AdafruitDS3502 Library Documentation

Release 1.0

Bryan Siepert

Mar 16, 2020

Contents

CircuitPython library for the Maxim DS3502 I2C Potentionmeter

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.

1.1 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-ds3502
```

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

```
sudo pip3 install adafruit-circuitpython-ds3502
```

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-ds3502
```

CHAPTER 2

Usage Example

```
from time import sleep
import board
import adafruit_ds3502
from analogio import AnalogIn

####### NOTE ################
# this example will not work with Blinka/rasberry Pi due to the lack of analog pins.
# Blinka and Raspberry Pi users should run the "ds3502_blinka_simpletest.py" example

i2c = board.I2C()
ds3502 = adafruit_ds3502.DS3502(i2c)
wiper_output = AnalogIn(board.A0)

while True:

    ds3502.wiper = 127
    print("Wiper set to %d"%ds3502.wiper)
    voltage = wiper_output.value
    voltage *= 3.3
    voltage /= 65535
    print("Wiper voltage: %.2f"%voltage)
    print("")
    sleep(1.0)

    ds3502.wiper = 0
    print("Wiper set to %d"%ds3502.wiper)
    voltage = wiper_output.value
    voltage *= 3.3
    voltage /= 65535
    print("Wiper voltage: %.2f"%voltage)
    print("")
    sleep(1.0)

    ds3502.wiper = 63
```

(continues on next page)

(continued from previous page)

```
    print("Wiper set to %d"%ds3502.wiper)
    voltage = wiper_output.value
    voltage *= 3.3
    voltage /= 65535
    print("Wiper voltage: %.2f"%voltage)
    print("")
    sleep(1.0)
```

CHAPTER 3

Contributing

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

CHAPTER 4

Documentation

For information on building library documentation, please check out this guide.

CHAPTER 5

Table of Contents

5.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/ds3502_simpletest.py

```
from time import sleep
import board
from analogio import AnalogIn
import adafruit_ds3502

####### NOTE ################
# this example will not work with Blinka/rasberry Pi due to the lack of analog pins.
# Blinka and Raspberry Pi users should run the "ds3502_blinka_simpletest.py" example

i2c = board.I2C()
ds3502 = adafruit_ds3502.DS3502(i2c)
wiper_output = AnalogIn(board.A0)

while True:

    ds3502.wiper = 127
    print("Wiper set to %d" % ds3502.wiper)
    voltage = wiper_output.value
    voltage *= 3.3
    voltage /= 65535
    print("Wiper voltage: %.2f V" % voltage)
    print("")
    sleep(1.0)

    ds3502.wiper = 0
    print("Wiper set to %d" % ds3502.wiper)
    voltage = wiper_output.value
```

(continues on next page)

(continued from previous page)

```
    voltage *= 3.3
    voltage /= 65535
    print("Wiper voltage: %.2f V" % voltage)
    print("")
    sleep(1.0)

    ds3502.wiper = 63
    print("Wiper set to %d" % ds3502.wiper)
    voltage = wiper_output.value
    voltage *= 3.3
    voltage /= 65535
    print("Wiper voltage: %.2f V" % voltage)
    print("")
    sleep(1.0)
```

5.2 adafruit_ds3502

CircuitPython library for the Maxim DS3502 I2C Digital Potentionmeter

- Author(s): Bryan Siepert

5.2.1 Implementation Notes

Hardware:

- Adafruit DS3502

Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: https://github.com/adafruit/circuitpython/releases
- 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_ds3502.**DS3502**(*i2c_bus*, *address=40*)
Driver for the DS3502 I2C Digital Potentiometer.

Parameters

- **i2c_bus** (*I2C*) – The I2C bus the DS3502 is connected to.
- **address** – The I2C device address for the sensor. Default is `0x40`.

set_default(*default*)
Sets the wiper's default value and current value to the given value

Parameters **new_default** – The value from 0-127 to set as the wiper's default.

wiper
The value of the potentionmeter's wiper.

Parameters **wiper_value** – The value from 0-127 to set the wiper to.

CHAPTER 6

Indices and tables

- genindex
- modindex
- search

Python Module Index

a

Index