Woolsey Workshop CircuitPython 74HC165 Library Documentation

Release 1.0

John Woolsey

Jul 26, 2021

CONTENTS

CircuitPython driver for 74HC165 shift register.

CHAPTER

ONE

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 or individual libraries can be installed using circup.

CHAPTER

TWO

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 woolseyworkshop-circuitpython-74hc165
```

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

```
sudo pip3 install woolseyworkshop-circuitpython-74hc165
```

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 woolseyworkshop-circuitpython-74hc165
```

CHAPTER

THREE

USAGE EXAMPLE

```
import time
import board
import digitalio
import wws_74hc165

latch_pin = digitalio.DigitalInOut(board.D5)
sr = wws_74hc165.ShiftRegister74HC165(board.SPI(), latch_pin)

pin1 = sr.get_pin(1)

while True:
    print(f"pin 1 = {pin1.value}")
    time.sleep(1)
```

Also see the Adding Digital I/O To Your CircuitPython Compatible Board: Part 2 - The 74HC165 tutorial on Woolsey-Workshop.com for additional usage information.

CHAPTER

FOUR

CONTRIBUTING

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

CHAPTER

FIVE

DOCUMENTATION

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

CHAPTER

SIX

TABLE OF CONTENTS

6.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/74hc165_simpletest.py

```
# SPDX-FileCopyrightText: 2017 Scott Shawcroft, written for Adafruit Industries
# SPDX-FileCopyrightText: Copyright (c) 2021 John Woolsey for Woolsey Workshop
#
# SPDX-License-Identifier: Unlicense


import time
import board
import digitalio
import wws_74hc165

latch_pin = digitalio.DigitalInOut(board.D5)
sr = wws_74hc165.ShiftRegister74HC165(board.SPI(), latch_pin)

pin1 = sr.get_pin(1)

while True:
    print(f"pin 1 = {pin1.value}")
    time.sleep(1)
```

6.2 wws_74hc165

CircuitPython driver for 74HC165 shift register.

- Author(s): John Woolsey

6.2.1 Implementation Notes

Based on Adafruit_CircuitPython_74HC595 driver library.

Hardware:

- 74HC165 Shift Register

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

`class wws_74hc165.DigitalInOut(pin_number, shift_register_74hc165)`
Digital input/output of the 74HC165. The interface is exactly the same as the `digitalio.DigitalInOut` class, however note that by design this device is INPUT ONLY! Attempting to write outputs or set direction as output will raise an exception.

Specify the pin number of the shift register (0...7) and the ShiftRegister74HC165 instance.

`property direction`
`Direction` can only be set to `INPUT`.

`property pull`
Pull-up/down is not supported, return None for no pull-up/down.

`switch_to_input(**kwargs)`
`DigitalInOut switch_to_input`

`switch_to_output(value=False, **kwargs)`
`switch_to_output` is not supported.

`property value`
The value of the pin, either True for high or False for low.

`class wws_74hc165.ShiftRegister74HC165(spi, latch, number_of_shift_registers=1)`
Initialize the 74HC165 on the specified SPI bus and indicate the number of shift registers being used.

`get_pin(pin)`
Convenience function to create an instance of the DigitalInOut class pointing at the specified pin of this 74HC165 device.

`property gpio`
The raw GPIO input register. Each bit represents the input value of the associated pin (0 = low, 1 = high).

`property number_of_shift_registers`
The number of shift register chips.

CHAPTER

SEVEN

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

W

INDEX