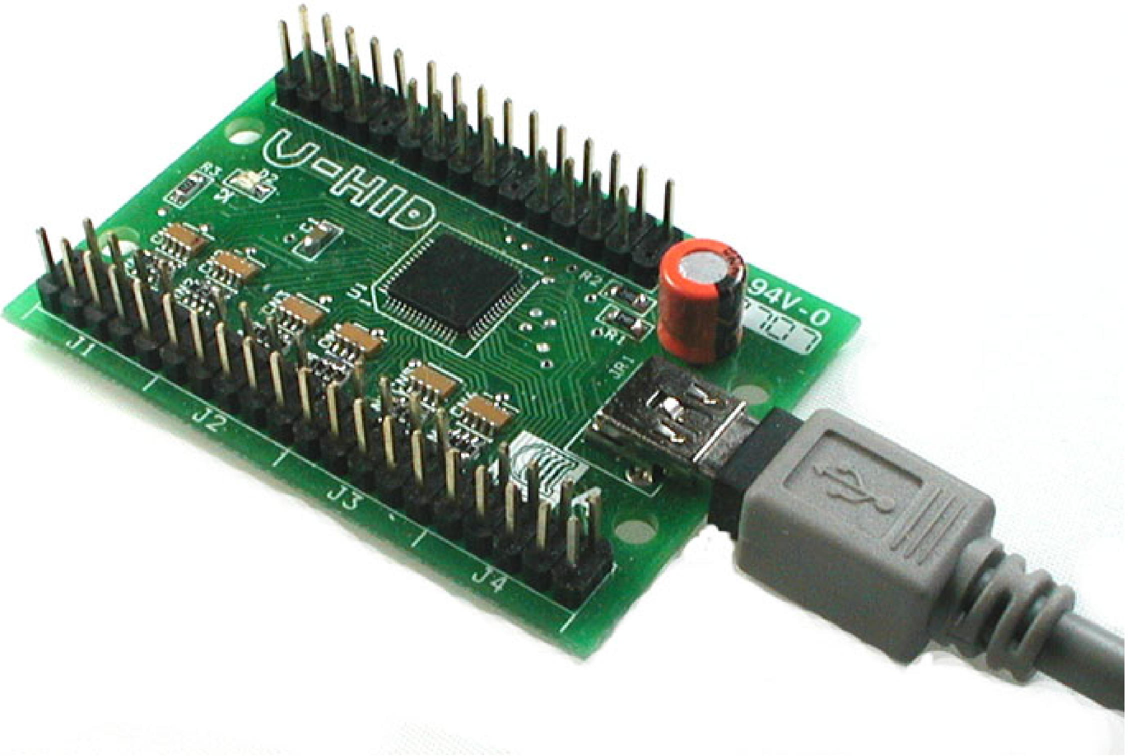
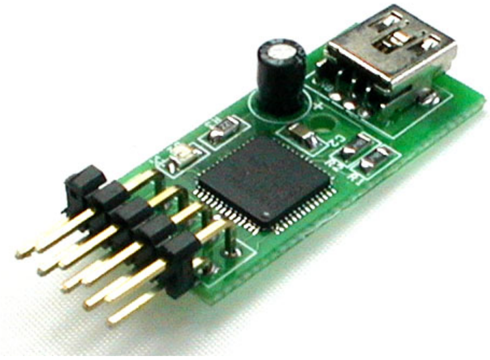


Pac-LED64   PacDrive   U-HID LEDs USBBUTTON

Software Development Kit



1. Introduction

The Ultimarc PacDrive, PacLED64 & U-HID SDK is a collection of source code examples for controlling the PacDrive, PacLED64, USBButton & U-HID LED controller hardware by Ultimarc.

For more information on these devices please visit:

- PacDrive - http://www.ultimarc.com/pacdrive.html
- U-HID - http://www.u-hid.com
- USBButton - http://www.usbbutton.com

It contains source code projects in the following languages:

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

Note regarding Device IDs

This SDK covers a number of different products and allows all to be used in combination with each other.

So you must do the following:

1. Call PacInitialize() to get the number of devices

2. Iterate through them and call PacGetDeviceType to get the device type (DEVICETYPE_USBBUTTON is 5)

3. The index for the device is then the id you use to call into the DLL

2. Common Functions (all boards)

int PacSetCallbacks((void)(int id) attach, (void)(int id) remove)

- Set the callbacks to receive device attach and removal messages
- Id is the id of the device being attached or removed
- When a device is removed all device id's below it will be moved up a position

int PacInitialize()

- Initialize all PacDrive, PacLED64 and U-HID Devices
- Returns the number of PacDrives', PacLED64s' and U-HIDs' on the PC or '0' if none are found

void PacShutdown()

- Shutdown all PacDrive, PacLED64 and U-HID Devices
- No return value

int PacGetDeviceType(int id)

- Returns the Device Type of the device specified by id
- 0 is Unknown, 1 is PacDrive, 2 is U-HID and 3 is PacLED64

int PacGetVersionNumber(int id)

- Returns the Version Number of the device specified by id

3. PacDrive / U-HID LED Functions

bool PacSetLEDStates(int id, ushort data)

- Sets LED states on the PacDrive or U-HID specified by id
- Each bit represents an LED on or off (Eg. 0xFFFF = all on, 0x0 = all off, 0xAAAA = every second LED on)
- Returns true for success and false for failure

bool PacSetLEDState(int id, int port, bool state)

- Sets a single LED state on the PacDrive or U-HID specified by id
- Port is the LED number
- State is the LED value (true or false)

4. PacLED64 Functions

Introduction:

The PacLED64 can be controlled in direct or script mode. When the board is powered up, it will check if a script is present in flash and if so, will begin to run it. This will happen on power-up (USB and LED power required) and before the host has initialized the USB bus.
Scripts are repeated continuously.
When sending commands to be stored in a script, it is usual to start the script with a SetLEDFadeTime and a SetScriptStepDelay otherwise the script will run in an unpredictable way.
When a first command is sent from the host, the script stops running and the command is processed.
Additionally, when the first command is sent, the Fade Time is set to zero.

The following commands can be sent in script or direct mode:

bool Pac64SetLEDIntensity(int id, int port, byte intensity)

- Sets an LED's intensity on the PacLED64 specified by id
- Port is the LED number or -1 for all
- Intensity is a value from 0 to 255 (0 being off and 255 being full intensity)

Note: In many applications this is the only command required, with the possible addition of one initial "*SetLEDFadeTime*". The other commands are mainly provided for compact script sizes.

bool Pac64SetLEDStates(int id, int group, byte data)

- Sets LED states on the PacLED64 specified by *id*
- *Group* specifies the LED group as follows:

Group Number	LED Numbers
1	1-8
2	9-16
3	17-24
4	25-32
5	33-40
6	41-48
7	49-56
8	57-64

- *Data* specifies LED states. Each bit represents an LED state (Eg. 0xFF = all enabled, 0x0 = all off, 0xAA = every second LED enabled)
- Returns true for success and false for failure
- NOTE: “Enabled” means “set to its previously stored brightness setting from earlier *SetLEDIntensity* command.

bool Pac64SetLEDState(int id, int group, int port, bool state)

- Sets a single LED state on the PacLED64 specified by *id*
- *Port* is the LED number within a group
- *Group* specifies the LED group as follows:

Group Number	LED Numbers
1	1-8
2	9-16
3	17-24
4	25-32
5	33-40
6	41-48
7	49-56
8	57-64

- *State* is the LED value (true = enabled, or false = off)
- NOTE: “Enabled” means “set to its previously stored brightness setting from earlier *SetLEDIntensity* command.

bool Pac64SetLEDStatesRandom(int id)

- Sets all LED’s to random states on the PacLED64 specified by id

bool Pac64SetLEDFadeTime(int id, byte fadeTime)

- Sets the LED’s fade time on the PacLED64 specified by id. This value is remembered by the board and used for all subsequent LED commands.

bool Pac64SetScriptStepDelay(int id, byte stepDelay)

- Sets the script step delay on the PacLED64 specified by id

bool Pac64SetLEDFlashSpeeds(int id, byte flashSpeed)

- Sets all LED’s flash speed on the PacLED64 specified by id
- FlashSpeed is the speed of the flash (0 = always on, 1 = 2 secs, 2 = 1 sec, 3 = 0.5 sec)

bool Pac64SetLEDFlashSpeed(int id, int port, byte flashSpeed)

- Sets one LED's flash speed on the PacLED64 specified by id
- Port is the LED number
- FlashSpeed is the speed of the flash (0 = always on, 1 = 2 secs, 2 = 1 sec, 3 = 0.5 sec)

The following commands are direct-mode only and cannot be incorporated in a script:

bool Pac64StartScriptRecording(int id)

- Starts recording a script on the PacLED64 specified by id. All subsequently-sent commands will be stored in the script and then executed at power-on. Max script length is 32 steps.

bool Pac64StopScriptRecording(int id)

- Stops recording a script on the PacLED64 specified by id. At the end of the script, the device inserts a "goto start" so the script is looped when executed at power-on.

bool Pac64RunScript(int id)

- Runs the script on the PacLED64 specified by id (This also occurs at power-on).

bool Pac64ClearFlash(int id)

- Clears the flash on the PacLED64 specified by id. Any saved script will no longer be run at power-on.

bool Pac64SetDeviceId(int id, int newId)

- Sets the Device Id on the PacLED64 specified by id (1 to 4).

bool Pac64SetLEDIntensities(int id, byte data)*

- Sets all LEDs to a brightness level specified in a 64 byte array.
- NOTE: This is a long command and requires a delay of 20ms before sending any further command. This command should only be used when more than half the LEDs need to be changed. Do not use for setting a small number of LEDs, use *SetLEDIntensity* for this.

5. U-HID Functions

*bool PacProgramUHid(int id, char *sFilePath)*

- Programs a U-HID device using a .raw file exported from U-Config
- Returns true for success and false for failure

6. ServoStik Functions

bool PacSetServoStik4Way();

- Sets all attached ServoStik devices to 4 way

bool PacSetServoStik8Way();

- Sets all attached ServoStik devices to 8 way

7. USBButton Functions

bool USBButtonConfigurePermanent(char data)*

- Sends all configuration data to button and stores permanently
- Pointer to a 62-byte array

Returns true for success and false for failure

bool USBButtonConfigureTemporary(char data)*

- Sends all configuration data to button and stores in RAM only
- Pointer to a 62-byte array

Returns true for success and false for failure

bool USBButtonConfigureColor(char red, char green, char blue)

- Sets color of button (unpressed state)

Returns true for success and false for failure

NOTE: The format of data for the above commands is contained within a separate document which will be supplied to volume purchasers of the product on request.

BOOL USBButtonGetState();

Returns state of the button

8. Examples in C++

- Set every second LED on the first Pacdrive device

```
int deviceCount = PacInitialize();
PacSetLEDStates(0, 0xAAAA);
PacShutdown();
```

- Set the second LED on

```
PacSetLEDState(0, 1, true);
```

- Output the full device path of the first device

```
char sDevicePath[256];
PacGetDevicePath(0, sDevicePath);
printf(“%s\n”, sDevicePath);
```

- Program the U-HID

```
PacProgramUHid(0, "C:\Settings.raw");
```

9. Release History

- 10-3-2012 – 1.6 – Added support for Flash Speed
- 19-9-2010 – 1.5 – Added support for the PacLED64
- 6-1-2010 – 1.4 – Added PacSetLEDState function
- 2-6-2009 – 1.3 – Added PacProgramUHid function
- 1-7-2008 – 1.2 – Added support for U-HID
- 17-10-2007 – 1.1 – Bug fix
- 4-9-2007 – 1.01 – Minor fixes
- 3-9-2007 – 1.0 – First Release

10. Contact

Andy Warne (PacDrive, PacLED64 & U-HID Manufacturer)

andy@ultimarc.com / www.ultimarc.com

Ben Baker (PacDrive, PacLED64 & U-HID SDK Developer)

benbaker@headsoft.com.au / www.headsoft.com.au