

基于 USB 2.0 Slave FIFO 模式下软件开发框架

袁 卫^{1,2}

(1 渭南师范学院 信息与教育科学系, 陕西 渭南 714000; 2 西安电子科技大学, 西安 710071)

摘 要: 以 Cypress 公司的基于 USB 2.0 的 EZ-USB FX2 系列芯片之一 CY7C68013 为例, 简要介绍了其内部结构, 并详细说明在 Slave FIFO 模式下实现批量数据传输中相应软件系统, 包括固件程序、Windows 驱动程序及应用程序的开发过程。

关键词: Slave FIFO 模式; 固件程序; 驱动程序; 应用程序

中图分类号: TP311 **文献标志码:** A **文章编号:** 1009—5128 (2007) 02—0057—02

收稿日期: 2006—03—29

基金项目: 渭南师范学院科研基金项目 (06YKS032)

作者简介: 袁卫 (1973—), 男, 陕西渭南人, 渭南师范学院信息与教育科学系讲师, 西安电子科技大学物理与电子专业硕士研究生。

通用串行总线 (Universal Serial Bus, 简称 USB) 作为外围设备与计算机之间的新型标准接口, 与传统的 I/O 连接模式相比具有可热插拔、即插即用、快速、双向和价格低廉等特性获得了越来越多用户的接受。但由于其晦涩的英文资料和其复杂的通信协议, 使广大的开发者望而却步, 本文就以 CYPRESS 公司的 EZ-USB FX2 系列芯片之一 CY7C68013 为例详细的介绍其内部结构组成以及在 Slave FIFO 模式下与 FPGA 之间的数据通信的软件开发过程。

1 USB 2.0 接口芯片 CY7C68013 的内部结构

CYPRESS 公司的 EZ-USB FX2 系列芯片之一 CY7C68013 是最早符合 USB 2.0 协议的微控制器之一。它集成有 1 个增强型的 8051 处理器 (称为 CPU 核), 其和标准的 8051 的指令完全兼容, 其主频最高可设为 48MHz, 而性能是标准 8051 的 5~10 倍; 1 个串行接口引擎 (SIE) 和 1 个 USB 收发器 (统称 USB 核), 在高速模式下, USB 核可通过执行 USB 本身的协议来完成数据传输, 其数据的传输以数据包为单位, 而不需要 CPU 的参与, 克服了 CPU 传送数据的“带宽瓶颈”, 并简化 8051 代码的编写; 8.5kB 片上 RAM, 主要完成存储固件程序; 7 个端点缓冲区, 包括 3 个 64 字节的端点 0、端点 1 IN、端点 1 OUT 和四个 (EP2、EP4、EP6、EP8) 可配置不同大小缓冲的 IN 或 OUT 端点。数据传输有全速传输 (12Mbps) 和高速传输 (480Mbps), 并具有 USB 协议所规定的 4 种传输方式, 即控制方式、中断方式、批量传输、和同步传输方式。

CY7C68013 与外部设备有 3 种不同的接口模式: Ports 模式、Slave FIFO 模式和 GPIF 模式。Ports 模式下, 其数据的传输主要在 CPU 核的参与下完成, 数据的传输是通过执行指令实现, 因此数据的传输率比较低。对大批量数据传输一般采用后两种方式, Slave FIFO 方式是从机方式, 外部控制器 (如 FPGA 或单片机) 可像普通 FIFO 操作一样对 FX2 的多层缓冲 FIFO (由 EP2、EP2、EP4、EP6、EP8 组成) 进行读写, 而不考虑该包的大小, 传输速率可明显提高; GPIF 方式, 称通用可编程接口方式, 是主机方式, EZ-USB FX2 可由软件编程输出读写控制波形, 除了可对 FX2 的四个 FIFO 端点进行读写外, 还几乎可以对任何通用总线接口进行访问, 如 ASIC、DSP 和存储器等, 使用非常灵活。下面我们将以常用的 Slave FIFO 方式为例, 详细介绍 USB 的软件开发过程。

2 系统软件的开发

USB 系统软件主要包括设备底层固件程序、USB 设备驱动程序和应用程序 3 个部分。

2.1 底层固件设计

底层固件程序开发有两种方式, (1) 对于熟悉 8051 汇编语言的用户来说, 可以直接利用汇编语言编写结构紧凑、高效的固件程序; (2) 还可以利用现成的固件程序框架函数, 根据设备的要求添加相应的用

户函数,完成所要求的功能。两种方法相比较,第一种方法,对开发人员要求必须熟悉汇编语言及系统硬件结构;对于第二种方法,CYPRESS公司为开发者提供了固件程序框架来实现主机与外设之间的通信,用户只需在这些框架函数里添加自己的程序代码就可以实现特定的功能。不仅可以简化了程序的编写,也缩减了开发周期。如下将以一个 CY7C68013和 FPGA在 Slave FIFO方式下通信的应用实例说明如何在开发者提供的框架程序中实现特定功能。其框架函数主要是初始化函数 ID_Init()和功能函数 ID_Poll(),前一个函数主要完成系统的初始化,如主频设置、端点设置、数据传输方式等等;后一个函数完成系统功能,包括对端点的控制操作等。

```
Void ID_Init ( void ) //初始化函数,
```

```
{ CPUCS = 0x10; //工作频率设为 48MHz
  IFCONFIG = 0xC3; //设置 FX2的工作模式为 Slave FIFO模式
  EP2FIFOCFG = 0x6D; //端点 2输出,配置 16位数据线
  SYNCDELAY;
  EP2CFG = 0xA2; //端点 2输出,批量传输,双缓冲
  SYNCDELAY;
  ... ..
}
```

```
Void ID_Poll ( void )
```

```
{ //此函数在设备运行时反复被调用,用户任务函数可写在此函数内,由于读 FIFO的功能由 FPGA完成,此处可不填写代码 }
```

2.2 驱动程序设计

该系统需要两个驱动程序,即通用驱动程序和下载固件程序的驱动。通用驱动用于完成外设和用户之间的通信及控制;而下载固件程序的驱动则负责在连接 USB 总线后把特定的固件程序下载到 FX2的 RAM中,并由增强型 8051,即 CPU核执行,同时模拟断开与 USB 总线的连接以完成对外设的重新设置。通用驱动程序一般不需要重新编写,可以使用 Cypress公司已经编好的驱动 ezusb.sys即可。由于在 Windows 2000以上的操作系统中已经新增了媒体存储设备的驱动程序,并可使用批量传输方式功能,所以直接选择 bulkusb.sys驱动程序。而下载固件驱动则必须定做,其定做过程在文献 [2]和一些驱动开发网中已有详细的介绍,在此就不在详述。

2.3 应用程序设计

应用程序是系统与用户的接口,其通过通用驱动程序来完成对外设的控制和通信。在编写用户程序时,首先要建立与外设的连接,然后才能实施数据的传输。本设计是通过调用 Visual C 6.0编译环境中的 API函数访问设备驱动程序的。首先,查找设备,打开设备的句柄,然后进行读写和控制操作,最后关闭设备的句柄。程序中主要用到的两个 API函数,分别是 CreateFile()和 DeviceIoControl(),数据的传输就是通过这两个函数来实现的。程序实例如下:

```
Handle DeviceHandle; //设备句柄
//获得 USB 设备句柄
DeviceHandle = CreateFile( "\\.\EZUSB0; //打开 USB 设备
                          GENERIC_WRITE, //设置访问方式
                          FILE_SHARE_WRITE, //设置为共享方式
                          NULL,
                          OPEN_EXISTING, //设置创建方式
                          0,
                          NULL);

//关闭设备
CloseHandle(DeviceHandle);
//USB 设备的 I/O操作
DeviceIoControl (DeviceHandle;
                 IOCTL_EZUSB_BULK_READ; //读 USB 设备
                 LPVOID lpInBuffer; //输入数据缓冲区指针
```

(下转第 68 页)

越来越多地出现在校园网的信息化解决方案中,这就要求内网安全管理实现从单一系统到异构平台的过渡,从而避免了由异构平台的不可管理引起的安全盲点的出现。

3 结语

网络安全技术要通过创新安全理论、整合各种安全解决方案、整合网络安全资源,构建综合的动态网络来实现安全防护的效果。在理论上,通过对各种安全理论的整合,构建全新的网络安全理论;在方案上,整合网络安全产品,构建智能型的立体防护体系;在技术上,整合网络安全融合技术,构建全面动态的安全防护体系,从而满足不同领域网络安全多系统集成问题的需要。

最后,安全技术和安全管理不可分割,它们必须同步推进。因为即便有了好的安全设备和系统,如果没有好的安全管理方法并贯彻实施,那么安全也是空谈。安全管理的目的在于两点:一是最大程度地保护网络,使其安全地运行;二是一旦发生黑客事件后能最大程度地挽回损失。

参考文献:

- [1] 郭牧. 网络安全技术发展趋势分析 [J]. 通信世界周刊, 2005, 10(17): 54.
- [2] 黄声烈, 陈思, 国黄毅. 网络安全融合技术 [J]. 现代情报, 2005, (8).
- [3] 谢希仁. 计算机网络 [M]. 大连: 大连理工大学出版社, 2004.

[责任编辑 牛怀岗]

Current Situation of Campus Network Security and Its Development Trend

ZHANG Guo-jun

(Department of Computer Science, Weinan Teachers University, Weinan 714000, China)

Abstract: The current situation of campus network security and its limitation are summarized, especially expounding technology of firewall, intrusion detection system and anti-virus

Key words: network security; UTM; MS; SCM

(上接第 58 页)

```

Sizeof (BULK_TRANSFER_CONTROL); // 输入数据缓冲区大小
LPVOID lpOutBuffer; // 输出数据缓冲区指针
DWORD nOutBufferSize; // 输出数据缓冲区大小
LPDWORD lpByteReturned; // 接收输出数据字节计数器变量的指针
NULL);

```

总之,编写应用程序访问设备驱动程序分三步:首先用 CreateFile 取得设备句柄,然后用 DeviceIoControl 与设备进行读写操作,最后利用 CloseHandle 函数关闭设备句柄。

3 结论

基于 USB 2.0 接口的数据传输系统设计的软件开发过程包括应用软件开发,固件程序开发和驱动程序开发三部分,以上以 Slave FIFO 方式举例说明了其软件的开发过程,对 USB 2.0 程序开发有一定的实践作用。

参考文献:

- [1] 王成儒, 李英伟. USB 2.0 原理与工程开发 [M]. 北京: 国防工业出版社, 2004.
- [2] 边海龙, 贾少华. USB 2.0 设备的设计与开发 [M]. 北京: 人民邮电出版社, 2004.
- [3] 徐爱钧, 彭秀华. 单片机高级语言 C51 应用程序设计 [M]. 北京: 电子工业出版社, 1998.

[责任编辑 牛怀岗]

Development of Software Framework Based on USB 2.0 Slave FIFO method

YUAN Wei^{1, 2}

(1 Department of Information and Education Science, Weinan Teachers University, Weinan 714000, China;

2 Xidian University, Xi'an 710071, China)

Abstract: A brief introduction of the basic structure of EZ-USB FX2 CY7C68013 is provided. The development method of firmware, driver and application in Slave FIFO method to realize bulk data transfers

Key words: Slave FIFO; firmware; driver; application