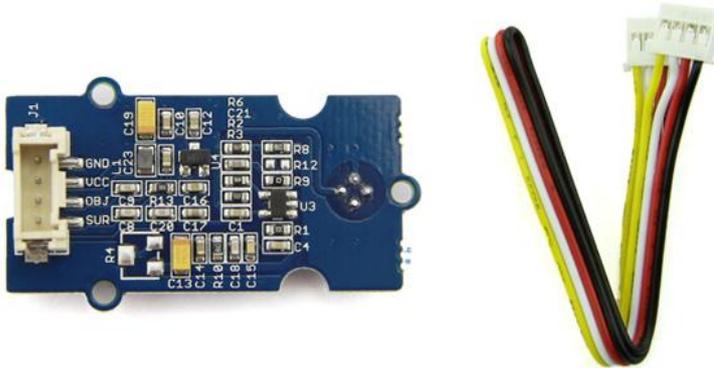


Grove - Infrared Temperature Sensor

Introduction

3.3V 5.0V Analog



The Infrared temperature sensor is a non-contact temperature measure model. It is composed of 116 elements of thermocouple in series on a floating micro-membrane, the black Surface of the sensor is good to absorb the incident thermal infrared radiation, which might trigger a voltage response at output. This sensor outputs an analog voltage (0~1.1V) according to target temperature.

Old version: v0.92.

Specifications

- Voltage: 3-5V
- Measuring Current Supply: 160-200 uA
- Measuring Range: -10~100°C
- Holding Time: 2S
- Operating Temperature: -10~80 °C
- Storage Temperature: -35-80 °C

Tip

More details about Grove modules please refer to [Grove System](#)

Platforms Supported

Arduino	Wio	BeagleBone	Raspberry Pi	LinkIt ONE
				

Caution

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

Demonstration

The following sketch demonstrates a simple application of measure the measure the surrounding temperature around the sensor and the temperature of the target which is in front of the sensor. And print the result on the serial monitor.

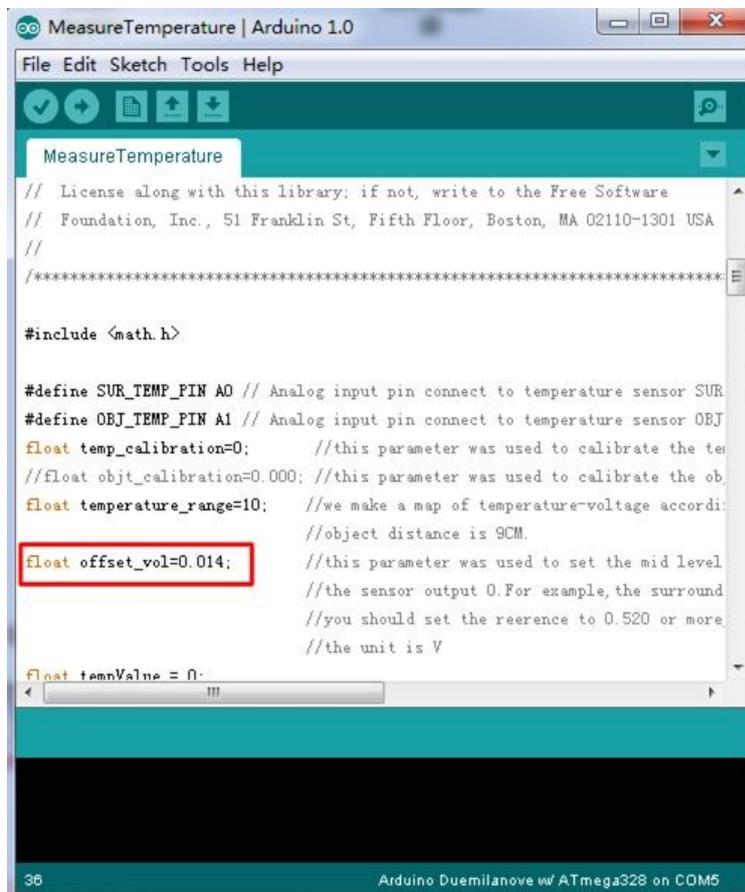
- Connect this module to Seeeduno using Grove-Base Shield port A0 and A1.
- Download the [Demo Code](#) and open it.

Before measuring temperature, you need simple setting. Following the instruction below before your test and you will gain a accurately result.

Step 1: Regulate the sensor voltage

After uploading the demo program, make the sensor in normal environment more than 5 minutes that make the sensor temperature the same as surrounding temperature. Then open the serial monitor to check the voltage which the sensor output. Ideally, when the ambient temperature is equal to the temperature sensor, the infrared sensor (TP-538U) output is 0V. We should regulate the reference voltage which offset at 0.5V by hardware. As shown below, the sensor voltage is 0.014V, we

just need to change the offset_vol value as 0.014 which you obtain from the serial monitor in program.

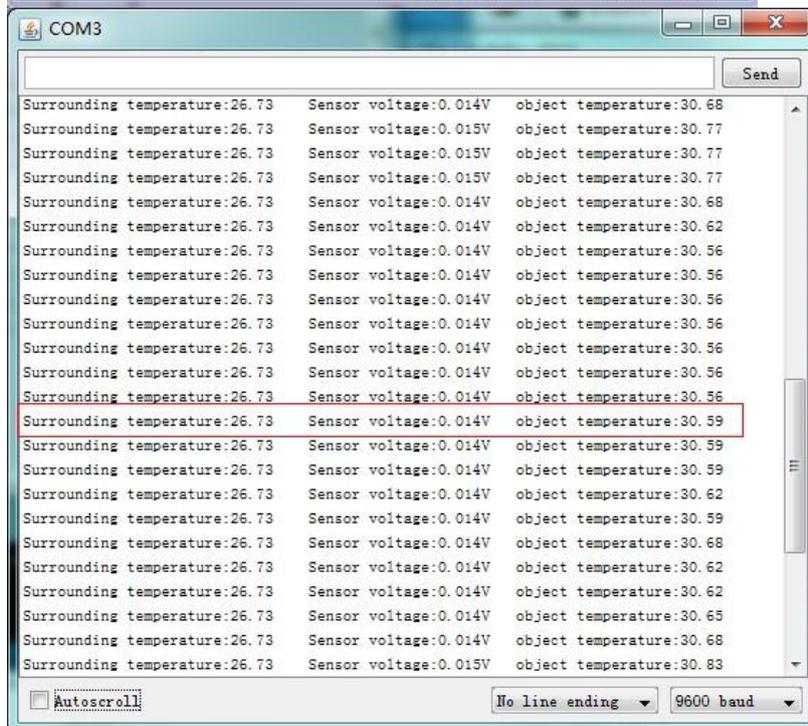


```
MeasureTemperature | Arduino 1.0
File Edit Sketch Tools Help
MeasureTemperature
// License along with this library: if not, write to the Free Software
// Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
//
/*****

#include <math.h>

#define SUR_TEMP_PIN A0 // Analog input pin connect to temperature sensor SUR
#define OBJ_TEMP_PIN A1 // Analog input pin connect to temperature sensor OBJ
float temp_calibration=0; //this parameter was used to calibrate the te
//float objt_calibration=0.000; //this parameter was used to calibrate the ob
float temperature_range=10; //we make a map of temperature-voltage accordi
//object distance is 9CM.
float offset_vol=0.014; //this parameter was used to set the mid level
//the sensor output 0. For example, the surround
//you should set the reerence to 0.520 or more
//the unit is V

float tempValue = 0;
```



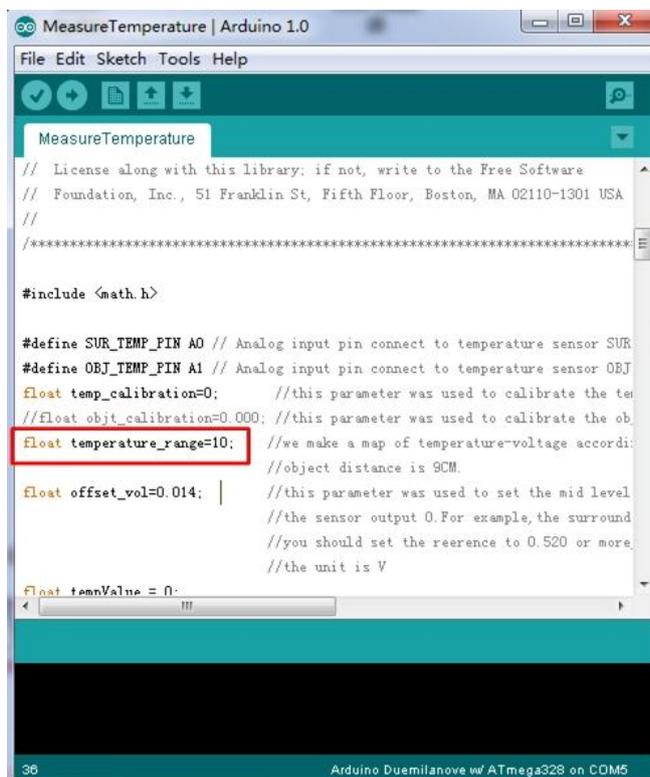
```
COM3
Send
Surrounding temperature:26.73 Sensor voltage:0.014V object temperature:30.68
Surrounding temperature:26.73 Sensor voltage:0.015V object temperature:30.77
Surrounding temperature:26.73 Sensor voltage:0.015V object temperature:30.77
Surrounding temperature:26.73 Sensor voltage:0.015V object temperature:30.77
Surrounding temperature:26.73 Sensor voltage:0.014V object temperature:30.68
Surrounding temperature:26.73 Sensor voltage:0.014V object temperature:30.62
Surrounding temperature:26.73 Sensor voltage:0.014V object temperature:30.56
Surrounding temperature:26.73 Sensor voltage:0.014V object temperature:30.59
Surrounding temperature:26.73 Sensor voltage:0.014V object temperature:30.59
Surrounding temperature:26.73 Sensor voltage:0.014V object temperature:30.59
Surrounding temperature:26.73 Sensor voltage:0.014V object temperature:30.62
Surrounding temperature:26.73 Sensor voltage:0.014V object temperature:30.59
Surrounding temperature:26.73 Sensor voltage:0.014V object temperature:30.68
Surrounding temperature:26.73 Sensor voltage:0.014V object temperature:30.62
Surrounding temperature:26.73 Sensor voltage:0.014V object temperature:30.62
Surrounding temperature:26.73 Sensor voltage:0.014V object temperature:30.65
Surrounding temperature:26.73 Sensor voltage:0.014V object temperature:30.68
Surrounding temperature:26.73 Sensor voltage:0.015V object temperature:30.83
```

Step 2: Regulate the sensor detected distance

According to our experiment, the sensor's nominal measure distance is 9CM, but we can't ensure all sensor have the same character. So if you want to have accurate result, you need to regulate it with Ice and water mixture to measure the 0°C, and use boiled water to regulate the 100°C. After that, you can obtain the effective distance of the sensor.

The specific method for measuring is fill with ice and water in a dark container which has a flat surface. Waiting for the container drop to 0°C, keep the sensor 9CM between object, move the sensor forward or back and check the result, if output is 0°C, note down the distance value. The same method to check boiled water. When you gain a pair of value, make a deal with average calculation. You can begin to measure in a rated distance which you just obtained now.

Now we can measure the surrounding temperature around the sensor. The sensor is apply in a nominal distance, you can have a try in other distance, but the distance-temperature diagram neither sensor's manufacturer nor we obtained, you can draw it follow the two instructions above. We reserve variable "**temperature_range**" in demo code. We assume that the target distance is 3 cm, the coefficient which you measured may be 5 more or less. Wish you have a fun try.

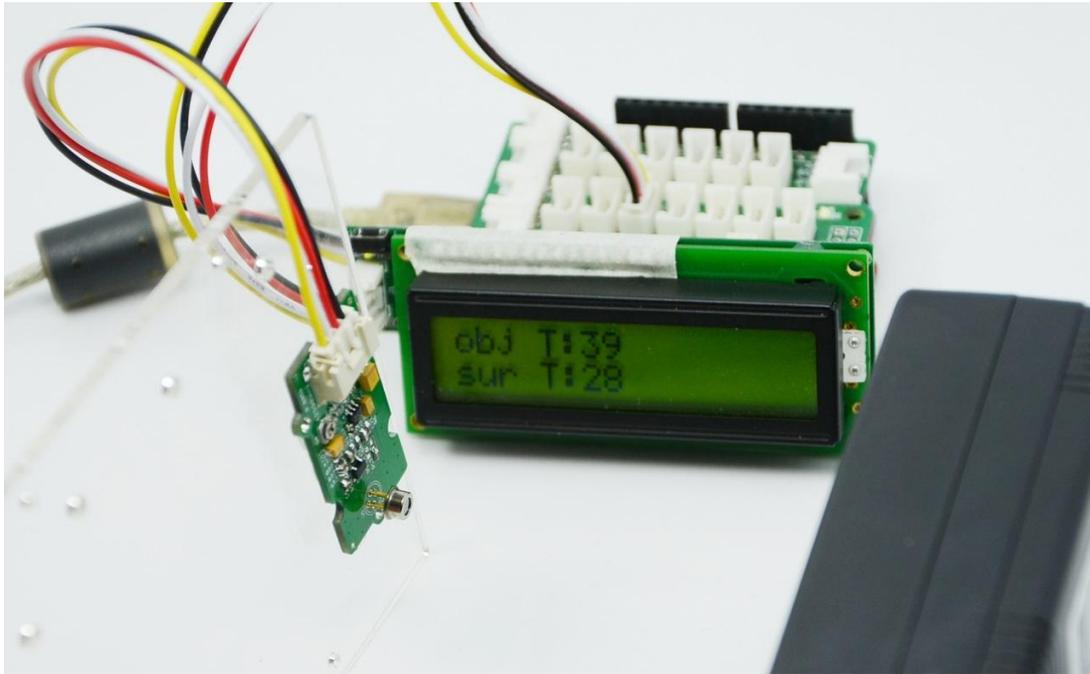


```
MeasureTemperature | Arduino 1.0
File Edit Sketch Tools Help
MeasureTemperature
// License along with this library; if not, write to the Free Software
// Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
//
//*****
#include <math.h>

#define SUR_TEMP_PIN AO // Analog input pin connect to temperature sensor SUR
#define OBJ_TEMP_PIN A1 // Analog input pin connect to temperature sensor OBJ
float temp_calibration=0; //this parameter was used to calibrate the te
//float objt_calibration=0.000; //this parameter was used to calibrate the ob
float temperature_range=10; //we make a map of temperature-voltage accordi:
//object distance is 9CM.
float offset_vol=0.014; //this parameter was used to set the mid level
//the sensor output 0.For example,the surround
//you should set the reerence to 0.520 or more
//the unit is V

float tempValue = 0;
```

Advanced application example:



Note

1. The demo code does not support Atmega168.
2. In order to gain an accurate measurement, the distance(D) and target diameter(S) ratio D:S must be less than 0.5.

