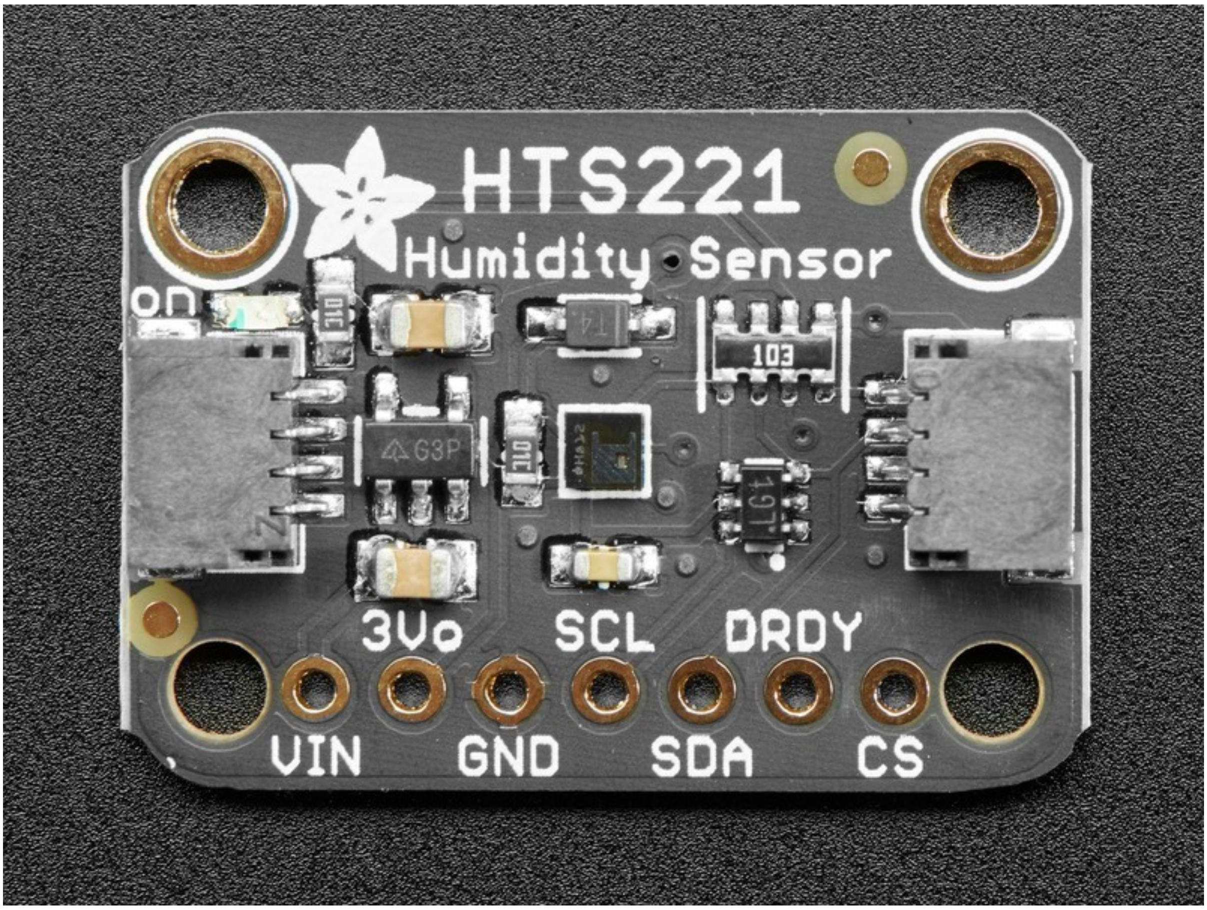
Adafruit HTS221 - Temperature & Humidity Sensor

Created by Bryan Siepert



https://learn.adafruit.com/adafruit-hts221-temperature-humidity-sensor

Last updated on 2024-03-08 03:43:20 PM EST

Table of Contents

Overview 3

Pinouts 5

- Power Pins:
- I2C Logic Pins
- SPI Logic pins:
- Other Pins

Arduino 6

- I2C Wiring
- Library Installation
- Load Example

Arduino Docs 10

Python & CircuitPython 11

- CircuitPython Microcontroller Wiring
- Python Computer Wiring
- CircuitPython Installation of HTS221 Library
- Python Installation of HTS221 Library
- CircuitPython & Python Usage

Python Docs 14

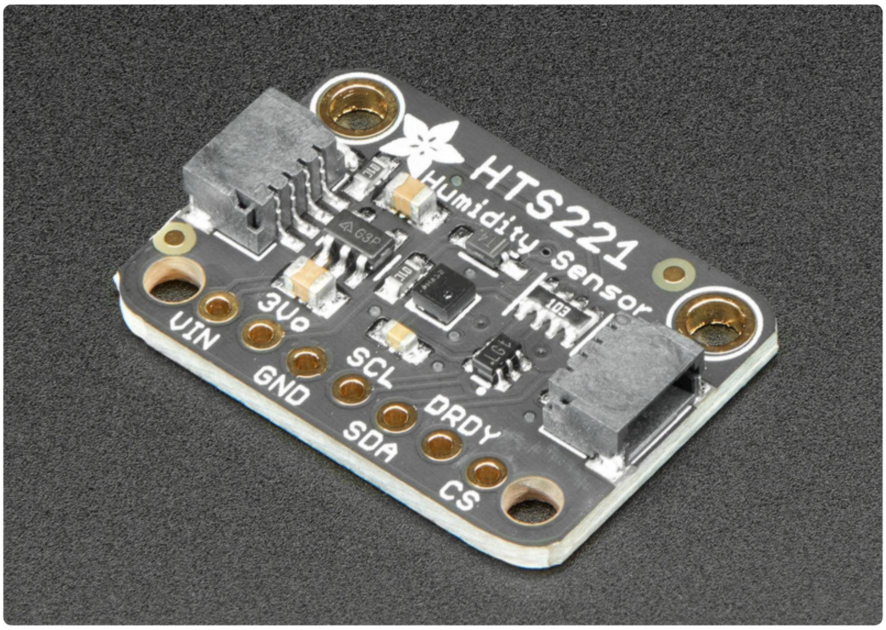
WipperSnapper 15

- What is WipperSnapper
- Wiring
- Usage

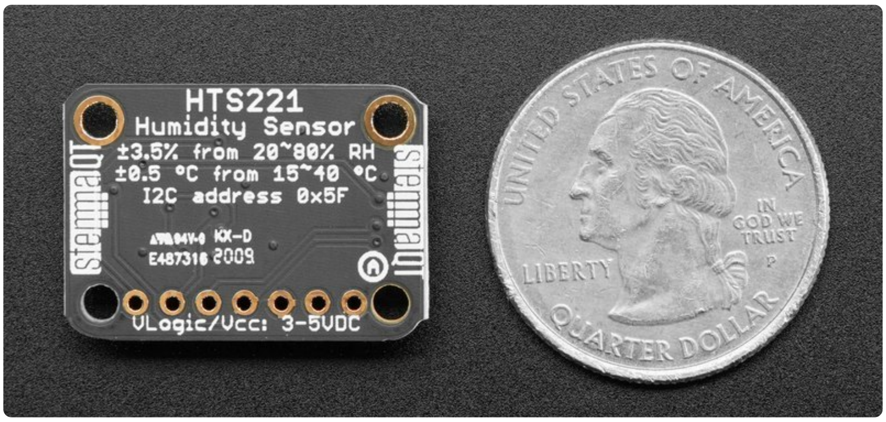
Downloads 22

- Files
- Schematic
- Fab Print

Overview

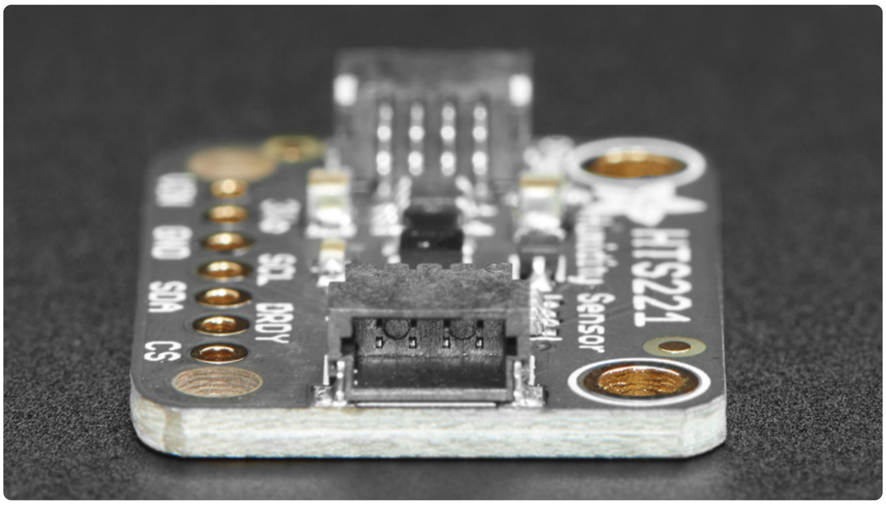


Humidity is the concentration of water vapor (water when it's a gas) in the air. Some places have more and make you feel soggy and hot like you're in a sauna. Other places don't have much, which makes it easier to get nose bleeds and static zaps. These are just a few examples of how humidity can have a noticeable impact on things that grow, like mushrooms or humans, as well as things that are sensitive to moisture, like delicate electronics, or humans.



This is where the **HTS221 Temperature and Humidity Sensor** by ST comes in. With an HTS221, you can take measurements of the relative humidity which will tell you if the environment you're measuring is bone dry or sopping wet. The HTS221 can measure relative humidity from **0%-100% rH** with a sensitivity of 0.004% and 3.5% accuracy. It can also measure temperature from **-40 to 120** degrees C, with a resolution of 0.016°C with ±0.5 °C accuracy between 15 and +40 °C.

The HTS221 is an inexpensive humidity sensor which makes it great for every day measurement applications. Want to measure the humidity of your bathroom? Sure! How about your mushroom farm? Perfect!

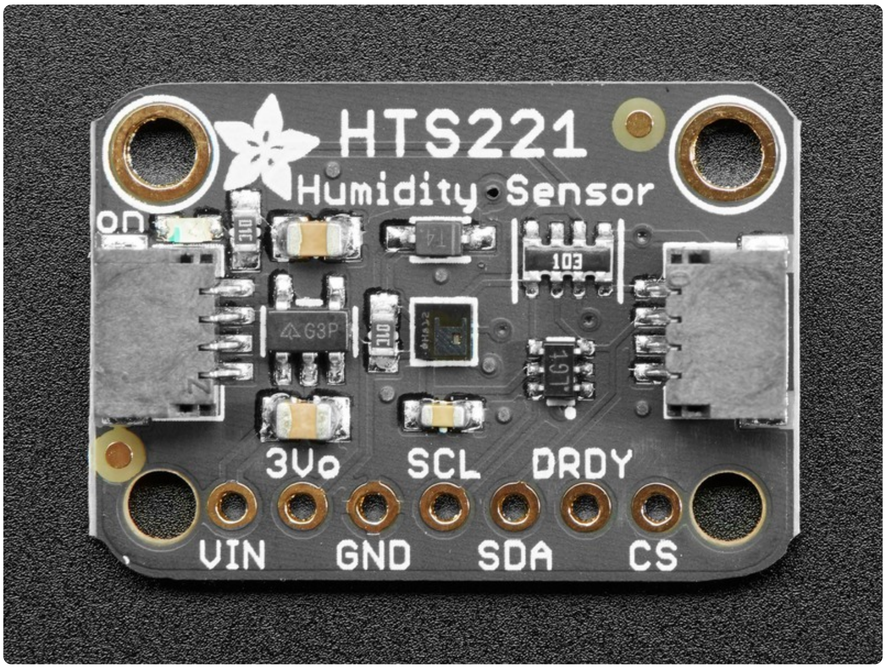


The HTS221 lets you choose between a **3-wire SPI** or the ever-popular **I2C** which only needs two wires for data. If you're an I2C fan (who isn't?), we've included our handy dandy SparkFun Qwiic (https://adafru.it/Fpw) compatible STEMMA QT **(https://adafru.it/Ft4)** connectors for the I2C bus **so you don't even need to solder!**

Since the HTS221 is a super small 2mm square, we've put it on a breakout board with level shifting and a power regulator, so it's easy to use on a breadboard with either a 5V logic level device like a Metro 328 or Arduino Uno, or with 3.3V logic level devices like a Feather M4 or Raspberry Pi of your choice.

Lastly, it wouldn't be an Adafruit breakout if it didn't come with **libraries for Arduino and CircuitPython or Python**. We've also written **example code** and instructions on the pages that follow (https://adafru.it/JEd), so you'll be ready to tell all your friends the approximate relative humidity level and temperature in only a few minutes.

Pinouts



Power Pins:

- **Vin** - this is the power pin. Since the sensor chip uses 3 VDC, we have included a voltage regulator on board that will take 3-5VDC and safely convert it down. To power the board, give it the same power as the logic level of your microcontroller - e.g. for a 5V micro like Arduino, use 5V
- **3Vo** - this is the 3.3V output from the voltage regulator, you can grab up to 100mA from this if you like
- **GND** - common ground for power and logic

I2C Logic Pins

- **SCL** - I2C clock pin, connect to your microcontroller's I2C clock line. This pin is level shifted so you can use 3-5V logic, and there's a **10K pullup** on this pin.
- **SDA** - I2C data pin, connect to your microcontroller's I2C data line. This pin is level shifted so you can use 3-5V logic, and there's a **10K pullup** on this pin.
- STEMMA QT **(https://adafru.it/Ft4)** - These connectors allow you to make I2C connections to dev boards with **STEMMA QT** connectors or to other things with various associated accessories (https://adafru.it/Ft6)

SPI Logic pins:

Because the HTS221 is only capable of 3-pin SPI, take care when reading the pin descriptions below and consult the wiring diagram on the Arduino page

All pins going into the breakout have level shifting circuitry to make them 3-5V logic level safe. Use whatever logic level is on **Vin!**

- **SCL** - This is the **S**PI **C**lock pin, its an input to the chip
- **SDA** - For 3-pin SPI, this pin does double duty and is connected to your device's **MISO and MOSI** pins:
  - **MISO -** Connect directly to the HTS221's **SDA** pin
  - **MOSI -** Connect to **SDA through a 1K resistor**
- **CS** - this is the **C**hip **S**elect pin, drop it low to start an SPI transaction. Its an input to the chip

If you want to connect multiple HTS221's to one microcontroller, have them share the SCL and SDA pins. Then assign each one a unique CS pin.

Other Pins

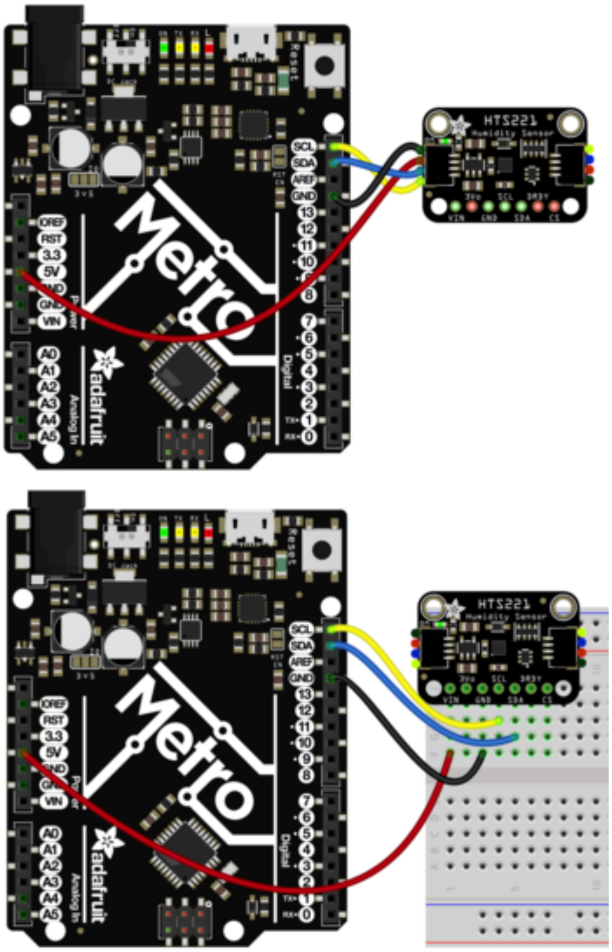
- **DRDY** - Data ready pin. The HTS221 can be configured to change the state of this pin to signify that new pressure or temperature measurements are available

Arduino

I2C Wiring

Use this wiring if you want to connect via I2C interface

The HTS221's I2C address is **0x5F**.



Connect **board VIN (red wire)** to **Arduino 5V** if you are running a **5V** board Arduino (Uno, etc.). If your board is **3V,** connect to that instead.

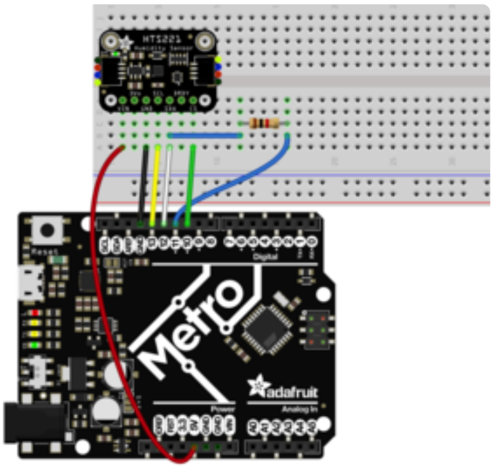
Connect **board GND (black wire)** to **Arduino GND**

Connect **board SCL (yellow wire)** to **Arduino SCL**

Connect **board SDA (blue wire)** to **Arduino SDA**

The final results should resemble the illustration above, showing an Adafruit Metro development board.

Because the HTS221 is only capable of 3-pin SPI, take care when reading the wiring instructions below and consult the wiring diagram to verify your connections

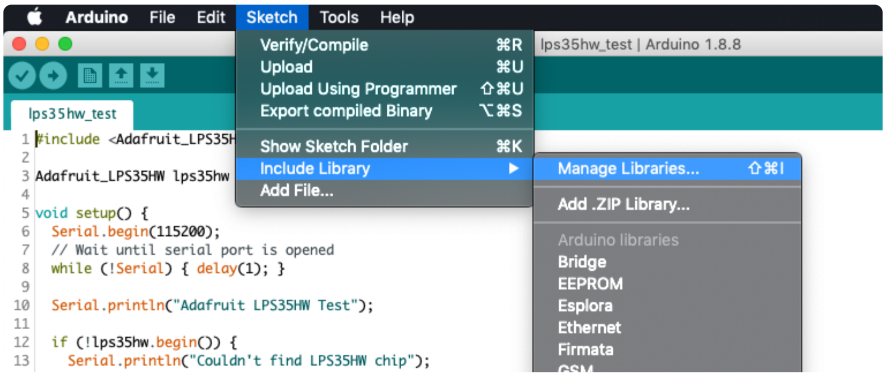


Connect **Vin** to the power supply, 3V or 5V is fine. Use the same voltage that the microcontroller logic is based off of
Connect **GND** to common power/data ground
Connect the **SCL** pin to **Digital #13** but any pin can be used later
Connect the **SDA** pin to **Digital #12** but any pin can be used later
Make a second connection to the **SDA** pin, **through a 1K resistor** from **Digital #11.** Any pin can be used later but it must be connected to SDA **through a 1K resistor**
Connect the **CS** pin **Digital #10** but any pin can be used later

Later on, once we get it working, we can adjust the library to use hardware SPI if you desire, or change the pins to others.

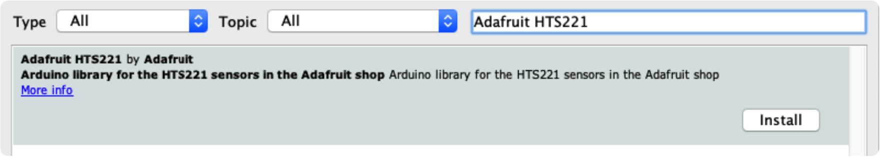
Library Installation

You can install the **Adafruit HTS221 Library** for Arduino using the Library Manager in the Arduino IDE.



```
#include <Adafruit_LPS35H

Adafruit_LPS35HW lps35hw

void setup() {
  Serial.begin(115200);
  // Wait until serial port is opened
  while (!Serial) { delay(1); }

  Serial.println("Adafruit LPS35HW Test");

  if (!lps35hw.begin()) {
    Serial.println("Couldn't find LPS35HW chip");
```

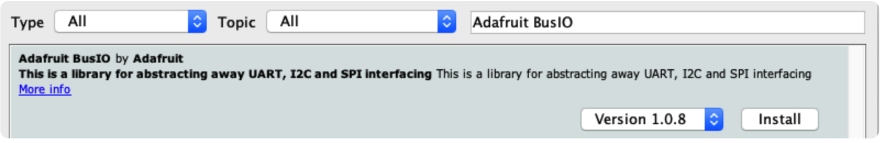
Click the **Manage Libraries ...** menu item, search for **Adafruit HTS221** and select the **Adafruit HTS221** library:



Type All Topic All Adafruit HTS221

Adafruit HTS221 by **Adafruit**
Arduino library for the HTS221 sensors in the Adafruit shop Arduino library for the HTS221 sensors in the Adafruit shop
More info

Then follow the same process for the **Adafruit BusIO** library.

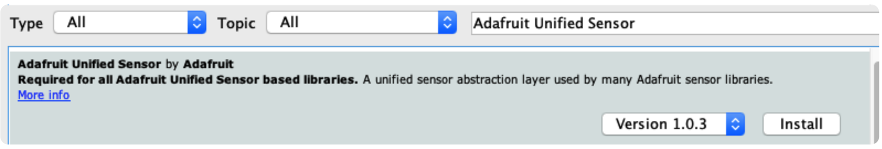


Type All Topic All Adafruit BusIO

Adafruit BusIO by **Adafruit**
This is a library for abstracting away UART, I2C and SPI interfacing This is a library for abstracting away UART, I2C and SPI interfacing
More info

Version 1.0.8

Finally follow the same process for the **Adafruit Unified Sensor** library:



Type All Topic All Adafruit Unified Sensor

Adafruit Unified Sensor by **Adafruit**
Required for all Adafruit Unified Sensor based libraries. A unified sensor abstraction layer used by many Adafruit sensor libraries.
More info

Version 1.0.3

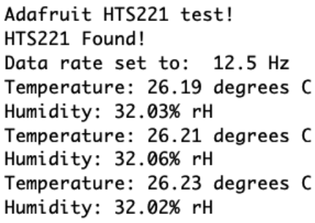
Load Example

Open up **File** -> **Examples** -> **Adafruit HTS221** -> **adafruit_hts221_test** and upload to your Arduino wired up to the sensor.

Depending on whether you are using I2C or SPI, change the pin names and comment or uncomment the following lines.

```
if (!hts.begin_I2C()) {
//  if (!hts.begin_SPI(HTS_CS)) {
//  if (!hts.begin_SPI(HTS_CS, HTS_SCK, HTS_MISO, HTS_MOSI)) {
```

Once you upload the code and open the Serial Monitor (**Tools->Serial Monitor**) at **115200** baud, you will see the current configuration printed, followed by the humidity and temperature measurements. You should see something similar to this:



```
Adafruit HTS221 test!
HTS221 Found!
Data rate set to:  12.5 Hz
Temperature: 26.19 degrees C
Humidity: 32.03% rH
Temperature: 26.21 degrees C
Humidity: 32.06% rH
Temperature: 26.23 degrees C
Humidity: 32.02% rH
```

The moisture from your breath should make the readings change, so give it a try by breathing on the sensor (the tiny black square in the center of the breakout) and seeing what happens!

```
// Basic demo for reading Humidity and Temperature
#include <Wire.h>
#include <Adafruit_HTS221.h>
#include <Adafruit_Sensor.h>

// For SPI mode, we need a CS pin
#define HTS_CS 10
// For software-SPI mode we need SCK/MOSI/MISO pins
#define HTS_SCK 13
#define HTS_MISO 12
#define HTS_MOSI 11

Adafruit_HTS221 hts;
void setup(void) {
  Serial.begin(115200);
  while (!Serial) delay(10);     // will pause Zero, Leonardo, etc until serial
console opens

  Serial.println("Adafruit HTS221 test!");

  // Try to initialize!
  if (!hts.begin_I2C()) {
//  if (!hts.begin_SPI(HTS_CS)) {
//  if (!hts.begin_SPI(HTS_CS, HTS_SCK, HTS_MISO, HTS_MOSI)) {
    Serial.println("Failed to find HTS221 chip");
    while (1) { delay(10); }
  }
  Serial.println("HTS221 Found!");

//  hts.setDataRate(HTS221_RATE_1_HZ);
  Serial.print("Data rate set to: ");
  switch (hts.getDataRate()) {
   case HTS221_RATE_ONE_SHOT: Serial.println("One Shot"); break;
   case HTS221_RATE_1_HZ: Serial.println("1 Hz"); break;
   case HTS221_RATE_7_HZ: Serial.println("7 Hz"); break;
   case HTS221_RATE_12_5_HZ: Serial.println("12.5 Hz"); break;
  }

}

void loop() {

  sensors_event_t temp;
  sensors_event_t humidity;
  hts.getEvent(&humidity, &temp);// populate temp and humidity objects with fresh
data
  Serial.print("Temperature: "); Serial.print(temp.temperature); Serial.println("
degrees C");
  Serial.print("Humidity: "); Serial.print(humidity.relative_humidity);
Serial.println("% rH");

  delay(500);
}
```

Arduino Docs

Arduino Docs (https://adafru.it/Jcs)

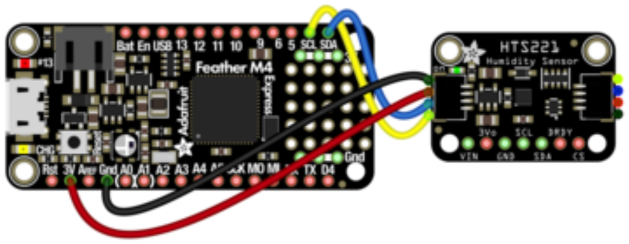
Python & CircuitPython

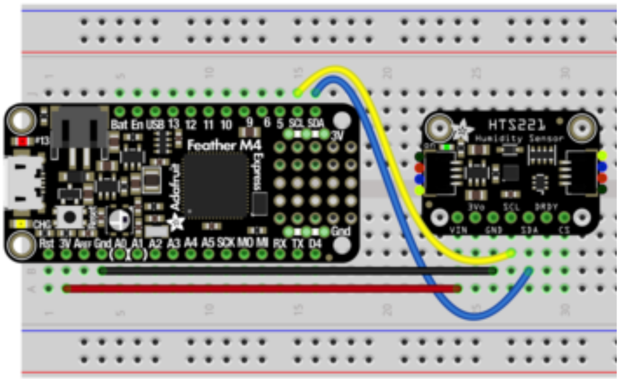
It's easy to use the HTS221 sensor with Python and CircuitPython, and the Adafruit CircuitPython HTS221 (https://adafru.it/JDV) module. This module allows you to easily write Python code that reads humidity and temperature measurements.

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).

CircuitPython Microcontroller Wiring

First wire up a HTS221 to your board for an I2C connection, exactly as shown below. Here's an example of wiring a Feather M4 to the sensor with I2C:



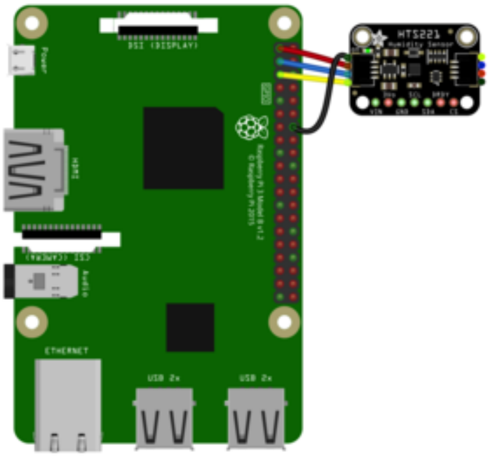


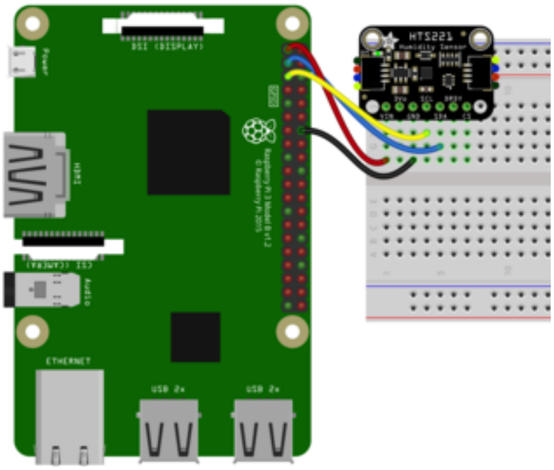
Board 3V to **sensor VIN (red wire)**
Board GND to **sensor GND (black wire)**
Board SCL to **sensor SCL (yellow wire)**
Board SDA to **sensor SDA (blue wire)**

Python Computer Wiring

Since there are 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).

Here's the Raspberry Pi wired with I2C:





Pi 3V to **sensor VCC (red wire)**
Pi GND to **sensor GND (black wire)**
Pi SCL to **sensor SCL (yellow wire)**
Pi SDA to **sensor SDA (blue wire)**

CircuitPython Installation of HTS221 Library

You'll need to install the Adafruit CircuitPython HTS221 (https://adafru.it/JDV) library on your CircuitPython board.

First make sure you are running the latest version of Adafruit CircuitPython (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/ENC). Our CircuitPython starter guide has a great page on how to install the library bundle (https://adafru.it/ABU).

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

- **adafruit_hts221.mpy**
- **adafruit_bus_device**
- **adafruit_register**

Before continuing make sure your board's **lib** folder or root filesystem has the **adafruit_hts221.mpy, adafruit_bus_device**, and **adafruit_register** files and folders copied over.

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

Python Installation of HTS221 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)!

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

- `sudo pip3 install adafruit-circuitpython-hts221`

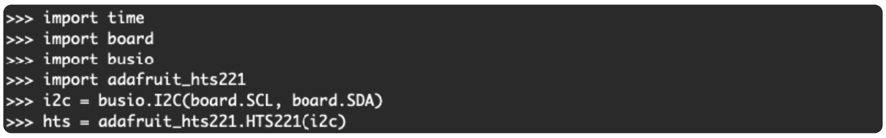
If your default Python is version 3 you may need to run 'pip' instead. Just make sure you aren't trying to use CircuitPython on Python 2.x, it isn't supported!

CircuitPython & Python Usage

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

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

```
import time
import board
import busio
import adafruit_hts221

i2c = busio.I2C(board.SCL, board.SDA)
hts = adafruit_hts221.HTS221(i2c)
```

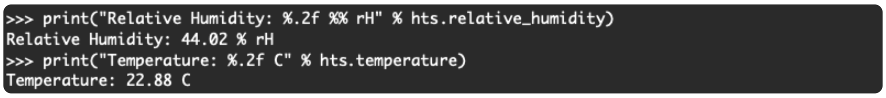


```
>>> import time
>>> import board
>>> import busio
>>> import adafruit_hts221
>>> i2c = busio.I2C(board.SCL, board.SDA)
>>> hts = adafruit_hts221.HTS221(i2c)
```

Now you're ready to read values from the sensor using 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.

```
print("Relative Humidity: %.2f %% rH" % hts.relative_humidity)
print("Temperature: %.2f C" % hts.temperature)
```



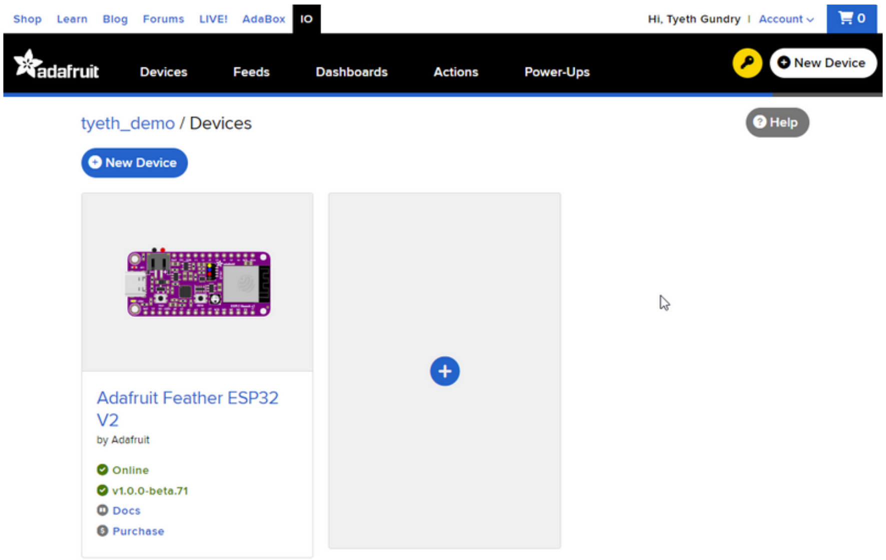
```
>>> print("Relative Humidity: %.2f %% rH" % hts.relative_humidity)
Relative Humidity: 44.02 % rH
>>> print("Temperature: %.2f C" % hts.temperature)
Temperature: 22.88 C
```

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

import time
import board
import adafruit_hts221

i2c = board.I2C()  # uses board.SCL and board.SDA
# i2c = board.STEMMA_I2C()  # For using the built-in STEMMA QT connector on a
microcontroller
hts = adafruit_hts221.HTS221(i2c)

data_rate = adafruit_hts221.Rate.label[hts.data_rate]
print("Using data rate of: {:.1f} Hz".format(data_rate))
print("")

while True:
    print("Relative Humidity: {:.2f} % rH".format(hts.relative_humidity))
    print("Temperature: {:.2f} C".format(hts.temperature))
    print("")
    time.sleep(1)
```

Python Docs

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

WipperSnapper



What is WipperSnapper

WipperSnapper is a firmware designed to turn any WiFi-capable board into an Internet-of-Things device without programming a single line of code. WipperSnapper connects to Adafruit IO (https://adafru.it/fsU), a web platform designed (by Adafruit! (https://adafru.it/Bo5)) to display, respond, and interact with your project's data.

Simply load the WipperSnapper firmware onto your board, add credentials, and plug it into power. Your board will automatically register itself with your Adafruit IO account.

From there, you can add components to your board such as buttons, switches, potentiometers, sensors, and more! Components are dynamically added to hardware, so you can immediately start interacting, logging, and streaming the data your projects produce without writing code.

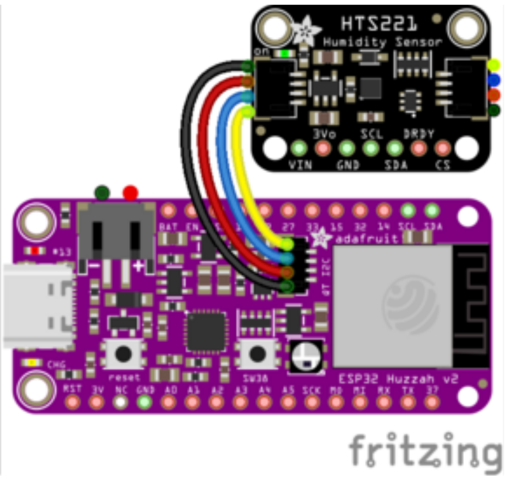
If you've never used WipperSnapper, click below to read through the quick start guide before continuing.

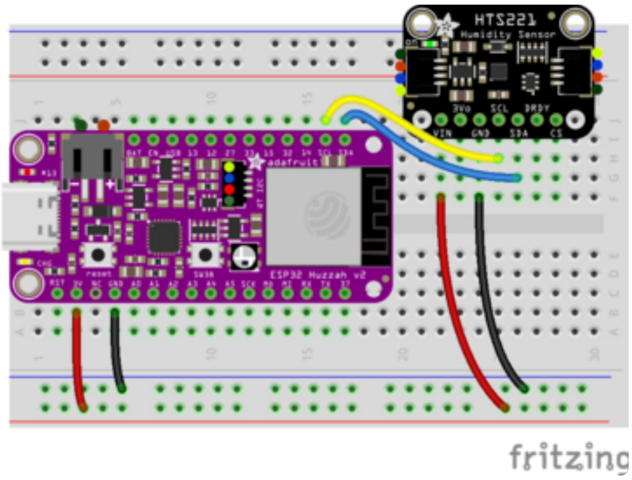
Quickstart: Adafruit IO WipperSnapper

https://adafru.it/Vfd

Wiring

First, wire up an HTS221 to your board exactly as follows. Here is an example of the HTS221 wired to an Adafruit ESP32 Feather V2 (http://adafru.it/5400) using I2C with a STEMMA QT cable (no soldering required) (http://adafru.it/4210)



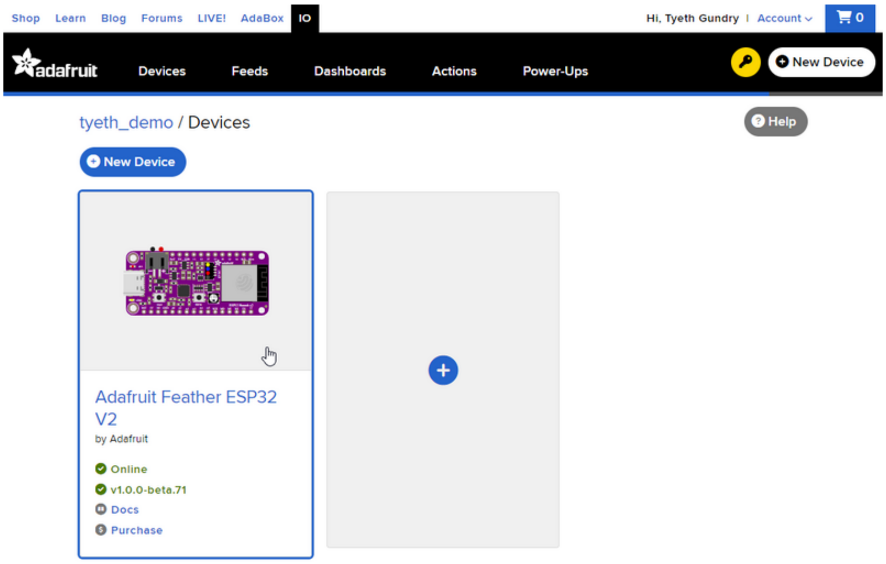


Board 3V to **sensor VIN (red wire on STEMMA QT)**
Board GND to **sensor GND (black wire on STEMMA QT)**
Board SCL to **sensor SCL (yellow wire on STEMMA QT)**
Board SDA to **sensor SDA (blue wire on STEMMA QT)**

Usage

Connect your board to Adafruit IO Wippersnapper and navigate to the WipperSnapper board list (https://adafru.it/TAu).

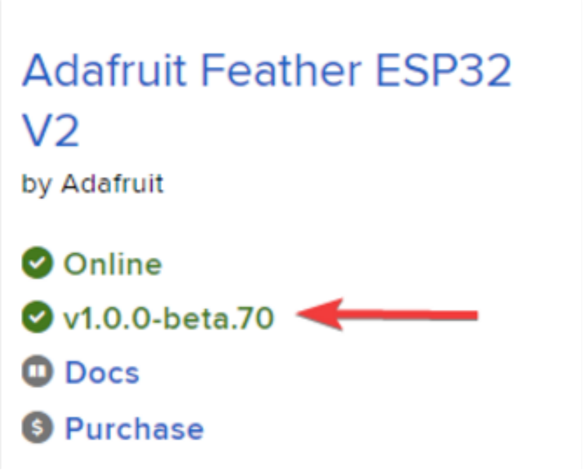
On this page, **select the WipperSnapper board you're using** to be brought to the board's interface page.



Adafruit Feather ESP32 V2
by Adafruit

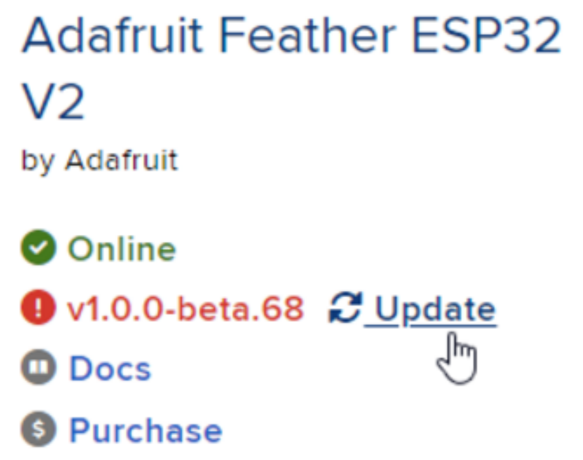
✓ Online
✓ v1.0.0-beta.71
Docs
Purchase

If you do not see your board listed here - you need to connect your board to Adafruit IO (https://adafru.it/Vfd) first.



Adafruit Feather ESP32 V2
by Adafruit

✓ Online
✓ v1.0.0-beta.70
Docs
Purchase



Adafruit Feather ESP32 V2
by Adafruit

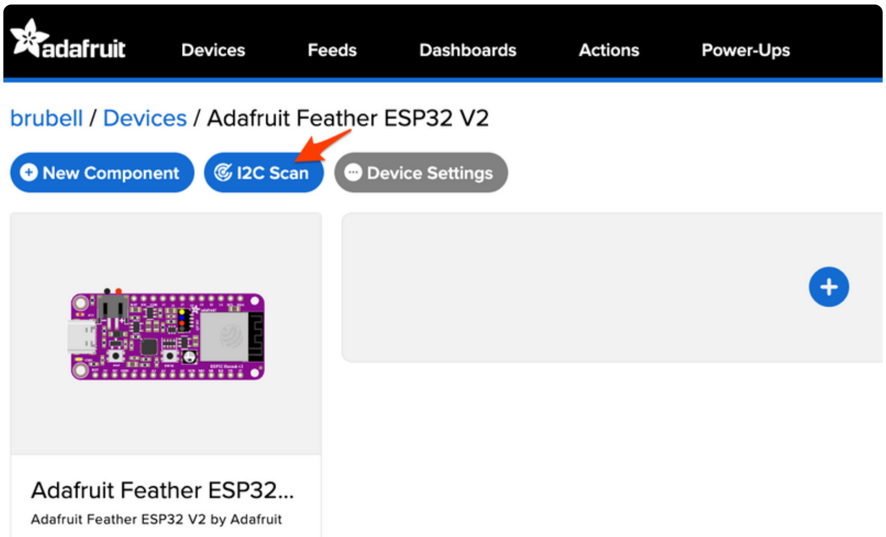
✓ Online
! v1.0.0-beta.68 Update
Docs
Purchase

On the device page, quickly **check that you're running the latest version of the WipperSnapper firmware**.

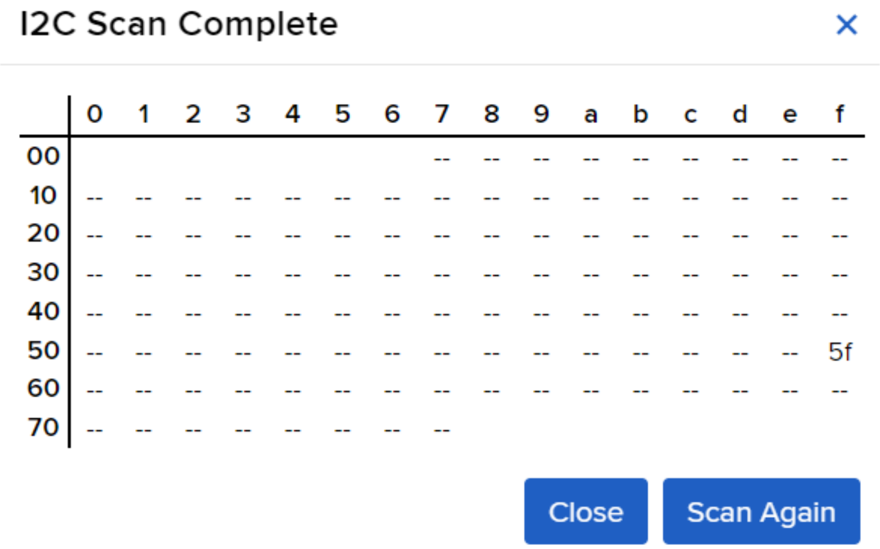
The device tile on the left indicates the version number of the firmware running on the connected board.

If the firmware version is green with a checkmark - continue with this guide.
If the firmware version is red with an exclamation mark "!" - update to the latest WipperSnapper firmware (https://adafru.it/Vfd) on your board before continuing.

Next, make sure the sensor is plugged into your board and click the **I2C Scan** button.


adafruit
Devices
Feeds
Dashboards
Actions
Power-Ups
brubell / Devices / Adafruit Feather ESP32 V2
New Component
I2C Scan
Device Settings
+
Adafruit Feather ESP32...
Adafruit Feather ESP32 V2 by Adafruit

You should see the HTS221's default I2C address of `0x5F` pop-up in the I2C scan list.



I2C Scan Complete

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
00								--	--	--	--	--	--	--	--	--
10	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
20	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
30	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
40	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
50	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	5f
60	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
70	--	--	--	--	--	--	--	--								

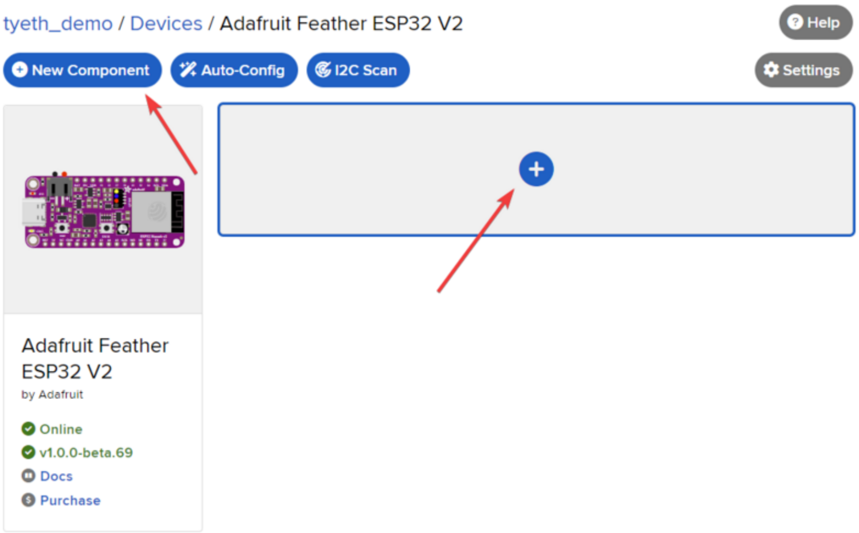
I don't see the sensor's I2C address listed!

First, double-check the connection and/or wiring between the sensor and the board.

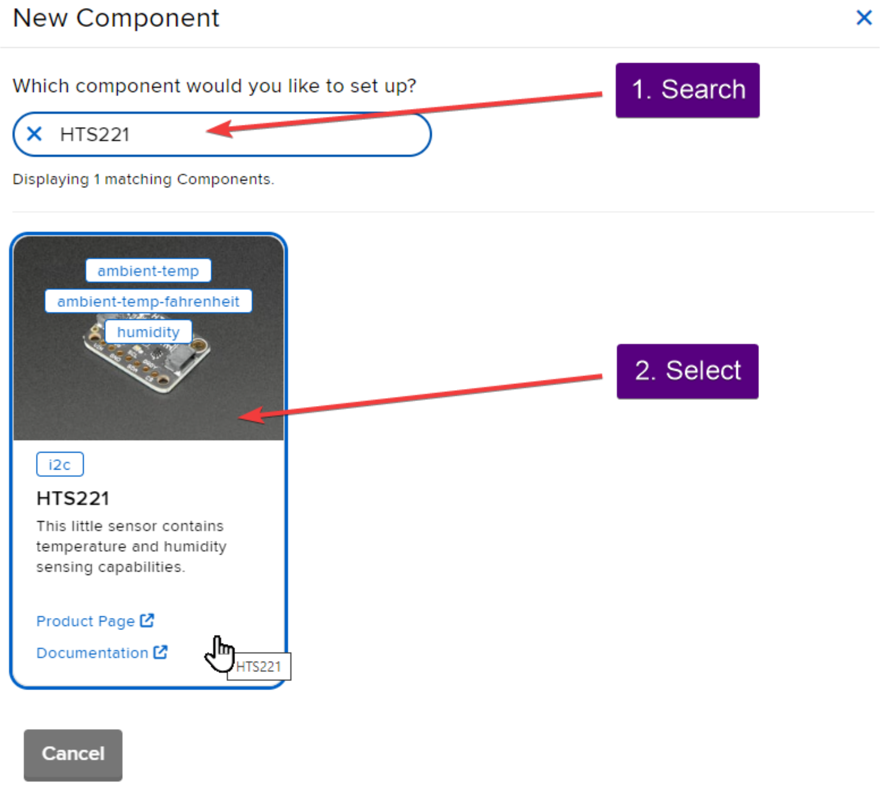
Then, reset the board and let it re-connect to Adafruit IO WipperSnapper.

With the sensor detected in an I2C scan, you're ready to add the sensor to your board.

Click the New Component button or the + button to bring up the component picker.



Adafruit IO supports a large amount of components. To quickly find your sensor, type `HTS221` into the search bar, then select the **HTS221** component.


New Component

Which component would you like to set up?

HTS221

Displaying 1 matching Components.

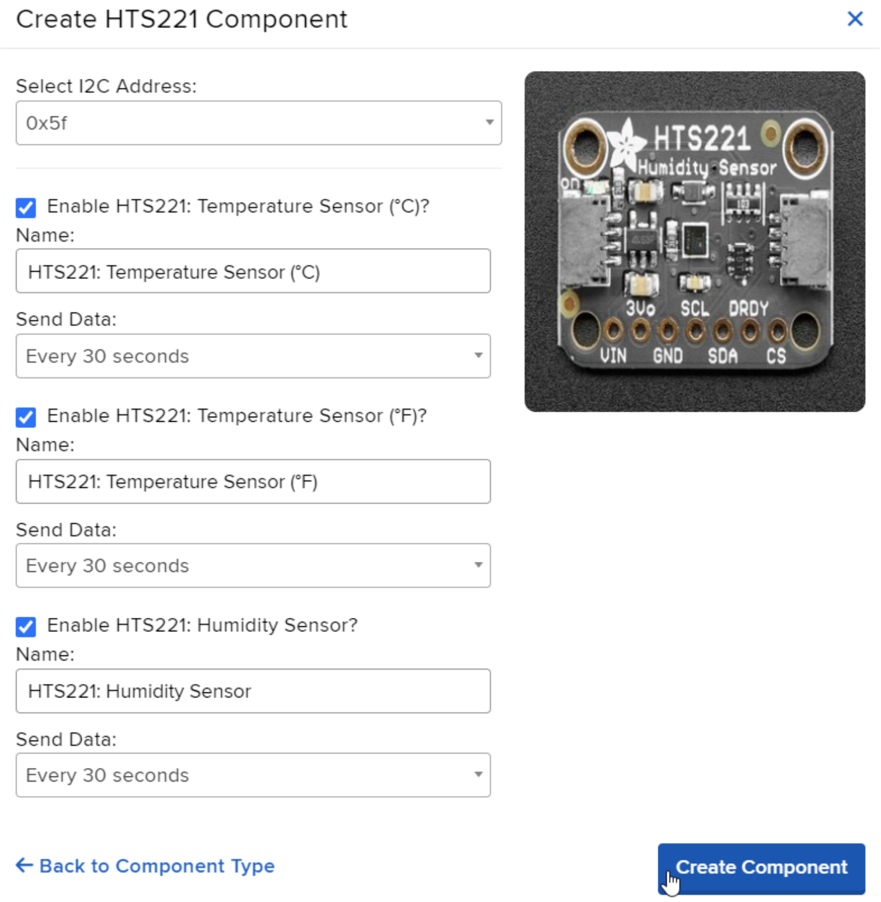
HTS221

This little sensor contains temperature and humidity sensing capabilities.

On the component configuration page, the **HTS221**'s sensor address should be listed along with the sensor's settings.

The **Send Every** option is specific to each sensor's measurements. This option will tell the Feather how often it should read from the **HTS221** sensor and send the data to Adafruit IO. Measurements can range from every 30 seconds to every 24 hours.

For this example, set the **Send Every** interval to every 30 seconds.



Create HTS221 Component

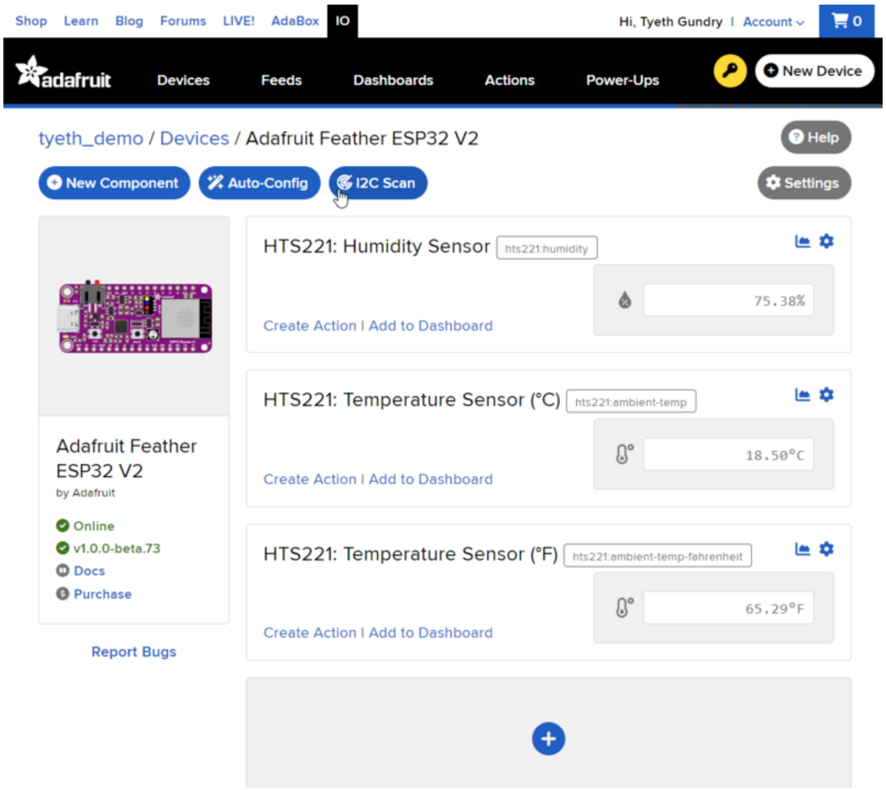
Select I2C Address: 0x5f

☑ Enable HTS221: Temperature Sensor (°C)?
Name: HTS221: Temperature Sensor (°C)
Send Data: Every 30 seconds

☑ Enable HTS221: Temperature Sensor (°F)?
Name: HTS221: Temperature Sensor (°F)
Send Data: Every 30 seconds

☑ Enable HTS221: Humidity Sensor?
Name: HTS221: Humidity Sensor
Send Data: Every 30 seconds

Your device interface should now show the sensor components you created. After the interval you configured elapses, WipperSnapper will automatically read values from the sensor(s) and send them to Adafruit IO.



tyeth_demo / Devices / Adafruit Feather ESP32 V2

HTS221: Humidity Sensor hts221:humidity — 75.38%

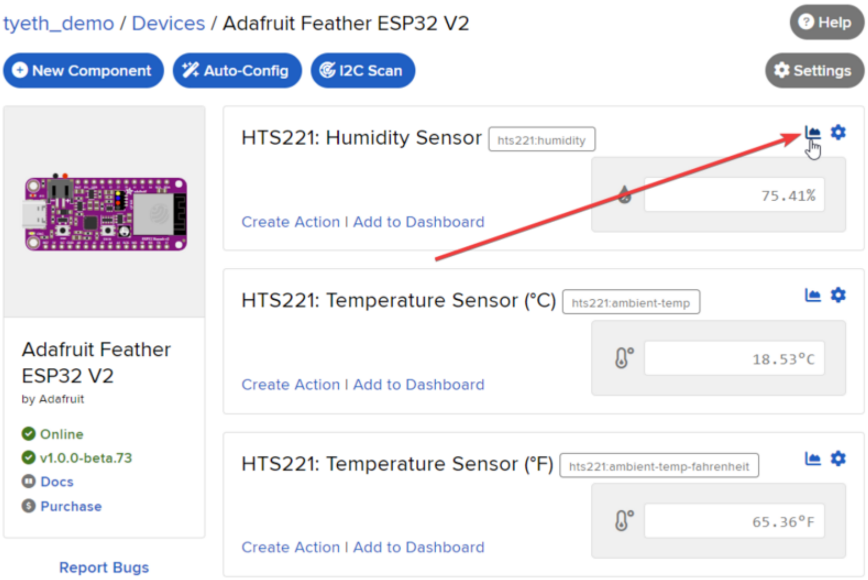
HTS221: Temperature Sensor (°C) hts221:ambient-temp — 18.50°C

HTS221: Temperature Sensor (°F) hts221:ambient-temp-fahrenheit — 65.29°F

Adafruit Feather ESP32 V2
by Adafruit
Online
v1.0.0-beta.73

Shop Learn Blog Forums LIVE! AdaBox IO
Hi, Tyeth Gundry | Account
0
adafruit
Devices
Feeds
Dashboards
Actions
Power-Ups
New Device
Help
New Component
Auto-Config
I2C Scan
Settings
Create Action | Add to Dashboard
Create Action | Add to Dashboard
Create Action | Add to Dashboard
Docs
Purchase
Report Bugs

To view the data that has been logged from the sensor, click on the graph next to the sensor name.



tyeth_demo / Devices / Adafruit Feather ESP32 V2

Help

New Component | Auto-Config | I2C Scan | Settings

Adafruit Feather ESP32 V2
by Adafruit

- Online
- v1.0.0-beta.73
- Docs
- Purchase

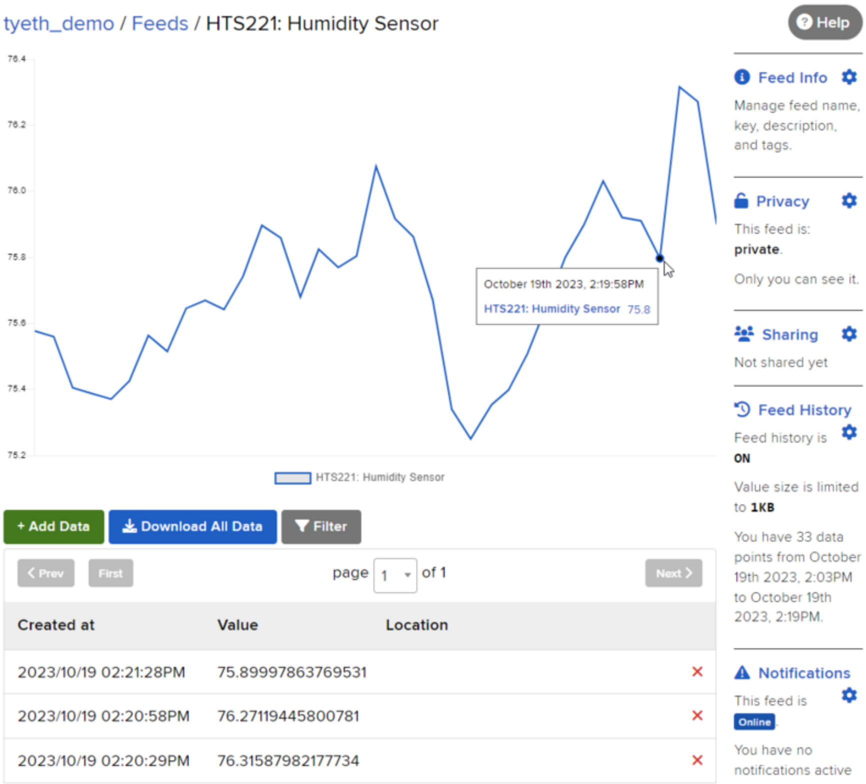
Report Bugs

HTS221: Humidity Sensor hts221:humidity
75.41%
Create Action | Add to Dashboard

HTS221: Temperature Sensor (°C) hts221:ambient-temp
18.53°C
Create Action | Add to Dashboard

HTS221: Temperature Sensor (°F) hts221:ambient-temp-fahrenheit
65.36°F
Create Action | Add to Dashboard

Here you can see the feed history and edit things about the feed such as the name, privacy, webhooks associated with the feed and more. If you want to learn more about how feeds work, check out this page (https://adafru.it/10aZ).



tyeth_demo / Feeds / HTS221: Humidity Sensor

Help

+ Add Data | Download All Data | Filter

< Prev | First | page 1 of 1 | Next >

Created at	Value	Location	
2023/10/19 02:21:28PM	75.89997863769531		×
2023/10/19 02:20:58PM	76.27119445800781		×
2023/10/19 02:20:29PM	76.31587982177734		×

Feed Info
Manage feed name, key, description, and tags.

Privacy
This feed is: **private**.
Only you can see it.

Sharing
Not shared yet

Feed History
Feed history is **ON**
Value size is limited to **1KB**
You have 33 data points from October 19th 2023, 2:03PM to October 19th 2023, 2:19PM.

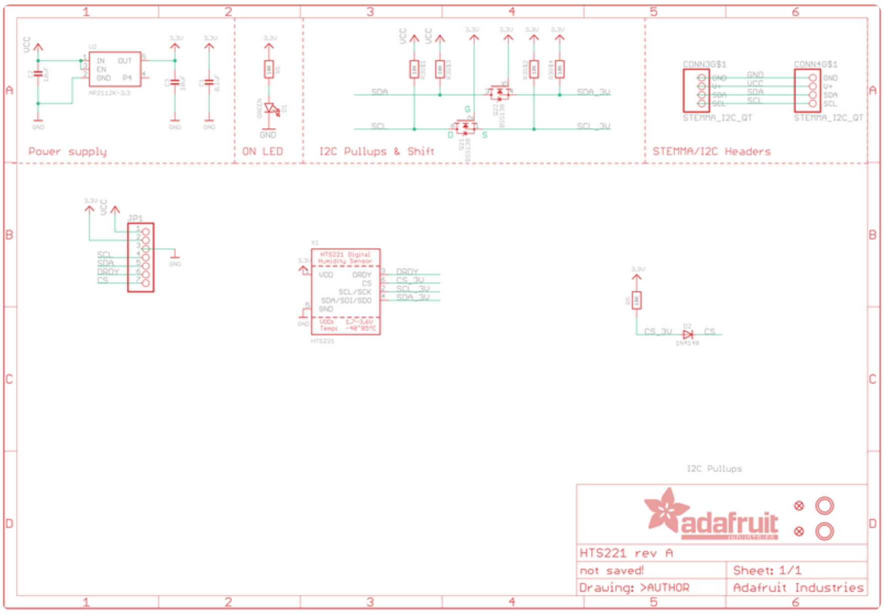
Notifications
This feed is Online.
You have no notifications active

Downloads

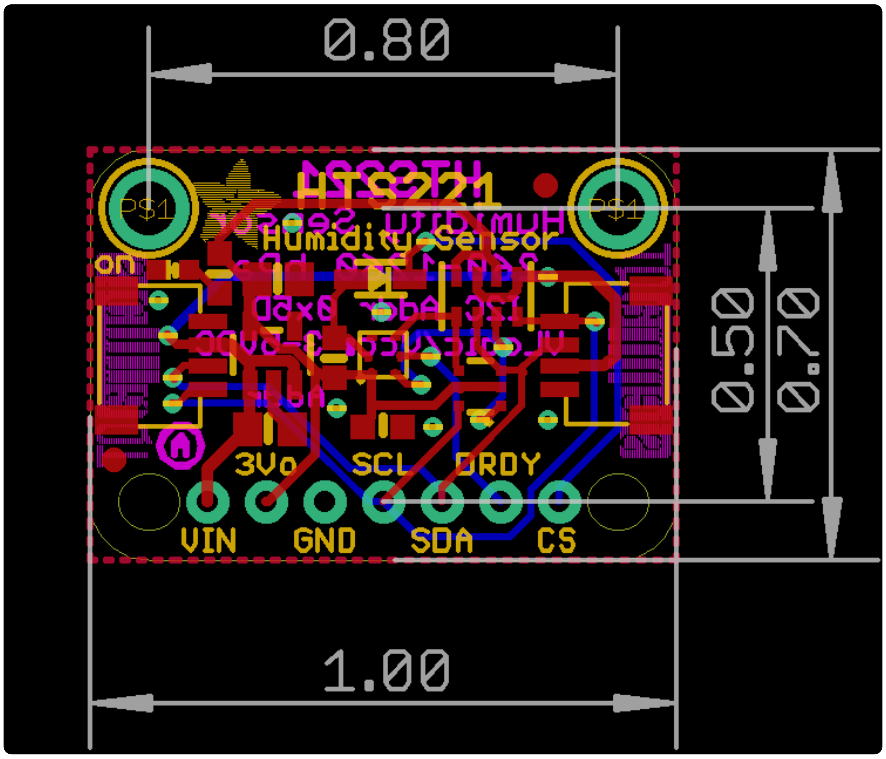
Files

- HTS221 Datashet (https://adafru.it/JDW)
- EagleCAD files on GitHub (https://adafru.it/JDX)
- Fritzing object in Adafruit Fritzing Library (https://adafru.it/JDY)

Schematic



Fab Print


0.80
HTS221
Humidity Sensor
on
3Vo
SCL
DRDY
VIN
GND
SDA
CS
0.50
0.70
1.00