

マイコン実習④ Raspberry Pi Pico W × Python デバイス制御 I

1. はじめに

スマートフォンや自動車、家電製品に搭載されている組み込み技術は日本の基盤産業に欠かせない技術である。IoT はソフトウェア技術、ハードウェア技術、ネットワーク技術が三位一体で進化してきた分野であり、AI の導入により「知能をもったデバイス」が生み出されている。センサ、モータ、ディスプレイの制御を通して、機械の知性とは何かを考えてもらいたい。

2. 目的

Raspberry Pi Pico W と Circuit Python を利用して赤外線センサの反射を読み取り、アクチュエータや OLED を制御する。

3. 基礎知識

① Python

ガイド・ヴァンロッサム (蘭) により開発される。同氏はアムステルダム大学卒業後、技術研究機関や Google に在籍した。1980 年代後半、クリスマス休暇中にスクリプト言語の開発に着手し、1989 年前後に原型が完成。イギリスのコメディ番組『モンティ・パイソン』から名付け、1991 年に公開した。オープンソースという形で公開され、柔軟な設計と読みやすい文法によって徐々に注目を集める。インタプリタベースでありながら高い生産性と大規模開発に耐えうる拡張性を両立させたことが、多くのエンジニアに歓迎された。インデント (字下げ: Tab キー) によるオフサイドルールを特徴とする

図 1 Guido van Rossum(1956-)

② Circuit Python

MicroPython から派生した、教育および初心者向けのオープンソースのプログラミング言語である。

図 2 組み込み機器向けに軽量化された Python

かつての組み込み開発では、開発環境や開発言語処理系は、異なるベンダが提供するドキュメントを読み合わせ調整・工夫する必要があった。Arduino は AVR マイコンに Arduino Core ブートローダを書き込み、PC 側にシリアルポートドライバと Arduino IDE をインストールすれば開発環境が整う点が画期的である。Micro Python は Arduino の利点に加えて疑似ファイルシステムを持ち、Python 言語で記述可能となった。Circuit Python は USB ストレージの形で疑似ファイルシステムへのアクセス機能を提供するため、シリアルポートドライバが不要である。

③ SPI(Serial Peripheral Interface)

モトローラ社 (現 NXP 社) が開発した通信インタフェースで同期式・全二重通信のプロトコル。マスタと複数スレーブ間で高速なデータ転送を行うために設計されている。

・ SCLK (Serial Clock)

クロック信号 (マスタが生成)

・ MOSI (Master Out Slave In)

マスタ → スレーブへのデータ

・ MISO (Master In Slave Out)

スレーブ → マスタへのデータ

・ SS/CS (Slave Select / Chip Select)

スレーブ選択 (個別制御)

④ モータドライバ

一般的にモータなど動力部を有するデバイスには他のデバイスと比較して大きな電流が必要とされており、モータ起動時には瞬間的に1A以上流れる場合がある。TC78H653FTGは、出力ドライブトランジスタに低ON抵抗のDMOS素子を採用したDCブラシ付きモータ2個またはステッピングモータ1個駆動用のDual H-ブリッジドライバICである。正転/逆転/ストップ/ブレーキの4モードを選択でき、過電流検出、過熱検出、低電源電圧検出機能を内蔵している。

表1 入出力ファンクション

| IN 入力モード (MODE=L)、スモールモード (LARGE=L)時 | | | | | | | | モード | | | | |
|--------------------------------------|-------|------|-----------|-----------|-----------|-----------|-----|------|------|------|-------|----------|
| MODE | LARGE | STBY | IN1/ENB_A | IN2/PHA_A | IN3/ENB_B | IN4/PHA_B | | OUT1 | OUT2 | OUT3 | OUT4 | モード |
| L | L | H | H | L | X | X | | H | L | X | X | 正転 |
| | | H | L | H | X | X | | L | H | X | X | 逆転 |
| | | H | X | X | H | L | X | X | H | L | L | 正転 |
| | | H | X | X | L | H | X | X | L | H | L | 逆転 |
| | | H | H | H | H | H | L | L | L | L | L | ショートブレーキ |
| | | H | L | L | L | L | L | L | OFF | OFF | OFF | OFF |
| L | X | X | X | X | X | | OFF | OFF | OFF | OFF | スタンバイ | |

注: 2つのHブリッジを同時に設定し動作することが出来ます。
注: X: Don't Careになります。

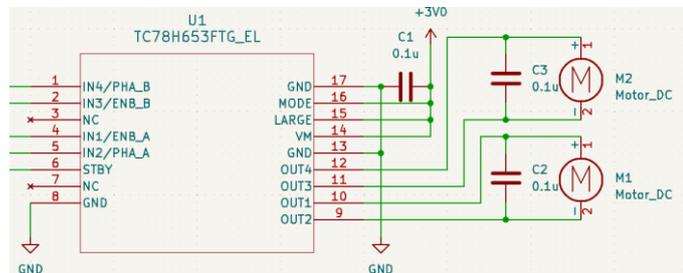


図3 モータドライバとモータ接続回路

⑤ フォトリフレクタ

光の反射を利用して物体の有無や位置を検出する。シンプルながら非常に実用的であり、自動ドアや変位センサなどに利用される。フォトインタラプタ(透過性)に比べ出力電流が小さく、外乱光がノイズとなるため、工夫が必要となる。

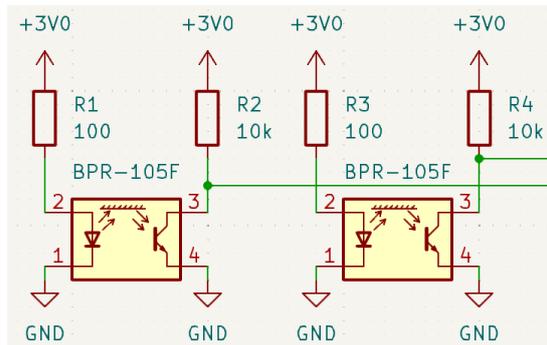


図5 フォトリフレクタ回路図

⑥ OLED

Organic Light Emitting Diode (有機発光ダイオード)は、有機材料に電流を通すことで発光し、薄くて柔軟な画面にテキスト、グラフィックス、画像を表示できるデバイスである。制御チップSSD1331はI²CとSPIの制御方法があり、ArduinoやRaspberryPi等で活躍するAdafruit製のライブラリを使用している。

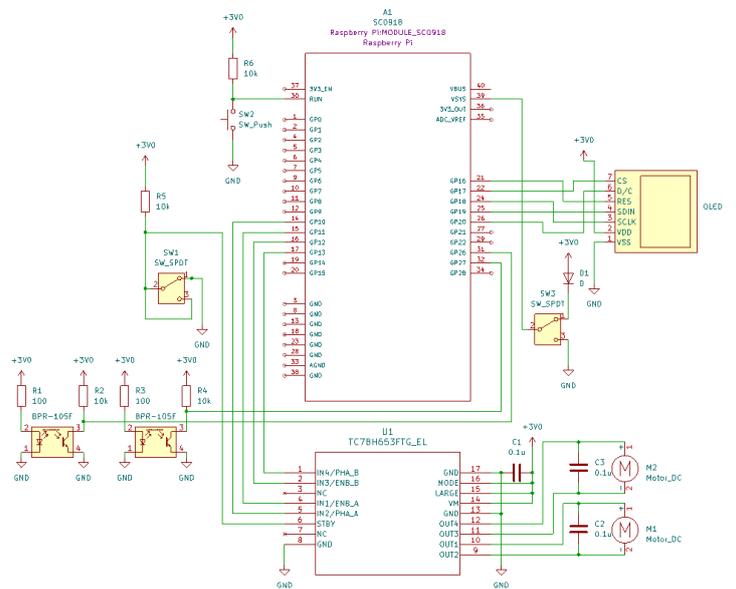
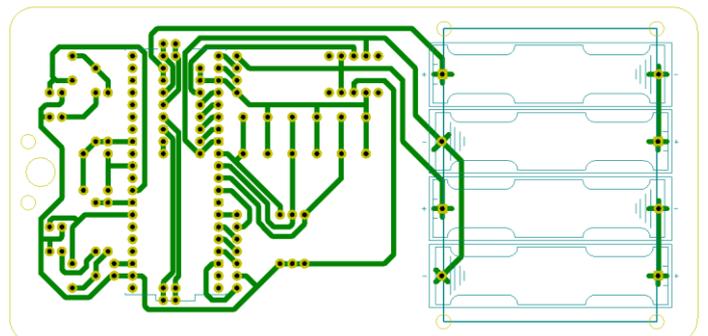


図4 フォトリフレクタのしくみ



4. 開発環境

Board

└ Raspberry Pi Pico W with rp2040

Firmware

└ adafruit-circuitpython-raspberry_pi_pico_w-en_US-8.2.10

Library

└ File

└ adafruit_ssd1331.mpy

└ dir

└ adafruit_display_shapes

└ adafruit_display_text

└ adafruit_imageload

└ adafruit_motor

5. 実習内容

□演習 1

教材配布 (S:ドライブ) から

「thonny-4.1.6-windows-portable」と

「code.py」をダウンロードしよう。

□演習 2

Raspberry Pi Pico W と PC を接続し

CIRCUITPY (E:ドライブ) に「code.py」

を格納しよう。

□演習 3

Portable Thonny を起動し、code.py の

Author (氏名) と Date (yyyy/mm/dd)

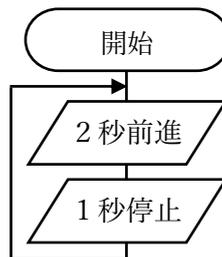
を記述しよう。

□演習 4

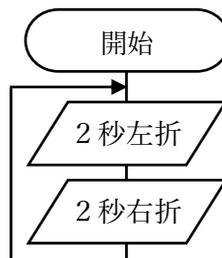
センサ感度を確認しよう。

| L | R | sensorL | | sensorR | |
|---|---|---------|---|---------|---|
| | | 16bit | V | 16bit | V |
| 白 | 白 | | v | | v |
| 黒 | 黒 | | v | | v |

□課題 1 流れ図に示す通り制御しよう。

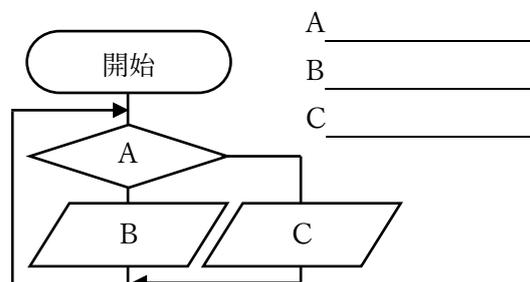


□課題 2 流れ図に示す通り制御しよう。



□課題 3 閾値を () V とする。

センサ 1 個によるライントレーシングプログラムの流れ図を設計し、制御しよう。



A _____
B _____
C _____

□課題 4 下表空欄に 0 か 1 を書き込み、

センサ 2 個によるライントレーシングプログラムを作成しよう。

| センサ | 白白 | 白黒 | 黒白 | 黒黒 |
|---------|--------|--------|--------|--------|
| 動作 | 前進 | 右折 | 左折 | 停止 |
| OLED | [W][W] | [W][B] | [B][W] | [B][B] |
| motorR1 | | | | |
| motorR2 | | | | |
| motorL1 | | | | |
| motorL2 | | | | |

