

基于 ARM 的数据采集系统并行总线的驱动设计

刘少真^{1,2}, 孔祥成¹, 裴国玺¹, 乐琪¹, 慕振成^{1,2}

(1. 中国科学院高能物理研究所, 北京 100049; 2. 中国科学院研究生院, 北京 100049)

摘要: 针对中国散裂中子源(CSNS)波形记录与报警系统的应用需求, 设计实现了 ARM - Linux 平台下基于 AHB(高速外围总线)总线通信的 FPGA 设备驱动程序。对数据通信接口设计、FPGA 设备驱动的开发方法、内存映射机制和中断处理等关键环节做了详细描述。该设计已应用于 CSNS 波形记录与报警系统中, 性能稳定。

关键词: CSNS 波形记录与报警系统; ARM; AHB 总线; FPGA 设备驱动; 中断处理

中图分类号: TL 503.6 **文献标识码:** A **文章编号:** 0258-0934(2011)04-0399-04

加速器控制系统, 其数据采集和通信多采用基于商业的嵌入式实时操作系统 Vxworks 开发^[1]。而 Linux 作为一个低成本的操作系统, 能够为系统开发节省大量资金, 并且具有开放源码的特点, 方便系统的开发与调试。ARM 微处理器支持嵌入式 Linux 操作系统, 将前端数据采集系统中广泛使用的 FPGA 与基于 ARM 的嵌入式应用相结合应用于加速器数据采集传输系统中, 则可在实现高速数据采集的同时降低系统的成本。

CSNS 波形记录与报警系统的数据采集和处理部分采用 FPGA + ARM 的设计方案。该系统除了能够提供每秒钟一次的波形显示之外, 还可以记录在发生异常的时刻前后多达 3 min 的数据。其中 ARM 微处理器的任务是利用 100 M 以太网实现与上位机数据交换, 并完成上位机对负责前端采集的 FPGA 各种参数的配置。可以想象, ARM 的任务比较繁重, 这就要求 ARM 与 FPGA 之间的通信驱动程序合理可靠。尽管本项目理论上 ARM 对 FPGA 的数据

吞吐率只需要 160 kbps ~ 1.6 MBPS, 但是实际上为了有效安排 ARM 及 FPGA 任务的调度, ARM 对 FPGA 的通信吞吐率的设计值要达到实际使用的 10 倍以上。本设计是为该系统开发的专用驱动, 以能够提供大的数据吞吐率, 简化 FPGA 接口复杂性作为设计目标。

1 系统硬件设计

1.1 硬件平台

CSNS 波形记录与报警系统的硬件部分, 选用以 Samsung 公司的 ARM9 系列处理器 S3C2440A^[2]为核心并集成 AHB 总线的嵌入式主板, 加上自行设计基于 FPGA^[3]的带有 AHB 总线通信接口的数据采集板。由于本设计的目标就是基于 FPGA 的数据采集板, 下文将其定义为 FPGA 设备。

1.2 ARM 与 FPGA 接口设计

ARM 的 AHB^[2]总线主要用于连接高速外设, 如存储器控制器、中断控制器等, 传送时序基于 HCLK。S3C2440A 的 AHB 总线包括 32 位的地址总线和 32 位数据总线, 其存储器控制器控制对外部 0 ~ 1 G L/O 内存空间的访问(分为 8 个 Bank, 片选信号为 nGCS0 ~ nGCS7)。本设计中 FPGA 作为一个外部存储设备与 ARM

收稿日期: 2010 - 12 - 28

作者简介: 刘少真(1987 -), 女, 河南省平顶山市人, 博士研究生, 从事嵌入式系统开发。

的一些控制信号相连, ARM 和 FPGA 接口设计图如图 1 所示。

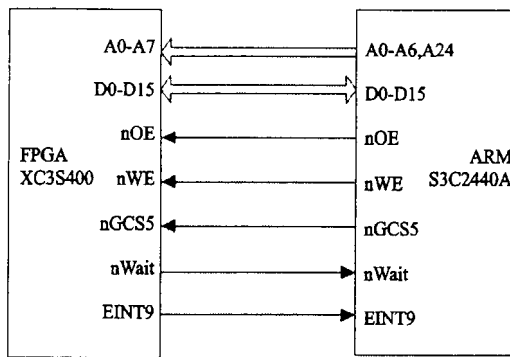


图 1 ARM 和 FPGA 接口设计图

其中, A0 ~ A6, A24 为地址信号(本设计用到 32 根地址线中的 8 根), nGCS5 为片选信号, nWE 和 nOE 分别为写使能和读使能信号, 以上信号均为 ARM 的输出信号。nWait 为等待信号, 是 ARM 的输入信号。D0 ~ D15 为数据信号, 为双向信号。该系统中 FPGA 与 ARM 之间采用 16 位并行数据传输, Bank5 的数据总线宽度由 BWSCON[21:20] 的值决定, 默认设置即为 16 位, 因此无需改动。图 2 为 S3C2440A 的外部 nWait 读时序。

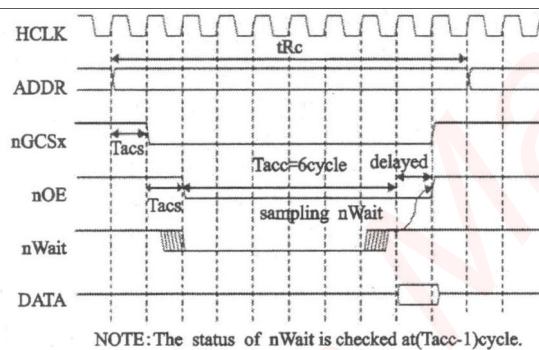


图 2 外部 nWait 读时序图

2 FPGA 设备驱动的设计

Linux 下的设备分为三类, 分别为: 字符设备, 块设备和网络设备。本设计中, 将 FPGA 作为字符设备来开发其设备驱动。Linux 2.6 内核中使用 cdev 结构体描述字符设备^[4]。cdev 结构体的重要成员 file_operations 定义了字符设备驱动提供给虚拟文件系统的接口函数。

FPGA 设备驱动程序要完成的任务有以下几点: 字符设备的注册和注销、I/O 内存资源的申请和释放、虚拟地址的映射和释放、中断的申请、注册和释放、file_operations 结构体中一些成

员函数的实现。下面重点讲述模块加载函数、中断处理函数以及读写函数的实现。由于模块加载函数和模块卸载函数实现的功能是成对出现的, 故此仅对模块加载函数进行详细介绍。

2.1 模块加载函数

通过交叉编译器 arm - linux - gcc 将驱动程序编译为内核模块, 这样可以动态加载模块, 方便调试。驱动程序中以“module_init (fpga_init ())”的形式指定了 fpga_init () 为模块加载函数。在调试阶段, 使用 insmod 或 modprobe 命令加载模块时, fpga_init () 就会自动被执行。该函数完成字符设备的注册、I/O 内存资源的申请、虚拟地址的映射。

2.1.1 字符设备的注册

在驱动程序的开头定义了 fpga_devt 结构体, 其成员为 cdev 结构体和 semaphore 结构体。之后定义了 fpga_devt 类型的变量 fpga_dev。在 fpga_init () 函数中, 首先申请主设备号 fpga_major, 然后初始化 fpga_dev.cdev 的成员, 最后向系统添加 FPGA 字符设备。关键代码如下:

```

{
    dev_t devno = MKDEV ( fpga_major, 0 );
    register_chrdev_region ( devno, 1, DEVICE_NAME );
    cdev_init ( &fpga_dev.cdev, &fpga_fops );
    cdev_add ( &fpga_dev.cdev, devno, 1 );
}

```

2.1.2 I/O 内存资源的申请、虚拟地址的映射

在本设计中, 将 ARM 地址线 A[24, 6 ~ 0] 接入 FPGA 的 I/O 管脚, 由于 nGCS5 对应的 Bank5 的物理地址空间范围为 0x28000000 ~ 0x29FFFFFF, 所以本 FPGA 设备所处的物理地址范围为 0x28000000 ~ 0x2800007F (A24 = 0) 和 0x29000000 ~ 0x2900007F (A24 = 1)。由于 Linux 驱动程序和应用程序都只能对虚拟地址进行操作, 因此需要把 FPGA 设备的物理地址映射为虚拟地址。首先使用 Linux 内核提供的 request_mem_region () 函数, 申请 FPGA 所处物理地址空间的一段内存空间, 申请成功之后该段内存空间即被标志已经使用, 不能被其他驱动申请。之后调用 ioremap () 函数将该段空间的物理地址映射为虚拟地址。

需要注意的是, 在 FPGA 的物理地址被映射为虚拟地址之后, 要使用 Linux 内核提供的 ioread16 ()、iowrite16 () 等函数来完成对虚拟

地址的读写。

2.2 中断处理函数

在 `fpga_open()` 函数中调用 `set_irq_type()` 设置中断触发的类型,本设计中将外部中断 EINT9 配置为下降沿触发式。调用 `request_irq()` 函数申请 IRQ,在该函数中定义了中断处理函数 `receive()`。当 EINT9 中断发生时, `receive()` 会自动被调用。在该函数中,首先读出中断状态值,然后根据不同的状态值,读出对应的数据。数据读出完毕之后,会释放信号量。主要代码如下:

```
static irqreturn_t receive(int irq, void * dev_id)
{
    int_status = ioread16(addr_int_status);
    if(int_status&0x0004)
        ioread16(addr_standard_data);
    else if(int_type&0x0001)
        ioread16(addr_normal_data);
    up(&fpga_dev.sem);
    return IRQ_HANDLED;
}
```

2.3 读写函数

读写函数的主要代码如下:

```
static ssize_t fpga_read(struct file * filp, char __user * bufp, size_t count, off_t * ppos)
{
    down_interruptible(&fpga_dev.sem);
    copy_to_user(bufp, kbuf, count);
}
```

在 Linux 设备驱动中,多个并发执行的单元对共享资源的访问很容易导致竞态,解决竞态问题,需要保证对共享资源的互斥访问。因此在 `fpga_init()` 函数中,调用 `sema_init()` 函数将信号量 `fpga_dev.sem` 初始化为 0。 `fpga_read` 函数必须等待 `receive()` 函数从 FPGA 中读取完数据之后,才能执行 `copy_to_user(bufp, kbuf, count)` 操作。这是信号量用于同步的典型实例。

写函数主要是通过 `copy_from_user()` 将数据从用户空间拷贝至内核空间,再调用 `io_write16()` 函数将数据写到数据总线。

2.4 FPGA 设备地址分配

CSNS 波形记录与报警系统需要用到 12 个地址,用以写入不同的参数和读出不同类型的数据。表 1 为本 FPGA 设备地址分配及主要功

能描述。限于篇幅,只列出有代表意义的 6 个地址。

表 1 主要地址分配及功能描述

FPGA 地址	对应 ARM 地址	功能描述
0x 80	0x29000000	ARM 读出正常波形数据
0x 81	0x29000001	ARM 读出标准波形数据
0x 82	0x29000002	ARM 读出中断状态
0x 83	0x29000003	ARM 写入系统工作模式
0x 84	0x29000004	ARM 写入采样率
0x 85	0x29000005	ARM 写入校验偏差值

3 程序测试

CSNS 快循环同步加速器(RCS)的主二极磁铁工作电流为带直流偏置的 25 Hz(周期为 0.04 s)的正弦波^[5]。在实验室模拟该波形作为系统的测试波形(如图 3 所示)。FPGA 负责进行波形采集,采样率为 10 k,即每个波形采样 400 个点。FPGA 内部存储 800 个点之后,向 ARM 发送中断,ARM 将数据读出。图 4 为测试波形和从 ARM 接收到的两组数据还原的波形对比图。

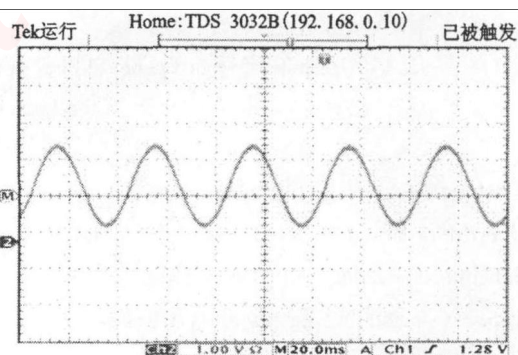


图 3 测试信号的波形

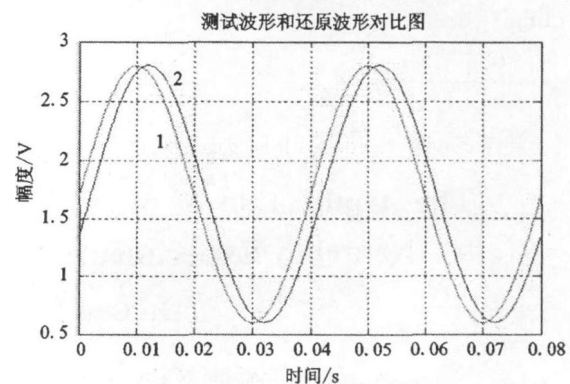


图 4 波形对比图

在图 4 中,1 为测试波形,数学表达式为 $1.7 + 1.1 * \sin(2 * \pi * 25 * t + \theta)$,2 为由采集数据还原出的波形,从图中可以看出两者之间

只相差了一个相位。通过调整 θ 可以使两个波形重合,但是为了便于辨别两个波形,这里取 $\theta = 0$ 。测试证明了 FPGA 设备驱动能够准确、快速地读出数据,满足系统中 FPGA 与 ARM 之间通信的要求。

4 结语

本文详细讲述了 ARM - Linux2.6 平台下, ARM 与 FPGA 并行总线接口的设备驱动设计实现方法。最终将该设计应用于 CSNS 波形记录与报警系统中,证明驱动的性能达到了预期效果。本设计数据吞吐率实测达到 112 MBPS,为系统节约了数据获取运行时间,解决了系统中 FPGA 与 ARM 之间进行通信的关键问题。成功开发出的以 ARM 和 FPGA 为核心的嵌入式数据采集系统,为嵌入式 Linux 操作系统进一步应用于加速器控制系统打下了基础,对其

他 ARM - Linux 相关设备驱动的开发有一定的参考价值。

参考文献:

- [1]王春红. BEPCII 控制系统前端 VME 系统的研究[J]. 核电子学与探测技术,2004,24(5):521-525.
- [2]Datasheet: S3C2440A 32 - BIT CMOS MICROCONTROLLER USERS MANUAL Revision 1[R/OL]. <http://www.samsungsemi.com>; Samsung Electronics Co., Ltd., 2004.
- [3]Datasheet: Spartan - 3 FPGA Family: Complete Data Sheet [R/OL]. <http://www.xilinx.com>; Xilinx, 2005.
- [4]宋宝华. Linux 设备驱动开发详解[M]. 北京:人民邮电出版社,2008:115-156.
- [5]中国科学院高能物理研究所,中国散裂中子源工程(CSNS)概念设计报告[R]. 北京:中国科学院高能物理研究所,2004.

Design and Implementation of Embedded Data Communication Interface Driver Based on ARM

LIU Shao - zhen^{1,2}, KONG Xiang - cheng¹, PEI Guo - xi¹, LE Qi¹, MU Zhen - cheng^{1,2}

- (1. Institute of High Energy Physics, Chinese Academy of Sciences, Beijing 100049, China,
2. Graduate University of the Chinese Academy of Sciences, Beijing 100049, China)

Abstract: According to the application requirement of CSNS wave recording and alarming system, a FPGA device driver based on AHB bus under ARM - Linux platform was designed and implemented. Data communication interface, the method of writing FPGA device driver, memory mapping mechanism and interrupt handling were given detailed discussion. It has been used in CSNS wave recording and alarming system, showing stable and reliable performance.

Key words: CSNS wave recording and alarming system, ARM, AHB bus, FPGA device driver, interrupt handling

(上接第 394 页,Continued from page 394)

The Application of MVC Design Pattern in Daya Bay Reactor Neutrino Experiment's Online Safety Training System

LIU Guan - chuan, CHU Yuan - ping

(Institute of High Energy Physics, Beijing 100049, P. R. China)

Abstract: The article made an introduction to MVC, which is an architectural pattern used in software engineering. It specified the advantages and disadvantages of MVC and also the application of MVC in Daya Bay nuclear reactor neutrino experiment online safety training system.

Key words: MVC design pattern, CodeIgniter, Daya Bay Reactor Neutrino Experiment