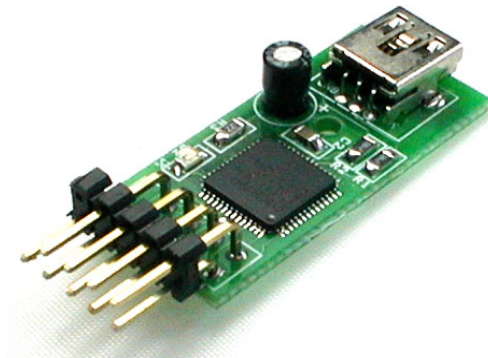


**PacDrive • U-HID • Blue-HID
Nano-LED • PacLED64 • I-Pac Ultimate I/O
ServoStik • USB Button**

Software Development Kit



1. Introduction.....	4
2. Common Functions (all boards).....	5
int PacSetCallbacks((void)(int id) attach, (void)(int id) remove).....	5
int PacInitialize().....	5
void PacShutdown().....	5
int PacGetDeviceType(int id)	5
int PacGetVendorId(int id)	5
int PacProductId(int id).....	5
int PacGetVersionNumber(int id).....	5
int PacGetVendorName(int id)	6
int PacGetProductName(int id).....	6
int PacGetSerialNumber(int id)	6
int PacGetDevicePath(int id)	6
3. PacDrive / U-HID / Blue-HID LED Functions.....	7
bool PacSetLEDStates(int id, ushort data).....	7
bool PacSetLEDState(int id, int port, bool state).....	7
4. PacLED64 / Nano-LED / I-Pac Ultimate I/O Functions	8
Introduction:.....	8
bool Pac64SetLEDIntensities(int id, byte *data).....	8
bool Pac64SetLEDIntensity(int id, int port, byte intensity)	8
bool Pac64SetLEDStates(int id, int group, byte data)	9
bool Pac64SetLEDState(int id, int group, int port, bool state)	9
bool Pac64SetLEDStatesRandom(int id).....	10
bool Pac64SetLEDFadeTime(int id, byte fadeTime)	10
bool Pac64SetScriptStepDelay(int id, byte stepDelay)	10
bool Pac64SetLEDFlashSpeeds(int id, byte flashSpeed)	10
bool Pac64SetLEDFlashSpeed(int id, int port, byte flashSpeed)	10
bool Pac64StartScriptRecording(int id).....	10
bool Pac64StopScriptRecording(int id)	10
bool Pac64RunScript(int id)	11
bool Pac64ClearFlash(int id)	11
bool Pac64SetDeviceId(int id, int newId).....	11
5. U-HID Functions	12
bool PacProgramUHid(int id, char *sFilePath)	12
6. ServoStik Functions	13
bool PacSetServoStik4Way().....	13
bool PacSetServoStik8Way().....	13
7. USB Button Functions	14
bool USBButtonConfigurePermanent(int id, byte *data).....	14
bool USBButtonConfigureTemporary(int id, byte *data)	14
bool USBButtonConfigureColor(int id, byte red, byte green, byte blue).....	14
bool USBButtonGetState(int id, bool *state).....	14



8. Examples in C++	15
9. Release History	16
10. Contact	16
Andy Warne (Hardware Manufacturer).....	16
Ben Baker (PacDrive SDK Developer)	16

1. Introduction

The Ultimarc PacDrive SDK is a collection of source code examples for controlling the PacDrive, U-HID, Blue-HID, Nano-LED, PacLED64, I-Pac Ultimate I/O, ServoStik & USB Button 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/home/index.php>
- Blue-HID - <http://www.u-hid.com/home/index.php>
- Nano-LED - <https://www.ultimarc.com/nanoled.html>
- PacLED64 - <https://www.ultimarc.com/pacled64.html>
- I-Pac Ultimate I/O - <https://www.ultimarc.com/ipacuio.html>
- ServoStik - <https://www.ultimarc.com/servostik.html>
- USB Button - <http://www.usbbutton.com/>

It contains source code projects in the following languages:

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

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 devices
- Returns the number of devices on the PC or 0 if none are found

void PacShutdown()

- Shutdown all devices
- No return value

int PacGetDeviceType(int id)

- Returns the Device Type of the device specified by id

Device Type	Name	LED Channels	Brightness Levels	RGB LEDs
0	Unknown			
1	PacDrive	16		
2	U-HID	16		
3	Blue-HID	16		
4	Nano-LED	60	256	20
5	PacLED64	64	256	21
6	I-Pac Ultimate I/O	96	256	32
7	ServoStik			
8	USB Button			

int PacGetVendorId(int id)

- Returns the Vendor Id of the device specified by id

int PacProductId(int id)

- Returns the Product Id of the device specified by id

int PacGetVersionNumber(int id)

- Returns the Version Number of the device specified by id

int PacGetVendorName(int id)

- Returns the Vendor Name of the device specified by id

int PacGetProductName(int id)

- Returns the Product Name of the device specified by id

int PacGetSerialNumber(int id)

- Returns the Serial Number of the device specified by id

int PacGetDevicePath(int id)

- Returns the Device Path of the device specified by id

3. PacDrive / U-HID / Blue-HID LED Functions

bool PacSetLEDStates(int id, ushort data)

- Sets LED states on the device 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 device specified by id
- Port is the LED number
- State is the LED value (true or false)

4. PacLED64 / Nano-LED / I-Pac Ultimate I/O 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 Pac64SetLEDIntensities(int id, byte *data)

- Sets the LED intensities on the device specified by id
- Data is an array of byte data (60 bytes for Nano-LED, 64 bytes for PacLED64 and 96 bytes for I-Pac Ultimate I/O)
- Byte values are from 0 to 255 (0 being off and 255 being full intensity)

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 instead.

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

- Sets an LED's intensity on the device specified by id
- Port is the LED number
- 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 device 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
9	65 - 72
10	73 - 80
11	81 - 88
12	89 - 96

- *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 device 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
9	65 - 72
10	73 - 80

11	81 - 88
12	89 - 96

- *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 device specified by id

bool Pac64SetLEDFadeTime(int id, byte fadeTime)

- Sets the LED’s fade time on the device 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 device specified by id

bool Pac64SetLEDFlashSpeeds(int id, byte flashSpeed)

- Sets all LED’s flash speed on the device 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 device 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 device 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).

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 a ServoStik to 4-Way mode
- Returns true for success and false for failure

bool PacSetServoStik8Way()

- Sets a ServoStik to 8-Way mode
- Returns true for success and false for failure

7. USB Button Functions

bool USBButtonConfigurePermanent(int id, byte *data)

- Programs a USB Button device and stores it permanently in memory
- Returns true for success and false for failure

bool USBButtonConfigureTemporary(int id, byte *data)

- Programs a USB Button device and stores it temporarily in memory
- Returns true for success and false for failure

bool USBButtonConfigureColor(int id, byte red, byte green, byte blue)

- Set the color of the USB Button (RGB color values)
- Returns true for success and false for failure

bool USBButtonGetState(int id, bool *state)

- Get the current state of the USB Button
- Returns true for success and false for failure

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

- 26-5-2020 – 2.4 – Improved Shutdown
- 10-1-2020 – 2.3 – Added support for new Ultimate I/O PID's
- 26-7-2018 – 2.2 – Hotplug fix
- 2-4-2018 – 2.1 – Ultimate I/O
- 18-8-2017 – 2.0 – Updated documentation
- 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 (Hardware Manufacturer)

andy@ultimarc.com / www.ultimarc.com

Ben Baker (PacDrive SDK Developer)

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