

基于 Cy7c67300 USB 控制器和 FPGA 的 USB 接口传输系统

黄 义 翟国芳 万 昊 成桂梅
(北京空间机电研究所 北京 100094)

摘 要: 描述了一个基于 PC 和 FPGA 通信为主要开发目标的 USB 接口传输系统及解决方案。采用 Cypress 的 CY7C67300 USB 控制器作为硬件芯片,并通过 Cypress 自主研发的软件来设计此系统固件程序。最终实现 PC 与 FPGA 的通信,达到系统设计目标。

关键词: USB; FPGA; CY7C67300; 通信; 控制器

中图分类号: TN91 **文献标识码:** A **国家标准学科分类代码:** 510.5010

USB Interface data transfer system based on Cy7c67300 USB controller and FPGA

Huang Yi Zhai Guofang Wan Min Cheng Guimei
(Beijing Institute of Space Mechanice & Electricity, Beijing 100094, China)

Abstract: An USB interface data transfer system and solution based on the communication between PC and FPGA as the main purpose for development is described in this paper. The CY7C67300 USB controller developed by Cypress Company was adopted as the hardware chip in this paper and the firmware program of the system was designed with the software made by the Cypress itself. So we could reach the goal of the system design to realize the final communication between PC and FPGA.

Keywords: USB; FPGA; CY7C67300; communication; controller

1 引 言

随着图像技术的发展,上下位机形式的计算机通信数据越来越大,经常会收发数百兆的数据。而传统的 RS232 串口通信协议就显得有些力不从心了,速度低,研制周期长,致使工作效率下降。

现在 USB 接口的传输速度已经大幅度提升,USB3.0 超高速接口的传输速度已经达到 5 Gbits/s,而且 USB 接口已经作为常规接口被应用于 PC 和许多外接设备上。

以 USB2.0 全速接口为例,其传输速度已经远远大于串口的传输速度,所以使用 USB 接口进行 PC 与 FPGA 之间的图像数据传输,可以减少数据传输时间,增加工作效率。

传输系统将使用 Cypress 的 USB2.0 全速芯片 CY7C67300 作为 USB 控制器进行设计开发。

2 系统结构与原理

2.1 基于 Cy7c67300 与 FPGA 的 USB 传输系统

如图 1 所示,USB 传输系统硬件主要由 PC、基于

FPGA 的 USB 固件(Cy7c67300)和 FPGA 逻辑 3 部分组成。软件部分由 PC 应用程序(C++)、USB 固件程序(C 语言)以及 FPGA 接口程序(VHDL)3 部分组成。

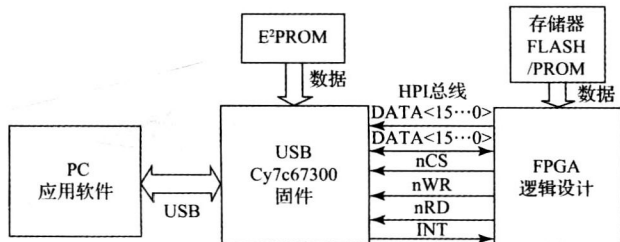


图 1 基于 FPGA 的 USB 传输系统系统

2.2 USB 数据传输类型选择

USB 体系一共定义了 4 种类型的传输,它们是:

控制传输:主要用于设备连接时对设备进行枚举以及其他因设备而定的特定操作。

中断传输:用于对延迟要求严格、小量数据的可靠传

输,如键盘、游戏手柄等。

批量传输:用于对延迟要求宽松,大量数据的可靠传输,如 U 盘等。

同步传输:用于对可靠性要求不高的实时数据传输,如摄像头、USB 音响等。

系统是要求由上位 PC 通过 USB 接口向 FPGA 发送命令,并接收由 FPGA 送回的命令和数据。所以不会在全双工模式下工作,但系统可能会经常收发一些比较大的数据包,所以对传输速度有较高的要求。如表 1 所示,能够很直观地发现,4 种传输模式中传输速度最快的是批量传输模式。其中速可达到 1 216 Kbytes/s,高速可达到 53 248 Kbytes/s,能够高于其他传输模式将近 1 倍。所以综上所述,系统选择批量传输方式作为 USB 的数据传输方式。

表 1 不同数据传输类型的最大传输速率对比

传输类型	每一个端点的最大数据传输速率(Kbytes/s)		
	高速	低速	中速
控制	24	832	15 872
中断	0.8	64	24 576
批量	不允许	1 216	53 248
同步		1 023	24 576

3 固件程序原理与设计

3.1 固件程序架构

如图 2 所示,固件程序框架分为 3 层即 3 个子系统。最底层是 BIOS 层,BIOS 层搭建了程序事件的基本框架,其中包含 PC 端的硬件程序驱动。BIOS 层向上面 1 层是构架层(Frameworks Subsystem),此层主要用于定义接口号,接口功能,传输类型、速度等信息。在往上 1 层是应用层(Application Subsystem)即应用程序代码。想要实现的一些程序功能,主要在这 1 层中进行编写。每个层又有 3 个不同的任务构成。

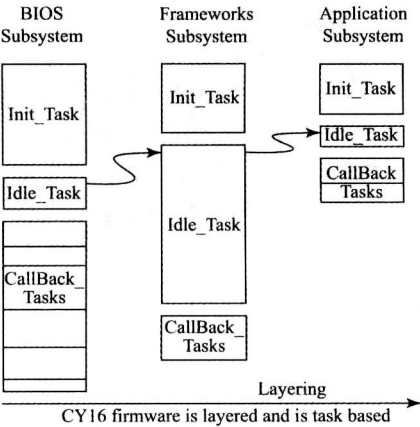


图 2 CY16 固件程序架构及任务构成示意

Init_Task(初始化任务)是负责设立 Idle_Task(空闲任务)和 CallBack_Tasks(回调任务)的。Init_Task 只运行 1

次。Idle_Task 在本系统中为空既无具体程序,当此层中无程序运行时,使系统进入待机状态。CallBack_Tasks 中运行的程序,作为一个结果或某些事件发生。此事件可能是一个硬件中断,软件中断,或者一些在 Idle_Task 中的数据

3.2 Cy7c67300 独立工作模式下的固件程序开发

3.2.1 固件程序开发流程

固件程序开发流程如图 3 所示。

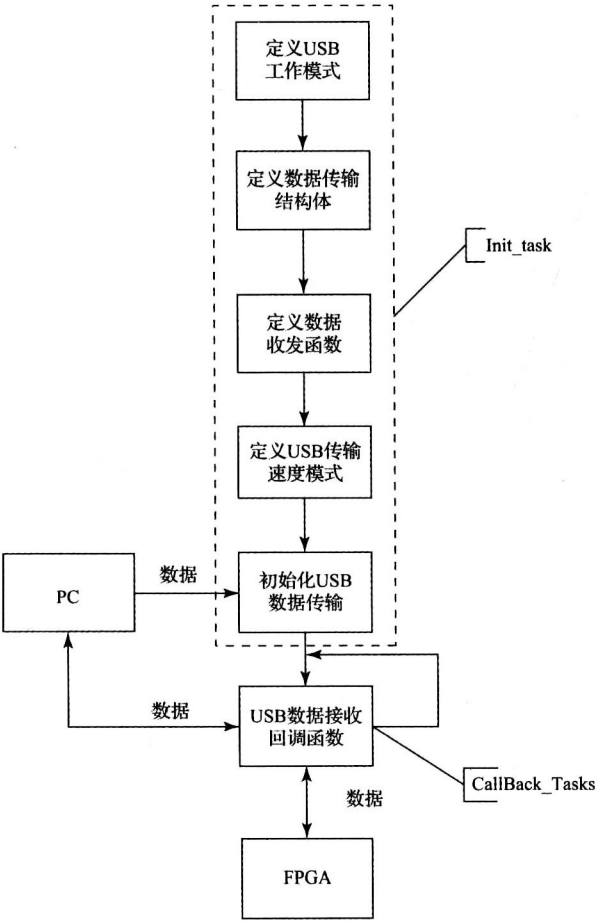


图 3 固件程序开发流程

3.2.2 PC 与 USB 接口通信固件程序设计

首先,需要定义收发端点和相关配置信息,配置信息如下所示:

工作模式:

SYSTEM_MODE_SLAVE(独立工作模式)

数据包大小:

64(这是数据包最大值,减小会降低传输速度)

数据函数:

```
static void siel_bulk_out_done(void);
static void siel_bulk_in_done(void);
```

上面所示函数用于发送/接收上位 PC 的数据。

然后,系统将开始初始化数据收发程序,如果无数据收

发时进入待机模式,当再有数据产生时,进入回调函数开始数据传输。

```
static void sie1_bulk_in_done(void)
{
    sie1_bulk_frame.length=BULK_BUFFER_SIZE;
    sie1_bulk_frame.buffer = sie1_bulk_buffer;
    sie1_bulk_frame.done_func= sie1_bulk_out_done;
    sub_receive(SIE1,DE2_BULK_OUT_PIPE,&sie1_bulk_frame);
}
```

以上程序就是 USB 芯片用来接收来自上位 PC 的回调函数。上面程序中,利用了之前定义的结构体 Sie1_bulk_frame 分 3 行程序去完成传输过程。Buffer 指定数据缓存空间,length 计算数据包大小,dong_func 指定回调函数。usb_receive()函数主要完成接收数据的过程。

接收完数据后回调发送函数 sie1_bulk_out_done(void),进入数据发送模式,没有数据则再次进入待机模式,等待再次有数据发送或接收。系统中的回调接收函数 sie1_bulk_in_done(void)的结构与 sie1_bulk_out_done(void)的形式相同,在这里就不在赘述。

3.2.3 USB 端与 FPGA 接口通信固件程序设计

发送数据到 FPGA 需要遵循图 4 所示的发送时序,根据发送时序编写了 write_fpga(uint16 data)函数用于发送数据。

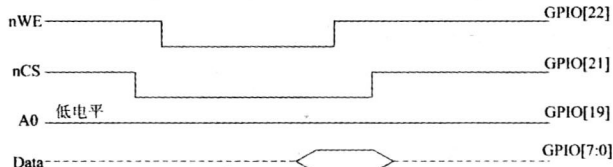


图 4 USB 端向 FPGA 主控芯片发送数据时序

结合图 4 时序图可知,发送数据给 FPGA 主控芯片,首先将 A0 置为低电平,然后将 nCS 使能信号置低电平,nWE 置为低电平,准备发送数据。在 nWE 上升沿时开始发送数据,然后将 nCS 置高电平。这一过程为一个数据发送循环。

```
while(i<64)
{
    write_fpga(sie1_bulk_buffer[i]);
    i++;
}
```

还需要添加以上语句在回调函数中,才能成功将上位 PC 传来的数据转发出去。

在读取 FPGA 主控芯片数据时,由图 5 所示,将 A0 置为高电平,然后将 nCS 使能信号置低电平,nRD 置为低电平,准备数据接收。在 nRD 上升沿时开始接收数据,然后 nCS 使能信号置为高电平。这一过程为一个数据接收循环。其他程序语句编写与发送相同,在此不再赘述。

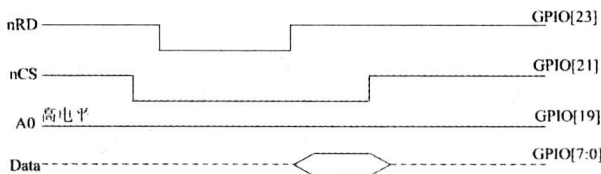


图 5 USB 端向 FPGA 主控芯片发送数据时序

4 Cy7c67300 独立工作模式下 FPGA 程序开发及 PC 应用软件

4.1 Cy7c67300 独立工作模式下 FPGA 程序开发

图 6 所示的是 USB 芯片向 FPGA 发送数据时的发送方法。利用 2 个 FIFO 来实现 USB 端的写模块向 FPGA 端的读模块传输数据的。由 FPGA 向 USB 发送数据是应用 Camera Link 接口来完成发送的,整个发送接收过程的时序关系,参见图 4 图 5。

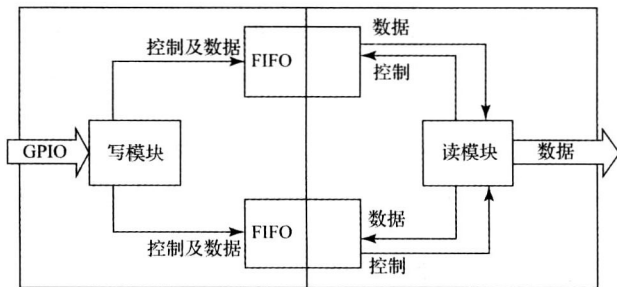


图 6 FPGA 数据接收发送示意

USB 通道控制参数:

输入数据位宽:14 bit

输入数据每一帧的个数:4 096

输出数据位宽:14 bit

主时钟频率:80 MHz

4.2 Cy7c67300 独立工作模式下 PC 软件

PC 应用软件是使用 Visual C++ 编写的。当将带有 Cy7c67300 的 USB 设备连接到 PC 后,软件能够识别出接口,会显示已连接的设备端口名称。

此软件可通过 USB 接口向连接设备发送数据,数据的发送个数可以调节,最大为 2 048 个。用于验证接口是否能够发送接收数据。软件可支持发送 4 种数据类型,分别为常数(Constant Byte)、随机数(Random Byte)、渐变数(Incrementing Byte)和渐变双字数(Incrementing DWORD)。

5 系统验证

5.1 PC 与 USB 通信验证

首先,将固件程序烧录到 Cy7c67300 的 E²PROM 中,然后将设备与上位 PC 连接,将会识别出本设备端口。

通过 PC 应用程序可看到,设备的端口名称和接口地址,然后通过软件向 USB 发送数据进行验证,本次验证选

择每组数据发送 50 个渐变数。

然后通过 Bus Hound 软件对数据发送/接收情况进行验证,如图 7 所示,可以看到数据已经发送成功,并且 USB 控制器返回的数据也接收成功,数据个数,类型正常。

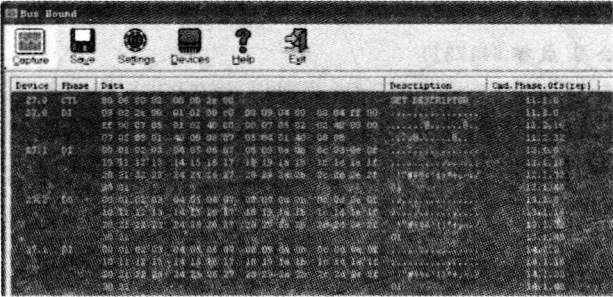


图 7 Bus Hound 数据捕捉示意

5.2 USB 控制器与 FPGA 通信验证

利用 ChipScope Pro Analyser 来捕捉数据发送时序,以验证 USB 控制器的数据是否成功发送到 FPGA。

图 8 所示,发送的是从零开始的渐变数,数据位信号显示捕捉到的是渐变数,而且数据开始发送时刻为 WE 上升沿,和设计的发送时序图吻合,证明通信正常。

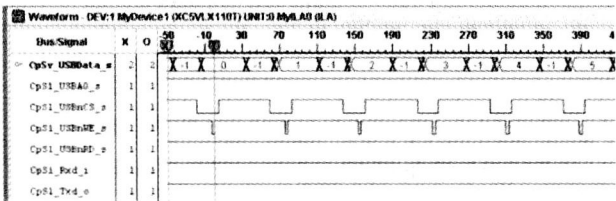


图 8 ChipScope Pro Analyser 抓取通信时序

FPGA 向 USB 控制器发送数据采用 Camera Link 接口实现。选择向上位 PC 发送一个渐变型的图像数据,右上图 9 所示,与发送图片类型相同,证明通信正常。

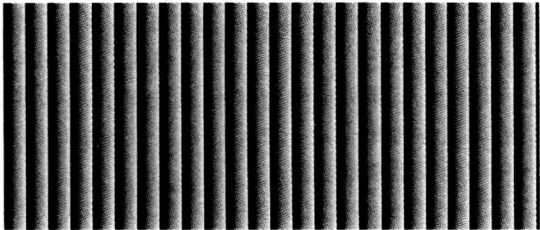


图 9 上位 PC 收到的图像数据

6 结 论

系统是使用 Cypress 的 CY7C67300 USB 控制器作为硬件芯片,在独立工作模式下,实现基于 USB 接口的 FPGA 与上位 PC 之间的高速通信。

经验证,USB 控制器可接收上位 PC 发送的数据,并将数据转发到上 FPGA 的功能;FPGA 可成功接收数据,并可发送数据到 USB 控制,并转发到上位 PC。

综上所述,实现了一个可通过 USB 接口实现上位 PC 与 FPGA 的通信传输系统。

参考文献

[1] 任晓东. CPLD/FPGA 高级应用开发指南[M]. 北京:电子工业出版社,2003.

[2] 王超,胡鑫,王化祥,等. 基于 CY7C68013 的嵌入式 RS232-USB 转换器设计[J]. 电子测量技术, 2007,30(11):139-141.

[3] 王晓利,龙兵,李力. 基于 FPGA 的 USB 数 I/O 设备设计[J]. 电子测量技术,2011,34(10):57-68.

[4] 徐亚东,蒋建国,齐美彬. 基于 USB 总线的图像采集与处理系统[J]. 国外电子测量技术,2007,26(6):45-47.

[5] PETER J. Ashenden, the Designer's Guide to VHDL[M]. 北京:机械工业出版社,2005.

[6] 杜勇. FPGA/VHDL 设计入门与进阶[M]. 北京:机械工业出版社,2011.

[7] 余祖俊,杜伟智,史红梅. 基于 USB 的通信信号采集分析系统[J]. 电子测量与仪器学报. 2005,19(5):85-89.

[8] 斐向东,陈萧,谭秋林,等. 基于 USB-FIFO 的 FPGA 与上位机通信的设计与实现[J]. 计算机测量与控制, 2012,20(4):1073-1075.

[9] 薛园园. USB 应用开发实例讲解[M]. 北京:人民邮电出版社,2009.

[10] 李英伟. USB 2.0 原理与工程开发[M]. 北京:国防工业出版社,2007.

[11] 吴磊,郭超平,申世涛. 基于 CY7C68013 与 FPGA 的便携式数据采集系统[J]. 计算机应用. 2012,32(21):164-166.

作者简介

黄义,1989 年出生,助理工程师,大学本科,主要研究方向为视频电路软件程序设计与测试。

翟国芳,1984 年出生,助理工程师,硕士研究生,主要研究方向为视频电路软件程序设计与测试。

万旻,1978 年出生,高级工程师,硕士研究生,主要研究方向为视频电路软硬件设计,信号处理电路软硬件设计等。

成桂梅,1972 年出生,高级工程师,硕士研究生,主要研究方向为视频电路软硬件设计,信号处理电路软硬件设计等。