

統合環境Hew (Ver 3.01) 添付スタートアップ関数を使用した場合の 新ワークスペースおよびプロジェクトを登録する方法 (SH-2版)

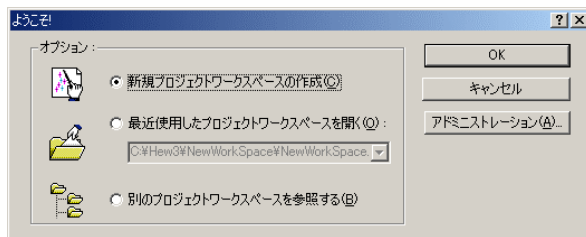
ルネサスC言語用統合環境「Hew Ver 3.01 (release1)」で H-debugger 用に新ワークスペース/プロジェクトを登録する手順方法を説明します。

説明を明確にするために、名前等を仮に決めて例に沿って説明を進めます。

ワークスペース名	NewWorkSpace2		
プロジェクト名	Project2		
登録モジュール名	SH705x.c	Cファイル	メインモジュール (アプリ用)
Hew添付ファイル	Resetprg.c Intprg.c Dbsct.c Vecttbl.c	Cファイル	スタートアップモジュール 割込みハンドラモジュール 定数転送用セクション管理宣言 ベクター定義モジュール
CPUタイプ	SH7051F		

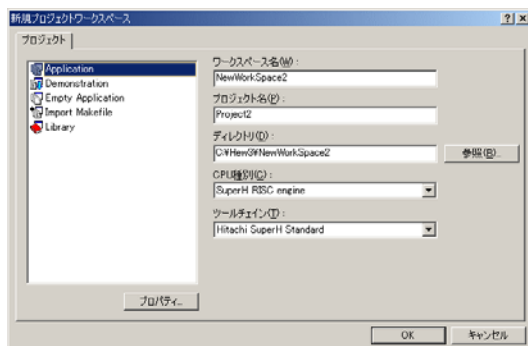
1. 新ワークスペースの登録方法

“HEW” 起動させます。



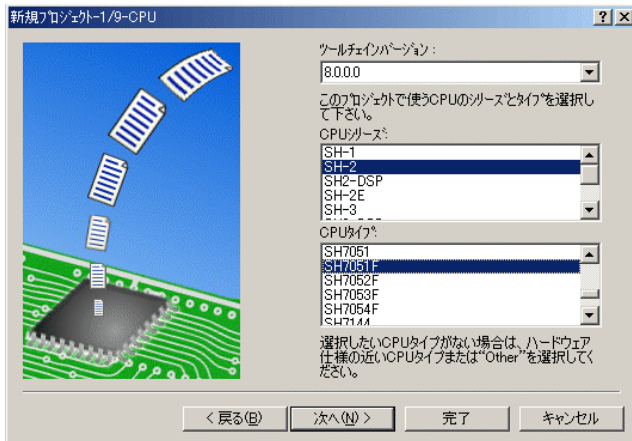
“新規プロジェクトワークスペース”をチェックしての **OK** をクリックする。

もしくは、**キャンセル**後に、[ファイル]-[新規ワークスペース]をクリックします。



ワークスペース名	“NewWorkSpace2”
プロジェクト名	“Project2”
ディレクトリ	“C:\HEW3\NewWorkSpace2”
CPU種別	“SuperH RISC engine”
ツールチェーン	“Hitachi SuperH Standard”
プロジェクト	Application

この項目を確認後、**OK** をクリックして下さい。



CPU シリーズを“SH-2”に選択する。
CPU タイプを“SH7051F”に選択する。

確認後、**次へ>**をクリックします。



次へ>をクリックして下さい。



ここでのHewが用意したROM化支援関数は、使用しませんので全てのチェックを外して下さい。

main() 関数生成を“None”に選択する。

確認後、**次へ>**をクリックして下さい。



C言語ライブラリの選択です。この例では、その他ライブラリを使用しません。

次へ>をクリックして下さい。



スタックボトムの設定です。RAM位置を示す為「HFFFFFFC」にする。

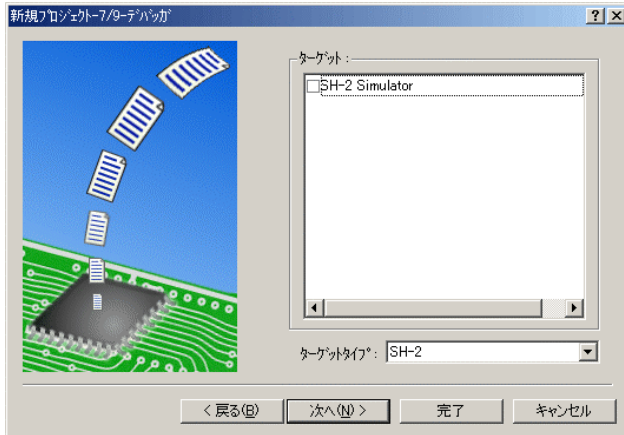
スタックサイズはデフォルト値にします。

次へ>をクリックして下さい。



ここで明示されたHew作成スタートアップ関数を使用しますので、デフォルトの状態で、

次へ>をクリックして下さい。



シミュレータの設定ですが使用しませんのでチェック無しの状態で、

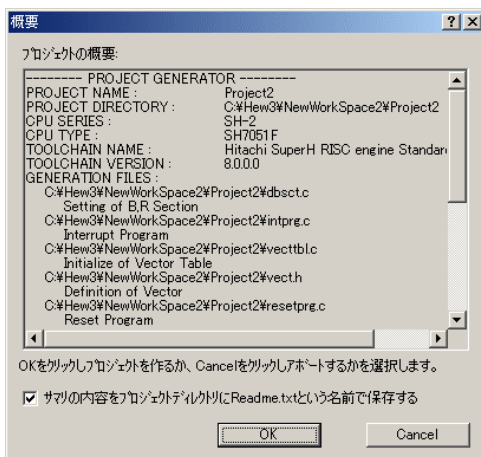
次へ>をクリックして下さい。



ここで最終になります。

使用するCモジュールを表示します。

この状態で**完了**をクリックして下さい。



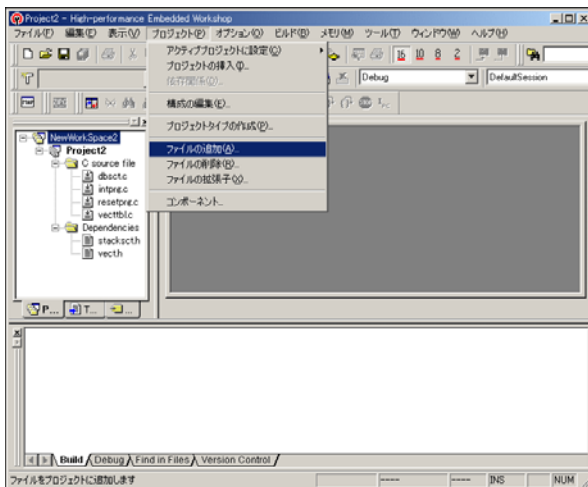
確認画面が表示されますので、**OK**をクリックして下さい。

ここまでの操作が新規プロジェクトの登録方法です。

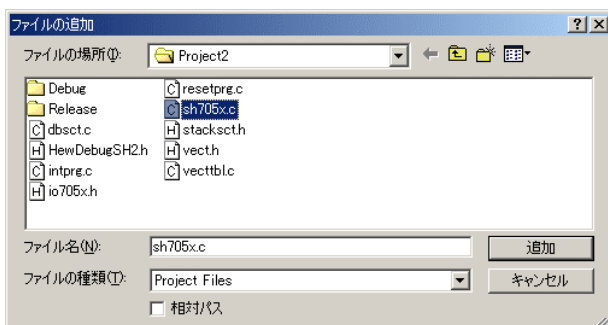
2. プロジェクトに希望モジュール(ソースファイル)を登録する方法

準備: 作成済みのソースファイルを”C:\Hew3\NewWorkSpace2\Project2”にコピーして下さい。

SH705x.c ” ¥sample¥HewSH22_0¥SH7051¥”
Io705x.h (左記ソースは、製品 CD の上記フォルダに入っています。)
(iodefine.h) 又は、Hew提供の I/O 定義ファイルを使用しても構いません。
HewDebugsh2.h



[プロジェクト]-[ファイルの追加]をクリックします。



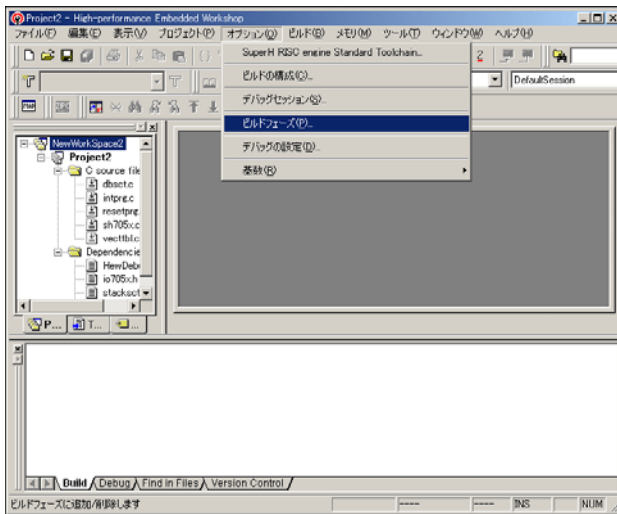
下記 1 ファイルを指定して下さい。

Sh705x.c

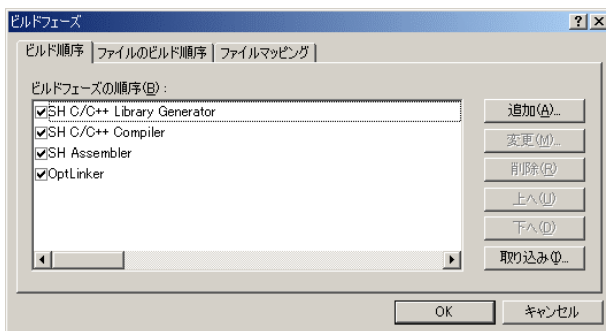
選択後、**追加**をクリックします。

この操作によりプロジェクトにモジュールが登録されました。

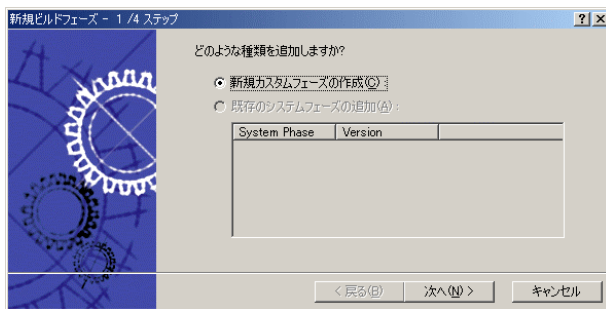
3. シンボルコンバータ「HCsymconv」を登録する。



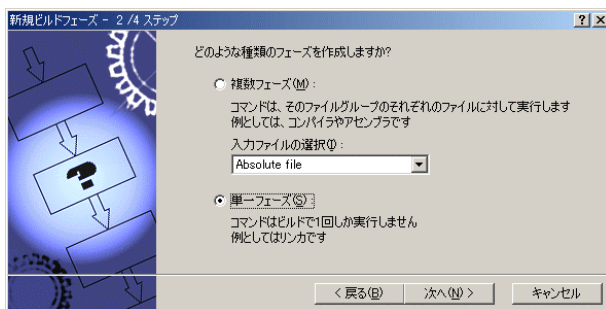
[オプション]-[ビルドフェーズ]をクリックします。



追加をクリックします。

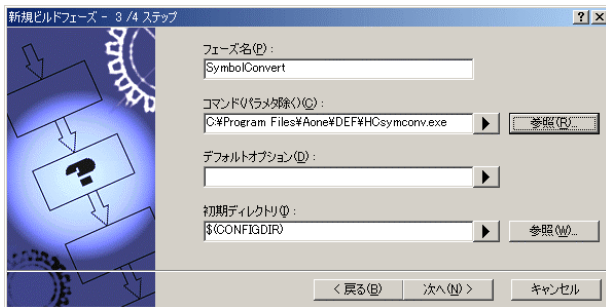


次へ>をクリックします。

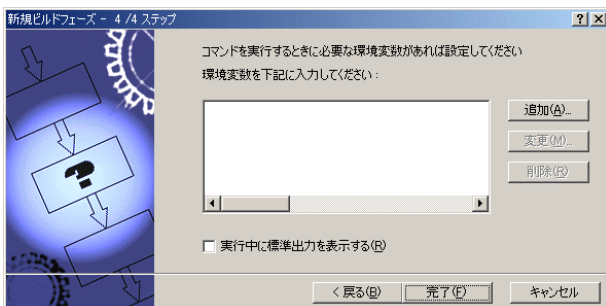


単一フェーズ側にチェックをします。

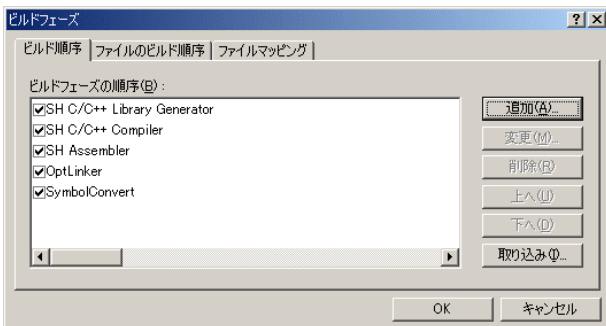
次へ>をクリックします。



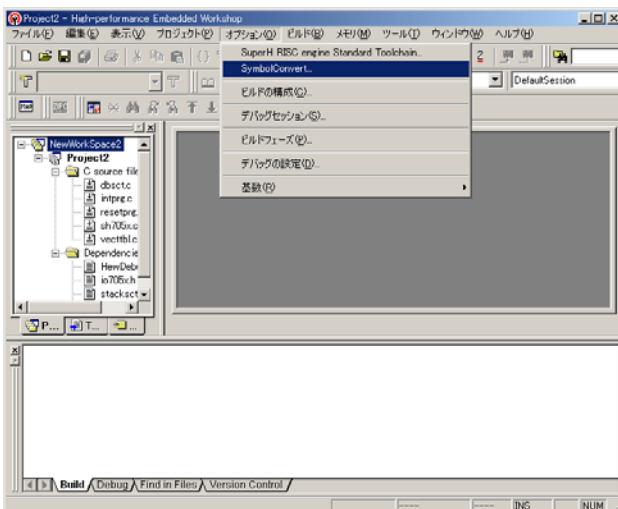
フェーズ : SymbolConvert
 コマンド : C:\ProgramFiles\Aone\DEF\HCSymconv.exe を選択する。
 (デフォルト)
 初期ディレクトリ : \$(CONFIGDIR)
 [次へ>] をクリックします。



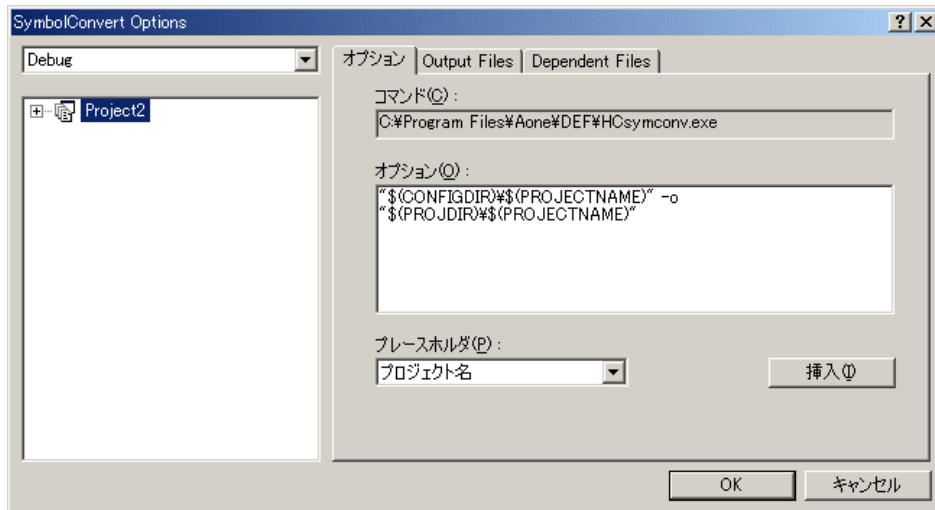
[完了] をクリックします。



[OK] をクリックします。



[オプション]-[SymbolConvert] をクリックします。



オプションに下記内容を設定する。

"\$(CONFIGDIR)\\$(PROJECTNAME)" -o "\$(PROJDIR)\\$(PROJECTNAME)"
 (入力ファイル名) (出力先名)

注意事項

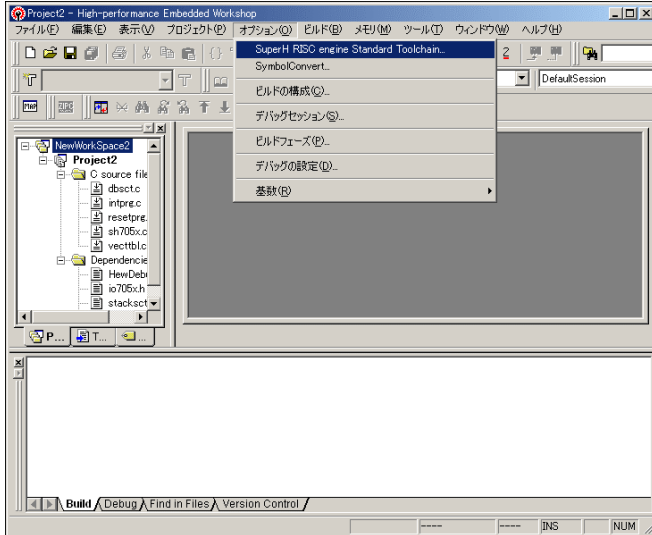
- 1) ディレクトリ名に ' ' スペースを使用している場合は、“ダブルクォートで囲んで下さい。
 “\$(CONFIGDIR)\\$(PROJECTNAME)” -o “\$(PROJDIR)\\$(PROJECTNAME)”
- 2) \$(PROJECTNAME)の先頭に「¥」記号を挿入して下さい。(手入力)
- 3) オプションSW「-o」の両端には、スペースを入れてください。(手入力)
- 4) この設定例は、後説明の「*.mot」ファイルの生成されるディレクトリと同じ場所にシンボルコンバータが生成する「*.sym/*.lin」を置く為の指定です。
 <コンフィグレーションDIR>に生成させたい場合は、
 “\$(CONFIGDIR)\\$(PROJECTNAME)”
 の指定のみで構いません。
 この場合は「*.mot」の生成場所を同じく<コンフィグレーションDIR>にして下さい。

追加事項 (HCSymconv.exe スイッチ説明)

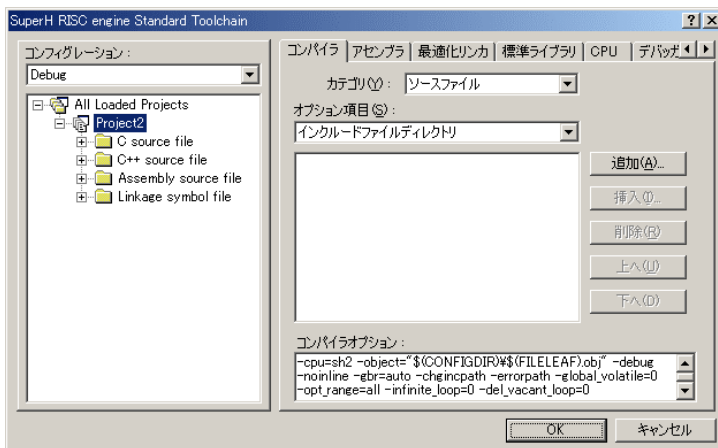
- 1) [-o] (省略可) 出力ファイル名を指定
- 2) [-r] (省略可) モジュール毎のディレクトリ情報を作成しない。ELF専用(Ver3.2xxから)
- 3) [-s] (省略可) ラインシンボル情報をソート(アドレス順)しない。(Ver3.2xxから)
- 4) [-i] (省略可) 重複モジュール情報を削除する。(Ver3.3xxから)
- 5) [-g] (省略可) スタティック変数をグローバル化する。(Ver3.6xxから)
- 6) [-m] (省略可) 重複モジュール情報をCソースにマージする。(Ver3.80Bから)
- 7) [-f] (省略可) 使用インクルードファイルをCViewに登録する。(Ver3.80Bから)

4 . ツール(ライブラリ)の設定

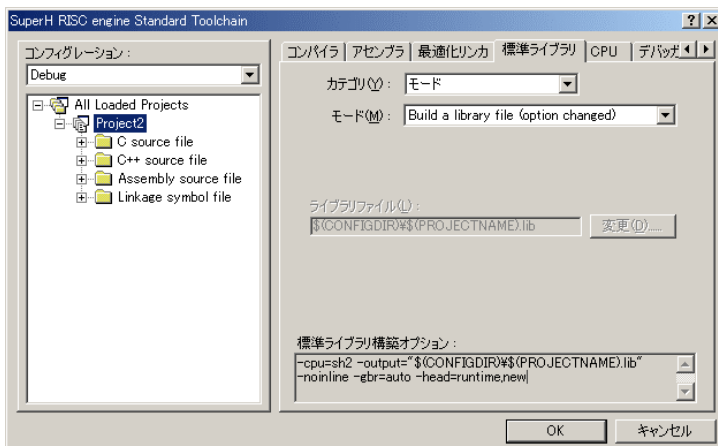
HEWは、プロジェクトごとにC言語用ライブラリを作成する仕様になっています。
ライブラリを作成および設定の確認をします。



[オプション]-[Hitach SuperH RISC engine Standard Toolchain]をクリックします。



「標準ライブラリ」タグをクリックする。

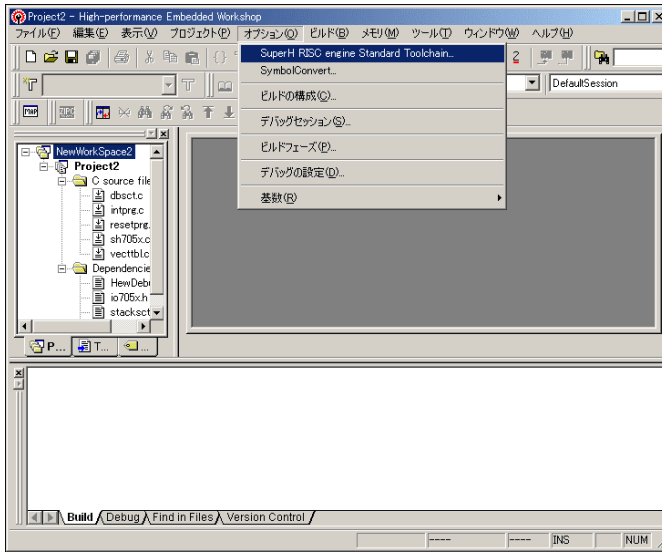


カテゴリのモードが「Build a Library file(option changed)」指定になっている事を確認します。

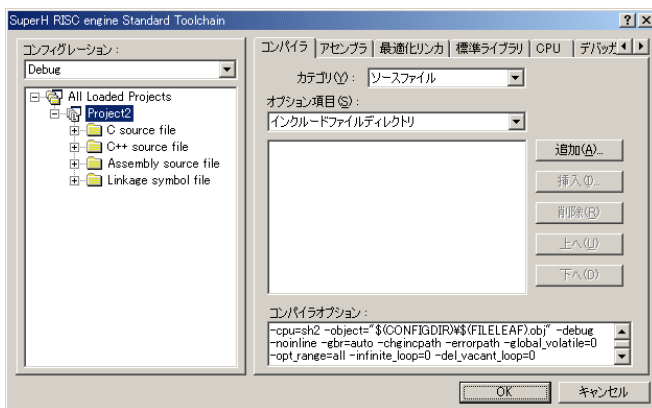
この指定によりオプション変更時のみライブラリを作成する事になります。

OKをクリックする。

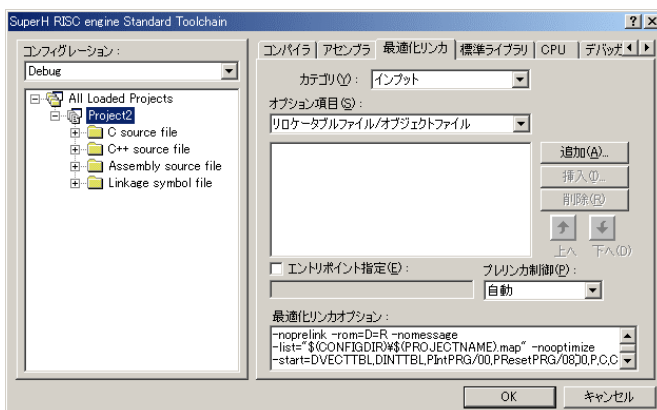
5 . ツール(リンカ)の設定



[オプション]-[Hitach SuperH RISC engine Standard Toolchain]をクリックします。



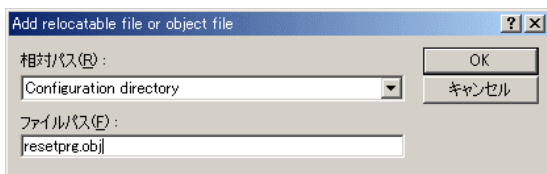
「最適化リンカ」タグをクリックする。



カテゴリの「インプット」を選択する。

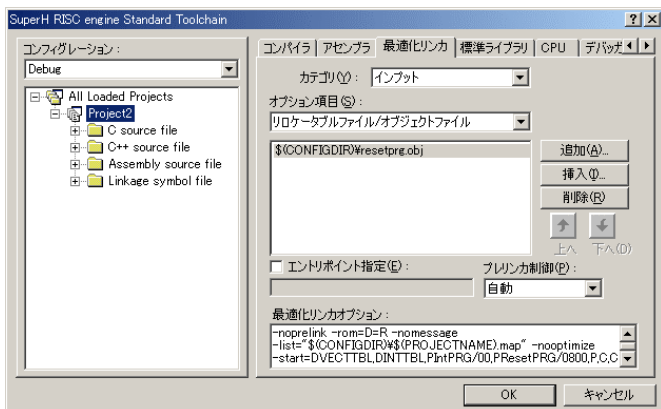
オプション項目の「リロケートブルファイル/オブジェクトファイル」を選択する。

追加をクリックする。



相対パス「Configuration directory」に選択します。
ファイルパス「resetprg.obj」と入力します。

OKをクリックします。

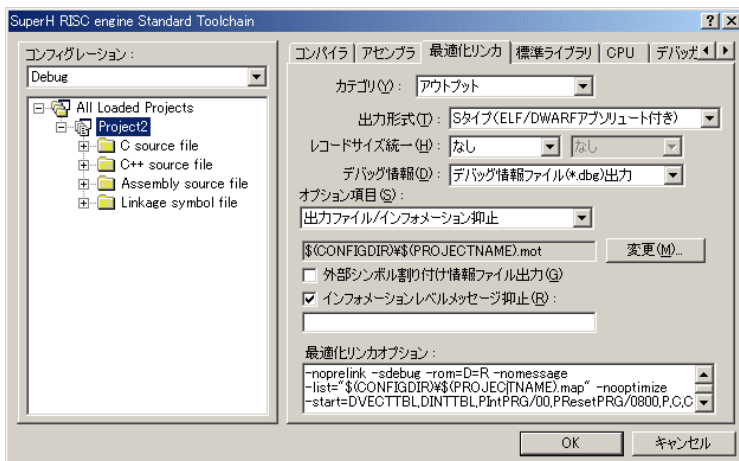


ファイルがセットされたのを確認します。

この指定は、“最初にこのリスト順にモジュールをリンクしなさい”との指示になります。

(DEFのC Viewにてアドレス順に他モジュールを表示させたい場合は、追加設定してください。)

(重要) 特に「resetprg.obj」の先頭に **SP** を設定するコードがありますので先頭を 0×1000 番地にする為の設定ですが、後説明のセクション指定で「PRResetPRG」の先頭番地を 0×1000 番地にしていますので必要ありませんが、同セクション名の他モジュールを追加した場合でも、必ず最初にリンクさせるための手続きです。



カテゴリの「アウトプット」を選択する。

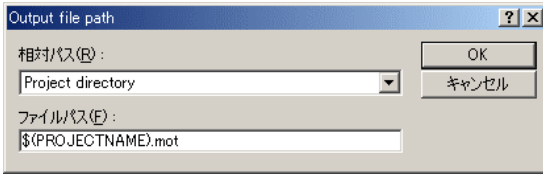
出力形式の「Sタイプ (ELF/DWARFアブソリュート付き)」を選択する。

デバッグ情報の「デバッグ情報ファイル (*.dbg)出力」を選択する。

「出力ファイル/インフォメーション抑止」の

変更をクリックします。

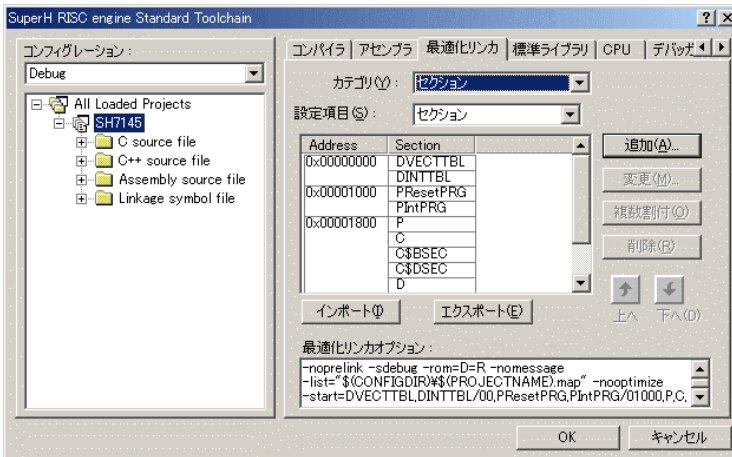
(重要) シンボリックデバッグを可能にするために必要な設定です。



相対パス「Project directory」に設定します。

OKをクリックする。

(重要) この指定は、HEXファイルをCソースファイルのある同じディレクトリに置くための設定です。絶対条件として、「*.mot/*.sym/*.lin」は、同じ場所に置く必要があります。
Hc symconvで出力ファイルを「Configuration directory」にした場合は、上記の指定も「Configuration directory」にして下さい。今回の使用例は、「Project directory」になっています。



カテゴリの「セクション」を選択する。

下図のようにセクション指定をする。

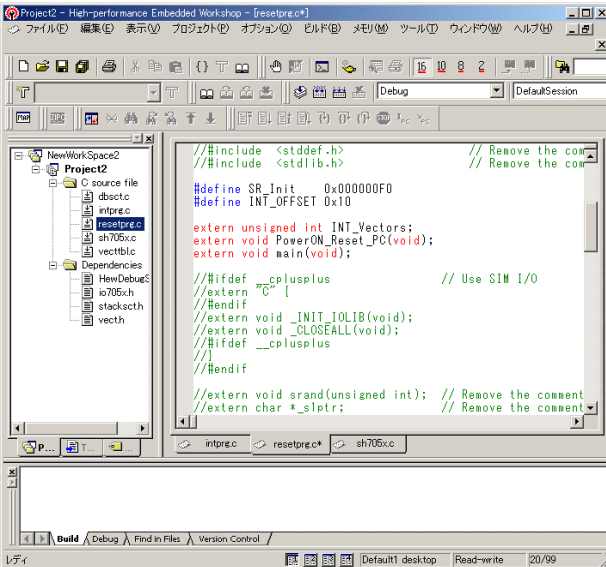
0x00000000	DVECTTBL DINTTBL
0x00001000	PResetPRG PIntPRG
0x00001800	P C C\$DSEC C\$BSEC D
0xFFFFE840	B R
0xFFFFFBFC	S

OKをクリックする。

(重要) 「Bセクション」0xFFFFE840 (0xFFFFE800) は、ソースブレイク使用時の例です。
DEFバージョン6.50xから、ソースブレイクを使用する場合は、モニターワーク方式をスタック方式に選択する必要があります。

6. スタートアップおよびベクターの変更と追加

1) <resetprg.c>の変更

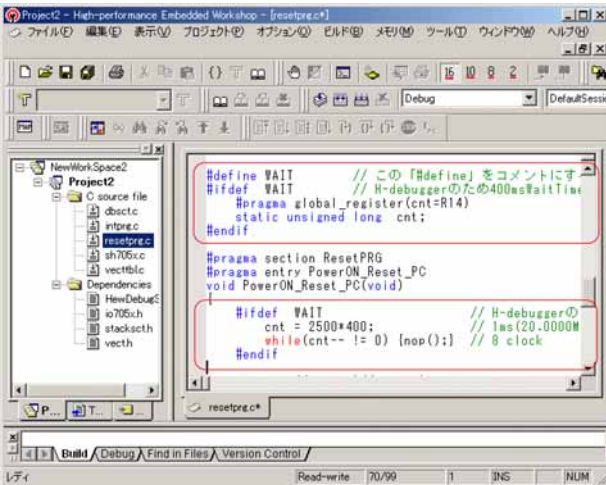


①SR_Init 値を「0x00000F0」で変更しません。

(備考)

「resetprg.c」関数内でブレークポイント及びトレース/ステップ実行をしたい場合は、DEF 操作のショートPB「E I」をクリックして下さい。

この例での割り込み許可は、「_main()」関数内に記述してあります。

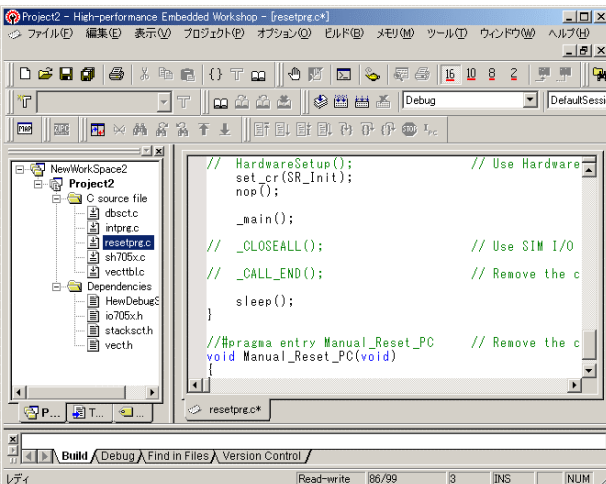


①左図のようなソースを追加しますとリセット時ソフトタイマ挿入になります。このソースを追加することにより下記メリットがあります。

②ソフトタイマが不必要な場合は「#define WAIT」をコメントにして下さい。

③ソフトタイマによるメリット

- DEFの「RstMon」操作時SP値が設定値のままである。
- DEFの「RstMon」操作時「main()」プログラムまで走行しないため初期設定等による内部レジスタが汚れない。



①「main()」を「_main()」に変更特に意味はありません。SH-2のサンプルは全て「_main()」になっているので共通化する為です。

2) ブレークおよびトレース/ステップ実行について

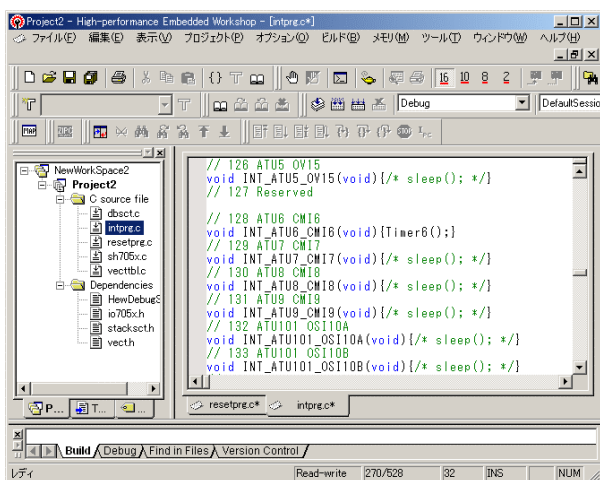
UBCブレーク割り込みのプライオリティは、15になっていますのでCPUのSRレジスタの割り込みマスクレベルを14以下に設定する必要がありますので、下記方法のどちらかを都合に応じて実施して下さい。

なお、トレース/ステップはUBCを利用したソフト判定での実施となります。

【方法①】 「set_imask(14)」を記述する。

【方法②】 DEF操作でのショートPB「DI」をクリックする。

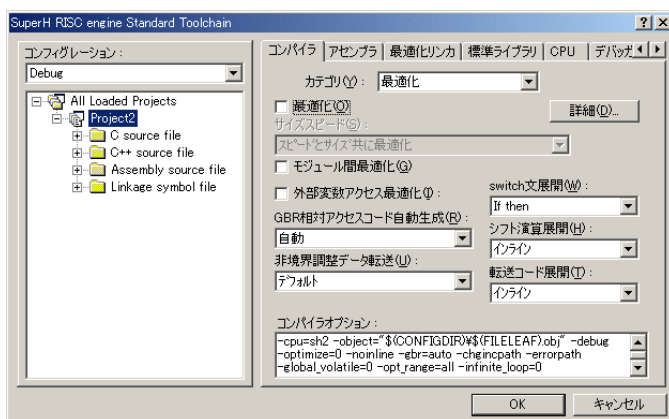
3) <intprg.c>の変更



今回使用の「SH705x.C」は、ATU6のCMI6割り込みを使用した例ですのでベクターを設定します。

①「vector 128」に「Timer6();」関数を登録します。

4) コンパイラの「最適化」を外す



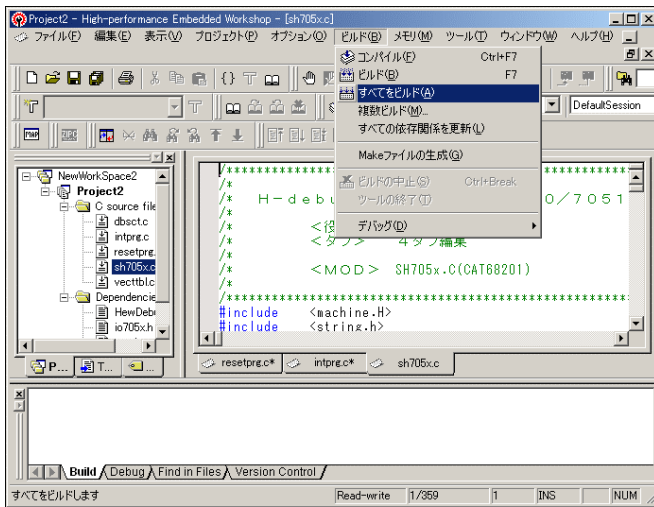
①Toolchainの「コンパイラ」を選択

②カテゴリ「最適化」を選択

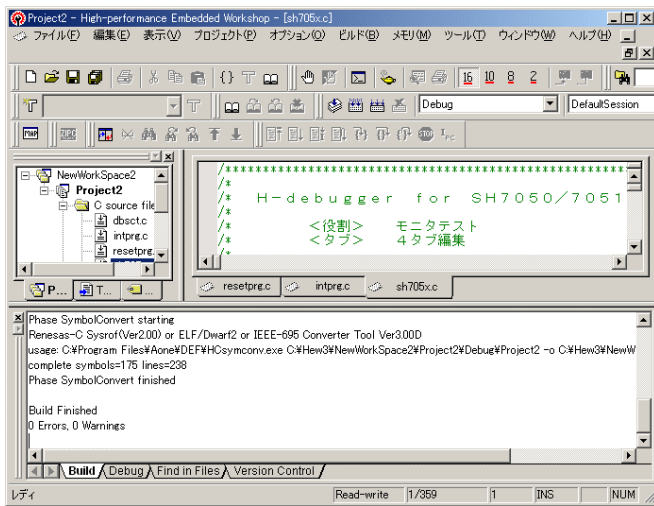
③「最適化」のチェックを外す。

④OKをクリックする。

7. ビルドの実行

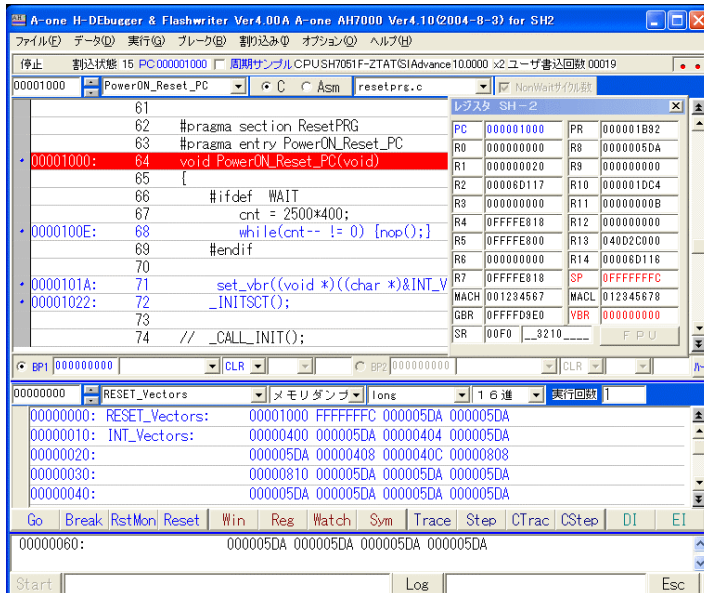


[ビルド] - [すべてをビルド]をクリックします。

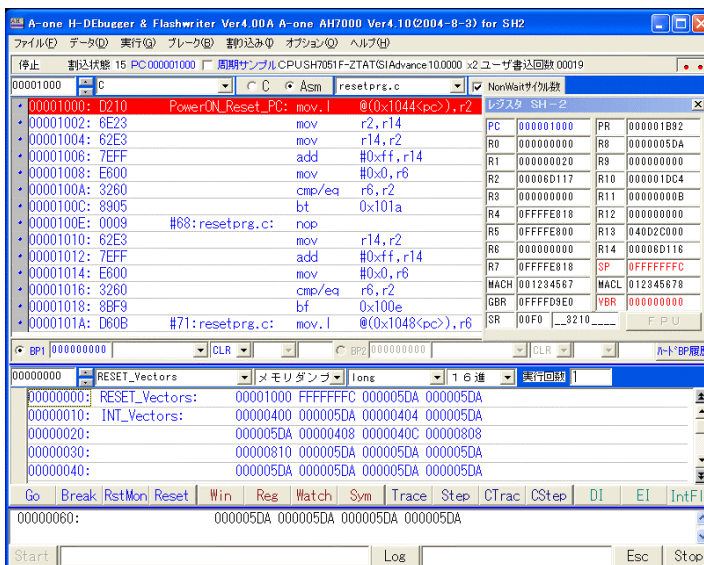


「0 Error 0 Warnings」になり作業終了です。

8. DEFでの確認



- ① 0番地のリセットベクター値「0x1000」が確認できます。
- ② 4番地のスタックポインタ値「0xFFFFF0FC」が確認できます。
- ③ レジスタ画面を確認して下さい。



- ① ソフトタイマの逆アセンブラ表示です。(参考まで)
- ② SR 値および SP 値がリセット状態であることが確認できます。

これで「H-Debugger」用に設定ができたのが確認できました。

以上