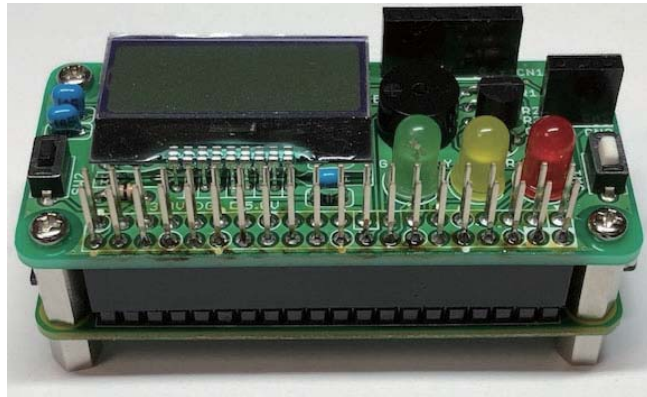


Raspberry Pi Zero WH 用 IoT 学習 HAT キット



【キットには Raspberry Pi Zero WH は含まれていません】

Raspberry Pi Zero WH を使って、手軽に IoT を学ぶことのできる拡張ボード（HAT : Hardware Attached on Top）のキットです。

★特徴

- ・ Raspberry Pi Zero WH と同一外形で、ピンソケットで重ねて接続
- ・ 表面実装部品を使用していないので、自分で組み立てが可能
- ・ 連結ピンソケットで Raspberry Pi Zero WH と接続するので HAT に接続していても、連結ピン経由の配線で I2C、SPI、UART の機能を使用可能
- ・ 8 文字×2 行の LCD、LED（赤・黄・緑）、φ 9 mm のブザーを実装
- ・ 焦電型赤外線（人感）センサー 搭載用 3P ソケット実装
- ・ BME280（温度・湿度・気圧センサー） 搭載用 6P ソケット実装
- ・ シャットダウン用スイッチ実装（黒色：別途スクリプト使用）
- ・ 自由に使えるトリガ用スイッチ実装（白色）
- ・ 多くの Python サンプルプログラムによる IoT 学習が可能

★別売

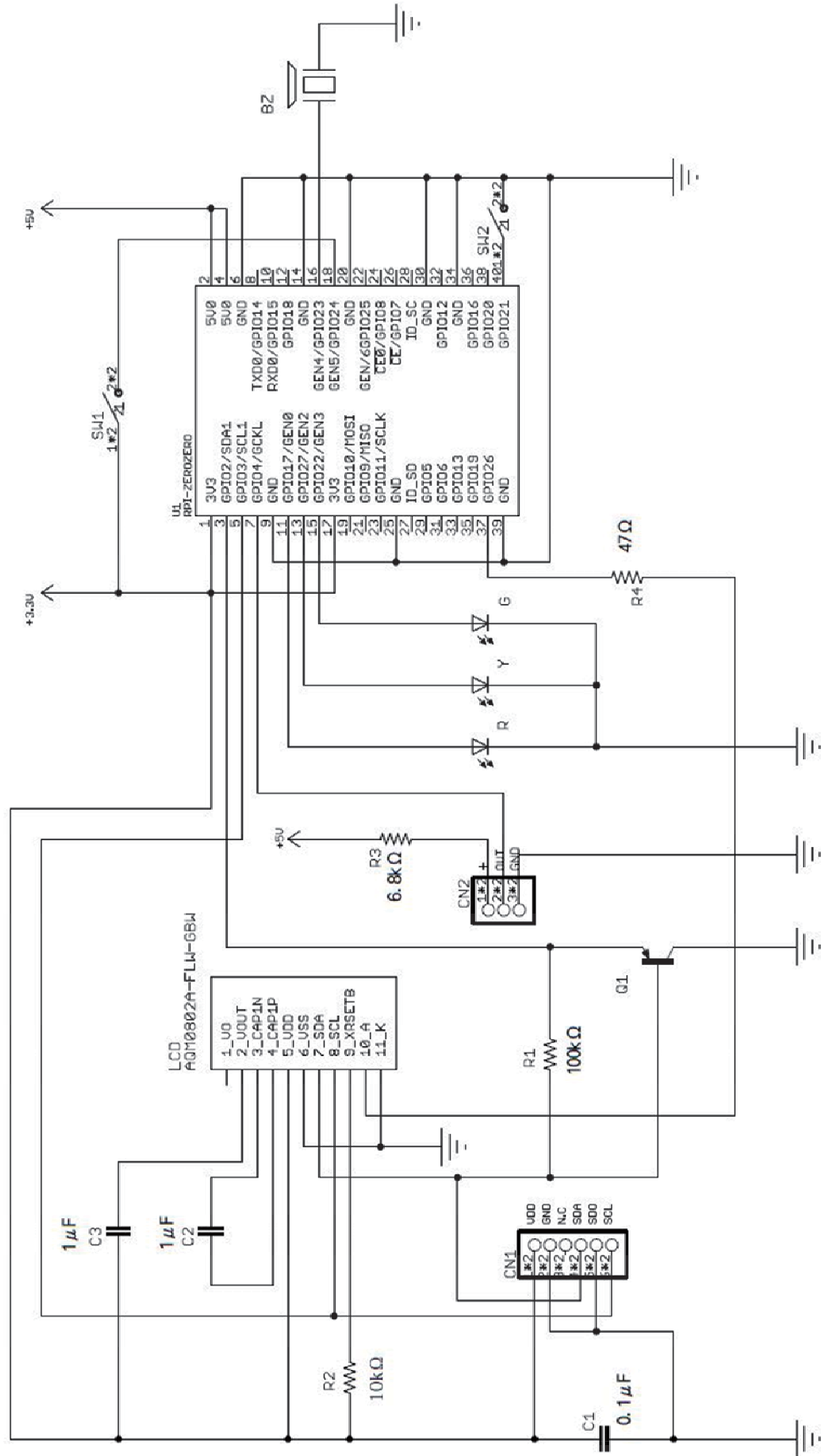
- ・ Raspberry Pi Zero WH（通販コード M-12961）
- ・ 焦電型赤外線（人感）センサー（通販コード M-09002）
- ・ BME280（温度・湿度・気圧センサー）（通販コード K-09421）
- ・ 別途 マイクロ SD カードや USB ケーブルが必要

★参考図書

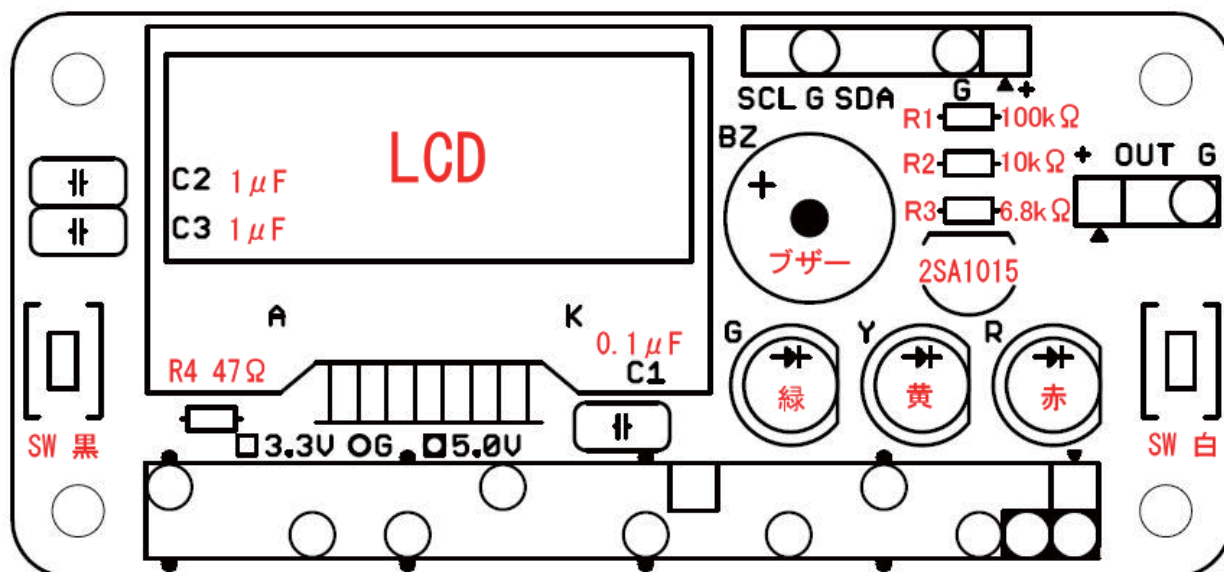
Raspberry Pi Zero による IoT 入門 - Zero W 対応 -

著者：今井一雅 出版社：コロナ社 ISBN978-4-339-00901-9

回路图



部品配置図（実体図）



部品表

IoT 学習 HAT 部品表（本キットに含まれているもの）

部品名	部品の内容	仕様	個数
R1	抵抗 1/6W (カラーコード：茶黒黄金)	100kΩ	1
R2	抵抗 1/6W (カラーコード：茶黒橙金)	10kΩ	1
R3	抵抗 1/6W (カラーコード：青灰赤金)	6.8kΩ	1
R4	抵抗 1/6W (カラーコード：黄紫黒金)	47Ω	1
C1	コンデンサ (本体表記：104)	0.1μF 50V	1
C2.C3	コンデンサ (本体表記：105)	1μF 50V	2
Q1	PNP トランジスタ	2SA1015 (ランク：Y、GR 等)	1
SW1	スイッチ	白	1
SW2	スイッチ	黒	1
LCD	I2C 接続小型キャラクタ LCD モジュール	8×2 行 (バックライト付き)	1
CN1	ピンソケット 6P	メス 1×6	1
CN2	ピンソケット 3P	メス 1×3	1
BZ	圧電ブザー	直径 9mm 高さ 5.5mm	1
G	抵抗内蔵 LED 緑	5mm 5V 用	1
Y	抵抗内蔵 LED 黄	5mm 5V 用	1
R	抵抗内蔵 LED 赤	5mm 5V 用	1
	連結ピンソケット	2×20 (40P)	1
	なべ小ねじ (+)	M2.6×5	4
	六角両メネジ	FB26-7	4
	六角オネジ・メネジ	MB26-11	4

※本キットに含まれていないもの

マイコンボード（ラズベリーパイゼロ ダブルエイチ）

	Raspberry Pi Zero WH	通販コード M-12961	1
--	----------------------	---------------	---

センサー関係

	BME280 仕様 温湿度・気圧センサ	通販コード K-09421	1
--	---------------------	---------------	---

	焦電型赤外線（人感）センサ	通販コード M-09002	1
--	---------------	---------------	---

その他 各自で準備していただくもの

	MicroSD カード Class10 8GB 以上	Raspbian イメージを書込済	1
--	----------------------------	-------------------	---

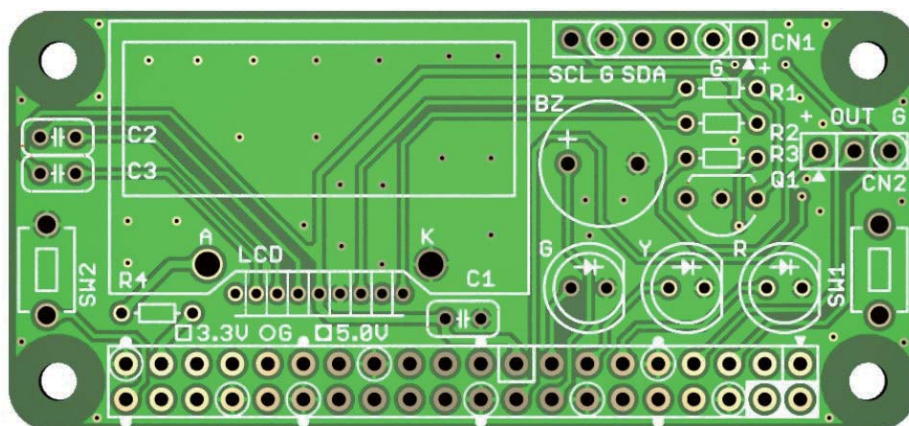
	USB micro B ケーブル	通販コード C-07607 など	1
--	------------------	------------------	---

	両面テープ	5~10mm 幅（組立時使用）	適量
--	-------	-----------------	----

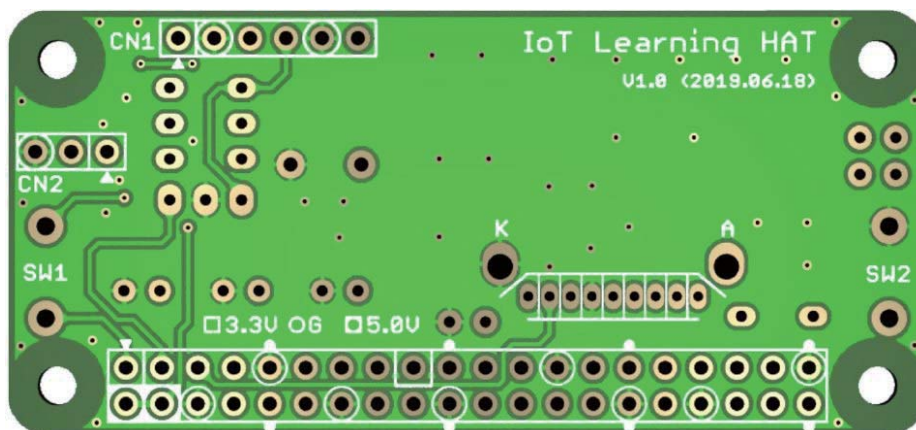
組立手順

(1) プリント基板の見方

プリント基板は主に部品を配置する部品面（表面）と、はんだ付けするはんだ面（裏面）があります。



部品面（表面）

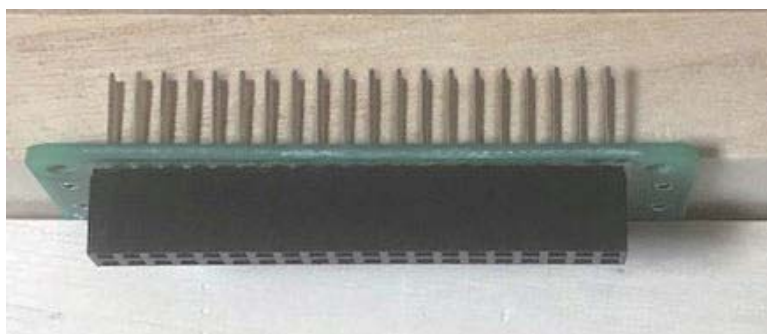


はんだ面（裏面）

(2) 連結コネクタの挿入およびはんだ付け

連結コネクタははんだ面（裏面）より挿入し、部品面（表面）側をはんだ付けする。

「他の部品と逆側から挿入・はんだ付けするので注意」



最初に端の1ヶ所はんだ付けとし、浮きがないように対角線上のランドのはんだ付けを行い連結コネクタがプリント基板に密着していることを確認して他のピンをはんだ付けする。はんだ付けするときはあまり長時間はんだゴテでピンを熱すると連結コネクタの樹脂部分を

溶かしてしまうことがあるので注意すること。

はんだ付けした後ははんだが富士山形になっていることがよい。



(3) 部品の挿入およびはんだ付け (LCD 以外)

部品配置図 (実体図) を見ながら各部品を挿入しはんだ付けしていく。

ブザー・トランジスタ・LED は挿入方向があるので部品配置図 (実体図) や部品面 (表面) に印刷されているシルク印刷 (白色の印刷) に合わせて挿入していく。

抵抗は 5.08mm ピッチになっているので 1/6W を使用し、
右の

写真のように抵抗本体のすぐ横で足を成形すると、ほぼ
5.08mm ピッチになり挿入しやすい。



はんだ付けの時どうしても逆さ向けにしなければならないので
部品の浮きが発生しやすい。部品の足をハの字に成形して

浮きが発生しにくくしても浮くことがあるので、部品のランド 1 ヶ所をはんだ付けし、浮き
補正を行ってから両側をはんだ付けするようにして浮きがでないように注意する。

(抵抗値やコンデンサ値の読み方)



R1: 100 k Ω (カラーコード 茶黒黄金)



R2 : 10 k Ω (カラーコード 茶黒橙金)



R3 : 6.8 k Ω (カラーコード 青灰赤金)



R4 : 47 Ω (カラーコード 黄紫黒金)



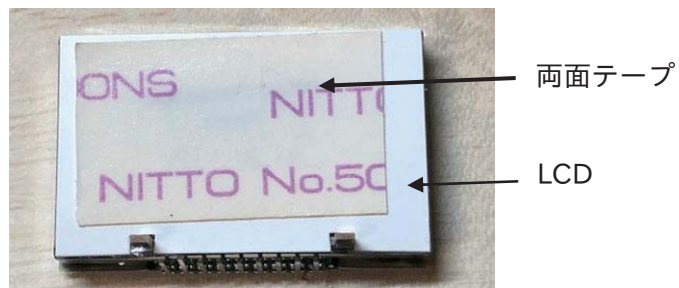
C1 : コンデンサ 0.1 μ F (表記 104)



C2・C3 : コンデンサ 1 μ F (表記 105)

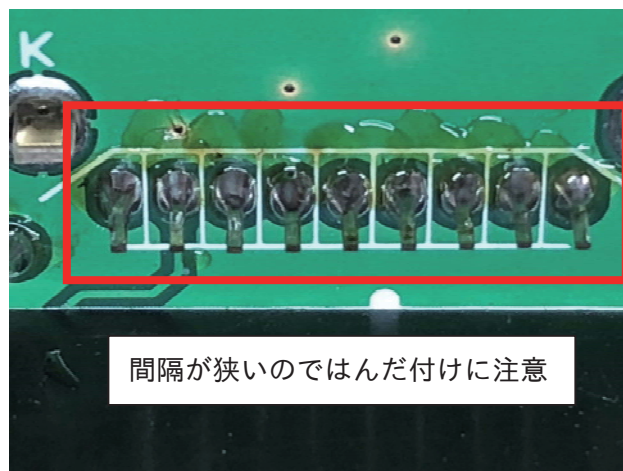
(4) LCD の挿入およびはんだ付け

LCD は浮きがあると LCD が割れたりすることがあるので、裏に両面テープを貼り、プリント基板に貼り付けて軽く押してプリント基板に固定する。



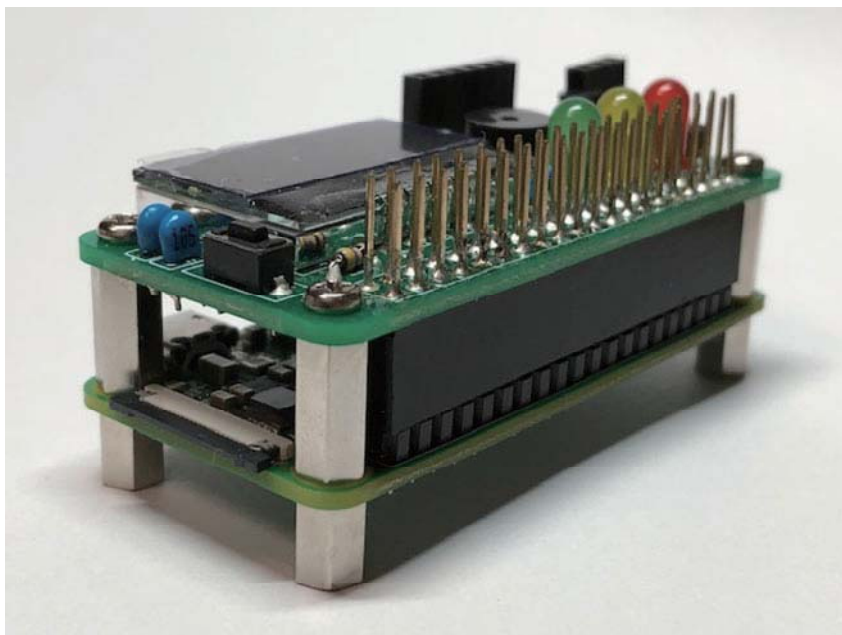
両面テープで貼り付けることにより、 はんだ付けの為逆さ向けても浮きが発生しにくい。

LCD のピンピッチが他の部品より狭いので はんだブリッジしないように注意 が必要。

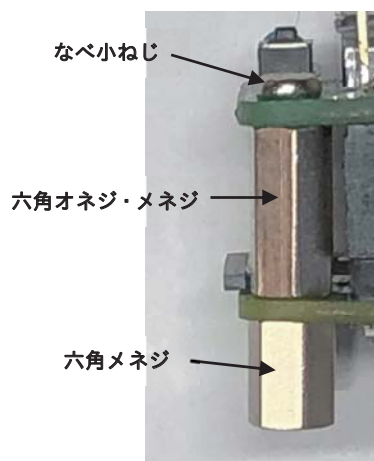


Raspberry Pi Zero WH との接続

まず Raspberry Pi Zero WH に下記の様にスペーサ（なべ小ねじ以外）を取り付けて、下の写真のように Raspberry Pi Zero WH のピンに IoT 学習 HAT のソケットを差し込みねじ止めする。

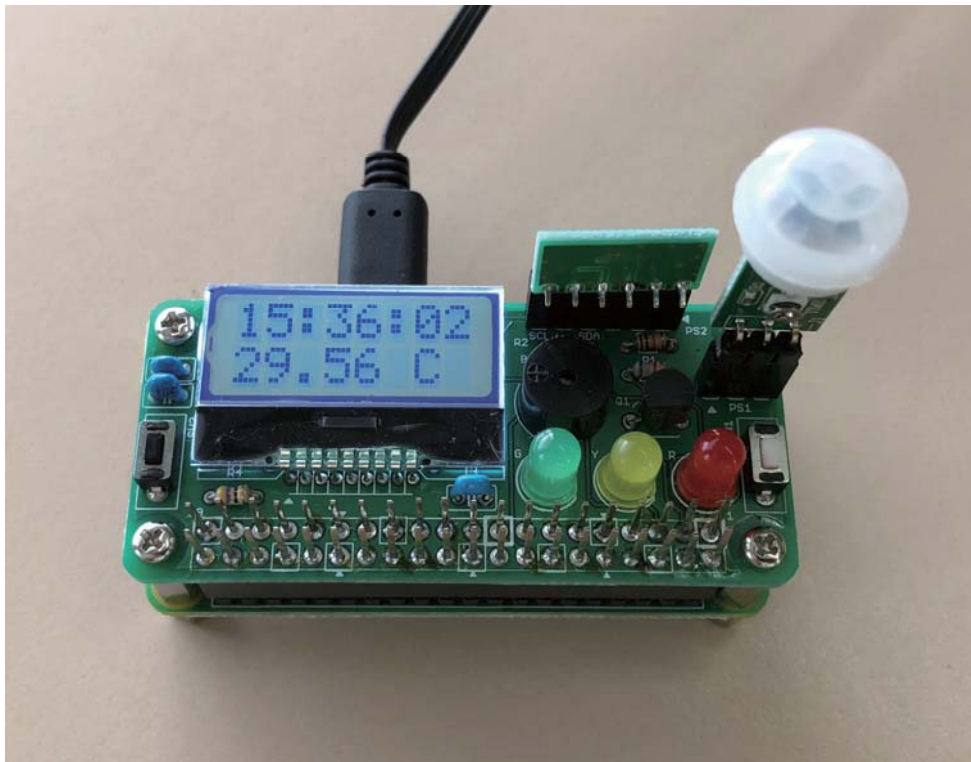


スペーサーの取り付け方

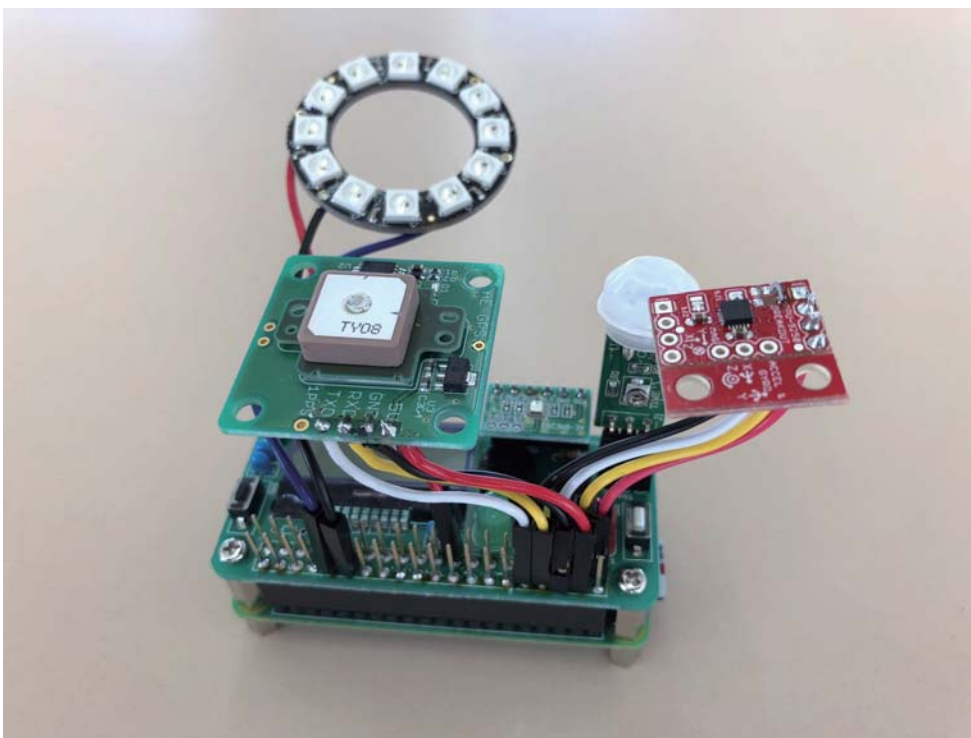


センサー等の接続例

- (1) 別売りの焦電型赤外線（人感）センサーと BME280（温度・湿度・気圧センサー）を接続した例



- (2) 連結ピンソケットに GPS モジュールや 9 軸センサーなどを接続した例

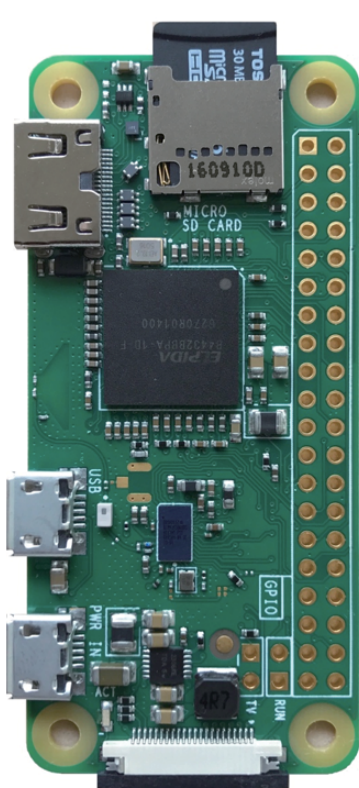


ピン配置

ピン配列

連結ピンソケットを使用しているため IoT 学習 HAT のピン配列も Raspberry Pi と同じ

Raspberry Pi Zero



Raspberry Pi Zero 40Pin GPIO ピンレイアウト

		ピン番号			
+3.3V	+3.3V	①	②	+5V	
I2C_SDA	GPIO 2	3	④	+5V	+5V
I2C_SCL	GPIO 3	5	6	GND	GND
	GPIO 4	7	8	GPIO 14	UART0 TX
	GND	9	10	GPIO 15	UART0 RX
	GPIO 17	11	12	GPIO 18	PWM
	GPIO27	13	14	GND	
	GPIO22	15	16	GPIO23	
+3.3V	+3.3V	⑮	18	GPIO24	
SPI_MOSI	GPIO 10	19	20	GND	
SPI_MISO	GPIO 9	21	22	GPIO25	
SPI_SCLK	GPIO 11	23	24	GPIO 8	SPI_CEO
GND	GND	25	26	GPIO 7	SPI_CE 1
ID_SD	GPIO 0	27	28	GPIO 1	ID_SC
	GPIO 5	29	30	GND	
	GPIO 6	31	32	GPIO12	
	GPIO 13	33	34	GND	
	GPIO 19	35	36	GPIO 16	
	GPIO26	37	38	GPIO20	
	GND	39	40	GPIO21	

IoT 学習 HAT で使用しているピンの一覧と PWM の対応

I/O 部品名	ピンの名前	PIN 番号
LED1(R:赤)	GPIO17	11
LED2(Y:黄)	GPIO27	13
LED3(G:緑)	GPIO22	15
ブザー(BZ)	GPIO23	16
スイッチ 1(SW1:白)	GPIO24	18
スイッチ 2(SW2:黒)	GPIO21	40
LCD バックライト	GPIO26	37
人感センサー(CN2)	GPIO4	7
温湿度・気圧センサ(CN1)	(I2C_1)	3,5
PWM0 #1	GPIO18	12
PWM0 #2	GPIO12	32

動作確認

(1) Wiringpi のインストール

秋月電子通商の HP より学習 HAT のテストプログラムをダウンロードする。

IoT 学習 HAT を動作させるには Wiringpi のインストールが必要になるので下記のようにインストールする。

(host 名が raspberrypi の場合)

```
pi@raspberrypi:~ $ sudo apt-get install python-dev
pi@raspberrypi:~ $ sudo pip install wiringpi2

. . . . (警告が沢山出ます) . . . .

Successfully installed wiringpi2 wiringpi
Cleaning up...
```

(2) /boot/config.txt の編集

LCD を使用する場合、標準の I2C のボーレートでは動かない場合があるので赤字の 1 行を追加しボーレートを初期設定の 100kHz から 50kHz に変更する。

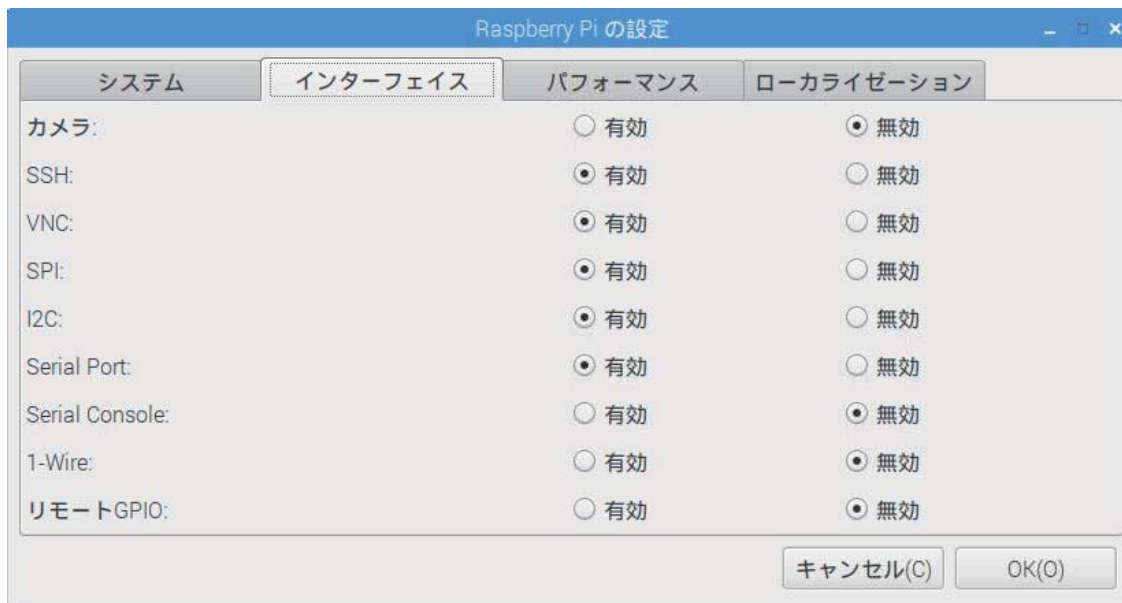
```
pi@raspberrypi:~ $ sudo nano /boot/config.txt

# Uncomment some or all of these to enable the optional hardware
interfaces
dtparam=i2c_arm=on
#dtparam=i2s=on
#dtparam=spi=on
の部分に下記の 1 行をくわえてください

# Uncomment some or all of these to enable the optional hardware interfaces
dtparam=i2c_arm=on
dtparam=i2c_arm_baudrate=50000
#dtparam=i2s=on
#dtparam=spi=on
```

(3) I2C の有効化

Raspberry Pi の設定 — インターフェイス で I2C を有効にしてください



(4) test プログラムを実行する

ダウンロードしたプログラムを解凍し、/home/pi の直下に LHAT というフォルダーを作成してその中に保存する。その中には下記の様な Python のプログラムが入っているので順番に実行して動作確認を行う（実行例：python test1.py）。

test1.py : LED チェック

LED が赤→黄→緑の順番に点灯・消灯します。

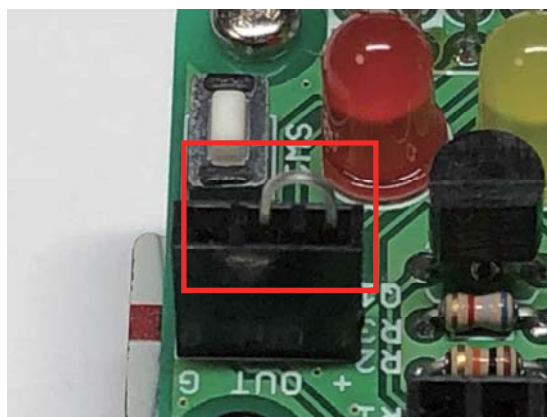
test2.py : ブザーチェック

ブザーから音楽（音）が流れます。

test3.py : LCD チェック

LCD のバックライトが点灯し、LCD に「Welcome IoT LHAT」と表示されます。

test4.py : CN 2（人感センサーの確認）



写真のように CN2 の OUT と + を短絡（ショート）させて実行すると LED が赤→黄→緑の順番に点灯・消灯します。

test5.py : SW1 チェック（タクトスイッチの使い方の確認）

プログラムを実行させて SW1（白い SW）を押すと LED が赤→黄→緑の順番に点灯・消灯します。

その他下記の様なファイルが同梱されています。

display.py : LCD ライブラリー LCD を使用する時に必要なプログラム
test プログラムと同じフォルダーに入れて置いてください。
必要なプログラム (test3.py) が読み込み動作します。

shutdown-off.sh : SW2 で Raspberry Pi をシャットダウンするためのスクリプト
使用方法については同梱の「shutdown-off.sh の使用方法.txt」
を読んでください。

readme.txt : 同梱プログラムの内容一覧

(注)

test プログラムには IoT 学習 HAT の動作を確認する為の最低限のものしか入っていません。
CN1 に BME280 など I2C デバイスを挿して使用する為にはデバイスに対応した
ライブラリープログラムが必要になりますので、各自インターネット等で入手してください。

(例 https://github.com/adafruit/Adafruit_Python_BME280)

CN1 に I2C デバイスを接続する場合はデバイス側のプルアップ抵抗を外してください。

test プログラムが動かなかった場合のトラブルシューティング

(過去の動作不良の体験より)

(1) まずハンダ付けがしっかりされているか確認してください。

はんだ不足やいもはんだがないか注意してください。

(2) test1.py が動かない場合

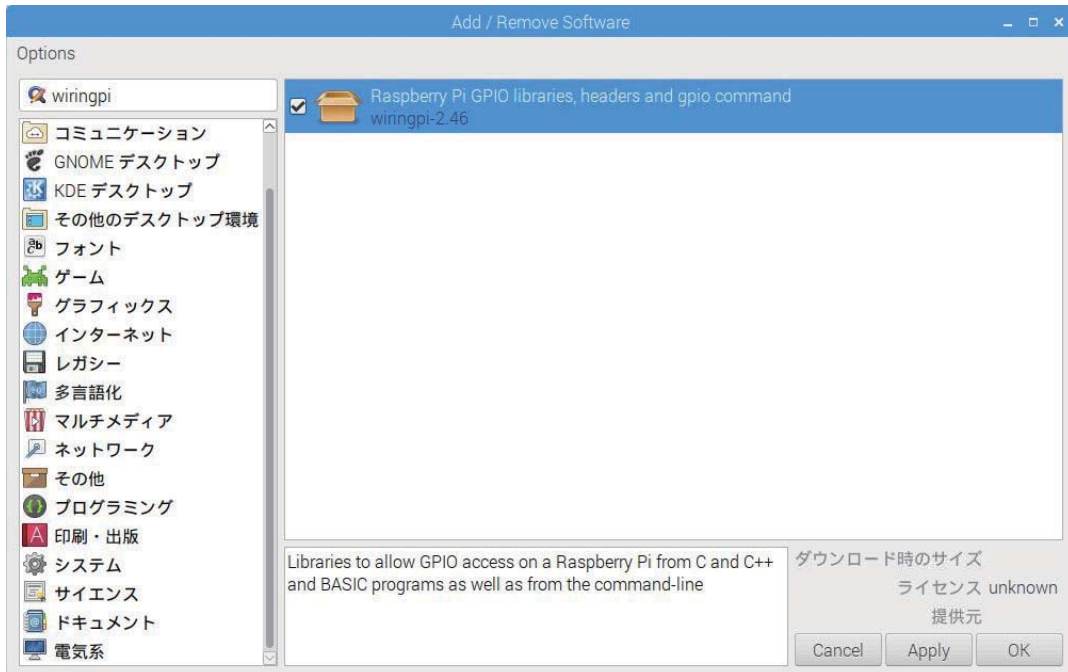
1. LED の挿入方向が合っているか確認してください。

(部品面の白いマークと向きが一致しているか)

(3) test2.py が動かない場合

1. Wiringpi がインストールされているか確認してください。

設定→ADD/Remove Software で Wiringpi を検索してみてください。



見つからない場合やどうしても動かない場合は再度インストールし直してみてください。

2. ブザーのはんだ付けを確認してください。

(4) test3.py

1. バックライトが点灯しない場合は R4 のはんだを確認してください。

2. 文字が表示されない場合は I2C が動作しているか確認してください。

SSH もしくは VNC 等のターミナルで

i2cdetect -y 1 と入力して下記のように 3e が表示されるか確認してください。

```
pi@space:~ $ i2cdetect -y 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  3e  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

3e が表示されない場合は I2C が動いていないので

Raspberry Pi の設定 — インターフェイス で I2C が有効になっているか確認してください。

3. 有効にしても 3e が表示されない場合は

過去に一番多かったトラブルは LCD のはんだ不良です はんだの確認を入念にトランジスタ Q 1 の向きが合っているか

トランジスタ Q 1、抵抗 R 1、LCD のはんだ付けの確認してください。

コンデンサ C1・C2・C3 のはんだの確認をしてください。
(短絡していることもあります)

(5) test4.py

CN2 の OUT (真ん中) と + (抵抗側) の端子を短絡させているか
CN2 のはんだを確認してください。

(6) test5.py

SW1 (白) のはんだを確認してください。

(7) shutdown-off.sh (shutdown-off.sh の使用方法.txt を読んでください)

SW2 (黒) を 3 秒以上押し続けると Raspberry Pi Zero がシャットダウンします。
3 秒以上押し続けると Raspberry Pi Zero のシャットダウンが開始され Raspberry
Pi Zero 緑の LED が点滅しシャットダウンが行われます。
Raspberry Pi Zero がフリースした時などに安全にシャットダウンできます。
SW2 のはんだ付けを確認してください。