
Adafruit VL53L0X Library Documentation

Release 1.0

Tony DiCola

Aug 13, 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_v15310x	12
5.2.1	Implementation Notes	12
6	Indices and tables	13
	Python Module Index	15

CircuitPython driver for the VL53L0X distance 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 usage in the `examples/v15310x_simpletest.py` file.

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-v15310x --
↳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.

5.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/vl53l0x_simpletest.py

```
1 # Simple demo of the VL53L0X distance sensor.
2 # Will print the sensed range/distance every second.
3 import time
4
5 import board
6 import busio
7
8 import adafruit_vl53l0x
9
10 # Initialize I2C bus and sensor.
11 i2c = busio.I2C(board.SCL, board.SDA)
12 vl53 = adafruit_vl53l0x.VL53L0X(i2c)
13
14 # Optionally adjust the measurement timing budget to change speed and accuracy.
15 # See the example here for more details:
16 # https://github.com/pololu/vl53l0x-arduino/blob/master/examples/Single/Single.ino
17 # For example a higher speed but less accurate timing budget of 20ms:
18 #vl53.measurement_timing_budget = 20000
19 # Or a slower but more accurate timing budget of 200ms:
20 #vl53.measurement_timing_budget = 200000
21 # The default timing budget is 33ms, a good compromise of speed and accuracy.
22
23 # Main loop will read the range and print it every second.
24 while True:
25     print('Range: {0}mm'.format(vl53.range))
26     time.sleep(1.0)
```

5.2 adafruit_vl53l0x

CircuitPython driver for the VL53L0X distance sensor. This code is adapted from the pololu driver here: <https://github.com/pololu/vl53l0x-arduino>

See usage in the examples/vl53l0x_simpletest.py file.

- Author(s): Tony DiCola

5.2.1 Implementation Notes

Hardware:

- Adafruit VL53L0X Time of Flight Distance Sensor - ~30 to 1000mm (Product ID: 3317)

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_vl53l0x.VL53L0X` (*i2c, address=41, io_timeout_s=0*)
Driver for the VL53L0X distance sensor.

measurement_timing_budget

The measurement timing budget in microseconds.

range

Perform a single reading of the range for an object in front of the sensor and return the distance in millimeters.

signal_rate_limit

The signal rate limit in mega counts per second.

CHAPTER 6

Indices and tables

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

a

`adafruit_vl53l0x`, [11](#)

A

adafruit_vl53l0x (module), 11

M

measurement_timing_budget
(adafruit_vl53l0x.VL53L0X attribute), 12

R

range (adafruit_vl53l0x.VL53L0X attribute), 12

S

signal_rate_limit (adafruit_vl53l0x.VL53L0X attribute),
12

V

VL53L0X (class in adafruit_vl53l0x), 12