International Islamic University, Islamabad

Faculty of Engineering and Technology

Department of Electrical and Computer Engineering

Project Proposal: GSM-Enabled Intrusion Detection for Remote Security Monitoring

Project Name: HomeGuard+

Project Members: Muhammad Yaseen

Abdul Moeed Tahir

Roll No: 21-FET/BSCE/F-22

13-FET/BSCE/F-22

Submitted To : Engr. Asad Ehsan

1. Project Title:

GSM-Enabled Intrusion Detection for Remote Security Monitoring

2. Project Objective:

The main objective of this project is to develop a robust and reliable Intrusion Detection System (IDS) that notifies the user about unauthorized access in a specific area. The system will use a Passive Infrared (PIR) Sensor for detecting motion (human presence or movement) and an Arduino microcontroller to process the sensor data. When the sensor detects motion, the system will trigger an alert mechanism, which will notify the user by sending a phone call via GSM module (SIM800L). The system is intended for use in security applications such as homes, offices, or restricted areas.

3. System Components:

- **Arduino Microcontroller**
- **PIR Sensor (Passive Infrared Sensor)**.
- **SIM800L GSM Module**
- **Buzzer**
- **Red LED**
- **Voltage Step-Down Converter**

4. Project Overview:

This project uses a combination of sensors and communication technologies to build an intelligent intrusion detection system. When the PIR sensor detects motion, it triggers the Arduino to activate both an audible alarm (buzzer) and a visual indicator (LED). Additionally, the system will send a phone call to a pre-programmed mobile number using the SIM800L GSM module, notifying the user of potential intrusion.

- **Step 1:** PIR sensor detects motion.
- **Step 2:** Arduino activates buzzer and LED.
- **Step 3:** SIM800L GSM module initiates a phone call.
- **Step 4:** After 5 seconds, the call is terminated, and the system resets.

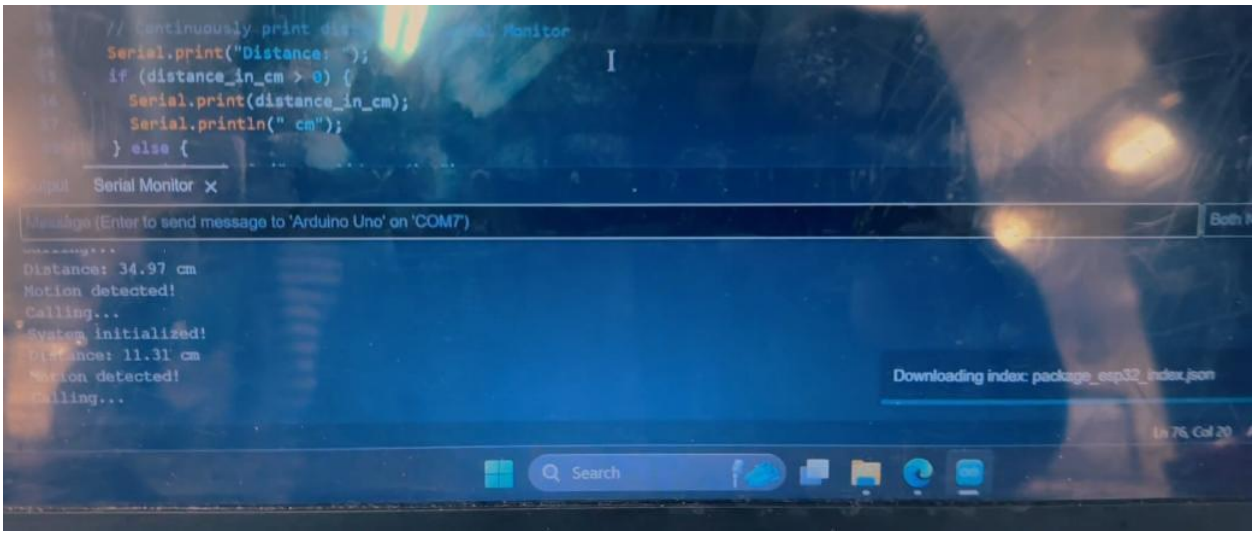
6. Code:

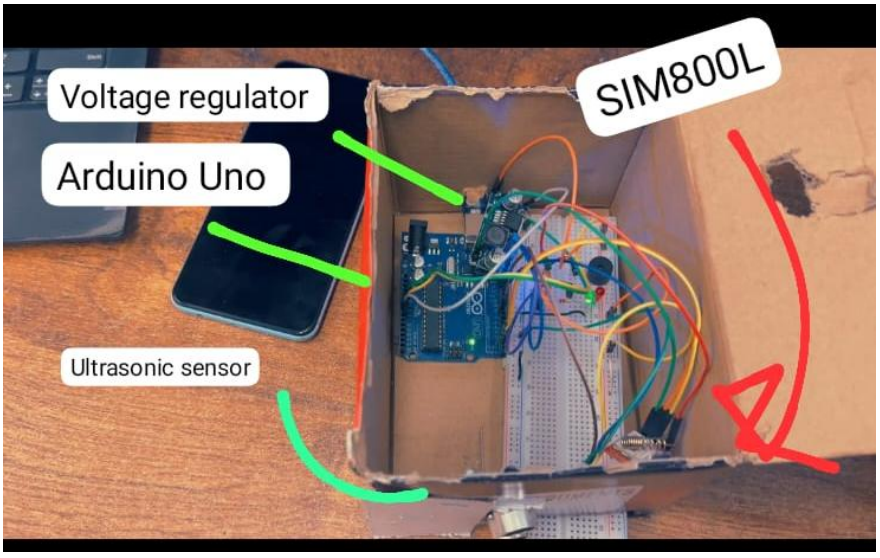
```
#include <SoftwareSerial.h>
SoftwareSerial mySerial(3, 2); // GSM module connected to pins 3 (TX) and 2 (RX)
#define trigPin 8
#define echoPin 9
int redled = 13;
int buzzer = 12;

void setup() {
  Serial.begin(9600);
  mySerial.begin(9600); // For SIM900A
  Serial.println("System initialized!");

  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(redled, OUTPUT);
  pinMode(buzzer, OUTPUT);

  digitalWrite(redled, LOW);
  digitalWrite(buzzer, LOW);
}

float getFilteredDistance() {
  float totalDistance = 0;
  int validReadings = 0;

  for (int i = 0; i < 5; i++) { // Take 5 readings for smoothing
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);

    long time_duration = pulseIn(echoPin, HIGH);

    // Convert time to distance
    float distance = time_duration * 0.034 / 2;

    // Filter out invalid readings
    if (time_duration > 0 && distance >= 2 && distance <= 400) { // Valid range:
2 cm to 400 cm
      totalDistance += distance;
      validReadings++;
    }
    delay(10); // Small delay between readings
  }

  // Return the average of valid readings, or 0 if no valid readings
  return (validReadings > 0) ? (totalDistance / validReadings) : 0;
```

```
}

void loop() {
  float distance_in_cm = getFilteredDistance();

  // Continuously print distance to Serial Monitor
  Serial.print("Distance: ");
  if (distance_in_cm > 0) {
    Serial.print(distance_in_cm);
    Serial.println(" cm");
  } else {
    Serial.println("No valid reading");
  }

  // Check for motion (distance >= 5 cm and valid)
  if (distance_in_cm >= 5 && distance_in_cm <= 100) { // Valid range: 5 cm to 100
cm
    digitalWrite(redled, HIGH);
    digitalWrite(buzzer, HIGH); // Activate buzzer
    Serial.println("Motion detected!");
    Serial.println("Calling...");
    delay(5000);
    digitalWrite(buzzer, LOW);
    digitalWrite(redled, LOW);
    // Send AT commands to GSM module
    mySerial.println("AT"); // Handshake test
    updateSerial();

    mySerial.println("ATD+923191380568;"); // Dial the number
    updateSerial();

    delay(2000); // Wait for the call duration
    mySerial.println("ATH"); // Hang up
    updateSerial();
  } else {
    digitalWrite(redled, LOW);
    digitalWrite(buzzer, LOW); // Deactivate buzzer
  }

  // Add a small delay to prevent overwhelming the Serial Monitor
  delay(200);
}

void updateSerial() {
  while (Serial.available()) {
    mySerial.write(Serial.read()); // Forward data to GSM module
  }
  while (mySerial.available()) {
    Serial.write(mySerial.read()); // Forward data to Serial Monitor
```

```
  }
}
```

7. Visuals:


Serial.print("Distance: ");
if (distance_in_cm > 0) {
Serial.print(distance_in_cm);
Serial.println(" cm");
} else {
Serial Monitor
Message (Enter to send message to 'Arduino Uno' on 'COM7')
Distance: 34.97 cm
Motion detected!
Calling...
System initialized!
Motion detected!
Calling...
Downloading index: package_esp32_index.json
Search


Voltage regulator
Arduino Uno
SIM800L
Ultrasonic sensor

8. Working Principle:

- **PIR Sensor Detection:** Detects motion through infrared radiation changes.
- **Arduino Activation:** When motion is detected, Arduino activates the buzzer and LED, and sends AT commands to the SIM800L GSM module to dial a number.
- **SIM800L GSM Module:** Dials a pre-programmed number and establishes a call. After 20 seconds, the call is terminated.
- **Voltage Step-Down Module:** Provides stable power to the SIM800L module.
- **Program:** The system is controlled by a simple Arduino program that utilizes Software Serial for communication between the Arduino and the SIM800L GSM module.

9. Proposed Workflow:

1. **Initialization:** Arduino starts, and system waits for PIR sensor input.
2. **Motion Detection:** PIR sensor detects motion, and Arduino activates buzzer/LED.
3. **Call Notification:** Arduino sends AT command to SIM800L to make a call.
4. **Call Termination:** After 20 seconds, the call disconnects, and the system resets.
5. **Reset:** System waits for the next detection.

10. Applications:

- **Home Security:** Monitor entrances and backyards for unauthorized entry.
- **Office Security:** Monitor restricted areas or sensitive rooms.
- **Industrial Security:** Secure warehouses and factories.
- **Automated Security Systems:** Part of an integrated security solution with remote monitoring.

11. Challenges and Future Scope:

- **Power Supply:** Ensure reliable power, possibly through solar or battery.
- **SMS Alerts:** Future versions could send SMS alerts for more detailed notifications.
- **Multiple Sensors:** Network multiple PIR sensors for better coverage.
- **Cloud Integration:** Implement cloud-based monitoring for real-time analysis.

12. Conclusion:

This Intrusion Detection System offers a practical solution for real-time security monitoring, providing immediate alerts through phone calls. The combination of a PIR sensor, Arduino, and SIM800L GSM module allows the system to function autonomously, making it suitable for both residential and commercial security applications. The project demonstrates the integration of microprocessor interfacing with sensors and communication systems, offering a reliable and affordable security solution.