LED Strip Spectrum Analyzer: An RGB Lighting Dream

Jiin So*

Physics Department, Boston University

(Dated: April 29, 2021)

Today, RGB lighting through LEDs have been a project many take on to decorate their rooms. This is especially popular among the gaming enthusiasts. Taking this to another level, instead of using static RGB lights, a WS2812B LED strip of individually addressable LEDs can be transformed into an audio spectrum analyzer. An Arduino Uno board intakes audio signals from an external device, performs a Fourier analysis of the incoming signal, then uses the information to assign a color to specific LEDs depending on the amplitude of frequency the LED corresponds to. The colors are taken from specific color palettes to give pleasure to the eyes. A push-button switch can be used to change the color schema modes. Using an audio jack splitter, we have the option to connect it to external speakers or headphones, so that the music can be enjoyed both audibly and visually.

I. IMPLEMENTATION

Using a 3.5 mm audio jack, both left and right audio signals connects to the Arudino's analog input AO after high-pass filters. The high-pass filters will have a 3dB point about 380 Hz, filtering out the 60 Hz noise as well as audio signals too low for the human ear to hear [5]. I will use the FFT library [2] and the ADC in the Arduino to sample the analog audio signal and put them into 32 different bins of frequency, each with a corresponding amplitude[1]. This amplitude will then be transferred into the data line of the WS2812B LED strip to address the individual LEDs, where each LED represents a different frequency range, and the hue of the LED will be determined by the amplitude of the frequency range bin. This will be done using the FastLED library [4]. The 330 resistor will act to reduce noise in the data line, and the 1000 uF will act to stabilize the power source [3]. The switch will allow the user to change between different color palettes as well as change to a "stationary" mode in which it simply glows without the spectrum analyzer effects.

II. BLOCK DIAGRAM AND DESIGN SKETCH

Attached are the block digrams and design sketch of the project. Note that the Arduino will be powered separately from the 5V 10A external power supply connected to the LED strip.

III. LIST OF NEEDED PARTS

- 1 of Arudino Uno
- 1 of WS2812B LED strip: 5m white 60 leds/m
- 1 of external power supply: AC to DC converter 5 V 15 A

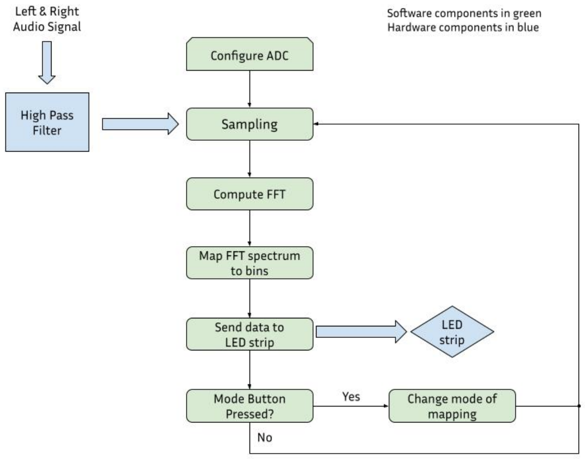

FIG. 1. Block diagram of flow of data and procedures

- 1 of 3.5 mm audio jack female terminal block
- 1 of 3.5 mm audio jack headphone splitter
- 1 of 3.5 mm audio jack male to male connector
- 3 of 100 kΩ resistor
- 3 of 5 kΩ resistor
- 1 of 330 Ω resistor
- 1 of 10 μF capacitor
- 1 of 1000 μF capacitor
- 2 of 100 nF capacitor
- 1 of push button

* jiinso@bu.edu

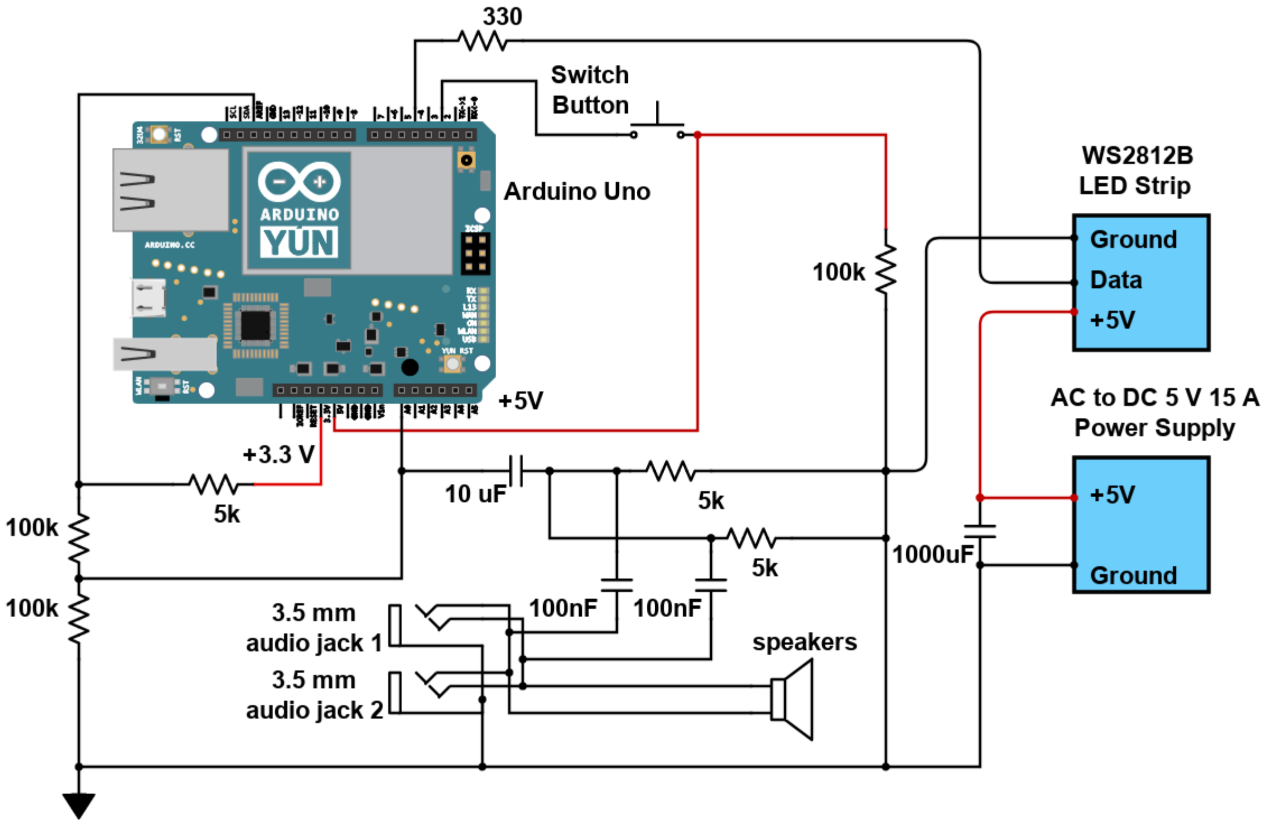

FIG. 2. Circuit design sketch: designed through Digikey Schemeit

IV. SCHEDULE

Week 3/29 ~ 4/4

- Obtain all needed parts
- Start wiring the frequency-audio inputs with the Arduino Uno

Week 4/5 ~ 4/11

- Test frequency signal of audio input
- Test frequency spectrum analysis of audio input (songs) using FFT library code

Week 4/12 ~ 4/18

- Test out the WS2812B LED strip with power and addressing
- Wire rest of the parts according to the design sketch
- Learn how to send frequency spectrum data to individual LEDs using the FastLED library

Week 4/19 ~ 4/25

- Create different color palettes for different modes of operation of the system: making things look pretty!
- Optional – switch operational mode to frequency-color, and amplitude-brightness: set each bin of LEDs to a certain color (representing frequency), change brightness of the bins.

REFERENCES

[1] Mads Aasvik. *What Is FFT and How Can You Implement It on an Arduino?* URL: https://www.norwegiancreations.com/2017/08/what-is-fft-and-how-can-you-implement-it-on-an-arduino/.

[2] Enrique Condes. *kosme/arduinoFFT*. URL: https://github.com/kosme/arduinoFFT.

[3] Dejan. *How To Control WS2812B Individually Addressable LEDs using Arduino.* URL: https://howtomechatronics.com/tutorials/arduino/how-to-control-ws2812b-individually-addressable-leds-using-arduino/.

[4] kriegsman. *FastLED*. URL: https://github.com/FastLED/FastLED.

[5] Shajeeb. *32-Band Audio Spectrum Visualizer Analyzer*. URL: https://create.arduino.cc/projecthub/shajeeb/32-band-audio-spectrum-visualizer-analyzer-902f51.