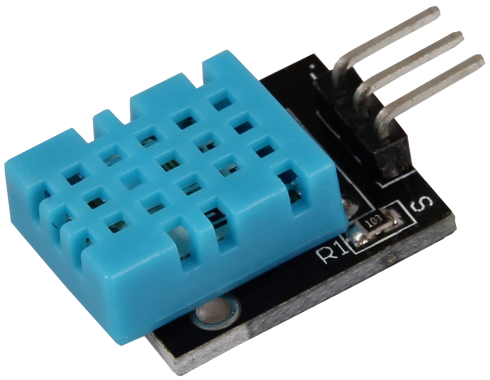


Combi Sensor

KY-015

TABLE OF CONTENTS

1. Overview

2. Pin-Assignment

3. Code Example Arduino

4. Code Example Raspberry Pi

5. Information and take-back obligations

6. Support

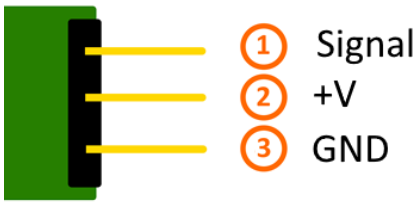
1. OVERVIEW

Dear customer,
Thank you for choosing our product.
In the following, we will show you what to observe during commissioning and use. Should you encounter any unexpected problems during use, please do not hesitate to contact us.

Chipset	DHT11
Communication protocoll	1-Wire
Measuring range air humidity	20 - 90% RH
Measuring range temperature	0 - 50°C

Please note that the sensor only provides a new measurement result every 2 seconds. It is designed for long-term recordings.

2. PIN-ASSIGNMENT



Pin Assignment Arduino	
Signal	Pin D2
+V	Pin 5V
GND	Pin GND

Pin Assignment Raspberry Pi	
Signal	GPIO23 [Pin 16]
+V	3,3V [Pin 1]
GND	Masse [Pin 6]

Copyright by Joy-IT

3. CODE EXAMPLE ARDUINO

This sensor does not output its measurement result as an analog signal on an output pin, but communicates this digitally coded. There are several ways to control this sensor module. The **Adafruit_DHT Library**, published by Adafruit under the following link under the **OpenSource MIT license**, has proven to be particularly accessible.

The example below uses this library - we recommend downloading it from Github, unpacking it and copying it to the Arduino Library folder located by default at (C:\User\[Username]\Documents\Arduino\libraries) to make it available for this code example and subsequent projects. Alternatively, this is also included in the download package.

You can download the code example here or copy and paste the following sections:

```
// Adafruit_DHT Library is inserted
#include "DHT.h"

// Here the respective input pin can be declared
#define DHTPIN 2

// The sensor is initialized
#define DHTTYPE DHT11   // DHT 11
DHT dht(DHTPIN, DHTTYPE);

void setup()
{
  Serial.begin(9600);
  Serial.println("KY-015 Test - Temperatur und Luftfeuchtigkeits-Test:");

  // Measurement is started
  dht.begin();
}

// main loop
// The program starts the measurement and reads out the measured values
// A pause of 2 seconds is inserted between the measurements,
// so that a new measurement can be recorded during the next run.
void loop() {

  // Zwei Sekunden Pause zwischen den Messungen
  delay(2000);

  // Two seconds pause between measurements
  float h = dht.readHumidity();
  // Temperature is measured
  float t = dht.readTemperature();
```

```
  // Here it is checked whether the measurements have run without errors.
  // If an error is detected, an error message is displayed here
  if (isnan(h) || isnan(t)) {
    Serial.println("Fehler beim Auslesen des Sensors");
    return;
  }

  // Output to the serial console
  Serial.println("-----------------------------------------------------------");
  Serial.print("Luftfeuchtigkeit: ");
  Serial.print(h);
  Serial.print(" %\t");
  Serial.print("Temperatur: ");
  Serial.print(t);
  Serial.print(char(186)); //Output <°> Symbol
  Serial.println("C ");
  Serial.println("-----------------------------------------------------------");
  Serial.println(" ");
}
```

4. CODE EXAMPLE RASPBERRY PI

The program uses the corresponding **Adafruit_Python_DHT library** from Adafruit to control the DHT11 sensor installed on this sensor module. These were published under the following link unter under the MIT OpenSource-Licence. This must be installed first.

To do this, open a terminal window on your Raspberry Pi and enter the following commands:

```
sudo apt-get install git
```

```
sudo apt-get update
```

```
sudo apt-get install build-essential python-dev
```

Use the following command to download the latest version of the Adafruit_DHT Library:

```
git clone https://github.com/adafruit/Adafruit_Python_DHT.git
```

Now install the library:

```
cd Adafruit_Python_DHT/
```

```
sudo python setup.py install
```

You can download the code example here. Alternatively, you can copy the following section.

Then start the example with the following command:

```
sudo python KY-015-RPi_Kombi-Sensor_Temperatur_Feuchtigkeit.py
```

```
#!/usr/bin/python
# coding=utf-8

# Required modules are imported and set up
import RPi.GPIO as GPIO
import Adafruit_DHT
import time

# The pause of two seconds between measurements is set here
sleeptime = 2

# Sensor should be set to Adafruit_DHT.DHT11,
# Adafruit_DHT.DHT22, or Adafruit_DHT.AM2302.
DHTSensor = Adafruit_DHT.DHT11

# The pin to which the sensor module is connected can be declared here
GPIO_Pin = 23

print('KY-015 Sensortest - Temperatur und Luftfeuchtigkeit')

try:
    while(1):
        # Measurement is started and the result is written to the corresponding vari-
ables

        Luftfeuchte, Temperatur = Adafruit_DHT.read_retry(DHTSensor, GPIO_Pin)

        print("-------------------------------------------------------------")
        if Luftfeuchte is not None and Temperatur is not None:

            # Das gemessene Ergebnis wird in der Konsole ausgegeben
            print('Temperatur = {0:0.1f}°C  | rel. Luftfeuchtigkeit =
{1:0.1f}%'.format(Temperatur, Luftfeuchte))

        # Because the Raspberry Pi is disadvantaged for real-time applications due to
the Linux operating system,
        # communication may fail due to timing problems.
        # In this case an error message is displayed - a result should be available
at the next attempt
        else:
            print('Fehler beim Auslesen - Bitte warten auf nächsten Versuch!')
        print("-------------------------------------------------------------")
        print("")
        time.sleep(sleeptime)

# Cleanup after the program is finished
except KeyboardInterrupt:
    GPIO.cleanup()
```

5. INFORMATION AND TAKE-BACK OBLIGATIONS

Symbol on electrical and electronic equipment

This crossed-out dustbin means that electrical and electronic equipment does not belong in household waste. You must return the old devices to a collection point. Before disposal, you must separate old batteries and accumulators which are not enclosed in the old device from it.

Return options
As an end user, you can dispose of your old device (which essentially fulfils the same function as the new one purchased from us) free of charge when purchasing a new device. Small appliances where no external dimensions are larger than 25 cm can be delivered in normal household quantities, irrespective of the purchase of a new appliance.

Possibility of return at our company location during opening hours
Simac Ltd, Pascalstr. 8, 47506 Neukirchen-Vluyn, Germany

Possibility of return in your area
We will send you a parcel stamp with which you can return the device to us free of charge. Please contact us by e-mail at Service@joy-it.net or by phone.

Information about packaging
If you do not have suitable packaging material or do not wish to use your own, please contact us and we will send you suitable packaging.

6. SUPPORT

We also support you after your purchase. If there are any questions left, or if you encounter any problems, please feel free to contact us by mail, phone or by our ticket-system on our website.

E-Mail: service@joy-it.net
Ticket-System: http://support.joy-it.net
Phone: +49 (0)2845 98469 – 66 (11- 18 o‘clock)

For more information, please visit our website:

www.joy-it.net

Copyright by Joy-IT