
AdafruitLSM6DS Library Documentation

Release 1.0

Bryan Siepert

Oct 25, 2021

Contents

1	Dependencies	3
2	Installing from PyPI	5
3	Usage Example	7
4	Documentation	9
5	Contributing	11
6	Documentation	13
7	Table of Contents	15
7.1	LSM6DSOX Simple test	15
7.2	LSM6DSO32 Simple test	15
7.3	LSM6DS Simple test	16
7.4	ISM330DHCX Simple test	16
7.5	LSM6DS Full test	17
7.6	Pedometer Example	18
7.7	Rate test	18
7.8	adafruit_lsm6ds	19
7.8.1	Implementation Notes	19
7.9	This module provides the <code>adafruit_lsm6ds.ism330dhcx</code> subclass of LSM6DS sensors . . .	20
7.10	This module provides the <code>adafruit_lsm6ds.lsm6ds33</code> subclass of LSM6DS sensors	21
7.11	This module provides the <code>adafruit_lsm6ds.lsm6dso32</code> subclass of LSM6DS sensors	22
7.12	This module provides the <code>adafruit_lsm6ds.lsm6dsox</code> subclass of LSM6DS sensors	22
8	Indices and tables	25
	Python Module Index	27
	Index	29

CircuitPython helper library for the LSM6DS family of motion sensors from ST

CHAPTER 1

Dependencies

This driver depends on:

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

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

CHAPTER 2

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-lsm6ds
```

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

```
sudo pip3 install adafruit-circuitpython-lsm6ds
```

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-lsm6ds
```


CHAPTER 3

Usage Example

```
import time
import board
from adafruit_lsm6ds.lsm6dsox import LSM6DSOX

i2c = board.I2C() # uses board.SCL and board.SDA
sox = LSM6DSOX(i2c)

while True:
    print("Acceleration: X:%.2f, Y: %.2f, Z: %.2f m/s^2"%(sox.acceleration))
    print("Gyro X:%.2f, Y: %.2f, Z: %.2f radians/s"%(sox.gyro))
    print("")
    time.sleep(0.5)
```


CHAPTER 4

Documentation

API documentation for this library can be found on [Read the Docs](#).

CHAPTER 5

Contributing

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

CHAPTER 6

Documentation

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

7.1 LSM6DSOX Simple test

Ensure your LSM6DSOX device works with this simple test.

Listing 1: examples/lsm6ds_lsm6dsox_simpletest.py

```
1 # SPDX-FileCopyrightText: Copyright (c) 2020 Bryan Siepert for Adafruit Industries
2 #
3 # SPDX-License-Identifier: MIT
4 import time
5 import board
6 from adafruit_lsm6ds.lsm6dsox import LSM6DSOX
7
8 i2c = board.I2C() # uses board.SCL and board.SDA
9 sensor = LSM6DSOX(i2c)
10
11 while True:
12     print("Acceleration: X:%.2f, Y: %.2f, Z: %.2f m/s^2" % (sensor.acceleration))
13     print("Gyro X:%.2f, Y: %.2f, Z: %.2f radians/s" % (sensor.gyro))
14     print("")
15     time.sleep(0.5)
```

7.2 LSM6DSO32 Simple test

Ensure your LSM6DSO32 device works with this simple test.

Listing 2: examples/lsm6ds_lsm6dso32_simpletest.py

```
1 # SPDX-FileCopyrightText: Copyright (c) 2020 Bryan Siepert for Adafruit Industries
2 #
3 # SPDX-License-Identifier: MIT
```

(continues on next page)

(continued from previous page)

```

4 import time
5 import board
6 from adafruit_lsm6ds.lsm6dso32 import LSM6DSO32
7
8 i2c = board.I2C() # uses board.SCL and board.SDA
9 sensor = LSM6DSO32(i2c)
10
11 while True:
12     print("Acceleration: X:%.2f, Y: %.2f, Z: %.2f m/s^2" % (sensor.acceleration))
13     print("Gyro X:%.2f, Y: %.2f, Z: %.2f radians/s" % (sensor.gyro))
14     print("")
15     time.sleep(0.5)

```

7.3 LSM6DS Simple test

Ensure your LSM6DS device works with this simple test.

Listing 3: examples/lsm6ds_lsm6ds33_simpletest.py

```

1 # SPDX-FileCopyrightText: Copyright (c) 2020 Bryan Siepert for Adafruit Industries
2 #
3 # SPDX-License-Identifier: MIT
4 import time
5 import board
6 from adafruit_lsm6ds.lsm6ds33 import LSM6DS33
7
8 i2c = board.I2C() # uses board.SCL and board.SDA
9 sensor = LSM6DS33(i2c)
10
11 while True:
12     print("Acceleration: X:%.2f, Y: %.2f, Z: %.2f m/s^2" % (sensor.acceleration))
13     print("Gyro X:%.2f, Y: %.2f, Z: %.2f radians/s" % (sensor.gyro))
14     print("")
15     time.sleep(0.5)

```

7.4 ISM330DHCX Simple test

Ensure your ISM330DHCX device works with this simple test.

Listing 4: examples/lsm6ds_ism330dhcx_simpletest.py

```

1 # SPDX-FileCopyrightText: Copyright (c) 2020 Bryan Siepert for Adafruit Industries
2 #
3 # SPDX-License-Identifier: MIT
4 import time
5 import board
6 from adafruit_lsm6ds.ism330dhcx import ISM330DHCX
7
8 i2c = board.I2C() # uses board.SCL and board.SDA
9 sensor = ISM330DHCX(i2c)
10

```

(continues on next page)

(continued from previous page)

```

11 while True:
12     print("Acceleration: X:%.2f, Y: %.2f, Z: %.2f m/s^2" % (sensor.acceleration))
13     print("Gyro X:%.2f, Y: %.2f, Z: %.2f radians/s" % (sensor.gyro))
14     print("")
15     time.sleep(0.5)

```

7.5 LSM6DS Full test

LSM6DS Full tests

Listing 5: examples/lsm6ds_full_test.py

```

1  # SPDX-FileCopyrightText: Copyright (c) 2020 Bryan Siepert for Adafruit Industries
2  #
3  # SPDX-License-Identifier: MIT
4  import time
5  import board
6
7  # pylint:disable=no-member
8  from adafruit_lsm6ds import Rate, AccelRange, GyroRange
9
10 from adafruit_lsm6ds.lsm6dsox import LSM6DSOX as LSM6DS
11
12 # from adafruit_lsm6ds.lsm6ds33 import LSM6DS33 as LSM6DS
13 # from adafruit_lsm6ds.lsm6dso32 import LSM6DSO32 as LSM6DS
14 # from adafruit_lsm6ds.ism330dhcx import ISM330DHCX as LSM6DS
15
16 i2c = board.I2C() # uses board.SCL and board.SDA
17 sensor = LSM6DS(i2c)
18
19 sensor.accelerometer_range = AccelRange.RANGE_8G
20 print(
21     "Accelerometer range set to: %d G" % AccelRange.string[sensor.accelerometer_range]
22 )
23
24 sensor.gyro_range = GyroRange.RANGE_2000_DPS
25 print("Gyro range set to: %d DPS" % GyroRange.string[sensor.gyro_range])
26
27 sensor.accelerometer_data_rate = Rate.RATE_1_66K_HZ
28 # sensor.accelerometer_data_rate = Rate.RATE_12_5_HZ
29 print("Accelerometer rate set to: %d HZ" % Rate.string[sensor.accelerometer_data_
30     ↳rate])
31
32 sensor.gyro_data_rate = Rate.RATE_1_66K_HZ
33 print("Gyro rate set to: %d HZ" % Rate.string[sensor.gyro_data_rate])
34
35 while True:
36     print(
37         "Accel X:%.2f Y:%.2f Z:%.2f ms^2 Gyro X:%.2f Y:%.2f Z:%.2f radians/s"
38         % (sensor.acceleration + sensor.gyro)
39     )
40     time.sleep(0.05)

```

7.6 Pedometer Example

Example showing how to use the device as a pedometer

Listing 6: examples/lsm6ds_pedometer.py

```

1  # SPDX-FileCopyrightText: Copyright (c) 2020 Bryan Siepert for Adafruit Industries
2  #
3  # SPDX-License-Identifier: MIT
4  """ This example shows off how to use the step counter built
5  into the ST LSM6DS series IMUs. The steps are calculated in
6  the chip so you don't have to do any calculations!"""
7
8  import time
9  import board
10
11 # pylint:disable=no-member
12 from adafruit_lsm6ds.lsm6ds33 import LSM6DS33
13 from adafruit_lsm6ds import Rate, AccelRange
14
15 i2c = board.I2C() # uses board.SCL and board.SDA
16 sensor = LSM6DS33(i2c)
17
18 # enable accelerometer sensor @ 2G and 26 Hz
19 sensor.accelerometer_range = AccelRange.RANGE_2G
20 sensor.accelerometer_data_rate = Rate.RATE_26_HZ
21 # no gyro used for step detection
22 sensor.gyro_data_rate = Rate.RATE_SHUTDOWN
23
24 # enable the pedometer
25 sensor.pedometer_enable = True
26
27 while True:
28     print("Steps: ", sensor.pedometer_steps)
29     time.sleep(1)

```

7.7 Rate test

Example showing a Rate test

Listing 7: examples/lsm6ds_rate_test.py

```

1  # SPDX-FileCopyrightText: Copyright (c) 2020 Bryan Siepert for Adafruit Industries
2  #
3  # SPDX-License-Identifier: MIT
4  import board
5
6  # pylint:disable=no-member,unused-import
7  from adafruit_lsm6ds import Rate
8  from adafruit_lsm6ds.lsm6dsox import LSM6DSOX as LSM6DS
9
10 # from adafruit_lsm6ds.lsm6ds33 import LSM6DS33 as LSM6DS
11 # from adafruit_lsm6ds.lsm6dso32 import LSM6DSO32 as LSM6DS
12 # from adafruit_lsm6ds.ism330dhcx import ISM330DHCX as LSM6DS
13

```

(continues on next page)

(continued from previous page)

```

14 i2c = board.I2C() # uses board.SCL and board.SDA
15 sensor = LSM6DS(i2c)
16
17 while True:
18     sensor.accelerometer_data_rate = Rate.RATE_12_5_HZ
19     sensor.gyro_data_rate = Rate.RATE_12_5_HZ
20     for i in range(100):
21         print(
22             "(%.2f, %.2f, %.2f, %.2f, %.2f, %.2f" % (sensor.acceleration + sensor.
↳gyro)
23             )
24         print()
25
26     sensor.accelerometer_data_rate = Rate.RATE_52_HZ
27     sensor.gyro_data_rate = Rate.RATE_52_HZ
28     for i in range(100):
29         print(
30             "(%.2f, %.2f, %.2f, %.2f, %.2f, %.2f" % (sensor.acceleration + sensor.
↳gyro)
31             )
32         print()
33
34     sensor.accelerometer_data_rate = Rate.RATE_416_HZ
35     sensor.gyro_data_rate = Rate.RATE_416_HZ
36     for i in range(100):
37         print(
38             "(%.2f, %.2f, %.2f, %.2f, %.2f, %.2f" % (sensor.acceleration + sensor.
↳gyro)
39             )
40         print()

```

7.8 adafruit_lsm6ds

CircuitPython helper library for the LSM6DS family of motion sensors from ST

- Author(s): Bryan Siepert, Jose David M.

7.8.1 Implementation Notes

Hardware:

- Adafruit LSM6DSOX 6 DoF Accelerometer and Gyroscope (Product ID: 4438)
- Adafruit ISM330DHCX - 6 DoF IMU - Accelerometer and Gyroscope (Product ID: 4502)
- Adafruit LSM6DSOX 6-DoF Accelerometer and Gyroscope (Product ID: 4692)
- Adafruit LSM6DS33 6-DoF Accel + Gyro IMU (Product ID: 4480)
- Adafruit ISM330DHCX + LIS3MDL FeatherWing - High Precision 9-DoF IMU (Product ID: 4569)
- Adafruit LSM6DSOX + LIS3MDL - Precision 9 DoF IMU (Product ID: 4517)
- Adafruit LSM6DS33 + LIS3MDL - 9 DoF IMU with Accel / Gyro / Mag (Product ID: 4485)
- Adafruit LSM6DSOX + LIS3MDL FeatherWing - Precision 9-DoF IMU (Product ID: 4565)

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
- Adafruit's Register library: https://github.com/adafruit/Adafruit_CircuitPython_Register

class `adafruit_lsm6ds.LSM6DS` (*i2c_bus*, *address=<sphinx.ext.autodoc.importer.MockObject object>*)

Driver for the LSM6DSOX 6-axis accelerometer and gyroscope.

Parameters

- **i2c_bus** (*I2C*) – The I2C bus the LSM6DSOX is connected to.
- **address** (*int*) – The I2C device address. Defaults to 0x6A

pedometer_steps

The number of steps detected by the pedometer. You must enable with `pedometer_enable` before calling. Use `pedometer_reset` to reset the number of steps

reset ()

Resets the sensor's configuration into an initial state

acceleration

The x, y, z acceleration values returned in a 3-tuple and are in $m / s ^ 2$.

gyro

The x, y, z angular velocity values returned in a 3-tuple and are in radians / second

accelerometer_range

Adjusts the range of values that the sensor can measure, from +/- 2G to +/-16G Note that larger ranges will be less accurate. Must be an `AccelRange`

gyro_range

Adjusts the range of values that the sensor can measure, from 125 Degrees/s to 2000 degrees/s. Note that larger ranges will be less accurate. Must be a `GyroRange`.

accelerometer_data_rate

Select the rate at which the accelerometer takes measurements. Must be a `Rate`

gyro_data_rate

Select the rate at which the gyro takes measurements. Must be a `Rate`

pedometer_enable

Whether the pedometer function on the accelerometer is enabled

high_pass_filter

The high pass filter applied to accelerometer data

temperature

Temperature in Celsius

7.9 This module provides the `adafruit_lsm6ds.ism330dhcx` subclass of LSM6DS sensors

class `adafruit_lsm6ds.ism330dhcx.ISM330DHCX` (*i2c_bus*, *address=<sphinx.ext.autodoc.importer.MockObject object>*)

Driver for the ISM330DHCX 6-axis accelerometer and gyroscope.

Parameters

- **i2c_bus** (*I2C*) – The I2C bus the device is connected to.
- **address** (*int*) – The I2C device address. Defaults to 0x6A

Quickstart: Importing and using the device

Here is an example of using the `ISM330DHCX` class. First you will need to import the libraries to use the sensor

```
import board
from adafruit_lsm6ds.ism330dhcx import ISM330DHCX
```

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 = ISM330DHCX(i2c)
```

Now you have access to the `acceleration` and `gyro` attributes

```
acc_x, acc_y, acc_z = sensor.acceleration
gyro_x, gyro_y, gyro_z = sensor.gyro
```

gyro_range

Adjusts the range of values that the sensor can measure, from 125 Degrees/s to 4000 degrees/s. Note that larger ranges will be less accurate. Must be a `GyroRange`. 4000 DPS is only available for the ISM330DHCX

7.10 This module provides the `adafruit_lsm6ds.lsm6ds33` subclass of LSM6DS sensors

class `adafruit_lsm6ds.lsm6ds33.LSM6DS33` (*i2c_bus*, *address*=`<sphinx.ext.autodoc.importer._MockObject object>`)

Driver for the LSM6DS33 6-axis accelerometer and gyroscope.

Parameters

- **i2c_bus** (*I2C*) – The I2C bus the LSM6DS33 is connected to.
- **address** (*int*) – The I2C device address. Defaults to 0x6A

Quickstart: Importing and using the device

Here is an example of using the `LSM6DS33` class. First you will need to import the libraries to use the sensor

```
import board
from adafruit_lsm6ds.lsm6ds33 import LSM6DS33
```

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 = LSM6DS33(i2c)
```

Now you have access to the `acceleration` and `gyro` attributes

```
acc_x, acc_y, acc_z = sensor.acceleration
gyro_x, gyro_z, gyro_z = sensor.gyro
```

7.11 This module provides the `adafruit_lsm6ds.lsm6dso32` subclass of LSM6DS sensors

class `adafruit_lsm6ds.lsm6dso32.LSM6DSO32` (*i2c_bus*, *address*=<*sphinx.ext.autodoc.importer._MockObject object*>)

Driver for the LSM6DSO32 6-axis accelerometer and gyroscope.

Parameters

- **`i2c_bus`** (*I2C*) – The I2C bus the LSM6DSO32 is connected to.
- **`address`** – The I2C device address. Defaults to 0x6A

Quickstart: Importing and using the device

Here is an example of using the `LSM6DSO32` class. First you will need to import the libraries to use the sensor

```
import board
from adafruit_lsm6ds.lsm6dso32 import LSM6DSO32
```

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 = LSM6DSO32(i2c)
```

Now you have access to the `acceleration` and `gyro`: attributes

```
acc_x, acc_y, acc_z = sensor.acceleration
gyro_x, gyro_z, gyro_z = sensor.gyro
```

7.12 This module provides the `adafruit_lsm6ds.lsm6dsox` subclass of LSM6DS sensors

class `adafruit_lsm6ds.lsm6dsox.LSM6DSOX` (*i2c_bus*, *address*=<*sphinx.ext.autodoc.importer._MockObject object*>)

Driver for the LSM6DSOX 6-axis accelerometer and gyroscope.

Parameters

- **`i2c_bus`** (*I2C*) – The I2C bus the LSM6DSOX is connected to.
- **`address`** (*int*) – The I2C device address. Defaults to 0x6A

Quickstart: Importing and using the device

Here is an example of using the `LSM6DSOX` class. First you will need to import the libraries to use the sensor

```
import board
from adafruit_lsm6ds.lsm6dsox import LSM6DSOX
```

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 = LSM6DSOX(i2c)
```

Now you have access to the acceleration and gyro: attributes

```
acc_x, acc_y, acc_z = sensor.acceleration
gyro_x, gyro_y, gyro_z = sensor.gyro
```


CHAPTER 8

Indices and tables

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

a

adafruit_lsm6ds, [19](#)
adafruit_lsm6ds.ism330dhcx, [20](#)
adafruit_lsm6ds.lsm6ds33, [21](#)
adafruit_lsm6ds.lsm6dso32, [22](#)
adafruit_lsm6ds.lsm6dsox, [22](#)

A

acceleration (*adafruit_lsm6ds.LSM6DS attribute*), 20
accelerometer_data_rate (*adafruit_lsm6ds.LSM6DS attribute*), 20
accelerometer_range (*adafruit_lsm6ds.LSM6DS attribute*), 20
adafruit_lsm6ds (*module*), 19
adafruit_lsm6ds.ism330dhcx (*module*), 20
adafruit_lsm6ds.lsm6ds33 (*module*), 21
adafruit_lsm6ds.lsm6dso32 (*module*), 22
adafruit_lsm6ds.lsm6dsox (*module*), 22

G

gyro (*adafruit_lsm6ds.LSM6DS attribute*), 20
gyro_data_rate (*adafruit_lsm6ds.LSM6DS attribute*), 20
gyro_range (*adafruit_lsm6ds.ism330dhcx.ISM330DHCX attribute*), 21
gyro_range (*adafruit_lsm6ds.LSM6DS attribute*), 20

H

high_pass_filter (*adafruit_lsm6ds.LSM6DS attribute*), 20

I

ISM330DHCX (*class in adafruit_lsm6ds.ism330dhcx*), 20

L

LSM6DS (*class in adafruit_lsm6ds*), 20
LSM6DS33 (*class in adafruit_lsm6ds.lsm6ds33*), 21
LSM6DSO32 (*class in adafruit_lsm6ds.lsm6dso32*), 22
LSM6DSOX (*class in adafruit_lsm6ds.lsm6dsox*), 22

P

pedometer_enable (*adafruit_lsm6ds.LSM6DS attribute*), 20

pedometer_steps (*adafruit_lsm6ds.LSM6DS attribute*), 20

R

reset () (*adafruit_lsm6ds.LSM6DS method*), 20

T

temperature (*adafruit_lsm6ds.LSM6DS attribute*), 20