Emily Robey

CST 120 Spring ‘24

05/05/24

**CST 120 Project Proposal:**

**Light it Up**

**Project name:** Light it Up

**Group Members:** Emily Robey

**Project goal:** Control an LED strip via claps and a website

**Project overview:**

LED lights are everywhere and we use them for a variety of purposes every day. My project proposes to allow users a unique way to control programmable LED strips via both clapping and through a website interface.

This project has three main parts, user control via website, user control via claps and the different states of the lights. The user will be able to access a basic website that allows them to switch between web control and clap control options for the LEDs. In either state, the user can switch between light colors, solid vs. blinking lights, or even set their lights to respond, or “dance”, to music. The web interface can set the same states, but also serves as a “switch” to allow the user to go back and forth between “web mode”, or “clap mode”.

The practical purpose of this project is two-fold. First, it’s entertaining to have lights that dance and go between different modes, who doesn’t enjoy a good clap-controlled light? Second, it serves as a security device. The user can decide to turn lights on when they aren’t home and can even change which lights are on so it looks like they’re inside playing with the lights.

In addition to practical use, the project that I am proposing contains the building blocks for much larger projects. All of the pieces that I am planning on bringing together open the doors for a variety of interesting and unique projects that I would like to build including controlling lighting around my home and inside of my snake enclosures and building a system that will alert me when any of the enclosures get above, or below a certain temperature, or humidity. This is my primary interest in pursuing this project as I want to test my technical ability to build such devices before trusting my snakes’ comfort and potentially safety with my own homemade devices.

**Design overview – Hardware:**

The design of this project is relatively simple. A breadboard will be used to connect all of the pieces together i.e. Arduino + wifi, microphone, LED Strip, power supplies. The Arduino and LED strip will have separate power supplies as this is required for the number of lights that I want to be able to power. I am looking into a case of sorts to contain everything in a neat and safe way; I may see if I can 3D print something in the machining lab at OIT.

A 150 light programmable LED Strip will be connected to a 5V AC/DC power supply adapter with a 100 mF capacitor placed directly into the 5V and ground of the adapter to ensure a more stable voltage to the LED strip. The Arduino will also be powered by an identical 5V AC/DC power supply adapter that will be plugged into its barrel power socket. Even though they are independently powered, the Arduino and LED grounds will be connected to complete the circuit.

The LED strip input will be connected to the Arduino via the breadboard, the microphones DO pin and a 330-OHM resistor to prevent burning out either the Arduino, or the LED strip. There will be wires going to at least 3 pins of the Arduino to control microphone input, the LED lights and the WIFI module. There is a possibility that I may need to add more pin connections, but as of right now this design appears sufficient.

**Parts required**

* Arduino Uno WiFi REV2
  + Could also use a wifi shield, but this is easier/less code
* Arduino Uno R3
  + For some of the initial prototyping of non-wifi parts
* 2 \* AC/DC adaptor DC 5V
  + ALITOVE 5V 5A AC to DC Power Supply Adapter Converter 5.5x2.5mm Plug AC 100V~240V Input for WS2812B WS2811 SK6812 LED Pixel Strip Light CCTV Camera Security System
* Microphone
  + DEVMO 5PCS Microphone Sensor High Sensitivity Sound Detection Module Compatible with Arduino PIC AVR
* Breadboard
  + From Arduino kit
* Single LEDs for initial prototypes
  + From Arduino kit
* WS2812B LED strip lights
  + ALITOVE WS2812B LED Strip 16.4ft 150 LEDs Individually Addressable RGB LED Pixel Strip Lights 5050 SMD Dream Color Digital Programmable LED Lighting Waterproof IP67 Black PCB DC 5V for Decor Lighting
* Wires
  + From Arduino kit
* 300 OHM & 560 OHM Resistors
  + From Arduino kit – will need a few different ones probably
* 100 mF Capacitor
  + From Arduino kit – will \*potentially\* need more depending on which route I end up taking, but definitely need one for AC/DC power adapter -> LED light connection
* Small screwdriver set
  + Or single 5/64th

**Design overview – Code:**

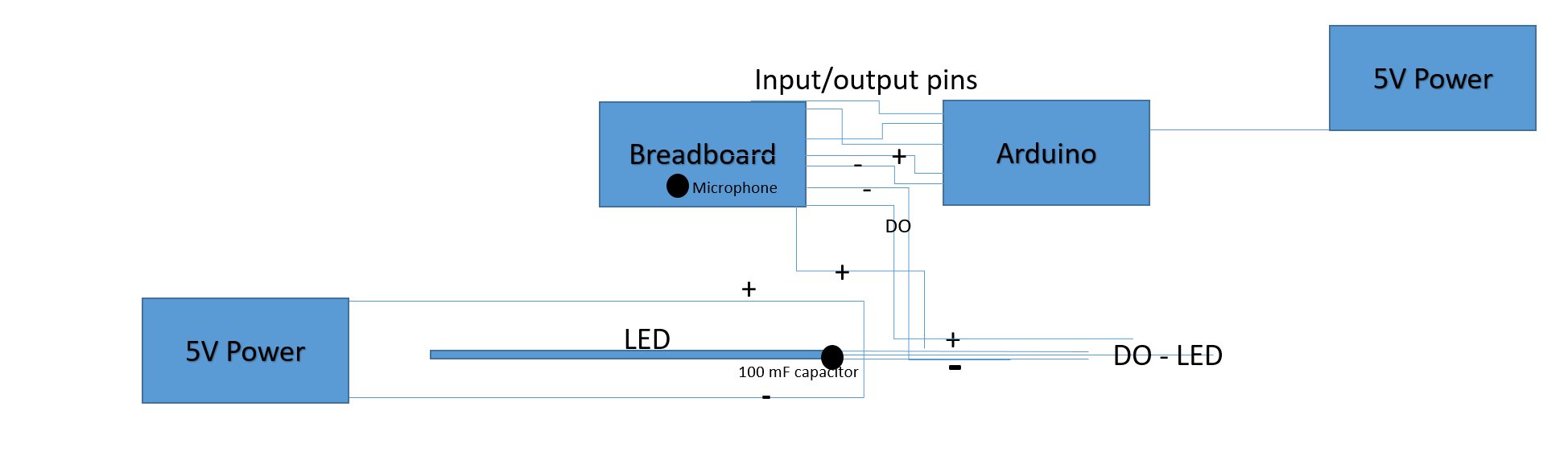
When compared to the hardware, the code for this project is the more complicated part. As of right now, I will primarily be using the FastLED, SPI and WiFiNINA libraries along with the standard Arduino IDE library. I will also be using basic HTML, CSS and potentially a bit of Javascript to build a simple website for users to interact with.

The goal is for users to be able to fully control the lights based on the mode that they’re in. While in the clap state the microphone will pick up claps (based on a frequency detection range in the code). The code will add up claps making adjustments to the LED strip state as it goes. The adjustments to the LED lights will come from algorithms that calculate what color, intensity, speed of change, or on/off status each mode should produce along with the FastLED library which allows for excellent control over each individual LED light on the strip. During clap mode, the microphone serves two purposes, first to pick up claps and second to adjust the lights colors and patterns based on the musical frequencies when in music mode.   
  
When in web mode, the microphone will only be used when the LED strip is in music mode to pick up the frequencies of the music and output the correct light color and pattern. In all other instances, the user will control the state of the LED strip via buttons embedded on the Light it Up website. In addition to the LED state change buttons, the user will be able to push a button to put the Arduino into “clap” or “web” mode depending on their preference. The result of the user’s input will be passed from the website to the code via the WiFiNINA library which handles all POST and GET request in an easy to use manner. Both the LED strip and program states will be reflected as functions in the code that perform the logic in each LED state, or follow the logic for clap functionality, or web functionality.  
  
The LED strip states will require a bit of code, as mentioned above, including algorithms to handle the “dancing” of the lights. Given the time constraints, I will be using relatively simple algorithms based on vibrations and frequencies of the music that will control all aspects of the lights in a consistent manner (there are much more exact ways to get the lights to respond to music, but it requires much more code and hardware). The FastLED library makes controlling the hardware inside of the LED strips very easy to do.

The code logic is broken up into the following sections:

* **Microphone functionality**
  + Picking up claps and frequencies/vibrations from music
* **LED states**
  + Clap once – turn all lights on in 1 color
  + Clap twice – turn all lights on in another color
  + Clap three times – Blink lights in all colors
  + Clap four times – lights respond to music
  + Clap five times – turn lights off
  + Note: I may add in additional colors, patterns, etc, but I want to start with at least 1 solid color switch
* **States for the entire program** 
  + Web vs. clap
  + I will use the website like a switch to change the modes i.e. if it’s in “clap” mode, or if it’s in “website control” mode.
* **WIFI connection**
  + Handling input/output related to the website
  + Handling the WIFI connection itself
* **Front end – website to control lights**
  + Activate the different light options listed above from a website as well so that I can turn lights on/off even when I’m not home.
  + Same modes as clap state, but basic buttons will be used to change modes

**Basic Project Diagram**



**Other Diagrams**

**Note:** These are basic diagrams that I either found online and have followed, or made myself with fritzing for each piece of the project. I will be combining these pieces together to make my finished product. I want to get each piece independently (I’m almost done with that) and then I will consolidate everything into a single project that responds to all of the required inputs with the appropriate outputs.

**Wifi light control with single LED**

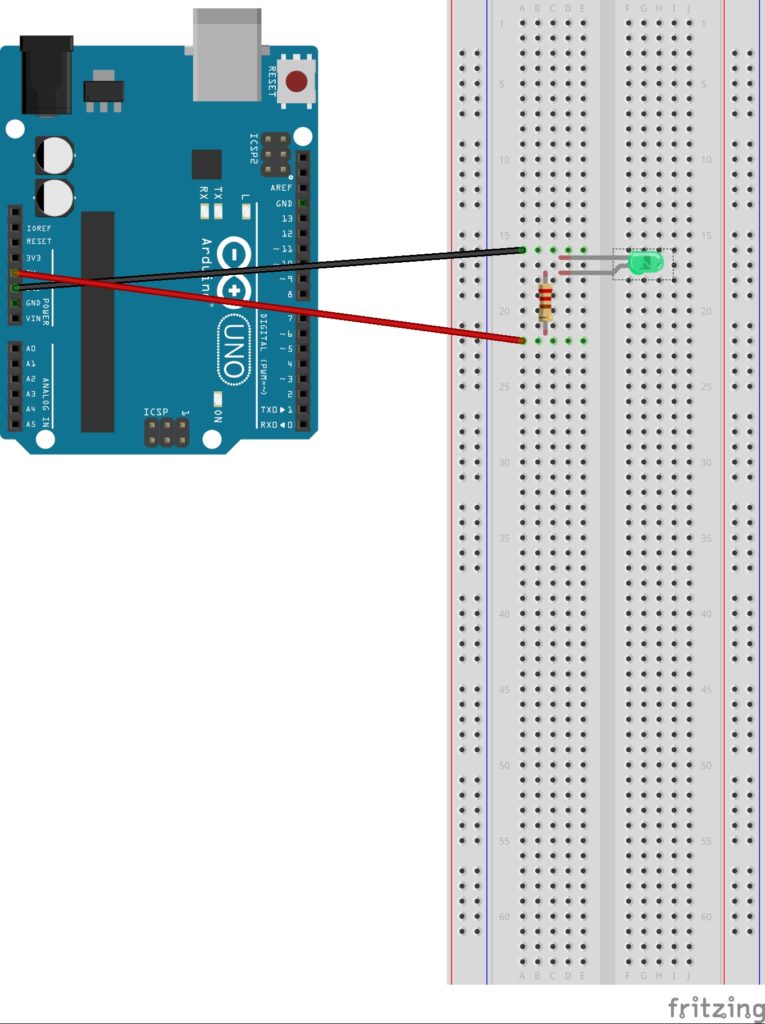


Figure 1: WIFI Controlled LED light  
From: https://www.elithecomputerguy.com/2019/06/arduino-uno-wifi-turn-led-on-and-off

**Controlling lights with the Arduino**

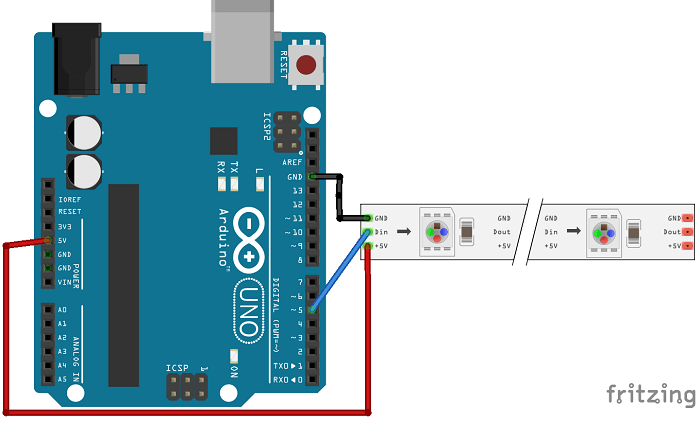


Figure 2: Connecting the LED strip to the Arduino, note each LED draws 60 mA of electricity and the Arduino via USB can only handle 400 mA and via barrel power connector 900 mA so an additional external power supple is needed for the 150 lights I’d like to control.   
From: https://www.hackster.io/a886a/double-clap-led-lamp-2b7dcd

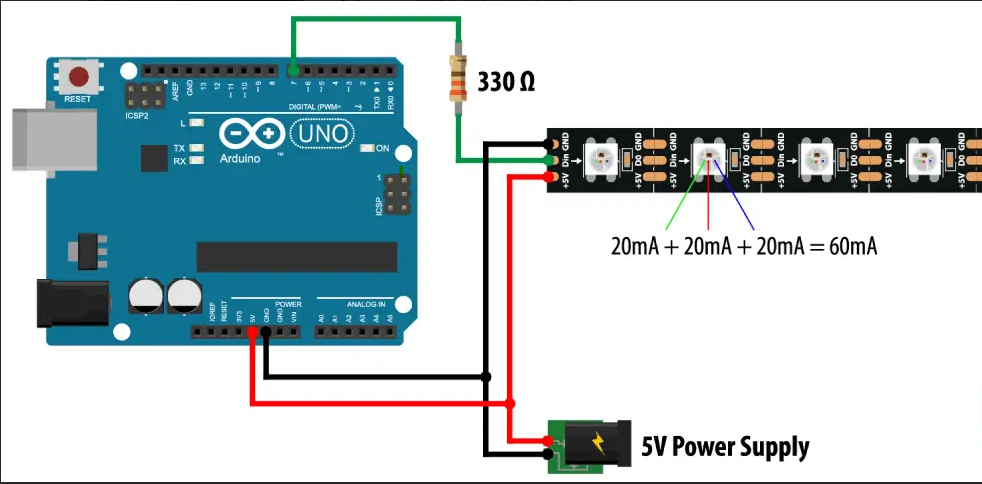
****

Figure 3: Additional power supply for LED strip  
From: <https://howtomechatronics.com/tutorials/arduino/how-to-control-ws2812b-individually-addressable-leds-using-arduino/>

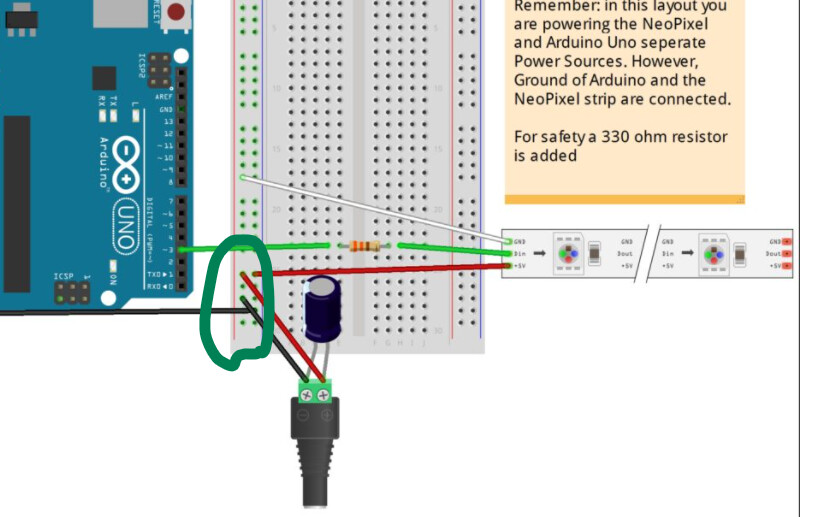


Figure 4:How to set up the external power supple for the LED strip  
From: https://forum.arduino.cc/t/using-capacitors-on-projects/876138/7

**Setting up the microphone + lights**

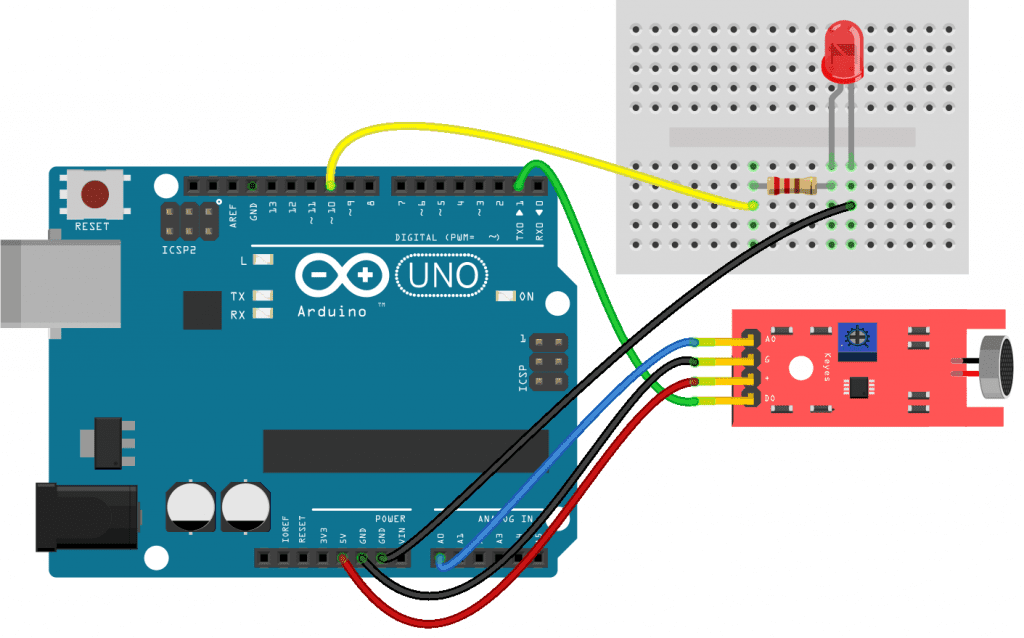


Figure 5: Connect LED light to Microphone  
From: https://www.etechnophiles.com/7-great-arduino-beginner-projects-with-code-in-2020/#diy-music-reactive-lights

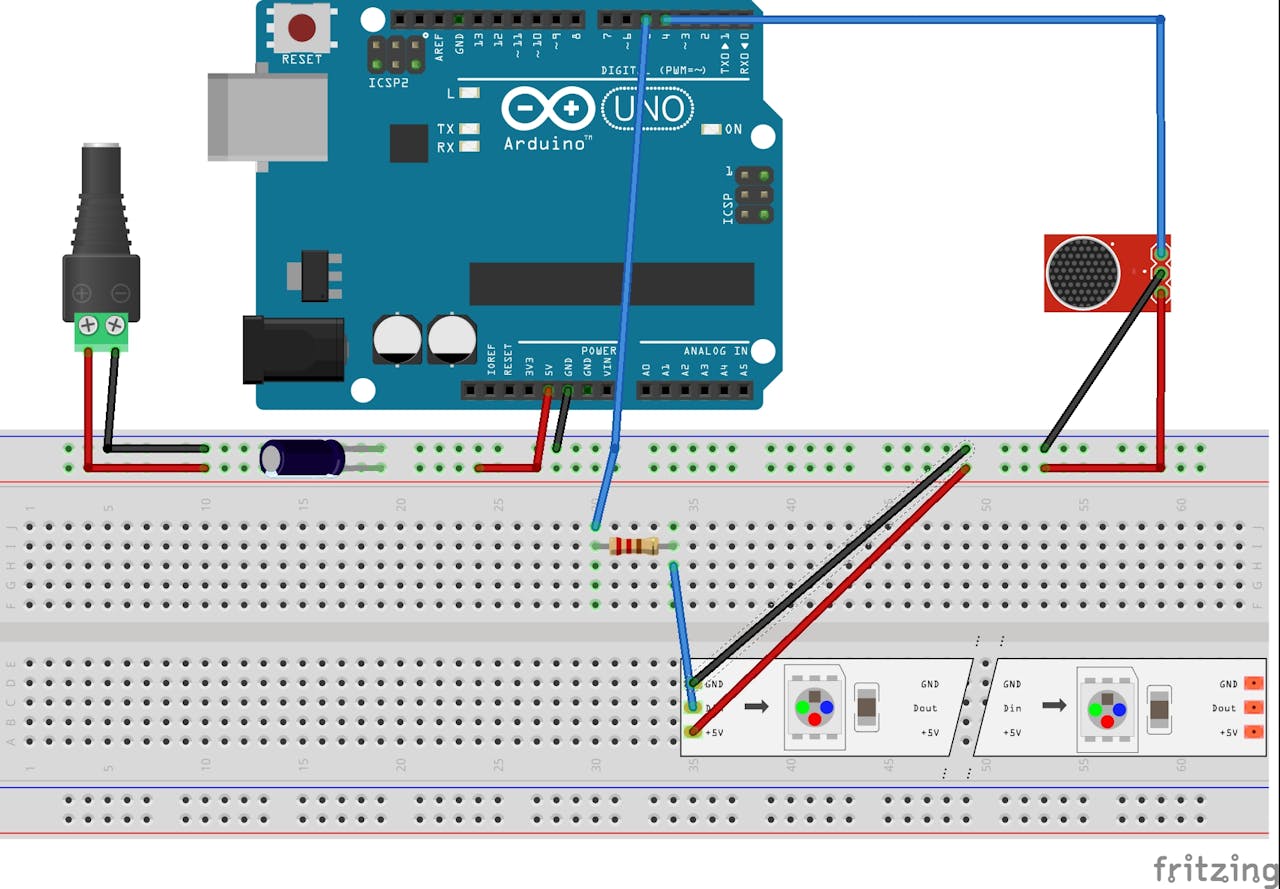


Figure 5: Connecting the LED strip to the microphone  
From: <https://www.hackster.io/a886a/double-clap-led-lamp-2b7dcd>

**Resources**:

**Clap switch**

<https://www.youtube.com/watch?v=uneaM07mwtg>

<https://forum.arduino.cc/t/a-clapper-switch-once-again/643723/2>

<https://forum.arduino.cc/t/led-strip-on-off-by-clapping/624148>

<https://www.hackster.io/a886a/double-clap-led-lamp-2b7dcd>

**Wifi Control**

<https://www.elithecomputerguy.com/2019/06/arduino-uno-wifi-turn-led-on-and-off/>

https://www.youtube.com/watch?v=wy-nkW\_H3io

**General info on using WS2812B LED Strip**

<https://randomnerdtutorials.com/guide-for-ws2812b-addressable-rgb-led-strip-with-arduino/>

<https://howtomechatronics.com/tutorials/arduino/how-to-control-ws2812b-individually-addressable-leds-using-arduino/>

**Not the correct LED Strip, but this was partially useful for connecting the LEDs** - <https://learn.adafruit.com/rgb-led-strips/usage>

**FastLED documentation -** <https://fastled.io/>

**Microphone use/setup**

<https://www.youtube.com/watch?v=Sj142ZOLEhM>

<https://www.circuitbasics.com/how-to-use-microphones-on-the-arduino/>

<https://www.etechnophiles.com/7-great-arduino-beginner-projects-with-code-in-2020/#diy-music-reactive-lights>

**Other potentially useful links  
Set timers (talks about individual addressing)**- <https://forum.arduino.cc/t/set-timer-for-led-strip-individually-for-each-led/928794>

**Using a capacitor/resistor to smooth out sound and power supply** - https://forum.arduino.cc/t/using-capacitors-on-projects/876138/7