Spectrum Technology

V1.0 2019/9/21

はじめてのロボット・カー開発キット

～GoPiGo3を使い、自動走行から音声制御による走行まで～

実践編（抜粋版）

GoPiGo
Build & program your own robot.

DEXTER INDUSTRIES

Raspberry Pi

Google Cloud Platform

amazon alexa

スペクトラム・テクノロジー株式会社

https://spectrum-tech.co.jp

sales@spectrum-tech.co.jp

Spectrum Technology

ロボット・カー開発キット　目次

抜粋版のため、ページと実際は違います

ロボット・カー開発キット　目次

抜粋版のため、ページと実際は違います

ロボット・カー開発キット　目次

抜粋版のため、ページと実際は違います

Pi運用マニュアル

Raspberry Pi

① Raspberry Piについて

既に全世界で1000万台以上販売された手のひらサイズのコンピュータです。LinuxベースのRaspbianOSで動作しております。今回、GoPiGoのCPUとして使用しています。

② Linux基本コマンド

A) システム関係

- 起動：電源を入れると自動で起動します。
- 再起動：# reboot
  又は、アプリケーション>ログアウト>再起動；左上のメニューから
- 終了： # shutdown
  又は、アプリケーション>ログアウト>シャットダウン；左上のメニューから
- ログアウト # logout
  又は、アプリケーション>ログアウト>ログアウト；左上のメニューから
- 日本語／英語の入力切替：キーボードのctl+jを同時に押します。又は右上のアイコン（右から7個目）からプルダウンで選択

Pi運用マニュアル

Raspberry Pi

② Linux基本コマンド

B) ディレクトリ操作、コピー、移動、削除

root@:~# cd /root/Documents　　ディレクトリの切り替え
root@:/root/Documents# ls　　ファイルとディレクトリの表示（表示したら操作したいファイルを右クリックでコピペして操作します
root@:~# cp ファイル名　ディレクトリ　　配下のディレクトリのファイルを別のディレクトリへコピー
root@:~# mv ファイル名　ディレクトリ　　配下のディレクトリのファイルを別のディレクトリへ移動
root@:~# rm ファイル名　　ファイルの削除
便利な機能　rm –help　　コマンドのオプションが分からない場合は、ヘルプで問い合わせる。すべてのコマンド共通(マイナスを2個とhelp)

C) ユーザ権限、プロセス他

root@:~ $ su -　　スーパーユーザ（root)に切り替え、パスワードを入力
root@:~# ps a　　現状の動いているプロセスを表示
root@:~# kill　　特定のプロセスを強制終了
root@:~# apt-get install pkg　　パッケージのインストールなどに使用
root@:~# date　　日付、時間の設定を行います。
root@:~# leafpad /etc/network/interfaces　　インタフェースに記述してい内容を変更します。Viよりも使いやすいです。

D) モジュール、usb、メモリ、HDDなどの表示

root@:~# lsmod　　linuxのモジュールリスト表示
root@:~# lsusb　　usbのデバイス表示
root@:~# free -mt　　メモリ使用状態表示
root@:~# df -h　　HDD（マイクロSD）の使用状態表示

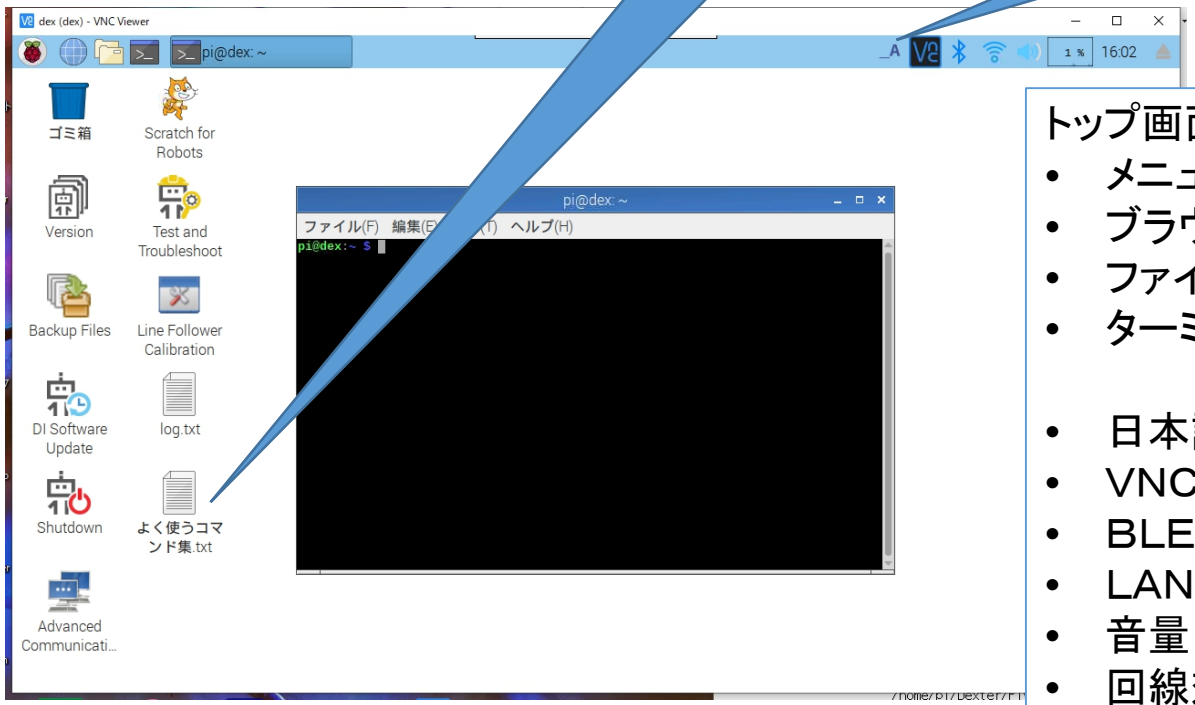
Pi運用マニュアル

③ 基本操作

A) 表示画面と内容

トップ画面によく使うコマンド.txtをアップしてます

Anthyが出ない場合は、一度Japaneseを選択後、再度Anthyを選択してください。



トップ画面（上段のタスクバーで選択）

- メニュー
- ブラウザ
- ファイルマネジャ
- ターミナル

- 日本語入力
- VNC
- BLE
- LAN／WiFi
- 音量
- 回線効率
- 時刻

GoPiGo基本操作

Spectrum Technology

GoPiGo基本操作
①. GoPiGo　概要

- GoPiGoは、Dexter社が、小学生から大学・社会人まで幅ひろく、ロボットを使ってプログラム習得することを目的に開発されました。既に、数多く導入され活用されています。火星探査プロジェクトでも有名です。
- 特徴として、カメラ、距離センサ、Lineセンサ、温湿度センサなど数多くのセンサがあり、そのプログラムも公開されています。
- プログラム言語もScratch，Python，Java、Cなど準備されており、OSとしては、Raspbian，DEX　OSなど利用用途により使い分けできます。初心者から上級者まで幅広く対応できます。
- https://www.dexterindustries.com/gopigo3/

GoPiGo基本操作
②. GoPiGo ソフトウエアー覧

本開発キットのインストールしているソフトウエアの概要です。

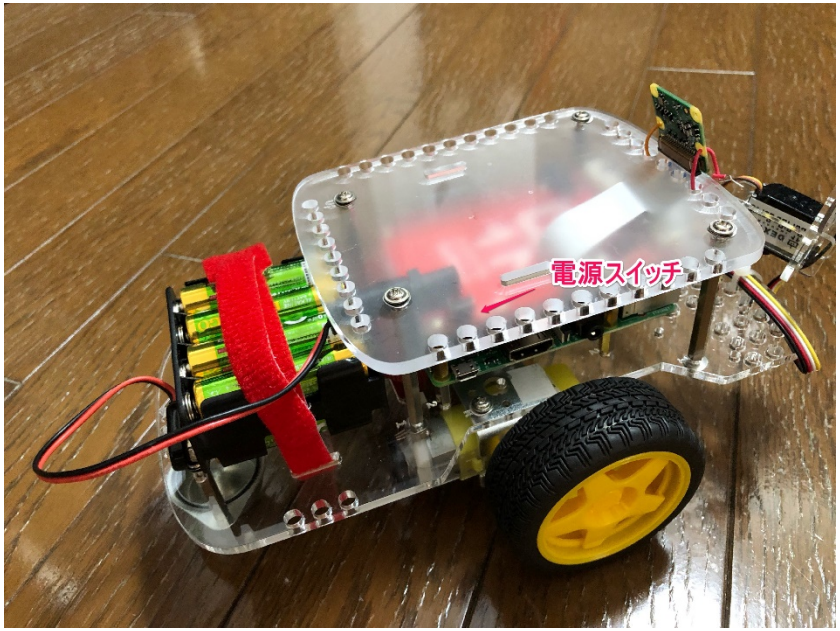
区分	ソフト名	バージョン	備考
OS	dexter	v9	Raspbian stretch version
プログラム言語	python3	3.5.3	
	python2	2.7.13	
	scratch	1.4	
画像	Opencv	3.4	
	picamera	1.13	
	yolo	yolov3	ローカルで動作
	google-cloud-vision		グーグル・クラウド
音声（Alexa）	AVS(alexa voice service) device sdk	V1.13	
その他	Pkg, Pip関係プログラム多数		

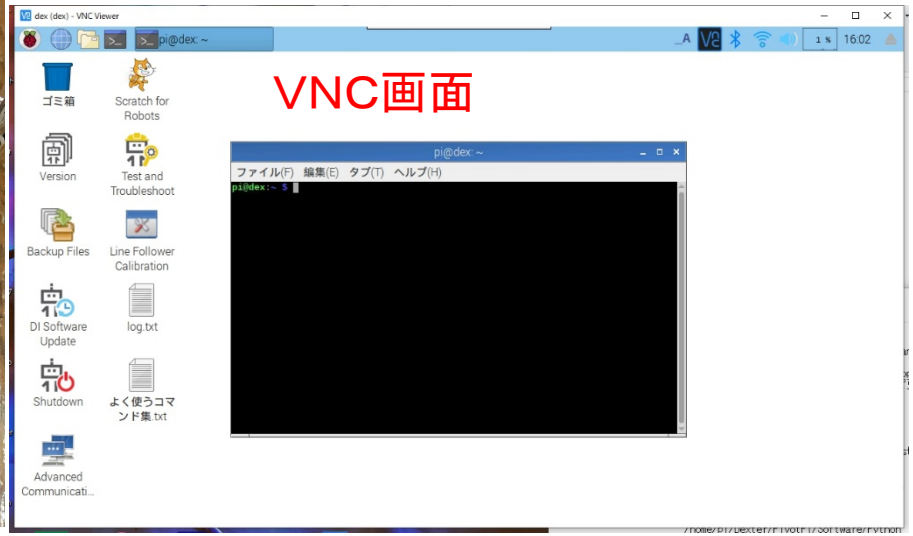
Spectrum Technology

GoPiGo基本操作
④. GoPiGo起動

- GoPiGoの基本部、カメラなどの組み立て、取付が完了しましたら、電源をONにして起動します。
- Teraterm，VNC ViewerでPC等からアクセスします。

Teraterm
IPアドレス:192.168.1.xx
ユーザ名：pi
パスフレーズ：robots1234
VNC接続
パスワード：設定したもの



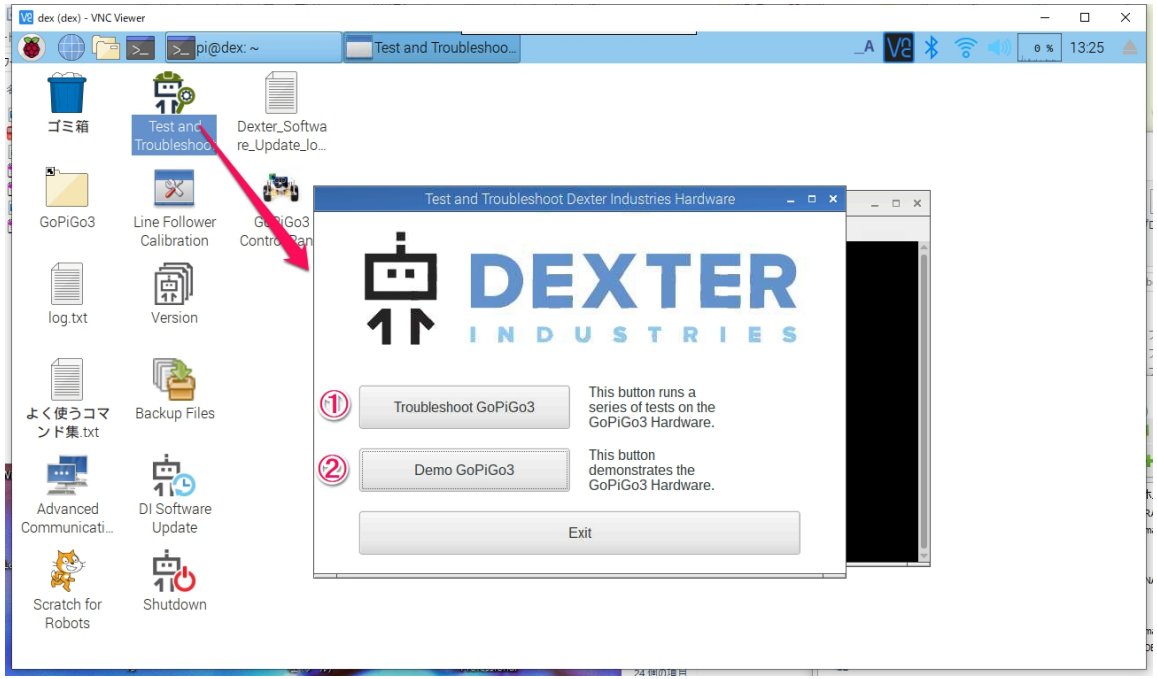


GoPiGo基本操作
⑥. GoPiGo3　テスト＆デモ

- GoPiGo3のテストとデモを行います。

① ハードウエアのテストを行います。

② デモを行います。動き出しますので、注意して実施してください。



Spectrum Technology

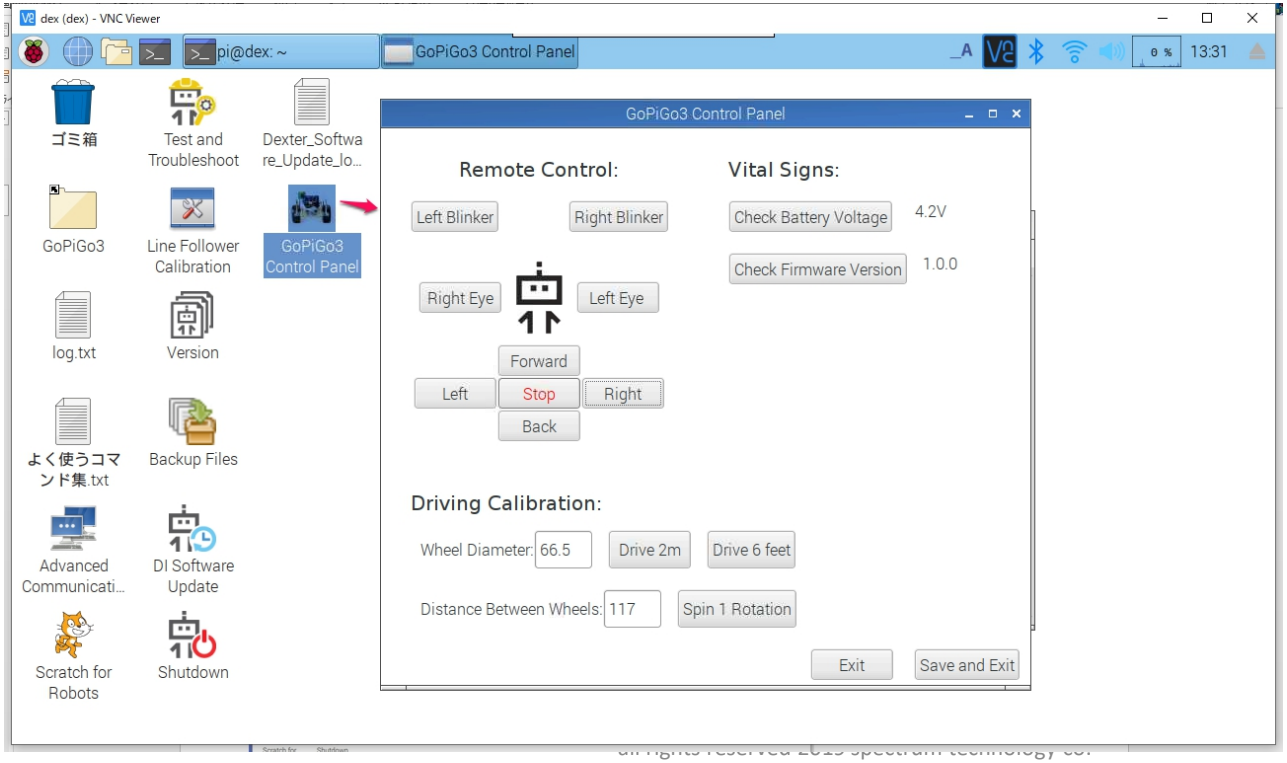
GoPiGo基本操作
⑦. GoPiGo3　コントロールパネル

- GoPiGo3のコントロールパネル。

① LEDライトの点灯。

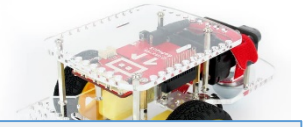
② ロボットの動作（前進、後退、左右）

③ 運転補正、ファームウエアなどの確認など



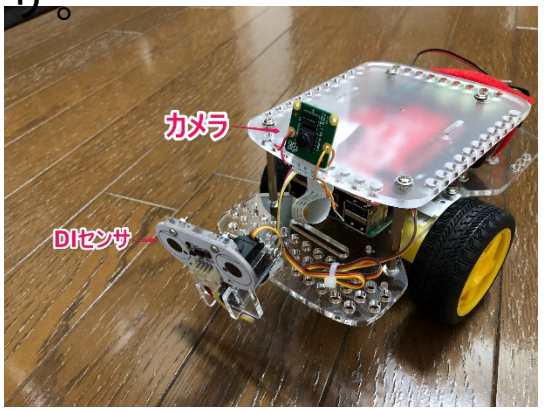
Spectrum Technology

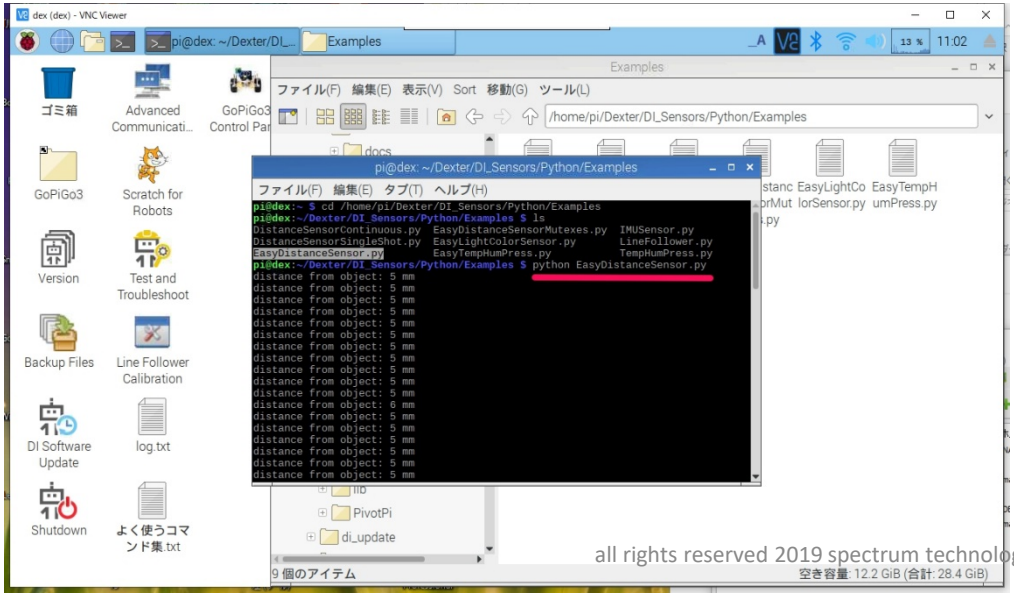
GoPiGo基本操作
⑧. GoPiGo　各種センサ類試験
A). DIセンサ

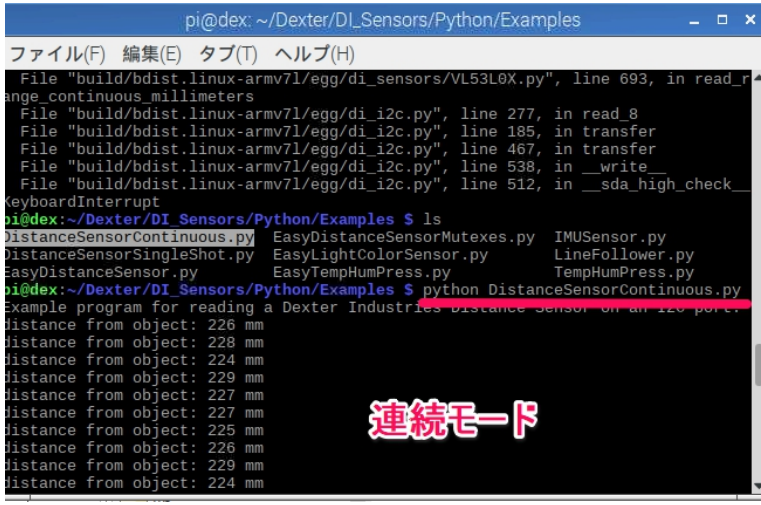


```
$ cd
/home/pi/Dexter/DI_Sensors/Pyth
on/Examples
$ python EasyDistanceSensor.py
```

- Distanceセンサ：センサから障害物までの距離を測るセンサ。
- https://di-sensors.readthedocs.io/en/master/index.html
- cd /home/pi/Dexter/DI_Sensors/Python/Examples
- python EasyDistanceSensor.py
- python DistanceSensorContinuous.py　　連続モード
- python DistanceSensorSingleShot.py　　シングルモード



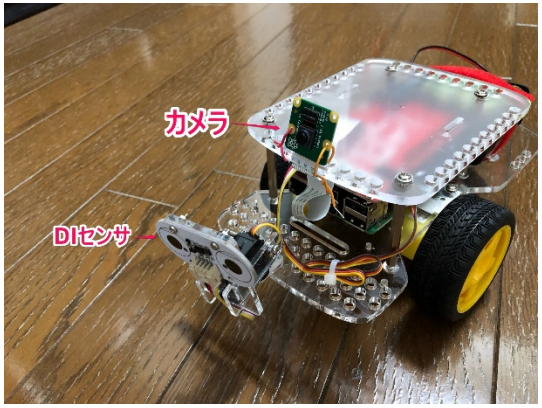


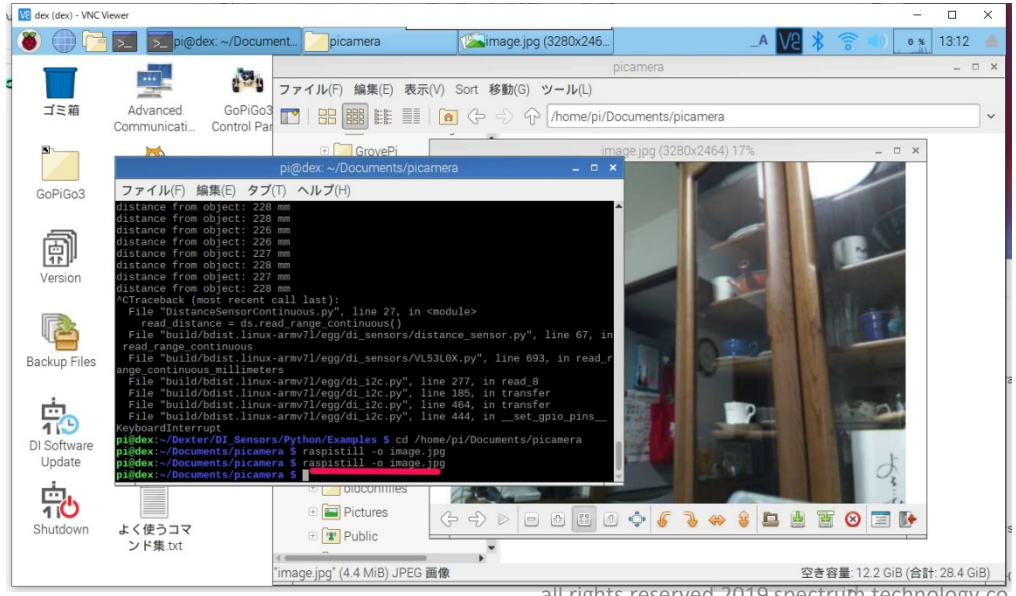


GoPiGo基本操作
⑧. GoPiGo　各種センサ類試験
B). カメラ(写真)

- カメラ:写真及び動画撮影できます。
- https://picamera.readthedocs.io/en/release-1.13/
- cd /home/pi/Documents/picamera
- raspistill -o image.jpg　写真
- python picam_img.py　test1.jpgで出力

```
$ cd
/home/pi/Documents/picamera
$ raspistill -o image.jpg
$ python picam_img.py
```



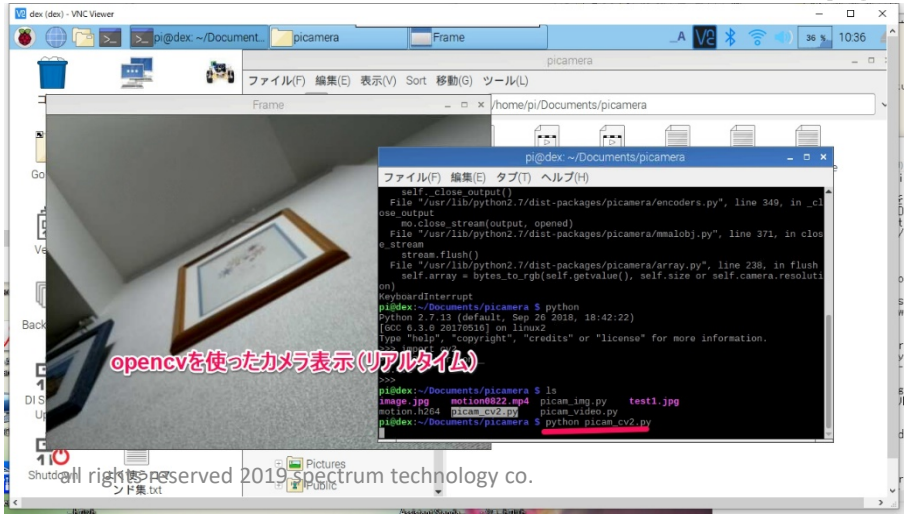


Spectrum Technology

GoPiGo基本操作
⑧. GoPiGo　各種センサ類試験
D). カメラ(リアルタイム動画)

```
$ cd
/home/pi/Documents/picamera
$ python picam_cv2.py
```

- カメラ:写真及び動画撮影できます。
- https://picamera.readthedocs.io/en/release-1.13/
- cd /home/pi/Documents/picamera
- python picam_cv2.py　リアルタイム動画表示
- 画面サイズ:640x480、Opencvを使って表示
- Opencvは、インテルが開発した画像用のプログラムで、顔認識、物体認識には必須です。一度インストールすると簡単にpythonで動作できます。

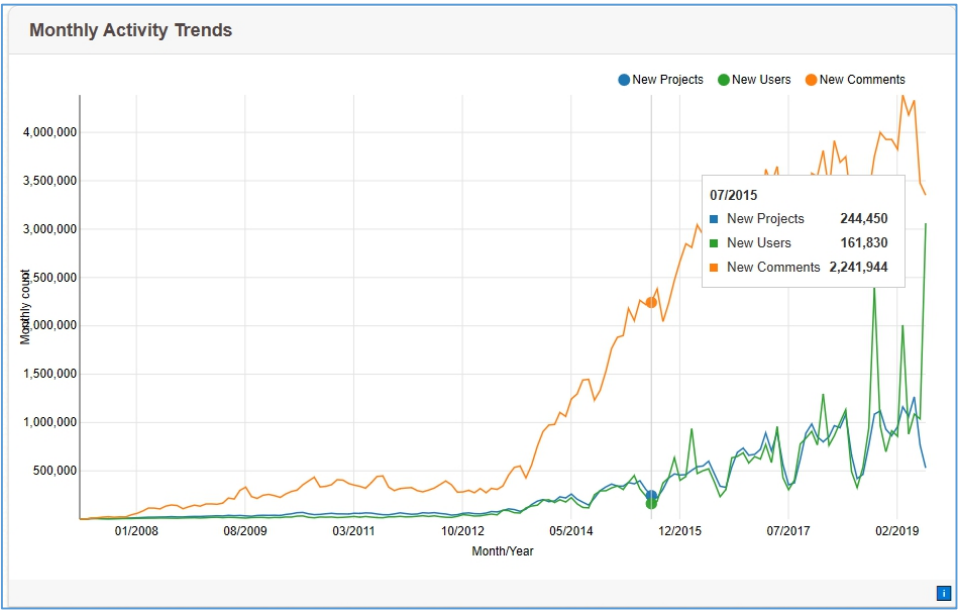


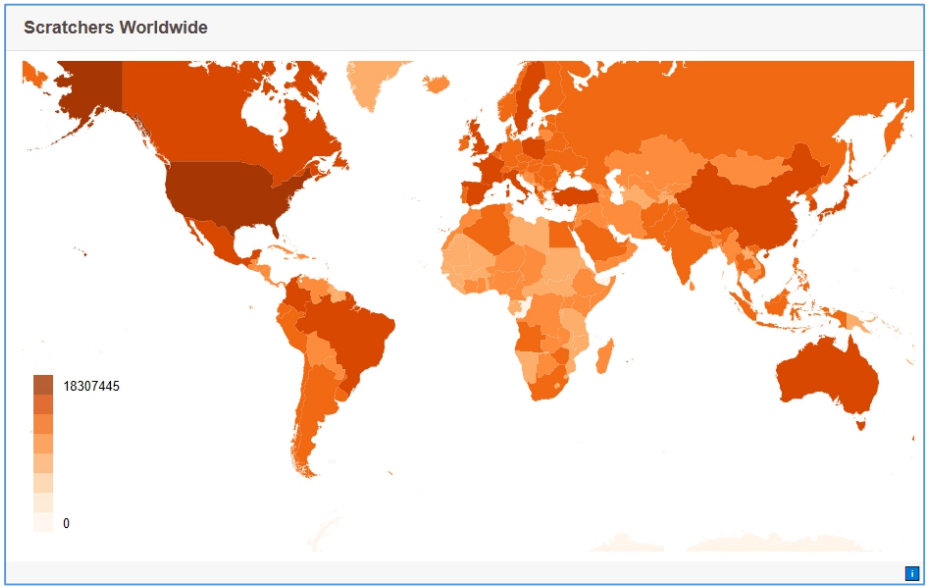
Scratch学习

Spectrum Technology

Scratch学習
①. Scratch概要

- Scratchについて
- MITで開発した初心者向けのプログラミング言語。月間1百万ユーザ、グローバルに利用
- https://scratch.mit.edu/



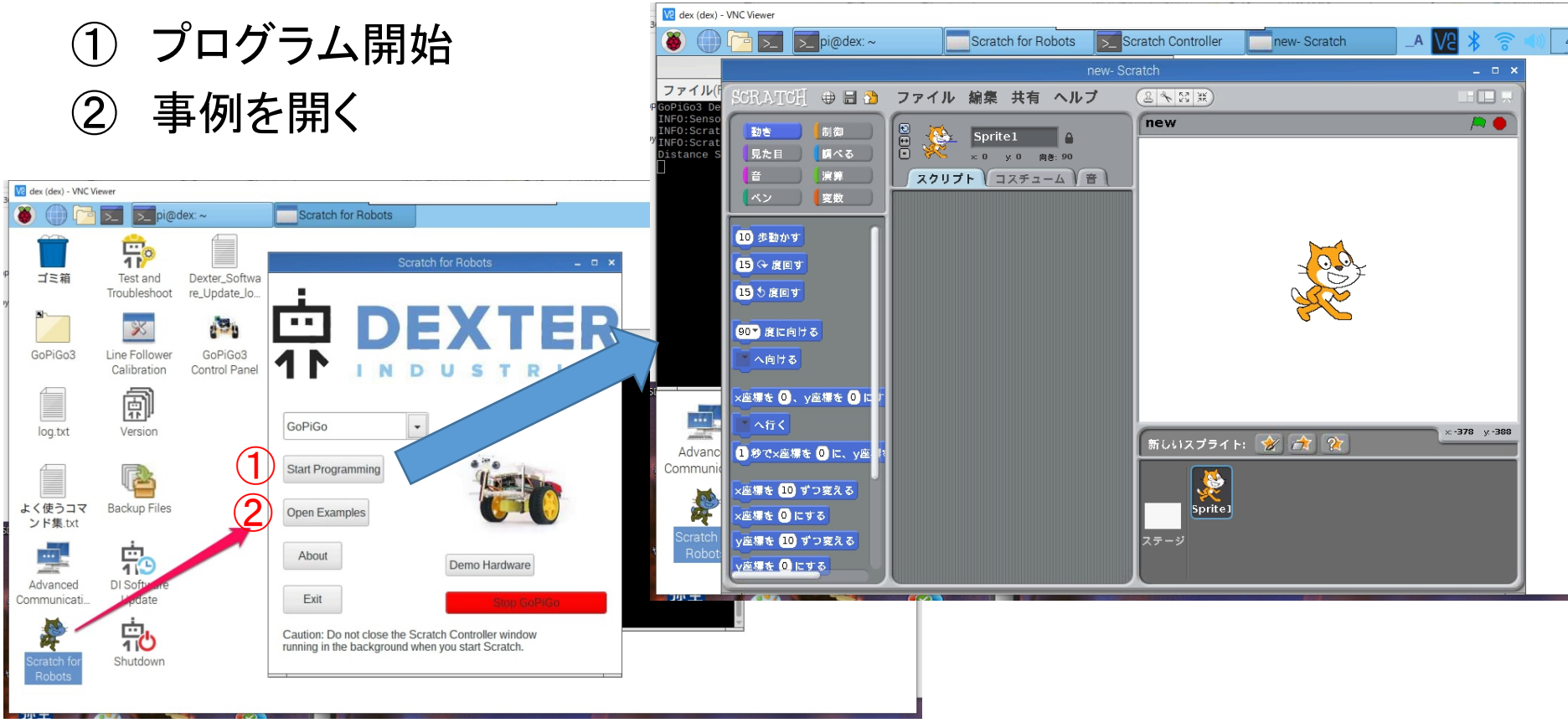


②. Scratchプログラム学習
基本操作

- Scratch言語を使ったRobotの動作
- https://scratch.mit.edu/

① プログラム開始

② 事例を開く

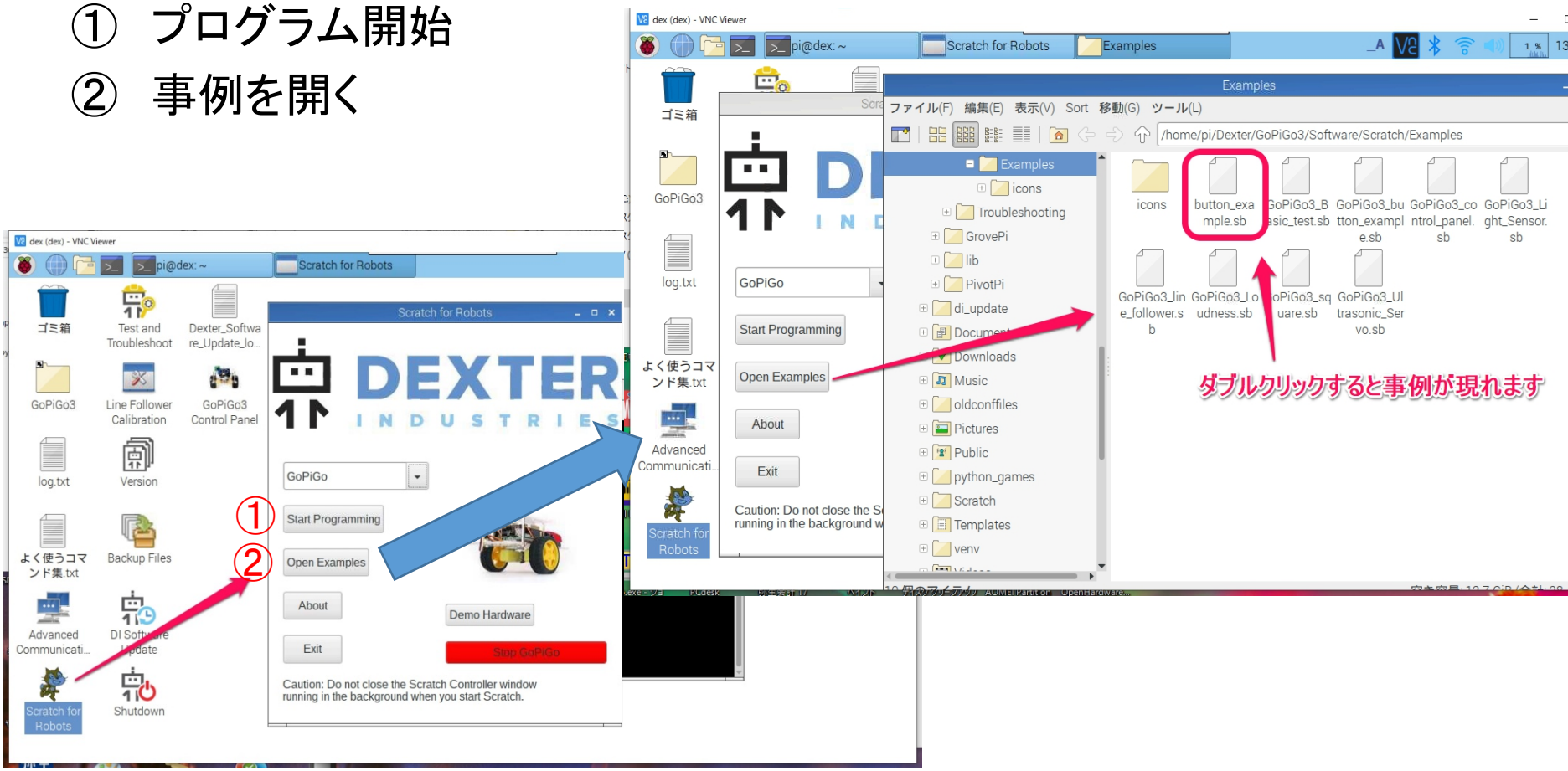


②. Scratchプログラム学習
基本操作

- Scratch言語を使ったRobotの動作

① プログラム開始

② 事例を開く



②. Scratchプログラム学習

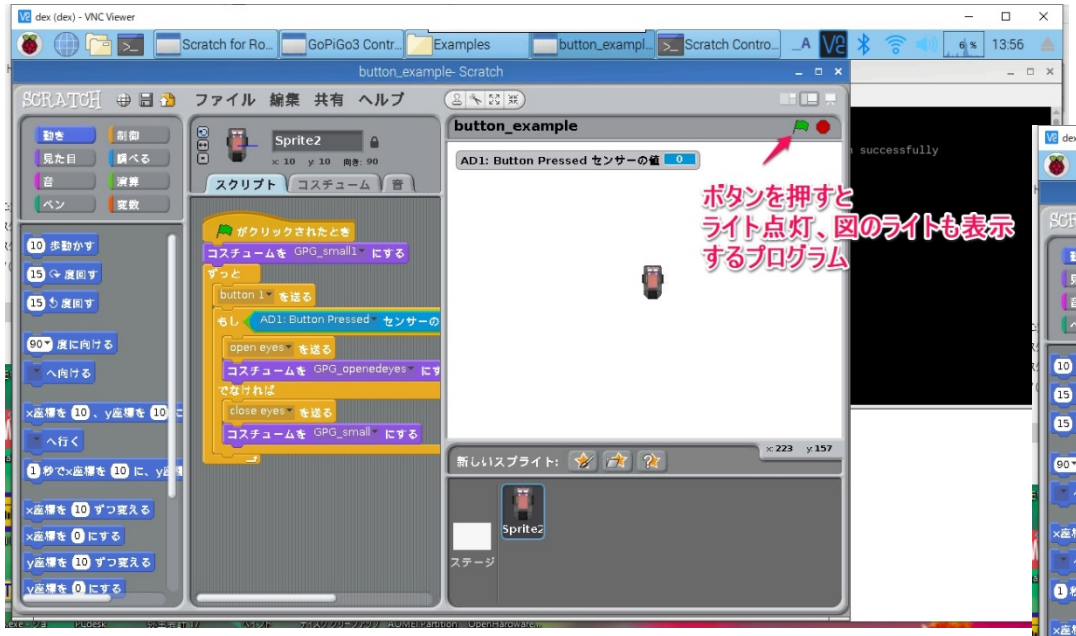
ライト点灯事例

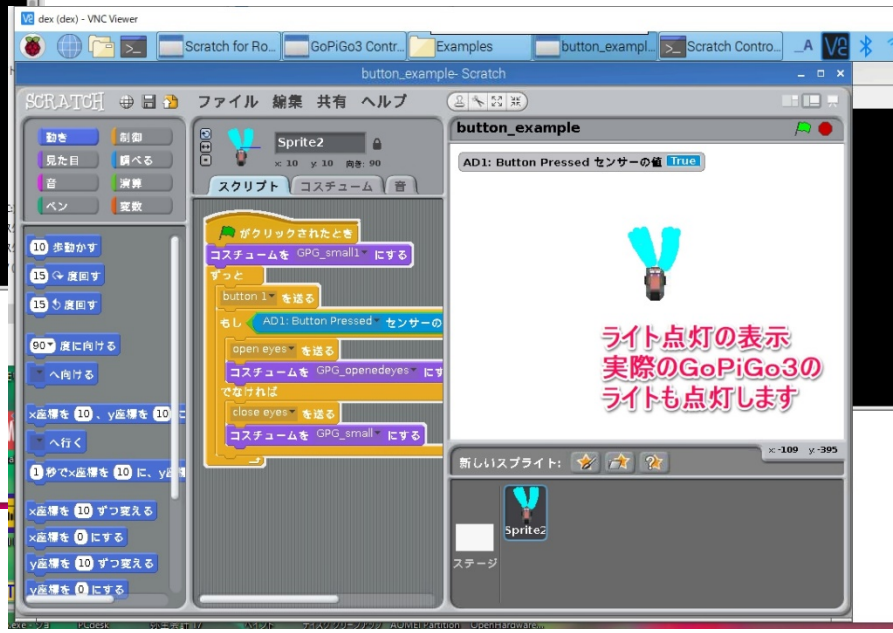
- Scratch言語を使ったRobotの動作

① プログラム開始

② 事例を開く(ライト点灯事例)

```
$ cd
/home/pi/Dexter/GoPiGo3/Software/Scratch/Examples
button_example.sb
```





②. Scratchプログラム学習

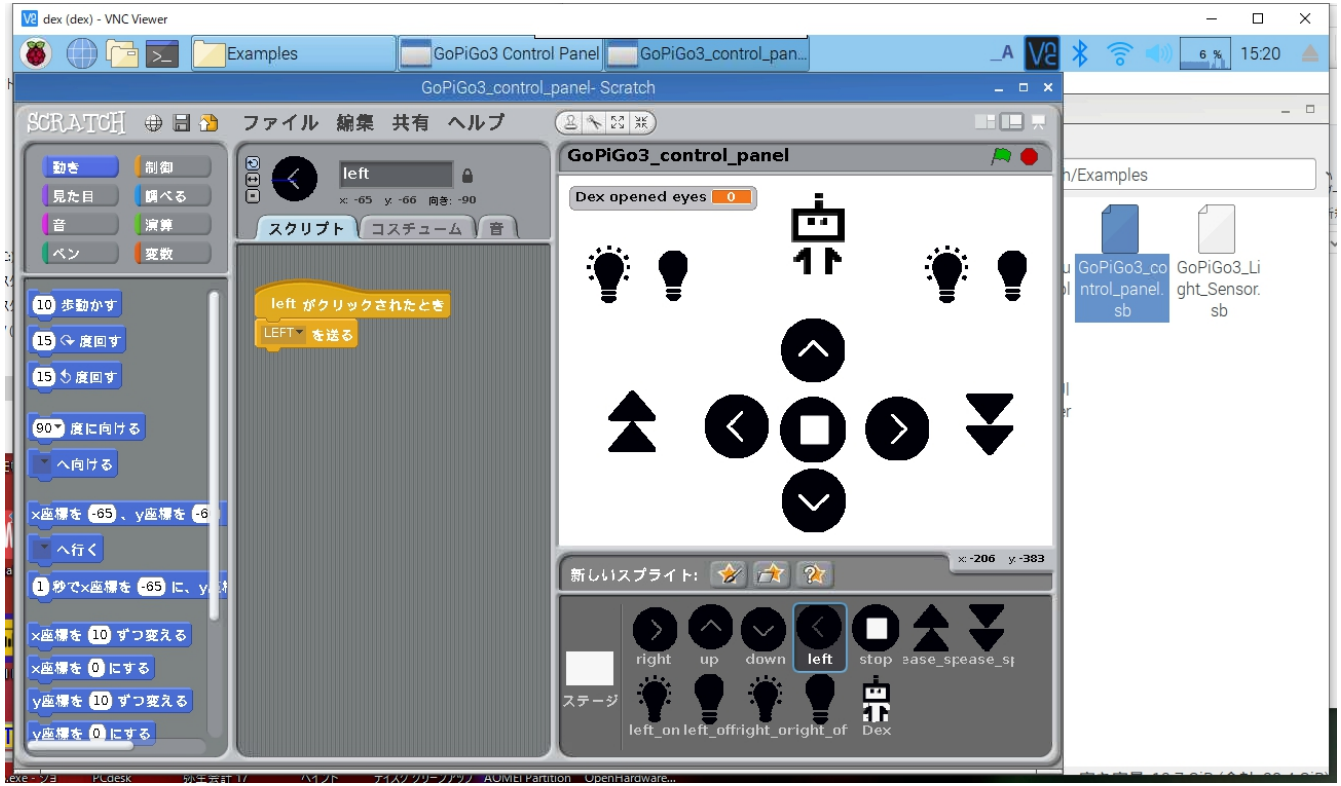
コントロールパネル事例

- Scratch言語を使ったRobotの動作

① プログラム開始

② 事例を開く(コントロールパネル事例)

```
$ cd
/home/pi/Dexter/GoPiGo3/Software/Scratch/Examples
GoPiGo3_control_panel.sb
```



②. Scratchプログラム学習

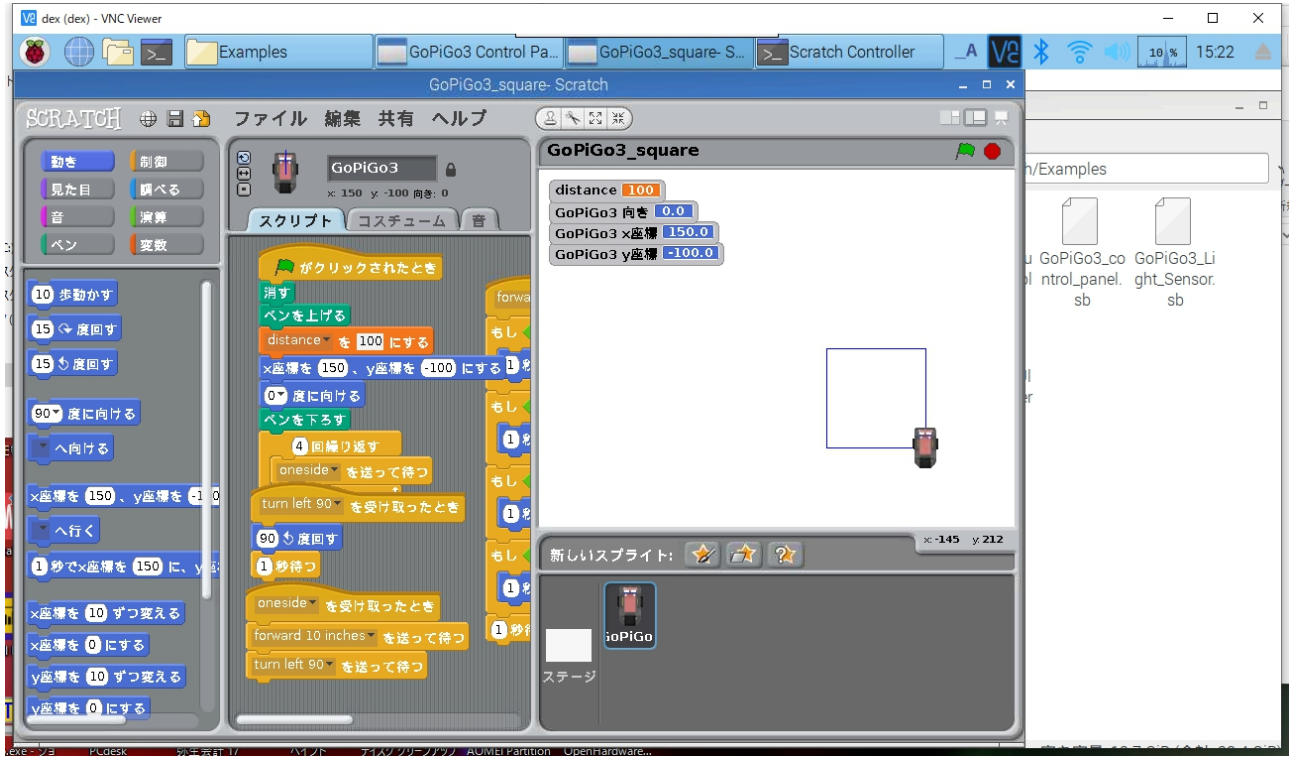
正方形動作事例

- Scratch言語を使ったRobotの動作

① プログラム開始

② 事例を開く（正方形動作事例）

```
$ cd
/home/pi/Dexter/GoPiGo3/Software/Scratch/Examples
GoPiGo3_square.sb
```



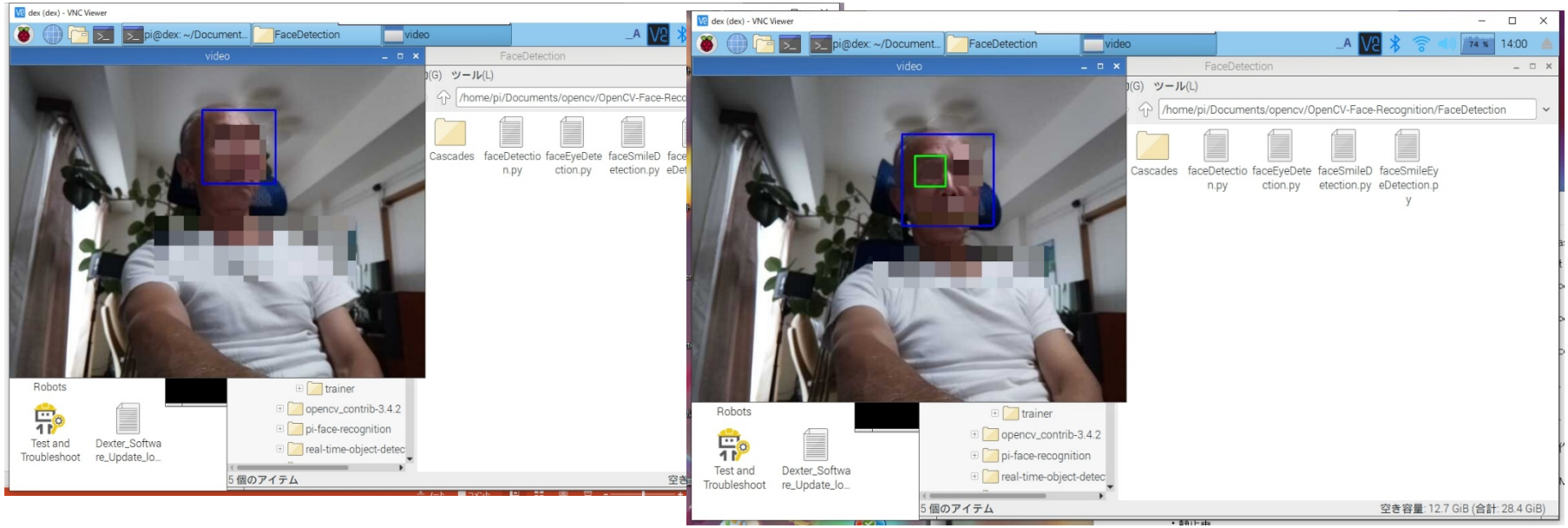
ST Spectrum Technology

画像認識開発

画像認識　開発
①. Opencv
A). 顔認識（カメラ）

コマンド
$ cd /home/pi/Documents/opencv/OpenCV-Face-Recognition/FaceDetection

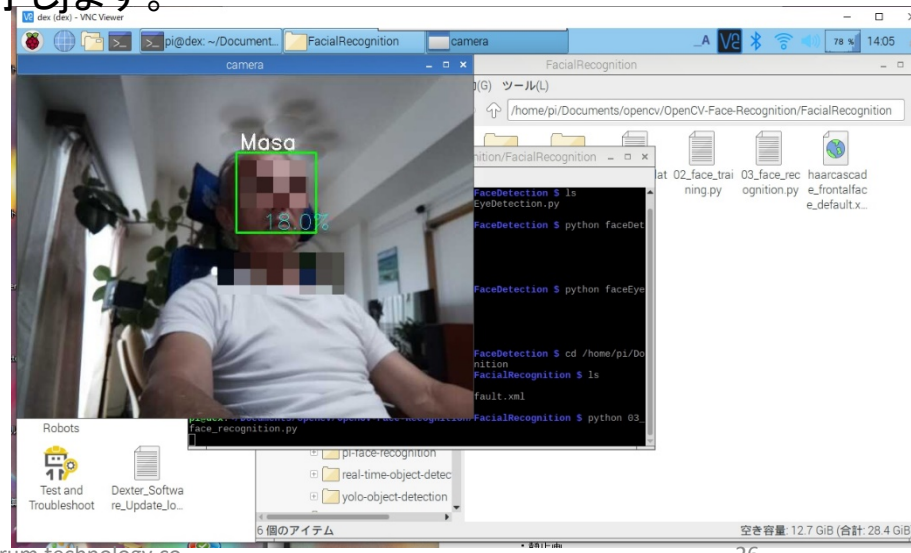
- Opencvを使ったwebカメラでの顔認識です。
- https://www.instructables.com/id/Real-time-Face-Recognition-an-End-to-end-Project/
- python faceDetection.py
- python faceEyedetection.py　　他。CPUのみで動作のため遅い。



画像認識　開発
①. Opencv
B). 顔検出（カメラ）

コマンド
$ cd /home/pi/Documents/opencv/OpenCV-Face-Recognition/FacialRecognition

- Opencvを使った、webカメラでの登録した顔を検出します。
- https://www.instructables.com/id/Real-time-Face-Recognition-an-End-to-end-Project/
- 顔の登録
- python3 01_face_dataset.py
  - Face id：0-3までを入力してください。
  - 自動で顔のスキャンが始まります。数分で終了しjます。
- python3 02_face_training.py
- python3 03_face_recognition.py
  - カメラで認識します
  - 名前と確度がでます。
  - 26行目のface id=1をMasaにしてます。変更
  - してください。

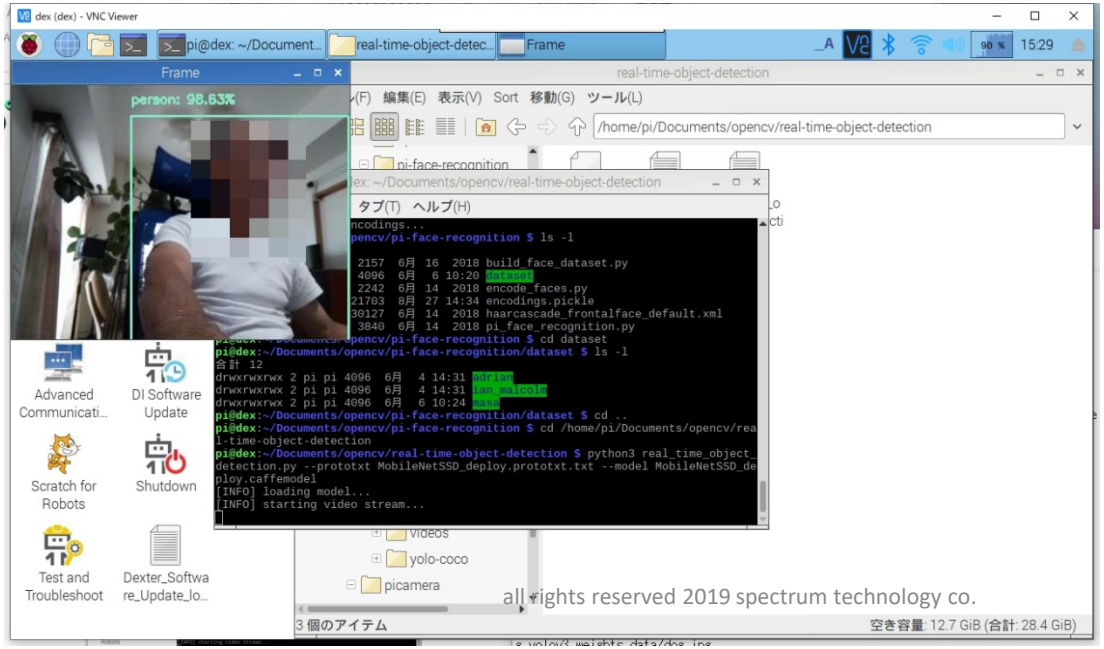


Spectrum Technology

画像認識　開発
①. Opencv
D). 物体検出（カメラ）

コマンド
$ cd /home/pi/Documents/opencv/real-time-object-detection

- https://www.pyimagesearch.com/2017/09/18/real-time-object-detection-with-deep-learning-and-opencv/
- 物体検出（カメラ）
- python3 real_time_object_detection.py --prototxt MobileNetSSD_deploy.prototxt.txt --model MobileNetSSD_deploy.caffemodel
- かなり重い



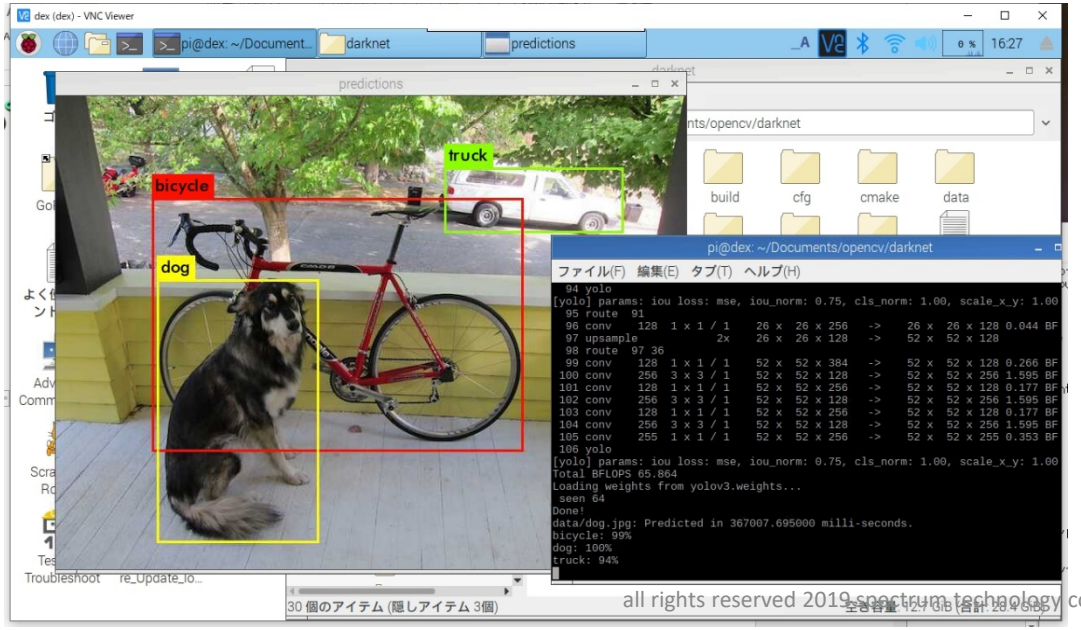
python3のみで動作

Spectrum Technology

画像認識　開発
②. Yolo
A). 物体検出（写真）

コマンド
$ cd /home/pi/Documents/opencv/darknet

- Darknetとして有名なYoloです。基本c言語で書いたものをpython3でも使える用になってます。
- https://pjreddie.com/darknet/yolo/
- 一番基本の写真の例です。
- ./darknet detect cfg/yolov3.cfg yolov3.weights data/dog.jpg
- 表示まで5分位かかります。動画は、GPU付きのもので行ってください。



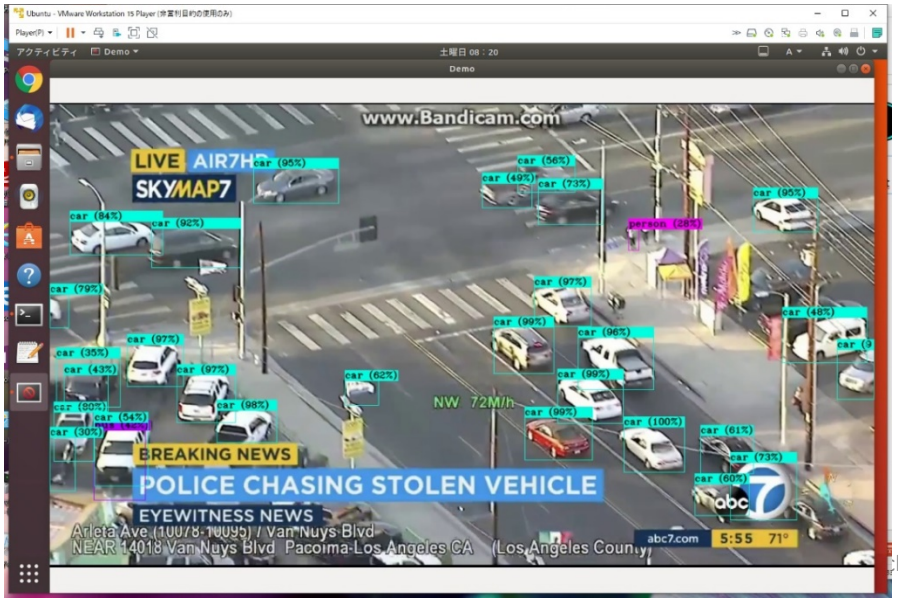
自転車、車、犬を検出

画像認識　開発
②. Yolo
C). 物体検出（ビデオ）

コマンド
$ cd /home/pi/Documents/opencv/yolo-object-detection

- https://www.pyimagesearch.com/2018/11/12/yolo-object-detection-with-opencv/
- 物体検出（ビデオ）
- python3 yolo_video.py --input videos/car_chase_02.mp4 --output output/car_chase_02.avi --yolo yolo-coco
- GPUを使用しないと動きません「はじめての顔認識・物体認識用AI開発キット」を購入ください

python3のみで動作

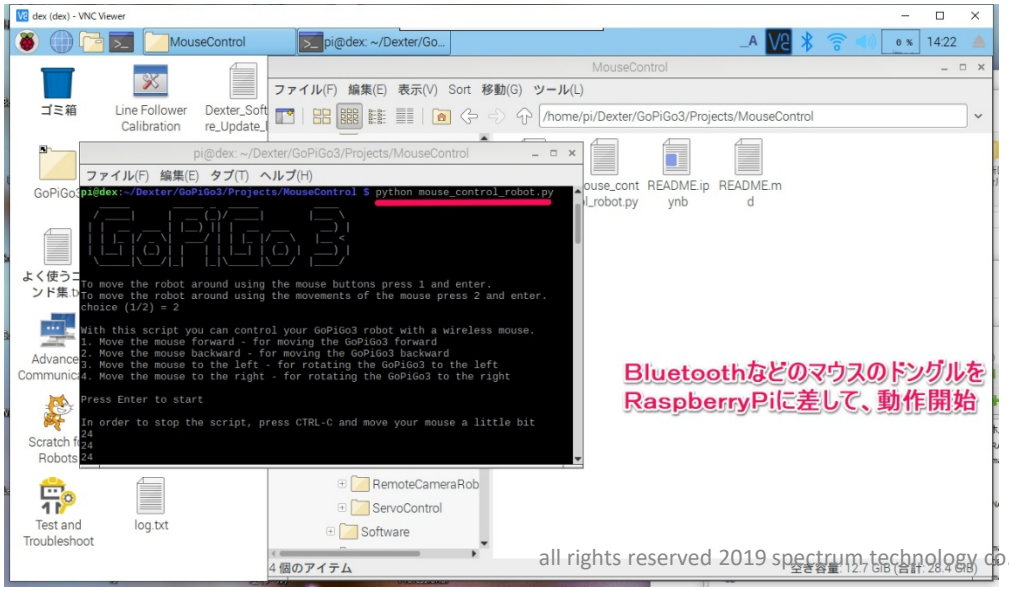


応用例

応用例
②. マウスコントロール

コマンド
$ cd
/home/pi/Dexter/GoPiGo3/Projects/MouseControl
$ python mouse_control_movement.py

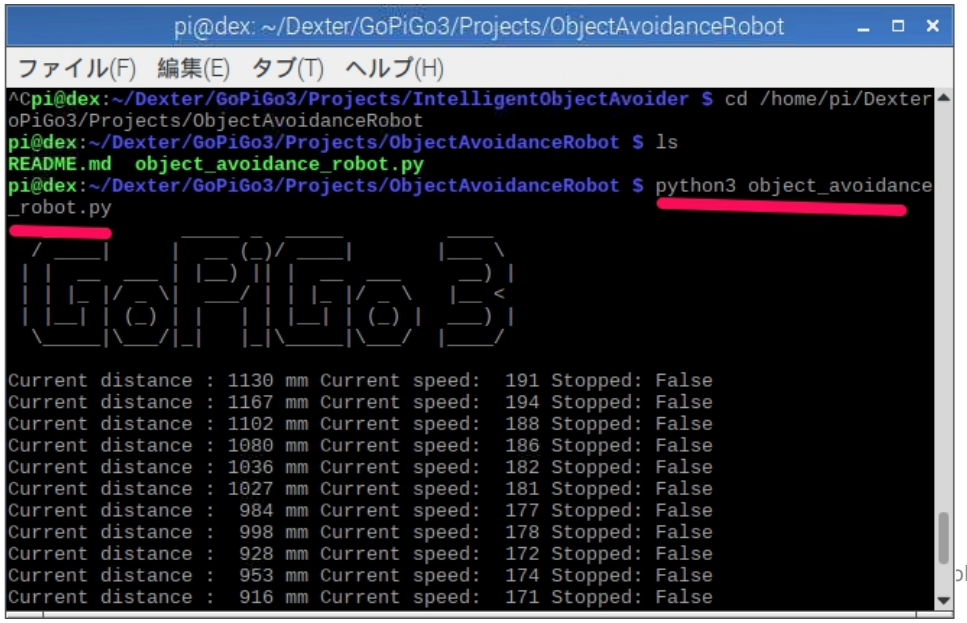
- https://edu.workbencheducation.com/cwists/preview/26765x
- https://youtu.be/aR7vXHT9pZk
- 概要：ワイヤレスマウスを使って、GoPiGoを動かす。
- 対象：高校、大学など
- 言語：Python
- cd /home/pi/Dexter/GoPiGo3/Projects/MouseControl
- python mouse_control_movement.py
- マウスのドングルをRaspberry　PiのUSBに差して、マウスを前後、左右に動かすと動作



応用例
③. 障害物検知自走

コマンド
$ cd
/home/pi/Dexter/GoPiGo3/Projects/ObjectAvoidanceRobot
$ python3 object_avoidance_robot.py

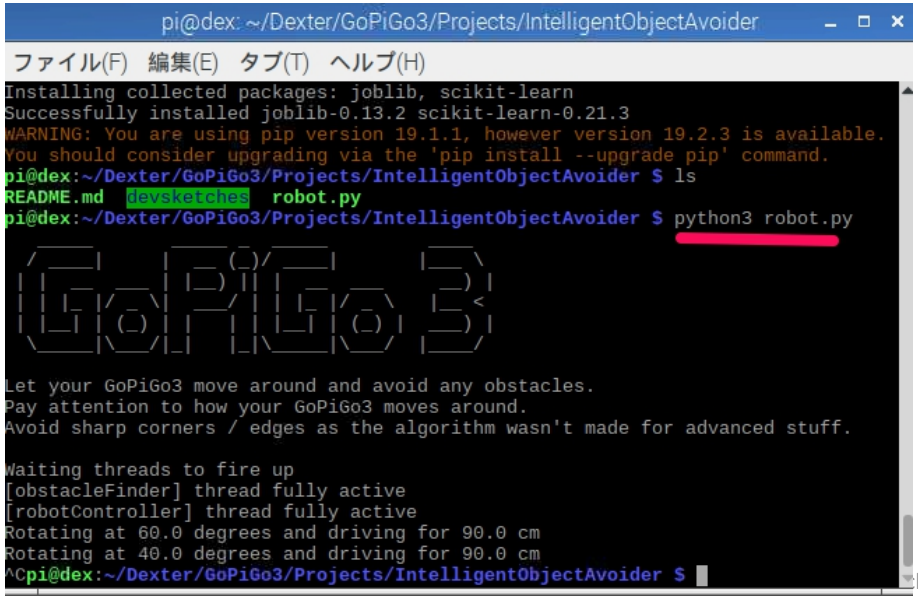
- 概要：DIセンサを使って、GoPiGoを動かす。
- 対象：高校、大学など
- 言語：Python
- cd /home/pi/Dexter/GoPiGo3/Projects/ObjectAvoidanceRobot
- python3 object_avoidance_robot.py
- DIセンサで、障害物を検知しながら自走。障害物があると自動停止



応用例
④. 障害物回避自走

コマンド
$ cd
/home/pi/Dexter/GoPiGo3/Projects/IntelligentObjectAvoider
$ python3　robot.py

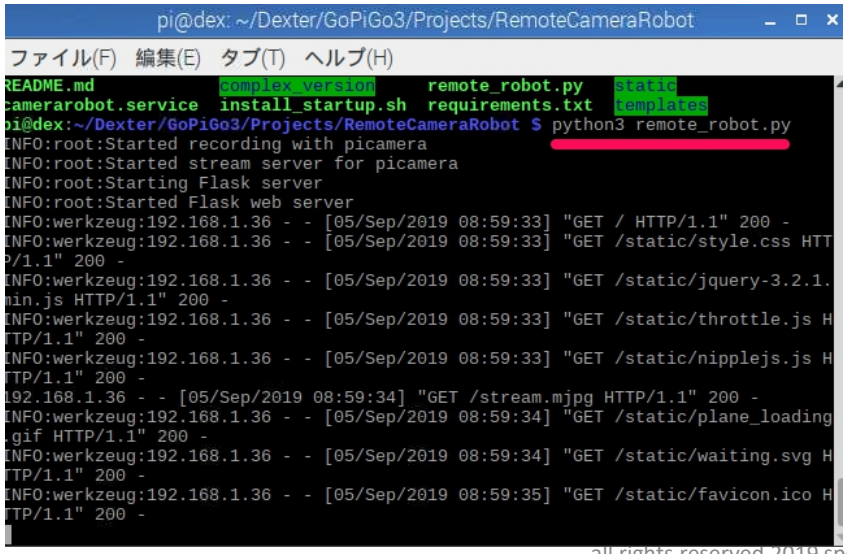
- 概要：DIセンサ、サーボを使って、GoPiGoを動かす。
- 対象：高校、大学など
- 言語：Python
- cd /home/pi/Dexter/GoPiGo3/Projects/IntelligentObjectAvoider
- python3　robot.py
- DIセンサ（サーボ付き）で、障害物を回避しながら自走

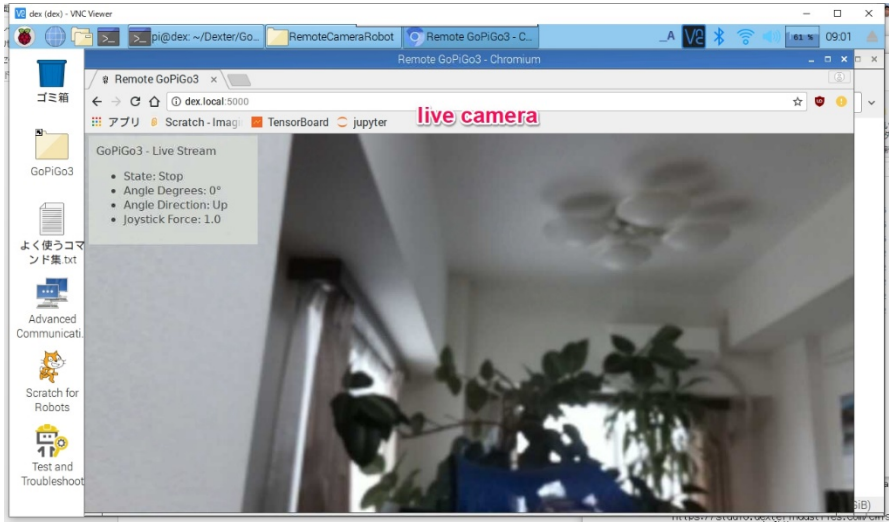


応用例
⑤. Webカメラ配信

コマンド
$ cd /home/pi/Dexter/GoPiGo3/Projects/RemoteCameraRobot
$ python3 remote_robot.py

- 概要：webカメラでライブ配信。
- 対象：高校、大学など
- 言語：Python
- cd /home/pi/Dexter/GoPiGo3/Projects/RemoteCameraRobot
- python3 remote_robot.py
- ブラウザを立ち上げて、http://dex.local:5000にアクセス





応用例
⑥. 自動走行（レーン追尾）

コマンド
$ cd
/home/pi/Dexter/GoPiGo3/Projects/Pixy2LaneTracker
$ python3 pixy2_lane_follower.py

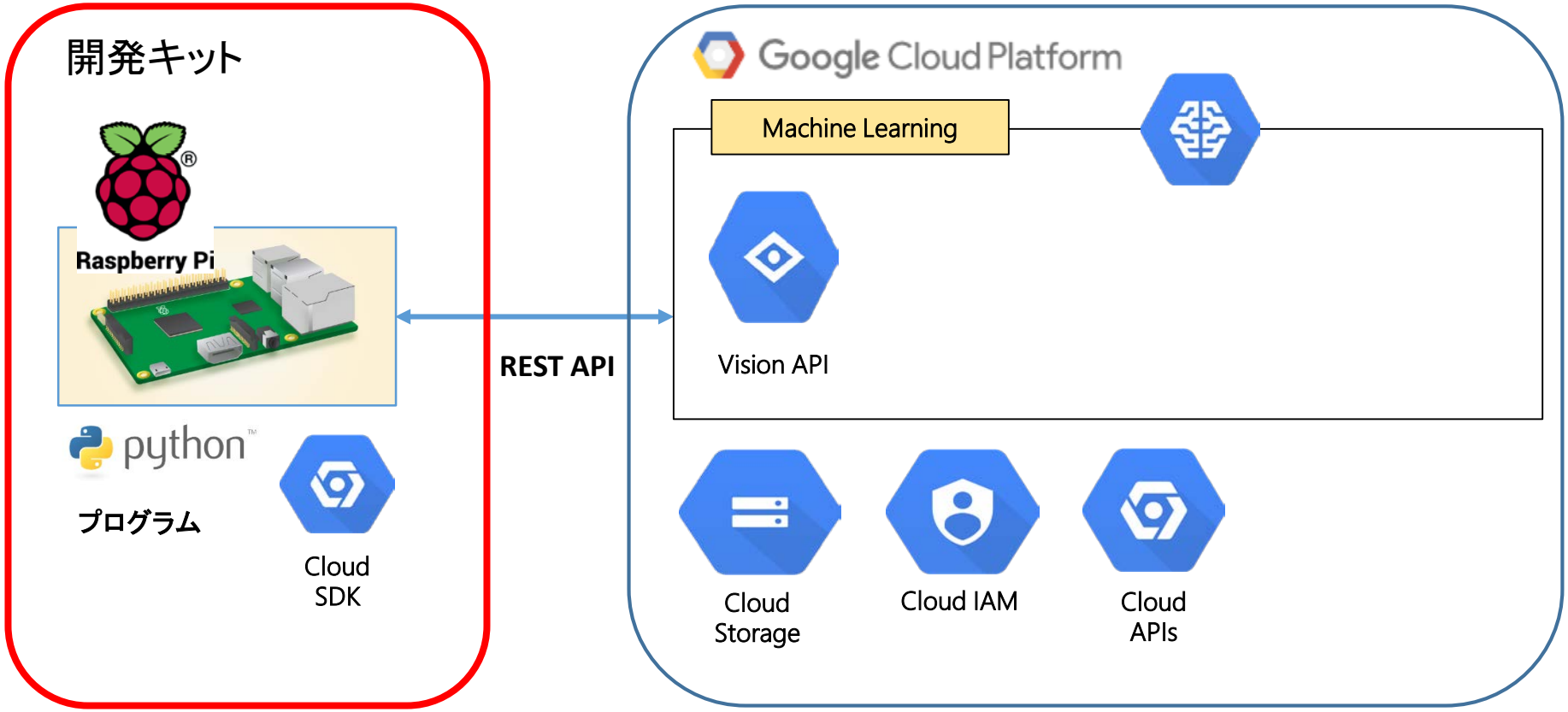
- https://github.com/DexterInd/GoPiGo3/tree/master/Projects/Pixy2LaneTracker
- 概要：Pixy2カメラでレーンを追尾しながら自動走行。
- 対象：高校、大学など
- 言語：Python
- cd /home/pi/Dexter/GoPiGo3/Projects/Pixy2LaneTracker
- python3 pixy2_lane_follower.py

Pixy2カメラが必要。
オプションで提供

GCP開発

③. グーグル・クラウド開発

1.開発キット全体像



Spectrum Technology

③. グーグル・クラウド開発

2. 開発キットプログラム一覧

A:実用可能
B:チャレンジ
C:試験段階

クラウドAI名	プログラム名	機能内容	実用度（当社評価）	信頼度（当社評価）	備考
Cloud vision api（画像認識）	OCR(文字認識）	画像内のテキストを検出、抽出できます。幅広い言語がサポートされており、言語の種類も自動で判別されます。	A	90%	手書き文字はC:10%位
	Face(顔検出）	画像に含まれる複数の人物の顔を検出できます。感情や帽子の着用といった主要な顔の属性についても識別されます。	A	90%	特定の人の顔認識はできません
	Label(ラベル検出）	乗り物や動物など、画像に写っているさまざまなカテゴリの物体を検出できます。	A	90%	
	Logo(ロゴ検出）	画像に含まれる一般的な商品ロゴを検出できます	C	10%	文字のないロゴは検出できない

③. グーグル・クラウド開発

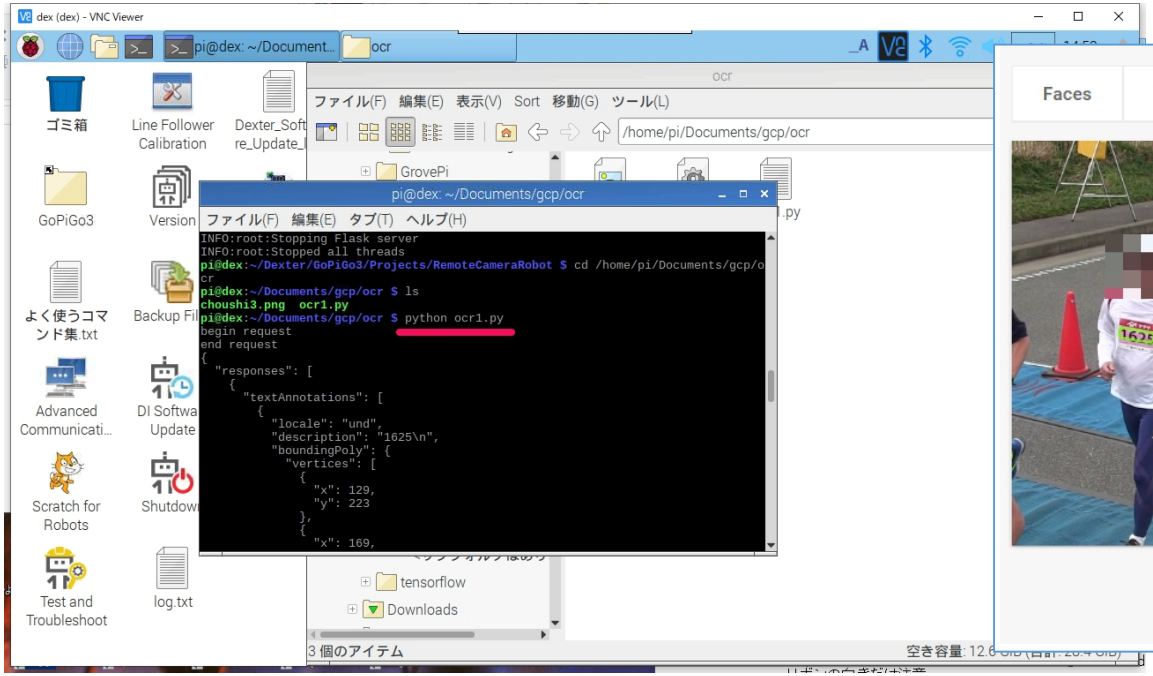
3.Cloud vision

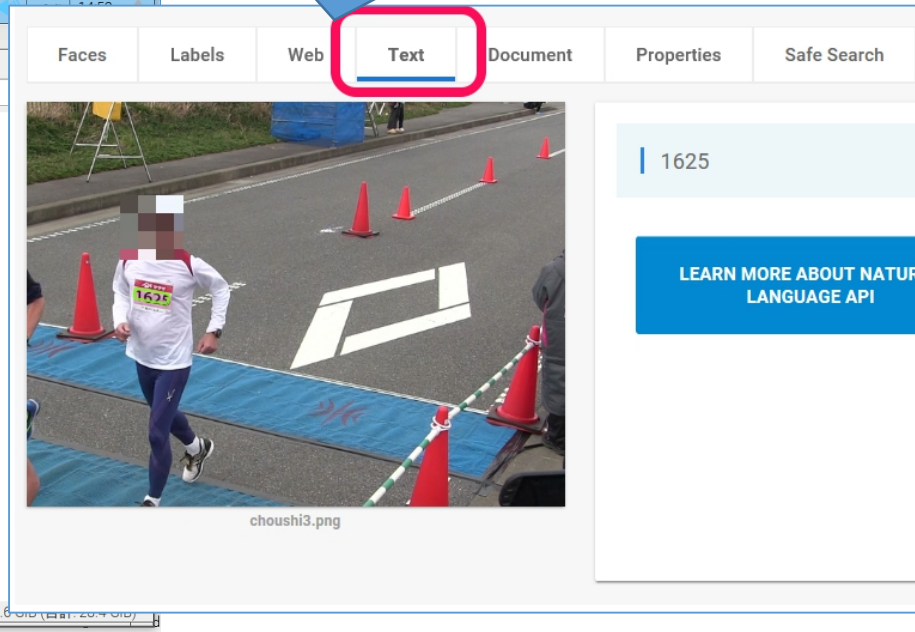
① 1_OCR(文字認識)

- テキスト認識：ocr1.py（画像からテキスト抽出）
  - Raspberry piからocr1.pyを実行します。
  - Api キーがあってると、結果が出力されます。
  - 出力は、piの画面上とファイルのdata1.jsonの双方にでます。

コマンド
$ cd /home/pi/Documents/gcp/1_ocr
$ python ocr1.py

Webでの確認
https://cloud.google.com/vision/?authuser=1&hl=ja





③. グーグル・クラウド開発

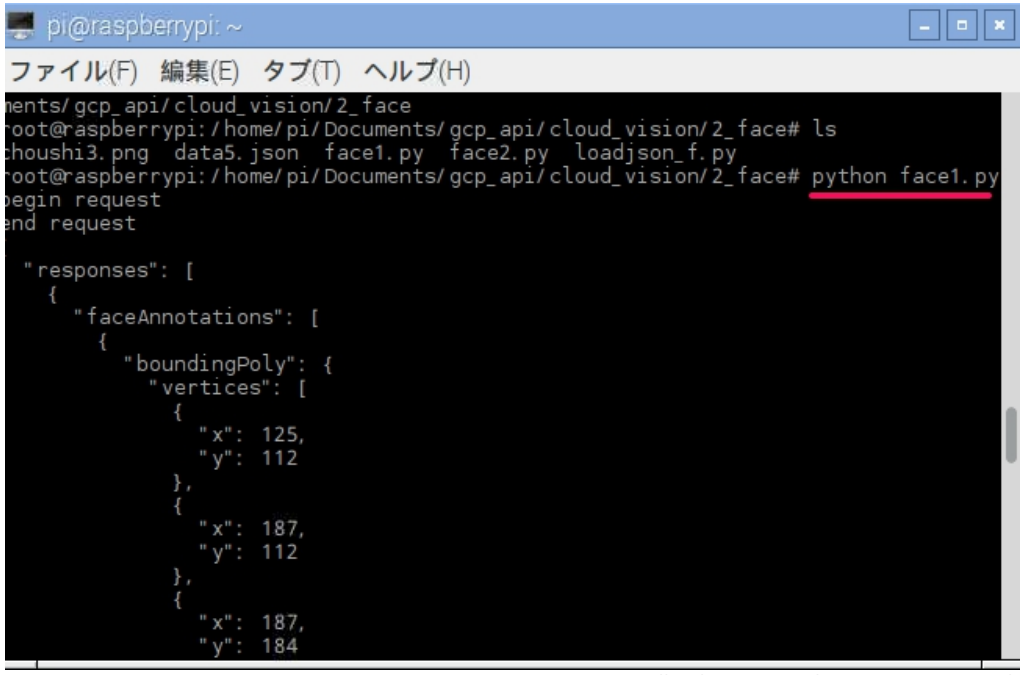
3.Cloud vision

② 2_face(顔検出）

- face1.py（http使用）
  - Raspberry piからface1.pyを実行します。
  - Api キーがあってると、結果が出力されます。
  - 出力は、piの画面上とファイルのdata5.jsonの双方にでます。

コマンド
cd /home/pi/Documents/gcp/2_face
python face1.py

Webでの確認
https://cloud.google.com/vision/?authuser=1&hl=ja



顔認識は、できません。プライバシー保護

③. グーグル・クラウド開発

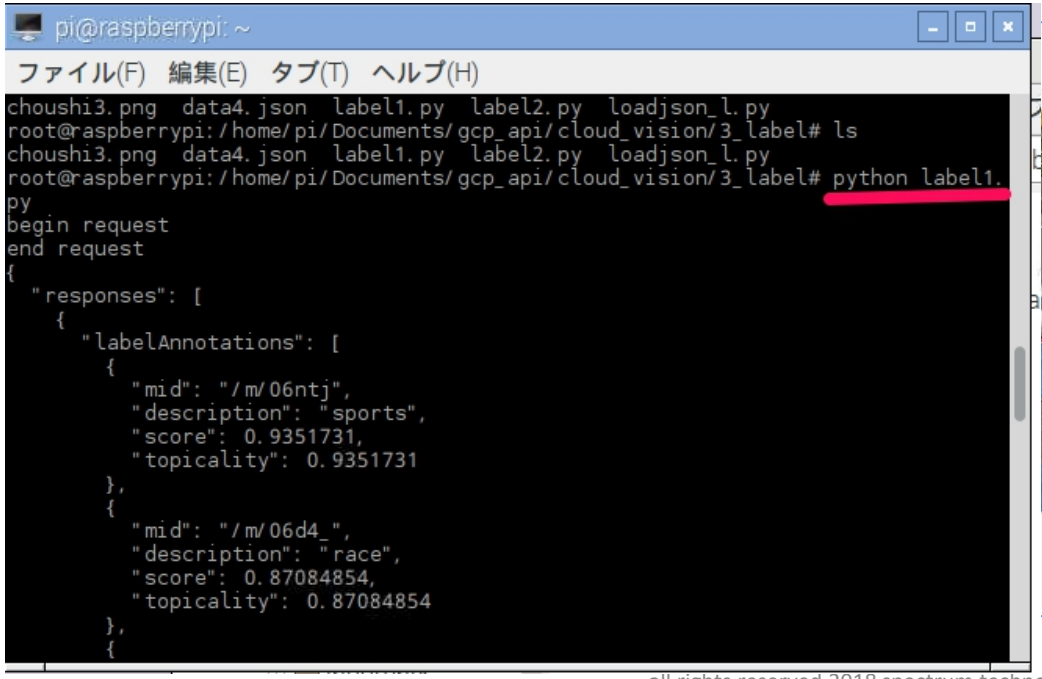
3.Cloud vision

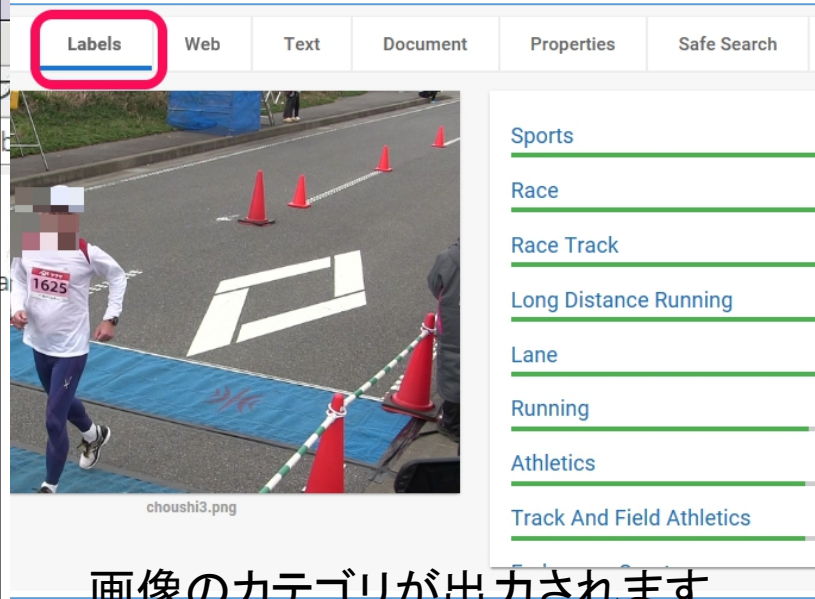
③ 3_label（ラベル検出）

- label1.py（http使用）
  - Raspberry piからlabel1.pyを実行します。
  - Api キーがあってると、結果が出力されます。
  - 出力は、piの画面上とファイルのdata4.jsonの双方にでます。

コマンド
cd /home/pi/Documents/gcp/3_label
python label1.py

Webでの確認
https://cloud.google.com/vision/?authuser=1&hl=ja





画像のカテゴリが出力されます

③. グーグル・クラウド開発

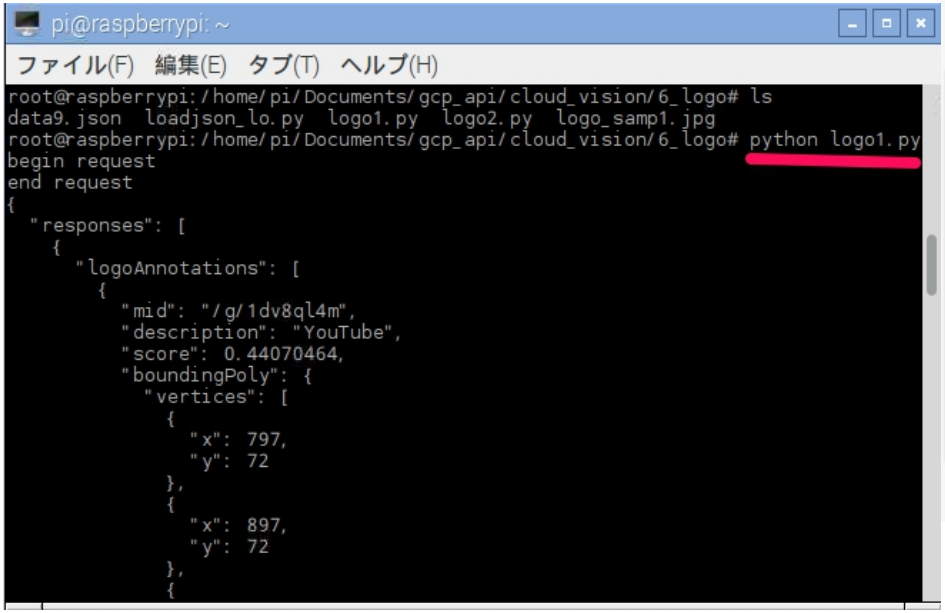
3.Cloud vision

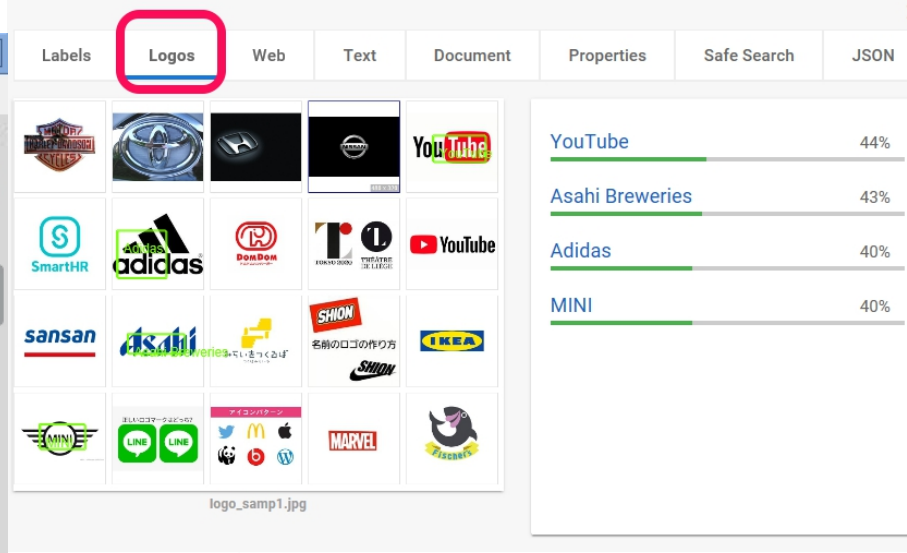
④ 6_logo（ロゴ検出）

- logo1.py（http使用）
  - Raspberry piからlogo1.pyを実行します。
  - Api キーがあってると、結果が出力されます。
  - 出力は、piの画面上とファイルのdata9.jsonの双方にでます。

コマンド
cd /home/pi/Documents/gcp/6_logo
python logo1.py

Webでの確認
https://cloud.google.com/vision/?authuser=1&hl=ja





ロゴ検出できます。テキストがない場合は検出できません

Spectrum Technology

③. グーグル・クラウド開発

3.Cloud vision

Google Cloud Platform

⑤ face(顔認識)

- camera-vision-face.py(ライブラリ使用)
  - Raspberry piからcamera-vision-face.pyを実行します。
  - Picameraで写真が撮られて、顔の感情分析をしてコマンド画面に出力されます。

コマンド
cd /home/pi/Documents/gcp/GoogleVisionTutorials
python camera-vision-face.py

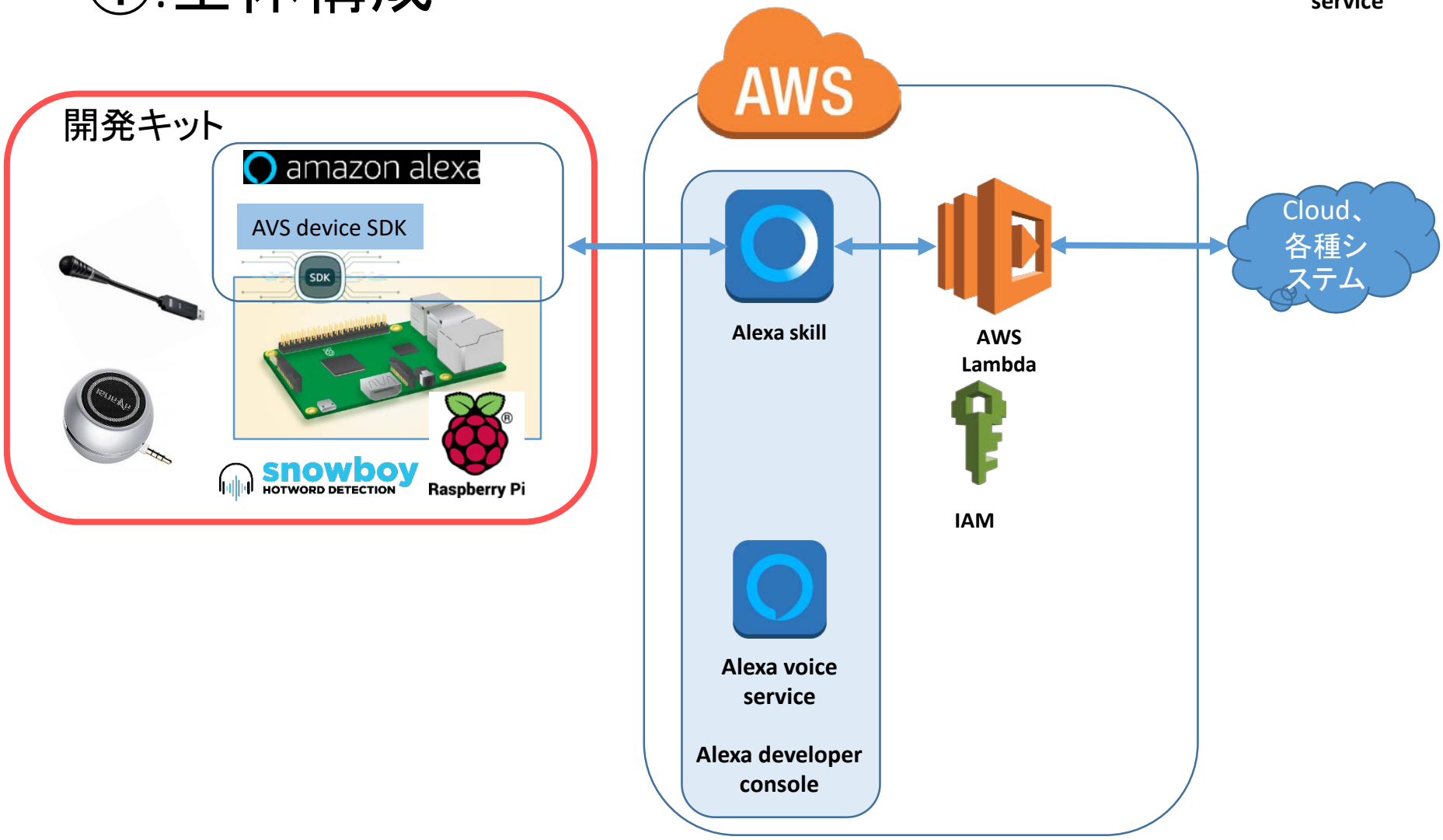
```
pi@dex: ~/Documents/gcp/GoogleVisionTutorials
ファイル(F)  編集(E)  タブ(T)  ヘルプ(H)
-rw-r--r-- 1 root root    911  4月   4 13:45 README.md
-rw-r--r-- 1 root root  27010  4月   4 13:45 camel.jpg
-rw-r--r-- 1 root root   1904  4月   4 13:45 camera-vision-face.py
-rw-r--r-- 1 root root   1905  4月   4 13:45 camera-vision-label.py
-rw-r--r-- 1 root root   1915  4月   4 13:45 camera-vision-logo.py
-rw-r--r-- 1 root root 180430  4月   4 13:45 face-google-vision.jpg
-rw-r--r-- 1 root root 599266  4月   4 13:58 image.jpg
-rw-r--r-- 1 root root 175829  4月   4 13:45 wheel.jpg
pi@dex:~/Documents/gcp/GoogleVisionTutorials $ python camera-vision-face.py
{
    "responses": [
        {
            "faceAnnotations": [
                {
                    "angerLikelihood": "VERY_UNLIKELY",
                    "blurredLikelihood": "VERY_UNLIKELY",
                    "boundingPoly": {
                        "vertices": [
                            {
                                "x": 408
                            },
                            {
                                "x": 926
                            },
```

Alexa開発

Alexa voice service

Spectrum Technology

Alexa開発
①.全体構成

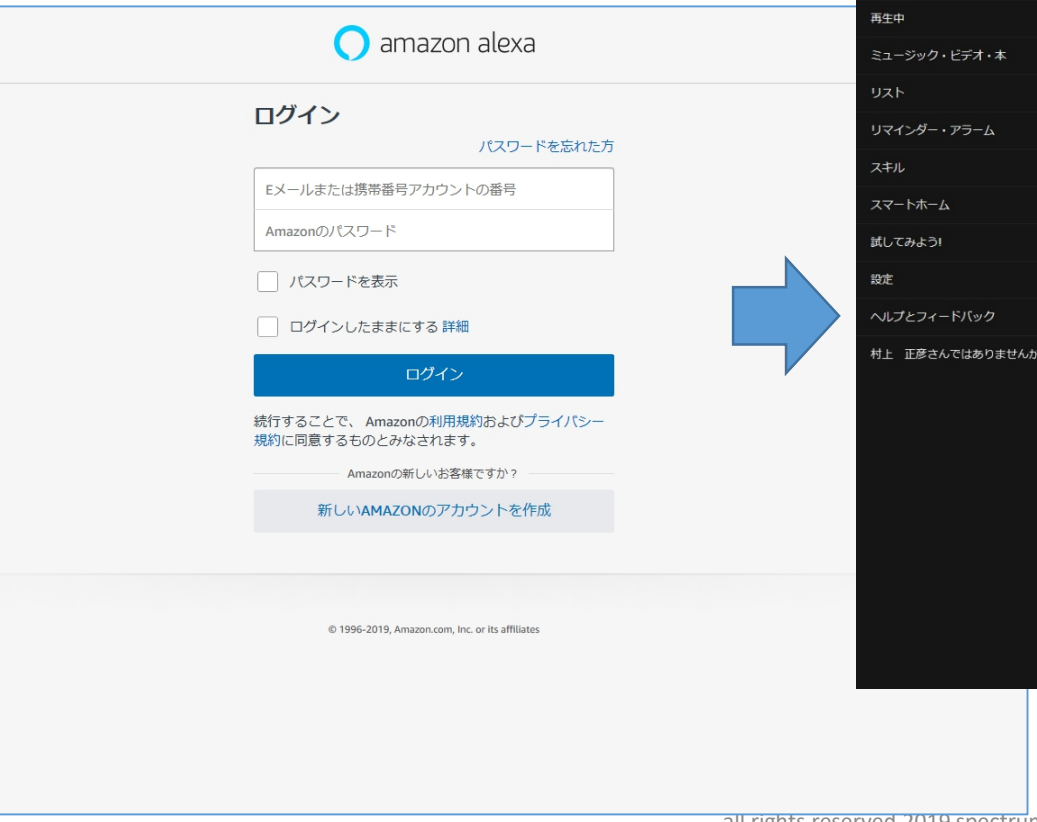


Spectrum Technology

Alexa開発
④. Amazon. co. jp設定

Echo

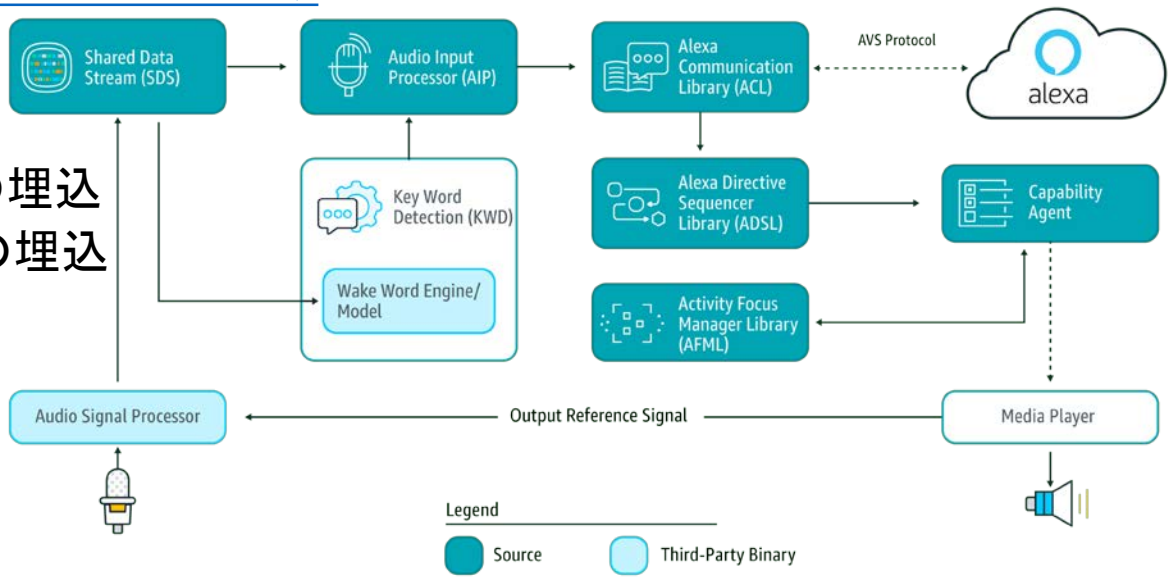
- Amazon　alexaログイン
  - Amazon. co. jpのアカウントでログインします。
    https://alexa.amazon.co.jp/



Alexa開発
⑥. Alexa Voice Service開発

Alexa voice service

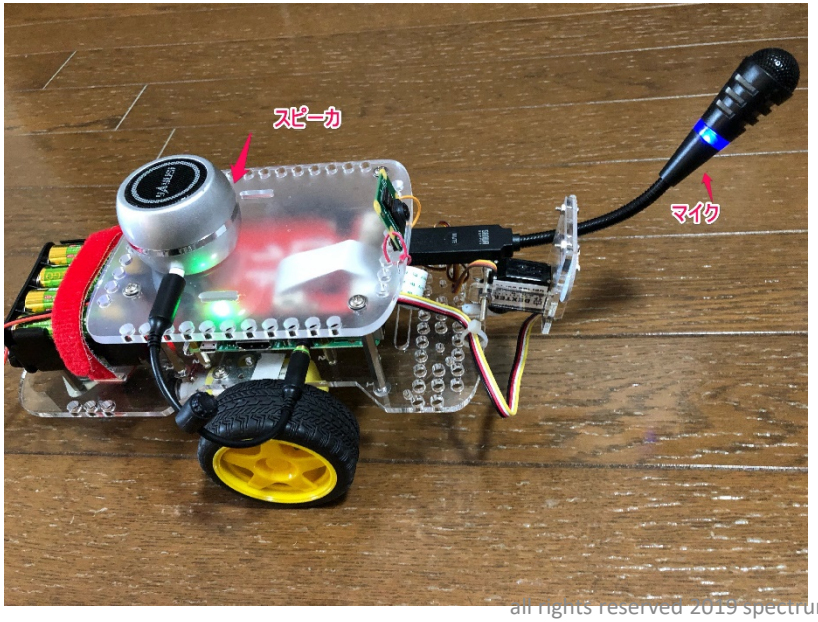
- 概要
  - Alexa Voice Service（以下AVSと略します）を使って、自社製品にAlexaの機能を搭載可能になります。Echoなどのデバイスを使わないで、スピーカ、マイクを付けてAlexaを実現します。AVS Device SDKをRaspberry Piに搭載し、開発環境を提供します。また、スピーカとマイクを付属しており、Alexaの端末として利用できます。
  - https://developer.amazon.com/ja/alexa-voice-service
  - https://alexa.github.io/avs-device-sdk/
- 用途
  - 高機能スマートスピーカ
  - TVなどの家電製品への埋込
  - スマートホーム製品への埋込



Alexa開発
⑥. Alexa Voice Service開発

Alexa voice service

- マイク、スピーカ設定
  - https://github.com/alexa/avs-device-sdk/wiki/Raspberry-Pi-Quick-Start-Guide-with-Script
  - マイク、スピーカ接続
    - マイク：USBにマイクを接続してください。青いLEDが点灯、赤が点灯はMuteなのでMuteを触って解除してください。
    - スピーカ：イヤホンジャックに接続して、裏面のボタンを押し、青いLEDが点灯するのを確認してください。最初は、内蔵の電池をUSBミニに接続して充電してから使用してください。

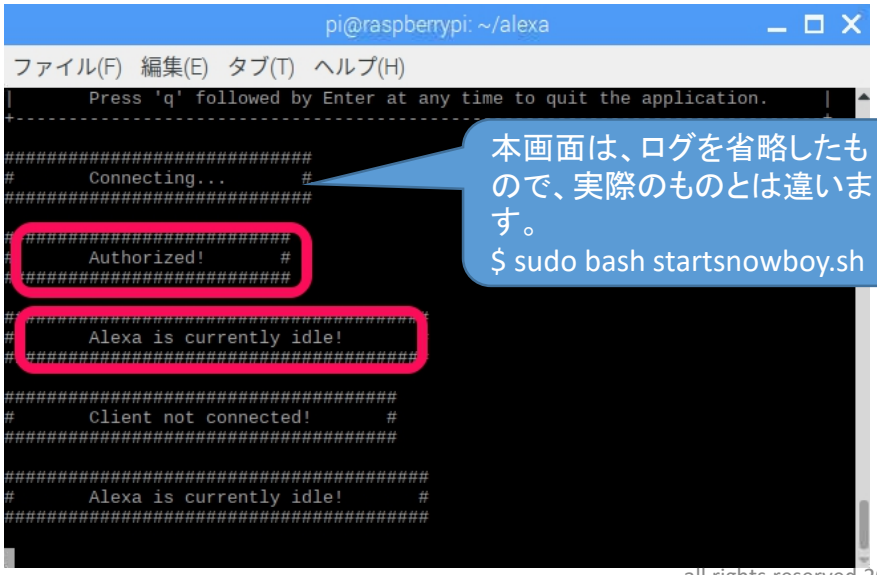


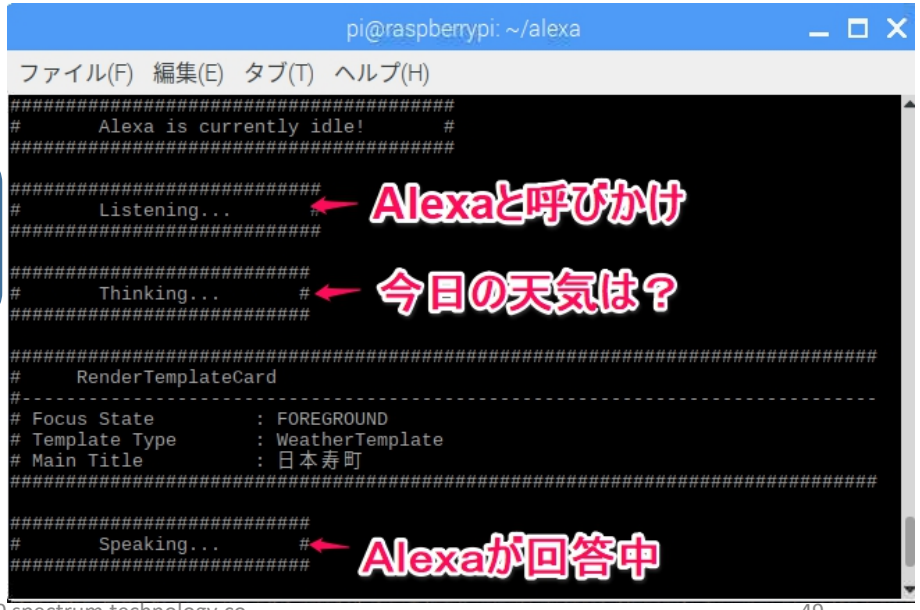
Alexa開発
⑥. Alexa Voice Service開発

Alexa voice service

Piのフォルダ位置
/home/pi/alexa/
$ sudo bash startsnowboy.sh

- AVS sample APP動作
  - コマンド画面でAuthorized, Alexa is currently idleが出れば、正常に動作中
  - マイクに向かって、Alexa、今日の天気は？と呼びかけて見ましょう。（Listeningになれば、AlexaのWakeupを認識した状態になります）
  - 日本語への切替、c, 1, 6と入力



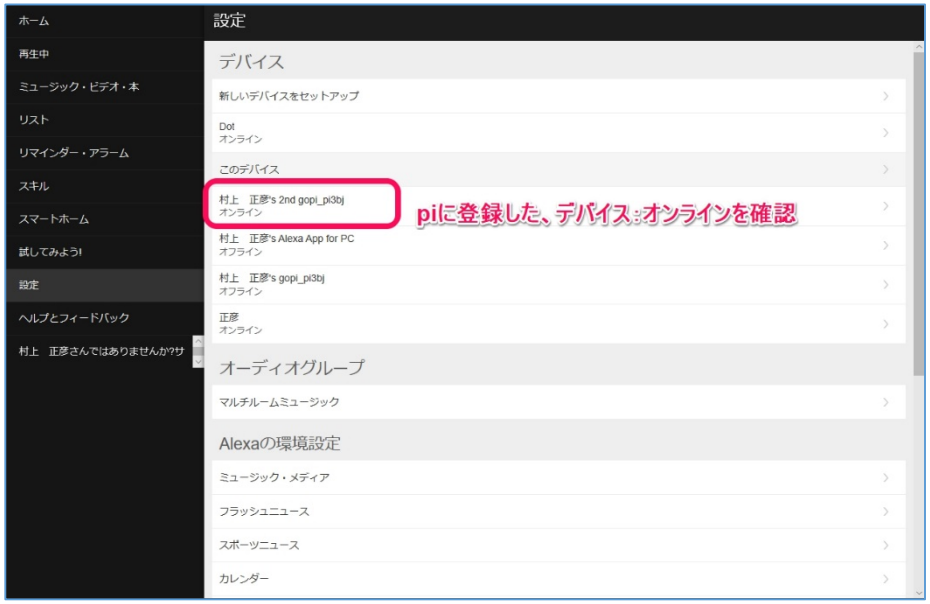


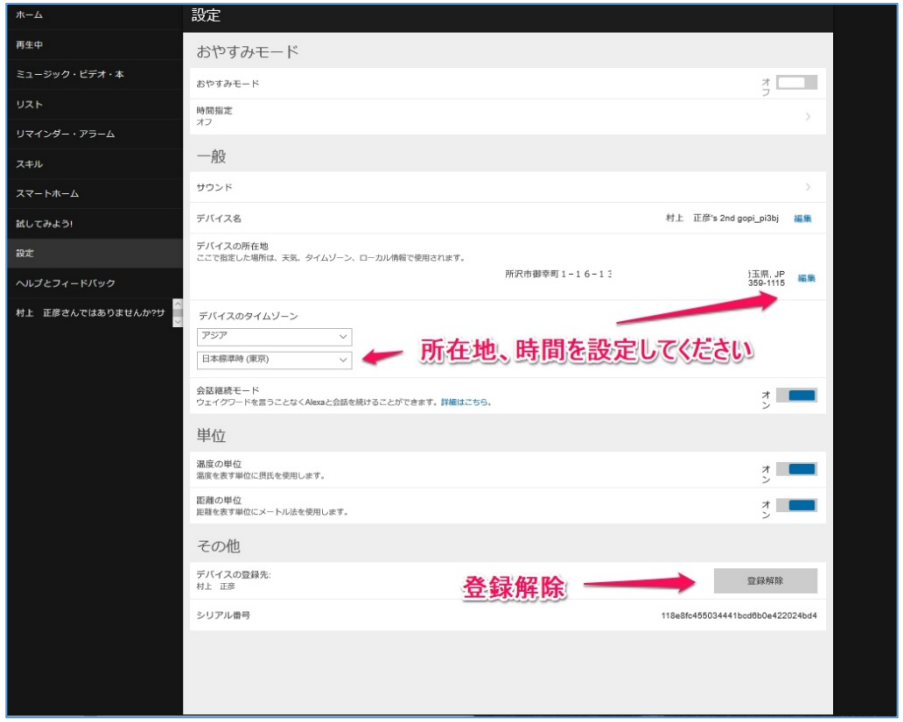
Spectrum Technology

Alexa開発
⑥. Alexa Voice Service開発

Alexa voice service

- デバイス確認
  - https://alexa.amazon.co.jp/spa/index.html#settings
  - 登録したPiのAlexa端末の確認を上記のURLから実施
  - デバイスがオンラインになっていることを確認
  - 端末を選択して、場所、時間の設定を行ってください。
  - 登録解除も可能です。





Alexa開発
⑥. Alexa Voice Service開発
・音声によるGoPiGo操作

音声によるGoPiGo操作

・GoPiGo動作

- 音声のコマンドをサーバで受信し、そ内容をPiのサーボに伝えます。

発話例

- アレクサ、前方向
- アレクサ、右方向

GoPiGo動作

- Go aheadの場合、1秒前進。
- プログラムで変更可