

Raspberry Pi Pico W

その他のサンプルプログラム ユーザーガイド

Version 1.0

2024/2/4



目次

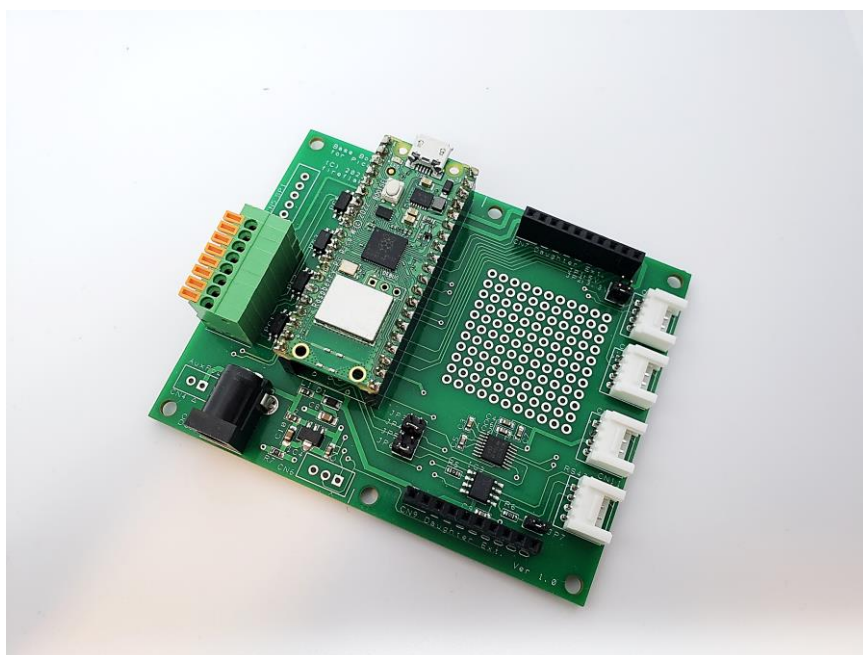
1. 始めに	3
2. サンプルプログラムの概要	3
3. プログラムのインストール方法	4
3.1. PC から Pico W に接続してプログラムの読み書きが出来るようにする	4
3.2. Thonny などの IDE から Pico W にプログラムを書き込む	6
4. プログラムの実行方法	9
5. 各プログラムの説明	10
5.1. Google スプレッドシートにデータを送信する	12
5.2. LINE Notify にデータを送信する	18
5.3. UECS (というプロトコル) でデータを送受信する	21
6. ライセンスについて	22
7. おわりに	22

1. 始めに

この資料は、fireflake のボードなしで、Pico W 単独でも動くサンプルプログラムのユーザーガイドです。主には、何らかの Web システムなどと Pico W を連携させるためのサンプルプログラムになります。

という意味では、私たちが書く必然性が低い内容なのかもしれませんが、ただこういったサンプルもある事でユーザーの方々が作れるものの幅が広がるように思い、こういった資料としてまとめてみました次第です（お役に立てば幸いです）。

とはいえ説明の都合上、サンプルプログラムを使う際には、お手元に、Raspberry Pi Pico W, fireflake 製ベースボードの 2 点があり、下の画像のように組み立ててあるものとします。



基本的に本資料の情報が正となりますが、サンプルプログラムのバージョンによっては、本資料の説明と動作が異なるかもしれません。その場合にはサンプルプログラムそれ自体を正とします。

2. サンプルプログラムの概要

こちらのサンプルプログラムは MicroPython で書かれています。MicroPython は組み込み向けに最適化された Python です。Pico W では C/C++ も動作しますが、MicroPython の方が手軽に使う事が出来るため、これを採用しました。お好みに応じて C, C++ を使うことも、もちろん可能です。

サンプルプログラムには、単機能のプログラムのみが含まれます。

単機能のプログラムとは、ベースボード上のフォトカプラから電圧出力を行う、あるいはベースボードに接続したセンサーのデータを取得する、といったシンプルなプログラムです。

サンプルプログラムは、プログラム本体と関連ライブラリーを Pico W にインストールするだけで動作します。

3. プログラムのインストール方法

ここでは、サンプルプログラムを Pico W にインストールする方法を説明します。

3.1. PC から Pico W に接続してプログラムの読み書きが出来るようにする

この方法については、Raspberry Pi 公式サイトの説明が、（英語ですが）分かりやすくまとまっています。

<https://www.raspberrypi.com/documentation/microcontrollers/micropython.html>

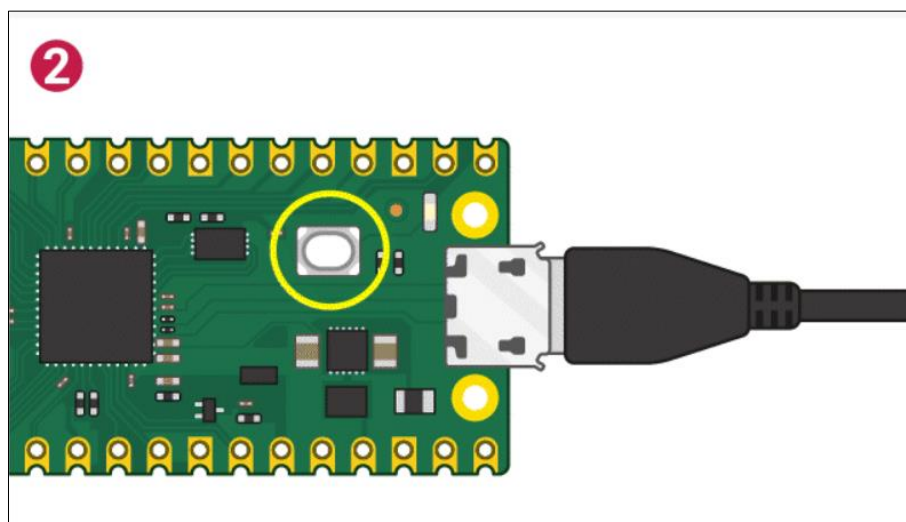
以下は、公式サイトの説明を抜粋したものになります。

まず、公式サイトから UF2 ファイルをダウンロードしてください。「Download the correct MicroPython UF2 file for your board:」と書いてある箇所で「Raspberry Pi Pico W」のダウンロードリンクをクリックしてください。

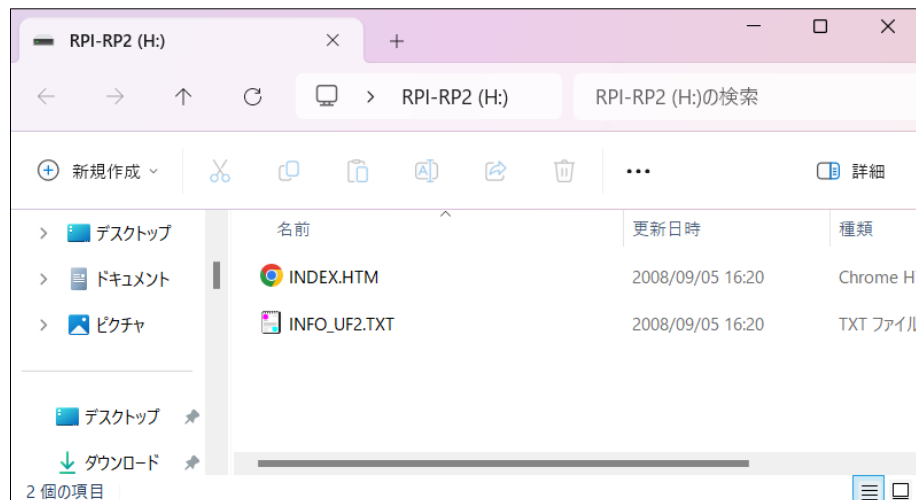
Download the correct MicroPython UF2 file for your board:

- Raspberry Pi Pico
- Raspberry Pi Pico W with Wi-Fi and Bluetooth LE support

Pico W の BOOTSEL ボタンを押したままにして、PC の USB ポートに接続します。接続出来たら、BOOTSEL ボタンを
はなします。



Pico W が、RPI-RP2 という大容量ストレージデバイスとして認識されます。Windows の場合、RPI-RP2 のエクスプローラー画面が開きます。



さきほどダウンロードした MicroPython UF2 ファイルを RPI-RP2 のエクスプローラー画面にドラッグ&ドロップします。すると Pico W が再起動して、内部で MicroPython が実行され、USB シリアル経由でプログラムの読み書きと実行ができるようになります。

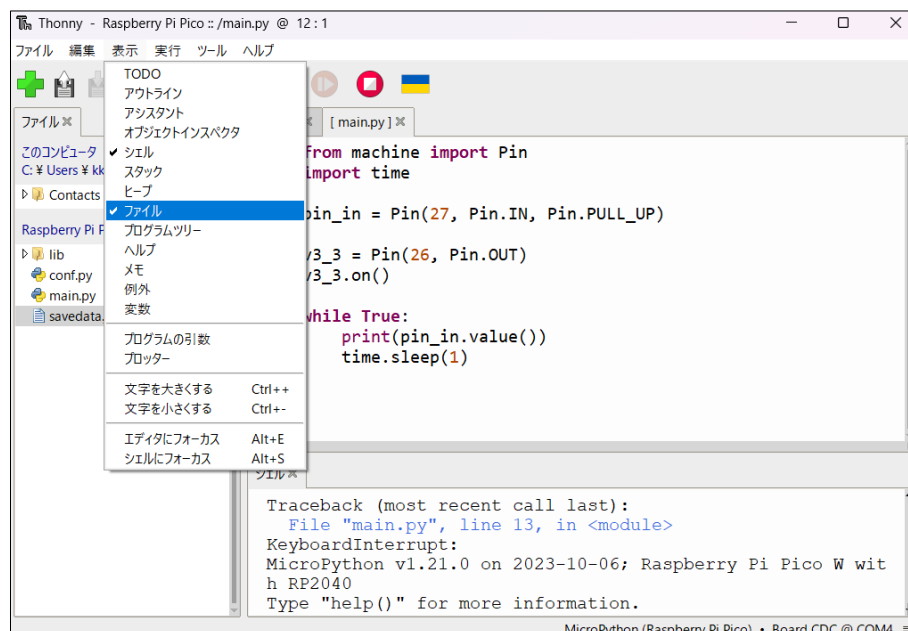
3.2. Thonny などの IDE から Pico W にプログラムを書き込む

サンプルプログラムを Pico W に書き込むには、MicroPython 向けの IDE を使うと便利です。ここでは Thonny を例にして説明します。

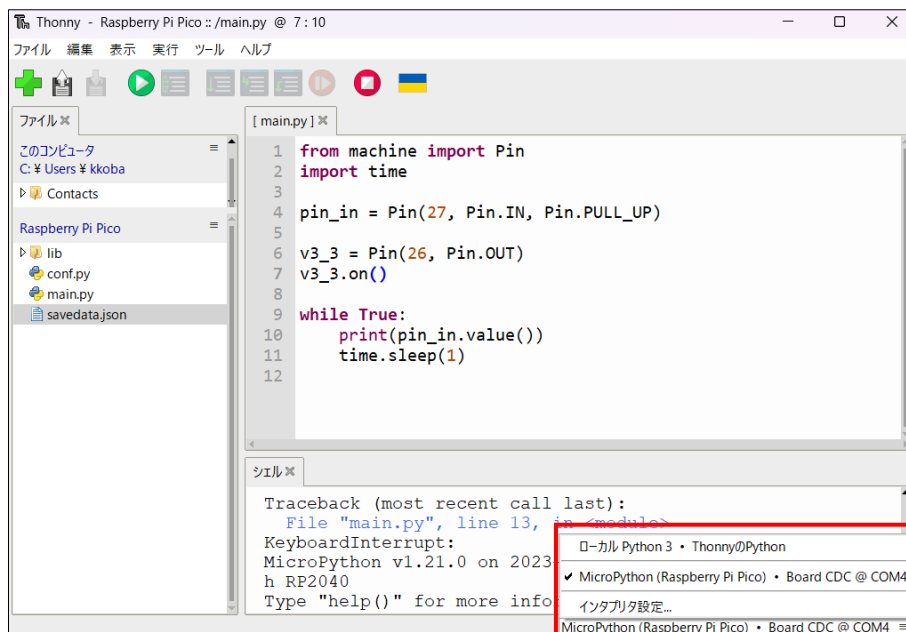
Thonny 公式サイト(<https://thonny.org/>)

Thonny については Web 上に使い方やインストール方法の情報が多くあるので、ここではその説明は行いません。ここでは Thonny のインストールと起動が済んでいるものとして説明します。

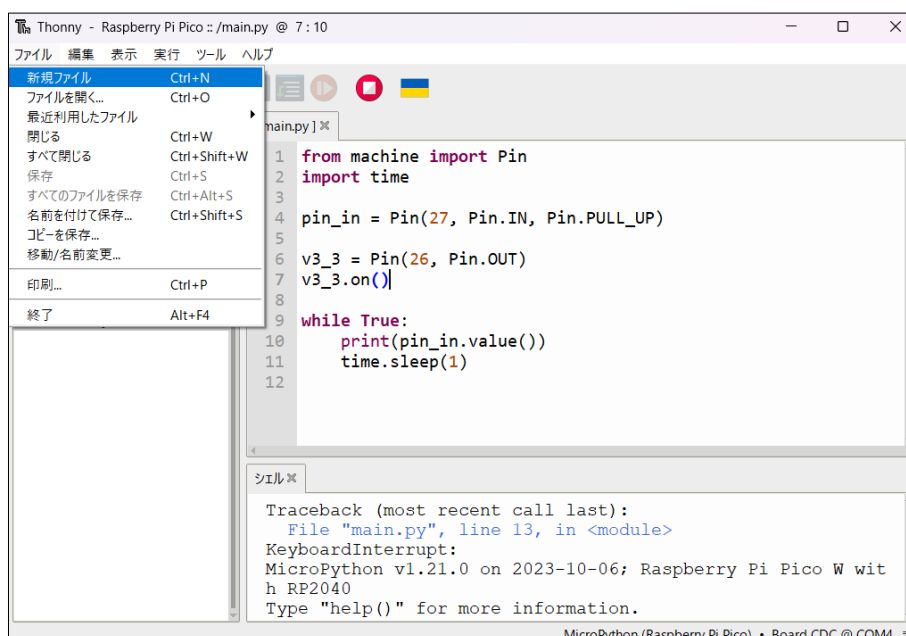
Thonny の表示メニューを開いて、ファイル表示にチェックを入れてください。すると左側にファイルツリーが表示されます。ここでファイルツリーの中に、先ほど PC に接続した Pico W が「Raspberry Pi Pico」として表示されているので、それを確認してください。



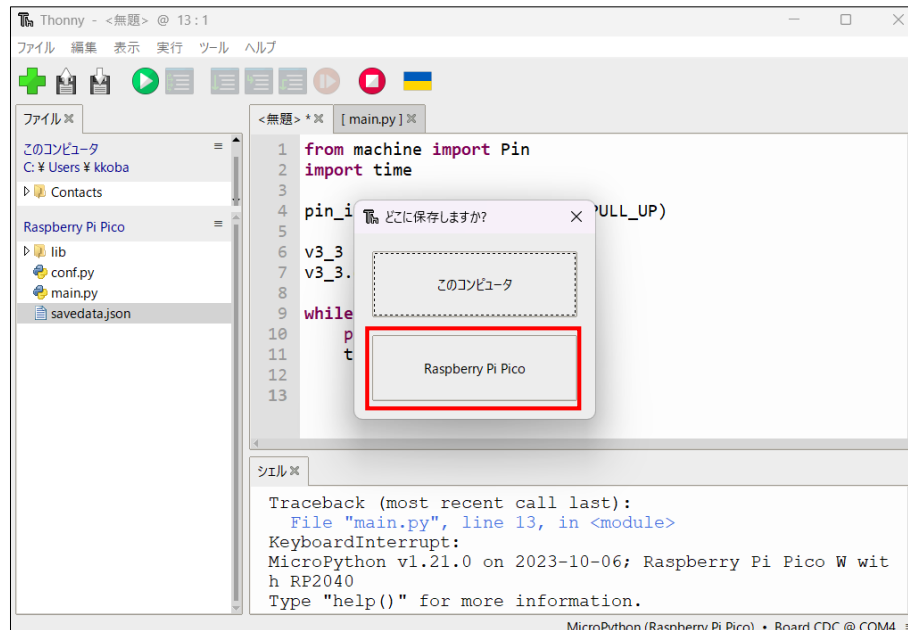
もしもない場合は、Thonny 右下のシリアルポート割り当てメニューを開いて、Pico W が接続されているシリアルポートを選択してください。



Pico W がいる事を確認したら、ファイルメニューから「新規ファイル」を選びます。無題のファイルが開くので、インストールしたいサンプルプログラムのコードをそこにコピーします。



無題のファイルを、サンプルプログラムのファイルと同じ名前を付けて Pico W に保存します。これを繰り返して、動かしたいサンプルプログラムのコードをすべて Pico W に保存してください。lib フォルダの作成が必要なプログラムの場合は、ファイルツリーの中で右クリックして「新しいディレクトリ」メニューを選び、lib と名付けて OK ボタンを押すと lib フォルダが出来ます。サンプルプログラムをすべて Pico W に保存したらインストールは完了です。

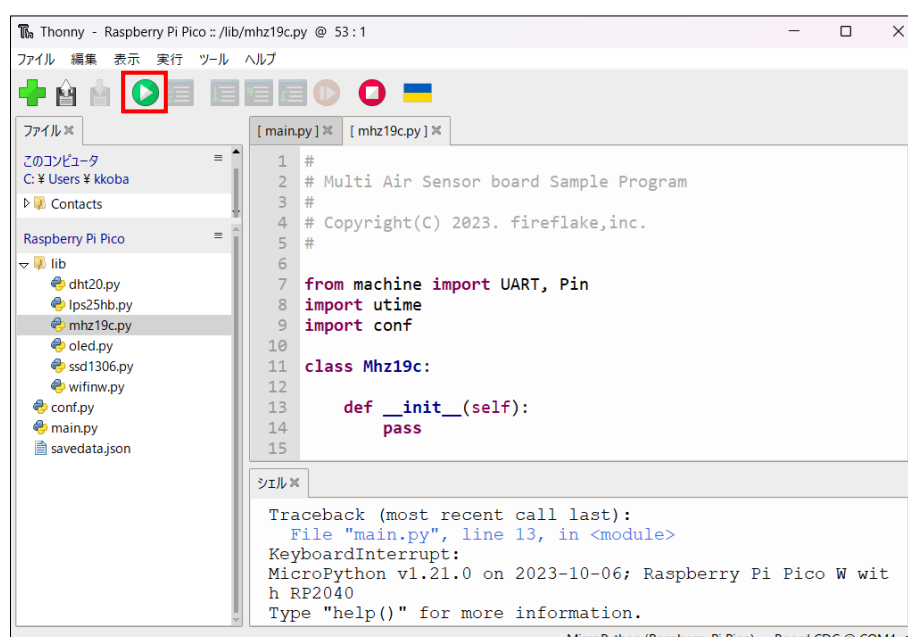


4. プログラムの実行方法

ここでは、サンプルプログラムを Pico W 上で実行する方法を説明します。

プログラムの実行には 2 種類の方法があります。ひとつは Pico W と Thonny を接続して、Pico W に入っているプログラムを Thonny で開いて編集して保存し※、その状態で実行ボタンを押す方法です。これを行うと Pico W 上でそのプログラムが実行されます。

※：Pico W に入っているプログラムを Thonny で開いて編集し、保存せずに実行ボタンを押すと、そのプログラムが Pico W のフラッシュメモリに書き込まれたうえで実行されます。ただし複数のファイルを編集してひとつも保存せずに実行ボタンを押すと、フロントに表示されているプログラムだけがフラッシュメモリに書き込まれて実行されます。



もうひとつは、Pico W にただ電源投入する方法です。分かりやすいイメージとしては、Pico W が載ったベースボードに AC アダプタから電源供給して、独立電源で動作させるというものです。これを行うと、自動的に Pico W にインストールされたプログラムが実行されます。この場合、プログラムは必ず main.py が実行されるという制約がある※ので注意してください。またこの方法で動かすと、エラーやログはどこにも表示されません。エラーやログを確認するには、それらを何らかのデバイス（例えば OLED）に表示するなどの仕組みを組んだうえでプログラムを実行する必要があります。

※：boot.py という名前のファイルが Pico W にインストールされていれば、それは main.py より先に動作しますが、べつの目的があって用意されている仕組みなので、通常は main.py から実行してください。先に動作します。

5. 各プログラムの説明

ここからは、各サンプルプログラムの説明をします。

サンプルプログラムは、ff-pico-examples/others/src/以下に、下記の構成で置いてあります。

src

└google_sheets_send.py

└line_notify_send.py

└uecs_send.py

└uecs_recieve.py

サンプルプログラムの一覧は下記になります。

- Google スプレッドシートにデータを送信する
- LINE Notify にデータを送信する
- UECS（というプロトコル）でデータを送信する
- UECS（というプロトコル）でデータを受信する

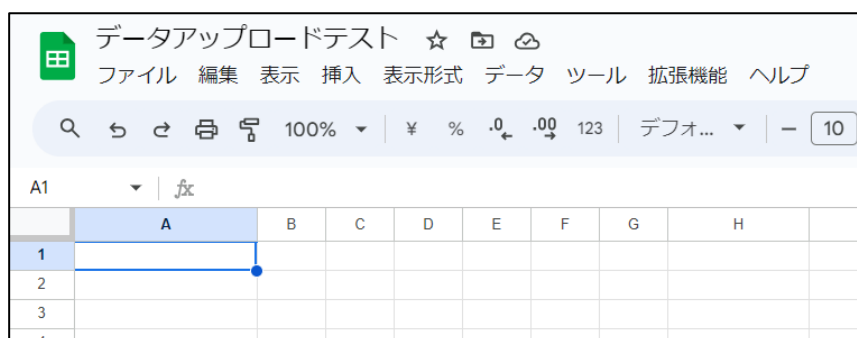
※注意

ベースボードで開発やテストを行う際に、配線を間違えたり過大な電圧をかけると故障に繋がるリスクがあります。

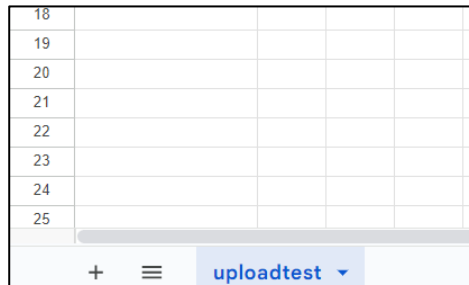
実際にベースボードにセンサーモジュールなどを接続してサンプルプログラムを動作させる前に、あらかじめベースボード取扱説明書とベースボード仕様書をよく読んで、安全に使っていただけるようお願いします。

5.1. Google スプレッドシートにデータを送信する

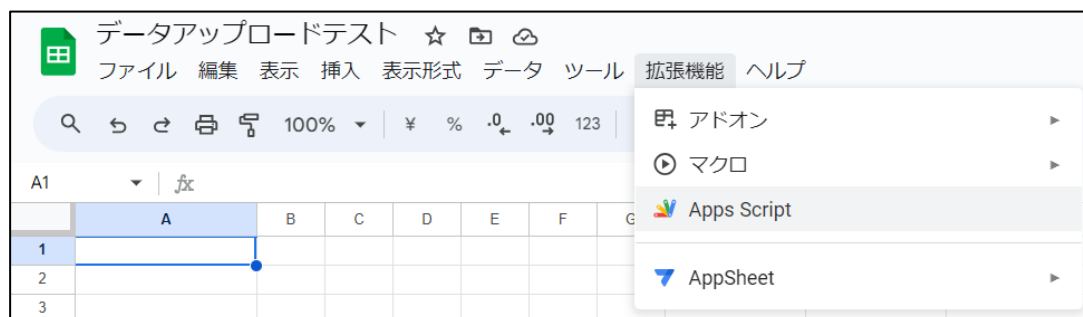
- このプログラムは、無料で使えるスプレッドシートの Google スプレッドシートと連携し、そこにダミーデータをアップロードするプログラムです。Google スプレッドシートについては、Web などに多数情報があるので説明は割愛します。また Google スプレッドシートは、fireflake にとって外部のサービスになるため、それに関するご質問などをいただいても応じられないケースがあります。すみませんが、その点はご了承ください。
- Google スプレッドシートに Pico W からデータを送るには、Google スプレッドシート側の準備と、Pico W 側の準備がいます。まず Google スプレッドシート側の準備から説明しますね。
- まず新しいスプレッドシートを作成してください。名前はなんでも良いですが、今回は「データアップロードテスト」とします。



- シート名を変更します。「uploadtest」とします。



- 拡張機能メニューから、Apps Script をクリックします。



- プロジェクト画面が現れるので、プロジェクト名を「受信データ書き込みプログラム」とします。



- スクリプトエディタに、Pico W から受信したデータを、さきほど作成したスプレッドシートの uploadtest シートに書き込むためのプログラムを書きます。プログラム全文は以下の通りです。このプログラムの書き方について知りたい方は Google Apps Script とか GAS などのワードで Web 検索をしてみてください。

```
function doGet(e) {
  // Pico W から送られたデータを取り出す
  var param1 = e.parameter.data1;
  var param2 = e.parameter.data2;
  var param3 = e.parameter.data3;

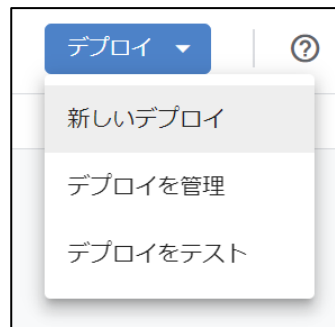
  // uploadtest のシートを取得
  var sheetName = 'uploadtest'
  var spreadsheet = SpreadsheetApp.getActiveSpreadsheet();
  var sheet = spreadsheet.getSheetByName(sheetName);

  // シートの 1 列目にデータ受信したときの時刻、2～4 列目に Pico W から受信したデータを追記する
  sheet.appendRow([new Date(), param1, param2, param3]);
}
```

- フロッピーディスクのアイコンを押して、プログラムを保存します。



- ウィンドウの右上の方にあるデプロイメニューから、新しいデプロイをクリックします。



- 新しいデプロイと書いてあるポップアップが現れます。種類の選択のところにある歯車のアイコンを押すとプルダウンメニューが現れるので、ウェブアプリをクリックします。



- 設定を以下のように変更し、デプロイボタンを押します。

設定

説明

新しい説明文

Pico Wからのデータを受信するプログラム

ウェブアプリ

次のユーザーとして実行:

自分

このウェブ アプリケーションを実行するために、あなたのアカウント データを使用することを許可します。

アクセスできるユーザー

全員

ライブラリとしても利用できます。 [詳細](#)

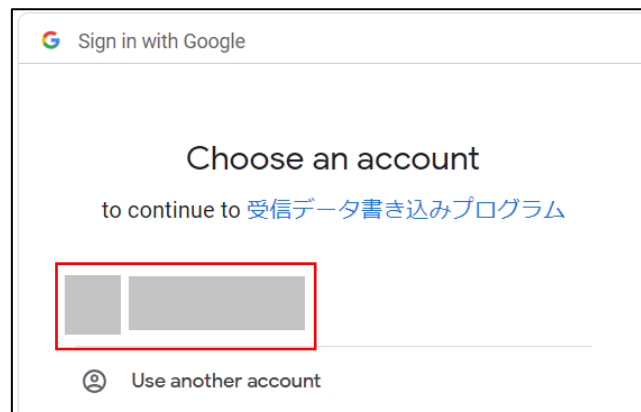
キャンセル

デプロイ

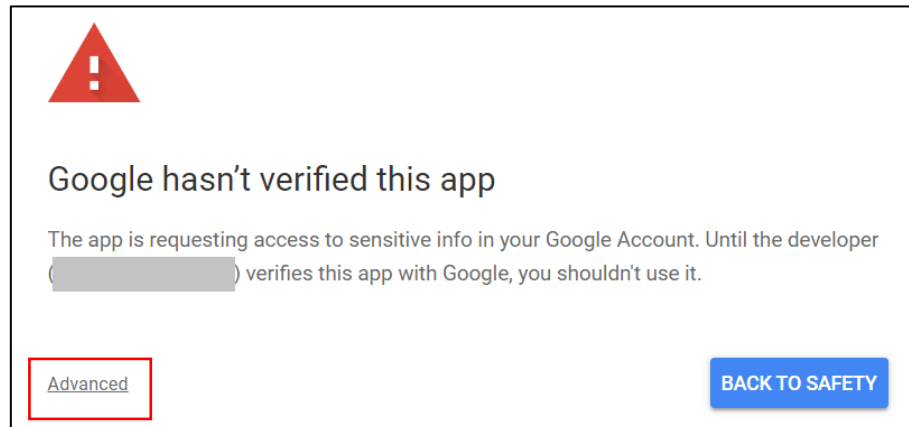
- アクセス承認ボタンが現れるので押します。現れるには少し時間がかかる事があります。



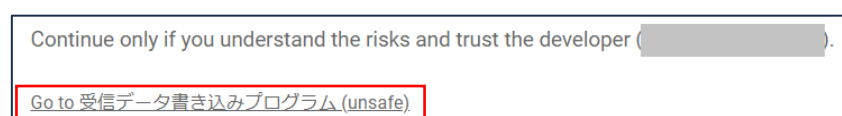
- アカウント選択のポップアップが現れるので、自分のアカウントを選択します。



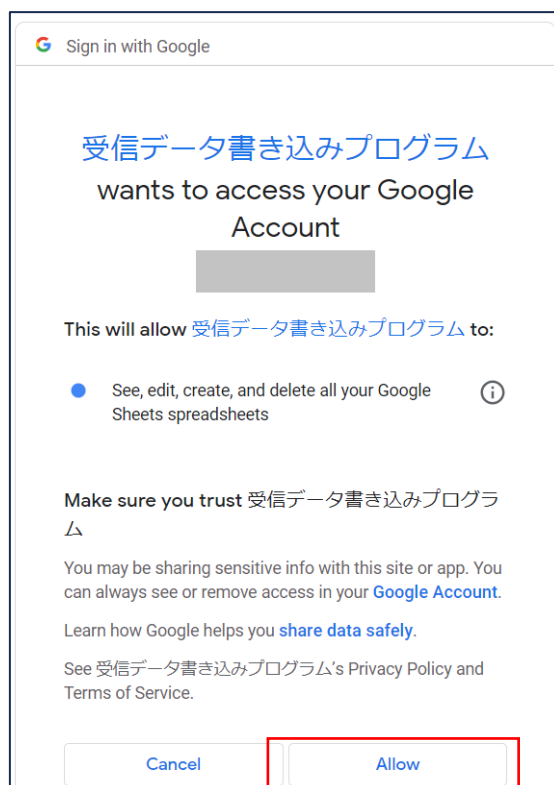
- 警告が現れます。この内容で進めて OK なら、左下の薄灰色の advanced をクリックしてください。



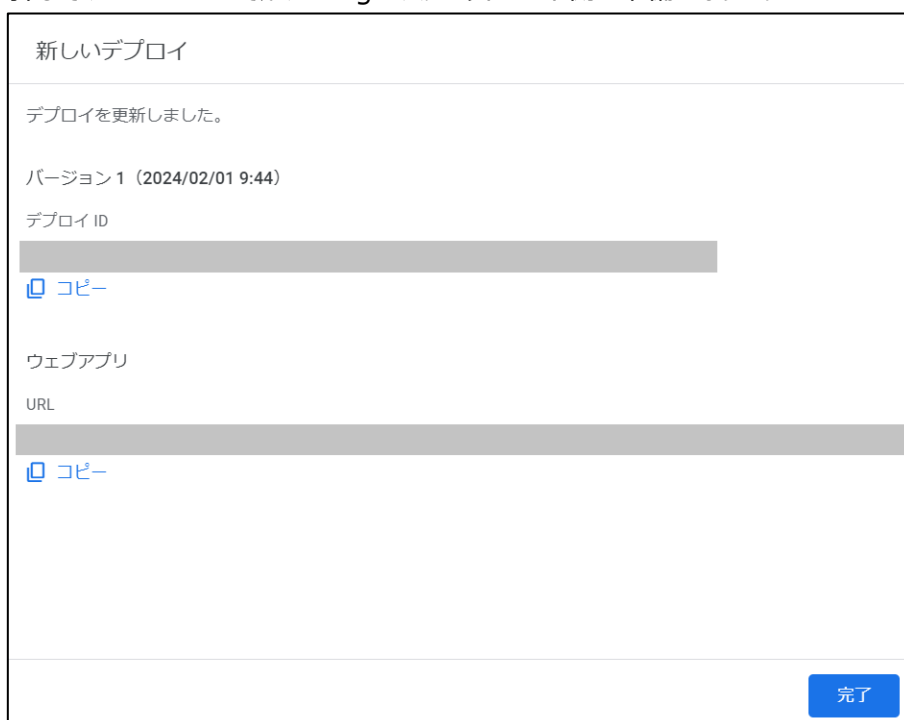
- 再度警告が現れます。この内容で進めて OK なら、左下の薄灰色の Go to 受信データ書き込みプログラム (unsafe) をクリックしてください。



- Google アカウントにアクセスするためのリクエストが表示されます。この内容で進めて OK なら、Allow ボタンをクリックしてください。



- デプロイが更新され、デプロイ ID とウェブアプリの URL が表示されます。ここでウェブアプリの URL をコピーし、メモ帳などに保存してください。この URL は、あとで Pico W 側のプログラムに書き込みます。メモ帳への保存が終わったら完了ボタンを押してください。ここまでが、Google スプレッドシート側の準備になります。

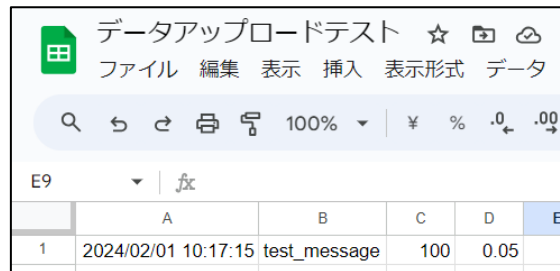


- 次に、Pico W 側の準備を行います。Thonny から動作させる場合、下記の構成でサンプルプログラムを Pico W にインストールしてください。

.Raspberry Pi Pico (Pico W は Thonny 上ではこのように表記されます)

└─google_sheets_send.py

- google_sheets_send.py の 13～14 行目に、Pico W を接続する Wi-Fi の SSID とパスワードを書いてください。また、15 行目に、先ほどコピーしたウェブアプリの URL を書いてください。プログラムを保存して、Thonny の実行ボタンを押すと、先ほど作成したスプレッドシートに以下のようにデータがアップロードされます。



The screenshot shows a Google Sheet interface. The title bar says 'データアップロードテスト'. The menu bar includes 'ファイル', '編集', '表示', '挿入', '表示形式', and 'データ'. The toolbar shows search, undo, redo, copy, paste, and zoom (100%). The active cell is E9. The spreadsheet data is as follows:

	A	B	C	D	E
1	2024/02/01 10:17:15	test_message	100	0.05	

- 独立電源で動作させる場合は、上記の構成でサンプルプログラムを Pico W にインストールし、google_sheets_send.py のファイル名を main.py に変えたうえで、Pico W に電源投入してください。

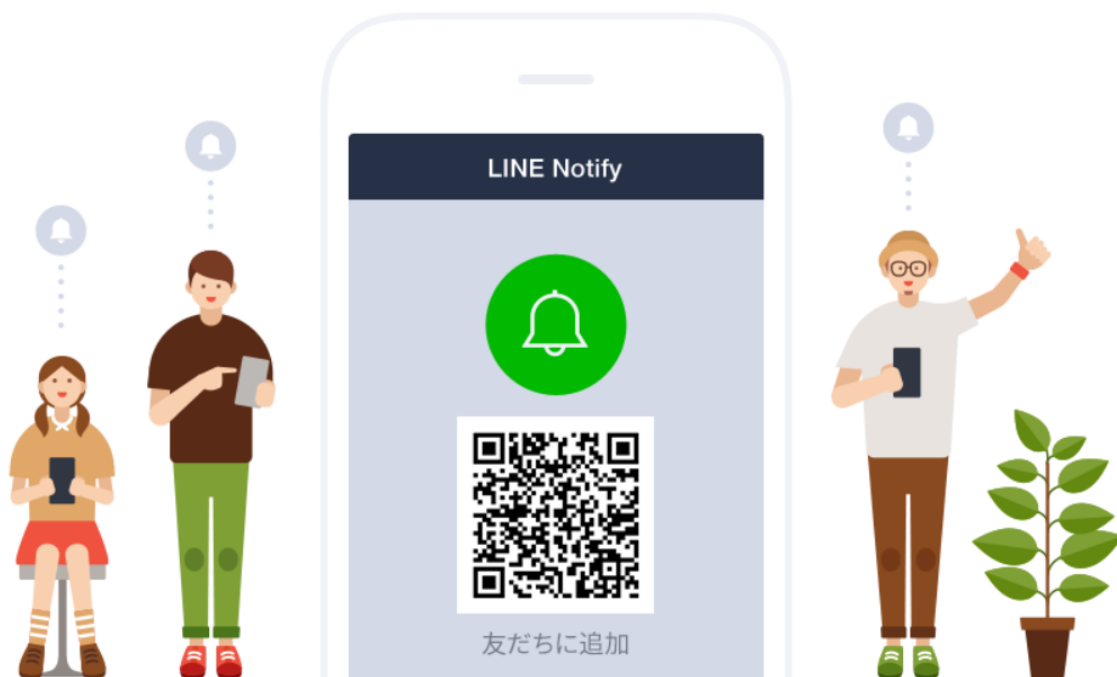
5.2. LINE Notify にデータを送信する

- このプログラムは、LINE のサービスである LINE Notify と連携し、そこにダミーデータをアップロードするプログラムです。ダミーデータをアップロードすると LINE の通知が来るので、何かのアラートなどに使えます。LINE Notify については、Web などに多数情報があるので説明は割愛します。また LINE Notify は、fireflake にとって外部のサービスになるため、それに関するご質問などをいただいても応じられないケースがあります。すみませんが、その点はご了承ください。
- LINE Notify に Pico W からデータを送るには、LINE Notify 側の準備と、Pico W 側の準備がいります。まず LINE Notify 側の準備から説明します。
- まず LINE Notify のサイトにいきます。少し下にスクロールすると QR コードが表示されているので、スマートフォンで読み取って、LINE Notify を友だちに追加してください。

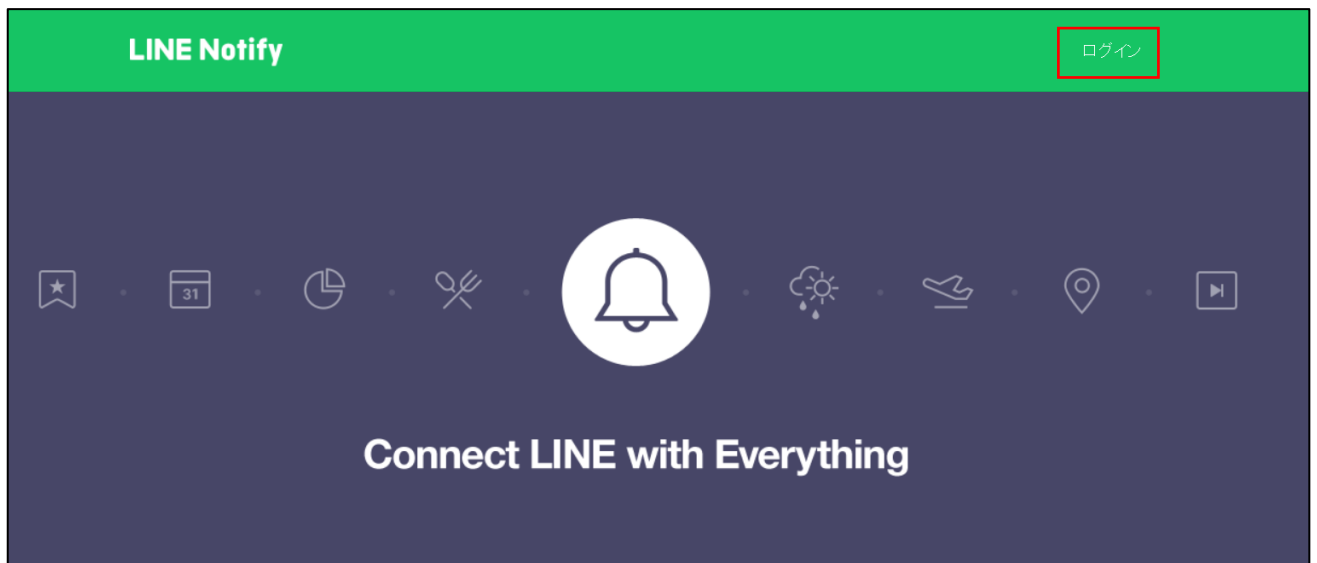
<https://notify-bot.line.me/ja/>

Webサービスからの通知をLINEで受信

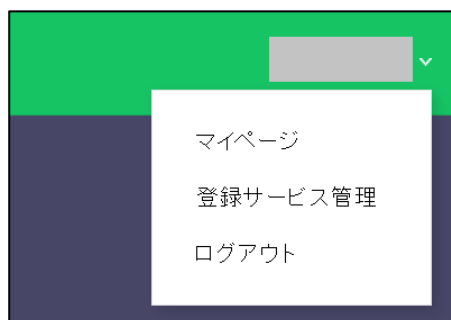
Webサービスと連携すると、LINEが提供する公式アカウント"LINE Notify"から通知が届きます。
複数のサービスと連携でき、グループでも通知を受信することが可能です。



- LINE Notify の画面右上のログインリンクからログインします。ログインの際、ログイン画面に表示されるパスコードをスマートフォンの LINE に入力する必要があるため、スマートフォンもお手元においておくとスムーズです。



- ログインしたら、画面右上のメニュー（さっきまでログインと表示されていたところ）からマイページを開きます。



- マイページにいったら、アクセストークンを発行します。

アクセストークンの発行(開発者向け)

パーソナルアクセストークンを利用することで、Webサービスの登録をせずに通知を設定することができます。

トークンを発行する

[LINE Notify API Document](#)

- トークン名は Pico W からのデータ受信とし、1:1 で LINE Notify から通知を受け取るを選択します。その後、発行するボタンを押してください。

トークン名を記入してください (通知の際に表示されます)

Pico Wからのデータ受信

通知を送信するトークルームを選択してください

🔍 Search by group name

1:1でLINE Notifyから通知を受け取る

※パーソナルアクセストークンを公開すると、連携したトークルーム名とプロフィールの名前を第三者が取得出来るようになります。

発行する

- トークンが発行されます。必ず、コピーしてメモ帳などに保存してください。保存を終えたら画面を閉じます。

発行したトークンはこちらです。

このページから移動すると、新しく発行されたトークンは二度と表示されません。トークンをコピーしてからページを移動して下さい。

コピー

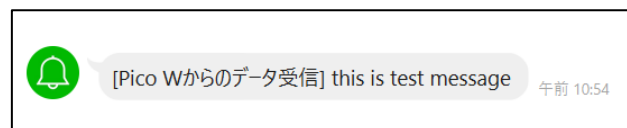
閉じる

- 次に、Pico W 側の準備を行います。Thonny から動作させる場合、下記の構成でサンプルプログラムを Pico W にインストールしてください。

.Raspberry Pi Pico (Pico W は Thonny 上ではこのように表記されます)

└─line_notify_send.py

- line_notify_send.py の 14～15 行目に、Pico W を接続する Wi-Fi の SSID とパスワードを書いてください。また、16 行目に、先ほどコピーしたトークンを書いてください。プログラムを保存して、Thonny の実行ボタンを押すと、LINE に通知が来ます。



- 独立電源で動作させる場合は、上記の構成でサンプルプログラムを Pico W にインストールし、line_notify_send.py のファイル名を main.py に変えたうえで、Pico W に電源投入してください。ただこれをして、通信結果を何らかのディスプレイに表示する仕組みなどがなければ、見た目上は何も起きません。

5.3. UECS (というプロトコル) でデータを送受信する

- このプログラムは、UECS というプロトコルでデータを送信するものです。UECS は、農業などで使われるハウス（温室）や植物工場などで、計測や制御のデータをやり取りするためのプロトコルです。UECS について詳しくはこちらを参照してください (<https://www.uecs.jp/>)。
- Thonny から動作させる場合、下記の構成でサンプルプログラムを Pico W にインストールし、uecs_receive.py または uecs_send.py を開いている画面で、Thonny の実行ボタンを押してください。

.Raspberry Pi Pico (Pico W は Thonny 上ではこのように表記されます)

└─uecs_receive.py

.Raspberry Pi Pico (Pico W は Thonny 上ではこのように表記されます)

└─uecs_send.py

- 独立電源で動作させる場合は、上記の構成でサンプルプログラムを Pico W にインストールし、uecs_receive.py または uecs_send.py のファイル名を main.py に変えたうえで、Pico W に電源投入してください。ただこれをして、出力した事を何らかのディスプレイに表示する仕組みなどがなければ、見た目上は何も起きません。

6. ライセンスについて

サンプルプログラムのうち、fireflake 制作のプログラムは MIT ライセンスで配布されます。それ以外のプログラムは、それぞれのプログラムが配布されているライセンスに従います。fireflake 制作のプログラムは下記になります。

google_sheets_send.py

line_notify_send.py

uecs_receive.py

uecs_send.py

7. おわりに

本資料が、皆さんがもの作りを楽しめる一助になれば幸いです。Happy Hacking!!