

基于 AHB 总线协议的 DMA 控制器设计

赵 强^{1,2}, 陈 岚¹

(1 中国科学院 微电子所, 北京 100029; 2 西安电子科技大学, 陕西 西安 710071)

摘 要: 提出了基于 AMBA AHB 总线协议的 DMA 控制器设计方法, 阐述了 AHB 总线的特征架构以及 DMA 数据传送方式, 分析了 DMA 在数据传送应用中的特征作用, 提出了一种功能较为完备的 DMA 控制器设计方案, 给出了较为详细的模块设计. 最后, 描述了验证环境的搭建和验证过程, 并通过软件验证, 完成了 DMAC 所要实现的功能.

关键词: AHB; DMA; DMAC

中图分类号: TN402

文献标识码: A

文章编号: 1000-7180(2014)02-0129-03

DMA Controller Design Based on AHB Bus Protocol

ZHAO Qiang^{1,2}, CHEN Lan¹

(1 Institute of Microelectronics of Chinese Academy Sciences, Beijing 100029, China;

2 Xi'an Electronic and Science University, Xi'an 710071, China)

Abstract: In this paper, we propose a method about the DMA controller design which based on AMBA AHB bus protocol, and describes the characteristics of the AHB bus architecture and DMA data transfer mode, analysis the application of the DMA about data transfer, we propose a more complete functional design of the DMA controller, give a more detailed module designs. Finally describes a method to set up the verification environment and verification process, and through software's validation, correct completion of DMAC's functions to be achieved.

Key words: AHB; DMA; DMAC

1 引言

AHB 是为提出高性能可综合设计要求而产生的新一代 AMBA 总线^[1-3]. AMBA AHB 是位于 APB 之上的新一级总线, 并且实现了高性能以及高系统时钟频率, 具有突发传输、分块处理、单周期总线主机移交、单时钟沿操作、非三态执行以及更宽的数据总线架构(64 位或者 128 位)特征.

直接存取器存取(DMA)^[4-5]是计算机应用系统中用于提高数据传输速率和微处理器使用效率的一项重要技术. 外设与微机内存之间的信息交换, 可以通过程序查询方式和中断方式进行. 这两种方式都是在 CPU 控制下, 通过 CPU 执行指令来完成的. 数据传送方向为外设 $\longleftrightarrow$ CPU $\longleftrightarrow$ 内存. 这两种方式每传送 1 个字符都需要耗用较长时间. 对于一些高速外

设, 如果传送速度太慢, 不仅降低传送效率, 还会造成数据丢失, 导致传送出错. 基于这个原因, 提出了 DMA(direct memory access)方式. 这种方式是不经过 CPU 干预, 直接在外设与存储器之间进行数据传送的方式. 实现 DMA 传送, 需要有一个专用硬件 DMA 控制器(DMAC), 在 DMA 传送期间, CPU 要让出对 AHB 总线的控制权, 交给 DMA 控制器控制. 由于数据直接在存储器和外设之间传送, 不经过 CPU 干预, 所以传送速度大大提高, 可接近于存储器的最快存储速度. 本文基于 AHB 总线协议, 提出一种 DMA 控制器设计方案, 并给出关键模块设计.

2 总体设计简介

2.1 基于 AMBA AHB 的典型微控制器

基于 AMBA 的典型微控制器由一个高性能系

收稿日期: 2013-04-17; 修回日期: 2013-06-23

基金项目: SiGe-BiCMOS 产品平台开发(2009ZX02303-004)

统中枢总线组成,能够支持外部存取器带宽,包括 CPU、片上存取器和其他直接数据存取(DMA)设备,加上一个桥接器连接到较窄的挂接着较低带宽外设的 APB 总线.图 1 表示了包含 AHB 和 APB 的一个典型系统.

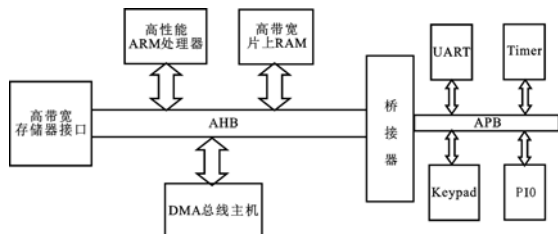


图 1 基于 AMBA AHB 的典型系统

在 AMBA 系统架构中,有 AHB 总线和 APB 总线. AHB 总线上通常会挂一些高速设备,DMAC 属于高速设备挂在 AHB 总线上.而 APB 总线上通常会挂一些低速设备,图 1 就是在一个常见的 SoC 系统下,DMAC 所在的位置.系统中可以使用多通道来提高数据传输效率,但是在大多数应用中,DMAC 只需有一个 master 接口和一个通道就能满足数据传输的要求.而 slave 接口则是连接 CPU 的信号或者编程用.

2.2 DMA 控制器总体设计

基于 AHB 总线的 DMA 控制器在 CPU 启动数据传送之后,作为一个主机接管了 AHB 总线,完成数据传送.作为代替 CPU 工作的一个 AHB 总线主机,首先是由 CPU 通过对寄存器进行配置,根据配置的信息来完成要求的数据传送.配置的寄存器包括:①控制命令寄存器.寄存传送数据的发起方式、传送数据的大小以及通道和地址的控制命令;②初始地址寄存器.通过 CPU 对这些寄存器的配置,DMAC 根据寄存器的值可以正确的寻址进行数据的传送.

本设计中 DMAC 采用单通道数据传送方式,支持双向传输;对于目的或源地址,采用递增的地址产生方式;DMA burst 尺寸可通过编程改变,以更有效地传输数据;支持 8 bit、16 bit 和 32 bit 宽度的传输;AHB master/slave 的编程接口,拥有 32 bit AHB master 总线宽度;DMA 传输完成中断请求.

3 DMA 控制器的实现

首先,完成的工作就是根据功能需求将 DMAC 进行子功能模块划分,如图 2 所示.然后,用 Verilog 实现各个子模块.最后,进行各个模块以及总体功能仿真.从图 2 可以看出 DMAC 分成四个模块,分别

为 master 接口模块、slave 接口模块、数据传送控制模块和地址产生控制模块.

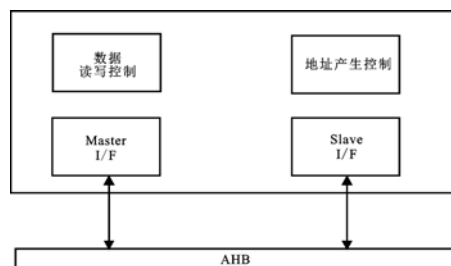


图 2 DMAC 模块划分

本设计中的 IP 支持四种方式的数据传送,分别是 Memory to Memory、Memory to Peripheral、Peripheral to Memory、Peripheral to Peripheral.这四种方式的传送都采用 DMAC 作为主控制器. DMAC 发起的数据传送请求有两种:当 memory 进行数据传送时则不需要发起请求,直接开始进行传送.当 peripheral 需要进行数据传送时,则要等待外设自身发起的请求信号.对于以上两种请求信号,是由译码器根据 CPU 给的信息进行判断.

在进行数据传送时,据需要 CPU 通过 slave 接口对 DMAC 进行配置,将信息给到 DMAC 中,这些信息首先会被存到通道寄存器中,之后这些信息被解读,根据解读的信息 DMAC 进行后续的工作.具体的工作过程为:CPU 会将外部设备的属性(memory or peripheral)、数据传送的长度、数据传送数据的源和终端设别的起始地址,通过 slave 接口配置到通道寄存器中.通道寄存器将信息以信号的方式引到 DMAC 中的不同模块中.译码器通过对寄存器中信息的解读,判读外部设备的属性,将判断信号给到总线请求模块中,总线根据来自译码器以及由外设通过硬件握手给出的信号进行判断,最终发出数据传送请求.请求信号通过 master 接口给出,由 cpu 来判断是否给予 DMAC 总线使用权,当条件满足,DMAC 获得了总线使用权后,开始进行数据传送.数据传送过程是由两个部分组成,首先是源和终端地址状态机根据寄存器中给出的传送起始地址以及自身状态机的跳转,将地址信号给到地址模块,地址模块判断来自两个状态机的地址信号,根据 FIFO 中给出的控制信号进行有效性判断,将其中的一个地址作为当下有效地址,并通过 master 接口给到 AHB 总线,总线通过地址进行寻址.根据来自 DMAC 内部的判断信号,总线寻址之后进行数据读写,将总线数据搬到此地址下,或者是将此地址下的数据转移到总线上. FIFO 顶层模块中包含数据控制模块和读写使能判断模块,数据控制模块

中的状态机进行数据的存取. 本设计的 FIFO 深度为 16, 宽度为 32 bits. 将来自主机接口的数据存在 FIFO 的缓存中, 当 FIFO 满时, 读写使能模块将改变使能信号, 根据使能信号的跳转, 数据控制模块将 FIFO 缓存中的数据通过 master 接口给到总线, 通过总线把数据存到终端设备中. 当所有要求的数据传送完毕, DMAC 会给出一个中断信号, 这个中断信号将通过 master 接口以及总线告知 CPU, 此时 CPU 将会给出一个中断清除信号, 同时将会收回 DMAC 的总线使用权, 或者是重新对 DMAC 进行下一次配置.

本文设计的 DMAC IP 对于大量连续地址下的数据传送有很高的效率, 每次进行 16 个 32 bits 的数据传送, 提高了数据传送的速率. 由于 AHB 总线每次在进行读写数据之前, 总线必须是由空闲状态跳转到连续或者非连续状态, 以及由于外设对于总线的读写响应速度的影响, 总线只有在收到外部设备的响应信号才可以进行下一个数据的读写, 这样在数据开始传送的开始阶段会占用比较多的周期. 本设计采用 16 个数据的 burst 传送, 有效地提高了传送效率.

4 实验验证

此 IP 验证是将划分的各个子模块进行验证, 为每个子模块编写 testbench, 进行功能验证, 在功能正确的基础上, 着重注意时序的正确性. 所有子模块验证结束之后, 将模块集成到现有的验证. 平台上进行验证. 本设计中验证平台包含了完整的 AHB IP 核和一个挂在 SOC 系统上的简单 slave 行为模型 (图 3). 利用脚本中给出的 ahb_write32 以及 ahb_read32 等函数进行 testcase 编写. 在 testcase 中先进行传送数据前的信息配置, 在 SOC 系统上, 根据系统分给 DMAC 的地址区间, 将给出的基地址加上 DMAC IP 内部的寄存器的偏移地址进行数据信息的配置, 利用 ahb_write32 函数将源/终端起始地址写到对应的地址寄存器当中, 将数据长度信息先配置到寄存器当中, 待所有的信息配置好之后, 最后进行使能位的配置, 保证数据的准确配置. 最后, 利用 ahb_read32_expect 函数对配置信息进行验证. 若是配置的信息不正确, 将会有报错现象. 验证需要用到的工具可以利用 Modelsim 或者是 VCS 软件进行验证. 本次设计中采用的是 VCS 软件验证.

在设计中, IP 的验证是测试了连续地址下的 26 个数据传送, 源起始地址给定的 32'h00801400, 终端起始地址为 32'h00801500. 测试数据来自 SOC

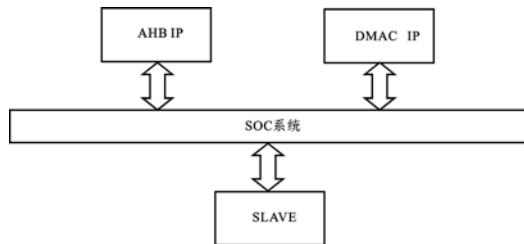


图3 验证平台

系统划分的一个地址区间, 在这个地址区间上挂了一个简单的 SLAVE 接口模块, 这个简单的行为模型是由一些寄存器构成的, 经过验证, 数据的传送 (读写) 时序正常. 读写时序如图 4、图 5 所示.

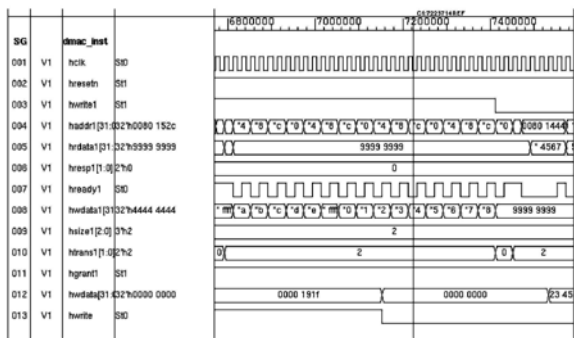


图4 DMAC 读取数据时序图

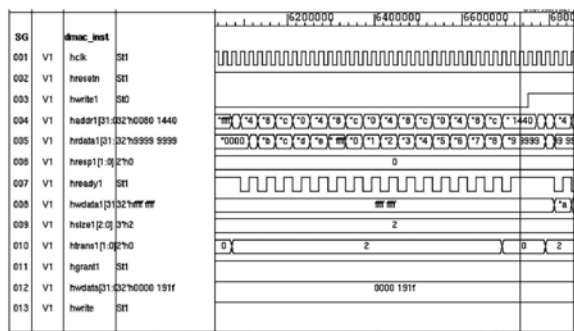


图5 DMAC 写数据时序图

数据的读写时序正常, 符合总线协议的时序要求, 数据传送正确. 在传送完 26 个数据之后, htrans1 信号变为 2'b00, 此时 AHB 总线处于空闲状态, 不会有数据的传送. INT 信号作为中断响应信号, 等待 CPU 给予的清“0”信号进行清除中断, 完整的传送完成.

5 结束语

本文对基于 AHB 总线的 DMAC 进行了介绍, 提出了较为详细的 DMA 控制器实现方式, 从功能上对 DMAC 进行了剖析. 阐述了 DMAC 的模块构成, 并对完成的 DMAC IP 进行功能和时序的验证, 得到比较理想的结果. (下转第 136 页)

- [2] Wolfgang R B, Delp E J. A watermark for digital images[C]// Image Processing, 1996. Onlando: IEEE, 1996: 219-222.
- [3] 谢建全, 阳春华. 一种大容量的自适应信息隐藏方法[J]. 计算机应用, 2007, 27(5): 1035-1037.
- [4] 赵玉霞, 康宝生. 基于混沌序列的抗剪切变换攻击数字图像隐藏新算法[J]. 计算机应用与软件, 2010, 27(3): 100-103.
- [5] 赵培东, 谢剑英. 基于坐标系复原的抗任意剪切盲水印算法[J]. 计算机工程, 2008, 34(6): 134-136.
- [6] 周翔翔, 尹忠海, 王欣, 等. 基于结构化编码的抗剪切图像数字水印[J]. 计算机应用, 2009, 29(5): 1324-1326.
- [7] 彭波, 王平. 抗剪切鲁棒图像数字水印的研究与实现[J]. 中国农业大学学报, 2006, 11(4): 105-108.
- [8] 王丽, 赵媛媛, 赵耀. 一种抗剪切的鲁棒数字水印[J]. 数据采集与处理, 2006, 21(3): 330-333.
- [9] 朱莉莉, 侯迎坤, 陈涛. 抗大面积剪切的小波变换域图像水印算法[J]. 泰山学院学报, 2010, 32(5): 34-39.
- [10] 赵玉霞. 基于混沌系统与提升小波的抗剪切攻击的彩色图像盲水印算法[J]. 工程图学学报, 2010(4): 81-85.
- [11] Kutter M, Bhattacharjee S K, Ebrhimi T. Towards second generation watermarking schemes[C]// ICIP'99. Kobe, Japan: IEEE, 1999: 320-323.
- [12] 韩亚丹, 闰德勤. 一种基于特征的抗剪裁盲水印算法[J]. 中国图象图形学报, 2007, 12(4): 574-580.
- [13] 同鸣, 闫涛, 姬红兵. 一种抵抗强剪切攻击的鲁棒性数字水印算法[J]. 西安电子科技大学学报, 2009(1): 22-27.
- [14] Areef T E, Heniedy H S, Ouda Mansour O M. Optimal transform domain watermark embedding via genetic algorithms[C]// Information and Communications Technology, 2005. Cariro: IEEE, 2005: 607-617.

作者简介:

雷 芸 (1979-), 讲师, 研究方向为教育技术、图像处理.

(上接第 131 页)

参考文献:

- [1] 梁科, 李国峰, 王锦, 等. 通用多通道高性能 DMA 控制器设计[J]. 天津大学学报, 2008, 41(5): 621-626.
- [2] 颜建峰, 吴宁. 基于 PCI 总线的 DMA 高速数据传输系统[J]. 电子科技大学学报, 2007, 36(5): 858-861.
- [3] 史昕蕾, 杨军, 陆生礼. 嵌入式 SOC 中的 DMA 控制器的设计与优化[J]. 电子工程师, 2004(1): 5-7.
- [4] 谭震, 钱晓东, 马瑜兰. 12 位 A/D 转换器与 PC 机 DMA 接口的简化设计[J]. 测控技术, 1997(6): 54-56.

- [5] 王首义, 徐艳. 在 Windows NT 下开发高速 DMA 数据采集程序[J]. 信息技术, 2001(1): 59-61.

作者简介:

赵 强 男, (1988-), 硕士研究生. 研究方向为软件工程设计集成电路设计方向.

陈 岚 女, (1968-), 研究员, 博士生导师. 研究方向为移动物联网系统架构及核心芯片、纳米芯片可制造性设计方法(DFM)、SoC/IP 核验证、评测、低功耗设计技术.