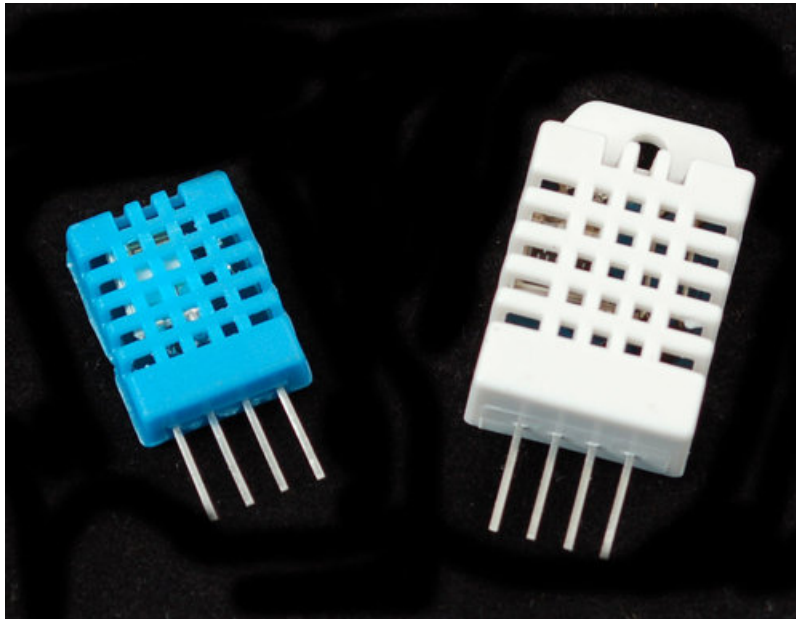


DHTxx Sensors

Created by lady ada



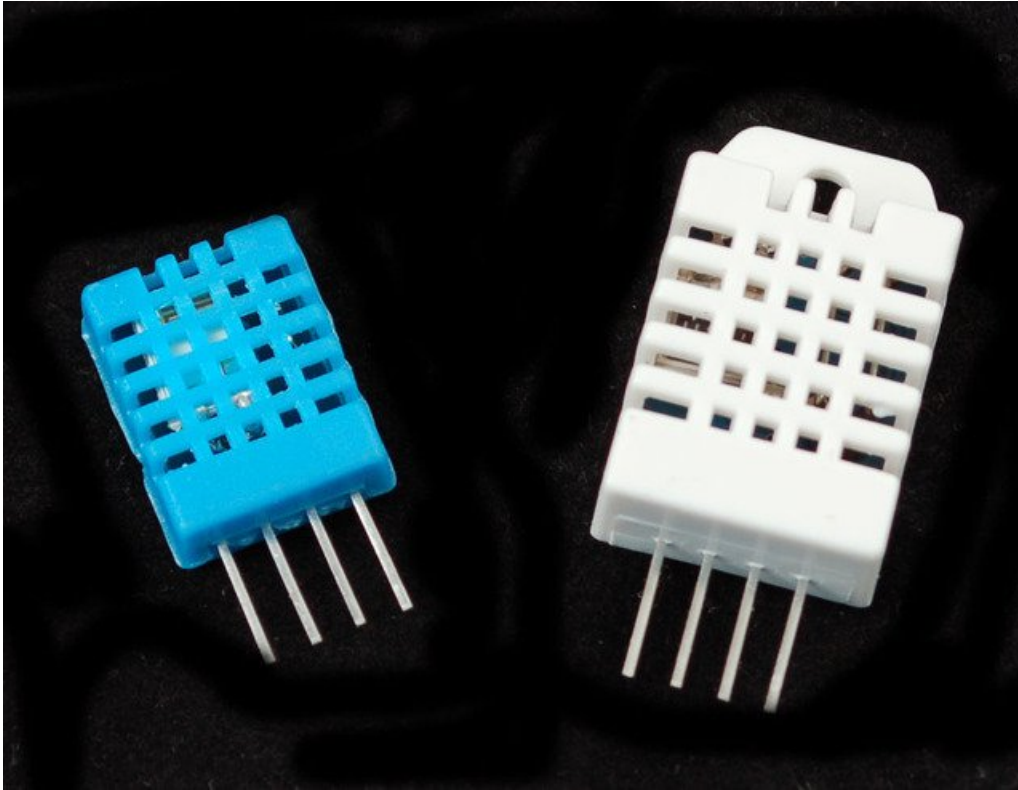
Last updated on 2017-10-25 04:30:33 PM UTC

Guide Contents

Guide Contents	2
Overview	3
DHT11 vs DHT22	3
Connecting to a DHTxx Sensor	5
Using a DHTxx Sensor	7
CircuitPython Code	9
Adafruit CircuitPython Module Install	9
Usage	9
Downloads	12

Overview

This tutorial covers the low cost [DHT temperature & humidity sensors](#). These sensors are very basic and slow, but are great for hobbyists who want to do some basic data logging. The DHT sensors are made of two parts, a capacitive humidity sensor and a [thermistor](#). There is also a very basic chip inside that does some analog to digital conversion and spits out a digital signal with the temperature and humidity. The digital signal is fairly easy to read using any microcontroller.



DHT11 vs DHT22

We have two versions of the DHT sensor, they look a bit similar and have the same pinout, but have different characteristics. Here are the specs:

DHT11

- Ultra low cost
- 3 to 5V power and I/O
- 2.5mA max current use during conversion (while requesting data)
- Good for 20-80% humidity readings with 5% accuracy
- Good for 0-50°C temperature readings $\pm 2^\circ\text{C}$ accuracy
- No more than 1 Hz sampling rate (once every second)
- Body size 15.5mm x 12mm x 5.5mm
- 4 pins with 0.1" spacing

DHT22

- Low cost
- 3 to 5V power and I/O
- 2.5mA max current use during conversion (while requesting data)

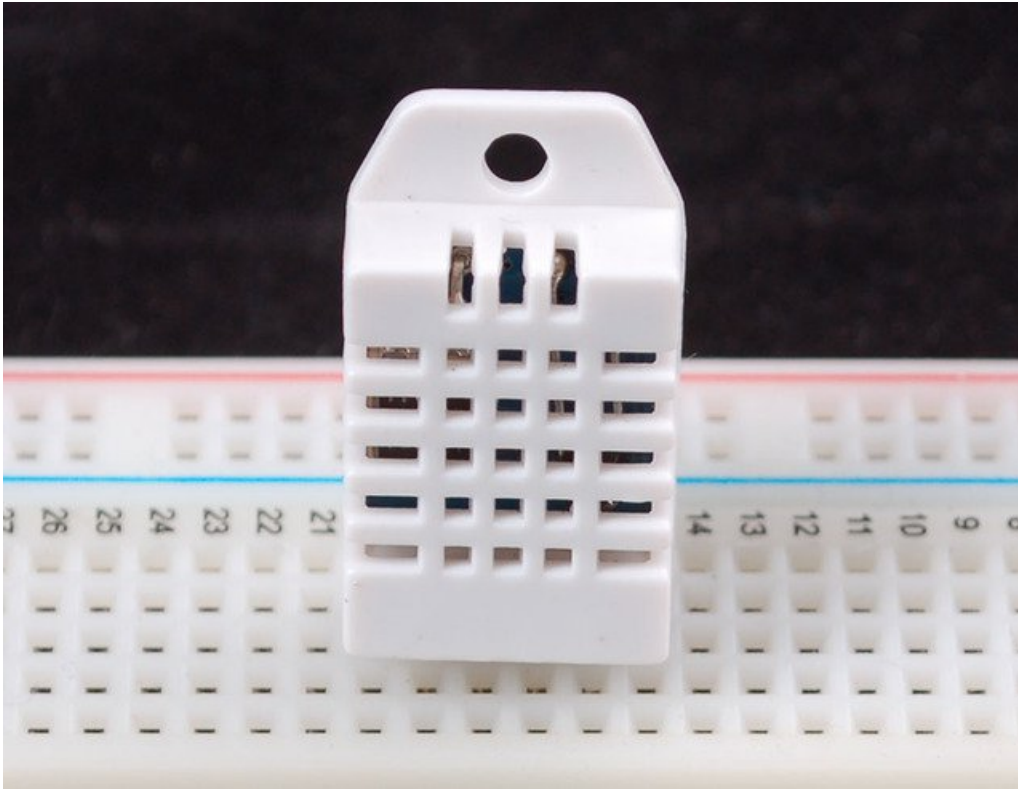
- Good for 0-100% humidity readings with 2-5% accuracy
- Good for -40 to 125°C temperature readings $\pm 0.5^{\circ}\text{C}$ accuracy
- No more than 0.5 Hz sampling rate (once every 2 seconds)
- Body size 15.1mm x 25mm x 7.7mm
- 4 pins with 0.1" spacing

As you can see, the [DHT22](#) is a little more accurate and good over a slightly larger range. Both use a single digital pin and are 'sluggish' in that you can't query them more than once every second or two.

You can pick up both the [DHT11](#) and [DHT22](#) from the [adafruit shop](#)!

Connecting to a DHTxx Sensor

Luckily it is trivial to connect to these sensors, they have fairly long 0.1"-pitch pins so you can plug them into any breadboard, perfboard or similar.

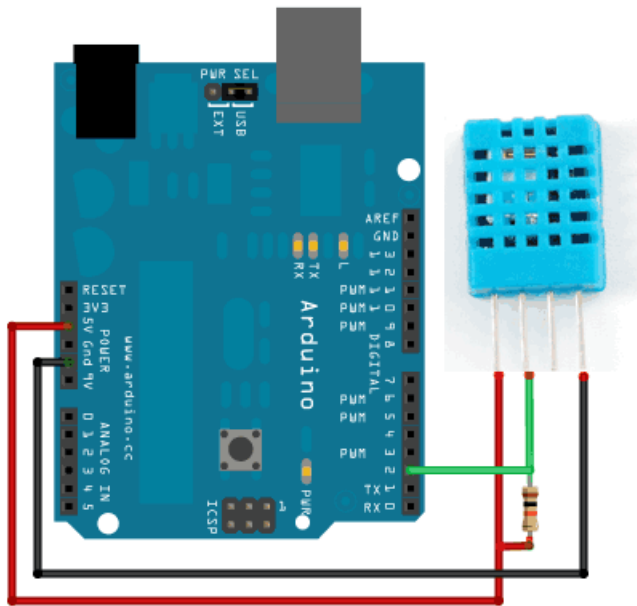


Likewise, it is fairly easy to connect up to the DHT sensors. They have four pins

- VCC (3 to 5V power)
- Data out
- Not connected
- Ground

Simply ignore pin 3, its not used. You will want to place a 10K resistor between VCC and the data pin, to act as a medium-strength pull up on the data line. The Arduino has built in pullups you can turn on but they're very weak, about 20-50K

This diagram shows how we will connect for the testing sketch. Connect data to pin 2, you can change it later to any pin.

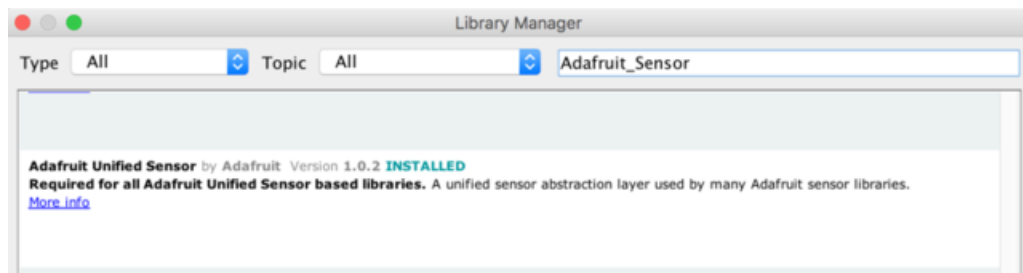


Using a DHTxx Sensor

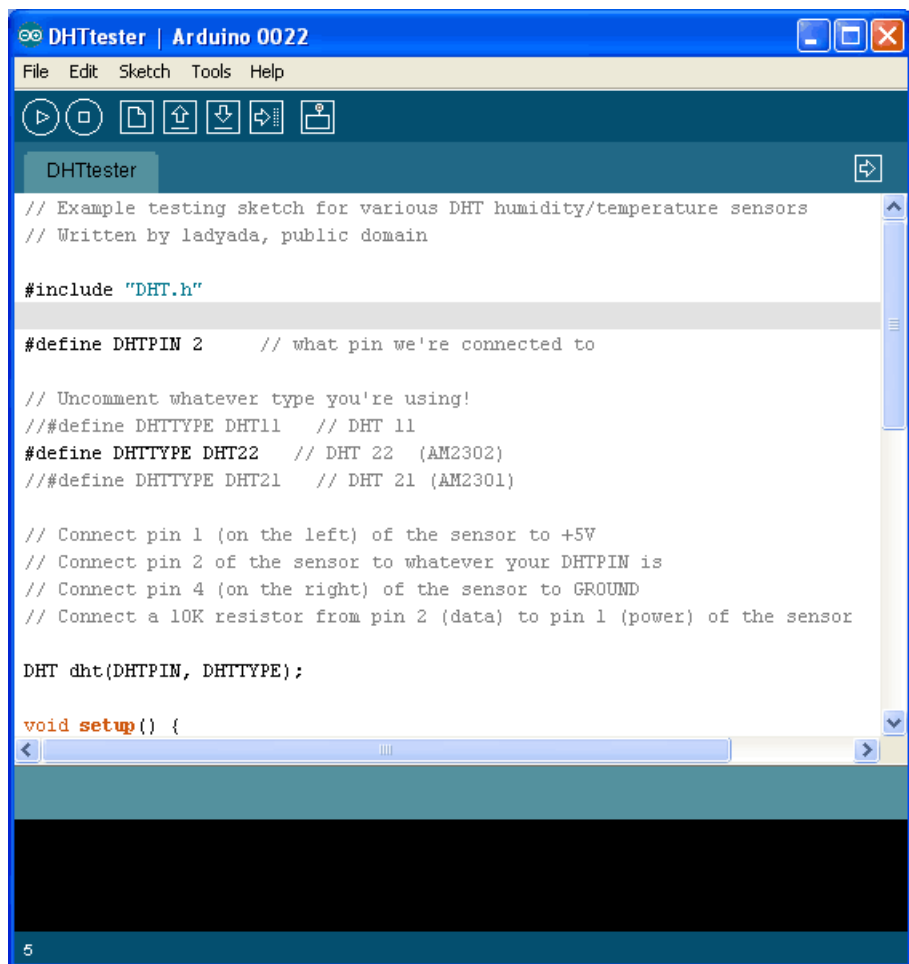
To test the sketch, we'll use an Arduino. You can use any microcontroller that can do microsecond timing, but since its a little tricky to code it up, we suggest verifying the wiring and sensor work with an Arduino to start.

Begin by downloading the DHT library from our [github repository](#). To download, click the **DOWNLOADS** button in the top right corner. Rename the uncompressed folder **DHT** and make sure that it contains the **dht.cpp** file and others. Then drag the **DHT** folder into the *arduinofolder/libraries/* folder. You may have to create that libraries sub-folder if it doesnt exist. Restart the IDE.

IMPORTANT: As of version 1.3.0 of the DHT library you will also need to install the [Adafruit_Sensor](#) library, which is available from the Arduino library manager:



Now load up the Examples->DHT->DHTtester sketch



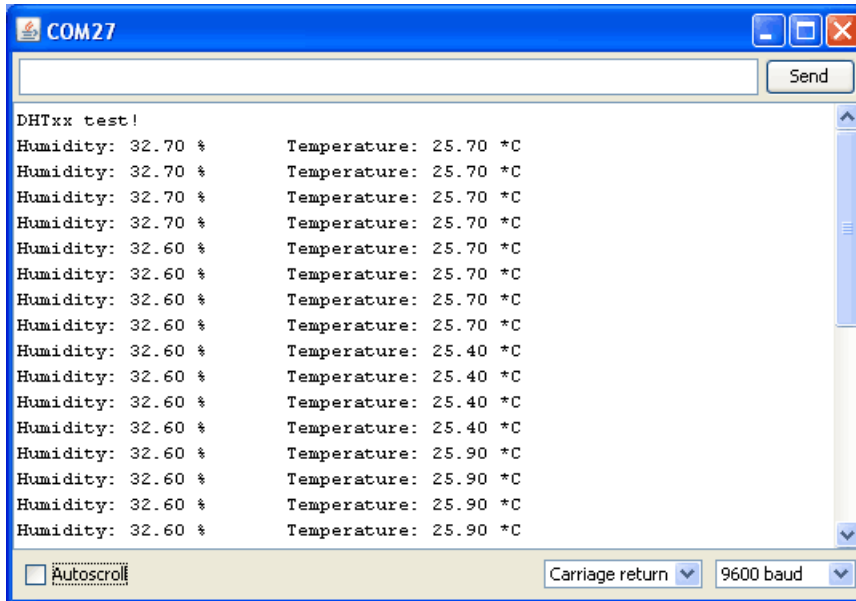
If you're using a **DHT11** sensor, comment out the line that sets the type:

```
//#define DHTTYPE DHT22  // DHT 22  (AM2302)
```

and uncomment the line that says:

```
#define DHTTYPE DHT11  // DHT 11
```

This will make the data appear correctly for the correct sensor. Upload the sketch!



You should see the temperature and humidity. You can see changes by breathing onto the sensor (like you would to fog up a window) which should increase the humidity.

CircuitPython Code

Adafruit CircuitPython Module Install

To use the DHT sensor with your [Adafruit CircuitPython](#) board you'll need to install the [Adafruit_CircuitPython_DHT](#) module on your board.

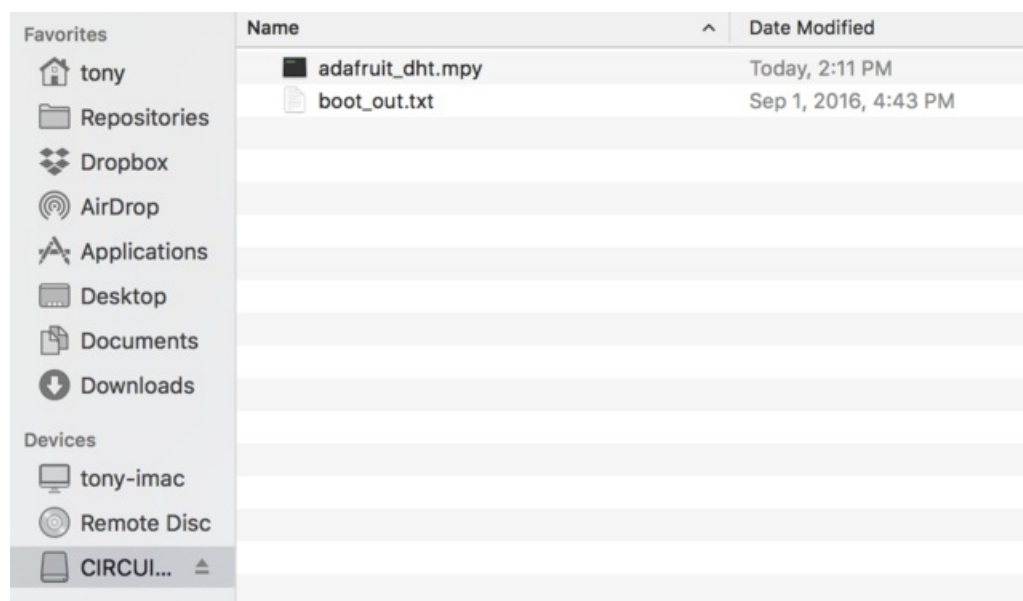
First make sure you are running the [latest version of Adafruit CircuitPython](#) for your board. In particular for **Gemma M0**, **Trinket M0**, and **M0 basic** boards you must be running **CircuitPython 2.1.0** or higher to have access to the necessary pulseio module!

Next you'll need to install the necessary libraries to use the hardware--carefully follow the steps to find and install these libraries from [Adafruit's CircuitPython library bundle](#). For example the Circuit Playground Express guide has [a great page on how to install the library bundle](#) for both express and non-express boards.

Remember for non-express boards like the Trinket M0, Gemma M0, and Feather/Metro M0 basic you'll need to manually install the necessary libraries from the bundle:

- `adafruit_dht.mpy`

Before continuing make sure your board's lib folder or root filesystem has the `adafruit_dht.mpy` module copied over.



Usage

To demonstrate the usage of the DHT sensor module you can connect to your board's serial REPL and run Python code to read the temperature and humidity.

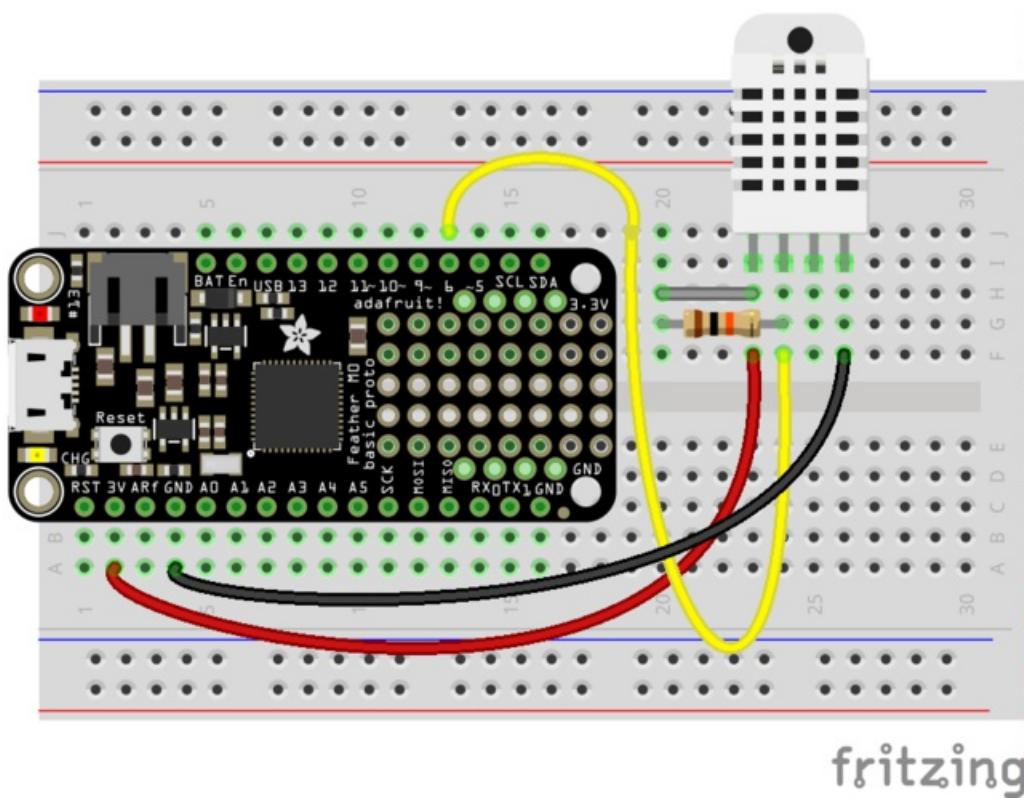
First [connect to the board's serial REPL](#) so you are at the CircuitPython >>> prompt.

Next import the `board` and `adafruit_dht` modules, these are necessary modules to initialize and access the sensor:

```
import board
import adafruit_dht
```

Be sure you've wired your DHT sensor to your board as shown in the previous pages. You must connect the power, ground, and signal line to your board. The signal line should be connected to a digital I/O pin and in some cases requires a 10k pull-up resistor to power as mentioned (when using the simple DHT11 and DHT22 sensors). Make note of which digital input you've connected the signal line to before continuing!

In this example we'll use a Feather M0 and DHT22 sensor connected to pin D6:



Fritzing Source

<https://adafru.it/A0o>

You may also want to try powering the DHT sensor from 5V (we found sometimes it really needs more power) but still having the 10K pull-up resistor to 3.3V volts)

Now create an instance of either the **DHT11** or **DHT22** class, depending on the type of sensor you're using (for the AM2302 sensor use the DHT22 class). You must pass in the pin which is connected to the signal line, for example a DHT22 or AM2302 sensor connected to board pin D6 would need this code:

```
dht = adafruit_dht.DHT22(board.D6)
```

Note for a DHT11 sensor you'd instead use `adafruit_dht.DHT11` in place of the `adafruit_dht.DHT22` code above.

At this point you're all set and ready to start reading the temperature and humidity! You can do this by reading the **temperature** property which returns temperature in degrees Celsius:

```
dht.temperature
```

```
>>> dht.temperature
22.2
>>>
```

To read the humidity grab the value of the **humidity** property, it will return the percent humidity as a floating point value from 0 to 100%:

```
dht.humidity
```

```
>>> dht.humidity
53.2
>>>
```

In most cases you'll always get back a temperature or humidity value when requested, but sometimes if there's electrical noise or the signal was interrupted in some way you might see an exception thrown to try again. It's normal for these sensors to sometimes be hard to read and you might need to make your code retry a few times if it fails to read. However if you always get errors and can't ever read the sensor then double check your wiring (don't forget the pull-up resistor if needed!) and the power to the device.

That's all there is to reading temperature and humidity from CircuitPython with the DHT series of sensors!

Downloads

- [Arduino library and example code for DHT sensors](#)
- [Adafruit_Sensor library](#) (required by the DHT library above)
- [DHT11 datasheet](#) (in chinese, so see the DHT22 datasheet too!)
- [DHT22 datasheet](#)