Jan Axelson's
Lakeview Research

Search this site [GO]

Home | Books | PORTS Forum
USB | Serial Port | Mass Storage | Ethernet | More

The developer's resource for computer interfacing, especially USB, serial (COM) ports, mass storage, and embedded networking. (Formerly *Lvr.com*)

Home > USB Central > The HID Page

The HID Page

For developers of USB devices in the human interface device (HID) class.

Basics | Tools | My example code | Other code

Basics

My HID FAQ.

HID webpage from the USB-IF.

Windows programming

You can use Windows' built-in HID (human interface device) drivers to communicate with devices that conform to the USB's HID class specification. There's no need for a custom driver; the device uses the drivers included in Windows. Use any programming language that supports calling API functions. The device doesn't have to have a "human interface." Any device that can function within the limits of the HID specification (control and interrupt transfers only) may be able to be designed as a HID.

Microsoft's WDK has documentation for the HID functions and an overview of how to use them. The WDK also includes the header files to use with Visual C++ programs that access HID-class devices (hidsdi.h, hidusage.h, hidpi.h).

Human Input Devices. The Windows HID API from Microsoft.

Microsoft's DirectX technology includes DirectInput, which supports communications with HID-class game controllers (in both directions, in spite of the name), without requiring custom drivers. Developing Games from Microsoft has more information.

Use Raw Input to read data from a specific keyboard, mouse, or game controller.

A Usenet discussion about using HIDs under Windows CE.

Articles

HIDs Up by Jan Axelson, Embedded Systems Programming. From Embedded Systems Programming.

Using the HID class eases the job of writing USB device drivers by Stuart Allman. From EDN.

Making USB C# friendly. Article by Ashley Deakin in VSJ.

Books

USB Complete includes three chapters on HID firmware and application programming.

Chapter 13: Human Interface Devices. From *Programming the Microsoft Windows Driver Model, Second Edition* by Walter Oney.

Tools

Software

HIDMaker creates device firmware and a Windows application from information you provide. Also available are the AnyHID test program and the USBWatch software-only protocol analyzer. From Trace Systems Inc.

The SimpleHIDWrite utility tests HID-class devices. The HidTest utility also tests HIDs with a variety of API calls. HidTest.zip contains the application and HidTestSource.zip contains the source files in Delphi. All from Robert Marquardt.

USB HID API Function Library. From Kadtronix.

HID API for Linux, Mac OS X, and Windows. From Alan Ott.

USB HID driver for Windows. From embedded24.net.

YAT: USB HID based serial communication. From Matthias.

Hardware

MouseWarrior, KeyWarrior, JoyWarrior, IO-Warrier, and moreavailable as programmed chips or kits. From Code Mercenaries.

My example code

Host applications and device firmware to work with them. Thank you to everyone who has helped by contributing bug reports and often the fixes along with them.

Host applications

My HID applications below communicate with generic (custom) HIDs. Each supports exchanging Input, Output, and Feature reports and shows how to search for devices that use a particular interface GUID. All except the Visual Basic 6 code show how to use RegisterDeviceNotification and WM_DEVICE_CHANGE messages to detect when a device is attached or removed.

There are two versions of the application. One uses Filestreams; the other uses ReadFile and WriteFile.

Discussion about using Filestreams vs. ReadFile and WriteFile.

For Windows 10

To use my *generic_hid* project in Windows 10, open the *.csproj* file in Notepad or another text editor and delete this information, which relates to the signing certificate:

```
<PropertyGroup>
<ManifestCertificateThumbprint>xxxxx xxxxxx</ManifestCertificateThumbprint>
<ManifestKeyFile>xxxxxxxx.pfx</ManifestKeyFile>
<GenerateManifests>true</GenerateManifests>
```

```
    <SignManifests>false</SignManifests>
    </PropertyGroup>
```

Visual C# (C Sharp)

generic_hid_cs_62. Created with Visual Studio 2012 for the .NET Framework 4.5 or later. Uses v4.5's Filestream ReadAsync and WriteAsync methods. Uses System.Management class (WMI) to detect device arrival and removal. Requires Windows Vista or later and an attached USB generic Human Interface Device (HID). **This application does not run on Windows XP or earlier because the .NET 4.5 Framework will not install on these OSes.** Created 11/12/13.

generic_hid_cs_60. Created with Visual Studio 2012 for the .NET Framework 4.5 or later. Uses v4.5's Filestream ReadAsync and WriteAsync methods. Requires Windows Vista or later and an attached USB generic Human Interface Device (HID).**This application does not run on Windows XP or earlier because the .NET 4.5 Framework will not install on these OSes.** Created 2/11/13.

generic_hid_cs_50. Created with Visual Studio 2008 for the .NET Framework 2.0 or later. Uses Filestreams. Updated 4/1/11.

generic_hid_cs_46. Created with Visual Studio 2008 for the .NET Framework 2.0 or later. Uses ReadFile and WriteFile.

usbhidio_V2.3.cs. My VB HIDClass application ported to C#. From Gordon Vance. (Bug report: change: if (MyEnvironment.Version >= Version98SE) to if (MyEnvironment.Version <= Version98SE).

Visual Basic .NET

generic_hid_vb_50. Created with Visual Studio 2008 for the .NET Framework 2.0 or later. Uses Filestreams. Updated 4/1/11.

generic_hid_vb_46. Created with Visual Studio 2008 for the .NET Framework 2.0 or later. Uses ReadFile and WriteFile.

Visual Basic 6

Usbhidio2

Visual C++ 6

Usbhidio_vc6. This project will load into and run in Visual Studio. The project requires the header files hid.lib, hid.h, and hidsdi.h from the WDK.

1. If you get this errror on attempting to compile:

DBT_DEVTYP_DEVICEINTERFACE, PDEV_BROADCAST_DEVICEINTERFACE, HDEVNOTIFY, DEVICE_NOTIFY_WINDOW_HANDLE undeclared

Set WINVER = 0x0500 or higher in stdafx.h

For more info, go to groups.google.com and search on:

"DEV_BROADCAST_DEVICEINTERFACE" "undeclared identifier"

2. HidD_GetInputReport and HidD_SetInputReport require Windows XP or later.

Linux

Generic HID example using libusb-1.0.

Generic HID device firmware

HID firmware for communicating with the Windows host code above.

Microchip PIC18F4550

device-hid-generic-lvr.zip is a modification of Microchip's USB example titled Device - HID - Custom Demos. My code adds support for vendor-defined control transfers. Written for the PIC 18F4550 and Microchip's MPLAB mcc18 compiler v3.44 with the Microchip Libraries for Applications v2012-10-15. Tested with MPLAB v8.88 and MPLAB X v1.60. Updated 1/20/13.

Older versions

Use the newest Framework and firmware unless you have a compelling reason not to.

Fir the Framework V2.6a. pic_usb_device_hid_generic_10.zip

For the Framework V2.5. generic_hid_fsusb_fw2-5.zip
For the Framework V2.3. Generic_HID_c18_fsusb_fwv2-3.zip
For the Framework. V2.1. Generic_HID_C18_FSUSB
For the Framework V1.x. mchpfsusb_ghid.zip

Cypress Semiconductor EZ-USB FX2

fx2hid.zip runs on Cypress' FX2 EZ-USB chips at full and high speeds. It's adapted from Cypress' frameworks examples and requires the full version of the Keil C compiler.

Cypress Semiconductor Encore

Enchid.zip for Cypress Encore chips (CY637xx). Assembly code.

Cypress Semiconductor EZ-USB

fwhid.zip for Cypress EZ-USB chips. In C. More information.

Cypress Semiconductor EZ-USB Buttons and Lights

A version of John Hyde's buttons and lights code for the USBSimm module. (The module is no longer available.) My code supports interrupt Out transfers for Output reports.

Cypress Semiconductor CY63000

Usbhidio.zip (61k) contains assembly-code firmware and host software for communicating with a Cypress CY7C63000 microcontroller (obsolete). The code runs on Cypress' 3650 Developer's Kit and CY3640 Starter Kit. Cypress has discontinued the Starter Kit. The Hi-Lo EPROM programmer included in the Starter Kit was Cypress part CY3649. The host software is Visual Basic 6 code. Bug list and fixes for usbhidio.

Cypress Semiconductor CY63000 firmware plus Visual Basic code using an ActiveX Exe server

HIDDemo2.zip (70k) is an enhanced version of usbhidio. Like usbhidio that uses an ActiveX Exe server to handle reading from the device.

Other code

Code from other sources.

Device firmware for use with my .NET HID applications

These firmware examples are designed to work with my PC HID applications above.

Cypress Semiconductor 66113

Hidhub1.zip contains a version of my generic HID-class firmware for Cypress' CY7C66113 hub & peripheral chip. The firmware enumerates as a compound device made up of a 4-port hub and my usbhidio device. It runs on the 3652 development kit and will communicate with my hidvb and usbhidioc software. This code is provided by Mat Laibowitz.

Microchip PIC

Firmware for the PIC16C745/765 to enable using my HID .NET and usbhidioc software. From Microchip. For more PIC advice, see USB and PIC Microprocessors

16C745 and 18F2455 from Alan Macek.

The PICBasic Pro Basic compiler includes instructions to perform HID communications, for use with PICMicros with USB support. From microEngineering Labs Inc.

My Very First USB Peripheral. A description of a PIC16C745 HID project. From Technology Stir Fry.

More HID firmware

Atmel

Atmel's AT90USBKey demonstration board for the AT90USB AVR microcontroller has generic HID, keyboard, and mouse applications.

Cypress Semiconductor

I have a version of John Hyde's buttons and lights firmware that supports interrupt Out transfers.

The MO1002 USB to UART is a HID device that translates between asynchronous serial data and USB using Windows' HID drivers. Includes schematics, firmware, and a Visual C++ application. The chip used is Cypress' CY7C63001. From Moto Development Group.

HCC Embedded

Embedded host and device HID support.

Microchip

Microchip's Microchip Libraries for Applications includes HID examples..

ST-Ericsson (formerly NXP Semiconductors/Philips)

Interfacing a Game Boy Advance to a PC using a PIC16F877 and PDIUSBD12. From Rob Meerman.

Host code

Cross platform C# library for talking to connected devices such as USB and HID-class devices. Supports .NET, Android, UWP, Linux. From Christian Findlay.

HIDSharp. A multiplatform library for USB HID devices. Allows use by VB6 and MS Access programs. From Zer's Programming Page.

HID USB Driver / Library. A C# DLL. From Florian Leitner.

HidLibrary for .NET. From Mike O'Brien.

HID component for C# .NET. A project from the Avans Hogeschool in The Netherlands.

Controlling the Logitech iFeel Mouse. Experiments with force feedback. Daniel C. Moore

Topics

USB
USB embedded hosts & OTG
Serial ports
Mass storage
Embedded Ethernet and Internet
More topics

Books

USB Complete
USB Embedded Hosts
Serial Port Complete
USB Mass Storage
Embedded Ethernet and Internet Complete
More books

Connect

About
Contact
Support
How to Order
Newsroom
Site Map