

Teil 2

Lektion

10

**DHT11 Temperatur-
und Luftfeuchtigkeitssensor**

Übersicht

In dieser Lektion lernen Sie, wie man einen DHT11 Sensor zum Messen der Temperatur und der Luftfeuchtigkeit benutzt.

Dieser Sensor ist präzise genug für die meisten Projekte, bei denen auf die Temperatur und die Luftfeuchtigkeit geachtet werden muss.

Wir werden wieder eine Bibliothek verwenden, die speziell für diesen Sensor angepasst wurde, sodass unser eigener Code kurz und knapp ausfällt.

Benötigte Bauteile:

- (1) x Elegoo Uno R3
- (1) x DHT11 Temperatur- und Luftfeuchtigkeitssensor
- (4) x W-M Kabel (Weiblich zu Männlich DuPont Jumper Kabel)

Einführung in die Komponenten

Temperatur- und Luftfeuchtigkeitssensor:

Der digitale DHT11 Temperatur- und Luftfeuchtigkeitssensor ist ein Verbundsensor, der ein digitales Signal mit den Werten für Temperatur und Luftfeuchtigkeit ausgibt. Die eingebauten digitalen Module sorgen dafür, dass der Sensor zuverlässige Werte ausgibt und sorgen außerdem für lange Haltbarkeit. Der Sensor enthält ein NTC Temperaturmessgerät und ist an einem 8-bit -Mikrocontrollerangeschlossen.

Anwendungsmöglichkeiten: Heizung, Lüftung, Klimatechnik; Luftentfeuchter; Prüf- und Inspektionsausrüstung; Automobilbau, Wetterstationen, Haushaltsgeräte, Feuchtigkeitsregulatoren und medizinische Geräte.

Produktspezifikationen

Relative Feuchtigkeit:

Auflösung: 16Bit Wiederholbarkeit: $\pm 1\%$ RH

Genauigkeit: Bei 25°C $\pm 5\%$ RH

Austauschbarkeit: vollkommen austauschbar Antwortzeit: 1 / e (63%) von 25°C 6s

1m / s Luft 6s

Hysterese: $< \pm 0.3\%$ RH Langzeitstabilität: $< \pm 0.5\%$ RH / Jahr in

Temperatur

Auflösung: 16Bit Wiederholbarkeit:
 $\pm 0.2^\circ\text{C}$ Temperaturbereich: Bei 25°C
 $\pm 2^\circ\text{C}$

Antwortzeit: 1 / e (63%) 10S

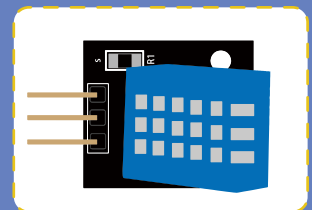
Betriebsspannung: DC 3,5~5,5V

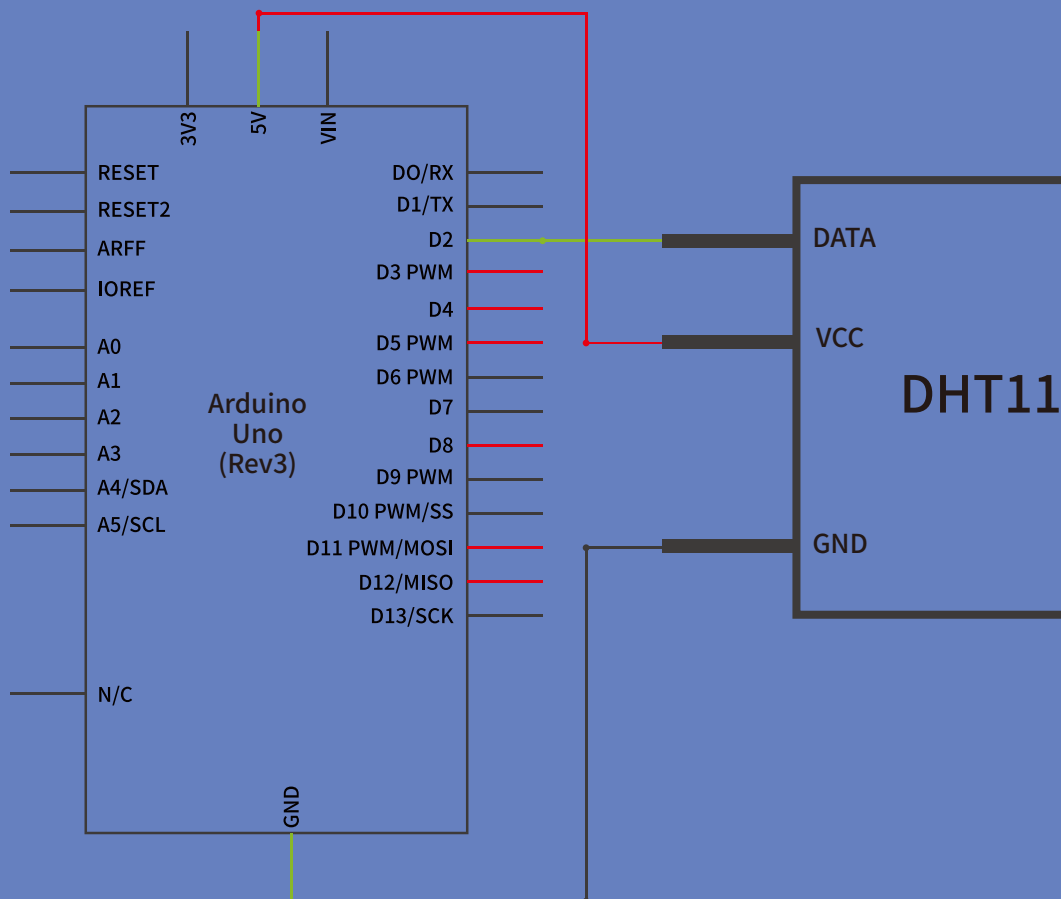
Betriebsstrom: 0.3mA, standby 60µA (gemessen)

Abtastabstand: mehr als 2 Sekunden

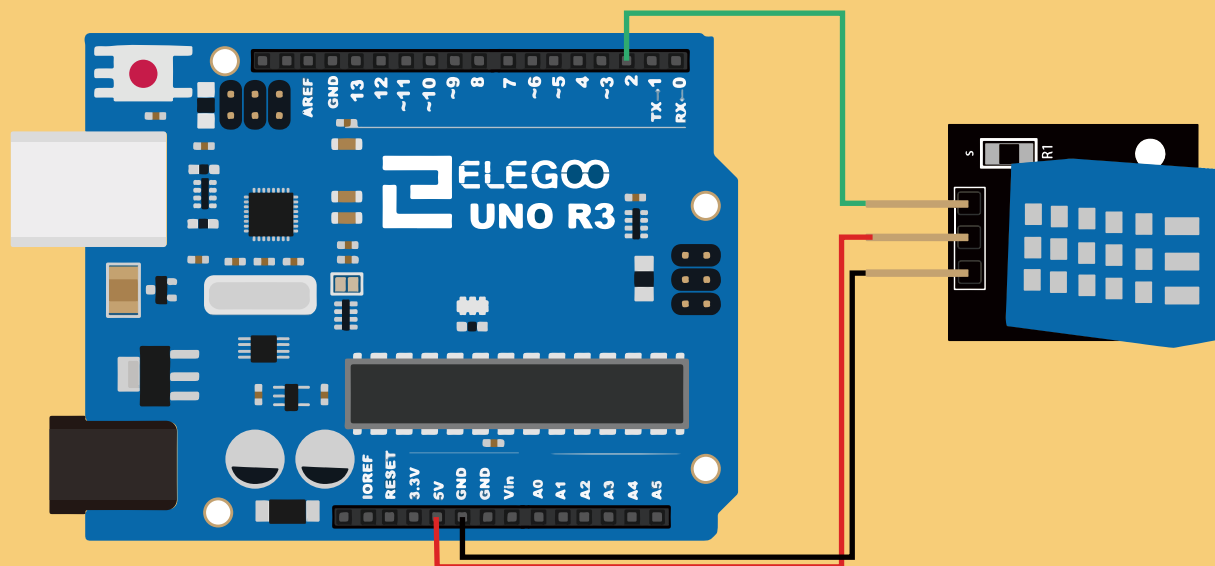
Pin Description:

1. VDD Stromanschluss 3,5~5,5V DC
2. DATA Serieller Datenanschluss
3. GND, negativer Anschluss





Verbindungsschema



Die Anschlüsse sind: Signal-Nr.65292Spannung und Boden, die an jeden Pin auf unserer UNO angeschlossen werden können.

Schaltplan

Code

- Nach dem Verbinden der Komponenten öffnen Sie bitte den Sketch im Code-Ordner unter "DHT11_Example" und la
- Bei Fragen zum Hochladen eines Sketches schauen Sie sich bitte Lektion 5 in Teil 1 noch einmal an.
- Bevor Sie diesen Sketch hochladen können, müssen Sie die „SimpleDHT“-Bibliothek installiert haben. Sonst wird s

static

[Variable Scope & Qualifiers]

Description

Das "static" Schlüsselwort wird verwendet, um Variablen zu erstellen, die nur für eine Funktion sichtbar sind. Im Gegensatz zu lokalen Variablen, welche bei jedem Aufruf einer Funktion erstellt werden, bleiben statische Variablen über den Funktionsaufruf hinaus bestehen und behalten ihre Daten zwischen Funktionsaufrufen bei.

Als "static" deklarierte Variablen werden nur beim ersten Aufruf einer Funktion erstellt und initialisiert.

```
static const int DHT_SENSOR_PIN = 2;
```

float

[Data Types]

Description

"float" ist der Datentyp für Gleitkommazahlen: eine Zahl mit Dezimalpunkt. Gleitkommazahlen werden häufig verwendet, um analoge und kontinuierliche Werte zu approximieren, da sie eine höhere Auflösung als Ganzzahlen haben. Gleitkommazahlen können so groß wie 3,4028235E + 38 und so niedrig wie -3,4028235E + 38 sein. Sie werden als 32 Bit (4 Byte) Informationen gespeichert.

```
float temperature;
```

```
float humidity;
```

Syntax

```
float var = val;
```

Parameters

var: variabler name.

val: Wert, den Sie dieser Variable zuweisen.

Bool

[Data Types]

Beschreibung

Ein Bool enthält einen von zwei Werten: true oder false. Jede Bool-Variable belegt ein Byte Speicher.

Syntax

```
bool var = val;
```

Parameters

var: variabler name.

val: Wert, den Sie dieser Variable zuweisen.

unsigned long

[Conversion]

Description

Konvertiert einen Wert in den "unsigned long" Datentyp.

Parameters

x: a value of any type

Returns

unsigned long

```
static bool measure_environment( float *temperature, float *humidity )
{
    static unsigned long measurement_timestamp = millis( );

    /* Measure once every four seconds. */
    if( millis( ) - measurement_timestamp > 3000ul )
    {
        if( dht_sensor.measure( temperature, humidity ) == true )
        {
            measurement_timestamp = millis( );
            return( true );
        }
    }

    return( false );
}
```

- Upload the program then open the monitor, we can see the data as below:
(It shows the temperature of the environment, we can see it is 27 to 25 degrees with 45.0% humidity)
- Click the Serial Monitor button to turn on the serial monitor. The basics about the serial monitor are introduced in details in part 2 Lesson 4.

