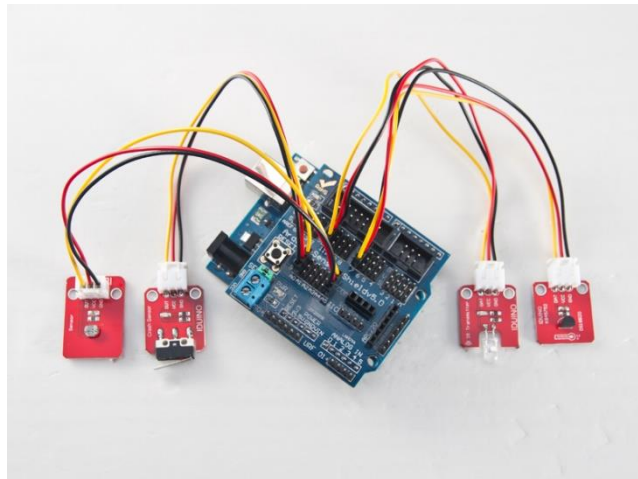


DS18B20 Temperature Sensor(SE029)



1. Introduction

This module is temperature sensor with chip DS18B20, It's different from other NTC-MF523950 temperature sensor(ST1147) or LM35 temperature sensor(SE039). This module has one indicator light, which would be on when this module's voltage signal is changed. And, this module has integrated 3-pin terminal, which can be simply and tidily connected with Arduino sensor expansion board, like the following picture:



The Module's feature as below:

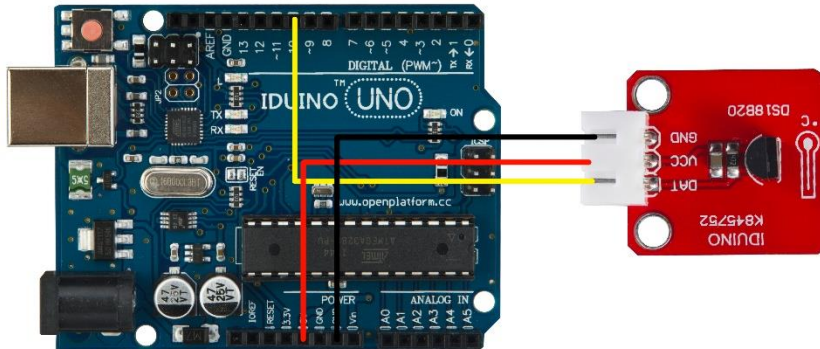
Feature	Value
Chip	DS18B20
Temperature Range	-55°C~+125°C
Accuracy	+/- 0.5°C
Supply voltage	5V DC

2.Pinout

Pin	Description
DAT	Signal pin
VCC	Power(reference voltage:5V DC)
GND	Ground

3 Example

This is a simple code for the DS18B20 temperature module, Wire as below:



Example code :

```
*****Code begin*****
```

```
#include <OneWire.h>
```

```
// DS18S20 Temperature chip i/o  
OneWire ds(10); // on pin 10
```

```
void setup(void) {  
  // initialize inputs/outputs  
  // start serial port  
  Serial.begin(9600);  
}
```

```
void loop(void) {
```

```
  //For conversion of raw data to C  
  int HighByte, LowByte, TReading, SignBit, Tc_100, Whole, Fract;
```

```
  byte i;  
  byte present = 0;  
  byte data[12];
```

IDUINO for Maker's life

```
byte addr[8];

if ( !ds.search(addr)) {
    Serial.print("No more addresses.\n");
    ds.reset_search();
    return;
}

Serial.print("R=");
for( i = 0; i < 8; i++) {
    Serial.print(addr[i], HEX);
    Serial.print(" ");
}

if ( OneWire::crc8( addr, 7) != addr[7]) {
    Serial.print("CRC is not valid!\n");
    return;
}

if ( addr[0] == 0x10) {
    Serial.print("Device is a DS18S20 family device.\n");
}
else if ( addr[0] == 0x28) {
    Serial.print("Device is a DS18B20 family device.\n");
}
else {
    Serial.print("Device family is not recognized: 0x");
    Serial.println(addr[0],HEX);
    return;
}

ds.reset();
ds.select(addr);
ds.write(0x44,1);          // start conversion, with parasite power on
at the end

delay(1000);             // maybe 750ms is enough, maybe not
// we might do a ds.depower() here, but the reset will take care of it.

present = ds.reset();
ds.select(addr);
ds.write(0xBE);          // Read Scratchpad

Serial.print("P=");
```

IDUINO for Maker's life

```
Serial.print(present,HEX);
Serial.print(" ");
for ( i = 0; i < 9; i++) {           // we need 9 bytes
  data[i] = ds.read();
  Serial.print(data[i], HEX);
  Serial.print(" ");
}
Serial.print(" CRC=");
Serial.print( OneWire::crc8( data, 8), HEX);
Serial.println();

//Conversion of raw data to C
LowByte = data[0];
HighByte = data[1];
TReading = (HighByte << 8) + LowByte;
SignBit = TReading & 0x8000; // test most sig bit
if (SignBit) // negative
{
  TReading = (TReading ^ 0xffff) + 1; // 2's comp
}
Tc_100 = (6 * TReading) + TReading / 4;    // multiply by (100 * 0.0625)
or 6.25

Whole = Tc_100 / 100; // separate off the whole and fractional portions
Fract = Tc_100 % 100;

if (SignBit) // If its negative
{
  Serial.print("-");
}
Serial.print(Whole);
Serial.print(".");
if (Fract < 10)
{
  Serial.print("0");
}
Serial.print(Fract);

Serial.print("\n");
//End conversion to C
}*****Code End*****
```