Grove - Gas Sensor(MQ5)

The Grove - Gas Sensor(MQ5) module is useful for gas leakage detection (in home and industry). It is suitable for detecting H2, LPG, CH4, CO, Alcohol. Due to its high sensitivity and fast response time, measurements can be taken as soon as possible. The sensitivity of the sensor can be adjusted by using the potentiometer.

Note
The sensor value only reflects the approximated trend of gas concentration in a permissible error range, it DOES NOT represent the exact gas concentration. The detection of certain components in the air usually requires a more precise and costly instrument, which cannot be done with a single gas sensor. If your project is aimed at obtaining the gas concentration at a very precise level, then we do not recommend this gas sensor.

Features

- Wide detecting scope
- Stable and long life
- Fast response and High sensitivity

Specification

Item	Parameter	Min	Typical	Max	Unit
VCC	Working Voltage	4.9	5	5.1	V
PH	Heating consumption	0.5	-	800	mW
RL	Load resistance		adjustable		
RH	Heater resistance	-	31±10%	-	Ω
Rs	Sensing Resistance	10	-	60	kΩ
Scope	Detecting Concentration	200	-	10000	ppm

Applications

- Gas leakage detection.
- Toys.

Hardware Overview

This is an Analog output sensor. This needs to be connected to any one Analog socket in Grove Base Shield. The examples used in this tutorial makes uses of A0 analog pin. Connect this module to the A0 port of Base Shield.

It is possible to connect the Grove module to Arduino directly by using jumper wires by using the connection as shown in the table below:

Arduino	Gas Sensor
5V	VCC
GND	GND
NC	NC
Analog A0	SIG

The output voltage from the Gas sensor increases when the concentration of gas increases. Sensitivity can be adjusted by varying the potentiometer. Please note that the best preheat time for the sensor is above 24 hours. For detailed information about the MQ-5 sensor, please refer to the data-sheet provided in **Resources** section.

Platforms Supported

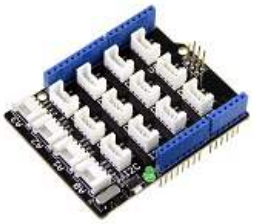
Arduino	Raspberry Pi	BeagleBone	Wio	LinkIt ONE

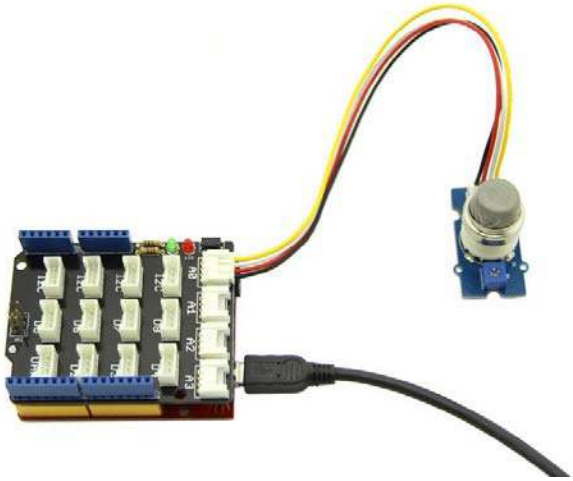
Caution
The platforms mentioned above as supported is/are an indication of the module's software or theoritical compatibility. We only provide software library or code examples for Arduino platform in most cases. It is not possible to provide software library / demo code for all possible MCU platforms. Hence, users have to write their own software library.

Getting Started

Play With Arduino

Materials required

Seeeduino V4.2	Base Shield	Grove - Gas Sensor(MQ5)
		



Connect the Grove - Gas Sensor(MQ5) to A0 port as shown in the picture above.

Gas Detection : Basic Example

In this example, the sensor is connected to A0 pin. The voltage read from the sensor is displayed. This value can be used as a threshold to detect any increase/decrease in gas concentration.

Note
You need an extra tool to find a certain threshold for various air condition. And then set the threshold in code.

```
void setup() {
    Serial.begin(9600);
}

void loop() {
    float sensor_volt;
    float sensorValue;

    sensorValue = analogRead(A0);
    sensor_volt = sensorValue/1024*5.0;

    Serial.print("sensor_volt = ");
    Serial.print(sensor_volt);
    Serial.println("V");
    delay(1000);
}
```

Measurement : Approximation

This examples demonstrates a way to know the approximate concentration of Gas. As per the data-sheet of the MQ5 sensors, these equations are tested for standard conditions and are not calibrated. It may vary based on change in temperature or humidity.

1. Keep the Gas Sensor in clean air environment. Upload the program below.

```
void setup() {
    Serial.begin(9600);
}

void loop() {
    float sensor_volt;
    float RS_air; //  Get the value of RS via in a clear air
    float R0;  // Get the value of R0 via in H2
    float sensorValue;

        /*--- Get a average data by testing 100 times ---*/
    for(int x = 0 ; x < 100 ; x++)
    {
        sensorValue = sensorValue + analogRead(A0);
    }
    sensorValue = sensorValue/100.0;
```

```
17        /*-----------------------------------------------*/
18
19    sensor_volt = sensorValue/1024*5.0;
20    RS_air = (5.0-sensor_volt)/sensor_volt; // omit *RL
21    R0 = RS_air/6.5; // The ratio of RS/R0 is 6.5 in a clear air from Graph
22(Found using WebPlotDigitizer)
23
24    Serial.print("sensor_volt = ");
25    Serial.print(sensor_volt);
26    Serial.println("V");
27
28    Serial.print("R0 = ");
29    Serial.println(R0);
30    delay(1000);
  }
```

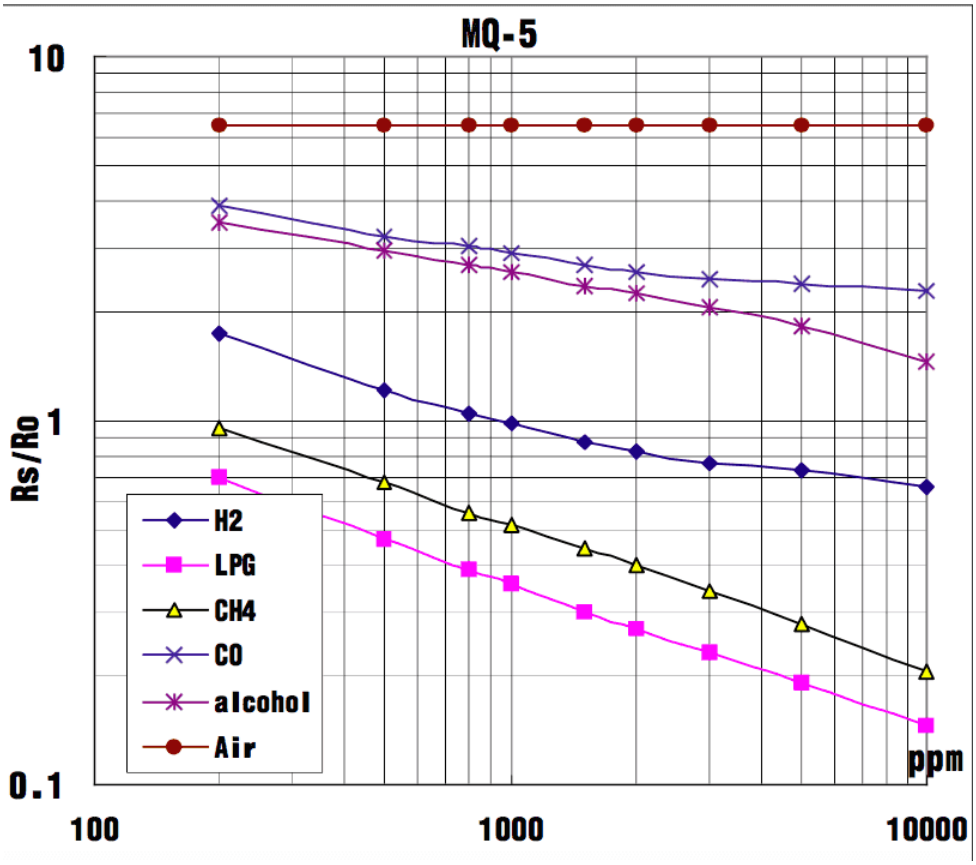
1. Then, open the serial monitor of Arduino IDE. Write down the value of R0 and this needs to be used in the next program. Please node down the R0 after the reading stabilizes.

Replace the R0 below with value of R0 tested above . Expose the sensor to any one of the gas listed above.

```
 1void setup() {
 2    Serial.begin(9600);
 3}
 4
 5void loop() {
 6
 7    float sensor_volt;
 8    float RS_gas; // Get value of RS in a GAS
 9    float ratio; // Get ratio RS_GAS/RS_air
10    int sensorValue = analogRead(A0);
11    sensor_volt=(float)sensorValue/1024*5.0;
12    RS_gas = (5.0-sensor_volt)/sensor_volt; // omit *RL
13
14          /*-Replace the name "R0" with the value of R0 in the demo of
15First Test -*/
16    ratio = RS_gas/R0;  // ratio = RS/R0
17          /*-----------------------------------------------------------
18--------*/
19
20    Serial.print("sensor_volt = ");
21    Serial.println(sensor_volt);
22    Serial.print("RS_ratio = ");
23    Serial.println(RS_gas);
24    Serial.print("Rs/R0 = ");
25    Serial.println(ratio);
26
27    Serial.print("\n\n");
28
29    delay(1000);

  }
```

Now, we can get the concentration of gas from the figure below.

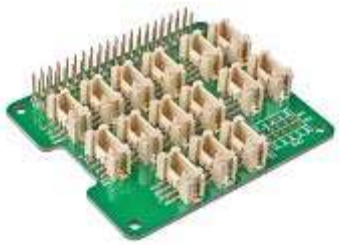


According to the figure, we can see that the minimum concentration we can test is 200ppm and the maximum is 10000ppm, in a other word, we can get a concentration of gas between 0.02% and 1%. However, we can't provide a formula because the relation between ratio and concentration is nonlinear.

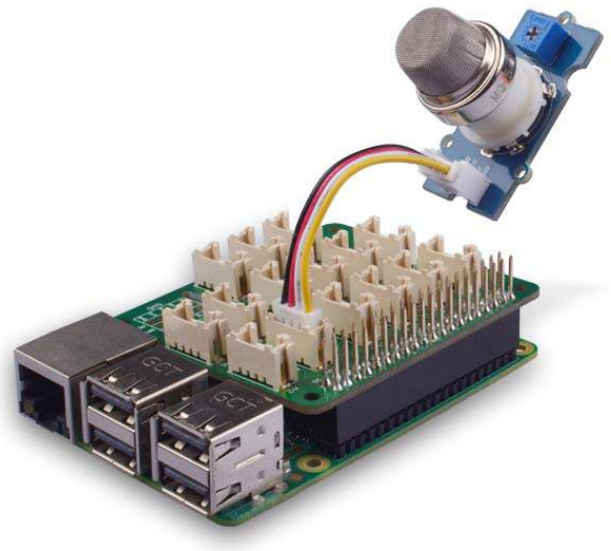
Play With Raspberry Pi (With Grove Base Hat for Raspberry Pi)

Hardware

- **Step 1**. Things used in this project:

Raspberry pi	Grove Base Hat for RasPi	Grove - Gas Sensor(MQ5)
		

- **Step 2**. Plug the Grove Base Hat into Raspberry.
- **Step 3**. Connect the Grove - Gas Sensor(MQ5) to port A0 of the Base Hat.
- **Step 4**. Connect the Raspberry Pi to PC through USB cable.



Note
For step 3 you are able to connect the Grove - Gas Sensor(MQ5) to **any Analog Port** but make sure you change the command with the corresponding port number.

Software

- **Step 1**. Follow Setting Software to configure the development environment.
- **Step 2**. Download the source file by cloning the grove.py library.

```
cd ~
git clone https://github.com/Seeed-Studio/grove.py
```

- **Step 3**. Excute below commands to write the code.

```
cd grove.py/grove
nano grove_gas_sensor_mq5.py
```

Then you should copy following code in this file and hit `Ctrl` + `X` to quit and save.

```
import math
import sys
import time
from grove.adc import ADC
```

```
 5
 6
 7class GroveGasSensorMQ5:
 8
 9    def __init__(self, channel):
10        self.channel = channel
11        self.adc = ADC()
12
13    @property
14    def MQ5(self):
15        value = self.adc.read(self.channel)
16        return value
17
18Grove = GroveGasSensorMQ5
19
20
21def main():
22    if len(sys.argv) < 2:
23        print('Usage: {} adc_channel'.format(sys.argv[0]))
24        sys.exit(1)
25
26    sensor = GroveGasSensorMQ5(int(sys.argv[1]))
27
28    print('Detecting...')
29    while True:
30        print('Gas value: {0}'.format(sensor.MQ5))
31        time.sleep(.3)
32
33if __name__ == '__main__':
34    main()
```

- **Step 4**. Excute below commands to run code.

```
1python grove_gas_sensor_mq5.py  0
```

Success

If everything goes well, you will be able to see the following result

```
 1pi@raspberrypi:~/grove.py/grove $ python grove_gas_sensor_mq5.py  0
 2Detecting...
 3Gas value: 28
 4Gas value: 28
 5Gas value: 27
 6Gas value: 26
 7Gas value: 26
 8^CTraceback (most recent call last):
 9  File "grove_gas_sensor_mq5.py", line 69, in <module>
10    main()
11  File "grove_gas_sensor_mq5.py", line 66, in main
12    time.sleep(.3)
13KeyboardInterrupt
```

You can quit this program by simply press `Ctrl` + `C`.

Notice
You may have noticed that for the analog port, the silkscreen pin number is something like **A0, A1**, however in the command we use parameter **0** and **1**, just the same as digital port. So please make sure you plug the module into the correct port, otherwise there may be pin conflicts.

Resources

Suggest Reading / References

- How to choose a Gas Sensor
- What's LEL

Schematic

- Grove Gas Sensor - EAGLE (Schematic and Board) files
- Grove Gas Sensor - PDF Schematic

Datasheet

- MQ-5 Datasheet

Tech Support

Please submit any technical issue into our forum or drop mail to techsupport@seeed.cc.