Experiment No.: 05

Experiment Name: Study and Observation of Arduino Uno Based Gas Sensor

Objective:

Theory:

Required Hardwares with Quantity:

Required Software:

Working Procedure:

Hardware Arrangement Diagram:

Sketch:

Result and Output:

Conclusion:

Basic Knowledge:

A gas sensor is a device that detects the presence and concentration of gases in the environment. These sensors can be used in various applications, including:

1. **Industrial Safety**: To monitor toxic or explosive gases in factories or chemical plants.
2. **Environmental Monitoring**: To measure air quality and pollutant levels.
3. **Automotive**: In vehicles to detect harmful gases or monitor exhaust emissions.
4. **Home Safety**: For smoke and carbon monoxide detection.
5. **Agriculture**: To monitor gases released by soil and plants.

Types of Gas Sensors

1. **Electrochemical Sensors**: Detect specific gases by measuring the current produced during a chemical reaction.
2. **Semiconductor Sensors**: Change their electrical resistance when exposed to gas.
3. **Infrared Sensors**: Use infrared light to detect gas concentrations based on absorption characteristics.
4. **Photoionization Detectors (PID)**: Use ultraviolet light to ionize gas molecules, detecting their presence based on the resulting current.

   Gas Sensor modules:
   1. MQ-2 Flammable Gas and Smoke Sensor Module
   2. MQ-3 Alcohol Detector Gas Sensor Module
   3. MQ-4 Methane Gas Sensor Module

4. MQ-135 Air Quality and Hazardous Gas Sensor Module (Ammonia (NH3), Benzene (C6H6), Alcohol, Carbon Dioxide (CO2), Smoke, Volatile Organic Compounds (VOCs))

MQ-7 Carbon Monoxide Gas Sensor Module, MQ-5 (natural gas, and LPG), MQ-6 (butane and LPG), MQ-9 (CO, and flammable gases), MQ131 (ozone), MQ136 (Hydrogen sulfide gas), MQ137 (ammonia), MQ138 (benzene, alcohol, propane, toluene, formaldehyde gas, and hydrogen), MQ214 (methane, and natural gas), MQ303A (alcohol, smoke, Ethanol), MQ306A (LPG and butane), MQ307A(CO), MQ309A(CO and flammable gas).

What is MQ-135 sensor?

MQ-135 is an air quality or air pollution measuring sensor device. It can detect various chemical contents in air and **give appropriate voltage variation at the output pin depending on the chemical concentration in air**. It can detect alcohol, Benzene, smoke, NH3, butane, propane etc. if anyone of the stated chemical concentration rises, the sensor converts the chemical concentration in air to appropriate voltage range, which can be processed by Arduino or any microcontroller.

Key Features of the MQ135 Sensor:

1. **Type**: It is a **semiconductor gas sensor** that uses a sensitive material (typically tin dioxide) that changes its resistance in the presence of target gases.
2. **Operating Voltage**: Typically operates at 5V.
3. **Detection Range**: The MQ135 can detect gas concentrations in the range of a few parts per million (ppm) to thousands of ppm, depending on the specific gas.
4. **Output**: The sensor provides an analog voltage output that varies with the concentration of the detected gases. This output can be read by a microcontroller or other electronic device.
5. **Applications**: Commonly used in air quality monitoring, indoor air quality applications, and various projects involving gas detection (e.g., in smart homes or industrial settings).

Concentration range is 10-1000 ppm (parts per million). The MQ-135 can detect NH3 and Alcohol gas concentrations anywhere from 10-300 ppm and Benzene from 10-1000ppm.

PPM: 1 PPM means 1 part of a substance per 1,000,000 parts of the total mixture.
In terms of concentration, PPM can also be approximated as:

- 1 PPM = 1 mg/L (milligram per liter) for liquids.
- 1 PPM = 1 $\mu g/m^3$ (microgram per cubic meter) for gases.

PPM is commonly used to measure pollutants in air and water, such as carbon dioxide, nitrogen dioxide, and various hazardous chemicals.
Air Quality: A report might indicate that the concentration of CO2 in the atmosphere is 400 PPM, meaning there are 400 molecules of CO2 for every 1,000,000 molecules of air.

Construction of MQ-135

1. **Sensing Element**:
    - The core of the MQ-135 sensor is its sensing element, typically made of a **tin dioxide (SnO2)** semiconductor. This material is sensitive to various gases.

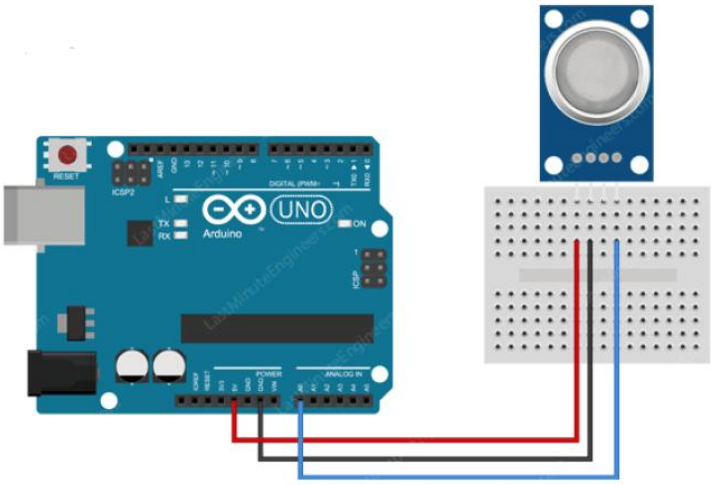
    - The resistance of the tin dioxide changes when exposed to target gases. The sensor detects this change in resistance to determine the concentration of the gas.
2. **Heating Element**:
    - The sensor includes a **heating element** that maintains the temperature of the sensing material. The temperature is crucial for the sensor's operation because the sensitivity of the tin dioxide varies with temperature.
    - This heating element is usually made of a wire wound around the sensing element and powered by a small voltage.
3. **Sensor Housing**:
    - The MQ-135 is encased in a protective **housing**, typically made from plastic or metal, to shield the sensing element from dust and physical damage while allowing gas molecules to diffuse into the sensor.
    - The housing has openings or perforations to facilitate gas entry.
4. **Electrodes**:
    - The sensing element has electrodes that connect to the external circuit. These electrodes measure the resistance changes in the tin dioxide when exposed to gases.
5. **Output Pins**:
    - The sensor has multiple pins for connections:
        - **VCC**: Power supply (typically 5V).
        - **GND**: Ground connection.
        - **AO**: Analog output pin, providing a voltage proportional to the detected gas concentration.
        - **DO**: Digital output pin, which can be used to trigger an alert when a specific gas concentration exceeds a threshold.
6. **Calibration Resistor**:
    - A calibration resistor is often included in the circuit to adjust the sensor's sensitivity and response. The resistance value can influence how the sensor responds to different gas concentrations.

Working Principle

- When the MQ-135 is heated, the tin dioxide surface becomes reactive, and when it encounters specific gases, a chemical reaction occurs, altering its resistance.
- The change in resistance is measured and translated into an analog voltage output, which can be read by a microcontroller or an analog-to-digital converter for further processing and analysis.

Code(s):

1. **Finding the threshold value**

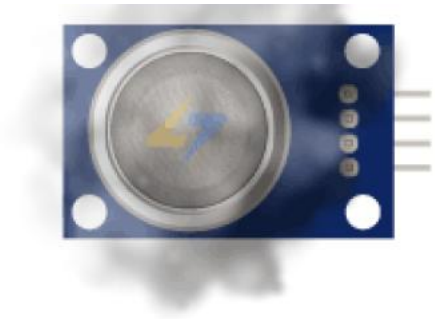


```
#define MQ135pin 0
float sensorValue; // variable to store sensor value
void setup() {
Serial.begin(9600); // sets the serial port to 9600
Serial.println("MQ135 warming up!");
delay(20000); // allow the MQ135 to warm up
}
void loop() {
sensorValue = analogRead(MQ135pin); // read analog input pin 0
Serial.print("Sensor Value: ");
Serial.println(sensorValue);
delay(2000); // wait 2s for next reading
}
```

When you run the sketch, you should see readings similar to the ones below:

- In the absence of smoke/gas (around 100)
- In the presence of smoke/gas (around 400)

Status: Smoke Absent
Test Reading: ~100



Status: Smoke Present
Test Reading: ~400

2. Determines whether the gas concentration is within acceptable limits

```
/* Change the threshold value with your own reading */
#define Threshold 400
```

```
#define MQ135pin 0
float sensorValue; //variable to store sensor value
void setup() {
Serial.begin(9600); // sets the serial port to 9600
Serial.println("MQ135 warming up!");
delay(20000); // allow the MQ135 to warm up
}
void loop() {
sensorValue = analogRead(MQ135pin); // read analog input pin 0
Serial.print("Sensor Value: ");
Serial.print(sensorValue);
if(sensorValue > Threshold)
{
Serial.print(" | Smoke detected!");
}
Serial.println("");
delay(2000); // wait 2s for next reading
}
```

3. Detecting the Presence of Smoke/Gas using Digital Output (D0)

```
#define MQ135pin 8
int sensorValue; //variable to store sensor value
void setup() {
Serial.begin(9600); // sets the serial port to 9600
Serial.println("MQ135 warming up!");
delay(20000); // allow the MQ135 to warm up
}
void loop() {
sensorValue = digitalRead(MQ135pin); // read digital output pin
Serial.print("Digital Output: ");
Serial.print(sensorValue);
// Determine the status
if (sensorValue) {
Serial.println(" | Smoke: -");
} else {
Serial.println(" | Smoke: Detected!");
}
delay(2000); // wait 2s for next reading
}
```