
AdafruitVL6180X Library Documentation

Release 1.0

Tony DiCola

Sep 03, 2019

Contents

1	Dependencies	3
2	Usage Example	5
3	Contributing	7
4	Building locally	9
4.1	Sphinx documentation	9
4.2	Sphinx documentation	10
5	Table of Contents	11
5.1	Simple test	11
5.2	adafruit_v16180x	12
5.2.1	Implementation Notes	12
6	Indices and tables	15
	Python Module Index	17
	Index	19

CircuitPython module for the VL6180X distance sensor. See [examples/vl6180x_simpletest.py](#) for a demo of the usage.

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/vl6180x_simpletest.py` for a demo of the 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-v16180x --
↳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.

4.2 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.

5.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/vl6180x_simpletest.py

```
1  # Demo of reading the range and lux from the VL6180x distance sensor and
2  # printing it every second.
3  # Author: Tony DiCola
4  import time
5
6  import board
7  import busio
8
9  import adafruit_vl6180x
10
11
12  # Create I2C bus.
13  i2c = busio.I2C(board.SCL, board.SDA)
14
15  # Create sensor instance.
16  sensor = adafruit_vl6180x.VL6180X(i2c)
17
18  # Main loop prints the range and lux every second:
19  while True:
20      # Read the range in millimeters and print it.
21      range_mm = sensor.range
22      print('Range: {0}mm'.format(range_mm))
23      # Read the light, note this requires specifying a gain value:
24      # - adafruit_vl6180x.ALS_GAIN_1 = 1x
25      # - adafruit_vl6180x.ALS_GAIN_1_25 = 1.25x
26      # - adafruit_vl6180x.ALS_GAIN_1_67 = 1.67x
27      # - adafruit_vl6180x.ALS_GAIN_2_5 = 2.5x
```

(continues on next page)

(continued from previous page)

```

28 # - adafruit_vl6180x.ALS_GAIN_5 = 5x
29 # - adafruit_vl6180x.ALS_GAIN_10 = 10x
30 # - adafruit_vl6180x.ALS_GAIN_20 = 20x
31 # - adafruit_vl6180x.ALS_GAIN_40 = 40x
32 light_lux = sensor.read_lux(adafruit_vl6180x.ALS_GAIN_1)
33 print('Light (1x gain): {0}lux'.format(light_lux))
34 # Delay for a second.
35 time.sleep(1.0)

```

5.2 adafruit_vl6180x

CircuitPython module for the VL6180X distance sensor. See examples/simpletest.py for a demo of the usage.

- Author(s): Tony DiCola

5.2.1 Implementation Notes

Hardware:

- Adafruit VL6180X Time of Flight Distance Ranging Sensor (VL6180) (Product ID: 3316)

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_vl6180x.VL6180X` (*i2c*, *address=41*)

Create an instance of the VL6180X distance sensor. You must pass in the following parameters:

Parameters `i2c` – An instance of the I2C bus connected to the sensor.

Optionally you can specify:

Parameters `address` – The I2C address of the sensor. If not specified the sensor's default value will be assumed.

range

Read the range of an object in front of sensor and return it in mm.

range_status

Retrieve the status/error from a previous range read. This will return a constant value such as:

- `ERROR_NONE` - No error
- `ERROR_SYSERR_1` - System error 1 (see datasheet)
- `ERROR_SYSERR_5` - System error 5 (see datasheet)
- `ERROR_ECEFAIL` - ECE failure
- `ERROR_NOCONVERGE` - No convergence
- `ERROR_RANGEIGNORE` - Outside range ignored
- `ERROR_SNR` - Too much noise
- `ERROR_RAWUFLOW` - Raw value underflow

- ERROR_RAWOFLOW - Raw value overflow
- ERROR_RANGEUFLOW - Range underflow
- ERROR_RANGEOFLOW - Range overflow

read_lux (*gain*)

Read the lux (light value) from the sensor and return it. Must specify the gain value to use for the lux reading: - ALS_GAIN_1 = 1x - ALS_GAIN_1_25 = 1.25x - ALS_GAIN_1_67 = 1.67x - ALS_GAIN_2_5 = 2.5x - ALS_GAIN_5 = 5x - ALS_GAIN_10 = 10x - ALS_GAIN_20 = 20x - ALS_GAIN_40 = 40x

CHAPTER 6

Indices and tables

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

a

[adafruit_vl6180x](#), 12

A

`adafruit_vl6180x` (*module*), 12

R

`range` (*adafruit_vl6180x.VL6180X attribute*), 12

`range_status` (*adafruit_vl6180x.VL6180X attribute*),
12

`read_lux()` (*adafruit_vl6180x.VL6180X method*), 13

V

`VL6180X` (*class in adafruit_vl6180x*), 12