Grove - I2C High Accuracy Temp&Humi Sensor(SHT35)

Grove - I2C High Accuracy Temp&Humi Sensor(SHT35) is based on SHT3x-DIS, which is the next generation of Sensirion's temperature and humidity sensors. It builds on a new CMOSens® sensor chip that is at the heart of Sensirion's new humidity and temperature platform. The SHT3x-DIS has increased intelligence, reliability and improved accuracy specifications compared to its predecessor. Its functionality includes enhanced signal processing, two distinctive and user selectable I2C addresses and communication speeds of up to 1 MHz.

Features

- High accuracy of ±1.5 %RH and ±0.1 °C
- Fully calibrated, linearized, and temperature compensated digital output
- I2C Interface with communication speeds up to 1MHz and two user selectable addresses
- Very fast start-up and measurement time

Specification

| Item	Value
Operating Voltage	3.3V / 5V
Specified Temperature Range	-40°C to +125°C
Temperature Resolution	0.01°C
Temperature Accuracy Tolerance	±0.1 °C
Specified Humidity Range	0%RH to +100%RH
Humidity Resolution	0.01%RH
Humidity Accuracy Tolerance	±1.5 %RH
Interface	I²C
I²C Address	0x45(default) / 0x44(optional)

Applications

- Industrial Freezers and Refrigerators
- Food Processing
- Personal Computers and Servers
- PC Peripherals
- Consumer Electronics
- Handheld/Portable Devices

Hardware Overview

Pin Out
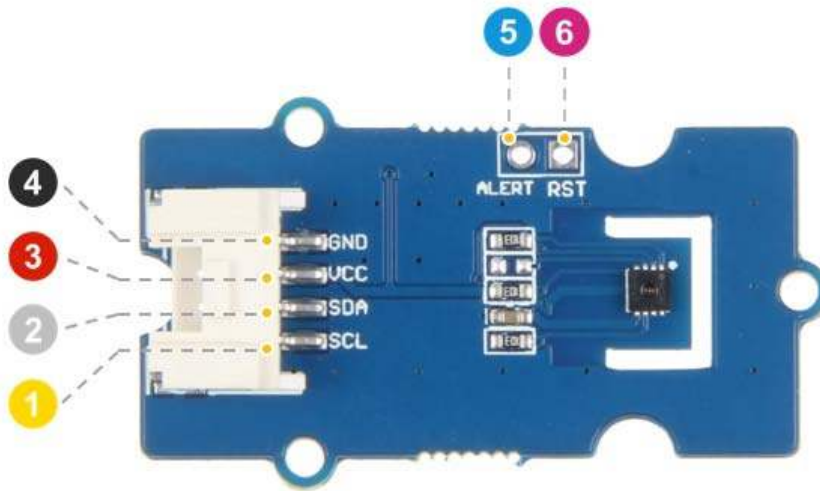
④ GND: connect this module to the system GND
③ VCC: you can use 5V or 3.3V for this module
② SDA: I²C serial data
① SCL: I²C serial clock

⑤ ALERT: reserved interrupt pin for customer use
⑥ RST: connect to the reset pin

0 x 44 optional

0 x 45 default
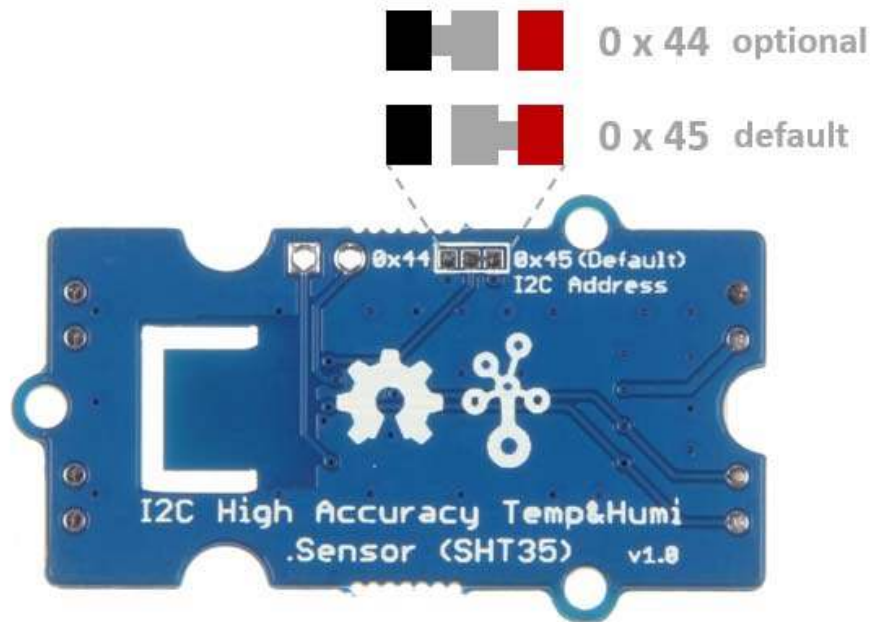
Schemaitc

Power

This module is based on **SHT35**, the input voltage of this chip range from 2.15v-5.5v, so you can use both 3.3v and 5v pin of Arduino to supply for this module.

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-SHT35 Sensor

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 High Accuracy Temp&Humi Sensor(SHT35) 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.
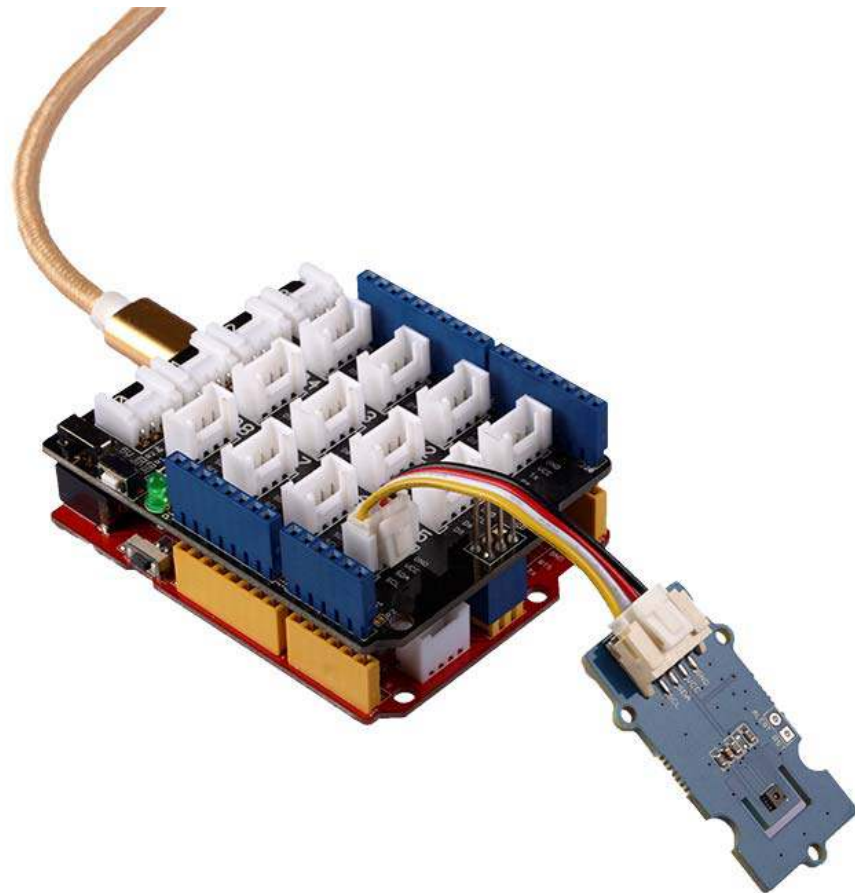
Note
If we don't have Grove Base Shield, We also can directly connect this module to Seeeduino as below.

| Seeeduino	Grove Cable	Grove - I2C High Accuracy Temp&Humi Sensor(SHT35)
GND	Black	GND
5V or 3.3V	Red	VCC
SDA	White	SDA
SCL	Yellow	SCL

Software

Attention

If this is the first time you work with Arduino, we strongly recommend you to see Getting Started with Arduinobefore the start.

- **Step 1.** Download the Grove_touch_sensor_CY8C40XX 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 → Grove Temperature sensor SHT35 → basic_demo**.
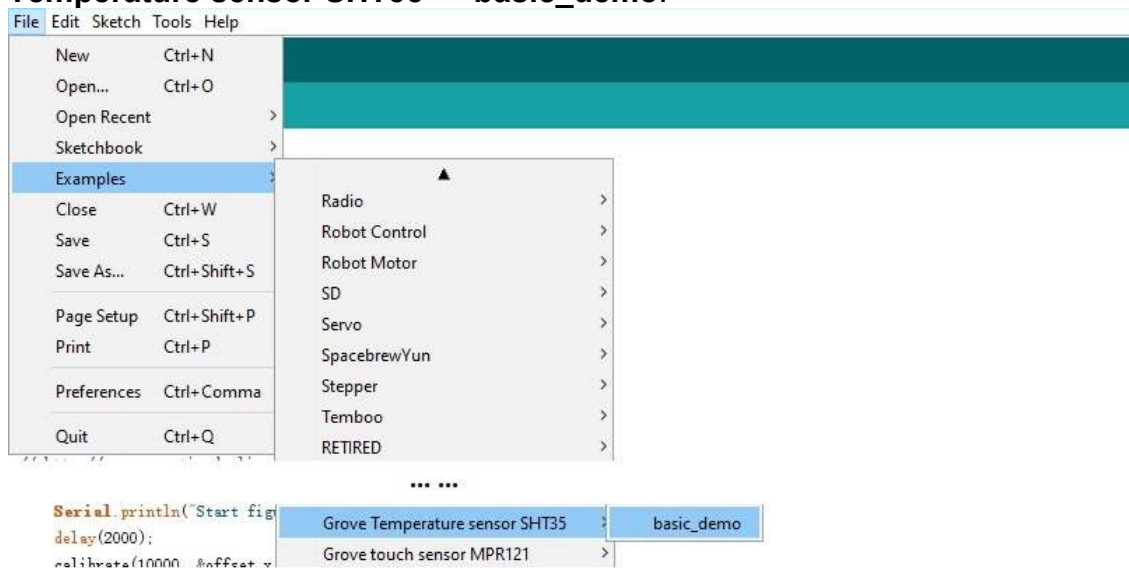
  b. Open it in your computer by click the **basic_demo.ino** which you can find in the folder **XXXX\Arduino\libraries\Seeed_SHT35-master\examples\basic_demo**, **XXXX** is the location you installed the Arduino IDE.
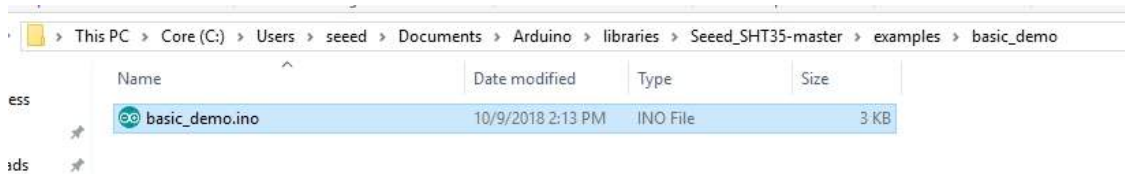
  c. Or, you can just click the icon in upper right corner of the code block to copy the following code into a new sketch in the Arduino IDE.

```cpp
1 #include "Seeed_SHT35.h"
2
3
4 /*SAMD core*/
5 #ifdef ARDUINO_SAMD_VARIANT_COMPLIANCE
6  #define SDAPIN  20
7  #define SCLPIN  21
8  #define RSTPIN  7
9  #define SERIAL SerialUSB
10 #else
11  #define SDAPIN  A4
12  #define SCLPIN  A5
13  #define RSTPIN  2
14  #define SERIAL Serial
15 #endif
16
17 SHT35 sensor(SCLPIN);
18
19
20 void setup()
21 {
22    SERIAL.begin(115200);
23    delay(10);
24    SERIAL.println("serial start!!");
25    if(sensor.init())
26    {
27      SERIAL.println("sensor init failed!!!");
28    }
29    delay(1000);
30 }
31
32
33 void loop()
34 {
35    u16 value=0;
36    u8 data[6]={0};
37    float temp,hum;
38    if(NO_ERROR!=sensor.read_meas_data_single_shot(HIGH_REP_WITH_STRCH,&temp,&hum))
39    {
40      SERIAL.println("read temp failed!!");
41      SERIAL.println("   ");
42      SERIAL.println("   ");
43      SERIAL.println("   ");
44    }
45    else
46    {
47      SERIAL.println("result======>");
48      SERIAL.print("temperature =");
49      SERIAL.println(temp);
50
51      SERIAL.print("humidity =");
52      SERIAL.println(hum);
53
54      SERIAL.println("   ");
55      SERIAL.println("   ");
56      SERIAL.println("   ");
```

```
57  }
58  delay(1000);
59}
```

- **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 open the Serial Monitor , it may show as below:

```
1 serial start!!
2 =>
3 temperature =24.10
4 humidity =51.09
5
6
7 result======>
8 temperature =24.10
9 humidity =50.96
10
11
12 result======>
13 temperature =24.10
14 humidity =51.04
15
16
17 result======>
18 temperature =24.11
19 humidity =51.09
```

Tech Support

Please do not hesitate to submit the issue into our forum