Sparkle Skirt

Created by Becky Stern

https://learn.adafruit.com/sparkle-skirt

Last updated on 2024-06-03 01:18:52 PM EDT

Table of Contents

Overview

Make clothing that lights up when you move! This is a sewing (no soldering) Flora project using 12 color-changing NeoPixels and the Flora accelerometer/compass module. Spikes in the sensor readings cause the pixels to flash, and sparkle. The pixels are sewn to the skirt's lining and are diffused by the outer layer of fabric, whether it be a sheer overlay or a cutout lace design.

If a skirt doesn't appeal to you, use this circuit and code on a hat, a belt buckle, and even a blinky dog collar.

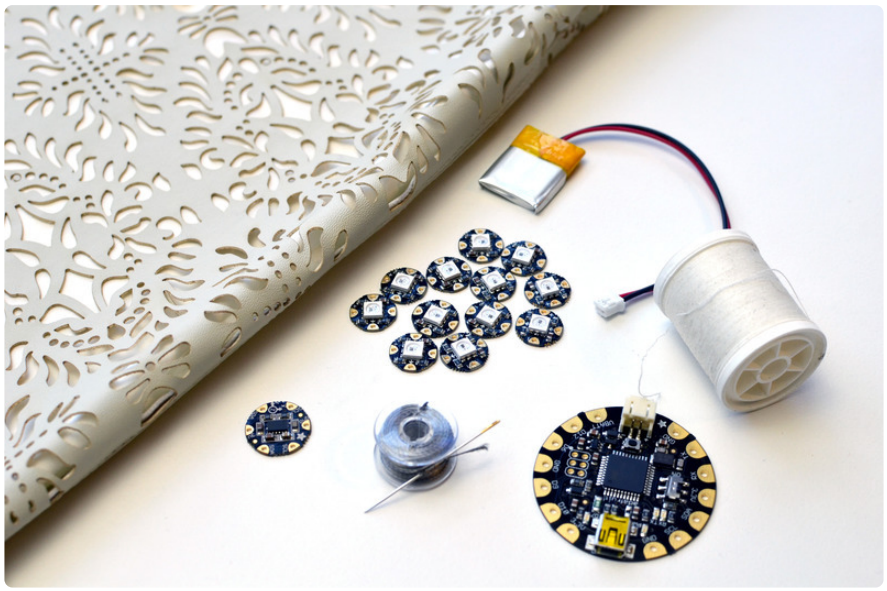
Prerequisite guides:

- Getting Started with FLORA (https://adafru.it/aSZ)
- Conductive thread (https://adafru.it/c2A)

- FLORA Accelerometer (https://adafru.it/aYS)
- FLORA RGB NeoPixels (https://adafru.it/aRT)

Model photos by John de Cristofaro, construction assistance and modeling by Risa Rose. Hat tip to Bona Kim (https://adafru.it/19nC).

Tools & Supplies



Bill of materials:

- FLORA main board (http://adafru.it/659)
- FLORA accelerometer + compass module (http://adafru.it/1247)
- 12 FLORA RGB NeoPixels (http://adafru.it/1260)
- 150mAh LiPo battery (http://adafru.it/1317) or 3xAAA battery pack (http://adafru.it/727)
- Conductive thread (http://adafru.it/641)
- Standard thread
- Lined skirt with either a sheer overlay (https://adafru.it/c9B) or cutout top layer (https://adafru.it/c9C)

Sharp scissors are a must!

You'll use a needle (http://adafru.it/615) and to stitch up the circuit.

Conductive thread (https://adafru.it/aVx) will carry signal and current to the Flora NeoPixels.

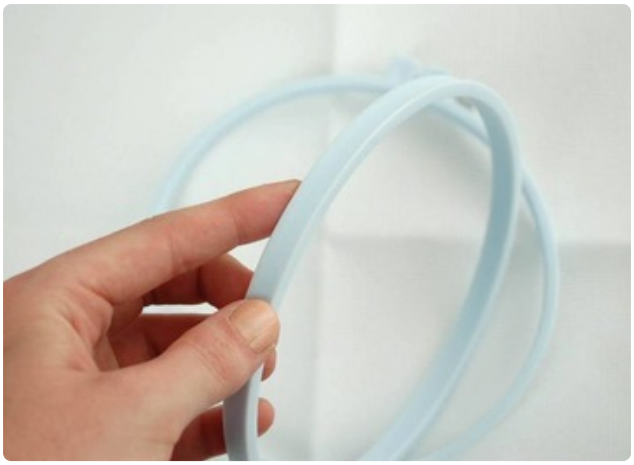
You will need a good quality basic multimeter that can measure voltage and continuity.

Click here to buy a basic multimeter. (http://adafru.it/71)

Click here to buy a top of the line multimeter. (http://adafru.it/308)

Click here to buy a pocket multimeter. (http://adafru.it/850)

Don't forget to learn how to use your multimeter too! (https://adafru.it/aOy)



We recommend beginners use an embroidery hoop to stabilize the circuit-supporting fabric during sewing. It makes things a lot easier and results in a better-looking finished project. For more information on using an embroidery hoop correctly, check out our Cross Stitch tutorial (https://adafru.it/aUT).

Clear nail polish or fray check to seal knots

To mark out your circuit use a water-soluble embroidery marker or tailor's chalk.

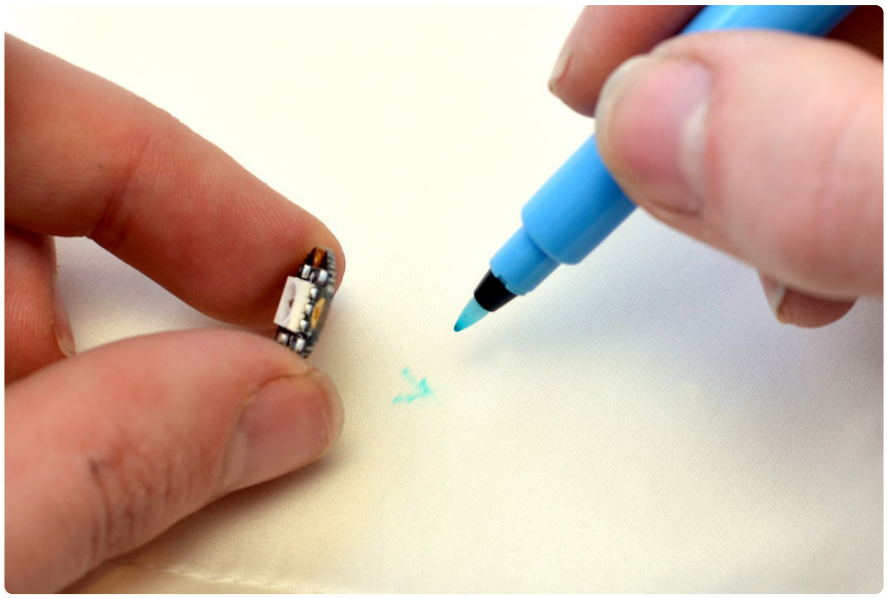
Layout & Circuit Diagram

Select a skirt and start planning where your pixels will go. Above is a diagram showing how to connect all the parts of this circuit (click to enlarge).

Since the conductive thread has nontrivial resistance, pay close attention to the extra connecting power and ground lines on either side of the skirt-- they will help your colored lights shine to their full potential.

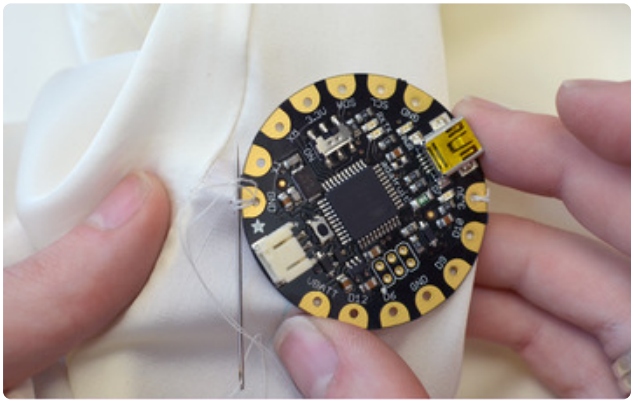
Lift the outside layer of the skirt away from the lining. If they are attached with small threads, snip them.

Place your components on the front of the skirt lining according to the diagram. We'll do the back pixels separately, after all six of the front pixels are working.

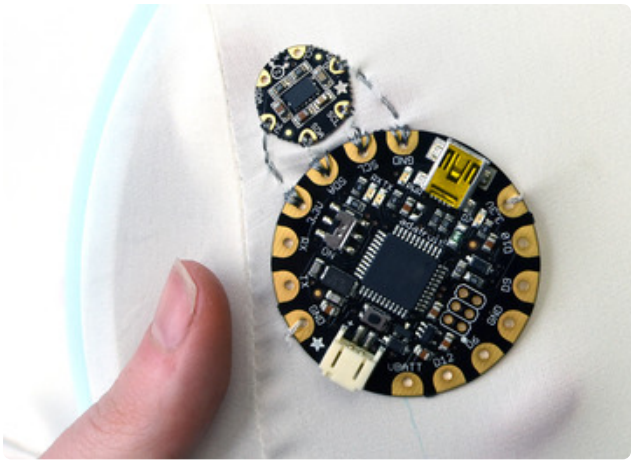


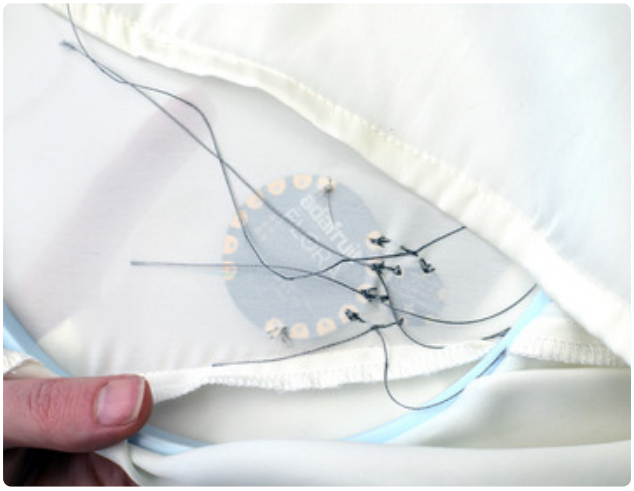
Lift each component and make a mark with a water-soluble marking pen (or just use a very fine tip permanent marker to make a tiny dot).

Sew Circuit



Begin by tacking down the FLORA main board to the skirt lining with regular thread, using two unused pads. Set up the fabric in a large embroidery hoop.





Stitch the four connections between the Flora main board and the accelerometer board. Check out the Flora Accelerometer Guide (https://adafru.it/aYS) for more info.

Then begin stitching to connect your pixels data pads (marked with inward facing arrows and outward facing arrows) from FLORA pad D6.

Read our Conductive Thread (https://adafru.it/aVx) guide for more tips on working with this odd steel fiber!

Knot, seal and clip the ends of the threads, being careful to avoid unintended connections (aka shorts).

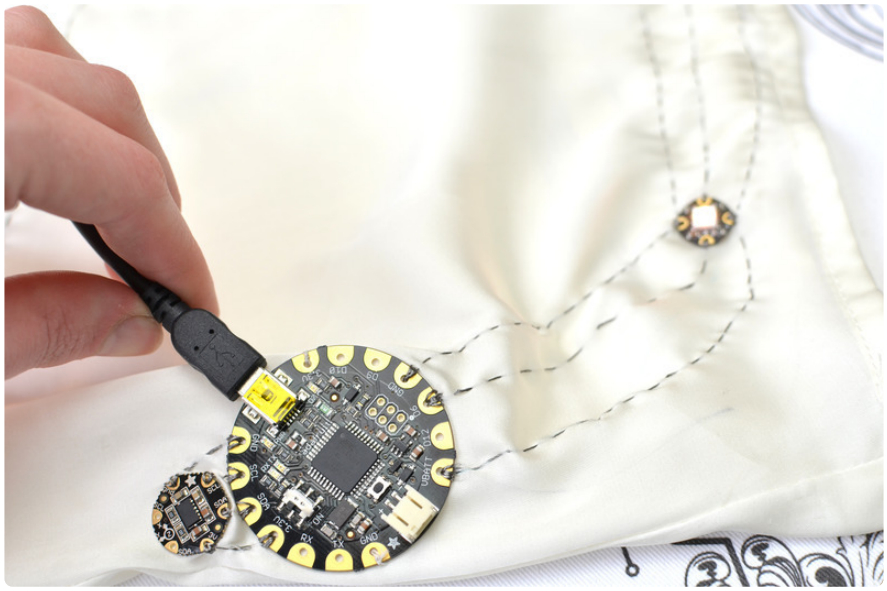
Using one extra long piece of conductive thread, stitch the power bus for the six pixels on the front of the skirt. Repeat with another for the ground line.

We'll test these pixels first before connecting more on the back!

Don't power up your skirt until you've checked for shorts! Double check your tails are clipped and your skirt is laying flat on a nonconductive surface. Never stitch

pixels with the power connected, and never connect power with the skirt bunched up in your lap!

Code & Battery

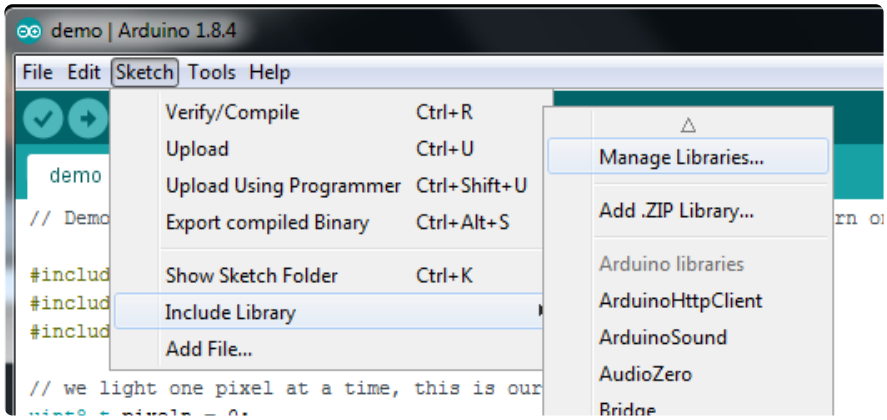


Double check that all your conductive thread tails have been trimmed and your skirt is laying flat on a nonconductive surface, and then connect your FLORA main board to your computer with a USB cable.

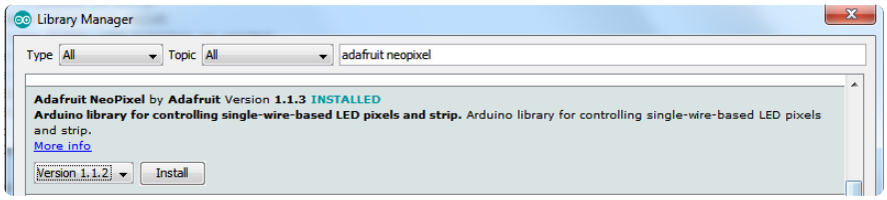
Open the Adafruit Arduino IDE, which you can download from the Getting Started with FLORA guide (https://adafru.it/aVE).

You will need the NeoPixel library (https://adafru.it/aZU) for this project-- if you've never installed an Arduino library before, read our guide on the topic (https://adafru.it/aYM)!

Open up the Library manager

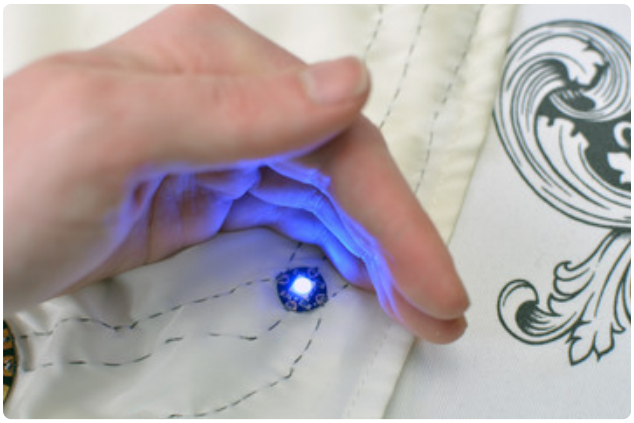


Find the **Adafruit NeoPixel** library and install it...



First test out all six pixels by uploading the sketch in File >> Examples >> Adafruit_NeoPixel >> strandtest.

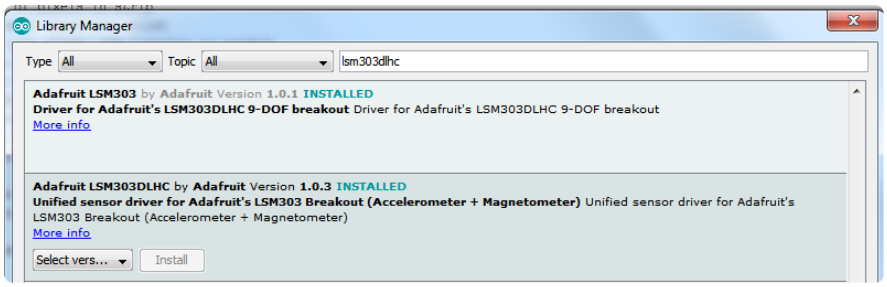
If all of your pixels light up and change color, you're good to continue to the back! Disconnect the USB cable before stitching more pixels on the back. For more tips on stitching up a line of NeoPixels, read the corresponding page in our LED Ampli-Tie guide (https://adafru.it/c9D).



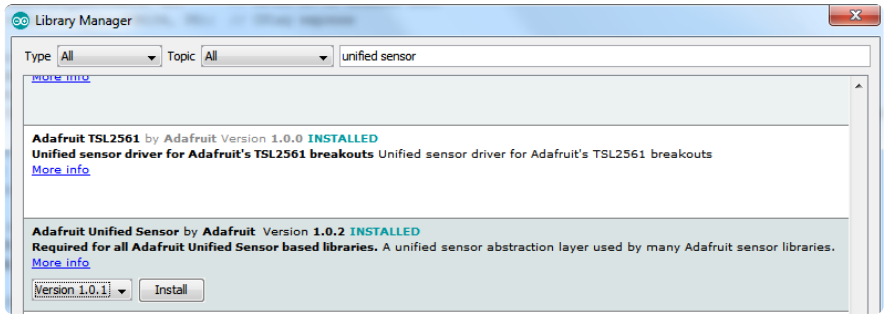
When you have pixels on the front and back of the skirt, place a piece of scrap fabric in between them for testing (so they don't short each other out), or put the skirt on a dress form. When you're wearing the skirt, your body will keep the front and back from touching.

If your skirt has lots of folds, you may have trouble with your pixel connections shorting against each other. You can insulate them under a piece of ribbon, wear a slip under the skirt, and also add some wobble to your pixel line so when the skirt folds, traces aren't up against each other.

Next test out the accelerometer, install the **Adafruit LSM303DLHC library** using the library manager. Make sure you grab the one with **DLHC** in the name



Also, install the **Adafruit Unified Sensor** library



Now test by uploading the sketch in File >> Examples >> Adafruit_LSM303 >> Accel_Sensor.

Open up the serial monitor and watch for changing motion values. For more information, check out our FLORA Accelerometer guide (https://adafru.it/aYS).

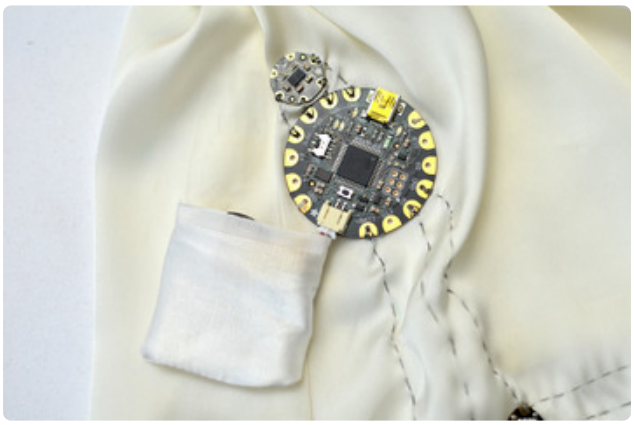
Click the Download File button below to get the code.

The colors can be specified in the `myFavoriteColors` array, and the sensitivity to motion can be defined with `MOVE_THRESHOLD`.

```
// SPDX-FileCopyrightText: 2017 Limor Fried for Adafruit Industries
//
// SPDX-License-Identifier: MIT

#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_LSM303DLH_Mag.h>
#include <Adafruit_NeoPixel.h>

// Parameter 1 = number of pixels in strip
// Parameter 2 = pin number (most are valid)
// Parameter 3 = pixel type flags, add together as needed:
//   NEO_RGB     Pixels are wired for RGB bitstream
//   NEO_GRB     Pixels are wired for GRB bitstream
//   NEO_KHZ400  400 KHz bitstream (e.g. FLORA pixels)
//   NEO_KHZ800  800 KHz bitstream (e.g. High Density LED strip)
Adafruit_NeoPixel strip = Adafruit_NeoPixel(6, 6, NEO_GRB + NEO_KHZ800);
Adafruit_LSM303DLH_Mag_Unified accel = Adafruit_LSM303DLH_Mag_Unified(54321);

// Here is where you can put in your favorite colors that will appear!
// just add new {nnn, nnn, nnn}, lines. They will be picked out randomly
//                                    R   G   B
uint8_t myFavoriteColors[][3] = {{200,   0, 200},   // purple
                                 {200,   0,   0},   // red
```

```
                                {200, 200, 200},   // white
                              };
// don't edit the line below
#define FAVCOLORS sizeof(myFavoriteColors) / 3

// mess with this number to adjust TWINklitude :)
// lower number = more sensitive
#define MOVE_THRESHOLD 45

void setup()
{
  Serial.begin(9600);

  // Try to initialise and warn if we couldn't detect the chip
  if (!accel.begin())
  {
    Serial.println("Oops ... unable to initialize the LSM303. Check your wiring!");
    while (1);
  }
  strip.begin();
  strip.show(); // Initialize all pixels to 'off'
}

void loop()
{
  /* Get a new sensor event */
  sensors_event_t event;
  accel.getEvent(&event);
  Serial.print("Accel X: "); Serial.print(event.acceleration.x); Serial.print(" ");
  Serial.print("Y: "); Serial.print(event.acceleration.y);       Serial.print(" ");
  Serial.print("Z: "); Serial.print(event.acceleration.z);     Serial.print(" ");

  // Get the magnitude (length) of the 3 axis vector
  // http://en.wikipedia.org/wiki/Euclidean_vector#Length
  double storedVector = event.acceleration.x*event.acceleration.x;
  storedVector += event.acceleration.y*event.acceleration.y;
  storedVector += event.acceleration.z*event.acceleration.z;
  storedVector = sqrt(storedVector);
  Serial.print("Len: "); Serial.println(storedVector);

  // wait a bit
  delay(100);

  // get new data!
  accel.getEvent(&event);
  double newVector = event.acceleration.x*event.acceleration.x;
  newVector += event.acceleration.y*event.acceleration.y;
  newVector += event.acceleration.z*event.acceleration.z;
  newVector = sqrt(newVector);
  Serial.print("New Len: "); Serial.println(newVector);

  // are we moving
  if (abs(newVector - storedVector) > MOVE_THRESHOLD) {
    Serial.println("Twinkle!");
    flashRandom(5, 1);  // first number is 'wait' delay, shorter num == shorter
twinkle
    flashRandom(5, 3);  // second number is how many neopixels to simultaneously
light up
    flashRandom(5, 2);
  }
}

void flashRandom(int wait, uint8_t howmany) {

  for(uint16_t i=0; i<howmany; i++) {
    // pick a random favorite color!
    int c = random(FAVCOLORS);
    int red = myFavoriteColors[c][0];
    int green = myFavoriteColors[c][1];
```

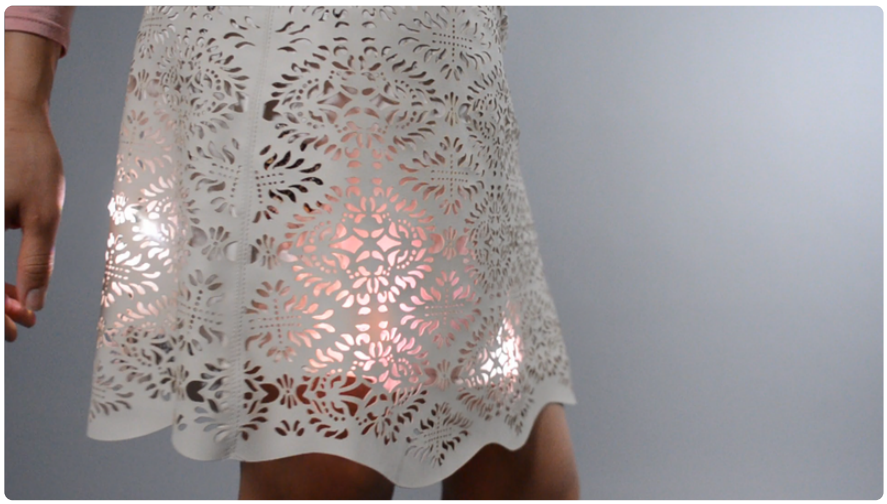
```
    int blue = myFavoriteColors[c][2];

    // get a random pixel from the list
    int j = random(strip.numPixels());
    //Serial.print("Lighting up "); Serial.println(j);

    // now we will 'fade' it in 5 steps
    for (int x=0; x < 5; x++) {
      int r = red * (x+1); r /= 5;
      int g = green * (x+1); g /= 5;
      int b = blue * (x+1); b /= 5;

      strip.setPixelColor(j, strip.Color(r, g, b));
      strip.show();
      delay(wait);
    }
    // & fade out in 5 steps
    for (int x=5; x >= 0; x--) {
      int r = red * x; r /= 5;
      int g = green * x; g /= 5;
      int b = blue * x; b /= 5;

      strip.setPixelColor(j, strip.Color(r, g, b));
      strip.show();
      delay(wait);
    }
  }
  // LEDs will be off when done (they are faded to 0)
}
```



Use a scrap piece of coordinating fabric to sew a small battery pouch. We doubled ours over, used the sewing machine to stitch in an L, then clipped the corner and turned the pouch right-side out.

Stitch the battery pouch to the skirt lining with regular thread in a place nearby the FLORA main board.

Wear it!



Wear your new flashy skirt to your next party, out dancing, make one for each bridesmaid, and be sure to document and share your creations online! Change the colors to match the rest of your outfit or suit the occasion.