RT Embedded http://www.kontronn.com

多通道串行通信设备的 Linux 驱动程序实现

邹荣士[1,2],郭立红[1],司玉美[1,2],赵海波[1,2]

(1. 中国科学院 长春光学精密机械与物理研究所,长春 130033; 2. 中国科学院 研究生院, 北京 100039)

摘　要:以自行开发的 PCI 接口通信卡为例,简要介绍了 PCI 总线在光电测量数据通信系统中的应用,重点讨论了 Linux 设备驱动程序开发方法及虚拟文件层机制等相关概念。针对该通信设备,介绍其在 Linux 下采用内核模块编程方式,实现中断和 DMA 方式数据传送的方法。实验测得此传输方案的本地速率可达 24.096 Mb/s,传输和响应速度较传统读写方式有了显著提高,数据可稳定传输在 460 kb/s,满足系统传输需求。

关键词:计算机应用;设备驱动;直接内存访问(DMA);PCI 总线;内核

中图分类号:TP316;TP336　**文献标识码**:A　**文章编号**:1671-5497(2007)01-0164-04

Implementation of drivers for multi-channel data communication devices in Linux operating system

Zou Rong-shi[1,2], Guo Li-hong[1], Si Yu-mei[1,2], Zhao Hai-bo[1,2]

(1. *Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun*, 130033, *China*; 2. *Graduate University of Chinese Academy of Sciences, Beijing* 100039, *China*)

Abstract: The application of PCI bus in opto-electronics measurement device was introduced based on a self-developed PCI communication card. The approaches of developing device drives in Linux system and related concepts, such as Virtual File System (VFS) mechanism were discussed. For this communication card, the kernel module programming method in Linux, the interruption of implementation and the Direct Memory Access (DMA) were illustrated. Experimental results show that, with this scheme, the local rate can reach 24.096 Mb/s. Comparing with conventional method the transmission and real-time response speed are significantly improved. Data can be steadily transmitted at 460 kb/s which meets the system requirement.

Key words: computer application; device driver; direct memory access (DMA); PCI bus; kernel

随着图像和信息处理技术的进步,当今光电测量设备对各分系统与主控计算机之间的信息交换要求越来越高。为满足高速海量数据信息的传输要求,基于传统 ISA 总线的设备已经不能适应技术发展的需要。PCI 总线作为一种低成本、高性能的局部总线协议,支持即插即用、线性突发传输、总线主控,传送过程不需主机干预[1],目前被广泛应用于高速数据通信和采集。另一方面,光电测量设备对硬件电路的设计有特定的要求,普通的 PCI 通信卡不适合作为专用的通信设备,因此需要开发专用 PCI 硬件设备。在军事及一些测控领域,对系统的可靠性和健壮性有很高的要

收稿日期:2006-03-20.

基金项目:中国科学院二期创新项目(C04708Z).

作者简介:邹荣士(1981),男,博士研究生. 研究方向:Linux 操作系统底层开发技术. E-mail:zours@ciomp.ac.cn

通讯联系人:郭立红(1964),女,研究员,博士生导师. 研究方向:计算机应用. E-mail:guolh@ciomp.ac.cn

Created in Master PDF Editor

RT Embedded http://www.kontronn.com

求;同时还要求系统具有可定制性和良好的扩展能力。Linux 由于源代码开放,内核可定制,安全性高等特点,逐渐成为广泛应用的操作系统。另外,新开发的硬件设备在使用时必须具有满足实际需要的驱动程序。作者介绍了通信卡的结构,重点讨论了在 Linux 下驱动程序的开发方法,提出开辟预缓冲,运用 DMA 和中断读取方式实现传输的方案,并给出了其实现过程。

1 PCI 通信卡硬件结构

1.1 PCI 总线特点

PCI 总线是现今主流的局部总线协议,它采用并行信号机制,时钟频率一般为 0～33 MHz;总线带宽 32 位,可做 64 位扩展;支持猝发传输模式;PCI 2.1 和 PCI 2.2 标准定义的总线传输速率可达 512 Mb/s,还具有低随机访问延迟(对从总线上的主控寄存器到从属寄存器的写访问延迟为 60 ns)、支持资源的自动分配、提供数据和地址的校验功能等特点。

PCI 总线结构具有明显的优点,但总线协议复杂,时序关系要求严格,接口的设计难度较大。因此,许多公司设计了 PCI 总线到本地总线的接口控制芯片,将复杂的 PCI 总线接口转换为相对简单的用户接口。利用这种专用芯片,再根据需要设计外围电路就可以把设计聚焦在特定的应用之上。选用的 PCI 9054 接口芯片是 PLX 公司推出的 33 MHz、32 位 PCI 总线接口控制器,它符合 PCI 2.2 规范,可同时支持 3.3 V 和 5 V 两种信号环境,并且具有电源管理功能,性价比较高[2]。

1.2 PCI 数据通信卡结构框图

根据系统需求,该多通道通信卡接收外围多路状态数据,要求对收到的数据快速响应和存储。其基本结构框图如图 1 所示。

主控计算机与分系统之间的数据交换采用 RS422 接口。RS422 是被广泛采用的在通讯距离为几十米到几千米时的全双工通讯方式。它采用平衡发送、差分接收的方式,灵敏度高,能检测低达 200 mV 的电压,传输信号可在很远处得到恢复[3]。本通信卡采用 SN75175 作为差分接收器,SN75174 作为差分发送器。

通用异步收发器(UART)采用 TL16C554。TL16C554 内部集成 4 个改进的 TL16C550 异步通信单元(ACE);在外设接收到串行数据时,TL16C554 自动完成串并转换并存储;在 FIFO

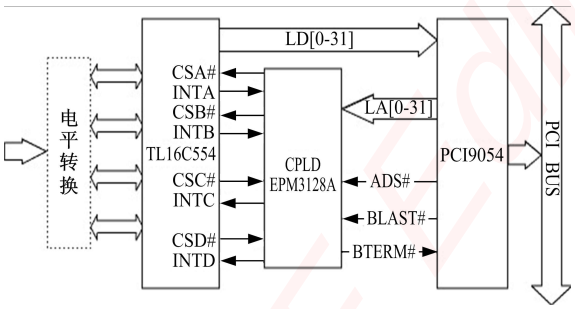

图 1 PCI 多通道数据通信卡硬件结构框图

Fig. 1 Hardware block diagram of PCI multi-channel data communication card

模式下,每个 ACE 使用 16 字节的 FIFO 作为缓冲以减少中断次数。数据到达后,若引起 FIFO 深度中断,在 CPLD 控制下向 PCI9054 提出申请,产生 PCI 中断信号[4]。主机发起 DMA 数据传输,数据由 TL16C554FIFO 转入主机内存。采用 CPLD 控制的优点在于可动态改变控制逻辑,硬件电路简单可靠。选用的 EPM3128A 是 ALTERA 公司的 MAX3000 系列可编程逻辑器件[5]。

2 Linux 驱动程序设计

2.1 内核模块编程

Linux 操作系统中,设备驱动程序可以静态编译进入内核;也可以以模块的方式动态加载。模块化的设备驱动程序提供了与静态编译进入内核的代码同样的功能,同时又使得内核映像不过于庞大,提供了最大的灵活性。系统提供 insmod/rmmod 命令将模块动态链接到内核。在模块编程中,用户需要编写内核模块接口函数 init_module()/cleanup_module()。这两个函数在模块被加载和移除时由内核调用。

Linux 将外部设备分为 3 种基本类型:字符设备、块设备、网络设备。设备驱动程序模块分为字符模块、块模块和网络模块[6]。字符设备操作的对象面向字节流,块设备与字符设备的主要区别在于内核内部管理数据的方式不同。字符设备驱动程序的接口相对清晰而且易于使用。根据硬件的特点,本驱动程序使用内核提供的字符驱动程序接口,以模块的方式动态加载。

2.2 虚拟文件系统和设备文件

虚拟文件系统(VFS)是在内核中实现的软件层,目的在于对异构文件系统提供统一化的处理接口。Linux 内核将设备操作视为文件操作,用

RT Embedded http://www.kontronn.com

户态程序对设备操作的接口类似于普通文件操作[7]。驱动程序开发者需要面向虚拟文件系统层,提供必要的操作接口,这些接口包括open( )、release( )、read( )、write( )、lseek( )、ioctl( )等系统调用。它们存在于file_operations结构中。具体可参见<linux/fs.h>。

如果未采用devfs文件系统,则内核对外部设备的访问通过位于文件系统/dev目录下的设备文件实现。设备文件包含了对应设备的主设备号和次设备号。要求主设备号范围为1-254,且在内核空间中保持唯一。主设备号标识设备对应的驱动程序,次设备号用来区分同种设备的不同实例,由驱动程序使用,具体含义由驱动程序解释。内核通过设备号唯一标识设备。设备文件可通过mknod命令创建。用户态程序对硬件设备的操作过程见图2。

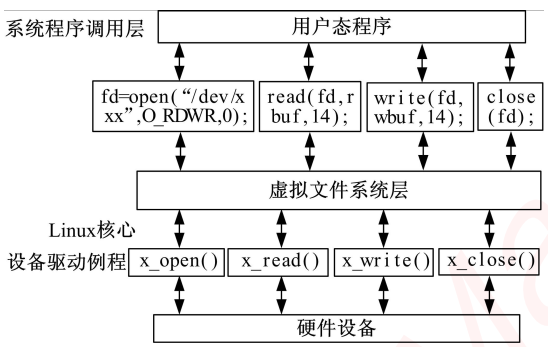

图2 Linux硬件设备管理模型示意图

Fig. 2 Schematic digram of management model of Linux hardware equipment

3 Linux对PCI设备接口支持

3.1 设备的初始化

支持硬件设备的即插即用是PCI总线的优点之一。内核在系统启动时完成对所有PCI设备的扫描、登录和分配资源等初始化操作,并建立起系统中所有PCI设备的拓扑结构。设备驱动程序对PCI设备的访问通过pci_dev结构实现,pci_dev全面描述了PCI设备的详细信息,包括设备所在总线位置、设备在/proc/bus/pci的入口信息、编码后的设备功能索引、设备类别代码、为此设备分配的pci_driver结构、为此设备分配的中断号及内存资源等[8]。当PCI驱动程序需要对设备进行初始化时,可以按如下步骤进行:

(1)调用函数pci_present( )检查PCI总线是否已经被Linux内核支持。

(2)填充pci_driver结构,完成硬件初始化。

(3)调用pci_module_init( )函数注册PCI设备的驱动程序。

pci_driver结构是内核实现PCI设备驱动程序管理的核心数据结构,它描述了驱动程序名字、设备驱动程序关心的设备列表(struct pci_device_id)、新设备插入探测函数(int (*probe)( ))、设备移出函数(void (*remove( )))、设备挂起、唤醒、唤醒事件等函数。在调用pci_module_init( )时,内核调用pci_register_driver( )将新的pci_driver结构挂在内核pci_driver链表上。发生相应事件时,内核通过查找此链表调用对应处理函数。

3.2 中断与DMA操作

PCI 9054集成两个DMA通道,均支持块DMA和分散/集中DMA。块DMA多用于一次传输集中的大块数据,分散/集中DMA将本地总线或PCI总线上分散的数据一次性地通过DMA方式传输到PCI或本地存储空间。本驱动程序采用BLOCK DMA与中断结合的方式读取通过多通道串口卡传输的数据信息。

读取数据的传输流程如下:

(1)在进程调用read时,驱动程序中相应的函数分配一个DMA缓冲区,随后指示硬件传输它的数据;进程进入睡眠。

(2)硬件将数据写入DMA缓冲区并在完成时产生一个中断。

(3)中断处理程序获得输入数据,应答中断,最后唤醒进程,通知数据到达。

设备驱动程序采取的中断服务程序流程如图3所示。

可以看出,高效的DMA处理依赖于高效的中断处理过程。内核对来自外部设备的中断处理过程如下:在接收到外部中断信息后,Linux内核查询IDT表,IDT表依照中断源的位置按序组成,并且对应中断服务程序的入口地址,依据找到的地址,系统调用相应的服务程序进入中断处理,处理完成后恢复中断前的现场并返回[9]。从CPU中断接收线上出现中断信号到进入设备驱动程序中断服务程序所需要的时间称为中断延迟。中断延迟的大小与内核的工作载荷及采用的调度策略有关[10]。

RT Embedded http://www.kontronn.com

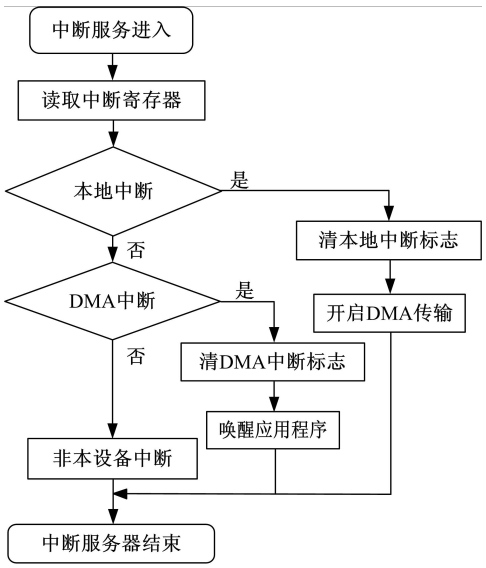

图 3　PCI 多通道卡中断处理流程

Fig. 3　Flow chart of interrupt processing for PCI multi-channel communication card

4　结束语

本通信设备结合 PCI 总线接口技术和 Linux 模块化编程技术,利用 PCI 9054 的总线主控 DMA 能力实现了光电测控系统中主控计算机与分系统的快速可靠通信。经实测,在 Redhat9.0 kernel 2.4.20-8 的操作系统中,本通信卡的外部传输速率可以稳定在 512 kb/s。本地传输速度大约在 24.096 Mb/s 左右,满足了系统设计要求。

参考文献:

[1] 韩慧,于守谦,刘亚斌,等. 基于 PCI 总线图像采集卡开发的图像处理系统[J]. 计算机测量与控制, 2003,11(6):455-457.
Han-hui, Yu Shou-qian, Liu Ya-bin, et al. Image processing system based on the matrox meteor-Ⅱ Image grab card[J]. Computer Measurement and Control, 2003,11(6):455-457.

[2] PCI9054 Data Book. PLX Technology Inc[EB/OL]. [2006-01-09]http://www.plxtech.com.

[3] 张增辉,沈激,韦东山,等. 基于 PCI 总线的 RS422 并行数据接口[J]. 核电子学与探测技术, 2005, 25(6):838-840.
Zhang Zeng-hui, Shen-ji, Wei Dong-shan, et al. Parallel data grabbing card based on PCI bus RS422[J]. Nuclear Electronics and Detection Technology, 2005, 25(6):838-840.

[4] 司玉美,邹荣士,郭立红. PCI 串口通信卡 WDM 驱动程序设计与实现[J]. 微计算机信息, 2005,21(11/12):145-157.
Si Yu-mei, Zou Rong-shi, Guo Li-hong. Design and implementation of WDM driver for PCI serial communication card[J]. Microcomputer Information, 2005,21(11/12):145-147.

[5] 吴忠杰,林君,谢宣松. 基于 PCI 总线的高速数据采集系统[J]. 吉林大学学报:信息科学版, 2005,23(2):167-171.
Wu Zhong-jie, Lin-jun, Xie Xuan-song. Design and implementation of high speed data acquisition system based on PCI bus[J]. Journal of Jilin University (Infomation Science Edition), 2005,23(2):167-171.

[6] Rubini Alessandro, Corbet Jonathan. Linux Device Drivers(2nd ed)[M]. USA: O'REILLY, 2001:6-7.

[7] 王军,王宏,徐皑冬,等. 基于 Linux 的 EPA 无线通信卡的实现[J]. 计算机应用, 2005, 25(10):2436-2438.
Wang-jun, Wang-hong, Xu Kai-dong, et al. Realization of EPA wireless communication card based on Linux[J]. Computer Applications, 2005, 25(10):2436-2438.

[8] 姜明华,周敬利,黄晓涛. Linux 下加密卡驱动程序的开发与性能分析[J]. 计算机工程, 2004,30(16):185-187.
Jiang Ming-hua, Zhou Jing-li, Huang Xiao-tao, Development and performance analysis of device driver for encryption card under Linux[J]. Computer Engineering, 2004,30(16):185-187.

[9] 李善平,刘文峰,李程远,等. Linux 内核 2.4 版源代码分析大全[M]. 北京:机械工业出版社, 2002:34-35.

[10] 历海燕,李新明. Linux 抢占式内核的研究与实现[J]. 计算机工程与设计, 2005, 26(9):2395-2399.
Li Hai-yan, Li Xin-ming. Research and implementation on preemptive kernel of Linux[J]. Computer Engineering and Design, 2005, 26(9):2395-2399.

Created in Master PDF Editor