

Project Title: LED Mode Controller with OLED and Buzzer

Name: Ali Faisal

Reg. No: 23-NTU-CS-1010

Board Used: ESP32 Devkit V1

Display Used: SSD1306 OLED (I2C)

Simulation Tool: Wokwi

PURPOSE

This project demonstrates **mode-based LED control** using **buttons**, an **OLED display**, and a **buzzer**.

Each press of the **Mode button** changes how LEDs behave (OFF, Blink, ON, PWM fade), and each action is displayed on the OLED while the buzzer gives feedback sounds.

PIN MAPPING

Component	Pin Name	ESP32 Pin	Connection Type	Description
LED 1 (Green)	LED1	GPIO 17	Digital Output	Turns ON/OFF or blinks alternately
LED 2 (Magenta)	LED2	GPIO 18	Digital Output	Turns ON/OFF or blinks alternately
LED 3 (Blue)	LED3	GPIO 19	PWM Output	Used for smooth brightness fading (PWM)
Mode Button	BTN_MODE	GPIO 26	Input (Pull-up)	Changes mode each time it's pressed
Reset Button	BTN_RESET	GPIO 25	Input (Pull-up)	Resets to "Both OFF" mode
Buzzer	BUZZER	GPIO 15	Digital Output (tone)	Plays short beeps for feedback

Component	Pin Name	ESP32 Pin	Connection Type	Description
OLED Display (SDA)	SDA	GPIO 21	I2C Data	Sends display data
OLED Display (SCL)	SCL	GPIO 22	I2C Clock	Sends display clock signals
OLED Display (VCC)	VCC	3.3V	Power	Supplies power to OLED
OLED Display (GND)	GND	GND	Ground	Common ground reference
LED Common	Cathodes	GND	Ground	LED negative side
Button Ground Pins	BTN1, BTN2 GND sides	GND	Input ground for button press	

HARDWARE SUMMARY

- **3 LEDs (17, 18, 19)** → show different patterns.
 - **2 Buttons (25, 26)** → change or reset mode.
 - **1 Buzzer (15)** → provides feedback with tone.
 - **1 OLED (21, 22)** → displays current mode text.
-

SOFTWARE WORKING LOGIC

1 Setup Phase

- All pins are configured:
 - LEDs as **OUTPUT**
 - Buttons as **INPUT_PULLUP**
 - Buzzer as **OUTPUT**
 - OLED initialized on address **0x3C** (I2C).
 - Default mode → “Both OFF”.
-

2 Mode Button Operation

When **Mode button (GPIO 26)** is pressed:

- The mode number increments ($1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow$ back to 1).
 - A short delay (200 ms) prevents accidental double press (debouncing).
 - Each mode triggers:
 - **OLED update**
 - **Different beep sound**
 - **Different LED behavior**
-

3 Reset Button Operation

When **Reset button (GPIO 25)** is pressed:

- Mode resets to **Mode 1 (Both OFF)**.
 - All LEDs turn OFF.
 - A **low-frequency beep** confirms reset.
 - OLED shows “Reset to OFF”.
-

OPERATING MODES

Mode No.	Mode Name	LED Behavior	OLED Message	Buzzer Frequency
1	Both OFF	All LEDs OFF	“Both OFF”	800 Hz
2	Alternate Blink	LED1 and LED2 blink one after the other every 500ms	“Alternate Blink”	1000 Hz
3	Both ON	LED1 and LED2 remain ON	“Both ON”	1200 Hz
4	PWM Fade	LED3 fades in and out smoothly using PWM	“PWM Fade”	1500 Hz
RESET	Reset to OFF	Resets mode to OFF (Mode 1)	“Reset to OFF”	400 Hz

WORKING CYCLE

1. **Power ON** → Display shows “Both OFF”.
 2. **Press MODE Button** →
 - Changes LED pattern.
 - Displays mode on OLED.
 - Plays buzzer tone.
 3. **Press RESET Button** →
 - All LEDs OFF.
 - Returns to starting mode.
 - “Reset to OFF” message displayed.
-

BUZZER FUNCTION

Each time the mode changes, the buzzer sounds briefly:

- Higher pitch = higher mode.
 - Reset = low pitch.
- This gives instant **audio confirmation** of a mode change.
-

OLED DISPLAY

Displays current mode text:

- “Both OFF”
- “Alternate Blink”
- “Both ON”
- “PWM Fade”
- “Reset to OFF”

This helps visually confirm which mode is active.

```
void beepBuzzer (int freq, int dur) {
    tone (Buzzer, freq, dur);
    delay (dur + 50);
    noTone (Buzzer);
}

void setup () {
    pinMode (led1, OUTPUT);
    pinMode (led2, OUTPUT);
    pinMode (led3, OUTPUT);
    pinMode (BTN_mode, INPUT_PULLUP);
    pinMode (BTN_Reset, INPUT_PULLUP);
    pinMode (Buzzer, OUTPUT);

    display.begin (320, 130, 3, 16, CAP16C, 0x3c);
    display.clearDisplay ();
    display.display ();

    showMsg ("Both OFF");
}

void loop () {
    if (digitalRead (BTN_mode) == low) {
        delay (200);
        mode++;
        if (mode > 4) mode = 1;
    }
}
```

switch (mode) {

case 1:

digitalWrite (led1, low);

digitalWrite (led2, low);

showmsg ("Both OFF");

beepBuzzer (800, 120);

break;

case 2:

showmsg ("Alternate Blink");

beepBuzzer (1000, 120);

break;

case 3:

digitalWrite (led1, High);

digitalWrite (led2, High);

showmsg ("Both ON");

beepBuzzer (200, 200);

break;

case 4:

showmsg ("Pwm fade");

beepBuzzer (1500, 120);

break;

}

```
if (digitalRead(BTN_Reset) == low) {  
    delay(200);  
    mode = 1;  
    digitalWrite(LED1, low);  
    digitalWrite(LED2, low);  
    analogWrite(LED3, 0);  
    showmsg("Reset to OFF");  
    beepbuzzer(Mode, 200);  
  
    if (mode == 2) {  
        if (millis() - premillis >= 500) {  
            premillis = millis();  
            ledstate = !ledstate;  
            digitalWrite(LED1, ledstate);  
            digitalWrite(LED2, !ledstate);  
        }  
    }  
  
    if (mode == 4) {  
        for (int i=0, i<=255 ; i++) {  
            analogWrite(LED3, i);  
            delay(5);  
        }  
        for (int i=255 ; i>=0 ; i--) {  
            analogWrite(LED3, i);  
            delay(5);  
        }  
    }  
}
```



