

UiFlow2によるAIスタックチャン

1. はじめに

「スタッツチャン」は、ししかわさんが開発・公開している手乗りサイズのコミュニケーションロボットです。2021年にオープンソースロボットとして公開され、回路図、3Dモデル、ファームウェアなどが公開されており、誰でも自由に作成、機能拡張を行うことができます。

オリジナルのスタッツチャンのファームウェアは、「Moddable SDK」というJavaScriptを使ったソフトウェアプラットフォームを使用しており、独自の機能拡張を行うには、Web開発の知識（と機能によっては組込み開発の知識）を必要としています。

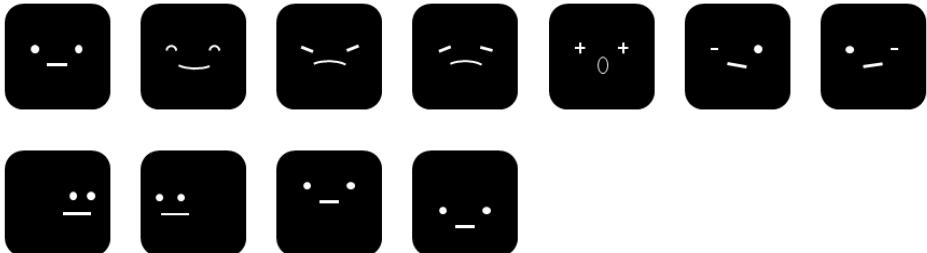
多くの方にスタッツチャンを楽しんで頂くためには、少しハードルが高いと考え、M5Stackの公式開発環境である「UiFlow2ファームウェア」を使った基本機能ライブラリ「UiFlow スタックチャン」を作成しました。

2. UiFlow スタックチャンについて

UiFlow2は、スタッツチャンの頭脳であるM5Stackが公式にサポートしている開発環境の1つです。Webブラウザによるビジュアルプログラミングをサポートしており、ノーコードでプログラム開発を行うことができます。また、UiFlow2は、MicroPythonという組込み向けのPythonスクリプトをベースにしており、Pythonプログラムを使った独自の機能拡張も容易に行うことができます。
「UiFlow スタックチャン」では、オリジナルのスタッツチャンでサポートしている機能の他に、AI機能として、Google Cloudを利用した音声対話をすることができます。
また、Webサーバー機能も実装しており、様々な操作を行なうことができます。

2.1 機能

UiFlow スタックチャンの主な機能は、以下の通りです。

- 様々な表情
 - 瞬き、笑い、怒り、悲しみ等の様々な表情を画面に出力できます。また、Pythonプログラムで独自の表情を作成可能です。
- モータ制御
 - SG90 または Dynamixel のモータを制御し、上下左右に動作します。
- 音声認識
 - Google Cloud Speech-to-textを使って、音声認識を行うことができます。音声の入力は、M5Stack CoreS3のマイクを使って入力することができます。
- 音声合成
 - Google Cloud Text-to-speechを使った音声合成により発話することができます。設定により、発話音をカスタマイズ可能です。
- フェイストラッキング

- M5Stack CoreS3に内蔵されたカメラを使った顔検出を使ったフェイストラッキングを行うことができます。
- Webサーバ
 - 独自実装のWebサーバ機能を利用することができ、REST APIを使ってスタックチャンを操作することができます。

2.2 ビジュアル・プログラミング

UiFlow スタックチャンを使用すると、UiFlow2のビジュアル・プログラミングが可能になります。カスタムブロックとして、見えないボタン(Button)、メインの機能(StackChan)を使うことができます。Google Geminiを使った AIスタックチャンのプログラムは、下のように実装されています。



このプロジェクトは、UiFlow スタックチャンのアプリケーションプログラムです。UiFlow2のProject Zoneを検索画面から "**Stack-chan-RT**"と検索して読み込むことができます。このプロジェクトでは、プログラムのためのブロックはありますが、実行するためのUiFlow スタックチャンライブラリをインストールする必要があります。

3. インストール

3.1 フームウェアのインストール

UiFlow スタックチャン のプログラムを使用するためには、UiFlow2のファームウェアをM5Stackにインストールする必要があります。

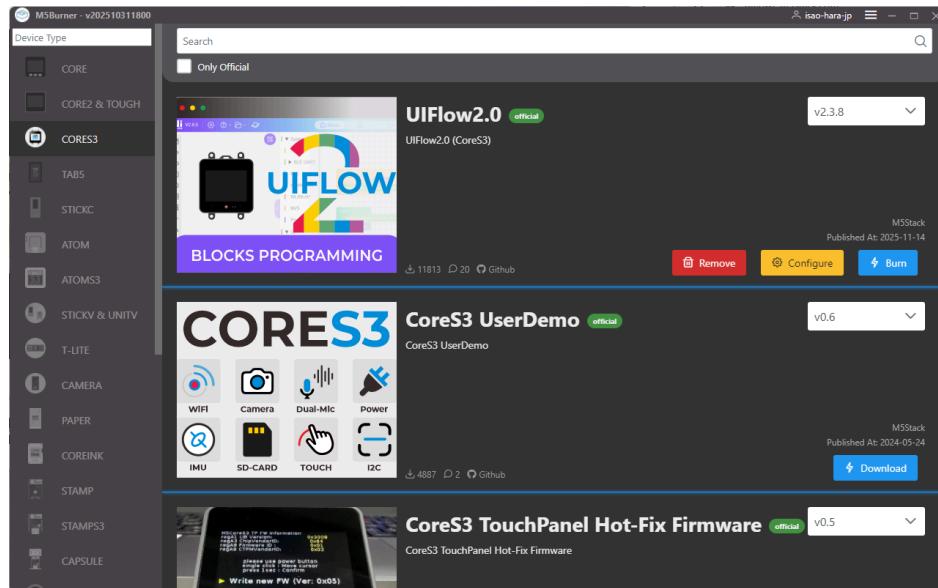
ファームウェアのインストールの詳細は、「UiFlow2 Web IDE クイックスタート (https://docs.m5stack.com/ja/uiflow2/uiflow_web)」を参照してください。

UiFlow2ファームウェアのインストールには、「M5Buner」というアプリケーションが必要です。このアプリケーションは、Windows、MacOS、Linuxで動作し、M5StackとUSB接続を介して、ファームウェアをインストールすることができます。

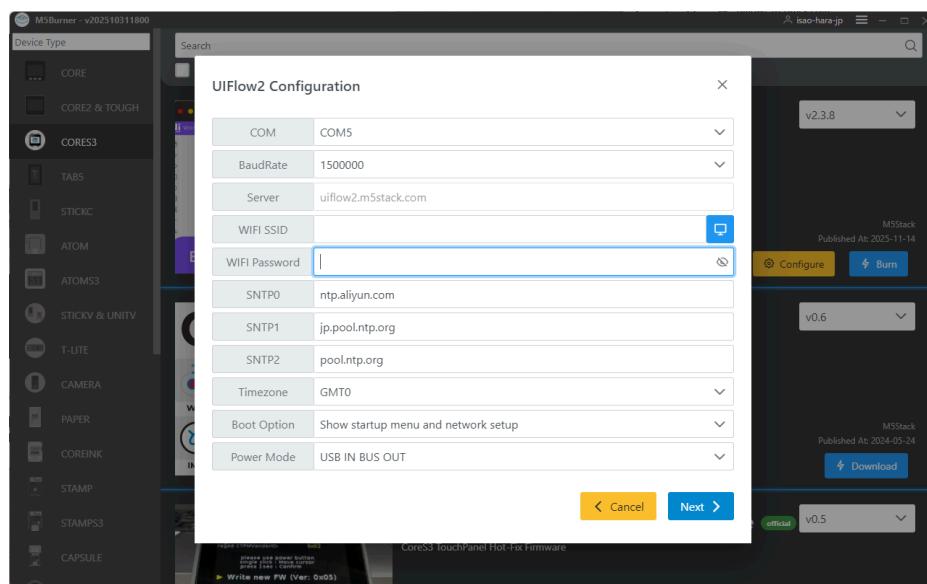
M5Stack CoreS3のファームウェアのインストール方法は、こちら (<https://docs.m5stack.com/ja/uiflow2/m5cores3/program>) を参照して下さい。

なお、M5Burnerを使用する際には、M5 Stack Community (<https://community.m5stack.com/register>)へのログインが必要になりますので、事前にアカウント登録をすましておいてください。

M5Bunerを起動し、CoreS3を選択すると下記のようにファームウェアの一覧が表示されます。「UiFlow2.0」の「Download」を押下した後ファームウェアがダウンロードされ下図のようになりますので、[Configure]を押下します。

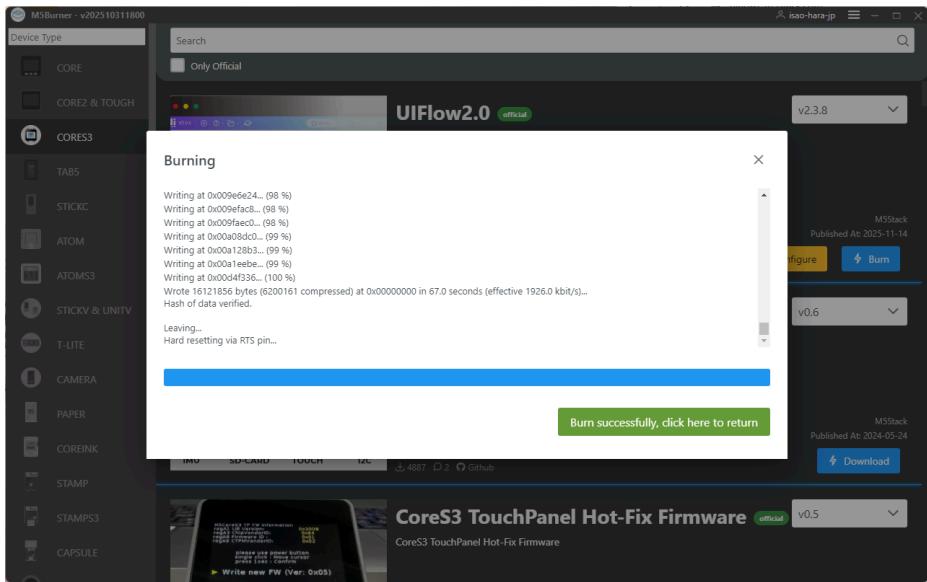


UiFlow2のカスタマイズ設定画面が表示されますので、WifiのSSIDとPasswordを入力してください。



Wifiの設定を入力後、【Next】を押下しファームウェアのインストールを行います。

インストール終了後、下のような画面になりますので、M5Stackのリセットボタンで再起動します。
(UiFlow2のファームウェアのインストール時には、安定したネットワーク環境が必要ですので注意して下さい。)



Uiflow2のファームウェアのインストール後に、再起動すると下図のような画面が表示されることを確認してください。
もし異なる画面が表示されるようでしたら、ファームウェアのインストールを最初からやり直してください。



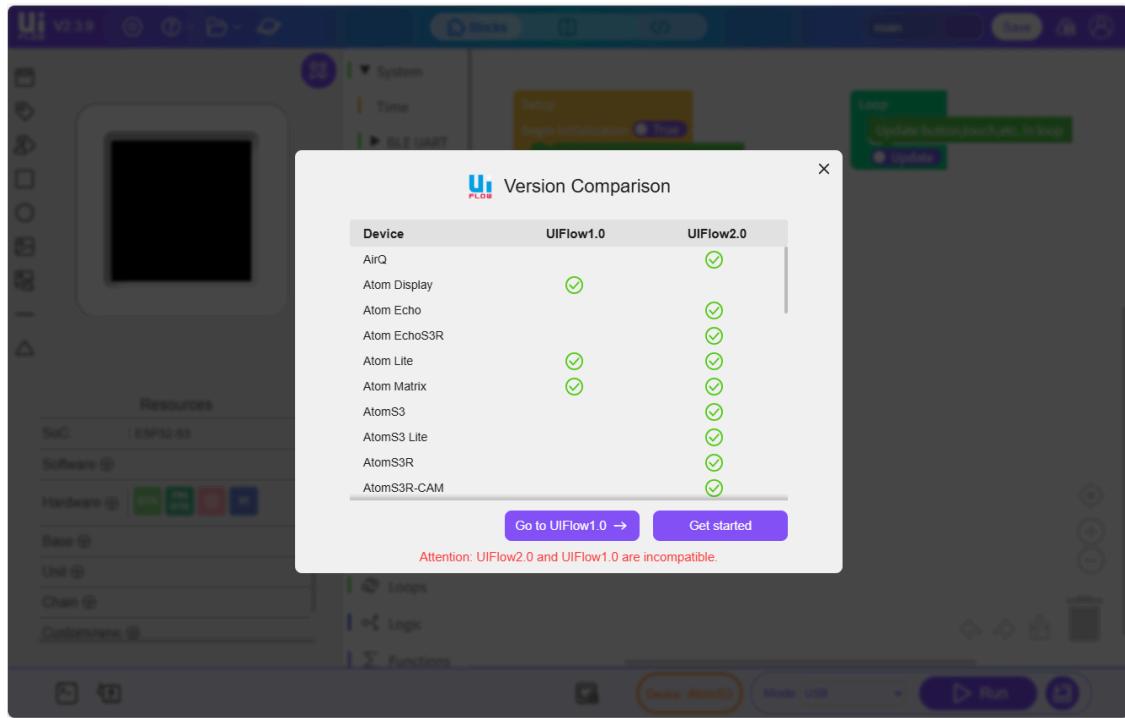
以上でファームウェアのインストールは完了です。

3.2 UIFlow スタックチャンライブラリのインストール

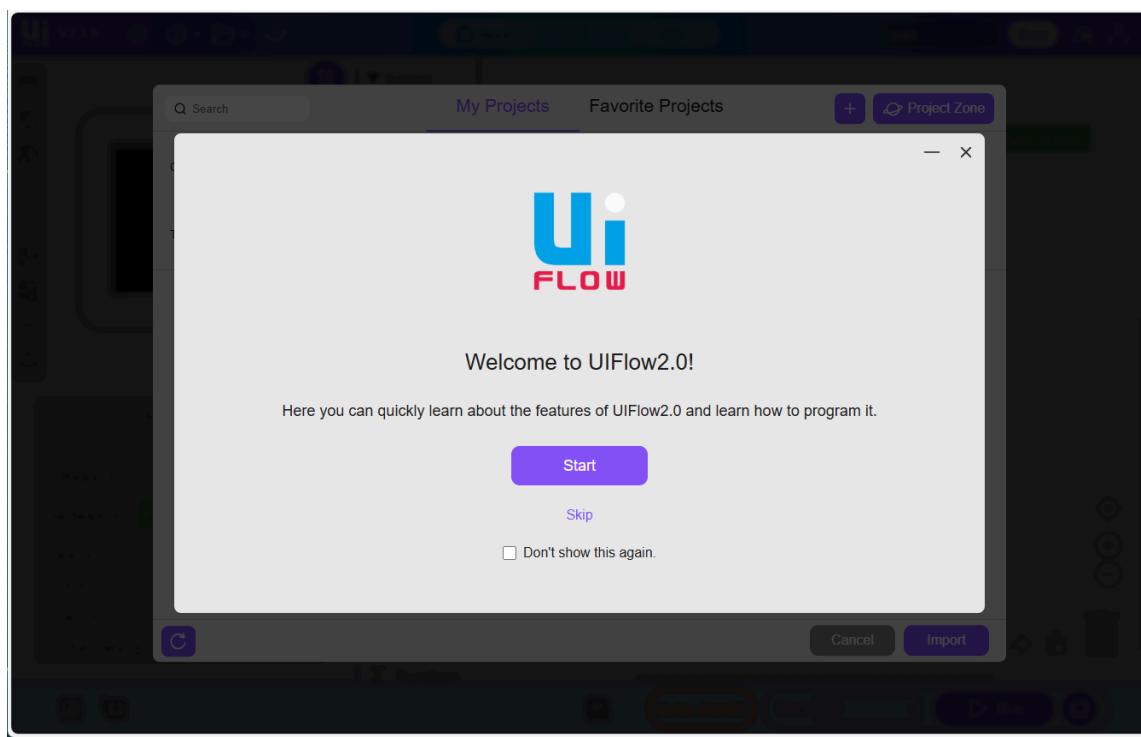
UIFlow スタックチャンライブラリは、Github(<https://github.com/haraisao/stackchan-uiflow2>)で公開しています。このリポジトリの内容をダウンロードし、libsの下にあるプログラムをアップロードしても良いのですが、ここでは、UIFlow2の「Project Zone」に公開しているインストーラプログラムを使用します。(前述のアプリケーションプログラムとは異なりますので、注意してください)

WebブラウザでUIFlow 2.0([UIFlow2](#))にアクセスします。

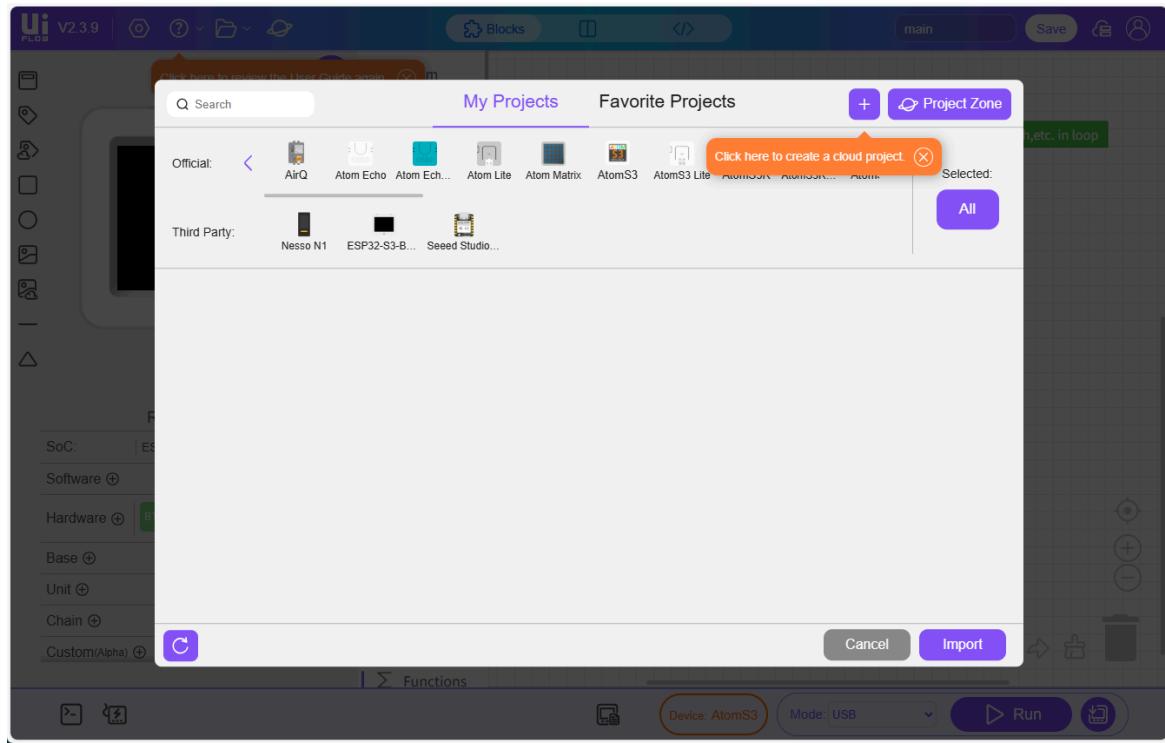
UIFlow2へログインせずにアクセスすると下図のような画面がでますので【Get started】ボタンを押下して次に進みます。



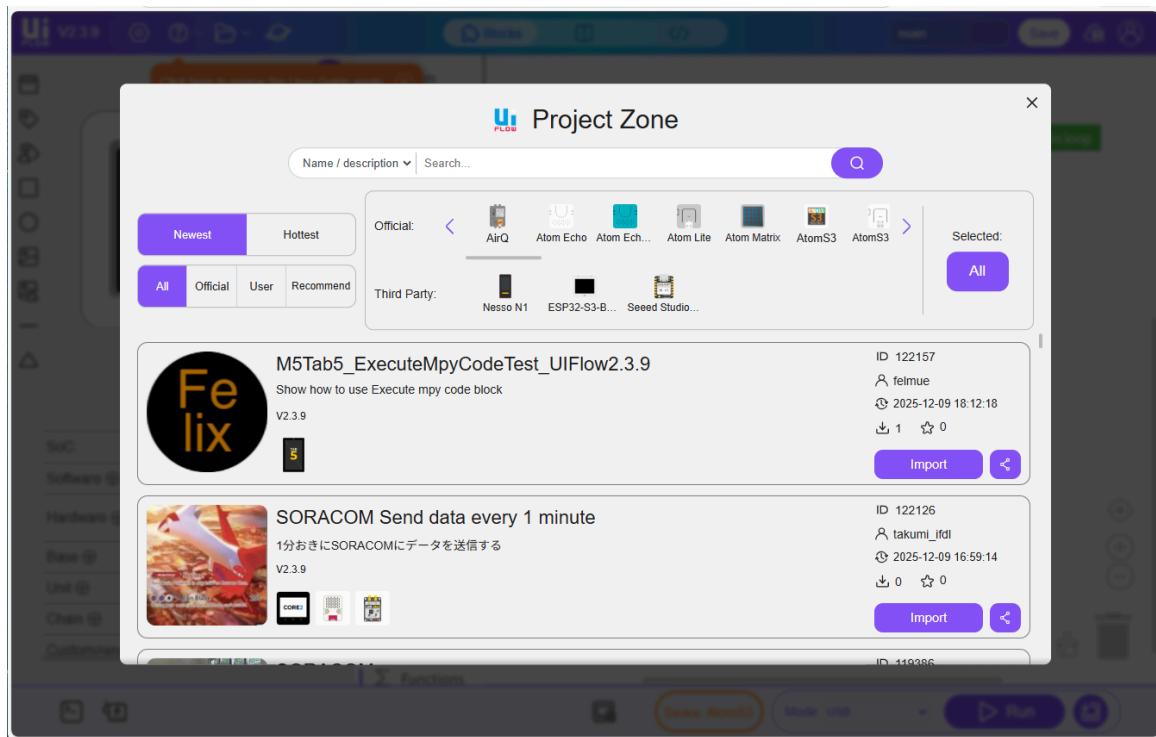
次に、下図のようにWelcomeダイアログが表示されますので、ボタンの下にある「Skip」をクリックします。



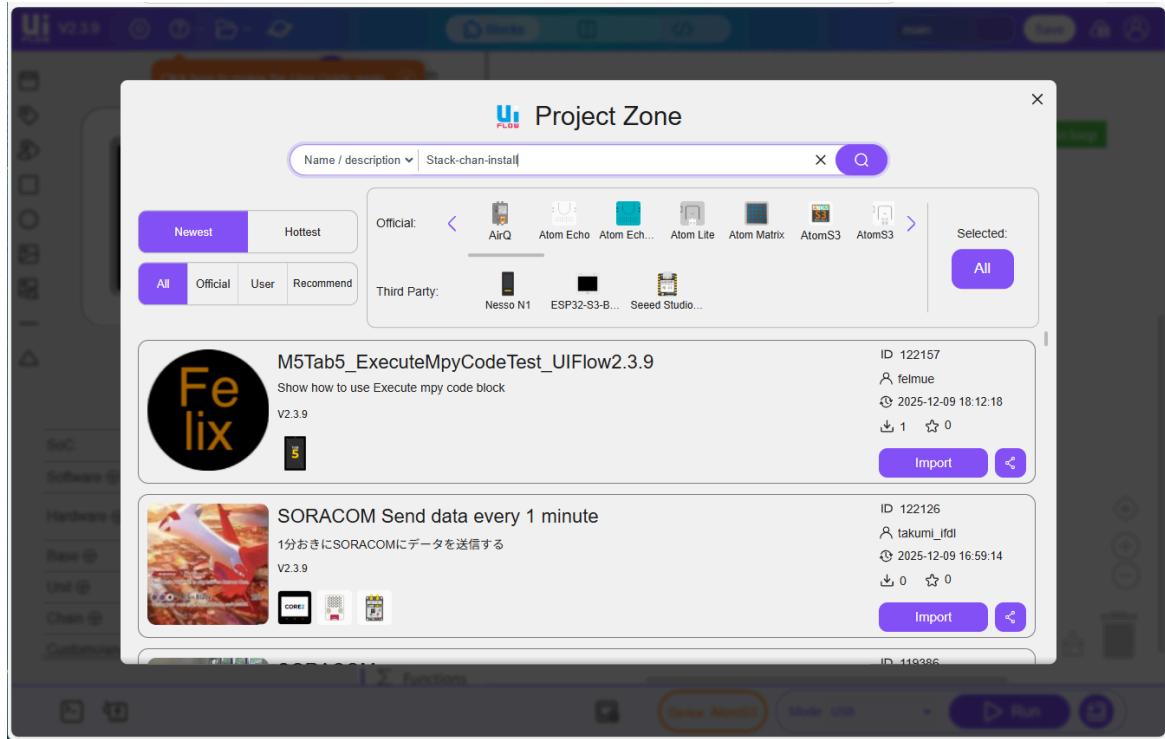
次は、下図のようにハードウェアの選択画面に移行しますので、UiFlow スタックチャンライブラリのインストーラプロジェクトを開きます。



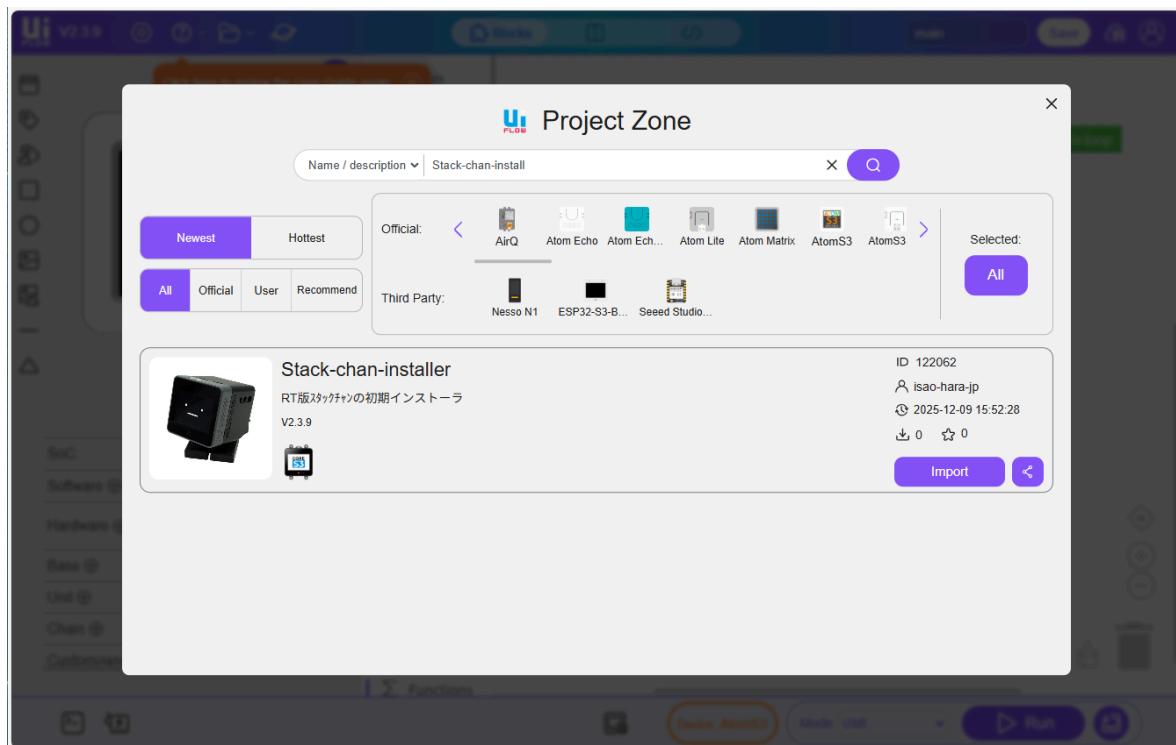
上記の画面の右上にある【Project Zone】ボタンを押下し、プロジェクト選択ダイアログに進みます。



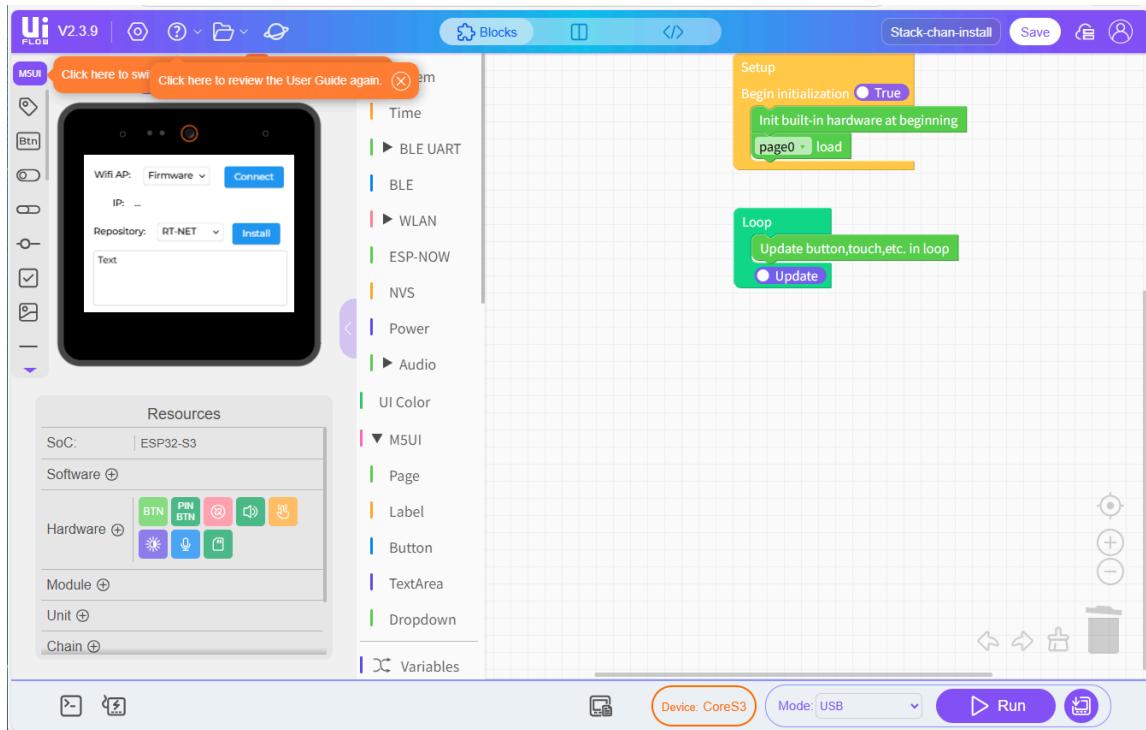
インストーラプロジェクトは、”Stack-chan-installer”という名称で公開していますので、上部の検索窓に”Stack-chan-installer”と入力し、を押下してください。



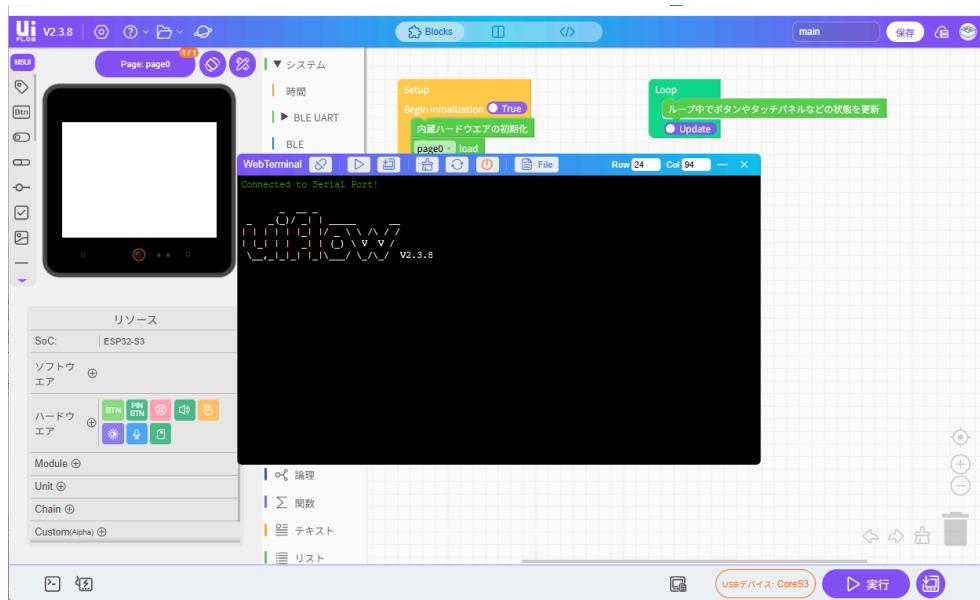
検索すると下図のようにインストーラのプロジェクトが表示されますので【Import】ボタンを押下し、プロジェクトを読み込みます。



プロジェクトの読み込みが完了すると、下図のような画面に移行します。



パソコンとスタックチャンのM5StackのUSBケーブル接続を確認後、右下部の ボタンを押下すると、接続ポートの選択ダイアログが表示されますので、接続ポートを選択します。M5Stackとの接続が完了すると下図のようにWeb Terminalが表示されます。



インストーラを実行するために、この状態で再度 を押下し、インストーラを起動します。インストーラが正常に起動すれば、M5Stackの画面は下図のようになります。



インストーラの起動後は、Wifiの接続を行います。UiFlow2のファームウェインストール時に設定した無線LANアクセスポイントへの接続を確認するために「[Connect](#)」を押下してください。正常に接続できれば、現在のIPアドレスが表示されます。

次に、UiFlow スタックチャン ライブラリのダウンロードを実行します。ダウンロードリポジトリとして、「**1.RT-NET**」と「**2.Org**」の2つがあります。どちらのリポジトリを選んでも同じプログラムがダウンロードされますが、初期設定ファイルが少し異なります。(起動後に修正できますのでどちらのリポジトリを選択して頂いても構いません)

RT版のスタックチャンをご使用の場合には、「**1.RT-NET**」を選択し、「[Install](#)」をタップしてください。Githubのサイトに接続し、必要なライブラリ、プログラム、設定ファイル、WebページがM5Stackにダウンロードされます。

ダウンロード完了後に、M5Stackを再起動させてください。

3.3 設定ファイルの作成

UiFlow AIスタックチャンを動作させるには、`apikey.txt`, `wlan.json`, `stackchan.json`という3つの設定ファイルが必要になります。

これらの設定ファイルは、以下のようになっています。

- **stackchan.json**: UiFlow スタックチャンのメインの設定ファイル、モータや音声認識、音声合成、LLMサーバーなど設定を記載する。
- **wlan.json**: UiFlow AIスタックチャンは無線LANが必須になります。あらかじめHome, Work, Mobileの3カ所のアクセスポイント設定を記録しておいてください。動作環境によって自動接続を行います。
- **apikey.txt**: このファイルは、GoogleやOpenAIなどのクラウドサービスにアクセスするための設定ファイルです。

各設定ファイルは、/flash にインストールされています。設定ファイルの修正は、UiFlow2のWeb Terminal からダウンロードすることができますので、ご使用の環境に合わせて、修正後、再度アップロードしてください。

初期設定の各ファイルを下記に示します。

stackchan.json:

ご使用のモーターにしたがって、motorの項目を修正してください。SG90のモーターを使っている場合には、ご使用のポートのGPI番号に合わせてください。初期設定では、CoreS3のPortAに横方向のモーター、PortBに縦方向のモーターに接続にしています。

```
{  
    ## 使用しているモータの種別 ( Dynamixel or SG90 )  
    "motor": "Dynamixel",  
    "SG90": {"pan": 2, "tilt": 9, "offset": 56},  
    ## 音声合成のエンジン (google or voicevox)  
    "tts": "google",  
    "google": {  
        "lang": "ja-JP", "speakingRate": "1.2", "ssmlGender": "FEMALE",  
        "voiceName": "ja-JP-Standard-A", "pitch": "5.0",  
        "volumeGain": "0",  
        "sampleRate": "8000", "effectsProfileId": "",  
        "sampleRateHertz": 8000},  
    "voicevox": {"host": "192.168.0.100", "voice_id": 1},  
    ## 音声合成のエンジン (google or vosk)  
    "asr": "google",  
    "vosk": {"host": "192.168.0.100"},  
    ## Webサーバーのポート番号  
    "web_server": 80,  
    ## Cameraの初期化  
    "camera_setup": true,  
    ## カメラ画像を180度回転  
    "vflip": false,  
    ## 音声対話エンジン (gemini or openai)  
    "dialog": "gemini",  
    "gemini": {"model": "/gemini-2.5-flash:generateContent",  
              "lang": "ja_JP"},  
    "openai": {"model": "gpt-5"},  
    ## 生成AIのプロンプト  
    "prompt": "あなたは、小さなスーパーロボット「スタックチャン」です。現在、東京  
にいます。対話の応答は、東京にいることを前提に、30字以内で答えてください."  
}
```

wlan.json:

このファイルは、Wifiアクセスポイントの設定になります。Home、Work、Mobileの3つを登録することができます。ご使用の環境に合わせて設定してください。

なお、現在の実装では、Firmwareのインストール時に設定した SSIDとパスワードは、最初に接続を試みますので、同じ情報を記載しなくてもWifiの接続は可能です。

```
{  
    "Home": {"essid": "", "passwd": ""},  
    "Work": {"essid": "", "passwd": ""},  
    "Mobile": {"essid": "", "passwd": ""}  
}
```

apikey.txt:

このファイルには、音声認識、音声合成、LLMによる生成AIを使用する場合のAPI Keyを記載します。現在サポートしているCloudサービスは、Google Speech API、Gemini、ChatGPTです。

GOOGLE_SPEECH_KEY=Replace your Google speech api key

GEMINI_KEY=Replace your Gemini api key

OPENAI_KEY=Replace your OpenAI api key

UiFlow AIスタックチャンでは、Webブラウザで内部のファイルを編集する機能を実装しています。各設定ファイルは、このファイル編集機能を使って修正することができます。UiFlow AIスタックチャンを起動後、WebServerをOnにし、http://<UiFlow AIスタックチャンのIPアドレス>/edit_file.htmlにアクセスすると、簡易なファイル編集機能を使うことができます。

3.4 htmlファイルの作成

UiFlow AIスタックチャンは、Webサーバーを内蔵しています。HTMLファイルを作成することで、REST API経由で内部機能の呼び出しを行うことができます。

簡単なサンプルHTMLファイルは、/flash/html フォルダにあります。

デフォルトのHTMLファイルは、以下のようになります。

このHTMLファイルでは、首振りや顔の表情の変更、カメラからの画像の取得、音声合成、音声認識の開始などを実行することができます。

音声認識を開始すると、認識した音声をLLMに送信し、応答を音声合成で発話します。

そのほかに、stackchan.jsonの内容を変更する「パラメータの修正」とM5Stack内部のファイルを編集する「ファイルの修正」というページがあります。

アールティ版 スタックチャンの操作

首振り操作

水平方向:0
垂直方向:0



Face

顔の表情: 通常 (瞬きあり) ▾

リンク

- [音声認識 & 音声合成](#)
- [ファイルの修正](#)
- [パラメータの修正](#)

音声合成

発話する文字列: TTS

音声認識

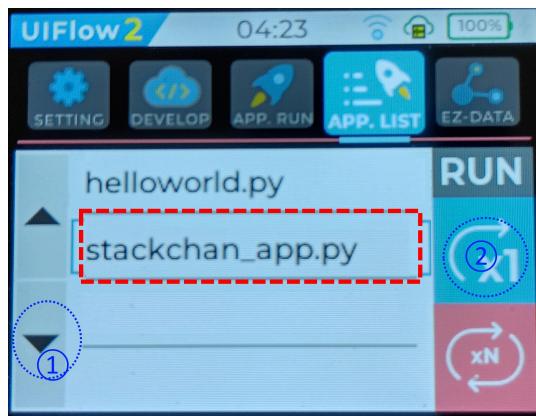
最大録音時間 (秒) : 5
発話区間推定 閾値: 41
発話区間推定 連続検出回数: 1

[Back to Top](#)

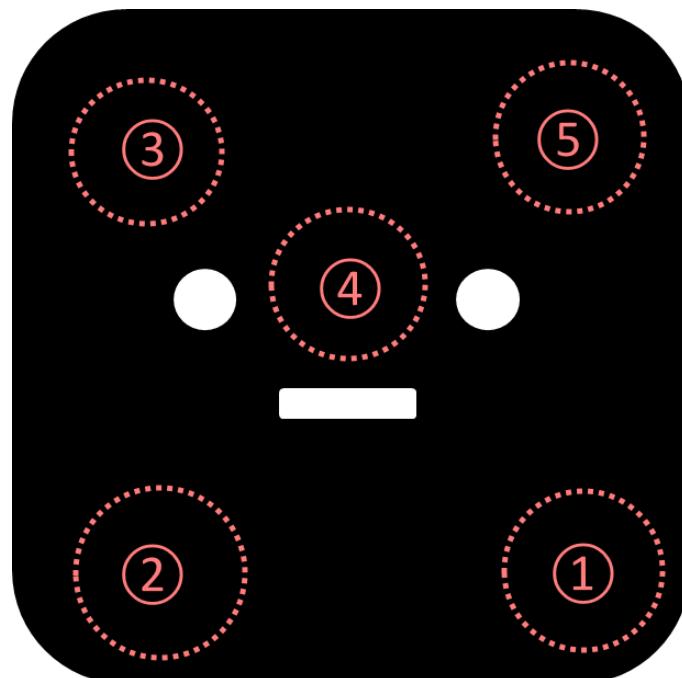
以上でインストール完了になります。

4. UiFlow AIスタックチャンの実行

UiFlow AIスタックチャンの起動プログラムは、APP.LISTにある stackchan_app.py になります。
M5Stackを起動後、下図のように起動プログラムを選択し、実行してください。



このプログラムを起動すると、画面には以下のような顔が表示されます。



図の赤点線の○で囲まれた部分には、下記の機能のタッチボタンを割り当てています。

- ① 現在のバッテリーの充電状況を表示、顔の表情をランダムに変更
- ② 無線LANアクセスポイントに接続し、IPアドレスと表示、WebサーバのOn/Off
- ③ ランダム動作のOn/Off(900秒で停止します)
- ④ 対話の開始
- ⑤ 顔トラкиングのOn/Off

Webサーバーも起動していますので、IPアドレスにWebブラウザでアクセスすれば、前述のサンプルページが表示されます。

【注意】

現在の実装では、顔トラッキングを長時間有効にしているとハングアップする場合があります。ハングアップした場合には、リセットボタンでリセットしてください。

5. UIFlow AIスタックチャン内部のファイル編集

UIFlow AIスタックチャンには、M5Stack内部のファイルをオンライン編集する機能を実装しています。

M5Stackの起動後、無線LANアクセスポイントに接続ボタン(前章の②)を押下して、Webサーバーをオンにします。(②部分を押下すると、WebサーバーのOn/Offが切り替わります)

Webブラウザで、M5StackのIPアドレスに接続後、トップページにある「ファイルの修正」というリンクをクリックすると、下のような画面が表示されます。

編集したいファイルを左側のセレクターで選択し、ファイルを修正・保存を行ってください。

設定ファイルやライブラリプログラムを更新した場合には、再起動後にその修正が反映されます。(htmlファイルは、再読み込みで修正が反映されます)

File manager

DIR: html

Filename: html/index.html

Dirs:

images
js

Files:

asr_tts.html
edit_file.html
favicon.ico
index.html
params.html

```
1  <!DOCTYPE html>
2  <html lang="ja">
3      <head>
4          <title>Hello Stack-chan</title>
5          <meta name="viewport" content="width=device-width, initial-
6          <meta charset="utf-8">
7
8          <script type="text/javascript" src="js/joy.min.js"></script>
9
10         <script type="text/javascript">
11             async function get_camera_image(){
12                 const res = await fetch('/get_camera_image', {
13                     method: "POST", body: ""
14                 });
15                 const { width, height, data } = await res.json();
16                 const canvas = document.getElementById('canvas');
17                 canvas.width = width;
18                 canvas.height = height;
19                 const ctx = canvas.getContext('2d');
20                 const imageData = ctx.createImageData(width, height);
21
22                 const binary = atob(data); // Base64 → binary
23                 const buffer = new Uint8Array(binary.length);
24                 for (let i = 0; i < binary.length; i++) {
25                     buffer[i] = binary.charCodeAt(i);
26                 }
27
28                 for (let i = 0, j = 0; i < buffer.length; i += 2, j++) {
29                     const value = (buffer[i] << 8) | buffer[i + 1];
30
31                     // RGB565 → 8bit RGB変換
32                     const r = ((value >> 11) & 0x1F) << 3;
33                     const g = ((value >> 5) & 0x3F) << 2;
```

選択されていません
[Back to Top](#)

