## DS18B20

#### Overview：

这个实验将使用一个DS18B20传感器检测温度并显示出来。

#### **Materials：**

Arduino Uno x 1

DS18B20 sensor x 1

DuPont wires x 3

#### **Product description :**

#### DS18B20是常用的数字温度传感器，具有体积小，硬件开销低，抗干扰能力强，精度高的特点。DS18B20数字温度传感器接线方便，封装成后可应用于多种场合，如管道式，螺纹式，磁铁吸附式，不锈钢封装式，型号多种多样，有LTM8877，LTM8874等等。主要根据应用场合的不同而改变其外观。封装后的DS18B20可用于电缆沟测温，高炉水循环测温，锅炉测温，机房测温，农业大棚测温，洁净室测温，弹药库测温等各种非极限温度场合。耐磨耐碰，体积小，使用方便，封装形式多样，适用于各种狭小空间设备数字测温和控制领域。**Technical Parameters ：**

DS18B20 单线数字温度传感器，即“一线器件”，其具有独特的优点：

　　（ 1 ）采用单总线的接口方式。 与微处理器连接时仅需要一条口线即可实现微处理器与 DS18B20 的双向通讯。

　　（ 2 ）测量温度范围宽，测量精度高。 DS18B20 的测量范围为 -55 ℃ ~+ 125 ℃ ; 在 -10~+ 85°C范围内，精度为 ± 0.5°C 。

　　（ 3 ）在使用中不需要任何外围元件。

　　（ 4 ）持多点组网功能。 多个 DS18B20 可以并联在惟一的单线上，实现多点测温。

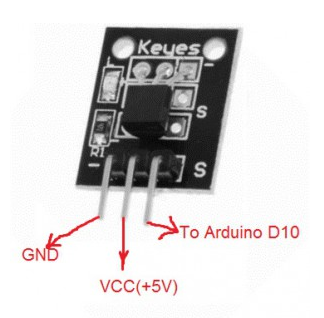
　　（ 5 ）供电方式灵活。 DS18B20 可以通过内部寄生电路从数据线上获取电源。因此，当数据线上的时序满足一定的要求时，可以不接外部电源，从而使系统结构更趋简单，可靠性更高。

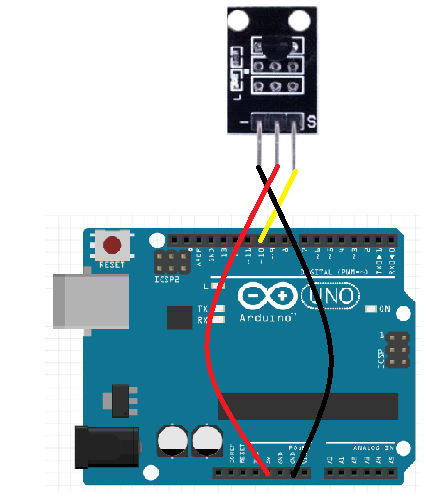
　　（ 6 ）测量参数可配置。 DS18B20 的测量分辨率可通过程序设定 9~12 位。

　　（ 7 ） 负压特性电源极性接反时，温度计不会因发热而烧毁，但不能正常工作。

　　（ 8 ）掉电保护功能。 DS18B20 内部含有 EEPROM ，在系统掉电以后，它仍可保存分辨率及报警温度的设定值。

#### **Wiring diagram:**





**Example code:**

|  |
| --- |
| **/\* DS18S20 Temperature Sensor Module sample project**  **\*/**  **#include <OneWire.h>**  **int inPin=10; // define D10 as input pin connecting to DS18S20 S pin**  **OneWire ds(inPin);**    **void setup(void) {**  **Serial.begin(9600);**  **}**    **void loop(void) {**  **int HighByte, LowByte, TReading, SignBit, Tc\_100, Whole, Fract;**  **byte i;**  **byte present = 0;**  **byte data[12];**  **byte addr[8];**    **if ( !ds.search(addr)) {**  **ds.reset\_search();**  **return;**  **}**      **ds.reset();**  **ds.select(addr);**  **ds.write(0x44,1);**    **delay(1000);**      **present = ds.reset();**  **ds.select(addr);**  **ds.write(0xBE);**        **for ( i = 0; i < 9; i++) {**  **data[i] = ds.read();**  **}**  **Serial.print("Temperature: ");**  **LowByte = data[0];**  **HighByte = data[1];**  **TReading = (HighByte << 8) + LowByte;**  **SignBit = TReading & 0x8000;**  **if (SignBit)**  **{**  **TReading = (TReading ^ 0xffff) + 1;**  **}**  **Tc\_100 = (6 \* TReading) + TReading / 4;**  **Whole = Tc\_100 / 100;**  **Fract = Tc\_100 % 100;**  **if (SignBit)**  **{**  **Serial.print("-");**  **}**  **Serial.print(Whole);**  **Serial.print(".");**  **if (Fract < 10)**  **{**  **Serial.print("0");**  **}**  **Serial.print(Fract);**  **Serial.print(" C\n");**  **}** |

**Experimental phenomena：**

you will see temperature results in the monitor window.

