felis / USB_Host_Shield_2.0

Revision 2.0 of USB Host Library for Arduino. http://www.circuitsathome.com

1,123 commits **7** branches **5** releases **22** contributors

Lauszus Merge pull request #232 from YuuichiAkagawa/pr_usbh_midi_031 ··· Latest commit 2295571 on 27 Apr

examples	Fix type mismatch warning	a month ago
.gitattributes	Removed content from merge conflict	a year ago
.gitignore	commit	3 years ago
.gitmodules	Add RTClib	3 years ago
.travis.yml	Warnings in the MIDI library were fixed upstream: FortySevenEffects/a…	a month ago
BTD.cpp	Remove execute permission from source files	a month ago
BTD.h	Remove execute permission from source files	a month ago
BTHID.cpp	Compatibility with Arduino's core HID / Mouse / Keyboard libraries	5 months ago
BTHID.h	Fixed typo	a month ago
PS3BT.cpp	Merge pull request #215 from sieren/fix-new-rbl-sdk	2 months ago
PS3BT.h	Replaced all double variables with float	8 months ago
PS3Enums.h	Can now set the rumble and light on the PS4 controller	2 years ago
PS3USB.cpp	Remove execute permission from source files	a month ago
PS3USB.h	Compatibility with Arduino's core HID / Mouse / Keyboard libraries	5 months ago
PS4BT.h	manual merge	a year ago
PS4Parser.cpp	Put DPADEnum into the source file, so it does not conflict with the o…	3 months ago
PS4Parser.h	Put DPADEnum into the source file, so it does not conflict with the o…	3 months ago
PS4USB.h	Compatibility with Arduino's core HID / Mouse / Keyboard libraries	5 months ago
PSBuzz.cpp	Compatibility with Arduino's core HID / Mouse / Keyboard libraries	5 months ago
PSBuzz.h	Compatibility with Arduino's core HID / Mouse / Keyboard libraries	5 months ago
README.md	update MIDI driver v0.3.1	a month ago
SPP.cpp	Check HCI Handle using function in BluetoothService class	2 years ago
SPP.h	Print::write not on RBL	2 months ago
Usb.cpp	Don't set epAttribs to 0, as this will set bmNakPower to 0 as well. I…	7 months ago
Usb.h	Put avrpins include back at original order	a year ago
UsbCore.h	Add delay between multiple calls to dispatchpkt inside InTransfer	11 months ago
Wii.cpp	Remove execute permission from source files	a month ago
Wii.h	Remove execute permission from source files	a month ago
WiiCameraReadme.md	CRLF -> LF	a year ago
XBOXOLD.cpp	Don't set epAttribs to 0, as this will set bmNakPower to 0 as well. I…	7 months ago
XBOXOLD.h	Compatibility with Arduino's core HID / Mouse / Keyboard libraries	5 months ago
XBOXONE.cpp	Don't set epAttribs to 0, as this will set bmNakPower to 0 as well. I…	7 months ago
XBOXONE.h	Replace boolean with bool	a year ago

XBOXRECV.cpp	Exclude specific enum defines and cast	2 months ago
XBOXRECV.h	Merge branch 'xxxajk'	a year ago
XBOXUSB.cpp	Exclude specific enum defines and cast	2 months ago
XBOXUSB.h	Compatibility with Arduino's core HID / Mouse / Keyboard libraries	5 months ago
address.h	Don't set epAttribs to 0, as this will set bmNakPower to 0 as well. I…	7 months ago
adk.cpp	Don't set epAttribs to 0, as this will set bmNakPower to 0 as well. I…	7 months ago
adk.h	all sources to LF	a year ago
avrpins.h	Added support for STM32F446 Nucleo board	8 months ago
cdc_XR21B1411.cpp	cdc_XR21B1411.cpp changed date in the header	a year ago
cdc_XR21B1411.h	Fixed: "warning: "XR_REG_ERROR_STATUS_BREAK" redefined"	a year ago
cdcacm.cpp	Don't set epAttribs to 0, as this will set bmNakPower to 0 as well. I…	7 months ago
cdcacm.h	Fixed "warning: no return statement in function returning non-void"	a year ago
cdcftdi.cpp	Add possibility to set a custom FTDI PID, to be able to connect to cu…	5 months ago
cdcftdi.h	Add possibility to set a custom FTDI PID, to be able to connect to cu…	5 months ago
cdcprolific.cpp	Merge #110	a year ago
cdcprolific.h	Merge #110	a year ago
confdescparser.h	Janitorial tab removal	a year ago
controllerEnums.h	Exclude specific enum defines and cast	2 months ago
gpl2.txt	all sources to LF	a year ago
hexdump.h	Fix all demos, fix all wrong uses of virtual keyword, some spacing cl…	2 years ago
hidboot.cpp	Compatibility with Arduino's core HID / Mouse / Keyboard libraries	5 months ago
hidboot.h	Some corrections as agreed with **@Lauszus**. Added flag bRptProtoEnable …	4 months ago
hidcomposite.cpp	Fixed tab/space and added comments	2 months ago
hidcomposite.h	Fixed tab/space and added comments	2 months ago
hidescriptorparser.cpp	Compatibility with Arduino's core HID / Mouse / Keyboard libraries	5 months ago
hidescriptorparser.h	Compatibility with Arduino's core HID / Mouse / Keyboard libraries	5 months ago
hiduniversal.cpp	Compatibility with Arduino's core HID / Mouse / Keyboard libraries	5 months ago
hiduniversal.h	Compatibility with Arduino's core HID / Mouse / Keyboard libraries	5 months ago
hidusagestr.h	Janitorial tab removal	a year ago
hidusagetitlearrays.cpp	Janitorial tab removal	a year ago
keywords.txt	Corrected spelling mistake "Purble" to "Purple"	a year ago
library.json	Added SPL to supported frameworks	7 months ago
library.properties	Release version 1.2.1	6 months ago
macros.h	Janitorial tab removal	a year ago
masstorage.cpp	Don't set epAttribs to 0, as this will set bmNakPower to 0 as well. I…	7 months ago
masstorage.h	Janitorial tab removal	a year ago
max3421e.h	Janitorial tab removal	a year ago
max_LCD.cpp	Replace deprecated byte variable with uint8_t	8 months ago
max_LCD.h	Janitorial tab removal	a year ago
message.cpp	all sources to LF	a year ago
message.h	all sources to LF	a year ago
parsetools.cpp	all sources to LF	a year ago
parsetools.h	all sources to LF	a year ago
printhex.h	Janitorial, whitespace fixes. No new code.	3 years ago
settings.h	Fix warnings on Arduino Due	a month ago

sink_parser.h	Janitorial, whitespace fixes. No new code.	3 years ago
usb_ch9.h	Janitorial tab removal	a year ago
usbh_midi.cpp	Fix unused variable warning	a month ago
usbh_midi.h	update MIDI driver v0.3.1	a month ago
usbhid.cpp	Compatibility with Arduino's core HID / Mouse / Keyboard libraries	5 months ago
usbhid.h	Compatibility with Arduino's core HID / Mouse / Keyboard libraries	5 months ago
usbhost.h	Added support for STM32F446 Nucleo board	8 months ago
usbhub.cpp	Don't set epAttribs to 0, as this will set bmNakPower to 0 as well. I…	7 months ago
usbhub.h	Janitorial tab removal	a year ago
version_helper.h	Don't use deprecated prog_ macro	a year ago
xboxEnums.h	Can now turn LED off by calling setLedOn(OFF) for all controllers	2 years ago

README.md

USB Host Library Rev.2.0

The code is released under the GNU General Public License.

build passing

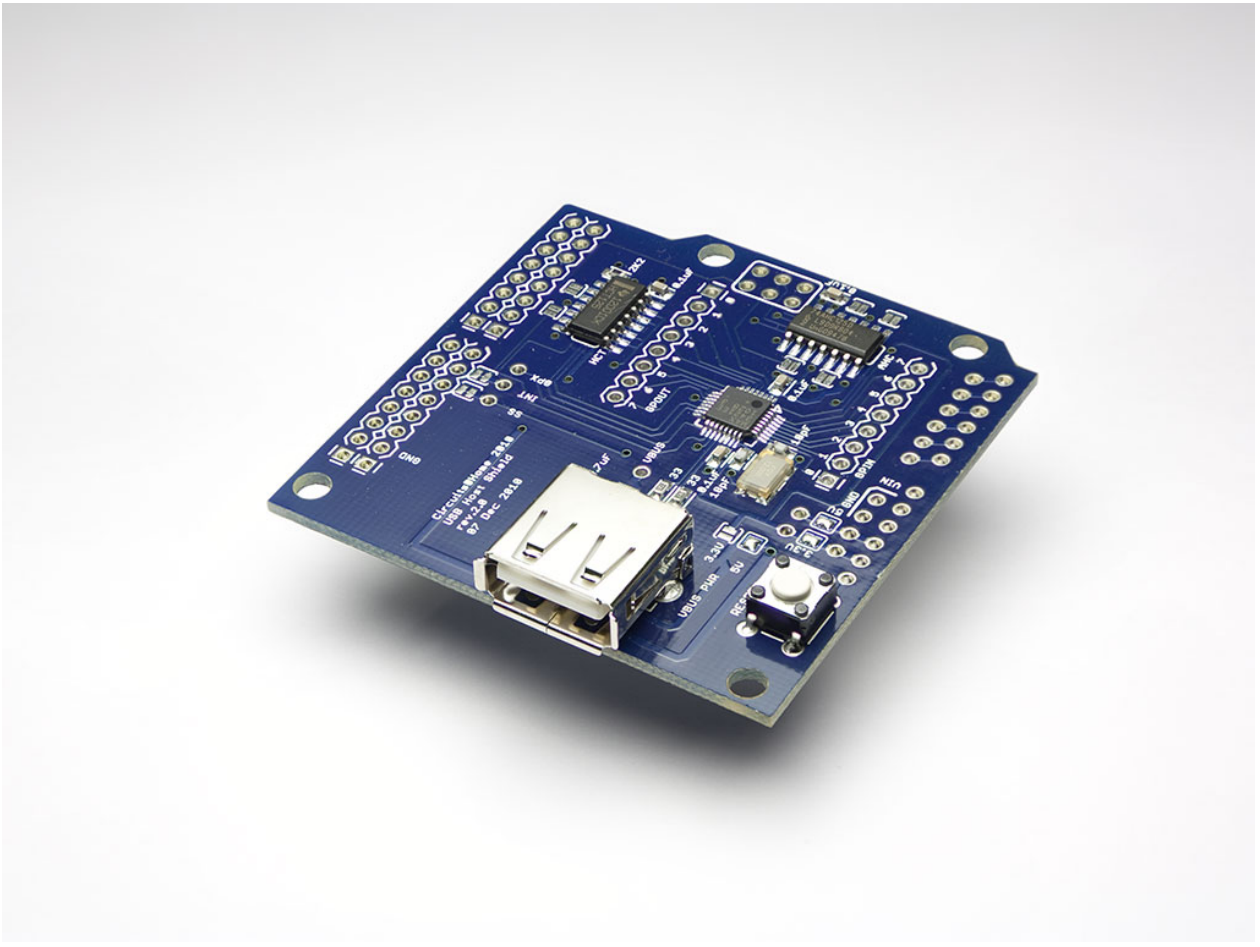
Summary

This is Revision 2.0 of MAX3421E-based USB Host Shield Library for AVR's.

Project main web site is: http://www.circuitsathome.com.

Some information can also be found at: http://blog.tkjelectronics.dk/.

The shield can be purchased at the main site: http://www.circuitsathome.com/products-page/arduino-shields or from TKJ Electronics: http://shop.tkjelectronics.dk/product_info.php?products_id=43.



For more information about the hardware see the Hardware Manual.

Developed By

- **Oleg Mazurov, Circuits@Home** - mazurov@circuitsathome.com
- **Alexei Glushchenko, Circuits@Home** - alex-gl@mail.ru
  - Developers of the USB Core, HID, FTDI, ADK, ACM, and PL2303 libraries
- **Kristian Lauszus, TKJ Electronics** - kristianl@tkjelectronics.com
  - Developer of the BTD, BTHID, SPP, PS4, PS3, Wii, Xbox, and PSBuzz libraries
- **Andrew Kroll** - xxxajk@gmail.com
  - Major contributor to mass storage code
- **guruthree**
  - Xbox ONE controller support
- **Yuuichi Akagawa** - @YuuichiAkagawa
  - Developer of the MIDI library

Donate

Help yourself by helping us support you! Many thousands of hours have been spent developing the USB Host Shield library. Since you find it useful, please consider donating via the button below. Donations will allow us to support you by ensuring hardware that you have can be acquired in order to add support for your microcontroller board.

Donate

Table of Contents

How to include the library

Arduino Library Manager

First install Arduino IDE version 1.6.2 or newer, then simply use the Arduino Library Manager to install the library.

Please see the following page for instructions: http://www.arduino.cc/en/Guide/Libraries#toc3.

Manual installation

First download the library by clicking on the following link: https://github.com/felis/USB_Host_Shield_2.0/archive/master.zip.

Then uncompress the zip folder and rename the directory to "USB_Host_Shield_20", as any special characters are not supported by the Arduino IDE.

Now open up the Arduino IDE and open "File>Preferences". There you will see the location of your sketchbook. Open that directory and create a directory called "libraries" inside that directory. Now move the "USB_Host_Shield_20" directory to the "libraries" directory.

The final structure should look like this:

- Arduino/
    - libraries/
        - USB_Host_Shield_20/

Now quit the Arduino IDE and reopen it.

Now you should be able to go open all the examples codes by navigating to "File>Examples>USB_Host_Shield_20" and then select the example you will like to open.

For more information visit the following sites: http://arduino.cc/en/Guide/Libraries and https://learn.adafruit.com/adafruit-all-about-arduino-libraries-install-use.

How to use the library

Documentation

Documentation for the library can be found at the following link: http://felis.github.com/USB_Host_Shield_2.0/.

Enable debugging

By default serial debugging is disabled. To turn it on simply change `ENABLE_UHS_DEBUGGING` to 1 in settings.h like so:

```
#define ENABLE_UHS_DEBUGGING 1
```

Boards

Currently the following boards are supported by the library:

- All official Arduino AVR boards (Uno, Duemilanove, Mega, Mega 2560, Mega ADK, Leonardo etc.)
- Arduino Due, Intel Galileo, Intel Galileo 2, and Intel Edison
  - Note that the Intel Galileo uses pin 2 and 3 as INT and SS pin respectively by default, so some modifications to the shield are needed. See the "Interface modifications" section in the hardware manual for more information.
  - Note native USB host is not supported on any of these platforms. You will have to use the shield for now.
- Teensy (Teensy++ 1.0, Teensy 2.0, Teensy++ 2.0, Teensy 3.x, and Teensy LC)
  - Note if you are using the Teensy 3.x you should download this SPI library as well: https://github.com/xxxajk/spi4teensy3. You should then add `#include <spi4teensy3.h>` to your .ino file.
- Balanduino
- Sanguino
- Black Widdow
- RedBearLab nRF51822
- Digilent chipKIT
  - Please see: http://www.circuitsathome.com/mcu/usb/running-usb-host-code-on-digilent-chipkit-board.
- STM32F4
  - Currently the NUCLEO-F446RE is supported featuring the STM32F446. Take a look at the following example code: https://github.com/Lauszus/Nucleo_F446RE_USBHost.

The following boards need to be activated manually in settings.h:

- Arduino Mega ADK
  - If you are using Arduino 1.5.5 or newer there is no need to activate the Arduino Mega ADK manually
- Black Widdow

Simply set the corresponding value to 1 instead of 0.

Bluetooth libraries

The BTD library is a general purpose library for an ordinary Bluetooth dongle. This library make it easy to add support for different Bluetooth services like a PS3 or a Wii controller or SPP which is a virtual serial port via Bluetooth. Some different examples can be found in the example directory.

The BTD library also makes it possible to use multiple services at once, the following example sketch is an example of this: PS3SPP.ino.

BTHID library

The Bluetooth HID library allows you to connect HID devices via Bluetooth to the USB Host Shield.

Currently HID mice and keyboards are supported.

It uses the standard Boot protocol by default, but it is also able to use the Report protocol as well. You would simply have to call `setProtocolMode()` and then parse `HID_RPT_PROTOCOL` as an argument. You will then have to modify the parser for your device. See the example: BTHID.ino for more information.

The PS4 library also uses this class to handle all Bluetooth communication.

For information see the following blog post: http://blog.tkjelectronics.dk/2013/12/bluetooth-hid-devices-now-supported-by-the-usb-host-library/.

SPP library

SPP stands for "Serial Port Profile" and is a Bluetooth protocol that implements a virtual comport which allows you to send data back and forth from your computer/phone to your Arduino via Bluetooth. It has been tested successfully on Windows, Mac OS X, Linux, and Android.

Take a look at the SPP.ino example for more information.

More information can be found at these blog posts:

- http://www.circuitsathome.com/mcu/bluetooth-rfcommspp-service-support-for-usb-host-2-0-library-released
- http://blog.tkjelectronics.dk/2012/07/rfcommspp-library-for-arduino/

To implement the SPP protocol I used a Bluetooth sniffing tool called PacketLogger developed by Apple. It enables me to see the Bluetooth communication between my Mac and any device.

PS4 Library

The PS4BT library is split up into the PS4BT and the PS4USB library. These allow you to use the Sony PS4 controller via Bluetooth and USB.

The PS4BT.ino and PS4USB.ino examples shows how to easily read the buttons, joysticks, touchpad and IMU on the controller via Bluetooth and USB respectively. It is also possible to control the rumble and light on the controller and get the battery level.

Before you can use the PS4 controller via Bluetooth you will need to pair with it.

Simply create the PS4BT instance like so: `PS4BT PS4(&Btd, PAIR);` and then hold down the Share button and then hold down the PS without releasing the Share button. The PS4 controller will then start to blink rapidly indicating that it is in pairing mode.

It should then automatically pair the dongle with your controller. This only have to be done once.

For information see the following blog post: http://blog.tkjelectronics.dk/2014/01/ps4-controller-now-supported-by-the-usb-host-library/.

Also check out this excellent Wiki by Frank Zhao about the PS4 controller: http://eleccelerator.com/wiki/index.php?title=DualShock_4 and this Linux driver: https://github.com/chrippa/ds4drv.

PS3 Library

These libraries consist of the PS3BT and PS3USB. These libraries allows you to use a Dualshock 3, Navigation or a Motion controller with the USB Host Shield both via Bluetooth and USB.

In order to use your Playstation controller via Bluetooth you have to set the Bluetooth address of the dongle internally to your PS3 Controller. This can be achieved by first plugging in the Bluetooth dongle and wait a few seconds. Now plug in the controller via USB and wait until the LEDs start to flash. The library has now written the Bluetooth address of the dongle to the PS3 controller.

Finally simply plug in the Bluetooth dongle again and press PS on the PS3 controller. After a few seconds it should be connected to the dongle and ready to use.

Note: You will have to plug in the Bluetooth dongle before connecting the controller, as the library needs to read the address of the dongle. Alternatively you could set it in code like so: PS3BT.ino#L20.

For more information about the PS3 protocol see the official wiki: https://github.com/felis/USB_Host_Shield_2.0/wiki/PS3-Information.

Also take a look at the blog posts:

- http://blog.tkjelectronics.dk/2012/01/ps3-controller-bt-library-for-arduino/
- http://www.circuitsathome.com/mcu/sony-ps3-controller-support-added-to-usb-host-library
- http://www.circuitsathome.com/mcu/arduino/interfacing-ps3-controllers-via-usb

A special thanks go to the following people:

1. *Richard Ibbotson* who made this excellent guide: http://www.circuitsathome.com/mcu/ps3-and-wiimote-game-controllers-

on-the-arduino-host-shield-part
2. *Tomoyuki Tanaka* for releasing his code for the Arduino USB Host shield connected to the wiimote: http://www.circuitsathome.com/mcu/rc-car-controlled-by-wii-remote-on-arduino

Also a big thanks all the people behind these sites about the Motion controller:

- http://thp.io/2010/psmove/
- http://www.copenhagengamecollective.org/unimove/
- https://github.com/thp/psmoveapi
- http://code.google.com/p/moveonpc/

Xbox Libraries

The library supports both the original Xbox controller via USB and the Xbox 360 controller both via USB and wirelessly.

Xbox library

The XBOXOLD class implements support for the original Xbox controller via USB.

All the information are from the following sites:

- https://github.com/torvalds/linux/blob/master/Documentation/input/xpad.txt
- https://github.com/torvalds/linux/blob/master/drivers/input/joystick/xpad.c
- http://euc.jp/periphs/xbox-controller.ja.html
- https://github.com/Grumbel/xboxdrv/blob/master/PROTOCOL#L15

Xbox 360 Library

The library support one Xbox 360 via USB or up to four Xbox 360 controllers wirelessly by using a Xbox 360 wireless receiver.

To use it via USB use the XBOXUSB library or to use it wirelessly use the XBOXRECV library.

Note that a Wireless controller can NOT be used via USB!

Examples code can be found in the examples directory.

Also see the following blog posts:

- http://www.circuitsathome.com/mcu/xbox360-controller-support-added-to-usb-host-shield-2-0-library
- http://blog.tkjelectronics.dk/2012/07/xbox-360-controller-support-added-to-the-usb-host-library/
- http://blog.tkjelectronics.dk/2012/12/xbox-360-receiver-added-to-the-usb-host-library/

All the information regarding the Xbox 360 controller protocol are form these sites:

- http://tattiebogle.net/index.php/ProjectRoot/Xbox360Controller/UsbInfo
- http://tattiebogle.net/index.php/ProjectRoot/Xbox360Controller/WirelessUsbInfo
- https://github.com/Grumbel/xboxdrv/blob/master/PROTOCOL

Xbox ONE Library

An Xbox ONE controller is supported via USB in the XBOXONE class. It is heavily based on the 360 library above. In addition to cross referencing the above, information on the protocol was found at:

- https://github.com/quantus/xbox-one-controller-protocol
- https://github.com/torvalds/linux/blob/master/drivers/input/joystick/xpad.c
- https://github.com/kylelemons/xbox/blob/master/xbox.go

Wii library

The Wii library support the Wiimote, but also the Nunchuch and Motion Plus extensions via Bluetooth. The Wii U Pro Controller and Wii Balance Board are also supported via Bluetooth.

First you have to pair with the controller, this is done automatically by the library if you create the instance like so:

```
WII Wii(&Btd, PAIR);
```

And then press 1 & 2 at once on the Wiimote or the SYNC buttons if you are using a Wii U Pro Controller or a Wii Balance Board.

After that you can simply create the instance like so:

```
WII Wii(&Btd);
```

Then just press any button on the Wiimote and it will then connect to the dongle.

Take a look at the example for more information: Wii.ino.

Also take a look at the blog post:

- http://blog.tkjelectronics.dk/2012/08/wiimote-added-to-usb-host-library/

The Wii IR camera can also be used, but you will have to activate the code for it manually as it is quite large. Simply set `ENABLE_WII_IR_CAMERA` to 1 in settings.h.

The WiiIRCamera.ino example shows how it can be used.

All the information about the Wii controllers are from these sites:

- http://wiibrew.org/wiki/Wiimote
- http://wiibrew.org/wiki/Wiimote/Extension_Controllers
- http://wiibrew.org/wiki/Wiimote/Extension_Controllers/Nunchuck
- http://wiibrew.org/wiki/Wiimote/Extension_Controllers/Wii_Motion_Plus
- http://wiibrew.org/wiki/Wii_Balance_Board
- The old library created by *Tomoyuki Tanaka*: https://github.com/moyuchin/WiiRemote_on_Arduino also helped a lot.

PS Buzz Library

This library implements support for the Playstation Buzz controllers via USB.

It is essentially just a wrapper around the HIDUniversal which takes care of the initializing and reading of the controllers. The PSBuzz class simply inherits this and parses the data, so it is easy for users to read the buttons and turn the big red button on the controllers on and off.

The example PSBuzz.ino shows how one can do this with just a few lines of code.

More information about the controller can be found at the following sites:

- http://www.developerfusion.com/article/84338/making-usb-c-friendly/
- https://github.com/torvalds/linux/blob/master/drivers/hid/hid-sony.c

HID Libraries

HID devices are also supported by the library. However these require you to write your own driver. A few example are provided in the examples/HID directory. Including an example for the SteelSeries SRW-S1 Steering Wheel.

MIDI Library

The library support MIDI devices.
You can convert USB MIDI keyboard to legacy serial MIDI.

- USB_MIDI_converter.ino
- USB_MIDI_converter_multi.ino

For information see the following page: http://yuuichiakagawa.github.io/USBH_MIDI/.

Interface modifications

The shield is using SPI for communicating with the MAX3421E USB host controller. It uses the SCK, MISO and MOSI pins via the ICSP on your board.

Note this means that it uses pin 13, 12, 11 on an Arduino Uno, so these pins can not be used for anything else than SPI communication!

Furthermore it uses one pin as SS and one INT pin. These are by default located on pin 10 and 9 respectively. They can easily be reconfigured in case you need to use them for something else by cutting the jumper on the shield and then solder a wire from the pad to the new pin.

After that you need modify the following entry in UsbCore.h:

```
typedef MAX3421e<P10, P9> MAX3421E;
```

For instance if you have rerouted SS to pin 7 it should read:

```
typedef MAX3421e<P7, P9> MAX3421E;
```

See the "Interface modifications" section in the hardware manual for more information.

FAQ

> When I plug my device into the USB connector nothing happens?

- Try to connect a external power supply to the Arduino - this solves the problem in most cases.
- You can also use a powered hub between the device and the USB Host Shield. You should then include the USB hub library: `#include <usbhub.h>` and create the instance like so: `USBHub Hub1(&Usb);` .

> When I connecting my PS3 controller I get a output like this:

```
Dualshock 3 Controller Enabled

LeftHatX: 0 LeftHatY: 0 RightHatX: 0 RightHatY: 0
LeftHatX: 0 LeftHatY: 0 RightHatX: 0 RightHatY: 0
LeftHatX: 0 LeftHatY: 0 RightHatX: 0 RightHatY: 0
LeftHatX: 0 LeftHatY: 0 RightHatX: 0 RightHatY: 0
LeftHatX: 0 LeftHatY: 0 RightHatX: 0 RightHatY: 0
```

- This means that your dongle does not support 2.0+EDR, so you will need another dongle. Please see the following list for tested working dongles.

> When compiling I am getting the following error: "fatal error: SPI.h: No such file or directory".

- Please make sure to include the SPI library like so: `#include <SPI.h>` in your .ino file.