

# DHT22 library for Arduino

## 1.0.0

Generated by Doxygen 1.8.14



# Contents

<b>1</b>	<b>DHT22 temperature and humidity sensor library for Arduino</b>	<b>1</b>
<b>2</b>	<b>Class Index</b>	<b>5</b>
2.1	Class List . . . . .	5
<b>3</b>	<b>File Index</b>	<b>7</b>
3.1	File List . . . . .	7
<b>4</b>	<b>Class Documentation</b>	<b>9</b>
4.1	DHT22 Class Reference . . . . .	9
4.1.1	Detailed Description . . . . .	9
4.1.2	Constructor & Destructor Documentation . . . . .	9
4.1.2.1	DHT22() . . . . .	9
4.1.3	Member Function Documentation . . . . .	10
4.1.3.1	available() . . . . .	10
4.1.3.2	begin() . . . . .	10
4.1.3.3	readHumidity() . . . . .	10
4.1.3.4	readTemperature() . . . . .	11
<b>5</b>	<b>File Documentation</b>	<b>13</b>
5.1	src/DHT22.cpp File Reference . . . . .	13
5.1.1	Detailed Description . . . . .	13
5.2	src/DHT22.h File Reference . . . . .	13
5.2.1	Detailed Description . . . . .	14
5.2.2	Macro Definition Documentation . . . . .	14
5.2.2.1	DHT22_MIN_READ_INTERVAL . . . . .	14
5.2.2.2	DHT22_NUM_DATA_BITS . . . . .	14
	<b>Index</b>	<b>15</b>



# Chapter 1

## DHT22 temperature and humidity sensor library for Arduino

This is an optimized AM2303 temperature and humidity sensor on a [DHT22](#) breakout.

### Library features

- Synchronous 16-bit temperature read
- Synchronous 16-bit humidity read

### AM2303 specifications

- Voltage: 3.3 .. 5V
- Ultra-low power:
  - Typical 15uA dormancy
  - Typical 500uA measuring
- Single wire serial interface
- Humidity:
  - Range: 0 .. 99.9 RH (Relative Humidity)
  - Resolution: 0.1 RH
  - Accuracy: +/-2 RH (at 25 degree Celsius)
- Temperature:
  - Range: -40 .. +125 degree Celsius
  - Resolution: 0.1 degree Celsius
  - Accuracy: +/- 0.4 degree Celsius
- Minimum read interval: 2000 ms

### Hardware

Connection [DHT22](#) - Arduino

DHT22	Arduino UNO/Nano/Leonardo/Mega2560
GND	GND
VCC	5V (or 3.3V)
DAT	D2

#### Connection DHT22 - ESP8266

DHT22	ESP8266 / WeMos D1 R2 / ESP12E / NodeMCU
GND	GND
VCC	3.3V
DAT	Arduino pin 2 -> GPIO4 = D4

**Note:** Some ESP8266 boards uses Arduino pin 2 -> GPIO4 which is D4 text on the board. Make sure you're using the right pin.

#### Connection DHT22 - Lolin32

DHT22	WeMos Lolin32
GND	GND
VCC	3.3V
DAT	2

### Supported Arduino Boards

- All ATmega328P MCU:
  - Arduino UNO
  - Arduino Nano
- All ATmega32U4 MCU's:
  - Arduino Leonardo
  - Pro Micro
- All ATmega2560 MCU's:
  - Arduino Mega2560
- All ESP8266 boards:
  - WeMos D1 R2
  - NodeMCU
- All Lolin32 boards:
  - WeMos Lolin32
- Other MCU's may work, but are not tested.

### Library dependencies

- None

## Documentation

[Doxygen](#)

[AM2303 datasheet](#)

[DHT22 datasheet](#)

## Examples

Examples | [ErriezDHT22](#) | [Example](#)

## Usage

### Initialization

```
{c++}
#include <DHT22.h>

// Connect DHT22 data pin to Arduino DIGITAL pin
#define DHT22_PIN 2

DHT22 sensor = DHT22(DHT22_PIN);

void setup()
{
    // Initialize serial port
    Serial.begin(115200);
    Serial.println(F("DHT22 temperature and humidity sensor example\n"));

    // Initialize sensor
    sensor.begin();
}
```

### Read temperature and humidity

```
{c++}
void loop()
{
    // Check minimum interval of 2000 ms between sensor reads
    if (sensor.available()) {
        // Read temperature from sensor
        int16_t temperature = sensor.readTemperature();

        // Read humidity from sensor
        int16_t humidity = sensor.readHumidity();

        // Print temperature
        Serial.print(F("Temperature: "));
        Serial.print(temperature / 10);
        Serial.print(F("."));
        Serial.print(temperature % 10);
        Serial.println(F(" *C"));

        // Print humidity
        Serial.print(F("Humidity: "));
        Serial.print(humidity / 10);
        Serial.print(F("."));
        Serial.print(humidity % 10);
        Serial.println(F(" %\n"));
    }
}
```

### Serial output

DHT22 temperature and humidity sensor example

Temperature: 17.7 \*C  
Humidity: 41.0 %

Temperature: 17.8 \*C  
Humidity: 41.1 %

...





## Chapter 2

# Class Index

### 2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">DHT22</a>	
<a href="#">DHT22</a> sensor class	9



## Chapter 3

# File Index

### 3.1 File List

Here is a list of all documented files with brief descriptions:

src/ <a href="#">DHT22.cpp</a>	
<a href="#">DHT22</a> (AM2303) Humidity and Temperature sensor example for Arduino	13
src/ <a href="#">DHT22.h</a>	
<a href="#">DHT22</a> (AM2303) Humidity and Temperature sensor example for Arduino	13



## Chapter 4

# Class Documentation

### 4.1 DHT22 Class Reference

DHT22 sensor class.

```
#include <DHT22.h>
```

#### Public Member Functions

- [DHT22](#) (uint8\_t pin)  
*Constructor [DHT22](#) sensor.*
- void [begin](#) ()  
*Initialize sensor.*
- bool [available](#) ()  
*Check if new temperature or humidity read is allowed.*
- int16\_t [readTemperature](#) ()  
*Read temperature from sensor.*
- int16\_t [readHumidity](#) ()  
*Read humidity from sensor.*

#### 4.1.1 Detailed Description

DHT22 sensor class.

Definition at line 58 of file DHT22.h.

#### 4.1.2 Constructor & Destructor Documentation

##### 4.1.2.1 DHT22()

```
DHT22::DHT22 (
    uint8_t pin ) [explicit]
```

Constructor [DHT22](#) sensor.

**Parameters**

<i>pin</i>	Data pin sensor.
------------	------------------

Definition at line 37 of file DHT22.cpp.

### 4.1.3 Member Function Documentation

#### 4.1.3.1 available()

```
bool DHT22::available ( )
```

Check if new temperature or humidity read is allowed.

The application should call this function and check if a new temperature and humidity can be read to prevent too fast sensor reads.

**Returns**

true: Available, interval between sensor reads  $\geq$  2000 ms. false: Not available, interval between sensor reads too short.

Definition at line 79 of file DHT22.cpp.

#### 4.1.3.2 begin()

```
void DHT22::begin ( )
```

Initialize sensor.

Call this function from setup().

Definition at line 59 of file DHT22.cpp.

#### 4.1.3.3 readHumidity()

```
int16_t DHT22::readHumidity ( )
```

Read humidity from sensor.

**Returns**

Signed humidity with last digit after the point.  $\sim 0$ : An error occurred

Definition at line 121 of file DHT22.cpp.

#### 4.1.3.4 readTemperature()

```
int16_t DHT22::readTemperature ( )
```

Read temperature from sensor.

Returns the actual temperature, or a cached temperature when read interval is too short.

##### Returns

Signed temperature with last digit after the point ~0: An error occurred

Definition at line 99 of file DHT22.cpp.

The documentation for this class was generated from the following files:

- [src/DHT22.h](#)
- [src/DHT22.cpp](#)





## Chapter 5

# File Documentation

### 5.1 src/DHT22.cpp File Reference

**DHT22** (AM2303) Humidity and Temperature sensor example for Arduino.

```
#include "DHT22.h"
```

#### 5.1.1 Detailed Description

**DHT22** (AM2303) Humidity and Temperature sensor example for Arduino.

Source: <https://github.com/Erriez/ErriezDHT22>

### 5.2 src/DHT22.h File Reference

**DHT22** (AM2303) Humidity and Temperature sensor example for Arduino.

```
#include <Arduino.h>
```

#### Classes

- class **DHT22**  
*DHT22 sensor class.*

#### Macros

- #define **DHT22\_MIN\_READ\_INTERVAL** 2000  
*Enable debug prints to Serial.*
- #define **DHT22\_NUM\_DATA\_BITS** (5 \* 8)
- #define **DEBUG\_PRINTLN**(...) {}  
*Debug print configuration.*

### 5.2.1 Detailed Description

[DHT22](#) (AM2303) Humidity and Temperature sensor example for Arduino.

Source: <https://github.com/Erriez/ErriezDHT22>

### 5.2.2 Macro Definition Documentation

#### 5.2.2.1 DHT22\_MIN\_READ\_INTERVAL

```
#define DHT22_MIN_READ_INTERVAL 2000
```

Enable debug prints to Serial.

Minimum interval between sensor reads in milli seconds

Definition at line 40 of file DHT22.h.

#### 5.2.2.2 DHT22\_NUM\_DATA\_BITS

```
#define DHT22_NUM_DATA_BITS (5 * 8)
```

Number of data bits is 5 Bytes \* 8 bits: 1 Byte: Humidity high 1 Byte: Humidity low 1 Byte: Temperature high 1 Byte: Temperature low 1 Byte: Parity

Definition at line 48 of file DHT22.h.

# Index

available

DHT22, [10](#)

begin

DHT22, [10](#)

DHT22, [9](#)

available, [10](#)

begin, [10](#)

DHT22, [9](#)

readHumidity, [10](#)

readTemperature, [10](#)

DHT22.h

DHT22\_MIN\_READ\_INTERVAL, [14](#)

DHT22\_NUM\_DATA\_BITS, [14](#)

DHT22\_MIN\_READ\_INTERVAL

DHT22.h, [14](#)

DHT22\_NUM\_DATA\_BITS

DHT22.h, [14](#)

readHumidity

DHT22, [10](#)

readTemperature

DHT22, [10](#)

src/DHT22.cpp, [13](#)

src/DHT22.h, [13](#)