National Textile University, Faisalabad

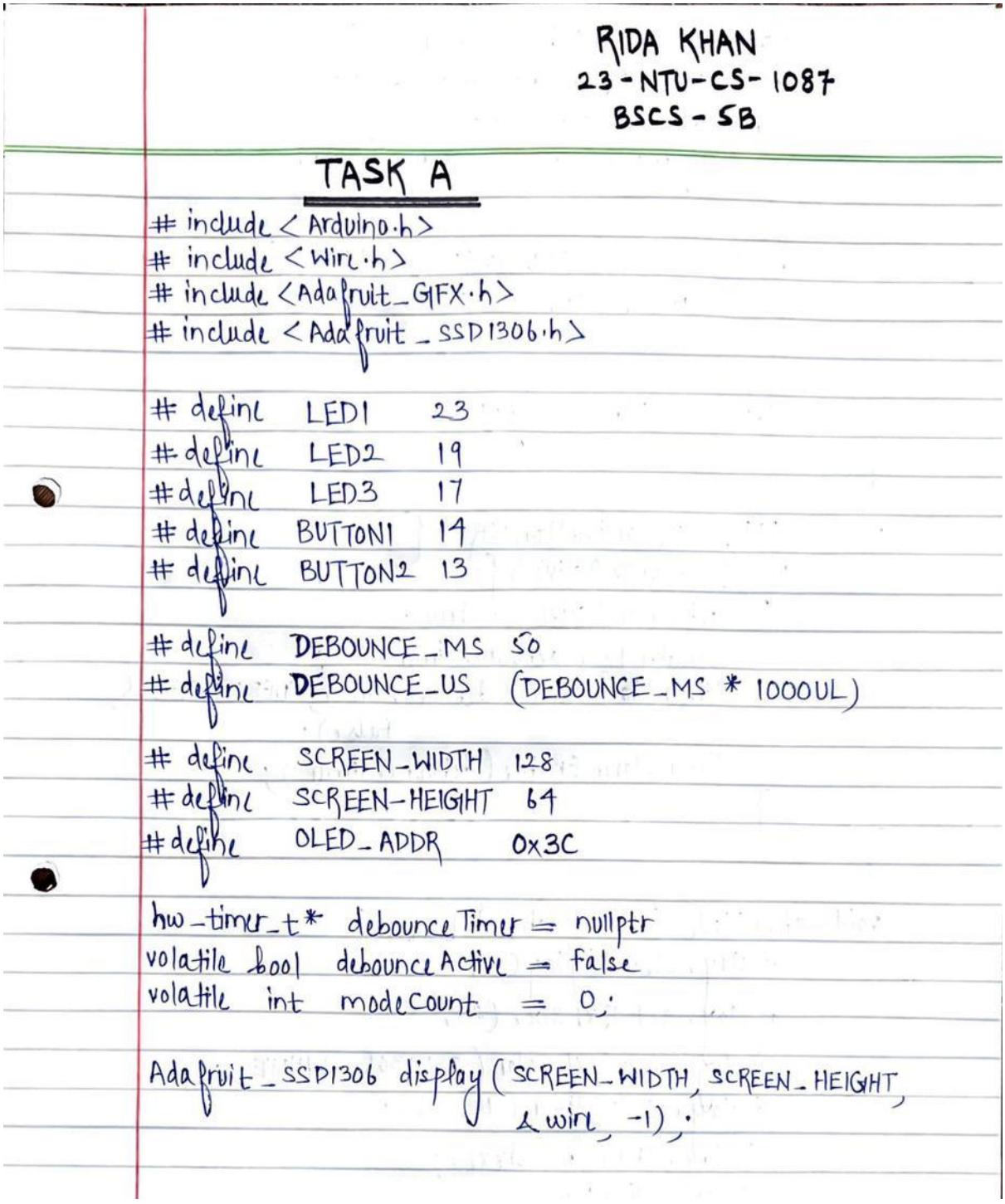
Department of Computer Science

Name:	Rida Khan
Class:	BSCS – 5B
Registration No:	23-NTU-CS-1087
Course Name:	Embedded IOT Systems
Submitted To:	Sir Nasir Mahmood
Submission Date:	25th Oct, 2025

Assignment 1 – Question 3

Code (Handwritten)

Task A:



RIDA KHAN
23-NTU-CS-1087
BSCS-5B

TASK A

```
#include <Arduino.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define LED1 23
#define LED2 19
#define LED3 17
#define BUTTON1 14
#define BUTTON2 13

#define DEBOUNCE_MS 50
#define DEBOUNCE_US (DEBOUNCE_MS * 1000UL)

#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_ADDR 0x3C

hw_timer_t* debounceTimer = nullptr
volatile bool debounceActive = false
volatile int modeCount = 0;

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT,
                         &Wire, -1);
```

```
void IRAM_ATTR onDebounceTimer() {
    if (digitalRead(BUTTON1) == LOW) {
        modeCount++;
        if(modeCount > 3) modeCount = 0;
    }
    if (digitalRead(BUTTON2) == LOW) {
        modeCount = 0;
    }
    debounceActive = false;
}

void IRAM_ATTR onButtonISR() {
    if(!debounceActive) {
        debounceActive = true;
        timerWrite(debounceTimer, 0);
        timerAlarmWrite(debounceTimer, DEBOUNCE_US,
                        false);
        timerAlarmEnable(debounceTimer);
    }
}

void showMode(const char* text) {
    display.clearDisplay();
    display.setTextSize(2);
    display.setTextColor(SSD1306_WHITE);
    display.setCursor(10, 26);
    display.println(text);
    display.display();
}
```

```
void setup() {
  ledcSetup(2, 5000, 8);
  ledcAttachPin(LED3, 2);

  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);

  digitalWrite(LED1, LOW);
  digitalWrite(LED2, LOW);

  pinMode(BUTTON1, INPUT_PULLUP);
  pinMode(BUTTON2, INPUT_PULLUP);
  attachInterrupt(BUTTON1, onButtonISR, FALLING);
  attachInterrupt(BUTTON2, onButtonISR, FALLING);

  debounceTimer = timerBegin(3, 80, true);
  timerAttachInterrupt(debounceTimer, &onDebounceTimer,
                                              true);
  Wire.begin(21, 22);
  if(!display.begin(SSD1306_SWITCHCAPVCC, OLED_ADDR))
  {  for(;;);
  }

  display.clearDisplay();
  showMode("ALL OFF");
}
```

```
void loop() {
  switch (modeCount) {
    case 0:
      digitalWrite(LED1, LOW);
      digitalWrite(LED2, LOW);
      ledcWrite(2,0)
      showMode("BOTH OFF");
      break;

    case 1:
      showMode("ALTERNATE");
      digitalWrite(LED1, LOW);
      digitalWrite(LED2, HIGH);
      ledcWrite(2,0);
      delay(400);
      digitalWrite(LED1, HIGH);
      digitalWrite(LED2, LOW);
      ledcWrite(2,0);
      delay(400);
      break;

    case 2:
      digitalWrite(LED1, HIGH);
      digitalWrite(LED2, HIGH);
      ledcWrite(2,0);
      showMode("BOTH ON");
      break;

    case 3:
      digitalWrite(LED1, LOW);
      digitalWrite(LED2, LOW);
      showMode("PWM FADE");
```

```
for (int d = 0; d <= 255; d = d+5) {
        ledcWrite (2, d);
        delay (10);
}
for (int d = 255; d >= 0; d = d-5) {
        ledcWrite (2, d);
        delay (10);
}
break;
}
delay (50);
}
```

Task B:

RIDA KHAN
23-NTU-CS-1087
BSCS-5B

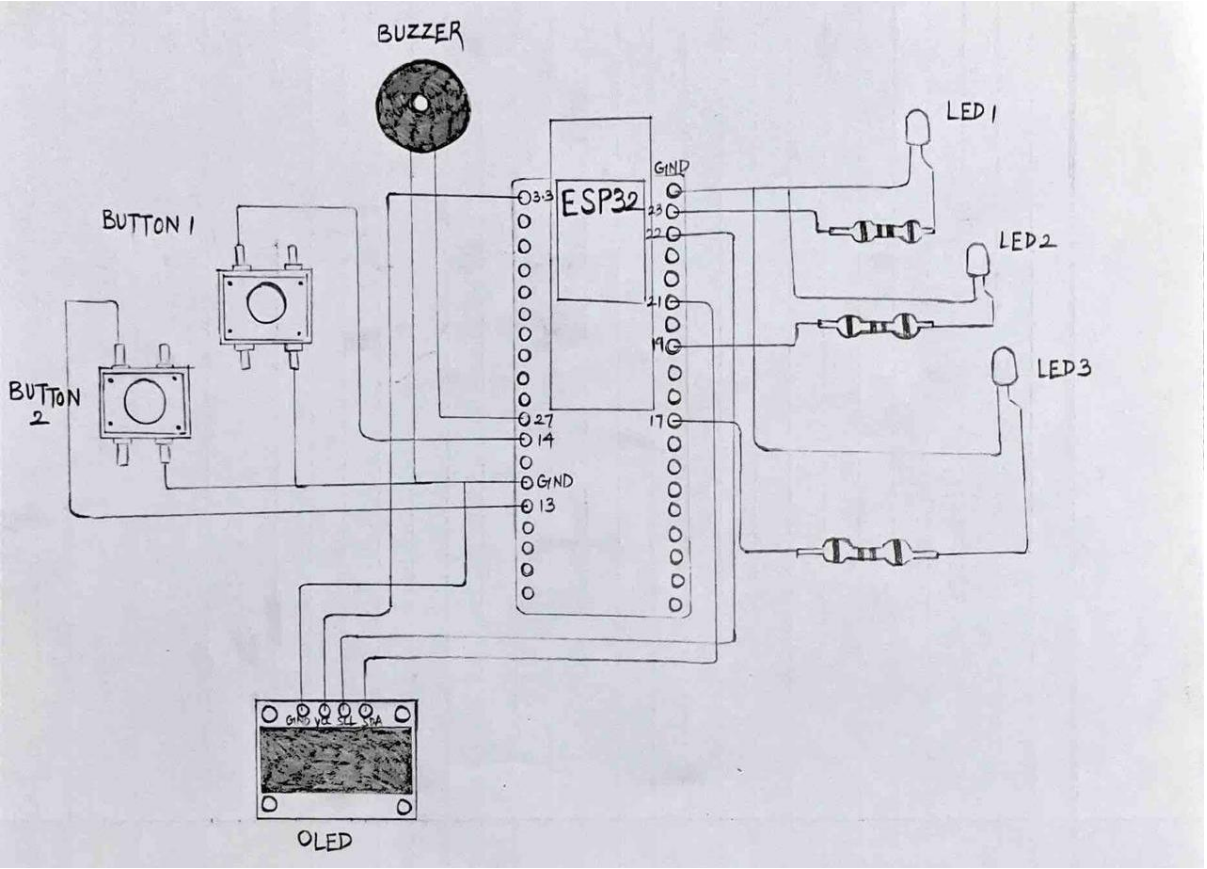
TASK B

```
#include <Arduino.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define LED1 23
#define BUTTON1 14
#define BUZZER_PIN 27

#define PWM_CH 0
#define FREQ 2000
#define RESOLUTION 10

#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_ADDR 0x3C

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT,
                         &Wire, -1);

bool ledstate = false;
unsigned long press start = 0;
bool ButtonPressed = false;

void playBuzzerTone() {
    int melody[] = {
        330, 392, 330, 440, 494, 523, 392,
        262, 330, 392, 262, 196, 262, 330
    };
```

```
    for (int i=0; i<8; i++) {
      ledcWriteTone(PWM_CH, melody[i]);
      delay(200);
    }
    ledcWrite(PWM_CH, 0);
}

void setup() {
    pinMode(LED1, OUTPUT);
    pinMode(BUTTON1, INPUT_PULLUP);

    ledcSetup(PWM_CH, FREQ, RESOLUTION);
    ledcAttachPin(BUZZER_PIN, PWM_CH);

    Wire.begin(21,22);
    if(!display.begin(SSD1306_SWITCHCAPVCC,
                      OLED_ADDR)) {
        for(;;);
    }

    display.clearDisplay();
    display.setTextSize(2);
    display.setTextColor(SSD1306_WHITE);
    display.setCursor(10,26);
    display.println("READY");
    display.display();
}
```

```
void loop() {
  if (digitalRead(BUTTON1) == LOW) {
    if (!buttonPressed) {
      buttonPressed = true;
      pressStart = millis();
    }
  } else if (buttonPressed) {
    unsigned long pressDuration = millis() -
                                  pressStart;
    buttonPressed = false;
    if (pressDuration < 1500) {
      ledState = !ledState;
      digitalWrite(LED1, ledState ? HIGH :
                                    LOW);
    display.clearDisplay();
    display.setCursor(10, 26);
    display.println(ledState ? "LED ON" : "LED OFF");
    display.display();
    } else {
      display.clearDisplay();
      display.setCursor(10, 26);
      display.println("BUZZER!");
      display.display();
      playBuzzerTone();
    }
    delay(50)
}
```

Wokwi Diagram (Drawn & Labelled)

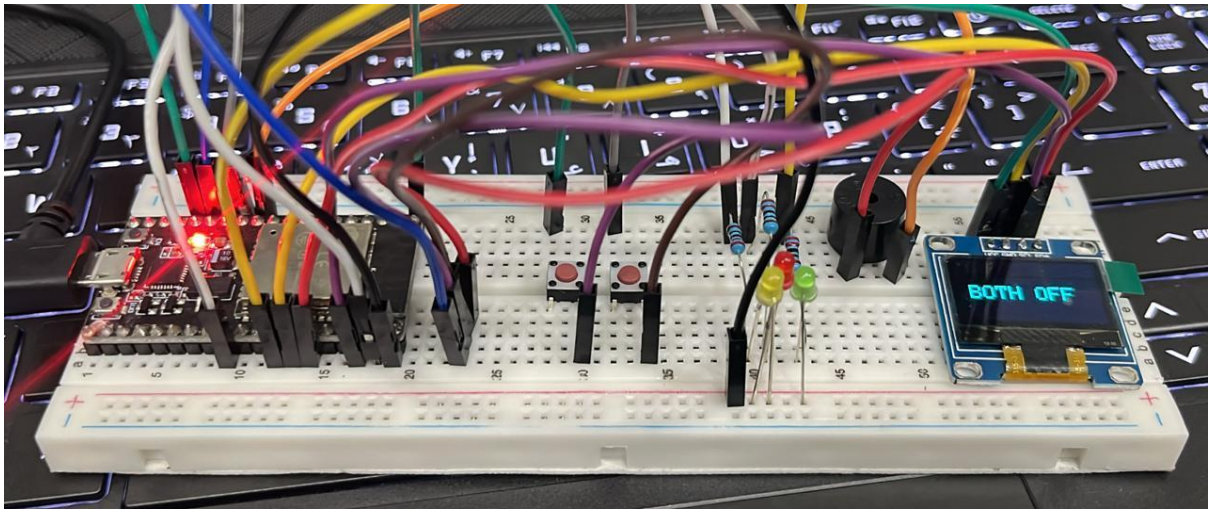


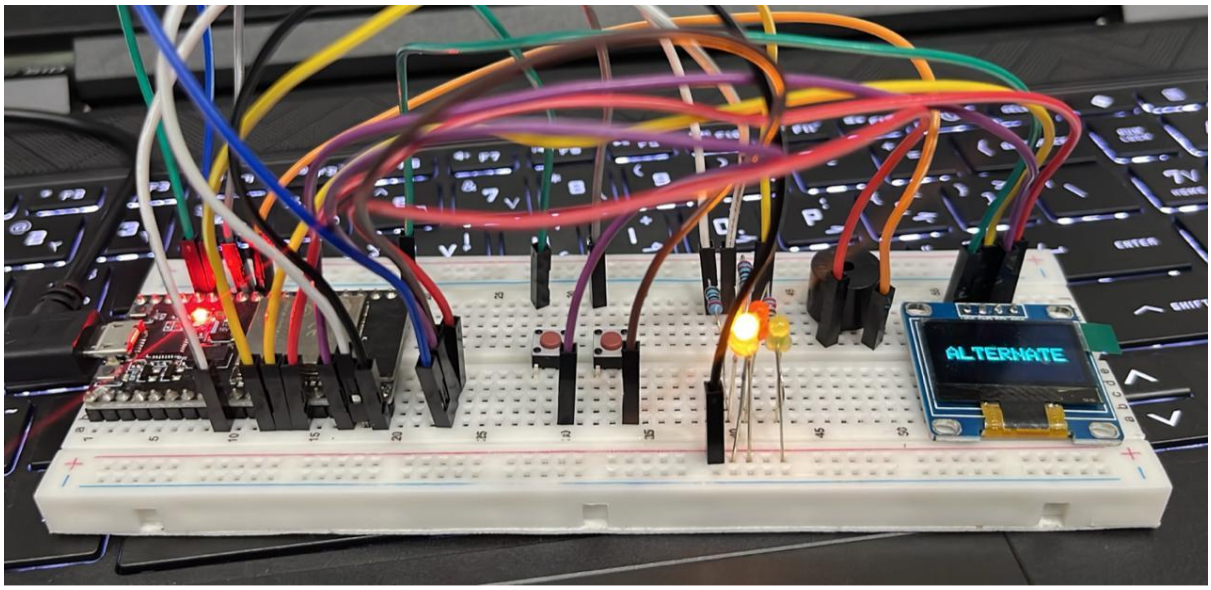
Pin Map

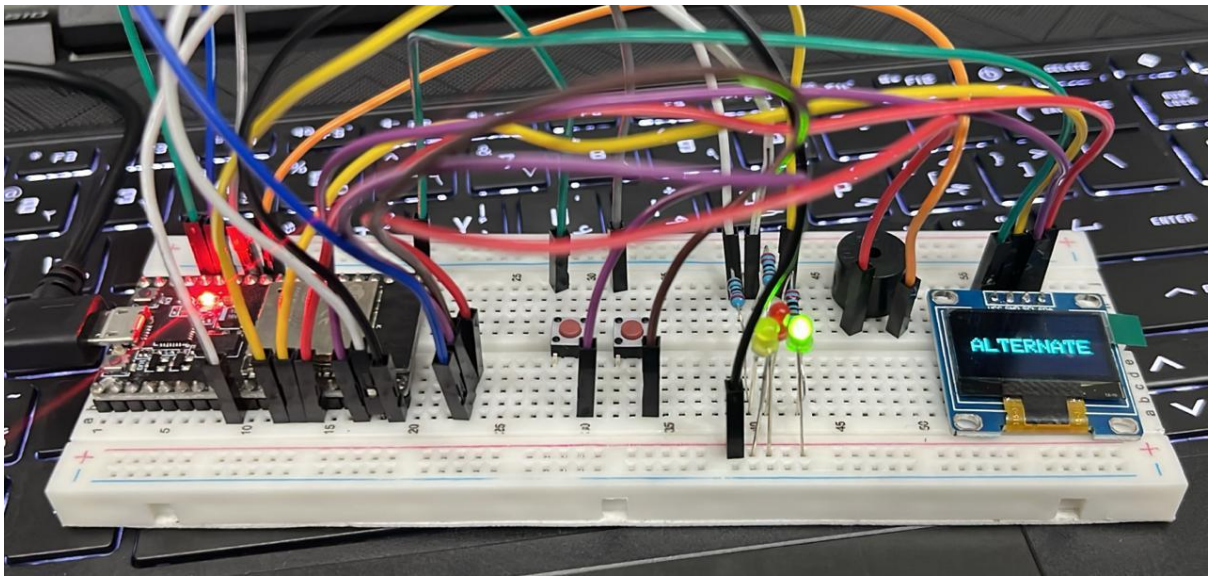
Component	ESP32 Pin
LED 1 (+)	GPIO 23
LED 1 (-)	GND
LED 2 (+)	GPIO 19
LED 2 (-)	GND
LED 3 (+)	GPIO 17
LED 3 (-)	GND
Button 1 (Mode)	GPIO 14
Button 2 (Reset)	GPIO 13
Buzzer (+)	GPIO 27
Buzzer (-)	GND
OLED SDA	GPIO 21
OLED SCL	GPIO 22
OLED VCC	3.3V
OLED GND	GND

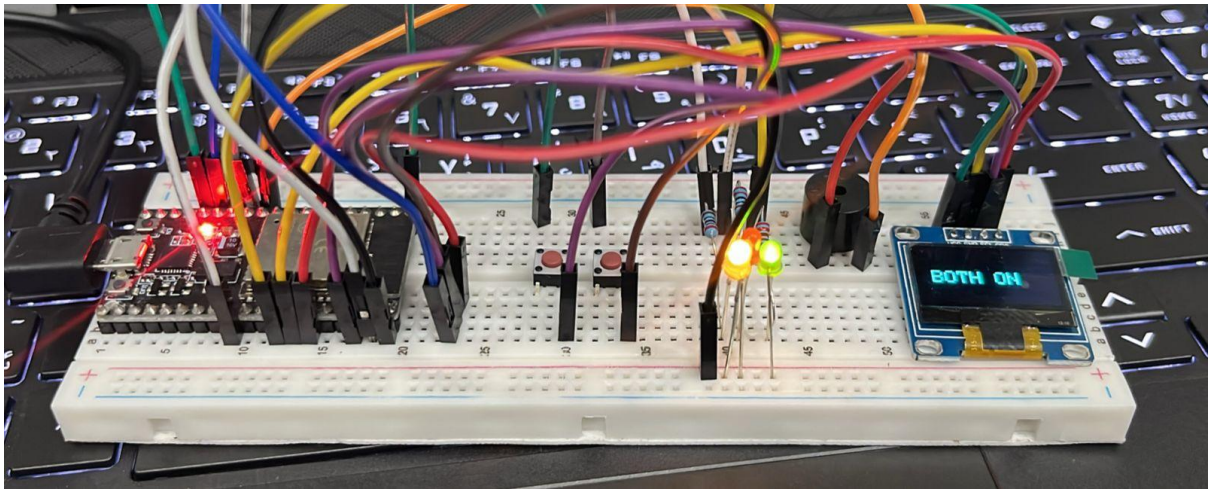
Pictures of Kit

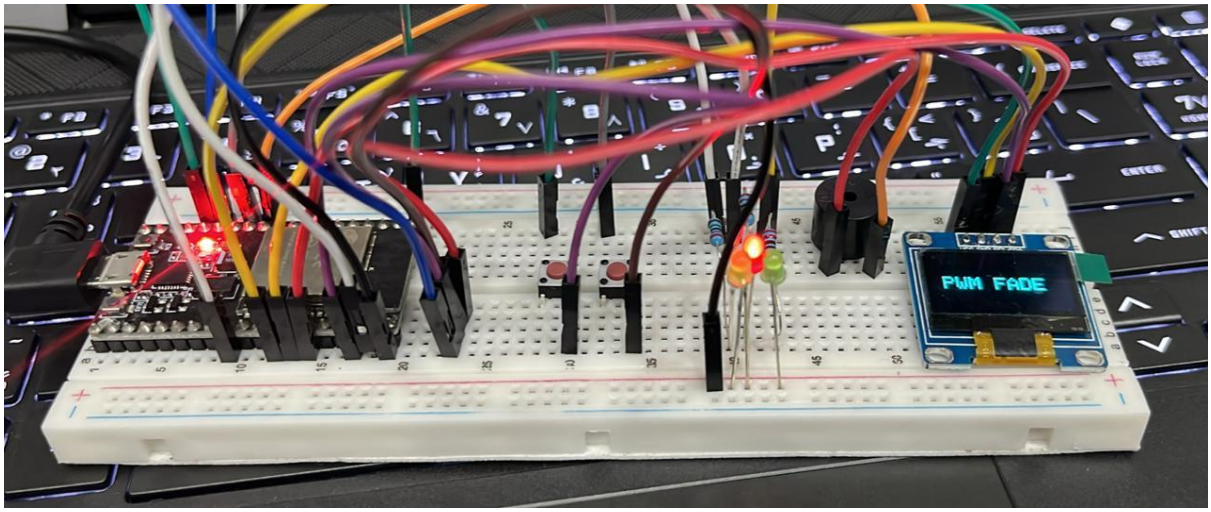
Task A:



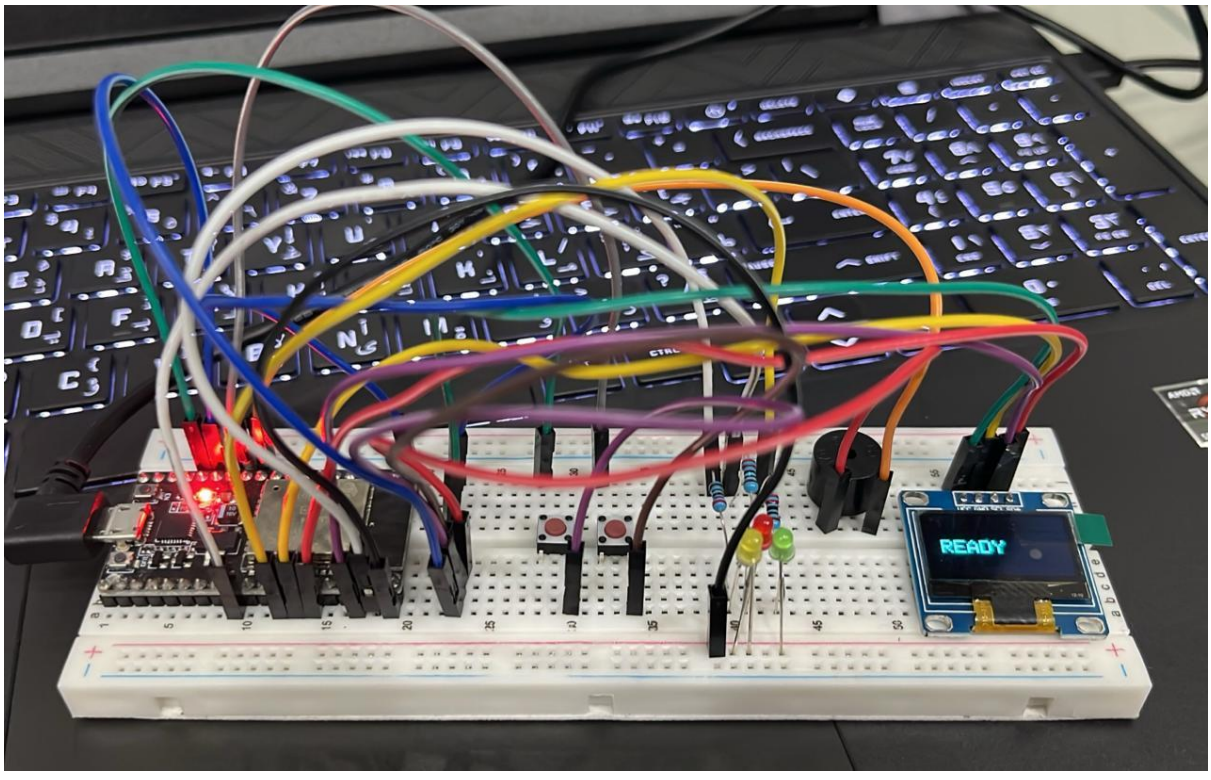


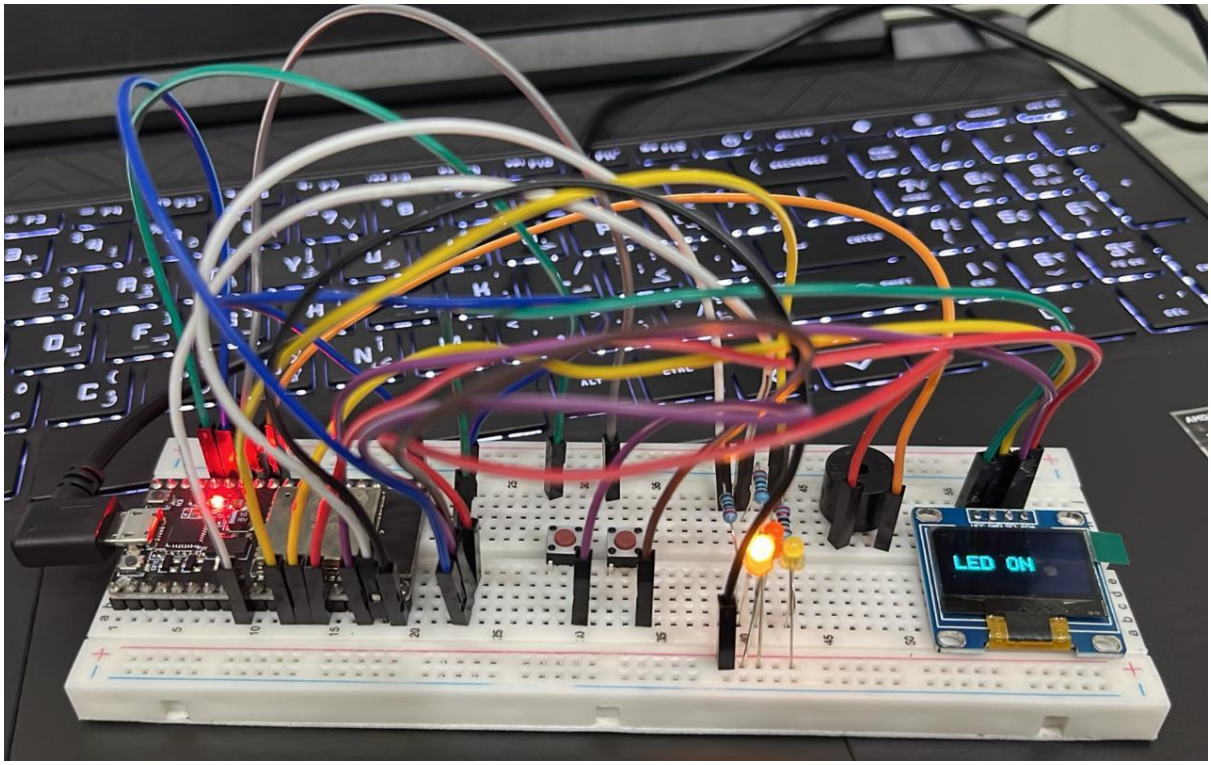


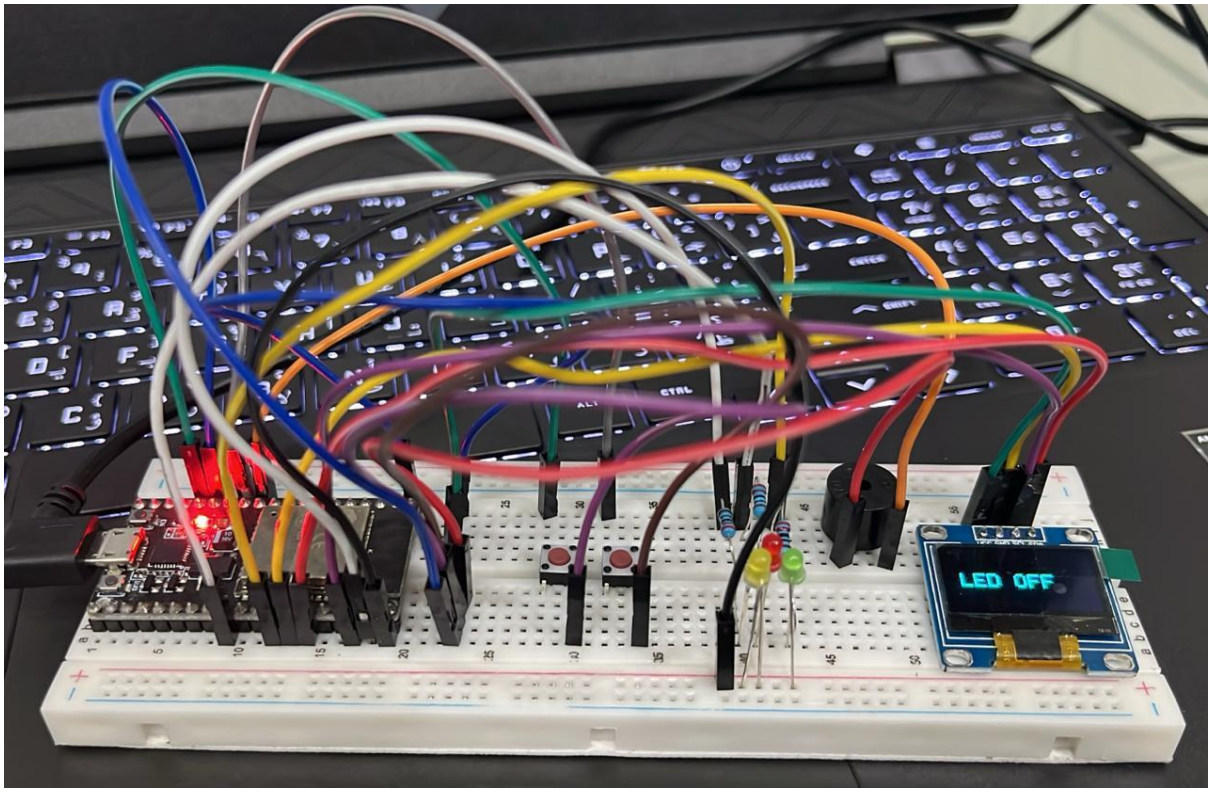


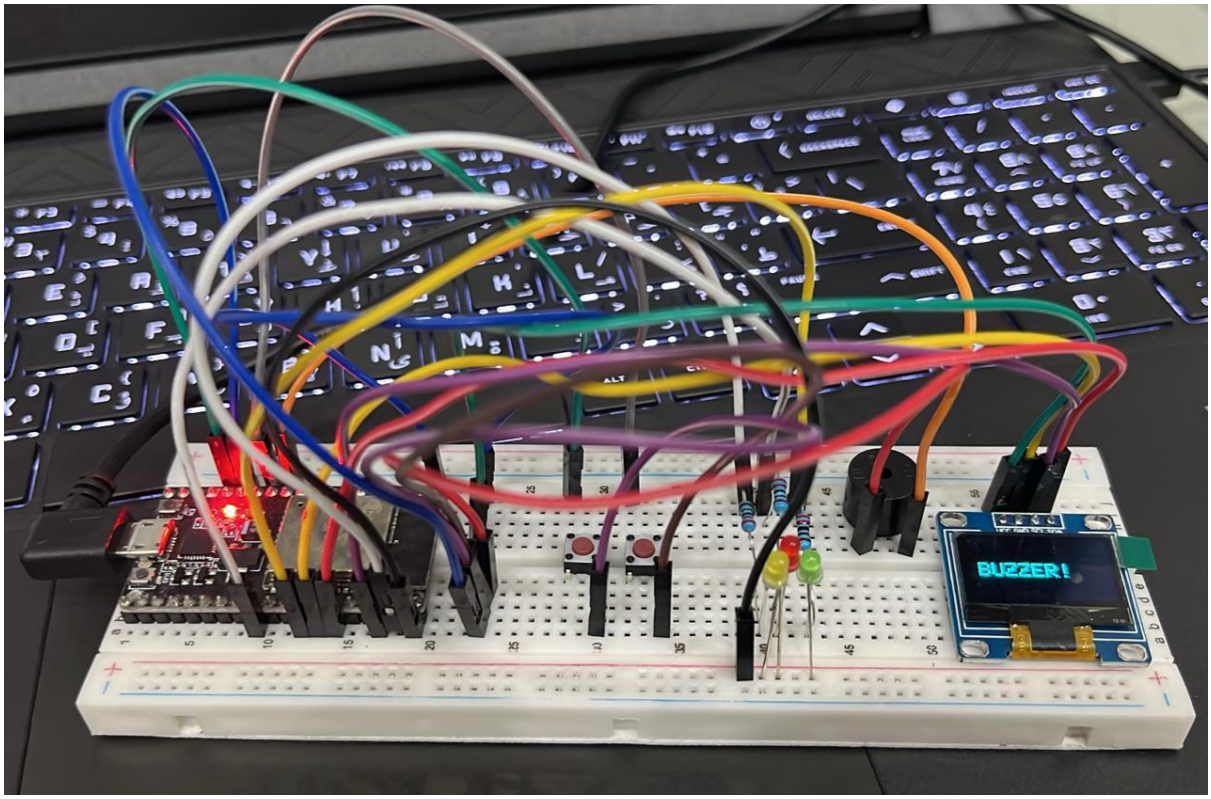


Task B:




LED ON


LED OFF



Wokwi Link

Task A: https://wokwi.com/projects/445525900687682561

Task B: https://wokwi.com/projects/445683571638419457

GitHub Link

https://github.com/nahkadir/embedded-iot-system/tree/main/assignment1-23-NTU-CS-1087