**Course Project – Module 5**

**College of Engineering and Information Sciences**

**Course Number: CEIS114**

# **Multiple Traffic Light Controller with Cross Walk**

# PARTS LIST

* PC running Arduino IDE
* ESP32 Board
* Two sets of Colored LEDs: Red, Yellow and Green
* Button
* Wires
* Breadboard

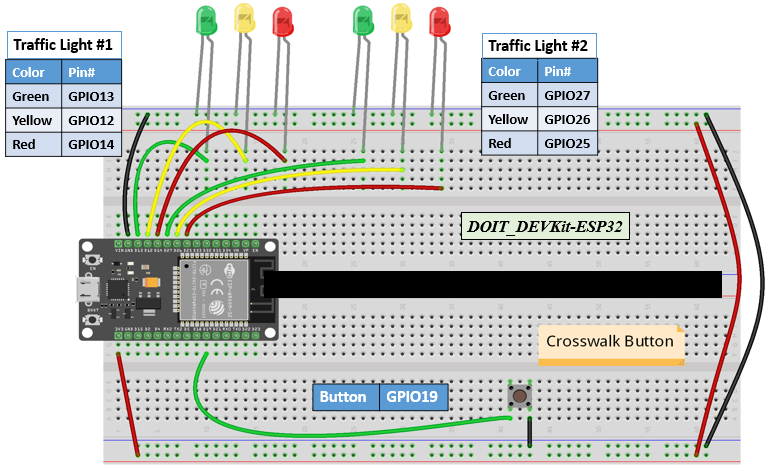
Deliverables

* Complete the Course Project PowerPoint Deliverable
* Include a picture of your circuit
* Screenshot of Arduino IDE code from your computer
* Screenshot of output in Serial Monitor from your computer

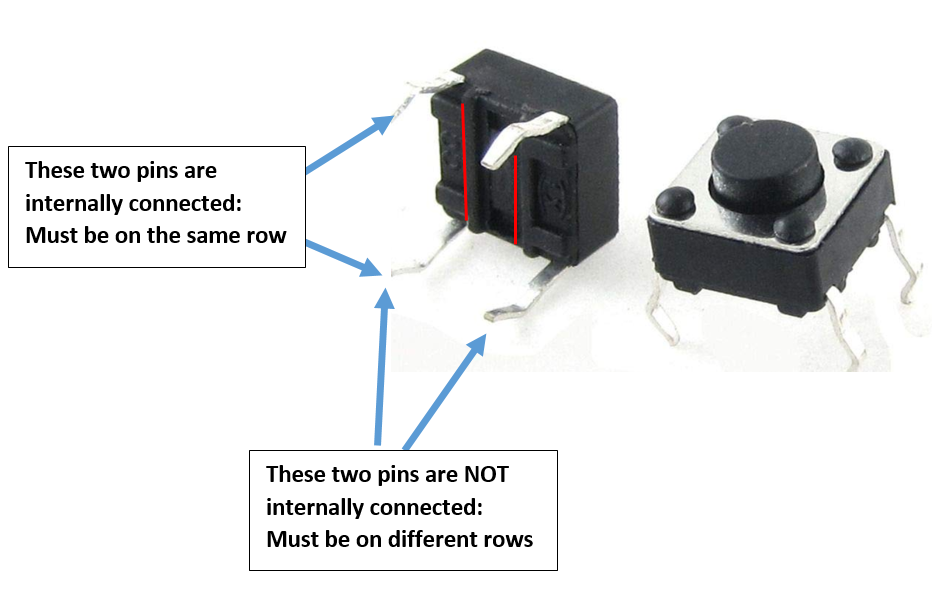
# PROCEDURE

1. Assemble the circuit as shown in Figure 1
2. Connect all the wires as shown in Figure 1. Make sure to connect the Red, Yellow, and Green wires to the appropriate pins in the ESP32 Board.
3. Open the Arduino IDE and load the code shown in Figure 2.
4. Connect the ESP32 Board to your computer and check if the board is recognized. Make sure that the port and the board are correctly selected. Also, make sure the upload speed is set to 115200 in the Tools menu.
5. Run the code and observe the output in the serial monitor.
6. Press the wire for activate the cross walk and observe the output.

**Figure 1: Multiple Traffic Lights with Crosswalk Button**



***For the button interfacing, please refer to the figure below:***



**Figure 2: Code for Two Traffic Lights and Crosswalk Button**

***// === Replace this text with your Name ====***

***// Module #5 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*

*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*

*int Xw\_value;*

*const int Xw\_button = 19; //Cross Walk button*

*// the setup function runs once when you press reset or power the board*

*void setup() {*

*pinMode(Xw\_button, INPUT\_PULLUP); // 0=pressed, 1 = unpressed button*

*Serial.begin(115200);*

*pinMode(red\_LED1, OUTPUT); // initialize digital pin 14 (Red LED1) as an output.*

*pinMode(yellow\_LED1, OUTPUT); // initialize digital pin 12 (yellow LED1) as an output.*

*pinMode(green\_LED1, OUTPUT); // initialize digital pin 13 (green LED1) as an output.*

*pinMode(red\_LED2, OUTPUT); // initialize digital pin 25(Red LED2) as an output.*

*pinMode(yellow\_LED2, OUTPUT); // initialize digital pin 26 (yellow LED2) as an output.*

*pinMode(green\_LED2, OUTPUT); // initialize digital pin 27 (green LED2) as an output.*

*}*

*// the loop function runs over and over again forever*

*void loop() {*

***// read the cross walk button value:***

*Xw\_value=digitalRead(Xw\_button);*

*if (Xw\_value == 0 ){ // if crosswalk button (X-button) pressed*

*digitalWrite(yellow\_LED1 , LOW); // This should turn off the YELLOW LED1*

*digitalWrite(green\_LED1, LOW); // This should turn off the GREEN LED1*

*digitalWrite(yellow\_LED2 , LOW); // This should turn off the YELLOW LED2*

*digitalWrite(green\_LED2, LOW); // This should turn off the GREEN LED2*

*for (int i=10; i>0; i--){*

*Serial.print(" Count = ");*

*Serial.print(i);*

*Serial.println(" == Walk == ");*

*digitalWrite(red\_LED1, HIGH); // This should turn on the RED LED1*

*digitalWrite(red\_LED2, HIGH); // This should turn on the RED LED2*

*delay(500); //wait 0.5 seconds*

*digitalWrite(red\_LED1, LOW); // This should turn off the RED LED1*

*digitalWrite(red\_LED2, LOW); // This should turn off the RED LED2*

*delay(500); //wait 0.5 seconds*

*} // End of counter*

*} //*

*else // No Emergency ===*

*{*

*Serial.println(" == Do Not Walk == ");*

*// 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*

*delay(1000); //Extended time for Red light#1 before the Green of the other side turns ON*

*// The next three lines of code turn on the red LED2*

*digitalWrite(red\_LED2, LOW); // This should turn off the RED LED2*

*digitalWrite(yellow\_LED2 , LOW); // This should turn off the YELLOW LED2*

*digitalWrite(green\_LED2, HIGH); // This should turn on the GREEN LED2*

*delay(2000); // wait for 1 second*

*// 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*

*// The next three lines of code turn on the red LED2*

*digitalWrite(red\_LED2, LOW); // This should turn off the RED LED2*

*digitalWrite(yellow\_LED2 , HIGH); // This should turn on the YELLOW LED2*

*digitalWrite(green\_LED2, LOW); // This should turn off the GREEN LED2*

*delay(2000); // wait for 1 second*

*// The next three lines of code turn on the red LED2*

*digitalWrite(red\_LED2, HIGH); // This should turn on the RED LED2*

*digitalWrite(yellow\_LED2 , LOW); // This should turn off the YELLOW LED2*

*digitalWrite(green\_LED2, LOW); // This should turn off the GREEN LED2*

*delay(1000); //Extended time for Red light#2 before the Green of the other side turns ON*

*// The next three lines of code turn on the yellow LED1*

*digitalWrite(red\_LED1, LOW); // This should turn off the RED LED1*

*digitalWrite(yellow\_LED1 , LOW); // This should turn off the YELLOW LED1*

*digitalWrite(green\_LED1, HIGH); // This should turn on the GREEN LED1*

*delay(2000); // wait for 1 second*

*// The next three lines of code turn on the yellow LED1*

*digitalWrite(red\_LED1, LOW); // This should turn off the RED LED1*

*digitalWrite(yellow\_LED1 , HIGH); // This should turn on the YELLOW LED1*

*digitalWrite(green\_LED1, LOW); // This should turn off the GREEN LED1*

*// The next three lines of code turn on the red LED2*

*digitalWrite(red\_LED2, HIGH); // This should turn on the RED LED2*

*digitalWrite(yellow\_LED2 , LOW); // This should turn off the YELLOW LED2*

*digitalWrite(green\_LED2, LOW); // This should turn off the GREEN LED2*

*delay(2000); // wait for 1 second*

*}// Emergency Button closing ============*

*}*

## Sample Serial Monitor output:

***Capture similar Serial Monitor data and submit with your deliverable***

*54*

*==* ***Do Not Walk*** *==*

*23*

*Count = 10 == Walk ==*

*Count = 9 == Walk ==*

*Count = 8 == Walk ==*

*Count = 7 == Walk ==*

*Count = 6 == Walk ==*

*Count = 5 == Walk ==*

*Count = 4 == Walk ==*

*Count = 3 == Walk ==*

*Count = 2 == Walk ==*

*Count = 1 == Walk ==*

*73*

*== Do Not Walk ==*