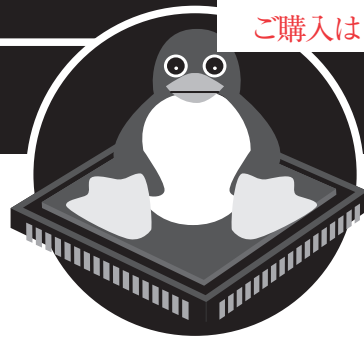


さまざまなシングル・ボード・コンピュータ向けに  
自分専用カスタム!

# Yocto Projectではじめる 組み込みLinux開発入門



第8回

Wi-Fi経由のLチカにトライ! ウェブ・アプリ用レシピ作り

三ツ木 祐介

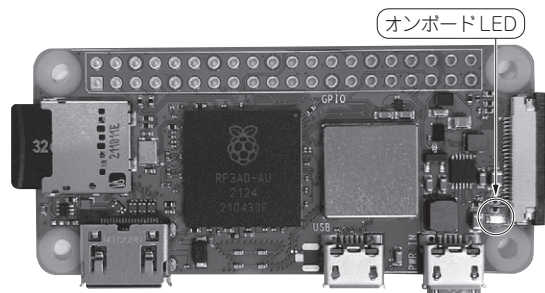


写真1 今回やること…ラズベリー・パイ Zero 2 W で動く LED 点滅用ウェブ・アプリを作ってレシピに組み込む

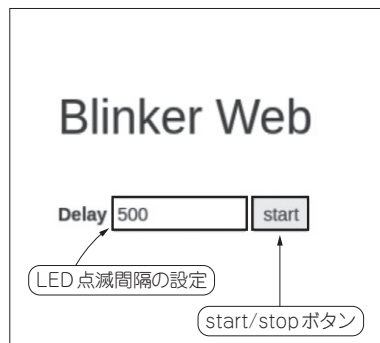


図1 作成するウェブ・アプリケーション「blinker\_web」実行時の画面イメージ  
LED点滅間隔の設定欄とstart/stopボタンを備える

## ● 今回やること…Lチカ用ウェブ・アプリを作る

前回(第7回, 2023年5月号)は, デバイス・ドライバを使ったLED点滅(通称Lチカ)の実装について解説しました(写真1)。ターゲット・ボードはラズベリー・パイ Zero 2 Wです。

GPIOドライバおよびLEDドライバを使う方法を紹介し, それぞれのスクリプトを作成しましたが, これらを単に起動時に自動実行するだけでは, 無線LANを使えるようにしたメリットが生かせていません。また, 実装しているのはただのLチカですが, ユーザが任意のタイミングで開始や停止, 点滅間隔の変更などができると面白そうです。

画面や入力デバイスのないヘッドレス構成では, ユーザからの入力を受け付ける方法として, ターゲット側で実行しているウェブ・アプリケーションにネットワーク越しにアクセスする方法がよく用いられます。今回は, 図1のようなウェブ・アプリケーションを作成し, 無線LAN越しにLチカの開始, 停止, 点滅間隔の変更を行えるようにします。

### ステップ1… ウェブ・アプリケーションの作成

#### ● 軽量フレームワーク「Flask」を使う

ウェブ・アプリケーションの実装には, Flask<sup>(1)</sup>を使います。Flaskは, Pythonで開発されているウェブ・アプリケーション・フレームワークです。Jinja

記法<sup>(2)</sup>によってHTMLを動的に出力できるので, Flaskが提供する機能を使えば, HTML自体の記述を大幅に減らせます。

Flaskは, Pythonで実装されているwaitress<sup>(3)</sup>というウェブ・サーバで実行できるので, ApacheやNginxなどの本格的なウェブ・サーバを導入しなくても, 手軽にウェブ・アプリケーションを提供できます。

#### ● こんなウェブ・アプリケーションを作る

##### ▶機能

ウェブ・アプリケーションの名前は, blinker\_webとします。blinker\_webでは次の機能を提供します。

- 点滅間隔の設定
- 開始/停止ボタン

##### ▶画面イメージと処理の流れ

画面のイメージを図1に示します。大まかな処理の流れとしてはブラウザで画面を操作するとサーバにPOSTリクエストを発行し, リクエストに応じてLEDを制御します。サーバにリクエストの詳細を伝えるためにJSONデータを使用します。JSONデータはPOSTリクエストと一緒に送信されます。処理の流