

Project Documentation: Alarm System using ESP32, BLE, PIR Sensor, and OLED Display

Project Overview

This project is an alarm system based on the ESP32 microcontroller, utilizing Bluetooth Low Energy (BLE) for communication with external devices, a PIR sensor for motion detection, and an OLED display for information presentation. The system allows arming and disarming the alarm via a physical button or through a mobile application via BLE. When motion is detected by the PIR sensor, the system activates an LED and displays a corresponding message on the OLED display.

Key Features

1. **Arming and Disarming the Alarm:**
 - Using a physical button.
 - Using a mobile application via BLE (commands "ALARM ON" and "ALARM OFF").
2. **Motion Detection:**
 - The PIR sensor detects motion and notifies the system.
 - When motion is detected, the system turns on the LED and displays a message on the OLED display.
3. **Information Display:**
 - The OLED display shows the current alarm status (armed/disarmed) and information about detected motion.
4. **LED Control:**
 - The LED is turned on when motion is detected and the alarm is armed.

Project Structure

1. System Initialization

- **GPIO Initialization:**
 - Button (GPIO_NUM_4) for arming/disarming the alarm.
 - PIR sensor (GPIO_NUM_18) for motion detection.
 - LED (GPIO_NUM_2) for alarm status indication.
- **OLED Display Initialization:**
 - SSD1306 display connected via the I2C interface.
- **BLE Initialization:**

- Configuration of the BLE server with a GATT service for communication with the mobile application (nRF Connect application for example).

2. Button Handling

- **GPIO Interrupt:**
 - The button is handled by an interrupt service routine (ISR) that detects button presses and sends information to a queue.
- **Main Task:**
 - The button_task receives information from the queue and toggles the alarm state (armed/disarmed).

3. PIR Sensor Handling

- **PIR Task:**
 - The pir_task monitors the PIR sensor state and, upon detecting motion, turns on the LED and displays a message on the OLED display.

4. BLE Communication

- **BLE Server:**
 - The BLE server handles two characteristics:
 - **Read Characteristic:** Returns system status information.
 - **Write Characteristic:** Receives commands from the mobile application (e.g., "ALARM ON", "ALARM OFF")
- **BLE Advertising:**
 - The system advertises itself as a device named "BLE-Server" and is discoverable by mobile applications.

5. OLED Display

- **Information Display:**
 - The OLED display shows the current alarm status and information about detected motion.

Setup Instructions

1. **Hardware Connections:**
 - Connect the button to GPIO_NUM_4.
 - Connect the PIR sensor to GPIO_NUM_18.
 - Connect the LED to GPIO_NUM_2.
 - Connect the OLED display to the appropriate I2C pins (SDA, SCL).
2. **Code Compilation and Upload:**

- Compile the code using ESP-IDF.
- Upload the compiled program to the ESP32.

3. System Startup:

- Upon startup, the system will display "System ready" on the OLED display.
- Use the button to arm or disarm the alarm.
- Use the mobile application to control the alarm and LED via BLE.

Conclusion

This project demonstrates how the ESP32 can be used to create a simple alarm system with a user interface (button, OLED display) and wireless communication (BLE). The system is easy to expand and can be adapted for various applications, such as home or office monitoring.

Example photos



