



MAC Address Finder

Created by Ron Dear



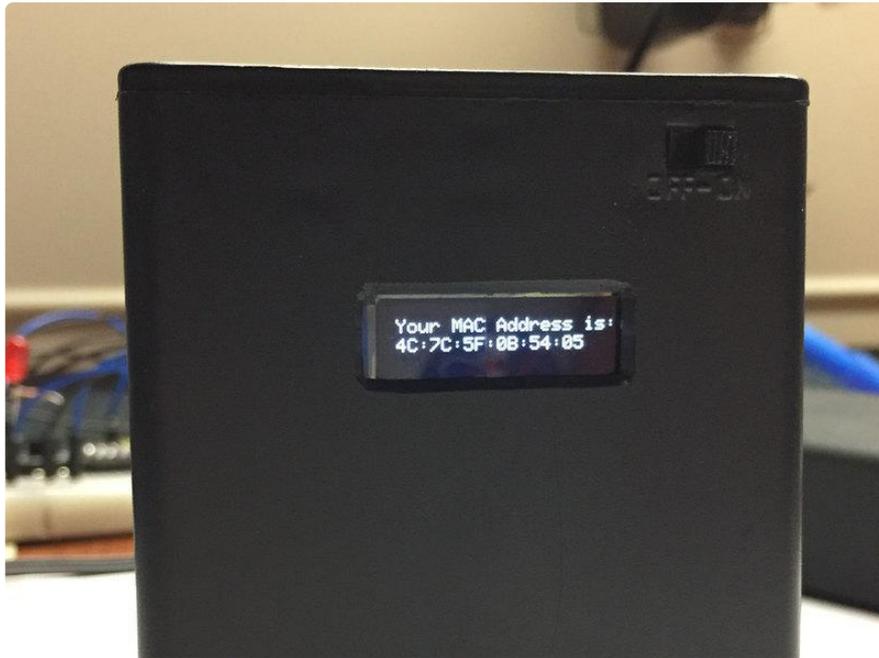
<https://learn.adafruit.com/mac-address-finder>

Last updated on 2023-08-29 03:15:32 PM EDT

Table of Contents

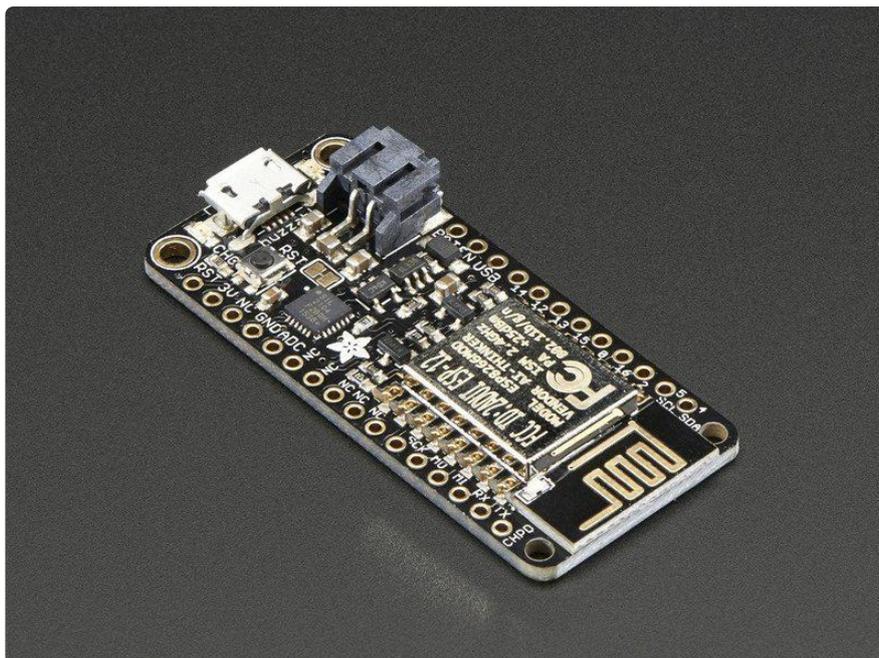
Overview	3
Parts and Tools	4
Assembly of the Main Board	5
Preparing the Case	8
Connecting the Battery and Switch	10
The Code	12
Putting it all Together	15

Overview



Sometimes finding the MAC address of a wireless device is easy and sometimes it's not. With this project you can find the MAC address of any wireless device in just a few seconds.

What makes this project so simple is the Feather Huzzah. With a few lines of code you can create an access point that other wireless devices can join. Then the Feather will print out the MAC address of that device!



In just a short amount of time you can have a device that is a welcome addition to any IT person's tool belt.

Parts and Tools

Let's start by getting all the parts together.



For this project I used the following parts:

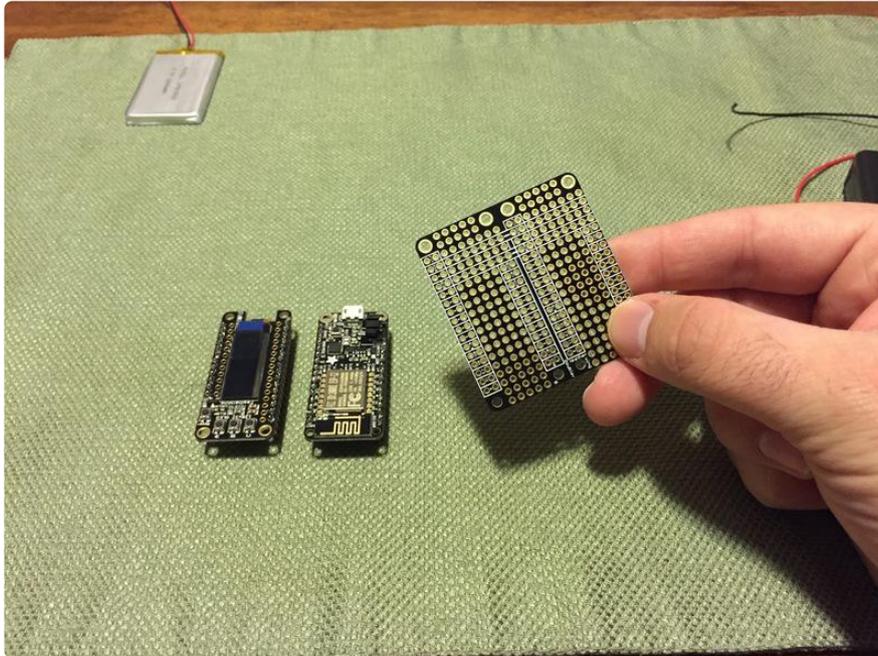
- [Feather Huzzah \(\)](#)
- [FeatherWing OLED \(\)](#)
- [Lithium Polymer Battery \(\)](#)
 - Pictured is a 1200 mAh battery but I ended up using a 500 mAh battery I had as it fit inside the case better
- [FeatherWing Doubler \(\)](#)
 - I could have stacked the OLED and Huzzah but I like the way they look next to each other on the Doubler
- [4 x AA battery case \(\)](#) that I had and was not using
 - I could have 3D printed a case, and might still at some point, but I wanted to put this one to good use

You will also need something similar to the tools pictured above. I mainly used these tools to hack the case to make the Huzzah, OLED and battery fit inside so in reality you can use anything you have to cut away parts of the plastic case.

Assembly of the Main Board

There were several things I had to take into account to make it fit inside the case nicely. I played around with the arrangement of the components for a while before settling on the final layout.

We'll start with the Huzzah, OLED and the Doubler.



There's something to be aware of here. I wanted the OLED screen to be on the top, however, when you put it on the top the Lipo connector is on the inside of the board. What I did was arrange everything so that the connector faced out. The only issue with this is that then the OLED is upside down.

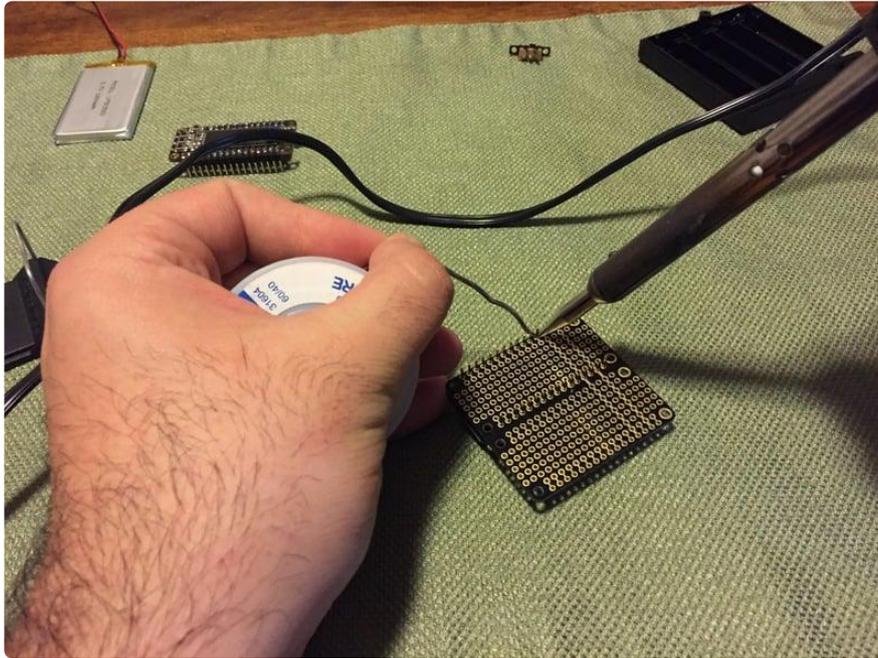


Notice in this image the OLED board is upside down.

Luckily for us there is a line of code we can put in to compensate for this and flip the text on the screen so everything is right side up again.

```
display.setRotation(2); //This rotates everything on the screen 180 degrees
```

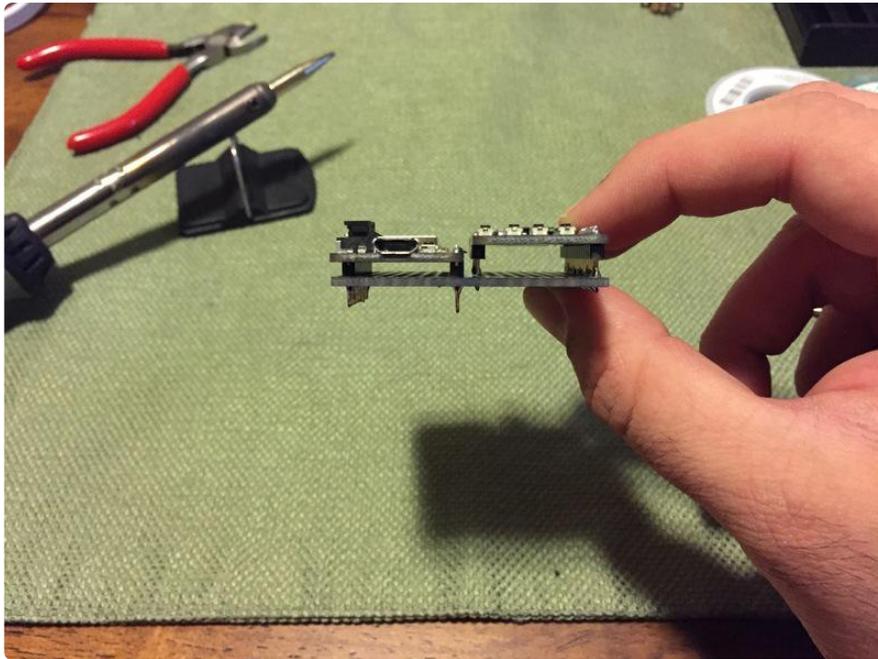
I started by flipping the doubler over and soldering the Huzzah into place.



One thing I forgot to do at the time, and so it's not shown here, is to solder wires for power and ground. There were very hard to thread through after I soldered the Huzzah and OLED into place.

One other consideration I made was the height of the OLED board. Since the Feather has a lipo connector, it is taller than the screen. I wanted the screen to be as close to the front of the case as possible so I couldn't just place it flush with the protoboard.

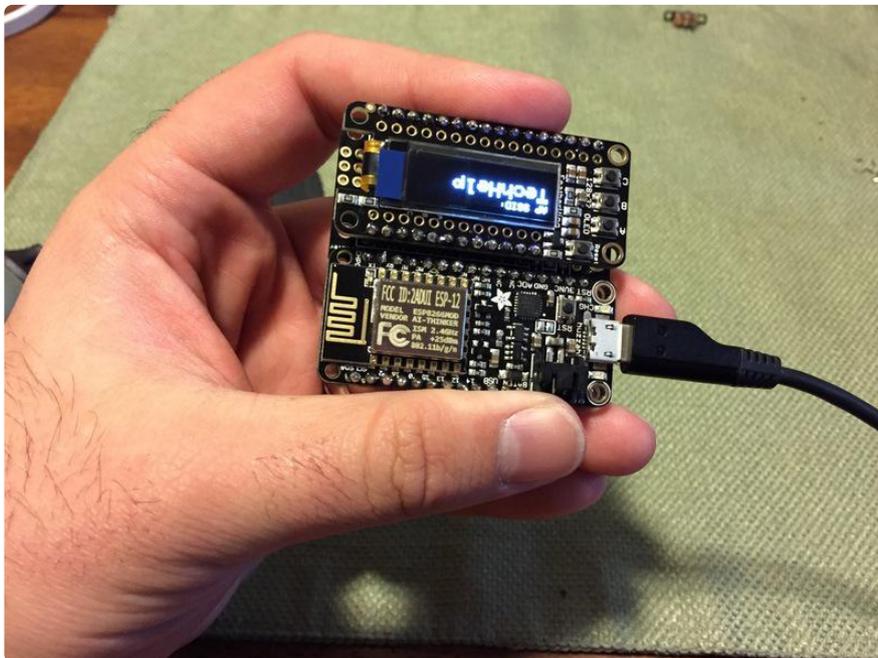
When I was done soldering the OLED board everything looked like this. Note that the OLED board is on the right in this image.



Once everything is put together I uploaded my code to see if everything worked.

Notice the the text on the screen is upside down. Again we'll address this in the code when we get there.

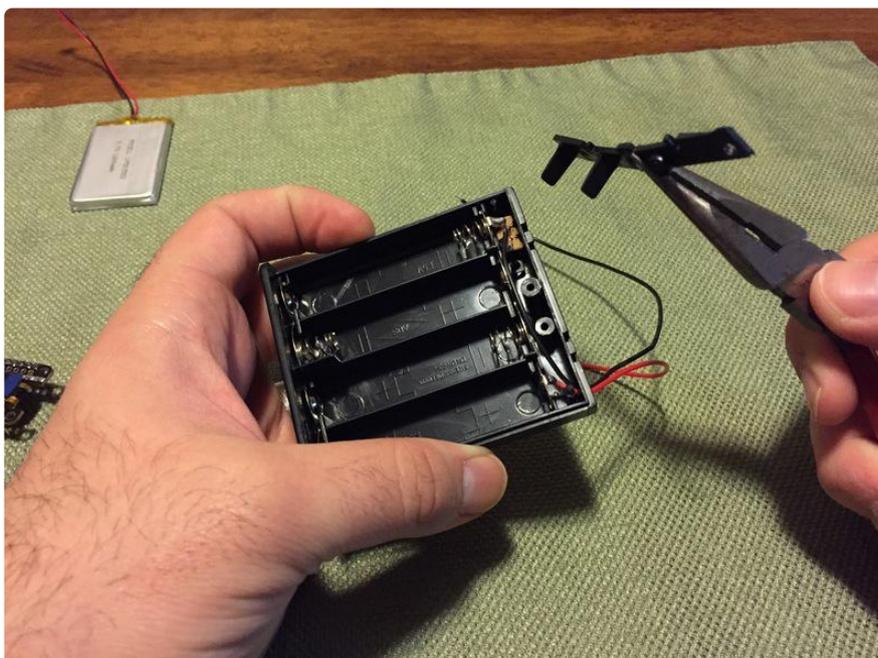
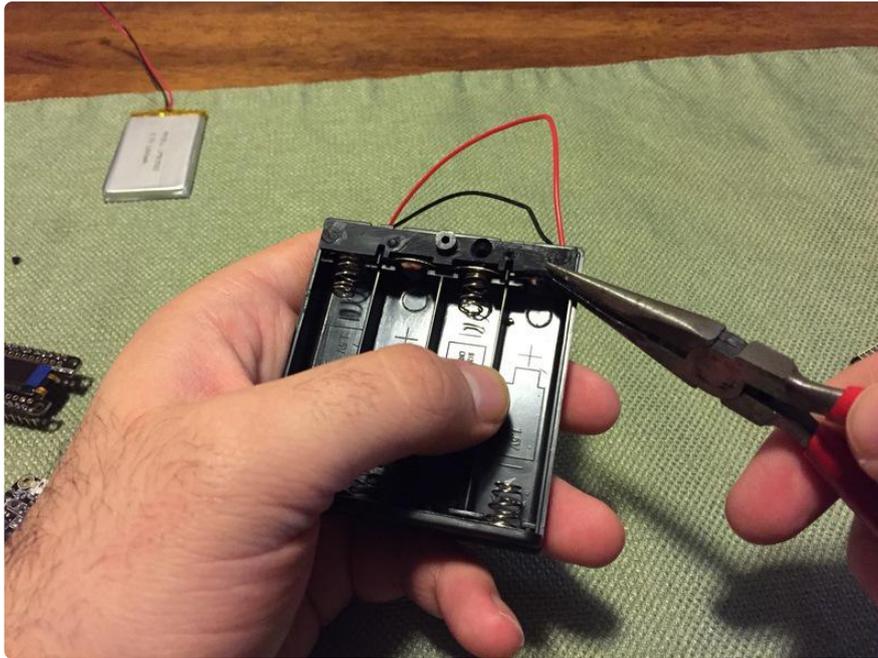
It works!

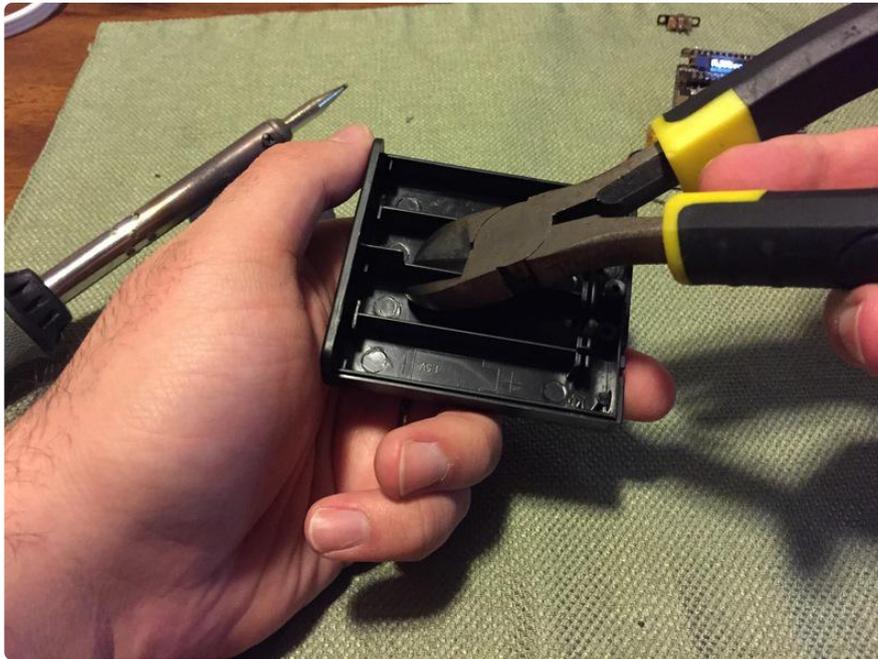


Preparing the Case

Since the case I chose to use is a AA battery holder it is compartmentalized inside. I needed to open it up and make room for the board, battery and wires. I used small pliers and cutters to break away the parts I didn't need.

I made sure to leave enough of the internal structure to ensure a snug fit so everything didn't bounce around when being moved.

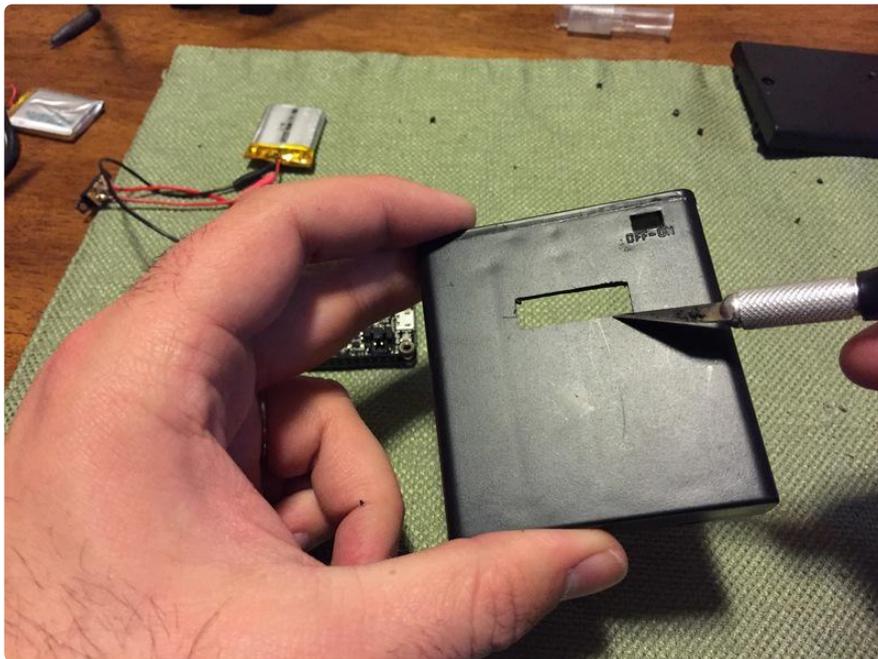




Make sure not to damage the area around the power switch. We'll use the switch later to turn the device on and off.

Once I made room for the board I had to cut an opening for the OLED screen. I started with a small hole, that I made with a small phillips head screw driver, around where I thought the center of the screen was and then slowly made it larger and rectangular to better fit the display.

I mostly used a hobby knife going slowly and checking very often to make sure everything would line up.

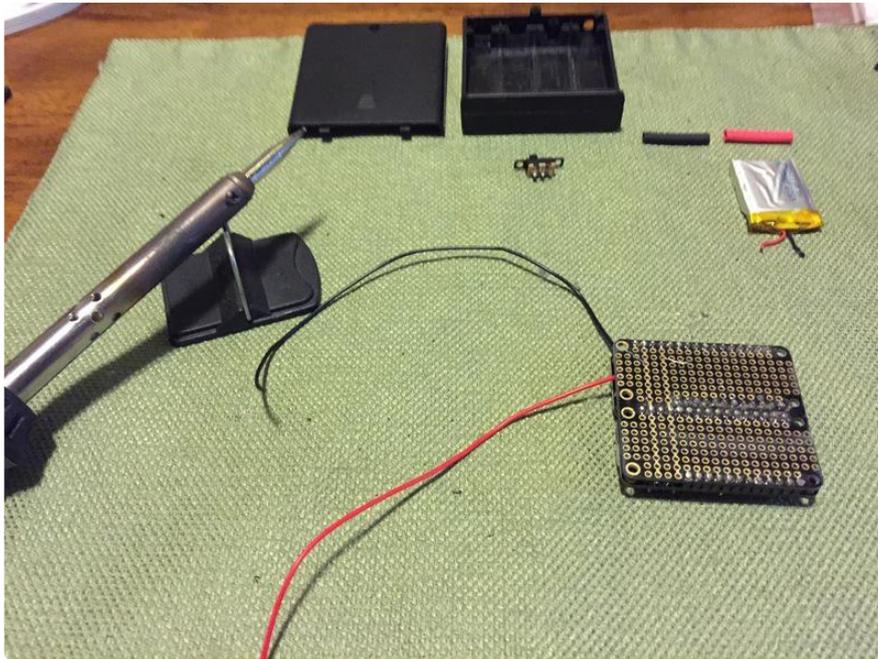


Once I got it to the size I wanted, I used the knife moving side to side to put a bevel on each of the four sides of the opening.

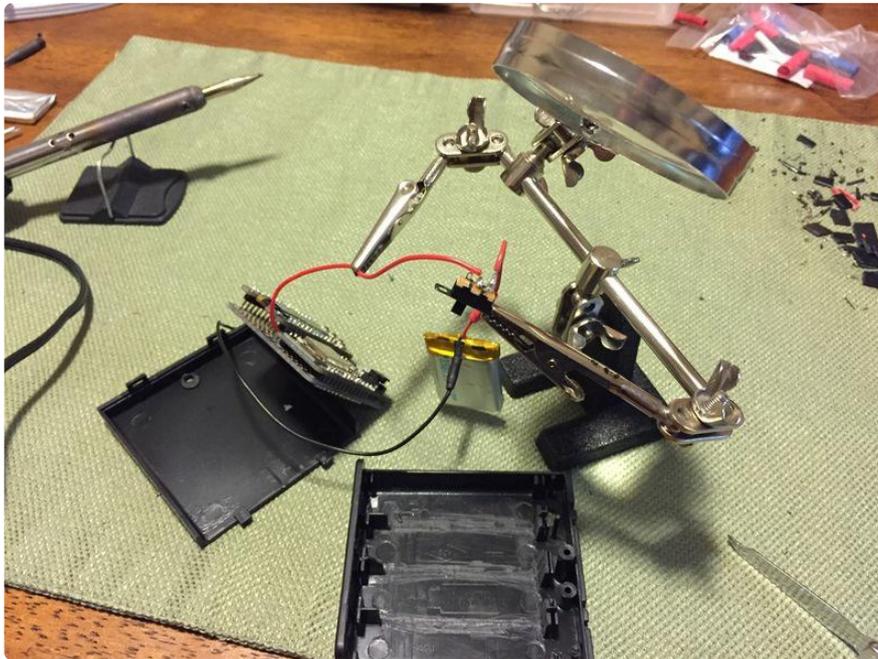
Connecting the Battery and Switch

One of the last few steps is to connect everything. The battery leads need to be soldered to the switch and to the board.

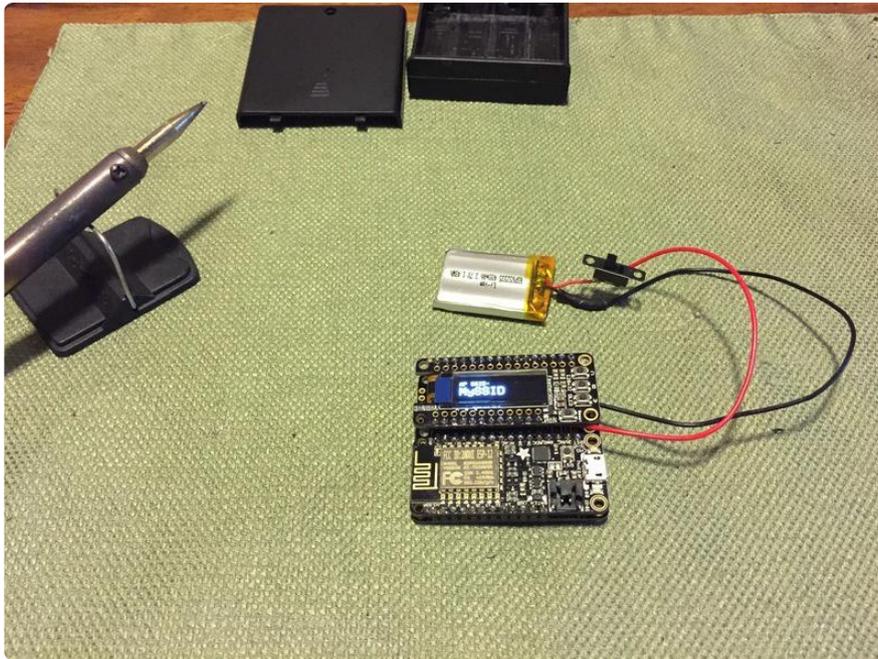
In the following image you can see the wires I was preparing to solder to the board to provide power. They are connected to the GND and BAT pins.



I soldered the GND wire directly to the battery and then soldered the red positive wire to the switch. I then also soldered the positive lead from the battery to the switch. I used shrink tubing around the solder points to reinforce them as well as to protect from accidental shorting of the battery.



When everything was done I tested it again and it worked.



The Code

Now that everything is wired up, it's time for the code.

```
#include <SPI.h>;
#include <Wire.h>;
#include <Adafruit_GFX.h>;
#include <Adafruit_SSD1306.h>;
#include <ESP8266WiFi.h>;
#include <ESP8266WebServer.h>;

extern "C" {
#include<user_interface.h>;
}

int connectedClients = 0;

long nextCheck = 0;    //Keeps track of the value of millis to only check for new
Clients every 5 seconds.

Adafruit_SSD1306 display = Adafruit_SSD1306();

/* configuration wifi */
const char *ssid = "MAC Finder";
const char *password = "";

#ifdef ESP8266
#define BUTTON_A 0
#define BUTTON_B 16
#define BUTTON_C 2
#define LED      0
#endif

#ifdef SSD1306_LCDHEIGHT != 32
#error("Height incorrect, please fix Adafruit_SSD1306.h!");
#endif
```

```

void setup() {
    delay(1000);

    //Initialize the display
    display.begin(SSD1306_SWITCHCAPVCC, 0x3C); // initialize with the I2C addr 0x3C
    (for the 128x32)
    display.setRotation(2);
    display.clearDisplay();

    display.setTextSize(1);
    display.setTextColor(WHITE);
    display.clearDisplay();

    initializeAP(); //Setup the Huzzah as an access point
}

void initializeAP() {
    WiFi.softAP(ssid, password);

    //I've noticed that most times the huzzah doesn't let clients connect
    //as soon as it starts up. Here I give it 5 seconds to "warm up"
    //While it does this it prints

    display.clearDisplay();
    display.display();
    display.setTextSize(1);
    display.setCursor(0,0);
    display.print("Initializing AP");

    //The two bytes that define the bitmap for the loading dots
    static const unsigned char PROGMEM waiting_dot[] =
    { B00011000,
      B00011000 };

    for (int i = 0; i < 10; i++) {

        display.drawBitmap(i * 10, 20, waiting_dot, 8, 2, WHITE);
        display.display();

        delay(1000);
    }

    display.clearDisplay();
    display.display();
    display.setCursor(0,0);

    displaySSID();
}

void loop() {
    if (millis() >= nextCheck) {
        client_status();
        nextCheck = millis() + 5000;
    }
}

void displaySSID() {
    display.setTextSize(1);
    display.clearDisplay();

```

```

display.setCursor(0,0);

display.print("AP SSID: ");

display.setTextSize(2);
display.setCursor(0, 10);
display.println(ssid);
display.display();
}

void client_status() {

  unsigned char number_client;
  struct station_info *stat_info;

  struct ip_addr *IPaddress;
  IPaddress address;
  int i=1;

  number_client= wifi_softap_get_station_num(); // Count of stations which are
connected to ESP8266 soft-AP
  stat_info = wifi_softap_get_station_info();

  if (number_client > connectedClients) {
    display.setTextSize(1);
    display.clearDisplay();
    display.setCursor(0, 0);
    display.setTextSize(1);
    display.print("Client Connected!");
    display.display();
    delay(2000);
  }

  if (number_client < connectedClients) {
    display.setTextSize(1);
    display.clearDisplay();
    display.setCursor(0, 0);
    display.setTextSize(1);
    display.print("Client Disconnected!");
    display.display();
    delay(2000);

    if (number_client == 0) {
      displaySSID();
    }
  }
}

connectedClients = number_client;

while (stat_info != NULL) {

  display.clearDisplay();

  display.setCursor(0, 0);
  display.setTextSize(1);
  display.print("Client MAC:");

  display.setCursor(0, 10);

  //Get the 6 octets of the client MAC address. Check to see if the value of each
octet is less than
//0xF. If it is we will print a leading 0 to avoid any confusion.
  if (stat_info->bssid[0] <= 0xF) {
    display.print("0");
  }
}

```

```

display.print(stat_info-&gt;bssid[0],HEX);
display.print(":");

if (stat_info-&gt;bssid[1] &lt;= 0xF) {
  display.print("0");
}

display.print(stat_info-&gt;bssid[1],HEX);
display.print(":");

if (stat_info-&gt;bssid[2] &lt;= 0xF) {
  display.print("0");
}

display.print(stat_info-&gt;bssid[2],HEX);
display.print(":");

if (stat_info-&gt;bssid[3] &lt;= 0xF) {
  display.print("0");
}

display.print(stat_info-&gt;bssid[3],HEX);
display.print(":");

if (stat_info-&gt;bssid[4] &lt;= 0xF) {
  display.print("0");
}

display.print(stat_info-&gt;bssid[4],HEX);
display.print(":");

if (stat_info-&gt;bssid[5] &lt;= 0xF) {
  display.print("0");
}

display.print(stat_info-&gt;bssid[5],HEX);

display.display();

stat_info = STAILQ_NEXT(stat_info, next);
i++;
}
}

```

Upload the code to the Feather Huzzah and test it out. Once you verify it works it's time to put everything together.

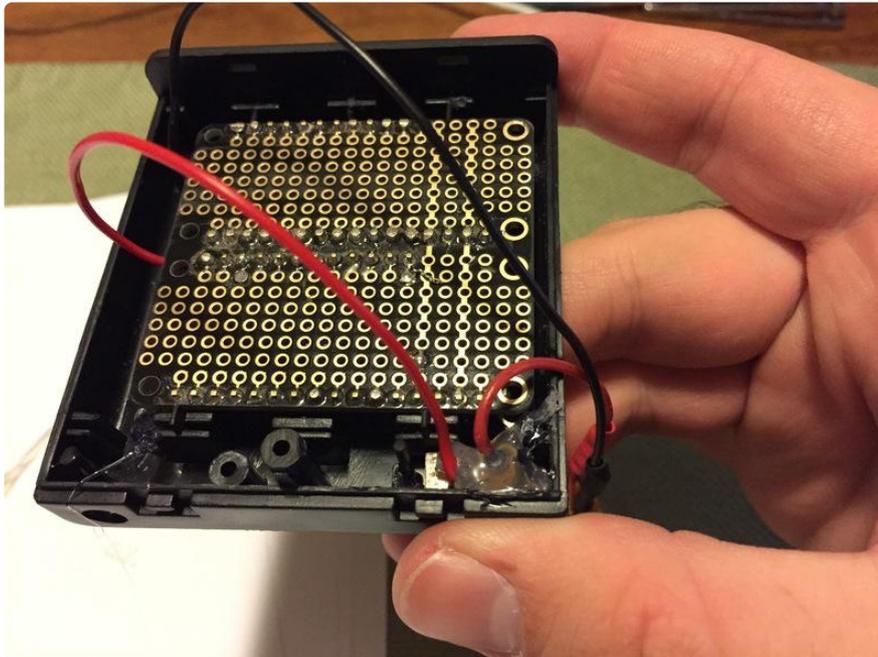
Putting it all Together

Now it's time to get everything to close up nicely. For this I fit the board in, saw where I needed more space, shaved some plastic off with the knife, sanded edges down as needed and rinsed and repeated until everything fit.

I actually had to sand down the solder joints on the bottom of the Doubler to get a few more precious millimeteres of space. You don't want to have to force everything in. Doing this means you run the risk of puncturing the lipo battery. Nobody wants that.

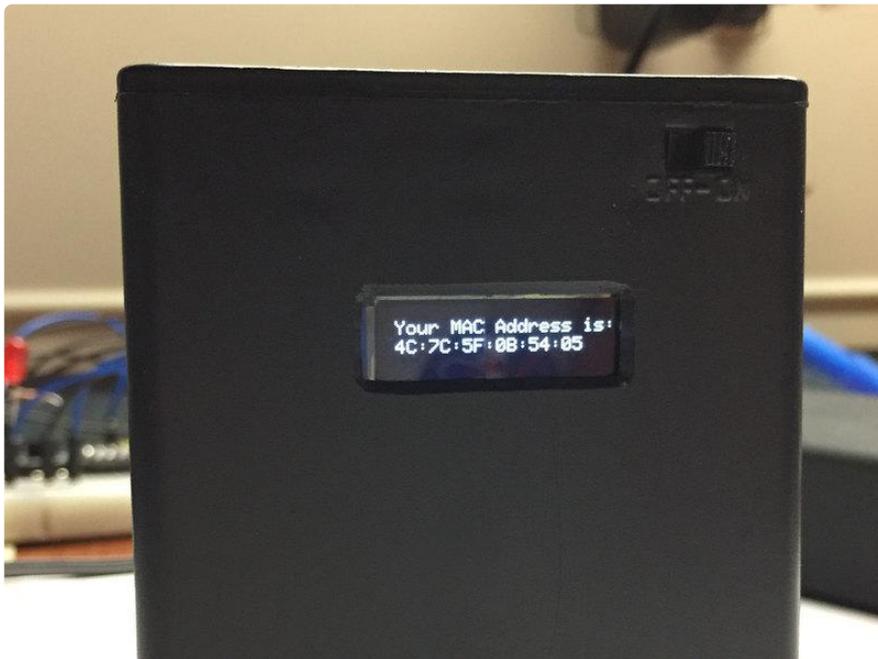
When everything was in I added some hot glue to the switch to keep it in place as I didn't put any on the board as I did not make an opening for a USB cable to charge the battery.

In my case this is fine as I will be removing the board and using it for many different purposes. If, however, you'd like to make one you would do so the same as was done for the OLED screen.



After some cable management, i.e. shoving the extra wires anywhere they would fit, I slid the back cover on and it was done!





Now nothing will stop you from getting the MAC address of any wireless device you encounter.