Creating a Multiple Traffic Light Controller with a Cross Walk and an Emergency Buzzer with secured IoT Control via Web

CEIS 114 FINAL PROJECT COMPONENT — OPTION 1

INTRODUCTION

- This presentation documents my learning journey for IoT (internet of things)

- We were tasked with creating a a simulation of traffic lights at an intersection

- These traffic lights are required to do the following:

  - Not all cross traffic to be both green at once

  - Allow pedestrians to request a red light / walk scenario

  - Force both directions to have a red light flashing when an emergency button was pressed to allow for emergency vehicles to bypass traffic
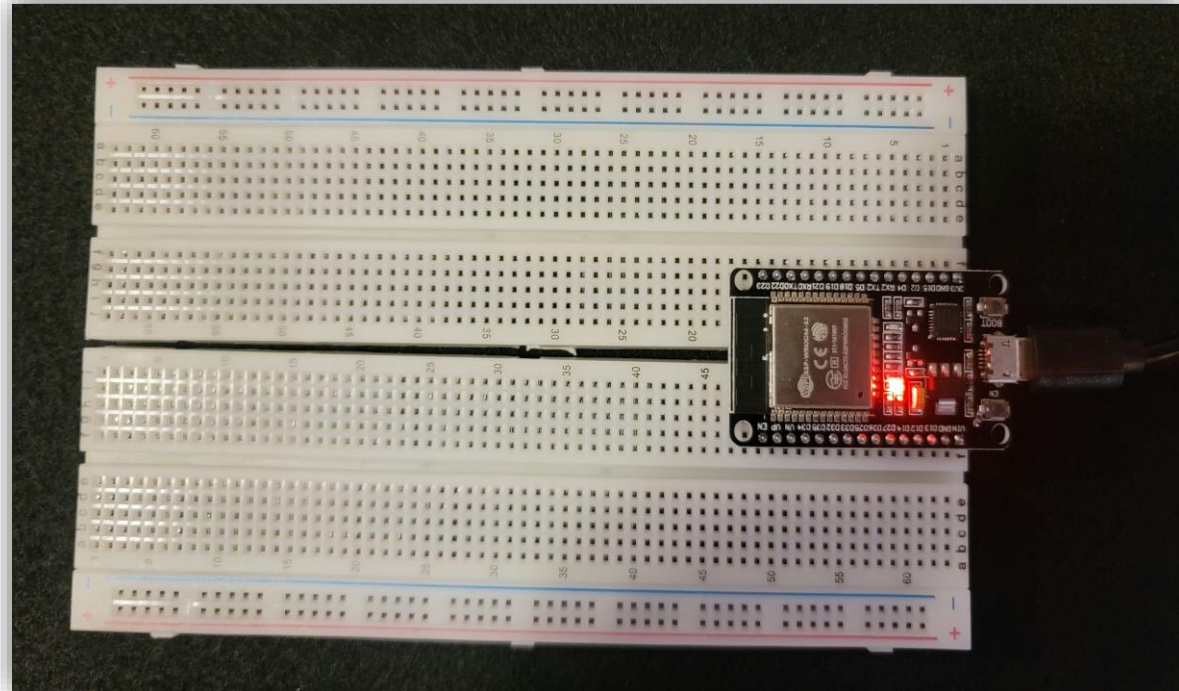
WEEK 2

INVENTORY

- ESP 32 Board
- Colored LEDs: Red, Yellow, Green, and Blue
- Wires
- Breadboard(s)
- LCD Unit with I2C Adapter
- Active Buzzer
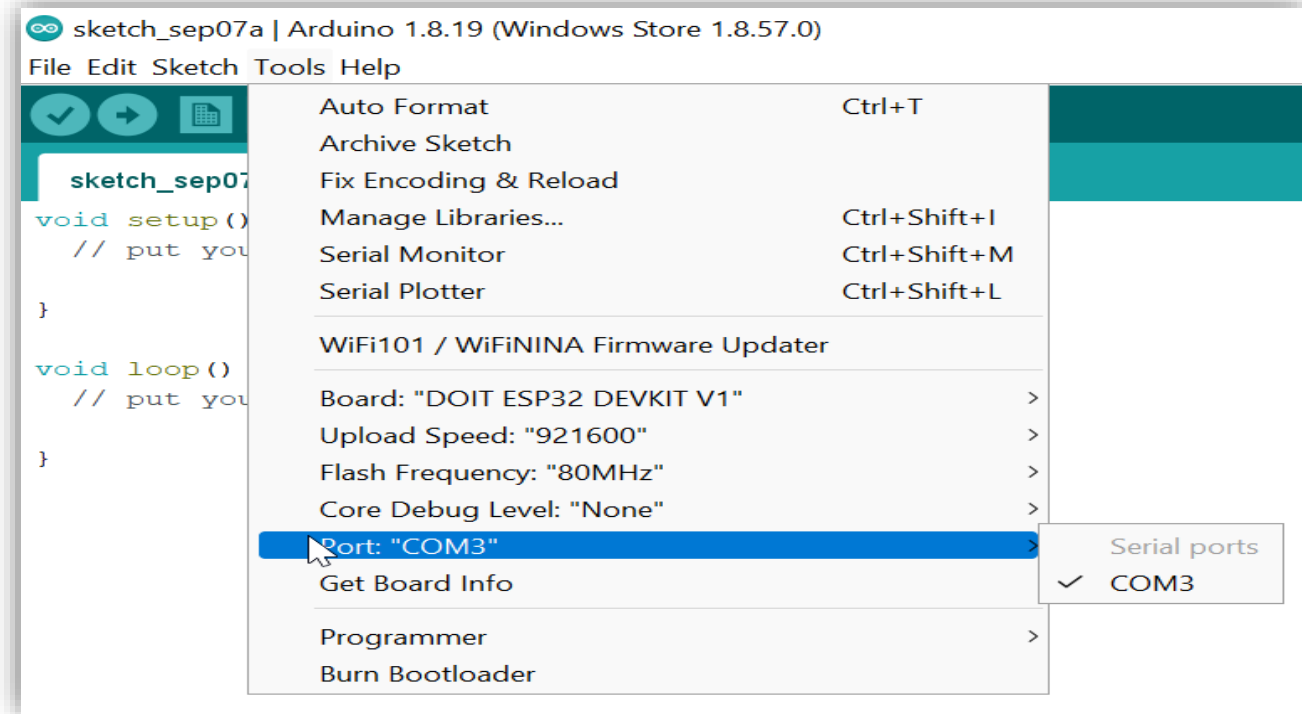- Mini Router
- Push Button(s)
- PIR Motion Sensor
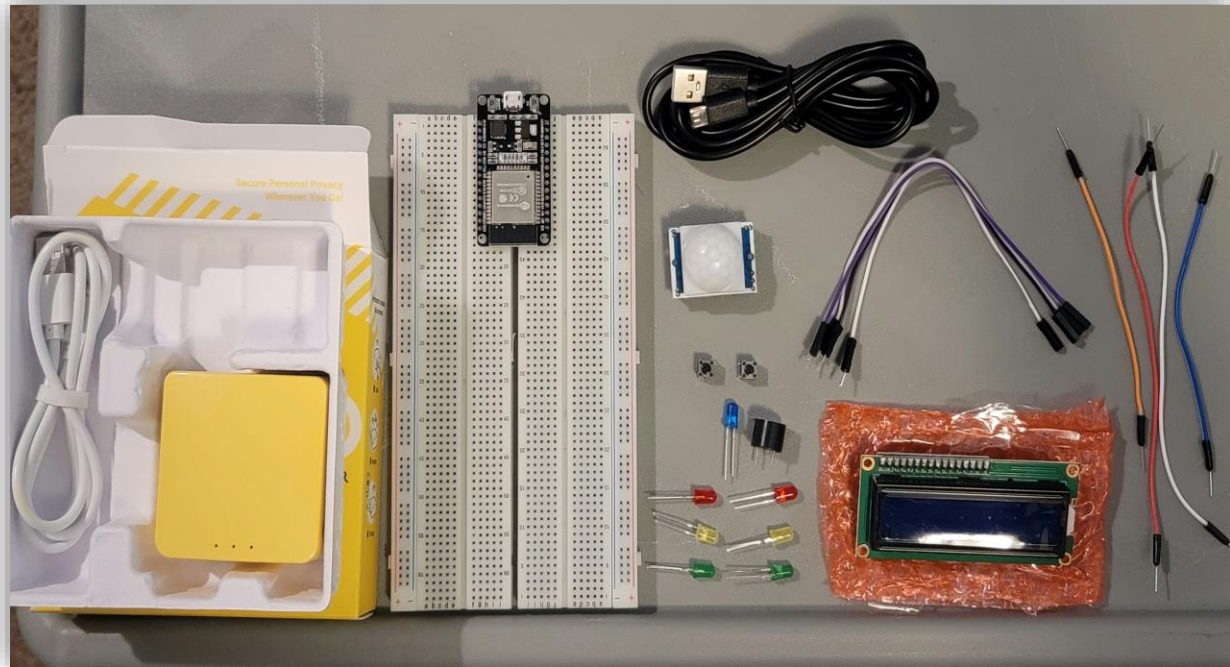
ESP32
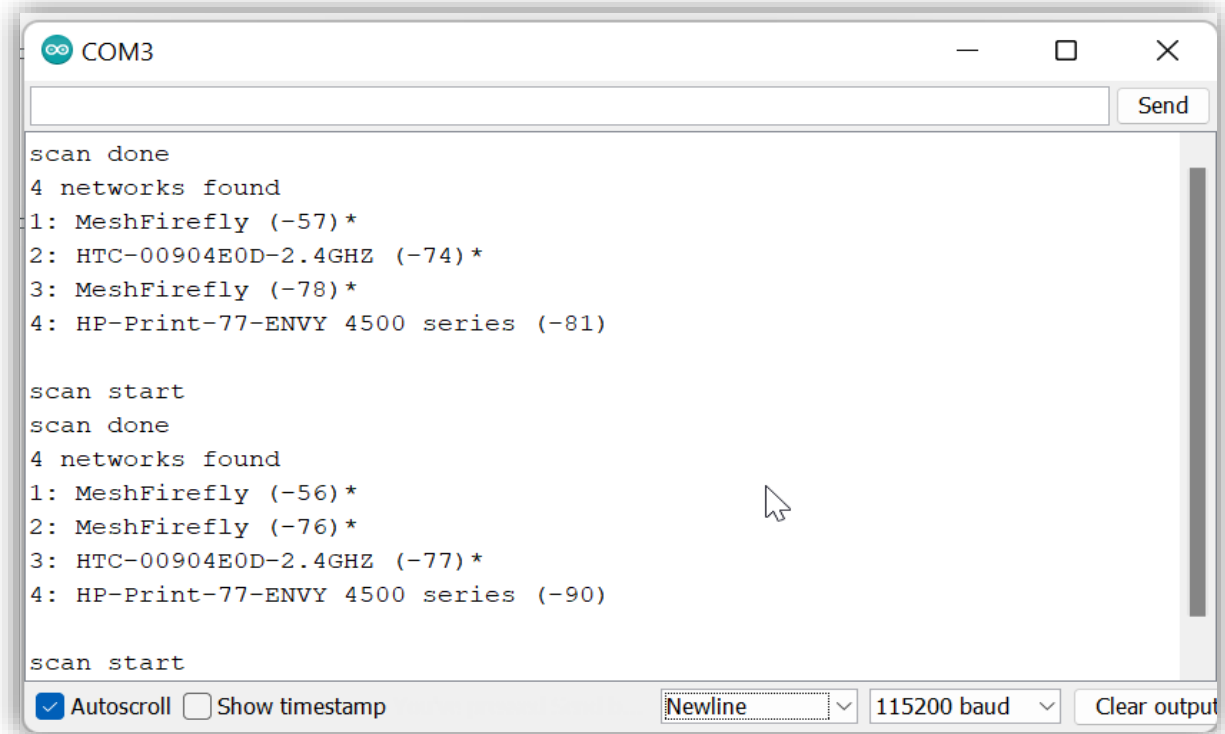
- Microcontroller mounted and powered ON

INSTALLATION OF ARDUINO IDE

- Screenshot of Arduino IDE with **Port** selected from Tools menu

ESP32 WI-FI SCAN

- Screenshot of Serial Monitor in Arduino IDE showing the available networks


```
scan done
4 networks found
1: MeshFirefly (-57)*
2: HTC-00904E0D-2.4GHZ (-74)*
3: MeshFirefly (-78)*
4: HP-Print-77-ENVY 4500 series (-81)

scan start
scan done
4 networks found
1: MeshFirefly (-56)*
2: MeshFirefly (-76)*
3: HTC-00904E0D-2.4GHZ (-77)*
4: HP-Print-77-ENVY 4500 series (-90)

scan start
```
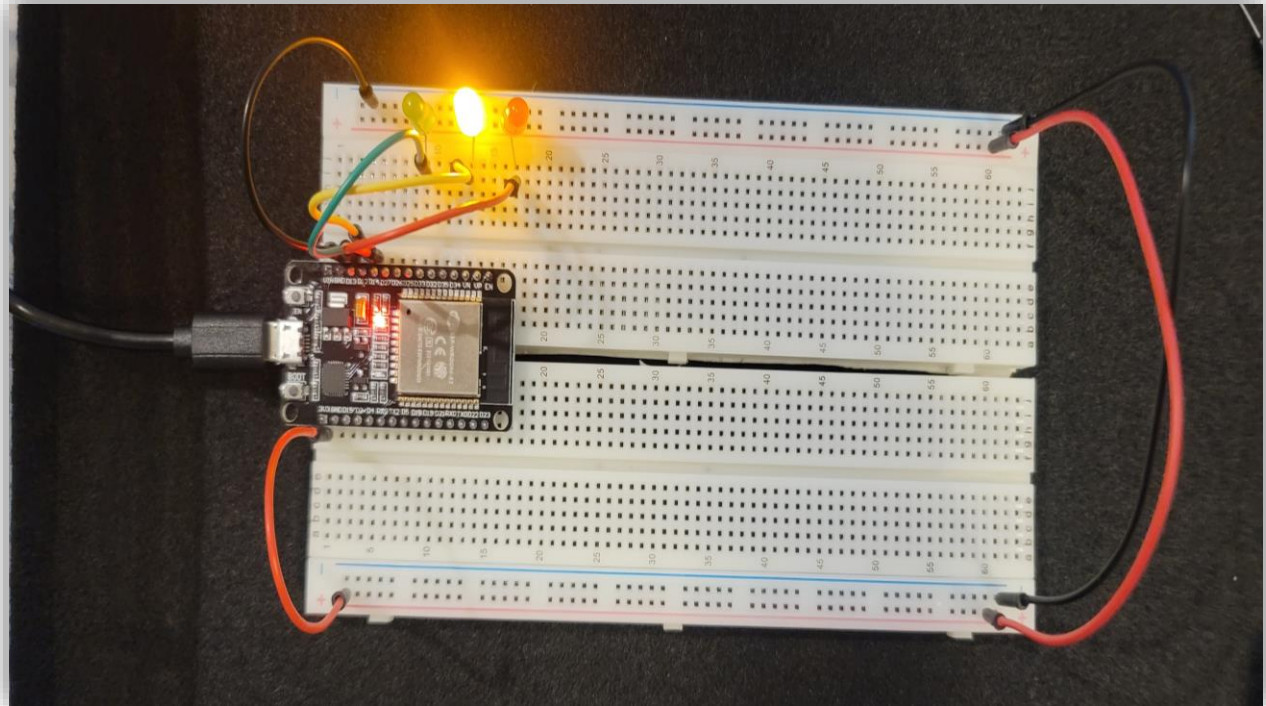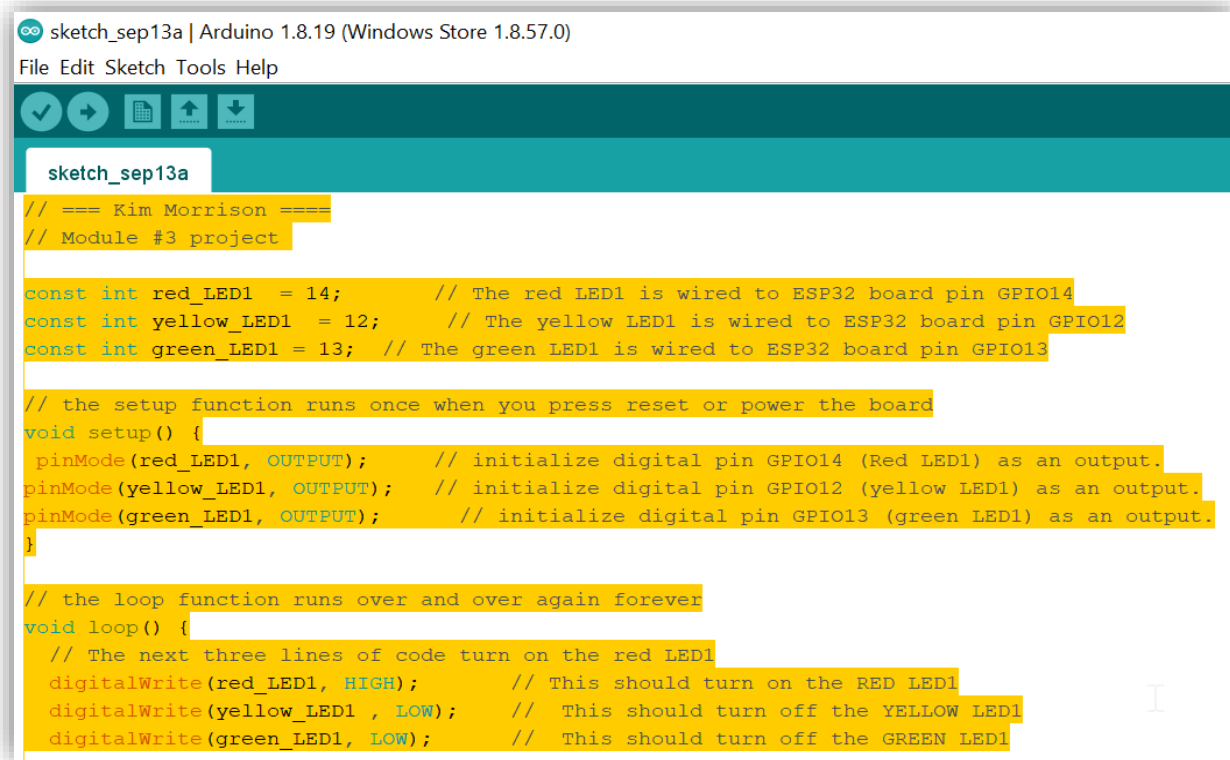
WEEK 3

PICTURE OF CIRCUIT WITH WORKING LEDS

- ESP 32 Board

- Colored LEDs: Red, Yellow and Green

- Wires

- Breadboard

SCREENSHOT OF CODE IN ARDUINO IDE

- Screenshot of code in Arduino IDE showing your name in the comment

sketch_sep13a | Arduino 1.8.19 (Windows Store 1.8.57.0)

File Edit Sketch Tools Help

sketch_sep13a

```
// === Kim Morrison ====
// Module #3 project

const int red_LED1  = 14;      // The red LED1 is wired to ESP32 board pin GPIO14
const int yellow_LED1  = 12;      // The yellow LED1 is wired to ESP32 board pin GPIO12
const int green_LED1 = 13;   // The green LED1 is wired to ESP32 board pin GPIO13

// the setup function runs once when you press reset or power the board
void setup() {
  pinMode(red_LED1, OUTPUT);      // initialize digital pin GPIO14 (Red LED1) as an output.
  pinMode(yellow_LED1, OUTPUT);    // initialize digital pin GPIO12 (yellow LED1) as an output.
  pinMode(green_LED1, OUTPUT);       // initialize digital pin GPIO13 (green LED1) as an output.
}

// the loop function runs over and over again forever
void loop() {
  // The next three lines of code turn on the red LED1
  digitalWrite(red_LED1, HIGH);       // This should turn on the RED LED1
  digitalWrite(yellow_LED1 , LOW);     //  This should turn off the YELLOW LED1
  digitalWrite(green_LED1, LOW);       //  This should turn off the GREEN LED1
```
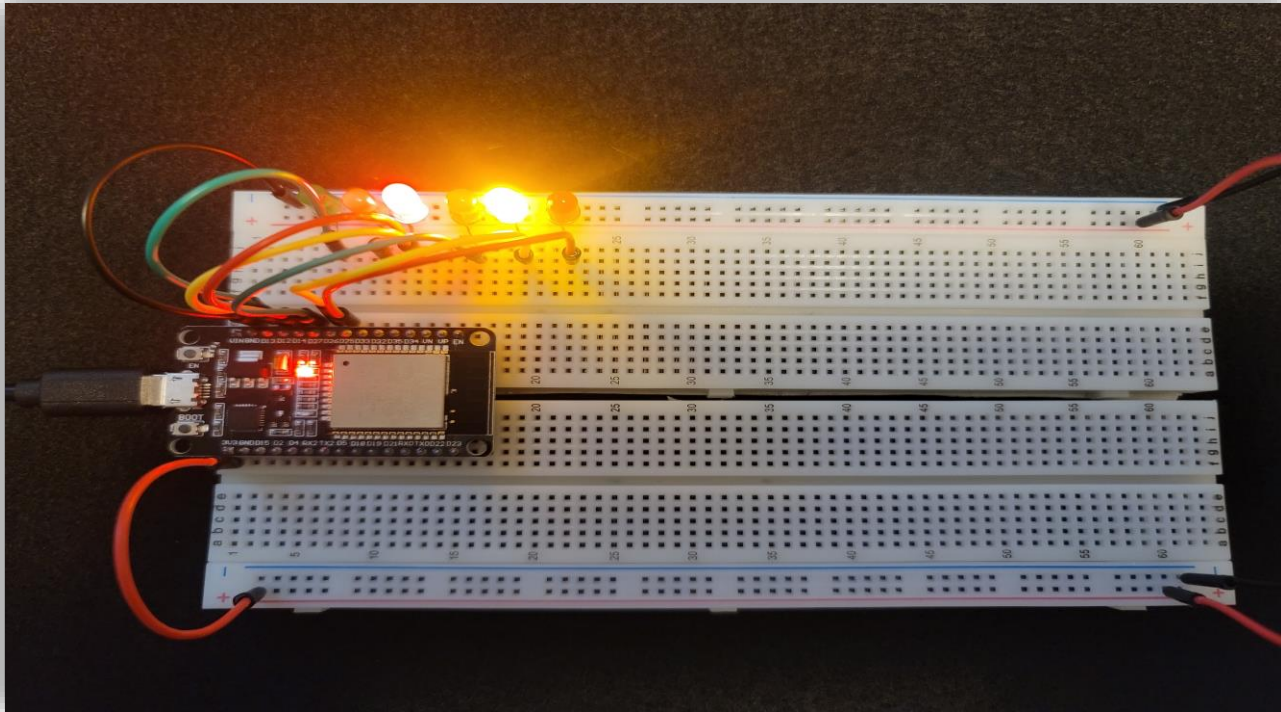
WEEK 4

PICTURE OF CIRCUIT WITH WORKING LEDS

- ESP 32 Board

- Colored LEDs: Red, Yellow and Green (two sets)

- Wires

- Breadboard

SCREENSHOT OF CODE IN ARDUINO IDE

- Screenshot of code in Arduino IDE showing **your name in the comment**


```
sketch_sep13a | Arduino 1.8.19 (Windows Store 1.8.57.0)
File Edit Sketch Tools Help

  sketch_sep13a §
// === Kim Morrison ====
// Module #4 project

// Define some labels
const int red_LED1  = 14;   // The red LED1 is wired to ESP32 board pin GPIO14
const int yellow_LED1  =12;   // The yellow LED1 is wired to ESP32 board pin GPIO12
const int green_LED1 = 13; // The green LED1 is wired to ESP32 board pin GPIO13
const int red_LED2  = 25;   // The red LED2 is wired to Mega board pin GPIO25
const int yellow_LED2  = 26;   // The yellow LED2 is wired to Mega board pin GPIO 26
const int green_LED2 = 27; // The green LED2 is wired to Mega board pin GPIO 27

// the setup function runs once when you press reset or power the board
void setup() {
 pinMode(red_LED1, OUTPUT);  // initialize digital pin GPIO14 (Red LED1) as an output.
 pinMode(yellow_LED1, OUTPUT);   // initialize digital pin GPIO12 (yellow LED1) as an output.
 pinMode(green_LED1, OUTPUT);    // initialize digital pin GPIO13 (green LED1) as an output.
 pinMode(red_LED2, OUTPUT);    // initialize digital pin GPIO25(Red LED2) as an output.
 pinMode(yellow_LED2, OUTPUT); // initialize digital pin GPIO26 (yellow LED2) as an output.
 pinMode(green_LED2, OUTPUT); // initialize digital pin GPIO27 (green LED2) as an output.
}

// the loop function runs over and over again forever
void loop() {
  // The next three lines of code turn on the red LED1
  digitalWrite(red_LED1, HIGH);      // This should turn on the RED LED1
  digitalWrite(yellow_LED1 , LOW);         //  This should turn off the YELLOW LED1
```
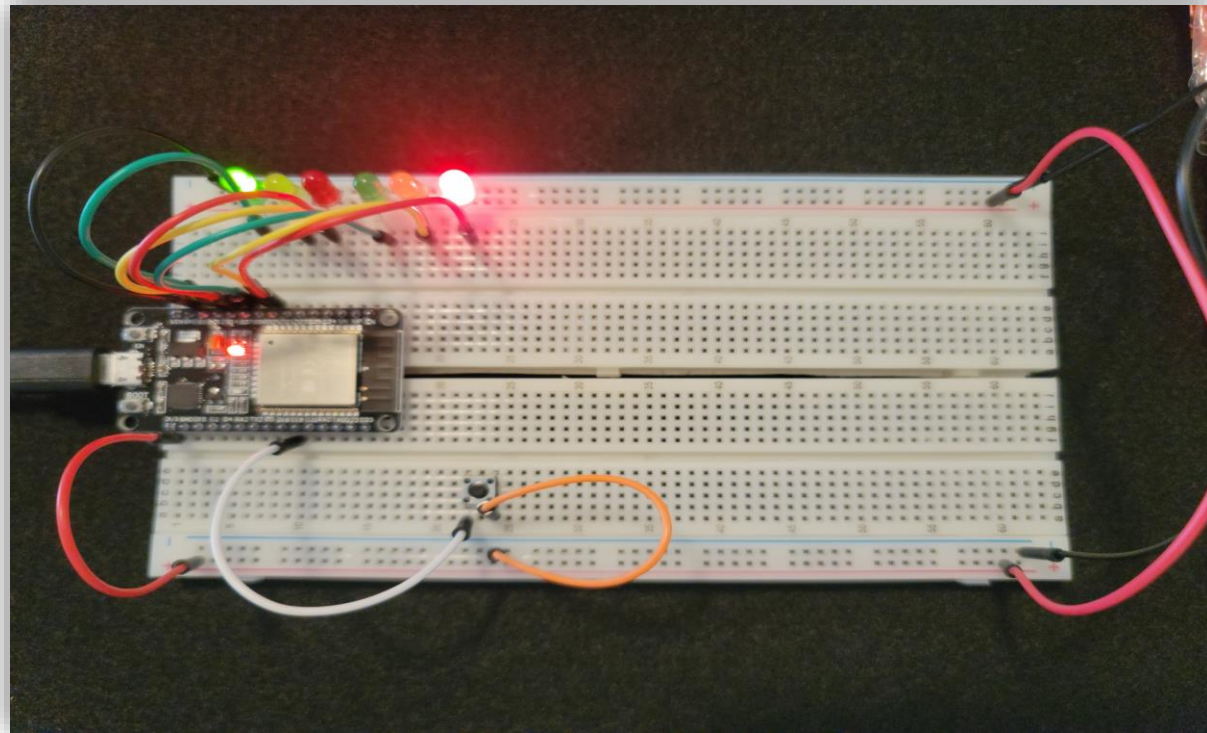
WEEK 5

PICTURE OF CIRCUIT WITH WORKING LEDS

- ESP 32 Board

- Colored LEDs: Red, Yellow and Green (two sets)

- Push Button

- Wires

- Breadboard

SCREENSHOT OF CODE IN ARDUINO IDE

- Screenshot of code in Arduino IDE showing your name in the comment

SCREENSHOT OF SERIAL MONITOR IN ARDUINO IDE

- Screenshot of output in Serial Monitor

WEEK 6

PICTURE OF CIRCUIT WITH WORKING LEDS AND LCD DISPLAY

- ESP 32 Board

- Colored LEDs: Red, Yellow and Green (two sets)

- Push Button

- LCD Unit with Message Display

- Wires

- Breadboard

SCREENSHOT OF CODE IN ARDUINO IDE

- Screenshot of code in Arduino IDE showing your name in the comment

SCREENSHOT OF SERIAL MONITOR IN ARDUINO IDE

- Screenshot of output in Serial Monitor


```
COM3

17:15:44.614 ->       == Do Not Walk ==
17:15:54.609 ->    Count =   10   == Walk ==
17:15:55.645 ->    Count =   9    == Walk ==
17:15:56.684 ->    Count =   8    == Walk ==
17:15:57.767 ->    Count =   7    == Walk ==
17:15:58.809 ->    Count =   6    == Walk ==
17:15:59.846 ->    Count =   5    == Walk ==
17:16:00.931 ->    Count =   4    == Walk ==
17:16:01.968 ->    Count =   3    == Walk ==
17:16:03.053 ->    Count =   2    == Walk ==
17:16:04.092 ->    Count =   1    == Walk ==
17:16:05.130 ->    Count =   0    == Walk ==
17:16:06.262 ->       == Do Not Walk ==
```

WEEK 7

FEEDBACK GIVEN TO TEAMMATES

Isaiah Nixon
Wednesday

[CEIS114 Project Template Module 6 Complete .pptx](#) ⤓

Hello, class and group. I have decided to do option 2 for my Final Course Project. My process for building these final powerpoint deliverables is as follows: I put together the powerpoint to reflect my progress on the project up to this point, then put together the final portion of the device, and finally I will put together the last portion of the powerpoint to show my finished project and discuss the challenges I faced along the way.

↩ Reply   👍

Kimberly Morrison (*She/Her*)
Wednesday

Have you thought about the design of your power point? Adding some color will make your pictures pop and grab the reader's attention.

I think your picture clarity is wonderful.

↩ Reply   👍

FEEDBACK GIVEN TO TEAMMATES

Noah Locklear
Wednesday

[Presentation 17 (1).pdf](#) ↓

This was my module 6 project. I plan on taking all the sides from all the projects and bringing them into one on a Wix website. I will talk about what I have learned and the possible projects in the future that I can use this information.

↩ Reply   👍

Kimberly Morrison (*She/Her*)
Thursday

I am very jealous of your LCD picture. Mine needs to be redone.

I think your plan is a good one. One question I have for you, what do you think your biggest challenge was while executing this project?

FEEDBACK REFLECTIONS

Andrea Williams
Oct 11, 2022

Kimberly,

I took a look at your ppt and it looks great. I was amazed at how nicely your jumper wires were placed on your breadboard in each picture you submitted. It looks very professional and the setup of your ppt itself definitely works. I still have to do my final circuit and make my ppt but yours is inspirational. Great job.

Reply   (1 like)

FEEDBACK REFLECTIONS

Robert Vargas
Sunday

Hello Kimberly,

 I taken a look at the power point that you gave and it very good so far. Just the thought of being a rough draft, I think you are heading in the right direction for your Final Prestation Project. Many of the pictures provided is very clear and readable. Only thing which is my own opinion is that the challenges slide will be either second or third to the last slide but that just myself. More than that I think the rough draft is very good and neatly done.
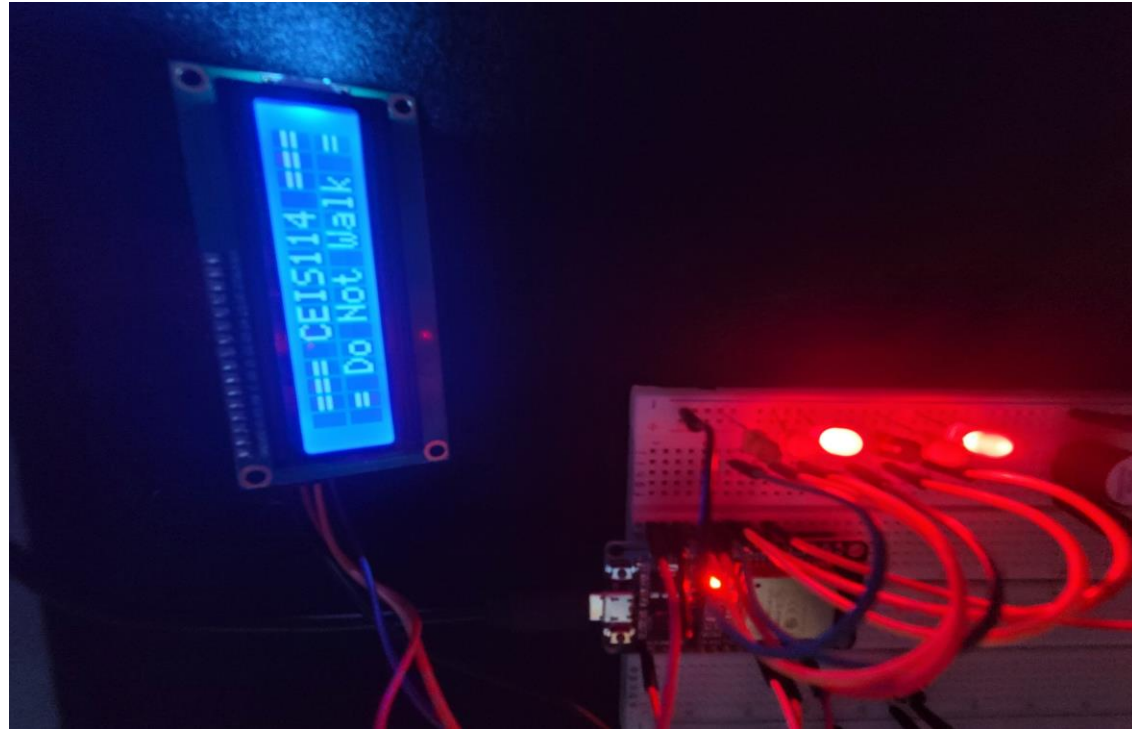
Robert Vargas
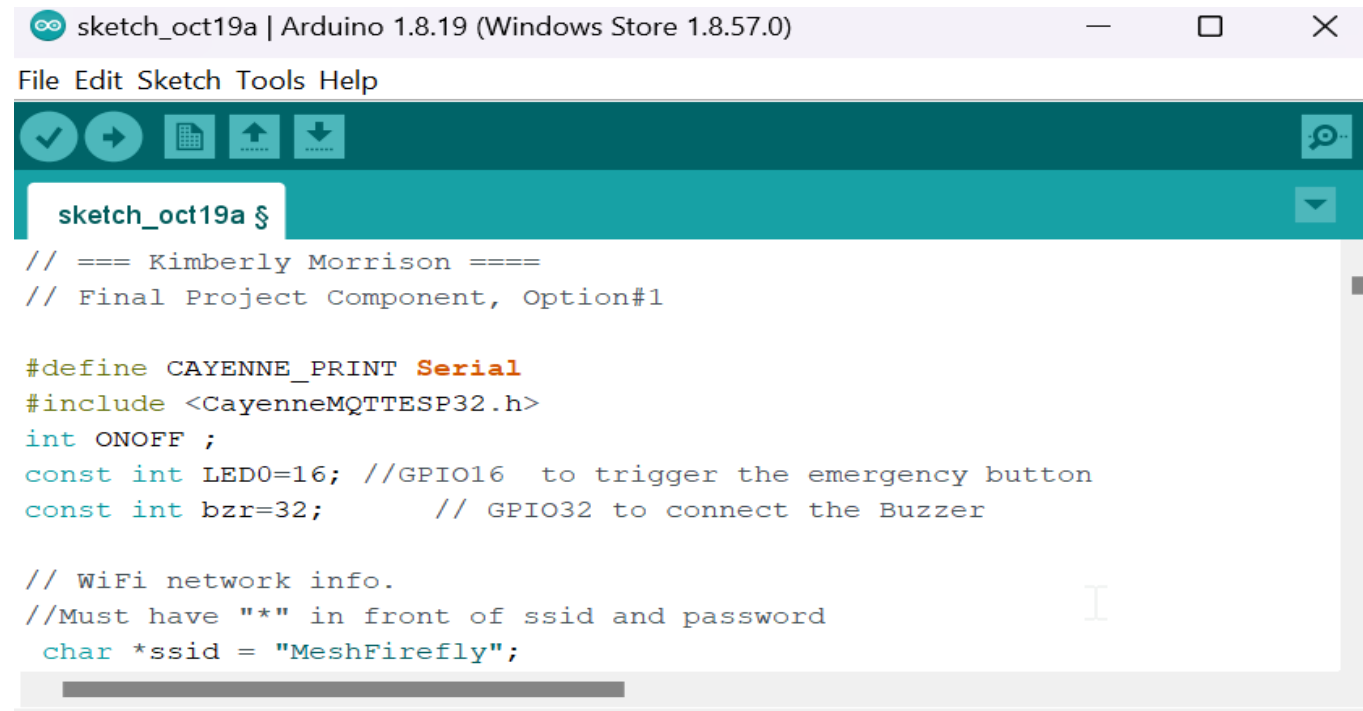
WEEK 8 – FINAL COURSE PROJECT

PICTURE OF CIRCUIT WITH WORKING LEDS AND LCD DISPLAY

- ESP 32 Board

- Colored LEDs: Red, Yellow and Green (two sets)

- One Blue LED – Emergency Light

- Push Button

- LCD Unit

- Buzzer

- Wires

- Breadboard

SCREENSHOT OF CODE IN ARDUINO IDE

Screenshot of code in Arduino IDE showing your name in the comment

sketch_oct19a | Arduino 1.8.19 (Windows Store 1.8.57.0)

File Edit Sketch Tools Help

sketch_oct19a §

```
// === Kimberly Morrison ====
// Final Project Component, Option#1

#define CAYENNE_PRINT Serial
#include <CayenneMQTTESP32.h>
int ONOFF ;
const int LED0=16; //GPIO16  to trigger the emergency button
const int bzr=32;        // GPIO32 to connect the Buzzer

// WiFi network info.
//Must have "*" in front of ssid and password
 char *ssid = "MeshFirefly";
```

SCREENSHOT OF SERIAL MONITOR IN ARDUINO IDE

Screenshot of output in Serial Monitor

SCREENSHOT OF SERIAL MONITOR IN ARDUINO IDE

Screenshot of output in Serial Monitor

CHALLENGES

- One of my LEDs refused to light no matter how it was wired, and I had to troubleshoot the issue

- Having to debug code that I did not write to determine why the application failed to upload to the board

- Remembering to hold the reset button on the ESP32 board until the Arduino IDE connected to the board for uploads

SKILLS LEARNED

- Understanding how to troubleshoot small components

- Understanding and being able to identify icons and what they mean for gates on circuit boards

- Better recall of how to determine binary numbers

CONCLUSION

- It was very interesting to see how small components for smart devices have become of the last decade.

- There were lots of ideas shared on how students could continue their engineering journey with this class and the plethora of information on the internet