



# ENABLED CONTROLLER WIRELESS INSTRUCTIONS MANUAL

An accessible adaptive switch interface box

**Version 1.2**

<https://github.com/milador/Enabled-Controller-Wireless>

## Contents

Components List .....	2
Software.....	3
Installing Arduino IDE.....	3
Installing Arduino Libraries .....	3
Board Support Packages .....	3
Required Software and libraries .....	3
Uploading Software .....	5
Hardware Assembly .....	6
Printed circuit Board design.....	6
Enclosure design .....	6
Enabled-Controller Wireless Assembly.....	7
Joystick Assembly.....	43

## Components List

- 1.** ADAFRUIT ITSY-BITSY NRF52840 EXPRESS **x 1**
- 2.** ADDRESS LED DISC SER RGB 5MM **x 1**
- 3.** 14 POSITION HEADER **x 2**
- 4.** CONN JACK STEREO 3.5MM R/A **x 8**
- 5.** CONN JACK 4COND 3.5MM **x 2**
- 6.** HEX NUT 0.217" STEEL M3 **x 2**
- 7.** MACH SCREW PAN HEAD PHILLIPS M3 **x 2**
- 8.** QWIIC JST CONNECTOR - SMD 4-PIN **x 1**
- 9.** PRO TRINKET LI-BATTERY BACKPACK **x 1**
- 10.** CONN HDR 3POS 0.1 GOLD PCB **x 1**
- 11.** BATTERY LITHIUM 3.7V 500MAH **x 1**
- 12.** **(OPTIONAL)** SWITCH TOGGLE SPDT **x1**

The bill of materials can also be downloaded from GitHub repository under main directory.

[https://github.com/milador/Enabled-Controller-Wireless/blob/master/Components/Enabled\\_Controller\\_Wireless\\_BOM.csv](https://github.com/milador/Enabled-Controller-Wireless/blob/master/Components/Enabled_Controller_Wireless_BOM.csv)

## Software

The Enabled-Controller Wireless software can be downloaded from the GitHub repository under software directory.

[https://github.com/milador/Enabled-Controller-Wireless/tree/master/Software/Arduino/Enabled\\_Controller\\_Wireless\\_Software/Enabled\\_Controller\\_Wireless\\_Software.ino](https://github.com/milador/Enabled-Controller-Wireless/tree/master/Software/Arduino/Enabled_Controller_Wireless_Software/Enabled_Controller_Wireless_Software.ino)

### Installing Arduino IDE

The Arduino IDE is required to compile and upload the source code to the MCU in your Adafruit ItsyBitsy board. You can download and install the Arduino from official Arduino website that you can find using following link:

<https://www.arduino.cc/en/software>

### Installing Arduino Libraries

The following instructions on how to install additional Arduino libraries helps you to get started with setting up Arduino IDE with required libraries and dependencies.

<https://www.arduino.cc/en/guide/libraries>

### Board Support Packages

You can find the official instructions to install Board Support Packages on Adafruit website using following link:

<https://learn.adafruit.com/adafruit-itsybitsy-nrf52840-express/arduino-support-setup>

Alternatively, you can perform following instructions to install Board Support Packages:

1. Open and start the Arduino IDE
2. Go to **File > Preferences**
3. Add following link as a new line under **Additional Board Manager URLs**
  - [https://www.adafruit.com/package\\_adafruit\\_index.json](https://www.adafruit.com/package_adafruit_index.json)
4. Restart the Arduino IDE
5. Open the Boards Manager option from the **Tools > Board** menu and install **Adafruit nRF52 Boards by Adafruit**
6. Wait until the IDE finishes installing the cross-compiling toolchain and tools associated with Board Support Package. This may take few minutes.
7. That's it! The installation of Board Support Packages is finished.

### Required Software and libraries

The wireless version of the software requires the following files and libraries:

- [StopWatch library](#)
- [EasyMorseBlue library](#)
- Adafruit\_LittleFS library
- InternalFileSystem library
- [ArduinoJson library](#)

- [Adafruit\\_NeoPixel library](#)
- [Enabled\\_Controller\\_Wireless\\_Software.ino](#)

**StopWatch** library helps to calculate the reaction time and timeout in the morse code interface.

The **EasyMorseBlue** library is the Bluetooth version of **EasyMorse** library which is used to enable the end user to use morse code or a sequence of dots and dashes as input method.

**Adafruit\_LittleFS** and **InternalFileSystem** libraries are part of internal libraries of Arduino core for **Adafruit nRF52** and you do not need to install them. Those two libraries are necessary to save and modify the settings.

**ArduinoJson** is used to save the configuration settings in JSON text format.

Enabled-Controller is using Adafruit\_NeoPixel library to provide visual feedback using RGB LED.

1. Visit the [StopWatch library](#) github repository page.
2. Click on **Code > Download Zip** to download StopWatch library.
3. Extract **Stopwatch\_RT-master.zip** file
4. Rename **Stopwatch\_RT-master** folder to **Stopwatch** folder under **Stopwatch\_RT-master** subdirectory.
5. Copy or move **Stopwatch** folder to Arduino installation library subdirectory. As an example: This is found under **C:\Program Files (x86)\Arduino\libraries** in windows 10.
6. Visit the [EasyMorseBlue library](#) github repository page.
7. Click on **Code > Download Zip** to download **EasyMorseBlue** library.
8. Extract **EasyMorse.zip** file
9. Rename **EasyMorseBlue-master** folder to **EasyMorseBlue** folder under **EasyMorseBlue-master** subdirectory.
10. Copy or move **EasyMorseBlue** folder to Arduino installation library subdirectory. As an example: This is found under **C:\Program Files (x86)\Arduino\libraries** in windows 10.
11. Visit the [ArduinoJson library](#) github repository page.
12. Click on **Code > Download Zip** to download **ArduinoJson** library.
13. Extract **ArduinoJson.zip** file
14. Rename **ArduinoJson-master** folder to **ArduinoJson** folder under **ArduinoJson-master** subdirectory.
15. Copy or move **EasyMorseBlue** folder to Arduino installation library subdirectory. As an example: This is found under **C:\Program Files (x86)\Arduino\libraries** in windows 10.
16. Visit the [Enabled\\_Controller\\_Wireless\\_Software.ino](#) raw source code file under **Enabled-Controller** github repository page.
17. Right click on the source code or any place on this page and select **Save Page As...**
18. Select the directory you would like to save the software in your computer.
19. Change **File name** from **Enabled\_Controller\_Wireless\_Software** to **Enabled\_Controller\_Wireless\_Software.ino**
20. Change **Save as type** to **All Files**.
21. Click on **Save** button.
22. Open the directory you selected in **step 13**.
23. Double left click or open **Enabled\_Controller\_Wireless\_Software** file

24. Arduino IDE will ask your permission to create a new sketch folder named ***Enabled\_Controller\_Wireless\_Software*** and move ***Enabled\_Controller\_Wireless\_Software.ino*** under this folder.
25. Click on the ***Ok*** button.
26. Arduino IDE should now open the ***Enabled\_Controller\_Wireless\_Software.ino*** file automatically.

### Uploading Software

***Note: Make sure all files are included in your local copy of Software directory before uploading it to the Adafruit ItsyBitsy board. The libraries can be installed in Arduino libraries.***

You can go ahead and upload the downloaded **.ino** code to Adafruit ItsyBitsy board using Arduino IDE once all the necessary libraries are installed.

1. Start the Arduino IDE
2. Open ***Enabled\_Controller\_Wireless\_Software.ino***
3. Select the Board under ***Tools > Board > Adafruit nRF52 Boards*** as ***Adafruit ItsyBitsy nRF52840 Express***.
4. Select the correct port number under ***Tools > Port*** which should show ***COM XX (Adafruit ItsyBitsy nRF52840 Express)***

***Note: It's very important to make sure the correct Board and port number are selected as selecting the wrong board may result problems with bootloader of Adafruit ItsyBitsy board.***

5. Press the Verify button to make sure there is no problem with the software and libraries
6. Press Upload button

You can now go ahead and upload the software. Arduino IDE will show you a ***Done Uploading*** message indicating the software is uploaded to your Enabled-Controller.

The LED on the Enabled-Controller main board blinks two times in ***teal*** to indicate the start of initialization process has been started and it will blink again two times in green to indicate end of the initialization process.

You can also open the Serial Monitor in Arduino IDE on 115200 baud-rate to read initialization information about the version of software.

## Hardware Assembly

### Printed circuit Board design

The printed circuit boards (PCB) can be downloaded from GitHub repository under Hardware directory.

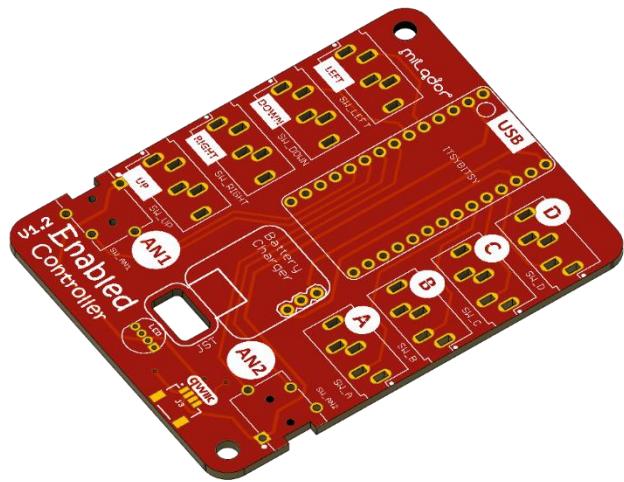
<https://github.com/milador/Enabled-Controller-Wireless/tree/master/Hardware/PCB>

### Enclosure design

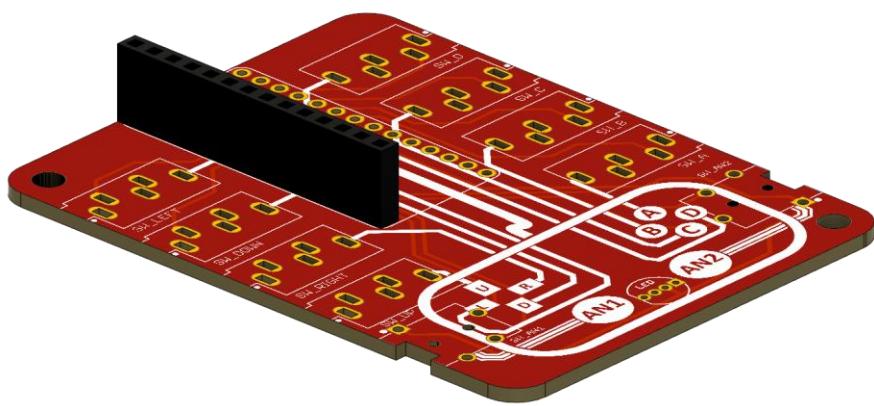
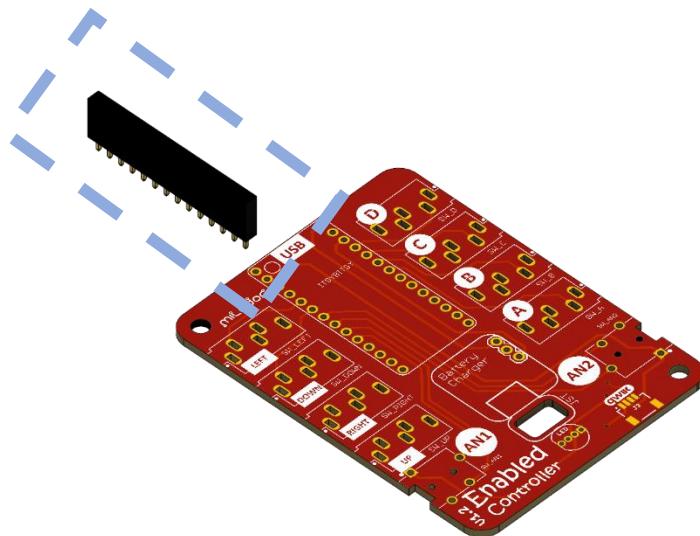
The enclosure/housing files in STL format can be downloaded from GitHub repository under Hardware directory.

<https://github.com/milador/Enabled-Controller-Wireless/tree/master/Hardware/Enclosure>

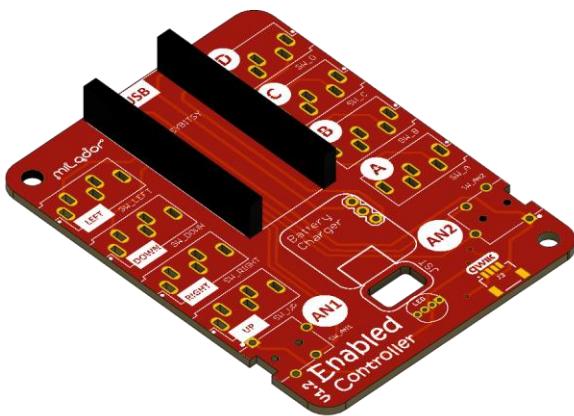
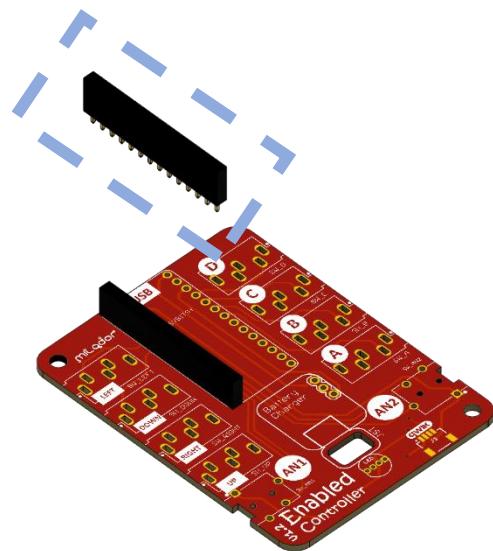
## Enabled-Controller Wireless Assembly



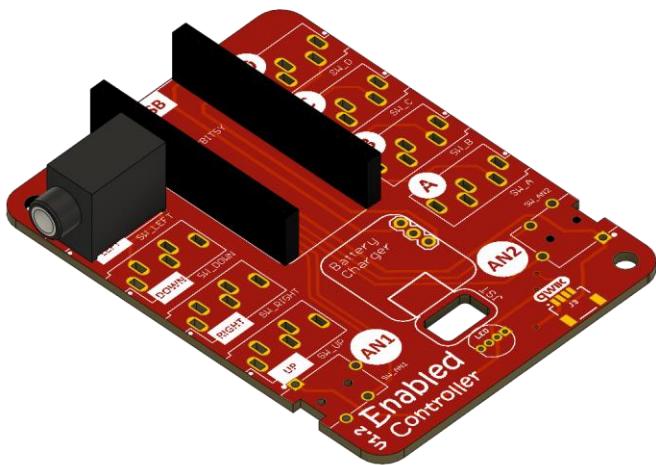
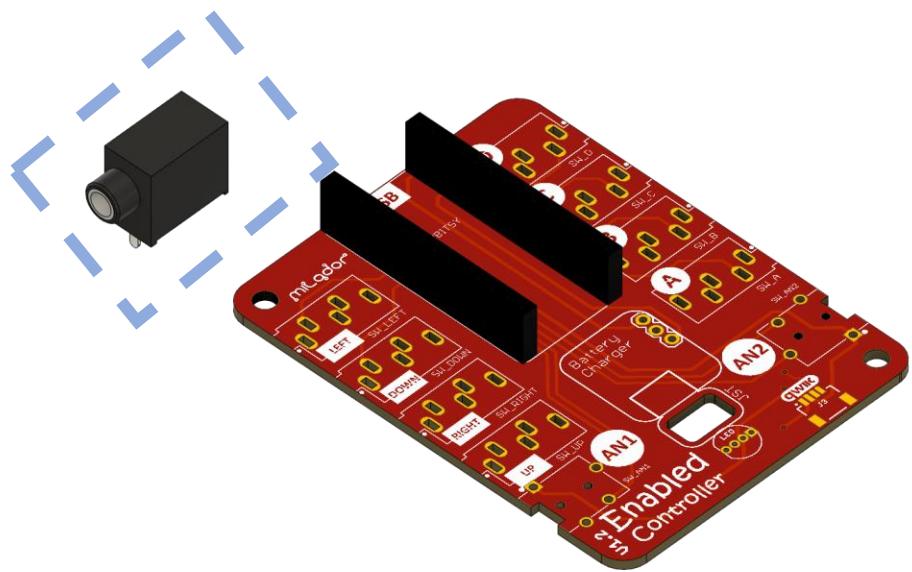
**Step 1: Enabled-Controller board x 1.**



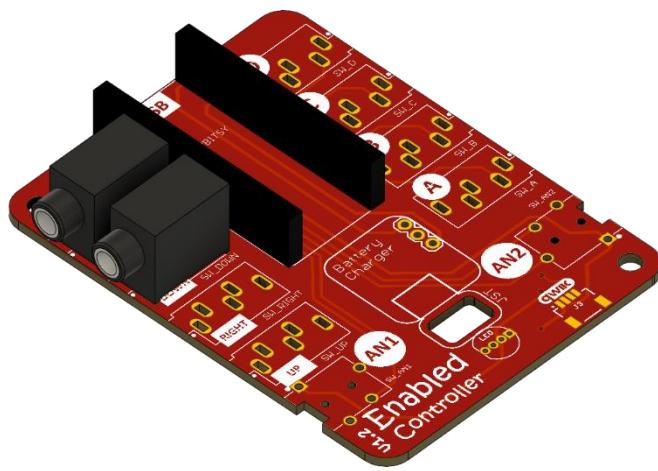
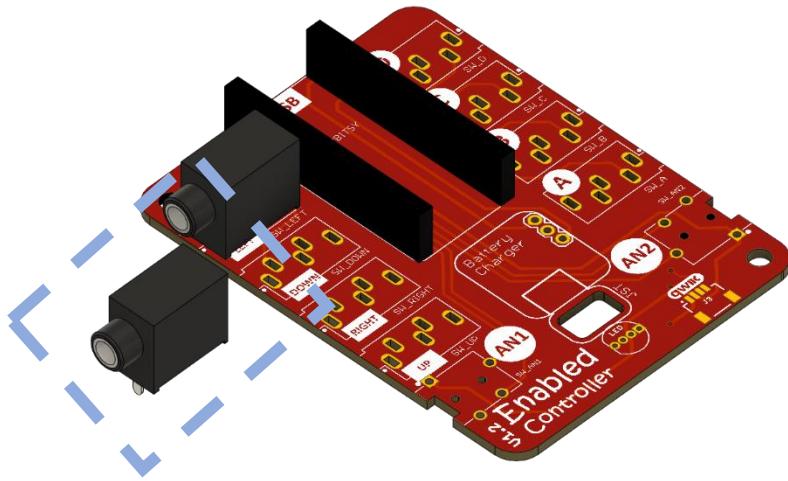
**Step 2: Solder the first 14 position female header over the Enabled-Controller board x 1.**



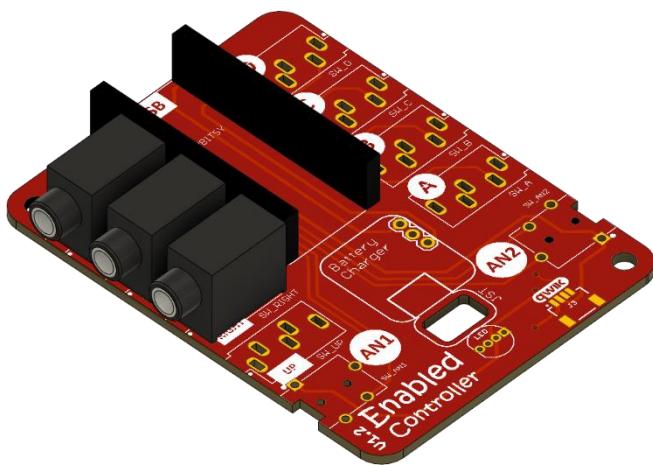
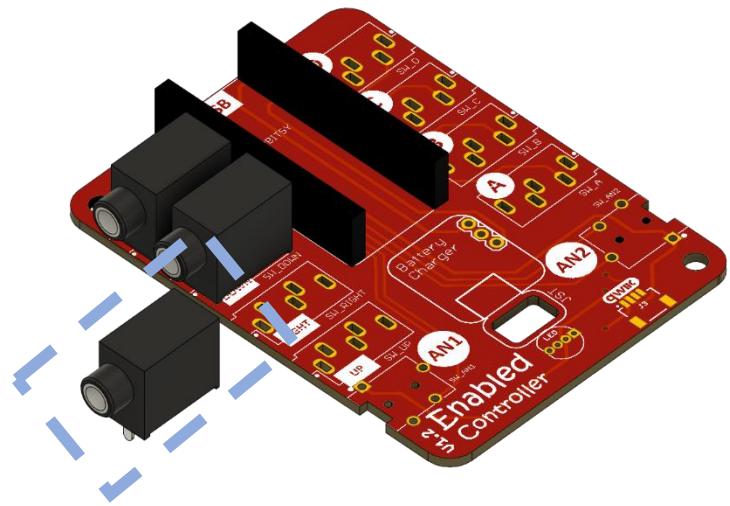
**Step 3: Solder the second 14 position female header over the Enabled-Controller board x 1.**



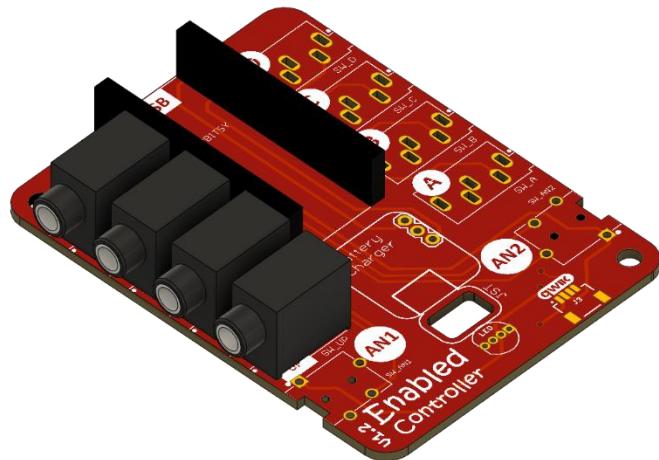
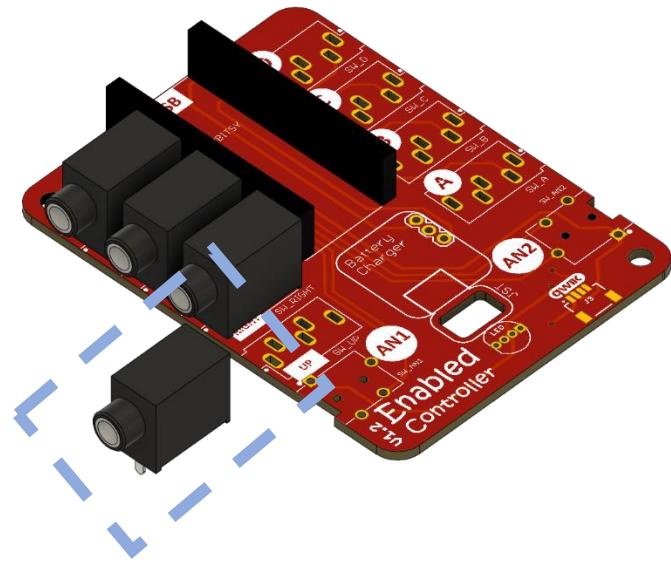
**Step 4: Position a 3.5mm jack on the Left switch outline and solder the pins x 1.**



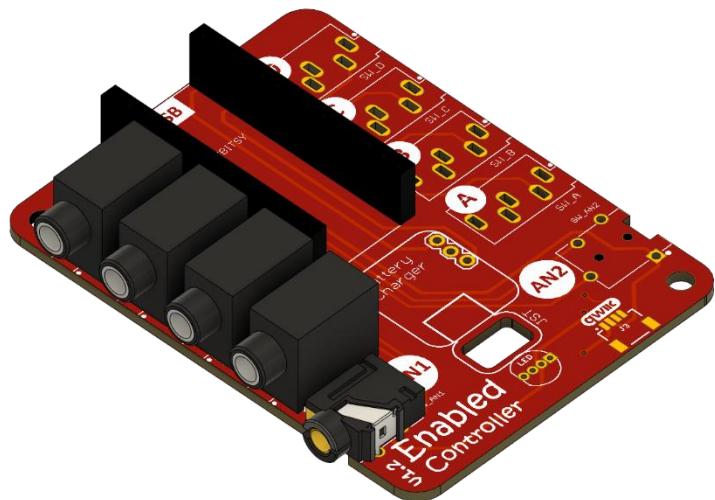
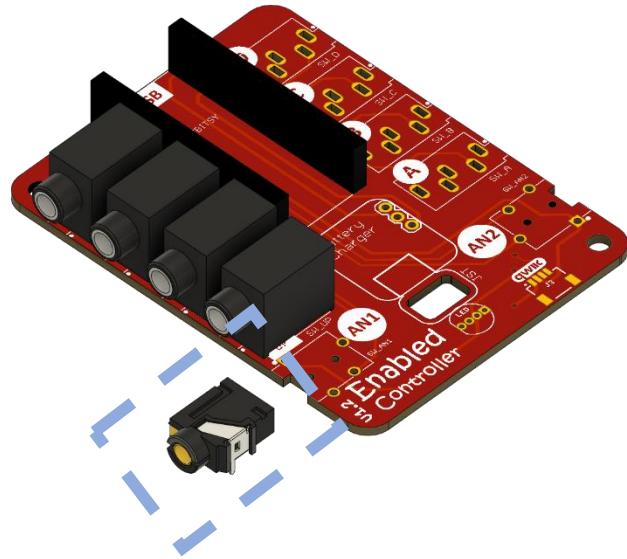
**Step 5: Position a 3.5mm jack on the Down switch outline and solder the pins x 1.**



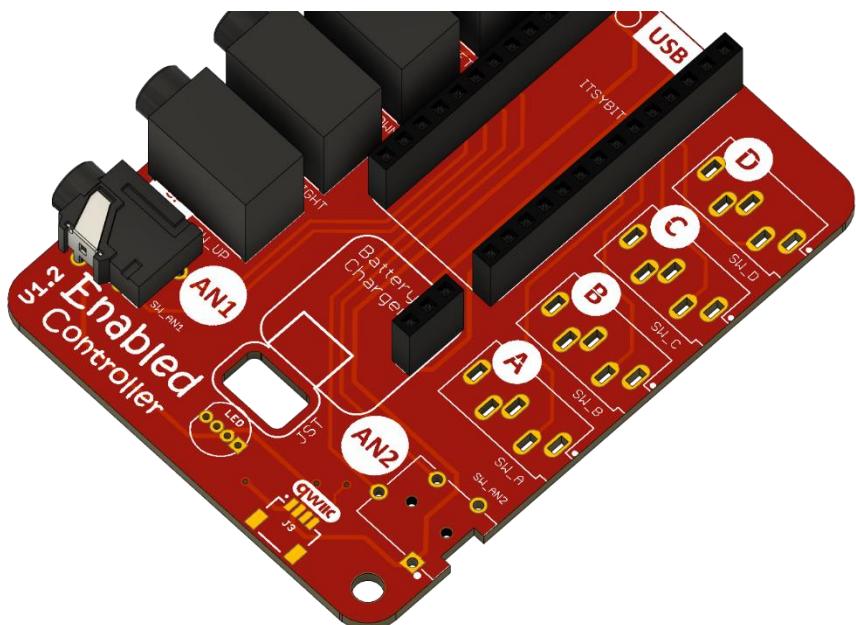
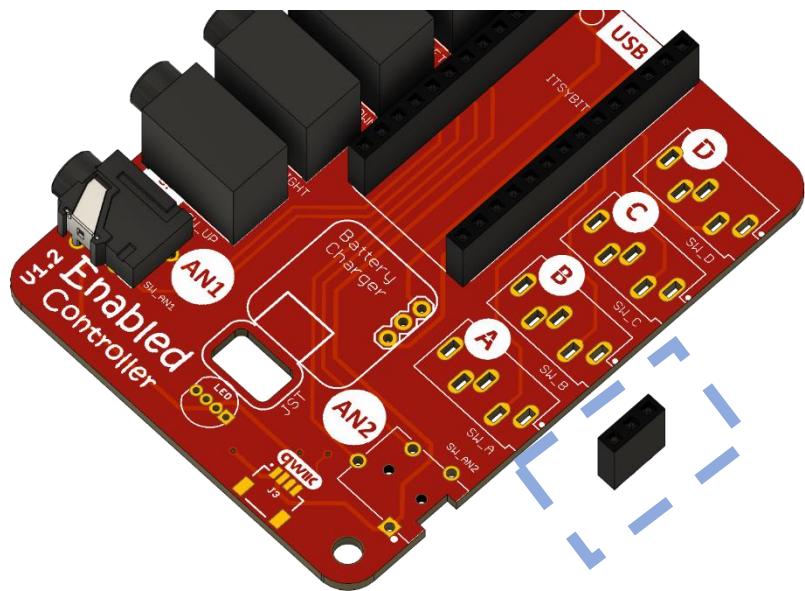
**Step 6: Position a 3.5mm jack on the Right switch outline and solder the pins x 1.**



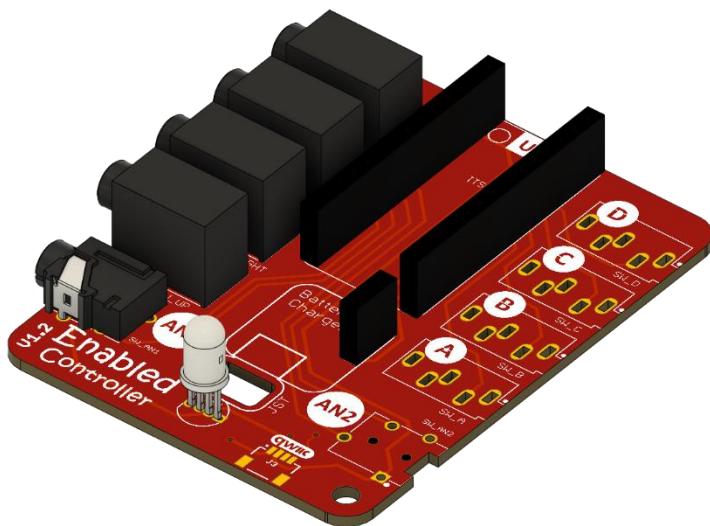
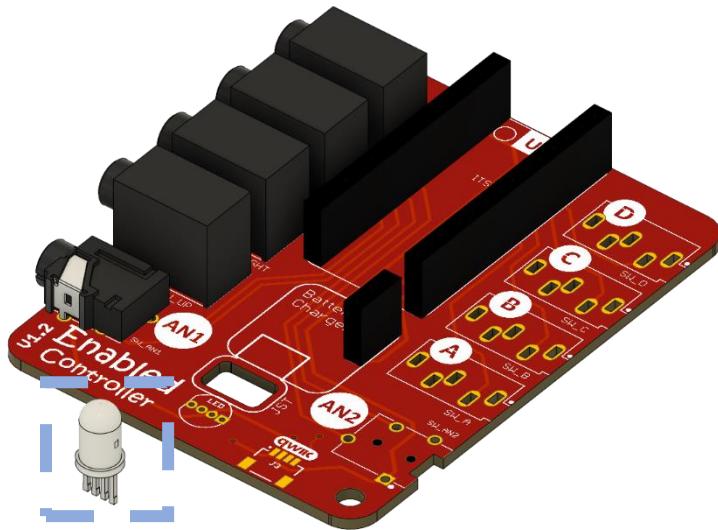
**Step 7: Position a 3.5mm jack on the Up switch outline and solder the pins x 1.**



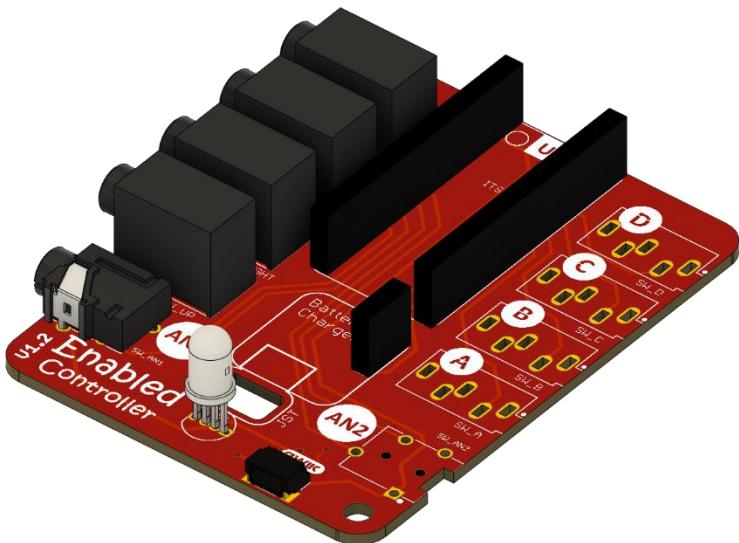
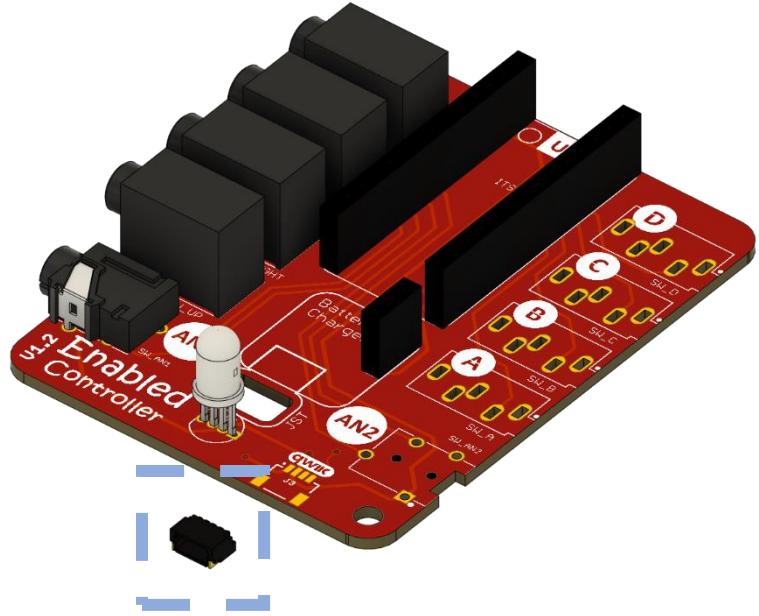
**Step 8: Position a 4 Conductor 3.5mm jack on the AN1 switch outline and solder the pins x 1.**



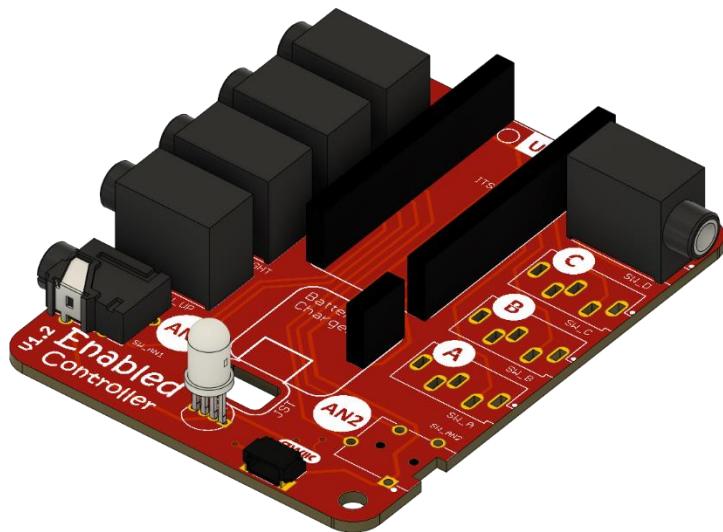
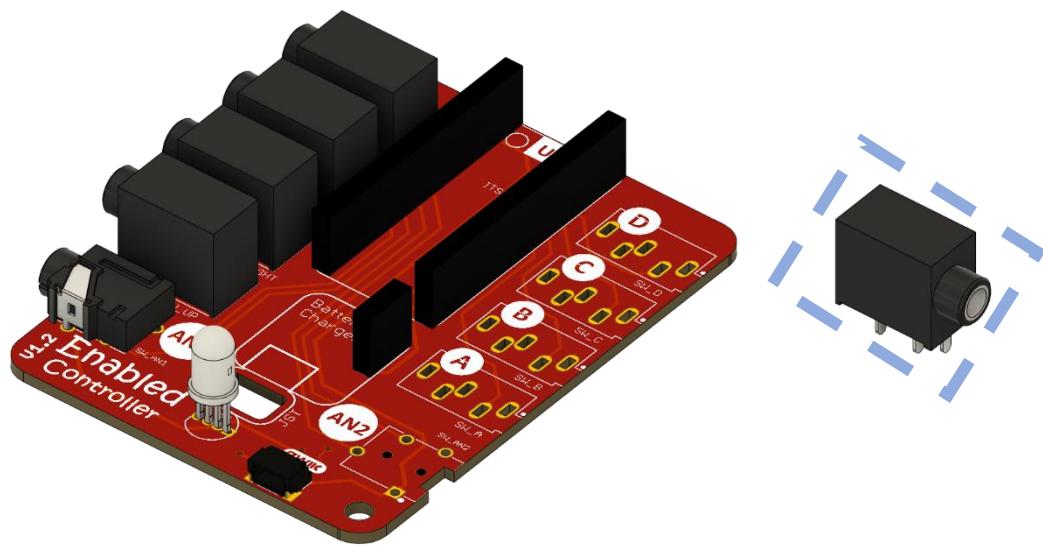
**Step 9: Solder a 3 position female header outlined for the battery charger**



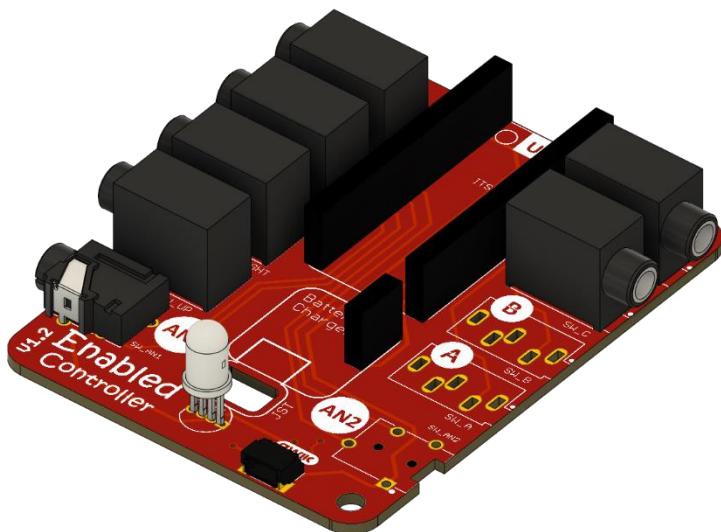
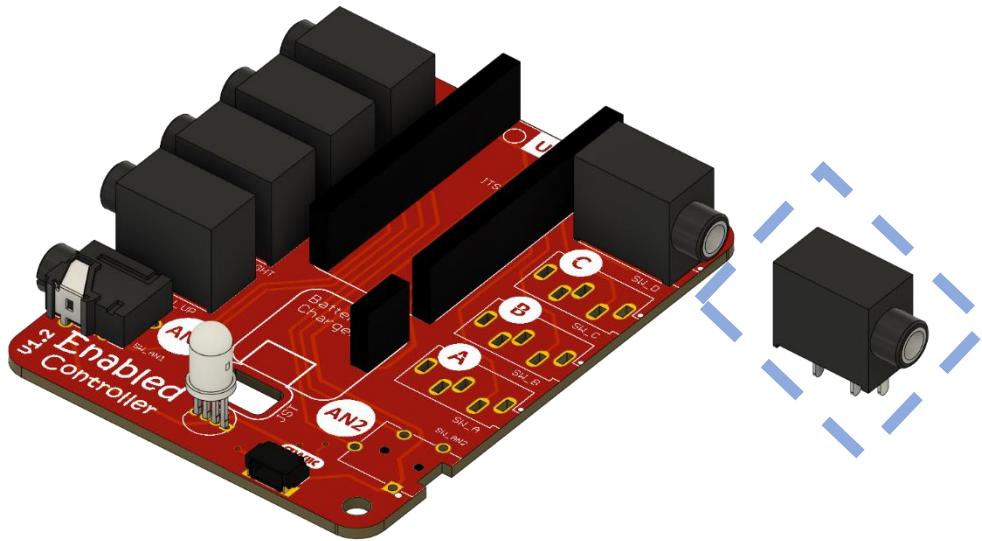
**Step 10: Position and solder the RGB Led pins x 1.**



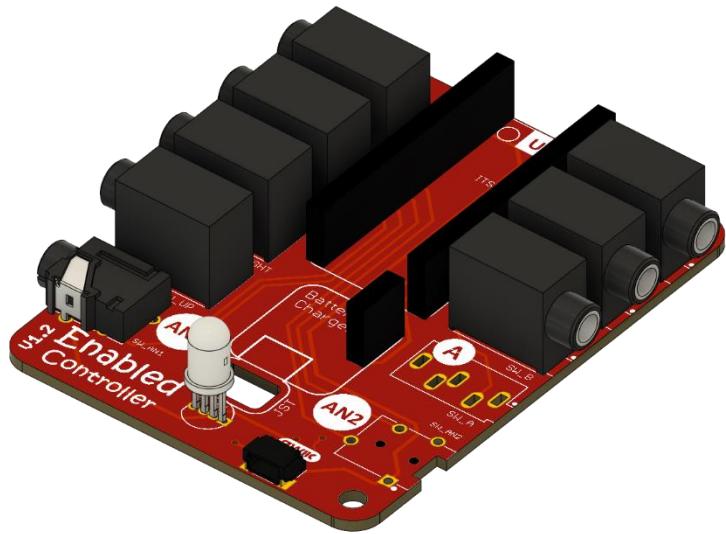
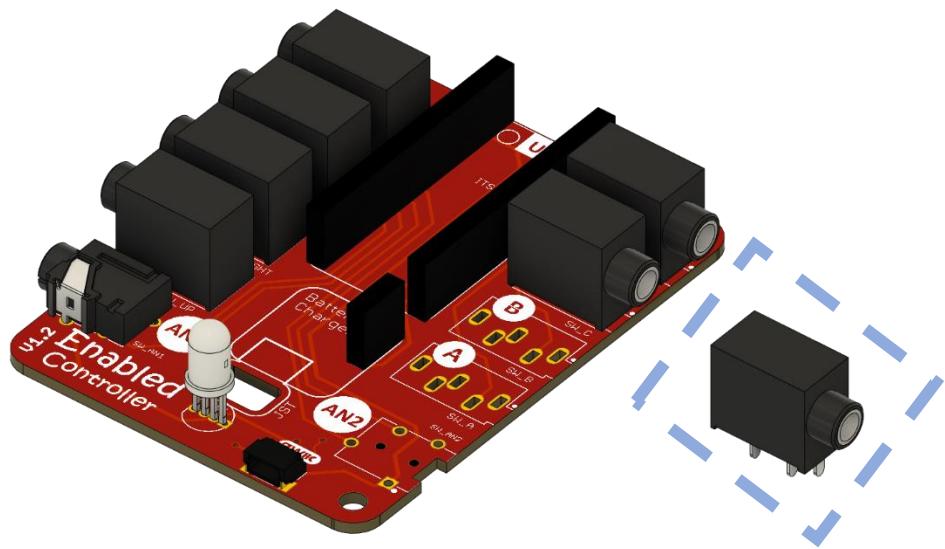
**Step 11: Position the Qwiic JST Connector using the outline and solder the pins**



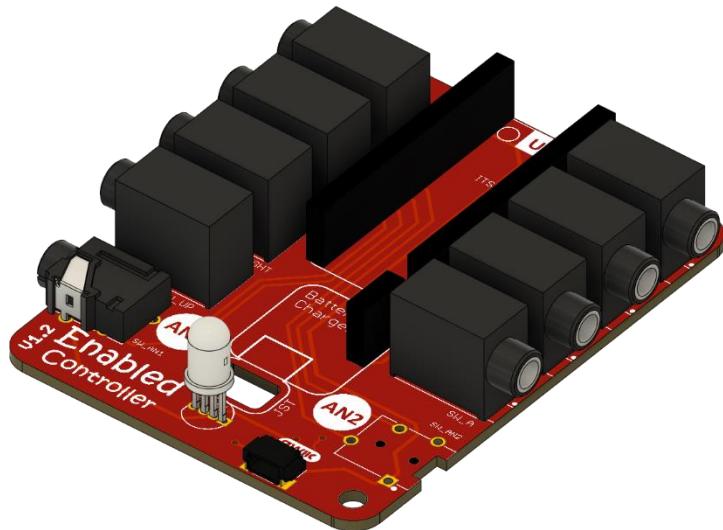
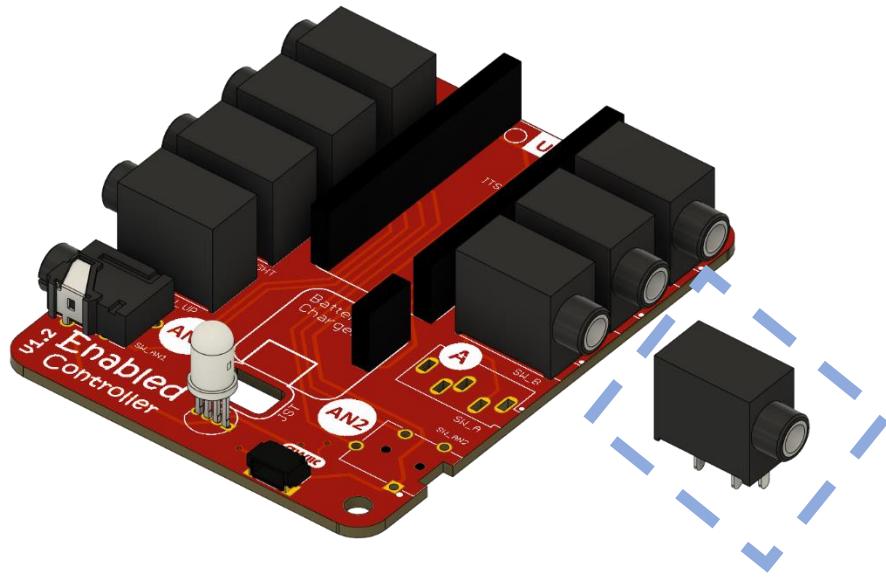
**Step 12: Position a 3.5mm jack on the D switch outline and solder the pins x 1.**



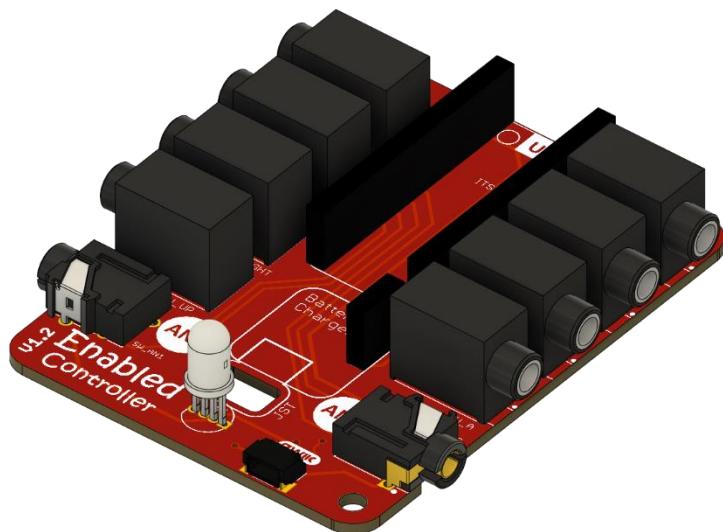
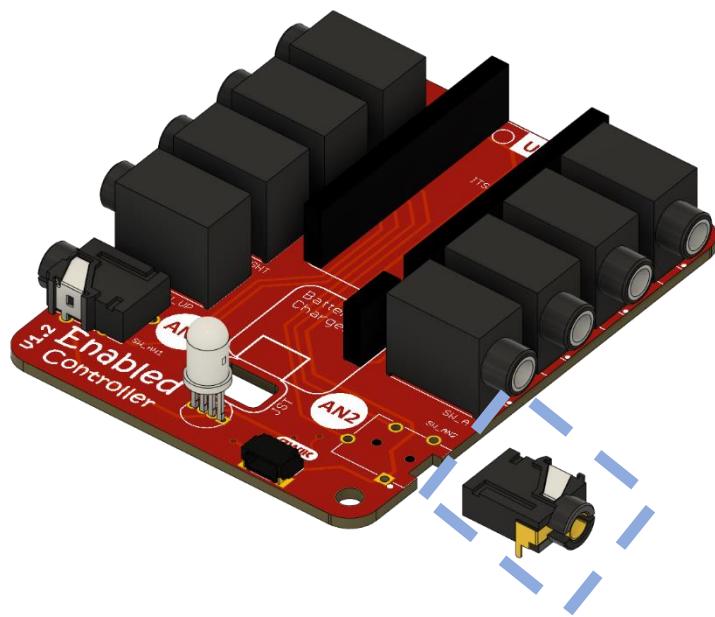
**Step 13: Position a 3.5mm jack on the C switch outline and solder the pins x 1.**



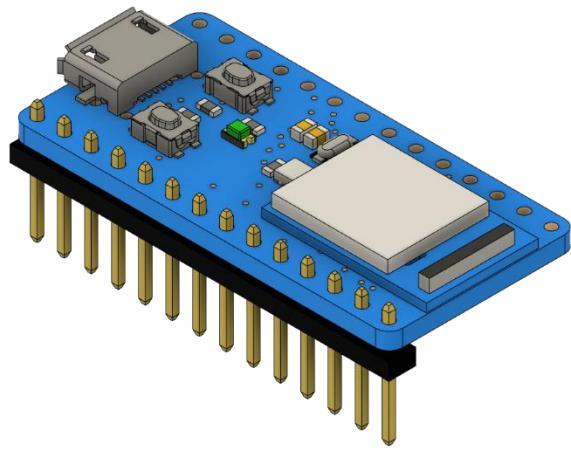
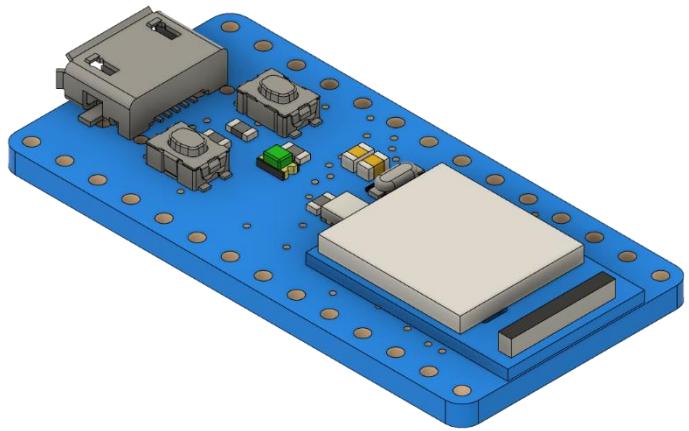
**Step 14: Position a 3.5mm jack on the B switch outline and solder the pins x 1.**



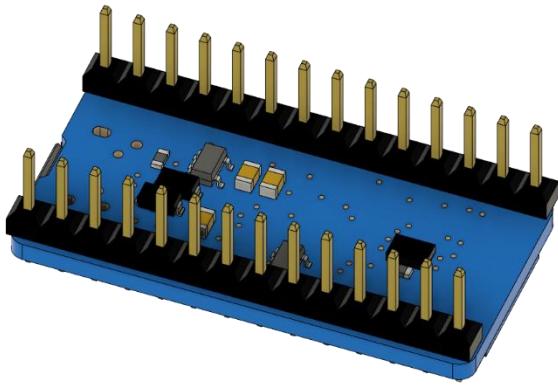
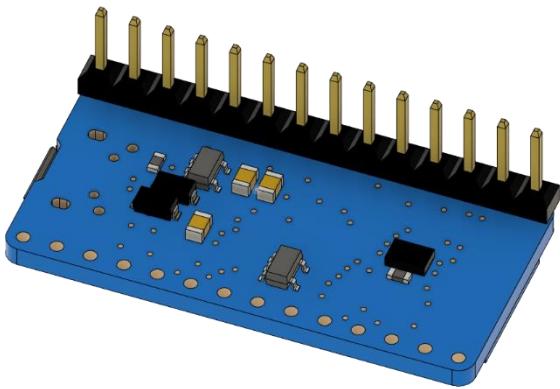
**Step 15: Position a 3.5mm jack on the A switch outline and solder the pins x 1.**



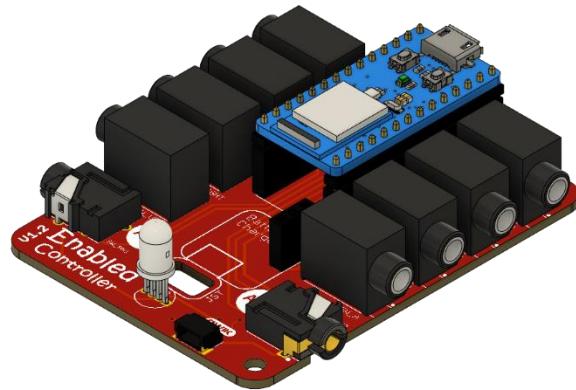
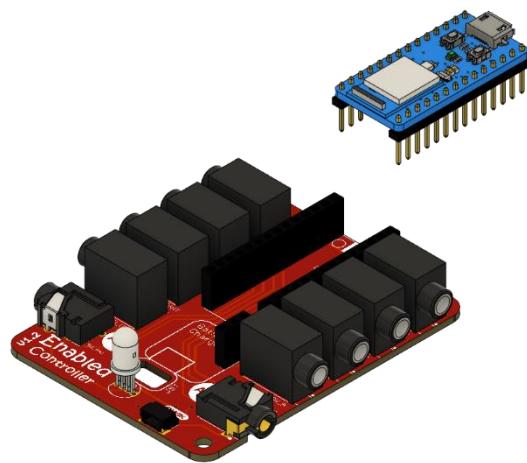
**Step 16: Position a 4 Conductor 3.5mm jack on the AN2 switch outline and solder the pins x 1.**



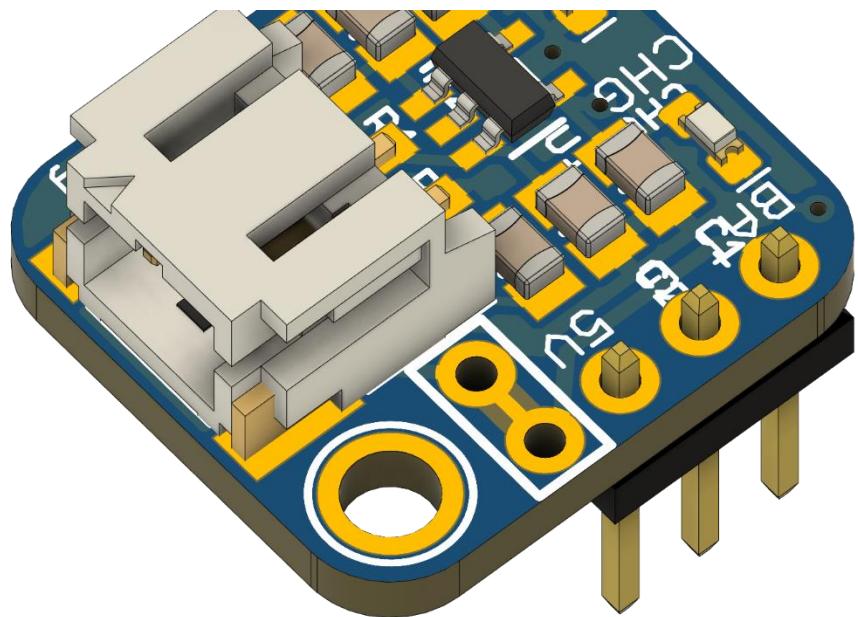
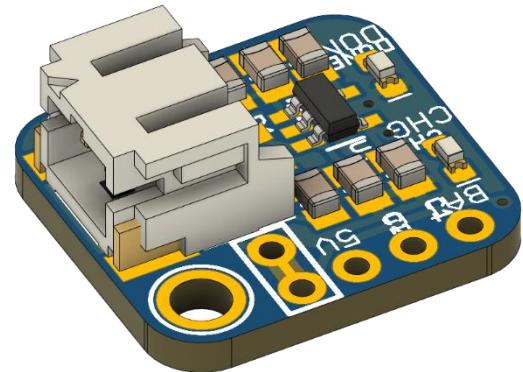
**Step 17: Position and solder the first 14 position male header on ItsyBitsy board x 1.**



**Step 18: Position and solder the second 14 position male header on ItsyBitsy board x 1.**



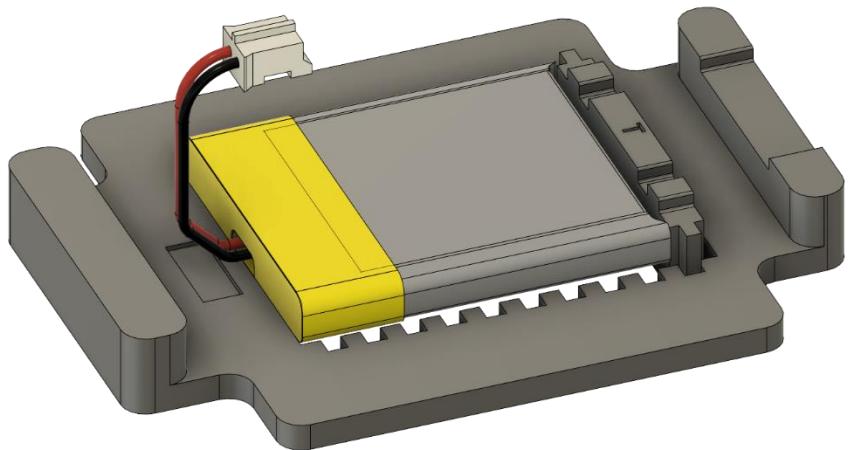
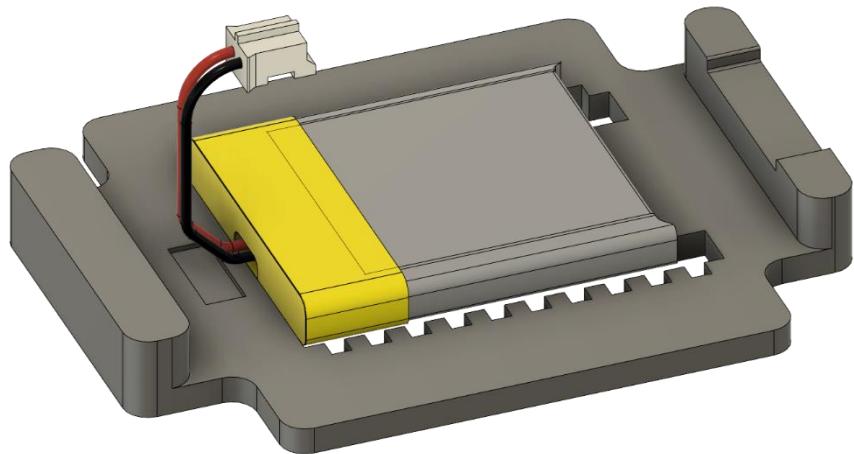
**Step 19: Position and stack the soldered ItsyBitsy board over Enabled-Controller board.**



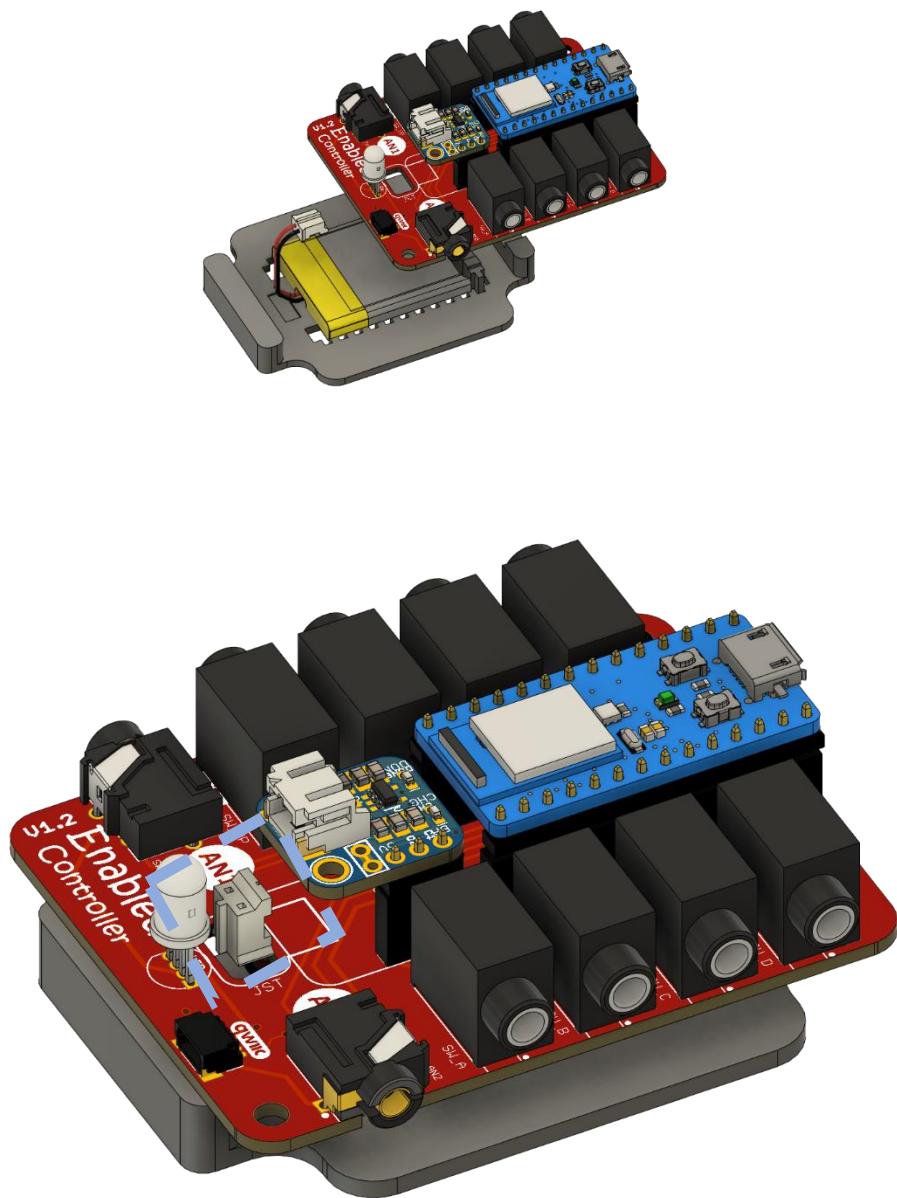
**Step 20: Position and solder a 3 position male header under the LI-BATTERY Backpack x 1.**



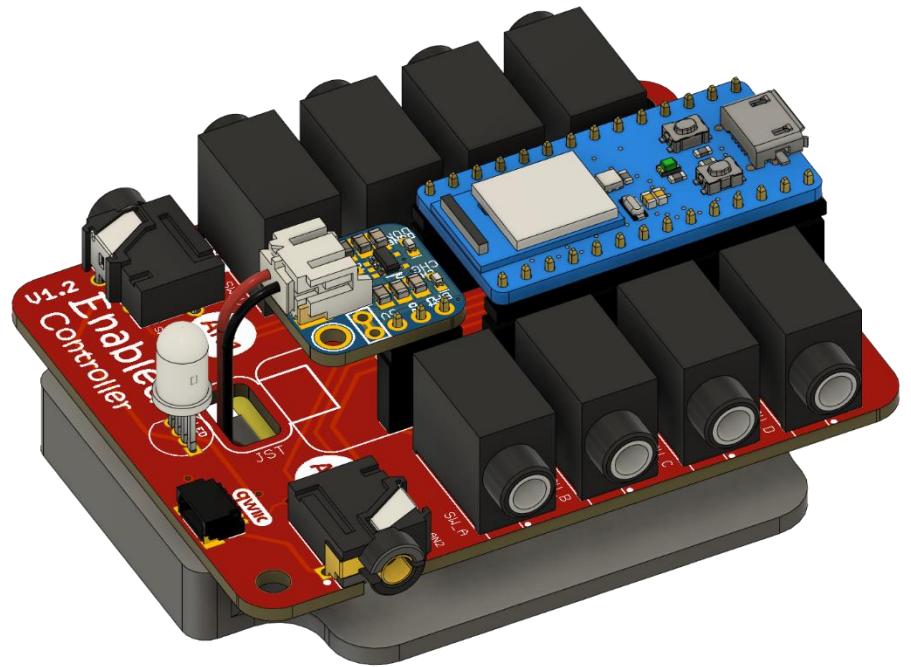
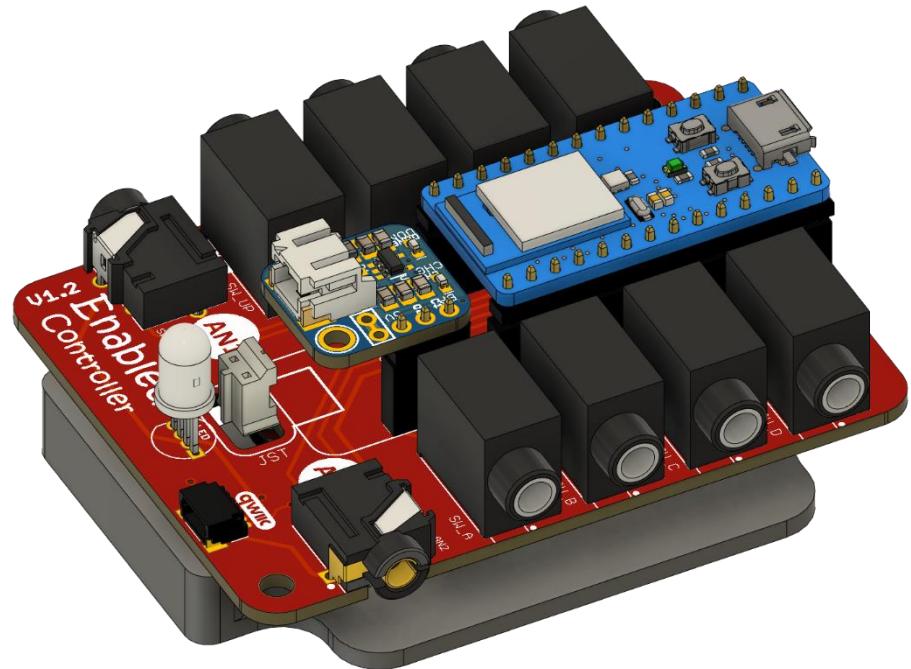
**Step 21: Position and stack the soldered LI-BATTERY Backpack over Enabled-Controller board.**



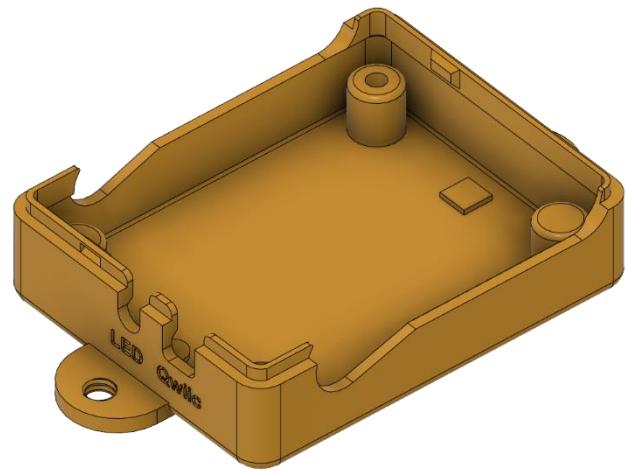
**Step 22: Place and secure the battery in battery holder ( Use Tape if needed ).**



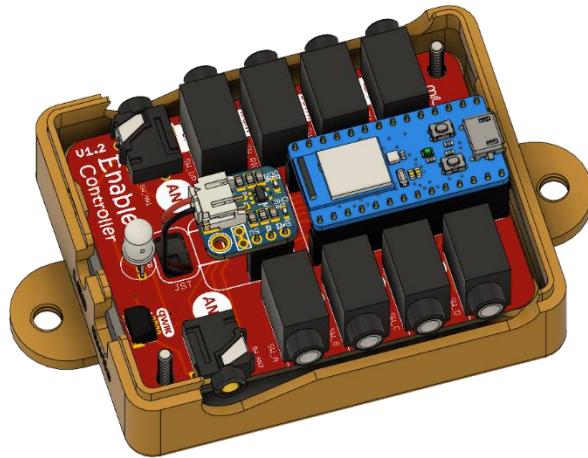
**Step 23: Pass the Battery JST cable through the PCB cutout to the other side.**



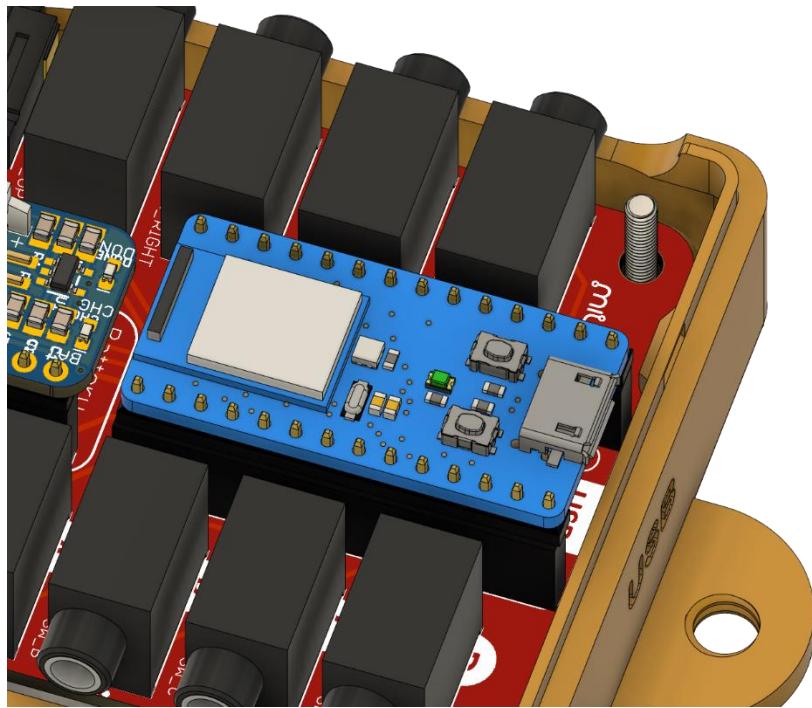
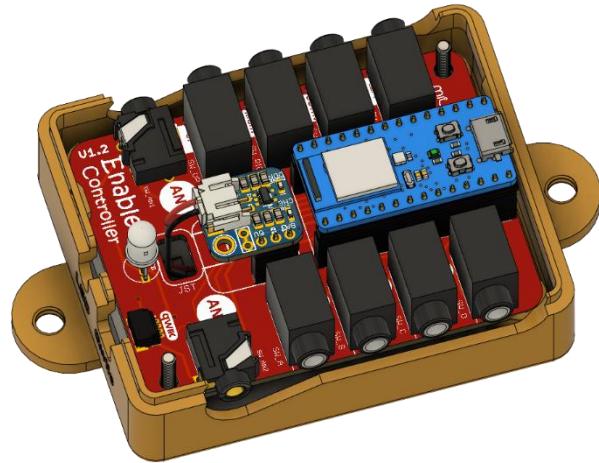
**Step 24: Connect the Battery JST cable to the LI-BATTERY Backpack JST connector.**



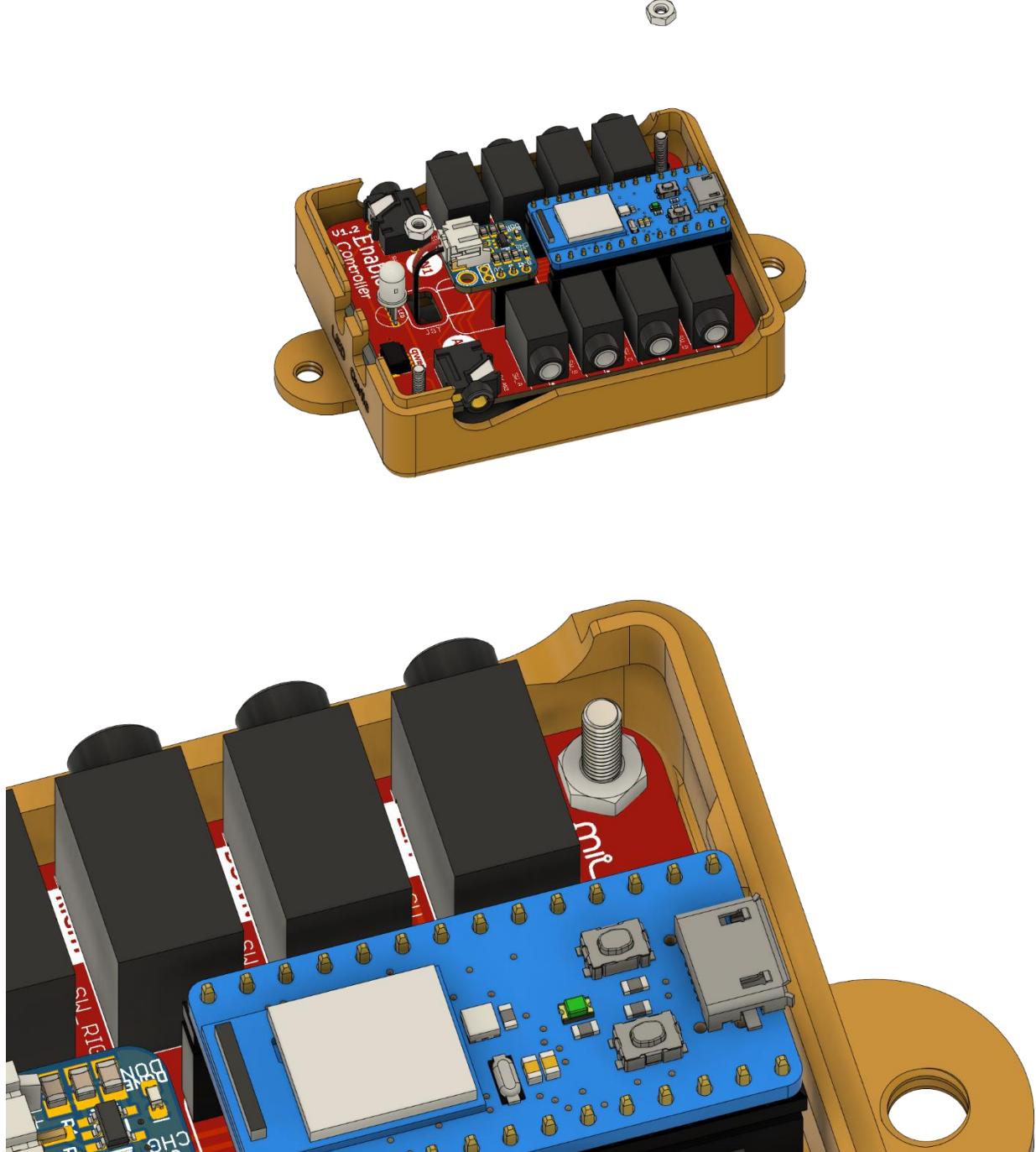
**Step 25: Position the assembled Enabled-Controller board in bottom enclosure.**



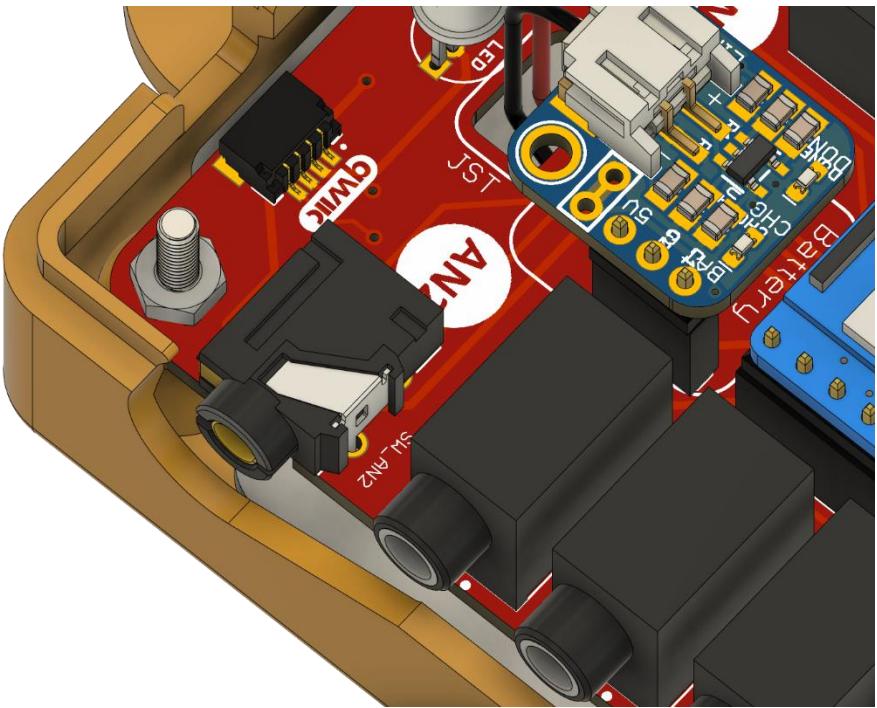
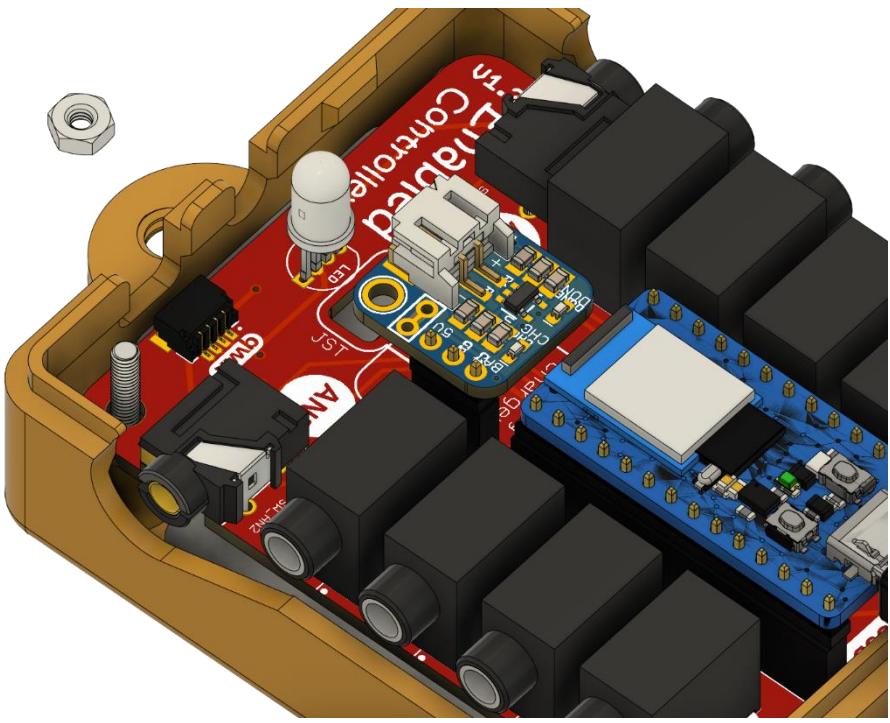
**Step 26: Position the M3 screws under the bottom enclosure x 2.**



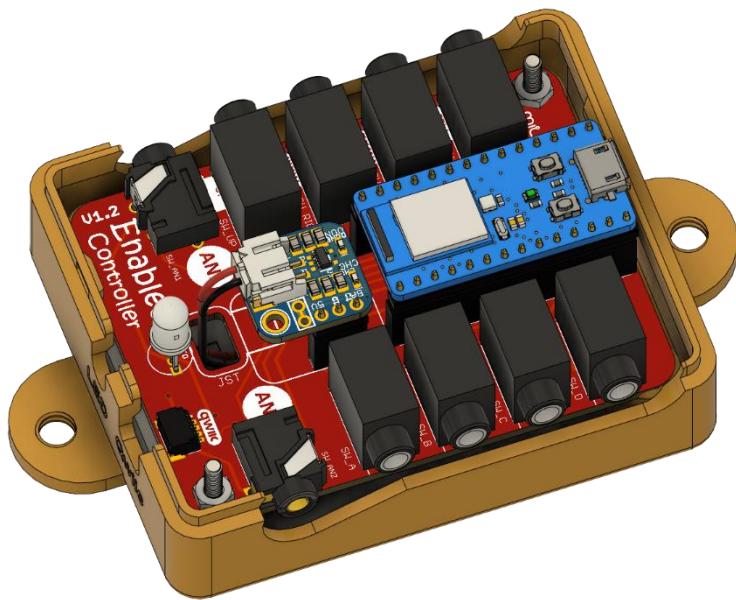
**Step 27: Insert the M3 screws and push them through the Enabled-Controller drill holes x 2.**



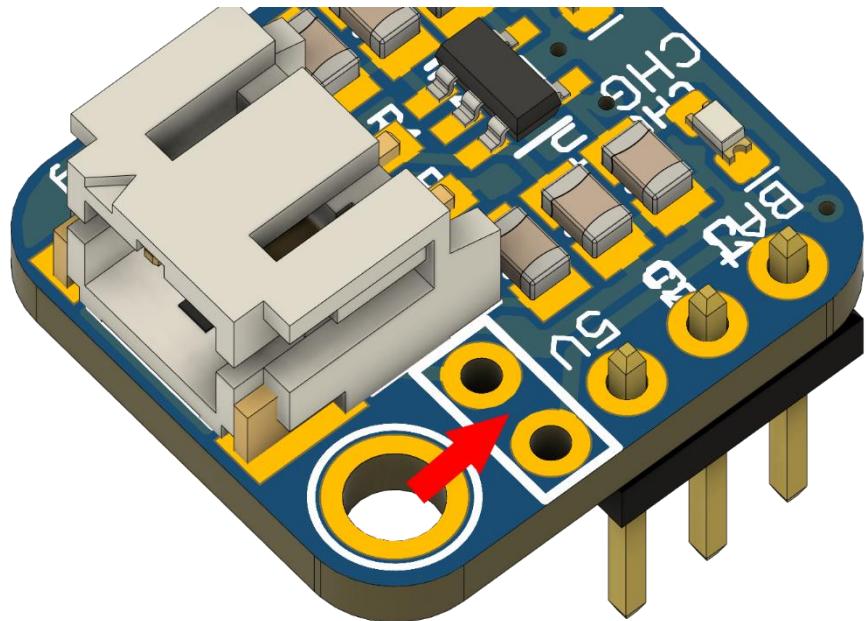
**Step 28: Insert the first M3 screw nut x 1.**



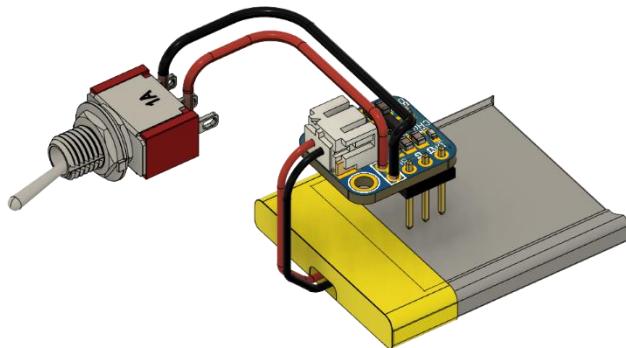
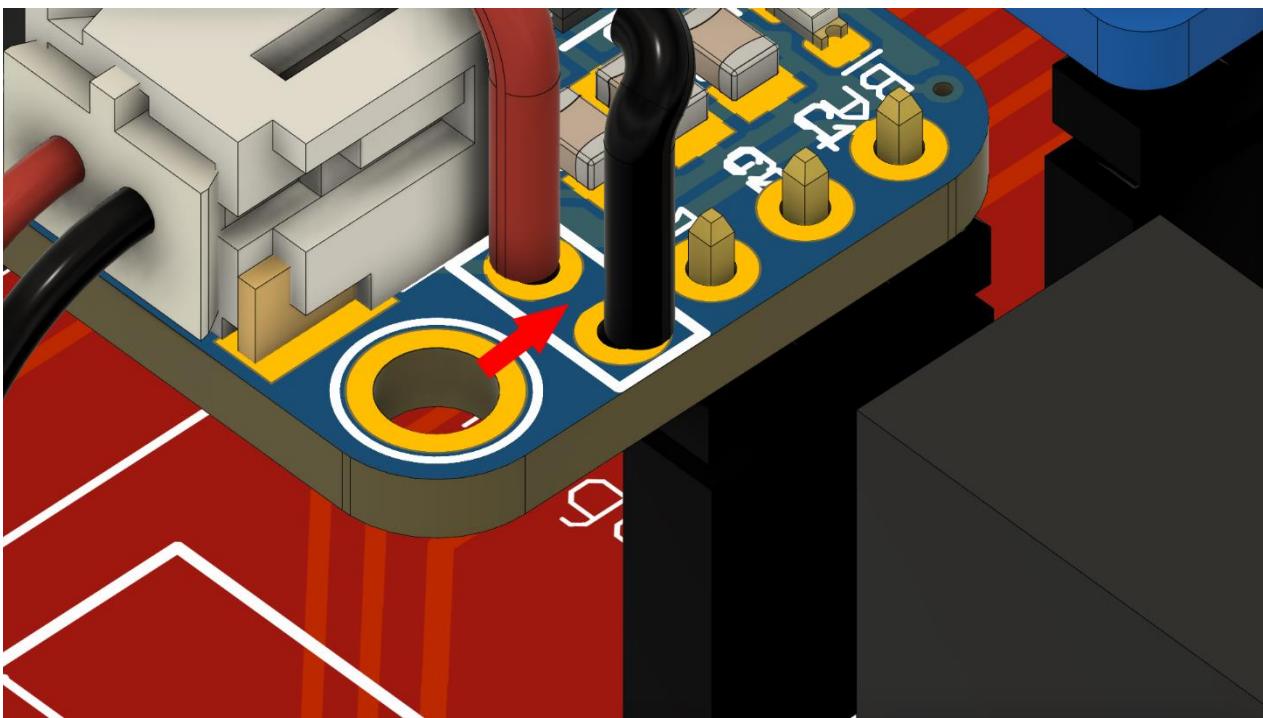
**Step 29: Insert the second M3 screw nut x 1.**



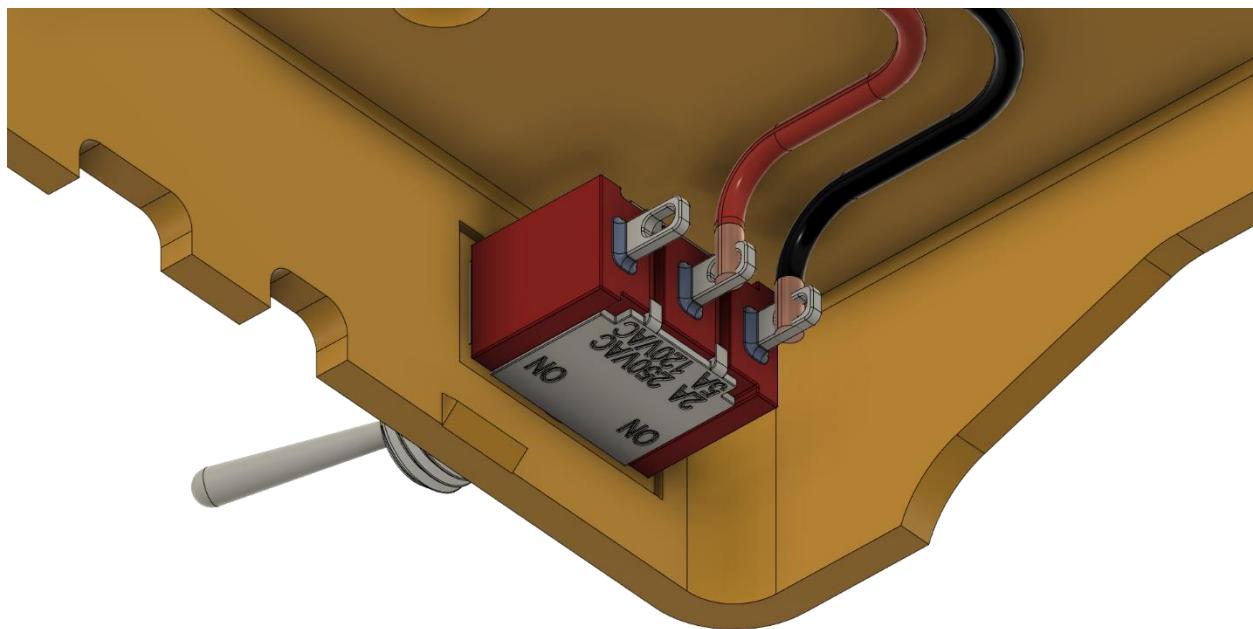
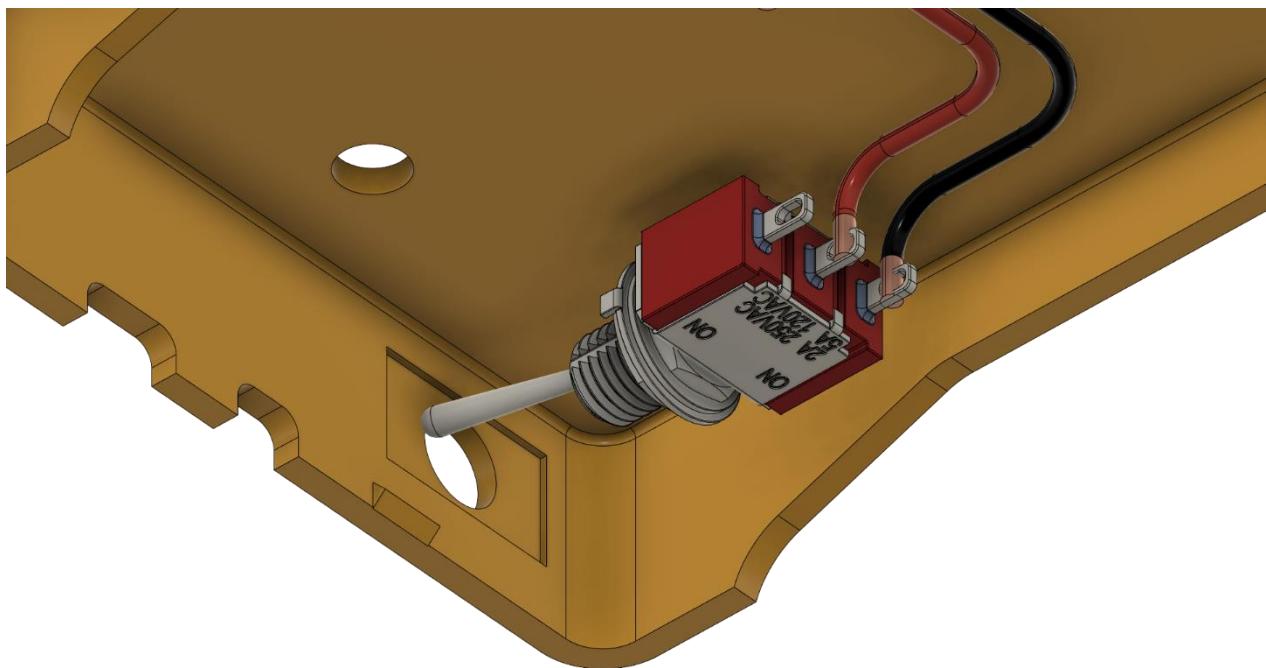
**Step 30:** Make sure the M3 screws hold the Enabled-Controller board by tightening the nuts.



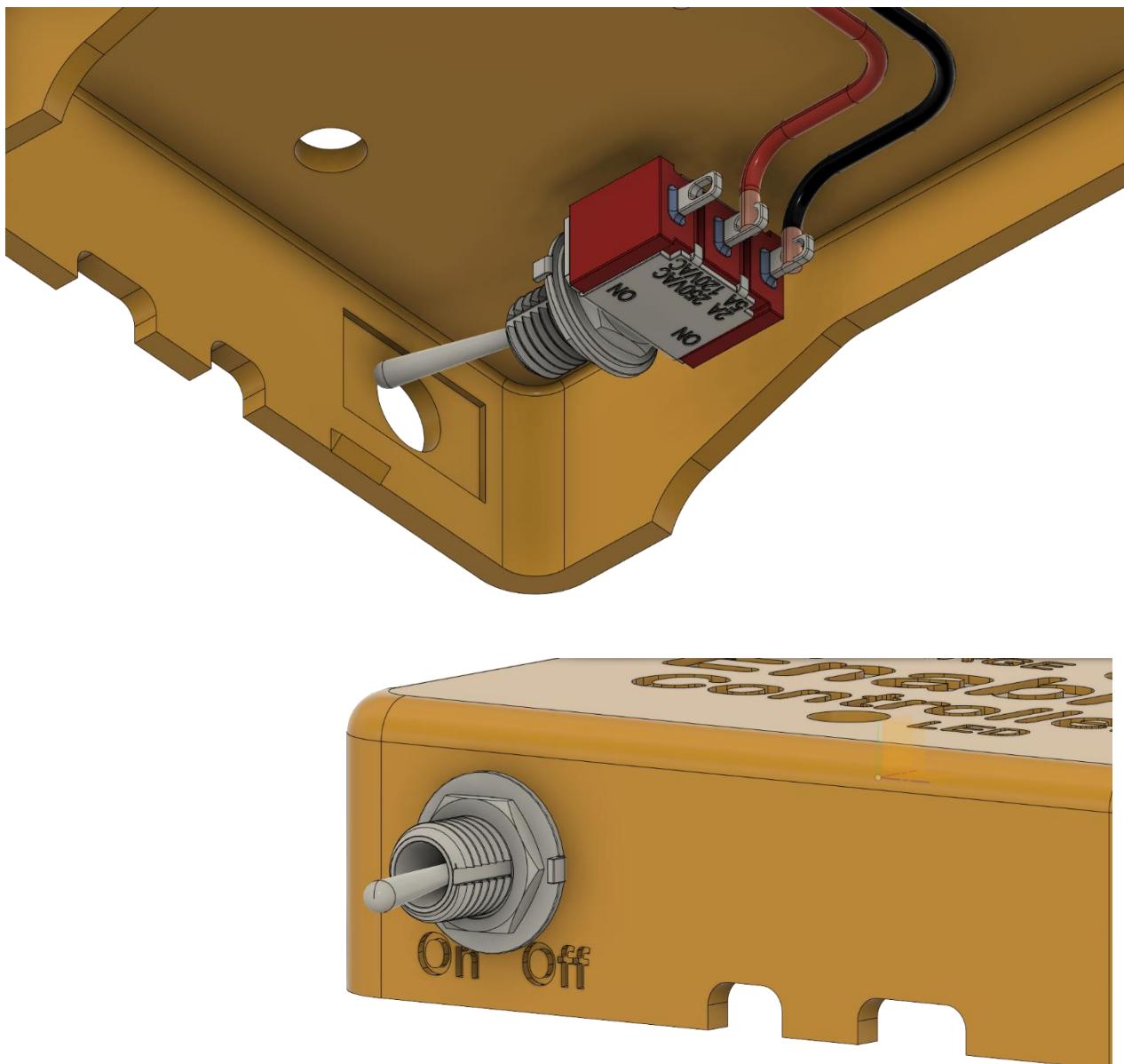
**Step 31 (Optional):** Use a cutter and carefully cut the trace between the solder pads.



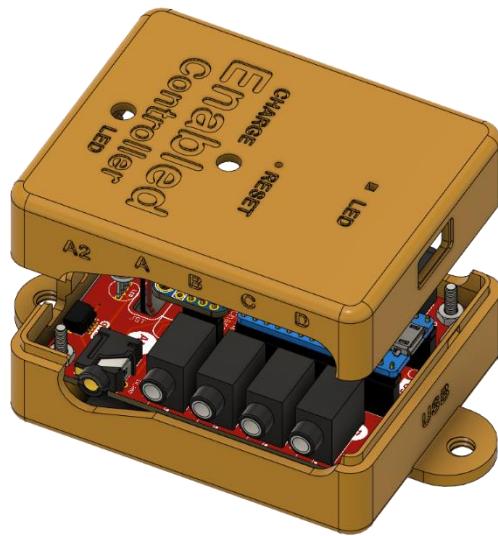
**Step 32 (Optional): Solder the LI-BATTERY Backpack to Power Switch x 1.**



**Step 33 (Optional): Insert the Power toggle switch into the top enclosure piece.**



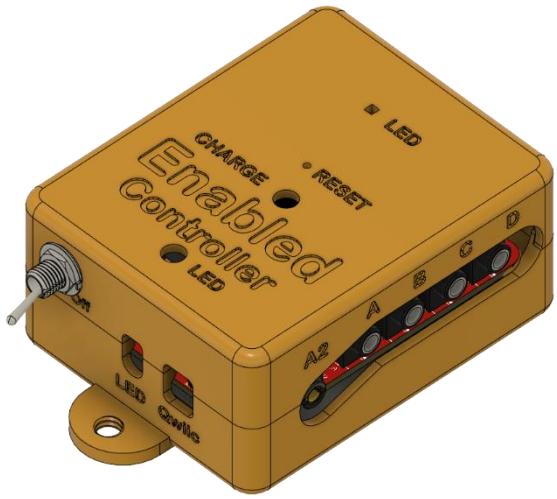
**Step 34 (Optional): Secure the Power toggle switch using the retainer ring.**



**Step 35: Insert the top enclosure over the bottom enclosure and snap them together.**



**Step 36: Connect a Micro USB-B cable through USB port to program the device.**



**Step 37 – That's it! Your Enabled-Controller-Wireless is fully assembled and ready to use.  
ready to use.**

## Joystick Assembly

Enabled-Controller-Wireless includes two 4 conductors input 3.5mm jacks reserved to connect up to two dual axis joysticks. The order of pins in joystick boards can vary depending on the brand but the following general configuration can be used to attach analog joystick. Enabled-Controller-Wireless is designed to work with analog joystick which rated to work with 3.3V input voltage.

### 4 conductor plug (TRRS) Joystick Connection

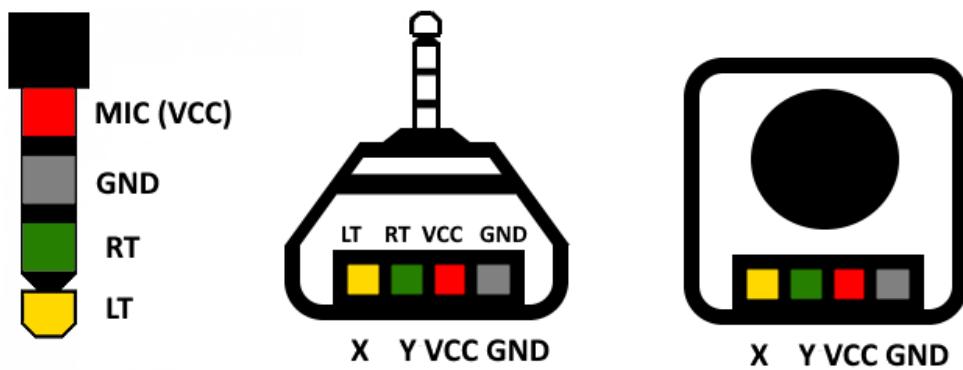


Figure 1: 4 Conductor plug (TRRS) pin mapping

The following table can summarize figure 1 as following:

PIN	FUNCTION
1 (Sleeve)	Microphone – VCC (3.3V)
2 (Ring 2)	GND
3 (Ring 1)	RT(Right Audio Channel)
4 (Tip)	LT(Left Audio Channel)

Table 1: 4 Conductor plug (TRRS) pin mapping