

## Introduction

Holzworth Instrumentation uses the USB HID transfer protocol for all its USB devices because it simplifies installation, improves ease of use and provides the best PC stability. This application note outlines how USB HID works with Holzworth's instrument products and the DLLs.



## About USB Human Interface Device (HID)

The USB Human Interface Device is a class of USB device that uses a low bandwidth with a high update rate USB transfer, closely emulating the traditional serial port. USB HID interrupts are divided into 1000 time slots inside each 1 second interval with a maximum update rate of 1ms. Each data transfer can be up to 64 bytes, depending on the firmware of the USB device for a maximum data transfer of 64kBytes/s. One advantage is that this data transfer is guaranteed, if requested by the software.

USB HID devices actually have several advantages: Guaranteed data transfer; virtually all modern operating systems recognize USB HID devices as this is the same protocol used for keyboards, mice, track pads, game controllers, *etc.*; USB HID devices do not require installation of device drivers because the operating system contains imbedded commands allowing for direct access to these HID devices.

USB HID allows for the best path to platform independence. Since USB is a third party standard which is independent of any operating system manufacturer, the device manufacturer simply complies with the standard for guaranteed recognition and operation.

## DLL Access to USB HID in Windows

MS Windows provides direct access to all USB HID devices. Unfortunately, this function is spread out over multiple DLLs via a relatively complex interface. To simplify this interface, Holzworth provides its virtual instruments with two DLLs. The Hardware DLL (HolzworthUSB.dll) is used to run all low level operations, simplifying the direct MS Windows interface. The function of the Instrument DLL is to act as an interface between the user application and the Hardware DLL. Note that the Hardware DLL cannot be accessed directly by the user. The user is given access to the Instrument DLL (ex: HolzworthHS1001A.dll for the HS1001A) for each specific device being used.

The Instrument DLL converts all high level commands to a byte level USB transfer that is understood by the virtual instrument and transferred by the Hardware DLL. Holzworth provides both string and integer commands to make interfacing with user applications as easy as possible. For further details, refer to Holzworth Application Notes:

**AN102-1.0**  
**AN103-1.0**

**Functions of the HolzworthHS1001A.dll**  
**Using the HolzworthHS1001A.dll with Matlab™**