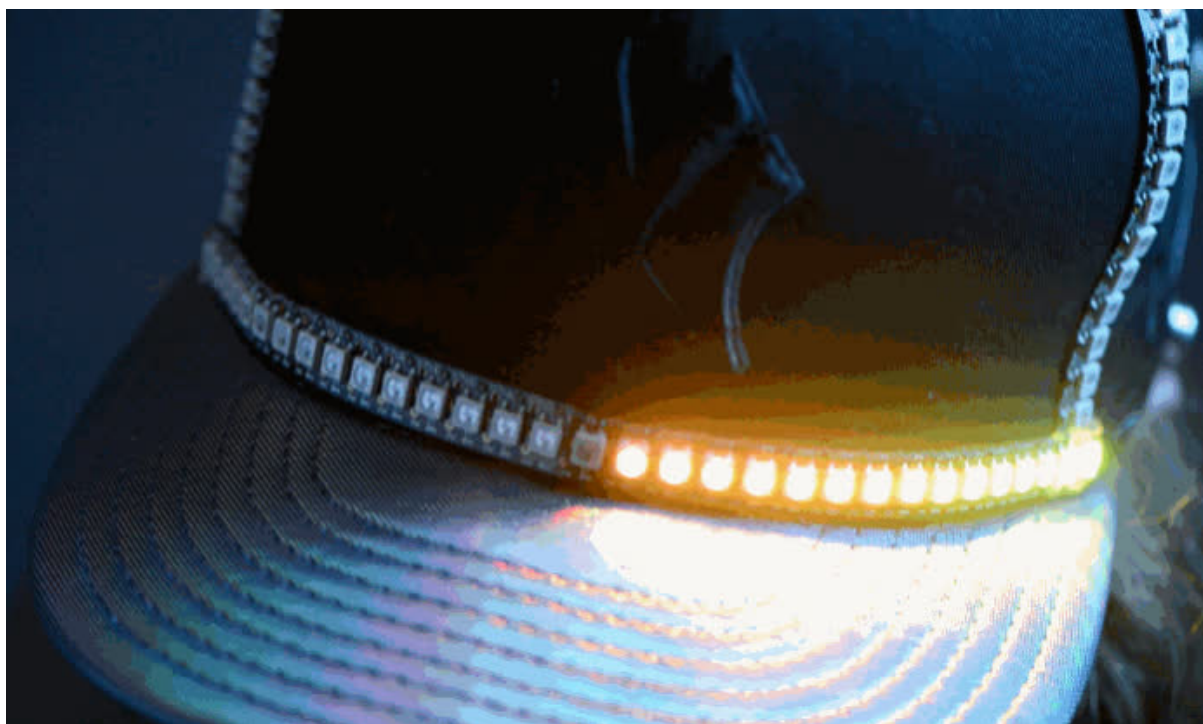




VU Meter Baseball Hat

Created by Becky Stern



<https://learn.adafruit.com/vu-meter-baseball-hat>

Last updated on 2024-06-03 01:31:41 PM EDT

Table of Contents

Overview	3
Circuit Diagram	5
Add LEDs to hat	6
Add Mic & FLORA	12
Wear it!	22

Overview

Add sound-reactive NeoPixels to your baseball cap! Use our 144-density NeoPixel strip as a wraparound VU meter with a small microphone at the side. The FLORA microcontroller takes input from the mic and sends the LEDs flashing according to the volume in the room!

This intermediate level wearables project requires some precision soldering and sewing. We recommend checking out the following guides before you begin:

- [Getting Started with FLORA \(https://adafru.it/dwi\)](https://adafru.it/dwi)
- [Adafruit Guide To Excellent Soldering \(https://adafru.it/drl\)](https://adafru.it/drl)
- [Adafruit NeoPixel Überguide \(https://adafru.it/dhw\)](https://adafru.it/dhw)

You'll have installed the Adafruit Arduino software by now, OR you can program your FLORA directly from your browser using [Codebender \(https://adafru.it/dwj\)](https://adafru.it/dwj). Fancy! For this project you'll need:

- [FLORA main board \(http://adafru.it/659\)](http://adafru.it/659)
- [Electret Microphone Amplifier \(http://adafru.it/1063\)](http://adafru.it/1063)
- [144-density NeoPixel strip \(http://adafru.it/1506\)](http://adafru.it/1506)
- [500mAh lipoly battery \(http://adafru.it/1578\)](http://adafru.it/1578) and [charger \(http://adafru.it/1304\)](http://adafru.it/1304)
- [alligator clips \(http://adafru.it/1592\)](http://adafru.it/1592)
- [soldering iron \(http://adafru.it/180\)](http://adafru.it/180) and [solder \(http://adafru.it/145\)](http://adafru.it/145)
- [wire strippers \(http://adafru.it/527\)](http://adafru.it/527), [pliers \(http://adafru.it/1368\)](http://adafru.it/1368), and [flush diagonal cutters \(http://adafru.it/152\)](http://adafru.it/152)
- (optional but very helpful) [third hand tool \(http://adafru.it/291\)](http://adafru.it/291)
- stranded wire (ribbon cable works great too)
- [sewing needle \(http://adafru.it/615\)](http://adafru.it/615) and thread
- scissors
- sewing pins or masking tape
- sturdy tape like gaff or packing
- baseball cap (we recommend getting one size up to save room for the battery)



The hat displays animations based on the audio volume from the microphone, just like the [Ampli-Tie \(https://adafru.it/doN\)](https://adafru.it/doN). If you're looking for a project that reacts to sound but doesn't require (much) soldering, check it out!

76 eye-blasting pixels adorn the front of the baseball cap design. The small segments of flex strip connected with stranded ribbon cable and the FLORA microcontroller can hang out on the outside or inside of the cap.

The circuit is powered by a 500mAh lipoly battery which is stored in the hat's band at the back of the head.





Circuit Diagram

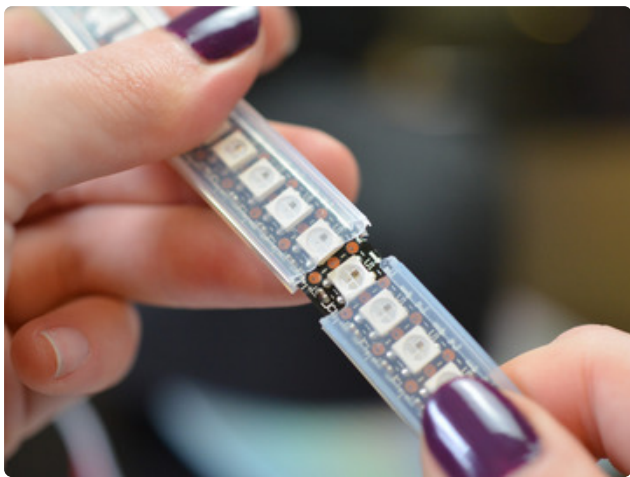


The soldered circuit connections are as follows (click to enlarge image):

NeoPixel strip + -> FLORA VBATT
NeoPixel strip - -> FLORA GND
NeoPixel strip center data pin -> FLORA D6

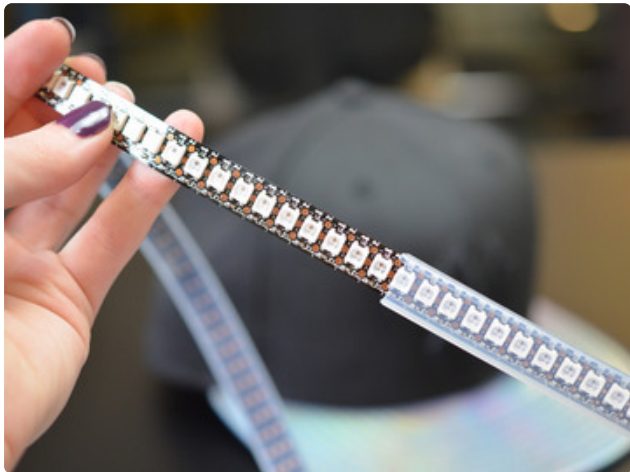
Electret mic VCC -> FLORA 3.3V
Electret mic GND -> FLORA GND
Electret mic OUT -> FLORA D9

Add LEDs to hat

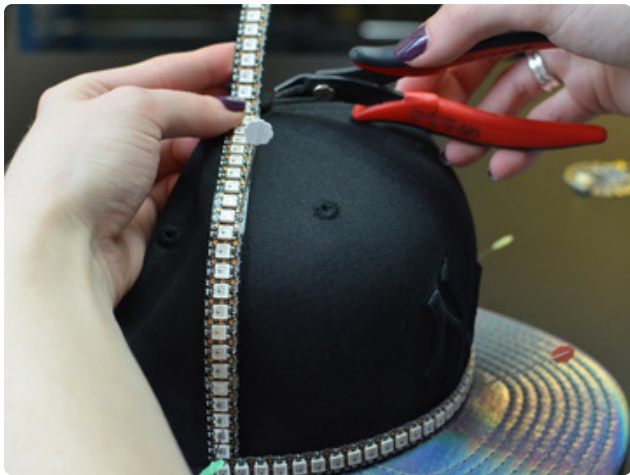
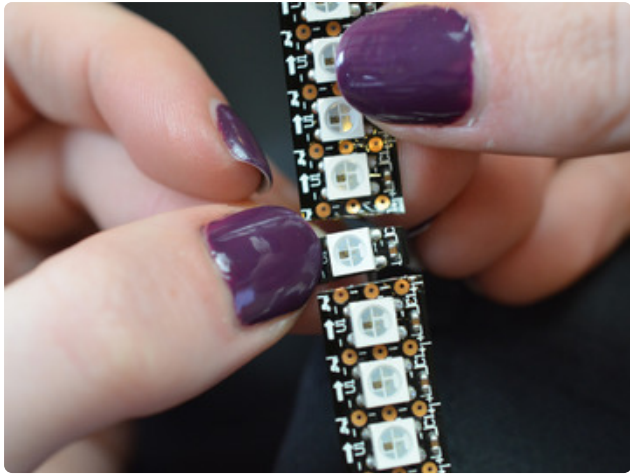


Remove the silicone sheathing from your NeoPixel strip. It's easiest to cut into it with scissors or diagonal cutters and then slide small sections off at a time.

Hold the strip up to your hat's brim to measure the length for the first piece. The design is up to you!



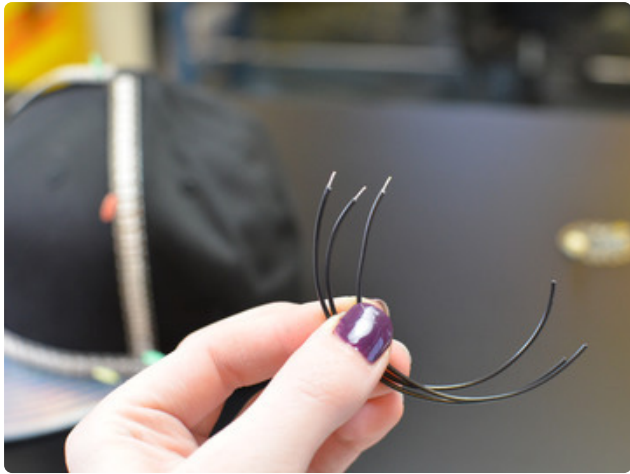




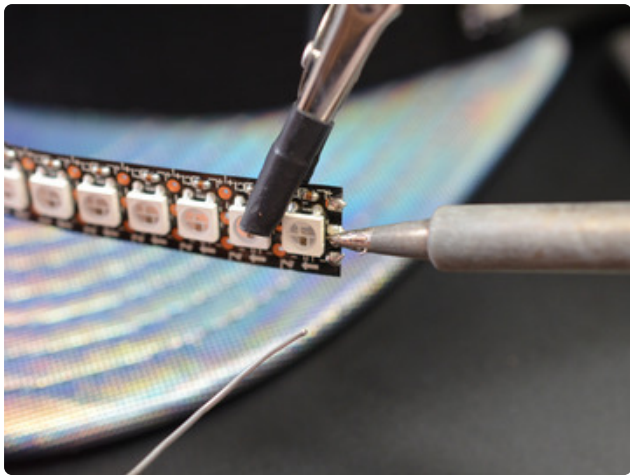
When cutting segments of strip to compose your design, you'll always have a "sacrificed" pixel. Cut close to the waste pixel to preserve the size of the solder pads on the neighboring pixels.

Cut three pieces to the desired length, using pins or masking tape to mock up your design on the hat.



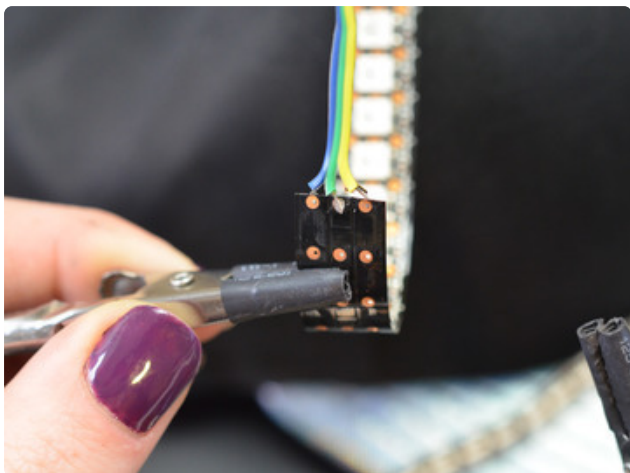
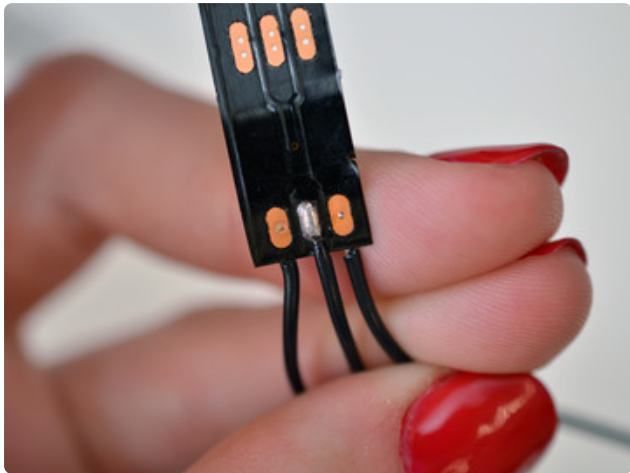


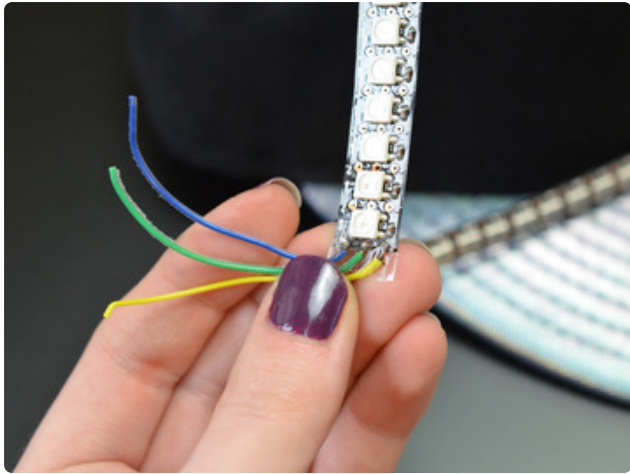
Next prep three pieces of stranded wire (or one piece of three-channel ribbon wire) by stripping and tinning the ends.



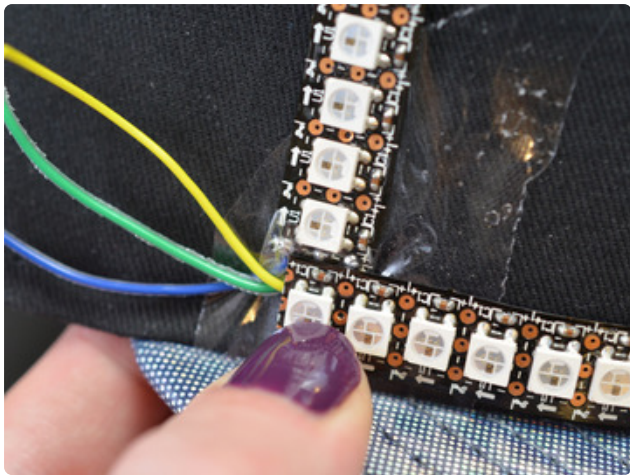
To solder wires to NeoPixel strip, identify the input end (arrows facing away). First solder wires to + and -, then solder the signal wire on the opposite side of the flex PCB. This way your solder joints will be less likely to crowd each other.

Repeat for the remaining pieces of NeoPixel strip in your design, soldering three wires to each input side.

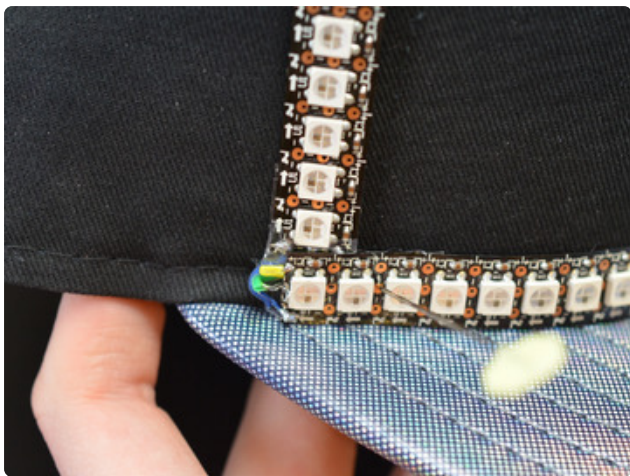




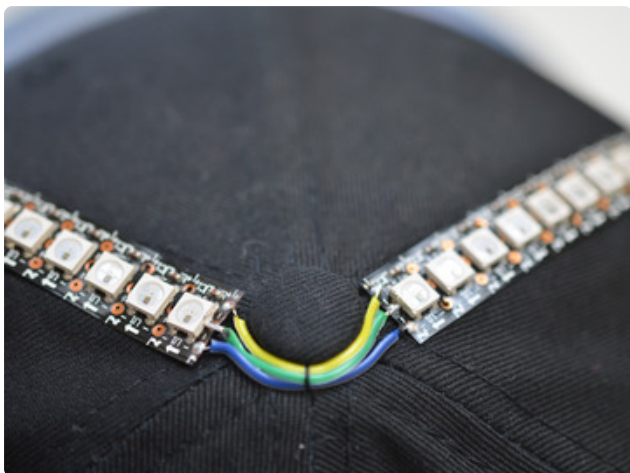
For places where the two pieces of NeoPixel strip will come into close contact with one another, wrap the soldered end in tape so the contacts don't short to the other segment. Tape/pin in place and carefully solder to the output end of the previous strip.



This is a tricky technique that can take practice to achieve a tidy result. You may want to test it out on scrap pieces of strip before soldering the pieces meant for your hat!

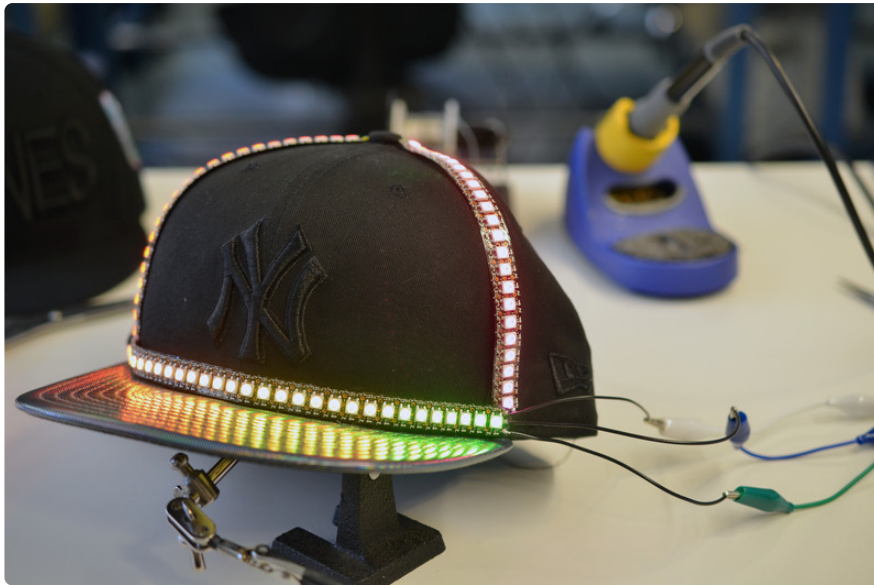


Likewise make the connection at the top of the cap, and stitch down any loose wires with a bit of plain thread.





Stitch your NeoPixel strip to your hat. You can use the holes provided or just stitch around the whole flex PCB. Near the brim of the hat, be sure to sew underneath the hem band. You only need to make a stitch every few pixels or so to keep the strip firmly following the contour of the hat.



Use alligator clips to hook up the input end of the strip to FLORA. Program with the NeoPixel sample code to be sure your solder connections are solid and there are no shorts. All your pixels should light up! If they don't, go back and test your circuit with a multimeter and resolder/repair/reinsulate any connections that are causing trouble.

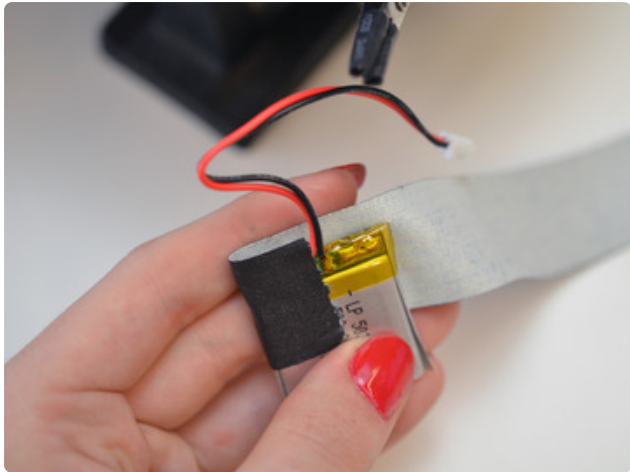
Add Mic & FLORA



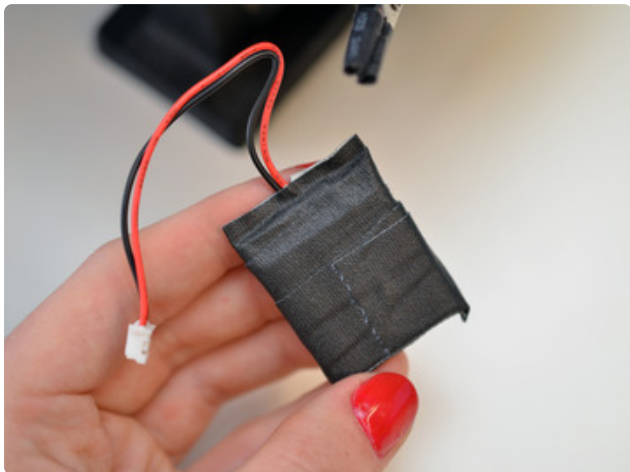
Once your NeoPixels are hooked up nicely, it's time to add the rest of the circuit. If your FLORA is going on the outside of your hat, solder those wires up directly.

If you want to hide FLORA on the inside of your hat, cut a small hole near the NeoPixel strip and slide the wires through before soldering them to FLORA.

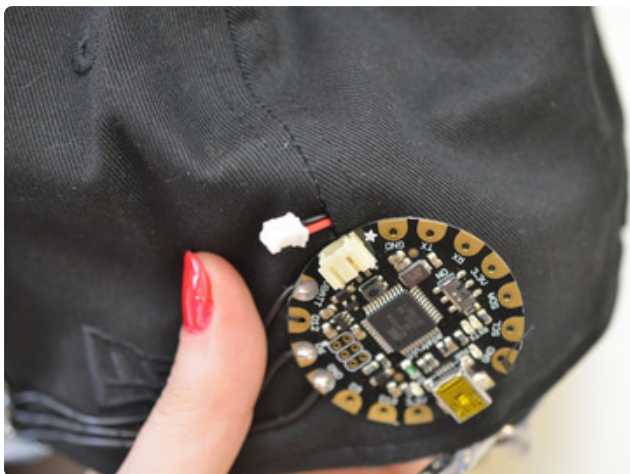
Refer to the [circuit diagram \(https://adafruit.it/dxT\)](https://adafruit.it/dxT) to be sure you're wiring everything up correctly!



Wrap your battery in gaffers tape to provide a bit of strain relief to the wires and protect it from light wear.



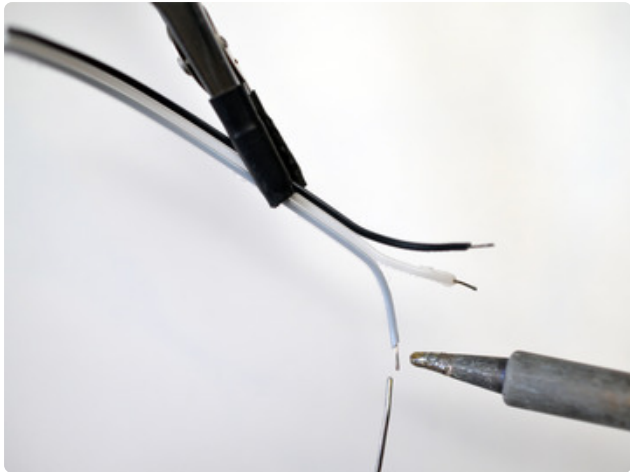
You can hide this battery in the band of the hat, and run the wire to the outside (if that's where your FLORA is), or along the inside of the band to FLORA.



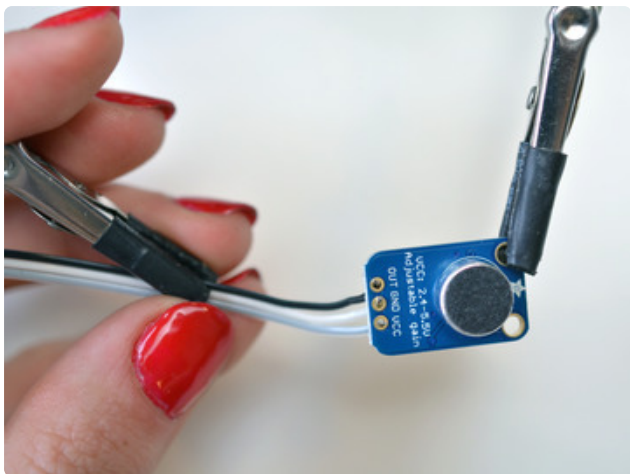
Test that your NeoPixel sample code is still working on battery power and now that you've soldered FLORA.



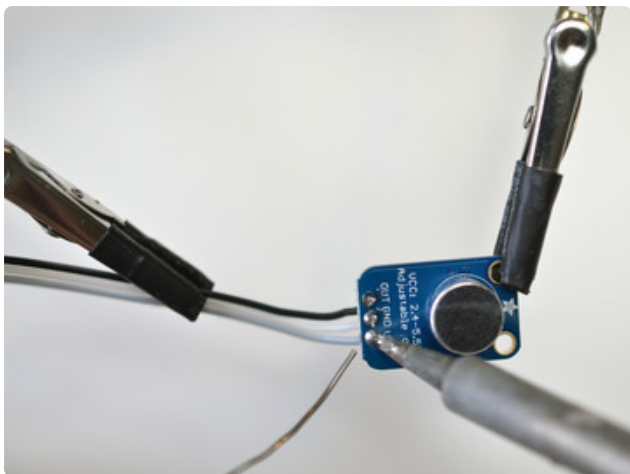


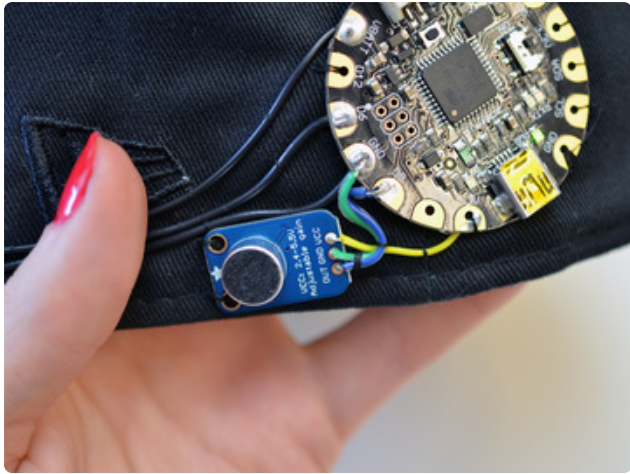


To add the microphone, first prep a three-channel ribbon wire (or three pieces of plain stranded wire) by stripping and tinning the ends.



Set up the wire and the mic in a third hand tool to make it easier to solder the wires to the holes on the breakout board. Trim the excess wire short with flush diagonal cutters.





On the outside, it's easy to just solder the wires according to the circuit diagram and stitch the mic down.



If you want to hide the mic inside the hat, mark where you'd like it to go with a pencil.



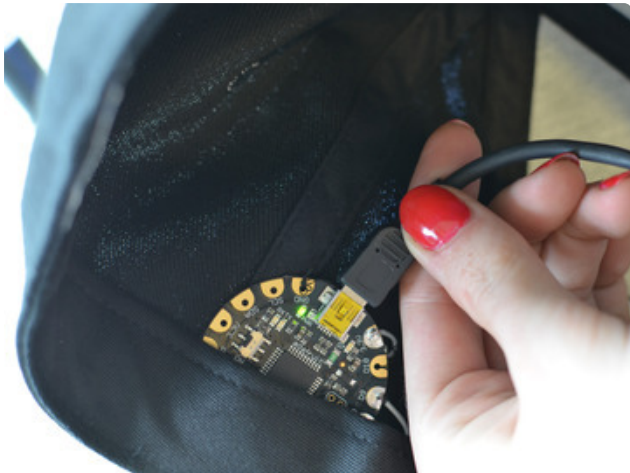
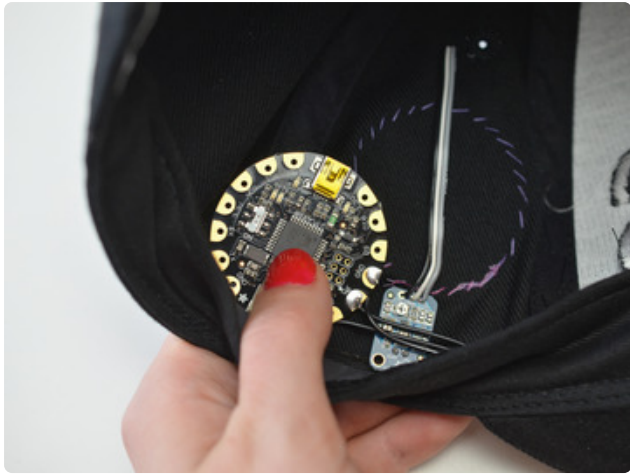
Carefully trim a mic-sized circle at the mark using very sharp scissors.





Optional step! If your hat is mostly synthetic (acrylic/polyester), you can singe the edge of the cut hole with a lighter to make it tidier. This hat is part wool, so doing this smells like burned hair. Hey, it's optional!





Arrange FLORA and the mic board inside the hat as you'd like them, then solder the connections to the mic according to the circuit diagram.

Then stitch the FLORA and mic board to the inside of the hat.

Plug FLORA in via USB (make sure the battery switch is OFF) and load with the following sound-sensitive code (very barely modified from the [Ampli-Tie sketch \(https://adafruit.it/dxU\)](https://adafruit.it/dxU)):

```

/*
LED VU meter for Arduino and Adafruit NeoPixel LEDs.

Hardware requirements:
- Most Arduino or Arduino-compatible boards (ATmega 328P or better).
- Adafruit Electret Microphone Amplifier (ID: 1063)
- Adafruit Flora RGB Smart Pixels (ID: 1260)
  OR
- Adafruit NeoPixel Digital LED strip (ID: 1138)
- Optional: battery for portable use (else power through USB or adapter)
Software requirements:
- Adafruit NeoPixel library

Connections:
- 3.3V to mic amp +
- GND to mic amp -
- Analog pin to microphone output (configurable below)
- Digital pin to LED data input (configurable below)
See notes in setup() regarding 5V vs. 3.3V boards - there may be an
extra connection to make and one line of code to enable or disable.

Written by Adafruit Industries. Distributed under the BSD license.
This paragraph must be included in any redistribution.
*/

#include <Adafruit_NeoPixel.h>;

#define N_PIXELS 76 // Number of pixels in strand
#define MIC_PIN A9 // Microphone is attached to this analog pin
#define LED_PIN 6 // NeoPixel LED strand is connected to this pin
#define DC_OFFSET 0 // DC offset in mic signal - if unsure, leave 0
#define NOISE 10 // Noise/hum/interference in mic signal
#define SAMPLES 60 // Length of buffer for dynamic level adjustment
#define TOP (N_PIXELS + 2) // Allow dot to go slightly off scale
#define PEAK_FALL 4 // Rate of peak falling dot

byte
  peak = 0, // Used for falling dot
  dotCount = 0, // Frame counter for delaying dot-falling speed
  volCount = 0; // Frame counter for storing past volume data
int
  vol[SAMPLES], // Collection of prior volume samples
  lvl = 10, // Current "dampened" audio level
  minLvlAvg = 0, // For dynamic adjustment of graph low & high
  maxLvlAvg = 512;
Adafruit_NeoPixel
  strip = Adafruit_NeoPixel(N_PIXELS, LED_PIN, NEO_GRB + NEO_KHZ800);

void setup() {

  // This is only needed on 5V Arduinos (Uno, Leonardo, etc.).
  // Connect 3.3V to mic AND TO AREF ON ARDUINO and enable this
  // line. Audio samples are 'cleaner' at 3.3V.
  // COMMENT OUT THIS LINE FOR 3.3V ARDUINOS (FLORA, ETC.):
  // analogReference(EXTERNAL);

  memset(vol, 0, sizeof(vol));
  strip.begin();
}

void loop() {
  uint8_t i;
  uint16_t minLvl, maxLvl;
  int n, height;

  n = analogRead(MIC_PIN); // Raw reading from mic

```

```

n = abs(n - 512 - DC_OFFSET); // Center on zero
n = (n <= NOISE) ? 0 : (n - NOISE); // Remove noise/hum
lvl = ((lvl * 7) + n) >> 3; // "Dampened" reading (else looks twitchy)

// Calculate bar height based on dynamic min/max levels (fixed point):
height = TOP * (lvl - minLvlAvg) / (long)(maxLvlAvg - minLvlAvg);

if(height < 0L) height = 0; // Clip output
else if(height > TOP) height = TOP;
if(height > peak) peak = height; // Keep 'peak' dot at top

// Color pixels based on rainbow gradient
for(i=0; i<N_PIXELS; i++) {
    if(i >= height) strip.setPixelColor(i, 0, 0, 0);
    else strip.setPixelColor(i, Wheel(map(i, 0, strip.numPixels()-1, 30, 150)));
}

// Draw peak dot
if(peak > 0 && peak <= N_PIXELS-1)
strip.setPixelColor(peak, Wheel(map(peak, 0, strip.numPixels()-1, 30, 150)));

strip.show(); // Update strip

// Every few frames, make the peak pixel drop by 1:
if(++dotCount >= PEAK_FALL) { //fall rate
    if(peak > 0) peak--;
    dotCount = 0;
}

vol[volCount] = n; // Save sample for dynamic leveling
if(++volCount >= SAMPLES) volCount = 0; // Advance/rollover sample counter

// Get volume range of prior frames
minLvl = maxLvl = vol[0];
for(i=1; i<SAMPLES; i++) {
    if(vol[i] < minLvl) minLvl = vol[i];
    else if(vol[i] > maxLvl) maxLvl = vol[i];
}
// minLvl and maxLvl indicate the volume range over prior frames, used
// for vertically scaling the output graph (so it looks interesting
// regardless of volume level). If they're too close together though
// (e.g. at very low volume levels) the graph becomes super coarse
// and 'jumpy'...so keep some minimum distance between them (this
// also lets the graph go to zero when no sound is playing):
if((maxLvl - minLvl) < TOP) maxLvl = minLvl + TOP;
minLvlAvg = (minLvlAvg * 63 + minLvl) >> 6; // Dampen min/max levels
maxLvlAvg = (maxLvlAvg * 63 + maxLvl) >> 6; // (fake rolling average)
}

// Input a value 0 to 255 to get a color value.
// The colors are a transition r - g - b - back to r.
uint32_t Wheel(byte WheelPos) {
    if(WheelPos < 85) {
        return strip.Color(WheelPos * 3, 255 - WheelPos * 3, 0);
    } else if(WheelPos < 170) {
        WheelPos -= 85;
        return strip.Color(255 - WheelPos * 3, 0, WheelPos * 3);
    } else {
        WheelPos -= 170;
        return strip.Color(0, WheelPos * 3, 255 - WheelPos * 3);
    }
}

```

```
}  
}
```

Wear it!



Enjoy your new beat-flashing cap! Wear it to parties, bars, birthday parties (yours?), bar mitzvahs and any other occasion where you'd like to be the center of attention.

The hat is not water resistant, however. Turn it off if it starts raining, and let it dry completely before turning it back on. If you place the FLORA inside the hat, you can [perform some measures to ruggedize the outside](https://adafru.it/b120249) (<https://adafru.it/b120249>) for drizzle-resistance.

Remove the battery to spot clean as needed.

Don't wear this hat with wet hair or leave it on while exercising/sweating excessively.

You're not in danger of being shocked but you could damage your new creation if it gets wet while it's powered on!

