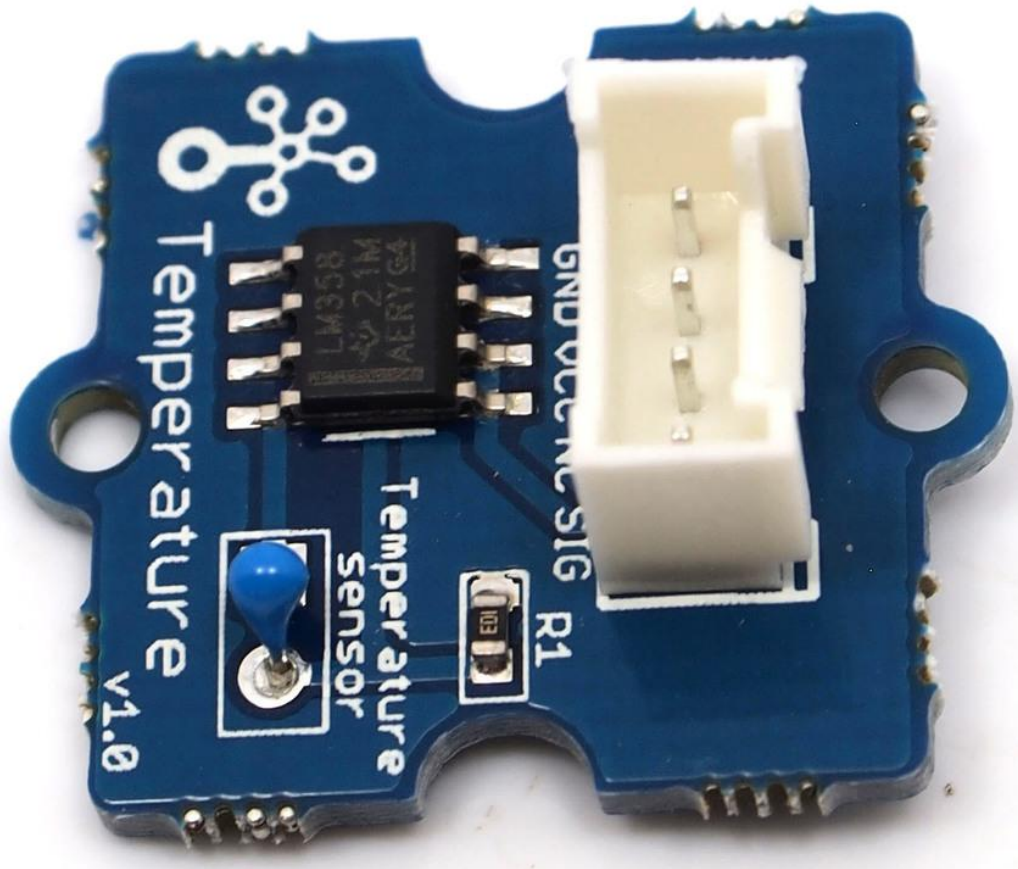
Grove - Temperature Sensor SKU: 101020015



Grove - 温度传感器使用热敏电阻来检测环境温度。 当环境温度降低时，热敏电阻的电阻将增加。 这是我们用来计算环境温度的这个特点。 该传感器的可检测范围为-40 - 125ºC，精度为±1.5ºC。

产品参数

- 工作电压: 3.3 ~ 5V
- 在25℃下，最大额定功率：300mW
- 零功率电阻：10KΩ
- 工作温度范围：-40∽+125℃

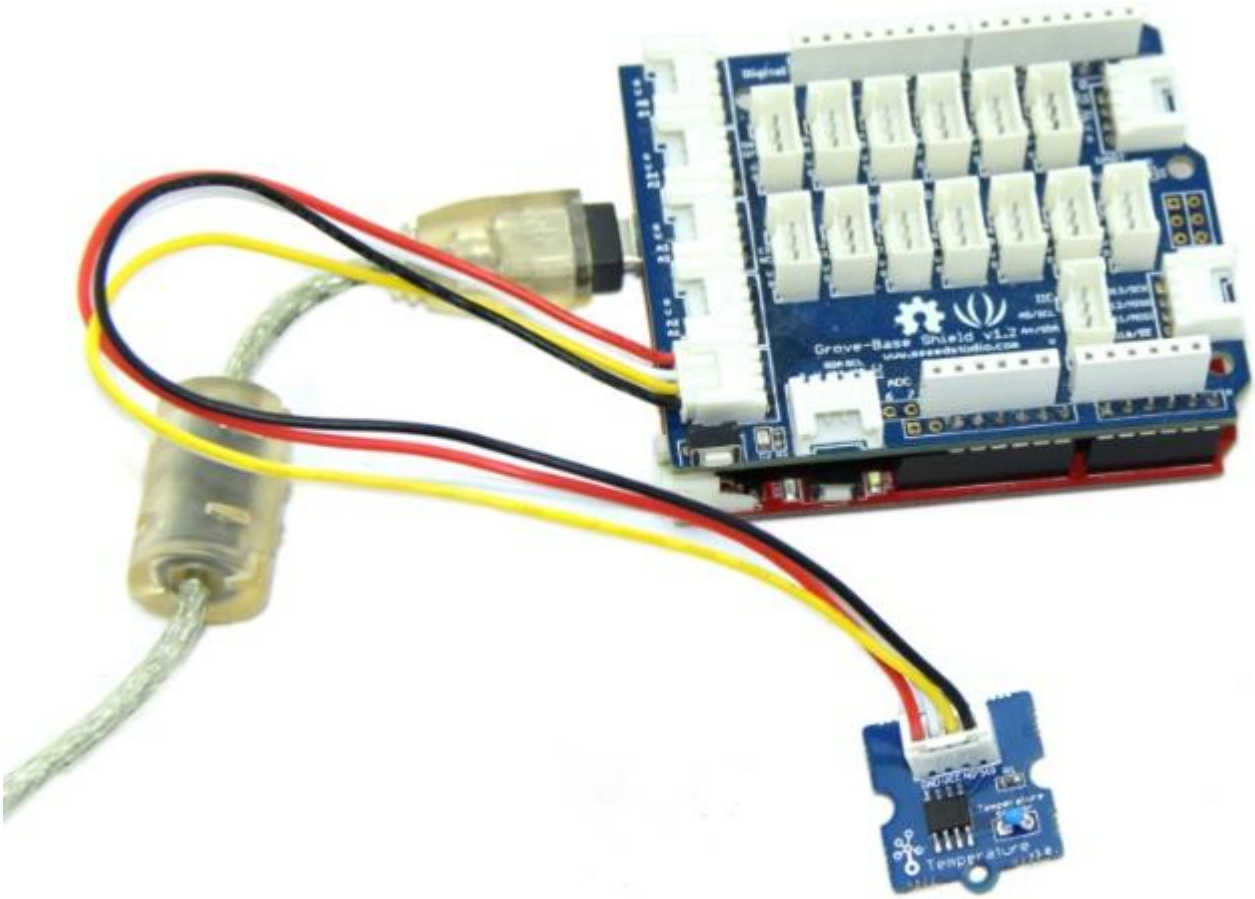
示范

使用Arduino

以下是一个示例，说明如何从传感器读取温度信息。

1.使用4针Grove连接线将模块连接到Grove - Basic Shield的模拟端口 **A0** 上。

2.将Grove-Basic Shield插入Arduino。 3.使用USB数据线将Arduino连接到PC。

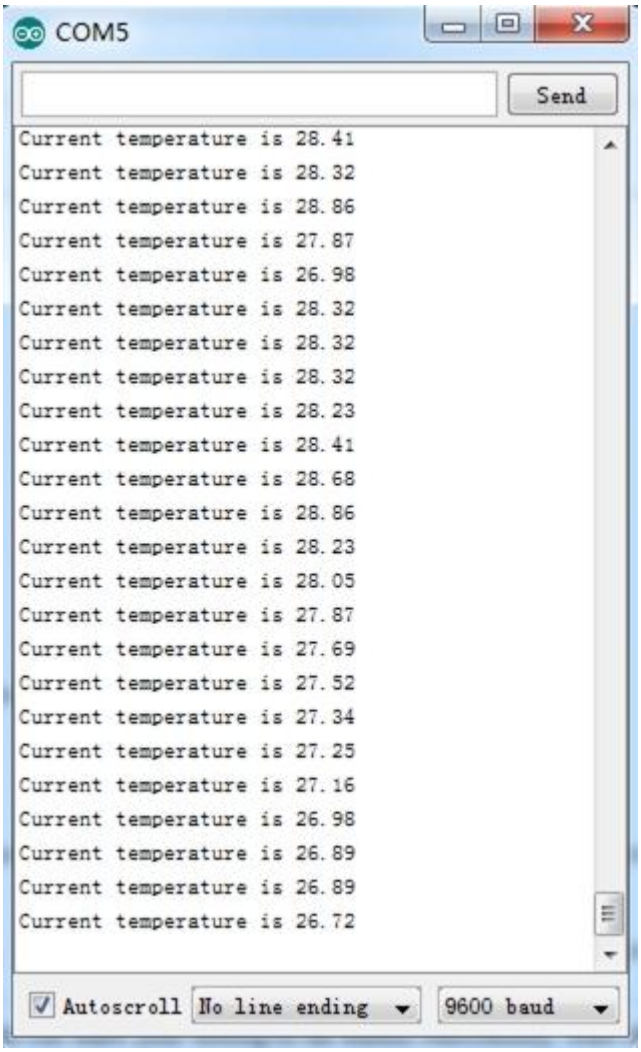


4.下载以下程序到你的板子.如果您不清楚怎么下载代码到您的板子里，请点击这里。

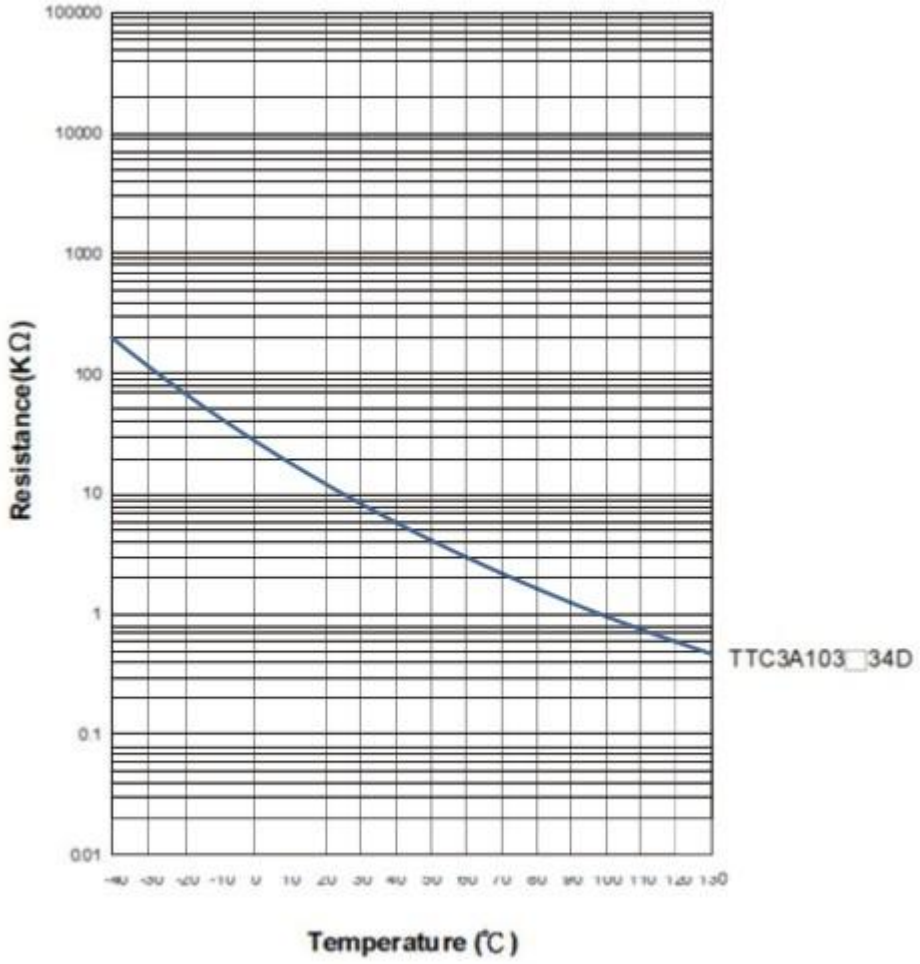
```
/*
/* Grove - Temperature Sensor demo v1.0
*  This sensor detects the environment temperature,
*  Connect the signal of this sensor to A0, use the
*  Serial monitor to get the result.
*  By: http://www.seeedstudio.com
*/
#include &lt;math.h&gt;
int a;
float temperature;
int B=3975;                  //B value of the thermistor
float resistance;

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

```
void loop()
{
    a=analogRead(0);
    resistance=(float)(1023-a)*10000/a; //get the resistance of the sensor;
    temperature=1/(log(resistance/10000)/B+1/298.15)-273.15;//convert to
temperature via datasheet ;
    delay(1000);
    Serial.print("Current temperature is ");
    Serial.println(temperature);
}
```

5.您可以通过串行监视器检查读数。 默认单位为摄氏度。



作为参考，以下是我们在该传感器上使用的热敏电阻TTC3A103 * 39H的电阻曲线。 温度越高，电阻越小。



使用 Raspberry Pi

1.你应该有一个raspberry pi和一个grovepi或grovepi +。

2.您需要完成配置开发环境，否则遵循 说明 完成配置

3.硬件连接

- 使用grove连接线将传感器插入Grovepi插座 **D3**。

4.浏览演示目录：

```
cd yourpath/GrovePi/Software/Python/
```

- 找到到这行代码

```
nano grove_temperature_sensor.py   # "Ctrl+x" to exit #
```

```
import time
import grovepi
```

```
#Connect the Grove Temperature Sensor to analog port A0
#SIG,NC,VCC,GND
sensor = 0

while True:
try:
temp = grovepi.temp(sensor,'1.1')
print "temp =", temp
time.sleep(.5)

except KeyboardInterrupt:
break
except IOError:
print "Error"
```

5.运行这个示例

```
sudo python grove_temperature_sensor.py
```

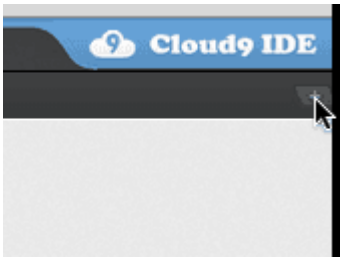
使用 Beaglebone Green

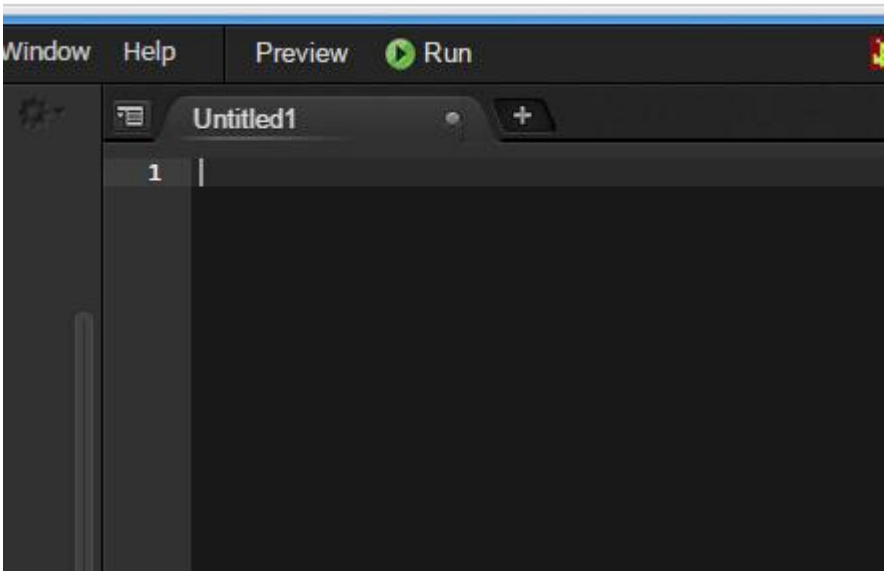
要开始编辑BBG上的程序，可以使用Cloud9 IDE。

为了熟悉Cloud9 IDE，我们进行一些简单的练习，先创建一个简单的应用程序来闪烁BeagleBone上的4个可编程LED灯，这是学习编程一个好的开始。

如果这是您第一次使用Cloud9 IDE，请参考 这里。

第一步: 点击右上角的"**+**"创建一个新文件。





第二步: 将以下代码复制并粘贴到新选项卡中

```
from Adafruit_I2C import Adafruit_I2C
import time

ADDR_ADC121 = 0x50

REG_ADDR_RESULT = 0x00
REG_ADDR_ALERT = 0x01
REG_ADDR_CONFIG = 0x02
REG_ADDR_LIMITL = 0x03
REG_ADDR_LIMITH = 0x04
REG_ADDR_HYST = 0x05
REG_ADDR_CONVL = 0x06
REG_ADDR_CONVH = 0x07

i2c = Adafruit_I2C(ADDR_ADC121)

class I2cAdc:
def __init__(self):
i2c.write8(REG_ADDR_CONFIG, 0x20)

def read_adc(self):
"Read ADC data 0-4095."
data_list = i2c.readList(REG_ADDR_RESULT, 2)
        #print 'data list', data_list
data = ((data_list[0] & 0x0f) << 8 | data_list[1]) & 0xfff
return data

if __name__ == '__main__':
    # Connect the Grove - I2C ADC to I2C Grove port of Beaglebone Green.
adc = I2cAdc()
while True:
print 'sensor value ', adc.read_adc()
time.sleep(.2)
```

第三步: 通过单击名称为以“grove_i2c_adc.py”命名保存文件

第四步: 创建新文件将以下代码复制到新选项卡中，并使用.py扩展名保存。

```
import time
import math
import grove_i2c_adc
import Adafruit_BBIO.GPIO as GPIO

BUZZER = "P9_22"            # GPIO P9_22
GPIO.setup(BUZZER, GPIO.OUT)

# The threshold to turn the buzzer on 28 Celsius
THRESHOLD_TEMPERATURE = 28

adc = grove_i2c_adc.I2cAdc()

#   The argument in the read_temperature() method defines which Grove board(Grove
Temperature Sensor) version you have connected.
#   Defaults to 'v1.2'. eg.
#       temp = read_temperature('v1.0')          # B value = 3975
#       temp = read_temperature('v1.1')          # B value = 4250
#       temp = read_temperature('v1.2')          # B value = 4250
def read_temperature(model = 'v1.2'):
"Read temperature values in Celsius from Grove Temperature Sensor"
    # each of the sensor revisions use different thermistors, each with their own
B value constant
if model == 'v1.2':
bValue = 4250  # sensor v1.2 uses thermistor ??? (assuming NCP18WF104F03RC until
SeeedStudio clarifies)
elif model == 'v1.1':
bValue = 4250  # sensor v1.1 uses thermistor NCP18WF104F03RC
else:
bValue = 3975  # sensor v1.0 uses thermistor TTC3A103*39H

total_value = 0
for index in range(20):
sensor_value = adc.read_adc()
total_value += sensor_value
time.sleep(0.05)
average_value = float(total_value / 20)

    # Transform the ADC data into the data of Arduino platform.
sensor_value_tmp = (float)(average_value / 4095 * 2.95 * 2 / 3.3 * 1023)
resistance = (float)(1023 - sensor_value_tmp) * 10000 / sensor_value_tmp
temperature = round((float)(1 / (math.log(resistance / 10000) / bValue + 1 /
298.15) - 273.15), 2)
return temperature

# Function: If the temperature sensor senses the temperature that is up to the
threshold you set in the code, the buzzer is ringing for 1s.
```

```
# Hardware: Grove - I2C ADC, Grove - Temperature Sensor, Grove - Buzzer
# Note: Use P9_22(UART2_RXD) as GPIO.
# Connect the Grove Buzzer to UART Grove port of Beaglebone Green.
# Connect the Grove - I2C ADC to I2C Grove port of Beaglebone Green, and then
connect the Grove - Temperature Sensor to Grove - I2C ADC.
if __name__ == '__main__':

while True:
try:
            # Read temperature values in Celsius from Grove Temperature Sensor
temperature = read_temperature('v1.2')

            # When the temperature reached predetermined value, buzzer is ringing.

print "temperature = ", temperature

except IOError:
print "Error"
```

第5步: 将Grove温度传感器连接到的Grove I2C端口I2C ADC。

第6步: 运行代码 你会发现终端每2秒输出一次温度值。

资源下载

- Grove - Temperature Sensor v1.0 Eagle File
- Demo code on github