Experiment - 4

Student Name: Pranjal Singh **UID: 22BCS13041**
Branch: BE-CSE **Section/Group:FL-IOT-601/A**
Semester: 6th **Date of Performance: 13/02/25**
Subject Name: IOT Edge ML LAB **Subject Code: 22CSP-367**

1. Aim: Build a security system with any sensor and alerts using Blynk.

2. Objective: To design and implement a security system using sensors (e.g., PIR motion sensor, magnetic door sensor, or ultrasonic sensor) and integrate it with the Blynk platform to send real-time alerts.

3. Hardware Used:

1. PIR Motion Sensor (HC-SR501)
2. ESP8266/NodeMCU (or any Wi-Fi-enabled microcontroller)
3. Buzzer/LED (for local alerts, optional)
4. Blynk App (installed on your smartphone)
5. Breadboard and jumper wires
6. Ultrasonic Sensor (HC-SR04)

4. Procedure:

1. Connect the Hardware: PIR Sensor Pinout:

1. **VCC:** Connect to 3.3V or 5V (depending on the sensor model).
2. **GND:** Connect to GND.
3. **OUT:** Connect to a digital pin on ESP8266 (e.g., D5).
4. **PIR VCC** → NodeMCU 3.3V
5. **PIR GND** → NodeMCU GND
6. **PIR OUT** → NodeMCU D5
7. **Buzzer/LED (optional)** → D2

2. Set Up Blynk:

1. Download and install the Blynk app (iOS/Android).
2. Create a new project and select ESP8266 as the device.
3. Note down the Auth Token sent to your email.
4. Add a Notification Widget in the app for alerts.

3. Install Libraries in Arduino IDE:

1. Blynk Library:
    1. Go to Tools > Manage Libraries and search for Blynk.
    2. Install the Blynk library.
2. ESP8266 Board Support:
3. Go to File > Preferences and add the following URL to the Additional Boards Manager.
4. http://arduino.esp8266.com/stable/package_esp8266com_index.json
   Go to Tools > Board > Boards Manager and install the ESP8266 package.

4. Code:

```
#define BLYNK_TEMPLATE_ID "YourTemplateID"

#define BLYNK_DEVICE_NAME "SecuritySystem"

#define BLYNK_AUTH_TOKEN "YourAuthToken"

#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
// Blynk and Wi-Fi credentials
char auth[] = "YourAuthToken";
char ssid[] = "Your_SSID";
char pass[] = "Your_PASSWORD";


// PIR sensor pin int pirPin = D5; int
buzzerPin = D2;
void setup() { Serial.begin(115200); Blynk.begin(auth, ssid, pass);
```

```
pinMode(pirPin, INPUT); pinMode(buzzerPin, OUTPUT);
digitalWrite(buzzerPin, LOW);


Serial.println("Security system ready."); }
void loop() { Blynk.run();
if (digitalRead(pirPin) == HIGH) {
Serial.println("Motion Detected!");
Blynk.notify("Alert! Motion Detected at Home.");
digitalWrite(buzzerPin, HIGH);
Turn on buzzer/LED delay(5000); // Alert duration
digitalWrite(buzzerPin, LOW); // Turn off buzzer/LED }
}
```

Blynk Code

```
#define BLYNK_PRINT Serial #include
<ESP8266WiFi.h> #include
<BlynkSimpleEsp8266.h> BlynkTimer
timer;
char auth[] = "xxxxx"; //Enter the authentication code sent by Blynk to your Email char ssid[] =
"xxxxx"; //Enter your WIFI SSID
char pass[] = "xxxxx"; //Enter your WIFI Password int
flag=0;
void notifyOnButtonPress()
{
int isButtonPressed = digitalRead(D1);
if (isButtonPressed==1 && flag==0)
{ Serial.println("Someone Opened the door");
Blynk.notify("Alert : Someone Opened the door");
flag=1;
}
else if (isButtonPressed==0)
{ flag=0;
```

```
} }
void setup()
{
Serial.begin(9600);
Blynk.begin(auth, ssid, pass); pinMode(D1,INPUT_PULLUP);
timer.setInterval(16000L,notifyOnButtonPress); }
void loop()
{ Blynk.run();
timer.run(); }
```

6. Output:

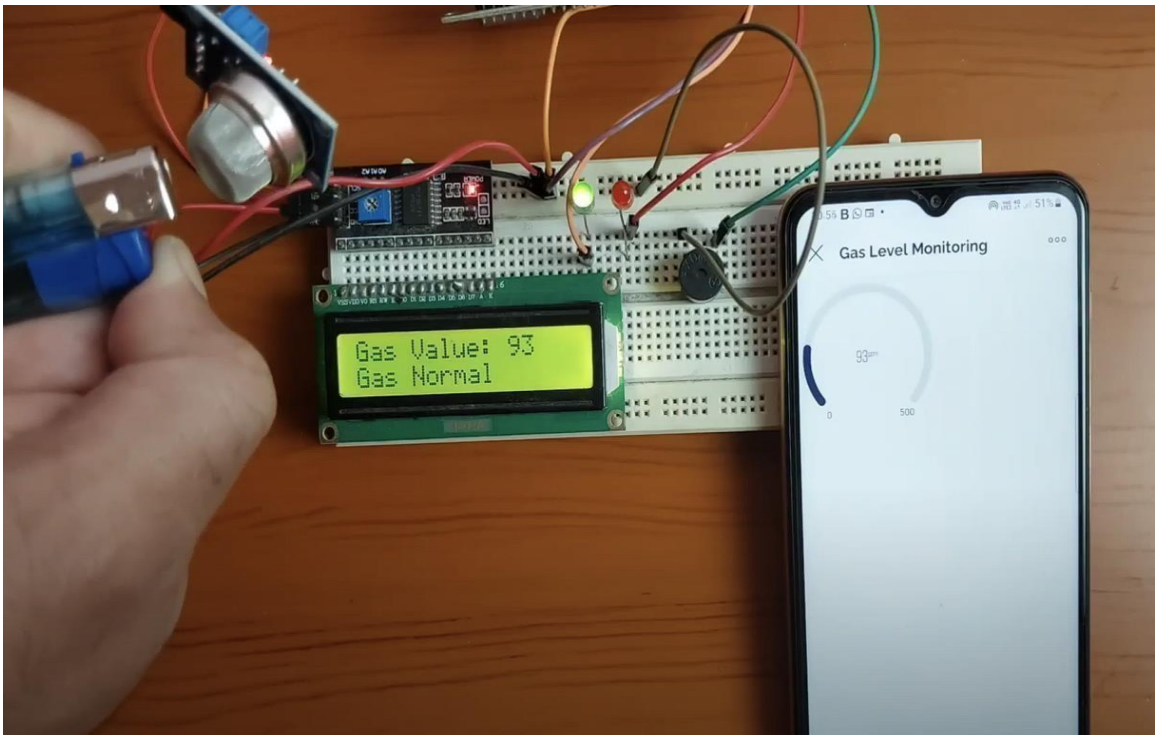

Fig 1

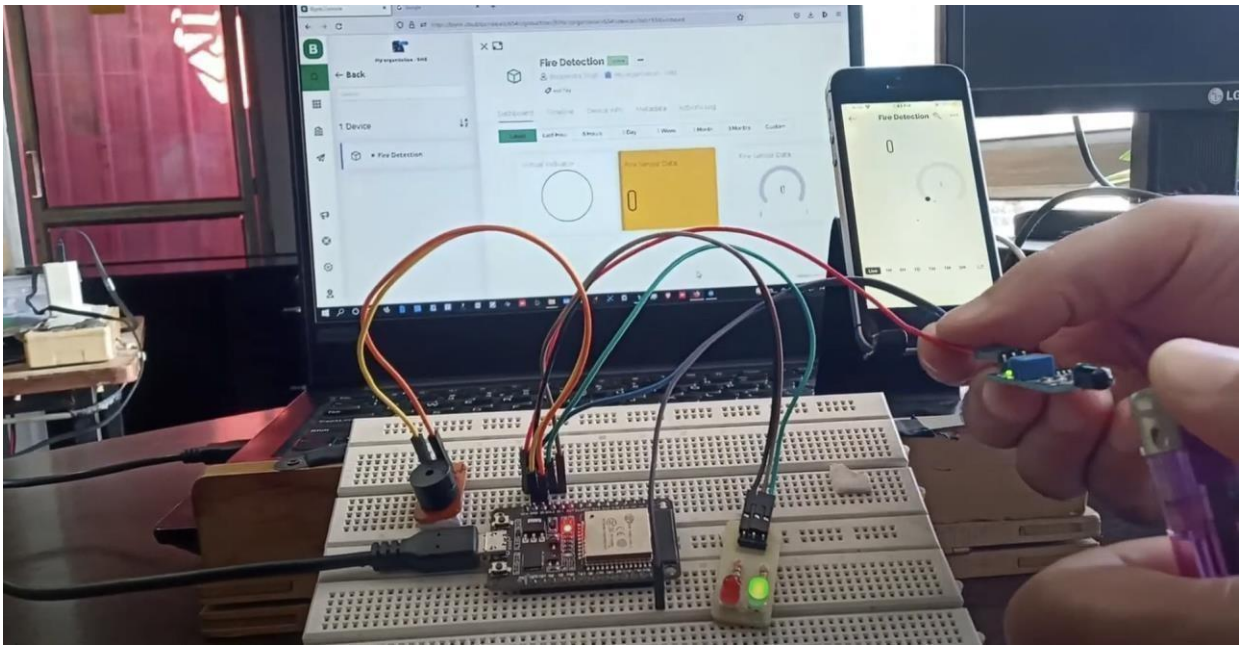

Fig 2

7. Learning Outcome:

1. **IoT and Blynk Integration** – Learn how to connect sensors with Blynk for real- time monitoring and remote alerts.
2. **Sensor and Hardware Interfacing** – Gain hands-on experience in working with motion, door, or gas sensors and microcontrollers like ESP8266/ESP32.
3. **Alert Mechanisms** – Implement real-time notifications via Blynk (push alerts, email, or SMS) and physical alerts using buzzers or LEDs.
4. **Embedded Programming** – Develop coding skills in C++ (Arduino IDE) or Micro Python to process sensor data and trigger security actions.