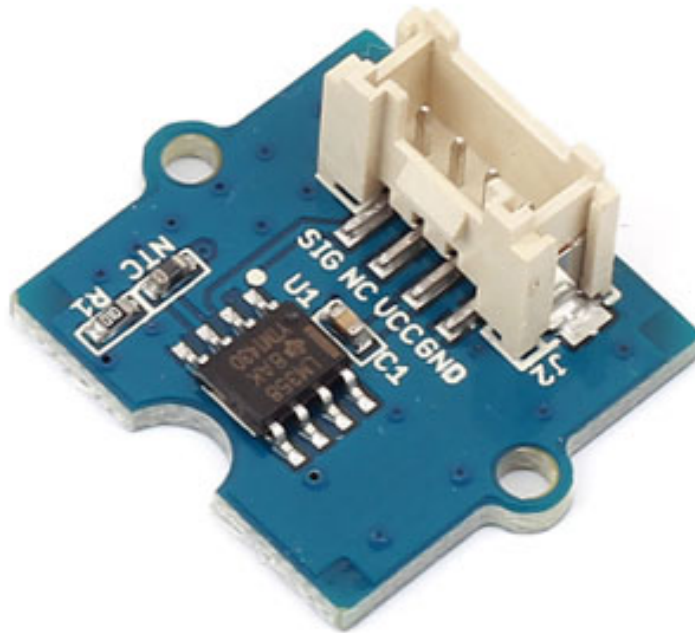


Grove - Temperature Sensor V1.2



The Grove - Temperature Sensor uses a **Thermistor** [https://files.seeedstudio.com/wiki/Grove-Temperature_Sensor_V1.2/res/NCP18WF104F03RC.pdf] to detect the ambient temperature. The resistance of a thermistor will increase when the ambient temperature decreases. It's this characteristic that we use to calculate the ambient temperature.

The detectable range of this sensor is -40 - 125°C, and the accuracy is $\pm 1.5^\circ\text{C}$

Note: This wiki works with Grove - Temperature sensor V1.1 as well, for V1.0 please refer to [Grove - Temperature Sensor](#) [https://wiki.seeedstudio.com/Grove-Temperature_Sensor]

[Get One Now](#) 

[<https://www.seeedstudio.com/Grove-Temperature-Sensor-p-774.html>]

Specifications

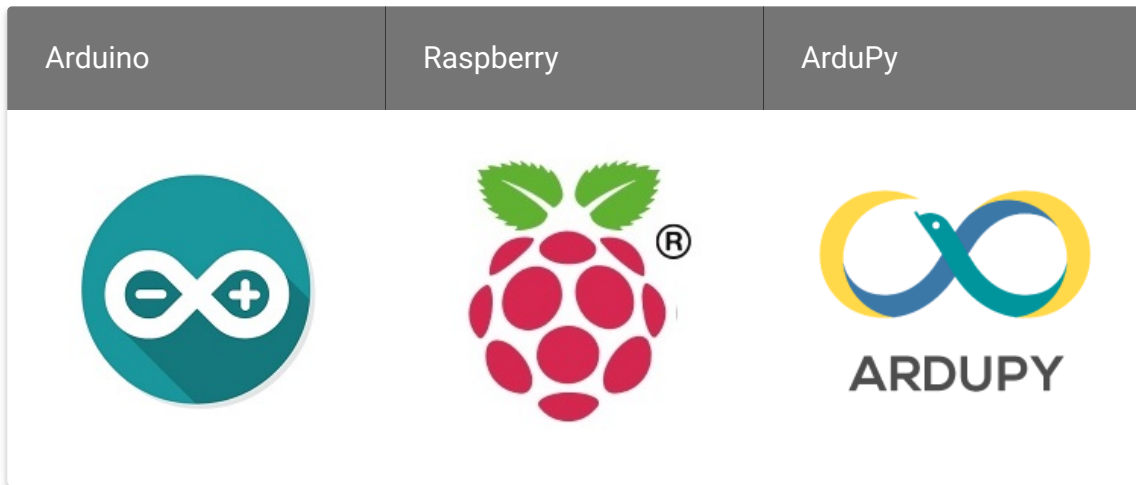
- Voltage: 3.3 ~ 5V
- Zero power resistance: 100 K Ω
- Resistance Tolerance: $\pm 1\%$
- Operating temperature range: -40 ~ +125 °C
- Nominal B-Constant: 4250 ~ 4299K



Tip

More details about Grove modules please refer to [Grove System](#) [https://wiki.seeedstudio.com/Grove_System/]

Platforms Supported

**Caution**

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

After this section, you can make Grove - Temperature Sensor V1.1/1.2 run with only few steps.



**Note**

If this is the first time you work with Arduino, we firmly recommend you to see [Getting Started with Arduino](https://wiki.seeedstudio.com/Getting_Started_with_Arduino/) [https://wiki.seeedstudio.com/Getting_Started_with_Arduino/] before the start.

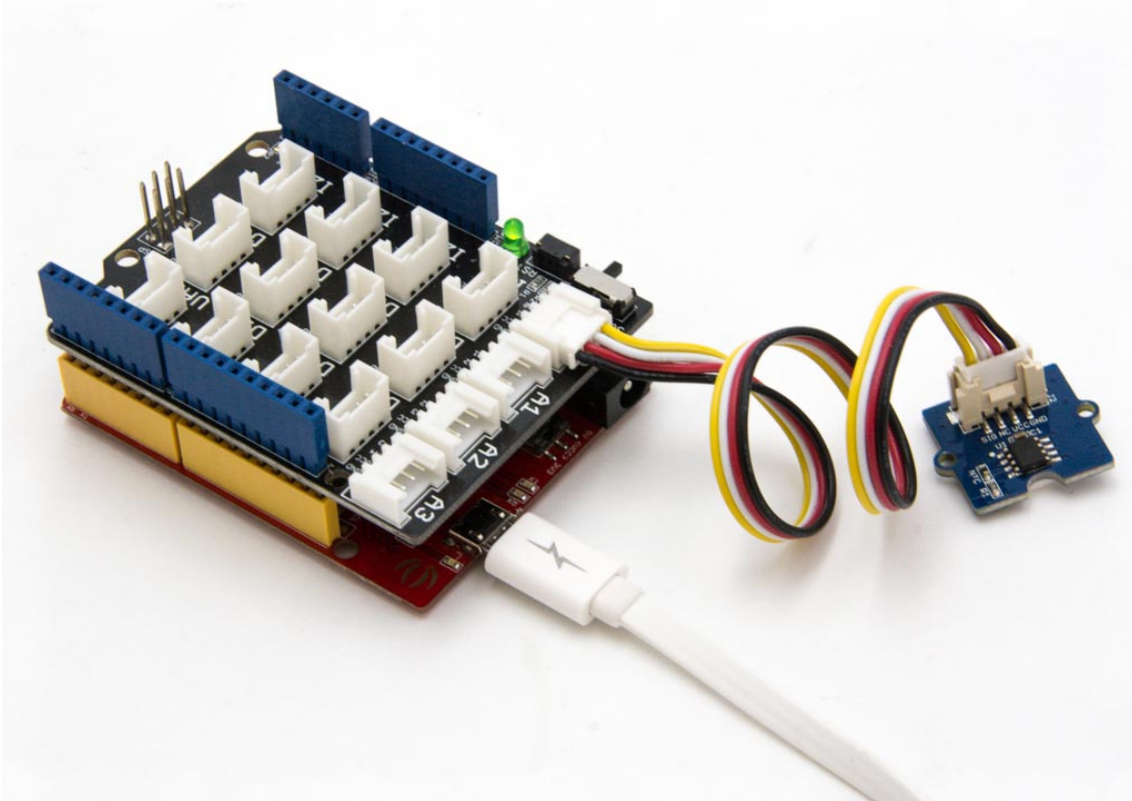
Play With Arduino

Hardware

- **Step 1.** Prepare the below stuffs:

| Seeeduino V4.2 | Base Shield |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  |  |
| <p>Get One Now [https://www.seeedstudio.com/Seeeduino-V4.2-p-2517.html]</p> | <p>Get One Now [https://www.seeedstudio.com/Base-Shield-V2-p-1378.html]</p> |

- **Step 2.** Connect Grove - Temperature Sensor to port **A0** of Grove-Base Shield.
- **Step 3.** Plug Grove - Base Shield into Seeeduino.
- **Step 4.** Connect Seeeduino to PC via a USB cable.



Note

If we don't have Grove Base Shield, We also can directly connect Grove_Ultrasonic_Ranger to Seeduino as below.

| Seeduino | Grove - Temperature Sensor |
|---------------|----------------------------|
| 5V | Red |
| GND | Black |
| Not Conencted | White |
| A0 | Yellow |

Software

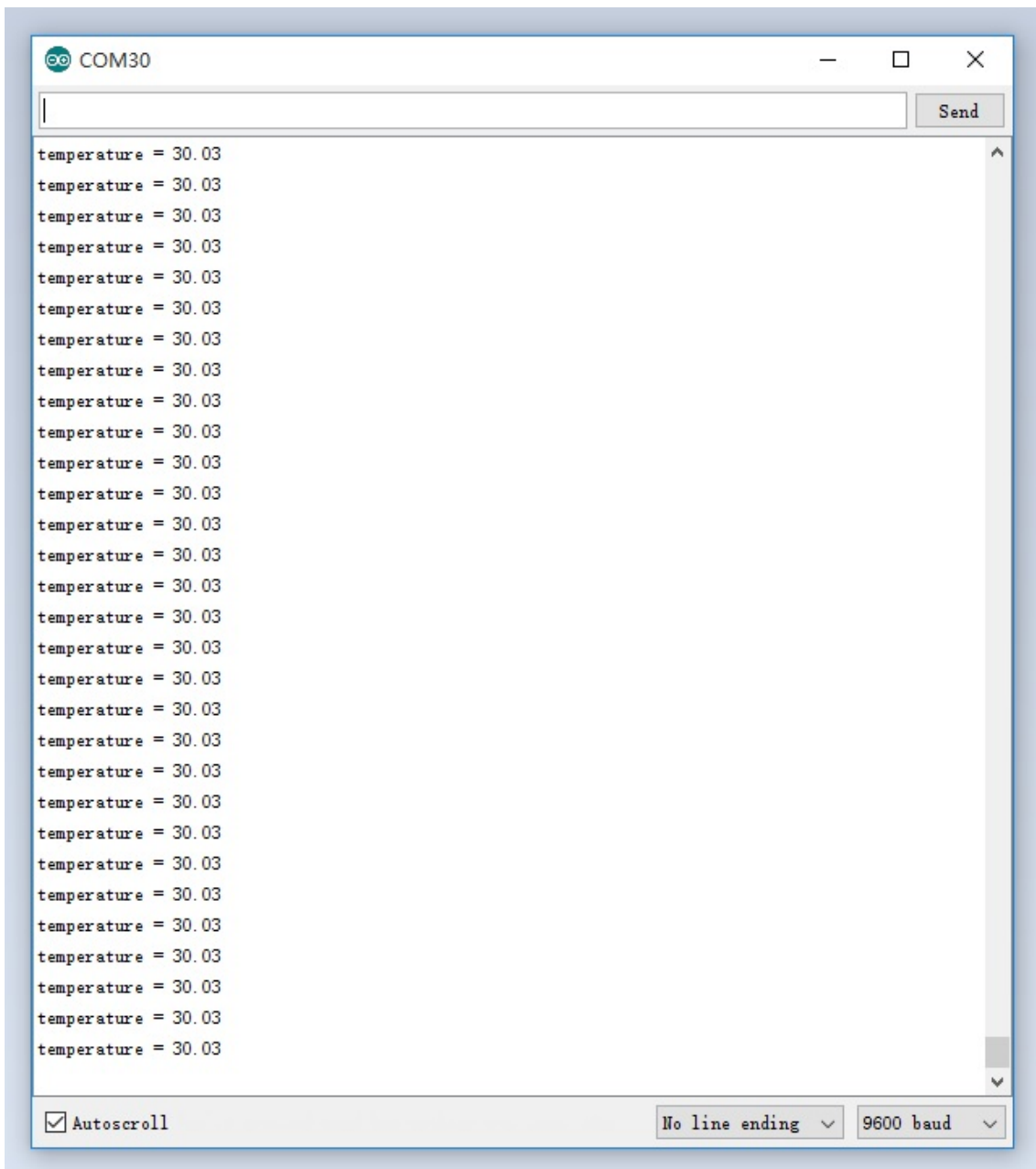
- **Step 1.** Launch Arduino IDE and click **File>New** to open a new page. Copy the following code into the new page and upload. If you do not know how to upload the code, please check [How to upload code](https://wiki.seeedstudio.com/Upload_Code/) [https://wiki.seeedstudio.com/Upload_Code/].

```
1 // Demo code for Grove - Temperature Sensor V1.1/1.2
2 // Loovee @ 2015-8-26
3
4 #include <math.h>
5
6 const int B = 4275; // B value of the thermistor
7 const int R0 = 100000; // R0 = 100k
8 const int pinTempSensor = A0; // Grove - Temperature Sensor
9
10 #if defined(ARDUINO_ARCH_AVR)
11 #define debug Serial
12 #elif defined(ARDUINO_ARCH_SAMD) || defined(ARDUINO_ARCH_ESP8266)
13 #define debug SerialUSB
14 #else
15 #define debug Serial
16 #endif
17
18 void setup()
19 {
20     Serial.begin(9600);
21 }
22
23 void loop()
24 {
25     int a = analogRead(pinTempSensor);
26
27     float R = 1023.0/a-1.0;
28     R = R0*R;
29
30     float temperature = 1.0/(log(R/R0)/B+1/298.15)-273.15;
31
32     Serial.print("temperature = ");
33     Serial.println(temperature);
34
```

```
35     delay(100);  
36 }
```

Step 2. Open the **Serial Monitor** of Arduino IDE by click **Tool->Serial Monitor**. Or tap the **Ctrl+Shift+M** key at the same time. if every thing goes well, you will get the temperature.


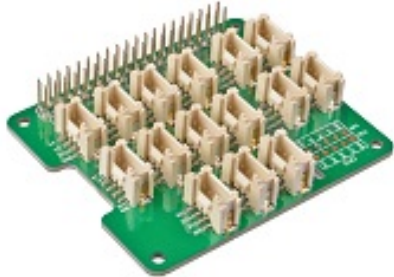
The result should be like:



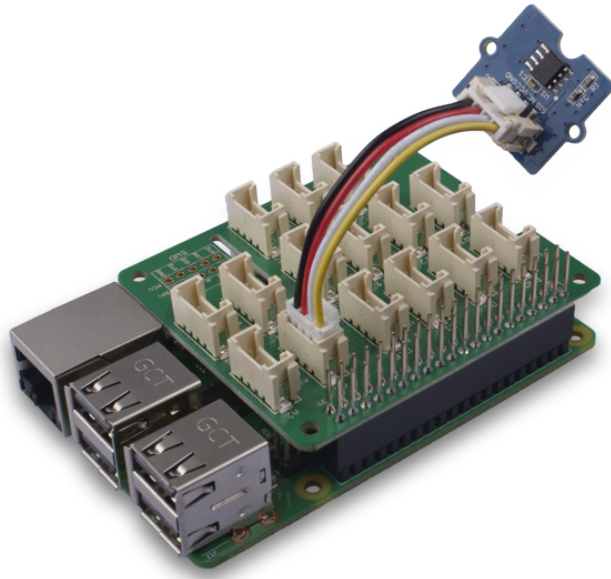
Play With Raspberry Pi (With Grove Base Hat for Raspberry Pi)

Hardware

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

| Raspberry pi | Grove Base Hat for RasPi |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  |  |
| <p>Get ONE Now [https://www.seeedstudio.com/Raspberry-Pi-3-Model-B-p-2625.html]</p> | <p>Get ONE Now [https://www.seeedstudio.com/Grove-Base-Hat-for-Raspberry-Pi-p-3186.html]</p> |

- **Step 2.** Plug the Grove Base Hat into Raspberry.
- **Step 3.** Connect the temperature sensor 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 temperature sensor to **any analog Port** but make sure you change the command with the corresponding port number.

Software



Attention

If you are using **Raspberry Pi with Raspberrypi OS >= Bullseye**, you have to use this command line **only with Python3**.

- **Step 1.** Follow [Setting Software](https://wiki.seeedstudio.com/Grove_Base_Hat_for_Raspberry_Pi/#installation) [https://wiki.seeedstudio.com/Grove_Base_Hat_for_Raspberry_Pi/#installation] to configure the development environment.
- **Step 2.** Download the source file by cloning the grove.py library.

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

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

```
1 cd grove.py/grove
2 python3 grove_temperature_sensor.py 0
```

Following is the grove_temperature_sensor.py code.

```
1 import sys
2 import time
3 from grove.factory import Factory
4
5
6 def main():
7     from grove.helper import SlotHelper
8     sh = SlotHelper(SlotHelper.ADC)
9     pin = sh.argv2pin()
10
11     sensor = Factory.getTemper("NTC-ADC", pin)
12
13     print('Detecting temperature...')
14     while True:
15         print('{} Celsius'.format(sensor.temperature))
16         time.sleep(1)
17
18
19 if __name__ == '__main__':
20     main()
```



Success

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



```
1 pi@raspberrypi:~/grove.py/grove $ python3 grove_temperati
2 Hat Name = 'Grove Base Hat RPi'
3 Detecting temperature...
4 24.7473402633 Celsius
5 24.7473402633 Celsius
6 24.7473402633 Celsius
7 24.7112751977 Celsius
8 24.7112751977 Celsius
9 ^CTraceback (most recent call last):
10   File "grove_temperature_sensor.py", line 53, in <modul
11     main()
12   File "grove_temperature_sensor.py", line 49, in main
13     time.sleep(1)
14 KeyboardInterrupt
```

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 **A1**, **A0**, 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.

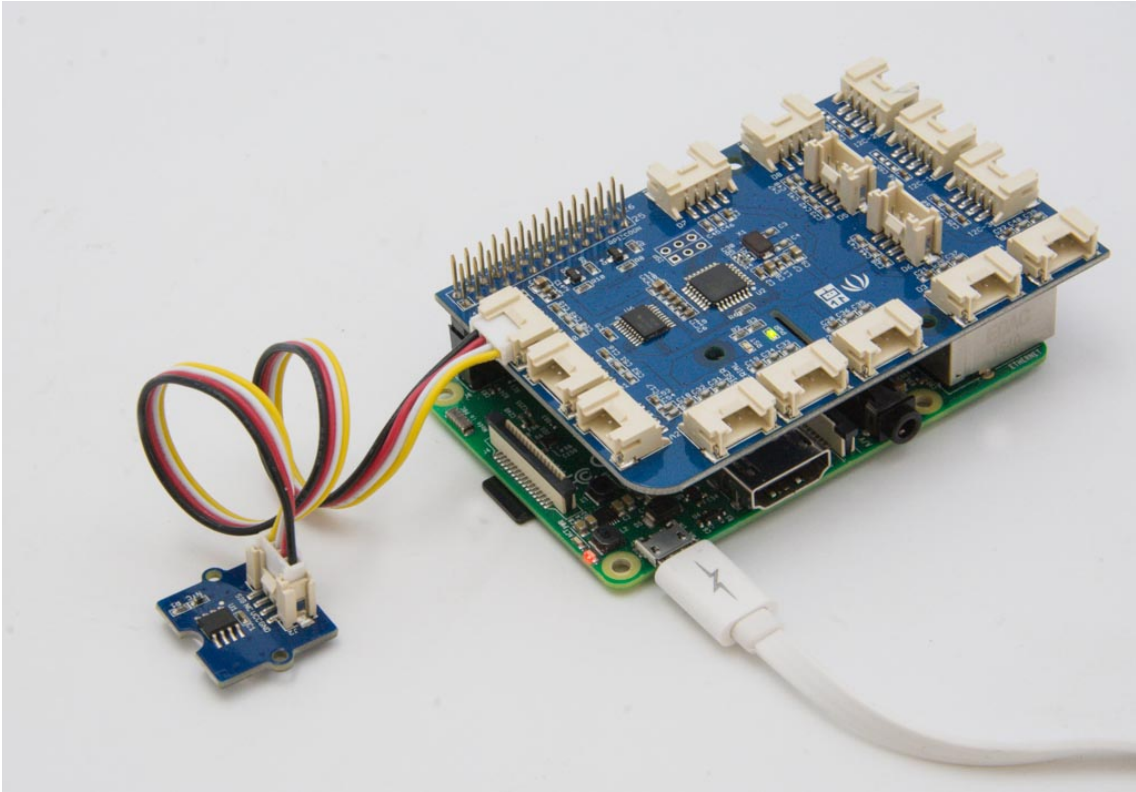
Play With Raspberry Pi (with GrovePi_Plus)

Hardware

- **Step 1.** Prepare the below stuffs:

| Raspberry pi | GrovePi_Plus |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|
|  |  |
| <p>Get One Now [https://www.seeedstudio.com/Raspberry-Pi-3-Model-B-p-2625.html]</p> | <p>Get One Now [https://www.seeedstudio.com/GrovePi-Plus-p-2241.html]</p> |

- **Step 2.** Plug the GrovePi_Plus into Raspberry.
- **Step 3.** Connect Grove - Temperature Sensor ranger to **A0** port of GrovePi_Plus.
- **Step 4.** Connect the Raspberry to PC via USB cable.



Software

- **Step 1.** Follow [Setting Software](https://www.dexterindustries.com/GrovePi/get-started-with-the-grovepi/setting-software/) [https://www.dexterindustries.com/GrovePi/get-started-with-the-grovepi/setting-software/] to configure the development environment.
- **Step 2.** Follow [Updating the Firmware](https://www.dexterindustries.com/GrovePi/get-started-with-the-grovepi/updating-firmware/) [https://www.dexterindustries.com/GrovePi/get-started-with-the-grovepi/updating-firmware/] to update the newest firmware of GrovePi.



Tip

In this wiki we use the path `~/GrovePi/` instead of `/home/pi/Desktop/GrovePi`, you need to make sure Step 2 and Step 3 use the same path.



Note

We firmly suggest you to update the firmware, or for some sensors you may get errors.



Attention

If you are using **Raspberry Pi with Raspberrypi OS >= Bullseye**, you have to use this command line **only with Python3**.

- **Step 3.** Git clone the Github repository.

```
1 cd ~
2 git clone https://github.com/DexterInd/GrovePi.git
```

- **Step 4.** Excute below commands to use the Grove - Temperature Sensor to meansure the temperature.

```
1 cd ~/GrovePi/Software/Python
2 sudo python3 grove_temperature_sensor.py
```

Here is the grove_temperature_sensor.py code.

```
1 # NOTE:
2 # The sensor uses a thermistor to detect ambient tempe.
3 # The resistance of a thermistor will increase when th
4 #
5 # There are 3 revisions 1.0, 1.1 and 1.2, each using a
6 # Each thermistor datasheet specifies a unique Nominal
7 #
8 # The second argument in the grovepi.temp() method def
9 # Defaults to '1.0'. eg.
10 # temp = grovepi.temp(sensor) # B value = 3.
11 # temp = grovepi.temp(sensor, '1.1') # B value = 4.
12 # temp = grovepi.temp(sensor, '1.2') # B value = 4.
13
14 import time
```

```
15 import grovepi
16
17 # Connect the Grove Temperature Sensor to analog port A0
18 # SIG,NC,VCC,GND
19 sensor = 0
20
21 while True:
22     try:
23         temp = grovepi.temp(sensor,'1.2')
24         print("temp =", temp)
25         time.sleep(.5)
26
27     except KeyboardInterrupt:
28         break
29     except IOError:
30         print ("Error")
```



The result should be like:

```
1 pi@raspberrypi:~/GrovePi/Software/Python $ sudo python3
2
3 ('temp =', 25.28652137917777)
4 ('temp =', 25.28652137917777)
5 ('temp =', 25.28652137917777)
6 ('temp =', 25.28652137917777)
7 ('temp =', 25.368489566400115)
8 ('temp =', 25.61468397498203)
9 ('temp =', 27.43501590142614)
10 ('temp =', 27.85285590636829)
11 ('temp =', 27.18509952680688)
12 ('temp =', 26.852756540240193)
```

Play With Wio Terminal (ArduPy)

Hardware

- **Step 1.** Prepare the below stuffs:

| Wio Terminal | Grove - Temperature Sensor |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  |  |
| <p>Get One Now [https://www.seeedstudio.com/Wio-Terminal-p-4509.html]</p> | <p>Get One Now [https://www.seeedstudio.com/Grove-Temperature-Sensor-p-774.html]</p> |

- **Step 2.** Connect Grove - Temperature Sensor to **A0** port of Wio Terminal.
- **Step 3.** Connect the Wio Terminal to PC through USB Type-C cable.



Software

- **Step 1.** Follow [ArduPy Getting Started](https://wiki.seeedstudio.com/ArduPy/) [https://wiki.seeedstudio.com/ArduPy/] to configure the ArduPy development environment on Wio Terminal.
- **Step 2.** Make sure that the ArduPy firmware is flashed into Wio Terminal. For more information, please follow [here](https://wiki.seeedstudio.com/ArduPy/#ardupy-aip-cli-getting-started) [https://wiki.seeedstudio.com/ArduPy/#ardupy-aip-cli-getting-started].

```
1 aip build
2 aip flash
```

- **Step 3.** Copy the following code and save it as `ArduPy-temp.py` :

```
1 from machine import Pin, ADC
2 from machine import LCD
3 from machine import Sprite
4 import time, math
```

```

5
6 raw = ADC(Pin(13))
7 lcd = LCD()
8 spr = Sprite(lcd) # Create a buff
9 B = 4275 # B value of the thermistor
10 R0 = 100000 # R0 = 100k
11
12 def temp(reading):
13     R = 1023.0 / reading - 1.0
14     R = R0*R
15     temperature = 1.0/(math.log(R/R0)/B+1/298.15)-273.15
16     return temperature
17
18 def main():
19     spr.createSprite(320, 240)
20     while True:
21         spr.setTextSize(2)
22         spr.fillSprite(spr.color.BLACK)
23         spr.setTextColor(lcd.color.ORANGE)
24         spr.drawString("Temperature Reading", 45, 10)
25         spr.drawFastHLine(40, 35, 240, lcd.color.DARKGREY)
26         spr.setTextColor(lcd.color.WHITE)
27         spr.drawString("- ", 20, 50)
28         spr.drawFloat(temp(raw.read()), 3, 40,50)
29         spr.drawString("C", 120, 50)
30         spr.pushSprite(0,0)
31         time.sleep_ms(500)
32
33         print("Temperature: ", temp(raw.read()), "C")
34
35 if __name__ == "__main__":
36     main()

```


- **Step 4.** Save the `ArduPy-temp.py` in a location that you know. Run the following command and **replace** `<YourPythonFilePath>` with your `ArduPy-temp.py` location.

```
1 aip shell -n -c "runfile <YourPythonFilePath>"
```



```
2 # Example:
3 # aip shell -n -c "runfile /Users/ansonhe/Desktop/ArduPy-
```

- **Step 5.** We will see the temperature value display on terminal as below, and displaying on the Wio Terminal LCD screen.

```
1 ansonhe@Ansons-Macbook-Pro ~: aip shell -n -c "runfile 
2 Positional argument (/dev/cu.usbmodem141101) takes precedence
3 Connected to ardupy
4 Temperature: 28.08603 C
5 Temperature: 28.50415 C
6 Temperature: 28.16953 C
7 Temperature: 28.25308 C
8 Temperature: 28.08603 C
9 Temperature: 28.16953 C
10 Temperature: 28.08603 C
11 Temperature: 28.16953 C
12 Temperature: 28.33671 C
13 Temperature: 28.16953 C
14 Temperature: 28.25308 C
```



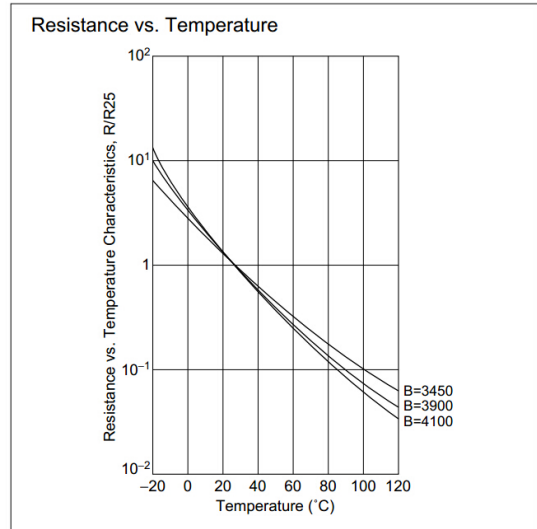
Reference

If you want to know how the algorithm of temperature coming, please refer to the below image:

1. Zero-power Resistance of Thermistor: R
 $R=R_0 \exp B (1/T-1/T_0)$ (1)
 R: Resistance in ambient temperature T (K)
 (K: absolute temperature)
 R₀: Resistance in ambient temperature T₀ (K)
 B: B-Constant of Thermistor

2. B-Constant
 as (1) formula
 $B= \ell n (R/R_0) / (1/T-1/T_0)$ (2)

3. Thermal Dissipation Constant
 When electric power P (mW) is spent in ambient temperature T₁ and thermistor temperature rises T₂, there is a formula as follows
 $P=C (T_2-T_1)$ (3)
 C: Thermal dissipation constant (mW/°C)
 Thermal dissipation constant is varied with dimensions, measurement conditions, etc.



Schematic Online Viewer



Resources

- **[Zip]** [Grove - Temperature Sensor v1.1 Eagle File](https://files.seeedstudio.com/wiki/Grove-Temperature_Sensor_V1.2/res/Grove_-_Temperature_sensor_v1.1.zip)
[https://files.seeedstudio.com/wiki/Grove-Temperature_Sensor_V1.2/res/Grove_-_Temperature_sensor_v1.1.zip]
- **[PDF]** [Grove - Temperature Sensor v1.1.PDF](https://files.seeedstudio.com/wiki/Grove-)
[https://files.seeedstudio.com/wiki/Grove-

Temperature_Sensor_V1.2/res/Grove_-
_Temperature_sensor_v1.1.pdf]

- **[PDF]** [Temperature Sensor datasheet](https://files.seeedstudio.com/wiki/Grove-Temperature_Sensor_V1.2/res/NCP18WF104F03RC.pdf)
[https://files.seeedstudio.com/wiki/Grove-
Temperature_Sensor_V1.2/res/NCP18WF104F03RC.pdf]

Projects

Temperature sensor Grove module:

Lesson 11 : Temperature se...



Lección 11 : Módulo Grove ...



Tech Support

Please submit any technical issue into our [forum](https://forum.seeedstudio.com/)
[<https://forum.seeedstudio.com/>].



[https://www.seeedstudio.com/act-4.html?utm_source=wiki&utm_medium=wikibanner&utm_campaign=newproducts]