Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of "Quality Parts,Customers Priority,Honest Operation,and Considerate Service",our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Terminal sensor adapter V2 SKU: DFR0055

Contents

Introduction

This is universal sensor adapter V2.0, it can be accessed by digital input modules and analog input module. The module with Arduino sensor expansion board, in combination.Improvment compare with the old version,we add a 10K resistor between A & B, A & C respectively,so that you needn't to connect a addition resistor while the sensor must have a pull-up or pull-down resistor. Why need a pull-up or pull down resistor, please refer to"Uderstanding pull-up and pull-down resistors"
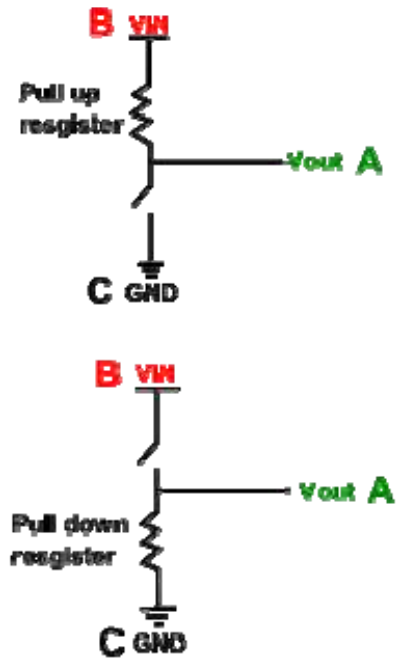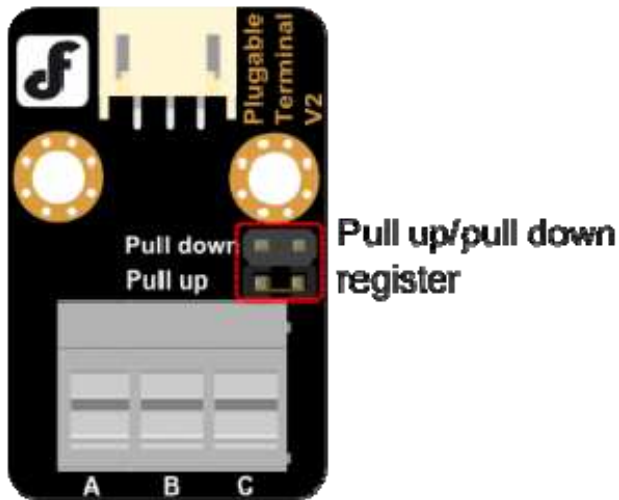
Generally speaking, this terminal provides a much easier way to connect a switch or any modules that requires pull-up/down resistor to microcontroller

Specifications

- Voltage: +5 V
- Current: <1000mA
- Size: 22x34mm

Pin Out

Pull up/pull down register

PIN	CONNECTION
A	Data Signal
B	VIN
C	GND

Application with DS18B20

When using temperature Waterproof DS18B20 Digital temperature sensor,need to connect a pull-up resistor for it drived with Arduino microcontroller. The terminal sensor adapter will supply a simple ad stable connection in case without soldering.

Fig2: Application CD
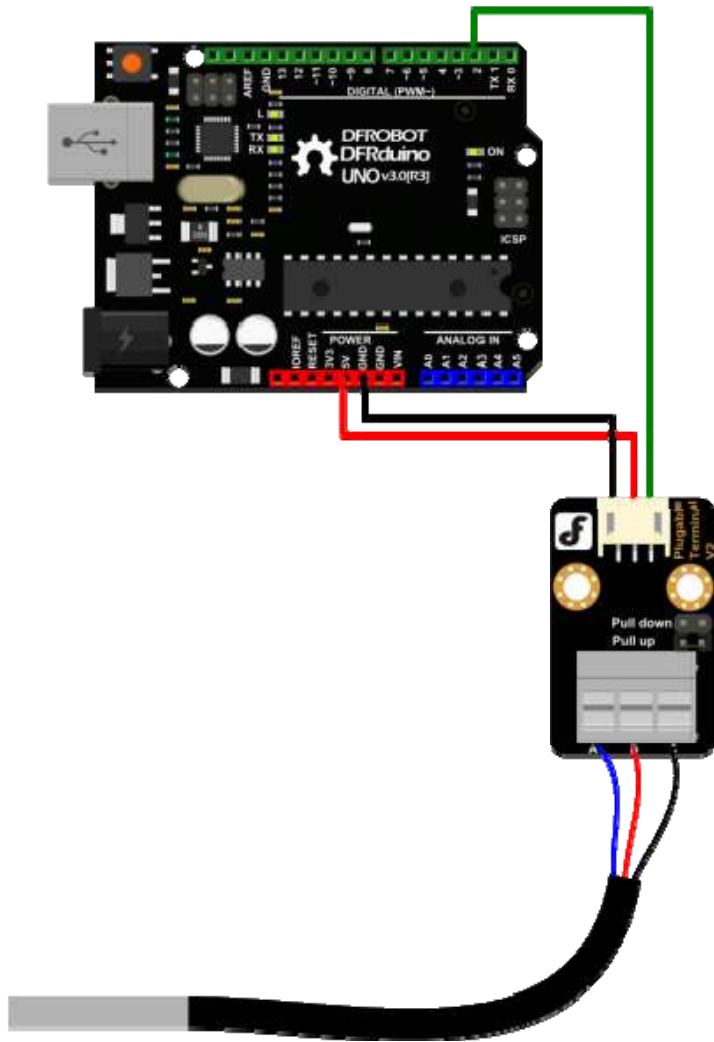
Sample Code

```
/****************************************************************/

 Insturciton:

 Connection:

 1.Plugable Terminal Sensor Adapter & Waterproof DS18B20 Digital Temperature
Sensor

               A    ---->     Blue(DATA SIGNAL)

               B    ---->     RED   (VIN)

               C    ---->     Black (GND)
```

```
  2.Waterproof DS18B20 Digital Temperature Sensor & Arduino board
              1(A)    ---->      Digital Pin2
              2(B)    ---->      5V/3.3V
              3(C)    ---->      GND


 Setting for the Pull-up Register/Pull-down Register Selection Jumpers
      When connect DS18B20 with the adapter,please choose to use the
      Pull-up Register Jumper
 *************************************************************/
#include <OneWire.h>


int DS18S20_Pin = 2; //DS18S20 Signal pin on digital 2


//Temperature chip i/o
OneWire ds(DS18S20_Pin);  // on digital pin 2


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


void loop(void) {
  float temperature = getTemp();
  Serial.println(temperature);


  delay(2000); //just here to slow down the output so it is easier to read


}



float getTemp(){
  //returns the temperature from one DS18S20 in DEG Celsius


  byte data[12];
  byte addr[8];
```

```
if ( !ds.search(addr)) {

    //no more sensors on chain, reset search

    Serial.println("no more sensors on chain, reset search!");

    ds.reset_search();

    return -1000;

}


if ( OneWire::crc8( addr, 7) != addr[7]) {

    Serial.println("CRC is not valid!");

    return -1000;

}


if ( addr[0] != 0x10 && addr[0] != 0x28) {

    Serial.print("Device is not recognized");

    return -1000;

}


ds.reset();

ds.select(addr);

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


byte present = ds.reset();

ds.select(addr);

ds.write(0xBE); // Read Scratchpad



for (int i = 0; i < 9; i++) { // we need 9 bytes

  data[i] = ds.read();

}


ds.reset_search();


byte MSB = data[1];
```

```
    byte LSB = data[0];


    float tempRead = ((MSB << 8) | LSB); //using two's compliment
    float TemperatureSum = tempRead / 16;


    return TemperatureSum;


}
```

Version history

[Terminal sensor adapter V1.1](#)