

ELECTRONIC
ENGINEER

Universal Serial Bus Interface Design

USB 接口设计

李 立 / 编著

Universal Serial Bus Interface Design



北京电子科技出版社

Beijing University of Posts and Telecommunications Press

引言



什么是 USB?

USB 就是通用串行总线 (Universal Serial Bus) 的英文缩写。

在本书中,我们用通用术语“总线”来描述与设备进行电气连接的硬件。这是一个广义的定义,它不仅包括 PCI 总线,还包括 SCSI 卡、并行口、串行口、USB 集线器等等。实际上,它可以是任何能插入多个设备的硬件设备。

USB 总线和 RS232 类似,也是一种串行外围设备连接的总线形式。所不同的是,USB 的传输速度是标准串口的数十倍,而且可以支持多达 127 个外设同时级联。USB 总线提出的目的就是企图把当前 PC 固有的串口、并口、PS2 (键盘鼠标接口), 甚至是 SCSI 等连接外部设备的总线接口统一为单一的 USB 总线接口。

USB 总线接口作为外设的通用接口,有上述其他接口无法比拟的优势。首先,它的传输速度快,USB 1.1 总线规范规定的最大传输速度为 12 Mb/s,比一般的 PC 外置端口都快。而在 USB2.0 总线规范里,定义的最高传输速度已经达到了 480 Mb/s,可以满足包括视频设备在内的多种外部设备数据传输的需要。其次,它支持热插拔,而且是真正的即插即用设备。无论在 PC 开机工作或不开机的情况下,都能任意插拔,而不会对硬件产生任何损坏。再次,它可以支持双向、同步传输,能够保证某些需要连续数据流的设备的需要,譬如音频设备等。还有,它是一种廉价的而有高性价比的接口,硬件的结构相对简单,外围设备接口设计容易。此外,USB 总线可以提供最多 500 mA 的电源,对于一般的小型外设可以通过总线供电,不再需要外部电源。

USB 总线接口还有一个优势:目前主流的操作系统和绝大多数的 PC 都支持 USB 总线。微软 Windows 98 以后的版本都对 USB 总线有良好的支持,大多数的 USB 总线设备也有了相应的驱动程序支持。每台 PC 机一般都有 1~4 个不等的 USB 接口,一般的台式机大多有 2 个。这说明,你不需要为 USB 总线设备再增加任何有关 USB 总线插卡或根集线器的开销,主板的生产商都已经帮你集成了。



你的外围设备需要 USB 吗?

USB 总线的成功推出,的确对外设的接口产生了广泛的影响,USB 总线设备如键盘、鼠标、外置存储器、摄像头、显示器、扫描仪等产品在市场上都已经普遍存在。

那么你开发的外设需要采用 USB 总线接口吗?

这要从两个方面看:

- 你的计算机系统是否支持 USB ?

在 Windows 95.950B 版本之前,微软公司的操作系统都不支持 USB 总线接口,到了 Windows 98 推出后才有比较好的支持度。与以往的 VXD (虚拟设备驱动程序)模式不同,在 Windows 98 及以上的版本中,一般的 USB 驱动程序都是 WDM (Win32 Driver Model,视窗驱动模型)方式的。所以在 Windows 98、Windows Me、Windows 2000 中的 USB 总线设备驱动程序基本都可以兼容,但这种方式不支持 Windows NT 操作系统。因此,需要确定的是设备客户端程序是否支持上述操作系统下的 WDM 式驱动程序。但如果你的客户端程序是 VC/VB/C 开发的 Win32 程序的话,那么需要修改的源程序就不会很多。

最近的 LINUX、FREEBSD 等操作系统也开始支持 USB 总线设备了,具体的限制需要通过了解它们的说明才能确定。

另外,你的计算机硬件是否有 USB 端口也值得关注。虽然现在的 PC 绝大多数都有 USB 总线接口了,但少数笔记本电脑却不提供 USB 总线供电,你的外设必须有另外的供电设备。如果你的计算机没有 USB 总线接口,那么还要考虑添加 USB 总线适配卡的成本,以及 USB 总线适配卡所占用的计算机资源是否可以得到满足。

- USB 能否满足你的设备要求?

USB 总线主要的特点之一就是传输速率快。在理想情况下,USB 1.1 的峰值传输速度能达到 12 Mb/s,但是在传输时往往未必能时刻以峰值速度传输,所以必须考虑实际的传输速度能否满足设备的要求。当然,USB 2.0 可以支持的传输速度高达 480 Mb/s,可以满足当前绝大多数外设的要求,目前支持 USB 2.0 的主机逐渐增多。

另外,如果你的设备必须使用同步实时传输的话,由于 USB 总线对同步传输的错误不能纠正,因而还需要外设对偶尔出错能有一定的容忍度。

当你的设备连接到 USB 总线上时,还要考虑总线上其他设备对它的影响。如果没有其他的 USB 总线设备,当然是最理想的情况了,但是一旦有其他外设,必须要考虑这个因素是否会对你的设备产生不利影响。

如果以上的问题对你开发的设备都不成问题,那么恭喜你,现在就可以从本书开始你的 USB 总线接口设计了!



这本书讲什么?

这本书的目的就是能够使你很快地进入到 USB 总线的设计流程中去,提供的都是和实际动手设计有关的知识内容,配合固件代码的实例,使你能很快地开始并完成你的设计,所需时间起码比我们在开始学习 USB 总线时的摸索时间加上走弯路所花费的时间少得多。

本书的第 1 章对 USB 总线的发展、现状以及相关的问题作简单的介绍。

第 2 章简述 USB 总线规范的重点内容,精练叙述 USB 总线的重要概念以及与实际制作有关的重点内容。这样的介绍使重点突出,可以加速学习的过程。

第 3 章对本书所使用的范例芯片 EZ-USB 系列作细致的描述,这个系列的芯片非常适合初学 USB 总线接口设计的开发人士使用,因为它集成了很多底层的 USB 总线操作,有时甚至可以轻易地通过简单的编程来完成 USB 总线的传输,而不必知道 USB 总线传输的具体

实现过程。另外，EZ-USB 系列芯片提供的范例和通用驱动程序会节省很多开发时间。

第 4 章的内容讲述有关 WDM 驱动的开发信息。这些信息不仅对开发 USB 总线设备的驱动程序有用，而且对于开发其他 WDM 类型驱动程序也很有帮助。

第 5 章详细说明一个 USB 总线的驱动是如何生成的以及怎样加载这样的驱动。并且，这里还介绍 EZ-USB 的 USB 总线通用驱动程序的详细说明。

第 6 章给读者讲述如何开发固件程序 (Firmware)。好的固件程序能够使 USB 总线接口更加有效率。

第 7 章介绍固件程序开发工具的使用以及固件程序的源代码分析，还有客户端程序开发的实例，让你快速上手。本章还有一个 USB 总线音频采集设备的实例分析，通过这个实例的介绍，读者能够进一步细化对 USB 总线设备开发的认识。

总之，本书完全从实际设计的角度出发，针对设计中的关键问题逐一分析介绍。这也是本书名称的由来。通过本书的学习以及你的聪明与实干，那么开发一个成功的 USB 总线接口的效率就会大大提高。



更多的网上资料

进行 USB 总线接口开发，需要经常访问的站点是

<http://www.usb.org>

这个站点是 USB 的官方站点，提供关于 USB 总线的很多信息。可以从这个站点免费下载 USB 总线的最新规范和其他相关文档。而且这个站点还提供了一些与 USB 总线相关的测试工具软件和关于 USB 总线接口开发的论坛，从中可以学到很多经验。

学习开发 USB 总线的 Windows 驱动程序时，可以从微软的站点

<http://www.microsoft.com/hwdev/driver>

<http://msdn.microsoft.com/developer>

中得到有关的资料。微软的 MSDN 有经常更新的开发资料，从中可以找到有关 USB 总线的最新驱动程序的范例和说明。

本书采用的范例芯片是 EZ-USB 2131 芯片，更多有关这个芯片的资料从

<http://www.cypress.com/usb>

中可以找到，当然这里面还有 Cypress 公司其他系列 USB 总线接口芯片的资料以及开发指南等。

当然，在任何一个网络搜索引擎上，都可以找到若干与 USB 总线相关的站点和开发论坛，这些对学习 USB 总线或多或少都会有帮助。



致谢与声明

本书在编写过程中得到了很多人的帮助，没有他们的无私指教，也就没有这本书的出版。在这里，谨向对本书有过帮助的所有人表示感谢。特别需要感谢的是西安电子科技大学

学出版社的臧延新编辑和其他编审人员，感谢他们对本书出版所给予的热情支持和帮助。本书的所有内容都经过了他们斟词酌句的审校。

由于作者在 USB 总线接口的设计方面所作的工作并不能够包括有关 USB 总线接口设计的所有方面，在学习和理解 USB 总线协议的过程中也难免挂一漏万，有些深入的问题还有待进一步的认识，所以希望读者朋友能够针对书中的不足给予指正和谅解。

在本书中提到的 Windows、Windows 95、Windows 98、Windows 2000、Windows NT 是微软公司的注册商标。EZ-USB 是 Cypress 半导体公司的注册商标，IBM 是国际商用机器公司的注册商标，Philips 是飞利浦公司的注册商标，其他提到的注册商标所有权也归各自的公司所有，本书仅仅是引用其名称。相关的产品资料请到各自公司的网站查询。

编著者
2002 年 8 月

目 录

第 1 章 USB 简介.....	1
1.1 USB 出现的动因.....	1
1.2 适用的对象和目标.....	2
1.3 设计原则和特性.....	2
1.4 USB 性能.....	4
1.4.1 不同接口传输速率对比.....	4
1.4.2 USB 的传输方式.....	4
1.4.3 USB 接口设备供电.....	5
1.4.4 USB 接口连接距离.....	5
1.5 USB 2.0.....	5
1.6 全速 USB 总线接口控制器芯片简介.....	6
第 2 章 USB 1.1 规范概述.....	8
2.1 USB 总线体系结构.....	8
2.2 电气连接特性.....	9
2.2.1 USB 电缆.....	9
2.2.2 USB 传输信号状态.....	10
2.2.3 USB 电源管理.....	11
2.3 数据的编码和解码.....	12
2.4 USB 总线传输协议.....	12
2.4.1 包的格式.....	13
2.4.2 数据字段的格式.....	15
2.4.3 数据交换的格式.....	18
2.5 USB 总线通道.....	19
2.5.1 外设端点.....	20
2.5.2 管道.....	21
2.6 USB 总线的数据传输类型.....	22
2.6.1 控制传输.....	23
2.6.2 中断传输.....	28
2.6.3 批传输.....	31
2.6.4 同步传输.....	34
2.7 USB 主机的硬件与软件.....	37
2.7.1 主机控制器功能.....	38
2.7.2 软件功能.....	39
2.7.3 主机控制器驱动器(HCD).....	42

2.7.4	USB 总线驱动程序(USBD)	42
2.8	USB 总线设备	49
2.8.1	USB 总线设备状态	50
2.8.2	通用 USB 总线设备操作	52
2.8.3	USB 总线设备请求	54
2.8.4	标准设备请求	56
2.8.5	描述表	61
2.8.6	标准描述表的定义	62
2.8.7	设备类定义	67
第 3 章	EZ-USB 系列接口控制芯片	68
3.1	EZ-USB 的特性	68
3.2	EZ-USB 芯片结构	68
3.3	EZ-USB 传输类型与控制	69
3.3.1	批传输	70
3.3.2	中断传输	73
3.3.3	同步传输	73
3.3.4	控制传输	74
3.3.5	数据切换控制	75
3.3.6	批传输端点中断控制	76
3.3.7	自动指针控制	77
3.4	EZ-USB 微处理器	77
3.5	EZ-USB 端点	78
3.5.1	批传输端点	78
3.5.2	EZ-USB 的控制端点	78
3.5.3	EZ-USB 的中断端点	79
3.5.4	EZ-USB 的同步端点	79
3.6	EZ-USB 的存储空间	79
3.6.1	8051 存储器	79
3.6.2	EZ-USB 扩展存储器	81
3.6.3	EZ-USB 的 CS#和 OE#信号	82
3.6.4	EZ-USB 的 ROM	83
3.7	EZ-USB 的输入和输出	84
3.7.1	介绍	84
3.7.2	I/O 端口	84
3.7.3	I ² C 总线控制器	86
3.7.4	发送 I ² C 数据	86
3.7.5	接收 I ² C 数据	86
3.7.6	I ² C 自举启动	87
3.8	EZ-USB 的枚举和再次枚举	88

3.8.1	介绍	88
3.8.2	默认的 USB 总线设备	89
3.8.3	EZ-USB 内核对 EP0 设备请求的响应	89
3.8.4	固件装载	91
3.8.5	枚举模式	92
3.8.6	没有串行 EEPROM 的情况	93
3.8.7	串行 EEPROM 存在且首字节为 0xB0	93
3.8.8	串行 EEPROM 存在且首字节为 0xB2	94
3.8.9	再次枚举	95
3.8.10	多重再次枚举	95
3.8.11	缺省的描述符	95
3.9	EZ-USB 的快速传输模式	104
3.10	EZ-USB 中断	104
3.11	EZ-USB 的复位和电源管理	105
3.12	EZ-USB 寄存器	105
3.12.1	介绍	105
3.12.2	批数据缓冲区	106
3.12.3	同步数据 FIFO	107
3.12.4	同步字节数寄存器	108
3.12.5	CPU 寄存器	110
3.12.6	端口设置	110
3.12.7	输入/输出端口寄存器	112
3.12.8	230 kbaud UART 操作	114
3.12.9	同步控制/状态标志寄存器	114
3.12.10	I ² C 总线寄存器	115
3.12.11	中断	117
3.12.12	端点 0 控制和状态寄存器	122
3.12.13	端点 1 ~ 7 的控制和状态寄存器	124
3.12.14	全局 USB 寄存器	127
3.12.15	快速传输	132
3.12.16	初始化数据	134
3.12.17	同步 FIFO 大小	134
3.13	EZ-USB 产品序列及特点	135
第 4 章	Windows WDM 驱动程序基本结构	136
4.1	设备和驱动程序的层次结构	136
4.1.1	设备的递归枚举	138
4.1.2	设备对象之间的关联	138
4.1.3	WDM 驱动程序对象	139
4.1.4	WDM 设备对象	141

4.1.5	建立设备堆栈.....	145
4.2	DriverEntry 例程.....	145
4.2.1	DriverEntry 概述.....	146
4.2.2	DriverUnload 例程.....	147
4.2.3	驱动程序再次初始化例程.....	148
4.3	AddDevice 例程.....	148
4.3.1	创建设备对象.....	148
4.3.2	为设备命名.....	150
4.3.3	设备接口.....	152
4.3.4	其他全局性的设备初始化操作.....	155
第 5 章	USB 设备驱动程序设计.....	160
5.1	Windows USB 驱动程序接口.....	161
5.1.1	USB 总线软件体系结构.....	162
5.1.2	USB D.....	162
5.1.3	UHCI 和 OpenHCI.....	163
5.1.4	根集线器驱动程序.....	163
5.1.5	USB 总线设备客户软件.....	163
5.1.6	USB 总线设备驱动程序装载.....	164
5.2	USB D 的 IOCTL.....	164
5.3	USB D 结构定义.....	165
5.4	URB 的定义.....	169
5.5	开发设备驱动程序.....	171
5.5.1	设计工具包 DDK 的安装.....	171
5.5.2	设备驱动程序的设计.....	174
5.5.3	利用 DriverStudio 开发 USB 驱动程序.....	176
5.6	EZ-USB 通用驱动程序说明.....	184
5.6.1	引言.....	184
5.6.2	建立 EZ-USB GPD.....	185
5.6.3	装载 EZ-USB GPD.....	185
5.6.4	用户态和 GPD 的接口.....	186
5.6.5	I/O 控制程序代码参考.....	187
第 6 章	EZ-USB 固件程序开发.....	188
6.1	固件基础.....	188
6.1.1	硬件设备内容.....	188
6.1.2	固件与报表.....	189
6.1.3	识别一个设备为 HID.....	189
6.1.4	HID 特定请求.....	190
6.1.5	数据传输.....	191
6.1.6	固件的执行.....	191

6.2	EZ-USB 固件.....	192
6.3	EZ-USB 固件程序的结构.....	194
6.3.1	EZ-USB 固件程序框架简介.....	194
6.3.2	固件程序框架的建立.....	196
6.3.3	EZ-USB 固件程序框架函数.....	196
6.4	EZ-USB 固件全局变量.....	204
6.5	EZ-USB 设备描述表.....	205
6.5.1	描述符.....	205
6.5.2	设备描述符(Device Descriptor).....	206
6.5.3	EZ-USB 描述表.....	206
6.6	EZ-USB 固件库.....	209
6.7	用汇编语言开发固件代码.....	210
第 7 章	EZ-USB 设备开发实例.....	215
7.1	开发工具包.....	215
7.2	批传输实例.....	215
7.2.1	固件代码.....	216
7.2.2	固件代码的调试.....	230
7.2.3	固件代码的下载.....	234
7.2.4	客户端程序.....	235
7.3	USB 设备的开发.....	239
7.3.1	简介.....	239
7.3.2	USB 总线音频采集设备.....	239
7.3.3	软件的实现.....	241
附录 A	USB 音频设备固件程序源代码.....	243
附录 B	EZ-USB 寄存器.....	261
附录 C	EZ-USB 2131Q 接口连接原理图.....	269
附录 D	词汇表.....	270

第 1 章 USB 简介

通用串行总线 USB 的规范是 IBM、Compaq、Intel、Microsoft、NEC 等多家公司联合制订的。USB 总线规范草案最早提出于 1994 年底，几经修订，推出了版本号为 0.7 的正式版本，接着于 1996 年推出了 USB 1.0 的正式版本，到了 1998 年，又推出了修订的版本 USB 1.1。在 USB 1.1 版本中定义了两种速度 USB 的传输工作模式，低速(Low Speed)模式的数据传输速度为 1.5 Mb/s；全速(Full Speed)模式使得 USB 的传输速度峰值达到了 12 Mb/s，这一版本得到了计算机业界的广泛响应，到 1999 年，USB 已经被广泛应用。受到这一潮流的鼓舞，2000 年底，USB 组织又推出了新版本的规范——USB 2.0，这个版本将 USB 总线的传输速度提高到了 480 Mb/s 的水平，支持它的芯片称为高速(High Speed)系列。

1.1 USB 出现的动因

随着电脑应用的拓展，PC 机的外设越来越多，在机箱的外部出现了各种各样的外设接口：并行口、串行口、键盘口、鼠标口、游戏手柄接口、音频输入输出口、视频输出口、网络接口等等。由于各种外部设备不断增加，计算机所提供用于连接外设的资源日渐捉襟见肘，时常因为不能提供更多的接口，而导致各种 I/O 的冲突，加之各种中低速外设缺少一个双向、低价、即插即用的统一的总线，限制了外部设备的开发。因此，简化外设扩充方法使之方便易行，便成为各个 PC 机厂家面临的重大研究课题。

在这个背景下，Microsoft 公司于 1994 年提出了即插即用(Plug and Play, PnP)技术方案。这种技术解决了用户为外设手工设置资源参数的困难，改由系统自动进行设置。但新外设的安装仍然显得相对麻烦，而且扩充外设数量的问题也没有解决。因此，在 1996 年召开的面向 PC 机硬件技术工作者会议上，Compaq、Intel 和 Microsoft 三家厂商提出了设备插架(Device Bay)的概念。基于这样的概念，Intel、Compaq、Microsoft、NEC、North Telecomm、IBM 等数家计算机业界的大厂商，开发出了适合中低速的外部设备使用的 USB 总线接口，它是一种典型的设备插架结构的总线。此后，随着 USB 总线规范的推广日益深入，至今，各种基于 USB 总线接口的外设已达千余种之多。USB 逐渐被几乎所有 PC 制造商所接受，目前绝大多数的 PC 都配置有 1~4 个标准 USB 总线接口，而且通过 USB 集线器(HUB)的扩充，可以支持多达 127 个外设同时连接。

在 USB 提出之前，PC 机主板上所能提供的外部接口比较少，一般仅有一个并口和两个串口，所能连接的设备十分有限，而且拔插设备的时候需要关机(带电拔插会造成烧毁设备的严重后果)，传输速度也很慢，在要求高传输率的场合根本无法满足要求。

而 USB 总线接口则很好地解决了以上所有问题。采用 USB 接口的设备无一例外地支持热拔插(带电拔插),使微软的“Plug and Play”得以真正实现,用户在开机状态时即可将设备连接到电脑主机上,免除了使用户感到厌烦的重新启动过程。USB 接口所能连接的设备多达 127 个,而且可以同时使用。速度方面,USB 1.1 总线规范定义了 12 Mb/s 的带宽,足以满足大多数诸如键盘、鼠标、Modem、游戏手柄以及摄像头等设备的要求。同时总线能够提供 500 mA 的电流,一些耗电量比较小的设备就不必使用外接电源了。

1.2 适用的对象和目标

USB 的对象最初瞄准的是诸如调制解调器、扫描仪、键盘、个人数字助理(PDA)、鼠标、操纵杆等这样的低速设备。通过不断的发展,USB 已能胜任更高速的数据传输,因此适用的范围也逐步扩大到了如外置大容量存储器、数码相机、视频系统等这样需要大量传输数据的外部设备当中去。目前,USB 2.0 的发展甚至到了可以支持实时的高分辨率视频传输的水平。随着 USB 传输数据速度的提高,传输实时性的增强,可以预见其用途将更加宽广。

表 1-1 按照 USB 可以达到的数据传输率对其进行了分类。大体上说,全速传输多用于同步设备,低速传输多用于交互设备。虽然 USB 设计的初衷是针对台式电脑而不是应用于可移动的环境下,但是目前的应用早已不局限于此。软件体系通过对各种主机控制器提供必要的支持,以保证未来能对 USB 有良好的可扩充性。

表 1-1 USB 传输速率的应用

类 别	应 用	特 性
低速 · 交互设备 · 10 ~ 20 kb/s	键盘、鼠标、游戏杆	低价格、热插拔、易用性
全速 · 电话、音频、压缩视频 · 500 kb/s ~ 12 Mb/s	ISDN、PBX、POTS	低价格、易用性、热插拔、限定带宽和延迟
高速 · 视频、磁盘 · 25 ~ 480 Mb/s	音视频处理、磁盘	高带宽、限定延迟、易用性

设计 USB 总线接口的目标是,可以在一个开放的体系下广泛地使用不同厂家所生产的设备。USB 总线规范改进了便携式商务电脑或家用电脑的原有的扩展外设的体系结构,从而为系统生产商和外设开发商提供了足够的空间来创造多功能的产品,开发广阔的市场,使他们不必因为害怕失去兼容性而在产品中被迫使用陈旧的接口。

1.3 设计原则和特性

USB1.1 的总线规范是对 PC 机现有的体系结构的扩充。USB 总线接口的设计主要遵循

以下几个原则：易于扩充多个外围设备；价格低，且支持 12Mb/s 的数据传输；充分支持声音音频和压缩视频等实时数据；协议灵活，能够同时进行异步和同步的数据传输；能兼容各种不同设备的技术，提供一个标准接口，广泛接纳各种设备；综合不同 PC 机的结构和体系特点，赋予 PC 机新的功能。

USB 的规范能针对不同的性能价格比要求提供多种选择，以满足不同的系统和部件不同功能的需要，其主要特性可归结为以下 8 点。

(1) 终端用户的易用性：

- ？ 为接缆和连接头提供了单一模型；
- ？ 电气特性与用户无关；
- ？ 自检外设，自动地进行设备驱动、设置；
- ？ 外设可以动态连接、动态重置。

(2) 广泛的适用性：

- ？ 适应不同设备，传输速率从几 kb/s 到十几 Mb/s；
- ？ 在同一线上支持同步、异步两种传输模式；
- ？ 支持对多个设备的同时操作；
- ？ 可同时操作 127 个物理设备；
- ？ 在主机和设备之间可以传输多个数据和信息流；
- ？ 支持多功能的设备；
- ？ 利用低层协议，提高了总线利用率。

(3) 同步传输带宽：

- ？ 确定的带宽和低延迟适合电话系统和音频的应用；
- ？ 同步工作可以利用整个总线带宽。

(4) 灵活性：

- ？ 直接发送一系列指定大小的数据包，允许对设备缓冲器大小进行选择；
- ？ 通过指定数据缓冲区大小和执行时间，支持各种数据传输率；
- ？ 通过协议对数据流进行缓冲处理。

(5) 健壮性：

- ？ 在协议中使用差错处理/差错恢复机制；
- ？ 完全实时热插拔；
- ？ 可以对有缺陷的设备进行鉴别。

(6) 与 PC 产业的一致性：

- ？ 协议的易实现性和完整性；
- ？ 与 PC 机的即插即用体系结构一致；
- ？ 与现存操作系统有良好衔接的接口。

(7) 性价比：

- ？ 以低廉的价格提供传输速率为 1.5 Mb/s 的子通道；
- ？ 将外设和主机硬件进行了最优化的集成；
- ？ 促进了低价格外设的发展；
- ？ 廉价的电缆和连接头；

? 运用商业技术降低成本。

(8) 可升级性：

? 体系结构的可升级性支持在一个系统中同时存在多个 USB 主机控制器。

1.4 USB 性能

1.4.1 不同接口传输速率对比

在 USB1.1 规范中，有全速和低速两种传输速率，全速方式的传输速率为 12 Mb/s，低速方式的传输速率为 1.5 Mb/s。正如大家所知道的那样，像键盘、鼠标、游戏手柄这样的设备，只要低速方式就可以满足它们的要求了；而 ZIP、扫描仪以及打印机等设备就需要用到全速方式。因此，USB 总线接口与其他类型的接口相比，其优越性就体现在这里，它的传输速度适合各种类型的外设。表 1-2 比较了常用的几种计算机接口的速度，从这个表中可以清楚地了解到它们的差异。

表 1-2 不同接口传输速率对比表

接口类型	理论最大传输速率/b/s
USB at low speed	1.5 Mb/s
USB at full speed	12 Mb/s
USB 2.0 at high speed	480 Mb/s
Serial port	115 kb/s
IrDA(红外接口)	115 kb/s
IEEE 1394a	400 Mb/s
SCSI	40 Mb/s
Fast SCSI	80 Mb/s
Ultra SCSI-3	160 Mb/s
I ² C	400 kb/s
IEEE 488 (GPIB)	8 Mb/s

USB 接口虽然不是最快的，但从它普及的速度来看，在未来一两年内，也许会逐渐替代串行口和并行口，甚至可能终结 IEEE 1394a 总线(也称火线，Firewire)接口，成为个人电脑上的标准配置。而且现在几乎所有主板上都内置了 1~4 个 USB 接口，用户不需要另外购买适配器，减少了成本，这也是 USB 能够流行起来的一个重要原因。

1.4.2 USB 的传输方式

针对设备对系统资源需求的不同，在 USB 规范中规定了四种不同的数据传输方式。

(1) 控制(Control)传输方式。该方式用来配置和控制主机到 USB 设备的数据传输方式和类型。设备控制指令、设备状态查询及确认命令均采用这种传输方式。当 USB 设备收到这些数据 and 命令后，将依据先进先出的原则处理到达的数据。

(2) 中断(Interrupt)传输方式。虽然该方式传送的数据量很小，但这些数据需要及时处理，以达到实时效果。此方式主要用在键盘、鼠标以及操纵杆等设备上。

(3) 同步(Isochronous)传输方式。该方式用来连接需要连续传输数据且对数据的正确性要求不高，而对时间极为敏感的外部设备，如麦克风、喇叭以及电话等。同步传输方式以固定的传输速率，连续不断地在主机与 USB 设备之间传输数据，在传送数据发生错误时，USB 并不处理这些错误，而是继续进行新数据的传送。

(4) 批(Bulk)传输方式。该方式用来传输要求正确无误的大批量的数据。通常打印机、扫描仪和数字相机以这种方式与主机连接。

1.4.3 USB 接口设备供电

USB 线缆中包含有两根电源线及两根数据线。部分设备，像读卡器、摄像头、游戏手柄等耗电比较少的设备可以直接通过 USB 接口供电。可通过 USB 接口供电的设备又分低电量模式和高电量模式，前者最大可获得 100 mA 的电流，后者最大可获得 500 mA 的电流。倘若设备需要更大的电流，那就只好通过外接电源来供电了。

1.4.4 USB 接口连接距离

USB 提供低速与全速两种数据传送速度规格。全速传送时，结点间连接距离为 5 m，连接使用 4 芯电缆(电源线 2 条，信号线 2 条)。USB 全速传输速率与标准的串行端口相比，大约快 100 倍；与标准的并行端口相比，也快近 10 倍。因此，USB 能支持高速的通信接口(例如综合业务网 ISDN 等)，使用户拥有足够的带宽供新的数字外设使用。

1.5 USB 2.0

本书的主要内容实际上是针对 USB 1.1 的，所详细介绍的器件及书中提到的例程都围绕着全速的 USB 总线连接。但是，随着 USB 的发展，更快速的 USB 2.0 已经发布了。实际上，这两个版本中大部分的内容都是相似或近似的，如果要从 USB 1.1 转到 USB 2.0，要特别注意新规范中那些改动的部分。

USB 2.0 发布的主要原因是由于 USB 1.1 所规定的最大传输速率只有 12 Mb/s，远低于 400 Mb/s 的 IEEE 1394a 总线。Apple 和 SONY 由于制定 IEEE 1394a 总线标准获得了可观的标准使用许可费用。因此，为了与 IEEE 1394a 竞争，以 Intel 为主的一些硬件厂商和开发商又联合制订了 USB 2.0 规范。USB 2.0 是一项开放的高速通信协议总线规范。如果 USB 2.0 能够像 USB 1.1 那样广泛应用，Intel 和其他硬件厂商就能节约大笔的许可费用，相应地，USB 2.0 也会更迅速地普及。

USB 2.0 总线规范将传输速率提升至 480 Mb/s，大约 60 MB/s，足以满足大多数外设的速率要求。USB 2.0 总线规范中的增强主机控制器接口(EHCI)定义了一个与 USB 1.1 相兼容

的架构,它可以用 USB 2.0 总线接口的驱动程序驱动 USB 1.1 总线接口设备。所有支持 USB 1.1 总线接口的设备都可以直接在 USB 2.0 总线接口上使用而不必担心兼容性的问题,而且像 USB 连接线、插头、插孔等附件也都可以直接使用。但是反之,除非厂商在设计之初就将其设计为 USB 1.1 和 2.0 通用的,否则 USB 2.0 的设备不可能在 USB 1.1 上正常工作。虽然 USB 2.0 的这种架构对设备更为敏感,但占用的 CPU 时间较少,并降低了软件的复杂性。

USB 2.0 总线规范特别定义的微帧结构,使得 USB 2.0 总线接口设备在小缓存的情况下依然能够以高速率传送数据。不过,如果外设并不支持如此高的速度的话,系统将还原到较早的 12 Mb/s(全速)或 1.5 Mb/s(低速)速度上来。

当前 PC 上的应用软件大部分还是为操作 USB 1.1 总线接口外设而设计的,这一点尚未得到及时的改变。因此 PC 系统软件必须要加入支持 USB 2.0 总线接口外设的能力,以便使性能优化,并能够监测、优化外设的配置。

USB 2.0 HUB 有了更高的帧传送率,这样才能保证主机与支持 USB 2.0 总线接口的高速外设之间有更通畅的数据流。但是,为了兼容过去已有的标准,数据传送率匹配的职能增加了 HUB 的复杂性以及在高速传送时需要的临时缓存。最简单的情况是 HUB 与支持 USB 2.0 总线接口的外设通信。这个时候 HUB 只是简单地转交上行和下行电缆传来的高速信号,和过去 USB 1.1 总线接口所做的非常相似。这时允许 USB 2.0 总线接口外设使用 USB 中大部分的带宽。但为了和 USB 1.1 总线接口的外设通信,USB 2.0 HUB 就必须有调整数据流速度的匹配能力。换句话说,HUB 将负责管理转换从主控制器出来的高速数据流到低速的 USB 1.1 总线接口设备。USB 2.0 HUB 的这个能力使得兼容 USB 1.1 总线的接口设备能够和支持 USB 2.0 总线接口的设备一道工作,但又不会消耗 USB 2.0 总线规范所提供的高带宽。

从现在的情况看,在 USB 同 IEEE 1394a 的这场较量中,USB 已经在市场中计算机外设应用上占到了先机。我们在市场上可以看到大量的 USB 设备:USB HUB、Modem、游戏手柄、电视盒、软驱、CD-ROM、CD-RW、硬盘,甚至声卡、音箱、手机充电器、手写板、网络卡、键盘、鼠标、摄像头等,显示器、扫描仪、打印机的供货量也逐渐多了起来。虽然这些设备中大都使用全速模式和低速模式,但可以预见在不久的将来,在所有高、中、低速外设中,USB 总线接口将成为极其重要的标准接口。

为了应付这样的局面,IEEE 1394a 也正积极开发第二代总线规范 IEEE 1394b,其传输速度将超过 3.2 Gb/s。在高速的接口标准之争中,也许是市场而不是技术本身将会成为这场竞争最后的裁决者。

1.6 全速 USB 总线接口控制器芯片简介

目前,市场上的 USB 总线接口控制器芯片有很多种,规模大的芯片厂家几乎都有自己的 USB 总线接口控制器产品,当然,不同厂家的芯片在规格和性能上也有不同之处。

表 1-3 列出了常见的支持全速 USB 总线接口控制器芯片之间的参数比较。

表 1-3 各种全速 USB 总线接口控制器芯片的比较

芯片 型号	EZ-USB 2131	8X931A	USS825	USBN9602	NET2888	PDI USB12
厂商	Cypress	Intel	Luccent	Panasonic	NetChip	Philips
电源	3.3 V	5 V	3.3 V	5 V	3.3 V	3.3 V
端点数	32	3	15	7	6	7
包中最大字节数	1024	1024	1024	64	64	128
双缓冲	有	无	有	无	无	有
微控制器接口	异步串行 I ² C	异步串行 键盘口	并行	并行 Microwire	并行	并行 I ² C
引脚数	44/80	68	44/48	28	48	28
内置微控制器	8051	8051	需外接 微控制器	需外接 微控制器	需外接 微控制器	需外接 微控制器
程序存储器容量 /Byte	8k	8k	-	-	-	-
外部存储器容量 /Byte	64k	256k	-	-	-	-
通用 I/ O 个	24	32	-	-	-	-

其中 EZ-USB 2131 系列的 USB 总线接口控制器由于本身内置有 8051 兼容的微控制器，而且有较方便的开发流程，其固件程序易升级、易调试，所以被选作本书进行 USB 外设开发的范例芯片。实际上，只要掌握一种芯片开发 USB 设备的方法，使用其他的控制器芯片开发 USB 设备完全可以触类旁通，因为各种控制芯片都符合标准的 USB 总线规范，而且在结构上也大同小异。

无论使用哪种 USB 总线接口控制器芯片，只要实现的设备是符合 HID(Host Interface Device, 主机接口设备)类的硬件，都可以直接使用 Windows 操作系统的 HID 驱动，无需从头开始编写底层的驱动程序。而 EZ-USB 更有其特有的通用 USB 驱动程序配合其枚举和重枚举的需要。除了这个通用驱动程序，EZ-USB 还附带有大量的源代码和例子，使得开发的速度进一步加快。