

National Textile University, Faisalabad



Department of Computer Science

Name:	Rizwana Bashir
Class:	BSCS-5 th -B
Registration No:	23-NTU-CS-FL-1088
Assignment:	1
Submission Date:	26-Oct-2025
Course Name:	Embedded IoT Systems
Submitted To:	Sir Nasir Mahmood

Embedded IoT Systems Assignment 1

Task B: Buzzer Tone.

```
Assignment_1_Task_B_Buzzer_tone > src > main.cpp > SCREEN_WIDTH
1 // Rizwana Bashir
2 // 23-NTU-CS-1088
3
4 #include <Wire.h>
5 #include <Adafruit_GFX.h>
6 #include <Adafruit_SSD1306.h>
7
8 // OLED configuration
9 #define SCREEN_WIDTH 128
10 #define SCREEN_HEIGHT 64
11 #define OLED_ADDR 0x3C
12 Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);
13
14 // Pin definitions
15 #define BUTTON_PIN 35 // Push button pin
16 #define GREEN_LED 19
17 #define BLUE_LED 18
18 #define PURPLE_LED 17
19 #define BUZZER_PIN 5
20
21 // Variables for press detection
22 unsigned long pressStartTime = 0;
23 bool buttonPressed = false;
24 bool ledsOn = false;
25
26 void setup() {
```

```
26 void setup() {
27     Serial.begin(115200);
28
29     // Pin modes
30     pinMode(BUTTON_PIN, INPUT_PULLUP); // internal pull-up
31     pinMode(GREEN_LED, OUTPUT);
32     pinMode(BLUE_LED, OUTPUT);
33     pinMode(PURPLE_LED, OUTPUT);
34     pinMode(BUZZER_PIN, OUTPUT);
35
36     // Initialize LEDs and buzzer off
37     digitalWrite(GREEN_LED, LOW);
38     digitalWrite(BLUE_LED, LOW);
39     digitalWrite(PURPLE_LED, LOW);
40     digitalWrite(BUZZER_PIN, LOW);
41
42     // Initialize OLED
43     if (!display.begin(SSD1306_SWITCHCAPVCC, OLED_ADDR)) {
44         Serial.println("OLED init failed!");
45         while (true); // stop execution
46     }
47
48     display.clearDisplay();
49     display.setTextSize(1);
50     display.setTextColor(SSD1306_WHITE);
51     display.setCursor(0, 10);
52     display.println("System Ready...");
53     display.display();
54 }
```

```

56 void loop() {
57     int buttonState = digitalRead(BUTTON_PIN);
58
59     // Button pressed (active LOW)
60     if (buttonState == LOW && !buttonPressed) {
61         buttonPressed = true;
62         pressStartTime = millis();
63     }
64
65     // Button released
66     if (buttonState == HIGH && buttonPressed) {
67         buttonPressed = false;
68         unsigned long pressDuration = millis() - pressStartTime;
69
70         if (pressDuration < 1500) {
71             // SHORT PRESS → Toggle LEDs
72             ledsOn = !ledsOn;
73             digitalWrite(GREEN_LED, ledsOn);
74             digitalWrite(BLUE_LED, ledsOn);
75             digitalWrite(PURPLE_LED, ledsOn);
76             showMessage(ledsOn ? "Short Press: LEDs ON" : "Short Press: LEDs OFF");
77             Serial.println(ledsOn ? "LEDs turned ON" : "LEDs turned OFF");
78         } else {
79             // LONG PRESS → Play buzzer tone
80             showMessage("Long Press: Buzzer!");
81             Serial.println("Buzzer Tone Playing...");
82             tone(BUZZER_PIN, 1000); // 1kHz tone
83             unsigned long buzzerStart = millis();
84             while (millis() - buzzerStart < 800) {
85                 // non-blocking: can still check button if needed
86             }

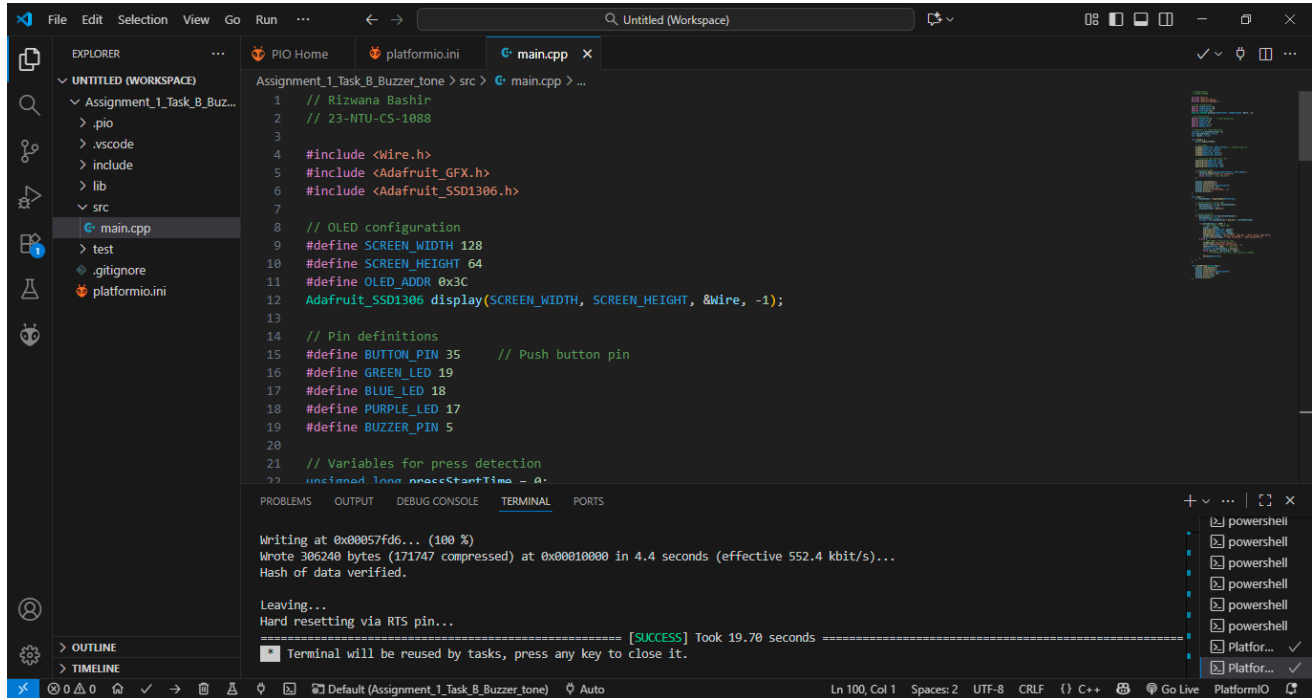
```

```

87         noTone(BUZZER_PIN);
88     }
89 }
90
91
92 void showMessage(String msg) {
93     display.clearDisplay();
94     display.setTextSize(1);
95     display.setTextColor(SSD1306_WHITE);
96     display.setCursor(0, 20);
97     display.println(msg);
98     display.display();
99 }

```

Code Built successfully:



The screenshot displays the PlatformIO IDE interface. The Explorer panel on the left shows the project structure for 'Assignment_1_Task_B_Buzzer_tone'. The main editor window displays the 'main.cpp' file with the following code:

```
1 // Rizwana Bashir
2 // 23-NTU-CS-1088
3
4 #include <Wire.h>
5 #include <Adafruit_GFX.h>
6 #include <Adafruit_SSD1306.h>
7
8 // OLED configuration
9 #define SCREEN_WIDTH 128
10 #define SCREEN_HEIGHT 64
11 #define OLED_ADDR 0x3C
12 Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);
13
14 // Pin definitions
15 #define BUTTON_PIN 35 // Push button pin
16 #define GREEN_LED 19
17 #define BLUE_LED 18
18 #define PURPLE_LED 17
19 #define BUZZER_PIN 5
20
21 // Variables for press detection
22 unsigned long pressStartTime = 0;
```

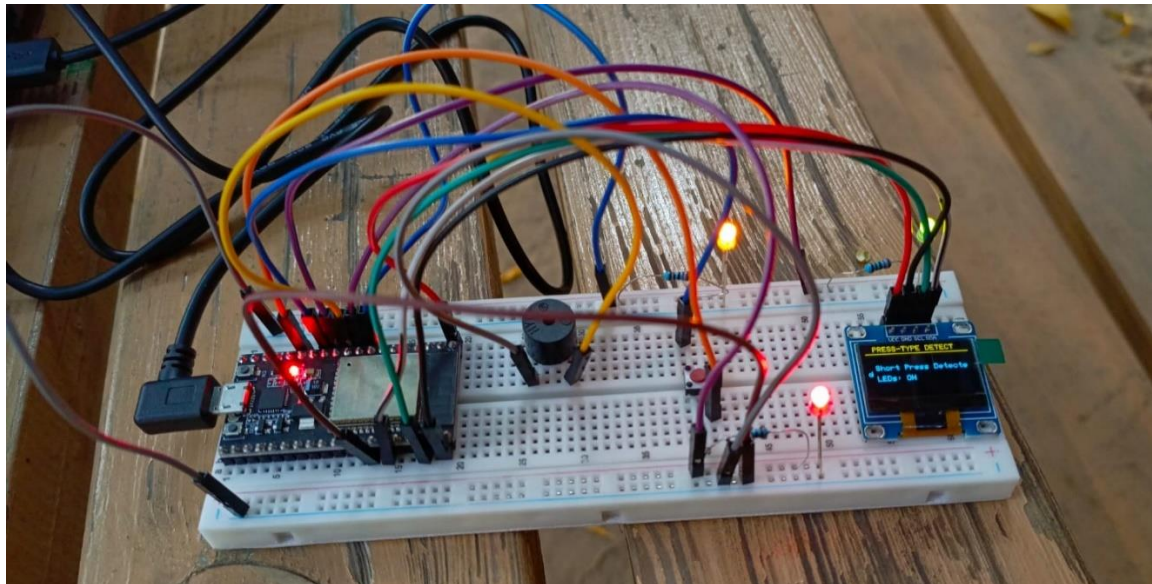
The TERMINAL panel at the bottom shows the successful compilation output:

```
Writing at 0x00057fd6... (100 %)
Wrote 386240 bytes (171747 compressed) at 0x00010000 in 4.4 seconds (effective 552.4 kbit/s)...
Hash of data verified.

Leaving...
Hard resetting via RTS pin...

===== [SUCCESS] Took 19.70 seconds =====
Terminal will be reused by tasks, press any key to close it.
```

Output on Hardware:



Handwritten code and Diagram

Task-B Assignment #1 "Buzzer with Press detection Cycle"

```
#include <Wire.h>
#include <Adafruit-GFX.h>
#include <Adafruit-SSD1306.h>

// OLED Setup
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_ADDR 0x3C
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);

// Pin configuration
#define BUTTON_PIN 3
#define GREEN_LED 19
#define BLUE_LED 18
#define PURPLE_LED 17
#define BUZZER_PIN 5

// Variables for button press detection
unsigned long pressStartTime = 0;
bool buttonPressed = false;
bool ledOn = false;

void setup() {
  Serial.begin(115200);
  pinMode(BUTTON_PIN, INPUT_PULLUP);
```

```
  // Button configuration
  buttonPressed = true;
  pressStartTime = millis();
}
if (pressDuration < 1500) {
  ledOn = !ledOn;
  digitalWrite(GREEN_LED, ledOn);
  digitalWrite(BLUE_LED, ledOn);
  showMessage(ledOn ? "Short press: LEDs ON!" : "Short press: LEDs OFF");
} else {
  showMessage("Long press: Buzzer!");
  Serial.println("Buzzer Tone Playing...");
  tone(BUZZER_PIN, 1000);
  while (millis() - buzzesStart < 800) {
    noTone(BUZZER_PIN);
  }
}
```

```
  // Pin modes
  pinMode(GREEN_LED, OUTPUT);
  pinMode(BLUE_LED, OUTPUT);
  pinMode(PURPLE_LED, OUTPUT);
  pinMode(BUZZER_PIN, OUTPUT);
  // Initially all OFF
  digitalWrite(GREEN_LED, LOW);
  digitalWrite(BLUE_LED, LOW);
  digitalWrite(PURPLE_LED, LOW);
  digitalWrite(BUZZER_PIN, LOW);

  if (!display.begin(SSD1306_SWITCHCAPVCC, OLED_ADDR)) {
    Serial.println("OLED init failed!");
    while (true);
  }
  display.clearDisplay();
  display.setTextSize(1);
  display.setCursor(0, 10);
  display.display();

  void loop() {
    int buttonState = digitalRead(BUTTON_PIN);
    if (buttonState == LOW && !buttonPressed) {
```

```
      void showMessage(String msg) {
        // OLED display messages
        display.clearDisplay();
        display.setTextSize(1);
        display.setTextColor(SSD1306_WHITE);
        display.setCursor(0, 20);
        display.println(msg);
        display.display();
      }
    }
  }
```


Diagram:

