


統合環境 Hew4(Ver4.09.00.007)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

CAT専用仕様【CAT321-SH7267-NOCACHE版】

ルネサスC言語用統合環境「Hew4 Ver4.09.00.007」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	ReneCAT321		
プロジェクト名	Project		
登録モジュール名	appinfo.c sh7267h.c timer.c	Cファイル	ブートモード1/3用アプリ情報用 メインモジュール (アプリ用) タイマー管理
Hew添付ファイル	resetprg.c intrprg.c dbsct.c vecttbl.c iodefine.h vect.h stacksct.h tyoedef.h	Cファイル ヘッダファイル	スタートアップモジュール 割込みベクターモジュール 定数転送用セクション管理宣言 割込みベクターテーブル I/O定義ビットフィールド記述用 割込みベクターテーブル定義 スタックサイズ定義 変数型定義の使用例
CPUタイプ	SH7267		
備考	キャッシュ無効のサンプル		

 【ポイント】

「SH-2A H-UDI版」の場合、Hew4 デフォルト設定から変更に必要な箇所は、

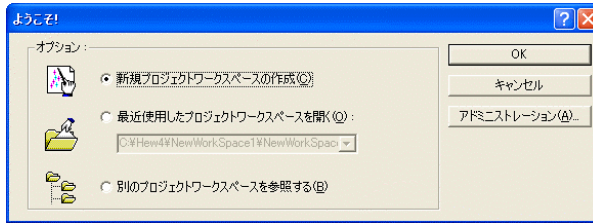
- 1) コンパイラの「最適化」を外す。
- 2) セクションアドレスを変更する。

の2点になります。

1. 新ワークスペースの登録方法

“Hew4” 起動させます。

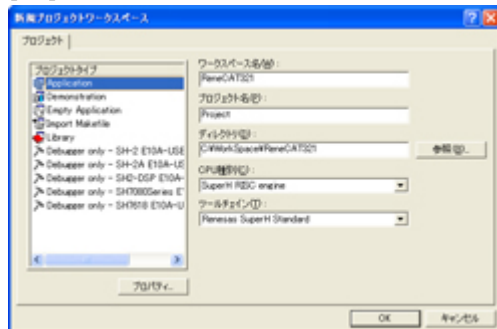
[1-1]



“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。

[1-2]



プロジェクトタイプ: Application

ワークスペース名 “ReneCAT321”

プロジェクト名 “Project”

デフォルト “C:\Workspace”

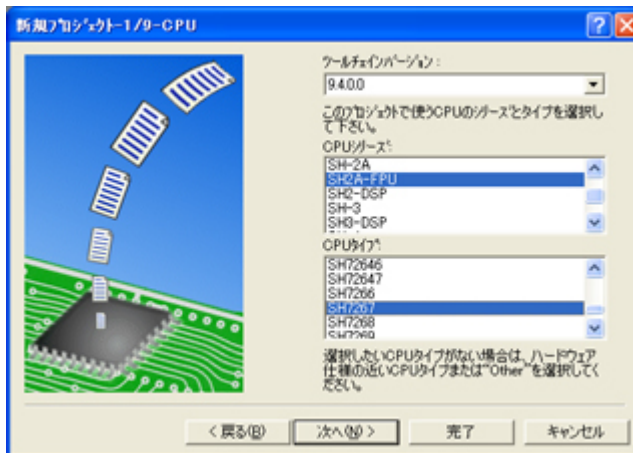
CPU種別 “SuperH RISC engine”

ツールチェーン “Renesas 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。

[1-3]



CPUシリーズを「SH2A-FPU」に選択する。
CPUタイプを「SH7267」を選択する。
確認後、

次へ>をクリックします。

[1-4]



CPUスペックを確認後

次へ>をクリックします。

[1-5]



本サンプルでは、

①I/O ライブラリを使用しませんのでチェックを外して下さい。

②ヒープメモリを使用しませんのでチェックを外して下さい。

main() 関数生成は「None」に選択する。

③I/O レジスタ定義ファイルは使用しますのでチェックして下さい。

ハードウェアセットアップ関数生成は「None」に選択する。

確認後、**次へ**をクリックします。

[1-6]



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。

次へをクリックします。

[1-7]



①スタックポイント「HFFF9_0000」

②スタックサイズ「H400」

次へをクリックします。

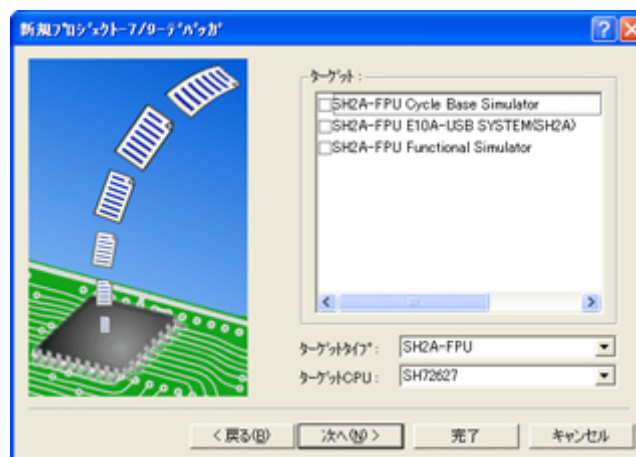
[1-8]



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、

次へ>をクリックします。

[1-9]



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>をクリックします。

[1-10]



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ます。

[1-11]



確認画面が表示されますので、

OKをクリックします。

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

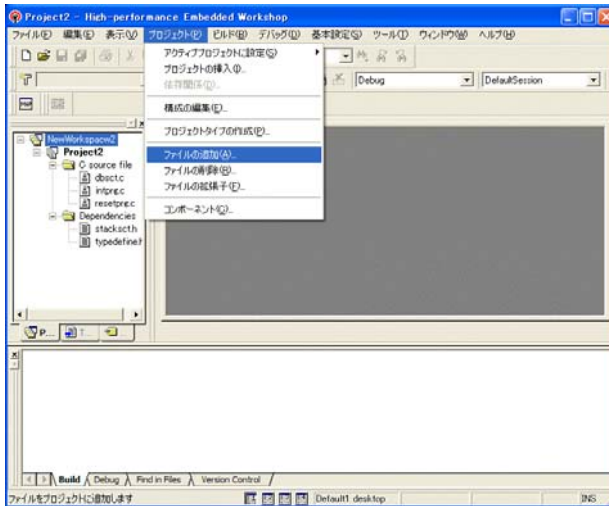
準備： HP よりダウンロードします。（ルネサスC）
<http://www.aone.co.jp/tools/AH7000/renesas/#CAT>

ReneCAT321_NOCACHE_100.LZH

作成済みの 5 ファイルを”C:\¥WorkSpace¥ReneCAT321¥Project”にコピーします。

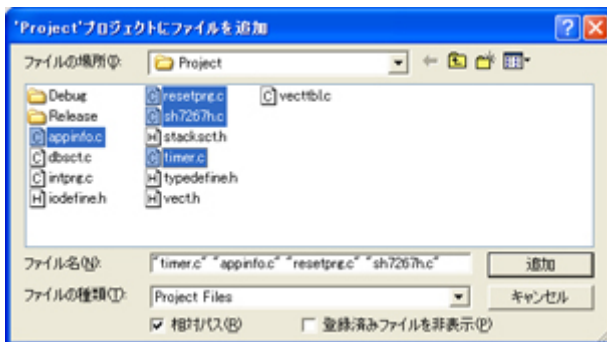
- | | |
|---------------|----------------------------|
| ①appinfo.c | アプリケーション情報定義 |
| ②sh7267h.c | メイン |
| ③timer.c | タイマー管理 |
| ④resetprg.c | CAT321 用 BSC 設定+Point!!の追加 |
| ⑤typedefine.h | 変数型定義 |

[2-1]



[プロジェクト]-
[ファイルの追加]をクリックします。

[2-2]



下記 4 ファイルを指定して下さい。

- ①appinfo.c
 - ②sh7267h.c
 - ③timer.c
 - ④resetprg.c
- 選択後、

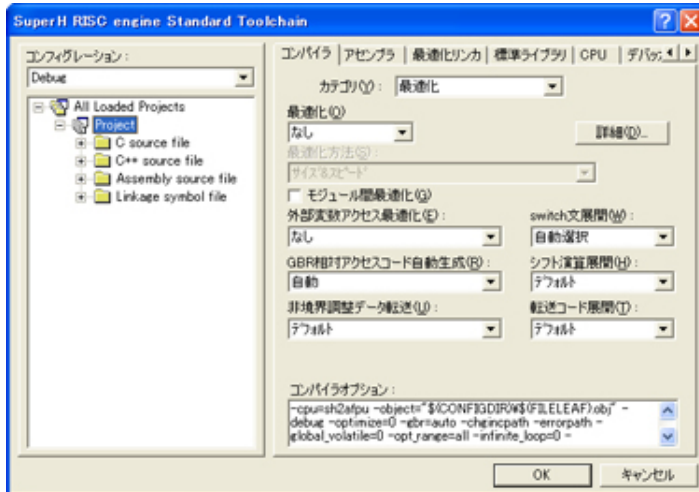
追加をクリックします。

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す。

3. コンパイラの設定

コンパイラの「最適化」を外す

[3-1]



①[ビルド] -
[SuperH engine Standard Toolchain]
をクリックします。

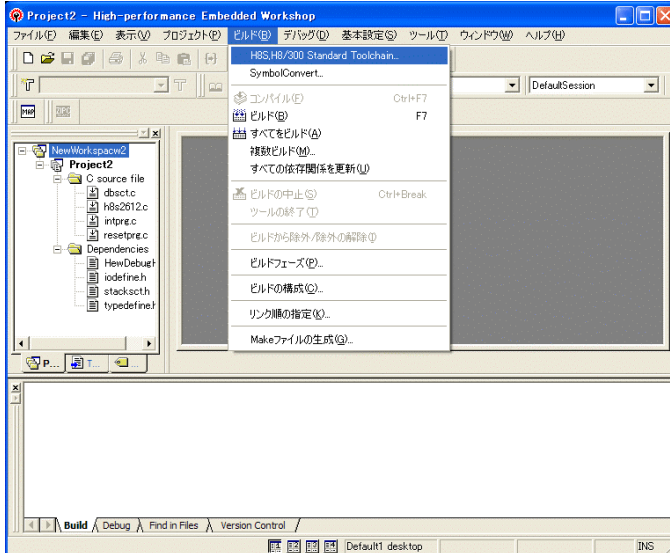
- ②「コンパイラ」を選択
- ③カテゴリ「最適化」を選択
- ④「最適化」「なし」を選択

⑤OKをクリックします。

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。

[4-1]



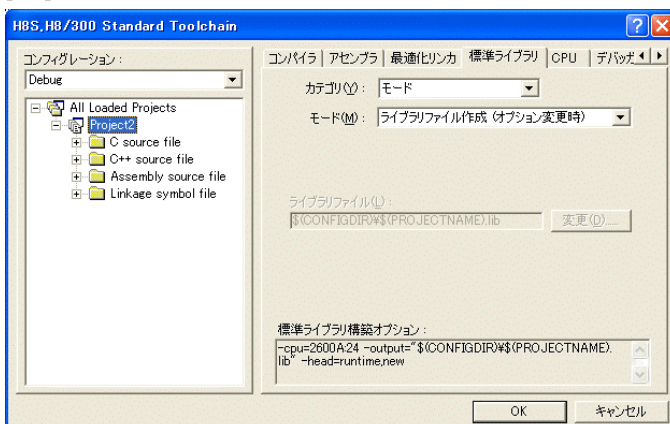
[ビルド]-
[SuperH engine Standard Toolchain]
をクリックします。

[4-2]



[標準ライブラリ]タグをクリックし
ます。

[4-3]



カテゴリのモードが
「ライブラリファイル作成 (オプション変更時)」指定になっている事
を確認します。(Default)

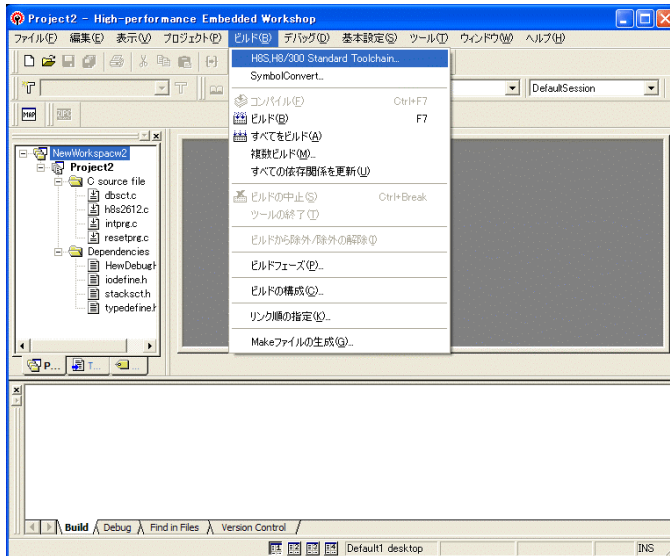
この指定によりオプション変更時の
みライブラリを作成する事になります。

デフォルトのまま

OKをクリックします。

5. ツール (リンカ) の設定

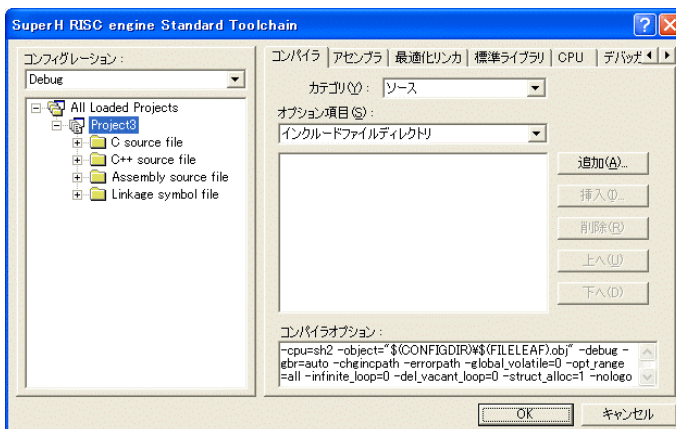
[5-1]



[ビルド]-

[SuperH engine Standard Toolchain]
をクリックします。

[5-2]



「最適化リンカ」タグをクリックし
ます。

[5-3]

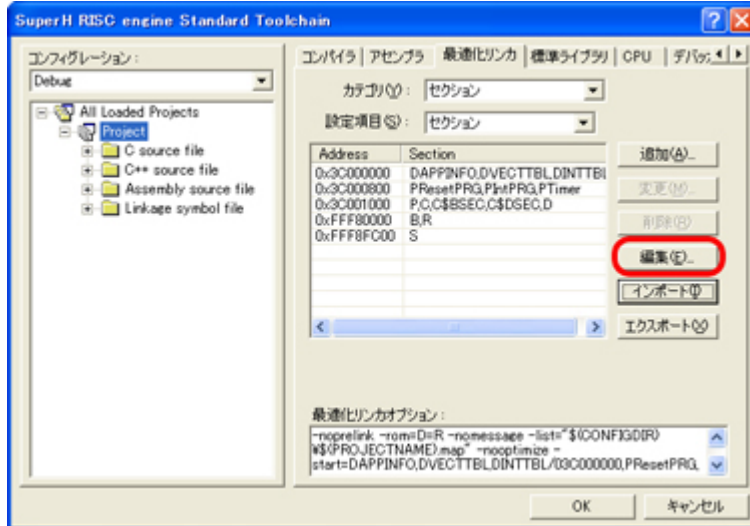


- ①カテゴリの「出力」を選択する。
- ②出力形式の「Sタイプ (ELF/DWARF アブソリュート付き)」を確認する。
- ③デバッグ情報の「出力 (出力ファイル内)」を確認する。
- ④ オプション項目の「出力ファイル」
\${CONFIGDIR}¥\$(PROJECTNAME).mot
を確認する。

全てデフォルトです。

[5-4]

セクションアドレスを変更する。



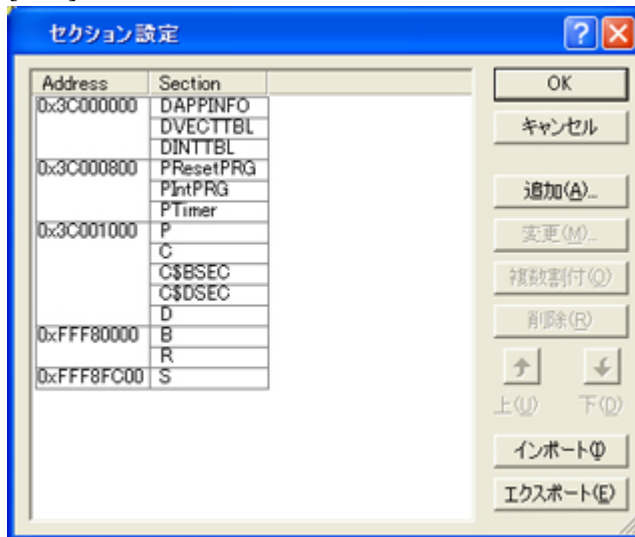
カテゴリの「セクション」を選択する。

① **編集** をクリックします。

② 変更後、

OK をクリックします。

[5-4-1]



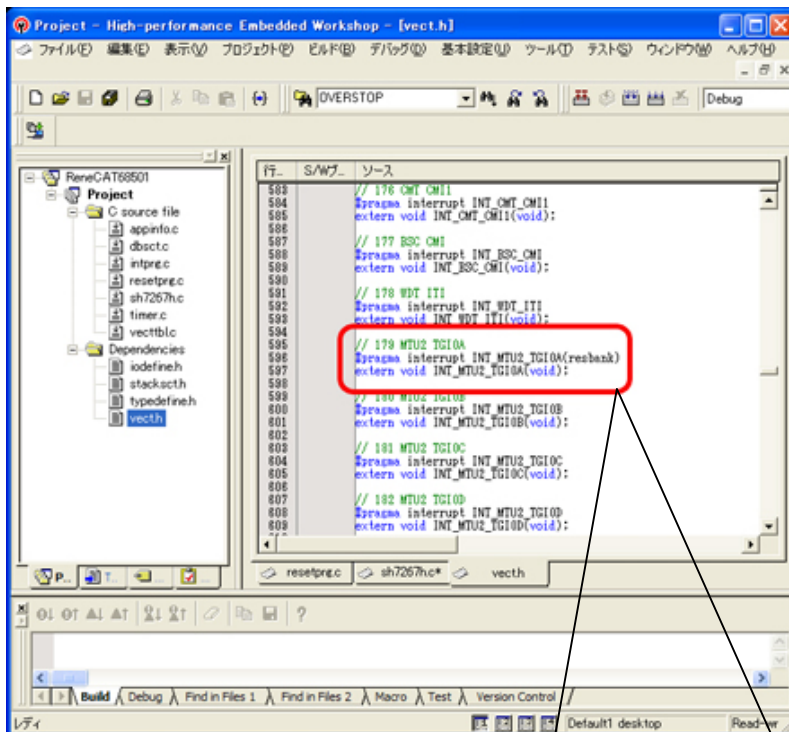
0x3C00_0000	DAPPINFO,DVECTTBL,DINTTBL
0x3C00_0800	PResetPRG,PIntPRG,PTimer
0x3C00_1000	P,C,C\$DSEC,C\$BSEC,D
0xFFFF_0000	B,R
0xFFFF_FC00	S

セクション名 **DAPPINFO** は、必ずトップに設定して下さい。

6. ベクター等の変更

1) vect.h の変更

[6-1]



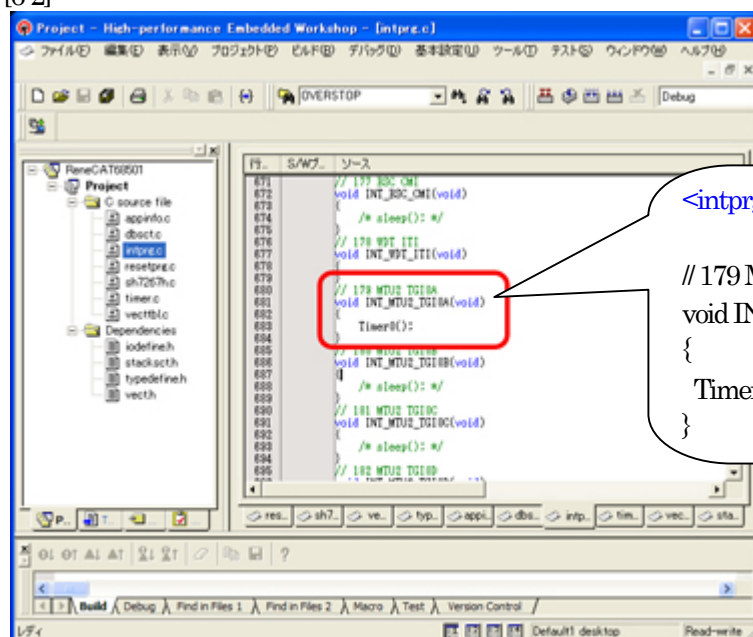
本サンプルでは、「vector 179」の MTU2 TGIOA の割込みをバンクレジスタモードにて使用しますので「vect.h」にバンクレジスタを使用するための定義をします。

```
// 179 MTU2 TGIOA
#pragma interrupt INT_MTU2_TGIOA(resbank)
extern void INT_MTU2_TGIOA(void);

#pragma interrupt INT_MTU2_TGIOA(resbank) と定義します。
```

2) intprg.c の変更

[6-2]



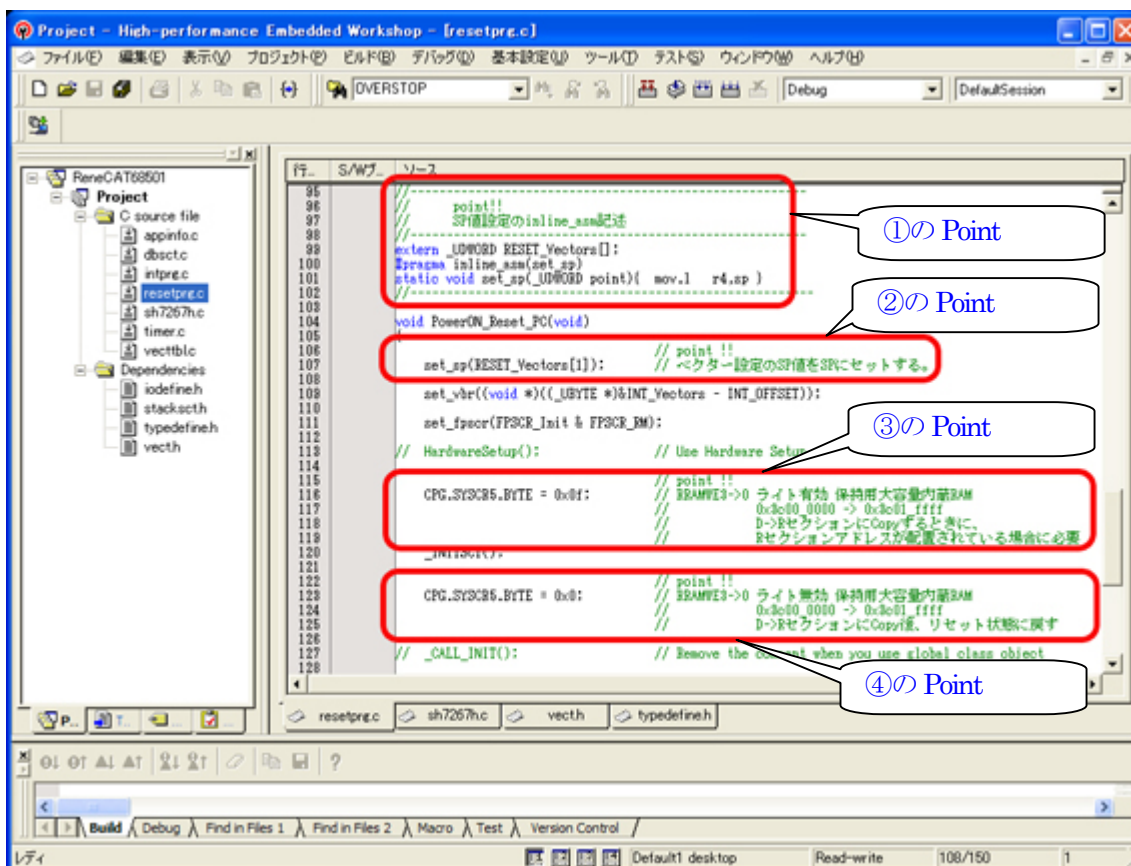
①vector 179 に「Timer0;」関数名を登録します。

```
<intprg.c>

// 179 MTU2 TGIOA
void INT_MTU2_TGIOA(void)
{
    Timer0;
}
```

3) resetprg.c の説明

[6-3]



①の Point

ベクターテーブルに登録されたスタックポイント値をスタックにセットする inline_asm 関数

②の Point

ブートモード1/3の場合、ローダプログラムがスタックポイント値を設定してからこの関数に飛ばす仕様になっていますが、アプリケーションソフトの開発中はローダプログラムとは切り離されていますので、リセット関数のトップでスタックポインタの設定をしておきます。

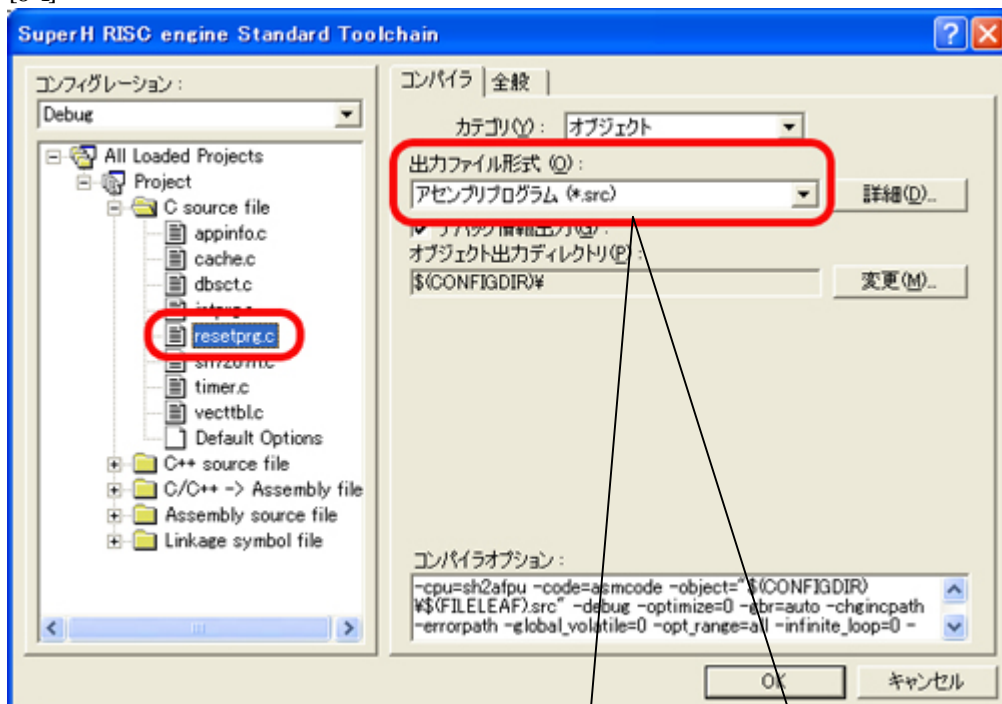
③の Point

SH7267の場合、保持用大容量内蔵RAM(0x3c00_0000->0x3c01_ffff)エリアがデフォルトで書き込み禁止になっています。D->RセクションにCopyするときに、Rセクションアドレスが配置が範囲に入っている場合は、この位置(関数_INTTSCT();の前)で書き込みを許可します。

④の Point

③の処置をした場合は、この位置(関数_INTTSCT();の後)でデフォルトに戻す為、書き込み禁止にします。

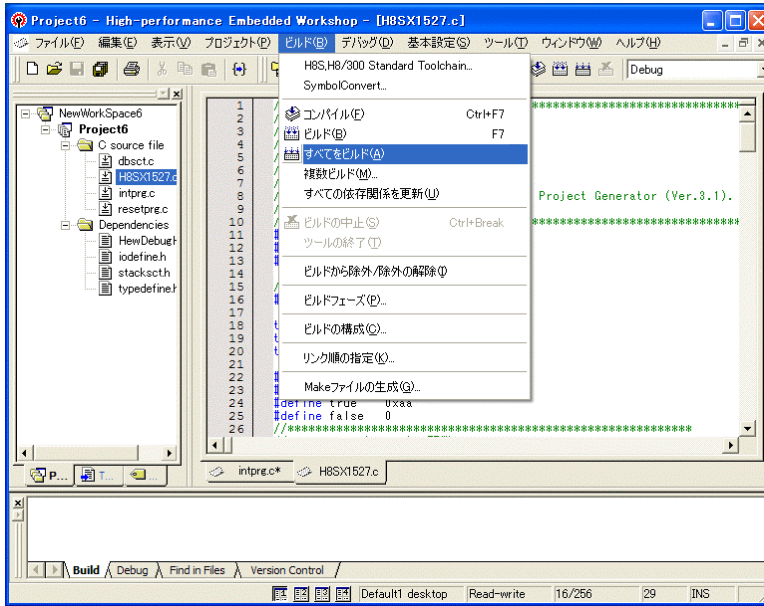
[6-4]



resetprg.c は、inline_asm 記述をしていますので、<コンパイラ>、
カテゴリ<オブジェクト>、出力ファイル形式を
「アセンブリプログラム(*.src)」に指定する必要があります。

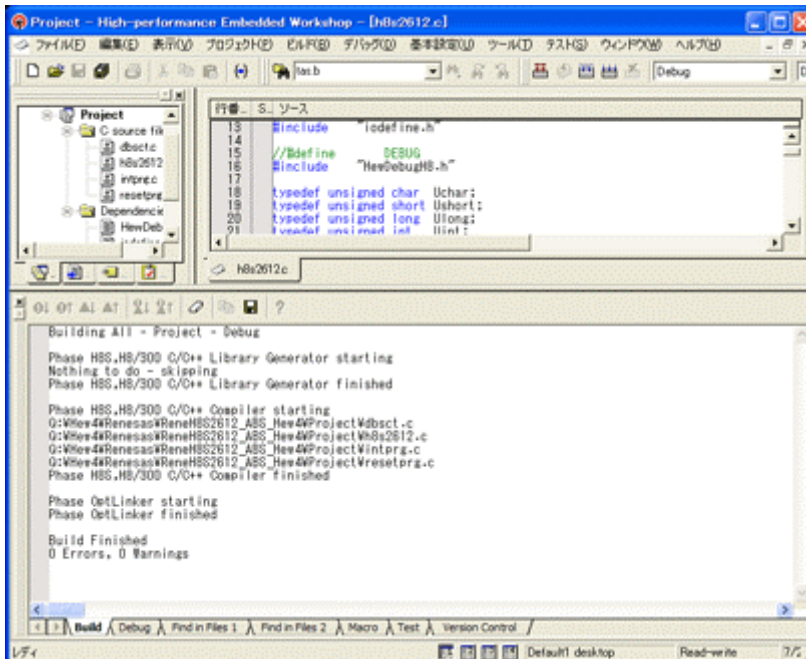
7. ビルドの実行

[7-1]



[ビルド] -
[すべてをビルド]をクリック
します。

[7-2]

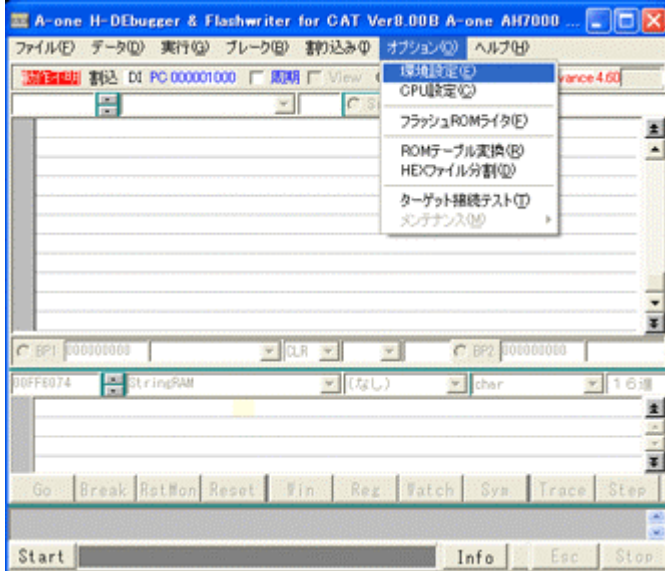


「0 Error 0 Warnings」
になり作業終了です。

8. DEFでデバッグの準備をする

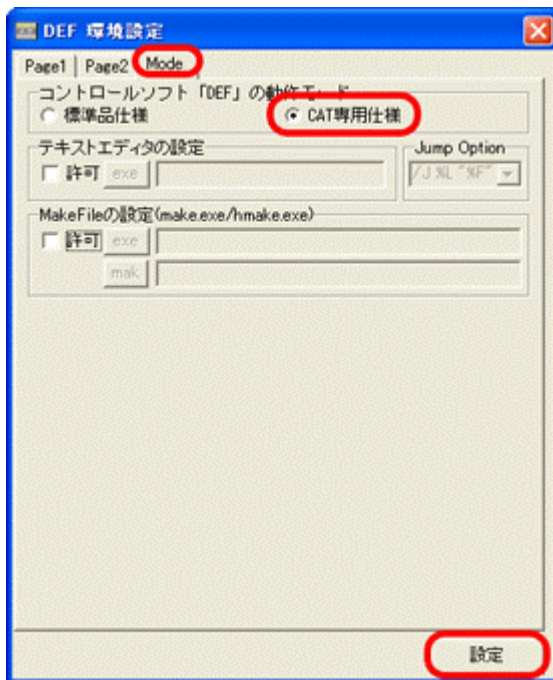
1) DEFをCAT専用仕様モードにします。

[8-1]



<オプション> - <環境設定>
をクリックします。

[8-2]



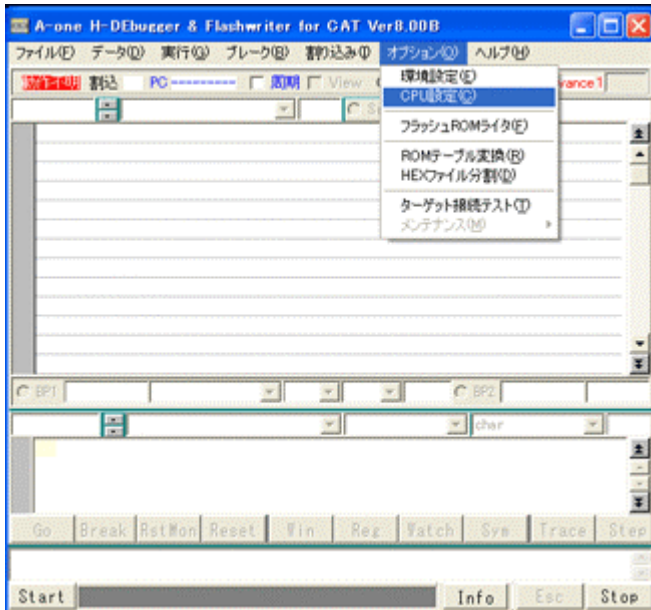
<Mode>を選択する。

「CAT専用仕様」をチェックします。

「設定」をクリックします。

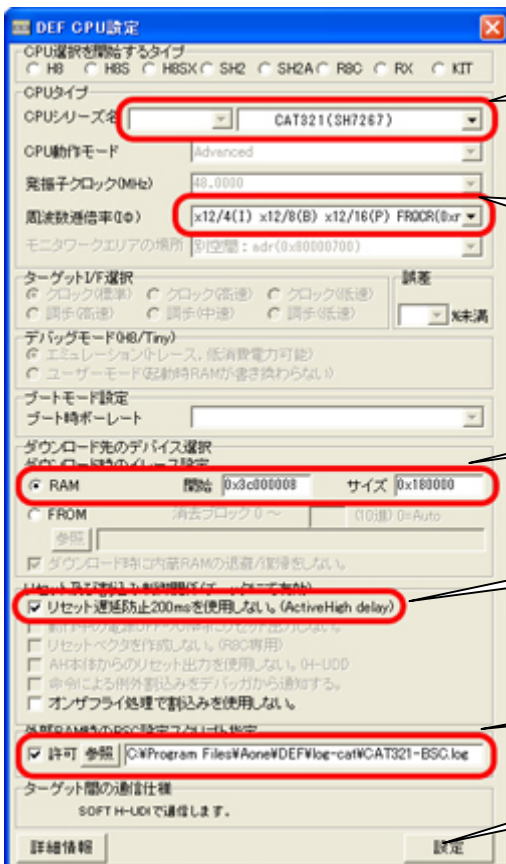
2) DEFのCPU設定をします。

[8-3]



<オプション>-<CPU設定>
をクリックします。

[8-4]



CPU シリーズ名
「CAT321(SH7267)」を選択します。

「x12/4(1) x12/8(B) x12/16(p) FRQCR(0xn103)」
を選択します。

RAM
開始 0x3C00_0008 サイズ 0x180000
を確認します。

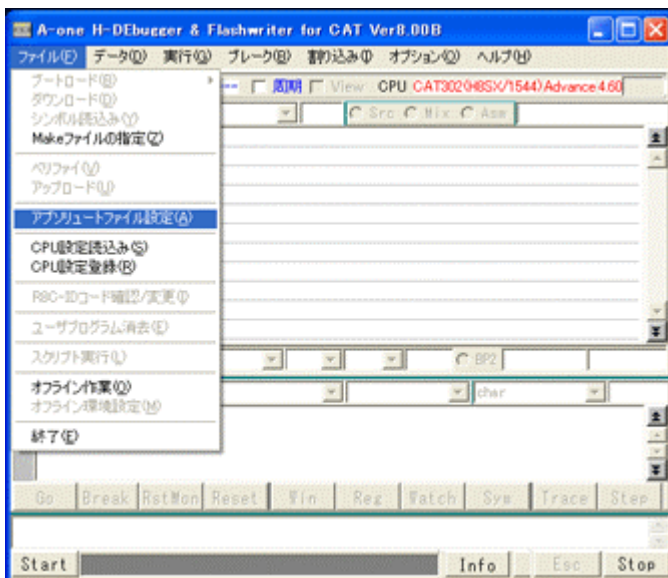
「リセット遅延防止..... 使用しない。」
をチェック します。

「..\Aone\DEF\log-cat\CAT321-BSC.log」
を確認します。

「設定」をクリックします。

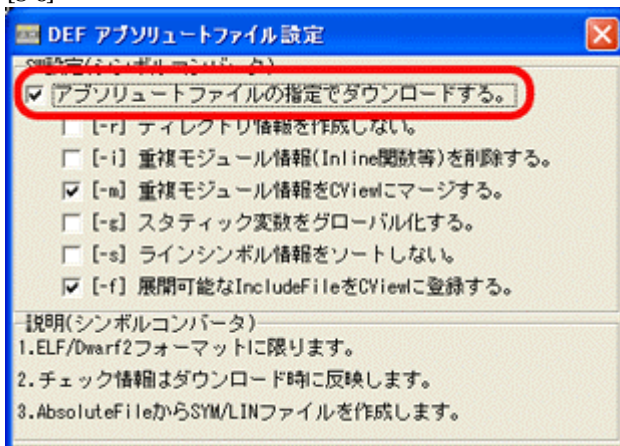
3) 転送をアブソリュート指定方式にします。

[8-5]



<ファイル>—
<アブソリュートファイル設定>
をクリックします。

[8-6]



「アブソリュートファイルの指定でダウンロードする。」に「チェック 」を入れます。

Hew4 デフォルト設定の状態で使用する場合、
[r]をチェックしないで下さい。
アブソリュートファイルから直接
「*.SYM*.LIN」ファイルを作成します。
この設定は記憶します。

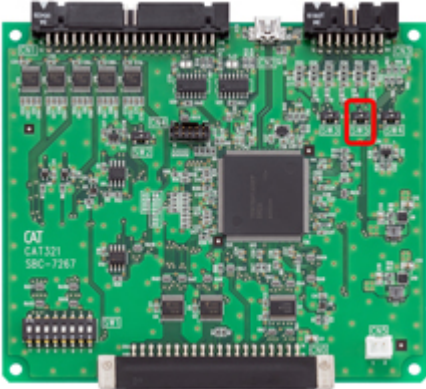
[X] をクリックします。

その他スイッチに関する説明は、

<http://www.aone.co.jp/tools/AH7000/renesas/index.html> に登録してある

「シンボルコンバータのスイッチ説明と登録方法の説明(SymConvHEW.pdf)」資料を参照して下さい。

4) CAT321 のジャンパ設定を確認します。

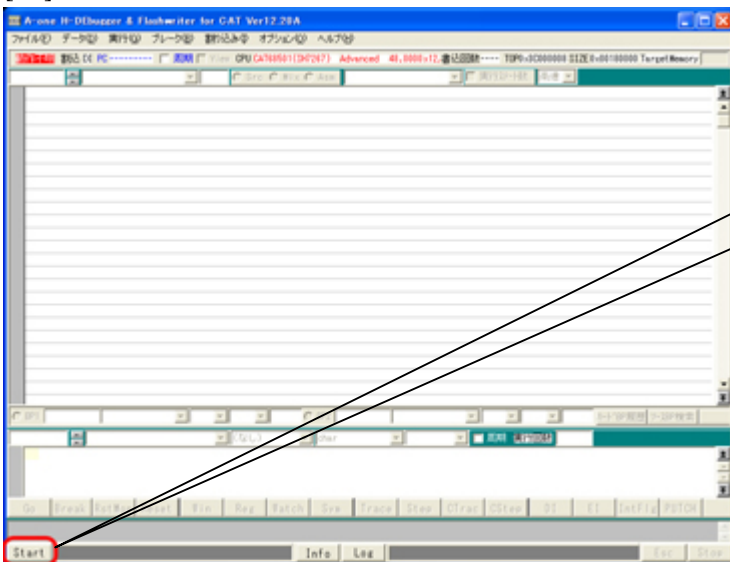


SW5 を「B側」のデバッグモードにして下さい。

デバッグ時の SW 設定【CAT321】○側の設定

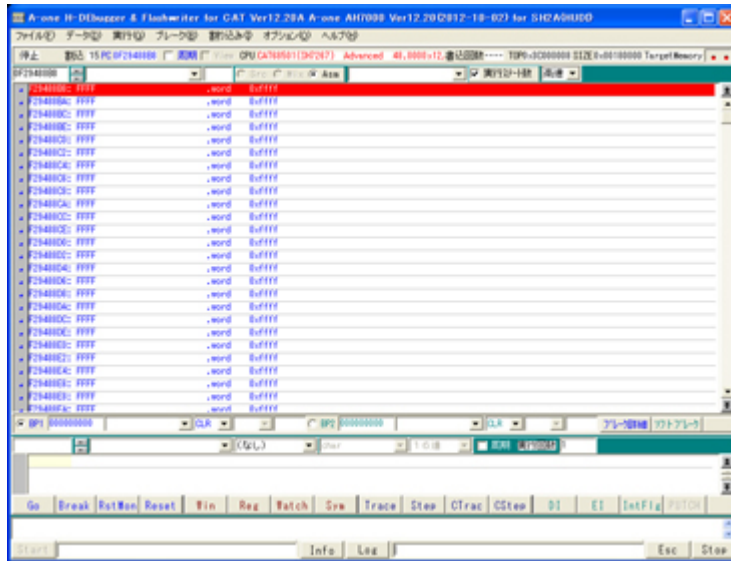
	A	B
SW3	○通常動作	設定禁止
SW4	○低速ブート	高速ブート
SW5	通常動作	○デバッグモード

5) CAT321 とデバッガ「AH7000」を接続します。【CAT321 側を電源 ON にして下さい】
[8-7]



DEF 画面の左下の「Start」をクリックします。

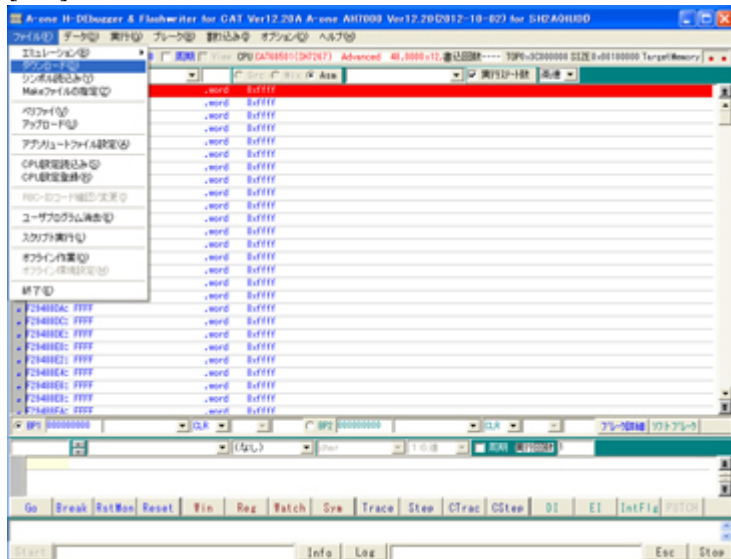
[8-10]



エミュレーションモード遷移が成功しますと、このような画面になります。

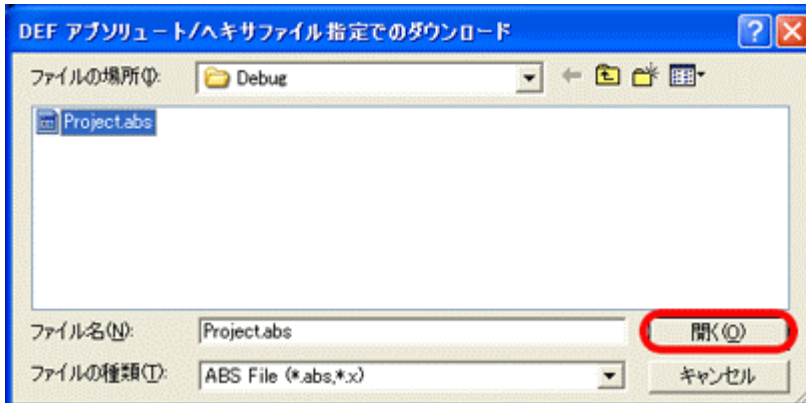
6) サンプルソフトをダウンロードします。

[8-11]



<ファイル>-
<ダウンロード>をクリックします。

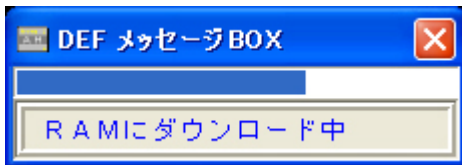
[8-12]



サンプル
「Project.abs」を指定します。

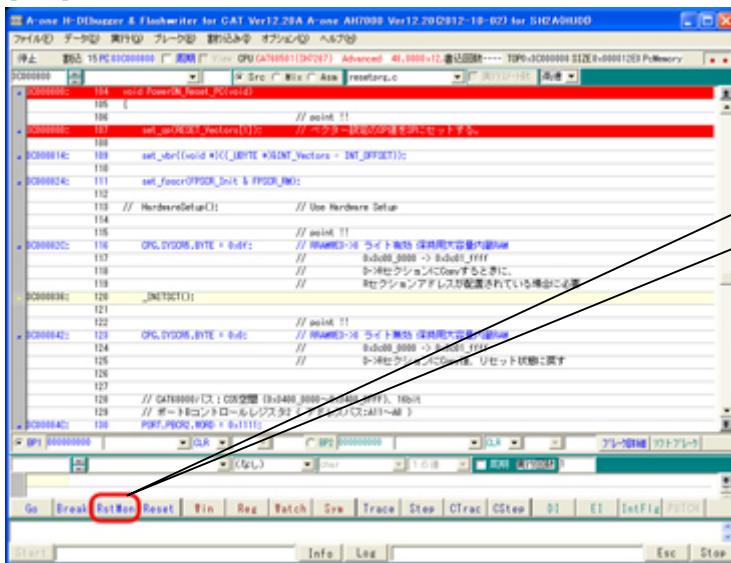
「開く」をクリックします。

[8-13]



ダウンロード中は、このようなインジケータを表示します。

[8-14]



ダウンロードが成功しますと、この表示になります。

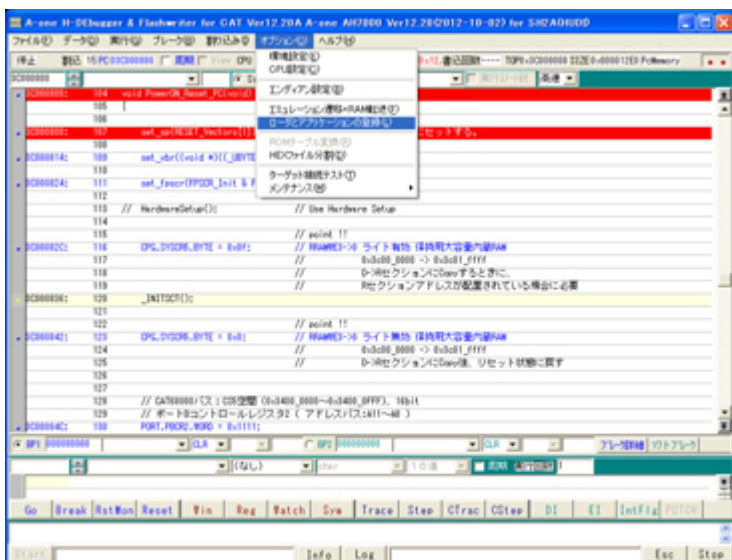
DEFのShortPB
「RstMon」をクリックします。

0x3C00_0800番地に
「resetprg.c」の
「PowerON_Reset」
関数になっていることが確認できます。

ここまでの、CAT 専用仕様における CAT321 の新規プロジェクト方法とデバッグを開始するまでの説明です。

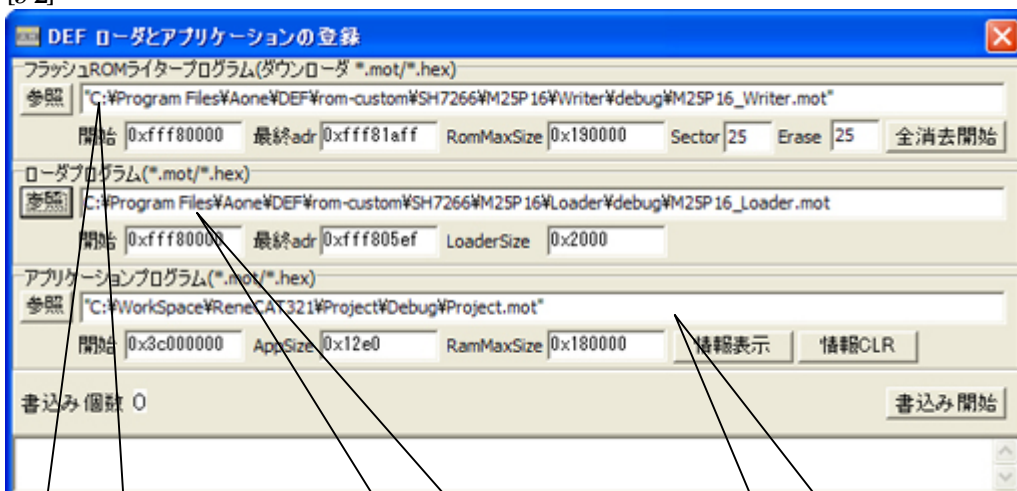
9. デバッグ終了後、アプリケーションソフトをシリアルフラッシュROMに登録します。

[9-1]



<オプション>—
<ローダとアプリケーションの登録>をクリックします。

[9-2]

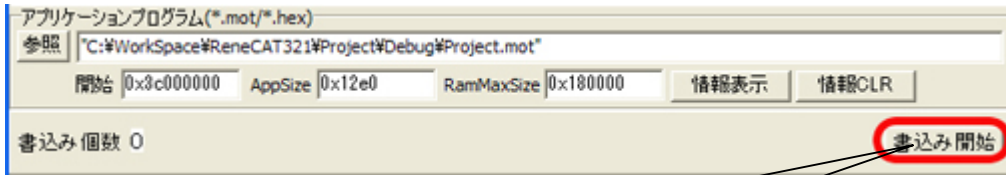


<確認 1>
デフォルトにてライタープログラム
「M25P16_Writer.mot」
が登録されていますので確認します。

<確認 2>
デフォルトにてローダプログラム
「M25P16_Loader.mot」
が登録されていますので確認します。

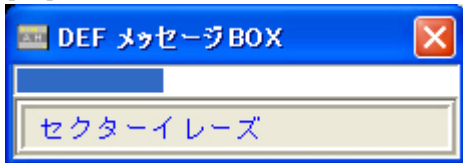
<操作 1>
「参照」をクリックしてシリアルフラッシュROMに登録するアプリケーションプログラム
「Project.mot」を指定します。

[9-3]



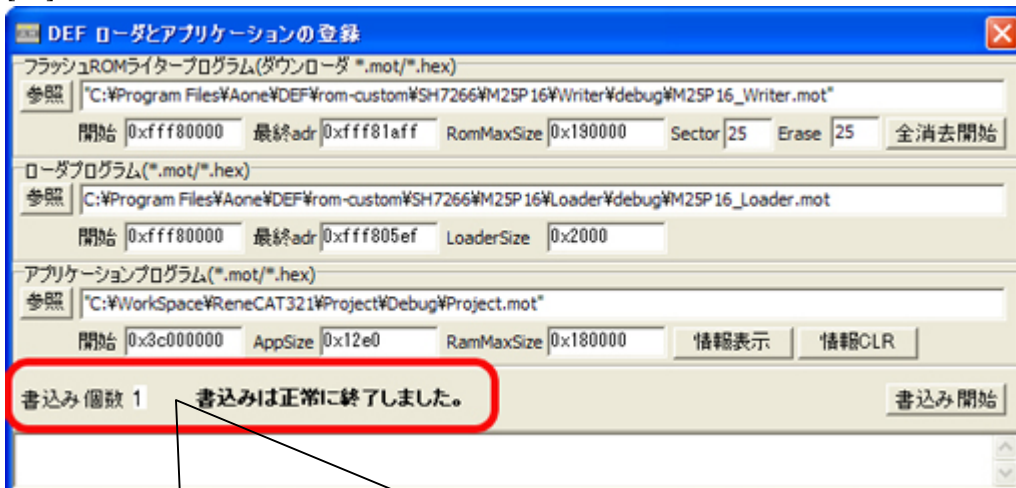
<操作2>
 「書き込み開始」をクリックします。

[9-4]



書き込み中は、この様なインジケータを表示します。

[9-5]



<確認>
 書き込みが正常終了しますと、このメッセージが表示され個数が+1 されます。
 CAT321 基板上の SW 設定を通常動作に変更する。

通常時の SW 設定 【CAT321】 ○側の設定

	A	B
SW3	○通常動作	設定禁止
SW4	○低速ブート	高速ブート
SW5	○通常動作	デバッグモード

ローダとアプリケーションの登録方法に関する詳細説明は、
<http://www.aone.co.jp/tools/AH7000/renesas/index.html> に登録してある
 「ブートモード1/3対応のアプリケーション作成方法(SfApplication.pdf)」資料を参照して下さい。

以上にて本説明を終了します。