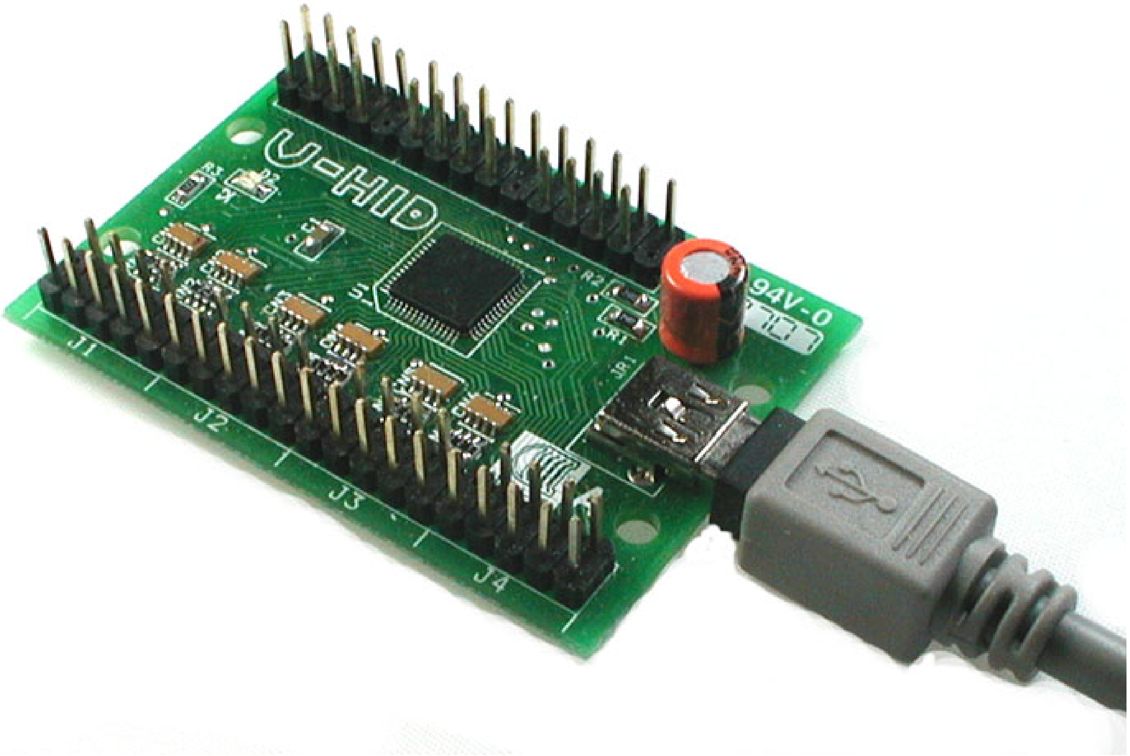
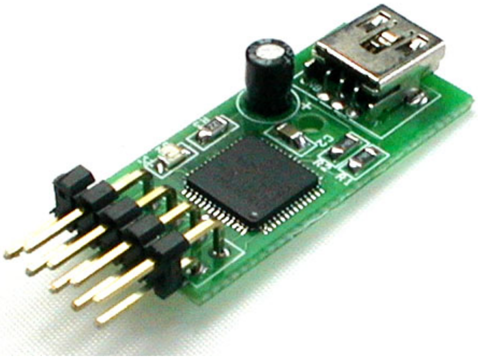


UNIVERSAL HUMAN INTERFACE DEVICE

TECHNICAL MANUAL



Section 1: Introduction

HID is a computing term for a class of devices called Human Interface Devices. These are certain types of PC peripherals, usually connected via USB. Windows and Mac OS support a number of "native" HID devices using in-built drivers. The U-HID board appears as the following HID devices on the PC:

- **Keyboard**
- **Gamepad (including analog and digital controls)**
- **Mouse**
- **LED outputs**

The way in which the board emulates natively-supported devices means it requires no special drivers or host software, with the exception of LED output control.
The board has 50 connections (or 8 for the Nano version) which can be configured to any of these HID device types providing a truly universal interface. For example, you can connect pushbuttons to generate keyboard keystrokes, trackballs to act as a mouse, pedals to act as analog game controller joysticks, or hundreds of other device types. The PC will see all the devices as something it recognises, so all standard applications will work.

Connections can be assigned as:

- Keyboard key
- Gamepad Button
- Gamepad POV Hat
- Gamepad analog axis
- Gamepad analog axis with auto calibration and dead zone
- Mouse button (including double-click)
- Quadrature Mouse Pair. X or Y axis. Normal or reverse direction.
- Quadrature left/right button presses.
- PC-controlled LED output using special software
- PC Num/Scroll/Caps lock LEDs
- U-HID locally-controlled LED output based on the state of any switch.
- 5 Volt output (30ma max)
- Ground

Features

- No special drivers or application changes needed except for PC-controlled LEDs which has an API with code examples.
- Simple open pin-header connectors allow use of separate or composite modular wiring harnesses.
- U-Config configuration utility allows the connections to be configured for each device type, and keycodes and other attributes set.
- Switches handled by super-fast logic using an individual de-bounce state counter for each input giving ultimate performance for gaming applications.
- No key ghosting or blocking ever.
- Advanced keycode/button programming with in-built shift feature. Includes macro capability
- 12-bit analog axis support
- Configuration utiity interacts real-time with the board and all settings stored instantly in Flash ROM on the board.
- Single Mini-USB connector for PC (or MAC) connection. No separate power needed
- 50 Truly independent inputs allow all switches to be commoned to ground. No matrix or diodes
- Modular 8-way + ground harnesses available.
- Extended switch harnesses available.
- ESD protected inputs. Vital for high-footfall public applications.
- All settings stored in Flash ROM and retained after power-off.
- Configuration can be changed on-the-fly using a command-line. The board can assume a completely different layout instantly.
- Simple-to-use PC Configuration application.
- Up to 8 boards can be connected. All are configured using the same U-Config utility.

Advanced Features:

Key and Button functions can have one primary and one secondary code assignment.
Secondary assignments are invoked by first activating a “U-HID Shift” connection.
Any connection can be assigned as a shift.
Unique separately-programmable “button down” and “button up” events allow use of all types of normally-closed, normally-open, or active high/low controls.
“Button down” and “button up” can be separately assigned as primary or secondary codes.
Controls can be assigned as “normal” or “pulse”.
Toggle (push on-push off) and flip-flop modes available.

Programmable analog offset and scale factor for all analog axes.
X and Y axes can be configured with auto calibration and a dead-zone.
Extended macro capability.
User-upgradeable firmware for future enhancements and customizations.

Examples of control types which can be connected

- Pushbuttons
- Switch-type joysticks
- Toggle switches
- Toggle switches with integral LED (does not require a second connection for the LED)
- Momentary pushbuttons with integral LED (LED can be controlled without any host application, or can be host-controlled)
- LEDs associated with a switch. (LED can be controlled without any host application)
- LED indicators driven by a PC application.
- Other devices can be controlled either directly or via an external high-current driver
- Analog joysticks
- Incremental Rotary encoders (Spinners).
- Low-resolution rotary encoders (360 degree volume control type)
- 12-way or N-way rotary switches used as encoders (requires simple diode circuit)
- Trackballs
- Pots
- Optical steering wheels
- Pot-type steering wheels
- Pot-type throttle pedals
- Analog Voltage sources (0-5 volts).
- TTL logic signals.

Control counts which can be configured:

Total 50 connection pins.

Max 8 analog axes on 50 possible pins (Windows limitation)
Max 32 Gamepad buttons on 50 possible pins (Windows limitation)
Max 8 Quadrature mouse devices on 16 possible pins out of the 50.
Max 3 Quadrature mouse axes (X, Y and Z, Windows limitation)
Max 50 keyboard keys on 50 possible pins.
Max 16 LEDs in any combination of PC or U-HID controlled. (USB 500 mA current limitation).
Max 50 low-current output drivers.

Section 2: Connecting Devices

Important! Never connect any device to the U-HID which sends more than 5 volts otherwise damage could occur!

Connecting Switches and Pushbuttons

Any pin can be defined as a switch.

You can define a switch as:

- Mouse button (including double-click)
- Gamepad button or POV Hat direction.
- Keyboard key.

  Each switch input has a primary assignment and a secondary assignment (optional).

  Both the primary and secondary assignments can be used in key action settings and the secondary assignment is also activated in place of primary when a "U-HID Shift" key is held down.

  You can assign a different action for the event when the switch is closed (ie grounded) and for when the switch is opened.
  The two possible keycodes (or buttons) coupled with the way in which you define closed/open events gives a mass of flexibility in the types of control you can connect without using any external circuitry.

The keycode which is sent when you activate the switch is defined using the U-Config utility. You can assign any connection to send any keypress.
Switches can also be assigned as game controller buttons, which are numbered 0-32 (32 is a Windows limitation).
Mouse button selections available are left, right, middle or left double-click.

Example 1: Simple keyboard-type button (Normally Open)

For this you would set the “down” action to be the required keycode and “up” action to be “Clear”.

Example 2: Simple keyboard-type button (Normally Closed)

Set “down” action to be “clear” and Up action to be the required keycode.

Example 3: Toggle switch with integral LED

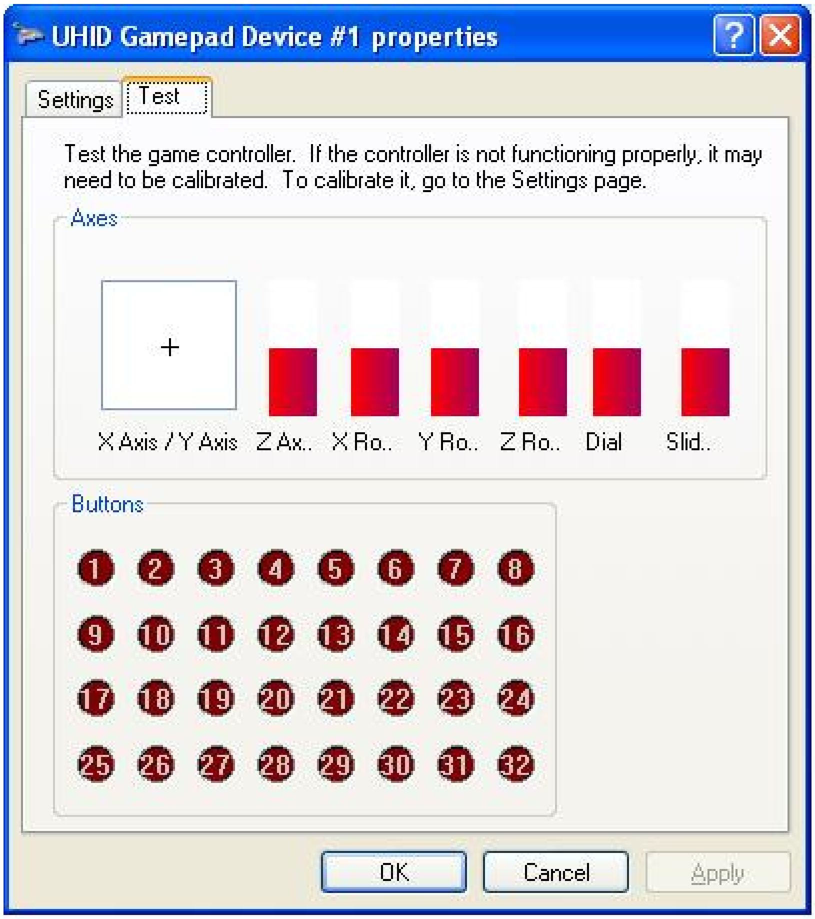
This type of switch sends a 5 volt signal when ON, and is open-circuit when OFF. Let’s say you want to turn on a control when you switch “on” the toggle but you don’t want to generate masses of key-presses which the toggle is still down. You can send one key-press only, by configuring:“down” action to be “clear” and “up” action to be “pulse” and the required keycode. Setting it up this way means the internal LED in the switch will function properly with no external circuitry to control it.

ESD Protection

Connectors J1, J2 and J3 have ESD static discharge protection. These 24 pins should be used in preference over the others when connecting switches. ESD can be a problem in warm dry environments and can cause false triggering or even damage.

Testing Switches

Switches defined as keyboard keys can be tested using Notepad or any program which displays text.
Gamepad buttons can be tested using Control Panel, Game Controllers. The display is shown below:

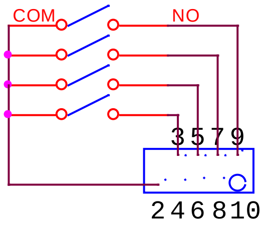


UHID Gamepad Device #1 properties

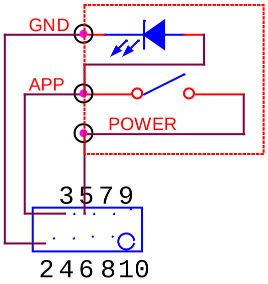
Settings | Test

Test the game controller. If the controller is not functioning properly, it may need to be calibrated. To calibrate it, go to the Settings page.

Examples of Switch Connections

EXAMPLE: FOUR PUSHBUTTON SWITCHES			
		PIN	CONFIG
		2	GND (FIXED)
		3	SWITCH (KEY OR BUTTON)
		4	
		5	SWITCH (KEY OR BUTTON)
		6	
		7	SWITCH (KEY OR BUTTON)
		8	
		9	SWITCH (KEY OR BUTTON)
		10	

EXAMPLE: TOGGLE SW WITH INBUILT LED



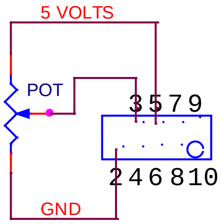
PIN	CONFIG
2	GND (FIXED)
3	SWITCH (KEY OR BUTTON)
4	
5	5 VOLTS
6	
7	
8	
9	
10	

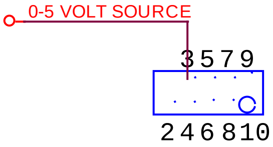
NOTE: For this type of switch, the switch must be configured as:
DOWN = CLEAR   UP = <REQUIRED KEY>

Connecting Analog Devices

Important! Never connect any device to the U-HID which sends more than 5 volts otherwise damage could occur!

A pin assigned to Analog senses a voltage on the pin and controls a gamepad X, Y, or other axis on the PC.
Analog resolution is 12 bit.
The voltage can vary between 0 and 5 volts.
Usually this voltage will come from a potentiometer (pot) which is part of a steering wheel, handwheel, throttle pedal, or other type of analog control.
You can assign any pin to one of 8 possible analog axes. (8 is a Windows limitation).
To test, go into “Control Panel”, “Game Controllers”. Then select the board number (if you have more than one board).You will see the X, Y crosshair (axes 1 and 2 on the U-HID board) and also the other 6 axes as red bars.

EXAMPLE: POTENTIOMETER AS AN ANALOG CONTROL			
		PIN	CONFIG
		2	GND (FIXED)
		3	ANALOG INPUT
		4	
		5	5 VOLTS
		6	
		7	
		8	
		9	
		10	

EXAMPLE: VOLTAGE SOURCE AS AN ANALOG CONTROL		
	PIN	CONFIG
	2	GND (FIXED)
	3	ANALOG INPUT
	4	
	5	
	6	
	7	
	8	
	9	
	10	

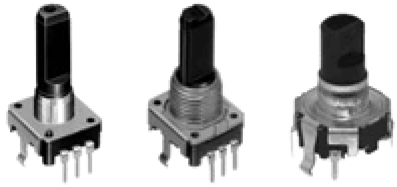
Connecting Optical or Mechanical Encoders (Spinners, Trackballs etc)

High-resolution Quadrature devices usually consist of a pair of optical sensors and an encoder wheel. Examples of such devices are:

- Spinners such as the Ultimarc SpinTrak
- Trackballs such as the Ultimarc U-Trak
- Incremental Optical Rotary Encoders
- Incremental Mechanical Encoder eg ALPS EC12 Series

- 12-way rotary switches such as the Happ Controls Rotary Joystick. See App Note AN1001

- Linear Motion Sensors



ALPS EC12 Mechanical

Ultimarc U-Trak

Ultimarc SpinTrak

Quadrature devices can be used in two ways. They can be used to control the mouse pointer in an X, Y or Z direction, or they can be used to repeatedly pulse gamepad buttons. In general, mouse mode is used for high-resolution optical encoders and button mode for low-res mechanical but there may be applications where the opposite mode is useful.

Quadrature devices use two input pins per device, and optical devices will require GND and 5V pins. Mechanical encoders require GND only, and this is the center pin on 3-pin devices.

Windows allows one instance of each axis when defined as Mouse (ie X, Y, Z) but if you are using multiple U-HID boards, Windows can recognise each board as a different mouse, using DirectInput.

Devices should be capable of operating with a 5 volt supply (not required for mechanical encoders).
This can be obtained either by configuring a pin as 5V or using one of the fixed 5V sources on connector J7
They also require ground, which can be connected to a configured pin or one of the ground pins on each modular connector.

Note that only certain pins can be configured for quadrature devices. These can be seen in U-Config by clicking on the "Quadrature" line in the PCB image. Appropriate pins will be highlighted. Two pins are used per device. An X-Y device such as a trackball counts as two devices and therefore uses 4 pins.

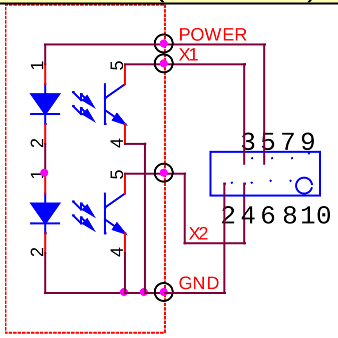
Mouse Mode

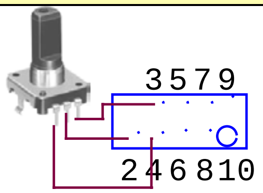
Encoders configured in this more simply move the mouse pointer in the X, Y or Z direction. Note the Z axis in Windows performs differently to X and Y. It accepts a fixed displacement per data packet rather than a variable one. This means the Z axis might not be suitable for many applications as it is very sensitive when using a high-res encoder.

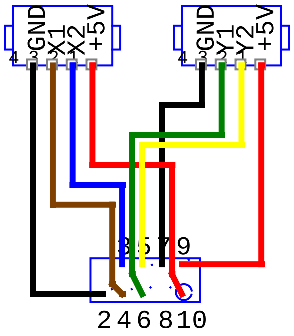
Button Pulse Mode

In this mode, the quadrature device is assigned to a pair of Game Controller buttons. The pairs are buttons 1,2, buttons 3,4 etc up to 15,16.
When the encoder is turned in one direction, the first button will be repeatedly "pressed". When turned in the other direction, the other button of the pair will be repeatedly "pressed".
The press time is configurable. This is a global setting on the "Calibration" page of U-Config. The setting here can be set from 2 to 32. The figure is actually the number of USB data packets sent for each press. Between each press, there must of course be a "release" period and this is the same time as the press.
It might be necessary to experiment with this setting as there is a trade-off. Too fast a setting will cause presses to be missed by the application, and too slow will cause slow response of the control.
When this mode is used for high-resolution encoders, the buttons will be pulsed at the maximum configured speed which is much higher than the real pulse-rate from the encoder.

Examples of Quadrature Device connections

EXAMPLE: OPTICAL WHEEL (SPINNER)			
	 POWER X1 X2 GND 1 2 5 4 3 5 7 9 2 4 6 8 10		CONFIG
		2	GND (FIXED)
		3	QUADRATURE
		4	(AUTOMATICALLY SET)
		5	5 VOLTS
		6	
		7	
		8	
		9	
		10	

EXAMPLE: ALPS LOW-RES ENCODER		
	PIN	CONFIG
	2	GND (FIXED)
	3	QUADRATURE
	4	(AUTOMATICALLY SET)
	5	
	6	
	7	
	8	
	9	
	10	

EXAMPLE: U-TRACK TRACKBALL HARNESS		
 GND X1 X2 +5V GND Y1 Y2 +5V 3 5 7 9 2 4 6 8 10	PIN	CONFIG
	2	GND (FIXED)
	3	QUADRATURE X
	4	(AUTOMATICALLY SET)
	5	QUADRATURE Y
	6	(AUTOMATICALLY SET)
	7	GND
	8	
	9	+5 VOLTS
	10	+5 VOLTS

Connecting LEDs or other output devices

A limit of 16 LEDs applies to the board to stay within USB specificatons on current draw (500mA). The U-Config utility will allow you to configure more than this but a LED count on the status line will show red as a warning. You can configure more than 16 if you are using the outputs as low-current drivers, eg for TTL signals.

The U-HID will drive standard LEDs only, max current 30 mA

Pins on J1, J2 and J3 have integral resistors so you can directly connect standard LEDs to these. Other pins will need a series 220R resistor, or use 5 volt

LEDs with an in-built resistor.

LEDs are connected between the U-HID pin and ground.
The pre-made modular harness can be used to connect LEDs.

Other devices can be connected but the following applies:
Max current draw: 30 mA
No external voltage must be applied to the controlled device.
The U-HID board supplies 5 volts or zero. No other voltage is available or should be used.

U-HID Controlled LEDs

U-HID Controlled LEDs are LEDs which are turned on or off by switches which are connected to other pins on the board.
Up to 16 connections can be assigned as U-HID-controlled LEDs. When you assign these LEDs, you link it to any other connection. The other connection must be defined as a switch. This other connection will control the state of the LED.
Its important to realize that the LED state is not actually controlled by the other switch connection pin directly, but by the boards stored internal state of the control connected to that pin. This means the control of the LED is affected by how the switch is defined as well as its actual open/closed state.

Example1

Assume you have a pushbutton which has an internal LED. It is a momentary button so cannot be used to directly switch the LED since it would only light while the button is pressed.
You have this button configured as a toggle (push on, push off).
You can assign the LED to this button, and it will light when the button is pressed and go dark when it is pushed again, in sync with the buttons action in the application.

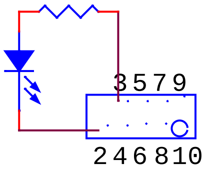
Example 2

Assume you have a toggle switch which grounds one connection when closed.
This could not be used on its own to also control a LED.You could link a U-HID controlled LED to this connection and it would light while the switch s closed.

PC Controlled LEDs

Up to 16 connections can be defined as PC-Controlled LEDs. You can also use these for other output functions.
These LEDs are controlled by a software application on the PC. There are sample APIs for developing your own applications to add LED control.

You can also assign LEDs to behave as keyboard CAPS, NUM and SCROLL lock LEDs. They will be turned off or on in the same way as your regular keyboard LEDs.

EXAMPLE: LED (STANDARD TYPE)		
	PIN	CONFIG
	2	GND (FIXED)
	3	LED OUTPUT (PC or UHID)
	4	
	5	
	6	
	7	
	8	
	9	
	10	

Connecting to Power and Ground

You can assign any connection as power output (+5V) or ground.
You need to be aware of the following limitations:

- Power and ground pins can only source/sink 30mA max current per pin.
- Total current draw must not exceed 500mA

If you need more current than 30mA then you can use one of the fixed power pins on connector J7. See the U-Config program board image for details of which pins these are. In fact it is good practice to use these pins whenever possible for a 5 volt source as they are powered directly from the USB supply.
Using assigned pins instead offers advantages of modular connectors and is

particularly useful for optical devices which always need a supply for the optical switches and dont have a large current requirement. As each modular plug has one fixed ground connection you would rarely need to assign a ground pin.

Section 3: Configuration

Understanding The U-HID Shift Feature

The internal shift feature allows you to define a connection (or connections) which when activated, cause all other switch connections to send an alternate command. If you only want each pin to send one key or button, you can skip this section.

This is rather like a keyboard shift key which, when held, causes the keyboard keys to send different characters. When the internal shift connection is activated and held, the Secondary function of all other keys is sent instead of the Primary.

EXAMPLE 1:
In a gaming controller you might wish to assign an admin function such as COIN INSERT and you don't wish to have a dedicated button on the panel for this. You could make PLAYER1 START an internal shift connection. Then assign a secondary COIN INSERT function to the switch which is normally "PLAYER 1 FIRE". So to insert a coin, hold PLAYER1 START and press PLAYER1 FIRE.

EXAMPLE 2:
You have a control panel which is used for two completely different functions and you wish these functions to be selectable using a toggle switch. You could assign a spare connection as an internal shift connection, and then connect a toggle switch to it. Then assign all Primary functions to your first control panel type and all Secondary functions to the second type. Switching the toggle would flip the panel from one mode to the other. (Be aware you can also achieve this by re-programming the board on-the-fly though!).

Defining Internal Shift Connections

Normally you would perform this on only one connection, which will act as the shift key.

Most users would only require to define this function once, so to avoid clutter on the display, it is hidden unless the connection is completely un-assigned. This means with the default configuration, which has all connections assigned, you won't see this option.

To assign a connection as an internal shift:

Click on the connection and select "not used". A check box will appear with the option "This is an Internal Shift Switch".

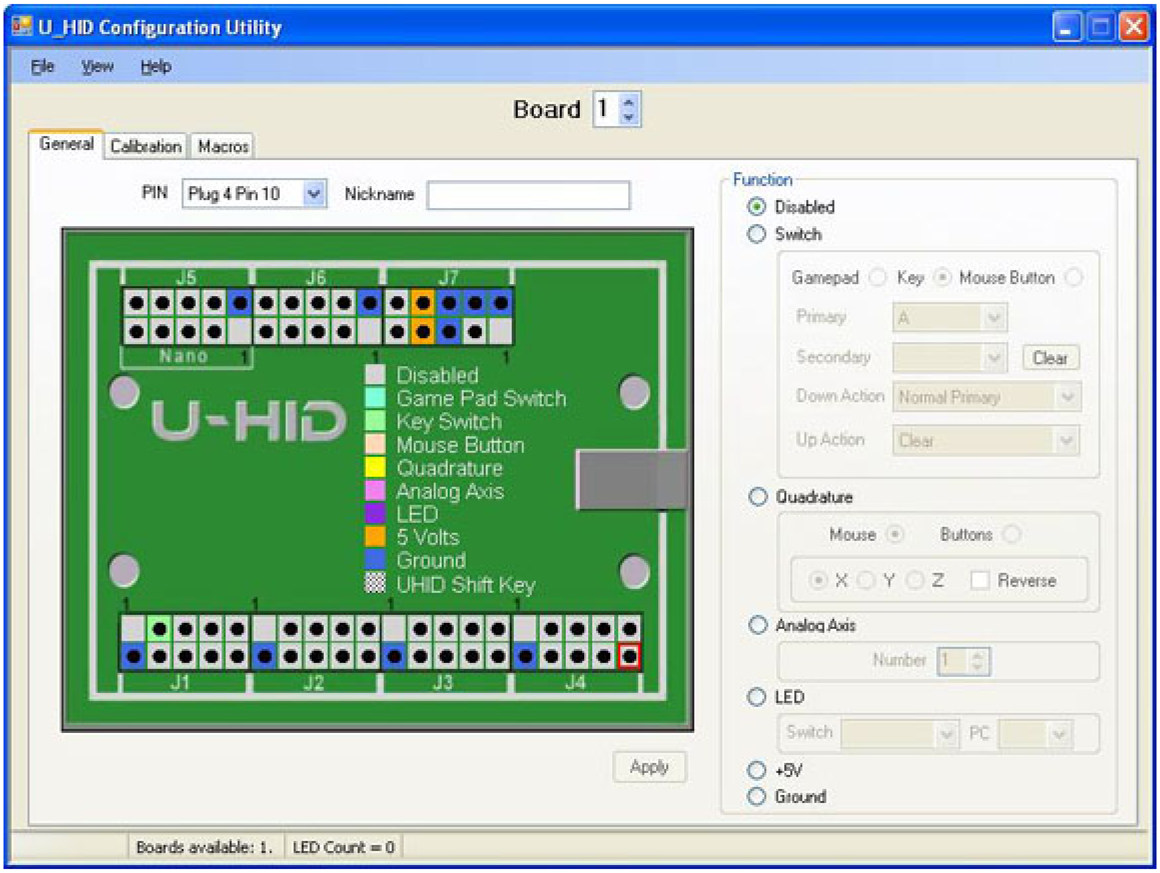
Enable the check box. This will be highlighted in red and will remain visible while this connection is selected.

Checking this box limits what you can do with the other selections for this switch. You are limited to assigning a "Pulse action on Open" , or no action at all. Any other type of assignment is invalid for an internal shift connection.

You cannot assign this connection as an LED, Analog or Quadrature input.The ability to assign a "pulse on open" action means you can also have this connection perform its own function as well as being a shift key.

If the key is pressed and released, it will send its own function as a pulse, provided no other connection has been activated in the meantime. This is how Example 1 works, above. The Player 1 Start function of this key still works even though it's a shift key as well. You can assign any number of the connections to be internal shift, but they all will enable the same shift functionality in addition to their own individual assigned code (if any).

Using the U-Config Configuration Utility



Board 1

PIN Plug 4 Pin 10

Nickname

◉ Disabled
○ Switch

Gamepad ○ Key ● Mouse Button ○

Primary A

Secondary

Down Action Normal Primary

Up Action Clear

○ Quadrature

Mouse ● Buttons ○

● X ○ Y ○ Z □ Reverse

○ Analog Axis

Number 1

○ LED

○ +5V
○ Ground

Boards available: 1. LED Count = 0

This utility is used for the following:

- Assigning a board number (1-8) if you have multiple boards.
- Configuring each pin of the board as the correct device type, ie keyboard, gamepad, analog, quadrature mouse, power.
- Configuring each pin with the correct keycode, button number, axis or other parameter
- Assigning macros
- Configuring analog axis scale factor and offset
- Loading and saving configurations
- Initiating a firmware upgrade
- Downloading a configuration "on the fly" using a command-line task

The graphical picture of the board shows the way in which the pins are divided into sections, each being one 9-pin connector (plus key). Pin 1 is the key and pin2 is ground. The utility interacts with the board in real time, every time a change is made. The configuration is stored in flash ROM on the board so it remains after power off/on. When the utility is run, it reads the configuration from the connected board(s) and displays it.
You can also store the entire configuration as a file for later downloading on-the-fly for multiple configurations.

Function of each GUI feature:

GENERAL tab (main window)

PIN drop down
This displays the selected pin you are configuring. You can select pins from here or from the main graphic.

NICKNAME
You can type text in here which will remind you of which device this pin is connected to. The test is saved with the configuration (on the PC only) and is not sent to the board.

FUNCTION area
Here you configure the selected pin as the required device type and assign values to it.You can select one of the radio buttons to define the pin as the following:

Switch	Includes keyboard key, gamepad button and mouse buttons
Quadrature Mouse	Includes trackballs, optical steering wheels etc
Analog	Gamepad type control
LED output	Includes U-HID and PC Controlled
+5 Volts out	Max 30 mA supply
Ground.	Max 30 mA sink

APPLY button

This button applies the changes you have made to the pin. Changes are immediately sent to the board.

GRAPHIC AREA

You can click on any pin on the graphic to start configuration of this pin. Additionally, if you click on the pin types listed in the center of the PCB, this will show which pins are allowed to be configured as this type of control.You will notice that certain pins on P7 are fixed. These are high-current power/ground pins and are hard-wired and cannot be changed.
Each 10-pin connector area has one hard-wired ground pin.

CALIBRATION tab

This tab allows you to set scale factor and offset for each analog input. The default values give a full-scale travel with the cross-hair centered at mid-position of the pot. It may be necessary to experiment with these settings and you can observe the result by opening Control Panel, Game Controllers in Windows, and clicking on the came controller device to display the calibration screen.

In addition the Calibration tab also contains the setting for Quadrature Button press time. This is the length of time (actually the number of USB packets) which the game controller button is pressed for when using Quadrature Buttons. This is a global setting.

Detail Configuration Example (Switch)

The screenshot above shows this configuration.
We will look at how a switch is connected and configured.
Firstly, the switch is connected with one terminal routing to the appropriate pin and the other to a ground pin.

- Select the pin by clicking on it in the graphic area.
- Select the "Switch" radio button in the Function area.

You now need to select what type of control this will be, whether a mouse button, keyboard key or gamepad button.

- Select keyboard key.

In the Primary drop-down, all possible keyboard keys will be available, plus macros. Macros are defined separately, see later.

- Select which keyboard key you need.

Note that the U-HID sends key scan codes, just like a keyboard. It does not send characters so has no concept of upper/lower case. An upper-case character is two keys (ie shift and the character key). There is no need to assign a secondary code unless you wish to use the shift feature or other special functions, see later.

Then, you need to tell the board how and when to send the keycode you have chosen. This is done with the Down and Up Action drop downs. These can be selected as follows:

Clear

This removes the keycode associated with this pin from the buffer, so it is no longer sent

Normal Primary

This sends (and holds) the primary keycode you selected above, to the PC. If the U-HID Shift button is pressed, it will send the secondary code instead.

Normal Secondary

This sends (and holds) the secondary keycode but you have not assigned one at this point so no action.

Pulse Primary

This sends a short pulse, simulating the key being pressed then released, even though it might still be physically closed (eg a toggle switch left closed)

Pulse Secondary

This is the same as above but pulses the secondary code.

Toggle

This is not explicitly an option, but is the result of setting both the Down and Up actions to Normal Primary. This will cause the selected keycode to alternately be sent or not sent on alternate button presses.

Flip-Flop

This is not explicitly an option but is the result of setting Down action to Secondary and Up action to Primary. This will cause the primary or secondary key to be sent on alternate button presses.

See-Saw

This is not explicitly an option but is the result of setting Down action to Primary and Up action to Secondary. This will cause the primary key to be sent on press and secondary on release. Normal or Pulse mode can be used.

The most common usage of keyboard keys is to set Down Action to Normal Primary and Up Action to Clear.

Important Note about keyboard auto-repeat:
On USB keyboards (and therefore the U-HID) the auto-repeat of the keyboard is done in Windows. So if you assign any keycode to be pressed permanently (which can occur in many configurations) you will get an endless string of characters in your application. For this reason, many of the advanced combinations should only be used with gamepad buttons, which do not have a problem being held in the pressed position.
You should also avoid defining any keyboard keys from being "pressed" by default as this will cause boot failures (stuck key) and may cause unexpected problems in the Windows GUI.

Other switch types:

Mouse Button
You can select left, middle, right and left double-click from the drop-down.

Gamepad Button
These are numbered 1-32 as shown in "Control Panel, Game Controllers".

Quadrature Mouse Selection

This selection always uses two pins. When you assign the pin as this device type, the opposite pin will also automatically be assigned.
You can select X, Y or Z (wheel) axis.
Trackballs use two axes (4 pins).

Quadrature Button Selection

This selection uses two pins. When you assign the pin as this device type, the

opposite pin will also automatically be assigned.
You can select one of 8 pairs of game controller buttons which will be pressed repeatedly when the encoder is turned. Note you can configure the "press" time on the Calibration tab.

Analog Axis

These are numbered 1-8. You can view analog axes using Control Panel, Game Controllers.
The first two axes are displayed as X and Y.

Conventional analog axes

The analog behavior can be fine tuned by adjusting the offset and scale factor in the calibration window (tab on the top of the display). To observe changes, open the Game Controller window in Windows. In Win7 and later this is accessed by “Devices and Printers” and right-click on the U-HID device.

Analog X Y with deadzone and auto-calibration

Normally, the host will see a linear change in the analog value for each axis. But the X and Y axes can be configured in the drop down to emulate a game controller which has auto-centering at power-up and a dead-zone of no movement in the center of travel.
In this mode, the amount of dead-zone is adjustable by changing the “offset” value in the calibration window (tab). The scale factor may need to be adjusted for full travel.

LED Output

There are two types of LED.

U-HID controlled (switch)

This type of LED is switched on when the switch allocated to it is sending its keycode or button (which might not necessarily mean the switch is physically closed, depending on how the switch pin is configured).
In the drop-down, you can select which switch (referenced by its pin) controls this LED.

PC Controlled LED

This type of LED is switched on/off by a PC application. These are numbered 1-16.

Macros

Macros are sequences of key-presses. Each key is logically pressed in turn, and held, then after the last key in the macro is sent, all are released. This means for example, the sequence “Ctrl-Alt-Del” can be programmed.

Up to 30 macros can be programmed and there is a limit of 84 macro keys (total of all keys in all macros).

Macros are firstset-up using the Macro tab on the main U-Config screen. Each macro is assigned a number. Then the marco number is used in the main configuration.

For example to configure “Ctrl-Alt-Del” as a macro:

1. Select the Macros tab in the main window
2. Click “New”
3. From the drop-down, select L-CTRL
4. Click “Add Entry”
5. Repeat steps 3 and 4 for “L-Alt” and “Del”
6. Click “Add macro”

The macro will now be shown in the list and a number is assigned to it. You can now use this macro number in the main configuration screen in the same way as any other keycode etc.

Other Functions

FILE menu

Open, Save

These selections allow loading/saving of .HID configuration files, so you can have multiple saved configurations.

Import/Export

You can import and export configurations to a CSV or XML file so you can edit them outside of the application or generate using a different application.

Assign ID

This is an important first step when you are using more than one board. You will need to connect ONE board, and assign it as number 2 or higher. Once all boards are assigned with different IDs, they can all be connected to their own USB ports. You can check which board is which by opening Control Panel, Game Controllers. The board ID# is displayed a the top of each Game Controller window.

Firmware Upgrade

This initiates a firmware upgrade. The board assumes a different USB ID and a driver will be installed. See page 30 for more details.

VIEW menu

There is only one option, "View U-HID Shift Option". This is normally hidden in the main area to avoid clutter.
This is only used when assigning a pin to be a U-HID shift key.
The shift option allocates a key (or keys) which, when held, cause other pressed keys to send their secondary instead of primary keycodes. This over-rides the Primary/Secondary Action defined for keys.

Running from a command-line

The U-Config utility can be run from a command-line (ie Batch file) by using the command “U-Config <name of HID file>”. The board will be programmed with the contents of the HID file.

Output Software API

Introduction

The Ultimarc PacDrive & U-HID SDK is a collection of source code examples for controlling the PacDrive & U-HID LED controller hardware by Ultimarc.
It can be downloaded from:
http://www.ultimarc.com/pacdrive.html

It contains source code projects for the following languages:

- C#
- C++
- Delphi
- VB6
- VB.NET

API Functions

int PacInitialize()

- Initialize all PacDrive and U-HID Devices
- Returns the number of PacDrives and U-HID's on the PC
- They are sorted in order of PacDrive's Version Numbers, then U-HID's PID.

void PacShutdown()

- Shutdown all PacDrive and U-HID Devices
- No return value

bool PacSetLEDStates(int id, USHORT data)

- Sends data to the PacDrive or U-HID specified by id
- Returns true for success and false for failure
- You can use up to 16 PacDrive's and 16 U-HID's (id 0 to id 31). The order they are sorted is PacDrive by Version number first then U-HID by PID.

Technical Data

	U-HID	**U-HID NANO (where different)**
Dimensions mm (excl connectors)		
W	38	15
L	56	35
H	10	10
Mounting Hole Dia	3	2
Mounting Hole rectangle	49.5 X 16	n/a
Operating Temp deg C	-10 - +85	
Max Suppy Voltage	5.25 volts	
Max voltage allowed on inputs	Supply plus 0.5 volts	
Normal USB supply voltage	5 volts	
Quiescent Supply Current (approx)	14 mA	
Pull-Up Resistors (except when configured as analog input)	5.6 Kohms Note 2	
Pin Configured as Output		
Number of pins available (total)	16	8
Number of internally controlled available	16	8
Number of host-controlled available	16	8
Min output voltage when ON	4 Volts	
Max output voltage when OFF	0.75 V	
Max output current per pin	30 mA	100 mA on pins 8,9,10
Max output current per board	500mA	
Pin Configured as Analog		
Number of pins available	48 (Note 1)	8
Number of axes	8	
Analog Resolution	12 Bits	
Input Impedance	10 Mohms	
Input voltage swing	GND to Supply (5V)	

Pin Configured as Quadrature		
Number of pins available	12 pairs	4 pairs
Number of axes	3 (X, Y, Z)	
Input DC requirements	Source drives low. Pulled up via 5.6 Kohms	
Encoder type	2 bit gray code	
Host interface resolution	16 bits	
Pin configured as switch		
Number of pins available	50	8
Input DC requirements	Source drives low. Pulled up via 5.6 Kohms	
Number of pins with ESD Protection	24	0
De-Bounce lock-out of same pin	60 ms	
Max latency	10 ms	
Max number of macros	30	
Max number of total keys in macros	84	

Notes:
1: 24 pins have ESD protection and can only be used as analog in certain cases.
2: When switch configured as Down = Clear, the resistor becomes a pull-down.

Connectors

The open pin headers can be used with a variety of 0.1 in (2.54mm) connectors.
Example housings are Molex p/n 22-55-2401 (40 way) or 22-55-2101 (10 way)
Crimp terminals for these are 16-02-0102.
A partial-length connector can be used in any location, for example 10-way, 12-way etc will fit.
Note a selection of ready-made harnesses are available from Ultimarc.com

Upgrading Firmware

The “Upgrade Firmware” button under “file” causes the board to change to a different USB device and reset. Unlike the U-HID itself, this device does require a custom driver.

After resetting, the driver should install automatically, then a program called “uupload.exe” should start. The firmware is upgraded using this program.

If the uupload program does not detect the board, it is likely the driver install failed. Do the following:

- Check there is a new USB device in Device Manager, by expanding the “USB Controllers” entry. This may have an exclamation mark against it.
- Right-click and select “upgrade driver software”.
- Follow the sequence and point the driver search to the “drivers” folder which is contained in the U-Config install folder.
- The driver should install.
- Re-run the uupload.exe program and select the firmware.

Appendix: U-HID-G

The U-HID-G has a three-axis accelerometer for sensing gravity and motion acceleration, plus it has a complete U-HID-NANO expansion port, configurable for buttons, keystrokes, further analog or quadrature.
It can serve as a gaming interface for flight controls, steering wheels, joysticks, pinball nudge, or whatever else you can think of that can be accomplished by sensing the acceleration that is reported by the U-HID-G to your computer. In addition to gaming, the U-HID-G may provide motion feedback for musical instruments, exercise equipment, alarm systems, and just about anything else that works based on motion.
Motion sensing is available using two methods. The first is analog sensing. In this mode, the motion sensing appears as analog gamepad axes, X, Y and Z, to enable easy usage with standard PC applications which recognize game controllers. The sensing is fully adjustable using the calibration screen in the U-Config program.

The second mode is digital, in which a "nudge" of the sensor sends a keystroke or button signal to the PC. For each of the six nudge directions, a key or button can be configured. This mode is used, for example in pinball nudge detection. The sensitivity can be adjusted using the U-Config calibration screen, and you can set the detection so that a key or button is "pressed" for different orientations of the board if you have an application for this.

Configuration

The 8-way port on the board is configured in the same way as the U-HID Nano, for keys, buttons, analog or quadrature inputs. Refer to the relevant configuration sections elsewhere in this document.

The on-board accelerometer occupies the first three analog axes and these can have calibration scale factor and offset applied in the normal way using the U-Config program. Axis 1 is X, axis 2 is Y and axis 3 is Z (vertical bar).
The board is shipped with a default calibration but for precision use, you will need to adjust the settings. Do this as follows:

1. Open Control Panel, Game Controllers.

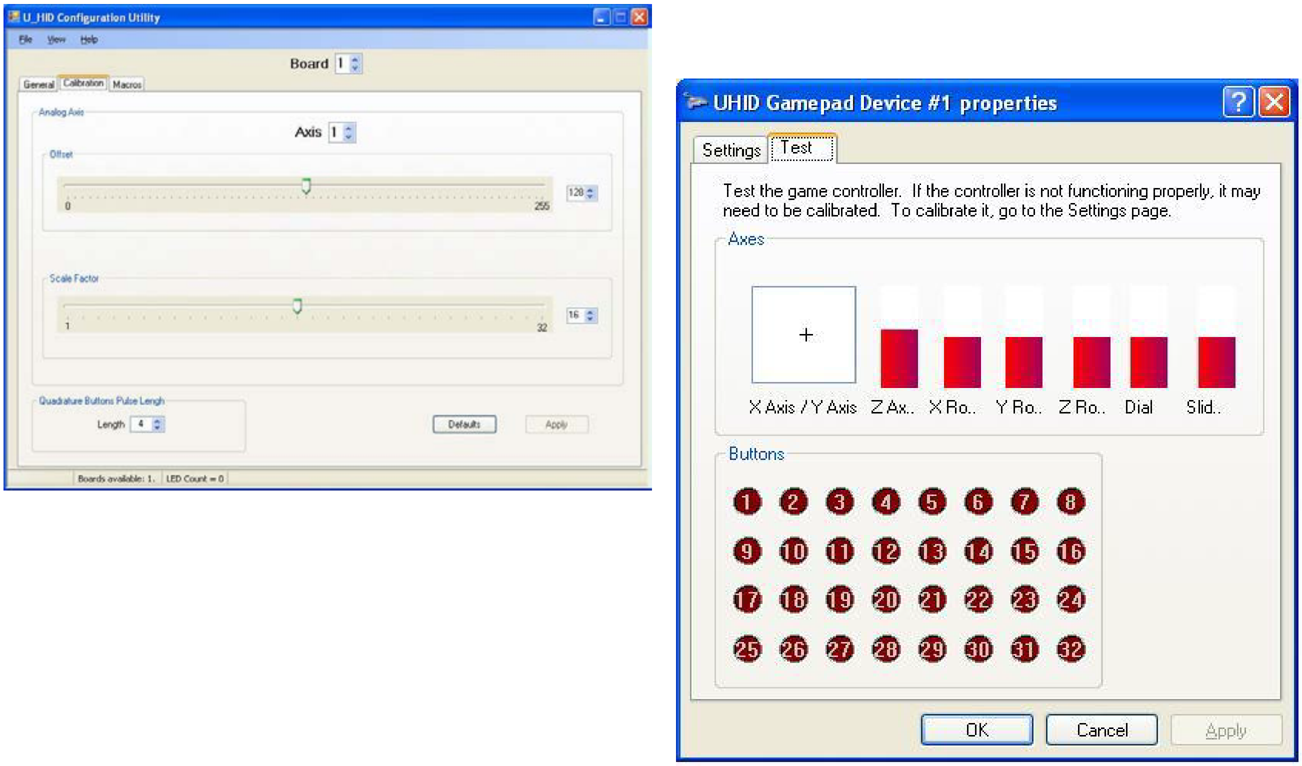
2. Click on the U-HID device to show the gamepad test window. This displays a crosshair for X and Y , and vertical bar (the left bar) for Z.

3. Place the board in the orientation it will be mounted.

4. Open U-Config and go to the Calibration tab.

5. Adjust the Scale Factor slider to increase or decrease the sensitivity (moving left increases the sensitivity). Experiment with moving the board.

6. Adjust the Offset slider to center the crosshair or vertical bar. (unless you need an offset applied for a special application in which case adjust as required). You will need to click "apply" each time the setting is changed.



U_HID Configuration Utility

Board 1

Axis 1

Offset 128

Scale Factor 16

Quadrature Buttons Pulse Lengh: Length 4

Boards available: 1. LED Count = 0

UHID Gamepad Device #1 properties

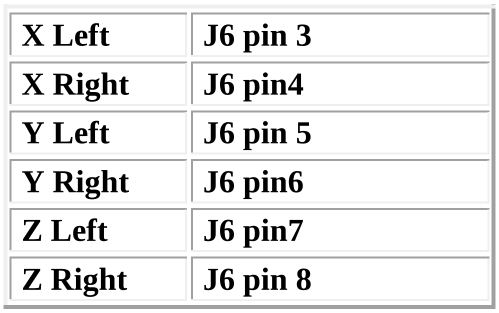
Test the game controller. If the controller is not functioning properly, it may need to be calibrated. To calibrate it, go to the Settings page.

Windows used for adjusting sensitivity

Configuring Button/Key (nudge) Mode

You can configure "virtual switches" for sending signals to the PC when the board is moved in each of the six directions. These are configured in the J6

section of the U-Config program. Note that J6 does not exist as a physical connector on the board, this is for configuring these "virtual switches" only. Switches and buttons are configured in the same way as a "real" switch, so that pulse, normally open or closed etc can be set.
Each directional nudge is assigned as follows:



X Left	**J6 pin 3**
X Right	**J6 pin4**
Y Left	**J6 pin 5**
Y Right	**J6 pin6**
Z Left	**J6 pin7**
Z Right	**J6 pin 8**

If any virtual switches are configured, the board switches to "auto-center" mode. This renders the Offset slider on the calibration screen inoperable. The centering of the sensor is set each time the board is powered up, or U-Config downloads a new configuration.

After configuring virtual switches, you will need to set the sensitivity. This is done using the U-Config program, calibration tab. Open this tab, and adjust the scale factor to the required setting. Moving the slider to the left increases the sensitivity. If you increase the sensitivity to a high level, the keys or buttons will be pressed when the board is oriented in specific directions. Lowering the sensitivity will create a button press on a "nudge" of the board. You can monitor the setting using Control Panel, Game Controllers, and open the device test window.

U-HID-G Technical Specification (In addition to main U-HID Tech Specs)

	U-HID-G
W	24
L	35
H	10
Mounting Hole Dia	3
Mounting Hole rectangle	27 x 18
Accel Analog Resolution	12 Bits
Host interface resolution	16 bits
Acceleration	+- 3.5g