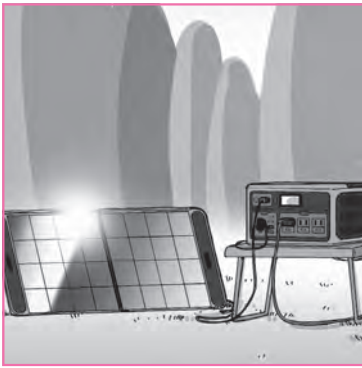


## 第4章

入手しやすいWi-Fi付きESP32マイコンでMPPT制御に挑戦!

# 太陽光電力IoTロガーの製作

田口 海詩 Uta Taguchi



マイコンを用いた製品のプログラミング開発では現在でもCやC++言語の利用が主流です。しかし、マイコンを用いた試作品や趣味の電子工作のように手っ取り早く動くものを作りたい場合には、MicroPython言語の利用をお勧めします。ESP32マイコ

ンにはビルド済みMicroPythonファームウェアが準備されており、現在も改良やメンテナンスが行われています。MicroPythonをお勧めする理由を紹介しつつ、ESP32を用いた太陽光発電電力のIoT測定ロガー(写真1)の製作をしてみます。

### MicroPythonを勧める理由

#### ● 対話型プログラミングがとにかく実験に便利

プログラム開発方法には対話型プログラミングという手法があり、素早く試作品を作りたい人たちの間で注目を集めています。プログラム・コード片を直接マイコンに投げかけ、マイコンがどのような動作するかを観察しながらプログラム開発を行っていく方法です。この開発方法は、あたかもマイコンと対話しているように見えるため、対話型プログラミングと呼ばれています。MicroPythonはインタープリタなので、プログラム・コード片をマイコンに直接投げることが可能です。

対話型プログラミングでプログラム開発を行うには、図1に示す対話型評価環境(Read-Eval-Print Loop: REPL)を準備する必要があります。REPLを用いてマイコンに直接プログラム・コード片を読み込ませ(Read)、マイコンや周辺機器を動作させ(Eval)、その結果を表示し(Print)、その結果からまたプログラ

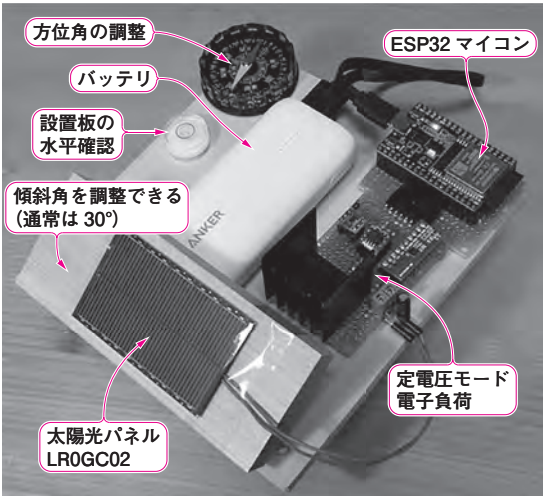


写真1 太陽電池MPPT評価実験のようす  
太陽光パネルで発電する電力が最大になるように、電子負荷の負荷条件をESP32マイコンを用いてMPPTプログラムで探索する。測定したデータはESP32のWi-Fi機能でクラウドに転送

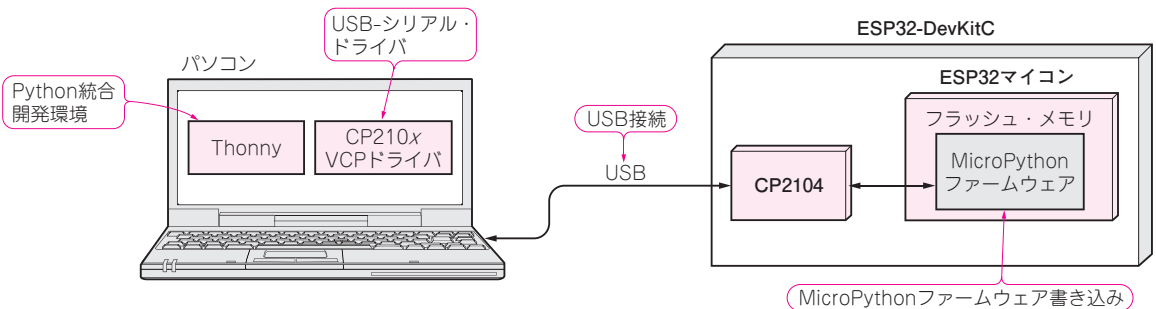


図1 対話型プログラミングは電子回路のトライアル&エラーが簡単に行える  
マイコンを用いた電子回路の開発にREPL開発環境を用いると、プログラム・コード片を直接マイコンに投げかけて電子回路のふるまいを観察しながらプログラムの動作確認を行うことができる