

第1章 電源ON直後のマイコンの動作と参照するデータ

一般的なマイコンの起動処理と例外ベクタ・テーブル

豊山 祐一

● 第2部の概要

第2部ではマイコンの起動処理プログラムを作成します。OSといえどもマイコンのプログラムなので、起動処理が必要です。

マイコンに電源が入りリセットが行われると起動処理が実行されます。起動処理の主な目的はハードウェアを初期化してプログラムの実行環境を整えることです。C言語のプログラムであれば、main関数が実行されるまでの処理と言えます。

起動処理は通常はOSやファームウェアによって実行されますので、アプリケーションのプログラムを作成している限りは意識することはありません。しかしOSの開発ではこの起動処理のプログラムから作成する必要があります。そこで、第2部では起動処理を行い、main関数を実行するまでのプログラムを作成します。最終的にはOSのmain関数を実行するのが目標ですが、ここでは起動すべきOSがまだないので、OSの代わりにhello, worldを表示するmain関数を実行します。

● この章でやること

本章では一般的な起動処理と、関連する例外ベクタ・テーブルについて説明します。そして、ラズベリー・パイ Pico (以降、Pico) のマイコン RP2040 の起動処理と例外ベクタ・テーブルについて見てみます。

起動処理は、使用するマイコンやボードの仕様によって決まりますが、大きくは以下の2つのタイプに分けられます。



● ROM上のプログラムを実行するタイプ

RTOSを使用する組み込みシステムのマイコンの多くは、ROM上に配置されたプログラムを実行します。ArmのマイコンであればCortex-Mがこれに該当します。

このタイプのマイコンの起動処理は比較的にシンプ

ルです。マイコンがリセットされるとROM上の定められたアドレスからプログラムを実行しますので、そこに起動処理のプログラムを配置しておきます。

Cortex-Mの場合は、ROMの先頭に例外ベクタ・テーブルがあり、そこにリセット時に実行開始するプログラムのアドレスが設定されています(例外ベクタ・テーブルはこの後に説明)。リセット時に実行されるプログラムはリセット・ハンドラと呼ばれます。

● RAM上にプログラムを転送して実行するタイプ

プログラムをRAM上に転送して実行するタイプのマイコンは、比較的高機能なプロセッサなどに多くあります。ArmのマイコンであればCortex-Aがこれに該当します。OSにLinuxを使用する場合は主にこのタイプのマイコンを使用します。

電源投入時にはRAMの内容は消去されていますので、まずブートROMと呼ばれる起動処理専用のROM上のプログラムが実行され、外部記憶装置からRAMにプログラムが転送されます。外部記憶装置はフラッシュ・メモリやSDカード、ハード・ディスクなどさまざまです。

ブートROMのプログラムは通常はユーザが変更できません。また、RAMへのプログラムの転送はブート・シーケンスと呼ばれる数段階の処理が行われるのが一般的です。

● 例外と割り込みの動作

例外と割り込みは、特定のイベントが発生すると実行中のプログラムに割り込んで、ハンドラと呼ばれる特別なプログラムを実行する、マイコンのハードウェアの機能です。ハンドラの実行が終了すると、元のプログラムが再開されます。

例外と割り込みはハードウェアにより実現されるので、具体的な仕様はマイコンの種類により決まります。なおArmのマイコンでは、割り込みは例外の一種として扱われています。

例外や割り込みのハンドラは、一般的にはC言語で単純に記述できません。ハンドラの起動や復帰の方法