# Lesson 11 DHT11 Temperature and Humidity Sensor

## Overview

In this tutorial we will learn how to use a DHT11 Temperature and Humidity Sensor. It’s accurate enough for most projects that need to keep track of humidity and temperature readings.

Again we will be using a Library specifically designed for these sensors that will make our code short and easy to write.

## Component Required:

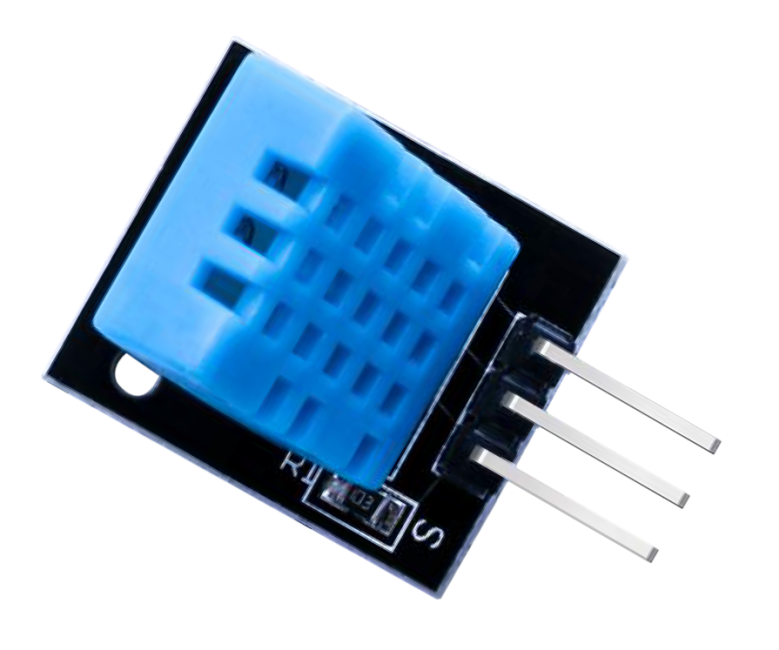
(1) x Elegoo Uno R3

(1) x DHT11 Temperature and Humidity module

(3) x F-M wires (Female to Male DuPont wires)

## Component Introduction

### Temp and humidity sensor:



DHT11 digital temperature and humidity sensor is a composite Sensor which contains a calibrated digital signal output of the temperature and humidity. The dedicated digital modules collection technology and the temperature and humidity sensing technology are applied to ensure that the product has high reliability and

excellent long-term stability. The sensor includes a resistive sense of wet components and a NTC temperature measurement devices, and connects with a high-performance 8-bit microcontroller.

Applications: HVAC, dehumidifier, testing and inspection equipment, consumer goods, automotive, automatic control, data loggers, weather stations, home appliances, humidity regulator, medical and other humidity measurement and control.

Product parameters Relative humidity: Resolution: 16Bit Repeatability: ±1% RH

Accuracy: At 25℃ ±5% RH

Interchangeability: fully interchangeable Response time: 1 / e (63%) of 25℃ 6s

1m / s air 6s Hysteresis: <± 0.3% RH

Long-term stability: <± 0.5% RH / yr in Temperature:

Resolution: 16Bit Repeatability: ±0.2℃ Range: At 25℃ ±2℃

Response time: 1 / e (63%) 10S Electrical Characteristics Power supply: DC 3.5～5.5V

Supply Current: measurement 0.3mA standby 60μA

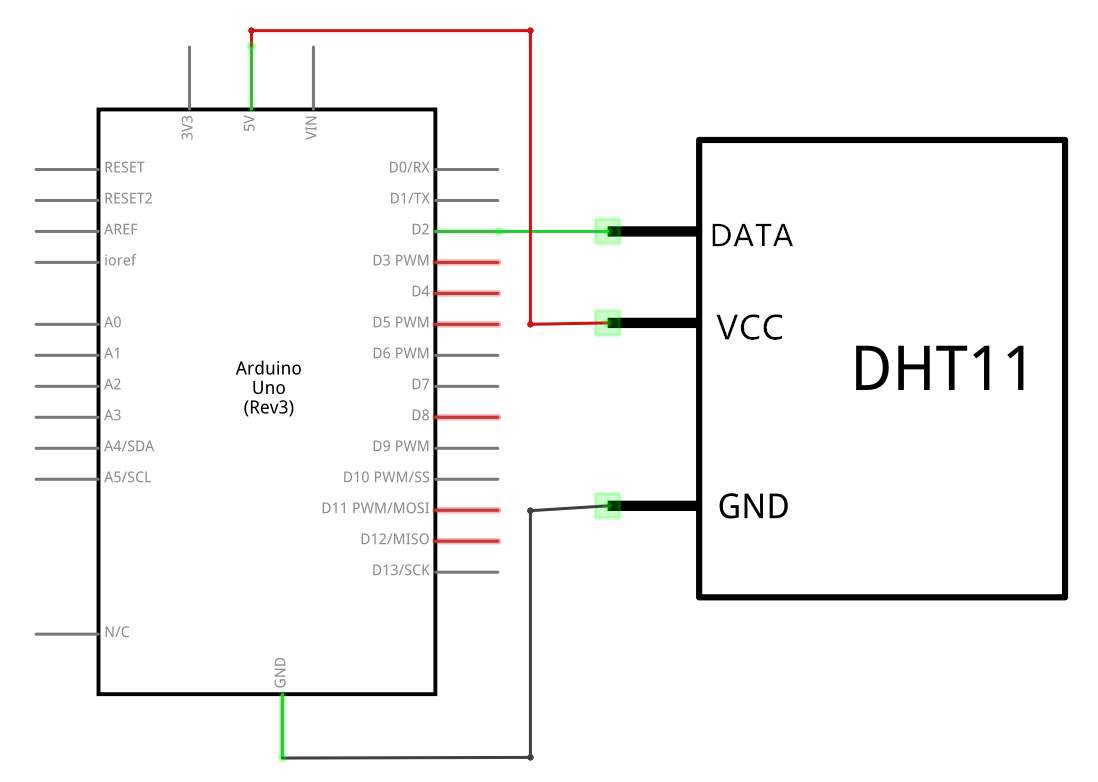
Sampling period: more than 2 seconds Pin Description:

1. the VDD power supply 3.5～5.5V DC
2. DATA serial data, a single bus
3. NC, empty pin
4. GND ground, the negative power

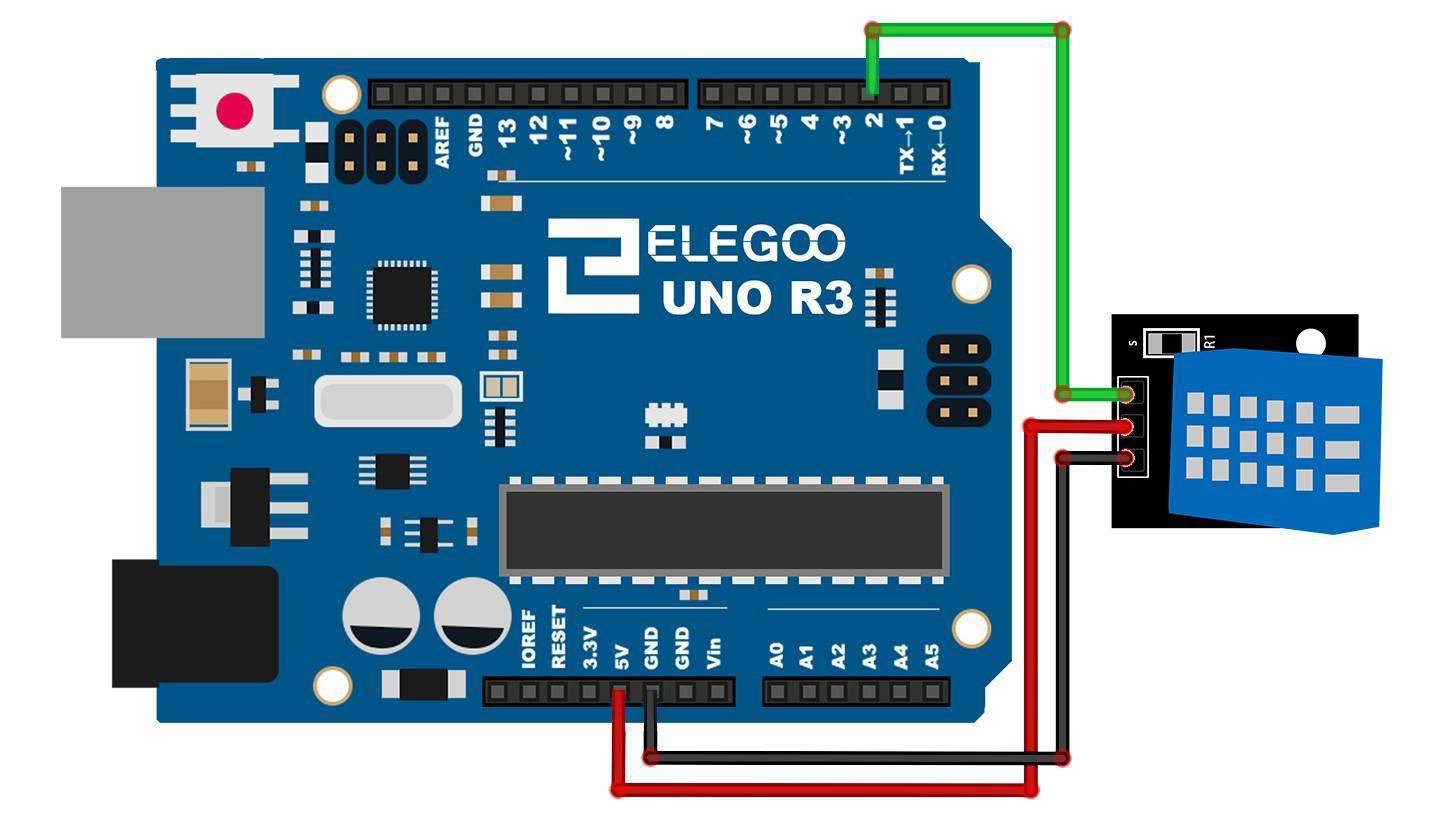
**92** / **223**

## Connection

### Schematic



**93** / **223**



**Wiring diagram**

**94** / **223**

As you can see we only need 3 connections to the sensor, since one of the pin is not used.

The connections are: Voltage, Ground and Signal which can be connected to any Pin on our UNO.

## Code

After wiring, please open the program in the code folder- DHT11\_Example and click UPLOAD to upload the program. See Lesson 5 in part 1 for details about program uploading if there are any errors.

Before you can run this, make sure that you have installed the <DHT> library

or re-install it, if necessary. Otherwise, your code won't work.

For details about the tutorial on the loading of library file, see Lesson 5 in part 1.

|  |
| --- |
| static const int DHT\_SENSOR\_PIN = 2; |

static

[Variable Scope & Qualifiers]

Description

The static keyword is used to create variables that are visible to only one function. However unlike local variables that get created and destroyed every time a function is called, static variables persist beyond the function call, preserving their data between function calls.

Variables declared as static will only be created and initialized the first time a function is called.

|  |
| --- |
| float temperature;  float humidity; |

float

[Data Types]

Description

Datatype for floating-point numbers, a number that has a decimal point. Floating-point numbers are often used to approximate analog and continuous values because they have greater resolution than integers. Floating-point numbers can be as large as 3.4028235E+38 and as low as -3.4028235E+38. They are stored as 32 bits (4 bytes) of information.

Syntax

float var = val;

Parameters

var: variable name.

val: the value you assign to that variable.

float

[数据类型]

描述

浮点数的数据类型，一个带小数点的数字。 浮点数通常用于近似模拟和连续值，因为它们的范围高于整数。 浮点数可以大到3.4028235E + 38并且低至-3.4028235E + 38。 它们存储为32位（4字节）信息。

句法

float var = val;d

参数

var：变量名。

val：您为该变量赋值的值。

|  |
| --- |
| 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 );  } |

Bool

[Data Types]

Description

A bool holds one of two values, true or false. (Each bool variable occupies one byte of memory.)

Syntax

bool var = val;

Parameters

var: variable name.

val: the value to assign to that variable.

(unsigned long)

[Conversion]

Description

Converts a value to the unsigned long data type.

Syntax

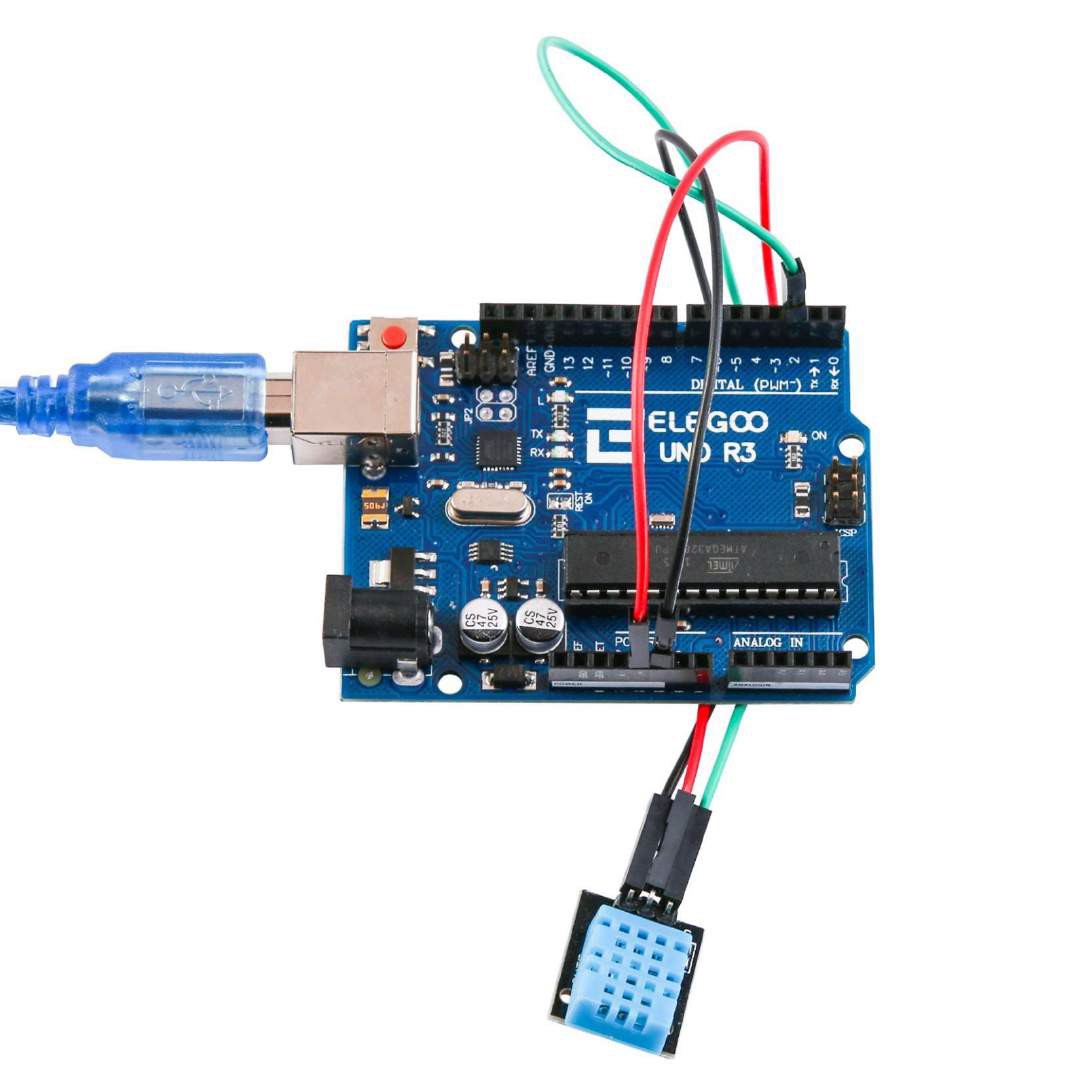
(unsigned long)x

Parameters

x: a value of any type

Returns

unsigned long



**Example picture**

**95** / **223**

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 22 degree)

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.

