Department of Electrical and Electronics Engineering

Third-Year B. Tech. (ECE/ECE AI-ML)

Robotics and Automation
ECE3002B

PBL Report: “Raspberry-Pi Based House Alarm System”

Madhura Patwardhan
1032221059 (ECE-AIML)

Nidhi Katwe
1032220498 (ECE-AIML)

Annie D’Souza
1032220722 (ECE-AIML)

Prisha Waikar
1032221022 (ECE-CORE)

Since 1983

Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

MIT-WPU
।। विश्वशान्तिर्ध्रुवं ध्रुवा ।।

ACKNOWLEDGEMENT

We would like to express our deepest gratitude to everyone who contributed to completing this report.

First, we are grateful to Prof. Dr Pooja Gundewar and Prof. Manisha Wani for their invaluable guidance, insights, and continuous support throughout this project. Their encouragement and expertise have been instrumental in shaping the direction and quality of this work.

We also sincerely thank MIT WPU for providing us with the necessary resources and facilities to carry out our research effectively.

Finally, We are immensely grateful to our family, friends, and colleagues for their unwavering support, understanding, and patience throughout this endeavour.

Thank you all for your support and guidance.

TABLE OF CONTENTS

PROBLEM STATEMENT

Problem Statement:

Development of a Cost-Effective, Real-Time Surveillance System for Enhanced Security in Residential and Commercial Spaces.

In today's landscape, security systems are essential for safeguarding residential and commercial spaces. However, existing security systems often involve high costs, limited flexibility, and a lack of real-time response capabilities.

This project seeks to address these challenges by developing a cost-effective, efficient, and customizable surveillance system using the Raspberry Pi as the core controller. Integrating a PIR motion sensor, Pi camera, LED light, buzzer, and smoke detector sensor, this system provides real-time monitoring and SMS alerts in response to motion or smoke detection.

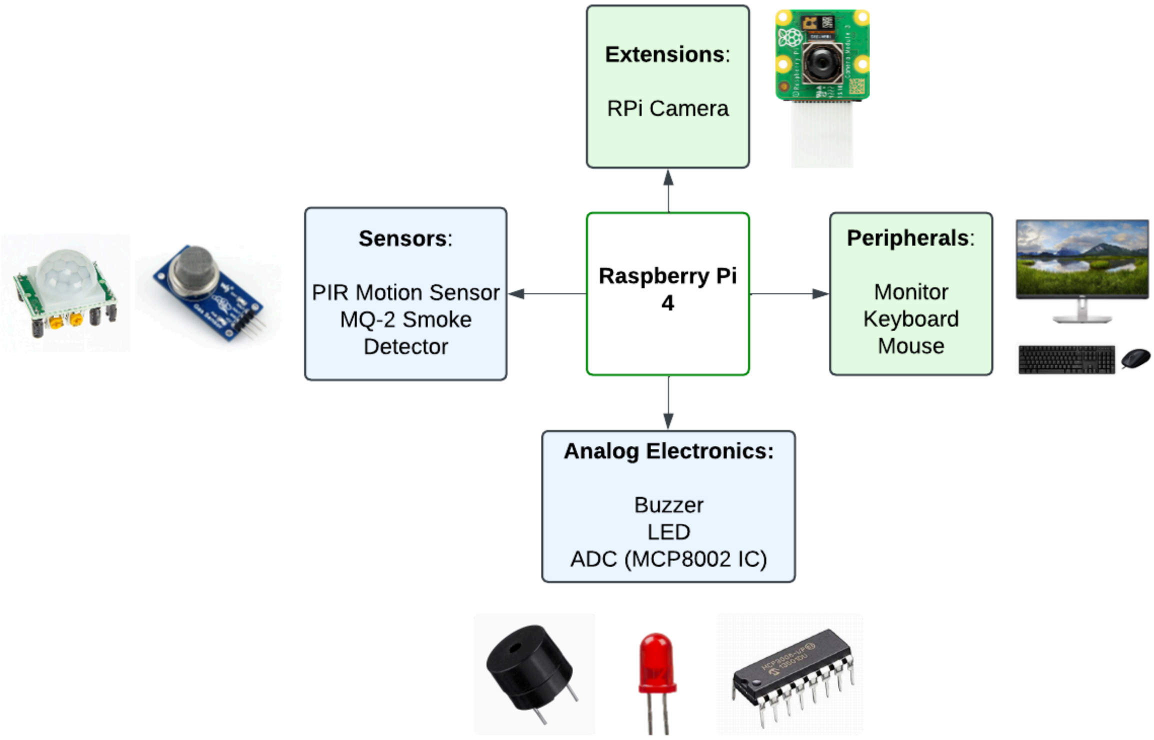
he goal is to create a scalable solution that enhances safety for homes and businesses by combining motion detection, video surveillance, smoke detection, and immediate alert notifications.

OBJECTIVE

1. Develop a security system that utilises a PIR motion sensor to detect motion and a Pi camera to capture video footage or images in response to detected motion.
2. Integrate a smoke detector sensor to facilitate early detection of fire hazards within the monitored area.
3. Incorporate LED lights and buzzers as alarm mechanisms to activate upon detection of motion or smoke, providing immediate visual and auditory alerts.
4. Implement a real-time SMS notification system to promptly alert the user via mobile phone in case of a security breach or fire detection, ensuring timely response and mitigation.

METHODOLOGY

Block Diagram



Wiring Connections

MQ2 Sensor to **MCP3008**:

VCC (MQ2) → 5V (Raspberry Pi)
GND (MQ2) → GND (Raspberry Pi)
AO (MQ2) → Channel 0 (CH0) on MCP3008 (analog output to MCP3008)
DO (MQ2) → Not used in this setup (digital output not required)

MCP3008 to **Raspberry Pi:**

VDD → 3.3V (Raspberry Pi)
VREF → 3.3V (Raspberry Pi)
AGND → GND (Raspberry Pi)
DGND → GND (Raspberry Pi)
DIN → GPIO 10 (MOSI on Raspberry Pi, pin 19)
DOUT → GPIO 9 (MISO on Raspberry Pi, pin 21)
CLK → GPIO 11 (SCLK on Raspberry Pi, pin 23)
CS/SHDN → GPIO 8 (CE0 on Raspberry Pi, pin 24)

PIR Motion Sensor:

VCC (PIR) → 5V (Raspberry Pi)
OUT (PIR) → GPIO 21 (Raspberry Pi)
GND (PIR) → GND (Raspberry Pi)

Other Components:

LED → GPIO 12 (with a resistor in series)

Buzzer → GPIO 17

Libraries Used:

1. sudo update
2. sudo upgrade
3. sudo apt-get install python3-gpac
4. sudo apt-get install python3-rpi.gpio
5. sudo apt-get install python3-spidev
6. sudo apt-get install libcamera-apps

Our Tech Stack (Softwares Used)

1. SD card formatter
2. Official Raspbian imager
3. Pushbullet (for alerts)

SPECIFICATIONS OF COMPONENTS

The components used in the project are:

1. Raspberry Pi 4:
-Expected Output Voltage: 3.3V GPIO output pins; 5V power output on designated pins

2. Raspberry Pi camera:
-Gain: Not typically specified (adjustable through software)
-Expected Output Voltage: Digital video data output to Raspberry Pi

3. Power supply C-type:
-Threshold Voltage: 5V
-Expected Output Voltage: 5V

4. Micro HDMI to HDMI cable:
-Expected Output Voltage: Dependent on the HDMI signal

5. Keyboard + mouse (wired):
-Threshold Voltage: 5V

6. SD card adapter

7. SD card - 32 GB

8. Breadboard:
-Expected Output Voltage: Based on connected components (commonly 3.3V or 5V)

9. LED:
-Threshold Voltage: Typically 2V-3.2V (depends on colour)
-Expected Output Voltage: Light emission at 2V-3.2V with a typical current of 20mA

10. Buzzer:
-Threshold Voltage: ~3V
-Expected Output Voltage: Sound output at 85dB when supplied with 3-5V DC

11. PIR motion sensor:
-Threshold Voltage: ~3.3V (on output pin for motion detection)
-Expected Output Voltage: 0V (no motion) to 3.3V (motion detected)

12. MQ-2 gas sensor:
-Threshold Voltage: ~2.5V for gas detection initiation (dependent on gas concentration)
-Gain: Variable based on sensitivity settings (range adjustment needed)
-Expected Output Voltage: 0V to 5V (analog output, varies with gas concentration)

13. ADC (MCP3008 IC):
-Threshold Voltage: Dependent on the reference voltage (typically 2.7V - 5V)
-Expected Output Voltage: Digital output to SPI (up to 10-bit resolution for 0-1023 levels)

14. Jumper wires:

-Expected Output Voltage: Dependent on connected components

ALGORITHM / CODE

```python
import RPi.GPIO as GPIO
import time
import subprocess
import datetime
import spidev  # SPI library for ADC communication

# Setup SPI for MCP3008
spi = spidev.SpiDev()
spi.open(0, 0)
spi.max_speed_hz = 1350000

# GPIO setup
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)

# GPIO Pins Configuration
PIR_PIN = 21     # PIR Motion Sensor
LED_PIN = 12     # LED
BUZZER_PIN = 17 # Buzzer
GAS_THRESHOLD = 300  # Set this to the gas level threshold you want to detect

# Setup GPIO pins
GPIO.setup(PIR_PIN, GPIO.IN)       # PIR sensor as input
GPIO.setup(LED_PIN, GPIO.OUT)      # LED as output
GPIO.setup(BUZZER_PIN, GPIO.OUT)  # Buzzer as output

# Function to read from MCP3008
def read_adc(channel):
    adc = spi.xfer2([1, (8 + channel) << 4, 0])
    data = ((adc[1] & 3) << 8) + adc[2]
    return data
```

```
# Function to send Pushbullet notification
def send_pushbullet_notification(message):
    API_KEY = "YOURKEY"  # Replace with your actual Pushbullet API key
    subprocess.run([
        "curl",
        "-u", f"{API_KEY}:",
        "-X", "POST",
        "https://api.pushbullet.com/v2/pushes",
        "-d", "type=note",
        "-d", "title=Alert",
        "-d", f"body={message}"
    ])

# Function to record video
def record_video(duration=10):
    timestamp = datetime.datetime.now().strftime("%Y%m%d_%H%M%S")
     video_file = f"/home/madhura/vid/motion_{timestamp}.h264"   # Change the path
as needed
    subprocess.run([
        "libcamera-vid",
        "-t", str(duration * 1000),  # duration in milliseconds
        "-o", video_file
    ])
    print(f"Video recorded: {video_file}")

# Initialize LED and Buzzer to be off
GPIO.output(LED_PIN, GPIO.LOW)
GPIO.output(BUZZER_PIN, GPIO.LOW)

# Main loop
try:
    print("Motion and Gas Detector is running...")
    while True:
        motion_detected = GPIO.input(PIR_PIN)
        gas_level_analog = read_adc(0)
# Read MQ2 analog value from ADC channel 0

        # Motion Detection
        if motion_detected:
            GPIO.output(LED_PIN, GPIO.HIGH)
            GPIO.output(BUZZER_PIN, GPIO.HIGH)
            print('Motion Detected')
            send_pushbullet_notification("Alert: Motion Detected.")
            record_video(duration=10)  # Record video for 10 seconds
            time.sleep(2)  # Keep LED and Buzzer on for 2 seconds
            GPIO.output(LED_PIN, GPIO.LOW)
            GPIO.output(BUZZER_PIN, GPIO.LOW)
```

```python
        # Gas Detection with Analog Threshold
        if gas_level_analog > GAS_THRESHOLD:
            GPIO.output(LED_PIN, GPIO.HIGH)
            GPIO.output(BUZZER_PIN, GPIO.HIGH)
            print(f'Gas Detected! Analog Value: {gas_level_analog}')
                      send_pushbullet_notification(f"Alert:  Gas  Detected!  Level: 
{gas_level_analog}")
            record_video(duration=10)  # Record video for 10 seconds
            time.sleep(2)  # Alert duration
            GPIO.output(LED_PIN, GPIO.LOW)
            GPIO.output(BUZZER_PIN, GPIO.LOW)
        # Short delay to prevent excessive CPU usage
        time.sleep(0.1)

except KeyboardInterrupt:
    print("\nExiting program...")
    GPIO.cleanup()
    spi.close()  # Close SPI connection on exit
```

RESULT

The Raspberry Pi-based surveillance security system successfully achieved its project objectives, summarized as follows:

1. Motion Detection and Video Surveillance:
The PIR motion sensor accurately detected human motion, triggering the Pi Camera to capture video or still images, which were stored for later review. The camera delivered high-quality outputs under standard lighting.

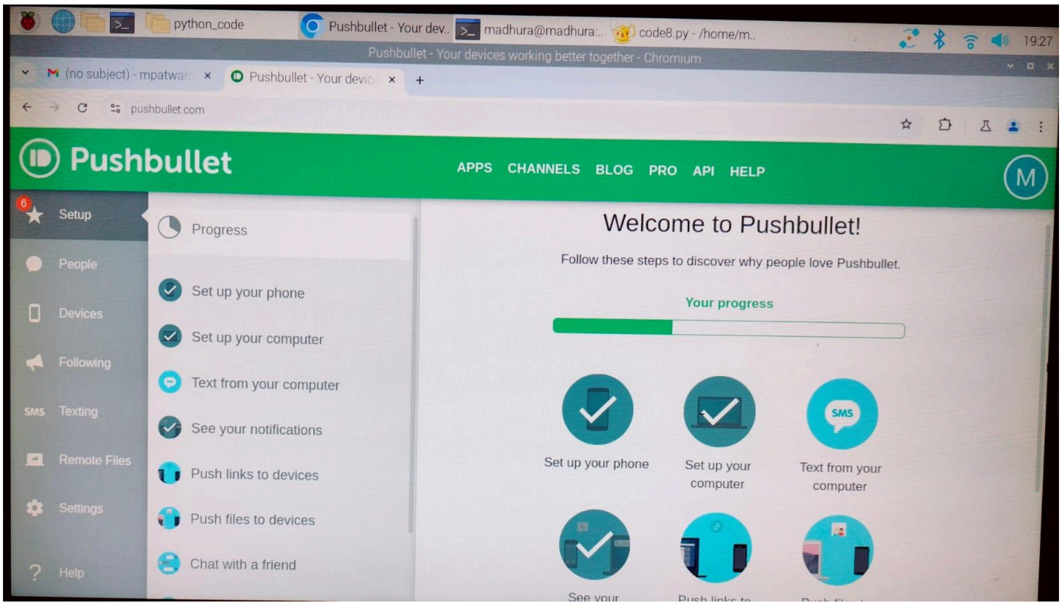
2. Fire Detection:
The smoke detector sensor identified smoke or fire hazards, sending a signal to the Raspberry Pi, which activated the alert system for immediate responses.

3. Alarm System:
The LED light and buzzer are activated correctly upon detecting motion or smoke, providing visual and auditory alerts to ensure rapid response to potential threats.

4. SMS Notification:
An SMS module was integrated to send real-time alerts to the user's phone upon detecting motion or smoke, ensuring swift notification of security threats or fire hazards.

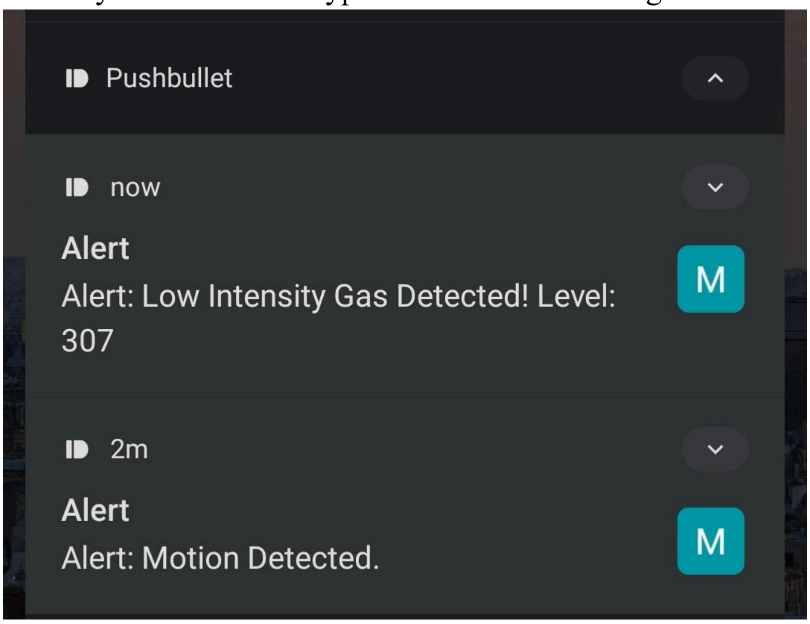
5. System Reliability:
The Raspberry Pi 4 acted as a reliable central controller, managing sensor inputs and outputs with minimal delay and consistent performance during testing.

Pushbullet extension on Raspberry Pi 4 OS:



Pushbullet notification alert on mobile:

Smoke Detection works on the basis of levels of ppm.
The system can detect type of smoke that is being detected.



CONCLUSION

The Raspberry Pi 4-based surveillance security system, which includes a PIR motion sensor, Pi camera, smoke detector, LED light, buzzer, and SMS module, proved an effective and efficient security solution for homes and businesses.

- ➔ Accurate motion detection was complemented by timely video capture.
- ➔ Proactive smoke detection ensured quick responses to fire hazards.
- ➔ The alarm system provided immediate alerts, while the SMS notifications informed users of threats promptly.

FUTURE PROSPECTS

1. **Integrating a database management system**

Case Study: A shopkeeper locks his shop and goes home, the next day he finds out a burglary has occurred. How does our system prevent this?

- The PIR motion sensor on the lock will detect the motion of the lock being broken.
- Camera records a video (similar to a CCTV) in the direction of the motion.
- Raspberry-Pi checks the image of the person with an already loaded data base of known faces.
- If it is a known person, no alert will be given.
- If the person's face is unknown, the buzzer and led turn ON
- Notification is sent to owner's phone.

2. **Rotating Camera**

Integrating the RPi Camera with a RPi controlled Servo or DC motor allows the camera to go a full 360 degrees based on the location of motion or smoke detected.

3. Using our system over a wide area network

Using frequency operated transceiver modules allow us to use multiple sensors in multiples locations, covering a wider area and reducing the cost of deploying other systems.

4. Remote-access control Dashboard

Other than convenient mobile notifications, a remotely-accessible dashboard, i.e a RPi compatible display can be deployed remotely so as to monitor all security parameters at once.

5. Use of other sensors in the MQ series

- MQ-2 - Methane, Butane, LPG, smoke
- MQ-3 - Alcohol, Ethanol, smoke
- MQ-4 - Methane, CNG Gas
- MQ-5 - Natural gas, LPG
- MQ-6 - LPG, butane gas
- MQ-7 - Carbon Monoxide
- MQ-8 - Hydrogen Gas
- MQ-9 - Carbon Monoxide, flammable gasses
- MQ131 - Ozone
- MQ135 - Air Quality (CO, Ammonia, Benzene, Alcohol, smoke)
- MQ136 - Hydrogen Sulfide gas
- MQ137 - Ammonia
- MQ138 - Benzene, Toluene, Alcohol, Acetone, Propane, Formaldehyde gas, Hydrogen
- MQ214 - Methane, Natural gas

These can be integrated to make the system more accurate, reliable and detect a variety of other harmful gases or simply to monitor the effects of some known gases present around us.