National Textile University, Faisalabad

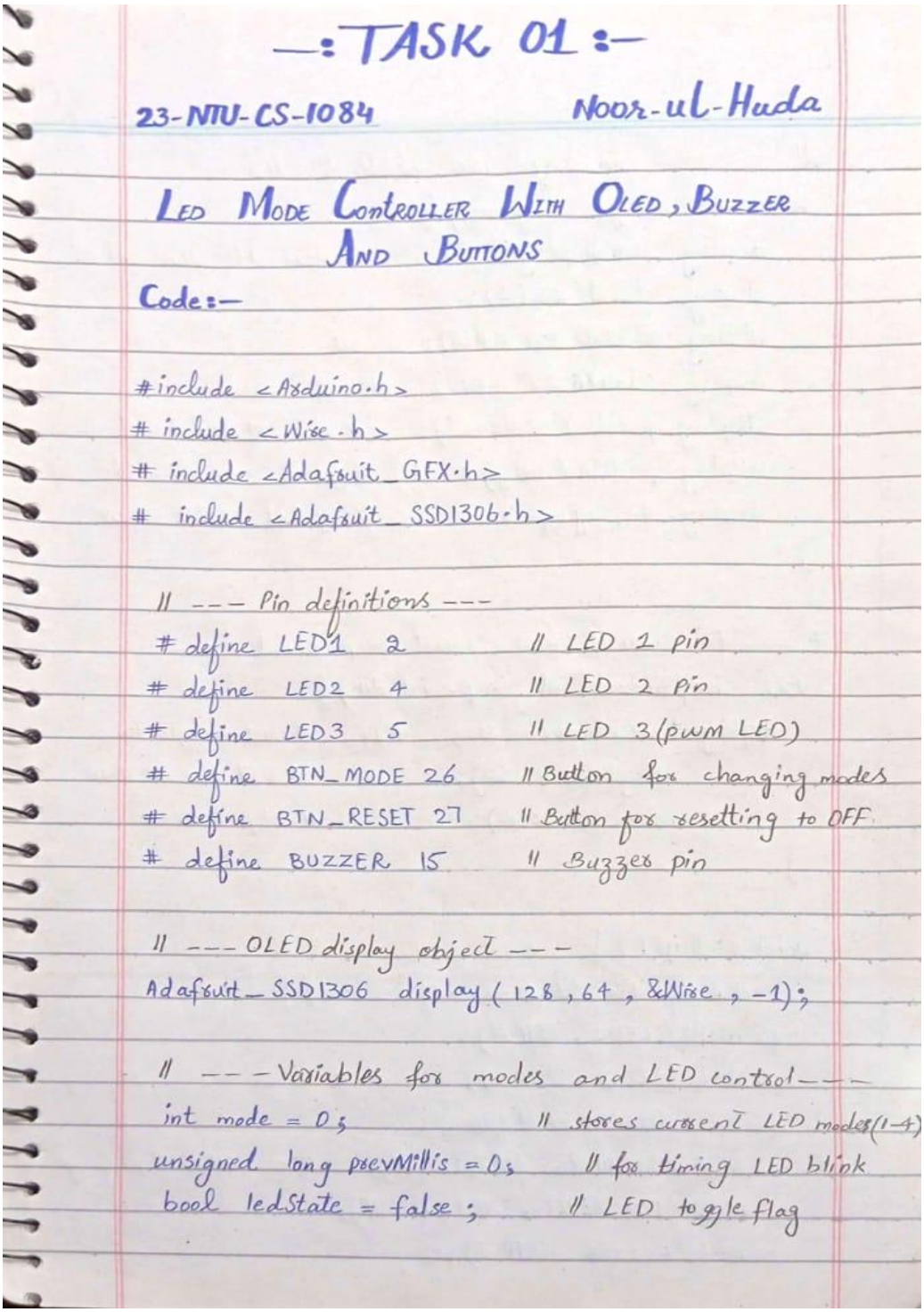
Department of Computer Science

Name:	Noor Ul Huda
Class:	BSCS_B 5th Semester
Registration No:	23-NTU-CS-1084
Assignment :	01
Course Name:	Embedded IOT System
Submitted To:	**Sir Nasir Mehmood**
Submission Date:	26/10/2025

Assignment 01

Task 01

Handwritten code:



—: TASK 01 :—

23-NTU-CS-1084 Noor-ul-Huda

LED MODE CONTROLLER WITH OLED, BUZZER AND BUTTONS

Code:—

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

// --- Pin definitions ---
#define LED1  2          // LED 1 pin
#define LED2  4          // LED 2 pin
#define LED3  5          // LED 3 (pwm LED)
#define BTN_MODE 26      // Button for changing modes
#define BTN_RESET 27     // Button for resetting to OFF
#define BUZZER 15        // Buzzer pin

// --- OLED display object ---
Adafruit_SSD1306 display(128, 64, &Wire, -1);

// ---Variables for modes and LED control---
int mode = 0;            // stores current LED mode (1-4)
unsigned long prevMillis = 0;   // for timing LED blink
bool ledState = false;   // LED toggle flag
```

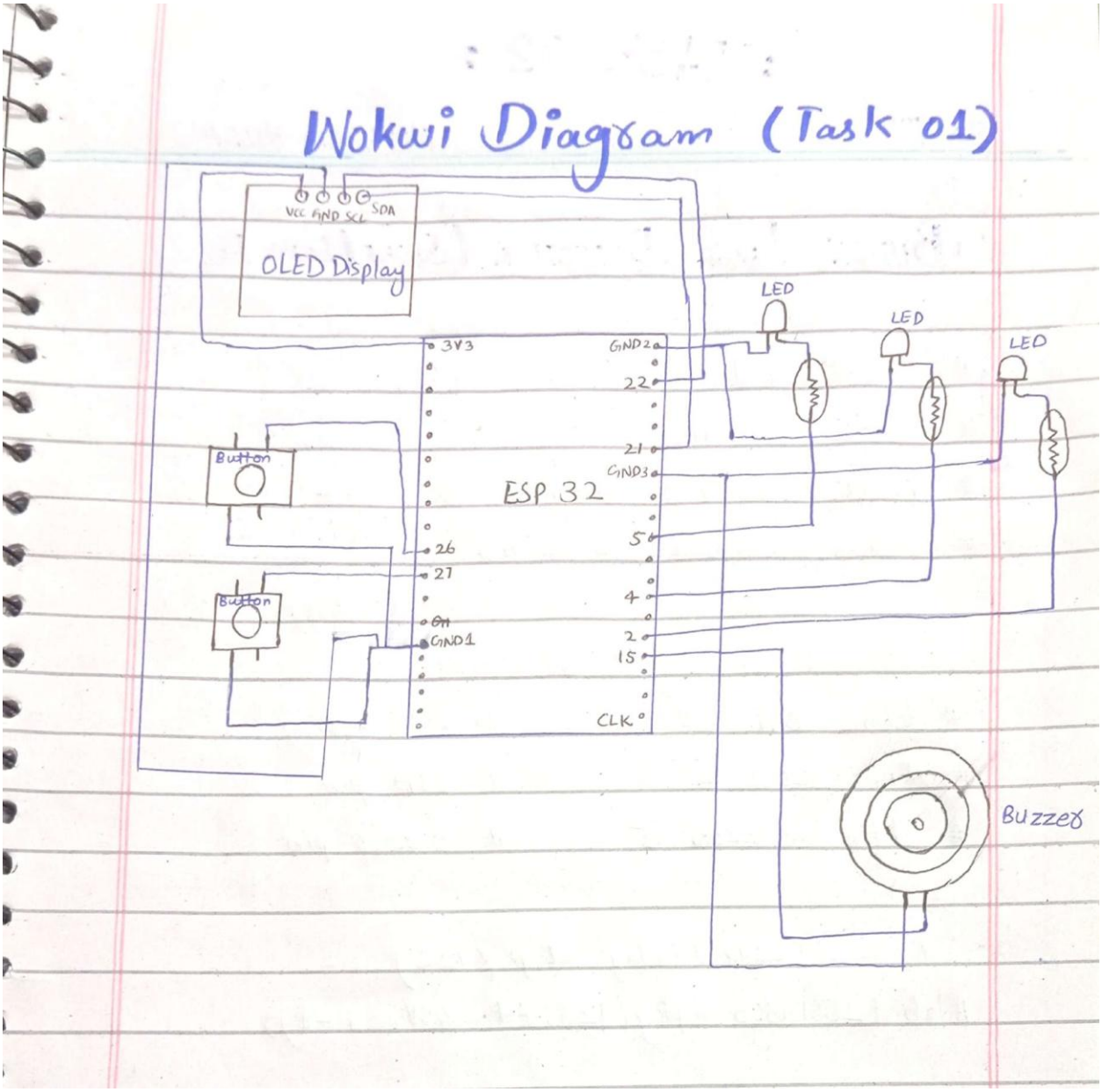
```
// --- Show message on OLED :Function ---
void showMsg(String msg) {
  display.clearDisplay();          // clear old screen content
  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.setCursor(0, 20);
  display.print("Mode: ");          // show "Mode:" label
  display.println(msg);            // print current mode
  display.display();               // update OLED screen
}

// --- Function: make a short beep sound ---
void beepBuzzer(int freq, int dur) {
  display tone(BUZZER, freq, dur);   // start buzzer tone
        delay(dur + 50);             // wait for duration
        noTone(BUZZER);              // stop buzzer
}

void setup() {
  // --- Pin setup ---
  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);
  pinMode(LED3, OUTPUT);
  pinMode(BTN_Mode, INPUT_PULLUP);   // use internal pull-up
  pinMode(BTN_RESET, INPUT_PULLUP);
  pinMode(BUZZER, OUTPUT);
```

```
// --- Initialize OLED display ---
display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
display.clearDisplay();
display.display();

showMsg("Both OFF");      // show default mode at start
}

void loop(){
  // --- Check Mode button ---
  if (digitalRead(BTN_MODE) == LOW) {
    delay(200);                    // debounce delay
    mode++;                        // next mode
    if (mode > 4) mode = 1;        // wrap back to mode 1 after 4

    // --- Change behaviours based on selected mode ---
    Switch (mode) {
      case 1:
        // Mode 1: Both LEDs OFF
        digitalWrite(LED1, LOW);
        digitalWrite(LED2, LOW);
        showMsg("Both OFF");
        beepBuzzer(800, 120);      // short beep
        break;
      case 2:
        // Mode 2: Alternate Blink
```

```
        showMsg("Alternate Blink");
        beepBuzzer(1000, 120);
        break;
    case 3:
        // Mode 3: Both LEDs ON
        digitalWrite(LED1, HIGH);
        digitalWrite(LED2, HIGH);
        showMsg("Both ON")
        beepBuzzer(1200, 120);
        break;
    case 4:
        // Mode 4: PWM Fade
        showMsg("PWM Fade");
        beepBuzzer(1500, 120);
        break;
  }
}

// --- Check Reset button ---
if (digitalRead(BTN_RESET) == LOW) {
  delay(200);
  mode = 1;                    // go back to mode 1 (OFF)
  digitalWrite(LED1, LOW);
  digitalWrite(LED2, LOW);
  analogWrite(LED3, 0);
  showMsg("Reset to OFF")
```

```
        beepBuzzer(400, 200);
    }

    //--- Mode behaviors ---
    if (mode == 2) { // Alternate blink mode
        if (millis() - prevMillis >= 500) {
            prevMillis = millis();
            ledState = !ledState;        // toggle state
            digitalWrite(LED1, ledState);
            digitalWrite(LED2, !ledState);
        }
    }
    if (mode == 4) { // PWM Fade mode
        // increase LED brightness
        for (int i = 0; i <= 255; i++) {
            analogWrite(LED3, i);
            delay(5);
        }
        // decrease LED brightness
        for (int i = 255; i >= 0; i--) {
            analogWrite(LED3, i);
            delay(5);
        }
    }
}
```

Wokwi Diagram:

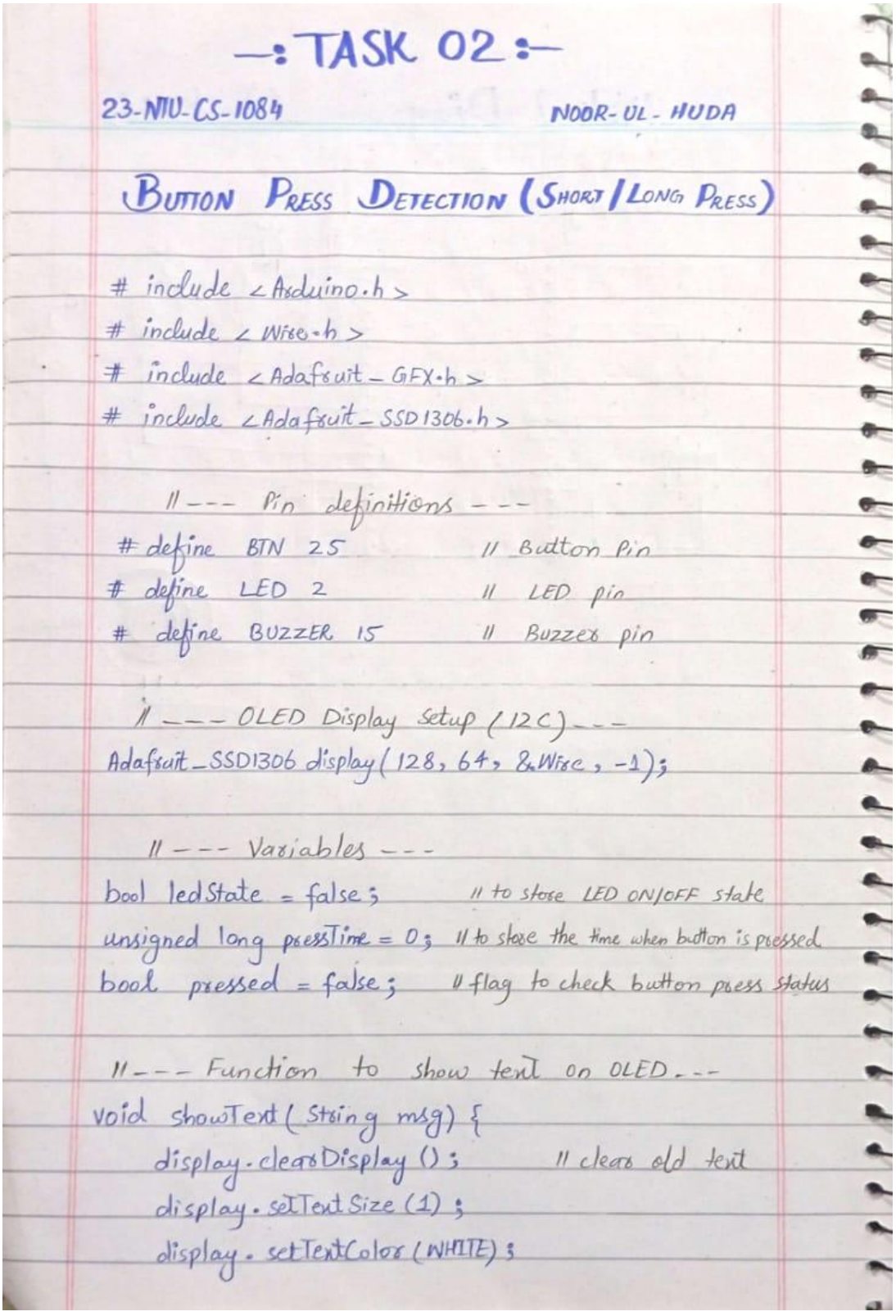


Wokwi link Task 01:

https://wokwi.com/projects/445712677554764801

Task 02

Handwritten code:



—: TASK 02 :—

23-NTU-CS-1084 NOOR-UL-HUDA

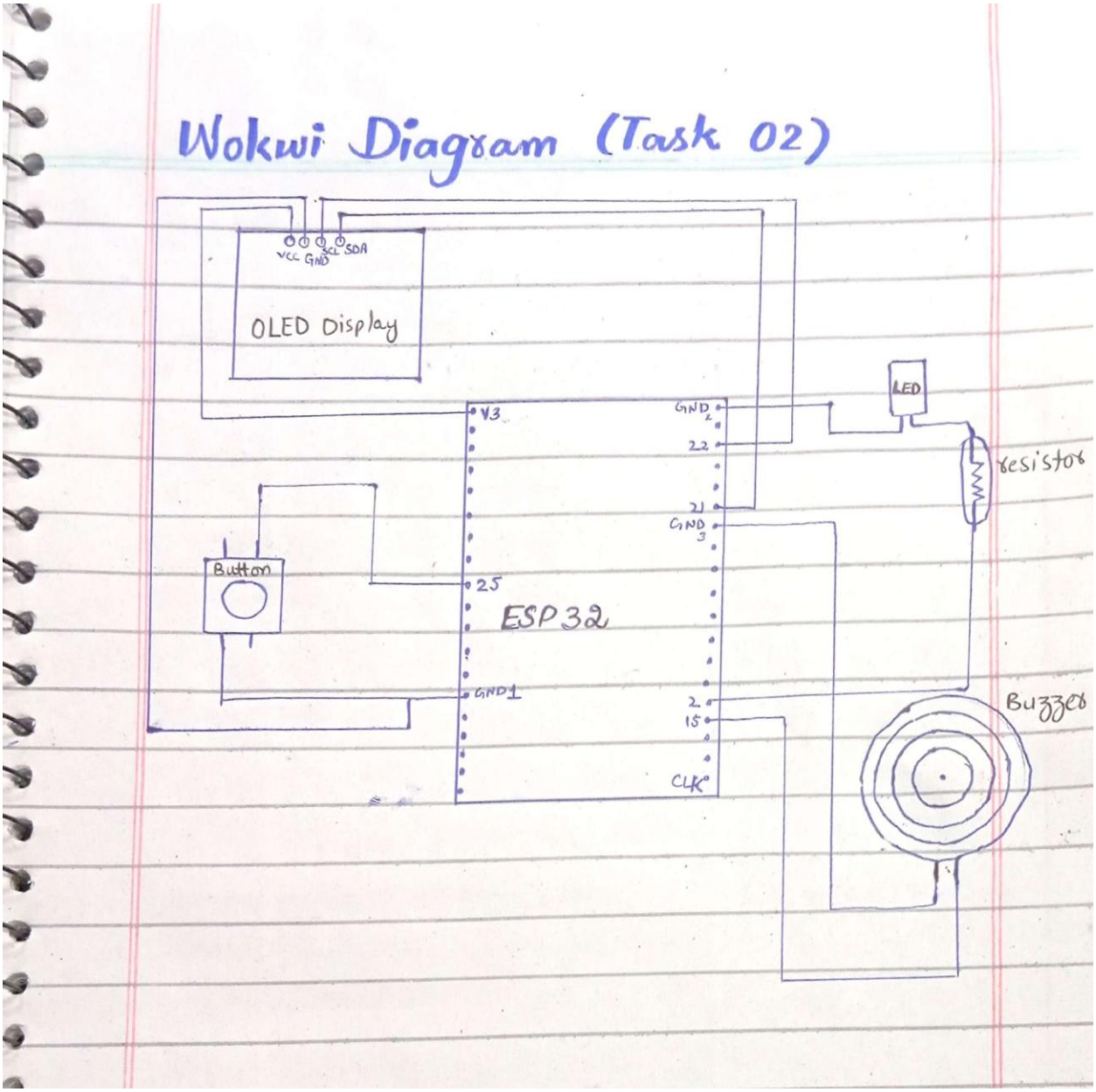
BUTTON PRESS DETECTION (SHORT / LONG PRESS)

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

// --- Pin definitions ---
# define BTN 25          // Button Pin
# define LED 2           // LED pin
# define BUZZER 15       // Buzzer pin

// --- OLED Display Setup (I2C) ---
Adafruit_SSD1306 display(128, 64, &Wire, -1);

// --- Variables ---
bool ledState = false;          // to store LED ON/OFF state
unsigned long pressTime = 0;    // to store the time when button is pressed
bool pressed = false;           // flag to check button press status

// --- Function to show text on OLED ---
void showText(String msg) {
    display.clearDisplay();     // clears old text
    display.setTextSize(1);
    display.setTextColor(WHITE);
```

```
    display.setCursor(0, 20);     //position for message
    display.println(msg);         // print message
    display.display();            // update OLED screen
}

void setup() {
    pinMode(BTN, INPUT_PULLUP);   // button as input with internal pull-up
    pinMode(LED, OUTPUT);         // LED as output
    pinMode(BUZZER, OUTPUT);      // buzzer as output

    //--- Initialize the OLED display ---
    display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
    showText("Ready...");         // startup message
}

void loop() {
  // --- Check if button is pressed down ---
  if (digitalRead(BTN) == LOW && !pressed) {
    pressed = true;               // mark button as pressed
    pressTime = millis();         // save press start time
  }
  // --- check if button is released ---
  if (digitalReal(BTN) == HIGH && pressed) {
    unsigned long duration = millis() - pressTime;
    pressed = false;              // reset press flag
```

```
        // --- Long press detection (>1.5s) ---
        if (duration > 1500) {
          tone(BUZZER, 1000, 500);
          showText("Long Press → Buzzed");
        }
        // --- Short Press Detection ---
        else {
          ledState = !ledState;    // toggle LED state
          digitalWrite(LED, ledState);
          showText("Short Press → LED Toggle");
        }
      }
    }
```

Wokwi Diagram



Wokwi link Task 02:
https://wokwi.com/projects/445714036162391041

Github link:
https://github.com/noorulhudaa53-bot/Embedded-IoT-1084.git