

第2章

安定性や対応マイコンの多さなどから試作に使いやすい

ITRON仕様OS TOPPERS/ASPの載せ方

石岡 之也

第2章ではリアルタイムOS搭載第2弾として、TOPPERS/ASPの搭載にチャレンジします。

● ITRON仕様OSとは

ITRON^{アイトロン}は、坂村 健氏らによるTRONプロジェクト(現トロンフォーラム)が策定した組み込みリアルタイムOS(RTOS)のカーネルの仕様です。1980年代から仕様策定や公開が始まり、現在でも使われている国産のRTOSです。ITRON仕様のOSには、有償/無償を含めて複数の製品があります。今回はラズベリー・パイPico(以降、Pico)のコアCortex-M0+と同じコアを持つマイコンをサポートしているTOPPERS/ASPを移植して動かすことにしました。TOPPERS/ASPは、TOPPERSプロジェクトがソースコードを公開しているものです。

● NUCLEO向けのTOPPERS/ASPを流用して作る

TOPPERS/ASPは、TOPPERSプロジェクトが管理、公開しているITRON仕様RTOSの1つです。新機能の開発は行われていませんが、バグ・フィックスや各種マイコン・ボードへの対応は継続して行われているので、安定性や対応しているマイコンの多さなどから、試行などに使いやすいRTOSです。

今回はCortex-M0+コアのマイコンSTM32L073RZを搭載するNUCLEO-L073RZボード(STマイクロエレクトロニクス)用のソースコードを流用しました。

● 実験すること

写真1のボードを作成し、タスクの動作や切り替えを確認しました。書き込んだTOPPERS/ASPのプログラムが動作しているかどうかはLEDの点滅によって確認ができました。

TOPPERS/ASPのPico対応

RTOSの主要な制御は、マイコンや周辺ハードウェアの初期化、タスクの切り替え時に行うコンテキス

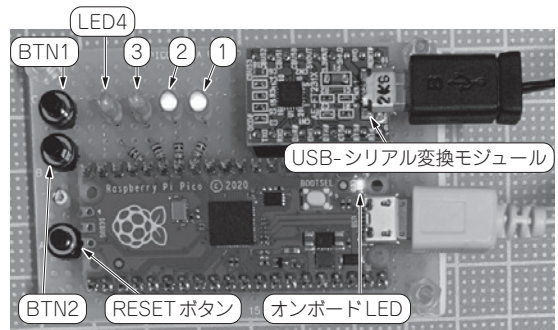


写真1 NUCLEO向けのTOPPERS/ASPをPicoに移植した

ト・スイッチ、例外や割り込みのハンドリングです。また、マイコンの初期化やコンテキスト・スイッチ、割り込みのハンドリングはマイコン・コアの種類に大きく依存し、Picoのマイコン・コアであるCortex-M0+用のプログラムがあれば、ほぼそのまま利用できます。周辺ハードウェアの初期化はRTOSがサポートするハードウェアにもよりますが、TOPPERS/ASPを動かすためには以下の項目への対応が必要になります。

● システム・クロック

システム・クロックは、Pico専用の処理が必要です。TOPPERS/ASPのシステム・クロック初期化処理から公式SDK内の`clocks_init()`を呼び出す処理へ改造しました。

● オンチップ・ハードウェア初期化

オンチップ・ハードウェアの初期化はPico専用の処理が必要です。TOPPERS/ASPのハードウェア初期化処理内に公式SDK内の`runtime_init()`で実行しているオンチップ・ハードウェアのリセット処理を追加しました。また、Picoのマイコンは2コア構成ですが、今回流用したTOPPERS/ASPのコードはマルチコアに対応していないことと、ブート時に2コアの一方を止めておく必要があるため、TOPPERS/