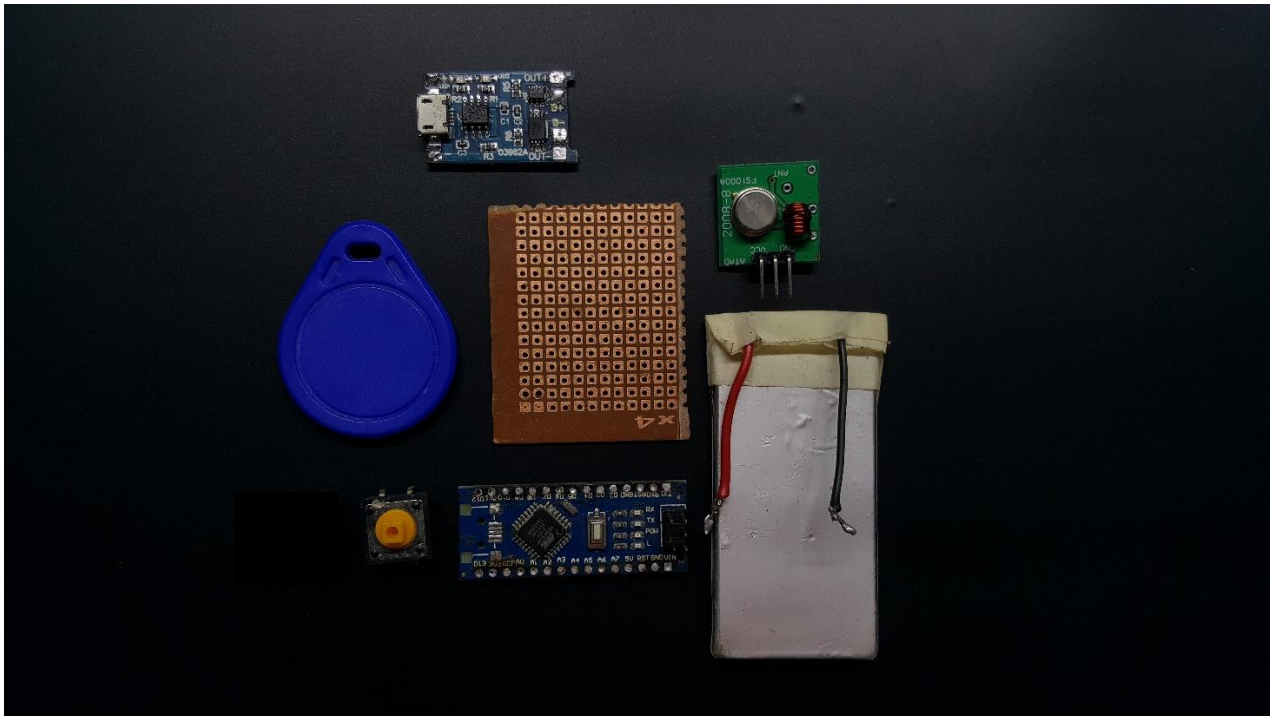
4.1: Making the key fob

4.1.1:

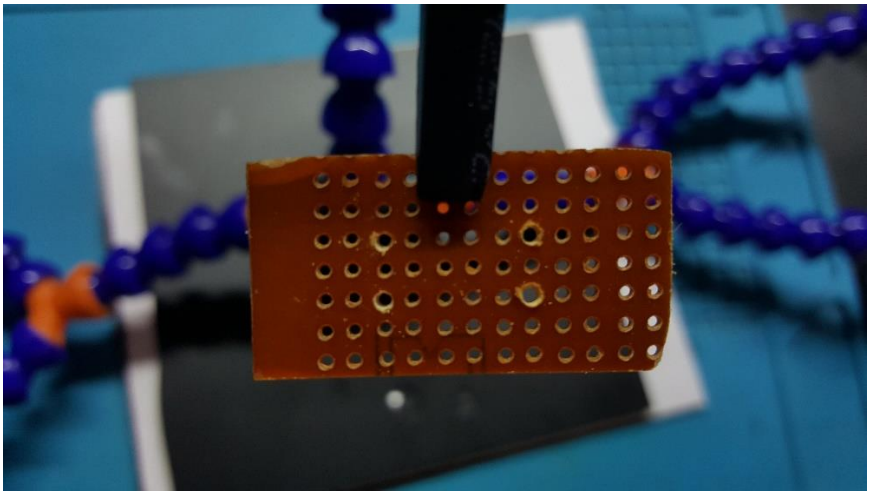
Gather the materials



Parts Used:

- 1 x Arduino Nano
- 1 x Perf Board
- 1 x RF 433mHz Transmitter module
- RFID Tag
- TP4056
- 3.7V Li-po battery (I have used a 250mAh battery instead of the one shown in the image).
- 1 x Push button

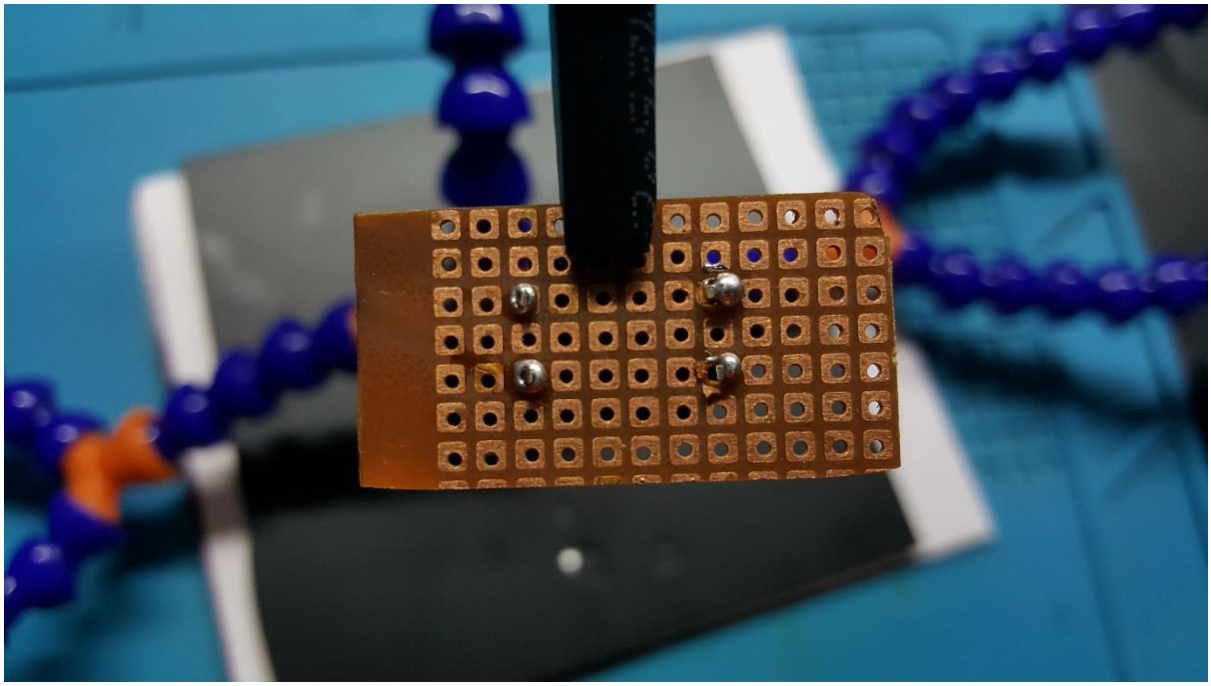
4.1.1: Prepare your PCB : Cut the PCB to a small size and make 4 holes for the button.



4.1.2:

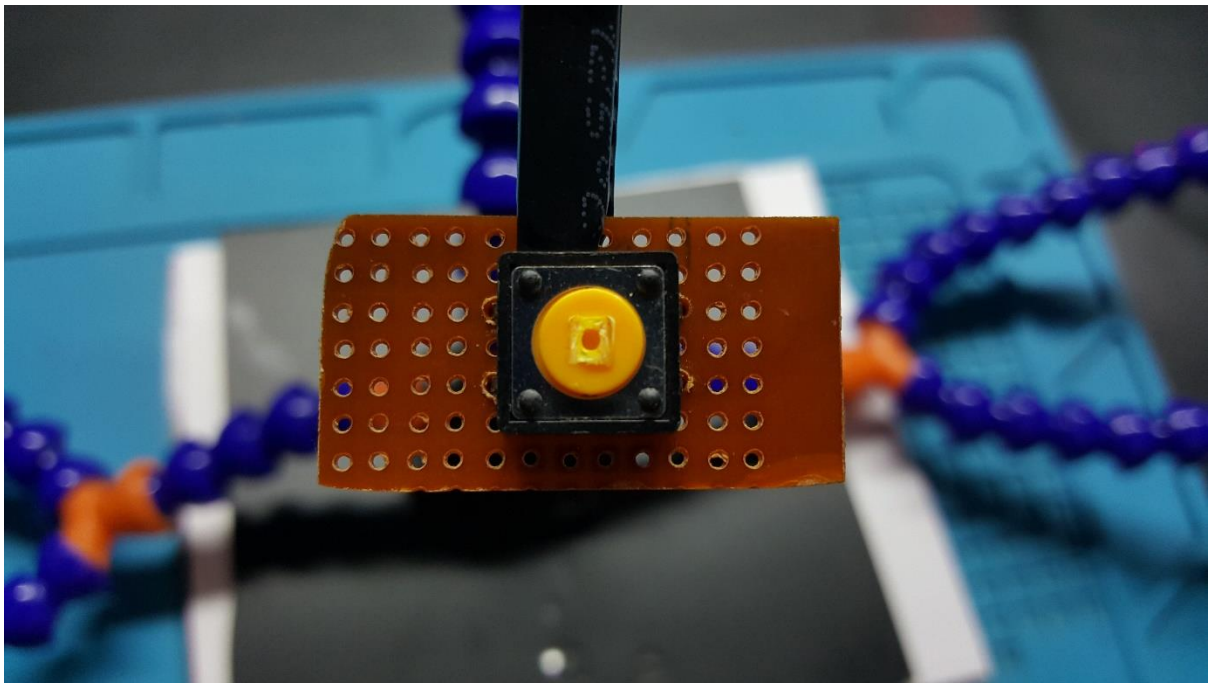
Solder the button : Place the button's pins in their respective holes and solder them.

Then, trim the excess pins of the button.



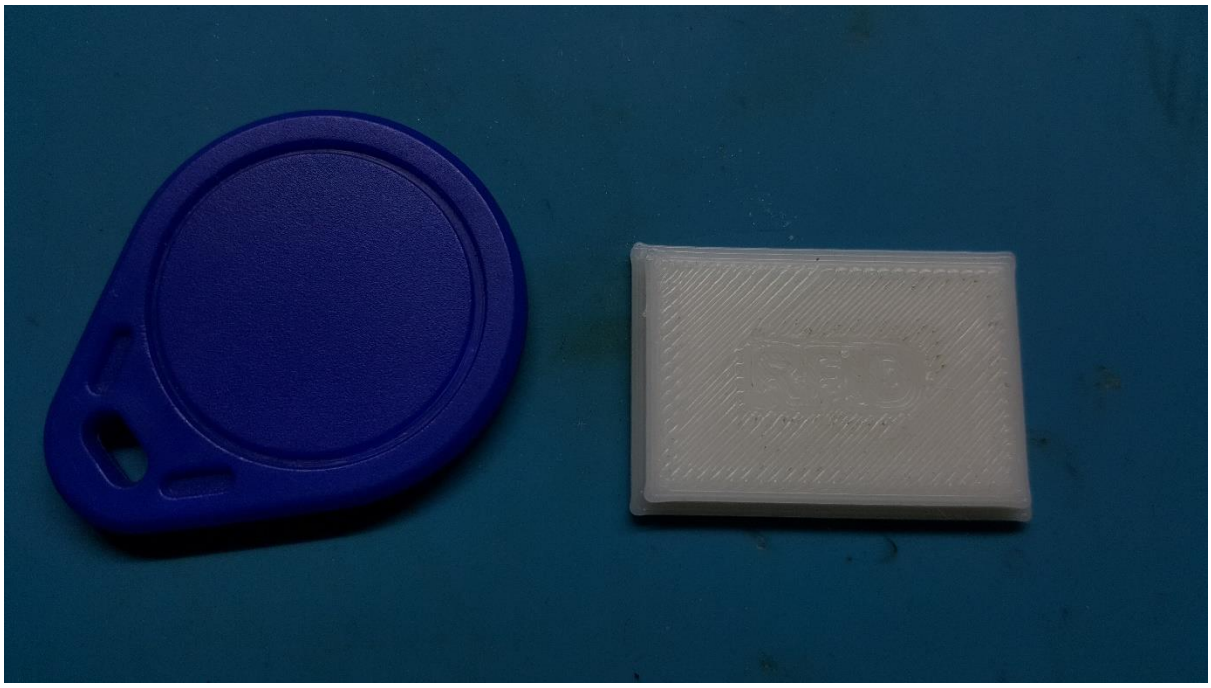
4.1.3:

Trim the top of the button: Trim the top of the button in order to reduce height.



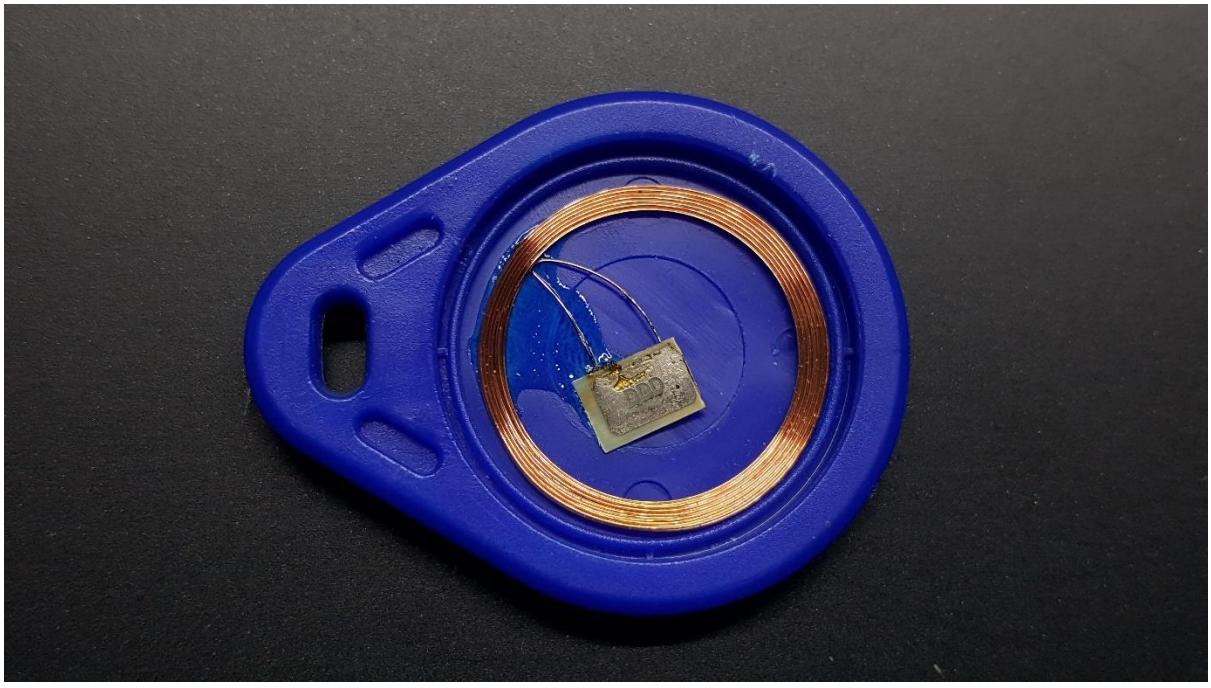
4.1.4:

Prepare the RFID and the 3D printed piece.

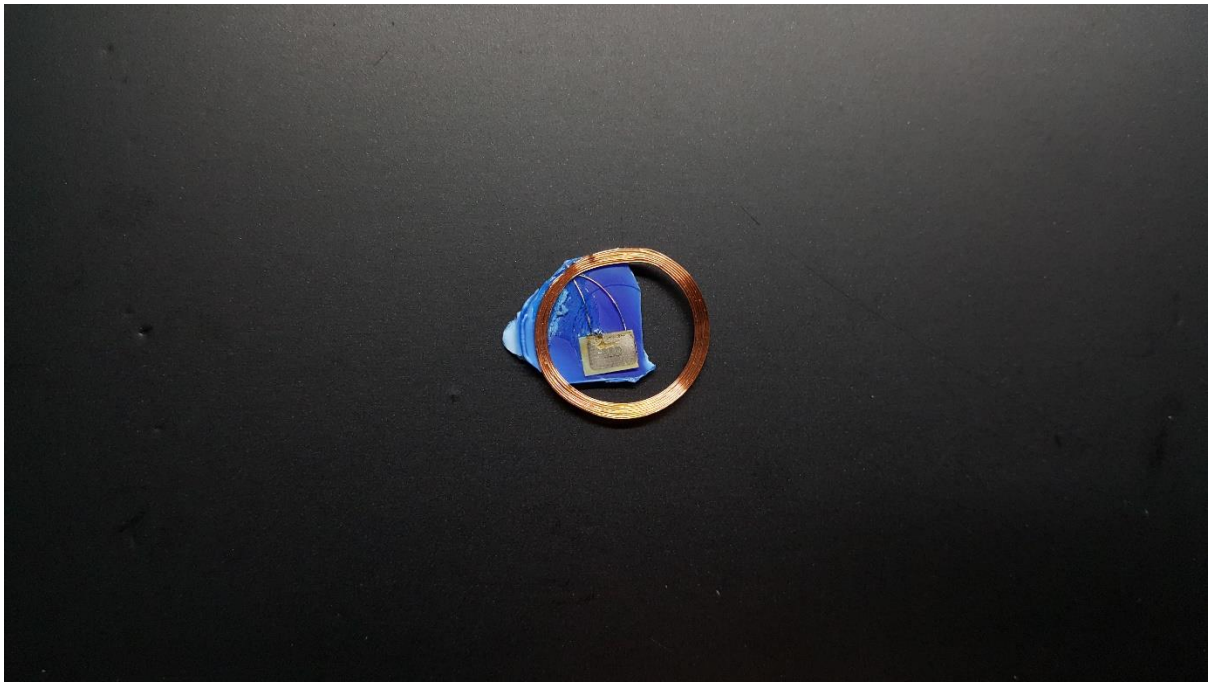


4.1.5:

Open the RFID tag : You need to be really careful when doing this part in order to avoid damaging the tag.

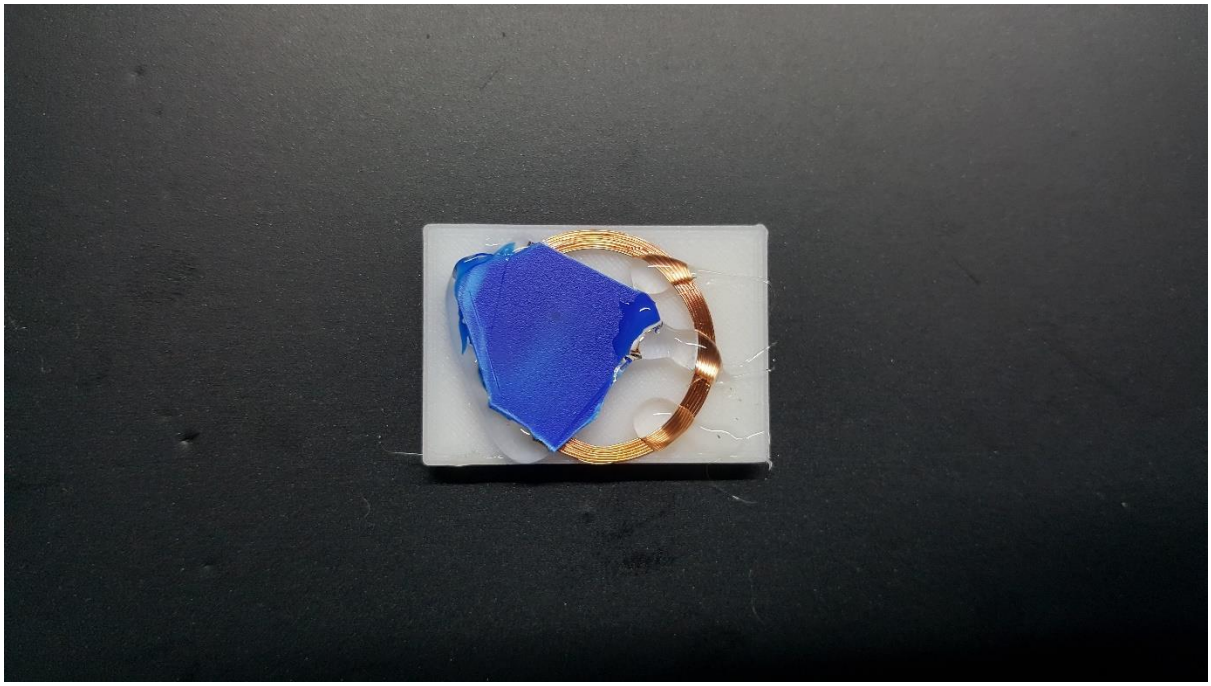


After opening it, I trimmed all of the plastic (blue), leaving the parts where it is glued.



4.1.6:

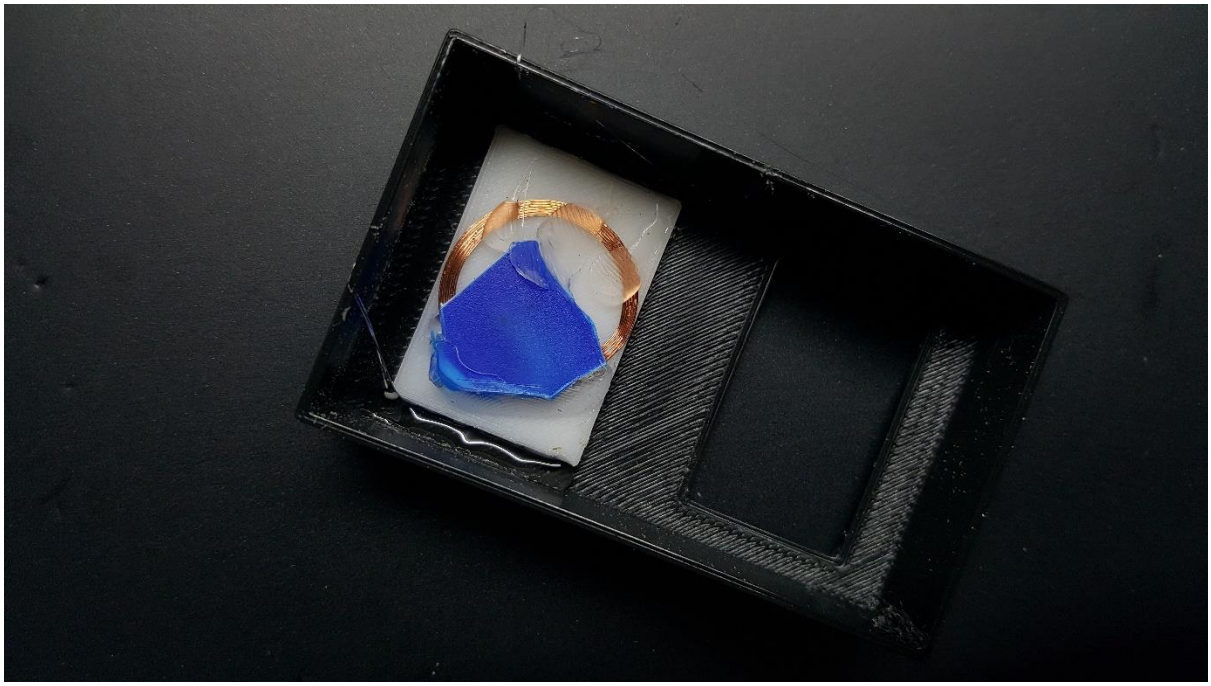
Securing RFID to the 3D printed piece.



Even though it doesn't look nice, it would make it much more safer for the chip inside.

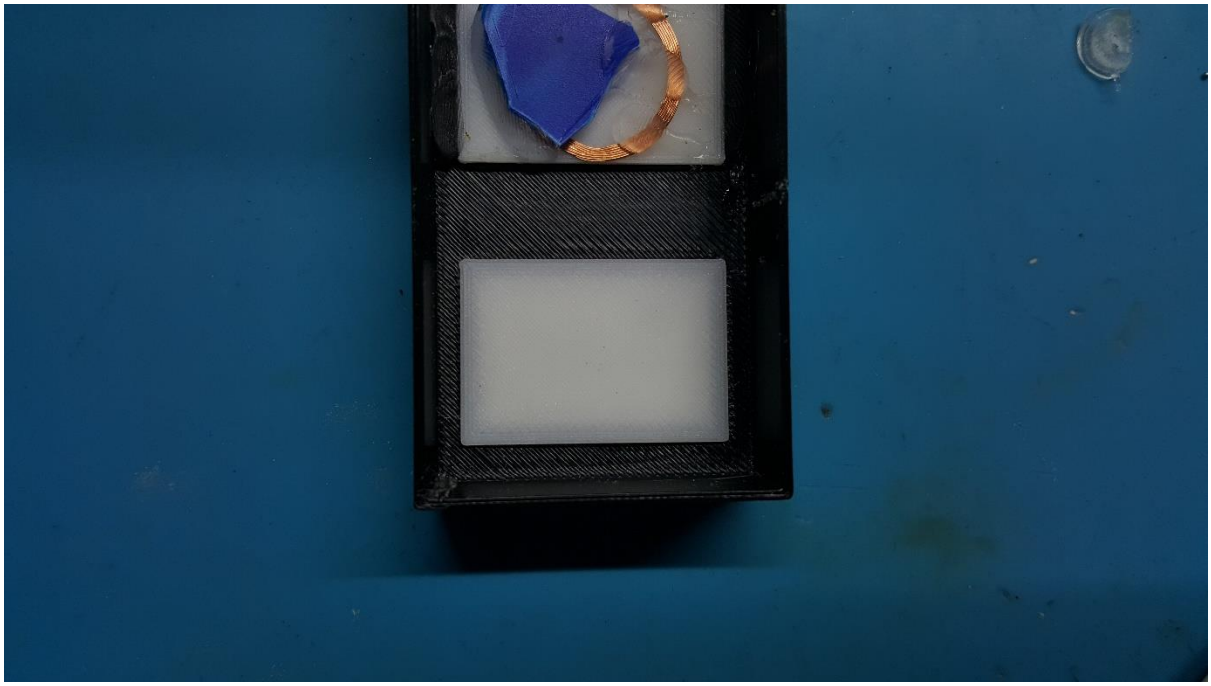
4.1.7:

Placing the piece inside the key fob.

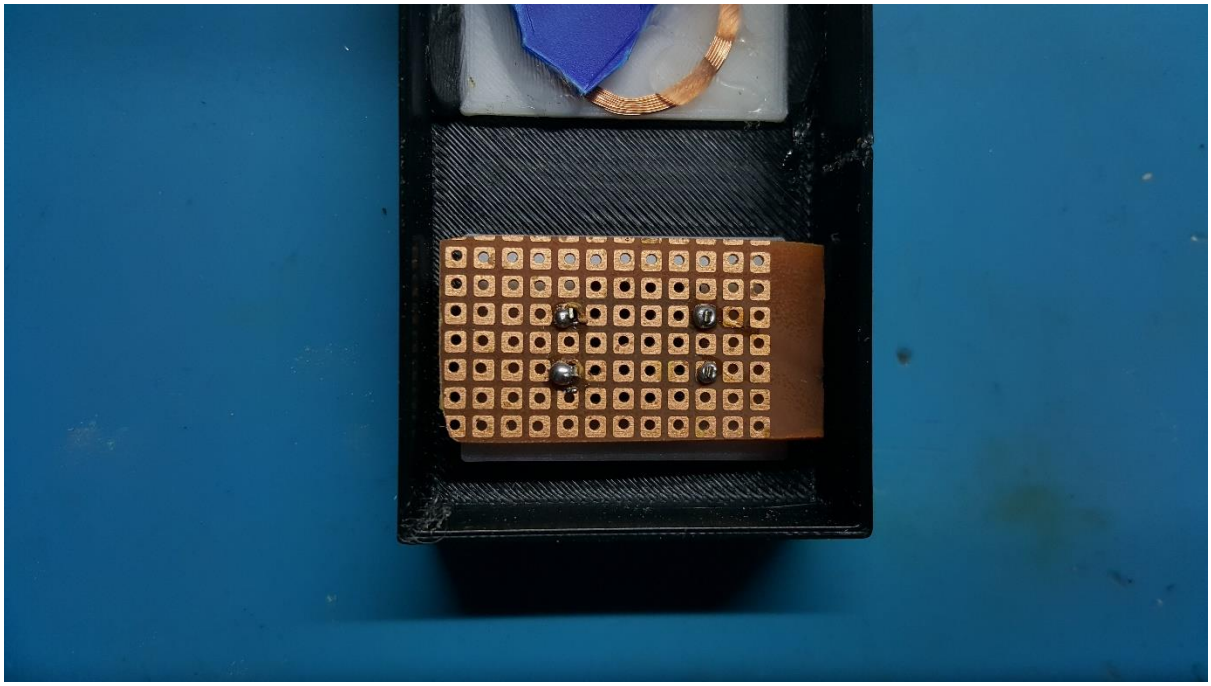


4.1.8:

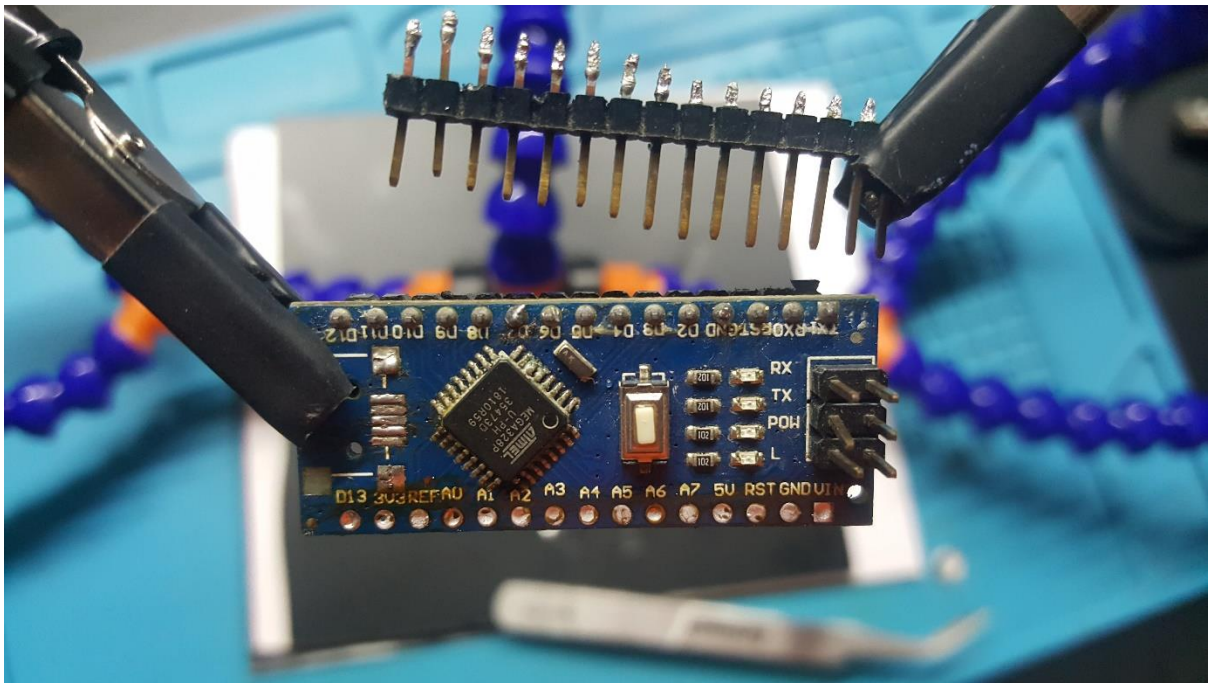
Place the other 3D printed piece inside the box

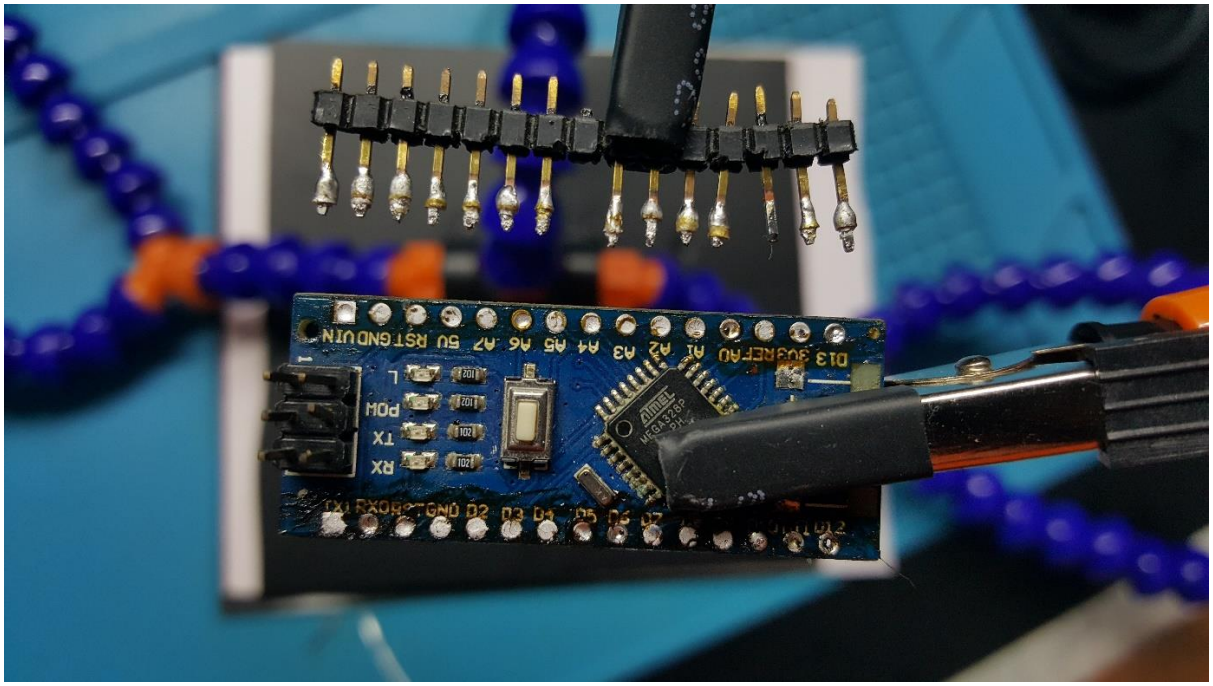


Now, glue the circuit (button) after placing it on top of this piece.



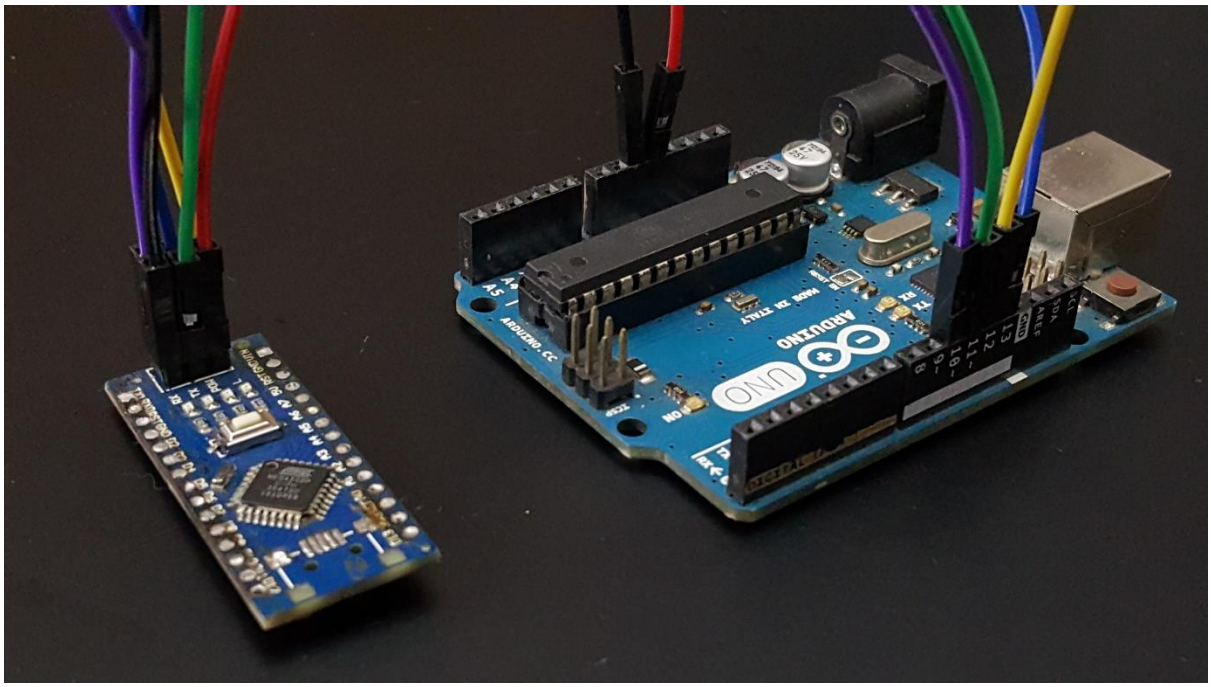
4.1.9: Prepare the board : Start by desoldering the pins.



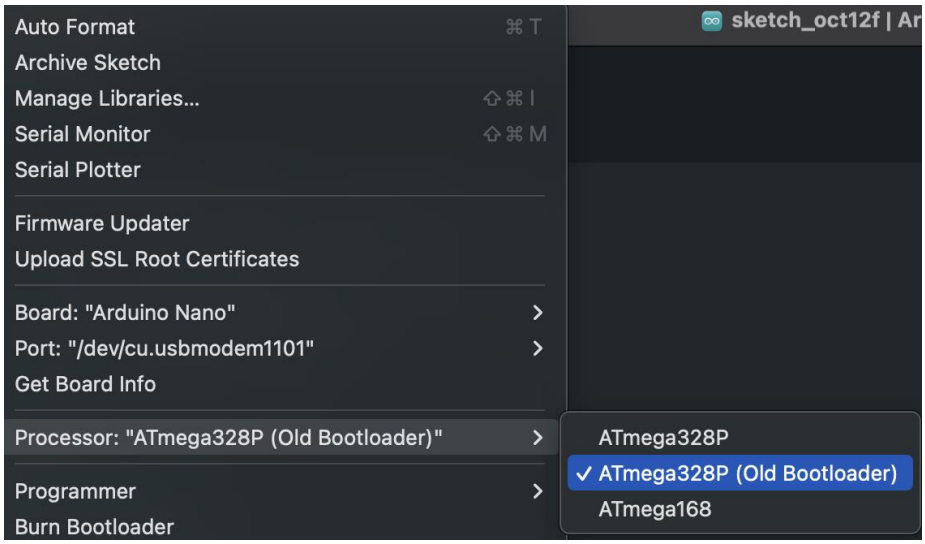


4.1.10:

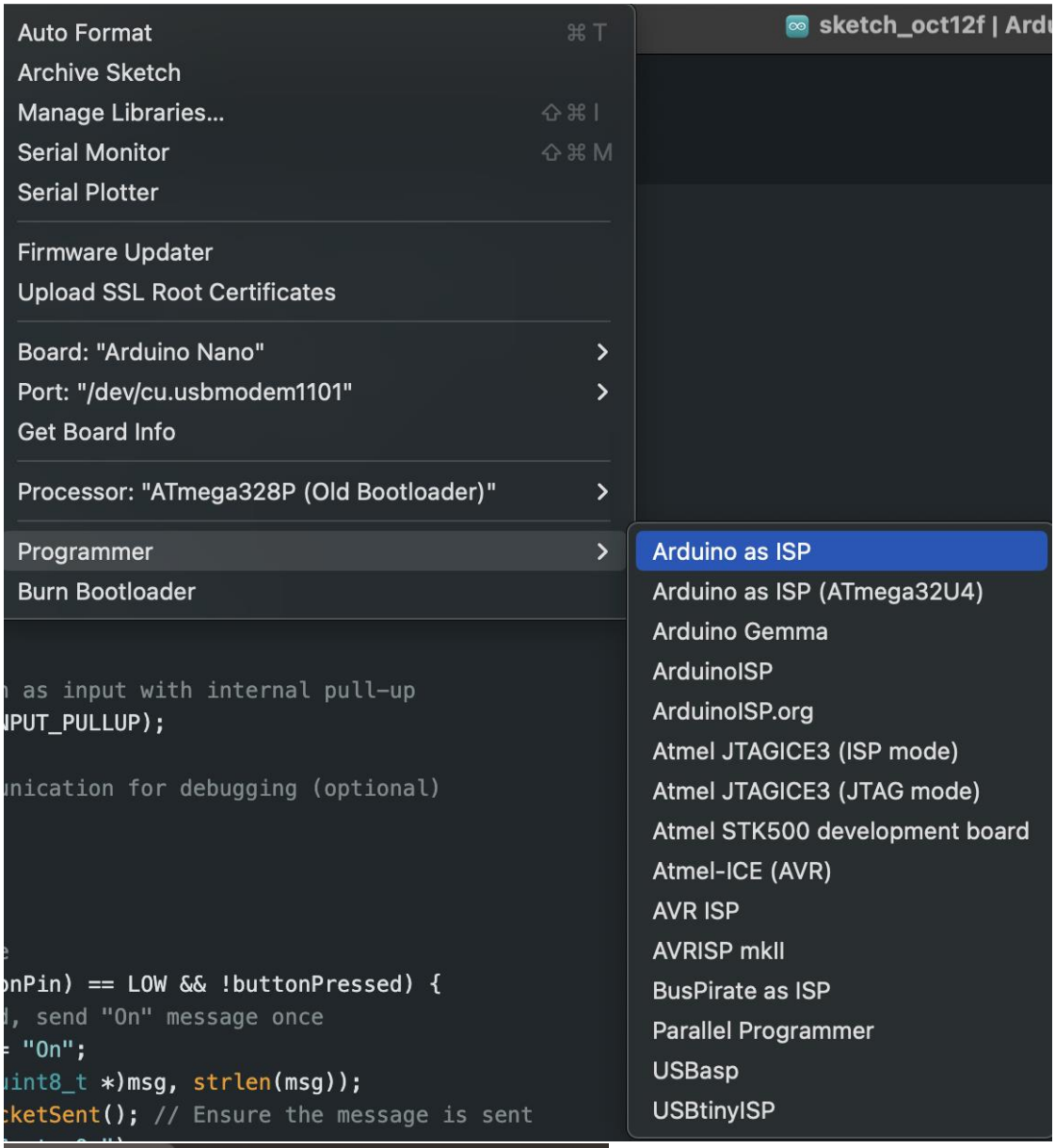
Programming : If you are like me, with an Arduino board without the USB port, then do check out ICSP programming. I have programmed mine with ICSP programming, using an Arduino UNO as a programmer

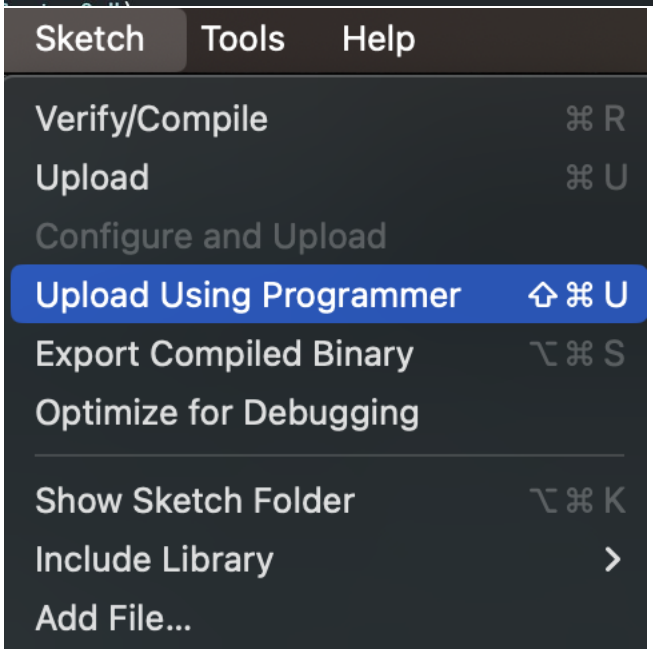


After uploading the ArduinoISP code,



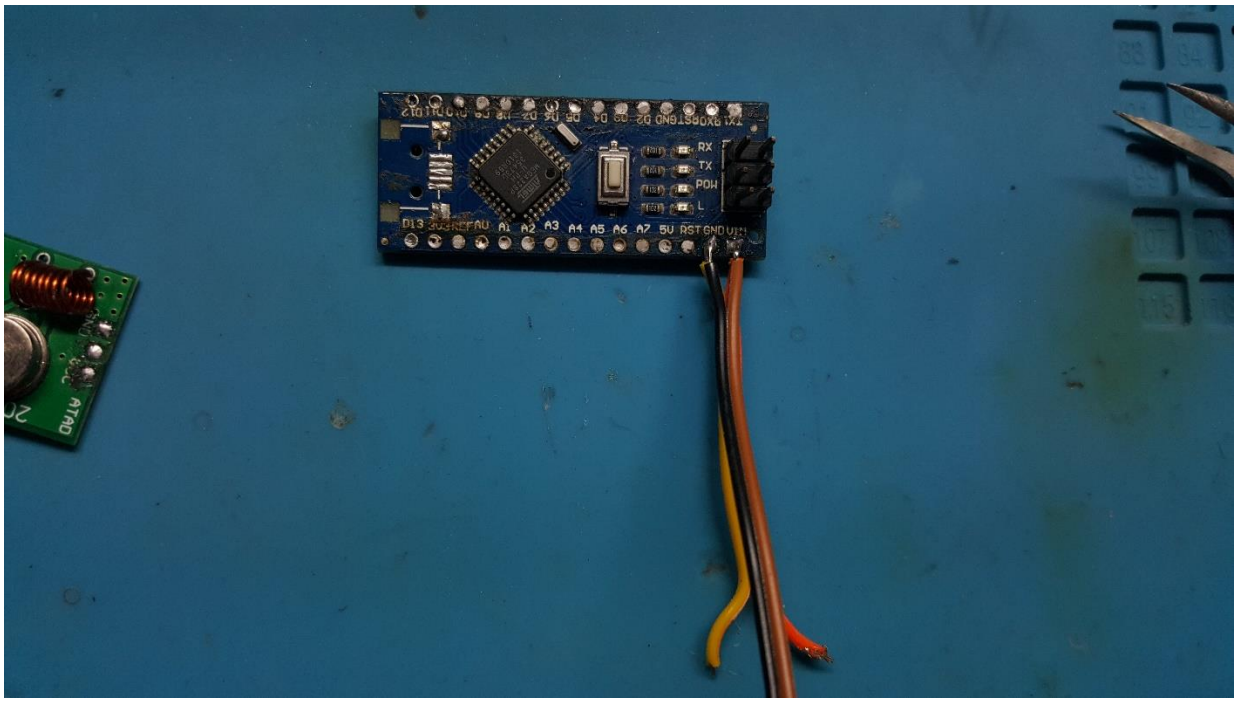
Select your right board, open the code to be uploaded and click on “Upload using programmer”

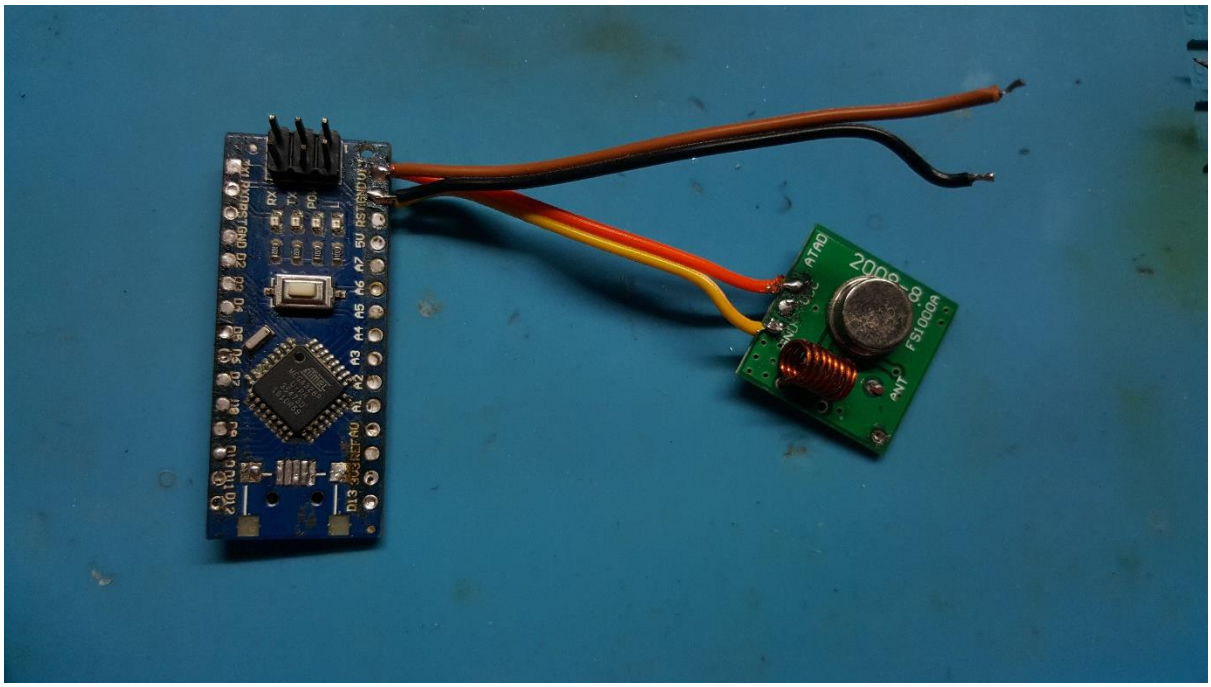

sketch_oct12f | Ardu
Auto Format ⌘T
Archive Sketch
Manage Libraries... ⇧⌘I
Serial Monitor ⇧⌘M
Serial Plotter
Firmware Updater
Upload SSL Root Certificates
Board: "Arduino Nano"
Port: "/dev/cu.usbmodem1101"
Get Board Info
Processor: "ATmega328P (Old Bootloader)"
Programmer
Burn Bootloader
Arduino as ISP
Arduino as ISP (ATmega32U4)
Arduino Gemma
ArduinoISP
ArduinoISP.org
Atmel JTAGICE3 (ISP mode)
Atmel JTAGICE3 (JTAG mode)
Atmel STK500 development board
Atmel-ICE (AVR)
AVR ISP
AVRISP mkII
BusPirate as ISP
Parallel Programmer
USBasp
USBtinyISP
as input with internal pull-up
PUT_PULLUP);
nication for debugging (optional)
nPin) == LOW && !buttonPressed) {
, send "On" message once
= "On";
int8_t *)msg, strlen(msg));
ketSent(); // Ensure the message is sent

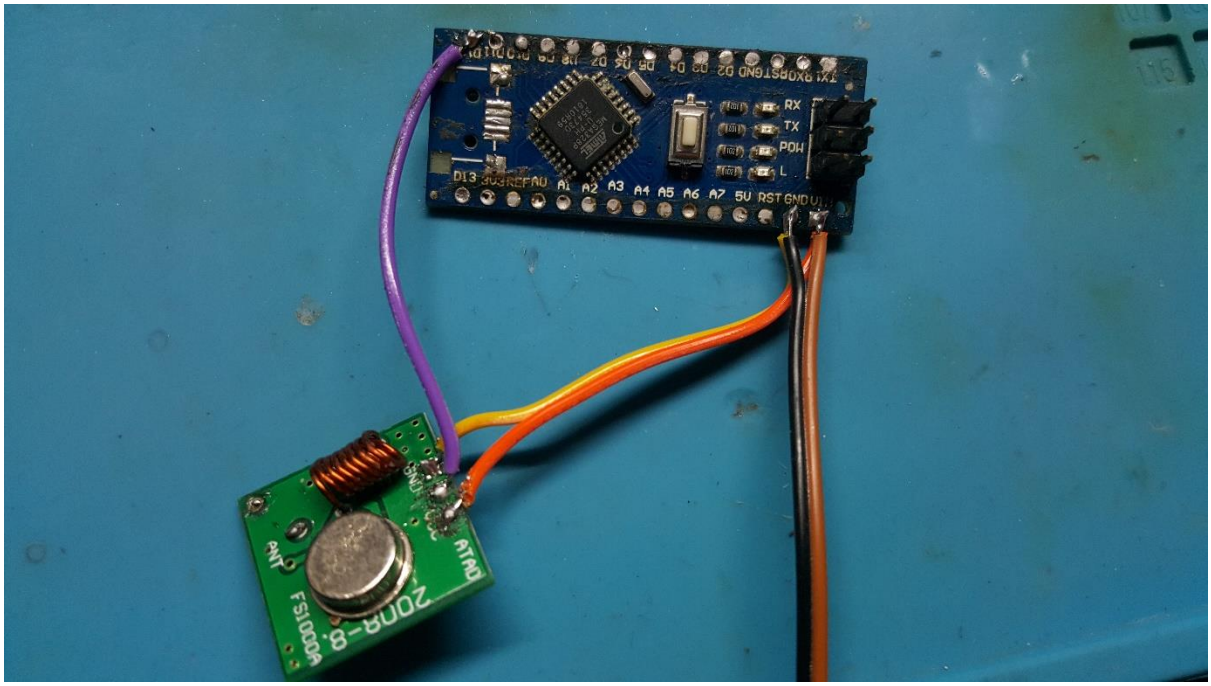

Sketch
Tools
Help
Verify/Compile ⌘R
Upload ⌘U
Configure and Upload
Upload Using Programmer ⇧⌘U
Export Compiled Binary ⌥⌘S
Optimize for Debugging
Show Sketch Folder ⌥⌘K
Include Library
Add File...

4.1.11:

Soldering the Transmitter module to the Arduino

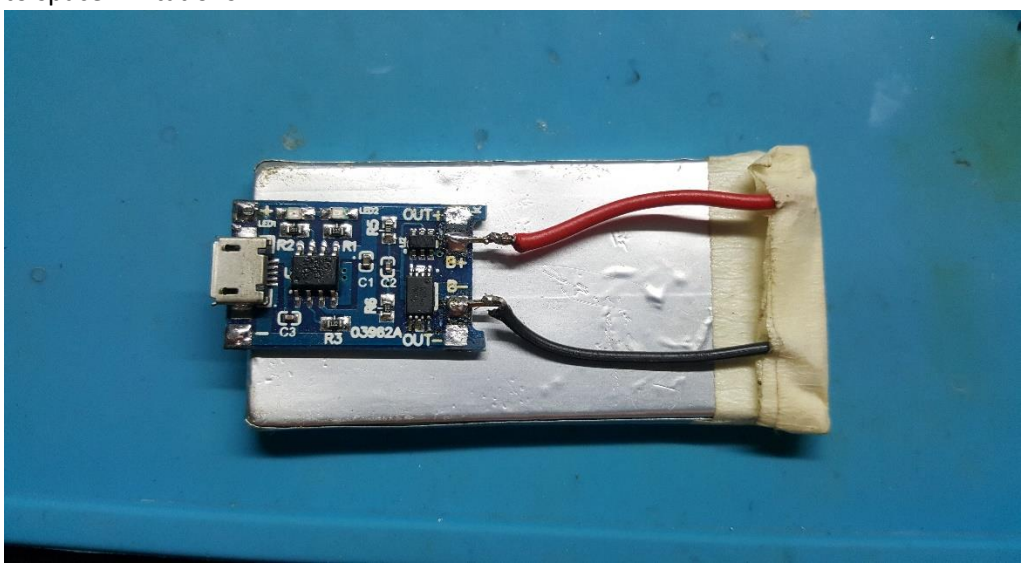






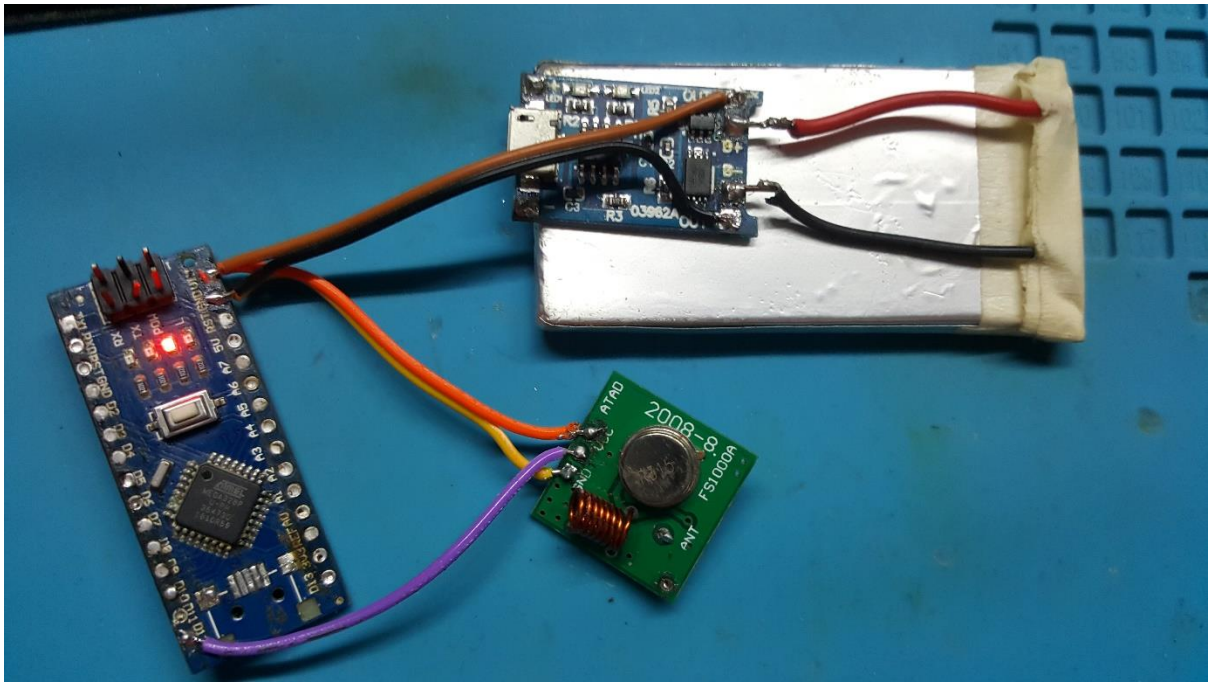
4.1.12:

Solder TP4056 with Battery : Even though the battery is this one, It was changed later due to space limitations



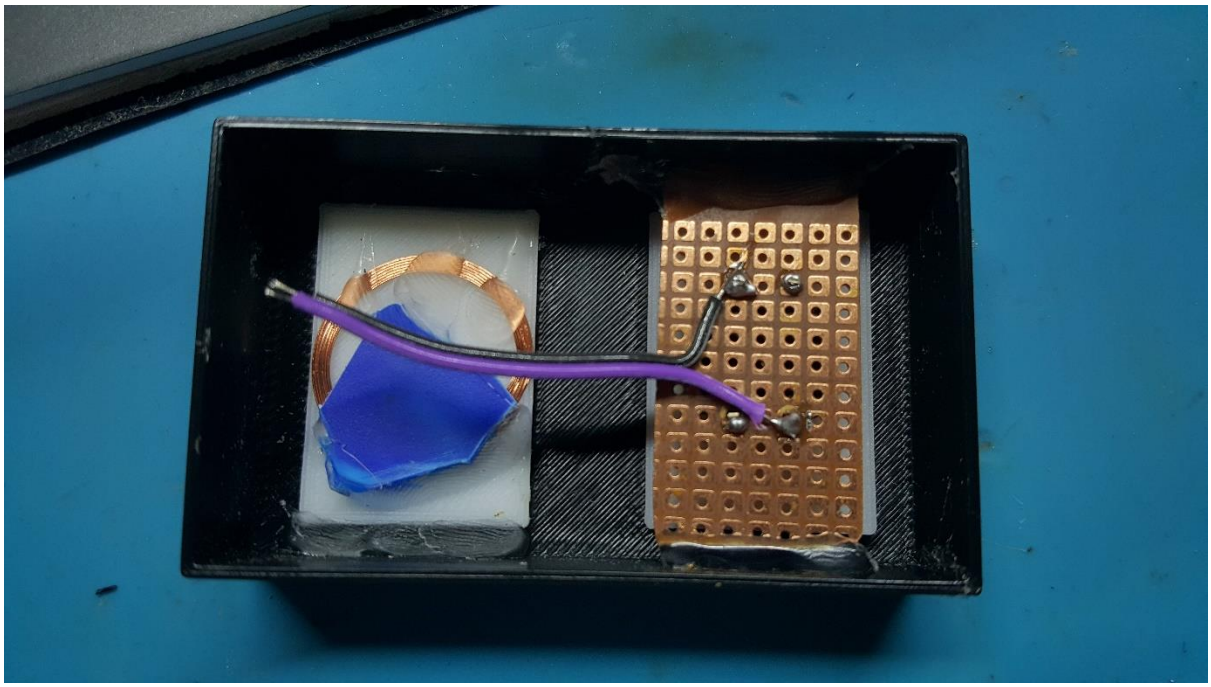
4.1.13:

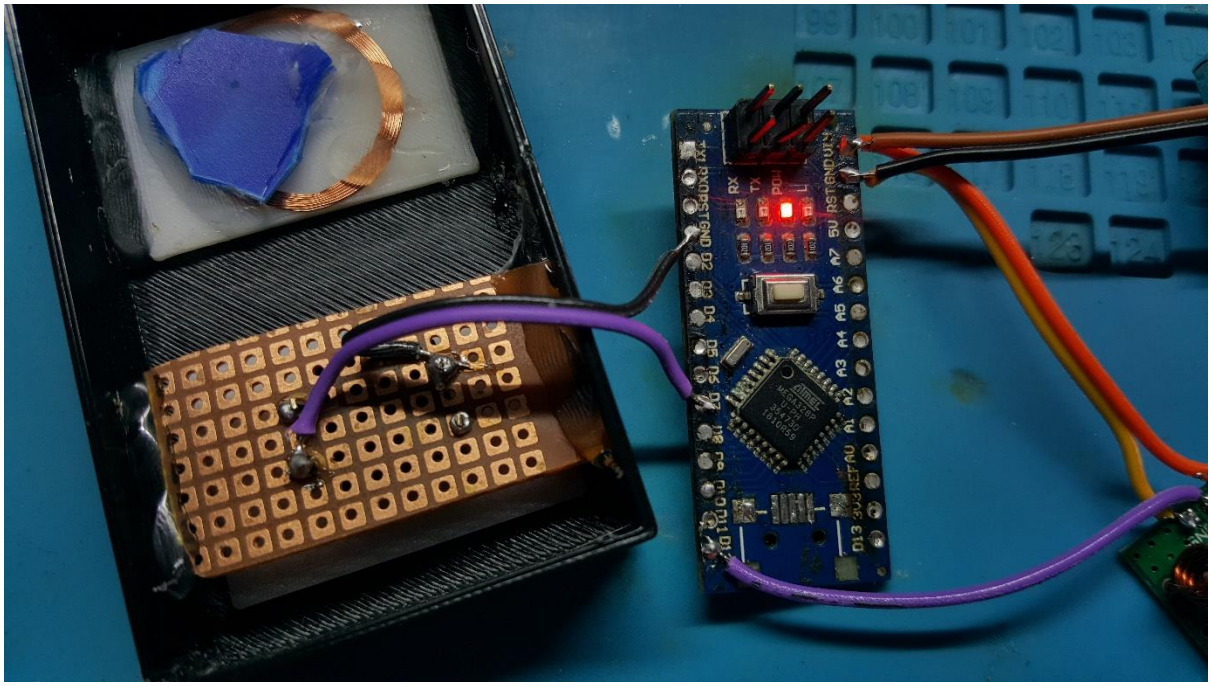
Solder the Arduino Nano with the TP4056



4.1.14:

Solder wires to the button and then to D7 and GND to the Arduino Nano board.



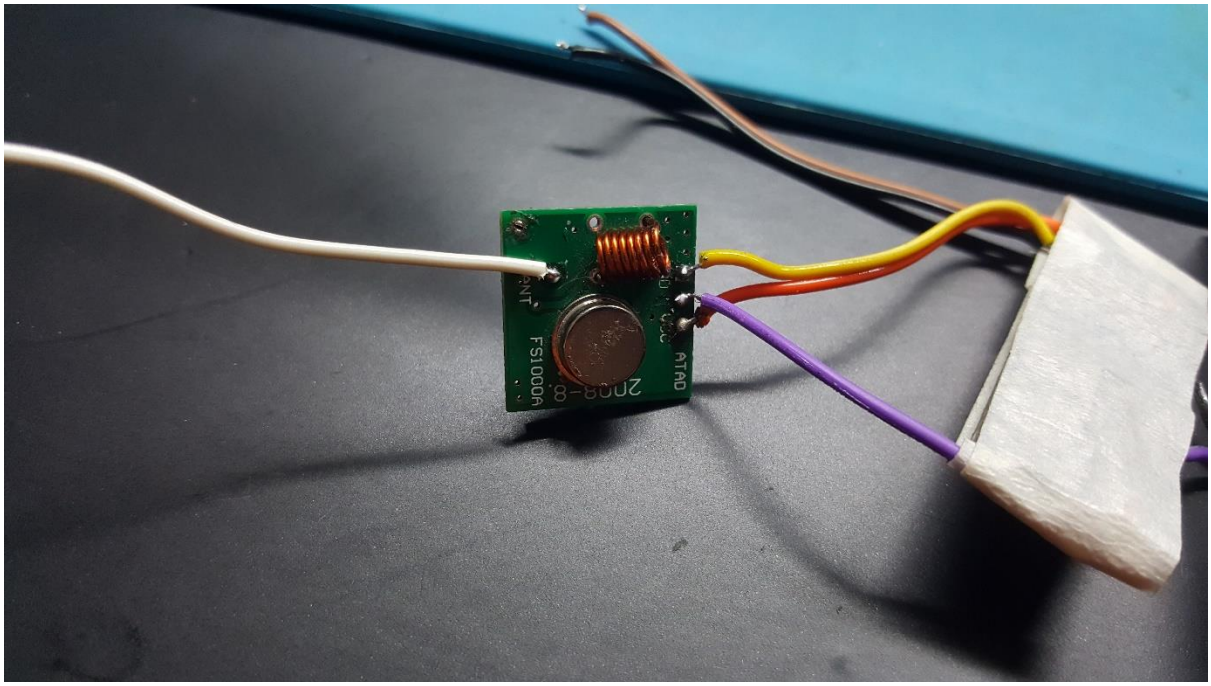


4.1.15:

Prepare the antenna : Cut 17.3cm of wire

4.1.16:

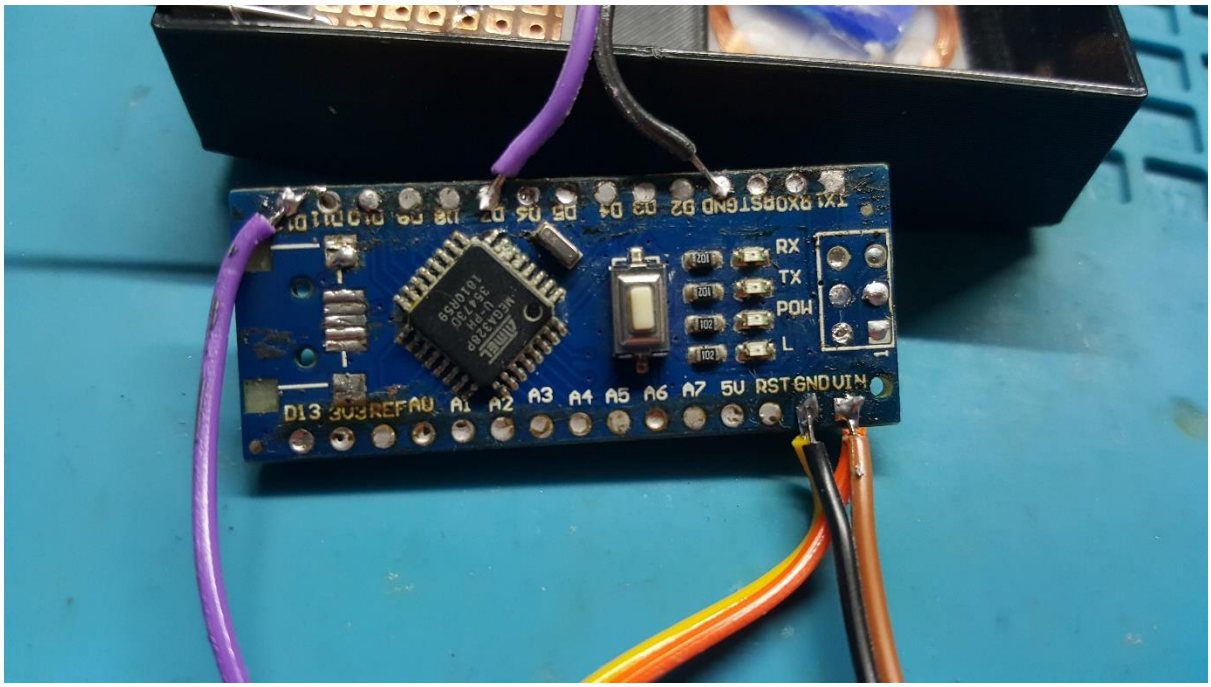
Solder the antenna to the transmitter board



4.1.17:

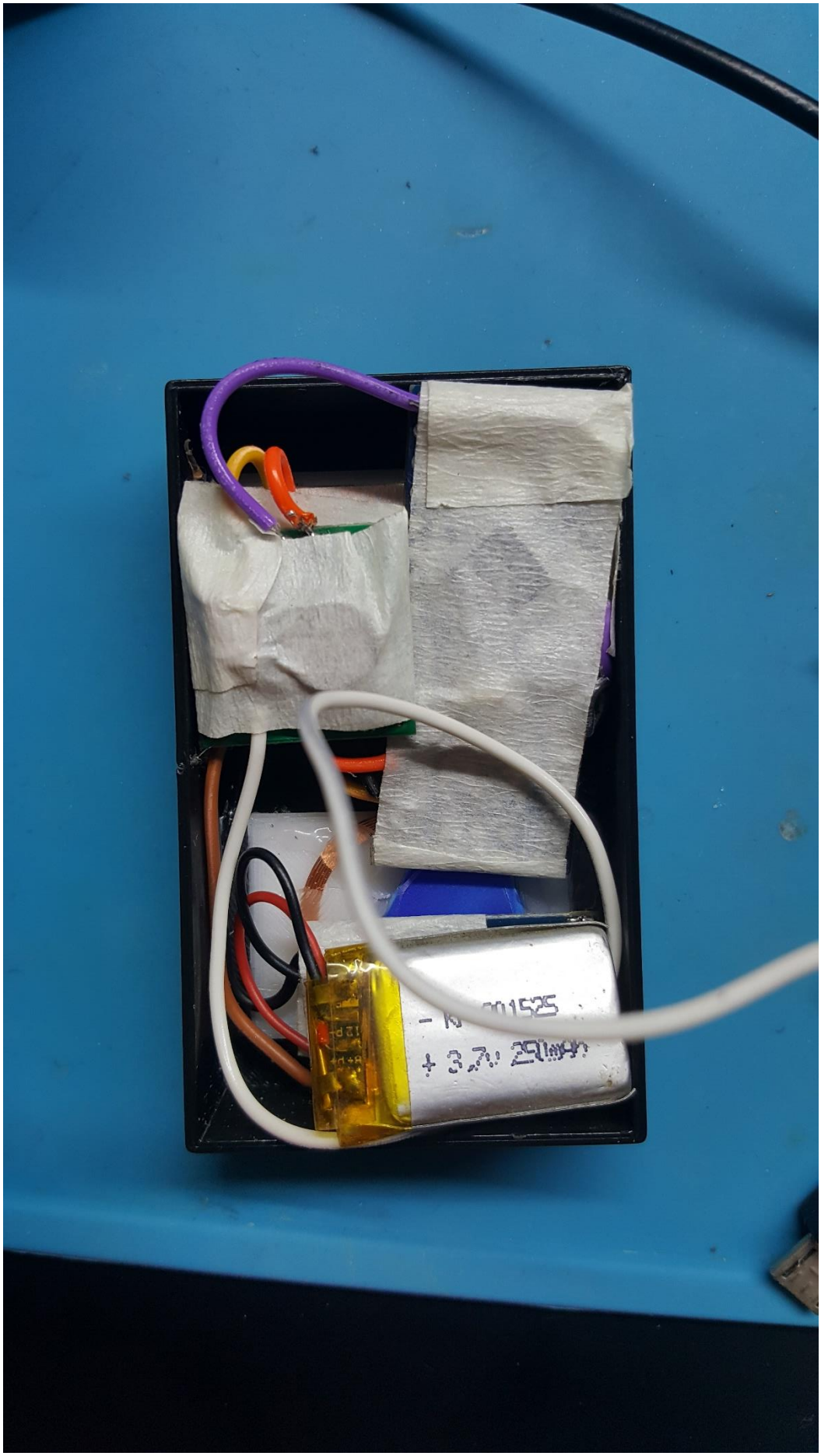
Disconnect the battery and desolder the ICSP pins (if it works correctly only).

A code has been attached in the instructable at 4.1 regarding the testing of RF and testing of RFID functionalities.



4.1.18:

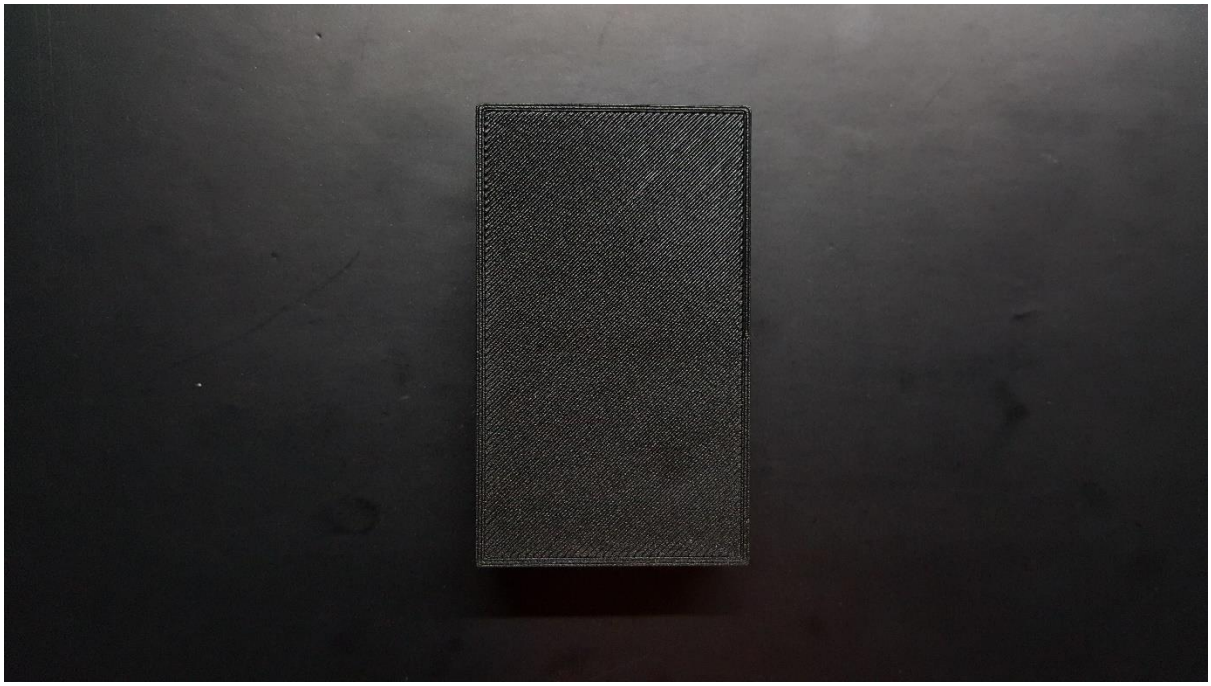
Cover everything with masking tape and arrange everything in the key fob.



The battery has been changed due to lack of space with the old one.

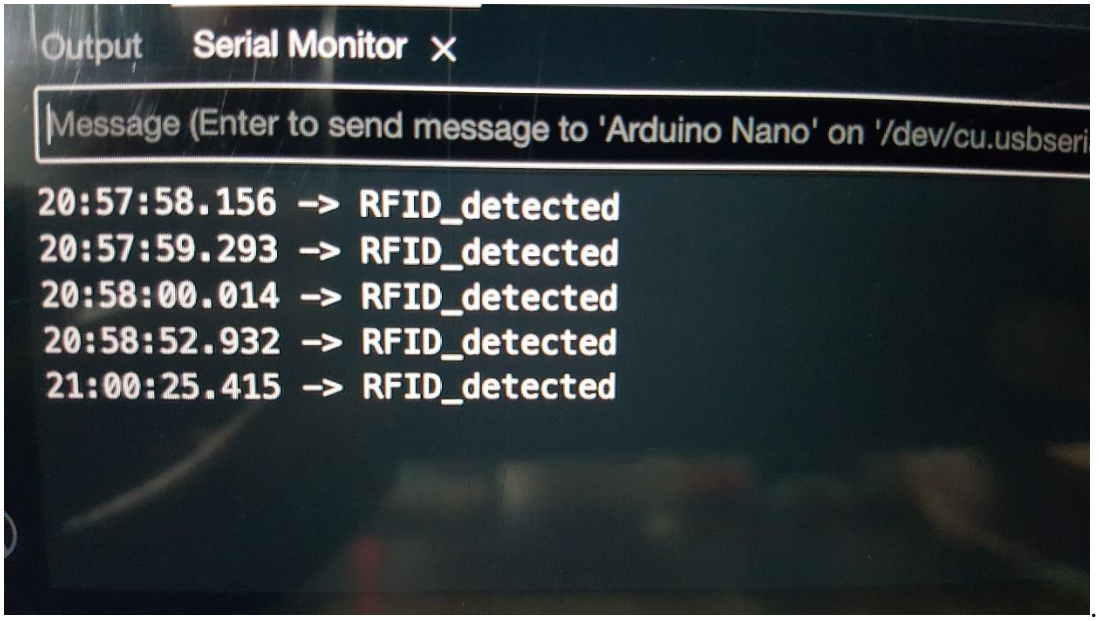
4.1.19:

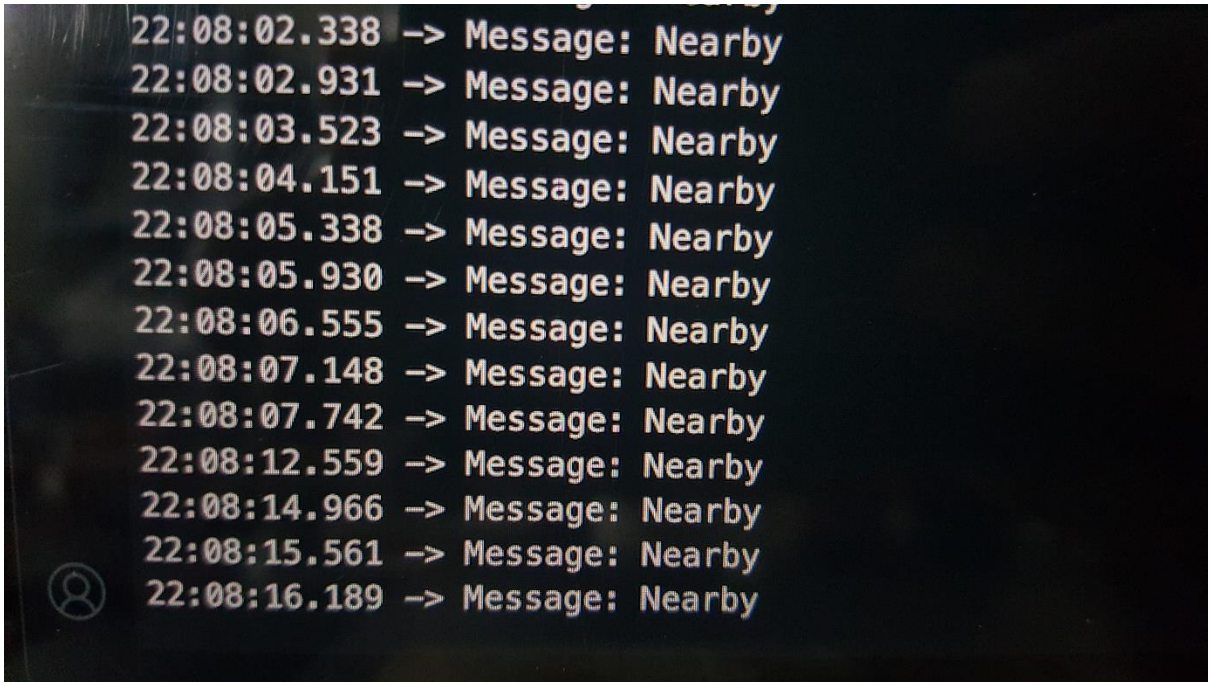
Put the back panel.



Testing :

Even though everything works, the RF only works within 1m of range. I am planning to fix this later





Even though it is a success, I wasn’t much satisfied with the range.