AdafruitMatrixKeypad Library Documentation

Release 1.0

ladyada

Jun 07, 2021

Contents

This simple helper library lets you create objects that will scan and detect keypresses on passive matrix keypads

CHAPTER 1

Dependencies

This driver depends on:

- Adafruit CircuitPython

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-matrixkeypad
```

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

```
sudo pip3 install adafruit-circuitpython-matrixkeypad
```

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-matrixkeypad
```

CHAPTER 3

Usage Example

```
import adafruit_matrixkeypad
from digitalio import DigitalInOut
import board

# Classic 3x4 matrix keypad
cols = [DigitalInOut(x) for x in (board.D2, board.D0, board.D4)]
rows = [DigitalInOut(x) for x in (board.D1, board.D6, board.D5, board.D3)]
keys = ((1, 2, 3),
        (4, 5, 6),
        (7, 8, 9),
        ('*', 0, '#'))

keypad = adafruit_matrixkeypad.Matrix_Keypad(rows, cols, keys)

while True:
    keys = keypad.pressed_keys
    if keys:
        print("Pressed: ", keys)
    time.sleep(0.1)
```

CHAPTER 4

Contributing

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

CHAPTER 5

Documentation

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

CHAPTER 6

Table of Contents

6.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/matrixkeypad_simpletest.py

```
# SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
# SPDX-License-Identifier: MIT

import time
import digitalio
import board
import adafruit_matrixkeypad

# Membrane 3x4 matrix keypad - https://www.adafruit.com/product/419
cols = [digitalio.DigitalInOut(x) for x in (board.D9, board.D6, board.D5)]
rows = [digitalio.DigitalInOut(x) for x in (board.D13, board.D12, board.D11, board.
↪D10)]

# 3x4 matrix keypad - Rows and columns are mixed up for https://www.adafruit.com/
↪product/3845
# Use the same wiring as in the guide with the following setup lines:
# cols = [digitalio.DigitalInOut(x) for x in (board.D11, board.D13, board.D9)]
# rows = [digitalio.DigitalInOut(x) for x in (board.D12, board.D5, board.D6, board.
↪D10)]

keys = ((1, 2, 3), (4, 5, 6), (7, 8, 9), ("*", 0, "#"))

keypad = adafruit_matrixkeypad.Matrix_Keypad(rows, cols, keys)

while True:
    keys = keypad.pressed_keys
    if keys:
```

(continues on next page)

(continued from previous page)

```
25          print("Pressed: ", keys)
26      time.sleep(0.1)
```

6.2 adafruit_matrixkeypad

CircuitPython library for matrix keypads

- Author(s): ladyada

6.2.1 Implementation Notes

Hardware:

- Flexible 3x4 Matrix Keypad <https://www.adafruit.com/product/419>
- Phone-style 3x4 Matrix Keypad <https://www.adafruit.com/product/1824>

Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: https://github.com/adafruit/circuitpython/releases

class adafruit_matrixkeypad.**Matrix_Keypad**(*row_pins*, *col_pins*, *keys*)
Driver for passive matrix keypads - any size

pressed_keys
An array containing all detected keys that are pressed from the initalized list-of-lists passed in during creation

CHAPTER 7

Indices and tables

- genindex
- modindex
- search

Python Module Index

a

Index