

アルゴリズム検証や組み込み後の動作検証にも

# 未知のモジュールをインタプリタ型言語Pythonで試しながら動かす

羽田野 大理

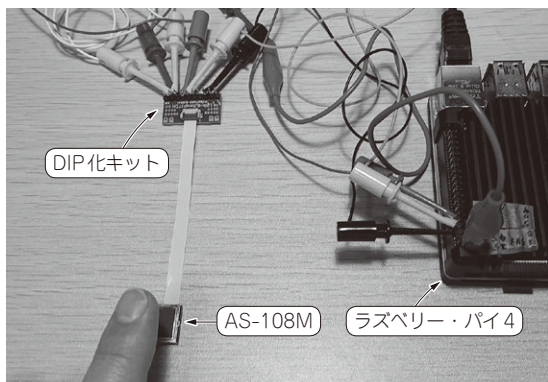


写真1 使ったことがない指紋認証モジュールをPythonとラズベリー・パイで制御した

デバイスを制御する場合、一般的にはマイコンを用意し、CやC++を使ってプログラミングをすることが多いと思います。しかし、組み込み向けCやC++ではライブラリの制約があり、デバイスの簡単な動作確認をするだけでも手間がかかると感じる人が多いです。

使用するデバイスや開発環境にもよりますが、Pythonを使って、デバイス制御を行える場合があります。

## Pythonを使う理由

本稿では、プロトタイピングなどに使いやすいPythonを使って、ラズベリー・パイで指紋認証モ

表1 指紋認証モジュールAS-108Mの主な仕様

仕様	内容
サイズ	15.8 × 14.5 × 3mm
読み取り解像度	160 × 160 (508 dpi)
接続方法	UARTまたはUSB
UART設定	57,600bps, 8ビット, パリティなし, ストップ・ビット2
認識時間	0.6s以下
電源電圧	3.3V
価格	2,793円(スイッチサイエンス)

ジュールの制御を行う例を紹介します(写真1)。ドライバが提供されていないモジュールや、自分で作ったモジュールを使うためにドライバを用意する場合、手順ごとにモジュールの挙動を確認しながら開発を進めたい場合があります。Pythonはインタプリタなので、ちょっとしたプログラムの確認を行いやすいですし、C言語に比べ文字列操作などのデータ整形がやりやすいです。

最終的にはC/C++で実装する場合でも、取りあえずモジュールを動かしたい場合などに特に便利です。

## ラズパイで指紋認証モジュールを使う準備

### ● 今回使ったもの

今回実験に使ったものを次に示します。

- ラズベリー・パイ4
- 指紋認証装置 AS-108M (Sparkfun Electronics)
- フレキシコネクタ DIP化キット (秋月電子通商)
- ICクリップ (5本程度)

使用したAS-108Mは、モジュールだけで指紋特徴データの算出、認証、指紋探索が可能です。主な仕様を表1に示します。

### ● 指紋認証装置との接続

ラズベリー・パイとAS-108MとはUARTで接続します。AS-108Mにはフィルム・コネクタが付いていますが、そのままでは使いづらいため、今回はDIPに変換する基板を使用しました。

この構成の場合、AS-108MとDIP基板とでピン配置が逆になるので気を付けてください。例えばAS-108Mの1番のピンがDIP基板上では10番のピンになります。図1に配線を示します。

### ▶ 配線に問題がないかの確認

配線が終わった段階で動作を確認しておきます。前述の配線図では使っていない1番のピン(DIP基板の10番ピン)は、指がセンサに乗っていると3.3Vのアクティブ“H”の信号を出力します。

このピンにテストを当て、センサに指を乗せたと