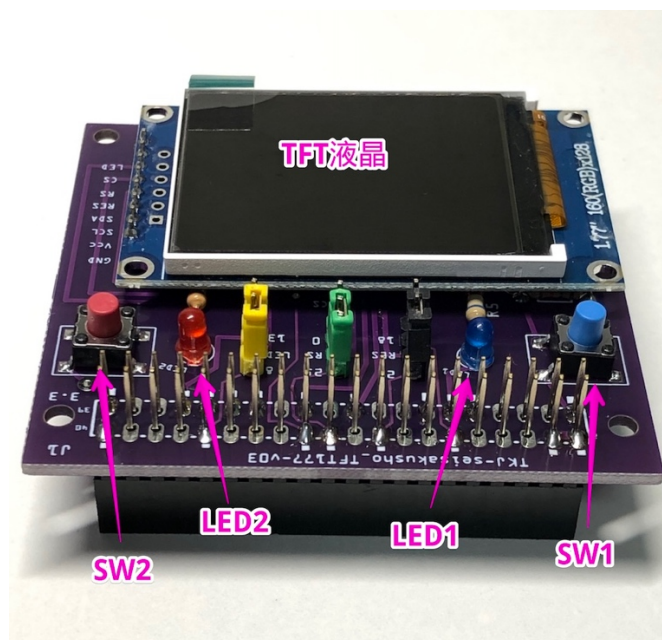


## TFT17 取扱説明. V1.1

T F T 1.77 表示器とスイッチ 2 個、L E D 2 個を実装したテスト用のラズパイ専用基板です。

ブレッドボードで何か作ろうとしたときに、最低限の U I が必要ですが、それらもブレッドボード上に作るのは面倒です。最低限の U I を備えたこの基板があれば、作るべき物だけに集中できるので、作業がはかどります。

装置名称



使用方法

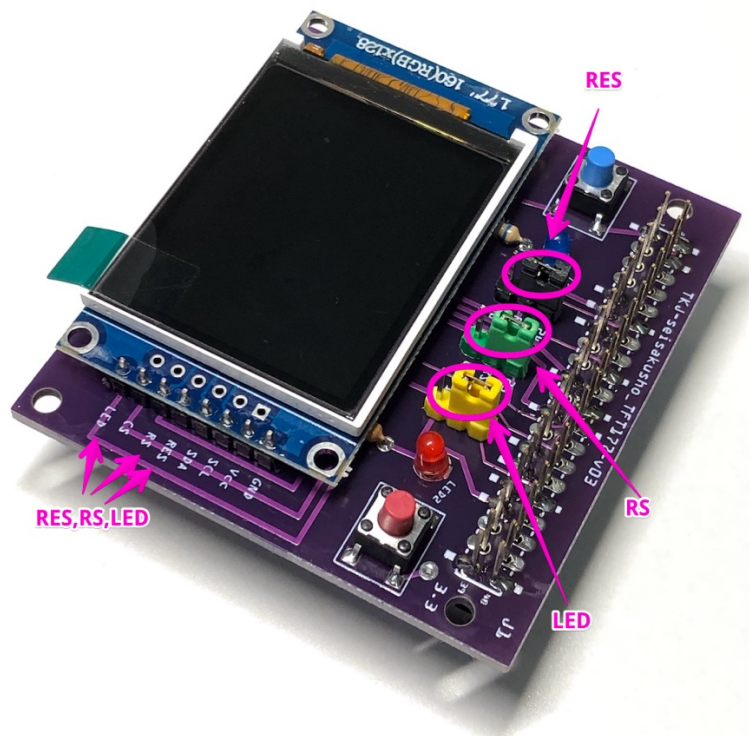
### 1. sw

- |         |         |                  |
|---------|---------|------------------|
| (ア) sw1 | GPIO #5 | 通常 0 でスイッチを押して 1 |
| (イ) sw2 | GPIO #6 | 通常 0 でスイッチを押して 1 |

### 2. LED

- |          |          |             |
|----------|----------|-------------|
| (ア) LED1 | GPIO #17 | 0 で消灯、1 で点灯 |
| (イ) LED2 | GPIO #27 | 0 で消灯、1 で点灯 |

### 3. 液晶



(ア) 液晶の信号は

- ① Vcc,GND 電源
- ② SCL,SDA データ信号線      GPIO #11,#10
- ③ RS,CS 制御信号線      GPIO 後述
- ④ LED バックライト      GPIO 後述

で構成されています。

RS,CS,LED のみターミナルブロックの位置で設定できるようになっています。SCL,SDA 信号線や SW,LED は固定です。

(イ) ターミナルブロックの位置による GPIO 接続

- ① ターミナルブロックを液晶側に設定した場合を位置 1 とします。  
GPIO 端子側を位置 0 とします。
- ② RES      位置 1:GPIO # 18、位置 0:GPIO # 24
- ③ RS      位置 1:GPIO # 0 、位置 0:GPIO # 25
- ④ LED      位置 1:GPIO # 13、位置 0:GPIO # 12
- ⑤ のように変わります。
- ⑥ 全ての信号線を変更できるわけではないですが、この 3 本だけ設定により変更が可能な設計になっています。

(ウ) この設定により使用するプログラムが変わることに注意が必要です。サンプルプログラムの末尾に 0 と 1 がついていますが、これが上記のターミナルブロックの位置と対応しています。ただし、ターミナルブロックの位置は 3 つ同時に同じ側に設定する必要があります。たがい違いに設定すると動作しません。

#### 4. サンプルプログラム

(ア) `git clone https://github.com/momorara/tft177` でラズパイにダウンロードしてください。

(イ) LED と sw のテスト用プログラム

① `test_LED.py`

② `test_sw.py`

(ウ) 表示用テストプログラム (ターミナルブロック位置 0 用)

① `test_tft_12345_0.py`

② `test_tft_abcdef_0.py`

③ `test_tft_漢字_0.py`

④ `test_tft_image_0.py`

⑤ `all_test_0.py` 上記テストを全て実行します。

⑥ `lcd177_0.py` 液晶に表示するためのライブラリ

(エ) 表示用テストプログラム (ターミナルブロック位置 1)

① `test_tft_12345_1.py`

② `test_tft_abcdef_1.py`

③ `test_tft_漢字_1.py`

④ `test_tft_image_1.py`

⑤ `all_test_1.py` 上記テストを全て実行します。

⑥ `lcd177_1.py` 液晶に表示するためのライブラリ

(オ) 全て python3 で書かれています。

`$ python3 test_tft_12345_0.py` などとして実行してください。

#### 5. ライブラリ `lcd177_0.py`、`lcd177_1.py` の使い方

(ア) 使用するプログラムと同じディレクトリに配置し

`import lcd177_0` or `import lcd177_1` として、インポートし  
関数として呼び出して使います。

(イ) 文字系(以下ターミナルブロック位置 0 用で説明)

`lcd177_0.init('on')` として表示開始 バックライト点灯

`lcd177_0.disp('message')` として表示する

`lcd177_0.size(16)` としてフォントサイズ指定 デフォルト 12

lcd177\_0.color('green') として文字色指定 デフォルト 白 ,#0000FF も可能

lcd177.\_0init('off') として表示終了 バックライト消灯

lcd177\_0.init('reset') として液晶をリセット

lcd177\_0.disp('message',size,color)size 指定、色指定も可能

color は white,blue,red,green が使える

#0000FF といった指定も可能

ただし、色、サイズ、message のエラーチェックはしていないので、要チェック  
(ウ) イメージ

lcd177\_0.dsp\_file(dsp\_file) として dsp\_file にファイルのパスを入れて渡す。

#### 6. 基板バージョンによる違い

(ア) 2024/2/10 現在基板のバージョンは v03-2 まであります。

(イ) しかし、全てのバージョンで上記プログラムは使用できますし、機能、性能に違いはありません。仕様変更は製作しやすくするための若干の配置変更、そして今後のキット販売に向けての簡素化が目的です。

(ウ) 特にキット販売に向けての簡素化では、ハード的にプルダウン抵抗をソフトウェアによるプルダウンに変更することにより部品点数を減らしています。

(エ) 途中のバージョンでは基板にパターンがあるのに抵抗が乗っていないバージョンがありますが、出荷時にテストしていますので、動作に問題はありません。

(オ) ただし、プログラムは最新版を使用してください。

以上