

SensorPico 取扱説明書 v2.3

2023/7/21

2023/9/18

2023/9.23

2024/5/16

2024/5/30

1. 起動方法

(ア) Thonny から起動する場合

- ① THPCW.py WBGT あり
- ② THPC_ambient.py WBGT なし

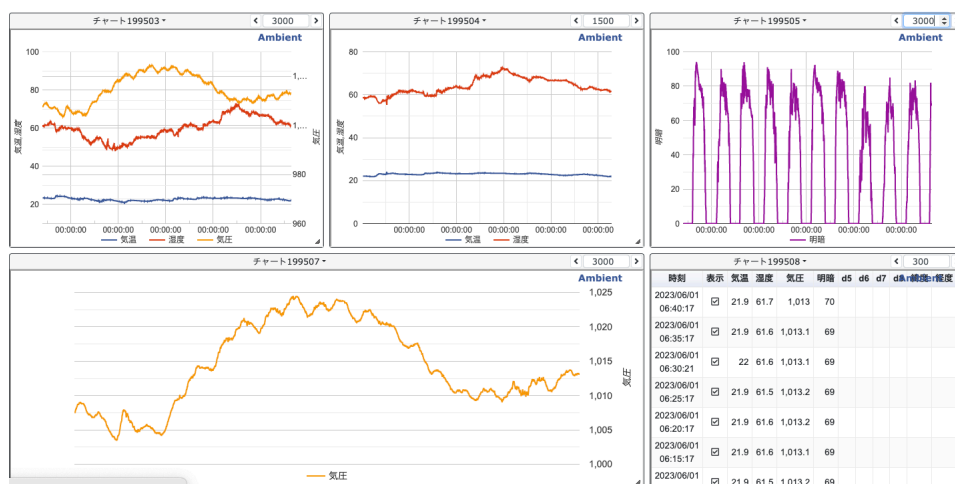
(イ) 自動起動

- ① main.py WBGT あり

2. 機能概要

(ア) RaspberryPi PicoW を使用して、気温、湿度、気圧、明暗を測定し OLED に表示するとともに ambient にデータを送信します。

(イ) Ambient で時系列にグラフ表示します。



(ウ) Ambient は無料では 3000 点のデータを表示できます。毎分データを送信すると約 2 日、5 分毎に送信すると 10 日分表示する事ができます。

(エ) 送信データについて

```
{"d1": temp, "d2": humi, "d3": press, "d4": Cds, "d5": stat, "d6": temp_cpu }
```

(オ) 上記の順番でデータを ambient に送っています。Stat と temp_cpu は状態と cpu 温度なので、必要なければ、表示する必要はありません。

(カ) 測定温度と CPU 温度の差を d7 に WBGT を d8 に追加しています。

(キ) WBGT は温度と湿度から簡易的に求めたものです。

3. インストール

(ア) インストール手順書を参照してください。

4. 設定ファイル

config.py に設定値があります。必要に応じ変更してください。

(ア) Wi-Fi_set

- ① Wi-Fi 接続をせずに動作させたい時は 0 とします。

(イ) ID_PASS

- ① Wi-Fi の ssid とパスワードを設定

(ウ) Hosei によるキャリブレーションについて

- ① センサーのオフセットがずれている場合に補正値を設定します。通常温度と湿度はあまりずれていません。ただし、気圧はずれている事があるので、キャリブレーションの必要がある場合があります。
- ② キャリブレーションについて
 - 1. センサーは個々にオフセット値がありますが、ずれている場合があります。信頼できる温度、湿度、気圧計がある場合は、測定範囲の中央値あたりで、同じような値になるように Hosei 値を求めて設定することで、キャリブレーションすることが可能です。
- ③ 信頼できる計器がない場合特に気圧を知る方法として、iPhone の場合は、siri を起動して「気圧」と言うと教えてくれます。ただし、より精度を考える場合は、気圧観測点とセンサーのある場所との高低差を考慮する必要があります。

(エ) Ambi

- ① ambient のチャンネル ID とライトキーを設定

(オ) i2c_ini

- ① センサーと OLED の i2c チャンネルと SDA のピン番号を設定
i2c は 0 と 1 の 2 チャンネルがあり今回の基板では、0 を使用
SDA は i2c0 の GP0 を使用しているので、0 のままで良いです。何かの都合で変更する場合のみ設定を変えてください。

(カ) measu_cycle

- ① 計測周期を設定 分単位です。

(キ) Cds_ini

- ① Cds の測定範囲を設定します。
- ② 測りたい一番明るい状態を Cds_max
- ③ 測りたい一番暗い状態を Cds_min
- ④ に設定します。
- ⑤ 28 は GPIO pin 番号です変更しないでください。

5. Ambient の設定

(ア) Ambient の HP にて、アカウントを取得して、設定を行ってください。

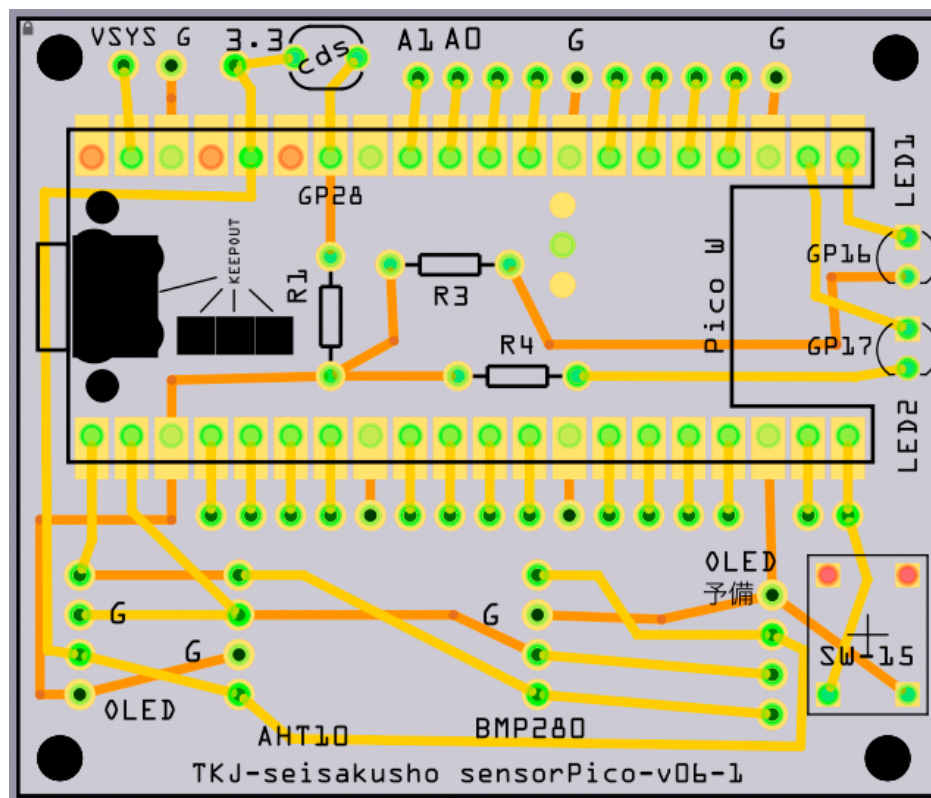
<https://ambidata.io/docs/gettingstarted/>

(イ) あと、「qiita ambient」などと検索すると丁寧に説明しているサイトがあるので、そちらを参照してみてください。

(ウ) d1:気温、d2:湿度、d3:気圧、d4:明暗となっています。

(エ) pico へのプログラムアップロード、config.py の設定、ambient の設定が正常であれば、pico を電源に接続すると自動起動し、ambient にデータが記録されます。

6. LED とスイッチ



(ア)

(イ) 接続 GPIO #

- ① LED1 #16 GPIO 1 で点灯
- ② LED2 #17 GPIO 1 で点灯
- ③ タクトスイッチ # 15 スイッチ ON で GPIO が 0 となる
 1. タクトスイッチのプルアップはソフト処理したので、R2 は未装着となる。
- ④ 気温、湿度、気圧センサーは i2c=0 GPIO は SDA:0 SCL:1
- ⑤ Cds は ADC2 の GPIO#28

7. サンプルプログラム

(ア) スイッチと LED のデモプログラム

① SWLED_DEMO.py

1. 起動すると LED が交互に点灯し、スイッチを押すと 2 つの LED が点灯する。これを繰り返します。

webApp のサンプルプログラム

② web_LED_sensor.py

1. 起動すると OLED に ip アドレスが表示される。
2. このアドレスをブラウザで表示させると

RaspberryPi Pico LED Control and Sensor



3. 上記のように表示される。LED1 ON をクリックすると LED1 が点灯し、LED1 OFF をクリックすると LED1 が消灯する。
4. センサー表示は何かのボタンを押した際に更新されます。

8. OLED の装着について



向かって左のコネクタに OLED を装着したし場合
(ただし、右側のコネクタはオプション)



右側のコネクタに OLED を装着した場合。
それぞれ基板からはみ出さないように設計しています。
(写真は旧バージョンです)

逆方向に装着すると故障の要因になりますので、間違わないようにしてください。

9. WBGT について

(ア) 温度と湿度から簡易的に WBGT を求めて ambient に投げるとともに LED を点灯させます。

暑さ指数(WBGT)について <https://www.wbgt.env.go.jp/wbgt.php> 等を参考にしてください。

(イ) LED 点灯

- | | | |
|--------|----------|-----------|
| ① 警戒 | 25 から 28 | LED1 点灯 |
| ② 厳重警戒 | 28 から 31 | LED2 点灯 |
| ③ 危険 | 31 以上 | LED1,2 点灯 |

(ウ) 暑さ指数換算表にない場合は、0 とします。ただし、上限を超えると 35 と表示。

10. ウィルス警告

(ア) 温度 20 度や湿度 45 度を下回ると LED が点灯します。

以上