Lesson 14 DHT11 Temperature and Humidity Sensor

Introduction

In this lesson, you will learn how to use a DHT11 Temperature and Humidity Sensor.

Hardware Required

- ✓ 1 * RexQualis UNO R3
- ✓ 1 * DHT11 Temperature and Humidity module
- ✓ 3 * F-M Jumper Wires

Principle

DHT11 Temperature and Humidity Sensor



DHT11 output calibrated digital signal. It applies exclusive digital-signal-colle cting-technique and humidity sensing technology, assuring its reliability and st ability. Its sensing elements are connected with an 8-bit single-chip computer. Every sensor of this model is temperature compensated and calibrated in an accurate calibration chamber and the calibration-coefficient is saved in the

type of program in OTP memory when the sensor is detecting, it will cite the coefficient from memory.

Small size & low consumption & long transmission distance(100m) enable DH

T11 to be suited in all kinds of

harsh application occasions. Single-row packaged with four pins, making the c onnection very convenient.

Supply voltage: DC 3.3 to 5.5V

Measuring range (T) : -20 to +60 Celsius(-4 to +140 Fahrenheit)

Measuring range (RH): 5 to 95% relative humidity

Typ. Temperature accuracy: ± 2 Celsius

Typ. Humidity accuracy: \pm 5%RH at 25 Celsius

Long term drift(T): <1 Celsius/year

Long term drift(RH) : <1%RH/year

Resolution(T): 0.1 Celsius

Resolution(RH): 1%RH

Sensor Type: Capacitive sensor

Interface: One line digital

Housing material: ABS

Net weight: 1g

Code interpretation

#define DHT11_PIN 0 // pin A0

byte read_dht11_dat()

{

byte i = 0;

```
byte result=0;
```

```
for(i=0; i< 8; i++){
```

while(!(PINC & _BV(DHT11_PIN))); // wait for 50us

```
delayMicroseconds(30);
```

```
if(PINC & _BV(DHT11_PIN))
```

```
result |=(1<<(7-i));
```

```
while((PINC & _BV(DHT11_PIN))); // wait '1' finish
```

}

```
return result;
```

```
}
```

```
void setup()
```

```
{
```

```
DDRC |= _BV(DHT11_PIN);
```

```
PORTC |= _BV(DHT11_PIN);
```

```
Serial.begin(19200);
```

```
Serial.println("Ready");
```

```
}
```

```
void loop()
```

```
{
```

```
byte dht11_dat[5];
```

```
byte dht11_in;
```

byte i;

// start condition

// 1. pull-down i/o pin from 18ms

PORTC &= ~_BV(DHT11_PIN);

delay(18);

PORTC |= _BV(DHT11_PIN);

delayMicroseconds(40);

DDRC &= ~_BV(DHT11_PIN);

```
delayMicroseconds(40);
```

dht11_in= PINC & _BV(DHT11_PIN);

if(dht11_in){

Serial.println("dht11 start condition 1 not met");

return;

```
}
```

delayMicroseconds(80);

dht11_in = PINC & _BV(DHT11_PIN);

if(!dht11_in){

Serial.println("dht11 start condition 2 not met");

return;

}

delayMicroseconds(80);

// now ready for data reception

```
for (i=0; i<5; i++)
```

```
dht11_dat[i] = read_dht11_dat();
```

DDRC |= _BV(DHT11_PIN);

PORTC |= _BV(DHT11_PIN);

```
byte dht11_check_sum =
```

```
dht11_dat[0]+dht11_dat[1]+dht11_dat[2]+dht11_dat[3];
```

// check check_sum

```
if(dht11_dat[4]!= dht11_check_sum)
{
   Serial.println("DHT11 checksum error");
```

```
}
```

```
Serial.print("Current humdity = ");
```

```
Serial.print(dht11_dat[0], DEC);
```

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

```
Serial.print(dht11_dat[1], DEC);
```

Serial.print("% ");

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

```
Serial.print(dht11_dat[2], DEC);
```

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

```
Serial.print(dht11_dat[3], DEC);
```

Serial.println("C ");

delay(2000);

Experimental Procedures

Step 1:Build the circuit



Schematic Diagram



Step 2: Open the code:

DHT11_Temperature_and_Humidity_Sensor_Code

| Blinking_LED_C File Edit Sketch | ode Arduino 1.8.5 Tools Help | | | |
|--|--|----------------------|------------|--|
| | | ø | | |
| 🙆 Open an Arduino sketch | | | | |
| 查找范围(I): |] DHT11_Temperature_and_Humidity_Senso 👻 | 3 🗊 😕 🛄 - | | |
| Ca. | 名称 | 修改日期 | 0 | |
| 最近访问的位置 | DHT11_Temperature_and_Humidity_Sense | or_Code.ino 2017/1 | 2/20 21:39 | |
| 桌面 桌面 库 计算机 | | | | |
| | ٠ (ا | | P. | |
| 网络 | 对象名称 (N): | - | 打开 (0) | |
| | 对象类型(T): 【All Files (*.*) | • | 取消 | |
| | | | | |
| viobal variables use 9 bytes (D.%) of dynamic memory, leaving 2059 bytes for 1 | | | | |
| III Arduine/Genuine Une on COM118 | | | | |
| | AldunoiSerun | | | |

Step 3: Attach Arduino UNO R3 board to your computer via USB cable and check that the 'Board Type' and 'Serial Port' are set correctly.

Step 4: Upload the code to the RexQualis UNO R3 board.

Step 5: Open the Serial Monitor, alter the baud rate to 19200, then you can see the data as below:

(How to use the Serial Monitor is introduced in details in Lesson 0 Preface)

| 💿 COM121 (Arduino/Genuino Uno) | |
|---|----------------------------|
| | Send |
| Ready | * |
| Current humdity = 22.0% temperature = 24.4C | |
| Current humdity = 22.0% temperature = 24.4C | |
| Current humdity = 22.0% temperature = 24.5C | |
| Current humdity = 23.0% temperature = 24.5C | |
| Current humdity = 29.0% temperature = 24.5C | |
| Current humdity = 38.0% temperature = 24.6C | = |
| Current humdity = 48.0% temperature = 24.7C | |
| Current humdity = 60.0% temperature = 24.9C | |
| Current humdity = 70.0% temperature = 25.10 | |
| Current humdity = 80.0% temperature = 25.3C | |
| Current humdity = 87.0% temperature = 25.7C | |
| Current humdity = 92.0% temperature = 26.0C | |
| Current humdity = 95.0% temperature = 26.0C | |
| Current humdity = 87.0% temperature = 26.1C | |
| Current humdity = 75.0% temperature = 26.0C | |
| Current humdity = 69.0% temperature = 26.0C | |
| Current humdity = 62.0% temperature = 26.00 | - |
| Autoscroll No line ending - | [19200 baud 🔹 Clear output |



If it isn' t working, make sure you have assembled the circuit correctly, verified and uploaded the code to your board. For how to upload the code and install the library, check Lesson 0 Preface.