オンチップデバッギングエミュレータ E1
導入ガイド＜RX610 グループ編＞

R20AN0022JJ0101
Rev.1.01
2010.09.07

要旨

本導入ガイドは、オンチップデバッギングエミュレータ E1 を初めて使用されるお客様を対象とした導入ガイドです。E1 エミュレータを開封してからプログラム実行/停止に至るまでの過程を迷わず進められるように、できる限りシンプルに解説することを心掛けています。

目次

1. はじめに 2

2. 製品概要 3
2.1 E1 エミュレータに付属の製品 3
2.2 別途ご用意いただくもの 3

3. インストール 6
3.1 インストールの前に 6
3.2 エミュレータソフトウェアのインストール 6
3.3 RXファミリ用C/C++コンパイラパッケージのインストール 11
3.4 USBドライバのインストール 14
3.5 エミュレータデバッガおよびコンパイラのインストールの確認 15
3.6 USBドライバのインストールの確認 16

4. 使ってみましょう 18
4.1 エミュレータ使用手順 18
4.2 エミュレータの接続 19
4.3 High-performance Embedded Workshopの起動とチュートリアルの読み込み 19
4.4 E1 エミュレータ起動の設定 21
4.5 ソースファイルのオープン 24
4.6 ビルド（コンパイル＆リンク） 25
4.7 プログラムのダウンロード 26
4.8 CPUのリセット 26
4.9 プログラムの実行および停止 27
4.10 ブレーク機能 28
4.11 C言語の変数の参照/設定 31
4.12 メモリウィンドウによるデータの参照／設定 39
4.13 レジスタウィンドウによるレジスタ値の参照／設定 42
4.14 I/OウィンドウによるI/Oレジスタ値の参照／設定 43
4.15 パフォーマンス解析 46
4.16 トレース機能 49
4.17 デバッグの終了 54
4.18 ユーザボード単体での動作確認 55

5. 補足事項 56
5.1 エミュレータ関連ドキュメント 56
5.2 無償評価版RXファミリ用C/C++コンパイラパッケージ 58
5.3 E20 エミュレータに関して 59

6. よくある質問 60

1.　はじめに

このアプリケーションノートでは、E1 エミュレータを単体でご購入いただいた方が、ユーザシステムとして RX610 グループのマイコンを搭載した CPU ボードを使用することを前提に記載しておりますが、別の RX610 グループのマイコンを搭載した別の基板に対しても同様の操作手順で E1 エミュレータを使用することができます。

本書に沿って手順を進めることにより、プログラムインストールから簡単な使い方までを実践することができます。

なお、本書の解説には、以下の機器を用いて解説しています。

(1) ホスト PC
(2) E1 エミュレータ
(3) RX610 搭載　CPU ボード（型名：R0K556100C000BE）
Renesas Starter Kit for RX610 (型名：R0K556100S000BE) に付属しています。

2. 製品概要

E1 エミュレータシステムのシステム構成を図 2.1に示します。

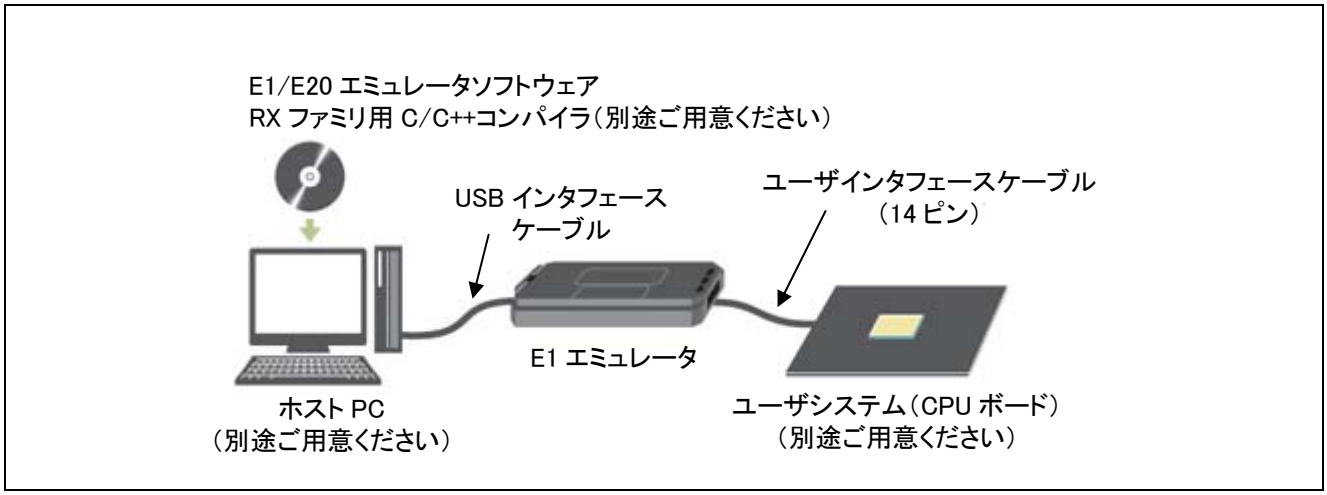


図 2.1 エミュレータシステムの構成

2.1 E1 エミュレータに付属の製品

2.1.1 E1 エミュレータ

ルネサス製オンチップデバッギングエミュレータです。High-performance Embedded Workshop および E1/E20 エミュレータデバッガと組み合わせて、効率のよいデバッグ環境をご提供します。

また、デバッグ終了時には、フラッシュ開発ツールキット（Flash Development Toolkit）と組み合わせて使用することにより、フラッシュライタとしてご使用いただくことができます。

2.1.2 USB インタフェースケーブル

E1 エミュレータとホスト PC を接続するケーブルです。

2.1.3 ユーザインタフェースケーブル

E1 エミュレータとユーザシステムを接続するユーザインタフェースケーブルです。

2.1.4 E1/E20 エミュレータソフトウェア

E1/E20 エミュレータソフトウェアには以下のソフトウェアが含まれています。

(1) High-performance Embedded Workshop

High-performance Embedded Workshop は、C コンパイラ／アセンブラ／エミュレータソフトウェア／エディタなどのツール群を共通 GUI に統合して、ソフトウェアの開発効率を改善するためのツールです。E1 エミュレータを使用したデバッグ操作などは、この GUI 上から操作することになります。

(2) E1/E20 エミュレータデバッガ

E1/E20 エミュレータデバッガはホスト PC 上で動作するソフトウェアです。エミュレータデバッガは E1 エミュレータを介してマイコンと通信を行い、高機能なデバッグ環境を提供します。エミュレータデバッガは、High-performance Embedded Workshop に組み込まれて動作しますので、お客様が直接エミュレータデバッガを意識することはありません。以下 E1 エミュレータデバッガの特徴を示します。

- プログラムの実行および停止、ブレーク機能、ウォッチ機能などの基本デバッグ機能
- 256 分岐またはサイクルのトレース情報取得
- 2 点間のプログラム実行時間を計測可能な時間計測機能(パフォーマンス計測機能)
- サイクルスチール方式を採用した、メモリウィンドウの自動更新機能

これらのデバッグ機能を使用することにより、デバッグ時の問題点の解析や評価を容易に行うことができ、デバッグ期間の大幅な短縮に貢献できます。

(3) オートアップデートユーティリティ

オートアップデートユーティリティは、ルネサスマイクロコンピュータ開発環境のアップデート情報をインターネット経由で自動的に収集し、必要なアップデートを確実に行う手段を提供するツールです。

詳細に関しては、オートアップデートユーティリティインストール後、図 2.2のメニューから、「Renesas AutoUpdate添付資料」を参照してください。

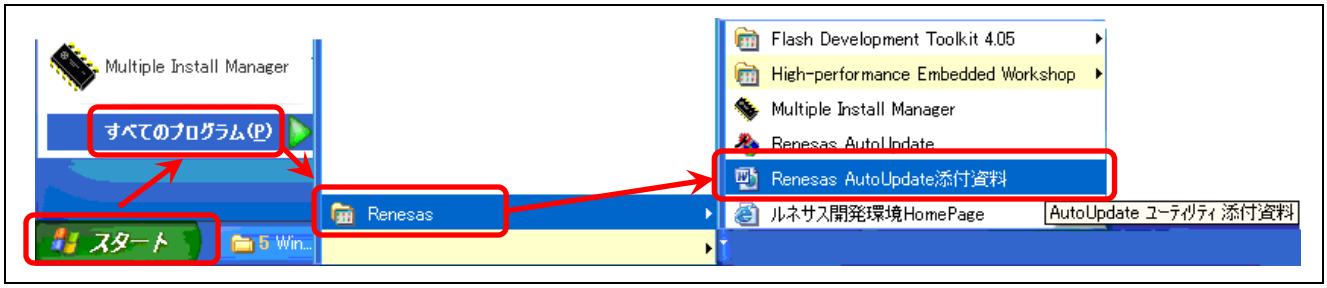


図 2.2　Renesas AutoUpdate 添付資料の参照

2.2　別途ご用意いただくもの

以下の機器および製品については別途ご用意ください。

- ホスト PC
- ユーザシステム（CPU ボード）
- RX ファミリ用 C/C++コンパイラパッケージ（※１）

※1　RXファミリ用C/C++コンパイラパッケージには、製品版と無償評価版があります。無償評価版Cコンパイラについては、「5.2　無償評価版RXファミリ用C/C++コンパイラパッケージ」を参照ください。

2.2.1　ホスト PC

E1 エミュレータ付属の各ソフトウェアは、表 2.1に示すOSバージョンおよびホストPC上での動作を推奨しています。

表 2.1　ホスト PC の動作環境

項番	項目	動作環境	
		Windows®2000 Windows® XP 32 ビット版	Windows Vista® 32 ビット
1	ホスト PC	USB2.0(フルスピード/ハイスピード)【注意】を備えた IBM PC/AT 互換機	
2	CPU	Pentium® 3　1GHz 以上を推奨	Pentium® 4　3GHz 以上または、 Core™2Duo　1GHz 以上を推奨
3	最小稼動メモリ容量	推奨 1GB 以上（+ロードモジュールのファイルサイズの 10 倍以上）	推奨 1.5GB 以上（＋ロードモジュールのファイルサイズの 10 倍以上）
4	ハードディスク容量	インストールディスク容量 600MB 以上 （スワップ領域を考慮してメモリ容量の 2 倍以上（推奨 4 倍以上）の空き容量をご用意ください）	
5	マウスなどの ポインティングデバイス	ホストマシン本体に接続可能で Windows® 2000、Windows® XP および Windows Vista®に対応している、マウスなどのポインティングデバイス	
6	ディスプレイ	モニタ解像度 1024 × 768 以上	
7	CD-ROM ドライブ	E1 エミュレータソフトウェアのインストール、または E1 エミュレータユーザーズマニュアルを参照するために必要	

注意：E1 エミュレータは USB1.1 のホストマシンにも接続できますが、本製品を快適にご使用になるため USB2.0（ハイスピード）に対応したホストマシンをご使用ください。

2.2.2 ユーザシステム

E1 エミュレータを使用する場合には、マイコンの登載されたユーザシステムが必ず必要です。ユーザシステムには、E1 エミュレータ接続用のコネクタも用意されている必要があります。

本書ではユーザシステムの代わりに Renesas Starter Kit for RX610 (型名：R0K556100S000BE) に含まれている CPU ボード (型名：R0K556100C000BE) を使用します。

RX610 グループの搭載されている他の CPU ボードでも同様の手順で進められるように説明しています。

2.2.3 RX ファミリ用 C/C++コンパイラパッケージ

E1 エミュレータには、コンパイラはバンドルされていません。ソースファイルをコンパイルするために、RX ファミリ用 C/C++コンパイラパッケージが必要です。

「RX ファミリ用 C/C++コンパイラパッケージ」は C 言語・アセンブリ言語ソースプログラムからデバッグ情報ファイルを生成します。

3. インストール

E1 エミュレータを使用するためには以下の製品をインストールする必要があります。

- E1/E20 エミュレータソフトウェア（E1 に付属の CD によりインストール）
  本製品には、E1 エミュレータデバッガ、統合開発環境 High-performance Embedded Workshop、およびオートアップデートユーティリティが含まれています。
- RX ファミリ用 C/C++コンパイラパッケージ
  本製品には、C/C++コンパイラ一式、シミュレータデバッガ、統合開発環境 High-performance Embedded Workshop、およびオートアップデートユーティリティが含まれています。

3.1 インストールの前に

(1) E1 エミュレータは、エミュレータソフトウェアのインストール前にパソコンと接続しないでください。
エミュレータソフトウェアのインストールにより E1エミュレータ用の USB ドライバがパソコンにインストールされ、新しいハードウェア検出時の導入が自動的に行われるようになります。
エミュレータソフトウェアのインストール前に E1 エミュレータをパソコンに接続すると、E1 エミュレータ用の USB ドライバが存在しないため、E1 エミュレータを認識できません。この場合、Windows のデバイスマネージャを使用して、不明な USB デバイスを削除してください。その後、再度付属ソフトウェアをインストールし、E1 エミュレータを接続しなおしてください。

(2) コンパイラパッケージの製品版と無償評価版の両方をお持ちの場合は、製品版を利用してください。
既に RX シリーズ用 C コンパイラパッケージの製品版をインストール済みの場合、または本製品と併せて同コンパイラパッケージ製品版をご購入の場合、製品版をお使いください。

3.2 エミュレータソフトウェアのインストール

(1) E1 エミュレータ購入時に付属されている CD-ROM を読み込むと、インストール案内が記載されている画面（HTML 形式）が自動的に表示されます。(表示されない場合は CD-ROM 内の“README_J.HTM”を実行してください)
表示されましたら、図 3.1中に示した[インストール]をクリックしてください。

(2) 図 3.1のようにフォルダが表示されますので、【HewInstMan,exe】を選択し実行してください。

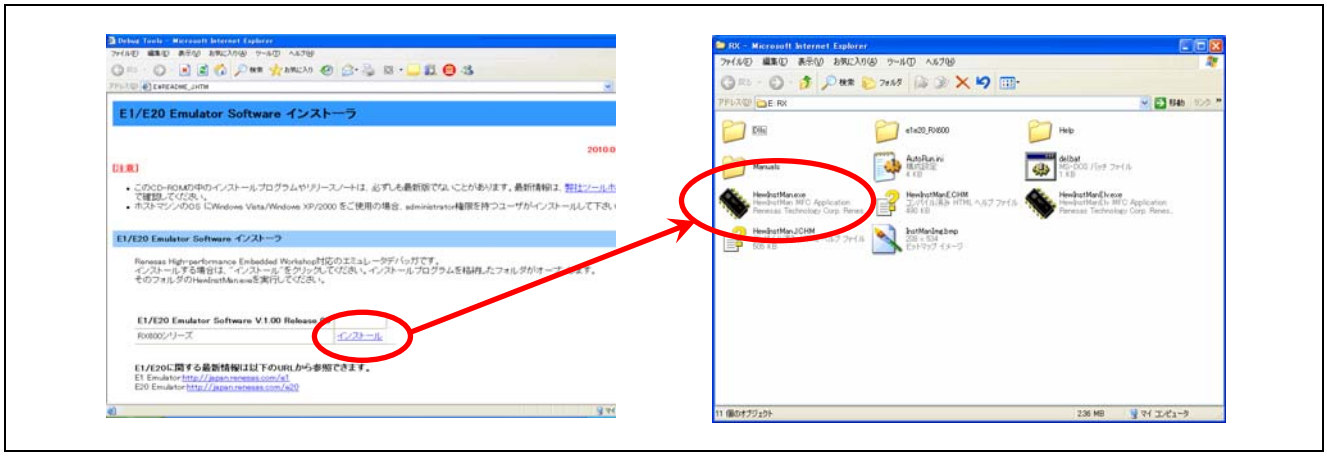


図 3.1 インストール案内画面

(3) 【HewInstMan,exe】を実行すると、図 3.2のように[High-performance Embedded Workshopインストールマネージャ]画面が表示されます。
初めてインストール作業を行う場合[インストールマネージャ ヘルプ]も表示されますので内容を確認してください。[インストールマネージャ ヘルプ]の確認後、標準インストール(推奨)のボタンを押してください。

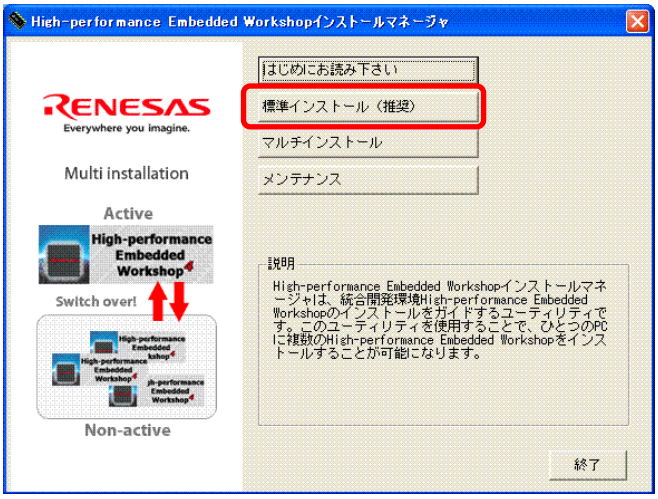


図 3.2 High-performance Embedded Workshop インストールマネージャ

備考 本資料では、標準インストールを使用する方法について説明していますが、マルチインストールを使用することで、複数の High-performance Embedded Workshop 環境を 1 台の PC に構築することも可能です。

(4) 図 3.3の「インストール先の選択」ダイアログボックスが表示されます。インストール先のフォルダを変更する場合は、[変更]ボタンを押して指定してください。ファイルをインストールするフォルダを確認して[次へ]ボタンを押してください。本書ではデフォルトのインストール先フォルダを利用します。

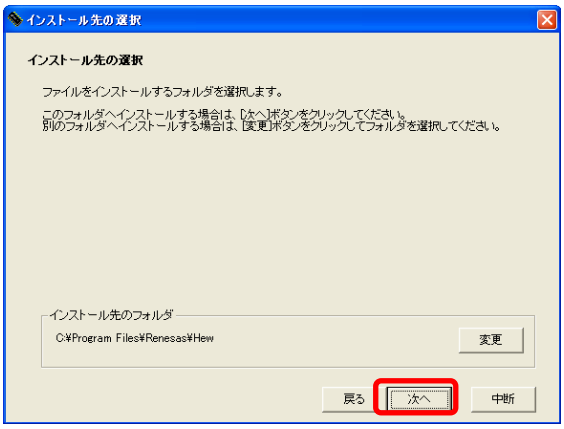


図 3.3 インストール先の選択

(5) 図 3.4のように[インストール製品の選択]ダイアログボックスが表示されたら、インストールするソフトウェア製品を選択して[インストール]ボタンを押してください。
本書では、ルネサスマイクロコンピュータ開発環境のアップデート情報をインターネット経由で自動的に収集し、必要なアップデートを確実に行う手段を提供するオートアップデートツールも選択します。

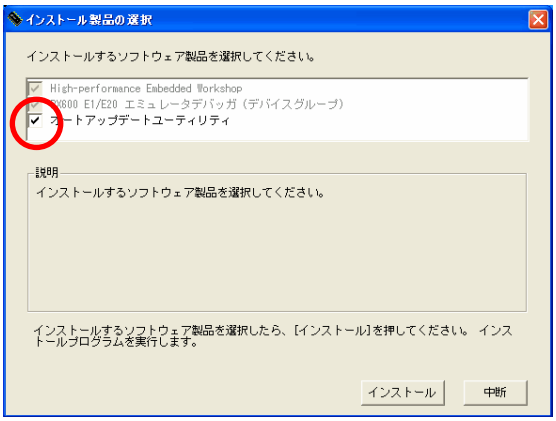


図 3.4　インストール製品の選択(ソフトウェア)

(6) High-performance Embedded Workshopのインストールが自動的に開始されますので、図 3.5のようにインストールを行ってください。

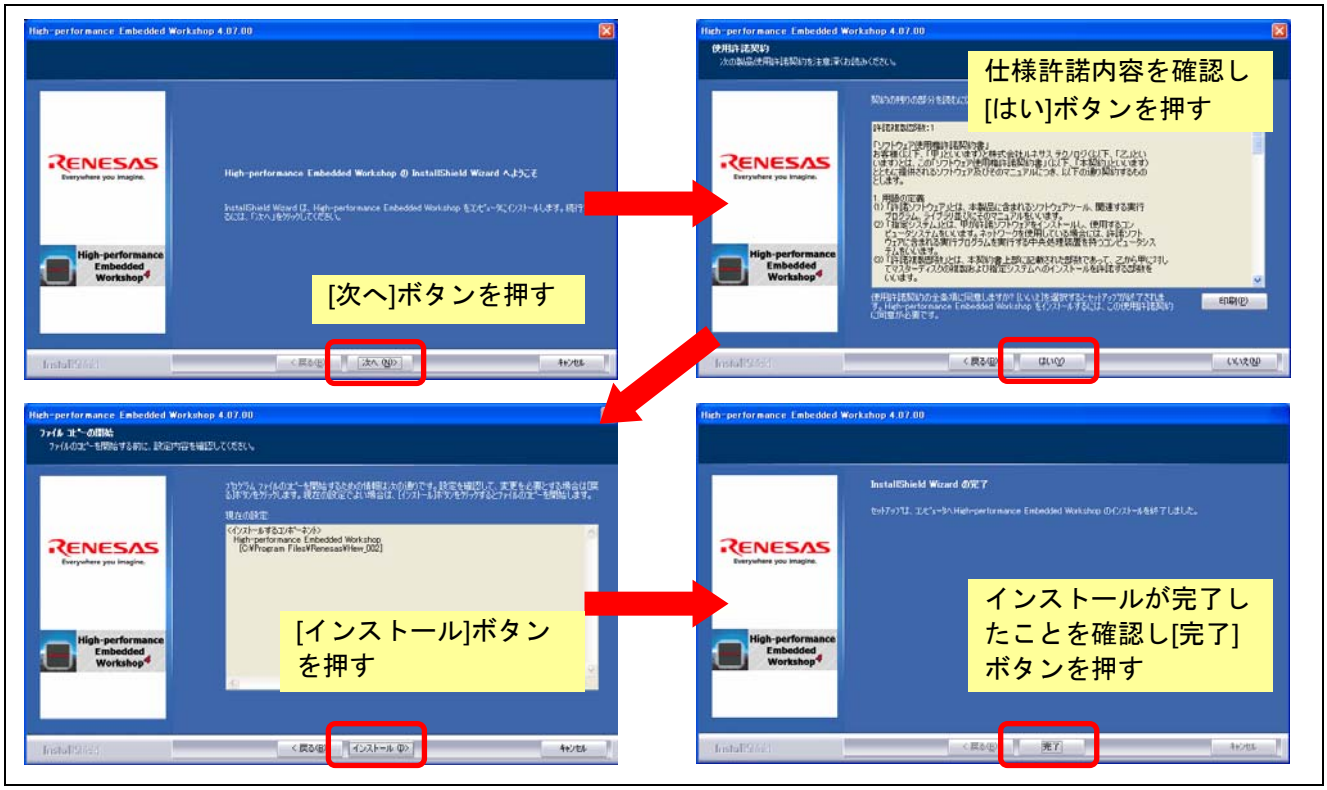


図 3.5 High-performance Embedded Workshop のインストール

(7) 続いてE1/E20 エミュレータソフトウェアのインストールが開始しますので、図 3.6の手順でインストールを行ってください。

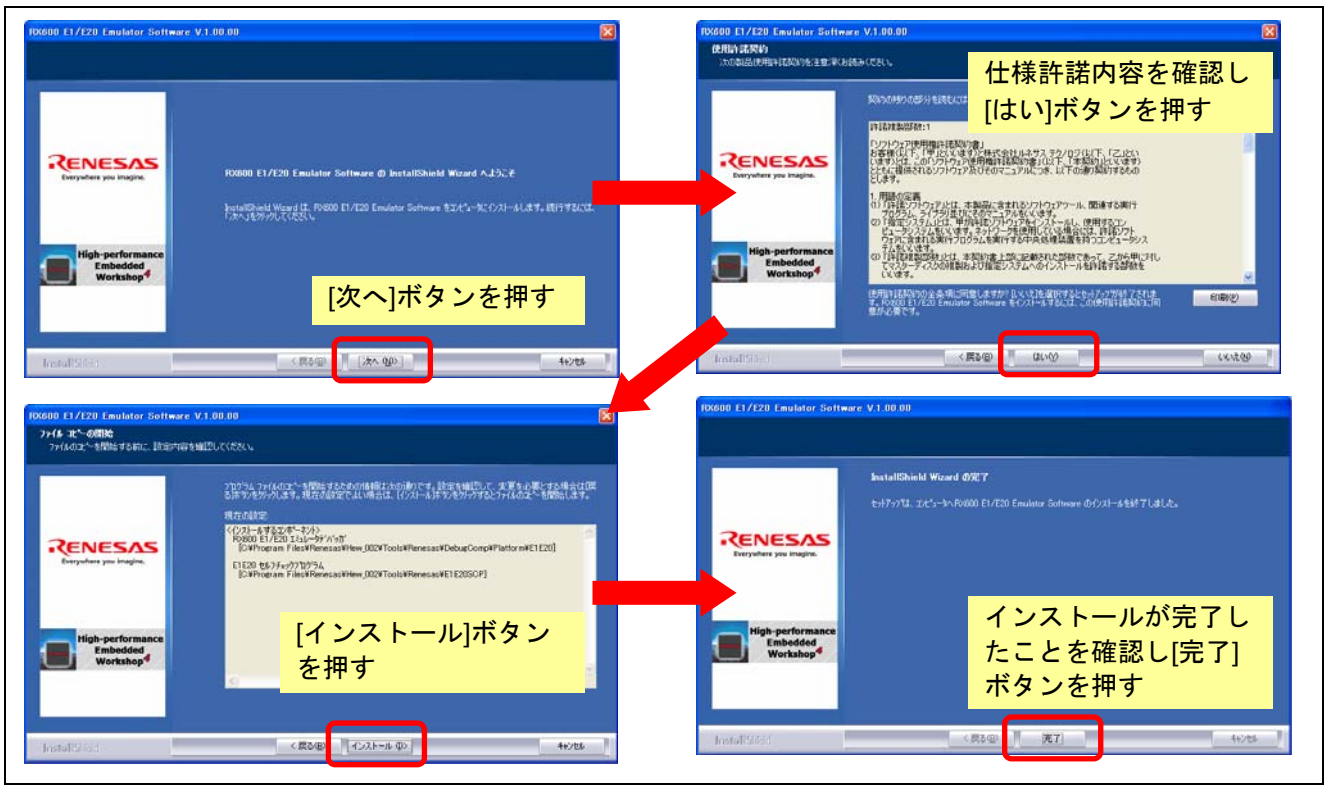


図 3.6 E1/E20 エミュレータソフトウェアのインストール

(8) 次にオートアップデートユーティリティ（AutoUpdate Utility）のインストールが開始されますので、図 3.7 の手順でインストールを行ってください。

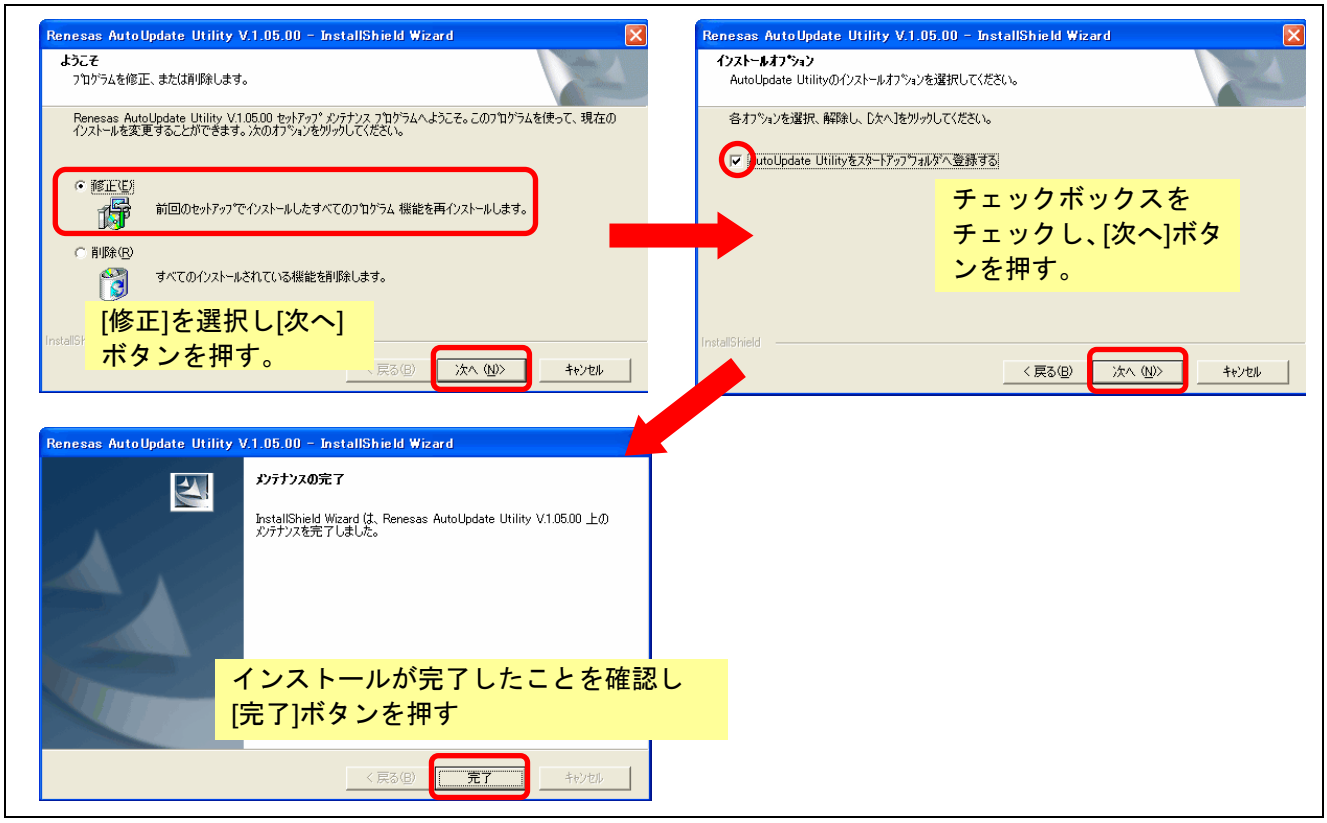


図 3.7　オートアップデートユーティリティのインストール

(9) すべてのソフトウェアのインストールが完了すると、図 3.8のダイアログが表示されます。この後、「ツール製品のユーザ登録のご案内」ボタンを押して、ルネサスのWebサイトからユーザ登録をすることもできます。本資料では、ユーザ登録手順の説明は省略し、[終了]ボタンを押してインストールを終了します。

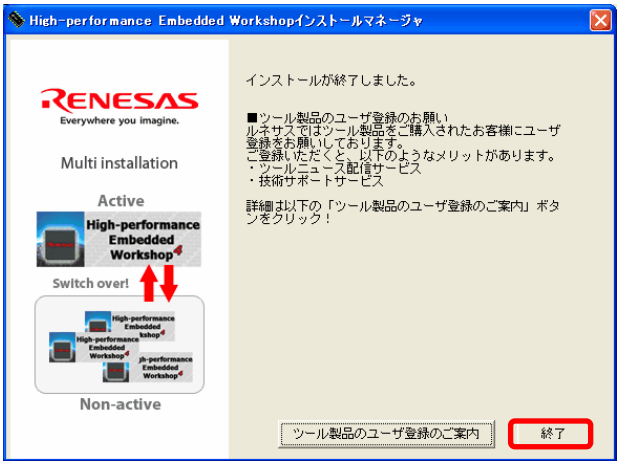


図 3.8　インストール終了

3.3 RX ファミリ用 C/C++コンパイラパッケージのインストール

(1) 無償評価版Cコンパイラをインストールする場合は、ダウンロードしたexeファイル（例：ccrxv100r00_ev.exe）をダブルクリックしてください。図 3.9の画面が開きますので、[Next]ボタンを押してください。製品版をインストールする場合は、CDをドライブにセットすることで、図 3.10の画面からインストールが開始します。
無償評価版Cコンパイラのダウンロードについては、「5.2.2 無償評価版Cコンパイラのダウンロード方法」を参照してください。

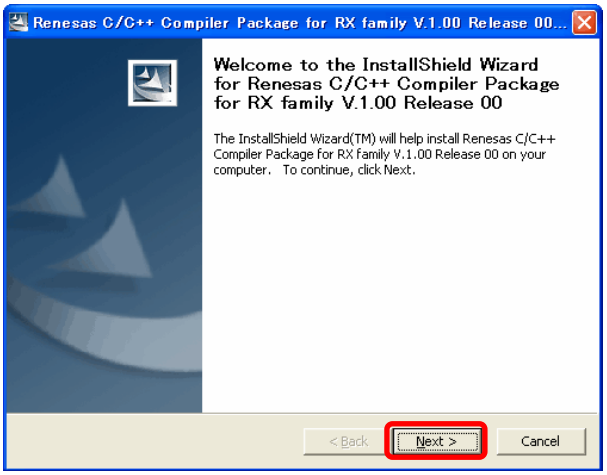


図 3.9 無償評価版 C コンパイラのインストール開始画面

(2) 図 3.10が表示されたら、「標準インストール（推奨）」を選択します。

図 3.10 High-performance Embedded Workshop インストールマネージャ

(3) インストールするソフトウェア製品を選択して[インストール]ボタンを押してください。
「RX ファミリ用 C/C++コンパイラパッケージ」の選択は必須です。
オートアップデートユーティリティのインストールは任意です。すでにオートアップデートユーティリティがホスト PC インストールされている場合は、ここでインストールする必要はありませんが、ここでインストールを選択しても、新しいほうのバージョンのオートアップデートユーティリティがホスト PC に残りますので、問題はありません。

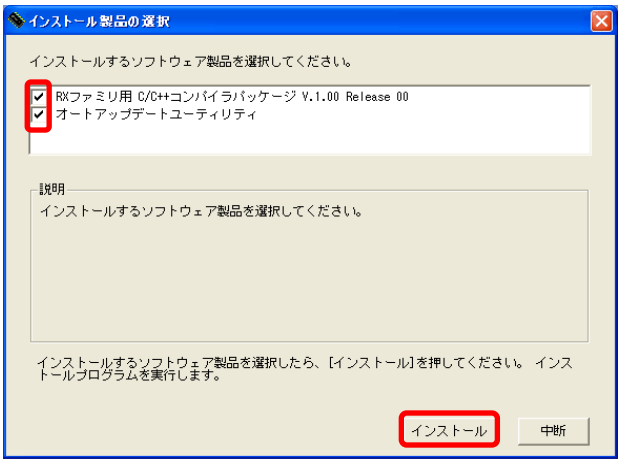


図 3.11　インストール製品の選択

(4) C/C++コンパイラパッケージのインストールが開始されますので、図 3.12に従ってインストールを行ってください。

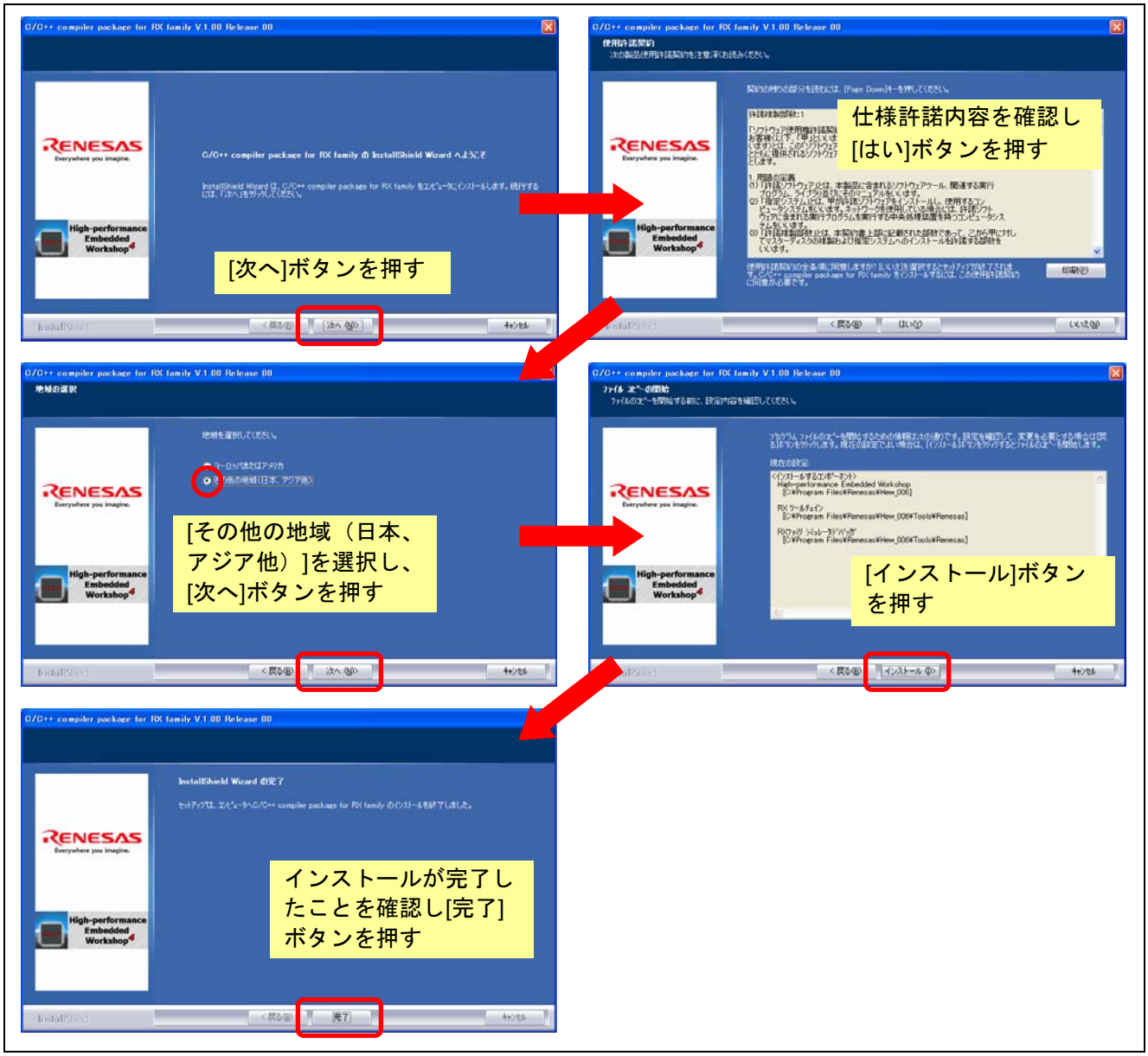


図 3.12　C/C++コンパイラパッケージのインストール

(5) オートアップデートユーティリティのインストールを選択していた場合、自動的にオートアップデートユーティリティのインストールが開始されますので、指示に従ってインストールしてください。オートアップデートユーティリティのインストール手順については、ここでは省略します。

(6) インストールが完了すると、図 3.13のダイアログが表示されます。[終了]ボタンを押してインストールを終了します。

図 3.13　インストールの終了

3.4　USB ドライバのインストール

(1) E1 エミュレータ本体とホストPCをUSBケーブルで接続してください。図 3.14のダイアログが表示されますので、説明に従いUSBドライバのインストールを行って下さい。
本図は Windows XP の例です。OS により画面が異なります。

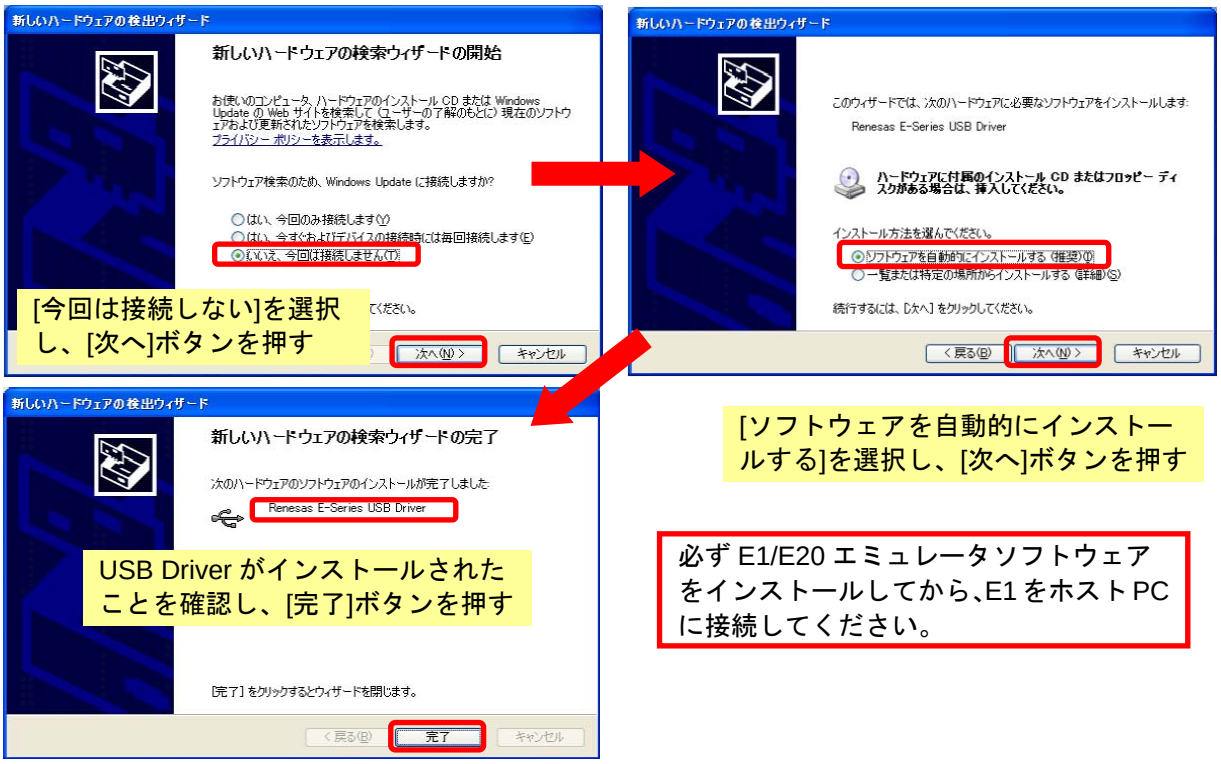


図 3.14　新しいハードウェアの検索ウィザード

3.5　エミュレータデバッガおよびコンパイラのインストールの確認

以下の手順に従い、エミュレータデバッガおよびコンパイラが正しくインストールされているかを確認してください。

(1) スタートメニューから図 3.15のように[Multiple Install Manager]を選択してください。

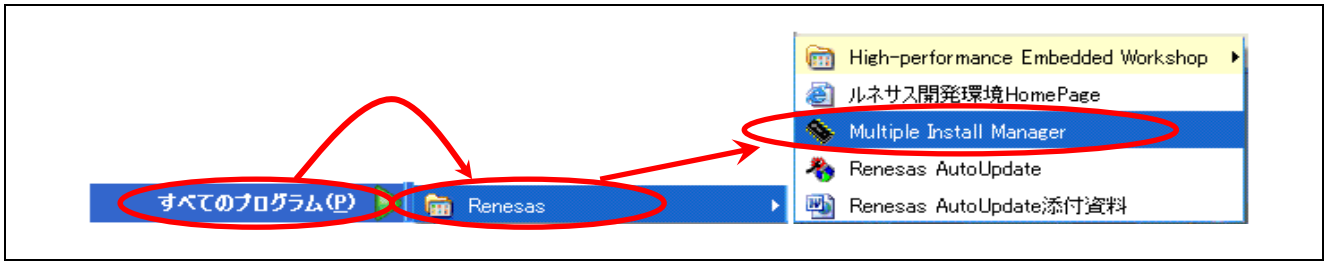


図 3.15　Multiple Install Manager の起動

(2) [Multiple Install Manager]を起動すると、図 3.16のようにメンテナンス画面が表示されます。構成を確認し、必要なインストールが完了しているか確認します。確認できなかった場合は、再度インストールを行ってください。

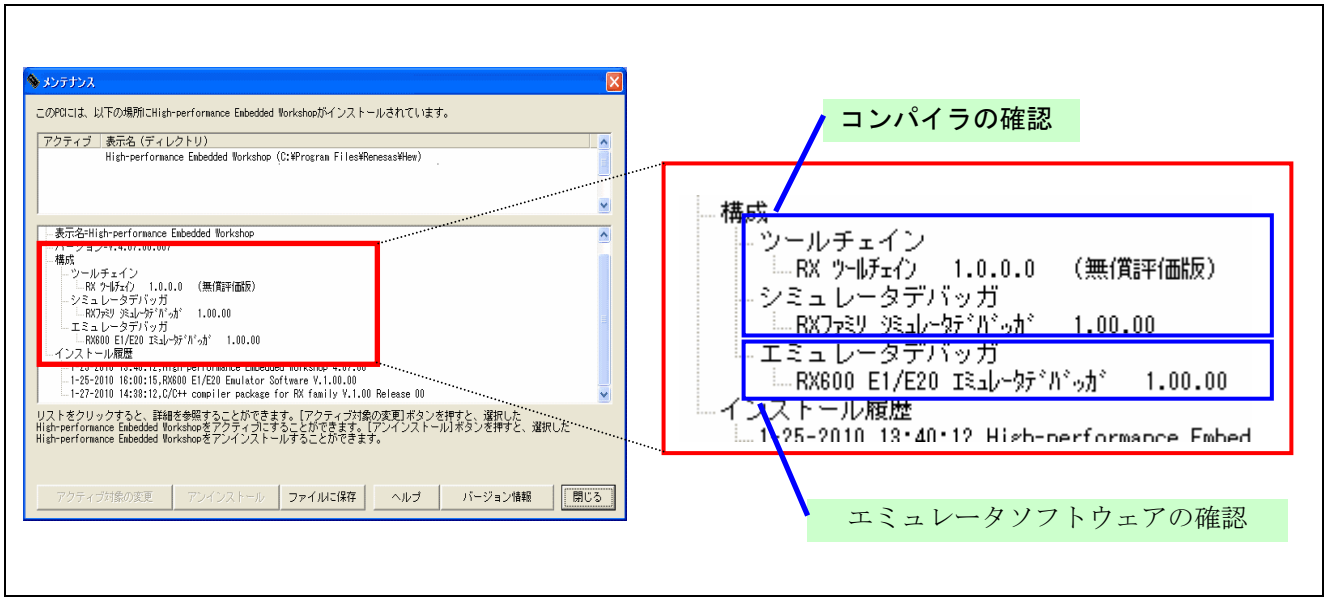


図 3.16　インストールされている製品の確認

3.6 USB ドライバのインストールの確認

エミュレータとホスト PC を USB インタフェースケーブルで接続した後に、E1 エミュレータの ACT ランプの状態を確認することで、USB ドライバが正しく動作しているかを確認することが可能です。

表 3.1にACTランプと通信状態の関係および確認項目を示します。

表 3.1 エミュレータ接続状況と確認項目

ACT ランプの状態	PC とエミュレータ間の通信状態	確認項目
点灯	E1 エミュレータが使用可能な状態	USB ドライバは正しくインストールされています。
点滅	USB ドライバが認識されていないため、E1 エミュレータが認識されていない状態	USB ドライバのインストールができているかを確認してください。(以降の説明を参照してください)
消灯	通信ができていない状態	USB ケーブルが正しく接続されているか（緩んでいないか）、ホスト PC およびエミュレータに電源が入っているか、各機器が壊れていないかを確認してください。

USB ドライバが正しくインストールされているかどうかは以下の手順で確認できます。

(1) マイコンピュータのプロパティを表示します。(システムのプロパティ画面)

(2) システムのプロパティ画面から[ハードウェア]タブを開き、図 3.17のように[デバイスマネージャ]ボタンを押してください。

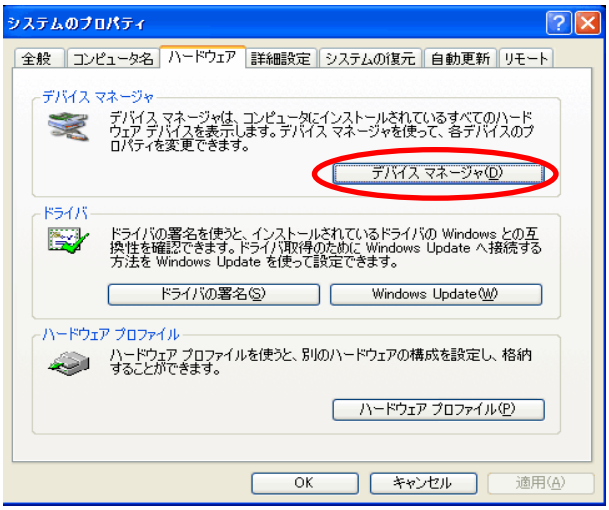


図 3.17 システムのプロパティ画面

(3) 図 3.18のように【Renesas Emulator】カテゴリおよび【Renesas E-Series USB Driver】の項目があることを確認します。

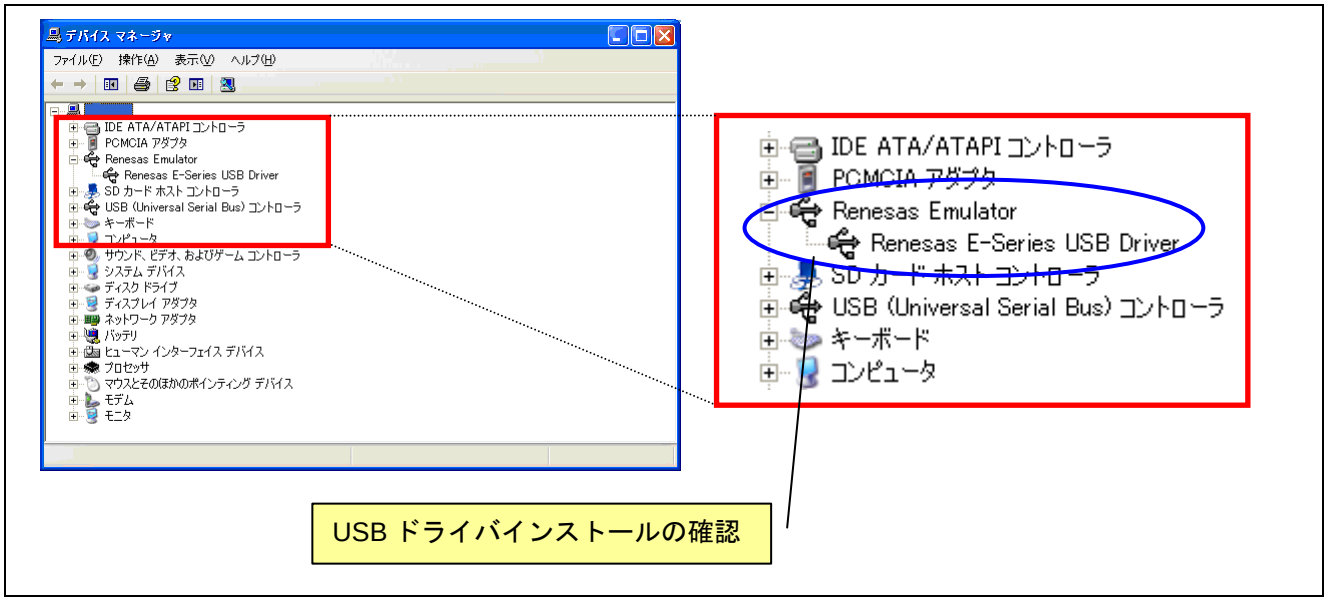


図 3.18　デバイスマネージャ画面

【Renesas Emulator】カテゴリ内に【！】または【？】マークのアイコンが付いている場合があります。その場合には、ドライバのインストールに失敗している可能性がありますので、【！】や【？】となっているドライバを一旦削除した上で、「3.1　インストールの前に」を確認し、再度インストール作業を行ってください。

図 3.19　USB ドライバが正常にインストールされていない場合の表示例

【Renesas Emulator】項目を確認できなかった場合は、ドライバインストール前にエミュレータソフトウェアのインストールの完了を確認した上で、再度「3.4　USBドライバのインストール」を行ってください。

【Renesas Emulator】カテゴリ内に【Renesas E-Series USB Driver】を確認できたが、ACT ランプが点滅している状態で E1 エミュレータが使用できない場合は、一度 Renesas Emulator のドライバを削除し、再度インストール作業を行ってください。

4. 使ってみましょう

本章では、E1 エミュレータソフトウェアインストール時にホスト PC へインストールされるチュートリアルプログラムを用いて、E1 エミュレータの基本的な使い方について説明します。

4.1 エミュレータ使用手順

本章では、High-performance Embedded Workshop の起動からチュートリアルプログラムのダウンロード、プログラム実行および E1 エミュレータの主なデバッグ機能の使用方法について説明します。

手順としては表 4.1のようになります。(必要なインストールに関しては、全て完了している前提とします)

表 4.1 チュートリアルプログラム実行手順

順番	操作	参照先
1	USB ケーブルを使用し、E1 エミュレータを PC に接続	4.2 エミュレータの接続
2	ユーザシステムを接続	
3	High-performance Embedded Workshop の起動	4.3 High-performance Embedded Workshop の起動とチュートリアルの読み込み
4	ワークスペースを開く	
5	E1 エミュレータ起動設定	4.4 E1 エミュレータ起動の設定
6	コンフィグレーションプロパティ	
7	ソースファイルのオープン	4.5 ソースファイルのオープン
8	ビルド（コンパイル＆リンク）	4.6 ビルド（コンパイル＆リンク）
9	プログラムのダウンロード	4.7 プログラムのダウンロード
10	CPU のリセット	4.8 CPUのリセット
11	プログラムの実行および停止	4.9 プログラムの実行および停止
12	ブレーク機能	4.10 ブレーク機能
13	C 言語の変数を参照/設定する	4.11 C言語の変数の参照/設定
14	メモリウィンドウによるデータの参照／設定	4.12 メモリウィンドウによるデータの参照／設定
15	レジスタの参照／設定	4.13 レジスタウィンドウによるレジスタ値の参照／設定
16	I/O レジスタの参照／設定	4.14 I/Oウィンドウ
17	パフォーマンス解析	4.15 パフォーマンス解析
18	トレース機能	4.16 トレース機能
19	デバッグの終了	4.17 デバッグの終了
20	ユーザボード単体での動作確認	4.18 ユーザボード単体での動作確認

4.2　エミュレータの接続

E1 エミュレータシステムの構成を図 4.1に示します。

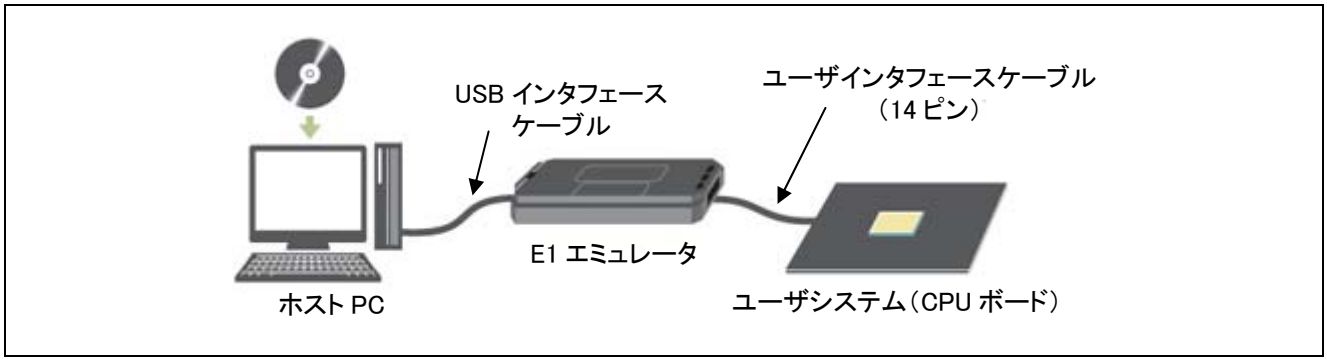


図 4.1　エミュレータシステムの構成

以下の手順でエミュレータ E1 システムを接続してください。

(1) E1 エミュレータとホスト PC を、USB ケーブルを使用して接続します。
(2) E1 エミュレータとユーザシステム（CPU ボード）を、ユーザインタフェースケーブルを使用して接続します。

注意：ユーザシステムに別途電源を供給する場合は、電源が供給されていないことを確認して、E1 を接続してください。

4.3　High-performance Embedded Workshop の起動とチュートリアルの読み込み

以下の手順で High-performance Embedded Workshop を起動し、チュートリアル用のワークスペースを読み込みます。

(1) 図 4.2のように[スタート]メニューの[すべてのプログラム]から[Renesas]→[High-performance Embedded Workshop]→[High-performance Embedded Workshop]と進み、起動します。

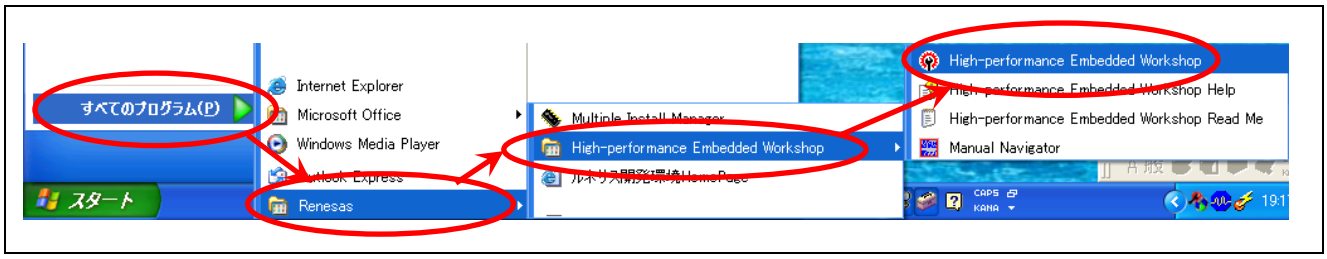


図 4.2　High-performance Embedded Workshop の起動

(2) 図 4.3のようにHigh-performance Embedded Workshop上に[ようこそ]ダイアログボックスが表示されます。[別のプロジェクトワークスペースを参照する]ラジオボタンを選択して[OK]ボタンを押してください。

(3) 図 4.3のように[ワークスペースを開く]ダイアログボックスが表示されます。
”Tutorial.hws”を指定し[選択]ボタンを押してください。
本製品のソフトウェアのインストールが完了している場合、標準では次のフォルダ位置にワークスペース ”Tutorial.hws” が格納されています。

```
<OS がインストールされているドライブ>:
¥WorkSpace¥Tutorial¥E1E20¥RX600¥Tutorial_LittleEndian
```

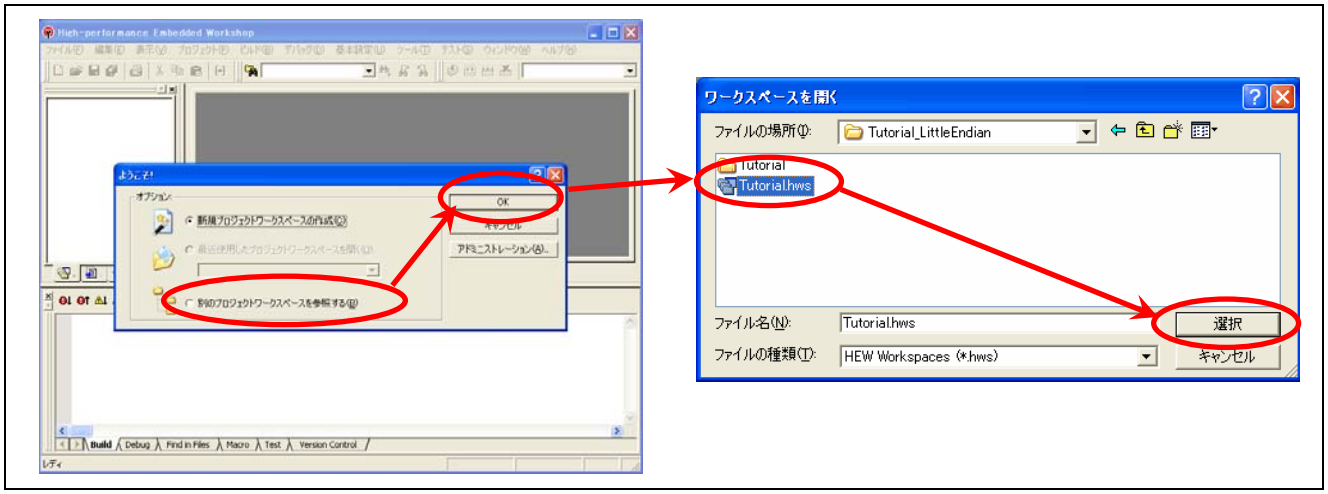


図 4.3 ようこそダイアログボックス選択

インストールされたチュートリアルワークスペースの作成時のHigh-performance Embedded Workshopのバージョンが、現在使用中のHigh-performance Embedded Workshopのバージョンよりも古い場合、図 4.4のダイアログボックスが表示されます。

その場合、現在の High-performance Embedded Workshop のバージョンとワークスペースのバージョンを合わせる必要があるため、ワークスペースを新しいバージョンにアップデートします。[OK]ボタンを押してください。

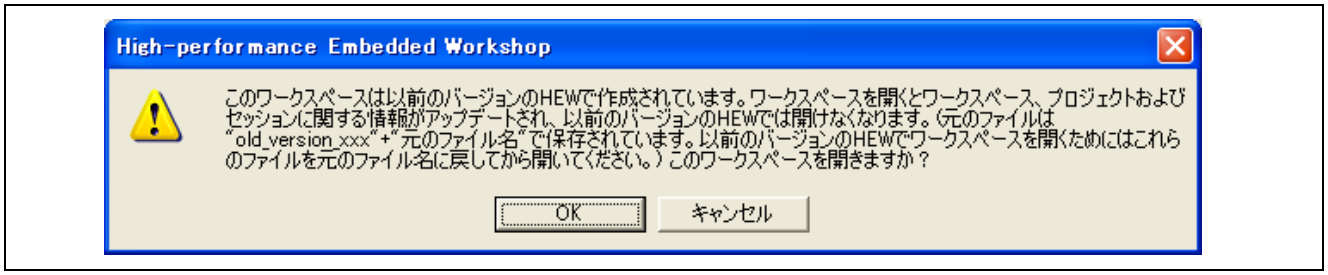
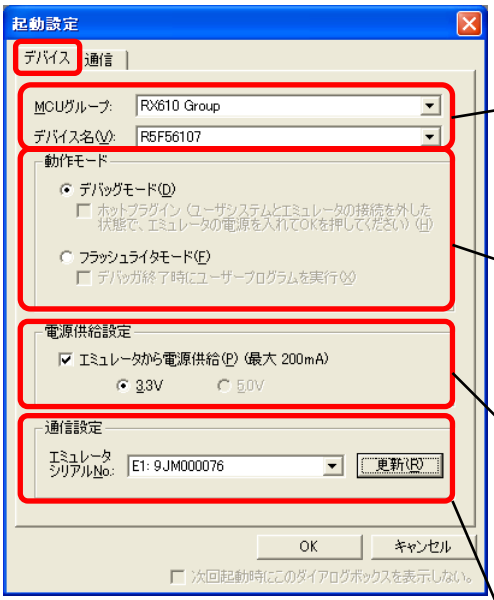


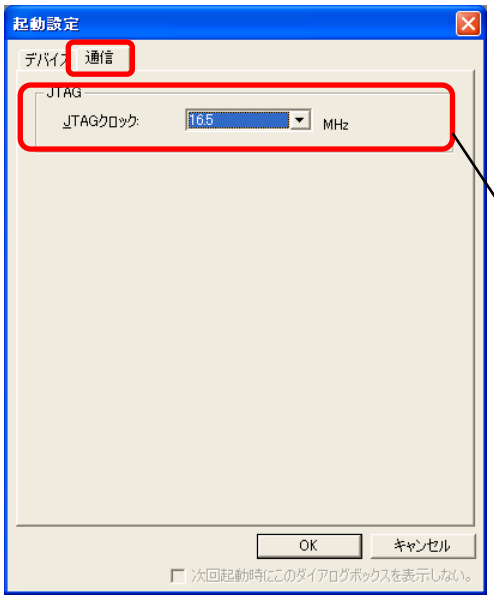
図 4.4 ワークスペースのバージョンが古い場合のダイアログボックス

4.4 E1 エミュレータ起動の設定

(1) 図 4.5の[起動設定]ダイアログが表示されますので、図のように設定を行い、[OK]ボタンを押してください。

[起動設定]ダイアログが表示されない場合は、図 4.6の操作をしてください。





実際にユーザシステムに登載されているマイコンを選択します。本書で使用しているユーザシステムの場合は以下を選択してください。
MCU グループ　：　RX610 Group
デバイス名　　：　R5F56107

E1 を使用してデバッグを行うときは「デバッグモード」を、E1 を簡易フラッシュライタとして使用する場合は、「フラッシュライタモードを選択します。ここでは、「デバッグモード」を選択してください。

ユーザシステムにエミュレータから電源を供給する場合はチェックします。
RX610 グループでは、3.3V のみ選択が可能です。ここでは、図のように選択し、3.3V をユーザシステムに供給してください。

使用しているエミュレータのシリアル番号が表示されます。
シリアル番号が表示されていない場合は、更新ボタンを押してください。それでも表示されない場合は、USB ケーブルの接続および USB ドライバの設定を確認してください。

マイコンと E1 間の JTAG クロックを指定します。通常はデフォルトの 16.5MHz を選択してください。

図 4.5　起動設定

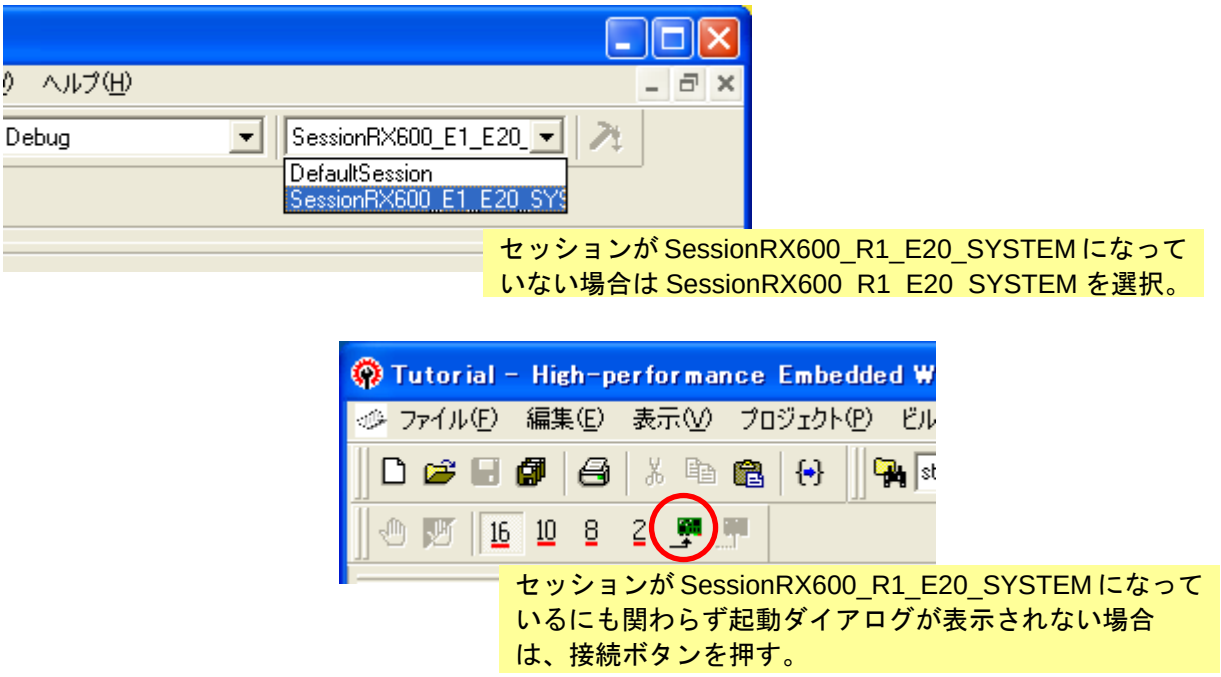
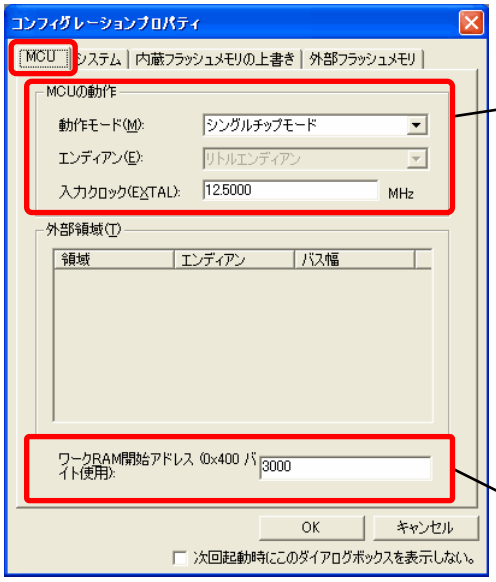


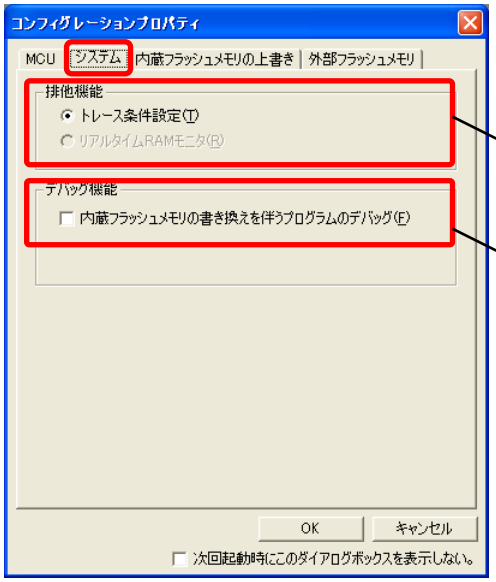
図 4.6　セッションの切り替えと接続ボタン

接続処理中は、図 4.7のように[接続中]画面が表示されます。

図 4.7　エミュレータ接続中画面

(2) 図 4.8のように[コンフィグレーションプロパティ]が表示されます。
[MCU]、[システム]タブ設定が図 4.8のようになっていることを確認し、[OK]ボタンを押してください。





動作モード：
デバッグ対象のプログラムで使用するマイコンの動作モードを選択してください。
本書では、シングルチップモードを選択します。
エンディアン：
ユーザシステムにて設定されているエンディアンが表示されます。
入力クロック：
マイコンの XTAL 端子に接続されているクロック周波数を指定します。
今回使用する RX610 用 CPU ボードでは 12.5MHz が接続されています。

エミュレータのためのワーク RAM 領域を指定します。
通常デフォルトでかまいませんが、指定された領域が、DMA または DTC 機能の転送元、および転送先となる場合は、別のアドレスに変更してください。
ここではデフォルトのままとします。

E1 ではリアルタイム RAM モニタに対応していないため、排他機能の選択はできません。

内蔵フラッシュメモリの消去、書き換えを行うプログラムをデバッグする場合にチェックします。
チュートリアルプログラムでは、内蔵フラッシュの書き換えは行わないため、ここではチェックしません。

この他の設定については、デフォルトのままとします。詳細については、「E1/E20 エミュレータユーザーズマニュアル」を参照してください。

図 4.8 コンフィグレーションプロパティダイアログ

(3) [接続中]画面が表示されます。

(4) 正常に接続が完了すると、図 4.9のようにアウトプット画面に「Connected」が表示され、画面上部のデバッグ関連のボタンがアクティブになります。

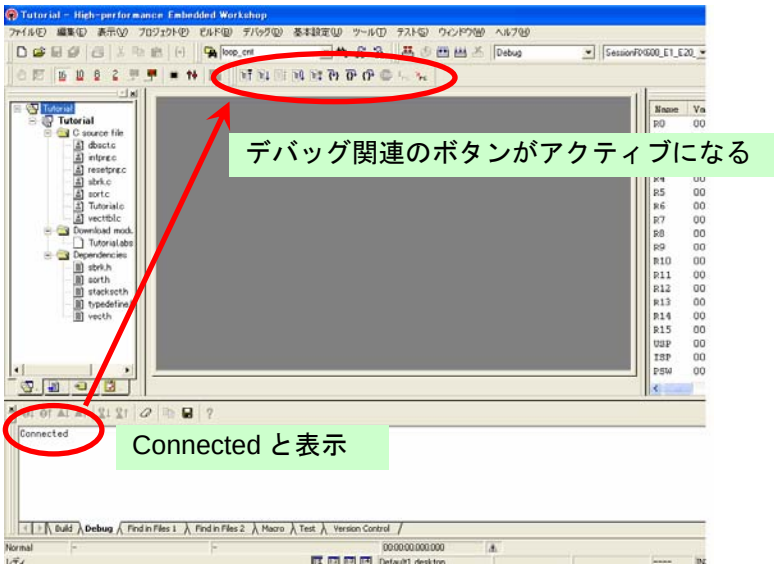


図 4.9　エミュレータ接続の確認

4.5　ソースファイルのオープン

ソースファイルをオープンするには、ワークスペースウィンドウのソースファイル名をダブルクリックします。たとえば、Tutorial.cをオープンするには、図 4.10のようにします。

オープンしたウィンドウは、エディタウィンドウと呼ばれ、ソースの編集も行うことができます。

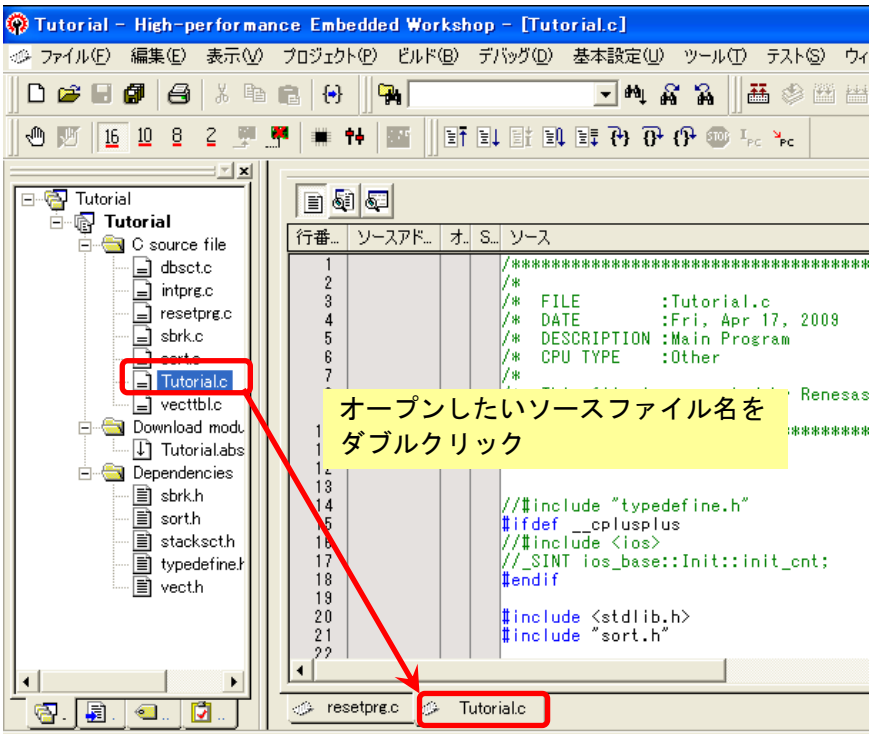


図 4.10　ソースファイルのオープン

4.6　ビルド（コンパイル＆リンク）

プログラムのビルドを行う場合には、ツールバーのビルドボタンを押します。

ビルドが完了すると、アウトプットウィンドウにビルド結果を表示します。チュートリアルプログラムの場合、いくつかのワーニングが出力されますが、無視していただいてかまいません。

注意：1 回目のビルドでは、標準ライブラリのコンパイルも行うため、ビルドに時間がかかります。2 回目以降は、オプションを変更しない限り、標準ライブラリのコンパイルは不要なため、時間は短縮されます。

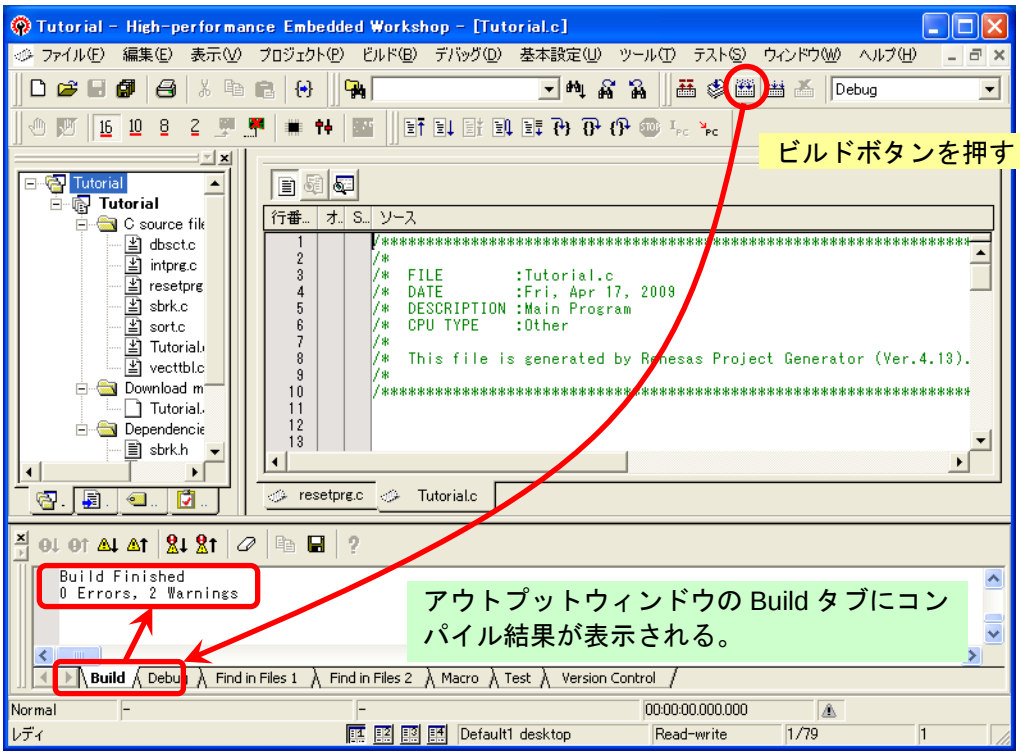


図 4.11　ビルドの実行とビルド結果

チュートリアルプログラムのワークスペースでは、ビルド後にダウンロードを自動的に行う設定になっています。図 4.12のダイアログが表示された場合、「はい」ボタンを押すことにより、プログラムのダウンロードが行われます。

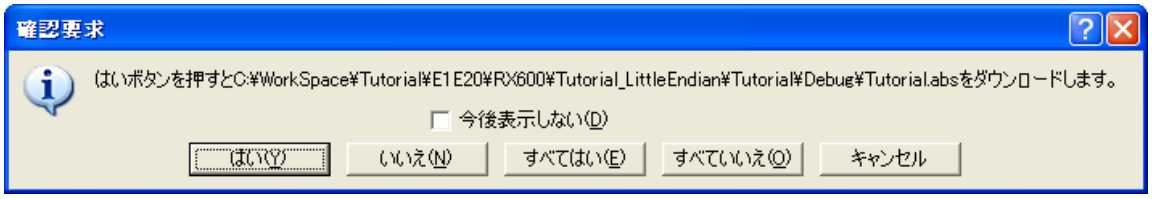


図 4.12　ダウンロード確認要求

4.7 プログラムのダウンロード

ビルド後の自動ダウンロード機能以外に、以下の方法でプログラムをダウンロードすることが可能です。

(1) 図 4.13のようにワークスペース内の[Download module]に展開表示されているロードモジュールファイル （本章では“Tutorial.abs”）をダブルクリックします。

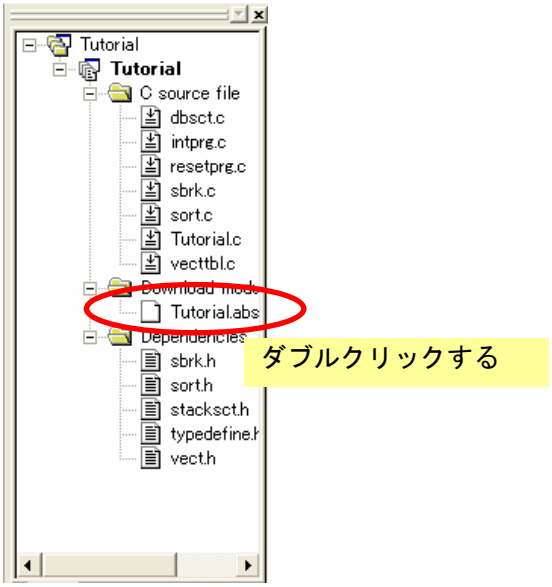


図 4.13 プログラムのダウンロード

(2) プログラムのロードが完了すると、図 4.14のように下矢印がつきます。

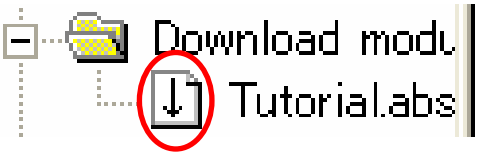


図 4.14 プログラムのダウンロード完了

4.8 CPU のリセット

CPU をリセットする場合は、[デバッグ]メニューから[CPU のリセット]を選択するか、ツールバー上の[CPU リセット]ボタンを選択してください。

CPU リセットによりレジスタは初期値に設定され､プログラムカウンタにリセットベクタが設定されます。

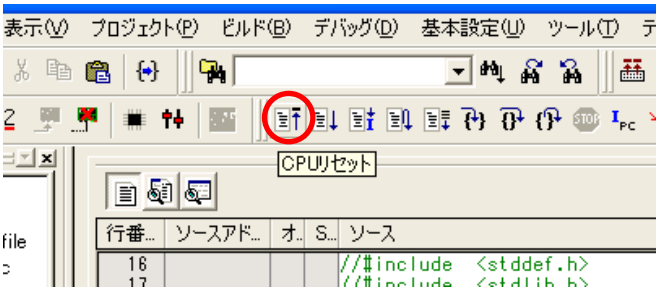


図 4.15 CPU リセットボタン

4.9　プログラムの実行および停止

図 4.16に示す［実行］ボタンを押すと、プログラムの実行を開始します。

また、CPU のリセットをした後にプログラムの実行を行う[リセット後実行]ボタンがあります。

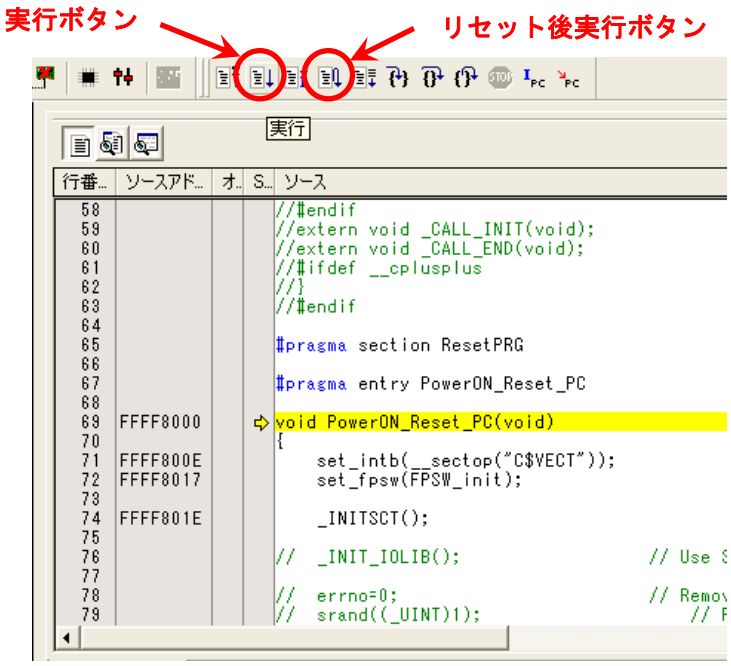


図 4.16　実行ボタン

プログラム実行中には図 4.17に示す[停止]ボタンが有効になります。

このボタンを押すことによってプログラムを停止することができます。

図 4.17　停止ボタン

停止後のエディタウィンドウを図 4.18に示します。

停止した時のプログラムカウンタに該当する行は、黄色帯で表示されます。

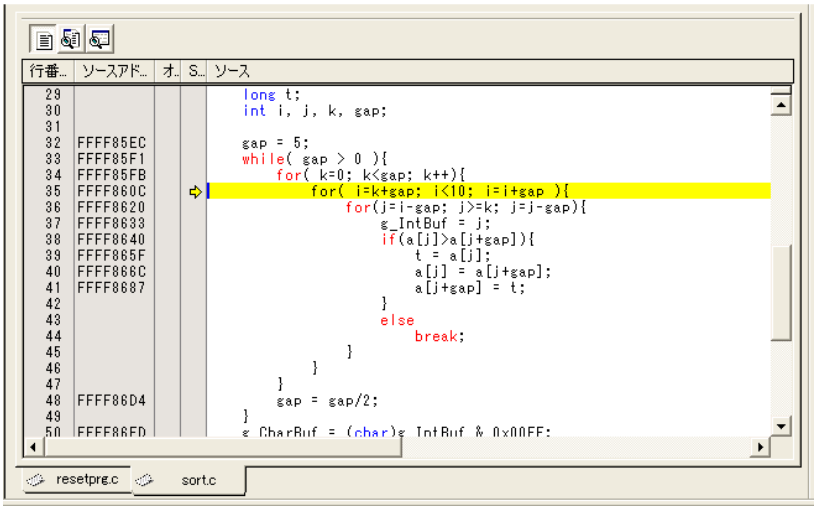


図 4.18　プログラム停止後のエディタウィンドウ

4.10　ブレーク機能

E1 エミュレータはブレーク機能として、ソフトウェアブレーク機能およびオンチップブレーク機能を搭載しています。

ソフトウェアブレーク機能およびオンチップブレーク機能の特徴を表 4.2に示します。こちらも参考にしながら用途に合わせたブレーク設定をしてください。

表 4.2　ブレーク機能仕様

ブレーク機能	設定ポイント	ブレーク条件	フラッシュの書き換え	プログラム実行中
ソフトウェアブレーク				
	256 点	指定アドレス	あり	設定可能 （但し、内蔵 RAM 領域のみ）
オンチップブレーク				
実行前 PC ブレーク	8 点	指定アドレス	なし	設定可能
イベントブレーク（実行アドレス）		指定アドレス	なし	設定可能
イベントブレーク（データアクセス）	4 点	データアクセス	なし	設定可能
トレースフルブレーク	-	トレースバッファが満杯	なし	設定不可

ソフトウェアブレークは多数の設定が可能ですが、命令を専用のブレーク命令に書き換えることにより実現しているために、ブレークポイント設定後のプログラム実行時に書き換えのための時間が必要となります。このため、通常はオンチップブレークを使用していただき、ソフトウェアブレークは頻繁に設定／削除の必要のない箇所へ使用することをお勧めします。

ここでは、フラッシュ書き換えの必要ないオンチップブレーク機能の設定方法について説明します。

(1) ブレークポイントの設定を行います。
図 4.19に示しますオンチップブレークカラムでダブルクリックしますと、青色の丸が表示されます。
ブレークポイントは、ソースアドレスカラムにアドレスが表示されている行にのみ設定可能です。

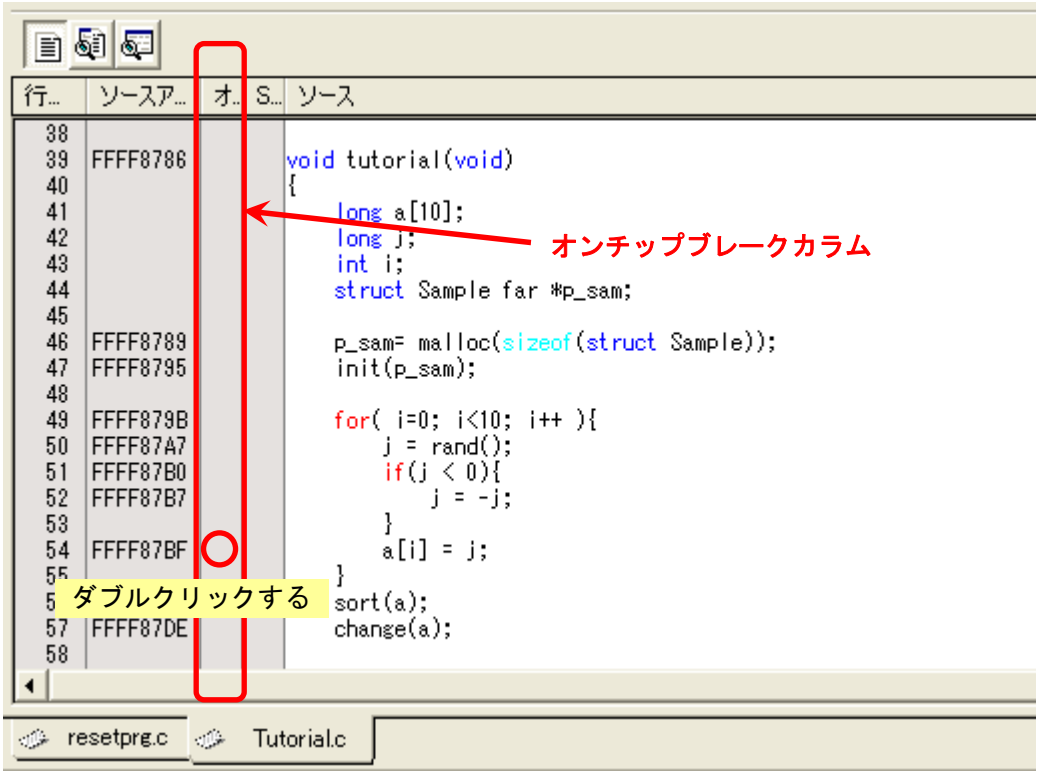


図 4.19　ブレークポイントの設定

(2) 図 4.20のようにリセット後実行ボタンを押し、プログラムを実行します。
プログラムを実行すると、先ほど設定したブレークポイントでプログラムの実行が停止します。
停止した時のプログラムカウンタに該当する行は図 4.20のように黄色帯で表示され、アウトプットウィンドウにブレークにより停止したことを示す「:Before break」のメッセージが表示されます。

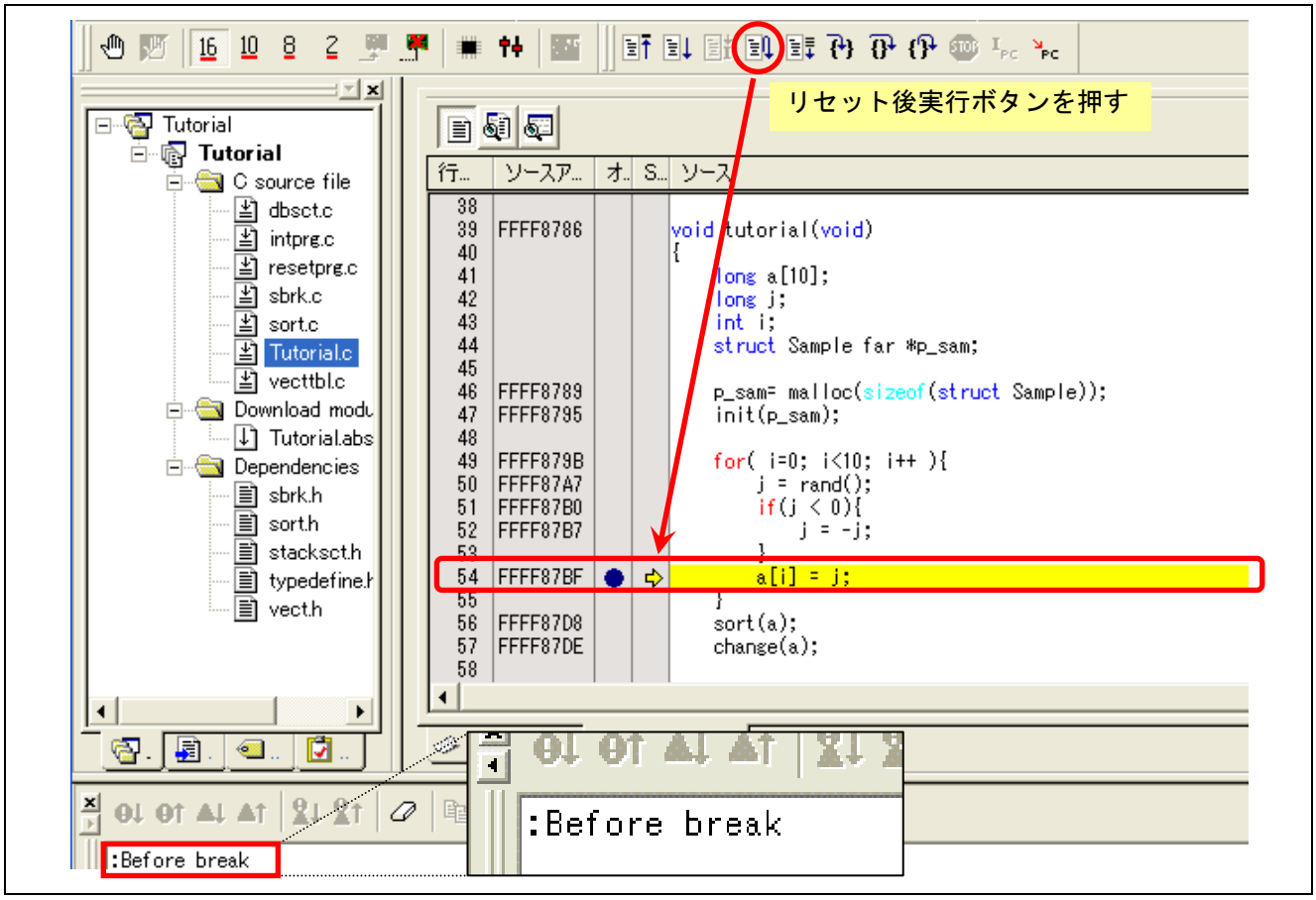


図 4.20　ブレークポイントで停止時の表示

(3) ブレークポイントの設定を解除するには、図 4.21のように再度青色の丸をダブルクリックします。

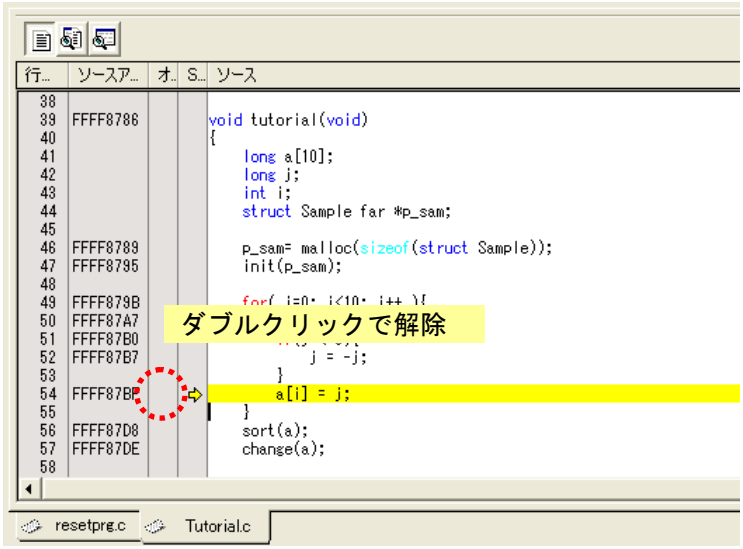


図 4.21　ブレークポイント設定の解除

4.11　C 言語の変数の参照/設定

C 言語の変数を参照/設定する方法として以下の機能があります。

1　ウォッチウィンドウによる変数値の参照／設定
プログラム内で使用されている任意の変数値の参照／設定を行うことができます。
プログラム停止時だけでなく、プログラム実行中に値の参照／設定を行うことが可能です。

2　ローカルウィンドウによる変数値の参照／設定
現在の PC 値が指し示す関数内で使用されているローカル変数を自動的に表示し、参照／設定を行うことができます。

3　ツールチップウォッチ機能による変数値の参照
表示されているソースファイル上の変数にカーソルを合わせることで、簡単に変数の値を確認することができる機能です。

4.11.1　ウォッチウィンドウによる変数値の参照／設定

E1 エミュレータではウォッチウィンドウを使用することで、プログラム内で使用する任意の変数値を確認することができます。

本項では、ウォッチ機能を使用して、サンプルプログラム“sort.c”ファイル内のグローバル変数“int 型 g_IntBuf”値の変化を確認する方法を説明します。

注意：ローカル変数を登録した場合は、現在の PC がそのローカル変数を宣言した関数内にあるときしか、値を参照することはできません。

(1) 変数内容の参照には、ウォッチウィンドウを使用します。
図 4.22のように表示->シンボル->ウォッチと選択し、ウォッチウィンドウを開いてください。

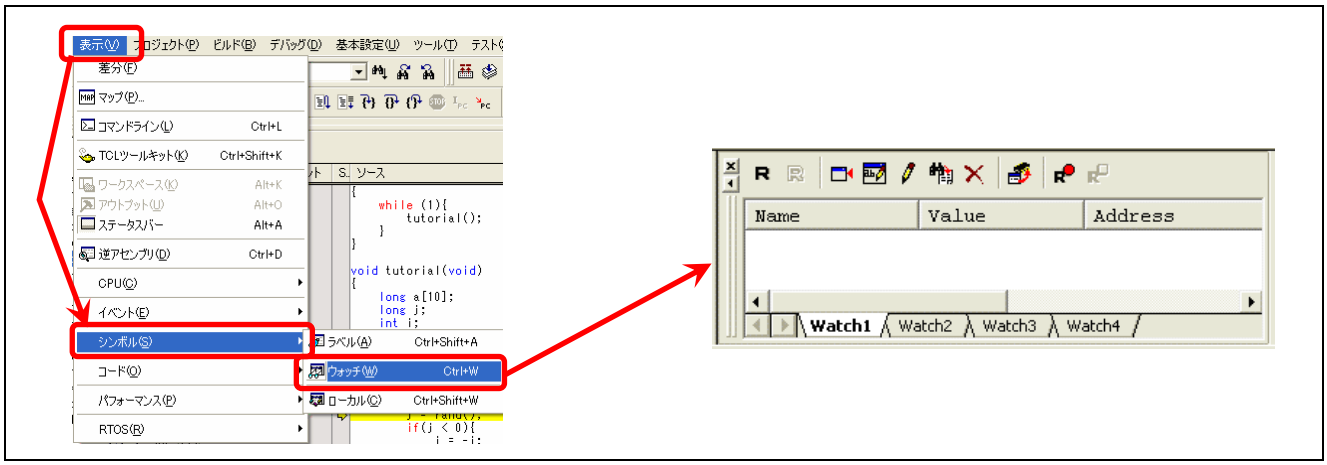


図 4.22　ウォッチウィンドウ

(2) g_IntBufを選択しウォッチウィンドウにドラッグ＆ドロップします。図 4.23のようにウォッチウィンドウにg_IntBufが追加されますので、確認してください。

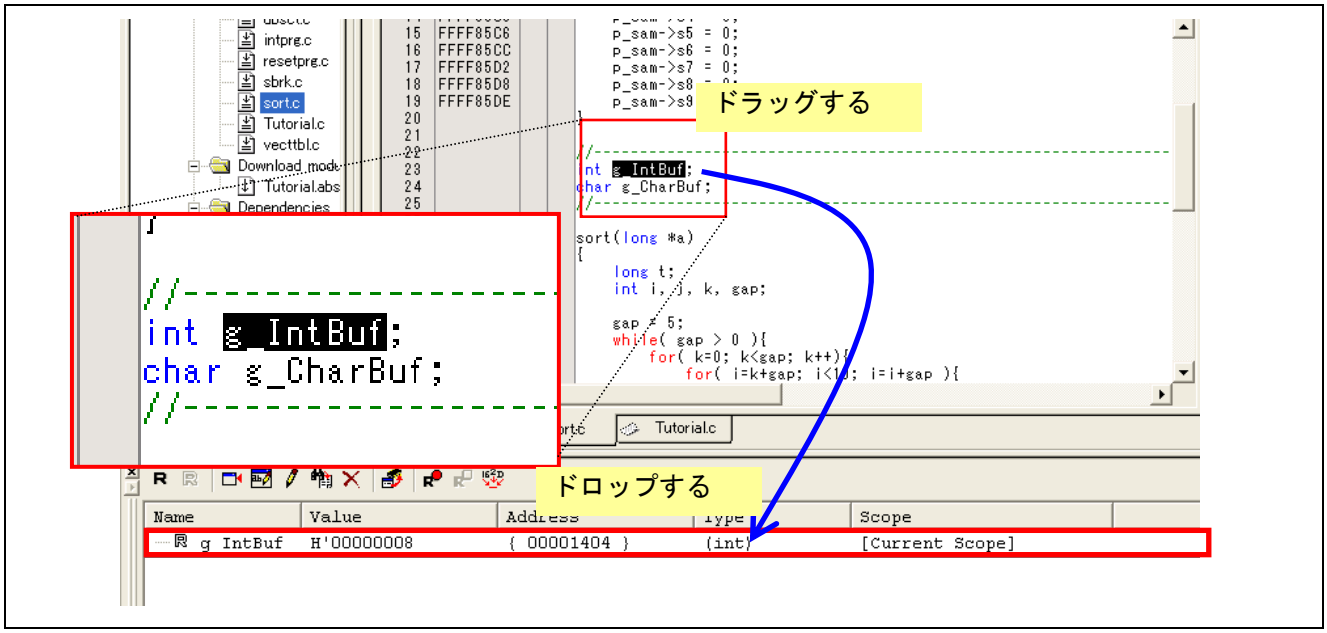


図 4.23　ウォッチ設定

(3) 図 4.24を参考に「g_IntBuf = j;」が記述されているsort.c 37 行目にオンチップブレークポイントを設定します。ブレークポイントの設定方法に関しては、「4.10　ブレーク機能」を参照して下さい。

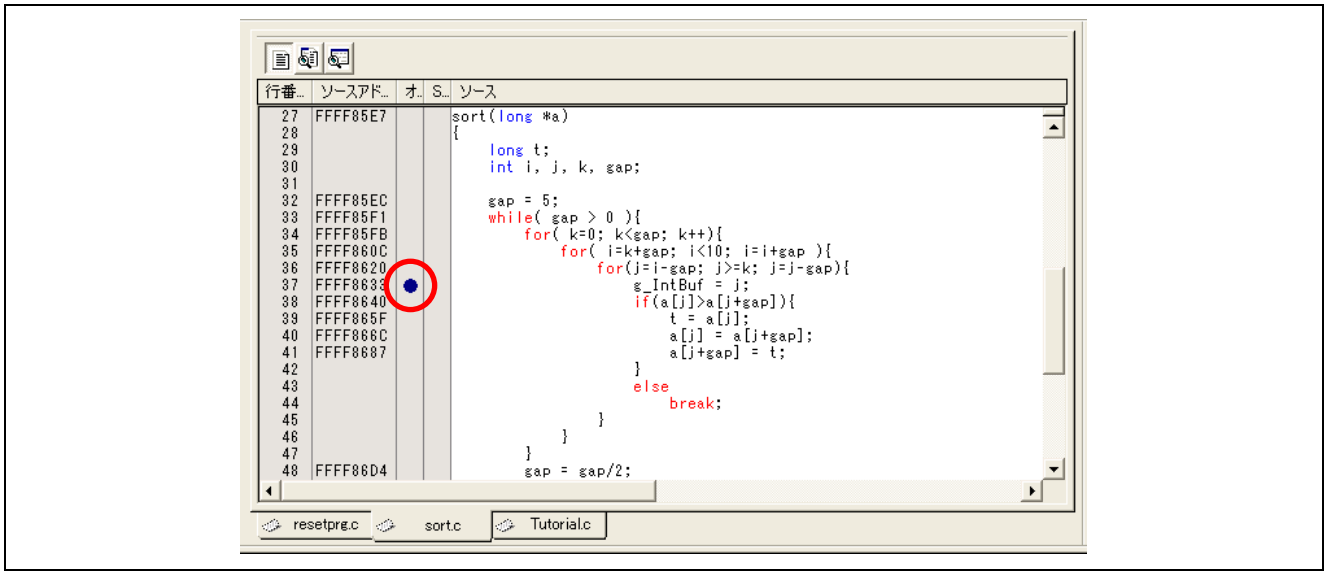


図 4.24　ブレークポイントの設定

(4) 上記まで設定が完了しましたら、[リセット後実行]ボタンを押して、プログラムを実行します。

(5) 1 度目のブレークでは指定行実行前に止まるので、g_IntBuf には初期値 0 が入っています。

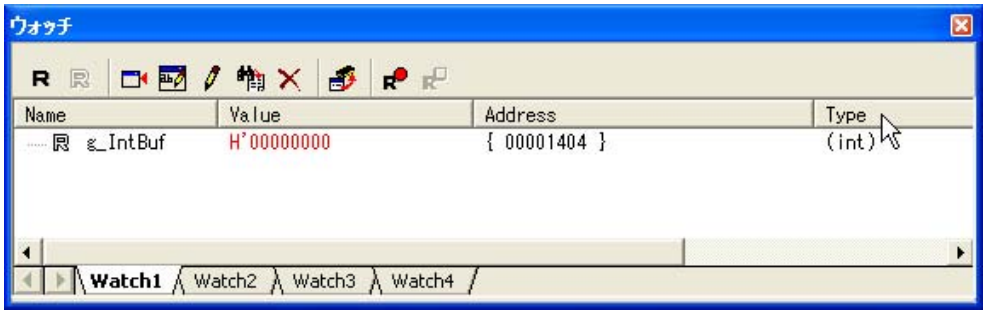


図 4.25　初回ブレーク時の g_IntBuf の値

(6) 次に[リセット後実行]ボタンではなく、[実行]ボタンよりプログラムを再度実行すると、同じポイントでブレークします。ブレーク後、ウォッチウィンドウから g_IntBuf 値の更新を確認できます。この動作を繰り返しながら、g_IntBuf が 0（初期値）→0→1→2→3→・・・と変化することが確認できます。

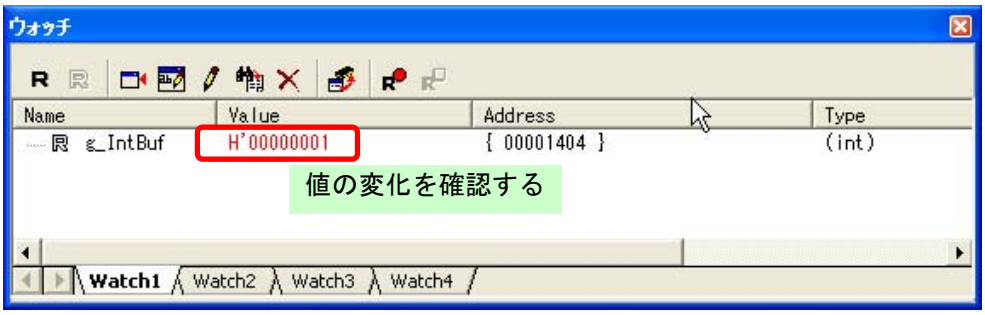


図 4.26　数回目にブレークした時の g_IntBuf の値

(7) ウォッチウィンドウには、4 つのタブがありますので、4 種類の変数の組み合わせを切り替えて使用することも可能です。

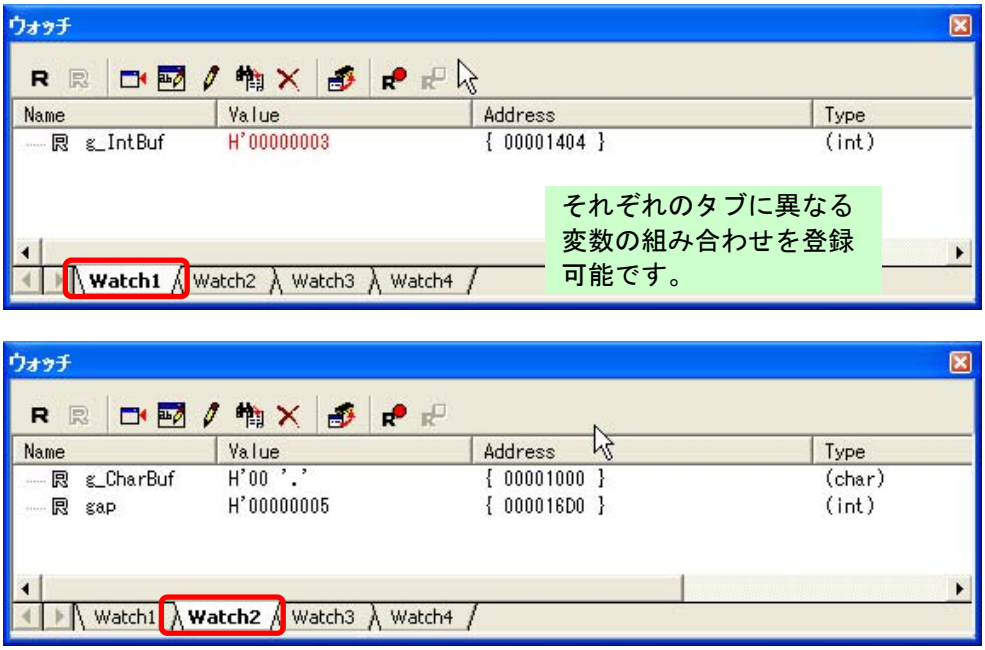


図 4.27　ウォッチウィンドウのタブ切り替え

(8) 変数の値を変更する方法は図 4.28のように 2 種類あります。ひとつは、変数の値をシングルクリックしてのインプレース編集です。もうひとつは、変数をダブルクリックすることにより、表示されるダイアログで編集する方法です。

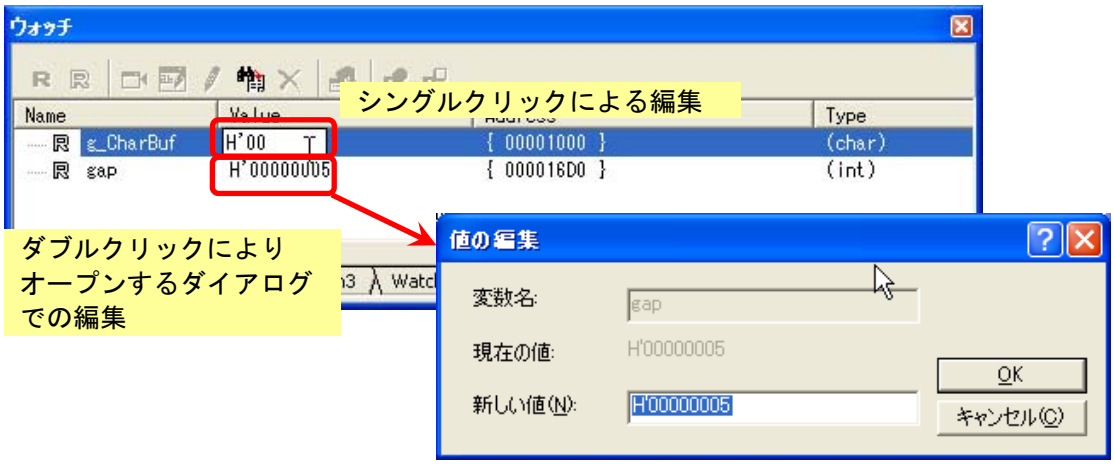


図 4.28　値の変更

(9) ウォッチウィンドウの自動更新機能を有効にすることでプログラム実行中の値の変化を確認することも可能です。

自動更新の確認を行うため、設定しているブレークポイントを解除して下さい。

加えて、自動更新を有効にするため、ウォッチウィンドウで、変数を選択してから、図 4.29のように[自動更新有効化]ボタンを押してください。自動更新が有効になった変数は、Rマークが白抜きから塗りつぶされたRとなります。

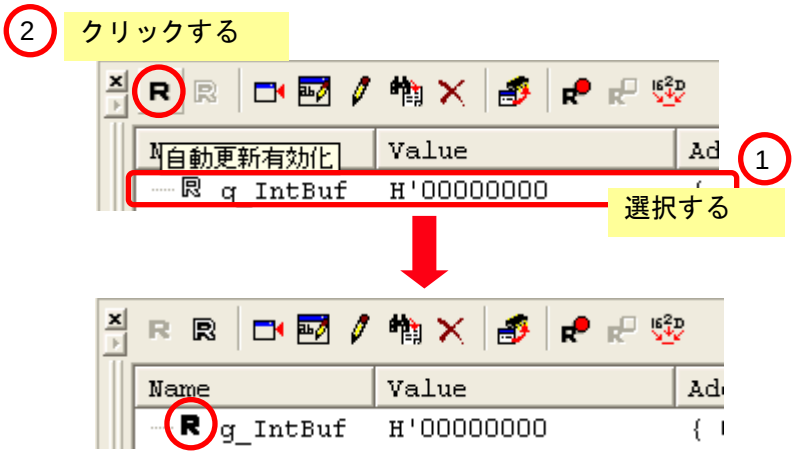


図 4.29　ウォッチ対象の自動更新化

(10) 自動更新有効化しましたら、プログラムを実行して下さい。
g_IntBuf の変化を確認できます。

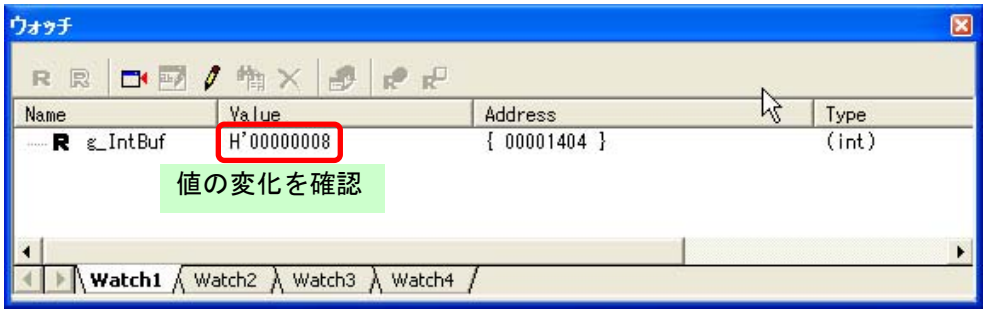


図 4.30　自動更新中のウォッチウィンドウの表示

(11) 値の変化を記録することも可能です。
自動更新の記録開始はブレーク中にのみ行うことができますので、いったんプログラムをブレークしてください。ブレーク後に、図 4.31で示されたアイコンをクリックすると自動更新の記録を開始します。

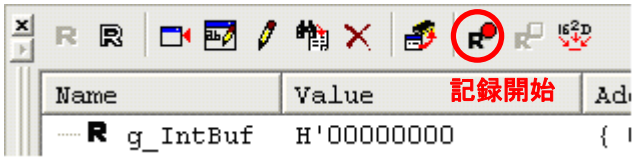


図 4.31　自動更新記録の開始

(12) 記録開始時に図 4.32の設定画面が表示されます。
ウォッチウィンドウの記録を保存したいシートを選択し、保存するファイル名を入力してください。

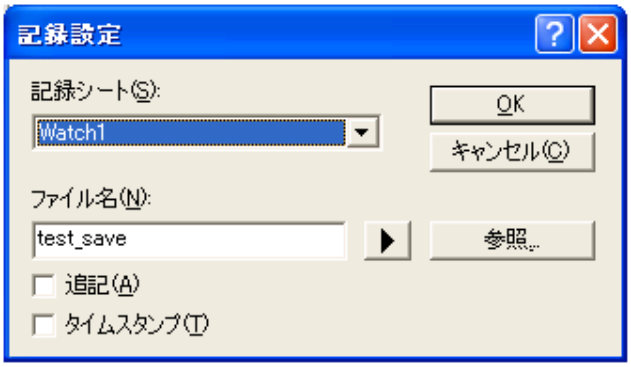


図 4.32　記録設定画面

(13) 記録を開始しましたら、プログラムを実行してください。
その後、自動更新で値が更新されるのを確認しましたら、適当なタイミングで、記録の終了ボタンをクリックしてください。記録設定で指定したフォルダに指定したファイル名のテキストファイルが生成されます。生成されたファイルの例を図 4.33に示します。

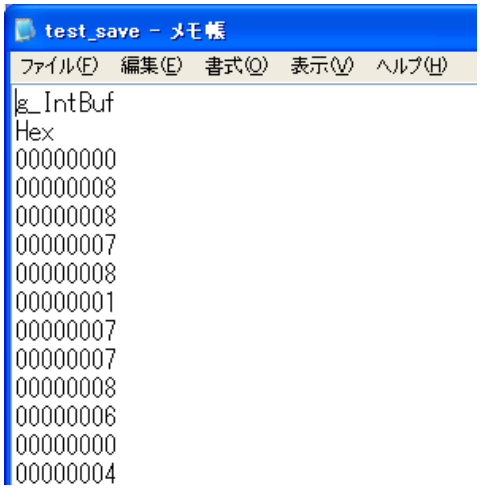


図 4.33　自動更新記録の結果

(14) 自動更新の記録を終了するために、記録終了ボタンを押します。

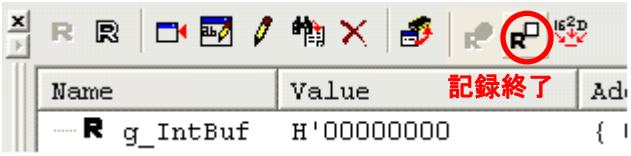


図 4.34　自動更新記録の終了

4.11.2　ローカルウィンドウによる変数値の参照／設定

ローカル変数の値を参照する場合は、ローカルウィンドウを使用します。

ウォッチウィンドウとは異なり、プログラム停止時の PC 値が指し示す関数内で使用されているローカル変数を自動的に表示し、参照／設定を行うことができます。

ローカル変数はスタックに格納されているため、プログラム実行中に値を参照することはできません。

(1) 図 4.35のように表示->シンボル->ローカルと選択し、ローカルウィンドウを開いてください。

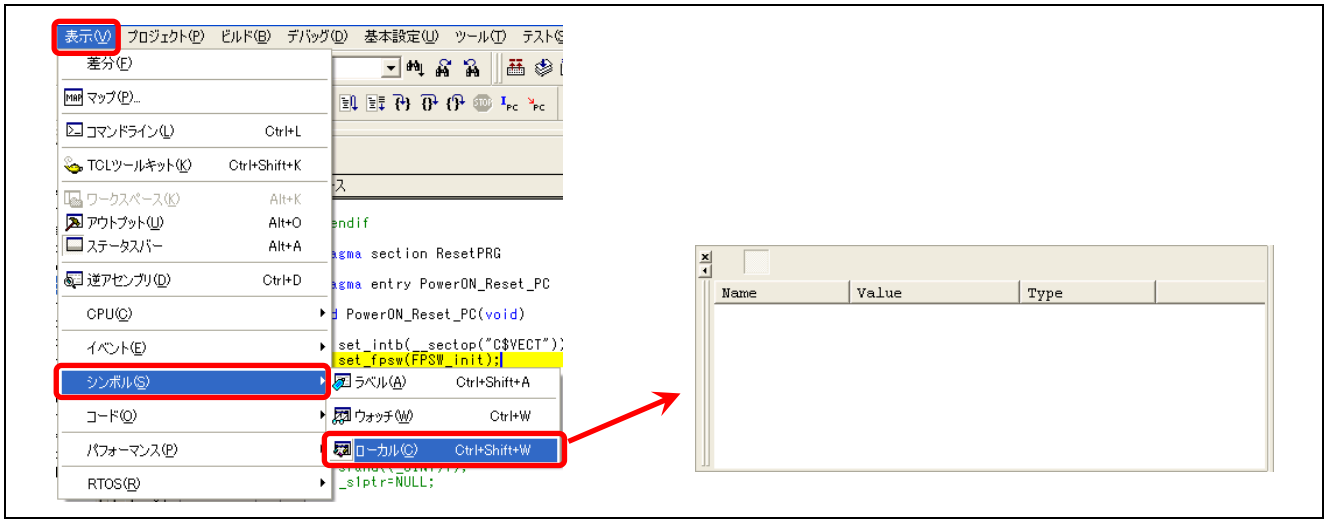


図 4.35　ローカルウィンドウの表示

(2) プログラムを実行し、適当な箇所でプログラムを停止してください。
図 4.36のように停止時のPC値が指し示す関数内で使用されているローカル変数が自動的に表示されます。

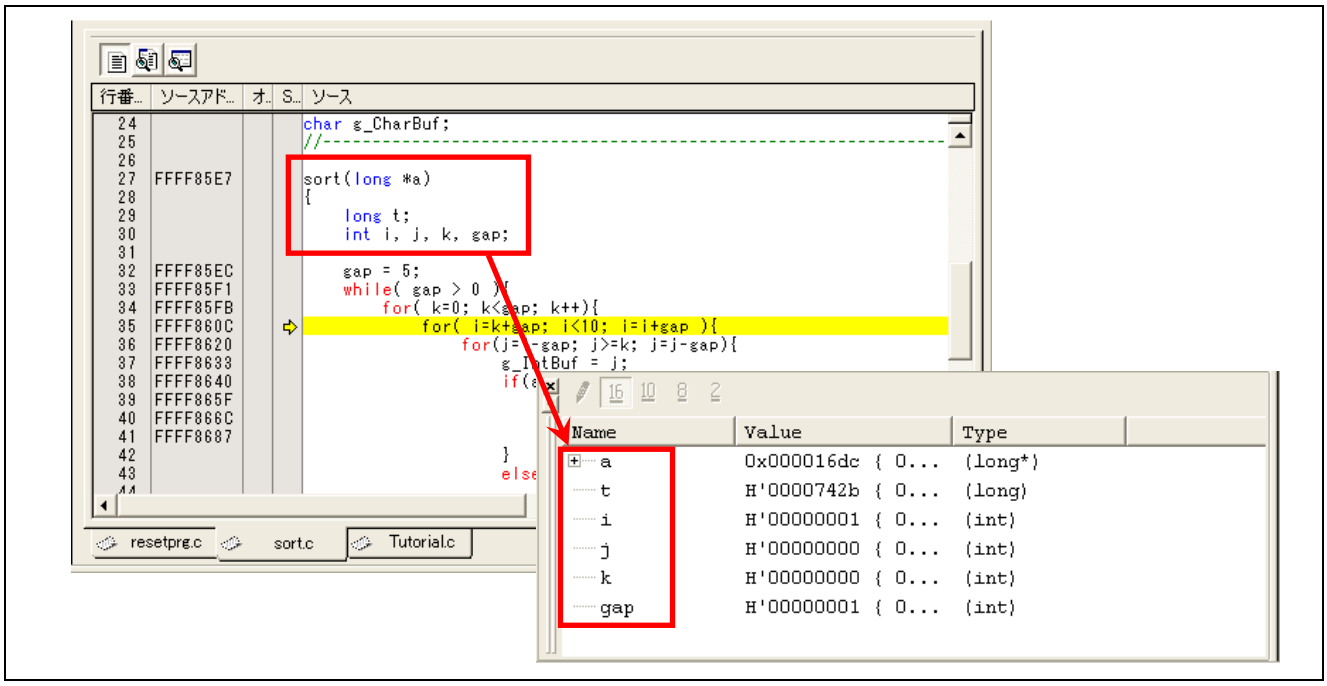


図 4.36　ローカルウィンドウの参照

4.11.3　ツールチップウォッチ機能による変数値の参照

プログラムの実行中および停止中に、ツールチップ表示（マウスポイントによるウォッチ）が可能です。

値を参照したい変数にマウスカーソルを合わせると図 5.26 のように変数の値を参照することができます。

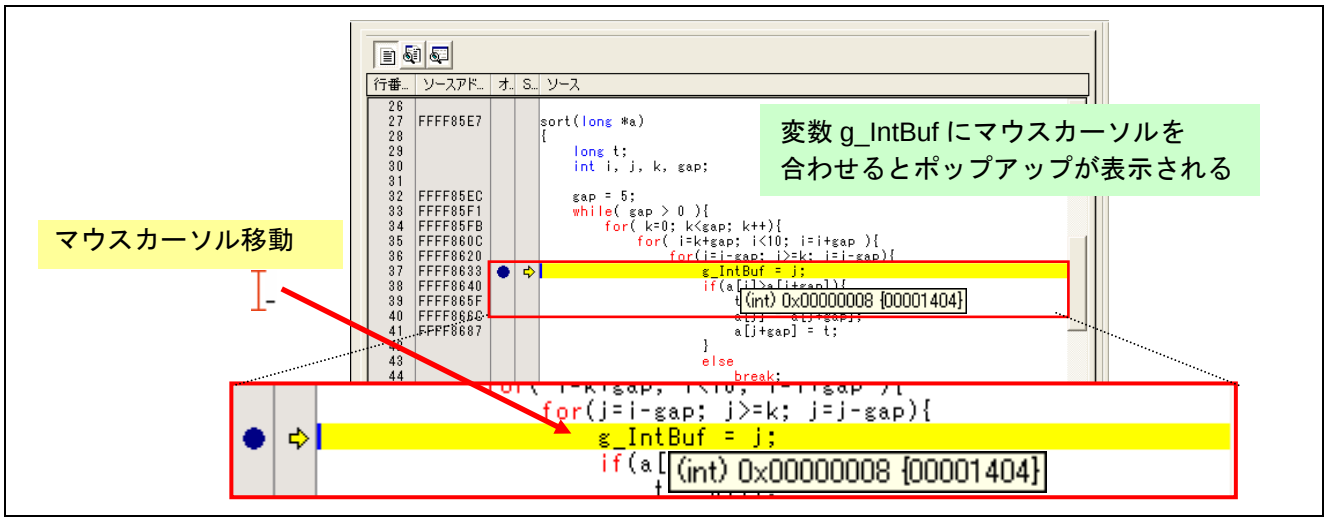


図 4.37　ツールチップウォッチ

4.12　メモリウィンドウによるデータの参照／設定

ウォッチ機能では、すでに宣言された変数を参照することができますが、任意のアドレス空間を参照／設定したい場合は、メモリウィンドウを使用します。

(1) 図 4.38のように表示->CPU->メモリと選択し、メモリウィンドウを開いてください。

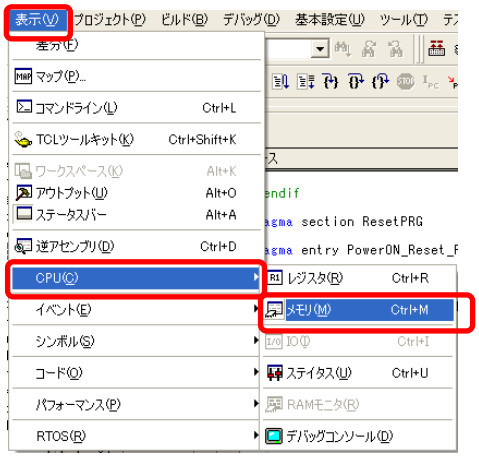


図 4.38　メモリウィンドウの表示

(2) 図 4.39の表示開始アドレスのダイアログが表示されますので 1000H番地を表示開始アドレスに指定してください。指定後、メモリウィンドウが図 4.40のように開きます。

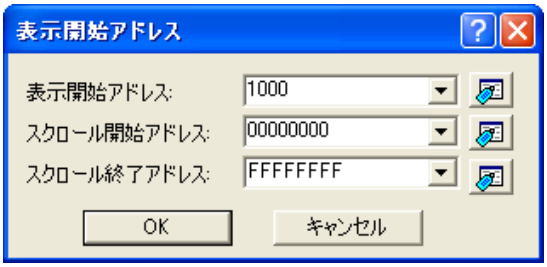


図 4.39　表示開始アドレス入力ダイアログ

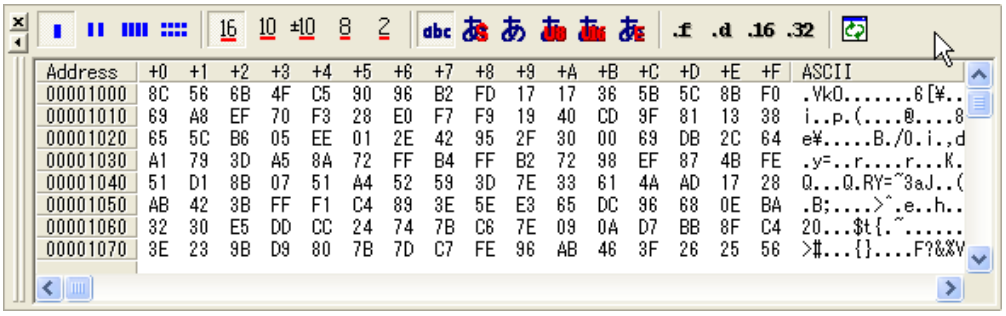


図 4.40　メモリウィンドウの表示

(3) 表示サイズを切り替えることも可能です。アクセスサイズの変更方法に関して、図 4.41に示します。

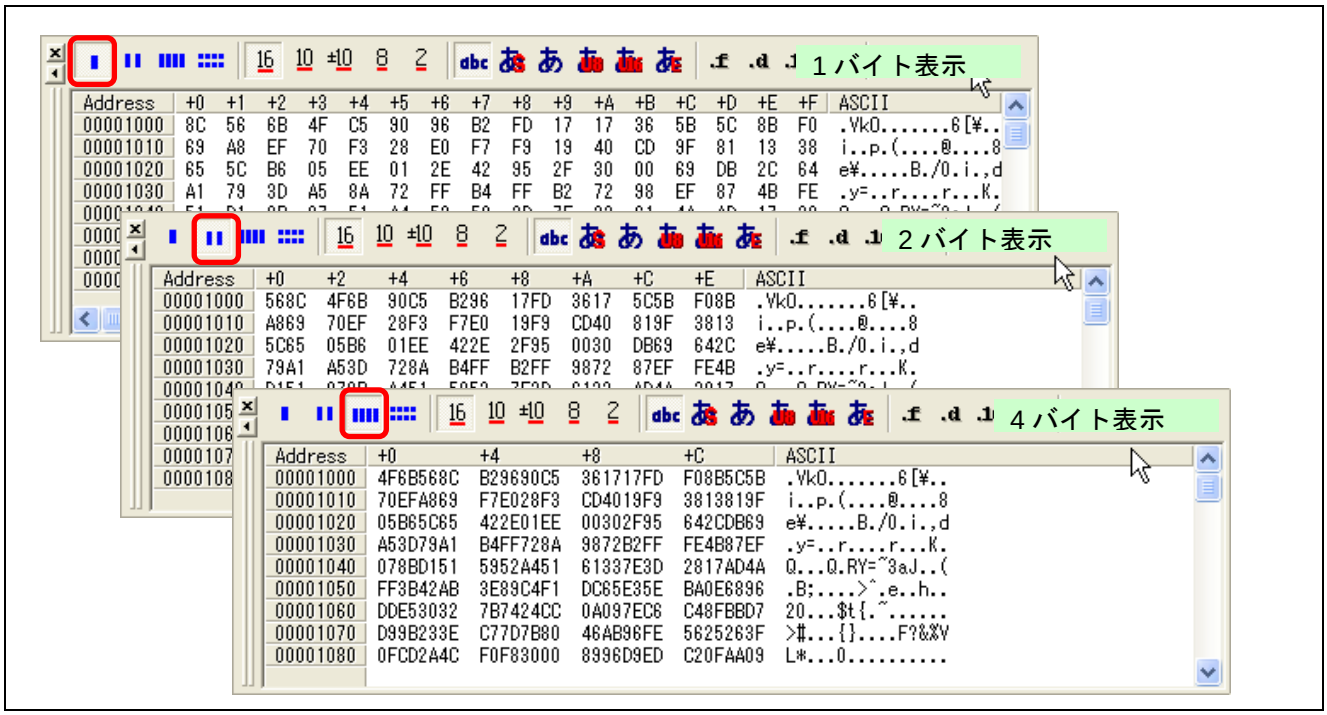


図 4.41　メモリウィンドウでの表示サイズの変更

注意：メモリウィンドウでは、原則として表示サイズで指定したサイズでメモリにアクセスします。

注意：I/O レジスタ領域に対してメモリウィンドウを使用する場合は、必ず参照するレジスタのアクセスサイズにあわせた表示サイズを選択する必要があります。I/O ウィンドウでは自動的にアクセスサイズに合わせたアクセスをしますので、I/O レジスタ領域を参照／設定する場合は、I/O ウィンドウを使用してください。

(4) メモリの値を変更するには、2 種類の方法があります。

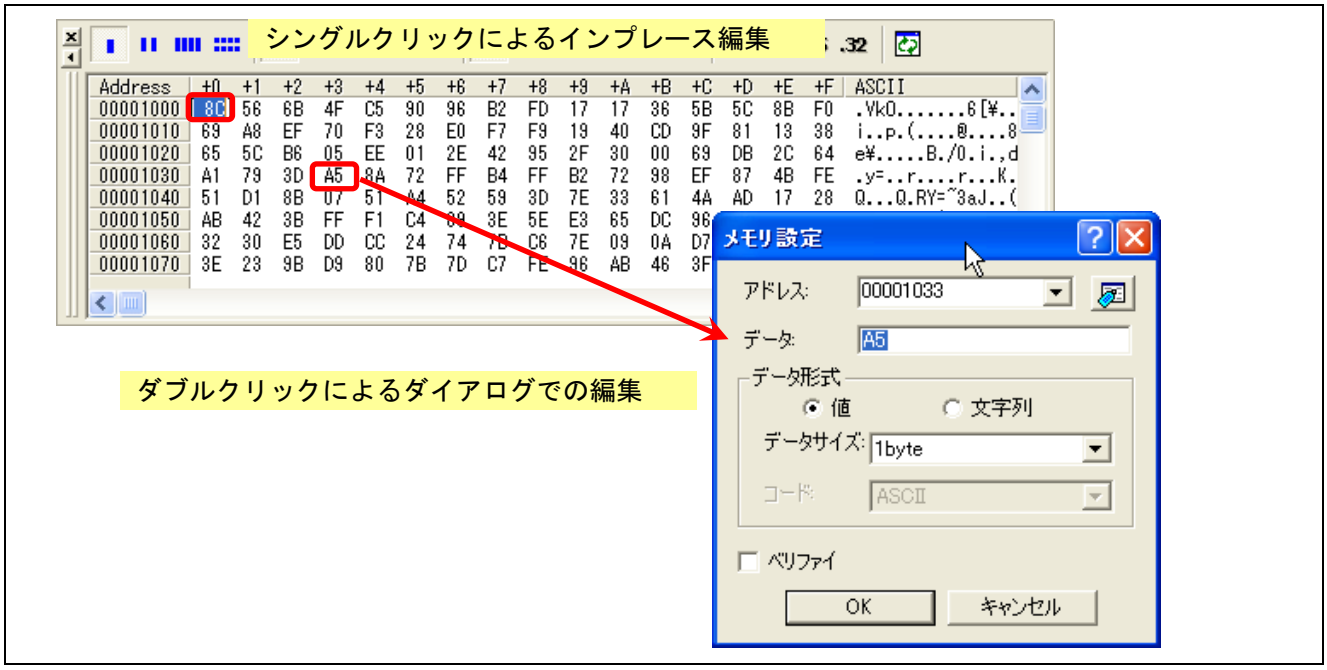


図 4.42　メモリの値の変更方法

(5) メモリウィンドウに表示されている値は、ブレーク時の値です。更新ボタンを押すことで最新の値を読み出すことができます。

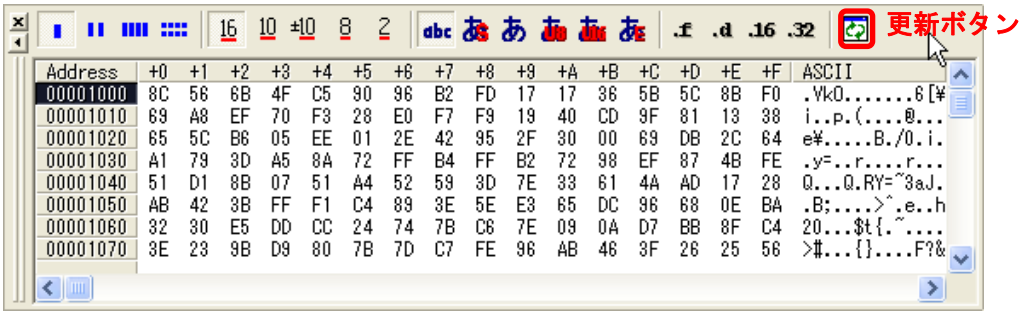


図 4.43　表示内容の更新

(6) ウォッチウィンドウと同様に、プログラム実行中に値の変化を確認することも可能です。

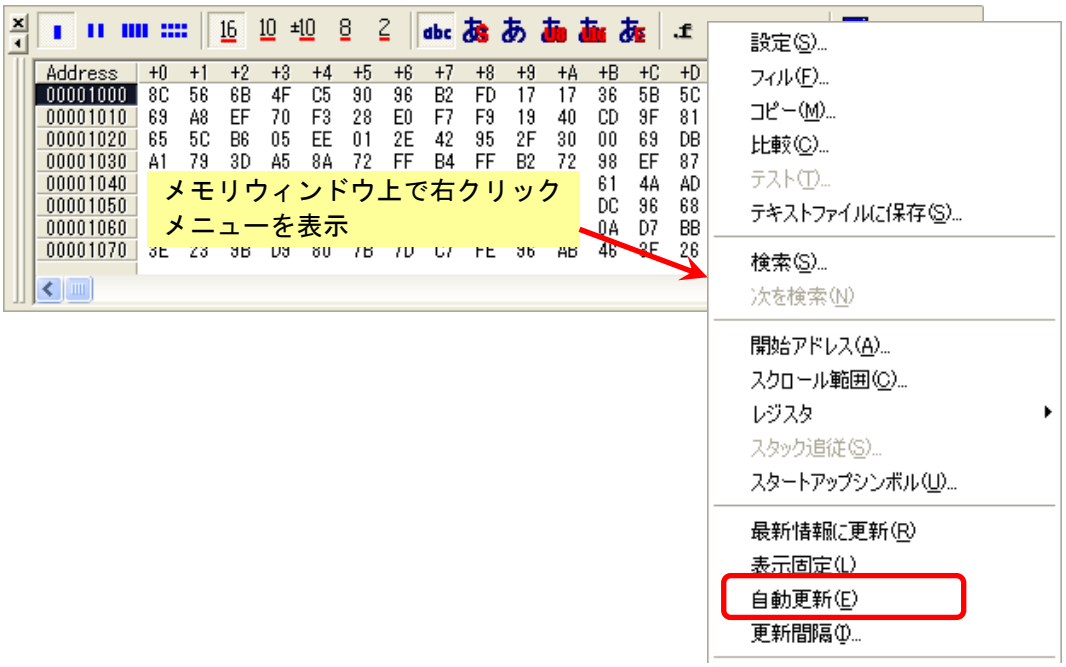


図 4.44　自動更新機能の有効化

4.13　レジスタウィンドウによるレジスタ値の参照／設定

レジスタウィンドウでは、汎用レジスタおよび制御レジスタの値を参照することができます。

例として、プログラムカウンタ(PC)値を参照してみましょう。

(1) 図 4.45のように表示->CPU->レジスタと選択し、レジスタウィンドウを開いてください。

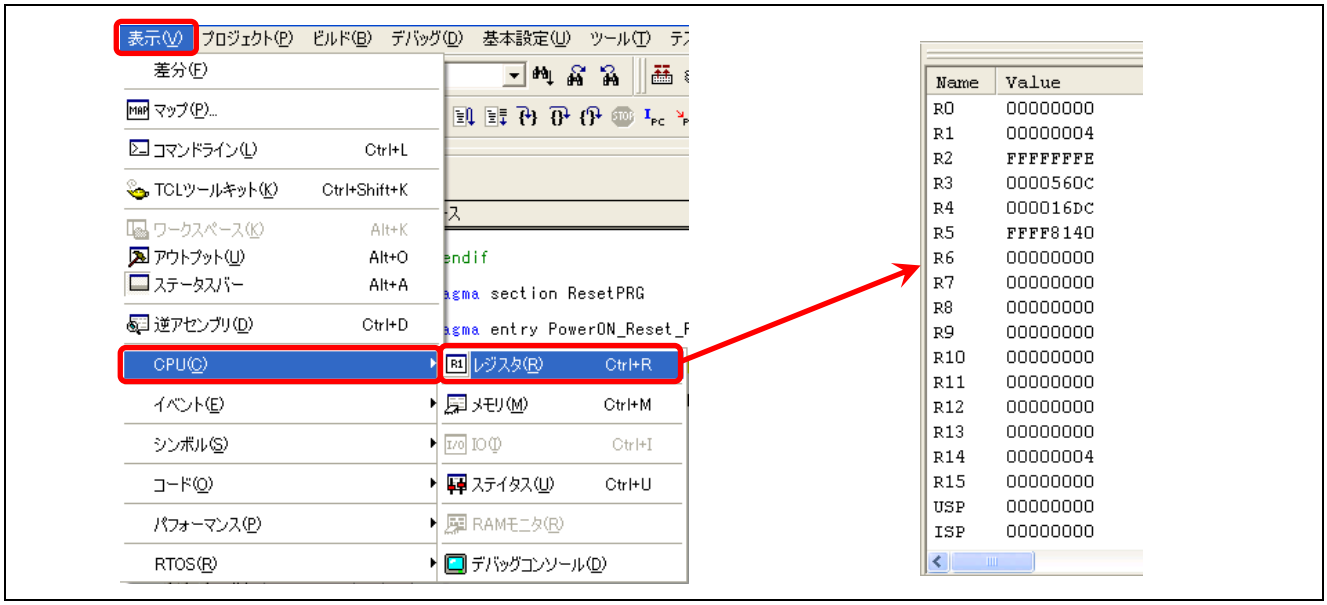


図 4.45　レジスタウィンドウの表示

(2) レジスタウィンドウの表示を確認したら、[リセット]ボタンを押してください。
図 4.46で示すように、プログラムカウンタがCPUリセット後、リセットベクタに書かれた値(FFFF 8000H)となっていることが確認できます。

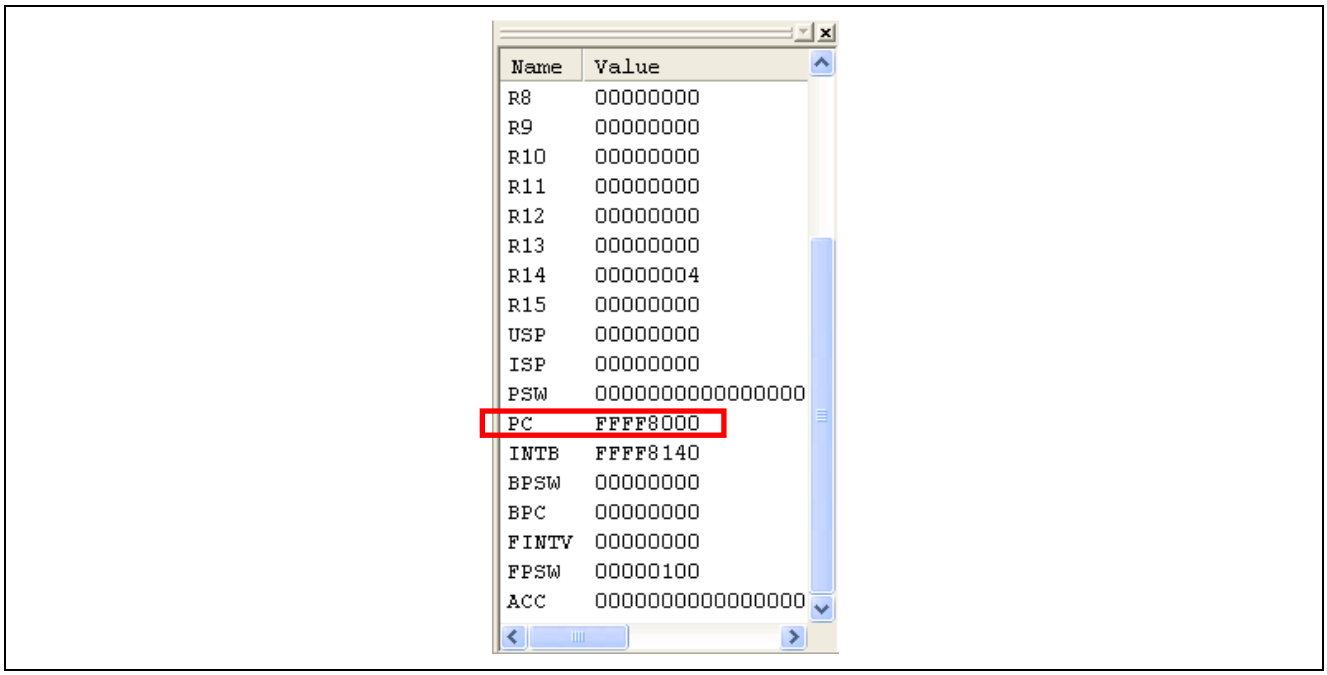


図 4.46　レジスタウィンドウの参照

注意：実行中のレジスタの値は参照することはできません。

(3) レジスタの値は、レジスタの上でダブルクリックするか、シングルクリックすることで変更が可能です。

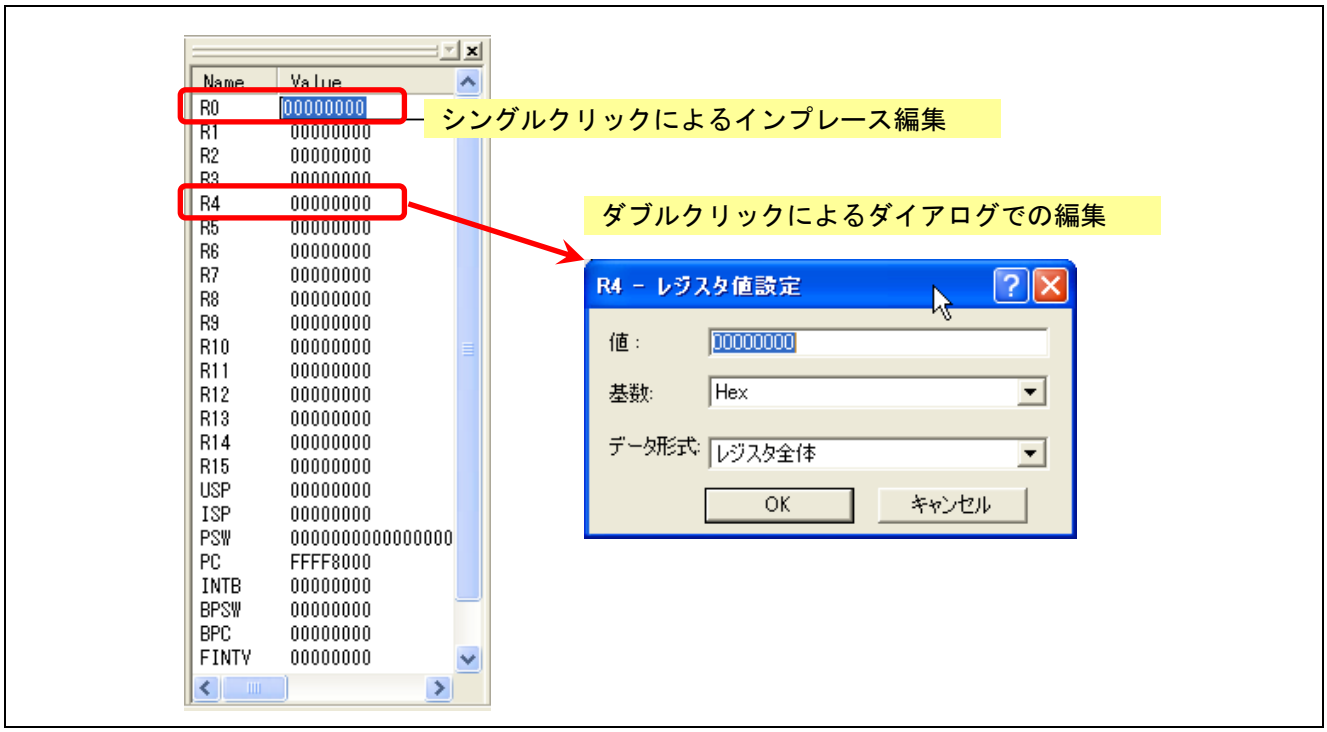


図 4.47　レジスタ値の変更

4.14　I/O ウィンドウによる I/O レジスタ値の参照／設定

I/O レジスタの値を参照／設定する場合、I/O ウィンドウを使用します。

(1) 図 4.48のように表示->CPU->I/Oと選択し、I/Oウィンドウを開いてください。

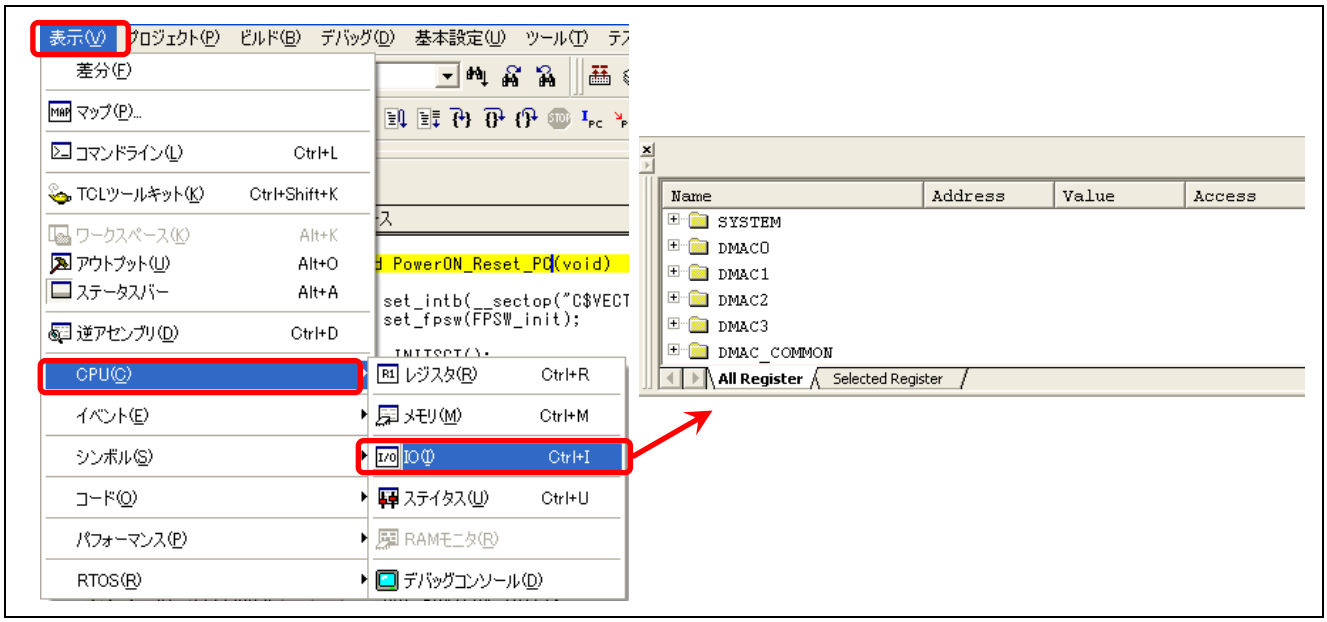


図 4.48　I/O ウィンドウの表示

(2) コンペアマッチタイマ 0 (CMT0)のコンペアマッチタイマコントロールレジスタ(CMCR)の値を参照してみましょう。

図 4.49のようにCMT0 を選択しモジュールを展開するとし、値を確認することができます。
この値はブレークしたときの値ですので、現在の値を参照したい場合は、右クリックメニューから「最新の状態に更新」を選択してください。

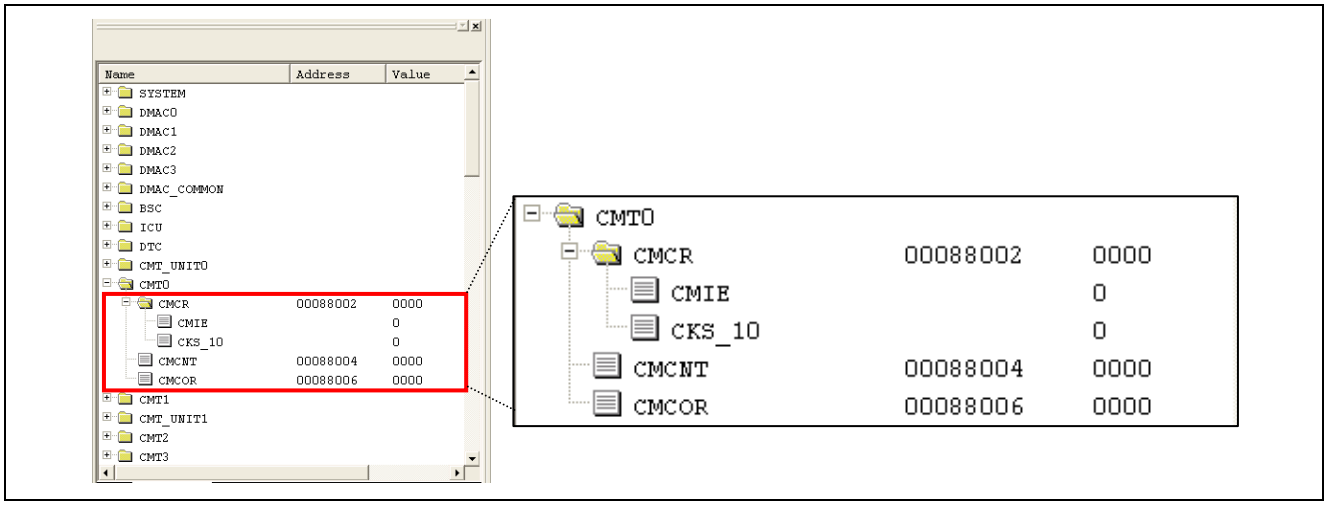


図 4.49　I/O ウィンドウでの値の参照

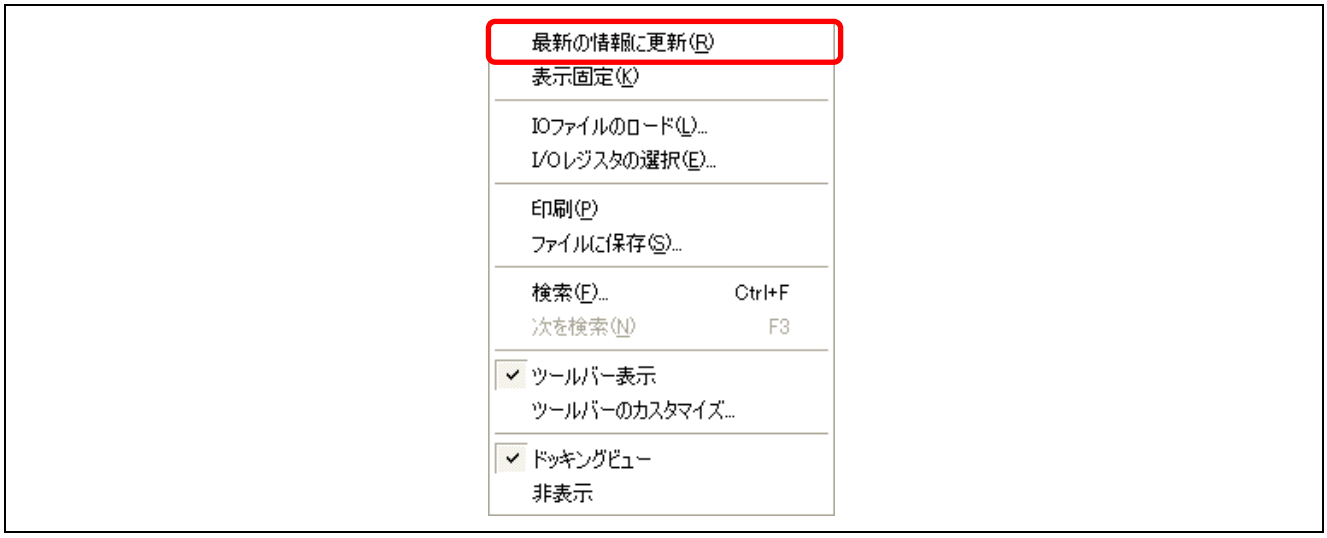


図 4.50　最新の情報に更新メニュー

(3) I/O レジスタの値は、I/O レジスタ変数の上でダブルクリックすることで変更が可能です。

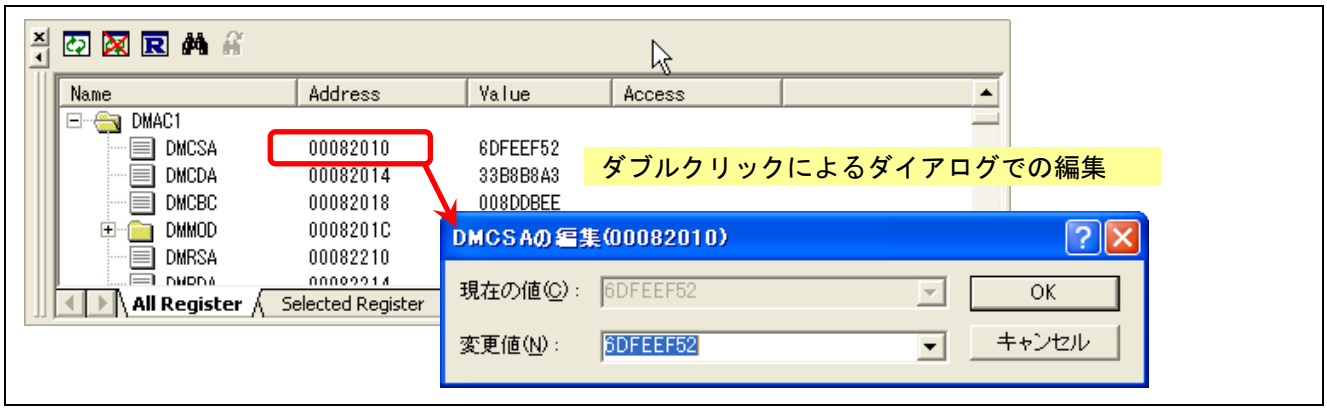


図 4.51　I/O ウィンドウでの値の変更

(4) 任意のレジスタのみを表示したい場合は、Selected Register タブを使用します。

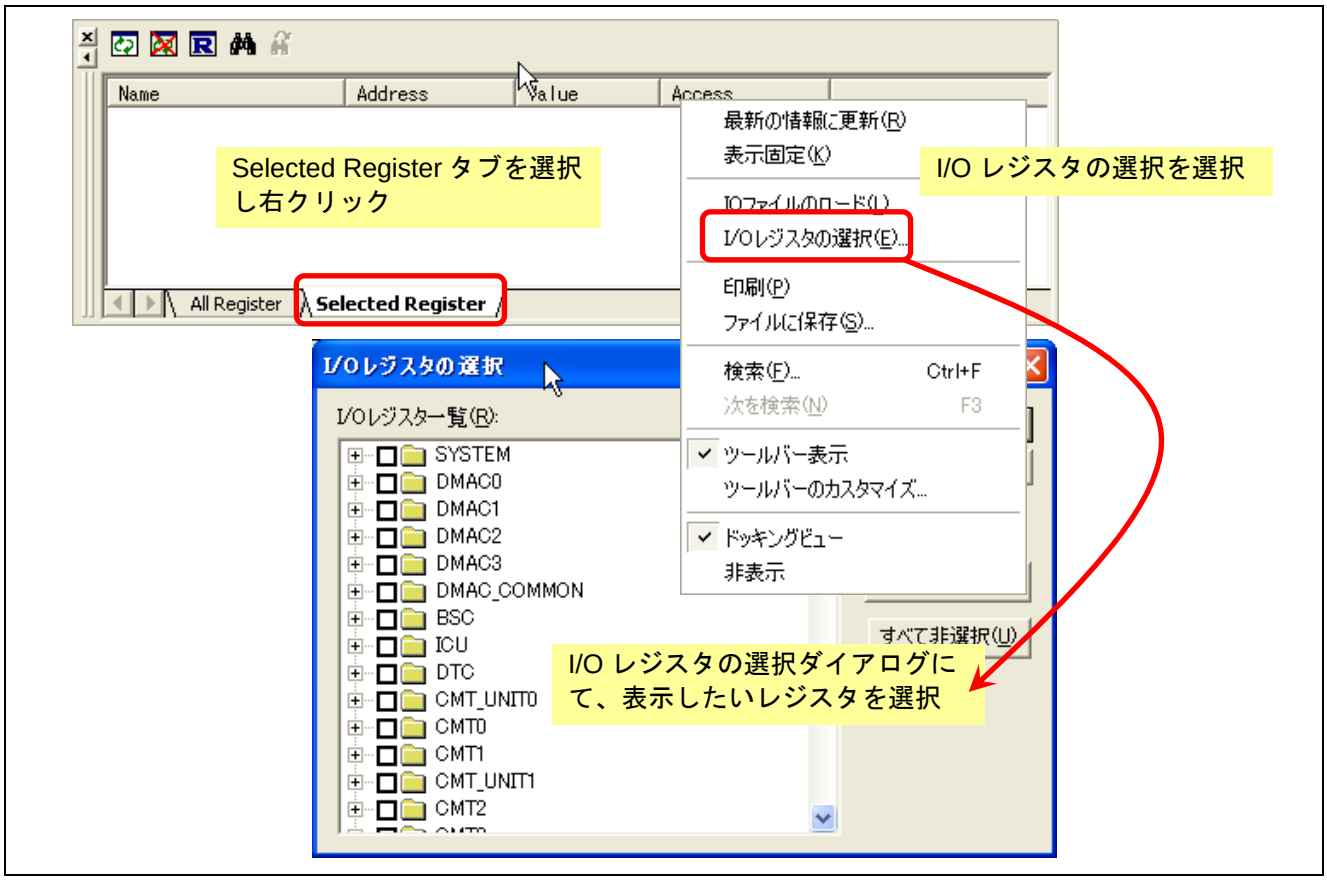


図 4.52　Selected Register タブ

4.15　パフォーマンス解析

E1 エミュレータでは、ユーザが設定した最大 2 区間の経過サイクル（処理時間）、割り込み/例外処理の経過サイクル（処理時間）、または割り込み/例外の発生回数などの測定が可能です。

(1) パフォーマンス解析ウィンドウを表示します。
以下の図 4.53のように表示->パフォーマンス->パフォーマンス解析と選択し、パフォーマンス解析ウィンドウを表示してください。

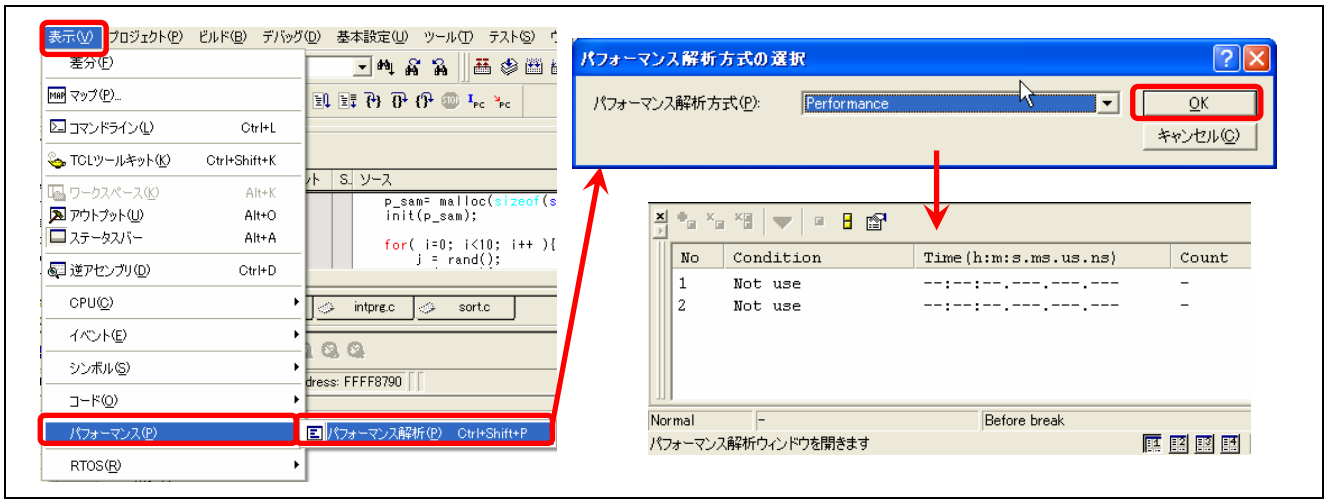


図 4.53　パフォーマンス解析ウィンドウの表示

(2) 図 4.54のようにパフォーマンス解析ウィンドウの“[No.1] Not use”を選択し、ダブルクリックすることで、パフォーマンスの条件設定ウィンドウが開きます。

(3) パフォーマンス条件設定ウィンドウのプルダウンメニューに表示されている計測項目が選択可能です。

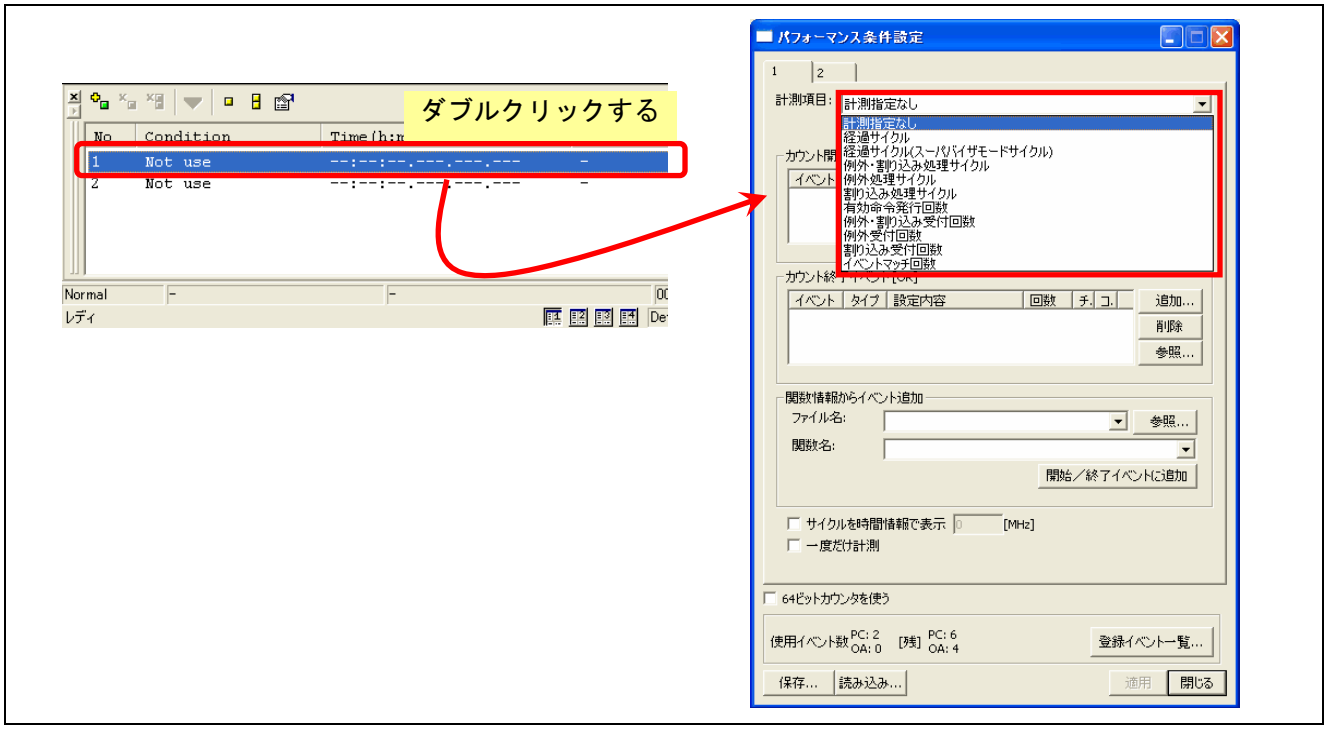


図 4.54　パフォーマンス条件設定ウィンドウの表示

(4) ここでは、sort 関数と change 関数の実行時間を測定してみましょう。
パフォーマンス条件を以下のように設定して下さい。
カウント開始・終了イベントには関数名のドラッグ＆ドロップが使用できます。

1 パフォーマンス条件設定ダイアログの計測項目を「経過サイクル」に設定
2 エディタウィンドウにてsortを選択し、パフォーマンス条件設定ダイアログのカウント開始イベント欄へドラッグ＆ドロップ
3 パフォーマンス条件設定ダイアログのカウント終了イベント欄の追加ボタンを押して、イベント設定ダイアログをオープン
4 イベント設定ダイアログにて、計測終了アドレスとして、Tutorial.cの 59 行目のアドレス「FFFF87E4」を入力
5 イベント設定ダイアログの「OK」ボタンを押す
6 パフォーマンス条件設定ダイアログの「サイクルを時間情報で表示」をチェックし、周波数欄に「25」と入力、さらに「一度だけ計測」をチェック。
「サイクルを時間情報で表示」を使用することで、測定した経過サイクル数を時間に換算することができます。チュートリアルプログラムの設定では、CPU のクロックは 25MHz で動作しますので、25 と入力しています。
また、「一度だけ計測」を使用することで、繰り返し成立する条件下においても、最初の一回だけ計測することができます。この機能を使用しない場合は、累積での測定となります。
7 最後に、設定内容を確認して、パフォーマンス条件設定ダイアログの「適用」ボタンを押す

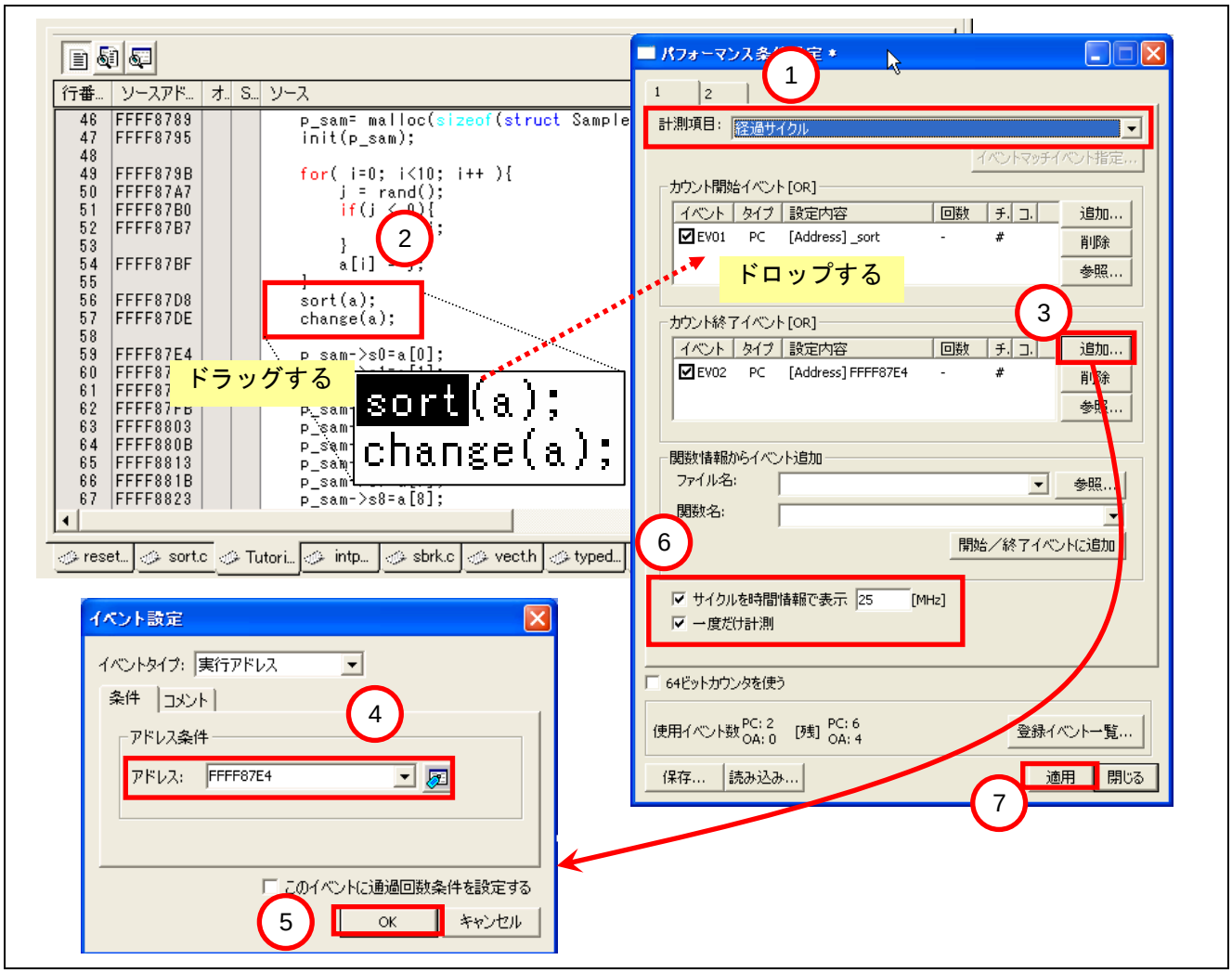


図 4.55 パフォーマンス条件の設定

注意：この例では、sort および change 関数が、この場所からしか呼び出されないことを想定しています。

(5) 図 4.56のようにパフォーマンス解析ウィンドウに設定が反映されますので、確認して下さい。

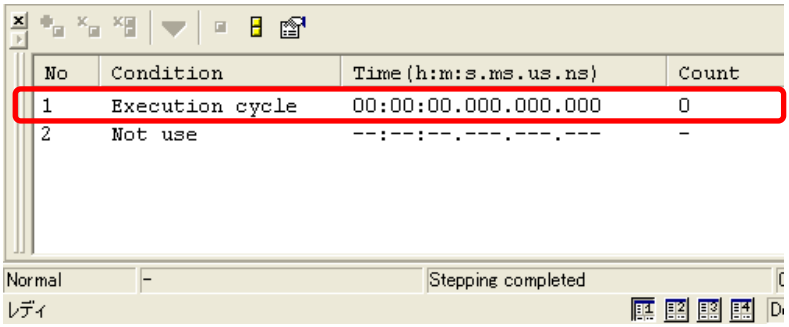


図 4.56　条件設定後のパフォーマンス解析ウィンドウ

(6) ウィンドウの更新が確認できましたら、[リセット後実行]ボタンよりプログラムを実行してください。
設定した計測区間を通過すると、パフォーマンス解析画面が図 4.57のように更新されます。

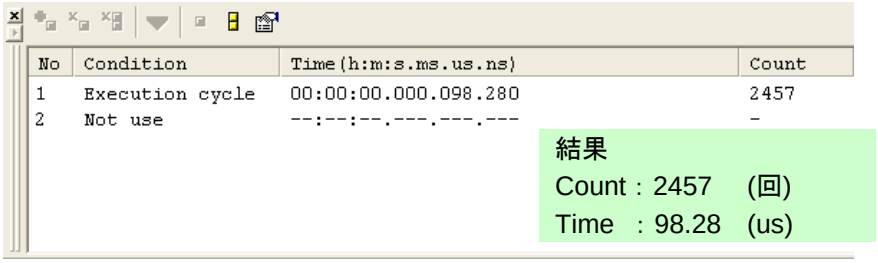


図 4.57　パフォーマンス解析結果

(7) 測定した結果をクリアする場合には、ブレーク後に「すべてクリア」ボタンを押します。
測定結果をクリアしない場合、次回の実行時にその結果が累積されます。

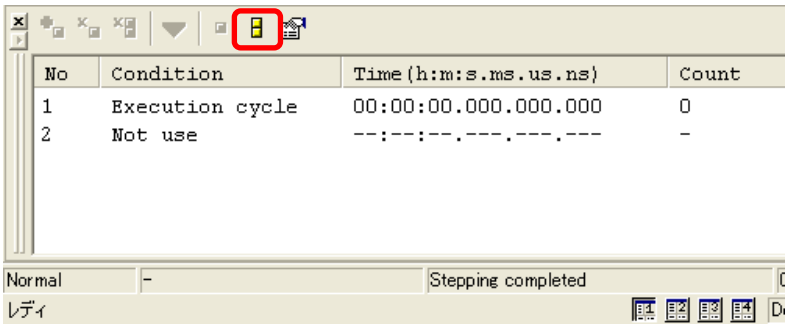


図 4.58　パフォーマンス解析結果のクリア

4.16　トレース機能

トレース機能では、ユーザプログラム実行中に、分岐 PC や指定したデータアクセス情報を取得し、トレースメモリ内に保持する機能です。そのトレース情報を使用してアプリケーションの実行の流れを追跡し、問題発生ポイントを調べることができます。

E1 エミュレータでは内蔵トレース機能が使用できます。

内蔵トレース機能は、マイコンに内蔵されているトレースバッファを使用して実現します。

イベント機能（実行 PC イベント、オペランドアクセスイベント）により、CPU バスを対象とした分岐トレース、データトレースが可能で、分岐元/分岐先アドレス情報、データアクセス情報を取得することができます。また、取得したトレース結果はトレースウィンドウ上でバス/逆アセンブリ/ソースレベルの形式で表示することができます。

(1) 図 4.59のように表示->コード->トレースと選択し、トレースウィンドウを表示します。

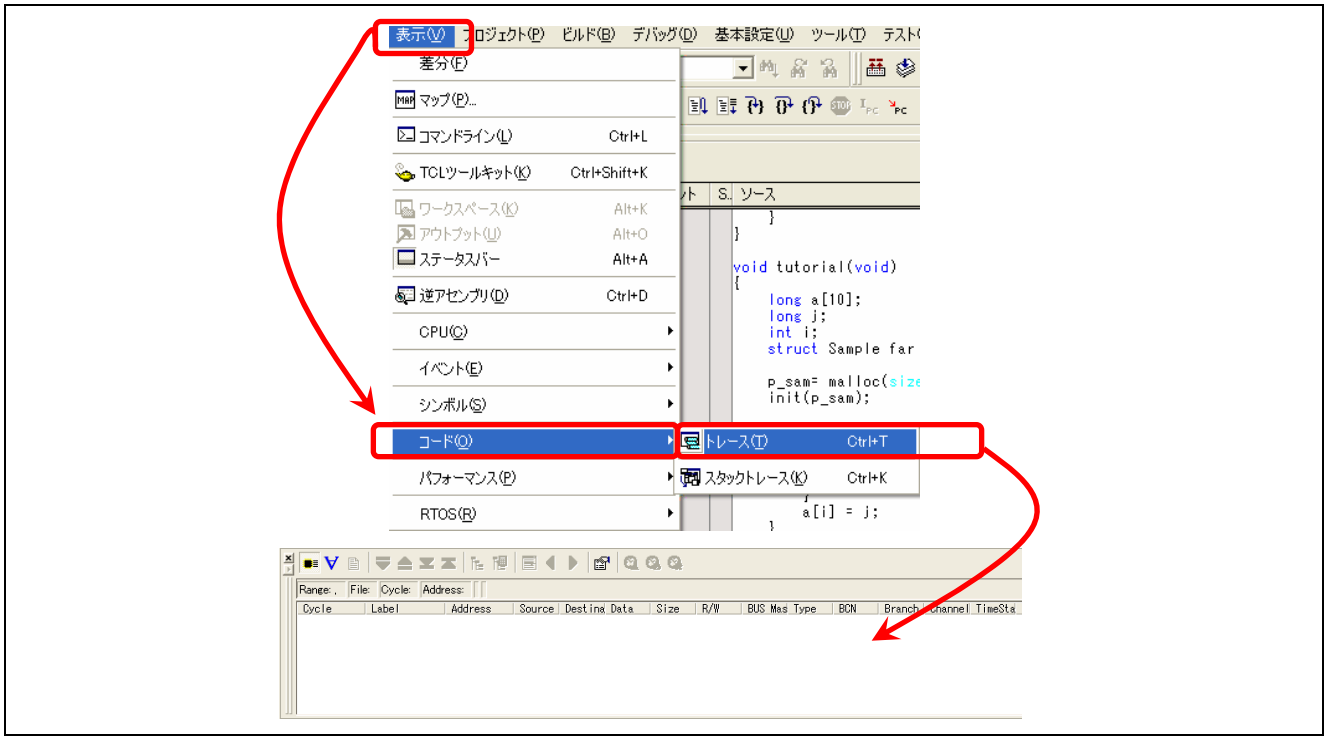


図 4.59　トレースウィンドウの表示

(2) トレース結果を取得する条件を設定するため、トレース条件設定ウィンドウを表示します。図 4.60のように、トレースウィンドウから[トレース設定]ボタンを押します。

図 4.60　トレース条件設定ボタン

(3) トレース条件設定画面で条件の設定をします。
E1 エミュレータを使用する場合、表 4.3のようなトレース条件を選択できます。
本章では図 4.61のように設定してください。この設定により、ユーザプログラムが停止する直前の 256 分岐PCを記録することができます。

表 4.3　トレース条件設定内容（E1 エミュレータ使用時）

トレースタブ		
取得モード		
	フリー	プログラムが停止するか、トレース取得停止ポイントが成立するまで記録します。
	フル	トレースバッファがいっぱいになるまで記録します。
出力モード		
	外部出力しない	MCU 内蔵トレースバッファを使用します。 E1 エミュレータではその他設定の選択不可です。
データ種別		
	分岐	分岐時のトレースデータを取得します。
	分岐 ＋ データアクセス	分岐時、またはデータアクセス時にトレースデータを取得します。
	データアクセス	データアクセス時にトレースデータを取得します。
トレース容量		
	設定不可	

注意：その他のトレース取得設定条件に関しては、マニュアルを参照してください

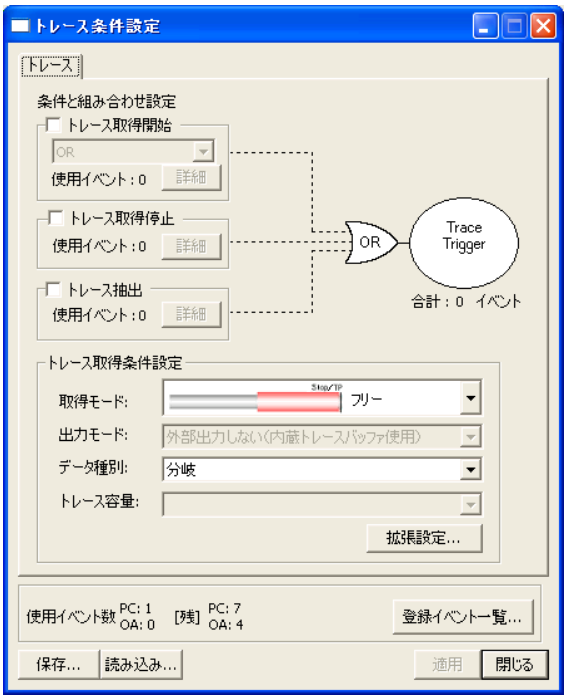


図 4.61　トレース条件設定

(4) プログラムを停止するため、図 4.62のように、Tutorial.cの 59 行目の「p_sam->s0=a[0];」の処理が記述されている部分にオンチップブレーク設定をします。ブレークポイントの設定方法に関しては、「4.10　ブレーク機能」を参照して下さい。

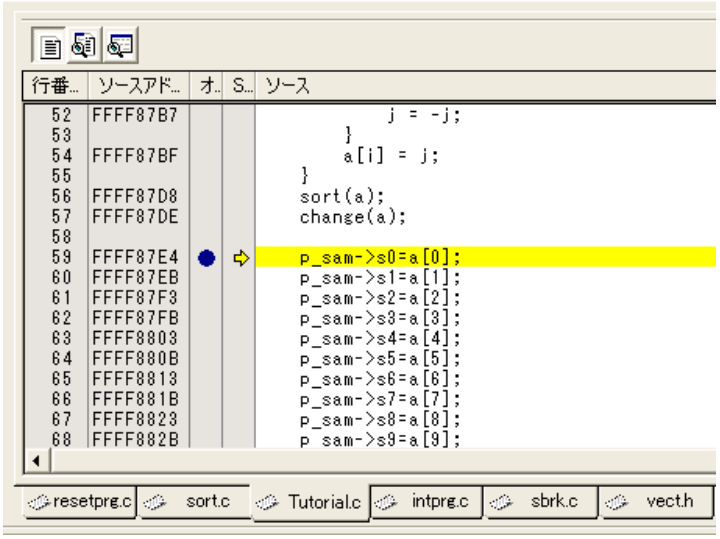


図 4.62　ブレークポイントの設定

(5) [リセット後実行]ボタンを押して、プログラムを実行します。
設定したブレークポイントでプログラムが一時中断すると、トレースウィンドウが更新されます。
表示されたトレースウィンドウからプログラムの軌跡を辿ることができます。

(6) 表示モード切替ボタンを押して、バス表示モード(図 4.63)からソース表示モード(図 4.64)へ表示を変更してください。

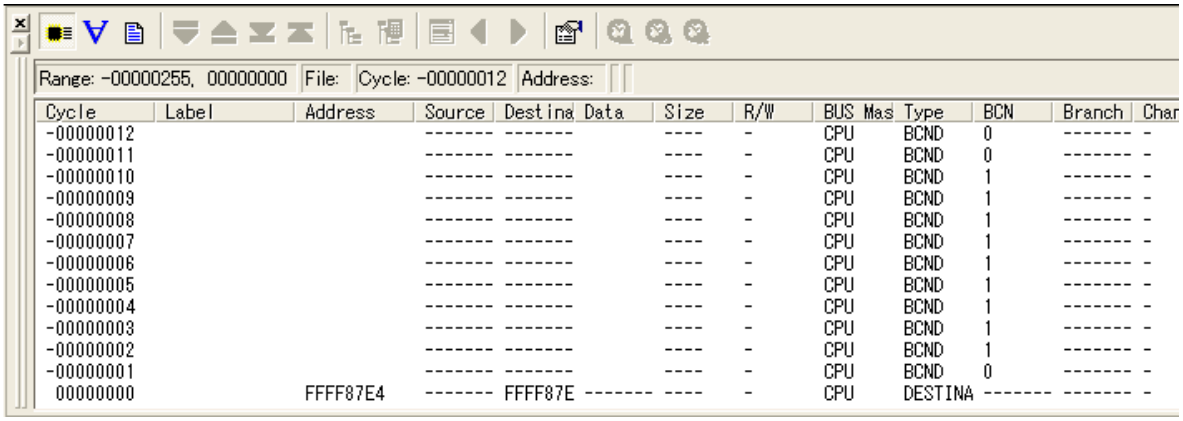


図 4.63　ブレーク後のトレースウィンドウ（バス表示モード）

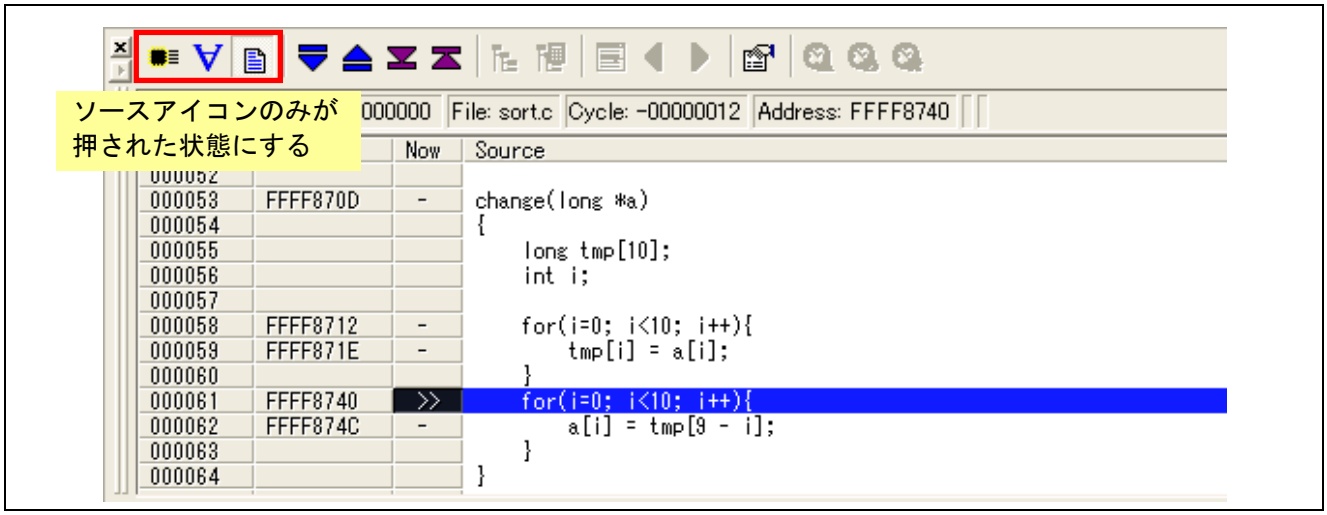


図 4.64　ソース表示モード

(7) トレースウィンドウの表示モードについて図 4.65に示します

図 4.65は、バス表示モード、逆アセンブラ表示モードおよびソース表示モード全て有効に設定されたウィンドウです。

このようにトレース表示モード切替ボタンによって、トレースウィンドウに表示する情報を選択できます。

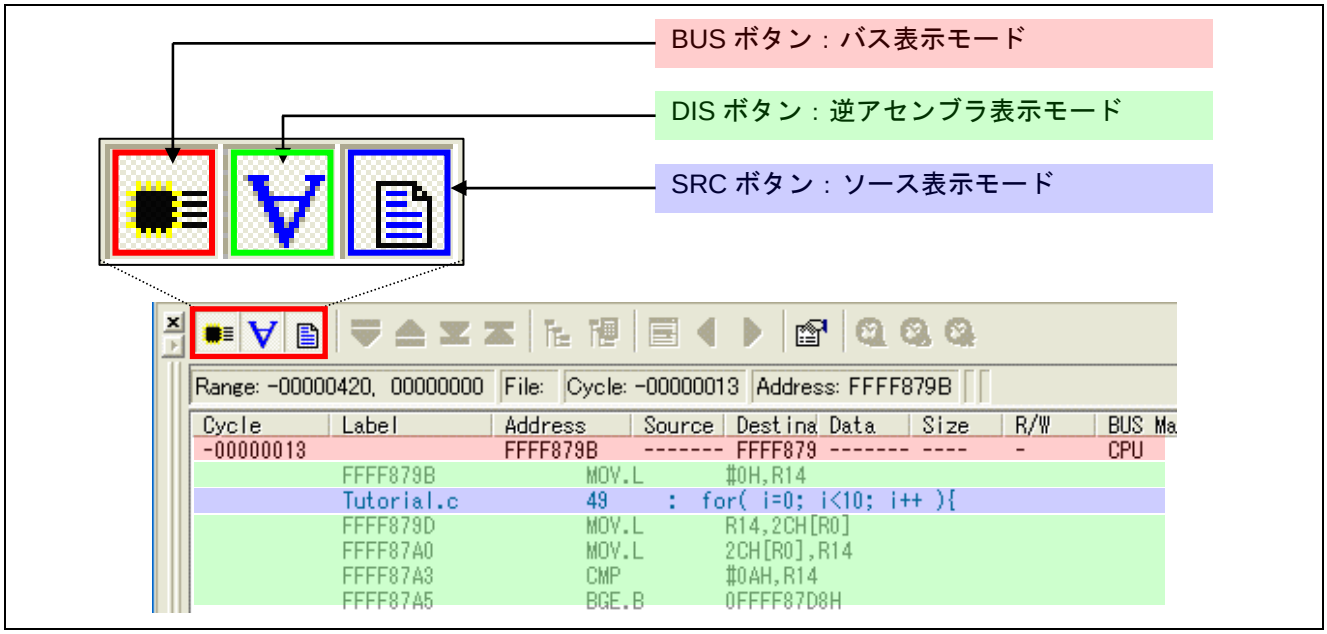


図 4.65　トレースウィンドウの表示モード

(8) ソース表示モードでは、トレース情報を解析してプログラムの実行履歴を、図 4.66のように逆方向にステップして追いかける[バックステップ機能]、図 4.67のように順方向への[ステップ機能]が使用できます。この例では、change 関数内のループ箇所にあるため、ステップボタンを押しても現在の位置が変わらない場合があります。その場合は、ウィンドウの右上の Cycle と Address を確認してください。この値が変わっている場合は、ループ中にあることがわかります。

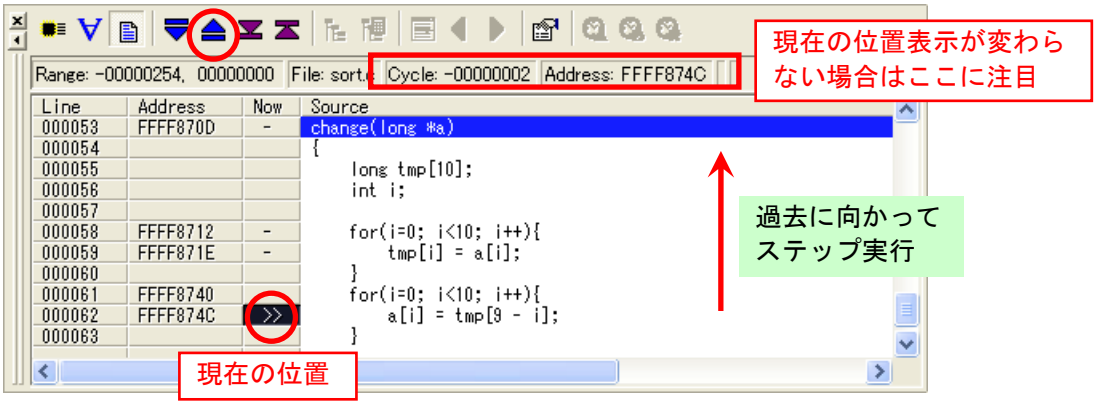


図 4.66　逆方向ステップ

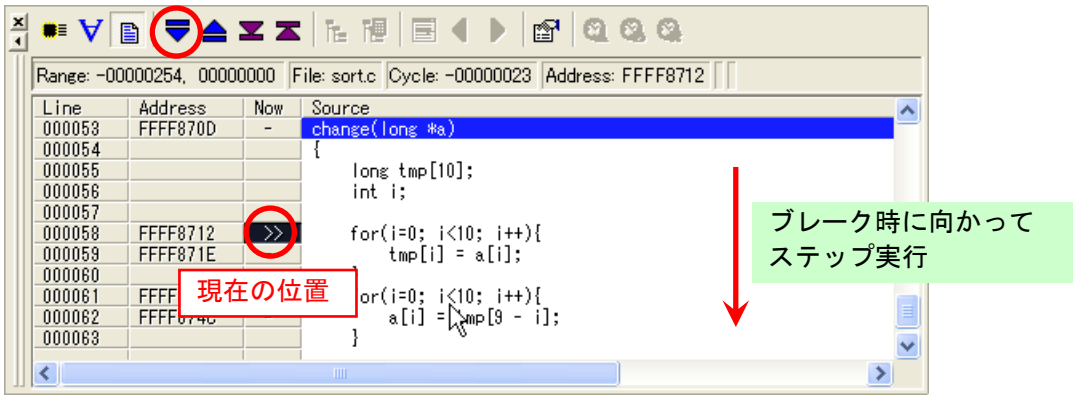


図 4.67　順方向ステップ

4.17　デバッグの終了

デバッグを終了するには、ウィンドウ右上の×マークを押してください。ダイアログが表示されたら「はい」を選んでください。これにより、ウィンドウの状態などが保存され、次回デバッグ時に同じウィンドウの状態でデバッグを開始することができます。

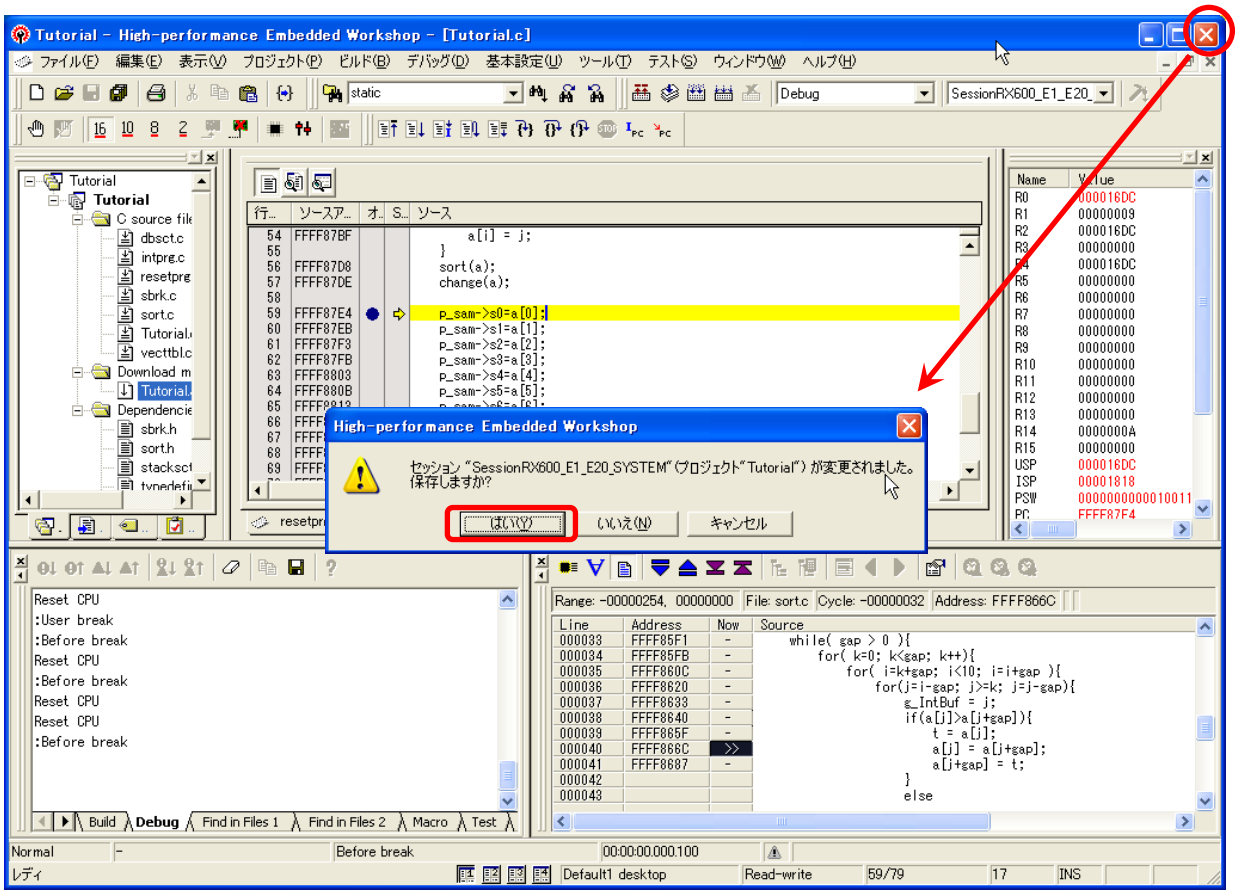


図 4.68　デバッグの終了

4.18　ユーザボード単体での動作確認

デバッグが終了し、ユーザボード単体でプログラムを動作させる場合には、以下の図のようにフラッシュライタモードを使用してプログラムをダウンロードしてください。（すでに説明している操作の説明は省略しています。）

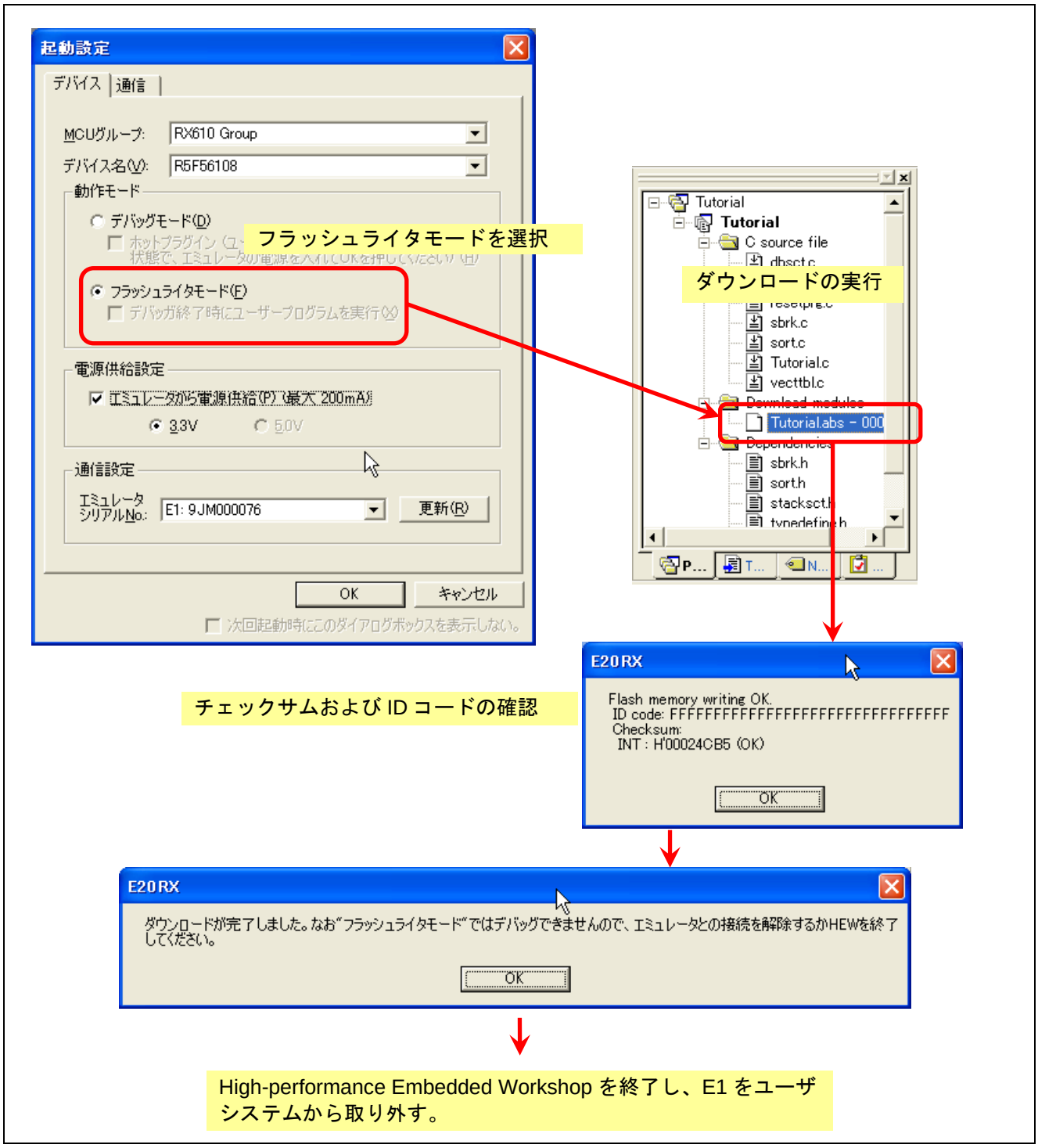


図 4.69　ユーザボード単体での動作確認手順

5. 補足事項

5.1 エミュレータ関連ドキュメント

E1 エミュレータおよび、High-performance Embedded Workshop には本書で取り上げた機能以外にも便利な機能を豊富に備えています。各製品の仕様の詳細については、下記の関連ドキュメントを参照してください。

(1) High-performance Embedded Workshop ユーザーズマニュアル
High-performance Embedded Workshop 共通の機能（ビルド、ファイル操作、共通のデバッグ機能など）について記載しています。

(2) RX ファミリ用 E1/E20 エミュレータ ユーザーズマニュアル
E1/E20 エミュレータ共通の使用方法および注意事項を記載しています。
E1/E20 エミュレータ固有デバッグ機能については、ユーザーズマニュアル別冊 (RX610 グループ ご使用時の補足説明)を参照してください。

(3) E1/E20 エミュレータ ユーザーズマニュアル別冊 (RX610 グループ ご使用時の補足説明)
E1/E20 エミュレータを RX610 グループの MCU と組み合わせて使用するときの注意事項を記載しています。本マニュアルはマイコングループ毎に用意されています。

(4) RX ファミリ C/C++コンパイラパッケージ ユーザーズマニュアル
C/C++コンパイラ、アセンブラ、リンカなどについての説明が書かれています。

各ドキュメントを参照するは、[スタート]メニューから[すべてのプログラム] -> [Renesas] -> [High-performance Embedded Workshop] -> [Manual Navigator] を選び、Manual Navigator を起動してください。

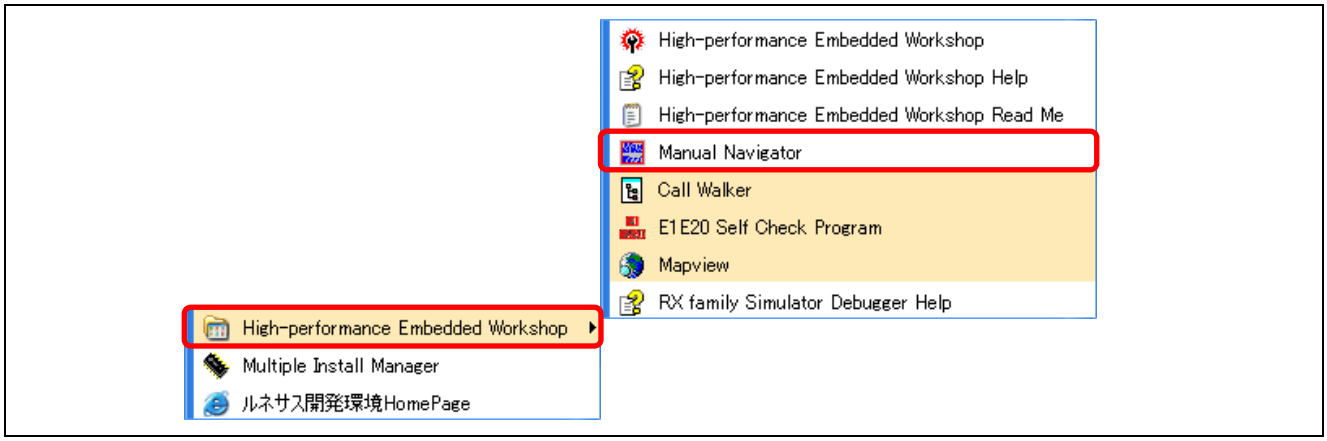


図 5.1 Manual Navigator の起動

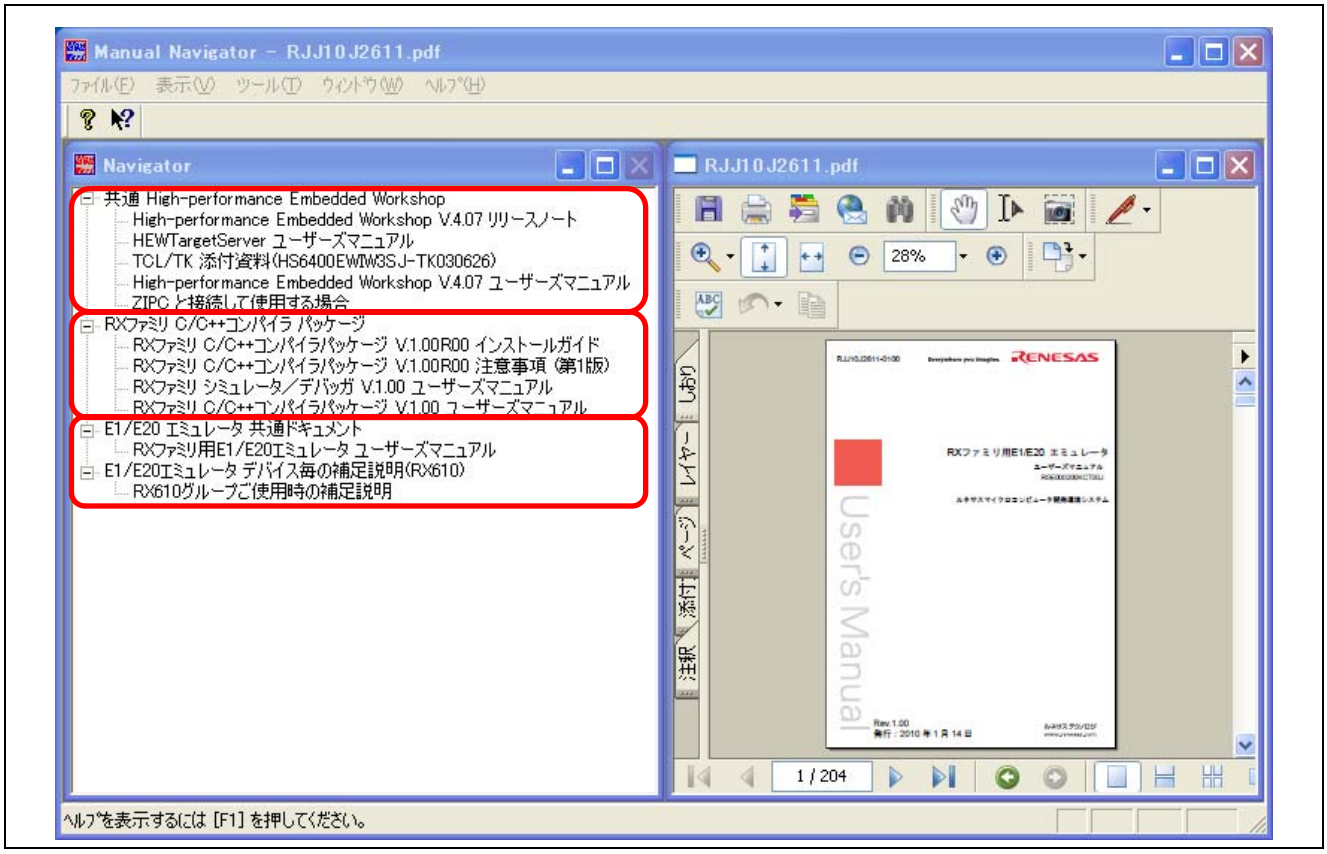


図 5.2　Manual Navigator の画面

また、技術情報、制限事項など有用な情報を Web に掲載していますので以下の URL も参考にしてください。

http://japan.renesas.com/e1	オンチップデバッギングエミュレータE1
http://japan.renesas.com/hew	High-performance Embedded Workshop
http://japan.renesas.com/rx_c	RXファミリ用C/C++コンパイラパッケージ

5.2 無償評価版 RX ファミリ用 C/C++コンパイラパッケージ

5.2.1 無償評価版 C コンパイラの制限事項

無償評価版 C コンパイラは、製品版のコンパイラのご購入前に、コンパイラの性能を評価していただくための製品です。無償評価版Cコンパイラには以下の制限事項があります。

(1) 初めて評価版ソフトウェアをインストールした後、最初にビルドを行った日から 60 日間は使用制限ありません。

(2) 61 日以降はリンクサイズが 128KB 以内に制限されます。期限切れになった場合、アウトプットウィンドウに下記のメッセージが表示されます。

```
Software license problem: Duration of Trial License of UNKNOWN is exhausted.
Maximum link size limited to 128KB code+data.
```

5.2.2 無償評価版 C コンパイラのダウンロード方法

無償評価版 C コンパイラは、以下の URL からダウンロード可能です。

http://japan.renesas.com/tool_evaluation

RX マイコンの評価用には、「RX ファミリ用 C/C++コンパイラパッケージ」をダウンロードしてください。

図 5.3 評価版ソフトウェアダウンロードサイト

5.3 E20 エミュレータに関して

E1 エミュレータよりも高機能版のエミュレータに、E20 エミュレータがあります。

E20 エミュレータでは、エミュレータに大容量のトレースメモリを搭載することで、大容量トレース（2M 分岐またはサイクル）を実現しています。また、トレース情報を使用することでリアルタイムにメモリ内容を参照可能な「リアルタイム RAM モニタ機能」など、強力なデバッグ機能をサポートしています。

表 5.1　E1 とE20 の違い

エミュレータ機能		E1 （R0E000010KCE00)	E20 (R0E000200KCT00)
イベント	実行 PC	8ch	
	オペランドアクセス	合計 4ch （CPU バスが対象） 3ch：アドレス/データマスク可、1ch：アドレスマスク or 範囲/データマスク可 アクセスサイズ(1,2,4 バイト)およびアクセス(R,W,R/W)指定可能	
	組み合わせ	OR/累積 AND/状態遷移、 8 ビットパスカウンタ（いずれか 1 つのイベントに指定可能）	
ブレーク	ソフトウェアブレーク	256ch	
	オンチップブレーク	イベントを使用	
トレース	容量	256 分岐 or サイクル	Min 2M 分岐 or サイクル
	トレースデータタイプ	分岐、オペランドアクセス（CPU バスが対象）	
	データトレース条件	イベントで設定可能	
	開始/終了条件	プログラムの実行停止、イベントによる条件設定	
	トレースデータの表示	逆アセンブル／C ソース表示も可能 C ソース表示モードではバックステップ機能もサポート	
RAM モニタ	メモリウィンドウによる自動更新	○ （サイクルスチールによる実現、 1 ワード参照／設定毎に数サイクルのプログラム停止発生）	
	リアルタイム RAM モニタ	×	Max 4Kバイト（開発中） （データ、最終アクセス履歴） DMA/DTC によるアクセスは監視不可。
パフォーマンス計測		32 ビットカウンタ×2 本 または 64 ビットカウンタ×1 本 （経過サイクル、例外/割り込み処理サイクル/回数、命令実行回数、イベント回数）	

E20 エミュレータの詳細に関しては、「RX ファミリ用 E1/E20 エミュレータ ユーザーズマニュアル」を参照してください。

6. よくある質問

(1) ワークスペース読み込み後に図 6.1の「Toolchain ‘Renesas RX Standard Toolchain’, version ‘x.xx.x’ is missing」というワーニングメッセージが表示される。
このメッセージは、対応するコンパイラがインストールされていないか、ワークスペースを作成したときに使用していたバージョンのコンパイラがインストールされていないことを示します。
対応するコンパイラをインストールするか、指示に従って、コンパイラのバージョンを選択してください。

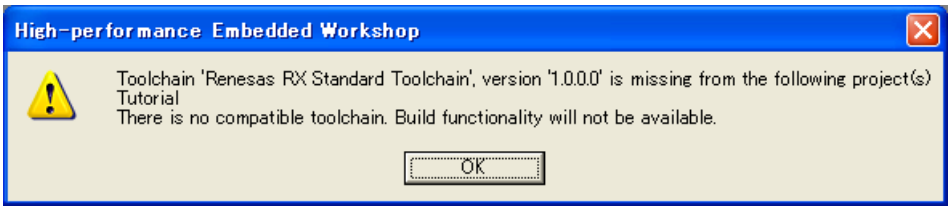


図 6.1 ワーニングメッセージ

(2) デバッガ起動時に図 6.2の「デバイスIDコードが選択したデバイスのものと一致しません。」というワーニングが表示される。
起動設定ダイアログで設定したデバイス名と、E1 に接続されているユーザシステムに登載しているデバイス名が異なっています。起動設定ダイアログにて、実際に接続されているデバイス名を指定してください。

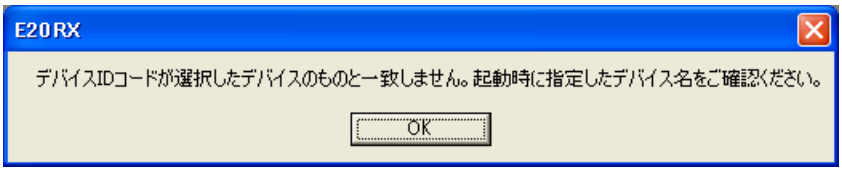


図 6.2 ワーニングメッセージ

(3) デバッガ起動時に図 6.3の「ファームウェアの書き換え」ダイアログが表示される。
エミュレータが対応しているファームウェアのバージョンと、E1 エミュレータ内のファームウェアのバージョンが異なっている場合に表示されます。この組み合わせでデバッグを行う場合は、表示されているバージョンにかかわらず「はい」ボタンを押してください。

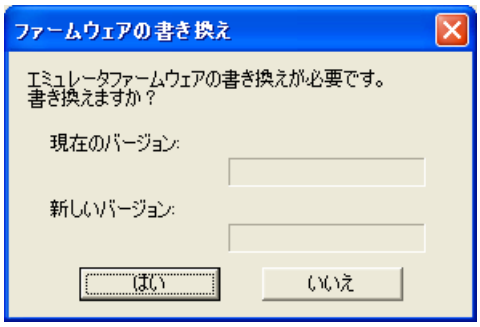


図 6.3 ファームウェアの書き換えダイアログ

(4) デバッガは正常に起動できたが、ダウンロード後に正しくプログラムが表示されない。

RX600 ファミリのマイコンは、マイコンが使用するエンディアンを選択することが可能です。プログラム作成時に指定したエンディアンと、マイコンに設定されているエンディアンが異なっていると、ダウンロード後に正しくリセットベクタの値を参照できず、ただしくプログラムが表示されない場合があります。

プログラム作成時に指定したエンディアンは、High-performance Embedded Workshop の[ビルド]->[RX Standard Toolchain]メニューによりオープンする、RX Standard Toolchain の CPU タブ（コンパイラの設定）にて確認できます。

マイコンに設定されているエンディアンは、コンフィグレーションプロパティダイアログのエンディアン欄に表示されます。

エンディアンが一致していない場合は、コンパイラまたはマイコンの設定の設定を変更してください。

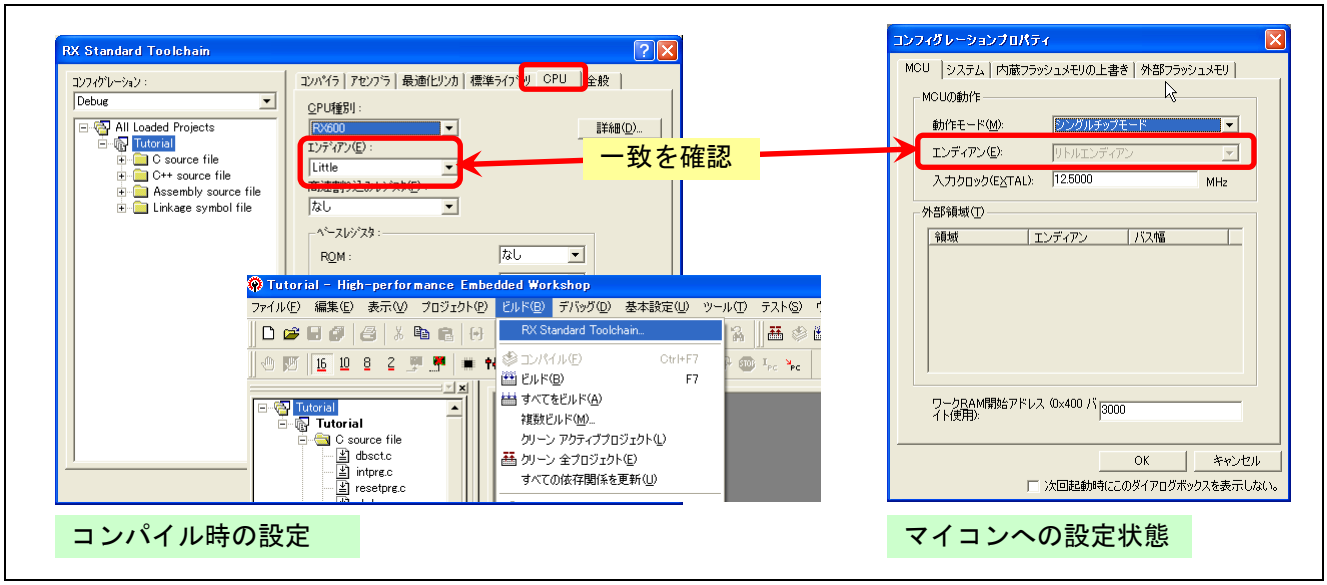


図 6.4　エンディアン一致の確認

(5) デバッグコマンドを実行しても受け付けられない。

起動設定ダイアログにて、フラッシュライタモードを選択していませんか？アウトプットウィンドウに以下のような表示が出ていれば、フラッシュライタモードでエミュレータが動作しています。フラッシュライタモードでは、ダウンロードのみ行うことができます。デバッグコマンドを実行するためには、デバッグモードを選択してください。

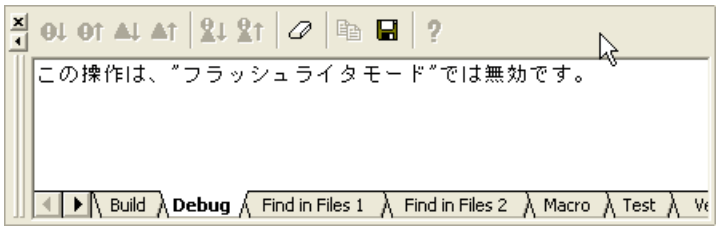


図 6.5　アウトプットウィンドウの表示

(6) Windows® XP 環境において、E1/E20 エミュレータソフトウェア V.1.00 Release00 をインストール後、E1/E20 エミュレータ以外のエミュレータで接続できなくなった。

以下の手順で、USB ドライバの動作状態を確認してください。

エミュレータをホストマシンに接続し、エミュレータに電源を投入する。

Windows のデバイスマネージャでデバイスリストを表示する。

ルネサス製 USB ドライバのプロパティを開き、全般タブを開く。

「デバイスの状態」欄に「このデバイスのドライバを再インストールする必要があります。」と表示されていた場合、下記手順で、USB ドライバをインストールしてください。

1 デバイスマネージャ上から Renesas Emulator カテゴリ内にある Renesas E-Series USB Driver を右クリックし、メニュー内の[削除]を 選択する。
2 エミュレータ電源を OFF する。 (注意)
3 エミュレータ電源を ON する。 (注意)
4 新しいハードウェアの検出ウィザードが開始される。 "ソフトウェアを自動的にインストールする(推奨)" を選択し、「次へ」ボタンをクリックする。
5 新しいハードウェアの検出ウィザードに 「下の一覧からハードウェアに最適なソフトウェアを選んでください。」 と表示される。
6 複数の inf ファイルが表示される場合は、何れでもよいので一つを選択し、「次へ」ボタンをクリックする。
7 新しいハードウェアの検出ウィザードに 「次のハードウェアのソフトウェアのインストールが完了しました。」 と表示されたことを確認し、「完了」ボタンをクリックする。

注意：エミュレータ電源の OFF/ON について
オンチップエミュレータ製品の中には、ホストマシンに USB ケーブル接続することでエミュレータに電源を供給する製品があります。該当するエミュレータ製品は、USB ケーブルを抜き差しし、エミュレータ電源の OFF/ON を行ってください。

以下のエミュレータが対象となります。E8a は対象外です。

— E10A-USB
— E1
— E20
— E8
— E7
— E10T-USB
— E100
— E30A
— E200F

(7) その他のよくあるお問い合わせ

E1 エミュレータをはじめ、ルネサス製品に関するお客様からの質問事項および回答内容をまとめてウェブサイト上に「よくあるお問い合わせ」として掲載しております。

E1 エミュレータのページからリンク内にある「よくあるお問い合わせ」をクリックしてご利用ください。

ホームページとサポート窓口

ルネサス エレクトロニクスホームページ
http://japan.renesas.com/

お問合せ先
http://japan.renesas.com/inquiry

すべての商標および登録商標は，それぞれの所有者に帰属します。

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2010.06.01	—	初版発行
1.01	2010/09/07	—	図と文章の内容差異修正 図表番号抜け修正 補足事項の文章切れ修正

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。

RENESAS

ルネサス エレクトロニクス株式会社

■営業お問合せ窓口　http://www.renesas.com

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

ルネサス エレクトロニクス販売株式会社　〒100-0004　千代田区大手町2-6-2（日本ビル）　(03)5201-5307

■技術的なお問合せおよび資料のご請求は下記へどうぞ。
総合お問合せ窓口：http://japan.renesas.com/inquiry

© 2010 Renesas Electronics Corporation. All rights reserved.
Colophon 1.0