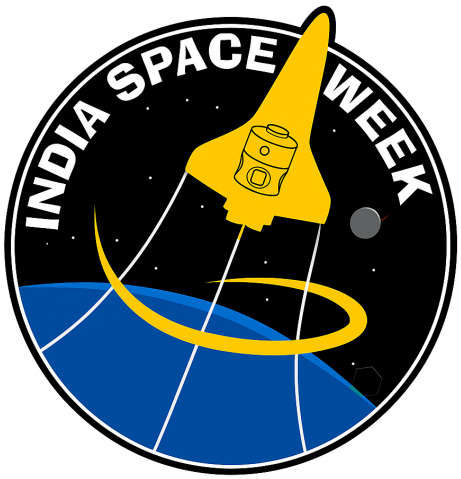

Smart Motion Sentinel 1.0

An IoT-Based Intrusion Detection and Monitoring System

Submitted as part of the

India Space Lab – Summer Internship Program 2025

Submitted by:

Samarth Trivedi
Department of Embedded Systems & IoT
IIITDM Kancheepuram

Date: July 2025 **Location:** Delhi

CERTIFICATE

This is to certify that the report titled **"Smart Motion Sentinel 1.0: An IoT-Based Intrusion Detection and Monitoring System"**, submitted by **Samarth Trivedi**, is a bona fide record of the project work carried out during the **India Space Lab – Summer Internship Program 2025**.

The project has been carried out under our guidance and supervision, and it is certified that this work has not been submitted anywhere else for any academic or professional purpose.

Director
India Space Lab

Place: Delhi **Date:** _____

Contents

Abstract

Smart Motion Sentinel 1.0 is a lightweight and robust intrusion detection and monitoring system based on the ESP32 microcontroller. Using a PIR motion sensor, an OLED display, a buzzer, and a red LED, the system detects unauthorized motion, instantly alerts users, and visually logs the events. Developed during the India Space Lab Summer Internship 2025, this project demonstrates practical IoT security with affordable, accessible hardware. The report details hardware and software design, integration challenges, system testing, applications, and future scope, illustrating how IoT solutions can secure critical environments with simplicity and reliability.

1 Introduction

1.1 Background

Security in sensitive environments—such as research labs, data centers, and cleanrooms—has become increasingly important due to the rise in automation and high-value assets. The evolution of the Internet of Things (IoT) has made it possible to design smarter, more efficient, and more cost-effective security solutions using embedded systems and networked devices.

1.2 Project Motivation and Objectives

The primary motivation for this project was to develop a compact, affordable, and efficient motion detection and alert system leveraging IoT and embedded design principles. Key objectives included:

- Integrating common sensors and actuators with the ESP32 microcontroller.
- Providing instant local feedback (audio, visual, and display) upon intrusion detection.
- Creating a system that is reliable, easy to reproduce, and scalable for advanced features.
- Gaining hands-on experience with real-world IoT security challenges during the India Space Lab internship.

2 System Overview

2.1 System Architecture

The Smart Motion Sentinel 1.0 is structured around four main functional units:

1. **Sensing:** The PIR sensor detects human movement in its range.
2. **Processing:** The ESP32 receives input from the PIR sensor and processes motion events.
3. **Alerting:** The system activates a buzzer and LED as soon as motion is detected.
4. **Display and Logging:** An OLED module displays system status and event counts; serial output logs each event for monitoring.

2.2 Circuit Schematic

Figure 1 shows the complete wiring and integration of all system components.

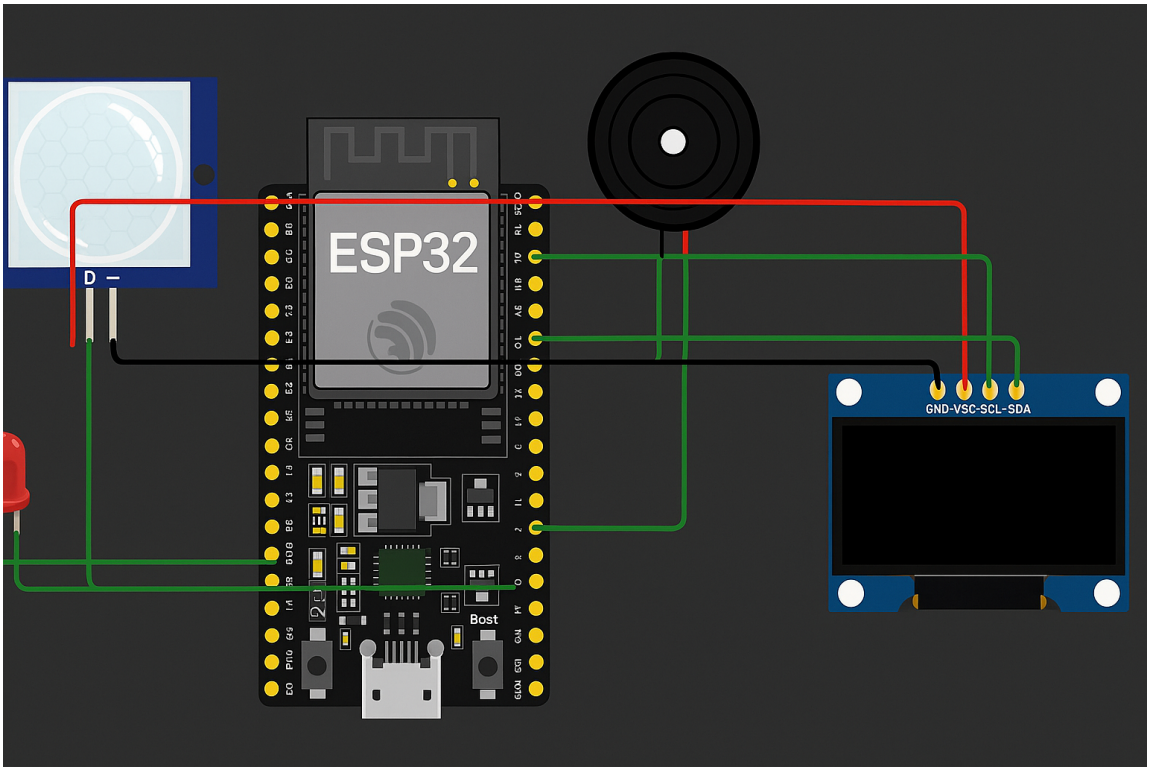

Figure 1: Circuit diagram of Smart Motion Sentinel 1.0: All major components (PIR sensor, LED, buzzer, OLED, ESP32) are connected for efficient real-time monitoring and alerts.

2.3 ESP32 DevKit V4 Board

The ESP32 DevKit V4 board (Figure 2) acts as the brain of the system, coordinating sensor input, event logic, and all alert/display outputs.

Figure 2: ESP32 DevKit V4 used as the main controller, providing multiple GPIOs, Wi-Fi/Bluetooth capabilities, and robust performance for embedded IoT applications.

3 Hardware Components

Component	Description
ESP32 DevKit V4	Powerful microcontroller with integrated Wi-Fi and Bluetooth, used as the main processing and control unit.
PIR Motion Sensor	Detects changes in infrared radiation to sense human presence.
SSD1306 OLED Display	128x64 pixel I2C module for visual display of system status, alerts, and motion event count.
Active Buzzer	Emits audible sound during motion detection for immediate alert.
Red LED	Provides a visible indicator when intrusion/-motion is detected.
Breadboard and Wires	Enables rapid prototyping, testing, and easy modifications.

3.1 Component Highlights

- **ESP32:** Dual-core, low power, multiple digital/analog pins, widely supported in IoT.
- **PIR Sensor:** Highly sensitive to human body movement, immune to noise from non-infrared sources.
- **OLED Display:** Clear, low-power feedback; essential for local user interface.
- **Buzzer & LED:** Dual signaling ensures alert is noticed both visually and audibly.

4 Working Principle and Methodology

The working of Smart Motion Sentinel 1.0 follows a simple yet effective sequence:

- The PIR sensor monitors the area continuously.
- When movement is detected, it sends a HIGH signal to the ESP32.
- The ESP32 immediately:
  - Turns on the red LED.
  - Activates the buzzer for an audible alert.
  - Updates the OLED display with "Motion Detected!" and increments the event count.
  - Logs the event through the serial interface.
- When no motion is detected, the system status is updated to "Area Secure", the LED and buzzer are deactivated, and both OLED/serial are updated accordingly.

5 ESP32 Pin Mapping

Component	**Function**	**ESP32 Pin**
PIR Sensor	OUT	GPIO 13
PIR Sensor	VCC	3.3V
PIR Sensor	GND	GND
Buzzer	Positive (+)	GPIO 4
Buzzer	Ground	GND
LED	Anode	GPIO 2
LED	Cathode	GND
OLED VCC	Power	3.3V
OLED GND	Ground	GND
OLED SDA	I2C Data	GPIO 21
OLED SCL	I2C Clock	GPIO 22

6 Libraries and Software Tools

6.1 Libraries Used

- **Adafruit GFX Library**: For graphics/text rendering on OLED.
- **Adafruit SSD1306 Library**: For controlling the SSD1306 OLED display.
- **Wire**: Standard Arduino I2C communication library.

6.2 Development Environment

- **Arduino IDE**: Used for ESP32 programming.
- **Wokwi Simulator**: Used for early-stage simulation and debugging.

7 Software Implementation

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

#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_RESET -1

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET
    );

#define PIR_PIN     13
#define LED_PIN     2
#define BUZZER_PIN  4

int motionCount = 0;
bool motionPreviouslyDetected = false;

void setup() {
  pinMode(PIR_PIN, INPUT);
  pinMode(LED_PIN, OUTPUT);
  pinMode(BUZZER_PIN, OUTPUT);
  Serial.begin(115200);

  if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
    Serial.println("OLED initialization failed");
    while (true);
  }

  display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.setCursor(0, 0);
  display.println("Smart Motion Sentinel");
  display.display();
  delay(1500);
  display.clearDisplay();
}
```

```
void loop() {
  int motion = digitalRead(PIR_PIN);

  if (motion == HIGH) {
    if (!motionPreviouslyDetected) {
      motionCount++;
      Serial.print("Motion␣Detected!␣Count:␣");
      Serial.println(motionCount);
      motionPreviouslyDetected = true;
    }

    digitalWrite(LED_PIN, HIGH);
    digitalWrite(BUZZER_PIN, HIGH);
    showStatus("Motion␣Detected", motionCount);
  } else {
    if (motionPreviouslyDetected) {
      Serial.println("Motion␣ended.");
    }

    digitalWrite(LED_PIN, LOW);
    digitalWrite(BUZZER_PIN, LOW);
    showStatus("No␣Motion", motionCount);
    motionPreviouslyDetected = false;
  }

  delay(200);
}

void showStatus(String status, int count) {
  display.clearDisplay();
  display.setCursor(0, 0);
  display.setTextSize(1);
  display.println("Smart␣Motion␣Sentinel");
  display.setCursor(0, 20);
  display.print("Status:␣");
  display.println(status);
  display.setCursor(0, 40);
  display.print("Count:␣");
  display.println(count);
  display.display();
}
```

Listing 1: Smart Motion Sentinel Arduino Code

8 Wokwi JSON Circuit Configuration

```
{
  "version": 1,
  "author": "Samarth␣Trivedi",
  "editor": "wokwi",
  "parts": [
    { "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 0, "left":
        0, "attrs": {} },
    { "type": "wokwi-pir-motion-sensor", "id": "pir1", "top": -15.2, "
       left": -103.38, "attrs": {} },
    { "type": "wokwi-led", "id": "led1", "top": 102, "left": -111.4, "
       attrs": { "color": "red" } },
    { "type": "wokwi-buzzer", "id": "bz1", "top": -36, "left": 117, "
       attrs": { "volume": "0.1" } },
    { "type": "board-ssd1306", "id": "oled1", "top": 89.54, "left":
       201.83, "attrs": { "i2cAddress": "0x3c" } }
  ],
  "connections": [
    [ "esp:TX", "$serialMonitor:RX", "", [] ],
    [ "esp:RX", "$serialMonitor:TX", "", [] ],
    [ "pir1:VCC", "esp:3V3", "red", [ "v0" ] ],
    [ "pir1:OUT", "esp:13", "green", [ "v0" ] ],
    [ "pir1:GND", "esp:GND.3", "black", [ "v0" ] ],
    [ "led1:A", "esp:2", "green", [ "v0" ] ],
    [ "led1:C", "esp:GND.1", "green", [ "v0" ] ],
    [ "bz1:2", "esp:4", "green", [ "v0" ] ],
    [ "bz1:1", "esp:GND.3", "green", [ "v0" ] ],
    [ "oled1:GND", "esp:GND.3", "black", [ "v0" ] ],
    [ "oled1:VCC", "esp:3V3", "red", [ "v0" ] ],
    [ "oled1:SCL", "esp:22", "green", [ "v0" ] ],
    [ "oled1:SDA", "esp:21", "green", [ "v0" ] ]
  ],
  "dependencies": {}
}
```

Listing 2: Wokwi Configuration

9 Testing, Results, and Validation

9.1 Testing Methodology

Testing included:

- **Virtual Simulation:** Using Wokwi, initial debugging, and logic verification.

- **Physical Testing:** Breadboard implementation, motion detection accuracy, and response time checks.
- **System Reliability:** Continuous operation to assess stability and false positives.
- **User Feedback:** Observed system responsiveness to human movement and the clarity of alerts/display.

9.2 Observations and Results

- The system detected motion quickly and triggered both audio and visual alarms instantly.
- The OLED displayed status and count clearly; serial logs were accurate and consistent.
- The system was stable during prolonged use and demonstrated robustness against noise and power fluctuations.
- The modularity of the hardware allows for future expansion (e.g., adding wireless notification or logging features).

10 Learning Journey with India Space Lab

This internship provided a rich, hands-on experience across multiple advanced domains:

1. **Advanced Drone Technology (Air Taxi):** Explored UAV systems and the importance of real-time feedback, inspiring robust event monitoring in Sentinel 1.0.
2. **CanSat and CubeSat (Student Satellite):** Learned the importance of compact, reliable system design, directly influencing the project's focus on small-form-factor, reliable operation.
3. **Rocketry Training Program:** Gained practical knowledge of sensor telemetry and real-time data, essential for IoT monitoring.
4. **Remote Sensing (GIS):** Understood applications of automated security and monitoring, a natural fit for IoT-based intrusion detection.
5. **Space Astronomy and Entrepreneurship Sessions:** Broadened understanding of translating technical projects into real-world solutions.
6. **Project Work:** The culmination of the above experiences was the development of Smart Motion Sentinel 1.0, blending hardware, software, and system integration skills.

11 Applications and Future Scope

11.1 Practical Applications

- **Smart Homes:** As a motion-activated security solution for homeowners.
- **Hospital Security:** Protecting sensitive zones and equipment from unauthorized access.
- **Cleanrooms and Labs:** Ensuring strict access protocols in contamination-sensitive environments.
- **Airport and Data Center Surveillance:** Quick detection of breaches in high-value areas.

11.2 Scope for Future Enhancement

- Integrate Wi-Fi/Bluetooth for remote alerts or cloud-based logging.
- Expand to multi-sensor setups for large area coverage.
- Add environmental sensors for comprehensive monitoring (temperature, humidity, smoke).
- Enhance user interface, e.g., touch screen display or web dashboard.

12 Conclusion

This project demonstrates that an affordable, reliable, and expandable intrusion detection system can be built using ESP32 and basic peripherals. The design philosophy emphasizes simplicity, modularity, and real-world utility. The learning and practical exposure gained through the India Space Lab internship were instrumental in successfully realizing this project, paving the way for more advanced security and IoT solutions in the future.

Acknowledgments

I express my gratitude to the mentors at India Space Lab for their encouragement and technical guidance. My thanks to the Department of Embedded Systems & IoT, IIITDM Kancheepuram, for academic support and providing an innovative ecosystem. The opportunity to work on real hardware and participate in workshops has been invaluable in my learning journey.

References

- ESP32 Datasheet – Espressif
- Adafruit OLED Guide – Installing SSD1306 & GFX libraries
- Random Nerds OLED Guide – Wiring and Examples
- Wokwi Simulator Platform
- Workshop material and notes from India Space Lab Summer Internship 2025