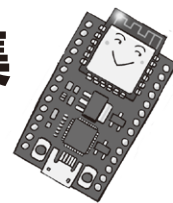


700円マイコンESP32ではじめる

逆引きMicroPythonプログラム集



角 史生

第2回

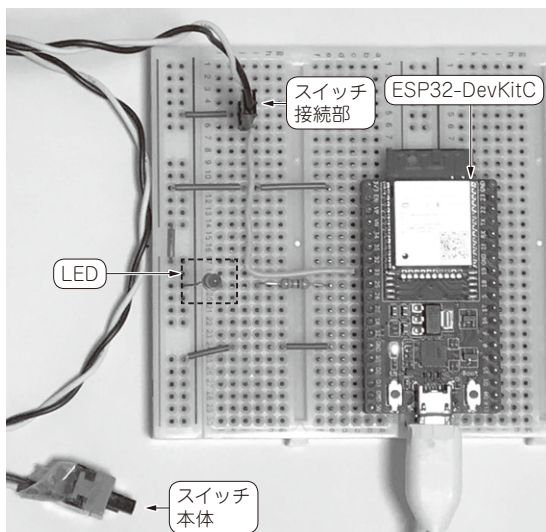
スイッチやボリューム、
ロータリ・エンコーダによる入力検出

写真1 ESP32-DevKitCとスイッチ，LEDをブレッドボード上で接続した様子

MicroPythonはリソースの少ないマイコン上でPython3と同じようにプログラミングできる環境の実現を目指して開発された言語処理系です。MicroPythonの特徴が生かせる開発用途としてプロトタイプ開発が挙げられます。

プロトタイプ開発では、試作、テスト、修正を繰り返しながら開発が進みますが、MicroPythonを用いることでトライ&エラーが容易になります。連載では用途別にMicroPythonの使用例を整理します。

1-1 スイッチを押したときだけLEDが点灯する

初めにスイッチ操作に連動してLEDを点灯させる例を示します。GPIOを入力、出力として使う場合の設定を示すとともに、ポーリング方式によるLED点灯および、割り込み方式によるLED点滅を行います。

● 回路

最も基本的な、スイッチを押したときにLEDが点

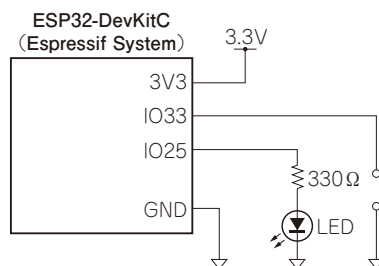


図1 LEDとの接続回路

灯する例を示します。回路図を図1に示します。写真1に示すのはブレッドボードを用いた試作回路です。連載で例として示す回路はいずれも、ESP32の電源をUSBケーブルから供給することを前提にしています。

● プログラム

MicroPythonのプログラムをリスト1に示します。LEDを制御するためIO25を出力に設定し、スイッチの状態を取得するためIO33を入力に設定しています。IO33は内蔵プルアップ抵抗を有効に設定することで、スイッチのプルアップ抵抗を省略可能にしています。スイッチが押されていないときは内蔵プルアップ抵抗により3.3Vとなり、スイッチが押されるとGNDに接続され0Vになります。これによりIO33の入力値が1から0に変化します。

スイッチに接続されたIO33の入力値を無限ループ内で取得し、値が0になるとスイッチが押されたと判断して、LEDを点灯させるためにIO25の出力を1に設定します。スイッチの接点が離れた場合、IO33の値が1になり、LEDを消灯させるためにIO25の出力を0に設定します。

1-2 スイッチを押すごとにLEDが点灯/消灯する

● 回路

回路は1-1項と同じものを使用します。プログラムでスイッチを押すごとにLEDが点灯/消灯を繰り返