

基于 Wave Panel 的 HID 设备通讯接口设计

崔 阳

(中国劳动关系学院, 北京 100048)

摘 要: HID 设备研发和使用中所面临的重要问题之一是设备与主机应用程序的通讯。文中以视频编辑专用 HID 设备 Wave Panel 为基础, 首先简要介绍了 HID 设备的性能特点和 Windows 环境下的 HID 通讯机制, 在此基础上设计出一个实用的 HID 设备通讯接口, 该通讯接口利用相关 Windows API 函数实现对 HID 设备的识别、打开和关闭, 并以异步模式完成设备与主机应用程序的双向通讯过程, 最后在 Wave Panel 对该接口进行了编程实验。实验结果证明该接口的通用性较好, 能够较好地实现 HID 设备与主机应用程序的通讯功能。

关键词: HID 设备; 通讯接口; 中断传输; Windows API

中图分类号: TP393.09

文献标识码: A

文章编号: 1673-629X(2013)05-0138-04

doi: 10.3969/j.issn.1673-629X.2013.05.036

Design of HID Communication Interface Based on Wave Panel

CUI Yang

(China Institute of Industrial Relations, Beijing 100048, China)

Abstract: Communication between Human Interface Devices (HID) and the computer applications is one of the important problems in HID development and utilization. Based on Wave Panel of a special video editing HID, characteristics and the communication mechanism of HID in Windows environment are introduced at first, then at the basis, a practical communication interface is designed. Some related Windows API are utilized by the interface to identify, open and close HID, and the two-way communication between HID and the computer applications is achieved in asynchronous mode. The experimental results on Wave Panel show that the interface has good generality and communication function can be better accomplished through it.

Key words: HID; communication interface; interrupt transmission; Windows API

0 引 言

HID (Human Interface Device, 人机接口设备) 是一类低速 USB 设备, 典型的 HID 设备有键盘、鼠标、游戏杆等。相对于 U 盘、打印机等高速 USB 设备, HID 设备具有使用方便、响应迅速、研发成本低廉等优点, 其应用领域十分广泛。当前 HID 设备正在向专用化方向快速发展^[1]。

Wave Panel 是苹果公司研发的视频色彩编辑软件 Color 支持一型专用的数据输入和显示 HID 设备, 该设备面板上设置有按键、解码器和二维跟踪球三种控制器, 能够快捷、精确地对各类视频色彩参数进行设置, 并将控制和状态信息同步显示在设备微型显示屏上, 从而改变了传统“鼠标+键盘”的视频编辑软件控制方式。Wave Panel 没有提供 Windows 环境下与应用程序

通讯的功能, 因此可基于 Wave Panel 设计一个功能完备、通用性强的通讯接口, 并加以验证。这对企业和科研机构根据实际需求灵活地配备和控制专用 HID 设备有很大帮助。

1 Wave Panel 的性能特征

HID 设备除满足 USB 标准外, 还具备一些与人机交互密切相关的特点, 主要包括:

1) 通常使用控制方式或中断方式通讯。两种传输方式的区别在于控制方式适用于对延迟没有特别限制的数据, 而中断方式适用于低延迟、周期性的数据^[2]。Wave Panel 的数据传输方式为中断方式。

2) 单次传输的数据量小, 且这些数据以状态变化信息或控制信息为主。其中低速 HID 设备单次传输数据最大为 8 字节, 全速设备为 64 字节, 高速设备为 1024 字节。主机与 HID 设备交换的数据存储在报表 (Report) 结构内^[3,4]。

Wave Panel 的输入报表长度为 27 字节, 其中 0 字节用于记录报表 ID, 1~6 字节的前 41 比特分别记录

收稿日期: 2012-08-28; 修回日期: 2012-11-30

基金项目: 中央高校基本科研业务费专项基金项目 (12zy019)

作者简介: 崔 阳 (1979-), 男, 博士, 讲师, 研究方向为知识工程与知识发现。

41 个按键的按下或抬起状态,另有 7 比特未用,置为 0;7~19 字节分别记录 13 个解码器的状态;20~25 字节每 2 字节分别记录一个二维跟踪球 X 轴和 Y 轴的状态;26 字节留作固件升级使用。Wave Panel 的输出报表长度为 41 字节,其中 0~5 字节为控制命令,6~40 字节为显示信息的字符。

3)使用轮询方式检测设备状态变化情况。HID 设备可能在任何时刻向主机发送数据,例如鼠标发生移动或键盘有按键按下。所以 HID 设备会在一个固定的时间间隔内轮询各控制口一次,当发现设备状态有变化时生成输入报表发送给主机^[5]。但轮询的间隔时间要由用户对 HID 设备的操作速度决定。如果间隔时间过小,则用户来不及在两次轮询间隔内完成一次操作;反之,又会使 HID 设备的响应速度无法与用户的操作速度匹配,造成延迟^[6]。

Wave Panel 规定,每 40 毫秒对所有控制器端口轮询一次,若其中任何一个控制器的状态在两次轮询间隔内发生改变,就会在下次轮询前生成输入报表发送给主机。40 毫秒是基于视频每帧显示时间而确定的。

2 HID 设备的通讯过程

HID 设备是 Windows 操作系统最早支持的 USB 设备。在 Windows 98 及以后的各版本中都已内置了 HID 设备的驱动程序,主机与 HID 设备通讯时无须再编写专门的驱动程序^[7,8]。这就简化了 HID 设备与主机应用程序通讯接口的设计和研发。

通过分析,可以将 HID 设备与主机应用程序的通讯划分为设备识别和打开、读报表、写报表和设备关闭四个模块。

2.1 Wave Panel 的识别和打开

HID 设备连接到主机后,以厂商 ID (VID)、产品 ID (PID) 和产品版本号 (PVN) 三项属性值的组合作为设备唯一标识^[9]。Wave Panel 的三项属性值可以从产品说明书直接获取。由于一台主机可能同时连接有多台 HID 设备,所以在通讯前应用程序必须要识别并打开 Wave Panel,这一过程需要调用多个 Windows API 函数实现。基本方法是首先获取当前与主机连接的所有 HID 设备信息的集合,然后逐一枚举集合中的各设备,并比较 VID、PID 和 PVN 值。当三项属性值都与 Wave Panel 相等时表示已经找到,调用 CreateFile() 将该 Wave Panel 打开,并保存 Wave Panel 设备句柄。该模块的流程如图 1 所示。

流程中调用的 Windows API 函数和结构体变量类型分别在 hidpi. h、hidsdi. h、hidusage. h、setupapi. h 和 hid. lib、setupapi. lib 几个文件中定义并实现^[10]。主要包括:

1)HidD_GetHidGuid():获取 HID 设备的 GUID 值,并保存在一个 GUID 结构体变量中;

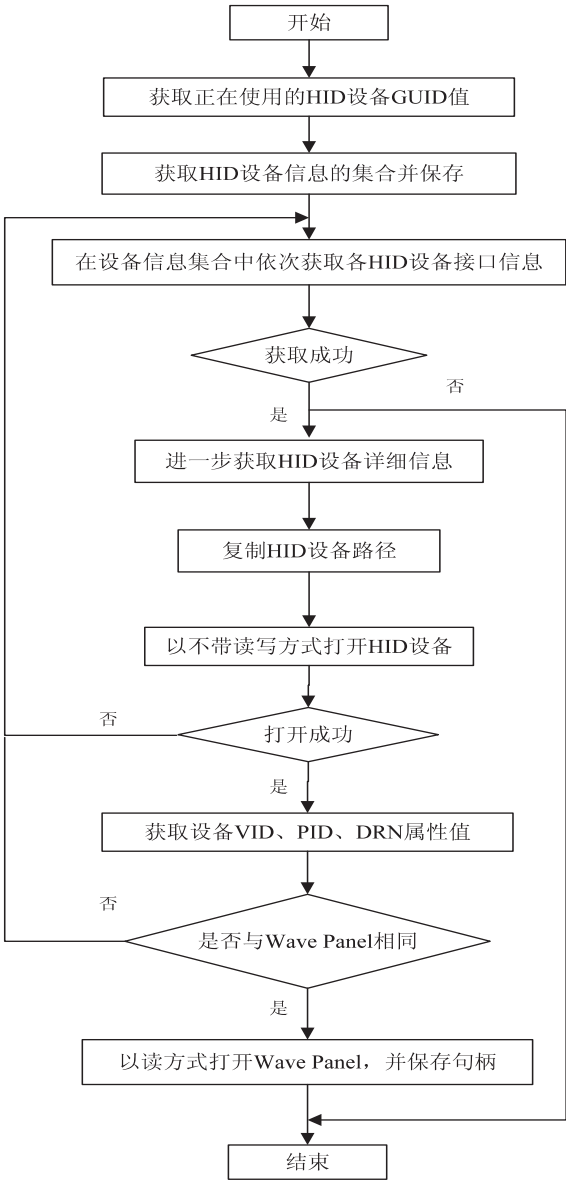


图 1 Wave Panel 识别和打开流程

2)SetupDiGetClassDevs():根据 HID 设备的 GUID 值获取 HID 设备信息的集合,并保存在一个 HDEVINFO 结构体类型的变量中;

3)SetupDiEnumDeviceInterfaces():在设备信息集合中依次获取编号为 $i(i = 0, 1, \cdots)$ 的设备接口信息,保存在一个 SP_DEVICE_INTERFACE_DATA 结构体类型变量中;

4)SetupDiGetDeviceInterfaceDetail():当获取设备接口信息成功时进一步获取该设备的详细信息,并保存设备路径;

5)HidD_GetAttributes():获取设备属性值并保存在一个 HIDD_ATTRIBUTES 结构体类型变量中。VID、PID 和 PVN 值即为该结构体变量的三项成员;

6) SetupDiDestroyDeviceInfoList():销毁设备信息

集合。

2.2 读报表

读报表是指 HID 设备向主机输入数据。当 Wave Panel 面板上任意控制器状态发生变化时,都会在下一次轮询时将这些数据生成输入报表,并发送给主机。主机上的应用程序接收输入报表并对数据进行解析,然后做出相应的处理。

由于 Wave Panel 可能在任何时候与主机以中断方式进行通讯,因此应用程序需要维护一个异步调用的读报表线程。该线程在 Wave Panel 没有向主机输入数据时阻塞,以减轻系统负载。当有数据输入时,线程恢复运行,调用 ReadFile() 将输入数据保存在缓冲区中,并对数据进行解析。这之后线程将再次被阻塞,直到有新的输入数据到来。应用程序在 Wave Panel 成功打开后就创建读报表线程,应用程序运行结束前该线程将一直存在。读报表流程如图 2 所示。

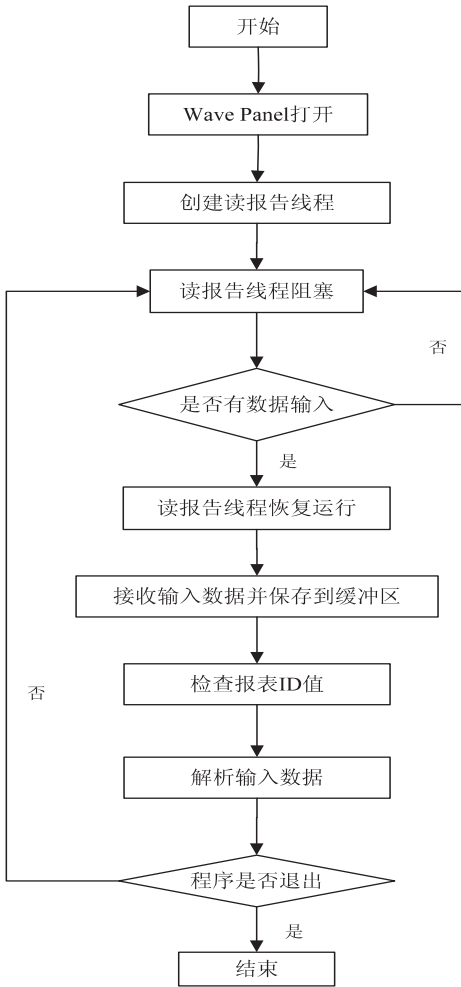


图 2 Wave Panel 读报表流程

2.3 写报表

与读报表相反,写报表是指主机向 HID 设备输出数据。写报表对延迟的要求较高,但这一功能不是必需的,根据 HID 设备的实际用途而定^[11]。鼠标、键盘、游戏杆等较简单的设备只向主机发送数据,这些数据

通常是一些控制信息和状态信息^[12],例如改变 HID 设备液晶屏上的显示信息、控制设备上发光元件的亮灭等。

应用程序的写报表线程流程与读报表线程类似,首先将要输出的数据按 Wave Panel 规定的格式进行封装,复制到一个缓冲区内,再调用 WriteFile() 将缓冲区数据输出到 Wave Panel 即可,这些信息主要是与视频帧状态变化相关的信息。

2.4 设备关闭

设备关闭模块功能比较简单。当主机与 Wave Panel 的通讯结束时,调用 CloseHandle() 将打开的 Wave Panel 关闭即可。

3 Wave Panel 通讯接口的设计

以上述通讯过程为基础,设计一个名为 CHidComm 的通讯接口,其总体框架为:

```
class CHidComm: public IUnknown
{
    .....
public:
    HRESULT __stdcall OpenDevice();
    HRESULT __stdcall CloseDevice();
protected:
    static UINTReadReportThread();
    static UINTWriteReportThread();
private:
    DWORD VID, PID, PVN;
    UCHARReadBuffer[ READ_LENGTH];
    UCHARWriteBuffer[ WRITE_LENGTH];
    .....
}
```

其中,成员函数 OpenDevice() 和 CloseDevice() 分别用于识别、打开和关闭 Wave Panel。ReadReportThread() 用于创建读报表线程。由于线程为异步方式执行,因此 ReadReportThread() 在调用 ReadFile() 时要使用一个 OVERLAPPED 结构变量,该变量含有一个 I/O 完成后触发事件的句柄。当 HID 设备没有向主机输入数据时,事件处于无信号状态,线程被阻塞;当有数据输入时,事件被触发,线程恢复运行,接收并将数据保存在读报表缓冲区,供数据解析函数使用。WriteReportThread() 与 ReadReportThread() 的功能类似,用于创建写报表线程。

成员变量 VID、PID 和 PVN 用于保存三项属性值,ReadBuffer 和 WriteBuffer 分别定义读报表缓冲区和写报表缓冲区,其大小由 HID 设备的输入报表和输出报表长度决定。S 实际上,该接口不仅可以用于 Wave Panel,同样也适用于其它 HID 设备。只需要重新设置 VID、PID 和 PVN 值以及 ReadBuffer、WriteBuffer 缓冲

基于Wave Panel的HID设备通讯接口设计

作者: [崔阳](#)
作者单位: [中国劳动关系学院, 北京100048](#)
刊名: [计算机技术与发展](#)
英文刊名: [Computer Technology and Development](#)
年, 卷(期): 2013(5)

本文链接: http://d.g.wanfangdata.com.cn/Periodical_wjtz201305038.aspx