

# ESP32マイコンの MicroPython 100倍高速化

井田 健太

## 実験に必要なもの

### ● ハードウェア

ここまではMicroPythonのunixポートを使ってPC上で実行していました。次はESP32上で動かしてみます。

ESP32での実行には、ESP32を使ったモジュールESP32-WROOM-32の開発キットであるESP32-DevKitCを使います(第2部 第1章 写真1)。

### ● Myシンプル計算モジュールをESP32用に組み込むための修正

unixポートと同様にesp32ポートのmpconfigport.hを変更して、fastmathモジュールを組み込みます(リスト1)。

### ● ESP32用開発環境の準備&コンパイル

まずESP32用のバイナリをコンパイルするために、ESP32用のコンパイラとESP-IDFを用意します。このあたりの手順はバージョンによって異なりますので、実際には公式のドキュメントのSetup Toolchainの項を参照します。

以下に執筆時点での手順を記載します。この手順では、ESP32用コンパイラとESP-IDFをユーザのホーム・ディレクトリの下にespディレクトリにダウンロードして展開します。他の場所にしたい場合は適宜読み替えます。

次のコマンドを実行してESP32用コンパイラをインストールします。ただしWindowsでESP-IDFのall-in-oneパッケージを使っている場合はこの手順は飛ばします。

```
$ cd ~
$ mkdir -p esp
$ cd esp
$ wget https://dl.espressif.com/dl/xtensa-esp32-elf-linux64-1.22.0-80-g6c4433a-5.2.0.tar.gz
```

```
$ tar xf xtensa-esp32-elf-linux64-1.22.0-80-g6c4433a-5.2.0.tar.gz
$ ls
xtensa-esp32-elf xtensa-esp32-elf-linux64-1.22.0-80-g6c4433a-5.2.0.tar.gz
```

ESP-IDFをGitHubから取得します。

```
$ git clone https://github.com/espressif/esp-idf
$ cd esp-idf
$ git checkout 30545f4cccec7460634b656d278782dd7151098e
$ git submodule update --init
$ export ESPIDF=~/.esp/esp-idf
$ export PATH=~/.esp/xtensa-esp32-elf/bin:$PATH
$ cd (MicroPythonのソースコードのディレクトリ)/ports/esp32
$ make -j8
```

MicroPythonが対応しているESP-IDFのバージョンをチェックアウト

途中、ESP-IDFをcloneした後にgit checkoutで特定のバージョンをチェックアウトしています。これは、MicroPythonのソースコードをビルドするのにESP-IDFの特定のバージョンが必要だからです。

現在のMicroPythonのESP32ポートがどのESP-IDFのバージョンに対応しているかは、ports/esp32/Makefile内のESPIDF\_SUPHASH変数に設定されています。

```
ports/esp32/Makefile 24行目
ESPIDF_SUPHASH := 30545f4cccec7460634b656d278782dd7151098e
```

環境変数ESPIDFが指しているESP-IDFのソースコードが、このバージョンでない場合、警告が表示され、MicroPythonのビルドに失敗したり、ビルドできても動かなかつたりする場合があります。極力指定されたりバージョンを使用するようにします。

ESP-IDFのcloneとチェックアウトが終わったら、忘れずに、