

Raspberry Pi Web ストリーミングカメラ 設定手順

1. 準備

配線

docs フォルダの Connection_v1-2.jpeg を参考にして、ラズベリーパイと各機器を接続します。Raspberry Pi への給電は、USB Dip キットの VCC から Raspberry Pi の 5V ピンで行います。

必要なパッケージのインストール

Raspberry Pi を使ってモータを制御するために必要な Python パッケージをインストールします。コマンドプロンプトを開き、次のコマンドを実行します。

```
sudo apt install pigpio python3-pigpio
```

```
sudo apt install python3-flask
```

pigpio は、Raspberry Pi の GPIO ピンを PWM 制御するために使用します。Flask は Web サーバーを構築するために使用します。

2. LED のテスト

LED 周りの配線をテストします。

コマンドプロンプトを開き、以下のコマンドを実行して pigpiod を起動します。

```
sudo pigpiod
```

次に src フォルダの "test_led.py" を実行します。1 秒毎に赤→緑→青の順に LED の色が変われば成功です。

異なる場合は LED と RaspberryPi の接続等を確認します。

3. 加速度センサのテスト

I2C と必要なパッケージのインストール

Raspberry Pi の I2C を有効にします。設定からコンフィギュレーションを開き、インターフェース画面で I2C を有効にします。

"smbus", "matplotlib", "deque" の Python パッケージが未インストールの場合はインストールします。コマンドプロンプトから以下のコマンドでインストールします。

```
sudo pip install "パッケージ名"
```

加速度センサのテスト

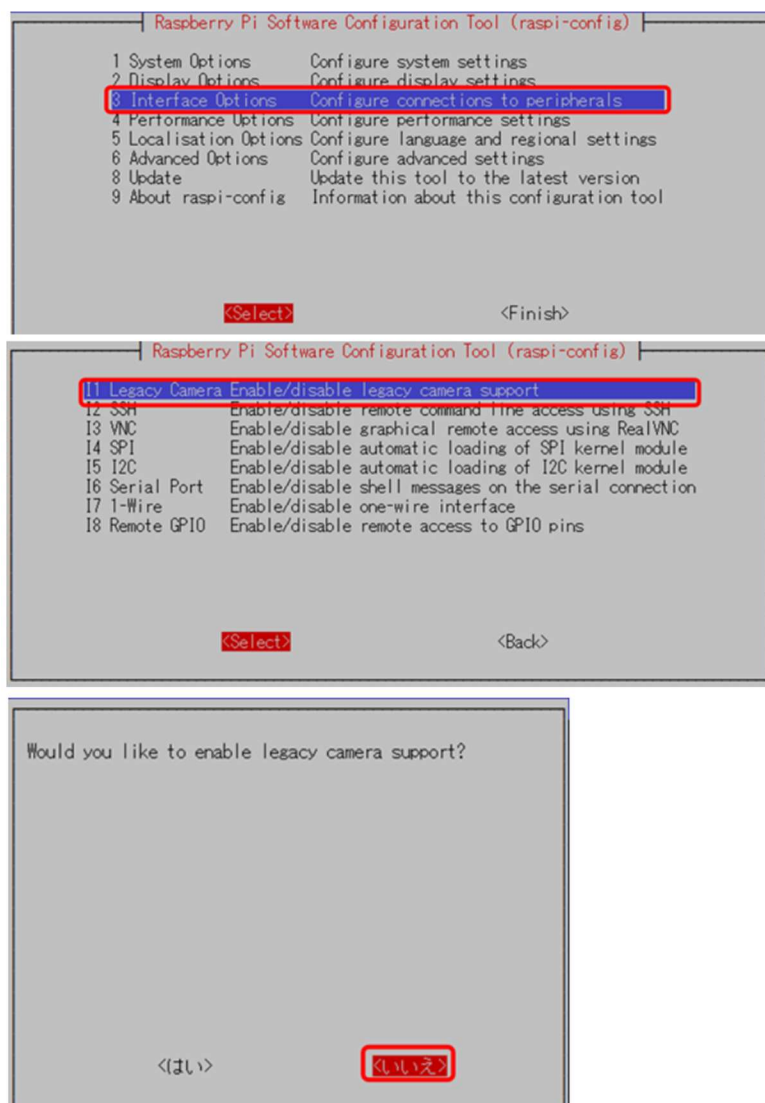
src フォルダの "test_acc.py" を実行します。グラフ描画ウィンドウが開き、X, Y, Z 軸の加速度の値がリアルタイムで表示されます。

値が動かない場合は加速度センサと Raspberry Pi の接続等を確認します。

4. カメラのセットアップとテスト

libcamera モードへの切り替え

- （以降の説明は Raspberry Pi OS が Bullseye であることを前提としています。）
Raspberry Pi OS Bullseye では、デフォルトで libcamera モードが使用されるようになっていますが、古い設定ではレガシーカメラモードが有効になっている可能性があります。以下の手順で、libcamera モードに切り替えます。
- ターミナルを開き、`sudo raspi-config` コマンドを実行します。下図の順番に操作し、**Legacy camera support を無効**にし、Raspberry Pi を再起動します。

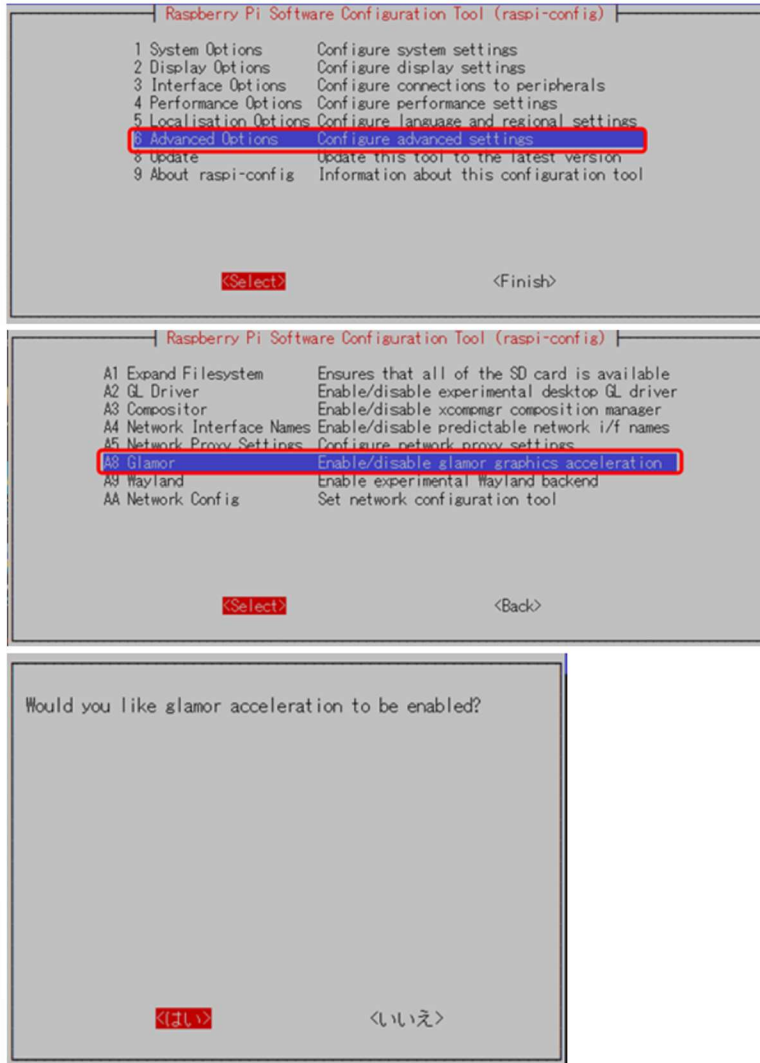


Glamor の有効化

- Glamor は、GPU によるハードウェアアクセラレーションを活用した X ウィンドウシステムの描画を高速化する技術です。OpenCV などを使用したカメラ映像の処理

を効率化するために、Glamor を有効にします。

2. ターミナルを開き、`sudo raspi-config` コマンドを実行します。下図の順番に操作し、Glamor acceleration を有効にし、Raspberry Pi を再起動します。



カメラモジュールの指定

1. “/boot/config.txt”にカメラモジュールを指定します。”/boot/”フォルダに移動し、スーパーユーザ権限で”config.txt”を開きます。
2. ファイルの末尾に下記行を追記します。
“dtoverlay=ov5647”
3. Raspberry Pi を再起動します。
4. ターミナルを開き、“libcamera-hello --list-cameras”を実行します。下図のように ov5647 が認識されていれば完了です。
5. ターミナルで“libcamera-hello”を実行します。カメラの接続がうまくできている場合、プレビューが 5 秒間表示されます。

```
yaopi@camera:~$ libcamera-hello --list-cameras
Available cameras
-----
0 : ov5647 [2592x1944] (/base/soc/i2c0mux/i2c@1/ov5647@36)
    Modes: 'SGBRG10_CS12P' : 640x480 [58.92 fps - (16, 0)/2560x1920 crop]
    1296x972 [43.25 fps - (0, 0)/2592x1944 crop]
    1920x1080 [30.62 fps - (348, 434)/1928x1080 crop]
    2592x1944 [15.63 fps - (0, 0)/2592x1944 crop]
```

OpenCV および picamera2 のインストール

次に、ターミナルで以下のコマンドを実行し、必要なライブラリをインストールします。

“sudo pip3 install opencv-python”

“sudo apt install python3-picamera2”

“sudo apt install libatlas3-base”

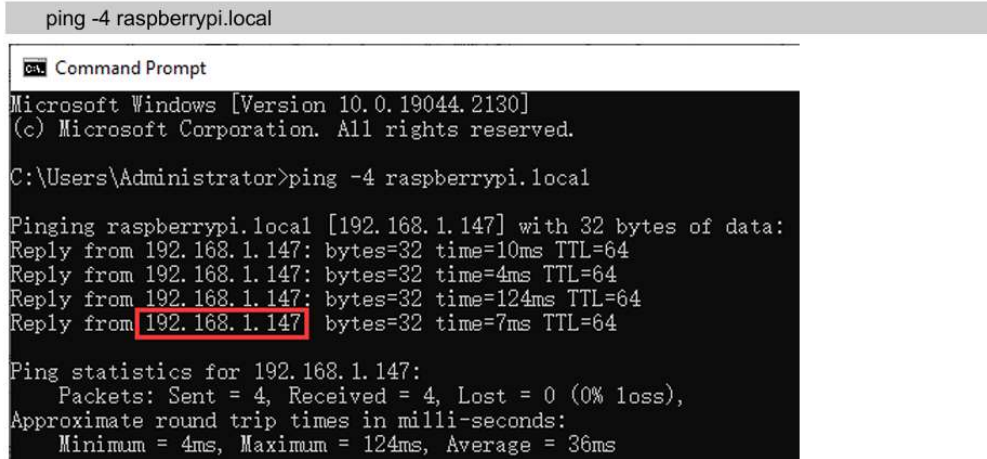
カメラテスト

1. カメラテストプログラム “test_camera.py” を github からダウンロードし、Raspberry Pi に保存します。
2. プログラムを実行して、プレビューが表示され、また “test_camera.py” のフォルダに “test.jpg” という名前で画像ファイルが作成されることを確認して下さい。
3. import cv2 で numpy のモジュールに関するエラーが表示される場合、numpy のバージョンと opencv のバージョンとの組み合わせが悪い場合があります。
numpy を apt-get python3-numpy でインストールしている場合、numpy のバージョンが古すぎる可能性があります。apt-get でインストールした numpy をアンインストールし、pip3 でインストールし直しましょう。
pip3 で numpy==2.X.Y のバージョンをインストールした場合、numpy が新しすぎる可能性もあります。pip3 install numpy==1.XX.YY でバージョンを指定します。

Web ストリーミングの確認

1. Raspberry Pi を WiFi に接続します。
2. ターミナルで”ping -4 raspberrypi.local”を実行し、Raspberry Pi の IP アドレスを確認します。

```
ping -4 raspberrypi.local
```



```
Microsoft Windows [Version 10.0.19044.2130]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator>ping -4 raspberrypi.local

Pinging raspberrypi.local [192.168.1.147] with 32 bytes of data:
Reply from 192.168.1.147: bytes=32 time=10ms TTL=64
Reply from 192.168.1.147: bytes=32 time=4ms TTL=64
Reply from 192.168.1.147: bytes=32 time=124ms TTL=64
Reply from 192.168.1.147: bytes=32 time=7ms TTL=64

Ping statistics for 192.168.1.147:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 4ms, Maximum = 124ms, Average = 36ms
```

Then 192.168.1.147 is my Raspberry Pi IP.

3. プログラム “test_webcamera.py” を github からダウンロードし、Raspberry Pi に保存します。
4. プログラムを実行します。WEB サーバーが立ち上がります。
5. Raspberry Pi と同一の WiFi に接続した機器で Web ブラウザを立上げ、Raspberry Pi にアクセスします。url 欄に “IP アドレス:8000” と打ち込むことでアクセスが可能です。

例 : <http://192.168.1.100:8000/>

ブラウザにアクセスすると、カメラの映像がリアルタイムで表示されます。

5. モータの回転方向テスト

1. プログラム “test_motor.py” を実行します。
2. うまく配線できていれば、左右のモータが 1 秒間前進方向に回転した後、1 秒間後進方向に回転して停止します。
3. モータの動きが前後逆の場合は、モータの配線を入れかえます（例：モータの線 1 を BO1、モータの線 2 を BO2 に接続していた場合、線 1 を BO2 に、線 2 を BO1 に接続し直す。）

6. プログラムの実行

1. ここまでのテストが完了すれば、準備 OK です。プログラム “main_v1-2.py” を実行します。これにより web サーバーが起動します。
2. PC やスマホでラズベリーパイの IP アドレスにアクセスすると、下図の画面が表示されます。
3. 矢印にカーソルを合わせる、もしくは矢印をタップすると、モータが動作します。カメラで撮影した映像が常時表示されます。”yellow” “purple” “skyblue” と書かれたボタンを押すとその色に LED が光ります。また、加速度センサの値を常時グラフに表示しています。

Raspberry Pi Radicon Car

