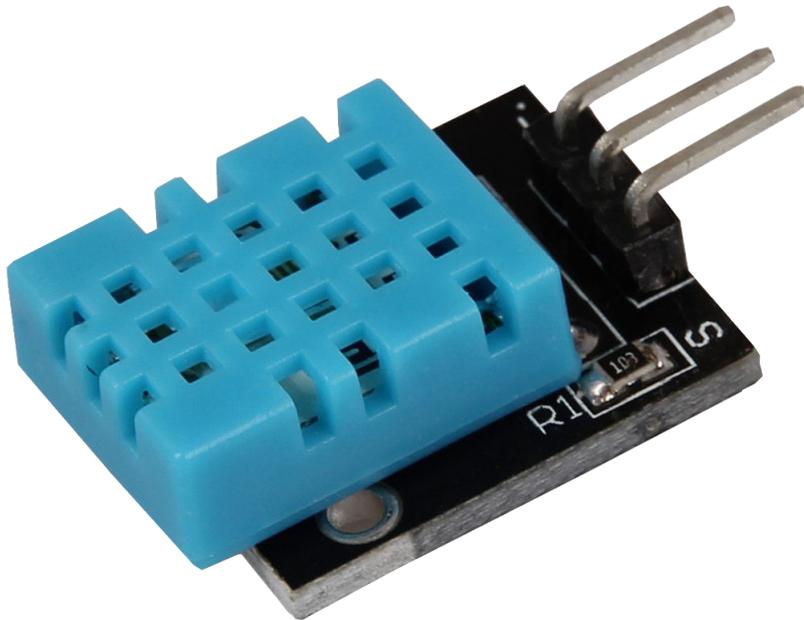# KY-015

Temperature and humidity sensor

1. GENERAL INFORMATION

Dear customer,

Thank you for purchasing our product. In the following, we will show you how to use your product.

Should you encounter any unexpected problems during use, please do not hesitate to contact us.

Please note that the sensor provides a new measurement result only about every 2 seconds; therefore, it is rather designed for long-term recordings.

Chipset	DHT11
Communication protocol	User defined protocol that uses a single bus (not 1-Wire)
Measuring range of air humidity	20 - 90 % RH
Measuring range of temperature	0 - 50 °C

1. Connection

Arduino	DHT11
Pin 2	Signal
Pin 5V	Vcc
Pin GND	GND
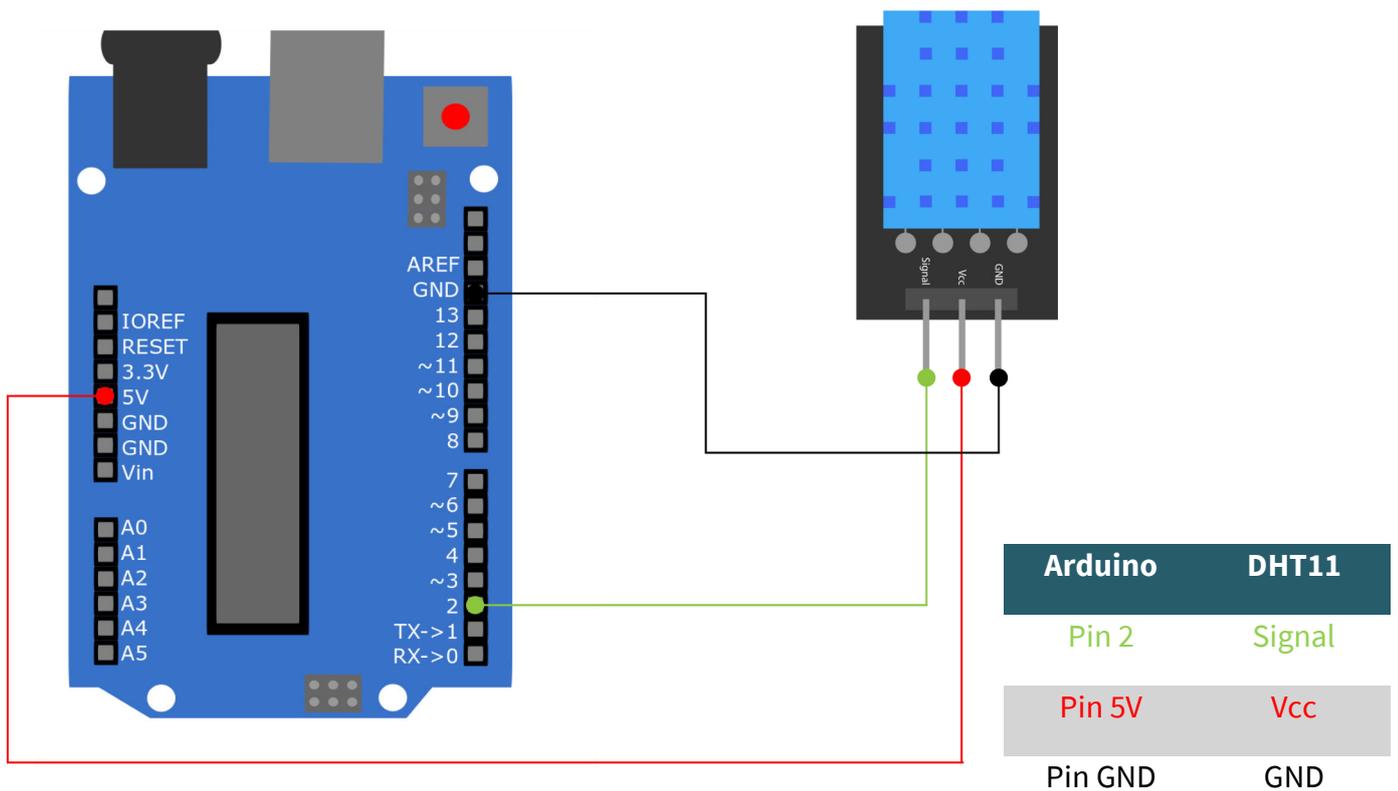
2. Code example

This sensor does not output its measurement result as an analog signal to an output pin, but communicates it digitally and encoded. There are several possibilities to control this sensor module. The **Adafruit DHT-sensor-library**, which the company Adafruit has published under the **MIT-License**, has proven to be particularly accessible. You can download the library **here** as a .zip file or add it directly into the Arduino IDE.

If you have downloaded the .zip file, you can include it under *Sketch → Library → .ZIP add Library …* . Restart your IDE afterwards. You can also include the library directly from the IDE. To do this, click on *Sketch → Include library → Manage libraries …* . There you can search for the *DHT sensor library* in the search bar and install it. Here you also have to restart the IDE afterwards.

Before you start the code, make sure that the correct board at *Tools → Board* and the correct COM port at *Tools → Port* is selected.

You can use the examples of the library, but you must note, that the initialisation of the sensor is related to DHT22 and not DHT11. To do so, comment out the corresponding line for DHT22 and remove the comment lines from the lines for DHT11.

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

// The respective input pin can be declared here
#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 - Test for Temperature and humidity:");

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

// Main program loop
// Teh 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 with the next run
void loop() {

  // Two seconds pause between measurements
  delay(2000);

  // Air humidity is measured
  float h = dht.readHumidity();
  // Temperature is measured
  float t = dht.readTemperature();

  // Here it is checked whether the measurements have been carried out without errors
  // If an error is detected, a error will be displayed
  if (isnan(h) || isnan(t)) {
    Serial.println("Error while reading the sensor");
    return;
  }

  // Output in serial console
  Serial.println("--------------------------------------------------------------");
  Serial.print("Air humidity: ");
  Serial.print(h);
  Serial.print(" %\t");
  Serial.print("Temperature: ");
  Serial.print(t);
  Serial.print(char(186)); //Output of <°> symbol
  Serial.println("C ");
  Serial.println("--------------------------------------------------------------");
  Serial.println(" ");
}
```
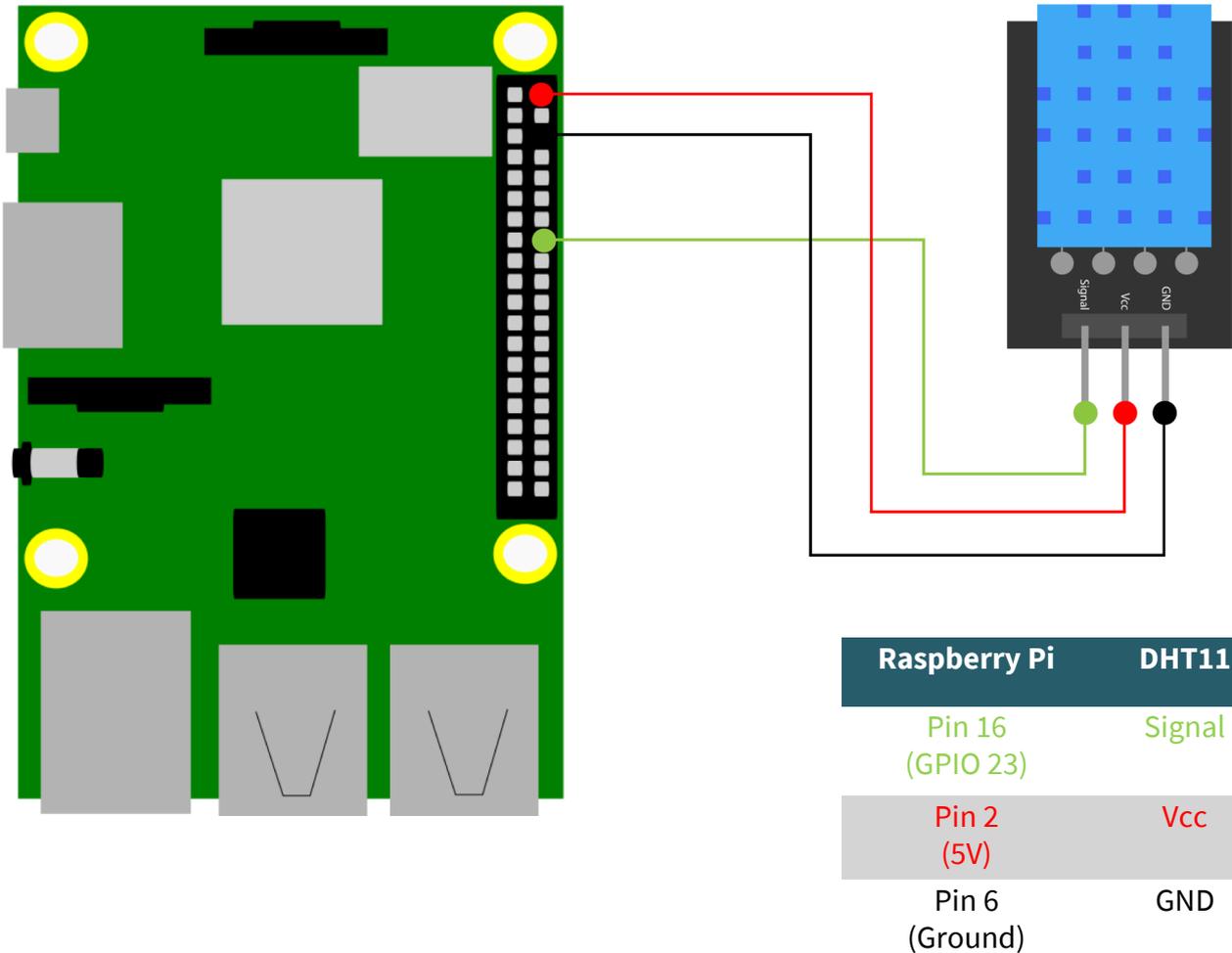
Copy the code into your Arduino DIE and upload it. Now open the serial
monitor and set the baud rate to 9600. Now the temperature and
humidity will be displayed.

1. Connection

Raspberry Pi	DHT11
Pin 16 (GPIO 23)	Signal
Pin 2 (5V)	Vcc
Pin 6 (Ground)	GND

2. Code example

We offer an edited library suitable for the use with the Raspberry Pi, which you can download **here**. First, you have to execute the following commands:

```
sudo apt-get update
git clone https://github.com/joy-it/Adafruit_Python_DHT.git
cd Adafruit_Python_DHT
sudo python3 setup.py install
```

Now you have installed the library. You can use the following command to open a file and copy the following code into it so that you can try out the DHT11 with a sample code.

```
nano dht11.py
```

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

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

# 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

# Here you can declare the pin to which the sensor module is connected
GPIO_Pin = 23

print('KY-015 sensor test - temperature and humidity)

try:
    while(1):
        # Measurement is started and the result is written to the corresponding variables
        Luftfeuchte, Temperatur = Adafruit_DHT.read_retry(DHTSensor, GPIO_Pin)

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

            # The measured result is displayed in the console
            print('Temperature = {0:0.1f}*C | Air humidity = {1:0.1f}%'.format(Temperatur, Luftfeuchte))
```

```python
        # Because the Raspberry Pi is disadvantaged by the Linux operating system for real-time applications,
        # communication may fail due to timing issues.
        # in this case an error message is displayed - a result should be available at the next attempt
        else:
            print('Error during readout - Please wait for next try!')

        print("----------------------------------------------------------------")
        print("")
        time.sleep(sleeptime)

# Clean up after the program is closed
except KeyboardInterrupt:
    sys.exit(0)
```

Now you can save the file with CTRL + O and close it with CTRL + X. Finally, enter the following command in the console to run the program.

```
sudo python3 dht11.py
```

4. OTHER INFORMATION

Our information and redemption obligation according to
the Electrical and Electronic Equipment Act (ElektroG)

Symbol on electrial and electronic products:
This crossed-out bin means that electrical and electronic products do **not** belong into the household waste. You must hand over your old appliance to a registration office. Before you can hand over the old appliance, you must remove used batteries and accumulators which are not enclosed by the device.

Return options:
As the end user, you can hand over with the purchase of a new device your old appliance (which has essentially the same functions as the new one) free of charge for disposal.
Smaller devices which do not have outer dimensions greater than 25 cm can be submitted independently of the purchase of a new product in normal household quantities.

Possibility of restitution at our company location during our opening hours:
Simac GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn

Possibility of restitution nearby:
We send you a parcel stamp with which you can send us your old appliance free of charge. For this possibility, you must contact us via e-mail at service@joy-it.net or via telephone.

Information about packaging:
Please package your old appliance safe during transport. Should you not have a suitable packaging material or you do not want to use your own material, you can contact us and we will send you an appropriate package.

5. SUPPORT

If any questions remained open or problems may arise after your purchase, we are available by e-mail, telephone and ticket support system to answer these.

E-Mail: service@joy-it.net
Ticket-system: http://support.joy-it.net
Telephone: +49 (0)2845 98469 – 66 (10 - 17 o'clock)

For further information visit our website:
www.joy-it.net