

マルチコア・プログラムを ベアメタルで実行する

豊山 祐一

本章ではPicoで簡単なマルチコアのプログラムを実行します。まずは2個のLEDを2つのCPUコアのそれぞれのプログラムから点滅させます。



Picoでは初期状態ではユーザのプログラムを実行するときに動作しているのはCPUコア0のみです。CPUコア1は休止状態です。CPUコア1でプログラムを実行させるには、まずCPUコア1を起床させ、実行するプログラムを指定します。

● CPUコア1は起動時に休止状態になる

そもそもCPUコア1はどのようにして休止状態となっているのでしょうか。

PicoのマイコンRP2040の場合は、電源を入れるとCPUコア0とCPUコア1の両方が同時に実行を開始します。つまり電源投入直後はCPUコア1も稼働しています。

このとき、2つのCPUコアは同一の初期状態であり、マイコン内蔵のROMに格納されている同じブート・プログラムの実行を始めます。このROMは製造時にメーカーのプログラムが書き込まれており、変更できません。

ブート・プログラムでは、SIO (Single-cycle IO block) のCPUIDレジスタの値を調べ、CPUコアごとのプログラムへ分岐します。CPUコア0は、通常のブートのシーケンスを実行し、最終的にはユーザのアプリケーション・プログラムを実行します。一方、CPUコア1はインターコアFIFOからデータの取得を試み、データがない場合は休止状態となります。起動時にはインターコアFIFOにデータはありませんので、ここでCPUコア1は休止状態となります。

● WFE命令による休止とSEV命令による起床

CPUコア1は、ブート・プログラムの中でWFE (Wait For Event) 命令を実行して休止状態になります。

正確にはWFE命令はイベントが発生するまでCPUコアを低消費電力モードに移行させます。低消費電力モードではCPUコアはプログラムを実行しません。

CPUコアをWFE命令による低消費電力モードから復帰させる方法は、規定の外部イベントを発生させるか、他のCPUコアのプログラムからSEV (Send Event) 命令を実行するかです。

つまり、CPUコア0のプログラムでSEV命令を実行すれば、CPUコア1の実行を再開させることができます。

● CPUコア1の起床シーケンス

再開したCPUコア1は、WFE命令以降のブート・プログラムを実行します。ここでは次の順番で処理が行われます。

- ① CPUコア1はインターコアFIFOからデータを受信し、データが“0”、“0”、“1”の順番で受信すると②に進みます。もし、データの値が異なったり、受信できなかったりした場合は、CPUコア1はWFE命令を実行して再び休止します
- ② インターコアFIFOからデータを受信し、その値を例外ベクタ・テーブルのアドレスとして設定します
- ③ インターコアFIFOからデータを受信し、その値をスタックの開始アドレスとして、SP (スタックポインタ) に設定します
- ④ インターコアFIFOからデータを受信し、その値をプログラムの実行アドレスとして、実行を開始します

以上から、CPUコア1を起床させるには、CPUコア0のプログラムからSEV命令の後、インターコアFIFOに定められた順番でデータを送信すればよいことが分かります。

また、CPUコア1はインターコアFIFOから受信したデータを、その都度そのままCPUコア0に向けてインターコアFIFOで送信しています。よってCPUコア0は送信したデータと同じデータが返ってくることで、相互の通信が成功していることを確認できます。