

# DSP 芯片中双通道 DMA 的研究与设计

魏建磊, 王 茹

(西安微电子技术研究所, 陕西 西安 710054)

**摘 要:**在 DSP 芯片中采用 DMA 技术, 可将 DSP 从数据传输任务中解放出来, 专门从事复杂的信号处理任务, 提高系统整体性能。文中介绍一种高性能 32 位双通道可编程 DMA 控制器的设计, 重点讨论了该 DMA 的主要功能、设计构思、控制单元及主要部分的电路实现。通过对具体电路的分析和讲解把相应功能模块进行了介绍。通过在 Cadence (R) NC-verilog 上的仿真结果检测了 DMA 工作正确性。

**关键词:**数字信号处理器; 直接存储器存取; 外部存储器接口; 中断; 仿真

**中图分类号:** TP333

**文献标识码:** A

**文章编号:** 1673-629X(2008)03-0036-04

## Design and Simulation of 2-Channel DMA in DSP

WEI Jian-lei, WANG Ru

(Xi'an Microelectronics Technology Institute, Xi'an 710054, China)

**Abstract:** Adopting the technology of DMA in DSP chips, could liberate DSP from the assignment of the data transmission and let it engage in the task of complicated signal processing wholly, which could enhance the whole performance of system. Presents a programmable 2-channel & 32-bit direct memory access controller with high performance in DSP. And it emphasizes on the discussion of the primary function, design concept, control unit and the circuit realization of the main part in DMA. And it introduces the corresponding function of module through the analysis and the explanation to the special circuits. Finally, the simulation result on Cadence(R) NC is used to detect the correctness of DMA felicitously.

**Key words:** DSP; DMA; EMI; interrupt; simulation

## 0 引言

数字信号处理器(Digital Signal Processor, DSP)适用于对大量实时数据进行复杂运算的系统,它具有功能强、处理速度快、性能价格比好以及速度功耗比高等特点,已被广泛应用于具有高性能、高速度要求的场合,特别是互联网、通讯、家电、智能设备、仪器仪表、军事、航天等高速信息处理场合<sup>[1]</sup>。

直接存储器存取(DMA, Direct Memory Access)方式主要用于需要高速大批量进行数据传送的系统中,以提高数据的吞吐量。在一般的程序控制传送的方式(包括编程 I/O 和中断驱动 I/O)下,数据从外设到存储器,或从存储器到外设,都要经过 CPU 的累加器中转,若再加上检查 I/O 状态以及修改内存地址等操作,花费不少时间<sup>[2]</sup>。采用 DMA 传送方式是让存储器(高速器件)与高速外设(如 A/D 数据转换器等)直接

交换数据,不需 CPU 干预,减少了中间环节,并且内存地址的修改、数据传送完毕的结果报告都由硬件完成<sup>[3]</sup>。这样,数据传送就只取决于存储器的存取时间,因此大大提高了传送速度。

文中讨论的双通道 DMA 在设计上的主要特点是: a. 用于存储器与存储器之间数据传输的通道有两个,这样可以更好地缓解当 DMA 对优先级较低的数据进行传输时接收到优先级更高的数据传输任务时所产生的数据丢失问题; b. 数据的传输宽度为 32 位,可来往于处理器映射中任何位置之间,如:传输可以在片内存储器、片外存储器及片内串口之间进行; c. 由于该 DMA 有其独立的内部地址和数据总线支持,所以可以实现 CPU 与 DMA 控制器的操作并发进行,即 DMA 传输速率与 CPU 相等; d. 设定了 CPU/DMA 优先级和 DMA 内部两通道间的两种优先级判别模式。

## 1 基本框架设计

如果 DMA 要将一数据块从存储器的一个区域传输至另一个区域,应顺序执行以下过程: CPU 首先对

收稿日期: 2007-06-23

作者简介: 魏建磊(1982-),男,天津人,硕士研究生,研究方向为嵌入式计算机体系结构;王 茹,硕士研究生,研究方向为嵌入式计算机体系结构;导师:沈绪榜,中国科学院院士。

DMA 控制器(DMAC)初始化,告诉 DMAC 数据的传输参数,当 I/O 数据准备好时,DMAC 向 CPU 发出总线请求,通过优先级仲裁后,CPU 决定是否交出总线控制权。若 CPU 响应这个请求,把自己驱动总线的驱动器关掉,令总线处在浮空状态,并给 DMAC 发送中断响应信号,随后,开始存储器与 I/O 之间的数据传输。在规定的传输次数到达时,DMAC 在把总线的控制权退还给 CPU<sup>[4]</sup>。传输过程示意图如图 1 所示。

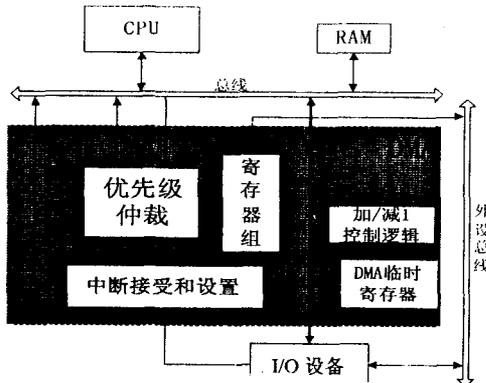


图 1 DMA 传输过程示意图

其中,寄存器组由全局状态控制寄存器、源地址/目标地址寄存器以及传输个数寄存器组成,地址的加/减 1 操作由加/减 1 控制逻辑部分完成,再加上 DMA 同步以及对 CPU 的中断设置模块共同组成了 DMA 的控制单元,该部分在下文做详细介绍;优先级仲裁单元的设计也将在下文中着重介绍。

该 DMA 能够与慢速外部存储器及外设接口进行数据通信,同时并不减小 CPU 总的处理能力,将数据从一个存储空间传送到另一个存储空间,并且这些存储器可以是片内的,也可以是片外的<sup>[5]</sup>。此外,对于 8/16 位的整形数据,将在芯片中的外部存储器接口(EMI, External Memory Interface)模块中转换为 32 位以后实现 DMA 传输。

## 2 控制单元设计

从上面的分析可知控制单元的主要组成部分,其中,寄存器组是必不可少的,下面逐一进行介绍。

### 2.1 全局控制状态寄存器

全局状态控制寄存器保留系统对可变参数(如传输的起始和目的地址、需要传输的个数、同步模式等)的配置,这些配置将直接参与各部分的控制<sup>[6]</sup>。

在一次传输的开始,全局控制寄存器接收 CPU 写入的命令字,在传输过程中控制 DMAC 工作的状态;传输结束时,用下一次的控制信号覆盖当前状态,没有传输任务时状态寄存器清零。以 DMA0 做为例子进行介绍,如表 1 所示。

表 1 DMA0 全局控制寄存器格式

位数	14	13	12	11	10	9	8
意义	PRI MODE	DMA0 PRI		TCINT	TC	SYNC	
读写情况	R/W	R/W		R/W	R/W	R/W	
位数	7	6	5	4	3	2	1 0
意义	DECDST	INCDST	DECSRC	INCSRC	STAT	START	
读写情况	R/W	R/W	R/W	R/W	R/W	R/W	

START 位控制 DMA 开始(11)和停止(00);STAT 为状态位,每个周期改变一次,只能读不能写;DECDST, INCDST, DECSRC, INCSRC 位分别表示目标地址减少、目标地址增加、源地址减少、源地址增加;SYNC 位表示同步模式;TC=1 表示当传输个数计数器为 0 时传输过程终止;TCINT=1 表示设置 CPU 中断;DMA0 PRI 位表示 CPU/DMA 的优先级模式;PRI MODE 位表示 DMA0/DMA1 两个通道的优先级模式。

### 2.2 源与目标地址寄存器

源地址寄存器和目标地址寄存器分别指定待传输的数据(块)在存储器中的读出和写入的首地址,数据宽度为 24 位,可寻址 16M 的存储空间。并在传输开始后,根据全局控制寄存器设置的四个控制位 DECSRC, INCSRC, DECDST 和 INCDST 使源地址寄存器、目标地址寄存器的值在完成相应的存储器存取后增加、减小或不变。

### 2.3 传输个数计数器寄存器

记录待传输的数据块的大小即字节数,数据宽度为 24 位,并在开始传输数据后实现自动减 1 操作,然后通过一个比较 0 逻辑判断当前输出是否等于零。如果计数寄存器的当前值为 1,那么 DMA 通道将再完成一次传输后被中止。因此,DMA 通道可以完成的最小传输字数目为 1。如果事先将 DMA 全局状态控制器中的 TCINT 位设置为 1,则传输个数计数器在计数到 0 时可产生一个 DMA 中断,并且如果 TC=1 传输结束,否则将重复进行。

### 2.4 加/减 1 控制逻辑

DMA 根据在全局状态控制寄存器中控制源与目标地址增加或减少的控制位,再配合读写允许信号进行加/减 1 的逻辑控制,电路实现如图 2 所示。

W/R\_E 为 DMA 读/写允许信号,W/R\_E=1 时执行写操作,对目标地址进行加/减 1 或者不变的操作;W/R\_E=0 时执行读操作,对源地址进行加/减 1 或者不变的操作。DMA\_E 为内部两通道选通信号,DMA\_E=1 选通该通道,DMA\_E=0 关闭该通道;全局状态控制寄存器的四个控制位 DECSRC, INCSRC, DECDST 和 INCDST 分别表示源地址减 1、源地址加 1、目的地址减 1 和目的地址加 1 四种操作;输出 DEC/INC 用于控制加/减 1 操作,当 DEC/INC=1 执行减 1

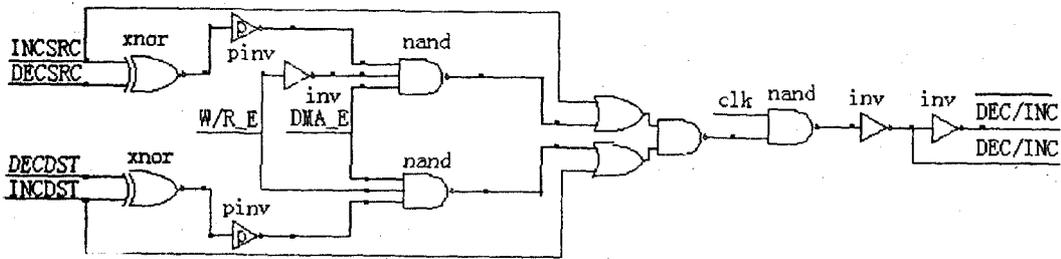


图 2 加/减 1 控制逻辑

操作,同理,DEC/INC=0 执行加 1 操作。

### 2.5 中断的设置和接收单元

\* 中断的设置:对发给 DMA 的来自不同部件的中断优先级排序已经在 CPU 中断控制中进行了设定,因此,一旦有被允许的较高优先级的 DMA 中断出现,则优先级较低的中断即被屏蔽,并产生一个有 DMA 中断的判断信号 INT\_DMA=1;否则,表示尚无被允许的 DMA 中断出现。

\* 中断的接收(即同步):SYNC[1:0]是全局状态控制寄存器中的同步模式设置位,可以实现的同步方式有四种,分别为无同步(SYNC[1:0]=00)、源同步(SYNC[1:0]=01)、目标同步(SYNC[1:0]=10)和源及目标同步(SYNC[1:0]=11)。输入信号 ST 表示 DMA 传输所处的状态:DMA 准备进行读操作时,ST=1;DMA 准备进行写操作时,ST=0。INT\_ACC=1 表示中断信号被屏蔽,DMA 处于忙碌状态;INT\_ACC=0 表示中断信号可以被接受。表达式如下:

$$INT\_ACC = \overline{SYNC<0> \cdot ST + SYNC<1> \cdot \overline{ST}}$$

### 2.6 优先级仲裁单元

#### 2.6.1 DMA/CPU 优先级

DMAC 利用其内部 DMA 总线传输数据,当与 CPU 的存取发生冲突时,仲裁是必不可少的。当它们之间不存在访问冲突时,CPU 和 DMA 控制器的存取是并行完成的<sup>[7]</sup>。仲裁规则由 DMA 通道的 DMA PRI 位(即全局状态控制寄存器的第 12 位和第 13 位)界定,该值由用户设置,在重新编程之前不发生变化,详见表 2。

表 2 全局控制寄存器 DMA PRI 位功能

DMA PRI[1:0]	功能
00	DMA 的优先级低于 CPU 的存取优先级
01	保留
10	环形仲裁
11	DMA 的优先级高于 CPU 的存取优先级

系统复位时,全局控制寄存器将被清零,系统默认 CPU 优先;环形仲裁,是指设置 CPU 和 DMA 通道的优先级使它们轮流存取(但不对等)。当它们在连续的指令周期中发生访问冲突时,CPU 具有较高的优先级;若在下一个指令周期中,两者再次发生对同样资源的访问冲突时,DMA 具有更高优先级。

信号 CIR 为环形仲裁的结果,CIR=1 时,表示该模块 CPU 有存取请求。若此时,DMA 两个通道中至少有一个对同一资源也提出了存取请求,则二者存在冲突。CPU 或 DMA 的优先级哪个较高是由全局状态控制位 DMA PRI[1:0]决定的,详见表 2,该功能的实现电路如图 3 所示。

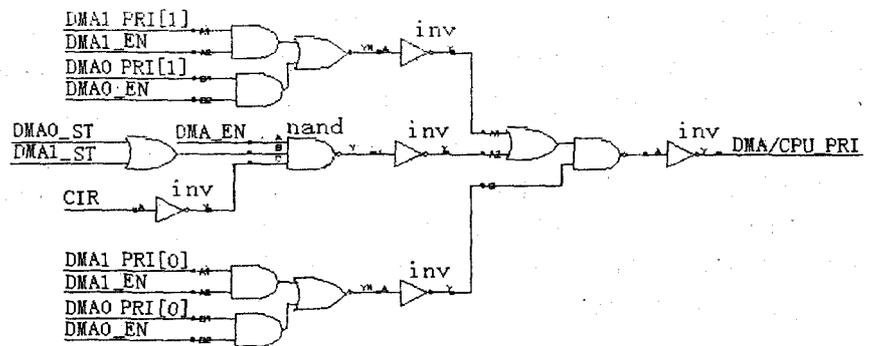


图 3 DMA/CPU 优先级仲裁单元

#### 2.6.2 CPU 内部两通道间优先级

由于该 DMA 为双通道,在进行数据存取时共用同一内部数据总线和地址总线,所以要求有一个内部仲裁的优先级配置以避免两通道的冲突,从而分时占用总线,在此设计两种优先级配置方式,如下:

\* 固定优先级配置。即通道 0 的优先级较高,通道 1 的优先级较低。通过仿真实例来说明固定优先级配置,详见下文。

\* 旋转优先级配置。即最近服务的通道优先级降至最低(复位后装入默认值)。该模块的控制电路主要由两部分组成,前一部分通过接受全局状态控制寄存器中 PRI\_MODE 位来判断内部优先级为哪种方式,并与复位信号相或控制门控时钟的开关;后一部分进行内部优先级仲裁,决定哪个通道优先级较高,如图 4 所示。

DMA0\_ST 和 DMA1\_ST 是两个通道占用 DMA 数据总线的状态信号,当其中任何一个为低电平,即只有一个通道发出了传输请求时,输出信号与 PRI\_MODE 的设置无关;当它们都为高电平,即两通道都发出了请求时,输出信号将由 PRI\_MODE 的设置决定。当 FIX\_PRI 控制的一路被选通时,输出信号  $DMA0\_EN=1, DMA1\_EN=0$ , DMA 通道 0 将优先占用 DMA 总线传输数据。而当 ROT\_PRI 控制的一路被选通时,输出信号的取值与前一周期正好相反,即二者的优先级在不断交替。

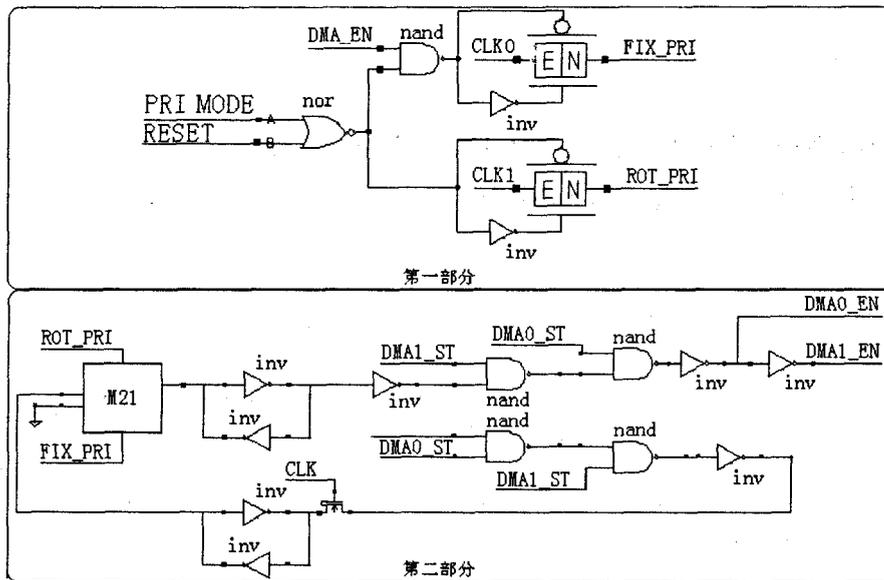


图 4 内部旋转优先级仲裁单元

### 3 仿真结果及分析

文中使用的仿真工具是 Cadence (R) NC-verilog,对 DMA 在两种优先级模式下的工作状态进行了仿真和分析。

#### 3.1 固定优先级模式

在固定优先级模式中,先设定  $\{ST[1]\_DMA1, ST[2]\_DMA1\} = 11$ ,即 DMA1 首先被启动进行传输,在 DMA1 刚从地址为 800005(源地址)的存储单元中取出数据 00000006 暂存在临时寄存器后,再设定  $\{ST[1]\_DMA1, ST[2]\_DMA1\} = 11$ ,即 DMA0 传输被启动,由于内部优先级被设置为固定优先级,所以 DMA 地址和数据总线的控制权交给 DMA0, DMA1 的传输被挂起, DMA0 的传输开始。此时将上周期 DMA1 存在临时寄存器的数据 00000005 放到此时 DMA0 所指的目的地地址  $DST\_DMA0 = 809800$  中。

#### 3.2 旋转优先级模式

在旋转优先级模式中,两个 DMA 的存取过程采

取源同步方式进行数据传输(即只有中断到来时才执行读操作,然后执行写操作)。

在第一个中断 INT1、INT0 到来时有 DMA0 执行读存在 80010 中的数据 00000011,由于是旋转优先级, DMA0 执行读存在 800003 中的数据 00000004,且由于采用的是读同步方式,写操作是在随后依次执行的,最后,等待第二个中断 INT1、INT0 的到来完成类似的操作。

## 4 结束语

讨论了 DMA 控制器的主要功能和基本框架结构,并在此基础上进行了满足要求的功能单元设计,进而实现了对电路的设计。该 DMA 可以完成 32 位整数的传输,且具有双通道、同步方式和优先级可根据需要进行配置等特点。并根据仿真结果使预期的设想得以实现。不过,在对 DMA 的设计方面还存在需要改进的地方,比如:可以考虑是否能将对源及目标地址寄存器进行加/减 1 运算的单元与寄存器合并,从而减少芯片面积;可以考虑是否能够计算出较为折中的通道数,实现 DMA 性能优越性与控

制复杂性的比值最大。

#### 参考文献:

- [1] 王爱英. 计算机组成与系统结构[M]. 第 3 版. 北京:清华大学出版社,2002.
- [2] 张丹红,游珍珠. DSP 的多领域应用研究[J]. 计算机技术与发展,2006,16(3):206-207.
- [3] 李方慧,王飞,何佩琨. TMS320C6000 系列 DSPs 原理与应用[M]. 第 2 版. 北京:电子工业出版社,2003:112-119.
- [4] 钱刚. 嵌入式定浮点合并的 FALU 研究与设计[D]. 西安:西安微电子技术研究所,2002:44-50.
- [5] Rabaey J M, Chandrakasan A, Nikolic B. Digital Integrated Circuits: A Design Perspective[M]. 2nd ed. [s. l.]: [s. n.], 2003:330-340.
- [6] 唐威. DMA 控制器的设计与仿真[J]. 微电子学与计算机,2002(12):2-3.
- [7] 张骏. DMA 控制器的优化和改进[J]. 计算机工程与应用,2005(18):4-5.