

第1章

ラズベリーパイ財団公式資料に書かれた一番シンプルな方法

# ラズベリー・パイ4でPicoのプログラムを作る

編集部

ラズベリー・パイ Pico (以下、Pico と呼ぶ) の開発に使えるプログラミング言語としてラズベリーパイ財団 (以下、公式と呼ぶ) では、C/C++ と MicroPython を用意しています。公式で用意しているこれらの言語以外にも Arduino フレームワークや Rust で開発できるようにする動きもあるようです。

開発に使う環境としては基本的に Linux となるため、Windows PC を使って開発したい場合、WSL や VirtualBox といった仮想環境を利用します。もしくは Windows からラズベリー・パイへ接続し、ラズベリー・パイから Pico に接続する方法を利用します。

本誌 2021 年 8 月号でも、これらの環境構築方法を紹介しました。これらの方法は、PC から直接 Pico の開発ができるメリットがありますが、初心者には少し難しい部分もあるかもしれません。

## ● 最も初心者向けの方法でやってみる

ここでは公式の英語ドキュメント<sup>(1)</sup>の解説を元に、開発環境の構築方法を紹介します (分かりやすさを重視して一部変更や省略している)。

この方法では PC が不要な代わりに、ラズベリー・パイが必要ですが、Windows 上に Linux 環境を構築したり、ネットワーク越しに Linux へ接続したりする必要がなく、初心者にも最も試しやすいと思います (写真1)。

ラズベリー・パイのセットアップ方法も第2章で紹介します。初めてラズベリー・パイを使う方は、まず第2章を参照してラズベリー・パイの準備をしてください。

### Picoのプログラムを作る1…C/C++

今回はラズベリー・パイ4を利用しましたが、3B や 3B+ でも方法は変わりません。

以下の作業はラズベリー・パイにキーボード、マウス、ディスプレイを接続して行います。

まずセットアップ用のシェル・スクリプトをダウンロードするため、ラズベリー・パイがインターネットにつながった状態で、次のコマンドをコマンドラインから入力します。

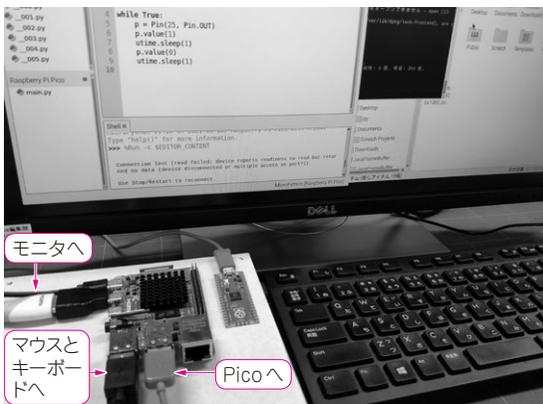


写真1 なんだかんだでラズベリー・パイ4から直接Picoを開発するのがシンプル

```
$ wget https://raw.githubusercontent.com/raspberrypi/pico-setup/master/pico_setup.sh
```

取得したシェル・スクリプトに実行権限を付与し、実行します。

```
$ chmod +x pico_setup.sh
$ ./pico_setup.sh
```

このシェル・スクリプトによって自動で行われる作業の抜粋を次に示します。

- カレント・ディレクトリにpicoディレクトリを作る
- pico-sdk など関連ファイルをダウンロード
- PICO\_SDK\_PATH など環境変数の定義を ~/.bashrc に追記
- サンプル・プログラムをビルド
- 書き込みに使うツール (picotool) をダウンロード
- デバッグ用ツール (picoprobe) をダウンロード
- コンパイル済みの OpenOCD (デバッグ用ソフトウェア) をダウンロード
- 統合開発環境として使う VS Code をインストール
- ラズベリー・パイの UART を設定