

基于 DSP 和小波分析技术的实时噪声消除系统

董光波^{1,2}, 谢桂海², 孙增圻¹

(1. 清华大学计算机系, 北京 100084; 2. 军械工程学院计算机工程系, 石家庄 050003)

摘要: 利用 TMS320VC5502 微处理器和小波分析技术, 实现了对强噪声语音信号的分析与去噪处理, 在 DSP/BIOS 控制机制下建立了基于 DMA、双缓冲区切换和中断响应机制等技术的音频信号采集与处理的程序框架, 并对小波频带阈值去噪、DMA 与外设接口配置、DSP 内存使用等问题提出了具体的解决方法, 为数字信号处理领域相关问题的研究提供了有效例证。

关键词: DSP; 小波分析; 噪声消除; TMS320VC5502

Real-time Noise Cancellation System Based on DSP and Wavelet

DONG Guangbo^{1,2}, XIE Guihai², SUN Zengqi¹

(1. Department of Computer, Tsinghua University, Beijing 100084;

2. Department of Computer Engineering, Ordnance Engineering College, Shijiazhuang 050003)

【Abstract】 Using TI's TMS320VC5502 and wavelet analysis technology, the paper implements the analysis and cancellation processed of audio signals within the high noise, creates the program infrastructure of audio signal sampling and processing based on DMA data transfer, double buffer switched and interrupt response mechanism under the control of DSP/BIOS. To the question of wavelet frequency band threshold noise cancellation, interface configuration between DMA and peripheral equipments, using method of DSP memory and so on, the paper presents concrete resolvents, which gives an efficient illustration for the research of related problems in the field of digital signal processing.

【Key words】 DSP; Wavelet analysis; Noise cancellation; TMS320VC5502

DSP 处理器以其强大的功能、灵活性和高性价比等优势成为音频信号处理领域的研究热点。TI、ADI、Agere 和 M 等公司不断推出的新产品更加推动了 DSP 在该领域的应用研究, 基于 DSP 的语音编码、压缩、识别、增强和去噪等技术得到了突飞猛进的发展。

小波分析作为数字信号处理领域的先进技术被广泛应用于各个领域, 特别是针对非平稳的强噪声信号的分析与处理更显示出良好的性能。小波分析快速算法和具有良好适应性的小波基函数的构建方法被不断改进, 并在应用中取得了较好的效果。

本文以某实时噪声消除系统的研究为背景, 利用 TI 公司的 TMS320VC5502 微处理器和小波分析技术, 实现了对强噪声语音信号的分析与去噪处理, 开发了具有较好自适应性的有源抗噪耳罩。该系统在某基地强噪声环境下语音指令的传达应用中取得了良好的效果。

1 系统结构设计

为保证较好的通用性和学术研究价值, 本系统首先以 TMS320VC5502 EVM 板、PC 机和 TDS560 仿真器为开发平台, 在 CCS2.0 集成环境中实现了所有系统模块的设计与仿真开发, 建立了具有指导意义的程序开发框架, 并利用实测数据验证了系统的效果。

1.1 TMS320VC5502 及 EVM 开发板简介

TMS320VC5502 是基于 TMS320VC55XTM 系列 CPU 处理器核心的定点微处理器, 利用改进的并发和电源管理技术获得了高性能和低能耗, 以独特的内部总线结构实现单指令周期内多条读写指令的执行和独立于 DSP CPU 的 DMA 数据传输。本系统使用的 Z 型号 CPU 时钟频率是 200MHz, 具有

16kB 指令 Cache, 6kB 的双臂存取 RAM (DARAM), 大大提高了容量和存取性能。片内集成了功能强大的外围设备, 主要有 6 通道的 DMA 控制器、McBSP、I2C、UART 等, 为特定系统的开发提供强有力的支持^[1]。

TDS5502 EVM 板是集成有 TMS320VC5502 DSP 芯片和大量外设的、能独立运行的开发和测试平台。提供符合 JTAG 标准的仿真接口, 方便地通过 CCS 环境与 PC 机进行通信。板上的 TLV320AIC23 模块为本系统音频信号的采集、传输和回放功能的实现提供了硬件保障。利用该开发板和相关辅助开发工具, 能够方便地实现系统的仿真开发与测试, 提高开发效率^[2]。

1.2 系统结构

本系统充分利用了 TMS320VC5502 EVM 开发板的设备资源。利用 TLV320AIC23B stereo audio CODEC 的高性能模数转换器 (ADC)、数模数据器 (DAC)、过采样技术 (Oversampling) 和基于 McBSP 可编程接口的音频输入输出模块, 实现音频信号的采集与回放; 利用 McBSP 和 DMA 实现独立于 DSP CPU 的高效的音频数据传输功能; 利用 DSP/BIOS 的中断响应机制和 PING-PONG 双缓冲区切换技术, 实现系统的流程控制; 利用优化的 DSP Library 指令实现基于 C 语言的小波分析数据处理模块。

系统的总体结构如图 1 所示。

TDS560 仿真器负责 EVM 板与 PC 机之间的实时通信。在 CCS 开发环境中完成源代码后, 利用该仿真器加载进 DSP

作者简介: 董光波(1976—), 男, 博士, 研究方向: 数字信号处理; 谢桂海、孙增圻, 教授、博导

收稿日期: 2005-02-17 E-mail: donggb@mail.tsinghua.edu.cn

CPU, 接收 CCS 指令并完成相应的响应处理。程序源代码、数据空间以及其他代码都在 DSP 的片内存储空间 DARAM 中, 一旦程序开始执行, 如果没有分析和测试操作, 其执行过程与 PC 机无关。因此, 该系统结构既仿真了 DSP 的工作过程, 又提高了开发效率, 并能利用 CCS 方便地进行调试。

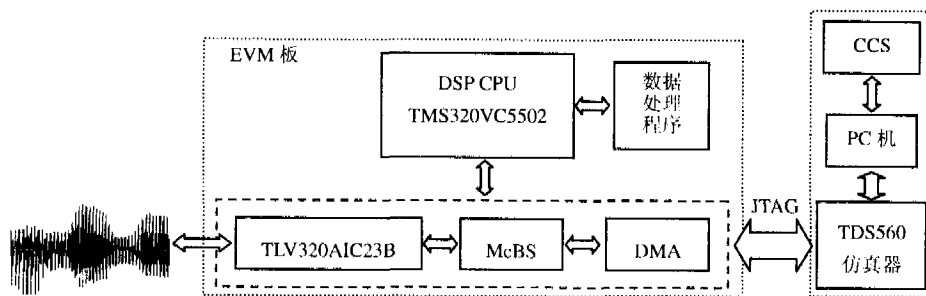


图1 系统结构

2 软件设计与算法

2.1 信号的采集与传输

2.1.1 基于 TLV320AIC23B 的信号采集与回放

系统开发的最终目标是以集成小波分析去噪软件的 DSP 为核心的有源抗噪耳罩, 应用于强噪声环境下语音指令的传达, 因此, 音频信号的输入接口应是 Microphone, 采集的信号数据经过模数转换、滤波后进入 DSP 的片内存储空间, 然后调用集成的小波分析去噪软件进行去噪处理, 得到的新数据经数模转换、放大后被输出到耳机。

基于上述需求, 在系统设计时充分利用了 TMS320VC5502 EVM 开发板上的 TLV320AIC23B stereo audio CODEC 模块。TLV320AIC23B 是一个高性能的多媒体数字信号编解码器, 该模块具有模数转换器、数模转换器, 使用了高效的 Multibit sigma-delta 和过采样数字插补滤波技术, 支持 16、20、24 和 32b 格式从 8kHz~96kHz 采样频率的音频信号的采集与回放, 并具有较高的信噪比 (SNR) 和较低的能耗^[3]。利用 TLV320AIC23B 的上述特性, 利用 TI McBSP 可编程接口, 辅以有效地编程控制, 能够灵活地实现音频数据的输入、输出控制。本系统充分挖掘 TLV320AIC23B 的特性, 根据 SPI 和 2-wire 操作规范, 通过控制接口对设备寄存器进行编程配置。TLV320AIC23B 的寄存器配置方法为:

```
AIC23_Params config = { \
    0x001c, /* 0 EVM5502_AIC23_LEFTINVOL Left line \
input channel volume */ \
    0x001c, /* 1 EVM5502_AIC23_RIGHTINVOL Right line \
input channel volume */ \
    0x01f9, /* 2 EVM5502_AIC23_LEFTHPVOL Left \
channel headphone volume */ \
    0x01f9, /* 3 EVM5502_AIC23_RIGHTHPVOL Right \
channel headphone volume */ \
    0x0010, /* 4 EVM5502_AIC23_ANAPATH Analog \
audio path control */ \
    0x0000, /* 5 EVM5502_AIC23_DIGPATH Digital \
audio path control */ \
    0x0000, /* 6 EVM5502_AIC23_POWERDOWN Power \
down control */ \
    0x0043, /* 7 EVM5502_AIC23_DIGIF Digital \
audio interface format */ \
    0x000d, /* 8 EVM5502_AIC23_SAMPLERATE Sample \
rate control */ \
    0x0001 /* 9 EVM5502_AIC23_DIGACT Digital \
interface activation */ \
};
```

在系统初始化时, 调用 DSP 的 CSL 命令配置并启动 TLV320AIC23B, 使其进入自动工作状态:

```
AIC23_setParams(&config);
```

2.1.2 基于 McBSP 和 DMA 的数据传输

音频信号被 TLV320AIC23B 采集并经模数转换后需要传送到 DSP 进行处理, 该传输过程有多种实现方式, 如 DMA、I2C、UART、PIPE、GPIO 等。为保证系统良好的实时性, 并尽可能地为 DSP CPU 处理节省时间, 本系统采用了 McBSP - DMA 的传输方式, 利用 McBSP 高效的全双工通信和与 CPU、DMA 方便灵活的中断和事件控制机制, 实现数据流从 TLV320AIC23B 到 DMA 的传输, 并利用 DMA 模块在不干扰 DSP CPU 的情况下独立地完成数据传输。本系统在 DSP/BIOS 机制下利用 McBSP 和 DMA 的可编程控制接口实现了 TLV320AIC23B、McBSP、DMA

和 DSP 之间高效地数据传输。

2.2 小波频带阈值去噪算法

根据实际音频信号的特点, 本系统选用了小波频带阈值的去噪算法^[4], 并针对噪声大大强于语音有效信号、信噪比较低的情况, 设计了前端硬件滤波电路以提高信噪比, 并对软件算法利用迭代法进行了优化, 取得了较好的去噪效果。

优化后的小波去噪算法的程序实现为:

(1) 设定相对阈值 K 、每次处理的数据长度 $L1$ 、参考噪声数据长度 $L2$ 、小波分解的级数 A 和小波滤波器等参数。

(2) 假定含噪数据的前 $L2$ 个数据点为噪音, 根据设定的小波分解级数 A 进行小波变换, 得到 N 个小波分解子频带, 得出噪音的 N 个频带系数绝对值的最大值的估计 $M(i), i=0,1,\dots,N-1$ 。

(3) 根据设定的相对阈值 K , 计算各子频带的修正阈值 $D(i)=K*M(i), i=0,1,\dots,N-1$ 。

(4) 对设定长度的含噪数据作 A 级小波分解, 利用前述得到的子频带修正阈值 $D(i)$, 对第 i 个频带的系数 $x(j)$ 进行软滤波: $x(j)=\text{sgn}(x(j))*\text{Max}\{\text{abs}(x(j))-D(i),0\}$ 。

(5) 对处理后的小波系数进行相应的小波反变换, 得到去噪后的语音信号。

利用该算法对实际强噪声环境下的一段含噪语音进行去噪处理, 采用 10kHz 采样频率, 16b 数据格式, 得到的对比结果如图 2 所示。

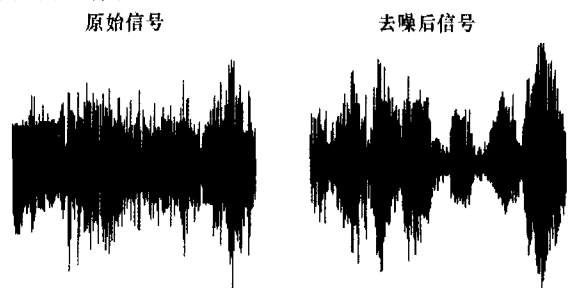


图2 去噪效果对比

2.3 流程控制

为保证系统的实时性, 提高开发效率, 采用了 DSP/BIOS 控制机制, 利用 DSP/BIOS 对象和分析工具、模块化的 API 函数和优化的指令库、多种线程模型和高效的线程间通信技术, 在集成的 CCS 开发环境中实现了系统流程的实时控制, 并高效、准确地控制线程间通信。

系统主要由两大模块构成: 一是系统的主函数, 是程序执行的入口并完成需要的初始化工作; 二是 DSP/BIOS 控制

模块，通过预先定义的各种对象模型和控制逻辑自动运行。系统启动时，首先执行主函数，完成设备的初始化和参数配置工作后，把控制权转交给 DSP/BIOS，预先定义的控制逻辑开始运行，完成承担的具体任务，并协调各种事件和中断。

在本系统中，音频信号被 TLV320AIC23 模块采集，通过 McBSP 和 DMA 以数据帧的格式自动发往 DSP 的片内存储器 DARAM，即当一帧数据采集完后，DMA 向 CPU 发出硬件中断请求，CPU 利用 DSP/BIOS 的 HWI 对象响应该中断，在 HWI 处理程序中首先重新配置 DMA，使数据采集过程继续进行并保证正确性，同时利用 SWI 对象产生软件中断请求，启动 SWI 线程并在该线程中执行数据处理程序。处理完的数据通过 DMA、McBSP 输出到 TLV320AIC23 进行回放。图 3 是系统的流程结构。

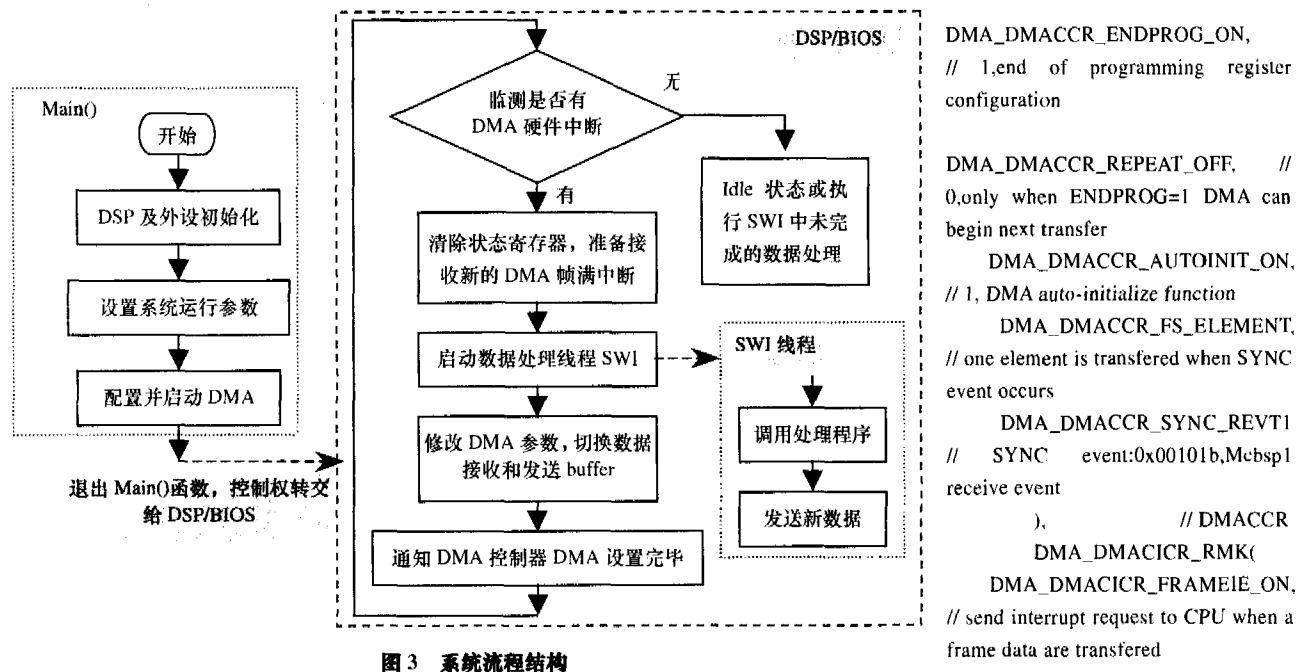


图 3 系统流程结构

由于数据处理模块可能会占用较多的 CPU 时间周期，因此，为了保证 DMA 数据采集和传输模块的正常工作，在 HWI 响应程序中首先清除状态寄存器，然后启动数据处理线程 SWI，此时由于 HWI 没有执行完，较低优先级的 SWI 并不会执行，处于等待状态。等修改完 DMA 参数后通知 DMA 控制器，使得 DMA 模块继续工作，数据采集和传输过程不会被中断。前述工作占用的时间要小于采集完一帧数据的时间，因此不会造成数据帧丢失。等 HWI 执行完后，SWI 开始执行，数据处理程序被调用，同时，DMA 模块正在采集下一帧数据，这种数据采集与处理过程并行的方式提高了系统的实时性和执行效率。

3 关键技术的设计与实现

3.1 DMA 通道的配置方法

本系统采用 DMA 方式采集和传输数据，不需要占用 CPU 资源。利用两个高优先级的 DMA 通道 Channel4 和 Channel5 分别负责数据接收和发送，采用 16b 数据类型，当设定的同步事件发生时采用元素级数据传送，以帧中断事件向 CPU 发出中断请求。采用自动初始化的工作方式，当完成一帧数据传送并发出中断请求后根据控制指令立即开始新数据传输。

接收通道的主要参数配置方法为：

```

static DMA_Config dmaCfgReceive = {
    DMA_DMALCSDP_RMK(
        DMA_DMALCSDP_DST_DARAM,
        // destination port is DARAM
        DMA_DMALCSDP_SRC_PERIPH,
        // source port is peripheral device, McBSP1
        DMA_DMALCSDP_DATATYPE_16BIT
        // 16 bit data type
    ),
    // DMALCSDP
    DMA_DMALCCR_RMK(
        DMA_DMALCCR_DSTAMODE_POSTINC,
        // DARAM 目标地址自动增值
        DMA_DMALCCR_SRCAMODE_CONST,
        // 数据源地址不变
        DMA_DMALCCR_ENDPROG_ON,
        // 1,end of programming register
        // configuration
        DMA_DMALCCR_REPEAT_OFF, //
        // 0,only when ENDPROG=1 DMA can
        // begin next transfer
        DMA_DMALCCR_AUTOINIT_ON,
        // 1, DMA auto-initialize function
        DMA_DMALCCR_FS_ELEMENT,
        // one element is transferred when SYNC
        // event occurs
        DMA_DMALCCR_SYNC_REVT1
        // SYNC event:0x00101b,Mcbsp1
        // receive event
    ),
    // DMALCCR
    DMA_DMALCICR_RMK(
        DMA_DMALCICR_FRAMEIE_ON,
        // send interrupt request to CPU when a
        // frame data are transferred
    ),
    // DMALCICR
    (DMA_AdrPtr)((Uint32)(MCBSP_DRR11_ADDR<<1)), //
    DMALCSSAL, MCBSP port
    0, // DMALCSSAU
    NULL, // DMALCDSAL, to be loaded when
    system running
    0, // DMALCDSAU
    BUFFSIZE, // DMALCEN: Length of data buffer
    1, // DMALCFN
    0, // DMALCFI
    0 // DMALCEI };
    
```

发送通道与此类似，只是在源地址和目标地址属性的配置上不同，与接收通道相应属性互换配置。

3.2 PING-PONG buffer 的处理机制

本系统采用双接收 buffer 的方式来保证实时性和数据完整性。DMA 控制器从 McBSP 获取数据并存储在 DARAM 的 buffer 中，设置两个接收 buffer，分别称为 PING 和 PONG。第 1 帧数据首先被顺次存储在 PING 中，当 PING 采集满后，通过修改 DMA 源地址参数把存储区切换成 PONG，新采集的数据被存储在 PONG 中，同时，CPU 执行对 PING 中数据的处理。当 PONG 采集满后，再切换到 PING 接收，而 PONG 中的数据被处理。这种数据传输机制保证了在处理上一帧数

(下转第 274 页)

相关的簇是跟踪和诊断系统，它通过数字无线接口与可计算的簇通信。

根据 TTA 的容错策略，由于接收簇要循环访问高速运转的车轮系统，并实时解码，因此有必要在 RC-operator 和可计算的簇之间添加两个容错的 CNI，这样建立了两个节点级冗余；汽车硬件簇四轮系统提供了特殊的应用级容错，因为任意某个车轮的失效信息会从其余 3 个车轮的安全状态得到补偿，所以车轮系统实现了无冗余的容错。

当今多数硬件的体系结构只提供单一的 CPU 处理中间件和应用程序。TTA 的 FTU CNI 的实现引入了一个专用的中间件处理器。通过消除中间件和主机之间不必要的通信，显著提高了终端的可预测性。中间件的处理器负责处理中间件内部事件和普通的错误，而主机处理器负责执行主机上的应用程序。这种硬件体系结构很大程度上解决了分布式实时系统的软件重用问题。

4 结束语

TTA 架构已经发展了 20 多年，趋于完善。期间提出的相

(上接第 249 页)

据的同时，还能接收新数据，既不影响数据采集，又防止原数据被刷新。CPU 可以充分利用采集满下一帧数据的时间处理数据，容易满足实时性要求。

4 结论

本文利用 DSP 微处理器和小波分析技术，实现了强噪声环境下语音的实时噪声消除。基于 EVM 开发板、仿真器、CCS 和 PC 机建立了具有较好通用性的程序开发框架，针对 DMA 传输方式、数据采集与处理等具体问题提出了相应的解决方案，对类似研究具有较好的参考价值。

参考文献

1 Texas Instruments. TMS320VC5502 Fixed-point Digital Signal

关概念、实现和评价都是围绕“可靠的全局时基效用性”这一个核心进行的。总的看来，TTA 系统有如下特点：

- (1)可预测性：由于采用确定的时间触发调度机制，因此计算时延是可预测的；
- (2)可测试性：自动按时间触发协议和调度在恰当的时刻对计算结果进行自检；
- (3)组合能力：可由独立设计和测试的组件或子系统来构成系统；
- (4)复制决定性：复制组件的行为在组件之间有一致性；
- (5)成员制：依靠通信网络上的节点，某一时间故障状态将自动广播。

TTA 跨越了分布式实时系统发展的整个阶段。目前，TTA 已经在嵌入式分布式系统中占有一席之地，而且高可靠性的实时性越来越成为关注的焦点，TTA 的发展前景令人鼓舞。

参考文献

1 Ademaj A. Achieving Fail Silence in the Time-triggered Architecture[DB/OL]. <http://www.vmars.tuwien.ac.at>, 2000.
2 Kopetz H. The Time-triggered Architecture[DB/OL]. <http://www.vmars.tuwien.ac.at>, 2001.

Processor Data Manual[Z]. 2004-06.

2 闻亭科技发展有限公司. TDS5502 EVM 用户使用手册 V1.0[Z]. 2004-03.
3 Texas Instruments. TLV320AIC23B Stereo Audio CODEC Data Manual[Z]. 2003-03.
4 Yang Dali, Xu Mingxing, Wu Wenhui. A Noise Cancellation Method Based on Wavelet Transform[C]. Beijing: The Second International Symposium on Chinese Spoken Language Processing (ISCSLP), 2000-10.
5 沈戈, 高德远, 樊晓桢. 数字信号处理器 (DSP) 结构设计及发展趋势[J]. 计算机工程与应用, 2003, 39(7): 4-6.
6 Texas Instruments. TMS320VC5502 EVM Help[Z]. 2004-07.

(上接第 251 页)

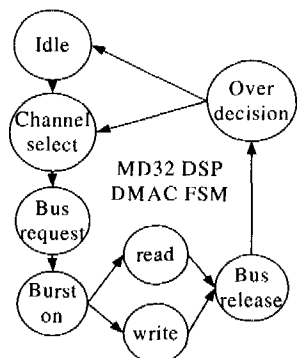


图 4 MD16DSP 的 DMA 控制器状态机

MD16 的 DMA 控制器主状态机有 4 个状态，MD32 的 DMA 控制器状态机有 8 个状态。这样的设计保证了 FSM 在编码时无冗余状态。

4 DMA 控制器的设计结果验证对比

MD16 和 MD32 的 DMA 控制器设计都是采用 Verilog 硬件描述语言进行 RTL (寄存器传输级) 的代码描述^[3]，对其进行功能仿真。使用 Design Compiler 结合标准单元库对设计模块进行综合、优化，分析得到的面积、速度报告(见表 1)。由于 RTL 结构描述较好，在面积、功耗以及速度方面，得到了预期的结果。综合的结果也通过了门级仿真和功能验证。

表 1 MD16 和 MD32 的 DMA 控制器综合结果

DMA 控制器	MD16	MD32
面积 (门)	2 777	3 230
动态功耗 (mW)	23.075 1	23.987 6
静态功耗 (mW)	137.390 6	141.643 1
时延 (ns)	5.61	5.83

同时，我们还通过 FPGA 系统仿真板实现了 DMA 控制器和 DSP 核的成功互连，功能满足了设计要求。

5 结束语

本文通过 MD16 和 MD32 的 DMA 控制器设计研究，从系统的角度，综合考虑软件和硬件两个层面的因素，给出了一种简洁、有效的 DMA 控制器设计方法。

参考文献

1 Agarwala S, Fuoco C, Anderson T. A Multi-level Memory System Architecture for High Performance DSP Applications[C]. Proceedings of the 2000 International Conference on Computer Design, 2000: 17-20.
2 Qiao Lufeng, Wang Zhigong. Design of DMA Controller for Multichannel PCI Bus frame Engine and Data Link Manager[C]. IEEE 2002 International Conference on Communications, Circuits and Systems and West Sino Expositions, 2002-06: 1481-1485.
3 Bhasker J. 徐振林译. Verilog HDL 硬件描述语言[M]. 北京: 机械工业出版社, 2000.