Hterm+Renesasモニタでdebug目的のHEWでのコンパイル（タイマ割り込みを使う場合）

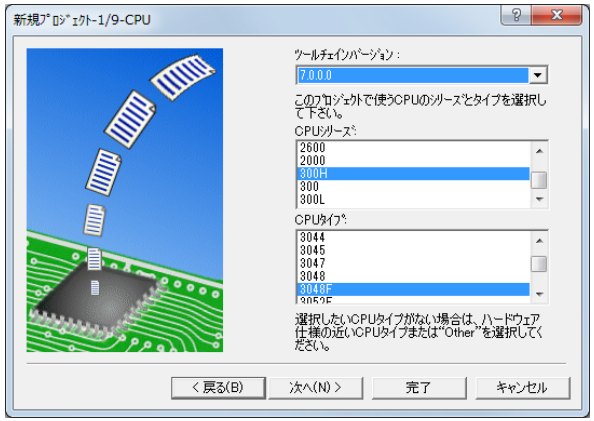
小坂

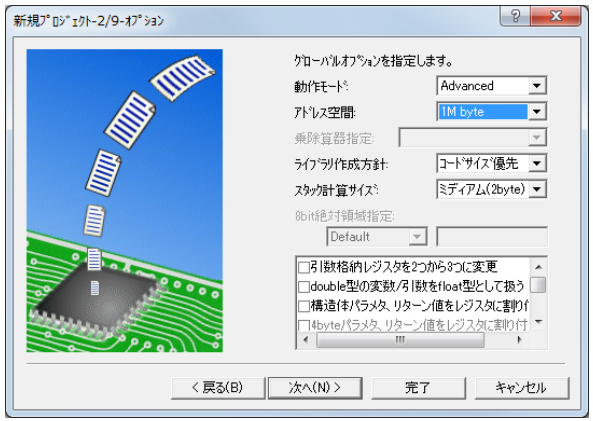
要点

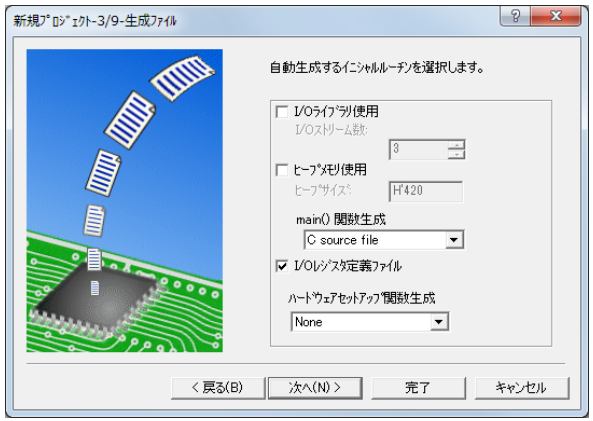
- H8/3048はRAMが小さく，モニタでdebugすることのできるユーザ領域は約3kbyteなので大きなプログラムは実行できない
- 幸い，モニタにはprintfとscanf相当のルーチンがあるため，ユーザプログラムを小さくできる.

プロジェクト生成時

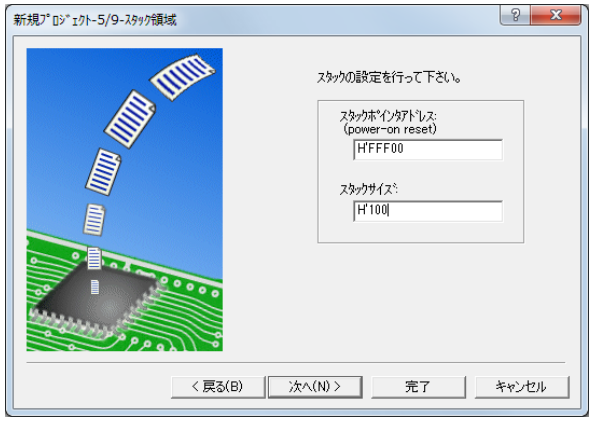
- アドレス空間は1Mbyte
- HEAPは使わないので削除
- ライブラリは何も追加しない
- スタックサイズは0x100に一応設定
- スタックポインタの設定はせず，モニタに任せる
- 割り込みベクタは自分で生成

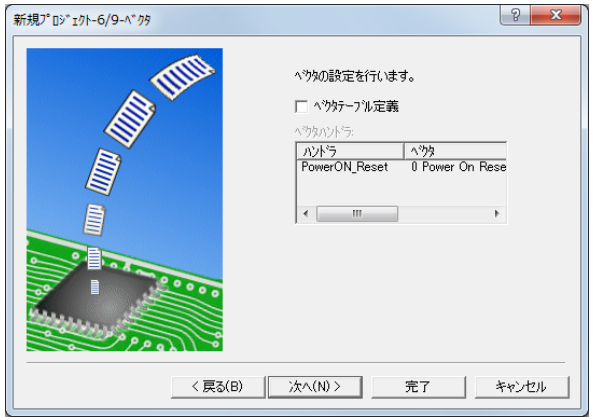

新規ﾌﾟﾛｼﾞｪｸﾄ-1/9-CPU
ツールチェインバージョン：
7.0.0.0
このﾌﾟﾛｼﾞｪｸﾄで使うCPUのｼﾘｰｽﾞとタイプを選択して下さい。
CPUｼﾘｰｽﾞ：
2600
2000
300H
300
300L
CPUﾀｲﾌﾟ：
3044
3045
3047
3048
3048F
選択したいCPUタイプがない場合は、ハードウェア仕様の近いCPUタイプまたは"Other"を選択してください。
< 戻る(B)
次へ(N) >
完了
キャンセル


新規ﾌﾟﾛｼﾞｪｸﾄ-2/9-ｵﾌﾟｼｮﾝ
ｸﾞﾛｰﾊﾞﾙｵﾌﾟｼｮﾝを指定します。
動作ﾓｰﾄﾞ：
Advanced
ｱﾄﾞﾚｽ空間：
1M byte
乗除算器指定：
ﾗｲﾌﾞﾗﾘ作成方針：
ｺｰﾄﾞｻｲｽﾞ優先
ｽﾀｯｸ計算ｻｲｽﾞ：
ﾐﾃﾞｨｱﾑ(2byte)
8bit絶対領域指定：
Default
引数格納レジスタを2つから3つに変更
double型の変数/引数をfloat型として扱う
構造体パラメタ、リターン値をレジスタに割り付
4byteパラメタ、リターン値をレジスタに割り付
< 戻る(B)
次へ(N) >
完了
キャンセル


新規ﾌﾟﾛｼﾞｪｸﾄ-3/9-生成ﾌｧｲﾙ
自動生成するｲﾆｼｬﾙﾙｰﾁﾝを選択します。
I/Oﾗｲﾌﾞﾗﾘ使用
I/Oｽﾄﾘｰﾑ数：
3
ﾋｰﾌﾟﾒﾓﾘ使用
ﾋｰﾌﾟｻｲｽﾞ：
H'420
main() 関数生成
C source file
I/Oﾚｼﾞｽﾀ定義ﾌｧｲﾙ
ﾊｰﾄﾞｳｪｱｾｯﾄｱｯﾌﾟ関数生成
None
< 戻る(B)
次へ(N) >
完了
キャンセル


新規ﾌﾟﾛｼﾞｪｸﾄ-4/9-標準ﾗｲﾌﾞﾗﾘ
ﾗｲﾌﾞﾗﾘ：
C言語の選択 (L)：
C (C89)
runtime：実行時ルーチン
new：メモリ操作用ライブラリ
ctype.h：文字操作用ライブラリ
math.h：数値計算用ライブラリ
mathf.h：数値計算用ライブラリ(各関数に
stdarg.h：可変個の実引数アクセス用ライ
stdio.h：入出力用ライブラリ
stdlib.h：標準処理用ライブラリ
string.h：文字列操作用ライブラリ
全て有効
全て無効
< 戻る(B)
次へ(N) >
完了
キャンセル

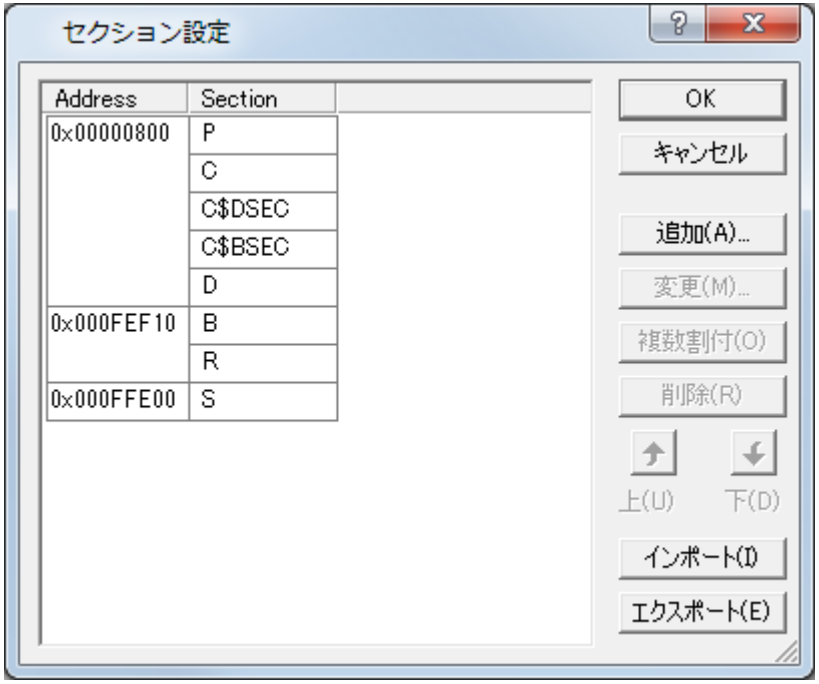

新規ﾌﾟﾛｼﾞｪｸﾄ-5/9-ｽﾀｯｸ領域
ｽﾀｯｸの設定を行って下さい。
ｽﾀｯｸﾎﾟｲﾝﾀｱﾄﾞﾚｽ：
(power-on reset)
H'FFF00
ｽﾀｯｸｻｲｽﾞ：
H'100
< 戻る(B)
次へ(N) >
完了
キャンセル

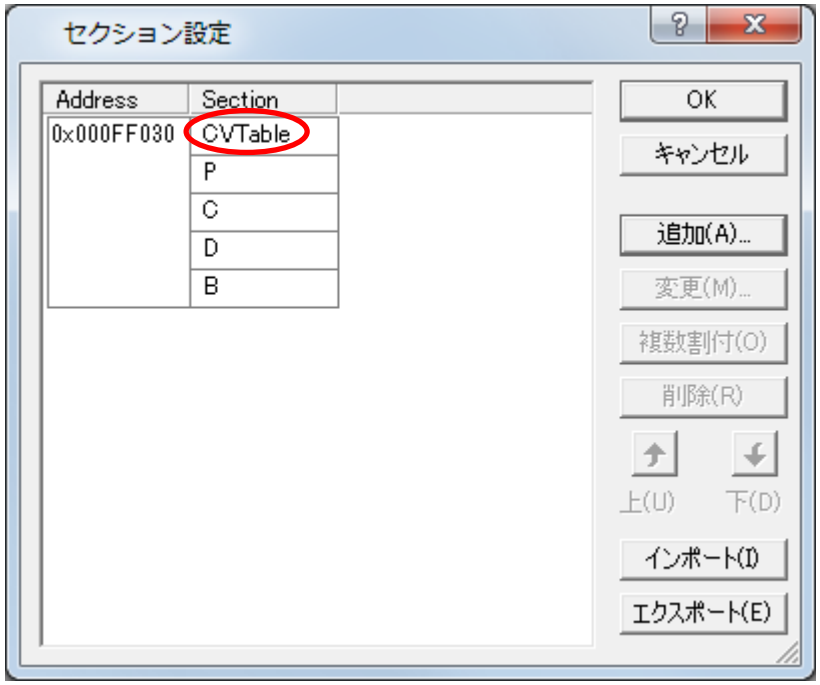

新規ﾌﾟﾛｼﾞｪｸﾄ-6/9-ﾍﾞｸﾀ
ﾍﾞｸﾀの設定を行います。
ﾍﾞｸﾀﾃｰﾌﾞﾙ定義
ﾍﾞｸﾀﾊﾝﾄﾞﾗ：
ﾊﾝﾄﾞﾗ
ﾍﾞｸﾀ
PowerON_Reset
0 Power On Rese
< 戻る(B)
次へ(N) >
完了
キャンセル

コンパイル時のオプション

- Release，Debug切り替えでDebugに
- dbsct.cはプロジェクトから削除
- ビルドのtoolchaineで次の設定を行う
  - コンパイラ：最適化なし
  - 最適化リンカ：入力：エントリポイント：「_main」に設定
  - 最適化リンカ：出力：出力形式：バイナリ（ELF/DWARFアブソリュート付）
  - 最適化リンカ：出力：オプション項目：ROMからRAMへマップを削除
  - 最適化リンカ：セクション：次のページのように設定

セクションをすべてRAMのユーザ領域に移動，Vtable追加



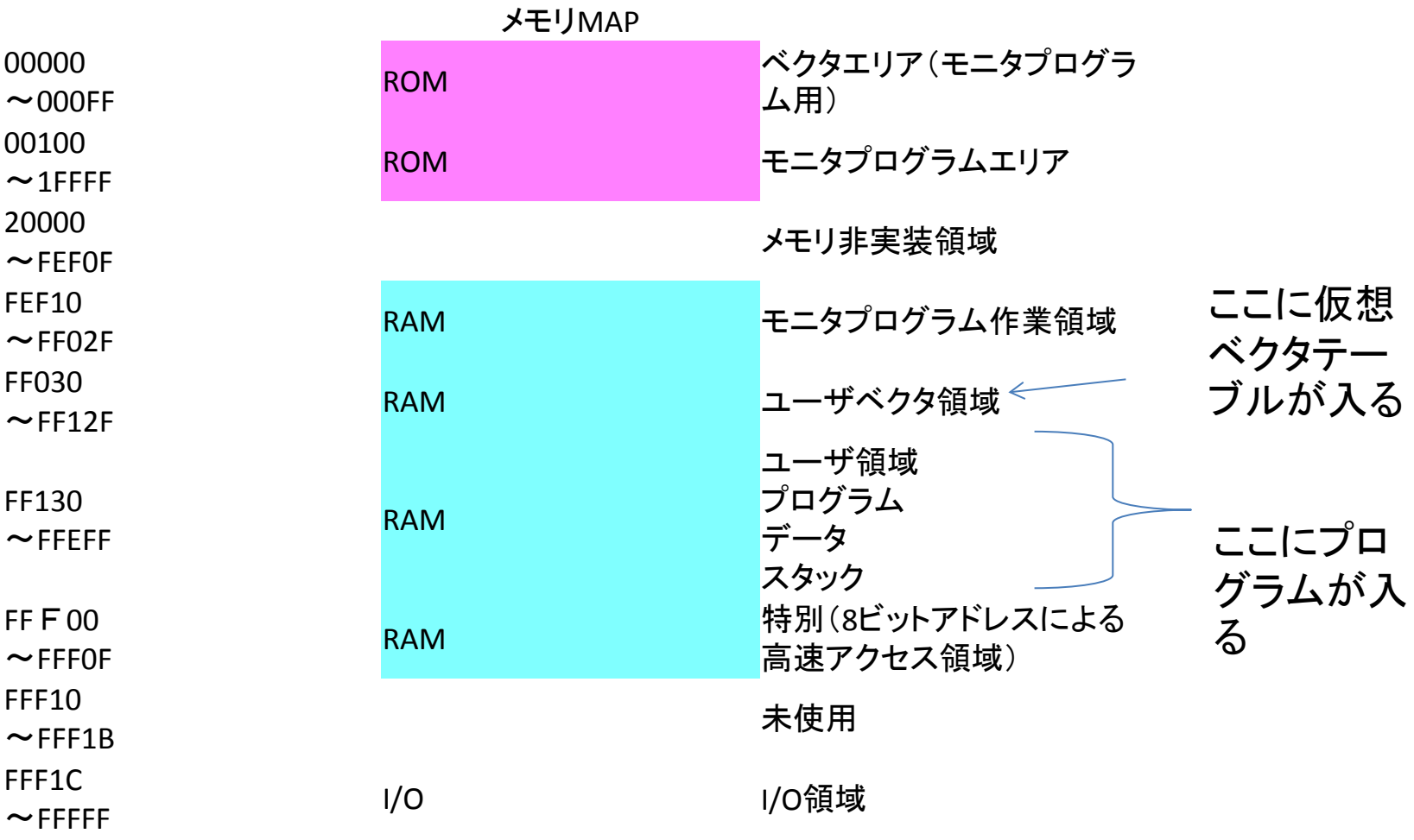


CVtableというのはベクタテーブル用のセクション名
後で作るベクタテーブルセクション名の前にCを付けたもの

ビルドしてHtermへ

- vectortable.cを追加する
- HEWでビルドすると
- プロジェクトのディレクトリ内の¥Debugに xxx.absができる
- 同じ場所にxxx.mapができるのでサイズを確認

モニタのメモリマップ


メモリMAP
00000
～000FF
00100
～1FFFF
20000
～FEF0F
FEF10
～FF02F
FF030
～FF12F
FF130
～FFEFF
FF F00
～FFF0F
FFF10
～FFF1B
FFF1C
～FFFFF
ROM
ROM
RAM
RAM
RAM
RAM
I/O
ベクタエリア（モニタプログラム用）
モニタプログラムエリア
メモリ非実装領域
モニタプログラム作業領域
ユーザベクタ領域
ユーザ領域
プログラム
データ
スタック
特別（8ビットアドレスによる高速アクセス領域）
未使用
I/O領域
ここに仮想ベクタテーブルが入る
ここにプログラムが入る

Htermでのソースレベルデバッグ

- Htermを起動し，F9キーで h83048DebugIntMiniSample.absをロードする.
- ソースファイルは複数ある
- h83048DebugIntMiniSample.cを表示して，ブレークポインタを設定して，F5キーでゴー（起動は，main()の先頭）

Htermでのデバッグの様子

Breakポイントをセットしなければ，LED1が1秒ごとに点灯消灯する．
Breakポイントを割り込み関数中とmainの両方にセットしておくと，両方で止まる