DHT22 Temperature and humidity module SKU:SEN0137

DHT22 Temperature and humidity module SKU:SEN0137

Introduction

DHT22 capacitive humidity sensing digital temperature and humidity module is one that contains the compound has been calibrated digital signal output of the temperature and humidity sensors. Application of a dedicated digital modules collection technology and the temperature and humidity sensing technology, to ensure that the product has high reliability and excellent long-term stability.

The sensor includes a capacitive sensor wet components and a high-precision temperature measurement devices, and connected with a high-performance 8-bit microcontroller. The product has excellent quality, fast response, strong anti-jamming capability, and high cost.

Standard single-bus interface, system integration quick and easy. Small size, low power consumption, signal transmission distance up to 20 meters, making it the best choice of all kinds of applications and even the most demanding applications.

DHT22 has higher precision and can replace the expensive imported SHT10 temperature and humidity sensor.

It can measure the environment temperature and humidity to meet the high demand.The product has high reliability and good stability.If it's used and combined with special sensor Arduino expansion board,it will be easily implemented the interactive effect which related to the temperature and humidity perception.

Caution: DHT22 digital temperature and humidity sensor is designed for analog sensor interfaces.The analog port will be used as the digital which will not occupy the original digital port of the Arduino.The lines of the sensor which can transform the analog function to digital that can be use on digital port.

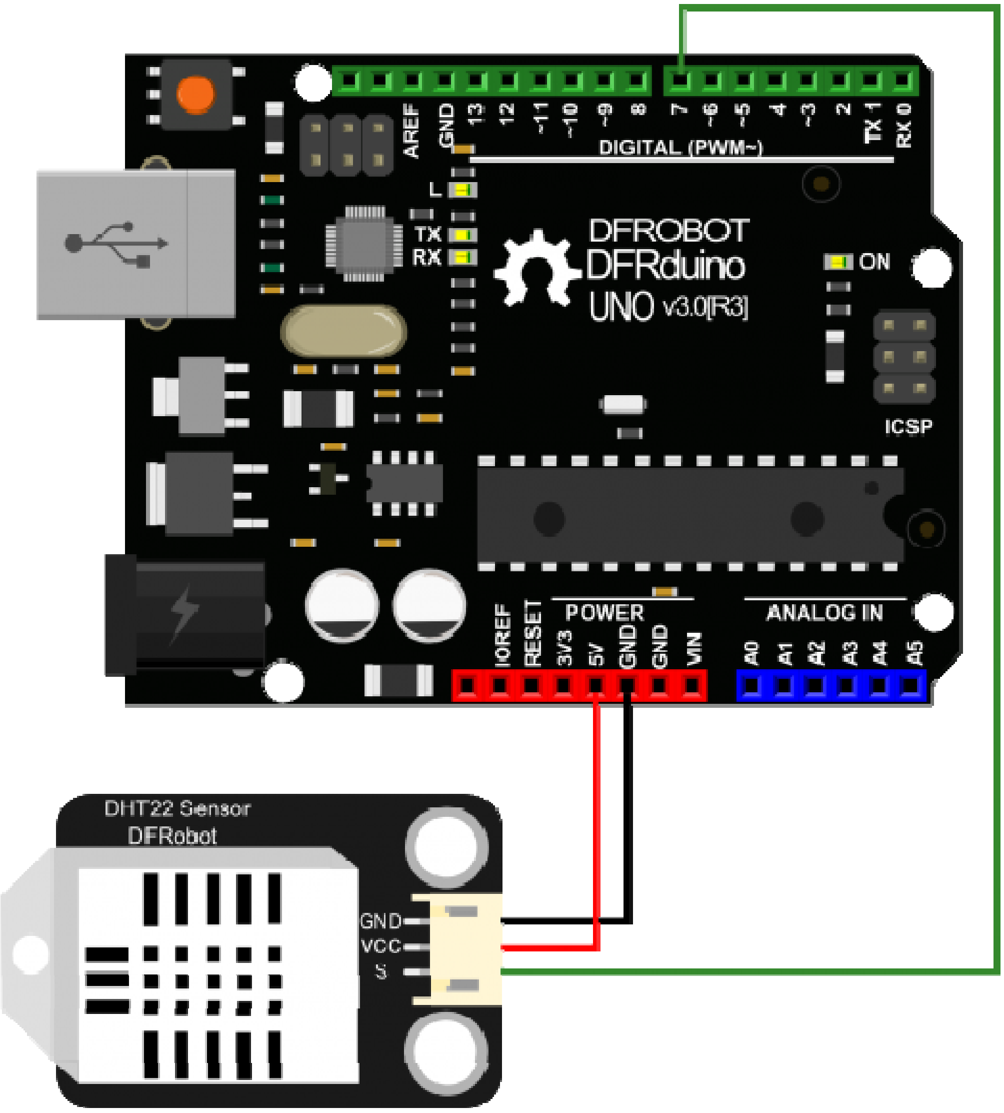
```
Note:The line of the DHT22 sensor module is analog--digital
```

Specifications

- Supply voltage: 5V
- Output voltage: 0-3.3V
- Temperature range:-40-80℃ resolution0.1℃ error <±0.5℃
- Humidity range:0-100%RH resolution0.1%RH error±2%RH
- size: 38 x 20mm

Tutorial

Connection



Sample code

Please download the Library of **DHT22** at first.This program is used to test the temperature and humidity of the DHT22

```
//DHT11 -- DIGITAL 7
#include <DHT22.h>
// Only used for sprintf
#include <stdio.h>

// Data wire is plugged into port 7 on the Arduino

#define DHT22_PIN 7

// Setup a DHT22 instance
DHT22 myDHT22(DHT22_PIN);

void setup(void)
{
  //start serial port
  Serial.begin(9600);
  Serial.println("DHT22 Library Demo");
}

void loop(void)
{
  DHT22_ERROR_t errorCode;
  // The sensor can only be read from every 1-2s, and requires a minimum
  // 2s warm-up after power-on.
  delay(2000);

  Serial.print("Requesting data...");
  errorCode = myDHT22.readData();
  switch(errorCode)
  {
     case DHT_ERROR_NONE:
      Serial.print("Got Data ");
      Serial.print(myDHT22.getTemperatureC());
      Serial.print("C ");
      Serial.print(myDHT22.getHumidity());
      Serial.println("%");
      /*Alternately, with integer formatting which is clumsier but more compact to store and*/
      /*can be compared reliably for equality:*/
      char buf[128];
      sprintf(buf, "Integer-only reading: Temperature %hi.%01hi C, Humidity %i.%01i %% RH",
                   myDHT22.getTemperatureCInt()/10, abs(myDHT22.getTemperatureCInt()%10),
                   myDHT22.getHumidityInt()/10, myDHT22.getHumidityInt()%10);
      Serial.println(buf);
      break;
    case DHT_ERROR_CHECKSUM:
      Serial.print("check sum error ");
      Serial.print(myDHT22.getTemperatureC());
      Serial.print("C ");
      Serial.print(myDHT22.getHumidity());
      Serial.println("%");
      break;
    case DHT_BUS_HUNG:
      Serial.println("BUS Hung ");
      break;
    case DHT_ERROR_NOT_PRESENT:
      Serial.println("Not Present ");
      break;
    case DHT_ERROR_ACK_TOO_LONG:
      Serial.println("ACK time out ");
      break;
    case DHT_ERROR_SYNC_TIMEOUT:
      Serial.println("Sync Timeout ");
      break;
    case DHT_ERROR_DATA_TIMEOUT:
      Serial.println("Data Timeout ");
      break;
    case DHT_ERROR_TOOQUICK:
      Serial.println("Polled to quick ");
      break;
   }
  }
```