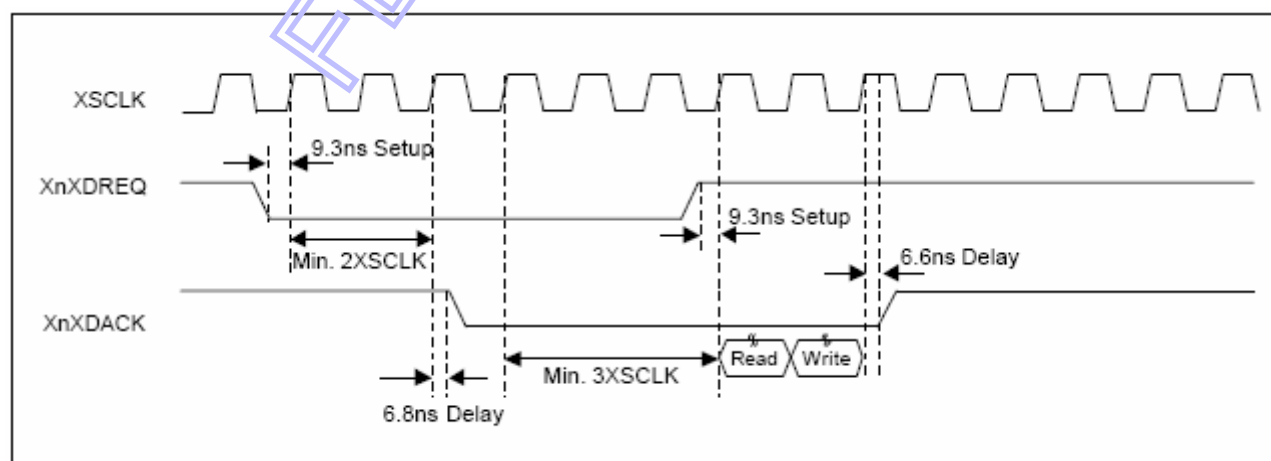


图 8--1. 基本 DMA 时序图

需求/同步交换模式比较

需求和同步交换模式都涉及到XnXDREQ和XnXDACK协议。图8--2指出了两种模式之间的不同

在操作（单一/突发操作）最后，DMA组织双同步XnXDREQ。

需求模式

- 如果XnXDREQ尚未使能，那么下一个操作马上开始。否则它一直等XnXDREQ被设为无效。

同步交换模式

- 如果XnXDREQ无效，DMA取消XnXDACK两个周期。否怎，一直等XnXDREQ被设为无效。

注：XnXDREQ只在XnXDACK无效（high）后被使能（low）

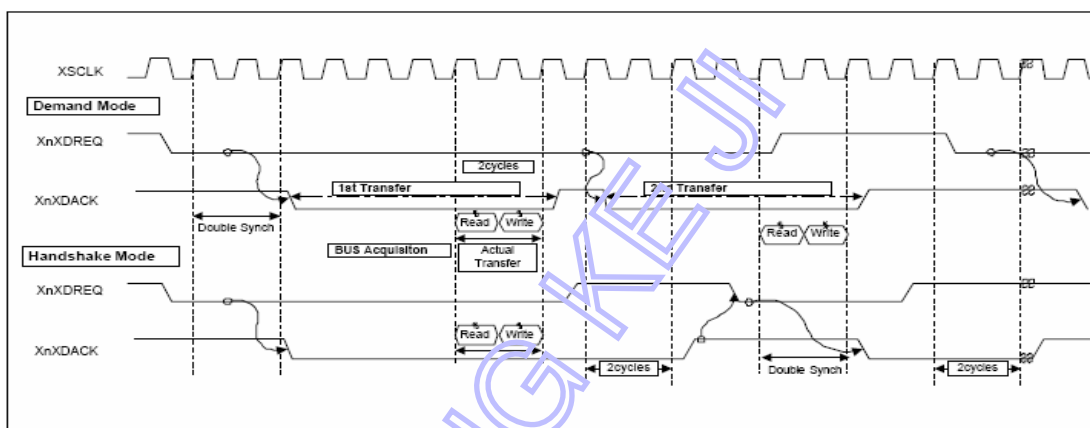


图8—2需求模式和同步交换模式比较

传送容量

- 有两种不同的传输方式，unit和Burst 4
- 在传输大量数据时，DMA牢牢占据总线，因此其他总线主控不能得到总线

Burst 4传输容量

有四组连续的读、写操作在Burst 4中分别完成

注 Unit传输型：执行单读、写

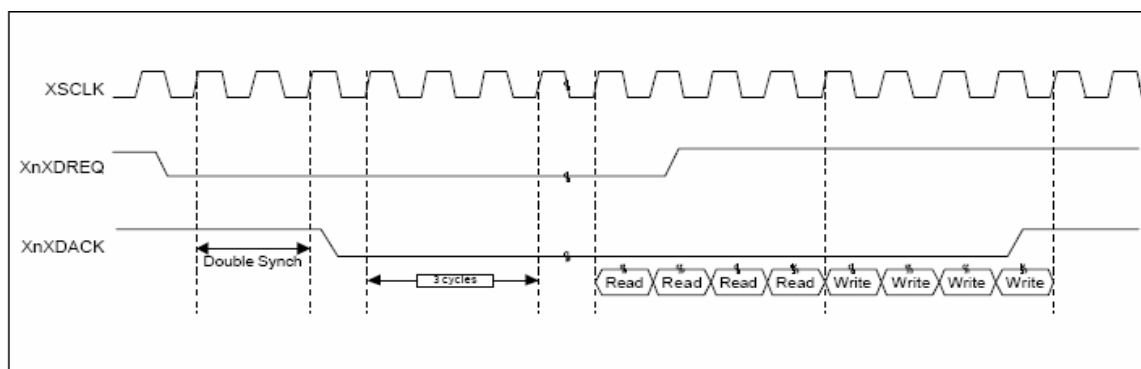


图8--3.Burst 4传输型

举例

Unit传输容量的单一服务需求模式

每一个Unit传输（单一服务模式）都需要XnXDREQ使能。这个操作一直延续到XnXDREQ被使能（需求模式），一对读、写（单一传输容量）操作被执行。

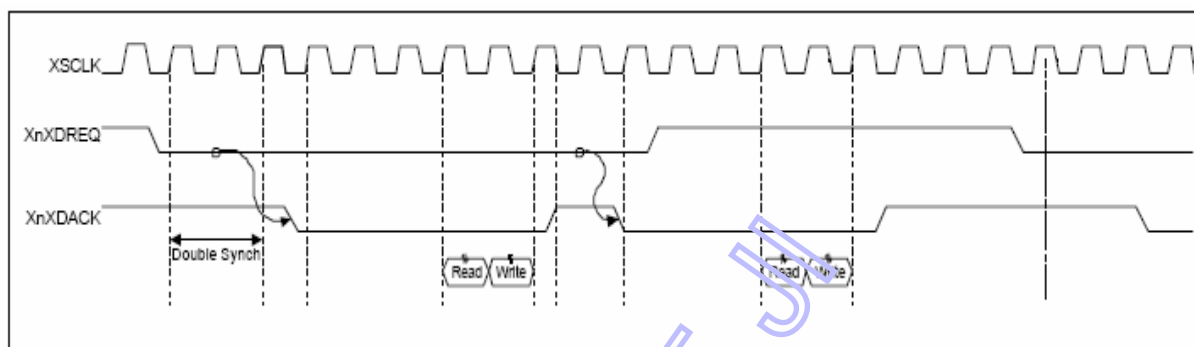


图8--4.UnitUnit传输容量的单一服务需求模式

Unit传输容量的单一服务单同步交换模式

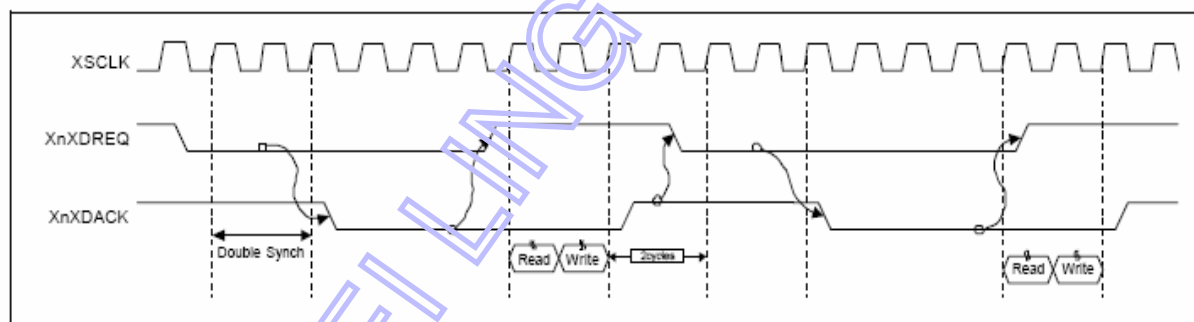


图8--5.Unit传输容量的单一服务单同步交换模式

Unit传输容量的整体服务同步交换模式

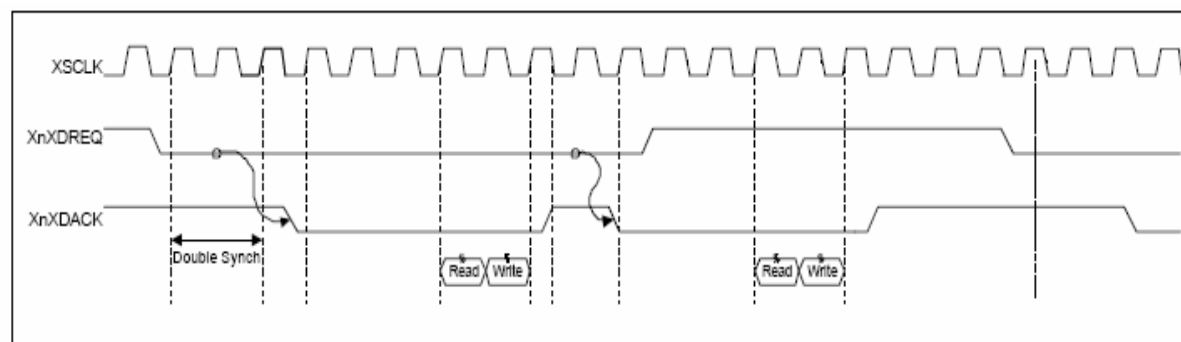


图8--6.Unit传输容量的整体服务同步交换模式

DMA专用寄存器

每一路DMA有九个控制寄存器（4路一共有36个）。其中6个控制寄存器控制DMA传输，另外的3个监听DMA控制器状态。详细说明如下。

DMA初始源寄存器 DMA INITIAL SOURCE (DISRC) REGISTER

寄存器	地址	R/W	说明	重新赋值
DISRC0	0x4B000000	R/W	DMA 0 initial source register	0x00000000
DISRC1	0x4B000000	R/W	DMA 1 initial source register	0x00000000
DISRC2	0x4B000000	R/W	DMA 2 initial source register	0x00000000
DISRC3	0x4B000000	R/W	DMA 3 initial source register	0x00000000

DISRCn	Bit	说明	初始状态
S_ADDR	[30:0]	源数据传输基地址（起始地址）。当CURR_SRC值为0，DMA ACK值为1时，这个bit值被写入CURR_SRC	0x00000000

DMA初始源控制寄存器 DMA INITIAL SOURCE CONTROL (DISRCC) REGISTER

寄存器	地址	R/W	说明	重新赋值
DISRC0	0x4B000004	R/W	DMA 0 initial source control register	0x00000000
DISRC1	0x4B000044	R/W	DMA 1 initial source control register	0x00000000
DISRC2	0x4B000084	R/W	DMA 2 initial source control register	0x00000000
DISRC3	0x4B0000C4	R/W	DMA 3 initial source control register	0x00000000

DISRCCn	Bit	说明	初始状态
LOC	[1]	Bit 1用来选择源位置 0: 源在系统总线（AHB）上 1: 源在外围设备总线（APB）上	0
INC	[0]	Bit 0用来选择地址增量 0=增加 1=不变 如果为0，在突发和单一传出模式传输之后地址随数据量增加。 如果为1，地址在传输后不变。（在突发模式，地址在传输过程中增加，但是在传输结束后恢复到初始值。）	0

DMA初始目的寄存器 DMA INITIAL DESTINATION (DIDST) REGISTER

寄存器	地址	R/W	说明	重新赋值
DIDST0	0x4B000008	R/W	DMA 0 initial destination register	0x00000000
DIDST1	0x4B000048	R/W	DMA 1 initial destination register	0x00000000
DIDST2	0x4B000088	R/W	DMA 2 initial destination register	0x00000000
DIDST3	0x4B0000B8	R/W	DMA 3 initial destination register	0x00000000

DIDSTn	Bit	说明	初始状态
D_ADDR	[30:0]	目标传输基地址（起始地址）。如果CURR_DST为0，DMA ACK为1，这个bit值被写入CURR_SRC	0x00000000

DMA初始目的控制寄存器 DMA INITIAL DESTINATION (DIDST) REGISTER

寄存器	地址	R/W	说明	重新赋值
DIDSTC0	0x4B00000C	R/W	DMA 0 initial destination control register	0x00000000
DIDSTC1	0x4B00004C	R/W	DMA 1 initial destination control register	0x00000000
DIDSTC2	0x4B00008C	R/W	DMA 2 initial destination control register	0x00000000
DIDSTC3	0x4B0000CC	R/W	DMA 3 initial destination control register	0x00000000

DIDSTCn	Bit	说明	初始状态
CHK_INT	[2]	在自动重装载被设置时，选择中断发生的时间。 0: TC到达0时发生中断。 1: 自动重装载完成后发生中断。	0
LOC	[1]	Bit1用来选择目标文件位置 0: 目标文件在系统总线上（AHB） 1: 目标文件在外部设备总线上（APB）	0
INC	[0]	Bit 0用来选择地址增量 0=增加 1=不变 如果为0，在所有单一和突发传输模式之后地址随数据量增加。	0

		如果为1，传输之后地址不变。（在突发模式中，地址在传输过程中增加，但在传输之后恢复成原来的值。）	
--	--	--	--

DMA控制寄存器 DMA CONTROL (DCON) REGISTER

寄存器	地址	R/W	说明	重新赋值
DCON0	0x4B000010	R/W	DMA 0 source register	0x00000000
DCON1	0x4B000050	R/W	DMA1 source register	0x00000000
DCON2	0x4B000090	R/W	DMA 2 source register	0x00000000
DCON3	0x4B0000D0	R/W	DMA 3 source register	0x00000000

DCONn	Bit	说明	初始状态
DMD_HS	[31]	在需求模式和同步交换模式中选一种 0: 选择需求模式 1: 选择同步交换模式 这两种模式中，DMA控制器开始传输并通过给DACK使能从而给DREQ使能。这两种模式的区别在于是否等待DACK被无效与否 在同步交换模式，DMA控制器在开始新的传输前等待DREQ被设为无效。如果发现DREQ无效，马上把DACK设为无效，并等待DREQ另一次使能。 相反的，在需求模式，DMA控制器不等待DREQ被设为无效。如果DREQ使能，直接将DACK设为无效并开是下一次传输。	0
SYNC	[30]	DREQ/DACK同步选择 0: DREQ和DACK同步到达PCLK(APB 时钟). 1: DREQ和DACK同步到达HCLK(APB 时钟) 因此，为了设置在AHB系统总线上，这位必须置1，为了设置在APB系统总线上，这位需要置0. 为了设置在外系统总线上，使用者需要依靠外部系统和AHB系统和APB系统同步来选择这位	0
INT	[29]	CURR_TC（数字终端）禁止和允许中断 0: CURR_TC中断禁止，使用者必须查看寄存器形式中的传输个数（轮巡检测） 1: 所有传输结束后产生中断请求（CURR_TC为0）	0
TSZ	[28]	选择原子操作传输类型（传输执行任一时刻DMA在释放总线之前占有总线） 0: 执行一个unit传输 1: 执行一个4 burst传输	0

SERVMODE	[27]	<p>在单一服务模式和整体同步模式中选择</p> <p>0: 所有原子操作之后, DMA停止并等待另一个DMA请求时选择单一服务模式</p> <p>1: 一个请求到达原子操作并被重复直到传输个数为0.在这个模式中额外要求是不必要的</p> <p>注意, 在整体服务模式, 所有原子操作之后DMA释放总线, 然后试着再占据总线, 防止其他总线主控得不到总线</p>	0
HWSRCSEL	[26:24]	<p>每个DMA请求源</p> <p>DCON0: 000:nXDREQ0 001:UART0 010:SDI 011:Timer 100:USB device EP1</p> <p>DCON1: 000:nXDREQ1 001:UART1 010:I2SSDI 011:SPI 100:USB device EP2</p> <p>DCON2: 000:I2SSD0 001:I2SSDI 010:SDI 011:Timer 100:USB device EP3</p> <p>DCON3: 000:UART2 001:SDI 010:SPI 011:Timer 100:USB device EP4</p> <p>DCON0: 101:I2SSD0 110:PCMIN DCON1: 101:PCMOUT 110:SDI DCON2: 101:PCMIN 110:MICIN DCON3: 101:MICIN 110:PCMOUT</p> <p>这些位通过控制4-1 MUX来每个DMA请求源, 只有DCONn[23]选择H/W请求模式时这些位才有意义</p>	00
SWHWSEL	[23]	<p>在软件 (S/W请求模式) 和硬件 (H/W请求模式) 中选择DMA源</p> <p>0: 通过DMASKTRIG控制寄存器的SW_TRIG位选择S/W请求模式和DMA触发</p> <p>1: 通过[26:24]位触发DMA动作选择DMA源</p>	0
RELOAD	[22]	<p>设置重启开关</p> <p>0: 当前传输个数变为0自动重启 (执行所有必要传输)</p> <p>1: 当当前传输个数变为0时关闭DMA通路(DMA REQ)。通路开关位(DMASKTRIGn[1])置0(DREQ off), 防止进一步非本意开始新的DMA动作</p>	0
DSZ	[21:20]	<p>数据类型转换</p> <p>00 = Byte 01 = Half word</p>	00

		10 = Word 11 = reserved	
--	--	-------------------------------------	--

TC	[19:0]	初始传输个数 传输的字节数有下列公式计算： $DSZ \times TSZ \times TC$ 。 其中DSZ,TSZ (1 or 4)和TC分别表示 DCONn[21:20]数据量，DCONn[28]传输量和初 始传输个数。只有到CURR_TC为0并且DMA ACK 为1时，这个之被写入CURR_TC	00000
-----------	---------------	--	--------------

DMA状态寄存器 DMA CONTROL (DCON) REGISTER

寄存器	地址	R/W	说明	重新赋值
DSTAT0	0x4B000014	R	DMA 0 计数寄存器	000000h
DSTAT1	0x4B000054	R	DMA1 计数寄存器	000000h
DSTAT2	0x4B000094	R	DMA 2 计数寄存器	000000h
DSTAT3	0x4B0000D4	R	DMA 3 计数寄存器	000000h

DSTATn	Bit	说明	初始状态
STAT	[21:20]	这个控制器的形式 00: 表示DMA控制器为下一个DMA请求做准备 01: 表示DMA控制器正在在传输过程中	00b
CURR_TC	[19:0]	传输个数的当前值 传输个数通过所有原子操作的最后一个，初始调 整DCONn[19:0]寄存器的值和减少量。	00000h

当前DMA源寄存器 DMA CURRENT SOURCE (DCSRC) REGISTER

寄存器	地址	R/W	说明	重新赋值
DCSRC0	0x4B000018	R	DMA 0 当前源寄存器	0x00000000
DCSRC1	0x4B000058	R	DMA1 当前源寄存器	0x00000000
DCSRC2	0x4B000098	R	DMA 2 当前源寄存器	0x00000000
DCSRC3	0x4B0000D8	R	DMA 3 当前源寄存器	0x00000000

DCSRCn	Bit	说明	初始状态
CURR_SRC	[30:0]	适合DMA n的当前源地址	0x00000000

当前目的寄存器 **CURRENT DESTINATION (DCDST) REGISTER**

寄存器	地址	R/W	说明	重新赋值
DCDST0	0x4B00001C	R	DMA0 当前目的寄存器	0x00000000
DCDST1	0x4B00005C	R	DMA1 当前目的寄存器	0x00000000
DCDST2	0x4B00009C	R	DMA 2 当前目的寄存器	0x00000000
DCDST3	0x4B0000DC	R	DMA 3 当前目的寄存器	0x00000000

DCDSTn	Bit	说明	初始状态
CURR_DST	[30:0]	适合DMA _n 当前目标地址	0x00000000

DMA模板触发寄存器 **(DMA MASK TRIGGER (DMASKTRIG) REGISTER)**

寄存器	地址	R/W	说明	重新赋值
DMASKTRIG0	0x4B000020	R	DMA0 mask trigger register	000
DMASKTRIG1	0x4B000060	R	DMA1 mask trigger register	000
DMASKTRIG2	0x4B0000A0	R	DMA 2 mask trigger register	000
DMASKTRIG3	0x4B0000E0	R	DMA 3 mask trigger register	000

DMASKTRIGn	Bit	说明	初始状态
STOP	[2]	<p>停止DMA操作</p> <p>1: 当前原子操作结束DMA马上停止。如果当前没有运行原子操作，DMA立即停止。 CURR_TC, CURR_SRC和CURR_DST将被置0</p> <p>注意：由于可能发生当前原子操作，‘stop’操作可能循环几次。操作的完成（实际停止时间）在引导位置于off同时被发现。这个停止是“实际停止”。</p>	0
ON_OFF	[1]	<p>DMA引导开关位</p> <p>0: 关闭DMA引导（引导不理睬DMA请求）</p> <p>1: DMA引导开启和DMA请求是可以控制的。如果我们把DCONn[22]位放在“不自动重启”和（或）DMASKTRIGn的STOP位在“stop”，这位会自动关闭。</p>	0

		当 DCON[22] 位是“不自动重启”，这位在 CURR_TC 为 0 时变为 0 。如果 STOP 是 1 ，这位在当前远在操作结束时马上变为 0 。	
SW_TRIG	[0]	S/W 请求模式触发 DMA 1 : 向此控制器请求 DMA 动作 这个引导在必须选择 S/W 请求模式必须和 ON_OFF 位必须置 1 （导通）。当 DMA 开始运行，这位自动清零。	0

注意： 你可以修改DISRC，DIDST和TC field of DCON这些寄存器的置。这些修改只在完成当前调用（如CURR_TC变为0）后生效。另一方面，任何修改都会对其他寄存器和（或）领域产生直接影响。因此，谨慎修改这些寄存器和领域。

FEELING KEYJU