

Build Night Kit guide

Warning!

If your Trinket, Neopixel, or any part of your ghost starts getting hot, unplug you the battery pack immediately. Check your circuit for crossed wires. If this problem continues, contact makerstore@uottawa.ca and we will help solve the problem.

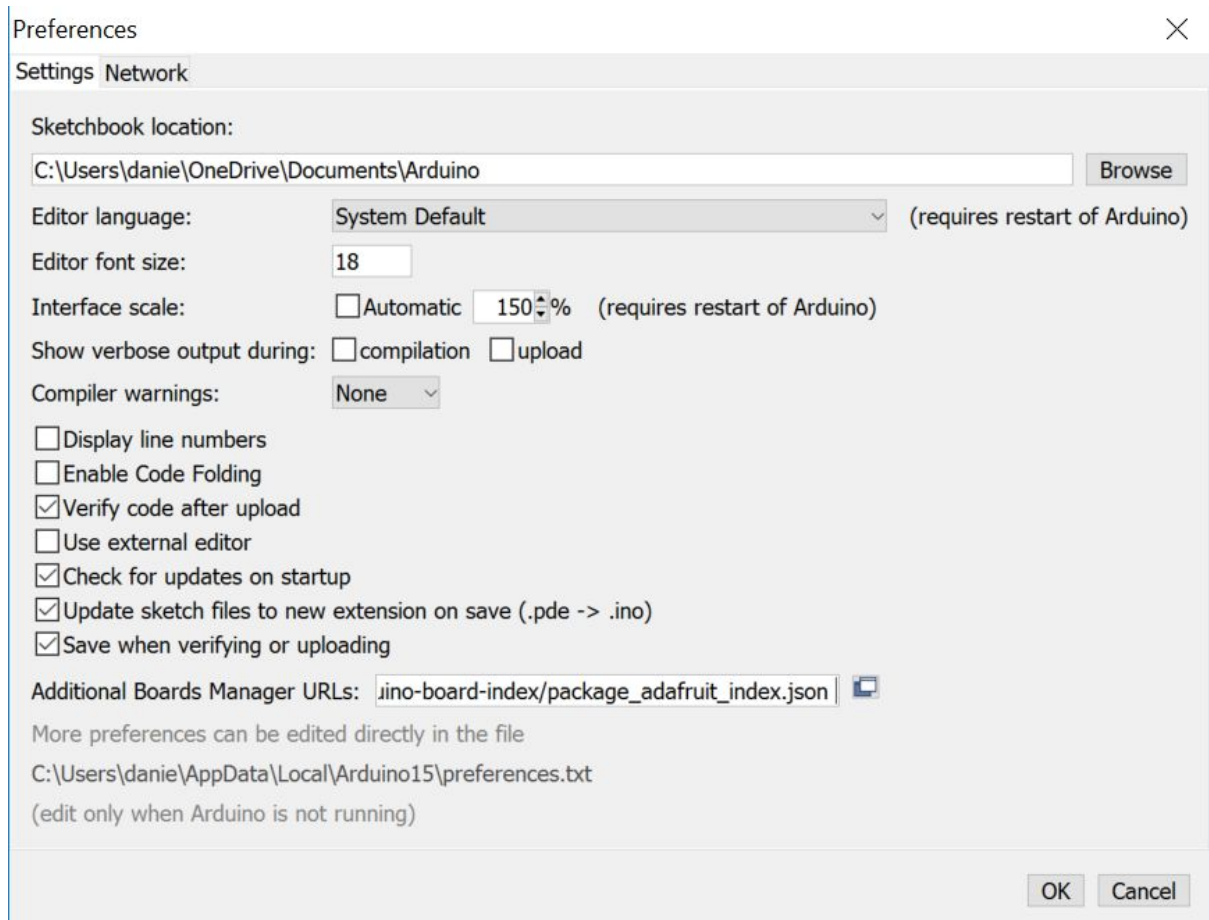
Do no leave it plugged in for long periods of time unattended.

Behind the scenes

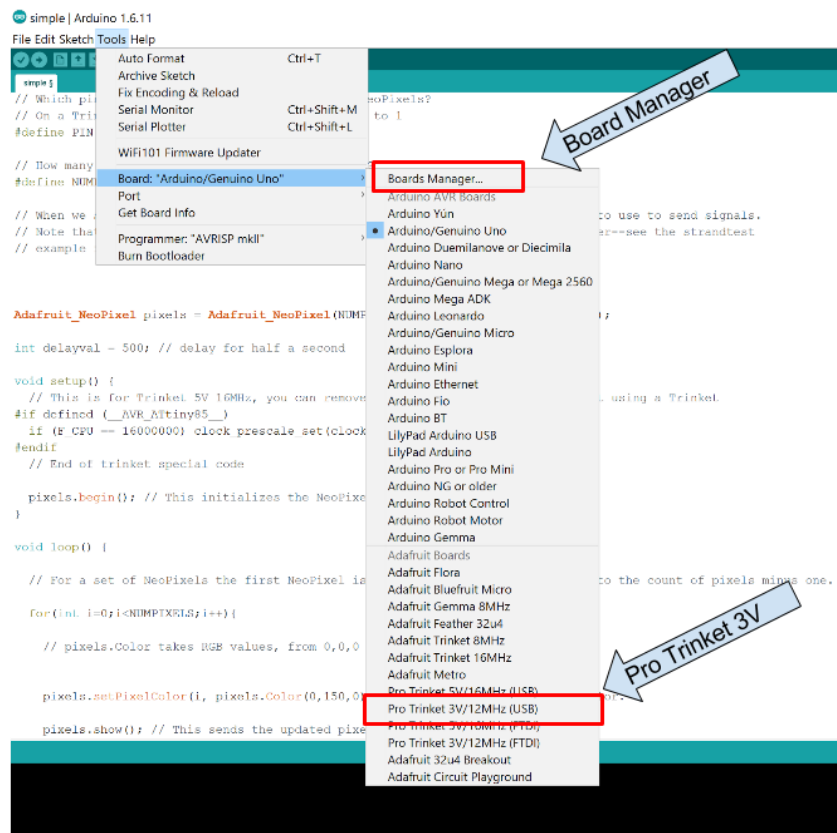
Setting up Arduino:

- Download the Arduino software from: <https://www.arduino.cc/en/Main/Software> and follow the installation instructions. The software is known as IDE, which means Integrated Development Environment. Think of it as a text editor that can process the code.
- Install the trinket board. See <https://learn.adafruit.com/adafruit-arduino-ide-setup/windows-setup?view=all> for more information
- a. Open Arduino.
- b. Click on File -> Preferences
- c. In the “Additional Boards Manager URLs:” box paste

https://adafruit.github.io/arduino-board-index/package_adafruit_index.json



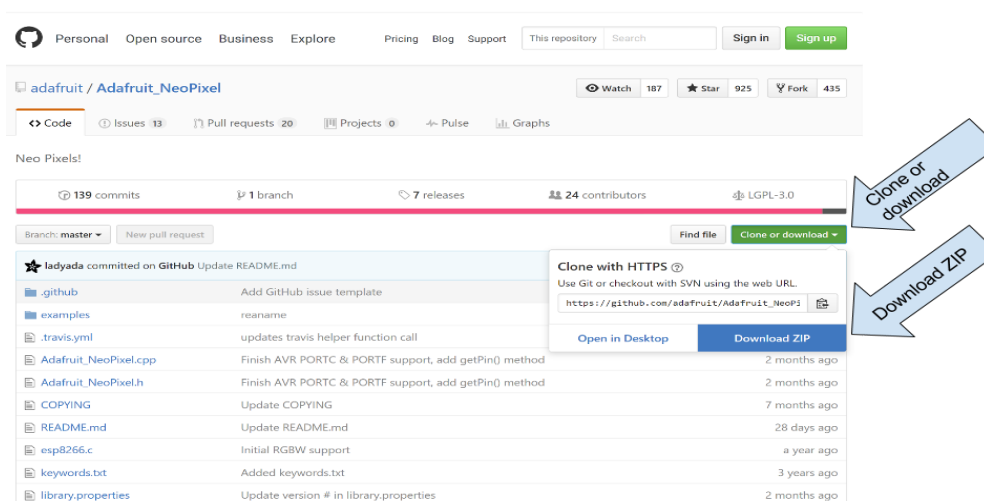
- a. Click ok
- b. Click Tools->Boards->Board Manager



- c. Select Type: Contributed. Install the **Adafruit AVR boards by Adafruit**
- d. Click Close

• Install the neopixel library

- a. Go to https://github.com/adafruit/Adafruit_NeoPixel
- b. Click “Clone or download”, select the “Download ZIP” option. Remember where you saved the zip



- c. In the Arduino IDE, click Sketch -> Include Library -> Add ZIP library. Find the zip. Once you have installed the library, it should remain installed every time you open the computer.



Set up example

In the Arduino IDE, click File -> Examples -> Adafruit NeoPixel -> Simple. A new window should appear with code in it. Make two small changes in the code. Where it says `#define PIN` change the number to the pin you are using. I am using pin 3. Where it says `#define NEOPIXELS` change the number to the number of neopixels you are using.

```
// NeoPixel Ring simple sketch (c) 2013 Shae Erisson
// released under the GPLv3 license to match the rest of the AdaFruit NeoPixel library
```

```
#include <Adafruit_NeoPixel.h>
#ifdef __AVR__
  #include <avr/power.h>
#endif
```

} Includes the necessary libraries

```
// Which pin on the Arduino is connected to the NeoPixels?
// On a Trinket or Gemma we suggest changing this to 1
```

```
#define PIN 6
```

Set this number to the pin that you are using

```
// How many NeoPixels are attached to the Arduino?
```

```
#define NUMPIXELS 16
```

Set this number to the number of Neopixels attached to that pin.

```
// When we setup the NeoPixel library, we tell it how many pixels, and which pin to use to
send signals.
```

How to make a ghost

1. Assemble materials

For this project you will need:

- Square piece of fabric (I will use 12in x12in)
- Batting or something fluffy (about 10cm x 10cm x 10cm)
- Ribbon (about 8in)
- Marker
- Needle (optional)
- Embroidery floss (optional)



2. Cut fabric

First, cut out a square piece of fabric. It can be any size but 12in x12in or bigger is recommended so the ghost isn't too small.



3. Make head

Place the batting in the center of your square of fabric and wrap the fabric around it while compressing the batting to make a head. Tie the ribbon around the neck to keep the batting in place.



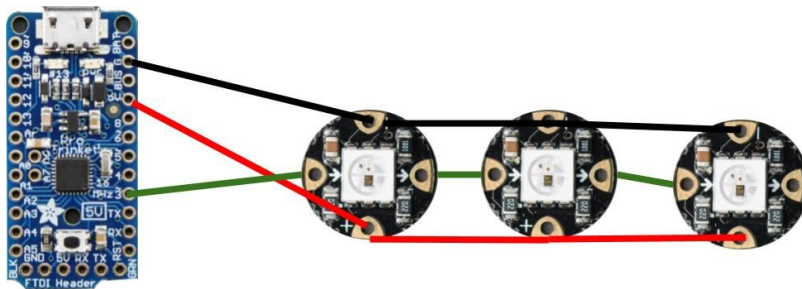
4. Decorate

Use the marker or a needle and embroidery floss to make a face on the ghost. Decorate in any other way you wish.



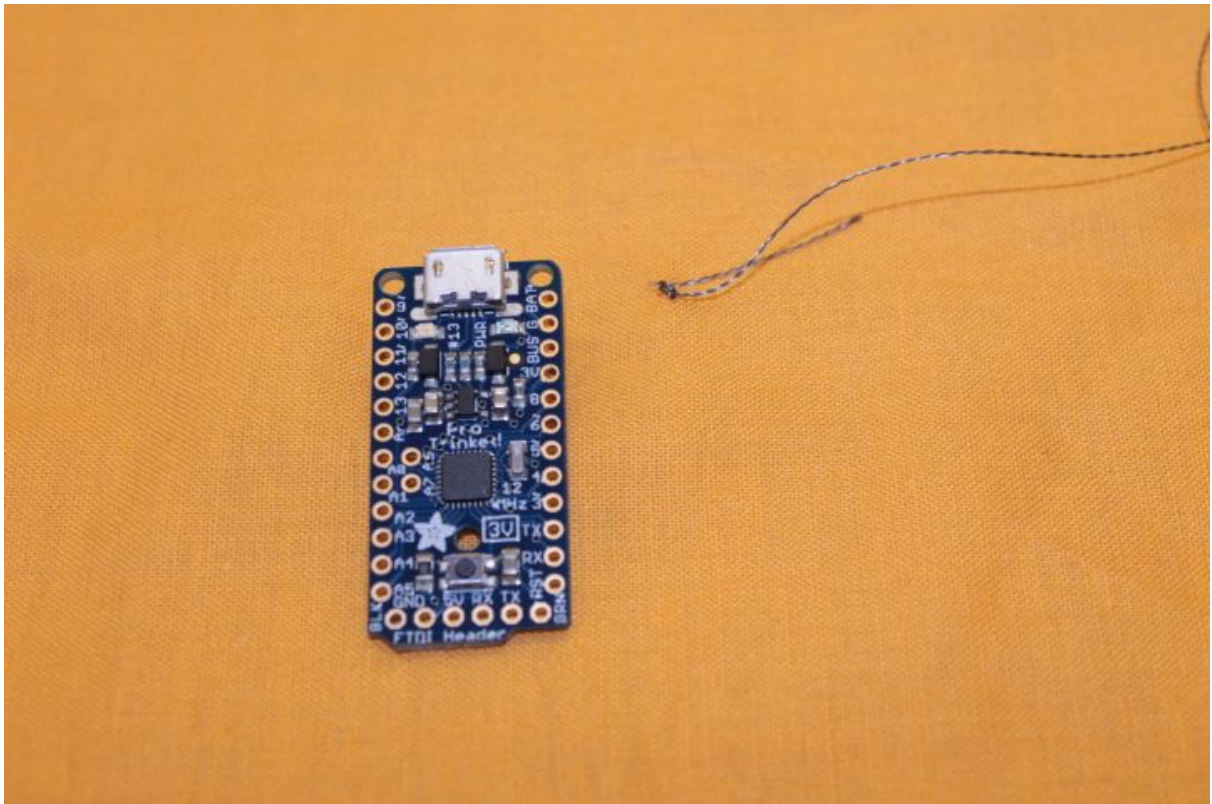
5. Plan your circuit and sew

Lay out your components on the ghost so you can see how to make the connections between all the parts. The diagram below shows that each neopixel (-) is connected to ground (G) on the trinket, the (+) is connected to power (3V) on the trinket and the arrow in (->), to a digital pin. Each new neopixel is attached in series, has the same ground and power and the arrow out is connected to the arrow in of the new light. As you can see in the next picture, the lines cross. For this not to cause problem, sew the red line on one side of the fabric and the green line on the other.

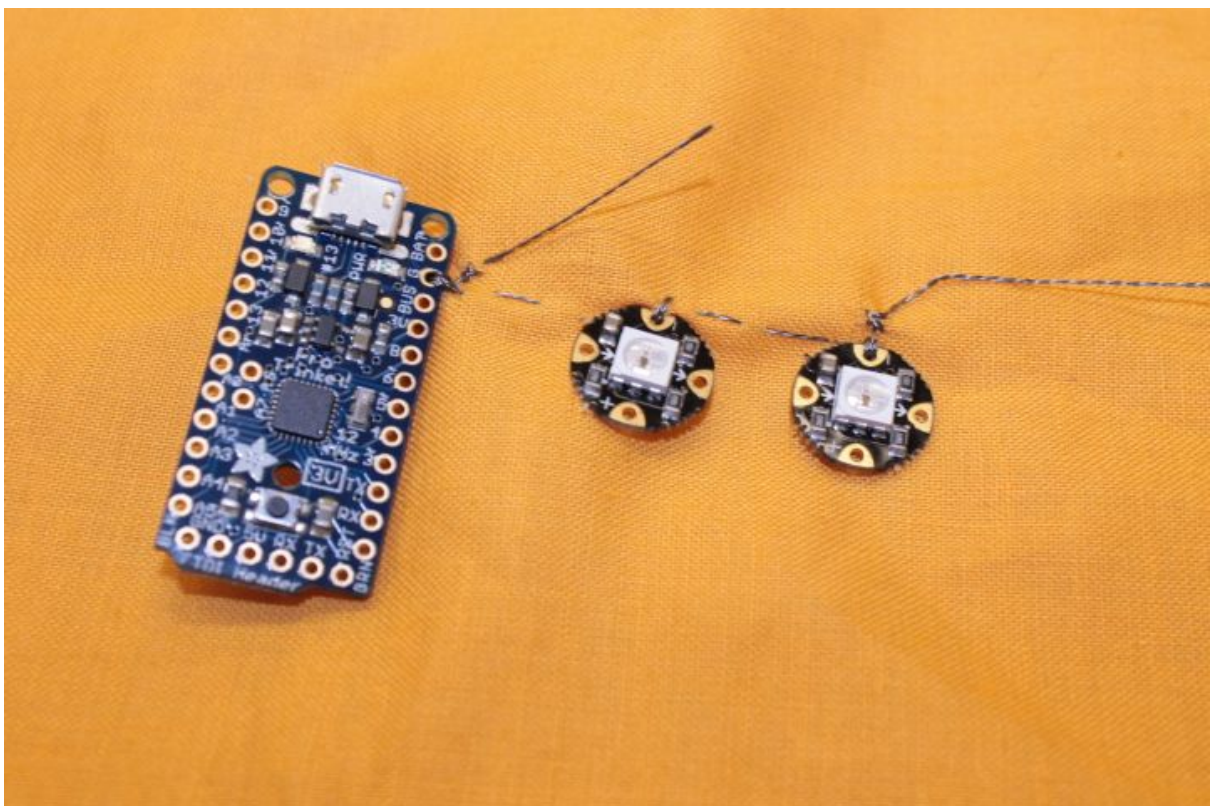


*Note, I found it hard to get a good connection using pin 3 because the metal lines cannot cross. If that happens you will get a short. In this explanation, one wire is on one side of the fabric and the other is on the other side.

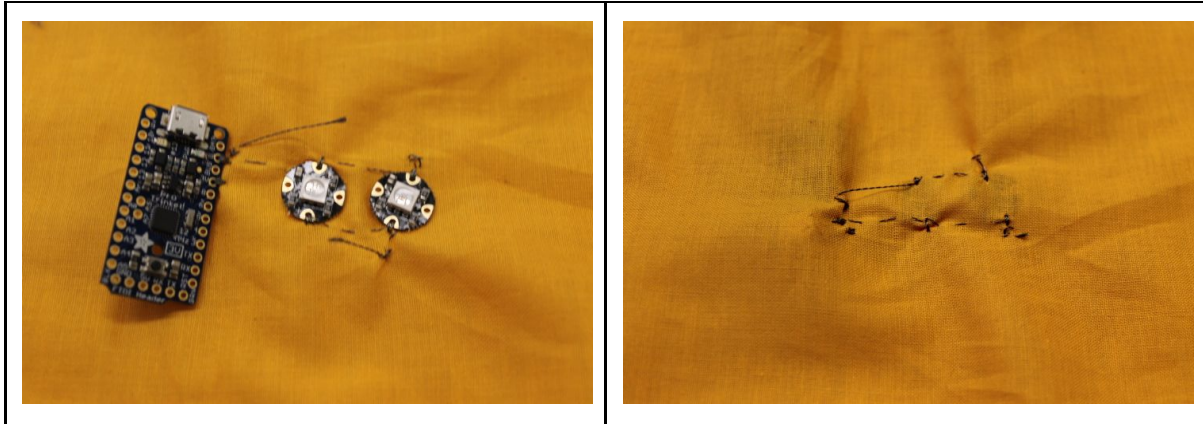
To sew the circuit start by making a knot



Sew the "G" to the the negative on the neopixel

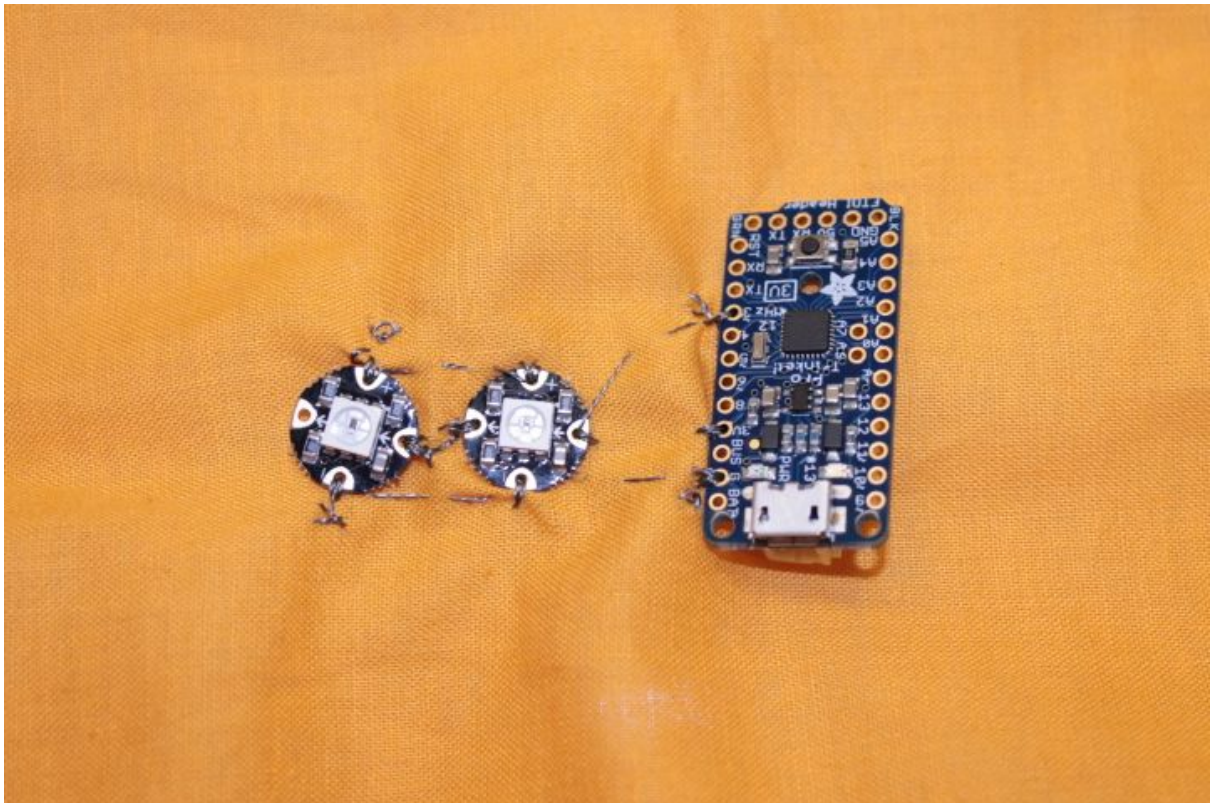


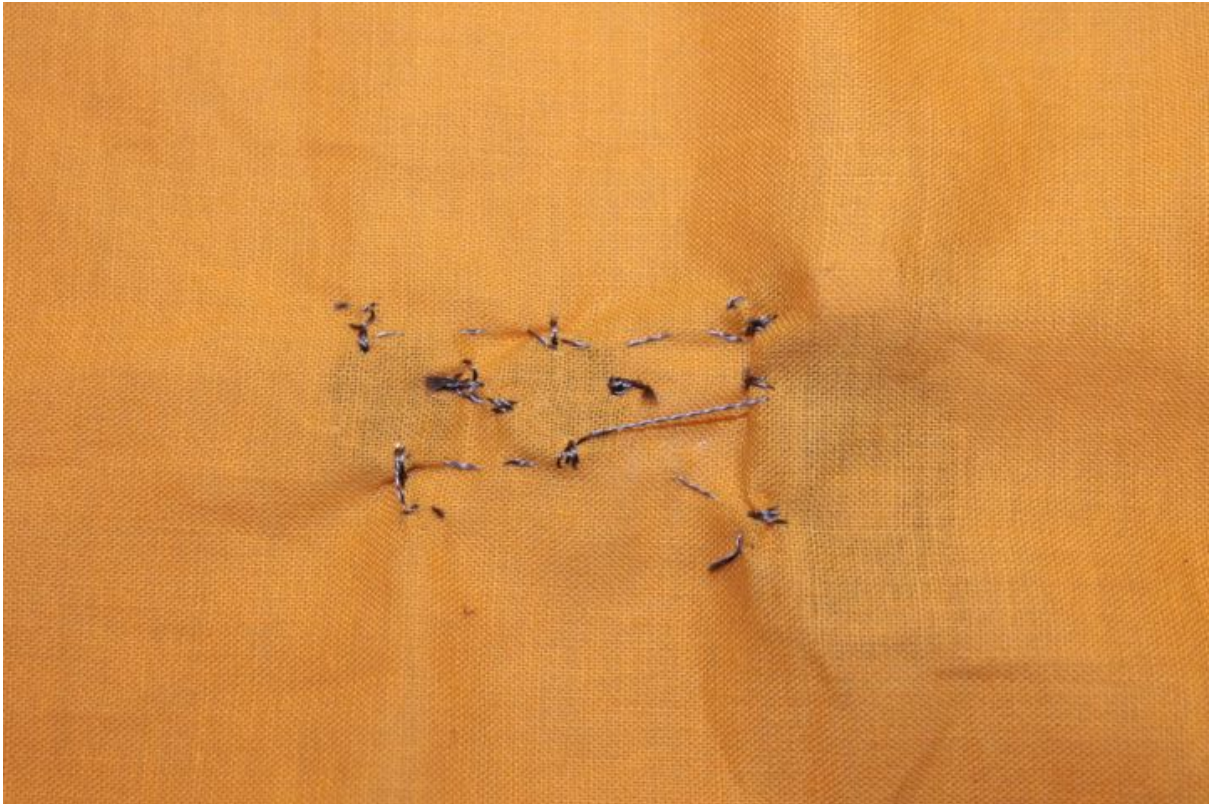
Now this is the tricky part. You need to make sure that your thread is only on one side of the fabric. You are going to connect the 3V to the “+” on the neopixel. As you can see in the picture below, it looks like there is no thread going from the trinket to the “+” on the neopixel. Actually it is underneath. Because you have one big loop and not many little stitches, you need to be extra careful not to pull too tight.



*Note: I am not the best at sewing, so there are a few wrinkles.

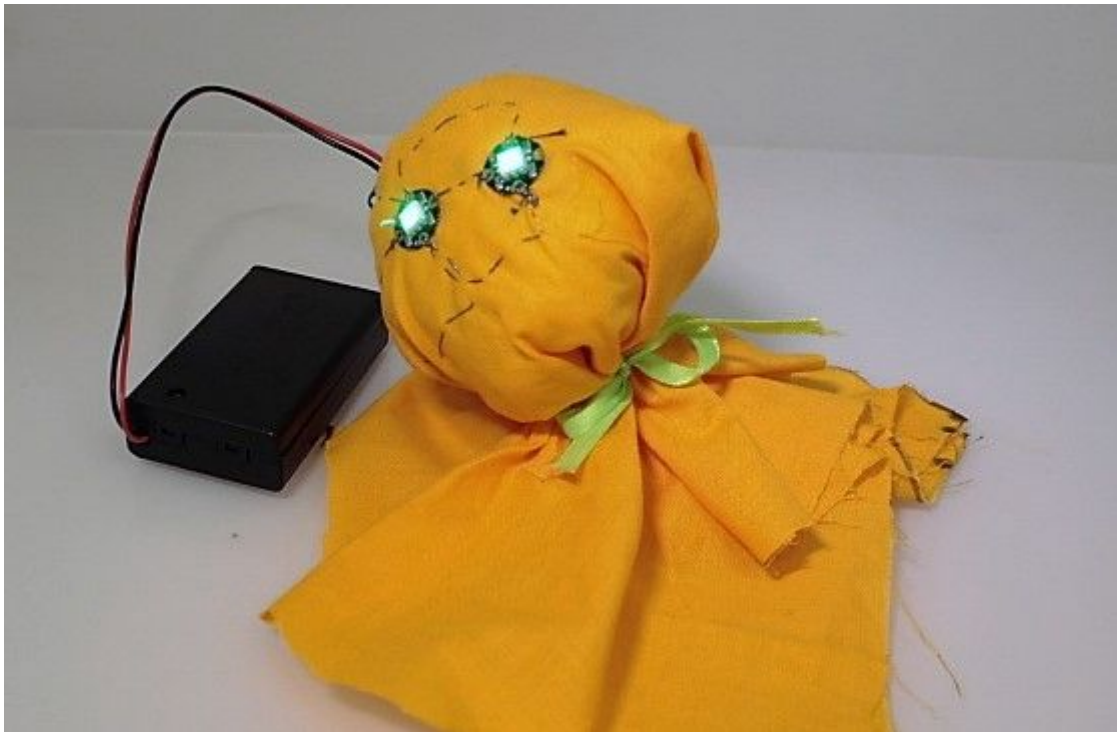
Now attach the pin to the up arrow, on the neopixel. This time you want to be above the fabric. You want to use different thread between the two neopixels





The components wired on the ghost can look like this:





6. Here is the example of the code that I used. Copy and paste it into your arduino IDE.

```
// NeoPixel Ring simple sketch (c) 2013 Shae Erisson
// released under the GPLv3 license to match the rest of the AdaFruit NeoPixel library

#include <Adafruit_NeoPixel.h>
#ifdef __AVR__
  #include <avr/power.h>
#endif

// Which pin on the Arduino is connected to the NeoPixels?
// On a Trinket or Gemma we suggest changing this to 1
#define PIN      3

// How many NeoPixels are attached to the Arduino?
#define NUMPIXELS 2

// When we setup the NeoPixel library, we tell it how many pixels, and which pin to use to
// send signals.
// Note that for older NeoPixel strips you might need to change the third parameter--see the
// strandtest
// example for more information on possible values.
```

```
Adafruit_NeoPixel pixels = Adafruit_NeoPixel(NUMPIXELS, PIN, NEO_GRB +
NEO_KHZ800);
```

```
int delayval = 500; // delay for half a second
```

```
void setup() {
```

```
    pixels.begin(); // This initializes the NeoPixel library.
}
```

```
void loop() {
```

```
    rainbow(20);
```

```
    pixels.show(); // This sends the updated pixel color to the hardware.
```

```
}
```

```
void rainbow(uint8_t wait) {
```

```
    uint16_t i, j;
```

```
    for(j=0; j<256; j++) {
```

```
        for(i=0; i<pixels.numPixels(); i++) {
```

```
            pixels.setPixelColor(i, Wheel((i+j) & 255));
```

```
        }
```

```
        pixels.show();
```

```
        delay(200);
```

```
    }
```

```
}
```

```
uint32_t Wheel(byte WheelPos) {
```

```
    WheelPos = 255 - WheelPos;
```

```
    if(WheelPos < 85) {
```

```
        return pixels.Color(255 - WheelPos * 3, 0, WheelPos * 3);
```

```
    }
```

```
    if(WheelPos < 170) {
```

```
        WheelPos -= 85;
```

```
        return pixels.Color(0, WheelPos * 3, 255 - WheelPos * 3);
```

```
    }
```

```
    WheelPos -= 170;
```

```
    return pixels.Color(WheelPos * 3, 255 - WheelPos * 3, 0);
```

```
}
```