

# Experiment: Arduino-Based Smoke Sensor Detection

## 1. Aim

To design and implement a smoke detection system using an **Arduino** and an **MQ-2/MQ-135 smoke sensor**, which triggers an alert when smoke levels exceed a predefined threshold.

## 2. Components Required

- **Arduino Uno** (or any compatible board)
- **MQ-2/MQ-135 Smoke Sensor**
- **Buzzer**
- **LED**
- **Resistors (220Ω, 1kΩ)**
- **Jumper Wires**
- **Breadboard**
- **Power Supply (5V from Arduino)**

## 3. Background Theory

The **MQ-2/MQ-135 smoke sensor** is widely used for gas detection in home safety systems. It can detect gases like **LPG, methane, butane, alcohol, smoke, and carbon monoxide**. The sensor has a **heater element** that ionizes gases, changing the resistance of the sensor. This change is converted into an **analog voltage**, which can be read by the Arduino.

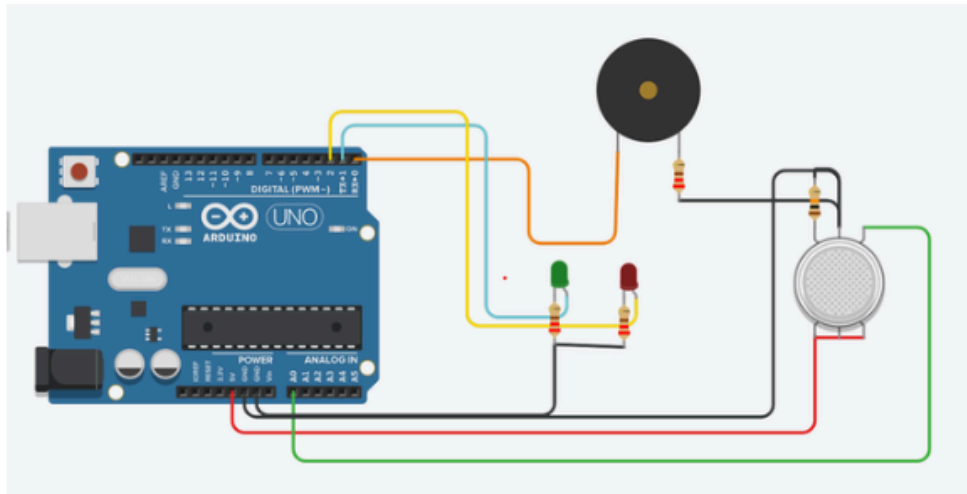
- **Working Principle:**
  - o The sensor outputs an **analog voltage** proportional to the smoke concentration.
  - o The Arduino processes this voltage and triggers an alert if it crosses the threshold.
  - o A **buzzer and LED** provide a visual and audible alert.

## 4. Circuit Diagram

Connections:

1. **MQ-2/MQ-135 Sensor**
  - o VCC → 5V (Arduino)
  - o GND → GND (Arduino)
  - o A0 → A0 (Arduino)
  - o D0 → Digital Pin (Optional, if using digital output)
2. **Buzzer**
  - o Positive → Pin 9 (Arduino)
  - o Negative → GND
3. **LED**
  - o Positive → Pin 8 (Arduino)
  - o Negative → GND (via a 220Ω resistor)

## Circuit Diagram



## 5. Arduino Code

```
cpp
CopyEdit
const int smokeSensorPin = A0; // Analog input for MQ-2
const int buzzerPin = 9;       // Buzzer pin
const int ledPin = 8;          // LED pin
const int threshold = 300;     // Smoke threshold value

void setup() {
  pinMode(buzzerPin, OUTPUT);
  pinMode(ledPin, OUTPUT);
  Serial.begin(9600); // Start Serial Monitor
}

void loop() {
  int sensorValue = analogRead(smokeSensorPin); // Read sensor value
  Serial.print("Smoke Level: ");
  Serial.println(sensorValue); // Display value in serial monitor

  if (sensorValue > threshold) { // If smoke detected
    digitalWrite(buzzerPin, HIGH);
    digitalWrite(ledPin, HIGH);
    Serial.println("WARNING: Smoke Detected!");
  } else {
    digitalWrite(buzzerPin, LOW);
    digitalWrite(ledPin, LOW);
  }

  delay(1000); // Wait 1 second before next reading
}
```

## 6. Steps

1. **Set up the hardware:** Connect the **MQ-2/MQ-135 sensor, buzzer, and LED** to the Arduino as per the circuit diagram.
2. **Upload the Code:** Use **Arduino IDE** to upload the above code.
3. **Monitor the Serial Output:** Open the **Serial Monitor** (9600 baud rate) to check real-time smoke levels.
4. **Test with Smoke:** Introduce **smoke (e.g., from a matchstick)** near the sensor.
5. **Observe the Response:** If the smoke level **exceeds the threshold**, the **buzzer and LED turn ON**.
6. **Verify Alert Deactivation:** Once the smoke clears, the **buzzer and LED should turn OFF**.

## 7. Result

The smoke detection system successfully detects smoke and triggers an **audible and visual alarm**. The sensor readings are displayed on the **Serial Monitor**, and the system functions as expected when exposed to smoke.