Bluefruit Controlled Macetech RGB LED Shades

Created by John Wall

https://learn.adafruit.com/ledshades

Last updated on 2023-08-29 03:06:16 PM EDT
# Table of Contents

## Project Overview  
- Recommended Precursor Guide  
- Tools Needed

## Parts and Modifications  
- RGB LED Shades  
- Microcontroller  
- Feather 32u4 Bluefruit LE  
- Sensors  
- Electret Microphone  
- Other Sensor Options  
- Parts list

## Wiring and Assembly  
- Circuit Connections  
- Part Placement  
- Soldering the Feather 32u4 Bluefruit LE  
- Prepare the header strip:  
- Add the breakout board:  
- And Solder!  
- Attach the Feather  
- Soldering the Microphone  
- Prepare the header strip:  
- Add the breakout board:  
- And Solder!  
- Attach the Microphone  
- Final Wiring  
- Shades Assembly  
- Build Complete!

## Software Preparation

## Required Arduino Libraries

## Software Testing  
- Neopixel Test  
- BLE Library Test  
- Custom Device Name  
- Test Success!(?)

## Software and Bluetooth Control  
- Clever Code with 2D arrays  
- Programming the Shades  
- You did it!  
- Powering your Shades  
- Bluefruit LE Connect App  
- Controlling the Shades  
- Modes  
- Color Picker  
- Animations  
- Guide Complete!
Project Overview

How do you make a pair of Macetech RGB Neopixel LED Shades even better? Custom hardware, sensors, and bluetooth from Adafruit of course! Read on to learn how to turn a pair of "Hacker Edition" Macetech Shades into fully controllable "smart shades" with the Adafruit Bluefruit app for iOS and Android and some clever Arduino programming!

Recommended Precursor Guide

Before starting this guide, I recommend that you look through the guide on controlling neopixels over bluetooth found here as this project will expand on this basic principle.
Tools Needed

For this project you will need a soldering iron, solder, edge cutters, wire strippers, and tweezers; a basic electronics hacker tool kit.

The Hakko FX-888D (http://adafru.it/1204), my personal favorite Soldering iron

60/40 rosin core solder (http://adafru.it/1886) for easy flowing, strong solder joints

Flush diagonal cutters (http://adafru.it/152) essential for quick, precise snips of wire and leads
One thing you really can't skimp on is a good pair of wire strippers (http://adafru.it/527), these are the best balance of price and quality from Hakko, a very good buy.

Fine tip curved tweezers (http://adafru.it/422), perfect for guiding wires accurately into place for soldering and keeping fingers away from very hot things.

With these tools gathered, we can move on to parts and modifying the LED Shades!
Parts and Modifications

RGB LED Shades

At the heart of this project lies a pair of "Hacker Edition" RGB LED shades from Macetech, a pair of shutter-style shades with a matrix of 68 Neopixel LEDs that face your amazed onlookers in rainbow glory. The shades are made of PCB material cut from the same panel attached with clever joints and hardware, and the power and data lines are broken out to a prototyping space on the right arm that begs for some hacker magic.
Microcontroller

To this frame you can mount any microcontroller and control the 68 Neopixels any way you like. You could add a Trinket (http://adafru.it/1501), Pro Trinket (http://adafru.it/2000), a Feather board (), any microcontroller you want to hack into the shades.

Feather 32u4 Bluefruit LE

The possibilities for a microcontroller are endless, but from Adafruit, the one that makes the most sense for this project is the Adafruit Feather 32u4 Bluefruit LE (http://adafru.it/2829), an Atmega32u4 (Arduino Leonardo) based microcontroller with on board bluetooth low energy, and a lithium polymer battery connection and charging circuitry which would allow you to power the shades with a rechargeable battery like this (http://adafru.it/1781) instead of USB!

Sensors

To add even more capability to your shades, why not hack on a sensor or two?

Electret Microphone

For my pair, I built in a microphone amplifier with auto-gain control, or adaptive volume so sounds of all volumes can be detected and don't max the audio level. It's
With this microphone, the shades can react to sound and glow to the music or ambient noise around you with beautiful animation and visualization. The automatic volume feature makes sure the audio level never passes the edge of the glasses, and even softer sounds can light up the shades.

Other Sensor Options

Make your shades your own! In the hacker spirit, add any sensor you like to available prototyping space on the arm of the shades! Add a light sensor for automatic dimming as to not blind your fans, an accelerometer to react to dance moves, a temperature display, go wild! A great place to start with a variety of sensors is the Sensor pack 900. (http://adafruit.it/176)
Parts list

In summary, to build my pair of shades with an Adafruit Feather and audio sensing you will need these parts:

- Hacker Edition Macetech RGB LED Shades ()
Adafruit Feather 32u4 Bluefruit LE (http://adafru.it/2829)

Electret Microphone Amplifier - MAX9814 with Auto Gain Control (http://adafru.it/1713)

Silicone Cover Stranded-Core Wire (http://adafru.it/2003). It's easily bendable, very thin, and perfect for wearables.

With parts and tools ready, we can begin assembly and testing of the shades!
Wiring and Assembly

Circuit Connections

There are only 3 required connections to run the LED shades: power, ground, and the Neopixel data pin. As long as your microcontroller is capable of running 68 Neopixels and has a pwm pin available, you can hack your own brain onto the prototyping space of the shades. For this project, we will use the default Neopixel control pin, D6, on the Adafruit Feather 32u4 Bluefruit LE to control our LEDs, and the other connections required are shown below.

In this configuration, the shades are powered by a USB power bank in your pocket plugged into the micro USB port of the Feather. This allows for a much longer run time, and less mass on your head for easy wearability. The EN connection to GND
keeps the shades on at all times when plugged in, so if you wanted to be able to switch them on and off without unplugging them, simply wire a slide switch between EN and GND. The microphone output is read by the analog input A0, and the circuit is powered by the USB power pin. To run the shades from battery power, wire the red connections to BATT, and wire in the switch between EN and GND to turn off the shades without unplugging the battery every time.

Part Placement

Find a position on the prototyping area that will fit the Feather and microphone's connections, I recommend aligning the feather's shorter side of pins in the top left of the area as I did to fit a row of headers to support and attach the board and make the required connections. Find a suitable spot for your sensor as well, and make sure it leaves the head of the side arm attachment screw uncovered.

Soldering the Feather 32u4 Bluefruit LE

From Ladyada's guides to the Feather 32u4 BLE and the auto-gain microphone, here's how to solder the header pins and attach them to the board:
Prepare the header strip:
Cut the strip to a length of 10 pins, and
make sure it lines up with the pins
BATT through 5 on the Feather. It will be
easier to solder if you insert it into a
breadboard - long pins down

Remember, only solder headers to the 10 pins EN-3, or your feather won't fit into
your prototyping area!
Add the breakout board:
Place the breakout board over the pins so that the short pins poke through the breakout pads

And Solder!
Be sure to solder all pins for reliable electrical contact.

(For tips on soldering, be sure to check out our Guide to Excellent Soldering (\)).

Attach the Feather

Once you've soldered the top row, you may remove the Feather from the breadboard and solder it into the prototyping area of the hacker edition LED shades arm so the row of headers is in the top left of the area, as shown on my pair.
1. For a stronger connection to the shades, remove the plastic headers spacer with pliers (optional)
2. Then place the Feather into the prototyping area as shown.
3. Solder one pin first to attach it, then hold the arm and reheat the joint, moving the arm to a right angle that's straight and parallel to the feather, and let the joint cool. Then solder all the other pins, and come back to the first one to attach the Feather to the arm.
Soldering the Microphone

Prepare the header strip:
Cut the strip to a length of 5, we will only solder 2 to attach it to the shades and use silicone wire for the remaining 2. It will be easier to solder if you insert it into a breadboard - long pins down.

Add the breakout board:
Place the breakout board over the pins so that the short pins poke through the breakout pads

And Solder!

Be sure to solder the two pins one pin in on the left (Out and Gain) well for reliable electrical contact.

(For tips on soldering, be sure to check out our Guide to Excellent Soldering).

Attach the Microphone

After you've finished soldering, use your flush cutters to carefully cut away the excess headers so your microphone has two header pins in OUT and GAIN.
Like attaching the Feather, find a place on the available prototyping area to insert the sensor's headers.

1. Press the sensor all the way to the board, and solder the Gain pin on the other side. Then reheat the joint, pull the sensor to sit parallel to the board, and let the joint cool. Then solder the Out pin, and resolder the first one in place.
2. To set the gain of the microphone, or its sensitivity, solder a bridge between the Vdd and Gain pins on the mic.

If you want to attach another sensor to your shades, simply anchor it in a similar way with headers or to a space on the prototyping area, and connect its output in the proper way to an analog pin on your Feather or other microcontroller.
Once you've finished soldering the Feather and Microphone, cut all protruding pins and clean any flux or leftover pin from the board. Now we can move on to wiring the circuit!

Final Wiring

Now, review and identify where the pins required for this project are located from the "Circuit Connections" sections above, and refer to the circuit diagram to plan out your wiring.

Once you have wire lengths that work for all your connections, strip the ends with your wire strippers, and it's good practice to tin your wires before wiring anything together. Once you have your wires ready, wire your pins together as the circuit diagram shows, paying attention to the labelling on the three Neopixel driving pins in
the top right of the image above. GND at the top, Neopixel data in the middle, and 5V at the bottom.

A properly cut, stripped, and tinned segment of wire
1. Wire the USB pin of the Feather to the 5V pin of the shades
2. Then, connect the microphone's VDD pin to the shades' 5V pin.
3. Wire the Feather's pin 6 to the shades' Neopixel data pin (the center of the 3 on the shades), and the mic's Out pin to A0 on the Feather.

Finally, ground the circuit in the same way you wired the power, between the Feather, shades, and mic's GND pins.
With the circuit all wired up, check your work against the circuit diagram, clean between pins, and if you’d like, test the connections with your multimeter for shorts. If it all looks good, the soldering for this project is complete! Read on to assemble your shades.

Shades Assembly

To assemble the frame of the shades, Garrett over at Macetech created this in-depth assembly video, have a look:

Build Complete!
Once your shades are wired and assembled, we can move on to testing and software, the clever programming behind this project! Get your Arduino IDE ready, we're goin' coding!

Software Preparation

To program the LED Shades, we must first install and set up the Arduino IDE to accept the Feather as a programmable board. From Lady Ada's Feather 32u4 Bluefruit LE guide on the learning system, here are the instructions to get all set up, then we'll move on to testing and code!

- **Arduino IDE Setup**
- **Using With the Arduino IDE**

Required Arduino Libraries

Once our Arduino IDE is set up for our Feather board, we can install the required Arduino libraries for this project by unzipping them and dropping them in your arduino "libraries" folder. If you haven't added Arduino libraries, follow this [handy guide](#) on the Learning System, but if you're a pro, go ahead and download them below.

- **Download the Adafruit Neopixel Library**
Software Testing

With our IDE set up and our libraries installed, we can get to programming and testing! The first test we should perform is a test of the Neopixel matrix with a modified example from the Neopixel library! This test will run the test code across the first row of the shades. Open your newly set up IDE and paste this code into the window. Go to your "Tools"->"Board" menu, and select the Feather 32u4. Plug your board in and find its serial port and select it, then upload. If you have trouble with any of this, follow this guide to upload to your Feather.

Neopixel Test

```c
#include "Adafruit_NeoPixel.h"
#ifdef __AVR__
#include "avr/power.h"
#endif
#define PIN 6

// Parameter 1 = number of pixels in strip
// Parameter 2 = Arduino pin number (most are valid)
// Parameter 3 = pixel type flags, add together as needed:
// NEO_KHZ800 800 KHz bitstream (most NeoPixel products w/WS2812 LEDs)
// NEO_KHZ400 400 KHz (classic 'v1' (not v2) FLORA pixels, WS2811 drivers)
// NEO_RGB  Pixels are wired for GRB bitstream (most NeoPixel products)
// NEO_RGBW  Pixels are wired for GRBW bitstream (NeoPixel RGBW products)
Adafruit_NeoPixel strip = Adafruit_NeoPixel(14, PIN, NEO_GRB + NEO_KHZ800);

void setup() {
  // This is for Trinket 5V 16MHz, you can remove these three lines if you are not
  // using a Trinket
  #if defined (__AVR_ATTiny85__)
    if (F_CPU == 16000000) clock_prescale_set(clock_div_1);
  #endif
  // End of trinket special code

  strip.begin();
  strip.show(); // Initialize all pixels to 'off'
}

void loop() {
  // Some example procedures showing how to display to the pixels:
  colorWipe(strip.Color(255, 0, 0), 50); // Red
  colorWipe(strip.Color(0, 255, 0), 50); // Green
  colorWipe(strip.Color(0, 0, 255), 50); // Blue
  colorWipe(strip.Color(0, 0, 0, 255), 50); // White RGBW
}
```

©Adafruit Industries
// Send a theater pixel chase in...
theaterChase(strip.Color(127, 127, 127), 50); // White
theaterChase(strip.Color(127, 0, 0), 50); // Red
theaterChase(strip.Color(0, 0, 127), 50); // Blue

rainbow(20);
rainbowCycle(20);
theaterChaseRainbow(50);
}

// Fill the dots one after the other with a color
void colorWipe(uint32_t c, uint8_t wait) {
for(uint16_t i=0; i<strip.numPixels(); i++) {
  strip.setPixelColor(i, c);
  strip.show();
  delay(wait);
}
}

void rainbow(uint8_t wait) {
  uint16_t i, j;
  for(j=0; j<256; j++) {
    for(i=0; i<strip.numPixels(); i++) {
      strip.setPixelColor(i, Wheel((i+j) & 255));
    }
    strip.show();
    delay(wait);
  }
}

// Slightly different, this makes the rainbow equally distributed throughout
void rainbowCycle(uint8_t wait) {
  uint16_t i, j;
  for(j=0; j<256*5; j++) { // 5 cycles of all colors on wheel
    for(i=0; i<strip.numPixels(); i++) {
      strip.setPixelColor(i, Wheel(((i * 256 / strip.numPixels()) + j) & 255));
    }
    strip.show();
    delay(wait);
  }
}

//Theatre-style crawling lights.
void theaterChase(uint32_t c, uint8_t wait) {
for (int j=0; j<10; j++) {  //do 10 cycles of chasing
  for (int q=0; q<3; q++) {
    for (uint16_t i=0; i<strip.numPixels(); i++) {
      strip.setPixelColor(i+q, c);    //turn every third pixel on
    }
    strip.show();
    delay(wait);
    for (uint16_t i=0; i<strip.numPixels(); i++) {
      strip.setPixelColor(i+q, 0);        //turn every third pixel off
    }
  }
}

//Theatre-style crawling lights with rainbow effect
void theaterChaseRainbow(uint8_t wait) {
  for (int j=0; j<256; j++) {     // cycle all 256 colors in the wheel
    for (int q=0; q<3; q++) {
      for (uint16_t i=0; i<strip.numPixels(); i++) {
        strip.setPixelColor(i+q, Wheel( (i+j) % 255));    //turn every third pixel
      }
    }
  }
}
BLE Library Test

If you would like to test the BLE library, perform the steps listed in the Feather 32u4 Bluefruit LE guide here:
Custom Device Name

One of the cool things you can do with the Feather 32u4 Bluefruit LE is add a custom device name! I named mine "Shades!" so it appears on my phone as such, and to do so yourself you can follow this guide here!

Custom Device Name Instructions

To save your custom name, disable the factory reset of the bluetooth module in the code.

Test Success!(?)

If your upload was a success, you should see a rainbow and other test patterns on the first row of your shades, and see a working bluetooth link! If not, check your code and wiring, and you'll get it I'm sure! Refer to the uploading guide or the previous steps and the circuit diagram to find your issue, or post it up in the Adafruit support forums. Now we can move on to the project's code!

Software and Bluetooth Control
Clever Code with 2D arrays

With a built and tested pair of custom Neopixel Shades, we can now move on to what makes these shades tick, the code! This program is a great use of what's called a two dimensional array, or essentially a grid of datatypes that can be accessed by row and column. Because the Neopixels are connected in a chain, if we want to control the leds from left to right on the shades, or control them by row and column as they appear on the front of the shades, we can't just count up from 0 as it would zig-zag. The array solves this by putting the addresses of the leds into a grid as they appear on the shades, so each can be addressed and controlled in a logical way. Read more [here](#) if you're interested.

Programming the Shades

```cpp
int ledArray[16][5] = // Because the Neopixels are in a chain, to access
// their addresses in a grid we use a 2D array. Nifty!
{ // their addresses in a grid we use a 2D array. Nifty!
  { -1, 14, 43, 44, -1},
  {33, 15, 42, 45, 67},
  {12, 16, 41, 46, 66},
  {11, 17, 40, 47, 65},
  {30, 18, 39, 48, 64},
  { 9, 13, 38, 49, 63},
  { 8, 20, 37, 50, -1},
  { 7, 21, -1, -1, -1},
  { 6, 22, -1, -1, -1},
  { 5, 23, 36, 51, -1},
  { 4, 24, 35, 52, 62},
  { 3, 25, 34, 53, 61},
  { 2, 26, 33, 54, 60},
  { 1, 27, 32, 55, 59},
  { 0, 28, 31, 56, 58},
  { -1, 29, 30, 57, -1}
};
```
Ok, now that we've nerded out a bit, we can move on to finally giving life to your shades! If your IDE is set up, your libraries are installed properly, and you've found somewhere to sit down, prepare yourself to witness the awesomeness of your new pair of Bluefruit controlled Macetech Neopixel shades! Plug in your shades, and upload the project code in the same way you tested the LEDs and bluetooth. If your upload is a success, read on to learn how to control your shades!

**Download the Neopixel Shades Code**

You did it!

Congrats on building your own pair of Ada-hacked Macetech RGB LED shades!

**Powering your Shades**

Those 68 Neopixels on your new shades need a lot of juice, so the best way to power them as you wear them out and about is a USB power bank like [this](#) one and a ~3-4 ft micro USB cable. At this project's comfortable brightness, the shades run at about ~100ma, which should last roughly 20+ hours!

Be careful! Some power banks can't handle the sharp current draw of the shades and will cause glitchy issues!
Bluefruit LE Connect App

Now they're built and programmed, download the Adafruit Bluefruit LE Connect app for iPhone or Android to get controlling them!
1. Open the app, and select the shades in the device list
2. Next, pick the "controller" option
Modes

Color Picker

![Color Picker](image)

R:255 G:0 B:218

SEND
The first mode to control the shades is the Bluefruit LE Connect's color picker. Select an RGB color value from the wheel, and press "send." The color will now zig-zag across the shades! To turn them off, slide the darkness bar to black and press send again.

Animations

The second mode to control the shades is a selection of four animations triggered by the "control pad" option of the "controller" menu, below "color picker."
The button functions are as follows:

1. Single color fading rainbow, all LEDs
2. Full spectrum rainbow flowing from left to right
3. Full spectrum rainbow VU meter L-R
4. Single color fading rainbow VU graph L-R

Guide Complete!

And that's it! Congrats on your custom pair of hacker edition Macetech RGB LED shades! Show us your pair on twitter @adafruit, and rule the dance floor, or your work bench, that's cool too, it's up to you!