
AdafruitHTU21D Library Documentation

Release 1.0

ktown

Jun 07, 2021

Contents

1	Dependencies	3
2	Installing from PyPI	5
3	Usage Example	7
4	Contributing	9
5	Documentation	11
6	Table of Contents	13
6.1	Simple test	13
6.2	adafruit_htu21d	13
6.2.1	Implementation Notes	14
7	Indices and tables	15
	Python Module Index	17
	Index	19

This driver enables you to use the Adafruit HTU21D-F temperature and humidity breakout with CircuitPython.

CHAPTER 1

Dependencies

This driver depends on:

- [Adafruit CircuitPython](#)
- [Bus Device](#)

Please ensure all dependencies are available on the CircuitPython filesystem. This is easily achieved by downloading the [Adafruit library and driver bundle](#).

Installing from PyPI

On supported GNU/Linux systems like the Raspberry Pi, you can install the driver locally [from PyPI](#). To install for current user:

```
pip3 install adafruit-circuitpython-htu21d
```

To install system-wide (this may be required in some cases):

```
sudo pip3 install adafruit-circuitpython-htu21d
```

To install in a virtual environment in your current project:

```
mkdir project-name && cd project-name  
python3 -m venv .env  
source .env/bin/activate  
pip3 install adafruit-circuitpython-htu21d
```


CHAPTER 3

Usage Example

```
import time
import board
from adafruit_htu21d import HTU21D

# Create sensor object, communicating over the board's default I2C bus
i2c = board.I2C() # uses board.SCL and board.SDA
sensor = HTU21D(i2c)

while True:
    print("\nTemperature: %0.1f C" % sensor.temperature)
    print("Humidity: %0.1f %% " % sensor.relative_humidity)
    time.sleep(2)
```


CHAPTER 4

Contributing

Contributions are welcome! Please read our [Code of Conduct](#) before contributing to help this project stay welcoming.

CHAPTER 5

Documentation

For information on building library documentation, please check out [this guide](#).

6.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/htu21d_simpletest.py

```
1  # SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
2  # SPDX-License-Identifier: MIT
3
4  import time
5  import board
6  from adafruit_htu21d import HTU21D
7
8  # Create sensor object, communicating over the board's default I2C bus
9  i2c = board.I2C() # uses board.SCL and board.SDA
10 sensor = HTU21D(i2c)
11
12
13 while True:
14     print("\nTemperature: %0.1f C" % sensor.temperature)
15     print("Humidity: %0.1f %% " % sensor.relative_humidity)
16     time.sleep(2)
```

6.2 adafruit_htu21d

This is a breakout for the Adafruit HTU21D-F Temperature & Humidity Sensor Breakout Board

- Author(s): ktown

6.2.1 Implementation Notes

Hardware:

- Adafruit HTU21D-F Temperature & Humidity Sensor Breakout Board (Product ID: 1899)

Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: <https://circuitpython.org/downloads>
- Adafruit's Bus Device library: https://github.com/adafruit/Adafruit_CircuitPython_BusDevice

class `adafruit_htu21d.HTU21D` (*i2c_bus*, *address=64*)

A driver for the HTU21D-F temperature and humidity sensor. ;param *i2c_bus*: The I2C bus the device is connected to ;param *int address*: (optional) The I2C address of the device. Defaults to 0x40

Quickstart: Importing and using the HTU21D-F

Here is an example of using the *HTU21D* class. First you will need to import the libraries to use the sensor

```
import board
from adafruit_htu21d import HTU21D
```

Once this is done you can define your `board.I2C` object and define your sensor object

```
i2c = board.I2C() # uses board.SCL and board.SDA
sensor = HTU21D(i2c)
```

Now you have access to the *temperature* and *relative_humidity* attributes

```
temperature = sensor.temperature
relative_humidity = sensor.relative_humidity
```

measurement (*what*)

Starts a measurement. Starts a measurement of either HUMIDITY or TEMPERATURE depending on the *what* argument. Returns immediately, and the result of the measurement can be retrieved with the *temperature* and *relative_humidity* properties. This way it will take much less time. This can be useful if you want to start the measurement, but don't want the call to block until the measurement is ready – for instance, when you are doing other things at the same time.

relative_humidity

The measured relative humidity in percent.

temp_rh_resolution

The temperature and relative humidity resolution

Have one of the following values:¹

value	RH res %	T res C
0	0.04 (12bit)	0.01 (14bit)
1	0.7 (8bit)	0.04 (12bit)
2	0.17 (10bit)	0.02 (13bit)
3	0.08 (11bit)	0.08 (11bit)

temperature

The measured temperature in degrees Celsius.

¹ HTU21D(F) RH/T Sensor IC Datasheet. TE connectivity. 2017. p13

CHAPTER 7

Indices and tables

- `genindex`
- `modindex`
- `search`

a

adafruit_htu21d, 13

A

`adafruit_htu21d` (*module*), 13

H

`HTU21D` (*class in adafruit_htu21d*), 14

M

`measurement()` (*adafruit_htu21d.HTU21D method*),
14

R

`relative_humidity` (*adafruit_htu21d.HTU21D attribute*), 14

T

`temp_rh_resolution` (*adafruit_htu21d.HTU21D attribute*), 14

`temperature` (*adafruit_htu21d.HTU21D attribute*),
14