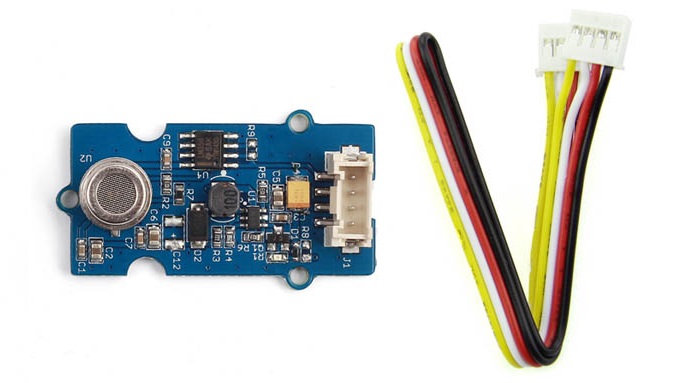
**Grove - Air Quality Sensor v1.3**

**Introduction**

This sensor is designed for comprehensive monitor over indoor air condition. It’s responsive to a wide scope of harmful gases, as carbon monoxide, alcohol, acetone, thinner, formaldehyde and so on. Due to the measuring mechanism, this sensor can’t output specific data to describe target gases’ concentrations quantitatively. But it’s still competent enough to be used in applications that require only qualitative results, like auto refresher sprayers and auto air cycling systems.

[https://github.com/SeeedDocument/Seeed-WiKi/raw/master/docs/images/300px-Get_One_Now_Banner-ragular.png](https://www.seeedstudio.com/Grove-Air-quality-sensor-v1.3-p-2439.html)

**Features**

* Responsive to a wide scope of target gases
* Cost efficient
* Durable
* Compatible with 5V and 3.3V

**Cautions**

1. Requires relatively clean air as an initial condition.
2. Long time exposure to highly polluted air can significantly weaken its sensitivity.
3. Coffre-fort et armoire forte:

**Usage**

**With Arduino**

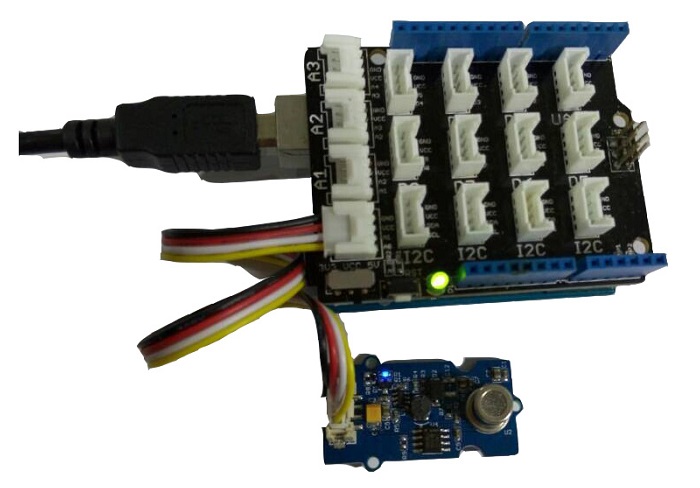
As described in Introduction, this sensor performs better in providing qualitative results over a wide scope of target gases. In this demo, we will define 4 statuses for reference in the ‘.cpp’ file. They are:

* a. air fresh – indicating a good air condition
* b. low pollution – indicating a rather low concentration of target gases exist.
* c. high pollution(without “Force signal active” message printed on serial monitor) – you should be aware of the pollution level and consider if some measures could be taken.
* d. high pollution(with “Force signal active” message printed on serial monitor) – instant measures should be taken to improve the air quality.

We encapsulated the decision structure in a ‘.cpp’ file. You can find information there on how to modify the thresholds.

Let’s try it out!

1.Connect the Sensor to A0 port of [Grove - Base Shield](https://www.seeedstudio.com/Grove-Base-Shield-p-754.html).



2.Download the File :[AirQuality\_Sensor Library](https://github.com/SeeedDocument/Grove_Air_Quality_Sensor_v1.3/raw/master/res/AirQuality_Sensor.zip).

3.Wait at least 2 minutes after powering on for heat-up of the sensor. Then open the example below by the path: File -> Example ->AirQuality\_Sensor->AirQuality\_Sensor.

*/\**

*AirQuality Demo V1.0.*

*connect to A1 to start testing. it will needs about 20s to start*

*\* By: http://www.seeedstudio.com*

*\*/*

#**include** "AirQuality.h"

#**include** "Arduino.h"

AirQuality airqualitysensor;

**int** current\_quality =-1;

**void** setup()

{

Serial.begin(9600);

airqualitysensor.init(14);

}

**void** loop()

{

current\_quality=airqualitysensor.slope();

**if** (current\_quality >= 0)*// if a valid data returned.*

{

**if** (current\_quality==0)

Serial.**println**("High pollution! Force signal active");

**else** **if** (current\_quality==1)

Serial.**println**("High pollution!");

**else** **if** (current\_quality==2)

Serial.**println**("Low pollution!");

**else** **if** (current\_quality ==3)

Serial.**println**("Fresh air");

}

}

ISR(TIMER2\_OVF\_vect)

{

**if**(airqualitysensor.counter==122)*//set 2 seconds as a detected duty*

{

airqualitysensor.last\_vol=airqualitysensor.first\_vol;

airqualitysensor.first\_vol=analogRead(A0); *// change this value if you use another A port*

airqualitysensor.counter=0;

airqualitysensor.timer\_index=1;

PORTB=PORTB^0x20;

}

**else**

{

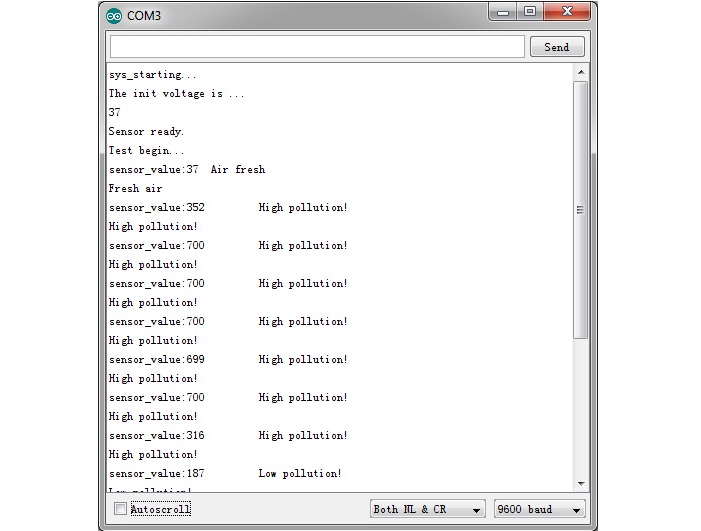
airqualitysensor.counter++;

}

}

4.Upload the code.

5.Open serial monitor.



To adjust the thresholds and indicating messages, refer to the decision structure below in the .cpp file.

**int** AirQuality::slope(**void**)

{

**while**(timer\_index)

{

**if**(first\_vol-last\_vol>400||first\_vol>700)

{

Serial.**println**("High pollution! Force signal active.");

timer\_index=0;

avg\_voltage();

**return** 0;

}

**else** **if**((first\_vol-last\_vol>400&&first\_vol<700)||first\_vol-vol\_standard>150)

{

Serial.**print**("sensor\_value:");

Serial.**print**(first\_vol);

Serial.**println**("\t High pollution!");

timer\_index=0;

avg\_voltage();

**return** 1;

}

**else** **if**((first\_vol-last\_vol>200&&first\_vol<700)||first\_vol-vol\_standard>50)

{

*//Serial.println(first\_vol-last\_vol);*

Serial.**print**("sensor\_value:");

Serial.**print**(first\_vol);

Serial.**println**("\t Low pollution!");

timer\_index=0;

avg\_voltage();

**return** 2;

}

**else**

{

avg\_voltage();

Serial.**print**("sensor\_value:");

Serial.**print**(first\_vol);

Serial.**println**("\t Air fresh");

timer\_index=0;

**return** 3;

}

}

**return** -1;

}

**With Raspberry Pi**

1.You need to prepare a raspberry pi and a grovepi or grovepi+.

2.You need to completed configuring the development enviroment, otherwise follow [here](https://seeeddoc.github.io/GrovePiPlus/" \l "Introducing_the_GrovePi.2B).

3.Connection - Plug the sensor to grovepi socket A0 by using a grove cable.

4.Navigate to the demos’ directory:

cd yourpath/GrovePi/Software/Python/

To see the code

nano grove\_air\_quality\_sensor.**py** # "Ctrl+x" **to** **exit** #

**import** time

**import** grovepi

*# Connect the Grove Air Quality Sensor to analog port A0*

*# SIG,NC,VCC,GND*

air\_sensor = 0

grovepi.pinMode(air\_sensor,"INPUT")

**while** **True**:

**try**:

*# Get sensor value*

sensor\_value = grovepi.analogRead(air\_sensor)

**if** sensor\_value > 700:

**print** "High pollution"

**elif** sensor\_value > 300:

**print** "Low pollution"

**else**:

**print** "Air fresh"

**print** "sensor\_value =", sensor\_value

time.sleep(.5)

**except** IOError:

**print** "Error"

5.Run the demo.

sudo **python** grove\_air\_quality\_sensor.**py**

**Resources**

* [AirQuality Sensor Library.zip](https://github.com/SeeedDocument/Grove_Air_Quality_Sensor_v1.3/raw/master/res/AirQuality_Sensor.zip)
* [Grove\_-\_Air\_quality\_sensor\_v1.3\_sch\_pcb.zip](https://github.com/SeeedDocument/Grove_Air_Quality_Sensor_v1.3/raw/master/res/Grove_-_Air_quality_sensor_v1.3_sch_pcb.zip)
* [Grove\_-\_Air\_quality\_sensor\_v1.3\_sch.pdf](https://github.com/SeeedDocument/Grove_Air_Quality_Sensor_v1.3/raw/master/res/Grove_-_Air_quality_sensor_v1.3_sch.pdf)
* [Air\_quality\_sensor\_MP503\_Chinese.pdf](https://github.com/SeeedDocument/Grove_Air_Quality_Sensor_v1.3/raw/master/res/Air_quality_sensor_MP503%20Chinese.pdf)
* [Air\_quality sensor\_MP503\_English.pdf](https://github.com/SeeedDocument/Grove_Air_Quality_Sensor_v1.3/raw/master/res/Mp503%20English.pdf)

**Help us make it better**

Welcome to the new documentation system of Seeed Studio. We have made a lot of progress comparing to the old wiki system and will continue to improve it to make it more user friendly and helpful. The improvement can't be done without your kindly feedback. If you have any suggestions or findings, you are most welcome to submit the amended version as our contributor via [Github](https://github.com/SeeedDocument/Seeed-WiKi) or give us suggestions in the survey below, it would be more appreciated if you could leave your email so that we can reply to you. Happy Hacking!