

## 第4章

# Arduino IDEで開発したスケッチをステップ実行! ブレイク・ポイント設定や変数などの情報を見える化

## 汎用デバッガ×VS Code で構築するデバッグ環境

丸石 康

### ● おなじみ Arduino IDEでもPicoの開発ができる

Arduinoは、ラズベリー・パイPico(以降、Pico)に搭載されているRP2040を公式にサポートすることを発表しました。Arduinoが提供している統合開発環境Arduino IDEは、ラズベリーパイ財団から提供されている公式SDK(Software Development Kit)に比べると、環境構築の手順がシンプルなことや、豊富なArduino用ライブラリの資産が利用できることなどのメリットがあります。

### ● 汎用デバッガとVS CodeがあればデバッグもOK!

Arduino IDEは、スケッチ(C++ベースのArduino専用プログラミング言語)のコンパイルと、シリアル・モニタを使ったデバッグができますが、ステップ実行や任意の変数の値確認など、いわゆるデバッガ機能はありません。2021年3月にベータ版がリリースされたArduino IDE 2.0には将来的にソースコード・デバッグ機能が追加されるようですが、本原稿執筆時点(2021年5月)では対応が十分ではありません。

そこで本章では、現時点での正式リリース版であるArduino IDE 1.8.15でコンパイルした実行ファイルをデバッグする方法を紹介します。本章で紹介するデバッグ方法では、写真1に示す汎用デバッガJ-Link(SEGGER)とVS Codeを使います。

### ステップ1: Arduino IDEにPicoの開発環境を追加する

#### ● Picoの開発にArduino IDEを使うメリット

##### ▶ (1) 豊富なArduinoライブラリが利用できる

Pico用のC++開発環境にArduino IDEを使えば、豊富なArduino用ライブラリの資産をPico用プログラムの開発に利用できます。

ライブラリ資産の中には、Picoの性能を生かせないものやPicoで使用できないものもありますが、コミュニティ・ベースの開発が進むにつれて問題は解消されていくと考えられます。

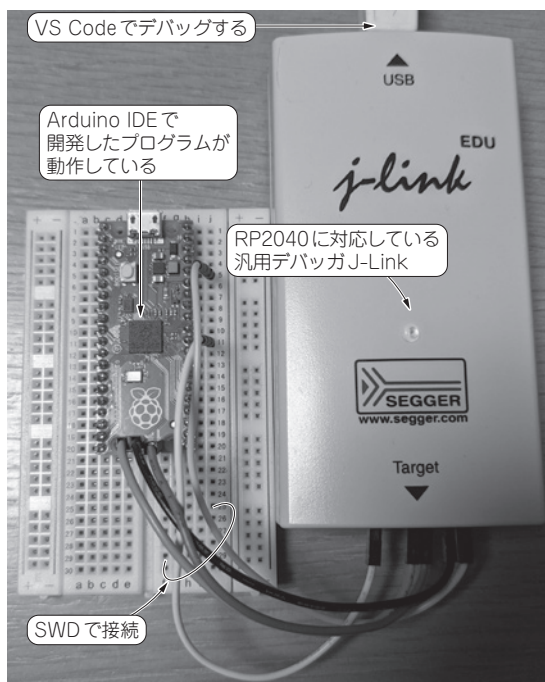


写真1 Arduino IDEで開発したPico用プログラムを汎用デバッガとVS Codeでデバッグしてみる  
Picoと汎用デバッガJ-Link EDUを接続した様子

##### ▶ (2) 少ない学習コストで利用できる

Arduino IDEと公式SDKは、どちらもコンパイラにGCC(GNU Compiler Collection)を使っていますが、サポートしているAPI(Application Programming Interface)が異なります。Arduino IDEは、Arduino APIをサポートしています。公式SDKには独自のAPIが用意されています。

Arduino APIは、通常だとマイコンによって異なるドライバなどのインターフェースの差分を吸収してくれるので、新規のマイコンやボードであっても少ない学習コストで開発できるようになります。