**National Textile University, Faisalabad**

A logo with blue and red squares

AI-generated content may be incorrect.

**Department of Computer Science**

|  |  |
| --- | --- |
| **Name:** | Meerab Zahra |
| **Class:** | BSCS 5th A |
| **Registration No:** | 23-NTU-CS-1049 |
| **Assignment:** | 1 |
| **Course Name:** | Embedded IoT Systems |
| **Submitted To:** | Sir Nasir Mahmood |
| **Submission Date:** | October 26, 2025 |

**Question 3 — Implementation**

Circuit Diagram: Design a Wokwi circuit and draw a neat hand-sketch including:

• 2 push buttons

• 3 LEDs

• 1 buzzer

• 1 OLED

**Task A — Coding:**

Use one button to cycle through LED modes (display the current state on the OLED):

1. Both OFF

2. Alternate blink

3. Both ON

4. PWM fade

Use the second button to reset to OFF.

**Code:**

//Meerab Zahra

//23-NTU-CS-1049

//Assignment1Q3\_PartA

#include <Arduino.h>

#include <Wire.h>

#include <Adafruit\_GFX.h>

#include <Adafruit\_SSD1306.h>

// OLED Setup

#define SCREEN\_WIDTH 128

#define SCREEN\_HEIGHT 64

Adafruit\_SSD1306 display(SCREEN\_WIDTH, SCREEN\_HEIGHT, &**Wire**, -1);

// Pins

#define MODE\_BUTTON 36

#define RESET\_BUTTON 34

#define LED1\_PIN 18

#define LED2\_PIN 19

#define LED3\_PIN 5

// PWM Channels

#define LED1\_CH 1

#define LED2\_CH 2

#define LED3\_CH 3

// PWM Settings

#define FREQ 5000

#define RESOLUTION 10

// Variables

int mode = 0;

unsigned long previousMillis = 0;

bool blinkState = false;

int fadeValue = 0;

int fadeDir = 1;

// Forward declaration

void updateDisplay();

void setup() {

  pinMode(MODE\_BUTTON, INPUT);    // Using external resistors

  pinMode(RESET\_BUTTON, INPUT);

  ledcAttach(LED1\_PIN, FREQ, 8);

  ledcAttach(LED2\_PIN, FREQ, 8);

  ledcAttach(LED3\_PIN, FREQ, 8);

  // OLED init

  if (!display.begin(SSD1306\_SWITCHCAPVCC, 0x3C)) { for(;;); }

  display.clearDisplay();

  display.setTextSize(1);

  display.setTextColor(SSD1306\_WHITE);

  display.setCursor(0, 20);

  display.print("Initializing...");

  display.display();

  mode = 0; // Start in initializing mode

  previousMillis = millis();

}

void loop() {

  // --- Mode button ---

  if (digitalRead(MODE\_BUTTON) == HIGH) {

    delay(200); // debounce

    if (mode == 0) mode = 1;   // Go from initializing to first mode

    else {

      mode++;

      if (mode > 4) mode = 1;

    }

    updateDisplay();

    while (digitalRead(MODE\_BUTTON) == HIGH); // wait for release

  }

  if (digitalRead(RESET\_BUTTON) == HIGH) {

  delay(200);

  mode = 99;

  ledcWrite(LED1\_PIN, 0);

  ledcWrite(LED2\_PIN, 0);

  ledcWrite(LED3\_PIN, 0);

  // Display "Circuit Reset" on OLED

  display.clearDisplay();

  display.setTextSize(1);

  display.setCursor(0, 20);

  display.print("Circuit Reset");

  display.display();

  while (digitalRead(RESET\_BUTTON) == HIGH);

}

  // --- Mode behavior ---

  switch (mode) {

    case 0: // Initializing

      ledcWrite(LED1\_PIN, 0);

      ledcWrite(LED2\_PIN, 0);

      ledcWrite(LED3\_PIN, 0);

      break;

    case 1: // All OFF

      ledcWrite(LED1\_PIN, 0);

      ledcWrite(LED2\_PIN, 0);

      ledcWrite(LED3\_PIN, 0);

      break;

    case 2: // Alternate Blink

      if (millis() - previousMillis > 500) {

        previousMillis = millis();

        blinkState = !blinkState;

        ledcWrite(LED1\_PIN, blinkState ? 255 : 0);

        ledcWrite(LED2\_PIN, blinkState ? 0 : 179);

        ledcWrite(LED3\_PIN, blinkState ? 128 : 0);

      }

      break;

    case 3: // All ON

      ledcWrite(LED1\_PIN, 255);

      ledcWrite(LED2\_PIN, 179);

      ledcWrite(LED3\_PIN, 128);

      break;

    case 4: // PWM Fade

     ledcWrite(LED1\_PIN, fadeValue);

      ledcWrite(LED2\_PIN, 255 - fadeValue);

      ledcWrite(LED3\_PIN, fadeValue / 2);

      fadeValue += fadeDir;

      if (fadeValue <= 0 || fadeValue >= 255) fadeDir = -fadeDir;

      delay(10);

      break;

  }

}

void updateDisplay() {

  display.clearDisplay();

  if (mode == 0) {

    display.setTextSize(1);

    display.setCursor(0, 20);

    display.print("Initializing");

  } else {

    display.setTextSize(1);

    display.setCursor(0, 0);

    display.print("Current Mode:");

    display.setTextSize(2);

    display.setCursor(0, 20);

    if (mode == 1) display.print("All OFF");

    else if (mode == 2) display.print("Blinking");

    else if (mode == 3) display.print("All ON");

    else if (mode == 4) display.print("PWM Fade");

  }

  display.display();

}

**Wokwi Circuit Diagram**

A computer chip with wires and a screen

AI-generated content may be incorrect.

**Project Goal**

* When the circuit has just started, it shows **Initializing** on the OLED.
* When the mode button is pressed for the first time, all LEDs are off and OLED displays current event.
* When mode button is pressed again, event changes and LEDs blink alternatively.
* When mode button is pressed for third time, OLED displays current event as All LEDs on.
* Pressing mode button again changes the event on OLED to PWM fade and LEDs behave according to that event.
* After pressing the mode button four times, it loops back and adapts to the first mode where all LEDs were off.
* Reset button when pressed anytime, it changes the event on OLED to circuit reset and all LEDs are off indicating that the circuit has been reset.

**Wokwi Circuit Output**

* **Simulation Start**

**A computer screen shot of a computer

AI-generated content may be incorrect.**

* **When mode button is pressed for first time**

**A screenshot of a computer

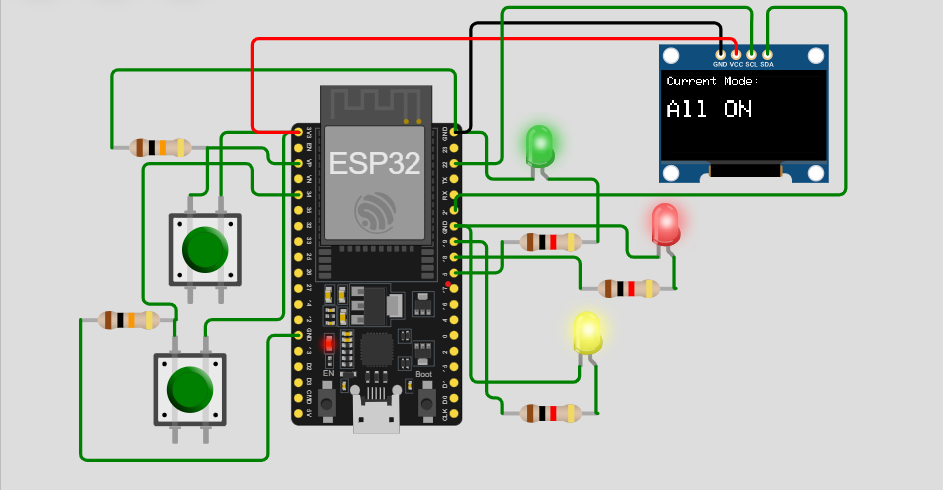
AI-generated content may be incorrect.**

* **When mode button is pressed for second time**

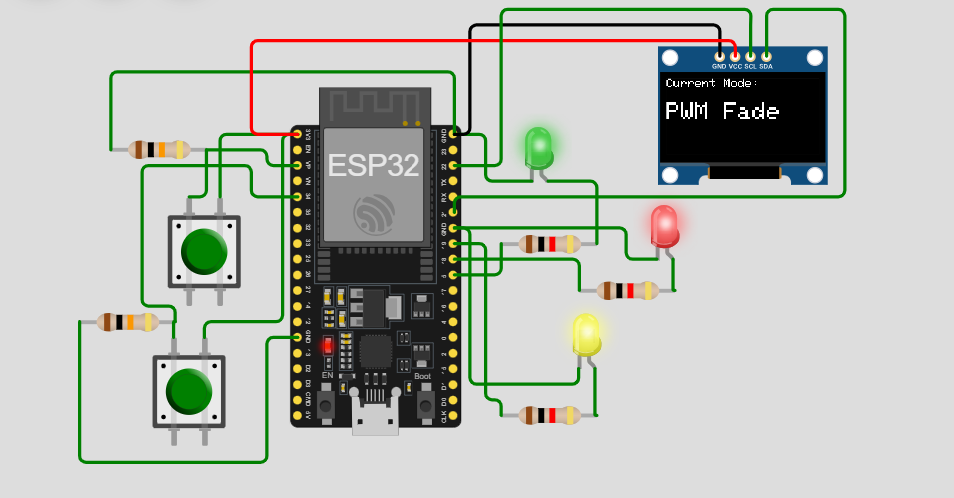
**A screenshot of a computer

AI-generated content may be incorrect.**

* **When mode button is pressed for third time**

****

* **When mode button is pressed for fourth time**

****

* **When mode button is pressed for fifth time, it loops giving same output as pressing mode button the first time**

**A screenshot of a computer

AI-generated content may be incorrect.**

* **When reset button is pressed**

**A computer circuit board with many wires

AI-generated content may be incorrect.**

**Hardware Circuit Setup**

In the designed circuit, ESP32 has been attached to a breadboard. A jumper wire from voltage pin (3.3V) of ESP32 has been connected to the voltage line of breadboard so voltage can be shared among components. Same has been done to share ground of ESP32 among the components. An OLED display has been connected to ESP32 with its ground wire connected to ground, voltage wire connected to voltage, SCL connected to pin 22 and SDA connected to pin 21. Red LED has been connected in such a way that its negative side is grounded and on positive side, 1 kilo ohm resistor is connected and that resistor is connected to pin 18. Green and Yellow LEDs are connected to ESP32 the same way but they use pins 5 and 19 respectively. There are two push buttons in the circuit also. One acts as mode button and the other as reset button. To connect button, take any of two pairs and use both pins. One pin is attached to voltage and to the other pin, 10 kilo ohm resistor is attached. The resistor side connected to button pin is then connected to GPIO and the other side of resistor is grounded. For mode button, GPIO 36 is used and for reset button, GPIO 34 is used.

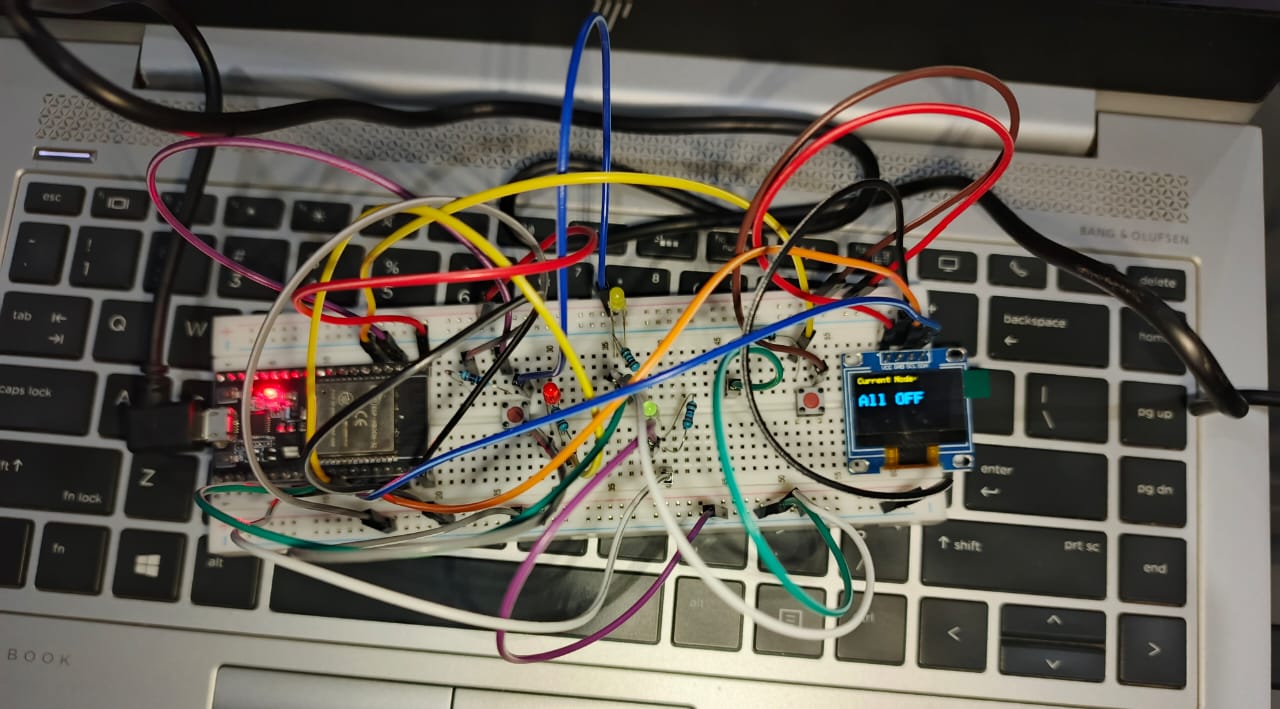
**Hardware Circuit Output**

* **Circuit Simulation**

A circuit board with wires

AI-generated content may be incorrect.

* **All LEDs Off state**



* **Alternate blinking state**

A computer with wires on it

AI-generated content may be incorrect.

* **All LEDs On state**

A computer circuit board with many wires

AI-generated content may be incorrect.

* **PWM fading state**

**A computer with wires on it

AI-generated content may be incorrect.**

* **Circuit reset state**

