

ラズベリー・パイ財団から発売された ハードウェア制御デバイスRP2040を試す プログラマブルI/O内蔵USB マイコン基板Raspberry Pi Pico

最終回
<3> Picoの特徴「PIO」の基本的な使い方

吉田 紹一 Shouchi Yoshida

ラズベリー・パイ財団は、独自開発したマイコンRP2040(写真1)と、そのRP2040を搭載したマイコン基板Raspberry Pi Pico(タイトルカット写真、以降Pico)を販売しています。550円程度と安価なことや、対応ボードが全世界で発売されつつあることから、注目が集まっています。

今回はRP2040の大きな特徴である高速プログラマブルI/O(PIO)の基本的な使い方を紹介します。開発環境はラズベリー・パイ4上に構築し、Visual Studio Code(以下、VSCoDe)をエディタとして使い(構築方法は前回参照)LEDの点滅まで行ってみます。

Pico マイコンRP2040 最大の特徴「PIO」とは

● ラズパイ Pico マイコン RP2040 の内部構成

ラズベリー・パイ財団が独自開発したCortec-M0+デュアル・コアのマイコンRP2040の仕様は次の通りです。

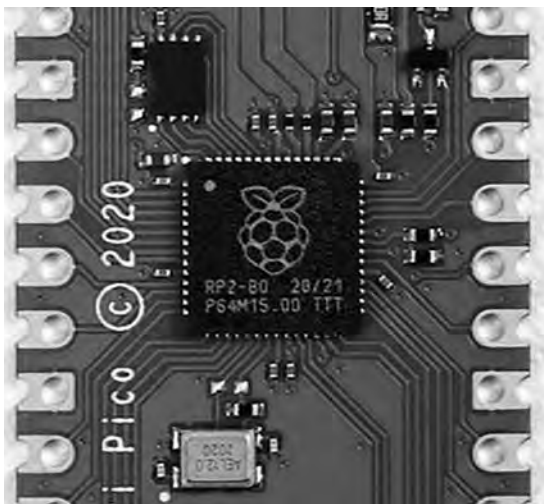


写真1 Raspberry Pi Picoに搭載されているマイコンRP2040最大の特徴「PIO」の基本的な使い方を紹介するRP2040は0.4 mmピッチの56ピンQFN(Quad Flat No Lead)パッケージ

- デュアル・コア Arm Cortex M0+ 最大動作周波数133 MHz
- SRAM 264 Kバイト(6バンク)
- 16 Mバイトまでのフラッシュ・メモリ対応QSPIバス
- 多機能GPIO 30個 そのうち4個はアナログ入力可能
- DMA コントローラ内蔵
- 2つのPLLにてシステム・クロックおよびUSBクロックを生成
- 内蔵された各種インターフェース
 - UART 2個
 - SPI 2個
 - I²C 2個
 - PWM16チャンネル
 - USB 1.1 コントローラ
- 内蔵温度センサ

公開されている内部ブロックを図1に示します。RP2040にはメインのCPUのほかにインターフェース制御用のサブCPUブロックPIO(Programmable Input/Output)が2つ内蔵されており、インターフェースをアセンブリ言語でプログラミングできます。

これにより、CPUの負担を増やさずに、特殊なプロトコルに対応できます。例えば、HDMI信号を作成してモニタ・ディスプレイに画像を出すことも可能です。

● CPUと独立して通信制御できる「PIO」

RP2040にはメインCPUのCortex M0+とは独立した2個のPIOブロックがあり、それぞれ4つのステート・マシン(SM)があります。図2に示すように、入出力されるデータを、一旦FIFOレジスタに蓄えることで、CPUとステート・マシンのタイミングのずれを吸収します。ステート・マシン用はプログラムで制御でき、さまざまなインターフェースに対応できます。

システム・クロックを元に動作しており、CPUが停止しても稼働し続けることができます。外的要因を