**Goals**

For this project, I created an LED array based on multiple rows of LED strips, that would react to a sound detection module attached to an Arduino Uno. The goal of this project was to get more comfortable using LED strips and an input sensor to process data. I ended up using a sound sensor for my input, since I could have the LED strips respond and light up to the music. I also used my Ender 3 Pro 3D printer to design the structure to support all of the electronics. Most of the circuitry will be done on a breadboard, with the tabs of the LED strips being soldered to wires, then going to the breadboard circuit. I also planned to make the design of the box practical enough to make the project easier to integrate.

**Process**

Between all of the projects I worked on during my first Summer of college, I think that this one took the most planning, mostly from learning how to use my 3D printer. For example, since this was my first time using a CAD modeling software, that took longer than expected. I went through two different iterations of my CAD design through this project. I made the model much more condensed with the second version, so that it would print faster and be less prone to errors. I ended up importing CAD models of my Arduino, breadboard, and other parts to make sure the dimensions were accurate. I also had some trouble with using the sound sensor, since I could never get the analog input exactly where I wanted. The analog input would return a voltage between 0 and 1023 based on how loud the surroundings were. The audio source had to be in a very specific location, which heavily altered the analog data. Since this didn’t work out well, I ended up using the digital output to sense whether sound was being detected in an on or off state. After putting the right resistor into the sound sensor to decrease the gain, this worked out well for me. The last part of prep for this project was testing and using the LED strips. The FastLED Arduino library made this much easier than I thought, and I am really happy with how the LEDs display. Once the individual parts were already, it was time to connect everything.

While my 3D print was still going on, which took around 10 hours, I soldered the pin headers for the sound sensor. Once the print was done, I began to move parts onto the build. I used adhesive pads to place the Arduino and sound sensor on it, and used the premade breadboard adhesive to each piece securely. I connected the ground, 5V, and digital pins from the sound sensor to the Arduino first. This was because there were less wires than the LED strips, and that it would be easier to get these plugged in first. Then, I cut the LED strips to have 10 LEDs each and soldered 3 8-inch jumper wires to the ends of each strip. These signals would give the 5V, ground, and digital signal to the Arduino microcontroller, so that I could control the output with the FastLED library. I used craft wire to bundle up each trio of wires for the LED strips to keep the back of the design more organized. I threaded the wires through the opening on the bottom of the 3D print and tested the LED strip before sticking it on the front. This was so I could resolder them easier if needed afterwards. All of the strips ended up working without having to be soldered again, which was very nice. Then, I placed a capacitor on the breadboard to try and help reduce flickering from the LED strips. By this point, everything was hooked up correctly, and it was time to code these parts together.

As mentioned before, I used the FastLED Arduino library to help me interface with the LED Strips. For each pin used, I defined each pin number used at the top of my code. I also defined constants such as how many LED strips I used, and how many LEDs were on each strip. I had to specify the brightness, color order, and type of LED strip to set it up correctly with FastLED. This library had a function to add LEDs by defining the type, pin, and color order. I also had to use a function named setBrightness with the brightness constant that I defined. After this, I had to set a pinMode up for the digital pin of the sound sensor so the input could be read during the loop of the code. This was all done in the setup() function for Arduino, which runs once. In the loop, I first initialized an integer named gate that would read the input of my digital pin on the sound sensor. This would return a 1 or 0 depending on if sound was being picked up or not, respectively. Then, I added an integer called hue that would repeatedly increment between the values 0 through 254 to change the colors of the reacted LEDs. I also added a conditional check so I could modify how fast the change in hues was over time. Next, I added some if else statements to check when the gate value was 1 or 0. If sound was detected, the LED strips would increase the number of changing colors by 2 for every strip. If sound wasn't detected, the LED strips would decrease the changing colors by one. I added two functions at the bottom of my code to help me display the LEDs. setWhite() was made to make all of the LED strips white, acting as a background. Then, I made the other function called setRainbowWithWhite(). This would call setWhite() from within itself, then write over the existing LED’s by defining a set amount to make the rainbow colors. I had an array called rainbowNum to keep track of how many LEDs in each strip should be one of the changing colors at any given time. At the bottom of my main loop, I added the line FastLED.show(), which displayed every LED that I set up at the start of my code. This wrapped most of my code for the sound responsive LED strips, but I plan on making more patterns for this project in the future, since it allows for a lot to be done with the LED strips.

**Reflection**

This project helped me get more familiar with a variety of engineering skills. For one, this project enabled me to get my feet wet with CAD modeling and 3D printing. I also enjoyed being able to work with a simple input/output system by using the sound sensor to affect the LED Strips. This helped me think about effective ways to verify the different components of my project before putting it all together, which was something new to me at the time. The LED strips got me hooked on using them, so I am sure that I will find ways to stick them in other projects when I have a chance. it is easier to use strips that are connected instead of being split up, since you need less wires and soldering. I plan on adding more patterns to this project since I have a 9 X 10 set of LEDs at my disposal now. Thanks for reading!