

基于 PCIE 的 SG DMA 高速数据传输系统

孙晓晔

(石家庄开发区冀科双实科技有限公司,河北 石家庄 050081)

摘要:结合高速数据加密卡这一具体应用,介绍了 PCI Express 总线协议、基于 PCI Express 的高速数据传输系统构成和以 FPGA 为核心的硬件电路,并对 Windows XP 系统下借助 WinDriver 生成 PCIE 设备驱动程序、采用 SG DMA 方式实现数据传输予以阐述,程序在 Arria II GX 开发板上调试成功。该系统采用 FPGA 的 PCIE 硬核实现 PCI Express 总线的接口逻辑,摒弃了专用总线接口芯片,大大提高了设计的灵活性和可扩展性;采用 SG DMA 方式又满足了数据传输量大时对速度的要求。

关键词:PCI-Express 总线;FPGA;直接存储器访问;WinDriver

中图分类号:TP39

文献标识码:A

文章编号:1673-629X(2013)09-0195-04

doi:10.3969/j.issn.1673-629X.2013.09.049

High Speed SG DMA Data Transmission System Based on PCIE

SUN Xiao-ye

(Shijiazhuang Development Zone Jikeshuangshi Science and Tech. Co., Ltd, Shijiazhuang 050081, China)

Abstract: Combined with the specific application for high-speed data encryption card, introduce PCI Express bus protocol, the structure of the high-speed data transmission system based on PCI Express and hardware circuit of FPGA as the core. The method of using WinDriver to generate PCIE device driver software and realize high speed data transferring with SG DMA under Windows XP operation system is also discussed. The program has already been debugged and DMA data transmission is successful between the computer and the Arria II GX development board. The system uses PCIE IP hard core instead of a dedicated interface chip to realize the PCI Express bus interface logic, greatly improving the flexibility of the design, and the use of SG DMA method can meet the requirement of high speed data transmission.

Key words: PCI-Express bus; FPGA; direct memory access; WinDriver

0 引言

随着技术的发展,具有丰富逻辑资源和强大数据处理能力的 FPGA 的应用,使得数据处理速度大大提高,再加上 DDR II、DDR III 等高速存储设备对数据存取速率的提升,对数据传输速度的要求也越来越高。文中针对高速数据加密卡这一具体应用,主要讨论了采用 PCI Express 总线实现 FPGA 与 PC 机之间的高速数据传输^[1-2]。数据传输采用 DMA 方式,DMA 传输不占用 CPU 资源,具有快速传输大量突发性数据的能力,可以很好地满足数据传输的实时性和高速性要求^[3-4]。

1 PCI Express 简介

PCI Express 是最新的总线和接口标准,是一项高性能、高带宽的 I/O 总线技术,采用了目前业内流行的

点对点串行连接,跟共享并行架构的 PCI 总线相比,每个设备都有自己的专用连接,不需要向整个总线请求带宽,而且可以把数据传输率提高到一个很高的频率,达到 PCI 所不能提供的高带宽。目前一个通道单向速度最高可达到 10 Gbps,而且还有相当大的发展潜力。PCI Express 有 x1、x2、x4、x8、x16、x32 多种通道模式,伸缩性非常强,可以满足不同系统设备对数据传输带宽不同的需求^[5]。

PCI Express 总线的拓扑方式使其具有很好的扩展性和灵活性,与 PCI 总线相比,不仅能提供更高的性能,接口引脚的减少也降低了电路板布线难度,减少了信号线间的干扰。

2 系统组成

基于 PCIE 的 SG DMA 高速数据传输系统由以

收稿日期:2012-12-11

修回日期:2013-03-14

网络出版时间:2013-05-09

基金项目:河北省科学院重大攻关项目(12620)

作者简介:孙晓晔(1981-),女,工程师,硕士,研究方向为嵌入式系统的设计和开发。

网络出版地址:<http://www.cnki.net/kcms/detail/61.1450.TP.20130509.1100.052.html>

FPGA 为核心的硬件电路板和 PC 机上 Windows 下的驱动和应用程序组成^[6],如图 1 所示。

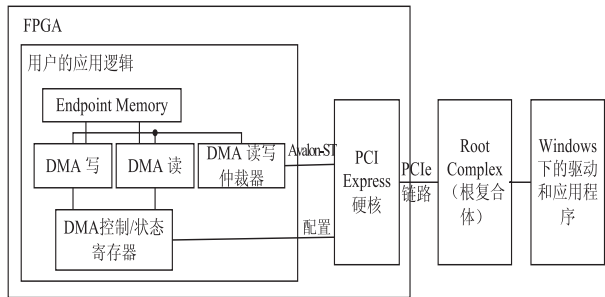


图 1 系统结构图

电路板插到 PC 机主板上的 PCIE 插槽中,板上的 FPGA 通过 PCIE 总线与 PC 机进行 DMA 数据通信^[7]。

3 系统硬件组成

系统硬件采用 Altera 公司的 Arria II GX 开发板套件。这里只介绍 PCIE 数据传输相关的部分。硬件主要由用于逻辑控制的 FPGA 和存储器以及电源电路、时钟电路、FPGA 配置电路等外围电路组成^[8]。FPGA 的逻辑直接使用 Altera 提供的 PCI Express 高性能参考设计的代码,该设计使用的是片上存储器,用户的应用逻辑与 PCI Express 硬核之间采用 128 位的 Avalon-ST 总线^[9]。

3.1 FPGA

FPGA 是整个硬件电路的核心,采用 Altera 公司的 EP2AGX125EF35,具有 124 100 个 LE,576 个 18×18 乘法器,12 个高速收发器,6 个 PLL,49 640ALMs,8 121 kbit 片上存储空间,核心电压为 0.9 V,而且性能适中,适合处理码速率在 20 Mbps 以下、中频载频低于 200 MHz 的中频采样系统。嵌入了可配置硬核 PCIe 接口。器件的硬核 IP 模块实现了 PCIe Gen1 (PIPE) PHY-MAC、数据链路和事务层。可以灵活配置这一 IP 模块,满足端点和根端口应用的需求,其 x1、x4 和 x8 路配置符合 PCIe1.1 和 2.0 的要求^[10]。该系统的 IP 硬核采用 Native Endpoint、x8 通道、最大速率 2.5 Gbps 的配置。

3.2 存储器

Flash 存储器采用 PC28F512P30BF,容量为 512 Mbit,16 位的数据接口,用来存储 FPGA 的逻辑,在板上上电时配置 FPGA。

3.3 电源电路

系统中混合了多种工作电压,有 12 V、5 V、3.3 V、2.5 V、1.5 V、1.1 V、0.9 V 等^[11],通过主板的 PCIE 插槽给板子供电。

3.4 时钟电路

板上有两种类型的时钟输入:全局时钟输入和收

发器参考时钟输入。全局时钟由可编程晶振产生 100 M 和 125 M 的 LVDS 时钟,分别输入到 FPGA 底部和顶部的 PLL,收发器参考时钟由板上 155.52 M 的晶振和可编程晶振产生。此外,还有来自于 PC 机主板的 100 MHz 差分时钟通过 PCIE 连接器输入到 FPGA。

3.5 FPGA 配置电路

提供两种配置方式:JTAG 配置和 FPP 配置。前者通过 USB-Blaster 线,在 Quartus II 编程器中采用 JTAG 模式配置 FPGA,该方式将代码下载到 FPGA 中后,需要重启 PC 机,以使得操作系统能检测到 PCI Express 设备,且这种方式具有掉电丢失的特点,只能用于 FPGA 逻辑调试阶段。后者是将 Quartus II 生成的 .sof 文件转换成 .flash 格式,然后烧写到板上的 Flash 中,在板子上电的时候由外部的 MAX II CPLD EPM2210 控制器从 Flash 中读取 16 位的数据,转换成 8 位的 FPP 格式,然后写到 FPGA 的配置专用引脚,对 FPGA 进行配置。这种快速被动并行 (FPP) 配置方式在 FPGA 逻辑调试成功后使用。

4 驱动程序设计

4.1 利用 WinDriver 开发驱动程序

该系统采用 WinDriver 来开发设备驱动程序^[12-13]。WinDriver 是 Jungo 公司的一套设备驱动程序开发组件,能直接探测硬件,可用于 PCI/PCMCIA/ISA 和 USB 等设备驱动程序的开发,并能很方便地生成驱动硬件的 API 函数。利用 WinDriver 开发设备驱动程序,不需要熟悉操作系统的内核,整个驱动程序中的所有函数都是工作在用户态下的,通过与 WinDriver 的 .Vxd 或者 .Sys 文件的交互来达到驱动硬件的目的^[14]。

该系统中利用 WinDriver 开发驱动程序的步骤大致如下:

- 配置并安装 PCIE 卡到 PC 机;
- 在 WinDriver 的向导中配置设备资源,并对 PCIE 卡进行测试,如对设备的存储空间进行读写、监听中断等;
- 选择开发环境和语言,利用向导生成设备驱动框架,这里采用 Microsoft Visual Studio 2005 C;
- 在相应的开发环境中,移植驱动框架中的有用代码,利用 WinDriver 提供的 API 函数,根据应用需要编写具体的功能代码,建立自己的驱动程序,并进行编译、调试、运行;
- 打包发布驱动程序。

在实际的开发过程中,也可以不利用向导,通过调用 WinDriver 的 API 直接写驱动代码。API 函数的调用顺序如下:

1) WDC_DriverOpen()。打开 WinDriver 驱动和 WDC 函数库。

2) WDC_PciScanDevices()。扫描 PCI 总线,根据供应商 ID 和设备 ID,定位要访问的 PCIE 设备。

3) WDC_PciGetDeviceInfo()。枚举该设备的资源信息(内存、I/O、中断)。

4) WDC_PciDeviceOpen()。打开设备。之后就可以利用 WDC_xxx API 与设备进行通信。若使用中断,可利用 WDC_IntEnable()/WDC_IntDisable()来开/关中断。

5) WDC_PciDeviceClose()。所有操作完成,不再使用该设备,需将其关闭。

6) WDC_DriverClose()。关闭 WinDriver 和 WDC 的库。

4.2 DMA 数据传输的实现

利用 WinDriver 生成的框架代码只能实现最基本的操作,要完成大批量数据的快速传输,需要采用 DMA 方式。在 DMA 的应用中,WinDriver 提供相应的 API 来将卡上的内存映射到用户态程序所能直接访问的地址空间中,然后直接对这个映射后得到的地址进行 DMA 操作^[15]。

DMA 有 Block(块)DMA 和 Scatter/Gather(分散/聚合)DMA 两种。前者只能对物理地址连续的一个数据块进行传输,否则需要多次启动 DMA 操作;而后者只启动一次 DMA 操作,就可以传输非连续存储的多个数据块。Block DMA 比较适合数据量小的情况,而对于大量的数据传输,如传输数据量大于 4 kB(一页)的情况下,SG DMA 的效率要高得多。该系统就采用 SG DMA 方式,即以分散的物理内存块映射连续的用户地址空间。

实现 SG DMA 传输时,需要主机内存中申请一块空间,用来存储描述符表。描述符表由一个表头和多个描述符组成,表头和每个描述符大小都为 4DW,其中每一个描述符对应一次 DMA 操作。描述符表头的第 3 个双字的低 16 位为一个计数单元(RCLAST),在传输开始之前表示还有多少个 DMA 操作等待传输,其值等于描述符表的大小减 1;在不使用中断的传输过程中,可以通过查询该字段来判断 DMA 操作是否完成^[16]。描述符表中,每个描述符包含了控制信息(是否更新描述符表头的 RCLAST 字段、是否采用中断)、DMA 长度、源端地址、目的端地址信息。DMA 操作的流程图如图 2 所示。

等待 DMA 完成时,需要判断采用的是中断还是轮询方式。若是轮询方式,查看描述符表表头第 3 个 DW 的低 16 位,若等于最后一个描述符的序号,则表示所有 DMA 都完成,若 10 s 内 DMA 未完成,则超时

退出。若是中断方式,则等待 DMA 传输结束的中断信号发生,转入中断处理程序运行。由于 Windows XP 不支持 MSI 中断方式,在 XP 下只能采用传统中断。传统 PCI 中断是电平敏感的中断。只要物理中断信号的电平为高,这种中断就会一直产生。如果核心态中断处理程序结束了,中断信号还未变低,操作系统将会不断地调用中断处理程序,导致系统挂起。为了避免这种情况,一收到中断信号,核心态下的中断处理程序就得立即清除中断。

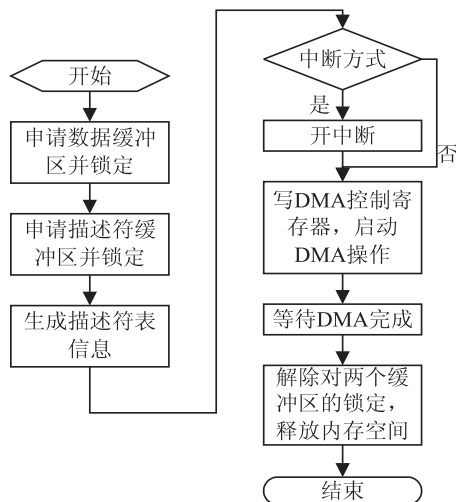


图2 DMA 传输流程图

DMA 操作中用到的主要 API 函数:

- WDC_WriteAddr32:写 DMA 控制寄存器;
- WDC_DMASGBufLock:为 SG DMA 操作申请内存空间并锁定(为了保证传输的连续进行,所分配的这段主机内存必须是非分页内存,以保证不会被交换出去,因此必须使用 WDC_DMASGBufLock 函数);
- WDC_DMASyncCpu:在进行传输之前,为了同步 CPU 高速缓冲寄存器和 DMA 缓存,需要先刷新 CPU 缓冲寄存器中的数据;
- WDC_DMASyncIo:当 DMA 传输结束时,数据可能还在 DMA 缓冲寄存器中,但还没有更新到系统内存中,这样 CPU 可能访问到错误的数据,为了保持一致性,应刷新 DMA 缓冲寄存器中的数据;
- WDC_Sleep:在等待 DMA 操作完成或判断 DMA 是否超时的时候,用来延时;
- WDC_DMABufUnlock:解除对不再使用的内存空间的锁定。

启动 DMA 操作时,写入 DMA 控制寄存器的描述符表地址和描述符表中每个描述符存的 PC 机端 64 位地址均为物理地址。另外,由于采用的 Altera 公司提供的 FPGA 逻辑参考设计代码,不支持 RC 存储器起始地址非 QWORD 对齐,在使用 128 位的 Avalon-ST 接口时,RC 存储器地址必须 128 位对齐。因此,为数据

和描述符申请的内存空间,如果物理起始地址不能被 16 整除,需要进行对齐处理,否则 DMA 传输的数据会出错。

5 结束语

文中针对高速数据加密卡这一具体应用,对基于 PCI Express 的 SG DMA 高速数据传输系统的构成、工作流程以及驱动程序开发进行了介绍,利用现有的 Altera 公司的开发板作为硬件平台,在其上调试成功。基于该思想设计的高速数据加密卡采用 FPGA 的 PCIE 硬核,与采用专门芯片相比,可以更有效地发挥 PCIE 的高带宽优势,并且可以灵活配置,从而从整体上提高系统性能。采用 WinDriver 驱动开发平台,大大降低了 Windows 系统下程序开发的难度,缩短了程序开发与调试的时间。采用 DMA 方式提高了数据传送的能力,可以满足高速数据传输的可靠性,适用于大量数据的高速传输。

参考文献:

- [1] 马 萍,唐卫华,李绪志. 基于 PCI Express 总线高速数采卡的设计与实现[J]. 微计算机信息,2008,24(9-1):116-118.
- [2] 潘玉霞,马游春,熊继军. 基于 PCI Express 总线的高速数据传输卡设计与实现[J]. 电子技术应用,2010,36(8):92-95.
- [3] 刘秀萍. PCI 总线高速数据传输技术实现[J]. 电脑知识与技术,2009,5(9):2339-2340.

(上接第 65 页)

对云计算环境下频繁项集的挖掘是有效可扩展的。

参考文献:

- [1] Agrawal R, Imielinski T, Swami A. Mining association rules between sets of items in large database [C]//Proc. of 1993 ACM SIGMOD Conf. on Management of Data. Washington DC:ACM Press,1993:207-216.
- [2] Agrawal R, Srikant R. Fast Algorithms for Mining Association Rules [C]//Proc. of the 20th Int'l Conference on Very Large Databases. Santiago, Chile: [s. n.], 1994.
- [3] Han J, Pei J, Yin Y. Mining frequent patterns without candidate generation [C]//Proc. of 2000 ACM SIGMOD Int'l Conf. on Management of Data. Dallas, TX, New York: ACM Press,2000:1-12.
- [4] 颜跃进,李舟军,陈火旺,等. 基于 FP-Tree 有效挖掘最大频繁项集[J]. 软件学报,2005,16(2):215-222.
- [5] 杨 云. FP-Growth 算法的改进[J]. 计算机工程与设计,2010,31(7):1506-1509.
- [6] 张玉芳,熊忠阳,彭 燕,等. 基于 FP-Tree 含正负项目的

- [4] Li B, Peng Y, Liu D T. A High Speed DMA Transaction Method for PCI Express Devices [J]. Journal of Electronic Science and Technology of China, 2009, 7(4):94-98.
- [5] PCI Express Base Specification Revision 1.1 [S]. PCI-SIG, 2005.
- [6] 项春雷. 基于 FPGA 的 PCI-E 接口设计 [D]. 郑州:信息工程大学,2011.
- [7] 王嘉良,赵曙光. 用 FPGA 实现 PCI-E 接口和 DMA 控制器设计 [J]. 计算机技术与发展,2011,21(6):181-184.
- [8] Arria II GX FPGA Development Kit User Guide [M]. [s. l.]: ALTERA Corporation, 2011.
- [9] PCI Express High Performance Reference Design [M]. [s. l.]: ALTERA Corporation, 2010.
- [10] IP Compiler for PCI Express User Guide [M]. [s. l.]: ALTERA Corporation, 2011.
- [11] Arria II GX FPGA Development Board Reference Manual [M]. [s. l.]: ALTERA Corporation, 2011.
- [12] 刘 波,库锡树,孙兆林. 基于 PCIE 总线协议的数据采集设备驱动程序实现 [J]. 工业控制计算机,2007,20(7):28-29.
- [13] 田 泽,刘 娟,王绮卉. 基于 WDM 的 PCIE 驱动设计和实现 [J]. 软件导刊,2010,9(4):9-10.
- [14] WinDriver PCI/ISA/CardBus User's Manual Version 10.10 [M]. [s. l.]: Jungo Ltd, 2009.
- [15] 刘晓光. 基于 PCI-E 接口数据采集系统软件设计与实现 [D]. 武汉:华中科技大学,2011.
- [16] 沈 辉,张 萍. FPGA 在 PCI Express 总线接口中的应用 [J]. 现代电子技术,2010(14):109-111.

- 频繁项集挖掘算法 [J]. 模式识别与人工智能,2008,21(2):246-253.
- [7] Agrawal R, Shafer J. Parallel mining of association rules [J]. IEEE Trans. on Knowledge and Data Engineering, 1996, 8(6):962-969.
- [8] Li Haoyuan, Wang Yi, Zhang Dong, et al. PFP: Parallel FP-Growth for Query Recommendation [C]//Proceedings of the 2008 ACM Conference on Recommender Systems. Lusanne, Switzerland: [s. n.], 2008:125-137.
- [9] Zhou L, Zhong Z, Chang J, et al. Balanced Parallel FP-Growth with Map-Reduce [C]//Proceedings of 2010 IEEE Youth Conference on Information Computing and Telecommunications (YC-ICT). Beijing: [s. n.], 2010:243-246.
- [10] 宋 威,刘文博,李晋宏. 基于动态裁剪频繁模式树的频繁项集并发挖掘算法 [J]. 山东大学学报:工学版,2011,41(4):49-55.
- [11] 邱 勇,兰永杰. 高效 FP-TREE 创建算法 [J]. 计算机科学,2004,31(10):98-100.
- [12] 谭峻松,首照宇. 一种分布式环境下动态挖掘频繁闭项集算法 [J]. 大众科技,2010(9):38-40.

基于PCIE的SG DMA高速数据传输系统

作者：[孙晓晔](#)，[SUN Xiao-ye](#)

作者单位：[石家庄开发区冀科双实科技有限公司, 河北 石家庄, 050081](#)

刊名：[计算机技术与发展](#)

ISTIC

英文刊名：[Computer Technology and Development](#)

年, 卷(期):[2013\(9\)](#)

本文链接: http://d.wanfangdata.com.cn/Periodical_wjfz201309049.aspx