

# IR Thermometer Sensor MLX90614

## SKU: SEN0206

We can divide temperature measurement in to two types: contact and non-contact. Contact measurement can only accurately measure temperature when the testing object and the sensor reach thermal equilibrium. This can mean longer response times and reading inaccuracies offset by ambient temperature. By contrast, non-contact measurement uses infra-red radiation to measure the temperature and does not require a direct touch. Additionally, this method of measurement can be read quickly and accurately.

In recent years non-contact measurement methods have been used for medical, environmental monitoring, home automation, automotive electronic, aerospace and military applications. Our latest infrared temperature measurement module is the MLX90614. This module measures the surface temperature by detecting infrared radiation energy and wavelength distribution. The IR temperature probe consists of an optical system, photoelectric detector, amplifier, signal processing and output module. The optical system collects the infrared radiation in its field of view and the infrared radiation energy is converted in to corresponding electrical signals when converging on the photoelectric detector. After being processed by the amplifier and signal processing circuit, the signal is converted in to a temperature value. The MLX90614 is self calibrating and has a low noise amplifier integrated in to the signal processing chip. The chip itself is a 17 bit ADC and DSP device, giving accurate and reliable results.

# Specification

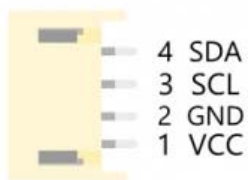
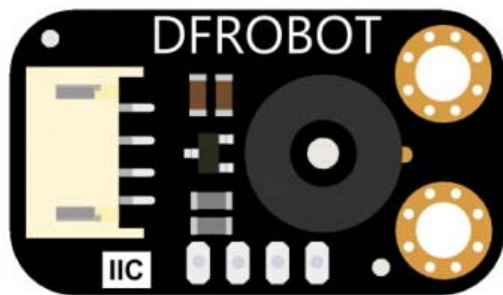
SEN0206 (MLX90614-BBC)

- Operating Voltage: 3.3V - 5V
- Operating Current: 1.2mA
- Temperature: -70.01°C to +382.19°C, (0.01 °C resolution)
- Interface Type: I2C
- Interface Line Sequence: VCC, GND, SCL, SDA
- FOV: 35°
- Dimensions: 31.5\*18 mm/1.24 x 0.7 inches

SEN0263 (MLX90614-DCI)

- Operating Voltage: 3.3V - 5V
- Operating Current: 1.2mA
- Temperature: -70.01°C to +270°C, (0.01 °C resolution)
- Interface Type: I2C
- Interface Line Sequence: VCC, GND, SCL, SDA
- FOV: 5°
- Dimensions: 31.5\*18 mm/1.24 x 0.7 inches

## Pin Outs



IR Thermometer Sensor-MLX90614 Interface Instruction		
Label	Name	Function
1	VCC	Positive Terminal
2	GND	Negative Terminal
3	SCL	I2C Clock Pin
4	SDA	I2C Data Pin

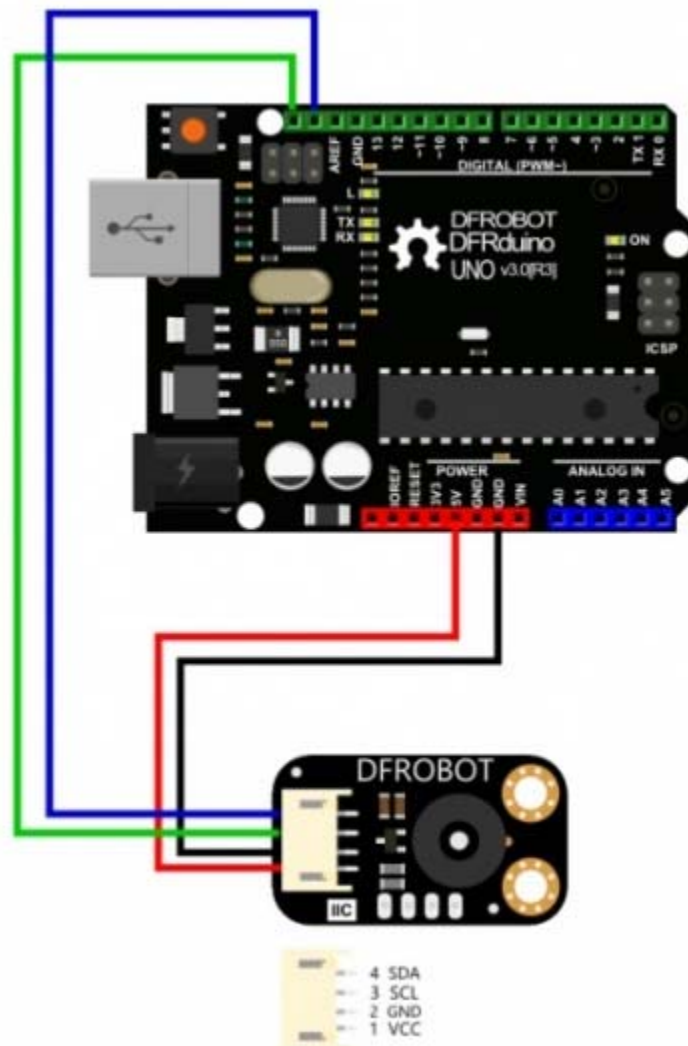
IR Thermometer Sensor-MLX90614

# Tutorial

## Requirements

- **Hardware**
- DFRduino UNO x 1
- IR Thermometer Sensor MLX90614 x 1
- Jumper Wires

## Connection Diagram



IR Thermometer Sensor-MLX90614 Wire Diagram

# Sample Code

Please make sure you have installed libraries before uploading! [Library Files and Examples>About Library installation.](#)

```
/*
 * IR Thermometer Sensor-MLX90614
 *
 * This example will measure the ambient temperature and object temperature through the I2C bus
 *
 * @author jackli(Jack.li@dfrobot.com)
 * @version V1.0
 * @date 2016-2-2
 *
 * GNU Lesser General Public License.
 * See <http://www.gnu.org/licenses/> for details.
 * All above must be included in any redistribution
 */
#include <Wire.h>
#include <IR_Thermometer_Sensor_MLX90614.h>

IR_Thermometer_Sensor_MLX90614 MLX90614 = IR_Thermometer_Sensor_MLX90614();

void setup() {
  Serial.begin(9600);
  MLX90614.begin();
}

void loop() {
  Serial.print("Ambient = "); Serial.print(MLX90614.GetAmbientTemp_Celsius());
  Serial.println(" *C");

  Serial.print("Object = "); Serial.print(MLX90614.GetObjectTemp_Celsius());
  Serial.println(" *C");

  Serial.print("Ambient = "); Serial.print(MLX90614.GetAmbientTemp_Fahrenheit());
  Serial.println(" *F");
}
```

```
Serial.print("Object = "); Serial.print(MLX90614.GetObjectTemp_Fahrenheit(
)); Serial.println(" *F");

Serial.println();

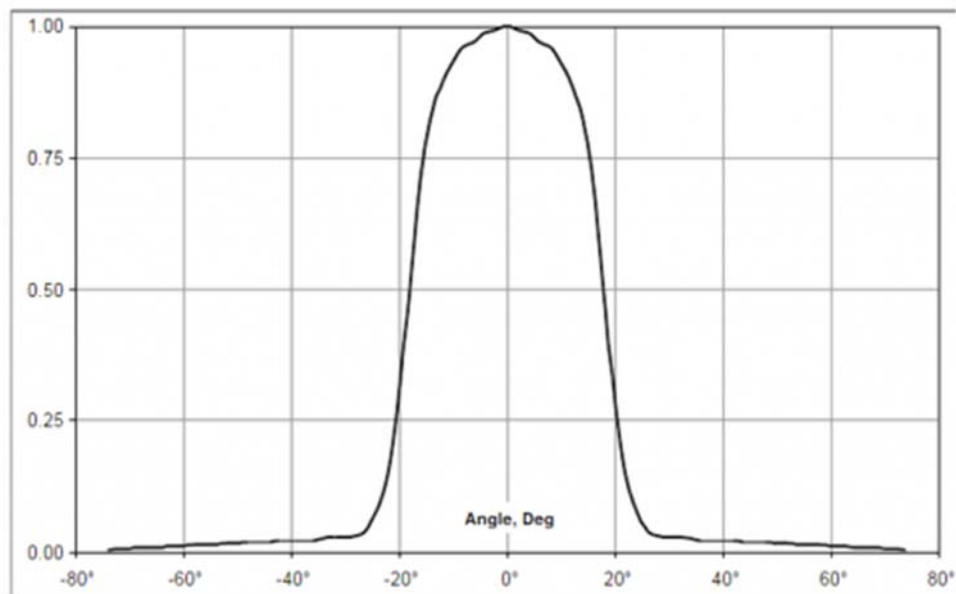
delay(500);

}
```

## Measurement Method

Before using the infrared temperature measurement module it is important to understand the concept of "field of view" (FOV). FOV is determined when a thermopile receives 50% of the radiation signal, and also related to the main axis of the sensor. As shown in the figure below, the size of the FOV is indicated on the horizontal axis. This measured temperature is actually the weighted average temperature of the object in the FOV and the measurement accuracy can only be ensured when the testing object totally covers the FOV of the infrared sensor. This means that the distance between the measurement terminal point and the bus bar must be ensured to meet the demands to guarantee the temperature measurement accuracy.

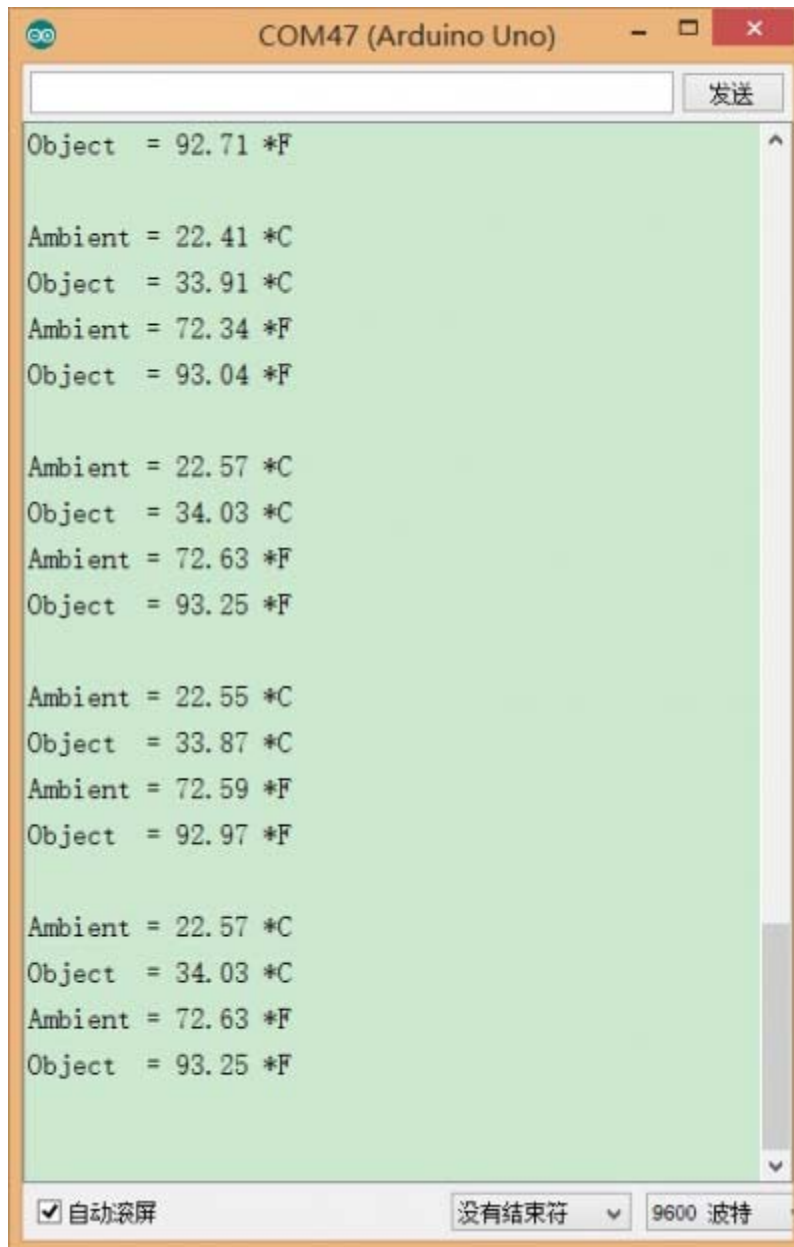
In greater detail, this module has a FOV of  $35^\circ$  so  $\tan 35^\circ = \frac{\text{radius of the testing object}}{\text{distance between the infrared sensor and testing object}}$ . e.g.: the radius of the testing object is 5cm, so the measuring distance is 7cm (that means the measurement result is most accurate within this scope). The FOV graph of this sensor is below:



The points of measurement all need to be in the field

# Result

IR Thermometer Sensor-MLX90614 Test Result



```
COM47 (Arduino Uno) 发送
Object = 92.71 *F
Ambient = 22.41 *C
Object = 33.91 *C
Ambient = 72.34 *F
Object = 93.04 *F
Ambient = 22.57 *C
Object = 34.03 *C
Ambient = 72.63 *F
Object = 93.25 *F
Ambient = 22.55 *C
Object = 33.87 *C
Ambient = 72.59 *F
Object = 92.97 *F
Ambient = 22.57 *C
Object = 34.03 *C
Ambient = 72.63 *F
Object = 93.25 *F
```

自动滚屏    没有结束符    9600 波特

IR Thermometer Sensor-MLX90614 Test Result

[https://www.dfrobot.com/wiki/index.php/IR\\_Thermometer\\_Sensor\\_MLX90614\\_SKU:\\_SEN0206](https://www.dfrobot.com/wiki/index.php/IR_Thermometer_Sensor_MLX90614_SKU:_SEN0206) 7-16-18