



AM2315 - Encased I2C Temperature/ Humidity Sensor

Created by Kattni Rembor



<https://learn.adafruit.com/am2315-encased-i2c-temperature-humidity-sensor>

Last updated on 2024-06-03 02:30:44 PM EDT

Table of Contents

Overview	3
Arduino Code	4
<ul style="list-style-type: none">• Install Adafruit Sensor• Download Adafruit_AM2315• Load Demo	
Python & CircuitPython	6
<ul style="list-style-type: none">• CircuitPython Microcontroller Wiring• Python Computer Wiring• CircuitPython Installation of AM2320 Library• Python Installation of AM2320 Library• CircuitPython & Python Usage• Full Example Code	
Python Docs	10

Overview



Finally we have an I2C-interface temperature & humidity sensor in a nice enclosed style. This sensor contains a thermistor temperature sensor and a capacitive humidity sensor. A small microcontroller inside does the readings and provides a simple I2C interface for reading the finished & calibrated output data. Especially nice is that this sensor is in a rugged case with mounting bracket, which makes it way superior to a normal PCB-mounted sensor.



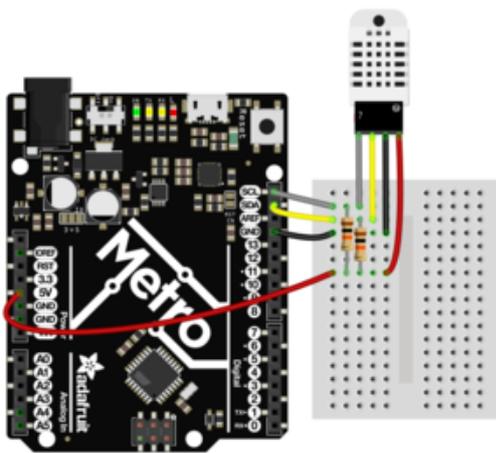
While it is not rated as 'weatherproof', this sensor would do much better for sensing where there might be wind, rain, zombies, etc. than SHT PCB-breakout sensors, and

the i2c interface makes it easier to interface with microcomputers that can't do the delicate timing of the DHT sensors.

Simply connect it up using I2C and you're ready to start sensing temperature and humidity. This guide will cover how to wire it up, and provide code examples. Let's get started!

Arduino Code

You can easily wire this breakout to any microcontroller, we'll be using an Arduino compatible. For another kind of microcontroller, just make sure it has I2C, then port the code - its pretty simple stuff!



Connect **Red wire** to the power supply, 3-5V is fine. Use the same voltage that the microcontroller logic is based off of. For most Arduinos, that is 5V

Connect **Black wire** to common power/ data ground

Connect the **White wire** pin to the I2C clock **SCL** pin on your microcontroller

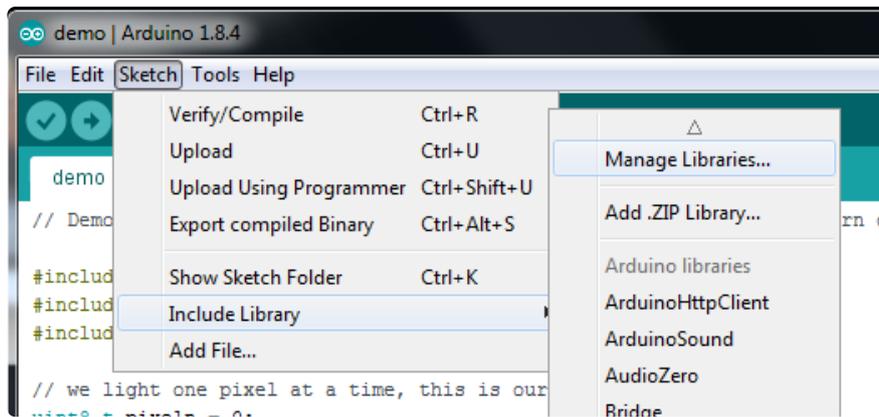
Connect the **Yellow wire** pin to the I2C data **SDA** pin on your microcontroller

Don't forget to install two 2K-10K pullup resistors from **SDA** and **SCL** to **Vin** (3V or 5V, whatever logic level your microcontroller is)!

The AM2315 has a default I2C address of **0x5C** and cannot be changed.

Install Adafruit Sensor

Open up the Library Manager in the Arduino IDE...



Search for **Adafruit_Sensor** to locate the Unified Sensor library and **Install** it



Download Adafruit_AM2315

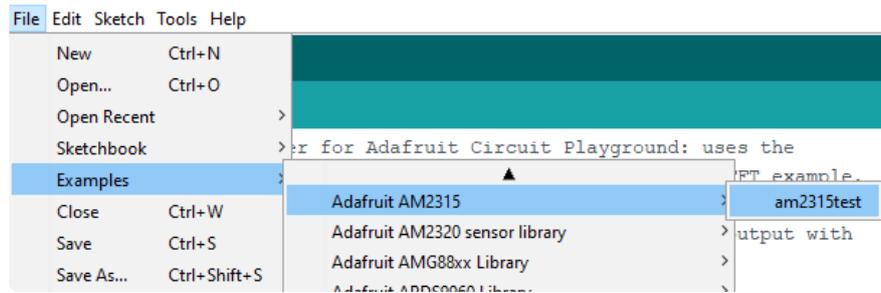
To begin reading sensor data, you will need to also install the [Adafruit_AM2315 library \(https://adafru.it/CGt\)](https://adafru.it/CGt).



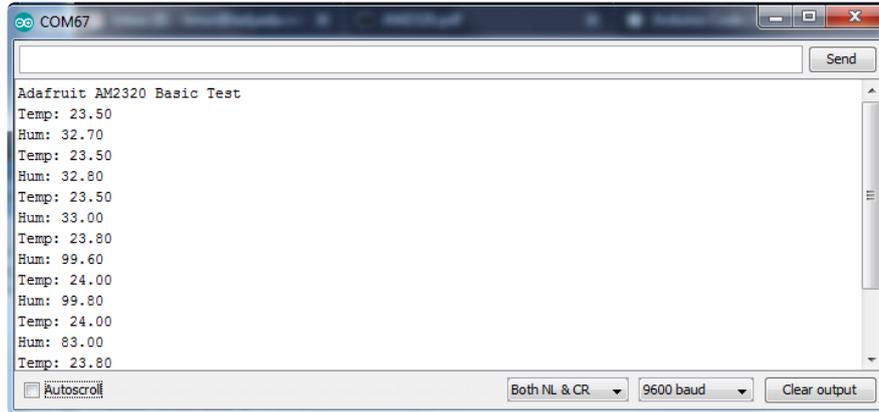
We also have a great tutorial on Arduino library installation at: <http://learn.adafruit.com/adafruit-all-about-arduino-libraries-install-use> (<https://adafru.it/aYM>)

Load Demo

Open up **File -> Examples -> Adafruit_AM2315 -> am2315test** and upload to your microcontroller wired up to the sensor



That's it! Now open up the serial terminal window at 9600 speed to begin the test.



You can try breathing on the sensor to increase the humidity.

The temperature output is in degrees Celsius.

Python & CircuitPython

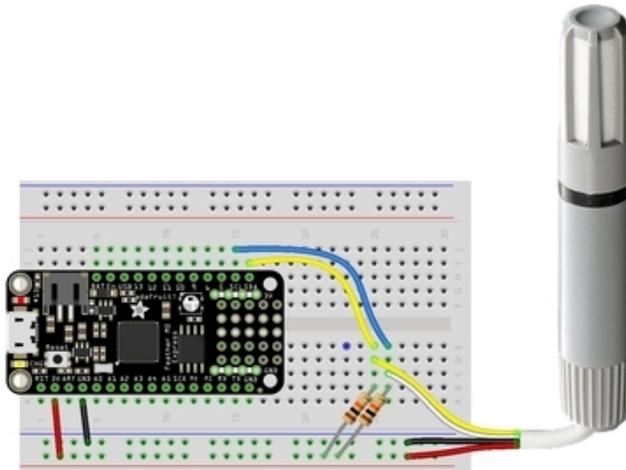
It's easy to use the AM2315 sensor with Python or CircuitPython and the [Adafruit CircuitPython AM2320](https://adafru.it/C3Z) (<https://adafru.it/C3Z>) module. That's not a typo! **The CircuitPython AM2320 library works perfectly with the AM2315 sensor.** This module allows you to easily write Python code that reads the humidity and temperature from the sensor.

You can use this sensor with any CircuitPython microcontroller board or with a computer that has GPIO and Python [thanks to Adafruit_Blinka, our CircuitPython-for-Python compatibility library](https://adafru.it/BSN) (<https://adafru.it/BSN>).

One thing to watch out for with the AM2315 is that it goes into 'sleep' mode, you may find if you are scanning the i2c bus that the device does not appear - try scanning again and it will appear the second time!

CircuitPython Microcontroller Wiring

First wire up a AM2315 to your board using an I2C connection. Here's an example of wiring a Feather M0 to the sensor with I2C, don't forget to add pull-ups for the I2C lines to 3.3V

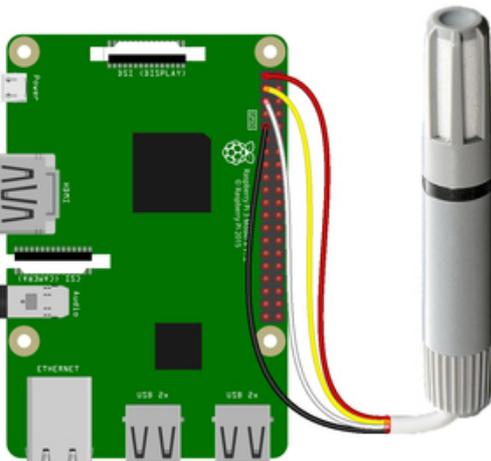


Board 3V to sensor VIN (red wire)
Board GND to sensor GND (black wire)
Board SCL to sensor SCL (white wire)
Board SDA to sensor SDA (yellow wire)

Python Computer Wiring

Since there's dozens of Linux computers/boards you can use we will show wiring for Raspberry Pi. For other platforms, [please visit the guide for CircuitPython on Linux to see whether your platform is supported \(https://adafru.it/BSN\)](https://adafru.it/BSN).

Here's the Raspberry Pi wired with I2C:



Pi 3V3 to sensor VIN (red wire)
Pi GND to sensor GND (black wire)
Pi SCL to sensor SCL (white wire)
Pi SDA to sensor SDA (yellow wire)

The Raspberry Pi has built-in pull up resistors on SDA/SCL so we don't include them here. If you are using some other computer, check if you need to add your own 2-10K pullups to 3.3V!

At this time this sensor does not appear to work well on the Raspberry Pi. We've seen mixed success, but YMMV.

CircuitPython Installation of AM2320 Library

Next you'll need to install the [Adafruit CircuitPython AM2320 \(https://adafru.it/C3Z\)](https://adafru.it/C3Z) library on your CircuitPython board. **Even though the library is called the AM2320 it also works on AM2315s!** Remember, even though you're using the AM2315 sensor, you'll be using the AM2320 library.

First make sure you are running the [latest version of Adafruit CircuitPython \(https://adafru.it/Amd\)](https://adafru.it/Amd) for your board.

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 \(https://adafru.it/zdx\)](https://adafru.it/zdx). Our introduction guide has [a great page on how to install the library bundle \(https://adafru.it/ABU\)](https://adafru.it/ABU) for both express and non-express boards.

For non-express boards like the Trinket M0 or Gemma M0, you'll need to manually install the necessary libraries from the bundle:

- `adafruit_am2320.mpy`
- `adafruit_bus_device`

You can also download the `adafruit_am2320.mpy` from [its releases page on Github \(https://adafru.it/C3-\)](https://adafru.it/C3-).

Before continuing make sure your board's lib folder or root filesystem has the `adafruit_am2320.mpy`, and `adafruit_bus_device` files and folders copied over.

We suggest using [Mu \(https://adafru.it/CEz\)](https://adafru.it/CEz) as a code editor as it includes serial access to the REPL and more.

Next [connect to the board's serial REPL \(https://adafru.it/Awz\)](https://adafru.it/Awz) so you are at the CircuitPython `>>>` prompt.

Python Installation of AM2320 Library

You'll need to install the **Adafruit_Blinka** library that provides the CircuitPython support in Python. This may also require enabling I2C on your platform and verifying you are running Python 3. [Since each platform is a little different, and Linux changes often, please visit the CircuitPython on Linux guide to get your computer ready \(https://adafru.it/BSN\)](https://adafru.it/BSN)!

Once that's done, from your command line run the following command:

- `sudo pip3 install adafruit-circuitpython-am2320`

Remember, **even though you're using the AM2315 sensor, you'll be using the AM2320 library.**

If your default Python is version 3 you may need to run 'pip' instead. Just make sure you aren't trying to use CircuitPython on Python 2.x, it isn't supported!

CircuitPython & Python Usage

To demonstrate the usage of the sensor we'll initialize it and read the humidity and temperature from the board's Python REPL.

Run the following code to import the necessary modules and initialize the I2C connection with the sensor:

```
import board
import busio
import adafruit_am2320
i2c = busio.I2C(board.SCL, board.SDA)
sensor = adafruit_am2320.AM2320(i2c)
```

Now you're ready to read values from the sensor using any of these properties:

- **relative_humidity** - The relative humidity measured by the sensor, this is a value from 0-100%.
- **temperature** - The temperature measured by the sensor, a value in degrees Celsius.

For example, to print the temperature and relative humidity:

```
print('Humidity: {0}%'.format(sensor.relative_humidity))
print('Temperature: {0}C'.format(sensor.temperature))
```

```
>>> print('Humidity: {0}%'.format(sensor.relative_humidity))
Humidity: 50.9%
>>> print('Temperature: {0}C'.format(sensor.temperature))
Temperature: 21.5C
```

If you want to take another reading, wait 2 seconds before requesting temperature or relative_humidity again. Data in the sensor is only updated every 2 seconds!

That's all there is to using the AM2315 with CircuitPython!

Below is a complete example that measures the sensor readings and prints them every two seconds. Save this as **code.py** on your board and open the REPL/Serial out to see the output.

Full Example Code

```
# SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
# SPDX-License-Identifier: MIT

import time
import board
import adafruit_am2320

# create the I2C shared bus
i2c = board.I2C() # uses board.SCL and board.SDA
# i2c = board.STEMMA_I2C() # For using the built-in STEMMA QT connector on a
microcontroller
am = adafruit_am2320.AM2320(i2c)

while True:
    print("Temperature: ", am.temperature)
    print("Humidity: ", am.relative_humidity)
    time.sleep(2)
```

Python Docs

[Python Docs \(https://adafru.it/C40\)](https://adafru.it/C40)