

Smart Room Monitoring and Automation System

Divyesh Bansal (23112036)
IIT Roorkee

1 Problem Statement

The goal of this project is to monitor and automate a room's environment by tracking:

- Temperature and humidity
- Light levels
- Door/window status

The system controls lighting and appliances automatically based on sensor readings.

2 Hardware Components

- **Microcontroller:** Raspberry Pi 3
- **Sensors:**
 - DHT11 (Temperature and Humidity)
 - LDR (Light Dependent Resistor)
 - Magnetic Reed Switch (Door/Window Position)
- **Actuators:**
 - RGB LED (status indicator: Fan ON = Green, Dark = Blue, Open Door = Red)
 - Relay (controls appliances like fan or light)
 - LCD Display (real-time environmental data)

3 Software Setup

3.1 SSH Connection to Raspberry Pi

To connect to the Raspberry Pi remotely from Windows using PowerShell:

```
# Find Pi IP using arp -a
# Connect via SSH
ssh pi@<Pi_IP>
```

After connecting via SSH, install required libraries:

```
sudo apt update
sudo apt install python3-pip
pip3 install RPi.GPIO adafruit-circuitpython-dht RPLCD
```

4 Hardware Connections

4.1 Pin Configuration on Raspberry Pi

- DHT11 DATA → GPIO4
- LDR → GPIO18
- Reed switch → GPIO17
- Relay IN → GPIO27
- RGB LED RED → GPIO23, GREEN → GPIO24, BLUE → GPIO25
- LCD (I2C) → SDA, SCL pins

5 Block Diagram

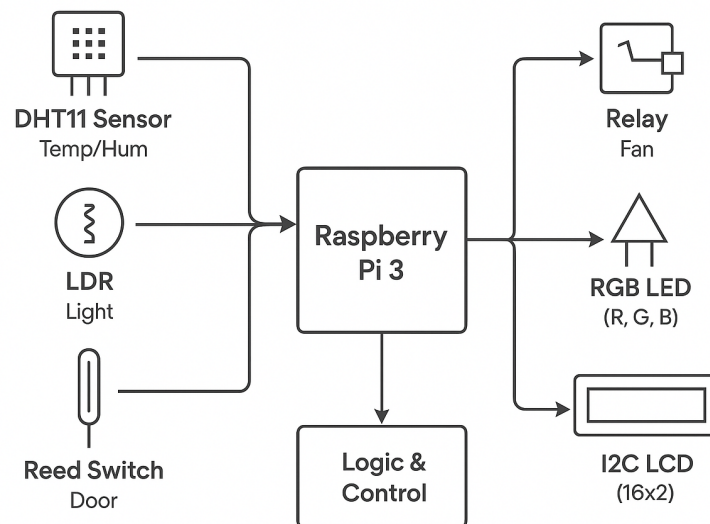


Figure 1: Block Diagram of Smart Room Monitoring and Automation System

5.1 Breadboard Setup

- Connect all sensor VCC pins to 3.3V or 5V and GND appropriately.
- RGB LED cathode to GND, anodes to GPIO pins via 220Ω resistors.
- Relay module VCC to 5V, IN to GPIO27, GND to GND.
- LCD connected via I2C bus.

6 Python Code for Automation

```
import time
import RPi.GPIO as GPIO
import board
import adafruit_dht
from RPLCD.i2c import CharLCD

# Pin Setup
DHT_PIN = board.D4
LDR_PIN = 18
REED_PIN = 17
RELAY_PIN = 27
RED_PIN = 23
GREEN_PIN = 24
BLUE_PIN = 25

# GPIO Setup
GPIO.setmode(GPIO.BCM)
GPIO.setup(LDR_PIN, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)
GPIO.setup(REED_PIN, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)
GPIO.setup(RELAY_PIN, GPIO.OUT)
GPIO.setup(RED_PIN, GPIO.OUT)
GPIO.setup(GREEN_PIN, GPIO.OUT)
GPIO.setup(BLUE_PIN, GPIO.OUT)

# LCD Setup
lcd = CharLCD('PCF8574', 0x27)
lcd.clear()

# DHT11 Setup
dhtDevice = adafruit_dht.DHT11(DHT_PIN)
temperature_prev = None
humidity_prev = None

def set_rgb(r, g, b):
    GPIO.output(RED_PIN, r)
    GPIO.output(GREEN_PIN, g)
    GPIO.output(BLUE_PIN, b)

try:
```

```

while True:
    try:
        temperature = dhtDevice.temperature
        humidity = dhtDevice.humidity
        if temperature is not None:
            temperature_prev = temperature
        if humidity is not None:
            humidity_prev = humidity
    except RuntimeError:
        temperature = temperature_prev
        humidity = humidity_prev

    light = GPIO.input(LDR_PIN)
    door = GPIO.input(REED_PIN)

    lcd.clear()
    if temperature is not None and humidity is not None:
        lcd.write_string(f"T:{temperature:.1f}C_H:{humidity:.1f}%")
    else:
        lcd.write_string("DHT11_Reading...")
    lcd.cursor_pos = (1, 0)
    lcd.write_string(f"L:{'BRIGHT' if light else 'DARK'}_D:{'CLOSE' if door else 'OPEN'}")

    if temperature is not None and temperature > 35:
        GPIO.output(RELAY_PIN, GPIO.HIGH)
        set_rgb(0, 1, 0)
    else:
        GPIO.output(RELAY_PIN, GPIO.LOW)
        set_rgb(0, 0, 0)

    if light == 0: # Dark
        set_rgb(0, 0, 1)

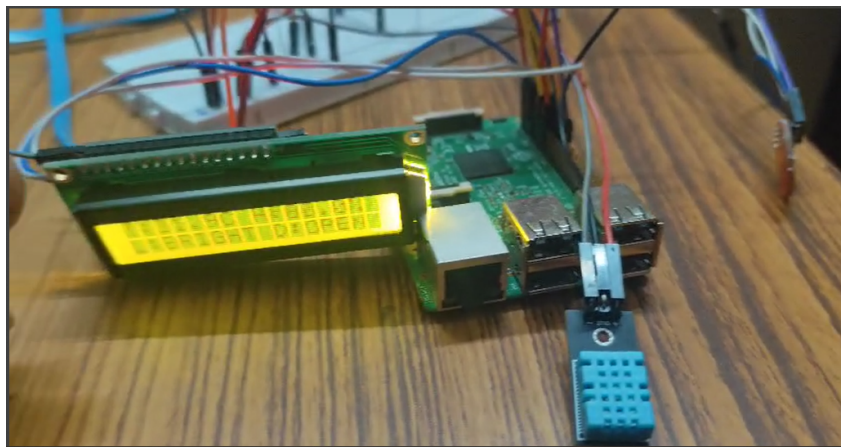
    if door == 0: # Door open
        set_rgb(1, 0, 0)

    time.sleep(2)

except KeyboardInterrupt:
    print("Exiting program...")

finally:
    lcd.clear()
    GPIO.cleanup()

```



7 Working Logic

- **Temperature Monitoring:** Fan turns ON if temperature exceeds 35°C. RGB LED turns green.
- **Light Detection:** RGB LED turns blue in darkness.
- **Door Monitoring:** RGB LED turns red if the door is open.
- **LCD Display:** Continuously shows temperature, humidity, light status, and door status.

8 Conclusion

The Smart Room Monitoring and Automation System effectively tracks environmental parameters and automates appliances based on conditions. RGB LED provides a clear status indication: **Fan ON = Green, Dark = Blue, Door Open = Red.**

[Watch on YouTube](#)