

ラズパイでonedrive同期とカメラセットアップ

2020/12/8 市川 rev.2

概要

ラズベリーパイを使って定期的にonedriveに写真をアップロードするためのドキュメントです。大きく分けて下記の二つの内容を組み合わせて作成します。

- ①ラズベリーパイとonedriveの同期設定
- ②ラズベリーパイで定期的に写真を撮り保存

※2020/12/1現在raspberryPi OS 64bit版ではカメラが使えません。そのため、32bit版を前提とします。

64bitではonedriveの同期プログラムのインストール手順が異なりますのでgithubのページを参照ください。

今回使用するonedrive同期はこちらgithub: <https://github.com/abraunegg/onedrive>

github onedrive同期インストール方法:

<https://github.com/abraunegg/onedrive/blob/master/docs/INSTALL.md>

準備するもの

- ・初期セットアップ済みのラズパイ一式
- ・Raspberry Pi Camera Module V2
- ・マイクロソフトアカウント
- ・インターネット環境



onedriveのインストール

インストール

必要なパッケージをインストールします。

```
sudo apt-get install libcurl4-openssl-dev
sudo apt-get install libsqlite3-dev
sudo apt-get install libxml2
sudo apt-get install pkg-config
```

圧縮ファイルもダウンロードして解凍します。

```
wget
https://github.com/ldc-developers/ldc/releases/download/v1.16.0/ldc2-1.1
6.0-linux-armhf.tar.xz
tar -xvf ldc2-1.16.0-linux-armhf.tar.xz
```

解凍しましたので、圧縮ファイルは削除します。

```
rm ldc2-1.16.0-linux-armhf.tar.xz
```

インストールします。

インストールに関して質問がでた場合は「y」と入力し、「enter」を押します。

必要がなくなったファイルは削除します。

```
sudo apt install libnotify-dev
sudo apt autoremove
```

githubからダウンロードします。

```
git clone https://github.com/abraunegg/onedrive.git
```

ダウンロードしたファイルのディレクトリに移動して、構成を実行します。

```
cd onedrive
./configure DC=~/ldc2-1.16.0-linux-armhf/bin/ldmd2
```

ビルド作業を自動化するツールでmakeというものを作成します。
この処理には時間が掛りますので少し待ちます。

```
make clean; make
```

作成したmakeをインストールします。

```
sudo make install
```

onedriveの初期設定

onedriveを起動します。

```
onedrive
```

初期セットアップを行います。

表示されたURLにブラウザからアクセスし、ログイン認証します。

認証先のURLを貼り付けます。下記に詳しい写真がありますのでご参照下さい。

```
Configuring Global Azure AD Endpoints
```

```
Authorize this app visiting:
```

```
#①ここに表示されるURLにアクセス
```

```
https://login.microsoftonline.com/common/oauth2/v2.0/authorize?client_id  
=xxxxxxx/common/oauth2/nativeclient
```

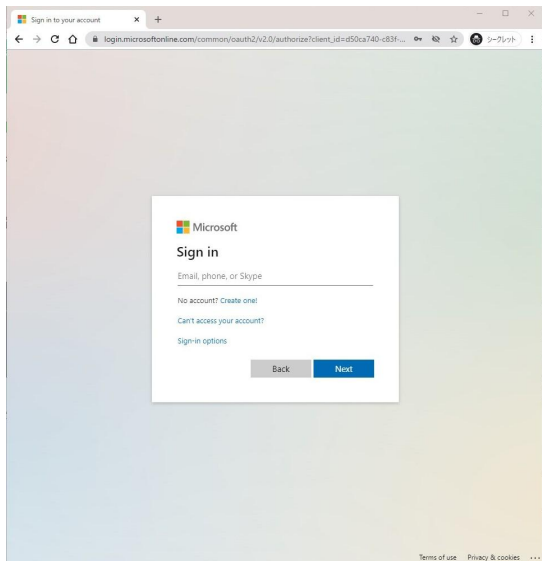
```
#②ここにURLを貼り付け
```

```
Enter the response uri:
```

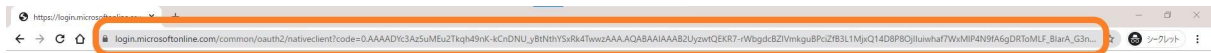
```
https://login.microsoftonline.com/common/oauth2/nativeclient?code=xxxxxx  
xxxx
```

```
Application has been successfully authorised, however no additional  
command switches were provided.
```

①の画面にアクセスし、ログインと許可をします。



②リダイレクトされたURLコピーして貼り付けます。



構成のテストをします。

```
onedrive --synchronize --verbose --dry-run
```

出力内容

```
Downloading file 異常時の写真/2020-11-17センサのコネクタが壊れた ... done.
Downloading file 異常時の写真/2020-11-17センサのコネクタが壊れた.jpeg ...
done.
Downloading file 異常時の写真/2020-11-18センサのコネクタが壊れた.jpeg ...
done.
Downloading file 異常時の写真/2020-11-18aa.jpeg ... done.
Downloading file 勤怠管理.xlsx ... done.
Uploading differences of ~/OneDrive
Processing .
The directory has not changed
Processing Notebooks
The directory has not changed
Processing 異常時の写真
The directory has not changed
Processing 勤怠管理.xlsx
The file has not changed
```

実際に同期を実行します。

```
onedrive --synchronize
```

システムサービスとして有効にします。

ラズベリーパイ起動時にonedriveが自動起動するようになります。

```
systemctl --user enable onedrive  
systemctl --user start onedrive
```

homeディレクトリに戻り、完了です。

```
cd ~
```

同期のログを確認するには下記のコマンドを実行します。

ctrl+cで確認を終了できます。

```
journalctl --user-unit=onedrive -f
```

```
-- Logs begin at Tue 2020-12-01 20:41:53 JST. --  
12月 01 21:04:20 raspi systemd[719]: Started OneDrive Free Client.  
12月 01 21:04:20 raspi onedrive[3985]: Configuring Global Azure AD  
Endpoints  
12月 01 21:04:21 raspi onedrive[3985]: Initializing the Synchronization  
Engine ...  
12月 01 21:04:21 raspi onedrive[3985]: Initializing monitor ...  
12月 01 21:04:21 raspi onedrive[3985]: OneDrive monitor interval  
(seconds): 300
```

raspiカメラの設定

カメラの接続

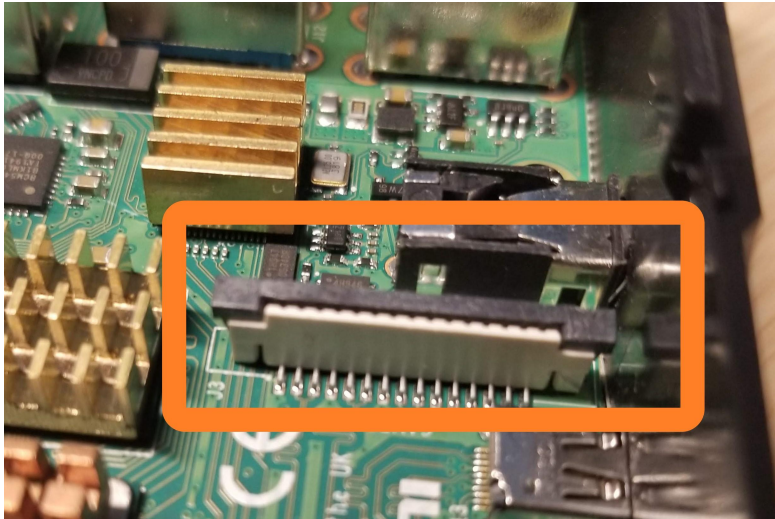
カメラを接続する前にラズパイの電源が入っている場合は落とします。

```
shutdown now
```

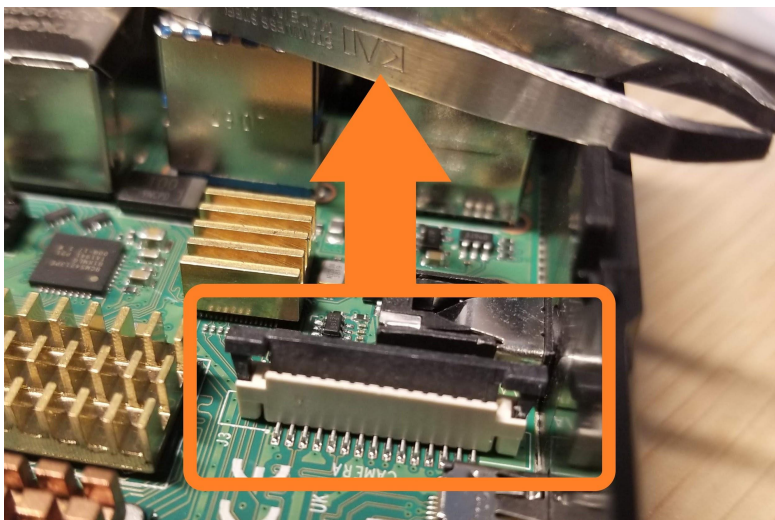
シャットダウン完了しましたら、コンセントから電源ケーブルを抜きます。

カメラを接続方法

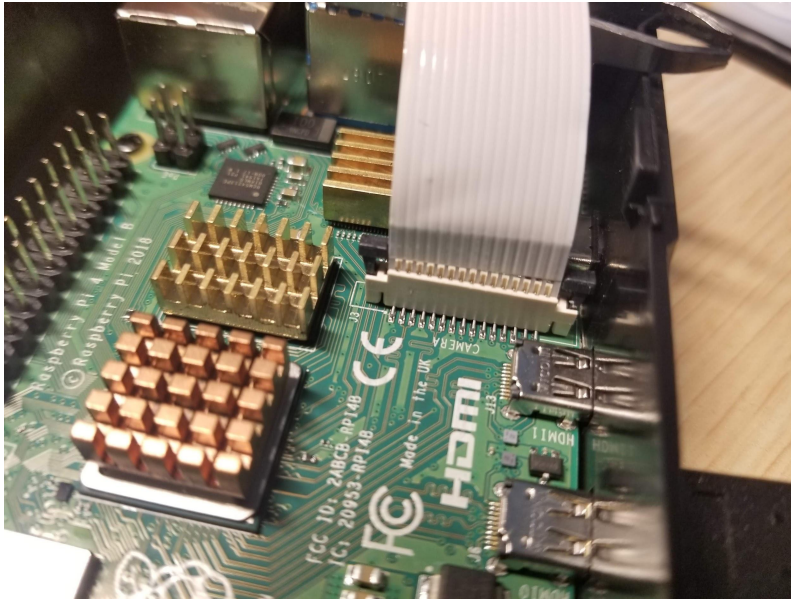
オレンジで囲まれた箇所がカメラの接続端子です。



接続端子が閉じていますので、持ち上げてカメラが接続できるようにします。

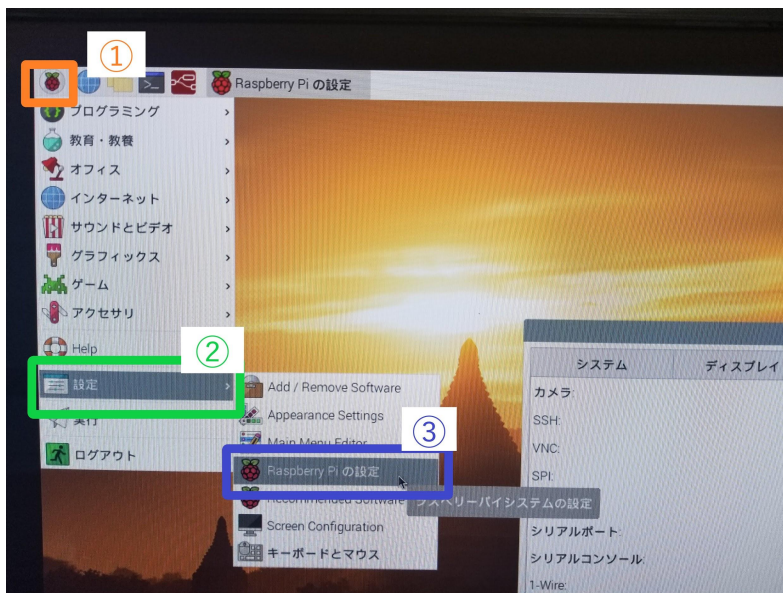


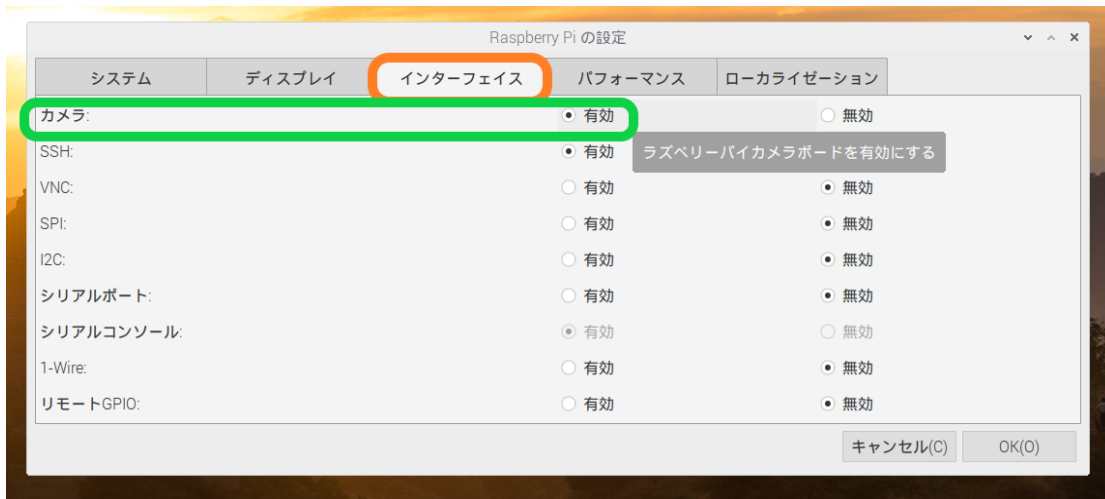
カメラを差し込み、先ほど持ち上げた部分を押し戻します。



カメラの有効化

ラズパイを起動し、カメラを有効にします。





カメラの認識確認

カメラが正常に接続されているか確認します。

```
vcgencmd get_camera
```

下記のように出力されれば正常に接続が来ています。

```
supported=1 detected=1
```


撮影のプログラム

/home/pi/にcamera.pyを作成します。
nanoエディタはctrl+sで保存、ctrl+xで終了できます。

```
nano camera.py
```

ファイルの中身はこちらをコピーしてください。

```
import datetime
from picamera import PiCamera
from time import sleep

now = datetime.datetime.now()
camera = PiCamera()
camera.resolution = (1920,1080)#カメラサイズの設定します
camera.rotation = 180 #カメラの向きを調整します 0 or 90 or 180 or 270
camera.start_preview()#alpha=230)#プレビューalphaの値で透過率をいじることができます
sleep(5)#5秒待機
camera.capture('/home/pi/OneDrive/raspicamera/image%s.jpg' %
now.strftime('%Y%m%d_%H%M%S'))#保存先を指定し保存します。
camera.stop_preview()
```

保存先のフォルダをonedrive同期フォルダ内に作成します。

```
cd OneDrive
mkdir raspicamera
```

ピントの調整

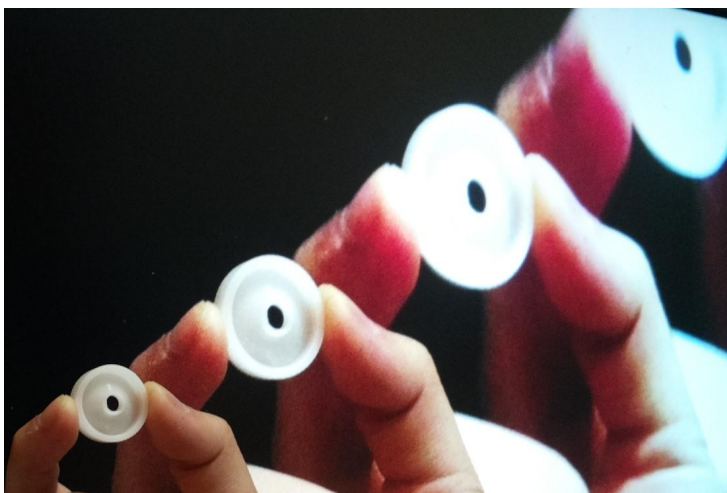
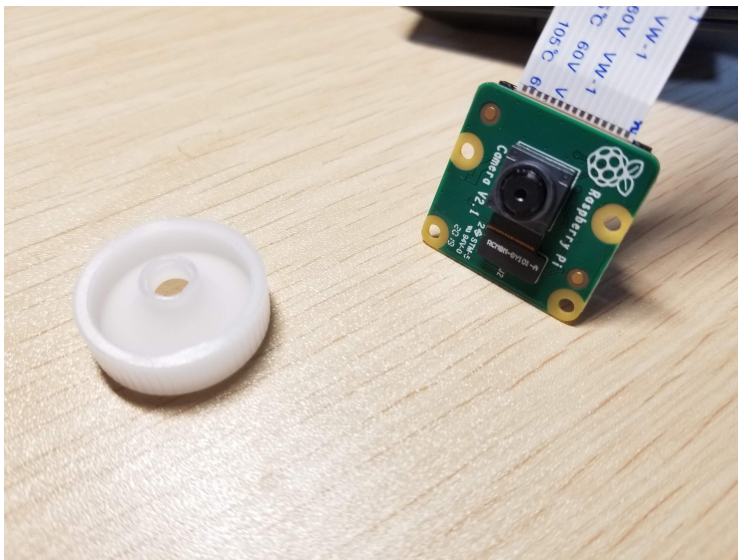
実際に使用する際にはピントを手動で調整してあげる必要があります。
付属の白い器具のしぼんだ部分をカメラにはめ、回転させることでピントが変わります。
上記のプログラムを「sleep(5)」を「sleep(60)」のように変更し、実行します。
ピント調整後は「sleep(5)」に戻します。

nanoエディタで編集

```
nano /home/pi/camera.py
```

pythonプログラムを実行します。プログラムを実行している間にピントを調整しましょう。

```
python /home/pi/camera.py
```



定期実行の設定

rootユーザーに移動して定期実行の設定ファイルを作成します。

```
sudo su -  
sudo mkdir bin  
cd bin  
sudo nano cronfile
```

crofileの中身は下記のようにします。

五分おきに先ほどのプログラムを実行することができます。

```
*/5 * * * * python /home/pi/camera.py >/dev/null 2>&1
```

cronの書き方

定期実行するために上記でcronを使いました。設定方法はこちらになります。

特に変更しなければ、飛ばして下さい。

* * * * * 実行内容

左から順に

分[0-59] 時[0-23] 日[1-31] 月[1-12] 曜日[0-7]「実行内容」となります。

複数指定 ⇒ カンマで区切ります。「0,30」

範囲指定 ⇒ ハイフンを使用します。「8-17」

間隔 ⇒ 右記のように記載します。「*/10(10分間隔)」

00,30 8-18 * * 5 毎週金曜日に8-18時の間 00分と30分に実行のように書きます。

*/10 8-18 * * 1-5 10分間隔で 8時から18時 月から金曜日

python /home/pi/camera.py >/dev/null 2>&1はpython実行 エラー出力を標準出力とマージンして捨てるという処理になります。

定期実行(cron)に登録

先ほど作成したファイルを登録します。

```
crontab cronfile
```

実際に登録されているか確認します。

```
crontab -l
```

出力内容が先ほど作成したcronfileの中身と同じであれば設定完了です。

```
*/5 * * * * python /home/pi/camera.py >/dev/null 2>&1
```

撮影されるか確認しましょう。SSHではない場合、急に画面が画像におおわれると思います。

定期実行時にカメラの画像が見えると作業の邪魔になるので、完全に透けるように変更しましょう

nanoエディタで編集

```
nano /home/pi/camera.py
```

編集箇所

```
camera.start_preview(alpha=0)
```

これで設定完了です。