

PCI Express 总线接口的 DMA 传输设计^①

王红志^{1,2}, 王超^{2,3}, 栾勇^{2,3}, 姚恺丰⁴

¹(中国科学院大学, 北京 100049)

²(中国科学院沈阳计算技术研究所 高档数控国家工程研究中心, 沈阳 110168)

³(沈阳高精数控智能技术股份有限公司, 沈阳 110168)

⁴(国家电网公司东北分部电力调控分中心, 沈阳 110180)

摘要: 在认真理解 PCI Express 协议及其应用技术的此基础上, 使用 FPGA 实现 PCI Express 总线接口的 DMA 传输设计. 本文主要从 DMA 发送模块、DMA 接收模块和 DMA 状态控制模块三个部分对设计进行详细的介绍. 最后, 在基于 Linux 系统的 PC 机中实现 PCI-E 的驱动程序, 使用 BEEcube 公司的 miniBEE 作为测试平台对设计进行检测并给出结果.

关键词: FPGA; PCI Express; DMA; Linux; 驱动程序

引用格式: 王红志, 王超, 栾勇, 姚恺丰. PCI Express 总线接口的 DMA 传输设计. 计算机系统应用, 2017, 26(10): 285-288. <http://www.c-s-a.org.cn/1003-3254/6007.html>

DMA Transmission Design of Bus Interface Based on PCI Express

WANG Hong-Zhi^{1,2}, WANG Chao^{2,3}, LUAN Yong^{2,3}, YAO Kai-Feng⁴

¹(University of Chinese Academy of Sciences, Beijing 100049, China)

²(National Engineering Research Center for High-End CNC, Shenyang Institute of Computing Technology, Chinese Academy of Sciences, Shenyang 110168, China)

³(Shenyang Golding NC Intelligent Tech. Co. Ltd., Shenyang 110168, China)

⁴(State Grid Corporation of Northeast Branch of the Power Control Sub-Centers, Shenyang 110180, China)

Abstract: The DMA transfer design of PCI Express bus interface is realized by FPGA based on the in-depth understanding of PCI Express protocol and its application technology. This paper mainly introduces three parts of the design: DMA sending module, DMA receiving module and DMA state control module in detail. Finally, it realizes PCI-E driver in one PC based on Linux system, then uses miniBEE as a test platform of BEEcube Company to test the design and give the results.

Key words: FPGA; PCI Express; DMA; Linux; driver

1 引言

系统总线也被叫做计算机总线、内部总线, 在系统中用来连接的 CPU、显卡、硬盘等功能部件. 随着计算机技术的快速发展, CPU 的处理速度越来越快, 最近二十年占据着主导地位的 PCI 及其拓展的 PCI-X 系列总线已满足不了实际的应用需求, 系统总线已经成为制约计算机系统性能继续提高的瓶颈.

在这个背景下, PCI Express(以下简称 PCI-E) 总线规范一提出, 迅速就得到了广泛的青睐. 与原有的总线相比, PCI-E 采用点对点串行连接的设备连接方式, 每个设备都拥有独立的数据链路, 可以达到远超出 PCI 总线的传输速率; 以数据包的方式进行数据传输, 保证了端对端的可靠性传输, 同时还支持电源管理和热插拔以及热交换等功能; 较好的灵活性, 可根据实际的带

① 基金项目: 高档数控机床与基础制造装备国家科技重大专项 (2014ZX04001041)

收稿时间: 2017-01-17; 采用时间: 2017-02-20

宽需求,给PCI-E的接口物理连接配置不同的链路模式,包括x1、x4、x8、x16以及x32链路模式;具有较好的向后兼容性,PCI-E继承PCI的配置空间,基于PCI总线的系统软件几乎可以不经修改,直接在PCI-E体系结构中使用。

基于以上,特别是高传输速率的优势,在高速的DMA控制器传输方式中,相比于原有总线尤为明显,因此PCI-E总线协议具有巨大的应用前景.目前PCI-E总线的最新规范为V3.0版本,本文基于较为成熟的V2.0版本,提出了应用于总线接口中的高速DMA传输的设计方案。

2 设计方案

本设计DMA模块介于DDR模块和上位机之间,在该模块的控制下,实现FPGA平台上数据以DMA的方式在DDR模块和PC上位机之间的传输.图1是基于FPGA的PCI-E接口DMA设计的模块图,其中DMA模块主要有DMA状态控制模块、发送引擎模块、接收引擎模块、DMA发送和接收模块几个部分.本设计选择Xilinx Virtex-6提供的Integrated Block for PCI-E IP核,该PCI-E IP核实现了PCI-E物理层、数据链路层和传输层间协议信号的转变,因此可以在它的基础上直接进行DMA控制设计,可以大大的降低设计的复杂度。

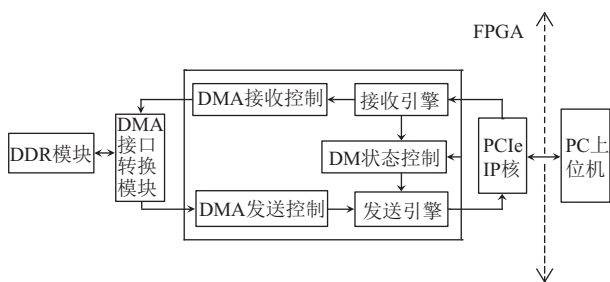


图1 PCI-E DMA接口DMA设计的模块

2.1 数据包 TLP 简介

PCI-E的层次结构与网络层次结构类似,有设备核心层、处理层、数据链路层和物理层.在各层间以数据包的形式传送信息,其中链路处理层数据包(TLP)是最主要的数据类型,其结构如图2所示.首先在核心层生产数据包,再依次通过事务层、数据链路层和物理层,最终发向另外一个PCI-E层次结构.数据包中的头和数据部分由FPGA的代码产生,其他部分

都是IP核自动生成的. PC上位机将地址等信息放在TLP头后,然后将该数据包发给FPGA, FPGA解析数据包,并将解析所得的结果保存在自己的空间中。

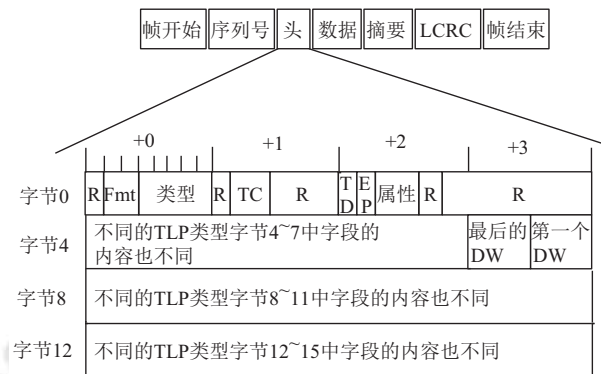


图2 TLP结构及其头格式

2.2 DMA传输设计

从图1中可知DMA传输模块主要分成发送模块、接收模块和DMA状态控制三个子模块,以下依次从这三个模块进行详细介绍。

2.2.1 DMA发送模块设计

发送模块分成发送引擎和DMA发送控制两个子模块. DMA发送控制子模块把要发送的数据转换成TLP的格式,然后将其传递给发送引擎模块;发送引擎子模块负责把待发送的数据生成TLP数据包,然后将其发送给PCI-E IP核.发送引擎子模块需要在状态机的控制下生成存储器读写TLP等请求.图3是发送引擎模块的状态机图。

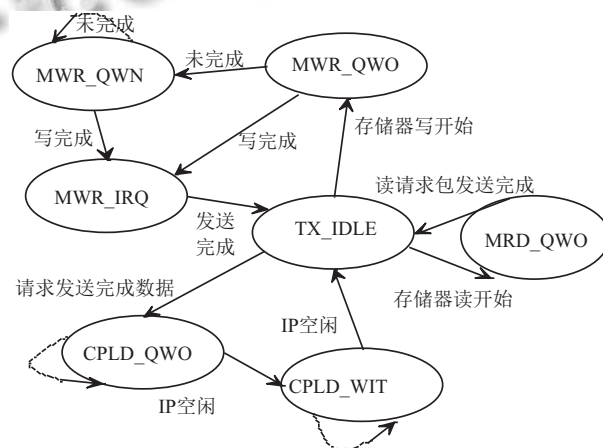


图3 发送引擎的状态机

引擎状态机默认处于TX_IDLE状态.当收到“存储器写开始”信号,状态机由空闲状态进入MWR_QWO

状态,将 32 地址和 64 位数据写入 PCI-E IP 核内.之后判断写操作是否完成,若未完成,进入 MWR_QWN 继续写入数据,之后每写入一次判断一次直到数据写入完成.写操作完成之后,便由 MWR_QWN 状态进入 MWR_IRQ 中断状态进行中断处理,最后恢复到 TX_IDLE 空闲状态,完成一次存储器写事务;当收到“存储器读开始”信号,状态机由空闲状态进入 MWR_QWO 状态,然后将 32 位的读地址写入 PCI-E IP 核内.存储器读属于非转发事务,所以 DMA 模块通过 PCI-E IP 向 PC 上位机发送读请求包,每发送一个读请求包,PC 上位机就向 DMA 模块发送一个读完成包,接收引擎负责接收数据;当收到“请求发送完成数据”信号,状态机由空闲状态进入 CPLD_QWO 状态,在这个状态下发送带数据的 32 位完成包,数据发送完之后进入 CPLD_WIT 状态,然后判断 PCI-E IP 是否发送完毕,若发送完就恢复 TX_IDLE 空闲状态,否则继续等待.

2.2.2 DMA 接收模块设计

接收模块分成接收引擎和 DMA 接收控制两个子模块.接收引擎子模块从 PCI-E IP 接收数据,然后通过模块内部的状态机完成寄存器读写和存储器读等操作事务;DMA 接收控制子模块负责把 TLP 数据包转换为 DDR 的数据位宽,以及向发送引擎模块发送存储器读请求包.图 4 是接收引擎模块的状态机图.

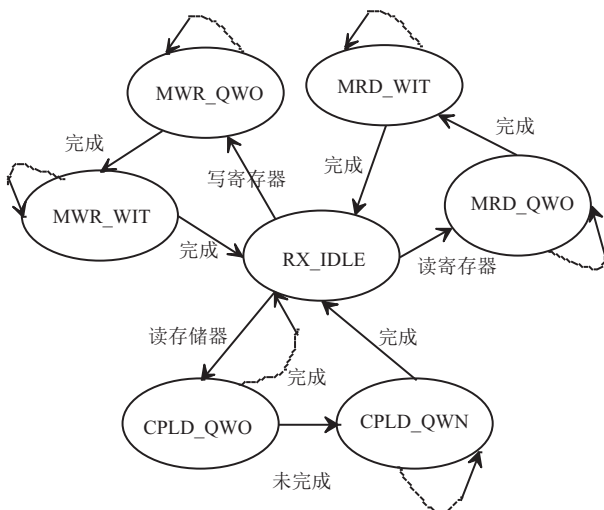


图 4 接收引擎状态机图

接收状态机默认处于 RX_IDLE 状态.存储器读完成事务:发送引擎向 PC 上位机发送存储器读请求,随后 PC 上位机会向接收引擎发送存储器读完成包,接收

引擎分析收到的数据包得到事务的类型等有用信息后,开始进行“存储器读”操作.状态机由 RX_IDLE 空闲状态转到 CPLD_QWO 状态,然后将接收到的 TLP 拆包得到有效数据发送给后端模块,若操作还未完成,进入 CPLD_QWN 状态继续接收数据,否则恢复到 RX_IDLE 空闲状态;寄存器读事务:接收引擎分析收到的 TLP 数据包,得到事务的类型等信息后,发起“寄存器读”操作.状态机由 RX_IDLE 空闲状态转到 MRD_QWO 状态,然后将解析 TLP 得到的读地址触发“请求读寄存器完成”信号后进入 MRD_WIT 等待状态,同时将 DMA 控制模块中读到的寄存器数值通过发送引擎传递给 PC 上位机,最后恢复到 RX_IDLE 空闲状态;寄存器写事务:接收引擎分析收到的 TLP 数据包,得到事务的类型等信息后,发起“寄存器写”操作.状态机由 RX_IDLE 空闲状态转到 MWR_QWO 状态,然后在该状态下解析 TLP 中 32 位的寄存器写地址和写数据,并将这些写入 DMA 控制模块中相应的寄存器中,若写入成功,恢复到 RX_IDLE 空闲状态,否则继续等待.

2.3 DMA 状态控制模块

DMA 状态控制模块解析从 PC 上位机接收到的命令,并通知接收引擎、发送引擎等相关模块完成对应的操作.PC 上位机中的驱动读写一组寄存器来控制 and 配置 DMA 状态控制模块,这些寄存器被映射到 PC 机的 PCI 地址空间中,同时通过硬连接和 FPGA 内的其他模块相连.

本设计 PCI-E 端点设备使用 BAR0 空间,设置其大小为 1024Byte,用以存储与 DMA 控制相关的存储器配置和 PCI-E IP 配置空间状态等信号,包括 TLP 数量、TLP 大小、DMA 地址、DMA 读写事务启动信号等.以 DMA 的方式通过 PCI-E 总线接口将 DDR 中的数据传给 PC 上位机,执行读写操作,都必须先完成 DMA 寄存器的配置,图 5 是 DMA 寄存器配置流程图.

3 驱动编写与测试

本设计在 Linux 系统中实现 PCI-E 的驱动程序.将 PCI-E 设备看作字符设备进行处理,系统首先把字符设备映射到文件系统的文件和目录中,调用读写等函数操作相应的设备.在 fs.h 头文件里声明结构体 file_operations,该结构体提供开关读写和 ioctl 等一系列操纵函数.载入驱动程序后,系统利用上述声明的结构体中所定义的函数操纵相应的设备.操纵过程分成

两步, 驱动程序首先将内核与设备挂接起来, 然后按照设备文件的类型, 调用相应的函数操纵设备。

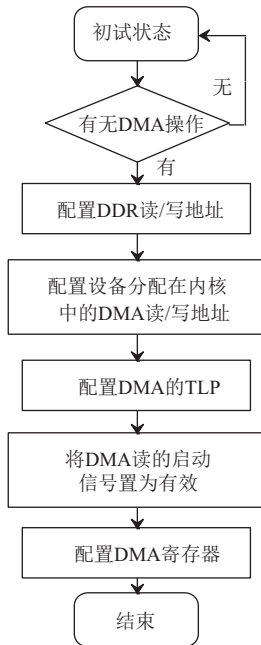


图5 DMA寄存器配置图

测试平台使用的是 BEEcube 公司的 miniBEE, 以 miniBEE 中的 Virtex6 FPGA 逻辑芯片实现数据的存储和控制, 除了 FPGA 处理板, 平台还包括 Linux 上位机和 PCI-E 总线. 再使用检测 FPGA 内部信号的 ChipScope 软件, 结合在 Linux 系统中实现的 PCI-E 设备驱动程序, 对本文设计的高速 DMA 传输进行了测试, 测试结果见图 6, PCI-E DMA 模块 DMA 读操作事务总线带宽可以达到 965MBps, DMA 写操作事务总线带宽可以达到 882MBps, 大大提升了数据传输的速度. 再将在 FPGA 里产生线性数并上传到 PC 机, 通过 matlab 查看这些线性数, 见图 7 可以看出数据是线性的, 说明数据在 DMA 中传输比较可靠。

```

[root@mBEE4 util]# ./data_gen 10240 file_10MB
10485760 bytes data generate success...
[root@mBEE4 util]# ./host_to_device file_10MB
*****Performance is:882.846133 MBps*****
[root@mBEE4 util]# ./device_to_host 10485760 file_10MB
*****Performance is:965.771201 MBps*****
  
```

图6 DMA读写速度实验结果

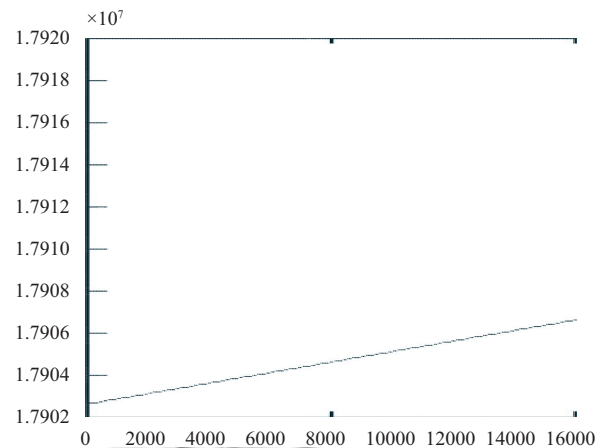


图7 线性数 matlab 图

4 结语

本设计通过 FPGA 内置的 PCI-E IP 核实现 DMA 传输的设计方案, 在实际的应用中可以灵活配置. 与其它 PCI-E 的 DMA 传输设计相比, 本设计主要针对 PCI-E 在数控控制系统中的应用, 并能更好的发挥 PCI-E 高传输速率的优势. 经测试, DMA 读写操作的速度可以很好的满足数控实际中的应用需求。

参考文献

- 1 王齐. PCI Express 体系结构导读. 北京: 机械工业出版社, 2010.
- 2 Corbet J, Rubini A, Kroah-hartman G. LINUX 设备驱动程序. 魏永明, 耿岳, 钟书毅, 译. 北京: 中国电力出版社, 2006.
- 3 李晓宁, 姚远程, 秦明伟. 基于 PCIe 的高速接口设计. 微型机与应用, 2016, 35(1): 27-29, 32.
- 4 信侃. 基于 Xilinx FPGA 的 PCIe 总线接口设计与实现. 无线电通信技术, 2014, 40(4): 94-96.
- 5 李硕, 刘芳, 肖依. 基于 FPGA 的 PCIe SSD 设计与实现. 中国科技论文, 2014, 9(4): 403-406.
- 6 汪精华, 胡善清, 龙腾. 基于 FPGA 实现的 PCIE 协议的 DMA 读写模块. 微计算机信息, 2010, 26(29): 7-9. [doi: 10.3969/j.issn.2095-6835.2010.29.003]
- 7 李木国, 黄影, 刘于之. 基于 FPGA 的 PCIe 总线接口的 DMA 传输设计. 计算机测量与控制, 2013, 21(1): 233-235, 249.
- 8 李丽斯, 崔志华, 殷晔, 等. 基于 FPGA 的 PCIe 总线 DMA 控制器的设计与验证. 计算机测量与控制, 2014, 22(4): 1166-1168.
- 9 邹晨, 高云. 基于 FPGA 的 PCIe 总线 DMA 传输的设计与实现. 电光与控制, 2015, 22(7): 84-88.