

リリースノート

# High-performance Embedded Workshop V.4.08

RJJ10J2737-0100 Rev.1.00 2010.07.01

弊社製品をご使用いただき厚く御礼申し上げます。High-performance Embedded Workshop を使用するにあたり注意事項がございます。ご留意いただけますようお願い申し上げます。

## 目次

1.	注意事項	2
1.1	ネットワークドライブ	
1.2	エディタのシンタックスカラーリング機能	
1.3	ファイル依存関係	3
1.4	ナビゲーション機能	
1.5	式の指定	
1.6	エラーが発生したファイルを開く	
1.7	オプションダイアログボックス上でのディレクトリパス指定	7
1.8	ワークスペースの作成言語	7
1.9	日本語版の表示言語	
1.10	エディタウィンドウの文字セット	7
	ツールアドミニストレーション	
	デバッグ機能(共通)	
	デバッグ機能(SuperH ファミリ、H8SX, H8S, H8 ファミリ、および RX ファミリ用デバッガ)	
	ウィンドウおよびダイアログボックス	
1.15	ヘルプ	14
	マクロ生成支援機能	
	コマンド	
	TCL コマンドと High-performance Embedded Workshop コマンドの親和性の拡張	
1.19	TCL/TK コマンド	17
1.20	TCL ツールキットとコマンドラインのコマンド	18
2.	ツールチェイン仕様補足	
2.1	ファイル拡張子	19
3.	ツールチェインアップグレード	20
4.	Microsoft® Windows Vista® および Windows® 7 互換性対応	
4 1	Windows Vista® および Windows® 7 における標準権限での使用について	23

## 1. 注意事項

## 1.1 ネットワークドライブ

### 1.1.1 PC 間の時計のずれ

ソースファイルや出力ファイルの更新日付は、保存した PC の時間で更新されます。

ネットワークで共有しているプロジェクトやソースファイルをビルドするとき、PC 間の時計のずれが原因で再ビルドが正確にできない場合があります。

このような場合、PC 間の時計を合わせるか、または[ビルド -> すべてをビルド]でプロジェクトをビルドしてください。

## 1.1.2 ネットワーク上プロジェクトのビルド

ネットワーク経由で開いているプロジェクトは、ビルド中にネットワークの状態に依存して、ファイルが開けないなどのエラーが発生することがあります。

例えば、C/C++コンパイラは次に示すエラーメッセージを出力する場合があります。

C3019(F) Cannot open source file

このような場合、エラーが発生したファイルを再度ビルドしてください。

### 1.1.3 プロジェクトへのファイルの追加

ネットワーク経由で、リモートドライブ上にあるファイルをプロジェクトに[プロジェクト ->ファイルの追加]などで追加するときに、ネットワークの状態に依存して High-performance Embedded Workshop がアプリケーションエラーで終了することがあります。

このような場合、再度追加を試みてください。アプリケーションエラーが何度も発生する場合は、ファイルをローカルドライブにコピーしてからプロジェクトに追加することをご検討ください。

## 1.2 エディタのシンタックスカラーリング機能

High-performance Embedded Workshop のエディタには、ソースプログラムファイルのコメント文を色付けするシンタックスカラーリング機能があります。ソースプログラムの書き方によって、シンタックスカラーリング機能が正しく動作しないことがあります。

このような場合、シンタックスカラーリング機能を外してください。シンタックスカラーリング機能を外すには、[基本設定 -> オプション]を選択し、[オプション]ダイアログボックスの[エディタ]タブで[シンタックスカラーリング]のチェックを外し、[OK]ボタンをクリックしてください。

C/C++言語のソースファイルは、以下のような場合、シンタックスカラーリング機能が正しく動作しません。

### (i) コメントをネストした場合

例: /\* /\* \*/ 下線部がコメントの色になる。

## (ii) 文字列内に"/\*"や"\*/"を記述した場合

なお、C/C++言語のソースファイルで、\* 文字や、/ 文字をコメント以外の目的で隣接して記述することを避けることで回避できます。

## 1.3 ファイル依存関係

High-performance Embedded Workshop は、ソースファイルがインクルードしているファイルを検索し、ワークスペースウィンドウの[Projects]タブに依存ファイルとして表示します。

この依存ファイルは、ファイル依存関係としてビルド可否の判断に利用されます。

ソースプログラムがシンタックス上正しく記述されていても、依存ファイルの検索中にアウトプットウィンドウの[Build]タブにエラーメッセージが出力された場合は、インクルードしているファイルが依存ファイルとして正しく検索されていないことがあります。

このような場合、(a) [ビルド -> すべてをビルド]を選択してすべてのファイルをビルドするか、または、(b) ワークスペースウィンドウの[Projects]タブで当該ソースファイル名を選択し、[ビルド -> コンパイル ファイル名]で当該ソースファイルをコンパイルしてください。

C/C++言語のソースファイルに記述されているインクルードファイルは、以下のような場合、依存ファイルとして正しく検索されません。

(i) 文字列内に"/\*"や"\*/"がある場合

例: char A[] = "/\*"; コメント開始とみなされる。 #include "file.h" このファイルは依存ファイルとして検索されない。 char B[] = "\*/"; コメント終了とみなされる。

High-performance Embedded Workshop で、#if、#ifdef、#define などのプリプロセッサ文をサポートするには、[ビルド]メニューからツールチェインのオプションダイアログボックスを表示し、[全般]タブで [依存関係検索にてプリプロセッサ文サポート]をチェックします。

表 1.1 にサポートしているプリプロセッサ文を示します。

表 1.1 プリプロセッサ文

プリプロセッサ文	説明
#define	識別子を定義する。プリプロセッサ文が識別子を含んでいると、識別子を定義
	された文字列で置き換える。
	例: #define NICE_FILE "nice.h"
	#include NICE_FILE
#undef	定義された識別子の置き換えを無効にする。
#include	指定されたファイルが現在のソースファイルに依存していることを示す。
#if~#else~#endif	#if <式>は、式を満たすソースコードの範囲のみ評価する。
#elif	式の結果が 0 以外のときは、#if から#else の範囲。
	式の結果が0のときは、#else から#endifの範囲。
Defined マクロ	defined(<識別子>)文は、識別子が定義されていれば1を返す。
	識別子が定義されていなければ0を返す。
	通常、#if defined(MACRO)として記述される。
#ifdef	#ifdef <識別子>文は、識別子が定義されていれば#if 1 と等価になる。
#ifndef	識別子が定義されていなければ#if 0 と等価になる。
	#ifndef <識別子>文は、識別子が定義されていれば#if 0 と等価になる。
	識別子が定義されていなければ#if 1 と等価になる。
#line	無視する。
#error	無視する。
#pragma	無視する。
#	無視する。

プリプロセッサ文に<式>がある場合、依存関係の検索では以下の 10 種類の演算子をサポートしています。これ以外の演算子を使用すると、依存関係の検索は正しく動作しません。

```
( ), !, <, <=, >, >=, ==, !=, &&, | |
```

[依存関係検索にてプリプロセッサ文サポート]がチェックされていると、C/C++言語のソースファイルに記述されているインクルードファイルは、以下のような場合、依存ファイルとして正しく検索されません。

- (ii) 以下に示すコンパイラパッケージに含まれるツールチェインを使用し、プリプロセッサ文の<式>が文字列を比較する条件判断文を含む場合
  - SuperH ファミリ用 C/C++コンパイラパッケージ V.5.1 ~ V.9.03 Release 02
  - H8SX, H8S, H8 ファミリ用 C/C++コンパイラパッケージ V.3.0A ~ V.7.00 Release 00
  - RX ファミリ用 C/C++コンパイラパッケージ V.1.00 Release 00
  - M16C シリーズ用 C コンパイラパッケージ V.5.30 Release 0~ V.5.45 Release 01
  - M32C シリーズ用 C コンパイラパッケージ V.5.40 Release 0 ~ V.5.42 Release 00
  - R32C シリーズ用 C コンパイラパッケージ V.1.01 Release 00 ~ V.1.02 Release 01
  - M32R ファミリ用 C/C++コンパイラパッケージ V.5.00 Release 00 ~ V.5.01 Release 01
  - 740 ファミリ用 C コンパイラパッケージ V.1.00 Release 1~ V.1.01 Release 02

```
例: #define A 'a'
#if (A == 'a') この記述はエラーになり、これ以降の行に記述
されているインクルードファイルは依存ファイル
として検索されない。
#include "file.h" このファイルは依存ファイルとして検索されない。
#endif
```

- (iii) 以下に示すコンパイラパッケージに含まれるツールチェインを使用し、アセンブリ言語の文で、命令のオペランドにイミディエイト値がある場合
  - SuperH ファミリ用 C/C++コンパイラパッケージ V.5.1 ~ V.9.03 Release 02
  - H8SX, H8S, H8ファミリ用 C/C++コンパイラパッケージ V.3.0A ~ V.7.00 Release 00
  - M16C シリーズ用 C コンパイラパッケージ V.5.30 Release 0 ~ V.5.45 Release 00
  - M32C シリーズ用 C コンパイラパッケージ V.5.40 Release 0 ~ V.5.41 Release 01
  - R32C シリーズ用 C コンパイラパッケージ V.1.01 Release 00 ~ V.1.02 Release 00
  - M32R ファミリ用 C/C++コンパイラパッケージ V.5.00 Release 00 ~ V.5.01 Release 01
  - 740 ファミリ用 C コンパイラパッケージ V.1.00 Release 1~ V.1.01 Release 02

- (iv) 以下に示すコンパイラパッケージに含まれるツールチェインを使用し、アセンブリ言語の文で、命令のオペランドに基数が付いた整数定数がある場合
  - SuperH ファミリ用 C/C++コンパイラパッケージ
  - H8SX, H8S, H8 ファミリ用 C/C++コンパイラパッケージ

アセンブリ言語のソースファイルに記述されているインクルードファイルは、以下のような場合、依存ファイルとして正しく検索されません。

(v) AS30、AS308、または AS100 対応のソースファイルに、インクルードファイルを指定するための指示 命令"..FILE"や"@"を含む場合

```
例: <sample.a30 ファイル>
.INCLUDE ..FILE@.inc この記述はエラーになり、sample.incは
依存ファイルとして検索されない。
```

## 1.4 ナビゲーション機能

### 1.4.1 C 言語関数定義ナビゲーション

High-performance Embedded Workshop は、C/C++言語のソースファイルから C 言語関数定義を検索してワークスペースウィンドウの Navigation タブに表示します。

検索された C 言語関数定義をダブルクリックすると、該当ファイルの該当行がエディタで開きます。 C 言語関数定義の検索は、プリプロセッサ文を無視しています。そのため、C 言語関数定義を正しく解析できない場合があります。

例えば、以下のような場合、2 つの func()関数定義をワークスペースウィンドウの Navigation タブ に表示します。

```
#define DEF 1
#ifdef DEF
void func(void)
{
}
#else
int func(int a)
{
}
```

## 1.4.2 ナビゲーション機能の"C++ Classes"のデフォルト設定

ナビゲーション機能の"C++ Classes"のデフォルト設定は無効(チェックなし)です。 設定状態は、[カテゴリ選択]ダイアログボックスで確認できます(図 1.1 参照)。 [カテゴリ選択]ダイアログボックスは、[Navigation]タブウィンドウ内で右クリックし、[カテゴリの 選択]を選択すると開きます。



図 1.1 [カテゴリ選択]ダイアログボックス

## 1.4.3 ナビゲーション機能の"C++ Classes"のウォーニングメッセージ

ナビゲーション機能の"C++ Classes"の設定を有効 (チェックあり) に変更すると、以下のウォーニングメッセージを表示します (図 1.2 参照)。

本機能を有効にするとバックグランドでナビゲーション機能が動作するため、PC の CPU 占有率が 100%近くになる場合があり(PC の性能による)、High-performance Embedded Workshop の反応が 遅くなることがあります。



図 1.2 ウォーニングメッセージ

## 1.5 式の指定

- (1) 式のシンボルに C++プログラムの関数名は指定できません。
- (2) 関数名として多重定義演算子は指定できません。

## 1.6 エラーが発生したファイルを開く

ビルドで C/C++ Compiler、Assembler がアウトプットウィンドウに出力したエラー / ウォーニングメッセージをダブルクリックすると当該ファイルを開き、当該行にカーソルを移動します。しかし、当該ファイルウィンドウがエディタウィンドウエリアで最小化されている場合、当該エラー / ウォーニング メッセージをダブルクリックしても開かれません。このような場合、当該ファイルを元のサイズに戻すか、または、最大化してください。

## 1.7 オプションダイアログボックス上でのディレクトリパス指定

High-performance Embedded Workshop の[ビルド]メニューからコンパイラなど各ツールのオプションダイアログボックスを表示できます。

図 1.3 に示すようなオプションダイアログボックスで"Custom directory"を選択した場合、"Directory"フィールドには必ず絶対パスを指定してください。

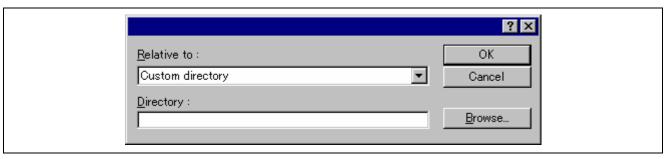


図 1.3 ディレクトリパス指定ダイアログボックス

## 1.8 ワークスペースの作成言語

High-performance Embedded Workshop V.4.08 をインストールした時に指定した言語(日本語か英語)と 異なる言語の High-performance Embedded Workshop V.3.01 で作成したワークスペースを開くと、ツールバーが表示されません。

その場合、[基本設定 -> カスタマイズ]の[ツールバー]タブでツールバーを選択して表示してください。

### 1.9 日本語版の表示言語

メニューおよびダイアログボックス内の文字列が一部英語で表示される場合があります。

## 1.10 エディタウィンドウの文字セット

エディタウィンドウは Unicode をサポートしていません。

## 1.11 ツールアドミニストレーション

## 1.11.1 コンポーネントおよびツールのアンインストール

High-performance Embedded Workshop V.4.05 以降、[ツールアドミニストレーション]ダイアログボックスの[アンインストール]ボタンを削除しました。

これにより、コンポーネントおよびツールは個別にアンインストールできなくなりました。 アンインストールが必要な場合は、Windows® のコントロールパネルから High-performance Embedded Workshop セット全体をアンインストールしてください。

### 1.11.2 コンポーネントおよびツールの登録解除

[ツールアドミニストレーション]ダイアログボックスの[登録解除]ボタンを使用して、コンポーネント、またはツールの High-performance Embedded Workshop への登録を解除すると、その後、これらを含む該当製品をインストールしても、登録解除したコンポーネントまたはツールは High-performance Embedded Workshop へ登録されません。

これらを登録するには、[ツールアドミニストレーション]ダイアログボックスの[登録]ボタンを使用してください。

[ツールアドミニストレーション]ダイアログボックスは、ワークスペースを開く前に、[ツール]メニューから[アドミニストレーション]を選択して開いてください。

## 1.12 デバッグ機能(共通)

## 1.12.1 ソースレベル実行

ステップインを実行すると標準 C ライブラリにも移行します。上位関数に戻るにはステップアウトを使用してください。

また、for および while 文では、1回のステップで次の行に進まないことがあります。進める場合はもう一度ステップしてください。

インライン関数内からステップアウトを実行した場合は、インライン関数の呼び出し元関数ではなく、インライン関数を呼び出した(非インライン)関数の呼び出し元関数まで実行します。

例: main()関数から関数(Func())を呼び出し、Func()関数からインライン関数(InFunc())を呼び出している場合

```
10 int a = 0;
11 main()
12 {
13
    Func();
14 }
15
16 void Func()
17 {
18
   InFunc();
19 }
21 inline void InFunc()
22 {
23
     a++;
```

23 行目でステップアウトを実行すると、ステップは 14 行目で停止します ( Func()関数内でステップアウトを実行した場合と同じ結果となります。19 行目では停止しません)。

# 1.12.2 ロードモジュール作成後のソースファイル位置移動

ロードモジュール作成後にソースファイルを移動させた場合、作成したロードモジュールのデバッグ中にソースファイルを指定するための[ファイルを開く]ダイアログボックスが表示されることがあります。

対応するソースファイルを選択し[開く]ボタンをクリックしてください。

### 1.12.3 エディタウィンドウ

エディタウィンドウに表示中のプログラムを変更し、ソースファイルとロードモジュールを再ロードしたときは、一旦エディタウィンドウを閉じて、開き直してください。 そのまま使用すると、エディタウィンドウの表示が不正になる場合があります。

### 1.12.4 逆アセンブリウィンドウ

逆アセンブリウィンドウのソースモードを表示し、さらにエディタウィンドウの混合モードを表示している場合に、ステップ実行、またはプログラム実行を行なうと、逆アセンブリウィンドウのソースモードで、プログラムカウンタ(PC)のアイコン(♥)が正しい位置に表示されない場合があります。

逆アセンブリウィンドウのソースモードでデバッグする場合は、エディタウィンドウをソースモードに切り替えてご使用ください。

### 1.12.5 スタックトレースウィンドウ

割込み関数を実行した状態でスタックトレースウィンドウを表示させると、割込み関数以前の表示が不正となります。

### 1.12.6 メモリウィンドウ

(1) メモリデータの検索

メモリデータの検索を開始する前に、メモリウィンドウの[Data]または[Code]カラム上で、マウスにより検索範囲を選択するか、または任意の位置をクリックしてください。

メモリウィンドウ上で右クリックしただけでは、検索を開始しても見つかったデータは強調表示されません。

### (2) 表示フォント

表示フォントをプロポーショナルフォントに設定している場合、表示が欠ける場合があります。 表示フォントを固定幅のフォントに変更してください。

# 1.12.7 エントリポイント

リンケージエディタの ENTRY オプション等でエントリポイントを指定した場合でも、プログラムのダウンロード時 PC へのエントリポイントアドレスの設定は行ないません。 プログラム実行前に PC の設定を行なってください。

### 1.12.8 レジスタウィンドウ

High-performance Embedded Workshop V.4.04 以降は、レジスタ値をセッションに保存しないようにしました。

# 1.13 デバッグ機能 ( SuperHファミリ、H8SX, H8S, H8 ファミリ、およびRXファミリ用デバッガ )

### 1.13.1 ラインアセンブル

ラインアセンブル時の入力基数のデフォルトは、基数の設定に関係なく 10 進数です。 16 進数を入力する場合は、H'または Ox を指定してください。

### 1.13.2 SYSROF 形式ファイルのロード

SYSROF 形式のデバッグ対象プログラムはロードできません。 デバッグ対象プログラムは ELF/DWARF2 形式で作成してください。

## 1.13.3 Hitachi Debugging Interface セッションファイル

Hitachi Debugging Interface で保存したセッションファイルは使用できません。 High-performance Embedded Workshop のプロジェクトワークスペースとして作成し直してください。 なお、Hitachi Debugging Interface のコマンドファイルは使用可能です。

### 1.13.4 プロファイル

プロファイルは、オーバーレイ機能には対応していません。

## 1.13.5 スタックトレースウィンドウ

インライン展開された関数は呼び出し履歴に表示されません。

### 1.13.6 変数値の参照

(1) インライン関数の局所変数

インライン関数内では、関数パラメータおよび関数内で定義した局所変数の値を以下のウィンドウまたは機能を使用して参照できません。

### (2) 最適化後の局所変数

最適化オプションでコンパイルされた C ソースの局所変数の値は、生成されたオブジェクトコードによっては、以下のウィンドウまたは機能を使用して正しく表示できないことがあります。 逆アセンブリウィンドウを表示し生成されたオブジェクトコードを確認してください。

- スタックトレースウィンドウ
- ローカルウィンドウ
- ウォッチウィンドウ
- ツールチップウォッチ機能
- インスタントウォッチ機能

### 1.13.7 変数値の変更

以下のウィンドウを使用して、変数値を変更する場合、入力するデータに日本語文字列を指定しないでください。

日本語文字列を入力する場合はメモリウィンドウを使用してください。

- ローカルウィンドウ
- ウォッチウィンドウ

### 1.13.8 ウォッチウィンドウ

プログラムのデバッグ中、ウォッチアイテムがスコープを外れると、展開中のウォッチアイテムは 自動的に展開を閉じます。

その場合、最初の子アイテムに設定されていたリアルタイム設定をウォッチアイテム全体の設定と して保持します。

### 1.13.9 オーバーレイの構成ダイアログボックス

オーバーレイセクションに連結させたセクションがその他のオーバーレイセクションにも連結されているかのように表示してしまう場合があります。

以下の(1)、(2)、(3)のようにオーバーレイセクションを配置し(例1)、ビルドの結果、各セクションが例2のように配置された場合、[オーバーレイの構成]ダイアログボックスには以下のように表示されます(例3)。

- 例 1: <Section>
  - (1) "P11, P12"
  - (2) "P21"
  - (3) "P31, P32"
- 例 2: <Sec> <Start End >
  - "P11" 0x1000 0x10FF (\*1)
    - "P12" 0x1100 0x12FF
    - "P21" 0x1000 0x10FF (\*1)
    - "P31" 0x1000 0x105F
    - "P32" 0x1060 0x11FF
  - \*1. 連結セクションを持つ"P11"とそれを持たない"P21"セクションの終了アドレスが同じ。
- 例 3: <Section>
  - (1) "P11, P12"
  - (2) "P21, P12" (\*2)
  - (3) "P31, P32"
  - \*2. "P12"セクションが"P21"にも連結されているかのように表示。

#### 1.13.10 コマンド

(1) WATCH EDIT コマンド

ウォッチウィンドウで、異なったスコープを指定した同名の変数を参照している場合、WATCH\_EDIT コマンドを使用してその変数の値を編集すると、ウォッチウィンドウにあるその名前を持つ最初の変数のみ値を変更します。

- 1.13.11 SuperH ファミリ、および H8SX, H8S, H8 ファミリ用シミュレータ・デバッガ
  - (1) メモリリソース設定

メモリリース変更機能の仕様がエミュレータと異なりますので、ご注意ください。 シミュレータ・デバッガの仕様は以下のとおりです。

- 変更前と変更後のメモリ属性(Read、Write、Read/Write)が同一の場合 メモリリソースサイズの変更と判断してメモリリソースサイズを指定値に変更します。
- 変更前と変更後のメモリ属性が異なる場合 メモリリソース属性の変更と判断して、指定範囲のメモリ属性を指定値に変更します。
- (2) PC ブレークポイント設定数と[条件を指定して実行]メニューのテンポラリ PC ブレークポイント設定数

PC ブレークポイント設定数と[条件を指定して実行]メニューのテンポラリ PC ブレークポイント設定数の合計は、最大 1,024 個です。

したがって、PC ブレークポイントを 1,024 個設定した状態では、[条件を指定して実行]メニューの テンポラリ PC ブレークポイントでの指定は無効となります。

PC ブレークポイントと[条件を指定して実行]メニューのテンポラリ PC ブレークポイントは、設定数の合計が 1.024 個以下で使用してください。

- (3) デバッグの設定ダイアログボックス
- 自動的にターゲットを接続しない [オプション]タブの[自動的にターゲットを接続しない]チェックボックスの設定にかかわらず、[デバッグの設定]完了時にターゲットを接続します。
- (4) SH-4 シミュレータ、SH-4 with BSC シミュレータ
- SH-4 with BSC シミュレータでは、DMA 転送において、転送元、転送先のアドレスの下位 3 ビットが違う場合、転送最後のデータが不正になります。

例: SAR0=2000 DAR0=4004 DMATCR 0=2 CHCR0=5491 2000 番地のメモリ内容: 0102030405060708

DMA 転送終了後の 4004 番地の内容: 0106

- SH-4 シミュレータ、SH-4 with BSC シミュレータともに、デコード完了したアドレスの命令を変更してもパイプラインリセット実行からの開始になりません。
- SH-4 with BSC シミュレータでは、ブレークデータの設定で指定したデータのサイズと異なるサイズでメモリアクセスを行なった場合、ブレークデータの条件が成立しても停止しない場合があります。その場合、メモリアクセスのサイズとブレークデータの設定で指定するデータのサイズを同してください。
- SH-4 シミュレータ、SH-4 with BSC シミュレータともに、倍精度 FDIV 命令、倍精度 FSQRT 命令のパイプラインが実機と異なります。F3 ステージのパイプが 1 サイクル多く表示されます。

#### (5) SH-3DSP シミュレータ

• DSP 繰り返し (ループ) 制御中の例外コード

DSP 繰り返し(ループ)制御中に例外が発生した場合、EXPEVT(例外事象レジスタ)に設定する例外コードがプログラミングマニュアルとは異なります。

DSP 繰り返し(ループ)制御中の例外コード

一般例外事象	プログラミングマニュアル	シミュレータ
TLB ミス例外 / TLB 無効例外(読み出し)	н'070	н'040
TLB ミス例外/TLB 無効例外(書き込み)	н'070	н'060
TLB 保護例外(読み出し)	H'0D0	H'0A0
TLB 保護例外(書き込み)	H'0D0	H'0C0
CPU <b>アドレスエラー</b> (読み出し)	н'070	H'0E0
CPU アドレスエラー(書き込み)	н'070	Н'100

● X/Y メモリアクセスの競合

XRAM メモリ ( XROM、 YROM、 YRAM メモリでも同様 ) に命令コードとデータを配置した場合、命令コードフェッチによる XRAM メモリアクセスと MOVX、 MOVY 命令による XRAM メモリアクセスが同一スロットになっても競合によるストールは発生しません。

このため、サイクル数が異なります。

4の倍数番地以外から実行した場合のパイプライン

4の倍数番地以外から実行した場合のフェッチのパイプラインがプログラミングマニュアルとは異なります。

このため、4 の倍数番地以外から実行した場合のフェッチで例外が発生した場合の動作が異なります。

例: 4 の倍数番地以外から実行した場合のパイプライン

プログラミングマニュアル シミュレータ
IF IF ID EX IF ID EX IF ID EX if ID EX if ID EX

- (6) トレース機能(SH3、SH3E、SH-3DSPのみ)
- FPU、MAC、DSP レジスタのアクセス情報
   FPU、MAC、DSP レジスタに書き込む命令を実行しても、トレース情報にアクセス情報は表示しません。
- パイプライン表示

プログラミングマニュアルの表記上ライトバックがない命令でも、命令動作上レジスタへの書き込みが発生する場合、シミュレータ・デバッガのトレース表示では、メモリアクセス,ライトバックのステージとして表示します。 サイクル数は正しく表示します。

例: TRAPA 例外のトレース表示

シミュレータ・デバッガでのトレース表示 プログラミング マニュアルの表						
	IF DE EX MA SW	IF DE EX MA SW				
05 NOP	06 05	06 05				
06 TRAPA #H'10	07 06 05	06				
	06	06				
	06 06	06				
	06 06	06				
	06 06 06 (06):SSR<-60000001	06				
	08 06 06 (06):SPC<-00001006	07				
	09 08	08 07				

(7) H8SX シミュレータ

H8SX シミュレータは、H8SX CPU のミドルモードをサポートしていません。

(8) シミュレータターゲットの選択

プロジェクト作成時のデバッガターゲット選択では、CPU、および動作モードと一致するシミュレータを選択してください。

- (9) SH2A-FPU シミュレータ・デバッガの内蔵 RAM 領域
  - SH2A-FPU シミュレータ・デバッガのメモリマップでは 0xFFF80000 番地から 0xFFFBFFFF 番地を内蔵 RAM 領域として表示しますが、0xFFFA0000 番地から 0xFFFBFFFF 番地は予約領域です。このため、SH2A-FPU シミュレータ・デバッガご使用時は以下の点にご注意ください。
- 内蔵 RAM 領域は 0xFFF80000 番地から 0xFFF9FFFF 番地以内を使用してください。
- スタック領域を内蔵 RAM 領域に割り当てる場合も、0xFFF80000 番地から 0xFFF9FFFF 番地以内となるようにスタックポインタ初期値を設定してください。
- (10) 旧バージョン用ワークスペースの保存

High-performance Embedded Workshop V.4.04 以降では、旧バージョン形式でワークスペースを保存すると、シミュレータ・デバッガの動作が非常に遅くなることがあります。

その場合、トレース容量を32,768以下にしてください。

## 1.14 ウィンドウおよびダイアログボックス

### 1.14.1 スクロール

- (1) インテリジェントマウスで上スクロールができない場合があります。 この場合は、ウィンドウ上のスクロールボタンを使用してください。
- (2) タッチパッドの操作面にスクロールゾーンがある PC の場合、逆アセンブリウィンドウ上、または混合モードまたは逆アセンブリモードで開いているエディタウィンドウ上でクリックボタンを押したまま、スクロールゾーンでスクロールすると、アプリケーションエラーが発生する場合があります。

上記のウィンドウでは、クリックボタンを押したまま、スクロールゾーンでスクロールしないでください。

# 1.14.2 ウィンドウ位置の保存

ウィンドウの表示位置は、以下の操作により移動する場合があります。

- セッションのセーブ後にセッションのリフレッシュ
- ターゲットの切断後の再接続
- バーチャルデスクトップの切り替え

### 1.15 ヘルプ

英語版 Windows® でヘルプ使用時に以下のメッセージボックスが表示される場合があります。
[Download]ボタンをクリックして"Japanese Text Display Support"をダウンロードするか、[Never download any of these components]チェックボックスをチェックしてから[Cancel]ボタンをクリックしてください。"Japanese Text Display Support"をダウンロードしなくても、ヘルプの表示には影響ありません。

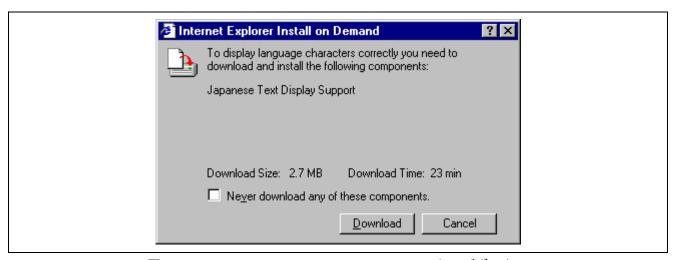


図 1.4 Internet Explorer Install on Demand メッセージボックス

## 1.16 マクロ生成支援機能

マクロ生成支援機能を使用し、[デバッグ -> 実行]、[デバッグ -> プログラムの停止]の選択を記録したマクロを実行すると、[デバッグ -> 実行]は実行されますが、[デバッグ -> プログラムの停止]は実行されません。

## 1.17 コマンド

1.17.1 MEMORY EDIT コマンド

文字を一重引用符 (') で囲っても ASCII 文字列をデータとして入力できません。 文字列をデータとして入力する場合は対応する数値を入力してください。

- 1.17.2 コマンドバッチファイル実行タイミング(デバッグの設定ダイアログボックス) [オプション]タブの[コマンドバッチファイル実行タイミング]でバッチファイルを設定した場合、 以下の制限があります。以下のコマンドの代わりにメニュー、ツールバー等、GUI 経由での実行を ご検討ください。
  - (1) [オプション]タブの[コマンドバッチファイル実行タイミング]で[At target connection]を選択した バッチファイル中に"INITIALIZE"コマンドを指定した場合、[コマンドライン]ウィンドウで "INITIALIZE"コマンドを実行しないでください。初期化する場合は、[デバッグ -> 初期化]を選択 してください。
  - (2) [オプション]タブの[コマンドバッチファイル実行タイミング]で[At target connection]を選択した バッチファイル中に以下のコマンドは指定しないでください。
  - (a) [OPEN\_WORKSPACE]コマンド
  - (b) [CHANGE\_PROJECT]コマンド
  - (c) [CHANGE\_CONFIGURATION]コマンド
  - (3) [オプション]タブの[コマンドバッチファイル実行タイミング]で[Before download of modules]または [After download of modules]を選択した場合、[コマンドライン]ウィンドウで"FILE\_LOAD"または "FILE\_LOAD\_ALL"コマンドを実行しないでください。
  - (4) [オプション]タブの[コマンドバッチファイル実行タイミング]で[Before download of modules]または [After download of modules]を選択したバッチファイル中に以下のコマンドは指定しないでください。
  - (a) [OPEN\_WORKSPACE]コマンド
  - (b) [CHANGE PROJECT]コマンド
  - (c) [CHANGE CONFIGURATION]コマンド
  - (d) [GO]コマンド
  - (e) [GO RESET]コマンド
  - (f) [GO TILL]コマンド
  - (g) [STEP]コマンド
  - (h) [STEP\_OUT]コマンド
  - (i) [STEP OVER]コマンド
  - (5) [オプション]タブの[コマンドバッチファイル実行タイミング]で[After reset]を選択した場合、[コマンドライン]ウィンドウで"RESET"コマンドを実行しないでください。
  - (6) [オプション]タブの[コマンドバッチファイル実行タイミング]で[After reset]を選択したバッチファイル中に以下のコマンドは指定しないでください。
  - (a) [OPEN WORKSPACE]コマンド
  - (b) [CHANGE PROJECT]コマンド
  - (c) [CHANGE\_CONFIGURATION]コマンド
  - (d) [RESET]コマンド

1.17.3 コマンドバッチファイル実行順序(デバッグの設定ダイアログボックス)

[オプション]タブの[コマンドバッチファイル実行順序]では相対パス形式でバッチファイルを指定すると、正しくファイルにアクセスできない場合があります。

プレースホルダを適用できないバッチファイルは絶対パス形式で指定してください。

1.17.4 コマンドラインのファイル指定

コマンドラインでファイルを指定する場合、プレースホルダを使用してください(TCL コマンドは除く)。

プレースホルダに含まれていないディレクトリを指定する場合は、絶対パスを指定してください。 絶対パスで指定すると、他のマシンやパスの内容が異なる環境に移動する場合、正しくファルを参 照できなくなるため、ファイル指定をやり直してください。

FILE\_LOAD ELF/DWARF2 \$(CONFIGDIR)\forall Y\forall demo.abs

1.17.5 コマンドのパラメータ区切り

TCL が有効の場合、TCL コマンドを使用できます。パラメータが Tab で区切られた TCL コマンドバッチの使用も可能です。

```
for {set i 0} {$i < 2} {incr i} {
   puts [memory_display 300 10]
}</pre>
```

TCL が無効の場合、TCL コマンドは使用できません。パラメータの区切りとして半角スペースのみ使用できます。そのため、Tab を使用すると以下のようなエラーになります。

```
>TCL
TCL Disabled
>memory_display 300 10
Error: Invalid command
```

1.17.6 CACHE コマンド

エミュレータ・デバッガのマニュアルに CACHE コマンドの記載がない場合、CACHE コマンドを 指定しないでください。

キャッシュのメモリアクセス単位は、0x3FF など固定値のため、指定したアクセス範囲より大きい 範囲のキャッシュメモリをアクセスする場合があります。

- 1.17.7 コマンド短縮形
  - REMOVE FILE コマンド

High-performance Embedded Workshop V.3.01 から短縮形"RF"を"REM"に変更しました。

1.18 TCLコマンドとHigh-performance Embedded Workshopコマンドの親和性の拡張

TCL コマンドと High-performance Embedded Workshop コマンドの親和性を拡張 (\*) したことにより、以下のような記述の場合、High-performance Embedded Workshop コマンド"memory\_display 300 10"の実行結果は出力されません。

```
for {set i 0} {$i < 2} {incr i} {
   memory_display 300 10
}</pre>
```

High-performance Embedded Workshop コマンドの実行結果を出力する場合は、TCL コマンド"puts"のパラメータとして High-performance Embedded Workshop コマンドを"[]"内に置いてください。

```
for {set i 0} {$i < 2} {incr i} {
   puts [memory_display 300 10]
}</pre>
```

\*. TCL コマンドと High-performance Embedded Workshop コマンドの親和性を拡張したことにより、TCL コマンド"set"のパラメータとして、High-performance Embedded Workshop コマンドの実行結果を変数へ代入できるようになりました。

以下の例では、High-performance Embedded Workshop コマンド"memory\_display 300 10"の実行結果が変数"md\_300\_10"に代入されます。

TCL コマンド"set"で変数"md\_300\_10"を指定すると、実行結果を参照できます。

```
set md_300_10 [memory_display 300 10]
```

### 1.19 TCL/TKコマンド

(1) 対話モードの中止方法

[TCL ツールキット]で対話モードからコマンド入力モードに戻るには、"/."を入力してください。 現在のモードが対話モードかコマンド入力モードかは、"/."を入力して判断してください。

(2) 日本語ディレクトリ

High-performance Embedded Workshop を日本語ディレクトリにインストールした場合、 High-performance Embedded Workshop を実行すると、TCL/TK のライブラリを仮想ドライブ("u:"ドライブ)に関連付けます。

すでに"u:"ドライブを使用している場合、High-performance Embedded Workshop の開始時に"u:"ドライブとの接続を解除し、High-performance Embedded Workshop の TCL/TK ライブラリと接続します。High-performance Embedded Workshop の終了時に、TCL/TK のライブラリは接続を解除します。元のパスとは、自動で再接続しません。

仮想ドライブを使いたくない場合、High-performance Embedded Workshop を英語ディレクトリにインストールしてください。

(3) ログファイルの内容の消去方法

[TCL ツールキット]を使用している場合、[Console]画面の内容がログファイルに出力されます。 ログファイルは、以下のディレクトリにテキスト形式ファイルで生成されます。

例1: Windows® 2000 または Windows® XP Operating System
C:\text{YDocuments and Settings}<-ユーザ名>\text{YLocal Settings}\text{Temp}\text{log.txt}

例 2: Windows Vista® または Windows® 7 Operating System C:\U00e7Users\u00e4<2-ザ名>AppData\u00e4Local\u00e4Temp\u00e4log.txt

[TCL ツールキット]を終了すると、ログファイルの内容は自動的に消去されます。 [TCL ツールキット]の使用中にログファイルの内容を消去したい場合、以下のコマンドを実行してください。

```
set dir $env(TEMP)
set dataFile [open $dir/log.txt {RDWR TRUNC}]
close $dataFile
```

## 1.20 TCLツールキットとコマンドラインのコマンド

- (1) "trace"コマンド
- (a) [TCL ツールキット]で TCL の"trace"コマンドを実行する場合、コマンド名をすべて小文字で指定してください。
- (b) [TCL ツールキット]で High-performance Embedded Workshop の"TRACE"コマンドを実行する場合、コマンド名をすべて大文字で指定してください。
- (c) [コマンドライン]ウィンドウで TCL の"trace"コマンドを実行する場合、コマンド名を"tcl\_trace"に置き換えて指定してください。
- (2) "clock"コマンド
- (a) [TCL ツールキット]で TCL の"clock"コマンドを実行する場合、コマンド名をすべて小文字で指定してください。
- (b) [TCL ツールキット]でエミュレータの"CLOCK"コマンドを実行する場合、コマンド名をすべて大文字で指定してください。
- (c) [コマンドライン]ウィンドウで TCL の"clock"コマンドを実行する場合、コマンド名を"tcl\_clock"に置き換えて指定してください。
- (3) "event"コマンド
- (a) [TCL ツールキット]で TK の"event"コマンドを実行する場合、コマンド名をすべて小文字で指定してください。
- (b) [TCL ツールキット]でエミュレータの"EVENT"コマンドを実行する場合、コマンド名をすべて大文字で指定してください。
  - "CLOCK"コマンドと"EVENT"コマンドは、一部のエミュレータではサポートしていません。

## 2. ツールチェイン仕様補足

## 2.1 ファイル拡張子

High-performance Embedded Workshop がツールチェインの各ツールをビルド実行したとき、コンフィグレーションディレクトリにサブコマンドファイルを残します。

C/C++ Compiler と Assembler のサブコマンドファイルの拡張子は、入力ファイル名の拡張子を表 2.1 のように変更したファイル名になります。

C/C++ Library Generator、OptLinker のサブコマンドファイルは、プロジェクト名に表 2.1 の拡張子を付加したファイル名になります。

サブコマンドファイルは隠しファイルの属性を持ちます。隠しファイルを表示する場合、ディレクト リを表示するウィンドウのプロパティをすべてのファイルを表示するように設定する必要があります。

表 2.1 ツールチェインサブコマンドファイル拡張子

shc         SuperH RISC engine C/C++ Compiler           sha         SuperH RISC engine Assembler           h8g         H8S, H8/300 C/C++ Library Generator           h8c         H8S, H8/300 Assembler           hla         H8S, H8/300 Assembler           hlk         OptLinker           m16cl         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Librarian           m16ci         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Librarian           m16cc         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Enginer           m16ct         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny mkmrtbl (MR30)           m32cl         M32C/80, M16C/80, M16C/70 Librarian           m32ci         M32C/80, M16C/80, M16C/70 C Compiler           m32ct         M32C/80, M16C/80, M16C/70 mr308tbl (MR308)           m32rl         M32R C Compiler           m32rl         M32R Librarian           m32ra         M32R Load module converter           m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Librarian           1001         R32C/100 Librarian           1002         R32C/100 Librarian           1003         R32C/100 C Compiler           74B         74C/C++ Library Generator <th>拡張子</th> <th>ファイルグループ</th>	拡張子	ファイルグループ
sha         SuperH RISC engine Assembler           h8g         H8S, H8/300 C/C++ Library Generator           h8c         H8S, H8/300 C/C++ Compiler           h8a         H8S, H8/300 Assembler           hlk         OptLinker           m16cl         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Linker           m16ci         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Librarian           m16cc         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny C Compiler           m16ct         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny mkmrtbl (MR30)           m32cl         M32C/80, M16C/80, M16C/70 Linker           m32ci         M32C/80, M16C/80, M16C/70 Librarian           m32ce         M32C/80, M16C/80, M16C/70 C Compiler           m32cl         M32C/80, M16C/80, M16C/70 mr308tbl (MR308)           m32rl         M32R C Compiler           m32rl         M32R Load module converter           m32ri         M32R Load module converter           m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Linker           74lb         740 Linker           100i         R32C/100 Librarian           100c         R32C/100 C Compiler           100t         R32C/100 mr100tbl (MR	shg	SuperH RISC engine C/C++ Library Generator
h8g         H8S, H8/300 C/C++ Library Generator           h8c         H8S, H8/300 C/C++ Compiler           h8a         H8S, H8/300 Assembler           hlk         OptLinker           m16cl         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Linker           m16ci         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Librarian           m16cc         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny C Compiler           m16ct         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny mkmrtbl (MR30)           m32cl         M32C/80, M16C/80, M16C/70 Linker           m32ci         M32C/80, M16C/80, M16C/70 Librarian           m32cc         M32C/80, M16C/80, M16C/70 C Compiler           m32ct         M32C/80, M16C/80, M16C/70 mr308tbl (MR308)           m32rl         M32R C Compiler           m32rl         M32R Librarian           m32re         M32R Load module converter           m32ra         M32R Assembler           74lk         740 Librarian           100l         R32C/100 Linker           74lb         740 Librarian           100c         R32C/100 Linker           100t         R32C/100 C Compiler           100t         R32C/100 Trioutbl (MR100)           rxg         RX C/C++ Compiler     <	shc	SuperH RISC engine C/C++ Compiler
h8c         H8S, H8/300 C/C++ Compiler           h8a         H8S, H8/300 Assembler           hlk         OptLinker           m16cl         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Linker           m16ci         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Librarian           m16cc         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny C Compiler           m16ct         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny mkmrtbl (MR30)           m32cl         M32C/80, M16C/80, M16C/70 Linker           m32ci         M32C/80, M16C/80, M16C/70 C Compiler           m32ct         M32C/80, M16C/80, M16C/70 c Compiler           m32rl         M32R C Compiler           m32rl         M32R C Compiler           m32ri         M32R Librarian           m32ra         M32R Load module converter           m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Librarian           1001         R32C/100 Librarian           1002         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxa         RX Assembler	sha	SuperH RISC engine Assembler
h8a         H8S, H8/300 Assembler           hlk         OptLinker           m16cl         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Linker           m16ci         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Librarian           m16cc         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny C Compiler           m16ct         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny mkmrtbl (MR30)           m32cl         M32C/80, M16C/80, M16C/70 Linker           m32ci         M32C/80, M16C/80, M16C/70 Librarian           m32cc         M32C/80, M16C/80, M16C/70 C Compiler           m32cl         M32C/80, M16C/80, M16C/70 mr308tbl (MR308)           m32rl         M32R C Compiler           m32rl         M32R Librarian           m32rm         M32R Load module converter           m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Librarian           1001         R32C/100 Linker           1001         R32C/100 Librarian           1002         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxa         RX Assembler	h8g	H8S, H8/300 C/C++ Library Generator
hlk         OptLinker           m16cl         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Linker           m16ci         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Librarian           m16cc         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny C Compiler           m16ct         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny mkmrtbl (MR30)           m32cl         M32C/80, M16C/80, M16C/70 Linker           m32ci         M32C/80, M16C/80, M16C/70 Librarian           m32cc         M32C/80, M16C/80, M16C/70 C Compiler           m32ct         M32C/80, M16C/80, M16C/70 mr308tbl (MR308)           m32rl         M32R C Compiler           m32ri         M32R Librarian           m32re         M32R Load module converter           m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Librarian           1001         R32C/100 Linker           1002         R32C/100 Liotarian           1003         R32C/100 Librarian           1004         R32C/100 Tempiler           7xc         RX C/C++ Library Generator           7xa         RX Assembler	h8c	H8S, H8/300 C/C++ Compiler
m16cl         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Linker           m16ci         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Librarian           m16cc         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny C Compiler           m16ct         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny mkmrtbl (MR30)           m32cl         M32C/80, M16C/80, M16C/70 Linker           m32ci         M32C/80, M16C/80, M16C/70 Librarian           m32cc         M32C/80, M16C/80, M16C/70 C Compiler           m32ct         M32C/80, M16C/80, M16C/70 mr308tbl (MR308)           m32rl         M32R C Compiler           m32ri         M32R Librarian           m32re         M32R Load module converter           m32re         M32R Assembler           74lk         740 Linker           74lk         740 Librarian           1001         R32C/100 Linker           1002         R32C/100 Librarian           1003         R32C/100 Librarian           1004         R32C/100 R32C/100 Librarian           1005         R32C/100 C Compiler           1006         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxa         RX Assembler	h8a	H8S, H8/300 Assembler
m16ci         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny Librarian           m16cc         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny C Compiler           m16ct         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny mkmrtbl (MR30)           m32cl         M32C/80, M16C/80, M16C/70 Librarian           m32ci         M32C/80, M16C/80, M16C/70 C Compiler           m32ct         M32C/80, M16C/80, M16C/70 mr308tbl (MR308)           m32rl         M32R C Compiler           m32ri         M32R Librarian           m32rm         M32R Load module converter           m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Librarian           100l         R32C/100 Linker           100i         R32C/100 Librarian           100c         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxc         RX C/C++ Compiler           rxa         RX Assembler	hlk	OptLinker
m16cc         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny C Compiler           m16ct         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny mkmrtbl (MR30)           m32cl         M32C/80, M16C/80, M16C/70 Linker           m32ci         M32C/80, M16C/80, M16C/70 Librarian           m32cc         M32C/80, M16C/80, M16C/70 C Compiler           m32ct         M32C/80, M16C/80, M16C/70 mr308tbl (MR308)           m32rl         M32R C Compiler           m32ri         M32R Librarian           m32rm         M32R Load module converter           m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Librarian           100l         R32C/100 Linker           100i         R32C/100 Librarian           100c         R32C/100 C Compiler           100t         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxc         RX C/C++ Compiler           rxa         RX Assembler	m16cl	
m16ct         M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny mkmrtbl (MR30)           m32cl         M32C/80, M16C/80, M16C/70 Linker           m32ci         M32C/80, M16C/80, M16C/70 Librarian           m32cc         M32C/80, M16C/80, M16C/70 C Compiler           m32ct         M32C/80, M16C/80, M16C/70 mr308tbl (MR308)           m32rl         M32R C Compiler           m32ri         M32R Librarian           m32rm         M32R Load module converter           m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Librarian           1001         R32C/100 Linker           100i         R32C/100 Librarian           100c         R32C/100 C Compiler           100t         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxc         RX C/C++ Compiler           rxa         RX Assembler	m16ci	
m32cl         M32C/80, M16C/80, M16C/70 Linker           m32ci         M32C/80, M16C/80, M16C/70 Librarian           m32cc         M32C/80, M16C/80, M16C/70 C Compiler           m32ct         M32C/80, M16C/80, M16C/70 mr308tbl (MR308)           m32rl         M32R C Compiler           m32ri         M32R Librarian           m32rm         M32R Load module converter           m32re         M32R Assembler           74lk         740 Linker           74lk         740 Librarian           100l         R32C/100 Librarian           100i         R32C/100 Librarian           100c         R32C/100 C Compiler           100t         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxc         RX C/C++ Compiler           rxa         RX Assembler	m16cc	
m32ci         M32C/80, M16C/80, M16C/70 Librarian           m32cc         M32C/80, M16C/80, M16C/70 C Compiler           m32ct         M32C/80, M16C/80, M16C/70 mr308tbl (MR308)           m32rl         M32R C Compiler           m32ri         M32R Librarian           m32rm         M32R Load module converter           m32rc         M32R C Compiler           m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Librarian           100l         R32C/100 Linker           100i         R32C/100 Librarian           100c         R32C/100 T C Compiler           100t         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxc         RX C/C++ Compiler           rxa         RX Assembler	m16ct	M16C/60, M16C/30, M16C/Tiny, M16C/20, M16C/10, R8C/Tiny mkmrtbl (MR30)
m32cc         M32C/80, M16C/80, M16C/70 C Compiler           m32ct         M32C/80, M16C/80, M16C/70 mr308tbl (MR308)           m32rl         M32R C Compiler           m32ri         M32R Librarian           m32rm         M32R Load module converter           m32rc         M32R C Compiler           m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Librarian           100l         R32C/100 Linker           100i         R32C/100 Librarian           100c         R32C/100 C Compiler           100t         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxc         RX C/C++ Compiler           rxa         RX Assembler	m32cl	M32C/80, M16C/80, M16C/70 Linker
m32ct         M32C/80, M16C/80, M16C/70 mr308tbl (MR308)           m32rl         M32R C Compiler           m32ri         M32R Librarian           m32rm         M32R Load module converter           m32rc         M32R C Compiler           m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Librarian           100l         R32C/100 Linker           100i         R32C/100 Librarian           100c         R32C/100 C Compiler           100t         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxc         RX C/C++ Compiler           rxa         RX Assembler	m32ci	
m32rl         M32R C Compiler           m32ri         M32R Librarian           m32rm         M32R Load module converter           m32rc         M32R C Compiler           m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Librarian           100l         R32C/100 Linker           100i         R32C/100 Librarian           100c         R32C/100 C Compiler           100t         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxc         RX C/C++ Compiler           rxa         RX Assembler	m32cc	·
m32ri         M32R Librarian           m32rm         M32R Load module converter           m32rc         M32R C Compiler           m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Librarian           100l         R32C/100 Linker           100i         R32C/100 Librarian           100c         R32C/100 C Compiler           100t         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxc         RX C/C++ Compiler           rxa         RX Assembler	m32ct	M32C/80, M16C/80, M16C/70 mr308tbl (MR308)
m32rm         M32R Load module converter           m32rc         M32R C Compiler           m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Librarian           100l         R32C/100 Linker           100i         R32C/100 Librarian           100c         R32C/100 C Compiler           100t         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxc         RX C/C++ Compiler           rxa         RX Assembler	m32rl	· ·
m32rc         M32R C Compiler           m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Librarian           100l         R32C/100 Linker           100i         R32C/100 Librarian           100c         R32C/100 C Compiler           100t         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxc         RX C/C++ Compiler           rxa         RX Assembler	m32ri	M32R Librarian
m32ra         M32R Assembler           74lk         740 Linker           74lb         740 Librarian           100l         R32C/100 Linker           100i         R32C/100 Librarian           100c         R32C/100 C Compiler           100t         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxc         RX C/C++ Compiler           rxa         RX Assembler	m32rm	M32R Load module converter
74lk       740 Linker         74lb       740 Librarian         100l       R32C/100 Linker         100i       R32C/100 Librarian         100c       R32C/100 C Compiler         100t       R32C/100 mr100tbl (MR100)         rxg       RX C/C++ Library Generator         rxc       RX C/C++ Compiler         rxa       RX Assembler	m32rc	·
74lb       740 Librarian         100l       R32C/100 Linker         100i       R32C/100 Librarian         100c       R32C/100 C Compiler         100t       R32C/100 mr100tbl (MR100)         rxg       RX C/C++ Library Generator         rxc       RX C/C++ Compiler         rxa       RX Assembler	m32ra	M32R Assembler
100I       R32C/100 Linker         100i       R32C/100 Librarian         100c       R32C/100 C Compiler         100t       R32C/100 mr100tbl (MR100)         rxg       RX C/C++ Library Generator         rxc       RX C/C++ Compiler         rxa       RX Assembler	74lk	740 Linker
100i         R32C/100 Librarian           100c         R32C/100 C Compiler           100t         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxc         RX C/C++ Compiler           rxa         RX Assembler	74lb	740 Librarian
100c         R32C/100 C Compiler           100t         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxc         RX C/C++ Compiler           rxa         RX Assembler	1001	R32C/100 Linker
100t         R32C/100 mr100tbl (MR100)           rxg         RX C/C++ Library Generator           rxc         RX C/C++ Compiler           rxa         RX Assembler	100i	R32C/100 Librarian
rxg RX C/C++ Library Generator rxc RX C/C++ Compiler rxa RX Assembler		·
rxc RX C/C++ Compiler rxa RX Assembler	100t	· · · · · · · · · · · · · · · · · · ·
rxa RX Assembler	rxg	
	rxc	·
r600t RX mkritbl (RI600)	rxa	
	r600t	RX mkritbl (RI600)

# 3. ツールチェインアップグレード

以下に示すツールチェインをプロジェクトで使用している場合、ツールチェインの新バージョンをインストールすることで、使用しているツールチェインをアップグレード(変更)できます。

- SuperH ファミリ用 C/C++コンパイラパッケージ V.5.1 およびそれ以降
- H8SX, H8S, H8 ファミリ用 C/C++コンパイラパッケージ V.3.0A およびそれ以降
- M16C シリーズ用 C コンパイラパッケージ V.5.20 Release 1 およびそれ以降
- M32C シリーズ用 C コンパイラパッケージ V.5.20 Release 1 およびそれ以降
- R32C シリーズ用 C コンパイラパッケージ V.1.01 Release 00 およびそれ以降
- M32R ファミリ用 C/C++コンパイラパッケージ V.4.20 Release 1 およびそれ以降
- 740 ファミリ用 C コンパイラパッケージ V.1.01 Release 01 およびそれ以降
- 740 ファミリ用アセンブラパッケージ V.4.10 Release 02 およびそれ以降
- (1) SuperH ファミリ用 C/C++コンパイラパッケージ V.5.1x、または H8SX, H8S, H8 ファミリ用 C/C++コンパイラパッケージ V.3.0x からのアップグレードの場合
- (a) 標準ライブラリ構築ツールの設定

標準ライブラリ構築ツールが追加されました。

High-performance Embedded Workshop はアップグレード時、IM OptInker のオプション情報をもとに、標準ライブラリ入力指定があればライブラリを生成するオプション( Mode: Build a library file )を設定します。標準ライブラリ入力指定がなければ、ライブラリを生成しないオプション( Mode: Do not add a library file )を設定します。

## (b) 最適化リンカの設定

IM Optlnker、Librarian、Stype Converter を統合し、OptLinker になりました。 アップグレード前の各ツールの登録状態により、アップグレード後の OptLinker のオプション設定が異なります。

表 3.1 にアップグレード時のオプション継承を示します。

表 3.1 アップグレード時のオプション継承

		アップグレード前			アップグレード後				
No	IM OptLinker の状態	Librarian の状態	Stype Converter の状態	Optinker の状態	IM OptLinker オプション情報	Librarian オプション情報	Stype Converter オプション情報		
1	_	_	_	_	×	×	×		
2	_	_			×	×	0		
3	_	_		•	×	×	0		
4	_		_		×	0	×		
5	_				×	0	×		
6	_				×	×	0		
7	_		_		×	0	×		
8	_			•	×	0	×		
9	_				×	0	×		
10		_	_		0	×	×		
11		_			0	×	0		
12		_			×	×	0		
13			_		0	×	×		
14					0	×	0		
15					×	×	0		
16			_		×	0	×		
17				•	×	0	×		
18				•	×	0	×		
19		_	_		0	×	×		
20		_			0	×	×		
21		_			0	×	0		
22			_	•	0	×	×		
23				•	0	×	×		
24				•	0	×	0		
25	•		_	•	0	×	×		
26	•			•	0	×	×		
27					0	×	0		

:未登録、 : 登録チェック無し、 : 登録チェック有り、 : 継承、x:継承しない

- (2) SuperH ファミリ用 C/C++コンパイラパッケージのアップグレードの場合
- (a) MAP 最適化用ビルドの自動化

High-performance Embedded Workshop V.2.00 (SH Ver.7.0B / Ver.7.0.01 / Ver.7.0.02) では、C コンパイラで最適化リンケージエディタが出力した外部シンボル割り付け情報を活用した最適化を実行するために、再ビルド用のカスタムフェーズを提供していました。

High-performance Embedded Workshop V.2.01 以降では、MAP 最適化実行時に自動的に再ビルド実行が可能になりました。

再ビルド用のカスタムフェーズは不要になるため、図3.1で示すように、[ビルド -> ビルドフェーズ]で[ビルドフェーズ]ダイアログボックスを開き、"Map optimize"フェーズのチェックを外してください。



図 3.1 [ビルドフェーズ]ダイアログボックス

- 4. Microsoft® Windows Vista® およびWindows® 7 互換性対応
- 4.1 Windows Vista® およびWindows® 7 における標準権限での使用について

Windows Vista® および Windows® 7 における標準権限での使用について High-performance Embedded Workshop V.4.05.00 以降の使用上の注意事項を以下に示します。

## 4.1.1 内容

High-performance Embedded Workshop および High-performance Embedded Workshop システム下のすべてのソフトウェアツール製品が Windows Vista® または Windows® 7 対応版であるにもかかわらず、管理者権限の要求画面が表示される(標準権限で動作しない)場合があります。

これは、V.4.04.01 以前の High-performance Embedded Workshop を Windows Vista® または Windows® 7 で動作させるために管理者権限で実行するプログラムの互換設定をして、その後 Windows Vista® 対応版の High-performance Embedded Workshop V.4.05.00 以降、または Windows® 7 対応版の High-performance Embedded Workshop V.4.08.00 以降にアップデートした場合に起こります。

### 補足:

High-performance Embedded Workshop は、以下の2点を満たす場合に、Windows Vista® またはWindows®7において標準権限で使用できます。

- (1) Windows Vista® 対応版(V.4.05.00 以降)、または Windows® 7 対応版(V.4.08.00 以降)である。
- (2) High-performance Embedded Workshop システム下で使用するすべてのソフトウェア製品が Windows Vista® または Windows® 7 対応版である。

上記以外の場合、およびソフトウェア製品のインストール時には管理者権限を必要とします。

### 4.1.2 回避策

High-performance Embedded Workshop システム下のすべてのソフトウェア製品を Windows Vista® または Windows® 7 対応版にしているにもかかわらず、管理者権限の要求画面が表示されている場合には、以下の方法で、互換モードを適用しない設定にしてください。

High-performance Embedded Workshop のショートカットのプロパティを開き、「互換性」タブで以下の 2 つのチェックボックスのチェックをはずして無効にする。

- 互換モードでこのプログラムを実行する
- 管理者としてこのプログラムを実行する

すべての商標および登録商標は,それぞれの所有者に帰属します。

## ご注意書き

- 1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
- 2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
- 3. 当社製品を改造、改変、複製等しないでください。
- 4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
- 5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されて いる当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の 法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
- 6. 本資料に記載されている情報は、正確を期すため慎重に作成したものですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報 の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
- 7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。

標準水準: コンピュータ、OA機器、通信機器、計測機器、AV機器、家電、工作機械、パーソナル機器、産業用ロボット

高品質水準: 輸送機器(自動車、電車、船舶等)、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器

(厚生労働省定義の管理医療機器に相当)

特定水準: 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器(生命維持装置、人体に埋め込み使用するもの、治療

行為(患部切り出し等)を行うもの、その他直接人命に影響を与えるもの)(厚生労働省定義の高度管理医療機器に相当)またはシステム等

- 8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
- 9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
- 10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制するRoHS指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
- 11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
- 12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。
- 注1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社がその総株主の議決権の過半数を直接または間接に保有する会社をいいます。
- 注2. 本資料において使用されている「当社製品」とは、注1において定義された当社の開発、製造製品をいいます。



■営業お問合せ窓口

http://www.renesas.com

※営業お問合せ窓口の住所・電話番号は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス販売株式会社 〒100-0004 千代田区大手町2-6-2 (日本ビル)

(03)5201-5307

	技術的	かお問合	サおよ	-7K 容料σ	ご請求は	下記へと	ごう	ギ
_	17 113 11 7.	Q-02 [H] [H	_ 00 0	· O 5-21-10.	CIHATIO	1 40 . C	. /	- 0
	総合お	問合せ窓	口:hf	ttp://japar	n.renesas.	com/inq	uiry	/