Adafruit TCS34725 Library Documentation

Release 1.0

Tony DiCola

Nov 27, 2018

Contents

1 Dependencies **3**

2 Usage Example **5**

3 Contributing **7**

4 Building locally **9**
4.1 Sphinx documentation 9

5 Table of Contents **11**
5.1 Simple test 11
5.2 `adafruit_tcs34725` 12
5.2.1 Implementation Notes 12

6 Indices and tables **15**

Python Module Index **17**

CircuitPython module for the TCS34725 color sensor.

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.

CHAPTER 2

Usage Example

See examples/tcs34725_simpletest.py for an example of the module's usage.

CHAPTER 3

Contributing

Contributions are welcome! Please read our Code of Conduct before contributing to help this project stay welcoming.

CHAPTER 4

Building locally

To build this library locally you'll need to install the circuitpython-build-tools package.

```
python3 -m venv .env
source .env/bin/activate
pip install circuitpython-build-tools
```

Once installed, make sure you are in the virtual environment:

```
source .env/bin/activate
```

Then run the build:

```
circuitpython-build-bundles --filename_prefix adafruit-circuitpython-tcs34725 --
↪library_location .
```

4.1 Sphinx documentation

Sphinx is used to build the documentation based on rST files and comments in the code. First, install dependencies (feel free to reuse the virtual environment from above):

```
python3 -m venv .env
source .env/bin/activate
pip install Sphinx sphinx-rtd-theme
```

Now, once you have the virtual environment activated:

```
cd docs
sphinx-build -E -W -b html . _build/html
```

This will output the documentation to `docs/_build/html`. Open the index.html in your browser to view them. It will also (due to -W) error out on any warning like Travis will. This is a good way to locally verify it will pass.

CHAPTER 5

Table of Contents

5.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/tcs34725_simpletest.py

```
# Simple demo of the TCS34725 color sensor.
# Will detect the color from the sensor and print it out every second.
import time

import board
import busio

import adafruit_tcs34725


# Initialize I2C bus and sensor.
i2c = busio.I2C(board.SCL, board.SDA)
sensor = adafruit_tcs34725.TCS34725(i2c)

# Main loop reading color and printing it every second.
while True:
    # Read the color as RGB bytes (0-255 values).
    r, g, b = sensor.color_rgb_bytes
    print('Detected color: #{0:02X}{1:02X}{2:02X}'.format(r, g, b))
    # Read the color temperature and lux of the sensor too.
    try:
        temp = sensor.temperature
        lux = sensor.lux
        print('Temperature: {0}K Lux: {1}'.format(temp, lux))
    except ZeroDivisionError:
        print("No light to measure")
    # Delay for a second and repeat.
```

(continues on next page)

(continued from previous page)

```
    time.sleep(1.0)
```

5.2 adafruit_tcs34725

CircuitPython module for the TCS34725 color sensor. Ported from the micropython-adafruit-tcs34725 module by Radomir Dopieralski: https://github.com/adafruit/micropython-adafruit-tcs34725

See examples/tcs34725_simpletest.py for an example of the usage.

- Author(s): Tony DiCola

5.2.1 Implementation Notes

Hardware:

- Adafruit RGB Color Sensor with IR filter and White LED - TCS34725 (Product ID: 1334)
- Flora Color Sensor with White Illumination LED - TCS34725 (Product ID: 1356)

Software and Dependencies:

- Adafruit CircuitPython firmware for the ESP8622 and M0-based boards: https://github.com/adafruit/circuitpython/releases
- Adafruit's Bus Device library: https://github.com/adafruit/Adafruit_CircuitPython_BusDevice

class adafruit_tcs34725.**TCS34725**(*i2c*, *address=41*)

Driver for the TCS34725 color sensor.

active

The active state of the sensor. Boolean value that will enable/activate the sensor with a value of True and disable with a value of False.

color

Read the RGB color detected by the sensor. Returns an int with 8 bits per channel.

Examples: Red = 16711680 (0xff0000), Green = 65280 (0x00ff00), Blue = 255 (0x0000ff), SlateGray = 7372944 (0x708090)

color_raw

Read the raw RGBC color detected by the sensor. Returns a 4-tuple of 16-bit red, green, blue, clear component byte values (0-65535).

color_rgb_bytes

Read the RGB color detected by the sensor. Returns a 3-tuple of red, green, blue component values as bytes (0-255).

cycles

The persistence cycles of the sensor.

gain

The gain of the sensor. Should be a value of 1, 4, 16, or 60.

integration_time

The integration time of the sensor in milliseconds.

interrupt

True if the interrupt is set. Can be set to False (and only False) to clear the interrupt.

lux
Return the detected light level in lux.

max_value
The minimum threshold value (AIHT register) of the sensor as a 16-bit unsigned value.

min_value
The minimum threshold value (AILT register) of the sensor as a 16-bit unsigned value.

temperature
Return the detected color temperature in degrees.

CHAPTER 6

Indices and tables

- genindex
- modindex
- search

Python Module Index

a

adafruit_tcs34725, 12

Index

A

active (adafruit_tcs34725.TCS34725 attribute), 12
adafruit_tcs34725 (module), 12

C

color (adafruit_tcs34725.TCS34725 attribute), 12
color_raw (adafruit_tcs34725.TCS34725 attribute), 12
color_rgb_bytes (adafruit_tcs34725.TCS34725 attribute), 12
cycles (adafruit_tcs34725.TCS34725 attribute), 12

G

gain (adafruit_tcs34725.TCS34725 attribute), 12

I

integration_time (adafruit_tcs34725.TCS34725 attribute), 12
interrupt (adafruit_tcs34725.TCS34725 attribute), 12

L

lux (adafruit_tcs34725.TCS34725 attribute), 12

M

max_value (adafruit_tcs34725.TCS34725 attribute), 13
min_value (adafruit_tcs34725.TCS34725 attribute), 13

T

TCS34725 (class in adafruit_tcs34725), 12
temperature (adafruit_tcs34725.TCS34725 attribute), 13