

**Title: LED Projects with OLED Display**

**Student Name: Minahal Fatima**

**Reg No: 23-NTU-CS-1050**

**Course: Embedded IoT Systems (CSE-3080)**

**Board: ESP32**

## **Abstract**

This assignment demonstrates two embedded IoT projects involving LED control and button interaction using an ESP32 microcontroller.

- **Task A:** Multi-mode LED controller with two buttons (MODE & RESET) and OLED display.
- **Task B:** Single button press detection to identify short and long presses, controlling an LED and buzzer with OLED feedback.

The projects integrate real-time feedback, interrupts, software debouncing, PWM fading, and I2C OLED display interfacing, providing practical experience in embedded system design.

---

## **Introduction:**

Embedded systems are widely used to control devices and provide interactive feedback in real time. Learning to manage **inputs (buttons)**, **outputs (LEDs, buzzer)**, and **displays (OLED)** is essential for IoT applications.

This assignment explores two common scenarios: multiple LED modes controlled by buttons (Task A) and detecting press types with a single button (Task B).

---

## **Objectives:**

### **Task A:**

- Implement 4 LED modes (OFF, alternate blink, both ON, PWM fade).
- Use two buttons (MODE and RESET) to cycle and reset modes.
- Display current mode on OLED.
- Provide buzzer feedback for button interaction.
- Learn interrupts and software debouncing.

### **Task B:**

- Detect short ( $<1.5s$ ) and long ( $\geq 1.5s$ ) button presses.
  - Toggle LED on short press; play buzzer on long press.
  - Display event on OLED.
  - Learn simple event-driven input detection and timing-based debouncing.
- 

## **Hardware Components:**

Component	Quantity	Description
ESP32	1	Microcontroller for processing
LEDs	3 (Task A) + 1 (Task B)	Visual output
Buzzer	1	Audio feedback
Push Buttons	2 (Task A) + 1 (Task B)	Input control
OLED Display	1	Shows mode or event
Jumper wires / Breadboard	As needed	Connections

---

## Pin Configuration:

### Task A – LED Mode Controller

Component	Pin
LED1 (Red)	2
LED2 (Green)	4
LED3 (Blue)	5
Buzzer	27
MODE Button	12

### **Component Pin**

RESET Button 14

OLED SDA 21

OLED SCL 22

### **Task B – Single Button Press Detection**

### **Component Pin**

LED 2

Buzzer 27

Button 12

OLED SDA 21

OLED SCL 22

---

### **Circuit Diagrams:**

- **Task A:** Connect three LEDs, buzzer, MODE and RESET buttons, and OLED to ESP32 as per pin configuration.
- **Task B:** Connect one LED, one buzzer, single button, and OLED to ESP32.

*Insert screenshots from /screenshots folder here.*

- Task A: mode0.png, mode1.png, mode2.png, mode3.png

- Task B: short\_press.png, long\_press.png

**Note:** Screenshots demonstrate the system behavior for different modes and button presses.

---

## **Software Design & Algorithm:**

### **6.1 Task A – LED Mode Controller**

1. Initialize all pins for LEDs, buzzer, buttons, and OLED display.
2. Attach interrupts to MODE and RESET buttons.
3. Detect MODE button press to cycle through 4 modes:
  - **Mode 0:** All LEDs OFF
  - **Mode 1:** Alternate blinking LEDs
  - **Mode 2:** Both LEDs ON
  - **Mode 3:** PWM fade effect on LED3
4. Detect RESET button press to set mode back to 0.
5. Update OLED to show the current mode.
6. Play buzzer tone briefly to confirm button press.
7. Software debouncing ensures correct button recognition.
8. Non-blocking timing is used for blinking and fade effects to maintain system responsiveness.

### **6.2 Task B – Single Button Press Detection**

1. Initialize pins for LED, buzzer, button, and OLED.
  2. Monitor button state continuously.
  3. Measure button press duration:
    - **Short press (<1.5s):** toggle LED ON/OFF
    - **Long press (≥1.5s):** play buzzer tone
  4. Display the detected event on OLED.
  5. Small delay ensures software debouncing and avoids misreading.
- 

## **Observations / Results:**

### **Task A – LED Mode Controller**

Mode LED Behavior		OLED Display
0	All LEDs OFF	Mode 0: Both OFF
1	Alternate blinking	Mode 1: Alternate Blink
2	Both LEDs ON	Mode 2: Both ON
3	PWM Fade on LED3	Mode 3: PWM Fade

### **Task B – Single Button Detection**

## Press Type LED/Buzzer

## OLED Display

Short      LED toggles ON/OFF      Short Press → LED ON/OFF

Long      Buzzer tone 0.6s      Long Press → Buzzer

Screenshots illustrate correct LED patterns, buzzer feedback, and OLED updates for each scenario.

---

## Wokwi Projects

- Task A : <https://wokwi.com/projects/445853794784446465>
  - Task B: <https://wokwi.com/projects/445858202457393153>
- 

## Conclusion

- Task A demonstrates **multi-mode LED control** using interrupts and software debouncing, combined with PWM fading and OLED feedback.
- Task B demonstrates **press-type detection** with a single button, event-driven LED and buzzer response, and OLED display of events.
- Both tasks illustrate core concepts of **embedded IoT systems**, including input/output handling, timing, interrupts, PWM, and display integration.
- The assignment reinforces practical skills for designing **interactive embedded systems**.

