お客様各位

カタログ等資料中の旧社名の扱いについて

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願い申し上げます。

ルネサスエレクトロニクス ホームページ（http://www.renesas.com）

2010年4月1日
ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社（http://www.renesas.com）

【問い合わせ先】http://japan.renesas.com/inquiry

RENESAS

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。

User's Manual

RENESAS

High-performance Embedded Workshop V.4.02

ユーザーズマニュアル

ルネサスマイクロコンピュータ開発環境システム

ルネサス エレクトロニクス
www.renesas.com

Rev.1.00　2006.12

安全設計に関するお願い

- 弊社は品質、信頼性の向上に努めておりますが、半導体製品は故障が発生したり、誤動作する場合があります。弊社の半導体製品の故障又は誤動作によって結果として、人身事故火災事故、社会的損害などを生じさせないような安全性を考慮した冗長設計、延焼対策設計、誤動作防止設計などの安全設計に十分ご留意ください。

本資料ご利用に際しての留意事項

- 本資料は、お客様が用途に応じた適切なルネサス テクノロジ製品をご購入いただくための参考資料であり、本資料中に記載の技術情報について株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズが所有する知的財産権その他の権利の実施、使用を許諾するものではありません。
- 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例の使用に起因する損害、第三者所有の権利に対する侵害に関し、株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズは責任を負いません。
- 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他全ての情報は本資料発行時点のものであり、株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズは、予告なしに、本資料に記載した製品又は仕様を変更することがあります。ルネサス テクノロジ半導体製品のご購入に当たりましては、事前に株式会社ルネサス テクノロジ、株式会社ルネサス ソリューションズ、株式会社ルネサス販売又は特約店へ最新の情報をご確認頂きますとともに、ルネサス テクノロジホームページ（http://www.renesas.com）などを通じて公開される情報に常にご注意ください。
- 本資料に記載した情報は、正確を期すため、慎重に制作したものですが万一本資料の記述誤りに起因する損害がお客様に生じた場合には、株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズはその責任を負いません。
- 本資料に記載の製品データ、図、表に示す技術的な内容、プログラム及びアルゴリズムを流用する場合は、技術内容、プログラム、アルゴリズム単位で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断してください。株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズは、適用可否に対する責任は負いません。
- 本資料に記載された製品は、人命にかかわるような状況の下で使用される機器あるいはシステムに用いられることを目的として設計、製造されたものではありません。本資料に記載の製品を運輸、移動体用、医療用、航空宇宙用、原子力制御用、海底中継用機器あるいはシステムなど、特殊用途へのご利用をご検討の際には、株式会社ルネサス テクノロジ、株式会社ルネサス ソリューションズ、株式会社ルネサス販売又は特約店へご照会ください。
- 本資料の転載、複製については、文書による株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズの事前の承諾が必要です。
- 本資料に関し詳細についてのお問い合わせ、その他お気付きの点がございましたら株式会社ルネサス テクノロジ、株式会社ルネサス ソリューションズ、株式会社ルネサス販売又は特約店までご照会ください。

製品の内容及び本書についてのお問い合わせ先

インストーラが生成する以下のテキストファイルに必要事項を記入の上、コンタクトセンタ csc@renesas.comまで送信ください。

¥SUPPORT¥製品名¥SUPPORT.TXT

株式会社ルネサス テクノロジ

コンタクトセンタ	csc@renesas.com
ユーザ登録窓口	regist_tool@renesas.com
ホームページ	http://japan.renesas.com/tools

はじめに

High-performance Embedded Workshop は、ルネサステクノロジマイクロコントローラの組み込み用アプリケーションの開発をサポートするツールです。おもな特徴をまとめると次のようになります。

- GUI を用いてコンパイラ、アセンブラ、リンケージエディタなどのオプション設定のカスタマイズができるプロジェクトビルドシステム
- プログラムを読みやすくするシンタックス色付け機能を持つ統合テキストエディタ
- ユーザ独自のツールを実行するための環境設定
- 同一アプリケーション内でのビルドおよびデバッグを可能にする統合化デバッガ
- バージョン管理サポート

High-performance Embedded Workshop は 2 つの目的で設計されています。1 つはユーザに開発ツールを提供すること、そしてもう 1 つは、それらのツール類を統合して使いやすくすることです。

このユーザーズマニュアルについて

High-performance Embedded Workshop システムについて述べています。High-performance Embedded Workshop の基本的な使い方に関する情報、High-performance Embedded Workshop 環境のカスタマイズ、High-performance Embedded Workshop のビルド機能、および各 High-performance Embedded Workshop 製品で共通なデバッグ機能について説明します。なお、SuperH RISC engine ファミリ用のデバッギングプラットフォームと連携して動作した際の図を使用しています。

デバッギングプラットフォームとは、統合開発環境 High-performance Embedded Workshop と連携して動作するエミュレータ、またはシミュレータのデバッガを指します。

デバッギングプラットフォームの詳細については、各製品に同梱されるエミュレータまたはシミュレータのユーザーズマニュアルおよびヘルプを参照してください。

C/C++言語、アセンブリ言語の書き方や、オペレーティングシステムの使い方、個々のデバイスに適したプログラムの書き方などについては説明していません。それらについては、各々のユーザーズマニュアルを参照してください。

High-performance Embedded Workshop は、インストール上、各種言語にカスタマイズされています。このユーザーズマニュアルでは、High-performance Embedded Workshop アプリケーションの日本語版について説明します。

注：

High-performance Embedded Workshop は Windows®バージョン 3.1 および Windows®95 では動作しません。

このユーザーズマニュアルの記号

このユーザーズマニュアルで使われている記号の意味を説明します。

記号	意味
[メニュー->メニューオプション]	'->' はメニューオプションを示します (例 [ファイル->名前を付けて保存])。
FILENAME.C	大文字の名前はファイル名を示します。
“文字列の入力”	下線は入力する文字列を示します (“” を省く)。
Key + Key	キー入力を示します。例えば、Ctrl+N キーでは Ctrl キーと N キーを同時に押します。

登録商標

Microsoft, MS-DOS, Windows, Windows NT は米国 Microsoft 社の米国およびその他の国における登録商標です。

Visual SourceSafe は Microsoft 社の米国およびその他の国における商標です。

IBM および AT は International Business Machines Corporation の登録商標です。

Intel,Pentium は米国 Intel Corporation の登録商標です。

その他、記載されている製品名は各社の商標または登録商標です。

目次

1. 概要 1
1.1 ワークスペース、プロジェクト、ファイル 1
1.2 メインウィンドウ 2
1.2.1 タイトルバー 2
1.2.2 メニューバー 2
1.2.3 ツールバー 3
1.2.4 ワークスペースウィンドウ 6
1.2.5 エディタウィンドウ 8
1.2.6 アウトプットウィンドウ 9
1.2.7 ステータスバー 13
1.3 ヘルプ機能 13
1.4 HEW を起動する 14
1.5 新規ワークスペースを作成する 15
1.6 ワークスペースを開く 15
1.7 古いワークスペースを使用する 16
1.8 ワークスペースを保存する 16
1.9 ワークスペースを閉じる 16
1.10 HEW を終了する 16
1.11 ツールシステム概要 17
1.12 コンフィグレーションおよびセッションの概要 17
1.13 マクロ生成支援機能とテスト支援機能の概要 21
1.13.1 テストシナリオの例 25
1.13.2 ステップ 1： マクロの記録 27
1.13.3 ステップ 2： マクロを編集する（記録内容を見る） 29
1.13.4 ステップ 3： マクロを実行する 30
1.13.5 ステップ 4： テストスイートを作成する 30
1.13.6 ステップ 5： テストスイートを編集する 31
1.13.7 ステップ 6： テストイメージファイルを作成する 33
1.13.8 ステップ 7： プログラムを変更しテストを実行する 35
1.13.9 ステップ 8： テスト結果を確認する（不一致の場合） 36
1.13.10 ステップ 9： プログラムを元に戻しテストを実行する 37
1.13.11 ステップ 10： テスト結果を確認する（一致した場合） 37
2. ビルドの基本 39

2.1 ビルド処理 39
2.2 ワークスペースウィンドウの Projects タブの構成 40
2.3 プロジェクトファイル 43
2.3.1 プロジェクトにファイルを追加する 43
2.3.2 ファイルおよびフォルダをドラッグアンドドロップする 44
2.3.3 プロジェクトからファイルを削除する 46
2.3.4 ビルドからプロジェクトファイルを除外する 47
2.3.5 除外したプロジェクトファイルをビルドに再び入れる 48
2.4 ワークスペースウィンドウでユーザフォルダを使う 48
2.5 ファイル拡張子とファイルグループ 49
2.5.1 アプリケーションとファイルグループを関連付けるには 50
2.5.2 新規ファイルグループに新規ファイル拡張子を登録する 52
2.5.3 新規ファイル拡張子を作成する 53
2.6 ファイルのビルド方法を設定する 54
2.7 ビルドのコンフィグレーション 55
2.7.1 ビルドコンフィグレーションを選択する 55
2.7.2 新規ビルドコンフィグレーションを追加する 56
2.7.3 ビルドコンフィグレーションを削除する 56
2.8 プロジェクトをビルドする 56
2.8.1 ファイルをコンパイルする 56
2.8.2 プロジェクトをビルドする 57
2.8.3 複数のプロジェクトをビルドする 57
2.8.4 ツールの実行を中止する 58
2.8.5 ビルドの中間ファイルおよび出力ファイルを削除する 59
2.8.6 アウトプットウィンドウの Build タブの構成 60
2.8.7 アウトプットウィンドウの Build タブの内容の制御 62
2.8.8 ビルド対象ファイルをワークスペースウィンドウにマーク表示する 62
2.9 ファイル依存関係 63
2.10 表示方法を指定する 66
2.11 ワークスペースにプロジェクトを追加する 68
2.12 アクティブプロジェクトを設定する 69
2.13 プロジェクト間の依存関係を指定する 70
2.14 ワークスペースからプロジェクトを削除する 71
2.15 ワークスペースのプロジェクト相対パス 71
3. ビルドの応用 73
3.1 ビルド実行の復習 73
3.1.1 ビルドとは? 73

3.2 カスタムビルドフェーズを作成する 74
3.3 ビルドフェーズの順序 78
3.3.1 ビルド順序タブ 79
3.3.2 ファイルのビルド順序タブ 82
3.3.3 ファイルマッピングタブ 83
3.4 カスタムビルドフェーズのオプション設定 84
3.4.1 オプションタブ 85
3.4.2 出力ファイルタブ 86
3.4.3 依存ファイルタブ 86
3.5 ビルドを管理する 87
3.6 ビルドの出力のログを取る 88
3.7 ツールチェインのバージョンを変更する 89
3.8 Make ファイルの生成 89
3.9 Make ファイルを使ったビルド 92
3.10 リンク順序をカスタマイズする 93
4. エディタの使用 95
4.1 エディタウィンドウ 95
4.2 複数のファイルを扱う 96
4.3 標準のファイル操作 97
4.3.1 新規ファイルを作成する 97
4.3.2 ファイルを編集する 97
4.3.3 ファイルを保存する 98
4.3.4 ファイルを開く 99
4.3.5 ファイルを閉じる 99
4.4 検索とファイル内の移動 101
4.4.1 テキストの検索 101
4.4.2 複数のファイル間でのテキスト検索 101
4.4.3 テキストを置換する 102
4.4.4 指定した行にジャンプする 103
4.5 ブックマーク 104
4.6 ファイルを印刷する 105
4.7 テキストのレイアウト 105
4.7.1 ページ設定 105
4.7.2 タブを変更する 106
4.7.3 自動インデント 107
4.8 ウィンドウを分割する 107
4.9 エディタのフォントを変更する 108

4.10 シンタックスを色付けする 109
4.10.1 色の変更 109
4.10.2 新規キーワードの作成 110
4.10.3 シンタックスの色付けを有効/無効にする 111
4.11 テンプレート 111
4.11.1 テンプレートを設定する 112
4.11.2 テンプレートを削除する 113
4.11.3 テンプレートを挿入する 114
4.12 括弧の組み合わせ 114
4.13 ファイルを読み取り専用属性にする 115
4.14 デバッグ時にファイルの編集を抑止する 115
4.15 エディタカラムの管理 116
4.16 カラムヘッダの表示/非表示を切り替える 117
4.17 エディタ内からファイルを開く 117
4.18 ツールチップウォッチ 117
4.19 式の評価 118
5. ツール管理 121
5.1 ツールの位置 122
5.2 HEW 登録ファイル 122
5.3 ツールを登録する 123
5.4 ツールの登録を取り消す 124
5.5 ツールのプロパティの参照と編集 125
5.6 ツールのアンインストール 127
5.7 テクニカルサポートについて 128
5.8 オンデマンドのコンポーネント 130
5.9 カスタムプロジェクトタイプ 131
6. 環境のカスタマイズ 133
6.1 ツールバーのカスタマイズ 133
6.2 ツールメニューのカスタマイズ 135
6.3 カスタムプレースホルダを使う 137
6.4 ワークスペースやプロジェクトのログ機能を使う 138
6.5 ヘルプシステムを構築する 138
6.6 キーボードショートカットをカスタマイズする 139
6.7 基本設定の影響範囲 141
6.7.1 カスタマイズダイアログボックスの影響範囲 141
6.7.2 オプションダイアログボックスの影響範囲 142
6.8 ワークスペースオプションを指定する 142

6.8.1 起動時に最後に開いたワークスペースを開く 142
6.8.2 ワークスペースを開いたときにファイルを表示する 142
6.8.3 ワークスペースを開いたときにワークスペース情報を表示する 143
6.8.4 ツール実行前にワークスペースを保存する 143
6.8.5 ワークスペース保存前に確認する 143
6.8.6 セッション保存前に確認する 144
6.8.7 自動バックアップ機能を有効にする 144
6.8.8 新規ワークスペースのデフォルトディレクトリを変更する 145
6.9 HEW エディタ以外のエディタを使う 146
6.10 表示するフォントのカスタマイズ 148
6.11 バーチャルデスクトップを使用する 149
7. バージョン管理 151
7.1 バージョン管理システムを選択する 153
7.2 バージョン管理設定内容のインポートとエクスポート 154
8. カスタムバージョン管理システムの使用 155
8.1 バージョン管理メニューオプションを定義する 155
8.1.1 システム定義メニューオプション 156
8.1.2 ユーザ定義メニューオプション 157
8.2 バージョン管理コマンドを定義する 158
8.3 引数を指定する 160
8.4 コメントを指定する 160
8.5 バージョン管理コマンドリターンコードオプション 160
8.6 ディレクトリをマッピングする 161
8.7 ディレクトリマッピング例 163
8.8 バージョン管理環境変数を設定する 164
8.9 バージョン管理システムの実行を制御する 164
8.10 ユーザ名とパスワードを変更する 165
8.11 カスタムバージョン管理システムの使用例 166
8.11.1 HEW と RCS を接続して使用する 166
8.11.2 HEW と CVS を接続して使用する 173
9. Visual SourceSafe の使用 181
9.1 ワークスペースに Visual SourceSafe を関連付ける 181
9.2 Visual SourceSafe コマンド 184
9.2.1 Visual SourceSafe にファイルを追加する 184
9.2.2 Visual SourceSafe からファイルを削除する 186
9.2.3 Visual SourceSafe からファイルの読み取り専用コピーを取得する 186
9.2.4 Visual SourceSafe からファイルの書き込み可能コピーをチェックアウトする 187

9.2.5 Visual SourceSafe にファイルの書き込み可能コピーをチェックインする 188
9.2.6 Visual SourceSafe のファイルの状態を表示する 189
9.2.7 Visual SourceSafe のチェックアウトコマンドを取り消す 190
9.2.8 Visual SourceSafe のファイル履歴を表示する 191
9.3 Visual SourceSafe コマンドのオプション 191
9.4 バージョン管理設定内容 192
9.5 ディレクトリをマッピングする 192
9.6 Visual SourceSafe コマンドを追加する 194
10. ネットワークを利用したプロジェクト共有機能 197
10.1 Windows XP Service Pack 2 を搭載した PC で本機能を使う場合の設定 198
10.2 ネットワークを利用したプロジェクト共有機能を使う 201
10.3 アドミニストレータのパスワードを設定する 201
10.4 新規ユーザを追加する 202
10.5 パスワードを変更する 203
10.6 ネットワークを利用したプロジェクト共有機能を使う場合の留意事項 204
11. ファイルの比較 205
11.1 差分ウィンドウを開く 205
12. ナビゲーション機能 209
12.1 C 関数と#define ナビゲーションの構成 211
12.2 C++ナビゲーションの構成 212
12.3 エディタ内から定義へジャンプする 214
12.4 ナビゲーション項目をドラッグアンドドロップする 215
12.5 スマートエディタ 216
13. マップ 219
13.1 セクション設定を管理する 220
13.1.1 Map Section Information ウィンドウを開く 221
13.1.2 セクション情報編集モードの開始と終了 223
13.1.3 セクショングループを追加する 224
13.1.4 セクションを追加する 224
13.1.5 オーバレイグループを追加する 225
13.1.6 選択項目を編集する 226
13.1.7 プライマリセクションに設定する 226
13.1.8 メモリマップを設定する 227
13.1.9 メモリリソースを自動的に確保する 227
13.1.10 セクション設定ツリーを印刷する 228
13.1.11 未割り当て領域を表示する 228
13.1.12 サイズ 0 のセクションを表示する 228

13.1.13 アドレスに対応するソースコードを表示する 228
13.1.14 セクションリストを印刷する 229
13.2 シンボル情報を見る 229
13.2.1 Map Symbol Information ウィンドウを開く 229
13.2.2 マップリストを印刷する 231
13.2.3 シンボルを検索する 231
13.2.4 シンボル情報をフィルタ表示する 232
13.2.5 アドレスに対応するソースコードを表示する 233
13.2.6 シンボル情報を印刷する 233
14. コマンドラインの使用 235
14.1 コマンドラインウィンドウを開く 235
14.2 コマンドファイルを設定する 237
14.3 コマンドファイルを実行する 237
14.4 コマンド実行を中断する 238
14.5 ログファイルを設定する 238
14.6 ログファイルへの出力を開始/停止する 238
14.7 ファイルのフルパスを入力する 238
14.8 プレースホルダを挿入する 239
14.9 ウィンドウの表示内容をすべて選択する 239
14.10 選択部分をクリップボードにコピーする 239
14.11 選択部分をクリップボードに切り取る 239
14.12 クリップボードの内容を貼り付ける 239
14.13 ウィンドウの表示内容をクリアする 239
14.14 直前の操作を元に戻す 239
14.15 括弧の呼応状態を確認する 240
15. マクロ生成支援機能の使用 241
15.1 マクロメニューとツールバー 241
15.2 マクロの設定ダイアログボックスを使用する 243
15.3 既存のマクロファイルをインポートする 244
15.4 マクロを記録する 245
15.5 マクロファイルにマクロを記録できる機能 246
15.5.1 マクロを記録できる機能（HEW 共通） 246
15.5.2 マクロを記録できる機能（デバッギングプラットフォーム依存） 249
15.6 マクロを実行する 252
15.7 マクロを編集する 252
15.8 マクロを割り当てる 253
15.9 アウトプットウィンドウの Macro タブの構成 254

16. テスト支援機能の使用 255
16.1 テストスイートを作成する 255
16.2 テストスイートを開く／閉じる 257
16.3 テストスイートを編集する 257
16.4 テストスイートにテストを追加する 259
16.5 テストイメージファイルを作成する 260
16.6 テストイメージファイルにテストイメージデータとして保存できる機能 262
16.6.1 テストイメージファイルに保存できる機能（HEW 共通） 262
16.6.2 テストイメージファイルに保存できる機能（デバッギングプラットフォーム依存） 267
16.7 テストイメージファイルを比較する 273
16.8 テストを実行する 274
16.9 テストブラウザを使用する 275
16.10 ワークスペースウィンドウの Test タブの構成 276
16.11 アウトプットウィンドウの Test タブの構成 277
17. デバッガの使用 279
17.1 デバッグの準備をする 279
17.1.1 デバッグの前にビルドを行う 279
17.1.2 デバッギングプラットフォームを選択する 279
17.1.3 プロジェクトの構成を編集する 290
17.1.4 デバッギングプラットフォームを構築する 291
17.1.5 ダウンロードモジュール 294
17.1.6 デバッグセッション 309
17.2 プログラムを表示する 315
17.2.1 エディタウィンドウを開く 316
17.2.2 逆アセンブリウィンドウを開く 323
17.2.3 現在の PC 位置を表示する 331
17.2.4 PC 位置の行を強調表示する 331
17.3 メモリを操作する 332
17.3.1 メモリウィンドウを開く 332
17.3.2 任意のアドレスにデータを設定する 334
17.3.3 アドレス領域を選択する 335
17.3.4 任意のアドレス領域を一定データで充填する 335
17.3.5 任意のアドレス領域のデータを別領域にコピーする 336
17.3.6 任意の 2 つのアドレス領域を比較する 337
17.3.7 任意のアドレス領域をテストする 338
17.3.8 任意のアドレス領域をテキストファイルに保存する 338
17.3.9 任意のアドレス領域のデータを検索する 339

17.3.10 表示開始アドレスを指定する 340
17.3.11 スクロール範囲を指定する 340
17.3.12 指定したレジスタが示すアドレスから表示する 341
17.3.13 スタックポインタ位置を追従する 341
17.3.14 ダウンロード時に表示するラベル位置を指定する 341
17.3.15 メモリ内容を手動更新する 341
17.3.16 メモリ内容の更新を抑止する 342
17.3.17 表示データ長を指定する 342
17.3.18 表示基数を指定する 342
17.3.19 表示コードを指定する 342
17.3.20 レイアウトを設定する 343
17.3.21 表示桁数を指定する 343
17.3.22 カバレッジ計測結果の表示/表示なしを切り替える 343
17.3.23 任意のアドレス領域を保存する 344
17.3.24 メモリにファイルをロードする 344
17.3.25 ウィンドウを分割表示する 345
17.3.26 アドレス領域を検証する 345
17.3.27 既存の色をカスタマイズする 346
17.4 メモリ内容を画像形式で表示する 346
17.4.1 画像ウィンドウを開く 346
17.4.2 ウィンドウを自動更新する 350
17.4.3 ウィンドウを更新する 350
17.4.4 ピクセル情報を表示する 350
17.5 メモリ内容を波形形式で表示する 350
17.5.1 波形ウィンドウを開く 351
17.5.2 ウィンドウを自動更新する 352
17.5.3 ウィンドウを更新する 352
17.5.4 拡大表示する 352
17.5.5 縮小表示する 352
17.5.6 最初のサイズに戻す 352
17.5.7 拡大/縮小倍率を設定する 353
17.5.8 横軸のサイズを設定する 353
17.5.9 カーソルを非表示にする 353
17.5.10 サンプリング情報を表示する 353
17.6 I/O レジスタを見る 353
17.6.1 IO ウィンドウを開く 354
17.6.2 I/O レジスタ表示を拡張する 355

17.6.3 ウィンドウを更新する 355
17.6.4 I/O ファイルをロードする 355
17.6.5 IO ウィンドウ表示内容を印刷する 356
17.6.6 IO ウィンドウ表示内容を保存する 356
17.6.7 I/O レジスタの内容を修正する 356
17.7 レジスタ内容を見る 356
17.7.1 レジスタウィンドウを開く 356
17.7.2 レジスタの表示基数を変更する 357
17.7.3 レジスタバンクを切り替える 358
17.7.4 レイアウトを設定する 358
17.7.5 表示するレジスタを選択する 359
17.7.6 レジスタの内容を設定する 360
17.7.7 フラグの値を設定する 360
17.7.8 ウィンドウを分割表示する 361
17.7.9 レジスタウィンドウの表示内容を保存する 361
17.7.10 レジスタの内容を使用する 361
17.7.11 値が変更された部分の色をカスタマイズする 361
17.8 ターゲットマイコンをリセットする 362
17.9 カーソル位置に PC 値を設定する 362
17.10 デバッギングプラットフォームを接続/接続解除する 362
17.11 デバッギングプラットフォームを初期化する 362
17.12 プログラムを実行する 362
17.12.1 実行を継続する 362
17.12.2 リセットから実行を開始する 363
17.12.3 カーソルまで実行する 363
17.12.4 条件を指定して実行する 364
17.12.5 リセット時に main 関数の先頭までプログラムを実行する 364
17.12.6 シングルステップ 365
17.12.7 複数のステップ 366
17.13 プログラムを停止する 367
17.13.1 停止ツールバーボタンによる停止 367
17.13.2 標準のブレークポイント（PC ブレークポイント） 367
17.14 現在の状態を表示する 368
17.15 関数呼び出し履歴を見る 369
17.15.1 スタックトレースウィンドウを開く 369
17.15.2 ソースコードを表示する 370
17.15.3 表示形式を設定する 370

17.16 外部デバッガを使う 371
17.16.1 日立デバッギングインタフェースを HEW と統合して構築する 371
17.16.2 PD デバッガを HEW と統合して構築する 372
17.16.3 外部デバッガを HEW と統合して構築する 372
17.17 複数デバッギングプラットフォームを同期動作させる 373
17.17.1 HEW アプリケーションでの同期 373
17.17.2 HEW デバッガターゲットの同期 374
17.18 デバッギングプラットフォームに依存するデバッグ機能 376
17.18.1 ラベルを見る 376
17.18.2 Elf/Dwarf2 のサポート 380
17.18.3 変数を表示する 383
18. テクニカルサポート 393
18.1 アップデートの確認 393
18.2 不具合レポートの作成 393

リファレンス 395

1. メインメニュー一覧 397
1.1 ファイルメニュー一覧 397
1.2 編集メニュー一覧 398
1.3 表示メニュー一覧 399
1.4 プロジェクトメニュー一覧 400
1.5 ビルドメニュー一覧 400
1.6 デバッグメニュー一覧 401
1.7 基本設定メニュー一覧 402
1.8 ツールメニュー一覧 403
1.9 テストメニュー一覧 403
1.10 ウィンドウメニュー一覧 404
1.11 ヘルプメニュー一覧 404
2. ウィンドウ一覧 405
3. コマンドライン一覧 407
3.1 コマンド一覧（アルファベット順） 407
3.2 コマンド一覧（機能別） 408
4. 正規表現 411
5. プレースホルダ 413
5.1 プレースホルダとは？ 413
5.2 プレースホルダを挿入する 413
5.3 使用できるプレースホルダ 415

5.4 プレースホルダを使うにあたって 416
6. I/O ファイルフォーマット 417
7. シンボルファイルフォーマット 419
8. キーボードのショートカット 421
9. HEW デバッガでドラッグアンドドロップする 425
10. ラベルを使用してコードを参照する 427
11. ウィンドウのツールバー 429
12. HEW1.x 用のツールチェインでビルドする 431
13. HMAKE ユーザガイド 433
13.1 コマンドライン 433
13.2 ファイルのシンタックス 433
13.3 記述部 434
13.4 コメント 436
13.5 メッセージコマンド 436

1. 概要

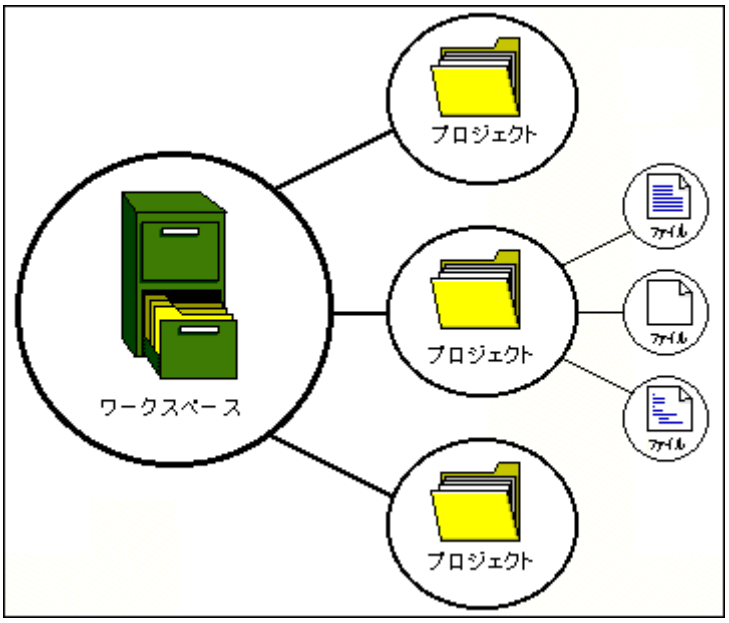
本ユーザーズマニュアルでは High-performance Embedded Workshop（以下、HEW と略します）バージョン V.4.02.00 の機能について記載しています。

この章では HEW の基本概念を説明します。

1.1 ワークスペース、プロジェクト、ファイル

ワードプロセッサでドキュメントを作成、修正できるのと同じように、HEW ではワークスペースを作成、修正できます。

ワークスペースはプロジェクトを入れる箱と考えることができます。同じように、プロジェクトはプロジェクトファイルを入れる箱と考えることができます。したがって各ワークスペースにはプロジェクトが 1 つ以上あり、各プロジェクトにはファイルが 1 つ以上あります。この構成を下の図に示します。

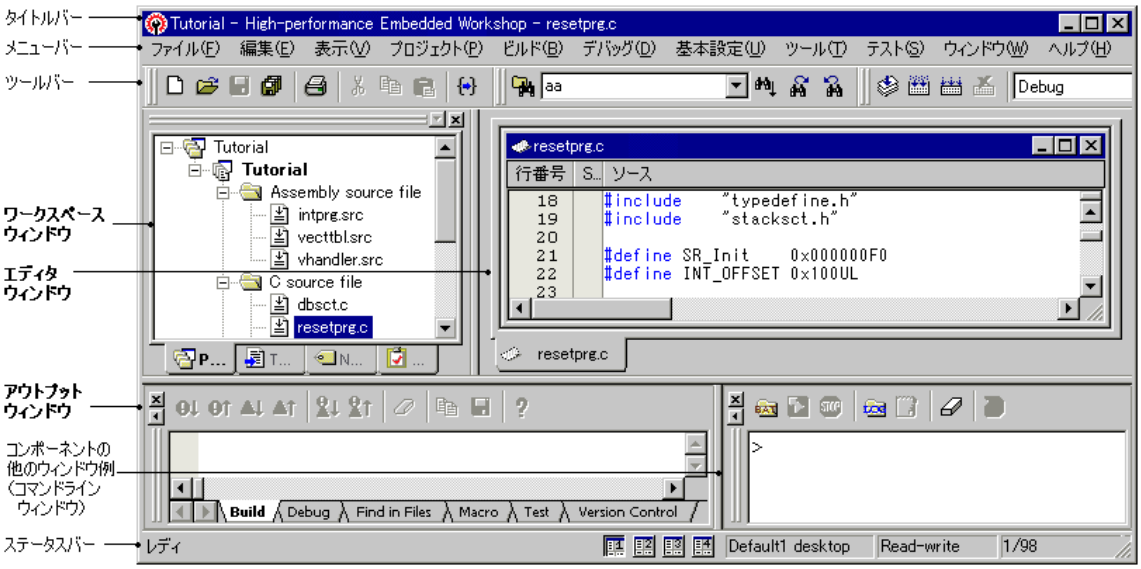


ワークスペースでは関連したプロジェクトを 1 つにまとめることができます。例えば、異なるプロセッサに対して 1 つのアプリケーションを構築しなければならない場合、または、アプリケーションとライブラリを同時に開発している場合などに便利です。さらに、ワークスペース内でプロジェクトを階層的に関連付けることができます。つまり、1 つのプロジェクトを構築すると、その子プロジェクトを最初に構築します。

ワークスペースを活用するには、まずワークスペースにプロジェクトを追加して、そのプロジェクトにファイルを追加しなければなりません。

1.2 メインウィンドウ

HEW にはメインウィンドウが 3 つあります。ワークスペースウィンドウ、エディタウィンドウ、アウトプットウィンドウです。ワークスペースウィンドウには現在そのワークスペースにあるプロジェクトやファイルを表示します。エディタウィンドウではファイルを表示、編集できます。アウトプットウィンドウにはさまざまな処理結果（ビルド、バージョン管理コマンドなど）を表示します。



1.2.1 タイトルバー

タイトルバーには、現在開いている、プロジェクト名、ファイル名を表示します。また、“最小化”ボタン、“最大化”ボタン、“閉じる”ボタンがあります。“最小化”ボタンをクリックすると Windows®のタスクバー上に HEW を最小化します。“最大化”ボタンをクリックすると HEW をフルスクリーンで表示します。“閉じる”ボタンをクリックすると HEW を閉じることができます。（これは[ファイル->アプリケーションの終了]を選択するか”Alt+F4”キーを押すのと同じです）。

1.2.2 メニューバー

メニューバーには次の 11 のメニューがあります。[ファイル]、[編集]、[表示]、[プロジェクト]、[ビルド]*、[デバッグ]、[基本設定]、[ツール]、[テスト]、[ウィンドウ]、および[ヘルプ]です。メニューのオプションは、すべてこれらメニューの下にグループ化されています。例えば、ファイルを開きたいときには[ファイル]メニューの下のオプションを選択します。ツールをセットアップしたいときには、[ツール]メニューを選択します。

*. HEW V.4.01 以降のバージョンで作成したデバッグ専用プロジェクト（[Debugger only - xxxxxx]）を使用している場合は、[ビルド]メニューは表示されません。HEW V.4.01 より前のバージョンで作成したデバッグ専用プロジェクトを使用している場合には、[ビルド]メニューは表示されます。

1.2.3 ツールバー

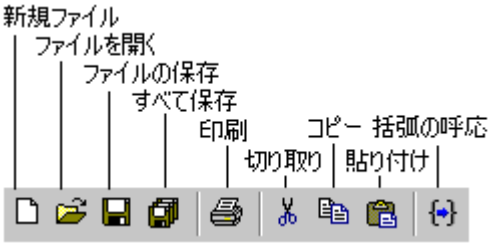
ツールバーにより使う頻度の高いオプションを簡単にアクセスできます。デフォルトでは[エディタ]、[検索]、[テンプレート]、[ブックマーク]、[デフォルト]、[標準]、[バージョン管理]、[マップ]、[ツール]、[マクロ]、[デバッグ]、[デバッグラン]の 12 のツールバーがあります（下の図を参照）。

デフォルトセッションの初期表示では、ツールバーには[バージョン管理]、周辺機能のボタンは表示しません。

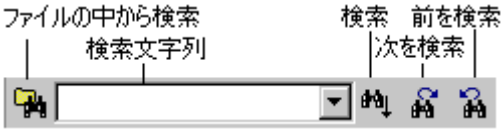
さらに、HEW V.4.01 以降のバージョンで作成したデバッグ専用プロジェクト（[Debugger only - xxxxxx]）を使用する場合、デフォルトでは[エディタ]、[検索]、[テンプレート]、[ブックマーク]、[デフォルト]、および[標準]は表示されません。

ツールバーの作成や変更は[基本設定->カスタマイズ]メニューオプションで指定できます（詳細は、「6.1 ツールバーのカスタマイズ」を参照してください）。

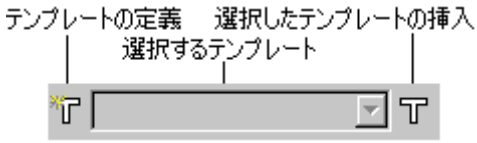
エディタツールバー



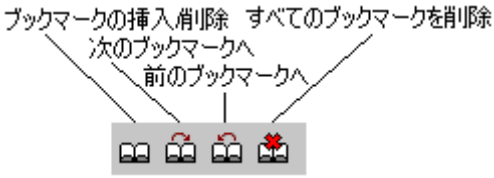
検索ツールバー



テンプレートツールバー

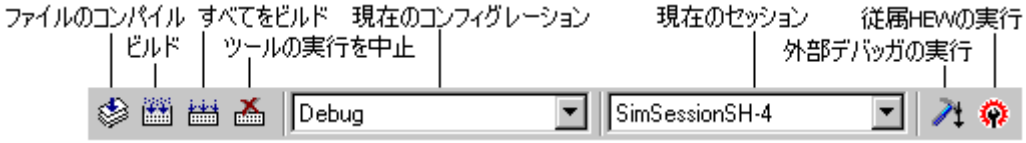


ブックマークツールバー



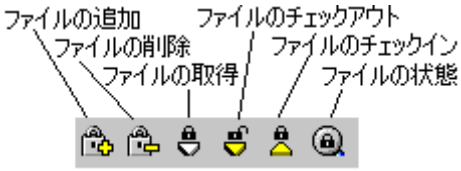
デフォルトツールバー

標準ツールバー

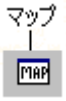


バージョン管理ツールバー

このツールバーは、現在のプロジェクトでバージョン管理ツールが使用されているときのみ使用できます。



マップツールバー



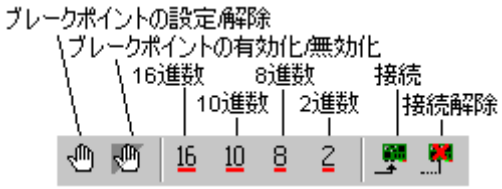
ツールツールバー

マクロツールバー

このツールバーには、下記の標準のマクロボタンの右側に作成したマクロを割り当てたカスタムのボタンを追加することが可能です。

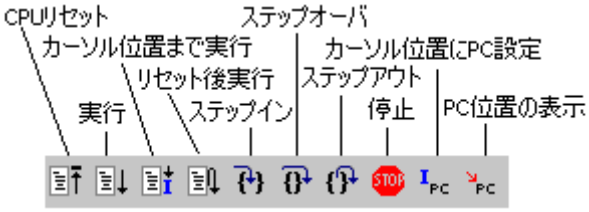
デバッグツールバー

このツールバーは、ターゲットが接続されたセッションを使用しているときのみ使用できます。

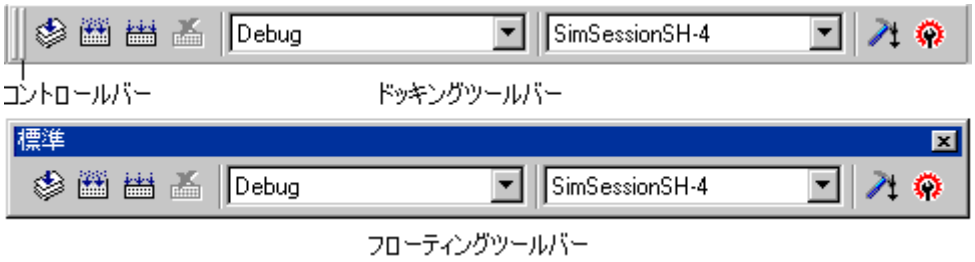


デバッグランツールバー

このツールバーは、ターゲットが接続されたセッションを使用しているときのみ使用できます。



[標準]ツールバーがドッキング（連結）状態のとき、下の図に示すコントロールバーを表示します。ドッキング状態の[標準]ツールバーの位置を移動したいときはコントロールバーを移動先までドラッグします。（ドラッグとは、マウスの左ボタンを押したまま目的の場所まで移動してからボタンを離すことをいいます。）下の図にドッキング状態、フローティング（浮遊）状態のツールバーを示します。



ツールバーをドッキング状態にするには

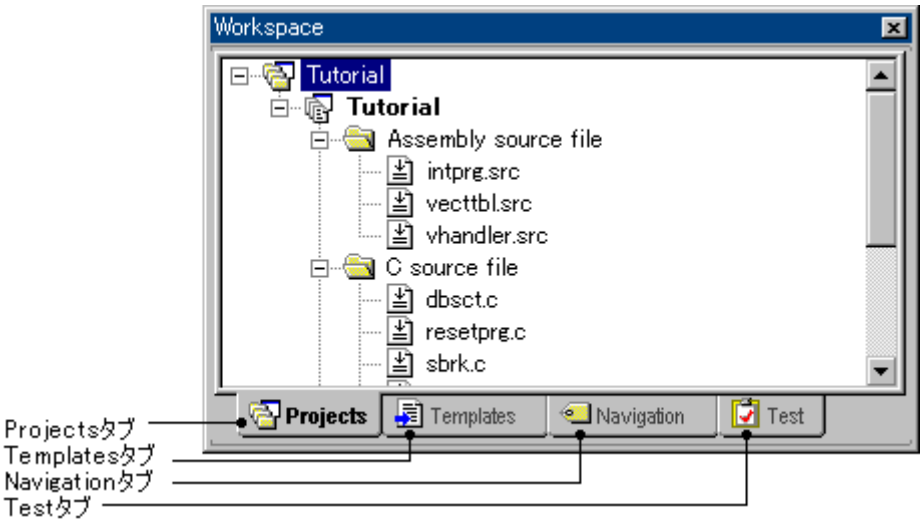
フローティング状態のツールバーのタイトルバーをダブルクリックしてください。または、フローティング状態のツールバーのタイトルバーをドッキング状態のウィンドウ、メニューバー、ツールバー、または HEW メインウィンドウの端までドラッグしてください。バーの形が変わります。

ツールバーをフローティング状態にするには

ドッキング状態のツールバーのコントロールバーをダブルクリックしてください。または、ドッキング状態のツールバーのコントロールバーを HEW のメインウィンドウおよびその他のドッキング状態のウィンドウ、メニューバー、またはツールバーの端から外れるようにドラッグしてください。

1.2.4 ワークスペースウィンドウ

ワークスペースウィンドウには 4 つのタブ（[Projects]タブ、[Templates]タブ、[Navigation]タブ、[Test]タブ）があります。



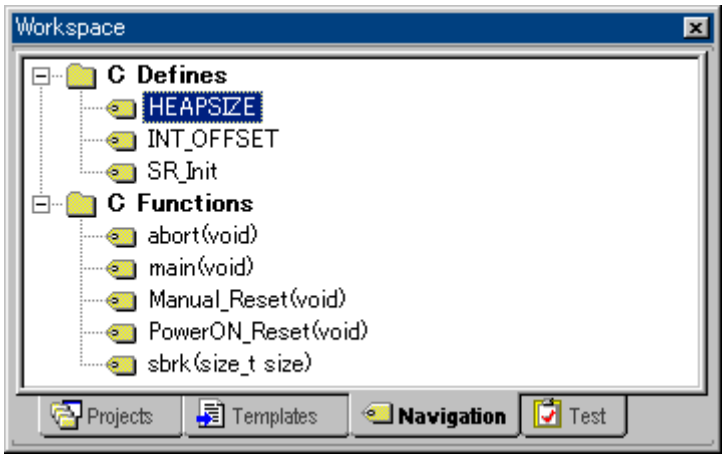
- **[Projects]タブ**

現在のワークスペース、プロジェクト、ファイルを表示します。アイコンのダブルクリックなどによりプロジェクトファイルや個々のファイルが開けます。[Projects]タブの詳細については「2.2 ワークスペースウィンドウの Projects タブの構成」を参照してください。

- **[Templates]タブ**

テンプレートの設定を表示します。テンプレートの詳細については「4.11 テンプレート」を参照してください。

- **[Navigation]タブ**



プロジェクトファイルの中のテキスト部分へジャンプできます。[Navigation]タブに実際に表示される内容は、現在、何がインストールされているかによって異なります。下の図に例えば ANSI 規格 の C マクロ名定義（C Defines）、および C 関数一覧（C Functions）を示します。ナビゲーションの詳細については「12. ナビゲーション機能」を参照してください。

- **[Test]タブ**

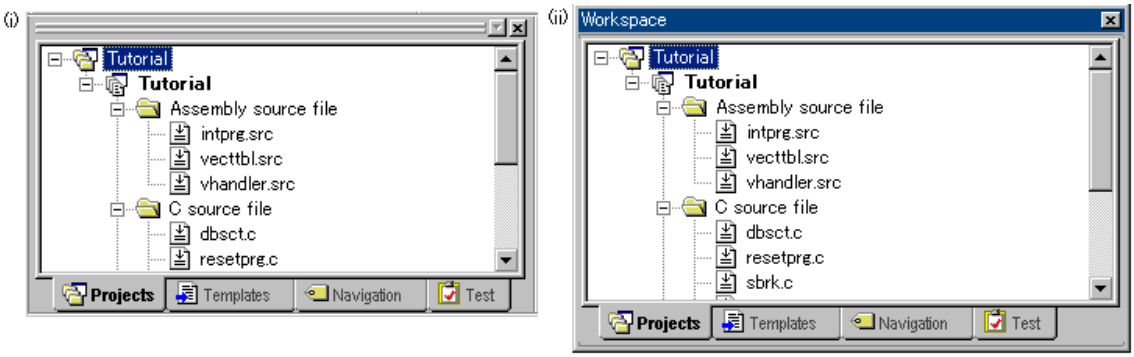
テスト支援機能としてテストスイートを設定および表示します。テスト支援機能の詳細については、「16. テスト支援機能の使用」を参照してください。

ワークスペースウィンドウやアウトプットウィンドウのドッキングビューとは

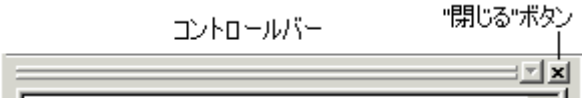
ウィンドウ上で右クリックしてください。ポップアップメニューを表示します。

[ドッキングビュー]にチェックマークが付いている場合はドッキングできます。チェックマークが外れている場合はドッキングできません。[ドッキングビュー]を選択するとチェックマークが付いたり外れたりします。

[ドッキングビュー]にチェックマークが付いている場合、ウィンドウを HEW メインウィンドウや他のドッキング状態のウィンドウの端にドッキングできます。同じく[ドッキングビュー]にチェックマークが付いている場合、ウィンドウを他の HEW のウィンドウ上や HEW メインウィンドウの外でフローティング状態にできます。下の図の(i)はドッキング状態のワークスペースウィンドウ、(ii)はフローティング状態のワークスペースウィンドウを示します。



ワークスペースウィンドウやアウトプットウィンドウがドッキング状態のとき、下の図に示すコントロールバーを表示します。ドッキング状態のウィンドウを移動したいとき、コントロールバーを移動先までドラッグしてください。



ワークスペースウィンドウやアウトプットウィンドウをドッキング状態にするには

1. ポップアップメニューで[ドッキングビュー]にチェックマークが付いていることを確認してください。
2. 次のいずれかを行ってください。

- フローティング状態のウィンドウのタイトルバーをダブルクリックしてください。

- フローティング状態のウィンドウのタイトルバーを、移動先のドッキング状態のウィンドウ、メニューバー、ツールバー、または HEW のメインウィンドウの端までドラッグしてください。

ワークスペースウィンドウやアウトプットウィンドウをフローティング状態にするには

1. ポップアップメニューで[ドッキングビュー]にチェックマークが付いていることを確認してください。
2. 次のいずれかを行ってください。

- ドッキング状態のウィンドウのコントロールバーをダブルクリックしてください。
- ドッキング状態のウィンドウのコントロールバーを、HEW のメインウィンドウや他のドッキング状態のウィンドウ、メニューバー、ツールバーの端から外れるようにドラッグしてください。
- ドッキング状態のウィンドウのコントロールバーを"Ctrl"キーを押しながらドラッグしてください。

ワークスペースウィンドウやアウトプットウィンドウを隠すには

次のいずれかを行ってください。

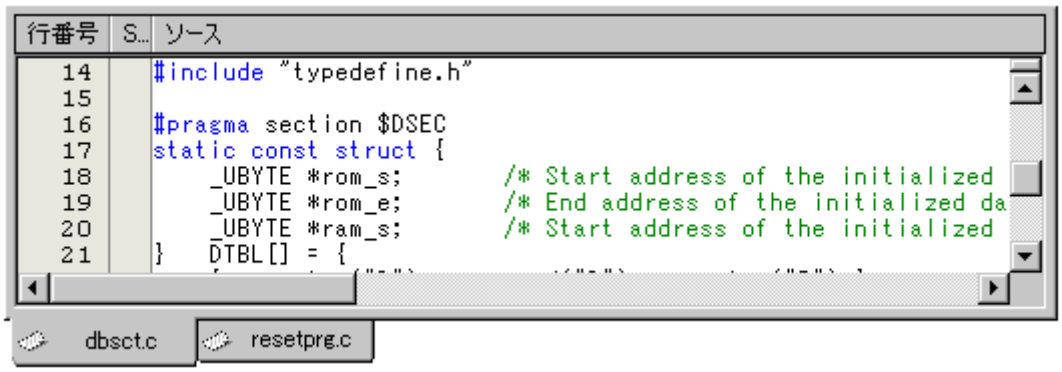
- ウィンドウの右上端にある"閉じる"ボタンをクリックしてください。
- フローティング状態のウィンドウの中で右クリックし、ポップアップメニューから[非表示]を選択してください。

ワークスペースウィンドウやアウトプットウィンドウを表示するには

ワークスペースウィンドウを表示するには[表示–>ワークスペース]を、アウトプットウィンドウを表示するには[表示–>アウトプット]を選択してください。

1.2.5 エディタウィンドウ

エディタウィンドウではプロジェクトのファイルを操作します。同時に複数のファイルを開いたり、任意の順序にファイルを切り替えたり、並べ替えたり、編集できます。デフォルトでは、エディタウィンドウはブック形式で表示します。つまり、各ファイルにタブが付いていてファイル間の行き来を容易にできます（下の図参照）。



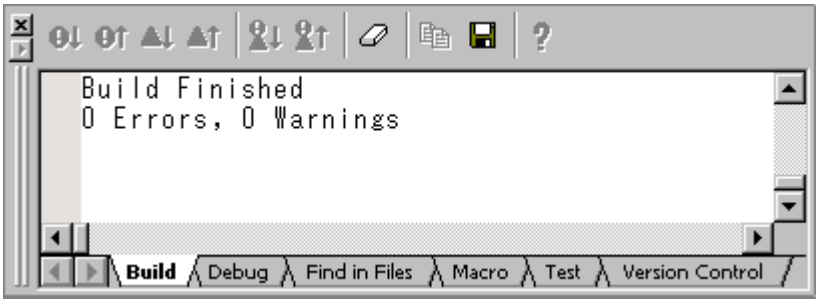
エディタには、ウィンドウ内の左側に余白（カラム）があります。これにより、ブックマークとブレークポイント(*)の位置を簡単に設定できます。カラムの用途やその表示する情報について知りたい場合は、そのカラム上にマウスポインタを置いてください。これを説明するツールチップとしてポップアップウィンドウを表示します。

エディタウィンドウは[表示形式]ダイアログボックスでカスタマイズできます。[表示形式]ダイアログボックスは[基本設定->表示の形式]メニューで開きます。[表示形式]ダイアログボックスでは、エディタウィンドウのフォントやテキストの色、タブ文字などが変更できます。また、HEW でインストールした他のウィンドウの外観も変更できます。HEW エディタ以外のエディタを使う場合は、使用するエディタを[オプション]ダイアログボックスで指定してください。[オプション]ダイアログボックスは、[基本設定->オプション]メニューオプションから開くことができます。エディタの使用方法については、「4. エディタの使用」を参照してください。

*. デバッギングプラットフォームがセッションに接続されている場合のみブレークポイントを設定できます。

1.2.6 アウトプットウィンドウ

デフォルトではアウトプットウィンドウに 6 つのタブ（[Build]タブ、[Debug]タブ、[Find in Files]タブ、[Macro]タブ、[Test]タブ、[Version Control]タブ）があります。



[Build]タブ

ビルド実行（コンパイラ、アセンブラなど）の出力を表示します。ソースファイルにエラーがある場合、アウトプットウィンドウの[Build]タブにアイコン、およびエラーメッセージをソースファイル名、行番号とともに出力します。

エラーメッセージの表示に関するツールバーボタンまたはポップアップメニュー項目をクリックすると、エラーメッセージ出力行を強調表示し、該当ソースがある場合はエディタに該当ソース行を表示します。

エラーメッセージ出力行をダブルクリックすると、該当ソースがある場合はエディタにエラーメッセージの該当ソース行を表示します。

エラーメッセージの表示に関する操作を行った場合は、ステータスバーにエラーメッセージを表示します。

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	機能
-		次のエラーの出力行を強調表示し、該当ソースがある場合はエディタに該当ソース行を表示します。
-		前のエラーの出力行を強調表示し、該当ソースがある場合はエディタに該当ソース行を表示します。
-		次のウォーニングの出力行を強調表示し、該当ソースがある場合はエディタに該当ソース行を表示します。
-		前のウォーニングの出力行を強調表示し、該当ソースがある場合はエディタに該当ソース行を表示します。
次エラー/ウォーニング/情報の表示		次のエラー、ウォーニング、またはインフォメーションの出力行を強調表示し、該当ソース行がある場合はエディタに該当ソース行を表示します。
前エラー/ウォーニング/情報の表示		前のエラー、ウォーニング、またはインフォメーションの出力行を強調表示し、該当ソースがある場合はエディタに該当ソース行を表示します。
ヘルプ		この出力行に関するヘルプを表示します。
ジャンプ	-	この出力行に関するソース行をエディタに表示します。
クリア		ウィンドウ内容をクリアします。
保存		ウィンドウ内容をテキストファイルに保存します。
コピー		ウィンドウ内容の選択部分を Windows(R)クリップボードにコピーします。
ツールバー表示	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ	-	ツールバーボタンをカスタマイズします。

[Build]タブではビルド実行により出力されたエラーメッセージ行の左側の余白（カラム）にエラーメッセージのエラーレベルに対応したアイコンを表示します。

アイコン名	アイコン	エラーメッセージのエラーレベル
Build Error		エラーであることを示します。
Build Warning		ウォーニングであることを示します。
Information		インフォメーションであることを示します。

[Debug]タブ

あらゆるデバッガ処理の出力を表示します。情報を表示する必要のあるデバッグツールから、アウトプットウィンドウの[Debug]タブに出力します。

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	機能
クリア		ウィンドウ内容をクリアします。
保存		ウィンドウ内容をテキストファイルに保存します。
コピー		ウィンドウ内容の選択部分を Windows(R)クリップボードにコピーします。
ツールバー表示	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ	-	ツールバーボタンをカスタマイズします。

[Find in Files]タブ

ファイル内から検索操作の結果を表示します。ファイル内から検索を使用するには、[編集->ファイルから検索]メニューオプションを選択するか、ツールバーの[ファイルの中から検索]ボタンをクリックしてください。ファイル内から検索の使い方の詳細については、「4.4.2 複数のファイル間でのテキスト検索」を参照してください。

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	機能
ジャンプ	-	関連するソース行を表示します。
クリア		ウィンドウ内容をクリアします。
保存		ウィンドウ内容をテキストファイルに保存します。
コピー		ウィンドウ内容の選択部分を Windows(R)クリップボードにコピーします。
ツールバー表示	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ	-	ツールバーボタンをカスタマイズします。

[Macro]タブ

マクロ生成支援機能として、現在のマクロの記録状況を表示します。[ツール->マクロの記録]による記録の開始から[ツール->マクロの停止]による記録の終了までに HEW マクロファイルに記録した HEW コマンドラインのコマンドなどの情報を表示します。マクロを記録しながら記録内容を確認できます。マクロ生成支援機能の詳細については、「15. マクロ生成支援機能の使用」を参照してください。

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	機能
クリア		ウィンドウ内容をクリアします。
保存		ウィンドウ内容をテキストファイルに保存します。
コピー		ウィンドウ内容の選択部分を Windows(R)クリップボードにコピーします。
ツールバー表示	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ	-	ツールバーボタンをカスタマイズします。

[Test]タブ

テスト支援機能として、現在のテスト実行の結果および進行状況を表示します。テスト実行の進行状況は、現在実行中のテストと、残りのテストの数で構成されています。エラーが発生した場合、このウィンドウに表示されます。テスト支援機能の詳細については、「16. テスト支援機能の使用」を参照してください。

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	機能
クリア		ウィンドウ内容をクリアします。
保存		ウィンドウ内容をテキストファイルに保存します。
コピー		ウィンドウ内容の選択部分を Windows(R)クリップボードにコピーします。
ツールバー表示	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ	-	ツールバーボタンをカスタマイズします。

[Version Control]タブ

バージョン管理操作の結果を表示します。このタブは、バージョン管理システムを使っているときだけ表示します。バージョン管理の詳細については、「7. バージョン管理」を参照してください。

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	機能
クリア		ウィンドウ内容をクリアします。
保存		ウィンドウ内容をテキストファイルに保存します。
コピー		ウィンドウ内容の選択部分を Windows(R)クリップボードにコピーします。
ツールバー表示	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ	-	ツールバーボタンをカスタマイズします。

”Shift+ESC”キーを押すとアウトプットウィンドウを閉じます。

アウトプットウィンドウはテキストの色、背景色、およびフォントを他のウィンドウと同じようにカスタマイズできます。

さらにアウトプットウィンドウの[Build]タブでは、エラーメッセージ行のテキストを他の行のテキストと異なる色で強調表示されるようにカスタマイズが可能です。

既存の色をカスタマイズするには

1. [基本設定->表示の形式]を選択し、[表示形式]ダイアログボックスを表示します。
2. [Output]アイテムをダイアログボックスの左側のツリーから選択し展開します。
3. 色を変更したいカテゴリを選択します。次の表に示すカテゴリが選択できます。

カテゴリ	カラータブの前面の色（デフォルト）	カラータブの背景の色（デフォルト）	カスタマイズされるタブ	カスタマイズされる出力内容
Text	SYSTEM	SYSTEM	すべてのタブ	すべて
Build Error Text	黒	白	Build	エラー行
Build Warning Text	黒	白	Build	ウォーニング行
Information Text	黒	白	Build	インフォメーション行

4. [カラー]タブの[前面]リストと[背景]リストの選択を変更します。
5. [OK]ボタンをクリックします。

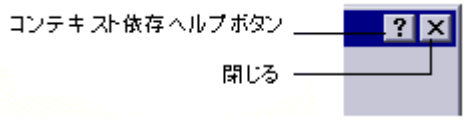
1.2.7 ステータスバー

現在の HEW の状態を表示します。下の図にステータスバーを示します。

1.3 ヘルプ機能

[ヘルプ]メニューは、High-performance Embedded Workshop メニューバーの右端にあります。[ヘルプ]メニューには[トピック]メニューオプションがあります。[トピック]メニューオプションを選択すると、High-performance Embedded Workshop ヘルプウィンドウのメイン画面を表示します。

特定のダイアログボックスに関するヘルプを参照したいときは、各ダイアログボックスの右上端にあるコンテキスト依存ヘルプボタンをクリックしてください（下の図を参照）。

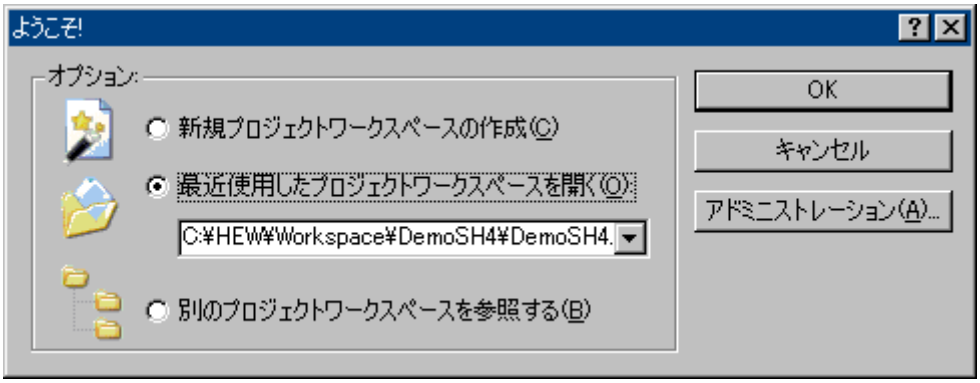


コンテキスト依存ヘルプボタンをクリックすると、マウスポインタが？(クエスチョンマーク)付きのポインタに変わります。この状態で、ダイアログボックスの一部をクリックすると、そ

の部分に関するヘルプを表示します。また、ダイアログボックスの一部を選択し F1 キーを押すと、そのヘルプを表示します。

1.4 HEW を起動する

HEW を起動するには Windows®の[スタート]メニューを開き、[すべてのプログラム->RENESAS->High-performance Embedded Workshop]を選択し、HEW のショートカットを選択します。起動画面に続き[ようこそ!]ダイアログボックスが表示されます。



HEW を起動しワークスペースを使用したことがある場合は、デフォルトで[最近使用したプロジェクトワークスペースを開く]ボタンが選択されています。一度もワークスペースを使用していない場合は、[新規プロジェクトワークスペースの作成]ボタンが選択されています。

新規ワークスペースを作成するには[新規プロジェクトワークスペースの作成]ボタンを選択し、[OK]ボタンをクリックしてください。

最近使用したワークスペースを開くには[最近使用したプロジェクトワークスペースを開く]ボタンを選択し、ドロップダウンリストから開きたいワークスペースを選択し、[OK]ボタンをクリックしてください。このリストには、[ファイル->最近使ったワークスペース]のサブメニューに表示されているものと同じ内容を表示します。

ワークスペースファイル（".HWS"ファイル）を指定してワークスペースを開くには[別のプロジェクトワークスペースを参照する]ボタンを選択し、[OK]ボタンをクリックしてください。

HEW にツールを登録したり、HEW からツールの登録を取り消したりするには[アドミニストレーション]ボタンをクリックしてください。

ワークスペースを開かずに HEW を使うには[キャンセル]ボタンをクリックしてください。

次回 HEW を起動する時に[ようこそ!]ダイアログボックスを表示しないようにするには、[基本設定->オプション]を選択し、表示されるオプションダイアログボックスの確認タブで[ようこそダイアログボックスの表示]チェックボックスをオフにしてください。

HEW を起動する時に起動画面を表示しないようにするには、オプションダイアログボックスの確認タブで[起動画面]チェックボックスをオフにしてください。

1.5 新規ワークスペースを作成する

新規にワークスペースを作成するには

1. [ようこそ!]ダイアログボックスから[新規プロジェクトワークスペースの作成]オプションを選択し、[OK]ボタンをクリックするか、[ファイル->新規ワークスペース]を選択してください。[新規プロジェクトワークスペース]ダイアログボックスを表示します。
2. [ワークスペース名]フィールドに新規ワークスペース名を入力してください。新規ワークスペース名は 32 文字以内で、半角英数字、半角下線のみ使用してください。特に、日本語文字、空白、または半角マイナス記号は使用しないでください。ワークスペース名を入力すると、自動的にワークスペースのサブディレクトリおよび[プロジェクト名]が追加されます。これは、必要に応じて変更できます。ワークスペースとプロジェクトは別名にできます。[参照]ボタンをクリックしてワークスペースを作成するディレクトリを選択したり、[ディレクトリ]フィールドに、ワークスペースを作成するディレクトリを入力できます。
3. ワークスペースの基本となる[CPU 種別]および[ツールチェイン]を選択してください。
4. 新規ワークスペースを作成するとき、HEW は自動的に[プロジェクト名]フィールドで指定したプロジェクトを作成して、新規ワークスペースに追加します。プロジェクトタイプリストには、使用可能なプロジェクトの種類（Application、Library など）を表示します。作成するプロジェクトの種類をリストから選択してください。表示するプロジェクトの種類は、現在の[CPU 種別]および[ツールチェイン]の組み合わせに有効な全種類となります。ツールチェイン専用、デバッグ専用（[Debugger only - xxxxxx]）、または HEW のデバッガおよびツールチェインの両方を構築するフルプロジェクトがあります。
5. 新規ワークスペースとプロジェクトを作成するには、[OK]ボタンをクリックしてください。選択したプロジェクトの作成をガイドするウィザードが開きます。

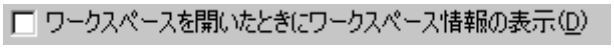
注：

同一ディレクトリにすでにワークスペースが存在する場合はワークスペースを作成できません。

1.6 ワークスペースを開く

ワークスペースを開くには

1. [ようこそ!]ダイアログボックスから[別のプロジェクトワークスペースを参照する]オプションを選んで[OK]ボタンをクリックするか、[ファイル->ワークスペースを開く]を選択してください。[ワークスペースを開く]ダイアログボックスを表示します。
2. 開きたいワークスペースファイル（”.HWS”ファイル）を選択します。
3. [選択]ボタンをクリックしてワークスペースを開いてください。ワークスペースを開くときに情報を表示するように設定している場合、[ワークスペースプロパティ]ダイアログボックスを表示します。設定していない場合、ワークスペースを開きます。[ワークスペースプロパティ]ダイアログボックスを表示するかどうかは[ワークスペースプロパティ]ダイアログボックスの[ワークスペースを開いたときにワークスペース情報の表示]チェックボックス、または、[オプション]ダイアログボックスの[ワークスペース]タブの[ワークスペースを開いたときにワークスペース情報の表示]チェックボックスのオンオフの有無によります。[オプション]ダイアログボックスは、[基本設定->オプション]メニューオプションから開くことができます。[ワークスペースプロパティ]ダイアログボックスで[OK]ボタンをクリックするとワークスペースを開きます。[キャンセル]ボタンをクリックするとワークスペースを開きません。



HEW では、最近開いた 4 ファイルを[ファイル]メニューの[最近使ったワークスペース]サブメニューに表示します。最近使ったワークスペースを再び開きたいときに使用してください。

最近使ったワークスペースを開くには

次のいずれかを行ってください。

- [ようこそ!]ダイアログボックスから[最近使用したプロジェクトワークスペースを開く]を選択し、ドロップダウンリストからワークスペース名を選択し、[OK]ボタンをクリックします。
- [ファイル->最近使ったワークスペース]を選択し、そのサブメニューからワークスペース名を選択します。

注：

HEW では、一度に 1 つのワークスペースしか開けません。したがって、すでに開いているワークスペースがあるときに別のワークスペースを開こうとすると、すでに開いているワークスペースを閉じてから新しいワークスペースを開く必要があります。

1.7 古いワークスペースを使用する

HEW では、以前のバージョンの HEW で作成したワークスペースも開くことができます。ワークスペースを開いたときに自動的にアップグレードします。アップグレードしたファイルの現在のディレクトリに、初期のワークスペースまたはプロジェクトファイルのバックアップを保存します。

1.8 ワークスペースを保存する

HEW のワークスペースを保存するには、[ファイル->ワークスペースの保存]を選択してください。

1.9 ワークスペースを閉じる

HEW のワークスペースを閉じるには、[ファイル->ワークスペースを閉じる]を選択してください。ワークスペースまたはそのプロジェクトに変更があった場合は、保存するかどうかを選択してください。

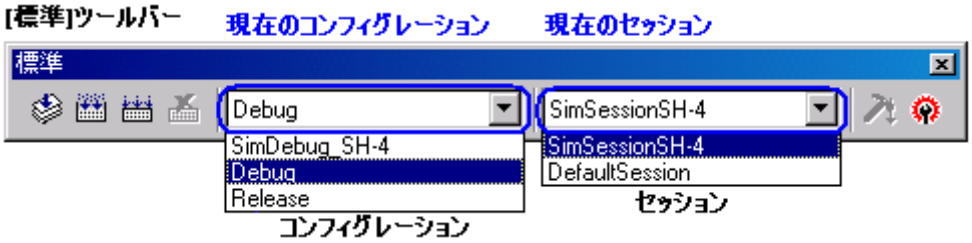
1.10 HEW を終了する

HEW を終了するには [ファイル->アプリケーションの終了]を選択するか、“Alt+F4”キーを押すか、システムメニューから[閉じる]オプションを選択してください。（システムメニューは HEW タイトルバーの最も左上側にあるアイコンをクリックすると開きます。）

1.11　ツールシステム概要

ユーザは、更にツールを追加することによって、HEW の機能を拡張できます。これを行うには、[ツールアドミニストレーション]ダイアログボックスでツールを登録しておくことが必要です。これらのツールを用いて、ウィンドウ、メニュー、およびツールバーを HEW システムに追加できます。ツールの例としては、HEW のデバッガおよびビルダツールがあります。デバッガツールはデバッガに関連するすべてのメニューとツールバーを追加し、ビルダツールも同じようにビルド機能に関するすべてのメニューとツールバーを追加します。システムに登録したツールによって、HEW の使い方が変わります。そのため、このユーザーズマニュアルに記載されたメニューのうち使用できないものもあります。例えば、ビルダツールがインストールされていない場合、[ビルド]メニューにはオプション設定のメニュー項目がありません。

1.12　コンフィグレーションおよびセッションの概要



[標準]ツールバーに現在のコンフィグレーションおよび現在のセッションをドロップダウンリストボックスで表示します。

この例は、プロジェクト作成時に Renesas SuperH Standard 用ツールチェインを選択し、ターゲットとするデバッギングプラットフォームに SH-4 Simulator を選択した場合のコンフィグレーションとセッションを示しています。

ドロップダウンリストボックスを選択し現在のコンフィグレーションおよび現在のセッションから別のコンフィグレーションおよび別のセッションへ切り替えることができます。

必要に応じてコンフィグレーションおよびセッションを追加したり削除したりすることができます。

- コンフィグレーション

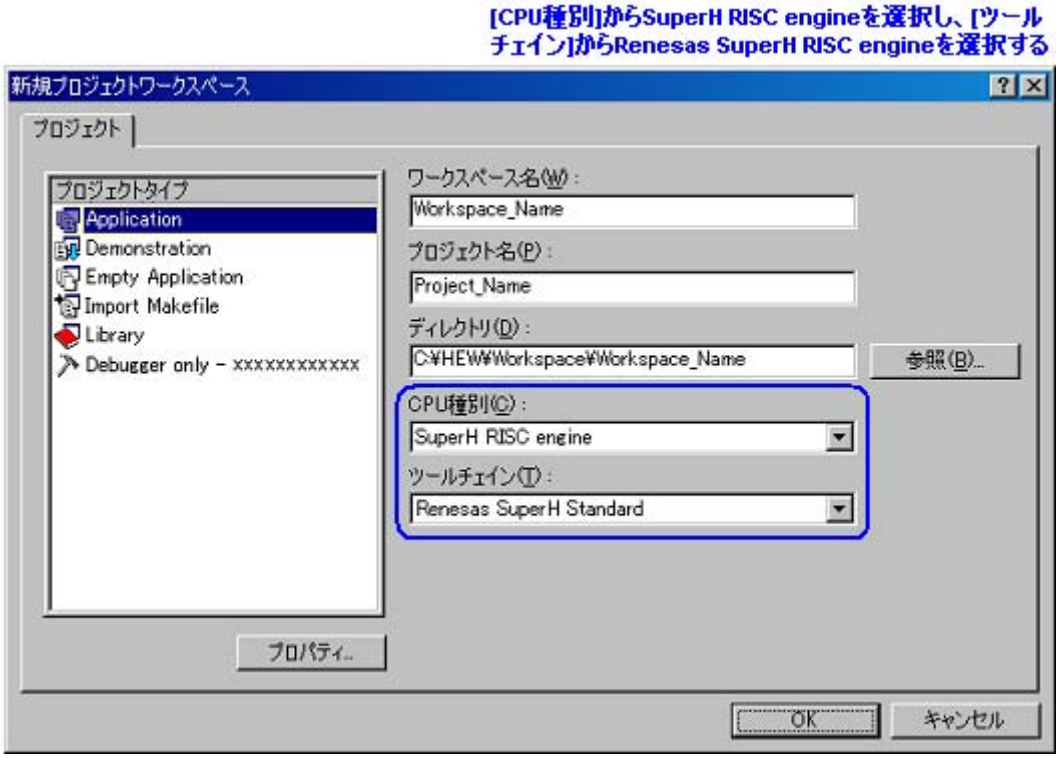
コンフィグレーションとは

デバッグ情報の出力の有無、および最適化の有無といったビルドのオプションの設定内容に名前を付けて保持しているものです。”ビルドコンフィグレーション”ともいいます。

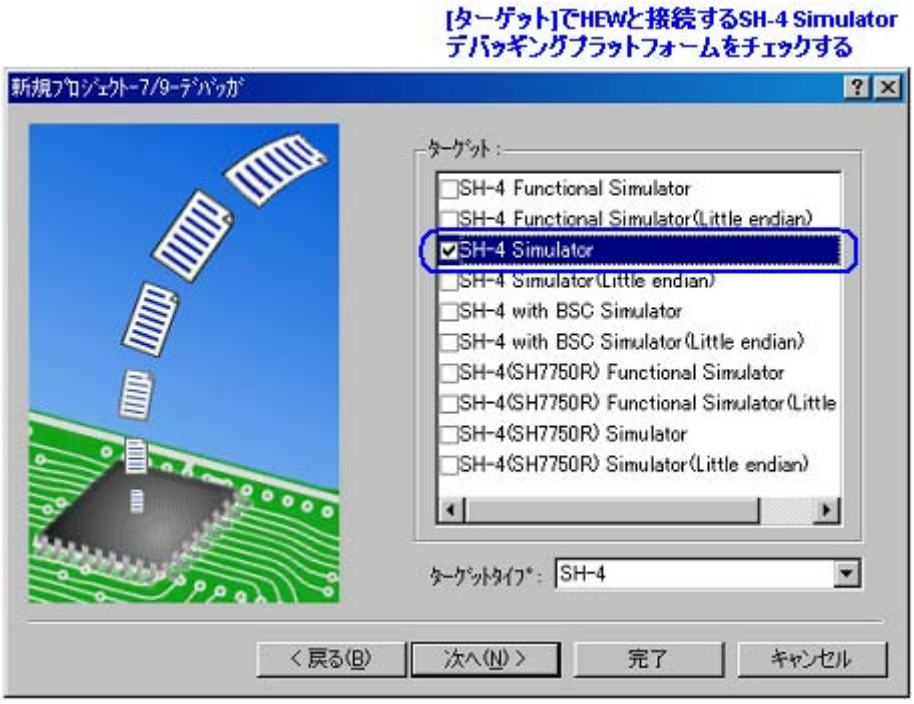
[標準]ツールバーの図の例では、SimDebug_SH-4 コンフィグレーション、Debug コンフィグレーション、および Release コンフィグレーションを表示します。

HEW が作成するコンフィグレーション

ツールチェインを選択しプロジェクトを作成した場合、Debug コンフィグレーションおよび Release コンフィグレーションを自動的に作成します。

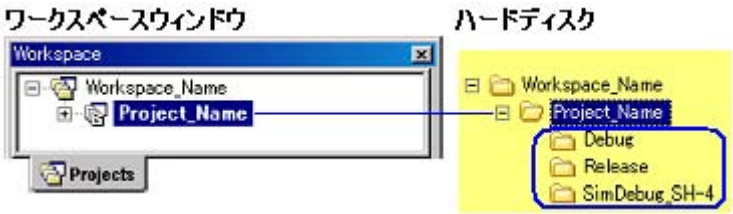


プロジェクトの作成時にターゲットとするデバッギングプラットフォームをチェックした場合は、チェックしたデバッギングプラットフォーム用のコンフィグレーションが作成されます。プロジェクトの作成時に[ターゲット]で”SH-4 Simulator”をチェックした場合は SimDebug_SH-4 コンフィグレーションが作成されます。コンフィグレーションの名前はプロジェクトの作成時に変更できます。



また、プロジェクトディレクトリ（*）下に各コンフィグレーション用のディレクトリがコンフィグレーションと同じ名前で作成されます。

*. プロジェクトディレクトリは、新規ワークスペースプロジェクト作成時のワークスペースディレクトリ下にプロジェクト名と同じ名前で作成されます。



ビルドのオプションの設定方法

コンフィグレーションに保持されるビルドのオプションを設定するには、[ビルド]メニューのビルドのオプションを設定するダイアログボックスでドロップダウンリストボックスからコンフィグレーションを選択し、コンパイラ、アセンブラ、および最適化リンカなどの各オプションを設定します。下の図は、Renesas SuperH Standard 用ツールチェインのビルドのオプションを設定するダイアログボックスです。

コンフィグレーションの詳細は、「2.7 ビルドのコンフィグレーション」を参照してください。

- セッション

セッションとは

おもにデバッギングプラットフォームに接続してデバッグするときに使用する **HEW のデバッグオプションの設定**（*****）、デバッギングプラットフォーム固有の様々なオプションの設定、およびメモリウィンドウ、またはレジスタウィンドウなどのウィンドウとその位置の情報などに名前を付け保持しているものです。”デバッグセッション”ともいいます。

[標準]ツールバーの図の例では、SimSessionSH-4 セッションおよび DefaultSession セッションを表示しています。各セッションの情報は、HEW プロジェクトでそれぞれ別のファイルに保存されます。

***.** HEW と接続するデバッギングプラットフォーム、デバッグ対象のオブジェクトフォーマット、ダウンロードモジュールの設定、デバッギングプラットフォームへ HEW が接続するタイミング、およびコマンドバッチを HEW が実行するタイミングなどのオプション設定があります。

HEW が作成するセッション

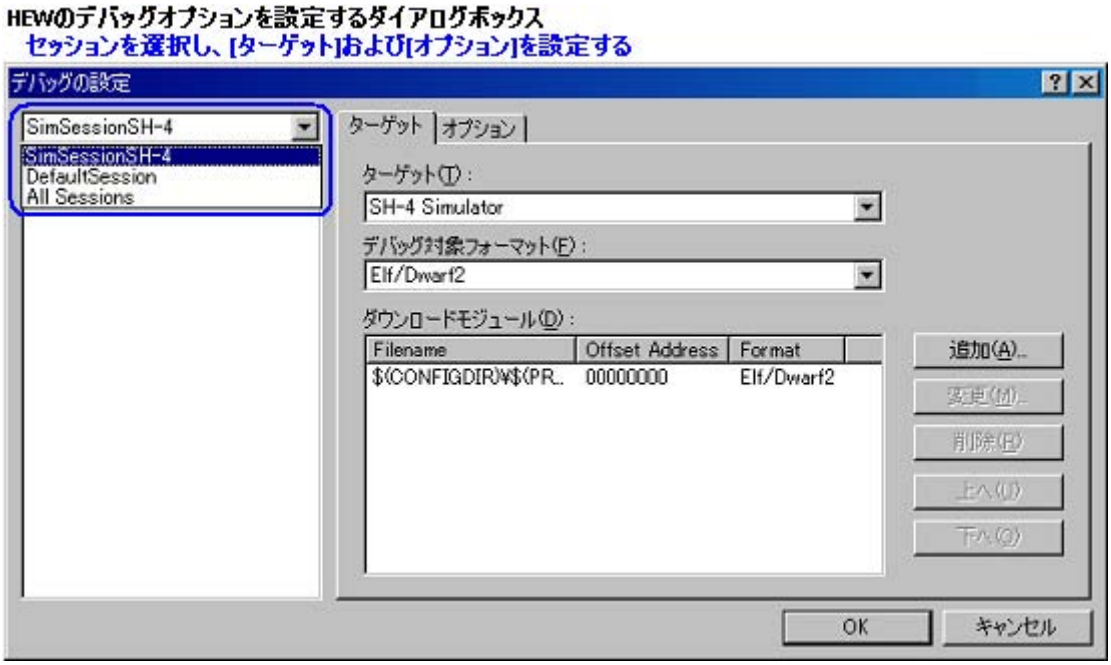
ツールチェインを選択しプロジェクトを作成した場合、DefaultSession セッションを自動的に作成します。

プロジェクトの作成時にターゲットとするデバッギングプラットフォームをチェックした場合は、チェックしたデバッギングプラットフォームを HEW と接続するセッションを自動的に作成します。

例えば、プロジェクトの作成時に[ターゲット]で”SH-4 Simulator”をチェックした場合は SimSessionSH-4 セッションが作成されます。

HEW のデバッグオプションの設定方法

セッションに保持される HEW のデバッグオプションを設定するには、[デバッグ]メニューの[デバッグの設定]ダイアログボックスでドロップダウンリストボックスからセッションを選択し、[ターゲット]タブおよび[オプション]タブで各オプションを設定します。



デバッグセッションの詳細は、「17.1.6 デバッグセッション」を参照してください。

1.13　マクロ生成支援機能とテスト支援機能の概要

テスト完了後にプログラムを修正した場合は再テストが必要になります。プログラム開発において、再テストに掛かる工数は非常に大きな割合を占めています。

テスト支援機能を使用し、特定のウィンドウのデータをテストイメージデータとしてファイルへ保存できます。これはテスト実行後のデータ比較に使用します。テスト支援機能は、テスト実行から実行後のデータ比較までの一連の操作をテストケースとして管理できます。テストの再実行および実行後のデータ比較によりテストの評価を繰り返し行なうことができます。

テストケースを作成する前にテストで実行するテストスクリプトのファイルを準備する必要があります。このテストスクリプトは HEW コマンドラインのコマンド形式です。

マクロ生成支援機能を使用し、デバッグ時に行ったユーザ操作など一部の操作を HEW コマンドラインのコマンドでファイルへ保存できます。このファイルをテストで実行するテストスクリプトのファイルとして使用することができます。

- マクロ生成支援機能

マクロ生成支援機能は、HEW システムのアプリケーション関連（***1**）、ビルド関連（***2**）、およびデバッグ関連（***3**）などの一部の操作を HEW コマンドラインのコマンドとして記録、または記録したコマンドを実行する機能です。

記録するファイル（HEW マクロファイル）は拡張子が HDC のコマンドバッチファイルで、編集可能です。このファイルは、HEW インストールディレクトリ下の「Macro」ディレクトリに設置されます。デフォルトで Default.hdc ファイルが作成されます。

HEW コマンドラインのコマンドに対応する HEW システムのすべての操作が記録できるわけではありません。各機能のメニュー項目一覧には”マクロの記録”欄にマクロの記録マーク（●）を表示し HEW マクロファイルへ記録可能かどうかを示しています。また、「15.5 マクロファ

イルにマクロを記録できる機能」に記録する HEW コマンドラインのコマンドを記載していますので参照してください。

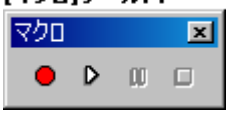
***1.** プロジェクトの変更、セッションの変更、コンフィグレーションの変更など

***2.** コンパイル、ビルドなど。機能のサポートは、デバッギングプラットフォームに依存します。

***3.** モジュールのダウンロード、メモリ値の変更、レジスタ値の変更、ソフトウェアブレークの設定／解除、プログラム実行など

マクロ生成支援機能は、[ツール]メニューと[マクロ]ツールバーで利用できます。

[マクロ]ツールバー

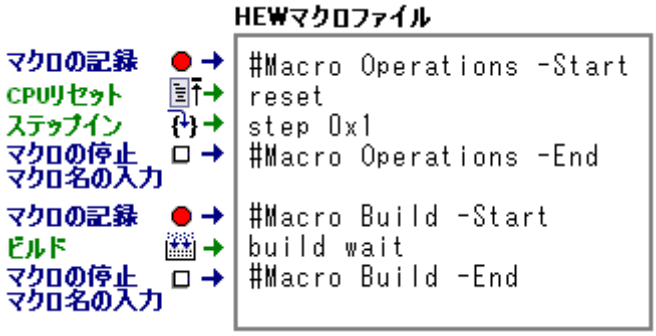


[マクロ]ツールバーが表示されていない場合は、[基本設定->カスタマイズ]を選択し[カスタマイズ]ダイアログボックスを表示してください。[カスタマイズ]ダイアログボックスの[ツールバー]タブで、[マクロ]チェックボックスをオンにしてください。

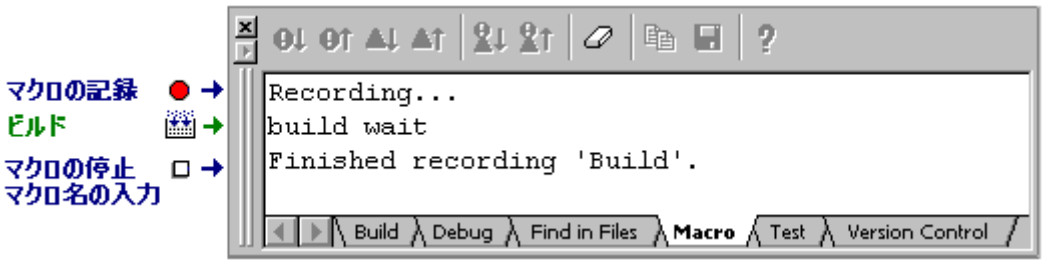
[ツール]メニュー	[マクロ]ツールバー	機能
マクロの設定	－	[マクロの設定]ダイアログボックスが表示されます。下記の機能を備えており、選択した HEW のマクロファイル毎に記録されたマクロがリストに表示されています。 ・マクロファイルの新規作成 ・マクロファイルのインポート ・マクロの記録 ・マクロの実行 ・マクロの編集 ・マクロの割り当て ・マクロの削除
マクロの記録	●	マクロの記録を開始します。
マクロの実行	▷	[マクロの選択]ダイアログボックスが表示されます。実行するマクロを選択します。
－	▯▯	マクロの記録または実行を一時停止します。
マクロの停止	□	マクロの記録または実行を停止します。

[マクロの記録]（●）を選択してから[マクロの停止]（□）を選択するまでが 1 つの”マクロ”で、1 つの HEW マクロファイルには複数のマクロ（HEW コマンドラインのコマンドの集まり）が記録できます。マクロには、複数の HEW コマンドラインのコマンドが含まれます。

下記は HEW マクロファイルとマクロの例です。



アウトプットウィンドウの[Macro]タブには現在のマクロの記録状況を出力します。上の図の Build マクロの記録状況は下記のように出力します。



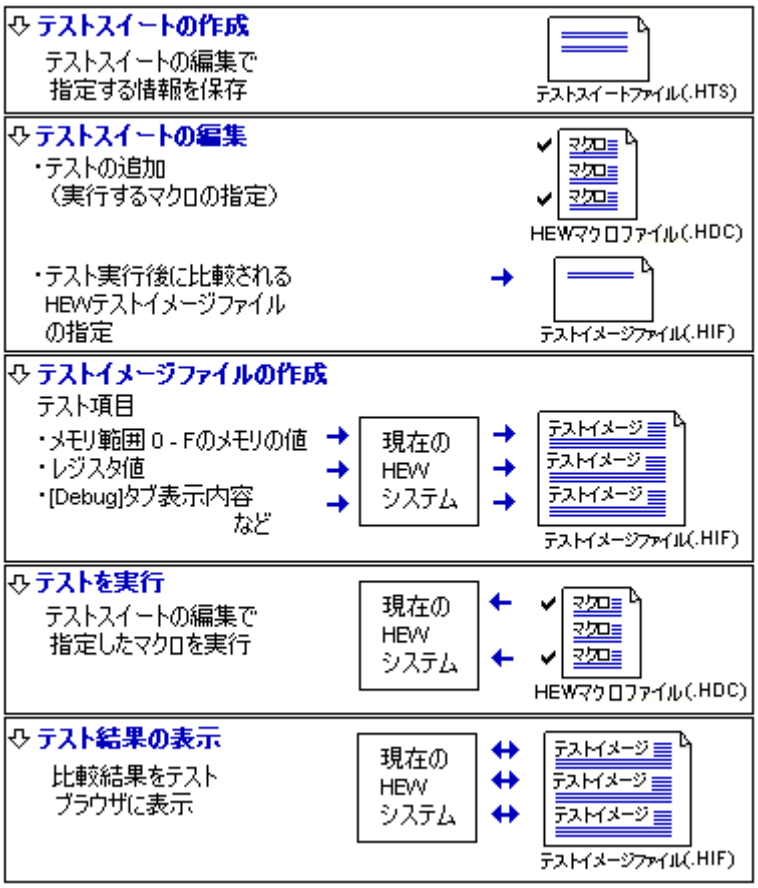
操作手順につきましては「1.13.1 テストシナリオの例」も参照してください。

- テスト支援機能

テスト支援機能は、比較したいテスト項目（テストイメージデータ）でテストイメージファイルを作成し、マクロ生成支援機能で生成したマクロまたは従来の HEW のコマンドバッチを実行して、テストイメージファイルと HEW システムを比較できる機能です。テストイメージデータはテストイメージファイル（.HIF）に保存します。

HEW システムのすべての機能のテストイメージデータを保存できるわけではありません。テストイメージデータを保存できる機能については、「16.6 テストイメージファイルにテストイメージデータとして保存できる機能」に記載していますので参照してください。

テスト支援機能の主な操作手順を下記に示します。操作手順については「1.13.1 テストシナリオの例」も参照してください。



テスト支援機能は、[テスト]メニューとワークスペースウィンドウの[Test]タブのポップアップメニューで利用できます。

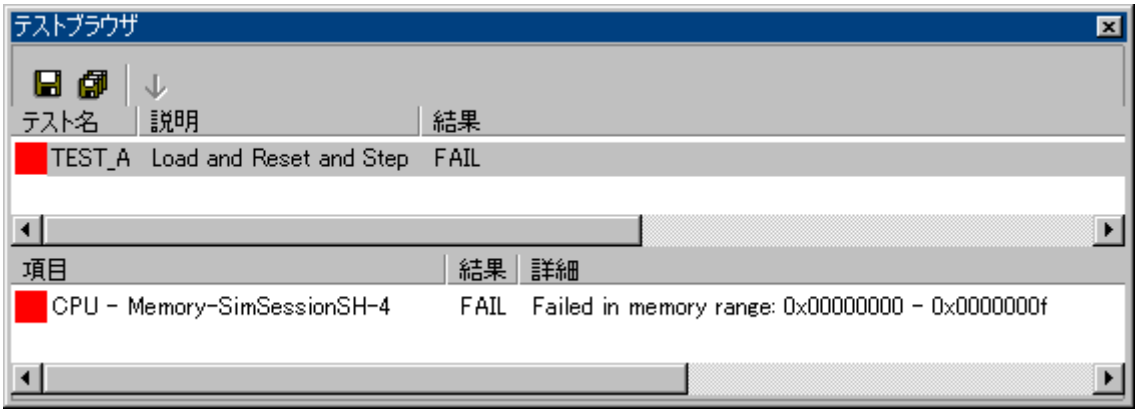
テスト支援機能の最初のステップは、テストスイート（テストの集まり）の作成です。[テスト->テストスイートの作成]メニューオプションで作成できます。テストスイートの作成後は、ワークスペースウィンドウの[Test]タブでも操作可能です。

テストスイートを編集し、テスト中に実行するマクロおよびテストベースのテストイメージファイルを指定します。

次にテスト内容として、テストイメージファイルに取得するためのいくつかのテスト項目を設定します。テストベースとしたい HEW の状態で、設定したテスト項目に基づきテストイメージデータを取得しテストイメージファイルに保存します。

テスト（マクロの集まり）を実行し、テストが完了すると指定したテスト項目が比較されます。結果がテストブラウザウィンドウに表示されます。

不一致（結果が”FAIL”）項目をダブルクリックすると詳細情報が確認可能です。



1.13.1 テストシナリオの例

マクロ生成支援機能とテスト支援機能を使用した主なビルド、デバッグ操作の流れをテストシナリオの一例として紹介します。

このテストシナリオでは、SuperH RISC engine 用シミュレータなどで Demonstration タイプを指定したプロジェクトやエミュレータデバッガなどで Tutorial プロジェクトに含まれているような C 言語で記述された 10 個のランダムデータを昇順/降順にソートするプログラムを使用します。配列に格納されるデータに差異が発生したケースを確認します。

ここでは、プログラムコード内の配列"a"をローカル変数からグローバル変数に編集してから使用します。

- テストの前に

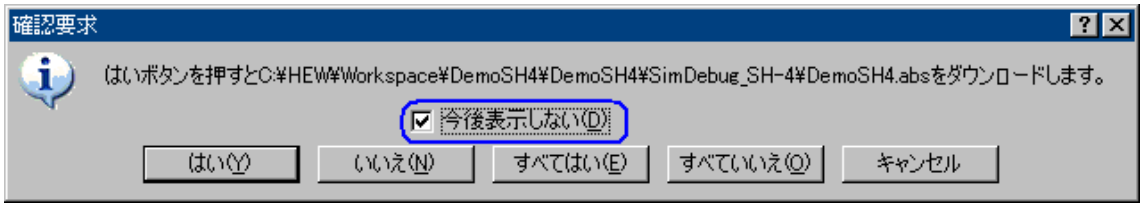
はじめにワークスペースを開きデバッギングプラットフォームに接続します。

1. 新規にプロジェクトワークスペースを作成するか既にある場合はこれを開きます。プロジェクトワークスペースの作成については「17.1.2 デバッギングプラットフォームを選択する」を参照してください。
2. [標準]ツールバーのセッションリストからデバッギングプラットフォームに接続するセッションを選択します。セッションの選択については「17.1.6 (1) セッションを選択する」を参照してください。

操作により確認要求ダイアログボックスが表示される場合があります。例えば、プログラムをすべてビルドした場合、デフォルトでダウンロードするか否かの確認要求ダイアログボックスが表示されます。プログラムをダウンロードする場合は「はい」ボタンをクリックします。

確認要求ダイアログボックスのボタンのクリックはマクロに記録されないため、次回マクロを実行すると、確認要求ダイアログボックスは表示されてしまいます。マクロ実行処理を継続するためにはいずれかのボタンをクリックしなくてはいけません。テスト支援機能を使用する場合、一連の操作を繰り返しテストするので処理を継続する必要があります。そのため、確認要求ダイアログボックスが表示されないよう次のように設定してください。

1. 確認要求ダイアログボックスの[今後表示しない]チェックボックスをオンにしてください。



テスト開始時の HEW の環境をテスト開始前に毎回同じ環境にするため、テストに使用したい HEW 環境を保存してください。

1. [ファイル->ワークスペースの保存]を選択してください。
2. [ファイル->セッションの保存]を選択してください。

テスト開始時に保存した HEW の環境を読み込みことで、毎回同じ環境でテストを開始できます。

- 本テストシナリオで使用するテストスクリプトの準備

このテストシナリオでは、マクロ生成支援機能を使用し、次の一連の操作を HEW コマンドラインのコマンドとしてマクロに記録します。

このマクロをテストスクリプトとしてテスト実行時に使用します。

操作の分類	操作	記録する HEW コマンドラインのコマンド
プログラム実行前の準備 ・各設定値の初期化 ・プログラムの準備	セッションの再読み込み （ステップ 1(2)参照）	refresh_session コマンド
	レジスタ値の初期化***1** （ステップ 1(3)参照）	register_set コマンド ***2**
	デモンストレーション用プログラムのビルド***1** （ステップ 1(4)参照）	build コマンド
プログラムダウンロード	デモンストレーション用プログラムのダウンロード （ステップ 1(5)参照）	file_load コマンド
変数の登録、プログラム実行、および実行結果の確認	メモリフィル機能で配列 a のメモリの値を 0 に設定 （ステップ 1(6)参照）	memory_fill コマンド
	CPU のリセット （ステップ 1(7)参照）	reset コマンド
	main 関数内の任意のカーソル行まで GO 実行 （ステップ 1(9)参照）	go_till コマンド
	ランダムデータを格納している配列をウォッチウィンドウへ登録***1** （ステップ 1(10)参照）	watch_add コマンド

***1.** 機能のサポートは、デバッギングプラットフォームに依存します。

***2.** エミュレータまたはシミュレータのヘルプを参照してください。

- テストシナリオの例

テスト支援機能を使用し、配列のデータ（配列のアドレスのメモリ内容）をテスト対象とします。

テストベースとなる HEW の状態のテストイメージデータをテストイメージファイルに保存した後、配列にデータを格納する回数を減らすようプログラムを変更した後にテストを実行し、配列に格納されているデータに差異が生じることをテストにより確認します。続いて格納する回数を元に戻し同じテストを実行します。最初のテストベースのテストイメージファイルと比較し配列に格納されているデータ値に差異が生じていないことを確認します。

ステップ 1： マクロの記録

ステップ 2： マクロを編集する（記録内容を見る）

ステップ 3： マクロを実行する

ステップ 4： テストスイートを作成する

ステップ 5： テストスイートを編集する

ステップ 6： テストイメージファイルを作成する

ステップ 7： プログラムを変更しテストを実行する

ステップ 8： テスト結果を確認する（不一致の場合）

ステップ 9： プログラムを元に戻しテストを実行する

ステップ 10： テスト結果を確認する（一致した場合）

1.13.2 ステップ 1： マクロの記録

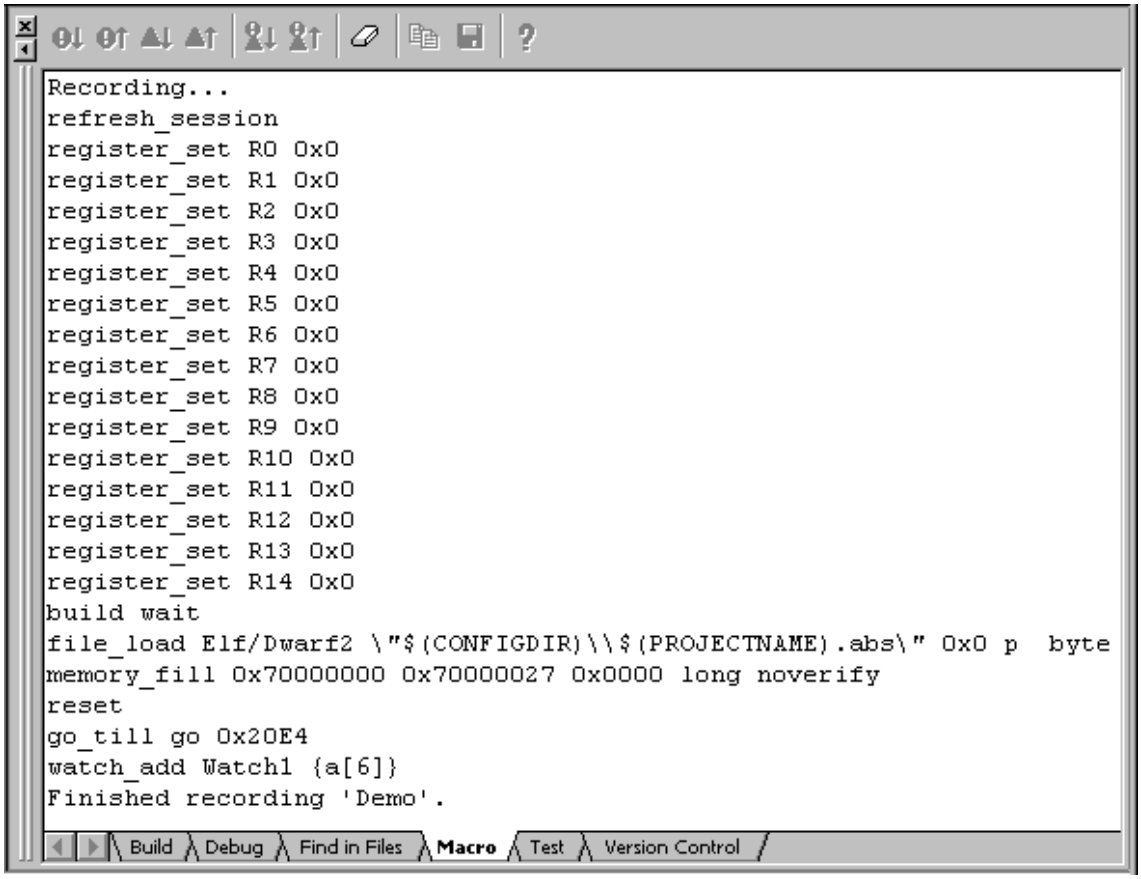
次の一連の操作でマクロ生成支援機能を使用し HEW コマンドラインのコマンドを記録します。

記録した HEW コマンドラインのコマンドは、アウトプットウィンドウの[Macro]タブに順次出力されます。ただし、操作によってはアウトプットウィンドウのタブが切り替わります。

1. [マクロ]ツールバーの[マクロの記録]ボタン（●）をクリックします。このときマウスポインタに[マクロの記録]アイコンがつきます（●）。アウトプットウィンドウの[Macro]タブに「Recording...」メッセージが出力されます。
2. [ファイル->セッションのリフレッシュ]を選択します。
   （**refresh_session コマンドが記録されます**）
3. [CPU]ツールバーの[レジスタ]ボタン（R1）をクリックしレジスタウィンドウを表示します。値を変更するレジスタ上をダブルクリックし[レジスタ値設定]ダイアログボックスを表示します。[値]に 0 を入力します。
   （**register_set コマンドが記録されます**）
4. [標準]ツールバーの[ビルド]ボタン（）をクリックしデモンストレーション用プログラムをビルドします。
   （**build コマンドが記録されます**）
5. ワークスペースウィンドウの[Projects]タブの[Download modules]フォルダ下のデモンストレーション用プログラムをダブルクリックしダウンロードします。
   （**file_load コマンドが記録されます**）
6. [CPU]ツールバーの[メモリ]ボタン（）をクリックし[表示開始アドレス]ダイアログボックスを表示します。[表示開始アドレス]に配列 a のアドレスを入力しメモリウィンドウを表示します。[メモリ]ウィンドウ上で充填する配列 a のアドレス領域をドラッグして選択してください。ウィンドウ内で右クリックしポップアップメニューから[フィル]を選択します。[メモリフィル]ダイアログボックスのフィルデータに 0 を入力します。
   （**memory_fill コマンドが記録されます**）

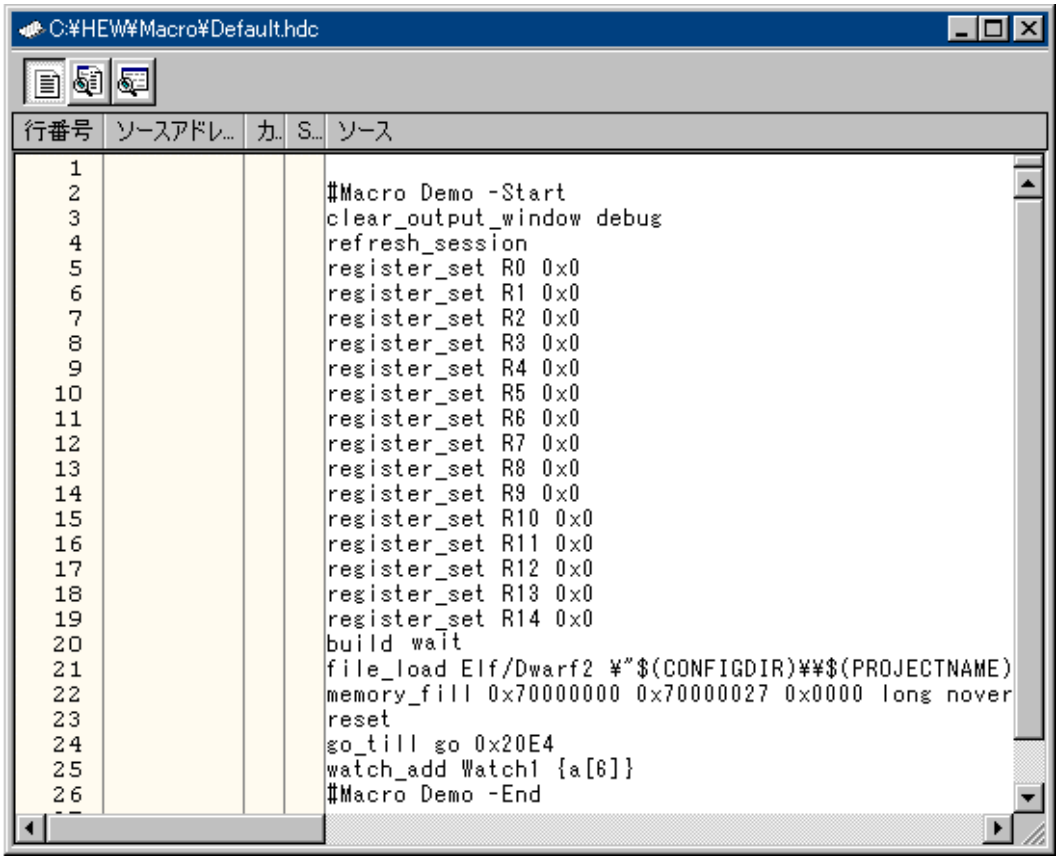
7. [デバッグラン]ツールバーの[CPU リセット]ボタン（）をクリックし CPU をリセットします。
   （**reset コマンドが記録されます**）
8. main 関数を含むソースファイルをエディタウィンドウに表示します。
9. main 関数の最終行上で右クリックしポップアップメニューから[カーソル位置まで実行]を選択します。
   （**go_till コマンドが記録されます**）
10. ランダムデータを格納している配列要素"a[6]"をウォッチウィンドウへ追加します。
    （**watch_add コマンドが記録されます**）
11. 最後に[マクロ]ツールバーの[マクロの停止]ボタン（□）をクリックします。マウスポインタは通常の矢印に戻ります。
12. マクロ名を入力する[マクロの新規追加]ダイアログボックスが表示されますので、マクロ名（例：”Demo”）を入力してください。アウトプットウィンドウの[Macro]タブに「Finished recording 'Demo'.」が出力されます。

アウトプットウィンドウの[Macro]タブを表示してください。記録した HEW コマンドラインのコマンドが確認できます。



1.13.3 ステップ 2： マクロを編集する（記録内容を見る）

1. [ツール->マクロの設定]を選択し[マクロの設定]ダイアログボックスを表示します。
2. マクロを記録した HEW マクロファイル”Default”が[使用中のマクロファイル]リストに表示されています。
3. [マクロ一覧]リストから”Demo”マクロを選択し、[編集]ボタンをクリックします。
4. エディタウィンドウに”Demo”マクロが記録された HEW のマクロファイル”Default.hdc”が表示されます。



HEW のマクロファイルには、マクロ名を識別する情報行の間に下記の操作に対応する HEW コマンドラインのコマンドが記録されています。

操作	記録する HEW コマンドラインのコマンド
セッションの再読み込み	refresh_session コマンド
レジスタ値の初期化***1**	register_set コマンド***2**
デモンストレーション用プログラムのビルド***1**	build コマンド
デモンストレーション用プログラムのダウンロード	file_load コマンド
メモリフィル機能でデータ領域のメモリの値を 0 に設定	memory_fill コマンド
CPU のリセット	reset コマンド
main 関数内の任意のカーソル行まで GO 実行	go_till コマンド
ランダムデータを格納している配列をウォッチウィンドウへ登録***1**	watch_add コマンド

***1.** 機能のサポートは、デバッギングプラットフォームに依存します。

***2.** エミュレータまたはシミュレータのヘルプを参照してください。

プログラム実行などを含むマクロの場合、アウトプットウィンドウの[Debug]タブには停止要因などの情報が表示されます。そのため、テスト結果を比較する際に差異が生じる場合があります。アウトプットウィンドウの各タブの表示内容をクリアする HEW のコマンドラインの clear_output_window コマンドがあります。あらかじめ”#Macro Demo -Start”の次の行にアウトプットウィンドウの[Debug]タブの情報をクリアする以下の行を追加することも可能です。

clear_output_window debug

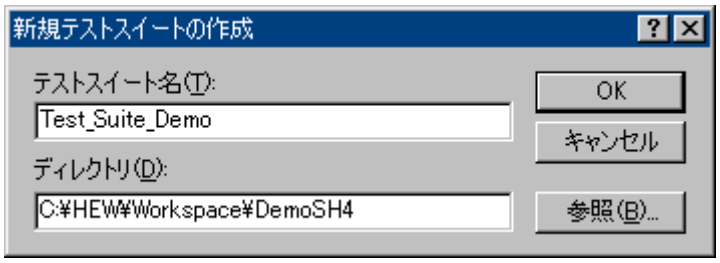
必要に応じて内容を編集後、[エディタ]ツールバーの[ファイルの保存]ボタン（）をクリックしてください。HEW のマクロファイル”Default.hdc”を閉じてください。

1.13.4 ステップ 3： マクロを実行する

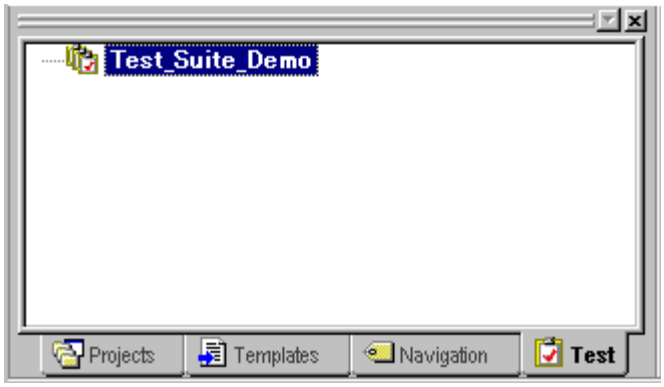
1. [マクロ]ツールバーの[マクロの実行]ボタン（）をクリックします。[マクロの選択]ダイアログボックスが表示されます。
2. ”Demo”マクロを選択し OK ボタンをクリックすると、記録したコマンドの実行を開始します。
3. このときマウスポインタに[マクロの実行]アイコンがつきます（）。
4. マクロの実行が終了するとマウスポインタは通常の矢印に戻ります。

1.13.5 ステップ 4： テストスイートを作成する

1. [テスト->テストスイートの作成]を選択し、[新規テストスイートの作成]ダイアログボックスを表示します。
2. [テストスイート名]に”Test_Suite_Demo”を入力します。



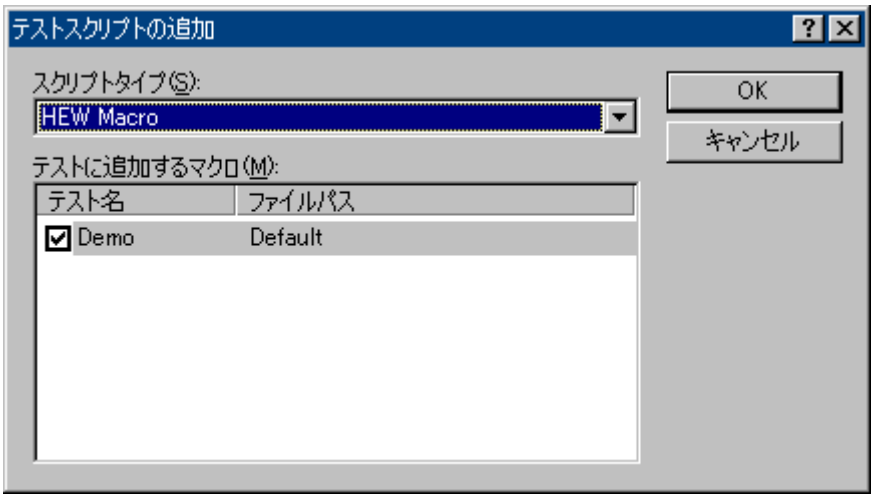
3. [ディレクトリ］はあらかじめワークスペースディレクトリが表示されています。必要に応じて編集してください。
4. [OK]ボタンをクリックします。
5. ワークスペースウィンドウの[Test]タブを表示してください。”Test_Suite_Demo”テストスイートアイコンが表示されています。



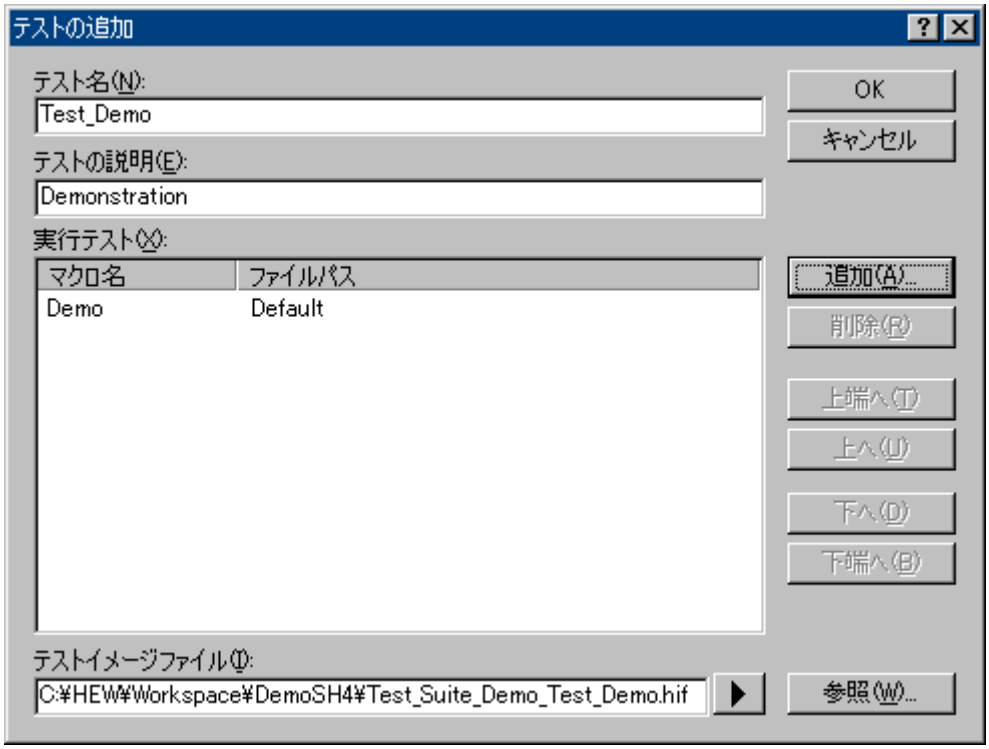
1.13.6 ステップ 5： テストスイートを編集する

テストを実行するために「ステップ 1」で記録したマクロを選択します。

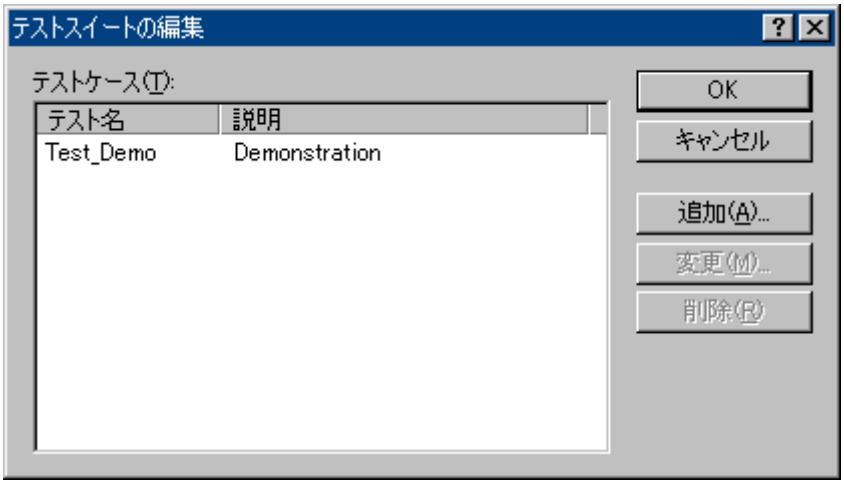
1. ”Test_Suite_Demo”テストスイートアイコン上で右クリックしポップアップメニューを表示します。
2. [テストスイートの編集]を選択し、[テストスイートの編集]ダイアログボックスを表示します。
3. [追加]ボタンをクリックし[テストの追加]ダイアログボックスを表示します。
4. [テスト名]に”Test_Demo”と入力します。
5. [テストの説明]に”Demonstration”と入力します。
6. [追加]ボタンをクリックし[テストスクリプトの追加]ダイアログボックスを表示します。
7. [テストに追加するマクロ]で”Demo”チェックボックスをオンにします。OK ボタンをクリックします。



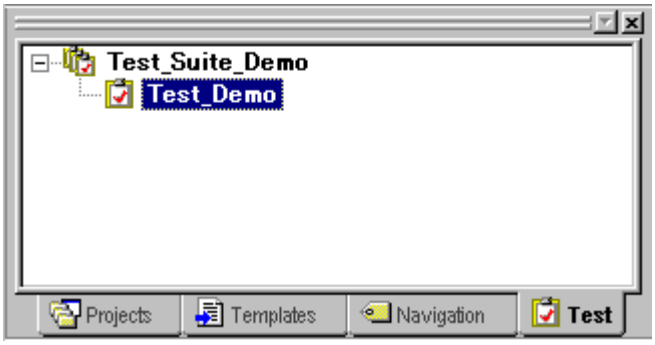
8. [テストイメージファイル]には、あらかじめ「ステップ 4」で入力したテストスイート名に'_'が付加されて表示されています。'_'に続いて[テスト名]に入力したテスト名”Test_Demo”が表示されます（"テストスイート名"+'_'+"テスト名"）。必要に応じて編集してください。



9. OK ボタンをクリックします。［テストケース]リストに”Test_Demo”テストが追加されました。



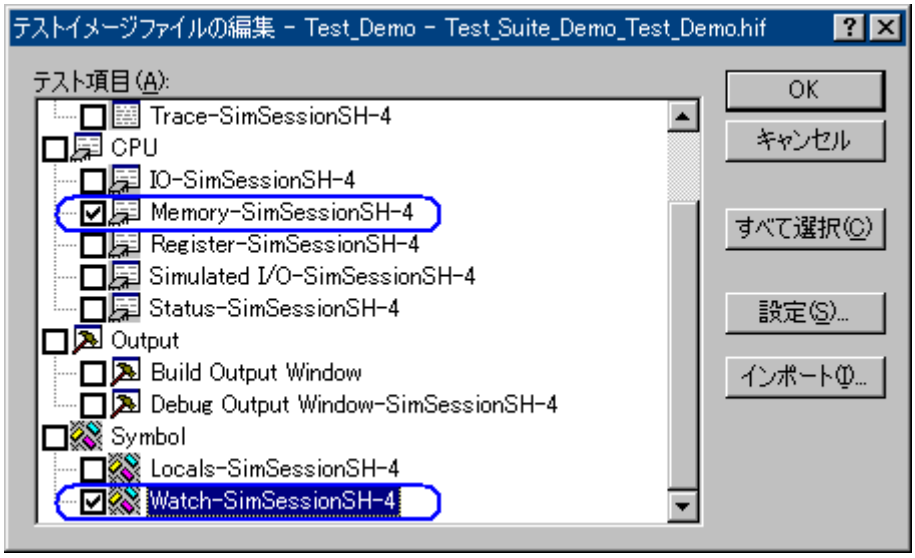
10. OK ボタンをクリックします。
11. ”Test_Suite_Demo”テストスイートアイコン下に”Test_Demo”テストアイコンが表示されました。



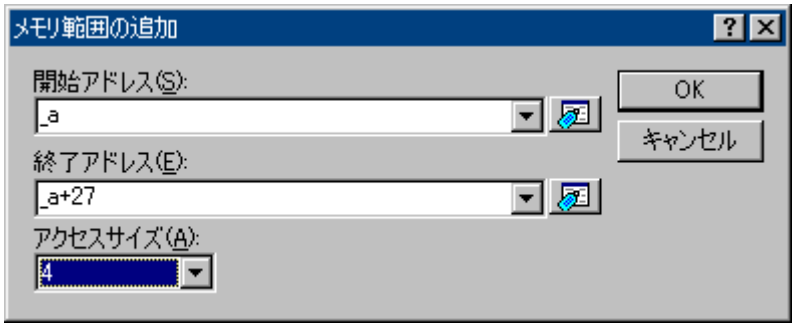
1.13.7 ステップ 6： テストイメージファイルを作成する

テスト項目として配列のアドレスのメモリ内容、およびウォッチウィンドウの内容を比較するため、それらのテストイメージデータの取得範囲を設定します。ステップ 5 で指定したテストイメージファイル"Test_Suite_Demo_Test_Demo.hif"にテスト項目のテストイメージデータが保存されます。

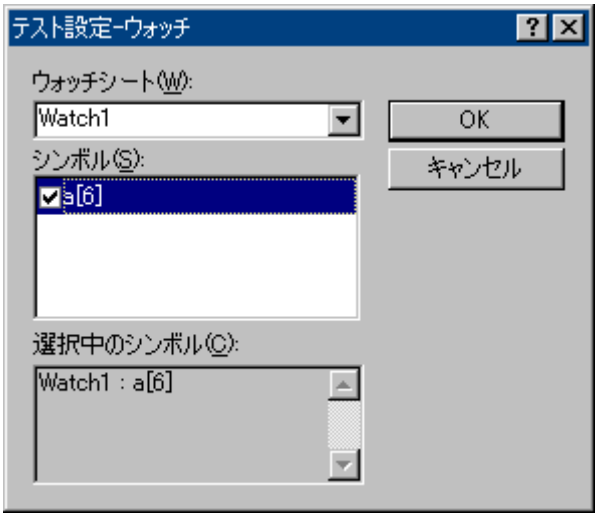
1. ”Test_Demo”テストアイコン上で右クリックしポップアップメニューを表示します。
2. [テストイメージファイルの編集]を選択し[テストイメージファイルの編集]ダイアログボックスを表示します。ダイアログボックスのタイトルには、テスト名”Test_Demo”およびテストイメージファイル名”Test_Suite_Demo_Test_Demo.hif”が表示されます。
3. [テスト項目]の[CPU]下の[Memory-xxxxxx]および[Symbol]下の[Watch-xxxxxx]のチェックボックスをオンにします。



4. 配列のアドレスのメモリ内容をテストイメージファイルに保存します。[テストイメージファイルの編集]ダイアログボックスで、[テスト項目]の[CPU]下の[Memory-xxxxxx]（xxxxxx は接続しているセッション名）をダブルクリックし、[テスト設定-メモリ]ダイアログボックスを表示します。
5. [追加]ボタンをクリックし、[メモリ範囲の追加]ダイアログボックスを表示します。
6. [開始アドレス]に配列の先頭アドレス、[終了アドレス]に配列の終了アドレスを入力し、[アクセスサイズ]を選択します。OK ボタンをクリックします。



7. [テスト設定-メモリ]ダイアログボックスで OK ボタンをクリックします。
8. ウォッチウィンドウの Watch1 シートの配列要素"a[6]"の内容をテストイメージファイルに保存します。[テストイメージファイルの編集]ダイアログボックスで、[テスト項目]の[Symbol]下の[Watch-xxxxxx]をダブルクリックし、[テスト設定-ウォッチ]ダイアログボックスを表示します。[シンボル]の"a[6]"のチェックボックスをオンにします。OK ボタンをクリックします。

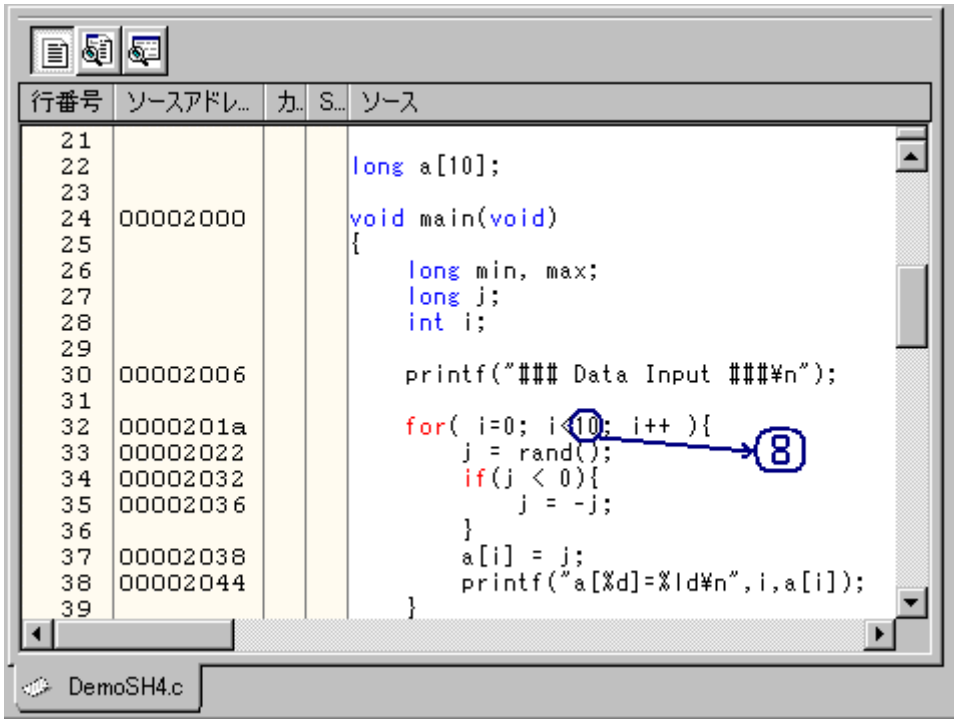


9. [テストイメージファイルの編集]ダイアログボックスで OK ボタンをクリックします。

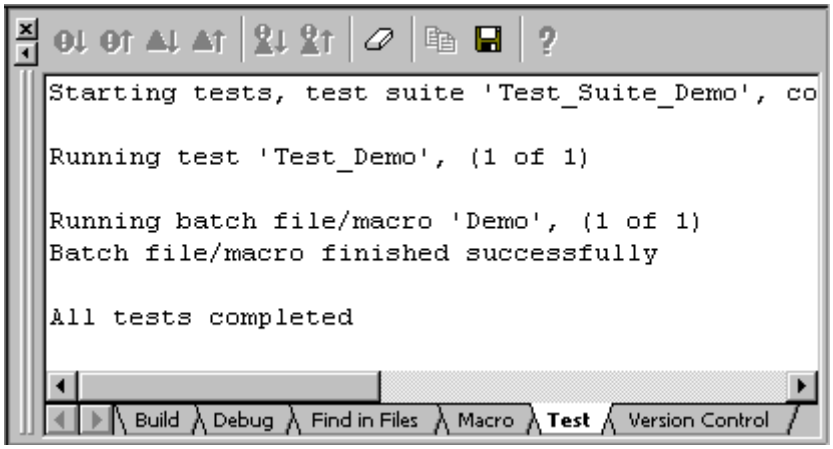
この時点の HEW システムの状態で、テスト項目の指定内容に基づいてテストイメージを取得し、ステップ 5 で生成したテストイメージファイルに保存されます。

1.13.8 ステップ 7： プログラムを変更しテストを実行する

この例では配列のメモリ領域の値をテスト対象としています。配列にデータを格納する回数を減らすようにプログラムを変更し、保存されたテストイメージと実際のメモリの値が不一致になる状態にします。



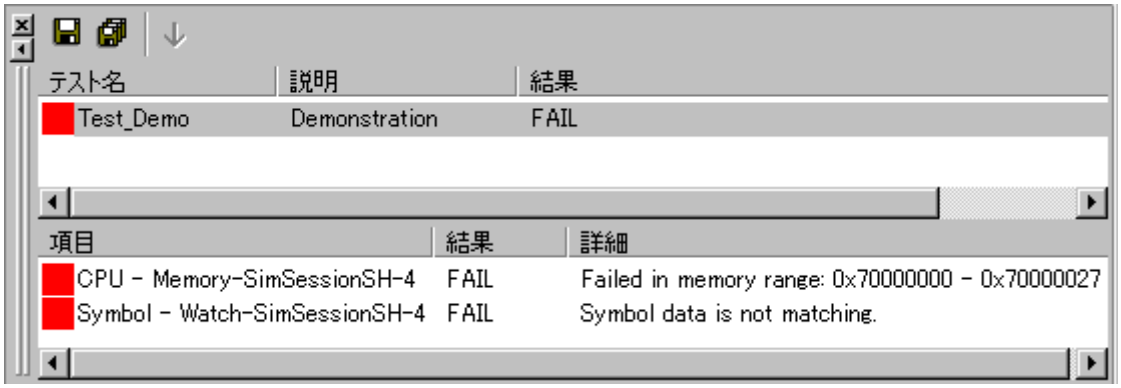
1. エディタウィンドウに main 関数を含むソースファイルを表示し配列にデータを格納する回数を減らすようにプログラムを変更します。
2. [エディタ]ツールバーの[ファイルの保存]ボタン（）をクリックします。
3. ”Test_Demo”テストアイコン上で右クリックしポップアップメニューを表示します。
4. [テストケースの実行]を選択しテストを開始します。
5. テストの進捗状況および結果がアウトプットウィンドウの[Test]タブに表示されます。



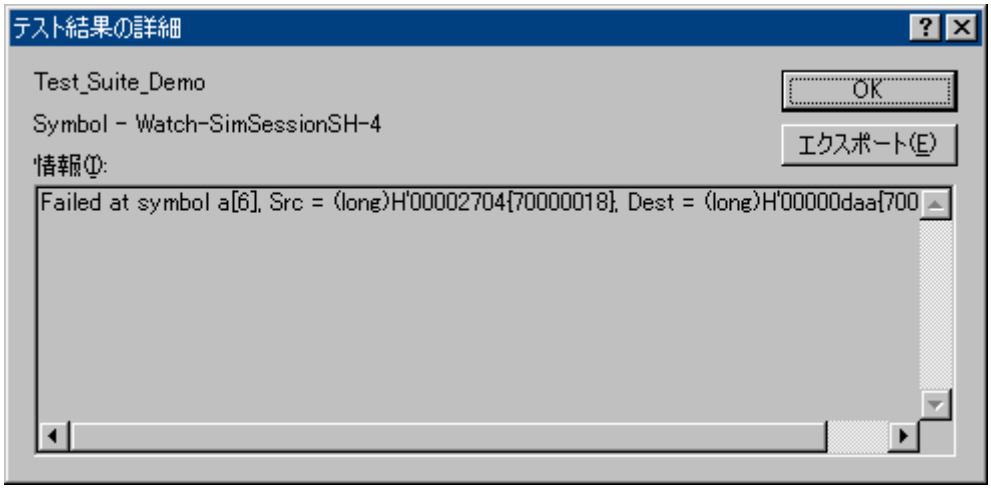
6. テストが完了すると、テストブラウザウィンドウが表示されます。

1.13.9 ステップ 8： テスト結果を確認する（不一致の場合）

テスト結果はテストブラウザウィンドウで確認できます。



1. 下側のペインで赤いマークの付いたテスト項目をダブルクリックすると不一致データの詳細を確認できます。
2. 例えば、[Symbol - Watch-xxxxxx]行をダブルクリックすると、次のようなテスト結果の詳細ダイアログボックスを表示します。配列要素のテストイメージファイルのテストイメージの値（Src=xxxxxxxxx）と今回のテストのテストイメージの値（Dest=xxxxxxxx）を表示します。



3. [エクスポート]ボタンをクリックすると、テスト項目の結果をテキストファイルへ保存することが可能です。
4. OK ボタンをクリックします。

1.13.10　ステップ 9： プログラムを元に戻しテストを実行する

配列にデータを格納する回数を元に戻すようプログラムを変更し、テストベースとテスト実行結果のメモリが一致する状態にします。

1. 配列にデータを格納する回数を 8 から 10 にプログラムを戻します。
2. [エディタ]ツールバーの[ファイルの保存]ボタン（ ）をクリックします。
3. ”Test_Demo”テストアイコン上で右クリックしポップアップメニューを表示します。
4. [テストケースの実行]を選択しテストを開始します。
5. テストが完了すると、テストブラウザウィンドウが表示されます。

1.13.11　ステップ 10： テスト結果を確認する（一致した場合）

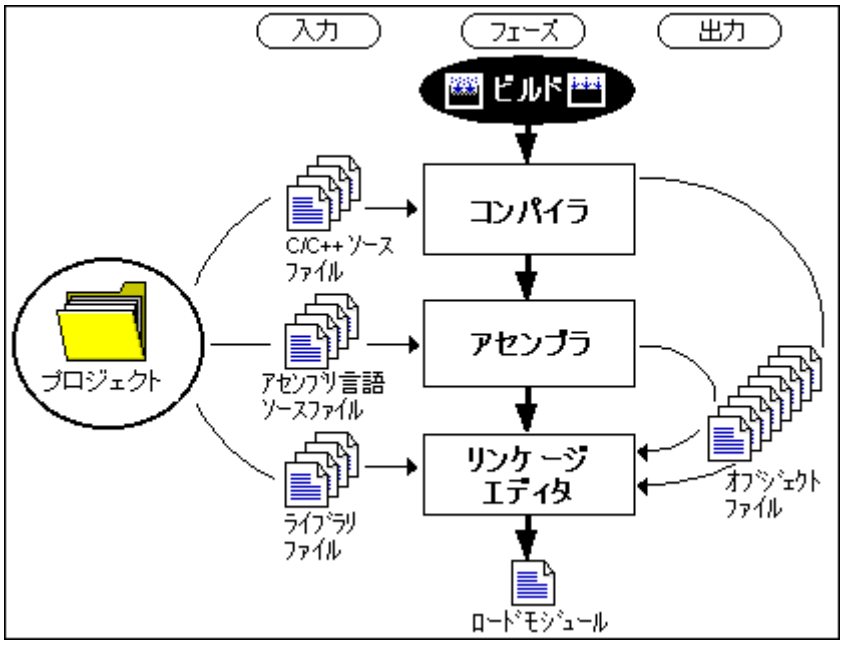
テスト結果はテストブラウザウィンドウで確認できます。テスト結果が一致しています。

2. ビルドの基本

この章では High-performance Embedded Workshop (HEW) の一般的な機能を説明します。より高度な機能については「3. ビルドの応用」を参照してください。

2.1 ビルド処理

ビルド処理の一般的な流れを以下の図に示します。HEW のインストール時に提供されるツールによってビルド処理は異なるため、以下の図とも異なる場合があります（例えば、コンパイラが無いなど）。ビルドの各フェーズにおいて、1 セットのプロジェクトファイルについてビルド処理を行います。それが完了すると、次のフェーズに移ります。

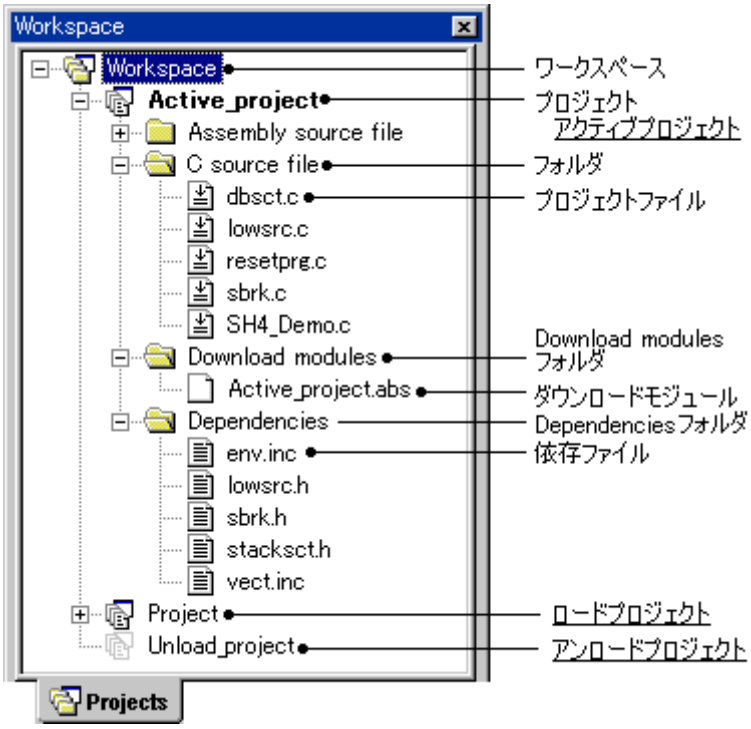


上記の例では、第一のフェーズがコンパイラ、第二のフェーズがアセンブラ、そして最後のフェーズがリンケージエディタです。コンパイラのフェーズでは、プロジェクトの C/C++ソースファイルを順次コンパイルします。アセンブラのフェーズでは、アセンブリ言語のソースファイルを順次アセンブルします。リンケージエディタのフェーズでは、すべてのライブラリファイルと、コンパイラフェーズとアセンブラフェーズからの出力ファイルをリンクして、ロードモジュールを作成します。

ビルド処理をカスタマイズする方法はいくつかあります。例えば、独自のフェーズを追加したり、あるフェーズを無効にしたり、フェーズを削除できます。これらのビルド実行の応用については、「3. ビルドの応用」を参照してください。

2.2 ワークスペースウィンドウの Projects タブの構成

ワークスペースウィンドウの[Projects]タブには現在のワークスペース、プロジェクト、ファイルを示します。アイコンをダブルクリックしてプロジェクトファイルや個々のファイルを開くことができます。



ワークスペース上で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

ポップアップメニュー項目	マクロの記録	機能
プロジェクトの挿入	-	ワークスペースにプロジェクトを追加します。
全プロジェクトの読み込み	-	すべてのプロジェクトを読み込みます。
ツリーの縮小	-	ワークスペースアイコン下のツリーを縮小します。
クリーン　全プロジェクト	●	すべてのプロジェクトでコンフィグレーションの中間ファイルおよび出力ファイルを削除します。
バージョン管理	-	バージョン管理システムを実行します。
表示の構成	-	ワークスペースの表示の構成を設定します。
プロパティ	-	ワークスペースのプロパティを表示します。

プロジェクト上で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

ポップアップメニュー項目		マクロの記録	機能
ビルド	ビルド	●	プロジェクトをビルドします。
	すべてをビルド	●	プロジェクトを全ビルドします。
	クリーン　アクティブプロジェクト	●	現在のコンフィグレーションの中間ファイルおよび出力ファイルを削除します。
	すべての依存関係を更新	-	プロジェクトのファイル依存関係を更新します。
アクティブプロジェクトに設定		●	プロジェクトをアクティブにします。

プロジェクトの削除	-	ワークスペースからプロジェクトを削除します。
プロジェクトのアンロード または プロジェクトの読み込み	-	プロジェクトをアンロードします。 または プロジェクトを読み込みます。
ファイルの追加	-	プロジェクトにファイルを追加します。
ファイルの削除	-	プロジェクトからファイルを削除します。
フォルダの追加	-	プロジェクトにフォルダを追加します。
ツリーの展開/縮小	-	プロジェクト下のツリーを展開または縮小します。
バージョン管理	-	バージョン管理システムを実行します。
表示の構成	-	ワークスペースの表示の構成を設定します。
プロパティ	-	プロジェクトのプロパティを表示します。

フォルダ（Download module フォルダ、および Dependencies フォルダを除く）上で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

ポップアップメニュー項目	マクロの記録	機能
フォルダの追加	-	フォルダを追加します。
フォルダ名の変更	-	フォルダ名を変更します。
フォルダの削除	-	フォルダを削除します。
表示の構成	-	ワークスペースの表示の構成を設定します。

プロジェクトファイル上で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

ポップアップメニュー項目	マクロの記録	機能
開く <ファイル名>	-	HEW エディタでファイルを開きます。
開く < ファイル名> （外部エディタ） ***1**	-	外部エディタでファイルを開きます。
コンパイル <ファイル名>	●	1 つのファイルをコンパイルします。
ビルドオプション	-	ファイルのビルド方法を設定します。
ファイルの追加	-	プロジェクトにファイルを追加します。
ファイルの削除	-	プロジェクトからファイルを削除します。
ビルドから除外 <ファイル名> または ビルドから除外の解除 <ファイル名>	-	ビルドからプロジェクトファイルを除外します。 または、 除外したプロジェクトファイルをビルドに再び入れます。
バージョン管理	-	バージョン管理システムを実行します。
表示の構成	-	ワークスペースの表示の構成を設定します。
ファイルの比較	-	ファイルを比較します。
プロパティ	-	ファイルのプロパティを表示します。

***1.** [基本設定->オプション]を選択すると表示される[オプション]ダイアログボックスの[エディタ]タブで、[外部エディタの使用]チェックボックスをオンにすると表示されます。

[Download module]フォルダ上で右クリックすると、使用可能なオプションをポップアップメニューで表示します。（デバッギングプラットフォーム接続時に使用可能）

ポップアップメニュー項目	マクロの記録	機能
全てダウンロード	●	[デバッグの設定]ダイアログボックスの[ターゲット]タブの[ダウンロードモジュール]リストに登録されているすべてのモジュール（ワークスペースウィンドウの[Projects]タブの[Download Modules]フォルダ下に表示されているダウンロードモジュールと同じです）をダウンロードします。
ダウンロードモジュールの追加	●	[ダウンロードモジュール]ダイアログボックスを表示し、ダウンロードモジュールを追加します。このモジュールをダウンロードします。

デバッグの設定	-	[デバッグの設定]ダイアログボックスを表示し、デバッグ設定を編集します。
表示の構成	-	ワークスペースの表示の構成を設定します。

ダウンロードモジュール上で右クリックすると、使用可能なオプションをポップアップメニューで表示します。（デバッギングプラットフォーム接続時に使用可能）

ポップアップメニュー項目	マクロの記録	機能
ダウンロード	●	モジュールを手動でダウンロードします。
ダウンロード(debug 情報のみ)	●	debug 情報のみモジュールを手動でダウンロードします。
アンロード	●	モジュールをアンロードします。
ダウンロードモジュールの追加	●	[ダウンロードモジュール]ダイアログボックスを表示し、ダウンロードモジュールを追加します。このモジュールをダウンロードします。
削除	-	選択したモジュールを削除します。
デバッグの設定	-	[デバッグの設定]ダイアログボックスを表示し、デバッグ設定を編集します。
表示の構成	-	ワークスペースの表示の構成を設定します。
モジュールの検索 ***2**	-	ダウンロードモジュールから取得したすべてのファイルを別のディレクトリで再検索するためのダイアログボックスを表示します。
プロパティ	-	選択したダウンロードモジュールの設定内容を[ダウンロードモジュール]ダイアログボックスに表示します。選択したダウンロードモジュールの設定内容を編集します。ダウンロード済みの場合は、アンロードされます。

***2**. デバッグ専用プロジェクト（[Debugger only - xxxxxx]）を使用している場合のみ表示されます。

ダウンロードモジュールから取得したファイル上で右クリックすると、使用可能なオプションをポップアップメニューで表示します。（デバッグ専用プロジェクト（[Debugger only - xxxxxx]）の場合に使用可能）

ポップアップメニュー項目	マクロの記録	機能
開く <ファイル名>	-	HEW エディタでファイルを開きます。
ファイルの追加	-	プロジェクトにファイルを追加します。
表示の構成	-	ワークスペースの表示の構成を設定します。
ファイルの検索	-	ダウンロードモジュールから取得した同じファイル名を別のディレクトリで選択するためのダイアログボックスを表示します。
プロパティ	-	ファイルのプロパティを表示します。

[Dependencies]フォルダ上で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

ポップアップメニュー項目	マクロの記録	機能
表示の構成	-	ワークスペースの表示の構成を設定します。

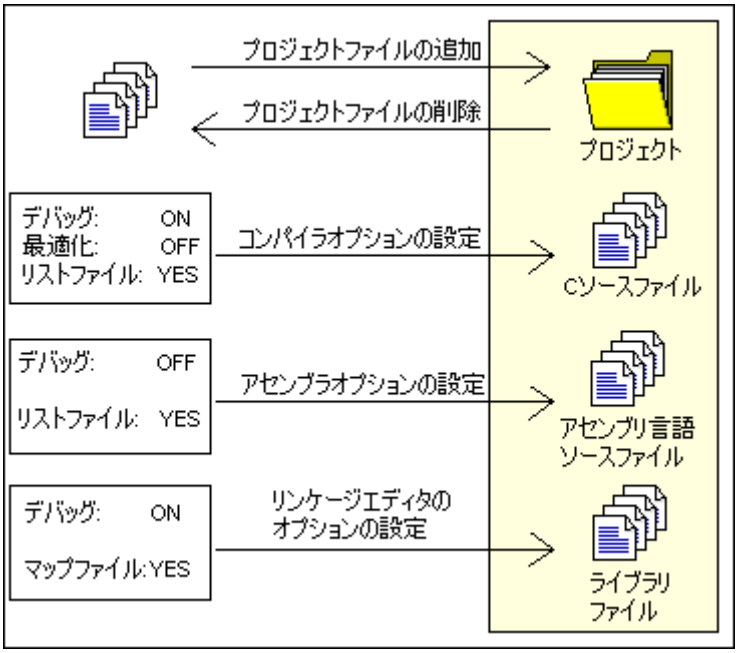
依存ファイル上で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

ポップアップメニュー項目	マクロの記録	機能
バージョン管理	-	バージョン管理システムを実行します。
表示の構成	-	ワークスペースの表示の構成を設定します。
ファイルの比較	-	ファイルを比較します。
プロパティ	-	ファイルのプロパティを表示します。

アクティブプロジェクト、ロードプロジェクト、アンロードプロジェクトについては「2.12 アクティブプロジェクトを設定する」を参照してください。

2.3 プロジェクトファイル

HEW を使ってアプリケーションをビルドするには、まず、どのファイルをプロジェクトに追加して、各ファイルをどのようにビルドすべきかを指定しなければなりません(以下の図を参照)。



2.3.1 プロジェクトにファイルを追加する

アプリケーションをビルドする前に、まず、アプリケーションを構成するファイルを指定します。

プロジェクトにファイルを追加するには

1. アクティブプロジェクトに対してのみファイルが追加できます。次のいずれかを行ってください。
   - [プロジェクト->ファイルの追加]を選択する。
   - ワークスペースウィンドウの[Projects]タブのプロジェクト上で右クリックし、ポップアップメニューから[ファイルの追加]を選択する。
   - ワークスペースウィンドウの[Projects]タブが選択されているときに"Insert"キーを押す。
2. ['<プロジェクト名>'プロジェクトにファイルを追加]ダイアログボックスを表示します。
3. 追加するファイルを選択してください。
4. ['<プロジェクト名>'プロジェクトにファイルを追加]ダイアログボックスの一番下にある[相対パス]チェックボックスをオンにすると、ファイルを「相対プロジェクトファイル」にできます。これにより、プロジェクトファイルがワークスペースファイルに相対的になり、ワークスペース構造の外に置くことも可能です。ソースツリー全体を再

配置すると、HEW はファイルの相対位置をチェックしファイルを検索できます。このチェックボックスのデフォルトはオンです。

5. ['<プロジェクト名>'プロジェクトにファイルを追加]ダイアログボックスの一番下にある[登録済みファイルを非表示]チェックボックスをオンにすると、現在のプロジェクトに追加されていないファイルだけを表示します。このチェックボックスのデフォルトはオフです。
6. [追加]ボタンをクリックしてください。

プロジェクトにファイルを追加するには、他にも方法があります。これを以下に説明します。

- エディタウィンドウ内で開いたファイルを右クリックするとポップアップメニューを表示します。ファイルがすでにプロジェクト内にある場合、[プロジェクトにファイルの追加]メニューオプションは使用できません。[プロジェクトにファイルの追加]を選択すると、現在のプロジェクトにファイルを追加できます。
- HEW では、Windows(R) Explorer からワークスペースウィンドウの[Projects]タブにファイルおよびフォルダをドラッグアンドドロップできます。次節の「2.3.2 ファイルおよびフォルダをドラッグアンドドロップする」を参照してください。

注：

HEW の認識できない形式のファイルを指定しても、プロジェクトに追加します。このファイルに関して、いくつかの機能が使用できなくなります。エディタ内でファイルを開く代わりにワークスペースウィンドウ内でこのファイルをダブルクリックすると、ファイルを開く動作を Windows®オペレーティングシステムに受け渡します。ファイルを Windows® Explorer 内で開いたときと同じように、ファイルを開くデフォルト動作が実行されます。現在、定義されている拡張子については、[ファイル拡張子]ダイアログボックスをご覧ください。詳細は、「2.5 ファイル拡張子とファイルグループ」を参照してください。

2.3.2 ファイルおよびフォルダをドラッグアンドドロップする

ワークスペースウィンドウの[Projects]タブに Windows® Explorer からファイル（あるいは複数のファイル）およびフォルダをドラッグできます。この場合、アクティブプロジェクトのプロジェクトまたはアクティブプロジェクトのユーザフォルダ上にのみドロップできます。複数のフォルダはサポートしていません。

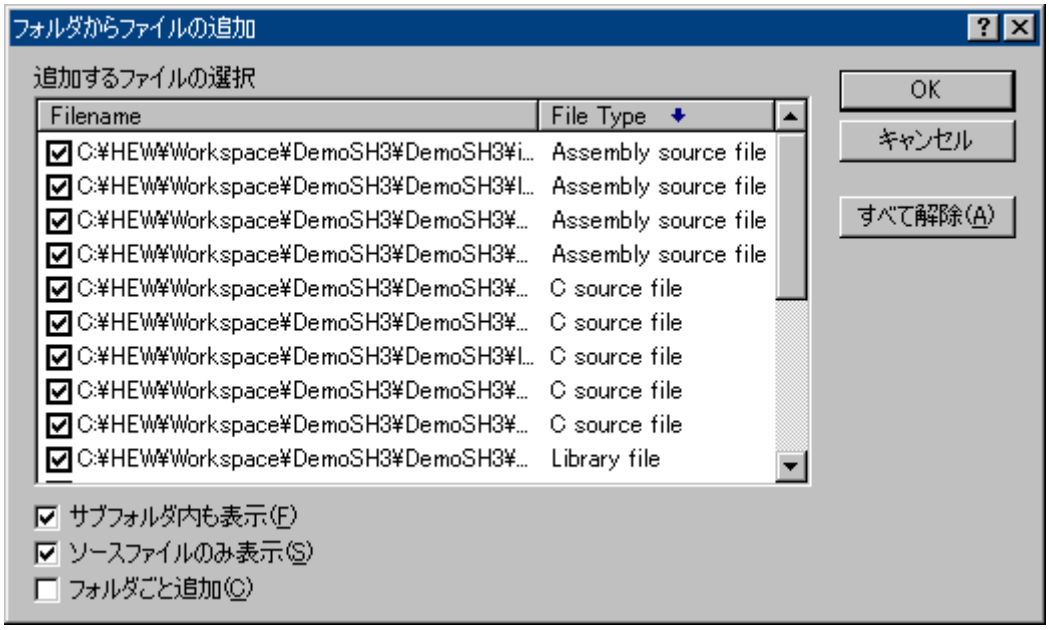
ファイルをドラッグアンドドロップする

ファイルのドロップ先により次のようになります。

- ファイルをアクティブプロジェクト上にドロップすると、ファイルは、そのファイルタイプのグループフォルダの下に追加されます。グループフォルダが存在しない場合は作成されます。例えば、ファイル「test.c」をアクティブプロジェクト上にドロップすると、「C source file」グループフォルダに追加されます。
- ファイルをユーザフォルダ上にドロップすると、ファイルはそのフォルダ下に直接追加されます。同じ名前のファイルでも異なるパスのファイルは追加されます。
- HEW の認識できない形式のファイルをアクティブプロジェクト上にドロップすると、プロジェクト下に直接追加されます。

フォルダをドラッグアンドドロップする

Windows® Explorer のフォルダをドロップすると、[フォルダからファイルの追加]ダイアログボックスを表示します。どのファイルを追加するかを選択できます。このダイアログボックスは、ドロップされたフォルダに含まれていたすべてのファイルのファイル名を表示します。各リスト項目はファイルのフルファイル名とファイルタイプを表示します。



1. リストのカラムヘッダをクリックすると、リスト項目はそのカラムによってソートされます。ダイアログボックスを表示すると、ファイルタイプ（昇順）、ファイル名（昇順）の順にソートされています。
2. リストは複数選択できます。選択内のチェックボックスのクリックまたは Enter キーの入力は、選択されたすべての項目をオンにするかオフにします(クリックされたチェックボックスの元の状態に依存します)。
3. [サブフォルダ内も表示]チェックボックスをオフにすると、ドロップされたフォルダのファイルだけが表示されます。
   オンにするとドロップされたフォルダのサブフォルダのファイルも表示されます。
   チェックボックスのデフォルトはオンです。
4. [ソースファイルのみ表示]チェックボックスをオンにすると、このプロジェクトのソースファイルとして登録された拡張子のファイルだけ(例えば「C source file」など)が表示されます。
   オフにするとファイルタイプがすべて表示されます。HEW の認識できない形式のファイルタイプについては、ダイアログボックスのファイルタイプカラムは[不明な形式]になります。
   チェックボックスのデフォルトはオンです。
5. [フォルダごと追加]チェックボックスをオンにすると、ファイルはツリー構造でドロップされたツリー下に追加されます。例えば、「data」という別のフォルダでドロップするフォルダ下にファイルがあると、「data」と呼ばれるユーザフォルダがプロジェクトに追加されます。ファイルはこの新しいユーザフォルダ下に追加されます。
   オフにすると、フォルダをユーザフォルダ上にドロップした場合、追加ファイルはす

べてこのフォルダ下に追加されます。フォルダをアクティブプロジェクトのツリー上にドロップすると、追加ファイルは、そのファイルタイプのグループフォルダの下に追加されます（ファイル「main.c」の場合、このファイルはプロジェクト下の「C source file」グループフォルダに追加されます）。グループフォルダが存在しない場合は作成されます。
チェックボックスのデフォルトはオフです。

6. [すべて解除]ボタンをクリックするとすべてのチェックボックスがオフになります。[すべて選択]ボタンをクリックするとすべてのチェックボックスがオンになります。
7. [OK]ボタンをクリックすると、そのチェックボックスをオンにしたすべてのファイルはプロジェクトに追加されます。このダイアログボックスで[OK]ボタンをクリックし次回このダイアログボックスを開くと、チェックボックスは最後に使用されたチェック状態を保存します。

2.3.3 プロジェクトからファイルを削除する

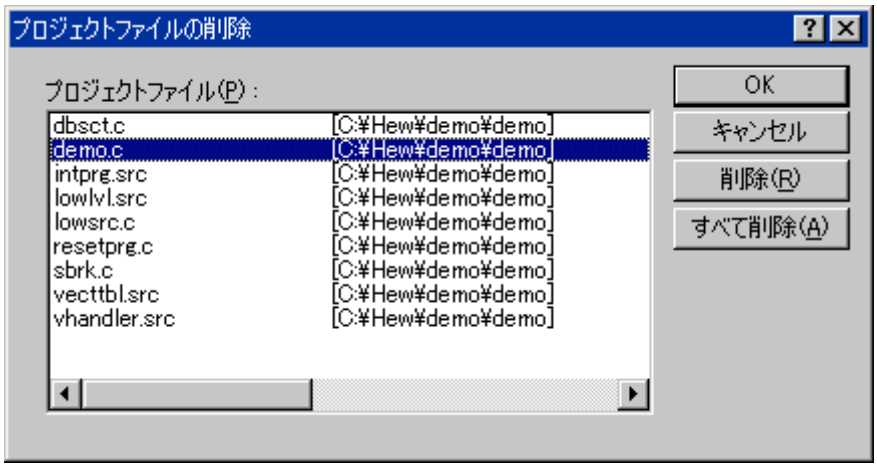
プロジェクトからファイルを削除できます。ファイルの削除は、1 つでも、複数でも、すべてのファイルをまとめてでもできます。

ファイルをプロジェクトから削除するには

1. 次のいずれかを行ってください。

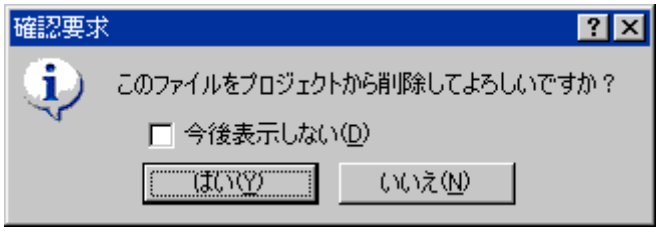
- [プロジェクト->ファイルの削除]を選択する。
- ワークスペースウィンドウの[Projects]タブのプロジェクト上で右クリックし、ポップアップメニューから[ファイルの削除]を選択する。

2. [プロジェクトファイルの削除]ダイアログボックスを表示します。
3. [プロジェクトファイル]リストから削除したいファイルを選択してください。複数のファイルを選択することもできます。
4. 選択したファイルを削除するには[削除]ボタンをクリックしてください。すべてのプロジェクトファイルを削除するには[すべて削除]ボタンをクリックしてください。
5. [OK]ボタンをクリックするとプロジェクトからファイルを削除します。



ワークスペースウィンドウの**[Projects]**タブで選択したファイルをプロジェクトから削除するには

1. ワークスペースウィンドウの[Projects]タブで削除したいファイルを選択してください。複数のファイルを選択するときは”Shift”キーまたは”Ctrl”キーを押してください。
2. “Delete”キーを押してください。
3. ファイルをプロジェクトから削除するかを確認するダイアログボックスが表示されます。ファイルをプロジェクトから削除する場合は「はい」を選択してください。



確認要求のダイアログボックスを表示しない場合、[今後表示しない]チェックボックスをオンにしてください。確認要求のダイアログボックスを再び表示するには、[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を表示します。[確認]タブの[プロジェクトからファイルを削除]チェックボックスをオンにしてください。このチェックボックスのデフォルトはオンです。

2.3.4 ビルドからプロジェクトファイルを除外する

プロジェクトのファイルはビルドから除外できます。

ビルドからプロジェクトのファイルを除外するには

1. ワークスペースウィンドウの[Projects]タブで、ビルドから除外したいファイルをクリックしてください。
   次のようにマウスを使用したり、またはキー入力で複数のファイルを選択することもできます。

- 複数のファイルを選択するには
  Ctrl キーを押しながら対象のファイルをクリックします。
- 複数のファイルを範囲で選択するには
  範囲選択を開始するファイルをクリックし、Shift キーを押しながら範囲選択を終了するファイルをクリックします。

2. 次のいずれかを行ってください。ファイルのアイコンに赤いバツ印が付き（ ）、ビルドからファイルを除外します。

- 右クリックしポップアップメニューから[ビルドから除外 <ファイル名>]を選択します。複数のファイルを選択した場合は<ファイル名>は表示されません。
- [ビルド->ビルドから除外/除外の解除]メニューオプションを選択します。

2.3.5 除外したプロジェクトファイルをビルドに再び入れる

除外したプロジェクトファイルは再びビルドに入れることができます。

除外したファイルをビルドに再び入れるには

1. ワークスペースウィンドウの[Projects]タブでファイルをクリックしてください。
   次のようにマウスを使用したり、またはキー入力で複数のファイルを選択することもできます。

- 複数のファイルを選択するには
  Ctrl キーを押しながら対象のファイルをクリックします。
- 複数のファイルを範囲で選択するには
  範囲選択を開始するファイルをクリックし、Shift キーを押しながら範囲選択を終了するファイルをクリックします。

2. 次のいずれかを行ってください。ファイルのアイコンの赤いバツ印が外れ（）、ファイルがビルド可能になります。

- 右クリックしポップアップメニューから[ビルドから除外の解除 <ファイル名>]を選択します。複数のファイルを選択した場合は<ファイル名>は表示しません。
- [ビルド->ビルドから除外/除外の解除]メニューオプションを選択します。

2.4 ワークスペースウィンドウでユーザフォルダを使う

ワークスペースウィンドウの[Projects]タブにユーザのフォルダを追加できます。これにより、プロジェクト内の特定の領域にファイルを論理的なグループにできます。フォルダ名として任意の名前をダイアログボックスで入力します。自動的に表示される[Download modules]フォルダおよび[Dependencies]フォルダに対して下記は操作できません。

ユーザフォルダを追加するには

1. ワークスペースウィンドウの[Projects]タブでプロジェクトを選択してください。
2. 右クリックし、[フォルダの追加]を選択してください。
3. [フォルダの追加]ダイアログボックスの[フォルダ]に名前を入力してください。
4. [OK]ボタンをクリックしてください。
5. ファイルをフォルダ上にドラッグアンドドロップして、フォルダの分類を変更できます。

フォルダにサブフォルダを追加するには

1. ワークスペースウィンドウの[Projects]タブでフォルダを選択してください。
2. 右クリックし、[フォルダの追加]を選択してください。
3. [フォルダの追加]ダイアログボックスの[フォルダ]に名前を入力してください。
4. [OK]ボタンをクリックしてください。
5. ファイルをフォルダ上にドラッグアンドドロップして、フォルダの分類を変更できます。

フォルダをドラッグアンドドロップするには

ワークスペースウィンドウの[Projects]タブでアクティブプロジェクトのフォルダをドラッグしてください。

- フォルダをアクティブプロジェクト上にドロップすると、フォルダはそのプロジェクト下にドロップされます。
- フォルダをアクティブプロジェクトのユーザフォルダ上にドロップすると、フォルダはそのフォルダ下にドロップされます。

Windows® Explorer のフォルダをワークスペースウィンドウの[Projects]タブにドラッグアンドドロップできます。詳細は「2.3.2 ファイルおよびフォルダをドラッグアンドドロップする」を参照してください。

ユーザフォルダを削除するには

1. ワークスペースウィンドウの[Projects]タブでフォルダを選択してください。
2. 次のいずれかを行ってください。このときフォルダは空である必要があります。

- "Delete"キーを押してください。
- 右クリックし、[フォルダの削除]を選択してください。

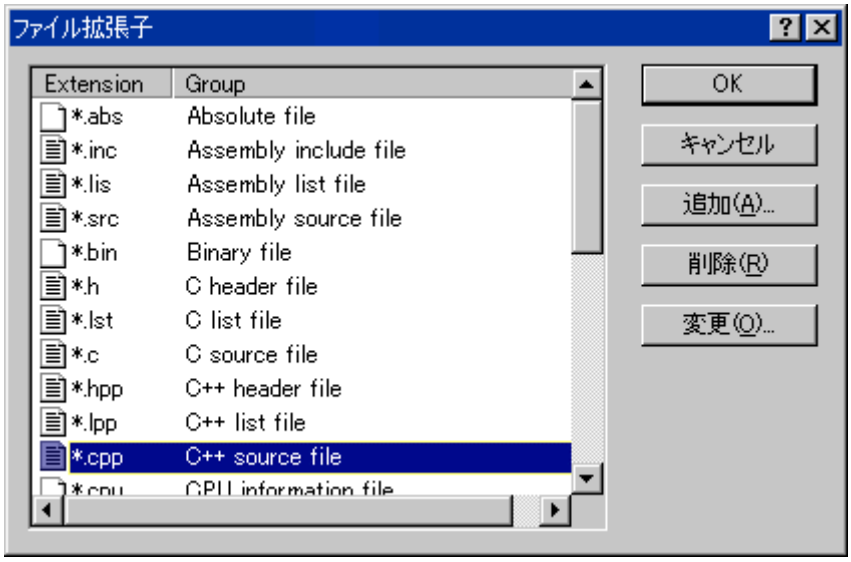
ユーザフォルダ名を変更するには

1. ワークスペースウィンドウの[Projects]タブでフォルダを選択してください。
2. 右クリックし、[フォルダ名の変更]を選択してください。
3. [フォルダ名の変更]ダイアログボックスの[フォルダ]に新しい名前を入力してください。
4. [OK]ボタンをクリックしてください。

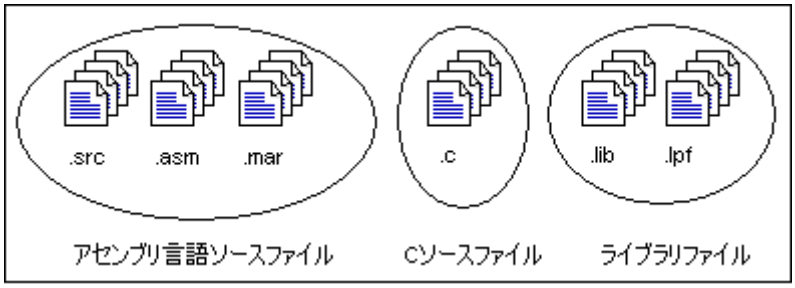
2.5 ファイル拡張子とファイルグループ

HEW は拡張子でファイルを識別します。拡張子は使用するツールによって定義します。例えば、コンパイラを使用すると拡張子.c が[C source file]グループに入り、コンパイラのフェーズに入力します。さらに、独自の拡張子を定義することもできます。例えば、プロジェクトでアセンブリ言語ソースファイルを使っている場合、デフォルトの拡張子が.src だとします。.src の代わりに違う拡張子（例: .asm）を使うとき、新しい拡張子を定義してそれを.src ファイルと同様に扱うように指定できます。

ファイルの拡張子は、[ファイル拡張子]ダイアログボックスで表示、変更できます。このダイアログボックスを表示するには[プロジェクト–>ファイルの拡張子]を選択してください。このダイアログボックスには現在のワークスペースで定義しているすべての拡張子とファイルグループを表示します。

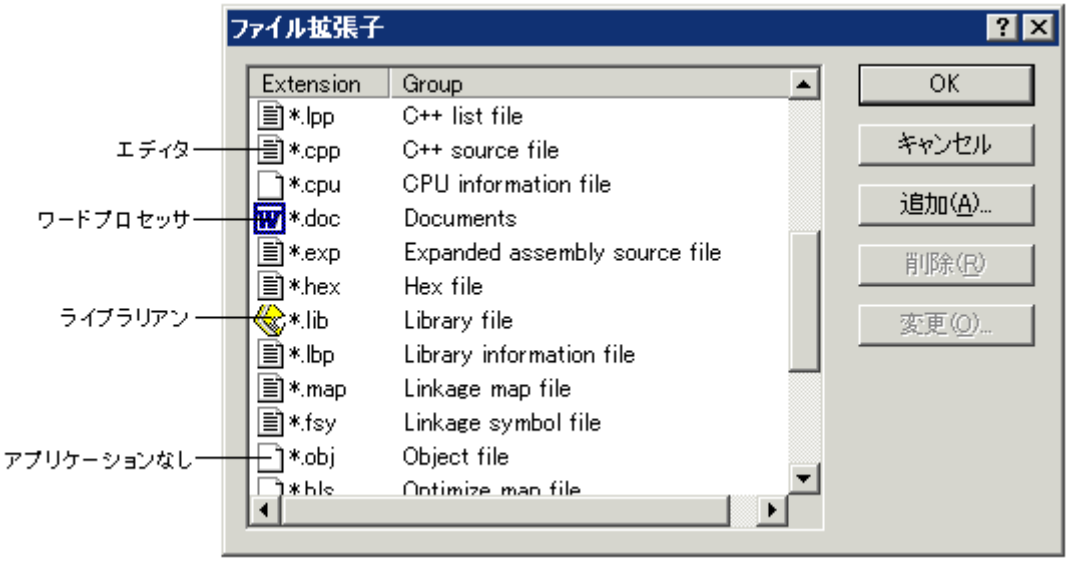


ファイル拡張子リストは 2 列に分かれています。左の列にはファイル拡張子、右の列にはファイルグループを表示します。下記の図に示すように、同じグループに多くのファイル拡張子が存在する場合があります。例えば、1 つのプロジェクト内でアセンブリ言語のソースファイルにいくつかの拡張子がある場合があります(例: .src, .asm, .mar など)。



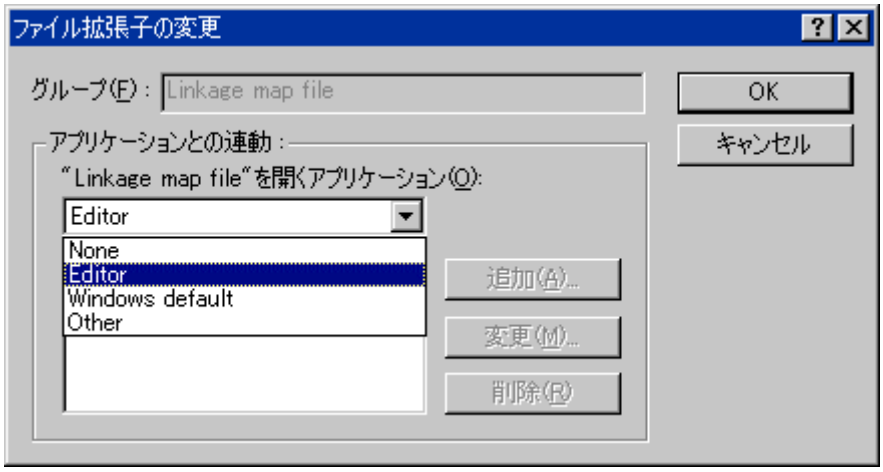
2.5.1 アプリケーションとファイルグループを関連付けるには

[ファイル拡張子]ダイアログボックスでは、エディタでファイルを開くための設定だけでなく、ファイルグループとアプリケーションとの関連付けができます。これを行うと、ワークスペースウィンドウの[Projects]タブでファイルをダブルクリックすると、適切なアプリケーションでファイルを開きます。

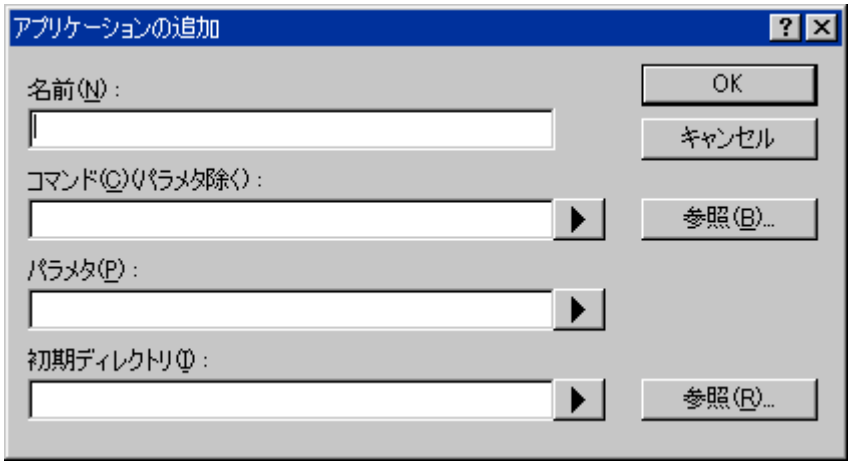


アプリケーションとファイルグループを関連付けるには

1. [プロジェクト->ファイルの拡張子]メニューオプションを選択してください。[ファイル拡張子]ダイアログボックスを表示します。
2. [ファイル拡張子]ダイアログボックスで関連付けるファイルグループを選択してください。
3. [変更]ボタンをクリックしてください。[ファイル拡張子の変更]ダイアログボックスを表示します。



4. 関連付けをしない場合は[None]を選択します。HEW のエディタまたは他のエディタでこの種類のファイルを開くには[Editor]を選択します。特定のアプリケーションでこの種類のファイルを開くには[Other]を選択します。[Other]を選択すると、すでに定義されたアプリケーションをドロップダウンリストから選択するか、または、新しいアプリケーションを定義できます。新しいアプリケーションを定義する場合は[追加]ボタンをクリックしてください。[アプリケーションの追加]ダイアログボックスを表示します。



[名前]フィールドにアプリケーション名を入力します。[コマンド]フィールドにアプリケーションのフルパスを入力します(パラメータは含めません)。[パラメタ]フィールドにファイルを開くのに必要なパラメータを入力します。必ず$(FULLFILE) プレースホルダを使って入力ファイルを指定してください。 (プレースホルダの詳細と使用方法については、「リファレンス 5. プレースホルダ」を参照してください。) [初期ディレクトリ]フィールドにアプリケーションを実行させる初期ディレクトリを入力します。[OK]ボタンをクリックするとアプリケーションを定義します。アプリケーションを変更するには[変更]ボタンをクリックしてください。[アプリケーションの変更]ダイアログボックスを表示します。このダイアログボックスは、[名前]フィールドが変更できないことを除いて[アプリケーションの追加]ダイアログボックスと同じです。設定を変更してから[OK]ボタンをクリックします。

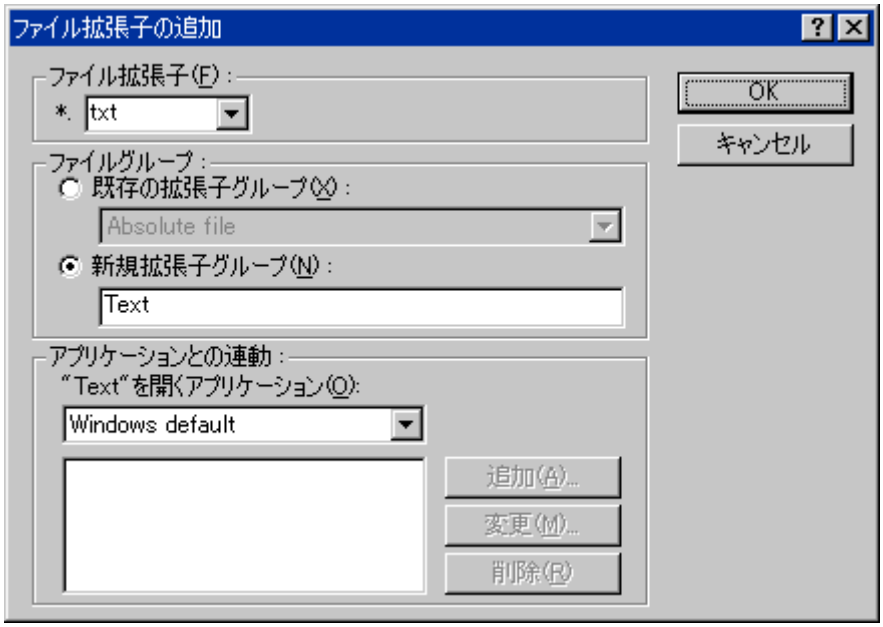
5. [OK]ボタンをクリックすると選択したファイルグループに対するアプリケーションを関連付けます。

2.5.2 新規ファイルグループに新規ファイル拡張子を登録する

新しい拡張子を作成するときは、その拡張子がすでに定義されたファイルグループに属するのか、新しいファイルグループを作成する必要があるか考慮してください。新しい種類のファイルを追加するときには新しいファイルグループを作成してください。作成方法を以下に説明します。

新規ファイルグループに新規ファイル拡張子を登録するには

1. [プロジェクト–>ファイルの拡張子]メニューオプションを選択してください。[ファイル拡張子]ダイアログボックスを表示します。
2. [追加]ボタンをクリックしてください。[ファイル拡張子の追加]ダイアログボックスを表示します。



3. [ファイル拡張子]フィールドに定義する拡張子を入力してください。拡張子には半角英数字と半角下線のみ使用してください。ドロップダウンリストには、現在のプロジェクトで未定義のすべての拡張子が含まれます。この拡張子のうち 1 つを選択すると、ファイル拡張子フィールドにテキストを自動的に表示します。
4. [新規拡張子グループ]オプションを選んで新しいファイルグループを定義する名前を入力してください。
5. この段階では、関連するアプリケーションの変更が可能です。“<拡張子グループ>を開くアプリケーション”のドロップダウンリストでは次から選択が可能です。
   - [Editor]
   - [None]
   - [Other]
   - [Windows default]

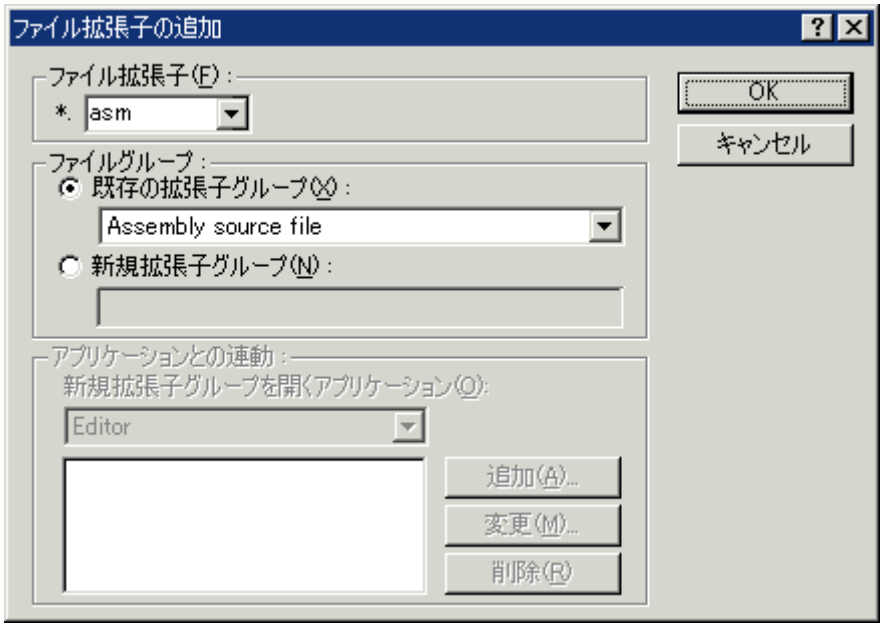
   [Editor]を選択すると、ワークスペースウィンドウ内のファイルを開く機能によって、ファイルを HEW エディタ内で開くことができます。[None]を選択した場合、HEW でファイルを開くことを抑止します。[Other]を選択すると、ファイルを開くための他のツールを指定できます。詳細は、「2.5.1 アプリケーションとファイルグループを関連付けるには」を参照してください。[Windows default]を選択すると、ワークスペースウィンドウ内のファイルを開く機能によって、開いたファイルを Windows®オペレーティングシステムに受け渡します。また、Windows® Explorer で定義されたとして、このファイル拡張子のデフォルト動作を選択します。
6. [OK]ボタンをクリックするとファイル拡張子リストに拡張子を追加します。

2.5.3 新規ファイル拡張子を作成する

HEW に登録されていない拡張子をもつプロジェクトを使っているときは、新しい拡張子を作成し（例えば、フェーズ内のデフォルトの拡張子が.asm でビルダの認識する拡張子が.src のとき）、新しい拡張子を既存のファイルグループに追加することが必要です。追加方法を以下に説明します。

新規ファイル拡張子を既存のファイルグループに追加するには

1. [プロジェクト->ファイルの拡張子]を選択してください。[ファイル拡張子]ダイアログボックスを表示します。
2. [追加]ボタンをクリックすると[ファイル拡張子の追加]ダイアログボックスを表示します。
3. [ファイル拡張子]フィールドに定義する拡張子を入力してください。ドロップダウンリストには、現在のプロジェクトで未定義のすべての拡張子が含まれます。この拡張子のうち 1 つを選択すると、ファイル拡張子フィールドにテキストを自動的に表示します。
4. [既存の拡張子グループ]オプションを選択してこの新しい拡張子をどのファイルグループに追加するか指定してください。
5. [OK]ボタンをクリックするとファイル拡張子リストに拡張子を追加します。



2.6 ファイルのビルド方法を設定する

プロジェクトに必要なファイルを追加したら、次のステップは HEW に各ファイルのビルド方法を指示します。このためには、[ビルド]メニューからメニューオプションを選択します。このメニューの名前は使用するツールによって異なります。

ビルドフェーズにオプションを設定するには

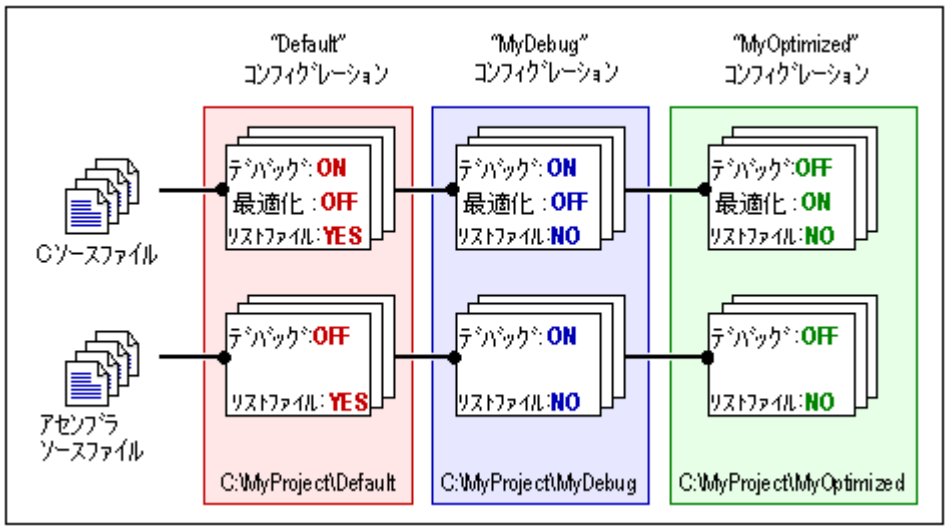
1. [ビルド]メニューからオプションを変更するフェーズのメニューを選択すると、オプションを指定するダイアログボックスを表示します。
2. オプションを設定してください。
3. [OK]ボタンをクリックしてください。

詳細は、コンテキスト依存のヘルプボタンをクリックするか、ヘルプを参照したい場所を選択し **F1** キーを押してください。

2.7 ビルドのコンフィグレーション

HEW では、ビルドのコンフィグレーションの中に、すべてのビルドのオプションを格納できます。つまり、すべてのオプションを格納してそれらに名前を付けることができます。後にそのコンフィグレーションを選択すると、各ビルドフェーズの各オプションを復帰させることができます。また、こうしたビルドのコンフィグレーションにより、ユーザはビルドのコンフィグレーション用にデバッガの設定を行うことができます。つまり、コンフィグレーションごとに異なるエンドプラットフォームをターゲットとして使用できます。

以下の図に"**Default**"、"**MyDebug**"、"**MyOptimized**"の 3 つのビルドコンフィグレーションを示します。"**Default**"ビルド構成では、各フェーズ（コンパイルとアセンブル）が標準設定されています。"**MyDebug**"ビルド構成では、各ファイルがデバッグ情報付きでビルドされています。"**MyOptimized**"ビルド構成では、各ファイルが最大限に最適化されデバッグ情報はありません。このプロジェクトの開発者は、オプションを設定するダイアログボックスに戻ってこれらを設定することなく、これらのビルドコンフィグレーションのうちどれでも選択することができます。



2.7.1 ビルドコンフィグレーションを選択する

ビルドコンフィグレーションを選択するには

1. [ビルド->ビルドの構成]を選択してください。[ビルドコンフィグレーション]ダイアログボックスを表示します。
2. [現在のコンフィグレーション]ドロップダウンリストから使用するビルドコンフィグレーションを選択してください。
3. [OK]ボタンをクリックしてください。

また、[標準]ツールバーのドロップダウンリストから異なるビルドコンフィグレーションを選択することもできます。

2.7.2 新規ビルドコンフィグレーションを追加する

新規ビルドコンフィグレーションを追加するには

1. [ビルド->ビルドの構成]を選択してください。[ビルドコンフィグレーション]ダイアログボックスを表示します。
2. [追加]ボタンをクリックしてください。[コンフィグレーションの追加]ダイアログボックスを表示します。
3. [コンフィグレーション名]フィールドに新しいビルドコンフィグレーション名を入力してください。入力すると、下に表示されるディレクトリがビルドコンフィグレーションに使われるディレクトリに変わります。[ベースコンフィグレーション]フィールドのドロップダウンリストにある既存コンフィグレーションの中から、コンフィグレーションの設定をコピーする元となるコンフィグレーションを選択します。両方のダイアログボックスの[OK]ボタンをクリックすると新しいビルドコンフィグレーションを作成します。

2.7.3 ビルドコンフィグレーションを削除する

ビルドコンフィグレーションを削除するには

1. [ビルド->ビルドの構成]を選択してください。[ビルドコンフィグレーション]ダイアログボックスを表示します。
2. 削除するビルドコンフィグレーションを選択し[削除]ボタンをクリックしてください。
3. [OK]ボタンをクリックすると[ビルドコンフィグレーション]ダイアログボックスを閉じます。

2.8 プロジェクトをビルドする

2.8.1 ファイルをコンパイルする

プロジェクトにある 1 つのファイルをコンパイルできます。

1 つのファイルをコンパイルするには

1. ワークスペースウィンドウの[Projects]タブからビルドするファイルを選択してください。
2. 次のいずれかを行ってください。

- [コンパイル]ツールバーボタン ( ) をクリックする。
- ポップアップメニューから[コンパイル <ファイル名>]を選択する。
- [ビルド->コンパイル]を選択する。
- ”Ctrl+F7”キーを押す。

すべての出力は、アウトプットウィンドウの[Build]タブに表示します。

2.8.2 プロジェクトをビルドする

[ビルド]オプションでは前回のビルド後に変更のあったファイルだけをコンパイルまたはアセンブルします。さらに、前回のビルド以後に変更のあったファイルに依存するソースファイルを再ビルドします。例えば、“TEST.C”にファイル“HEADER.H”が含まれており“HEADER.H”が前回のビルド以後に変更された場合、ファイル“TEST.C”を再コンパイルします。

ビルドするには

次のいずれかを行ってください。

- [ビルド]ツールバーボタン( )をクリックする。
- F7 キーを押す。
- [ビルド->ビルド]を選択する。
- ワークスペースウィンドウの[Projects]タブのプロジェクト上で右クリックし、ポップアップメニューから[ビルド->ビルド]を選択する。

[すべてをビルド]オプションでは変更の有無に関わらず、すべてのソースファイルをコンパイルまたはアセンブルして、新しく作成されたオブジェクトファイルをすべてリンクします。

“すべてをビルド”を実行するには

次のいずれかを行ってください。

- [すべてをビルド]ツールバーボタン( )をクリックする。
- [ビルド->すべてをビルド]を選択する。
- ワークスペースウィンドウの[Projects]タブのプロジェクト上で右クリックし、ポップアップメニューから[ビルド->すべてをビルド]を選択する。

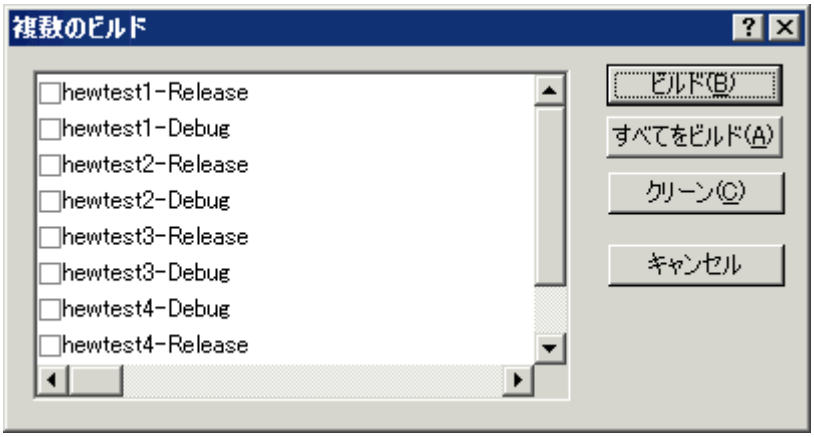
ビルド、あるいは全ビルドからのすべての出力は、アウトプットウィンドウの[Build]タブに表示します。プロジェクトファイルのどれかにエラーが起こると、ビルド処理も全ビルド処理も終了します。

2.8.3 複数のプロジェクトをビルドする

複数のプロジェクトやコンフィグレーションのビルド処理が行えます。

複数のプロジェクトをビルドするには

1. [ビルド->複数ビルド]を選択し、［複数のビルド］ダイアログボックスを表示します。
2. ［複数のビルド］ダイアログボックスでは、どのプロジェクトまたはコンフィグレーションを処理するかを選択できます。どのプロジェクトまたはコンフィグレーションを実行するか選択するには、実行したいプロジェクトとコンフィグレーションの組み合わせの横にあるチェックボックスをオンにします。例えば、プロジェクト“hewtest2”全体をビルド実行したい場合、“hewtest2-Debug”、“hewtest2-Release”を選択してオンにし、その他のチェックボックスをオフにしてください。



3. 項目を選択したら、[ビルド]ボタンをクリックしてください。選択したプロジェクトやコンフィグレーションのビルドを開始します。
4. また、選択した項目に対して、全ビルド実行したい場合、[すべてをビルド]ボタンをクリックしてください。
5. また、選択した項目の中間ファイルおよび出力ファイルを削除したい場合、[クリーン]ボタンをクリックしてください。このときビルドはされません。
6. 通常のビルドと同じように、ビルドの結果をアウトプットウィンドウの[Build]タブに表示します。

2.8.4 ツールの実行を中止する

ビルド中のツールの実行を中止することができます。

ツールの実行を中止するには

1. 次のいずれかを行ってください。

- [ツールの実行の中止]ツールバーボタン ( )をクリックする。
- [ビルド->ツールの中止]を選択する。

2. ツールの実行が直ちに中止されます。
3. ツールの実行が中止されると、アウトプットウィンドウの[Build]タブに”Build Stopped by User”というメッセージが表示されます。

注：

ビルド実行を中止したときに実行されていたツールによっては、無効なファイルを出力ファイルとして残していることがあります。ビルドを再実行する場合は、中止したツールの出力ファイルをすべて削除してください。

2.8.5 ビルドの中間ファイルおよび出力ファイルを削除する

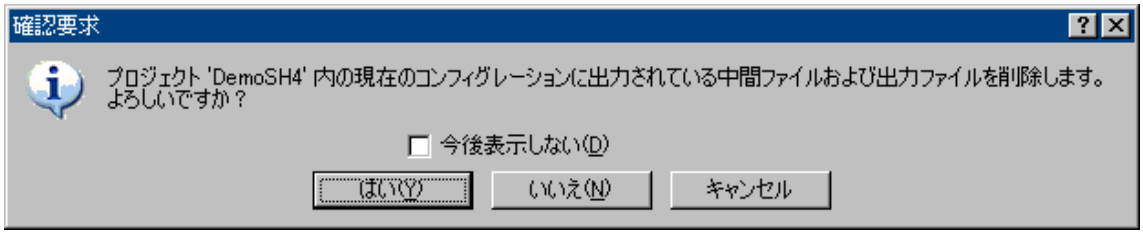
ワークスペースに含まれるプロジェクトのコンフィグレーションにあるビルドの中間ファイルおよび出力ファイルを削除できます。

現在のコンフィグレーションにあるビルドの中間ファイルおよび出力ファイルを削除するには

次のいずれかを行ってください。

- [ビルド->クリーン アクティブプロジェクト]を選択する。
- ワークスペースウィンドウの[Projects]タブのプロジェクト上で右クリックする。表示されるポップアップメニューから[ビルド->クリーン アクティブプロジェクト]を選択する。

すべてのビルドの中間ファイルおよび出力ファイルを削除するかを確認するダイアログボックスが表示されます。削除するには「はい」を選択してください。削除しない場合は「いいえ」または「キャンセル」を選択してください。



確認要求ダイアログボックスを表示しない場合、[今後表示しない]チェックボックスをオンにしてください。

確認要求ダイアログボックスを再び表示するには

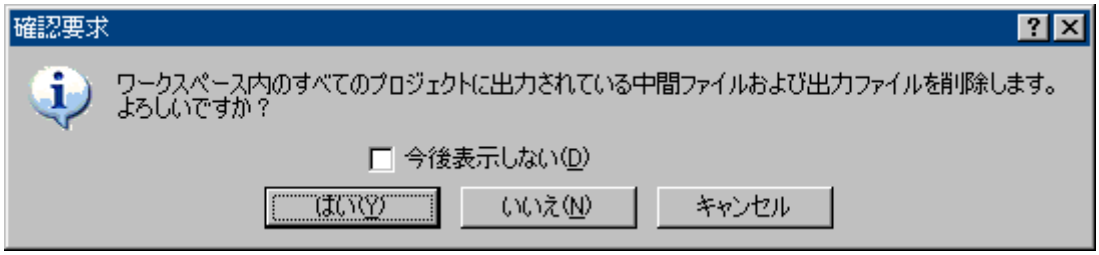
1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [確認]タブを選択してください。
3. [クリーン アクティブプロジェクト]チェックボックスをオンにしてください。このチェックボックスのデフォルトはオンです。
4. OK をクリックしてください。

すべてのプロジェクトのコンフィグレーションにあるビルドの中間ファイルおよび出力ファイルを削除するには

次のいずれかを行ってください。

- [ビルド->クリーン 全プロジェクト]を選択する。
- ワークスペースウィンドウの[Projects]タブのワークスペース上で右クリックする。表示されるポップアップメニューから[クリーン 全プロジェクト]を選択する。

すべてのビルドの中間ファイルおよび出力ファイルを削除するかを確認するダイアログボックスが表示されます。削除するには「はい」を選択してください。削除しない場合は「いいえ」または「キャンセル」を選択してください。



確認要求ダイアログボックスを表示しない場合、[今後表示しない]チェックボックスをオンにしてください。

確認要求ダイアログボックスを再び表示するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [確認]タブを選択してください。
3. [クリーン 全プロジェクト]チェックボックスをオンにしてください。このチェックボックスのデフォルトはオンです。
4. OK をクリックしてください。

中間ファイルおよび出力ファイルを削除したプロジェクト名およびコンフィグレーション名をアウトプットウィンドウの[Build]タブに表示します。

コンフィグレーションを複数選択しビルドの中間ファイルおよび出力ファイルを削除できます。詳細は、「2.8.3 複数のプロジェクトをビルドする」を参照してください。

注：

リンケージリストファイル（".MAP"）、アセンブリプログラム用リストファイル（".LIS"）などは削除されません。リロケータブルファイルおよびアブソリュートファイルを削除します。アブソリュートファイルの拡張子は指定したフォーマットにより異なります。

2.8.6 アウトプットウィンドウの Build タブの構成

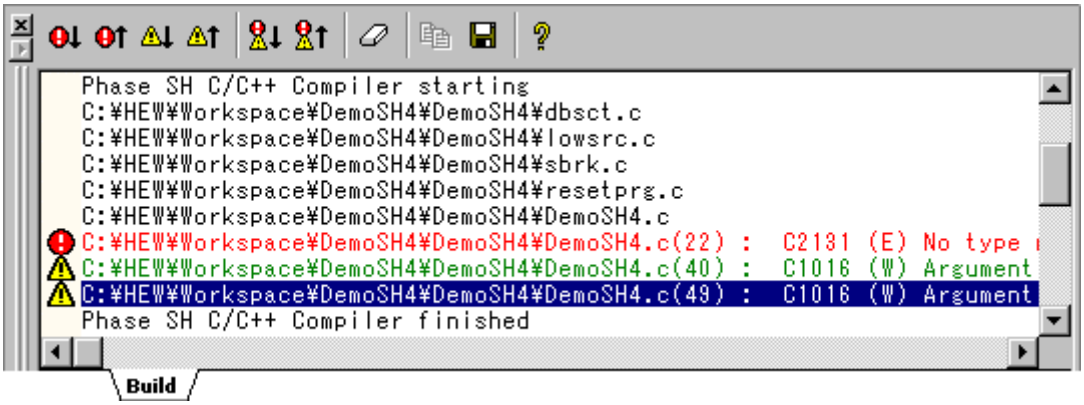
ビルド実行（コンパイラ、アセンブラなど）の出力結果を表示します。ソースファイルにエラーがある場合、アウトプットウィンドウの[Build]タブにアイコン、およびエラーメッセージをソースファイル名、行番号とともに出力します。

エラーメッセージの表示に関するツールバーボタンまたはポップアップメニュー項目をクリックすると、エラーメッセージ出力行を強調表示し、該当ソースがある場合はエディタに該当ソース行を表示します。

エラーメッセージ出力行をダブルクリックすると、該当ソースがある場合はエディタにエラーメッセージの該当ソース行を表示します。

エラーメッセージの表示に関する操作を行った場合は、ステータスバーにエラーメッセージを表示します。

エラーメッセージ行のテキストを他の行のテキストと異なる色で強調表示されるようにカスタマイズが可能です。下の図はエラー行のテキストに赤色、ウォーニング行のテキストに緑色を設定した例です。



ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	機能
-		次のエラーの出力行を強調表示し、該当ソースがある場合はエディタに該当ソース行を表示します。
-		前のエラーの出力行を強調表示し、該当ソースがある場合はエディタに該当ソース行を表示します。
-		次のウォーニングの出力行を強調表示し、該当ソースがある場合はエディタに該当ソース行を表示します。
-		前のウォーニングの出力行を強調表示し、該当ソースがある場合はエディタに該当ソース行を表示します。
次エラー/ウォーニング/情報の表示		次のエラー、ウォーニング、またはインフォメーションの出力行を強調表示し、該当ソース行がある場合はエディタに該当ソース行を表示します。
前エラー/ウォーニング/情報の表示		前のエラー、ウォーニング、またはインフォメーションの出力行を強調表示し、該当ソースがある場合はエディタに該当ソース行を表示します。
ヘルプ		この出力行に関するヘルプを表示します。
ジャンプ	-	この出力行に関するソース行をエディタに表示します。
クリア		ウィンドウ内容をクリアします。
保存		ウィンドウ内容をテキストファイルに保存します。
コピー		ウィンドウ内容の選択部分を Windows(R)クリップボードにコピーします。
ツールバー表示	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ	-	ツールバーボタンをカスタマイズします。

[Build]タブではビルド実行により出力されたエラーメッセージ行の左側の余白（カラム）にエラーメッセージのエラーレベルに対応したアイコンを表示します。

アイコン名	アイコン	エラーメッセージのエラーレベル
Build Error		エラーであることを示します。
Build Warning		ウォーニングであることを示します。
Information		インフォメーションであることを示します。

アウトプットウィンドウの[Build]タブでは、エラーメッセージ行のテキストを他の行のテキストと異なる色で強調表示されるようにカスタマイズが可能です。

既存の色をカスタマイズするには

1. [基本設定->表示の形式]を選択し、[表示形式]ダイアログボックスを表示します。
2. [Output]アイテムをダイアログボックスの左側のツリーから選択し展開します。
3. 色を変更したいカテゴリを選択します。次の表に示すカテゴリが選択できます。

カテゴリ	カラータブの前面の色（デフォルト）	カラータブの背景の色（デフォルト）	カスタマイズされるタブ	カスタマイズされる出力内容
Text	SYSTEM	SYSTEM	すべてのタブ	すべて
Build Error Text	黒	白	Build	エラー行
Build Warning Text	黒	白	Build	ウォーニング行
Information Text	黒	白	Build	インフォメーション行

4. [カラー]タブの[前面]リストと[背景]リストの選択を変更します。
5. [OK]ボタンをクリックします。

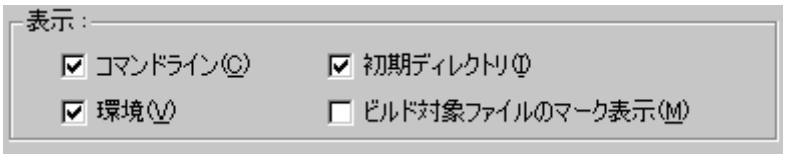
2.8.7 アウトプットウィンドウの Build タブの内容の制御

ビルド中にビルド実行情報（ファイルに適用したコマンドラインオプションなど）を表示すると便利です。HEW では、"ビルド"、"すべてをビルド"、"コンパイル"処理中、アウトプットウィンドウの[Build]タブにそのオプションを表示するかどうかを[オプション]ダイアログボックスで指定できます。

ビルド中にビルド情報の表示の有無を指定するには

1. [基本設定->オプション]を選択すると[オプション]ダイアログボックスを表示します。
2. [ビルド]タブを選択してください。
3. [表示]グループのチェックボックスを以下のように設定します。

- [コマンドライン]にはツール実行時のコマンドライン表示の有無を指定します。
- [環境]にはツール実行時の環境変数の表示の有無を指定します。
- [初期ディレクトリ]にはツールが起動されるディレクトリパスの表示の有無を指定します。

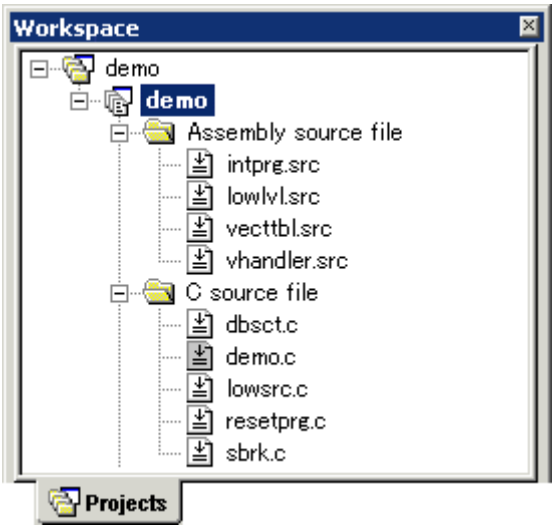


2.8.8 ビルド対象ファイルをワークスペースウィンドウにマーク表示する

前回のビルドで生成されたファイルより後に更新されたファイル（ビルド対象ファイル）をワークスペースウィンドウ上にマーク表示します。下記ウィンドウでは、ビルド対象ファイルは"demo.c"です。

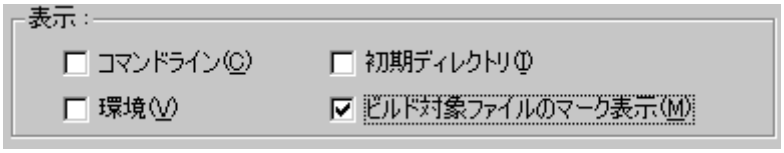
次に[ビルド]をクリックしたとき、このファイルをビルドします。アクティブプロジェクトの依存プロジェクトのファイルにもマーク表示します。

これらファイルのマーク表示は、ビルドに影響することが起こるたびに更新します。例えば、オプションの変更、ファイルの追加、依存関係の変更、ファイルの修正などです。



ビルド対象ファイルをワークスペースウィンドウにマーク表示するには

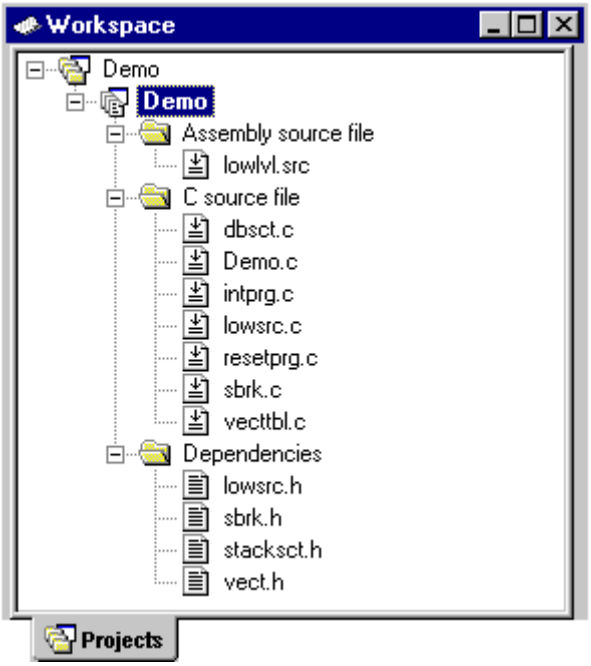
1. [基本設定->オプション]を選択し[オプション]ダイアログボックスを表示します。
2. [ビルド]タブを選択してください。
3. [ビルド対象ファイルのマーク表示]チェックボックスをオンにしてください。
4. [OK]ボタンをクリックしてください。



2.9 ファイル依存関係

多くの場合プロジェクトにはファイル間の依存関係があります。例えば、1 つの C ファイルはいくつかのファイルをインクルードします。複雑なプロジェクトでは、ソースファイルが他のインクルードファイルに依存するため、管理が難しくなります。しかし、HEW にはファイル依存関係をスキャンする機能があり、そのプロジェクトにあるすべてのファイルの依存関係をチェックできます。スキャンが完了すると、プロジェクトのファイル依存関係を示す最新のリストをワークスペースウィンドウの[Projects]タブに表示します。

最初に、すべての依存ファイルは[Dependencies]フォルダに表示されています。



プロジェクトのファイル依存関係は、以下の操作タイミングで自動更新されます。

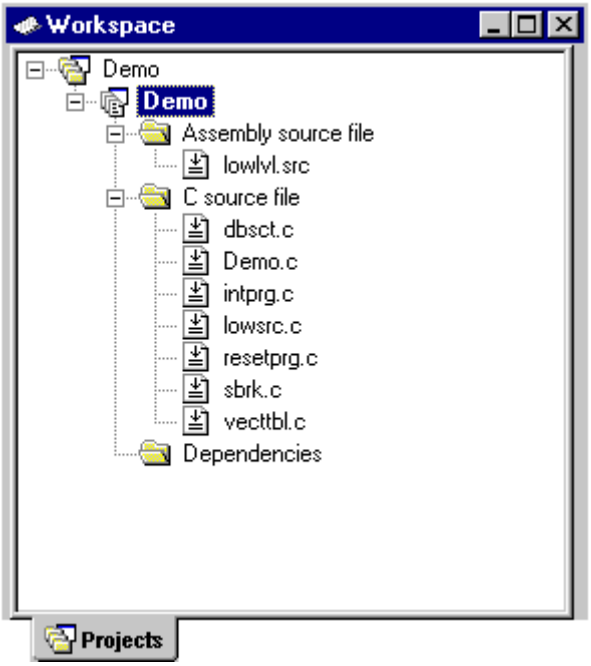
操作タイミング	主な操作
ワークスペースを開く	[ようこそ!]ダイアログボックスのオプションの選択
	[ファイル->ワークスペースを開く]
	[ファイル->新規ワークスペースの作成]
ワークスペースにプロジェクトを挿入する	[プロジェクト->プロジェクトの挿入]
アクティブプロジェクトに設定する	[プロジェクト->アクティブプロジェクトに設定]
ファイルを追加する	[プロジェクト->ファイルの追加]
ファイルを保存する	[ファイル->上書き保存]
ビルドする	[ビルド->ビルド]
依存関係を自動更新する	[依存関係の自動更新をしない]チェックボックスをオフにする

プロジェクトのファイル依存関係を自動更新しないようにするには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [ビルド]タブを選択します。
3. [依存関係の自動更新をしない]チェックボックスをオンにしてください。このチェックボックスのデフォルトはオフです。
4. OK をクリックしてください。

注：

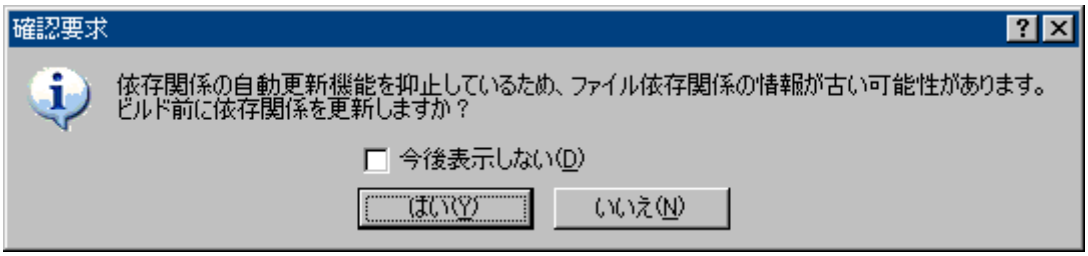
この設定を行うと、プロジェクトのファイル依存関係はすべてクリアされます。プロジェクトのファイル依存関係を手動で更新するには、「プロジェクトのファイル依存関係を手動で更新するには」を参照してください。



また、[ビルド->ビルド]を選択した場合、ビルド前に依存関係を更新するかを確認するダイアログボックスが表示されます。

依存関係の自動更新機能を抑止しているため、ファイル依存関係の情報が古い可能性があります。

ビルド前に依存関係を更新する場合は、「はい」を選択してください。「いいえ」を選択した場合、現在の依存関係の情報でビルドを実行します。



確認要求ダイアログボックスを表示しない場合、[今後表示しない]チェックボックスをオンにしてください。

確認要求ダイアログボックスを再び表示するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [確認]タブを選択してください。
3. [ビルド前に依存関係を更新]チェックボックスをオンにしてください。このチェックボックスのデフォルトはオンです。
4. OK をクリックしてください。

プロジェクトのファイル依存関係を手動で更新するには

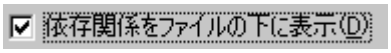
次のいずれかを行ってください。

- [ビルド->すべての依存関係を更新]を選択する。
- ワークスペースウィンドウの[Projects]タブのプロジェクト上で右クリックし、ポップアップメニューから[ビルド->すべての依存関係を更新]を選択する。

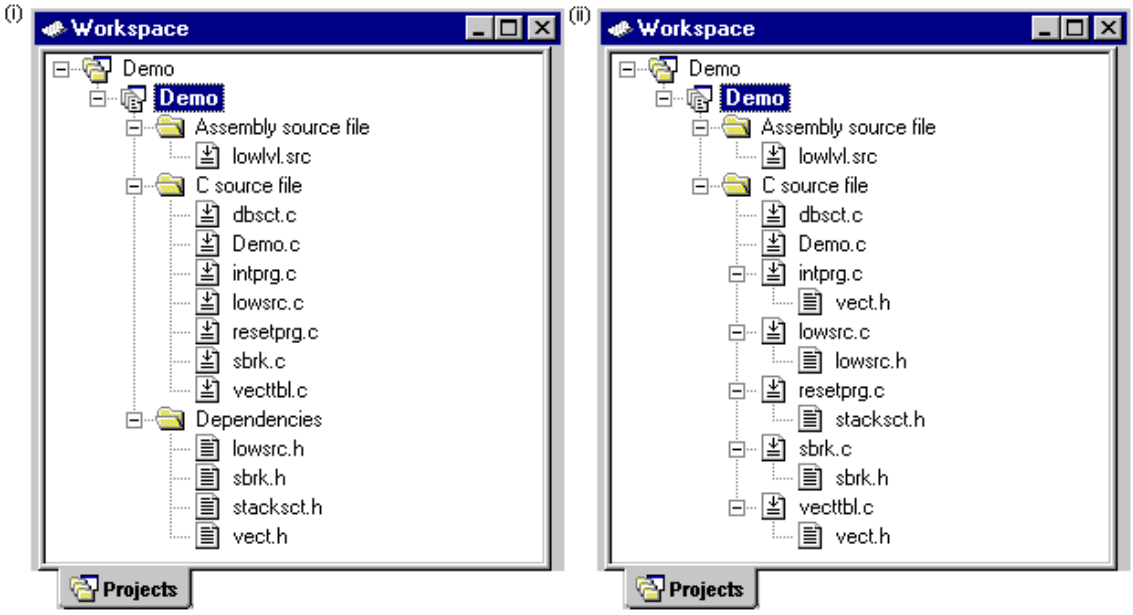
2.10 表示方法を指定する

ワークスペースウィンドウの[Projects]タブの中で右クリックすると、ポップアップメニューを表示します。その中から[表示の構成]メニューオプションを選択し、以下の情報の表示方法を設定してください。以下に[表示の構成]ダイアログボックスの各オプションについて説明します。

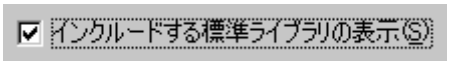
各ファイルの下に依存ファイルを表示する



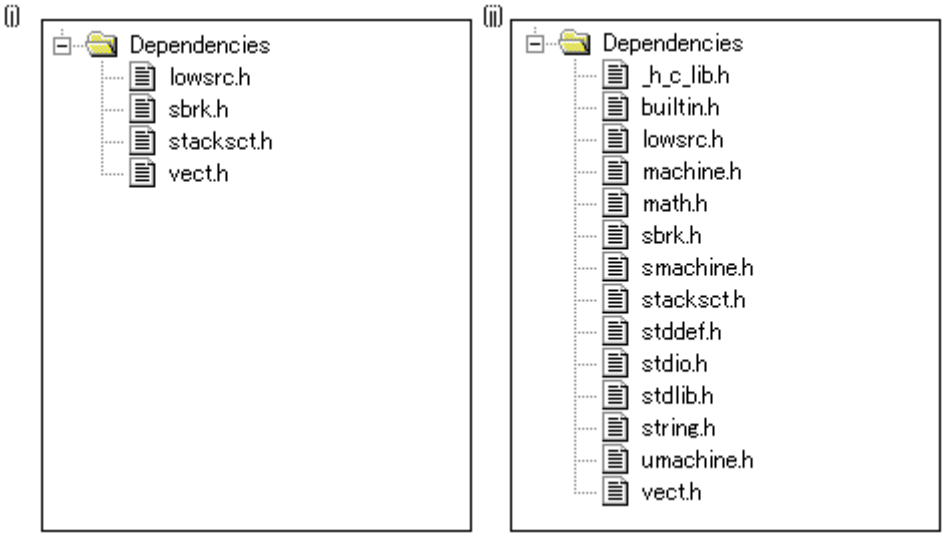
[依存関係をファイルの下に表示]チェックボックスをオンにすると、依存ファイルがそれをインクルードするソースファイルの下に平坦な構造で表示され、ファイル自体がフォルダになります（図(ii)を参照）。このオプションを選択しないと[Dependencies]フォルダにすべてのファイル依存を示します（図(i)を参照）。



標準ライブラリファイルのインクルードを表示する

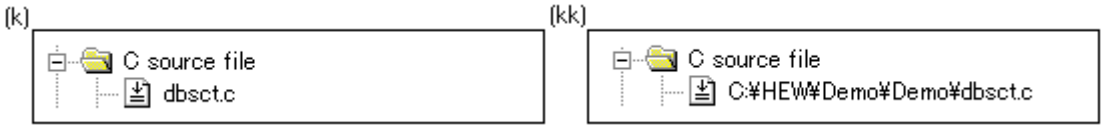


デフォルトでは、標準インクルードパスの依存ファイルは表示しません（図(j)を参照）。例えば、C コードで#include <stdio.h>などのインクルード文を書くと、stdio.h は依存ファイルとして表示しません。そのようなシステムインクルードファイルを表示するには、[インクルードする標準ライブラリの表示]チェックボックスをオンにしてください（図(jj)を参照）。



ファイルのパスを表示する

デフォルトでは、ファイルのパスは表示されません（図(k)を参照）。[ファイルパスの表示]チェックボックスをオンにすると、[Projects]タブのすべてのファイルをフルパス（ドライブ名からのパス）で表示します（図(kk)を参照）。



ファイルをタイムスタンプの順序でソートする

[ファイルのタイムスタンプによるソート]チェックボックスをオンにすると、ワークスペース内のファイルをタイムスタンプの順序で並べることができます。最新のファイルがリストの先頭、最も古いファイルがリストの最後になります。

このオプションを設定した後でファイルを更新した場合は、手動で表示順を更新しなければいけません。

手動で表示順を更新するには

ワークスペースウィンドウの[Projects]タブのポップアップメニューから[リフレッシュ]を選択してください。

2.11 ワークスペースにプロジェクトを追加する

ワークスペースを作成したとき、最初はプロジェクトが 1 つしかありません。しかし、後で、新しいプロジェクトや既存のプロジェクトを追加できます。

ワークスペースに新しいプロジェクトを追加するには

1. [プロジェクト->プロジェクトの挿入]を選択してください。[プロジェクトの挿入]ダイアログボックスを表示します。
2. [新規プロジェクト]ラジオボタンを選択してください。
3. [OK]ボタンをクリックしてください。[新規プロジェクトの挿入]ダイアログボックスを表示します。
4. [プロジェクト名]フィールドにプロジェクト名を入力してください。プロジェクト名は 32 文字以内で、半角英数字、半角下線のみ使用してください。特に、日本語文字、空白、または半角マイナス記号を使用しないでください。プロジェクト名を入力すると、HEW は自動的にサブディレクトリを追加します。これは不要であれば削除できます。
5. [参照]ボタンをクリックしてプロジェクトを作成するディレクトリを選択してください。または、[ディレクトリ]フィールドにディレクトリを入力できます。
6. プロジェクトタイプリストには使用可能なプロジェクトの種類を示します（Application、Library など）。このリストから作成するプロジェクトの種類を選択してください。
7. [OK]ボタンをクリックすると、プロジェクトを作成しワークスペースに追加します。

既存のプロジェクトをワークスペースに追加するには

1. [プロジェクト->プロジェクトの挿入]を選択してください。[プロジェクトの挿入]ダイアログボックスを表示します。
2. [既存プロジェクト]ラジオボタンを選択してください。
3. プロジェクトデータベースファイル（.HWP ファイル）へのフルパスを入力するか、[参照]ボタンをクリックしてプロジェクトデータベースファイルを指定してください。
4. [OK]ボタンをクリックするとそのプロジェクトをワークスペースに追加します。

2.12 アクティブプロジェクトを設定する

プロジェクトには 3 つの状態があります。アクティブプロジェクト、ロード（読み込まれた）プロジェクト、およびアンロードプロジェクトです。

ワークスペースには複数のプロジェクトを含めることができますが、アクティブなプロジェクトは 1 つだけです。このアクティブプロジェクトでビルド動作とデバッグ動作を実行します。そして、そのプロジェクト用のビルダやデバッガのオプションを変更できます。またアクティブプロジェクトは太字で表示します。

プロジェクトをアクティブにするには

以下の操作のいずれかを選択します。

- [プロジェクト->アクティブプロジェクトに設定]サブメニューからアクティブにしたいプロジェクトを選択してください。
- ワークスペースウィンドウの[Projects]タブからアクティブでないプロジェクトを選択してください。右クリックしてポップアップメニューを表示し、[アクティブプロジェクトに設定]オプションを選択してください。

プロジェクトが読み込まれていると、プロジェクトのディレクトリをオープンし、ファイルを見ることができます。またプロジェクトのビルダおよびデバッグオプションを変更することもできます。読み込みプロジェクトに対して[ツール]メニューのツールを実行することもできます。

プロジェクトをアンロードにするには

1. ワークスペースウィンドウの[Projects]タブからアクティブでないプロジェクトを選択してください。
2. 右クリックしてポップアップメニューを表示し、[プロジェクトのアンロード]を選択してください。一度にいくつかのプロジェクトを選択することもできます。

アンロードしたプロジェクトは、ワークスペースウィンドウの[Projects]タブのアイコンがグレー表示です。

アンロードしたプロジェクトがある場合は以下の操作によりプロジェクトを読み込むことができます。

ワークスペースのすべてのプロジェクトを読み込むには

1. ワークスペースウィンドウの[Projects]タブのワークスペースを選択してください。
2. 右クリックしポップアップメニューを表示し、[全プロジェクトの読み込み]を選択してください。

ワークスペースのプロジェクトを読み込むには

1. ワークスペースウィンドウの[Projects]タブからアンロードしたプロジェクトを選択してください。
2. 右クリックしてポップアップメニューを表示し、[プロジェクトの読み込み]を選択してください。一度にいくつかのプロジェクトを選択することもできます。

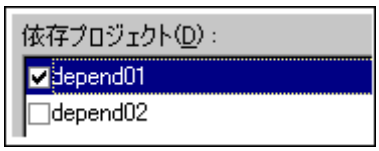
2.13 プロジェクト間の依存関係を指定する

ワークスペースのプロジェクトは、他のプロジェクトと依存関係を持たせることができます。ビルド処理をすると、依存プロジェクトを最初にビルドします。これは、ワークスペースのプロジェクトを他のプロジェクトが使用しているときなどに使用します。例えば、ワークスペースに 2 つのプロジェクトがあるとします。1 つはアプリケーションプロジェクトに含まれたライブラリだとします。この場合 2 番目のアプリケーションのビルド前にライブラリを正確にビルドし、また最新でなくてはなりません。そのため、ライブラリをアプリケーションプロジェクトの依存プロジェクトに指定します。こうすると、最新でないライブラリを先にビルドします。

依存プロジェクトをビルドするとき、HEW は依存プロジェクトがアクティブプロジェクトのビルドコンフィグレーションになるようにします。上記の例では、アクティブプロジェクトのビルドコンフィグレーションが“Debug”であるとき、HEW は、依存プロジェクトで“Debug”ビルドコンフィグレーションが選択されるようにします。このような一致したコンフィグレーションが存在しない場合、HEW は依存プロジェクトで最近使われたコンフィグレーションを使用します。

依存プロジェクトを指定するには

1. [プロジェクト->依存関係]を選択してください。[依存プロジェクト]ダイアログボックスを表示します。
2. 依存関係を指定する[プロジェクト]を選択してください。[依存プロジェクト]リストに（選択したプロジェクト以外の）ワークスペース内のすべてのプロジェクトを表示します。
3. [依存プロジェクト]リストには各プロジェクトにチェックボックスがあります。選択した[プロジェクト]が依存するプロジェクトのチェックボックスをオンにしてください。
4. [OK]ボタンをクリックしてください。



2.14　ワークスペースからプロジェクトを削除する

ワークスペースからプロジェクトを削除するには

1. ワークスペースウィンドウの[Projects]タブで削除したいプロジェクトを選択してください。
2. 右クリックしポップアップメニューを表示してください。
3. [プロジェクトの削除]メニューオプションを選択してください。

注：

ワークスペースからアクティブプロジェクトを削除することはできません。

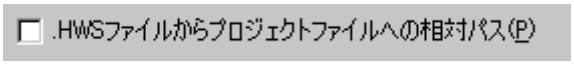
2.15　ワークスペースのプロジェクト相対パス

HEW では、プロジェクトをワークスペースに追加する場合、相対パスを使用してワークスペースに追加することが選択できます。これにより、プロジェクトをワークスペースディレクトリの上に置くことができ、HEW ワークスペースの再配置も正確にできます。プロジェクトは常にワークスペースと相対的であるため、プロジェクトがワークスペースの上のディレクトリにある場合、再配置のあと、HEW は同じ相対場所でプロジェクトを見つけようとします。このことは、複数のワークスペース間で共有したプロジェクトを使用するとき、特に便利です。

HEW の古いバージョンでは、このプロジェクトは再配置されておらず、オリジナルのプロジェクトパスをアクセスしようとしました。また、ワークスペースディレクトリのサブディレクトリにあったプロジェクトを再配置することのみ可能でした。これは現在のバージョンでも、HEW の標準的な動作です。

プロジェクト相対パスを設定するには

1. ワークスペースウィンドウでプロジェクトを選択してください。
2. 右クリックしてポップアップメニューを表示してください。
3. [プロパティ]を選択してください。
4. [.HWS ファイルからプロジェクトファイルへの相対パス]チェックボックスをオンにしてください。
5. [OK]ボタンをクリックしてください。



3. ビルドの応用

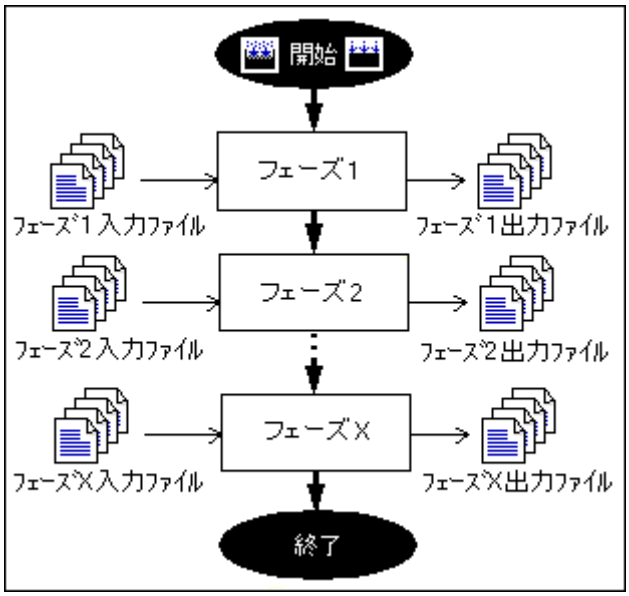
この章ではより高度なビルドの概念を説明します。

3.1 ビルド実行の復習

2 章「ビルドの基本」では、ビルド実行をコンパイラ、アセンブラ、リンケージエディタを用いて説明しました。HEW ではこれが一般的な実行環境です。しかし、ビルド実行を変更（フェーズの追加や削除など）するためには、ビルドの機能についてさらに知ることが必要です。

3.1.1 ビルドとは?

プロジェクトのビルドとは、複数の特定の入力ファイルに複数のツールを適用して期待する出力を得ることです。つまり、オブジェクトファイルを得るために、C/C++ソースファイルにコンパイラを適用したりアセンブリ言語のソースファイルにアセンブラを適用したりします。ビルドの各ステップ、または、各「フェーズ」において、様々な入力ファイルの集まりに各種ツールを適用します。以下の図にビルド処理を示します。



HEW ではビルド処理を変更できます。[ビルド->ビルドフェーズ]を選択し[ビルドフェーズ]ダイアログボックスを表示します。左には現在のプロジェクトで定義されたフェーズを示します。

この章では[ビルドフェーズ]ダイアログボックスが提供する様々な機能について説明します。

3.2 カスタムビルドフェーズを作成する

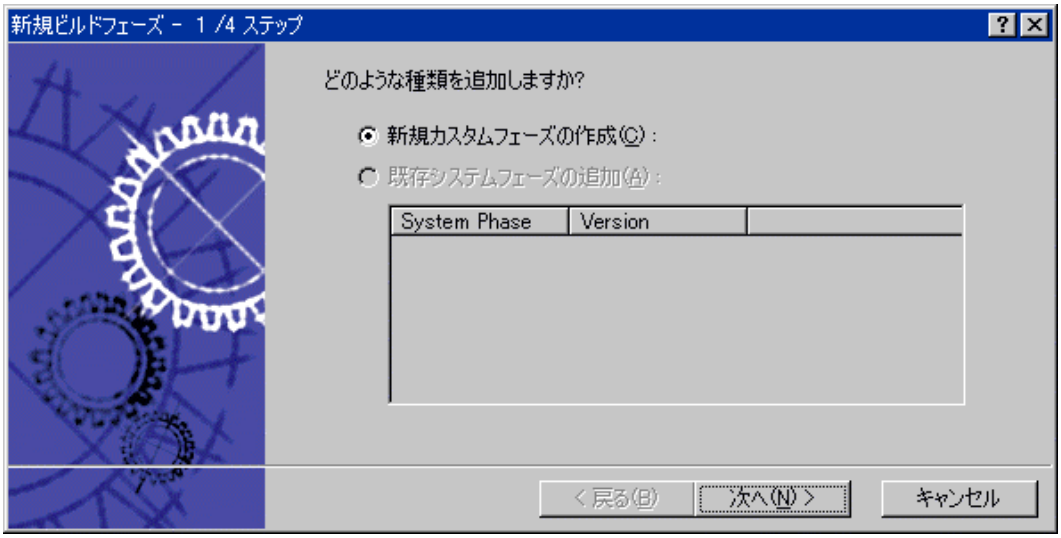
標準のビルドの前後または途中に他のツールを実行する場合、独自のビルドフェーズ（カスタムビルドフェーズ）を作成します。カスタムビルドフェーズを略してカスタムフェーズといいます。

新しいカスタムフェーズを作成するには

1. [ビルド->ビルドフェーズ]を選択すると[ビルドフェーズ]ダイアログボックスを表示します。
2. [追加]ボタンをクリックしてください。新しいビルドフェーズを作成するための[新規ビルドフェーズ]ダイアログボックスを表示します。
3. 以下の 4 つのステップにしたがってください。ステップを進めるには[次へ>]ボタンを、戻るには[<戻る]ボタンをクリックしてください。

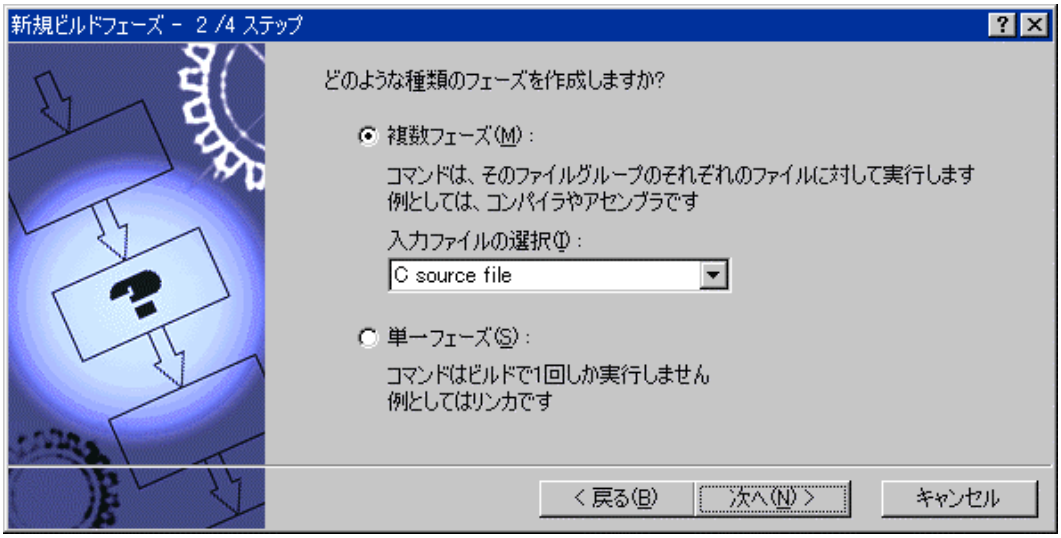
   ステップ 1

   [新規カスタムフェーズの作成]または[既存システムフェーズの追加]を選択します。システムビルドフェーズ（略してシステムフェーズといいます）は、使用しているツールチェイン（コンパイラ、アセンブラ、リンケージエディタ、ライブラリアンなど）内で定義済みのすぐに使用できるフェーズ、または、ユーティリティフェーズ（例えば、ファイルコピー、ソースコード複雑度解析ツールなど）です。システムフェーズがない場合、[既存システムフェーズの追加]ボタンが非アクティブになります。[新規カスタムフェーズの作成]を選んでカスタムフェーズを作成してください。

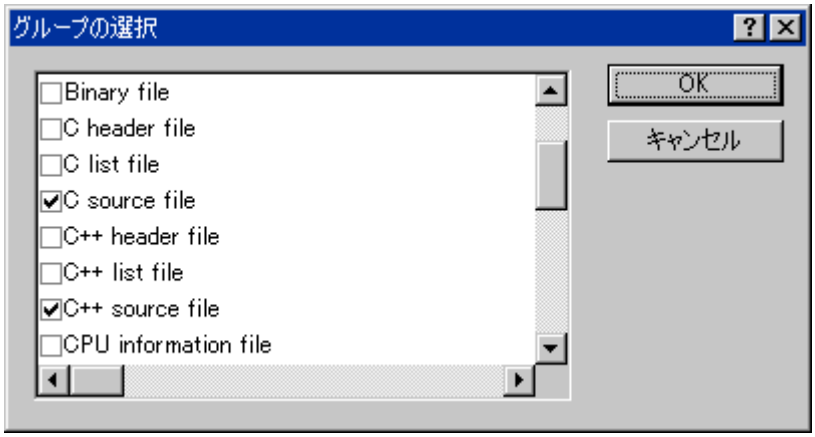


ステップ 2

作成するフェーズの種類を選択します。2 つの選択肢（[複数フェーズ]または[単一フェーズ]）があります。複数フェーズを実行すると特定のファイルグループに属するプロジェクト内の各ファイルにコマンドを適用します。[入力ファイルの選択]フィールドに[C source file]を選択すると、プロジェクト内のファイル毎にコンパイラを実行します。単一フェーズを選択するとビルド実行中に一度だけ実行します。



[入力ファイルの選択]ドロップダウンリストは、そのプロジェクト用に定義された現在のファイルグループを含みます。[入力ファイルの選択]ドロップダウンリストの中の[Multiple Groups]エントリを選択すると、複数の入力ファイルグループを定義できる[グループの選択]ダイアログボックスを表示します。



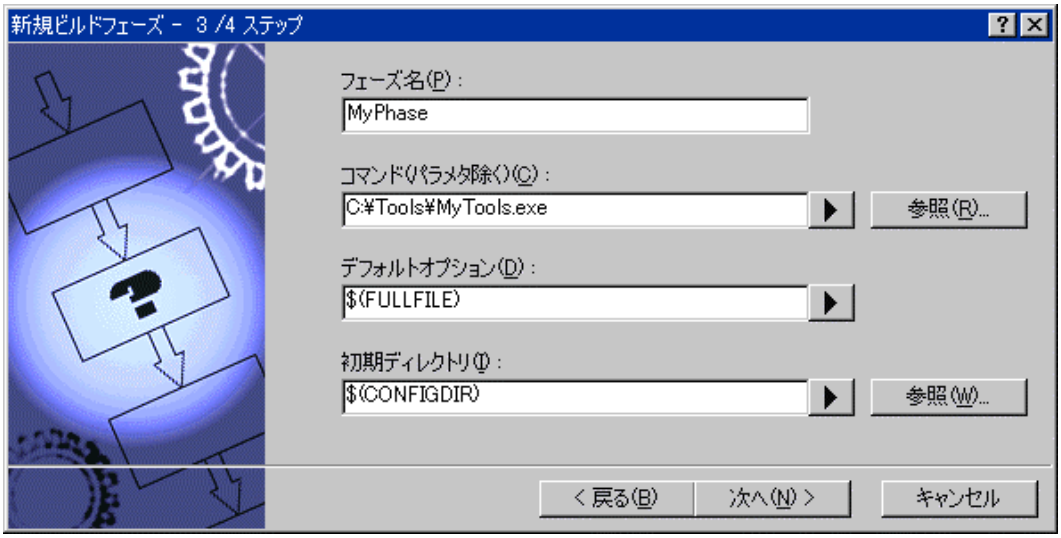
複数選択すると、[入力ファイルの選択]ドロップダウンリストは[Multiple Groups]と表示します。プロジェクトに追加されているカスタムフェーズ用に、複数の入力ファイルグループをこのダイアログボックスでを選択することができます。ファイルグループを選択するには、ファイルグループ名の隣にあるチェックボックスをオンにしてください。このダイアログボックスでは、1 つ以上のファイルグループが選択できます。

ステップ 3

フェーズのコマンドについての情報を入力します。[フェーズ名]フィールドにフェーズ名を入力します。[コマンド]フィールドにプログラムファイルのパスを入力します（コマンドラインオプションは含めません。オプションは HEW のメニューバーの[ビルド]メニューで指定します）。[デフォルトオプション]フィールドにフェーズのデフォルトのオプションを指定します。デフォルトオプションはプロジェクトに新しいファイルを追加するときに付加されるオプションです。[初期ディレクトリ] フィールドにはそのプログラムをどのディレクトリから実行するか（つまり、ツール実行前にどこにカレントディレクトリを設定するか）を入力します。

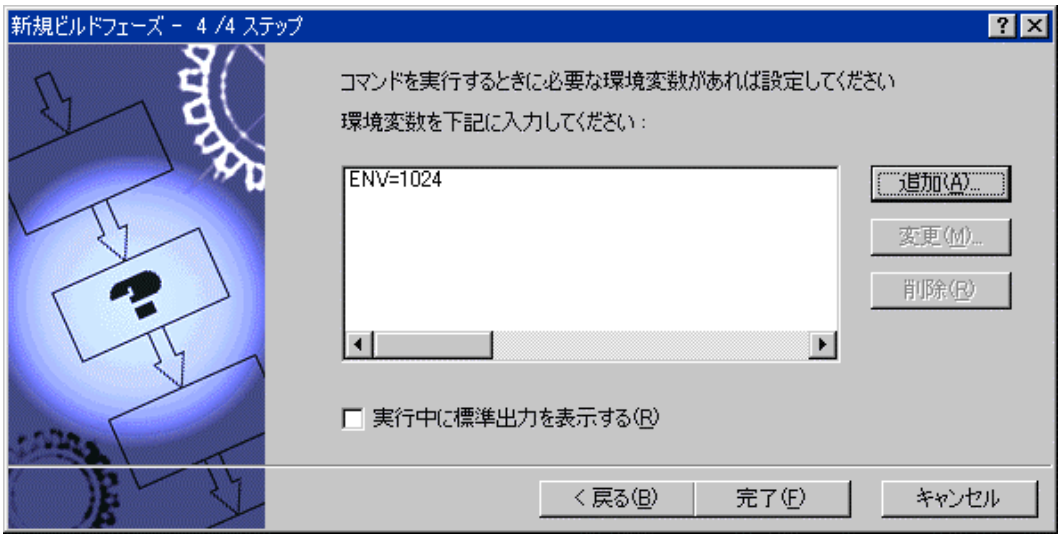
注：

[フェーズ名]には日本語文字（全角英数字も含む）を使用しないでください。また、HEW1.x で作成したプロジェクトを HEW2.X 以上で使用する場合、日本語名のカスタムフェーズについては、半角英数字等（半角下線、半角マイナス、空白は使用可能）の名称で設定し直してください。

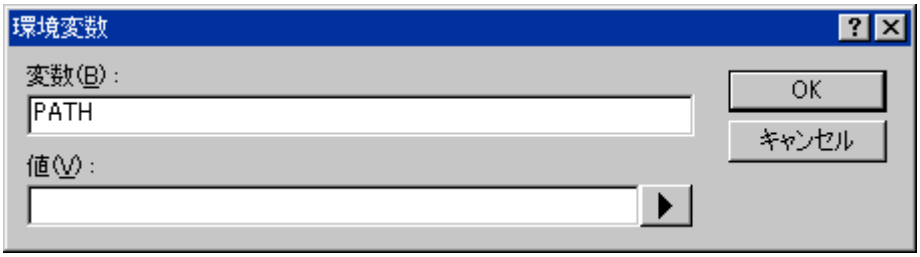


ステップ 4

フェーズに必要な環境変数を指定します。



新しい環境変数を追加するには、[追加]ボタンをクリックしてください。[環境変数]ダイアログボックスを表示します。新しい環境変数を追加するには[変数]フィールドに環境変数名を入力して[値]フィールドに環境変数の値を入力して[OK]ボタンをクリックします。環境変数を変更するには、ステップ 4 のダイアログボックスのリストから環境変数を選んで、[変更]ボタンをクリックします。[変数]フィールドと[値]フィールドを変更して[OK]ボタンをクリックすると、リストに変更した変数を追加します。環境変数を削除するには、ステップ 4 のダイアログボックスのリストから削除する環境変数を選んで、[削除]ボタンをクリックします。



追加するツールが動作中に出力を表示できる場合、ステップ 4 のダイアログボックスの[実行中に標準出力を表示する]オプションを使用してください。出力が発生するごとにツールの出力を表示します。このオプションが off に設定されると、HEW はツールに表示されている全出力を保存し、ツールが動作を終了したときアウトプットウィンドウに表示します。

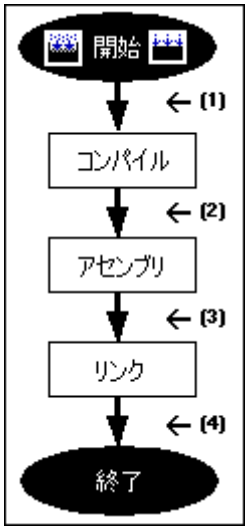
注：

[実行中に標準出力を表示する]を用いると、特定のオペレーティングシステムで特定のツールを使用するとき問題を引き起こすことがあります。もしツールが HEW の中でロックアップ、またはフリーズするといった問題がありましたら、[実行中に標準出力を表示する]オプションのチェックを外してください。

[完了]ボタンをクリックすると、指定した内容で新しいフェーズを作成します。デフォルトでは[ビルドフェーズ]ダイアログボックスの[ビルド順序]タブの[ビルドフェーズの順序]リストの最後に新しいフェーズを追加します。

3.3 ビルドフェーズの順序

標準的ビルド処理（下の図）では、(1)コンパイラの前、(2)アセンブラの前、(3)リンケージエディタの前、(4)リンケージエディタの後、の 4 ヶ所にフェーズを追加できます。ビルドの順序の中で好きな場所にカスタムフェーズを追加したりシステムフェーズを移動したりできます。ビルド処理を正しく実行させるためには、カスタムフェーズの出力が他のフェーズに入力される場合、フェーズの順序を正しく設定することが必要です。



[ビルド->ビルドフェーズ]を選択すると[ビルドフェーズ]ダイアログボックスを表示します。[ビルドフェーズ]ダイアログボックスでビルドフェーズの順序を変更できます。このダイアログボックスにはフェーズの順序に関するタブが 2 つあります。[ビルド順序]タブと[ファイルのビルド順序]タブです。変更後[OK]ボタンをクリックしてください。

3.3.1 ビルド順序タブ

[ビルド順序]タブは、[ビルド]（）または[すべてをビルド]（）操作で実行されるフェーズの順序を示します。各フェーズの左にあるチェックボックスのオンオフの有無はそのフェーズの有効/無効を示します。ビルド時はこのチェックボックスがチェックされているフェーズを実行します。

また、以下の操作が可能です。

[ビルド]または[すべてをビルド]でシステムフェーズ／カスタムフェーズの順序を変更するには

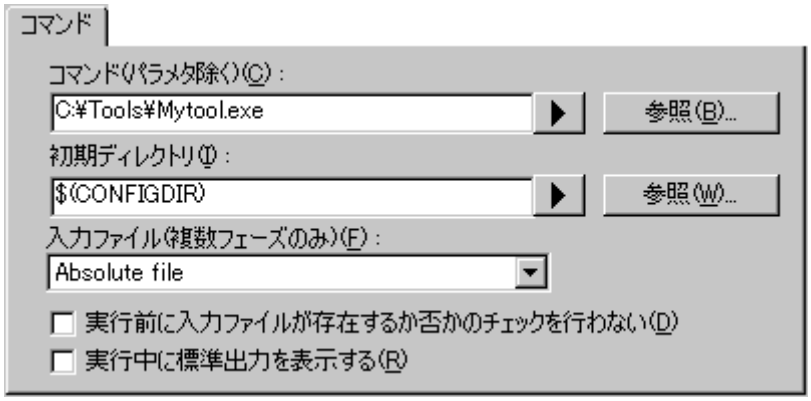
1. 移動するフェーズを選択してください。
2. [上へ]または[下へ]ボタンをクリックすると上下に移動します。

システムフェーズのプロパティを表示するには

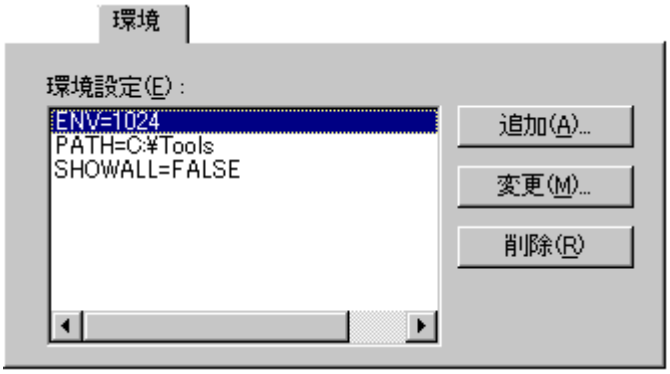
1. プロパティを表示するシステムフェーズを選択してください。
2. [変更]ボタンをクリックしてください。
3. ダイアログボックスで、システムフェーズのプロパティを表示します。[OK]ボタンをクリックしてください。

カスタムフェーズを変更するには

1. 変更するカスタムフェーズを選択してください。
2. [変更]ボタンをクリックしてください。Phase の変更ダイアログボックスの[コマンド]タブを表示します。



3. 必要に応じてフィールドの内容を変更してください。
4. 複数フェーズの場合、フェーズを実行する入力ファイルタイプを[入力ファイル（複数フェーズのみ）]ドロップダウンリストから選択してください。
5. 入力ファイルがなくてもフェーズの実行を中断したくない場合、[実行前に入力ファイルが存在するか否かのチェックを行わない]チェックボックスをオンにしてください。
6. 標準出力への出力をコマンド実行中に表示したい場合は、[実行中に標準出力を表示する]チェックボックスをオンにしてください。
7. [環境]タブを選択しフェーズの環境設定を行ってください。



8. 環境変数の追加は[追加]ボタン、変更は[変更]ボタン、削除は[削除]ボタンを使用してください。操作は前節と同じです。
9. [出力シンタックス]タブを選択しカスタムフェーズのエラー、ウォーニング、およびインフォメーションを定義してください。

アウトプットウィンドウの[Build]タブのエラーメッセージ出力行が[出力シンタックス]の設定と一致する場合、左側の余白にアイコンを表示したり、該当ソースをエディタに表示したりできます。

例えば、Renesas SuperH C/C++ Compiler の場合は下記のようなエラーメッセージをアウトプットウィンドウの[Build]タブに出力します。

C:¥HEW¥Workspace¥SH¥SH¥resetprg.c(70) : C2225 (E) Undeclared name "a"

これを定義する場合は、次の出力シンタックスを設定できます。

$(FULLFILE)($(LINE))*(E)*

この場合の出力されるエラーメッセージおよび指定するシンタックスの対応は下記になります。

エラーメッセージ例	出力シンタックス例	説明
C:¥HEW¥Workspace¥SH¥SH¥resetprg.c	$(FULLFILE)	フルパスファイル名部分をフルパスファイル名のプレースホルダ（$(FULLFILE)）で指定しています
(70)	($(LINE))	括弧付きの行番号部分を行番号のプレースホルダ（$(LINE)）を丸括弧で囲んで指定しています
: C2225	*	“:“およびエラーメッセージ番号部分をワイルドカードで指定しています
(E)	(E)	エラーを示す(E)をそのまま指定しています
Undeclared name "a"	*	エラーメッセージ部分をワイルドカードで指定しています

この正規表現フォーマットにはいくつかの制限があります。2 つの互いに曖昧な文字をすぐ隣に定義することはできません。例えば、"**"、"*$(FILEDIR)、または "$(FULLFILENAME)*"のような任意の組み合わせは定義できません。

・シンタックスタイプで"Error"を選択した場合：

アウトプットウィンドウの[Build]タブの左側の余白上にエラーアイコン( )を出力します。また、アウトプットウィンドウの[Build]タブのメッセージ出力行をダブルクリックすると、その行のフルパスファイル名および行番号に基づきエディタに該当ソース行を表示します。

・シンタックスタイプで"Warning"を選択した場合：

アウトプットウィンドウの[Build]タブの左側の余白上にウォーニングアイコン( )を出力します。また、アウトプットウィンドウの[Build]タブのメッセージ出力行をダブルク

リックすると、その行のフルパスファイル名および行番号に基づきエディタに該当ソース行を表示します。

・シンタックスタイプで"Other"を選択した場合：

アウトプットウィンドウの[Build]タブの左側の余白上にインフォメーションアイコン( )を出力します。

ビルドフェーズの終了時にエラーおよびウォーニング数の合計をアウトプットウィンドウの[Build]タブに表示します。

10. 出力シンタックスの追加は[追加]ボタン、変更は[変更]ボタン、削除は[削除]ボタンを使用してください。
11. 変更後[OK]ボタンをクリックしてください。

注：

システムフェーズは、[ツールアドミニストレーション]ダイアログボックスで、[環境]のみ変更できます。

カスタムフェーズを削除するには

1. 削除するフェーズを選択し、[削除]ボタンをクリックしてください。
2. 確認ダイアログボックスが表示された場合は、[はい]ボタンをクリックしてください。

カスタムフェーズを取り込むには

1. [取り込み]ボタンをクリックしてください。[カスタムフェーズ導入]ダイアログボックスを表示し、カスタムフェーズを取り込むための既存のプロジェクトを見ることができます。
2. カスタムフェーズを取り込みたいプロジェクトの位置を選択してください。選択すると、取り込み可能なプロジェクトのカスタムフェーズを並べた[フェーズの導入]ダイアログボックスを表示します。
3. どのフェーズを取り込むか決定したら、そのフェーズをリスト上で強調表示し、[OK]ボタンをクリックしてください。そのフェーズはビルドフェーズ]ダイアログボックスの[ビルドフェーズの順序]リストの最後に追加されます。

3.3.2 ファイルのビルド順序タブ

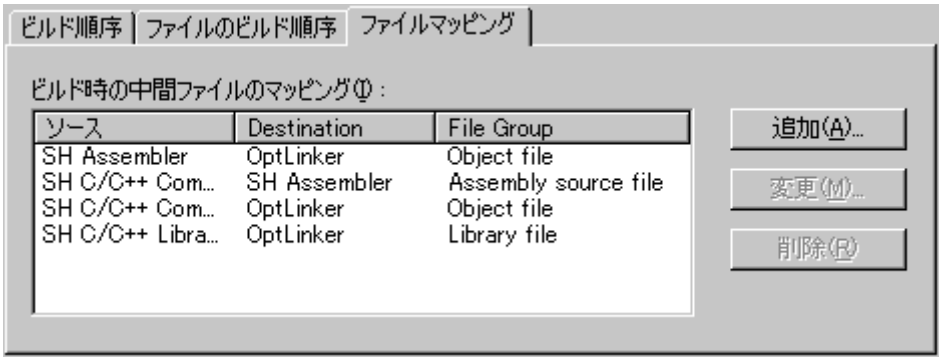
ワークスペースウィンドウから C ソースファイルを選んで[ビルド->コンパイル]を選択するか、[ファイルのコンパイル]ツールバーボタンをクリックすると、ファイルをコンパイルします。同じように、ワークスペースウィンドウからアセンブリ言語ソースファイルを選択し[ビルド->コンパイル]を実行すると、ファイルをアセンブルします。ファイルグループと実行するフェーズの関係は[ビルドフェーズ]ダイアログボックスの[ファイルのビルド順序]タブで管理されています。

リストには、[ファイルグループ]リストボックスのファイルグループに対してビルドファイル操作を選択したとき実行されるすべての現在のフェーズを表示します。下の図では“C source file”ファイルグループを選択しており、“Compiler”フェーズと“MyPhase”フェーズが関連付けされています。

[ビルド順序]タブのリストに新しいエントリを追加すると、自動的に[ファイルのビルド順序]タブの[フェーズの順序]リストに新しいエントリを追加します。例えば、“C source file”を入力とするフェーズを追加します。このフェーズは、“ファイルのビルド”操作を“C source file”に適用する時に実行するフェーズのリストに自動的に追加します。[ビルド->コンパイル]を選んだときに実行したくないフェーズがある場合、[フェーズの順序]リストのフェーズ名の左にあるチェックボックスをオフにしてください。

3.3.3 ファイルマッピングタブ

デフォルトでは、フェーズに入力できるファイルはプロジェクトに含まれるものだけです。つまり、[新規ビルドフェーズ]ダイアログボックスの[入力ファイルの選択]ドロップダウンリストに指定した種類のプロジェクトファイルだけです。[ビルド->ビルドフェーズ]を選択すると[ビルドフェーズ]ダイアログボックスを表示します。もし前のフェーズから出力されたファイル（中間ファイル）をフェーズで使いたい場合は、[ビルドフェーズ]ダイアログボックスの[ファイルマッピング]タブで定義してください。

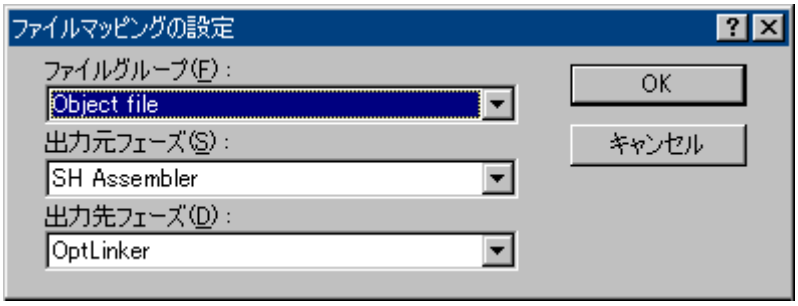


ファイルマッピングでは、あるフェーズ（出力元フェーズという）で作成したある種類の出力ファイル（中間ファイル）を、他のフェーズ（出力先フェーズという）に渡すように指定します。プロジェクトファイルに加えて中間ファイルも渡します。

ファイルのマッピングを追加するには

1. [追加]ボタンをクリックしてください。[ファイルマッピングの設定]ダイアログボックスを表示します。
2. [ファイルグループ]ドロップダウンリストボックスから新しいマッピングのためのファイルグループを選択してください。

3. [出力元フェーズ]ドロップダウンリストボックスから出力元フェーズ（ファイルを作成するフェーズ）を選択してください。
4. [出力先フェーズ]ドロップダウンリストボックスから出力先フェーズ（ファイルを渡す先のフェーズ）を選択してください。
5. [OK]ボタンをクリックすると新しいマッピングを追加します。



ファイルマッピングを変更するには

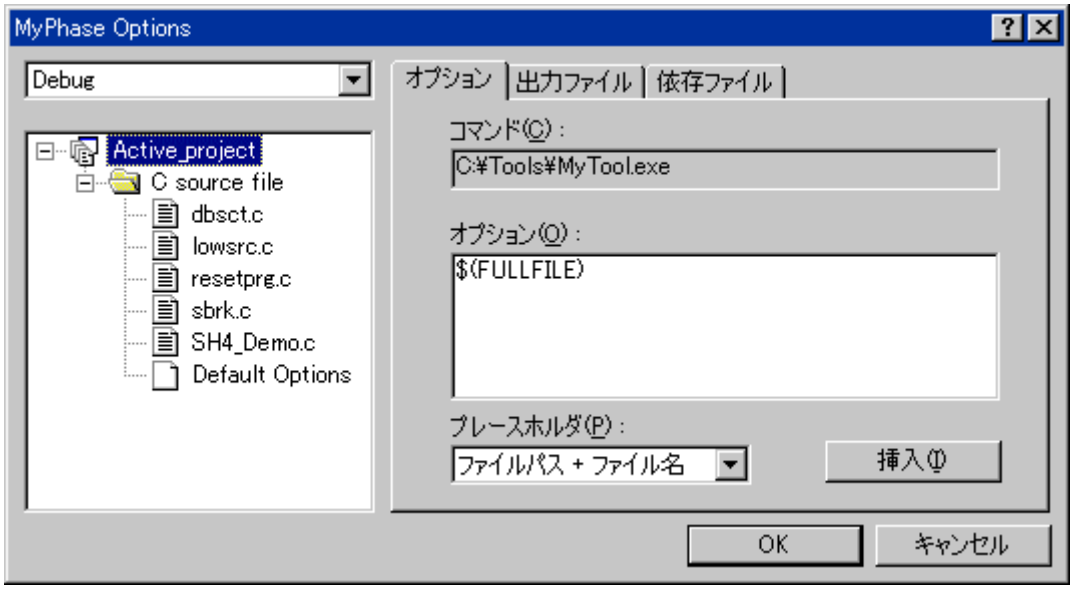
1. 変更するマッピングを選択してください。
2. [変更]ボタンをクリックしてください。[ファイルマッピングの設定]ダイアログボックスを表示します。
3. 必要に応じてオプションを変更してください。
4. [OK]ボタンをクリックすると変更が有効になります。

ファイルマッピングを削除するには

1. 削除するマッピングを選択してください。
2. [削除]ボタンをクリックしてください。
3. [OK]ボタンをクリックすると変更が有効になります。

3.4 カスタムビルドフェーズのオプション設定

カスタムフェーズを定義後、フェーズ実行時に使用するコマンドラインオプションを指定します。定義されたフェーズにはそれぞれ[ビルド]メニューにオプション設定用のメニューがあります。指定するフェーズのオプションを選択してください。起動するダイアログボックスは、選択したカスタムフェーズが単一フェーズか複数フェーズかによって異なります（新規ビルドフェーズウィザードでフェーズが作成されたときのフェーズ種類の選択によります）。

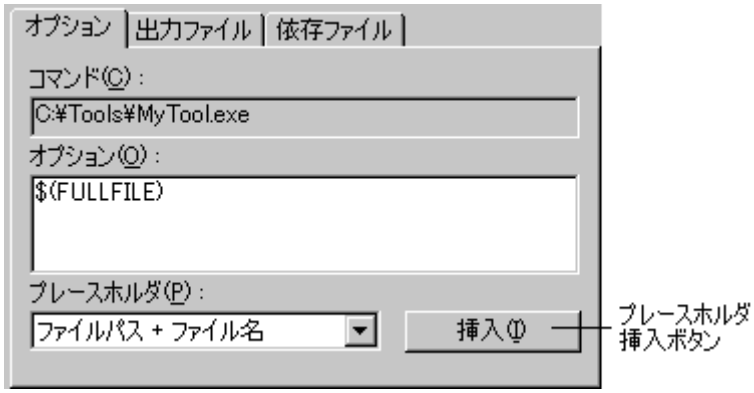


選択したフェーズが複数の場合、ダイアログボックスの左側にはプロジェクトファイルのリストが表示され、ファイル上でビルドオプションを指定できます。選択したフェーズが単一の場合、プロジェクトファイルのリストは表示されません。

右側には 3 つのオプションタブがあります。選択したファイルに適用するオプションをここで設定してください。また、ダイアログボックス左上のコンフィグレーションリストで、どのコンフィグレーションの情報を表示するか選択できます。各コンフィグレーションは[Multiple configurations]という名のエントリと一緒に並べられています。[Multiple configurations]を選択すると、ダイアログボックスが表示され、複数のコンフィグレーションを選択できます。

3.4.1 オプションタブ

[オプション]タブではフェーズに渡されるコマンドラインオプションを定義できます。[コマンド]フィールドではフェーズを定義したときに入力したコマンドを表示します。[オプション]フィールドにはコマンドに渡すコマンドライン引数を入力してください。プレースホルダを挿入する場合は、対応するプレースホルダを[プレースホルダ]ドロップダウンリストボックスから選択し、[挿入]ボタンをクリックしてください。プレースホルダの詳細については、「リファレンス 5. プレースホルダ」を参照してください。



3.4.2 出力ファイルタブ

[出力ファイル]タブではフェーズで作成される出力ファイルを指定します。HEW では、ファイルがこのフェーズを通過する前に、出力ファイルの日付が入力ファイルの日付より古いことをチェックしています。出力ファイル作成後入力ファイルが変更された場合、入力ファイルに対してこのフェーズを実行します。出力ファイルが最新の場合、入力ファイルに対してこのフェーズを実行しません。

注：

出力ファイルを指定しない場合フェーズを常に実行します。

出力ファイルを追加するには

1. [追加]ボタンをクリックしてください。[出力ファイルの追加]ダイアログボックスを表示します。
2. ファイル名を入力するか、[参照]ボタンで選択してください。
3. [OK]ボタンをクリックすると、リストに出力ファイルを追加します。

出力ファイルを変更するには

1. 変更する出力ファイルを選択してください。
2. [変更]ボタンをクリックすると[出力ファイルの変更]ダイアログボックスを表示します。
3. フィールドを変更して[OK]ボタンをクリックしてください。変更した項目をリストに追加します。

出力ファイルを削除するには

1. 削除する出力ファイルを選択してください。
2. [削除]ボタンをクリックしてください。

出力ファイルのパスにはプレースホルダを使用できます。

3.4.3 依存ファイルタブ

[依存ファイル]タブではフェーズに必要な依存ファイルを指定します。HEW では、各ファイルがこのフェーズを通過する前に、依存ファイルの日付が入力ファイルの日付より新しいか否かをチェックしています。チェック後、依存ファイルの日付が新しい場合（つまり、入力ファイル作成後に依存ファイルが変更された場合）このフェーズでファイルを実行します。依存ファイルの日付が入力ファイルの日付より古い場合、このフェーズは実行しません。

依存ファイルを追加するには

1. [追加]ボタンをクリックしてください。[依存ファイルの追加]ダイアログボックスを表示します。
2. ファイル名を入力するか、[参照]ボタンで選択してください。
3. [OK]ボタンをクリックすると、リストに出力ファイルを追加します。

依存ファイルを変更するには

1. 変更する依存ファイルを選択してください。
2. [変更]ボタンをクリックすると[依存ファイルの変更]ダイアログボックスを表示します。
3. フィールドを変更して[OK]ボタンをクリックすると、変更した項目をリストに追加します。

依存ファイルを削除するには

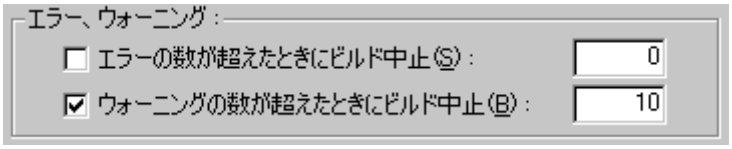
1. 削除するファイルを選択してください。
2. [削除] ボタンをクリックしてください。

依存ファイルのパスにはプレースホルダを使用できます。

3.5 ビルドを管理する

デフォルトでは、HEW はビルドのすべてのフェーズを実行し、途中で致命的なエラーが起こったときだけ中止します。これは[オプション]ダイアログボックスの[ビルド]タブで変更できます。

[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[ビルド]タブを表示してください。ある一定の回数を超えるエラーが起きた後にビルドを中止したいとき、[エラーの数が超えたときにビルド中止]チェックボックスをオンにして右のフィールドにエラー数を入力してください。指定した数を超えるウォーニングが表示されたときにビルドを中止したいとき、[ウォーニングの数が超えたときにビルド中止]チェックボックスをオンにして右のフィールドにウォーニング数を入力してください。



[ビルド]タブでは、エラー数やウォーニング数の制限のほかに、コマンドライン、環境、初期ディレクトリの表示の有無を設定できます。表示するには、それぞれのチェックボックスをオンにしてください。

注：

- 上記設定にかかわらず、致命的エラーが発生した場合、ビルドは停止します。
- [エラーの数が超えたときにビルド中止]をチェックして数値を指定した時、次に示す項目に注意してください。
  - i. ユーザが定義したカスタムビルドフェーズのエラー数は本機能の対象外です。C/C++ Compiler、Assembler、OptLinker などルネサスからあらかじめ提供されているシステムビルドフェーズで本機能は有効になります。
  - ii. エラーの数は 1 回のツールの実行で発生したエラーの数です。複数回のツールの実行で発生した累積エラー数は本機能と関係ありません。例えば、C/C++ Compiler の場合、1 回の 1 ファイルのコンパイルで発生したエラーの数が指定した数を超えた場合にビルドがストップします。各コンパイルのエラー数が指定した数以下ならば、ビルド中に複数ファイルをコンパイルした時の累積エラー数が指定したエラー数を超えてもビルドはストップしません。
  - iii. ビルドは該当フェーズ完了時にストップします。例えば、ビルド中に複数ファイルのコンパイルをする場合、最初のファイルのコンパイルでエラー数が指定した数値を超えた場合でも次のファイルのコンパイルを続けます。コンパイルのフェーズ全ファイルのコンパイルが完了したらビルドをストップします。
  - iv. 1 回のツールの実行でエラー数が指定した数を超えた場合、指定したエラー数に 1 を加えた数のエラーメッセージまでを Output ウィンドウに表示しますが、それ以上のエラーメッセージは表示しません。また、エラー数が指定した数値を超えた旨のメッセージもアウトプットウィンドウに表示されませんのでご注意ください。

なお、[ウォーニングの数が超えたときにビルド中止]についても[エラーの数が超えたときにビルド中止]と同様です。また、[ウォーニングの数が超えたときにビルド中止]と[エラーの数が超えたときにビルド中止]とは相互の関係はなく、それぞれ独立した機能です。

3.6 ビルドの出力のログを取る

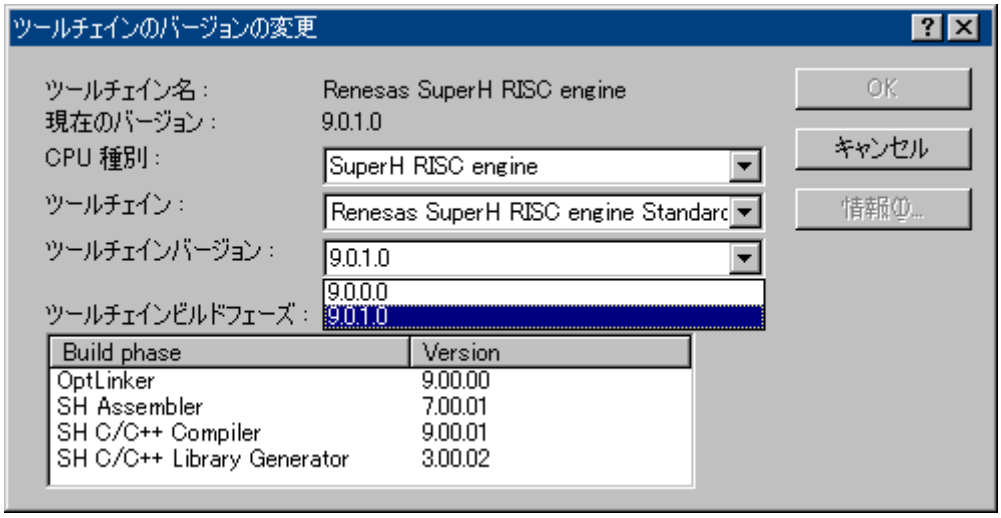
ファイルに各ビルドの結果を保存できます。

ログファイルを指定するには

1. [基本設定->カスタマイズ]メニューオプションを選択してください。[カスタマイズ]ダイアログボックスを表示します。
2. [ログ]タブを選択してください。
3. [ビルドログファイル生成]チェックボックスをオンにしてください。
4. [ディレクトリ]フィールドにログファイルのフルパスを入力するか、[参照]ボタンをクリックしてパスを選択するか、またはプレースホルダボタン( )をクリックし表示されるポップアップメニューからプレースホルダを選択してください。
5. [OK]ボタンをクリックしてください。

3.7 ツールチェインのバージョンを変更する

同じツールチェインの 2 つ以上のバージョンが HEW に登録されているとき[ツールチェインのバージョンの変更]ダイアログボックスでバージョンを選択できます。このダイアログボックスを表示するには[ツール->ツールチェインバージョンを変更]を選択してください。このダイアログボックスの[ツールチェインバージョン]ドロップダウンリストからバージョンを選択し、[OK]ボタンをクリックしてください。



ツールチェインを構成するツールの情報を表示するには[ツールチェインのバージョンの変更]ダイアログボックスの[ツールチェインビルドフェーズ]ドロップダウンリストからツールを 1 つ選択し、[情報]ボタンをクリックしてください。ツール情報ダイアログボックスにそのツールの情報を表示します。このダイアログボックスを閉じるには[閉じる]ボタンをクリックしてください。

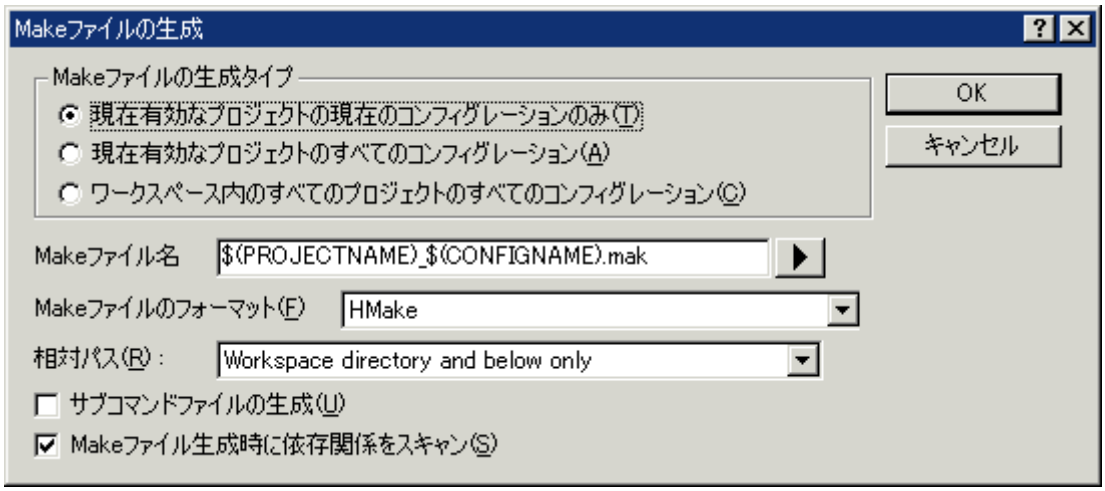
3.8 Make ファイルの生成

HEW では、ワークスペースに関連した Make ファイルを生成できます。生成可能な Make ファイルのフォーマットは、HMake、NMake、および GNUMake です。Make ファイルを実行する Make ツールは、DOS コマンドプロンプト上で動作するツールであり、生成した Make ファイルを使用することで、HEW を起動しなくてもプロジェクトをビルドすることができます。

生成した Make ファイルは、HEW 上で実行することができます。この設定は、「3.9 Make ファイルを使ったビルド」を参照してください。

Make ファイルを生成するには

1. Make ファイルを生成するワークスペースを開いてください。
   （ワークスペースに複数のプロジェクトを登録している場合は、Make ファイルを生成するプロジェクトをアクティブプロジェクトとしてください）
2. Make ファイルを生成するコンフィグレーションを選択してください。
3. [ビルド->Make ファイルの作成]を選択してください。
4. [Make ファイルの生成]ダイアログボックスが表示されます。



5. [Make ファイルの生成タイプ]から生成したい Make ファイルのタイプを選択してください。
   - 現在有効なプロジェクトの現在のコンフィグレーションのみ（デフォルト）
   - 現在有効なプロジェクトのすべてのコンフィグレーション
   - ワークスペース内のすべてのプロジェクトのすべてのコンフィグレーション
6. [Make ファイル名]エディットボックスに Make ファイルのファイル名を入力してください。プレースホルダを挿入する場合は、挿入する箇所にカーソルを移動して、プレースホルダボタン(▶)をクリックし、ポップアップメニューからプレースホルダを選択してください。
   Make ファイルのファイル名のデフォルト値は以下の通りです。
   - [現在有効なプロジェクトの現在のコンフィグレーションのみ]を選択した場合：
     $(PROJECTNAME)_$(CONFIGNAME).mak
   - [現在有効なプロジェクトのすべてのコンフィグレーション]を選択した場合：
     $(PROJECTNAME).mak
   - [ワークスペース内のすべてのプロジェクトのすべてのコンフィグレーション]を選択した場合：
     $(WORKSPNAME).mak
7. [Make ファイルのフォーマット]ドロップダウンリストから生成する Make ファイルのフォーマットを選択してください。生成可能な Make ファイルのフォーマットは、HMake、NMake、および GNUMake です。
8. [相対パス]ドロップダウンリストから Make ファイル内のディレクトリ表現方法を選択してください。

デフォルト値は、”Workspace directory and below only”です。選択可能なディレクトリ表現方法は以下の通りです。

オプション	ワークスペースディレクトリおよびそれ以下のディレクトリ	ワークスペースディレクトリの外のディレクトリ
None	絶対パス	絶対パス
Workspace directory and below only	相対パス	絶対パス
All	相対パス	相対パス

注：

SuperH または H8 ファミリのツールチェインをご使用の場合、”Workspace directory and below only”に固定されています。

9. HEW をインストールしたディレクトリに空白がある場合、Make ファイル生成機能で Make ファイルのフォーマットに”GNU Make”を選択すると、"GNU Make"コマンドが正しく動作しないことがあります。[サブコマンドファイルの生成]チェックボックスは、オンにすると Make ファイルが出力されるディレクトリ内にコマンドファイルを生成します。デフォルトはオフです。

注：

SuperH または H8 ファミリのツールチェインをご使用の場合、[Make ファイルのフォーマット]で GNUMake を選択すると[サブコマンドファイルの生成]チェックボックスはオンに固定されています。

10. [Make ファイルの生成時に依存関係をスキャン]チェックボックスは、オンにすると Make ファイルを生成する前に依存関係の更新を行います。デフォルトはオンです。
11. [OK]ボタンをクリックしてください。

HEW は、ワークスペースディレクトリ内に”make”ディレクトリを作り、その下に Make ファイルを生成します。[Make ファイルのフォーマット]で HMake を選択して作成した Make ファイルは、HEW インストールディレクトリにある HMAKE.EXE で実行することができます。

Make ファイルを編集する場合は、「リファレンス 13. HMAKE ユーザガイド」を参照してください。

注：

HEW をインストールしたディレクトリに空白がある場合、Make ファイル生成機能で Make ファイルのフォーマットに”GNU Make”を選択すると、"GNU Make"コマンドが正しく動作しないことがあります。

Make ファイル（HMake 用）を実行するには

1. DOS コマンドプロンプトウィンドウを開き、Make ファイルが生成されたディレクトリに移動してください。
2. HMAKE を実行してください。コマンドラインは、
   HMAKE.EXE <Make ファイル名>
   です。

注：

生成した Make ファイルが移動可能か否かは、プロジェクト自体が移動可能か否かに依存します。たとえば、出力ディレクトリやインクルードファイルディレクトリへのフルパスを含むコンパイラオプションがあると、異なるインストール環境下の別のユーザがビルドした場合、失

敗する可能性があります。一般的に、できるだけプレースホルダを使用して、フルパス、または特定のパスの使用はなるべく避けてください。

3.9 Make ファイルを使ったビルド

HEW は、HEW 標準のビルド処理のほかに、特定の Make ツール（hmake、nmake、gmake）を使用したビルドを実行することができます。実行可能な Make ファイルは、ワークスペースから生成した Make ファイルに限定されます。

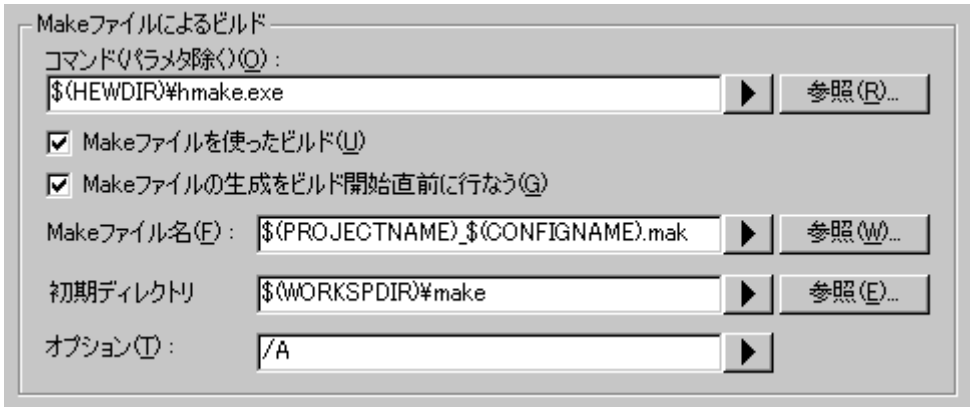
Make ファイルを使ったビルド方法について以下に説明します。

nmake、gmake は、お客様ご自身でご用意ください。

hmake は、HEW インストールディレクトリにインストールされます。

Make ツールの実行を設定するには

1. HEW のワークスペースを作成してください。
2. [ビルド->Make ファイルの生成]を選択し、Make ファイルを生成してください。Make ファイル生成の詳細は、前節の「3.8 Make ファイルの生成」を参照してください。
3. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスが表示されます。
4. [ビルド]タブを選択してください。



5. [コマンド（パラメタ除く）]エディットボックスに Make ファイルを実行する Make ツールを入力してください。デフォルトは、hmake（$(HEWDIR)¥hmake.exe）が設定されています。
6. [Make ファイルを使ったビルド]チェックボックスをオンにしてください。このチェックボックスをオンにすると、ビルドが標準のビルド処理ではなく Make ファイルを使ったビルドを実行するようになります。
7. [Make ファイルの生成をビルド開始直前に行なう]チェックボックスは、オンにするとビルド開始直前に毎回 Make ファイルを生成し直します。Make ファイルが、常に最新の HEW プロジェクトの状態に対応するようになります。
8. [Make ファイル名]エディットボックスに Make ファイルを入力してください。デフォルト値は、$(PROJECTNAME)_$(CONFIGNAME).mak が設定されています。
9. [初期ディレクトリ]エディットボックスに Make ファイルを実行するときのカレントディレクトリを入力してください。このエディットボックスが空の場合は、[Make ファ

イル名]エディットボックスに設定した Make ファイルが存在するディレクトリをカレントディレクトリとします。

10. [オプション]エディットボックスに Make ファイルをビルドするときの Make オプションを入力してください。
11. [OK]ボタンをクリックしてください。

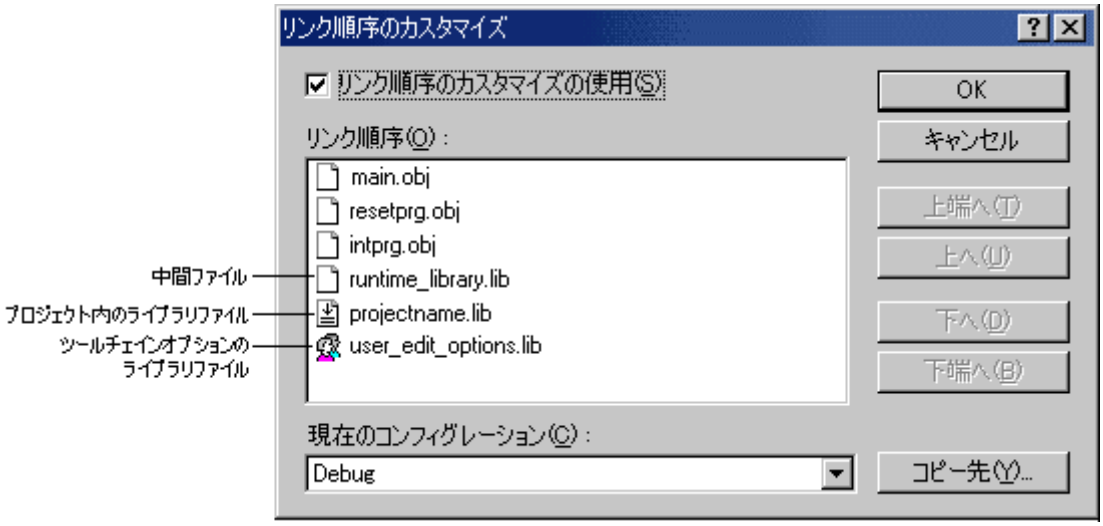
[Make ファイルを使ったビルド]チェックボックスをオンに設定すると、ビルドボタン、メニュー、およびキーボードのショートカットは Make ツールの実行制御に変更されます。標準のビルド処理同様、結果の出力はすべてアウトプットウィンドウの[Build]タブです。HEW がサポートしているビルドフェーズが実行された場合、エラーメッセージ出力行をダブルクリックすると、ソースファイルにジャンプすることができます。エラーメッセージ出力行を選択して F1 キーを押すと、エラーメッセージに関するヘルプを表示します。HMAKE.exe を使用する場合のみ、[すべてをビルド]ツールバーボタン( )をクリックすると、すべてのファイルを再ビルドすることができます。

3.10 リンク順序をカスタマイズする

HEW では、通常オブジェクトファイルはファイル名のアルファベット順にリンクしますが、リンクする順番をカスタマイズできます。

リンク順序をカスタマイズするには

1. [ビルド->リンク順の指定]を選択してください。[リンク順序のカスタマイズ]ダイアログボックスを表示します。
2. [リンク順序のカスタマイズの使用]チェックボックスをオンにしてください。
3. オブジェクトの順序を並びかえることができます。モジュールを選択し、[上端へ]ボタン、[上へ]ボタン、[下へ]ボタン、または[下端へ]ボタンをクリックして任意の位置に移動してください。
4. 各モジュールには、そのモジュールの属性によって異なるアイコンを表示しています。これを下図に示します。



5. [OK]ボタンをクリックしてください。

複数のコンフィグレーションを使用する場合、リンク順序を同じにする場合があります。現在のリンク順序の設定を他のコンフィグレーションにコピーする方法を以下に説明します。

他のコンフィグレーションにリンク順序をコピーするには

1. [ビルド->リンク順の指定]を選択してください。[リンク順序のカスタマイズ]ダイアログボックスを表示します。
2. [現在のコンフィグレーション]ドロップダウンリストで、コピー元のコンフィグレーションを選択してください。現在のコンフィグレーションをデフォルトで表示します。
3. [コピー先]ボタンをクリックすると[コンフィグレーションのコピー先選択]ダイアログボックスを表示します。現在のリンク順序を現在のプロジェクトのどのコンフィグレーションにコピーするかを指定します。コンフィグレーションを選択し、[OK]ボタンをクリックしてください。
4. [OK]ボタンをクリックしてください。

4. エディタの使用

この章では High-performance Embedded Workshop (HEW) が提供するエディタの使い方を説明します。

4.1 エディタウィンドウ

エディタウィンドウには、表示中や編集中のファイルがあります。

アクティブウィンドウのタイトルバーは他のウィンドウとは色が異なります。（図では“dbsct.c”がアクティブウィンドウです。）文字入力やテキスト貼り付けなどのテキスト操作はアクティブウィンドウでのみ行うことができます。

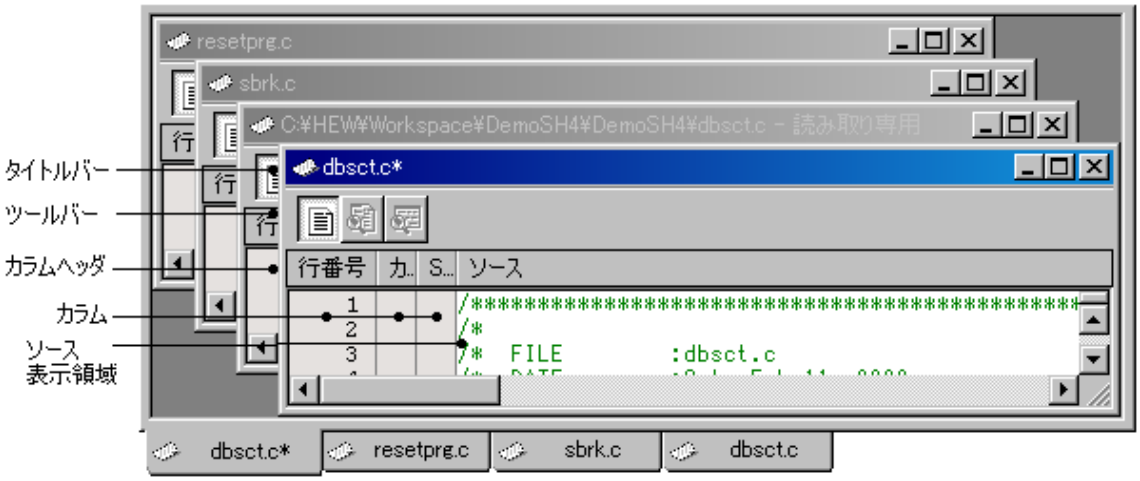
アクティブウィンドウから他のソースファイルウィンドウへ切り替えるには（他のウィンドウをアクティブウィンドウにするには）、以下の方法があります。

(a) そのウィンドウが表示されていればそれをクリックする。

(b) Ctrl+Tab キーまたは Ctrl+Shift+Tab キーを押して順次ウィンドウをアクティブにする。

(c) [ウィンドウ]メニューからウィンドウ名を選択する。

(d) エディタウィンドウの下の該当するタブを選択する。



タイトルバー	ファイルの情報（ファイル名、編集中のアスタリスク (*)の表示、および読み取り専用属性）を表示します。 アクティブプロジェクトのディレクトリ下に含まれているファイルの場合は、エディタウィンドウのタイトルバーにファイル名を表示します。 アクティブプロジェクトのディレクトリ下に含まれていないファイルの場合は、エディタウィンドウのタイトルバーに絶対パスを含むファイル名を表示します。
ツールバー	ツールバーは、デバッギングプラットフォームがセッションに接続されている場合のみ表示されます。各表示モードが有効である場合、ツールバーのボタンをクリックすると表示モードを切り替えます。

	各ボタン上でマウスポインタを静止させるとボタン名がポップアップ表示されます。
カラムヘッダ	カラムヘッダ上のマウスのドラッグで各カラムの幅を調節できます。また、カラムヘッダ上で右クリックするとポップアップメニューを表示します。カラムが利用できる場合は、エントリの横にはチェックマークがあります。エントリをクリックすると、各カラムの表示、非表示を切り替えます。 各カラムヘッダ上でマウスポインタを静止させると、カラム名がポップアップ表示されます。 カラムヘッダ自体の表示、非表示を切り替えることもできます。
カラム	ソース表示領域の左側にはいくつかの余白（カラム）があります。 各カラム上でマウスポインタを静止させると、カラム名がポップアップ表示されます。
ソース表示領域	シンタックスを強調表示したコードを含みます。エディタの 1 行の最大文字数（半角）は、8,192 文字です。 この領域内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

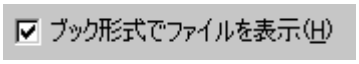
4.2 複数のファイルを扱う

プロジェクトのファイルを操作するときにはエディタウィンドウを使います。エディタによって、一度に複数のファイルを開いたり、ファイル間の切り替えをしたり、異なる構成に並べ替えたり、任意の順序に編集できます。ウィンドウ上での操作は Windows®のアプリケーションの一般的な機能であり、以下の[ウィンドウ]メニューから選ぶことができます。

メニューオプション	動作
[ウィンドウ->重ねて表示]	開いているすべてのウィンドウの左上が見えるように重ねて並べます。
[ウィンドウ->上下に並べて表示]	開いているすべてのウィンドウが互いに重ならずにエディタウィンドウの全体を占めるように並べます（垂直方向）。
[ウィンドウ->左右に並べて表示]	開いているすべてのウィンドウが互いに重ならずにエディタウィンドウの全体を占めるように並べます（水平方向）。
[ウィンドウ->アイコンの整列]	すべての最小化したウィンドウをエディタウィンドウの下に並べます。
[ウィンドウ->すべて閉じる]	開いているエディタウィンドウをすべて閉じます。

また、エディタのファイルはブック形式で表示できます。つまり、各ファイルにタブが付いていてファイル間の行き来を容易にできます。

ファイルをブック形式で表示するには



1. [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を表示します。
2. [エディタ]タブを選択してください。
3. ［ブック形式でファイルを表示]チェックボックスをオンにしてください。
4. [OK]ボタンをクリックしてください。

4.3 標準のファイル操作

4.3.1 新規ファイルを作成する

新しいエディタウィンドウを作成するには

次のいずれかを行ってください。

- “新規ファイル”ツールバーボタン ( ) をクリックする。
- ”Ctrl+N”キーを押す。
- [ファイル->新規作成]を選択する。

新しいウィンドウにはデフォルトで名前がつきます。ファイルを保存するときに名前を変更できます。

4.3.2 ファイルを編集する

HEW エディタは標準的な編集機能をサポートします。通常の方法（メニュー、ツールバー、キーボードのショートカット）で編集できるほか、各エディタウィンドウにあるポップアップメニューでも編集できます。ポップアップメニューを表示するには、開いたウィンドウ上で右クリックしてください。

以下の表に編集の基本操作を示します。

名称	機能	操作
元に戻す	最新の編集操作を取り消す	[編集->元に戻す] を選択 Ctrl+Z キーを押す
やり直し	最新の取り消した編集操作をやり直す	[編集->やり直し] を選択 Ctrl+Y キーを押す
切り取り	選択部分を削除し Windows®クリップボードに貼り付ける	[切り取り]ツールバーボタン( ) をクリック [編集->切り取り]を選択 [切り取り] - ポップアップメニューを選択 Ctrl+X キーを押す
コピー	選択部分をコピーし Windows®クリップボードに貼り付ける	[コピー]ツールバーボタン( ) をクリック [編集->コピー] を選択 [コピー] - ポップアップメニューを選択 Ctrl+C キーを押す
貼り付け	Windows®クリップボードの内容をコピーしてアクティブウィンドウのカーソル位置に貼り付ける	[貼り付け]ツールバーボタン( ) をクリック [編集->貼り付け]を選択 [貼り付け] - ポップアップメニューを選択 Ctrl+V キーを押す
削除	選択部分を削除する（Windows®クリップボードに貼り付けない）	[編集->削除] を選択 Delete キーを押す
すべて選択	アクティブウィンドウの内容すべてを選択する	[編集->すべて選択] を選択 Ctrl+A キーを押す

ファイルを編集すると、エディタウィンドウのタイトルバーにアスタリスク（*）を表示します。（例：filename.c*）

アスタリスクはファイルを保存するまで表示します。現在のウィンドウですべての編集が戻された場合には、アスタリスクの表示が消えます。

エディタでテキストを選択する

すべてのエディタと同じ方法でテキストを選択できます。ただし、列を選択するには、Alt キーを押した状態でマウスによりテキストを選択してください。

4.3.3 ファイルを保存する

エディタウィンドウの内容を保存するには

1. 内容を保存するウィンドウがアクティブであることを確認してください。
2. 次のいずれかを行ってください。

- [ファイルの保存]ツールバーボタン ( ) をクリックする。
- ”Ctrl+S”キーを押す。
- [ファイル->上書き保存]を選択する。

3. ファイルを新規に保存する場合、名前を付けて保存するダイアログボックスを表示します。ファイル名とディレクトリを指定して、[保存]ボタンをクリックしてください。
4. すでに保存したことのあるファイルの場合は、ダイアログボックスを表示せずにファイルを更新します。

エディタウィンドウの内容を新しい名前で保存するには

1. 内容を保存するウィンドウがアクティブであることを確認してください。
2. [ファイル->名前を付けて保存]を選択してください。
3. 名前を付けて保存するダイアログボックスを表示します。ファイル名とディレクトリを指定して[保存]ボタンをクリックしてください。

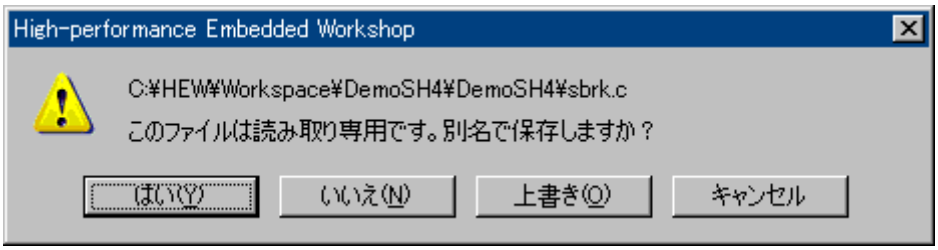
すべての開いているエディタウィンドウの内容を保存するには

1. 次のいずれかを行ってください。

- [すべて保存]ツールバーボタン ( ) をクリックする。
- [ファイル->すべて保存]を選択する。

2. 新規にファイルを保存する場合、名前を付けて保存するためのダイアログボックスを表示します。ファイル名とディレクトリを指定してください。すでに保存したことのあるファイルの場合は、ダイアログボックスを表示せずにファイルを更新します。

ファイルが読み取り専用属性の場合にファイルを保存しようとすると、別名で保存するか否かを確認するダイアログボックスが表示されます。



- 「はい」ボタンをクリックするとファイルを保存するダイアログボックスが表示されます。ファイル名を変更して保存してください。
- 「いいえ」ボタンをクリックするとファイルを保存しません。

- 「上書き」ボタンをクリックするとファイルを上書き保存します。ファイルは読み取り専用ではなくなります。
- 「キャンセル」ボタンをクリックするとファイルを保存しません。

4.3.4 ファイルを開く

[ファイルを開く]ダイアログボックスでファイルを開くには

1. 次のいずれかを行ってください。

- [ファイルを開く]ツールバーボタン( )をクリックする。
- “Ctrl+O”キーを押す。
- [ファイル->開く]を選択する。

2. [ファイルを開く]ダイアログボックスを表示します。右のディレクトリブラウザを使って、開きたいファイルのあるディレクトリに移動してください。[ファイルの種類]コンボボックスでファイルの種類を選択してください。(“All Files” (*.*) を指定すると、そのディレクトリのすべてのファイルを表示します。)
3. ファイルを選んで[開く]ボタンをクリックしてください。

HEW のメインウィンドウ上に Windows(R) Explorer からファイルをドラッグアンドドロップして開きます。

また、ワークスペースウィンドウの[Projects]タブのソースファイルは、次のいずれかの方法で個々のファイルが開けます。

- ファイルをダブルクリックする。
- ファイルを選択し、右クリックしてポップアップメニューから[開く <ファイル名>]オプションを選択する。
- ファイルが選択されフォーカスがある場合に Enter キーを押す。

ファイルは、HEW エディタ以外のエディタ（外部エディタと呼びます）で開くこともできます。詳細は、「6.9 HEW エディタ以外のエディタを使う」を参照してください。

開いた最新のファイル 4 つを[ファイル]メニューの[最近使ったファイル]サブメニューに追加します。この機能は最近開いたファイルを再び開きたいときに便利です。

最近利用したファイルを開くには

[ファイル->最近使ったファイル]メニューオプションを選択してこのサブメニューから開きたいファイルを選択してください。

4.3.5 ファイルを閉じる

個別にファイルを閉じるには、次のいずれかの方法で行ってください。

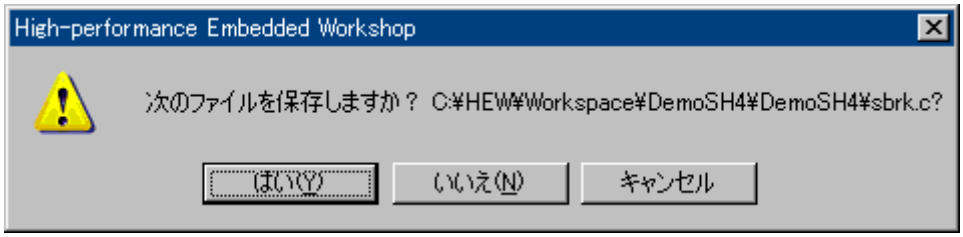
- エディタウィンドウのシステムメニュー（[最大化]表示されていないときの各ウィンドウの左上）をダブルクリックする。
- エディタウィンドウのシステムメニュー（[最大化]表示されていないときの各ウィンドウの左上）をクリックして[閉じる]メニューオプションを選択する。

- アクティブウィンドウであることを確認後”Ctrl+F4”キーを押す。
- アクティブウィンドウであることを確認後[ファイル->閉じる]を選択する。
- “閉じる”ボタン（[最大化]表示されていないときの各ウィンドウの右上）をクリックする。

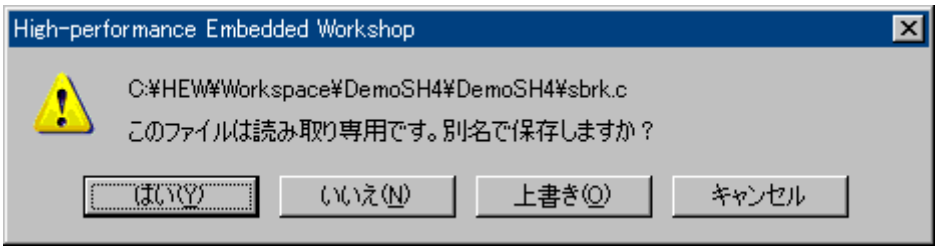
すべてのウィンドウを閉じるには

[ウィンドウ->すべて閉じる]を選択してください。

ファイルを編集し保存していない場合にファイルを閉じようとすると、保存するか否かを確認するダイアログボックスが表示されます。



- 「いいえ」ボタンをクリックするとファイルを保存せずにファイルを閉じます。
- 「キャンセル」ボタンをクリックするとエディタウィンドウに戻ります。
- 「はい」ボタンをクリックするとファイルを上書き保存しファイルを閉じます。ファイルが読み取り専用属性の場合に「はい」ボタンをクリックすると、別名で保存するか否かを確認するダイアログボックスが表示されます。



- 「はい」ボタンをクリックするとファイルを保存するダイアログボックスが表示されます。ファイル名を変更して保存してください。
- 「いいえ」ボタンをクリックするとファイルを保存せずにファイルを閉じます。
- 「上書き」ボタンをクリックするとファイルを上書き保存しファイルを閉じます。ファイルは読み取り専用ではなくなります。
- 「キャンセル」ボタンをクリックするとエディタウィンドウに戻ります。

4.4 検索とファイル内の移動

4.4.1 テキストの検索

現在のファイルのテキストを検索するには

1. 内容を検索するウィンドウがアクティブウィンドウであることを確認してください。
2. 検索開始位置にカーソルを置いてください。
3. 次のいずれかの操作を行って[検索]ダイアログボックスを表示してください。

- [検索]ツールバーボタン ( ) をクリックする。
- ”Ctrl+F”キーを押す。
- [編集->検索]を選択する。
- エディタウィンドウのポップアップメニューから[検索]を選択する。

4. [検索文字列]に検索するテキストを入力、またはドロップダウンリストボックスから以前検索した文字列を選択してください。ファイルウィンドウでテキストを選んでから[検索]ダイアログボックスを開くと、選んだテキストが自動的に[検索文字列]フィールドに入ります。
5. 単語単位で文字列を探す場合は、[単語単位で探す]チェックボックスをオンにしてください。このオプションを選択しない場合は、一部でも一致する文字列を検索します。
6. 大文字と小文字を識別する場合は[大文字、小文字を区別する]チェックボックスをオンにしてください。
7. 正規表現を使ってテキストを検索する場合[正規表現]チェックボックスをオンにしてください。詳細は「リファレンス 4. 正規表現」を参照してください。
8. [検索する方向] ラジオボタンにより、検索する方向を指定できます。[下へ]を選択すると、カーソル位置からファイルの下の方向に検索します。[上へ]を選択すると、カーソル位置から上の方向に検索します。
9. [次を検索]ボタンをクリックすると検索を始めます。検索を途中で終了するには[キャンセル]ボタンをクリックしてください。

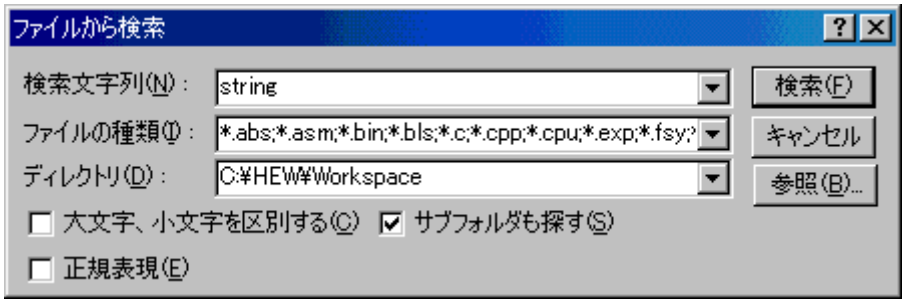
また、複数のファイル間での検索もできます。

4.4.2 複数のファイル間でのテキスト検索

複数のファイルでテキストを検索するには

1. 次のいずれかの操作を行って[ファイルから検索]ダイアログボックスを表示してください。

- [ファイルの中から検索]ツールバーボタン( )をクリックする。
- ”F4”キーを押す。
- [編集->ファイル内から検索]を選択する。



2. [検索文字列]フィールドに検索したいテキストを入力するか、ドロップダウンリストボックスから以前検索した文字列を選択してください。ウィンドウでテキストを選択してから[ファイルから検索]ダイアログボックスを開くと、そのテキストが自動的に[検索文字列]フィールドに入ります。
3. [ファイルの種類]フィールドに、検索するファイルの拡張子を指定してください。複数の拡張子を指定するときは、セミコロンで区切ってください（例：*.c;*.h）。
4. [ディレクトリ]フィールドに検索するファイルのディレクトリを指定してください。または、[参照]ボタンをクリックして画面上で検索するファイルを指定してください。
5. 大文字と小文字を識別したいときは[大文字、小文字を区別する]チェックボックスをオンにしてください。
6. 指定したディレクトリとその下のディレクトリすべてを検索したいときは、[サブフォルダも探す]チェックボックスをオンにしてください。[ディレクトリ]フィールドで指定したディレクトリだけを検索したいときは、[サブフォルダも探す]チェックボックスがチェックされていないことを確認してください。
7. 正規表現を使ってテキストを検索したいときは、[正規表現]チェックボックスをオンにしてください。詳細は「リファレンス 4. 正規表現」を参照してください。
8. [検索]ボタンをクリックすると検索を始めます。検索結果はアウトプットウィンドウの[Find in Files]タブに表示します。検索結果をダブルクリックすると、ファイルの該当文字列にジャンプします。実行中の[ファイル内から検索]操作を止めるには、[編集->ファイルから検索を中止]メニューオプションを選択します。ファイル内から検索操作が完了したらアウトプットウィンドウの文字列をダブルクリックすると、ファイルの該当文字列にジャンプすることができます。

4.4.3 テキストを置換する

テキストの置換は前節で説明したテキストの検索に似ています。異なる点は、テキスト検索後、他のテキストに置き換えるところです。

ファイルのテキストを置換するには

1. 内容を置換するウィンドウがアクティブであることを確認してください。
2. 検索を始める位置にカーソルを置いてください。
3. 次のいずれかの操作を行って[置換]ダイアログボックスを表示してください。

- ”Ctrl+H”キーを押す。
- [編集->置換]を選択する。
- エディタウィンドウのポップアップメニューから[置換]を選択する。

4. [置換文字列]フィールドに置換前のテキストを入力してください。または、ドロップダウンリストボックスから以前に検索した文字列を選んでください。ファイルウィンドウでテキストを選んでから[置換]ダイアログボックスを開くと、選んだテキストが自動的に[置換文字列]フィールドに入ります。
5. [置換後の文字列]フィールドに置換後のテキストを入力してください。または、ドロップダウンリストボックスから以前に検索した文字列を選んでください。
6. 単語単位で文字列を探す場合は、[単語単位で探す] チェックボックスをオンにしてください。このオプションを選択しない場合は、一部でも一致する文字列を検索します。
7. 大文字と小文字を識別したいときは[大文字、小文字を区別する] チェックボックスをチェックしてください。
8. 正規表現を使ってテキストを検索したいときは、[正規表現] チェックボックスをオンにしてください。詳細は「リファレンス 4. 正規表現」を参照してください。
9. [次を検索] をクリックすると、検索文字列、条件に合致したものの次を検索します。置換したい場合は[置換]をクリックしてください。[すべてを置換]をクリックすると、該当するすべての文字列を置換します。[キャンセル]をクリックすると、置換動作を停止します。なお、[置換]で[選択範囲のみ]を選択している場合はテキストの選択範囲が置換対象となり、[ファイル全体]を選択している場合はファイル全体が置換対象となります。[すべての開いているファイル]を選択している場合は、エディタで現在開いているファイルがすべて置換対象となります。

4.4.4 指定した行にジャンプする

ファイルの指定した行にジャンプするには

1. ファイルがアクティブであることを確認してください。
2. 次のいずれかの操作を行って[ジャンプ]ダイアログボックスを表示してください。

- ”Ctrl+G”キーを押す。
- [編集->ジャンプ]を選択する。
- エディタウィンドウのポップアップメニューから [ジャンプ]を選択する。

3. ダイアログボックスに指定する行番号を入力して、 [OK] ボタンをクリックしてください。
4. カーソルが指定した行の先頭に移ります。

4.5 ブックマーク

一度に多くの大容量のファイルを扱うとき、必要な行や領域を見つけるのが難しくなります。ブックマークをあらかじめ特定の行に設定しておくと、後でその行にジャンプできます。例えば、行数の多い C 言語ファイルの各関数の定義位置にブックマークを設定すると便利です。ブックマークは、解除するまでか、設定したファイルを閉じるまで有効です。

ブックマークを設定するには

1. ブックマークを設定する行にカーソルを置いてください。
2. 次のいずれかを行ってください。

- [編集->ブックマーク->ブックマークの挿入/削除]を選択する。
- ”Ctrl+F2”キーを押す。
- ポップアップメニューから [ブックマーク->ブックマークの挿入/削除]を選択する。
- [ブックマークの挿入/削除]ツールバーボタン()をクリックする。

3. その行にブックマークアイコン()を表示します。

ブックマークを解除するには

1. ブックマークを解除する行にカーソルを置いてください。
2. 次のいずれかを行ってください。

- [編集->ブックマーク->ブックマークの挿入/削除]を選択する。
- ”Ctrl+F2”キーを押す。
- ポップアップメニューから[ブックマーク->ブックマークの挿入/削除]を選択する。
- [ブックマークの挿入/削除]ツールバーボタン()をクリックする。

3. その行からブックマークアイコンが削除されます。

次のブックマークにジャンプするには

1. カーソルが検索するファイルの中にあることを確認してください。
2. 次のいずれかを行ってください。

- [編集->ブックマーク->次のブックマーク]を選択する。
- ”F2”キーを押す。
- ポップアップメニューから[ブックマーク->次のブックマーク]を選択する。
- [次のブックマークへ]ツールバーボタン()をクリックする。

同じファイルの 1 つ前のブックマークに戻るには

1. カーソルが検索するファイルの中にあることを確認してください。
2. 次のいずれかを行ってください。

- [編集->ブックマーク->前のブックマーク]を選択する。
- ”Shift+F2”キーを押す。
- ポップアップメニューから[ブックマーク->前のブックマーク]を選択する。

- [前のブックマークへ]ツールバーボタン( )をクリックする。

ファイル内のすべてのブックマークを解除するには

1. すべてのブックマークを解除するファイルのウィンドウをアクティブにしてください。
2. 次のいずれかを行ってください。

- [編集->ブックマーク->すべてのブックマークの削除]を選択する。
- ポップアップメニューから[ブックマーク->すべてのブックマークの削除]を選択する。
- [すべてのブックマークの削除]ツールバーボタン( )をクリックする。

4.6 ファイルを印刷する

ファイルを印刷するには

1. 印刷するファイルのウィンドウをアクティブにしてください。
2. 次のいずれかを行ってください。

- [印刷]ツールバーボタン ( ) をクリックする。
- ”Ctrl+P”キーを押す。
- [ファイル->印刷]を選択する。

4.7 テキストのレイアウト

4.7.1 ページ設定

HEW エディタからファイルを印刷するとき、印刷ダイアログボックスの設定により、印刷方法が変わります（例えば、片面印刷、両面印刷など）。また、[ページ設定]ダイアログボックスでは、印刷するテキストの余白（上下左右）を指定できます。プリンタによっては、A4 サイズの端まで印刷できない場合があるので、この指定が必要です。また、用紙の左端にとじしろを残したい場合などにも便利です。

ページの余白を設定するには

1. [ファイル->ページレイアウトの設定]を選択してください。[ページ設定]ダイアログボックスを表示します。
2. [余白]フィールドに必要な余白を入力してください（[インチ]または[ミリメートル]ラジオボタンを選択してください）。
3. [OK]ボタンをクリックしてください。

ページのヘッダとフッタを設定するには

1. [ファイル->ページレイアウトの設定]を選択してください。[ページ設定]ダイアログボックスを表示します。
2. ヘッダおよびフッタの編集フィールドに表示するテキストを入力してください。ページ番号、テキスト配置、日付の各フィールドとともに通常のプレースホルダが利用できます。プレースホルダはページが印刷される前に実際の値に展開されます。
3. [OK]ボタンをクリックしてください。

印刷の折り返しを設定するには

1. [ファイル->ページレイアウトの設定]を選択してください。[ページ設定]ダイアログボックスを表示します。
2. [折り返し]チェックボックスをクリックしてください。折り返し機能が有効になり、テキストをすべて印刷します。
3. [OK]ボタンをクリックしてください。

4.7.2 タブを変更する

タブの文字数を変更するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。[エディタ]タブを選択します。
2. [タブ幅]フィールドに必要なタブの文字数を指定してください。
3. [OK]ボタンをクリックしてください。

“Tab”キーを押すと、通常、タブ文字をファイルに入力します。しかし、タブ文字のかわりに空白文字列を入力したほうがよい場合があります。[オプション]ダイアログボックスの[エディタ]タブで、タブ文字を空白文字列として指定できます。

タブ文字の代わりに空白文字列を使用するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [エディタ]タブを選択してください。
3. タブ文字の代わりに空白文字列を使用する場合は[空白をタブとして使用]チェックボックスをオンにしてください。
4. [OK]ボタンをクリックしてください。

4.7.3 自動インデント

標準のエディタで Enter キーを押すと、カーソルは次の行の左端に移ります。自動インデントを設定すると、カーソルは、改行時、前行の最初の文字の下に移ります。したがって、自分でタブや空白を入力することなく、C/C++言語またはアセンブリ言語のコードをより速く見やすく入力できます。

以下に 2 つの例を示します。(i)は自動インデントの設定をしなかったときに Enter キーを押した場合の例を示します。カーソルは次の行の左端に移ります。"int z=20"の行は、上の 2 行と揃っていません。(ii)は自動インデントを設定して Enter キーを押した場合の例を示します。カーソルは前行の"int"の"i"の下に移ります。したがって、"int z=20"行を入力すると、テキストは自動的に前の行と整列（自動インデント）します。

```
void myFunction(void)
{
    int x = 5;
    int y = 10;
int z = 20;
```

(i) 自動インデントなし

```
void myFunction(void)
{
    int x = 5;
    int y = 10;
    int z = 20;
```

(ii) 自動インデントあり

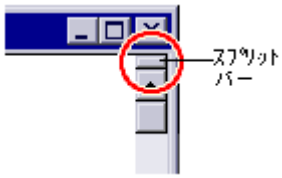
自動インデントを設定するには

1. [基本設定->オプション]メニューオプションを選択してください。[オプション]ダイアログボックスを表示します。
2. [エディタ]タブを選択してください。
3. [自動インデント]チェックボックスをオンにしてください。このチェックボックスのデフォルトはオンです。
4. [OK]ボタンをクリックすると、自動インデントを設定します。

☑ 自動インデント(N)

4.8 ウィンドウを分割する

HEW では、エディタウィンドウを 2 つに分割できます。以下に、エディタウィンドウの右上にある“閉じる”ボタンのすぐ下にあるスプリットバーボタンを示します。



ウィンドウを分割するには

スプリットバーボタンをダブルクリックしてウィンドウを 2 つに分割するか、スプリットバーボタンを押したままマウスを下に移動して分割したい位置でマウスボタンを離してください。

スプリットバーの位置を調節するには

スプリットバーボタンを押したままマウスを下に移動して、分割したい位置でマウスボタンを離してください。

ウィンドウの分割を解除するには

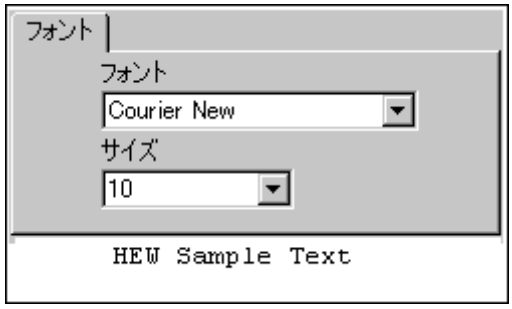
スプリットバーボタンをダブルクリックするか、スプリットバーボタンをウィンドウの一番上か一番下に移動してください。

4.9 エディタのフォントを変更する

HEW ではエディタのフォントを指定できます。ファイルの種類にかかわらず、エディタウィンドウはすべて同じフォントを使用します。

エディタのフォントを変更するには

1. [基本設定->表示の形式]を選択してください。[表示形式]ダイアログボックスを表示します。
2. ツリーの中の[Source]アイコンを選択してください。[表示形式]ダイアログボックスの右側の[フォント]タブが使用可能です。
3. [フォント]リストからフォントの種類を選択してください。
4. [サイズ]リストからフォントの大きさを選択してください。
5. フォントを変更するとそのフォントがどう見えるか下にサンプルテキストを表示します。
6. [OK]ボタンをクリックしてください。



4.10 シンタックスを色付けする

コードをより読みやすくするため、指定した文字列（キーワードなど）を異なる色で表示できます。例えば、C ソースコードのコメントを緑色で、C 言語の型（int など）を青色で表示できます。

色付け方法は、ファイルグループ単位で指定できます。例えば、C source file、C header file、Assembly source file、Assembly include file でそれぞれ異なった色付け方法を定義できます。

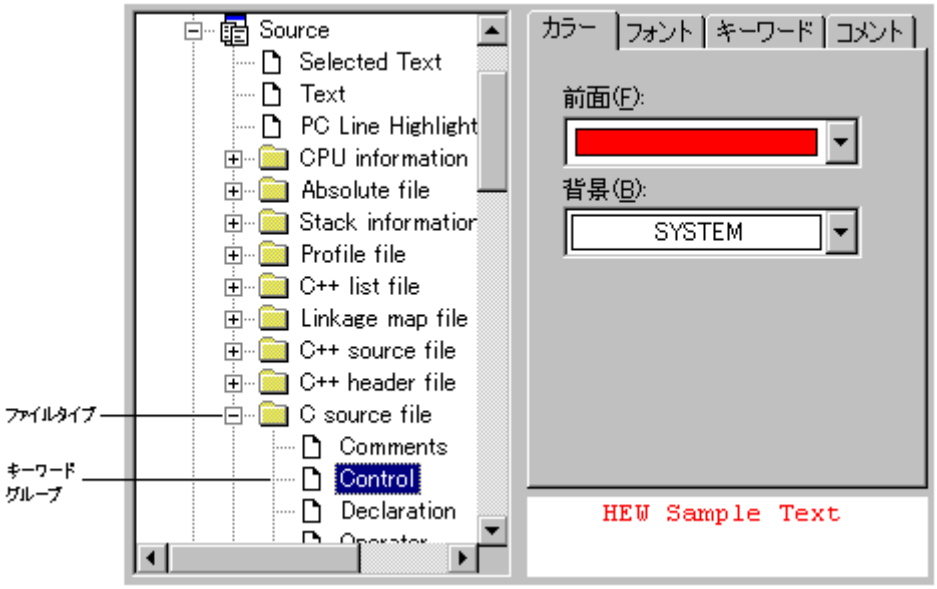
注：

新規ファイルを作成すると、デフォルトではファイルの拡張子がないためシンタックスの色付けは行いません。（エディタが自動的に新規ファイルに付ける名前には拡張子がありません）。新規ファイルにシンタックスの色付けをするには、上記の既存のファイルグループの拡張子を持つ名前でファイルを保存してください。ファイル拡張子の情報については「2.5 ファイル拡張子とファイルグループ」を参照してください。

4.10.1 色の変更

既存の色を変えるには

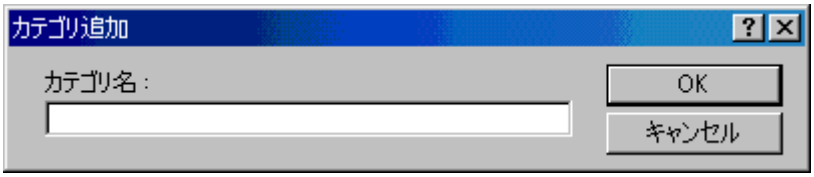
1. [基本設定->表示の形式]を選択してください。[表示形式]ダイアログボックスを表示します。
2. 色を変更したいアイテムを選択します。エディタの場合は、ダイアログボックスの左側のツリーにある"Source"を拡張してください。
3. シンタックスの色を編集したいファイルタイプを拡張してください。
4. 対象となるキーワードグループを選択してください。ダイアログボックスの右側のタブは、選択した項目によって変わります。
5. [カラー]タブを選択してください。
6. [前面]リストと[背景]リストの色を選択してください。[SYSTEM]を選択するとコントロールパネルで現在設定してある文字色と背景色になります。
7. [OK]ボタンをクリックしてください。



4.10.2 新規キーワードの作成

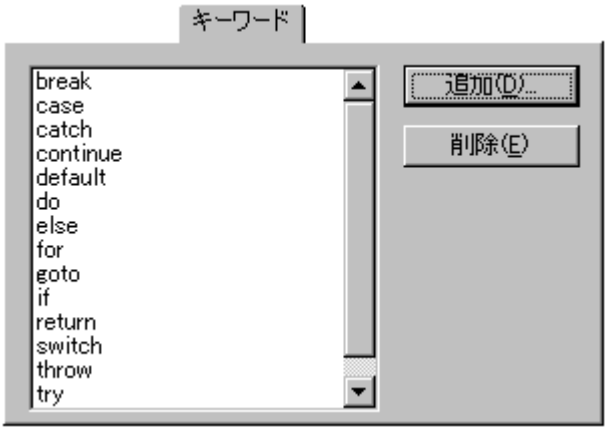
新規のキーワードグループを作成するには

1. [基本設定->表示の形式]を選択してください。[表示形式]ダイアログボックスを表示します。エディタの場合は、ダイアログボックスの左側のツリーにある[Source]を拡張してください。
2. キーワードグループを追加したいファイルタイプをツリーから選択してください。
3. ツリー下の[追加]ボタンをクリックすると[カテゴリ追加]ダイアログボックスを表示します。[カテゴリ名]フィールドにキーワードグループを入力し、[OK]ボタンをクリックすると新しいキーワードグループを追加します。キーワードグループ名を変更するには、キーワードグループを選択してツリー下の[変更]ボタンをクリックしてください。[カテゴリの変更]ダイアログボックスを表示します。[カテゴリ名]フィールドにキーワードグループを入力し、[OK]ボタンをクリックするとキーワードグループ名を変更します。キーワードグループを削除するにはキーワードグループを選択してツリー下の[削除]ボタンをクリックしてください。



新規のキーワードを作成するには

1. [基本設定->表示の形式]を選択してください。[表示形式]ダイアログボックスを表示します。
2. シンタックスの強調表示を変更したいアイテムを、ツリーの中のアイコンの下から選択してください。このアイテムはファイルタイプ（例：C 言語のソースファイル）および適切なキーワードグループ（例：識別子またはプリプロセッサ）となります。
3. [キーワード]タブを選択してください。



4. キーワードを追加するには[追加]ボタンをクリックしてください。[キーワード追加]ダイアログボックスを表示します。[キーワード]フィールドにキーワードを指定し[OK]ボタンをクリックしてダイアログボックスを閉じてください。キーワードを削除するにはキーワードを選択して[削除]ボタンをクリックしてください。

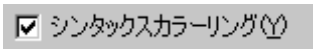
注：

[キーワード追加]ダイアログボックスの[キーワード]フィールドには半角英数字と半角下線、半角#のみ入力してください。特に、日本語文字やその他の文字を入力しないでください。

4.10.3 シンタックスの色付けを有効/無効にする

シンタックスの色付けを有効/無効にするには

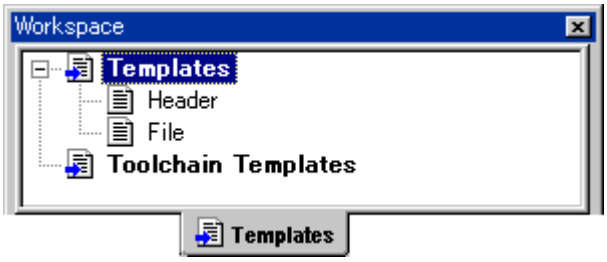
1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [エディタ]タブを選択してください。
3. [シンタックスカラーリング]チェックボックスをオンにすると有効になり、チェックしないと無効になります。設定後、[OK]ボタンをクリックしてください。



4.11 テンプレート

ソフトウェア開発時、同じテキストを繰り返し入力する場合がよくあります。例えば、関数定義、ループ、関数の機能のコメント欄などです。HEW では、現在アクティブなエディタウィンドウに、定型テキスト（テンプレート）を挿入できます。テンプレート設定後は、手作業で入力するかわりに定型テキストを簡単に挿入できるようになります。

テンプレート一覧を示します。この一覧はワークスペースウィンドウの[Templates]タブにあります。



HEW で追加した新しいテンプレートを[Templates]フォルダの下に表示します。[Toolchain Templates]フォルダには、現在のツールチェインによって HEW システムで使用するために提供される読み取り専用のテンプレートを表示します。

この一覧にあるテンプレートはドラッグアンドドロップしてエディタファイルに挿入できます。さらに、エディタからテキスト領域を[Templates]フォルダにドラッグアンドドロップして、簡単にテンプレートを作成することも可能です。

[Templates]フォルダ上で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

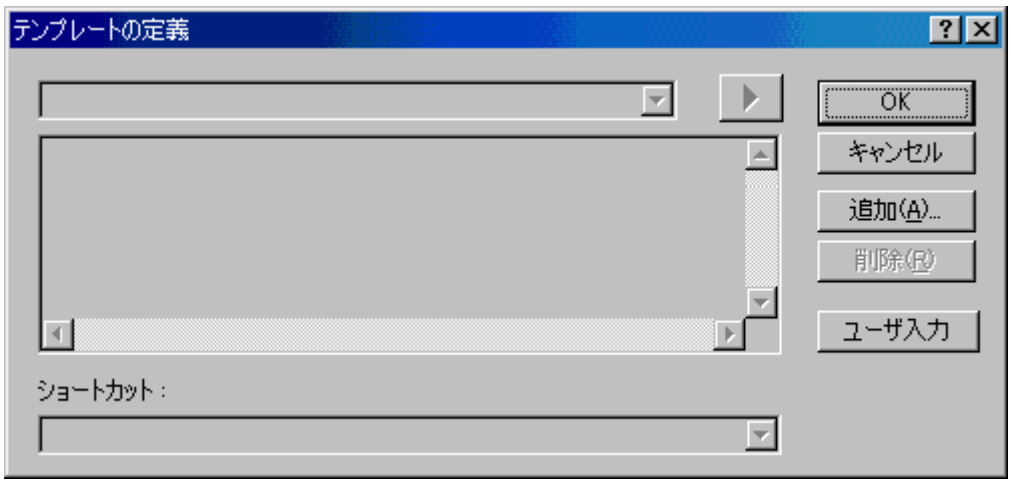
ポップアップメニュー項目	機能
追加	新しいテンプレートを追加します。
編集	現在選択しているテンプレートを編集します。
削除	現在選択しているテンプレートを削除します。

4.11.1 テンプレートを設定する

テンプレートを設定するには

1. 次のいずれかの操作を行って[テンプレートの定義]ダイアログボックスを表示してください。

- [テンプレートの定義]ツールバーボタン( )をクリックする。
- [編集->テンプレート->テンプレートの定義]を選択する。
- ワークスペースウィンドウの[Templates]タブの[Templates]フォルダ上で右クリックし、ポップアップメニューから[編集]を選択する。
- エディタウィンドウ内で右クリックし、ポップアップメニューから[テンプレート->テンプレートの定義]を選択する。



2. [追加]ボタンをクリックしてください。[テンプレートの追加]ダイアログボックスが表示され、選択したテンプレート名を尋ねます。この場合、独自のテンプレート名を選択してください。複製したテンプレート名のメッセージが表示され、テンプレートは追加されません。
3. 既存のテンプレートを変更したいときは、[テンプレート名]ドロップダウンメニューから変更するテンプレートを選択してください。
4. テンプレートテキストエリアにテキストを入力してください。他のエディタウィンドウからテキストをコピーし、”Ctrl+V”キーを押してこのダイアログボックスに貼り付けることができます。
5. テンプレートを設定するとき、次のプレースホルダを使うことができます。

メニューエントリ	プレースホルダ	説明
ファイルパス+ファイル名	$(FULLFILE)	ファイル名（フルパスを含む）
ファイル名	$(FILENAME)	ファイル名（パスを除き拡張子を含む）
ファイルリーフ	$(FILELEAF)	ファイル名（パスと拡張子を除く）
ワークスペース名	$(WORKSPNAME)	ワークスペース名
プロジェクト名	$(PROJECTNAME)	プロジェクト名
行番号	$(LINE)	テンプレートを挿入する最初の行番号
時間	$(TIME)	テンプレートが挿入される時間
日付, テキスト	$(DATE_TEXT)	現在の年月日をテキスト表示
日付, 日/月/年	$(DATE_DMY)	現在の日/月/年
日付, 月/日/年	$(DATE_MDY)	現在の月/日/年
日付, 年/月/日	$(DATE_YMD)	現在の年/月/日
ユーザ名	$(USER)	現在のユーザ
カーソル位置	$(CURSOR)	挿入カーソル：テンプレートを挿入した後カーソルをこの位置に設定

6. ‘$(CURSOR)’ を入力すると、テンプレートが挿入された後のカーソルはこの位置になります。‘$(CURSOR)’を設定しないと、テンプレートが挿入された後のカーソルはテンプレート最後の文字の後ろになります（通常の貼り付け操作と同じ）。
7. テンプレート用に確保されているショートカットキーは 10 個あります。ショートカットキーを使用する場合、[テンプレートの編集]ダイアログボックス下部にあるドロップダウンリストで選択してください。"Ctrl+0"～"Ctrl+9"の中から選択できます。

ユーザ入力

テンプレートを定義する際、ユーザ入力フィールドも定義できます。以下のプレースホルダを使って指定します。

$(USERINPUT<n:1-10>|”<some text>”)

'n'はユーザ入力識別子を表す数字です。これらのプレースホルダは手動で追加することも可能ですが、[テンプレートの定義]ダイアログボックスの[ユーザ入力]ボタンを使うとプレースホルダを自動的に追加できます。テンプレートをファイルに挿入するとダイアログボックスが表示され、ユーザは各フィールドにカスタムのテキストを入力できます。その後、このテキストをプレースホルダの替わりに挿入します。ユーザはこのようなユーザ入力フィールドを 10 個定義できます。

4.11.2 テンプレートを削除する

テンプレートを削除するには

1. 次のいずれかの操作を行って[テンプレートの定義]ダイアログボックスを表示してください。

- [テンプレートの定義]ツールバーボタン （)をクリックする。
- [編集->テンプレート->テンプレートの定義]を選択する。
- ワークスペースウィンドウの[Templates]タブの[Templates]フォルダ上で右クリックし、ポップアップメニューから[編集]を選択する。
- エディタウィンドウ内で右クリックし、ポップアップメニューから[テンプレート->テンプレートの定義]を選択する。

2. [テンプレート名]ドロップダウンリストから削除したいテンプレート名を選択し、[削除]ボタンをクリックしてください。
3. 確認ダイアログボックスが表示された場合は、[はい]ボタンをクリックしてください。

4. [OK]ボタンをクリックしてください。

テンプレートを削除するには、他にも方法があります。これを以下に説明します。

ワークスペースウィンドウの[Templates]タブで選択したテンプレートを削除するには

1. ワークスペースウィンドウの[Projects]タブで削除したいファイルを選択してください。複数のテンプレートを選択するときは”Shift”キーまたは”Ctrl”キーを押してください。
2. ワークスペースウィンドウの[Templates]タブの[Templates]フォルダ上で右クリックし、ポップアップメニューから[削除]を選択する。

4.11.3 テンプレートを挿入する

テンプレートを挿入するには

1. 次のいずれかの操作を行って[テンプレートの定義]ダイアログボックスを表示してください。

- ツールバーでテンプレートを選択し[テンプレートの挿入]ツールバーボタン( )をクリックする。
- [編集->テンプレート->テンプレートの挿入]を選択する。
- エディタウィンドウ内で右クリックし、ポップアップメニューから[テンプレート->テンプレートの挿入]を選択する。

2. [テンプレート名]ドロップダウンリストで挿入するテンプレートを選択し、OK ボタンをクリックします。選択したテンプレートが現在のエディタウィンドウに追加されます。

ALT キーとテンプレート番号を押すことでテンプレートを挿入する方法もあります(例えばテンプレート 4 を挿入するには ALT+4)。これらのショートカットは[テンプレートの定義]ダイアログボックスで定義できます。ダイアログボックスの下部のドロップダウンリストで選択可能です。

4.12 括弧の組み合わせ

複雑なソースコードは扱いにくいことがあります。特に、C 言語のブロックが互いに深いネスト構造になっている場合や、if 文で複雑な論理文が表現されている場合などです。HEW では、そのような場合のために、括弧の種類 { }, ( ) , [ ]ごとに括弧の間のテキストを強調表示できます。

括弧の組み合わせを見つけるには

1. 括弧の始めを強調表示するか、カーソルを括弧の前に置いてください。
2. 次のいずれかを行ってください。

- [括弧の呼応]ツールバーボタン （ ） をクリックする。
- “Ctrl+Shift+M”キーを押す。
- [編集->括弧の呼応]を選択する。
- ポップアップメニューから[括弧の呼応]を選択する。

ファイル全体の構造をチェックするために、カーソルをファイルの始めに置いて、括弧の組み合わせの操作を繰り返し行ってください。組み合わせがなくなるまで、括弧の組み合わせごとに次々と強調表示します。

4.13 ファイルを読み取り専用属性にする

HEW ではファイルを読み取り専用属性にできます。読み取り専用属性のファイルを保存するときは、別名で保存するかどうかを確認するダイアログボックスが表示されます。

ファイルを読み取り専用属性にするには

1. 次のいずれかを行ってください。

- ワークスペースウィンドウの[Projects]タブで、ファイルを右クリックしポップアップメニューを表示してください。複数のファイルを選択できます。
- エディタウィンドウを右クリックしポップアップメニューを表示してください。

2. [プロパティ]を選択すると[プロパティ]ダイアログボックスが表示されます。
3. [読み取り専用]チェックボックスをオンにしてください。複数のファイルを選択し、読み取り専用および書き込み可能なファイルが混在する場合は、チェックボックスはグレー（中間状態）です。チェックボックスのデフォルトはオフです。
4. エディタウィンドウのタイトルバーに「読み取り専用」を表示します。（例：filename.c - 読み取り専用）

注：

エディタウィンドウでファイルを開いている状態で、Windows® Explorer からそのファイルの読み取り専用属性のオンオフを切り替えた場合、エディタウィンドウのタイトルバーの表示内容は、更新されないためファイルの属性と一致しません。編集するか、ファイルを閉じてから再度開くまでエディタウィンドウのタイトルバーの表示内容を更新しません。

4.14 デバッグ時にファイルの編集を抑止する

デバッグ時にファイルの編集を抑止したい場合があります。

HEW ではデバッギングプラットフォームに接続しデバッグしている時にファイルの編集を抑止する機能があります。編集を抑止するファイルは HEW エディタで開いているすべてのファイルです。

デバッグ時にファイルの編集を抑止するには

1. [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を表示します。
2. [エディタ]タブを選択します。
3. [ターゲット接続中はファイルの編集を許可しない]チェックボックスをオンにすると、デバッギングプラットフォームに接続している時に、ファイルの編集を抑止することができます。デバッギングプラットフォームに接続していないときはファイルの編集は可能です。このチェックボックスのデフォルトはオフです。
4. [OK]ボタンをクリックします。

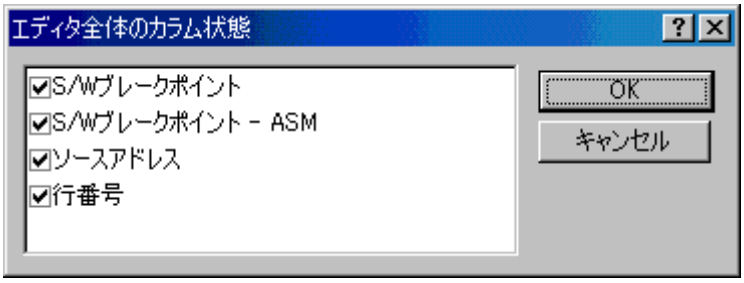
4.15 エディタカラムの管理

エディタのカラムは、デバッガの機能に応じて追加されます。また、カラムの表示、非表示を選択することが可能です。

すべての HEW エディタでカラムをオフにするには

1. 次のいずれかの操作を行って[エディタ全体のカラム状態]ダイアログボックスを表示してください。

- [編集->表示カラムの設定]を選択します。
- エディタウィンドウ内で右クリックし、ポップアップメニューから[表示カラムの設定]を選択する。



2. チェックボックスは、そのカラムが有効か無効かを示します。チェックしている場合は有効です。チェックボックスがグレー表示の場合、一部のエディタではカラムが有効で、別のエディタでは無効であることを意味します。
3. [OK]ボタンをクリックしてください。

1 つの HEW エディタでカラムをオフにするには

1. 削除したいカラムのあるエディタウィンドウを開きます。
2. 次のいずれかを行ってください。

- HEW エディタのヘッダ内で右クリックするとポップアップメニューが表示されます。カラムが利用できる場合は、エントリの横にはチェックマークがあります。エントリをクリックすると、カラムの表示、非表示を切り替えます。
- エディタウィンドウ内で右クリックし[カラム]を選択します。カスケードされたメニュー項目を表示します。各カラムがこのポップアップメニューに表示されます。カラムが有効である場合、名前の横にチェックマークがあります。エントリをクリックすると、カラムの表示、非表示を切り替えます。

HEW エディタのヘッダが表示されている場合は、カラムの幅をマウスのドラッグにより調節できます。

4.16　カラムヘッダの表示/非表示を切り替える

エディタウィンドウはカラムヘッダがあります。カラムヘッダ自体の表示、非表示の切り替えが可能です。

カラムヘッダの表示、非表示を切り替えるには

1. エディタウィンドウでソースコードまたは逆アセンブリを表示している領域内を右クリックしてください。
2. 表示されるポップアップメニューから[カラムヘッダの表示/非表示]を選択してください。

4.17　エディタ内からファイルを開く

エディタウィンドウに表示しているファイル名を元にそのファイルをエディタウィンドウで開くことができます。

エディタウィンドウに表示されているファイルを開くには

1. エディタウィンドウの[ソース]領域で表示したいファイル名の上を右クリックしてください。

2. ポップアップメニューから[開く 'ファイル名']を選択してください。

4.18　ツールチップウォッチ

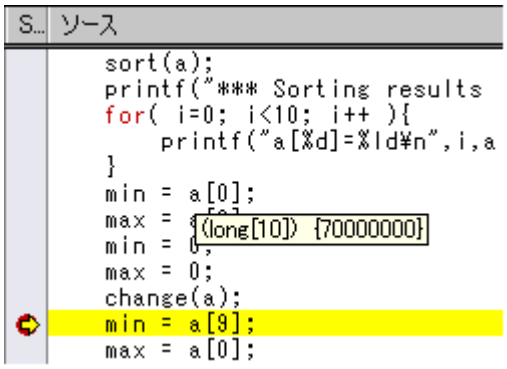
プログラムに定義された変数の値を参照するための機能です。値を参照したい変数を含むプログラムをエディタウィンドウまたは逆アセンブリウィンドウ（ソースモード）に表示し、変数名の上にマウスポインタを静止させると、ツールチップとしてポップアップウィンドウにウォッチ情報を表示します。

ツールチップウォッチ機能を使用するには

1. [基本設定->オプション]メニュー項目をクリックしてください。[オプション]ダイアログボックスを表示します。
2. [エディタ]タブを選択してください。
3. [ツールチップウォッチ]チェックボックスをオンにしてください。
4. [OK]ボタンをクリックしてください。

エディタウィンドウまたは逆アセンブリウィンドウ（ソースモード）にツールチップウォッチを表示するには

1. 確認したい変数を表示しているエディタウィンドウまたは逆アセンブリウィンドウ（ソースモード）を開きます。
2. 確認したい変数名の上にマウスのカーソルを静止させます。ツールチップとしてポップアップウィンドウに変数のウォッチ情報を表示します。



4.19 式の評価

[式の評価]ダイアログボックスを表示します。例えば、205*2 のような数式を入力して、その結果をデバッギングプラットフォームが現在サポートしているすべての基数で表示します。

式評価するには

1. [編集->式の評価]を選択してください。
2. [式の評価]ダイアログボックスを表示します。
3. 式を入力後、[評価]ボタンをクリックしてください。

簡単な式、そして括弧、基数とシンボルを持つ複雑な式を評価できます。

すべての演算子は同じ優先順位を持っていますが、括弧は評価の順序を変更できます。演算子は C/C++と同じ意味を持っています。式もまた数値が要求されるコマンドで使うことができます。

結果は、デバッギングプラットフォームが現在サポートしているすべての基数で表示します。

有効な演算子:

加算 (+)	減算 (-)	乗算 (*)	除算 (/)
論理 AND (&&) ***1**	論理 OR (||) ***1**	論理 NOT (!) ***1**	等しい (==)
ビットごとの AND (&)	ビットごとの OR (|)	ビットごとの NOT (~)	等しくない (!=)
左算術シフト (<<)	右算術シフト (>>)	より小さい (<)	より大きい (>)
剰余 (%) ***1**	ビットごとの排他的 OR (^)	以下 (<=)	以上 (>=)

***1.** 機能のサポートは、デバッギングプラットフォームに依存します。

レジスタ名**:**

・**SuperH RISC engine/H8/R8C(E7/E8)**ファミリの場合

レジスタ名を使うことができますが、先頭に"#"文字を付けなければいけません。例えば、(#pc + 205)*2 のような数式を入力して、その結果を現在サポートしているすべての基数で表示します。

・**M32C/M32R/M16C/R8C(E7/E8** 以外**)**ファミリの場合

レジスタ名を使うことができますが、先頭に"%"文字を付けなければいけません。例えば、(%pc + 205)*2 のような数式を入力して、その結果を現在サポートしているすべての基数で表示します。

文字定数**:**

文字定数として、シングルクォーテーション(')で囲まれた文字が使用できます。例えば、'A'等です。これらは、ASCII コードに変換され、1 バイトの即値として使用されます。

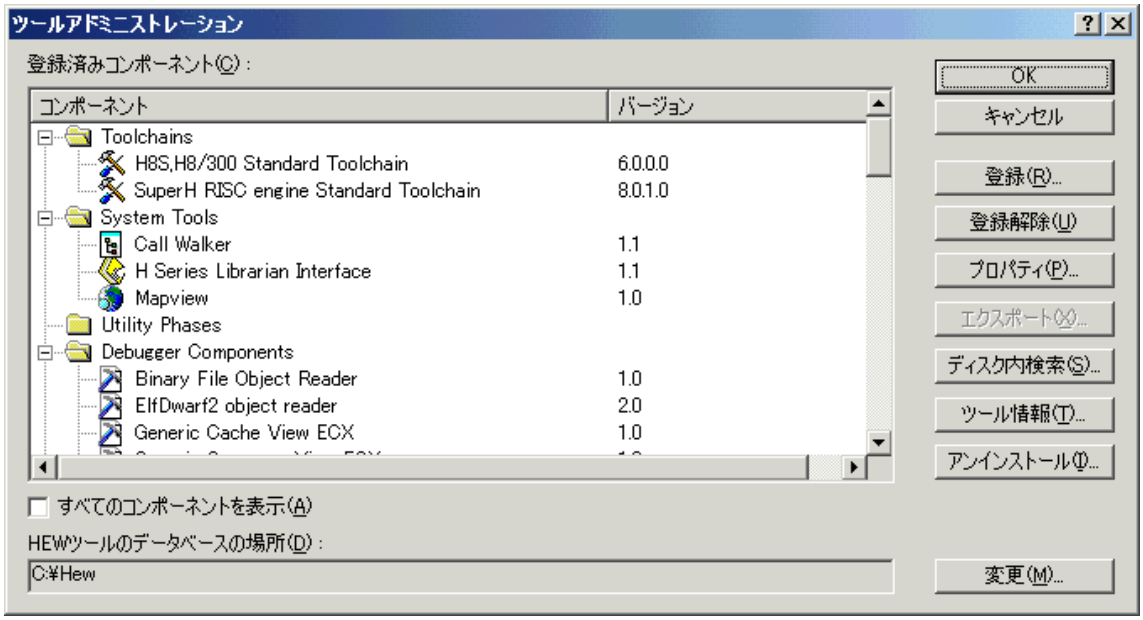
文字列リテラル**:**

文字列リテラルとして、ダブルクォーテーション(")で囲まれた文字列が使用できます。例えば、"abc"等です。

5. ツール管理

[ツールアドミニストレーション]ダイアログボックスで、High-performance Embedded Workshop (HEW) で使うツールを管理します。

このダイアログボックスは[ツール->アドミニストレーション]で開きます。すべてのワークスペースが閉じているときは変更可能で、ワークスペースが開いているときは参照のみ可能です。



HEW には 5 種類の標準ツールがあります。

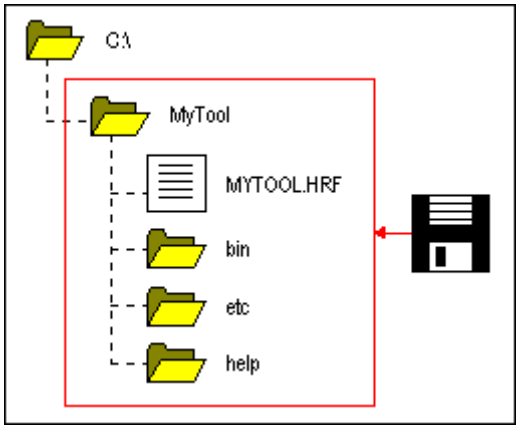
[Toolchains]	一連のビルドフェーズ（例:コンパイラ、アセンブラ、リンケージエディタ、ライブラリアン）。ビルド機能を実現します。
[System Tools]	[ツール]メニューからを選択することのできるアプリケーションです（.EXE）。ツールチェインをサポートする追加のアプリケーションです（例: 日立デバッギングインタフェースなどの外部デバッガまたは対話式グラフィカルライブラリアン）。
[Utility Phases]	特定のビルド機能をサポートする、あらかじめ用意されたビルドフェーズです（例:ソースコードの複雑度解析、ソースコードの行カウントなど）。特定のツールチェインに依存しない追加のビルド機能です。
[Debugger Components]	特定のデバッガ機能をサポートするツールです（例: デバッギングプラットフォーム、オブジェクトリーダなど）。
[Extension Components]	HEW システムのある領域における特定のキー機能をサポートするツールです。これらのツールはインストールすると、必ず登録します（例: HEW ビルダ、デバッガ、フラッシュサポート）。

5.1 ツールの位置

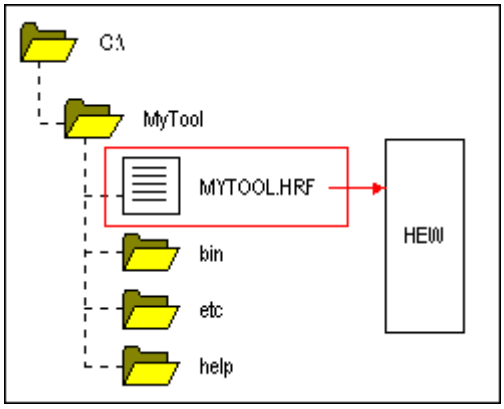
HEW では、新しいツールがインストールされるたびに HEW との連動に必要なツールの位置を自動的に保持します。インストール後、HEW はそのツールに関する情報（位置を含む）を保持します。これを登録と呼びます。初期登録は自動で行いますが、開発の途中で、プロジェクトのツールをより効率良く利用するためにユーザ自身でツールを登録することが必要になることがあります。

5.2 HEW 登録ファイル

HEW と互換性のあるツール（ツールチェイン、システムツール、またはユーティリティフェーズ）をインストールすると、拡張子**.HRF** のファイルもインストールします（以下の図を参照）。.HRF ファイルには HEW への登録に必要な情報を記述しています。



登録するには、そのツールの.HRF ファイルを[ツールアドミニストレーション]ダイアログボックスにロードします（以下の図を参照）。



HEW でツールを使うには、まず登録が必要です。[ツールアドミニストレーション]ダイアログボックスは現在登録されているツールを表示します。このダイアログボックスを開くには、ワークスペースがすべて閉じていることを確認して[ツール->アドミニストレーション]を選択してください。ワークスペースを開いた状態で[ツールアドミニストレーション]ダイアログ

ボックスをアクセスすると、ダイアログが開きますが変更はできません。HEW がデフォルトでインストールされると、新しいツールを自動的に登録します。

HEW はツールデータベースファイルに、ツール情報を格納します。デフォルトでは、このファイルを HEW アプリケーションディレクトリに作成します。しかし、ネットワーク環境で作業を行っている場合は、このディレクトリは他の場所に設定されることがあります。ツールディレクトリの場所は変更が可能です。

ツールの場所を変更するには

1. [ツール->アドミニストレーション]を選択してください。
2. [HEW ツールのデータベースの場所]の[変更]ボタンをクリックしてください。
3. 新しいツールがあるディレクトリのルートディレクトリを選択し、[OK]ボタンをクリックしてください。
4. ディレクトリが切り替わり、ツールの場所が新しいディレクトリに移ります。この場所にある新しいツールはスキャンする必要があります。スキャンには、[ディスク内検索]ボタンまたは[登録]ボタンを使用します。

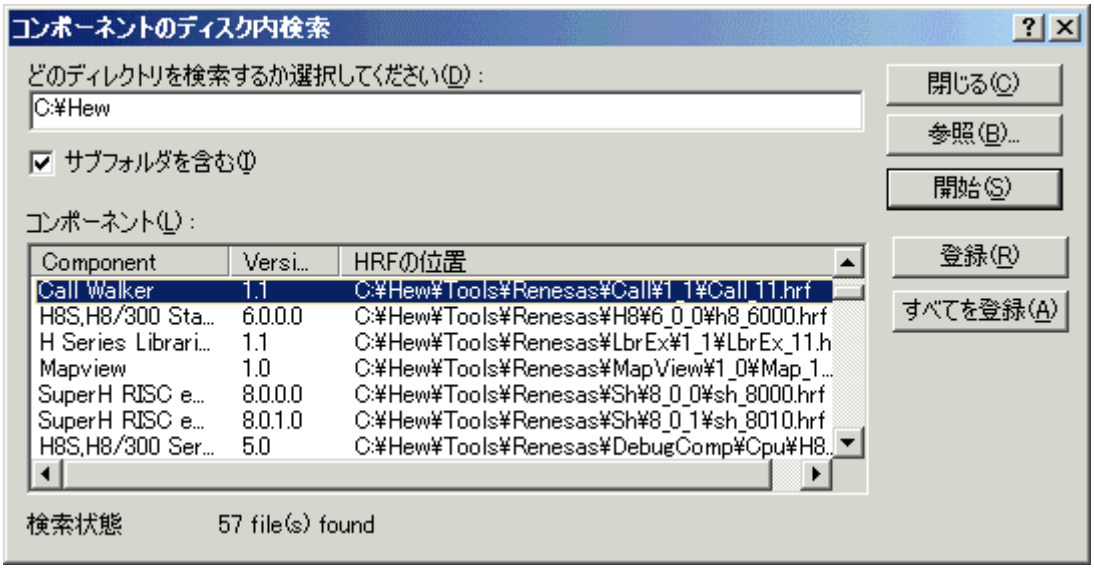
5.3 ツールを登録する

HEW は起動後にインストールしたすべてのツールを自動的に登録します。しかし、ときには、ユーザがツールを登録する必要があります。

ドライブを検索して HEW に互換性のあるツールを見つけることは、HEW のツールインストール情報が削除されたり破壊されたりしたときなどに有益です。なぜなら、ツール情報を再びすぐに作成することができるからです。

ツールを検索するには

1. [ツールアドミニストレーション]ダイアログボックスの[ディスク内検索]ボタンをクリックすると[コンポーネントのディスク内検索]ダイアログボックスを表示します。
2. [どのディレクトリを検索するか選択してください]フィールドに、検索するディレクトリを入力してください。または、[参照]ボタンをクリックしてディレクトリを選択してください。
3. [サブフォルダを含む]チェックボックスをオンにすると、指定したディレクトリとその下のディレクトリをすべて検索します。
4. [開始]ボタンをクリックすると検索を始めます。検索中、[開始]ボタンは[中断]ボタンに変わります。検索を途中で止めるには[中断]ボタンをクリックしてください。
5. [コンポーネント]リストに検索結果を表示します。個別にツールを登録するにはそのツールを選んで[登録]ボタンをクリックしてください。すべてのツールを登録するには[すべてを登録]ボタンをクリックしてください。
6. [閉じる]ボタンをクリックしてください。



ツールを 1 つ登録するには

1. [登録]ボタンをクリックすると標準のファイルを開くダイアログボックスが開きます。フィルタが[HEW Registration Files (*.hrf)]に設定されています。
2. 登録するファイルの.HRF ファイルをアクセスして選択し、[選択]ボタンをクリックしてください。
3. 選んだツールに関する情報を示すダイアログボックスを表示します。ツールを登録するには[登録]ボタンをクリックしてください。登録しない場合は[閉じる]ボタンをクリックしてください。

HEW 登録ファイルはツールがインストールされたルートディレクトリにあります。

5.4 ツールの登録を取り消す

新しいプロジェクト作成時に、登録された互換性のあるすべてのシステムツールがツールメニューに追加されます。ときにはユーザにとっては、これによって効率が下がり、使いにくいかもしれません。そのようなときは、登録を取り消すことができます。[ツールアドミニストレーション]ダイアログボックスでツールを選択し[登録解除]ボタンをクリックしてください。確認のダイアログボックスを表示します。登録を取り消す場合は[はい]をクリックしてください。

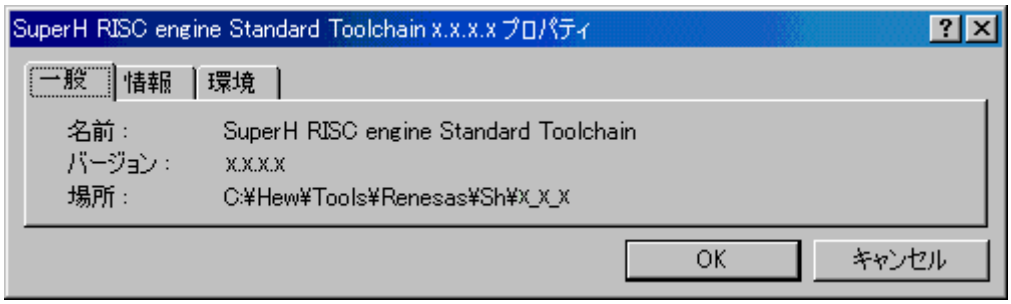
注：

ツールの登録を取り消しても、ハードディスクからツールがなくなることはありません。単に、HEW に格納されているそのツールに関する情報を削除するだけです（HEW から切断します）。この動作はいつでも元に戻すことができ、ツールを再登録できます（「5.3 ツールを登録する」参照）。ハードディスクからツールを削除（アンインストール）したいときは「5.6 ツールのアンインストール」を参照してください。

5.5 ツールのプロパティの参照と編集

ツールに関する情報を参照するには、ツールを[登録済みコンポーネント]リストから選択し[プロパティ]ボタンをクリックしてください。

[一般]タブでは名前、バージョン、位置を示します。



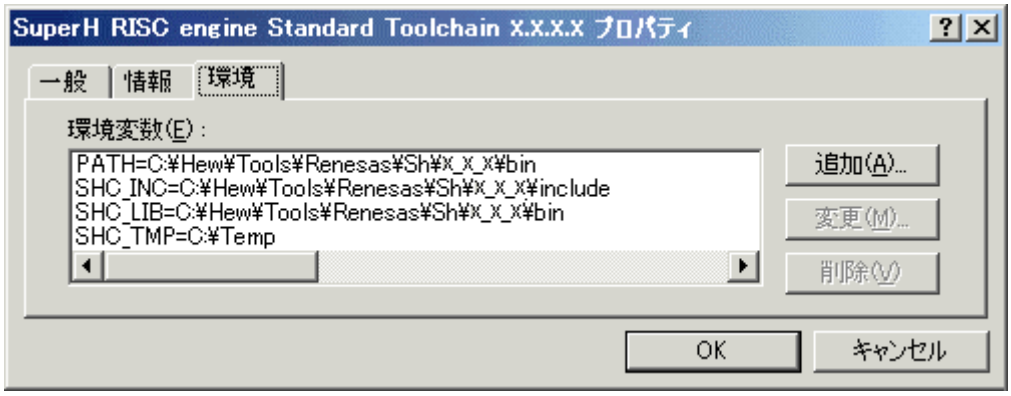
ツールについての情報を参照するには

[情報]タブをクリックしてください。例えば、著作権、履歴などを表示します。

ツールの互換性に問題がある場合、この[情報]エディットフィールドで問題を報告することがあります。

ツールの環境設定を参照、編集するには

[環境]タブを選択してください。このタブは、ツールチェインの環境を変更するときに最も一般的に使用します。



新しい環境変数を追加するには

1. [追加]ボタンをクリックしてください。[環境変数]ダイアログボックスを表示します。
2. [変数]フィールドに変数名を入力してください。
3. [値]フィールドに変数の値を入力してください。
4. [OK]ボタンをクリックすると、[環境]タブに新しい変数が加わります。

環境変数を変更するには

1. [環境]タブで変更する環境変数を選択してください。
2. [変更]ボタンをクリックしてください。[環境変数]ダイアログボックスを表示します。
3. [変数]フィールドと[値]フィールドを必要に応じて変更してください。
4. [OK]ボタンをクリックすると、環境変数を変更します。

環境変数を削除するには

1. [環境]タブから削除する環境変数を選択してください。
2. [削除]ボタンをクリックしてください。

注：

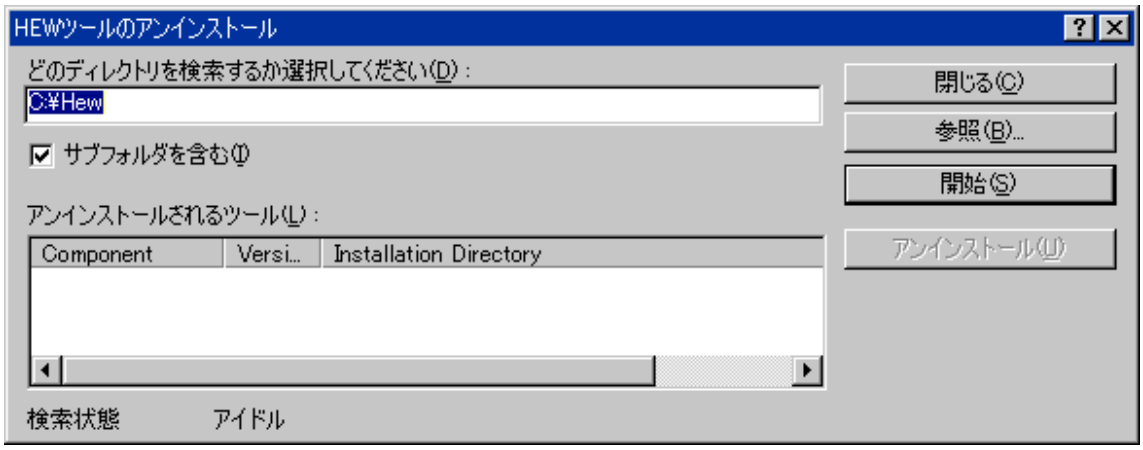
環境変数の値を柔軟に指定できるようにするため、プレースホルダポップアップメニューがあります。プレースホルダの詳細は、「リファレンス 5. プレースホルダ」を参照してください。

5.6 ツールのアンインストール

HEW には登録されていないツールを**ハードディスクから削除する**ための、アンインストール方法があります。

ツールをアンインストールするには

1. [ツール->アドミニストレーション]を選択してください。
2. [アンインストール]ボタンをクリックしてください。[HEW ツールのアンインストール]ダイアログボックスを表示します。

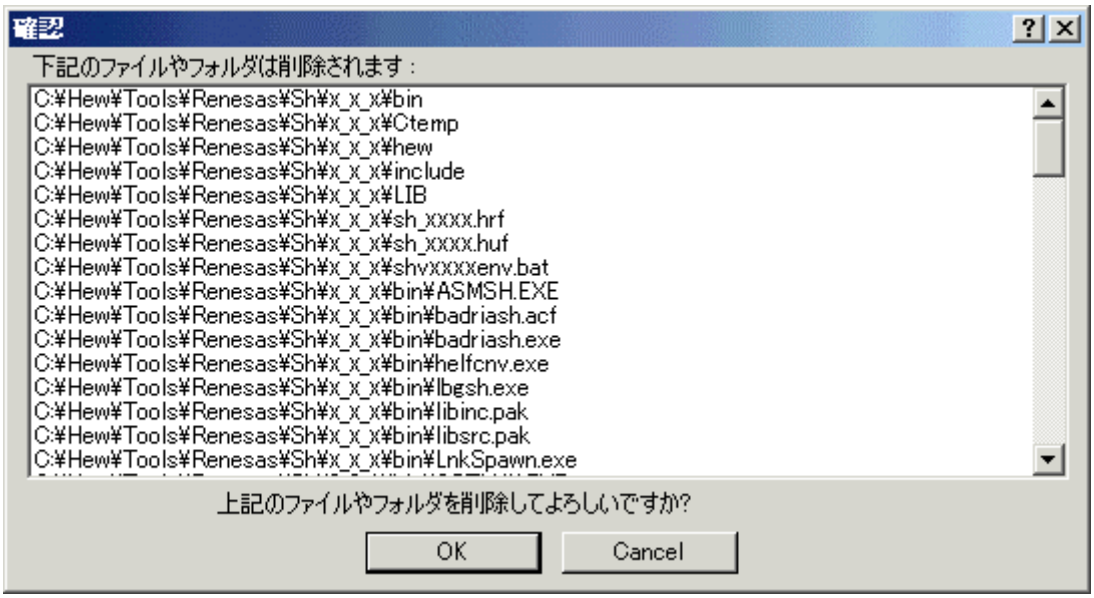


3. 一番上のフィールドに検索するディレクトリを入力するか、[参照]ボタンをクリックしてブラウズしてください。
4. [サブフォルダを含む]チェックボックスをオンにすると、指定したディレクトリの下のディレクトリをすべて検索します。
5. [開始]ボタンをクリックすると検索を始めます。検索中、[開始]ボタンは[中断]ボタンに変わります。[中断]ボタンをクリックすると、検索を中止します。
6. 検索結果を[アンインストールされるツール]リストに表示します。ツールを選んで[アンインストール]ボタンをクリックするとツールをアンインストールします。
7. [閉じる]ボタンをクリックしてダイアログボックスを終了してください。

現在 HEW に登録されていないツールのみアンインストールできます。登録されているツールをアンインストールしようとすると、最初にツールを登録解除するよう尋ねられます。

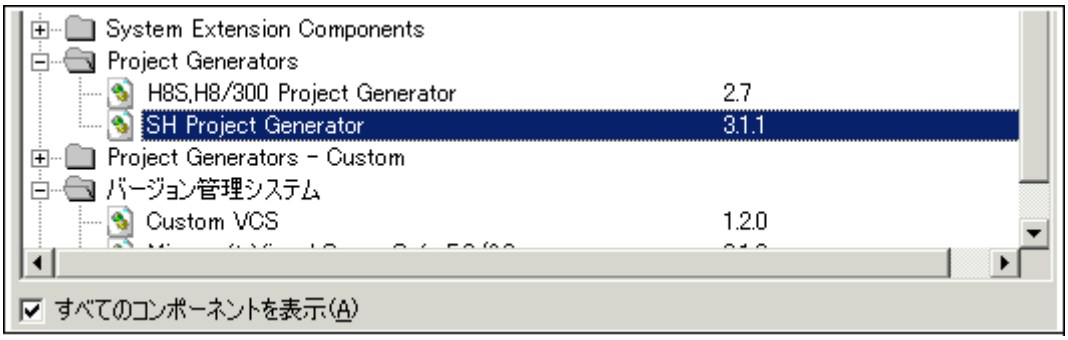
このようなときは、[ツール->アドミニストレーション]で HEW の[ツールアドミニストレーション]ダイアログボックスに戻り、ツールの登録を取り消してから、アンインストールを再実行してください。

HEW から登録をはずしたツールを選択し[登録解除]ボタンをクリックすると、[確認]ダイアログボックスを表示します。[確認]ダイアログボックスには削除されるすべてのファイルやフォルダを表示します。これらのファイルやフォルダを削除してよいことを確認して[OK]ボタンをクリックしてください。アンインストールを中止するときは、[Cancel]ボタンをクリックしてください。



5.7 テクニカルサポートについて

[ツールアドミニストレーション]ダイアログボックスでは、システムツールに関する情報を表示できます。これらのツールは HEW の一部であり、手動で登録または登録取り消しすることができません。[ツールアドミニストレーション]ダイアログボックスの [すべてのコンポーネントを表示]チェックボックスをオンにすると、隠れていたコンポーネントフォルダを表示します（下の図を参照）。



テクニカルサポートを受ける際、ツールに関する詳細をお尋ねすることがあります。そのときには、そのツールのフォルダを開き、ツールを選択し、[プロパティ] ボタンをクリックしてください。ここで表示されるダイアログボックスはこの章の前半で説明したものと同じように動作します。ただし、[環境]タブはありません。

HEW にはまた、登録されたツールに関する情報をファイルに出力する機能があるので、HEW システム全体の情報を得ることができます。もし HEW で問題がありましたら、テクニカルサポート担当宛にこの情報を送付してください。

ツール情報を出力するには

1. [ツール->アドミニストレーション]を選択してください。
2. [ツール情報]ボタンをクリックすると、[ツール情報ファイルの保存]ダイアログボックスを表示します。
3. ファイルの場所を選択し、[保存]ボタンをクリックしてください。
4. HEW に現在登録されているツールの情報をファイル出力します。

また、問題のあるコンポーネントは、[ツールアドミニストレーション]ダイアログボックスで確認できます。

1 つは登録済みコンポーネントが見つからない場合で、次のアイコンを表示します。

[Component not found]アイコン

もう 1 つは、コンポーネントは見つかったが、古いバージョンであったり他の依存するコンポーネントが使用不可であったりして使用できない場合で、次のアイコンを表示します。

[Incompatible component found]アイコン

注：

ツールにいずれかのエラーがある場合、以下の方法でさらに情報を得ることができます。

ツールエラーのフィードバックを取得するには

1. [ツール->アドミニストレーション]を選択してください。
2. 問題のあるツールをリストから選択してください。
3. [プロパティ]ボタンをクリックしてください。
4. [情報]タブを選択し、エディットフィールドを下までスクロールしてください。
5. この領域に問題の原因を表示します。

5.8 オンデマンドのコンポーネント

HEW バージョン 3.0 以降にはオンデマンドのコンポーネントという概念があります。これらのコンポーネントはアプリケーションやデバッガコンポーネントが自動的にロードするわけではありません。プロジェクトを生成する過程でユーザがロードするものです。

オンデマンドのコンポーネントをロードまたはアンロードするには

1. [プロジェクト->コンポーネント]を選択してください。
2. [コンポーネントギャラリー]ダイアログボックスを表示します。
3. ロードしたいコンポーネントを選択して[ロード]ボタンをクリックしてください。コンポーネントのイメージはロードされた状態に変わります。
4. コンポーネントをアンロードしたい場合は、コンポーネントを選択して[アンロード]ボタンをクリックしてください。コンポーネントのイメージはアンロードの状態に変わります。
5. [OK]ボタンをクリックし、変更を確認してください。

注：

ユーザのワークスペースにある各プロジェクトには、ロード／アンロードされた異なるコンポーネントを格納できます。複数のプロジェクトがある場合は、[Multiple projects]および[All projects]を使うことで、複数のプロジェクトにわたってコンポーネントのロード状態を変更できます。コンポーネントがあるプロジェクトでロードされ、他のプロジェクトではロードされない状態をユーザが選択した場合は、[中間状態]アイコンを表示します。

5.9 カスタムプロジェクトタイプ

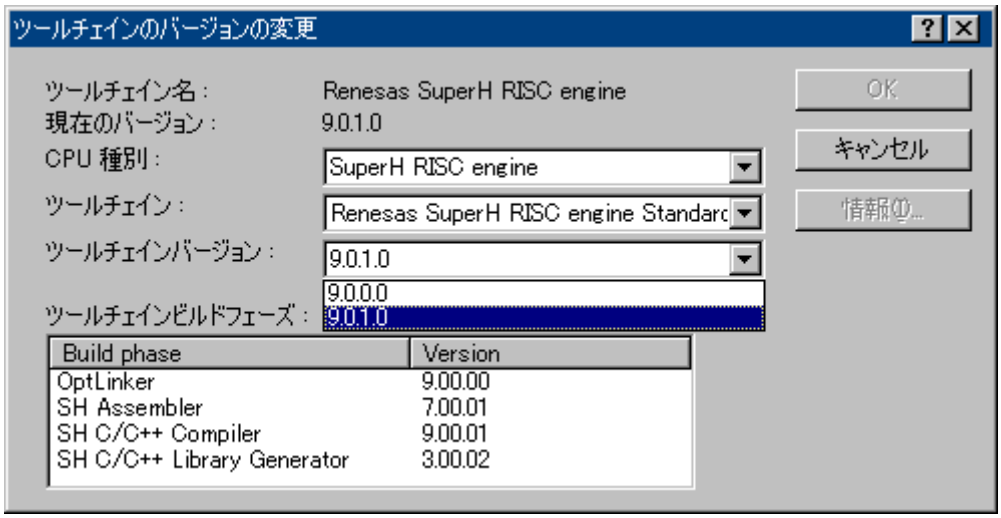
HEW の[プロジェクト->プロジェクトタイプの作成]メニューで、現在のプロジェクトの設定を利用してプロジェクトのテンプレートを作成できます。このテンプレートをカスタムプロジェクトジェネレータと呼びます。新しいプロジェクトタイプの名称とプロジェクト生成ウィザードのスタイルを指定できます。作成したカスタムプロジェクトジェネレータは、[ツールアドミニストレーション] ダイアログボックスで表示できます。カスタムプロジェクトジェネレータを他のユーザが使用するには、[ツールアドミニストレーション] ダイアログボックスで該当するカスタムプロジェクトジェネレータを選択して、[エクスポート] ボタンをクリックしてください。カスタムプロジェクトジェネレータの実行環境がインストール可能な実行ファイルにまとまります。このファイルを目的とする他のマシンで実行すると、カスタムプロジェクトジェネレータをインストールします。

一度カスタムプロジェクトタイプを作成すると、その後[新規プロジェクトワークスペース]ダイアログボックスに表示します。これを使用してプロジェクトの複製を作成できます。

複製したプロジェクトを HEW システムからエクスポートし、そのプロジェクトを使用する他のユーザに渡すことができます。

注：

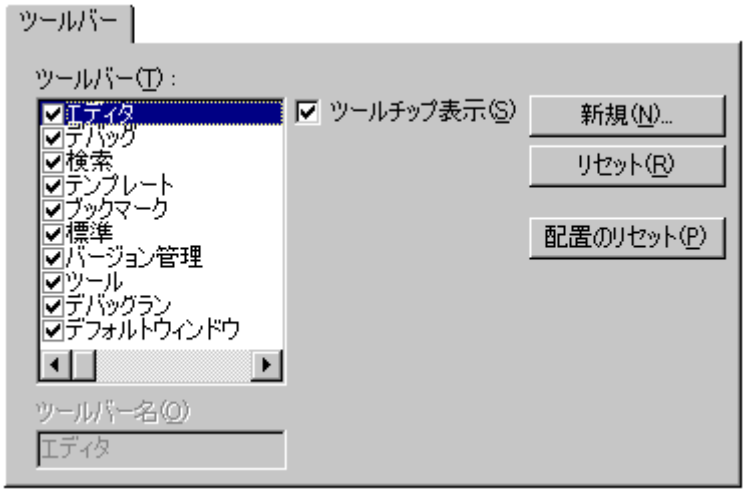
[プロジェクト->プロジェクトタイプの作成]により使用中のプロジェクトをもとにしたプロジェクトのテンプレートを作成できます。このテンプレートにはツールチェインのバージョン情報も含まれます。ご使用の HEW システムでツールチェインをバージョンアップした後に、このテンプレートを使ってプロジェクトを作成した場合、作成したプロジェクトのツールチェインバージョンが使用環境と一致しているかご確認ください。アップデート可能なツールチェインが登録されている場合、[ツール->ツールチェインバージョンを変更]により表示する[ツールチェインのバージョンの変更]ダイアログボックスでツールチェインバージョンを変更できます。



6. 環境のカスタマイズ

6.1 ツールバーのカスタマイズ

HEW ではツールバーを標準で提供します。また、[カスタマイズ]ダイアログボックスを使用して、新しいツールバーを作成できます。



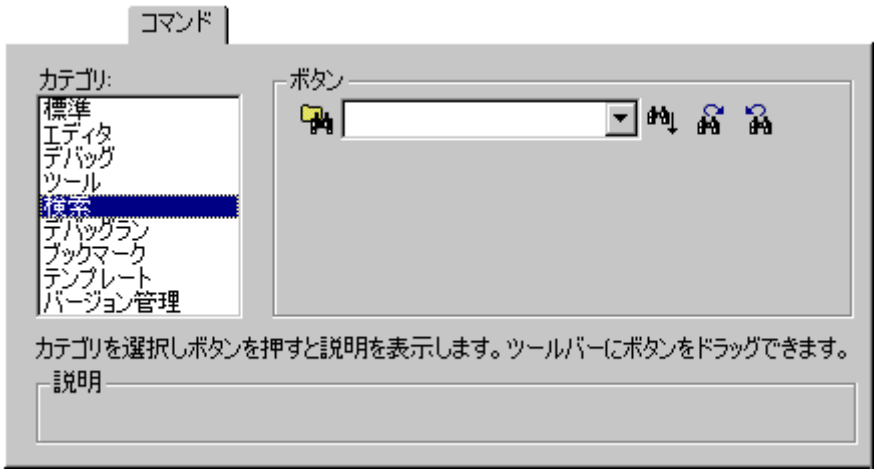
新しいツールバーを作成するには

1. [基本設定->カスタマイズ]を選択してください。[カスタマイズ]ダイアログボックスを表示します。[ツールバー]タブを選択してください。
2. [新規]ボタンをクリックしてください。[新規ツールバー]ダイアログボックスを表示します。
3. [ツールバー名]フィールドに新しいツールバー名を入力してください。
4. [OK]ボタンをクリックすると新しいツールバーを作成します。

作成した新しいツールバーにはボタンがなく、位置を固定せずに表示します。

ツールバーにボタンを追加するには

1. [基本設定->カスタマイズ]を選択してください。[カスタマイズ]ダイアログボックスを表示します。[コマンド]タブを選択してください。
2. [カテゴリ]リストからボタンのカテゴリを選択し、使用できるボタンを参照してください。[ボタン]エリアからボタンを選択するとそのボタンの機能を表示します。
3. ボタンをクリックしてダイアログボックスからツールバーにドラッグしてください。



ツールバーからボタンを削除するには

1. [基本設定->カスタマイズ]を選択してください。[カスタマイズ]ダイアログボックスを表示します。[コマンド]タブを選択してください。
2. ボタンをツールバーから[ボタン]エリアへドラッグしてください。

ユーザが定義したツールバー名を変えるには

1. [基本設定->カスタマイズ]を選択してください。[カスタマイズ]ダイアログボックスを表示します。[ツールバー]タブを選択してください。
2. [ツールバー]リストでユーザが定義したツールバーで名前を変更したいものを選択してください。
3. [ツールバー名]フィールドでツールバー名を変更してください。
4. [OK]ボタンをクリック新しいツールバー名を保存します。

ユーザ定義のツールバーを削除するには

1. [基本設定->カスタマイズ]を選択してください。[カスタマイズ]ダイアログボックスを表示します。[ツールバー]タブを選択してください。
2. [ツールバー]リストからユーザ定義のツールバーを選択すると[リセット]ボタンが[削除]ボタンに変化します。[削除]ボタンをクリックしてください。

標準のツールバーを初期状態に戻すには

1. [基本設定->カスタマイズ]を選択してください。[カスタマイズ]ダイアログボックスを表示します。[ツールバー]タブを選択してください。
2. [ツールバー]リストから標準のツールバーを選択し[リセット]ボタンをクリックしてください。

ツールバーの配置を初期状態に戻すには

1. [基本設定->カスタマイズ]を選択してください。[カスタマイズ]ダイアログボックスを表示します。[ツールバー]タブを選択してください。
2. [配置のリセット]ボタンをクリックしてください。

すべてのツールバーをデフォルト位置に戻します。

セッション中に設定した配置は、すべてリセットされるので注意してください。

ツールバーのツールチップを表示/非表示するには

1. [基本設定->カスタマイズ]を選択してください。[カスタマイズ]ダイアログボックスを表示します。[ツールバー]タブを選択してください。
2. [ツールチップ表示]チェックボックスをオンにすると表示、チェックしないと非表示になります。

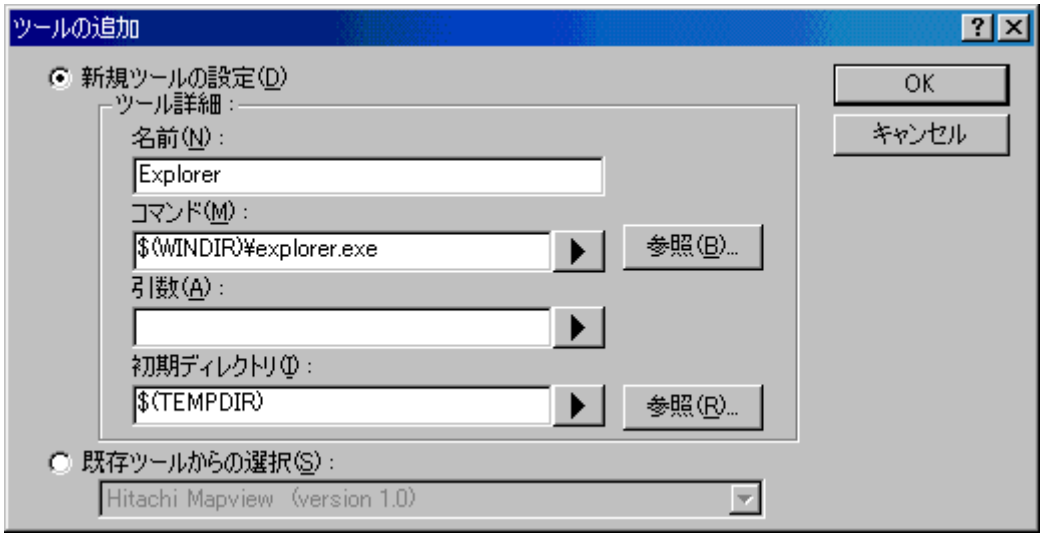
6.2 ツールメニューのカスタマイズ

ツールメニューをカスタマイズして新しいメニューオプションを含めることができます。

新しいメニューオプションを追加するには

1. [基本設定->カスタマイズ]を選択してください。[カスタマイズ]ダイアログボックスを表示します。[メニュー]タブを選択してください。最初に、全ワークスペースに適用できるグローバルアプリケーションワイドツールを追加する（[アプリケーション内有効:] に追加）か、または現在のワークスペースのみに適用できるワークスペースワイドツールを追加する（[ワークスペース内有効:] に追加）かどうかを決めてください。一度決定したら、ダイアログボックスの該当部分を選択するようにしてください。

2. [追加]ボタンをクリックしてください。[ツールの追加]ダイアログボックスを表示します。既存のシステムツールをメニューに追加するには、[既存ツールからの選択]ラジオボタンを選択し、ドロップダウンリストからシステムツールを選択し、[OK]ボタンをクリックしてください。
3. [名前]フィールドにツール名を入力してください。
4. [コマンド]フィールドにコマンドを入力してください。ただし、コマンドに渡す引数は入力しないでください。
5. [引数]フィールドにコマンドに渡す引数を入力してください。
6. [初期ディレクトリ]フィールドにツールを実行する初期ディレクトリを入力してください。
7. [OK]ボタンをクリックするとメニューオプションを[ツール]メニューに追加します。



新しいメニューオプションをリストの最後に追加します（ツールメニューの一番下）。

メニューオプションを変更するには

1. [基本設定->カスタマイズ]を選択してください。[カスタマイズ]ダイアログボックスを表示します。[メニュー]タブを選択してください。
2. 変更するメニューオプションを選択し[変更]ボタンをクリックしてください。
3. [ツールの変更]ダイアログボックスを変更後、[OK]ボタンをクリックしてください。

メニューオプションを削除するには

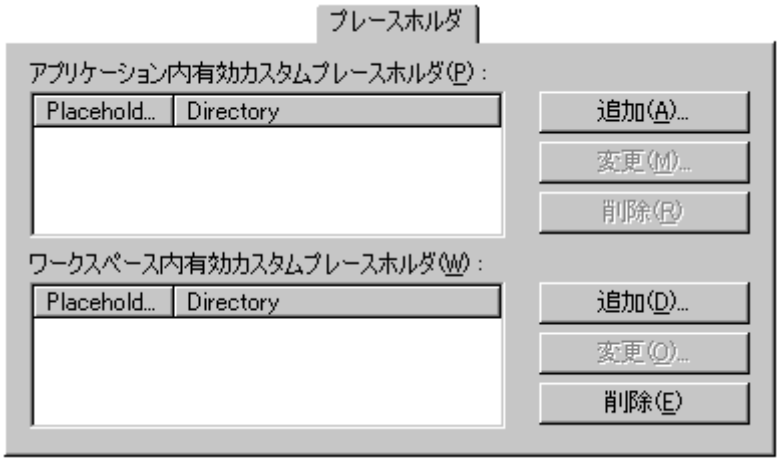
1. [基本設定->カスタマイズ]を選択してください。[カスタマイズ]ダイアログボックスを表示します。[メニュー]タブを選択してください。
2. 削除するメニューオプションを選択し[削除]ボタンをクリックしてください。

6.3 カスタムプレースホルダを使う

HEW では、ディレクトリを定義するときに、以前に定義したプレースホルダを指定できます。これによって、プロジェクトを再配置できます。

また、HEW では、カスタムプレースホルダを定義することもできます。このことは、ユーザがカスタムプレースホルダを定義し、そのディレクトリの値を決められることを意味します。一度定義されると、このプレースホルダは HEW の別の場所で有効になります。

[アプリケーション内有効プレースホルダ]に定義したプレースホルダは、HEW で使用するすべてのワークスペースやプロジェクトに対して有効です。それに対して、[ワークスペース内有効プレースホルダ]に定義したプレースホルダは、現在のワークスペースにおいてのみ有効です。



カスタムプレースホルダを追加するには

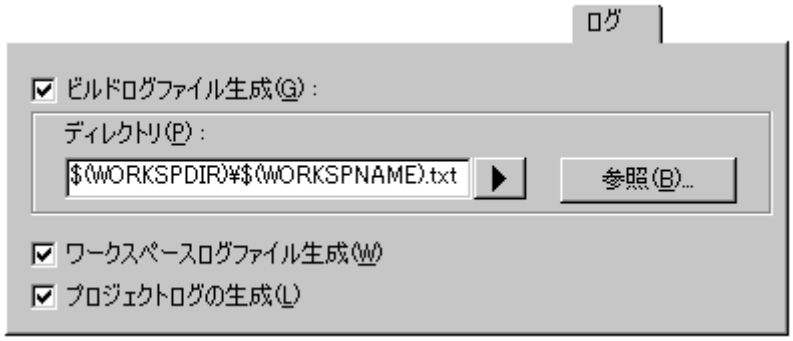
1. [基本設定->カスタマイズ]を選択してください。カスタマイズダイアログボックスを表示します。[プレースホルダ]タブを選択してください。
2. [アプリケーション内有効プレースホルダ]、あるいは[ワークスペース内有効ホルダ]のどちらを使いたいか、選択してください。リスト横にある[追加]ボタンをクリックしてください。
3. [新規カスタムプレースホルダ]ダイアログボックスを表示します。
4. このフィールドでは、プレースホルダの適切な名前、およびプレースホルダが意味する内容の説明を選択してください。
5. 次に、このプレースホルダと関連するディレクトリを選択してください。$(PROJDIR) のように、オプション設定等で定義済みのプレースホルダとして使用できます。

注：

HEW V.2.01 より、カスタムプレースホルダとしてユーザ定義のディレクトリを設定できます。このカスタムプレースホルダはツールチェインオプションの設定に使えます。ディレクトリパス指定の際は、下の図のように[ディレクトリ]に絶対パスを指定してください。

6.4 ワークスペースやプロジェクトのログ機能を使う

HEW では、ワークスペースとプロジェクトのログ機能をアプリケーションの中に統合しています。この機能は、[カスタマイズ]ダイアログボックスの[ログ]タブで設定できます。この設定により、ユーザ名や変更内容についてログファイルを作成できるため、ネットワークデータベースが動作しているときなどに特に便利です。



[ワークスペースログファイル生成]をクリックすると、ワークスペースのすべての変更箇所がワークスペースと同じ名前で拡張子が.log のファイルにログします。このファイルの位置は、ワークスペースファイルと同じディレクトリです。

[プロジェクトログの生成]をクリックすると、現在のワークスペースでのすべてのプロジェクトの変更箇所がプロジェクトと同じ名前で拡張子が.log のファイルにログします。このファイルの位置は、ワークスペースファイルと同じディレクトリです。

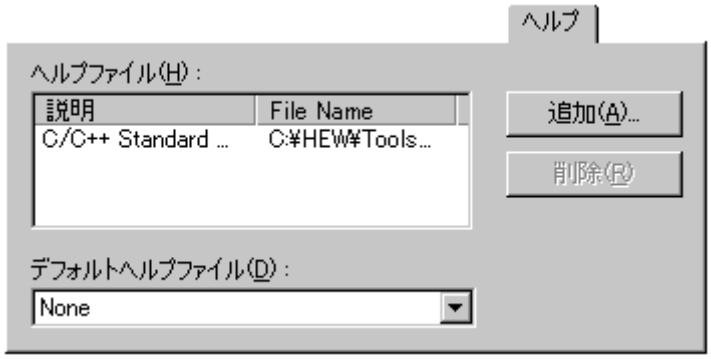
ワークスペースを保存するとログファイルを更新します。

6.5 ヘルプシステムを構築する

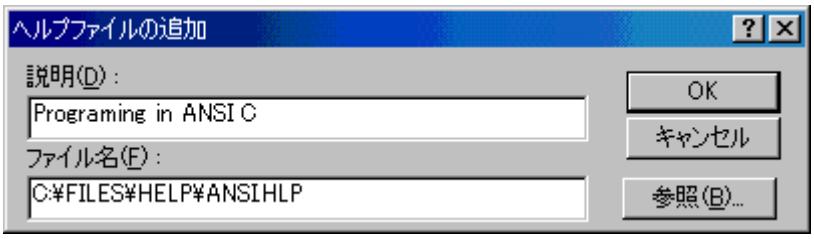
HEW ではエディタウィンドウでコンテキスト依存ヘルプを提供します。エディタウィンドウでテキストを選択し F1 キーを押すと、選択したテキストに関するヘルプを検索します。検索するヘルプファイルは[カスタマイズ]ダイアログボックスの[ヘルプ]タブに表示します。

新しいヘルプファイルを追加するには

1. [基本設定->カスタマイズ]を選択してください。[カスタマイズ]ダイアログボックスを表示します。[ヘルプ]タブを選択してください。



2. [追加]ボタンをクリックしてください。[ヘルプファイルの追加]ダイアログボックスを表示します。



3. [説明]フィールドにヘルプファイルの説明を入力してください。
4. [ファイル名]フィールドにヘルプファイルへのフルパスを入力してください（または[参照]ボタンをクリックしてファイルを選択してください）。
5. [OK]ボタンをクリックすると新しいヘルプファイルを定義します。

デフォルトのヘルプファイルを設定する場合は[デフォルトヘルプファイル]ドロップダウンリストからヘルプファイルを選択してください。F1 キーを押したときに任意のヘルプファイルを参照するには[(None)]を選択してください。

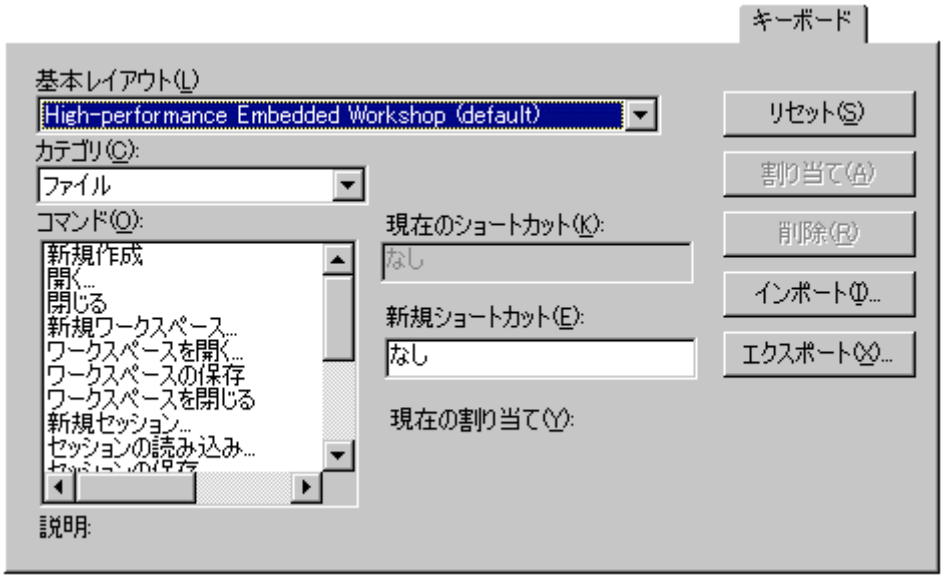
ヘルプファイルを削除するには

1. [基本設定->カスタマイズ]を選択してください。[カスタマイズ]ダイアログボックスを表示します。[ヘルプ]タブを選択してください。
2. 削除するファイルを選んで[削除]ボタンをクリックしてください。
3. [OK]ボタンをクリックしてください。

6.6 キーボードショートカットをカスタマイズする

HEW では、キーボードショートカットを自由にカスタマイズできます。HEW 以外のツールを以前使用していた場合、主な操作を特によく使うキーボードショートカットに変えることができます。

[基本設定–>カスタマイズ]を選択してください。[カスタマイズ]ダイアログボックスを表示します。[キーボード]タブを選択してください。



このダイアログボックスで、HEW のデフォルトのキーボードショートカット、あるいは、PD デバッガショートカットのどちらかを選択することができます。キーボードのレイアウト全体を変更するには、[基本レイアウト]ドロップダウンリストから項目を 1 つ選んでください。デフォルトでは、High-performance Embedded Workshop 設定を使用しています。

このダイアログボックスでいろいろな操作ができます。

新規キーボードショートカットを割り当てるには

1. 変更したいコマンドのメインメニューの[カテゴリ]を選択してください。メニューのあるコマンドのみ変更できます。いくつかの特殊なコマンド（カテゴリ名は"Grobal"）も変更できます。
2. ショートカットを割り当てたいコマンドを[コマンド]リストから選択してください。
3. 現在のショートカットを[現在のショートカット]フィールドに表示します。
4. [新規ショートカット]フィールドでキー入力してください。キーをいろいろ組み合わせることができます。例えば、[Ctrl+J]、[Ctrl+Shift+O]などです。選択したショートカットを既に使用している場合、[現在の割り当て]フィールドにコマンドを表示します。
5. [割り当て]ボタンをクリックしてください。
6. [OK]ボタンをクリックしてください。[OK]ボタンをクリックするまで変更は保存しません。

キーボードショートカットを削除するには

1. 変更したいコマンドのメインメニューの[カテゴリ]を選択してください。メニューのあるコマンドのみ変更できます。いくつかの特殊なコマンド（カテゴリ名は"Grobal"）も変更できます。
2. ショートカットを削除したいコマンドを[コマンド]リストから選択してください。
3. 現在のショートカットを[現在のショートカット]フィールドに表示します。
4. [削除]ボタンをクリックしてください。
5. [OK]ボタンをクリックしてください。

キーボードショートカットをすべてリセットするには

1. [リセット]ボタンをクリックしてください。すべてのショートカットが、現在選択しているキーボードの配置のデフォルト設定に戻ります。
2. [OK]ボタンをクリックしてください。[OK]ボタンをクリックするまで変更は保存しません。

[キーボード]タブでは、キーボードの設定を決められたファイルへインポートしたり、エクスポートしたりできます。これにより、あるマシンから別のマシンへ設定を転送することができるようになります。

キーボードショートカットをインポートするには

1. [基本設定–>カスタマイズ]を選択してください。[カスタマイズ]ダイアログボックスを表示します。[キーボード]タブを選んでください。
2. [基本レイアウト]リストから置き換えたいキーボードの配置を選択してください。
3. [インポート]ボタンをクリックしてください。
4. [レイアウトのインポート]ダイアログボックスを表示します。キーボードの配置をインポートするファイル名を選択してください。
5. [OK]ボタンをクリックしてください。

キーボードショートカットをエクスポートするには

1. [エクスポート]ボタンをクリックしてください。
2. [レイアウトのエクスポート]ダイアログボックスを表示します。現在選択しているキーボードの配置の設定をエクスポートするファイル名を選択してください。
3. [OK]ボタンをクリックしてください。

6.7 基本設定の影響範囲

6.7.1 カスタマイズダイアログボックスの影響範囲

[基本設定->カスタマイズ]を選択すると[カスタマイズ]ダイアログボックスを表示します。[カスタマイズ]ダイアログボックスの設定内容は各タブの項目により影響範囲が異なります。下の表に各タブの項目の影響範囲を示します。

タブ	コントロール	影響範囲
[ツールバー]	すべて	ワークスペース単位
[コマンド]	すべて	システム全体
[メニュー]	アプリケーション内有効	システム全体
	ワークスペース内有効	ワークスペース単位
[プレースホルダ]	アプリケーション内有効カスタムプレースホルダ	システム全体
	ワークスペース内有効カスタムプレースホルダ	ワークスペース単位
[デバッガ]	外部デバッガツール	プロジェクト単位
	外部デバッガのディレクトリ	デフォルト（システム全体および各プロジェクト）
	コマンドラインオプション	プロジェクト単位
	セッションファイル	プロジェクト単位
	ダウンロードモジュール	プロジェクト単位
[ログ]	すべて	ワークスペース単位
[ヘルプ]	すべて	ワークスペース単位
[キーボード]	すべて	システム全体

6.7.2 オプションダイアログボックスの影響範囲

[基本設定->オプション]を選択すると[オプション]ダイアログボックスを表示します。[オプション]ダイアログボックスの設定内容はシステム全体に影響します。

6.8 ワークスペースオプションを指定する

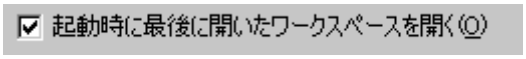
HEW では[オプション]ダイアログボックスでワークスペースの様々なオプションを設定できます。[基本設定–>オプション]の[ワークスペース]タブを選択してください。

6.8.1 起動時に最後に開いたワークスペースを開く

HEW を終了すると最後に開いたワークスペースを保存します。次に HEW を起動したとき、最後に開いたワークスペースを自動的に開きます。

起動時に最後に開いたワークスペースを開くには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [ワークスペース]タブを選択してください。
3. [起動時に最後に開いたワークスペースを開く]チェックボックスをオンにしてください。
4. [OK]ボタンをクリックします。

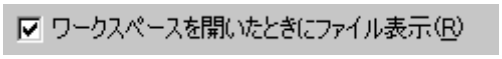


6.8.2 ワークスペースを開いたときにファイルを表示する

HEW は、ワークスペースを閉じるとき開いていたファイルを記憶します。そして、再びワークスペースを開くとき、HEW は同じファイルを復帰させる（つまり開く）ことができます。これにより、中断したセッションを引き続き行うことができます。このチェックボックスをオンにすると、ワークスペースを開いたときに前回開いていたファイルを開きます。

ワークスペースを開いたときにファイルを表示するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [ワークスペース]タブを選択してください。
3. [ワークスペースを開いたときにファイルを表示]チェックボックスをオンにしてください。
4. [OK]ボタンをクリックします。

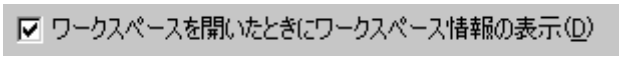


6.8.3 ワークスペースを開いたときにワークスペース情報を表示する

多くのワークスペースを使うと、各ワークスペースの内容を覚えておくのは難しくなります。HEW では各ワークスペースの説明を入力しておくことができます。ワークスペースを開くときは常にこの説明を表示します。

ワークスペースの説明を入力するには

1. ワークスペースウィンドウの[Projects]タブでワークスペースアイコンを選択してください。
2. 右クリックし、ポップアップメニューから[プロパティ]オプションを選択してください。[ワークスペースプロパティ]ダイアログボックスを表示します。
3. [情報]フィールドに説明を入力してください。
4. ワークスペースを開いたときに[ワークスペースプロパティ]ダイアログボックスを開くようにするには、[ワークスペースを開いたときにワークスペース情報の表示]チェックボックスをオンにしてください。このチェックボックスは[オプション]ダイアログボックスの[ワークスペース]タブの[ワークスペースを開いたときにワークスペース情報の表示]チェックボックスと同じ機能です。
5. [OK]ボタンをクリックします。

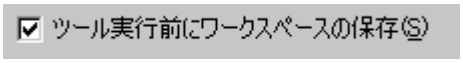


6.8.4 ツール実行前にワークスペースを保存する

このチェックボックスをオンにすると、ビルドフェーズを実行（[ビルド]、[すべてビルド]、[コンパイル]操作）する前や、バージョン管理コマンドを実行する前に、現在のワークスペースを保存できます。

ツール実行前にワークスペースを保存するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [ワークスペース]タブを選択してください。
3. [ツール実行前にワークスペースの保存]チェックボックスをオンにしてください。
4. [OK]ボタンをクリックします。

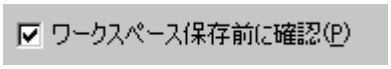


6.8.5 ワークスペース保存前に確認する

[ツール実行前にワークスペースを保存する]機能を使用する場合、ワークスペースを保存する前に、HEW は確認の画面を表示します。ツール実行前のワークスペースの保存について、詳細は「6.8.4 ツール実行前にワークスペースを保存する」を参照してください。

ワークスペース保存前に確認の画面を表示するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [ワークスペース]タブを選択してください。
3. [ワークスペース保存前に確認]チェックボックスをオンにしてください。
4. [OK]ボタンをクリックします。

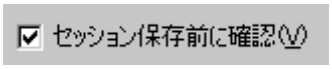


6.8.6 セッション保存前に確認する

このチェックボックスをオンにすると、HEW はセッションをディスクに保存する前にプロンプトを表示します。

セッションをディスクに保存する前にプロンプトを表示するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [ワークスペース]タブを選択してください。
3. [セッション保存前に確認]チェックボックスをオンにしてください。
4. [OK]ボタンをクリックします。



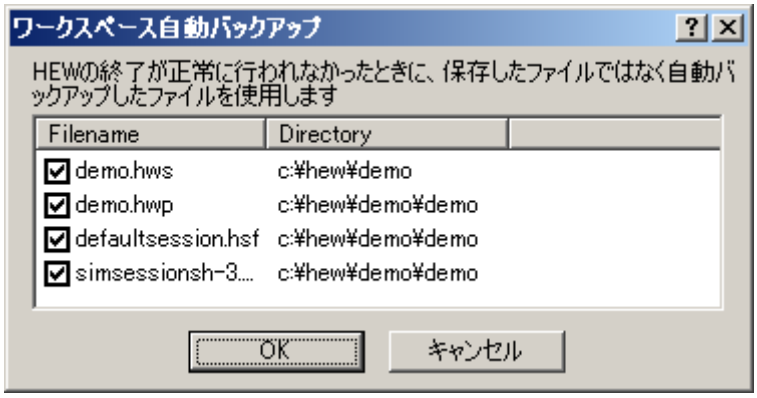
6.8.7 自動バックアップ機能を有効にする

このチェックボックスをオンにすると、指定した時間間隔（デフォルト 10 分）で、ワークスペース、プロジェクト、およびセッションファイルをバックアップします。加えた変更はすべて、テンポラリファイルに保存します。

自動バックアップ機能を有効にするには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [ワークスペース]タブを選択してください。
3. [自動バックアップ時間間隔]チェックボックスをオンにしてください。
4. 自動バックアップ機能を起動する時間（分）を選択してください（最大 60 分）。
5. [OK]ボタンをクリックします。

ワークスペースを開いて以下のダイアログボックスが表示される場合、前回そのワークスペースを使用したときに問題が発生したことを意味します。



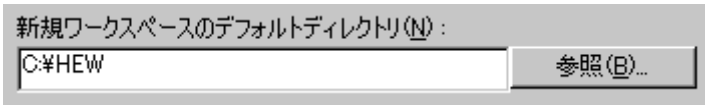
ファイルを回復するには、回復したいファイルの名前の横にあるチェックボックスをオンにし、[OK]ボタンをクリックしてください。[キャンセル]ボタンをクリックすると、自動バックアップファイルを削除し、元のファイルからロードを行います。

6.8.8 新規ワークスペースのデフォルトディレクトリを変更する

新しいワークスペースを作成すると[新規プロジェクトワークスペース]ダイアログボックスが開きます。このダイアログボックスにはその新しいワークスペースが作成されるディレクトリを入力するフィールドがあります。デフォルトでは、HEW インストールディレクトリが入力してあります。しかし、他のディレクトリ（例：“C:¥Workspaces”）にデフォルトを変更したい場合、このフィールドにそのディレクトリを入力するか、[参照]ボタンで位置を指定してください。

新規ワークスペースのデフォルトディレクトリを変更するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [ワークスペース]タブを選択してください。
3. 新しいワークスペースを作成するディレクトリを[新規ワークスペースのデフォルトディレクトリ]フィールドに入力するか、[参照]ボタンをクリックして参照してください。
4. [OK]ボタンをクリックします。



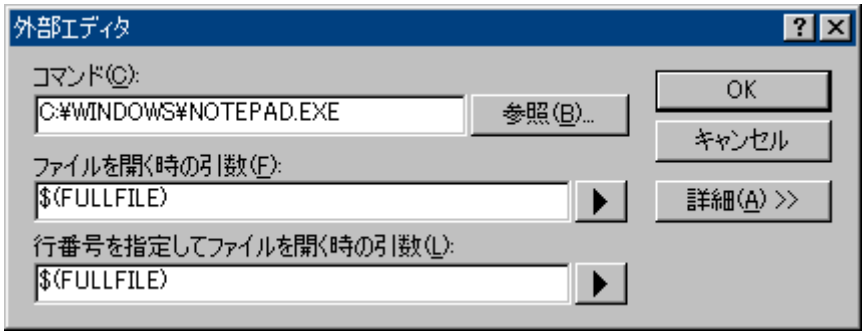
6.9 HEW エディタ以外のエディタを使う

HEW エディタ以外のエディタ（外部エディタと呼びます）も使うことができます。外部エディタを指定した場合、以下を操作すると外部エディタで開きます。

- [ファイル->開く]メニューを選択すると表示される[ファイルを開く]ダイアログボックスでファイルを選択したとき
- [ファイル->最近使ったファイル]メニューからファイルを選択したとき
- ワークスペースウィンドウの[Projects]タブのファイルをダブルクリックしたとき
- ワークスペースウィンドウの[Navigation]タブのエントリをダブルクリックしたとき
- アウトプットウィンドウの[Build]タブのエラーまたはウォーニングをダブルクリックしたとき
- アウトプットウィンドウの[Find in Files]タブのエントリをダブルクリックしたとき
- ワークスペースウィンドウの[Projects]タブのファイル上で右クリックしポップアップメニューから[開く <ファイル名> (外部エディタ)]を選択したとき
- [編集->ソースブレークポイント]メニューを選択すると表示される[ブレークポイント]ダイアログボックスで、[コードの編集]ボタンをクリックしたとき
- Windows(R) Explorer から HEW ウィンドウ上にファイルをドラッグアンドドロップしたとき（ただしワークスペースウィンドウの[Projects]タブのアクティブプロジェクトやユーザフォルダ上にファイルをドラッグアンドドロップしたときは、プロジェクトにファイルを追加します。このときファイルは開きません。）

HEW エディタ以外のエディタを指定するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [エディタ]タブを選択してください。
3. [外部エディタの使用]チェックボックスをオンにしてください。[外部エディタ]ダイアログボックスを表示します。



4. [コマンド]フィールドに実行可能ファイルのパス（引数をのぞいて）を入力してください。
5. [ファイルを開く時の引数]フィールドにファイルを開くために必要な引数を入力してください。開くファイルのパスには $(FULLFILE) プレースホルダを使ってください。
6. [行番号を指定してファイルを開く時の引数]フィールドにファイルの特定の行を開くために必要な引数を入力してください。開くファイルのパスには $(FULLFILE) プレースホルダを使ってください。また、カーソルを最初に置く行の番号には $(LINE) プレースホルダを使ってください。
7. [詳細>>]ボタンをクリックすると、次のファイルを開く操作の場合に外部エディタまたは内蔵エディタ（HEW エディタ）のどちらでファイルを開くかを選択できます。

- [ファイル->開く]メニューから：
  [ファイルを開く]ダイアログボックスでファイルを選択したとき
  デフォルトは"内蔵エディタ"が選択されています。
- ワークスペースウィンドウ上でダブルクリック：
  ワークスペースウィンドウの[Projects]タブのファイルをダブルクリックしたとき
  ワークスペースウィンドウの[Navigation]タブのエントリをダブルクリックしたとき
  デフォルトは"外部エディタ"が選択されています。
- アウトプットウィンドウ上でダブルクリック：
  アウトプットウィンドウの[Build]タブのエラーまたはウォーニングをダブルクリックしたとき
  アウトプットウィンドウの[Find in Files]タブのエントリをダブルクリックしたとき
  デフォルトは"外部エディタ"が選択されています。
- HEW ウィンドウにドラッグアンドドロップ：
  Windows(R) Explorer から HEW ウィンドウ上にファイルをドラッグアンドドロップしたとき（ただしワークスペースウィンドウの[Projects]タブのアクティブプロジェクトやユーザフォルダ上にファイルをドラッグアンドドロップしたときは、プロジェクトにファイルを追加します。このときファイルは開きません。）
  デフォルトは"内蔵エディタ"が選択されています。

8. [OK]ボタンをクリックします。

[外部エディタの使用]チェックボックスをオンにした場合、以下を操作すると常に外部エディタで開きます。

- [編集->ソースブレークポイント]メニューを選択すると表示される[ブレークポイント]ダイアログボックスで、[コードの編集]ボタンをクリックしたとき
- [ファイル->最近使ったファイル]メニューからファイルを選択したとき

また、ワークスペースウィンドウの[Projects]タブのファイル上で右クリックしポップアップメニューを表示すると、[開く <ファイル名>]メニューオプションの下に[開く <ファイル名> (外部エディタ)]メニューオプションが追加されます。

注：

HEW エディタ以外のエディタを使う場合、以下のことに注意してください。

- どのように起動しても、外部エディタを起動するたびに、エディタは新規に起動します。

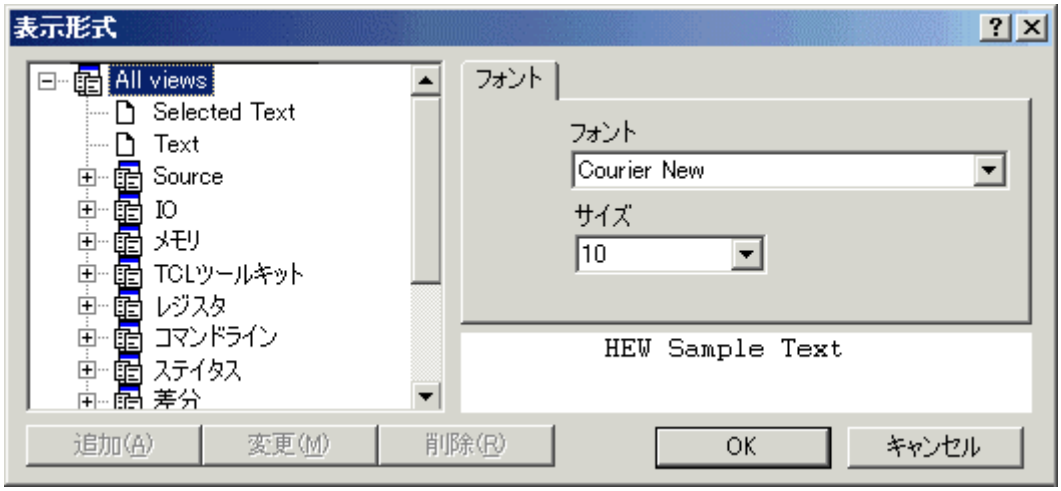
- ビルド操作（[ビルド]、[すべてビルド]、[コンパイル]）を行う前にファイルを保存してください。

6.10　表示するフォントのカスタマイズ

HEW のコンポーネントは違った外観で表示させることが可能です。表示するフォントやテキストの色を変更できます。

ウィンドウの外観を変更するには

1. [基本設定->表示の形式]を選択してください。[表示形式]ダイアログボックスを表示します。
2. 外観を変更したい表示を選択してください。すべての表示を変更するには、[All views]のカテゴリを選択してください。
3. ツリーの項目を展開し、外観の変更が可能なすべての項目を表示します。
4. 項目を選択してください。本ダイアログボックスの右側にある、タブの変更に注意してください。
5. [フォント]または[カラー]を変更してください。
6. 変更を保存するには[OK]ボタンをクリックしてしてください。



各ウィンドウに表示されているテキストの色やフォントなどが[表示形式]ダイアログボックスでカスタマイズ可能です。

項目	カテゴリ名	カスタマイズ対象
Source	Selected Text	エディタウィンドウおよび逆アセンブリウィンドウ（ソースモード）に表示されているテキストで選択した部分のテキスト行
	PC Line Highlight	[オプション]ダイアログボックスの[エディタ]タブで、[PC カーソル行のハイライト表示]チェックボックスがオンの場合に、エディタウィンドウおよび逆アセンブリウィンドウ（ソースモード）の PC（プログラムカウンタ）の位置がある行
	その他、各々のファイルタイプのシンタックスに基づいたキーワードグループ	エディタウィンドウおよび逆アセンブリウィンドウ（ソースモード）に表示されているコメントまたはキーワード*1 *1. 制御文、式文、宣言文、プリプロセッサ制御文、データ文、データ群、ディレクディブ、値
差分	LHS Different Lines	差分ウィンドウの左側のペインに表示されている不一致テキスト行
	LHS Moved Lines	差分ウィンドウの左側のペインに表示されている移動されたテキスト行
	RHS Different Lines	差分ウィンドウの右側のペインに表示されている不一致テキスト行
	RHS Moved lines	差分ウィンドウの右側のペインに表示されている移動されたテキスト行
レジスタ	変更	レジスタウィンドウに表示されている値が変更された部分のテキスト
メモリ	アクセス履歴あり	メモリウィンドウに表示されているカバレッジ計測時のコード実行部分のテキスト
	アクセス履歴なし	メモリウィンドウに表示されているカバレッジ計測時のコード未実行部分のテキスト
	カバレッジ計測領域外	メモリウィンドウに表示されているカバレッジ計測領域外の部分のテキスト
	メモリ領域外	メモリウィンドウに表示されているメモリ領域外の部分のテキスト
	変更	メモリウィンドウに表示されている値が変更された部分のテキスト
Output	Build Error	アウトプットウィンドウの[Build]タブに表示されているエラーメッセージ行
	Build Warning	アウトプットウィンドウの[Build]タブに表示されているウォーニングメッセージ行
	Information	アウトプットウィンドウの[Build]タブに表示されているインフォメーションメッセージ行
逆アセンブリ	PC Line Highlight	[オプション]ダイアログボックスの[エディタ]タブで、[PC カーソル行のハイライト表示]チェックボックスがオンの場合にエディタウィンドウおよび逆アセンブリウィンドウ（混合モードおよび逆アセンブリモード）で PC（プログラムカウンタ）の位置がある行
	ソース行	エディタウィンドウおよび逆アセンブリウィンドウ（混合モード）に表示されているソース行
全般	Text	各ウィンドウに表示されているテキスト

6.11 バーチャルデスクトップを使用する

HEW はバーチャルデスクトップの概念を導入しています。ウィンドウ配置をボタンのクリックで切り替えることができます。ボタンをクリックすると、ウィンドウ配置の設定によりツールバーとウィンドウが表示または非表示になります。

最大 4 パターンのデスクトップ構成が可能です。セッションを保存すると、各構成のウィンドウ位置がセッションファイルに保存されます。構成を切り替えるだけで他のウィンドウへアクセスできます。ツールバーとウィンドウはバーチャルデスクトップ構成に依存します。ソースファイルは、バーチャルデスクトップのシステムから独立しているため表示します。

デスクトップ構成に新しい名前を付けるには

1. [ウィンドウ->バーチャルデスクトップ]サブメニューから[デスクトップマネージャ]ダイアログボックスを選択してください。
2. [デスクトップマネージャ]ダイアログボックスで、名前を変更するデスクトップ構成を選択してください。

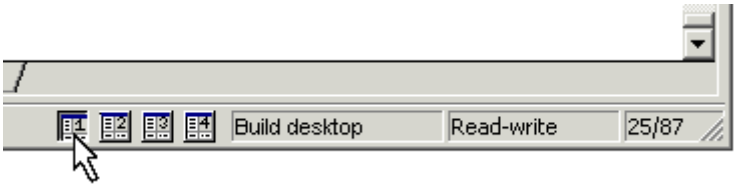
3. [名前の変更]ボタンをクリックしてください。新しい名前を入力し、[OK]ボタンをクリックしてください。
4. [OK] ボタンをクリックしてください。

デスクトップ構成を切り替えるには

デスクトップ構成の切り替えには、いくつかの方法があります。

もっとも簡単な方法は以下の通りです。

1. ステータスバー上のバーチャルデスクトップボタンをクリックします。



この例では、選択されたデスクトップ番号は”1”で、“Build”という名前が付いています。ボタンの右側のボックスに“Build desktop”と表示しています。別のデスクトップボタンをクリックするとボタンの右側のボックスの表示も切り替わります。一度クリックすると、新しい構成の形式でウィンドウを読み込みます。

もう 1 つの方法は以下の通りです。

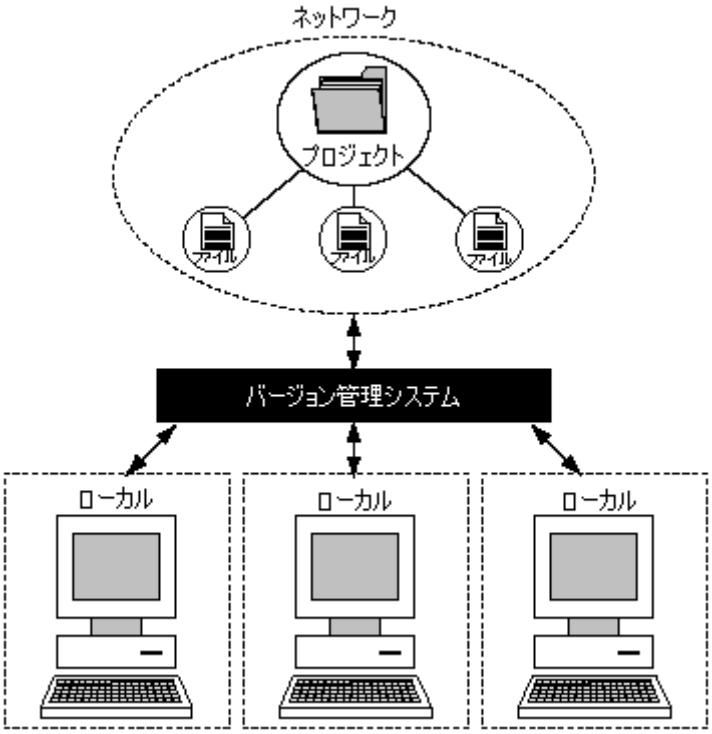
1. [ウィンドウ->バーチャルデスクトップ]サブメニューから表示したいデスクトップ構成を選択してください。
2. 選択したデスクトップ構成のサブメニューにチェックマークが付きます。

7. バージョン管理

High-performance Embedded Workshop (HEW) はバージョン管理システムと接続できます。プロジェクトでバージョン管理システムを使用する理由を以下に示します。

- プロジェクト開発環境の統合性を維持するため
- プロジェクトのバージョンを記録･保存するため
- ソースファイルに対するバージョン管理を行い、複数のユーザが 1 つのプロジェクトを共同開発できるようにするため

バージョン管理システムを使用するプロジェクトの一般例を示します。ここでは複数のユーザがソースコードを相互参照するために同じ共有ネットワークドライブを使用しています。バージョン管理システムはソースファイルの参照や更新を管理するために使用します。



- **HEW のバージョン管理の対象ファイル**

HEW のバージョン管理の対象ファイルは標準では以下の種類があります。

ファイルの種類	拡張子グループ名	ファイルの拡張子	ファイルのサポート CPU 種別
ワークスペースファイル	HEW Workspace	hws	共通
プロジェクトファイル	HEW Projects	hwp	共通
バージョン管理用ファイル	Configuration File	hvc	共通
ソースファイル	C source file	c	共通
	C++ source file	cpp	SuperH RISC engine, H8S, H8SX, H8/300
	Assembly source file	a30	M32C, M16C
		asm	SuperH RISC engine, H8S, H8SX, H8/300
		mar	
		src	
インクルードファイル	C header file	h	共通
	C++ header file	hpp	SuperH RISC engine, H8S, H8SX, H8/300
	Assembly include file	inc	共通

上記以外のファイルもプロジェクトに追加すると、バージョン管理の対象になります。

- **HEW と接続可能なバージョン管理システム**

HEW はバージョン管理システム自体を提供していません。あらかじめバージョン管理システムがマシンにインストールされている必要があります。HEW の GUI インタフェースを使用しバージョン管理システムと接続が可能です。

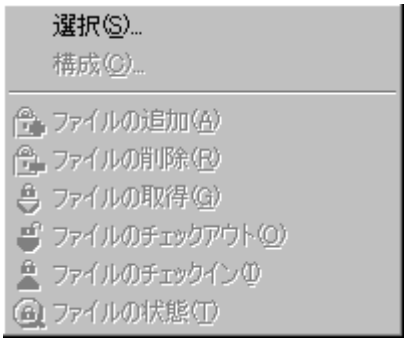
はじめに HEW と接続するバージョン管理システムを選択します。HEW と接続可能なバージョン管理システムは下記の 2 種類のいずれかです。

分類	バージョン管理システム選択時の項目名	説明
カスタマイズ設定によるバージョン管理	Custom VCS	MS-DOS コマンドプロンプトからコマンドラインコマンドを実行するバージョン管理システムを対象としています。 （CVS、RCS、PVCS など） バージョン管理システムのコマンドと HEW の GUI インタフェースを関連付けし定義する必要があります。
Microsoft(R) Visual SourceSafe 5.0 または 6.0 によるバージョン管理	Microsoft Visual SourceSafe 5.0/6.0	Microsoft(R) Visual SourceSafe 5.0 または 6.0 を対象としています。 主な機能をあらかじめ定義済みです。

7.1 バージョン管理システムを選択する

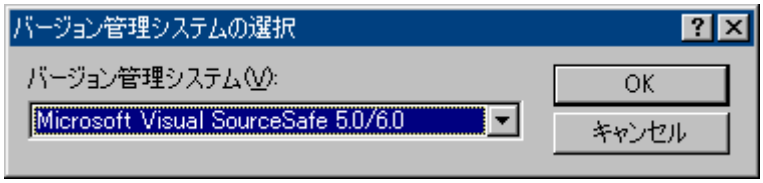
最初に、バージョン管理のサブメニューは以下のように表示します。

このとき、まだ現在のワークスペースでバージョン管理システムを選択していないため、[ツール->バージョン管理->選択]メニューオプションだけが使用できます。



バージョン管理システムを選択するには

1. [ツール->バージョン管理->選択]を選択します。[バージョン管理システムの選択]ダイアログボックスが表示されます。このダイアログボックスにはサポートするバージョン管理システムをすべて表示します。



2. [バージョン管理システム]リストからバージョン管理システムを選択します。
3. [OK]ボタンをクリックします。

- バージョン管理システムに”Custom VCS”を選択した場合は、[ツール->バージョン管理->構成]メニューオプションが使用できるようになります。「8. カスタムバージョン管理システムの使用」を参照してください。
- バージョン管理システムに”Microsoft Visual SourceSafe 5.0/6.0”を選択した場合は、[ツール->バージョン管理->選択]のほかのメニューオプションが使用できるようになります。「9. Visual SourceSafe の使用」を参照してください。

バージョン管理システムの選択を解除するには

1. [ツール->バージョン管理->選択]を選択します。[バージョン管理システムの選択]ダイアログボックスが表示されます。
2. [バージョン管理システム]リストから"<None>"を選択します。
3. [OK]ボタンをクリックします。

7.2 バージョン管理設定内容のインポートとエクスポート

ワークスペースごとに異なるバージョン管理設定を行うことができます。HEW ではそれぞれのバージョン管理設定を保存して他のワークスペースでインポートすることができます。これにより、複数のワークスペースで何度も同じバージョン管理設定を行う必要がなく、時間を節約することができます。

バージョン管理設定をエクスポートするには

1. [ツール->バージョン管理->構成]メニューオプションを選択してください。[バージョン管理設定]ダイアログボックスが表示されます。
2. [エクスポート]ボタンをクリックしてください。[現在のバージョン管理構成をエクスポート]ダイアログボックスが表示されます。
3. 設定内容を保存するディレクトリを選択してください。
4. ファイル名を入力し[保存]ボタンをクリックしてください。

バージョン管理設定をインポートするには

1. [ツール->バージョン管理->構成]メニューオプションを選択してください。[バージョン管理設定]ダイアログボックスが表示されます。
2. [インポート]ボタンをクリックしてください。[バージョン管理構成をインポート]ダイアログボックスが表示されます。
3. インポートするファイル（*.HVC） を選択してください。
4. [インポート]ボタンをクリックしてください。

8. カスタムバージョン管理システムの使用

「7.1 バージョン管理システムを選択する」でバージョン管理システムに"Custom VCS"を選択した場合、HEW と Custom VCS を接続するには次を定義していただく必要があります。

- バージョン管理メニューオプションを定義し、そのオプションに関連したカスタムバージョン管理のコマンド実行ファイル（.EXE）の位置、コマンドパラメータ、およびバージョン管理ツールコマンドを実行した結果の制御方法などの定義
- バージョン管理対象ファイルの位置（ディレクトリマッピング）、および環境変数の定義
- バージョン管理コマンド実行時の制御、およびユーザ設定など一般的なオプションの定義

これらを定義すると、HEW のメニューオプションまたはツールバーボタンを選択したときにカスタムのバージョン管理システムのコマンドが実行でき、その実行結果を HEW で見ることが可能になります。

バージョン管理システムのインストールや設定は、各バージョン管理システムのマニュアルを参照してください。

カスタムバージョン管理システムと HEW との具体的な操作方法は、「8.11 カスタムバージョン管理システムの使用例」を参照してください。

8.1 バージョン管理メニューオプションを定義する

カスタムのバージョン管理システムでは、[ツール->バージョン管理]サブメニューからオプションを選択するか、バージョン管理ツールバーボタンでバージョン管理コマンド実行ファイルを起動できます。関連するコマンドを実行し、出力をアウトプットウィンドウの[Version Control]タブに表示します。

バージョン管理メニューオプションまたはツールバーボタンによりコマンドを実行するには

1. ワークスペースウィンドウの[Projects]タブからバージョン管理コマンドを適用する項目（ワークスペース、プロジェクト、フォルダ、またはファイルなど）を選択してください。右クリックしてポップアップメニューを表示し、[バージョン管理]サブメニューから定義したメニューオプションを選択できます。メニューオプションを選択すると、ワークスペースウィンドウで選択した項目に応じたファイルに対し、メニューオプションに関連付けたコマンドが実行されます。
2. 例えば、ワークスペースアイコンを選択すると、そのワークスペースのプロジェクトのすべてのファイルがバージョン管理コマンドに渡ります。これには、HEW のシステムファイルも含まれます。
3. [ツール->バージョン管理]サブメニューからメニューオプションを選択するか、[バージョン管理]ツールバーボタンを選択してください。

カスタムバージョン管理を利用すると、バージョン管理システムの指定を任意に定義できます。[ツール->バージョン管理->構成] を選択し[バージョン管理設定]ダイアログボックスを表示してください。

[コマンド]タブには 2 つのメニューオプションのリストがあります。

- **システム定義メニューオプション([システムメニューオプションとツールバーボタン])**

1 つめのリスト[システムメニューオプションとツールバーボタン]は、[ツール->バージョン管理]サブメニューに常に表示されているメニューオプションです。これに対応する[バージョン管理]ツールバーがあります。このリストには最もよく使用されるバージョン管理コマンドに関連した 6 種類のメニューオプションが表示されています。

- **ユーザ定義メニューオプション([ユーザメニューオプション])**

2 つめのリスト[ユーザメニューオプション]は、ユーザ定義の追加メニューオプションで、定義すると[ツール->バージョン管理]サブメニューの最後に追加されます。システムメニューオプションに表示されていないメニューオプションを追加する場合はこのリストで定義してください。ユーザ定義のメニューオプションはバージョン管理ツールバーには表示されません。

8.1.1 システム定義メニューオプション

バージョン管理ツールバーボタンは 6 種類あります。これらのボタンは、最もよく使用されるバージョン管理コマンドへのショートカットとなります。最初にワークスペースを作成すると、これらのボタンはアクティブではありません。その理由は、まだバージョン管理コマンドがツールバーボタンに関連付けられていないためです。

ツールバーボタンは、[ツール->バージョン管理]サブメニューの 6 つのメニューオプションと同等です。[ツール->バージョン管理->ファイルの取得]メニューオプションはバージョン管理から[ファイルの取得]ツールバーボタンをクリックするのと同じ効果があります。ツールバーボタンの表示は固定されているので、ツールバーボタンをクリックしたときに実行するコマンドを定義してください。

ツールバーまたは[ツール->バージョン管理]サブメニューのシステム定義オプションからコマンドを起動するには、まず、バージョン管理システムを起動したときに実行する関連したコマンドを定義しなければなりません。オプション名とその説明を下記に示します。

オプション名	説明
ファイルの追加	バージョン管理システムに選択したファイルを追加する
ファイルの削除	バージョン管理システムから選択したファイルを削除する
ファイルの取得	バージョン管理システムから選択したファイルの読み取り専用ローカルファイルを取得する
ファイルのチェックアウト	バージョン管理システムから選択したファイルの書き込み可能なローカルファイルを取得する
ファイルのチェックイン	バージョン管理システムに選択したファイルのローカルコピーを戻し更新する
ファイルの状態	選択したファイルの状態を表示する

システムメニューオプションやツールバーボタンから実行するコマンドを定義するには

1. [ツール->バージョン管理->構成]メニューオプションを選択します。[バージョン管理設定]ダイアログボックスが表示されます。
2. [システムメニューオプションおよびツールバーボタン]リストからコマンドを定義するオプションを選択し[変更]ボタンをクリックすると、[定義コマンド]ダイアログボックスを表示します。
3. [追加]ボタンで選択したメニューオプションにコマンドを定義してください。詳細は「8.2 バージョン管理コマンドを定義する」を参照してください。
4. [OK]ボタンをクリックして[定義コマンド]ダイアログボックスを閉じてください。
5. [OK]ボタンをクリックして[バージョン管理設定]ダイアログボックスを閉じてください。

8.1.2 ユーザ定義メニューオプション

ユーザ定義のメニューオプションは複数作成できます。メニューオプション名も自由に付けられます。すでにユーザ定義のメニューオプションを作成している場合、コマンドの定義の変更、削除、または、メニューオプションの表示順序も指定できます。

ユーザ定義のメニューオプションはバージョン管理ツールバーには表示されません。

新しいバージョン管理メニューオプションを作成するには

1. [ツール->バージョン管理->構成]メニューオプションを選択してください。[バージョン管理設定]ダイアログボックスを表示します。
2. [ユーザメニューオプション]リストの横にある[追加]ボタンをクリックしてください。[メニューオプションの追加]ダイアログボックスを表示します。
3. [オプション]フィールドにメニューオプション名を入力してください。
4. [追加]ボタンをクリックしてください。[コマンド追加]ダイアログボックスを表示します。詳細は「8.2 バージョン管理コマンドを定義する」を参照してください。
5. [OK]ボタンをクリックし、[コマンド追加]ダイアログボックスを閉じてください。
6. [OK]ボタンをクリックし、[メニューオプションの追加]ダイアログボックスを閉じてください。
7. [OK]ボタンをクリックし、[バージョン管理設定]ダイアログボックスを閉じてください。

既存のバージョン管理メニューオプションの定義コマンドを変更するには

1. [ツール->バージョン管理->構成]メニューオプションを選択してください。[バージョン管理設定]ダイアログボックスを表示します。
2. [ユーザメニューオプション]リストから変更するメニューオプションを選択し、[変更]ボタンをクリックしてください。[定義コマンド]ダイアログボックスを表示します。
3. [コマンド]リストから変更するメニューオプションを選択し、[変更]ボタンをクリックしてください。[コマンド変更]ダイアログボックスを表示します。詳細は「8.2 バージョン管理コマンドを定義する」を参照してください。
4. [OK]ボタンをクリックし、[コマンド変更]ダイアログボックスを閉じてください。
5. [OK]ボタンをクリックし、[定義コマンド]ダイアログボックスを閉じてください。
6. [OK]ボタンをクリックし、[バージョン管理設定]ダイアログボックスを閉じてください。

既存のバージョン管理メニューオプションを削除するには

1. [ツール->バージョン管理->構成]メニューオプションを選択してください。[バージョン管理設定]ダイアログボックスを表示します。
2. [ユーザメニューオプション]リストから削除するメニューオプションを選択し、[削除]ボタンをクリックしてください。
3. [OK]ボタンをクリックし、[バージョン管理設定]ダイアログボックスを閉じてください。

バージョン管理メニューオプションの表示順序を変更するには

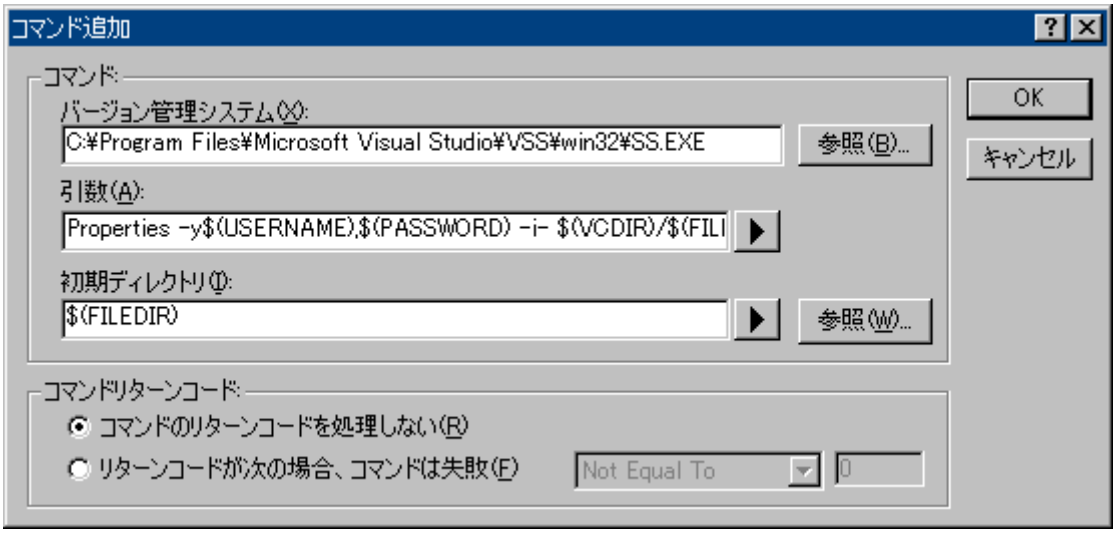
1. [ツール->バージョン管理->構成]メニューオプションを選択してください。[バージョン管理設定]ダイアログボックスを表示します。
2. 移動するメニューオプションを選択し、上に移動する場合は[上へ移動]ボタンを、下に移動する場合は [下へ移動]ボタンを必要に応じてクリックしてください。
3. [OK]ボタンをクリックし、[バージョン管理設定]ダイアログボックスを閉じてください。

8.2 バージョン管理コマンドを定義する

バージョン管理コマンドは、[定義コマンド]ダイアログボックスに表示されます。コマンドを定義する数には制限はなく、実行する順番を指定できます。既存のコマンドは変更または削除できます。

バージョン管理コマンドを定義するには

1. [定義コマンド]ダイアログボックスの[追加]ボタンをクリックしてください。[コマンド追加]ダイアログボックスを表示します。



2. [バージョン管理システム]フィールドにコマンド実行ファイルのフルパスを入力するか、[参照]ボタンをクリックしてコマンド実行ファイルの位置を指定してください。
3. [引数]フィールドにコマンドラインパラメータを入力してください。
4. [初期ディレクトリ]フィールドに実行ファイルを起動するディレクトリを入力するか、[参照]ボタンをクリックして起動ディレクトリを指定してください。ただし、特に起動ディレクトリを指定する必要がない場合は、$(FILEDIR) プレースホルダを指定してファイルのディレクトリと同じディレクトリでコマンドを実行してください。
5. [コマンドリターンコード]オプションを選択してください。オプションの設定は「8.5 バージョン管理コマンドリターンコードオプション」を参照してください。
6. [OK]ボタンをクリックすると新しいコマンドを定義します。

既存のバージョン管理コマンドを修正するには

1. 変更するコマンドを[定義コマンド]ダイアログボックスの[コマンド]リストから選択してください。
2. [変更]ボタンをクリックしてください。[コマンド変更]ダイアログボックスを表示します。
3. 必要に応じて修正を行い、[OK]ボタンをクリックしてください。

既存のバージョン管理コマンドを削除するには

1. 削除するコマンドを[定義コマンド]ダイアログボックスの[コマンド]リストから選択してください。
2. [削除]ボタンをクリックしてください。

バージョン管理メニューオプションの順序を変更するには

1. 移動するメニューオプションを[定義コマンド]ダイアログボックスの[コマンド]リストから選択してください。
2. 必要に応じて[上へ移動]または[下へ移動]ボタンをクリックしてください。

8.3 引数を指定する

引数は正しく指定しなければなりません。もし間違っていると、バージョン管理システムが正しく動作しません。また、1 つのバージョン管理コマンドを複数のファイルに適用する場合があるため、引数をフレキシブルな方法で指定することが重要です。これを行うために、[引数]フィールドにはプレースホルダボタン( )があります（プレースホルダの詳細は、「リファレンス 5. プレースホルダ」を参照してください）。プレースホルダボタンをクリックすると、使用できるプレースホルダをポップアップメニューで表示します。以下の表に各プレースホルダの説明と実際の値を示します。

ポップアップメニュー	プレースホルダ	実際の値
User login name	$(USERNAME)	現在のユーザログイン ([一般]タブ)
User login password	$(PASSWORD)	現在のユーザパスワード ([一般]タブ)
Version control directory	$(VCDIR)	仮想バージョン管理マップ ([プロジェクト]タブ)
Comment	$(COMMENT)	コマンド実行前に指定したコメント
File path + name	$(FULLFILE)	操作するファイル名とそのフルパス
Filename	$(FILENAME)	操作するファイル名（拡張子を含む）
File leaf	$(FILELEAF)	操作するファイル名（拡張子を含まない）
File extension	$(EXTENSION)	操作するファイルの拡張子
File directory	$(FILEDIR)	操作するファイルのディレクトリ
Configuration directory	$(CONFIGDIR)	現在のコンフィグレーションディレクトリ
Project directory	$(PROJDIR)	現在のプロジェクトのディレクトリ
Workspace directory	$(WORKSPDIR)	現在のワークスペースディレクトリ
Temp directory	$(TEMPDIR)	テンポラリディレクトリ
Command directory	$(EXEDIR)	バージョン管理実行ディレクトリ
Windows directory	$(WINDIR)	Windows®がインストールされているディレクトリ
Windows system directory	$(WINSYSDIR)	Windows®のシステムファイルがあるディレクトリ
Workspace name	$(WORKSPNAME)	現在のワークスペース名
Project name	$(PROJECTNAME)	現在のプロジェクト名
Configuration name	$(CONFIGNAME)	現在のコンフィグレーション名

8.4 コメントを指定する

バージョン管理コマンドに“$(COMMENT)”プレースホルダを含む場合、コマンド実行時に[コメント入力]ダイアログボックスが表示されるのでコメントを入力してください。

コメントはコマンドごとに指定できます。また、[全てのファイルにコメントを適用]チェックボックスをオンにし[OK]ボタンをクリックすると、すべてのファイルに同じコメントが指定できます。

8.5 バージョン管理コマンドリターンコードオプション

バージョン管理コマンドが実行されると、その出力はアウトプットウィンドウの[Version Control]タブに表示します。

コマンドの実行が完了するとリターンコードを取得します。コマンドを定義する際、このリターンコードを使用し、コマンド実行中にエラーが発生した、残りのコマンドが実行されるべきでないことを示すことができます (例：abort)。[コマンドリターンコード]グループに含まれる管理機能を使用すれば、この動作を指定できます。

コマンドのリターンコードでエラーを示すことができるならば、[リターンコードが次の場合、コマンドは失敗]オプションを選択して必要に応じてドロップダウンリストおよびエディットボックスを設定してください。[リターンコードが次の場合、コマンドは失敗]オプションが選択されている場合、HEW は各コマンドのリターンコードをチェックしてエラーが起こったかど

うか判定します。エラーが起こった場合、コマンド実行は停止してそれに続くコマンド実行やコマンドの後に続く処理（例：ビルド）は実行されません。

[コマンドのリターンコードを処理しない]オプションが選択されている場合、HEW は各コマンドのリターンコードをチェックしません。したがって、すべてのコマンドを実行します。

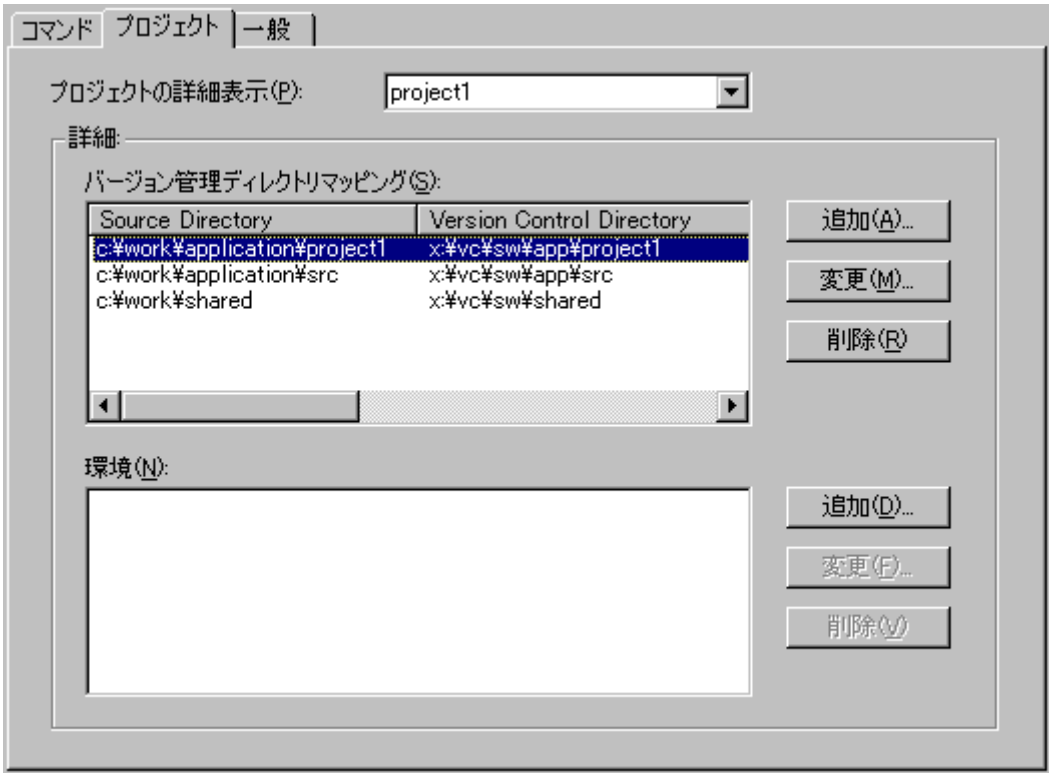
8.6 ディレクトリをマッピングする

ファイルの位置を指定するときには、プレースホルダを使用してください。そうでないと、そのコマンドは指定したファイルにしか適用できません。例えば、バージョン管理のアプリケーションに–GET コマンドが使われていて、ファイルの読み出し専用コピーを作成するとします。このとき、[引数]フィールドは以下のように指定できます。

-GET “c:¥vc¥files¥project¥main.c”

しかし、このコマンドを実行しても、ファイル MAIN.C しかコピーできません。この問題を解決するために、HEW にはプレースホルダとディレクトリのマッピングというシステムがあります。マッピングにより、どの「作業中の」ディレクトリ（つまり、作業中のソースファイルがあるディレクトリ）がどの「管理」ディレクトリ（つまり、バージョン管理システムに保存されているソースファイルのディレクトリ）に対応するかを指定します。これら 2 つのディレクトリ間のマッピングは、[バージョン管理設定]ダイアログボックスの[プロジェクト]タブで指定できます。

マッピングを定義すると、Version control directory プレースホルダ $(VCDIR)を使用してプロジェクトファイルを保存するディレクトリを示すことができます。



新しいマッピングを定義するには

1. [ツール->バージョン管理->構成]を選択してください。[バージョン管理設定]ダイアログボックスを表示します。
2. [プロジェクト]タブを選択してください。
3. [バージョン管理ディレクトリマッピング]リストの横にある[追加]ボタンをクリックしてください。[バージョン管理ディレクトリマッピング]ダイアログボックスを表示します。
4. [ソースディレクトリ]フィールドにマッピングを作成したいディレクトリを設定してください。次のいずれかを行ってください。

- ディレクトリを直接入力する。
- プレースホルダボタンをクリックして表示されるメニューから、Project directory、または Workspace directory を選択する。
- [参照]ボタンをクリックして表示される[ソースディレクトリを参照]ダイアログボックスで、ディレクトリを選び[選択]ボタンをクリックする。

5. [バージョン管理ディレクトリ]フィールドにソースディレクトリへマップするバージョン管理システムのディレクトリを入力してください。
6. [OK]ボタンをクリックしてください。

既存のマッピングを変更するには

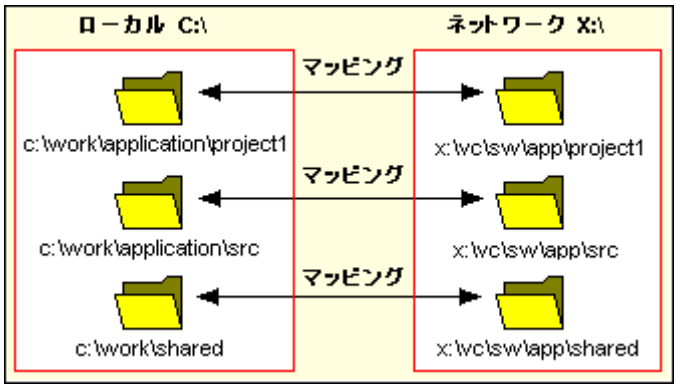
1. [ツール->バージョン管理->構成]を選択してください。[バージョン管理設定]ダイアログボックスを表示します。
2. [プロジェクト]タブを選択してください。
3. [バージョン管理ディレクトリマッピング]リストから変更するマッピングを選択し、[変更]ボタンをクリックしてください。[バージョン管理ディレクトリマッピング]ダイアログボックスを表示します。
4. 2 つのディレクトリを変更して[OK]ボタンをクリックしてください。

既存のマッピングを削除するには

1. [ツール->バージョン管理->構成]を選択してください。[バージョン管理設定]ダイアログボックスを表示します。
2. [プロジェクト]タブを選択してください。
3. [バージョン管理ディレクトリマッピング]リストから削除するマッピングを選択し[削除]ボタンをクリックしてください。

8.7 ディレクトリマッピング例

マッピングを定義すると、Version control directory プレースホルダ$(VCDIR)を使用してプロジェクトファイルを保存するディレクトリを示すことができます。以下に例を示します。ここではネットワークを共有したバージョン管理ドライブ(X:¥)と開発が行われているローカルドライブ(C:¥)からマッピングされた 3 つのディレクトリがあります。



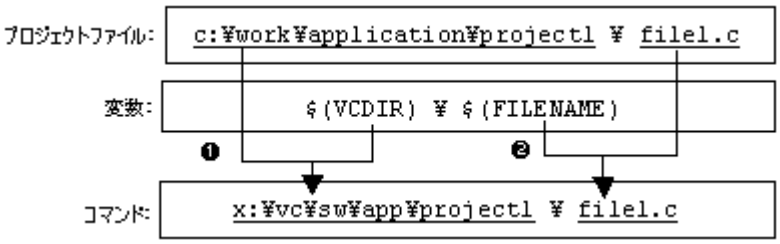
例えば、–GET コマンドが使われていてファイルの読み出し専用コピーを作成するバージョンコントロールのアプリケーションが選ばれているとします。プロジェクトのすべてのファイルを取得するには、次のコマンドを使用します。

-GET “$(VCDIR)¥$(FILENAME)”

そのプロジェクトファイルのコマンドを実行すると、$(VCDIR)をファイルマッピングの中の対応するバージョン管理ディレクトリに置き換えます。

例えば、FILE1.C が C:¥work¥application¥project1¥file1.c にあるとします。FILE1.C に-GET コマンドが適用されると、次のようになります。

1. $(VCDIR)が X:¥vc¥sw¥app¥project1 に置き換えられます。これは C:¥work¥application¥project1 のバージョン管理ディレクトリマッピングに対応しているからです。
2. $(FILENAME)が FILE1.C に置き換えられます。

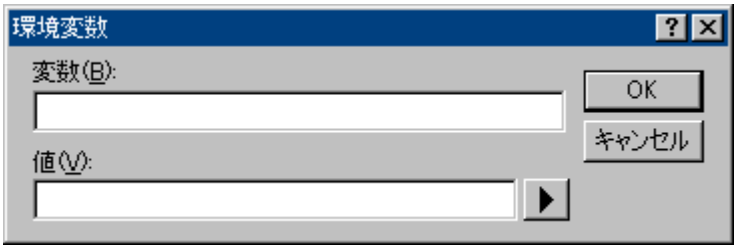


8.8 バージョン管理環境変数を設定する

現在の設定を確認するには[バージョン管理設定]ダイアログボックスの[プロジェクト]タブを表示してください。

新しい環境変数を追加するには

1. [環境]リストの横にある[追加]ボタンをクリックしてください。[環境変数]ダイアログボックスを表示します。



2. [変数]フィールドに環境変数名を入力してください。
3. [値]フィールドに環境変数の値を入力してください。
4. [OK]ボタンをクリックすると新しい変数が[環境]リストに追加されます。

環境変数を変更するには

1. 変更する環境変数を[環境]リストから選択してください。
2. [変更]ボタンをクリックしてください。
3. [変数]フィールドと[値]フィールドを必要に応じて変更してください。
4. [OK]ボタンをクリックすると変更した環境変数がリストに追加されます。

環境変数を削除するには

1. 削除する環境変数を[環境]リストから選択してください。
2. [削除]ボタンをクリックしてください。

8.9 バージョン管理システムの実行を制御する

[バージョン管理設定]ダイアログボックスの[一般]タブでは、使用するバージョン管理システムと HEW を制御できます。また、現在のバージョン管理コンフィグレーションファイルへのフルパスを表示します。

- [コマンド実行前に確認]チェックボックス

このチェックボックスをオンにすると、バージョン管理コマンドを実行する前にダイアログボックスが表示されます。このダイアログボックスではコマンドの実行の有無を確認することができます。[OK]をクリックすると選んだコマンドを実行します。[キャンセル]をクリックするとコマンドは実行しません。

- [DOS ウィンドウで実行]チェックボックス

デフォルトでは、バージョン管理コマンドの出力はアウトプットウィンドウの[Version Control]タブに結果を出力します。

このチェックボックスをオンにすると、各コマンドを DOS ウィンドウで実行します。これは、バージョン管理システムがユーザ入力を要求するメッセージに対して入力が可能になります。実行終了後に DOS ウィンドウを閉じる場合は、[コマンド終了後 DOS ウィンドウを閉じる]チェックボックスをオンにしてください。

- [バージョン管理のディレクトリの区切りに'/'を使用]チェックボックス

デフォルトでは、HEW がプレースホルダ$(VCDIR)を置き換えるときバックスラッシュ文字 '¥'（日本語 Windows®では円記号）を使ってディレクトリを区切ります。しかし、使用するバージョン管理システム(例： Visual SourceSafe)で、ディレクトリを区切るときにスラッシュ文字'/'を使う場合は 、[バージョン管理のディレクトリの区切りに'/'を使用]チェックボックスをオンにしてスラッシュ文字'/'でディレクトリを区切ってください。

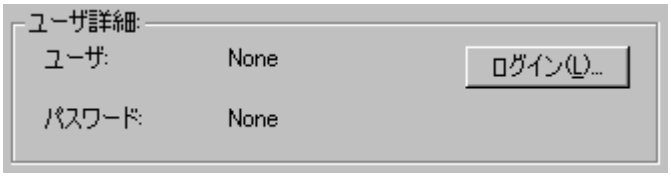
8.10 ユーザ名とパスワードを変更する

バージョン管理ツールでは一般的にユーザ名とパスワードの入力をコマンドラインで行う必要があります。これは、ファイルを保護し、どのファイルがどのユーザによって変更されたか記録するためです。バージョン管理システムでは 2 つのプレースホルダ"User login name", $(USERNAME)と"User login password", $(PASSWORD)をサポートします。コマンドを実行すると、これらのプレースホルダは[バージョン管理設定]ダイアログボックスの[一般]タブの現在の設定に置き換えます。

プレースホルダ$(USERNAME) と $(PASSWORD)に値を設定するには、最初にログインの指定をしておく必要があります。もしこれらのプレースホルダのうちのどちらかのプレースホルダを使うコマンドを実行する前にログイン指定をしていなかった場合、コマンド実行前にログインするよう要求します。

ユーザ名とパスワードを設定するには（ログインの指定）

1. [ツール->バージョン管理->構成]メニューオプションを選択してください。[バージョン管理設定]ダイアログボックスを表示します。
2. [一般]タブを選択してください。



3. [ログイン]ボタンをクリックしてください。[ログイン]ダイアログボックスを表示します。

4. [ユーザ名]フィールドにユーザ名を入力してください。

5. [パスワード]フィールドにパスワードを入力してください。
6. [確認のためパスワード再入力]フィールドにパスワードを再入力してください。
7. [OK]ボタンをクリックすると新しいユーザ名とパスワードを設定します。[パスワード]フィールドと[確認のためパスワード再入力]フィールドで異なるパスワードが入力された場合はもう一度パスワードを入力するよう要求します。

8.11 カスタムバージョン管理システムの使用例

カスタムバージョン管理システムは、MS-DOS コマンドプロンプトからコマンドラインコマンドを実行するバージョン管理システムを対象としています。

ここでは、HEW と下記のバージョン管理システムを接続して使用する例を説明します。

- HEW と RCS を接続して使用する
- HEW と CVS を接続して使用する

8.11.1 HEW と RCS を接続して使用する

HEW と RCS を接続して使用する例を紹介します。

- **RCS の環境**

バージョン管理システムは以下を使用します。

- GNU RCS
- GNU diff

この手順では、上記のバージョン管理システムが"C:¥RCS"にあると仮定します。

- ワークスペース

HEW のワークスペースは、以下のフォルダに作成してください。

C:¥WorkSpace¥rcs_sample

HEW のワークスペース作成後、RCS 管理用フォルダを作成してください。

C:¥WorkSpace¥rcs_sample¥rcs_sample¥RCS

ワークスペースの作成、バージョン管理の設定、バージョン管理機能のチェックイン、比較、チェックアウト、ログの確認の使用手順を説明します。

ステップ 1： ワークスペースを作成する

ステップ 2： カスタムバージョン管理システムを選択する

ステップ 3： バージョン管理を設定する

ステップ 4： バージョン管理機能を使用する（チェックイン）

ステップ 5： バージョン管理機能を使用する（比較）

ステップ 6： バージョン管理機能を使用する（チェックアウト）

ステップ 7： バージョン管理機能を使用する（ログの確認）

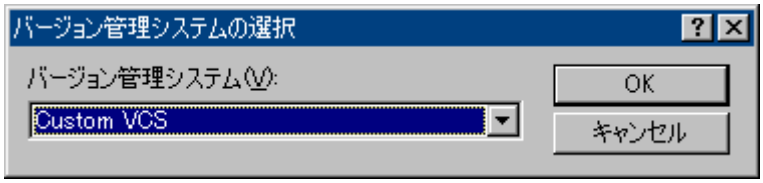
(1) ステップ 1： ワークスペースを作成する

1. [ファイル->新規ワークスペース]を選択し、[新規プロジェクトワークスペース]ダイアログボックスを表示します。
2. 下記を指定しワークスペースを作成してください。
   - ワークスペース名： rcs_sample
   - プロジェクト名： rcs_sample
   - ディレクトリ： C:¥WorkSpace¥rcs_sample
3. ワークスペース作成後、RCS 管理用フォルダを作成してください。

   C:¥WorkSpace¥rcs_sample¥rcs_sample¥RCS

(2) ステップ 2： カスタムバージョン管理システムを選択する

1. [ツール->バージョン管理->選択]を選択します。[バージョン管理システムの選択]ダイアログボックスが表示されます。



2. [バージョン管理システム]リストから"Custom VCS"を選択します。
3. [OK]ボタンをクリックします。

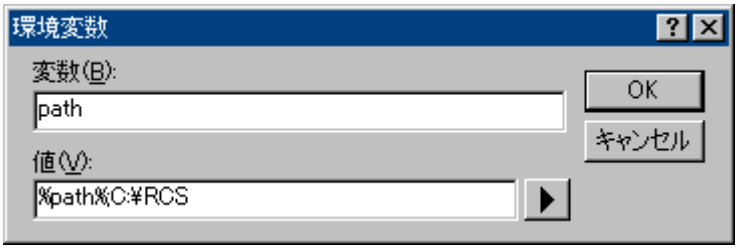
(3) ステップ 3： バージョン管理を設定する

[ツール->バージョン管理->構成]を選択し、[バージョン管理設定]ダイアログボックスを表示します。

- **環境変数を設定する**

1. [プロジェクト]タブを選択します。

2. [環境]にある[追加]ボタンをクリックします。
3. [環境変数]ダイアログボックスが表示されます。



4. [変数]に"path"、[値]に"%path%;C:¥RCS"を設定します。
5. [OK]ボタンをクリックします。
6. 同様に[追加]ボタンをクリックし、[環境変数]ダイアログボックスで下記の環境変数も設定します。

   LOGNAME=user

   TZ=JST-9

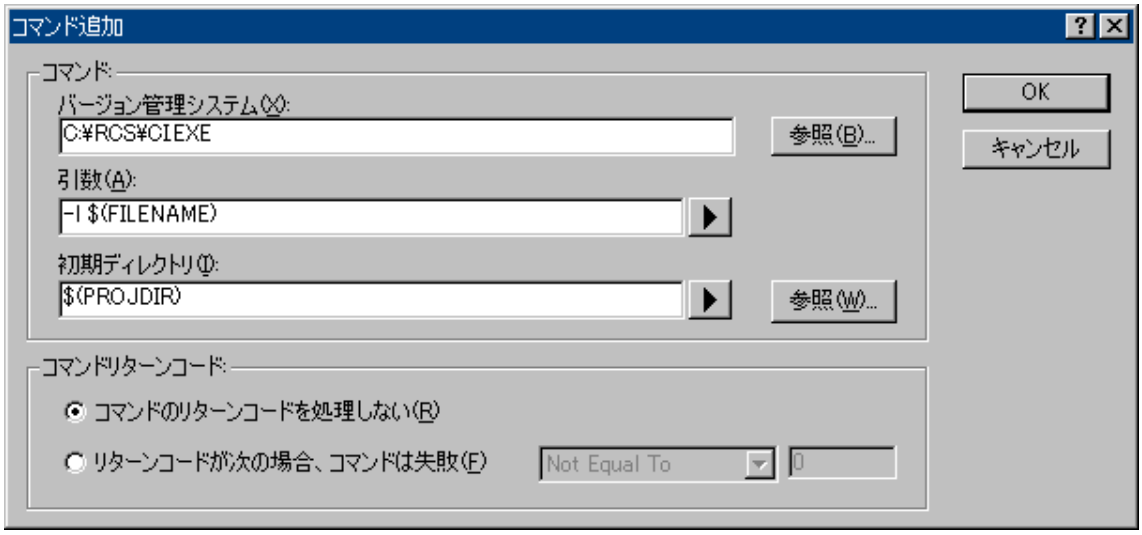
- 実行時の動作を設定する

1. [一般]タブを選択します。

2. [コマンド実行前に確認]チェックボックスをオフにします(コマンド実行前にファイル名を確認したい場合は、オンのままにしてください)。
3. [DOS ウィンドウで実行]チェックボックスをオンにします(RCS のコマンドに MS-DOS コマンドプロンプトから入力する必要があるため)。
4. [コマンド終了後 DOS ウィンドウを閉じる]チェックボックスをオフにします(エラーを MS-DOS コマンドプロンプトで確認するため)。

- コマンドを設定する

1. [コマンド]タブを選択します。
2. [ユーザメニューオプション]にある[追加]ボタンをクリックします。
3. [メニューオプションの追加]ダイアログボックスが表示されます。
4. [オプション]に"Check in"と入力し、[追加]ボタンをクリックします。
5. [コマンド追加]ダイアログボックスが表示されます。



6. [バージョン管理システム]に"C:¥RCS¥CI.EXE"を設定します。
7. [引数]に"-l $(FILENAME)"を設定します。
8. [初期ディレクトリ]に"$(PROJDIR)"を設定します。
9. [OK]ボタンをクリックします。
10. [メニューオプションの追加]ダイアログボックスで[OK]ボタンをクリックします。

ポイント：

バージョン管理によく使用するコマンドのみを登録し、あまり使用しないコマンドは、MS-DOS コマンドプロンプトから使用することをお勧めします。

同様に以下の 3 つのコマンドを設定してください。

コマンド	項目	入力内容
Check out	Version control executable	C:¥RCS¥CO.EXE
	Arguments	-l -f $(FILENAME)
	Initial directory	$(PROJDIR)
Diff	Version control executable	C:¥RCS¥RCSDIFF.EXE
	Arguments	(FILENAME)
	Initial directory	$(PROJDIR)
Log	Version control executable	C:¥RCS¥RLOG.EXE
	Arguments	(FILENAME)
	Initial directory	$(PROJDIR)

これで RCS ツールを使う準備ができました。

(4) ステップ 4： バージョン管理機能を使用する（チェックイン）

1. 後からファイルの変更履歴が分かるように"rcs_sample.c"ファイルの 12 行目にキーワードを入力します。

//Id

2. ワークスペースウィンドウの[Projects]タブの"rcs_sample.c"ファイルを選択し、[ツール->バージョン管理->Check in]を選択します。
3. MS-DOS コマンドプロンプトウィンドウが開きます。
4. プロンプト(">>")に続いてチェックイン時のコメントを入力し、Enter キーを押します。
5. ピリオド". "を 1 文字だけ入力し、Enter キーを押します。

```
rcs_sample.c,v  <--  rcs_sample.c
enter description, terminated with single '.' or end of file:
NOTE: This is NOT the log message!
>> .
initial revision: 1.1
done
C:¥RCS>
```

6. "exit"を入力し Enter キーを押します。MS-DOS コマンドプロンプトウィンドウが閉じます。
7. 表示される HEW の確認ダイアログボックスで[はい]ボタンをクリックしてください。キーワードがファイル名、リビジョン、日時、ユーザなどに変換されて表示します。

ポイント：

MS-DOS コマンドプロンプトウィンドウでコマンドの実行結果を確認しない場合、[バージョン管理設定]ダイアログボックスの[一般]タブの[コマンド終了後 DOS ウィンドウを閉じる]チェックボックスをオンにしてください。

(5) ステップ 5： バージョン管理機能を使用する（比較）

1. チェックインしたファイルと比較するため、"rcs_sample.c"ファイルを編集し上書きします。
2. ワークスペースウィンドウの[Projects]タブの"rcs_sample.c"ファイルを選択し、[ツール->バージョン管理->Diff]を選択します。
3. MS-DOS コマンドプロンプトウィンドウが開き、比較した結果が表示されます。
4. "exit"を入力し Enter キーを押します。MS-DOS コマンドプロンプトウィンドウが閉じます。

```
RCS file: rcs_sample.c,v
Working file: rcs_sample.c
head: 1.1
branch:
locks: strict
        user: 1.1
access list:
symbolic names:
keyword substitution: kv
total revisions: 1;     selected revisions: 1
description:
```

```
----------------------------
revision 1.1    locked by: user;
date: 2006/03/14 07:22:42;  author: user;  state: Exp;
Initial revision
=====================================================================
```

(6) ステップ 6： バージョン管理機能を使用する（チェックアウト）

ファイルで編集した内容を元に戻す場合は、チェックアウトコマンドを実行します。

1. ワークスペースウィンドウの[Projects]タブの"rcs_sample.c"ファイルを選択し、[ツール->バージョン管理->Check out]を選択します。
2. MS-DOS コマンドプロンプトウィンドウが開き、チェックアウトした結果が表示されます。
3. "exit"を入力し Enter キーを押します。MS-DOS コマンドプロンプトウィンドウが閉じます。
4. 表示される HEW の確認ダイアログボックスで[はい]ボタンをクリックしてください。ファイルが元に戻ります。

```
rcs_sample.c,v  <--  rcs_sample.c
enter description, terminated with single '.' or end of file:
NOTE: This is NOT the log message!
>> .
initial revision: 1.1
done
C:¥RCS>
```

(7) ステップ 7： バージョン管理機能を使用する（ログの確認）

編集した履歴を確認するため、ログコマンドを実行します。

1. ワークスペースウィンドウの[Projects]タブの"rcs_sample.c"ファイルを選択し、[ツール->バージョン管理->Log]を選択します。
2. MS-DOS コマンドプロンプトウィンドウが開き、ログが表示されます。
3. "exit"を入力し Enter キーを押します。MS-DOS コマンドプロンプトウィンドウが閉じます。

```
CS file: rcs_sample.c,v
Working file: rcs_sample.c
head: 1.1
branch:
locks: strict
        user: 1.1
access list:
symbolic names:
keyword substitution: kv
total revisions: 1;     selected revisions: 1
```

```
description:
----------------------------
revision 1.1    locked by: user;
date: 2006/03/14 07:22:42;  author: user;  state: Exp;
Initial revision
=============================================================================
```

8.11.2 HEW と CVS を接続して使用する

HEW と CVS を接続して使用する例を紹介します。

- **CVS ツールの環境**

バージョン管理システムは下記を使用します。

o CVS

この手順では、上記のバージョン管理システムが"C:¥cvs-1-11-17"にあると仮定します。

- ワークスペース

HEW のワークスペースは、以下のフォルダに作成してください。

C:¥WorkSpace¥sampleCVS

HEW のワークスペース作成後、CVS 用ソースファイルフォルダを作成してください。

C:¥src¥sampleCVS

注：

HEW と CVS を使用したバージョン管理では、管理対象ファイルをソースファイルのみとしています。ソースファイル以外のファイル（ワークスペースファイル、プロジェクトファイルなど）は、管理対象とすることができませんのでご注意願います。

ワークスペースの作成、バージョン管理の設定、バージョン管理機能のリポジトリの作成、モジュールの登録、チェックアウト、ファイルの編集、ステータスの確認、比較、結果の登録、ログの確認の使用手順を説明します。

ステップ 1： ワークスペースを作成する

ステップ 2： カスタムバージョン管理システムを選択する

ステップ 3： バージョン管理を設定する

ステップ 4： バージョン管理機能を使用する（リポジトリの作成）

ステップ 5： バージョン管理機能を使用する（モジュールの登録）

ステップ 6： バージョン管理機能を使用する（チェックアウト）

ステップ 7： バージョン管理機能を使用する（ファイルの編集）

ステップ 8： バージョン管理機能を使用する（ステータスの確認）

ステップ 9： バージョン管理機能を使用する（比較）

ステップ 10： バージョン管理機能を使用する（結果の登録）

ステップ 11： バージョン管理機能を使用する（ログの確認）

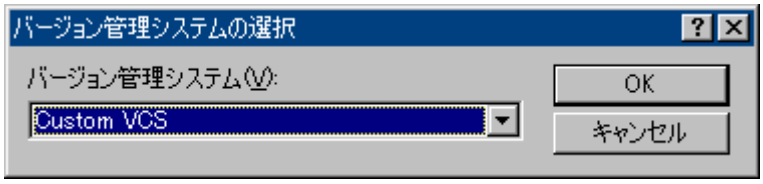
(1) ステップ 1： ワークスペースを作成する

1. [ファイル->新規ワークスペース]を選択し、[新規プロジェクトワークスペース]ダイアログボックスを表示します。
2. 下記を指定しワークスペースを作成してください。
   - ワークスペース名：sampleCVS
   - プロジェクト名：sampleCVS
   - ディレクトリ：C:¥WorkSpace¥sampleCVS
3. HEW を終了します。
4. CVS 用ソースファイルフォルダを作成してください。

   C:¥src¥sampleCVS
5. バージョン管理ツールで管理するファイルを"C:¥WorkSpace¥sampleCVS¥sampleCVS"から"C:¥src¥sampleCVS"へ移動します。
6. HEW を起動し、作成した"sampleCVS"ワークスペースを開きます。
7. [プロジェクト->ファイルの削除]メニューを選択し、[プロジェクトファイルの削除]ダイアログボックスを表示します。
8. バージョン管理するファイルを選択し、[削除]ボタンをクリックします。
9. [OK]ボタンをクリックします。
10. [プロジェクト->ファイルの追加]メニューを選択し、['sampleCVS'プロジェクトにファイルの追加]ダイアログボックスを表示します。
11. バージョン管理するファイルを選択し、[追加]ボタンをクリックします。

(2) ステップ 2： カスタムバージョン管理システムを選択する

1. [ツール->バージョン管理->選択]を選択します。[バージョン管理システムの選択]ダイアログボックスが表示されます。



2. [バージョン管理システム]リストから"Custom VCS"を選択します。
3. [OK]ボタンをクリックします。

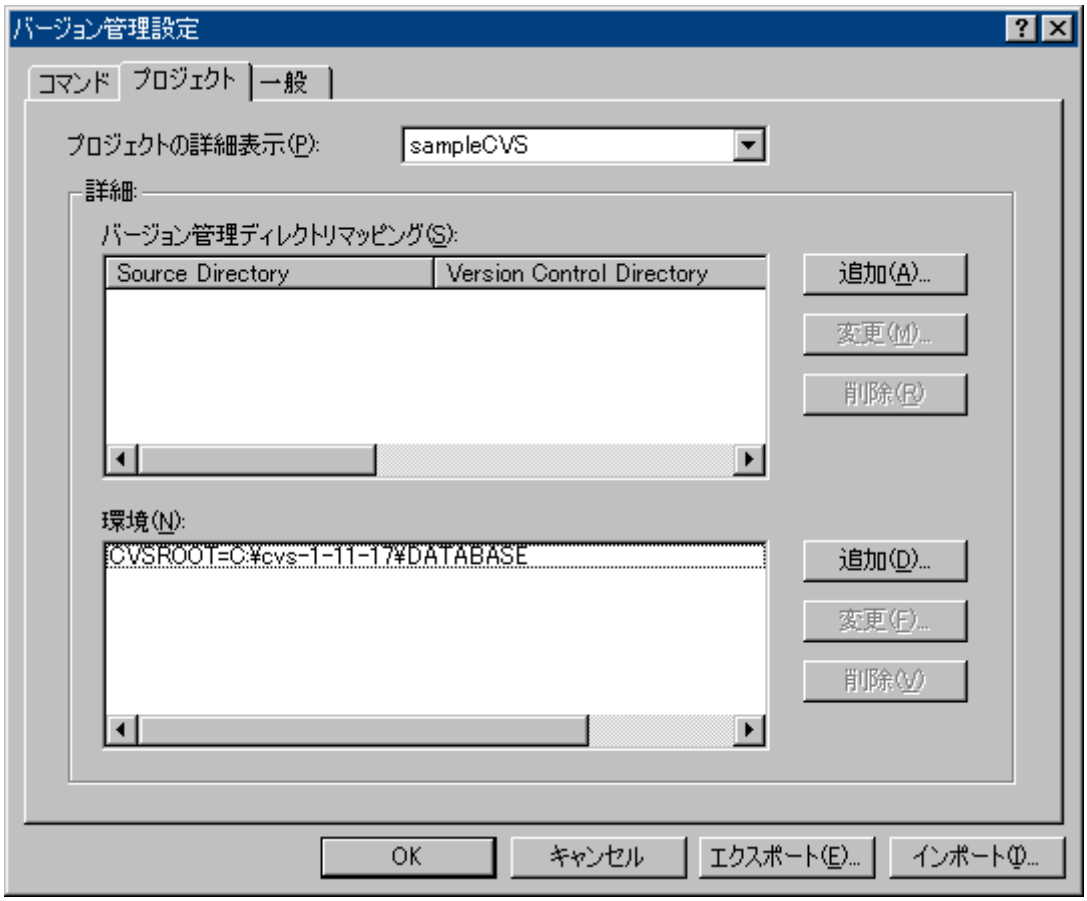
(3) ステップ 3： バージョン管理を設定する

[ツール->バージョン管理->構成]を選択し、[バージョン管理設定]ダイアログボックスを表示します。

- **環境変数を設定する**

1. [プロジェクト]タブを選択します。
2. [環境]にある[追加]ボタンをクリックします。

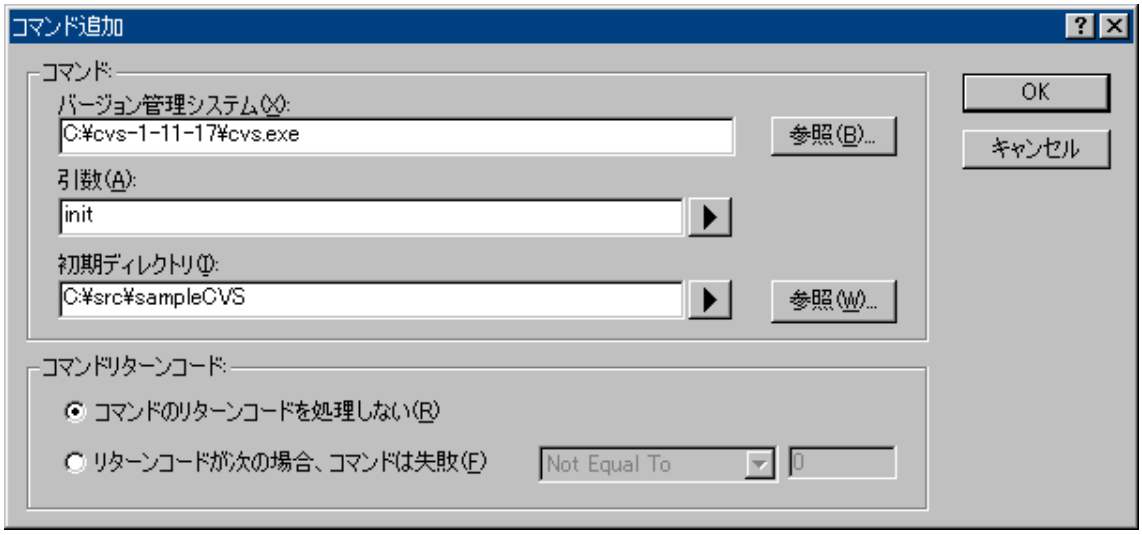
3. [環境変数]ダイアログボックスが表示されます。
4. [変数]に"CVSROOT"、[値]に"C:¥cvs-1-11-17¥DATABASE"を設定します。



5. [OK]ボタンをクリックします。

- コマンドを設定する

1. [コマンド]タブを選択します。
2. [ユーザメニューオプション]にある[追加]ボタンをクリックします。
3. [メニューオプションの追加]ダイアログボックスが表示されます。
4. [オプション]に"Initialize"と入力し、[追加]ボタンをクリックします。
5. [コマンド追加]ダイアログボックスが表示されます。
6. [バージョン管理システム]に"C:¥cvs-1-11-17¥cvs.exe"を設定します。
7. [引数]に"init"を設定します。
8. [初期ディレクトリ]に"C:¥src¥sampleCVS"を設定します。



9. [OK]ボタンをクリックします。
10. [メニューオプションの追加]ダイアログボックスで[OK]ボタンをクリックします。

同様に以下の 6 つのコマンドを設定してください。

コマンド	項目	入力内容
Register	Version control executable	C:¥cvs-1-11-17¥cvs.exe
	Arguments	import -m "New Source" sampleCVS Renesas rel-20060228
	Initial directory	C:¥src¥sampleCVS
Checkout	Version control executable	C:¥cvs-1-11-17¥cvs.exe
	Arguments	checkout sampleCVS
	Initial directory	C:¥src
Status	Version control executable	C:¥cvs-1-11-17¥cvs.exe
	Arguments	status $(FILENAME)
	Initial directory	C:¥src¥sampleCVS
Diff	Version control executable	C:¥cvs-1-11-17¥cvs.exe
	Arguments	diff $(FILENAME)
	Initial directory	C:¥src¥sampleCVS
Commit	Version control executable	C:¥cvs-1-11-17¥cvs.exe
	Arguments	commit -m "modify" $(FILENAME)
	Initial directory	C:¥src¥sampleCVS
Log	Version control executable	C:¥cvs-1-11-17¥cvs.exe
	Arguments	log $(FILENAME)
	Initial directory	C:¥src¥sampleCVS

これで CVS ツールを使う準備ができました。

(4) ステップ 4： バージョン管理機能を使用する（リポジトリの作成）

1. [ツール->バージョン管理->Initialize]メニューを選択します。
2. [Initialize]ダイアログボックスでファイルを選択し[OK]ボタンをクリックします。
3. アウトプットウィンドウの[Version Control]タブに下記のメッセージが表示されます。

   Executing C:¥cvs-1-11-17¥cvs.exe init

   Command(s) completed successfully.

(5) ステップ 5： バージョン管理機能を使用する（モジュールの登録）

1. [ツール->バージョン管理->Register]メニューを選択します。
2. [Register]ダイアログボックスでファイルを選択し[OK]ボタンをクリックします。
3. アウトプットウィンドウの[Version Control]タブに下記のメッセージが表示されます。

   Executing C:¥cvs-1-11-17¥cvs.exe import -m "New Source" sampleCVS Renesas rel-20060228

   N sampleCVS/dbsct.c

   (省略)

   No conflicts created by this import

   Command(s) completed successfully.

(6) ステップ 6： バージョン管理機能を使用する（チェックアウト）

1. [ツール->バージョン管理->Checkout]メニューを選択します。
2. [Checkout]ダイアログボックスでファイルを選択し[OK]ボタンをクリックします。
3. アウトプットウィンドウの[Version Control]タブに下記のメッセージが表示されます。

   Executing C:¥cvs-1-11-17¥cvs.exe checkout sampleCVS

   U sampleCVS/dbsct.c

   (省略)

   cvs.exe checkout: Updating sampleCVS

   Command(s) completed successfully.

(7) ステップ 7： バージョン管理機能を使用する（ファイルの編集）

HEW のエディタでファイルを編集し上書きします。例として"sampleCVS.c"ファイルを修正します。

(8) ステップ 8： バージョン管理機能を使用する（ステータスの確認）

1. "sampleCVS.c"ファイルのステータスを確認します。
2. [ツール->バージョン管理->Status]メニューを選択します。
3. [Status]ダイアログボックスでファイルを選択し[OK]ボタンをクリックします。
4. アウトプットウィンドウの[Version Control]タブに下記のメッセージが表示されます。

```
Executing C:¥cvs-1-11-17¥cvs.exe status sampleCVS.c
===================================================================
File: sampleCVS.c      Status: Locally Modified
   Working revision: 1.1.1.1 Tue Feb 28 02:46:30 2006
   Repository revision: 1.1.1.1 C:¥cvs-1-11-17¥DATABASE/sampleCVS/sampleCVS.c,v
   Sticky Tag: (none)
   Sticky Date: (none)
   Sticky Options: (none)
Command(s) completed successfully.
```

"sampleCVS.c"ファイルのステータスが、"Locally Modified"(修正されている)と確認できます。

(9) ステップ９：バージョン管理機能を使用する（比較）

1. "sampleCVS.c"ファイルの修正内容を確認します。
2. [ツール->バージョン管理->Diff]メニューを選択します。
3. [Diff]ダイアログボックスでファイルを選択し[OK]ボタンをクリックします。
4. アウトプットウィンドウの[Version Control]タブに下記のメッセージが表示されます。

```
Executing C:¥cvs-1-11-17¥cvs.exe diff sampleCVS.c
Index: sampleCVS.c
===================================================================
RCS file: C:¥cvs-1-11-17¥DATABASE/sampleCVS/sampleCVS.c,v
retrieving revision 1.1.1.1
diff -r1.1.1.1 sampleCVS.c
28c28,31
< printf("### Data Input ###¥n");
---
> //無限ループ追加
> while(1)
> {
> printf("### Data Input ###¥n"); //無限ループ開始
49a53,54
> //無限ループ終了
> }
Command(s) completed successfully.
```

"sampleCVS.c"ファイルの修正内容が、確認できます。

(10) ステップ 10： バージョン管理機能を使用する（結果の登録）

1. [ツール->バージョン管理->Commit]メニューを選択します。
2. [Commit]ダイアログボックスでファイルを選択し[OK]ボタンをクリックします。
3. アウトプットウィンドウの[Version Control]タブに下記のメッセージが表示されます。

```
Executing C:¥cvs-1-11-17¥cvs.exe commit -m "modify" sampleCVS.c
Checking in sampleCVS.c;
C:¥cvs-1-11-17¥DATABASE/sampleCVS/sampleCVS.c,v  <--  sampleCVS.c
new revision: 1.2; previous revision: 1.1
done
Command(s) completed successfully.
```

(11) ステップ 11： バージョン管理機能を使用する（ログの確認）

1. [ツール->バージョン管理->Log]メニューを選択します。
2. [Log]ダイアログボックスでファイルを選択し[OK]ボタンをクリックします。
3. アウトプットウィンドウの[Version Control]タブに下記のメッセージが表示されます。

```
Executing C:¥cvs-1-11-17¥cvs.exe log sampleCVS.c
RCS file: C:¥cvs-1-11-17¥DATABASE/sampleCVS/sampleCVS.c,v
Working file: sampleCVS.c
head: 1.2
branch:
locks: strict
access list:
symbolic names:
rel-20060228: 1.1.1.1
Renesas: 1.1.1
keyword substitution: kv
total revisions: 3; selected revisions: 3
description:
----------------------------
revision 1.2
date: 2006/02/28 03:39:22;  author: user;  state: Exp;  lines: +6 -1
modify
----------------------------
revision 1.1
date: 2006/02/28 02:46:30;  author: user;  state: Exp;
branches:  1.1.1;
Initial revision
----------------------------
revision 1.1.1.1
date: 2006/02/28 02:46:30;  author: user;  state: Exp;  lines: +0 -0
```

```
New Source
==================================================================
Command(s) completed successfully.
```

9. Visual SourceSafe の使用

HEW は Visual SourceSafe バージョン管理システムをサポートしています。

Visual SourceSafe バージョン管理システムでは、Visual SourceSafe データベースの中のプロジェクトと HEW のワークスペースの中のプロジェクトを関連付けます。[ツール->バージョン管理]サブメニューからメニューオプションを選択するか、[バージョン管理]ツールバーボタンをクリックすることにより、Visual SourceSafe バージョン管理システムで最もよく使用されるコマンドを容易に起動できます。

Visual SourceSafe のインストールおよびオプションの設定は、Visual SourceSafe のヘルプまたはマニュアルを参照してください。Visual SourceSafe のログイン名、パスワードは設定済みとして説明します。

HEW と Visual SourceSafe バージョン管理システムを接続した場合のバージョン管理の簡単な流れを以下に示します。他に利用できる機能について、詳しくは「9.2 Visual SourceSafe コマンド」を参照してください。

1. **ワークスペースと Visual SourceSafe を関連付ける**

   HEW のワークスペースの中のプロジェクトと Visual SourceSafe データベースの中のプロジェクトを関連付けます。

- "Microsoft Visual SourceSafe 5.0/6.0"を HEW と接続するバージョン管理システムとして選択する。
- Visual SourceSafe へログインする。
- Visual SourceSafe に作成するプロジェクト名、および作成する位置を設定する。

2. **ファイルを追加する**

   HEW のワークスペースで Visual SourceSafe のプロジェクトに追加するファイルを選択します。

3. **ファイルをチェックアウトする**

   HEW のワークスペースで編集するファイルを選択しチェックアウトします。

4. **ファイルをチェックインする**

   HEW のワークスペースで編集したファイルを選択しチェックインします。

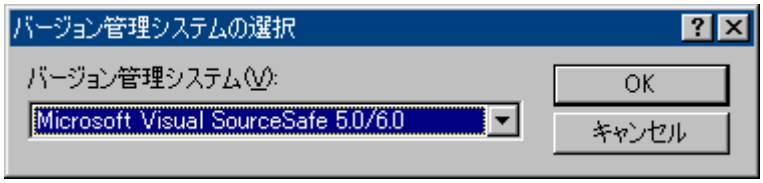
9.1 ワークスペースに Visual SourceSafe を関連付ける

以下の節では、Visual SourceSafe と HEW の現在のワークスペースとを関連付ける方法を説明します。

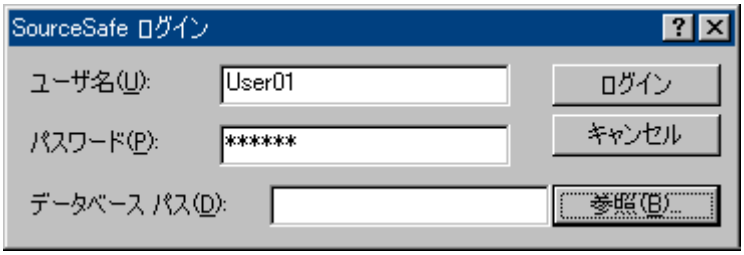
まず、Visual SourceSafe をバージョン管理システムとして選択します。

ワークスペースに Visual SourceSafe を関連付けるには

1. [ツール->バージョン管理->選択]を選択します。[バージョン管理システムの選択]ダイアログボックスが表示されます。
2. [バージョン管理システム]リストから[Microsoft Visual SourceSafe 5.0/6.0]を選択します。



3. [OK] ボタンをクリックします。[SourceSafe ログイン]ダイアログボックスが表示されます。

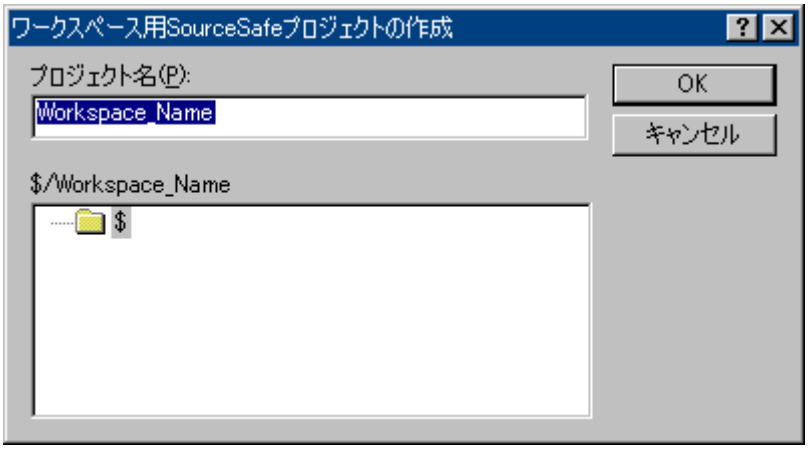


4. [ユーザ名]フィールドに Visual SourceSafe のログイン名を、[パスワード]フィールドにパスワードを入力してください。
5. [データベースパス]フィールドにプロジェクトを追加する Visual SourceSafe データベースファイル（SRCSAFE.INI）を指定します。次のいずれかを行ってください。

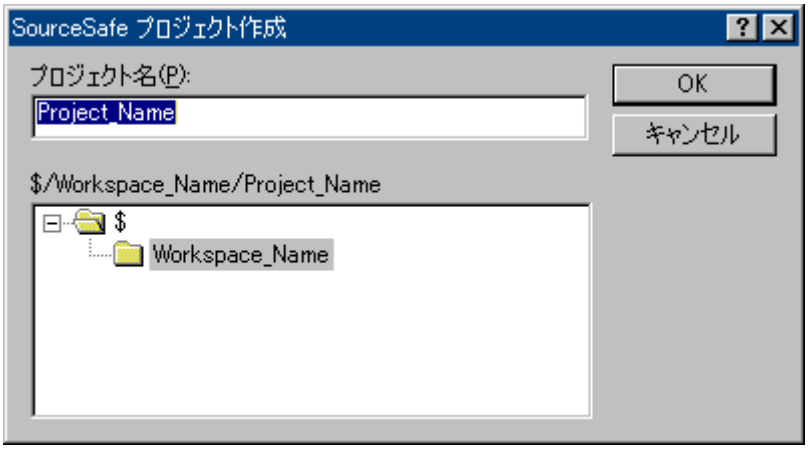
- [参照]ボタンをクリックしてください。[Visual SourceSafe データベースの選択]ダイアログボックスを表示します。プロジェクトを追加する Visual SourceSafe データベースファイルを選択し[選択]ボタンをクリックしてください。

- プロジェクトを追加する Visual SourceSafe データベースファイルへのフルパスを[データベースパス]フィールドに入力してください。

6. [SourceSafe ログイン]ダイアログボックスの[ログイン]ボタンをクリックしてください。[ワークスペース用 SourceSafe プロジェクトの作成]ダイアログボックスを表示します。



7. [プロジェクト名]フィールドに、あらかじめ HEW のワークスペース名が表示されます。データベースに作成するプロジェクト名（フォルダ名）です。プロジェクト名は変更できます。
8. [プロジェクト名]フィールド下のツリーにはステップ 5 で指定したデータベース構造が表示されます。[プロジェクト名]フィールドに指定したフォルダをどのフォルダ内に作成するかを選んでください。
9. [ワークスペース用 SourceSafe プロジェクトの作成]ダイアログボックスの[OK] ボタンをクリックしてください。
10. [SourceSafe プロジェクト作成]ダイアログボックスが表示されます。下の図の例では、HEW のワークスペース"Workspace_Name"がプロジェクト名"Workspace_Name"としてデータベースに追加されました。[プロジェクト名]フィールドには、次に作成する HEW のプロジェクト"Project_Name"があらかじめ表示されています。



11. [SourceSafe プロジェクト作成]ダイアログボックスの[OK] ボタンをクリックしてください。
12. 現在のワークスペースにあるプロジェクトの数だけ、ステップ 10～11 を繰り返してください。

Visual SourceSafe のプロジェクトとサブプロジェクト、HEW のワークスペースとプロジェクト、およびハードディスク上のワークスペースディレクトリの階層イメージを以下の図に示します。

HEW は、Visual SourceSafe 内に必要なプロジェクトを作成し、容易にアクセスできるように、バージョン管理ツールバーやメニューを設定します。

9.2 Visual SourceSafe コマンド

バージョン管理ツールバーやメニューを使用することにより以下の 8 つの操作が使用可能です。

ツール->バージョン管理メニュー	バージョン管理ツールバー	機能
ファイルの追加		Visual SourceSafe にファイルを追加します。
ファイルの削除		Visual SourceSafe からファイルを削除します。
ファイルの取得		Visual SourceSafe からファイルの読み取り専用コピーを取得します。
ファイルのチェックアウト		Visual SourceSafe からファイルの書き込み可能コピーをチェックアウトします。
ファイルのチェックイン		Visual SourceSafe にファイルの書き込み可能コピーをチェックインします。
ファイルの状態		Visual SourceSafe のファイルの状態を表示します。
チェックアウトの取り消し	-	Visual SourceSafe のチェックアウトコマンドを取り消します。
履歴表示	-	Visual SourceSafe のファイル履歴を表示します。

9.2.1 Visual SourceSafe にファイルを追加する

前節では、ハードディスク上のワークスペースディレクトリ（作業中ディレクトリ）と Visual SourceSafe のプロジェクトディレクトリ（Visual SourceSafe によって管理されたディレクトリ）との間でマッピングを確立しました。ハードディスクのプロジェクトディレクトリ（サブディレクトリを含む）に複数のソースファイルがあってもマッピング先の Visual SourceSafe のプロジェクトディレクトリにはファイルがありません。

まず、Visual SourceSafe にファイルを追加します。

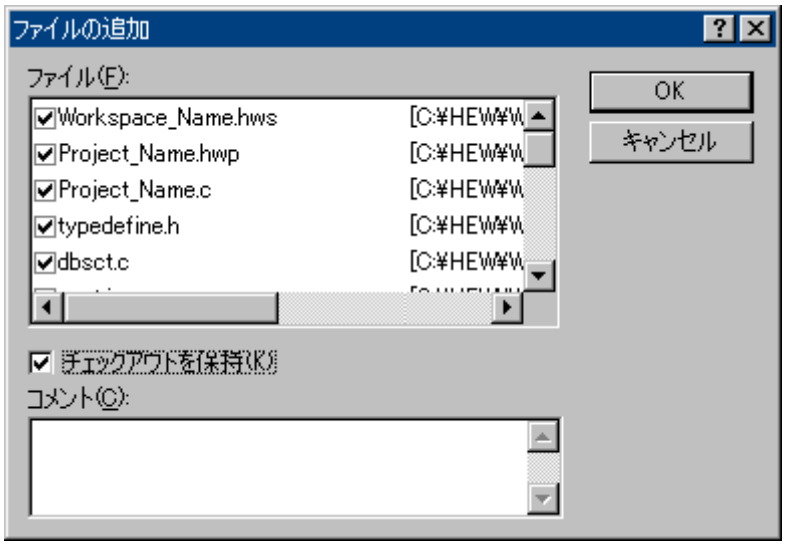
Visual SourceSafe にファイルを追加するには

1. ワークスペースウィンドウの[Projects]タブで、Visual SourceSafe に追加するファイル（複数選択可）を選択してください。ファイルフォルダ、プロジェクトフォルダ、ワークスペースフォルダ、またはそれらを混在して選択することもできます。プロジェクトフォルダ、またはワークスペースフォルダを選択すると、HEW のシステムファイルをファイルリストに追加します。例えば、プロジェクトフォルダを選択すると、プロジェクトファイル（.HWP）が[ファイル]リストへ追加されます。プロジェクトファイルがチェックアウトされ、バージョンが最後にロードしたときより新しい場合、そのプロジェクトを再読み込みしたいかどうかをユーザに問い合わせます。

2. 次のいずれかを行ってください。

- [ファイルの追加]ツールボタン ( ) をクリックする。
- [ツール->バージョン管理->ファイルの追加]メニューオプションを選択する。
- 右クリックしポップアップメニューを表示します。[バージョン管理->ファイルの追加]メニューオプションを選択する。

3. [ファイルの追加]ダイアログボックスが表示されます。追加するファイルの左のチェックボックスをオンにし[OK]ボタンをクリックしてください。



Visual SourceSafe にファイルを追加すると、作業中フォルダのローカルファイルは読み取り専用になります。ファイルを追加する際に[ファイルの追加]ダイアログボックスの[チェックアウトを保持]チェックボックスをオンにすると、ファイルを追加した後もファイルをチェックアウトされた状態で書き込み可能にすることができます。ファイルが追加されたことの確認や、プロジェクト内のすべてのファイル状態を表示するには、以下を行ってください。

1. ワークスペースウィンドウの[Projects]タブで、チェックしたいファイルのプロジェクトフォルダを選択してください。
2. 次のいずれかを行ってください。

- ファイルの状態ツールボタン（ ）をクリックする。
- [ツール->バージョン管理->ファイルの状態]メニューオプションを選択する。
- 右クリックしポップアップメニューを表示します。[バージョン管理->ファイルの状態]メニューオプションを選択する。

3. ファイルの状態の表示先は、アウトプットウィンドウの[Version Control]タブまたは[ファイルの状態]ダイアログボックスです。ファイル状態の表示先の設定は、「9.3 Visual SourceSafe コマンドのオプション」を参照してください。表示された情報により、ファイルがプロジェクトに追加されたかどうか、ファイルがチェックアウトされたかどうか、また、どのユーザによってチェックアウトされたなどの状態も確認できます。

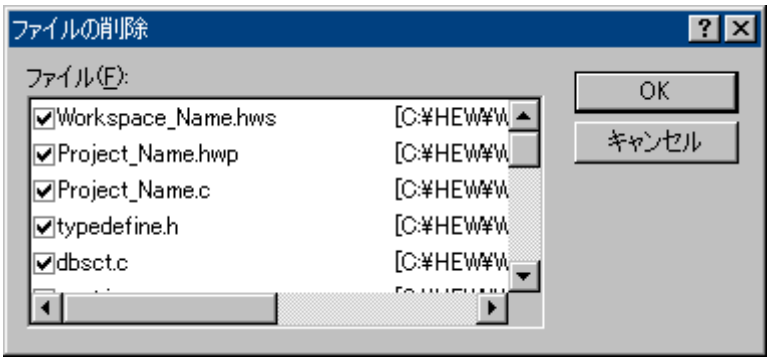
9.2.2 Visual SourceSafe からファイルを削除する

Visual SourceSafe に追加したファイルを Visual SourceSafe による管理から外す場合は、Visual SourceSafe からファイルを削除します。

Visual SourceSafe からファイルを削除するには

1. ワークスペースウィンドウの[Projects]タブで、Visual SourceSafe から削除するファイルを選択してください。ファイルフォルダ、プロジェクトフォルダ、ワークスペースフォルダ、またはこれらを混在させて選択することもできます。
2. 次のいずれかを行ってください。

- [ファイルの削除]ツールバーボタン ( ) をクリックする。
- [ツール->バージョン管理->ファイルの削除]メニューオプションを選択する。
- 右クリックしポップアップメニューを表示します。[バージョン管理->ファイルの削除]メニューオプションを選択する。



3. [ファイルの削除]ダイアログボックスが表示されます。削除するファイルの左のチェックボックスをオンにし[OK]ボタンをクリックしてください。

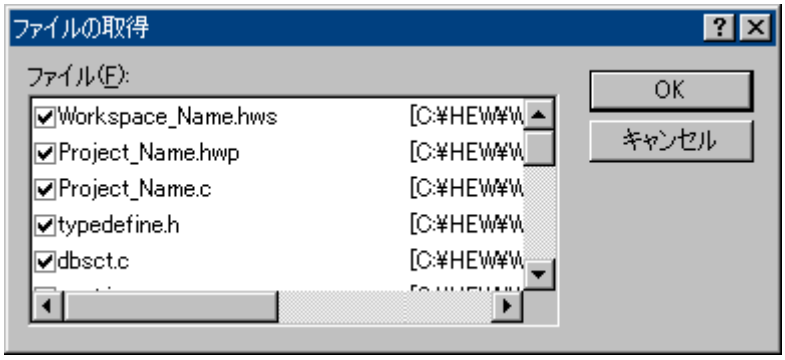
9.2.3 Visual SourceSafe からファイルの読み取り専用コピーを取得する

Visual SourceSafe はソースファイルを保護して、管理しているファイルの書き込み可能なコピーを一度に一人のユーザだけが取得できるようにします。しかし、どのユーザもすべてのファイルの読み取り専用コピーを取得できます。

Visual SourceSafe からファイルの読み取り専用コピーを取得するには

1. ワークスペースウィンドウの[Projects]タブで、Visual SourceSafe から取得するファイルを選択してください。ファイルフォルダ、プロジェクトフォルダ、ワークスペースフォルダ、またはこれらを混在して選択することもできます。
2. 次のいずれかを行ってください。

- [ファイルの取得]ツールバーボタン ( ) をクリックする。
- [ツール->バージョン管理->ファイルの取得]メニューオプションを選択する。
- 右クリックしポップアップメニューを表示します。[バージョン管理->ファイルの取得]メニューオプションを選択する。



3. [ファイルの取得]ダイアログボックスが表示されます。取得するファイルの左のチェックボックスをオンにし[OK]ボタンをクリックしてください。

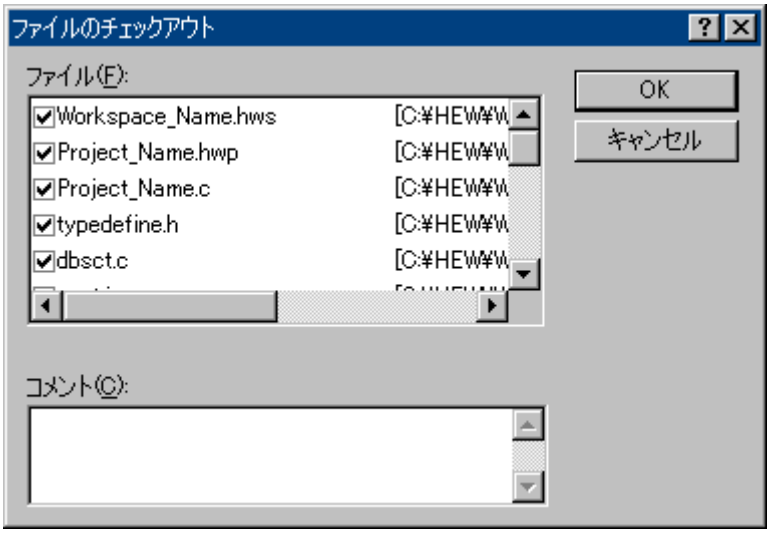
9.2.4 Visual SourceSafe からファイルの書き込み可能コピーをチェックアウトする

Visual SourceSafe はソースファイルを保護して、管理しているファイルの書き込み可能コピーを一度に一人のユーザだけが取得できるようにします。チェックアウト操作をすると、Visual SourceSafe からファイルの書き込み可能コピーをローカルドライブに取得します。これは、チェックアウトしようとするファイルがすでに他のユーザによりチェックアウト（ファイル名の横に青いチェックマーク）されていない場合のみ可能です。

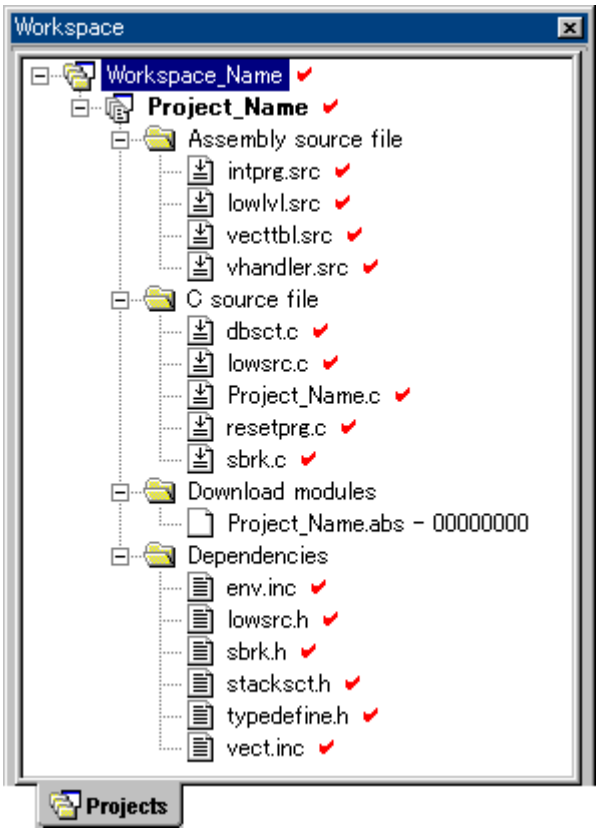
Visual SourceSafe からファイルの書き込み可能コピーをチェックアウトするには

1. ワークスペースウィンドウの[Projects]タブで、Visual SourceSafe からチェックアウトしたいファイルを選択してください。ファイルフォルダ、プロジェクトフォルダ、ワークスペースフォルダ、またはこれらを混在して選択することもできます。
2. 次のいずれかを行ってください。

- [ファイルのチェックアウト]ツールバーボタン ( ) をクリックする。
- [ツール->バージョン管理->ファイルのチェックアウト]メニューオプションを選択する。
- 右クリックしポップアップメニューを表示します。[バージョン管理->ファイルのチェックアウト]メニューオプションを選択する。



3. [ファイルのチェックアウト]ダイアログボックスが表示されます。チェックアウトするファイルの左のチェックボックスをオンにし[OK]ボタンをクリックしてください。
4. 操作が完了すると、ファイル名の横に赤いチェックマークが付き、チェックアウトされたことを示します。



9.2.5 Visual SourceSafe にファイルの書き込み可能コピーをチェックインする

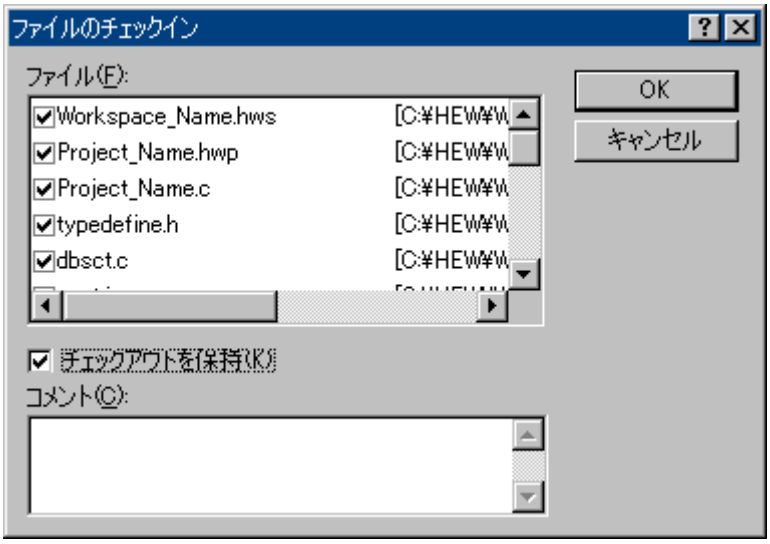
Visual SourceSafe はソースファイルを保護して、管理しているファイルの書き込み可能コピーを一度に一人のユーザだけが取得できるようにします。チェックアウト操作をすると、Visual SourceSafe からファイルの書き込み可能コピーをローカルドライブに取得します。チェックアウトしたファイルを編集してチェックインすると、編集結果を他のユーザが見られるようになります。

編集したファイルを Visual SourceSafe にチェックインするには

1. ワークスペースウィンドウの[Projects]タブで、Visual SourceSafe に再びチェックインするファイルを選択してください。複数のファイルを選択することができます。ファイルフォルダ、プロジェクトフォルダ、ワークスペースフォルダ、またはこれらを混在して選択することもできます。
2. 次のいずれかを行ってください。

- [ファイルのチェックイン]ツールバーボタン ( ) をクリックする。
- [ツール->バージョン管理->ファイルのチェックイン]メニューオプションを選択する。

- 右クリックしポップアップメニューを表示します。[バージョン管理->ファイルのチェックイン]メニューオプションを選択する。



3. [ファイルのチェックイン]ダイアログボックスが表示されます。チェックインするファイルの左のチェックボックスをオンにし[OK]ボタンをクリックしてください。
4. ファイル名の横の赤いチェックマークが外れ、チェックインされたことを示します。ファイルをチェックインする際に[ファイルのチェックイン]ダイアログボックスの[チェックアウトを保持]チェックボックスをオンにすると、ファイルをチェックインした後もファイルをチェックアウトされた状態で書き込み可能にすることができます。

9.2.6 Visual SourceSafe のファイルの状態を表示する

HEW プロジェクトにファイルが表示されても（ワークスペースウィンドウの[Projects]タブ）、ファイルが Visual SourceSafe に管理されているとは限りません。Visual SourceSafe に管理されているファイルのうち、チェックインされたり、チェックアウト（編集するため）されたりするものがあります。ファイルの状態コマンドは、現在のファイルの状態を表示します。

ファイルの状態の表示先は、アウトプットウィンドウの[Version Control]タブまたは[ファイルの状態]ダイアログボックスです。ファイル状態の表示先の設定は、「9.3 Visual SourceSafe コマンドのオプション」を参照してください。

Visual SourceSafe のファイルの状態を表示するには

1. ワークスペースウィンドウの[Projects]タブで、状態を表示するファイルを選択してください。複数のファイルを選択することもできます。ファイルフォルダ、プロジェクトフォルダ、ワークスペースフォルダ、またはこれらを混在して選択することができます。
2. 次のいずれかを行ってください。

- [ファイルの状態]ツールバーボタン ( ) をクリックする。
- [ツール->バージョン管理->ファイルの状態]メニューオプションを選択する。
- 右クリックしポップアップメニューを表示します。[バージョン管理->ファイルの状態]メニューオプションを選択する。

3. アウトプットウィンドウの[Version Control]タブまたは[ファイルの状態]ダイアログボックスにファイルの状態を表示します。

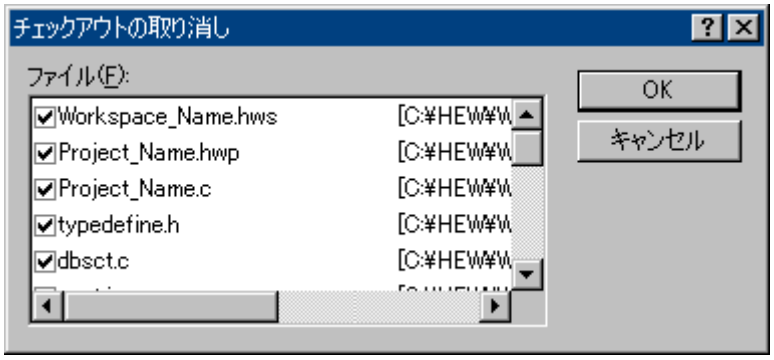
9.2.7 Visual SourceSafe のチェックアウトコマンドを取り消す

Visual SourceSafe はソースファイルを保護して、管理しているファイルの書き込み可能コピーを一度に一人のユーザだけが取得できるようにします。チェックアウト操作をすると、Visual SourceSafe からファイルの書き込み可能コピーをローカルドライブに取得します。チェックアウトしたファイルを編集してチェックインすると、編集結果を他のユーザが見られるようになります。しかし、もしチェックアウト操作を誤って行った場合、または、必要なくなった場合、チェックアウト操作を取り消すことができます。

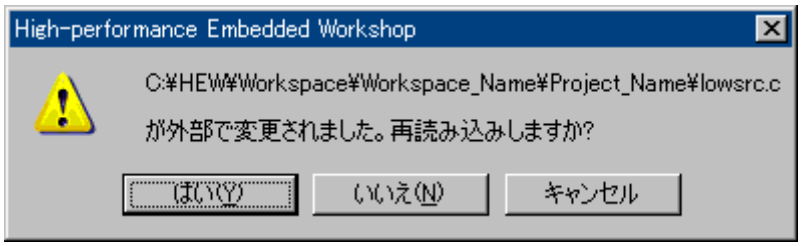
Visual SourceSafe からのファイルのチェックアウト操作を取り消すには

1. ワークスペースウィンドウの[Projects]タブで、以前のチェックアウト操作を取り消したいファイルを選択してください。ファイルフォルダ、プロジェクトフォルダ、ワークスペースフォルダ、またはこれらを混在して選択することもできます。
2. 次のいずれかを行ってください。

- [ツール->バージョン管理->チェックアウトの取り消し]メニューオプションを選択する。
- 右クリックしポップアップメニューを表示します。[バージョン管理->チェックアウトの取り消し]メニューオプションを選択する。



3. [チェックアウトの取り消し]ダイアログボックスが表示されます。チェックアウト操作を取り消すファイルの左のチェックボックスをオンにし[OK]ボタンをクリックしてください。
4. ファイルが変更されていると、以下の確認メッセージボックスが表示されます。
チェックアウト操作を継続する場合は、[キャンセル]ボタンをクリックしてください。
バージョン管理システムから変更前の内容を読み込むには[はい]ボタンをクリックしてください。 [いいえ]ボタンをクリックした場合は、ローカルファイルとバージョン管理システムのファイル内容が不一致になります。



9.2.8 Visual SourceSafe のファイル履歴を表示する

Visual SourceSafe はプロジェクトのファイルへの編集を管理します。ファイルが最初にプロジェクトに追加されたときからの編集内容の完全なファイル履歴を表示できます。

ファイル履歴の表示先は、アウトプットウィンドウの[Version Control]タブまたは[ファイルの履歴]ダイアログボックスです。ファイル履歴の表示先の設定は、「9.3 Visual SourceSafe コマンドのオプション」を参照してください。

Visual SourceSafe のファイル履歴を表示するには

1. ワークスペースウィンドウの[Projects]タブで、ファイル履歴を表示するファイルを選択してください。複数のファイルを選択することができます。ファイルフォルダ、プロジェクトフォルダ、ワークスペースフォルダ、またはこれらを混在してを選択することができます。
2. 次のいずれかを行ってください。

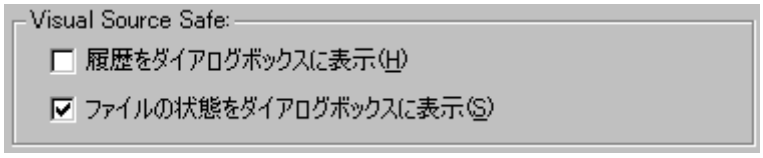
- [ツール->バージョン管理->履歴表示]メニューオプションを選択する。
- 右クリックしポップアップメニューを表示します。[バージョン管理->履歴表示]メニューオプションを選択する。

3. アウトプットウィンドウの[Version Control]タブまたは[ファイルの履歴]ダイアログボックスにファイル履歴を表示します。

9.3 Visual SourceSafe コマンドのオプション

[ツール->バージョン管理->構成]メニューオプションを選択すると、履歴コマンドや状態コマンドの表示方法を変更できます。

1. [ツール->バージョン管理->構成]を選択してください。[バージョン管理設定]ダイアログボックスを表示します。
2. [一般]タブを選択してください。

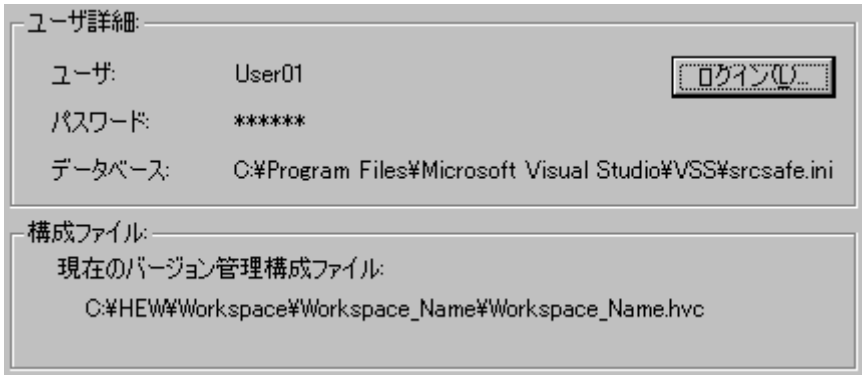


3. 履歴コマンドの結果をダイアログボックスで表示するには、[履歴をダイアログボックスに表示]チェックボックスをオンにしてください。履歴コマンドの結果をアウトプット ウィンドウの[Version Control]タブに表示するには、[履歴をダイアログボックスに表示]チェックボックスをオフにしてください。
4. 状態コマンドの結果をダイアログボックスで表示するには、[ファイルの状態をダイアログボックスに表示]チェックボックスをオンにしてください。状態コマンドの結果をアウトプット ウィンドウの[Version Control]タブに表示するには、[ファイルの状態をダイアログボックスに表示]チェックボックスをオフにしてください。

9.4 バージョン管理設定内容

HEW と Visual SourceSafe を接続して使用する際の一般的な表示、設定を行います。

1. [ツール->バージョン管理->構成]を選択してください。[バージョン管理設定]ダイアログボックスを表示します。
2. [一般]タブを選択してください。



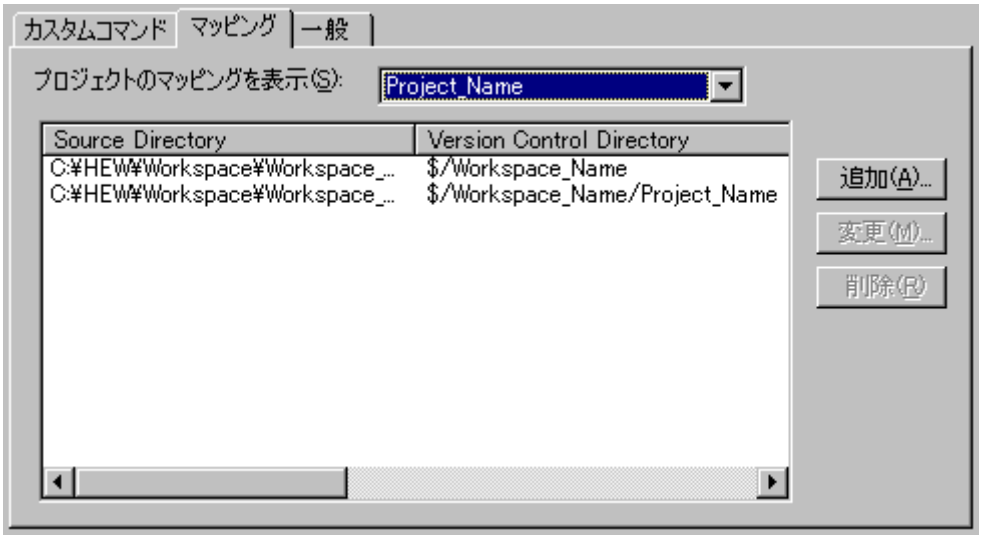
3. [ユーザ詳細]グループボックスに Visual SourceSafe にログインしているユーザ名、パスワード、およびデータベースを表示します。
4. [ログイン]ボタンをクリックすると、ユーザ名、パスワード、およびデータベースを変更できます。
5. [構成ファイル]グループボックスに現在のバージョン管理構成ファイルを表示します。バージョン管理設定の詳細が保存されている構成ファイルです。

9.5 ディレクトリをマッピングする

プロジェクトの現在のマッピング情報を表示します。

[プロジェクトのマッピングを表示]ドロップダウンリストでマッピングを表示するプロジェクトを選択します。リストに HEW のワークスペースおよびプロジェクトのディレクトリ、および Visual SourceSafe のプロジェクトのディレクトリを表示します。プロジェクトのマッピングは、追加、変更、または削除できます。

詳細は、「8.6 ディレクトリをマッピングする」を参照してください。



新しいマッピングを定義するには

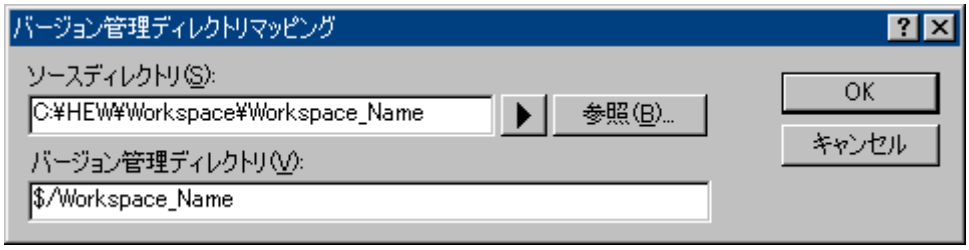
1. [ツール->バージョン管理->構成]を選択してください。[バージョン管理設定]ダイアログボックスを表示します。
2. [マッピング]タブを選択してください。
3. [追加]ボタンをクリックしてください。[バージョン管理ディレクトリマッピング]ダイアログボックスを表示します。
4. [ソースディレクトリ]フィールドに HEW のワークスペースまたはプロジェクトのディレクトリを設定してください。次のいずれかを行ってください。

- ディレクトリを直接入力する。
- プレースホルダボタンをクリックして表示されるメニューから、Project directory、または Workspace directory を選択する。
- [参照]ボタンをクリックして表示される[ソースディレクトリを参照]ダイアログボックスで、ディレクトリを選び[選択]ボタンをクリックする。

5. [バージョン管理ディレクトリ]フィールドにソースディレクトリへマップする Visual SourceSafe のプロジェクトのディレクトリを入力してください。
6. [OK]ボタンをクリックしてください。

既存のマッピングを変更するには

1. [ツール->バージョン管理->構成]を選択してください。[バージョン管理設定]ダイアログボックスを表示します。
2. [マッピング]タブを選択してください。
3. マッピングを変更するリストを選択し、[変更]ボタンをクリックしてください。[バージョン管理ディレクトリマッピング]ダイアログボックスを表示します。



4. [バージョン管理ディレクトリマッピング]ダイアログボックスで新しいマッピングを定義したときと同様の手順で設定し、マッピングを変更してください。
5. [OK]ボタンをクリックしてください。

既存のマッピングを削除するには

1. [ツール->バージョン管理->構成]を選択してください。[バージョン管理設定]ダイアログボックスを表示します。
2. [マッピング]タブを選択してください。
3. マッピングを削除するリストを選択し[削除]ボタンをクリックしてください。

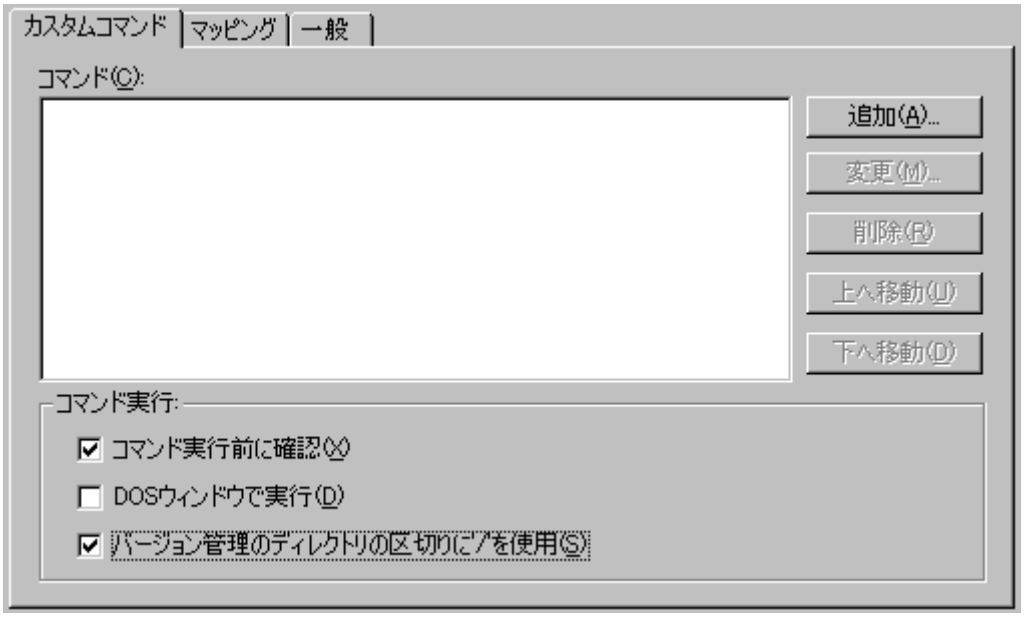
9.6 Visual SourceSafe コマンドを追加する

[バージョン管理設定]ダイアログボックスの[カスタムコマンド]タブでバージョン管理サブメニューにない Visual SourceSafe のコマンドを定義したメニューを追加できます。

このメニューを選択すると定義したコマンドが起動できます。このコマンド実行した際の出力情報はアウトプットウィンドウの[Version Control]タブに表示されます。

[ツール->バージョン管理->構成]を選択し、[バージョン管理設定]ダイアログボックスを表示します。

ここでは、Visual SourceSafe の Properties コマンドの定義を例に説明します。



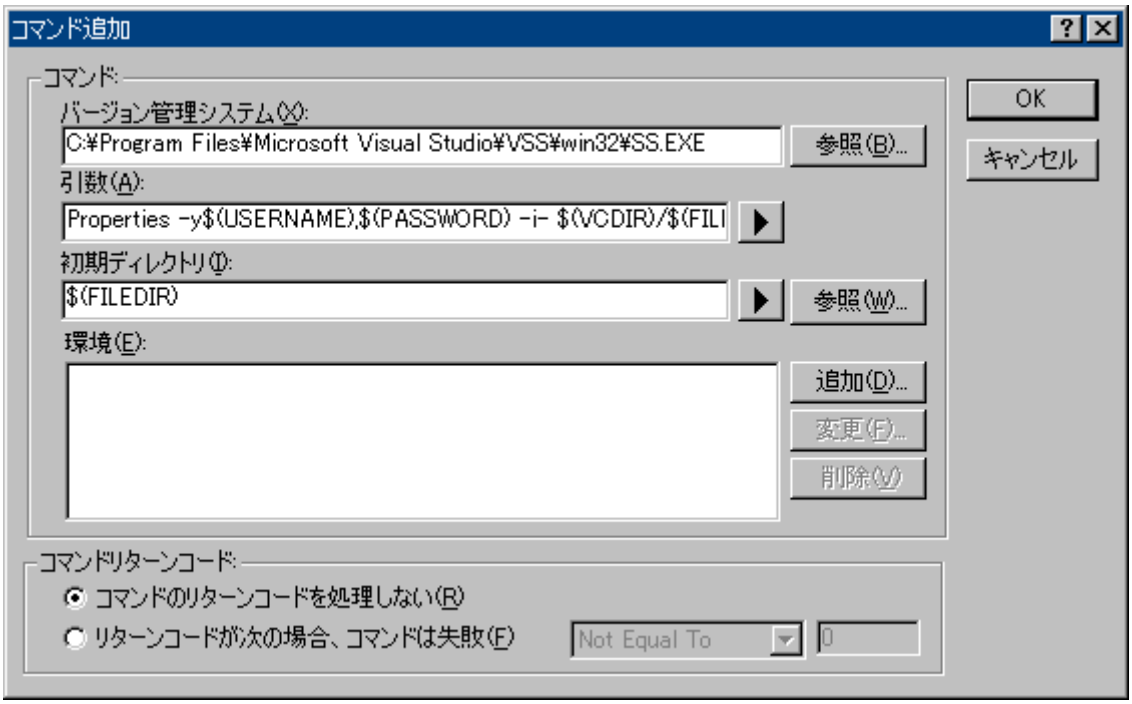
- バージョン管理コマンド実行時の制御

1. [コマンド実行前に確認]チェックボックスをオンにします。
2. [DOS ウィンドウで実行]チェックボックスをオフにします。Visual　SourceSafe はこのフォーマットを使用します。
3. [バージョン管理のディレクトリの区切りに'/'を使用]チェックボックスをオンにします。Visual　SourceSafe はこのフォーマットを使用します。

詳細は、「8.9 バージョン管理システムの実行を制御する」を参照してください。

- コマンド追加

1. [追加]ボタンをクリックします。[メニューオプションの追加]ダイアログボックスが表示されます。
2. [オプション]に"Properties"と入力し、[追加]ボタンをクリックします。[コマンド追加]ダイアログボックスが表示されます。



3. [バージョン管理システム]にバージョン管理システムの実行可能ファイル(SS.EXE)を入力してください。
4. [引数]に"Properties -y$(USERNAME),$(PASSWORD) -i- $(VCDIR)/$(FILENAME)"を設定します。
5. [初期ディレクトリ]に"$(FILEDIR)"を設定します。
6. [OK]ボタンをクリックします。
7. [メニューオプションの追加]ダイアログボックスで[OK]ボタンをクリックします。

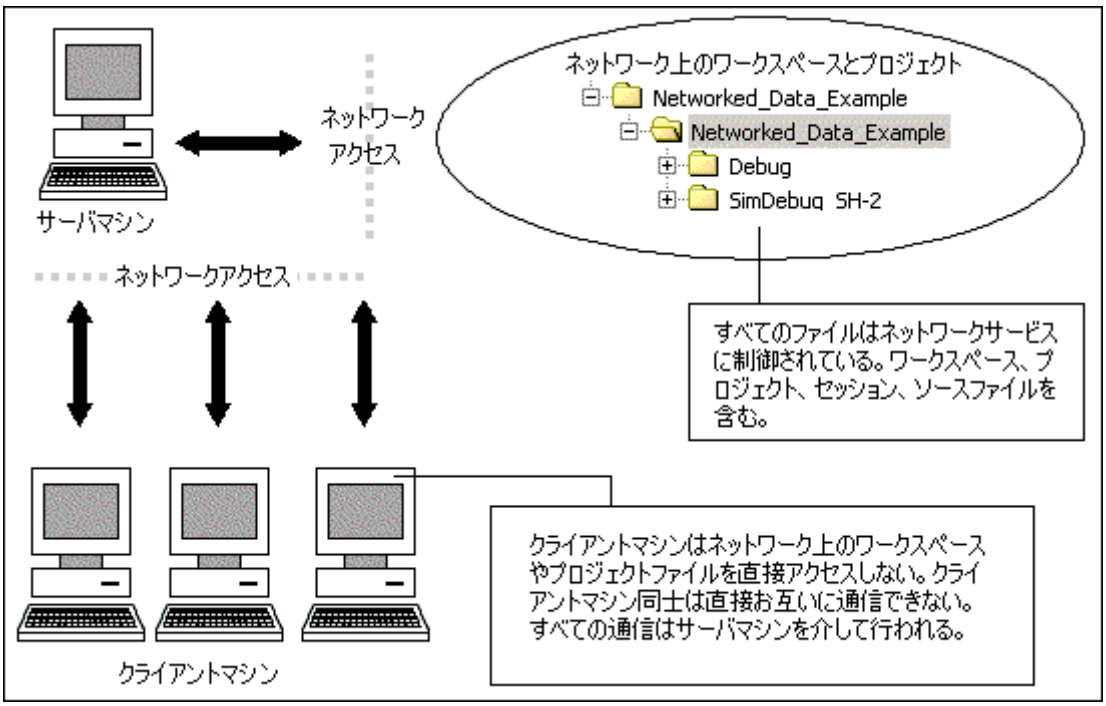
詳細は、「8.2 バージョン管理コマンドを定義する」を参照してください。

- コマンド実行
  1. [ツール->バージョン管理->Properties]メニューオプションを選択します。
  2. [Properties]ダイアログボックスでファイルを選択し[OK]ボタンをクリックします。
  3. アウトプットウィンドウの[Version Control]タブに結果情報が表示されます。

10. ネットワークを利用したプロジェクト共有機能

High-performance Embedded Workshop (HEW) は、ネットワークを介してワークスペースやプロジェクトを共有できます。この機能により、ユーザは共有したプロジェクトを同時に操作し、また、お互いの変更を見ることができます。この機能は、バージョン管理機能とともに使用することができます。バージョン管理機能とこの機能の主な違いは、ユーザがワークスペースやプロジェクトを変更、更新したときに、他のユーザがプロジェクトを再読み込みする必要がないことです。

このシステムは、ネットワーク上の 1 つのコンピュータをサーバマシンとして使うことで実現します。ほかのすべてのクライアントマシンはサーバマシンが提供するサービスを利用できます。例えば、クライアントマシンの 1 つが新規ファイルを追加すると、サーバマシンへファイルの追加が通知されます。すると、サーバマシンはほかのすべてのクライアントマシンにファイルの追加を通知します。この構成を下図に示します。



ネットワーク機能によってユーザはファイルへのアクセス権を与えられます。これによって、プロジェクトアドミニストレータから書き込みを許可されたユーザのみが、プロジェクトやソースファイルを変更できるようになります。アドミニストレータは、各ユーザの書き込み権を、プロジェクトの各自のエリアに制限して、ほかのエリアを読み出し専用にできます。こうして、各ユーザによるプロジェクトのほかの部分への競合や破壊を未然に防きます。これらの制限は異なるレベルに設定できます。

注：

- ある操作をクライアントマシンで実行している間は、他のクライアントマシンでロックされていることがあります。例えば、あるユーザがツールチェインオプションを変更している間、ほかのすべてのユーザはこのオプションを変更できません。
- ネットワーク機能を使うと HEW の性能は低下します。小規模なチームで作業する場合は、この機能を使わずにバージョン管理を使うことを推奨します。

Windows XP Service Pack 2 を搭載した PC での HEW 起動前に実施していただくこと

次節の「10.1 Windows XP Service Pack 2 を搭載した PC で本機能を使う場合の設定」を参照してください。

サーバマシンでのプロジェクトファイル設定手順

1. ネットワーク用のプロジェクトファイルを作成します。
2. [新規プロジェクトワークスペース]ダイアログボックスのディレクトリの指定には、サーバマシンとクライアントマシンから認識できるネットワークパス名を指定してください。
   (例)　¥¥WorkGroup¥Sever1¥ShareDirectory

ワークスペースの共有の仕組み

サーバマシンでのプロジェクトファイル設定手順の 2.で指定したディレクトリは、共有設定をしてください。HEW のネットワーク機能は、サーバマシンにあるソース、オブジェクト、その他のファイルを共有します。これらのファイルは、サーバマシン上のみに存在し、クライアントマシンからはネットワークを介してアクセスします。

クライアントマシンからワークスペースを開く方法

クライアントマシンからネットワーク上のワークスペースを開くには、[ワークスペースを開く]ダイアログボックスでネットワークパスを検索しワークスペースを選択します。

10.1 Windows XP Service Pack 2 を搭載した PC で本機能を使う場合の設定

Windows XP Service Pack 2 でセキュリティ強化機能が搭載されました。

Windows XP Service Pack 2 を搭載した PC の場合、OS の初期状態では HEW のネットワークを利用したプロジェクト共有機能を使用できません。HEW を起動する前に下記のように PC の設定を変更し、サーバマシン、クライアントマシンがネットワークを介して HEW システムへアクセスできるようにしていただく必要があります。サーバマシンおよびクライアントマシンが Windows XP Service Pack 2 を搭載している場合は両方とも設定してください。

- サーバマシンが **Windows XP Service Pack 2** を搭載している場合

ステップ **1**： **Firewall** の設定

1. Windows スタートメニューの[コントロール パネル]を選択します。
2. [コントロール パネル]で[ネットワークとインターネット接続]をクリックします。
3. [ネットワークとインターネット接続]で[Windows ファイアウォール]をクリックします。
4. [Windows ファイアウォール]ダイアログボックスが表示されます。
5. [例外]タブをクリックします。
6. [プログラムの追加]ボタンをクリックし、[プログラムの追加]ダイアログボックスを表示します。
7. [参照]ボタンをクリックし、HEW インストールディレクトリにある

   ¥System¥Sec¥HewServer¥Hew3Server.exe

   を登録します。OK ボタンをクリックし、[プログラムの追加]ダイアログボックスを閉じます。
8. [ポートの追加]ボタンをクリックし、[ポートの追加]ダイアログボックスを表示します。
9. ポート名とポート番号を入力し[TCP]ラジオボタンを選択します。

   名前： DCOM

   ポート番号： 135

   TCP： 選択
10. OK ボタンをクリックし、[ポートの追加]ダイアログボックスを閉じます。
11. [Windows ファイアウォール]ダイアログボックスの OK ボタンをクリックします。
12. [ネットワークとインターネット接続]を閉じます。
13. [コントロール パネル]を閉じます。

ステップ **2**： **DCOM** の設定

1. Windows スタートメニューの[ファイル名を指定して実行]を選択します。
2. [ファイル名を指定して実行]ダイアログボックスが表示されます
3. [名前]フィールドに"dcomcnfg"を入力し OK ボタンをクリックします。
4. [コンポーネントサービス]ウィンドウが表示されます。
5. ツリーを展開します。ツリーの展開時に[Windows セキュリティの重要な警告]ダイアログボックスが表示された場合は、[ブロックする]ボタンをクリックします。

   コンソールルート -> コンポーネントサービス -> コンピュータ -> マイコンピュータ
6. マイコンピュータを右クリックし、表示されるポップアップメニューから[プロパティ]メニューオプションを選択します。
7. [マイコンピュータのプロパティ]ダイアログボックスが表示されます。
8. [COM セキュリティ]タブをクリックします。
9. [アクセス許可]の[制限の編集]ボタンをクリックし、[アクセス許可]ダイアログボックスを表示します。
10. [グループ名またはユーザ名]リストで"ANONYMOUS LOGON"を選択します。

11. [ANONYMOUS LOGON のアクセス許可]リストで[リモートアクセス]の[許可]チェックボックスをオンにします。
12. OK ボタンをクリックし、[アクセス許可]ダイアログボックスを閉じます。
13. [起動とアクティブ化のアクセス許可]の[制限の編集]ボタンをクリックし、[起動許可]ダイアログボックスを表示します。
14. [グループ名またはユーザ名]リストで"Everyone"を選択します。
15. [Everyone のアクセス許可]リストで、[ローカルからの起動]および[リモートからのアクティブ化]の[許可]チェックボックスをオンにします。
16. OK ボタンをクリックし、[起動許可]ダイアログボックスを閉じます。
17. [マイコンピュータのプロパティ]ダイアログボックスの OK ボタンをクリックします。
18. [コンポーネントサービス]ウィンドウを閉じます。

- クライアントマシンが **Windows XP Service Pack 2** を搭載している場合

Firewall の設定

1. Windows スタートメニューの[コントロール パネル]を選択します。
2. [コントロール パネル]で[ネットワークとインターネット接続]をクリックします。
3. [ネットワークとインターネット接続]で[Windows ファイアウォール]をクリックします。
4. [Windows ファイアウォール]ダイアログボックスが表示されます。
5. [例外]タブをクリックします。
6. [プログラムの追加]ボタンをクリックし、[プログラムの追加]ダイアログボックスを表示します。
7. [参照]ボタンをクリックし、HEW インストールディレクトリにある

   Hew2.exe

   を登録します。OK ボタンをクリックし、[プログラムの追加]ダイアログボックスを閉じます。
8. [ポートの追加]ボタンをクリックし、[ポートの追加]ダイアログボックスを表示します。
9. ポート名とポート番号を入力し[TCP]ラジオボタンを選択します。

   名前： DCOM

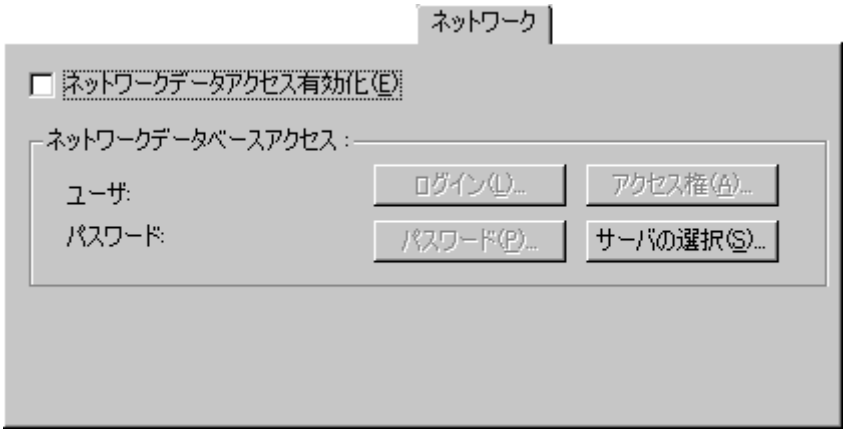
   ポート番号： 135

   TCP： 選択
10. OK ボタンをクリックし、[ポートの追加]ダイアログボックスを閉じます。
11. [Windows ファイアウォール]ダイアログボックスの OK ボタンをクリックします。
12. [ネットワークとインターネット接続]を閉じます。
13. [コントロール パネル]を閉じます。

10.2　ネットワークを利用したプロジェクト共有機能を使う

ネットワークを利用したプロジェクト共有機能を使うには

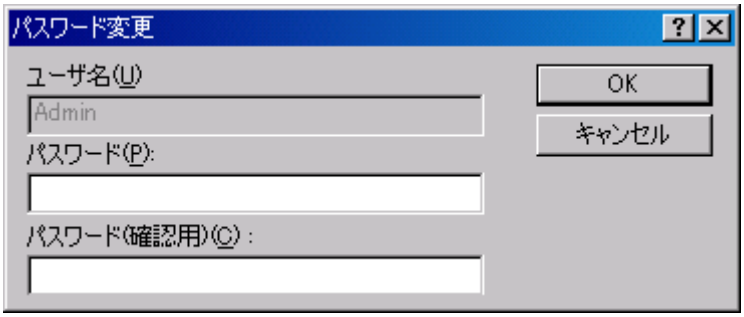
1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [ネットワーク]タブを選択してください。
3. [ネットワークデータアクセス有効化]チェックボックスをオンにしてください。パスワードなしでアドミニストレータをシステムに追加します。アドミニストレータのみが、ユーザを追加したり、ユーザアクセス権を変更したりできます。
4. [ネットワーク]ダイアログボックスで、アドミニストレータはパスワードを設定します。「10.2 アドミニストレータのパスワードを設定する」で説明します。
5. [オプション]ダイアログボックスを閉じます。
6. ダイアログボックスを閉じると、ワークスペースを保存して再び開くかどうか確認されます。これは、共有アクセスモードでワークスペースを再び開く必要があるためです。ワークスペースの変更を保存しないと、変更は無効になります。ネットワークを利用したプロジェクト共有機能を使う場合、[OK]ボタンをクリックしてください。
7. ワークスペースを再び開きログインすると、現在のあなたのアクセス権を示すダイアログボックスを表示します。例えば、アドミニストレータユーザの場合、レベルはアドミニストレータになります。このダイアログボックスを閉じると、HEW サーバウィンドウが開き、ネットワークを利用したプロジェクト共有機能が使えるようになります。



10.3　アドミニストレータのパスワードを設定する

パスワードを設定するには

1. 「10.1 ネットワークを利用したプロジェクト共有機能を使う」の手順の 4 から続けて操作してください。
2. [パスワード]ボタンをクリックしてください。ネットワークデータアクセスを有効にしたとき、このボタンも有効になります。
3. [パスワード変更]ダイアログボックスを表示します。



4. [ユーザ名]フィールドは読み出し専用です。この場合、Admin になります。
5. 両方のフィールドに新規パスワードを入力して、[OK]ボタンをクリックしてください。
6. [オプション]ダイアログボックスの[ネットワーク]タブにユーザ名とパスワードを設定します。

10.4 新規ユーザを追加する

ネットワークデータベースの初期設定では、アドミニストレータユーザとゲストユーザをシステムに追加します。HEW システムでは、以下のアクセス権のレベルを設定できます。

アドミニストレータ	HEW のすべての機能にフルアクセスできます。プロジェクトからユーザを削除したり、アクセス権レベルを変更できます。ワークスペース、プロジェクトファイル、ソースファイルを変更できます。
フルリード/ライトアクセス	ワークスペース、プロジェクトファイル、ソースファイルを変更できます。ユーザのアクセス権レベルは変更できません。
ファイルのみリード/ライトアクセス	ソースファイルのみ変更できます。すべてのプロジェクト設定を表示できますが、変更できません。
リードアクセス	すべてのソースファイルとプロジェクトファイルが表示できます。変更はできません。

どのユーザも、ネットワーク上で使用できるプロジェクトを開くときにはユーザ名とパスワードを入力します。入力しないと、アクセス権は与えられません。入力すると、ユーザには上記のアクセス権レベルのうちのいずれかのレベルを与えられます。

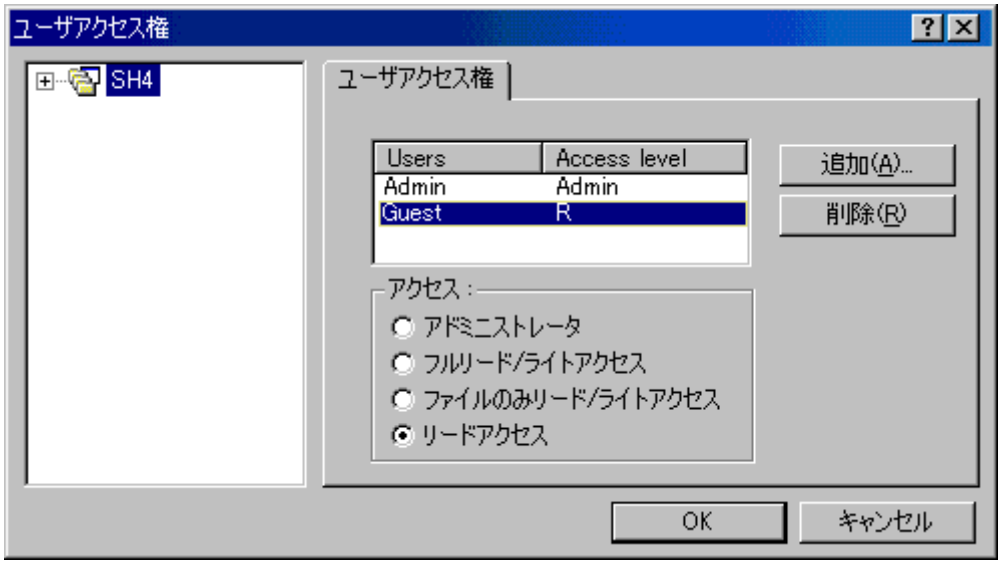
新規ユーザを追加するには

1. アドミニストレータのアクセス権をもつユーザがログインします。この詳細は、上記を参照してください。
2. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
3. [ネットワーク]タブを選択してください。
4. [アクセス権]ボタンをクリックしてください。[ユーザアクセス権]ダイアログボックスを表示します。
5. [追加]ボタンをクリックしてください。[新規ユーザ追加]ダイアログボックスを表示します。あなたをアドミニストレータユーザとして新規のログイン名とパスワードを追加できます。通常、パスワードはデフォルトのテキストか空白のままで設定します。次に、[OK]ボタンをクリックしてください。

6. [OK]ボタンをクリックすると、ユーザをリードアクセスのアクセス権レベルで追加します。アクセス権レベルを変更するには、ユーザ名を選んで必要なラジオボタンをクリックしてください。[OK]ボタンをクリックすると、アクセス権レベルの変更を保存します。

既存ユーザを削除するには

1. アドミニストレータのアクセス権をもつユーザがログインします。この詳細は、上記を参照してください。
2. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
3. [ネットワーク]タブを選択してください。
4. [アクセス権]ボタンをクリックしてください。[ユーザアクセス権]ダイアログボックスを表示します。
5. 削除するユーザ名を選択してください。
6. [削除]ボタンをクリックしてください。
7. [OK]ボタンをクリックするとアクセス権の変更を保存します。



10.5 パスワードを変更する

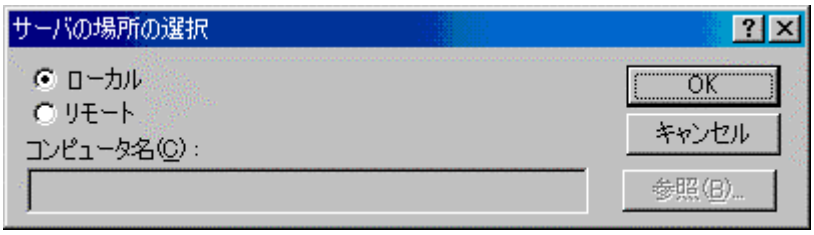
パスワードを変更するには

1. パスワードを変更する HEW ネットワークデータベースにログインして、[基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [ネットワーク]タブを選択してください。
3. [パスワード]ボタンをクリックしてください。
4. 旧パスワードを入力してください。
5. 新しいパスワードを入力して 3 番目のエディットボックスで同じパスワードを入力してください。

6. [OK]ボタンをクリックしてください。
7. 次に、[OK]ボタンをクリックするとパスワードの変更を保存します。

10.6 ネットワークを利用したプロジェクト共有機能を使う場合の留意事項

最初にネットワーク上のプロジェクトを共有するとき、HEW は自動的にサーバマシンを探して接続します。サーバマシンはマシン名で定義します。ワークスペースが、マシン名で見つからないときは、以下に示すダイアログボックスを表示します。サーバマシンの場所を入力するか参照して、[OK]ボタンをクリックしてください。設定するマシンをサーバマシンとして使用する場合は、ラジオボタンをデフォルト設定(ローカル)のままにしてください。



ワークスペースのサーバマシンとして使用していたマシンを他のマシンに接続しようとすると、以下のメッセージを表示します。[OK]ボタンをクリックすると新しいマシンに接続します。

注：

ネットワークを利用したプロジェクト共有機能を有効にした場合、ユーザは複数のワークスペースの中から 1 つのワークスペースにしかアクセスできません。ただし、同一マシン（サーバマシンとクライアントマシンが同じ）の場合、複数のワークスペースにアクセスできます。

11. ファイルの比較

High-performance Embedded Workshop (HEW) にはファイルの比較機能があります。ドライブの 2 つのローカルファイル、またはローカルファイルと Microsoft(R) Visual SourceSafe システムのファイルを比較して相違点を表示することが可能です。

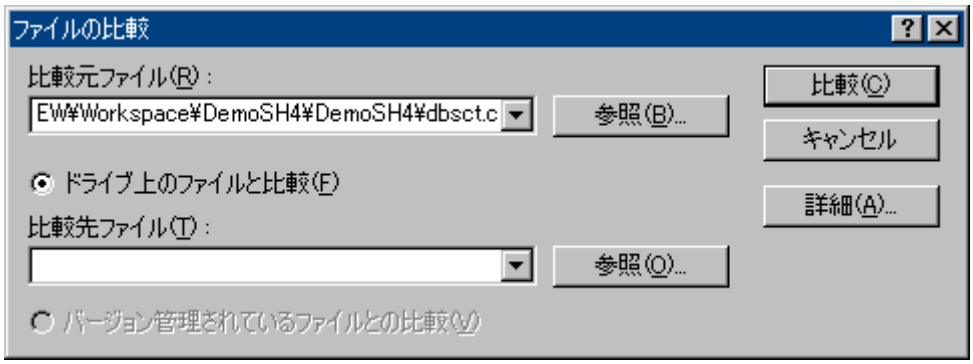
11.1 差分ウィンドウを開く

ローカルドライブの 2 つのファイルを比較するには

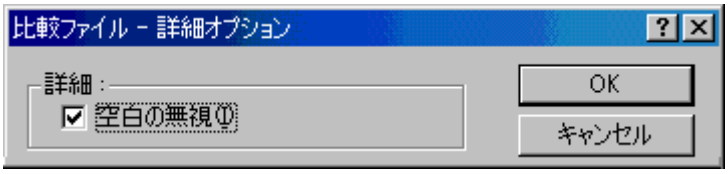
1. 次のいずれかを行ってください。

- ワークスペースウィンドウの[Projects]タブで比較したいファイルを選択してください。選択したファイル上で右クリックしポップアップメニューを表示してください。[ファイルの比較]メニューオプションを選択してください。
- [差分]ツールバーボタン( )をクリックしてください。
- [表示->差分]メニューを選択してください。ウィンドウ内で右クリックしポップアップメニューを表示してください。[比較]メニューオプションを選択してください。

2. [ファイルの比較]ダイアログボックスが表示されます。



3. [ドライブ上のファイルと比較]ラジオボタンが選択されていることを確認してください。
4. 比較する 2 つのファイルを入力してください。1.でワークスペースウィンドウからダイアログボックスを表示した場合は、[比較元ファイル]にファイルが入力されています。ドロップダウンリストボックスから前回指定したファイルを選択するか、[参照]ボタンをクリックしファイルを参照できます。
5. [詳細]ボタンをクリックすると、[比較ファイル-詳細オプション]ダイアログボックスを表示します。空白を考慮せずに相違点の比較ができます。[比較ファイル-詳細オプション]ダイアログボックスを終了するには[OK]ボタンをクリックしてください。



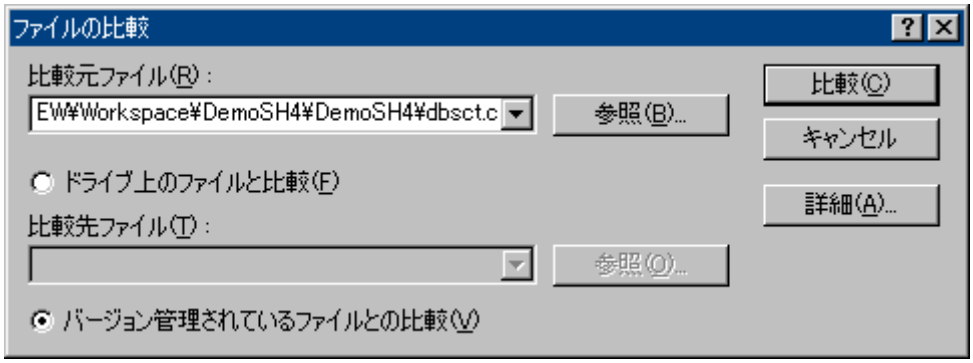
6. [比較]ボタンをクリックしてください。

ローカルファイルと Visual SourceSafe のファイルを比較するには

1. Visual SourceSafe システムと HEW が接続されている必要があります。また、比較するファイルはあらかじめ Visual SourceSafe システムに追加されている必要があります。
2. 次のいずれかを行ってください。

- ワークスペースウィンドウの[Projects]タブで比較したいファイルを選択してください。選択したファイル上で右クリックしポップアップメニューを表示してください。[ファイルの比較]メニューオプションを選択してください。
- [差分]ツールバーボタン( )をクリックしてください。
- [表示->差分]メニューを選択してください。ウィンドウ内で右クリックしポップアップメニューを表示してください。[比較]メニューオプションを選択してください。

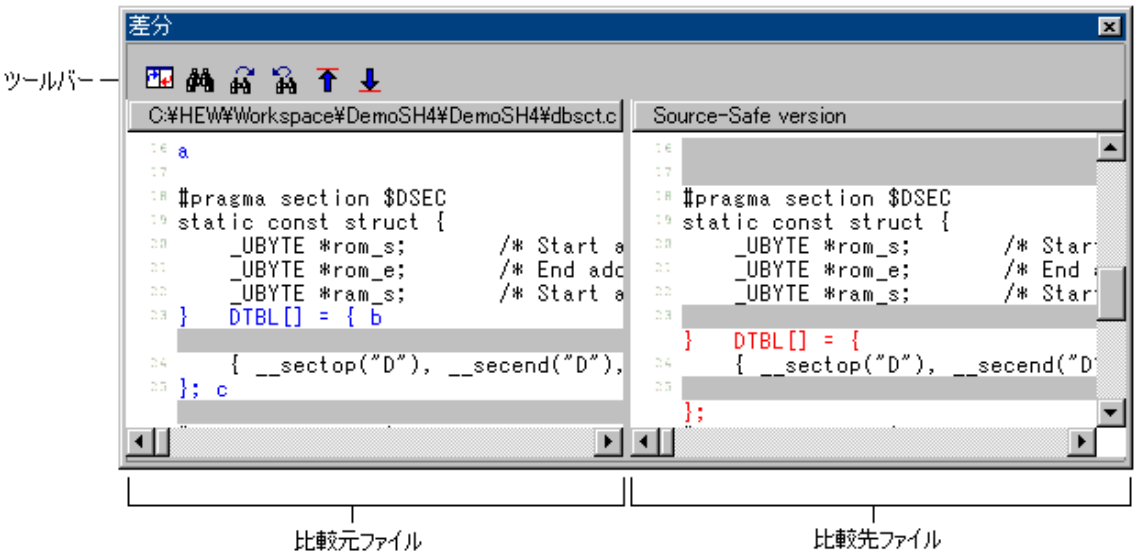
3. [ファイルの比較]ダイアログボックスが表示されます。



4. [バージョン管理されているファイルとの比較]ラジオボタンをクリックしてください。Visual SourceSafe システムと HEW が接続されている場合は、[バージョン管理されているファイルとの比較]ラジオボタンは選択可能になっています。
5. [比較元ファイル]に比較するファイルを入力してください。2.でワークスペースウィンドウからダイアログボックスを表示した場合は、[比較元ファイル]にファイルが入力されています。ドロップダウンリストボックスから前回指定したファイルを選択するか、[参照]ボタンをクリックしファイルを参照できます。
6. [詳細]ボタンをクリックすると[比較ファイル-詳細オプション]ダイアログボックスを表示します。空白を考慮せずに相違点の比較ができます。[比較ファイル-詳細オプション]ダイアログボックスを終了するには[OK]ボタンをクリックしてください。
7. [比較]ボタンをクリックしてください。

[ファイルの比較]ダイアログボックスが閉じ[差分]ウィンドウが開きます。

ウィンドウの構成



- 差分ウィンドウの左ペインに比較元ファイルを表示し、右ペインに比較先ファイルを表示します。
- ファイル名が各ペインの上部に表示されます。
- ペインの境界上にマウスカーソルを置くと左右の矢印のカーソルに変わります。クリックし希望する位置までマウスをドラッグするとペインの横幅を調節できます。

ウィンドウのオプション

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

本ウィンドウは、基本的なオプションをツールバーに割り付けています。

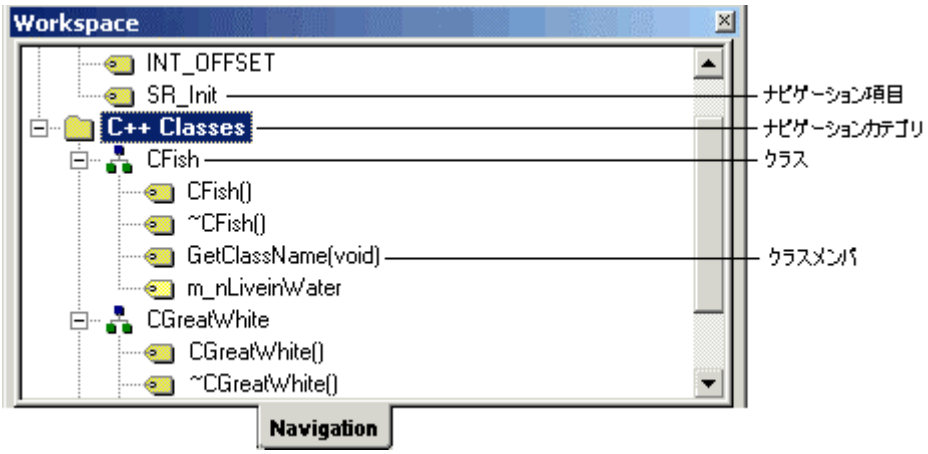
[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	マクロの記録	機能
比較		-	[ファイルの比較]ダイアログボックスを開きます。新たにファイルを比較し相違点を表示できます。
結果のファイル出力	-	-	ダイアログボックスが開き、現在の相違点の結果をテキストフォーマットで保存できます。
空白の無視		-	空白文字を無視します。このメニュー項目は[詳細オプション]ダイアログボックスの[空白の無視]オプションから選択することもできます。
検索		-	標準の[検索]ダイアログボックスを表示します。HEW エディタと同じダイアログボックスです。
前を検索		-	検索項目に当てはまる次の前方の文字列を検索します。
次を検索		-	検索項目に当てはまる次の後方の文字列を検索します。
前の差分		-	次の後方の相違点に自動的にジャンプします。
次の差分		-	次の前方の相違点に自動的にジャンプします。
再比較	-	-	表示をリフレッシュして再び手動で相違点の比較を実行します。この機能は、どちらかのファイルを比較後、変更したときに便利です。

ツールバー表示	-	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ	-	-	ツールバーボタンをカスタマイズします。

12. ナビゲーション機能

High-performance Embedded Workshop (HEW) にはナビゲーション機能があります。サポートされているすべてのナビゲーションの種類をワークスペースウィンドウの[Navigation]タブに表示します。



以下のナビゲーション機能を標準でサポートしています。

ナビゲーションの種類(カテゴリ)	機能
C Defines	C ソースファイルのすべての#define の表示
C Functions	C ソースファイルのすべての ANSI C 標準関数の表示
C++ Classes	C++ソースファイルのすべてのクラス、関数、メンバの表示

デフォルトでナビゲーション項目をカテゴリ単位で表示することができます。各カテゴリの下にアクティブプロジェクトのナビゲーション項目をアルファベット順に表示します。

カテゴリ単位でナビゲーション項目を表示するには

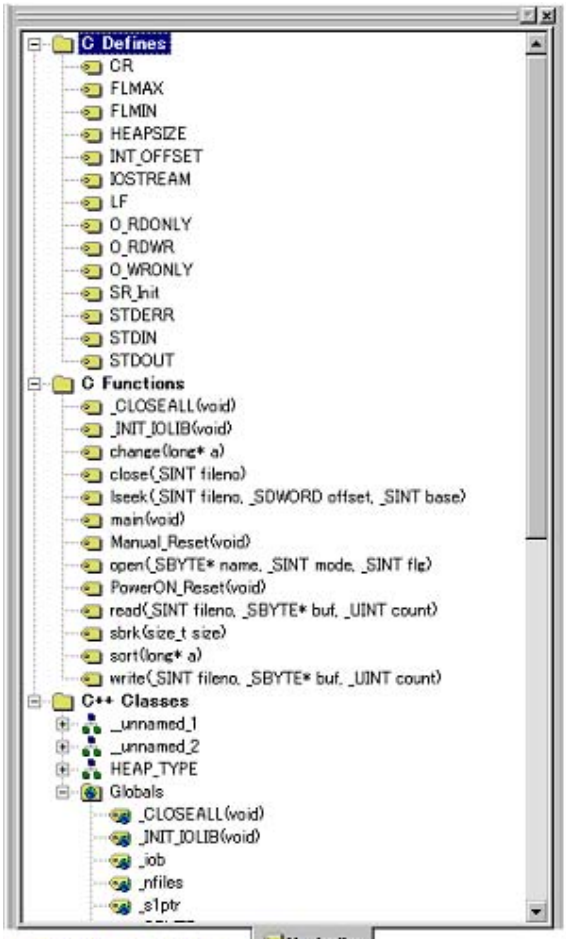
1. [Navigation]タブ内の任意の位置で右クリックしポップアップメニューを表示してください。
2. [ファイルグループ]のチェックをオフにしてください。デフォルトはチェックがオフです。

ナビゲーション項目をファイル単位で表示することができます。アクティブプロジェクト内の各ファイルをツリーで表示し、その下に各ファイルのナビゲーション項目をアルファベット順に表示します。

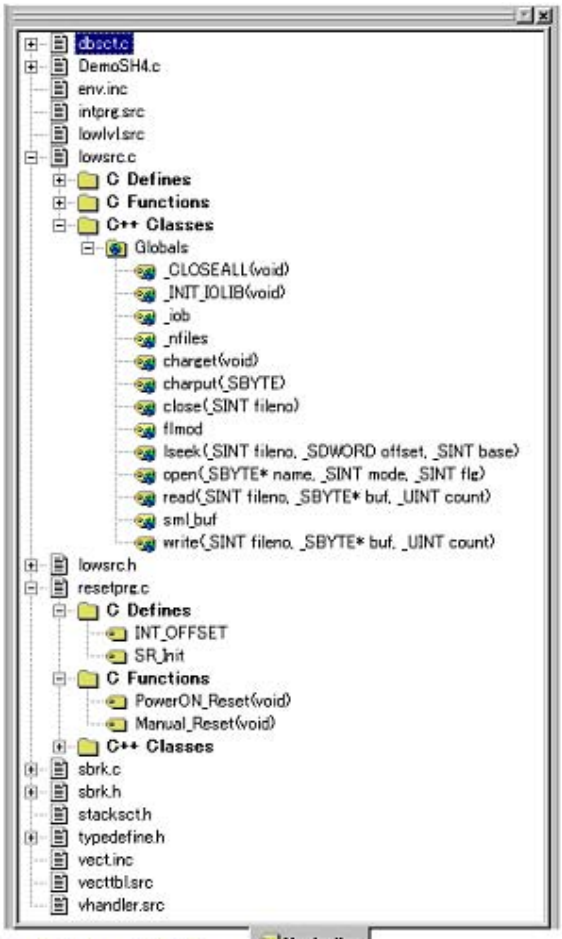
ファイル単位でナビゲーション項目を表示するには

1. [Navigation]タブ内の任意の位置で右クリックしポップアップメニューを表示してください。

2. [ファイルグループ]のチェックをオンにしてください。



ファイルグループ: オフ

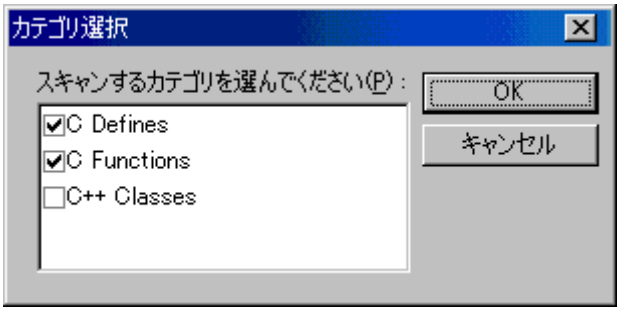


ファイルグループ: オン

特定のナビゲーションカテゴリの情報を必要としない場合はスキャンを無効にできます。

ナビゲーションカテゴリを無効にするには

1. [Navigation]タブ内の任意の位置で右クリックしポップアップメニューを表示してください。
2. [カテゴリの選択]を選択してください。[カテゴリ選択]ダイアログボックスを表示します。



3. 定義を参照する必要のないカテゴリのチェックボックスをオフにしてください。
4. [OK]ボタンをクリックしてください。

ナビゲーション表示を更新するには

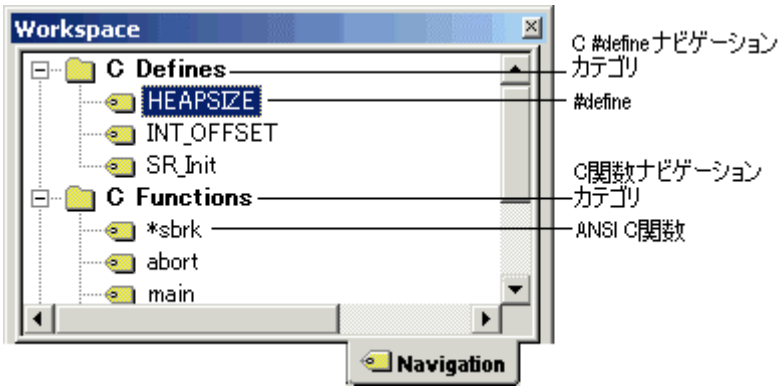
1. [Navigation]タブ内の任意の位置で右クリックしポップアップメニューを表示してください。
2. [リフレッシュ]を選択してください。

注：

- ファイルのスキャン状況に応じて、ナビゲーション項目を表示します。ファイルが多数あると、ナビゲーション表示の更新を終了するのに時間がかかります。
- ファイルを保存する際に再びスキャンします。ファイルを保存するまで、新しいクラスや関数のナビゲーション情報を表示しません。
- [ファイルグループ]と[アクセスグループ]は、同時にオンにできません。一方をオンにするともう一方はオフになります。

12.1 C 関数と#define ナビゲーションの構成

この機能では、ナビゲーション表示に関数と#define 定義を追加します。



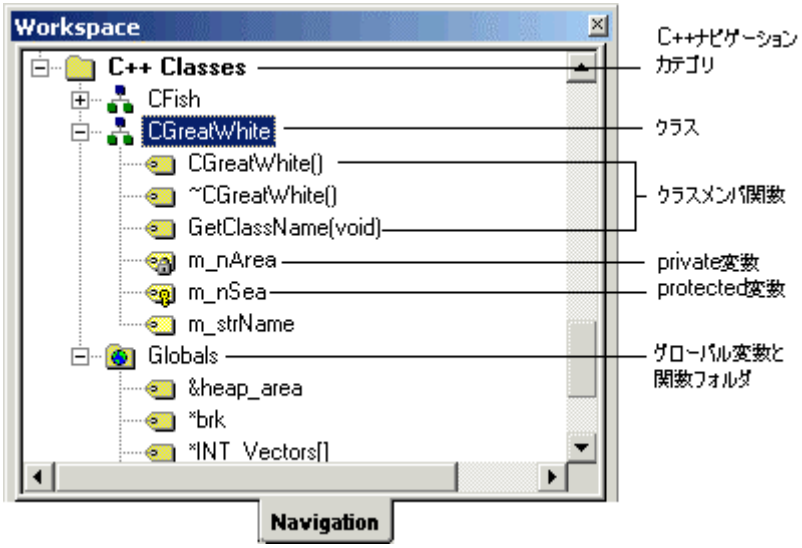
定義へジャンプするには

次のいずれかを行ってください。

- [Navigation]タブの関数または#define 定義項目上でダブルクリックしてください。
- [Navigation]タブの関数または#define 定義項目上で右クリックしポップアップメニューを表示してください。[定義へジャンプ]を選択してください。

12.2　C++ナビゲーションの構成

C++ナビゲーション機能は C++ソースファイルの表示で以下の構成をサポートします。情報の基本構成を下図に示します。



C++ナビゲーション表示では、アイコンの示す関数や変数の種類を示すためにいくつかのアイコンを用います。これらを下図に示します。

アイコン	説明
	public関数
	protected関数
	private関数
	public変数
	protected変数
	private変数

デフォルトで、C++ナビゲーション項目をダブルクリックすると、関連する C++ナビゲーション項目の宣言へジャンプします。このデフォルト動作は、ポップアップメニューの[ダブルクリックで定義へジャンプ]の選択で変更できます。デフォルトはチェックがオフです。チェックマークを付けるとナビゲーション項目をダブルクリックしたときに関連するナビゲーション項目の定義へジャンプします。

定義へジャンプするには

次のいずれかを行ってください。

- [Navigation]タブの C++ナビゲーション項目上で右クリックしポップアップメニューを表示してください。[ダブルクリックで定義へジャンプ]のチェックマークが付いていることを確認してください。[Navigation]タブの C++ナビゲーション項目上でダブルクリックしてください。
- [Navigation]タブの C++ナビゲーション項目上で右クリックしポップアップメニューを表示してください。[定義へジャンプ]を選択してください。

宣言へジャンプするには

次のいずれかを行ってください。

- [Navigation]タブの C++ナビゲーション項目上で右クリックしポップアップメニューを表示してください。[ダブルクリックで定義へジャンプ]のチェックマークが外れていることを確認してください。[Navigation]タブの C++ナビゲーション項目上でダブルクリックしてください。
- [Navigation]タブの C++ナビゲーション項目上で右クリックしポップアップメニューを表示してください。[宣言へジャンプ]を選択してください。

メンバ変数と関数を各々アルファベット順で表示するには

1. [Navigation]タブの C++ナビゲーション項目上で右クリックしポップアップメニューを表示してください。
2. [アクセスグループ]のチェックマークをオフにしてください。デフォルトはチェックがオフです。

public、**private**、**protected** メンバ変数と関数をグループ化して表示するには

1. [Navigation]タブの C++ナビゲーション項目上で右クリックしポップアップメニューを表示してください。
2. [アクセスグループ]のチェックマークをオンにしてください。デフォルトはチェックがオフです。

[ファイルグループ]と[アクセスグループ]は、同時にオンにできません。一方をオンにするともう一方はオフになります。

他の機能としては、特定の範囲の基底クラス、または派生クラスを表示します。

基底クラス、派生クラスを表示するには

1. [Navigation]タブのクラス上で右クリックしポップアップメニューを表示してください。
2. [派生クラスの表示]を選択すると、選択した範囲の派生クラスを表示します。[基底クラスの表示]を選択すると、選択した範囲の基底クラスを表示します。
3. 選択により、拡張ツリー形式で選択されたクラス構造をダイアログボックスに表示します。
4. [閉じる]ボタンをクリックしてください。

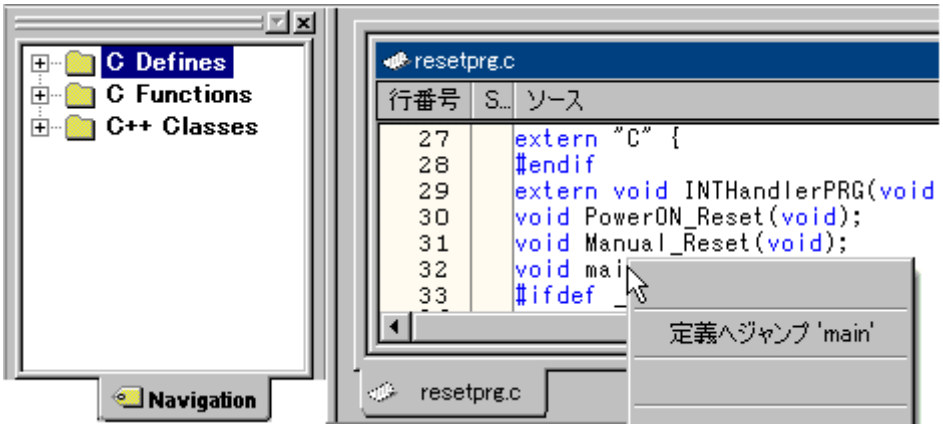
12.3 エディタ内から定義へジャンプする

エディタウィンドウに表示されているソースコードから#define、C 関数、または C++クラスなどのナビゲーション項目名を選択し、それが定義されている位置をエディタウィンドウに表示することができます。

定義位置を表示できるのは、ワークスペースウィンドウの[Navigation]タブで有効なカテゴリのナビゲーション項目です。

定義へジャンプするには

1. エディタウィンドウの[ソース]領域で定義位置を表示したいナビゲーション項目名上で右クリックしてください。
2. 表示されるポップアップメニューから[定義へジャンプ 'ナビゲーション項目名']を選択してください。



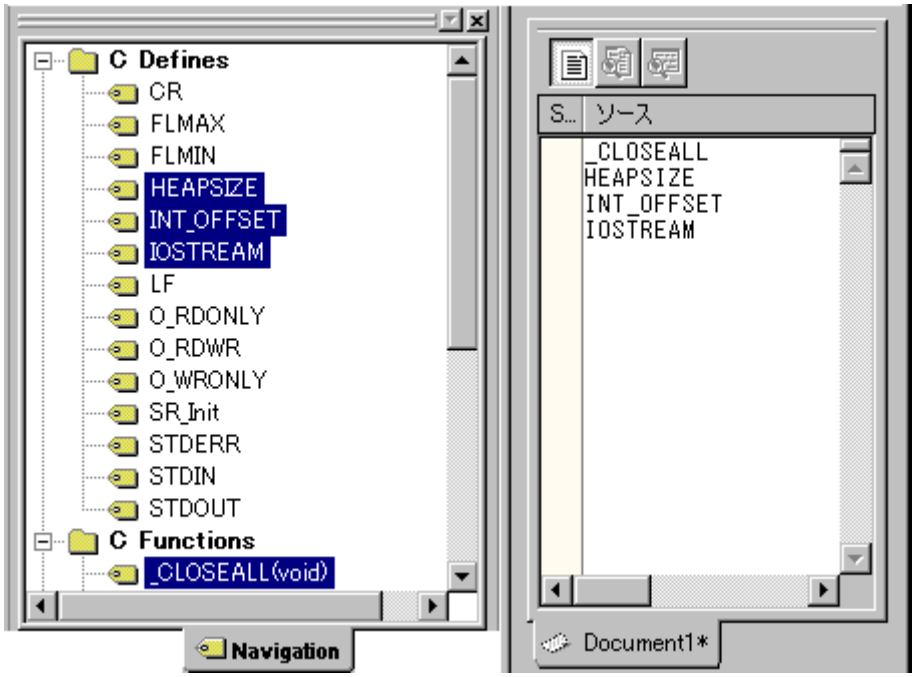
3. 該当するナビゲーション項目が含まれる複数存在する場合は、[ナビゲーションアイテムの選択]ダイアログボックスが表示されます。ファイル名および行番号がリストで表示されます。リスト項目をダブルクリックするか、リスト項目を選択して[OK]ボタンをクリックしてください。

12.4　ナビゲーション項目をドラッグアンドドロップする

ワークスペースウィンドウの[Navigation]タブに表示されている#define、C 関数、または C++クラスなどのナビゲーション項目名をドラッグアンドドロップできます。カテゴリ（C Defines, C Functions, C++ Classes ）、Globals フォルダ、ファイル項目はドラッグできません。

エディタで開いているファイルにナビゲーション項目をドラッグアンドドロップする

ファイルに関連した項目をエディタウィンドウ内にドラッグできるので、それらをタイプする必要がなくコードを書くことを容易にします。



- 複数のアイテムを同時にドラッグできます（別々の行に表示されます）。
- ナビゲーション項目名はすべてアルファベット順に挿入されます。
- 関数をドラッグする場合、関数名だけがドラッグされます。

12.5 スマートエディタ

HEW エディタの機能の 1 つにスマートエディタ機能があります。この機能はデフォルトではすべての C および C++ソースファイルで有効です。C 関数、#define、C++クラスやメンバ関数を使っているとき、HEW エディタは、C 関数、#define、および C++ナビゲーション情報にアクセスしスマートエディタリストをポップアップウィンドウで表示します。

スマートエディタ機能を使用するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [エディタ]タブを選択してください。
3. [Smart-editor 機能]チェックボックスをオンにしてください。
4. [OK]ボタンをクリックしてください。

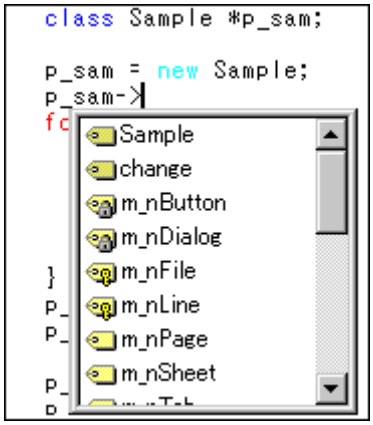
このオプションをオンにすると、操作している C および C++ファイルでスマートエディタ機能が有効になります。

注：

ワークスペースウィンドウの[Navigation]タブで、"C Defines"、"C Functions"、または"C++ Classes" ナビゲーションカテゴリがオフのとき、HEW のスマートエディタ機能は無効です。

HEW エディタを編集するとき、以下のエディタ操作によりスマートエディタリストをポップアップウィンドウで表示します。

- オブジェクトを使用し「.」や「->」でメンバをアクセスしているときは、ポップアップウィンドウが表示され、入力するより効率良くメンバを選択することができます。入力中、ポップアップウィンドウ内は上下矢印キーでメンバを選択できます。Enter キーを押すと、選択したメンバを追加します。「::」を使っているときもこのポップアップメニューを表示します。メンバを追加するまでは、Ctrl+Space キーを押すとポップアップウィンドウを表示できます。



- C および C++関数を使用しているときは、最初に開き括弧を入力すると、ポップアップウィンドウを表示します。このポップアップウィンドウにより、現在のオブジェクトでどの関数が使用できるかわかります。関数を選択すると自動的に残りのパラメータを入力します。閉じ括弧を入力するまでは、Ctrl+Shift+Space キーを押すとポップアップウィンドウを表示できます。

```
class Sample *p_sam;

p_sam = new Sample;
p_sam->change(
for( i=0;   void change(long *a)
     j = r
     if(j < 0){
         j = -j;
     }
     a[i] = j;
}
```

- HEW エディタウィンドウで、C 関数、#define、C++クラス、メンバ関数などを追加したい位置にポップアップウィンドウから選択した項目を追加することができます。次のいずれかを行ってください。
    - HEW エディタウィンドウ内で右クリックしポップアップメニューを表示します。[ナビゲーションリスト]を選択し、表示されるポップアップウィンドウで追加したい項目を選択し Enter キーを押す。
    - Ctrl+Space キーを押し、表示されるポップアップウィンドウで追加したい項目を選択し Enter キーを押す。

13. マップ

リンケージエディタのセクション設定、およびリンケージエディタが出力するリンケージリストファイルの情報からセクション情報とシンボル情報を表示します。下記の機能を備えています。

セクション情報

- アドレスーセクショングループーセクションの順にツリービュー表示
- ドラッグアンドドロップによるセクションの配置の変更が可能
- アドレス、セクショングループ、セクションの追加、変更、削除が可能
- 選択行のアドレスに対応したソースファイルをエディタウィンドウで表示可能
- 表示内容の印刷が可能

シンボル情報

- すべてまたはセクション毎のシンボル情報のリスト表示
- 各シンボル情報の検索、フィルタリング、およびソートが可能
- 選択行のアドレスに対応したソースファイルをエディタウィンドウで表示可能
- 表示内容の印刷が可能

- **対応 CPU**

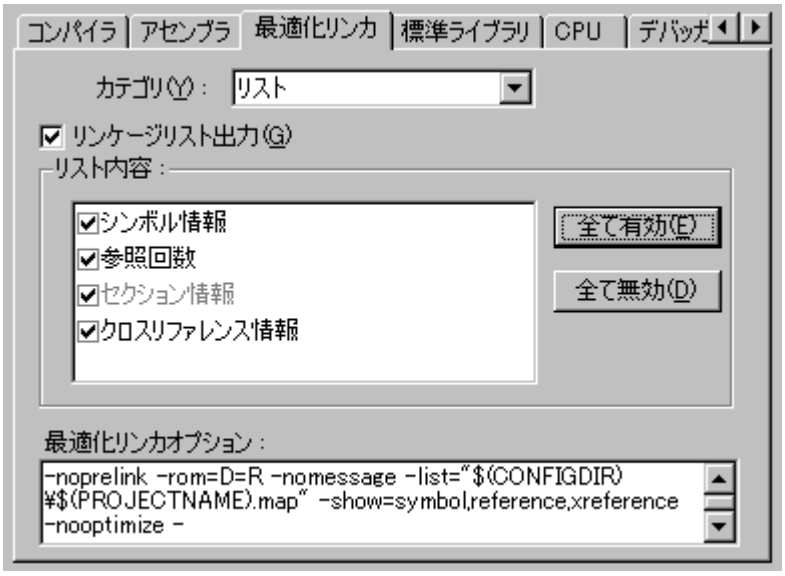
SuperH RISC engine および H8SX, H8S, H8 ファミリのコンパイラパッケージは、HEW のバージョンが V.4.01 以上の場合にマップ機能に対応しています。

M16C、M32C、M32R ファミリは、コンパイラパッケージに同梱される HEW のバージョンが V.4.01 以上の場合にマップ機能に対応しています。

- セクション情報およびシンボル情報を表示するには

セクション情報およびシンボル情報を表示するには、あらかじめ下記を行なう必要があります。

1. [ビルド->xxxxxx Standard Toolchain]を選択し、ツールチェインのオプション設定ダイアログボックスを表示します。
2. [最適化リンカ]タブの[カテゴリ]で”リスト”を選択します。
3. [リンケージリスト出力]チェックボックスをオンにします。これによりセクション情報が出力されます。
4. [全て有効]ボタンをクリックします。これによりシンボル情報が出力されます。
5. [ビルド->ビルド]を選択します。



注：

下記のようにサブコマンドファイルを指定した場合は、セクション情報ウィンドウにリンケージエディタのセクション設定情報は表示されません。

13.1　セクション設定を管理する

Map Section Information ウィンドウを使用すると、セクション情報の追加、編集、削除が GUI 上で簡単に行なえます。

セクション編集後の確認も容易です。

13.1.1 Map Section Information ウィンドウを開く

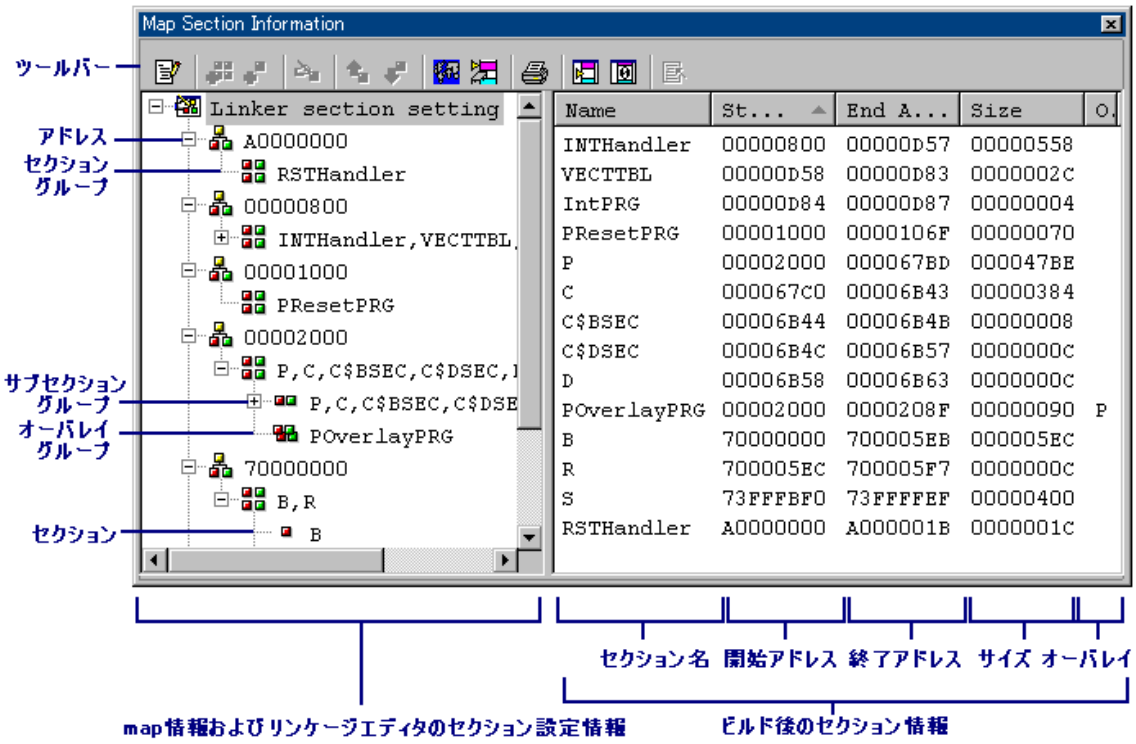
Map Section Information ウィンドウを開くには

1. 次のいずれかを行ってください。

- [マップ]ツールバーボタン（MAP）をクリックする。
- [表示->マップ]を選択する。

2. [マップ種別の選択]ダイアログボックスが表示されます。
3. [マップ種別]ドロップダウンリストから”Map Section Information”を選択します。
4. [OK]ボタンをクリックします。

ウィンドウの構成



- map 情報およびリンケージエディタのセクション設定情報を左側のペインに表示します。
  - リンケージエディタの[セクション]オプションの設定情報が表示されます。
  - 編集モード時、サブセクショングループ、オーバレイグループ、およびセクションは、ドラッグアンドドロップが可能です。異なるセクショングループにドロップした場合は項目の最後に配置されます。セクションは、セクショングループ、サブセクショングループ、オーバレイグループにドロップした場合、ドロップした項目の次に配置されます。
- ビルド後のセクション情報を右側のペインに表示します。
  - リンケージエディタが出力するリンケージリストファイル（.map）がある場合に表示されます。

- カラムヘッダ上のマウスのドラッグで各カラムの幅を調節できます。また、カラムヘッダ上で右クリックするとポップアップメニューを表示します。カラムが利用できる場合は、エントリの横にはチェックマークがあります。エントリをクリックすると、各カラムの表示、非表示を切り替えます。
- リストのカラムヘッダをクリックすると、リスト項目はそのカラムによってソートされます。ウィンドウを表示すると、"Start Address"の昇順でソートされています。
- ”Overlay”カラムにはオーバレイグループのプライマリセクション名が表示されます。

ウィンドウのオプション

- **map 情報およびリンケージエディタのセクション設定情報（左側のペイン）**

左側のペイン内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

本ウィンドウは、基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	機能
セクション情報編集モード		編集モードを切り替えます。
セクショングループの追加		セクショングループを追加します。
セクションの追加		セクションを追加します。
オーバレイグループの追加 ***1**		オーバレイグループを追加します。
選択項目の編集		選択項目を編集します。
切り取り		セクション情報を切り取ります。
コピー		セクション情報をコピーします。
貼り付け		セクション情報を貼り付けます。
削除		セクション情報を削除します。
1 つ上へ		セクション情報を 1 つ上へ移動します。
1 つ下へ		セクション情報を 1 つ下へ移動します。
プライマリセクションに設定		プライマリセクションに設定します。
メモリマップ設定 ***2**		メモリマップを設定するダイアログボックスを表示します。
メモリリソース自動確保 ***2**		メモリリソースを自動的に確保します。
印刷		セクション設定ツリーを印刷します。
ツールバー表示	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ	-	ツールバーボタンをカスタマイズします。

***1.** リンケージエディタがオーバレイをサポートしていない場合は使用できません。

***2.** 機能のサポートは、デバッギングプラットフォームに依存します。HEW V.4.02 では、SuperH RISC engine および H8SX, H8S, H8 ファミリ用のシミュレータデバッガがサポートしています。

- ビルド後のセクション設定情報（右側のペイン）

右側のペイン内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

本ウィンドウは、基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	機能
未割り当て領域の表示		未割り当て領域を表示します。
サイズ 0 のセクション表示		サイズ 0 のセクションを表示します。
ソースファイル表示		選択行のアドレスに対応したソースコードを表示します。
印刷		セクションリストを印刷します。

13.1.2 セクション情報編集モードの開始と終了

左側のペインに表示されるセクション情報を編集するには、編集モードに切り替える必要があります。デフォルトは編集モードではありません。

編集モードに移行するには

1. 左側のペイン内で右クリックしポップアップメニューを表示します。
2. [セクション情報編集モード]を選択します。これにより編集モードに移行します。
3. [セクション情報編集モード]にチェックマークが付きます。

編集モードに移行すると、下記のオプションが使用可能になります。

左側のペインの各項目上で右クリックすると下記の”✔”を記載したオプションが使用可能です。

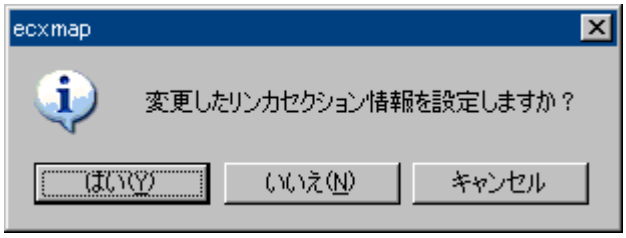
ポップアップメニュー項目	ツールバーボタン	項目					
		ルート	アドレス	セクショングループ	サブセクショングループ	オーバレイグループ	セクション
セクショングループの追加		✔	✔	✔	✔	✔	✔
セクションの追加			✔	✔	✔	✔	✔
オーバレイグループの追加 ***1**				✔	✔		✔ ***2**
選択項目の編集			✔	✔	✔	✔	✔
切り取り					✔	✔	✔
コピー					✔	✔	✔
貼り付け			✔	✔	✔	✔	✔
削除			✔	✔	✔	✔	✔
1 つ上へ							✔
1 つ下へ							✔
プライマリセクションに設定 ***1**						✔	

***1.** リンケージエディタがオーバレイをサポートしていない場合は使用できません。

***2**. SuperH RISC engine ファミリ用 C/C++コンパイラパッケージ V.9.00 Release 04、および H8, H8S, および H8SX ファミリ用 C/C++コンパイラパッケージ V.6.01 Release 02 以降のバージョンでサポートされます。

編集モードを終了するには

1. 左側のペイン内で右クリックしポップアップメニューを表示します。
2. [セクション情報編集モード]を選択します。
3. 下記の確認ダイアログボックスが表示されます。編集した結果をリンケージエディタのセクション設定に反映するには「はい」を選択してください。



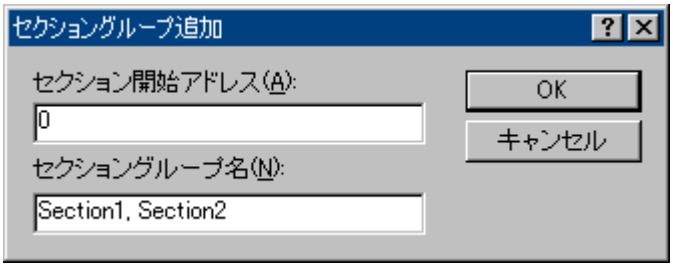
4. これにより編集モードが終了します。[セクション情報編集モード]のチェックマークが外れます。

13.1.3 セクショングループを追加する

編集モードでは、セクショングループの追加が可能です。

セクショングループを追加するには

1. 左側のペイン内で右クリックしポップアップメニューを表示します。
2. [セクショングループの追加]を選択します。
3. [セクショングループ追加]ダイアログボックスが表示されます。



4. [セクション開始アドレス]にセクション開始アドレスを入力します。
5. [セクショングループ名]にセクショングループ名を入力します。セクショングループ名をコンマで区切って指定すると、各々がセクションとして追加されます。上の図の例を設定すると、セクション"Section1"、およびセクション"Section2"が追加されます。

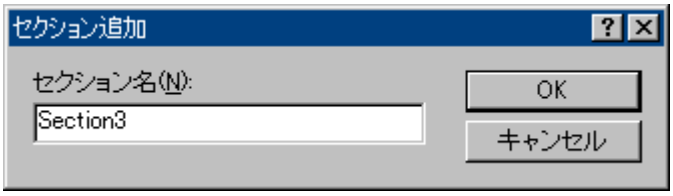
左側のペインのセクション設定ツリーで、アドレス下にセクショングループが追加されます。

13.1.4 セクションを追加する

編集モードでは、セクショングループ、サブセクショングループ、またはオーバレイグループ下にセクションの追加が可能です。

セクションを追加するには

1. 左側のペイン内のセクションを追加したいセクショングループ、サブセクショングループ、またはオーバレイグループ上で右クリックしポップアップメニューを表示します。
2. [セクションの追加]を選択します。
3. [セクション追加]ダイアログボックスが表示されます。



4. [セクション名]にセクション名を入力します。

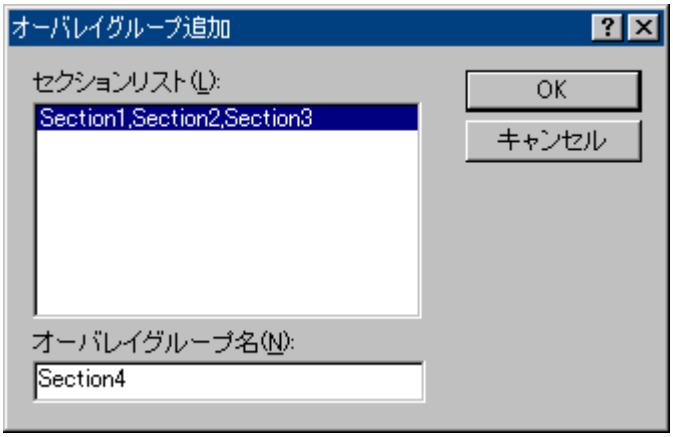
左側のペインのセクション設定ツリーで、セクショングループ、サブセクショングループ、またはオーバレイグループ下にセクションが追加されます。

13.1.5 オーバレイグループを追加する

編集モードでは、セクショングループ、サブセクショングループ、またはセクション（***1**）下にオーバレイ（***2**）グループの追加が可能です。

オーバレイグループを追加するには

1. 左側のペイン内のオーバレイグループを追加したいセクショングループ、サブセクショングループ、またはセクション（***1**）上で右クリックしポップアップメニューを表示します。
2. [オーバレイグループの追加]を選択します。
3. [オーバレイグループ追加]ダイアログボックスが表示されます。



4. [オーバレイグループ名]にオーバレイグループ名を入力します。

左側のペインのセクション設定ツリーで、セクショングループ、サブセクショングループ、またはセクション（***1**）下にオーバレイグループが追加されます。

***1.** SuperH RISC engine ファミリ用 C/C++コンパイラパッケージ V.9.00 Release 04、および H8, H8S, および H8SX ファミリ用 C/C++コンパイラパッケージ V.6.01 Release 02 以降のバージョンでサポートされます。

***2.** リンケージエディタがオーバレイをサポートしていない場合は使用できません。

13.1.6 選択項目を編集する

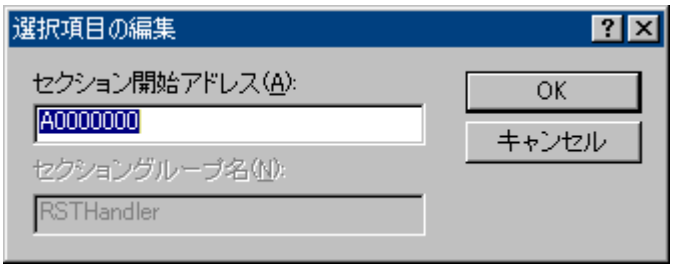
編集モードでは、アドレス値、およびセクショングループ、サブセクショングループ、オーバレイグループ、セクションの名前の編集が可能です。

選択項目を編集するには

1. 次のいずれかを行ってください。

- 左側のペイン内の編集したい項目上で右クリックしポップアップメニューを表示する。[選択項目の編集]を選択する。
- 左側のペイン内の編集したい項目をダブルクリックする。

2. [選択項目の編集]ダイアログボックスが表示されます。
3. 編集フィールドにデータを入力してください。



13.1.7 プライマリセクションに設定する

編集モードでは、選択したオーバレイグループをプライマリセクションに設定（*****）することが可能です。

プライマリセクションに設定するには

1. 左側のペイン内のプライマリセクションにしたいオーバレイグループ上で右クリックしポップアップメニューを表示します。
2. [プライマリセクションに設定]を選択します。

左側のペインのセクション設定ツリーで、選択したオーバレイグループがプライマリセクションに設定されます。

***.** リンケージエディタがオーバレイをサポートしていない場合は使用できません。

13.1.8 メモリマップを設定する

デバッギングプラットフォームと接続している場合にメモリマップを設定することができます。

メモリマップを設定するには

1. 左側のペイン内で右クリックしポップアップメニューを表示します。
2. [メモリマップ設定]を選択します。
3. メモリマップを設定するダイアログボックスが表示されます。アドレス空間の現在のマッピング状況がリストで表示されていますので、必要に応じてメモリマップおよびメモリリソースを設定してください。
4. [OK]ボタンをクリックします。

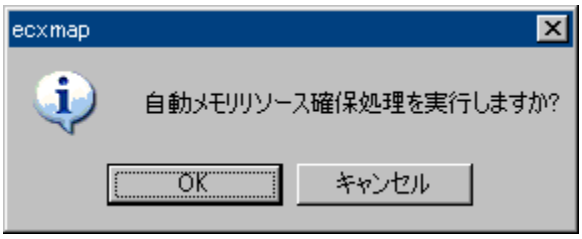
本機能のサポートは、デバッギングプラットフォームに依存します。

13.1.9 メモリリソースを自動的に確保する

リンケージエディタが出力するリンケージリストファイル（.map）がある場合、メモリマップおよびリンケージマップ情報に基づきメモリリソースを自動的に確保することができます。デバッギングプラットフォームと接続している場合に使用できます。

メモリリソースを自動的に確保するには

1. 左側のペイン内で右クリックしポップアップメニューを表示します。
2. [メモリリソース自動確保]を選択します。下記のダイアログボックスが表示されます。



3. 処理を継続する場合は[OK]をクリックしてください。
4. 処理が完了すると下記のダイアログボックスが表示されます。

本機能のサポートは、デバッギングプラットフォームに依存します。

13.1.10　セクション設定ツリーを印刷する

左側のペインのセクション設定ツリーを印刷できます。

セクション設定ツリーを印刷するには

1. 左側のペイン内で右クリックしポップアップメニューを表示します。
2. [印刷]を選択します。
3. 標準のプリントフォーマットを選択するダイアログボックスが表示されます。ここで、使用するプリンタとページ設定のオプションを選択することができます。[印刷]ボタンをクリックしてください。

13.1.11　未割り当て領域を表示する

セクションが割り当てられていない領域の表示が可能です。デフォルトでは表示されません。

未割り当て領域を表示するには

1. 右側のペイン内で右クリックしポップアップメニューを表示します。
2. [未割り当て領域の表示]を選択します。
3. [未割り当て領域の表示]にチェックマークが付きます。これにより右側のペインで、セクションが割り当てられていない領域がすべて表示されます。Name カラムに”Unallocated Area”と表示しています。

13.1.12　サイズ 0 のセクションを表示する

サイズ 0 のセクションの表示が可能です。デフォルトでは表示されません。

サイズ 0 のセクションを表示するには

1. 右側のペイン内で右クリックしポップアップメニューを表示します。
2. [サイズ 0 のセクション表示]を選択します。
3. 右側のペインで、サイズ 0 のセクションがすべて表示されます。このとき”End Address”カラムに終了アドレスは表示されません。

13.1.13　アドレスに対応するソースコードを表示する

選択している行の該当アドレスに対応したソースファイルをエディタウィンドウで開きます。

アドレスに対応するソースコードを表示するには

次のいずれかを行います。

- 右側のペイン内のソースを表示したいセクション情報行の上で右クリックし、ポップアップメニューを表示します。[ソースファイル表示]を選択します。
- 右側のペイン内のソースを表示したいセクション情報行をダブルクリックします。

13.1.14　セクションリストを印刷する

右側のペインのセクションリストを印刷できます。

セクションリストを印刷するには

1. 右側のペイン内で右クリックしポップアップメニューを表示します。
2. [印刷]を選択します。
3. 標準のプリントフォーマットを選択するダイアログボックスが表示されます。ここで、使用するプリンタとページ設定のオプションを選択することができます。[印刷]ボタンをクリックしてください。

13.2　シンボル情報を見る

Map Symbol Information ウィンドウを使用すると、セクションごとにシンボルを表示できます。シンボル数が多い場合も、検索機能、フィルタ機能で必要な情報だけを表示できます。

13.2.1　Map Symbol Information ウィンドウを開く

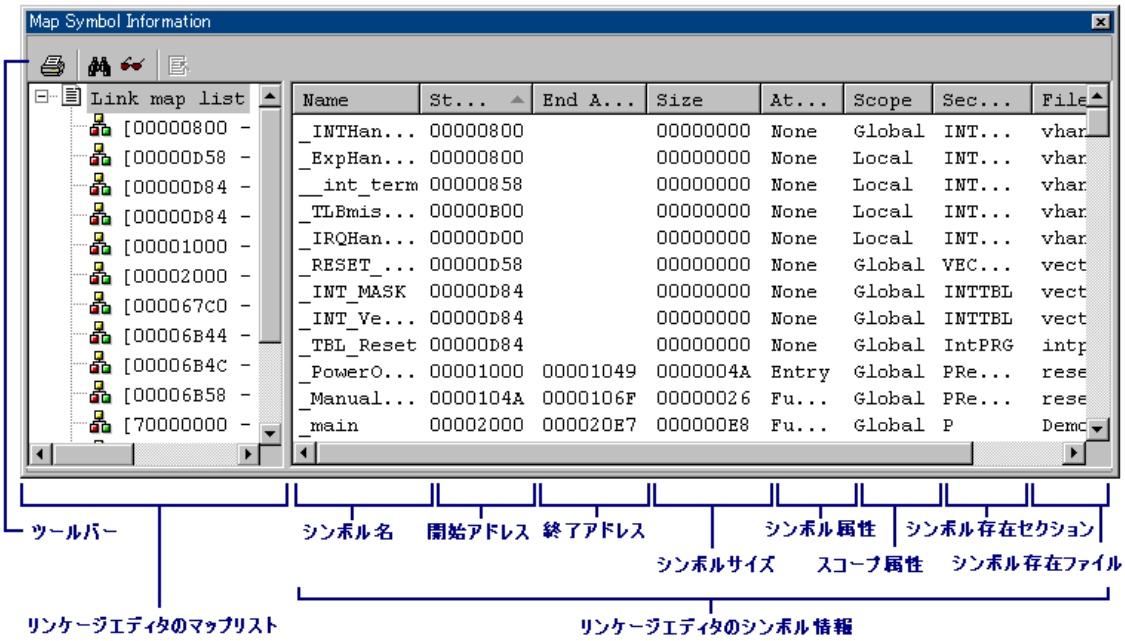
Map Symbol Information ウィンドウを開くには

1. 次のいずれかを行ってください。

- [マップ]ツールバーボタン（ ）をクリックする。
- [表示->マップ]を選択する。

2. [マップ種別の選択]ダイアログボックスが表示されます。
3. [マップ種別]ドロップダウンリストから”Map Symbol Information”を選択します。
4. [OK]ボタンをクリックします。

ウィンドウの構成



- リンケージエディタのマップリストを左側のペインに表示します。
  - o リンケージエディタが出力するリンケージリストファイル（.map）がある場合に表示されます。
  - o マップリストのルート(Linker map list)を選択すると、すべてのシンボル情報を右側のペインに表示します。
  - o マップリストのセクションを選択すると、選択したセクションに属するシンボル情報を右側のペインに表示します。
- リンケージエディタのシンボル情報を右側のペインに表示します。
  - o リンケージエディタが出力するリンケージリストファイル（.map）にシンボルリストがある場合に表示します。
  - o カラムヘッダ上のマウスのドラッグで各カラムの幅を調節できます。また、カラムヘッダ上で右クリックするとポップアップメニューを表示します。カラムが利用できる場合は、エントリの横にはチェックマークがあります。エントリをクリックすると、各カラムの表示、非表示を切り替えます。
  - o リストのカラムヘッダをクリックすると、リスト項目はそのカラムによってソートされます。ウィンドウを表示すると、"Start Address"の昇順でソートされています。
  - o ”Attribute”カラムにはシンボル属性を表示します。

    Entry：　エントリ関数
    Function：　関数名
    Data：　変数名
    None：　未設定(ラベル、アセンブラシンボル)
  - o ”Scope”カラムにはスコープ属性を表示します。

    Global：　グローバルシンボル
    Local：　ローカルシンボル

ウィンドウのオプション

- リンケージエディタのマップリスト（左側のペイン）

左側のペイン内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

本ウィンドウは、基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	機能
印刷		リンケージエディタのマップリストを印刷します。
ツールバー表示	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ	-	ツールバーボタンをカスタマイズします。

- リンケージエディタのシンボル情報（右側のペイン）

右側のペイン内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

本ウィンドウは、基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	機能
検索		シンボルを検索します。
次を検索		次のシンボルを検索します。
フィルタ		シンボル情報をフィルタ表示します。
ソースファイル表示		選択行のアドレスに対応したソースコードを表示します。
印刷		リンケージエディタのシンボル情報を印刷します。

13.2.2 マップリストを印刷する

左側のペインのリンケージエディタのマップリストを印刷できます。

マップリストを印刷するには

1. 左側のペイン内で右クリックしポップアップメニューを表示します。
2. [印刷]を選択します。
3. 標準のプリントフォーマットを選択するダイアログボックスが表示されます。ここで、使用するプリンタとページ設定のオプションを選択することができます。[印刷]ボタンをクリックしてください。

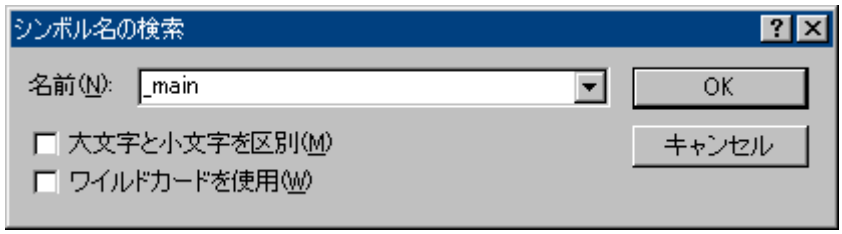
13.2.3 シンボルを検索する

リンケージエディタのシンボルの検索が可能です。

シンボルを検索するには

1. 右側のペイン内で右クリックしポップアップメニューを表示します。

2. [検索]を選択します。
3. [シンボル名の検索]ダイアログボックスが表示されます。



4. [名前]に検索するシンボルを入力、またはドロップダウンリストボックスから以前検索したシンボルを選択してください。
5. 大文字と小文字を識別する場合は[大文字と小文字を区別]チェックボックスをオンにしてください。
6. ワイルドカード（”?”または”*”）を使用する場合は[ワイルドカードを使用]チェックボックスをオンにしてください。

? :　任意の 1 文字
* :　任意の文字列

7. [OK]ボタンをクリックします。

右側のペインのリンケージエディタのシンボル情報で該当行が強調表示されます。

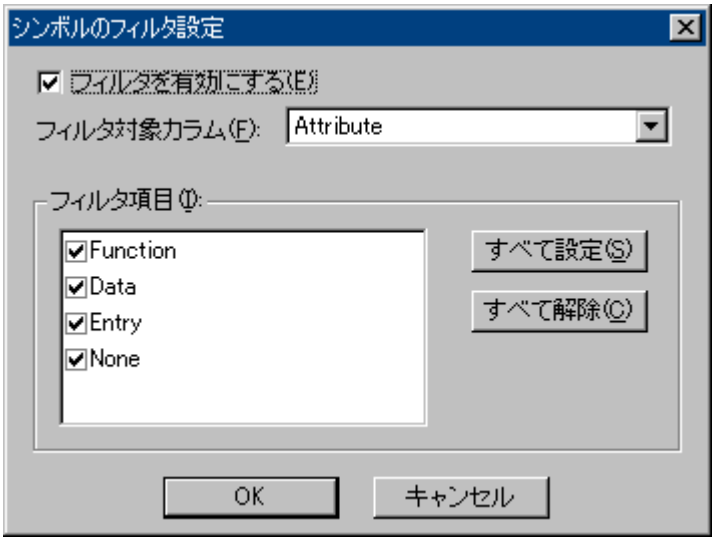
検索文字列の次の該当箇所を検索する場合は、[次を検索]を選択してください。

13.2.4　シンボル情報をフィルタ表示する

リンケージエディタのシンボル情報のフィルタ表示が可能です。

シンボル情報をフィルタ表示するには

1. 右側のペイン内で右クリックしポップアップメニューを表示します。
2. [フィルタ]を選択します。
3. [シンボルのフィルタ設定]ダイアログボックスが表示されます。



4. [フィルタを有効にする]チェックボックスをオンにします。チェックボックスのデフォルトはオフです。
5. [フィルタ対象カラム]でフィルタリングするカラムを選択します。
6. [フィルタ項目]でカラムに表示する項目を選択します。

フィルタ対象カラム	フィルタ項目	機能
Attribute (シンボル属性)	Function	関数でフィルタリングを行なう
	Data	データシンボルでフィルタリングを行なう
	Entry	実行開始位置でフィルタリングを行なう
	None	上記以外(ラベル等)のシンボルでフィルタリングを行なう
Scope (スコープ属性)	Global	グローバルシンボルでフィルタリングを行なう
	Local	ローカルシンボルでフィルタリングを行なう
Section (シンボル存在セクション)	セクション名	セクション名でフィルタリングを行なう
File (シンボル存在ファイル名)	ファイル名	ファイル名でフィルタリングを行なう

7. [すべて設定]ボタンをクリックするとフィルタ項目のすべてのチェックボックスがオンになります。
8. [すべて解除]ボタンをクリックするとフィルタ項目のすべてのチェックボックスがオフになります。
9. [OK]ボタンをクリックします。

右側のペインのリンケージエディタのシンボル情報が、選択した項目でフィルタ表示されます。

複数条件でのフィルタ表示を行なう場合は、さらに別のフィルタ対象カラムを選択し、そのカラム内のフィルタ項目を選択してください。

フィルタを無効にするには、[シンボルのフィルタ設定]ダイアログボックスで[フィルタを有効にする]チェックボックスをオフにしてください。

13.2.5 アドレスに対応するソースコードを表示する

選択している行の該当アドレスに対応したソースファイルをエディタウィンドウで開きます。

アドレスに対応するソースコードを表示するには

次のいずれかを行います。

- 右側のペイン内のソースを表示したいシンボル情報行の上で右クリックし、ポップアップメニューを表示します。[ソースファイル表示]を選択します。
- 右側のペイン内のソースを表示したいシンボル情報行をダブルクリックします。

13.2.6 シンボル情報を印刷する

右側のペインのリンケージエディタのシンボル情報を印刷できます。

シンボル情報を印刷するには

1. 右側のペイン内で右クリックしポップアップメニューを表示します。
2. [印刷]を選択します。

3. 標準のプリントフォーマットを選択するダイアログボックスが表示されます。ここで、使用するプリンタとページ設定のオプションを選択することができます。[印刷]ボタンをクリックしてください。

14. コマンドラインの使用

High-performance Embedded Workshop (HEW) コマンドラインインタプリタにより、ウィンドウメニュー/コマンドの代わりにテキストベースのコマンドで、デバッギングプラットフォームを制御できます。これはあらかじめ決まったコマンド列をバッチファイルから呼び出して、デバッギングプラットフォームに送る場合に特に便利です。さらに出力をログファイルに記録することもできます。

注：

- 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したいときは、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りますので、ファイル指定をやり直してください。

  (例) FILE_LOAD ELF/DWARF2 $(CONFIGDIR)¥¥demo.abs

14.1 コマンドラインウィンドウを開く

[表示->コマンドライン]を選択するか、[コマンドライン]ツールバーボタン( )をクリックすると表示します。

ウィンドウの構成

- テキストベースのコマンドを入力してデバッギングプラットフォームを制御できるウィンドウです。
- コマンド行をファイルから呼び出して実行すること、および出力結果をファイルに記録できます。
- [コマンドライン] ウィンドウ最終行のコマンドプロンプト(">")に続けて入力後、"Enter"キーを押すとコマンドを実行します。
- 使用できるコマンドについては、「リファレンス 3. コマンドライン一覧」およびオンラインヘルプを参照してください。
- ウィンドウタイトルとしてバッチファイル名とログファイル名をコロンで区切って表示します。

- 最終行で"Ctrl+↑"または"Ctrl+↓"を押すと直前に入力したコマンド行を表示することができます。
- 本ウィンドウで、HEW コマンドおよび TCL コマンドが入力できます。

ウィンドウのオプション

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

本ウィンドウは、基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

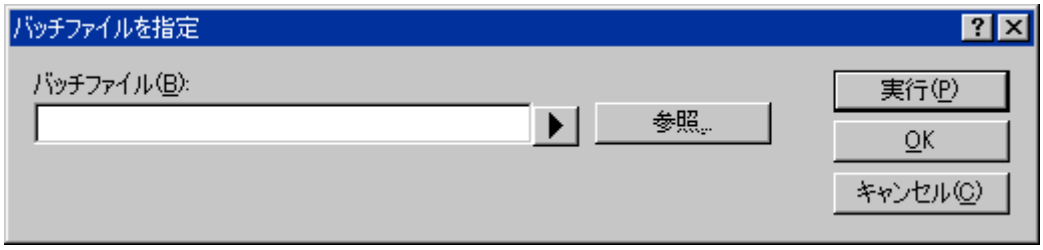
ポップアップメニュー項目		ツールバーボタン	マクロの記録	機能
バッチファイル指定			-	コマンドファイルを設定します。
バッチファイル実行			-	コマンドファイルを実行します。
バッチファイル停止			-	コマンドファイルの実行を停止します。
ログファイル指定			-	ログファイルを設定します。
ロギング開始/停止			-	ログファイルへの出力を開始/停止します。
参照			-	ファイルのフルパスを入力します。
プレースホルダ	コンフィグレーションディレクトリ	Configuration Directory	-	プレースホルダ$(CONFIGDIR)を挿入します。
	コンフィグレーション名		-	プレースホルダ$(CONFIGNAME)を挿入します。
	プロジェクトディレクトリ		-	プレースホルダ$(PROJDIR)を挿入します。
	プロジェクト名		-	プレースホルダ$(PROJECTNAME)を挿入します。
	ワークスペースディレクトリ		-	プレースホルダ$(WORKSPDIR)を挿入します。
	ワークスペース名		-	プレースホルダ$(WORKSPNAME)を挿入します。
	HEW インストールディレクトリ		-	プレースホルダ$(HEWDIR)を挿入します。
すべて選択			-	アクティブウィンドウのすべての内容を選択します。
コピー			-	選択部分をコピーし Windows®クリップボードに貼り付けます。
カット			-	選択部分を切り取り Windows®クリップボードに貼り付けます。
貼り付け			-	Windows®クリップボードの内容をコピーしてアクティブウィンドウのカーソル位置に貼り付けます。
クリア			-	ウィンドウの内容をクリアします。
元に戻す			-	直前の入力を元に戻します。
-			-	開き括弧（{または[）が入力され閉じ括弧（}または]）が入力されていない場合ボタンが点灯します（ボタン選択不可）。
ツールバー表示		-	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ		-	-	ツールバーボタンをカスタマイズします。

14.2　コマンドファイルを設定する

自動テストなどのあらかじめ定義した一連のコマンド行を実行する場合は、コマンドファイルを利用すると便利です。

コマンドファイルはテキストエディタで作成し、実行すべきコマンドラインを記述しておきます。コマンドファイル型名のデフォルトは”.hdc”です。

コマンドファイルを設定するには、[バッチファイルを指定]ダイアログボックスを使用します。ポップアップメニューから[バッチファイル指定]を選択すると、[バッチファイルを指定]ダイアログボックスを表示します。コマンドファイル名(*.hdc)を入力できます。 [OK]ボタンをクリックすると、設定したコマンドファイル名を、ウィンドウタイトルに表示します。[キャンセル]ボタンをクリックすると、設定を変更しないでダイアログボックスを閉じます。



14.3　コマンドファイルを実行する

コマンドファイルを実行するには、[バッチファイルを指定]ダイアログボックスで[実行]ボタンをクリックするか、ポップアップメニューから[バッチファイル実行]を選択します。[バッチファイル実行]メニューはコマンドファイル実行中にグレー表示となり、コマンドファイル実行が停止してユーザに制御が戻ったときに有効表示になります。

コマンドファイルを実行中にコマンドラインウィンドウを閉じようとすると、下記のメッセージボックスを表示します。

コマンドファイルの実行を強制終了し、コマンドラインウィンドウを閉じる場合は、[OK]ボタンをクリックしてください。

コマンドファイルの実行を継続する場合は、[キャンセル]ボタンをクリックしてください。この時コマンドラインウィンドウは閉じません。

14.4　コマンド実行を中断する

コマンド実行を中断する場合は、ポップアップメニューから [バッチファイル停止]を選択します。

[バッチファイル停止]メニューはコマンド実行中に有効表示になります。

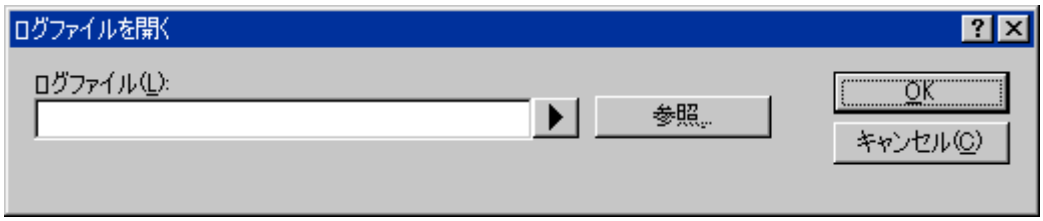
注：

TCL コマンドの場合、TCL コマンドが完了しないとコマンド実行が中断されないことがあります。

14.5　ログファイルを設定する

コマンド実行結果を保存するログファイルは[ログファイルを開く]ダイアログボックスで設定します。

[ログファイルを開く]ダイアログボックスを開くには、ポップアップメニューから[ログファイル指定]を選択します。



14.6　ログファイルへの出力を開始/停止する

ファイルへのロギング処理を実行するか、停止するかの切り替えは、ポップアップメニューの[ロギング開始/停止]で行います。ログファイルの内容は、ロギングが終了するか、チェックボックスをクリアしてロギングを一時的に停止しなければ表示できないことにご注意ください。ロギングを再び開始すると、ログファイルに追加します。

14.7　ファイルのフルパスを入力する

ポップアップメニューで[参照]を選択すると、参照ダイアログボックスを表示します。ここでファイルを選択し[開く]をクリックすると、フルパスファイル名をカーソル位置に貼り付けます。

カーソルが最終行にある場合のみ使用できます。

14.8　プレースホルダを挿入する

ポップアップメニューから[プレースホルダ]のサブメニューで選択したプレースホルダをカーソル位置に貼り付けます。

カーソルが最終行にある場合のみ使用できます。

[プレースホルダ]のサブメニュー	カーソル位置に貼り付けるプレースホルダ
コンフィグレーションディレクトリ	$(CONFIGDIR)
コンフィグレーション名	$(CONFIGNAME)
プロジェクトディレクトリ	$(PROJDIR)
プロジェクト名	$(PROJECTNAME)
ワークスペースディレクトリ	$(WORKSPDIR)
ワークスペース名	$(WORKSPNAME)
HEW インストールディレクトリ	$(HEWDIR)

14.9　ウィンドウの表示内容をすべて選択する

ポップアップメニューから[すべて選択]を選択すると、コマンドラインウィンドウの出力をすべて選択します。

14.10 選択部分をクリップボードにコピーする

ポップアップメニューから[コピー]を選択すると、選択したテキストブロックを Windows®クリップボードにコピーします。テキストブロックを選択している場合にのみ使用できます。

14.11 選択部分をクリップボードに切り取る

ポップアップメニューから[切り取り]を選択すると、選択したテキストブロックを削除し Windows®クリップボードに貼り付けます。現在入力中のカーソル位置のテキストブロックを選択している場合にのみ使用できます。

14.12 クリップボードの内容を貼り付ける

ポップアップメニューから[貼り付け]を選択すると、Windows®クリップボードの内容を現在入力中のカーソル位置に貼り付けます。カーソルが最終行にある場合のみ使用できます。

14.13 ウィンドウの表示内容をクリアする

ポップアップメニューから[クリア]を選択すると、コマンドラインウィンドウに表示している内容をすべてクリアします。

14.14 直前の操作を元に戻す

ポップアップメニューから[元に戻す]を選択すると、現在のカーソル位置で、直前の操作を元に戻します。

14.15 括弧の呼応状態を確認する

[開き括弧インジケータ]ツールバーボタン( )が表示されている場合、コマンドラインウィンドウに入力中の式にある括弧の呼応状態を確認することができます。[開き括弧インジケータ]ツールバーボタンの点灯状態により括弧の呼応状態を確認します。ボタンは呼応状態の表示のみで操作はできません。

括弧の開始である”[”または”{”が入力されネスト部分が入力されると、括弧の終了である”]”または”}”が入力されるまでは、[開き括弧インジケータ]ツールバーボタンが点灯( )します。

コマンドラインウィンドウへの TCL 関数の入力例	括弧の入力状況	ツールバーボタンの点灯状態
>set bit 1	なし	
1	なし	
>set value 1	なし	
1	なし	
>set r 1	なし	
1	なし	
>if {$bit & $value} {	開き括弧が入力された	
> set r 0$r	なし	
>} else {	閉じ括弧および開き括弧が入力された	
> set r 1$r	なし	
>}	閉じ括弧が入力された	
> 01	なし	

15. マクロ生成支援機能の使用

マクロ生成支援機能は、High-performance Embedded Workshop (HEW) システムのアプリケーション関連（***1**）、ビルド関連（***2**）、およびデバッグ関連（***3**）などの一部の操作を HEW コマンドラインのコマンドとして記録、または記録したコマンドを実行する機能です。

記録するファイル（HEW マクロファイル）は、拡張子が hdc のコマンドバッチファイルで編集可能です。このファイルは、HEW インストールディレクトリ下の「Macro」ディレクトリに設置されます。デフォルトで Default.hdc ファイルが作成されます。

HEW コマンドラインのコマンドに対応する HEW システムのすべての操作が記録できるわけではありません。各機能のメニュー項目一覧には”マクロの記録”欄にマクロの記録マーク（●）を表示し HEW マクロファイルへ記録可能かどうかを示しています。また、「15.5 マクロファイルにマクロを記録できる機能」に記録する HEW コマンドラインのコマンドを記載していますので参照してください。

***1**. プロジェクトの変更、セッションの変更、コンフィグレーションの変更など

***2**. コンパイル、ビルドなど。機能のサポートは、デバッギングプラットフォームに依存します。

***3**. モジュールのダウンロード、メモリ値の変更、レジスタ値の変更、ソフトウェアブレークの設定／解除、プログラム実行など

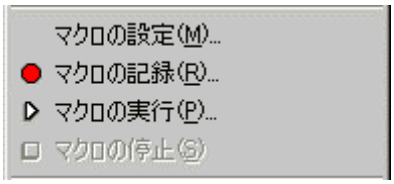
マクロ生成支援機能は[ツール]メニューと[マクロ]ツールバーで利用できます。

[マクロの記録]を選択してから[マクロの停止]を選択するまでが 1 つの”マクロ”で、1 つの HEW マクロファイルには複数のマクロを記録できます。マクロには、複数の HEW コマンドラインのコマンドが含まれます。

15.1 マクロメニューとツールバー

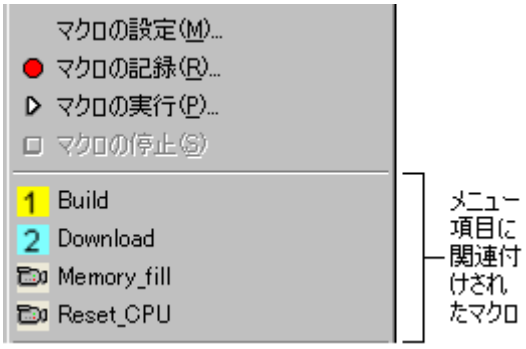
マクロ生成支援機能はメニューとツールバーで利用できます。

マクロのメニューは[ツール]メニューにあります。以下の標準のメニュー項目があります。



[マクロの設定]メニュー項目から[マクロの設定]ダイアログボックスを開くことができます。ここで、使用中のマクロファイルやマクロを管理します。この他のメニュー項目は[マクロ]ツールバー上の項目と同じです。

マクロをメニューに関連付けることもできます。これを行うとマクロの標準のメニュー項目の下にメニュー項目が追加されます。以下にその例を示します。

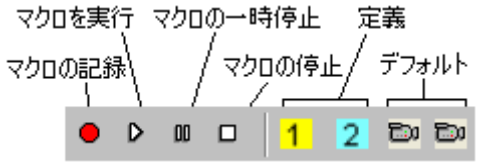


マクロをメニューおよびツールバーに関連付けた場合は、メニュー項目の横にアイコンが表示されます。

[マクロ]ツールバーの標準ボタンを以下に示します。

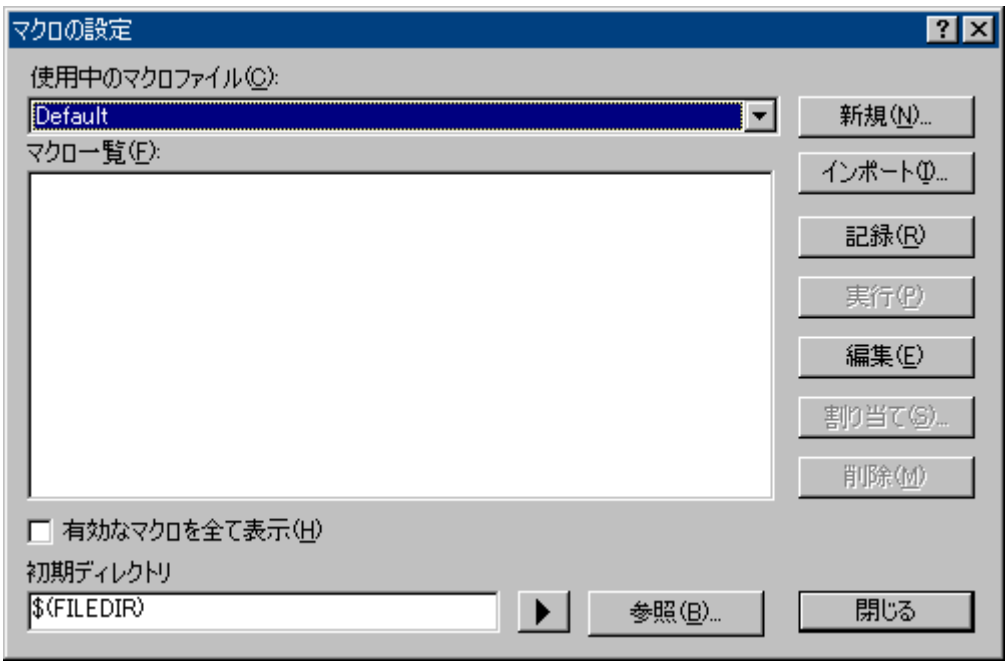
- [マクロの記録]ボタンでマクロの記録を開始します。このボタンをクリックすると、デバッグや HEW の管理操作は記録されます。これは[ツール]メニューの[マクロの記録]メニューと同じです。
- [マクロの実行]ボタンでマクロ機能の再生を開始します。マクロを定義すると[マクロの選択]ダイアログボックスが表示されます。再生する特定のマクロを指定してください。
- [マクロの一時停止]ボタンはマクロを記録または実行しているときのみ有効です。このボタンは現在の操作を一時停止して後で再開できるようにします。
- [マクロの停止]ボタンはマクロを記録または実行しているときのみ有効です。このボタンは実行中のマクロ操作を停止します。

マクロをメニューに関連付けできるのと同様に、マクロをツールバーに関連付けることもできます。マクロを関連付けたツールバーのアイコンには、定義したアイコンか、デフォルトのアイコンを使用することができます。マクロをツールバーに関連付けると、関連付けされたマクロが標準のマクロボタンの右側に表示されます。ボタンをクリックすると、関連付けされたマクロが起動します。



15.2 マクロの設定ダイアログボックスを使用する

[マクロの設定]ダイアログボックスから HEW システムの現在使用中のマクロファイルをアクセスしたり管理したりできます。[マクロの設定]ダイアログボックスを以下に示します。



[使用中のマクロファイル]ドロップリストには HEW システムで現在使用中のマクロファイルをすべて表示します。これらのファイルは HEW のインストールディレクトリの"Macro"フォルダに格納されています。"Macro"ディレクトリにコピーされたファイルはすべて自動的に[使用中のマクロファイル]ドロップリストに追加されます。

[使用中のマクロファイル]で選択されたファイルに含まれるマクロは、[マクロ一覧] リストボックスに表示されます。[使用中のマクロファイル]の選択を変更すると自動的にリストが更新されます。[有効なマクロを全て表示]チェックボックスをオンにすると、現在定義されているすべてのマクロを表示します。チェックしてこのオプションを指定することにより、[使用中のマクロファイル] リストボックスの値は無視されます。

[初期ディレクトリ]は相対パスを持つマクロファイルを実行するために初期ディレクトリを設定します。ディレクトリを変更するコマンドと同様にディレクトリを変更後、初期ディレクトリからマクロファイルを実行します。

新規のマクロファイルを作成するには

1. [ツール->マクロの設定]を選択し、[マクロの設定]ダイアログボックスを表示してください。
2. [新規]ボタンをクリックしてください。[マクロファイルの新規追加]ダイアログボックスが表示されます。
3. 新規のマクロファイル名を入力してください。マクロファイル名は 64 文字以内で、半角英数字、半角下線のみ使用してください。
4. OK をクリックすると、[使用中のマクロファイル]リストに新規マクロファイル名が追加されます。

既存のマクロファイルをインポートするには

1. [ツール->マクロの設定]を選択し、[マクロの設定]ダイアログボックスを表示してください。
2. [インポート]ボタンをクリックしてください。[インポートファイルの選択]ダイアログボックスが表示されます。
3. 既存のマクロファイルを選択してください。
4. [選択]をクリックしてください。[使用中のマクロファイル]リストに新しくマクロファイル名が追加されます。
5. リスト内のマクロファイル名を選択すると、インポートできるマクロが表示されます。

マクロファイルはテキストベースの HEW コマンドバッチファイルあるため、編集することができます。

既存のマクロファイルを編集するには

1. [ツール->マクロの設定]を選択し、[マクロの設定]ダイアログボックスを表示してください。
2. [使用中のマクロファイル]にあるマクロファイルを選択してください。
3. [マクロ一覧]から編集したいマクロを選択してください。
4. [編集]ボタンをクリックしてください。
5. [マクロの設定]ダイアログボックスが閉じファイルがエディタ内で表示されます。

通常のテキストファイルと同様に HEW ウィンドウ上にドラッグアンドドロップしてマクロファイル開くことも可能です。

既存のマクロファイルからマクロを削除するには

1. [ツール->マクロの設定]を選択し、[マクロの設定]ダイアログボックスを表示してください。
2. [使用中のマクロファイル]にあるマクロファイルを選択してください。
3. [マクロ一覧]から削除したいマクロを選択してください。
4. [削除]ボタンをクリックしてください。

[閉じる]ボタンをクリックすると[マクロの設定]ダイアログボックスを閉じます。このダイアログボックスでの操作はやり直しできません。

15.3 既存のマクロファイルをインポートする

すでに複数のマクロがある他のマシンから既存の HEW マクロファイルをインポートするには、以下の操作をしてください。

1 つの方法は、追加するマクロファイルを HEW ルートディレクトリの"Macro"フォルダにコピーすることです。自動的に追加されたファイルを検出して[マクロの設定]ダイアログボックスにその情報を追加します。追加したファイルはデフォルトでは使用できません。デフォルトのマクロファイルを追加したファイルに切り替えるには、[マクロの設定]ダイアログボックスを使用してください。

もう一つの方法は、[マクロの設定]ダイアログボックスを使用して、マクロファイルをインポートすることです。以下に操作方法を示します。

既存のマクロファイルを HEW にインポートするには

1. [ツール->マクロの設定]を選択し、[マクロの設定]ダイアログボックスを表示してください。
2. [インポート]ボタンをクリックしてください。
3. 標準のファイルブラウザが開くので、インポートしたい既存のマクロファイルを選択してください。ファイルを選択し[選択]ボタンをクリックしてください。
4. ファイルがインポートされます。自動的にそのファイルが[使用中のマクロファイル]ドロップダウンリストのデフォルトマクロファイルとなります。
5. インポートしたファイルが HEW ルートディレクトリの Macro フォルダにコピーされます。
6. [マクロの設定]ダイアログボックスで[閉じる]をクリックすると、インポートしたマクロファイルが使用可能になります。

15.4 マクロを記録する

マクロファイルを記録するには下記に示すいくつかの方法があります。

ツールバーまたはメニューからマクロを記録するには

1. [マクロ]ツールバーの[マクロの記録]ボタン(●)、または[ツール]メニューの[マクロの記録]メニュー項目をクリックしてください。マウスポインタにマクロの記録アイコンがつき(↖●)、記録されていることを示します。マクロが記録されているとき、[マクロの一時停止]ボタン(❚❚)および[マクロの停止]ボタン(□)やメニューが有効になります。記録の一時停止中はマウスポインタに[マクロの一時停止]アイコンがつきます(↖●❚❚)。
2. マクロ機能をサポートしないデバッギングプラットフォームを使用している場合、"WARNING: ターゲットがマクロ機能をサポートしていないため、記録内容は制限されます。続けますか？"という確認メッセージを表示します。「はい」をクリックすると記録を続けます。確認のダイアログボックスを表示しない場合、[今後表示しない]チェックボックスをオンにしてください。確認のダイアログボックスを再び表示するには、[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を表示します。[確認]タブの[マクロ機能をサポートしていないターゲットでの記録]チェックボックスをオンにしてください。デフォルトはチェックボックスがオンです。
3. [マクロの停止]をクリックするまで記録を続行します。マクロファイルに記録する操作については、次節の「15.5 マクロファイルにマクロを記録できる機能」を参照してください。
4. [マクロの停止]をクリックすると[マクロの新規追加]ダイアログボックスが表示されるのでマクロ名を入力してください。
5. OK をクリックするとマクロ名でマクロを保存し、現在のマクロファイルに追加します。キャンセルをクリックするとダイアログボックスを閉じマクロの記録を停止します。

[マクロの設定]ダイアログボックスでも記録を行えます。

[マクロの設定]ダイアログボックスからマクロを記録するには

1. [ツール->マクロの設定]を選択し、[マクロの設定]ダイアログボックスを表示してください。
2. [記録]ボタンをクリックしてください。[マクロの設定]ダイアログボックスが閉じ記録を開始します。

15.5 マクロファイルにマクロを記録できる機能

HEW V.4.02 では、HEW システムのすべての操作がスクリプトとしてマクロファイルに記録できるわけではありません。記録できる操作については、各機能のメニュー項目一覧にも”マクロの記録”として記録機能のサポートの有無を記載していますので参照してください。

- マクロを記録できる機能(HEW 共通)
- マクロを記録できる機能(デバッギングプラットフォーム依存)

15.5.1 マクロを記録できる機能（HEW 共通）

連携ツールの製品パッケージにバンドルされている HEW で共通の機能については、以下の機能を操作するとマクロをスクリプトにしマクロファイルに記録されます。

- **メニュー、ショートカットキー、およびツールバーボタンの操作**

マクロの記録中に下記のメニュー、ショートカットキー、またはツールバーボタンを操作すると、下の表の「記録する HEW コマンドラインのコマンド」欄にあるコマンドをマクロファイルに記録します。

メニュー	メニューオプション		ショートカットキー	ツールバーボタン	記録する HEW コマンドラインのコマンド
ファイル	ワークスペースを開く		-	-	OPEN_WORKSPACE
	ワークスペースの保存		-	-	SAVE_WORKSPACE
	ワークスペースを閉じる		-	-	CLOSE_WORKSPACE
	新規セッション		-	-	CHANGE_SESSION
	セッションのインポート		-	-	CHANGE_SESSION
	セッションの保存		-	-	SAVE_SESSION
	セッションのリフレッシュ		-	-	REFRESH_SESSION
	ダウンロードモジュールの追加		-	-	「ダイアログボックス」を参照してください。
	最近使ったワークスペース		-	-	OPEN_WORKSPACE
	最近ダウンロードしたモジュール		-	-	FILE_LOAD
編集	ブレークポイントの挿入/削除		F9		SET_DISASSEMBLY_SOFT_BREAK /SET_SOURCE_SOFT_BREAK
	ブレークポイントの有効化/無効化		Ctrl+F9		STATE_DISASSEMBLY_SOFT_BREAK /STATE_SOURCE_SOFT_BREAK
表示	ワークスペース		Alt+K		「ウィンドウ」を参照してください。
	逆アセンブリ		Ctrl+D		
	CPU	レジスタ	Ctrl+R		
		メモリ	Ctrl+M		
		IO	Ctrl+I		
プロジェクト	アクティブプロジェクトに設定		-	-	CHANGE_PROJECT
	プロジェクトの挿入		-	-	CHANGE_PROJECT
	構成の編集 ***2**		-	-	SAVE_SESSION

ビルド ***2**	コンパイル		Ctrl+F7		BUILD_FILE
	ビルド		F7		BUILD
	すべてをビルド		-		BUILD_ALL
	複数ビルド		-	-	「ダイアログボックス」を参照してください。
	クリーン アクティブプロジェクト		-	-	CLEAN
	すべてをクリーン		-	-	CLEAN
	ビルドの構成		-	-	「ダイアログボックス」を参照してください。
デバッグ	デバッグセッション		-	-	
	CPU のリセット		-		RESET
	実行		F5		GO
	リセット後実行		Shift+F5		GO_RESET
	カーソル位置まで実行		-		GO_TILL
	カーソル位置を PC 値に設定		-		REGISTER_SET ***1**
	条件を指定して実行		-	-	GO_TILL
	ステップイン		F11		STEP
	ステップオーバ		F10		STEP_OVER
	ステップアウト		Shift+F11		STEP_OUT
	ステップ		-	-	「ダイアログボックス」を参照してください。
	ステップモード	自動	-	-	STEP_MODE
		アセンブリ	-	-	STEP_MODE
		ソース	-	-	STEP_MODE
	プログラムの停止		-		HALT
	初期化		-	-	INITIALIZE
	接続 ***2**		-		CONNECT
	接続解除 ***2**		-		DISCONNECT
	メモリの保存		-	-	FILE_SAVE
	メモリのベリファイ ***2**		-	-	FILE_VERIFY
	ダウンロード	ダウンロードモジュール名	-	-	FILE_LOAD
		All Download Modules	-	-	FILE_LOAD_ALL
	アンロード	ダウンロードモジュール名	-	-	FILE_UNLOAD
		All Downloaded Modules	-	-	FILE_UNLOAD_ALL
基本設定	基数	16 進数	-		RADIX
		10 進数	-		RADIX
		8 進数	-		RADIX
		2 進数	-		RADIX

***1.** エミュレータまたはシミュレータのヘルプを参照してください。

***2.** 機能のサポートは、デバッギングプラットフォームに依存します。

- ウィンドウ

マクロの記録中にウィンドウの対象に対して、インプレース編集、カラムのダブルクリック、ポップアップメニューのクリックまたはクリックにより表示されるダイアログボックスで設定すると、下の表の「記録する HEW コマンドラインのコマンド」欄にあるコマンドをマクロファイルに記録します。

エディタウィンドウは、ワークスペースウィンドウの[Projects]タブ内のファイルをダブルクリックして表示してください。

ウィンドウ	対象	機能/操作	記録する HEW コマンドラインのコマンド
ワークスペースウィンドウの[Projects]タブ	ワークスペースのポップアップメニュー	クリーン 全プロジェクト ***2**	CLEAN
	プロジェクトのポップアップメニュー	ビルド ***2**	BUILD
		すべてをビルド ***2**	BUILD_ALL
		クリーン アクティブプロジェクト ***2**	CLEAN
		アクティブプロジェクトに設定	CHANGE_PROJECT
	プロジェクトファイルのポップアップメニュー	コンパイル <ファイル名> ***2**	BUILD_FILE
	Download modules フォルダのポップアップメニュー	全てダウンロード	FILE_LOAD
		ダウンロードモジュールの追加	「ダイアログボックス」を参照してください。
	Download modules のポップアップメニュー	ダウンロード	FILE_LOAD
		ダウンロード（debug 情報のみ）	FILE_LOAD
		アンロード	FILE_UNLOAD
		ダウンロードモジュールの追加	「ダイアログボックス」を参照してください。
	ダウンロードモジュール名	ダブルクリックによるダウンロード	FILE_LOAD
エディタおよび逆アセンブリ（ソースモード）	ポップアップメニュー	コンパイル '<ファイル名>' ***3**	BUILD_FILE
		ブレークポイントの挿入/削除	SET_SOURCE_SOFT_BREAK
		ブレークポイントの有効化/無効化	STATE_SOURCE_SOFT_BREAK
		カーソル位置まで実行	GO_TILL
		カーソル位置に PC 値を設定	REGISTER_SET ***1**
	S/W ブレークポイントカラム	ダブルクリックによるブレークポイントの挿入/削除	SET_SOURCE_SOFT_BREAK
エディタおよび逆アセンブリ（混合モード／逆アセンブリモード）	ポップアップメニュー	カーソル位置まで実行	GO_TILL
		カーソル位置に PC 値を設定	REGISTER_SET ***1**
		ブレークポイントの挿入/削除	SET_DISASSEMBLY_SOFT_BREAK
		ブレークポイントの有効化/無効化	STATE_DISASSEMBLY_SOFT_BREAK
	S/W ブレークポイント - ASM カラム	ダブルクリックによるブレークポイントの挿入/削除	SET_DISASSEMBLY_SOFT_BREAK
レジスタ	値	インプレース編集	REGISTER_SET ***1**
	フラグレジスタ ***2**	クリックによる編集	REGISTER_SET ***1**
	ポップアップメニュー	編集	REGISTER_SET ***1**
メモリ	値	インプレース編集	MEMORY_EDIT
	ポップアップメニュー／ツールバーボタン	設定	MEMORY_EDIT
		フィル	MEMORY_FILL
		コピー	MEMORY_MOVE
		比較***2**	MEMORY_COMPARE
		保存	FILE_SAVE

		読み込み	FILE_LOAD
IO	値	インプレース編集	MEMORY_EDIT
		ダブルクリックによる編集ダイアログボックスの表示	MEMORY_EDIT

***1.** エミュレータまたはシミュレータのヘルプを参照してください。

***2.** 機能のサポートは、デバッギングプラットフォームに依存します。

***3.** エディタウィンドウのみ表示します。

- ダイアログボックス

マクロの記録中にダイアログボックスで設定、またはツールバーの選択を変更すると、下の表の「記録する HEW コマンドラインのコマンド」欄にあるコマンドをマクロファイルに記録します。

ダイアログボックス	ツールバー	機能/操作	記録する HEW コマンドラインのコマンド
ダウンロードモジュール	-	[オフセット]、[フォーマット]、[ファイル名]、[アクセスサイズ]、[ダウンロード時のメモリベリファイ]オプションの入力	FILE_LOAD
複数のビルド ***1**	-	[ビルド]または[すべてをビルド]ボタンのクリック	BUILD_MULTIPLE
		[クリーン]ボタンのクリック	CLEAN
ビルドコンフィグレーション	SimDebug_SH-3	[現在のコンフィグレーション]ドロップダウンリストの切り替え	CHANGE_CONFIGURATION
デバッグセッション	SimSessionSH-3	[現在のセッション]ドロップダウンリストの切り替え	CHANGE_SESSION
プログラムステップ	-	[ステップオーバ]チェックボックスのオン	STEP_OVER
		[ステップオーバ]チェックボックスのオフ	STEP

***1.** 機能のサポートは、デバッギングプラットフォームに依存します。

15.5.2 マクロを記録できる機能（デバッギングプラットフォーム依存）

連携ツールの製品パッケージにバンドルされている HEW のデバッギングプラットフォームに依存する機能について、以下の機能を操作するとマクロをスクリプトにしマクロファイルに記録されます。機能のサポートは、デバッギングプラットフォームに依存しますのでエミュレータまたはシミュレータのユーザーズマニュアルおよびヘルプファイルを参照してください。

- **SuperH RISC engine および H8SX, H8S, H8 ファミリ用エミュレータ／シミュレータデバッガ**

マクロの記録中にウィンドウの対象に対して、インプレース編集、カラムのダブルクリック、ポップアップメニューのクリックまたはクリックにより表示されるダイアログボックスで設定すると、下の表の「記録する HEW コマンドラインのコマンド」欄にあるコマンドをマクロファイルに記録します。

メニュー	メニューオプション		ショートカットキー	ツールバーボタン	記録する HEW コマンドラインのコマンド
表示	シンボル	ラベル	Ctrl+Shift+A		「ウィンドウ」を参照してください。
		ウォッチ	Ctrl+W		

		ローカル	Ctrl+Shift+W		

- ウィンドウ

ウィンドウ	対象	機能/操作		記録する HEW コマンドラインのコマンド
ラベル	ポップアップメニュー／ツールバーボタン	追加		SYMBOL_ADD
		削除		SYMBOL_CLEAR
		すべてを削除		SYMBOL_CLEAR
		ロード		SYMBOL_ADD
	BP カラム	ダブルクリックによるブレークポイントの挿入/削除		SET_DISASSEMBLY_SOFT_BREAK
ウォッチ	ポップアップメニュー／ツールバーボタン	自動更新有効化		WATCH_AUTO_UPDATE
		全シンボル自動更新有効化		WATCH_AUTO_UPDATE
		自動更新無効化		WATCH_AUTO_UPDATE
		全シンボル自動更新無効化		WATCH_AUTO_UPDATE
		値更新記録	記録開始	WATCH_RECORD
			記録終了	WATCH_RECORD
		シンボル登録		WATCH_ADD
		値の編集		WATCH_EDIT
		削除		WATCH_DELETE
		全シンボル削除		WATCH_DELETE
		基数	16 進数表示	WATCH_RADIX
			10 進数表示	WATCH_RADIX
			8 進数表示	WATCH_RADIX
			2 進数表示	WATCH_RADIX
	+/-マーク	クリックによる配列の展開／縮小		WATCH_EXPAND
	値	インプレース編集		WATCH_EDIT
ローカル	ポップアップメニュー／ツールバーボタン	編集		WATCH_ADD WATCH_EDIT WATCH_DELETE
	値	インプレース編集		WATCH_ADD WATCH_EDIT WATCH_DELETE

- **SuperH RISC engine および H8SX, H8S, H8 ファミリ用シミュレータデバッガ**

マクロの記録中にウィンドウの対象に対して、インプレース編集、カラムのダブルクリック、ポップアップメニューのクリックまたはクリックにより表示されるダイアログボックスで設定すると、下の表の「記録する HEW コマンドラインのコマンド」欄にあるコマンドをマクロファイルに記録します。

メニュー	メニューオプション		ショートカットキー	ツールバーボタン	記録する HEW コマンドラインのコマンド
表示	CPU	I/O シミュレーション	Ctrl+Shift+I		「ウィンドウ」を参照してください。
	コード	カバレジ	Ctrl+Shift+O		
		トレース	Ctrl+T		
		イベントポイント	Ctrl+E		

- ウィンドウ

ウィンドウ	対象	機能/操作		記録する HEW コマンドラインのコマンド
I/O シミュレーション	ポップアップメニュー／ツールバーボタン	削除		SIMULATEDIO_CLEAR
カバレジ	ポップアップメニュー／ツールバーボタン	すべて有効		COVERAGE
		すべてクリア		COVERAGE
		測定範囲追加		COVERAGE_RANGE
		測定範囲編集		COVERAGE_RANGE
		有効		COVERAGE
		クリア		COVERAGE
		保存		COVERAGE_SAVE
		ロード		COVERAGE_LOAD
トレース	ポップアップメニュー／ツールバーボタン	設定		TRACE_ACQUISITION
イベント - Software Break	ポップアップメニュー／ツールバーボタン	追加	PC ブレークポイント	BREAKPOINT
			ブレークアクセス	BREAK_ACCESS
			ブレークデータ	BREAK_DATA
			ブレークレジスタ	BREAK_REGISTER
			ブレークシーケンス	BREAK_SEQUENCE
			ブレークサイクル	BREAK_CYCLE
		編集	PC ブレークポイント	BREAK_CLEAR BREAKPOINT
			ブレークアクセス	BREAK_CLEAR BREAK_ACCESS
			ブレークデータ	BREAK_CLEAR BREAK_DATA
			ブレークレジスタ	BREAK_CLEAR BREAK_REGISTER
			ブレークシーケンス	BREAK_CLEAR BREAK_SEQUENCE
			ブレークサイクル	BREAK_CLEAR BREAK_CYCLE
		有効		BREAK_ENABLE
		無効		BREAK_ENABLE
		削除		BREAK_CLEAR
		全て削除		BREAK_CLEAR
イベント - Software Event	ポップアップメニュー／ツールバーボタン	追加	PC ブレークポイント	BREAKPOINT
			ブレークアクセス	BREAK_ACCESS
			ブレークデータ	BREAK_DATA
			ブレークレジスタ	BREAK_REGISTER
			ブレークシーケンス	BREAK_SEQUENCE
			ブレークサイクル	BREAK_CYCLE
		編集	PC ブレークポイント	BREAK_CLEAR BREAKPOINT
			ブレークアクセス	BREAK_CLEAR BREAK_ACCESS
			ブレークデータ	BREAK_CLEAR BREAK_DATA
			ブレークレジスタ	BREAK_CLEAR BREAK_REGISTER
			ブレークシーケンス	BREAK_CLEAR BREAK_SEQUENCE
			ブレークサイクル	BREAK_CLEAR BREAK_CYCLE
		有効		BREAK_ENABLE

		無効	BREAK_ENABLE
		削除	BREAK_CLEAR
		全て削除	BREAK_CLEAR

15.6　マクロを実行する

マクロファイルを実行するには下記に示すいくつかの方法があります。

ツールバーまたはメニューからマクロを実行するには

1. [マクロの実行]ボタン()またはマクロツールバーや[ツール]メニューの[マクロの実行]メニュー項目を選択してください。
2. 定義されたマクロが 1 つの場合は自動的にそれを実行します。現在のデフォルトのマクロファイルに複数のマクロがある場合、ダイアログボックスが表示され、実行するマクロを選択できます。
3. 実行中はマウスポインタに[マクロの実行]アイコンがつきます()。
4. OK をクリックするとマクロを実行します。マクロ実行中、[マクロの一時停止]ボタン()および[マクロの停止]ボタン()やメニューが有効になります。実行の一時停止中はマウスポインタに[マクロの一時停止]アイコンがつきます()。

[マクロの設定]ダイアログボックスでマクロの割り当てを設定すると、マクロツールバー、メニュー、またはキーボードのショートカットから簡単にマクロを実行することもできます。割り当ての詳細は「15.8 マクロを割り当てる」を参照してください。

[マクロの設定]ダイアログボックスでマクロを実行することもできます。

[マクロの設定]ダイアログボックスからマクロを実行するには

1. [ツール->マクロの設定]を選択し、[マクロの設定]ダイアログボックスを表示してください。
2. 実行したい機能を含むマクロファイルを選択してください。
3. マクロ一覧から実行したマクロ機能を選択してください。
4. [実行]ボタンをクリックしてください。
5. [マクロの設定]ダイアログボックスが閉じマクロが実行されます。

15.7　マクロを編集する

マクロファイルは HEW ルートディレクトリの"Macro"ディレクトリに格納されています。これらのファイルはテキストベースの HEW コマンドラインのコマンドファイルなので、HEW のエディタウィンドウで手動により編集することもできます。編集結果は次回マクロを使用するときに反映されます。

マクロを編集するには以下の方法もあります。

マクロを編集するには

1. [ツール->マクロの設定]を選択し、[マクロの設定]ダイアログボックスを表示してください。

2. 編集したい機能があるマクロファイルを選択してください。
3. 編集するマクロ機能をマクロ一覧から選択してください。
4. [編集]ボタンをクリックしてください。
5. [マクロの設定]ダイアログボックスが閉じられ HEW のエディタウィンドウでファイルが開きます。

15.8 マクロを割り当てる

カスタムのメニュー項目、ツールバー、またはキーボードのショートカットにマクロを割り当てることができます。これにより、頻繁に使用するマクロにすばやくアクセスすることが可能となります。1 つのマクロを[ツール]メニュー、[マクロ]ツールバー、キーボードのショートカットにすべて同時に割り当てることもできます。カスタムのメニュー項目およびツールバーの表示順序は、アルファベット順になります。

ツールバーボタンにマクロを割り当てるには

1. [ツール->マクロの設定]を選択し、[マクロの設定]ダイアログボックスを表示してください。
2. 割り当てたい機能があるマクロファイルを選択してください。
3. 割り当てるマクロ機能を[マクロ一覧]から選択してください。
4. [割り当て]ボタンをクリックしてください。[割り当ての詳細]ダイアログボックスが表示されます。
5. [ツールバーの割り当て]チェックボックスをオンにしてください。
6. [説明]フィールドでマクロの記述を変更できます。マクロをツールバーボタンに割り当てたときにツールバーでのツールチップとして使用されます。
7. デフォルトのマクロツールバーボタン( )を使用するか、既存のビットマップファイル(*.bmp)から独自のツールバーボタンの画像を指定するかを選択できます。
8. 独自の画像を使用する場合、[参照]ボタンをクリックして標準のファイルブラウザを開いてください。これによりマシン上のファイルを手動で検索できます。
9. OK をクリックすると[マクロ]ツールバーに追加されます。

メニューにマクロを割り当てるには

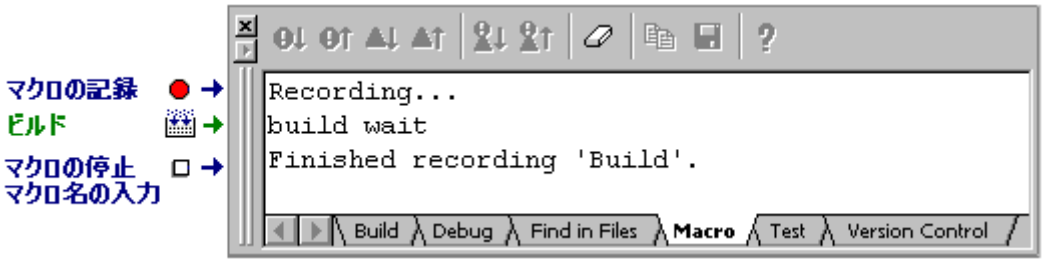
1. [ツール->マクロの設定]を選択し、[マクロの設定]ダイアログボックスを表示してください。
2. 割り当てたい機能があるマクロファイルを選択してください。
3. 割り当てるマクロ機能を[マクロ一覧]から選択してください。
4. [割り当て]ボタンをクリックしてください。[割り当ての詳細]ダイアログボックスが表示されます。
5. [メニュー名の割り当て]チェックボックスをオンにしてください。
6. [説明]フィールドでマクロの記述を変更できます。これはマクロをメニューに割り当てたときにメニューでのツールチップとして使用されます。
7. [ツール]メニューで表示したいメニュー名を入力してください。
8. OK をクリックすると[ツール]メニューに追加されます。

キーボードのショートカットにマクロを割り当てるには

1. [ツール->マクロの設定]を選択し、[マクロの設定]ダイアログボックスを表示してください。
2. 割り当てたい機能があるマクロファイルを選択してください。
3. 割り当てるマクロ機能を[マクロ一覧]から選択してください。
4. [割り当て]ボタンをクリックしてください。[割り当ての詳細]ダイアログボックスが表示されます。
5. [キーボードショートカットの割り当て]チェックボックスをオンにしてください。
6. マクロに割り当てたいキーボードのショートカットをドロップダウンリストで選択してください。
7. OK をクリックするとショートカットが使用可能になります。

15.9 アウトプットウィンドウの Macro タブの構成

アウトプットウィンドウの[Macro]タブは、現在のマクロの記録状況を出力します。[ツール->マクロの記録]による記録の開始から[ツール->マクロの停止]による記録の終了までに HEW マクロファイルに記録した HEW コマンドラインのコマンドなどの情報を出力します。マクロを記録中に記録内容を確認できます。



ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	機能
クリア		ウィンドウ内容をクリアします
保存		ウィンドウ内容をテキストファイルに保存します
コピー		ウィンドウ内容の選択部分を Windows(R)クリップボードにコピーします
ツールバー表示	-	ツールバー表示/表示なしを切り替えます
ツールバーのカスタマイズ	-	ツールバーボタンをカスタマイズします

16. テスト支援機能の使用

ユーザアプリケーションの回帰テストを行うためのテスト支援機能があります。回帰テストとはプログラムの修正後に前回成功したテストを再び実行し、その結果を検証するために実行されるものです。

このテスト支援機能を使用することで、HEW マクロおよびバッチファイルを実行し、HEW システムと元の取得データと比較し一致しているか確認することができます。

様々なテストやスクリプトを作成し、テスト手順を自動化することが可能です。テストが完了すると、結果がテストブラウザウィンドウに表示されます。これにより、複数の回帰テスト実行に必要な作業を大幅に軽減できます。

テスト支援機能では HEW の機能を実行する手段として、コマンドラインバッチファイルを使用します。これらの機能は、マクロ生成支援機能を使用するか、コマンドラインバッチファイルを編集することで作成できます。

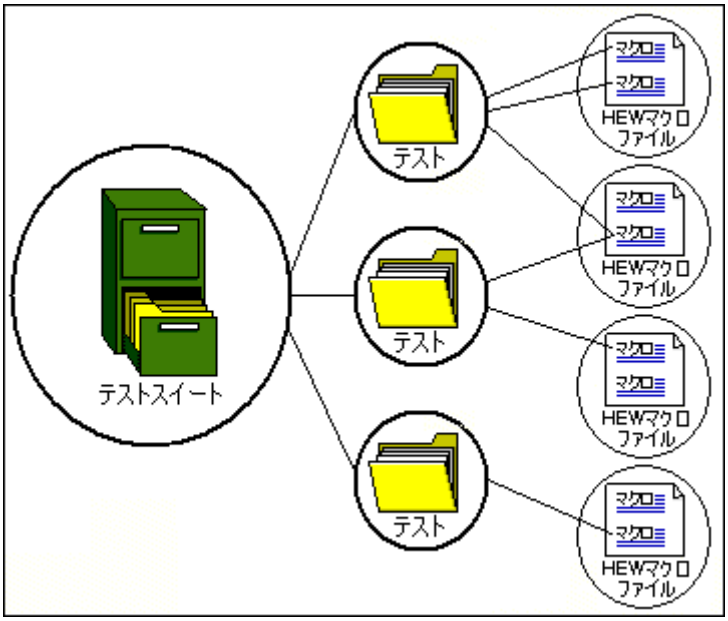
実際のテスト実行を自動化する必要がある場合、テスト支援機能はコマンドラインもサポートしています。テストスイートを開く／閉じる、テストデータを実行／比較するためのコマンドを備えています。

テスト支援機能の使用にあたっての最初のステップは、テストスイートの作成です。[テスト]メニューで作成できます。次にテストスイートを編集し、実際のテストを作成する必要があります。その後、[テストを実行]メニュー項目を使用してテストを実行することができます。

HEW システムのすべての機能のテストイメージデータを保存できるわけではありません。テストイメージデータを保存できる機能については、「16.6 テストイメージファイルにテストイメージデータとして保存できる機能」に記載していますので参照してください。

16.1 テストスイートを作成する

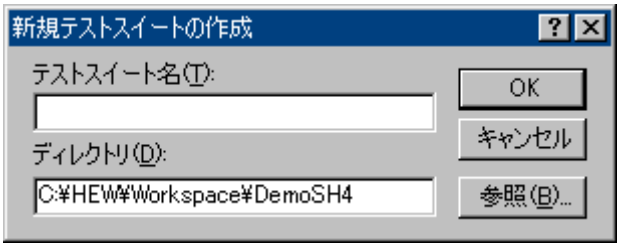
テストスイートはテストの集まりです。テストスイートの概念は、ワークスペースと似ています。実行するマクロを含む複数のテストを、テストスイートに含めることができます。



テストスイートは、現在の HEW ワークスペースに依存しません。このため、テストスイートを複数のワークスペースおよびプロジェクトで動作させ、1 つのテストスイートで様々な状況のテストを行なうことが可能です。テストスイート機能で使用できる"open_workspace"および"change_project"コマンドで、現在使用しているワークスペースを変更することができます。

テストスイートを作成するには

1. [テスト->テストスイートの作成]を選択してください。[新規テストスイートの作成]ダイアログボックスを表示します。



2. [テストスイート名]にテストスイート名を入力します。
3. [ディレクトリ］はあらかじめワークスペースディレクトリが表示されています。必要に応じて編集してください。
4. [OK]ボタンをクリックします。
5. テストスイートが作成されます。[テスト]メニューにある他のアイテムが使用可能になります。

テストスイートを作成すると、ワークスペースウィンドウの[Test]タブにテストスイートが追加されます。このタブにより、テストスイートの操作およびテストへ容易にアクセスできます。

テストスイートファイルは、保存先の場所に拡張子".HTS"（HEW テストスイート）の付いたファイルとして保存されます。

16.2　テストスイートを開く／閉じる

テストスイートを作成すると、拡張子".HTS"（HEW テストスイート）の付いたファイルとして保存されます。以前に作成したテストスイートファイルを開きたい場合は、次の操作を行なってください。

テストスイートを開くには

1. [テスト->テストスイートを開く]を選択してください。[テストスイートを開く]ダイアログボックスを表示します。
2. テストスイートファイルを選択し、[選択]ボタンをクリックしてください。テストスイートを読み込みます。
3. [テスト]メニューにある他のアイテムが使用可能になります。ワークスペースウィンドウの[Test]タブに、テストスイートの内容が表示されます。

最近開いた 4 ファイルを[ファイル->最近使ったテストスイート]のサブメニューに表示します。最近使ったテストスイートを再び開きたいときに使用してください。

テストスイートを閉じるには

[テスト->テストスイートを閉じる]を選択してください。現在のテストスイートが閉じ、ワークスペースウィンドウの[Test]タブからすべてのアイテムが削除されます。

これらの操作は、コマンドラインウィンドウから HEW コマンドラインのコマンドを使用することによっても可能です。対応するコマンドは"open_test_suite"と"close_test_suite"です。

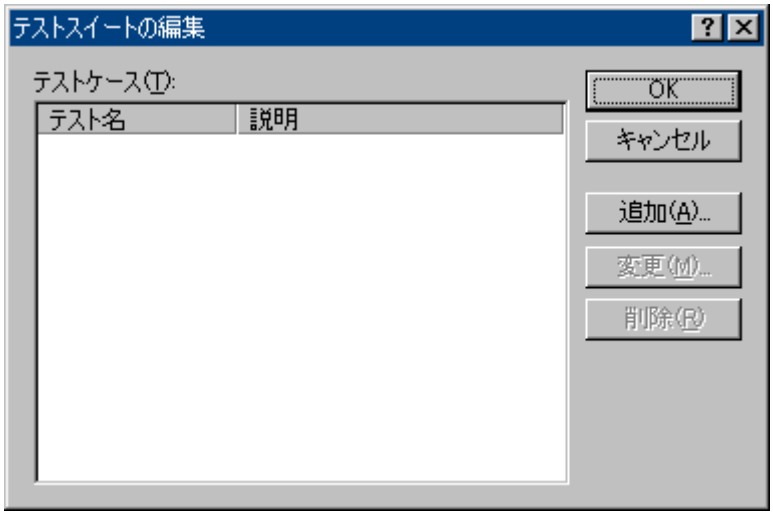
ワークスペースウィンドウの[Test]タブのポップアップメニューからも、テストスイートを閉じることができます。

16.3　テストスイートを編集する

テストスイートを作成したら、次のステップはテストの追加です。この操作により、[テストスイートの編集]ダイアログボックスにアクセスし、実行したいテストを追加することができます。

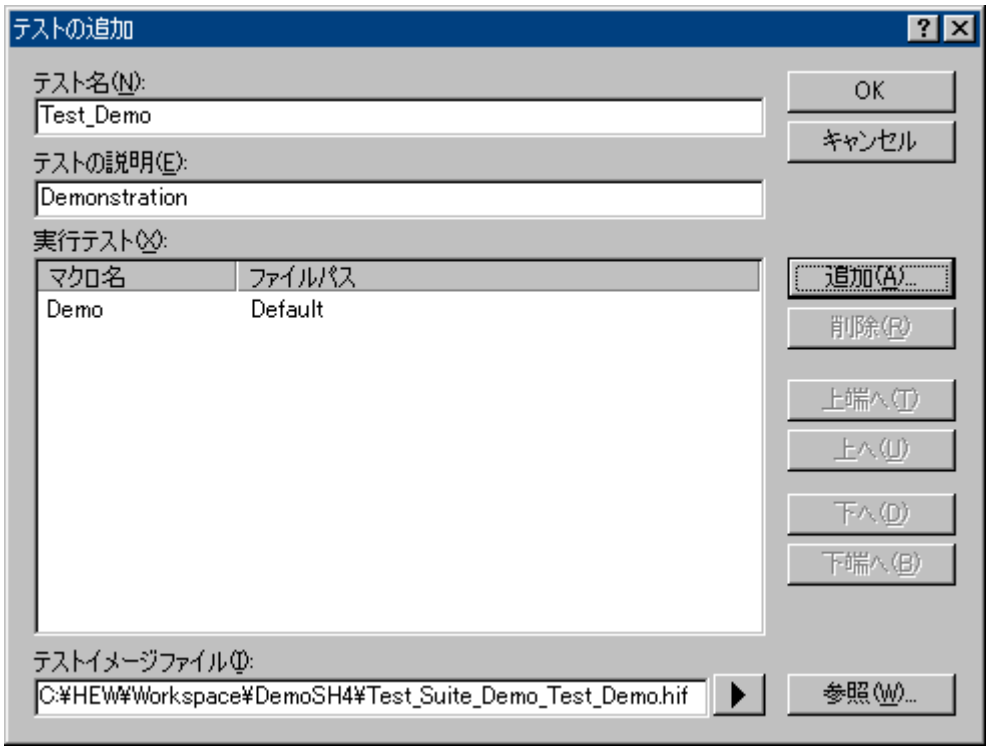
テストスイートを編集するには

1. [テスト->テストスイートの編集]を選択してください。[テストスイートの編集]ダイアログボックスを表示します。



[テストスイートの編集]ダイアログボックスは、現在定義されているテストケースを表示します。各テストケースには名前があり、テストの目的を説明するため、詳細な説明を付けることができます。

このダイアログボックスで[追加]ボタンをクリックすると、[テストの追加]ダイアログボックスが表示されます（下記参照）。



テストを選択し[変更]ボタンをクリックすると、選択したテストについて現在定義されている詳細を変更することができます。これらの詳細は、[テストの追加]で使用したダイアログボックスにも表示されます。[削除]ボタンをクリックすると、テストが削除されます。

[テストの追加]ダイアログボックスでは、後で実行するためのテストを設定できます。詳細は「テストスイートにテストを追加する」で説明します。

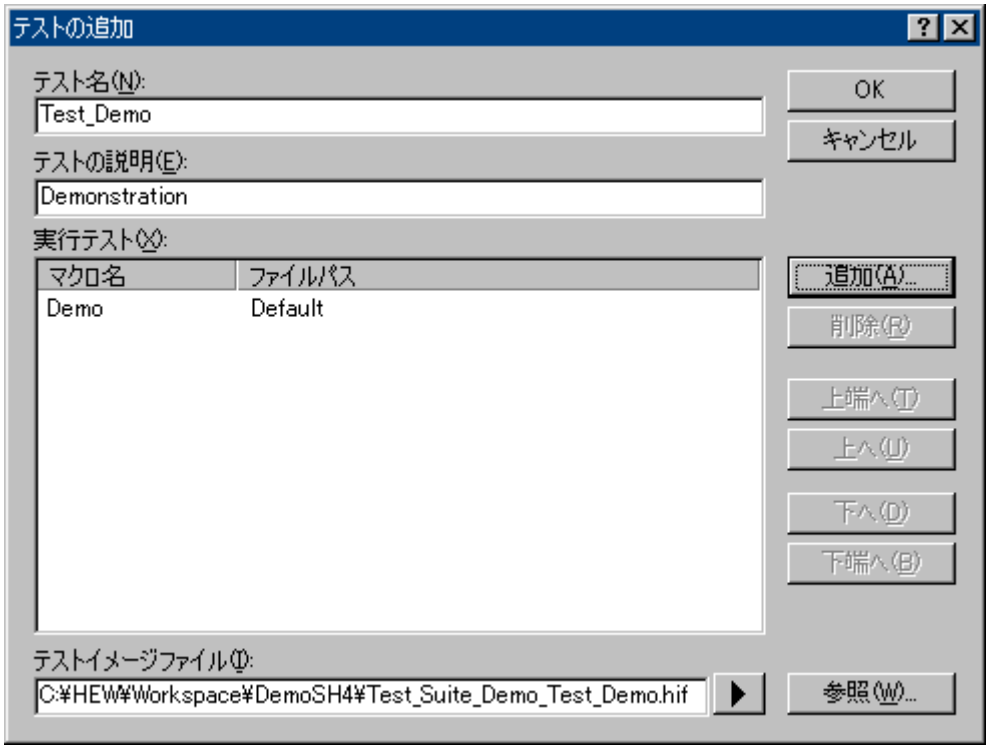
ワークスペースウィンドウの[Test]タブのポップアップメニューからも、テストスイートを編集することができます。

16.4 テストスイートにテストを追加する

テストスイートを作成したら、次のステップはテストの追加です。この操作により、[テストスイートの編集]ダイアログボックスにアクセスし、実行したいテストを追加することができます。

テストスイートにテストを追加するには

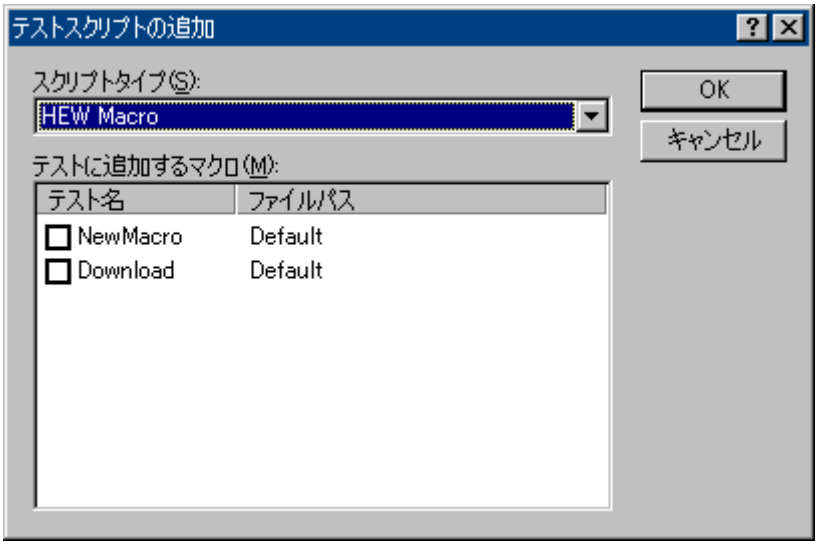
1. [テスト->テストスイートの編集]を選択してください。[テストスイートの編集]ダイアログボックスを表示します。
2. [追加]ボタンをクリックしてください。[テストの追加]ダイアログボックスを表示します。



新しいテストを追加するには以下のデータの設定が必要です:

1. テスト名を入力してください。名前にはスペースが含まれないようにしてください。
2. テストの説明を入力してください。後になってからも分かるよう、詳細な説明を記入してください。
3. [追加]ボタンをクリックし、実行するテストを設定してください。
4. [テストスクリプトの追加]ダイアログボックス（下記参照）が表示されます。
5. [スクリプトタイプ]ドロップダウンリストで、HEW マクロ（"HEW Macro"）または HEW コマンドラインバッチファイル（"TCL command line batch file"）が選択できます。

6. [スクリプトタイプ]ドロップダウンリストで"HEW Macro"を選択すると、すべての登録済みのマクロが[テストに追加するマクロ]リストに表示されます。
7. 各テストに対し、実行したいマクロファイルを複数選択することができます。使用したいマクロ名の横にあるチェックボックスをオンにしてください。
8. [スクリプトタイプ]ドロップダウンリストで"TCL command line batch file"を選択すると、[テストスクリプトの追加]ダイアログボックスが変更され、テスト実行時に実行するファイルを定義することができます。
9. OK をクリックすると、結果を保存し[テストの追加]ダイアログボックスに追加します。
10. 最後に、[テストイメージファイル]を指定することができます。このファイルには、テスト実行後に比較するシステム用の比較データが保存されます。”Test_Demo”テストアイコン上で右クリックしポップアップメニューを表示します。[テストイメージファイルの編集]を選択し[テストイメージファイルの編集]ダイアログボックスを表示します。[テスト->テストイメージファイルの作成]メニューオプションを使用することで、ファイルを作成できます。
11. OK をクリックすると、テストを追加し[テストスイートの編集]ダイアログボックスに追加します。



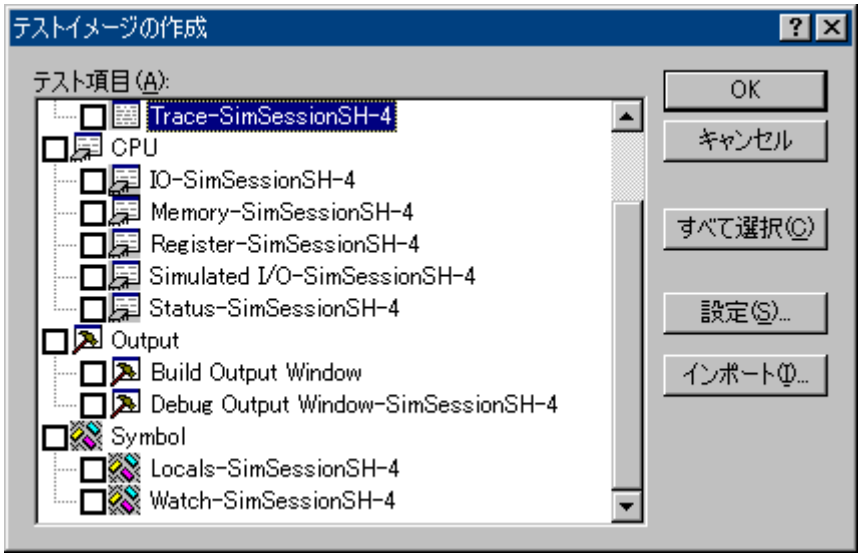
16.5 テストイメージファイルを作成する

テストイメージデータは、テストシステムでの比較に使用されます。比較することができるのは、テストイメージファイルに保存された項目のみです。

例えば、デバイス全体のメモリでなく、特定の範囲のメモリにあるメモリデータのみを比較したいとします。各テストで比較するアイテムの数が多いほど比較処理は遅くなり、複数のテストを実行する場合大きな差になります。

テストイメージファイルを作成するには

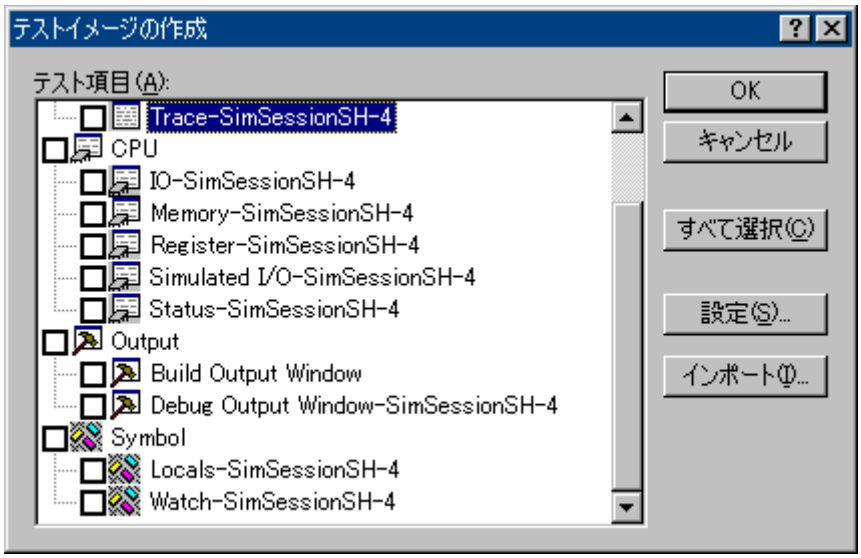
1. [テスト->テストイメージファイルの作成]を選択してください。[テストイメージの作成]ダイアログボックスを表示します。



2. [すべて選択]ボタンをクリックするとすべてのチェックボックスがオンになります。[すべて解除]ボタンをクリックするとすべてのチェックボックスがオフになります。
3. [インポート]ボタンをクリックすると、[テストイメージファイルのインポート]ダイアログボックスが表示されます。インポートしたい HIF ファイルをブラウズしてください。既存のテストイメージファイルの設定をインポートします。
4. ダイアログボックスに各テスト項目を表示します。テスト項目名の横にあるチェックボックスをオンにすると、そのテスト項目のテストイメージデータが保存されます。
5. ダイアログボックスでテスト項目を選択し、[設定]ボタンをクリックしてください。選択されたテスト項目の詳細設定ダイアログボックスが表示されます。テストイメージファイルに保存するテストイメージデータをカスタマイズすることができます。詳細は次節の「16.6 テストイメージファイルにテストイメージデータとして保存できる機能」を参照してください。
6. OK ボタンをクリックします。
7. [テストイメージファイルの保存]ダイアログボックスが表示されます。
8. [テストの追加]ダイアログボックスで指定した HEW テストイメージファイルを選択し保存するか、または新規 HEW テストイメージファイルに保存してください。チェックボックスがオンのテスト項目のテストイメージデータのみがテストイメージファイルに保存されます。

ワークスペースウィンドウの[Test]タブのポップアップメニューからも、テストイメージファイルを編集することができます。

16.6 テストイメージファイルにテストイメージデータとして保存できる機能



HEW V.4.02 では、HEW システムのすべての機能のテストイメージデータをテストイメージファイルに保存できるわけではありません。

テストイメージデータとしてテストイメージファイルに保存できる機能については、下記を参照してください。また、テスト実行後またはテストイメージファイルの比較の際に、各テスト項目のテスト結果が不一致（FAIL）の場合のテスト結果の詳細情報についても記載しています。

- テストイメージファイルに保存できる機能(HEW 共通)
- テストイメージファイルに保存できる機能(デバッギングプラットフォーム依存)

テストイメージデータを保存する際にテスト項目の詳細設定が必要な場合があります。その場合は、[テストイメージの作成]ダイアログボックスで各テスト項目をダブルクリックすると、テスト項目の詳細設定ダイアログボックスが表示されます。

テスト項目の詳細設定があるものは、テストイメージファイルに保存するテストイメージデータをカスタマイズできます。

テスト項目の詳細設定がないものは、テスト項目の左横のチェックボックスがオンの場合にはそのテスト項目のすべてのテストイメージデータがテストイメージファイルに保存されます。

テストイメージデータをテストイメージファイルに保存できる機能については、[表示]メニュー一覧にも”テストイメージデータの保存”としてファイルに保存マーク( )を記載していますので参照してください。

16.6.1 テストイメージファイルに保存できる機能（HEW 共通）

連携ツールの製品パッケージにバンドルされている HEW で共通の機能については、下記のテスト項目のテストイメージファイルに保存するテストイメージデータが選択できます。

機能と[テストイメージの作成]ダイアログボックスのテスト項目との対応を示します。

メニュー	メニューオプション		ショートカットキー	ツールバーボタン	ウィンドウ名	テストイメージの作成ダイアログボックス		
						テストグループ名	テスト項目名	テスト項目の詳細設定の有無
表示	アウトプット		Alt+O		アウトプットウィンドウの[Build]タブ	Output	Build Output Window	なし
					アウトプットウィンドウの[Debug]タブ		Debug Output Window	
	CPU	レジスタ	Ctrl+R		レジスタ	CPU	Register *	
		IO	Ctrl+I		IO		IO *	あり
		ステイタス	Ctrl+U		ステイタス		Status *	
		メモリ	Ctrl+M		メモリ		Memory *	
	Code	スタックトレース	Ctrl+K		スタックトレース	Code	Stack Trace *	

*. デバッギングプラットフォームに接続していない場合はテスト項目を指定できません。

詳細設定がないテスト項目

- Output - Build Output Window

テストイメージファイルに保存するテストイメージデータの内容	アウトプットウィンドウの[Build]タブのすべての表示内容です。	
テスト結果の詳細情報	Original	テストイメージファイルの内容
	New	現在のシステムまたは比較で使用するテストイメージファイルの内容
	表示例	**Original**: **New**: Building - New - Debug

- Output - Debug Output Window

テストイメージファイルに保存するテストイメージデータの内容	アウトプットウィンドウの[Debug]タブのすべての表示内容です。	
テスト結果の詳細情報	Original	テストイメージファイルの内容
	New	現在のシステムまたは比較で使用するテストイメージファイルの内容
	表示例	**Original**: Connected **New**: Step Normal End

- CPU - Register

テストイメージファイルに保存するテストイメージデータの内容	レジスタウィンドウのすべての表示内容です。	
テスト結果の詳細情報	Failed at register	不一致のあったレジスタ名
	Src	テストイメージファイルの内容
	Dest	現在のシステムまたは比較で使用するテストイメージファイルの内容
	表示例	**Failed at register** R11, **Src** = 0x00000000, **Dest** = 0x00000fff

詳細設定ができるテスト項目

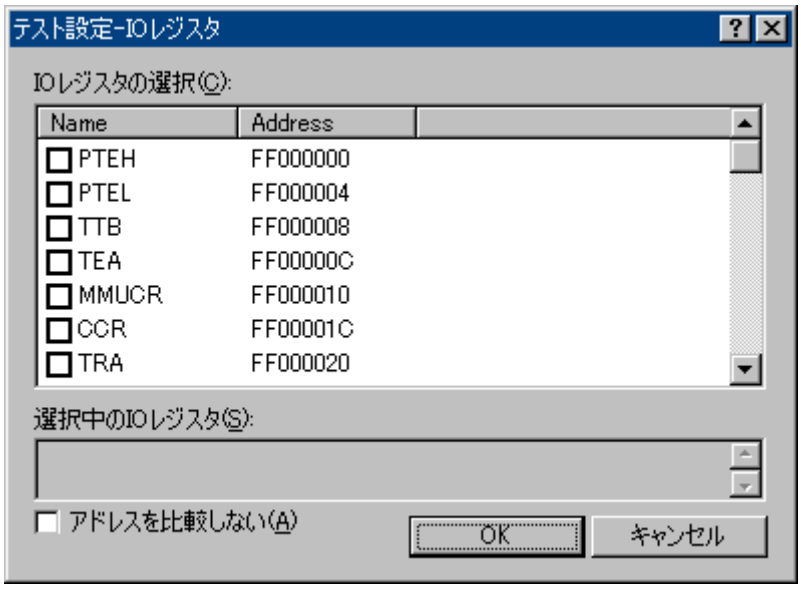
[テストイメージの作成]ダイアログボックスで各テスト項目をダブルクリックすると、テスト項目の詳細設定ダイアログボックスが表示されます。

- CPU - IO

テストイメージファイルに保存するテストイメージデータの内容	テスト項目の詳細設定値、およびテスト項目の詳細設定値で取得した IO ウィンドウの内容です。デフォルトで I/O レジスタは指定されていません。	
テスト結果の詳細情報	Failed at	不一致のあった I/O レジスタ名
	Src	テストイメージファイルの内容
	Dest	現在のシステムまたは比較で使用するテストイメージファイルの内容
	表示例	**Failed at** IPRC, register value is different: **Src** = 0000, **Dest** = FFFF

詳細設定するには

1. [テスト設定-IO レジスタ]ダイアログボックスを表示します。



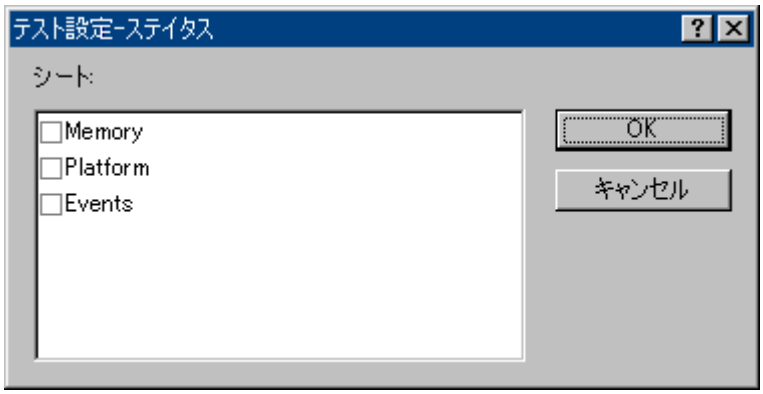
2. テストイメージファイルに保存する I/O レジスタを指定するには、[IO レジスタの選択]リストの I/O レジスタ名のチェックボックスをオンにします。チェックボックスのデフォルトはオフです。選択した I/O レジスタは[選択中の IO レジスタ]に表示されます。
3. 選択した I/O レジスタのアドレスを比較しない場合は、[アドレスを比較しない]チェックボックスをオンにします。チェックボックスのデフォルトはオフです。
4. OK ボタンをクリックします。

- CPU - Status

テストイメージファイルに保存するテストイメージデータの内容	テスト項目の詳細設定値、およびステイタスウィンドウのすべての内容です。	
テスト結果の詳細情報	Sheet	不一致のあったシート名
	Line	不一致のあった行番号
	Src	テストイメージファイルの内容
	Dest	現在のシステムまたは比較で使用するテストイメージファイルの内容
	表示例	Status differs. **Sheet** = Platform **Line** = 6 **Src** = Execute From, Pipeline Reset **Dest** = Execute From, EX Stage

詳細設定するには

1. [テスト設定-ステイタス]ダイアログボックスを表示します。



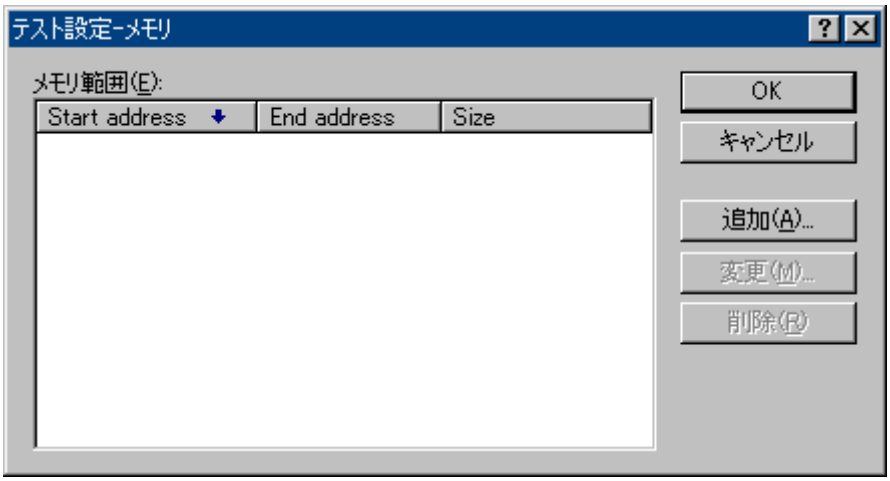
2. テストイメージファイルで比較するシートを指定するには、[Memory]、[Platform]、または[Events]チェックボックスをオンにします。チェックボックスのデフォルトはオフです。
3. OK ボタンをクリックします。

- CPU - Memory

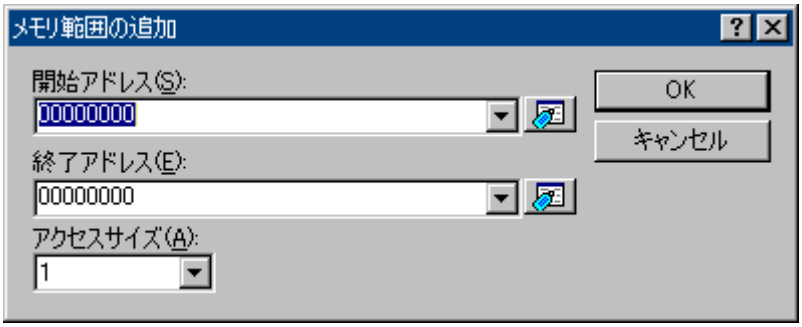
テストイメージファイルに保存するテストイメージデータの内容	テスト項目の詳細設定値、およびテスト項目の詳細設定値で取得したメモリウィンドウの内容です。デフォルトでメモリ範囲は指定されていません。	
テスト結果の詳細情報	Failed at address	不一致のあったアドレス
	Src	テストイメージファイルの内容
	Dest	現在のシステムまたは比較で使用するテストイメージファイルの内容
	表示例	**Failed at address** 0x70000014. **Src** = 0x00002f5a, **Dest** = 0x00002704

詳細設定するには

1. [テスト設定-メモリ]ダイアログボックスを表示します。



2. [テスト設定-メモリ]ダイアログボックスの[追加]ボタンをクリックします。
3. [メモリ範囲の追加]ダイアログボックスが表示されます。



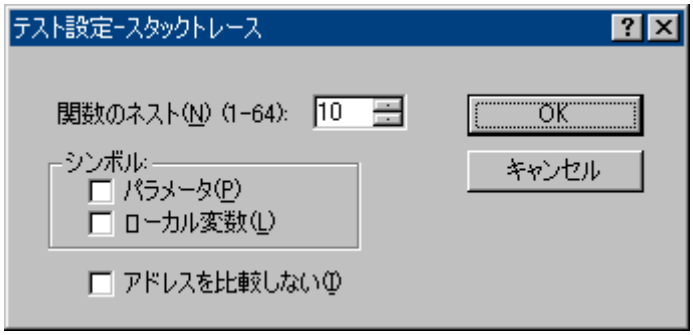
4. テストイメージファイルに保存するメモリ範囲は、[開始アドレス]、[終了アドレス]、および[アクセスサイズ]で指定します。
5. OK ボタンをクリックします。

- Code - StackTrace

テストイメージファイルに保存するテストイメージデータの内容	テスト項目の詳細設定値、およびスタックトレースウィンドウの関数コールのネスト数分の内容です。デフォルトで[関数のネスト]に 10 が指定されています。	
テスト結果の詳細情報	Src	テストイメージファイルの内容
	Dest	現在のシステムまたは比較で使用するテストイメージファイルの内容
	表示例	Value differs. **Src** = F, PowerON_Reset_PC(), { 0000080E } **Dest** = F, PowerON_Reset_PC(), { 0000081C }

詳細設定するには

1. [テスト設定-スタックトレース]ダイアログボックスを表示します。



2. テストイメージファイルに保存するスタックトレース範囲は、[関数のネスト]で指定した関数コールネスト数分の表示内容です。
3. パラメータまたはローカル変数を比較する場合は、[パラメータ]、[ローカル変数]チェックボックスをオンにします。チェックボックスのデフォルトはオフです。
4. アドレスを比較しない場合は、[アドレスを比較しない]チェックボックスをオンにします。チェックボックスのデフォルトはオフです。
5. OK ボタンをクリックします。

16.6.2 テストイメージファイルに保存できる機能（デバッギングプラットフォーム依存）

連携ツールの製品パッケージにバンドルされている HEW のデバッギングプラットフォームに依存する機能について、下記のテスト項目のテストイメージファイルに保存するテストイメージデータが選択できます。各機能のサポートは、デバッギングプラットフォームに依存しますのでエミュレータまたはシミュレータのユーザーズマニュアルおよびヘルプファイルを参照してください。

- **SuperH RISC engine および H8SX, H8S, H8 ファミリ用エミュレータシミュレータ／デバッガ**

機能と[テストイメージの作成]ダイアログボックスのテスト項目との対応を示します。デバッギングプラットフォームに接続していない場合テスト項目を指定できません。

メニュー	メニューオプション		ショートカットキー	ツールバーボタン	ウィンドウ名	テストイメージの作成ダイアログボックス		
						テストグループ名	テスト項目名	テスト項目の詳細設定の有無
表示	コード	トレース	Ctrl+T		トレース	Code	Trace	あり
	シンボル	ローカル	Ctrl+Shift+W		ローカル	Symbol	Locals	
		ウォッチ	Ctrl+W		ウォッチ		Watch	

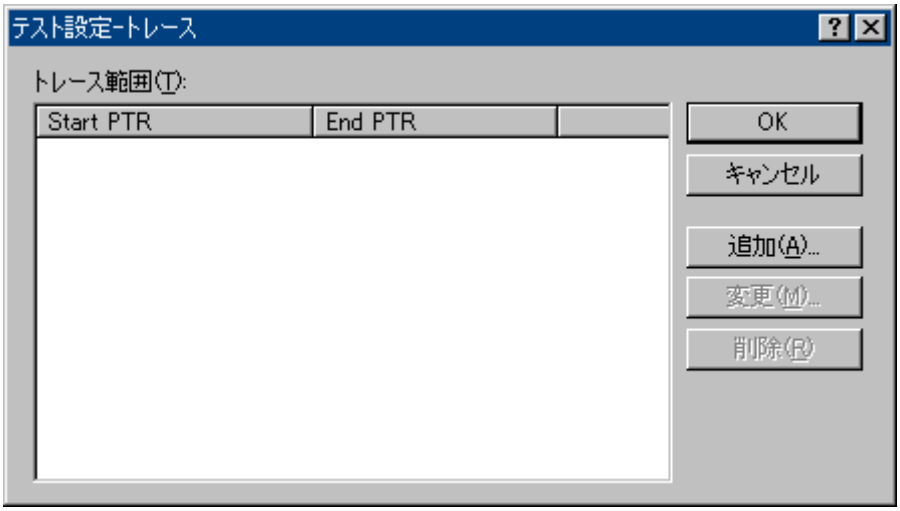
[テストイメージの作成]ダイアログボックスで各テスト項目をダブルクリックすると、テスト項目の詳細設定ダイアログボックスが表示されます。

- Code - Trace

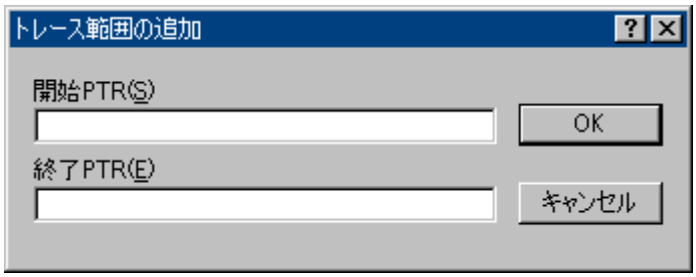
テストイメージファイルに保存するテストイメージデータの内容	テスト項目の詳細設定値、およびテスト項目の詳細設定値で取得したトレースウィンドウの内容です。デフォルトでトレース範囲は指定されていません。	
テスト結果の詳細情報	Comparing PTR	比較したトレース範囲
	Trace type	トレースタイプ
	Trace data is not matching. PTR	不一致のあったトレース範囲
	Src	テストイメージファイルの内容
	Dest	現在のシステムまたは比較で使用するテストイメージファイルの内容
	表示例	**Comparing PTR** -3 to -1. **Trace type** : トレース. **Trace data is not matching. PTR**:-3 to PTR:-1 Trace data at the beginning of difference. PTR:-3 **Src** : 0000001879 0000107C F->DEMMW MOV.L @R15+, R1 R1<-00000010 **Des**t: 0000001887 00000818 FD<E JSR @R5 PC<-00001000

詳細設定するには（トレース種別が単一の場合）

1. [テスト設定-トレース]ダイアログボックスを表示します。



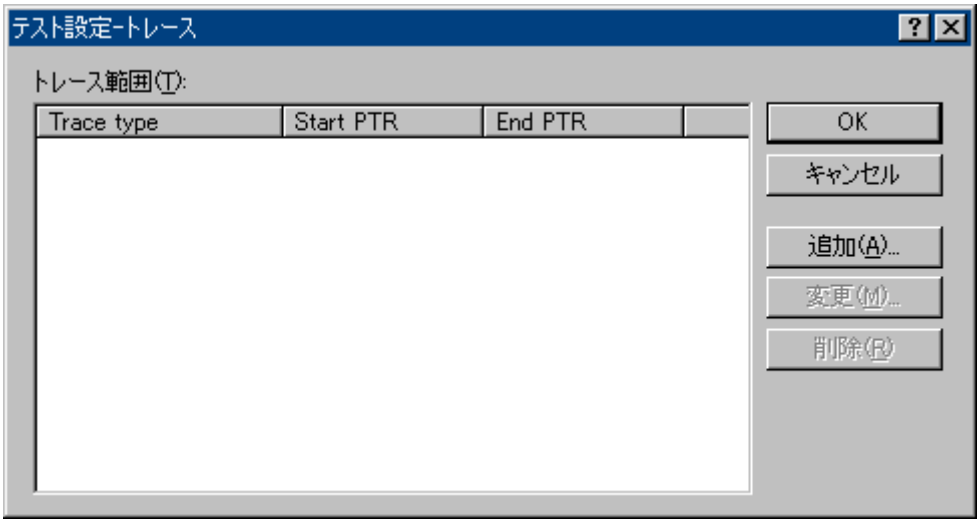
2. [テスト設定-トレース]ダイアログボックスの[追加]ボタンをクリックします。
3. [トレース範囲の追加]ダイアログボックスが表示されます。



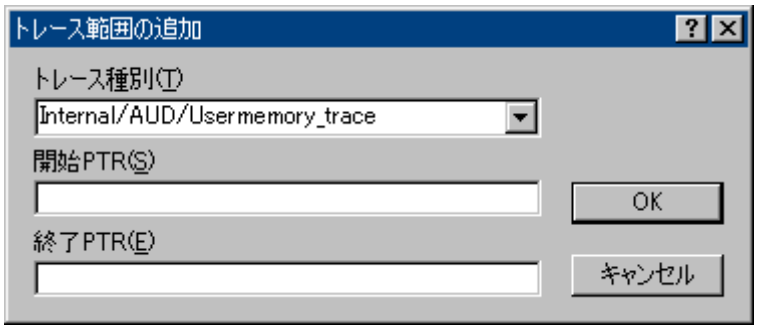
4. テストイメージファイルに保存するトレース範囲は、[開始 PTR]および[終了 PTR]で指定します。
5. OK ボタンをクリックします。
6. [テスト設定-トレース]ダイアログボックスで必要に応じて[変更]、または[削除]できます。
7. OK ボタンをクリックします。

詳細設定するには（トレース種別が複数の場合）

1. [テスト設定-トレース]ダイアログボックスを表示します。



2. [テスト設定-トレース]ダイアログボックスの[追加]ボタンをクリックします。
3. [トレース範囲の追加]ダイアログボックスが表示されます。



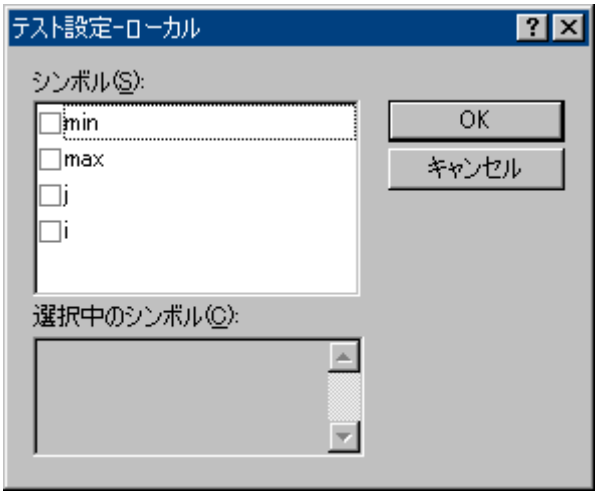
4. テストイメージファイルに保存するトレース範囲は、[トレース種別]、[開始 PTR]、および[終了 PTR]で指定します。
5. OK ボタンをクリックします。
6. [テスト設定-トレース]ダイアログボックスで必要に応じて[変更]、または[削除]できます。
7. OK ボタンをクリックします。

- Symbol - Locals

テストイメージファイルに保存するテストイメージデータの内容	テスト項目の詳細設定値、およびテスト項目の詳細設定値で取得したローカルウィンドウの内容です。デフォルトで各シンボルのチェックボックスはオフです。	
テスト結果の詳細情報	Failed at symbol	不一致のあったシンボル名
	Src	テストイメージファイルの内容
	Dest	現在のシステムまたは比較で使用するテストイメージファイルの内容
	表示例	**Failed at symbol** i, Src = (int)H'00000001{R14}, Dest = (int)H'00000000{R14}

詳細設定するには

1. [テスト設定-ローカル]ダイアログボックスを表示します。



2. テストイメージファイルに保存するシンボルを指定するには、シンボルのチェックボックスをオンにします。チェックボックスのデフォルトはオフです。選択したシンボルは[選択中のシンボル]に表示されます。
3. OK ボタンをクリックします。

- Symbol - Watch

テストイメージファイルに保存するテストイメージデータの内容	テスト項目の詳細設定値、およびテスト項目の詳細設定値で取得したウォッチウィンドウの内容です。デフォルトで各シンボルのチェックボックスはオフです。	
テスト結果の詳細情報	Failed at symbol	不一致のあったシンボル名
	Src	テストイメージファイルの内容
	Dest	現在のシステムまたは比較で使用するテストイメージファイルの内容
	表示例	**Failed at symbol** a[6], **Src** = (long)H'00002704{70000018}, **Dest** = (long)H'00000daa{70000018}

詳細設定するには

1. [テスト設定-ウォッチ]ダイアログボックスを表示します。

2. テストイメージファイルに保存するシンボルを指定するには、[ウォッチシート]を選択しシンボルのチェックボックスをオンにします。チェックボックスのデフォルトはオフです。選択したシンボルは[選択中のシンボル]に表示されます。
3. OK ボタンをクリックします。

- **SuperH RISC engine および H8SX, H8S, H8 ファミリ用シミュレータデバッガ**

機能と[テストイメージ作成]ダイアログボックスのテスト項目との対応を示します。デバッギングプラットフォームに接続していない場合はテスト項目を選択できません。

メニュー	メニューオプション		ショートカットキー	ツールバーボタン	ウィンドウ名	テストイメージ作成ダイアログボックス		
						テストグループ名	テスト項目名	テスト項目の詳細設定の有無
表示	CPU	I/O シミュレーション	Ctrl+Shift+I		I/O シミュレーション	CPU	Simulated I/O	あり
	コード	カバレジ	Ctrl+Shift+O		カバレジ	Code	Coverage	

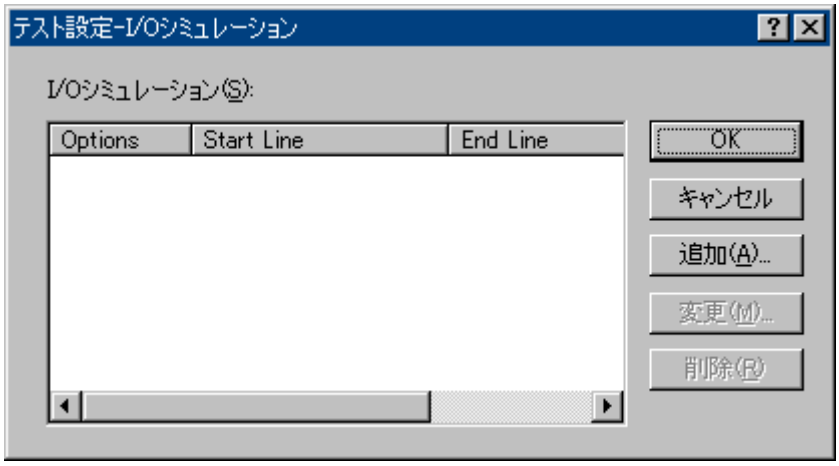
[テストイメージの作成]ダイアログボックスで各テスト項目をダブルクリックすると、テスト項目の詳細設定ダイアログボックスが表示されます。

- CPU - Simulated I/O

テストイメージファイルに保存するテストイメージデータの内容	テスト項目の詳細設定値、およびテスト項目の詳細設定値で取得した I/O シミュレーションウィンドウの内容です。デフォルトで保存する I/O シミュレーション範囲は指定されていません。	
テスト結果の詳細情報	Failed at Line	不一致のあった行番号
	src	テストイメージファイルの内容
	dest	現在のシステムまたは比較で使用するテストイメージファイルの内容
	表示例	**Failed at Line** 1: **src** = ### Data Input ### **dest** = a[1]=21468

詳細設定するには

1. [テスト設定-I/O シミュレーション]ダイアログボックスを表示します。



2. [テスト設定-I/O シミュレーション]ダイアログボックスの[追加]ボタンをクリックします。
3. [I/O シミュレーションの追加]ダイアログボックスが表示されます。

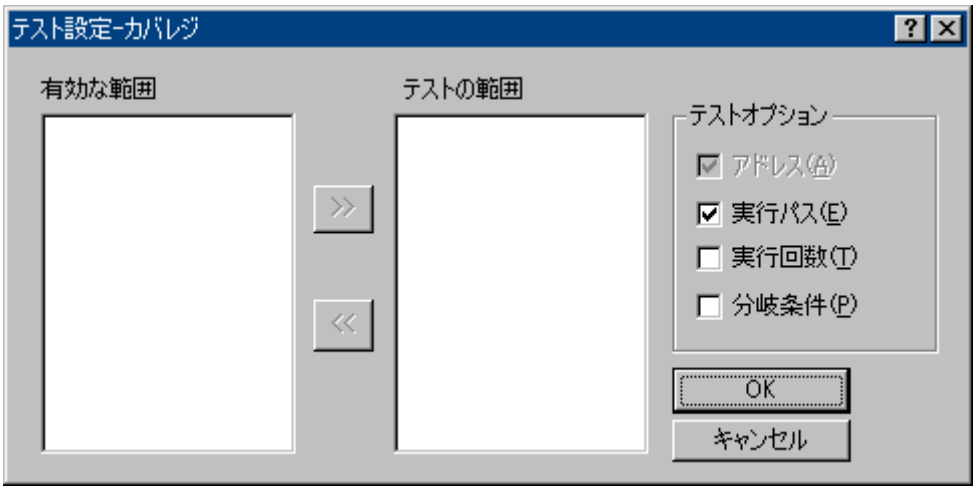
4. テストイメージファイルに保存する I/O シミュレーション範囲は、[全て]または[行番号]で指定します。
5. OK ボタンをクリックします。
6. [テスト設定-I/O シミュレーション]ダイアログボックスで必要に応じて[変更]、または[削除]が行えます。
7. OK ボタンをクリックします。

- Code - Coverage

テストイメージファイルに保存するテストイメージデータの内容	テスト項目の詳細設定値、およびテスト項目の詳細設定値で取得したカバレジウィンドウの内容です。	
テスト結果の詳細情報	Range	カバレジ測定範囲
	Instruction Execution	不一致のあったテストオプション
	mismatch at	不一致のあったアドレス
	Image	テストイメージファイルの内容
	System	現在のシステムまたは比較で使用するテストイメージファイルの内容
	表示例	**Range** DemoSH4.c: **Instruction Execution** Time **mismatch at** 0x20E4 -- **Image**: 1 **System**: 0

詳細設定するには

1. [テスト設定-カバレジ]ダイアログボックスを表示します。



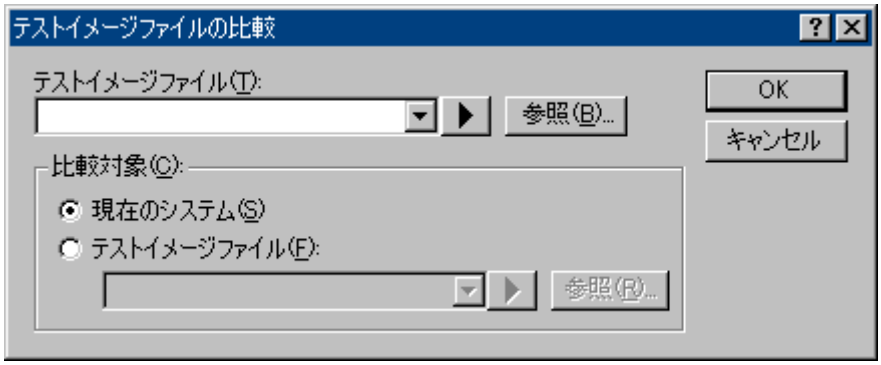
2. [有効な範囲]には設定されているカバレジ測定範囲が表示されます。
3. テストイメージファイルに保存するカバレジ測定範囲は、[>>]ボタンをクリックし[テストの範囲]に移動します。

4. テストイメージファイルにカバレジ測定範囲を保存する場合、アドレスは常に保存するため、テストオプションの[アドレス]チェックボックスは常にオンです（チェックボックスは選択できません）。テストオプションの[実行パス]、[実行回数]、または[分岐条件]チェックボックスをオンにします。[実行パス]チェックボックスのデフォルトはオンです。[実行回数]、または[分岐条件]チェックボックスのデフォルトはオフです。
5. OK ボタンをクリックします。

16.7 テストイメージファイルを比較する

テストイメージファイルを比較するには

1. [テスト->テストイメージファイルの比較]を選択してください。[テストイメージファイルの比較]ダイアログボックスを表示します。



2. [テストイメージファイル]に、比較したいテストイメージファイルのある場所を入力してください。このファイルは"*.HIF"ファイル(HEW テストイメージファイル)で、中にはテストイメージデータの詳細が保存されています。
3. 次に選択したファイルと比較する対象を選択してください。現在のシステム、または以前に保存された別のテストイメージファイルを選択できます。スクリプトを手動で実行し、現在のテストイメージデータを過去に保存したテストイメージデータと比較して確認したい場合は、[現在のシステム]オプションが便利です。
4. OK ボタンをクリックしてください。
5. テストブラウザに結果が読み込まれます。

テスト実行時でなくとも、テストイメージファイルの比較が可能です。実際にテストを実行した後に、現在のシステムまたは 2 つのテストイメージファイルを手動で比較することができます。

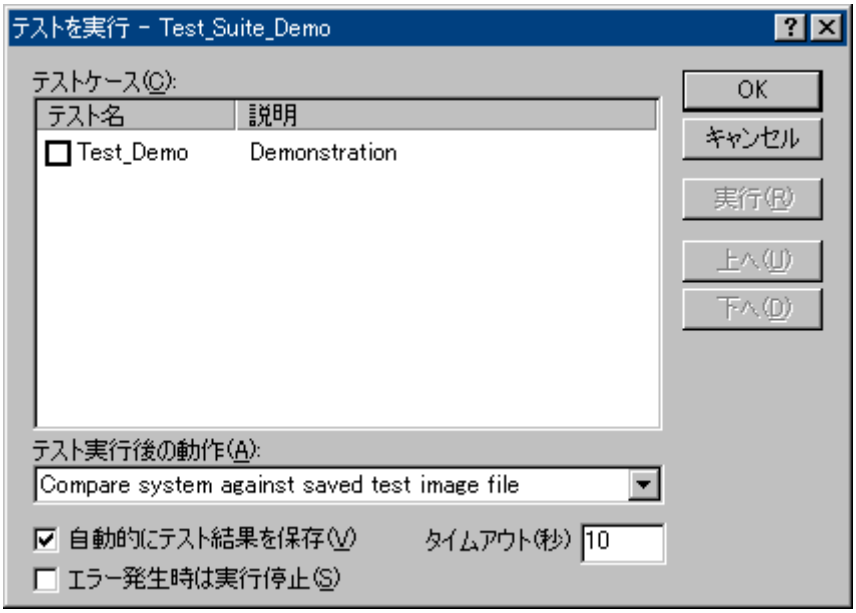
ワークスペースウィンドウの[Test]タブのポップアップメニューからも、テストイメージファイルの比較を行なうことができます。

16.8 テストを実行する

テストを選択し、その実行を自動化することができます。

テストを実行するには

1. [テスト->テストを実行]を選択してください。[テストを実行]ダイアログボックスを表示します。



2. HEW テストシステムで現在定義されているすべてのテストが、[テストケース]リストに表示されます。
3. チェックボックスをクリックすると、今回実行するテストが選択されます。
4. テストを選択して[上へ]および[下へ]ボタンをクリックすることで、テストの実行される順序を変更することもできます。
5. 1 つ以上のテストが選択されていれば、[実行]ボタンが使用可能です。[実行]ボタンをクリックすると、テストの実行を開始します。テスト実行中、アウトプットウィンドウの[Test]タブに情報が表示されます。
6. テストが完了するとテストブラウザが開き、実行されたすべてのテストの結果を表示します。

[テスト実行後の動作]ドロップダウンリストには 2 つのオプションがあります。

- "Compare system against saved test image file"オプションは、通常の動作で、関連するテストケースに添付されたテストイメージファイル(*.HIF)と現在の HEW システムを比較します。これらの結果はテストブラウザに追加され、テストの成功／失敗、失敗した理由が情報として提供されます。
- "Refresh test image file"オプションは、テストケースを実行後に関連するテストケースに添付されたテストイメージファイル(*.HIF)ファイルを更新します。

[自動的にテスト結果を保存]チェックボックスをオンにすると、各テストの実行結果を自動的にテキストファイルに保存することができます。このファイルの場所は、テストスイートと同

じディレクトリになります。ファイル名は、現在のテストスイート名とテスト実行時の日付が付加されています。

[エラー発生時は実行停止]チェックボックスをオンにすると、最初にエラーが発生した時点でテストの実行を停止します。最初のテストが原因で他のテストも失敗してしまう場合、何度もテストを実行せずに済みます。

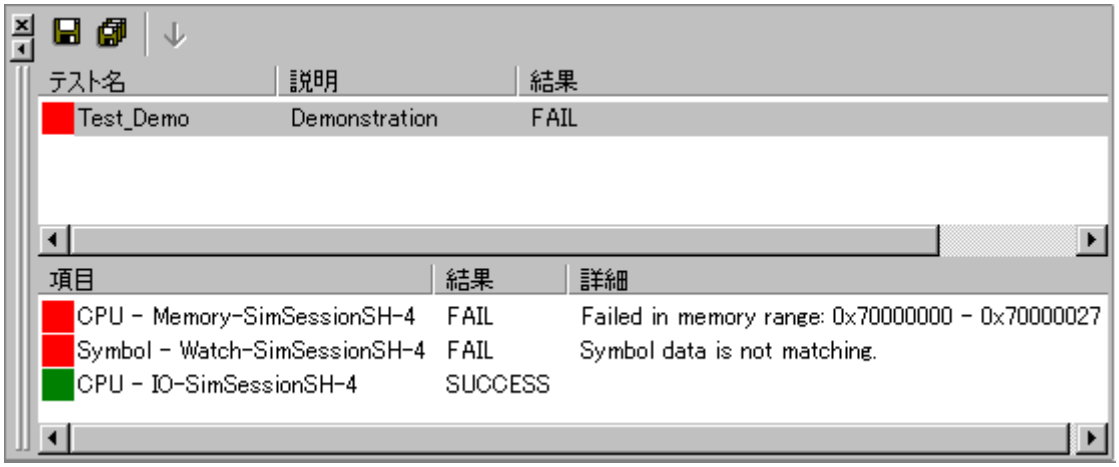
[タイムアウト]はユーザコードを実行する場合に使用します。場合によっては、バグがあるコードが実行されテスト実行中に終了しない可能性があります。テストがこのボックスに入力された秒数よりも長くかかる場合、実行を中止しテストが失敗したとみなします。

ワークスペースウィンドウの[Test]タブのポップアップメニューからも、テストを実行することができます。

16.9 テストブラウザを使用する

テストブラウザは、前回実行したテストの結果を表示します。

[テスト->テスト結果ブラウザ]を選択すると、テストブラウザが開きます。



テストブラウザ上のペインには、前回実行されたすべてのテストが表示されます。テスト名の左にある緑のアイコンは、テストが成功したことを示します。赤のアイコンは、テストが失敗したことを示します。

上のペインでテストを選択すると、その下のペインにテストの詳細が表示されます。テストブラウザの下のペインには、チェックされたすべてのコンポーネントが表示されています。コンポーネント名の左にある緑のアイコンは、テストが成功したことを示します。赤のアイコンは、テストが失敗したことを示します。テストが失敗した場合、このウィンドウに詳細が表示されます。

テスト項目をダブルクリックすると、テストイメージファイルの比較が失敗した原因について詳細な情報が表示されます。テストを選択し、ツールバーの[詳細]ボタンをクリックした場合も、この情報を参照することができます。

テストの結果は、ファイルにエクスポートすることができます。[エクスポート]と[すべてエクスポート]の 2 つのオプションがあります。[エクスポート]を選択すると、現在選択しているテストの結果だけを保存します。[すべてエクスポート]を選択すると、実行したすべてのテストの結果を保存します。

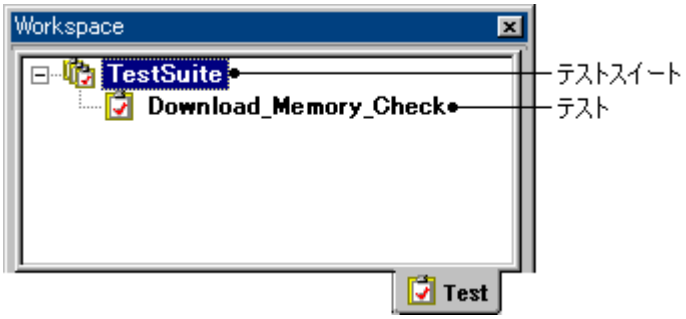
結果は、他の解析ツールにインポートするためのテキスト形式またはコンマ区切りファイル形式で保存されます。

ワークスペースウィンドウの[Test]タブのポップアップメニューからも、[テストブラウザ]を表示することができます。

16.10 ワークスペースウィンドウの Test タブの構成

ワークスペースウィンドウの[Test]タブは、他の[Projects]、[Templates]、[Navigation]タブの横にあります。このタブは、テストスイート内のテストに容易にアクセスできます。

テストスイートを開くと、このテストスイートはワークスペースウィンドウの[Test]タブに追加され、その下にテストスイート内のテストが追加されます。 これを以下に示します。



テストスイート名またはテストを右クリックすると、テスト機能への容易にアクセスできます。

テストスイートのアイコン( )を右クリックすると、次のオプションが使用できます。

ポップアップメニュー名	説明
テストスイートの編集	現在のテストスイートを編集します。HEW テストシステムへのテストの追加および削除ができます。
テストスイートを閉じる	現在開いているテストスイートを閉じます。
テストイメージファイルの比較	テストイメージファイルを、現在の HEW システム、または以前に保存された別のテストイメージファイルと比較します。結果はテストブラウザに表示されます。
テストを実行	テストスイートで定義した複数のテストを実行し、比較した結果をテストブラウザで見ることができます。テストの実行方法を指定する様々なオプションがあります。また、テストイメージファイルに小さな修正を加え更新する必要がある場合、このオプションを使用してテストイメージファイルを自動的にリフレッシュすることもできます。
テスト結果ブラウザ	前回実行したテストの結果を表示します。テストの成功／失敗と、失敗した要因の詳細を見ることができます。
プロパティ	[テストスイートのプロパティ]ダイアログボックスを表示します。テストスイートの名前、説明、場所、最後に修正された日付が表示されます。

テストのアイコン( )を右クリックすると、次のオプションが使用できます。

ポップアップメニュー名	説明
テストケースの実行	選択したテストケースを実行します。
テストケースの編集	選択したテストケースを編集します。
テストケースの削除	選択したテストケースを削除します。
テストイメージファイルの作成	選択したテストケースのテストイメージファイルを作成します。

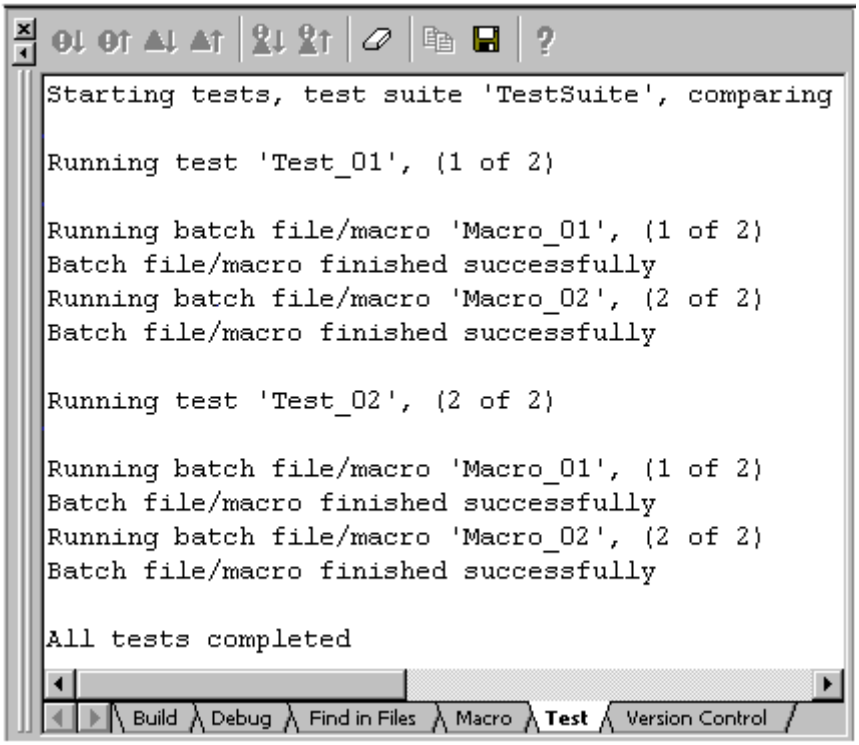
テストイメージファイルの比較	選択したテストケースに設定されたテストイメージファイルを現在のシステムと比較します。
テストイメージファイルの編集	選択したテストケースに設定されたテストイメージファイルを編集します。
テストイメージファイルの保存	現在の HEW システムの状態を、選択したテストケースに設定されたテストイメージファイルに保存します。
名前を付けて保存	現在の HEW システムの状態を、テストイメージファイル作成時のテスト項目への指定を元に、別のテストイメージファイルに保存します。
プロパティ	[テストのプロパティ]ダイアログボックスを表示します。テストの名前、説明、場所、最後に修正された日付が表示されます。

16.11 アウトプットウィンドウの Test タブの構成

アウトプットウィンドウの[Test]タブは、現在のテスト実行の結果および進行状況を出力します。テスト実行の進行状況は、現在実行中のテストと、残りのテストの数で構成されています。エラーが発生した場合、このウィンドウに出力されます。

Starting tests, test suite <テストスイート名>, <比較条件>, <保存条件>, <エラー条件>, <タイムアウト> **Running test**, <テスト名>, (X of Y) **Running batch file/macro**, <マクロ名>, (X of Y)

分類	項目	テストを実行ダイアログボックス		表示例(下図参照)
Starting tests, test suite	<テストスイート名>	-	-	'TestSuite'
	<比較条件>	[テスト実行後の動作]ドロップダウンリストボックス	Compare system against saved test image file (デフォルト)	comparing results
			Refresh test image file	refreshing results
	<保存条件>	[自動的にテスト結果を保存]チェックボックス	☑ (デフォルト)	saving results
			☐	not saving results
	<エラー条件>	[エラー発生時は実行停止]チェックボックス	☑	stop on failure
			☐ (デフォルト)	continue on failure
	<タイムアウト>	[タイムアウト(秒])エディットボックス	任意 (デフォルト 10)	timeout=10
Running test	<テスト名>	-	-	'Test_01' 'Test_02'
	(X of Y) X は現在のテスト番号。 Y はテストの総数	-	-	(1 of 2) (2 of 2)
Running batch file/macro	<マクロ名>	-	-	'Macro_01' 'Macro_02'
	(X of Y) X は現在のマクロ番号。 Y はマクロの総数	-	-	(1 of 2) (2 of 2)



ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	機能
クリア		ウィンドウ内容をクリアします
保存		ウィンドウ内容をテキストファイルに保存します
コピー		ウィンドウ内容の選択部分を Windows(R)クリップボードにコピーします
ツールバー表示	-	ツールバー表示/表示なしを切り替えます
ツールバーのカスタマイズ	-	ツールバーボタンをカスタマイズします

17. デバッガの使用

この章では、主に連携ツールの製品パッケージにバンドルされている HEW で共通のデバッグ機能について説明します。

デバッガの使用に必要な HEW の基本概念は「1.概要」を参照してください。

ご使用のデバッギングプラットフォームで利用可能な機能の詳細説明は、エミュレータまたはシミュレータのユーザーズマニュアルおよびヘルプを参照してください。

17.1 デバッグの準備をする

この節では、プログラムのデバッグを開始するための準備作業について説明します。デバッグを行うためのデバッギングプラットフォームの選択および構築方法、デバッグ対象ファイルのロード、およびデバッグセッションについて説明します。

17.1.1 デバッグの前にビルドを行う

プログラムを C/C++ソースレベルでデバッグするには、C/C++プログラムをビルドして、[Debug]オプションが有効な状態でリンクする必要があります。このオプションが有効の場合、コンパイラは、C/C++コードのデバッグに必要な情報をすべてアブソリュートファイルまたはマネージメントファイルに入れます。これらのファイルはその後「デバッグオブジェクトファイル」とよばれます。プロジェクトを作成すると、初期セットアップで通常のデバッグを設定します。

注：

- デバッグ用のオブジェクトファイルを生成する際はコンパイラおよびリンカの[Debug]オプションを必ず有効にしてください。
- デバッグオブジェクトファイルにデバッグ情報(例えば S レコードフォーマットなど)がない場合でもデバッギングプラットフォームにロードできますが、アセンブリ言語レベルでのデバッグとなります。

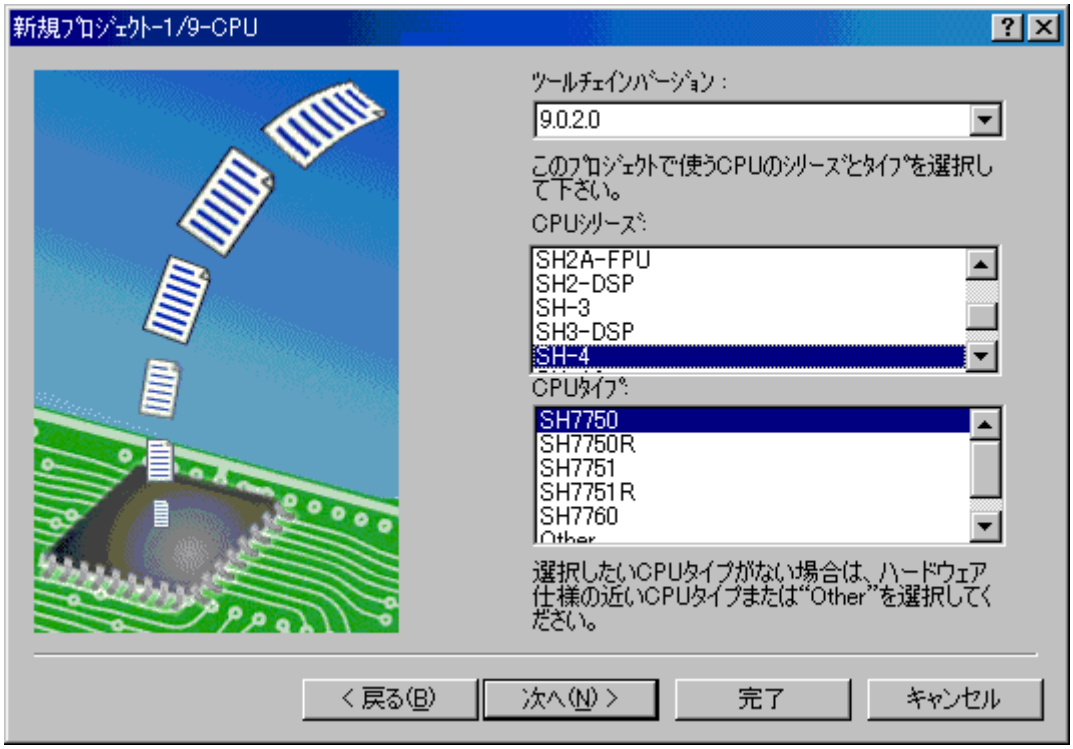
17.1.2 デバッギングプラットフォームを選択する

デバッギングプラットフォームの選択は HEW のインストール方法に大きく依存します。HEW にツールチェインをインストールしている場合、アプリケーションプロジェクトジェネレータはツールチェインおよびデバッグのターゲットを同時にセットアップできます。これによってターゲットおよびツールチェインオプションを一致させ、矛盾が起こらないようにします。
ツールチェインをインストールしていない場合には、デバッグ専用プロジェクトのみ選択することができます。HEW はデフォルトの設定で、[新規プロジェクトワークスペース]ダイアログボックスの中に、生成するそれぞれの CPU ファミリのデバッグ専用プロジェクト生成タイプを表示します。

[新規プロジェクトワークスペース]ダイアログボックスを使用して、ターゲット CPU に合ったプロジェクト生成タイプを選択することができます。

プロジェクトタイプ	説明
[Application]	C/C++言語で記述された初期ルーチンファイルを含む実行プログラムを生成するためのプロジェクト
[Assembly Application]	アセンブリ言語で記述された初期ルーチンファイルを含む実行プログラムを生成するためのプロジェクト
[Demonstration]	C 言語で記述されたデモンストレーションプログラムを作成するためのプロジェクト（Renesas SuperH RISC engine Standard ツールチェイン、または Hitachi H8S,H8/300 Standard ツールチェイン）
[C source startup Application]	C 言語で記述されたスタートアッププログラムを作成するためのプロジェクト（Renesas M16C Standard ツールチェイン、または Renesas M32C Standard ツールチェイン）
[Empty Application]	ツールチェイン環境の設定のみのプロジェクト(生成ファイルなし)
[Import Makefile]	既存のメイクファイルを取り込んで実行プログラムを作成するためのプロジェクト HEW3.0 より、GNU make フォーマットまたは、Hmake フォーマット（HEW が出力）のメイクファイルを解析し、作成するワークスペースにファイル情報を取り込むことができます。
[Library]	ライブラリファイルを作成するためのプロジェクト(生成ファイルなし)
[Debugger only - xxxxxx]	デバッグ専用プロジェクト(生成ファイルなし)

ここでは **SH-4** シミュレータデバッガを例に説明します。ご使用のデバッギングプラットフォームで利用可能な機能の説明は、ユーザーズマニュアルを参照してください。

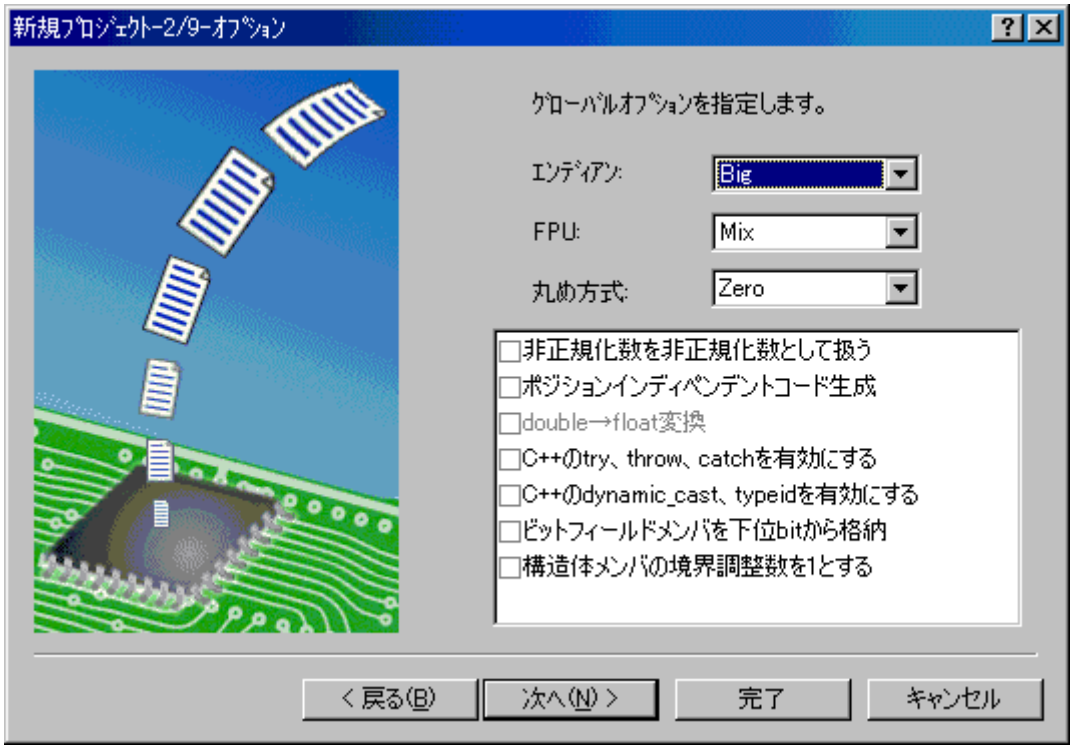


1. ステップ 1 ではツールチェインバージョンと CPU を選択します。使用する CPU の種類([CPU タイプ])は CPU のシリーズ([CPU シリーズ])ごとに分類しています。ツールチェインバージョン、[CPU シリーズ]、および[CPU タイプ]の選択により生成するファイルが異なりますので、開発するプログラムに対応した設定を選択してください。対応する CPU がない場合は、ハードウェア仕様の近い CPU または[Other]を選択してください。

ウィザード下部のボタンは[新規プロジェクト]ウィザード共通です。

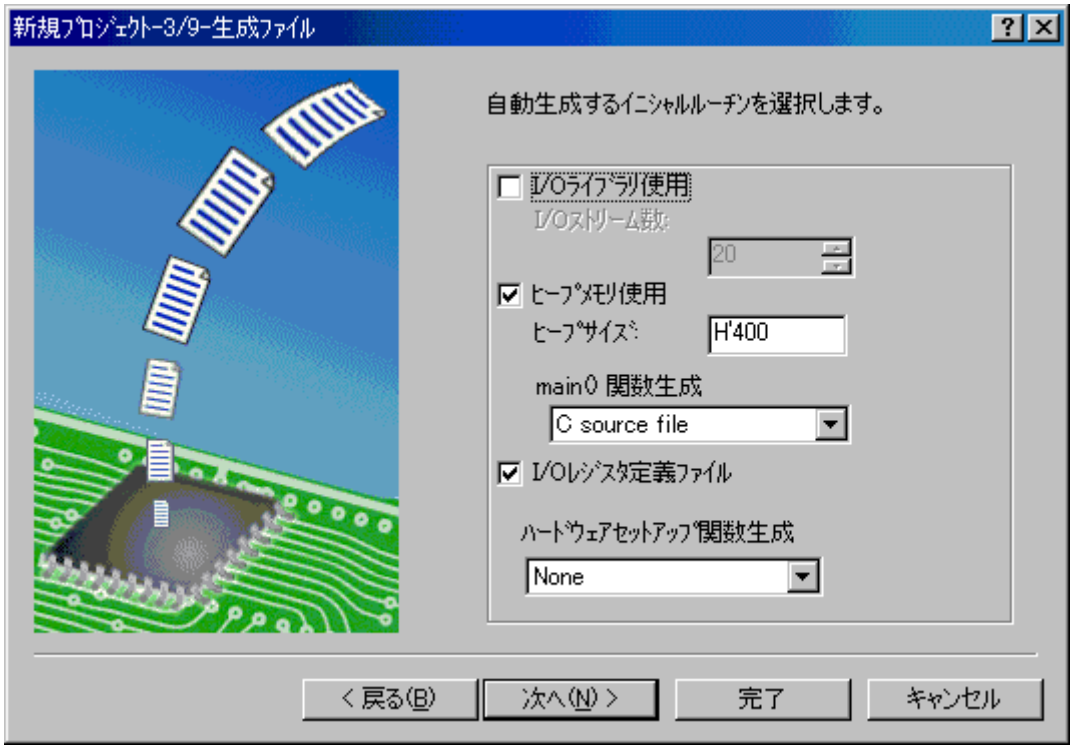
[次へ>]	次の画面に移ります
[<戻る]	前の画面に戻ります
[完了]	以降の選択はデフォルトとなり、[概要]ダイアログボックスを開きます
[キャンセル]	[新規プロジェクトワークスペース]ダイアログボックスに戻ります

ステップ 1 画面で[次へ>]をクリックすると、ステップ 2 に進みます。



2. ステップ 2 では全プロジェクトファイルで共通のオプションを設定します。
   設定できる項目はステップ 1 で選択した CPU に依存します。詳細はコンパイラユーザーズマニュアルで確認してください。

ステップ 2 画面で[次へ>]をクリックすると、ステップ 3 に進みます。



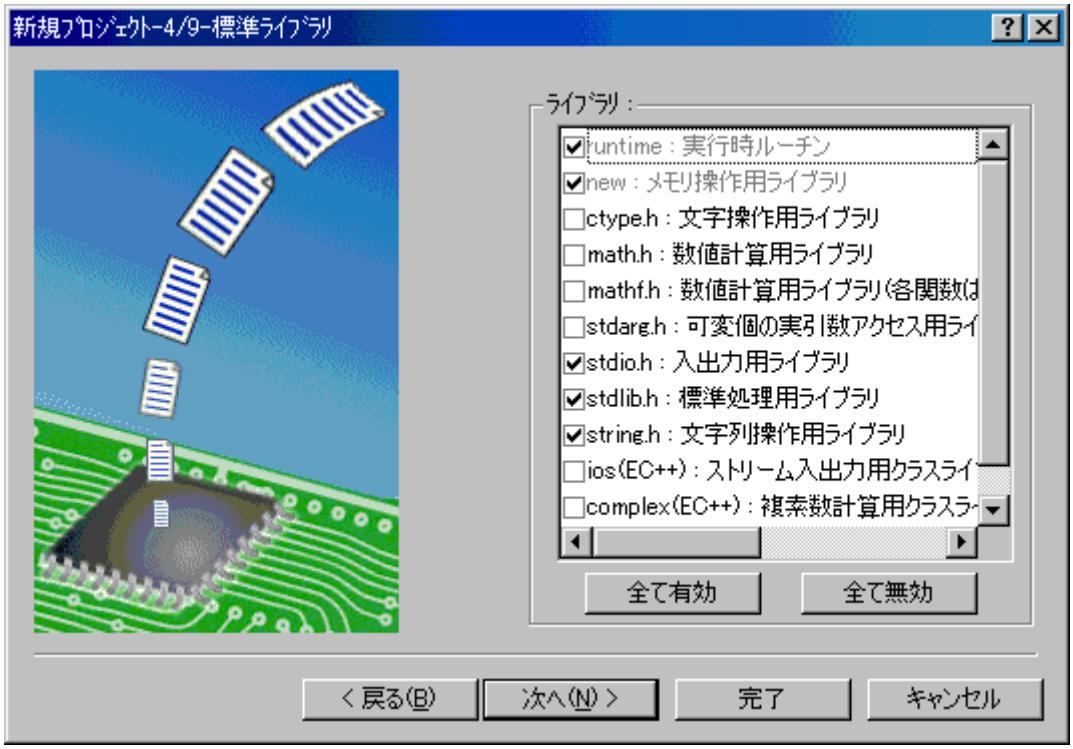
3. ステップ 3 では生成するファイルを設定します。

[I/O ライブラリを使用]	チェックすると標準入出力ライブラリを活用できます。
[I/O ストリーム数]	同時に使用する入出力ストリームの本数を設定します。
[ヒープメモリを使用]	チェックするとヒープ領域管理用の関数 sbrk()を使用できます。
[ヒープサイズ]	管理するヒープ領域サイズ単位を指定します。
[main()関数生成]	main 関数の雛型生成を選択できます。 メイン関数ファイル[(プロジェクト名).c(cpp)]を生成します。
[I/O レジスタ定義ファイル]	チェックすると C 言語で記述した I/O レジスタ定義ファイル(“iodefine.h”)を生成します。
[ハードウェアセットアップ関数生成]	I/O レジスタ初期設定プログラムの雛型生成を選択します。 : ハードウェア設定用のファイル(“hwsetup.c(cpp)” または ”hwsetup.src”)を生成します。

注：

既に作成した main 関数を組み込む場合は、[main()関数生成]で[None]を選択し、プロジェクトを作成後に main 関数を含んだファイルをプロジェクトに追加してください。なお、組み込む関数名が異なる場合は、resetprg.c の関数呼出し部分を修正する必要があります。また、プロジェクトジェネレータが生成するベクタテーブル定義および I/O レジスタ定義等のサンプルファイル内容は、使用する CPU のハードウェアマニュアルで確認してください。

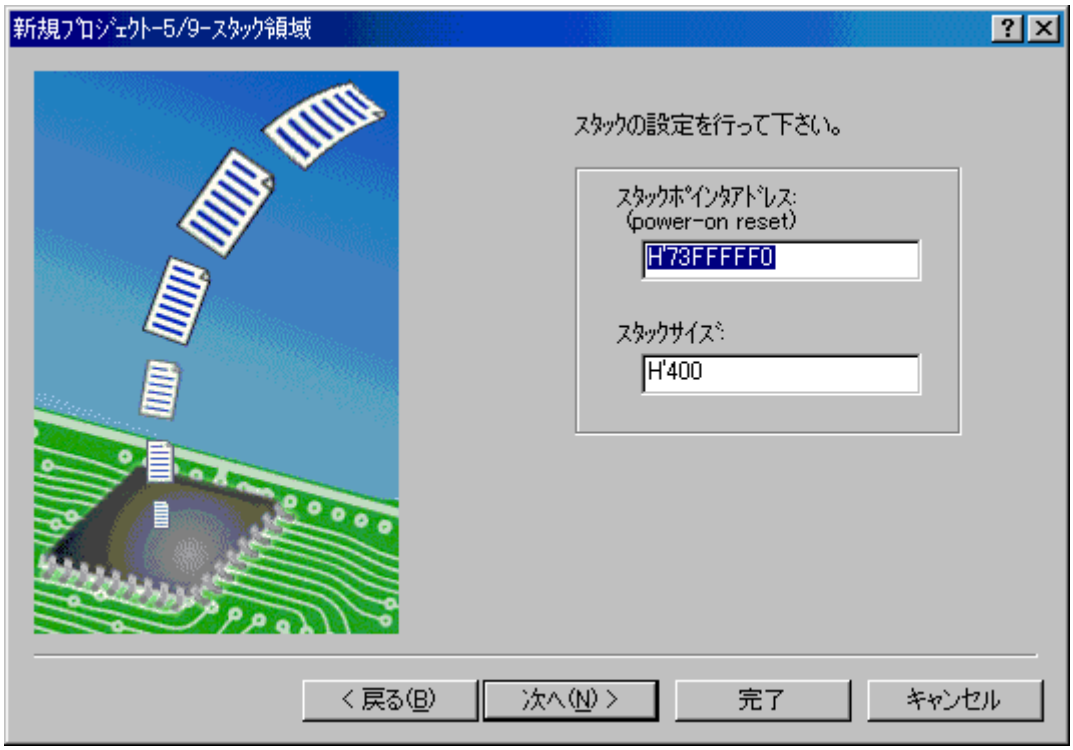
ステップ 3 画面で[次へ>]をクリックすると、ステップ 4 に進みます。



4. ステップ 4 では C/C++コンパイラで使用する標準ライブラリの構成を設定します。
   チェックした項目で宣言されている関数と runtime 関数を組み込みます。

[全て有効]　　すべての標準ライブラリ関数を選択します。
[全て無効]　　すべての標準ライブラリ関数を選択しません。
　　ただし、最低限必要な runtime、new は選択します。

ステップ 4 画面で[次へ>]をクリックすると、ステップ 5 に進みます。

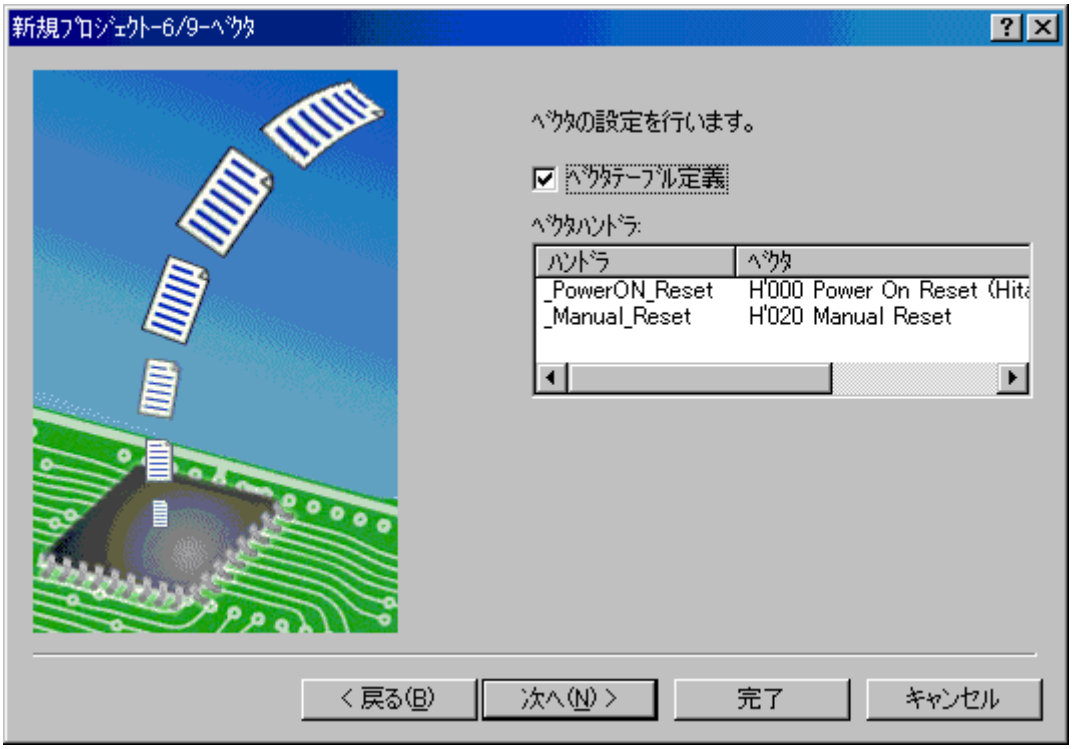


5. ステップ 5 ではスタック領域を設定します。スタックポインタの初期値とスタックサイズを設定します。
   スタック領域の初期値はステップ 1 で選択した CPU に依存します。

注：

スタック領域は HEW が生成する”stacksct.h”で定義しています。”stacksct.h”をエディタで編集した場合は HEW の[プロジェクト->構成の編集]では変更できません。

ステップ 5 画面で[次へ>]をクリックすると、ステップ 6 に進みます。



6. ステップ 6 ではベクタを設定します。

[ベクタテーブル定義]　チェックするとベクタ定義ファイルとベクタテーブル設定関数の定義ファイルを生成します。

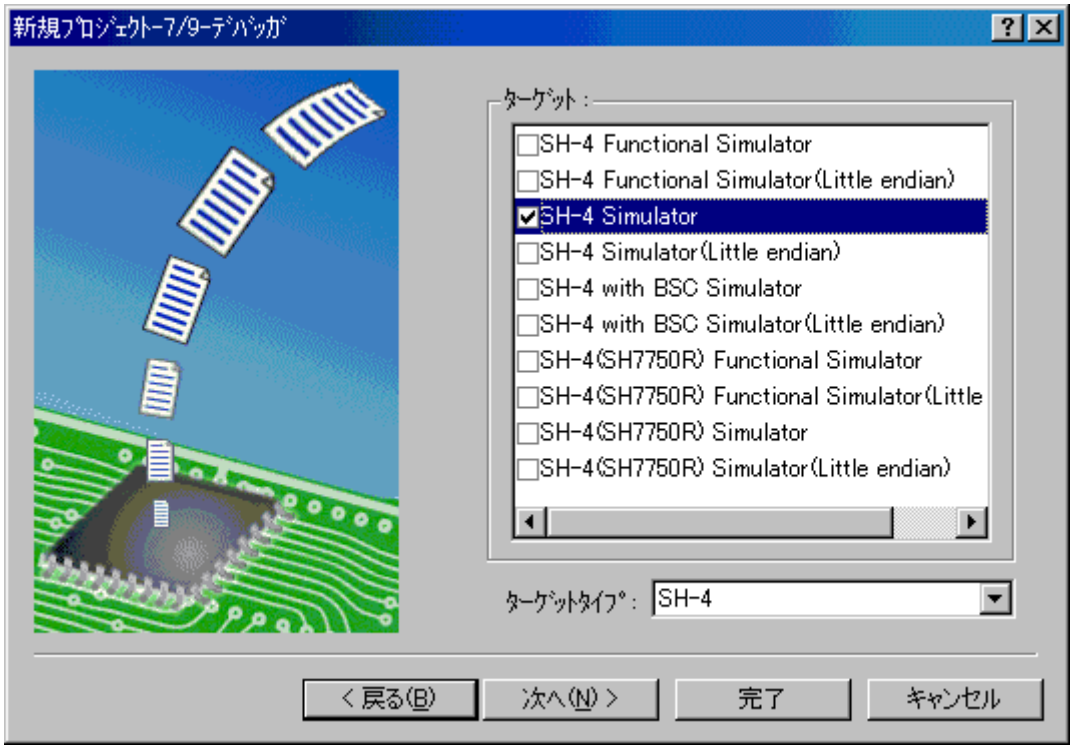
[ベクタハンドラ]　[ハンドラ]　リセットベクタのハンドラプログラム名を表示します。ハンドラプログラムを変更したい場合は、ハンドラプログラム名を選択してクリック後入力してください。なお、ハンドラプログラムを変更すると、リセットプログラム(“resetprg.c”)は生成しません。

[ベクタ]　ベクタの説明を表示します。

注：

生成されたリセットプログラム、割込み関数、リセットベクタハンドラ、および割込み要因レジスタ定義はサンプルであるため、使用される前に CPU のハードウェアマニュアルでご確認ください。

ステップ 6 画面で[次へ>]をクリックすると、ステップ 7 に進みます。



7. ステップ 7 ではデバッガターゲットを設定します。

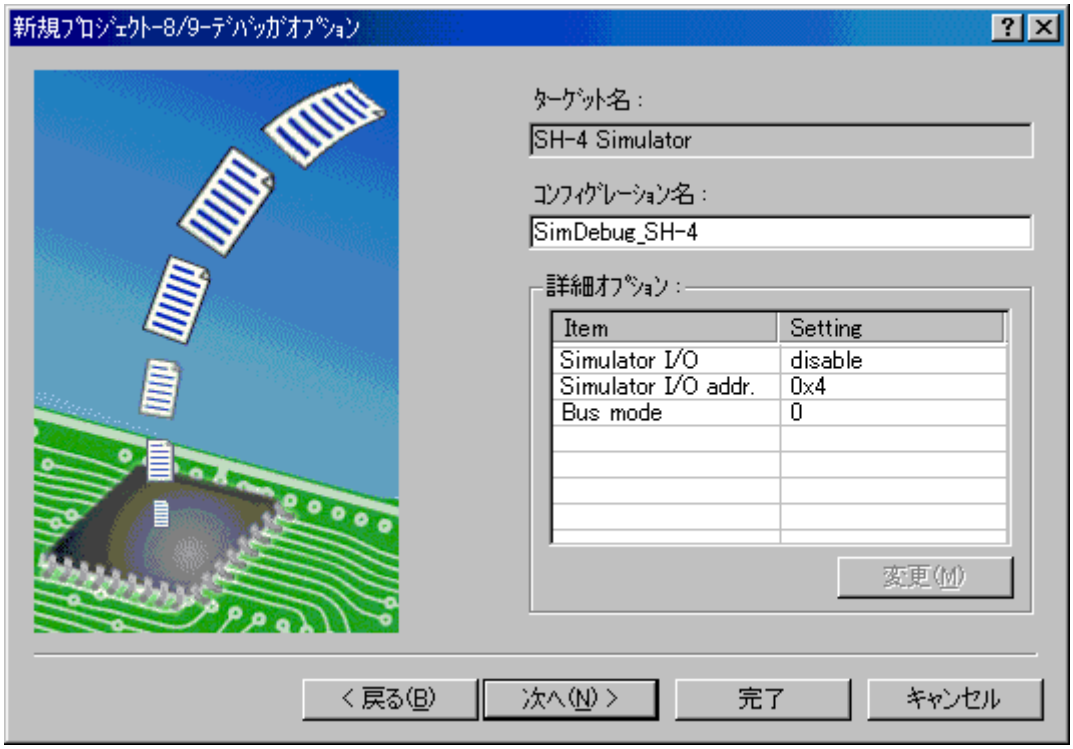
[ターゲット]	デバッガターゲットを設定します。デバッガターゲットを選択(チェック)してください。デバッガターゲットは未選択でも複数選択してもかまいません。
[ターゲットタイプ]	[ターゲット]に表示するターゲットの種類を指定します。

注：

ステップ 2 で設定したエンディアンはコンパイラの設定に反映します。

ステップ 7 で選択したデバッガターゲットのエンディアンとは独立しています。

ステップ 7 画面で[次へ>]をクリックすると、ステップ 8 に進みます。



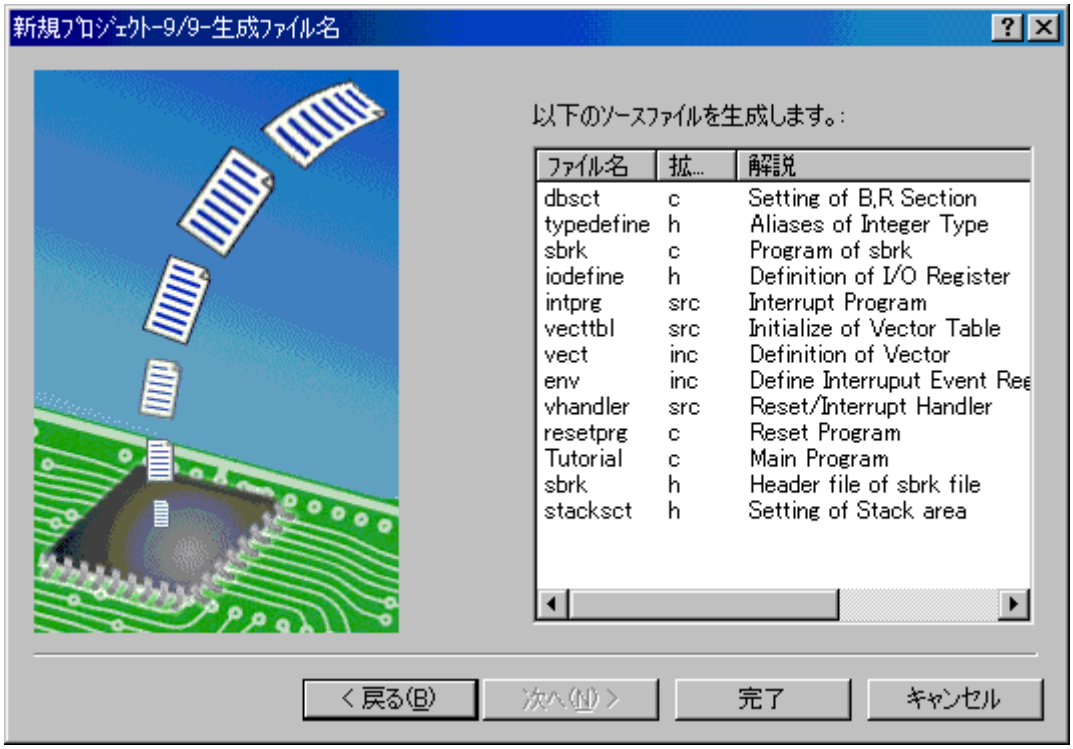
8. ステップ 8 では選択したデバッガターゲットのオプションを設定します。

[コンフィギュレーション名] HEW はデフォルトで[Release]と[Debug]の 2 つのコンフィグレーションを作成しますが、デバッガターゲットを選択すると選択したターゲット用のコンフィグレーションも作成します(ターゲット名を含んだ略称となります)。このコンフィグレーション名は[コンフィギュレーション名]で変更できます。

[詳細オプション] デバッガターゲットのオプションを設定します。変更する場合は、[Item]を選択して[変更]をクリックしてください。なお、変更できない項目の場合、[Item]を選択しても[変更]はグレーのままです。

- [Simulator I/O] ユーザプログラムから標準入出力またはファイル入出力を行うシステムコールは有効([Enable])または無効([Disable])
- [Simulator I/O addr.] 上記システムコールアドレス
- [Bus mode] シミュレータ・デバッガでは現状未使用

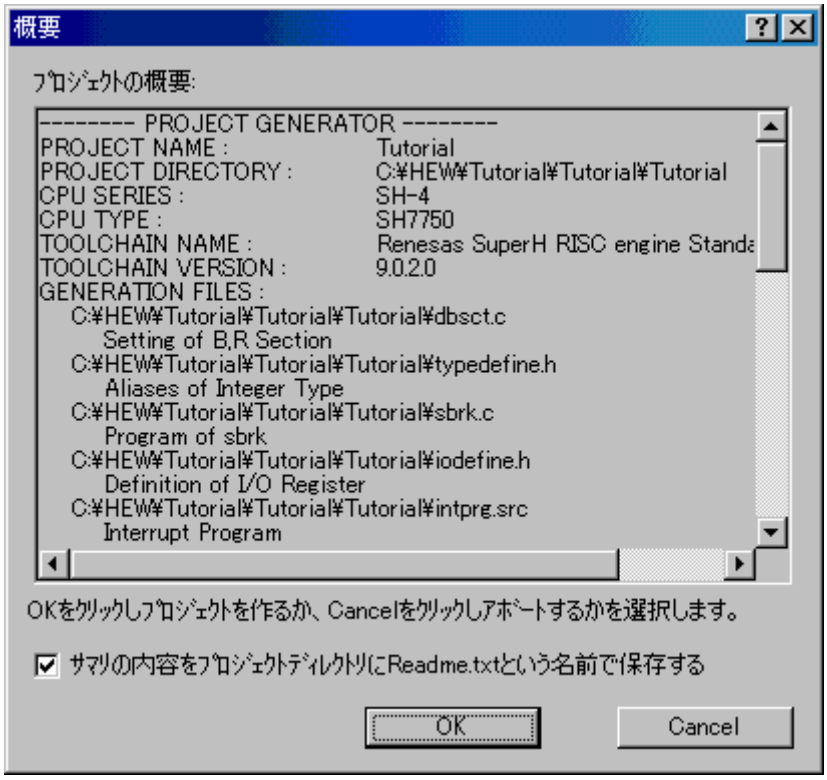
ステップ 8 画面で[次へ>]をクリックすると、ステップ 9 に進みます。



9. ステップ 9 ではこれまでの設定により HEW が生成するファイルをリスト表示します。

[ファイル名]　ファイル名。変更したい場合は、ファイル名を選択してクリック後入力してください。
[拡張子]　拡張子
[解説]　ファイルの説明

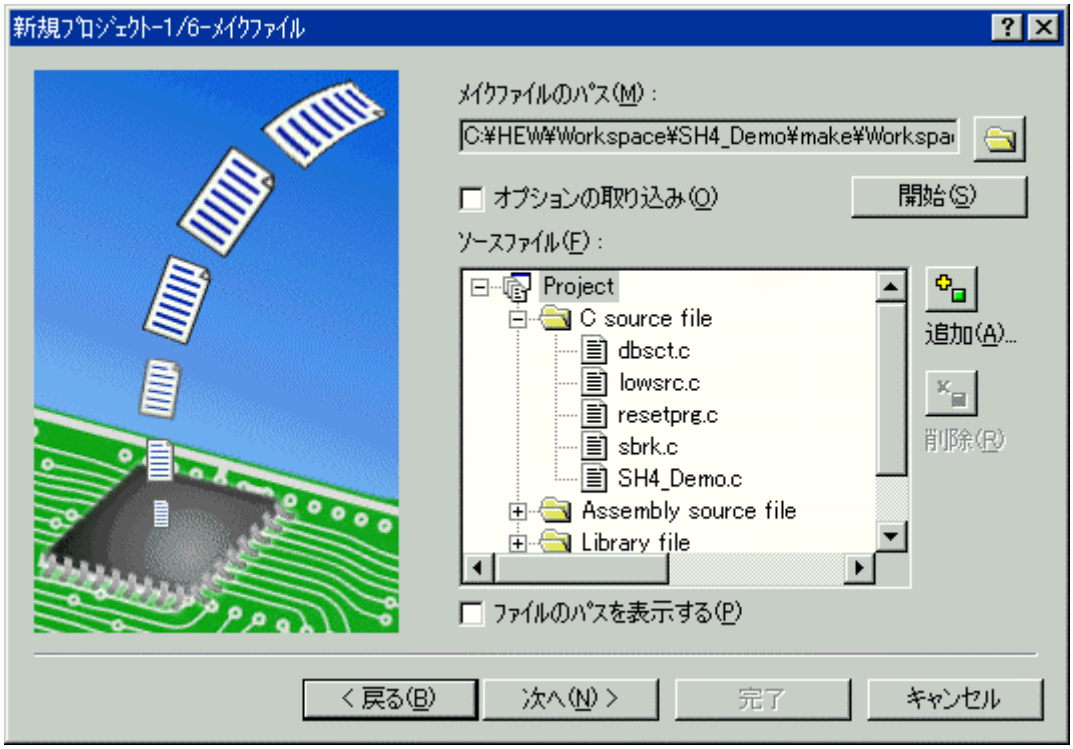
ステップ 9 画面で[完了]をクリックすると、[概要]ダイアログボックスを表示します。



10. プロジェクトジェネレータは、生成するプロジェクトに関する情報を[概要]ダイアログボックスに表示しますので、確認後[OK]ボタンをクリックしてください。[Cancel]ボタンをクリックすると[新規プロジェクト]ウィザードに戻ります。
なお、[サマリの内容をプロジェクトディレクトリに Readme.txt という名前で保存する]をチェックすると、[概要]ダイアログボックスで表示したプロジェクトの情報を”Readme.txt”という名称のテキストファイルでプロジェクトディレクトリに保存します。

(1) メイクファイルから情報を取り込みプロジェクトを作成する

HEW は、GNU make フォーマットまたは、Hmake フォーマット（HEW が出力）のメイクファイルを解析し、作成するワークスペースにファイル情報を取り込むことができます。[新規プロジェクトワークスペース]ダイアログボックスを開きプロジェクトタイプとして[Import Makefile]を選択します。ワークスペース名等の必要な情報を入力後、[OK]ボタンをクリックすると、[新規プロジェクト - メイクファイル]画面に進みます。



[メイクファイルのパス]でメイクファイルを選択すると[ソースファイル]にメイクファイルに登録されているソースファイルを表示します。メイクファイルに登録されているソースファイルを再び表示するには[開始]ボタンをクリックします。

コンパイラ等のツールオプションを取り込むには、[オプションの取り込み]チェックボックスをオンにします。

不要なファイルがある場合は、そのファイルを選択し[削除]ボタンで削除できます。また、さらに必要なファイルがある場合は、[追加]ボタンで追加できます。[ファイルのパスを表示する]チェックボックスをオンにするとファイルのフルパスを表示します。

17.1.3 プロジェクトの構成を編集する

SuperH または H8 ファミリのツールチェインをご使用の場合、プロジェクトジェネレータを使用してシミュレータを再度設定することが可能です。ただし、デモンストレーションプロジェクトでは無効です。

1. [プロジェクト->構成の編集]を選択すると[構成の編集]ダイアログボックスを表示します。
2. [ターゲット]タブをクリックしてください。
3. 使用したいターゲットを選択し、[OK]ボタンをクリックしてください。

17.1.4 デバッギングプラットフォームを構築する

デバッギングプラットフォームにプログラムをロードする前に、アプリケーションシステムに合うようにプラットフォームを設定します。必要な設定内容は、デバイスの種類、動作モード、クロックスピード、およびメモリマップなどです。その中でもプログラムをロードするためにデバッギングプラットフォームにメモリを設定する必要があるので、特にメモリマップの設定が大切です。HEW では、プロジェクトの生成処理でこの作業のほとんどが完了します。しかし、標準とは異なるボードのコンフィグレーションを使用する場合には、カスタマイズが必要になります。

(1) セットアップ

デバッギングプラットフォームをセットアップするには、[基本設定->シミュレータ]または[基本設定->エミュレータ]メニューオプションを選択します。このサブメニューの下に、デバッギングプラットフォームの構築に使用できるメニューがあります。

SuperH ファミリのシミュレータの場合、[システム]と[メモリリソース]のメニューがあります。どちらのオプションでも、必要に応じてシミュレータをカスタマイズし、セットアップできます。[デバッグ->デバッグの設定]ダイアログボックスから選択してデバッギングプラットフォーム依存のセットアップダイアログボックスを表示できます。

ご使用のデバッギングプラットフォームで利用可能な機能の詳細説明は、エミュレータまたはシミュレータのユーザーズマニュアルを参照してください。

(2) メモリマップ

デバッギングプラットフォームは、デバイスのアドレス空間が ROM、RAM、オンチップレジスタ、またはメモリのない領域なのかを知る必要があります。そのため、メモリマップを設定する必要があります。

プロジェクトジェネレータでデバイスの種類およびモードを選択すると、HEW は自動的にそのデバイスのマップおよびそのプロセッサが動作するモードを設定します。例えば、内部 ROM および RAM を持つデバイスにおいて、これらのメモリがデバイスメモリマップ内で位置する領域をデフォルトで設定します。

内部メモリを持たないデバイス、またはその代わりあるいはその追加として外部メモリを持つデバイスを使用する場合には、そういったメモリが存在することをデバッギングプラットフォームに知らせる必要があります。

補足説明:

エミュレータでデバッグを行い、かつ必要なメモリがオンチップまたは外部に存在しない場合は、アプリケーションのアドレス空間用にエミュレータのエミュレーションメモリをマッピングできます。

詳細は新規プロジェクトで選択したデバッギングプラットフォームに依存します。

メモリマップに関する追加情報は[ステイタス]ウィンドウの[Memory]タブにも表示します。”Target Device Configuration”領域にはデバイスのアドレス空間内のメモリを表示します。

注：

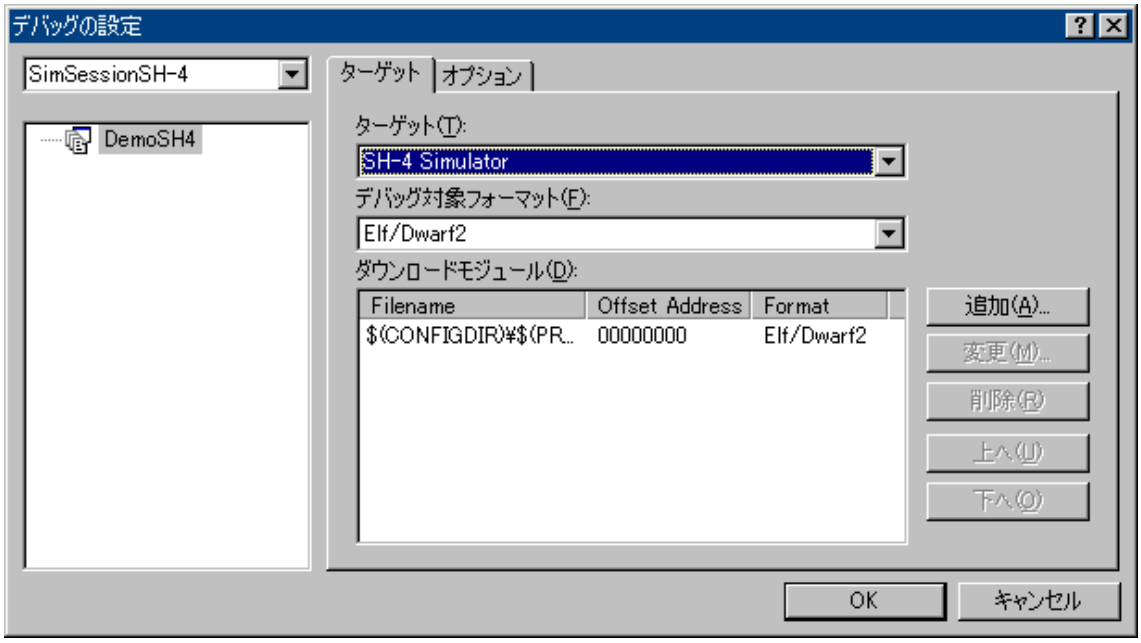
いくつかのエミュレータではページ長の制限から、入力アドレスと設定された領域アドレスが一致しない場合があります。

(3) HEW デバッガを設定する

HEW デバッガの通常動作では、ユーザのターゲットおよびダウンロードモジュールはプロジェクト生成過程で自動的に構築します。ただし時にはデバッグセッションを手動で構築する必要があります。古いツールチェインや、最新の HEW インタフェースをサポートしていないプロジェクトジェネレータを使用している場合などが考えられます。

デバッグセッションの設定を確認するには、［デバッグの設定］ダイアログボックスを表示します。次のいずれかを行ってください。

- [デバッグ->デバッグの設定]を選択する。
- ワークスペースウィンドウの[Projects]タブの[Download modules]フォルダまたはダウンロードモジュール上で右クリックする。表示されるポップアップメニューから[デバッグの設定]を選択する。



［デバッグの設定］ダイアログボックスの[ターゲット]タブでは、[ターゲット]、[デバッグ対象フォーマット]、および[ダウンロードモジュール]を選択できます。さらに[オプション]タブでは、コマンドラインバッチファイルへのアクセス、ターゲット接続時のアクセス、およびダウンロード時のアクセスのオプションを提供しています。

ターゲットを変更するには、次の操作が必要です。

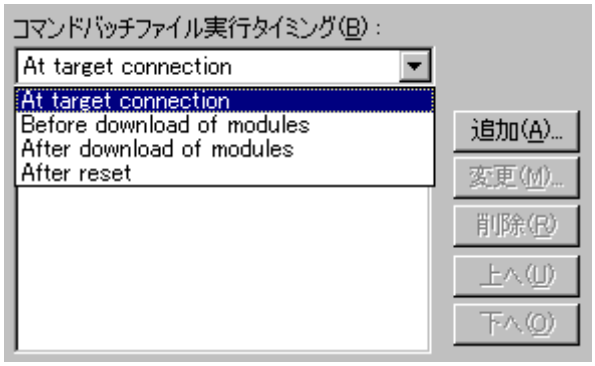
1. ダイアログボックス左側にあるツリーから、変更したいプロジェクトを選択してください。現在のプロジェクトのデフォルトとして設定します。
2. ツリーの上にあるドロップダウンリストから、変更したいセッションを選択してください。
3. ターゲットのリストを操作して、ターゲットを変更してください。これにより、過去に設定したターゲット固有の設定オプションを削除します。

(4) デバッグ設定のオプションを指定する

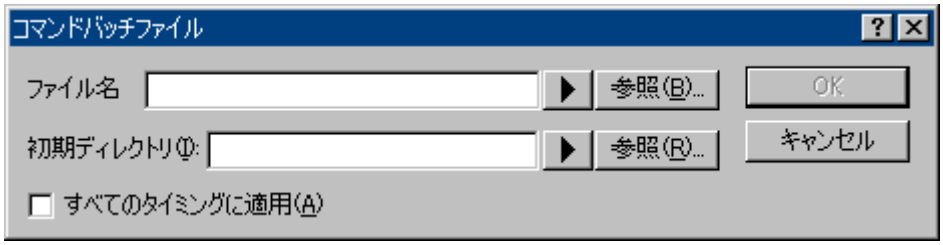
HEW デバッガは TCL コマンドライン機能と統合されています。特定のタイミングで自動的に実行できる、HEW デバッガ用のコマンドバッチファイルを記述できます。[コマンドバッチファイル実行順序]リストは、指定されたタイミングで実行するコマンドバッチファイルのリストです。このリストに対してコマンドバッチファイルを追加、変更、削除、上へ移動、下へ移動が可能です。

コマンドバッチファイルを自動実行するには

1. [デバッグ->デバッグの設定]を選択し、[デバッグの設定]ダイアログボックスを表示します。
2. [オプション]タブを選択します。
3. [コマンドバッチファイル実行タイミング]を選択してください。At target connection、Before download of modules、After download modules、After reset から選択できます。



4. [追加]ボタンをクリックしてください。[コマンドバッチファイル]ダイアログボックスを表示します。



5. [ファイル名]フィールドにコマンドバッチファイルのファイル名を入力します。[ファイル名]フィールドにプレースホルダを挿入するには、プレースホルダボタン(▶)をクリックし、ポップアップメニューからプレースホルダを選択します。ファイルを参照するには[参照]ボタンをクリックします。
6. [初期ディレクトリ]フィールドにコマンドバッチファイルを実行するディレクトリを入力します。[初期ディレクトリ]フィールドの設定により相対パスがあるコマンドバッチファイルの使用が可能になります。[初期ディレクトリ]フィールドにプレースホルダを挿入するには、プレースホルダボタン(▶)をクリックし、ポップアップメニューからプレースホルダを選択します。ディレクトリを参照するには[参照]ボタンをクリックします。
7. [すべてのタイミングに適用]チェックボックスをオンにすると、すべてのタイミング（At target connection、Before download of modules、After download modules、After reset）にコマンドバッチファイルを適用します。

8. [OK]ボタンをクリックすると、コマンドバッチファイルを追加します。
9. 追加後は、[上へ]ボタンと[下へ]ボタンを使って順番を移動できます。複数のコマンドバッチファイルを追加している場合のみ有効です。

その他のオプション

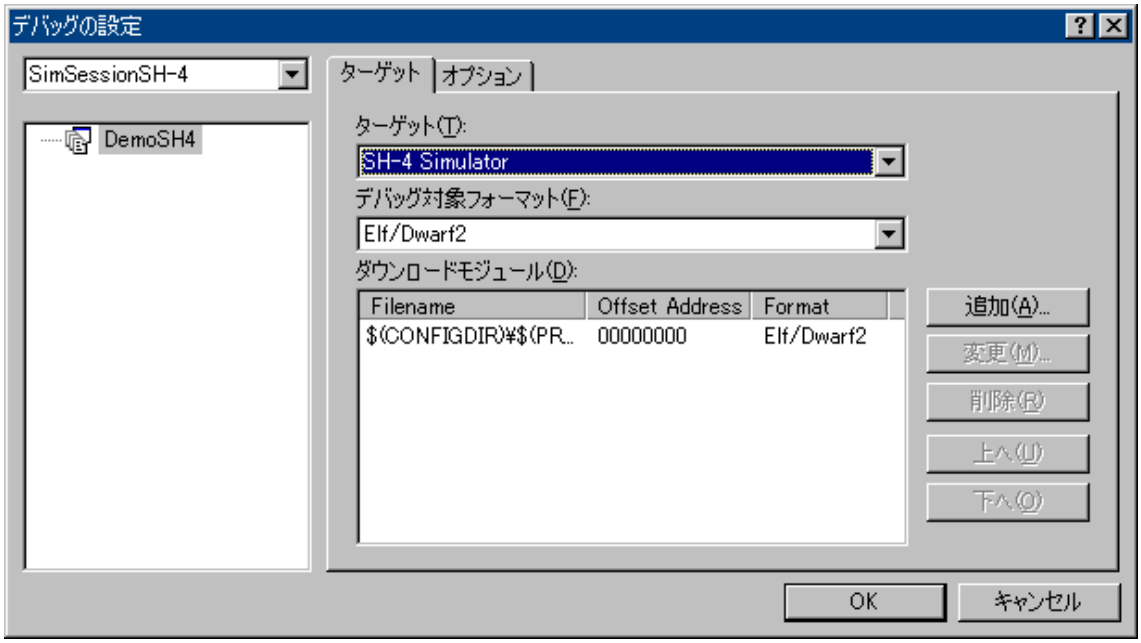
オプション	説明
[自動的にターゲットを接続しない] ***1**	このチェックボックスをオンにすると、[デバッグ->接続]を選択するまでターゲットに接続しません。 このチェックボックスのデフォルトはオフです。
[ビルド後のダウンロード]	このチェックボックスをオンにすると、モジュールはビルド後自動的にダウンロードします。このチェックボックスのデフォルトはオンです。
[ダウンロード後に CPU リセット] ***1**	このチェックボックスをオンにすると、モジュールのダウンロード完了後に自動的に CPU がリセットされます。 このチェックボックスは、選択されたターゲットでサポートされていると切り替えが可能になります。新しくターゲットを選択すると、このオプションはそのターゲットのデフォルトに設定されます。このチェックボックスのデフォルトはデバッギングプラットフォームに依存します。
[ダウンロード時にブレークポイントを削除]	このチェックボックスをオンにすると、モジュールのダウンロード時自動的にソフトウェアブレークポイントをすべて解除します。このチェックボックスのデフォルトはデバッギングプラットフォームに依存します。
[ターゲット接続時のバッチファイル実行が終わるまでメモリアクセスを無視]	このチェックボックスをオンにすると、デバッギングプラットフォーム接続時に自動実行するコマンドバッチファイルを指定している場合、そのコマンドバッチファイルの実行が終了するまで、ターゲットへのメモリアクセスを行いません。ターゲット接続時にメモリの初期設定をする必要がある場合に有用です。このチェックボックスのデフォルトはオフです。
[逆アセンブリ表示時のメモリアクセス範囲を最小限に抑える]	このチェックボックスをオンにすると、逆アセンブリ表示で表示範囲以外のメモリをリードしません。このチェックボックスのデフォルトはデバッギングプラットフォームに依存します。

***1.** 機能のサポートはデバッギングプラットフォームに依存します。

17.1.5 ダウンロードモジュール

(1) ダウンロードモジュールを指定する

プログラムをダウンロードするためのメモリがシステムに十分あることを確認したら､デバッグするプログラムのダウンロードを行うことができます。アプリケーションジェネレータに関する初期のダウンロードモジュールの選択は、リンカからの出力と同じように自動的に行います。デバッグ専用プロジェクトジェネレータについては、ダウンロードしたいモジュールを参照することができます。概要については、「17.1.2 デバッギングプラットフォームを選択する」を参照してください。



また、プロジェクトを作成したあとにダウンロードモジュールを手動で選択できます。これは、[デバッグの設定]ダイアログボックスを使用して行うことができます。このダイアログボックスでは、ワークスペース全体に渡りデバッグの設定を制御できます。ダイアログボックスの左にあるツリーは、現在のプロジェクトをすべて含みます。このツリーでプロジェクトを選択すると、そのプロジェクトの設定の表示、およびセッションドロップダウンリストには、セッションの選択を表示します。このリストボックスでは、複数のセッションを選択、およびすべてのセッションを選択もできます。複数のセッションを選択した場合には、1 つまたは複数のセッションの設定を同時に修正できます。[デバッグの設定]ダイアログボックスは、以下のデバッグオプションを表示します。

- 現在のプロジェクトのための現在のデバッグターゲットおよびセッションの選択
- 現在のプロジェクトのためのダウンロードモジュールおよびセッションの選択

ダウンロードモジュールのリストは、ターゲットにダウンロードするファイルの順番で表示されます。このリストに対してモジュールを追加、変更、削除、上へ移動、下へ移動が可能です。

新規のダウンロードモジュールを追加するには

1. 次のいずれかを行ってください。[デバッグの設定]ダイアログボックスを表示します。

- [デバッグ->デバッグの設定]を選択する。
- ワークスペースウィンドウの[Projects]タブの[Download modules]フォルダまたはダウンロードモジュール上で右クリックする。表示されるポップアップメニューから[デバッグの設定]を選択する。

2. プロジェクトのツリーで、ダウンロードモジュールを追加するプロジェクトおよびセッションを選択します。
3. [追加]ボタンをクリックしてください。[ダウンロードモジュール]ダイアログボックスを表示します。

4. すべてのフィールドは、ダウンロードモジュールが正確に構成されるようにセットアップします。最初のフィールドは[オフセット]です。これは、モジュールがロードされるメモリアドレスのオフセットです。デフォルトは 0 です。モジュールの追加後、[デバッグ->ビルド]のサブメニュー、ワークスペースウィンドウの[Projects]タブの[Download modules]フォルダ内、および[ファイル->最近ダウンロードしたモジュール]のサブメニューで、モジュール名の右横にオフセット値が表示されます。
5. [フォーマット]は、サポートしているオブジェクトフォーマットのリストを含みます。[デバッグの設定]ダイアログボックスの[デバッグ対象フォーマット]と一致する必要はありません。ただし、[デバッグ対象フォーマット]フィールドで指定したフォーマットと一致するモジュールのみデバッグできます。
6. [ファイル名]は、プレースホルダ、または絶対パス形式として指定できます。プレースホルダを使用することをお勧めします。
7. [アクセスサイズ]フィールドは、メモリアクセス時のアクセス幅を指定します。特定のアクセス幅でメモリにアクセスする必要がある場合、このオプションを使用してアクセス幅を指定してください。
8. [デバッグ情報のみのダウンロード]チェックボックスは、デバッグ情報だけをダウンロードする場合に使用します。
9. [ダウンロード時のメモリベリファイ]チェックボックスは、モジュールをダウンロードしたとき、ターゲットデバイスへ正確にダウンロードされたことを検証するために使用します。
10. [ターゲット接続時にダウンロード]チェックボックスは、ターゲットに接続したときに自動的にダウンロードを行う場合に使用します。
11. [OK]ボタンをクリックすると、[ダウンロードモジュール]リストの最後にモジュールが追加されます。

ダウンロードモジュールの追加と同時にターゲットにダウンロードするには、「17.1.5 (2) モジュールをダウンロードする」を参照してください。

ダウンロードモジュールの設定を変更するには

1. 次のいずれかを行ってください。

- [デバッグの設定]ダイアログボックスを表示し、[ダウンロードモジュール]のリストから設定を変更したいモジュールを選択し[変更]ボタンをクリックする。
- ワークスペースウィンドウの[Projects]タブの[Download modules]フォルダ内のダウンロードモジュール上で右クリックする。表示されるポップアップメニューから[プロパティ]を選択する。

2. [ダウンロードモジュール]ダイアログボックスを表示します。
3. 設定を変更後、[OK]ボタンをクリックします。

ダウンロードモジュールを削除するには

1. [デバッグの設定]ダイアログボックスを表示します。
2. [ダウンロードモジュール]リストから削除したいモジュールを選択し、[削除]ボタンをクリックします。

ワークスペースウィンドウの**[Projects]**タブで選択したダウンロードモジュールを削除するには

1. ワークスペースウィンドウの[Projects]タブで削除したいダウンロードモジュールを選択してください。複数のダウンロードモジュールを選択するときは”Shift”キーまたは”Ctrl”キーを押してください。
2. 次のいずれかを行ってください。

- 右クリックしポップアップメニューから[削除]を選択する。
- “Delete”キーを押す。

3. ダウンロードモジュールをプロジェクトから削除するかを確認するダイアログボックスが表示されます。ダウンロードモジュールをプロジェクトから削除する場合は「はい」を選択してください。削除しない場合は「いいえ」または「キャンセル」を選択してください。

確認要求のダイアログボックスを表示しない場合、[今後表示しない]チェックボックスをオンにしてください。確認要求のダイアログボックスを再び表示するには、[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を表示します。[確認]タブの[プロジェクトからダウンロードモジュールを削除]チェックボックスをオンにしてください。このチェックボックスのデフォルトはオンです。

モジュールをターゲットにダウンロードする順番を変更するには

1. [デバッグの設定]ダイアログボックスを表示します。
2. [ダウンロードモジュール]リストで、ダウンロードする順番を変更したいモジュールを選択し、[上へ]または[下へ]ボタンをクリックします。

[デバッグの設定]ダイアログボックスで行った変更はすべて、[OK]ボタンをクリックするまで更新しません。

[デバッグ対象フォーマット]は、リスト中の最初のダウンロードモジュールに設定しています。1 セッションにつき、デフォルトのデバッグ対象フォーマットを 1 つだけ指定できます。現在インストールしているオブジェクトフォーマットはすべて表示します。

(2) モジュールをダウンロードする

デバッグするためにオブジェクトプログラムをダウンロードします。

モジュールをダウンロードするには

次のいずれかを行ってください。

- [デバッグ->ダウンロード]のサブメニューからダウンロードしたいモジュールを選択する。
- ワークスペースウィンドウの[Projects]タブの[Download modules]フォルダ内でダウンロードしたいモジュールをダブルクリックする。
- ワークスペースウィンドウの[Projects]タブの[Download modules]フォルダ内のダウンロードしたいモジュール上で右クリックする。表示されるポップアップメニューから[ダウンロード]または[ダウンロード(debug 情報のみ)]を選択する。この場合モジュールは複数選択できます。

HEW では、最近ダウンロードした 4 モジュールを[ファイル->最近ダウンロードしたモジュール]のサブメニューに表示します。最近使ったモジュールを再びダウンロードしたいときに使用してください。

すべてのモジュールをダウンロードするには

次のいずれかを行ってください。

- [デバッグ->ダウンロード->All Download Modules]を選択してください。
- ワークスペースウィンドウの[Projects]タブの[Download modules]フォルダ上で右クリックする。表示されるポップアップメニューから[全てダウンロード]を選択する。

[デバッグ->デバッグの設定]を選択し、[デバッグの設定]ダイアログボックスを表示してください。[ターゲット]タブの[ダウンロードモジュール]で指定した順番でターゲットにダウンロードされます。

モジュールは存在してもダウンロードモジュールリストに追加されていない場合、ダウンロードモジュールリストへの追加と同時にターゲットにダウンロードすることができます。

ダウンロードモジュールリストへの追加と同時にターゲットにダウンロードするには

1. 次のいずれかを行ってください。

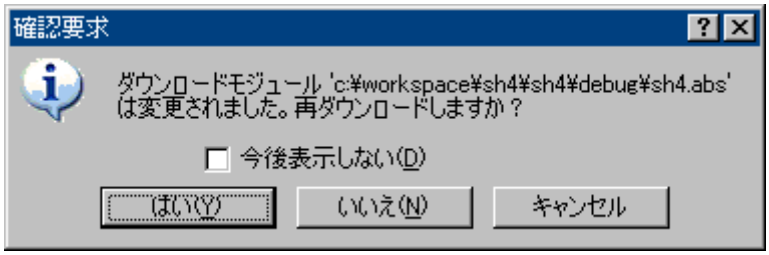
- [ファイル->ダウンロードモジュールの追加]を選択する。
- ワークスペースウィンドウの[Projects]タブの[Download modules]フォルダフォルダまたはダウンロードモジュール上で右クリックする。表示されるポップアップメニューから[ダウンロードモジュールの追加]を選択する。

2. [ダウンロードモジュール]ダイアログボックスを表示します。

3. すべてのフィールドは、ダウンロードモジュールが正確に構成されるようにセットアップします。最初のフィールドは[オフセット]です。これは、モジュールがロードされるメモリアドレスのオフセットです。デフォルトは 0 です。モジュールの追加後、[デバッグ->ビルド]のサブメニュー、ワークスペースウィンドウの[Projects]タブの[Download modules]フォルダ内、および[ファイル->最近ダウンロードしたモジュール]のサブメニューで、モジュール名の右横にオフセット値が表示されます。
4. [フォーマット]は、サポートしているオブジェクトフォーマットのリストを含みます。[デバッグの設定]ダイアログボックスの[デバッグ対象フォーマット]と一致する必要はありません。ただし、[デバッグ対象フォーマット]フィールドで指定したフォーマットと一致するモジュールのみデバッグできます。
5. [ファイル名]は、プレースホルダ、または絶対パス形式として指定できます。プレースホルダを使用することをお勧めします。
6. [アクセスサイズ]フィールドは、メモリアクセス時のアクセス幅を指定します。特定のアクセス幅でメモリにアクセスする必要がある場合、このオプションを使用してアクセス幅を指定してください。
7. [デバッグ情報のみのダウンロード]チェックボックスは、デバッグ情報のみをダウンロードする場合に使用します。
8. [ダウンロード時のメモリベリファイ]チェックボックスは、モジュールをダウンロードしたとき、ターゲットデバイスへ正確にダウンロードされたことを検証するために使用します。
9. [ターゲット接続時にダウンロード]チェックボックスは、ターゲットに接続したときに自動的にダウンロードを行う場合に使用します。
10. [OK]ボタンをクリックします。

[デバッグ->デバッグの設定]を選択し、[デバッグの設定]ダイアログボックスを表示してください。[ターゲット]タブの[ダウンロードモジュール]リストの最後にモジュールが追加されています。

また、既にダウンロードしたモジュールが HEW の外部で変更された場合、モジュールを再びダウンロードするかを確認するダイアログボックスが表示されます。モジュールを再びダウンロードする場合は「はい」を選択してください。ダウンロードしない場合は「いいえ」または「キャンセル」を選択してください。



確認要求ダイアログボックスを表示しない場合、[今後表示しない]チェックボックスをオンにしてください。

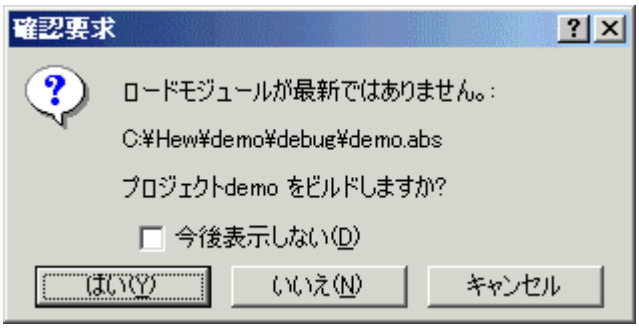
確認要求ダイアログボックスを再び表示するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [確認]タブを選択してください。
3. [古いダウンロードモジュールの再ダウンロード]チェックボックスをオンにしてください。このチェックボックスのデフォルトはオンです。
4. OK をクリックしてください。

(3) ダウンロード前に変更したソースファイルを確認する

HEW は、現在のプロジェクトのダウンロードモジュールをダウンロードする前にソールファイルが変更されているかどうかを確認します。

ファイルが変更されていた場合、確認のダイアログボックスが起動され、ダウンロード実行前にコードを再度ビルドするかどうかを確認するダイアログボックスが表示されます。ダウンロード実行前にコードを再度ビルドする場合は「はい」を選択してください。ダウンロード実行前にコードを再度ビルドしない場合は「いいえ」を選択してください。



確認要求ダイアログボックスを表示しない場合、[今後表示しない]チェックボックスをオンにしてください。

確認要求ダイアログボックスを再び表示するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [確認]タブを選択してください。

3. [ダウンロードモジュールの更新]チェックボックスをオンにしてください。このチェックボックスのデフォルトはオンです。
4. OK をクリックしてください。

(4) ダウンロード後にソースツリーを表示する（デバッグ専用プロジェクト）

デバッグ専用プロジェクト（[Debugger only - xxxxxx]）を使用している場合、ダウンロードするとワークスペースウィンドウの［Projects］タブ内にダウンロードモジュールを構成するソースファイルをツリーで表示します。

HEW はダウンロード時、ダウンロードモジュールを構成するソースファイルの情報を取得します。

相対パス名、またはファイル名の場合は、ダウンロードモジュールのディレクトリを基準にファイルを検索します。

この初期検索で見つからなかったファイルは[ファイルの検索]ダイアログボックス内に表示します。

ダウンロード時、[ファイルの検索]ダイアログボックスを表示しないようにできます。

ダウンロード時、[ファイルの検索]ダイアログボックスを表示しないようにするには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [確認]タブを選択してください。
3. [ディレクトリ参照ダイアログをダウンロード時に表示しない]チェックボックスをオンにしてください。このチェックボックスのデフォルトはオフです。
4. [OK]をクリックしてください。

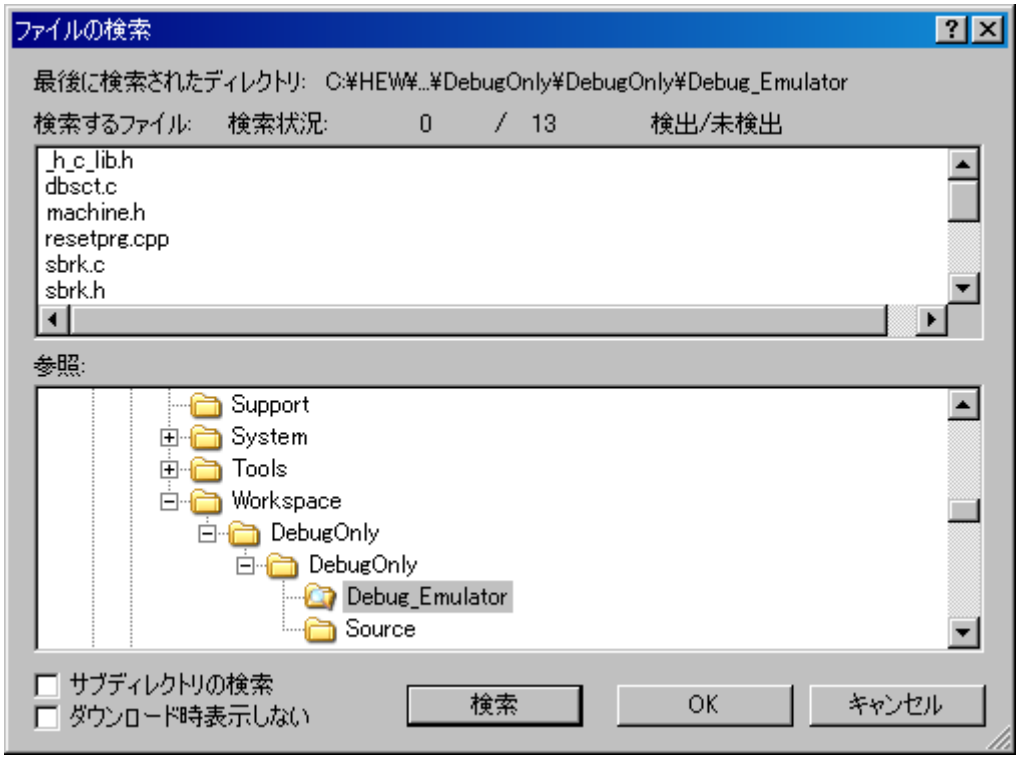
- **対応デバッガ**

デバッグ専用プロジェクトを作成できるデバッガがダウンロード後にソースツリーを表示する機能に対応しています。デバッガにより対応する条件が異なります。

対応デバッガ	対応条件
SuperH RISC engine および H8SX, H8S, H8 ファミリのエミュレータデバッガ	HEW のアップデートによって、HEW のバージョンが V.4.02 以上になった場合に対応しています。
M16C、M32C、M32R ファミリのエミュレータデバッガ	製品パッケージに同梱される HEW のバージョンが V.4.02 以上の場合に対応しています。

(a) ファイルを検索する

初期検索で見つからなかったソースファイルは、[ファイルの検索]ダイアログボックス内（[検索するファイル]リスト）に表示します。

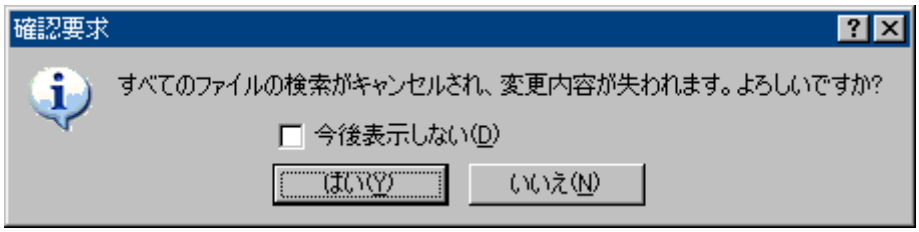


ファイルを検索するには

1. [最後に検索されたディレクトリ]は、ソースファイルを最後に検索したディレクトリを保持します。
2. [検索状況]フィールドは、見つからなかったファイル数（[未検出]）に対して見つかったファイル数（[検出]）を表示します。ダイアログボックスに表示されているファイルの数で、前回検索した時の数は含みません。
3. [検索するファイル]リストは、見つからなかったすべてのファイルのリストを保持します。リストにはファイル名のみを表示し、このファイル名を元に検索します。
4. [参照]フィールドは、ファイルを検索するためのディレクトリを選択します。
5. [サブディレクトリの検索]チェックボックスをオンにすると、選択されたディレクトリのすべてのサブディレクトリでファイルを検索します。このチェックボックスのデフォルトはオフです。
6. [ダウンロード時表示しない]チェックボックスをオンにすると、ダウンロード時に見つからなかったファイルがある場合もこのダイアログボックスを開きません。このチェックボックスのデフォルトはオフです。このチェックボックスは、[オプション]ダイアログボックスの[確認]タブにもあります。
7. [検索]ボタンをクリックすると、[参照]フィールドで選択したディレクトリにあるファイルを検索し、見つかったファイルは[検索するファイル]リストから消去されます。
   [参照]フィールドで検索済みのディレクトリは、フォルダと虫眼鏡のアイコン（ ）で表示されます。
   [検索]ボタンをクリックすると、検索を取り消せる[中止]ボタンに変わります。[中止]ボタンをクリックすると、検索を取り消す前に見つかったファイルは、場所が特定されない状態に戻ります。

8. [OK]をクリックすると、検索して見つかったファイルをそのディレクトリに特定します。すべてのファイルの検索が保存され、ダイアログボックスを閉じます。
9. [キャンセル]をクリックすると、すべてのファイルの検索が取り消され、ダイアログボックスを閉じます。

ファイルを検索した後で[キャンセル]を選択すると、下記の確認要求ダイアログボックスが表示されます。



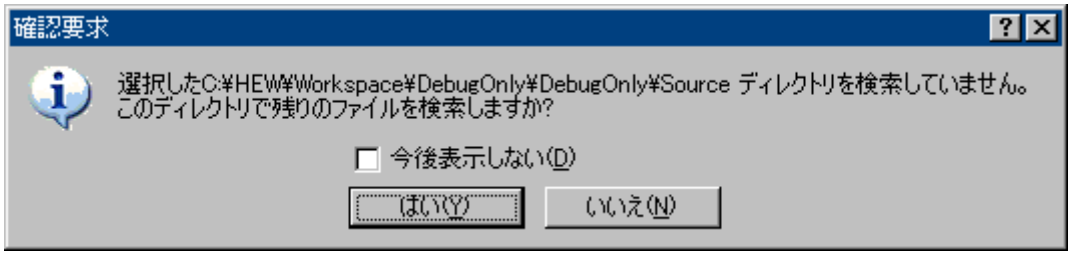
「はい」を選択すると、[ファイルの検索]ダイアログボックスを閉じます。「いいえ」を選択すると、[ファイルの検索]ダイアログボックスは閉じません。

確認要求ダイアログボックスを表示しない場合、[今後表示しない]チェックボックスをオンにしてください。[今後表示しない]を選択すると、「はい」を選択したときの動作がデフォルトです。

確認要求ダイアログボックスを再び表示するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [確認]タブを選択してください。
3. [ディレクトリ参照ダイアログでキャンセルした時の確認]チェックボックスをオンにしてください。このチェックボックスのデフォルトはオンです。
4. [OK]をクリックしてください。

ディレクトリを選択した後で[検索]を選択せずに[OK]を選択すると、下記の確認要求ダイアログボックスが表示されます。



「はい」を選択すると、ファイルを検索し、[ファイルの検索]ダイアログボックスは閉じません。必要に応じて検索は中止できます。

「いいえ」を選択すると、メッセージを表示し、[ファイルの検索]ダイアログボックスを閉じます。

確認要求ダイアログボックスを表示しない場合、[今後表示しない]チェックボックスをオンにしてください。[今後表示しない]を選択すると、「いいえ」を選択したときの動作がデフォルトです。

確認要求ダイアログボックスを再び表示するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [確認]タブを選択してください。
3. [ディレクトリ参照ダイアログで OK した時の検索の確認]チェックボックスをオンにしてください。このチェックボックスのデフォルトはオンです。
4. [OK]をクリックしてください。

(b) ソースツリーを表示する

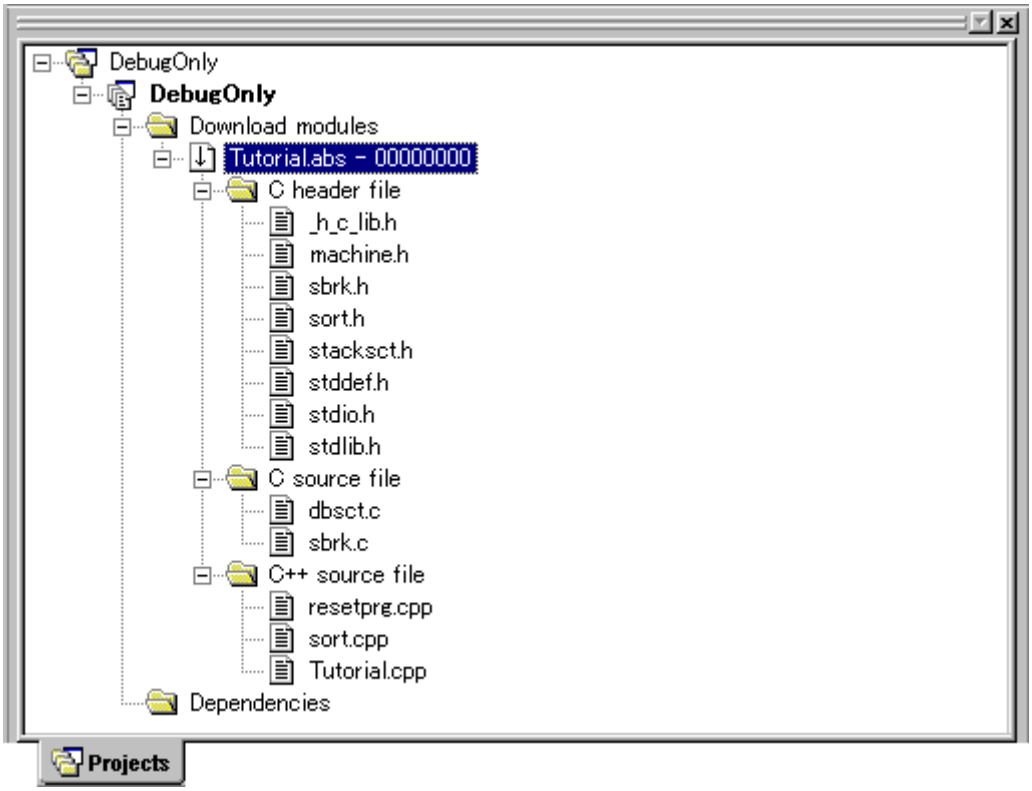
ダウンロードモジュールから取得したソースファイル名は、ワークスペースウィンドウのダウンロードモジュール下に表示します。

ソースファイルは、ファイルの種類でグループ分けし関連したフォルダ下に表示します。

ファイルの種類が定義されていない場合、ファイルの拡張子フォルダ下にグループ分けします。

この場合、ファイルをダブルクリックすると外部エディタでファイルを開きます。エディタウィンドウで開くには、「2.5 ファイル拡張子とファイルグループ」を参照してください。

見つかったソースファイルは、ダウンロードモジュール下に通常のファイルアイコン（）で表示します。

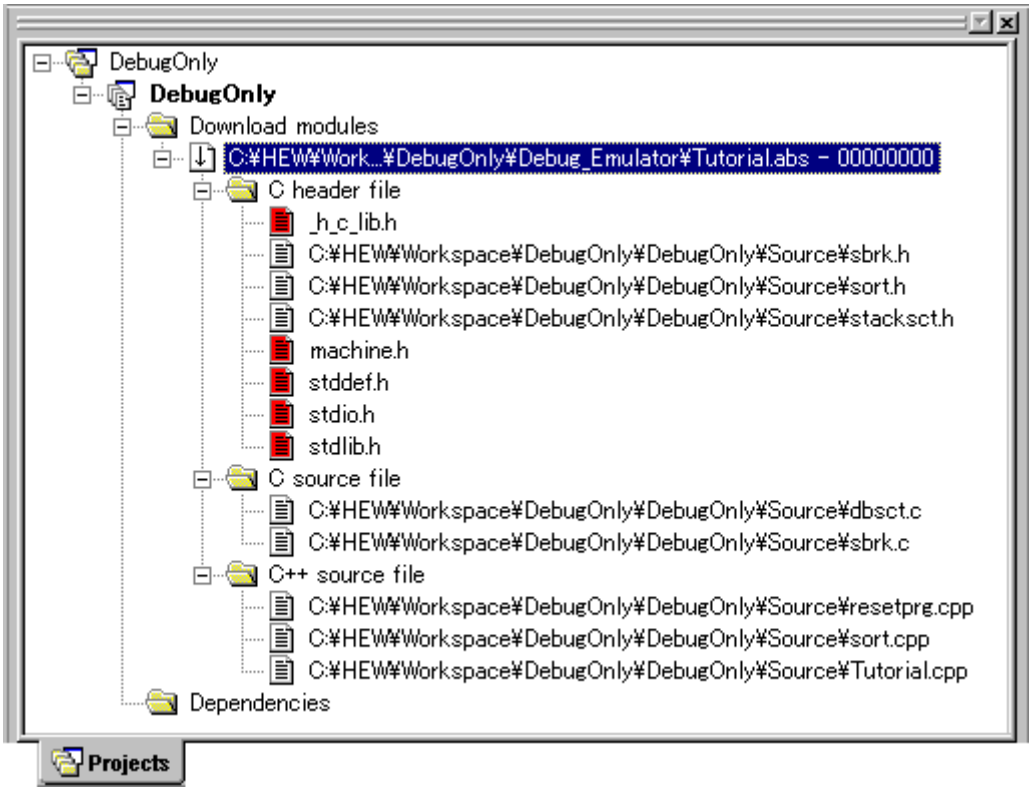


見つからなかったソースファイルは、ダウンロードモジュール下に赤いファイルアイコン（）で表示します。

これらのファイルを区別するためにファイルをフルパスで表示することができます。

すべてのファイルをフルパスで表示するには

1. ワークスペースウィンドウの[Projects]タブ内で右クリックします。
2. [表示の構成]メニュー項目を選択してください。[表示の構成]ダイアログボックスを表示します。
3. [ファイルパスの表示]チェックボックスをオンにしてください。
4. [OK]をクリックしてください。



ポップアップメニューからファイルを別のディレクトリで再検索することができます。

(c)　ファイルを別のディレクトリで再検索する

ダウンロードモジュール内のすべてのソースファイルを別のディレクトリで再検索するには、ダウンロードモジュールのポップアップメニューから行ってください。

ダウンロードモジュール内のすべてのソースファイルを別のディレクトリで再検索するには

1. ワークスペースウィンドウの[Projects]タブ内のダウンロードモジュール上で右クリックします。
2. [モジュールの検索]メニュー項目を選択します。Windows(R)の標準的なディレクトリを開くダイアログボックスを表示します。

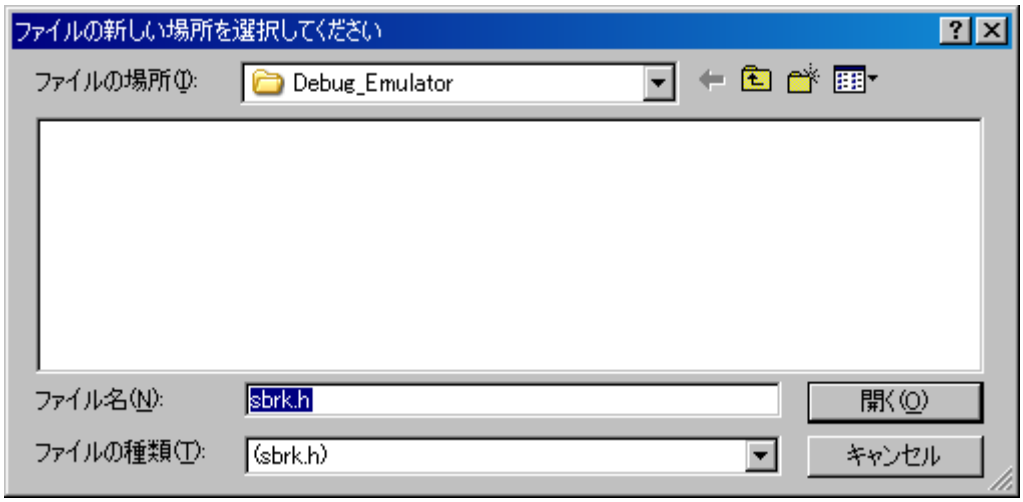
3. 検索するディレクトリを選択し、[開く]ボタンをクリックします。

選択したディレクトリで見つからなかったソースファイルは、[ファイルの検索]ダイアログボックスを表示し、これらのファイルを検索できます。

また、ソースファイルを別のディレクトリで再検索するには、ファイルのポップアップメニューから行ってください。

ファイルを別のディレクトリで再検索するには

1. ワークスペースウィンドウの[Projects]タブ内のファイル上で右クリックします。
2. [ファイルの検索]メニュー項目を選択します。Windows(R)の標準的なファイルを開くダイアログボックスを表示します。



3. 同じ名前のファイルだけを選択することができます。
4. ファイルを選択し、[開く]ボタンをクリックします。

複数のファイルを選択することもできます。その場合は、[ファイルの検索]ダイアログボックスを表示します。

(5) ダウンロード後に main 関数を表示する

モジュールのダウンロード後に main 関数を含むソースファイルを自動的に HEW エディタのソースモードで表示する機能があります。

ダウンロード後に main 関数を表示するには

1. [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を表示します。
2. [デバッグ]タブを選択します。
3. [ダウンロード後に main 関数を表示]チェックボックスをオンにします。デフォルトはオフです。
4. OK をクリックします。
5. モジュールをダウンロードします。ダウンロードについては、「17.1.5 (2) モジュールをダウンロードする」を参照してください。

(6) モジュールをアンロードする

ダウンロードしたモジュールを手動でアンロードします。

モジュールをアンロードすると、そのモジュールのシンボルは、HEW デバッギングシステムからなくなりますが、ターゲットのメモリ内容は変更しません。一度アンロードしたモジュールは、再度ロードするまでデバッグすることはできません。

モジュールをアンロードするには

次のいずれかを行ってください。

- [デバッグ->アンロード]のサブメニューからアンロードしたいモジュールを選択する。
- ワークスペースウィンドウの[Projects]タブの[Download modules]フォルダ内のアンロードしたいモジュールを選択し、右クリックしポップアップメニューを表示してください。ポップアップメニューから[アンロード]を選択する。この場合モジュールは複数選択できます。

すべてのモジュールをアンロードするには

- [デバッグ->アンロード->All Download Modules]を選択する。

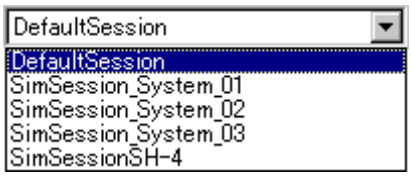
[デバッグ->デバッグの設定]を選択し、[デバッグの設定]ダイアログボックスを表示してください。[ターゲット]タブの[ダウンロードモジュール]リストで指定した順番でターゲットからアンロードされます。

17.1.6 デバッグセッション

HEW は、ビルダオプションをコンフィグレーションへ保存できます。同様に、HEW は、デバッグオプションをセッションに保存することもできます。

セッションには、デバッギングプラットフォーム、ダウンロードするプログラム、各デバッギングプラットフォームのオプションを保存できます。これは、各セッションが異なるプラットフォームをターゲットにできることを意味します。

この機能は異なるデバッグオプションを持つ多くの異なるセッションを許します。例えば、各セッションは同じターゲットで少し違うオプション設定にできます。これはレジスタ値やクロックスピードなどのターゲット設定を、セッションで簡単に変更したり切り替えたりできることを意味します。下図はこの典型例です。5 つのセッションがターゲットを共有しています。しかしセッションでは異なるオプション定義が可能です。



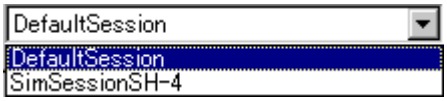
セッションは、コンフィグレーションとは直接関連がありません。これは、複数のセッションが同じダウンロードモジュールを共有し、プログラムの不要なリビルドを避けられることを意味します。

各セッションのデータは、別々のファイルで HEW プロジェクトに保存します。プロジェクト内でデータを共有したり、必要に応じて変更したりできます。

(1) セッションを選択する

セッションを選択するには

1. [デバッグ->デバッグセッション]を選択してください。[デバッグセッション]ダイアログボックスを表示します。
2. [現在のセッション]ドロップダウンリストから使用するセッションを選択してください。
3. [OK]ボタンをクリックしてください。



また、[標準]ツールバーのドロップダウンリストから異なるセッションを選択することもできます。

(2) セッションを追加する

ターゲットを取り付けてセットアップし、セッションを作成できます。セッションには名前を付け、ターゲットを選択することができます。

ターゲットを取り付けてセットアップし、新しいセッションを作成するには

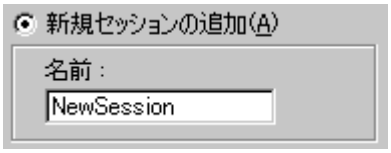
1. [ファイル->新規セッション]を選択してください。[新規セッション]ダイアログボックスを表示します。

2. 新しいセッションの名前を入力してください。セッション名は、32 文字以内で、半角英数字、半角下線のみ使用してください。特に、日本語文字、空白、または半角マイナス記号は使用しないでください。
3. 新しいセッションで使用するターゲットを選択してください。
4. 使用するセッションジェネレータを選択してください。デフォルトで選択されているものが適切ですが、時に複数のジェネレータが同じターゲットをサポートしている場合もあります。
5. [OK]ボタンをクリックしてください。生成プロセスを開始します。このプロセスは、選択した[使用するプロジェクトジェネレータ]に依存します。このとき、さらにターゲット設定オプションのダイアログを表示することがあります。
6. 設定を完了すると、現在のプロジェクトに新しいセッションを追加します。追加されたセッションは、メインツールバーの[セッション]ドロップダウンリストボックスの中にあります。

新しいセッションを追加できます。入力したセッション名のファイルを新しく作成します。ファイルが既に存在する場合はエラーを表示します。

新しい空のセッションを追加するには

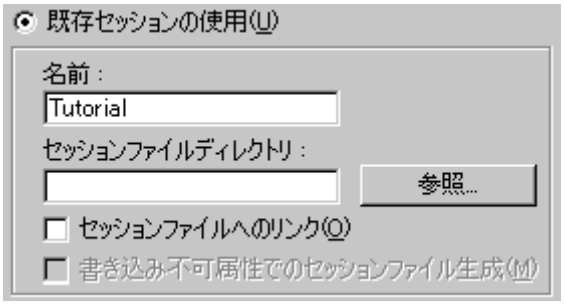
1. [デバッグ->デバッグセッション]を選択してください。[デバッグセッション]ダイアログボックスを表示します。
2. [追加]ボタンをクリックしてください。[新規セッション追加]ダイアログボックスを表示します。
3. [新規セッションの追加]ラジオボタンをチェックしてください。
4. セッションの名前を入力してください。
5. [OK]ボタンをクリックしてください。



別のファイルからセッションデータをインポートして新しいセッションを作成できます。インポート元ファイルのデータはそのまますべてコピーします。

既存のセッションを新しいセッションファイルにインポートするには

1. [デバッグ->デバッグセッション]を選択してください。[デバッグセッション]ダイアログボックスを表示します。
2. [追加]ボタンをクリックしてください。[新規セッション追加]ダイアログボックスを表示します。
3. [既存セッションの使用]ラジオボタンをチェックしてください。
4. セッションの名前を入力してください。
5. 現在のプロジェクトにインポートしたい既存のセッションファイルをブラウズしてください。
6. [OK]ボタンをクリックしてください。



これと同じ操作が、[ファイル->セッションのインポート]を使ってもできます。

1. [ファイル->セッションのインポート]を選択してください。[セッション名]ダイアログボックスを表示します。

2. 新しいセッションの名前を入力してください。
3. 新しいセッションにインポートしたいセッションファイルを選択してください。
4. [OK]ボタンをクリックしてください。ブラウズしたファイルと同じ設定で、新しい名前の付いた新しいセッションを追加します。

(3) リンクをセッションにインポートする

新しいセッションを HEW システムに追加するとき、セッションファイルをプロジェクトディレクトリにインポートまたはコピーしたりするのではなく、その場でセッションファイルにリンクできます。この方法は、ネットワーク環境で他のユーザとデバッガ情報を共有する場合に便利です。

既存のセッションファイルにリンクをインポートするには

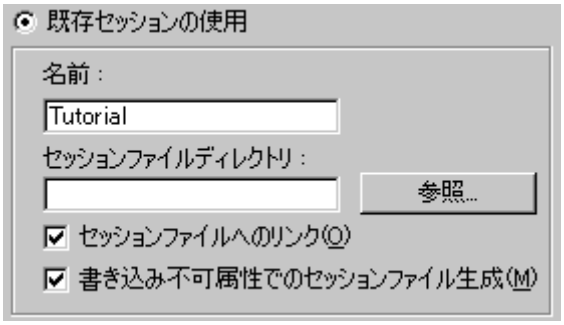
1. [デバッグ->デバッグセッション]を選択してください。[デバッグセッション]ダイアログボックスを表示します。
2. [追加]ボタンをクリックしてください。[新規セッション追加]ダイアログボックスを表示します。

3. [既存セッションの使用]ラジオボタンをチェックしてください。
4. セッションの名前を入力してください。
5. 現在のプロジェクトにインポートしたい既存のセッションファイルを参照してください。
6. [セッションファイルへのリンク]チェックボックスをオンにしてください。プロジェクトディレクトリに新しいセッションファイルを生成せず、HEW は既存のセッションファイルにリンクします。このファイルの場所はステップ 5 で入力したもので、すべてのセッションデータをこの場所に保存します。
7. [OK]ボタンをクリックしてください。

リンクしたセッションファイルをリードオンリーで使用できます。デバッガ設定ファイルを共有していて、データを誤って変更してしまうのを防ぐ場合に便利です。

リンクを既存のファイルにインポートし、リードオンリーにするには

1. [デバッグ->デバッグセッション]を選択してください。[デバッグセッション]ダイアログボックスを表示します。
2. [追加]ボタンをクリックしてください。[新規セッション追加]ダイアログボックスを表示します。
3. [既存セッションの使用]ラジオボタンをチェックしてください。
4. セッションの名前を入力してください。
5. 現在のプロジェクトにインポートしたい既存のセッションファイルを参照してください。
6. [セッションファイルへのリンク]チェックボックスをオンにしてください。
7. [書き込み不可属性でのセッションファイル生成]チェックボックスをオンにしてください。HEW はこのセッションへの変更を保存することはできません。このセッションを開いて、データを読み取ることができます。



(4) セッションを削除する

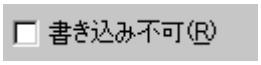
セッションを削除するには

1. [デバッグ->デバッグセッション]を選択してください。[デバッグセッション]ダイアログボックスを表示します。
2. 削除したいセッションを選択してください。
3. [削除]ボタンをクリックしてください。
4. [OK]ボタンをクリックしください。

現在のセッションを削除することはできません。

(5) セッションを書き込み不可にする

セッションを書き込み不可にするには



1. [デバッグ->デバッグセッション]を選択してください。[デバッグセッション]ダイアログボックスを表示します。
2. 書き込み不可にしたいセッションを選択してください。
3. [プロパティ]ボタンをクリックしてください。[セッションプロパティ]ダイアログボックスを表示します。
4. [書き込み不可]チェックボックスをオンにしてください。リンクをリードオンリーにします。これは、デバッガ設定ファイルを共有する場合、およびデータを誤って修正したくない場合に便利です。
5. [OK]ボタンをクリックしてください。

(6) セッション情報を保存する

セッションを保存するには

[ファイル->セッションの保存]メニュー項目を選択してください。

[基本設定->オプション]メニュー項目を選択し、[オプション]ダイアログボックスの[ワークスペース]タブを表示してください。[セッション保存前に確認]チェックボックスがオンの場合、セッション保存の前に、セッションを保存するかを確認するダイアログボックスを表示させることができます。ここで、[いいえ]を選択するとセッションへの変更を保存しません。

セッションを別名で保存するには

1. [デバッグ->デバッグセッション]を選択してください。[デバッグセッション]ダイアログボックスを表示します。
2. 保存したいセッションを選択してください。
3. [名前を付けて保存]ボタンをクリックしてください。[セッションの保存]ダイアログボックスを表示します。
4. 新しいファイルの場所をブラウズしてください。
5. セッションファイルを別の場所へエクスポートしたい場合は、[プロジェクトとのリンク]チェックボックスをオンにしないでください。現在のセッションの場所の代わりにこの場所を HEW で使用したい場合は、[プロジェクトとのリンク]チェックボックスをオンにしてください。
6. [保存]ボタンをクリックしてください。

これと同じ操作が、[ファイル->名前を付けてセッションを保存]を使っても可能です。

1. [ファイル->名前を付けてセッションを保存]を選択してください。[セッション名]ダイアログボックスを表示します。

2. 新しいセッションの名前を入力してください。
3. [OK]ボタンをクリックしてください。

(7) セッション情報のリフレッシュ

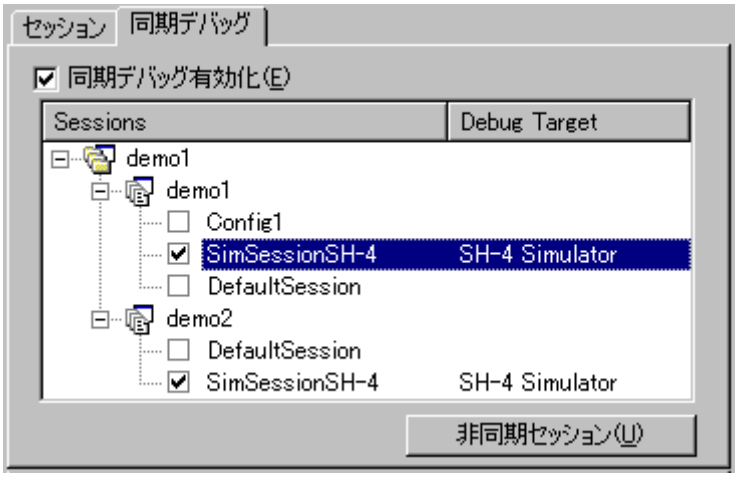
セッションを再び読み込むには

[ファイル->セッションのリフレッシュ]を選択してください。

これにより現在のセッションに加えられている変更を廃棄して、セッション情報を再び読み込みます。

(8) 複数ターゲットのデバッグ

複数ターゲットを同期してデバッグする方法は、「17.17 複数デバッギングプラットフォームを同期動作させる」を参照してください。



17.2 プログラムを表示する

この節では、プログラムをソースコードおよびアセンブリ言語ニーモニックとして表示する方法を説明します。

ソースコードは、次の 2 種類のウィンドウにある[ソースモード]で表示します。主な特徴を以下に示します。

エディタウィンドウ

- エディタウィンドウで表示するソースファイルは編集可能です。
- ソースコードをステップ実行などによりデバッグするとき、対応するソースファイル（現在のプログラムカウンタ(PC)のアドレスを含む）が変わると、別のウィンドウで自動的に開きます。対応するソースコード行がない場合にステップ実行すると、[ソースモード]は[逆アセンブリモード]に切り替わります。

逆アセンブリウィンドウ

- 逆アセンブリウィンドウで表示するソースファイルはすべて読み取り専用で編集できません。ソースファイルを編集するにはポップアップメニューからエディタウィンドウを開いて行います。
- ソースコードをステップ実行などによりデバッグするとき、逆アセンブリウィンドウは対応するソースファイル（現在の PC アドレスを含む）に自動的に切り替わります。逆アセンブリウィンドウが開いている間、エディタウィンドウは開かずに、必要に応じて逆アセンブリウィンドウ内にソースファイルを表示します。対応するソースコード行がない場合にステップ実行すると、[ソースモード]は[逆アセンブリモード]に切り替わります。
- デバッグ中に逆アセンブリウィンドウのソースモード表示を優先する機能があります。この機能を使用した場合の動作を次に示します。
    - 逆アセンブリウィンドウを[逆アセンブリモード]で開き、ソースのアドレスでブレークすると、逆アセンブリウィンドウは[ソースモード]に切り替わります。
    - ソースのアドレスでブレークしても、逆アセンブリウィンドウが開いている場合はエディタウィンドウを開きません。
    - プログラム実行前と後では、表示しているウィンドウの重なり順序を変更しません。

  逆アセンブリウィンドウの[逆アセンブリモード]でステップを続けたい場合は、[デバッグ->ステップモード->アセンブリ]を選択してください。

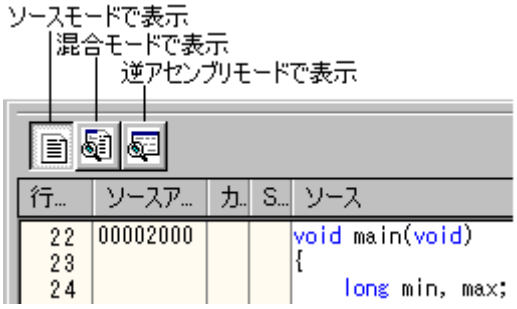
注：

ブレークすると HEW は PC 位置を表示します。プロジェクトがもともとのパスから移動した場合、ソースファイルを自動的に見つけることができない場合があります。この場合、HEW はソースファイル参照ダイアログボックスを開くので、手動でファイルを探すことができます。このパスは、このデバッグプロジェクトのほかのソースファイルを更新するために使用します。

17.2.1 エディタウィンドウを開く

ワークスペースウィンドウ上のソースファイルをダブルクリックするか、ソースファイル上で右ボタンをクリックしポップアップメニューを表示して[開く]を選択すると、エディタでファイルを開きます。

ツールバー



HEW V.4.00 以降のエディタでは、逆アセンブリ表示を統合しています。

モードを切り替えるツールバーがあります。各モードが有効である場合、ボタンをクリックすると表示を切り替えます。

以下の 3 つのモードがあります。

モード	機能
ソースモード	このモードは、標準的な HEW エディタで、ソースファイルを編集できます。また、ソースファイルを参照している場合は、キーワードを強調表示します。 行番号、アドレス、ブレークポイント、ブックマーク、およびソースコードを参照できます。
混合モード	ソースファイルの混合モードは、[逆アセンブリ]ウィンドウの混合モードとは異なります。継続する逆アセンブリを示すかわりに、ソースコードの各行に関連する逆アセンブリを示します。このモードは表示を編集できません。モジュールがダウンロードされている場合のみ有効になります。 行番号、ブレークポイント、アドレス、オブジェクトコード、ラベル、および混合（ソース・逆アセンブリ）を参照できます。
逆アセンブリモード	逆アセンブリモードは、実際に継続する逆アセンブリコードをアドレス順に示します。これは、[表示->逆アセンブリ]メニュー項目をクリックするのと同じです。ターゲットがセッションに接続している場合のみ有効になります。 ブレークポイント、アドレス、オブジェクトコード、ラベル、および逆アセンブリコードを参照できます。

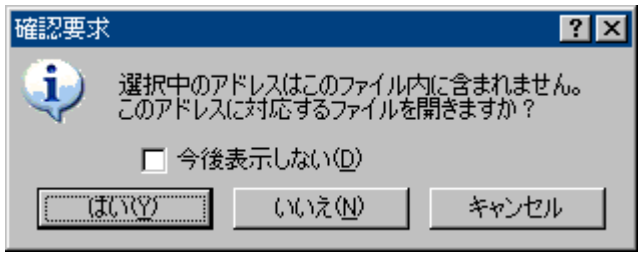
注：

以下の条件下で、[ソースモード]から[混合モード]へ切り替えることはできません。

1. ターゲットが現在のセッションに接続していない。
2. ダウンロードモジュールをダウンロードしていない。
3. 現在のプロジェクトに有効なデバッグ情報がない。
4. 現在表示しているファイルを編集したが、変更内容を保存していない。

[逆アセンブリモード]で、アドレスがそのソースファイルのアドレス範囲から外れ、別のソースファイルのアドレス範囲に変わった場合、[ソースモード]または[混合モード]へ切り替えると下記の確認ダイアログボックスが表示されます。

アドレスに対応する別のファイルを開く場合は「はい」を選択してください。切り替える前の位置で表示するには「いいえ」を選択してください。他のモードへ切り替えない場合は、「キャンセル」を選択してください。



確認要求ダイアログボックスを表示しない場合、[今後表示しない]チェックボックスをオンにしてください。

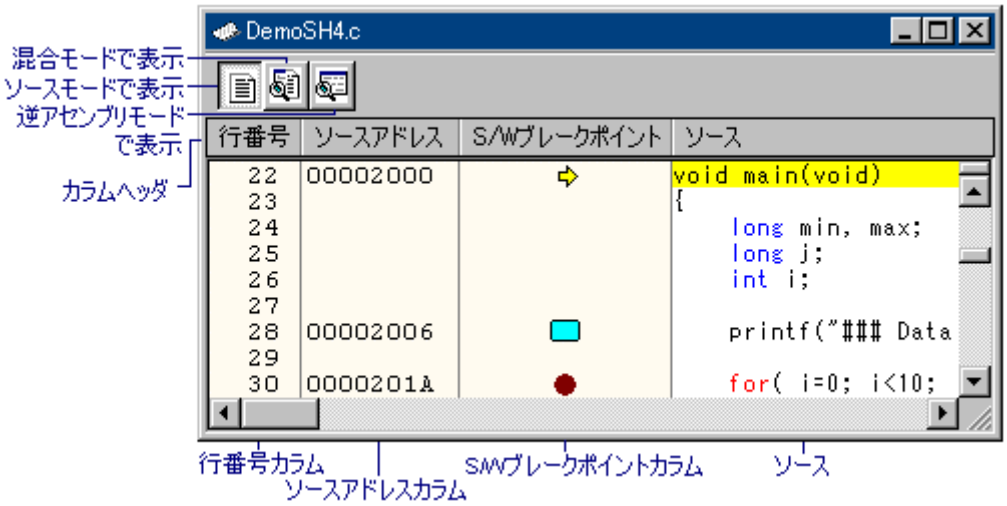
確認要求ダイアログボックスを再び表示するには

1. [基本設定->オプション]を選択してください。[オプション]ダイアログボックスを表示します。
2. [確認]タブを選択してください。
3. [表示モード切り替えで別ファイルを開く時の確認]チェックボックスをオンにしてください。このチェックボックスのデフォルトはオンです。
4. [OK]をクリックしてください。

(1) ソースコードを表示する

[ソースモードで表示]ボタンをクリックしたときにソースファイルのコードを表示します。

ウィンドウの構成



- 各モードが有効である場合、ツールバーのボタンをクリックすると表示を切り替えます。混合モードの逆アセンブリを表示するには、[混合モードで表示]ボタンをクリックしてください。逆アセンブリを表示するには、[逆アセンブリモードで表示]ボタンをクリックしてください。
- [ソース]領域の左側には以下に示すカラムがあります。

カラム名	内容
行番号	ソースファイルに対応する行番号を表示します。
ソースアドレス	モジュールをダウンロードすると、ソースファイルに対応するアドレスを表示します。
S/W ブレークポイント	PC 位置(➪)、PC ブレークポイント位置(●)、ブックマーク位置(▭)を表示します。ダブルクリックにより PC ブレークポイントを設定または解除します。

- [ソース]領域はシンタックスを強調表示したコードを含みます。

ウィンドウのオプション

[ソース]領域内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

ポップアップメニュー項目		マクロの記録	機能
コンパイル '<ファイル名>'		●	ファイルをコンパイルします。
開く '<ファイル名>'		-	選択されたファイル名を元にファイルを開きます。
切り取り		-	選択部分を削除し Windows®クリップボードに貼り付けます。
コピー		-	選択部分をコピーし Windows®クリップボードに貼り付けます。
貼り付け		-	Windows®クリップボードの内容をコピーしてアクティブウィンドウのカーソル位置に貼り付けます。
プロジェクトにファイルの追加		-	ファイルをプロジェクトに追加します。
定義へジャンプ '<ナビゲーション項目名>'		-	#define、C 関数、または C++クラスなどのナビゲーション項目の定義位置を表示します。
ナビゲーションリスト		-	スマートエディタ機能が有効な場合に、アルファベット順に C Define、C Function、または C++ Classes のナビゲーションリストをポップアップウィンドウで表示します。項目上をダブルクリックすると、カーソル位置にその文字列を貼り付けます。
検索		-	現在のファイルのテキストを検索します。
置換		-	テキストを置換します。
ジャンプ		-	指定した行にジャンプします。
括弧の呼応		-	括弧の組み合わせを見つけます。
ブックマーク	ブックマークの挿入/削除	-	ブックマークを設定/解除します。
	次のブックマーク	-	次のブックマークにジャンプします。
	前のブックマーク	-	同じファイルの 1 つ前のブックマークに戻ります。
	すべてのブックマークの削除	-	ファイル内のすべてのブックマークを削除します。
テンプレート	テンプレートの定義	-	テンプレートを設定します。
	テンプレートの挿入	-	テンプレートを挿入します。
ブレークポイントの挿入/削除		●	標準のブレークポイント(PC ブレークポイント)を挿入または削除します。
ブレークポイントの有効化/無効化		●	現在設定している PC ブレークポイントを有効または無効とします。
表示カラムの設定		-	表示カラムを設定します。
カラム	カラム名	-	表示カラムを設定します。
カラムヘッダの表示/非表示		-	カラムヘッダの表示/非表示を切り替えます。
インスタントウォッチ***1**		-	変数名を現在のテキストカーソル(マウスカーソルではありません)の位置から抜き出して名前とともにインスタントウォッチダイアログボックスを表示します。ソースのある行のみ有効になります。
カーソル位置まで実行		●	現在の PC アドレスからユーザプログラムの実行を開始します。プログラムは、PC がテキストカーソルの位置によって指定されたアドレスに達するか、別のブレーク条件が成立するまで実行します。

カーソル位置に PC 値を設定	●	PC の値をテキストカーソルの位置によって指定されたアドレスに変更します。
PC 位置を表示	-	PC 位置のソースファイルまたは逆アセンブリを表示します。
逆アセンブリ		現在のソース行に対応したアドレスの[逆アセンブリ]ウィンドウを表示します。
プロパティ	-	ファイルのプロパティを表示します。

***1.** 機能のサポートは、デバッギングプラットフォームに依存します。

(2) アセンブリ言語コードを表示する

[逆アセンブリモード]の表示開始アドレスは、[ソースモード]のカーソル位置に対応するアドレスとなります。逆アセンブリコードの表示は、デバッギングプラットフォームがセッションに接続されている場合のみ有効になります。

ウィンドウの構成

- 各モードが有効である場合、ツールバーのボタンをクリックすると表示を切り替えます。混合モードの逆アセンブリを表示するには、[混合モードで表示]ボタンをクリックしてください。ソースコードを表示するには、[ソースモードで表示]ボタンをクリックしてください。
- [逆アセンブリ]領域の左側には以下に示すカラムがあります。

カラム名	内容
S/W ブレークポイント - ASM	PC 位置(➪)、PC ブレークポイント位置(●)を表示します。ダブルクリックにより PC ブレークポイントを設定または解除します。
逆アセンブリアドレス	逆アセンブリアドレスを表示します。ダブルクリックにより[アドレス指定]ダイアログボックスを表示します。逆アセンブリの表示開始アドレスを入力してください。
オブジェクトコード	オブジェクトコードを表示します。
ラベル	ラベルを表示します。モジュールをダウンロードしていない場合このカラムは使用できません。

- [逆アセンブリモード]では、[逆アセンブリ]領域のアセンブリ言語コードをダブルクリックすると、[アセンブル]ダイアログボックスを表示します。アセンブリ言語コードを入力してください。

ウィンドウのオプション

[逆アセンブリ]領域内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

ポップアップメニュー項目	マクロの記録	機能
リフレッシュ	-	逆アセンブリの最新情報を取得し表示内容を更新します。
リフレッシュ抑止	-	[逆アセンブリモード]のメモリ範囲がリフレッシュされないようにロックします。
ソースの表示	-	アドレスに対応するソースの位置でエディタを表示します。
表示アドレス設定	-	新しい開始アドレスを指定します。
カーソル位置まで実行	●	現在の PC アドレスからプログラムを実行します。プログラムは、PC がテキストカーソルの位置によって指定されたアドレスに達するか、別のブレーク条件が成立するまで実行します。
PC 位置を表示	-	PC 位置の逆アセンブリを表示します。
カーソル位置に PC 値を設定	●	PC の値をテキストカーソルの位置によって指定されたアドレスに変更します。
編集	-	任意のアドレスの命令を変更します。
範囲で検索	-	指定した範囲で文字列を検索します。
コピー	-	選択部分をコピーし Windows®クリップボードに貼り付けます。
表示カラムの設定	-	表示カラムを設定します。
カラムヘッダの表示/非表示	-	カラムヘッダの表示、非表示を切り替えます。
テキストファイルに保存	-	指定した範囲をテキストで保存します。
印刷	-	指定した範囲を印刷します。
ブレークポイントの挿入/削除	●	標準のブレークポイント(PC ブレークポイント)を挿入または削除します。
ブレークポイントの有効化/無効化	●	現在設定している PC ブレークポイントを有効または無効とします。

(3) 逆アセンブリ表示をリフレッシュする

外部オペレーションが逆アセンブリ表示のメモリを変更する場合、HEW はメモリの変更を検知できません。例えば、表示されている範囲のメモリに対して外部フラッシュユーティリティを使用してプログラミングしても、逆アセンブリは変更されません。

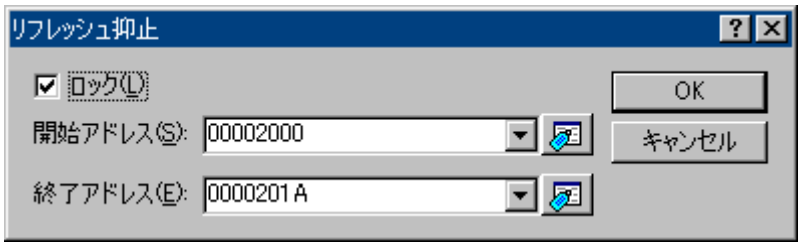
[逆アセンブリ]領域内（[逆アセンブリモード]）で右クリックし、ポップアップメニューから[リフレッシュ]を選択してください。

逆アセンブリの最新情報を取得し表示内容を更新します。

(4) 逆アセンブリ表示のリフレッシュを抑止する

逆アセンブリ表示のメモリ範囲がリフレッシュされないようにロックします。この機能を逆アセンブリのリフレッシュ抑止と呼びます。

[逆アセンブリ]領域内（[逆アセンブリモード]）で右クリックし、ポップアップメニューから[リフレッシュ抑止]を選択してください。[リフレッシュ抑止]ダイアログボックスを表示します。



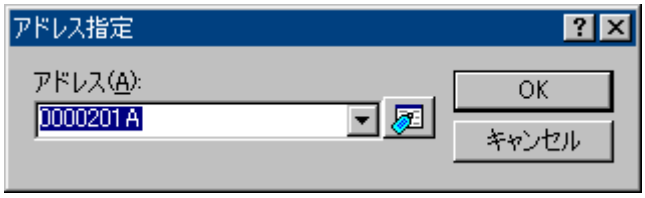
[ロック]チェックボックスをオンにしてください。他のコントロールが有効になります。

ロックおよびキャッシュする[開始アドレス]および[終了アドレス]を指定し[OK]ボタンをクリックしてください。ロックしたメモリ範囲のみを表示します。

(5) 特定のアドレスを見る

逆アセンブリ表示でプログラムを見ている場合、プログラム内の他の箇所を見たいときがあります。そのような場合、プログラム内のコードをスクロールせずに特定のアドレスにジャンプすることができます。

[逆アセンブリ]領域内（[逆アセンブリモード]）で右クリックし、ポップアップメニューから[表示アドレス設定]を選択します。[アドレス指定]ダイアログボックスを表示します。



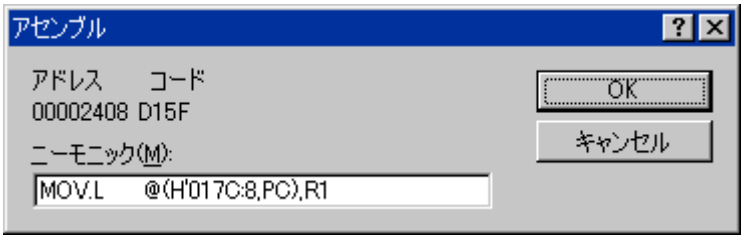
[アドレス]エディットボックスにアドレスを指定し、[OK]ボタンをクリックしてください。アドレスは、ラベル名で入力することも可能です。

逆アセンブリ表示を更新して入力したアドレスのコードを表示します。

オーバーロード関数またはクラス名を入力した場合、[関数選択]ダイアログボックスを開くので、関数を選択してください（機能のサポートは、デバッギングプラットフォームに依存します）。

(6) アセンブリ言語コードを編集する

逆アセンブリ表示（[逆アセンブリモード]）で編集したい命令をダブルクリックすることによって、アセンブリ言語コードを編集できます。[アセンブル]ダイアログボックスを表示します。



アドレス、オブジェクトコード、およびニーモニックを表示します。

[ニーモニック]フィールドに新しい命令を入力（または古い命令を編集）します。

”Enter”キーを押すと、メモリ内容を新しいオブジェクトコードに書き換えて、次の命令に移ります。

[OK]ボタンをクリックすると、メモリ内容を新しい命令コードに書き換えて、ダイアログボックスを閉じます。

[キャンセル]ボタンをクリックするか”Esc”キーを押すと、メモリ内容を書き換えずに、ダイアログボックスを閉じます。

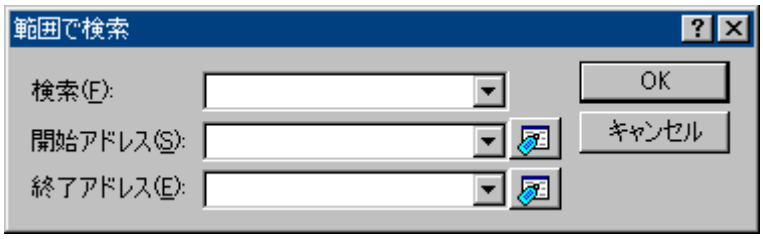
注：

アセンブリ言語コードは、現在のメモリ内容から表示しています。メモリ内容を修正すると、[アセンブル]ダイアログボックスおよび逆アセンブリ表示では、新しいアセンブリ言語コードを表示します。しかしソース表示は変更しません。これはソースファイルにアセンブラを含む場合も同じです。

(7) 逆アセンブリで範囲検索する

逆アセンブリ表示の特定のアドレス範囲でテキスト文字列を検索できます。

[逆アセンブリ]領域内（[逆アセンブリモード]]）で右クリックし、ポップアップメニューから[範囲で検索]を選択してください。[範囲で検索]ダイアログボックスを表示します。

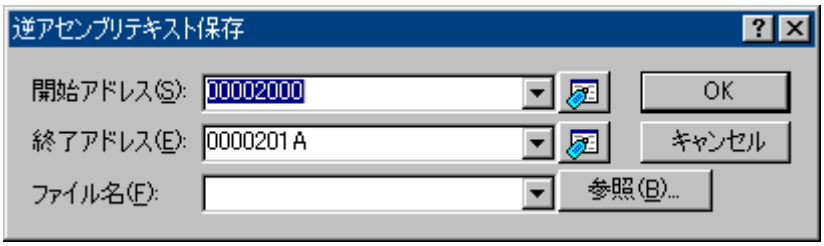


検索する文字列、開始アドレスおよび終了アドレスを指定してください。[OK]ボタンをクリックしてください。範囲にある最初にあらわれた該当文字列を選択します。

(8) 逆アセンブリテキストを保存する

特定のアドレス範囲の逆アセンブリ表示内容をテキストに保存することができます。

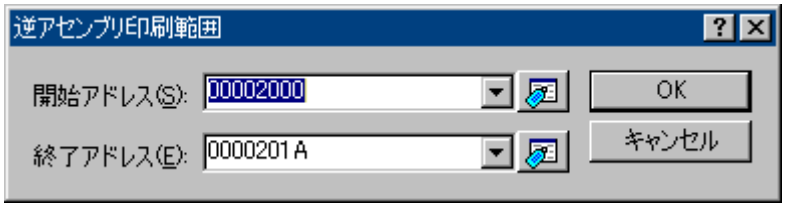
[逆アセンブリ]領域内（[逆アセンブリモード]）で右クリックし、ポップアップメニューから[テキストファイルに保存]を選択してください。[逆アセンブリテキスト保存]ダイアログボックスを表示します。



保存する開始アドレス、終了アドレス、およびファイル名を指定してください。[OK]ボタンをクリックしてください。

(9)　　逆アセンブリ表示を印刷する

逆アセンブリ表示を印刷するには、ウィンドウがアクティブな場合に[ファイル->印刷]メニュー、[印刷]ツールバーボタン( )を使うか、または[逆アセンブリ]領域内（[逆アセンブリモード]）のポップアップメニューにある[印刷]を使います。[印刷]を選択すると、[逆アセンブリ印刷範囲]ダイアログボックスを表示します。特定のアドレス範囲の表示内容を印刷することができます。



印刷する開始アドレスと終了アドレスを指定してください。[OK]ボタンをクリックすると、標準のプリントフォーマットを選択するダイアログボックスが表示されます。ここで、使用するプリンタとページ設定のオプションを選択することができます。[印刷]ボタンをクリックしてください。

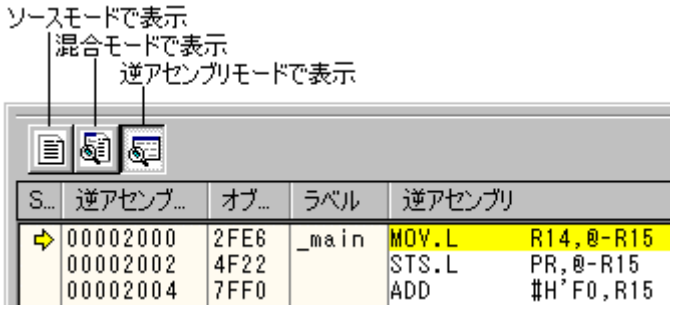
17.2.2　逆アセンブリウィンドウを開く

次のいずれかの方法によりアセンブリ言語レベルでコードを表示できます。

- [逆アセンブリ]ツールバーボタン ( ) をクリックする。
- [表示->逆アセンブリ]を選択する。
- ”Ctrl+D”キーを押す。

この場合、[逆アセンブリ]ウィンドウは現在のプログラムカウンタ（PC）の位置で開きます。

ツールバー



HEW V.4.02 以降の逆アセンブリウィンドウはソース表示を使用できます。

モードを切り替えるツールバーがあります。各モードが有効である場合、ボタンをクリックすると表示を切り替えます。

以下の 3 つのモードがあります。

モード	機能
ソースモード	現在の PC アドレスに対応するソースファイルがある場合、[逆アセンブリ]ウィンドウ内にソースファイルを開きます。 ステップ実行などによるプログラムカウンタ(PC)位置の変化に追従し、同一ウィンドウ上で PC に対応したソースファイルおよびコードを表示します。別のソースファイル領域へ移動しても、新たなウィンドウは開きません。 このモードは、ソースファイルを読み取り専用で編集できません。モジュールがダウンロードされている場合のみ有効になります。 また、ソースファイルを参照している場合は、キーワードを強調表示します。 行番号、アドレス、ブレークポイント、およびソースコードを参照できます。
混合モード	逆アセンブリウィンドウの混合モードは、ソースファイルの混合モードとは異なります。継続する逆アセンブリを示すかわりに、ソースコードの各行に関連する逆アセンブリを示します。 このモードはアセンブリ言語コードを編集できます。モジュールがダウンロードされている場合のみ有効になります。 ブレークポイント、アドレス、オブジェクトコード、ラベル、および逆アセンブリコードを参照できます。
逆アセンブリモード	逆アセンブリモードは、実際に継続する逆アセンブリコードをアドレス順に示します。 このモードはアセンブリ言語コードを編集できます。ターゲットがセッションに接続している場合のみ有効になります。 ブレークポイント、アドレス、オブジェクトコード、ラベル、および逆アセンブリコードを参照できます。

(1) ソースコードを表示する

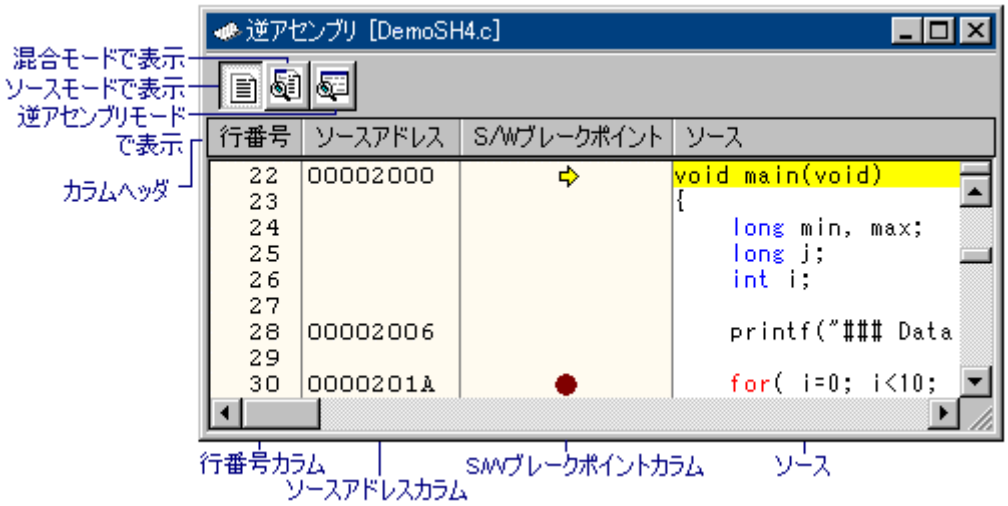
[逆アセンブリ]ウィンドウ内から[ソースモード]に切り替えるとき、現在の PC アドレスに対応するソースファイルがある場合、[逆アセンブリ]ウィンドウ内にソースファイルを開きます。

現在の PC アドレスに対応するソースファイルがない場合は、ソースファイルを開くためにダイアログボックスを表示します。ファイルを選択し[開く]ボタンをクリックしてください。逆アセンブリウィンドウのソースモードでは、テキストベースのファイルはプロジェクトのソースファイルに限らず、すべて選択できます。

このウィンドウで開くソースファイルは、すべて読み取り専用で編集できません。ソースファイルの表示とデバッグを目的とするためです。

この表示モードはモジュールがダウンロードされている場合のみ有効になります。

ウィンドウの構成



- 各モードが有効である場合、ツールバーのボタンをクリックすると表示を切り替えます。混合モードの逆アセンブリを表示するには、[混合モードで表示]ボタンをクリックしてください。逆アセンブリを表示するには、[逆アセンブリモードで表示]ボタンをクリックしてください。
- [ソース]領域の左側には以下に示すカラムがあります。

カラム名	内容
行番号	ソースファイルに対応する行番号を表示します。
ソースアドレス	モジュールをダウンロードすると、ソースファイルに対応するアドレスを表示します。
S/W ブレークポイント	PC 位置(➪)、PC ブレークポイント位置(●)を表示します。ダブルクリックにより PC ブレークポイントを設定または解除します。

- [ソース]領域はシンタックスを強調表示したコードを含みます。

ウィンドウのオプション

[ソース]領域内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

ポップアップメニュー項目	マクロの記録	機能
編集ソースの表示	-	編集するソースファイルをエディタウィンドウで開きます。
ファイルを開く	-	逆アセンブリウィンドウ内に表示するファイルを選択します。
コピー	-	選択部分をコピーし Windows®クリップボードに貼り付けます。
検索	-	現在のファイルのテキストを検索します。
ジャンプ	-	指定した行にジャンプします。
ブレークポイントの挿入/削除	●	標準のブレークポイント(PC ブレークポイント)を挿入または削除します。
ブレークポイントの有効化/無効化	●	現在設定している PC ブレークポイントを有効または無効とします。
カラムヘッダの表示/非表示		カラムヘッダの表示/非表示を切り替えます。
表示カラムの設定	-	表示カラムを設定します。
カラム　カラム名	-	表示カラムを設定します。
インスタントウォッチ ***1**	-	変数名を現在のテキストカーソル(マウスカーソルではありません)の位置から抜き出して名前とともにインスタントウォッチダイアログボックスを表示します。ソースのある行のみ有効になります。
カーソル位置まで実行	●	現在の PC アドレスからユーザプログラムの実行を開始します。プログラムは、PC がテキストカーソルの位置によって指定されたアドレスに達するか、別のブレーク条件が成立するまで実行します。
カーソル位置に PC 値を設定	●	PC の値をテキストカーソルの位置によって指定されたアドレスに変更します。
PC 位置を表示	-	PC 位置のソースファイルまたは逆アセンブリを表示します。
プロパティ	-	ファイルのプロパティを表示します。

***1.** 機能のサポートは、デバッギングプラットフォームに依存します。

(2) 編集するソースファイルを開く

逆アセンブリウィンドウのソースモードは、ソースを参照しデバッグが可能ですが編集はできません。

逆アセンブリウィンドウのソースモードで現在開いているソースファイルを編集することができます。ポップアップメニューから[編集ソースの表示]を選択してください。エディタウィンドウで編集するファイルを開きます。

また、ワークスペースウィンドウ、または[ファイル->開く]メインメニューによって手動で編集するファイルを開くこともできます。

(3) 逆アセンブリウィンドウで別のファイルを開く

逆アセンブリウィンドウ内にソースファイルを開くことができます。ポップアップメニューから[ファイルを開く]を選択してください。このメニュー項目は、逆アセンブリウィンドウにあるすべての表示モードのポップアップメニューで利用できます。

このメニュー項目を選択すると、ソースファイルを開くためにファイルを参照するダイアログボックスを表示します。テキストベースのファイルはプロジェクトのソースファイルに限らず、すべて選択できます。

(4) デバッグ中に逆アセンブリウィンドウのソースモード表示を優先する

デバッグ中に逆アセンブリウィンドウのソースモード表示を優先することができます。

デバッグ中に逆アセンブリウィンドウのソースモード表示を優先するには

1. [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を表示します。
2. [デバッグ]タブを選択します。
3. [デバッグ中に逆アセンブリウィンドウのソースモード表示を優先]チェックボックスをオンにします。チェックボックスのデフォルトはオフです。
4. [OK]をクリックします。

この機能を使用した場合の動作を次に示します。

- 逆アセンブリウィンドウを[逆アセンブリモード]で開き、ソースファイルのアドレスでブレークすると、逆アセンブリウィンドウは[ソースモード]に切り替わります。
- ソースファイルのアドレスでブレークしても、逆アセンブリウィンドウが開いている場合はエディタウィンドウを開きません。
- プログラム実行前と後では、表示しているウィンドウの重なり順序を変更しません。

逆アセンブリウィンドウの[逆アセンブリモード]でステップを続けたい場合は、[デバッグ->ステップモード->アセンブリ]を選択してください。

(5) アセンブリ言語コードを表示する

[逆アセンブリモード]の表示開始アドレスは現在の PC の位置で開きます。逆アセンブリコードの表示は、デバッギングプラットフォームがセッションに接続されている場合のみ有効になります。

ウィンドウの構成

- 各モードが有効である場合、ツールバーのボタンをクリックすると表示を切り替えます。混合モードの逆アセンブリを表示するには、[混合モードで表示]ボタンをクリックしてください。ソースコードを表示するには、[ソースモードで表示]ボタンをクリックしてください。
- [逆アセンブリ]領域の左側には以下に示すカラムがあります。

カラム名	内容
S/W ブレークポイント - ASM	PC 位置(➪)、PC ブレークポイント位置(●)を表示します。ダブルクリックにより PC ブレークポイントを設定または解除します。
逆アセンブリアドレス	逆アセンブリアドレスを表示します。ダブルクリックにより[アドレス指定]ダイアログボックスを表示します。逆アセンブリの表示開始アドレスを入力してください。
オブジェクトコード	オブジェクトコードを表示します。
ラベル	ラベルを表示します。モジュールをダウンロードしていない場合このカラムは使用できません。

- [逆アセンブリモード]および[混合モード]では、[逆アセンブリ]領域のアセンブリ言語コードをダブルクリックすると、[アセンブル]ダイアログボックスを表示します。アセンブリ言語コードを入力してください。

ウィンドウのオプション

[逆アセンブリ]領域内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

ポップアップメニュー項目	マクロの記録	機能
ファイルを開く	-	逆アセンブリウィンドウ内に表示するファイルを選択します。
リフレッシュ	-	逆アセンブリの最新情報を取得し表示内容を更新します。
リフレッシュ抑止	-	[逆アセンブリ]ウィンドウのメモリ範囲がリフレッシュされないようにロックします。
ソースの表示	-	アドレスに対応するソースの位置でエディタウィンドウを表示します。
表示アドレス設定	-	新しい開始アドレスを指定します。
カーソル位置まで実行	●	現在の PC アドレスからプログラムを実行します。プログラムは、PC がテキストカーソルの位置によって指定されたアドレスに達するか、別のブレーク条件が成立するまで実行します。
PC 位置を表示	-	PC 位置の逆アセンブリを表示します。
カーソル位置に PC 値を設定	●	PC の値をテキストカーソルの位置によって指定されたアドレスに変更します。

編集	-	任意のアドレスの命令を変更します。
範囲で検索	-	指定した範囲で文字列を検索します。
コピー	-	選択部分をコピーし Windows®クリップボードに貼り付けます。
表示カラムの設定	-	表示カラムを設定します。
カラムヘッダの表示/非表示	-	カラムヘッダの表示、非表示を切り替えます。
テキストファイルに保存	-	指定した範囲をテキストで保存します。
印刷	-	指定した範囲を印刷します。
ブレークポイントの挿入/削除	●	標準のブレークポイント(PC ブレークポイント)を挿入または削除します。
ブレークポイントの有効化/無効化	●	現在設定している PC ブレークポイントを有効または無効とします。

(6) 逆アセンブリ表示をリフレッシュする

外部オペレーションが逆アセンブリ表示のメモリを変更する場合、HEW はメモリの変更を検知できません。例えば、表示されている範囲のメモリに対して外部フラッシュユーティリティを使用してプログラミングしても、逆アセンブリは変更されません。

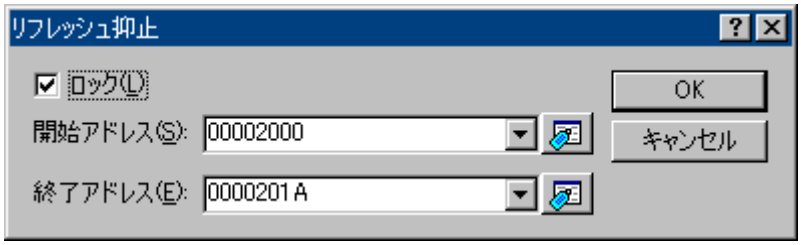
[逆アセンブリ]領域内（[逆アセンブリモード]または[混合モード]）で右クリックし、ポップアップメニューから[リフレッシュ]を選択してください。

逆アセンブリの最新情報を取得し表示内容を更新します。

(7) 逆アセンブリ表示のリフレッシュを抑止する

逆アセンブリ表示のメモリ範囲がリフレッシュされないようにロックします。この機能を逆アセンブリのリフレッシュ抑止と呼びます。

[逆アセンブリ]領域内（[逆アセンブリモード]または[混合モード]）で右クリックし、ポップアップメニューから[リフレッシュ抑止]を選択してください。[リフレッシュ抑止]ダイアログボックスを表示します。



[ロック]チェックボックスをオンにしてください。他のコントロールが有効になります。

ロックおよびキャッシュする[開始アドレス]および[終了アドレス]を指定し[OK]ボタンをクリックしてください。ロックしたメモリ範囲のみを表示します。

(8) 特定のアドレスを見る

逆アセンブリ表示でプログラムを見ている場合、プログラム内の他の箇所を見たいときがあります。そのような場合、プログラム内のコードをスクロールせずに特定のアドレスにジャンプすることができます。

[逆アセンブリ]領域内（[逆アセンブリモード]または[混合モード]）で右クリックし、ポップアップメニューから[表示アドレス設定]を選択します。[アドレス指定]ダイアログボックスを表示します。

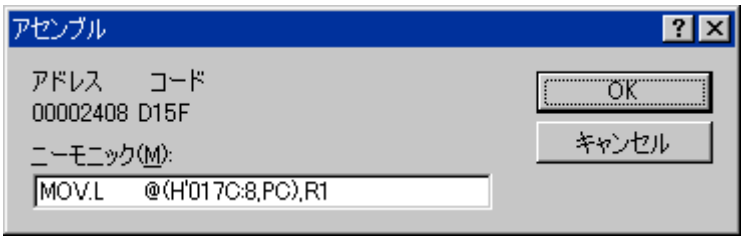
[アドレス]エディットボックスにアドレスを指定し、[OK]ボタンをクリックしてください。アドレスは、ラベル名で入力することも可能です。

逆アセンブリ表示を更新して入力したアドレスのコードを表示します。

オーバーロード関数またはクラス名を入力した場合、[関数選択]ダイアログボックスを開くので、関数を選択してください（機能のサポートは、デバッギングプラットフォームに依存します）。

(9) アセンブリ言語コードを編集する

逆アセンブリ表示（[逆アセンブリモード]または[混合モード]）で編集したい命令をダブルクリックすることによって、アセンブリ言語コードを編集できます。[アセンブル]ダイアログボックスを表示します。



アドレス、オブジェクトコード、およびニーモニックを表示します。

[ニーモニック]フィールドに新しい命令を入力（または古い命令を編集）します。

”Enter”キーを押すと、メモリ内容を新しいオブジェクトコードに書き換えて、次の命令に移ります。

[OK]ボタンをクリックすると、メモリ内容を新しい命令コードに書き換えて、ダイアログボックスを閉じます。

[キャンセル]ボタンをクリックするか”Esc”キーを押すと、メモリ内容を書き換えずに、ダイアログボックスを閉じます。

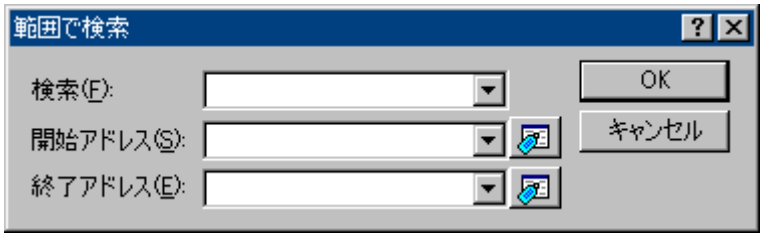
注：

アセンブリ言語コードは、現在のメモリ内容から表示しています。メモリ内容を修正すると、[アセンブル]ダイアログボックスおよび逆アセンブリ表示では、新しいアセンブリ言語コードを表示します。しかしソース表示は変更しません。これはソースファイルにアセンブラを含む場合も同じです。

(10) 逆アセンブリで範囲検索する

逆アセンブリ表示の特定のアドレス範囲でテキスト文字列を検索できます。

[逆アセンブリ]領域内（[逆アセンブリモード]または[混合モード]）で右クリックし、ポップアップメニューから[範囲で検索]を選択してください。[範囲で検索]ダイアログボックスを表示します。

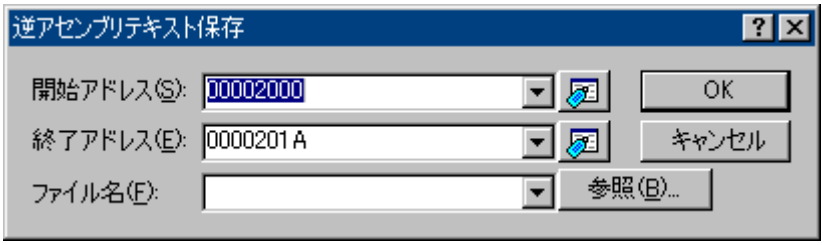


検索する文字列、開始アドレスおよび終了アドレスを指定してください。[OK]ボタンをクリックしてください。範囲の中で最初にあらわれた該当文字列を選択表示します。

(11) 逆アセンブリテキストを保存する

特定のアドレス範囲の逆アセンブリ表示内容をテキストに保存することができます。

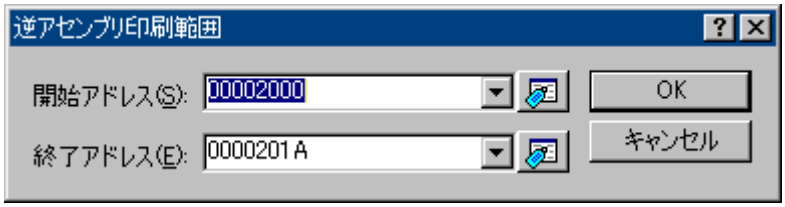
[逆アセンブリ]領域内（[逆アセンブリモード]または[混合モード]）で右クリックし、ポップアップメニューから[テキストファイルに保存]を選択してください。[逆アセンブリテキスト保存]ダイアログボックスを表示します。



保存する開始アドレス、終了アドレス、およびファイル名を指定してください。[OK]ボタンをクリックしてください。

(12) 逆アセンブリ表示を印刷する

逆アセンブリ表示を印刷するには、ウィンドウがアクティブな場合に[ファイル->印刷]メニュー、[印刷]ツールバーボタン( )を使うか、または[逆アセンブリ]領域内（[逆アセンブリモード]または[混合モード]）のポップアップメニューにある[印刷]を使います。[印刷]を選択すると、[逆アセンブリ印刷範囲]ダイアログボックスを表示します。特定のアドレス範囲の表示内容を印刷することができます。



印刷する開始アドレスと終了アドレスを指定してください。[OK]ボタンをクリックすると、標準のプリントフォーマットを選択するダイアログボックスが表示されます。ここで、使用するプリンタとページ設定のオプションを選択することができます。[印刷]ボタンをクリックしてください。

17.2.3 現在の PC 位置を表示する

PC 位置を表示するには、[PC 位置の表示]ツールバーボタン ( ) をクリックするか、[デバッグ->PC 位置を表示]を選択してください。現在の PC の位置で、エディタウィンドウまたは逆アセンブリウィンドウを開きます。

17.2.4 PC 位置の行を強調表示する

エディタウィンドウおよび逆アセンブリウィンドウでは、PC の位置は[S/W ブレークポイント]カラムにアイコンで表示されます。PC 位置の行のソースコードおよびアセンブリコード表示領域を強調表示する機能があります。

PC 位置の行を強調表示するには

1. [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を表示します。
2. [エディタ]タブを選択します。
3. [PC カーソル行のハイライト表示]チェックボックスをオンにします。チェックボックスのデフォルトはオンです。
4. [OK]ボタンをクリックします。

この行の色は、[表示形式]ダイアログボックスでカスタマイズできます。デフォルトの行の色は、テキスト（前面）が黒色で背景が黄色です。

PC 位置の行の色を変更するには

1. [基本設定->表示の形式]を選択し、[表示形式]ダイアログボックスを表示します。
2. PC 位置の行の色を変更したいアイテムをダイアログボックスの左側のツリーから選択し展開します。

- エディタウィンドウおよび逆アセンブリウィンドウ（ソースモード）の場合は[Source]を選択し拡張する。
- エディタウィンドウおよび逆アセンブリウィンドウ（混合モード、逆アセンブリモード）の場合は[逆アセンブリ]を選択し拡張する。

3. [PC Line Highlight]カテゴリを選択します。
4. [カラー]タブの[前面]リストと[背景]リストの選択を変更します。
5. [OK]ボタンをクリックします。

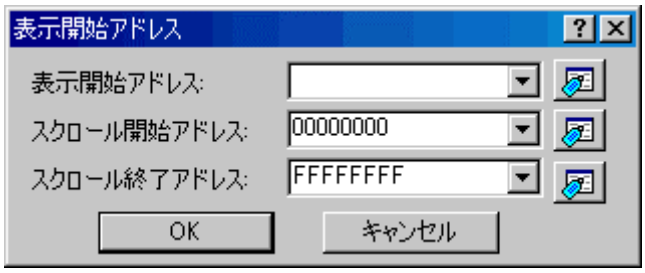
17.3 メモリを操作する

この節では、メモリ内容を操作する方法を説明します。操作の種類としては、色々なフォーマットでの表示、メモリブロックのフィル／移動、およびロードモジュールファイルのロード／ベリファイがあります。

17.3.1 メモリウィンドウを開く

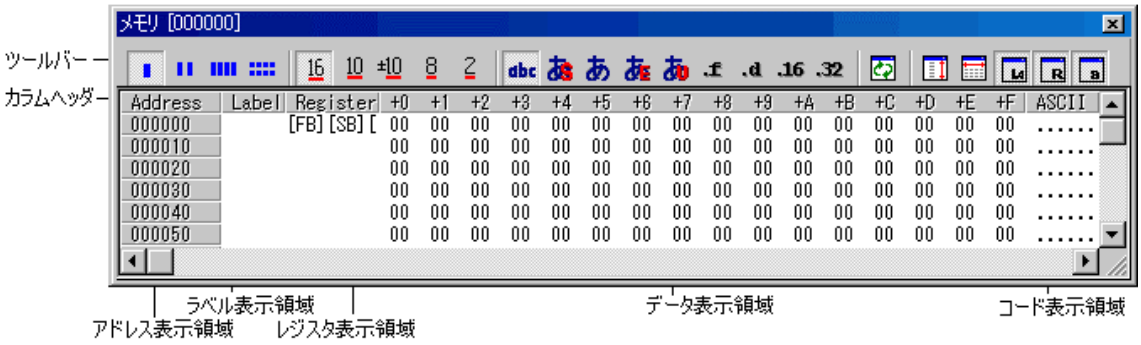
[メモリ]ウィンドウは、連続したメモリ内容をダンプ形式で表示するウィンドウです。

[メモリ]ウィンドウを開くには、[メモリ]ツールバーボタン( )をクリックするか、[表示->CPU->メモリ]を選択してください。メモリウィンドウでは、開く時に[表示開始アドレス]、[スクロール開始アドレス]、[スクロール終了アドレス]を指定できます。[表示開始アドレス]ダイアログボックスがオープンします。[表示開始アドレス]、[スクロール開始アドレス]、[スクロール終了アドレス]を指定してください。



[OK]ボタンをクリックするか、'Enter'キーを押すとダイアログボックスは閉じて[メモリ]ウィンドウが開きます。入力したスクロール開始アドレス、スクロール終了アドレス内でスクロールできます。

ウィンドウの構成



- データ表示領域カラムヘッダの[+n]は、[Address]からのメモリデータで、n は行の最初のアドレスからのオフセット値です。
- コード内容表示領域カラムヘッダは、コード名を表示します。
- レジスタ内容表示領域は、行頭のアドレスに割り付けられているレジスタ名を表示します。
- アドレス表示領域/データ表示領域/コード表示領域はインプレース編集により変更可能です。（「マクロの記録」）

- アドレス表示領域/ラベル表示領域をダブルクリックすることにより、表示開始アドレスを変更するためのダイアログボックスが開きます。
- データ表示領域/コード表示領域をダブルクリックすることにより、クリックしたアドレスのメモリ内容を変更するためのダイアログボックスが開きます。（「マクロの記録」）

ウィンドウのオプション

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

本ウィンドウは、基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目		ツールバーボタン	マクロの記録	機能
設定			●	任意のアドレスにデータを設定します。
フィル			●	任意のアドレス領域を一定データで充填します。
コピー			●	任意のアドレス領域のデータを別領域にコピーします。
比較***1**			●	任意の 2 つのアドレス領域のデータを比較します。
テスト***1**			-	任意のアドレス領域をテストします。
テキストファイルに保存			-	任意のアドレス領域をテキストファイルに保存します。
検索***1**			-	任意のアドレス領域のデータを検索します。
次を検索***1**			-	任意のアドレス領域の次のデータを検索します。
開始アドレス			-	表示開始アドレスを指定します。
スクロール範囲			-	スクロール範囲を指定します。
レジスタ(xxxxx)***1**		-	-	指定したレジスタが示すアドレスから表示します（レジスタは、製品により異なります）。
スタック追従***1**			-	スタックポインタ位置を追従します。
スタートアップシンボル			-	ダウンロード時に表示するラベル位置を指定します。
最新の情報に更新			-	メモリ内容を手動更新します。
表示固定		-	-	メモリ内容の更新を抑止します。
データ長	1byte		-	1 バイト単位で表示します。
	2byte		-	2 バイト単位で表示します。
	4byte		-	4 バイト単位で表示します。
	8byte		-	8 バイト単位で表示します。
基数	16 進数表示		-	16 進数で表示します。
	10 進数表示		-	10 進数で表示（符号なし）します。
	符号付 10 進数表示		-	10 進数で表示（符号付き）します。
	8 進数表示		-	8 進数で表示します。
	2 進数表示		-	2 進数で表示します。
表示コード	ASCII		-	ASCII コードで表示します。
	SJIS		-	SJIS コードで表示します。

	JIS		-	JIS コードで表示します。
	UNICODE		-	UNICODE で表示します。
	EUC		-	EUC コードで表示します。
	Float		-	単精度浮動小数点数で表示します。
	Double		-	倍精度浮動小数点数で表示します。
	16bit Fixed		-	16 ビット固定で表示します。
	32bit Fixed		-	32 ビット固定で表示します。
	24 bit Accum	-	-	24 ビット累算で表示します。
	40 bit Accum	-	-	40 ビット累算で表示します。
レイアウト	ラベル		-	ラベル表示領域の表示/表示なしを切り替えます。
	レジスタ		-	レジスタ表示部の表示/表示なしを切り替えます。
	コード		-	コード表示部の表示/表示なしを切り替えます。
カラム			-	表示桁数を指定します。
カバレッジ *1	有効		-	カバレッジ計測結果の表示/表示なしを切り替えます。
保存			●	任意のアドレス領域のメモリ内容を保存します。
読み込み			●	メモリにファイルをロードします。
分割		-	-	ウィンドウを分割表示します。
ツールバー表示		-	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ		-	-	ツールバーボタンをカスタマイズします。

***1.** 機能のサポートは、デバッギングプラットフォームに依存します。

17.3.2 任意のアドレスにデータを設定する

[メモリ]ウィンドウで任意のアドレスにデータを設定するには次の方法があります。

1. データ表示領域、コード表示領域でインプレース編集します。

2. 以下のいずれかの方法で[メモリ設定]ダイアログボックスを開きます。

- データ表示領域、コード表示領域をダブルクリックする。
- ポップアップメニューの[設定]を選択する。

[データ]領域に設定するデータ（値または文字）を入力してください。

[ベリファイ]チェックボックスでメモリベリファイを指定します。ベリファイ機能のサポートは、デバッギングプラットフォームに依存します。

値を設定する場合

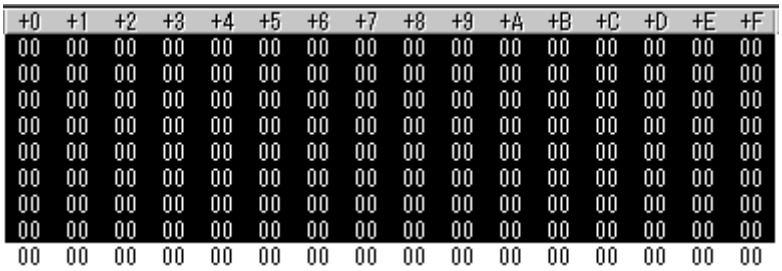
[データ形式]グループの[値]ボタンをクリックしてください。[データサイズ]領域にデータ長を指定してください。

文字を設定する場合

[データ形式]グループの[文字列]ボタンをクリックしてください。[コード]領域に文字コードを指定してください。

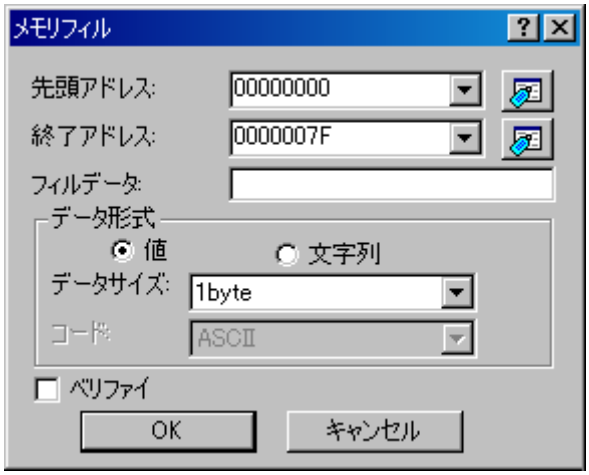
17.3.3 アドレス領域を選択する

メモリアドレス範囲が[メモリ]ウィンドウにある場合、最初のメモリユニット（[メモリ]ウィンドウディスプレイでの選択に従い）をクリックして最後のユニットまでマウスをドラッグすることによって領域を選択できます。選択した領域は強調表示します。



17.3.4 任意のアドレス領域を一定データで充填する

[メモリ]ウィンドウ上で充填するアドレス領域をドラッグして選択してください。ポップアップメニューから[フィル]を選択して[フィル]ダイアログボックスを開きます。



[フィルデータ]領域へ充填するデータ(値または文字)を入力してください。

[ベリファイ]チェックボックスでメモリベリファイを指定します。ベリファイ機能のサポートは、デバッギングプラットフォームに依存します。

充填するアドレス領域をドラッグしていない場合は、開始アドレス、終了アドレスを入力する必要があります。終了アドレスには先頭に'+'記号をつけることができ、この記号を入力すると開始アドレス＋入力した値が終了アドレスになります。

値を設定する場合

[データ形式]グループの[値]ボタンをクリックしてください。[データサイズ]領域でデータ長も指定してください。

文字列を設定する場合

[データ形式]グループの[文字列]ボタンをクリックしてください。[コード]領域で文字コードも指定してください。

表示データ長が 2 バイトの場合は、2 バイト分の文字が指定できます。

文字列を指定するには、[メモリ設定]ダイアログボックスを使用してください。

[メモリ設定]ダイアログボックスは、ポップアップメニューから[設定]を選択してください。

17.3.5 任意のアドレス領域のデータを別領域にコピーする

[メモリ]ウィンドウ上でコピー元アドレス領域をドラッグして選択してください。ポップアップメニューから[コピー]を選択して[メモリコピー]ダイアログボックスを開きます。

[コピー先アドレス]領域にコピー先の開始アドレスを入力してください。

[ベリファイ]チェックボックスでメモリベリファイを指定します。ベリファイ機能のサポートは、デバッギングプラットフォームに依存します。

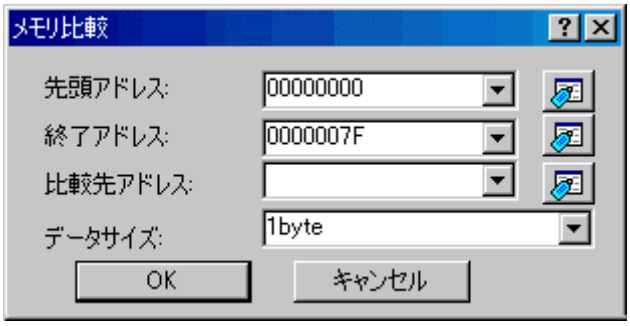
コピー元のアドレス領域をドラッグしていない場合は、コピー元開始アドレス/終了アドレスを入力する必要があります。終了アドレスには先頭に'+'記号をつけることができ、この記号を入力すると開始アドレス＋入力した値が終了アドレスになります。

ドラッグアンドドロップ

ドロップデータの種類	動作
[メモリ]ウィンドウ Data 領域の選択範囲	選択範囲のデータ内容を、ドロップ位置からの領域にコピーします。

17.3.6 任意の 2 つのアドレス領域を比較する

[メモリ]ウィンドウ上で比較元アドレス領域をドラッグして選択してください。ポップアップメニューから[比較]を選択して[メモリ比較]ダイアログボックスを開きます。



[比較先アドレス]領域に比較先アドレス、[データサイズ]領域に比較フォーマットを指定してください。

比較元のアドレス領域をドラッグしていない場合は、比較元開始アドレス/終了アドレスを入力する必要があります。終了アドレスには先頭に'+'記号をつけることができ、この記号を入力すると開始アドレス＋入力した値が終了アドレスになります。

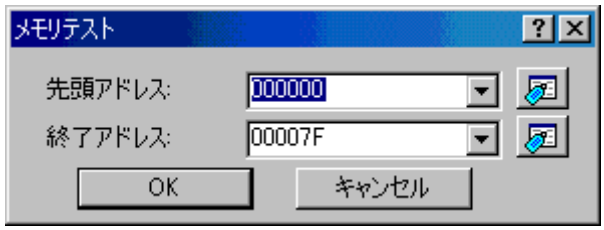
不一致個所があった場合は、そのアドレスをメッセージボックスに表示します。

比較したメモリの内容が一致した場合は、「比較成功」メッセージを表示します。

本機能のサポートは、デバッギングプラットフォームに依存します。

17.3.7 任意のアドレス領域をテストする

[メモリ]ウィンドウ上でテストするアドレス領域をドラッグして選択してください。ポップアップメニューから[テスト]を選択して[メモリテスト]ダイアログボックスを開きます。



テストするアドレス領域をドラッグしていない場合は、テスト開始アドレス/終了アドレスを入力する必要があります。

終了アドレスには先頭に'+'記号をつけることができ、この記号を入力すると開始アドレス＋入力した値が終了アドレスになります。

本機能のサポートは、デバッギングプラットフォームに依存します。

注：

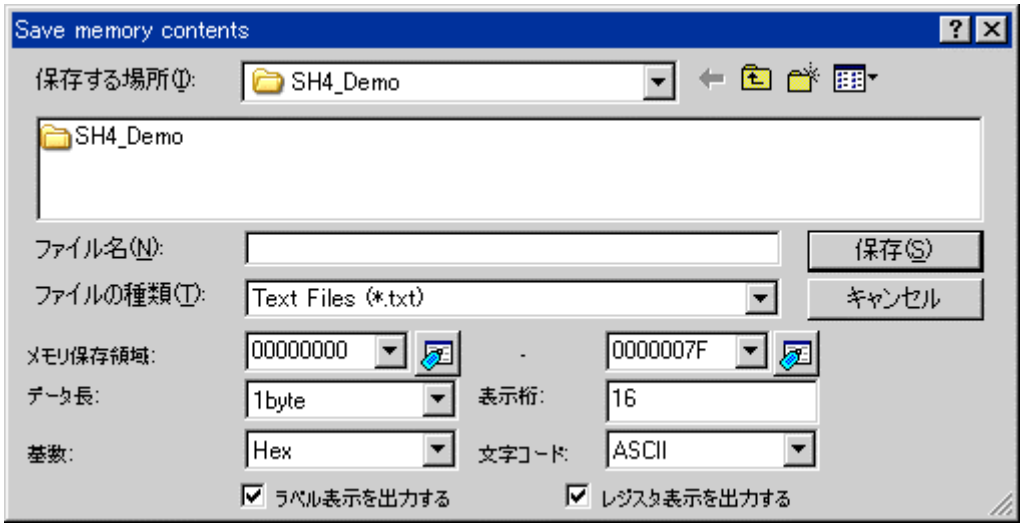
テストはデバッギングプラットフォームに依存しますが、すべての場合において、メモリの現在の内容を上書きします。**作成したプログラムまたはデータは、消去します。**

17.3.8 任意のアドレス領域をテキストファイルに保存する

[メモリ]ウィンドウ上でテキストファイルに保存するアドレス領域をドラッグして選択してください。ポップアップメニューから[テキストファイルに保存]を選択して[メモリ内容保存]ダイアログボックスを開きます。

[メモリ保存領域]領域で出力範囲、[データ長]でデータ長、[表示桁]領域で桁数、[基数]領域で基数、[文字コード]領域でコードを選択してください。[ラベル表示を出力する]領域でラベル表示領域の表示/表示なし、[レジスタ表示を出力する]領域でレジスタ表示領域を表示/表示なしを切り替えてください。

保存するアドレス領域をドラッグしていない場合は、出力範囲を入力する必要があります。



17.3.9 任意のアドレス領域のデータを検索する

[メモリ]ウィンドウ上で検索するアドレス領域をドラッグして選択してください。ポップアップメニューから[検索]を選択して[メモリ検索]ダイアログボックスを開きます。

[検索データ]領域で検索するデータ値を入力し、[検索データ形式]ドロップダウンリストで検索フォーマットを選択してください。[一致するデータを検索]チェックボックスでデータ一致／不一致、[後方検索]チェックボックスで検索方向を切り替えてください。

[検索データ形式]で"pattern"（パターン検索）を選択すると、最大 256 バイトのバイト列が検索できます。

検索するアドレス領域をドラッグしていない場合は、検索開始アドレス/終了アドレスを入力する必要があります。終了アドレスには先頭に'+'記号をつけることができ、この記号を入力すると開始アドレス＋入力した値が終了アドレスになります。

データが見つかると、見つかったデータを反転表示します。データを見つけることができなかった場合、[メモリ]ウィンドウの表示は以前と変わらず、データを見つけることができなかったことを知らせるメッセージボックスを表示します。

データが見つかった状態で、ポップアップメニューから[次を検索]を選択すると、次のアドレスから検索を続行します。

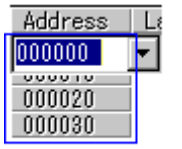
本機能のサポートは、デバッギングプラットフォームに依存します。

17.3.10　表示開始アドレスを指定する

スクロールバー、Page Up/Page Down キー、上下矢印キーで表示位置を変更してください。

表示アドレスを直接指定するには次の方法があります。

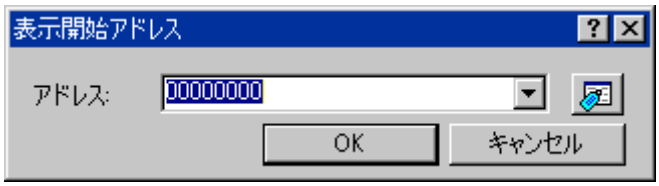
1. アドレス表示領域でインプレース編集します。



アドレス表示領域

2. また、以下のいずれかの方法で[表示開始アドレス]ダイアログボックスを開きます。

- アドレス表示領域をダブルクリックする。
- ポップアップメニューの[開始アドレス]を選択する。



[アドレス]領域に表示するアドレスを指定してください。

ドラッグアンドドロップ

操作	動作
[メモリ]ウィンドウの Address 領域のアドレスを選択して、別の[メモリ]ウィンドウの Address 領域にドロップ	ウィンドウの表示開始アドレスを、そのアドレスに変更します。
[逆アセンブリ]ウィンドウ Address 領域のアドレス（テキスト文字列）を選択して、[メモリ]ウィンドウの Address 領域にドロップ	ウィンドウの表示開始アドレスを、そのアドレスに変更します。
変数名（テキスト文字列）を選択して、[メモリ]ウィンドウの Address 領域にドロップ	ウィンドウの表示開始アドレスを、その変数のアドレスに変更します。

17.3.11　スクロール範囲を指定する

[メモリ]ウィンドウのポップアップメニューから[スクロール範囲]を選択してください。[スクロール範囲]ダイアログボックスが開きます。

表示するスクロール範囲を指定してください。

デフォルトのスクロール範囲は、0h～MCU の最大アドレスです。

17.3.12　指定したレジスタが示すアドレスから表示する

[メモリ]ウィンドウのポップアップメニュー[レジスタ]以下でレジスタを選択してください。

本機能のサポートは、デバッギングプラットフォームに依存します。

17.3.13　スタックポインタ位置を追従する

[メモリ]ウィンドウは、スタックポインタ位置に追従して表示アドレスを変更する機能を持っています(デフォルトは、スタックポインタ位置を追従しません)。スタックポインタ位置を追従するには、[メモリ]ウィンドウのポップアップメニューから[スタック追従]を選択してください。[追従スタックポインタ]ダイアログボックスが開きます。

ドロップダウンリストで追従するスタックポインタを選択してください。[メモリ]ウィンドウは、そのスタックポインタ位置を追従して表示アドレスを自動的に変更します。[追従スタックポインタ]ダイアログボックスで[Not Following]を選択するとスタックポインタ位置の追従を終了します。

本機能のサポートは、デバッギングプラットフォームに依存します。

17.3.14　ダウンロード時に表示するラベル位置を指定する

[メモリ]ウィンドウのポップアップメニューから[スタートアップシンボル]を選択してください。[スタートアップシンボルの設定]ダイアログボックスが開きます。

ドロップダウンリストに開始シンボルを指定してください。

17.3.15　メモリ内容を手動更新する

[メモリ]ウィンドウの内容を強制的にアップデートできます。

[メモリ]ウィンドウのポップアップメニューから[最新の情報に更新]を選択してください。

17.3.16　メモリ内容の更新を抑止する

ユーザプログラム実行停止時などに、自動的に[メモリ]ウィンドウ内容を更新しないようにできます。[メモリ]ウィンドウの表示がグレー表示に変わります。

[メモリ]ウィンドウのポップアップメニューから[表示固定]を選択してください。

17.3.17　表示データ長を指定する

[メモリ]ウィンドウのポップアップメニュー[データ長]以下で表示バイト長を指定してください。

以下のいずれかが選択できます。

- [1byte]: 1 バイト単位表示（デフォルト）
- [2byte]: 2 バイト単位表示
- [4byte]: 4 バイト単位表示
- [8byte]: 8 バイト単位表示

17.3.18　表示基数を指定する

[メモリ]ウィンドウのポップアップメニュー[基数]以下で表示基数を指定してください。

以下のいずれかが選択できます。

- [16 進数表示]: 16 進数で表示（デフォルト）
- [10 進数表示]: 10 進数で表示（符号なし）
- [符号付 10 進数表示]: 10 進数で表示（符号付き）
- [8 進数表示]: 8 進数で表示
- [2 進数表示]: 2 進数で表示

17.3.19　表示コードを指定する

[メモリ]ウィンドウのポップアップメニュー[表示コード]以下で表示コードを指定してください。

以下のいずれかが選択できます。

- [ASCII]: ASCII コードで表示（デフォルト）
- [SJIS]: SJIS コードで表示
- [JIS]: JIS コードで表示
- [UNICODE]: UNICODE で表示
- [EUC]: EUC コードで表示
- [Float]: 単精度浮動小数点数で表示
- [Double]: 倍精度浮動小数点数で表示
- [16 bit Fixed]: 16 ビット固定で表示
- [32 bit Fixed]: 32 ビット固定で表示
- [24 bit Accum]: 24 ビット累算で表示
- [40 bit Accum]: 40 ビット累算で表示

17.3.20　レイアウトを設定する

[メモリ]ウィンドウのポップアップメニュー[レイアウト]以下で表示するレイアウトを設定してください。以下を選択できます。

- [ラベル]: ラベル表示領域の表示/表示なしの切り替え
- [レジスタ]: レジスタ表示領域の表示/表示なしの切り替え
- [コード]: コード表示領域の表示/表示なしの切り替え

表示している場合はオプションにチェックマークが付いています。

ラベル表示領域、レジスタ表示領域、およびコード表示領域を表示しない場合:

Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
00000000	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000010	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

17.3.21　表示桁数を指定する

[メモリ]ウィンドウのポップアップメニューから[カラム]を選択すると、[データ表示桁]ダイアログボックスが開きます。

表示桁数を指定してください（1～256）。

17.3.22　カバレッジ計測結果の表示/表示なしを切り替える

[メモリ]ウィンドウは、カバレッジ計測表示がデフォルトで無効になっています。

有効にするには、[メモリ]ウィンドウのポップアップメニュー[カバレッジ]以下の[有効]を選択してください。

カバレッジ計測時、コード実行部分のデフォルトの色は、前面が黒色で背景が水色です。コード未実行部分のデフォルトの色は、前面が黒色で背景が灰色です。

カバレッジ計測時のコード実行部分および未実行部分の色は、[表示形式]ダイアログボックスでカスタマイズできます。詳細は、「17.3.27 既存の色をカスタマイズする」を参照してください。

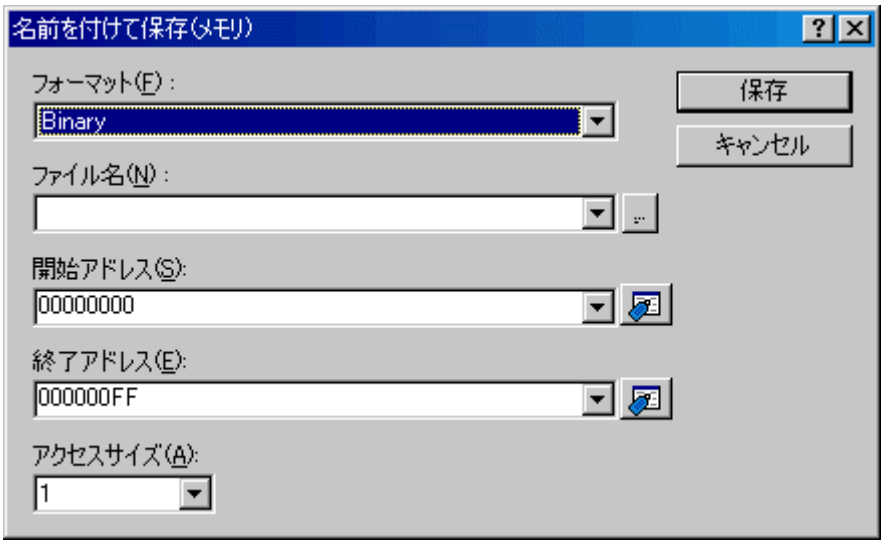
+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	ASCII
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....

本機能のサポートは、デバッギングプラットフォームに依存します。

17.3.23　任意のアドレス領域を保存する

[メモリ]ウィンドウ上で保存するアドレス領域をドラッグして選択してください。ポップアップメニューから[保存]を選択して[名前を付けて保存(メモリ)]ダイアログボックスを開きます。

（[デバッグ->メモリの保存]を選択して表示することもできます）



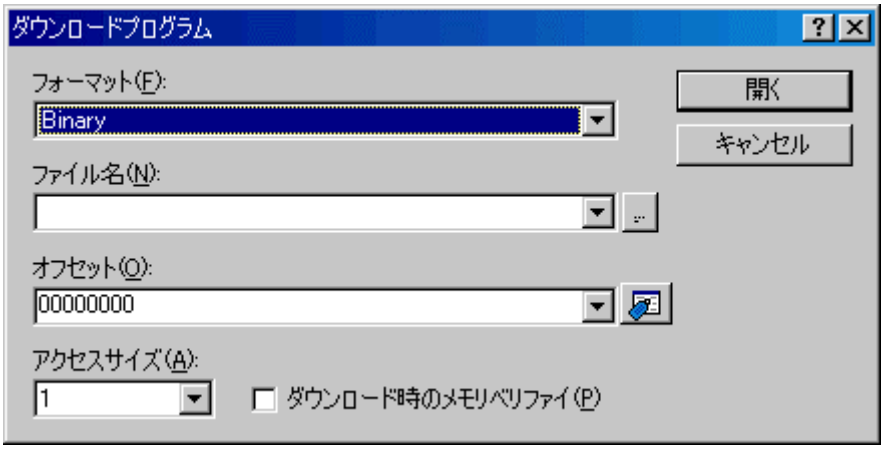
保存するアドレス領域をドラッグしていない場合は、保存開始アドレス/終了アドレスを入力する必要があります。終了アドレスには先頭に'+'記号をつけることができ、この記号を入力すると開始アドレス＋入力した値が終了アドレスになります。

[フォーマット]ドロップダウンリストにファイルフォーマット、[ファイル名]ドロップダウンリストにファイル名、[アクセスサイズ]ドロップダウンリストにアクセスサイズを入力します。[ファイル名]ドロップダウンリストには、メモリを保存するために使用した過去4つのファイル名を表示します。また[参照]ボタンをクリックすると、標準の[名前を付けて保存]ダイアログボックスを開きます。[アクセスサイズ]ドロップダウンリストから、保存時のアクセスサイズを選択できます。保存するメモリがリトルエンディアンの時は、アクセスサイズによってデータの並びが異なります。

ファイルの保存が完了すると、確認のメッセージボックスを表示することがあります。

17.3.24　メモリにファイルをロードする

デバッギングプラットフォームのメモリにファイルをロードできます。[メモリ]ウィンドウのポップアップメニューから[読み込み]を選択して、[ダウンロードプログラム]ダイアログボックスを開きます。



[フォーマット]ドロップダウンリストにファイルフォーマット、[ファイル名]ドロップダウンリストにファイル名、[オフセット]領域にオフセット、[アクセスサイズ]ドロップダウンリストにアクセスサイズ、[ダウンロード時のメモリベリファイ]チェックボックスでメモリベリファイを指定します。オフセットは、ロードアドレス値を変更するときはオフセット値、変更しないときは 0 を指定します。

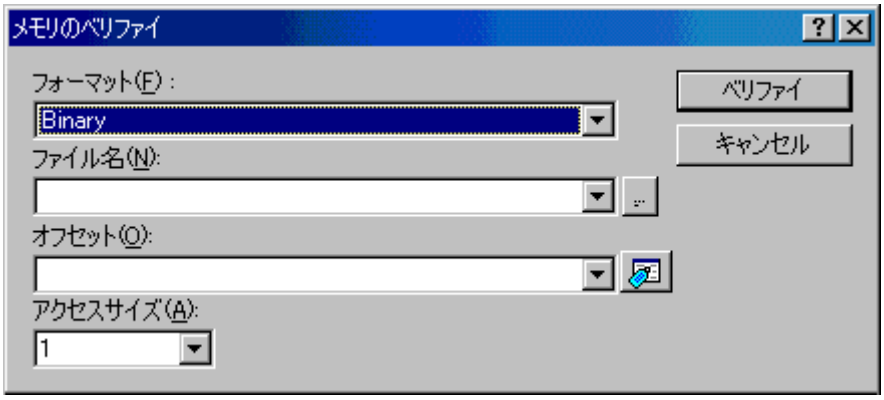
17.3.25　ウィンドウを分割表示する

[メモリ]ウィンドウを上下 2 分割で表示したいときは、ポップアップメニューから[分割]を選択後、分割バーを移動して分割します。

ポップアップメニューから[分割]を選択すると分割表示を解除します。

17.3.26　アドレス領域を検証する

メモリとファイル内容をベリファイします。[デバッグ->メモリのベリファイ]を選択して[メモリのベリファイ]ダイアログボックスを開きます。



[フォーマット]ドロップダウンリストにファイルフォーマット、[ファイル名]ドロップダウンリストにファイル名、[オフセット]領域にオフセット、[アクセスサイズ]ドロップダウンリストにアクセスサイズを指定します。

ベリファイが成功した場合は、「File verified OK」メッセージを表示します。

ベリファイの不一致個所があった場合は、「Verify failed」メッセージを表示します。

本機能のサポートは、デバッギングプラットフォームに依存します。

17.3.27　既存の色をカスタマイズする

メモリウィンドウはデータおよびコードを表示しているテキストの色およびフォントを他のウィンドウと同じように[表示形式]ダイアログボックスでカスタマイズできます（下表のカテゴリの”通常”）。

メモリウィンドウではこれに加えて、データおよびコード表示領域で、メモリアクセスおよびメモリの値が変更された部分の色を、[表示形式]ダイアログボックスでカスタマイズできます。

既存の色をカスタマイズするには

1. [基本設定->表示の形式]を選択し、[表示形式]ダイアログボックスを表示します。
2. [メモリ]アイテムをダイアログボックスの左側のツリーから選択し展開します。
3. 色を変更したいカテゴリを選択します。次の表に示すカテゴリが選択できます。

カテゴリ	カラータブの前面の色 （デフォルト）	カラータブの背景の色 （デフォルト）
通常	黒	白
アクセス履歴あり ***1** （カバレッジ計測：コード実行部分）	黒	水色
アクセス履歴なし ***1** （カバレッジ計測：コード未実行部分）	黒	灰色
カバレッジ計測領域外 ***1**	黒	白
メモリ領域外 ***1**	灰色	灰色
変更 （値が変更された部分）	赤	白

***1.** 機能のサポートは、デバッギングプラットフォームに依存します。

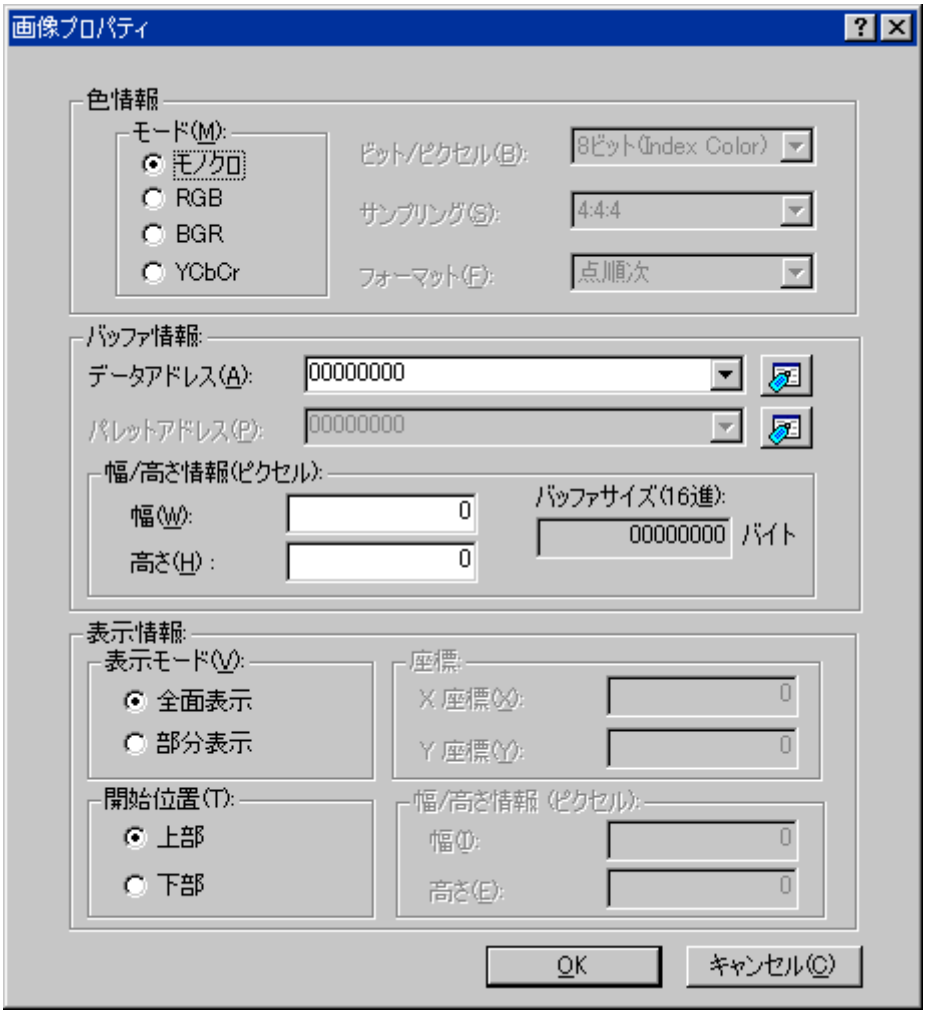
4. [カラー]タブの[前面]リストと[背景]リストの選択を変更します。
5. [OK]ボタンをクリックします。

17.4　メモリ内容を画像形式で表示する

[画像]ウィンドウを使用すると、メモリ内容を画像形式で表示できます。

17.4.1　画像ウィンドウを開く

[表示->グラフィック->画像]を選択するか、[画像]ツールバーボタン( )をクリックすると、[画像プロパティ]ダイアログボックスが開きます。



[画像プロパティ]ダイアログボックスでは[画像]ウィンドウの表示方法を指定します。

[色情報]　表示する画像のカラー情報を指定します。

- [モード]　フォーマットを指定します。
  - [モノクロ]　白黒で表現します。
  - [RGB]　R(赤)、G(緑)、B(青)で表現します。
  - [BGR]　B(青)、G(緑)、R(赤)で表現します。
  - [YCbCr]　Y(輝度)、Cb(青色の色差)、Cr(赤色の色差)で表現します。
- [ビット/ピクセル]　選択した[モード]によって、ビット/ピクセルを指定します(RGB/BGR 選択時有効)。
- [サンプリング]　サンプリングのフォーマットを指定します(YCbCr 選択時有効)。
  サンプリング例を以下に示します。

サンプリング比率	元のデータ	サンプリングデータ	画像(4*4 ピクセル)のデータ量
4:4:4	11,12,13,14 21,22,23,24 31,32,33,34 41,42,43,44	11,12,13,14 21,22,23,24 31,32,33,34 41,42,43,44	16 バイト
4:2:2		11,11,13,13 21,21,23,23 31,31,33,33 41,41,43,43	8 バイト
4:1:1		11,11,13,13 11,11,13,13 31,31,33,33 31,31,33,33	4 バイト
4:2:0		11,11,11,11 21,21,21,21 31,31,31,31 41,41,41,41	4 バイト

[フォーマット] 点順次/面順次を指定します(YCbCr 選択時有効)。

フォーマット	各データの順序
点順次（4:2:0 選択時無効）	Y,Cb,Cr,Y,Cb,Cr,Y,Cb,Cr,...,Y,Cb,Cr
面順次	Y,Y,Y,...,Y,Cb,Cb,Cb,...,Cb,Cr,Cr,Cr,...,Cr
面順次 2	Y,Y,Y,...,Y,Cb,Cr,Cb,Cr,Cb,Cr,...,Cb,Cr

[バッファ情報] データの格納場所、サイズ、パレットのアドレスを指定します。

[データアドレス] 表示する画像データのメモリ開始アドレスを指定します（16 進数表示）。

[パレットアドレス] カラーパレットデータのメモリ開始アドレスを指定します（16 進数表示）。(RGB/BGR の 8 ビット選択時有効)

[幅/高さ情報(ピクセル)] 画像の幅と高さを指定します。

[幅] 画像の幅を指定します（接頭辞省略時は 10 進数で入力、10 進数で表示）。

[高さ] 画像の高さを指定します（接頭辞省略時は 10 進数で入力、10 進数で表示）。

[バッファサイズ(16 進)] 幅と高さから画像のバッファサイズを表示します（16 進数表示）。

[表示情報] 画像全体中の表示部分の位置、サイズ、データ開始位置を指定します。（16 進数表示。RGB/BGR の 8 ビット選択時有効）

[表示モード] 画像の全体表示/部分表示を指定します。

[全面表示] 画像を全体表示します。

[部分表示] 画像を部分表示します。

[開始位置] データの表示開始位置を指定します。

[上部] 左上からデータを表示します。

[下部] 左下からデータを表示します。

[座標] 部分表示する画像の開始位置を指定します([部分表示]選択時有効）。

	[X 座標]	開始位置の X 座標を指定します（接頭辞省略時は 10 進数で入力、10 進数で表示）。
	[Y 座標]	開始位置の Y 座標を指定します（接頭辞省略時は 10 進数で入力、10 進数で表示）。
[幅/高さ情報(ピクセル)]	部分表示する画像の幅と高さを指定します。	
	[幅]	表示の幅を指定します（接頭辞省略時は 10 進数で入力、10 進数で表示）。
	[高さ]	表示の高さを指定します（接頭辞省略時は 10 進数で入力、10 進数で表示）。

[画像プロパティ]ダイアログボックスに設定後、[OK]ボタンをクリックすると[画像]ウィンドウが開きます。

[画像]ウィンドウ表示後もポップアップメニューの[プロパティ]を選択することで本ダイアログボックスを表示して表示内容を変更できます。

メモリの内容を画像で表示します。

- ウィンドウ内をダブルクリックするとマウスポインタの位置のピクセル情報を[ピクセル情報]ダイアログボックスに表示します。

ウィンドウのオプション

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

オプションメニュー項目		マクロの記録	機能
自動更新	更新しない	-	ウィンドウ内容を自動更新しません。
	停止	-	ユーザプログラム実行停止時にウィンドウ内容を更新します。
	リアルタイム	-	リアルタイム更新します。
更新		-	ウィンドウを更新します。
プロパティ		-	[画像プロパティ]ダイアログボックスを表示します。

17.4.2　ウィンドウを自動更新する

ポップアップメニューから[自動更新->更新しない]を選択すると、ウィンドウ内容を自動更新しません。

ポップアップメニューから[自動更新->停止]を選択すると、ユーザプログラム実行停止時にウィンドウ内容を更新します。

ポップアップメニューから[自動更新->リアルタイム]を選択すると、リアルタイム更新します。

17.4.3　ウィンドウを更新する

ポップアップメニューから[更新]を選択すると、直ちにウィンドウ内容を更新します。

17.4.4　ピクセル情報を表示する

ウィンドウ内をダブルクリックするとマウスポインタの位置のピクセル情報を[ピクセル情報]ダイアログボックスに表示します。

カーソル位置のピクセル情報を表示します。

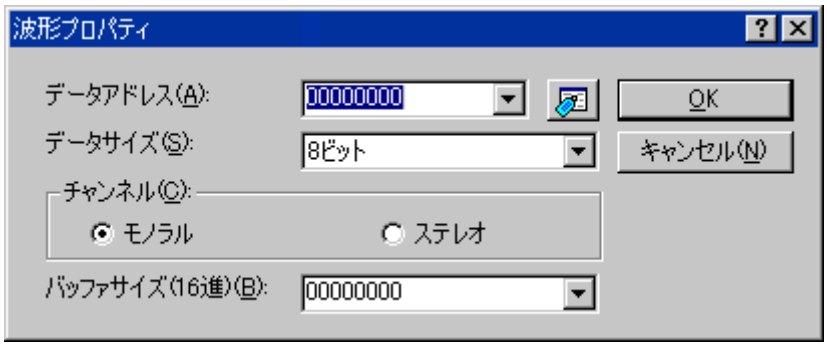
[カラーモード]	画像のフォーマットを表示します。	
[ピクセル]	カーソル位置のカラー情報を表示します（10 進数表示）。	
[位置]	カーソル位置を X 座標、Y 座標で表示します（10 進数表示）。	
	[X]	カーソル位置の X 座標を表示します。
	[Y]	カーソル位置の Y 座標を表示します。
[バッファサイズ]	バッファサイズを表示します（10 進数表示）。	
	[幅]	バッファの幅を表示します。
	[高さ]	バッファの高さを表示します。
[画像サイズ]	表示の幅と高さを表示します（10 進数表示）。	
	[幅]	幅を表示します。
	[高さ]	高さを表示します。

17.5　メモリ内容を波形形式で表示する

[波形]ウィンドウを使用すると、メモリ内容を波形形式で表示します。

17.5.1 波形ウィンドウを開く

[表示->グラフィック->波形]を選択するか、[波形]ツールバーボタン( )をクリックすると、[波形プロパティ]ダイアログボックスが開きます。



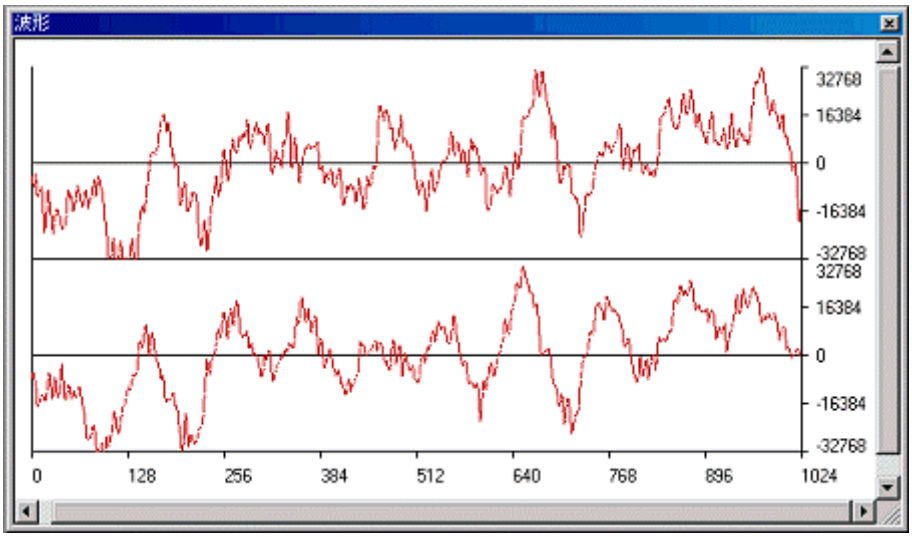
表示する波形形式を指定します。下記項目を指定できます。

[データアドレス]	データのメモリ開始アドレスを指定します（16 進数表示）。
[データサイズ]	8 ビット/16 ビットを指定します。
[チャンネル]	モノラル/ステレオを指定します。
[バッファサイズ(16 進)]	データのバッファサイズを指定します（16 進数表示）。

[波形プロパティ]ダイアログボックスに設定後、[OK]ボタンをクリックすると[波形]ウィンドウが開きます。

[波形]ウィンドウ表示後もポップアップメニューの[プロパティ]を選択することで本ダイアログボックスを表示して表示内容を変更できます。

メモリ内容を波形で表示します。横軸(X)にサンプリングデータ数、縦軸(Y)にサンプリング値を表示します。



- グラフ内でサンプリング情報を表示したい座標をダブルクリックすると、[サンプリング情報]ダイアログボックスが表示されます。

ウィンドウのオプション

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

オプションメニュー項目		マクロの記録	機能
自動更新	更新しない	-	ウィンドウ内容を自動更新しません。
	停止	-	ユーザプログラム実行停止時にウィンドウ内容を更新します。
	リアルタイム	-	リアルタイム更新します。
更新		-	ウィンドウを更新します。
伸張		-	拡大表示します。
圧縮		-	縮小表示します。
元に戻す		-	最初のサイズに戻します。
圧縮・伸張倍率	X2	-	拡大/縮小倍率を 2 倍に設定します。
	X4	-	拡大/縮小倍率を 4 倍に設定します。
	X8	-	拡大/縮小倍率を 8 倍に設定します。
スケール	128	-	横軸のサイズを 128 ピクセルに設定します。
	256	-	横軸のサイズを 256 ピクセルに設定します。
	512	-	横軸のサイズを 512 ピクセルに設定します。
カーソル削除		-	カーソルを非表示にします。
サンプリング情報		-	サンプリング情報を表示します。
プロパティ		-	[波形プロパティ]ダイアログボックスを表示します。

17.5.2 ウィンドウを自動更新する

ポップアップメニューから[自動更新->更新しない]を選択すると、ウィンドウ内容を自動更新しません。

ポップアップメニューから[自動更新->停止]を選択すると、ユーザプログラム実行停止時にウィンドウ内容を更新します。

ポップアップメニューから[自動更新->リアルタイム]を選択すると、リアルタイム更新します。

17.5.3 ウィンドウを更新する

ポップアップメニューから[更新]を選択すると、直ちにウィンドウ内容を更新します。

17.5.4 拡大表示する

ポップアップメニューから[伸張]を選択すると、横軸を拡大して表示します。

17.5.5 縮小表示する

ポップアップメニューから[圧縮]を選択すると、横軸を縮小して表示します。

17.5.6 最初のサイズに戻す

ポップアップメニューから[元に戻す]を選択すると、最初のサイズに戻して表示します。

17.5.7　拡大/縮小倍率を設定する

ポップアップメニューの[圧縮・伸張倍率]サブメニューで拡大/縮小倍率を 2、4、8 倍から選択します。

17.5.8　横軸のサイズを設定する

ポップアップメニューの[スケール]サブメニューで横軸のサイズを 128、256、512 ピクセルから選択します。

17.5.9　カーソルを非表示にする

ポップアップメニューの[カーソル削除]をチェックすると、カーソルを非表示にします。

17.5.10　サンプリング情報を表示する

グラフ内をクリックするとカーソル（緑色の縦線）が表示されます。カーソルは右矢印キーおよび左矢印キーで左右に移動できます。ウィンドウ内で右クリックし、ポップアップメニューから[サンプリング情報]を選択すると、[サンプリング情報]ダイアログボックスが表示されます。

また、グラフ内でサンプリング情報を表示したい座標をダブルクリックすると、[サンプリング情報]ダイアログボックスが表示されます。

[波形]ウィンドウのカーソル位置のサンプリング情報を表示します。下記情報を表示します。

[データサイズ]	8 ビット/16 ビットを表示します。	
[チャンネル]	データのチャンネルを表示します。	
[データ値]	[X]	カーソル位置の X 座標を表示します。
	[Y]	カーソル位置の Y 座標を表示します(ステレオ選択時は上下 2 つの Y 座標を表示します)。

17.6　I/O レジスタを見る

CPU および ROM/RAM と同様、マイクロコントローラには内蔵周辺モジュールがあります。デバイスによって周辺モジュールの数および型は異なりますが、代表的なモジュールとしては、DMA コントローラ、シリアルコミュニケーションインタフェース、A/D コンバータ、インテグレーテッドタイマユニット、バスステートコントローラおよびウォッチドッグタイマなどがあ

ります。マイクロコントローラのアドレス空間にマッピングしたアクセスレジスタは、内蔵周辺モジュールを制御します。

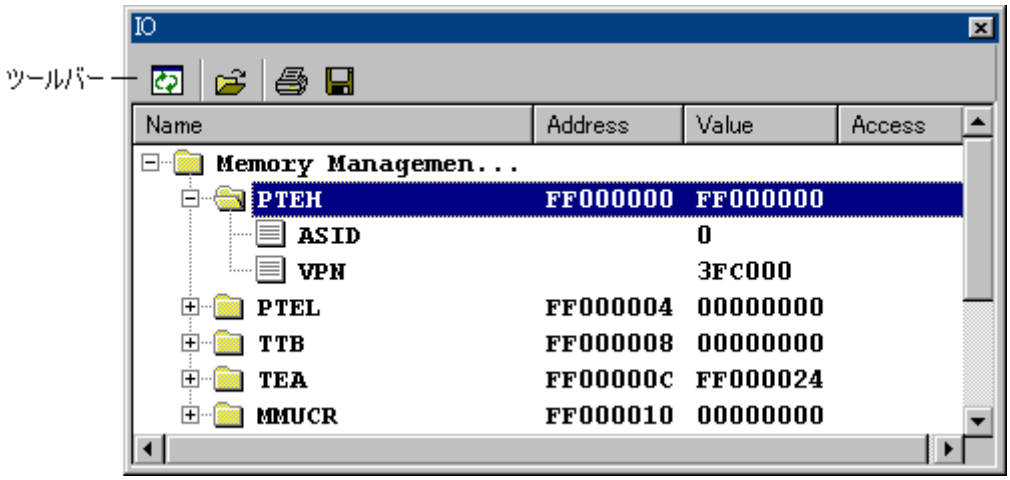
組込みマイクロコントローラアプリケーションでは、内蔵周辺レジスタを使用した設定が重要であるため、これらのレジスタの内容を見やすく表示しています。

[メモリ]ウィンドウは、連続したメモリアドレスのデータをバイト、ワード、ロングワード、単精度浮動小数点、倍精度浮動小数点、または ASCII 値として表示することができます。これに対し I/O レジスタは、非連続なメモリアドレスに異なるサイズでレジスタが割り付いているので、HEW はこれらのレジスタを簡単に確認したり設定したりすることができるように[IO]ウィンドウを提供します。

17.6.1 IO ウィンドウを開く

[IO]ウィンドウを開くには、[表示->CPU->IO]メニューオプションを選択するか、[IO の表示]ツールバーボタン( )をクリックします。内蔵周辺と一致するモジュールが I/O レジスタ情報を構成します。[IO]ウィンドウを最初に開くと、モジュール名の一覧のみを表示します。

ウィンドウの構成



- IO レジスタ表示を拡張します。
- IO レジスタ/ビットの値に更新があった場合、その値を赤色表示します。
- IO レジスタ/ビット表示行をダブルクリックすると、レジスタ値を変更するダイアログボックスが開きます。（「マクロの記録」）
- IO レジスタ/ビット表示行の Value 上をマウスでインプレースすると、レジスタ値を直接変更できます。（「マクロの記録」）

ウィンドウのオプション

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

本ウィンドウは、基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	マクロの記録	機能
最新の情報に更新		-	ウィンドウを更新します。
IO ファイルのロード		-	IO ウィンドウに表示する制御レジスタの情報を IO ファイルからロードします。
印刷		-	現在の表示内容を印刷します。
ファイルに保存		-	現在の表示内容をテキストファイルに保存します。
ツールバー表示	-	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ	-	-	ツールバーボタンをカスタマイズします。

17.6.2 I/O レジスタ表示を拡張する

I/O レジスタの名前、アドレス、および値を表示するには、モジュール名の左側にあるプラス記号(+)の拡張指示子をクリックするか、クリックまたはカーソルを使用することによってモジュール名を選択し、'→'キーを押します。モジュールの表示が拡張し、その周辺モジュールのそれぞれのレジスタおよびその名前、アドレス、および値を表示します（プラス記号がマイナス記号に変わります）。マイナス記号(-)の拡張指示子をクリックする（または'←'キーを押す）と、表示している I/O レジスタを閉じます。

ビットレベルで表示するには、同様に I/O レジスタを展開します。

注：

エミュレータデバッギングプラットフォームを使用している場合は、I/O レジスタからのリードがユーザプログラムの動作に影響することがあります。例えば、データレジスタのリードで割り込みをキャンセルできる場合があります。[IO]ウィンドウで I/O モジュールを拡張するとレジスタ値を表示するためにデータをリードします。I/O モジュールを閉じている場合は、データをリードしないためユーザプログラムの動作に影響しません。また、[メモリ]ウィンドウ(または、[逆アセンブリ]ウィンドウ)で I/O 領域を開くと同じ影響があることに注意してください。

17.6.3 ウィンドウを更新する

ポップアップメニューから[最新の情報に更新]を選択すると、直ちにウィンドウ内容を更新します。

17.6.4 I/O ファイルをロードする

IO ウィンドウに I/O ファイルを手動でロードすることができます。

I/O ファイルをロードするには

1. IO ウィンドウ内で右クリックしポップアップメニューを表示します。
2. [IO ファイルのロード]を選択します。[I/O ファイルを指定]ダイアログボックスが表示されます。
3. [I/O ファイル]フィールドにロードしたい I/O ファイルを指定します。[I/O ファイル]フィールドにプレースホルダを挿入するには、プレースホルダボタンをクリックしポップアップメニューからプレースホルダを選択します。ファイルを参照するには[参照]ボタンをクリックします。
4. [I/O ファイルのパス名を保存する]チェックボックスをオフにすると、セッションを保存するときに[I/O ファイル]フィールドのパス名を保存しません。次回このセッションを選択したとき、IO ウィンドウへのロード内容はデバッギングプラットフォームに依

存します。自動的に IO ファイルがロードされる場合は、このままこのチェックボックスをオフにしてください。CPU 選択に依存した I/O ファイルがデバッギングプラットフォームにより自動的にロードされています。
[I/O ファイルのパス名を保存する]チェックボックスをオンにすると、セッションを保存するときに[I/O ファイル]フィールドのパス名を保存します。次回このセッションを選択したときは前回ロードした I/O ファイルを IO ウィンドウにロードします。
このチェックボックスのデフォルトはオフです。

5. [OK]ボタンをクリックします。

I/O ファイルのフォーマットは「リファレンス 6. I/O ファイルフォーマット」を参照してください。

17.6.5 IO ウィンドウ表示内容を印刷する

現在 IO ウィンドウに表示している内容をテキストファイル形式で印刷することが出来ます。
ポップアップメニューから[印刷]を選択してください。

17.6.6 IO ウィンドウ表示内容を保存する

現在 IO ウィンドウに表示している内容をテキストファイルに保存することが出来ます。ポップアップメニューから[ファイルに保存]を選択してください。

17.6.7 I/O レジスタの内容を修正する

I/O レジスタの値を編集するには、ウィンドウに対して 16 進数を直接入力します。さらに複雑な式を入力するには、レジスタをダブルクリックするか'Enter'キーを押してレジスタの内容を修正するためのダイアログボックスを開きます。新しい数字または式を入力したら、[OK]ボタンをクリックするか'Enter'キーを押します。ダイアログボックスは閉じて新しい値をレジスタに書き込みます。

17.7 レジスタ内容を見る

アセンブリ言語レベルでデバッグを行う場合に、CPU の汎用レジスタの内容を簡単に見ることができます。これは、[レジスタ]ウィンドウを使用して行います。

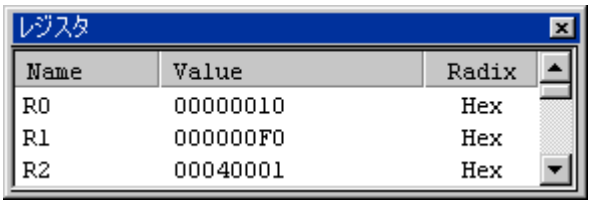
17.7.1 レジスタウィンドウを開く

レジスタウィンドウは、レジスタの内容やフラグの内容を表示するウィンドウです。ウィンドウからレジスタ/フラグの値を変更できます。

[レジスタ]ウィンドウを開くには、[レジスタ]ツールバーボタン( )をクリックするか、[表示->CPU->レジスタ]を選択します。[レジスタ]ウィンドウが開きレジスタおよびその値を表示します。

ウィンドウの構成

以下の図は、SH-4 用デバッガのレジスタウィンドウです。



- レジスタ表示行をダブルクリックすると、レジスタ値を変更するダイアログボックスが開きます。（「マクロの記録」）
- レジスタ表示行の Value 上をマウスでインプレースすると、レジスタ値を直接変更できます。（「マクロの記録」）
- フラグに対応したボタンをクリックすることにより、フラグの値を切り替えることができます。
- ポップアップメニューにより、各レジスタの表示基数変更、レジスタバンク切り替え等ができます(レジスタバンクの切り替えは、デバッギングプラットフォームがサポートしている場合に選択できます)。

ウィンドウのオプション

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

ポップアップメニュー項目		マクロの記録	機能
基数	16 進数表示	-	16 進数で表示します。
	10 進数表示	-	10 進数で表示します。
	8 進数表示	-	8 進数で表示します。
	2 進数表示	-	2 進数で表示します。
バンク 0 ***1**		-	レジスタバンク 0 のレジスタを表示します。
バンク 1 ***1**		-	レジスタバンク 1 のレジスタを表示します。
レイアウト	基数	-	基数表示領域の表示/表示なしを切り替えます。
	フラグ	-	フラグ表示部の表示/表示なしを切り替えます。
	設定	-	表示するレジスタを選択します。
編集		●	レジスタの内容を設定します。
分割		-	ウィンドウを分割表示します。
ファイルに保存		-	レジスタウィンドウの表示内容を保存します。

***1.** 機能のサポートは、デバッギングプラットフォームに依存します。

17.7.2 レジスタの表示基数を変更する

レジスタごとに表示基数が変更できます。

[レジスタ]ウィンドウのポップアップメニュー[基数]以下で表示基数を選択してください。

以下のいずれかが選択できます。

- [16 進数表示]: 16 進数で表示
- [10 進数表示]: 10 進数で表示

- [8 進数表示]: 8 進数で表示
- [2 進数表示]: 2 進数で表示

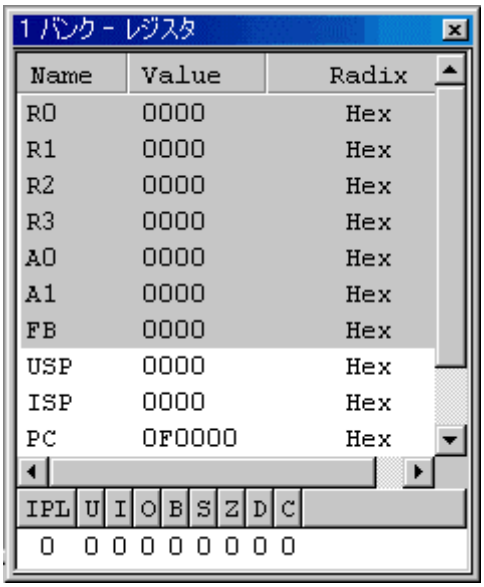
17.7.3 レジスタバンクを切り替える

レジスタウィンドウのオープン直後は、フラグの値に対応したバンクのレジスタ内容を表示します。

バンク 1 のレジスタ内容を参照する場合は、レジスタウィンドウをアクティブにした状態でメニュー[バンク 1]を選択してください。

バンク 1 固有のレジスタは、背景がグレー表示となります。

以下の図は、M16C 用デバッガのレジスタウィンドウです。



バンク 0 のレジスタ内容を参照する場合は、レジスタウィンドウをアクティブにした状態でメニュー[バンク 0]を選択してください(オプション[バンク 0]、[バンク 1]の操作では、フラグの値は変化しません)。

バンクの切り替えは、レジスタウィンドウのレジスタ表示部上で右クリックすると表示されるポップアップメニュー、または、フラグのフラグ切り替えからも可能です(フラグの値を変更した場合は、その値に応じてレジスタバンクが切り替わります)。

本機能のサポートは、デバッギングプラットフォームに依存します。

17.7.4 レイアウトを設定する

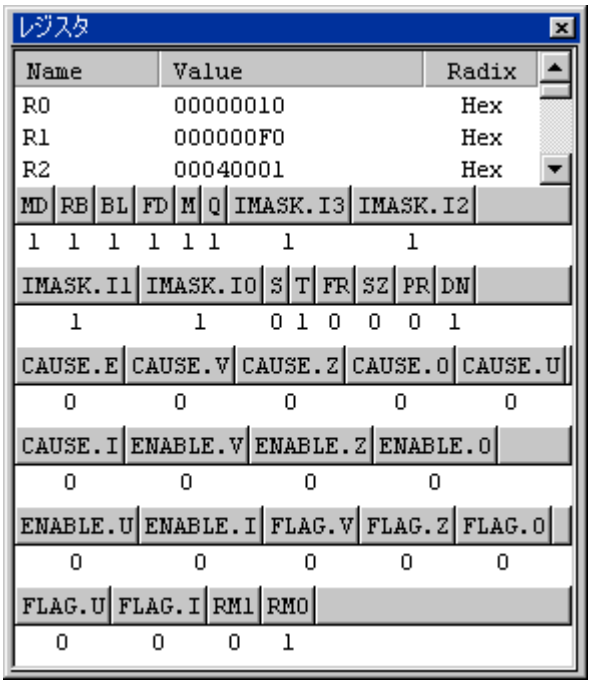
レイアウトを設定するには、[レジスタ]ウィンドウのポップアップメニュー[レイアウト]以下で選択してください。

下記から選択できます。

- [基数]: 基数表示領域の表示/表示なしの切り替え
- [フラグ]: フラグ表示部の表示/表示なしの切り替え

表示している場合はオプションにチェックマークが付いています。

基数表示領域、およびフラグ表示部を表示する場合:



17.7.5 表示するレジスタを選択する

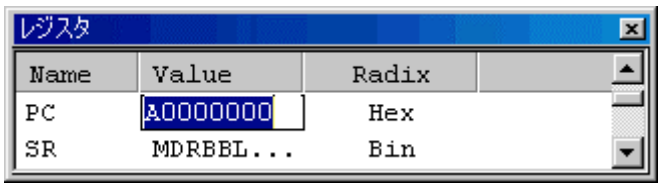
[レジスタ]ウィンドウのポップアップメニュー[レイアウト]以下で[設定]を選択してください。

以下の図は、SH-4 用デバッガの[設定]ダイアログボックスです。

17.7.6 レジスタの内容を設定する

レジスタの内容を設定するには、次の方法があります。

変更するレジスタの[Value]フィールドに値を直接入力する。



また、以下のいずれかの方法でレジスタ変更ダイアログボックスを開き、変更する値を入力する。

- 変更するレジスタ行をダブルクリックする。
- 変更するレジスタを選択して、ポップアップメニューの[編集]を選択する。

[値]フィールドには数字または C/C++の式を入力できます。

[基数] ドロップダウンリストから基数を選択できます。

[データ形式] ドロップダウンリストからオプションを選択して、レジスタの内容のすべて、マスクした領域、フローティングビットまたはフラグビットを修正できます(このリスト内容は、CPU のモデルおよび選択したレジスタによって異なります)。

新しい数字および式を入力したら[OK]ボタンをクリックするか'Enter'キーを押します。ダイアログボックスは閉じて、新しい値をレジスタに書き込みます。

17.7.7 フラグの値を設定する

フラグ表示している場合:

変更するフラグのボタンをクリックしてください。クリックごとにフラグの状態（0/1）が切り替わります。

複数ビットで構成されるフラグの場合は、変更値を入力するダイアログが開きます。

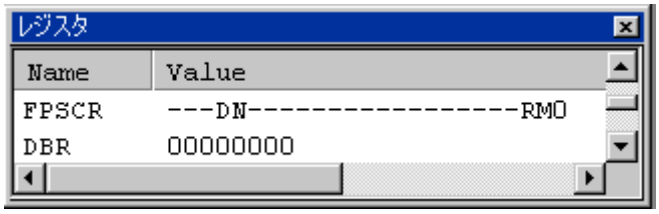
以下の図は、SH-4 用デバッガのレジスタウィンドウです。

フラグをレジスタ表示している場合:

フラグ行をダブルクリックしてください。ダイアログが開きます。

変更する値を入力してください。

以下の図は、SH-4 用デバッガのレジスタウィンドウです。（フラグ行（FPSCR））



17.7.8 ウィンドウを分割表示する

[レジスタ]ウィンドウを上下 2 分割で表示したいときは、ポップアップメニューから[分割]を選択後、分割バーを移動して分割します。

ポップアップメニューから[分割]を選択すると分割表示を解除します。

17.7.9 レジスタウィンドウの表示内容を保存する

[レジスタ]ウィンドウに表示しているデータをテキスト形式でファイルへ保存するには、[レジスタ]ウィンドウのポップアップメニューの[ファイルに保存]を選択します。名前を付けて保存ダイアログボックスを表示します。ファイル名を指定してください。

17.7.10 レジスタの内容を使用する

[逆アセンブリ]または[メモリ]ウィンドウのアドレス指定など、HEW の別のところで値を入力する場合、レジスタの中にある値を使用するためには、#PC または%PC などのように、レジスタ名の先頭に“#”記号（SuperH/H8/R8C（E7/E8）ファミリの場合）または“%”記号（M32C/M32R/M16C/R8C（E7/E8 以外）ファミリの場合）を付けてください。

本機能のサポートは、デバッギングプラットフォームに依存します。

17.7.11 値が変更された部分の色をカスタマイズする

レジスタウィンドウはレジスタおよびフラグの値を表示しているテキストの色およびフォントを他のウィンドウと同じように[表示形式]ダイアログボックスでカスタマイズできます。

レジスタウィンドウではこれに加えて、レジスタおよびフラグの値が変更された部分の色を、[表示形式]ダイアログボックスでカスタマイズできます。レジスタおよびフラグの値が変更された部分のデフォルトの色は、前面が赤色で背景が白色です。背景色は、[Text]カテゴリ選択時の背景色と同じ色になります。

値が変更された部分の色をカスタマイズするには

1. [基本設定->表示の形式]を選択し、[表示形式]ダイアログボックスを表示します。
2. [レジスタ]をダイアログボックスの左側のツリーから選択し展開します。
3. [変更]カテゴリを選択します。

4. [カラー]タブの[前面]リストの選択を変更します。
5. [OK]ボタンをクリックします。

17.8 ターゲットマイコンをリセットする

ターゲットマイコンのリセットを行うには、[CPU リセット]ツールバーボタン( )をクリックするか、[デバッグ]メニューから[CPU のリセット]を選択します。

ターゲットマイコンをリセットすることにより、内蔵 I/O レジスタの初期化、およびリセットベクタに設定されたアドレスを PC に設定します。

17.9 カーソル位置に PC 値を設定する

PC を現在のテキストカーソルの位置によって示されるアドレスに設定するには、[カーソル位置に PC 設定]ツールバーボタン( )をクリックするか、[デバッグ->カーソル位置に PC 値を設定]を選択します。

17.10 デバッギングプラットフォームを接続/接続解除する

デバッギングプラットフォームを接続するには

1. [接続]ツールバーボタン( )をクリックするか、[デバッグ]メニューから[接続]を選択します。

デバッギングプラットフォームを接続解除するには

1. [接続解除]ツールバーボタン( )をクリックするか、[デバッグ]メニューから[接続解除]を選択します。

機能のサポートは、デバッギングプラットフォームに依存します。

17.11 デバッギングプラットフォームを初期化する

[デバッグ->初期化]を選択します。

現在オープンしているウィンドウをクローズし、デバッギングプラットフォームとの接続を解除します。そして、デバッギングプラットフォームの再接続をします。

17.12 プログラムを実行する

この節では、作成したプログラムコードの実行方法について説明します。ここでは、プログラムを連続して実行させたり、シングルステップ実行を行ったり、同時に複数の命令を実行させたりします。

17.12.1 実行を継続する

作成したプログラムが停止すると、HEW は CPU の現在のプログラムカウンタ（PC）アドレス値に対応する[エディタ]および[逆アセンブリ]ウィンドウの行の左余白に黄色の矢印を表示します。ステップ実行を行った場合、または実行を続けた場合、この命令を次に実行します。

現在の PC アドレスから実行を継続するには、[実行]ツールバーボタン( )をクリックするか、[デバッグ->実行]を選択します。

前回停止時と異なるアドレスから実行を継続するには、下記の方法で PC を変更後に[実行]ツールバーボタン( )をクリックするか、[デバッグ->実行]を選択します。

- [レジスタ]ウィンドウ上で変更する。
- エディタウィンドウまたは逆アセンブリウィンドウ上で、テキストカーソル(マウスカーソルではありません)を変更したい行に移動してポップアップメニューから[カーソル位置に PC 値を設定]を選択する。

17.12.2 リセットから実行を開始する

ターゲットマイコンをリセットしてリセットベクタアドレスからプログラムを実行させるには、[デバッグ->リセット後実行]を選択するか、[リセット後実行]ツールバーボタン( ) をクリックします。

プログラムは、ブレークポイントにヒットするまで、またはブレーク条件が成立するまで実行を続けます。プログラムの実行はいつでも手動で停止できます。その方法としては、[デバッグ->プログラムの停止]を選択するか[停止]ツールバーボタン( ) をクリックします。

注：

プログラムはリセットベクタ位置に格納したアドレスから実行を開始します。したがって、この位置に自分のスタートアップコードのアドレスを含んでいることを確認することが重要です。

17.12.3 カーソルまで実行する

アプリケーションを実行している途中で、シングルステップ実行を複数回行うだけの比較的小さいセクションコードのみを実行したいと考える場合があります。この場合特定のポイントまで実行できると便利です。これは、[カーソル位置まで実行]機能を使用して行うことができます。

[カーソル位置まで実行]を使用するには

1. エディタウィンドウまたは逆アセンブリウィンドウが開いていて、プログラムを停止するアドレスを表示していることを確認します。
2. プログラムを停止するアドレスの行にテキストカーソルを置いてください。
3. ポップアップメニューから[カーソル位置まで実行]を選択します。

デバッギングプラットフォームは作成したコードを現在の PC 値から実行し、カーソル位置が示すアドレスまで実行します。

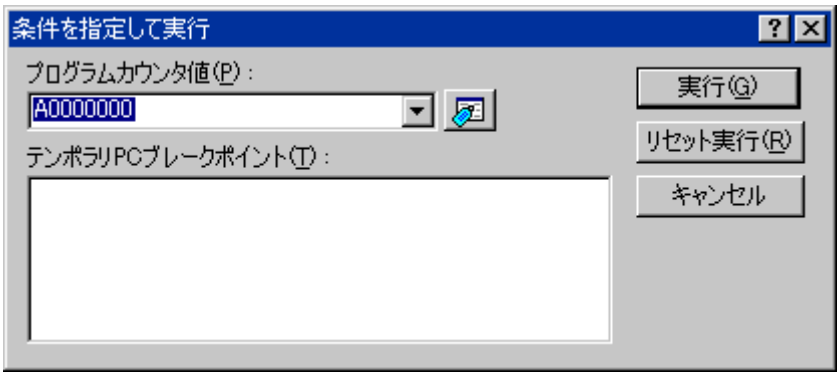
注：

- 作成したプログラムがこのアドレスのコードを実行しない場合、プログラムは停止しません。その場合、コードの実行を中止するには、'Esc キー'を押すか、[デバッグ->プログラムの停止]を選択するか、[停止]ツールバーボタン( )をクリックします。

- [カーソル位置まで実行]機能は、PC ブレークポイント機能を利用しています。そのため、すでに PC ブレークポイントが最大数設定してある場合は、本機能は利用できません。

17.12.4 条件を指定して実行する

[条件を指定して実行]ダイアログボックスを利用すると、任意のアドレスから命令を実行できます。[条件を指定して実行]ダイアログボックスを開くには、[デバッグ->条件を指定して実行]を選択します。



[条件を指定して実行]ダイアログボックスでは命令実行の条件を設定します。

[プログラムカウンタ値]	実行を開始する命令アドレスを設定します。初期値は現在の PC 値となります。
[テンポラリ PC ブレークポイント]	一時的な PC ブレークポイントを設定します。ダイアログボックスによる命令実行が停止すると、このブレークポイントは解除されます。

[実行]ボタンをクリックすると、設定した内容に従って命令を実行します。

[リセット実行]ボタンをクリックすると、リセットベクタから命令を実行します。

[キャンセル]ボタンをクリックすると、命令を実行しないで、ダイアログボックスを閉じます。

注：

[テンポラリ PC ブレークポイント]は PC ブレークポイント機能を利用しています。既に PC ブレークポイントを最大数利用している場合は本機能を利用できません。

17.12.5 リセット時に main 関数の先頭までプログラムを実行する

HEW は、リセット時にソフトウェアブレークポイントを設定せずに main 関数の先頭までプログラムを実行する機能があります。

この機能は main 関数に一時的にソフトウェアブレークポイントを設定します。プログラムの実行停止後にソフトウェアブレークポイントを解除します。

リセット時に main 関数の先頭までプログラムを実行するには

1. [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を表示します。
2. [デバッグ]タブを選択します。
3. [リセット時に main 関数の先頭まで実行]チェックボックスをオンにします。デフォルトはオフです。
4. ターゲットマイコンをリセットします。次のいずれかを行ってください。

- [CPU リセット]ツールバーボタン( )をクリックする。
- [デバッグ->CPU のリセット]を選択する。

注：

本機能はソフトウェアブレークポイント機能を利用しています。そのため、すでにソフトウェアブレークポイントが最大数設定してある場合は利用できません。

17.12.6 シングルステップ

作成したコードをデバッグするために、一度に 1 行だけまたは 1 つの命令だけステップ実行して、この命令がシステムにどのように影響するかを確認したい場合、ソース表示では、ソースライン 1 行だけをステップ実行します。逆アセンブリ表示では、アセンブリ言語命令単位にステップ実行します。命令が他の関数またはサブルーチンを呼び出す場合、オプションでその関数にステップインまたはステップオーバできます。命令が呼び出しを実行しない場合は、いずれのオプションでも、その命令を実行し、次の命令で停止します。

関数にステップイン実行を選択した場合は、関数の呼び出しを実行しその関数の最初の行または命令で停止します。

関数にステップイン実行するには

- [ステップイン]ツールバーボタン( )をクリックする。
- [デバッグ->ステップイン]を選択する。

関数をステップオーバ実行を選択した場合は、呼び出しおよびその関数(およびその関数が実行するすべての関数呼び出し)を実行して、呼び出し元の関数の次の行または命令で停止します。

関数をステップオーバ実行するには

- [ステップオーバ]ツールバーボタン( )をクリックする。
- [デバッグ->ステップオーバ]を選択する。

関数内の確認したい命令の実行が終了した場合や、誤って関数にステップイン実行した場合に、ステップアウト機能を使用すると関数内の残りのコードをすべてステップイン実行することなく呼び出し元の関数に戻ることができます。

関数からステップアウト実行するには

- [ステップアウト]ツールバーボタン( )をクリックする。
- [デバッグ->ステップアウト]を選択する。

ステップモードを選択するには

[デバッグ->ステップモード]から次のサブメニューを選択します。

サブメニュー	機能
[自動] （デフォルト）	エディタウィンドウがアクティブの場合はソースライン 1 行だけをステップ実行します。[逆アセンブリ]ウィンドウがアクティブの場合はアセンブリ言語命令単位にステップ実行します。
[アセンブリ]	アセンブリ言語命令単位にステップ実行します。
[ソース]	ソースライン 1 行だけをステップ実行します。

HEW エディタのソース表示でステップインしている際、デバッグ情報がないライブラリ関数（printf 等）はソース表示のままステップしたい場合があります。デフォルトでは[逆アセンブリ]ウィンドウへジャンプし、アセンブリ言語命令単位にステップします。

HEW エディタのソース表示でデバッグ情報がないアドレスにステップインした場合、アセンブリ言語命令単位にステップ実行しないようにできます。

デバッグ情報のないアドレスへステップインしないようにするには

1. [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を表示します。
2. [デバッグ]タブを選択します。
3. [デバッグ情報のないアドレスにステップインしない]チェックボックスをオンにします。チェックボックスのデフォルトはオフです。

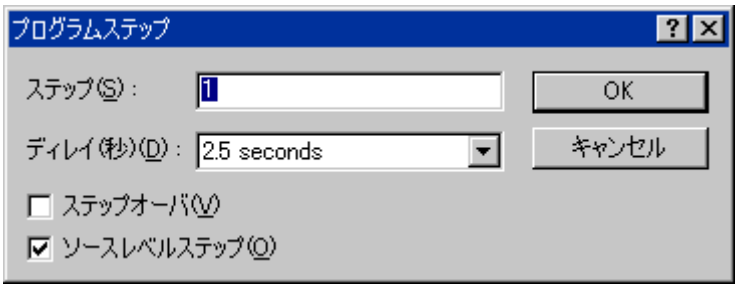
注：

本機能のサポートは、デバッギングプラットフォームに依存します。

17.12.7 複数のステップ

[プログラムステップ]ダイアログボックスを使用することにより、一度に複数のステップ実行ができます。このダイアログボックスでは、ステップ間の時間差を選択し、ステップ実行を自動的に行うよう設定できます。このダイアログボックスは、[デバッグ->ステップ]を選択して開きます。

[プログラムステップ]ダイアログボックスを表示します。



[ステップ]	連続実行するステップ数を設定します。デフォルトの値は、1 です。
[ディレイ（秒）]	コードを自動的にステップ実行するときのステップ間の遅延を選択します。デフォルトの値は、2.5 seconds です。 [No Refresh]（各ウィンドウの更新をしない）または 0～3 秒まで、0.5 秒単位で選択できます。

[ステップオーバ]	チェックすると関数呼び出しをステップオーバします。チェックボックスのデフォルトはオフです。
[ソースレベルステップ]	チェックするとソースレベルでステップします。チェックボックスのデフォルトはオンです。

[OK]ボタンをクリックするか、'Enter'キーを押してステップ実行を開始します。

17.13 プログラムを停止する

この節では、作成したプログラムの実行を停止する方法を説明します。停止手段として、[停止]ツールバーボタン( )を使用して停止する方法、および作成したコードの特定の場所にブレークポイントを設定することによって停止する方法について説明します。

17.13.1 停止ツールバーボタンによる停止

作成したプログラムが実行中の場合、[停止]ツールバーボタン( )を使用できます。しかし、プログラムが停止している場合は、[停止]ツールバーボタンはグレー表示( )となり使用できません。

[停止]ツールバーボタン( )をクリックするか、[デバッグ->プログラムの停止]を選択することによりプログラムが停止します。

前回のブレークの要因はアウトプットウィンドウの[Debug]タブで見ることができます。

17.13.2 標準のブレークポイント（PC ブレークポイント）

作成したプログラムをデバッグする場合、PC ブレークポイントにより指定した行または命令でプログラムの実行を停止させることができます。標準的な PC ブレークポイントを設定したり解除したりする方法を以下に示します。

エディタウィンドウ上で PC ブレークポイントを設定するには

1. PC ブレークポイントを設定する位置のエディタウィンドウまたは逆アセンブリウィンドウが開いていることを確認します。
2. プログラムを停止したい行でポップアップメニューの[ブレークポイントの挿入/削除]を選択するか、"F9"キーを押します。
3. 左余白に赤丸( )を表示します。これは、PC ブレークポイントを設定したことを示します。
4. ポップアップメニューの[ブレークポイントの有効化/無効化]を選択すると、現在設定しているブレークポイントの有効／無効の切り替えができます。

作成したプログラムを実行して PC ブレークポイントを設定したアドレスに達すると、"PC Breakpoint"というメッセージをアウトプットウィンドウの[Debug] タブに表示し、実行を中断し、エディタウィンドウまたは逆アセンブリウィンドウを更新し、停止位置を矢印で表示します。

注：

ブレーク発生時には、PC ブレークポイントを設定した行または命令を実行する直前で停止します。その PC ブレークポイントで停止した後に Go または Step を選択した場合、矢印で表示し

た行から実行します。また、複数ターゲットデバッグ時には一方のみ停止させるか両方停止させるかを指定できます。詳細は「17.17 複数デバッギングプラットフォームを同期動作させる」を参照してください。

[ブレークポイント]ダイアログボックスを使用して PC ブレークポイントの設定を編集するには

PC ブレークポイントの設定を編集するには、[編集->ソースブレークポイント]メニューを選択し、[ブレークポイント]ダイアログボックスを表示します。

- このダイアログボックスに現在設定している PC ブレークポイントを表示します。
- [コードの編集]ボタンにより PC ブレークポイントが存在するソースを見ることができます。
- [削除]、[すべて削除]ボタンにより、PC ブレークポイント 1 つまたは全てを削除できます。
- 各ブレークポイントのチェックボックスにより有効／無効の切り替えができます。

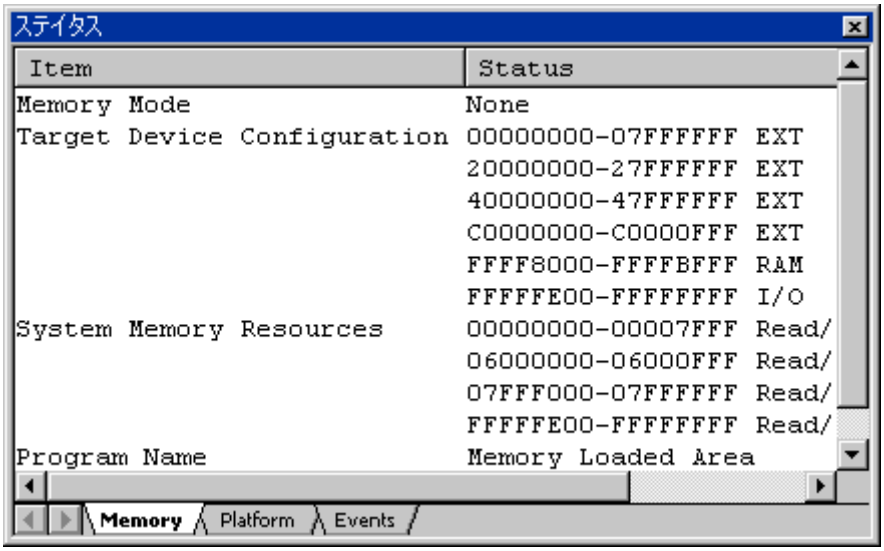
PC ブレークポイントを切り替えるには

PC ブレークポイントを設定したり、解除したりするには、PC ブレークポイントを設定している行の BP カラムをダブルクリックするか、行にカーソルを置き、F9 キーを使用します。その行で使用可能な標準のブレークポイントが有る場合、色のついた丸印(●)を左余白に表示します。

17.14 現在の状態を表示する

デバッギングプラットフォームの現在の状態を知るには[ステイタス]ウィンドウを使用します。[ステイタス]ウィンドウを開くには、[表示->CPU->ステイタス]を選択するか、[ステイタスの表示]ツールバーボタン( )をクリックします。

ウィンドウの構成



[ステイタス]ウィンドウには、3 枚のシートがあります。

・[Memory]シート

メモリマッピングおよび現在ロードしたオブジェクトファイルが使用するメモリエリアなど、現在のメモリステータスに関する情報を含んでいます。

・[Platform]シート

CPU 種別および動作モードなど、デバッギングプラットフォームのステータス情報、実行状態および実行統計情報を含んでいます。

・[Events]シート

リソース情報およびブレークポイント等のイベント情報に関する情報を含んでいます。

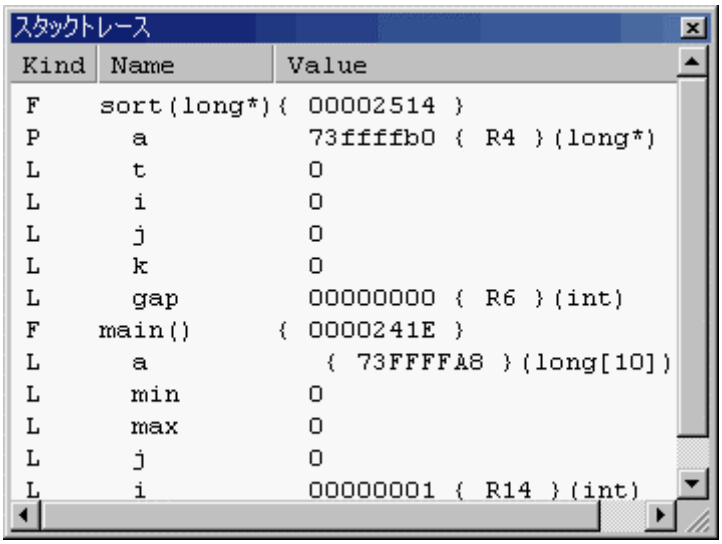
17.15 関数呼び出し履歴を見る

関数呼び出し履歴を表示する場合は、[スタックトレース]ウィンドウを使用します。

17.15.1 スタックトレースウィンドウを開く

[スタックトレース]ウィンドウを開くには、[表示->コード->スタックトレース]を選択するか、[スタックトレース]ツールバーボタン( )をクリックします。

ウィンドウ構成



表示する項目は以下の通りです。

[Kind] 該当シンボルのシンボル種別を示します。
F： 関数
P： 関数パラメータ ***1**
L： ローカル変数 ***1**

[Name] シンボル名を示します。

[Value] シンボルの値、アドレス、型を示します。

***1.** 機能のサポートは、デバッギングプラットフォームに依存します。

ウィンドウのオプション

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

ポップアップメニュー項目	マクロの記録	機能
ソースファイル表示	-	選択した関数に該当するソースコードを表示します。
表示設定	-	[スタックトレース]ウィンドウの表示形式を設定します。
コピー	-	選択部分をコピーし Windows®クリップボードに貼り付けます。

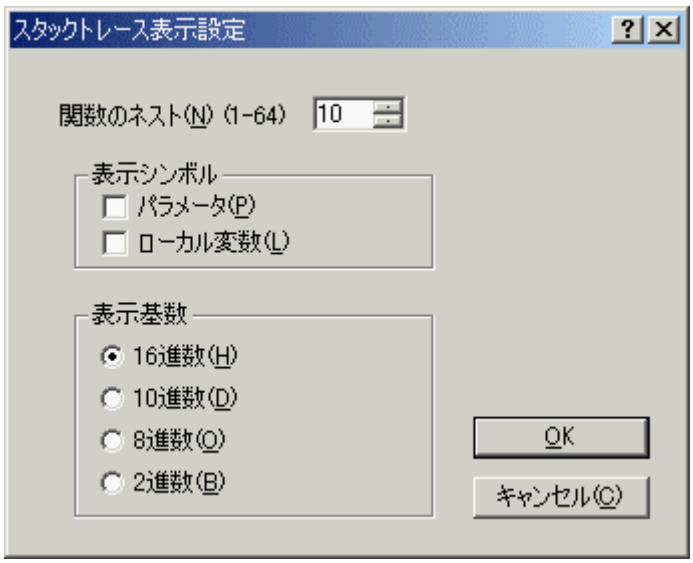
17.15.2 ソースコードを表示する

関数を選択した状態で、ポップアップメニューから[ソースファイル表示]を選択すると、エディタウィンドウを開いて選択した関数に該当するソースコードを表示します。

17.15.3 表示形式を設定する

ポップアップメニューから[表示設定]を選択すると、[スタックトレース表示設定]ダイアログボックスを表示します。

[スタックトレース]ウィンドウの表示形式を設定します。



[関数のネスト] [スタックトレース]ウィンドウに表示する関数コールネスト数を指定します。

[表示シンボル]***1** 関数以外に表示するシンボルを指定します。

[表示基数]***1** [スタックトレース]ウィンドウの表示基数を指定します。

***1** : 機能のサポートは、デバッギングプラットフォームに依存します。

17.16 外部デバッガを使う

HEW は外部デバッガツールを起動できます。もし他のデバッガを使用する場合は、[ツール]メニューに追加する必要があります。

[カスタマイズ]ダイアログボックスの[デバッガ]タブに外部デバッガに関連する情報を設定します。一部のターゲットが新しい環境でサポートされていない場合は、古いバージョンのデバッガを使用できます。[基本設定->カスタマイズ]を選択してダイアログボックスを表示し、[デバッガ]タブを選んでください。

最初に、[外部デバッガの選択] ドロップダウンリストで使用する外部デバッガを選択します。

外部デバッガを使用する場合、そのデバッガ固有の情報を構築する必要があります。

[Hitachi Debugger Interface (version 4.x or greater)]	日立デバッギングインタフェースを HEW と統合して構築する
[Renesas PD debugger]	PD デバッガを HEW と統合して構築する
[Other external debugger]	日立デバッギングインタフェースまたは PD デバッガ以外の外部デバッガを HEW と統合して構築する
[Non selected]	外部デバッガを使用しない

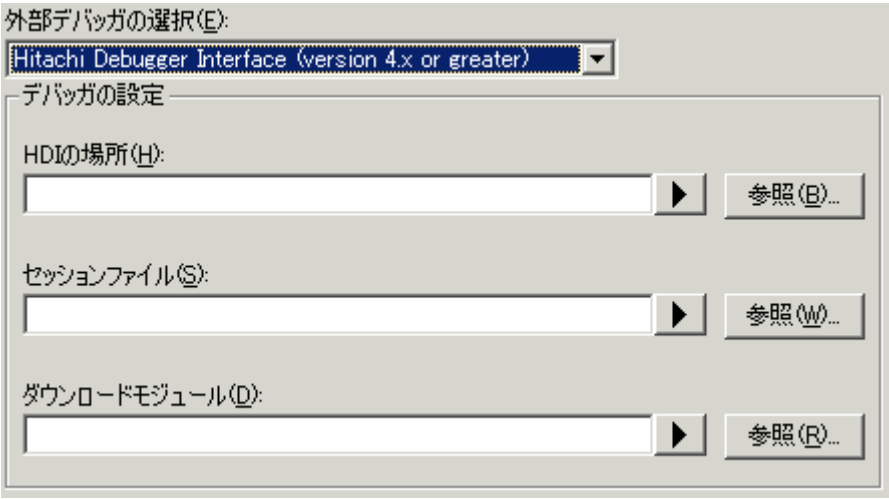
外部デバッガを起動するには、[外部デバッガの実行]ツールバーボタン( )をクリックしてください。

ビルド後にダウンロードモジュールが変更された場合、HEW から外部デバッガに切り替わり、すぐデバッグできるようになります。

外部デバッガを使用しているときにエディタウィンドウのどれかをダブルクリックすると HEW に切り替わり、ダブルクリックした行のソースファイルを開いた状態になります。[Other external debugger]を選択した場合は、この操作を行うことはできません。

17.16.1 日立デバッギングインタフェースを HEW と統合して構築する

日立デバッギングインタフェースを HEW と統合し、HEW の外部デバッガのオプションで起動できるよう設定するために必要な情報を以下に示します。



日立デバッギングインタフェースを HEW と統合して構築するには

1. 日立デバッギングインタフェースの場所を指定します。日立デバッギングインタフェース 4.0 以降のバージョンでない場合この動作は保証されません。
2. セッションファイルの場所を指定します。起動時にどのセッションをロードするか日立デバッギングインタフェースに伝えます。
3. ダウンロードモジュールの場所を指定します。

17.16.2 PD デバッガを HEW と統合して構築する

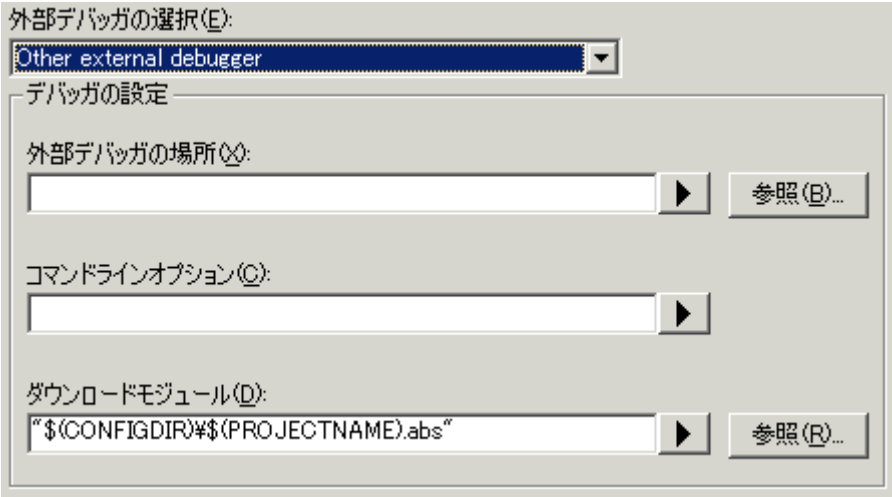
PD デバッガを HEW と統合し、HEW の外部デバッガのオプションで起動できるよう設定するために必要な情報を以下に示します。

PD デバッガを HEW と統合して構築するには

1. PD デバッガの場所を指定します。
2. PD Profile ファイルの場所を指定します。起動時にどのセッションをロードするか PD デバッガに伝えます。
3. コマンドラインオプションです。PD デバッガの動作を変更するオプションを指定できます。
4. ダウンロードモジュールの場所を指定します。

17.16.3 外部デバッガを HEW と統合して構築する

日立デバッギングインタフェースまたは PD デバッガ以外の外部デバッガを HEW と統合し、HEW の外部デバッガのオプションで起動できるよう設定するために必要な情報を以下に示します。



外部デバッガを HEW と統合して構築するには

1. 外部デバッガの場所を指定します。
2. コマンドラインオプションです。外部デバッガの動作を変更するオプションを指定できます。
3. ダウンロードモジュールの場所を指定します。

17.17 複数デバッギングプラットフォームを同期動作させる

HEW では複数のデバッギングプラットフォームを同時に操作できます。これを実現するために 2 通りの方法を提供しています。概要を以下に示します。HEW アプリケーションでの同期は HEW2.x から利用可能な方法です。HEW デバッガターゲットの同期は HEW3.x の複数ターゲットデバッグに適した新しい方法です。

17.17.1 HEW アプリケーションでの同期

複数デバッギングプラットフォームの同期動作を他の HEW から HEW を起動することで実現しています。ここで起動する側をマスタ HEW、起動された側をスレーブ HEW と呼びます。スレーブ HEW を起動するには、[ツール->従属 HEW の起動]を選択するか、[従属 HEW の実行]ツールバーボタン( )をクリックします。

スレーブ HEW はマスタ HEW と同様の操作が可能です。

マスタ HEW での下記動作をスレーブ HEW に通知します。これによりスレーブ HEW をマスタ HEW と同期させて動作させることができます。

- リセット実行
- 実行
- プログラムの停止

注：

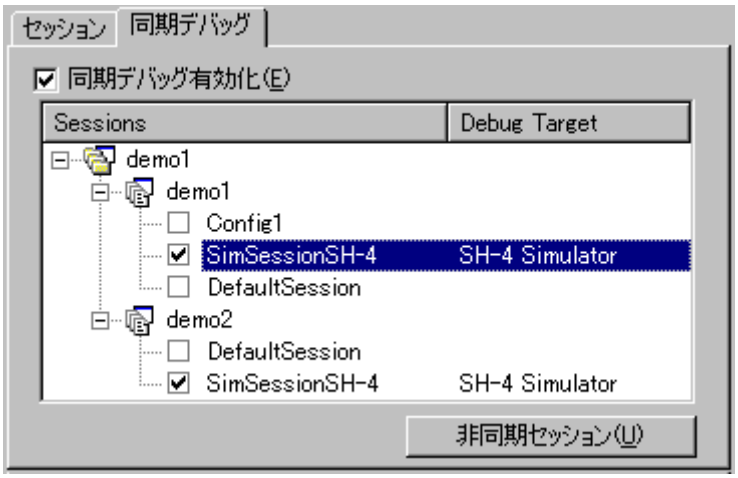
マスタ HEW から複数のスレーブ HEW アプリケーションを起動することはできますが、スレーブ HEW アプリケーションのネスト(スレーブ HEW からスレーブ HEW を起動する)はできません。

17.17.2 HEW デバッガターゲットの同期

HEW では、内部の複数ターゲットデバッグもサポートしています。これにより、複数のターゲットコンポーネントを同じ HEW アプリケーション内で接続できます。これらのターゲットは同時にデバッグできます。ユーザは異なるターゲットに複数のセッションを設定できます。そしてデバッグ時に、1 つのセッションのイベントに同期して他のセッションで同じイベントを引き起こすことができます。これは HEW アプリケーションでの同期と同様です。さらに、セッションを切り替えて同じアプリケーション内で何が起きているかを知ることができます。

HEW 内デバッガターゲットの同期を設定するには

1. [デバッグ->デバッグセッション]メニューを選択します。
2. ダイアログボックスの[同期デバッグ]タブを選択します。ダイアログボックスを下図に示します。
3. ワークスペース内で現在利用可能なすべてのセッションを表示するので、同期させたいセッションを選択します。
4. [同期セッション]ボタンをクリックします。選択したセッションのアイコンが"チェックなし"から"チェックあり"に変わります。
5. [同期デバッグ有効化]チェックボックスをオンにして本機能を有効にします。
6. [OK]ボタンをクリックします。



HEW 内デバッガターゲットの同期を利用するには

HEW 内デバッガターゲットの同期設定に続いて以下の操作を行います。

1. 標準ツールバーの[セッション]コンボボックスをクリックします。[Synch. Session]を選択します。このオプションはシステムに同期デバッグが追加されている場合のみ有効になります。
2. 一度選択すると HEW で同期デバッグ機能が利用可能となります。これにより[シンクロナイズセッション]という別のツールバーを追加します。
3. [シンクロナイズセッション]ツールバー上の有効/無効ツールバーボタンで、設定を変更しないで一時的に同期設定を切り替えることができます。
4. [シンクロナイズセッション]ツールバーでのセッション切り替えで現在見えるセッションを切り替えることができます。通常の HEW デバッグ状態ではこれはセッションを閉

じることを意味しますが、同期セッションでは、複数セッションを開くことができ、現在このツールバーで選択されているセッションを見ることができます。

5. これにより、複数のターゲットまたは CPU コアを同時にデバッグできます。セッションを変えるとウィンドウやウィンドウに表示されるデータを変えることができます。

注：

同期機能を有効にすると多くのことが同期して実行できます。

以下の表で、2 個の同期したセッションを例に同期デバッグ有効時の動作を示します。

ユーザ操作	ターゲットデバッグセッション 1	ターゲットデバッグセッション 2
いずれかのセッションで[実行]	“実行”	“実行”
いずれかのセッションで[ステップ]	“ステップ”	“ステップ”
いずれかのセッションで’ESC’を押す	“停止”	“停止”
-	ブレークポイントまたはユーザプログラム不正による”停止”	実行停止（’ESC’を押したことによる停止と同じ結果）
-	実行停止（’ESC’を押したことによる停止と同じ結果）	ブレークポイントまたはユーザプログラム不正による”停止”
いずれかのセッションで[CPU リセット]	“CPU リセット”	“CPU リセット”

以下の表で、2 個の同期したセッションを例に同期デバッグ無効時の動作を示します。

ユーザ操作	ターゲットデバッグセッション 1	ターゲットデバッグセッション 2
セッション 1 で[実行]	“実行”	何もしない。[実行]はユーザの入力が必要
セッション 2 で[実行]	何もしない。[実行]はユーザの入力が必要	“実行”
セッション 1 で[ステップ]	“ステップ”	何もしない。[ステップ]はユーザの入力が必要
セッション 2 で[ステップ]	何もしない。[ステップ]はユーザの入力が必要	“ステップ”
セッション 1 で’ESC’を押す	“停止”	ユーザプログラム実行中ならば、実行継続
セッション 2 で’ESC’を押す	ユーザプログラム実行中ならば、実行継続	“停止”
-	ブレークポイントまたはユーザプログラム不正による“停止”	ユーザプログラム実行中ならば、実行継続
-	ユーザプログラム実行中ならば、実行継続	ブレークポイントまたはユーザプログラム不正による”停止”
セッション 1 で[CPU リセット]	“CPU リセット”	何もしない
セッション 2 で[CPU リセット]	何もしない	“CPU リセット”

注：

標準のデバッグ時とのもう 1 つの相違点は、すべての同期セッションのダウンロードモジュールをワークスペースウィンドウに表示できることです。これにより、使用可能なセッションにモジュールを容易にダウンロードすることができるようになります。

17.18 デバッギングプラットフォームに依存するデバッグ機能

この節に記載している機能は、連携ツールの製品パッケージにバンドルされている HEW で共通のデバッグ機能ではありません。前バージョンのユーザーズマニュアルおよびヘルプに記載していたデバッグ機能です。SuperH RISC engine ファミリ用または H8, H8S, および H8SX ファミリ用のデバッギングプラットフォームを使用した場合のデバッグ機能の一部を記載しています。

この他のデバッギングプラットフォームに依存するデバッグ機能については、エミュレータまたはシミュレータのユーザーズマニュアルおよびヘルプを参照してください。

17.18.1 ラベルを見る

デバッグ情報を含んだユーザプログラムをロードした時に、ラベルを同時に登録します。ラベルを追加することも可能です。

[逆アセンブリ]ウィンドウにおいては、対応するアドレスの代わりとして、また命令オペランドの一部として、ラベルの最初の 8 文字を表示します。

注：

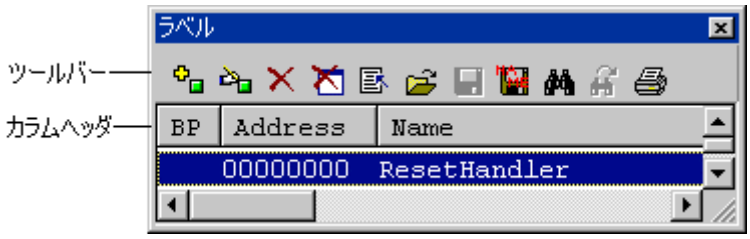
オペランドとラベルの値が一致すれば、命令のオペランドはラベル名に置き換わります。同じ値を持つラベルが 2 つ以上ある場合、アルファベット順で先に来るラベルを表示します。

ダイアログボックスで、アドレスまたは値を入力できる場合には、ラベルを使用することもできます。

(1) ラベルを一覧にする

現在のデバッグセッションに定義したラベルのすべてを見るには、[表示->シンボル->ラベル]を選択するか、[ラベルの表示]ツールバーボタン( )をクリックします。

ウィンドウ構成



- それぞれのカラムヘッダをクリックするとアルファベット順（ASCII コードによって）またはアドレス値でソートしたシンボルを表示できます。

- [BP]カラムをダブルクリックすると関数の入り口でソフトウェアブレークポイントをすばやく設定したり解除したりできます。（「マクロの記録」）

ウィンドウのオプション

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

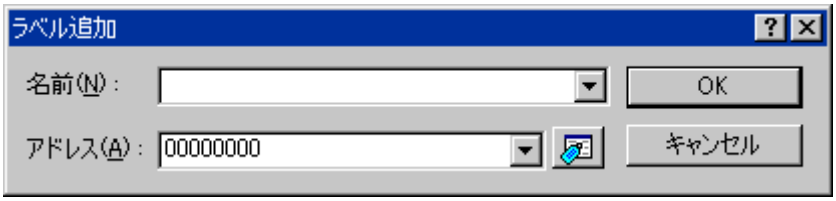
本ウィンドウは、基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	マクロの記録	機能
追加		●	ラベルを追加します。
編集		-	ラベルを編集します。
削除		●	ラベルを削除します。
すべてを削除		●	すべてのラベルを削除します。
ロード		●	ラベルをファイルからロードします。
上書き保存		-	ラベルをシンボルファイルに上書き保存します。
名前を付けて保存		-	ラベルをファイルに保存します。
検索		-	ラベルを検索します。
次を検索		-	ラベルが検索できた後、検索条件に一致する次のラベルを検索します。
ソースを表示		-	ラベルに対応するソースプログラムを表示します。
印刷		-	表示内容を印刷します。
ツールバー表示	-	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ	-	-	ツールバーボタンをカスタマイズします。

(2) ラベルを追加する

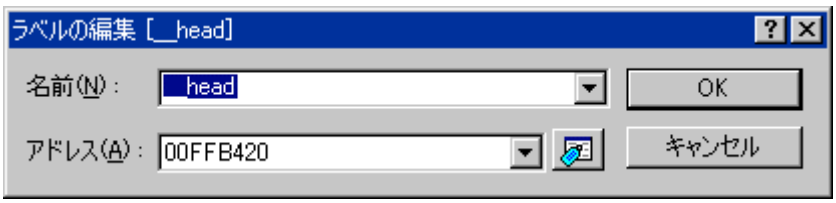
ラベルを追加するには、ポップアップメニューから[追加]を選択して、[ラベル追加]ダイアログボックスを表示します。



新しいラベル名を[名前]フィールドに入力し、対応する値を[アドレス]フィールドに入力して[OK]ボタンを押します。[ラベル追加]ダイアログボックスがクローズし、ラベルリストに新しいラベルを追加、更新します。多重定義関数やクラス名を入力したときは、[関数選択]ダイアログボックスが開くので、関数を選択して[アドレス]フィールドを設定します。詳細は「17.18.2 (3) 複数ラベルをサポートする」を参照してください。

(3) ラベルを編集する

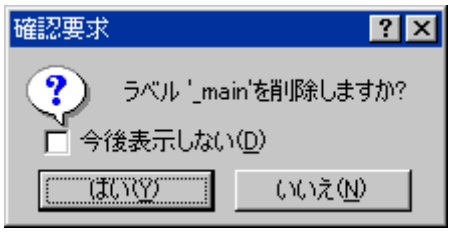
ラベルを編集するにはポップアップメニューから[編集]を選択します。[ラベルの編集]ダイアログボックスを表示します。



ラベル名と対応する値を編集して、[OK]ボタンを押すとラベルリストに編集を反映し、保存します。多重定義関数やクラス名を入力したときは、[関数選択]ダイアログボックスが開くので、関数を選択して[アドレス]フィールドを設定します。詳細は「17.18.2 (3) 複数ラベルをサポートする」を参照してください。

(4) ラベルを削除する

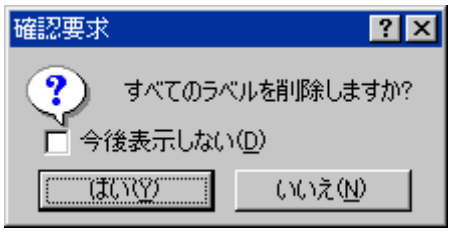
削除したいラベルを選択した状態で、ポップアップメニューから[削除]を選択します。この際、確認メッセージボックスを表示します。



[OK]ボタンを押すとラベルリストから削除し、ウィンドウを更新します。このメッセージボックスの表示が不要のときは、[今後表示しない]チェックボックスをオンにしてください。

(5) すべてのラベルを削除する

ポップアップメニューから[すべてを削除]を選択すると、リストからすべてのラベルを削除します。この際、確認メッセージボックスを表示します。



[OK]ボタンを押すと、すべてのラベルを HEW のシンボルテーブルから削除し、リスト表示もクリアします。メッセージボックスの表示が不要のときは、[今後表示しない]チェックボックスをオンにしてください。

(6) ラベルをファイルからロードする

シンボルファイルをロードして現在の HEW のシンボルテーブルに結合できます。ポップアップメニューから[ロード]を選択すると、シンボルをロードするダイアログボックスを表示します。

ダイアログボックスは、Windows®標準のファイルを開くダイアログボックスと同様です。ファイルを選択し、[開く]ボタンを押すとロードを開始します。シンボルファイルの標準拡張子は".sym"です。

(7) ラベルをファイルに保存する

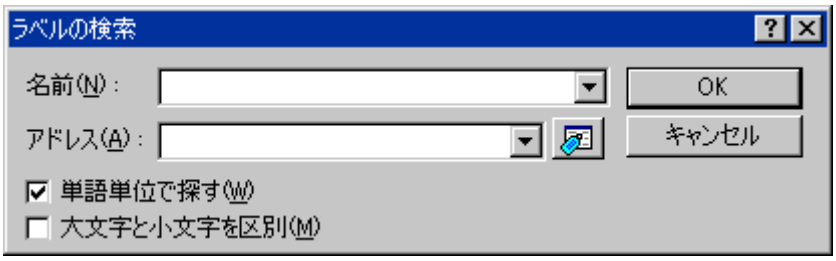
ポップアップメニューから[名前を付けて保存]を選択すると、シンボルを保存するダイアログボックスを表示します。ダイアログボックスは Windows®標準の名前を付けて保存ダイアログボックスと同様に操作できます。[ファイル名]フィールドにファイル名を入力し[保存]ボタンを押すとシンボルファイルにラベルリストを保存します。標準ファイル拡張子は".sym"です。

シンボルフォーマットについては「リファレンス 7. シンボルファイルフォーマット」を参照してください。

一度[名前を付けて保存]メニューでファイルに保存すると、以後はポップアップメニューの[上書き保存]で、現在のシンボルテーブルを同一シンボルファイルに保存できます。

(8) ラベルを検索する

ポップアップメニューから[検索]を選択すると、[ラベルの検索]ダイアログボックスを表示します。



1. [名前]フィールドに検索するラベル名を入力するか、ドロップダウンリストボックスから以前検索したラベル名を選んでください。
2. [アドレス]フィールドに検索するラベルのアドレスを入力するか、ドロップダウンリストボックスから以前検索したアドレスを選んでください。
3. [名前]フィールドに入力した名前と一致するラベルを検索する場合は、[単語単位で探す]チェックボックスをオンにしてください。このオプションをチェックしない場合は、[名前]フィールドに入力した名前と一部でも一致するラベルを検索します（[アドレス]フィールドは指定できません）。
4. ラベル名の大文字と小文字を識別したいときは[大文字、小文字を区別する]チェックボックスをオンにしてください。
5. [OK]ボタンをクリックすると検索を始めます。

注：

ラベルは、はじめの 1024 文字分しか情報を保持していません。したがって、ラベル名のはじめの 1024 文字分は重複しないようにしてください。

(9) 次を検索する

ラベルが検索できた後、ポップアップメニューから[次を検索]を選択すると、検索条件に一致する次のラベルを検索します。

(10) ラベルに対応するソースコードを表示する

ラベルを選択した状態で、ポップアップメニューから[ソースを表示]を選択すると、ラベルのアドレスに対応するソースファイルを開きます。

17.18.2 Elf/Dwarf2 のサポート

HEW は、C/C++およびアセンブリ言語で書いたアプリケーションのデバッグのために Elf/Dwarf2 オブジェクトファイルフォーマットをサポートします。

主な特長：

- ソースレベルデバッグ
- C/C++演算子
- C/C++式 (キャスト、ポインタ、参照など)
- あいまいな関数名
- オーバレイメモリロード
- ウォッチ-ローカル、およびユーザ定義
- スタックトレース

(1) C/C++演算子

以下の C/C++ 言語演算子を使用できます。

+, -, *, /, &, |, ^, ~, !, >>, <<, %, (,), <, >, <=, >=, ==, !=, &&, ||

Buffer_start + 0x1000

#R1 | B'10001101

((pointer + (2 * increment_size)) & H'FFFF0000) >> D'15

!(flag ^ #ER4)

(2) C/C++の式

式の例:

式	内容
Object.value	メンバの直接参照を指定します(C/C++)
p_Object->value	メンバの間接参照を指定します(C/C++)
Class::value	クラスを持つメンバの参照を指定します(C++)
*value	ポインタを指定します(C/C++)
&value	参照を指定します(C/C++)
array[0]	配列を指定します(C/C++)

Object.*value	ポインタを持つメンバの参照を指定します(C++)
::g_value	グローバル変数の参照を指定します(C++)
Class::function(short)	メンバ関数を指定します(C++)
(struct STR) *value	キャスト動作を指定します(C/C++)

(3) 複数ラベルをサポートする

プログラム言語の中の、例えば C++オーバーロード関数などでは、1 つのラベルが複数のアドレスを表す場合があります。各ダイアログボックスでこのようなラベル名を入力した場合、HEW は[関数選択]ダイアログボックスを使用してオーバーロード関数およびメンバ関数を表示します。

[関数選択]ダイアログボックスでは、オーバーロード関数またはメンバ関数を選択します。通常、一度に 1 つの関数を選択します。ただし、ブレークポイントを設定する場合においてのみ、複数の関数を選択できます。このダイアログボックスには 3 つの領域があります。

[関数名の選択]　同じ名前をもつ関数またはメンバ関数、およびその詳細情報を表示します。
- [名前]　関数名を表示します。
- [アドレス]　関数アドレスを表示します。

[関数名の指定]　設定する関数およびそれらの詳細情報を表示します。
- [名前]　関数名を表示します。
- [アドレス]　関数アドレスを表示します。

[カウンタ]　同じ名前をもつ関数またはメンバ関数の数を表示します。
- [全関数]　同じ名前をもつ関数またはメンバ関数を表示します。
- [選択関数]　[関数名の選択]リストボックスに表示する関数の数を表示します。
- [指定関数]　[関数名の指定]リストボックスに表示する関数の数を表示します。

(a) 関数を選択する

[関数名の選択]リストボックスから選択したい関数をクリックして、[>]ボタンをクリックします。

選択した関数を[関数名の指定]リストボックスに表示します。[関数名の選択]リストの関数すべてを選択するには、[>>]ボタンをクリックします。

(b) 関数の選択を解除する

[関数名の指定]リストボックスから選択を解除する関数をクリックして、[<]ボタンをクリックします。すべての関数の選択を解除するには、[<<]ボタンをクリックします。選択を解除した関数は、[関数名の指定]リストボックスから[関数名の選択]リストボックスへ戻します。

(c) 関数を設定する

[OK]ボタンをクリックして、[関数名の指定]リストボックスに表示した関数を設定します。関数を設定し、[関数選択]ダイアログボックスを閉じます。

[キャンセル]ボタンをクリックすると、関数を設定せずにダイアログボックスを閉じます。

(4) オーバレイプログラムのデバッグ

オーバレイ機能を利用するプログラムをデバッグできます。ここではオーバレイ機能を使用するための設定について説明します。

(a) セクショングループを表示する

オーバレイ（いくつかのセクショングループを同じアドレス範囲に割り当てる）を使用すると、アドレス範囲およびセクショングループを[オーバレイ]ダイアログボックスに表示します。

[デバッグ->オーバレイの構成]を選択して[オーバレイ]ダイアログボックスを開きます。

このダイアログボックスには 2 つの領域があります。[アドレス]リストボックスおよび[セクション名]リストボックスです。

[アドレス]リストボックスは、オーバレイプログラムが使用するアドレス範囲を表示します。アドレス範囲の 1 つをクリックして[アドレス]リストボックスのアドレス範囲を選択します。

[セクション名]リストボックスは、選択したアドレス範囲に割り当てたセクショングループを表示します。

(b)　セクショングループを設定する

オーバレイ関数を使用するときは、最も優先度の高いセクショングループを[オーバレイ]ダイアログボックスで選択していなければ、HEW は正しく動作しません。

まず[アドレス]リストボックスに表示したアドレス範囲の 1 つをクリックします。選択したアドレス範囲に割り当てたセクショングループを[セクション名]リストボックスに表示します。

表示しているセクショングループの中から最も優先度の高いセクショングループをクリックします。

セクショングループを選択したら、[OK]ボタンをクリックして優先度の設定を保存して、ダイアログボックスを閉じます。[キャンセル]ボタンをクリックすると、優先度の設定を保存せずにダイアログボックスを閉じます。

注：

オーバレイ関数が使用するアドレス範囲内では[オーバレイ]ダイアログボックスに指定したセクションのデバッグ情報を参照します。したがって、現在ロードしているプログラムと同じセクションを[オーバレイ]ダイアログボックスで選択します。

17.18.3　変数を表示する

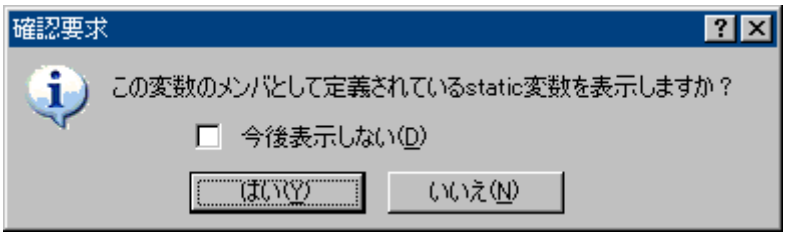
ソースプログラム上の変数の値を表示する方法について説明します。

変数を表示する際に以下に示す注意事項があります。

ルネサス製 SHC、H8C コンパイラで作成した ELF/DWARF2 フォーマットのロードモジュールをデバッグする場合、ウォッチウィンドウやローカルウィンドウ上で構造体を展開したときに static メンバ変数を通常のメンバ変数と同じように表示することができます。しかし、この機能はデバッグ情報量の多さによって変数表示に時間がかかることがあります。

ウォッチ/ローカルで変数展開時に static メンバを表示しないようにするには

1. [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を表示します。
2. [確認]タブを選択します。
3. [ウォッチ/ローカルで変数展開時に static メンバを表示]チェックボックスをオンにします。デフォルトはチェックボックスがオフです。
4. 次の確認ダイアログボックスが表示されたときに[いいえ]を選択してください。



この操作により変数展開時に static メンバ変数を表示しなくなり、表示時間を短縮できます。（[いいえ]を選んだ後も static メンバ変数を直接ウォッチウィンドウに追加することで static メンバ変数の参照は可能です）

(1) ツールチップウォッチ

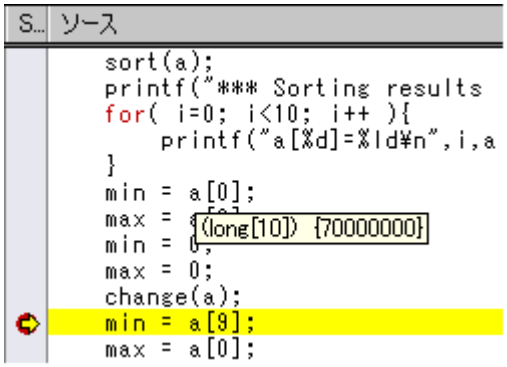
プログラムに定義された変数の値を参照するための機能です。値を参照したい変数を含むプログラムをエディタウィンドウまたは逆アセンブリウィンドウ（ソースモード）に表示し、変数名の上にマウスポインタを静止させると、ツールチップとしてポップアップウィンドウにウォッチ情報を表示します。

ツールチップウォッチ機能を使用するには

1. [基本設定->オプション]メニュー項目をクリックしてください。[オプション]ダイアログボックスを表示します。
2. [エディタ]タブを選択してください。
3. [ツールチップウォッチ]チェックボックスをオンにしてください。
4. [OK]ボタンをクリックしてください。

エディタウィンドウまたは逆アセンブリウィンドウ（ソースモード）にツールチップウォッチを表示するには

1. 確認したい変数を表示しているエディタウィンドウまたは逆アセンブリウィンドウ（ソースモード）を開きます。
2. 確認したい変数名の上にマウスのカーソルを静止させます。ツールチップとしてポップアップウィンドウに変数のウォッチ情報を表示します。

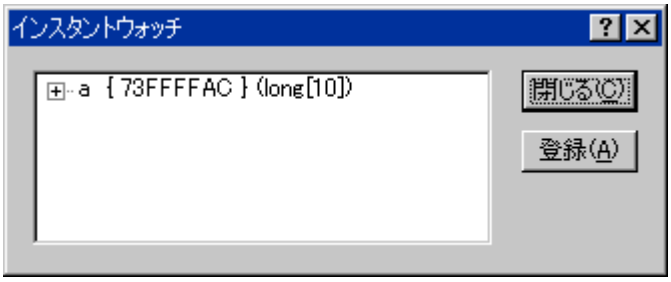


(2) インスタントウォッチ

確認したい変数を含むソースファイルをエディタウィンドウに表示します。

確認したい変数名の上にマウスのカーソルを置いてポップアップメニューから[インスタントウォッチ]を選択します。

[インスタントウォッチ]ダイアログボックスが開き、カーソル上の変数を表示します。



変数名の左側の'+'記号はクリックすれば情報を展開表示できることを、'-'記号は情報を縮小表示できることを示します。[登録]ボタンを押すと、変数を[ウォッチ]ウィンドウに登録して、ダイアログボックスを閉じます。[閉じる]ボタンを押すと、変数を[ウォッチ]ウィンドウに登録しないで、ダイアログボックスを閉じます。

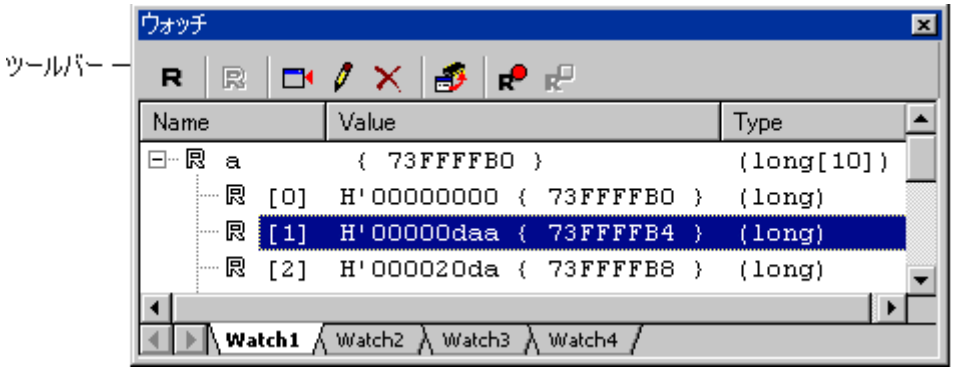
(3) ウォッチウィンドウ

[ウォッチ]ウィンドウを開くことにより、任意の変数について値を参照できます。

(a) ウォッチウィンドウを開く

[ウォッチ]ウィンドウを開くには、[表示->シンボル->ウォッチ]を選択するか、[ウォッチ]ツールバーボタン( )が使用可能であれば、クリックします。[ウォッチ]ウィンドウが開きます。

ウィンドウの構成



- C/C++ソースレベルの変数を表示・変更することができるウィンドウです。
- 本ウィンドウの内容は、アブソリュートファイル(*.abs)内のデバッグ情報から、C/C++ソースプログラムの情報がある場合にのみ表示します。コンパイラ等の最適化により、ソースプログラムの情報としてデバッグ情報にない場合は表示できません。また、マクロ宣言されたものについても表示できません。
- エディタウィンドウおよび逆アセンブリウィンドウから[ウォッチ]ウィンドウへのドラッグアンドドロップ機能により、C/C++ソースレベルの変数を登録することができます。

表示する項目は以下の通りです。

[Name]	変数名を表示します。
[Value]	変数の値、割付け位置を表示します。割付け位置は{}で囲んで表示します。
[Type]	変数の型を表示します。

R マークはその変数がリアルタイムに更新できることを示します。R マークが太字のとき、その変数の値をプログラムの実行時にリアルタイム更新します。

ウィンドウのオプション

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

本ウィンドウは、基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

ポップアップメニュー項目	ツールバーボタン	マクロの記録	機能
自動更新有効化	R	●	選択している変数の "R" マークが太字になり、リアルタイム更新します。
全シンボル自動更新有効化	R	●	すべての "R" マークが太字になり、リアルタイム更新します。
自動更新無効化	R	●	選択している変数の "R" マークが中抜きになり、リアルタイム更新を解除します。
全シンボル自動更新無効化	R	●	すべての "R" マークが中抜きになり、リアルタイム更新を解除します。

値更新記録	記録開始		●	[記録設定]ダイアログボックスを表示します。値更新の記録を開始します。
	記録終了		●	値更新の記録を終了します。
シンボル登録			●	[シンボル登録]ダイアログボックスを表示します。監視する変数または式を入力します。
値の編集			●	[値の変数]ダイアログボックスを表示して、変数の値を変更できるようにします。
削除			●	テキストカーソルの位置によって示された変数をウォッチウィンドウから削除します。
全シンボル削除			●	すべて変数をウォッチウィンドウから削除します。
基数	16 進数表示		●	16 進数で表示します。
	10 進数表示		●	10 進数で表示します。
	8 進数表示		●	8 進数で表示します。
	2 進数表示		●	2 進数で表示します。
コピー			-	選択部分をコピーし Windows®クリップボードに貼り付けます。
保存			-	表示内容をファイルに保存します。
メモリウィンドウへ移動			-	メモリウィンドウを表示します。
ツールバー表示		-	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ		-	-	ツールバーボタンをカスタマイズします。

(b) Watch アイテムを追加する

Watch アイテムを[ウォッチ]ウィンドウに追加するには、[ウォッチ]ウィンドウの[シンボル登録]ダイアログボックスを使用します。

[ウォッチ]ウィンドウから[シンボル登録]を使用するには

1. [ウォッチ]ウィンドウを開きます。
2. ポップアップメニューから[シンボル登録]を選択します。
3. [シンボル登録]ダイアログボックスが開きます。

4. 見たい変数名を入力して、[OK]ボタンをクリックします。その変数が[ウォッチ]ウィンドウに追加されます。

また、エディタウィンドウおよび逆アセンブリウィンドウ上のソースファイルから[ウォッチ]ウィンドウへ変数をドラッグアンドドロップしても追加できます。

注：

追加した変数がローカル変数で現在範囲外の場合には、HEW はその変数を[ウォッチ]ウィンドウに追加しますが、値には”Not available now”を表示します。

(c) Watch アイテムを拡張する

Watch アイテムがポインタ、アレイ、または構造体のとき、その名前の左側にプラス記号(+)の拡張指示子を表示します。つまり、Watch アイテムを拡張できるという意味です。Watch アイテムを拡張するには、プラス記号(+)をクリックします。拡張したアイテムは、タブ 1 つ分インデントをつけて、その要素(構造体またはアレイの場合)またはデータ値(ポインタの場合)を表示し、プラス記号がマイナス記号に変わります。Watch アイテムが要素にポインタ、構造体、またはアレイを含む場合、その横に拡張指示子を表示します。

拡張した Watch アイテムを元に戻すには、再び拡張指示子をクリックします。アイテムの要素は、元の単一のアイテムに戻り、マイナス記号はプラス記号に戻ります。

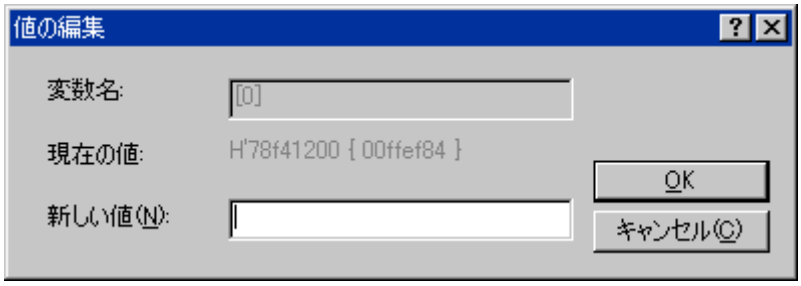
また、Watch アイテムを選択した状態で数字キー（1～9）を押すと、その数字の階層分 Watch アイテムを拡張できます。

(d) Watch アイテムの値を編集する

テストのためや、プログラムにバグがあるために値が正しくないときには、Watch 変数の値を変更できます。Watch アイテムの値を変更するには、値編集機能を使用します。

Watch アイテムの値を編集するには

1. ウィンドウに対して値を直接入力します。または、ポップアップメニューから[値の編集]を選択します。
2. [値の編集]ダイアログボックスが開きます。



3. [新しい値]フィールドに新しい値はまたは式を入力して[OK]ボタンをクリックします。[ウォッチ]ウィンドウを更新し、新しい値を表示します。

(e) Watch アイテムを削除する

特定の Watch アイテムを削除するには、そのアイテムを選択してポップアップメニューから[削除]を選択します。アイテムを削除し、ウォッチ表示を更新します。

すべての Watch アイテムを削除するには、ポップアップメニューから[全シンボル削除]を選択します。すべてのアイテムを削除し、ウォッチ表示を更新します。

[ウォッチ]ウィンドウで設定した Watch アイテムはセッションファイルに保存されます。

(f) 表示基数を変更する

変数を選択してポップアップメニューから[基数]を選択するとサブメニューで変数の表示基数を変更できます。

(g) 表示内容をファイルに保存する

[ウォッチ]ウィンドウの表示内容をファイルに保存するには、ポップアップメニューの[保存]を選択します。

[保存]を選択すると、[名前を付けて保存]ダイアログボックスを表示します。ファイル名を指定し、[ウォッチ]ウィンドウに表示している内容をセーブします。[Append]チェックボックスにチェックすると追加書きこみ、チェックしないと上書きします。

(h) メモリウィンドウを表示する

選択している変数が割り付いているメモリ領域を[メモリ]ウィンドウに表示できます。ポップアップメニューの[メモリウィンドウに移動]を選択すると、[メモリ]ウィンドウを開きます。

(i) リアルタイム更新を設定する

Watch アイテムで変数名の左に表示する”R”マークは、その変数がリアルタイムに更新できることを示します。

リアルタイム更新は[ウォッチ]ウィンドウのポップアップメニューで設定します。

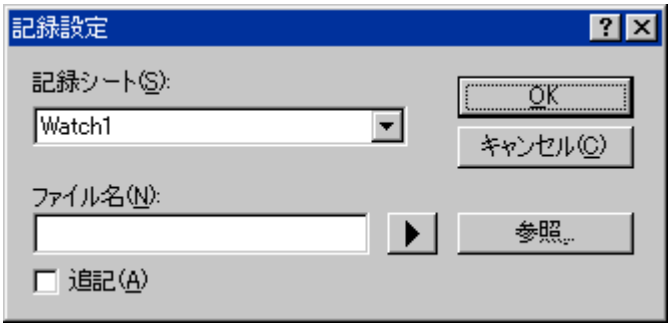
自動更新有効化	選択している変数の”R”マークが太字になり、リアルタイム更新します。
全シンボル自動更新有効化	すべての”R”マークが太字になり、リアルタイム更新します。
自動更新無効化	選択している変数の”R”マークが中抜きになり、リアルタイム更新を解除します。
全シンボル自動更新無効化	すべての”R”マークが中抜きになり、リアルタイム更新を解除します。

(j) 値更新を記録する

リアルタイム更新に設定した Watch アイテムは、値の変化をファイルに記録することができます。

値更新の記録を開始するには

1. ウィンドウ内を右クリックしポップアップメニューを表示します。
2. [値更新記録]を選択し、サブメニューから[記録開始]を選択します。
3. [記録設定]ダイアログボックスが表示されます。



4. [記録設定]ダイアログボックスでは、記録方法を指定します。

5. [記録シート]で値更新を記録するシートを選択してください。同時に記録できるシートは 1 枚です。
6. [ファイル名]で記録するファイル名を入力してください。[ファイル名]にプレースホルダを挿入するには、プレースホルダボタンをクリックし、ポップアップメニューからプレースホルダを選択します。ファイルを参照するには[参照]ボタンをクリックします。
7. [追記]チェックボックスをオンにすると入力したファイルに追記します。
8. [OK]ボタンをクリックすると、記録状態になります。ただし、次の場合は記録状態になりません。

- [記録シート]で指定したシートに自動更新が設定された Watch アイテムがない。
- [追記]チェックボックスがオンの場合、[ファイル名]に入力したファイルにすでに記録されている Watch アイテムと、[記録シート]領域で選択したシートの記録対象 Watch アイテムが一致しない。または、それらの表示基数が一致しない。

記録状態では、以下のタイミングで変数の値をファイルに記録します。

- 実行中の自動（リアルタイム）更新
- 実行停止時の値更新
- 値の編集時
- ウィンドウを操作したことによる表示の更新時

リアルタイム更新の記録を終了するには

1. ウィンドウ内を右クリックしポップアップメニューを表示します。
2. [値更新記録]を選択し、サブメニューから[記録終了]を選択します。

また次の場合、自動的に記録を停止します。

- ダウンロード、およびアンロード時
- 値更新記録の対象となっている Watch アイテムの展開が閉じられた
- 値更新記録の対象となっている Watch アイテムの表示基数を変更した
- 値更新記録の対象となっているシートで新たにリアルタイム更新対象 Watch アイテムを追加した
- 値更新記録の対象となっている Watch アイテムを並び替えた
- 値更新記録の対象となっている Watch アイテムのリアルタイム更新を解除した
- ウォッチウィンドウを閉じた

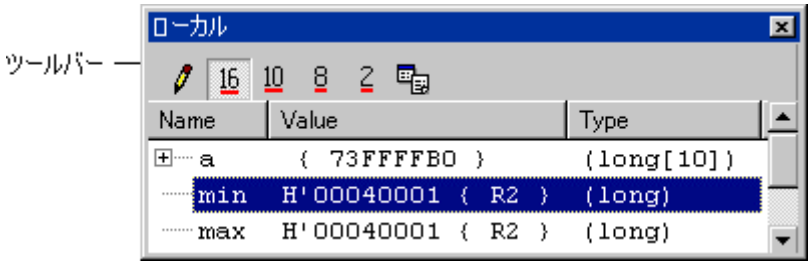
(4) ローカルウィンドウ

[ローカル]ウィンドウを開くことにより、ローカル変数の一覧とそれらの値を参照できます。

(a) ローカルウィンドウを開く

[ローカル]ウィンドウを開くには、[表示->シンボル->ローカル]を選択するか、[ローカル]ツールバーボタン( )をクリックします。

ウィンドウの構成



- [ローカル]ウィンドウが開き、現在のローカル変数を表示します。プログラムをデバッグしていくに従い、実行から停止またはブレークした後、[ローカル]ウィンドウを更新して現在のローカル変数とその値を表示します。ローカル変数を定義した時点で初期化していないと、ローカル変数に値を代入するまで[ローカル]ウィンドウの値は不定となります。
- このウィンドウは、アブソリュートファイル (*.abs)に含まれるデバッグ情報によって、現在の PC 位置から関数内にあるローカル変数に関連付けることができない場合は空白となります。

表示する項目は以下の通りです。

[Name]　変数名を表示します。
[Value]　変数の値、割付け位置を表示します。割付け位置は{}で囲んで表示します。
[Type]　変数の型を表示します。

ウィンドウのオプション

ウィンドウ内で右クリックすると、使用可能なオプションをポップアップメニューで表示します。

本ウィンドウは、基本的なオプションをツールバーに割り付けています。

[ツールバー表示]、[ツールバーのカスタマイズ]メニュー項目は、ツールバー内で右クリックしても、ポップアップメニューを表示します。

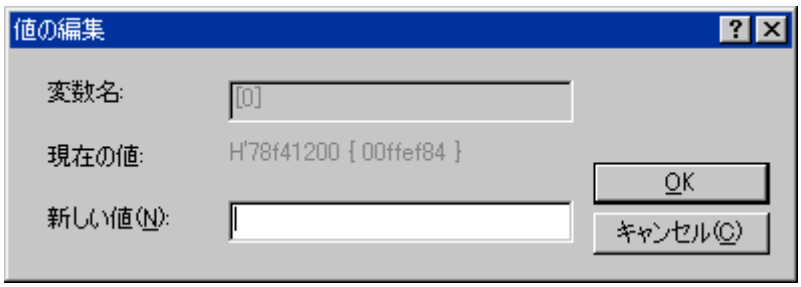
ポップアップメニュー項目		ツールバーボタン	マクロの記録	機能
値の編集			●	選択しているローカル変数の値を変更するダイアログボックスを表示します。
基数	16 進数表示	16	-	16 進数で表示します。
	10 進数表示	10	-	10 進数で表示します。
	8 進数表示	8	-	8 進数で表示します。
	2 進数表示	2	-	2 進数で表示します。
コピー			-	選択部分をコピーし Windows®クリップボードに貼り付けます。
ツールバー表示		-	-	ツールバー表示/表示なしを切り替えます。
ツールバーのカスタマイズ		-	-	ツールバーボタンをカスタマイズします。

(b) ローカル変数の値を編集する

テストのためや、プログラムにバグがあるために値が正しくないときには、ローカル変数の値を変更できます。ローカル変数の値を変更するには、値編集機能を使用します。

ローカル変数の値を編集するには

1. ウィンドウに対して値を直接入力します。または、ポップアップメニューから[値の編集]を選択します。
2. [値の編集]ダイアログボックスが開きます。



3. [新しい値]フィールドに新しい値はまたは式を入力して[OK]ボタンをクリックします。
4. [ローカル]ウィンドウを更新し、新しい値を表示します。

(c) 表示基数を変更する

変数を選択してポップアップメニューから[基数]を選択するとサブメニューで変数の表示基数を変更できます。

18. テクニカルサポート

18.1 アップデートの確認

HEW 製品のバージョンアップやサービスパックを確認するには

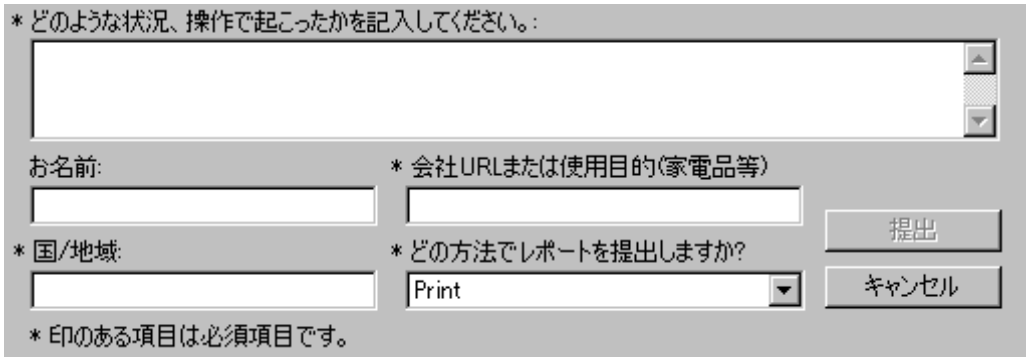
1. [ヘルプ->テクニカルサポート->アップデートの Web 確認]を選択してください。
2. デフォルトのウェブブラウザが起動して、お客さまの地域の HEW ダウンロードページを表示します。
3. HEW のバグの修正や新規機能の追加の更新情報を確認してください。

18.2 不具合レポートの作成

ときには、HEW アプリケーションで予期せぬ問題が起きるかもしれません。問題が起きてアプリケーションが破壊されると、自動的に HEW バグトラッキングプログラムが起動します。これにより、バグの情報を集約でき、様々な方法でテクニカルサポートに送ることができます。このバグトラッキングプログラムは手動で起動することもできます。この方法を以下に説明します。

HEW 不具合レポートを作成するには

1. [ヘルプ->テクニカルサポート->不具合レポートの作成]を選択してください。
2. HEW システムから詳細情報を生成します。これには多少時間がかかる場合があります。その後、[Submit a Bug Report]ダイアログボックスを表示します。
3. [どのような状況、操作で起こったか記入してください。]エディットボックスに起きた問題に関する情報を入力してください。入力必須項目です。
4. [お名前]エディットボックスにお客様の名前を入力してください。
5. [会社 URL または使用目的（家電品等）]エディットボックスにお客さまの会社の URL または開発製品の使用目的を入力してください。入力必須項目です。
6. [国/地域]エディットボックスにお住まいの国名または地域を入力してください。入力必須項目です。
7. 報告を作成したら、[どの方法でレポートを提出しますか?]ドロップダウンリストボックスで報告を提出する方法を選択してください。印刷、電子メール、ディスクに保存の 3 種類があります。選択必須項目です。
8. [提出]ボタンをクリックすると、選択した方法で報告を提出できます。



注：

*のある項目は入力必須項目です。これらの項目をすべて入力すると、[提出]ボタンがクリックできるようになります。

リファレンス

1. メインメニュー一覧

1.1 ファイルメニュー一覧

メニュー	メニューオプション	ショートカットキー	ツールバーボタン	マクロの記録*1	機能
ファイル	新規作成	Ctrl+N		-	新しいエディタを作成します。
	開く	Ctrl+O		-	ファイルを開きます。
	閉じる	Ctrl+F4	-	-	ファイルを閉じます。
	新規ワークスペース	-	-	-	新規にワークスペースを作成します。
	ワークスペースを開く	-	-	●	ワークスペースを開きます。
	ワークスペースの保存	-	-	●	ワークスペースを保存します。
	ワークスペースを閉じる	-	-	●	ワークスペースを閉じます。
	新規セッション	-	-	●	新しいセッションを作成します。
	セッションのインポート	-	-	●	既存のセッションインポートします。
	セッションの保存	-	-	●	セッションを保存します。
	名前を付けてセッションを保存	-	-	-	セッションを別名で保存します。
	セッションのリフレッシュ	-	-	●	セッションを再ロードします。
	ダウンロードモジュールの追加	-	-	●	新しくモジュールをダウンロードします。
	上書き保存	Ctrl+S		-	エディタウィンドウの内容を保存します。
	すべて保存	Ctrl+Shift+S		-	すべての開いているエディタウィンドウの内容を保存します。
	名前を付けて保存	-	-	-	エディタウィンドウの名前を新しい名前で保存します。
	ページレイアウトの設定	-	-	-	ページ設定します。
	印刷	Ctrl+P		-	ファイルを印刷します。
	最近使ったファイル	-	-	-	最近使ったファイルを開きます。
	最近使ったワークスペース	-	-	●	最近使ったワークスペースを開きます。
	最近使ったテストスイート	-	-	-	最近使ったテストスイートを開きます。
	最近ダウンロードしたモジュール	-	-	●	最近ダウンロードしたモジュールをダウンロードします。
	アプリケーションの終了	-	-	-	HEW を終了します。

***1.** 一部のメニュー項目に対する操作は、マクロ生成支援機能を使用し、HEW コマンドラインのコマンドを記録できます。このリストでは、メニュー項目の「マクロの記録」欄にマクロの記録マーク（●）を表示し、マクロファイルへ記録可能かどうかを示しています。これらの詳細は「15.5.1 マクロを記録できる機能(HEW 共通)」を参照してください。

1.2 編集メニュー一覧

メニュー	メニューオプション		ショートカットキー	ツールバーボタン	マクロの記録*1	機能
編集	元に戻す		Ctrl+Z	-	-	最新の編集操作を取り消します。
	やり直し		Ctrl+Y	-	-	最新の取り消した編集操作をやり直します。
	切り取り		Ctrl+X		-	選択部分を削除し Windows®クリップボードに貼り付けます。
	コピー		Ctrl+C		-	選択部分をコピーし Windows®クリップボードに貼り付けます。
	貼り付け		Ctrl+V		-	Windows®クリップボードの内容をコピーしてアクティブウィンドウのカーソル位置に貼り付けます。
	削除		Delete	-	-	選択部分を削除します（Windows®クリップボードに貼り付けません）。
	すべて選択		Ctrl+A	-	-	アクティブウィンドウの内容すべてを選択します。
	検索		Ctrl+F		-	現在のファイルのテキストを検索します。
	ファイルから検索		F4		-	複数のファイルでテキストを検索します。
	置換		Ctrl+H	-	-	テキストを置換します。
	ジャンプ		Ctrl+G	-	-	指定した行にジャンプします。
	括弧の呼応		Ctrl+Shift+M		-	括弧の組み合わせを見つけます。
	ブックマーク	ブックマークの挿入/削除	Ctrl+F2		-	ブックマークを設定/解除します。
		次のブックマーク	F2		-	次のブックマークにジャンプします。
		前のブックマーク	Shift+F2		-	同じファイルの 1 つ前のブックマークに戻ります。
		すべてのブックマークの削除	-		-	ファイル内のすべてのブックマークを削除します。
	テンプレート	テンプレートの定義	-		-	テンプレートを設定します
		テンプレートの挿入	Ctrl+Shift+T		-	テンプレートを挿入します
	ブレークポイントの挿入/削除		F9		●	標準のブレークポイント(PC ブレークポイント)を挿入/削除します。
	ブレークポイントの有効化/無効化		Ctrl+F9		●	現在設定している PC ブレークポイントの有効／無効の切り替えができます。
	表示カラムの設定		-	-	-	表示カラムを設定します。
	ソースブレークポイント		Ctrl+B	-	-	[ブレークポイント]ダイアログボックスを表示します。
	式の評価		-	-	-	式を評価します。

***1.** 一部のメニュー項目に対する操作は、マクロ生成支援機能を使用し、HEW コマンドラインのコマンドを記録できます。このリストでは、メニュー項目の「マクロの記録」欄にマクロの記録マーク（●）を表示し、マクロファイルへ記録可能かどうかを示しています。これらの詳細は「15.5.1 マクロを記録できる機能(HEW 共通)」を参照してください。

1.3 表示メニュー一覧

メニュー	メニューオプション		ショートカットキー	ツールバーボタン	マクロの記録*1	テストイメージデータの保存*2	機能
表示	差分		-		-	-	[差分]ウィンドウを表示します。
	マップ *3		-		-	-	2 種類のマップウィンドウを表示します。
	コマンドライン		Ctrl+L		-	-	[コマンドライン]ウィンドウを表示します。
	TCL ツールキット		Ctrl+Shift+K		-	-	詳細は別冊の「TCL/TK 添付資料」を参照してください。
	ワークスペース		Alt+K		●	-	ワークスペースウィンドウを表示します。
	アウトプット		Alt+O		-	*4	アウトプットウィンドウを表示します。
	逆アセンブリ		Ctrl+D		●	-	[逆アセンブリ]ウィンドウを表示します。
	CPU	レジスタ	Ctrl+R		●		[レジスタ]ウィンドウを表示します。
		メモリ	Ctrl+M		●		[メモリ]ウィンドウを表示します。
		IO	Ctrl+I		●		[IO]ウィンドウを表示します。
		ステイタス	Ctrl+U		-		[ステイタス]ウィンドウを表示します。
	グラフィック	画像	Ctrl+Shift+G		-	-	[画像]ウィンドウを表示します。
		波形	Ctrl+Shift+V		-	-	[波形]ウィンドウを表示します。
	コード	スタックトレース	Ctrl+K		-		[スタックトレース]ウィンドウを表示します。

***1.** 一部のウィンドウに対する特定の操作は、マクロ生成支援機能を使用し、HEW コマンドラインのコマンドを記録できます。このリストでは、メニュー項目の「マクロの記録」欄にマクロの記録マーク（●）を表示し、マクロファイルへ記録可能かどうかを示しています。これらの詳細は「15.5.1 マクロを記録できる機能(HEW 共通)」を参照してください。

このリストに記載していないデバッギングプラットフォームに依存するウィンドウでもコマンドが記録できるものがあります。これらの詳細は「15.5.2 マクロを記録できる機能(デバッギングプラットフォーム依存)」を参照してください。

***2.** 一部のウィンドウのデータは、テスト支援機能を使用し、テストイメージデータとしてテストイメージファイルに保存できます。このリストでは、テストイメージファイルに保存できるウィンドウのメニュー項目の「テストイメージファイルの保存」欄に[ファイルの保存]マーク（ ）を記載しています。これらの詳細は「16.6.1 テストイメージファイルに保存できる機能(HEW 共通)」を参照してください。

このリストに記載していないデバッギングプラットフォームに依存するウィンドウでもテストイメージファイルに保存できるものがあります。これらの詳細は「16.6.2 テストイメージファイルに保存できる機能(デバッギングプラットフォーム依存)」を参照してください。

***3.** 機能のサポートは、デバッギングプラットフォームに依存します。

***4.** [Build]タブまたは[Debug]タブのデータが保存可能です。

1.4 プロジェクトメニュー一覧

メニュー	メニューオプション	ショートカットキー	ツールバーボタン	マクロの記録*1	機能
プロジェクト	アクティブプロジェクトに設定	-	-	●	プロジェクトを現在のプロジェクトに設定します。
	プロジェクトの挿入	-	-	●	ワークスペースにプロジェクトを追加します。
	依存関係	-	-	-	依存プロジェクトを生成します。
	構成の編集 ***2**	-	-	●	プロジェクトの構成を編集します。
	プロジェクトタイプの生成	-	-	-	現在のプロジェクトの設定を利用してプロジェクトのテンプレートを作成します。
	ファイルの追加	-	-	-	プロジェクトにファイルを追加します。
	ファイルの削除	-	-	-	プロジェクトからファイルを削除します。
	ファイルの拡張子	-	-	-	ファイルの拡張子を定義します。
	コンポーネント	-	-	-	オンデマンドのコンポーネントを設定します。

***1.** 一部のメニュー項目に対する操作は、マクロ生成支援機能を使用し、HEW コマンドラインのコマンドを記録できます。このリストでは、メニュー項目の「マクロの記録」欄にマクロの記録マーク（●）を表示し、マクロファイルへ記録可能かどうかを示しています。これらの詳細は「15.5.1 マクロを記録できる機能(HEW 共通)」を参照してください。

***2.** 機能のサポートは、デバッギングプラットフォームに依存します。

1.5 ビルドメニュー一覧

メニュー	メニューオプション	ショートカットキー	ツールバーボタン	マクロの記録*1	機能
ビルド ***2**	Toolchain	-	-	-	ファイルのビルド方法を設定します。
	コンパイル	Ctrl+F7		●	ファイルをコンパイルします。
	ビルド	F7		●	プロジェクトをビルドします。
	すべてをビルド	-		●	プロジェクトを全ビルドします。
	複数ビルド	-	-	●	複数のプロジェクトをビルドします。
	クリーン アクティブプロジェクト	-	-	●	現在のコンフィグレーションの中間ファイルおよび出力ファイルを削除します。
	クリーン 全プロジェクト	-	-	●	ワークスペースのすべてのプロジェクトのすべてのコンフィグレーションを削除します。
	すべての依存関係を更新	-	-	-	プロジェクトのファイル依存関係を更新します。
	ツールの中止	Ctrl+Break		-	ビルド中のツールの実行を中止します。
	ビルドから除外/除外の解除	-	-	-	ファイルをビルドから除外または除外を解除します。
	ビルドフェーズ	-	-	-	ビルドフェーズを設定します。
	ビルドの構成	-	-	●	ビルドコンフィグレーションを設定します。
	リンク順の設定	-	-	-	リンク順をカスタマイズします。
	Make ファイルの生成	-	-	-	Make ファイルを生成します。

***1.** 一部のメニュー項目に対する操作は、マクロ生成支援機能を使用し、HEW コマンドラインのコマンドを記録できます。このリストでは、メニュー項目の「マクロの記録」欄にマクロの記録マーク（●）を表示し、マクロファイルへ記録可能かどうかを示しています。これらの詳細は「15.5.1 マクロを記録できる機能(HEW 共通)」を参照してください。

***2.** 本メニューは、HEW V.4.01 以降で作成したデバッグ専用プロジェクトを使用する場合は表示されません。

1.6 デバッグメニュー一覧

メニュー	メニューオプション		ショートカットキー	ツールバーボタン	マクロの記録*1	機能
デバッグ	デバッグセッション		-	-	●	デバッグセッションの一覧表示、および追加／削除等が可能な[デバッグセッション]ダイアログボックスを表示します。
	デバッグの設定		-	-	-	デバッグ時の条件やダウンロードモジュール等の設定を行う[デバッグの設定]ダイアログボックスを表示します。
	CPU のリセット		-		●	ターゲットマイコンをリセットします。
	実行		F5		●	現在の PC からユーザプログラムを実行します。
	リセット後実行		Shift+F5		●	リセットベクタアドレスからユーザプログラムを実行します。
	カーソル位置まで実行		-		●	現在の PC アドレスからユーザプログラムの実行を開始します。プログラムは、PC がテキストカーソルの位置によって指定されたアドレスに達するか、別のブレーク条件が成立するまで実行します。
	カーソル位置を PC 値に設定		-		●	PC の値をテキストカーソルの位置によって指定されたアドレスに変更します。
	条件を指定して実行		-	-	●	実行時の PC や PC ブレークポイントの設定が可能な[条件を指定して実行]ダイアログボックスを表示します。
	PC 位置を表示		Ctrl+Shift+Y		-	PC 位置を表示します。
	ステップイン		F11		●	ユーザプログラムの 1 ブロックを実行して停止します。
	ステップオーバ		F10		●	ユーザプログラムの 1 ブロックを実行して停止しますが、サブルーチンを呼び出す場合は、サブルーチンには入りません。
	ステップアウト		Shift+F11		●	現在の関数の終わりに到達するまでユーザプログラムを実行します。
	ステップ		-	-	●	ステップ動作の設定が可能な[プログラムステップ]ダイアログボックスを表示します。
	ステップモード	自動	-	-	●	エディタウィンドウがアクティブの場合はソースライン 1 行だけをステップ実行します。 [逆アセンブリ]ウィンドウがアクティブの場合はアセンブリ言語命令単位にステップ実行します。
		アセンブリ	-	-	●	アセンブリ言語命令単位にステップ実行します。
		ソース	-	-	●	ソースライン 1 行だけをステップ実行します。
	プログラムの停止		-		●	ユーザプログラムの実行を停止します。
	初期化		-	-	●	デバッギングプラットフォームを切断し、再接続します。
	接続*2		-		●	デバッギングプラットフォームを接続します。
	接続解除*2		-		●	デバッギングプラットフォームを接続解除します。
	メモリの保存		-	-	●	任意のアドレス領域を保存します。
	メモリのベリファイ*2		-	-	●	アドレス領域を検証します。
	ダウンロード		-	-	●	オブジェクトプログラムをロードします。
	アンロード		-	-	●	オブジェクトプログラムをアンロードします。

***1.** 一部のメニュー項目に対する操作は、マクロ生成支援機能を使用し、HEW コマンドラインのコマンドを記録できます。このリストでは、メニュー項目の「マクロの記録」欄にマクロの記録マーク（●）を表示し、マクロファイルへ記録可能かどうかを示しています。これらの詳細は「15.5.1 マクロを記録できる機能(HEW 共通)」を参照してください。

***2.** 機能のサポートは、デバッギングプラットフォームに依存します。

1.7 基本設定メニュー一覧

メニュー	メニューオプション		ショートカットキー	ツールバーボタン	マクロの記録*1	機能
基本設定	カスタマイズ		-	-	-	HEW アプリケーションの設定をカスタマイズします。
	オプション		-	-	-	HEW アプリケーションのオプションを設定します。
	表示の形式		-	-	-	ウィンドウの表示色、フォント、キーワードなどを設定します。
	基数	16 進数	-	16	●	基数を 16 進数に設定します。
		10 進数	-	10	●	基数を 10 進数に設定します。
		8 進数	-	8	●	基数を 8 進数に設定します。
		2 進数	-	2	●	基数を 2 進数に設定します。

***1.** 一部のメニュー項目に対する操作は、マクロ生成支援機能を使用し、HEW コマンドラインのコマンドを記録できます。このリストでは、メニュー項目の「マクロの記録」欄にマクロの記録マーク（●）を表示し、マクロファイルへ記録可能かどうかを示しています。これらの詳細は「15.5.1 マクロを記録できる機能(HEW 共通)」を参照してください。

1.8 ツールメニュー一覧

メニュー	メニューオプション		ショートカットキー	ツールバーボタン	機能
ツール	アドミニストレーション		-	-	アドミニストレーションを設定します。
	ツールチェインバージョンを変更		-	-	ツールチェインバージョンを変更します。
	バージョン管理	選択	-	-	バージョン管理システムを選択します。
		構成	-	-	構成を設定します。
		ファイルの追加	-		バージョン管理に選択したファイルを追加します。
		ファイルの削除	-		バージョン管理から選択したファイルを削除します。
		ファイルの取得	-		バージョン管理から選択したファイルの読み取り専用ローカルファイルを取得します。
		ファイルのチェックアウト	-		バージョン管理から選択したファイルの書き込み可能なローカルファイルを取得します。
		ファイルのチェックイン	-		バージョン管理に選択したファイルのローカルコピーを戻します（更新します）。
		ファイルの状態	-		選択したファイルの状態を表示します。
	外部デバッガの起動		-		外部デバッガを起動します。
	従属 HEW の起動		-		従属 HEW を起動します。
	マクロの設定		-	-	マクロの設定ダイアログボックスを表示します。
	マクロの記録		-		マクロの記録を開始します。
	マクロの実行		-		マクロを実行します。
	マクロの停止		-		マクロを停止します。

[マクロの停止]メニューオプションの下には記録したマクロを割り当てたカスタムメニューオプションを追加することが可能です。

1.9 テストメニュー一覧

メニュー	メニューオプション	ショートカットキー	ツールバーボタン	機能
テスト	テストスイートの作成	-	-	新規に空のテストスイートを作成します。テストの追加、実行を開始できます。
	テストスイートを開く	-	-	既存のテストスイートを開きます。
	テストスイートの編集	-	-	現在のテストスイートを編集します。テストスイートへのテストの追加および削除ができます。
	テストスイートを閉じる	-	-	現在開いているテストスイートを閉じます。
	テストイメージファイルの作成	-	-	テストイメージファイル（*.HIF）に保存するテストイメージデータの設定およびカスタマイズが可能です。
	テストイメージファイルの比較	-	-	テストイメージファイルを、現在の HEW システム、またはディスクにある別のテストイメージファイルと比較します。結果はテストブラウザに表示されます。
	テストを実行	-	-	テストスイートで定義した複数のテストを実行し、比較した結果をテストブラウザで見ることができます。テストの実行方法を指定する様々なオプションがあります。
	テスト結果ブラウザ	-	-	前回実行したテストの結果を表示します。テストの成功／失敗と、失敗した要因の詳細を見ることができます。

1.10　ウィンドウメニュー一覧

メニュー	メニューオプション		ショートカットキー	ツールバーボタン	機能
ウィンドウ	重ねて表示		-	-	開いているすべてのウィンドウを重ねて並べます。
	上下に並べて表示		-	-	ウィンドウを上下に並べます。
	左右に並べて表示		-	-	ウィンドウを左右に並べます。
	アイコンの整列		-	-	アイコンを整列します。
	すべて閉じる		-	-	開いているすべてのウィンドウを閉じます。
	バーチャルデスクトップ	デスクトップマネージャ	-	-	デスクトップ構成に新しい名前を付けます。
		Default1-4	-	-	デスクトップ構成を切り替えます。

1.11　ヘルプメニュー一覧

メニュー	メニューオプション		ショートカットキー	ツールバーボタン	機能
ヘルプ	トピック		-	-	High-performance Embedded Workshop ヘルプウィンドウを表示します。
	テクニカルサポート	不具合レポートの作成	-	-	High-performance Embedded Workshop 不具合レポートを作成します。
		アップデートの web 確認	-	-	High-performance Embedded Workshop 製品のバージョンアップやサービスパックを確認します。
	High-performance Embedded Workshop のバージョン情報		-	-	High-performance Embedded Workshop のバージョン情報を表示します。
	デバッガヘルプ		-	-	デバッギングプラットフォーム接続時エミュレータまたはシミュレータのヘルプウィンドウを表示します。

2. ウィンドウ一覧

ウィンドウ名	メニュー
差分	[表示->差分]
Map Section Information	[表示->マップ]
Map Symbol Information	
コマンドライン	[表示->コマンドライン]
Console ***1**	[表示->TCL ツールキット]
Workspace	[表示->ワークスペース
Output	[表示->アウトプット]
逆アセンブリ	[表示->逆アセンブリ]
レジスタ	[表示->CPU->レジスタ]
メモリ	[表示->CPU->メモリ]
IO	[表示->CPU->IO]
ステイタス	[表示->CPU->ステイタス]
画像	[表示->グラフィック->画像]
波形	[表示->グラフィック->波形]
スタックトレース	[表示->コード->スタックトレース]

***1**. 別冊の「TCL/TK 添付資料」を参照してください。

3. コマンドライン一覧

3.1 コマンド一覧（アルファベット順）

コマンド名	短縮形	説明
!	-	コメント
ADD_FILE	AF	カレントプロジェクトへのファイル追加
ASSERT	-	コンディションのチェック
AUTO_COMPLETE	AC	オートコンプリートの切り替え
BUILD ***1**	BU	カレントプロジェクトのビルド
BUILD_ALL ***1**	BL	カレントプロジェクトのすべてのビルド
BUILD_FILE ***1**	BF	ファイルのコンパイル
BUILD_MULTIPLE ***1**	BM	複数プロジェクトのビルド
CACHE ***2**	-	メモリキャッシュの有効化/無効化
CHANGE_CONFIGURATION	CC	コンフィグレーションの設定
CHANGE_PROJECT	CP	プロジェクトの設定
CHANGE_SESSION	CS	セッションの設定
CHANGE_SUB_SESSION ***2**	CB	同期デバッグが有効な場合のアクティブサブセッションの変更
CLEAN	CL	ビルドの中間ファイルおよび出力ファイルの削除
CLEAR_OUTPUT_WINDOW	COW	アウトプットウィンドウの各タブの表示内容クリア
CLOSE_TEST_SUITE	CTS	テストスウィートを閉じる
CLOSE_WORKSPACE	CW	ワークスペースを閉じる
COMPARE_TEST_DATA	CTD	テストデータの比較
CONNECT ***2**	CN	デバッギングプラットフォームの接続
DEFAULT_OBJECT_FORMAT	DO	デフォルトオブジェクトフォーマットの設定
DISCONNECT ***2**	DN	デバッギングプラットフォームの切断
ERASE	ER	[コマンドライン]ウィンドウの内容クリア
EVALUATE	EV	式の計算
FILE_LOAD	FL	オブジェクト(プログラム)ファイルのロード
FILE_LOAD_ALL	LA	すべてのオブジェクト(プログラム)ファイルのロード
FILE_SAVE	FS	メモリ内容のファイルセーブ
FILE_UNLOAD	FU	オブジェクト(プログラム)ファイルのアンロード
FILE_UNLOAD_ALL	UA	すべてのオブジェクト(プログラム)ファイルのアンロード
FILE_VERIFY ***2**	FV	ファイル内容とメモリ内容の比較
GENERATE_MAKE_FILE ***1**	GM	カレントワークスペースの make ファイル作成
GO	GO	ユーザプログラムの実行
GO_RESET	GR	リセットベクタからのユーザプログラムの実行
GO_TILL	GT	テンポラリブレークポイントまでのユーザプログラムの実行
HALT	HA	ユーザプログラムの停止
HELP	HE	コマンドラインヘルプ
INITIALIZE	IN	デバッギングプラットフォームの初期化
LOG	LO	ロギングファイルの操作
MEMORY_COMPARE ***2**	MC	メモリ内容の比較
MEMORY_DISPLAY	MD	メモリ内容の表示
MEMORY_EDIT	ME	メモリ内容の変更
MEMORY_FILL	MF	指定データによるメモリ内容の一括変更
MEMORY_FIND ***2**	MI	メモリ内容の検索
MEMORY_MOVE	MV	メモリブロックの移動

MEMORY_TEST ***2**	MT	メモリブロックのテスト
OPEN_TEST_SUITE	OTS	テストスイートを開く
OPEN_WORKSPACE	OW	ワークスペースを開く
QUIT	QU	HEW の終了
RADIX	RA	入力ラディックス(基数)の設定
REFRESH_SESSION	RSE	セッション情報の再読み込み
REMOVE_FILE	REM	カレントプロジェクトからのファイル削除
RESET	RE	CPU のリセット
RUN_TEST	RT	テストの実行
SAVE_SESSION	SE	現在のセッションの保存
SAVE_WORKSPACE	SW	ワークスペースの保存
SET_DISASSEMBLY_SOFT_BREAK	SDB	逆アセンブリレベルのソフトウェアブレークポイントの設定または解除
SET_SOURCE_SOFT_BREAK	SSB	ソースレベルのソフトウェアブレークポイントの設定または解除
SLEEP	-	コマンド実行の遅延
STATE_DISASSEMBLY_SOFT_BREAK	TDB	逆アセンブリレベルのソフトウェアブレークポイントの有効化または無効化
STATE_SOURCE_SOFT_BREAK	TSB	ソースレベルのソフトウェアブレークポイントの有効化または無効化
STEP	ST	ステップ実行(命令単位またはソース行単位)
STEP_MODE	SM	ステップモードの設定
STEP_OUT	SP	現在の関数を終了するまでのステップ実行
STEP_OVER	SO	ステップオーバ実行
STEP_RATE	SR	ステップ実行速度の設定、表示
SUBMIT	SU	コマンドファイルの実行
TCL	-	TCL の有効化/無効化
TOOL_INFORMATION	TO	ツール情報の出力
UPDATE_ALL_DEPENDENCIES ***1**	UD	カレントプロジェクトのすべての依存関係更新

***1.** ツールチェインが登録されていない場合は、使用できません。

***2.** 機能のサポートは、デバッギングプラットフォームに依存します。

各コマンドのシンタックスについてはオンラインヘルプを参照してください。

3.2 コマンド一覧（機能別）

HEW アプリケーション制御コマンド

コマンド名	短縮形	説明
ADD_FILE	AF	カレントプロジェクトへのファイル追加
CHANGE_CONFIGURATION	CC	コンフィグレーションの設定
CHANGE_PROJECT	CP	プロジェクトの設定
CHANGE_SESSION	CS	セッションの設定
CHANGE_SUB_SESSION ***1**	CB	同期デバッグが有効な場合のアクティブサブセッションの変更
CLEAR_OUTPUT_WINDOW	COW	アウトプットウィンドウの各タブの表示内容クリア
CLOSE_WORKSPACE	CW	ワークスペースを閉じる
EVALUATE	EV	式の計算
OPEN_WORKSPACE	OW	ワークスペースを開く
QUIT	QU	HEW の終了
RADIX	RA	入力ラディックス(基数)の設定
REFRESH_SESSION	RSE	セッション情報の再読み込み
REMOVE_FILE	REM	カレントプロジェクトからのファイル削除
SAVE_SESSION	SE	現在のセッションの保存

SAVE_WORKSPACE	SW	ワークスペースを保存する
TOOL_INFORMATION	TO	ツール情報の出力

***1.** 機能のサポートは、デバッギングプラットフォームに依存します。

ビルド関連（ツールチェインを登録している場合に使用可能）

コマンド名	短縮形	説明
BUILD	BU	カレントプロジェクトのビルド
BUILD_ALL	BL	カレントプロジェクトのすべてのビルド
BUILD_FILE	BF	ファイルのコンパイル
BUILD_MULTIPLE	BM	複数プロジェクトのビルド
CLEAN	CL	ビルドの中間ファイルおよび出力ファイルの削除
GENERATE_MAKE_FILE	GM	カレントワークスペースの make ファイル作成
UPDATE_ALL_DEPENDENCIES	UD	カレントプロジェクトのすべての依存関係更新

コマンドライン操作関連

コマンド名	短縮形	説明
!	-	コメント
ASSERT	-	コンディションのチェック
AUTO_COMPLETE	AC	オートコンプリートの切り替え
ERASE	ER	[コマンドライン]ウィンドウの内容クリア
HELP	HE	コマンドラインヘルプ
LOG	LO	ロギングファイルの操作
SLEEP	-	コマンド実行の遅延
SUBMIT	SU	コマンドファイルの実行
TCL	-	TCL の有効化/無効化

テスト支援機能関連

コマンド名	短縮形	説明
CLOSE_TEST_SUITE	CTS	テストスイートを閉じる
COMPARE_TEST_DATA	CTD	テストデータの比較
OPEN_TEST_SUITE	OTS	テストスイートを開く
RUN_TEST	RT	テストの実行

実行関連（デバッギングプラットフォーム接続時に使用可能）

コマンド名	短縮形	説明
GO	GO	ユーザプログラムの実行
GO_RESET	GR	リセットベクタからのユーザプログラムの実行
GO_TILL	GT	テンポラリブレークポイントまでのユーザプログラムの実行
HALT	HA	ユーザプログラムの停止
INITIALIZE	IN	デバッギングプラットフォームの初期化
RESET	RE	CPU のリセット
STEP	ST	ステップ実行(命令単位またはソース行単位)
STEP_MODE	SM	ステップモードの設定
STEP_OUT	SP	現在の関数を終了するまでのステップ実行
STEP_OVER	SO	ステップオーバ実行
STEP_RATE	SR	ステップ実行速度の設定、表示

メモリ操作関連（デバッギングプラットフォーム接続時に使用可能）

コマンド名	短縮形	説明
CACHE***1**	-	メモリキャッシュの有効化/無効化
FILE_LOAD	FL	オブジェクト(プログラム)ファイルのロード
FILE_LOAD_ALL	LA	すべてのオブジェクト(プログラム)ファイルのロード
FILE_SAVE	FS	メモリ内容のファイルセーブ
FILE_UNLOAD	FU	オブジェクト(プログラム)ファイルのアンロード
FILE_UNLOAD_ALL	UA	すべてのオブジェクト(プログラム)ファイルのアンロード
FILE_VERIFY ***1**	FV	ファイル内容とメモリ内容の比較
MEMORY_COMPARE ***1**	MC	メモリ内容の比較
MEMORY_DISPLAY	MD	メモリ内容の表示
MEMORY_EDIT	ME	メモリ内容の変更
MEMORY_FILL	MF	指定データによるメモリ内容の一括変更
MEMORY_FIND ***1**	MI	メモリ内容の検索
MEMORY_MOVE	MV	メモリブロックの移動
MEMORY_TEST ***1**	MT	メモリブロックのテスト

***1.** 機能のサポートは、デバッギングプラットフォームに依存します。

ソフトウェアブレーク設定関連（デバッギングプラットフォーム接続時に使用可能）

コマンド名	短縮形	説明
SET_DISASSEMBLY_SOFT_BREAK	SDB	逆アセンブリレベルのソフトウェアブレークポイントの設定または解除
SET_SOURCE_SOFT_BREAK	SSB	ソースレベルのソフトウェアブレークポイントの設定または解除
STATE_DISASSEMBLY_SOFT_BREAK	TDB	逆アセンブリレベルのソフトウェアブレークポイントの有効化または無効化
STATE_SOURCE_SOFT_BREAK	TSB	ソースレベルのソフトウェアブレークポイントの有効化または無効化

その他デバッグ関連（デバッギングプラットフォーム接続時に使用可能）

コマンド名	短縮形	説明
CONNECT ***1**	CN	デバッギングプラットフォームの接続
DISCONNECT ***1**	DN	デバッギングプラットフォームの切断
INITIALIZE	IN	デバッギングプラットフォームの初期化

***1.** 機能のサポートは、デバッギングプラットフォームに依存します。

各コマンドのシンタックスについてはオンラインヘルプを参照してください。

4. 正規表現

HEW エディタでは検索、置換、およびファイルから検索操作の際、文字列に特殊文字を指定できます。

下表に指定できる文字とその詳細を示します。

文字	機能
?	改行文字以外の任意の一文字と一致します。 例えば、t?p は“top”や“tip” と一致するが“trap”とは一致しません。
*	改行文字以外の文字列（0 個以上の文字列）と一致します。2 行にまたがって一致することはありません。*文字は残りのパターンが一致するための最短文字列と一致します。 例えば、t*o は“too”の“to”, “trowel”の“tro” , “sporty orange”の“ty o”と一致しますが、 “smart orange” とは一致しません（*文字は 2 行にまたがって一致することはないため）。
¥n	改行文字と一致します。¥n は行の終わりや 2 行にわたるパターンを検索するときに使います。 例 1 ;¥n 行末のセミコロンと一致します。 例 2 ;¥nif セミコロンと改行文字が続き次に if で始まる行が続く場合を検索します。
¥t	タブと一致します。 例 1 ¥t8 タブの次に 8 がある場合を検索します。 例 2 init¥t init の次にタブがある場合を検索します。
[ ]	括弧内の文字または文字の範囲の中の任意の一文字と一致します。括弧をネストすることはできません。 [-]は文字の範囲を示します。例[a-z]、[0-9]。範囲の始まりの文字は終わりの文字より小さい ASCII コードでなければなりません。 [~]は[~ と]の間の文字以外の一文字と一致します。改行文字がこの括弧内になければ、改行文字とも一致します。 例 1 [AEIOU] 大文字の母音をすべて検索します。 例 2 [<>?] 記号<, > , ?を検索します。 例 3 [A-Za-z0-9_] 大文字、小文字、数字、下線を検索します。 例 4 [~0-9] 数字以外の文字を検索します。 例 5 [¥t¥n] 空白、タブ、改行文字を検索します。 例 6 [¥]] ¥のすぐ後に]を置くと、記号] を検索します。
¥	正規表現を無視する特殊文字です。バックスラッシュの次の文字が正規表現のとき、通常の文字として扱います。バックスラッシュの次が非正規表現のとき、バックスラッシュは無視されます。 例 1 ¥* アスタリスクをすべて検索します。 例 2 ¥¥ バックスラッシュをすべて検索します。

注： バックスラッシュ文字‘¥’ は日本語 Windows®では円記号として表示されます。

5. プレースホルダ

プレースホルダは、HEW の複数のツールによって提供される機能です。ここではプレースホルダの使い方を説明します。

5.1 プレースホルダとは？

プレースホルダとは一時的にテキストに挿入される特殊文字列です。後に実際の値に置き換えます。例えば、HEW のプレースホルダの 1 つに、$(FULLFILE)があります。これは、すべてのパス付きのファイルを示します。

パラメータとしてファイルを編集できるエディタが c:¥myedit¥myeditor.exe であるとします。c:¥files ディレクトリのファイル FILE1.C を開き、このエディタを起動するには、以下のように直接指定することもできます。

c:¥myedit¥myeditor.exe c:¥files¥FILE1.C

しかし、このエディタを介して任意のファイルを開きたいとき、上記コマンドは c:¥files¥file1.c を開くだけのものであるため、問題が起きます。指定したエディタを使うときにその時点で選んだファイルを開くことができるようにするには、特定のファイル名を一般的なプレースホルダに置き換えます。以下に例を示します。

c:¥myedit¥myeditor.exe $(FULLFILE)

これで、HEW がエディタでファイルを開くとき、$(FULLFILE)を選ばれたファイル名で置き換えます。

5.2 プレースホルダを挿入する

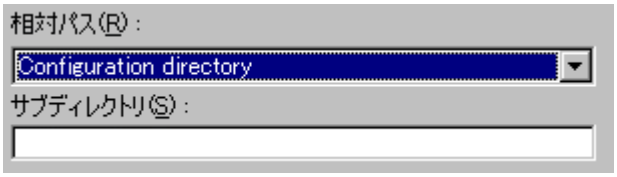
プレースホルダを挿入するには以下に示す方法があります。

例 1

1. プレースホルダを挿入したい場所にカーソルを置いてください。
2. 次に、必要なプレースホルダを編集フィールドの右のプレースホルダボタンをクリックしてください。
3. 必要なプレースホルダをポップアップメニューから選択してください。

例 2

1. “Custom directory”以外のプレースホルダをドロップダウンリストボックスから選択し、プレースホルダによって示されたディレクトリからの相対サブディレクトリを指定してください。
2. “Custom directory”を選択した場合、[サブディレクトリ]フィールドに絶対ディレクトリパスを指定してください。



例 3

1. プレースホルダを挿入したい場所にカーソルを置いてください。
2. 必要なプレースホルダをドロップダウンリストボックスから選択してください。
3. [挿入]ボタンをクリックしてください。

例 4

フィールドにプレースホルダを直接入力してください。大文字で入力して、“$(” で開始し“) ”で終了してください。

正しい

$(FILEDIR)

誤り

$(Filedir)

$(FILEDIR)

$FILEDIR

5.3 使用できるプレースホルダ

以下の表に、使用できるプレースホルダと意味、またその使用例を示します。

プレースホルダ	意味	プレースホルダの展開例
$(FULLFILE)	ファイル名 (フルパスを含む)	c:¥hew¥workspace¥project¥file.src
$(FILEDIR)	ファイルディレクトリ	c:¥hew¥workspace¥project
$(FILENAME)	ファイル名 (パスを除き拡張子を含む)	file.src
$(FILELEAF)	ファイル名（パスと拡張子を除く）	file
$(EXTENSION)	ファイルの拡張子	src
$(WORKSPDIR)	ワークスペースディレクトリ	c:¥hew¥workspace
$(WORKSPNAME)	ワークスペース名	workspace
$(PROJDIR)	プロジェクトディレクトリ	c:¥hew¥workspace¥project
$(PROJECTNAME)	プロジェクト名	project
$(CONFIGDIR)	コンフィグレーションディレクトリ	c:¥hew¥workspace¥project¥debug
$(CONFIGNAME)	コンフィグレーション名	debug
$(HEWDIR)	HEW インストールディレクトリ	c:¥hew
$(TCINSTALL)	ツールチェインインストールディレクトリ（オプション設定ダイアログボックス、および[新規ビルドフェーズ]ダイアログボックス（ステップ 3）上）	c:¥hew¥tools¥renesas¥sh¥9_0_1_1
$(TOOLDIR)	ツールインストールディレクトリ（Tools Administration 上）	c:¥hew¥tools¥renesas¥sh¥9_0_1_1
$(TEMPDIR)	テンポラリディレクトリ	c:¥temp
$(WINDIR)	Windows® ディレクトリ	c:¥windows
$(WINSYSDIR)	Windows® システムディレクトリ	c:¥windows¥system
$(EXEDIR)	コマンドディレクトリ	v:¥vc¥win32
$(USERNAME)	ユーザログイン（バージョン管理）	JHARK
$(PASSWORD)	ユーザパスワード（バージョン管理）	214436
$(VCDIR)	「仮想的」バージョン管理ディレクトリ	“c:¥project” は “x:¥vc¥project” へマッピングしている。
$(COMMENT)	コメント（バージョン管理）	“Please Enter Comment” ダイアログボックスを表示する。
$(LINE)	エラー/ウォーニングの行番号	12

上記の表では以下のことを仮定しています。

- ファイルパスは “c:¥hew¥workspace¥project¥file.src”。
- ワークスペース名 “workspace”の位置は “c:¥hew¥workspace”。
- プロジェクト名 “project” の位置は “c:¥hew¥workspace¥project”。
- コンフィグレーション名 “debug” にはコンフィグレーションディレクトリがあり、位置は “c:¥hew¥workspace¥project¥debug”。
- HEW.EXE を“c:¥hew”にインストールしている。
- ツールチェイン（コンパイラ、アセンブラ、リンケージエディタ）の*.HRF ファイルの位置は “c:¥hew¥tools¥renesas¥sh¥9_0_1_1”
  このディレクトリは[ビルド]メニューのオプション設定ダイアログボックス、および[新規ビルドフェーズ]ダイアログボックス（ステップ 3）上では$(TCINSTALL)として参照され、[ツールアドミニストレーション]ダイアログボックス上では$(TOOLDIR)として参照される。
- Windows®オペレーティングシステムを c:¥Windows にインストールしており、Windows®システムファイルを c:¥Windows¥System にインストールしている。
- バージョン管理実行可能パスが v:¥vc¥win32¥ss.exe である。バージョン管理システムにログインするユーザ名は JHARK でパスワードが 214436 であり、バージョン管理実行

可能ファイルへのコマンドラインには$(COMMENT)を指定している。c:¥project は[ツール->バージョン管理->構成]を選択すると表示される[バージョン管理設定]ダイアログボックスの[Projects]タブの x:¥vc¥project にマッピングしている。

- コンパイラまたはアセンブラのエラーが 12 行目で発生した。

注：

どのフィールドでもすべてのプレースホルダを使用できるとは限りません。例えば、プレースホルダ$(LINE) は依存ファイル位置を指定するときには意味をもちません。プレースホルダ$(USERNAME)、$(PASSWORD)、$(VCDIR)、$(COMMENT)はバージョン管理でのみ受け付けます。各編集フィールドで、使用できないプレースホルダを指定すると、警告メッセージを表示する場合があります。

5.4 プレースホルダを使うにあたって

プレースホルダによって、システムが使用する様々なファイルへのパスをフレキシブルに指定できます。

- パスまたはファイル名を入力する編集フィールドの右にプレースホルダボタン( )がある場合、プレースホルダをどのように使ってパスやファイルの指定をフレキシブルにできるかご考慮ください。
- いくつかの構成を使うとき、プレースホルダ$(CONFIGDIR) を使うと、現在の構成のディレクトリからファイルへの書き込みやそのファイルから現在の構成のディレクトリへの書き込みができます。
- できるだけプレースホルダを利用してください。プレースホルダは後で削除したり追加したりできます。

6. I/O ファイルフォーマット

HEW は、I/O レジスタ定義ファイルで取得する情報に基づいて、[IO]ウィンドウを作成します。デバッギングプラットフォームを選択すると、HEW は、選択したデバイスに対応する“<device>.IO”ファイルを検索し、存在する場合にはこのファイルをロードします。これは、I/O モジュール、およびそのレジスタのアドレスやサイズを記述するフォーマット済みテキストファイルです。ユーザはテキストエディタでこのファイルを編集し、ユーザアプリケーションに特有のメモリマップレジスタや周辺レジスタ（例えば、マイコンのアドレス空間にマップした ASIC デバイスのレジスタ）のサポートを追加できます。

ファイルフォーマット

各モジュール名を[Modules]定義セクションで定義し、モジュールの番号を、順番に付けていなければなりません。各モジュールはレジスタ定義セクションに対応しており、セクション内のエントリは、I/O レジスタを定義します。

[BaseAddress]はデバイスのための定義であり、そのデバイスでは、CPU モードによってアドレス空間の I/O レジスタの場所が移動します。この場合、[BaseAddress]値は、ある特有モードの I/O レジスタのベースアドレスです。また、レジスタ定義で使用するアドレスは、同じモードにおけるレジスタのアドレス位置です。I/O レジスタファイルを実際に使用する場合、定義したレジスタアドレスから[BaseAddress]値を引き、その結果のオフセットを選択したモードのベースアドレスに加算します。

各モジュールにはセクションがあり、オプションの依存性によって形成するレジスタを定義します。依存性は、モジュールが有効かどうかを確認するためにチェックします。各レジスタ名をセクションで定義し、レジスタの番号を、順番に付けていなければなりません。依存性は、dep=<reg> <bit> <value>のようにセクションに入力します。

1. <reg>は依存性のレジスタ ID です。
2. <bit>はレジスタのビット位置です。
3. <value>は値で、ビットは、有効であるモジュールに使用しなければなりません。

[Register]定義エントリは、id=<name> <address> [<size>[<absolute>[<format>[<bitfields>]]]]のフォーマットで入力します。

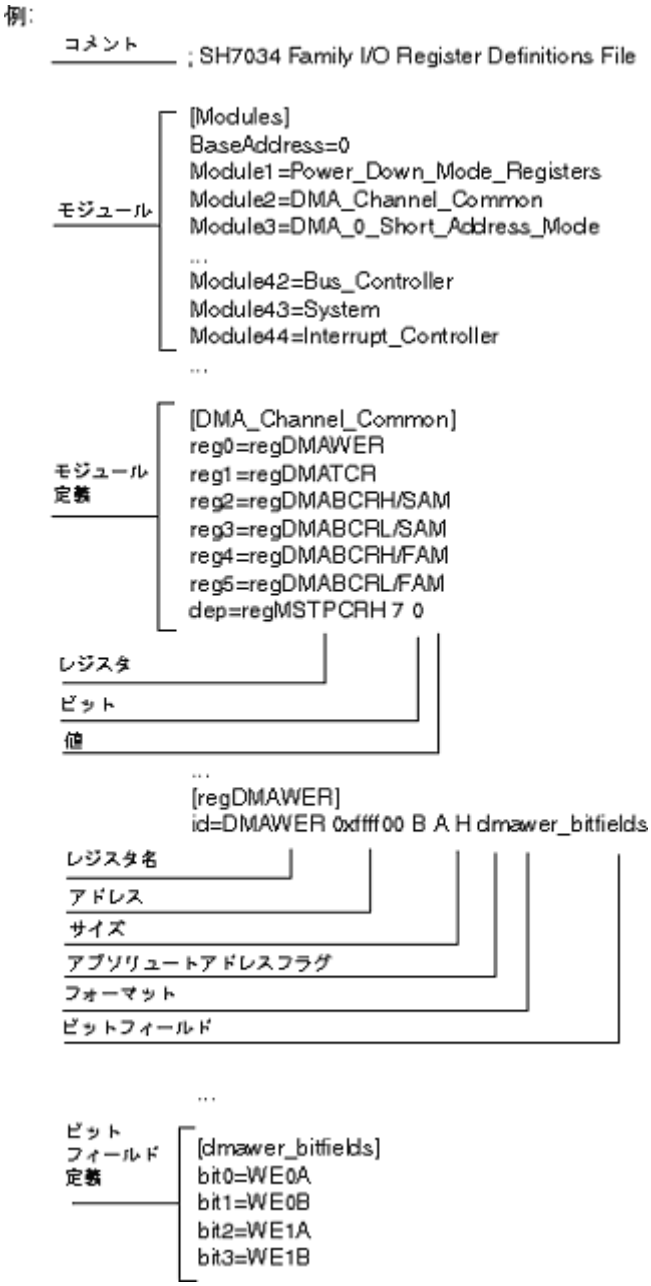
1. <name>は表示するレジスタ名です。
2. <address>はレジスタのアドレスです。
3. <size>は、B がバイトサイズ、W がワードサイズ、L がロングワードサイズを意味します（デフォルトはバイトです）。
4. <absolute>は、レジスタが絶対アドレスにある場合、A と設定します。これは、異なるモードの CPU によって I/O 空間アドレス範囲が移動する場合のみ関連します。レジスタが絶対アドレスにあると定義すると、ベースアドレスオフセットは計算せず、指定したアドレスを直接使用します。
5. <format>はレジスタを出力するためのフォーマットです。有効な値は、16 進数の場合は H、10 進数は D、2 進数は B です。
6. <bitfields>セクションは、レジスタのビットを定義します。

ビットフィールドセクションは、各エントリが bit<no>=<name>タイプのレジスタ内のビットを定義します。

1. <no>はビット番号です。
2. <name>はビットのシンボル名です。

コメント行を入れる場合、“;”で始めなければなりません。

次に例を示します。



7. シンボルファイルフォーマット

シンボルファイルを正確にデコードするためには、Pentica-B ファイルのフォーマットに基づいてファイルを作成する必要があります。

1. ファイルは、簡単な ASCII テキストファイルでなければなりません。
2. ファイルは、"BEGIN"で始めなければなりません。
3. 各シンボルは、個々の行で、まず、“H”で終了する 16 進数の値から始まり、次にスペース、シンボルテキストの順でなければなりません。
4. ファイルは、"END"で終了しなければなりません。

例：

```
BEGIN
11FAH Symbol_name_1
11FCH Symbol_name_2
11FEH Symbol_name_3
1200H Symbol_name_4
END
```

注：

機能のサポートは、デバッギングプラットフォームに依存します。

8. キーボードのショートカット

HEW のアプリケーションでは、主要なコマンドはすべてキーボードから操作できます。アプリケーションでのキーボードのコマンドを以下に示します。

機能キー	キー	機能
Alt	0-9	キーボードショートカットに割り当てたマクロを実行する。
Alt	K	ワークスペースウィンドウを開く。
Alt	O	アウトプットウィンドウを開く。
Alt	F4	アプリケーションを終了する。
Alt	Backspace	元に戻す(または、エディタで Ctrl+Z)。
Ctrl	0-9	テンプレートで予約されています。
Ctrl	A	エディタですべて選択する（他のウィンドウでも有効な場合あり）。
Ctrl	B	[ブレークポイント]ダイアログボックスを開く。
Ctrl	C	コピーする。
Ctrl	D	逆アセンブリウィンドウを開く。
Ctrl	F	[検索]ダイアログボックスを開く。
Ctrl	G	ジャンプする。
Ctrl	H	置換する。
Ctrl	I	IO ウィンドウを開く。
Ctrl	K	スタックトレースウィンドウを開く。
Ctrl	L	コマンドラインウィンドウを開く。
Ctrl	M	メモリウィンドウを開く。
Ctrl	N	エディタで新しいソースファイルを開く。
Ctrl	O	[ファイルを開く]ダイアログボックスを開く。
Ctrl	P	印刷する。
Ctrl	R	レジスタウィンドウを開く。
Ctrl	S	現在のファイルを保存する。
Ctrl	U	ステイタスウィンドウを開く。
Ctrl	V	貼り付ける。
Ctrl	X	切り取る。
Ctrl	Y	やり直す。
Ctrl	Z	元に戻す。
Ctrl	F2	ブックマークを設定、解除する。
Ctrl	F3	エディタで選択されているテキストの次を検索、または[検索]ダイアログボックスを開く。
Ctrl	F4	エディタウィンドウを閉じる。
Ctrl	F6	エディタでウィンドウを切り替える。
Ctrl	F7	ファイルをビルドする。
Ctrl	F9	ブレークポイントの設定、解除する。
Ctrl	Backspace	単語を削除する。
Ctrl	Break	ビルドを停止する。
Ctrl	Insert	コピーする。
Ctrl	Tab	エディタを切り替える。
Ctrl	Home	エディタのカーソルを現在のファイルの先頭に戻す。
Ctrl	End	エディタのカーソルを現在のファイルの最後に移動する。
Ctrl	↑	ウィンドウを上へスクロールする（カーソル位置はエディタの同じ場所）。

Ctrl	↓	ウィンドウを下へスクロールする（カーソル位置はエディタの同じ場所）。
Ctrl	←	エディタで、カーソルを 1 つ前の単語に移動する。
Ctrl	→	エディタで、カーソルを次の単語に移動する。
Ctrl+Alt	1-4	バーチャルデスクトップを切り替える。
Ctrl+Alt	PageUp	次のタブに移動する（例えばアウトプットウィンドウ）。
Ctrl+Alt	PageDown	1 つ前のタブに移動する（例えばアウトプットウィンドウ）。
Ctrl+Shift	8	エディタで、空白文字を表示する。
Ctrl+Shift	G	画像ウィンドウを開く。
Ctrl+Shift	K	TCL ツールキットを開く。
Ctrl+Shift	L	エディタでの行を削除する。
Ctrl+Shift	M	括弧の組み合わせを選択する。
Ctrl+Shift	S	すべて保存する。
Ctrl+Shift	T	テンプレートを挿入する。
Ctrl+Shift	U	エディタで選択されたテキストをすべて大文字にする。
Ctrl+Shift	V	波形ウィンドウを開く。
Ctrl+Shift	Y	PC 位置を表示する。
Ctrl+Shift	F3	エディタで選択されているテキストの前を検索、または[検索]ダイアログボックスを開く。
Ctrl+Shift	Tab	エディタを切り替える。
Ctrl+Shift	Home	カーソル位置からファイルの先頭までを選択する。
Ctrl+Shift	End	カーソル位置からファイルの最後までを選択する。
Ctrl+Shift	←	エディタで、1 つ前の単語を選択する。
Ctrl+Shift	→	エディタで、次の単語を選択する。
Shift	F1	コンテキスト依存ヘルプを表示する。
Shift	F2	前のブックマークへジャンプする。
Shift	F3	前を検索する。
Shift	F5	リセット後実行する。
Shift	F8	前のビルドエラーまたはウォーニングが発生したエディタを表示する。
Shift	F10	コンテキストメニューを表示する（ポップアップメニューでのマウスの右クリックと同様）。
Shift	F11	ステップアウト実行する。
Shift	F12	すべてのウィンドウをリフレッシュする。
Shift	Delete	エディタで切り取り操作する(またはエディタで Ctrl+X)。
Shift	ESC	アウトプットウィンドウを非表示にする。
Shift	Tab	エディタで、タブの位置を戻す。
Shift	Home	エディタで、カーソル位置から現在の行の先頭までを選択する。
Shift	End	エディタで、カーソル位置から現在の行の最後までを選択する。
Shift	↑	その行より上を選択する。
Shift	↓	その行より下を選択する。
Shift+Alt	Backspace	やり直す (または、エディタで Ctrl+Y)。
なし	F1	コンテキスト依存ヘルプを表示する。
なし	F2	次のブックマークへジャンプする。
なし	F3	次を検索する。
なし	F4	ファイル間で検索する。
なし	F5	実行する。
なし	F7	ビルドする。
なし	F8	次のビルドエラーまたはウォーニングが発生したエディタを表示する。
なし	F9	ブレークポイントを挿入/削除する。
なし	F10	ステップオーバ実行する。
なし	F11	ステップイン実行する。
なし	F12	ウィンドウをリフレッシュする。
なし	Delete	クリアする。

なし	Insert	挿入モードと上書きモードを切り替える。
なし	Enter	エディタでのキャリッジリターンを挿入する。
なし	Tab	エディタでタブを挿入する。
なし	ESC	停止する。
なし	Home	エディタで、カーソルを現在の行の先頭に移動する。
なし	End	エディタで、カーソルを現在の行の最後に移動する。
なし	PageUp	エディタのページを上に移動する。
なし	PageDown	エディタのページを下に移動する。
なし	↑	エディタで、カーソルを上に移動する。
なし	↓	エディタで、カーソルを下に移動する。
なし	←	エディタで、カーソルを左に移動する。
なし	→	エディタで、カーソルを右に移動する。
なし	Alt + マウスで選択	エディタでカラムを選択する。

注：

機能のサポートは、デバッギングプラットフォームに依存します。

9. HEW デバッガでドラッグアンドドロップする

HEW デバッガを使用中、各デバッグコンポーネントを他のコンポーネントと相互作用させることができます。他の表示への対象のドラッグで実現します。

以下に例を示します。

1. ラベル表示から他のデバッグ表示へラベルをドラッグできます。例えば、[逆アセンブリ]ウィンドウにラベルをドラッグした場合、そのラベルのあるアドレスへスクロールします。
2. 変数をエディタから[ウォッチ]ウィンドウへドラッグできます。変数が[ウォッチ]ウィンドウに追加されます。
3. 関数名をエディタから逆アセンブリへドラッグすると、逆アセンブリ表示はラベルの位置へジャンプします。

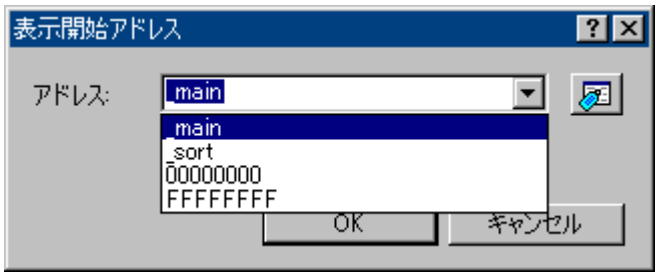
10. ラベルを使用してコードを参照する

デバッグモジュール内を検索するとき、ラベルを使用すると便利です。アドレスを入力できるエディットフィールドであれば、ラベルを使用することが可能です。このようなフィールドにラベルを入力すると、内蔵の式処理がラベルをチェックし、アドレスに変換します。これにより、ユーザは"_main+100"や"_MyFunction+100"などの評価を入力できます。

つまり、ラベルを使用するとき、評価する対象のアドレスは固定されていません。この方法は特に、コマンドラインバッチファイルを使ってブレークポイントを設定するときに便利です。コマンドラインバッチファイルで常に関数にブレークポイントを設定する必要がある場合、ラベルを使うことでこれを解決できます。

ラベルを使えば、バッチファイルの内容に影響を与えずにコードを変更できます。

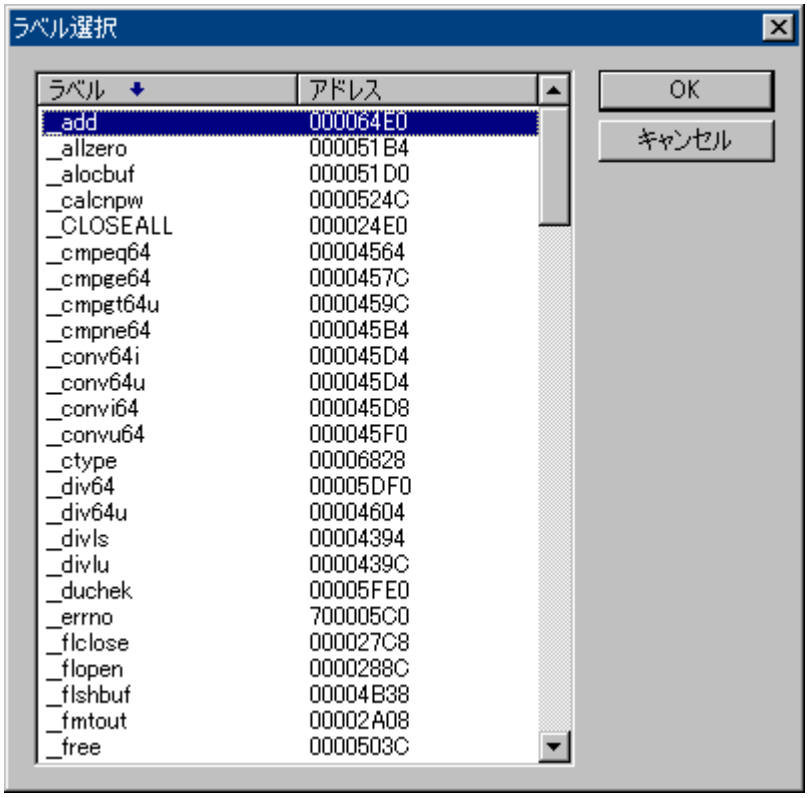
HEW V.3.1 以降では、他にもラベルを使用する簡単な方法をサポートしています。



上記のダイアログボックスに、ラベルを選択するリストも表示します。HEW のアプリケーション全体を通して、アドレスフィールドに入力された過去 20 件を、このリストに保存します。つまり、同じラベルを何度も入力する場合、この最近使用したアドレスエディットフィールドのリストを使えば、より効率的に入力できます。入力を評価するアドレスエディットフィールドの場合に、この操作が可能です。

アドレスを入力できるエディットフィールドの右にラベルブラウズボタン( )が付いています。

これをクリックすると、[ラベル選択]ダイアログボックスが表示され、現在のラベルのリストからラベルを選択できます。

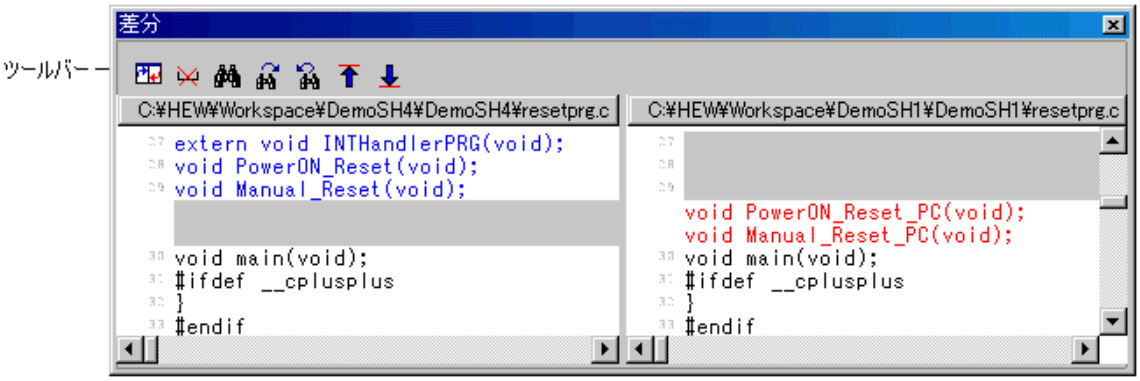


リストはラベル名の昇順でソートされ右側にはそのアドレス値を表示しています。各々のカラムヘッダをクリックすると、クリックしたカラムでソートできます。

11. ウィンドウのツールバー

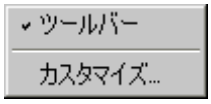
HEW4.0 以降のコンポーネントのウィンドウでは、クライアント領域にツールバーを表示しています。ツールバーボタンをクリックすると、ウィンドウの主な機能へアクセスできます。本機能は、HEW の様々なウィンドウにあります。

例として、[差分]ウィンドウを以下に示します。

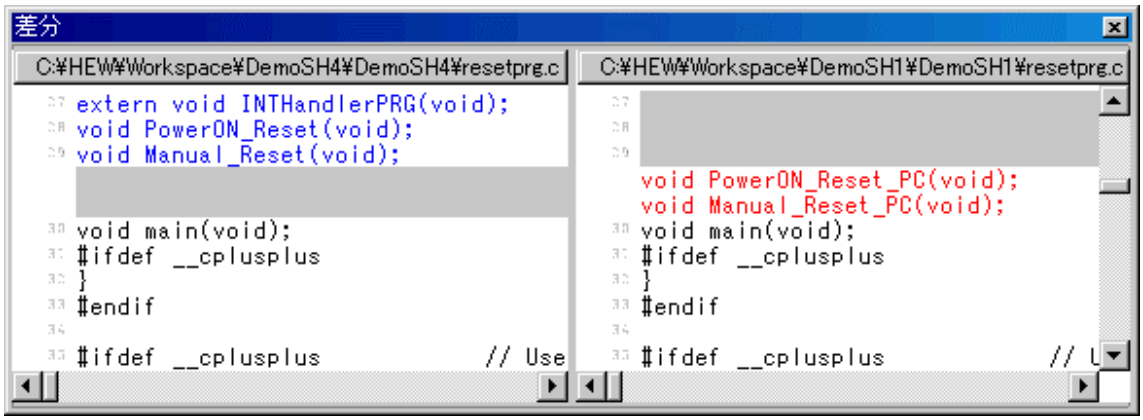


ツールバーをカスタマイズすることも可能です。

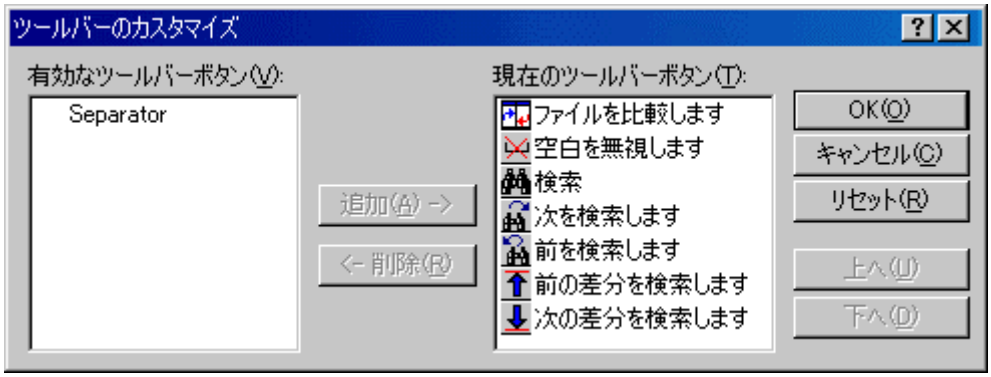
ツールバー上で右クリックすると、以下のメニューを表示します。



[ツールバー]メニューは、ツールバーの表示/表示なしを切り替えることができます。（下の図は表示なしの場合）



[カスタマイズ]メニューは、[ツールバーのカスタマイズ]ダイアログボックスを表示します。これを以下に示します。



このダイアログボックスでは、表示するボタンを選択したり、ボタンの表示順序を変更できます。[有効なツールバーボタン]は、ツールバーに現在表示されていないボタンをすべて表示します。[現在のツールバーボタン]は、ツールバーに現在表示されているボタンをすべて表示します。

ツールバーにボタンを追加するには

1. [有効なツールバーボタン]リストから、追加したいツールバーボタンを選択してください。
2. [追加]ボタンをクリックしてください。
3. [OK]ボタンをクリックしてください。

ツールバーのボタンを移動するには

1. [現在のツールバーボタン]リストから、移動したいツールバーボタンを選択してください。
2. 移動させたい位置まで、[上へ]あるいは[下へ]ボタンをクリックしてください。
3. [OK]ボタンをクリックしてください。

ツールバーからボタンを削除するには

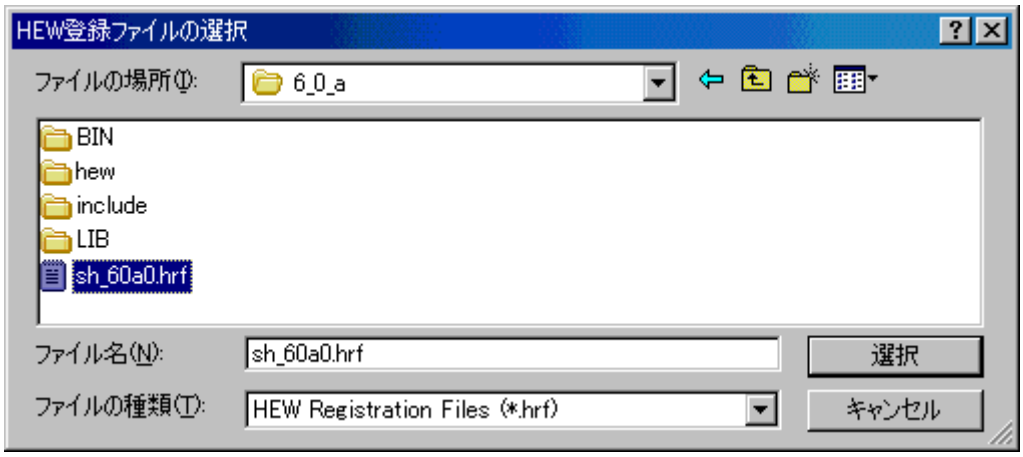
1. [現在のツールバーボタン]リストから、削除したいツールバーボタンを選択してください。
2. [削除]ボタンをクリックしてください。
3. [OK]ボタンをクリックしてください。

12. HEW1.x 用のツールチェインでビルドする

HEW1. x で作成したプロジェクトを、HEW2. x 以降に登録されている新しいツールチェインにアップグレードしないで使う場合、該当する旧バージョンのツールチェインを登録する必要があります。[ツール->アドミニストレーション]の[登録]ボタンで旧バージョンのツールチェインの”HRF”ファイルを選択してください（下の図参照）。HEW2. x 以降で、旧バージョンのツールチェインを使ったビルド実行が可能になります。

ただし、一度 HEW2.x 以降で開いたワークスペースは、HEW1. x では開けませんのでご注意願います。

また、HEW2. x 以降では、新規に旧バージョンのプロジェクトは作成できません。HEW1. x にある旧バージョンのプロジェクトを作成する場合には、HEW1. x をご使用ください。



13. HMAKE ユーザガイド

13.1 コマンドライン

ファイルで hmake プログラムを実行するときに使うコマンドラインを説明します。

基本構成

コマンドラインのシンタックスを以下に示します。

hmake <メイクファイル> <パラメータリスト>

拡張子なしでファイルを指定した場合、.mak を付加します。パラメータリストには、次の節に示す複数のパラメータを指定できます。また、パラメータは省略できます。パラメータリストはメイクファイル名の前に指定してもかまいません。各パラメータは、1 つ以上の空白文字で区切ってください。パラメータは、大文字と小文字を区別しません。パラメータとファイル名を指定しなかった場合、ヘルプ情報を表示します。

Exit コード

実行中のメイクファイルでシンタックスエラー発生時、またはメイクファイル実行中の処理で不当エラーコードが返されたとき、hmake はコード 1 で終了します。その他の場合、hmake はコード 0 で終了します（ファイルのシンタックスや exit コード条件の指定方法は、以下を参照してください）。

パラメータ

指定できるパラメータとその機能を示します。

パラメータ	機能
/A	入出力ファイル状態にかかわらず全コマンドを実行する。全ビルドと同じ
/N	入出力ファイルの使用状態により、（ノーマルで）実行すべきコマンドを判断して、表示するが、実行はしない
/?	ヘルプ情報の表示

13.2 ファイルのシンタックス

hmake ファイルでは、基本的に 4 種類の文（変数宣言、記述部、コメント、メッセージコメント）を使用します。hmake ファイルの作成にあたって、これらの文はどのような順序でも使用できます。ただし、変数を記述部や他の変数宣言に使用する前に、変数宣言をしてください。nmake ファイルで使用する最初の“all”文は、hmake ファイルでは必要ありません。メイクファイルでは、コマンドを出現する順序で実行します。“→”文字は、メイクファイルのシンタックスを正しくするためにタブ文字を使用すべきところを示すことに注意してください。

変数宣言

変数宣言をした後、その変数は、hmake ファイルのすべての部分で使うことができます。変数宣言のシンタックスを以下に示します。

<変数名> = <値>

変数名と'='符号の間や、値と'='符号の間に、複数の空白文字を入れることができます。値は、'¥'文字で区切って、数行にわたって書くことができます。メインテキスト内で値に'¥'文字を含む場合は文字どおり解釈されます。'¥'文字の後に新しい行が続くときのみ、値が次の行にわたっているとみなされます。変数宣言の例をいくつか示します。

```
EXECUTABLE = c:¥dir¥prog.exe
OUTPUT = c:¥dir2¥file1.out
INPUT = c:¥dir2¥file1.c
DEPEND = c:¥dir2¥file2.h ¥
        c:¥dir2¥file3.h ¥
        c:¥dir2¥file4.h
```

hmake ファイルの後半で変数を使うには、変数名の前に"$("を、後ろに")"を加えてください。"$()"文字付きの変数名を変数の値で置き換えます。次の記述部の節にその例を示します。変数名には、半角英数文字と半角下線を使用できます。別の変数宣言のなかで変数を使うことができますが、すべての変数は、使う前に宣言します。

13.3 記述部

概要

記述部には、ターゲット、依存ファイル、そしてターゲットが依存ファイルより新しい場合に実行するコマンド一覧を指定します。ターゲット、依存ファイルのどちらか、またはいずれもない場合、コマンドを常に実行します。コマンドを常に実行したいときには、依存ファイルを指定する必要はありません。記述部のシンタックスを以下に示します。

<ターゲット 1> <ターゲット 2> … : <依存ファイル 1> <依存ファイル 2> …

→<コマンド 1>

→<コマンド 2>

→…

→<コマンド n>

最後のターゲットと':' 文字の間や、最初の依存ファイルと':'文字の間に、複数の空白文字を入れることができます。最初のターゲットの前に空白文字を入れないでください。各ターゲットと各依存ファイルは、少なくとも 1 つの空白文字で区切ってください。タブ文字はコマンドを含む行のはじめに必要です。変数宣言の後で、変数を上記シンタックスで記述部に使うことができます。

記述部の例（変数宣言の後に変数を使う例）を以下に示します。

```
c:¥dir1¥file1.obj : c:¥dir1¥file1.c c:¥dir1¥file1.h
→gcc c:¥dir1¥file1.c
$(OUTPUT) : $(INPUT) $(DEPEND)
→$(EXECUTABLE) $(INPUT)
```

特殊なコマンド

記述ブロックで使用できる特殊なコマンドが 2 つあります。cd コマンドはカレントディレクトリを変更し、set コマンドはメイクファイル実行の間に使用する環境変数を設定します。両方とも、相当する DOS コマンドと同じ方法で使用します。

これらのコマンドを使用した有効な記述ブロックの例を、以下に示します。

```
CHANGEDIR :
→ cd c:¥dir1¥dir2
SETENV:
→ set VAR1=value1
→ set VAR2=value2
→ set VAR3=value3
```

$(EXECUTABLE)を置き換える CHANGEDIR と SETENV がファイル名でなくてもかまいません。これらは存在しないファイルとして扱われてコマンドを常に実行します。

サブコマンドファイル

hmake でサブコマンドファイルを生成するには、記述部のコマンド部分を以下のように指定してください(上記の<コマンド n>を置き換えます)。

```
→<コマンド開始> <<
→<サブコマンド 1>,
→<サブコマンド 2>,
→…
→<サブコマンド n>
→<<<コマンド終了>
```

これで、Windows®の一時的ファイルに、<サブコマンド 1>、<サブコマンド 2>などの行を含むサブコマンドファイルを生成します。このコマンドファイルはメイク処理が終了すると削除します。コマンドファイル名は 2 つの”<<”の間のすべてのテキストで置き換えられます。サブコマンドファイル名は、自動的に hmake により付けられます。

例：

```
c:¥dir1¥file1.obj : c:¥dir1¥file1.c c:¥dir1¥file1.h
→gcc @”<<
→-c -o c:¥dir1¥file1.obj c:¥dir1¥file1.c
<<”
```

生成されたサブコマンドファイル名が“c:¥temp¥hmk111.cmd”の場合、hmake は以下を実行します。(c:¥dir1¥file1.obj が更新されるとき) :

```
gcc @”c:¥temp¥hmk111.cmd”
```

サブコマンドファイル(c:¥temp¥hmk111.cmd)には以下を含みます。

```
-c -o c:¥dir1¥file1.obj c:¥dir1¥file1.c
```

記述部に 1 つ以上のコマンドを含むことができ、標準とサブコマンドファイルコマンドを混在して使用できます。

13.4 コメント

‘#’文字はコメントを示します。この文字が行の最初にあると、（次の改行文字があるまで）その行を無視します。コメントの例を次に示します。

```
# My hmake file
# Variable declaration
OUTPUT= c:¥dir1¥file1.obj
# Descriptor
$(OUTPUT) : c:¥dir1¥file1.c c:¥dir1¥file1.h
→set VAR1=value1
→gcc c:¥dir1¥file1.c
```

hmake ファイルではコメントには専用の行が必要です。コメントを他の文の最後に付けることはできません。

13.5 メッセージコマンド

メッセージコマンドは、メイクファイル実行中にテキストの行を標準出力に出力します。これらのテキスト行は、適切に実行されている実行ファイルの出力のなかで、メイクファイルに出現する順序で出力します。出力テキストのバッファリングは行いません。メッセージコマンドのシンタックスを以下に示します。

```
!MESSAGE <出力するテキスト>
```

改行文字が<出力するテキスト>の最後の文字の後にあると想定します。!MESSAGE と<出力するテキスト>の間の空白文字を無視します。メッセージコマンドの例を以下に示します。

```
!MESSAGE Executing C Compiler
```

High-performance Embedded Workshop V.4.02
ユーザーズマニュアル

発行年月日　2006年12月08日　　Rev.1.00

発行　　株式会社 ルネサス テクノロジ　営業企画統括部
　　〒100-0004　東京都千代田区大手町2-6-2

編集　　株式会社 ルネサス ソリューションズ　ツール開発部

© 2006. Renesas Technology Corp. and Renesas Solutions Corp., All rights reserved. Printed in Japan.

High-performance Embedded Workshop V.4.02
ユーザーズマニュアル

ルネサス エレクトロニクス株式会社
神奈川県川崎市中原区下沼部1753 〒211-8668

RJJ10J1854-0100