Grove - I2C UV Sensor (VEML6070)

The Grove - I2C UV Sensor(VEML6070) is an advanced ultraviolet (UV) light sensor with I2C protocol interface. Ultraviolet (UV) is electromagnetic radiation with a wavelength from 10 nm to 400 nm, shorter than that of visible light but longer than X-rays, this sensor detects 320-410nm light most effectively, and will converts solar UV light intensity to digital data.

This module is based on VEML6070, which has linear sensitivity to solar UV light and is easily adjusted by an external resistor.

What's more the active acknowledge (ACK) feature with threshold windows setting allows the UV sensor to send out a UVI alert message. Under a strong solar UVI condition, the smart ACK signal can be easily implemented by the software programming.

Features

- Excellent performance of UV radiation measurement under long time solar UV exposure
- Excellent UV sensitivity and linearity
- Excellent temperature compensation
- High dynamic detection resolution
- Support acknowledge feature (ACK)

Specification

| Item	Value
Operating Voltage	3.3V / 5V
Range of spectral sensitivity	320 ~ 410 nm
Peak Sensitivity	355 nm
UVA Sensitivity	5 µW/cm2/step(typical)
Interface	I²C
I²C Address	0x38(Data LSB) / 0x39(Data MSB)

Attention
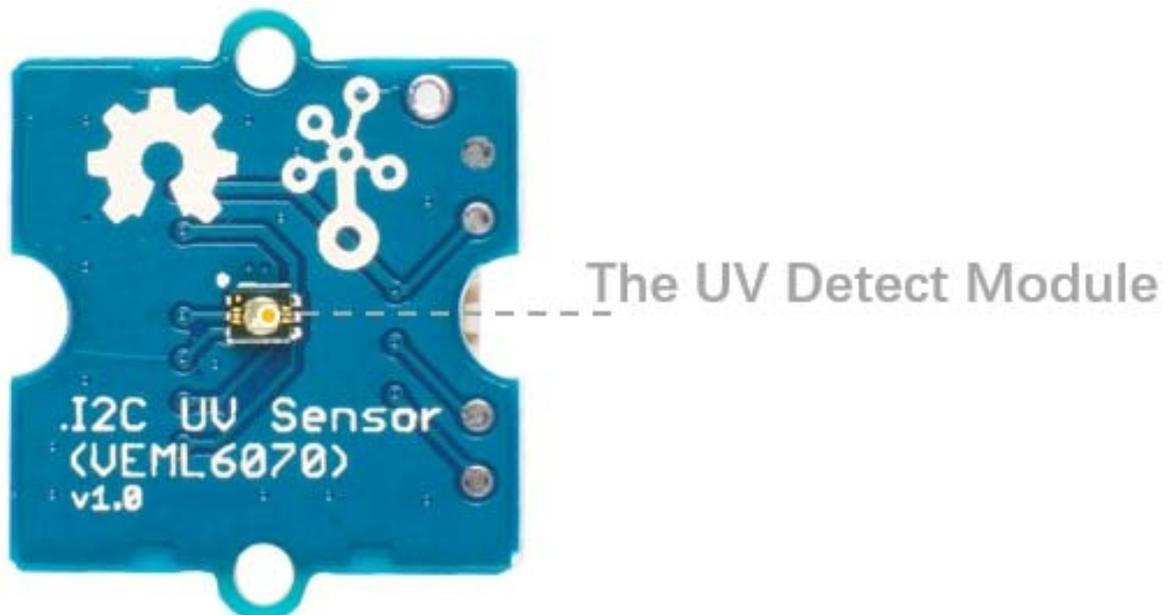Actually, this sensor has 3 I²C address, each address

Typical applications

- Solar UV indicator
- Cosmetic / outdoor sport handheld product
- Consumer products

Hardware Overview

Pin Out

5 ACK: interrupt pin, default low level, exceeding the threshold to become high
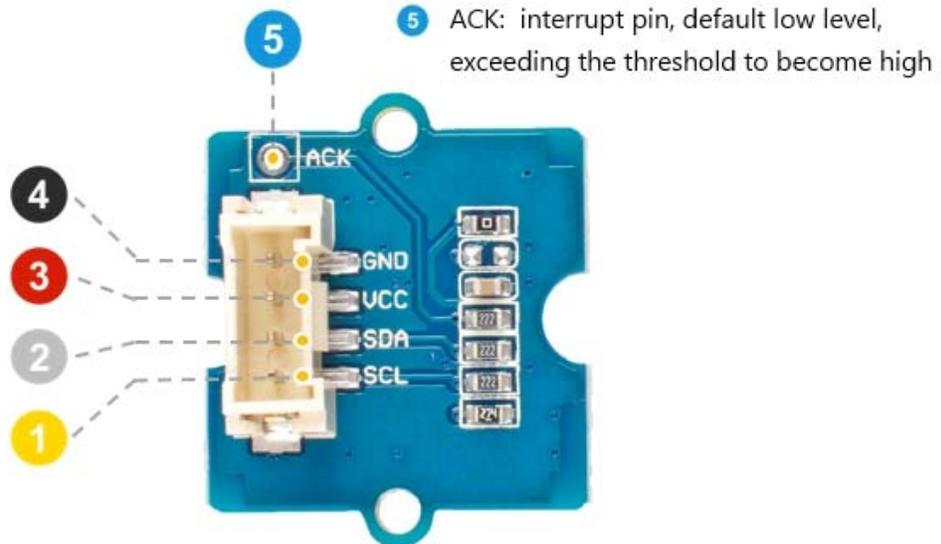
4 GND: connect this module to the system GND
3 VCC: you can use 5V or 3.3V for this module
2 SDA: I²C serial data
1 SCL: I²C serial clock

The UV Detect Module

I2C UV Sensor (VEML6070) v1.0

Platforms Supported

Arduino	Raspberry Pi	BeagleBone	Wio	LinkIt ONE

Caution

The platforms mentioned above as supported is/are an indication of the module's hardware 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

Hardware

Materials required

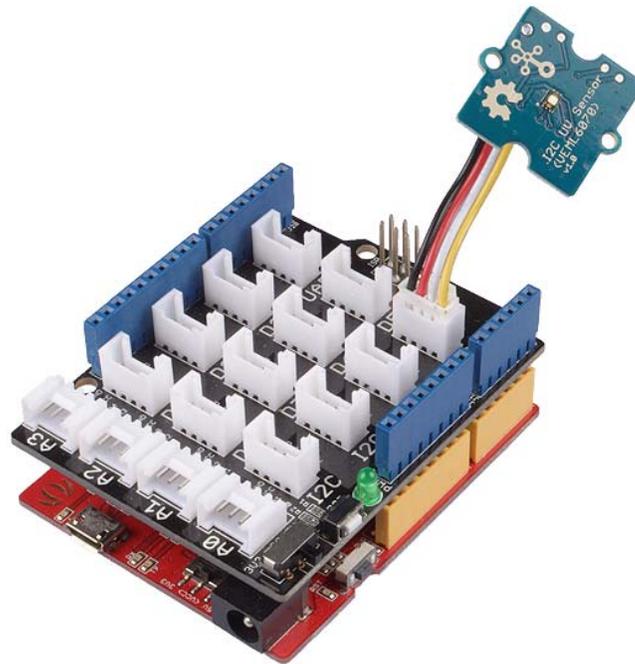| Seeeduino V4.2	Base Shield	Grove - I2C UV Sensor (VEML6070)

Note

1 Please plug the USB cable gently, otherwise you may damage the port. Please use the USB cable with 4 wires inside, the 2 wires cable can't transfer data. If you are not sure about the wire you have, you can click here to buy

2 Each Grove module comes with a Grove cable when you buy. In case you lose the Grove cable, you can click here to buy.

- **Step 1.** Connect the Grove - I2C UV Sensor (VEML6070) to port **I²C** of Grove-Base Shield.

- **Step 2.** Plug Grove - Base Shield into Seeeduino.
- **Step 3.** Connect Seeeduino to PC via a USB cable.

Seeeduino	Grove Cable	Grove - I2C UV Sensor (VEML6070)
GND	Black	GND
5V or 3.3V	Red	VCC
SDA	White	SDA
SCL	Yellow	SCL

Software

- **Step 1.** Download the Seeed_VEML6070 Library from Github.
- **Step 2.** Refer to How to install library to install library for Arduino.
- **Step 3.** Restart the Arduino IDE. Open the example, you can open it in the following three ways :

a. Open it directly in the Arduino IDE via the path: **File → Examples → Seeed_VEML6070 → INT_mode**.
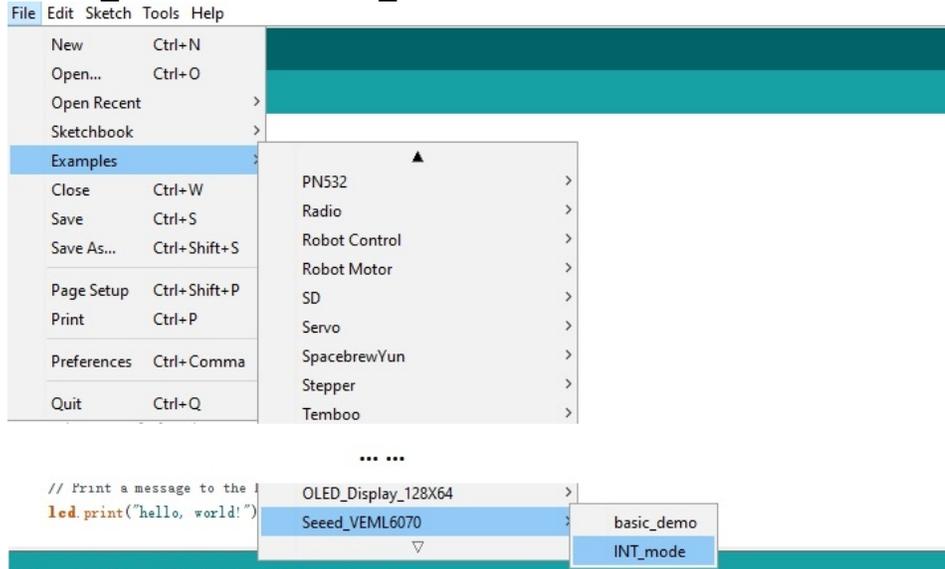
b. Open it in your computer by click the **INT_mode.ino** which you can find in the folder **XXXX\Arduino\libraries\Seeed_VEML6070-master\examples\INT_mode**, **XXXX** is the location you installed the Arduino IDE.
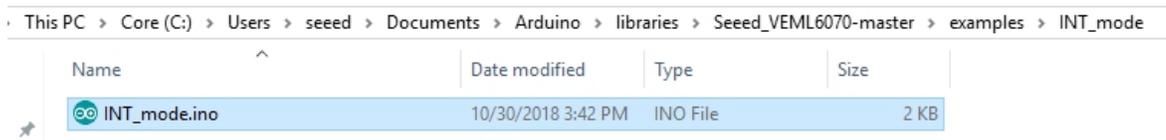
c. Or, you can just click the icon ⬚ in upper right corner of the code block to copy the following code into a new sketch in the Arduino IDE.

```
1  #include "Seeed_VEML6070.h"
2
3
4  /*SAMD core*/
5  #ifdef ARDUINO_SAMD_VARIANT_COMPLIANCE
6    #define SERIAL SerialUSB
7  #else
8    #define SERIAL Serial
9  #endif
10
11 VEML6070 sensor;
12
13 char *UV_str[]={"low level","moderate level","high_level","very high","extreme"};
14
15 err_t read_UV()
16 {
17   err_t ret=NO_ERROR;
18   u16 step;
19   sensor.wait_for_ready();
20   CHECK_RESULT(ret,sensor.read_step(step));
```

```
21    SERIAL.print("UV step = ");
22    SERIAL.println(step);
23    RISK_LEVEL level=sensor.convert_to_risk_level(step);
24    SERIAL.print("UV level is ");
25    SERIAL.println(UV_str[level]);
26    SERIAL.println(" ");
27    SERIAL.println(" ");
28    SERIAL.println(" ");
29    return ret;
30 }
31
32
33
34 void setup()
35 {
36    SERIAL.begin(115200);
37    delay(10);
38    SERIAL.println("serial start!!");
39    delay(1000);
40    if(sensor.init())
41    {
42      SERIAL.println("init failed!!!");
43    }
44    /*threshold is 145 steps*/
45    /*enable*/
46    sensor.set_interrupt(INT_145_STEP,ENABLE);
47 }
48
49
50 void loop()
51 {
52    if(read_UV())
53    {
54      SERIAL.print("read UV sensor failed!!");
55    }
56    //sensor.enable();
57    //sensor.disable();
58    delay(1000);
59 }
```

Note
There are 2 demos in the library:
basic_demo.ino
This example can get the UV index and UV level from the serial.polling for data.
INT_mode.ino
here is a INT pad on the sensor module which connect to ACK pin of VEML6070. You can set UV threshold by 102 steps or 145 steps(only two choises).The INT pin outputs low when the UV value beyond limit.You can attach INT pin to a interrupt pin of host,To improve the efficiency of program operation.
Attention
The library file may be updated. This code may not be applicable to the updated library file, so we recommend that you use the first two methods.

- **Step 4.** Upload the demo. If you do not know how to upload the code, please check How to upload code.

- **Step 5.** Open the **Serial Monitor** of Arduino IDE by click **Tool-> Serial Monitor**. Or tap the `Ctrl` + `Shift` + `M` key at the same time. Set the baud rate to **115200**.

Success
If every thing goes well, when you use UV light to illuminate this module, you will see information like:

```
1 4serial start!!
2 cmd reg=32
3 UV step = 20
4 UV level is low level
5
6
7
8 UV step = 125
9 UV level is low level
10
11
12
13 UV step = 511
14 UV level is low level
```

Alert Function

As we can see in the hardware Overview part, there is a ACK pin for user to work as a interrupt signal. The good news is that there are two threshold 102 step and 145 step to choose, the bad one is that you can not set your own number, only 102 and 145 are valid. The ACK pin default output low, once the UV value exceed the threshold you setted, the ACK pin output High. Let's come back to the example code **INT_mode.ino**.

Line 46, the default setting of threshold is 145, if you want to use the 102, you should just change the code as below:

```
1 sensor.set_interrupt(INT_145_STEP,ENABLE);
```

⇓⇓ ⇓⇓

```
1 sensor.set_interrupt(INT_102_STEP,ENABLE);
```

Project

This is the introduction Video of this product, simple demos, you can have a try.

Tech Support

Please do not hesitate to submit the issue into our forum