

Web_RemotePicoW 説明書

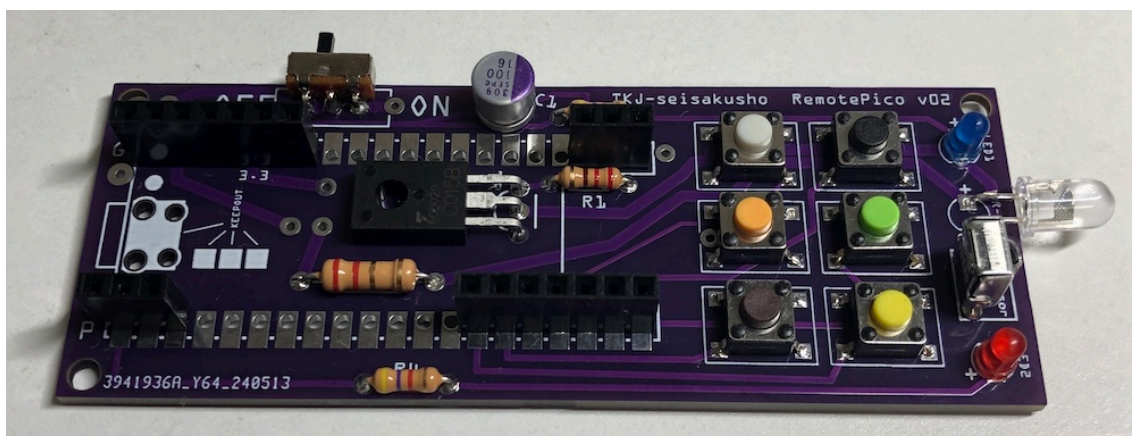
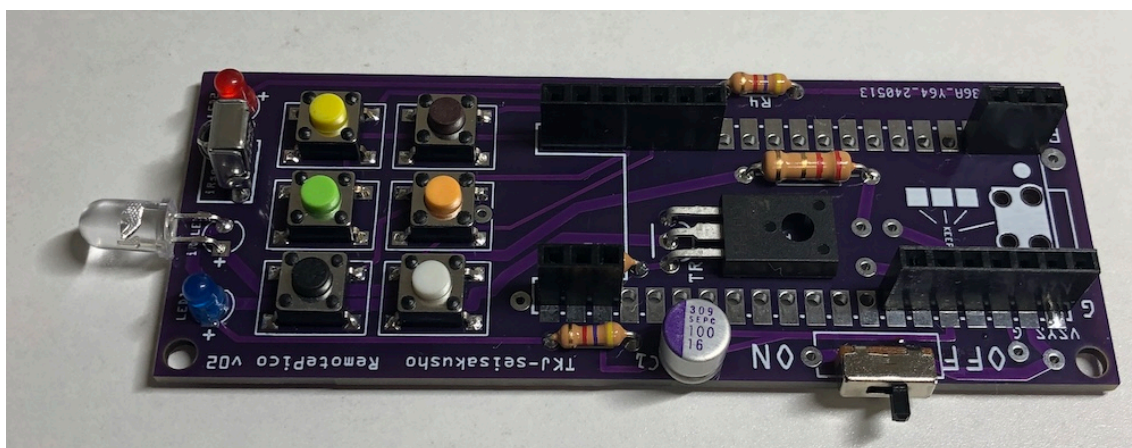
v1.0 2025/5/1

- ・本説明書は RemotePico をインターネット上の Web アプリから制御する Web_RemotePicoW の説明書です。RemotePico については、当該説明書を参照ください。
- ・元のプログラムが MIT License で提供されているので、本装置のプログラムも MIT いたします。

*** 注意事項 ***

(注) 電池を入れた状態で **USB** 接続しないでください。発熱、破損の恐れがあります。

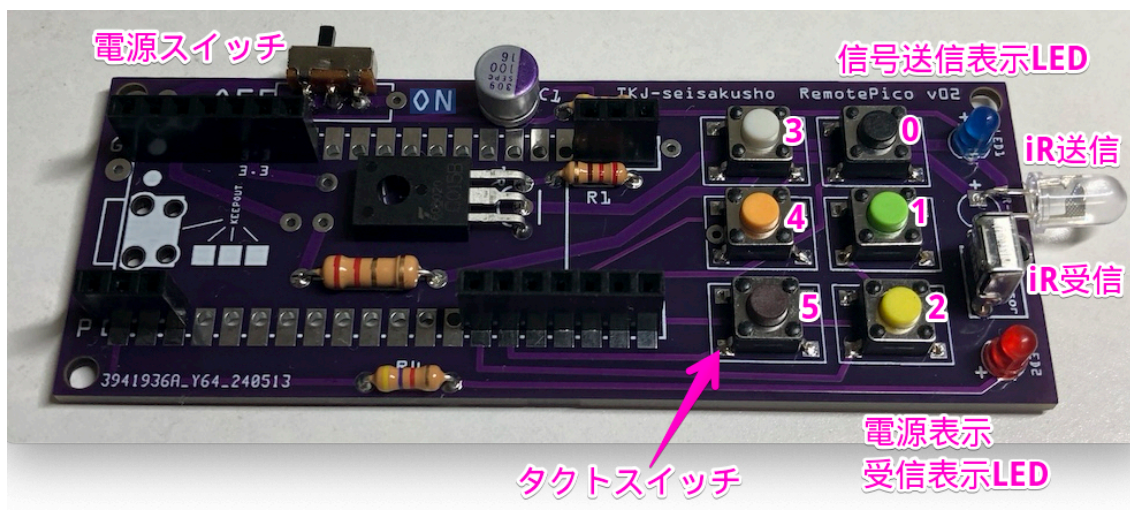
RemotePico 完成写真



スタントアローンで使用する場合のプログラムを GitHub にアップしています、

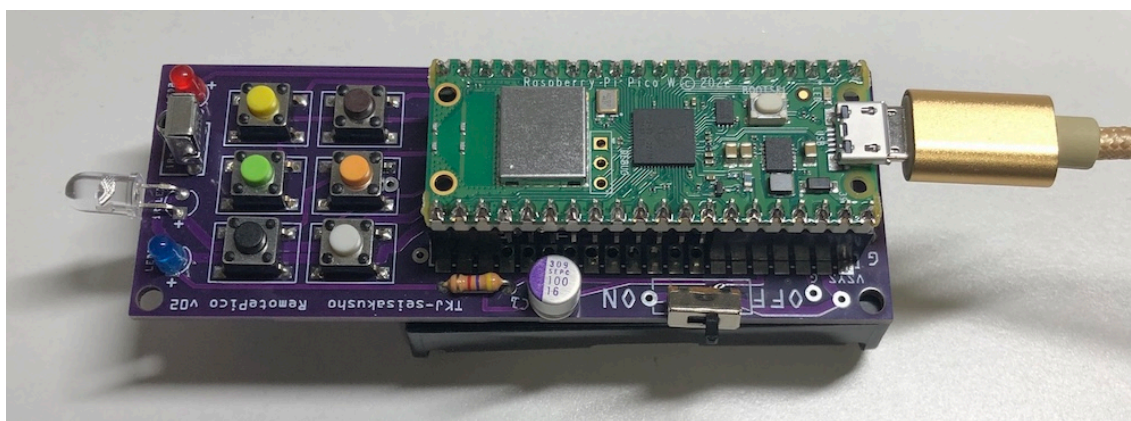
<https://github.com/momorara/RemotePico>

本説明書は、Web アプリを使用する場合の説明をメインとします。



本プログラムは常時 Wi-Fi 接続し、mqtt ブローカーと通信しますので、電池駆動ではなく AC アダプタ接続を想定しています。

使用方法



- プログラムをインストールした picoW をセッティング
- 電源スイッチを off のままとし、電池は接続せず、USB-microB に電源を接続
- タクトスイッチ 0,1,2,3,4,5 にリモコン信号を登録することが可能です。
(ア) 登録したい番号のタクトスイッチを長押しする。
(イ) 赤 LED が点灯する。

- (ウ) iR 受信センサーに向けて学習したいリモコンを向けて、学習したいボタンを押す。
- (エ) 20cm 程度離して行うとうまくいきやすい。
- (オ) 試しに登録したボタンを、短く押す → リモコン信号が送信され青 LED が点滅。

以上のようにスタントアローンと同等の使い方ができるとともに、以下説明の通り Web アプリからの mqtt 信号で制御することが可能になります。

Web_RemotePicoW のプログラムは

こちらのリポジトリに保存されています。

<<本プログラムを使うにあたっての準備>>

事前準備として、picoW にスタンドアローンのプログラムをアップロードして、一通り使用できることを確認しておいてください。

通常 web アプリを立ち上げるためにはレンタルサーバーを借りて html や javascript なりを使って手間のかかることをする必要がありますし、レンタルサーバーを借りるのも費用が必要となりますが、今回の方法は、全て無料サービスをつかって、ランニングコスト 0 円で運用できる方法です。ただし、使用する無料サービスへのサインアップや設定の手間が必要となります。

a. gitHub というサービスにサインアップする

(ア) こちらは、web 上でプログラムの管理をするサービスです。streamlit で使用する python プログラムを管理するのに使用します。

b. streamlit というサービスに gitHub のアカウントでログインする

(ア) こちらは、3 つまでの web アプリを無料で運用できるサービスで、さらに python のみで https で web アプリが作れるという大変便利なサービスです。

c. gitHub アカウントと Streamlit の連携を求められるので「Authorize」します。これにより、gitHub でプログラムを編集すると自動的に streamlit にデブロイされ web アプリが更新されます。

以上の 3 点を行なっていただく必要があります、ちょっとしたハードルがあります。ただし、これにより無料で Web アプリが運用できますし、python のみで web アプリが作れるというのですから非常に有用なサービスだと思っています。

<<制約事項>>

ランニングコスト無料で使う関係で以下の制約事項があります。

- a. streamlit の web アプリを 12 時間使用しなかったら、スリープ状態になり立ち上げるのに数秒から数十秒かかります。

(ア) <https://streamlit-docs.netlify.app/deploy/streamlit-community-cloud/manage-your-app>

- b. 通信に mqtt を使用しますが、無料のブローカーを使います。streamlit も gitHub もですが、無料で使うので、いつサービス内容が変更、もしくは停止されるかわからないリスクがあります。その場合は、別のサービスを探すか、有料サービスを使うかする必要があります。

- c. ・プログラム上の制約ですが、mqtt の通信に少し時間がかかるので、PM11:59 頃に操作すると動作しない場合があります。

・mqtt のブローカーへの問い合わせ等でネットの状況でフリーズした場合、WDT でリセットしますので、不定期に 1 分程度操作できない場合があります。

・mqtt の信号がネットの不調等で届かない場合があります。

以上のように操作が確実に実行される保証はないことを承知の上使用してください。

Streamlit サインアップ方法 (Community Cloud)

Streamlit を使って Web アプリをクラウド上に公開したい場合、Streamlit Community Cloud にサインアップします。

手順：

1. Web ブラウザで開く：

<https://streamlit.io/> にアクセス

2. 右上の「Sign in」または「Get Started」ボタンをクリック

3. GitHub アカウントでログイン：

GitHub アカウントが必要 (Streamlit Cloud は GitHub と連携してデプロイします)

4. GitHub 連携を許可：

GitHub アカウントと Streamlit の連携を求められるので「Authorize」します。

5. ダッシュボードに入る：

ログイン後、Streamlit のダッシュボードが表示され、そこから自分の GitHub リポジトリを選んでアプリを公開できます。

- ・サインアップに必要なもの

GitHub アカウント（事前に作成しておくともスムーズ）

アプリを GitHub に push しておく（app.py など）

- ・アプリ公開までの流れ（簡単な例）

App.py を作る

GitHub にアップロード

Streamlit Cloud にログイン

「New app」ボタンからリポジトリを選ぶ

main ブランチと app.py を指定して「Deploy」

アプリを公開する際の URL を指定できるので、指定する。

ある程度複雑なものにしておくとも良いです。この URL がわかれば、誰でもこのアプリを使う事が可能になってしまいます。

<<プログラムのセッティング>>

PicoW 側

- ・ main.py

- ・ send_file.py スタンドアローン時とおなじもの

- ・ UpyIrRx.py スタンドアローン時とおなじもの

- ・ UpyIrTx.py スタンドアローン時とおなじもの

をアップロード

ただし、main.py の設定行なっておく必要があります。

- ・ 設定内容

暗号キー：半角アルファベット a-z を並べ替えたもの

web アプリ側と同じものを設定

セキュリティコード：半角英数字で 4 から 8 文字程度

mqtt トピックス：初期設定値を必ず変えること

独自にユニークとなるように設定する

web アプリ側と同じものを設定

streamlit 側(web アプリ側)

- ・ web_RemotePicoW.py

mqtt トピックス：初期設定値を必ず変えること

暗号キー：半角アルファベット a-z を並べ替えたもの

スイッチ名称：必要であれば変更する

<<セキュリティについて>>

- ・mqtt の通信は平文で行なっています。そのためインターネット上で盗聴されたりコピーされたりします。単純に送信しているとコピーされると誤操作を起こされる可能性があるため、暗号キーで暗号化しており内容は分かりにくくなっています。また送信したタイミングで日時を含んで送っているため、単純にコピーしても誤動作は起こらないようになっています。

- ・web アプリは URL さえ判れば、誰でもボタンを押せてしまいますが、セキュリティコードを入力しないと picoW で弾かれるので、動作はしません。表示は数字と書いてありますが、半角英数字をセキュリティコードとして使えるので、分かりにくいものにしてください。

- ・picoW 側のセキュリティコードなど設定値の一部は config.py に保存しています。

Streamlit Cloud を無料版で使用する際には、GitHub のリポジトリをパブリックで公開する必要があります、GitHub や Streamlit を知っている人からはソースコードが丸見えとなります。そのため `henkan` と `topic` については Streamlit Cloud の `secrets.toml` に保存するようにしています。ここは、Streamlit Cloud にアクセスできる人しか見えない領域になっています。

<<Web アプリ画面の説明>>

温度: 24 / 湿度: 45

温度と湿度はダミー

Input date1

2025/04/29

Input date2

2025/04/29

この日付をキーとして
送るので、日が変わると
リロードしてください。

SW-0 @ RemotePico

SW-1 @ RemotePico

SW-2 @ RemotePico

SW-3 @ RemotePico

SW-4 @ RemotePico

SW-5 @ RemotePico

各スイッチと対応した
ボタンになっています。
名称は変更可能です。

False False False False False False

セキュリティコードを数字6桁で入力してください。

英数字が使えます。
入力しないと picoW で
弾かれます。

入力内容:

reset

<<picoW 側の操作・表示>>

- ・6つのタクトスイッチの使用方法是、RemotePicoと同じです。

・Web_RemotePicoW としての動作

電源をつなぐとボード上のLEDが点滅して、wifi接続、mqttブローカーとの接続を行ったのち、約5秒周期でボード上のLEDが短く点灯します。この状態でmqtt信号待ち状態となります。LEDが点滅しなかったり、点灯しっぱなしになった場合は、異常状態なので、USBを抜き差しするなどしてください。

<<Streamlit Cloud>>

Streamlit Cloud について説明しているサイトがありますので、必要に応じてそちらを参照
ください。

<https://toukei-lab.com/streamlit>

YouTube

<https://www.youtube.com/watch?v=xij69Qa1Qco>

(連絡先)川端 tkj-works@mbr.nifty.com

以上