

タイマ割り込みLED点灯制御 サンプル・プログラムの動作

鹿取 祐三

ここでは、第6章で利用したサンプル・プログラムを題材に、どんなシステムを作成するときにも必要となるスタートアップ手順と、基本的な内蔵周辺機能として、クロック発生回路、I/Oポート、コンペアマッチ・タイマ、割り込みコントローラの使い方について解説する。
(筆者)

1. C言語プログラムの スタートアップ手順

● スタートアップ処理

RX62Nを含むRXファミリのスタートアップ手順は、ルネサス エレクトロニクス(以下ルネサス)製のほかのマイコンとほとんど同じであり、図1のようになっています。従来からのM16C/R8C/H8/SHファミリを利用されたことのある読者であれば、見慣れた関数名でしょう。ここでは各関数の詳細な説明は省き、その概要のみを解説します。

RXファミリはリセットを含む割り込みに対して、ベクタ方式を採用しています。中でもリセットやNMI、一般例外に対しては、配置場所が決められている固定ベク

タ・テーブルを利用します。固定ベクタ・テーブルはアドレスの一番大きい場所へ配置し、その中でもリセット・ベクタは最上位アドレスFFFFFFFChとなっています。サンプル・プログラムで固定ベクタ・テーブルを定義しているのがvecttbl.cに記載されているFixed_Vectors配列であり、その要素の最後の初期値にはPowerON_Reset関数が登録されています。結果、リセットの入力によって、resetprg.cにあるPowerON_Reset関数が実行されます。

PowerON_Reset関数は、各種CPU内部レジスタの初期化を行った後、ハードウェアの初期化を行うHardwareSetup関数と静的変数の初期化を行う_INITSCTライブラリ関数、そしてこれらを実行後、main関数を呼び出します。

● ハードウェアの初期化

hwsetup.cにあるHardwareSetup関数では、ハードウェア的な初期化を行います。例えば、内部クロックの設定、外部メモリを使う際の外部バスの設定などを実施します。サンプル・プログラムでは内部クロックの初期化のみ記載されているので、外部バスを使う際は設定を追加してください。

_INITSCTライブラリ関数では、C言語のプログラムをROM化する際に必須の静的変数の初期化を行います。具体的には、初期値のある静的変数の初期値をROM領域からRAM領域へのコ

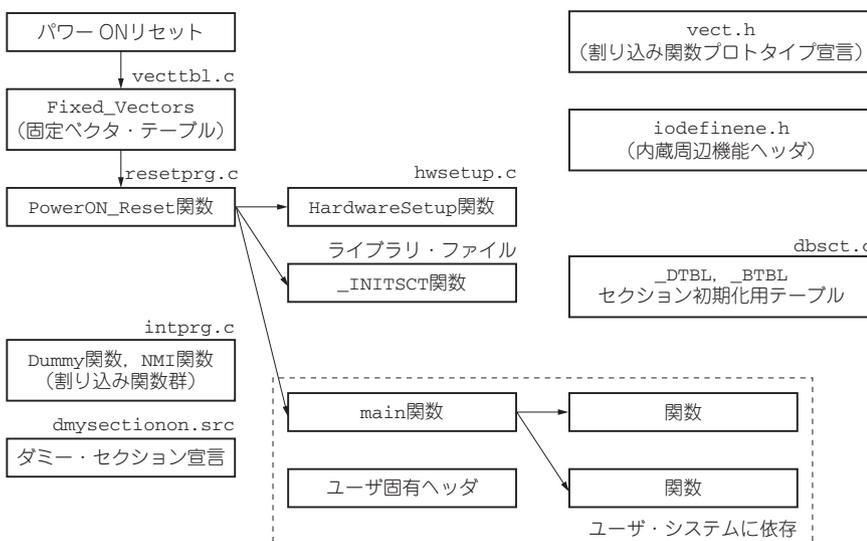


図1 RXファミリのC言語プログラムのスタートアップ手順