ECE 1000 Honors Project: Festive Holiday Lights

Summer Morris
Electrical Engineering Department
Tennessee Tech University
Cookeville, Tennessee
sdmorris42@tntech.edu

***Abstract*—The Festive Holiday Lights Project is an honors project for ECE 1000 Explorations in ECE. The project includes WS2811 RGB LED string lights that have been programmed to demonstrate different patterns. The purpose of this project is to create light patterns that evoke feelings of festivity and holiday fun. The result of this project includes four main patterns and two similar patterns that can be used on the RGB LED string lights.**

Keywords—RGB LED's, Raspberry Pi Pico, Tinker cad, light patterns, 3D design

I. Introduction

Christmas is a wonderful time for creativity to shine. Consumers can purchase several different Christmas light colors and patterns online. The goal of this project is to use a Raspberry Pi Pico and addressable RGB LED string lights to design light patterns using various colors. This project is important because promotes creativity and can enhance the festivity and joy experienced by others. This project was designed by Summer Morris, who is an electrical engineering major. This project was designed for the honors portion in ECE 1000 Explorations in ECE. To begin this project, research was conducted in order to figure out how to set up and program RGB LED's.

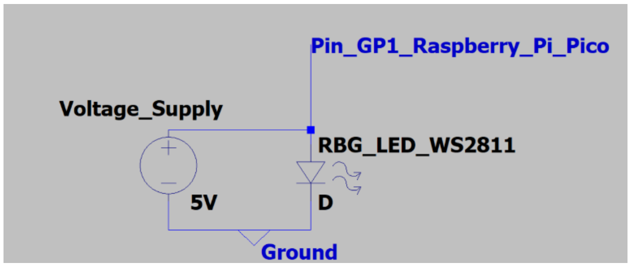
II. Background

To figure out how to wire the addressable RGB LED's to the Raspberry Pi Pico, a video made by TeCoEd on YouTube was used [5]. The project used an article written by Abhilekh Das [1] and an article from the Random Nerd Tutorials [2] website to code the RGB LED's to create a rainbow of colors that moves down the lights. An article written by Kattni Rembor [3] was used to change all the RGB LED's to one color and rotate between select colors. A video made by NerdCave [4] was used to randomly change the colors on the RGB LED's and create a spiral effect that will send one color that is a certain length down the lights. Help was also received from J.C. Williams, who is the teaching assistant for ECE 1000, to help code the lights to create certain patterns, such as using the percent sign to toggle the colors.

III. Project Description and formulation

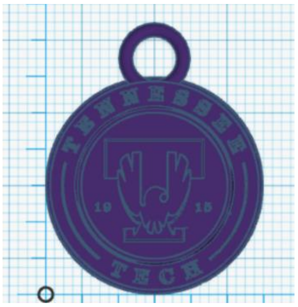
This project uses a Raspberry Pi Pico and an addressable RGB LED string lights (WS2811). The data pin on the WS2811 connects to GP1 on the Pico. The voltage pin and the ground pin on the RGB LED's connects to the 5 Volt pin and ground pin on the Raspberry Pi Pico respectively.

Fig. 1. Schematic of the RGB LED's.



Tinker Cad was used to design an ornament for the Christmas tree that the RGB LED's will go on. An image of the Tennessee Tech logo was imported and converted the file from a PNG file to STL file. By converting the image to a STL file, the image was able to be converted into a 3D design that could be 3D printed. The STL file was imported into Tinker Cad and was edited to create an ornament. The size of the logo was decreased to around 3 inches and a loop was added on the top of the logo that will allow the ornament to be hung on the Christmas Tree. The design was sent to J.C. Williams, the teaching assistant to ECE 1000, who 3D printed the ornament.

Fig. 2. 3D design of the ornament.



As mentioned previously in the report, many different patterns were designed for the addressable RGB LED string lights. The first pattern coded was a spiral or chasing effect. The RGB LED's will start purple, but the LED's will change to gold one by one starting from the bottom LED with a speed of 0.04 seconds. Once the gold reaches the top of the Christmas tree, purple will follow the gold in the same manner. The length of the gold being shifted is set to 49, but it can be increased or decreased using the "length" variable. The colors are able to shift forward by setting the variable "i," which represents a LED, into the variable "Pixel_Lights" which has been defined in the NeoPixel library, equal to the previous LED's state. Obtaining the previous state is achieved by subtracting the variable "i" by 1.

The next pattern will alternate the colors, purple and gold, along the string lights. The LED's will toggle between purple and gold every second. This pattern is achieved by defining the variable "i" as the LED number and dividing "i" by 2 with the percent sign. If the LED number is even, the color will start purple, but if the LED number is odd, the color will start gold. Similar to the first pattern, the interval "i" is shifted by subtracting "i" by 1. This in turn will create a toggling effect. By removing the section of code that subtracts "i" by 1, the code will just create an non-shifting pattern that alternates the LED's colors between purple and gold.

The next pattern created will change each LED to a random specified color every 0.5 seconds. This pattern was achieved by importing the random effect from the library. By creating a variable, "Colors" that holds the variables for gold and white, the random effect can set the RGB LED to either gold or white when 1. If the random effect is set to 0, then the RGB LED will turn purple. This effect is accomplished with an "if" and "else" statement.

The last pattern created will slightly change the hue in each RGB LED in the string lights to create a rainbow of colors. The colors will then shift down the string lights, creating a moving rainbow effect. This effect was accomplished by defining the position of the LED's and setting certain positions equal to red, green, and blue. By overlapping the numbers representing the positions slightly, through the use of less than and equal to symbols, each of the RGB LED's get varying amounts of red, green, and blue. This allows the RGB LED's to create rainbow effect. It is important to specify the number of bulbs on the string lights, or the LED's will not change color. As the number of bulbs on the string lights increases, the variations in color also increases. Since the location of the colors have been specified, by subtracting 1 from the location, the colors are able to shift down the string lights. This movement creates a shifting rainbow effect, which is perfect for the holiday season. The shifting of colors occurs every 50 milliseconds.

IV. Discussion and results

Many patterns were created for the RGB LED string lights. The first pattern turned the lights purple and sent a gold color up the lights, creating a spiral effect. The next pattern toggled the colors between purple and gold in an alternating pattern. By removing the piece of code that shifted the colors, the lights will display a stationary pattern that alternates between purple and gold. Another pattern will turn all RGB LED's to gold and will alternate between gold, purple, and white every second. Another pattern created will change randomly change the colors between gold, purple, and white every 0.5 seconds. The last code will create a rainbow effect that shifts the colors every 50 milliseconds. The WS2811 RGB LED string lights were used to decorate a small, white Christmas tree that will be displayed in the ECE department office at Tennessee Tech.

If more time was allocated for this project, the wires would be soldered onto the Raspberry Pi Pico for more permanent and stronger connections. A box would be 3D designed that can safely hold the Raspberry Pi Pico. Coding and testing the RGB LED's was the most enjoyable part of this project. It was extremely fascinating to watch the patterns on the RGB LED's.

V. Conclusion

The purpose of this project is to create festive and unique patterns with RGB LED's that can be enjoyed by Tech students during the holiday season. Some skills that have been acquired following the conclusion of this project includes working with Raspberry Pi Picos, coding with MicroPython, and coding RGB LED's. Other skills that this project has increased includes presentations skills, writing in IEEE format, and uploading projects onto GitHub. The results of this project includes many festive light patterns that envelops the spirit of Tennessee during the holiday season.

References

[1] A. Das, "NeoPixel WS2812B RGB LED with Raspberry Pi Pico (MicroPython)," *Electrocredible*, May 21, 2023. https://electrocredible.com/neopixel-micropython-raspberry-pi-pico-ws2812b/ (accessed Apr. 21, 2024).

[2] "MicroPython: WS2812B Addressable RGB LEDs with ESP32 and ESP8266 | Random Nerd Tutorials," *Random Nerd Tutorials*, 2024. https://randomnerdtutorials.com/micropython-ws2812b-addressable-rgb-leds-neopixel-esp32-esp8266/ (accessed Apr. 21, 2024).

[3] K. Rembor, "Getting Started with Raspberry Pi Pico and CircuitPython," *Adafruit*, Apr. 22, 2024. https://learn.adafruit.com/getting-started-with-raspberry-pi-pico-circuitpython/neopixel-leds (accessed Apr. 23, 2024).

[4] NerdCave, "Raspberry Pi Pico Tutorial : NeoPixels / WS2812B LED's," *www.youtube.com*, Mar. 14, 2022. https://www.youtube.com/watch?v=WpaXMcmwyeU (accessed Apr. 22, 2024).

[5] TeCoEd, "How to wire up WS2811 RGB LEDs to the Raspberry Pi," *www.youtube.com*, Feb. 14, 2020. https://www.youtube.com/watch?v=KJupt2LIjp4 (accessed Apr. 23, 2024).