

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ひ申し上げます。

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>

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のですが、誤りが無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、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と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。

SH7046 E6000H エミュレータ

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

SuperH™ファミリ / SH7046 シリーズ

SuperH™ファミリ / SH7047 シリーズ

SuperH™ファミリ / SH7144 シリーズ

HS7046EPH60HJ

安全設計に関するお願い

1. 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注意ください。

本資料ご利用に際しての留意事項

1. 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ルネサス テクノロジ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。
2. 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、ルネサス テクノロジは責任を負いません。
3. 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、ルネサス テクノロジは、予告なしに、本資料に記載した製品また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すは、事前にルネサス テクノロジ、ルネサス販売また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。
4. 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。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、ルネサス テクノロジはその責任を負いません。
5. 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。ルネサス テクノロジは、適用可否に対する責任を負いません。
6. 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、ルネサス テクノロジ、ルネサス販売または特約店へご照会ください。
7. 本資料の転載、複製については、文書によるルネサス テクノロジの事前の承諾が必要です。
8. 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ルネサス テクノロジ、ルネサス販売または特約店までご照会ください。

重要事項

- 本エミュレータをご使用になる前に、必ずユーザーズマニュアルをよく読んで理解してください。
- ユーザーズマニュアルは、必ず保管し、使用上不明な点がある場合は再読してください。
ユーザーズマニュアルをよく読まずに、本エミュレータを使用しないでください。

エミュレータとは：

ここで言うエミュレータとは、株式会社ルネサス テクノロジ（以下、「ルネサス」という。）が製作した次の製品を指します。
（1）エミュレータ本体、（2）PC インタフェースボード、（3）ユーザシステムインタフェースボード、（4）ケーブル
お客様のユーザシステムおよびホストコンピュータは含みません。

エミュレータの使用目的：

本エミュレータは、ルネサスマイクロコンピュータを使用したシステムの開発を支援する装置です。ソフトウェアとハードウェアの両面から、システム開発を支援します。

この使用目的に従って、本エミュレータを正しく使用してください。目的以外の本エミュレータの使用を堅くお断りします。

使用制限：

本エミュレータは、開発支援用として開発したものです。したがって、機器組み込み用として使用しないでください。また、以下に示す開発用途に対しても使用しないでください。

1. ライフサポート関連の医療機器用（人命にかかわる装置用）
2. 原子力開発機器用
3. 航空機開発機器用
4. 宇宙開発機器用

このような目的で本エミュレータの採用をお考えのお客様は、当社営業窓口へ是非ご連絡頂きますようお願い致します。

製品の変更について：

ルネサスは、本エミュレータのデザイン、性能を絶えず改良する方針をとっています。したがって、予告なく仕様、デザイン、およびユーザーズマニュアルを変更することがあります。

エミュレータを使用する人は：

本エミュレータは、ユーザーズマニュアルをよく読み、理解した人のみをご使用ください。

ユーザーズマニュアルをよく読まずに、本エミュレータを使用しないでください。

特に、本エミュレータを初めて使用する人は、本エミュレータをよく理解し、使い慣れている人から指導を受けることを強く勧めます。

保証の範囲：

ルネサスは、お客様が製品をご購入された日から1年間は、無償で故障品を修理、または交換いたします。ただし、

1. 製品の誤用、濫用、またはその他異常な条件下での使用
2. ルネサス以外の者による改造、修理、保守、またはその他の行為
3. ユーザシステムの内容、または使用
4. 火災、地震、またはその他の事故

により、故障が生じた場合はご購入日から1年以内でも有償で修理、または交換を行います。また、日本国内で購入され、かつ、日本国内で使用されるものに限ります。

その他の重要事項：

1. 本資料に記載された情報、製品または回路の使用に起因する損害または特許権その他権利の侵害に関しては、ルネサスは一切その責任を負いません。
2. 本資料によって第三者またはルネサスの特許権その他権利の実施権を許諾するものではありません。

著作権所有：

このユーザーズマニュアルおよび本エミュレータは著作権で保護されており、すべての権利はルネサスに帰属しています。このユーザーズマニュアルの一部であろうと全部であろうといかなる箇所も、ルネサスの書面による事前の承諾なしに、複写、複製、転載することはできません。

図について：

このユーザーズマニュアルの一部の図は、実物と違っていることがあります。

予測できる危険の限界：

ルネサスは、潜在的な危険が存在するおそれのあるすべての起こりうる諸状況や誤使用を予見できません。したがって、このユーザーズマニュアルと本エミュレータに貼付されている警告がすべてではありません。お客様の責任で、本エミュレータを正しく安全に使用してください。

安全事項

- 本エミュレータをご使用になる前に、必ずユーザーズマニュアルをよく読んで理解してください。
- ユーザーズマニュアルは、必ず保管し、使用上不明な点がある場合は再読してください。

シグナル・ワードの定義



これは、安全警告記号です。潜在的に、人に危害を与える危険に対し注意を喚起するために用います。起こり得る危害または死を回避するためにこの記号の後に続くすべての安全メッセージに従ってください。



危険は、回避しないと、死亡または重傷を招く差し迫った危険な状況を示します。ただし、本製品では該当するものではありません。



警告は、回避しないと、死亡または重傷を招く可能性がある潜在的に危険な状況を示します。



注意は、回避しないと、軽傷または中程度の傷害を招く可能性がある潜在的に危険な状況を示します。



安全警告記号の付かない注意は、回避しないと財物傷害を引き起こすことがある潜在的に危険な状況を示します。

注、留意事項は、例外的な条件や注意を操作手順や説明記述の中で、ユーザに伝達する場合に使用しています。



1. DC電源を内蔵していますので、触れる場所によっては感電する可能性があります。感電、火災等の危険防止および品質保証のために、お客様ご自身による修理や改造は行なわないでください。故障の際のアフターサービスにつきましては、ルネサスまたはルネサス特約店保守担当にお申し付けください。
2. エミュレータまたはユーザシステムのパワーオン時、全てのケーブル類の抜き差しを行なわないでください。抜き差しを行った場合、エミュレータとユーザシステムの発煙発火の可能性があります。また、デバッグ中のユーザプログラムの破壊の可能性があります。
3. エミュレータまたはユーザシステムのパワーオン時、エバチップボードまたはユーザシステムインタフェースボードとユーザシステムの抜き差しを行なわないでください。抜き差しを行った場合、エミュレータとユーザシステムの発煙発火の可能性があります。また、デバッグ中のユーザプログラムの破壊の可能性があります。
4. エバチップボードまたはユーザシステムインタフェースボードとユーザシステム上のICソケットは、ピン番号を確かめて正しく接続してください。接続を誤るとエミュレータとユーザシステムの発煙発火の可能性があります。
5. 電源給電については電源仕様に従って供給してください。使用する電源ケーブルは製品に添付のものを使用してください。仕様以外の電源電圧を加えないでください。

エミュレータ使用時の注意事項

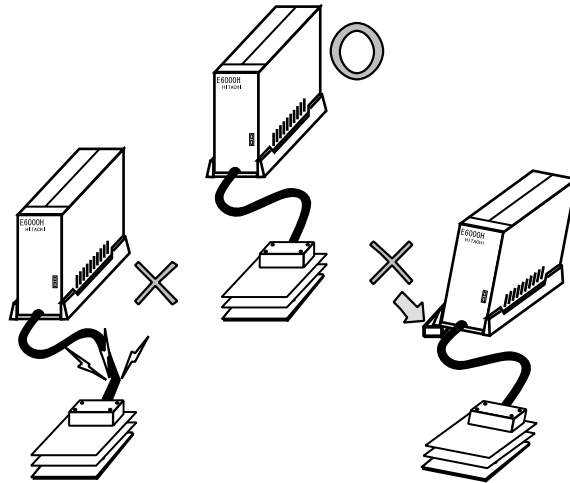
このエミュレータ使用時の注意事項に記載されている事項は、本エミュレータを使用するうえですべての場合に該当し、例外は存在しません。したがって、エミュレータを使用する前に以下に示されている警告文をよく読み、完全に理解してください。ただし、ここに記載されている事項はエミュレータ使用時における共通の警告のみが記載されており、これがエミュレータを使用するうえでのすべての警告ではありません。

警告

エミュレータまたはユーザシステムのパワーオン時には、すべてのケーブル、およびユーザインターフェースの抜き差しを行わないでください。抜き差しを行なった場合、エミュレータとユーザシステムの発煙発火、および機器の破壊の可能性があります。また、デバッグ中のユーザプログラムの破壊の可能性があります。

注意

エミュレータ本体とエバチップボードの位置関係により、トレースケーブルが大きく曲げられることがあります。このような状態で使用すると、ユーザインターフェース部に大きなストレスが加わり、接点、接触不良等の機械的破壊を招く原因となります。また、使用中にエミュレータ本体が動いてしまうと、ユーザインターフェース部に思わぬストレスを与えることになります。エミュレータ本体の設置位置に十分ご注意ください。



はじめに

このたびは、E6000H エミュレータをお買い上げ頂き、誠にありがとうございます。

注意

本エミュレータをご使用になる前に、必ず本マニュアルをよく読み、理解してください。誤った使用方法、接続方法は、本エミュレータ、ユーザプログラム、ユーザシステムの破壊につながります。

E6000H エミュレータは、ルネサスオリジナルマイクロコンピュータを使用したシステムの開発をソフトウェア、ハードウェアの両面からサポートする支援装置です。E6000H エミュレータは、ソフトウェア開発とデバッグのために単体で、あるいはユーザシステムのデバッグのためにユーザシステムインタフェースケーブルでユーザシステムに接続した状態で使用できます。

E6000H エミュレータの特徴は以下の通りです。

1. リアルタイムエミュレーション

リアルタイムにターゲットマイコンをエミュレーションすることができます。

2. 充実したデバッグ機能

ブレーク機能の充実、トレースの大容量化（128kサイクル）によりデバッグ効率を向上させます。

3. パラレルアクセス機能

エミュレーション実行中にトレース内容を表示したり、エミュレーションメモリの内容を参照、変更することができます。

4. パフォーマンス測定機能

サブルーチンの実行時間、実行回数測定などユーザプログラムの実行効率を測定することができます。

5. グラフィカルユーザインタフェースを提供

Windows®上で動作するHigh-performance Embedded Workshopを使用したグラフィカルユーザインタフェースを提供します。

High-performance Embedded 用に、C/C++言語およびアセンブリ言語で書いたアプリケーションの開発およびデバッグを簡単に行うためのグラフィカルユーザインタフェースを提供します。アプリケーションを実行するエミュレータのアクセス、計測、および変更に関して、High-performance Embedded Workshop は高機能でしかも直観的な手段を提供することを目的としています。

High-performance Embedded Workshop は、マイクロコンピュータの組み込み用アプリケーションの開発を強力に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。

- 使い勝手の良いインタフェースを活用したコンパイラ、アセンブラ、リンカージェディタなどのオプションが設定できるカスタマイズ可能な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化テキストエディタ。
- 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グ。
- 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。一つはユーザに強力な開発ツールを提供すること、そしてもう一つは、それらのツール類を統合して使いやすくすることです。

このマニュアルについて

本マニュアルではハードウェア編、およびデバッグ編の2部構成になっています。それぞれ以下の内容を説明しています。

ハードウェア編 使用前の準備、ハードウェア仕様、故障診断解析

デバッグ編 エミュレータ固有の機能、エミュレータ固有のデバッグ機能、チュートリアル、エミュレータソフトウェア仕様
および注意事項

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ク機能については、High-performance Embedded Workshop ユーザーズマニュアルを参照してください。

このマニュアルでは 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、各々のマニュアルを参照してください。

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 は International Business Machines 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。

このマニュアルの記号

このマニュアルで使われている記号の意味を説明します。

表 1 記号一覧

記号	意味
[Menu->Menu Option]	太字と'>'はメニューオプションを示します (例 [File->Save As...])
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
☞(「操作方法」マーク)	このマークが左端にあるとき、その右の文章は何かの操作方法を示します
バックスラッシュ文字'\'	文中および図中で使用しているファイルパス名の文字列中の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。

梱包品の確認

梱包を解いた後、納入品明細書に記入されている梱包品がそろっているか確認してください。確認した結果、梱包品に不足がありましたら、当エミュレータ購入元の営業担当までご連絡ください。

目次

ハードウェア編	1
1. 概要	3
1.1 使用前の注意	3
1.2 エミュレータハードウェア構成	4
1.2.1 E6000H 本体の構成（本マニュアルの写真は一部実物と異なります）	5
1.2.2 エバチップボードの構成	7
1.2.3 ユーザシステムインタフェースボードの構成	8
1.3 システム構成	9
1.3.1 PC インタフェースボードによるシステム構成	9
2. 使用前の準備	11
2.1 E6000Hエミュレータ使用までのフローチャート	11
2.2 エミュレータの接続	11
2.2.1 ユーザシステムとの接続	11
2.2.2 ユーザシステムインタフェースボードの接続	12
2.2.3 外部プローブ接続用コネクタ	13
2.2.4 クロックの選択	14
2.2.5 システムグランド系の接続	15
2.2.6 PC インタフェースボードについて	16
3. ハードウェア仕様	17
3.1 使用環境条件	17
3.2 エミュレータの外形寸法と質量	18
3.3 ユーザシステムインタフェース回路	19
3.3.1 ユーザシステムインタフェース回路	19
3.3.2 ユーザシステムインタフェースのディレイ時間	23
3.4 エミュレータとユーザシステムの接続	23
3.4.1 ユーザシステムとの接続方法	23
3.4.2 ユーザシステムインタフェースコネクタのピン配置	34
3.4.3 ユーザシステム接続時の注意事項	39
3.5 ターゲットマイコンのサポート	40
3.5.1 メモリ空間	40
3.5.2 低消費電力状態（スリープ、ソフトウェアスタンバイ、ハードウェアスタンバイ）	41
3.5.3 割り込み	41
3.5.4 コントロール用入力信号（_RES、_BREQ、_WAIT）	41
3.5.5 BSC（バスステートコントローラ）	41
3.5.6 WDT（ウォッチドッグタイマ）	41
3.5.7 A/D 変換器	42
3.5.8 E6000H エミュレータの状態と内蔵モジュール	42
3.5.9 端子機能	43
3.5.10 エミュレータのレジスタ初期値相違点	43
4. 故障解析（故障症状調査書）	45
4.1 故障解析	45
4.1.1 診断プログラムを実行するためのシステムセットアップ	45
4.1.2 診断プログラムのテスト項目	46

4.1.3	診断プログラムによる故障解析	47
4.2	故障症状調査書	51
デバッガ編		1
1.	概要	3
2.	デバッグの準備をする	5
2.1	High-performance Embedded Workshopの起動方法	5
2.1.1	新規にワークスペースを作成する場合（ツールチェイン未使用）	6
2.1.2	新規にワークスペースを作成する場合（ツールチェイン使用）	8
2.1.3	既存のワークスペースを指定する場合	12
2.2	エミュレータの接続	13
2.3	エミュレータの再接続	14
2.4	エミュレータの終了	14
3.	デバッグ	15
3.1	エミュレーション環境を設定する	15
3.1.1	コンフィグレーションプロパティダイアログボックスを開く	15
3.1.2	ピンファンクションコントローラに関連する設定を行う	17
3.1.3	接続するインタフェースを選択する	18
3.1.4	Memory Mapping ダイアログボックスを開く	19
3.1.5	メモリマップ設定を変更する	20
3.2	プログラムをダウンロードする	21
3.2.1	プログラムをダウンロードする	21
3.2.2	ソースコードを表示する	21
3.2.3	アセンブリ言語コードを表示する	23
3.2.4	アセンブリ言語コードを修正する	24
3.2.5	特定のアドレスを見る	24
3.2.6	現在のプログラムカウンタアドレスを見る	24
3.3	現在の状態を表示する	25
3.4	エミュレータの情報を定期的に読み出し表示する	26
3.4.1	[拡張モニタ]ウィンドウを開く	26
3.4.2	表示項目を選択する	27
3.5	リアルタイムにメモリ内容を表示する	27
3.5.1	モニタウィンドウを開く	28
3.5.2	モニタの設定内容を変更する	30
3.5.3	モニタの更新を一時的に停止する	30
3.5.4	モニタ設定を削除する	30
3.5.5	変数の内容をモニタする	30
3.5.6	モニタウィンドウを非表示にする	30
3.5.7	モニタウィンドウを管理する	31
3.6	変数の表示	32
3.6.1	ウォッチウィンドウ	32
3.7	イベントポイントを使用する	33
3.7.1	ソフトウェアブレークポイントを設定する	34
3.7.2	オンチップブレークポイントを設定する	36
3.7.3	オンエミュレータブレークポイントを設定する	38
3.7.4	イベントポイントの編集	40
3.7.5	イベントポイントの設定内容を変更する	40
3.7.6	イベントポイントを有効にする	40

3.7.7	イベントポイントを無効にする.....	40
3.7.8	イベントポイントを削除する.....	40
3.7.9	イベントポイントをすべて削除する.....	40
3.7.10	イベントポイントのソース行を表示する.....	40
3.8	トレース情報を見る.....	41
3.8.1	トレースウィンドウを開く.....	41
3.8.2	トレース情報を取得する.....	41
3.8.3	トレース情報取得条件を設定する.....	43
3.8.4	Trace レコードを検索する.....	48
3.8.5	トレース情報をクリアする.....	49
3.8.6	トレース情報をファイルに保存する.....	49
3.8.7	[エディタ]ウィンドウを表示する.....	49
3.8.8	ソース表示を整形する.....	49
3.8.9	トレース情報の取得を一時的に停止する.....	49
3.8.10	トレース情報の取得を再開する.....	50
3.8.11	取得したトレース情報から必要なレコードを抽出する.....	50
3.8.12	タイムスタンプの差を計算する.....	52
3.8.13	統計情報を解析する.....	52
3.8.14	取得したトレース情報から関数呼び出し箇所を抽出する.....	54
3.9	パフォーマンスを測定する.....	55
3.9.1	パフォーマンス解析ウィンドウを開く.....	56
3.9.2	実行効率測定条件を設定する.....	57
3.9.3	実行効率測定を開始する.....	62
3.9.4	測定条件を削除する.....	62
3.9.5	すべての測定条件を削除する.....	62
3.10	プロファイル機能を使用する.....	63
3.10.1	プロファイルを有効にする.....	63
3.10.2	測定方法を指定する.....	63
3.10.3	ユーザプログラムを実行し結果を確認する.....	63
3.10.4	List シート.....	63
3.10.5	Tree シート.....	64
3.10.6	ソースファイル表示.....	64
3.10.7	チャート表示.....	64
3.10.8	有効.....	64
3.10.9	関数呼び出しをトレースしない.....	65
3.10.10	検索.....	65
3.10.11	データ検索.....	65
3.10.12	データクリア.....	65
3.10.13	プロファイル情報の保存.....	65
3.10.14	テキスト形式で保存.....	65
3.10.15	表示設定.....	66
3.10.16	プロパティ.....	66
3.11	プロファイル-チャートウィンドウ.....	66
3.11.1	ソースファイル表示.....	67
3.11.2	チャート表示.....	67
3.11.3	有効.....	67
3.11.4	データクリア.....	67
3.11.5	チャートウィンドウを複数開く.....	67
3.11.6	プロファイル情報の保存.....	67
3.11.7	拡大.....	67
3.11.8	縮小.....	67

4.	チュートリアル.....	69
4.1	はじめに	69
4.2	High-performance Embedded Workshopの起動	70
4.3	チュートリアルプログラムのダウンロード	71
4.3.1	チュートリアルプログラムをダウンロードする	71
4.3.2	ソースプログラムを表示する	72
4.4	ソフトウェアブレークポイントの設定	73
4.5	レジスタ内容の変更	74
4.6	プログラムの実行	75
4.7	ブレークポイントの確認	76
4.8	シンボルの参照	77
4.9	メモリ内容の確認	78
4.10	変数の参照	79
4.11	ローカル変数の表示	82
4.12	プログラムのステップ実行	82
4.12.1	ステップインコマンドの実行	83
4.12.2	ステップアウトコマンドの実行	84
4.12.3	ステップオーバーコマンドの実行	85
4.13	プログラムの強制ブレーク	86
4.14	ターゲットマイコンのリセット	86
4.15	ブレーク機能	87
4.15.1	ソフトウェアブレーク機能	87
4.15.2	オンチップブレーク機能	92
4.16	トレース機能	94
4.16.1	フリートレースによるトレース情報の表示	95
4.16.2	トレース停止によるトレース情報の表示	96
4.16.3	条件トレースによるトレース情報の表示	99
4.16.4	統計	100
4.16.5	関数コール	103
4.17	スタックトレース機能	103
4.18	パフォーマンス測定機能	106
4.18.1	指定範囲内時間測定	106
4.19	モニタ機能	109
4.20	さてつぎは？	111
5.	本製品固有のソフトウェア仕様と注意事項.....	113
5.1	対応ハードウェアについて	113
5.2	デバッグプラットフォームについて	113
5.3	メモリ内容の表示と変更方法について	113
5.3.1	実行中のメモリ内容の表示と変更	113
5.3.2	パラレルアクセス機能停止時間参考値	114
5.3.3	モニタ機能	114
5.4	プログラムの実行について	114
5.4.1	ステップ実行	114
5.5	イベント機能について	115
5.5.1	ソフトウェアブレーク	115
5.5.2	オンチップブレーク	115
5.5.3	オンエミュレータブレーク	115
5.6	トレース機能について	116
5.6.1	トレース表示内容	116
5.6.2	トレース取得条件	116
5.6.3	トレース情報の検索	116
5.6.4	トレース情報のフィルタ	116
5.7	モニタ機能について	116
5.8	パフォーマンス測定機能について	117

5.8.1	測定誤差について	117
5.8.2	留意事項	117
5.9	プロファイル機能について.....	118
5.10	入力形式について	118
5.10.1	マスク入力	118
5.11	チュートリアルプログラムについて.....	119
5.11.1	チュートリアルプログラムの操作についての注意事項.....	119
5.12	メモリマップ機能について.....	119
5.12.1	エミュレーションメモリ	119
5.12.2	メモリマップ制御	119
6.	エラーメッセージ	123
6.1	E6000Hのエラーメッセージ	123
6.1.1	起動時のエラーメッセージ.....	123
6.1.2	実行中のエラーメッセージ.....	124
付録 A	メニュー一覧.....	125
付録 B	コマンドライン一覧	129

ハードウェア編

1. 概要

1.1 使用前の注意

<h3>注意</h3>

<p>本エミュレータをご使用になる前に、以下の注意事項を必ずよく読んで理解してください。誤った使い方は、本エミュレータ、ユーザプログラムおよびユーザシステムの破壊につながります。</p>

- (1) 製品を梱包箱から取り出し、納入品明細書に示されているものが揃っているか確認してください。
- (2) 製品に重量物を上積みするなどして、無理な力を加えないでください。
- (3) 次の条件を考慮して、E6000H本体を設置してください。
 - E6000H本体の背面に内部冷却用ファンがあります。E6000H本体と壁面や他の装置間は20cm以上離してください。
 - 直射日光に当たる場所や暖房機の近く等、高温となる場所に設置しないでください。詳細については、「3.1 使用環境条件」を参照してください。
 - 温度や湿度が極端に変化する場所に設置しないでください。
 - ホコリやチリが多い場所に設置しないでください
 - 振動が多い場所に設置しないでください。「3.1 使用環境条件」を参照してください。
- (4) 製品に過大な物理的衝撃や力を与えないでください。
- (5) E6000H本体に、指定された電圧、電源周波数以外の電源を供給しないでください。
- (6) 設置場所を移動する場合は、本製品に強い振動、衝撃が加わらないように注意してください。
- (7) ケーブルを接続した後は、接続位置が正しいことを再度確認してください。
接続方法については、「2 使用前の準備」を参照してください。
- (8) 全てのケーブルを接続し終えてから、E6000Hエミュレータおよび接続した各装置へ電源を投入してください。
また、電源が入っているときにケーブルの接続および取り外しをしないでください。
- (9) ターゲットマイコンとエミュレータとの相違点など、エミュレーション時の注意事項は「3.5 ターゲットマイコンのサポート」を参照してください。

1.2 エミュレータハードウェア構成

E6000H エミュレータは、E6000H 本体、エバチップボードにより構成されています。ユーザシステムインタフェースボード(オプション)を実装することにより、デバイスと同一パッケージで接続することができます。PC インタフェース(オプション)として、PC インタフェースボード(PCIバス、PC カードバス)、LAN アダプタ(ネットワーク接続)および USB アダプタ(USB インタフェース接続)があります。

これらのインタフェースを用いて、E6000H とホストコンピュータを接続することで、High-performance Embedded Workshop を用いたデバッグが可能になります。

PCIバス、PC カードバス仕様の PC インタフェースボード、LAN アダプタおよび USB アダプタについては、各取扱い説明書をお読みください。

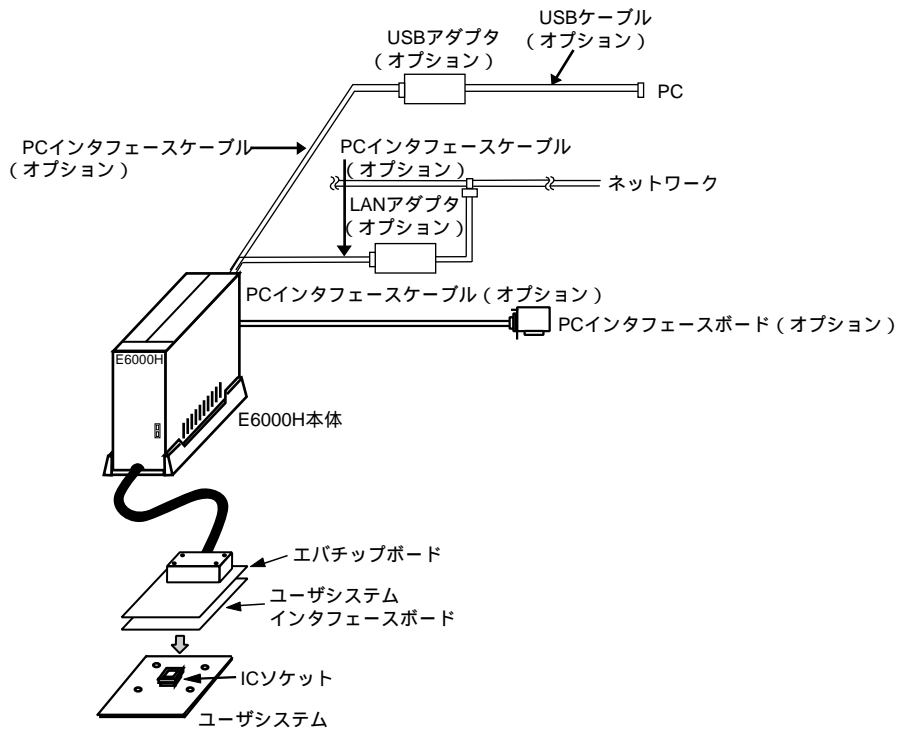


図1.1 エミュレータのハードウェア構成

1.2.1 E6000H 本体の構成（本マニュアルの写真は一部実物と異なります）

E6000H 本体における各部の名称を下記に示します。

（1）前面パネルの構成



図1.2 E6000H 本体の前面パネル

各部名称

- | | | |
|--------------|---|--------------------------|
| （a） POWERランプ | ： | E6000H本体が電源ON状態のとき点灯します。 |
| （b） RUNランプ | ： | ユーザプログラムが実行中のとき点灯します。 |

1. 概要

(2) 背面パネルの構成

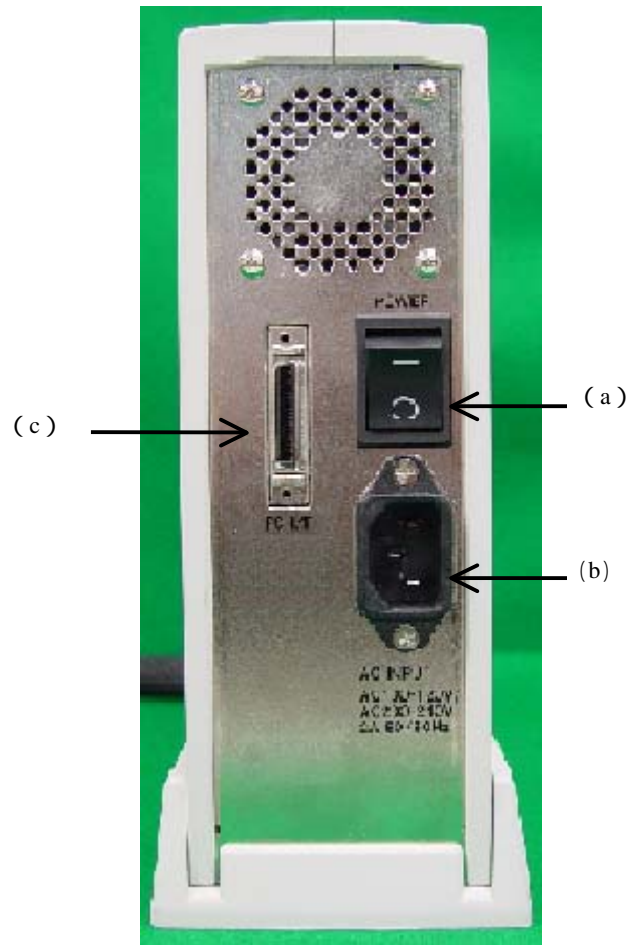


図1.3 E6000H 本体の背面パネル

各部名称

- | | |
|--------------------------|--|
| (a) POWERスイッチ | : 本スイッチをI (input) 側により、エミュレータ (E6000H本体とエパチップボード) に電源が供給されます。 |
| (b) AC電源用コネクタ | : AC100V入力用コネクタです。 |
| (c) PCインタフェースケーブル接続用コネクタ | : ホストコンピュータとE6000H本体を接続するインタフェースケーブルのコネクタです。ホスト側PC I/Fボードとインタフェースを取ります。ここにPCインタフェースボード、PCカードインタフェース、LANアダプタ、USBアダプタが接続されます。"PC I/F"と表示しています。 |

1.2.2 エバチップボードの構成

エバチップボードにおける各部の名称を下記に示します。

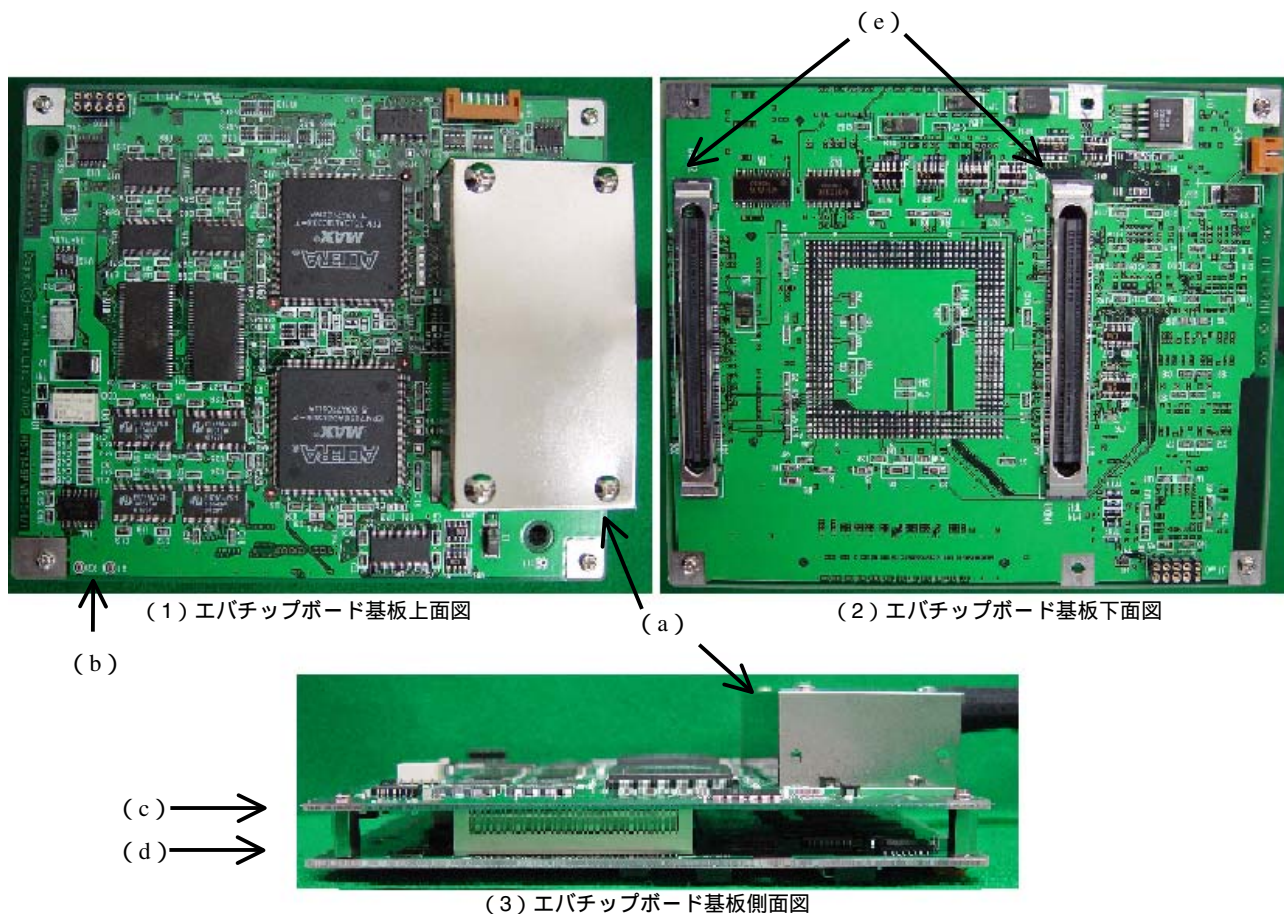


図1.4 エバチップボードの構成

各部名称

- | | | |
|-----------------------------------|---|--|
| (a) 本体-エバチップボードインタフェース
コネクタカバー | : | E6000H本体とエバチップボードを接続するコネクタ部を保護するカバーです。 |
| (b) 水晶発振子取り付け端子 | : | ターゲットマイコンに外部クロックを供給するための水晶発振子を取り付ける端子です。 |
| (c) HS7145PWB20H基板 | : | トレースケーブルと接続するコネクタを持ちます。 |
| (d) HS7145PWB30H基板 | : | エバチップを実装し、ユーザシステムインタフェースボードまたはユーザシステムと接続するコネクタ（専用コネクタ）を持ちます。 |
| (e) ユーザシステムインタフェースボード接続用
コネクタ | : | ユーザシステムインタフェースボードまたはユーザシステムを接続するためのコネクタです。 |

【注】 上記(a)～(e)を総称して、エバチップボードと呼びます。

1. 概要

1.2.3 ユーザシステムインタフェースボードの構成

ユーザシステムインタフェースボードにおける各部の名称を下記に示します。

(1) HS7145ECH61H基板上面図

(2) HS7145ECH61H基板下面図

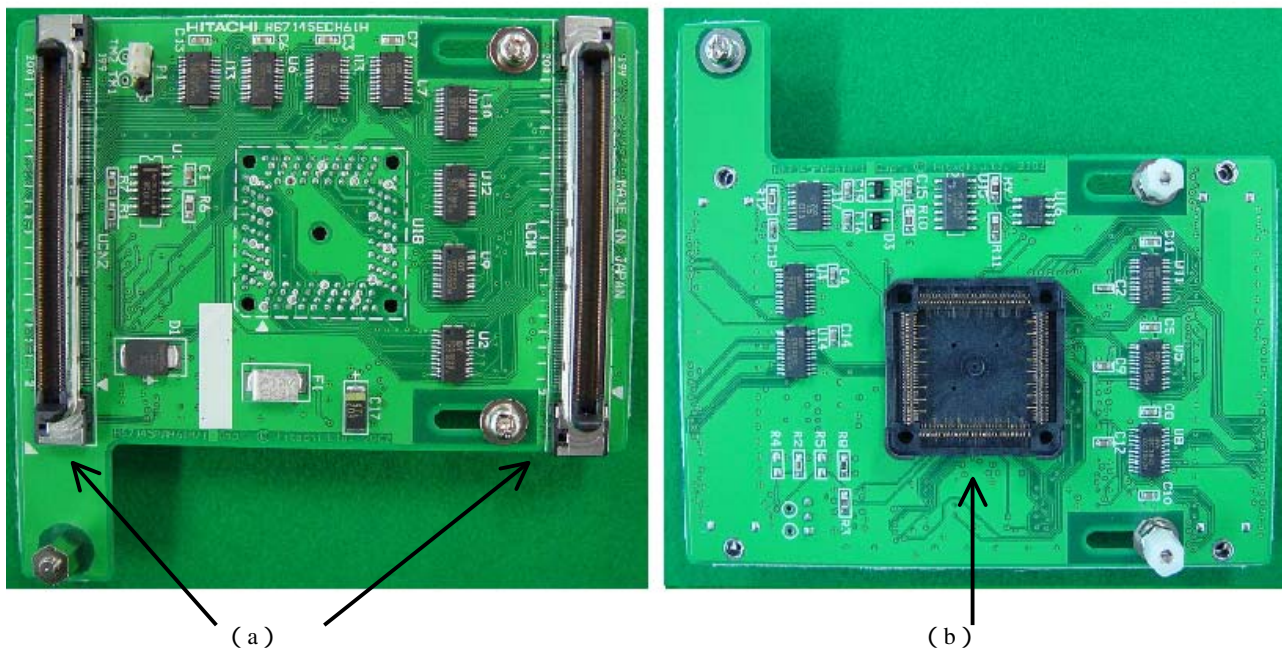


図1.5 エバチップボードの構成

各部名称

- | | | |
|---------------------|---|-------------------------|
| (a) エバチップボード接続用コネクタ | : | エバチップボードを接続するためのコネクタです。 |
| (b) ユーザシステム接続用コネクタ | : | ユーザシステムを接続するためのコネクタです。 |

1.3 システム構成

E6000H 本体は、ホストコンピュータ（オプションの PC インタフェースボード使用）と接続して使用してください。

1.3.1 PC インタフェースボードによるシステム構成

E6000H 本体は、PC インタフェースボード（オプション：PCI バス、PC カードバス）を介してホストコンピュータと接続することができます。PC インタフェースボードをホストコンピュータの各インタフェースボードに対応する拡張スロットなどに実装し、PC インタフェースボード付属のインタフェースケーブルで E6000H 本体と接続します。また、LAN アダプタは E6000H とネットワークを介してホストコンピュータと接続します。USB アダプタは E6000H とホストコンピュータを USB インタフェースで接続します。PCI バス、PC カードバス仕様の PC インタフェースボード、LAN アダプタおよび USB アダプタについては、各取扱い説明書をお読みください。PC インタフェースボードによるシステム構成を図 1.6 に、LAN アダプタによるシステム構成を図 1.7、USB アダプタによるシステム構成を図 1.8 に示します。

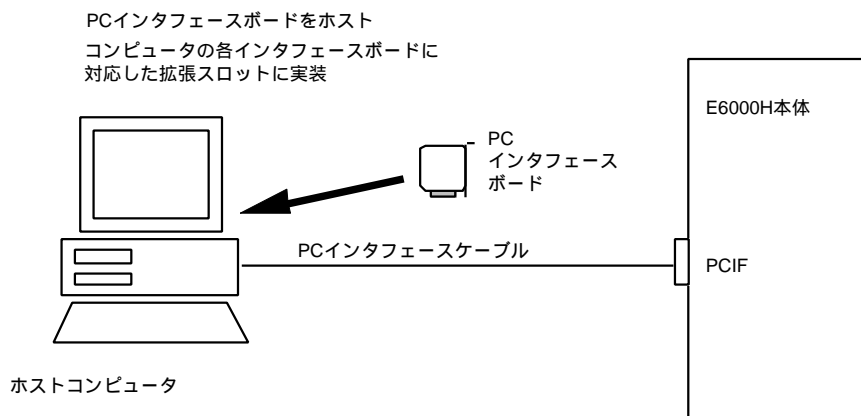


図1.6 PC インタフェースボードによるシステム構成

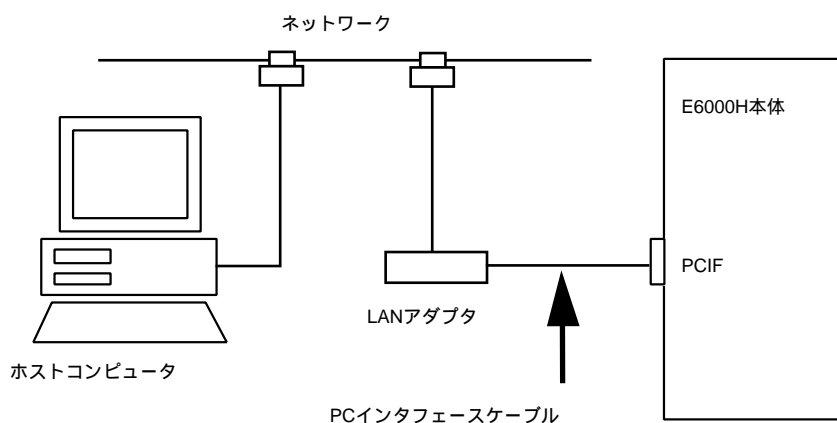


図1.7 LAN アダプタによるシステム構成

1. 概要

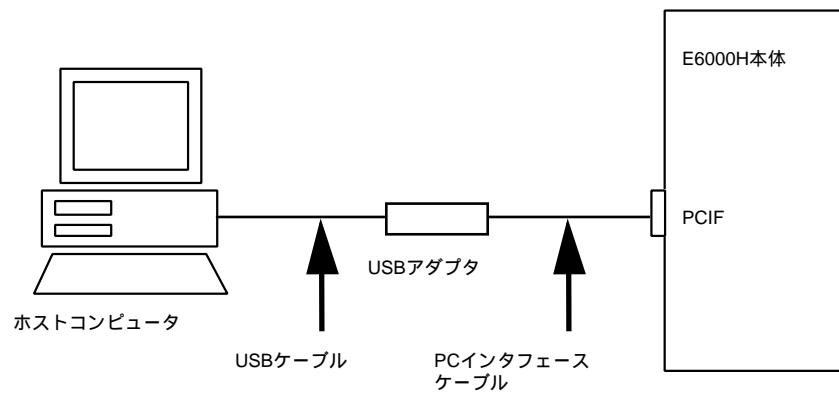


図1.8 USB アダプタによるシステム構成

2. 使用前の準備

2.1 E6000Hエミュレータ使用までのフローチャート

本章では、E6000H エミュレータ使用前の準備について説明します。E6000H エミュレータを使用しデバッグを開始するまでのフローチャートを図 2.1 に示します。

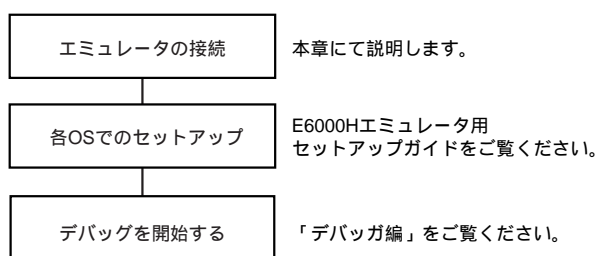


図 2.1 E6000H エミュレータ使用までのフローチャート

注意

準備にあたり、本章の内容を必ずよく読んで理解してください。誤った使い方は、本エミュレータ、ユーザプログラムおよびユーザシステムの破壊につながります。

2.2 エミュレータの接続

2.2.1 ユーザシステムとの接続

警告

エミュレータおよびユーザシステムの電源が入っているとき、全ての抜き差しを行わないでください。抜き差しを行った場合、エミュレータとユーザシステムを破壊する可能性があります。また、デバッグ中のユーザプログラムを破壊する可能性があります。

- (1) E6000H本体の電源がOFFであることを確認してください。E6000H本体前面右側にあるPOWERランプが消灯状態であれば電源はOFF状態です。
- (2) E6000H本体のAC電源ケーブルがAC電源用コンセントに接続されている場合は、AC電源用コンセントから外します。
- (3) E6000Hユーザシステムインタフェースボードの下面に実装しているコネクタにユーザシステム側のコネクタ1pinを合わせて接続します。コネクタ接続の際には、片側が浮いた状態にならないように必ずきちんと奥まで接続してください。

2.2.2 ユーザシステムインタフェースボードの接続

接続方法は、各 SH7046 E6000H シリーズ用ユーザシステムインタフェースボード取り扱い説明書を参照してください。

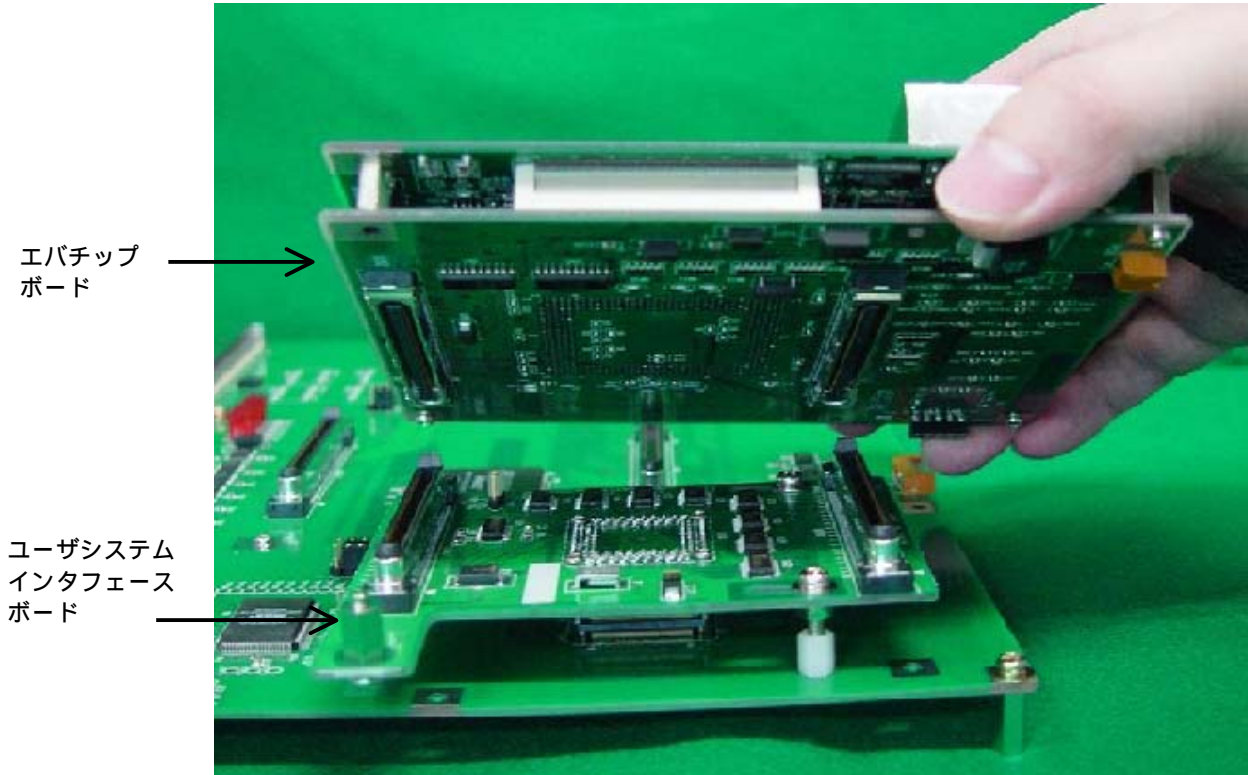


図2.2 ユーザシステムインタフェースボードの接続

警告

ユーザシステムインタフェースボードの接続、取り外しを行う場合は、必ずエミュレータおよびユーザシステムの電源すべてをオフにし、コネクタおよびICソケットのピン番号を確かめて作業してください。
電源がオンの状態で作業を行ったり、接続を誤ると、エミュレータ、ユーザシステムインタフェースボードまたは、ユーザシステムの破壊、発煙発火の可能性があります。

2.2.3 外部プローブ接続用コネクタ

注意

外部プローブには方向性があります。よく向きを確認してから挿入してください。誤った方向で挿入すると、コネクタ部を破壊する原因となります。

E6000H エバチップボードの外部プローブ接続用コネクタの端子を通して、外部信号のトレース、マルチエミュレータのブレーク検出をすることができます。

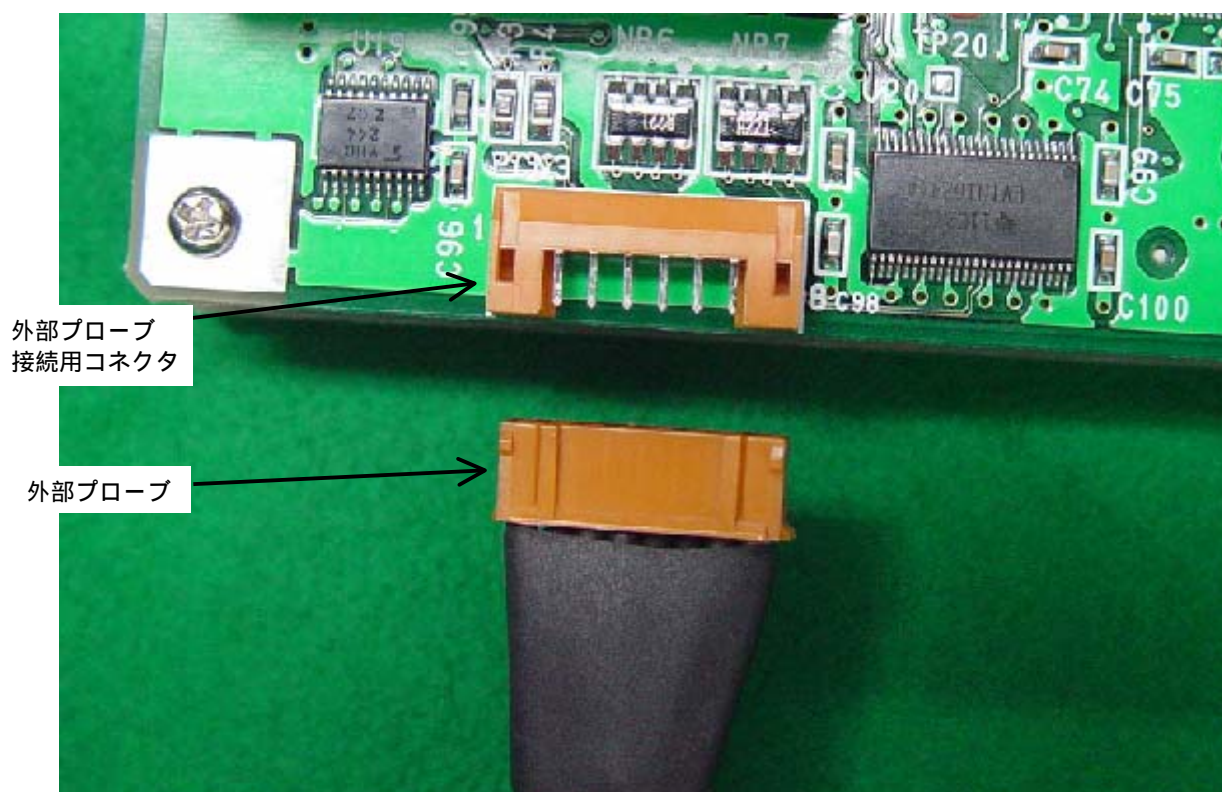


図2.3 外部プローブ接続用コネクタ

ピン番号	プローブ名	信号名	備考
1	1	プローブ入力 1	同期ブレーク用の入力端子
2	2	プローブ入力 2	
3	3	プローブ入力 3	
4	4	プローブ入力 4	
5	G	GND	GND の接続端子
6	TUV	トリガ出力	トリガモード時の出力端子
7	G	GND	GND の接続端子
8	T5V	RUN / ブレーク状態出力	RUNSTEP 時"High"レベル出力

2. 使用前準備

2.2.4 クロックの選択

エミュレータでは、ターゲットマイコンの入力クロックとして、エバチップボードに実装する水晶発振子、ユーザシステムからの外部クロック入力、およびエミュレータ内部の貸出しクロックの3種類をサポートしています。それぞれのクロックの設定は、[Configuration]ダイアログボックスを使用して行います。また、本エミュレータではターゲットマイコンのシステムクロック(ϕ)として最高動作周波数 50.0MHz (外部クロック 12.5MHz を 4 通倍した値)まで使用することができます。

選択できるクロックの選択は下記になります。

Target	外部クロック (ユーザシステムから EXTAL 端子にクロックを供給) 4.0 - 12.5MHz
Xtal	水晶発振子 5.0 - 12.5MHz
12.5MHz	エミュレータの貸出しクロック
10MHz	エミュレータの貸出しクロック
8MHz	エミュレータの貸出しクロック
6MHz	エミュレータの貸出しクロック
4MHz	エミュレータの貸出しクロック

(1) 水晶発振子を使用する場合

本製品には水晶発振子は添付していませんので、お客様が使用する周波数の水晶発振子を用意してください。

ただし、ターゲットマイコンの入力クロックとして水晶発振子を使用する場合の周波数範囲は 5.0 - 12.5MHz です。

次に水晶発振子を実装する手順を示します。

注意

エミュレータの電源が入っているとき、水晶発振子の抜き差しを行わないでください。抜き差しを行った場合、デバッグ中のユーザプログラムを破壊する可能性があります。

1. エミュレータの電源がOFFであることを確認してください (POWER LEDが消灯していることを確認してください)。
2. エバチップボードにある水晶発振子取り付け端子に水晶発振子を実装してください。(図2.4参照)
3. ユーザ電源をONし、エミュレータの電源をONすると、自動的に水晶発振子に設定されて立ち上がります。この機能により、ユーザシステムが接続されていない場合でも、ユーザシステムの動作周波数でユーザプログラムを実行することができます。

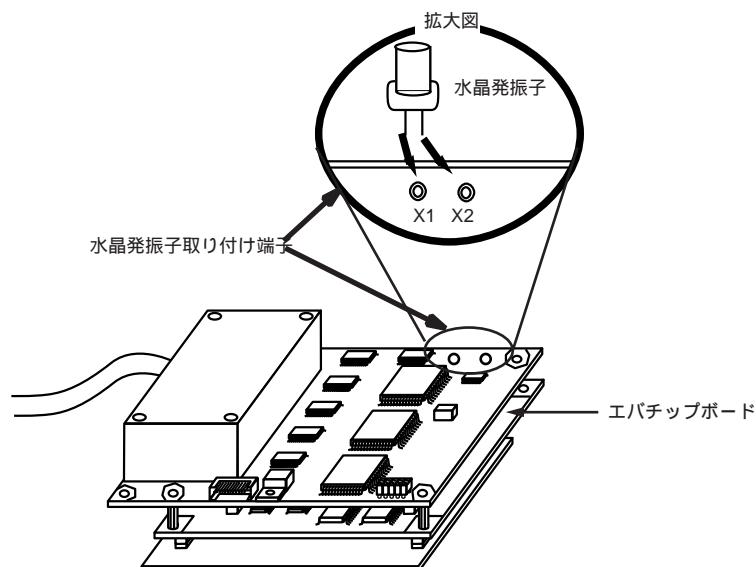


図2.4 水晶発振子の実装方法

(2) 外部クロックを使用する場合

1. エミュレータの電源がOFFであることを確認してください (POWER LEDが消灯していることを確認してください)。
2. ユーザシステムにエバチップボードを接続し、ユーザシステムからEXTAL端子にクロックを供給してください。
3. ユーザシステムの電源をONし、エミュレータの電源をONすると外部クロックに設定されて立ち上がります。

(3) エミュレータの貸出しクロックを使用する場合

[コンフィグレーションプロパティ]ダイアログボックスにてエミュレータの貸出しクロックを指定してください。

[参考] エミュレータのシステムプログラム起動時、次の順序で入力するクロックを自動的に選択します。

1. ユーザシステムから外部クロックが入力されている場合、ユーザシステムのクロックを選択
2. エバチップボードに水晶発振子が実装してある場合、水晶発振子を選択
3. エミュレータの貸出しクロックを選択

2.2.5 システムグランド系の接続

注意

システムグランドは必ずユーザシステム上で、フレームグランドとシグナルグランドを切り離してください。フレームグランドとシグナルグランドを接続した状態でエミュレータを接続すると、誤動作の恐れがあります。

エミュレータのシグナルグランドは、エバチップボードを介して、ユーザシステムのシグナルグランドに接続されます。

エミュレータ本体内部では、フレームグランドとシグナルグランドが接続されています。(図 2.5) ユーザシステムでは、フレームグランドとシグナルグランドを接続せず、フレームグランドのみを接続してください。

ユーザシステム内で、フレームグランドとシグナルグランドを切り離すのが難しい場合、図 2.6のようにエミュレータ本体の 100V 用電源のフレームグランドと同一のコンセントでグランドをとり、グランド電位が等しくなるようにしてください。

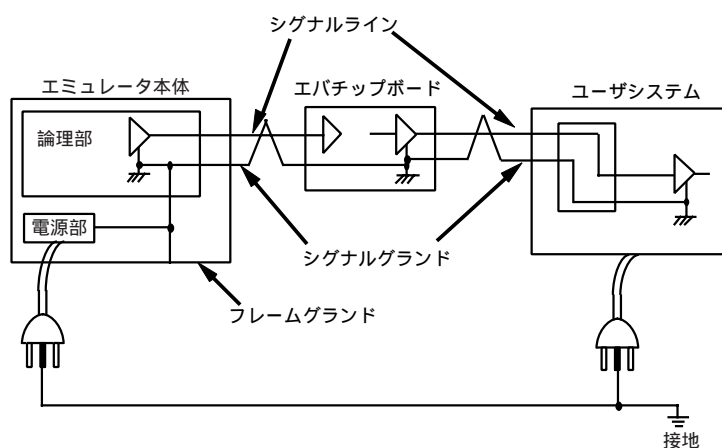


図 2.5 システムグランド系の接続

警告

エミュレータおよびユーザシステムの電源が入っているとき、すべてのケーブル類の抜き差しを行わないでください。抜き差しを行った場合、エミュレータとユーザシステムを破壊する可能性があります。また、デバッグ中のユーザプログラムを破壊する可能性があります。

ユーザシステムのシステムグラウンド系は、最適なグラウンド接続にしてノイズやグラウンドループなどの影響を最小にすることが必要です。したがって、エバチップボードとユーザシステムの接続において、エバチップボードの端子とユーザシステムのグラウンドが確実に接続されているか確認してください。

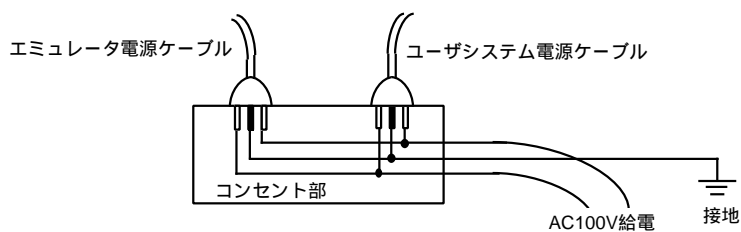


図2.6 フレームグラウンド接続例

2.2.6 PC インタフェースボードについて

PC インタフェースボード、LAN アダプタおよび USB アダプタについては、各製品付属の取り扱い説明書を参照してください。

3. ハードウェア仕様

3.1 使用環境条件

注意

E6000H エミュレータを使用する場合、表 3.1に示す条件を守ってください。この条件を満たさない状態でエミュレータを動作させた場合、本エミュレータ、ユーザプログラムおよびユーザシステムは正常に動作しないおそれがあります。

表3.1 使用環境条件

項番	項目	仕 様
1	温度	動作時 : 10 ~ 35 非動作時 : - 10 ~ 50
2	湿度	動作時 : 35 ~ 80% 結露なし 非動作時 : 35 ~ 80% 結露なし
3	振動	動作時 : 最大 2.45m/s ² 非動作時 : 最大 4.9m/s ² 梱包輸送時 : 最大 14.7m/s ²
4	AC 入力電源	電圧 : AC100V ± 10% 周波数 : 50/60Hz 消費電力 : 75W
5	周囲ガス	腐食性ガスのないこと

3. ハードウェア仕様

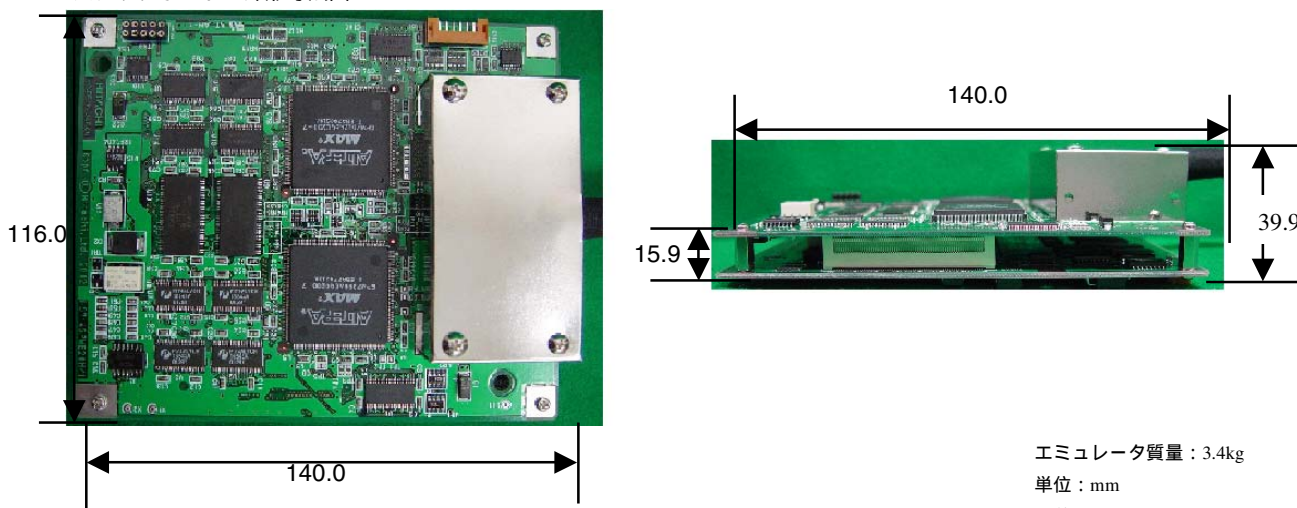
3.2 エミュレータの外形寸法と質量

図3.1 に E6000H エミュレータの外形寸法と質量を示します。

- E6000Hエミュレータ本体部の外形寸法図



- エバチップボードの外形寸法図



エミュレータ質量：3.4kg

単位：mm

公差：±1.0mm

図 3.1 E6000H エミュレータの外形寸法図

3.3 ユーザシステムインタフェース回路

3.3.1 ユーザシステムインタフェース回路

エミュレータに実装されている SH7046 とユーザシステム間は、バッファおよび抵抗等が入っている信号があります。したがって、エミュレータとユーザシステムを接続する場合、信号の遅れおよび FANIN、FANOUT に注意してください。

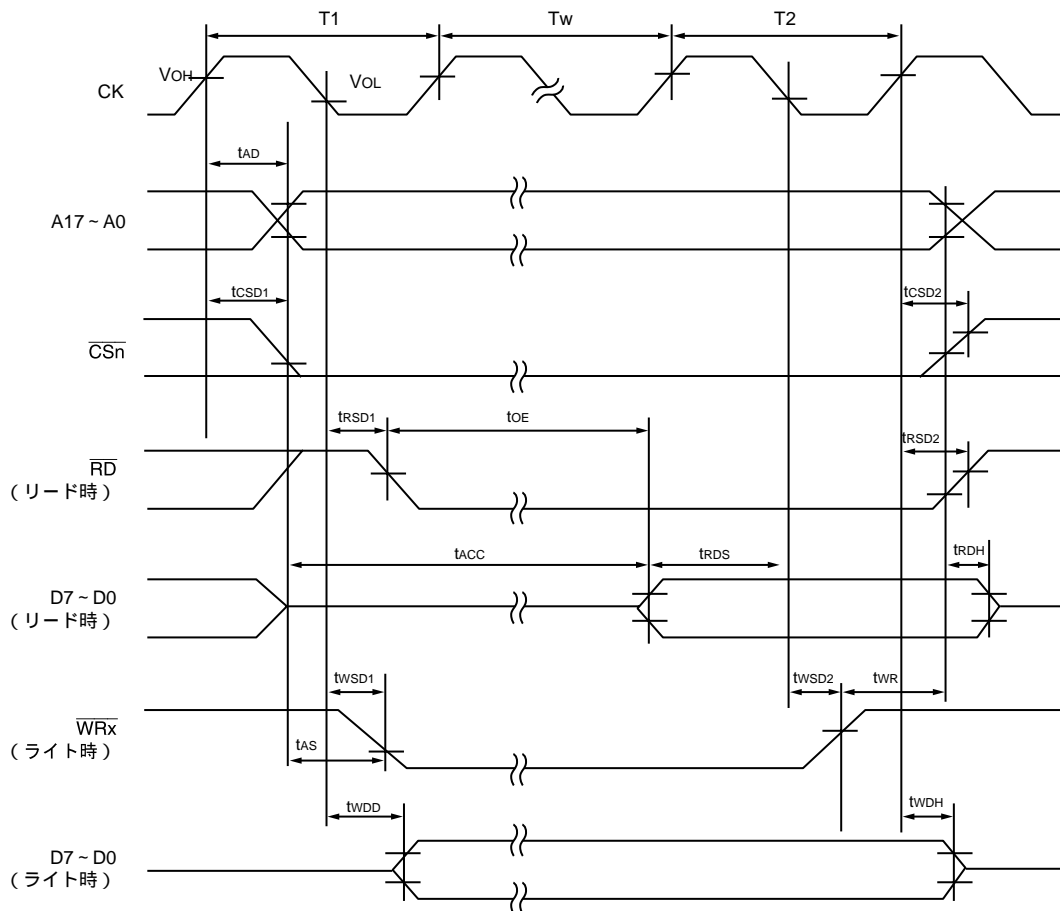
本エミュレータを使用した場合の AC タイミング値を表 3.2 に示します。

【注】 表 3.2 のエミュレータ使用時の値は実測した値で、参考値です。保証値ではありません。

表 3.2 エミュレータ使用時のバスタイミング (外部バスクロック : 50.0MHz)

項目	MCU 仕様 (ns)		エミュレータ使用時 (ns)	
	MIN	MAX	MIN	MAX
tRDS	15		16	
tACC	$tcyc \times (n+2) - 35$ (nはウェイト数)		$tcyc \times (n+2) - 37$ (nはウェイト数)	

図3.2に基本バサイクル(ソフトウェアウェイト)、図3.3~図3.8にユーザシステムと接続するエミュレータのインタフェース回路を示します。



【注】 tRDH: A17~A0、CSn、RDの最も早いネゲートタイミングから規定

図 3.2 基本バサイクル(ソフトウェアウェイト)

3. ハードウェア仕様

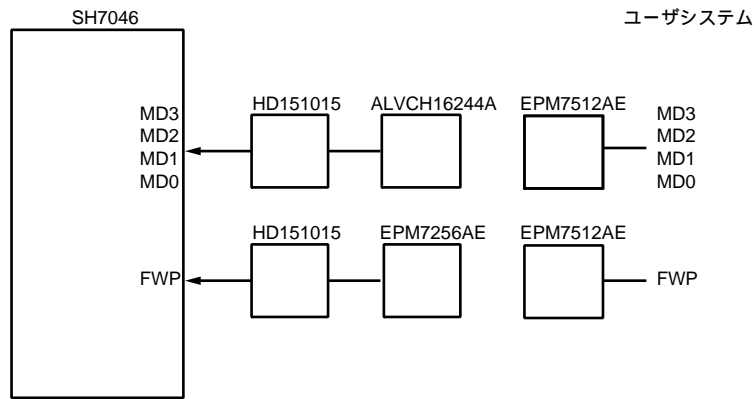


図 3.3 ユーザインタフェース回路 (1)

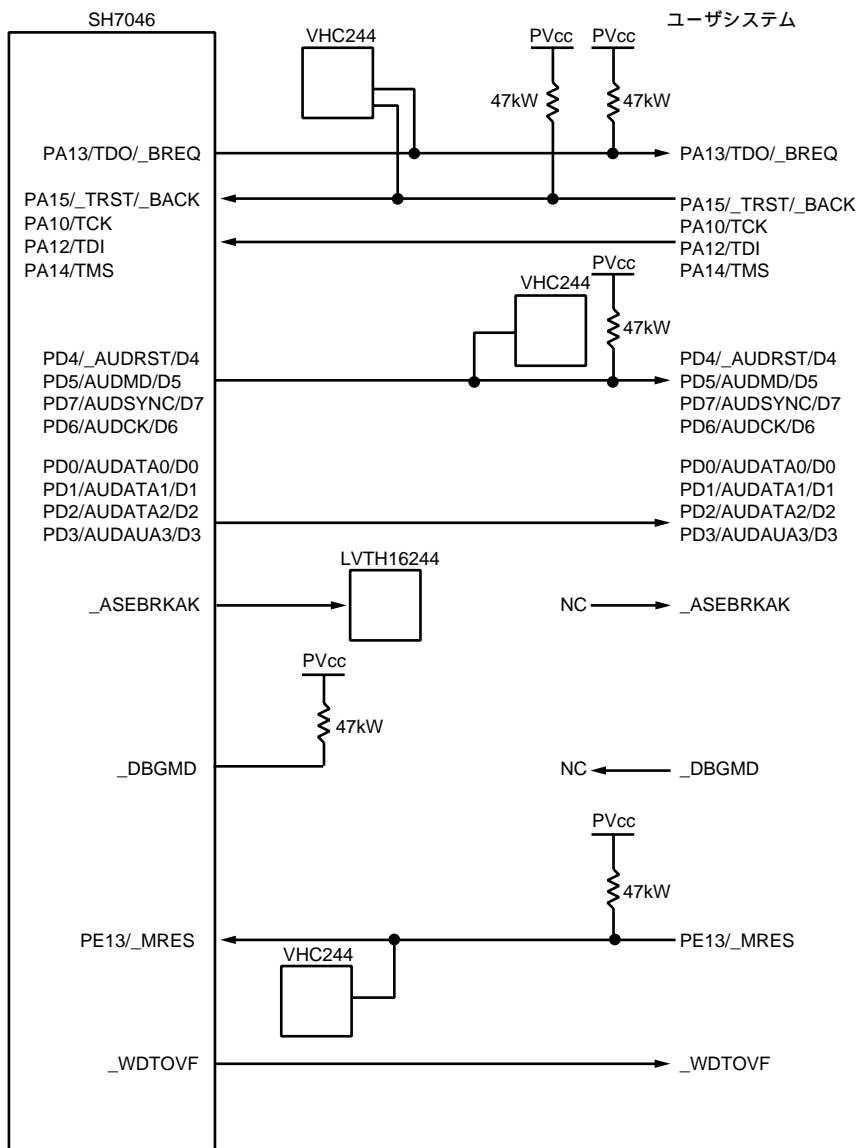


図 3.4 ユーザインタフェース回路 (2)

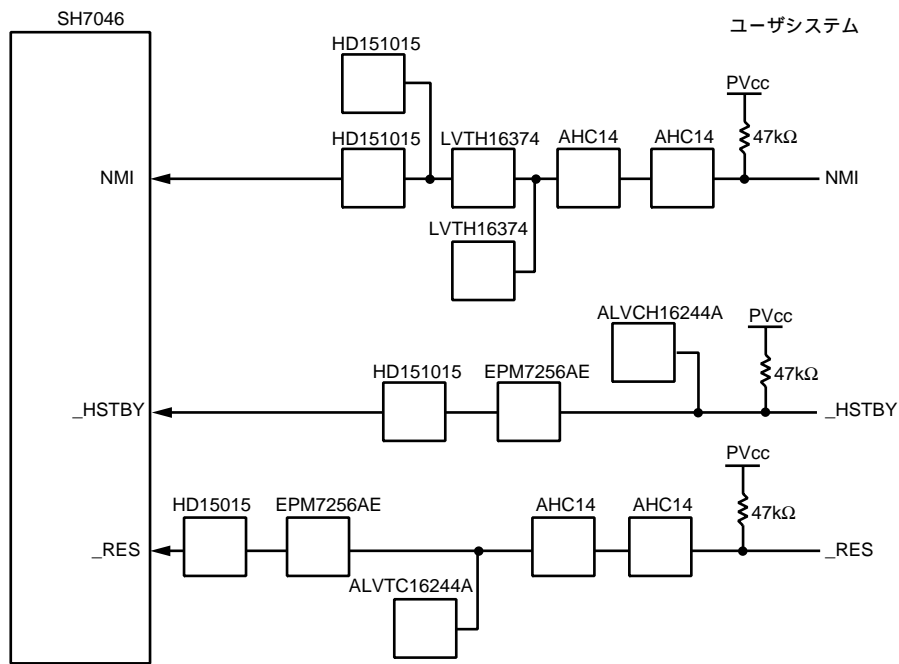


図 3.5 ユーザインタフェース回路 (3)

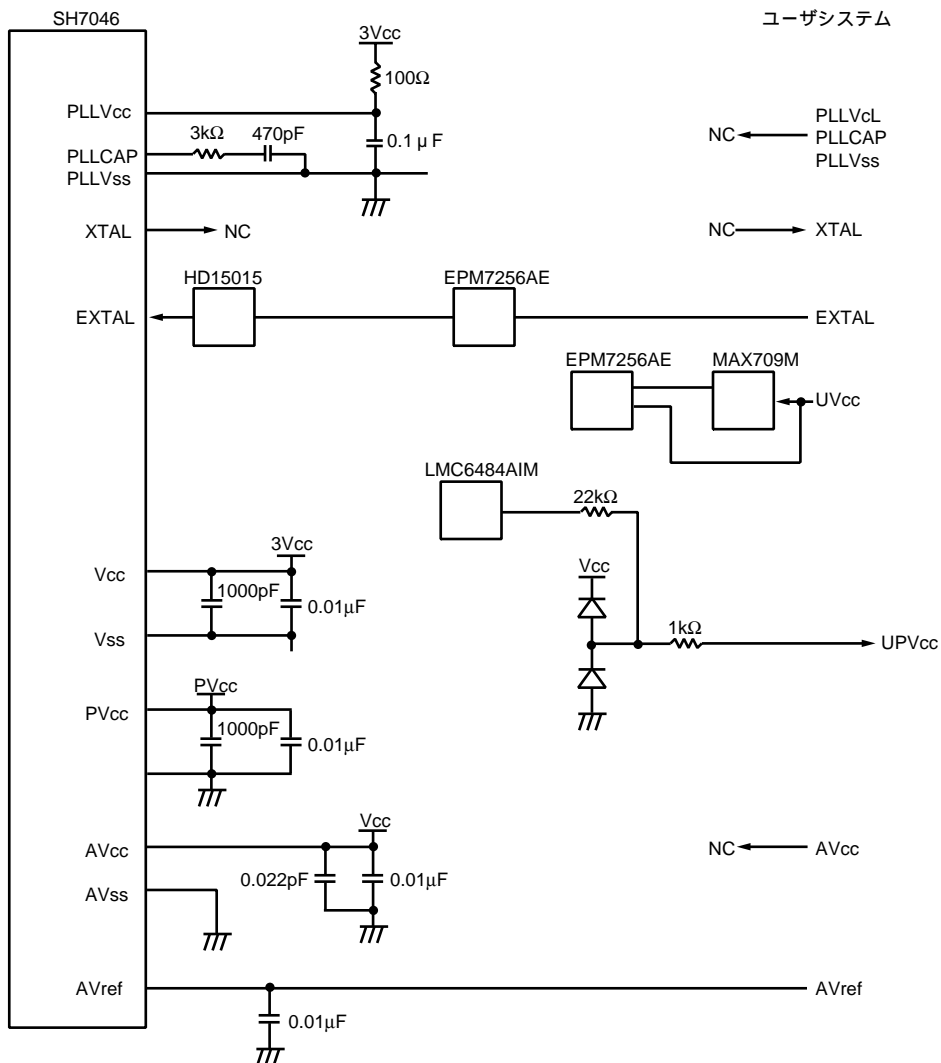


図 3.6 ユーザインタフェース回路 (4)

3. ハードウェア仕様

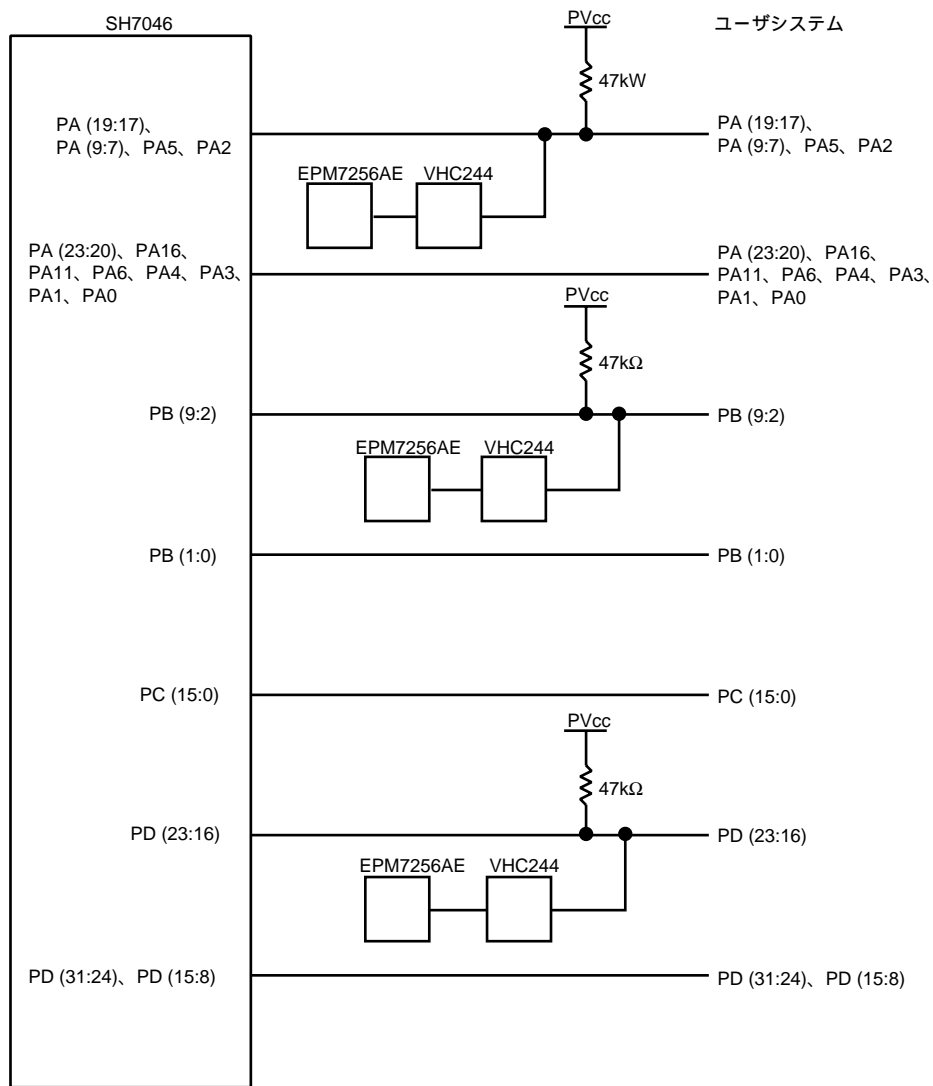


図 3.7 ユーザインタフェース回路 (5)

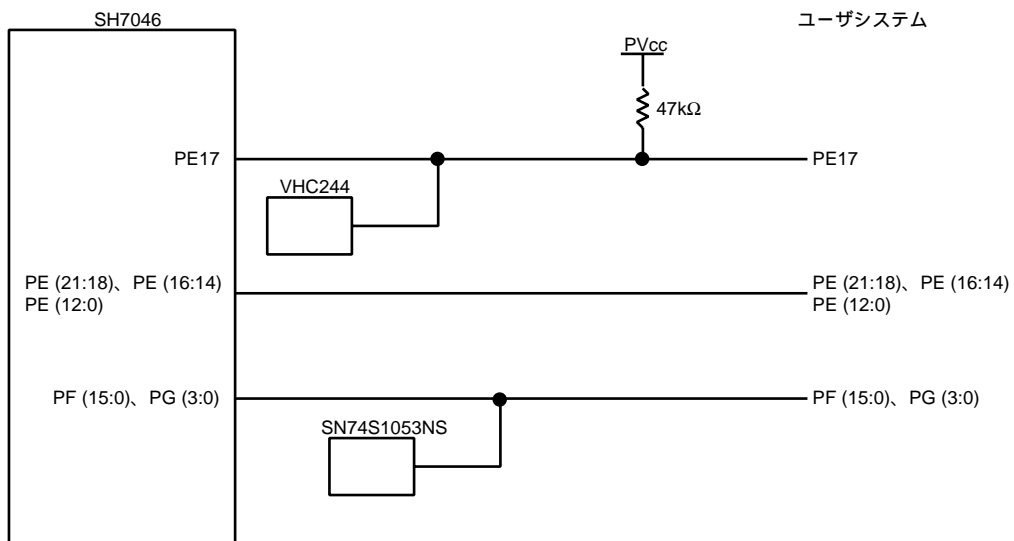


図 3.8 ユーザインタフェース回路 (6)

3.3.2 ユーザシステムインタフェースのディレイ時間

_RES 信号はエバチップボード上の論理を介してユーザシステムに接続されますので、ユーザシステムから SH7046 に信号が入力されるまで、表 3.3 に示すディレイ時間が生じます。

表 3.3 エバチップボード経由信号のディレイ時間

No	信号名	ディレイ時間 (ns)
1	_RES	15.0

3.4 エミュレータとユーザシステムの接続

3.4.1 ユーザシステムとの接続方法



警告

エミュレータおよびユーザシステムのパワーオン時、すべてのケーブル類、およびソケット類の抜き差しを行ないてください。

抜き差しを行った場合、エミュレータとユーザシステムの発煙発火の可能性があります。また、デバッグ中のユーザプログラムの破壊の可能性があります。

エミュレータとユーザシステムとの接続方法として、ユーザシステムインタフェースボードを使用します。

表 3.4 使用するユーザシステムインタフェースボードとユーザインタフェース

ユーザシステムインタフェースボードのタイプ	ユーザインタフェース
HS7046ECH61H	FP-80Q (IC149-080-017-B5) * ¹
HS7047ECH61H	FP-100M (IC149-100-054-B51) * ¹
HS7144ECH61H	FP-112B (IC149-112C15333-0B) * ¹
HS7145ECH61H	FP-144F (NQPAC144SD) * ²

【注】 *¹ IC149 シリーズは山一電機株式会社製

*² NQPAC シリーズは東京エレテック株式会社製

(1) ICソケットの取り付け

1. ICソケットの実装

各パッケージ用ICソケットをユーザシステムに実装します。1 ピンの位置を確認した上、はんだ付け前にICソケットの底面をエポキシ樹脂系の接着剤でユーザシステムに固定してください。

2. ICソケットのはんだ付け

ICソケットの固定の後、はんだ付けします。

はんだ付けされるリードの端面には必ずフィレットが生成されるようにしてください(通常より多めにはんだの量を調整してください)。

3. ハードウェア仕様

(2) HS7046ECH61Hを使用した接続



警告

エミュレータおよびユーザシステムのパワーオン時、すべてのケーブル類、およびソケット類の抜き差しを行わないでください。

抜き差しを行った場合、エミュレータとユーザシステムの発煙発火の可能性があります。また、デバッグ中のユーザプログラムの破壊の可能性があります。

- 【注】
1. HS7046ECH61Hの詳しい取り扱いについては製品に付属の取り扱い説明書を参照してください。
 2. 指定した QFP ソケット (IC149-080-017-B5) 以外の IC ソケットを使用した場合、本ユーザシステムインタフェースボードは使用できません。

エミュレータを接続するユーザシステム上に FP-80Q 対応 IC ソケット (IC149-080-017-B5 : 山一電機 (株) 製) を用意してください。ピン配置は SH7046 実チップと全く同じになっていますので、ハードウェアマニュアルのピン配置を参照してください。図 3.9 に HS7046ECH61H を使用した接続図、図 3.10 に HS7046ECH61H を使用した際の部品配置制限、図 3.11 にユーザシステム側 IC ソケット推奨マウントパッド寸法を示します。

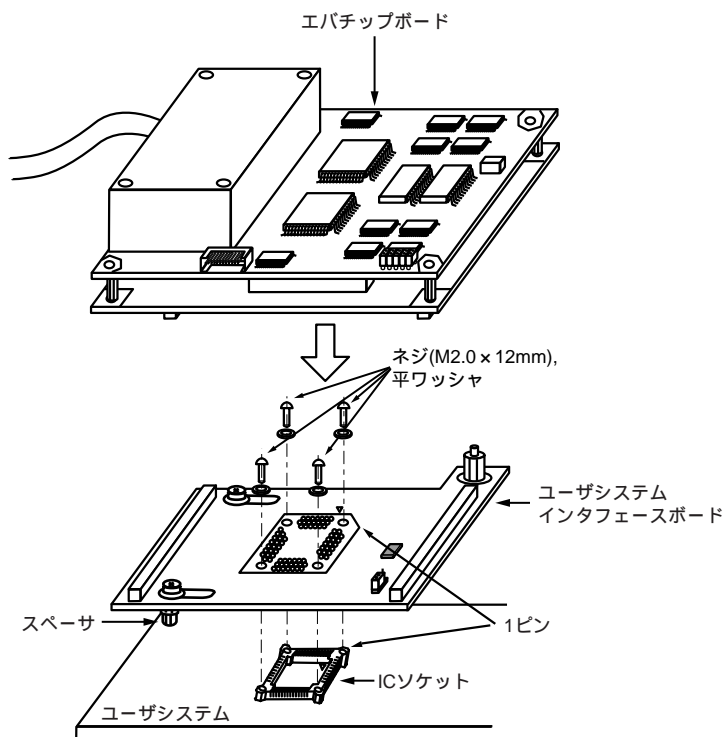


図 3.9 HS7046ECH61H を使用した接続

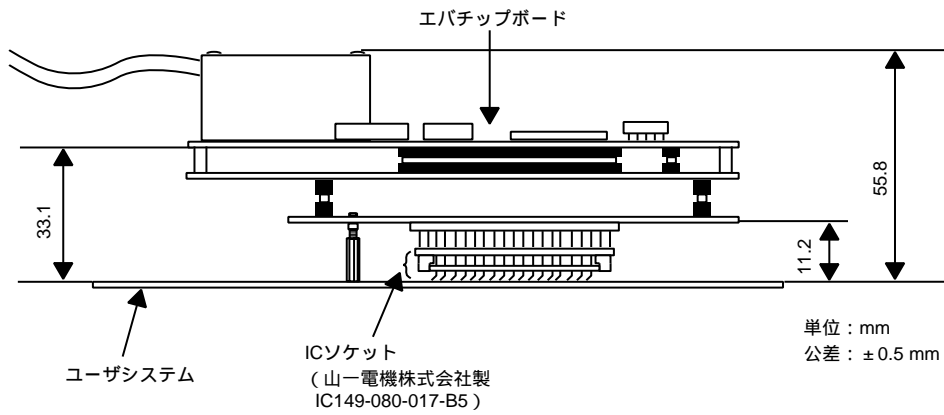


図 3.10 部品配置制限

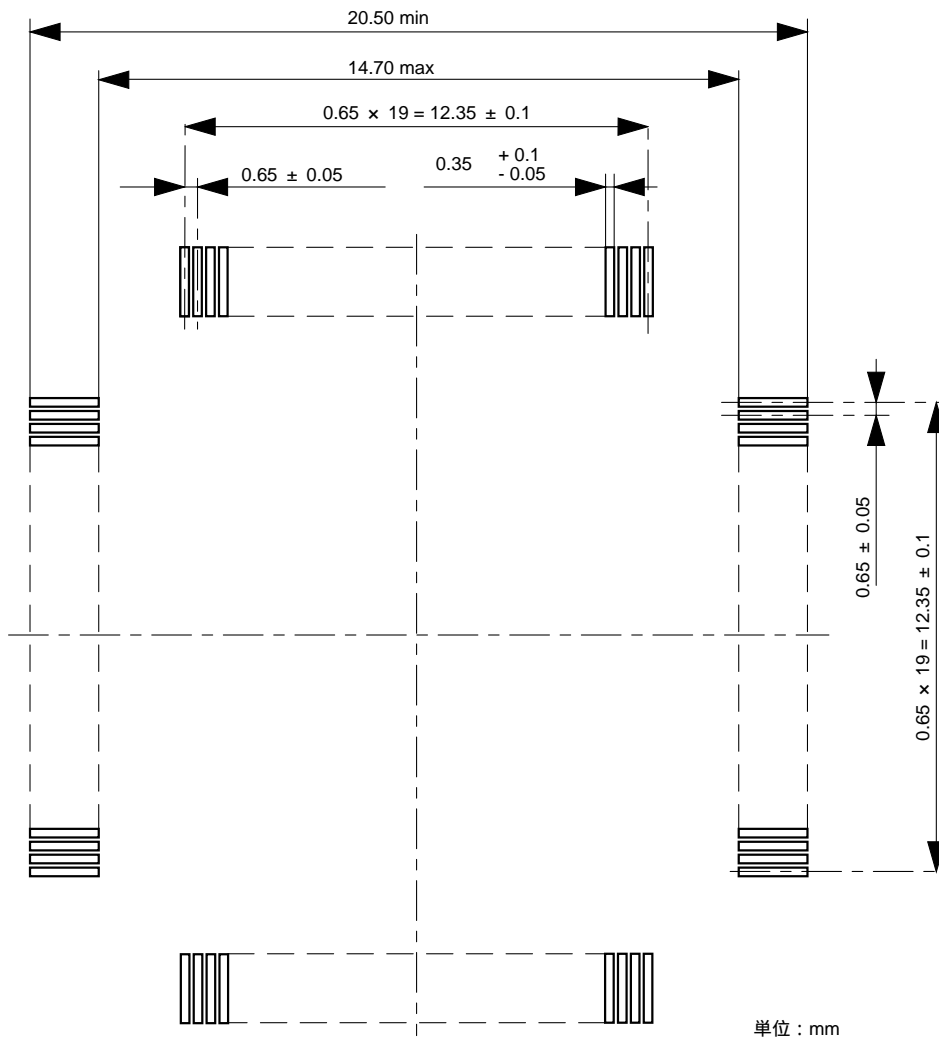


図 3.11 ユーザシステム側 IC ソケット推奨マウントパッド寸法

3. ハードウェア仕様

(3) HS7047ECH61Hを使用した接続

警告

エミュレータおよびユーザシステムのパワーオン時、すべてのケーブル類、およびソケット類の抜き差しを行わないでください。

抜き差しを行った場合、エミュレータとユーザシステムの発煙発火の可能性があります。また、デバッグ中のユーザプログラムの破壊の可能性があります。

- 【注】
1. HS7047ECH61Hの詳しい取り扱いについては製品に付属の取り扱い説明書を参照してください。
 2. 指定したQFPソケット（IC149-100-054-B51）以外のICソケットを使用した場合、本ユーザシステムインタフェースボードは使用できません。

エミュレータを接続するユーザシステム上にEP-100M対応ICソケット（IC149-100-054-B51：山一電機（株）製）を用意してください。ピン配置はSH7047実チップと全く同じになっていますので、ハードウェアマニュアルのピン配置を参照してください。図3.12にHS7047ECH61Hを使用した接続図、図3.13にHS7047ECH61Hを使用した際の部品配置制限、図3.14にユーザシステム側ICソケット推奨マウントパッド寸法を示します。

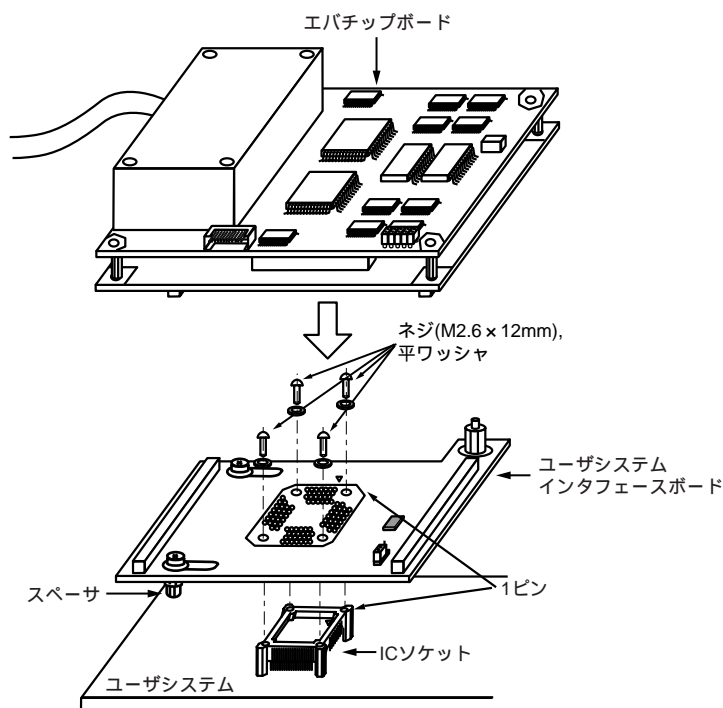


図 3.12 HS7047ECH61H を使用した接続

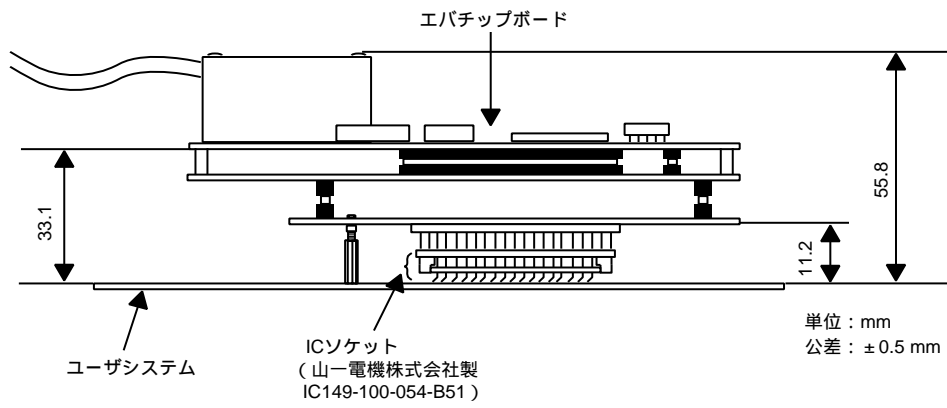


図 3.13 部品配置制限

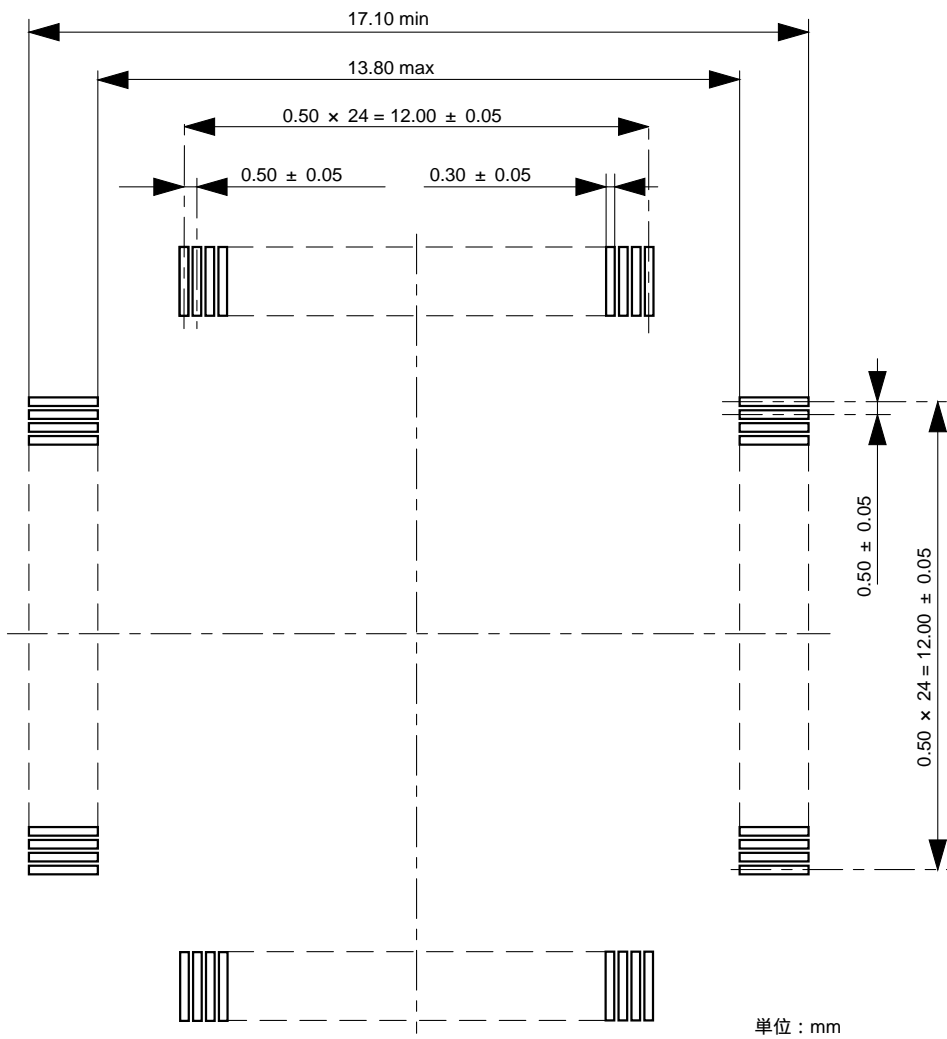


図 3.14 ユーザシステム側 IC ソケット推奨マウントパッド寸法

3. ハードウェア仕様

(4) HS7144ECH61Hを使用した接続



警告

エミュレータおよびユーザシステムのパワーオン時、すべてのケーブル類、およびソケット類の抜き差しを行わないでください。

抜き差しを行った場合、エミュレータとユーザシステムの発煙発火の可能性があります。また、デバッグ中のユーザプログラムの破壊の可能性があります。

- 【注】
1. HS7144ECH61Hの詳しい取り扱いについては製品に付属の取り扱い説明書を参照してください。
 2. 指定したQFPソケット (IC149-112C15333-0B) 以外のICソケットを使用した場合、本ユーザシステムインタフェースボードは使用できません。

エミュレータを接続するユーザシステム上にFP-112B対応ICソケット (IC149-112C15333-0B: 山一電機(株)製)を用意してください。ピン配置はSH7144実チップと全く同じになっていますので、ハードウェアマニュアルのピン配置を参照してください。図3.15にHS7144ECH61Hを使用した接続図、図3.16にHS7144ECH61Hを使用した際の部品配置制限、図3.17にユーザシステム側ICソケット推奨マウントパッド寸法を示します。

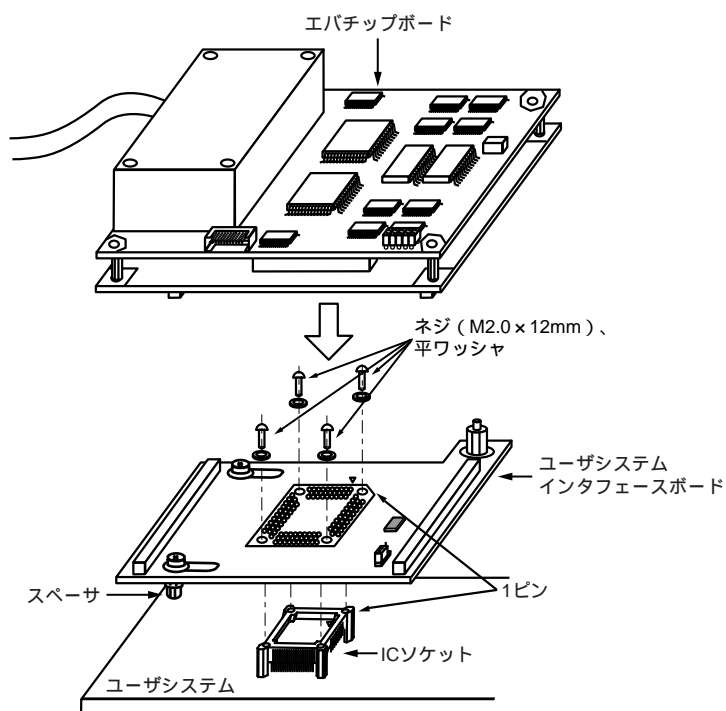


図 3.15 HS7144ECH61H を使用した接続

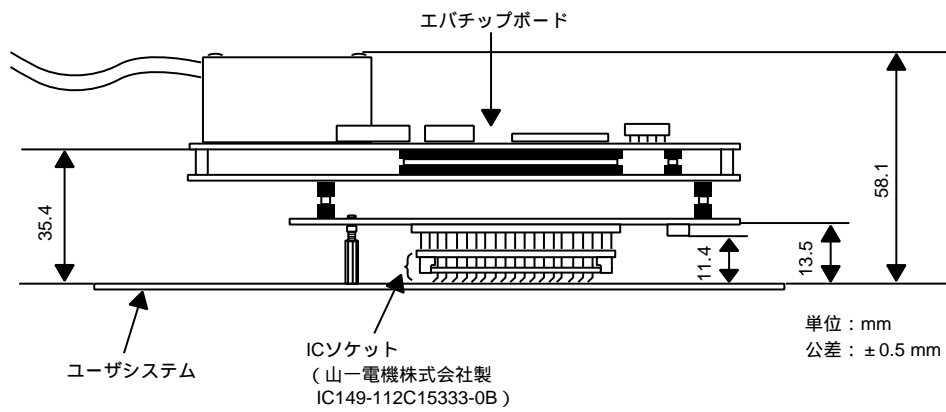


図 3.16 部品配置制限

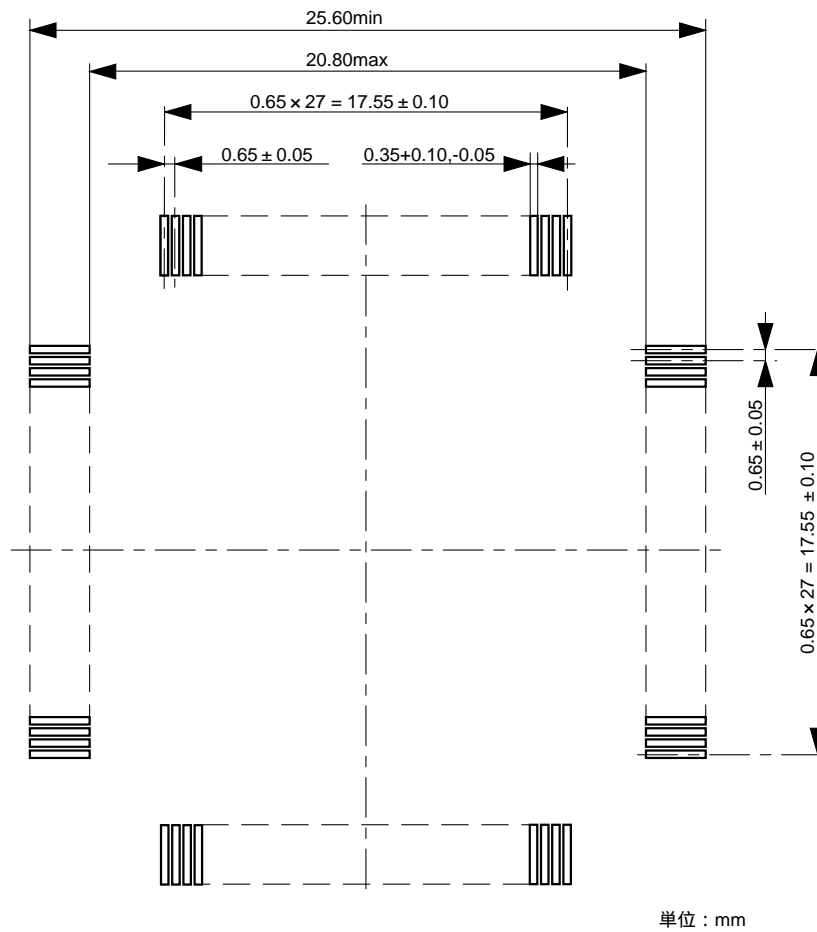


図 3.17 ユーザシステム側 IC ソケット推奨マウントパッド寸法

3. ハードウェア仕様

(5) HS7145ECH61Hを使用した接続

警告

エミュレータおよびユーザシステムのパワーオン時、すべてのケーブル類、およびソケット類の抜き差しを行わないでください。

抜き差しを行った場合、エミュレータとユーザシステムの発煙発火の可能性があります。また、デバッグ中のユーザプログラムの破壊の可能性があります。

- 【注】
1. HS7145ECH61Hの詳しい取り扱いについては製品に付属の取り扱い説明書を参照してください。
 2. 指定したQFPソケット(NQPACK144SD)以外のICソケットを使用した場合、本ユーザシステムインタフェースボードは使用できません。

エミュレータを接続するユーザシステム上にFP-144F対応ICソケット(NQPACK144SD:東京エレクトック(株)製)を用意してください。ピン配置はSH7145実チップと全く同じになっていますので、ハードウェアマニュアルのピン配置を参照してください。図3.18にHS7145ECH61Hを使用した接続図、図3.19にHS7145ECH61Hを使用した際の部品配置制限、図3.20にユーザシステム側ICソケット推奨マウントパッド寸法を示します。

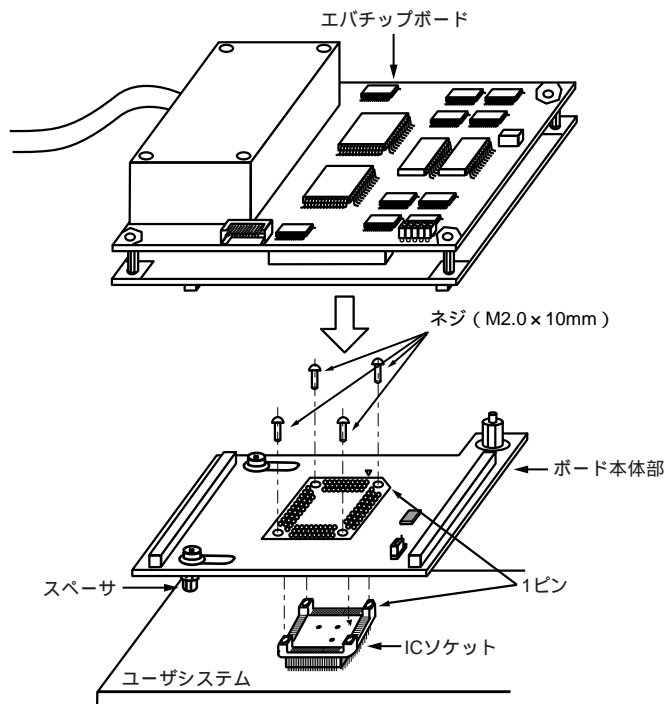


図 3.18 HS7145ECH61H を使用した接続

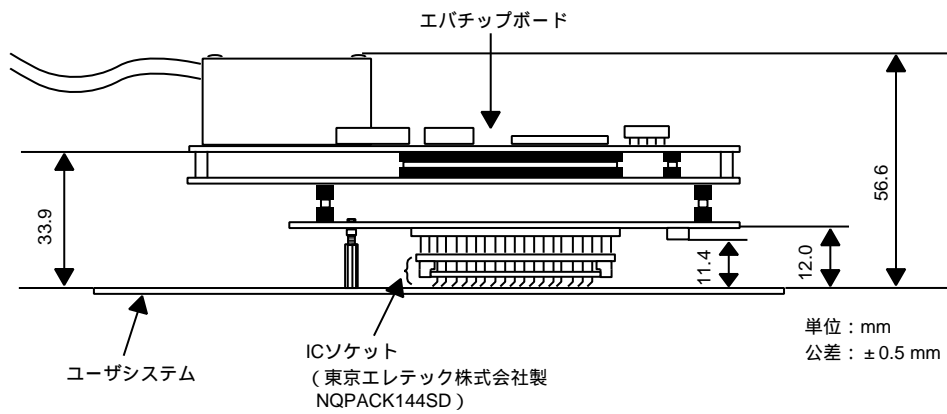


図 3.19 部品配置制限

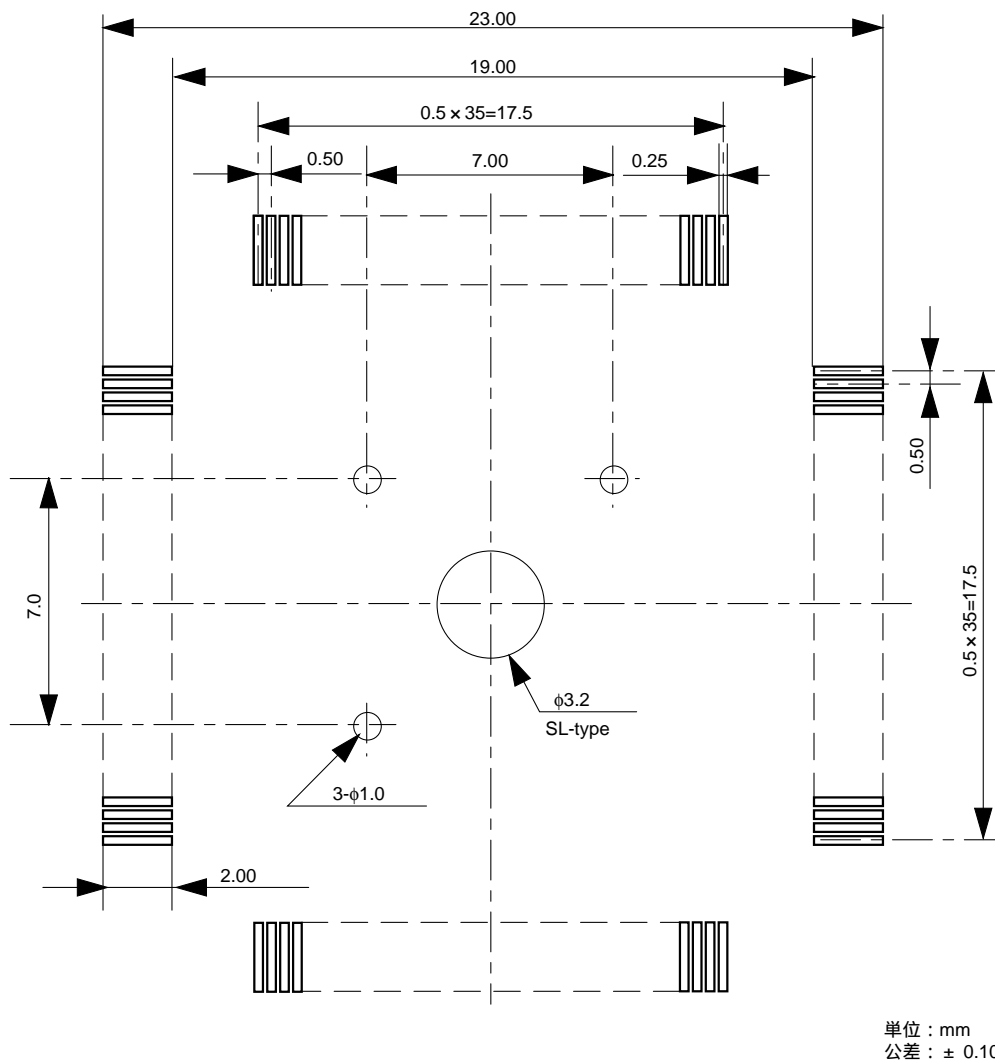


図 3.20 ユーザシステム側 IC ソケット推奨マウントパッド寸法

3. ハードウェア仕様

(6) 専用コネクタを使用した接続



警告

エミュレータおよびユーザシステムのパワーオン時、すべてのケーブル類、およびソケット類の抜き差しを行わないでください。

抜き差しを行った場合、エミュレータとユーザシステムの発煙発火の可能性があります。また、デバッグ中のユーザプログラムの破壊の可能性があります。

【注】 指定した専用コネクタ（WD-200P-VF85-N：日本航空電子（株）製）×2以外のコネクタを使用した場合、本エバチップボードは使用できません。

エミュレータを接続するユーザシステム上に専用コネクタ（WD-200P-VF85-N：日本航空電子（株）製）を用意してください。

図 3.21 に専用コネクタを使用した接続図、図 3.22 に専用コネクタを使用した際の部品配置制限、図 3.23 にユーザシステム側コネクタ実装位置を示します。

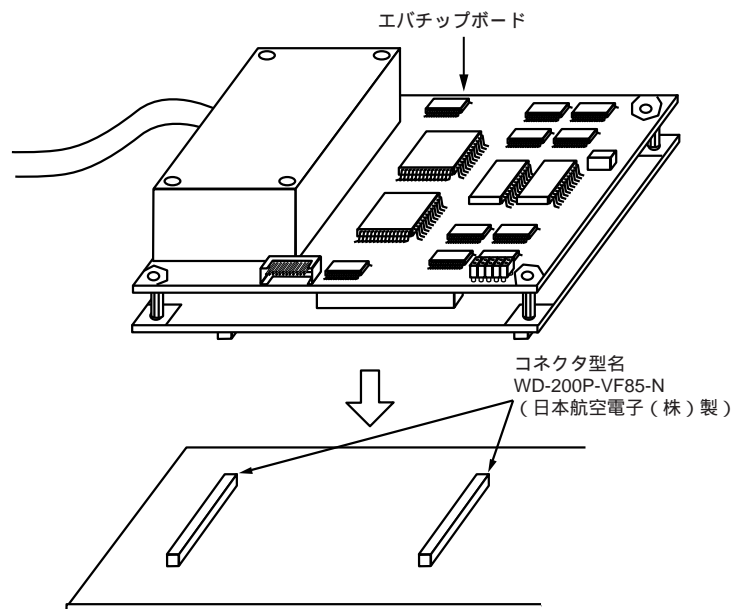


図 3.21 専用コネクタを使用した接続

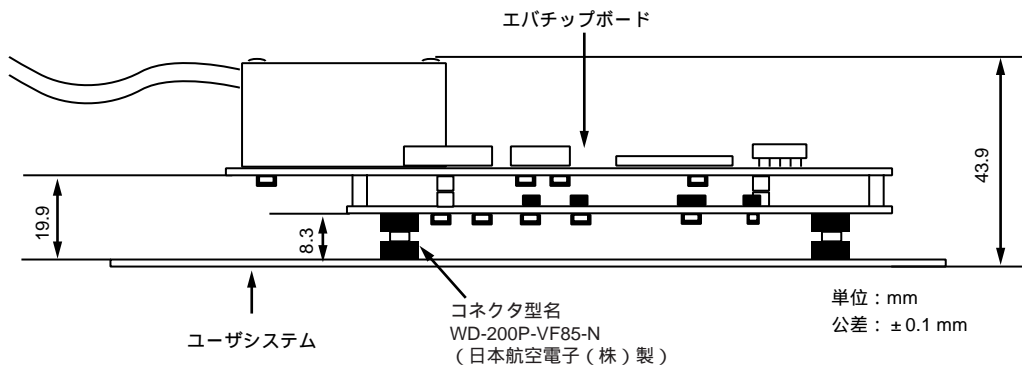


図 3.22 部品配置制限

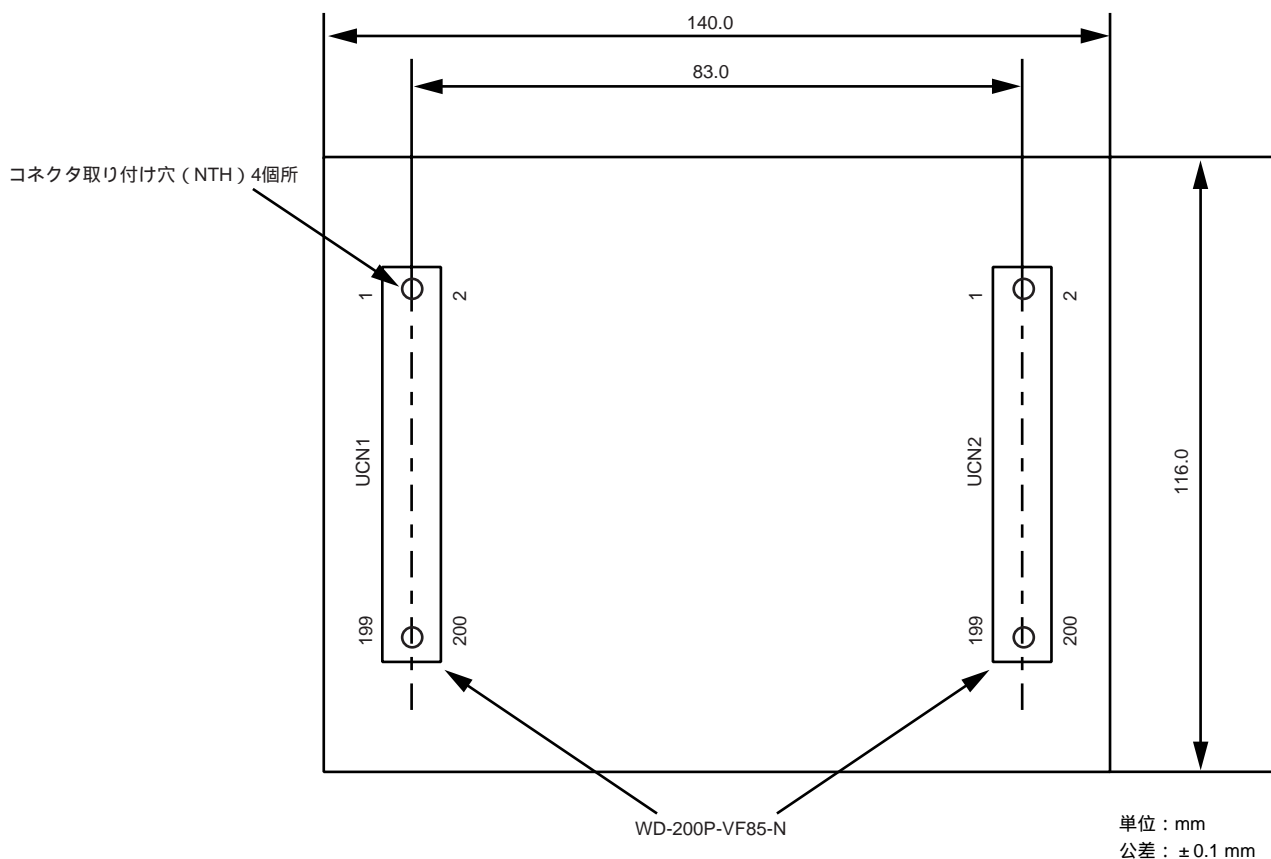


図 3.23 ユーザシステム側コネクタの実装位置

フットパターン寸法については、WD-200P-VF85-N のカタログを参照して設計してください。

3. ハードウェア仕様

3.4.2 ユーザシステムインタフェースコネクタのピン配置

HS7046EPH60H のユーザシステムインタフェースコネクタのピン配置を表 3.5 に示します。

表 3.5 HS7046EPH60H ピン配置

User I/F 1	Pin No.	Signal Name	User I/F 1	Pin No.	Signal Name
UCN1	1	GND	UCN1	41	N.C. (SIOWORD_N)
	2	GND		42	N.C. (SDB1_P)
	3	GND		43	N.C. (SWAIT_N)
	4	GND		44	N.C. (SDB0_P)
	5	GND		45	N.C. (SIOWR_N)
	6	GND		46	GND
	7	GND (TGBON1)		47	GND
	8	N.C. (SDB15_P)		48	N.C. (SHCANSTART_P)
	9	N.C. (SMS3_N)		49	N.C.
	10	N.C. (SDB14_P)		50	N.C.
	11	N.C. (SMS2_N)		51	N.C.
	12	N.C. (SDB13_P)		52	N.C.
	13	N.C. (SMS1_N)		53	N.C.
	14	N.C. (SDB12_P)		54	N.C.
	15	N.C. (SMS0_N)		55	N.C.
	16	GND		56	GND
	17	GND		57	GND
	18	N.C. (SDB11_P)		58	N.C.
	19	N.C. (SMSB3_N)		59	N.C.
	20	N.C. (SDB10_P)		60	N.C.
	21	N.C. (SMSB2_N)		61	N.C.
	22	N.C. (SDB9_P)		62	N.C.
	23	N.C. (SMSB1_N)		63	N.C.
	24	N.C. (SDB8_P)		64	N.C.
	25	N.C. (SMSB0_N)		65	GND
	26	GND		66	N.C.
	27	GND		67	N.C.
	28	N.C. (SDB7_P)		68	GND
	29	N.C. (SE_N)		69	N.C.
	30	N.C. (SDB6_P)		70	N.C.
	31	N.C. (SMSPT_N)		71	N.C.
	32	N.C. (SDB5_P)		72	N.C.
	33	N.C. (SMSE_N)		73	N.C.
	34	N.C. (SDB4_P)		74	N.C.
	35	N.C. (SIORD_N)		75	N.C.
	36	GND		76	N.C.
	37	GND		77	GND
	38	N.C. (SDB3_P)		78	GND
	39	N.C. (SSSSTBY_N)		79	PE21/PWOBB/SCK4/A15
	40	N.C. (SDB2_P)		80	N.C.

表 3.5 HS7046EPH60H ピン配置 (続き)

User I/F 1	Pin No.	Signal Name	User I/F 1	Pin No.	Signal Name
UCN1	81	PE20/PVOB/TXD4/A14	UCN1	121	5VCC
	82	PE13/TIOC4B/_MRES		122	PD2/D2/SCK2/AUDATA2
	83	PE19/PUOB/RXD4/A13		123	N.C.
	84	PE12/TIOC4A/TXD1/TXD3		124	PD3/D3/-/AUDATA3
	85	PE18/PWOA/-/A12		125	GND
	86	PE11/TIOC3D/RXD1/RXD3		126	PD4/D4/-/_AUDRET
	87	GND		127	GND
	88	GND		128	GND
	89	PE17/PVOA/_WAIT/A11		129	GND
	90	PE10/TIOC3C/TXD2/_WRL/HRxD0&HRxD		130	PD5/D5/-/AUDMD
	91	PE16/PUOA/_UBCTRG/A10		131	PB9/_IRQ7/A21/_ADTRG/-/-/-
	92	PE9/TIOC3B/SCK1/SCK3/HTxD0&HTxD1		132	PD6/D6/-/AUDCK
	93	PE15/TIOC4D/DACK1/_IRQOUT		133	PB8/_IRQ6/A20/_WAIT/-/-/-
	94	PE8/TIOC3A/SCK2/_WRH		134	PD7/D7/-/_AUDSYNC
	95	PE14/TIOC4C/DACK0		135	GND
	96	PE7/TIOC2B/RXD2/A9		136	PD8/D8/-/_UBCTRG
	97	N.C.		137	PB7/_IRQ5/A19/_BREQ/-/-/-
	98	GND		138	GND
	99	3VCC		139	PB6/_IRQ4/A18/_BACK/-/-/-
	100	PE6/TIOC2A/SCK3/A8		140	PD9/D9/-/
	101	3VCC		141	PB5/_IRQ3/_POE/-/SDA1/CK/HRxD2/-
	102	PE5/TIOC1B/TXD3/A7		142	PD10/D10/-/
	103	3VCC		143	PB4/_IRQ2/_POE2/-/SCL1/_POEHIZ/HTxD2/SC K4
	104	PE4/TIOC1A/RXD3/A6		144	PD11/D11/-/
	105	3VCC		145	PB3/_IRQ1/_POE1/-/SDA0/_VCOSTOP/-/TXD4
	106	PE3/TIOC0D/DRAK1/_CS3		146	PD12/D12/-/
	107	3VCC		147	N.C.
	108	GND		148	GND
	109	3VCC		149	PB2/_IRQ0/_POE0/-/SCL0/_OSCER/-/RXD4
	110	PE2/TIOC0C/_DREQ1/_CS2		150	PD13/D13/-/
	111	5VCC		151	PB1/A17/HRxD0/HRxD1/-/-/-/SCK4
	112	PE1/TIOC0B/DRAK0/_CS1		152	PD14/D14/-/
	113	5VCC		153	PB0/A16/HTxD0/HTxD1/-/-/-/
	114	PE0/TIOC0A/_DREQ0/_CS0		154	PD15/D15/-/
	115	5VCC		155	PC15/A15/-/
	116	PD0/D0/RXD2/AUDATA0		156	PD16/D16/_IRQ0/AUDATA0
117	5VCC	157	GND		
118	GND	158	GND		
119	5VCC	159	PC14/A14/-/		
120	PD1/D1/TXD2/AUDATA1	160	PD17/D17/_IRQ1/AUDATA1		

3. ハードウェア仕様

表 3.5 HS7046EPH60H ピン配置 (続き)

User I/F 1	Pin No.	Signal Name	User I/F 2	Pin No.	Signal Name
UCN1	161	PC13/A13/-/-	UCN2	1	GND
	162	PD18/D18/_IRQ2/AUDATA2		2	GND
	163	PC12/A12/-/-		3	GND
	164	PD19/D19/_IRQ3/AUDATA3		4	GND
	165	PC11/A11/-/-		5	GND
	166	GND		6	GND
	167	GND		7	GND
	168	PD20/D20/_IRQ4/_AUDRST		8	GND
	169	PC10/A10/-/-		9	N.C. (S0CLR0_N)
	170	PD21/D21/_IRQ5/AUDMD		10	N.C. (S1CLR0_N)
	171	PC9/A9/-/-		11	N.C. (S0CLR1_N)
	172	PD22/D22/_IRQ6/AUDCK		12	N.C. (S1CLR1_N)
	173	PC8/A8/-/-		13	N.C. (S0CLR2_N)
	174	PD23/D23/_IRQ7/_AUDSYNC		14	N.C. (S1CLR2_N)
	175	PC7/A7/-/-		15	N.C. (S0CLR3_N)
	176	GND		16	N.C. (S1CLR3_N)
	177	GND		17	GND
	178	PD24/D24/_DREQ0/_UBCTRG		18	GND
	179	PC6/A6/-/-		19	N.C. (S2CLR0_N)
	180	PD25/D25/_DREQ1/-		20	N.C. (S3CLR0_N)
181	PC5/D5/-/-	21	N.C. (S2CLR1_N)		
182	PD26/D26/_DACK0/-	22	N.C. (S3CLR1_N)		
183	PC4/A4/-/-	23	N.C. (S2CLR2_N)		
184	PD27/D27/_DACK1/-	24	N.C. (S3CLR2_N)		
185	GND	25	N.C. (S2CLR3_N)		
186	GND	26	N.C. (S3CLR3_N)		
187	PC3/A3/-/-	27	GND		
188	PD28/D28/_CS2/-	28	GND		
189	PC2/A2/-/-	29	N.C. (SAB0_P)		
190	PD29/D29/_CS3/-	30	MD4		
191	PC1/A1/-/-	31	N.C. (SAB1_P)		
192	PD30/D30/_IRQOUT/-	32	MD3		
193	PC0/A0/-/-/HRxD1	33	N.C. (SAB2_P)		
194	PD31/D31/_ADTRG/-	34	MD2		
195	GND	35	N.C. (SAB3_P)		
196	GND	36	MD1		
197	GND	37	GND		
198	GND	38	GND		
199	GND	39	N.C. (SAB4_P)		
200	GND	40	MD0		

表 3.5 HS7046EPH60H ピン配置 (続き)

User I/F 2	Pin No.	Signal Name	User I/F 2	Pin No.	Signal Name
UCN2	41	N.C. (SAB5_P)	UCN2	81	N.C. (S1REQ0_N)
	42	NMI		82	N.C. (SMMTADTRG_N)
	43	N.C. (SAB6_P)		83	N.C. (S1REQ1_N)
	44	_HSTBY		84	N.C. (S2REQ0_N)
	45	N.C. (SAB7_P)		85	N.C. (S1REQ2_N)
	46	N.C.		86	N.C. (S2REQ1_N)
	47	GND		87	GND
	48	VCC		88	N.C. (S2REQ2_N)
	49	N.C. (SAB8_P)		89	N.C. (S1REQ3_N)
	50	VCC		90	N.C. (S2REQ3_N)
	51	N.C. (SAB9_P)		91	N.C. (S3REQ0_N)
	52	VCC		92	N.C.
	53	N.C. (SAB10_P)		93	N.C. (S3REQ1_N)
	54	PVCC		94	N.C.
	55	N.C. (SAB11_P)		95	N.C. (S3REQ2_N)
	56	PVCC		96	N.C.
	57	GND		97	N.C. (S3REQ3_N)
	58	PVCC		98	N.C.
	59	N.C. (SVID4_N)		99	N.C. (S3REQ4_N)
	60	PVCC		100	N.C.
	61	N.C. (SVID3_N)		101	GND
	62	PVCC		102	AVCC
	63	N.C. (SVID2_N)		103	N.C.
	64	PVCC		104	AVCC
	65	N.C. (SVID1_N)		105	N.C. (EVAVCC)
	66	N.C.		106	AVCC
	67	GND		107	N.C. (EVAVCC)
	68	GND		108	AVCC
	69	N.C. (SVID0_N)		109	N.C. (EVAVCC)
	70	_RES		110	AVref
	71	N.C. (S0REQ0_N)		111	N.C. (EVAVCC)
	72	N.C. (SADTRG2_N)		112	AVref
	73	N.C. (S0REQ1_N)		113	N.C.
	74	N.C. (SADTRG1_N)		114	AVSS
	75	N.C. (S0REQ2_N)		115	N.C.
	76	N.C. (SADTRG0_N)		116	AVSS
77	GND	117	N.C.		
78	GND	118	AVSS		
79	N.C. (S0REQ3_N)	119	N.C.		
80	N.C. (SMTUADTRG_N)	120	AVSS		

3. ハードウェア仕様

表 3.5 HS7046EPH60H ピン配置 (続き)

User I/F 2	Pin No.	Signal Name	User I/F 2	Pin No.	Signal Name
UCN2	121	GND	UCN2	161	GND
	122	PF0/AN0		162	PG0/AN16
	123	N.C.		163	PA14/_RD/POE5/D6/TMS/-/-/-
	124	PF1/AN1		164	PG1/AN17
	125	N.C.		165	PA15/CK/_POE6/D7/_TRES/_BACK/-/-
	126	PF2/AN2		166	PG2/AN18
	127	PA0/RXD0/-/-/A0/_POE0/RXD2/-		167	PA16/-/_POE7/-/-/-/-/-
	128	PF3/AN3		168	PG3/AN19
	129	PA1/TXD0/-/-/A1/_POE1/TXD2/-		169	PA17/_WAIT/PCIO/-/-/-/-/-
	130	GND		170	N.C.
	131	GND		171	GND
	132	PF4/AN4		172	AVref
	133	PA2/SCK0/_DREQ0/_IRQ0/A2/PCIO/ACK2/-		173	PA18/_BREQ/DRAK0/-/-/-/-/-
	134	PF5/AN5		174	AVref
	135	PA3/RXD1/-/-/WRL/A3/_POE4/RXD3/-		175	PA19/_BACK/DRAK1/-/-/-/-/-
	136	PF6/AN6		176	AVref
	137	PA4/TXD1/-/-/WRH/A4/_POE5/TXD3/-		177	PA20/-/-/-/-/-/-/-
	138	PF7/AN7		178	AVref
	139	PA5/SCK1/_DREQ1/_IRQ1/A5/_POE6/ACK3/-		179	PA21/-/-/-/-/-/-/-
	140	GND		180	N.C.
	141	GND		181	GND
	142	PF8/AN8		182	EXTAL
	143	PA6/TCLKA/_CS2/_RD/-/RXD2/-/-		183	PA22/_WRHL/-/-/-/-/-/-
	144	PF9/AN9		184	GND
	145	PA7/TCLKB/_CS3/_WAIT/-/TXD2/-/-		185	PA23/_WRHH/-/-/-/-/-/-
	146	PF10/AN10		186	N.C. (CKEO_P)
	147	PA8/TCLKC/_IRQ2/D0/-/RXD3/-/-		187	_WDTOVF
	148	PF11/AN11		188	GND
	149	PA9/TCLKD/_IRQ3/D1/-/TXD3/-/-		189	PINMD1 【注】
	150	GND		190	FWP
	151	GND		191	PINMD0 【注】
	152	PF12/AN12		192	N.C. (SVRES_N)
	153	PA10/_CS0/_RD/D2/TCK/SCK2/-/-		193	GND
	154	PF13/AN13		194	GND (TGBON2)
	155	PA11/_CS1/_ADTRG/D3/-/SCK3/-/-		195	GND
	156	PF14/AN14		196	GND
	157	PA12/_WRL/_UBCTRG/D4/TDI/-/-/-		197	GND
	158	PF15/AN15		198	GND
	159	PA13/_WRH/_POE4/D5/TDO/_BREQ/-/-		199	GND
	160	GND		200	GND

【注】 PINMD1、PINMD0 については、ターゲットデバイスの認識信号のため、以下に示す表に従い、端子処理をしてください。

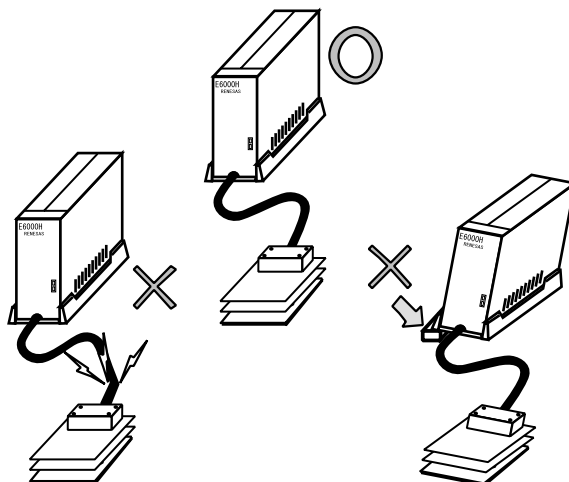
ターゲットデバイス	PINMD1	PINMD0
SH7046	GND	GND
SH7047	GND	Vcc (1k Ω Pull up)
SH7144	Vcc (1k Ω Pull up)	GND
SH7145	Vcc (1k Ω Pull up)	Vcc (1k Ω Pull up)

3.4.3 ユーザシステム接続時の注意事項

ユーザシステムとエバチップボードを接続する際には以下の点にご注意ください。

(1) エミュレータ本体の設置

エミュレータ本体とエバチップボードの位置関係により、トレースケールが大きく曲げられることがあります。このような状態で使用すると、ユーザインタフェース部に大きなストレスが加わり、接点、接触不良等の機械的破損をまねく原因となります。また、使用中にエミュレータ本体が動いてしまうと、ユーザインタフェース部に思わぬストレスを与えることになります。エミュレータ本体の設置位置に十分ご注意ください。



(2) 電源の確認

ユーザシステムに接続する際には、エミュレータ、ユーザシステムとも、電源が投入されていないことを確認してから行ってください。

(3) UVccの接続

エミュレータはユーザシステムの電源 ON/OFF をユーザシステムインタフェースコネクタ (UCN2) の UVcc 端子 (HS7046EPH60H : 48 ピン、50 ピン、52 ピン (VCC)、および 54 ピン、56 ピン、58 ピン、60 ピン、62 ピン、64 ピン (PVCC)) にて監視し、検出しています。ユーザシステムを接続した際にこの端子にユーザ電源を接続されませんと、エミュレータからはユーザシステムが接続されていないと認識されます。

ユーザ実機を接続する際は、この端子にユーザシステムの電源が接続されていることを確認してください。

3.5 ターゲットマイコンのサポート

3.5.1 メモリ空間

SH7046 は、アーキテクチャ上 4GB のメモリ空間を持っています。

(1) 内蔵ROM領域

1. 内蔵ROM領域へのアクセス。

E6000HエミュレータはSH7046の内蔵ROM用の代替RAMを搭載しています。内蔵ROMが存在するモードでは、内蔵ROMをアクセスするとこの代替RAMをアクセスします。内蔵ROM領域のアクセスは、ユーザプログラムの実行によるアクセスとエミュレータのコマンドで次のようになります。

ユーザプログラム実行によるアクセスはリードのみ可能です。内蔵ROM領域へのライトを行うとブレイクします。ただし、[コンフィグレーションプロパティ]ダイアログボックスにて、内蔵ROM領域ライト許可を有効にした場合は、リード/ライト可能になります。

エミュレータ機能（メモリウィンドウ、ロードなど）によるアクセスは、常にリード/ライト可能です。

内蔵ROM領域のアクセスはリード時1ステート、ライト時6ステートとなります。

2. フラッシュメモリ

E6000Hエミュレータは、フラッシュメモリをサポートしていません。

3. 内蔵ROM / 内蔵RAM容量

E6000Hエミュレータにおいては、内蔵ROM/内蔵RAM容量が製品チップと異なりますので、注意してください。なお、[ステータス]ウィンドウの表示は製品チップに合わせて表示しています。

表 3.6 内蔵 ROM/内蔵 RAM 容量

項目	エミュレータ	製品チップ
内蔵 ROM 容量	1MB (H'00000000 ~ H'0000FFFF)	製品チップのハードウェアマニュアルにてご確認ください。
内蔵 RAM 容量	16kB (H'FFFF0000 ~ H'FFFF3FFF)	
	16kB (H'FFFFC000 ~ H'FFFFFFF)	

(2) 内蔵I/O領域

内蔵 I/O 領域をアクセスするとエミュレータに搭載されている SH7046 内の内蔵 I/O をアクセスします。ユーザプログラムを、内蔵 I/O への書き込みやアクセスしたときにブレイクしたい場合はハードウェアブレイクまたは内蔵ブレイクを使用してください。

(3) 外部メモリ領域

SH7046 の外部メモリ領域には、エミュレータがサポートしている全メモリ属性を割り付けることができます。

3.5.2 低消費電力状態（スリープ、ソフトウェアスタンバイ、ハードウェアスタンバイ）

SH7046 には、低消費電力状態としてスリープモード、ソフトウェアスタンバイモード、ハードウェアスタンバイモードがあります。

（1）ハードウェアスタンバイモード

E6000H エミュレータでは、ユーザシステムからの_HSTBY 信号を SH7046 に入力していませんので、ハードウェアスタンバイモードはサポートしていません。

（2）スリープモード、ソフトウェアスタンバイモード

1. ブレークについて

スリープモード、ソフトウェアスタンバイモードは、通常の解除要因の他にブレーク条件の一致（強制ブレーク）によっても各状態が解除され、ブレークします。ブレーク後のユーザプログラム再開は、SLEEP命令の次の命令になります。

2. トレースについて

スリープモード、ソフトウェアスタンバイモードでのトレース情報は取得しません。

3. エミュレータ機能によるメモリアクセスについて

スリープモード、ソフトウェアスタンバイモードでのメモリ内容の表示と変更については、デバッグ編「5.3 メモリ内容の表示と変更方法について」を参照してください。

3.5.3 割り込み

実行およびステップ実行中の SH7046 の割り込みは、すべてユーザに開放しています。

エミュレーション停止中（ブレークモード）は、割り込み要因を保持し、エミュレーション復帰直後、割り込み処理に遷移します。

3.5.4 コントロール用入力信号（_RES、_BREQ、_WAIT）

SH7046 のコントロール用入力信号として _RES、_BREQ、_WAIT 信号があります。

_RES 信号は、実行によるエミュレーション中のみ有効です。（ステップ実行によるエミュレーション中、_RES 信号は無効になります。）

エミュレータのエミュレーション停止中（ブレーク中）では、_RES 信号は SH7046 に入力されません。

実行およびステップ実行中の _RES 信号は [コンフィグレーションプロパティ] ダイアログボックスにより入力を禁止することができます。

_BREQ、_WAIT 信号は、エミュレーション中/エミュレーション停止中（ブレーク中）にかかわらず常に有効です。

_BREQ、_WAIT 信号は、[コンフィグレーションプロパティ] ダイアログボックスにより入力を禁止することができます。

3.5.5 BSC（バスステートコントローラ）

ウェイトステートコントロールには、プログラムウェイトの挿入と WAIT 端子入力によるウェイトの挿入によるものがあります。

E6000H エミュレータでは、プログラムウェイトはエミュレーションメモリおよびユーザ外部メモリアクセス時に有効であるのに対し、WAIT 端子入力はユーザ外部メモリアクセス時のみ有効となります。

3.5.6 WDT（ウォッチドッグタイマ）

エミュレータのエミュレーション停止中（ブレーク中）は、WDT のタイマカウンタ (TCNT) のカウントアップを中断し、エミュレーション実行（ユーザモード）に復帰したときにカウントアップを再開します。

ブレークモード中にも、TCNT にクロックを供給するプリスケラは動作し続けます。ブレークモードに遷移する前後で、プリスケラの位相がずれる可能性があるため、オーバフローまでの時間がプリスケラのクロック周期で前後 1 サイクルずれることがあります。

3. ハードウェア仕様

3.5.7 A/D 変換器

A/D 変換器には、アナログ入力端子の他に AVcc、AVss、Avref、_ADTRG 端子がありますが、A/D 変換器は独立した電源で動作するため、AVcc（電源端子）はユーザシステム上で、必ず A/D 電源に接続してください。

- 【注】
1. A/D 変換器を使用しない場合は、AVcc を Vcc に接続してください。
 2. エバチップボードに実装している SH7046 とユーザシステムの間には、ユーザシステムインタフェースボード、プリント基板配線および保護回路等があります。このため、変換精度は SH7046 シリーズチップ、SH7144 シリーズチップよりも劣ります。A/D 変換器を使用したユーザシステムの最終評価の際は SH7046 シリーズチップ、SH7144 シリーズチップ（F-ZTAT マイコン）のご使用をお奨めします。

3.5.8 E6000H エミュレータの状態と内蔵モジュール

E6000H エミュレータの状態により、一部の内蔵モジュールは動作しません。以下に、E6000H エミュレータの状態と内蔵モジュールの関係を示します。

表 3.7 E6000H エミュレータの状態と内蔵モジュール

内蔵モジュール	エミュレーション停止中 (ブレイク中)	エミュレーション中 (実行、ステップ実行)
WDT (ウォッチドッグタイマ)	動作しない	動作する
MTU (マルチファンクションタイムパルスユニット)	動作する	動作する
MMT (モータマネジメントユニット)	動作する	動作する
CMT (コンベアマッチタイマ)	動作する	動作する
SCI (シリアルコミュニケーションインタフェース)	動作する	動作する
DTC (データトランスファコントローラ)	動作する* ¹	動作する
HCAN2	動作する	動作する
UBC (ユーザブレークコントローラ)	動作しない	動作する
AUD (アドバンストユーザデバッグ)	動作する	動作する
I/O ポート	動作する	動作する
A/D 変換器	動作する	動作する
H-UDI (ユーザデバッグインタフェース)	使用不可* ²	使用不可* ²

【注】 *1 DTC サイクル (ベクタリード、転送情報リード/ライト、データリード/ライト) 中にブレイクした場合、当該 DTC サイクルの終了まで DTC は動作を継続します。エミュレーション実行に復帰後、動作を再開します。

*2 E6000H エミュレータではサポートしていません。

3.5.9 端子機能

E6000H エミュレータにおいては、製品チップにない拡張機能をサポートしております。よって、SH7046/SH7047/SH7144/SH7145 において存在しない端子状態を表示することがあります。以下にエミュレータで状態を監視している端子と製品チップの相違点を示します。

表 3.8 端子と製品チップの相違点

端子名	エミュレータ	SH7046F	SH7047F	SH7144F	SH7145F
_RES	有り	有り	有り	有り	有り
NMI	有り	有り	有り	有り	有り
_IRQ0	有り	有り	有り	有り	有り
_IRQ1	有り	有り	有り	有り	有り
_IRQ2	有り	有り	有り	有り	有り
_IRQ3	有り	有り	有り	有り	有り
_IRQ4	有り	なし	なし	有り	有り
_IRQ5	有り	なし	なし	有り	有り
_IRQ6	有り	なし	なし	有り	有り
_IRQ7	有り	なし	なし	有り	有り
_WAIT	有り	なし	有り	有り	有り
_BREQ	有り	なし	有り	有り	有り
_HSTBY	有り	なし	有り	なし	なし

マルチプレクス端子である IRQ0～7、WAIT および BREQ 端子の状態をエミュレータで監視するには、[コンフィグレーションプロパティ] ダイアログボックスの [Pin Select Registers] ページにて対応するポートを正しく設定する必要があります。

3.5.10 エミュレータのレジスタ初期値相違点

- エミュレータは、システム起動時およびコマンドでSH7046をパワーオンリセットする場合、汎用レジスタおよびコントロールレジスタの一部を初期化していますので注意してください。

表 3.9 SH7046 とエミュレータのレジスタ初期値相違点

レジスタ名	エミュレータ		SH7046 (パワーオンリセット)
	起動時	リセット (Reset CPU)	
PC	パワーオンリセットベクタの PC 値	パワーオンリセットベクタの PC 値	パワーオンリセットベクタの PC 値
R0～R14	H'00000000	リセット前の値	不定
R15 (SP)	パワーオンリセットベクタの SP 値	パワーオンリセットベクタの SP 値	パワーオンリセットベクタの SP 値
SR	H'000000F0	H'000000F0	H'00000XFX* ¹
GBR	H'00000000	リセット前の値	不定
VBR	H'00000000	H'00000000	H'00000000
MACH	H'00000000	リセット前の値	不定
MACL	H'00000000	リセット前の値	不定
PR	H'00000000	リセット前の値	不定

【注】 X は不定の値

4. 故障解析（故障症状調査書）

4.1 故障解析

E6000H エミュレータ用診断プログラムによる故障解析の手順について示します。

4.1.1 診断プログラムを実行するためのシステムセットアップ

診断プログラムを実行するためには、以下に示す機器が必要です。なお、本テストプログラムの実行時はユーザシステムインタフェースボードおよびユーザシステムを接続しないでください。

- E6000Hエミュレータ（HS7046EPH60H）
- PC
- E6000 PCインタフェースボード（本補足説明書では、以下のいずれかを指します。PCのインタフェース仕様に合わせて以下のインタフェースボードのいずれか一製品をご用意ください。）
 - PCIバスインタフェースボード（HS6000EIC01H,HS6000EIC02H）
 - PCカードインタフェース（HS6000EIP01H）
 - LANアダプタ（HS6000ELN01H）
 - USBアダプタ（HS6000EIU01H,HS6000EIU02H）

1. PCにE6000PCインタフェースボードを挿入し、付属のPCインタフェースケーブルを接続してください。
2. PCインタフェースケーブルをE6000Hエミュレータ本体に接続してください。
3. E6000Hエミュレータ本体に、付属のAC電源ケーブルを接続してください。
4. PCを起動し、DOSプロンプト(Windows[®]98SE、Windows[®]Me)またはコマンドプロンプト(Windows NT[®]、Windows[®]2000、Windows[®]XP) のコマンド入力待ち状態にしてください。
その際、開いた状態が全画面表示でなかった場合、[Alt+Enter] キーを押し、全画面表示に切り替えてください。
また、元の状態に戻す場合も [Alt+Enter] キーを押すと、OSに関係なく、画面表示を切り替えることができます。

【注】 MS-DOS プロンプトにおいて、[Alt+Enter]の操作を行っても画面表示が切り替わらなかった場合は、下記のようにプロパティの[その他]ページの[Windows ショートカットキー]にある[Alt+Enter]のチェックボックスをチェックし、[更新]ボタンをクリックして更新してください。

4. 故障解析（故障症状調査書）

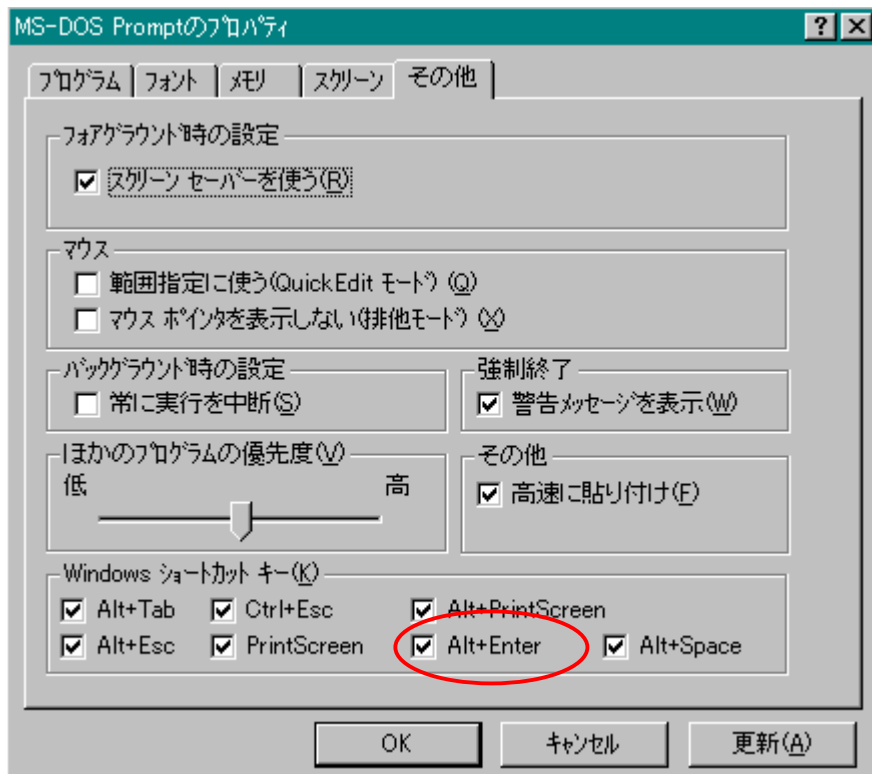


図 4.1 「その他」ページ

5. E6000Hエミュレータ本体の電源をオンにしてください。

【注】 診断プログラムを実行する場合には、必ず、E6000Hエミュレータの電源オンから始めてください。

診断プログラムにおいては、ハードウェアの初期状態のチェックを行っています。

よって、電源オン後、診断プログラムを実行する前に、High-performance Embedded Workshop の起動は行わないでください。

4.1.2 診断プログラムのテスト項目

本診断プログラムのテスト項目を表 4.1 に示します。

表 4.1 診断プログラムのテスト項目

テスト No.	テスト項目	テスト内容
1	Main Board Access	E6000H メインボード内レジスタテスト
2	Emulation Board Access	E6000H エミュレーションボード内レジスタテスト
3	Evaluation Board Access	E6000H エバチップボード内レジスタテスト
4	Basic Function	基本機能のテスト
5	GO to BREAK Time Measurement	実行時間測定機能のテスト
6	Emulation Monitor	エミュレーションモニタのテスト
7	G/A Break Function	G/A ブレーク機能のテスト
8	G/A Performance Analysis Function	G/A パフォーマンス測定機能のテスト
9	G/A Monitor Function	G/A モニタ機能のテスト
10	G/A Parallel RAM Monitor	G/A パラレル RAM モニタ機能のテスト
11	G/A Trace Function	G/A トレース機能のテスト
12	Combination	各機能の組合せテスト
13	Parallel Access	パラレルアクセス機能のテスト
14	PC Coverage	PC カバレッジ機能のテスト

4.1.3 診断プログラムによる故障解析

E6000H エミュレータに添付されている CD-R (HS7046EPH60SR) を PC の CD-ROM ドライブに挿入し、コマンドプロンプトでカレントディレクトリを <ドライブ>:\Diag フォルダに移動した後、使用している PC インタフェースボードの種類に従い、下記コマンドを入力すると直ちにテストプログラムが起動します。なお、診断プログラムの実行に際しては、High-performance Embedded Workshop がインストールされていることを前提としています。

(1) PCIバスインタフェースボード (HS6000EIC01H,HS6000EIC02H)

>TM7046 -PCI (RET)

(2) PCカードインタフェース (HS6000EIP01H)

>TM7046 -PCCD (RET)

(3) LANアダプタ (HS6000ELN01H)

>TM7046 -ELN (RET)

(4) USBアダプタ (HS6000EIU01H,HS6000EIU02H)

>TM7046 -USB (RET)

カレントディレクトリを <ドライブ>:\Diag フォルダに移動しない状態で ><ドライブ>:\Diag\TM7046 -PCI (RET) のように他のカレントディレクトリから診断プログラムを起動した場合は診断プログラムが正しく動作しません。必ず <ドライブ>:\Diag フォルダにカレントディレクトリを移動して診断プログラムを実行してください。

なお、>TM7046 -PCI -S (RET) のように、-S をコマンドラインに追加すると、No.1 から No.14 までのテストを繰り返し実行することができます。途中でテストを中断する場合は Q を入力してください。

- 【注】
1. <ドライブ>は CD-ROM ドライブのドライブ文字です。
 2. テストプログラム実行中は CD-ROM ドライブから CD-R を取り出さないでください。

4. 故障解析 (故障症状調査書)

テストが実行されているときに表示されるメッセージとテスト内容は次のようになります。テストは No.1 から No.14 までです (PCI インタフェースボード使用時にテスト時間は約 3 分です)。

```
E6000H SH7046 Emulator Tests Vx.x.xx.xxx
Loading driver .....OK (Use PCI)
Initializing driver .....OK
Searching for interface card .....OK
Checking emulator is connected .....OK
Emulator board information:
Main board ID:      H'0          Emulation board ID: H'002
Normal started at Wed Jan 22 09:43:43 2003
***** NORMAL TEST - Press 'Q' to stop *****
```

テストプログラムのスタートメッセージです。
x.x はバージョン番号です。
ホスト PC に PCI インタフェースボードが正しく接続されていることを示します。
ホスト PC と E6000H エミュレータが正しく接続されていることを示します。
E6000H エミュレータの ID 番号を示します。
診断プログラムの開始時間を示します。
(COUNT=0001)

1. Main Board Access

```
01) Registers Initial Value Check .....OK
02) Registers Write/Verify .....OK
03) DPRAM Address Decode Test .....SKIP
04) DPRAM Marching Test .....SKIP
05) Trace Memory Address Decode Test .....OK
06) Trace Memory Marching Test .....OK
07) G/A Registers Initial Value Check .....OK
08) G/A Registers Write/Verify .....OK
```

2. Emulation Board Access

```
01) Registers Initial Value Check .....OK
02) Registers Write/Verify .....OK
03) H-UDI Interface Registers Initial Value Check .....OK
04) H-UDI Interface Registers Write/Verify .....OK
05) MAPR Interface Registers Initial Value Check .....OK
06) MAPR Interface Registers Write/Verify .....OK
07) PRALR Interface Registers Initial Value Check .....OK
08) PRALR Interface Registers Write/Verify .....OK
```

3. Evaluation Board Access

```
01) Registers Initial Value Check .....OK
02) Registers Write/Verify .....OK
03) H-UDI IDCODE Check .....OK
04) Firmware BOOT .....OK
05) Configuration Set .....OK
06) Emulation Memory CS0 Test .....OK
07) Emulation Memory CS1 Test .....OK
08) Emulation Memory CS2 Test .....OK
09) Emulation Memory CS3 Test .....OK
10) Emulation Memory Test .....OK
11) INROM Test .....OK
12) INRAM Test .....OK
```

4. Basic Function

```
01) GO to BREAK .....OK
02) RESET GO .....OK
03) STEP .....OK
04) KEYBREAK .....OK
05) BRKCONT .....OK
06) ERAM WRITE PROTECT Test .....OK
07) INROM WRITE PROTECT Test .....OK
08) ERAM GOD Test .....OK
```

5. GO to BREAK Time Measurement

01) Counter Test ModeOK
02) EMU 10MHz MPU 20MHz Sampling 20nsOK
03) EMU 10MHz MPU 40MHz Sampling 1.6uOK
04) EMU 10MHz MPU 40MHz Sampling 52uOK
05) EMU 10MHz MPU 40MHz Sampling MPUOK
06) EMU 10MHz MPU 40MHz Sampling MPU/2OK
07) EMU 10MHz MPU 40MHz Sampling MPU/4OK
08) EMU 10MHz MPU 40MHz Sampling MPU/8OK
09) EMU 10MHz MPU 20MHz Sampling 100nsOK

6. Emulation Monitor

01) TRESOK
02) ASEST3 - 0OK
03) VCC3VNGOK
04) PVCC-NGOK

7. G/A Break Function

01) Address ConditionOK
02) Data ConditionOK
03) Control Signal Condition (ASEDSHH/HL/LH/HL)OK
04) Function Code Condition (ASEAST3-0)OK
05) Control Signal Condition (ASEAA3-0)OK
06) Control Signal Condition (DMA)OK

8. G/A Performance Analysis Function

01) Time Measurement (20ns Sampling)OK
02) Pckcr Emclk Test 4MHzOK
03) Pckcr Emclk Test 12MHzOK
04) Pckcr Emclk Test 20MHzOK
05) Pckcr Emclk Test 25MHzOK
06) Pckcr Emclk Test 32MHzOK
07) Pckcr Emclk Test 50MHzOK

9. G/A Monitor Function

01) RUNOK
02) VCCDOWNOK
03) NOCLKOK
04) TIMEOUTOK

10. G/A Parallel RAM Monitor

01) PRAM Monitor (BYTE)OK
02) PRAM Monitor (WORD)OK
03) PRAM Monitor (LONG WORD)OK
04) PRAM Monitor (ERAM CS0(8Bit) LONG WORD)OK
05) PRAM Monitor (ERAM CS1(16Bit) WORD)OK
06) PRAM Monitor (ERAM CS2(32Bit) BYTE)OK
07) PRAM Monitor (ERAM CS3(32Bit) LONG WORD)OK

11. G/A Trace Function

01) Free TraceOK
02) Trace StopOK
03) Time StampOK
04) Trace SuppressOK
05) Range TraceOK
06) Sequential Trace StopOK
07) TBM Overflow TraceOK
08) Timeout Trace StopOK
09) Subroutine TraceOK
10) INROM TraceOK

12. Combination

01) B to A Time Measurement (FPGA counter)OK
02) B to A Time Measurement (G/A counter)OK
03) D to C Time Measurement (G/A counter)OK

4. 故障解析 (故障症状調査書)

13. Parallel Access

```
01) INROM Parallel Read Access (Byte/Long) .....OK
02) INROM Parallel Write Access (Byte/Long) .....OK
03) ERAM Parallel Read Access (Byte/Long) .....OK
04) ERAM Parallel Write Access (Byte/Long) .....OK
05) ERAM Parallel Write Access (Byte/Word/Long) .....OK
06) ERAM Parallel Read Access (Byte/Word/Long) .....OK
07) INROM Parallel Write Access (Byte/Word/Long) .....OK
08) INROM Parallel Read Access (Byte/Word/Long) .....OK
```

14. PC Coverage

```
01) PC Coverage .....OK
```

Normal stopped at Wed Jan 22 09:46:55 2003

Tests run for 0H:3M:12S

Summary:

Tests performed 1 time(s).

1. Main Board Access	: 0 Error(s)
2. Emulation Board Access	: 0 Error(s)
3. Evaluation Board Access	: 0 Error(s)
4. Basic Function	: 0 Error(s)
5. GO to BREAK Time Measurement	: 0 Error(s)
6. Emulation Monitor	: 0 Error(s)
7. G/A Break Function	: 0 Error(s)
8. G/A Performance Analysis Function	: 0 Error(s)
9. G/A Monitor Function	: 0 Error(s)
10. G/A Parallel RAM Monitor	: 0 Error(s)
11. G/A Trace Function	: 0 Error(s)
12. Combination	: 0 Error(s)
13. Parallel Access	: 0 Error(s)
14. PC Coverage	: 0 Error(s)

診断プログラムの終了時間を示します。
診断プログラムの実行時間を示します。
各テスト項目でのエラー発生数の合計を示します。

4.2 故障症状調査書

貴社益々ご清栄のこととお喜び申し上げます。

この度、SH7046 E6000H エミュレータ（HS7046EPH60H）をご購入頂き、厚く御礼申し上げます。

さて、万一故障が発生したときには、お手数ですが次ページの故障症状調査書に症状をご記入の上、担当営業まで御連絡くださいますようお願い申し上げます。

故障症状調査書

ご購入営業担当 行

お客様ご芳名 会社名 _____
 担当者名 _____ 様
 TEL _____

調査項目	症 状
1 故障発生 年月日、時期	西暦 年 月 日 {システム立ち上げ時、システム動作時} *{ }内の該当時期を で囲んでください。
2 故障発生頻度	() {日、週、月}に()回発生 * ()内に該当数字を記入し、{ }内の該当時期を で囲んでください。
3 エラー発生時の システム構成	(1) エミュレータ側のシステム構成 ● E6000H エミュレータ (HS7046EPH60H) シリアル No. レビジョン (ケース裏面に表示しています：シリアル No.は数字4桁、レビジョンはそれに続くアルファベットです) ● ホストインタフェース * { }内の該当箇所を で囲んでください。 PCI インタフェースボード {HS6000EIC01H・HS6000EIC02H} PC カードインタフェース {HS6000EIP01H} LAN アダプタ {HS6000ELN01H} USB アダプタ {HS6000EIU01H・HS6000EIU02H} シリアル No. レビジョン (基板上に捺印表示しています) ● 付属 CD-R (HS7046EPH60SR) バージョン V (CD-R に V.x.xx Release xx と表示しています (x: 数字)) ● ご使用になっている PC メーカー名 型式 使用 OS {Windows®98SE、Windows® Me、WindowsNT®4.0、Windows®2000、Windows®XP} (2) 実機の接続 {YES、NO} (3) ユーザシステムインタフェースボードの有無 {有・無} 型名 * { }内の該当箇所を で囲んでください。
4 エラー発生時の 設定内容	(1) 動作モード：モード (2) ターゲットシステム電圧： V (3) 使用クロック： (貸出しクロック、Xtal 発振子、外部クロック入力 いずれかに) (4) 動作周波数： MHz

4. 故障解析（故障症状調査書）

調査項目	症 状
5 故障現象	
6 デバッグ時のエラー内容	
7 診断プログラムでのエラー内容	
8 High-performance Embedded Workshop がエミュレータと接続できない	エラーメッセージ内容

上記以外のエラーについては、下記に症状を記載いただくようお願いいたします。

デバッグ編

1. 概要

デバッグ編では、以下の内容を説明しています。

表1.1 デバッグ編について

章	タイトル	内 容
2	デバッグの準備をする	この章では、ワークスペースの作成から本エミュレータとの接続までを説明しています。
3	デバッグ	High-performance Embedded Workshopで共通な下記機能については、High-performance Embedded Workshop ユーザーズマニュアルを参照してください。 <ul style="list-style-type: none">・ デバッグの準備・ プログラムを表示する・ メモリ内容を参照 / 設定する・ メモリ内容を波形形式で表示する・ メモリ内容を画像形式で表示する・ 変数を参照 / 設定する・ I/Oレジスタを参照 / 設定する・ レジスタを参照 / 設定する・ プログラムを実行 / 停止、リセットする・ 関数呼び出し履歴を見る・ コマンドラインインタフェースのデバッグ・ Elf / Dwarf2のサポート・ ラベルを参照 / 設定する
4	チュートリアル	本製品ではチュートリアルプログラムを提供しています。この章ではチュートリアルプログラムを用いて本エミュレータの主な機能の使用方法を説明しています。
5	本製品固有のソフトウェア仕様と注意事項	この章では、本エミュレータに関するソフトウェア仕様および注意事項について説明しています。
6	エラーメッセージ	この章では、本エミュレータ使用時に発生するエラーメッセージの内容と対策を説明しています。

2. デバッグの準備をする

2.1 High-performance Embedded Workshop の起動方法

High-performance Embedded Workshop は以下の手順で起動します。

- (1) ホストコンピュータとE6000Hエミュレータを接続してください。
- (2) ユーザシステムインタフェースケーブルをご使用の場合は、E6000Hエミュレータのコネクタとユーザシステムインタフェースケーブルを接続します。ユーザシステムインタフェースケーブル未使用の場合は本手順は不要です。
E6000Hエミュレータの電源を入れてください。ユーザシステムをご使用の場合は、E6000Hエミュレータの電源を入れる前に、ユーザシステムの電源を入れてください。
- (3) [スタート]メニューの[プログラム]からHigh-performance Embedded Workshopを起動してください。
- (4) [ようこそ!]ダイアログボックスが表示されます。

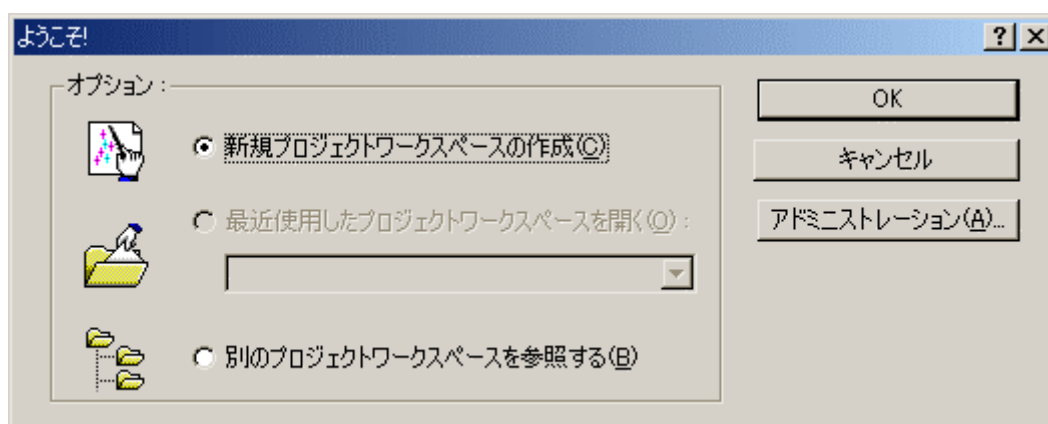


図2.1 ようこそ!ダイアログボックス

- [新規プロジェクトワークスペースの作成]ラジオボタン
ワークスペースを新規作成する場合に選択します。
- [最近使用したプロジェクトワークスペースを開く]ラジオボタン
既存のワークスペースを使用する場合に選択します。
開いたワークスペースの履歴が表示されます。
- [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
既存のワークスペースを使用する場合に選択します。
開いた履歴が残っていない場合に使用します。

ここでは、以下の3つの方法を説明します。

- [新規プロジェクトワークスペースの作成] – ツールチェインを使用しない場合
- [新規プロジェクトワークスペースの作成] – ツールチェインを使用する場合
- [別のプロジェクトワークスペースを参照する]

ツールチェインを使用する場合と使用しない場合では新規プロジェクトワークスペースの作成手順が異なります。本製品には、ツールチェインは含まれていません。ツールチェインは H8S, H8/300 シリーズ C/C++コンパイラパッケージまたは SuperH RISC engine C/C++コンパイラパッケージがインストールされている環境にて使用することができます。

ツールチェインを使用した新規プロジェクトワークスペースの作成についての詳細は、H8S, H8/300 シリーズ C/C++コンパイラパッケージまたは SuperH RISC engine C/C++コンパイラパッケージ付属のマニュアルを参照してください。

2 デバッグの準備をする

2.1.1 新規にワークスペースを作成する場合（ツールチェーン未使用）

- (1) [ようこそ!]ダイアログボックスで、[新規プロジェクトワークスペースの作成]ラジオボタンを選択し、[OK]ボタンをクリックしてください。

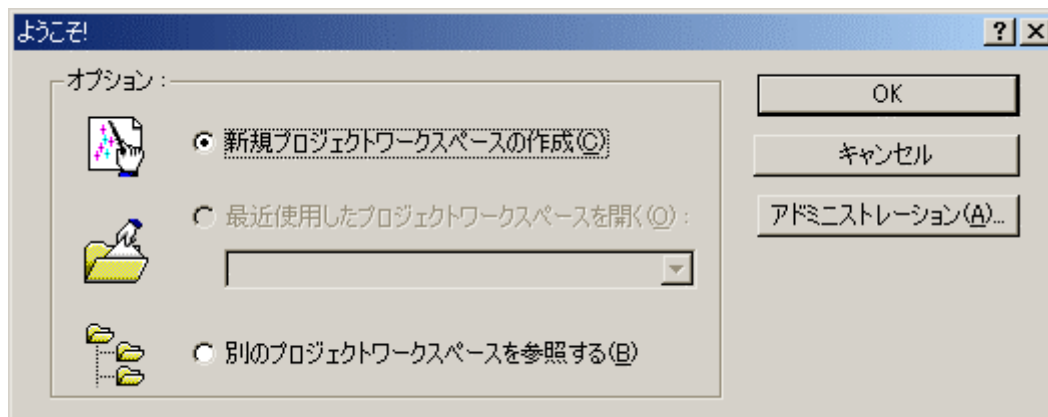


図2.2 ようこそ!ダイアログボックス

- (2) 新規プロジェクトワークスペースの作成を開始します。
以下の画面が開きます。

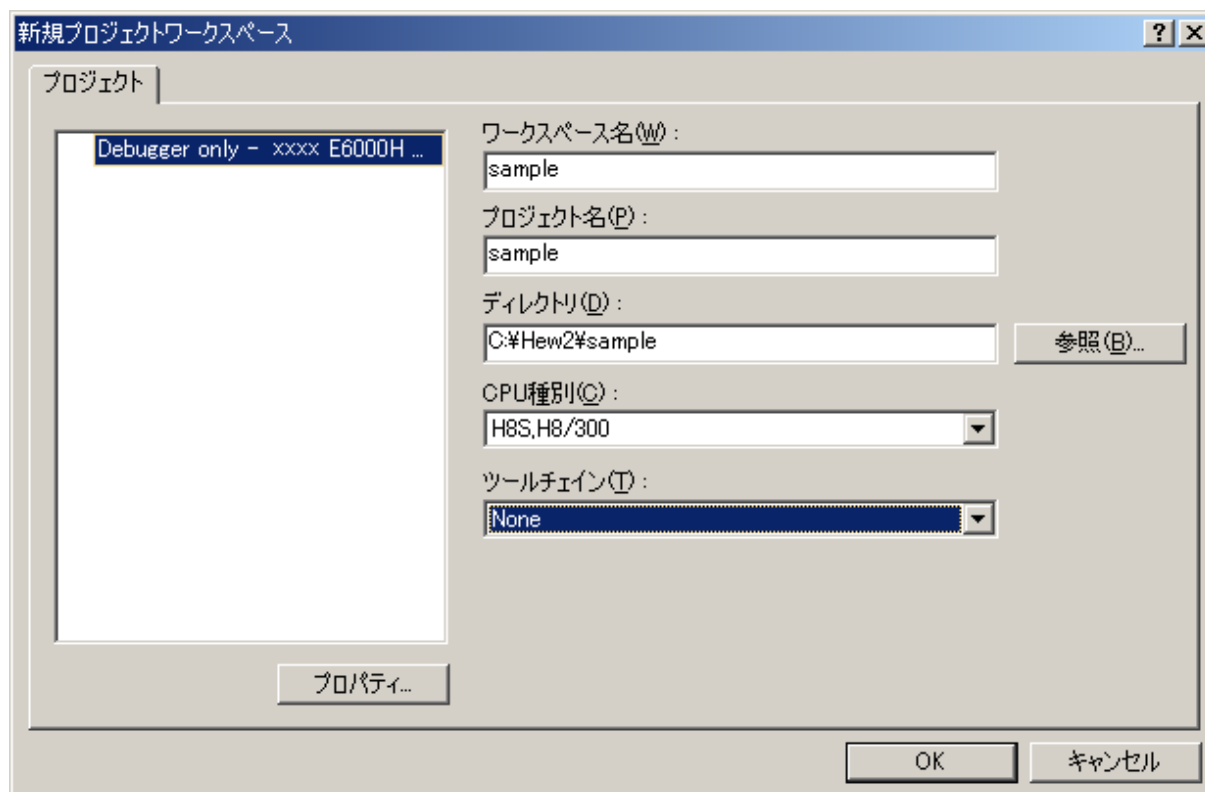


図2.3 新規プロジェクトワークスペースダイアログボックス

- [ワークスペース名]エディットボックス
新規作成するワークスペース名を入力してください。
- [プロジェクト名]エディットボックス
プロジェクト名を入力してください。ワークスペース名と同じでよければ、入力する必要はありません。
- [ディレクトリ]エディットボックス
ワークスペースを作成するディレクトリを入力してください。[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することもできます。
- [CPU 種別] ドロップダウンリストボックス
該当するCPUファミリを選択してください。

その他のリストボックスはツールチェイン設定用です。ツールチェインをインストールしていない場合は固定情報が表示されます。

[OK]ボタンをクリックしてください。

- (3) 次に、セッションファイルのターゲットプラットフォームを選択します。
以下の画面が表示されます。

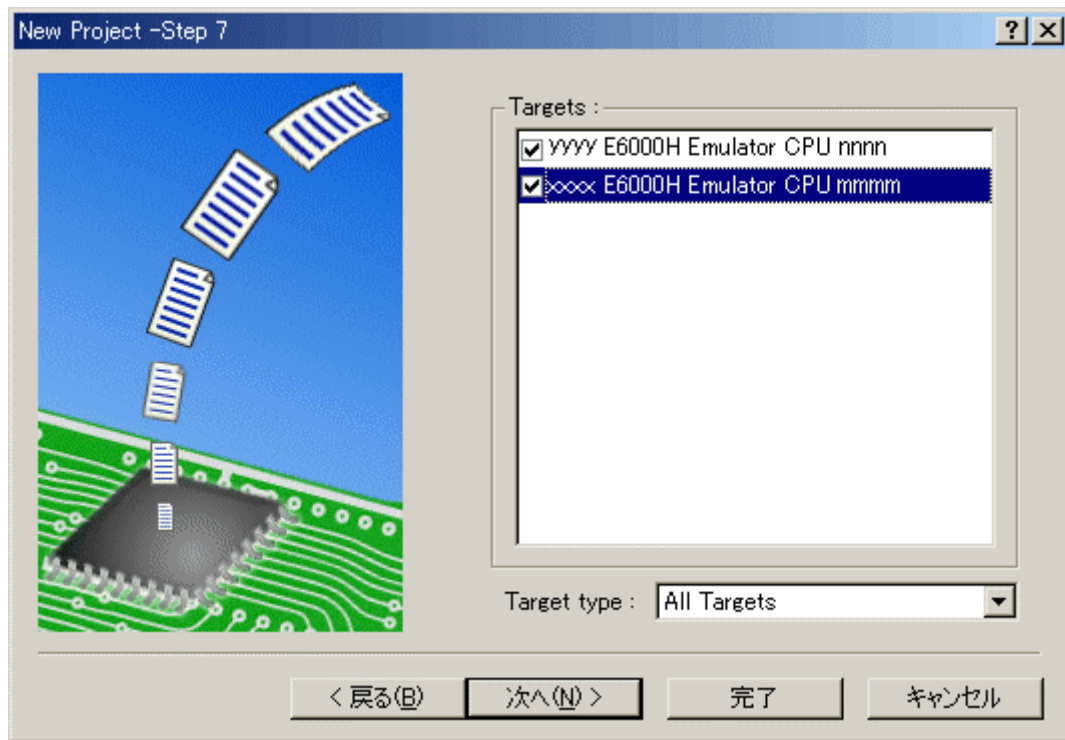


図2.4 デバッガターゲットダイアログボックス

ここでは、セッションファイルのターゲットプラットフォームを選択します。使用するターゲットプラットフォームにチェックし、[次へ]ボタンをクリックしてください。セッションファイルについての詳細は「2.4 デバッグセッション」を参照してください。

2 デバッグの準備をする

(4) 次に、コンフィグレーションファイル名を設定します。

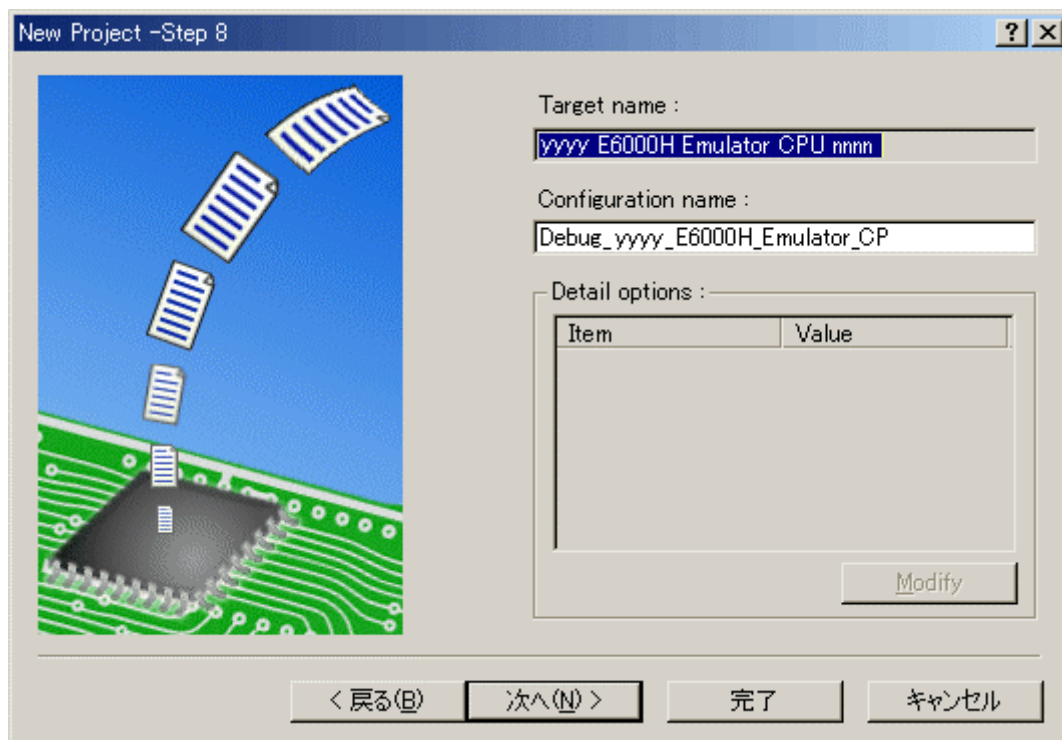


図2.5 デバッガオプションダイアログボックス

[デバッガターゲット]ダイアログボックス(図2.4)で複数のターゲットプラットフォームをチェックした場合には、[次へ]ボタンをクリックし、チェックしたターゲットプラットフォーム毎にコンフィグレーションファイル名を設定します。コンフィグレーションファイル名の設定が完了したら、E6000H エミュレータに関する設定は終了です。

[完了]ボタンをクリックすると[概要]ダイアログボックスが表示されます。[OK]ボタンをクリックすると、High-performance Embedded Workshop が起動します。

(5) 起動後、自動的に E6000H エミュレータが接続されます。

接続が完了すると、[アウトプット]ウィンドウの[Debug]タブに「Connected」と表示されます。

2.1.2 新規にワークスペースを作成する場合(ツールチェーン使用)

(1)[ようこそ!]ダイアログボックスで、[新規プロジェクトワークスペースの作成]ラジオボタンを選択し、[OK]ボタンをクリックしてください。

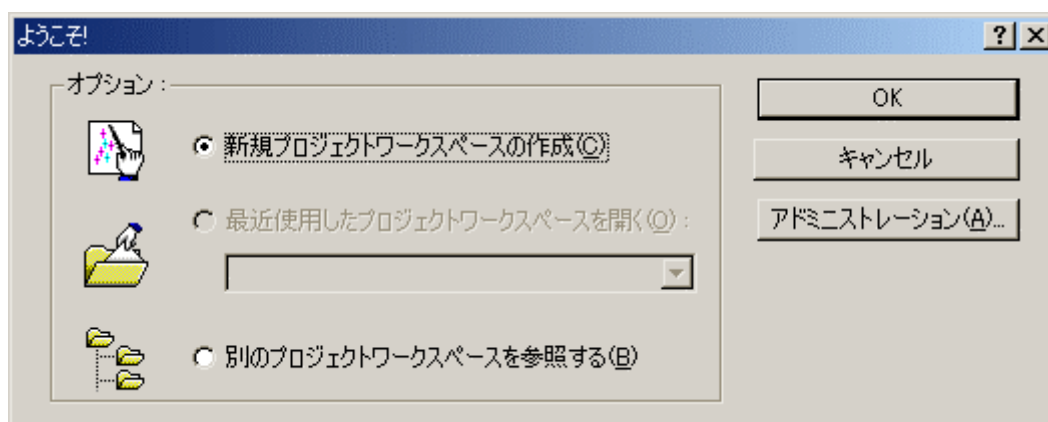


図2.6 ようこそ!ダイアログボックス

- (2) 新規プロジェクトワークスペースを作成します。
以下の画面が開きます。

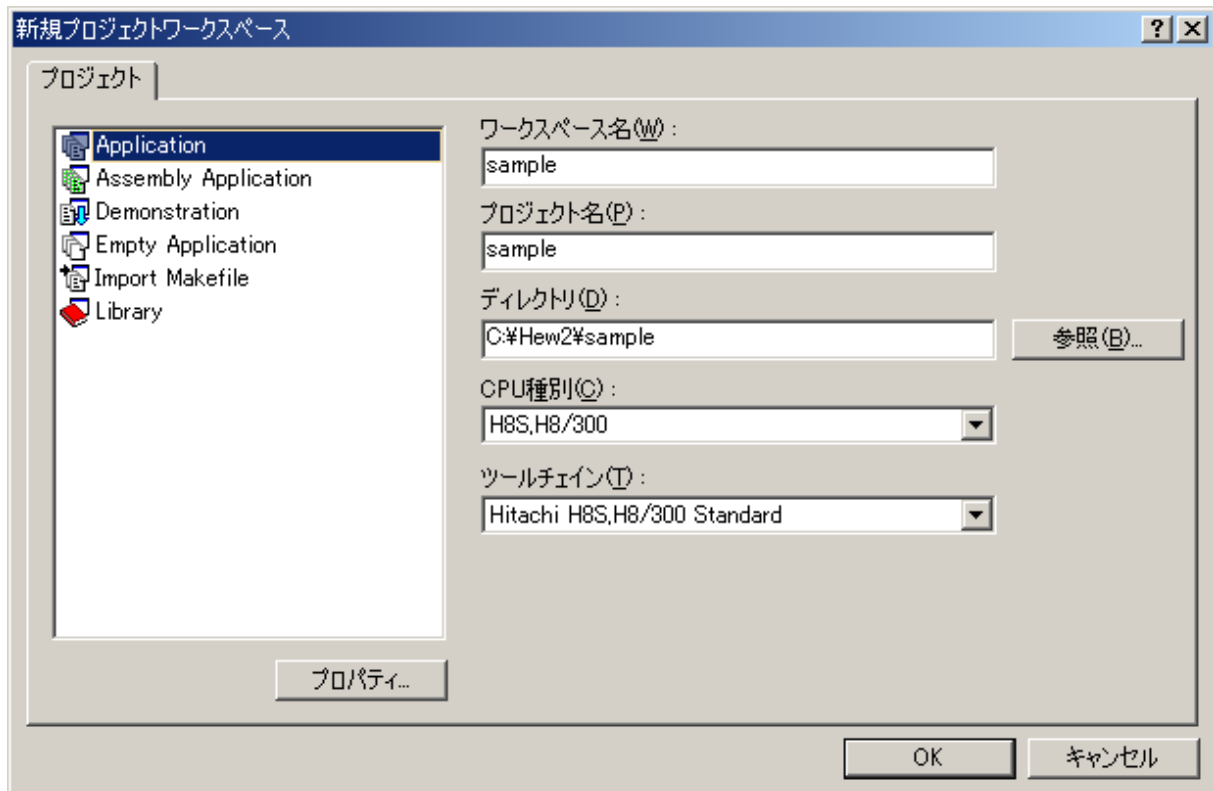


図2.7 新規プロジェクトワークスペースダイアログボックス

- [ワークスペース名]エディットボックス
新規作成するワークスペース名を入力してください。
- [プロジェクト名]エディットボックス
プロジェクト名を入力してください。ワークスペース名と同じであれば、入力する必要はありません。
- [ディレクトリ]エディットボックス
ワークスペースを作成するディレクトリを入力してください。[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することもできます。
- [CPU 種別]ドロップダウンリストボックス
該当するCPUファミリを選択してください。
- [ツールチェーン]ドロップダウンリストボックス
ツールチェーンをご使用になる場合、該当するツールチェーン名を選択してください。使用しない場合、[None]を選択してください。
- [プロジェクトタイプ]リストボックス
使用したいプロジェクトタイプを選択してください。

【留意事項】

E6000H エミュレータの場合、[Demonstration]を選択した場合に以下の注意事項があります。
[Demonstration]は H8S, H8/300 コンパイラパッケージまたは SuperH RISC engine C/C++コンパイラパッケージ付属のシミュレータ用のプログラムです。生成されたソースファイルを使用する場合、ソースファイル中の"Printf 文"を削除してください。

2 デバッグの準備をする

- (3) 次に、ツールチェインの設定を行いますので、必要な設定を行ってください。
ツールチェインの設定が終了したら、以下の画面が表示されます。

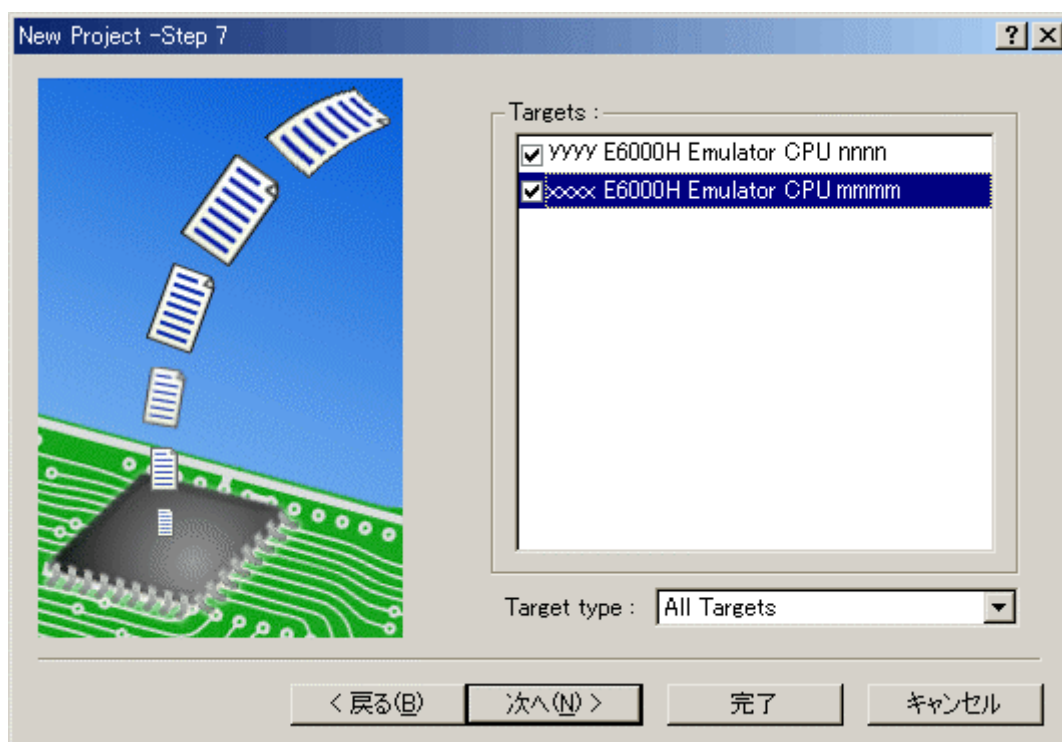


図2.8 新規プロジェクト-7/9-デバッグダイアログボックス

ここでは、セッションファイルのターゲットプラットフォームを選択します。使用するターゲットプラットフォームにチェックし、[次へ]ボタンを押してください。

(4) 次に、コンフィグレーションファイル名を設定します。

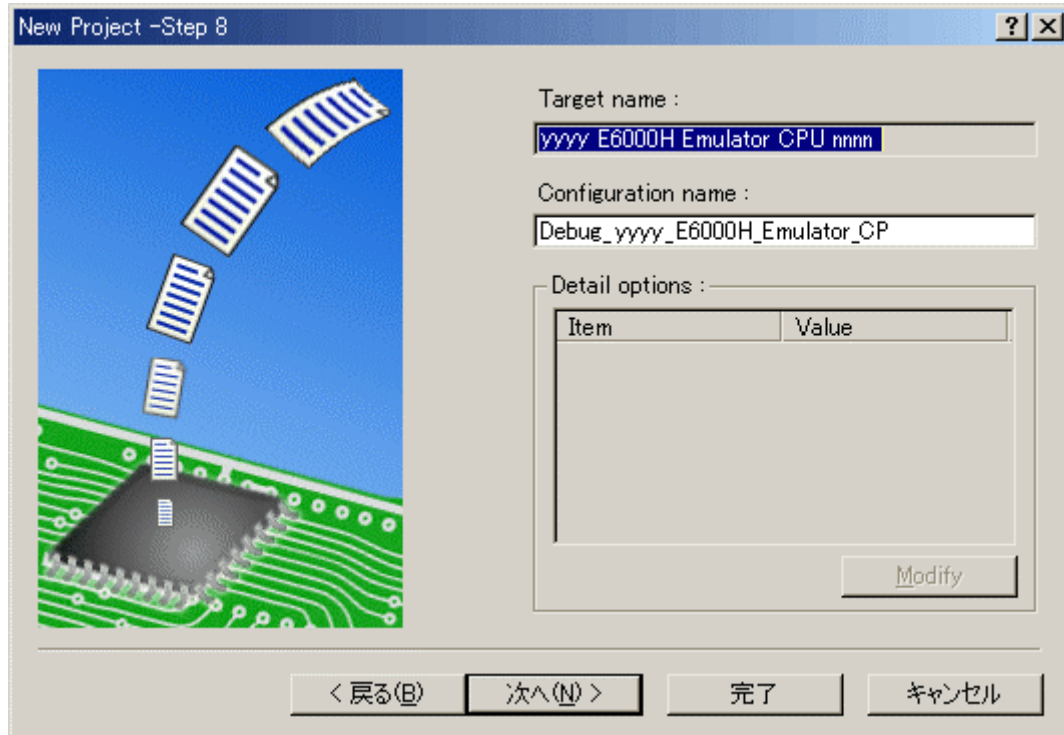


図2.9 新規プロジェクト-8/9-デバッガオプションダイアログボックス

[新規プロジェクト-7/9-デバッガ]ダイアログボックス(図2.8)で複数のターゲットプラットフォームをチェックした場合には、[次へ]ボタンをクリックし、チェックしたターゲットプラットフォーム毎にコンフィグレーションファイル名を設定します。コンフィグレーションファイル名の設定が完了したら、E6000H エミュレータに関する設定は終了です。

画面の指示に従い、新規ワークスペースの作成を完了してください。High-performance Embedded Workshop が起動します。

(5) 起動後、E6000H エミュレータを接続してください。

E6000H エミュレータは、High-performance Embedded Workshop 起動後すぐに接続する必要はありません。

E6000H エミュレータを接続する場合は、E6000H エミュレータ起動時の設定を行ってから接続する方法とE6000H エミュレータ起動時の設定を行わずに簡単に接続する方法があります。エミュレータの接続についての詳細は「2.2 エミュレータの接続」を参照してください。

2 デバッグの準備をする

2.1.3 既存のワークスペースを指定する場合

- (1) [ようこそ]ダイアログボックスで、[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、[OK]ボタンをクリックしてください。

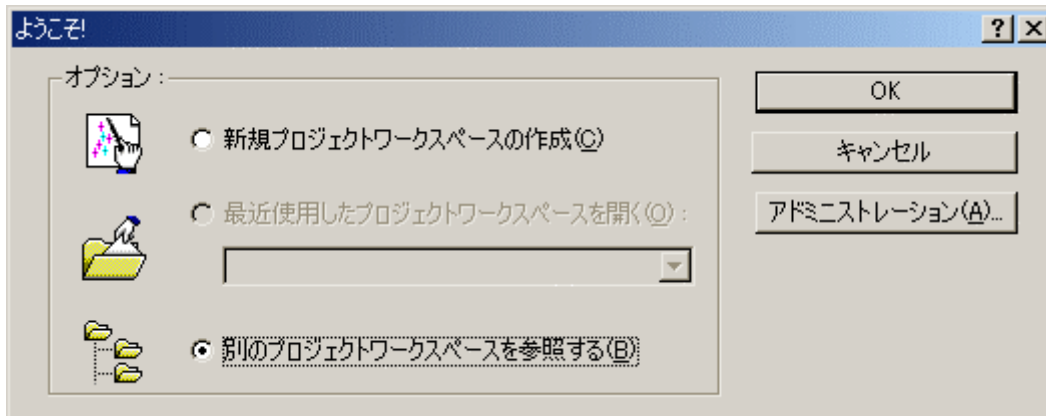


図2.10 ようこそ!ダイアログボックス

- (2) [ワークスペースを開く]ダイアログボックスが開きますので、ワークスペースが作成されているディレクトリを指定してください。

ディレクトリの指定後、ワークスペースファイル(拡張子 .hws)を選択し[開く]ボタンをクリックしてください。

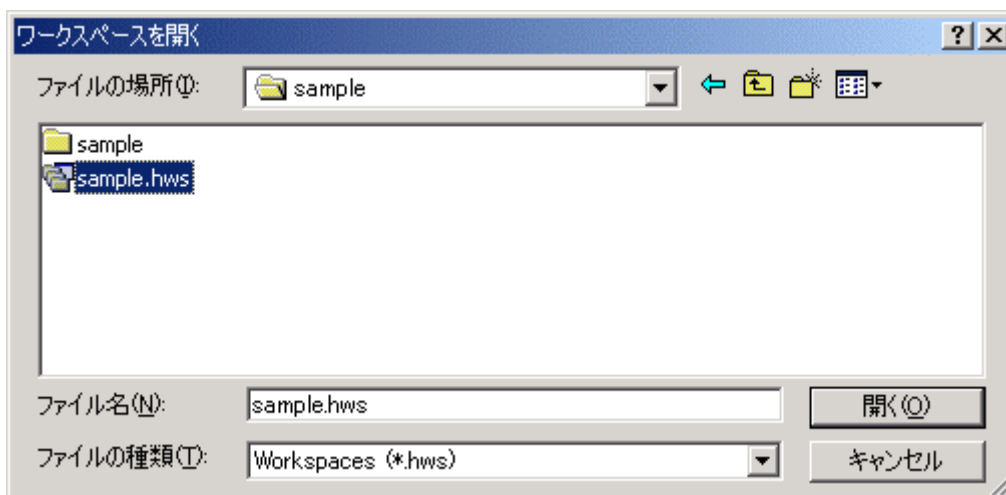


図2.11 ワークスペースを開くダイアログボックス

- (3) High-performance Embedded Workshop が起動され、指定したワークスペースの保存状態が復元されます。指定したワークスペースがエミュレータに接続された状態を保存していた場合には、エミュレータへの接続が自動で行われます。指定したワークスペースがエミュレータに接続されていない状態を保存していた場合に、エミュレータの接続を行う場合は、「2.2 エミュレータの接続」を参照してください。

2.2 エミュレータの接続

エミュレータの接続には、以下の方法があります。

(1) E6000H エミュレータ起動時の設定を行ってから接続する方法

[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開いてください。ここで、ダウンロードモジュールや起動時に自動的に実行するコマンドチェーンなどを登録することができます。

[デバッグの設定]ダイアログボックスの設定終了後、ダイアログボックスを閉じると、E6000H エミュレータが接続されます。

(2) E6000H エミュレータ起動時の設定を行わずに簡単に接続する方法

E6000H エミュレータを使用する設定があらかじめ登録されているセッションファイルに切り替えることにより、E6000H エミュレータを簡単に接続できます。

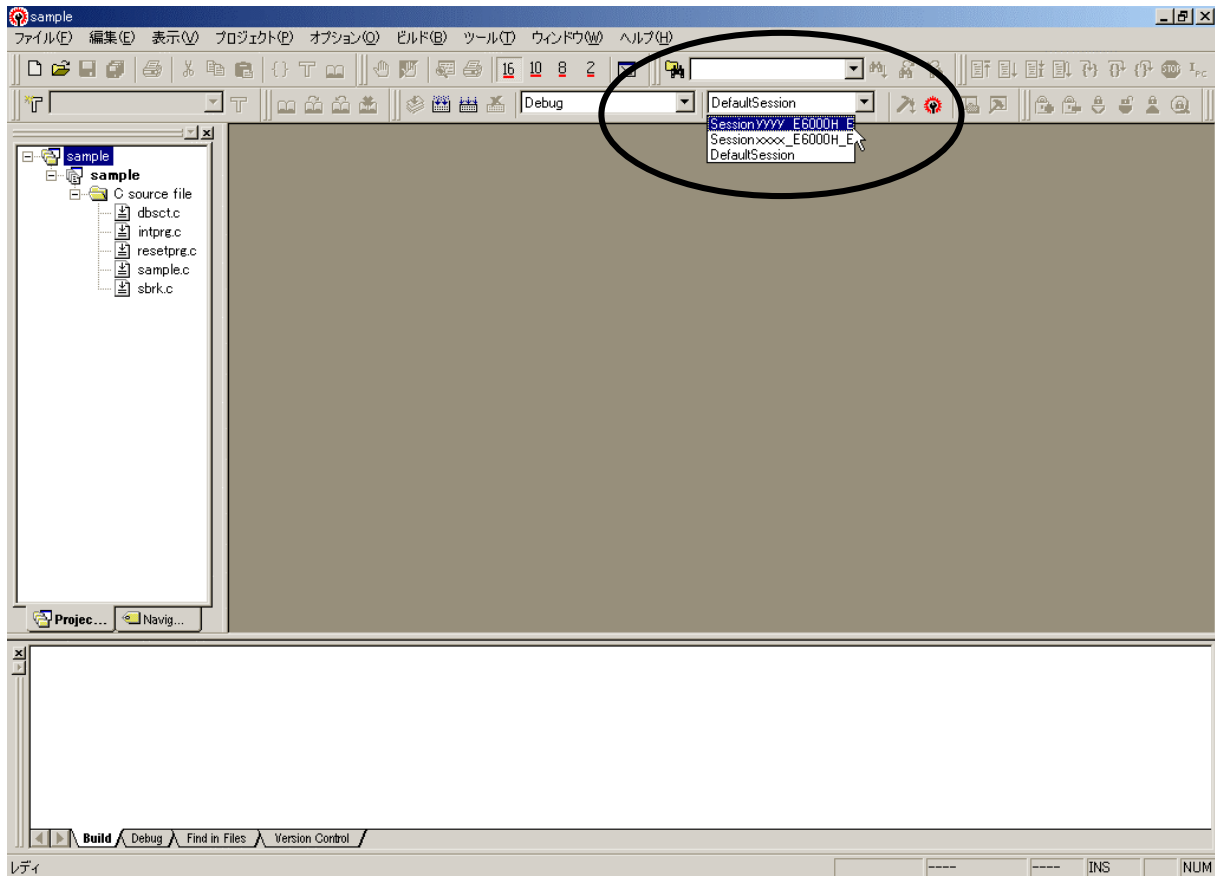


図2.12 セッションファイルの選択


上記図中の、丸印の中にあるリストボックスから、[新規プロジェクト-8/9-デバッガオプション]ダイアログボックス(図2.9)の[ターゲット名]テキストボックス内で設定されている文字列を含んだセッションファイル名を選択してください。

このセッションファイルには、E6000H エミュレータを使用する設定が登録されています。

[デバッグ->接続]を選択する事で、接続されます。

2.3 エミュレータの再接続

エミュレータ切断状態時に以下の方法で再接続を行うことができます。

[デバッグ -> 接続]を選択するか、接続ツールバーボタン  をクリックしてください。
エミュレータの接続が開始されます。


【注】 エミュレータの再接続を行う場合、あらかじめロードモジュールが High-performance Embedded Workshop に登録されている必要があります。

2.4 エミュレータの終了

エミュレータの終了方法は2通りあります。

- ・起動中のエミュレータの接続を解除する方法
- ・High-performance Embedded Workshop 自体を終了する方法

(1) 起動中のエミュレータの接続を解除する方法

[デバッグ -> 接続解除]を選択するか、接続解除ツールバーボタン  をクリックしてください。

(2) High-performance Embedded Workshop 自体を終了する方法

[ファイル->アプリケーションの終了]を選択してください。

メッセージボックスが表示されます。必要なら、[はい]ボタンをクリックし、セッションをセーブしてください。セーブ後、High-performance Embedded Workshop は終了します。不要なら、[いいえ]ボタンをクリックしてください。High-performance Embedded Workshop は終了します。

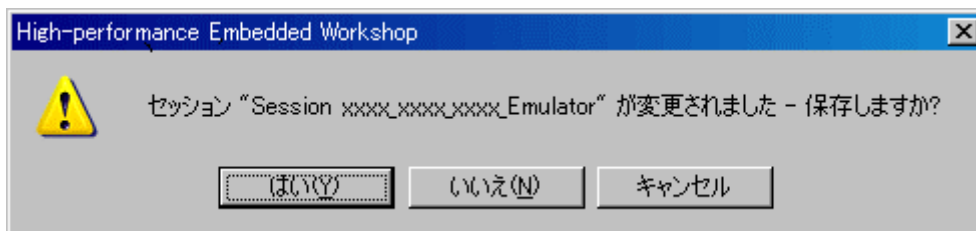


図2.13 セッション保存確認メッセージボックス


3. デバッグ

デバッグ操作と関連するウィンドウおよびダイアログボックスについて説明します。

3.1 エミュレーション環境を設定する

この節では、エミュレーションを行うための環境を設定する方法を説明します。デバッグを開始する前にエミュレーション環境を正しく設定する必要があります。

3.1.1 コンフィグレーションプロパティダイアログボックスを開く

[基本設定 -> エミュレータ -> システム...]を選択するか、[エミュレータシステム]ツールバーボタンをクリックすると、[コンフィグレーションプロパティ]ダイアログボックスが開きます。

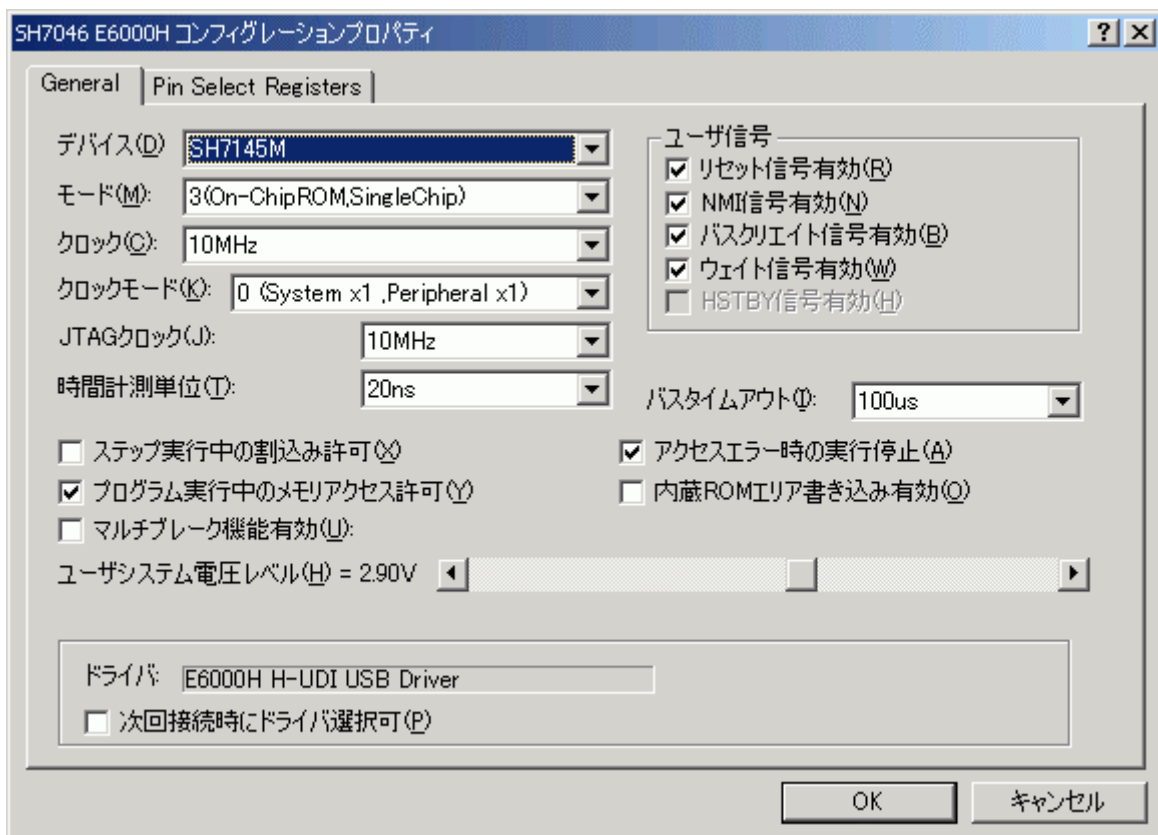


図3.1 コンフィグレーションプロパティダイアログボックス

3 デバッグ

[General]

[デバイス]	エミュレーションするターゲットマイコンを指定します。 詳細に関しては、それぞれのハードウェアマニュアルを参照してください。
[モード]	ターゲットマイコンの動作モードを指定します。
[クロック]	ターゲットマイコンの入力クロックを指定します。以下のいずれかから選択できます。 <ul style="list-style-type: none">• Emulator Clock (xMHz) E6000H 内蔵のクロック (X=4、6、8、10、12.5)• Target ユーザシステムのクロック信号• Xtal エバチップボード上の水晶発振子
[クロックモード]	ターゲットマイコンの動作クロックモードを設定します。
[JTAG クロック]	JTAG クロックを設定します。
[時間計測単位]	実行時間の測定およびパフォーマンスに使用するタイマの分解能を設定します。 分解能は以下のいずれかから選択できます。 実時間測定 52us、1.6us、20ns クロックカウンタ測定 CLOCK、CLOCK/2、CLOCK/4、CLOCK/8
[ステップ実行中の割り込み許可]	このチェックボックスをチェックすると、STEP 実行中の割り込みを受け付けます。
[アクセスエラー時の実行停止]	このチェックボックスをチェックすると、プログラムでアクセス禁止エリアにアクセス、または書き込み禁止エリアに書き込みが生じると、ブレーク (ユーザプログラム停止) します。
[プログラム実行中のメモリアクセス許可]	ユーザプログラム実行中にメモリアクセスを禁止するかどうかを指定します。
[内蔵ROMエリア書き込み有効]	このチェックボックスをチェックすると、ユーザプログラムによる内蔵 ROM エリアへのライトが可能になります。
[ユーザシステム電圧レベル]	ユーザシステム電圧レベルを指定します。 ユーザ PVcc が指定値よりも下がった場合、拡張モニタウィンドウの[User PVcc]に[Down]が表示されます。
[ユーザ信号]	このグループボックス内のチェックボックスをチェックすると、対応する制御端子の入力が有効になります。
[バスタイムアウト]	バスタイムアウト検出時間を指定します。100us、1.6ms、13ms、210ms が指定できます。
[マルチブレーク機能有効]	このチェックボックスをチェックすると、マルチブレーク機能が有効となります。マルチブレーク機能を有効にすると、トリガ出力とプローブ端子を使用して複数の E6000H を同時にブレークすることができます。
[ドライバ]	現在使用している E6000H ドライバを表示しています。
[次回接続時にドライバ選択可]	このチェックボックスをチェックすると、次回 E6000H 接続時にドライバを選択することができます。

【注】 クロックカウンタ設定の入力は、システムクロック ϕ になります。

E6000H エミュレータは、ターゲットマイコンとシリアル通信を行うことでデバッグ機能を実現しています。JTAG クロックはこのシリアル通信のための入力クロックとして使用します。JTAG クロックの設定をできるだけ大きな値に指定することで、ダウンロード性能およびメモリリード性能を向上することができます。ただし、JTAG クロックは、ターゲットマイコンの周辺モジュールクロック(P)より低くなるように設定してください。

3.1.2 ピンファンクションコントローラに関連する設定を行う

[コンフィグレーションプロパティ]ダイアログボックスの[Pin Select Registers]ページにて、ピンファンクションコントローラ（PFC）にて選択している端子を設定します。各信号に対応する端子を正しく設定することで、以下の機能が実現できます。

- ハードウェアブレイクおよびトレースの外部割り込み信号条件
- 実行状態表示の WAIT 信号の検出

ユーザプログラムが PFC に設定している端子を選択してください。

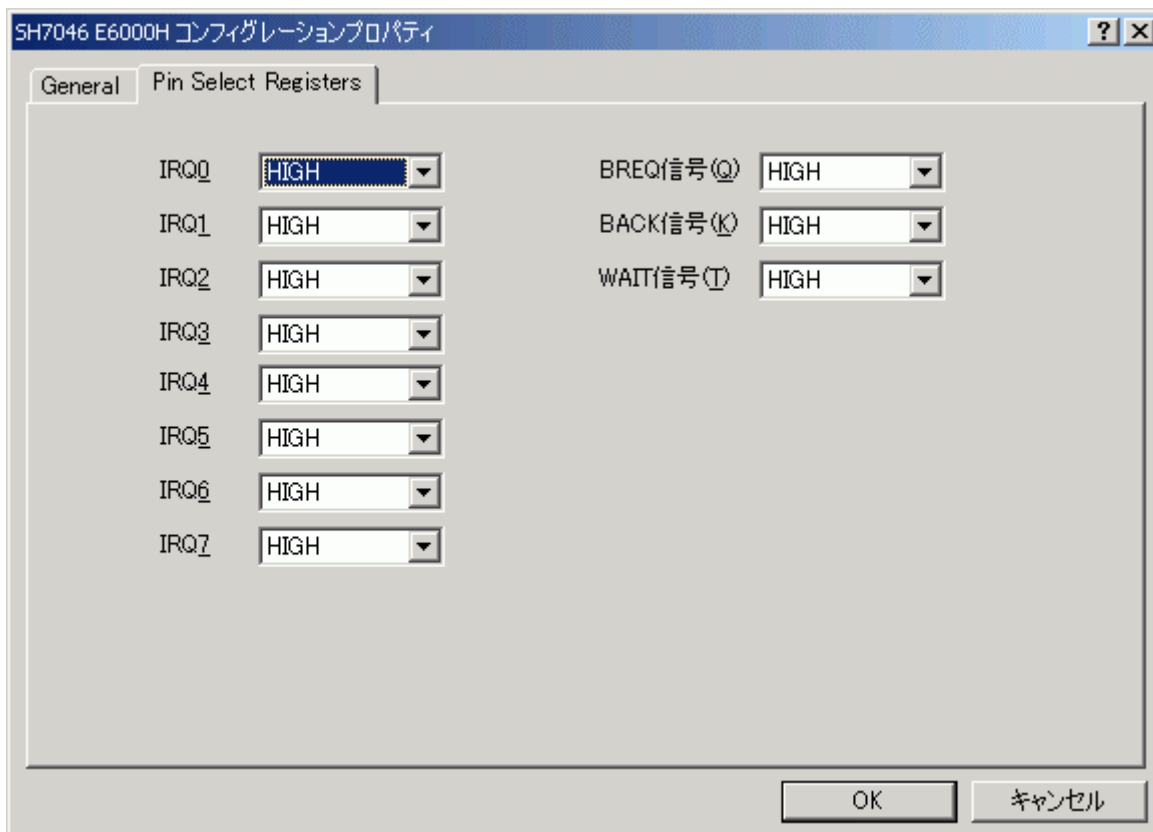


図3.2 Configuration Properties ダイアログボックス（Pin Select Registers ページ）

[Pin Select Registers]

- [IRQ0] IRQ0 信号に対応する端子（PA2, PB2, PD16, HIGH）を設定します。
- [IRQ1] IRQ1 信号に対応する端子（PA5, PB3, PD17, HIGH）を設定します。
- [IRQ2] IRQ2 信号に対応する端子（PA8, PB4, PD18, HIGH）を設定します。
- [IRQ3] IRQ3 信号に対応する端子（PA9, PB5, PD19, HIGH）を設定します。
- [IRQ4] IRQ4 信号に対応する端子（PB6, PD20, HIGH）を設定します。
- [IRQ5] IRQ5 信号に対応する端子（PB7, PD21, HIGH）を設定します。
- [IRQ6] IRQ6 信号に対応する端子（PB8, PD22, HIGH）を設定します。
- [IRQ7] IRQ7 信号に対応する端子（PB9, PD23, HIGH）を設定します。
- [BREQ 信号] BREQ 信号に対応する端子（PA13, PA18, PB7, HIGH）を設定します。
- [BACK 信号] BACK 信号に対応する端子（PA15, PA19, PB6, HIGH）を設定します。
- [WAIT 信号] WAIT 信号に対応する端子（PA7, PA17, PB8, PE17, HIGH）を設定します。

【注】 各設定の初期値はすべて HIGH になっています。HIGH 設定した場合には、信号の状態に関わらず High レベルとして画面に表示します。ただし、ターゲットマイコンの端子状態には影響を与えません。

3.1.3 接続するインタフェースを選択する

[コンフィグレーションプロパティ]ダイアログボックスの[次回接続時にドライバ選択可]項目をチェックすると、次回 E6000H 接続時にドライバを選択することができます。

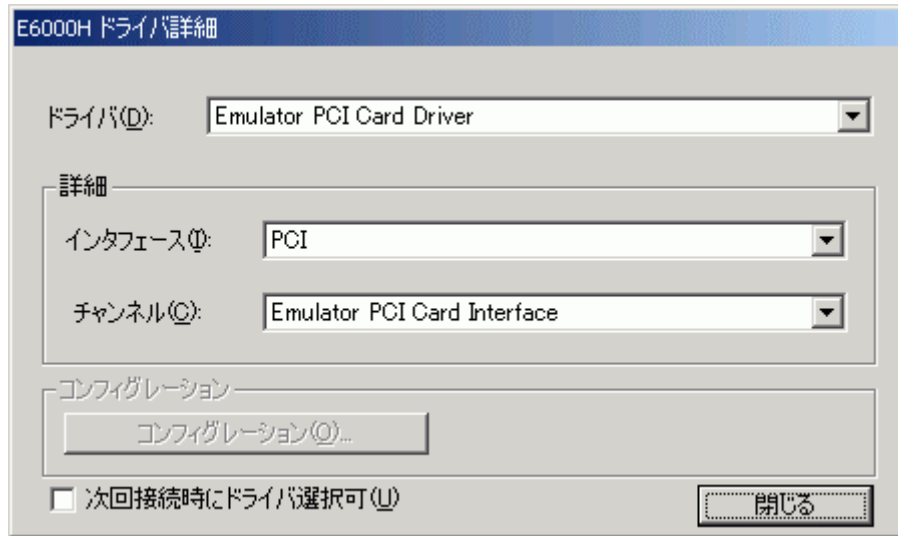



図3.3 E6000H ドライバ詳細ダイアログボックス

[ドライバ]	High-performance Embedded Workshop と E6000H エミュレータを接続するドライバを選択します。
[詳細]	接続するドライバの詳細を設定します。
[インタフェース]	選択したドライバがサポートしているインタフェースを選択します。 本エミュレータでは設定を変更する必要はありません。
[チャンネル]	選択したインタフェースのチャンネルを選択します。 本エミュレータでは設定を変更する必要はありません。
[コンフィグレーション]	ドライバの設定を行います。
[コンフィグレーション...]	ドライバがコンフィグレーションダイアログをサポートしている場合、設定ダイアログボックスを表示します。(本エミュレータでは使用できません。)
[次回接続時にドライバ選択可]	このチェックボックスをチェックすると、次回 E6000H 接続時にドライバを選択することができます。

3.1.4 Memory Mapping ダイアログボックスを開く

[基本設定 -> エミュレータ -> メモリリソース...]を選択するか、[メモリリソース]ツールバーボタンをクリックすると、[メモリマップ]ダイアログボックスが開きます。

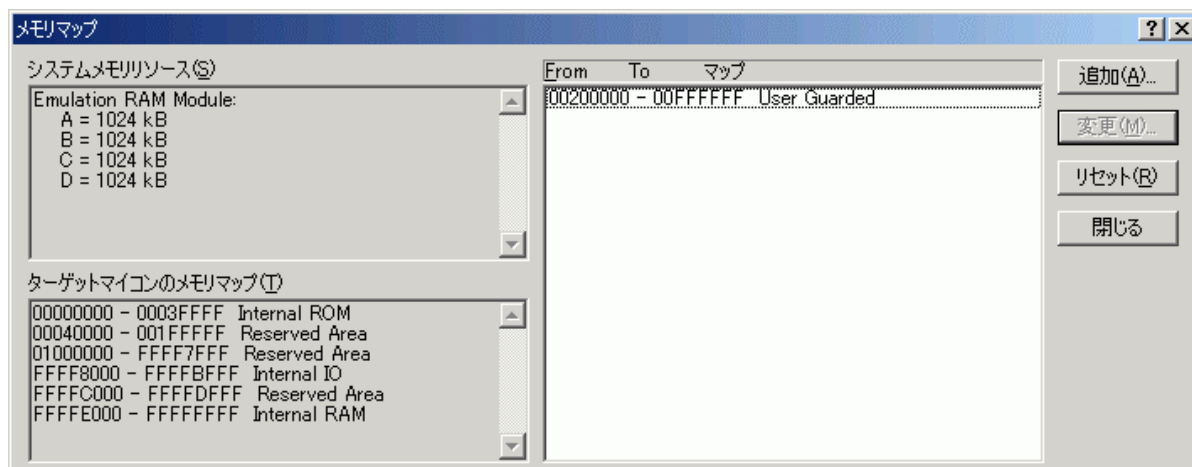


図3.4 メモリマップダイアログボックス

現在のメモリマップを表示しています。

- [追加...] メモリマップのアドレス範囲および属性を変更するため[メモリマップの編集]ダイアログボックスを開きます。
- [変更...] メモリマップのアドレス範囲および属性を変更するため[メモリマップの編集]ダイアログボックスを開きます。
- [リセット] メモリマップをデフォルト設定にリセットします。
- [閉じる] ダイアログボックスを閉じます。

ターゲットマイコンのメモリマップ情報は、[ステータス]ウィンドウの[Memory]シートに表示されます。

- 【注】** 詳細は「5.12 メモリマップ機能について」を参照してください。
[メモリマップ]ダイアログボックスの表示内容は製品により異なります。

3.1.5 メモリマップ設定を変更する

[メモリマップ]ダイアログボックスで[追加...]ボタンをクリックするか、または変更したいメモリマップ設定情報を選択し[変更...]ボタンをクリックすると[メモリマップの編集]ダイアログボックスが開きます。

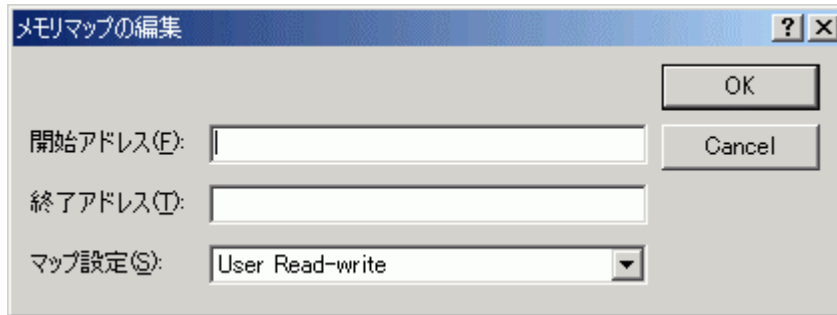


図3.5 メモリマップの編集ダイアログボックス

メモリマップのアドレス範囲および属性を変更します。

- | | |
|------------------------|---|
| [開始アドレス] | 範囲の開始アドレスを入力します。 |
| [終了アドレス] | 範囲の終了アドレスを入力します。 |
| [マップ設定] | マップ設定を選択します。
マップ設定は以下の属性があり、User (外部メモリ) および Emulator (エミュレーションメモリ) 属性については変更可能です。 |
| User Read-write | リードおよびライト可能な外部空間を設定します。 |
| User Read-only | 書き込み禁止の外部空間を設定します。 |
| User Guarded | アクセス禁止の外部空間を設定します。 |
| Emulator
Read-write | リードおよびライト可能なエミュレーションメモリを外部空間へ設定します。 |
| Emulator Read-only | 書き込み禁止のエミュレーションメモリを外部空間へ設定します。 |
| Emulator Guarded | アクセス禁止のエミュレーションメモリを外部空間へ設定します。 |
| No-access Area | メモリの存在しない空間です。 |

エミュレーションメモリは、ユーザシステム上のROMまたはフラッシュメモリの代わりとして使用することができます。

ユーザプログラムからアクセス禁止エリアにアクセス、または書き込み禁止エリアに書き込みが生じると、不正アクセスによりブレイクさせることができます。不正アクセスによるブレイクは[コンフィグレーションプロパティ]ダイアログボックスにて設定することができます。メモリウィンドウなどのデバッガ機能を使用したメモリのアクセスは、設定した属性に関係なく行うことができます。

【注】 シングルチップモード時は設定の変更ができません。

3.2 プログラムをダウンロードする

プログラムをダウンロードし、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見る方法を説明します。

【注】 ユーザプログラムがブレイクした時、[エディタ]ウィンドウは現在の PC がある位置を表示します。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（例えば、ELF/DWARF2 をベースにしたプロジェクトがもともとのパスから移動した場合）、ソースファイルブラウザダイアログボックスが表示されますので、手動でソースファイルを指定してください。

3.2.1 プログラムをダウンロードする

デバッグするロードモジュールをダウンロードします。

プログラムのダウンロードは[デバッグ -> ダウンロード]からロードモジュールを選択するか、[ワークスペース]ウィンドウの[Download modules]のロードモジュールを右クリックすると表示されるポップアップメニューより[ダウンロード]を選択します。

【注】 プログラムをダウンロードする場合、あらかじめロードモジュールとして High-performance Embedded Workshop に登録されている必要があります。

3.2.2 ソースコードを表示する

[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マウスの右ボタンをクリックして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]ウィンドウが表示されます。

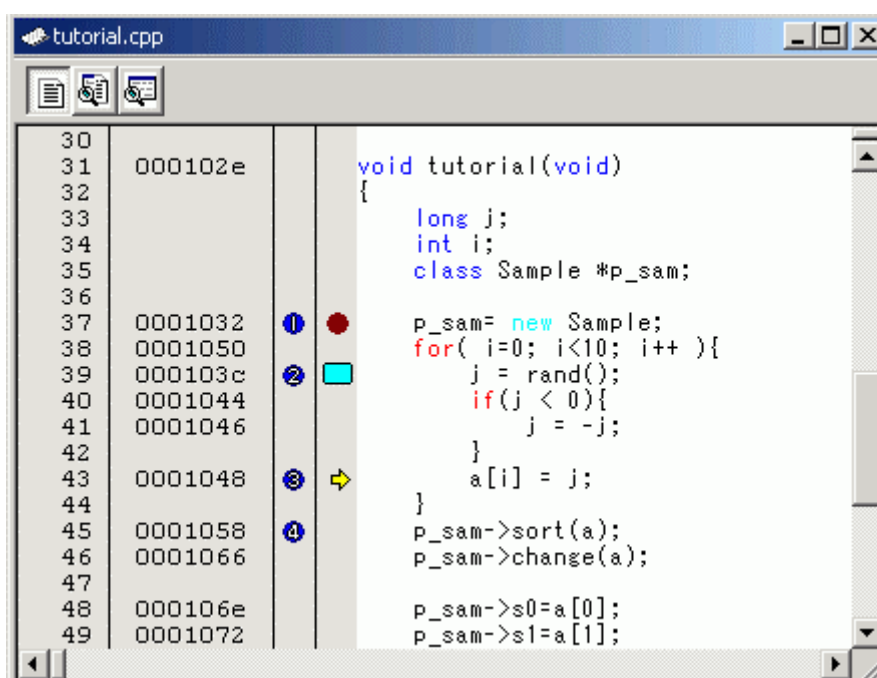


図3.6 [エディタ]ウィンドウ

本ウィンドウでは左端に行情報として下記を表示します。

- | | | |
|------|-------------------|----------------------|
| 1 列目 | (行番号カラム) | ソースファイルに対応する行番号 |
| 2 列目 | (ソースアドレスカラム) | ソース行に対応するアドレス情報 |
| 3 列目 | (オンチップブレイクカラム) | オンチップブレイク |
| 4 列目 | (S/W ブレイクポイントカラム) | PC、ブックマーク、ブレイクポイント情報 |

右側のソース表示画面をテキスト領域と呼びます。

行番号カラム

ソースファイルに対応する行番号を表示します。

3 デバッグ

ソースアドレスカラム

プログラムをダウンロードすると、ソースアドレスカラムに現在のソースファイルに対するアドレスを表示します。本機能は PC 値やブレークポイントをどこに設定するかを決めるときに便利です。

オンチップブレークカラム




オンチップブレークカラムには下記を表示します。

- ① オンチップブレークのチャンネル 1
- ② オンチップブレークのチャンネル 2
- ③ オンチップブレークのチャンネル 3
- ④ オンチップブレークのチャンネル 4
- Ⓟ オンチップブレークのリセットポイント

これらの設定は、ポップアップメニューからも可能です。

S/W ブレークポイントカラム

S/W ブレークポイントカラムには下記を表示します。

-  ブックマークを設定している
-  ソフトウェアブレークを設定している
-  PC 位置

⇒すべてのソースファイルでカラムをオフにするには

1. [エディタ]ウィンドウを右クリックしてください。または、[編集]メニューを選択してください。
2. [表示カラムの設定...]メニュー項目をクリックしてください。
3. [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ます。
4. チェックボックスは、そのカラムが有効が無効かを示します。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、一部のファイルではカラムが有効で、別のファイルでは無効であることを意味します。オフにしたいカラムのチェックボックスからチェックを外してください。
5. [OK]ボタンをクリックして、新しいカラム設定を有効にしてください。

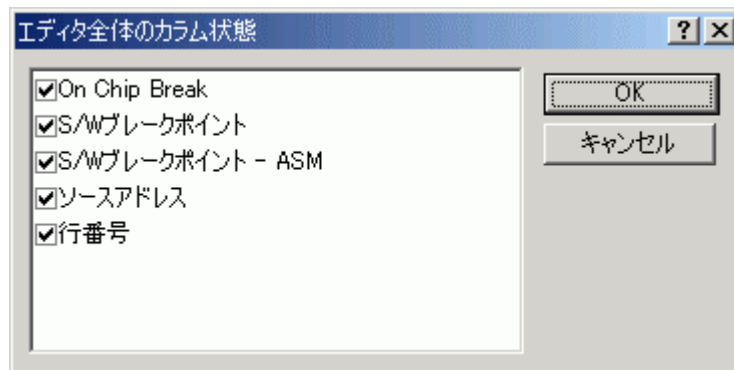



図3.7 エディタ全体のカラム状態ダイアログボックス

3.2.3 アセンブリ言語コードを表示する

ソースファイルが開いている場合、[エディタ]ウィンドウ上でマウスの右ボタンをクリックしてポップアップメニューから[逆アセンブリ表示]を選択すると、[逆アセンブリ]ウィンドウを表示します。[逆アセンブリ]ウィンドウの表示開始アドレスは、[エディタ]ウィンドウのカーソル位置に対応するアドレスとなります。

また、[エディタ]ウィンドウの[逆アセンブリ表示]ボタンを使用して、逆アセンブリコードを表示することもできます。

ソースファイルが存在しない場合、次のいずれかの方法で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” アクセラレータを使用する

この場合、[逆アセンブリ]ウィンドウは現在の PC の位置で開きます。アドレス、コード(オプション)、逆アセンブルされたニーモニック(可能なときはラベル付きで)を表示します。

オプションで、そのアドレスから開始するソースラインすべてを表示させる混合モード表示もサポートします。混合モードの逆アセンブルを表示するには、[混合モードで表示]ボタンをクリックしてください。

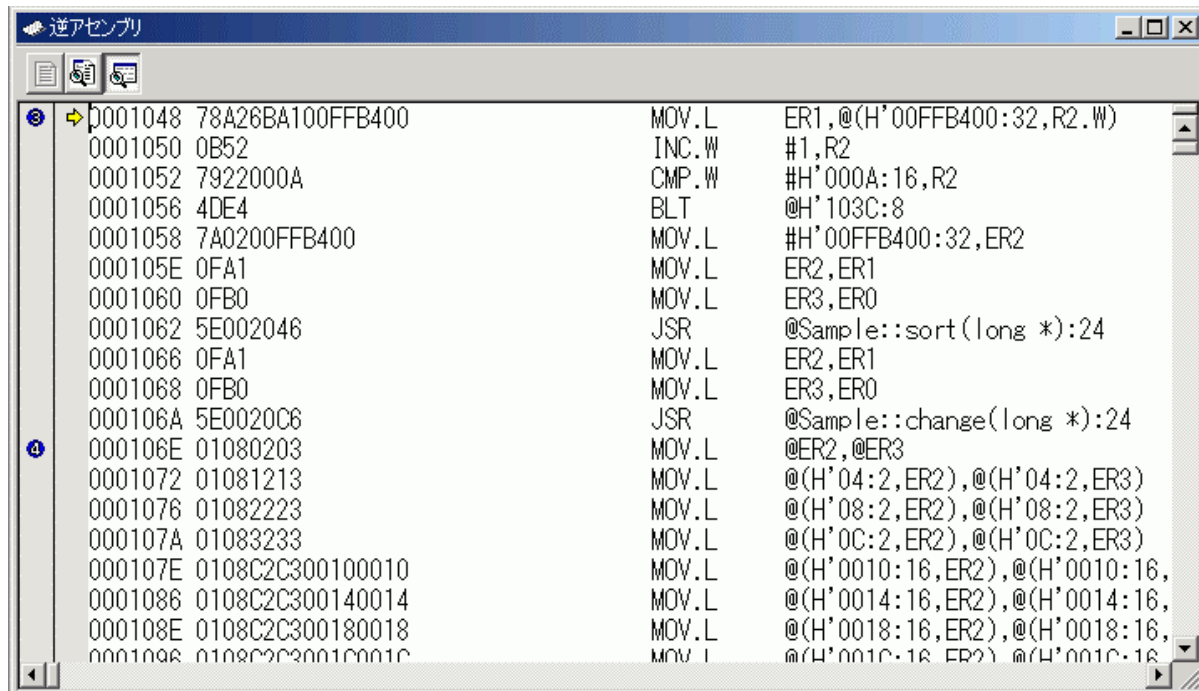


図3.8 逆アセンブリウィンドウ

本ウィンドウでは左はじに行情報として下記を表示します。

1行目 (オンチップブレークカラム) オンチップブレーク

2行目 (S/W ブレークポイント-ASM カラム) PC、ブレークポイント情報

使用方法はソースコードの表示ウィンドウと同じです。

3.2.4 アセンブリ言語コードを修正する

[逆アセンブリ]ウィンドウで修正したい命令をダブルクリックするか、ポップアップメニューから[編集...]を選択すると、[アセンブル]ダイアログボックスが開き、アセンブリ言語コードを修正することができます。



図3-9 アセンブルダイアログボックス

アドレス、命令コード、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命令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します。メモリの内容を修正すると、[逆アセンブリ]ウィンドウおよび[アセンブル]ダイアログボックスでは、新しいアセンブリ言語コードを表示します。しかし、[エディタ]ウィンドウに表示しているソースファイルは変更しません。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。

3.2.5 特定のアドレスを見る

[逆アセンブリ]ウィンドウを使って作成したプログラムを表示している場合、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]ウィンドウでアドレスをダブルクリックするか、ポップアップメニューから[表示アドレス設定...]を選択すると、[アドレス指定]ダイアログボックスを表示します。

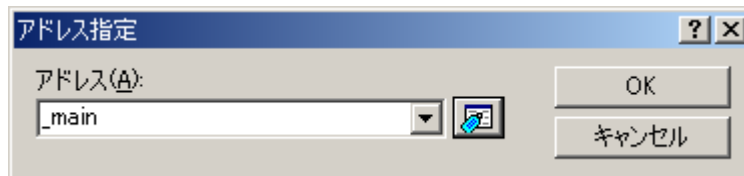


図3-10 アドレス設定ダイアログボックス

[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]ウィンドウを更新して入力したアドレス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。


3.2.6 現在のプログラムカウンタアドレスを見る

アドレスまたは値を入力できる場所では、式も入力することができます。先頭に"#"文字をつけたレジスタ名を入力すると、そのレジスタ内容を式の値として使用します。

[アドレス指定]ダイアログボックスで"#pc"という式を入力すると、[エディタ]または[逆アセンブリ]ウィンドウには、現在のPCアドレスを表示します。例えば、"#PC+0x100"といったPCレジスタおよびオフセットの式を入力することにより現在のPCのオフセットを表示することができます。

3.3 現在の状態を表示する

デバッグプラットフォームの現在の状態を知るには[ステータス]ウィンドウを表示します。

[ステータス]ウィンドウを開くには、[表示 -> CPU -> ステータス]を選択するか、[ステータスの表示]ツールバーボタンをクリックします。

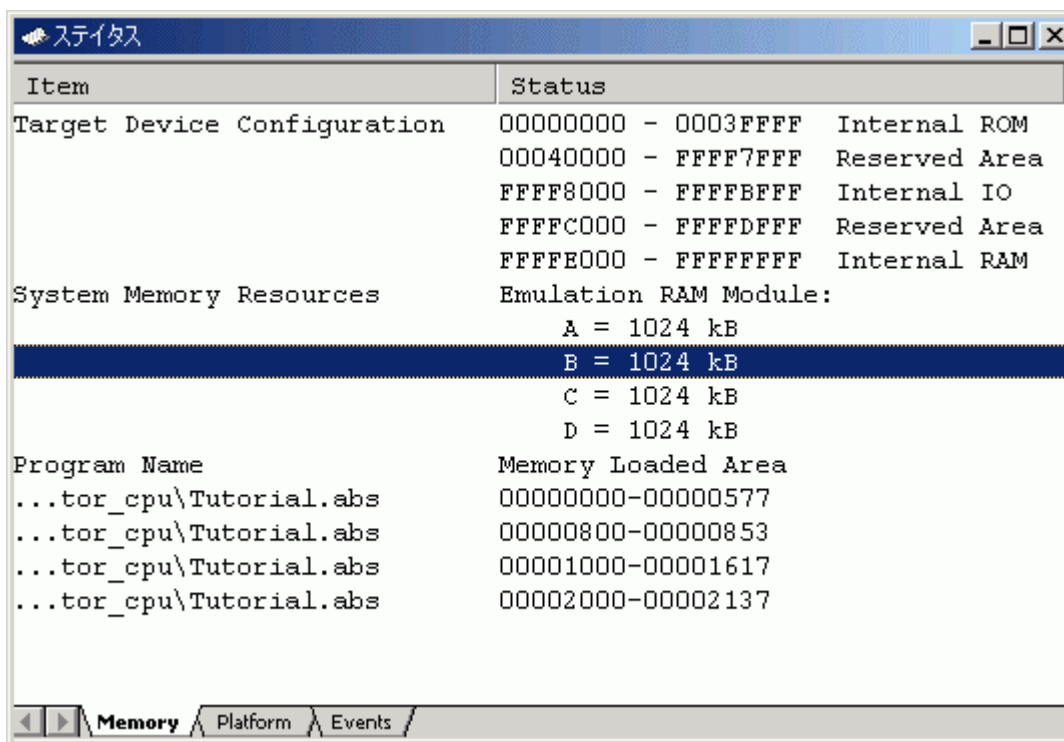


図3.11 ステータスウィンドウ

[ステータス]ウィンドウには、以下の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表示します。
- [Platform]シート
CPU種別および動作モードなど、エミュレーション環境情報を表示します。
- [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表示します。


【注】 本ウィンドウに表示する項目はご使用のエミュレータにより異なります。
詳細につきましては、オンラインヘルプをご参照ください。

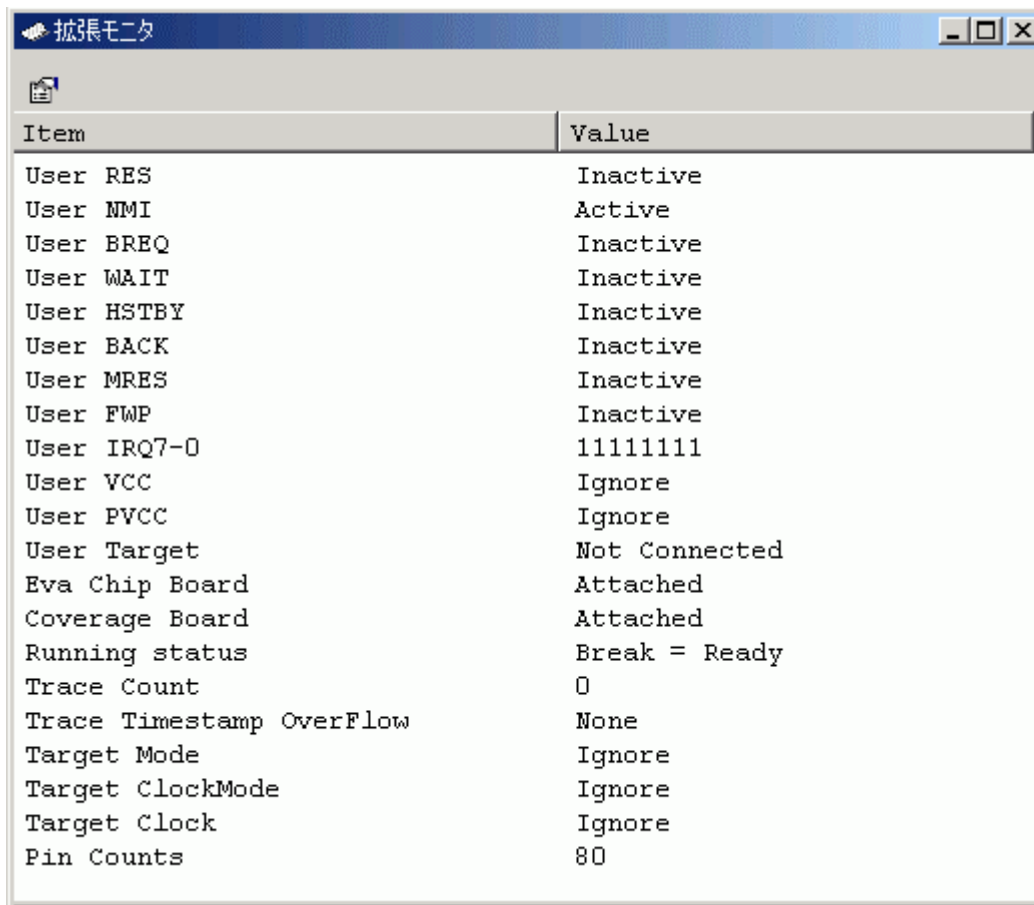
3.4 エミュレータの情報を定期的に読み出し表示する

ユーザプログラム実行中 / 停止中にかかわらず変化するエミュレータの情報を知るには[拡張モニタ]ウィンドウを使用します。

【注】 拡張モニタ機能は E6000H エミュレータのハードウェア回路によりユーザシステムやエミュレータ内部のターゲットマイコンから出力される信号をモニタするため、ユーザプログラムの実行に影響を与えることはありません。

3.4.1 [拡張モニタ]ウィンドウを開く

[拡張モニタ]ウィンドウを開くには、[表示 -> CPU -> 拡張モニタ]を選択するか、[拡張モニタ]ツールバーボタンをクリックします。表示項目の更新間隔は、ユーザプログラム実行中は約 1000ms、ブレイク中は約 5000ms です。



Item	Value
User RES	Inactive
User NMI	Active
User BREQ	Inactive
User WAIT	Inactive
User HSTBY	Inactive
User BACK	Inactive
User MRES	Inactive
User FWP	Inactive
User IRQ7-0	11111111
User VCC	Ignore
User PVCC	Ignore
User Target	Not Connected
Eva Chip Board	Attached
Coverage Board	Attached
Running status	Break = Ready
Trace Count	0
Trace Timestamp OverFlow	None
Target Mode	Ignore
Target ClockMode	Ignore
Target Clock	Ignore
Pin Counts	80

図3.12 拡張モニタウィンドウ

3.4.2 表示項目を選択する

ポップアップメニューから[プロパティ...]を選択すると[拡張モニタコンフィギュレーション]ダイアログボックスを表示します。

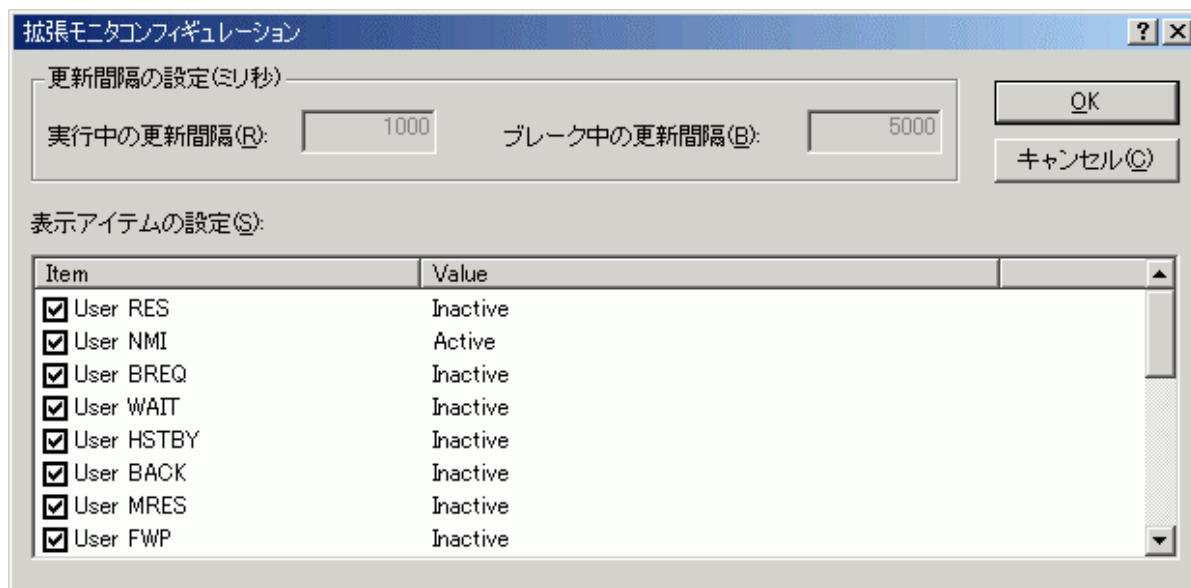


図3.13 拡張モニタコンフィギュレーションダイアログボックス

[拡張モニタ]ウィンドウに表示する各項を設定できます。

【注】 本ウィンドウに表示する項目はご使用のエミュレータにより異なります。詳細につきましては、オンラインヘルプをご参照ください。

3.5 リアルタイムにメモリ内容を表示する

ユーザプログラム実行中にメモリ内容をモニタするには[モニタ]ウィンドウを使用します。


モニタ機能はE6000Hエミュレータのバスモニタ回路によりターゲットマイコン内部のリード/ライト信号をトリガとしてアドレスバスおよびデータバスの値を保持し、該当するメモリの表示内容を更新するためリアルタイム性は損なわれません。

E6000Hエミュレータのバスモニタ回路に実装されたモニタチャンネル(8チャンネル)を使用し、最大8ポイントまたは合計256バイト設定できます。

各ポイントのモニタ範囲の一部または全部が重複する設定も可能です

【注】 1. ターゲットマイコンに内蔵のタイマカウンタなど、値の更新にターゲットマイコン内部のライト信号が発生しないエリアに対してはモニタできません。
2. メモリ内容の表示および変更方法は製品により異なります。また、ユーザプログラム実行中にメモリ内容の表示を更新するとリアルタイム性がなくなる場合があります。詳しくは「5.3 メモリ内容の表示と変更方法について」を参照してください。

3.5.1 モニタウィンドウを開く

[モニタ]ウィンドウを開くには、[表示 -> CPU -> モニタ -> モニタ設定...]を選択するか、[モニタ]ツールバーボタンをクリックして[Monitor Setting]ダイアログボックスを開きます。

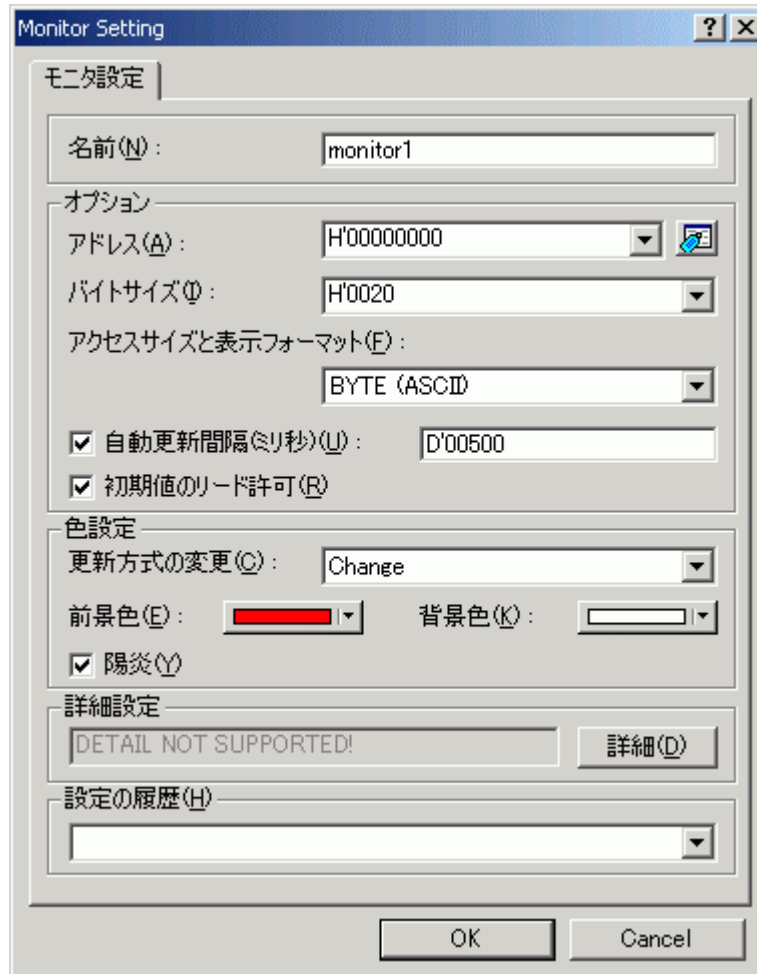


図3.14 Monitor Setting ダイアログボックス

[名前]	モニタウィンドウの名称を設定します。
[オプション]	モニタ条件を設定します。
[アドレス]	モニタを行う先頭アドレスを設定します。
[バイトサイズ]	モニタを行う範囲を設定します。
[アクセスサイズと表示フォーマット]	モニタウィンドウに表示するアクセスサイズを設定します。
[自動更新間隔(ミリ秒)]	モニタ取得間隔を設定します。(最小値は500ms)
[初期値のリード許可]	モニタウィンドウ OPEN 時に、モニタ表示エリアの値をリードします。
[色設定]	モニタの更新方法および色属性を設定します
[更新方法の変更]	モニタ中に変更があった値をどのように表示するかを設定します。 ([初期値のリード許可]選択時有効) No change: 色の変更は行いません。 Change: 色を変更します。 色は前景色オプション、背景色オプション、で設定します。 Gray: 値の変更のないデータを灰色表示します。 Appear: 値の変更があると表示します。変更なければ表示しません。
[前景色]	表示文字色を設定します。 ([Change]選択時有効)
[背景色]	背景色を設定します。 ([Change]選択時有効)
[陽炎]	チェックボックスにチェックがある場合、一定間隔更新のないデータの色を背景色オプションで設定した色に戻します。一定間隔とは、モニタ取得間隔の1回分です。 ([Change]、[Gray]、[Appear]選択時有効)
[詳細設定]	エミュレータ固有の項目を設定します。
[設定の履歴]	前回の設定内容呼び出します。

- 【注】** 1. 本エミュレータではモニタを行う先頭アドレスとして奇数アドレスは指定できません。
2. 前景色および背景色の設定はご使用のオペレーティングシステムにより使用できない場合があります。

設定完了後、[OK]ボタンをクリックすると[モニタ]ウィンドウが開きます。

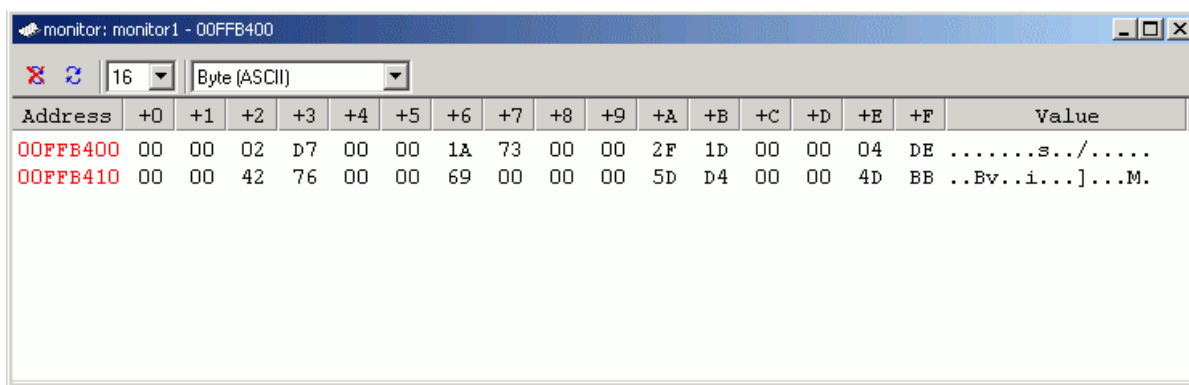


図3.15 モニタウィンドウ

ユーザプログラム実行中、自動更新間隔の設定値に応じて表示を更新します。

- 【注】** アドレス変更時またはメモリ内容変更時、データ内容が正しく表示されない場合は、ポップアップメニューより[最新の情報に更新]を選択してください。

3.5.2 モニタの設定内容を変更する

変更したい[モニタ]ウィンドウのポップアップメニューより[モニタ設定...]を選択すると、[Monitor Setting]ダイアログボックスが開き、設定内容を変更することができます。

また、ポップアップメニューの[色設定]メニューおよび[アクセスサイズと表示フォーマット]メニューより簡単に色設定およびアクセスサイズと表示フォーマットを変更できます。

3.5.3 モニタの更新を一時的に停止する

ユーザプログラム実行中、[モニタ]ウィンドウは設定した自動更新間隔にしたがって自動的に表示を更新します。

表示更新を停止させたい[モニタ]ウィンドウのポップアップメニューより[表示固定]を選択してください。

アドレス部の表示文字が黒色となり、表示更新を停止します。

再びポップアップメニューより[表示固定]を選択することにより停止状態は解除できます。

3.5.4 モニタ設定を削除する

削除したい[モニタ]ウィンドウのポップアップメニューより[閉じる]を選択すると、[モニタ]ウィンドウを閉じ、モニタ設定を削除します。

3.5.5 変数の内容をモニタする

任意の変数の値を参照するには、[ウォッチ]ウィンドウを使用します。

[ウォッチ]ウィンドウに登録した変数のアドレスが、モニタ機能で設定したモニタ範囲に存在する場合、該当する変数の値をモニタ機能により更新し表示することができます。

この機能によりリアルタイム性を損なわずに変数の内容を確認できます。

3.5.6 モニタウィンドウを非表示にする

モニタ機能を使用し、[ウォッチ]ウィンドウより変数の値をモニタする場合、[モニタ]ウィンドウを非表示にしておく画面を有効に活用できます。

現在設定しているモニタ情報は[表示 -> CPU -> モニタ]のサブメニューとしてリストされます。

モニタ設定リストは[モニタ]ウィンドウ名およびモニタ開始アドレスで構成されています。

リストの左側にチェックがある場合は該当の[モニタ]ウィンドウが表示されていることを示します。

モニタ設定リストより非表示にしたい[モニタ]ウィンドウ項目を選択すると、該当の[モニタ]ウィンドウが非表示となり、リストの左側にあったチェックマークが消えます。

非表示にした[モニタ]ウィンドウを再び表示するにはモニタ設定リストより非表示にした[モニタ]ウィンドウ項目を選択してください。

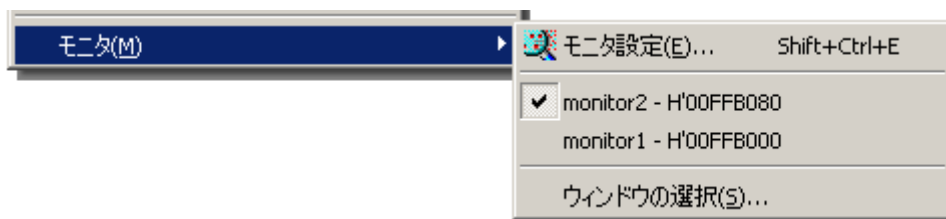


図3.16 モニタ設定リスト

3.5.7 モニタウィンドウを管理する

[表示 -> CPU -> モニタ -> ウィンドウの選択...]を選択すると表示される、[ウィンドウの選択]ダイアログボックスより、現在設定されているモニタ条件の確認、新規モニタ条件の追加、編集、削除などの操作を連続的に行うことができます。

また、現在設定されているモニタ条件を複数選択することにより、更新の一時停止、非表示、削除を一括して操作できます。

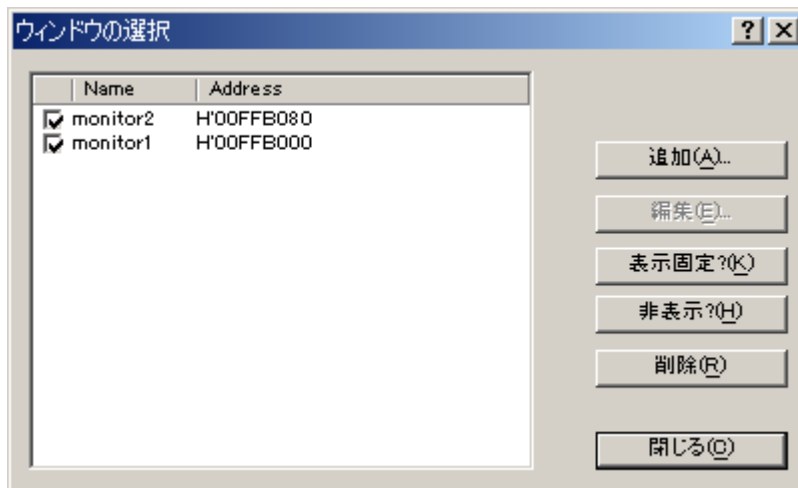


図3.17 ウィンドウの選択ダイアログボックス

[追加]	新規にモニタ条件を追加します。
[編集]	選択している[モニタ]ウィンドウの設定を変更します。 (複数選択時無効)
[表示固定 / 表示固定解除]	選択している[モニタ]ウィンドウの表示を自動更新または更新停止にします。
[表示 / 非表示]	選択している[モニタ]ウィンドウを表示または非表示にします。
[削除]	選択しているモニタ条件を削除します。
[閉じる]	本ダイアログボックスを閉じます。

3.6 変数の表示

本節では、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。

3.6.1 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することができます。

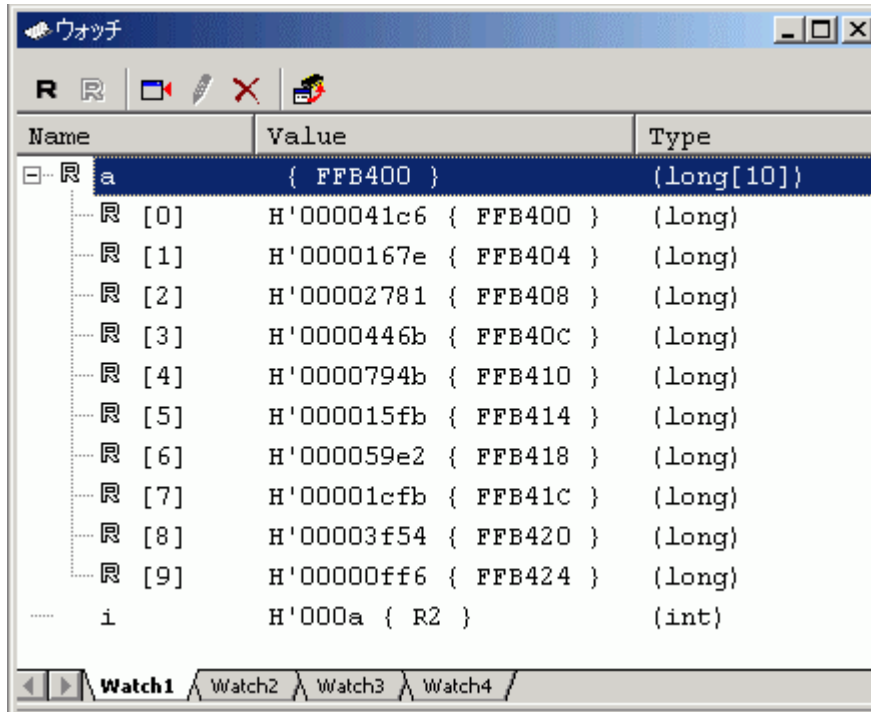


図3.18 ウォッチウィンドウ

R マークのある変数は、ユーザプログラム実行中に更新できることを示します。

R マークの色によって、ユーザプログラム実行中における値更新の実現手段が判別できます。

中抜き青色部	変数のアドレスがモニタ機能で設定されているモニタ範囲内で、モニタ機能を使用したデータリードが可能であることを示します。
青色部	モニタ機能を使用したデータリードによる値の更新を行うことを示します。
中抜き黒色部	変数のアドレスがモニタ機能で設定されているモニタ範囲外で、モニタ機能を使用したデータリードが不可能であることを示します。
黒色部	通常のデータリードにより値の更新を行うことを示します。


- 【注】
1. 本機能は変数ごと、また構造体であれば指定された構造体一括 / 要素ごとに設定できます。
 2. R マークの色は、モニタ設定を変更したときに変わります。
 3. レジスタに割り付けられている変数には設定することができません。
 4. メモリ内容の表示および変更方法は製品により異なります。また、ユーザプログラム実行中にメモリ内容の表示を更新するとリアルタイム性がなくなる場合があります。詳しくは「5.3 メモリ内容の表示と変更方法について」を参照してください。

3.7 イベントポイントを使用する

E6000H エミュレータでは、イベントポイントとして以下の3つのブレークをサポートしています。

ソフトウェアブレークポイント	指定アドレスの命令フェッチが行われた場合にユーザプログラムの実行を停止します。最大 255 ポイントまで設定できます。指定アドレスを専用命令で置き換えるため、ユーザシステム上の ROM またはフラッシュメモリなどの書き換え不可能な領域には設定できません。 [エディタ]ウィンドウおよび[逆アセンブリ]ウィンドウから設定することもできます。
オンチップブレークポイント	チップ内蔵のブレーク機能です。アドレスバス条件、データバス条件、Bus/Area 条件、成立回数条件を設定できます。ソフトウェアブレークが設定できないユーザシステム上の ROM またはフラッシュメモリなどに設定することもできます。複数のオンチップブレークポイントを組み合わせたシーケンシャルブレークを設定することができます。 [エディタ]ウィンドウおよび[逆アセンブリ]ウィンドウから設定することもできます。
オンエミュレータブレークポイント	E6000H エミュレータハードで実現しているブレーク機能です。アドレスバス条件、データバス条件、Bus/Area 条件、外部プローブ信号条件、外部割り込み信号条件、成立回数条件を設定できます。E6000H エミュレータハードで実現しているため、条件成立からブレークが発生するまで数サイクルかかることがあります。

ソフトウェアブレークポイント、オンチップブレークポイントおよびオンエミュレータブレークポイントは [イベントポイント]ウィンドウから設定することができます。

[イベントポイント]ウィンドウを開くには、[表示 -> コード -> イベントポイント]を選択するか、[イベントポイント]ツールバーボタン  をクリックします。

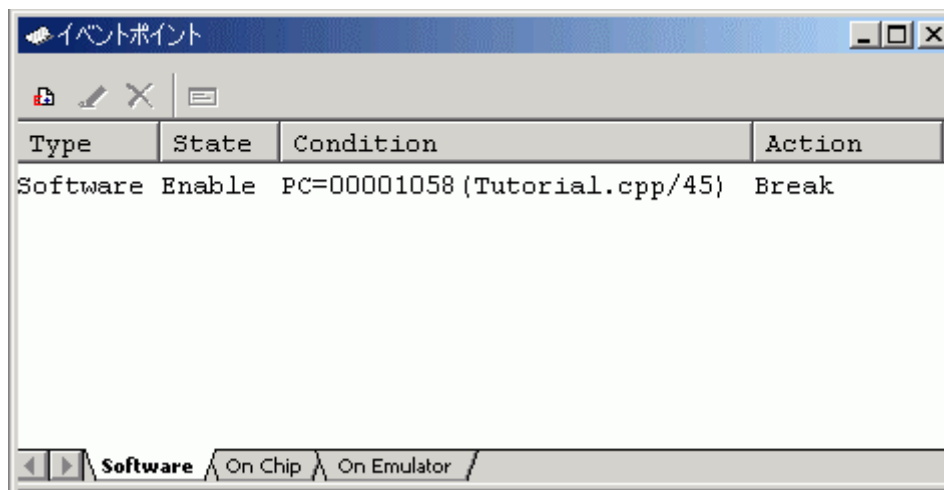


図3.19 イベントポイントウィンドウ

[イベントポイント]ウィンドウには、3枚のシートがあります。

- [Software]シート
ソフトウェアブレークポイントの設定内容を表示します。また、ソフトウェアブレークポイントの設定、変更および解除を行うことができます。
- [On Chip]シート
オンチップブレークポイントを表示、設定します。
- [On Emulator]シート
オンエミュレータブレークポイントを表示、設定します。

【注】 イベントポイントについての注意事項は「5.5 イベント機能について」を参照してください。

3.7.1 ソフトウェアブレークポイントを設定する

[イベントポイント]ウィンドウ - [Software]シートではソフトウェアブレークポイントの設定内容の表示、変更および追加ができます。

[Software]シート上のポップアップメニューから[追加...]または[編集...]を選択すると[ブレークポイントプロパティ]ダイアログボックス - [ソフトウェアブレーク]ページを表示します。

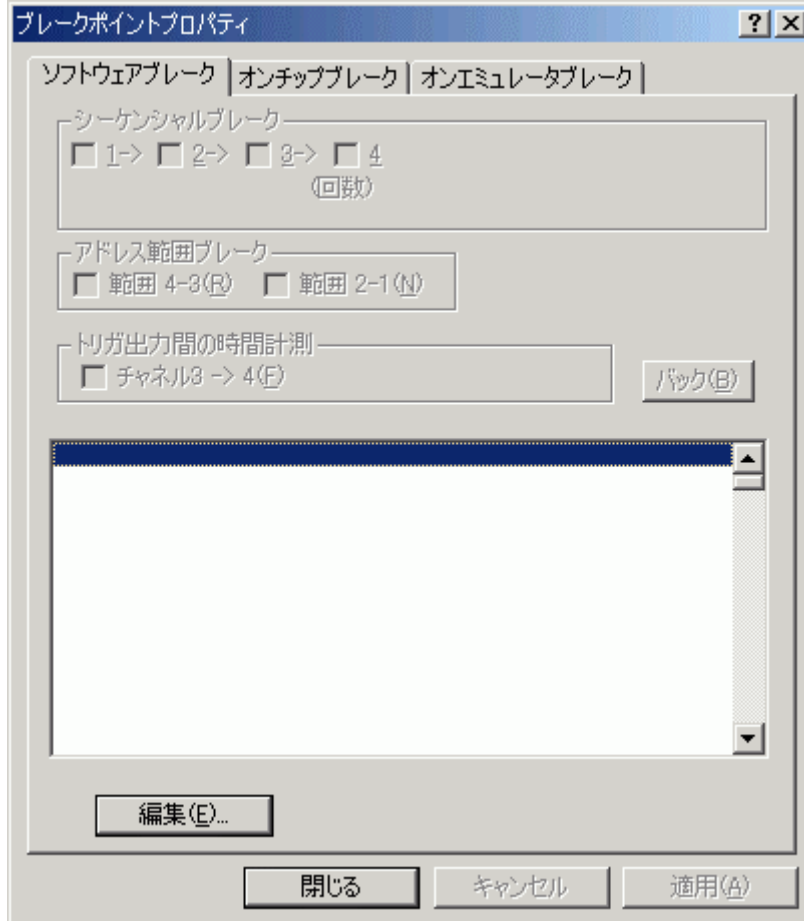


図3.20 ブレークポイントプロパティダイアログボックス (ソフトウェアブレークページ)

新しいソフトウェアブレイクポイントを追加する場合は、[ソフトウェアブレイク]ページのリストボックスから空白行を選択し、[編集...]ボタンをクリックしてください。既存の設定を変更する場合は、変更するソフトウェアブレイクポイントをリストボックスから選択し、[編集...]ボタンをクリックしてください。

ソフトウェアブレイクポイントを設定するための、[ソフトウェアブレイク]ダイアログボックスを表示します。

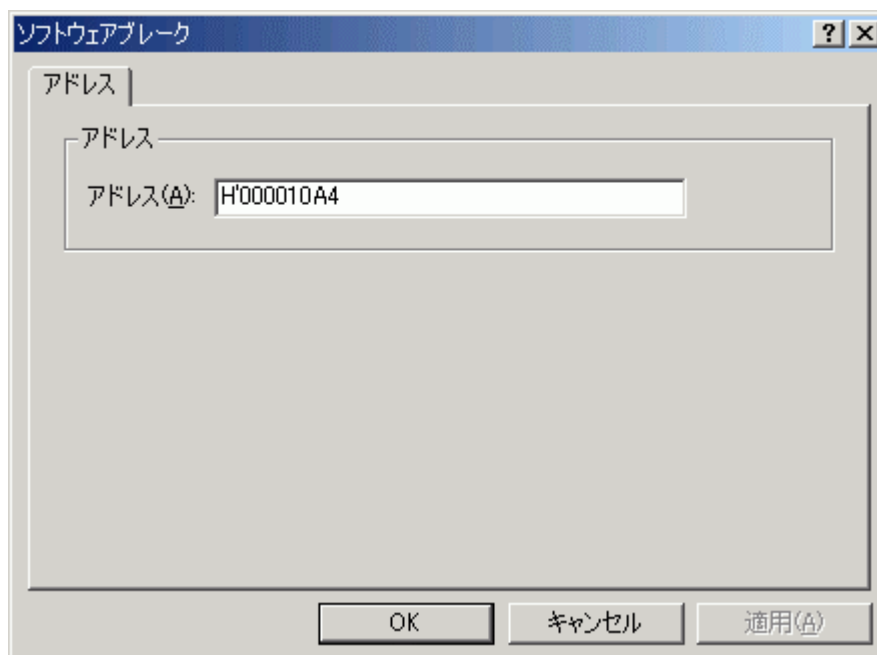


図3.21 ソフトウェアブレイクダイアログボックス

[アドレス]エディットボックスに設定するアドレスを指定し、[OK]ボタンを押してください。

3.7.2 オンチップブレークポイントを設定する

[イベントポイント]ウィンドウ - [On Chip]シートではオンチップブレークポイントの設定内容の表示、変更ができます。

[On Chip]シート上のポップアップメニューから[追加...]または[編集...]を選択すると[ブレークポイントプロパティ]ダイアログボックス - [オンチップブレーク]ページを表示します。

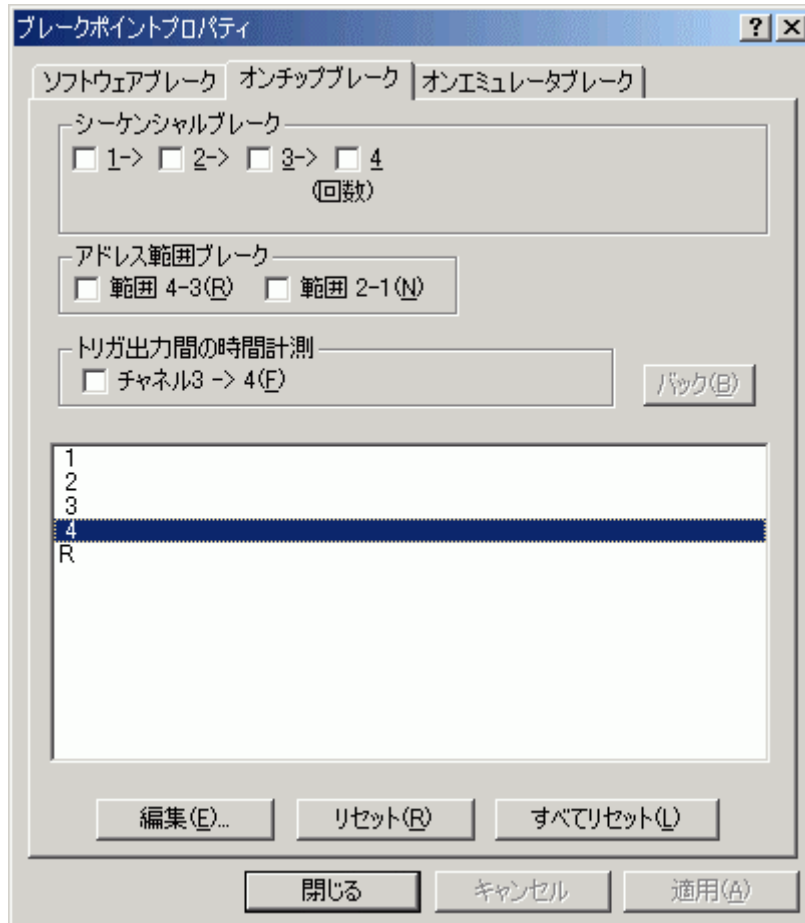


図3.22 ブレークポイントプロパティダイアログボックス (オンチップブレークページ)

[シーケンシャルブレーク]	オンチップブレークポイントを使用したシーケンシャルブレークを指定する場合に使用します。シーケンシャルブレークは複数チャンネルの条件が順番に成立した場合にブレークする機能です。シーケンシャルブレークは2ポイントから4ポイントの指定を選択できます。成立順序は1->2->3->4の順番です。 シーケンシャルブレークを使用する場合は、チェックボックスをチェックしてください。(2ポイントを使用する場合は3を、3ポイントを使用する場合は2を、4ポイントを使用する場合は1をチェックしてください。)
[アドレス範囲ブレーク]	チャンネルの組み合わせを指定し、ブレークの停止範囲を指定します。チャンネルの組み合わせは、以下を選択し指定します。 4-3: チャンネル4、3を使用しブレーク範囲を指定します。 2-1: チャンネル2、1を使用しブレーク範囲を指定します。
[トリガ出力間の時間計測]	2点間の時間計測をします。チャンネル3、4を使用し時間計測を行います。チャンネル3の成立後、チャンネル4が成立で時間が測定されます。測定結果は[ステータス]ウィンドウ - [Platform]シートの[RunTime Count]に表示されます。
[バック]	本ボタンをクリックすることにより、本ダイアログで設定した内容を、本ダイアログを表示した時点に戻します。
リストボックス	現在の設定状況をチャンネルごとに表示します。チャンネル番号のRは、シーケンシャルブレークのリセットポイントの設定になります。 設定のないチャンネルはチャンネル番号のみ表示します。
[編集...]	シーケンシャルブレークとして使用している場合は、チャンネル番号の横にSを表示します。 チャンネル指定後、本ボタンをクリックすることにより、指定したチャンネルに対するブレーク条件の設定を行う[オンチップブレーク チャンネル n]ダイアログボックスが表示されます。 (n: チャンネル番号)

- [リセット] 本ボタンをクリックすることにより、選択しているチャンネルの設定を解除します。
 [すべてリセット] すべてのチャンネルの設定を解除します。

[オンチップブレーク チャンネル n]ダイアログボックスでは、[アドレス]ページ、[データ]ページ、[バス/エリア]ページ、[回数]ページおよび[アクション]ページの各条件を組み合わせることで複雑なブレーク条件を設定することができます。

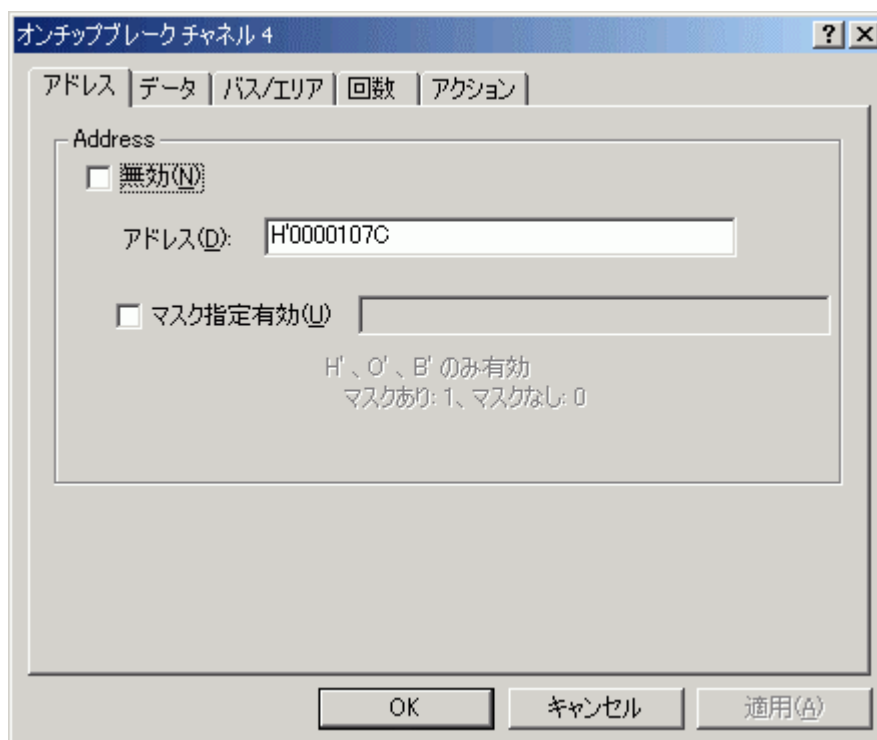


図3.23 [オンチップブレーク チャンネル n] ダイアログボックス

- | | |
|-----------|--|
| [アドレス] | アドレスバス条件を設定します。 |
| [無効] | アドレスバス条件を無効にします。 |
| [アドレス] | アドレスバス値を指定します。 |
| [マスク指定有効] | マスク条件を指定します。[マスク指定有効]を選択した場合に、マスクする値を設定します。マスクを行ったビットは、どんな値でも条件が成立することになります。 |
| [データ] | データバス条件を設定します。 |
| [無効] | データバス条件を無効にします。 |
| [データ値] | データバス条件の値を設定します。 |
| [マスク指定有効] | マスク条件を指定します。[マスク指定有効]を選択した場合に、マスクする値を設定します。マスクを行ったビットは、どんな値でも条件が成立することになります。 |
| [アクセスサイズ] | データアクセスサイズを指定します。 |
| [バス/エリア] | アクセスタイプ、バスステータスおよびリード/ライトサイクル条件を設定します。 |
| [アクセスタイプ] | アクセスタイプ条件を設定します。 |
| [バスステータス] | バスステータス条件を設定します。[無効]を選択した場合は、バスステータス条件を無効にします。 |
| [リード/ライト] | リード/ライト条件を指定します。[無効]を選択した場合は、リード/ライト条件を無効にします。 |
| [回数] | 条件成立回数を設定します。[無効]を選択した場合は、条件成立回数は1になります。 |
| [アクション] | |
| [ブレーク] | 設定条件成立時に実行停止します。 |
| [実行前] | 条件が成立したアドレスを実行後に停止します。 |
| [実行後] | 条件が成立するアドレスを実行前に停止します。 |
| [トリガ出力] | 設定条件成立時にトリガ出力を行います。 |

3.7.3 オンエミュレータブレークポイントを設定する

[イベントポイント]ウィンドウ - [On Emulator]シートではオンエミュレータブレークポイントの設定内容の表示、変更ができます。

[On Emulator]シート上のポップアップメニューから[追加...]または[編集...]を選択すると[ブレークポイントプロパティ]ダイアログボックス - [オンエミュレータブレーク] ページを表示します。

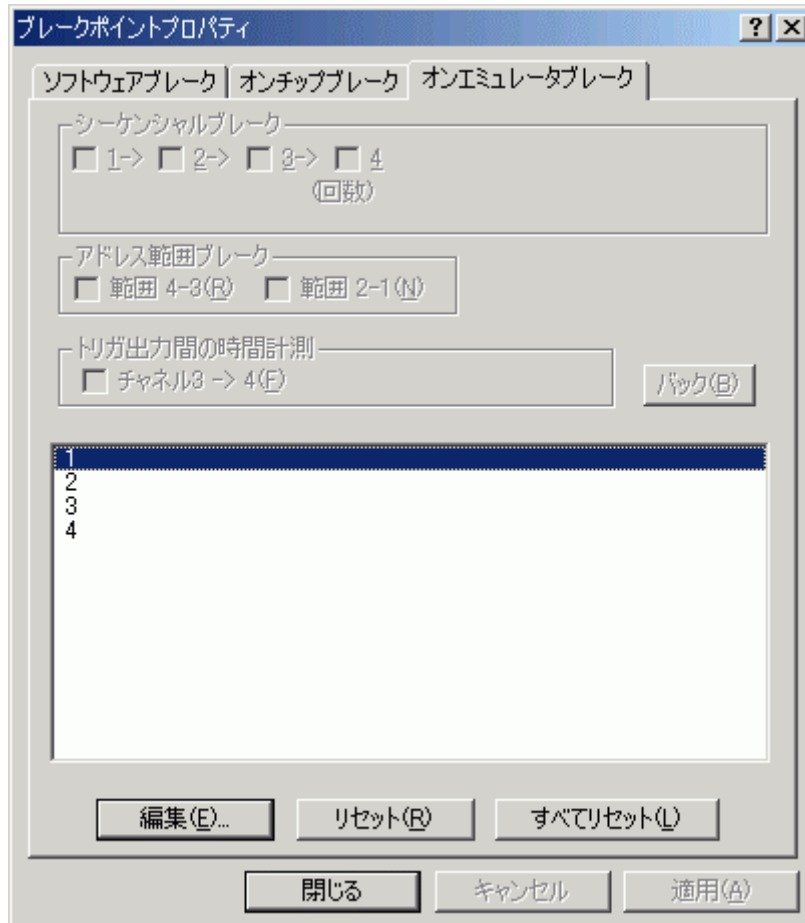


図3.24 ブレークポイントプロパティダイアログボックス (オンエミュレータブレークページ)

- | | |
|-----------|---|
| リストボックス | 現在の設定設定状況をチャンネルごとに表示します。
設定のないチャンネルはチャンネル番号のみ表示します。 |
| [編集...] | チャンネル指定後、本ボタンをクリックすることにより、指定したチャンネルに対するブレーク条件の設定を行う[オンエミュレータブレーク チャンネル n]ダイアログボックスが表示されます。
(n : チャンネル番号) |
| [リセット] | 本ボタンをクリックすることにより、選択しているチャンネルの設定を解除します。 |
| [すべてリセット] | すべてのチャンネルの設定を解除します。 |

[オンエミュレータブレーク チャンネル n]ダイアログボックスでは、[アドレス]ページ、[データ]ページ、[バス/エリア]ページ、[プローブ]ページ、[割込み]ページおよび[回数]ページの各条件を組み合わせることで複雑なブレーク条件を設定することができます。

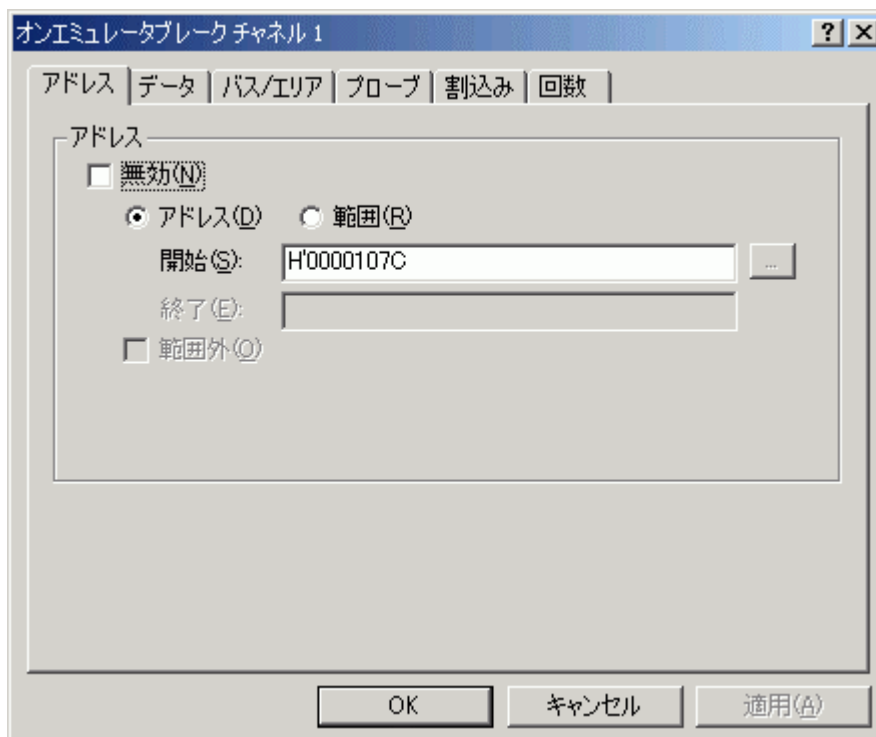


図3.25 [オンエミュレータブレーク チャンネル n] ダイアログボックス

[アドレス]	アドレスバス条件を設定します。
[無効]	アドレスバス条件を無効にします。
[アドレス]	[開始]で指定したアドレスバスの値をブレーク条件にする場合に選択します。
[範囲]	[開始](開始アドレス)~[終了](終了アドレス)で指定したアドレスバスの値の範囲でブレークします。
[範囲外]	[範囲]で設定した範囲以外のアドレスバスでブレークさせる場合に選択します。
[...]	関数のアドレス範囲を[開始]と[終了]に設定することができます。詳細は「5.10 入力形式について」を参照してください。
[データ]	データバス条件を設定します。
[無効]	データバス条件を無効にします。
[データ値]	データバス条件の値を設定します。
[マスク指定有効]	マスク条件を指定します。[マスク指定有効]を選択した場合に、マスクする値を設定します。マスクを行ったビットは、どんな値でも条件が成立することになります。
[この値以外]	設定した値以外をデータバス条件とする場合に選択します。
[アクセスサイズ]	データアクセスサイズを指定します。
[位置]	データバスの値を数値で設定します。有効なデータバスの位置を指定します。
[ロングワード]	なし
[ワード]	4n は上位ワード、4n+2 は下位ワードになります。
[バイト]	4n は上位ワードの上位バイト、4n+1 は上位ワードの下位バイト、4n+2 は下位ワードの上位バイト、4n+3 は下位ワードの下位バイトになります。
[バス/エリア]	アクセスタイプ、バスステータスおよびリード/ライトサイクル条件を設定します。
[アクセスタイプ]	アクセスタイプ条件を設定します。[無効]を選択した場合は、アクセスタイプ条件を無効にします。
[バスステータス]	バスステータス条件を設定します。[無効]を選択した場合は、バスステータス条件を無効にします。
[エリア]	エリア条件を指定します。[無効]を選択した場合は、エリア条件を無効にします。
[On chip ROM]	内蔵 ROM を条件とします。
[On chip RAM]	内蔵 RAM を条件とします。
[On chip I/O]	内蔵 I/O を条件とします。
[External]	エミュレーションメモリを割り付けていない外部エリアを条件とします。
[Emulator]	エミュレーションメモリを割り付けた外部エリアを条件とします。

3 デバッグ

[リード/ライト]	リード/ライト条件を指定します。[無効]を選択した場合は、リード/ライト条件を無効にします。
[プローブ]	外部プローブ信号 (PRB1 ~ PRB4) の状態 (HIGH/LOW) を条件にします。[無効]を指定した場合は、指定した外部プローブ信号の状態をブレイク条件としません。
[割込み]	IRQ 信号および NMI 信号を条件にします。[無効]を指定した場合は、各信号の状態をブレイク条件としません。
[回数]	条件成立回数を設定します。[無効]を選択した場合は、条件成立回数は 1 になります。

3.7.4 イベントポイントの編集

ソフトウェアブレイクポイント、オンチップブレイクポイント、オンエミュレータポイントに対する設定以外の操作方法はすべて共通となっています。

3.7.5 イベントポイントの設定内容を変更する

変更したいイベントポイントを選択後ポップアップメニューから[編集...]を選択すると、各イベントに対応した設定ダイアログボックスが開き、設定内容を変更することができます。[編集...]メニューはイベントポイントを 1 個選択しているときのみ有効となります。

3.7.6 イベントポイントを有効にする

イベントポイントを選択後ポップアップメニューから[有効]を選択すると、選択しているイベントポイントを有効にします。

3.7.7 イベントポイントを無効にする

イベントポイントを選択後ポップアップメニューから[無効]を選択すると、選択しているイベントポイントを無効にします。無効にした場合は、イベントポイントはリストには残りますが、指定した条件が一致してもイベントは発生しません。

3.7.8 イベントポイントを削除する

イベントポイントを選択後ポップアップメニューから[削除]を選択すると、選択しているイベントポイントを削除します。イベントポイントを削除しないで、詳細情報は保持したまま、条件が成立してもイベントを発生させないようにするには、[無効]オプションを使用します（「3.7.7 イベントポイントを無効にする」参照）。

3.7.9 イベントポイントをすべて削除する

ポップアップメニューから[すべて削除]を選択すると、すべてのイベントポイントを削除します。

3.7.10 イベントポイントのソース行を表示する

イベントポイントを選択後ポップアップメニューから[ソースを表示]を選択すると、ブレイクポイントのある[ソース]または[逆アセンブリ]ウィンドウをオープンします。[ソースを表示]メニューは対応するソースファイルをもつイベントポイントを 1 個選択しているときのみ有効となります。

3.8 トレース情報を見る

E6000H エミュレータでは命令の実行結果をトレース情報としてトレースバッファに取得し、表示することができます。


トレース情報は[トレース]ウィンドウに表示します。トレース情報の取得条件は、[トレース取得プロパティ]ダイアログボックスで設定します。

トレース機能は E6000H エミュレータのハードウェア回路によりバスサイクル単位にトレース情報を取得し、トレースバッファに保持するためリアルタイム性は損なわれません。

リアルタイムトレースバッファは、128k までのバスサイクルを保持でき、バッファがいっぱいになった場合は取得したトレース情報のうち最も古いデータに新しいデータを上書きします。

【注】 トレース機能についての注意事項は「5.6 トレース機能について」を参照してください。

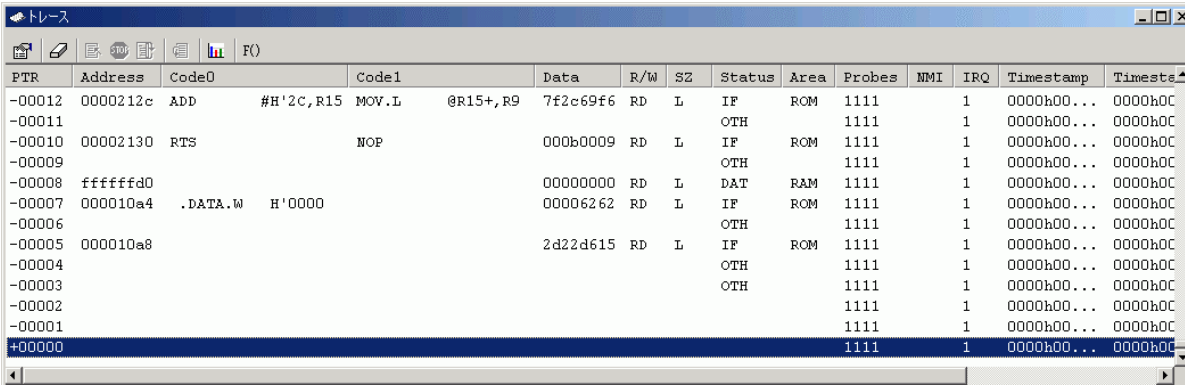
3.8.1 トレースウィンドウを開く

[トレース]ウィンドウを開くには、[表示 -> コード -> トレース]を選択するか、[トレース]ツールバーボタンをクリックします。

3.8.2 トレース情報を取得する

E6000H エミュレータはトレース情報の取得条件を設定しない場合、デフォルトで無条件に全バスサイクルをトレース取得します。(フリートレースモード)

フリートレースモードではユーザプログラムの実行開始と共にトレース取得を開始し、ユーザプログラムの停止によりトレース取得を停止します。取得したトレース情報は[トレース]ウィンドウに表示します。



PTR	Address	Code0	Code1	Data	R/W	SZ	Status	Area	Probes	NMI	IRQ	Timestamp	Timestamp	
-00012	0000212c	ADD	#H'2C,R15	MOV.L	@R15+,R9	7f2c69f6	RD	L	IF	ROM	1111	1	0000h00...	0000h0C
-00011									OTH	1111	1	0000h00...	0000h0C	
-00010	00002130	RTS		NOP		000b0009	RD	L	IF	ROM	1111	1	0000h00...	0000h0C
-00009									OTH	1111	1	0000h00...	0000h0C	
-00008	ffffffd0					00000000	RD	L	DAT	RAM	1111	1	0000h00...	0000h0C
-00007	000010a4	.DATA.W	H'0000			00006262	RD	L	IF	ROM	1111	1	0000h00...	0000h0C
-00006									OTH	1111	1	0000h00...	0000h0C	
-00005	000010a8					2d22d615	RD	L	IF	ROM	1111	1	0000h00...	0000h0C
-00004									OTH	1111	1	0000h00...	0000h0C	
-00003									OTH	1111	1	0000h00...	0000h0C	
-00002										1111	1	0000h00...	0000h0C	
-00001										1111	1	0000h00...	0000h0C	
+00000										1111	1	0000h00...	0000h0C	

図3.26 トレースウィンドウ

3 デバッグ

Trace ウィンドウの表示内容は製品により異なります。本製品にて表示する項目は以下の通りです。

[PTR]	トレースメモリ内のサイクル番号。 最後に取得されたサイクルの番号を 0 とし、古いサイクルにさかのぼって、順に-1、-2 と番号が小さくなります。ディレイカウントが設定されている場合は、トレース停止条件が成立したサイクル番号を 0 とし、成立後停止するまでに実行されたサイクル(ディレイ期間中のサイクル)には、最後に取得されたサイクルに向かって順に+1、+2 と番号が大きくなります。
[Address]	アドレスバスの値(8桁の16進数)を表示します。
[Code0]	上位側16bitの実行命令コードを表示します。
[Code1]	下位側16bitの実行命令コードを表示します。
[Data]	バイト、ワード、ロングのデータ。それぞれ16進の2桁、4桁、8桁で表示します。
[R/W]	アクセスサイクルの種類。読み出しはRD、書き込みはWRと表示します。
[SZ]	アクセスしたサイズを示します。それぞれB(バイト)、W(ワード)、L(ロング)で表示します。
[Status]	バス状態を表示します。 DMA: 内蔵DMAC/DTC実行サイクル、AUD: AUDサイクル、BRL: ユーザバスリリース、DAT: CPUデータアクセスサイクル(PC相対のデータアクセスを除く)、IF: CPU命令フェッチサイクル(PC相対のデータアクセスも含む)、OTH: 上記以外のケース
[Area]	アクセスエリアの種別を表示します。 U32、U16、U08: それぞれ32ビット、16ビット、8ビットの外部メモリ空間へのアクセス(エミュレーションメモリへのアクセスは除く)、E32、E16、E08: それぞれ32ビット、16ビット、8ビットのエミュレーションメモリへのアクセス、ROM、I/O、RAM: それぞれ内蔵ROM、内蔵I/O、内蔵RAMへのアクセス
[Probes]	4本のプローブ信号の状態(2進数4桁)。 左からProbe4、Probe3、Probe2、Probe1の順で表示します。
[NMI]	NMI入力の状態を表示します。
[IRQ]	IRQ入力の状態を表示します。
[Timestamp]	バスサイクルのタイムスタンプ。 タイムスタンプは、ユーザプログラム実行を開始するたびに0からカウントを始めます。測定時間の最小単位は、[トレース取得プロパティ]ダイアログボックスで設定してください。
[Source]	実行命令アドレスに対応するソースコードを表示します。
[Label]	[Address]に対応するラベル(ラベルが設定されている場合のみ表示します)。
[Timestamp-Difference]	前の行とのタイムスタンプの差分時間を表示します。

[トレース]ウィンドウ内の不要なカラムは非表示にすることができます。

カラムを非表示にする場合はヘッダカラム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より非表示にしたいカラムを選択してください。

カラムを再表示する場合は再度ポップアップメニューより該当のカラムを選択してください。

3.8.3 トレース情報取得条件を設定する

トレースバッファは有限であるため、バッファがいっぱいになった場合は最も古いトレースデータから順に新しいデータを上書きします。トレース情報の取得条件を設定することにより、有用なトレース情報のみを取得し、トレースバッファを有効に活用することができます。

トレース情報の取得条件には、以下の種類があります。

フリートレース	ユーザプログラムの実行を開始した時点からブレイクするまでのトレース情報を連続的に取得します。(トレース情報の取得条件を設定しない場合です。)
シーケンシャルトレース停止	トレース取得条件の成立順序を指定し、全ての条件が成立した時点でトレース取得を停止します。 最大7レベルの通過ポイントと1ポイントのリセット条件を設定することができます。トレース取得は停止しますが、ユーザプログラムはブレイクしません。
トレースバッファオーバーフローによるトレース停止	トレース取得時にエミュレータ本体のトレースバッファがオーバーフローした場合に取得を停止します。 トレース取得は停止しますが、ユーザプログラムはブレイクしません。
トレース停止	指定した条件が成立したときにトレース取得を停止します。このモードでは、ユーザプログラムの実行を停止せずに、トレース取得を停止します。 最大12ポイントの独立したトレース停止条件を設定することができます。 トレース取得は停止しますが、ユーザプログラムはブレイクしません。
アドレス範囲トレース	開始アドレスと終了アドレスで設定されている範囲(サブルーチン)の命令およびオペランドアクセスをトレースに取得します。ただし、指定されたサブルーチンが他のサブルーチンをコールした場合、コールされたサブルーチンはトレースされません。 最大12ポイントの独立したアドレス範囲トレースを設定することができます。
条件トレース	指定した条件が成立した箇所のみをトレース取得します。 最大12ポイントの独立した条件トレースを設定することができます。
条件アドレス範囲トレース	開始アドレスと終了アドレスで設定されている範囲(サブルーチン)の命令およびオペランドをアクセスし、設定されている条件と一致するバスサイクルのみトレース情報を取得します。 アドレス範囲トレースと条件トレースを組み合わせた取得になります。 最大6ポイントの独立したトレース停止条件を設定することができます。
2点間トレース	開始条件として指定したアドレス条件成立から終了条件として指定したアドレス条件成立までの2点間のトレースを取得します。
実行時間計測	トレース情報の取得条件を利用した2点間の実行時間計測を行います。
トリガ出力	指定した条件成立時に、トリガ端子からパルスを出力します。

トレース情報の取得条件はポップアップメニューから[設定...]を選択すると表示される[トレース取得プロパティ]ダイアログボックスで設定します。

[トレース取得プロパティ]ダイアログボックスは[条件設定]ページと[オプション設定]ページより構成されています。

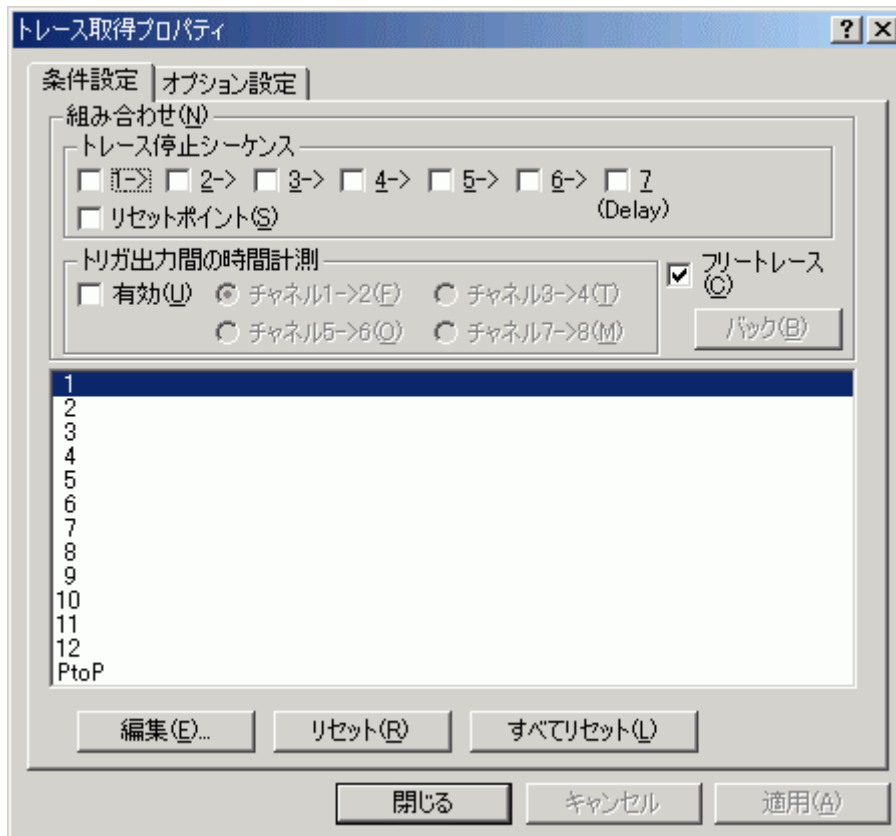


図3.27 トレース取得プロパティダイアログボックス (条件設定ページ)

[トレース停止シーケンス]	チャンネル1から7を使用したシーケンシャルトレース停止を指定する場合に使用します。シーケンシャルトレース停止は複数チャンネルの条件が順番に成立した場合にトレースの取得を停止する機能です。シーケンシャルトレース停止は2ポイントから7ポイントの条件指定と1ポイントのリセット条件を選択できます。成立順序は1->2->3->4->5->6->7の順番です。シーケンシャルトレース停止を使用する場合は、必要なチャンネルのチェックボックスをチェックしてください。リセット条件を設定する場合は、[リセットポイント]チェックボックスをチェックしてください。リセット条件はチャンネル8を使用します。リセット条件が成立した場合は、それまで成立したシーケンシャルトレース停止条件がすべて破棄され、再度先頭の条件からの成立をチェックします。シーケンシャルトレース停止を有効にした場合、チャンネル1から7のうちシーケンシャルトレース停止に使用しないチャンネルは単独の設定ができなくなります。
[トリガ出力間の時間計測]	実行時間計測を行うチャンネルを指定します。[有効]をクリックすることにより、トレース時間の計測を行います。チャンネルの組み合わせは、測定スタート - 測定エンドの組み合わせで、1 - 2、3 - 4、5 - 6、7 - 8の4通りです。
[フリートレース]	フリートレースを選択します。[フリートレース]が有効な場合はトレース情報の取得条件設定は無視されます。
[バック]	クリックすることにより、[組み合わせ]グループで設定した内容を、本ダイアログボックスを表示した時点に戻します。
リストボックス	現在の設定状況をチャンネルごとに表示します。設定のないチャンネルはチャンネル番号のみ表示します。シーケンシャルトレース停止として使用している場合はチャンネル番号の横にSを表示します。シーケンシャルトレース停止のリセット条件を有効している場合はチャンネル8の横にRを表示します。PtoPは2点間トレースに使用します。使用できないチャンネルはチャンネル番号の横にUNUSEDを表示します。
[編集...]	チャンネル指定後、本ボタンをクリックすることにより、指定したチャンネルの設定を行う[トレース取得条件 チャンネル n]ダイアログボックスが表示されます。(n:チャンネル番号、PtoP)
[リセット]	本ボタンをクリックすることにより、選択しているチャンネルの設定を解除します。
[すべてリセット]	すべてのチャンネルの設定を解除します。

(2) [オプション設定]ページ

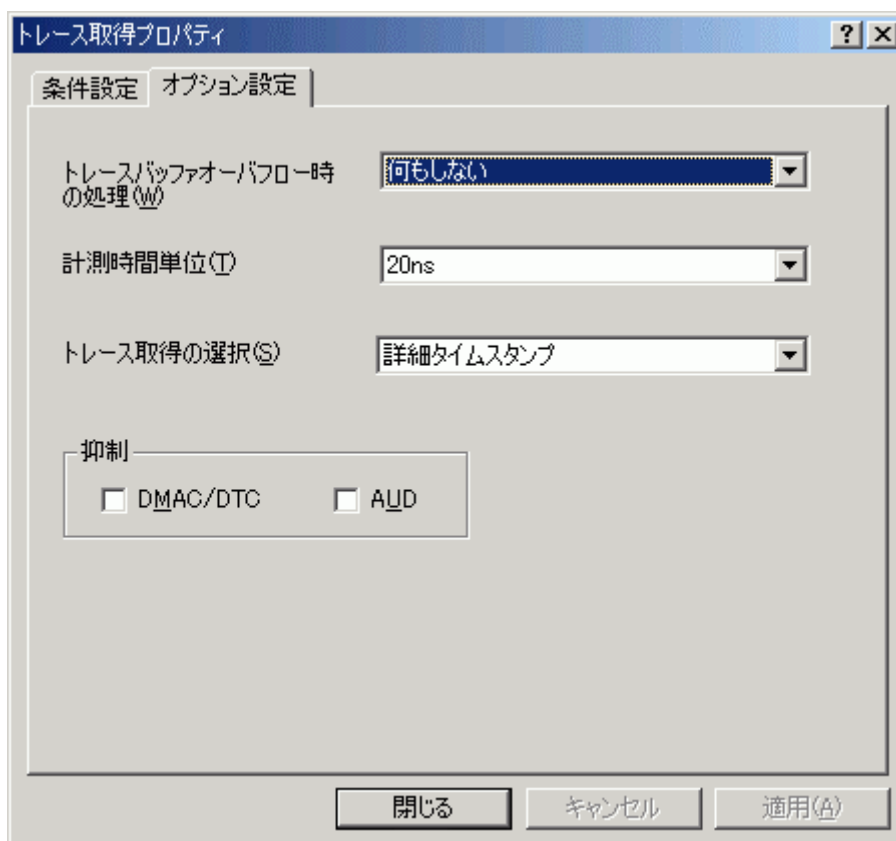


図3.28 トレース取得プロパティダイアログボックス（オプション設定ページ）

[トレースバッファ オーバーフロー時の処 理]	トレースバッファがいっぱいになった場合の動作を設定します。
[何もしない]	バッファがいっぱいになった場合は最も古いトレース情報から順に上書きします。
[トレース停止]	バッファがいっぱいになった場合はトレース取得を停止します。 トレース取得は停止しますが、ユーザプログラムの実行は停止しません。
[実行停止およびト レース停止]	バッファがいっぱいになった場合はユーザプログラムの実行を停止します。
[計測時間単位]	バストレース情報のタイムスタンプの最小時間を次の中から選択します。
[52us]	最小時間（52us）でタイムスタンプ情報を取得します。
[1.6us]	最小時間（1.6us）でタイムスタンプ情報を取得します。
[20ns]	最小時間（20ns）でタイムスタンプ情報を取得します。
[Clock]	システムクロック（ ）に同期したバスサイクルのクロック数を取得します。
[Clock/2]	システムクロック（ ）の 1/2 に同期したバスサイクルのクロック数を取得します。
[Clock/4]	システムクロック（ ）の 1/4 に同期したバスサイクルのクロック数を取得します。
[Clock/8]	システムクロック（ ）の 1/8 に同期したバスサイクルのクロック数を取得します。
[トレース取得の選択]	タイムスタンプと IRQ7-0 の表示形態を選択します。本 E6000H エミュレータにおいては、タイムスタンプと IRQ のトレースは同じハードリソースを使用し実現しています。このため、32 ビットタイムスタンプカウンタと IRQ7-0 のトレースを同時に行うことはできません。 使用状況にあわせて下記項目を選択してください。
[詳細タイムスタ ンプ]	タイムスタンプを 32 ビットで取得し表示します。IRQ 信号は取得しないため表示しません。 タイムスタンプ表示は 32 ビットタイムスタンプカウンタを全て有効にします。
[IRQ7-0 全て表示]	IRQ7-0 の状態をビット単位で表示します。 タイムスタンプカウンタの下位 16 ビットは 0 固定になります。
[抑制]	バスサイクルタイプの DMAC/DTC 及び AUD を取得しない指定をします。

3 デバッグ

(3) [トレース取得条件 チャンネル n]ダイアログボックス

[トレース取得条件 チャンネル n]ダイアログボックスは、シーケンシャルトレース停止の通過ポイントとリセットポイントの設定、トレース停止条件の設定、アドレス範囲トレース条件の設定、条件アドレス範囲トレース条件の設定、条件トレースの設定、2点間トレース条件の設定、実行時間計測およびトリガ出力の条件設定に使用します。

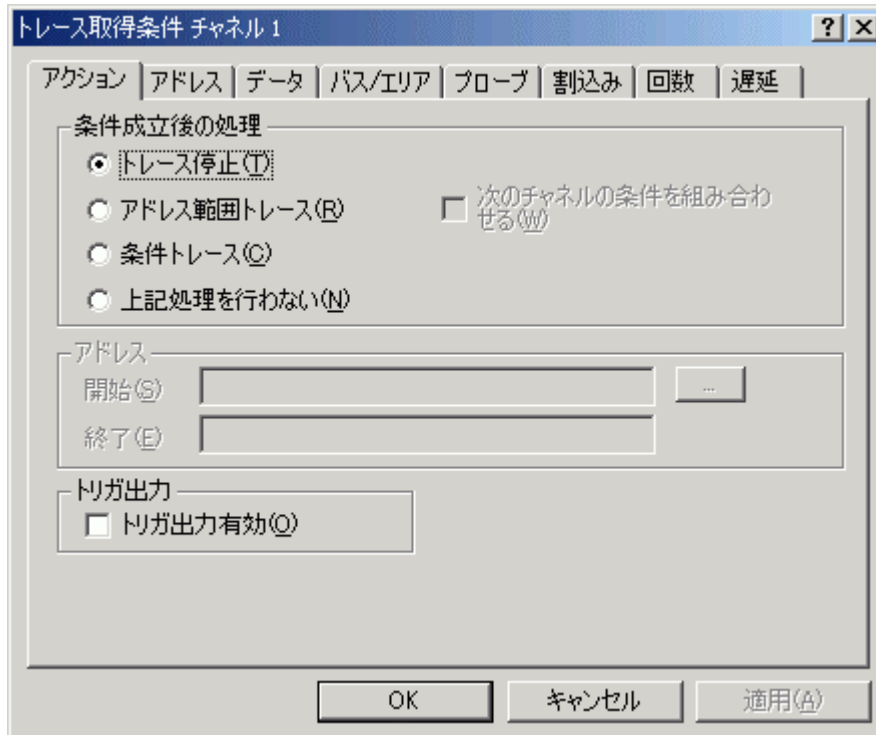


図3.29 [トレース取得条件 チャンネル n]ダイアログボックス

[トレース取得条件 チャンネル n]ダイアログボックスは、[アクション]ページ、[アドレス]ページ、[データ]ページ、[バス/エリア]ページ、[プローブ]ページ、[割込み]ページ、[回数]ページおよび[遅延]ページから構成されます。

各ページの条件を組み合わせることで複雑な設定を行うことができます。

[アクション]	
[条件成立後の処理]	条件成立時の処理を指定します。
[トレース停止]	トレース停止を設定します。
[アドレス範囲トレース]	アドレス範囲トレースを設定します。 [アドレス範囲トレース]を選択し、[次のチャンネルの条件を組み合わせる]をチェックすることで、条件アドレス範囲トレースを設定することができます。
[条件トレース]	条件トレースを設定します。
[上記処理を行わない]	条件が成立しても上記の処理は行わない設定を指定します。トリガ出力や実行時間の計測を行う場合に使用します。
[アドレス]	アドレス範囲トレースと条件アドレス範囲トレースのアドレス範囲の設定および2点間トレース条件の開始アドレスと終了アドレスを設定します。
[開始]	開始アドレスを設定します。
[終了]	終了アドレスを設定します。
[...]	関数のアドレス範囲を[開始]と[終了]に設定することができます。
[トリガ出力有効]	トレース条件成立後、トリガ出力します。
[アドレス]	アドレスバス条件を指定します。
[データ]	データバス条件を指定します。
[バス/エリア]	アクセスタイプ、バスステータスおよびリード/ライトサイクル条件を設定します。
[プローブ]	外部プローブ信号 (PRB1 ~ PRB4) の状態 (HIGH/LOW) を条件にします。
[割込み]	IRQ 信号および NMI 信号の状態を条件にします。
[回数]	条件成立回数を設定します。
[遅延]	条件成立後処理開始までの遅延サイクル数を指定します。遅延サイクルを利用することで、条件成立前後のトレース情報を確認することができます。[無効]を指定した場合は、遅延を行いません。

- 【注】 1. [アドレス]ページ、[データ]ページ、[バス/エリア]ページ、[プローブ]ページ、[割込み]ページおよび[回数]ページの設定はオンエミュレータブレークと同様です。オンエミュレータブレークについては、「3.7 イベントポイントを使用する」を参照してください。
2. アドレス範囲トレースの範囲指定は、開始アドレス 終了アドレスとなるように指定してください。
3. 条件アドレス範囲トレースはチャンネルを2つ使用します。条件アドレス範囲トレースを設定する場合は、奇数チャンネル (2n+1) にアドレス範囲トレースを設定し、偶数チャンネル (2n) に条件トレースを設定したのち[次のチャンネルの条件を組み合わせる]をチェックしてください。

3.8.4 Trace レコードを検索する

トレースレコードを検索するには[Trace Find]ダイアログボックスを使用します。
 [Trace Find]ダイアログボックスを開くには、ポップアップメニューの[検索...]を選択します。

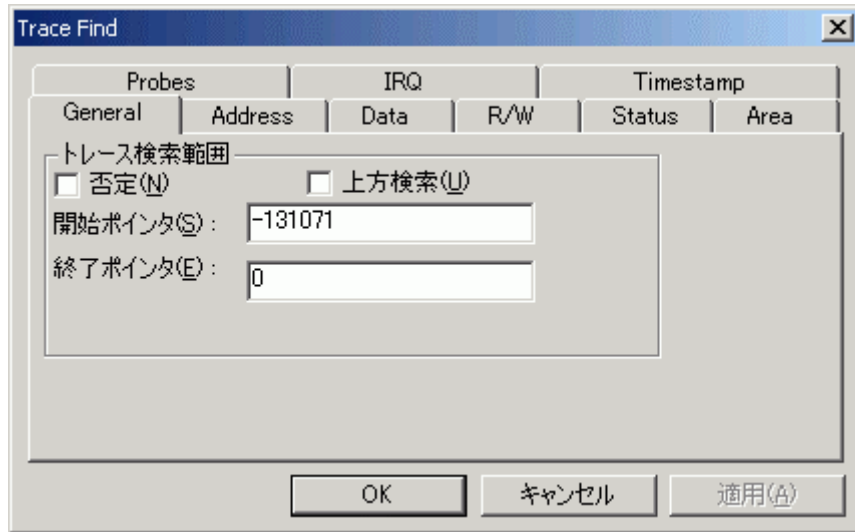


図3.30 Trace Find ダイアログボックス

[Trace Find]ダイアログボックスは下記ページより構成されています。

[General]	検索範囲を指定します。
[否定]	チェックすると他ページで設定した項目の否定条件で検索します。
[上方検索]	チェックすると上方検索を行います。
[開始ポイント]	検索を開始する PTR の値を入力します。
[終了ポイント]	検索を終了する PTR の値を入力します。
[Address]	実行アドレス条件を指定します。
[無効]	チェックすると実行アドレスを検出しません。
[値]	指定した実行アドレスを検出します。実行アドレス値を入力します。
[Date]	データ条件を指定します。
[無効]	チェックするとデータを検出しません。
[値]	指定したデータを検出します。データ値を入力します。
[R/W]	リード/ライト条件を指定します。
[無効]	チェックするとリード/ライト条件を検出しません。
[設定]	指定したリード/ライト条件を検出します。
	RD: リードサイクル
	WR: ライトサイクル
[Status]	バスステータス条件を指定します。
[無効]	チェックすると、バス条件を検出しません。
[設定]	指定したバス条件を検出します。
[Area]	アクセスエリア条件を指定します。
[無効]	チェックすると、エリア条件を検出しません。
[設定]	指定したエリア条件を検出します。
[Probes]	プローブ条件を指定します。
[無効]	チェックするとプローブ条件を検出しません。
[設定]	指定したプローブ条件を検出します。
	Don't care: 選択したプローブ条件を検出しません。
	High: プローブ信号状態が High
	Low: プローブ信号状態が Low

[IRQ]	IRQ 条件を指定します。
[無効]	チェックすると IRQ 条件を検出しません。
[設定]	指定した IRQ 条件を検出します。
	Don't care: 選択した IRQ 条件を検出しません。
	High: 信号状態が High
	Low: 信号状態が Low
[Timestamp]	タイムスタンプ条件を指定します。
[無効]	チェックするとタイムスタンプ条件を検出しません。
[値]	指定したタイムスタンプ値を検出します。タイムスタンプ値を入力します。(全ての桁の入力が必須)

各ページで条件を設定し、[OK]ボタンをクリックすることにより、サーチ条件を設定し、検索を開始します。[キャンセル]ボタンをクリックすると、設定しないでダイアログボックスを閉じます。

検索の結果一致するトレースレコードが見つかった場合は当該レコード行を強調表示します。一致するトレースレコードが見つからなかった場合は、メッセージダイアログボックスを表示します。

トレース情報の検索は各ページで設定した条件がすべて一致するトレース情報のみを検索します。

トレースレコードが検索できた場合は、ポップアップメニューで[次を検索]を選択すると、次のトレースレコードを検索できます。

3.8.5 トレース情報をクリアする

トレース情報をクリアするには、ポップアップメニューから[クリア]を選択します。トレース情報を保持しているトレースバッファを空にします。複数の[トレース]ウィンドウが開いているときは、それらは同じバッファをアクセスしているため、すべての[トレース]ウィンドウをクリアすることになります。

3.8.6 トレース情報をファイルに保存する

トレース情報をファイルに保存するには、ポップアップメニューから[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。[トレース]ウィンドウに表示しているトレース情報をテキストファイルとして保存します。保存する範囲を、[PTR]の範囲によって指定することができます(すべてのバッファをセーブするには、数分かかることがあります)。このファイルは保存のみ可能で、[トレース]ウィンドウへの読み込みはできないことに注意してください。

【注】 トレース情報をフィルタリングした場合、保存する範囲の指定はできません。フィルタリングした結果 [トレース]ウィンドウに表示されたトレース情報すべてを保存します。保存する範囲を指定したい場合は[Trace Filter]ダイアログボックスの[General]ページよりフィルタ範囲を指定してください。フィルタ機能については「3.8.11 取得したトレース情報から必要なレコードを抽出する」を参照してください。

3.8.7 [エディタ]ウィンドウを表示する

トレースレコードに対応する[エディタ]ウィンドウを表示するには2通りの方法があります。

- (1) トレースレコード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
- (2) トレースレコードをダブルクリックする

上記の操作により、[エディタ]ウィンドウあるいは[逆アセンブル]ウィンドウを開いてソース表示し、選択した行をカーソルで示します。

3.8.8 ソース表示を整形する

ポップアップメニューで[ソーストリム]を選択すると、ソースプログラムの左側の空白を取り除きます。

取り除いた状態だと[ソーストリム]メニューの左にチェックが付きます。チェックありの状態では[ソーストリム]メニューを選択すると取り除いた空白を元に戻します。

3.8.9 トレース情報の取得を一時的に停止する

ユーザプログラム実行中、一時的にトレース情報の取得を停止するにはポップアップメニューから[停止]を選択します。

トレース取得を中止し、トレース表示を更新します。

ユーザプログラムを停止せずにトレース情報の取得のみ停止し、トレース情報を確認する場合などに使用します。

3.8.10 トレース情報の取得を再開する

ユーザプログラム実行中、一時的にトレース情報の取得を停止した場合、再度トレース情報の取得を再開するにはポップアップメニューから[リスタート]を選択します。

3.8.11 取得したトレース情報から必要なレコードを抽出する

取得したトレース情報から必要なレコードのみを抽出するにはフィルタ機能を使用します。

フィルタ機能はハードウェアにより取得したトレース情報をソフトウェアによりフィルタリングします。

取得条件を設定してトレース情報を取得する[トレース取得プロパティ]設定と異なり、取得したトレース情報に対し何度もフィルタ設定を変更することで必要な情報が簡単に抽出でき、データの分析に役立ちます。

フィルタ機能を使用してもトレースバッファの内容は変更されません。

トレースバッファは有限ですので、[トレース取得プロパティ]設定により有用なトレース情報をより多く取得することで、より効果的にデータの分析が可能となります。

フィルタ機能を使用するには[Trace Filter]ダイアログボックスを使用します。フィルタ範囲を指定します。

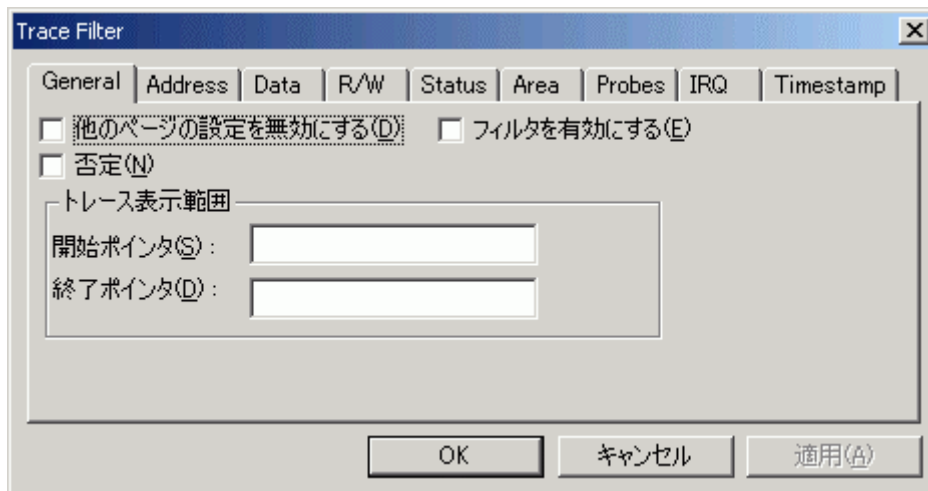


図3.31 Trace Filter ダイアログボックス

[Trace Filter]ダイアログボックスを開くには、ポップアップメニューの[フィルタ...]を選択します。

[Trace Filter]ダイアログボックスは下記ページより構成されています。

- [General] 検索範囲を指定します。
- [他のページ設定を無効にする] チェックするとサイクル番号のみ指定できます。他のオプションはすべて無効になります。
 - [フィルタを有効にする] チェックするとフィルタを有効にします。
 - [否定] チェックすると他ページで設定した項目の否定条件でフィルタリングします。
 - [開始ポインタ] フィルタを開始する PTR の値を入力します。
 - [終了ポインタ] フィルタを終了する PTR の値を入力します。
- [Address] 実行アドレス条件を指定します。
- [無効] チェックすると実行アドレスをフィルタしません。
 - [設定] 指定した実行アドレスを検出します。
 - [ポイント] 単一実行アドレスを入力します。
 - [範囲] 実行アドレス範囲を入力します。
 - [開始] 単一実行アドレスまたは実行アドレス範囲の開始アドレスを入力します。
 - [終了] 実行アドレス範囲の終了アドレスを入力します。
- [Data] データ条件を指定します。
- [無効] チェックするとデータをフィルタしません。
 - [設定] 指定したデータを検出します。
 - [ポイント] 単一データを入力します。
 - [範囲] データ範囲を入力します。
 - [開始] 単一データまたはデータ範囲の最小値を入力します。
 - [終了] データ範囲の最大値を入力します。
- [R/W] リード/ライト条件を指定します。
- [無効] チェックするとリード/ライト条件を検出しません。
 - [設定] 指定したリード/ライト条件を検出します。
 - [RD] チェックするとリードサイクルを検出します。
 - [WR] チェックするとライトサイクルを検出します。
- [Status] バスステータス条件を指定します。(複数指定可能です。)
- [無効] チェックすると、バス条件を検出しません。
 - [設定] 指定したバス条件を検出します。
- [Area] アクセスエリア条件を指定します。(複数指定可能です。)
- [無効] チェックすると、エリア条件を検出しません。
 - [設定] 指定したエリア条件を検出します。([無効] 選択時無効)
- [Probes] プロブ条件を指定します。
- [無効] チェックするとプロブ条件を検出しません。
 - [設定] 指定したプロブ条件を検出します。
 - Don't care: 選択したプロブ条件を検出しません。
 - High: プロブ信号状態が High
 - Low: プロブ信号状態が Low
- [IRQ] IRQ 条件を指定します。
- [無効] チェックすると IRQ 条件を検出しません。
 - [設定] 指定した IRQ 条件を検出します。
 - Don't care: 選択した IRQ 条件を検出しません。
 - High: IRQ 信号状態が High
 - Low: IRQ 信号状態が Low
- [Timestamp] タイムスタンプ条件を指定します。
- [無効] チェックするとタイムスタンプ値を検出しません。
 - [設定] 指定したタイムスタンプ値を検出します。
 - [ポイント] 単一タイムスタンプを指定します。
 - [範囲] タイムスタンプ範囲を指定します。
 - [開始] 単一タイムスタンプまたはタイムスタンプ範囲の最小値を入力します。
 - [終了] タイムスタンプ範囲の最大値を入力します。

各ページでフィルタ条件を設定し、[OK]ボタンをクリックすることにより、フィルタ条件にしたがいフィルタリングを行います。[キャンセル]ボタンをクリックすると、[Trace Filter]ダイアログボックスを開いた時点の設定のままダイアログボックスを閉じます。

フィルタリングは各ページで設定したフィルタ条件が 1 つ以上一致するトレース情報のみを[トレース]ウィンドウに表示します。

フィルタリングを行ってもトレースバッファの内容は変更されませんので、何度もフィルタ条件を変更しデータの分析ができます。

3.8.12 タイムスタンプの差を計算する

タイムスタンプ情報取得時、トレース結果より指定した2点間の時間差を計算するには、ポップアップメニューから[タイムスタンプ差...]を選択します。

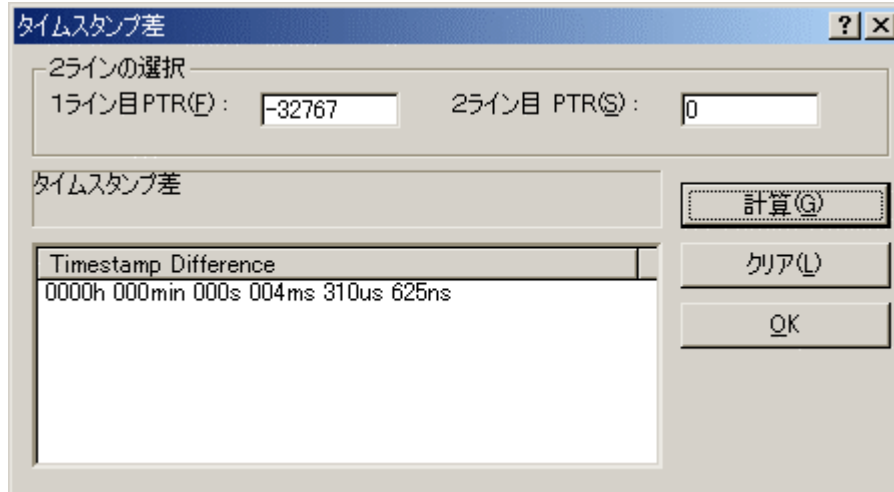


図3.32 タイムスタンプ差ダイアログボックス

- | | |
|-------------|--|
| [2ラインの選択] | タイムスタンプの差を計算するトレースレコードを指定します。 |
| [1ライン目 PTR] | 差分を計測する1点目のポイントを指定します。
デフォルト値は、[トレース]ウィンドウ上で選択されているラインのポイントが表示されます。 |
| [2ライン目 PTR] | 差分を計測する2点目のポイントを指定します。 |
| [タイムスタンプ差] | 2点間の時間差を計算した結果を表示します。 |
| [計算] | 指定した2点間の差分を計算し、結果を[タイムスタンプ差]リストに表示します。 |
| [クリア] | [タイムスタンプ差]リストのすべての結果を消去します。 |
| [OK] | ダイアログボックスを閉じます。
このとき、[タイムスタンプ差]リストのすべての結果は消去されます。 |

3.8.13 統計情報を解析する

指定された条件で統計情報の解析を実行するには、ポップアップメニューから[統計...]を選択します。
[統計]ダイアログボックスが開きます。

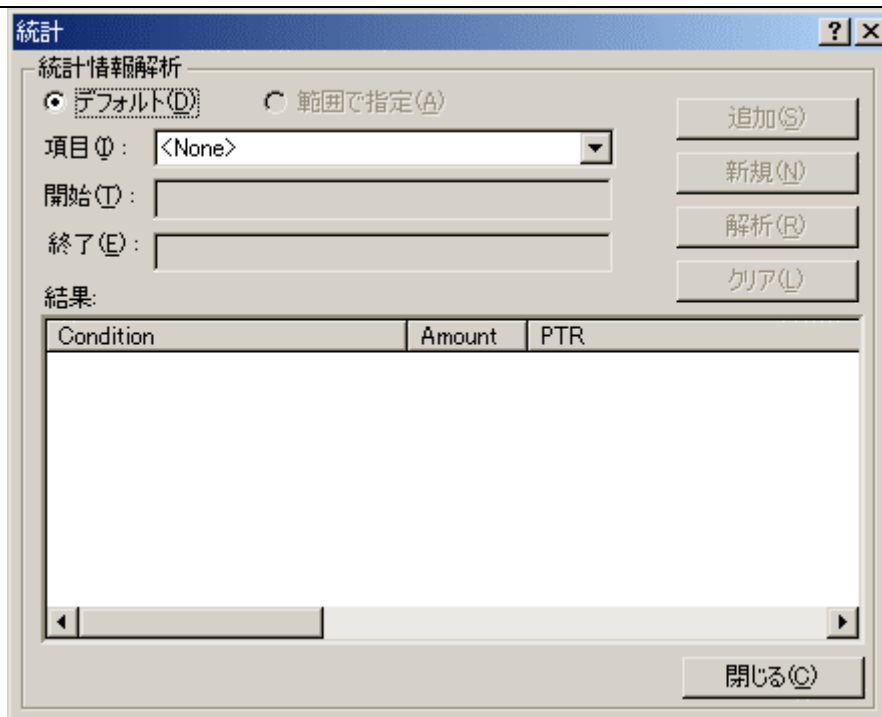


図3.33 統計ダイアログボックス

[統計情報解析]	統計情報を解析するための設定を行います。
[デフォルト]	単一の入力値または文字列を指定します。
[範囲で指定]	入力値または文字列を範囲で指定します。
[項目]	解析対象項目を指定します。
[開始]	入力値または文字列を指定します。 範囲で指定する場合は開始値を設定します。
[終了]	範囲で指定する場合は終了値を設定します。 ([範囲で指定] 選択時有効)
[追加]	現在の条件に追加設定します。
[新規]	新しい条件を指定します。
[解析]	統計情報解析の結果を取得します。
[結果]	すべての条件と統計情報解析結果を削除します。
[閉じる]	ダイアログボックスを閉じます。 このとき、[結果]リストのすべての結果は消去されます。

本ダイアログボックスは、トレース情報の統計情報解析に使用します。[項目]オプションで解析対象項目を指定し、[開始]オプションおよび[終了]オプションで入力値または文字列を指定します。[新規]ボタンまたは[追加]ボタンにより条件を設定し[解析]ボタンをクリックすると、統計情報を解析し[結果]リストに解析結果を表示します。

【注】 本エミュレータでは[PTR]項目のみ範囲で指定可能です。それ以外の項目は単一の文字列で指定してください。
統計情報の解析における文字列の判定は[トレース]ウィンドウに表示される文字列と比較し、完全一致したものだけをカウントします。ただし、大文字小文字は区別しません。また、空白の数も考慮しません。

3.8.14 取得したトレース情報から関数呼び出し箇所を抽出する

取得したトレース情報から関数呼び出し箇所のみを抽出するには、ポップアップメニューから[関数コール...]を選択します。

[関数コール箇所の表示]ダイアログボックスが開きます。



図3.34 関数コール箇所の表示ダイアログボックス

- | | |
|------|--------------------------|
| [設定] | 関数呼び出し箇所の抽出を行うかどうか設定します。 |
| [許可] | 関数呼び出し箇所の抽出を行います。 |
| [無効] | 関数呼び出し箇所の抽出を行いません。 |

[許可]オプションを選択した場合、取得したトレース情報より関数呼び出しを行っているサイクルのみを抽出し表示します。関数呼び出し箇所の抽出を行ってもトレースバッファの内容は変更されません。

フリートレースのトレース結果または、関数の呼び出しを含んだトレース情報に対して本機能を使用することにより、関数の呼び出し順序を調べることができます。

3.9 パフォーマンスを測定する

ユーザプログラムの実行効率を測定するにはパフォーマンス解析機能を使用します。

パフォーマンス解析機能はE6000Hエミュレータのハードウェアパフォーマンス測定回路により指定範囲の実行効率を測定するため、リアルタイム性は損なわれません。

実行効率測定の設定は測定用途に応じて下記5つのモードより選択できます。

表3.1 実行効率測定条件で設定可能な測定モード

測定モード	測定内容	測定用途
指定範囲内時間測定	指定した範囲の実行時間および実行回数を測定します。	関数の処理時間のうち関数内から呼び出す子関数の処理時間を除いた処理時間を測定する場合などに使用します。
指定アドレス間時間測定	指定したアドレス間の実行時間および実行回数を測定します。	関数の処理時間を測定する場合などに使用します。
指定アドレス範囲間時間測定	指定範囲から別の指定範囲までの実行時間を測定します。	アセンブリプログラムなど、サブルーチンを連続して配置したプログラムにおいて、連続するサブルーチンのうちのいずれかが呼ばれてから別の連続するサブルーチンのいずれかが呼ばれるまでの実行時間を測定する場合などに使用します。
領域アクセス回数測定	指定した範囲から別の指定した範囲に対するアクセス回数を測定します。	特定の関数からのグローバル変数に対するアクセス回数を測定する場合などに使用します。
指定範囲内コール回数測定	指定した範囲から別の指定した範囲をコールした回数を測定します。	特定の関数からの関数コール回数を測定する場合などに使用します。

実行効率測定の設定はE6000Hハードウェアパフォーマンス測定回路に実装されたパフォーマンスチャネル(8チャネル)を使用し、最大8ポイントまで設定できます。

ただし、指定アドレス範囲間時間測定、領域アクセス回数測定、指定範囲内コール回数測定の設定では1条件設定あたり連続した2ポイントを使用するため、これらの測定モードを使用する場合、設定可能ポイント数は最大4ポイントとなります。


表3.2 実行効率測定条件のモード設定

測定モード	ポイント							
	1	2	3	4	5	6	7	8
指定範囲内時間測定								
指定アドレス間時間測定								
指定アドレス範囲間時間測定		x		x		x		x
領域アクセス回数測定		x		x		x		x
指定範囲内コール回数測定		x		x		x		x

: 設定可, x: 設定不可

【注】 指定範囲内時間測定および指定アドレス間時間測定は1ポイントを使用し、指定アドレス範囲間時間測定・領域アクセス回数測定および指定範囲内コール回数測定は連続した2ポイントを使用します。2ポイントを使用するモードから1ポイントを使用するモードに変更したとき、また、1ポイントを使用するモードから2ポイントを使用するモードに変更したとき、設定されていた条件は削除されます。

3.9.1 パフォーマンス解析ウィンドウを開く

[パフォーマンス解析]ウィンドウを開くには、[表示 -> パフォーマンス -> パフォーマンス解析]を選択するか、[パフォーマンス解析]ツールバーボタンをクリックして[パフォーマンス解析方式の選択]ダイアログボックスを開きます。

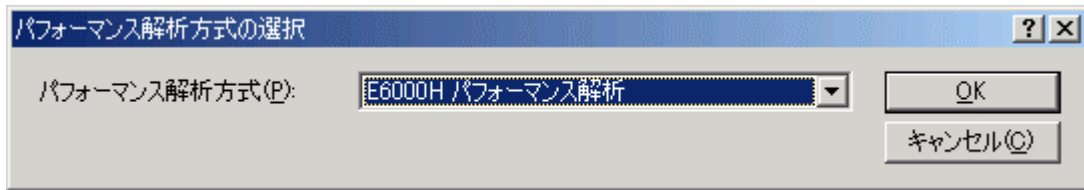


図3.35 パフォーマンス解析方式の選択ダイアログボックス

[E6000H パフォーマンス解析]を選択し[OK]ボタンをクリックすると[パフォーマンス解析]ウィンドウが開きます。

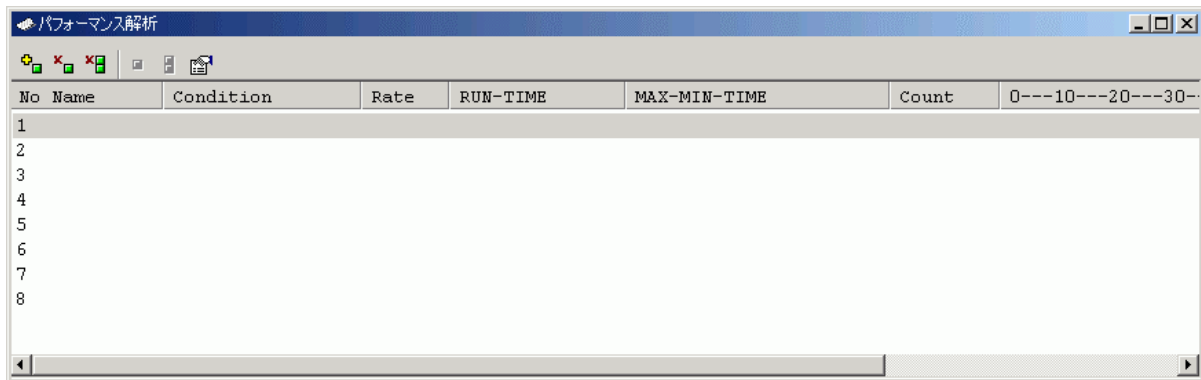


図3.36 パフォーマンス解析ウィンドウ

直前のプログラム実行でユーザが選択したエリアの実行時間比率をパーセント、ヒストグラムおよび数値で表示します。

[パフォーマンス解析]ウィンドウ内の不要なカラムは非表示にすることができます。

カラムを非表示にする場合はヘッダカラム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より非表示にしたいカラムを選択してください。

カラムを再表示する場合は再度ポップアップメニューより該当のカラムを選択してください。

3.9.2 実行効率測定条件を設定する

[パフォーマンス解析]ウィンドウでは測定条件の設定内容の表示および変更ができます。

条件を設定するポイントを選択し、ポップアップメニューから[設定...]を選択すると[パフォーマンス解析プロパティ]ダイアログボックスを表示します。

実行効率測定条件は[測定モード]オプションにより5つのモードから選択可能です。

表3.3 実行効率測定条件

	測定モード
1	指定範囲内時間測定
2	指定アドレス間時間測定
3	指定アドレス範囲間時間測定
4	領域アクセス回数測定
5	指定範囲内コール回数測定

使用するモードに応じて実行効率測定条件を設定します。

各モードにより設定するパラメータが異なります。

[パフォーマンス解析プロパティ]ダイアログボックスにはアドレス範囲指定時、関数名を入力することにより関数のアドレス範囲を自動的に入力する入力補助機能があります。

[パフォーマンス解析プロパティ]ダイアログボックス上にある[...]ボタンをクリックすると表示される[関数範囲入力]ダイアログボックスより、関数名を指定すると自動的に関数のアドレス範囲を入力することができます。



図3.37 関数範囲入力ダイアログボックス

- 【注】
1. 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が開きますので、関数を選択してください。
 2. 算出されるアドレスは参考値です。場合により関数の終了アドレスが異なる場合があります。[逆アセンブリ]ウィンドウにより関数の最終命令を確認し、[終了アドレス]の設定値を最終命令のアドレスに補正してください。
(一般的に関数の最終命令は RTS 命令となります。)
アドレス入力ではアドレス値以外にラベル名および式の指定も可能です。

3 デバッグ

(1) 指定範囲内時間測定

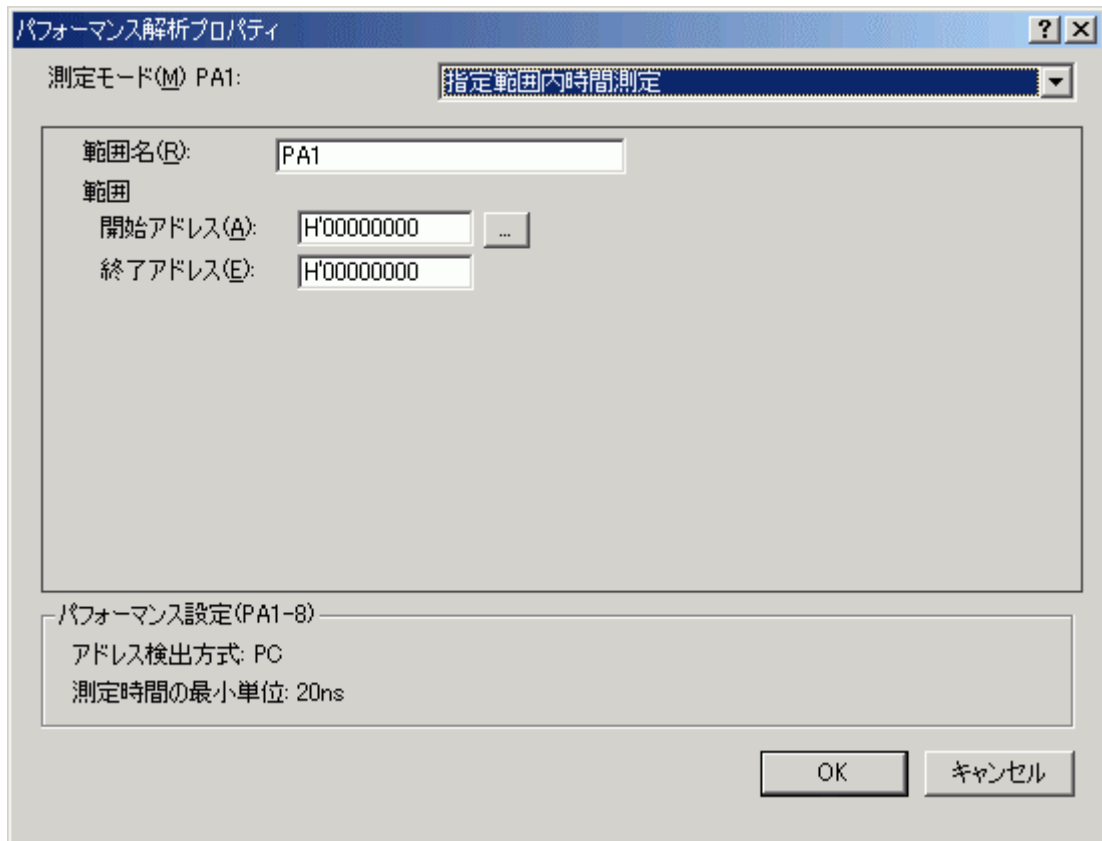


図3.38 指定範囲内時間測定

- [範囲名] 範囲の名称を指定します。
[範囲] 指定範囲内時間測定を行う範囲を指定します。
[開始アドレス] 開始アドレスを指定します。
[終了アドレス] 終了アドレスを指定します。

<開始アドレス>と<終了アドレス>で設定された範囲の実行時間および実行回数を測定します。
測定時間は<開始アドレス>と<終了アドレス>の範囲のプログラムプリフェッチで測定を開始し、範囲以外のプログラムプリフェッチで測定を中断します。再度、設定範囲のプログラムプリフェッチで測定を開始します。
実行回数は、設定範囲の<終了アドレス>のプログラムをフェッチするたびにカウントします。
測定結果には、設定範囲内から呼び出された処理の実行時間は含まれません。

(2) 指定アドレス間時間測定

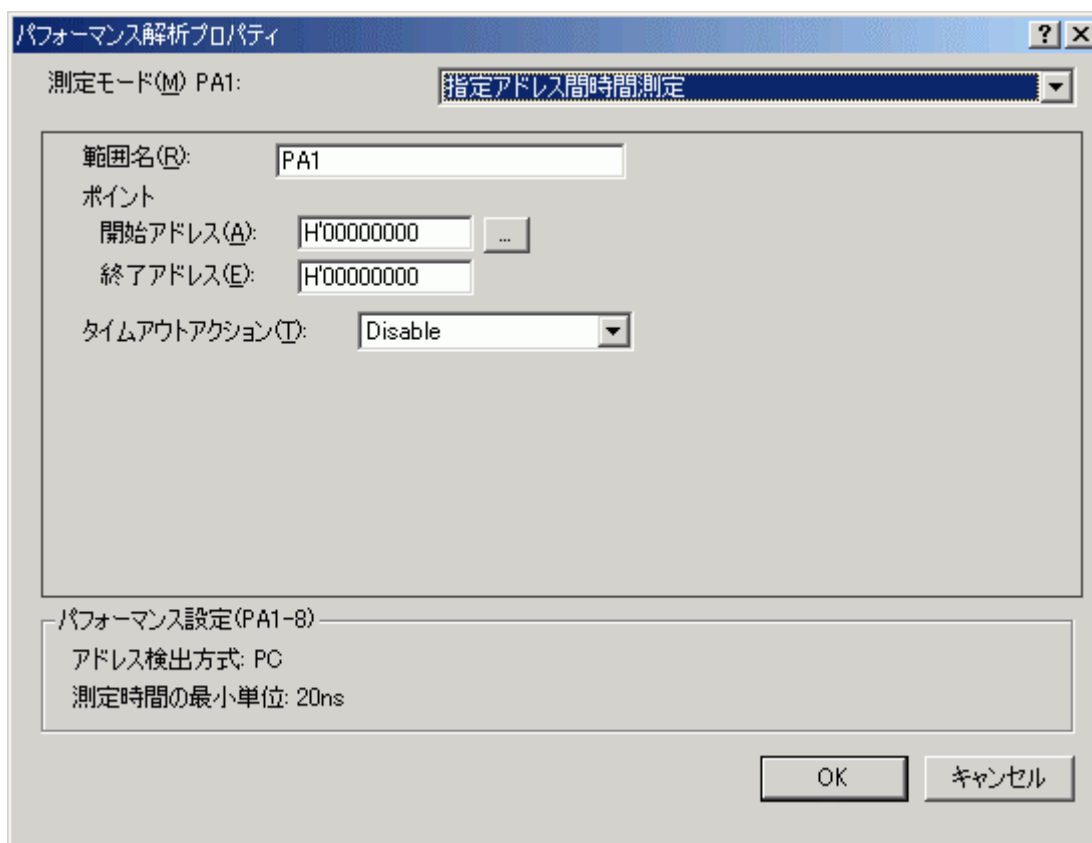


図3.39 指定アドレス間時間測定

[範囲名]	範囲の名称を指定します。
[ポイント]	指定アドレス間時間測定を行う範囲を指定します。
[開始アドレス]	開始アドレスを指定します。
[終了アドレス]	終了アドレスを指定します。
[タイムアウトアクション]	実行時間測定タイムアウトまたはカウントアウトが発生した時の動作を指定します。 無効:実行時間測定タイムアウトおよびカウントアウト値の設定を禁止します。 実行停止:実行時間測定タイムアウトまたはカウントアウト発生時、ユーザプログラムを停止します。 トレース停止:実行時間測定タイムアウトまたはカウントアウト発生時、トレース取得を停止します。 (チャンネル1のみ設定可能)
[タイムアウト値]	実行時間測定タイムアウト値を指定します。 測定時間の最小単位が160ns、40ns、20nsの指定は、 時 h 分 min 秒 s ミリ秒 ms マイクロ秒 us ナノ秒 ns(例: 1h 2min 3s 123ms 456us 789ns)で入力し、 ターゲットクロック指定は、16進で10桁(例: 123456789A)と入力します。 開始、終了アドレス間の1回ごとの計測値がタイムアウト値を超えたときにブレイクします (トータル時間ではありません)。 (チャンネル1のみ設定可能)
[回数]	実行回数測定カウントアップ値を指定します。 実行回数がカウントアップ値を超えたときにブレイクします。 (チャンネル1のみ設定可能)

<開始アドレス>と<終了アドレス>で設定された範囲の実行時間および実行回数を測定します。
測定時間は、<開始アドレス>のプログラムプリフェッチで測定を開始し、<終了アドレス>のプログラムプリフェッチで測定を中断します。
実行回数は、設定範囲の<終了アドレス>のプログラムをプリフェッチするたびにカウントします。
測定結果には、設定範囲内から呼び出された処理の実行時間を含みます。
ポイント1~4の場合に、設定された範囲の最大、最小時間を測定します。

3 デバッグ

(3) 指定アドレス範囲間時間測定

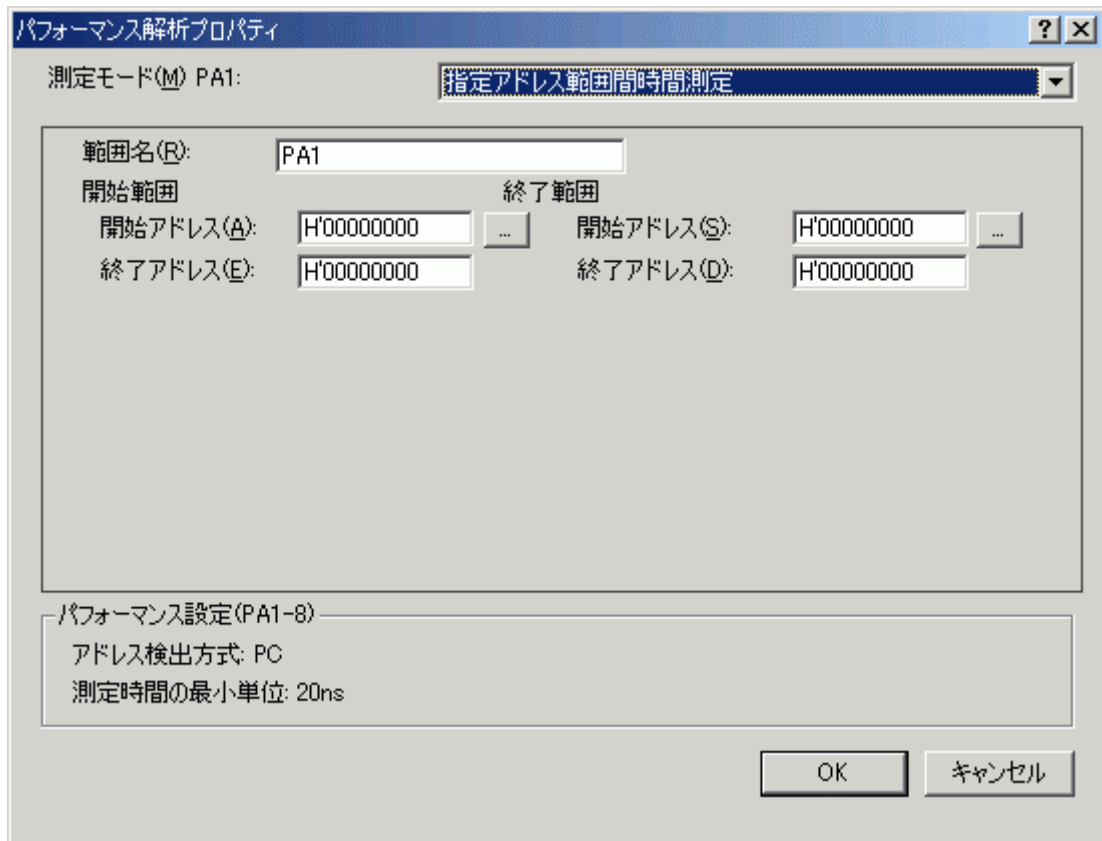


図3.40 指定アドレス範囲間時間測定

- | | |
|----------|-----------------------------|
| [範囲名] | 範囲の名称を指定します。 |
| [開始範囲] | 指定アドレス範囲間時間測定を行う開始範囲を指定します。 |
| [開始アドレス] | 先頭アドレスを指定します。 |
| [終了アドレス] | 最終アドレスを指定します。 |
| [終了範囲] | 指定アドレス範囲間時間測定を行う終了範囲を指定します。 |
| [開始アドレス] | 先頭アドレスを指定します。 |
| [終了アドレス] | 最終アドレスを指定します。 |

<開始アドレス範囲>のプリフェッチサイクルで時間測定を開始し、<終了アドレス範囲>のプログラムプリフェッチサイクルで測定を中断します。また、実行回数は、<終了アドレス範囲>を通過するたびにカウントアップします。

(4) 領域アクセス回数測定

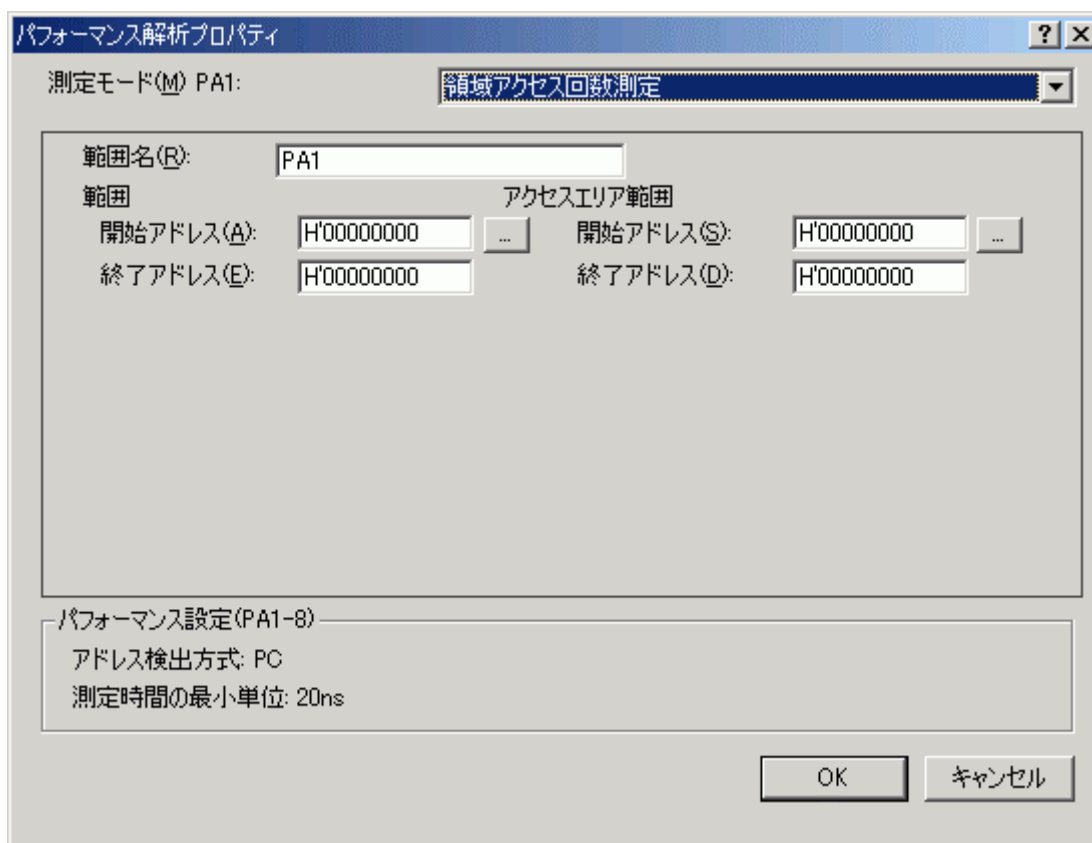


図3.41 領域アクセス回数測定

- | | |
|-------------|----------------------------------|
| [範囲名] | 範囲の名称を指定します。 |
| [範囲] | 領域アクセス回数測定を行う範囲を指定します。 |
| [開始アドレス] | 先頭アドレスを指定します。 |
| [終了アドレス] | 最終アドレスを指定します。 |
| [アクセスエリア範囲] | 領域アクセス回数測定を行うアクセス領域アドレス範囲を指定します。 |
| [開始アドレス] | 先頭アドレスを指定します。 |
| [終了アドレス] | 最終アドレスを指定します。 |

<開始アドレス>と<終了アドレス>で設定されている範囲から<アクセス領域アドレス範囲>で設定されている領域をアクセスした回数を測定します。また、範囲内の実行時間は、指定範囲内時間測定を用いて測定します。

3 デバッグ

(5) 指定範囲内コール回数測定

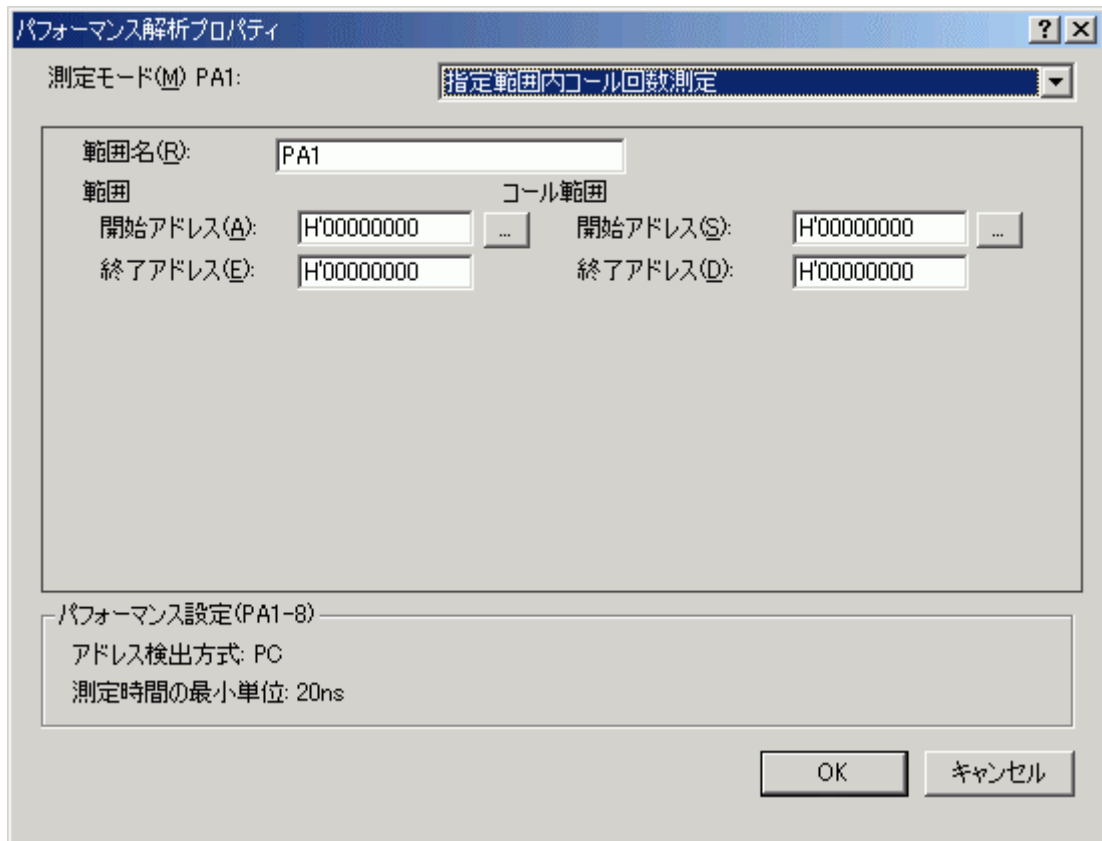


図3.42 指定範囲内コール回数測定

- | | |
|----------|--|
| [範囲名] | 範囲の名称を指定します。 |
| [範囲] | 指定範囲内コール回数測定を行う範囲を指定します。 |
| [開始アドレス] | 先頭アドレスを指定します。 |
| [終了アドレス] | 最終アドレスを指定します。 |
| [コール範囲] | 指定範囲内コール回数測定を行うコール範囲を指定します。
コール範囲は、指定サブルーチンの開始アドレスと終了アドレスを指定してください。 |
| [開始アドレス] | 先頭アドレスを指定します。 |
| [終了アドレス] | 最終アドレスを指定します。 |

<開始アドレス>、<終了アドレス>で設定されている範囲から<コール範囲>で設定されている範囲をコールした回数を測定します。また、範囲内の実行時間は、指定範囲内時間測定を用いて測定します。

<コール範囲>は、指定サブルーチンの開始アドレスと終了アドレスを指定してください。

3.9.3 実行効率測定を開始する

ユーザプログラムを実行すると前回の測定結果をクリアした後、設定した実行効率測定条件にしたがい自動的に実行効率測定を開始します。

ユーザプログラムを停止すると、測定結果を[パフォーマンス解析]ウィンドウに表示します。

3.9.4 測定条件を削除する

測定条件を選択した状態で、ポップアップメニューから[リセット]を選択すると、選択された測定条件を削除します。

3.9.5 すべての測定条件を削除する

ポップアップメニューから[全てリセット]を選択すると、設定している測定条件をすべて削除します。

3.10 プロファイル機能を使用する

3.10.1 プロファイルを有効にする

[表示->パフォーマンス->プロファイル]を選択し、[プロファイル]ウィンドウをオープンします。
[プロファイル]ウィンドウのポップアップメニューで[有効]メニューオプションを選択します（メニューにチェックマークが付きます）。

3.10.2 測定方法を指定する

プロファイルデータの測定時に、関数呼び出しをトレースするかどうかを指定できます。関数呼び出しをトレースすると、ユーザプログラム実行時の関数呼び出し関係をツリー形式で表示できるようになります。関数呼び出しをトレースしないと、関数呼び出し関係を表示できませんが、プロファイルデータの測定時間を短縮することができます。

関数呼び出しをトレースしないようにするためには、[プロファイル]ウィンドウのポップアップメニューで[関数呼び出しをトレースしない]を選択します（メニューにチェックマークが付きます）。

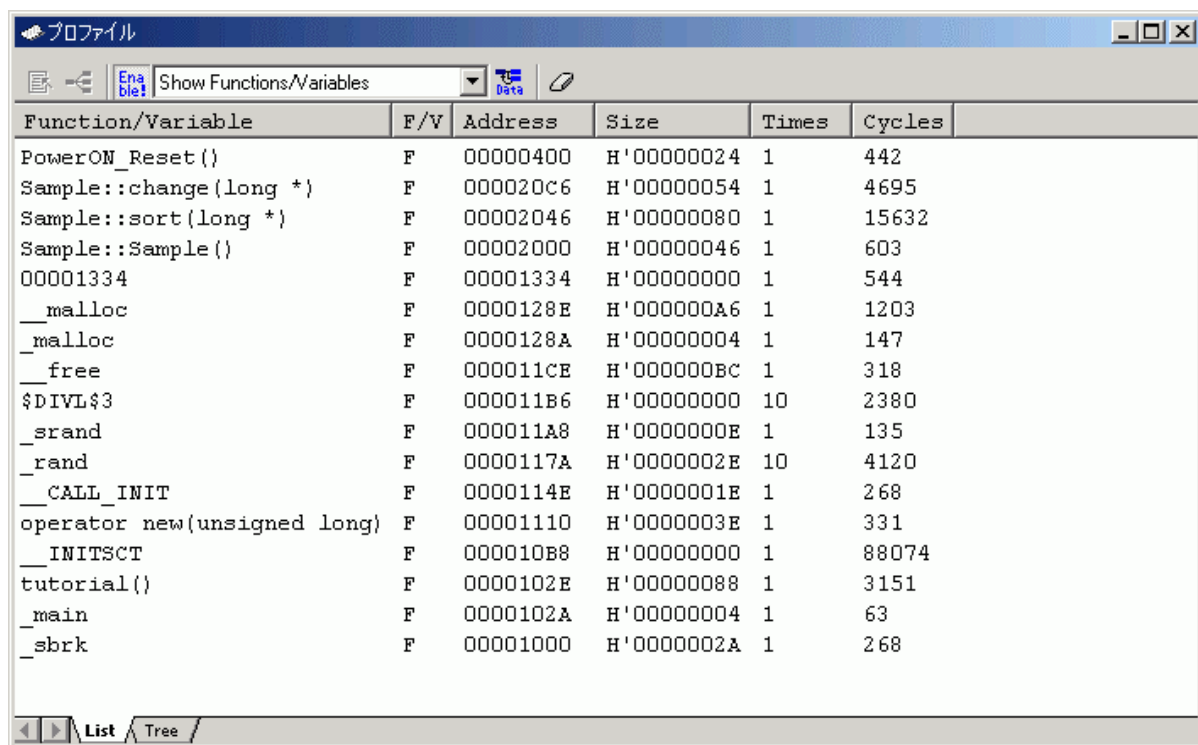
また、OSによるタスクスイッチなど、通常の方法以外で関数を呼び出しているプログラムの場合、関数呼び出しを正しく表示できない場合がありますので、関数呼び出しをトレースせずにプロファイルデータを測定してください。

3.10.3 ユーザプログラムを実行し結果を確認する

ユーザプログラムを実行し、停止すると[プロファイル]ウィンドウに測定結果を表示します。

[プロファイル]ウィンドウには、[List]シートと[Tree]シートがあります。

3.10.4 List シート



Function/Variable	F/V	Address	Size	Times	Cycles
PowerON_Reset()	F	00000400	H'00000024	1	442
Sample::change(long *)	F	000020c6	H'00000054	1	4695
Sample::sort(long *)	F	00002046	H'00000080	1	15632
Sample::Sample()	F	00002000	H'00000046	1	603
00001334	F	00001334	H'00000000	1	544
__malloc	F	0000128E	H'000000A6	1	1203
__malloc	F	0000128A	H'00000004	1	147
__free	F	000011CE	H'000000BC	1	318
\$DIVL\$3	F	000011B6	H'00000000	10	2380
_srand	F	000011A8	H'0000000E	1	135
_rand	F	0000117A	H'0000002E	10	4120
__CALL_INIT	F	0000114E	H'0000001E	1	268
operator new(unsigned long)	F	00001110	H'0000003E	1	331
__INITST	F	000010B8	H'00000000	1	88074
tutorial()	F	0000102E	H'00000088	1	3151
_main	F	0000102A	H'00000004	1	63
_sbrk	F	00001000	H'0000002A	1	268

図3.43 プロファイルウィンドウ (List シート)

本ウィンドウは、関数とグローバル変数のアドレス、サイズ、関数呼び出し回数および実行サイクル数を表示します。

カラムヘッダをクリックすると、アルファベットまたは数値の昇降順にソートして表示します。

[Function/Variable]列または[Address]列をダブルクリックすると、該当するアドレスに対応したソースプログラムを表示します。

3 デバッグ

ウィンドウ内でマウスの右ボタンをクリックするとポップアップメニューを表示します。このポップアップメニューは「3.10.5 Tree シート」を参照してください。

【注】1. プロファイル機能の注意事項については「5.9 プロファイル機能について」を参照してください。

3.10.5 Tree シート

Function	Address	Size	Stack Size	Times	Cycles
C:\Hew3\Tools\Renesas\DebugCo...					
PowerON_Reset()	00000400	H'00000024	H'00000000	1	442
__INIT\$CT	000010B8	H'00000000	H'00000000	1	88074
_srand	000011A8	H'0000000E	H'00000000	1	135
__CALL__INIT	0000114E	H'0000001E	H'00000000	1	268
main	0000102A	H'00000004	H'00000000	1	63
tutorial()	0000102E	H'00000088	H'00000000	1	3151
Sample::Sample()	00002000	H'00000046	H'00000000	1	603
Sample::change(long *)	000020C6	H'00000054	H'00000000	1	4695
Sample::sort(long *)	00002046	H'00000080	H'00000000	1	15632
_rand	0000117A	H'0000002E	H'00000000	10	4120

図3.44 プロファイルウィンドウ (Tree シート)

本ウィンドウは、関数の呼び出し関係をツリー構造で表示します。また、各関数のアドレス、サイズ、スタックサイズ、関数呼び出し回数および実行サイクル数を表示します。スタックサイズ、関数呼び出し回数は、実際の関数を呼び出した経路における値を表示します。

[Tree]シートは、[プロファイル]ウィンドウのポップアップメニューで[関数呼び出しをトレースしない]をチェックしていない時のみ有効です。

[Function]列の関数をダブルクリックすると、ツリー構造を拡張または収縮表示します。また、'+'/'-'キーでも拡張/収縮表示することができます。[Address]列をダブルクリックすると、該当するアドレスに対応したソースプログラムを表示します。

ウィンドウ内でマウスの右ボタンをクリックするとポップアップメニューを表示します。このメニューは以下のオプションを含みます。

3.10.6 ソースファイル表示

選択している行の該当アドレスに対応したソースプログラムを表示します。

3.10.7 チャート表示

選択している行の関数に着目した[プロファイル-チャートウィンドウ]を表示します。

3.10.8 有効

プロファイルデータ収集の有効/無効を切り替えます。プロファイルデータ測定が有効のとき、メニューテキストの左にチェックマークを表示します。

3.10.9 関数呼び出しをトレースしない

本メニューをチェックすると、プロファイルデータ測定時に関数呼び出しをトレースしません。たとえば、OS のタスクスイッチのように通常の方法以外で関数が呼び出されるプログラムのデータを測定する場合に使用します。

[プロファイル]ウィンドウの[Tree]シートで関数呼び出し関係を表示するためには、本メニューをチェックせずにプロファイルデータを測定してください。

また、測定結果のプロファイル情報ファイルを使用して、最適化リンケージエディタによる最適化を行う場合も、本メニューをチェックしないでください。

3.10.10 検索...

[Function]列の文字列を検索する[テキスト検索]ダイアログボックスを表示します。検索したい文字列をエディットボックスに入力し、[次を検索]ボタンまたは、ENTER キーを入力すると、検索を開始します。

3.10.11 データ検索...

[データ検索]ダイアログボックスを表示します。

[カラム]コンボボックスで検索カラムを、[検索データ]グループで検索方向を設定し、[次を検索]ボタンまたは、ENTER キーを入力すると、検索を開始します。また、連続して[次を検索]ボタンまたは、ENTER キーを入力すると、次に大きいデータ（最小値の場合は小さいデータ）を検索します。

3.10.12 データクリア

関数呼び出し回数のカウントおよびプロファイルデータをクリアします。[プロファイル]ウィンドウの[List]シートおよび[プロファイルチャート]ウィンドウのデータもクリアします。

3.10.13 プロファイル情報の保存...

[プロファイル情報の保存]ダイアログボックスを表示します。プロファイル結果をプロファイル情報ファイル（拡張子は".pro"）に保存します。最適化リンケージエディタは、プロファイル情報を元に、ユーザプログラムの最適化を行うことができます。プロファイル情報を使用した最適化についての詳細は、最適化リンケージエディタのマニュアルを参照してください。

【注】 [関数呼び出しをトレースしない]メニューをチェックして測定した結果のプロファイル情報では、最適化リンケージエディタによる最適化は行えません。

3.10.14 テキスト形式で保存...

[プロファイルデータをテキスト形式で保存]ダイアログボックスを表示します。表示している状態をテキストファイルに保存します。

3.10.15 表示設定

このメニューには下記サブメニューがあります。（以下の説明には[List]シートのみメニューも含まれます）

(1) 関数と変数を表示

[Function/Variable]列で、関数およびグローバル変数の両方表示します。

(2) 関数を表示

[Function/Variable]列で、関数のみを表示します。

(3) 変数を表示

[Function/Variable]列で、グローバル変数のみを表示します。

(4) 未実行関数を表示しない

実行した関数のみ表示することができます。最適化リンケージエディタが出力するスタック使用量情報ファイル（拡張子：sni）がロードモジュールと同一ディレクトリに存在しない場合、このチェックボックスの設定にかかわらず、実行関数のみ表示します。

(5) 子関数の実行結果を含んで表示

表示するプロファイルデータに、関数内で呼び出した子関数のプロファイルデータを含めるかどうかを設定します。

3.10.16 プロパティ...

SH7046 E6000H エミュレータではサポートしていません。

3.11 プロファイル-チャートウィンドウ

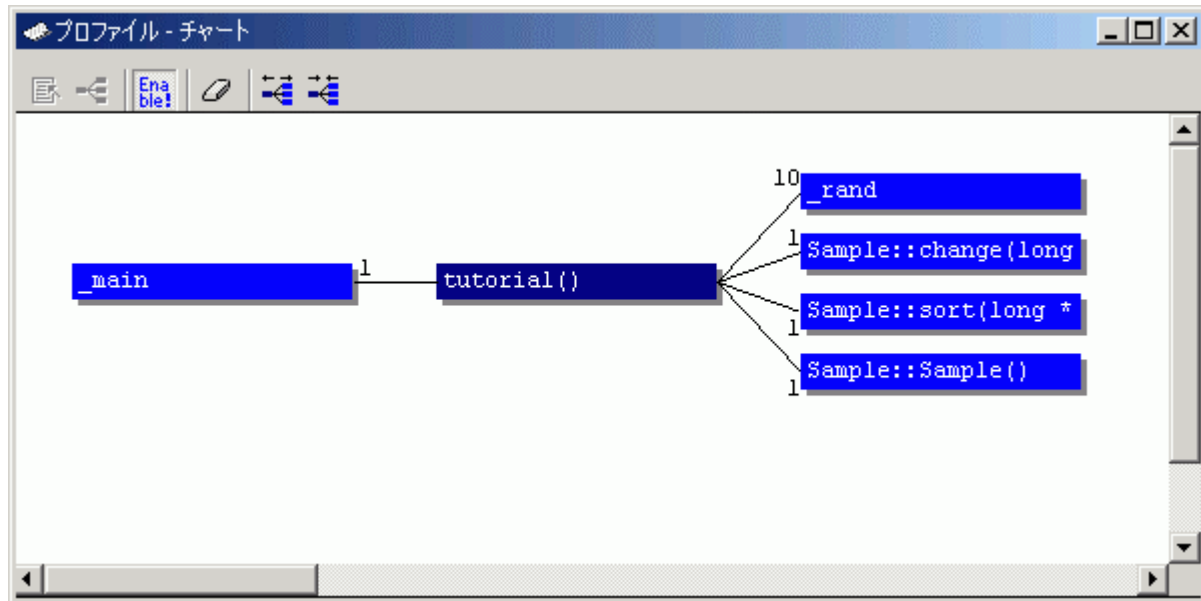


図3.45 プロファイル-チャートウィンドウ

ある関数に着目した呼び出し関係を表示します。[プロファイル-チャート]ウィンドウは、[プロファイル]ウィンドウの[List]シートまたは[Tree]シートから開きます。[プロファイル]ウィンドウの[List]シートまたは[Tree]シートで選択した関数に着目した呼び出し関係を表示します。着目した関数を中心に、左側に呼び出し元関数、右側に呼び出し先関数を表示します。呼び出し元関数および呼び出し先関数横の数値は、呼び出し回数を示します。

ウィンドウ内でマウスの右ボタンをクリックするとポップアップメニューを表示します。このメニューは以下のオプションを含みます。

3.11.1 ソースファイル表示

マウスの右ボタンをクリックしたときの位置にある関数の該当アドレスに対応したソースプログラムまたは逆アセンブルを表示します。マウスの右ボタンをクリックしたときの位置が関数ではない場合、このメニューオプションはグレー表示となります。

3.11.2 チャート表示

マウスの右ボタンをクリックしたときの位置にある関数に着目した[プロファイル-チャート]ウィンドウを表示します。マウスの右ボタンをクリックしたときの位置が関数ではない場合、このメニューオプションはグレー表示となります。

3.11.3 有効

プロファイルデータ収集の有効/無効を切り替えます。プロファイルデータ測定が有効のとき、メニューテキストの左にチェックマークを表示します。

3.11.4 データクリア

関数呼び出し回数のカウントをクリアします。[プロファイル]ウィンドウの[List]シートおよび[Tree]シートのデータもクリアします。

3.11.5 チャートウィンドウを複数開く

[プロファイル-チャート]ウィンドウを表示する際、既に Profile-Chart ウィンドウが開いているとき、別のウィンドウを開くか同ウィンドウに表示するかを設定します。メニューテキストの左にチェックマークを表示していれば、別のウィンドウを開きます。

3.11.6 プロファイル情報の保存...

[プロファイル情報の保存]ダイアログボックスを表示します。プロファイル結果をプロファイル情報ファイル(拡張子は".pro")に保存します。最適化リンケージエディタは、プロファイル情報を元に、ユーザプログラムの最適化を行うことができます。プロファイル情報を使用した最適化についての詳細は、最適化リンケージエディタのマニュアルを参照してください。

3.11.7 拡大

各関数の間隔を広げて表示します。また、'+' キーでも広げて表示することができます。

3.11.8 縮小

各関数の間隔を縮めて表示します。また、'-' キーでも縮めて表示することができます。

4. チュートリアル

4.1 はじめに

E6000H エミュレータの主な機能を紹介するために、チュートリアルプログラムを提供しています。このプログラムを用いて説明します。

このチュートリアルプログラムは、C++言語で書かれており、10個のランダムデータを昇順/降順にソートします。

チュートリアルプログラムでは、以下の処理を行います。

main 関数ではソート処理を繰り返し実行するため tutorial 関数の呼び出しを繰り返します。

tutorial 関数ではソートするランダムデータを生成し、sort 関数および change 関数を順に呼び出します。

sort 関数では tutorial 関数で生成したランダムデータを格納した配列を入力し、昇順にソートします。

change 関数では sort 関数で昇順にソートした配列を入力し、降順にソートします。

チュートリアルプログラムは、tutorial.cpp ファイルで提供しています。コンパイルされたロードモジュールは、Tutorial.abs ファイルとして Dwarf2 フォーマットで提供しています。

- 【注】
1. 再コンパイルを行った場合、本章で説明しているアドレスとずれることがあります。
 2. 本章は、一般的な E6000H エミュレータの使用例です。各製品の仕様については、「3 デバッグ」またはオンラインヘルプを参照してください。
 3. 各製品に添付される Tutorial.abs の動作アドレスは、製品によって異なります。本章で使用するアドレスについては、対応するソース行より各製品上で該当のアドレスを確認し、適宜読み替えて操作してください。
 4. 本チュートリアルでは SH7046 E6000H エミュレータを例に説明しています。ファイルのパス情報や、図の画面は各製品により異なりますので適宜読み替えて操作してください。

4.2 High-performance Embedded Workshop の起動

「2.1.3 既存のワークスペースを指定する場合」の手順に従ってワークスペースを開きます。

ディレクトリは以下を指定してください。

OS インストールドライブ\Workspace\Tutorial\E6000H\7046

【注】 ソフトウェアのバージョンによっては、上記ディレクトリを指定できない場合があります。その場合は以下のディレクトリを指定してください。

High-performance Embedded Workshop インストール先ディレクトリ
\Tools\Renesas\DebugComp\Platform\E6000H\7046\Tutorial

ファイルは以下を指定してください。

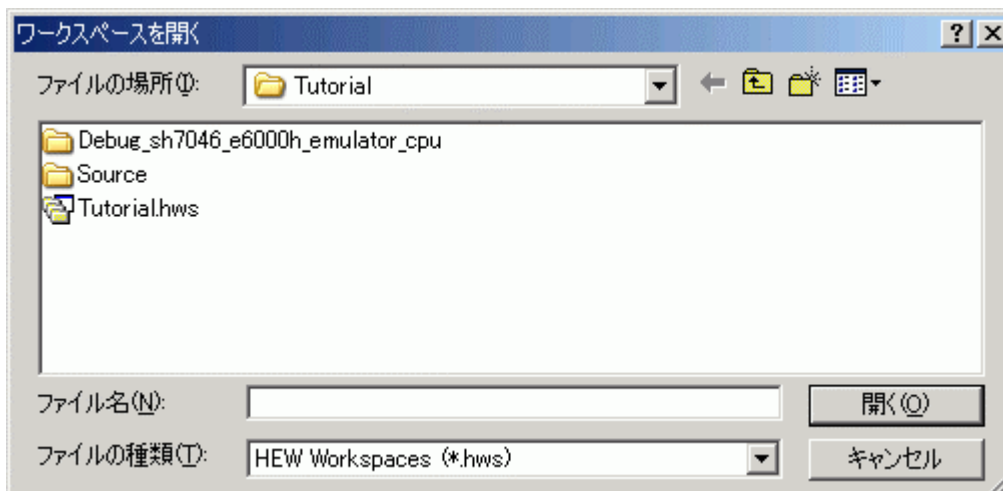


図4.1 ワークスペースを開くダイアログボックス

4.3 チュートリアルプログラムのダウンロード

4.3.1 チュートリアルプログラムをダウンロードする

デバッグしたいオブジェクトプログラムをダウンロードできます。
[Download modules]の[Tutorial.abs]から[ダウンロード]を選択します。

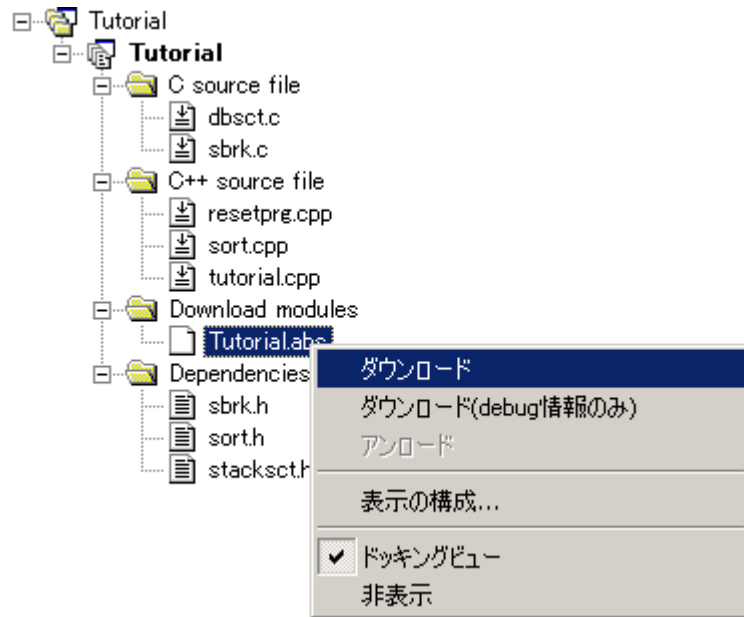


図4.2 チュートリアルプログラムのダウンロード

4.3.2 ソースプログラムを表示する

High-performance Embedded Workshop では、ソースレベルでプログラムをデバッグできます。

[C++ source file]の[tutorial.cpp]をダブルクリックします。

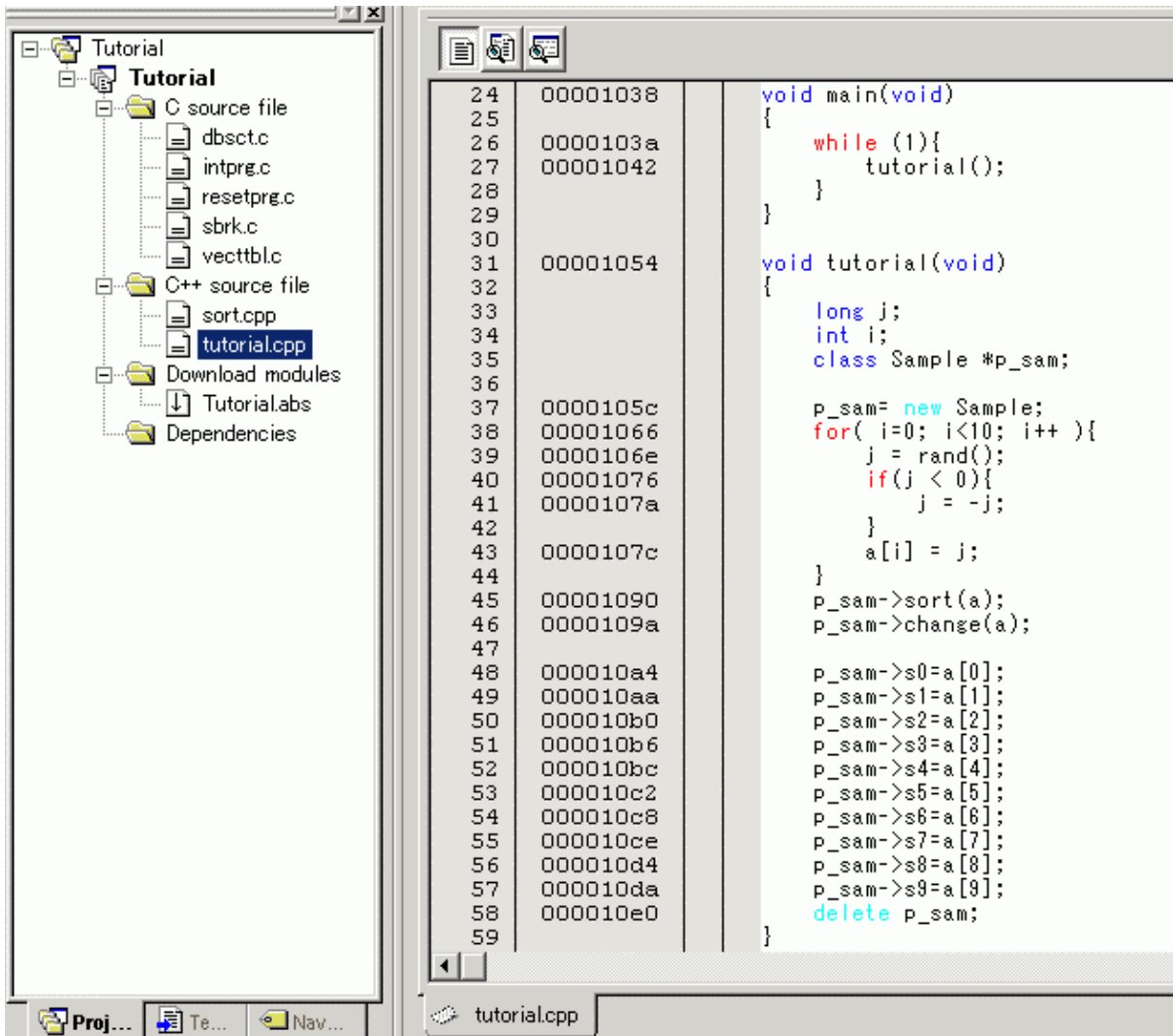


図4.3 [エディタ]ウィンドウ (ソースプログラムの表示)

必要であれば、見やすいフォントとサイズに変更することも可能です。変更方法については High-performance Embedded Workshop ユーザーズマニュアルを参照してください。

[エディタ]ウィンドウは、最初はプログラムの先頭を示しますが、スクロールバーを使って他の部分を見ることができます。

4.4 ソフトウェアブレークポイントの設定

簡単なデバッグ機能の1つにソフトウェアブレークポイントがあります。

[エディタ]ウィンドウにおいて、ソフトウェアブレークポイントを簡単に設定できます。たとえば、sort関数のコール箇所にソフトウェアブレークポイントを設定します。

sort関数コールを含む行の[S/W ブレークポイント]カラムをダブルクリックしてください。

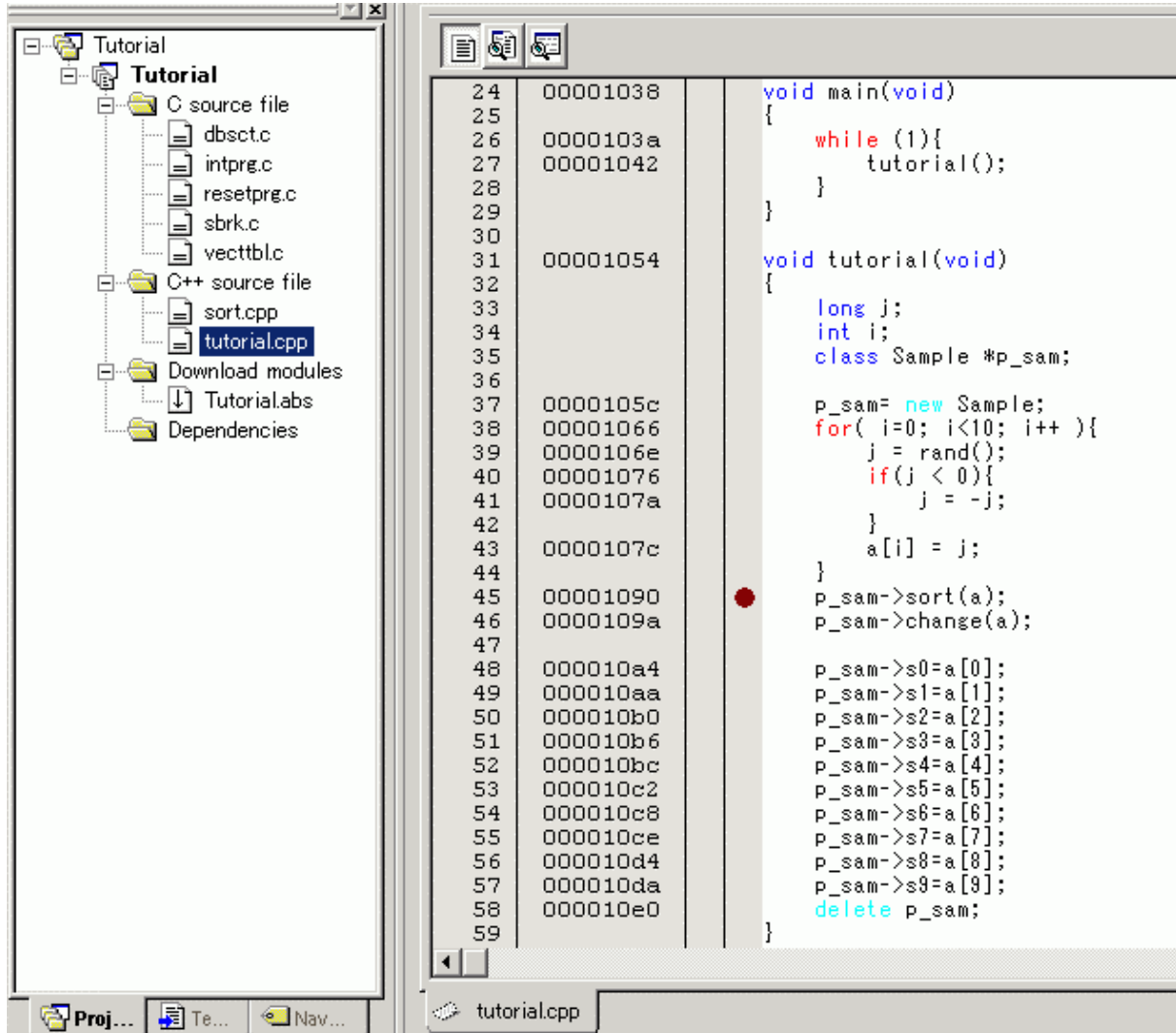



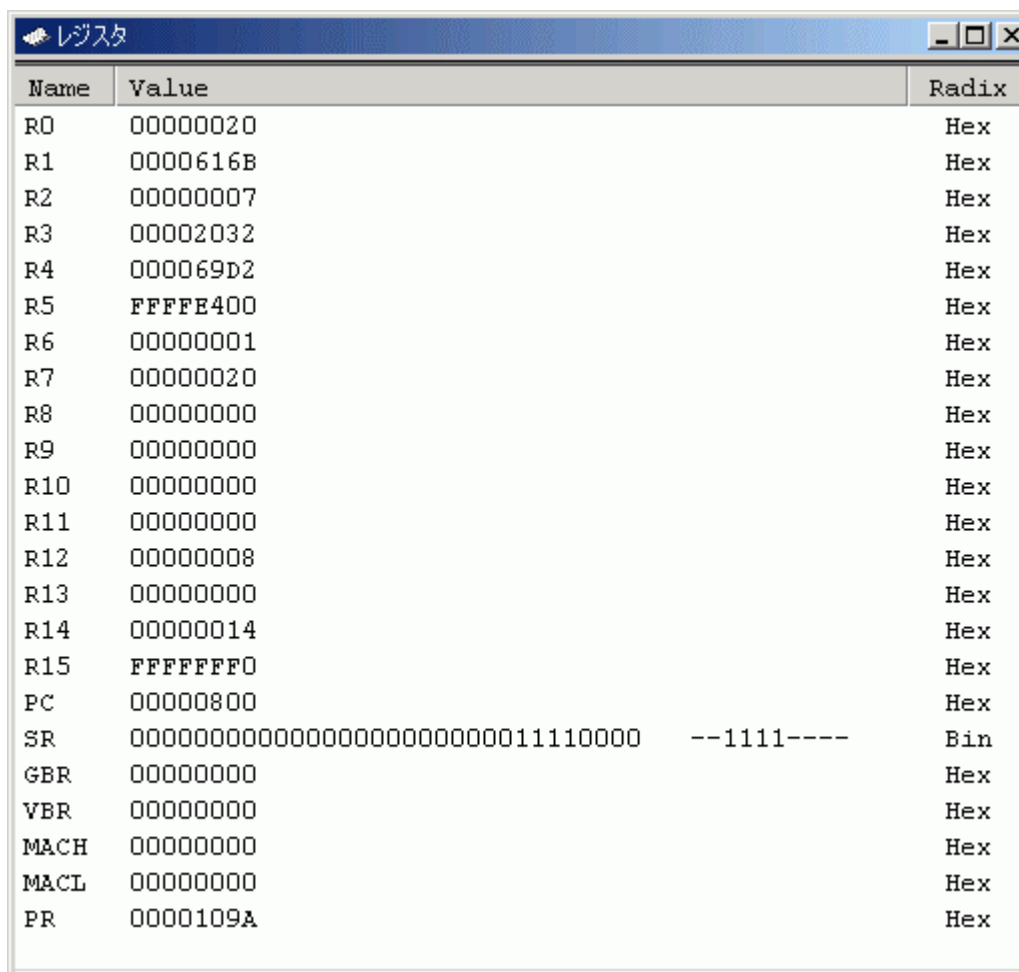
図4.4 [エディタ]ウィンドウ (ソフトウェアブレークポイントの設定)

sort関数を含む行に"•"と表示されます。この表示によりソフトウェアブレークポイントが設定されたことを示します。

4.5 レジスタ内容の変更

プログラムを実行する前に、プログラムカウンタの値を設定してください。

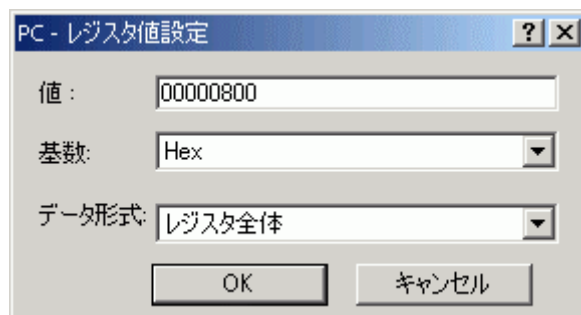
[表示 -> CPU -> レジスタ]を選択するか、[レジスタ]ツールバーボタンをクリックすると、[レジスタ]ウィンドウが表示されます。



Name	Value	Radix
R0	00000020	Hex
R1	0000616B	Hex
R2	00000007	Hex
R3	00002032	Hex
R4	000069D2	Hex
R5	FFFFE400	Hex
R6	00000001	Hex
R7	00000020	Hex
R8	00000000	Hex
R9	00000000	Hex
R10	00000000	Hex
R11	00000000	Hex
R12	00000008	Hex
R13	00000000	Hex
R14	00000014	Hex
R15	FFFFFFF0	Hex
PC	00000800	Hex
SR	000000000000000000000000000011110000	Bin
GBR	00000000	Hex
VBR	00000000	Hex
MACH	00000000	Hex
MACL	00000000	Hex
PR	0000109A	Hex

図4.5 レジスタウィンドウ

プログラムカウンタ(PC)を変更する場合には、[レジスタ]ウィンドウで[PC]の数値エリアをマウスでダブルクリックすると、以下のダイアログボックスが表示され、値の変更が可能です。本チュートリアルプログラムでは、H'00000800を設定し、[OK]ボタンをクリックしてください。



PC - レジスタ値設定

値:

基数:

データ形式:

OK キャンセル

図4.6 レジスタダイアログボックス(PC)

4.6 プログラムの実行

プログラムの実行方法について説明します。

プログラムを実行する場合は、[デバッグ]メニューから[実行]を選択するか、ツールバー上の[実行]ボタンを選択してください。



図4.7 実行ボタン

実行を開始すると、ステータスバーに現在のアドレスバスの値、およびターゲットマイコンの動作状態を表示します。

プログラムはソフトウェアブレークポイントを設定したところまで実行されます。プログラムが停止した位置を示すために[エディタ]ウィンドウの[S/W ブレークポイント]カラムに矢印が表示されます。また、[Break = Software Break]メッセージがステータスバーに表示されます。

【注】 ブレーク後にソースファイルを表示する際に、ソースファイルパスを問い合わせる場合があります。ソースファイルの場所は以下です。

OS インストールドライブ\Workspace\Tutorial\E6000H\7046\Source

ソフトウェアのバージョンによっては、上記ディレクトリを指定できない場合があります。その場合は以下のディレクトリを指定してください。

High-performance Embedded Workshop インストール先ディレクトリ
 \Tools\Renesas\DebugComp\Platform\E6000H\7046\Tutorial\Source

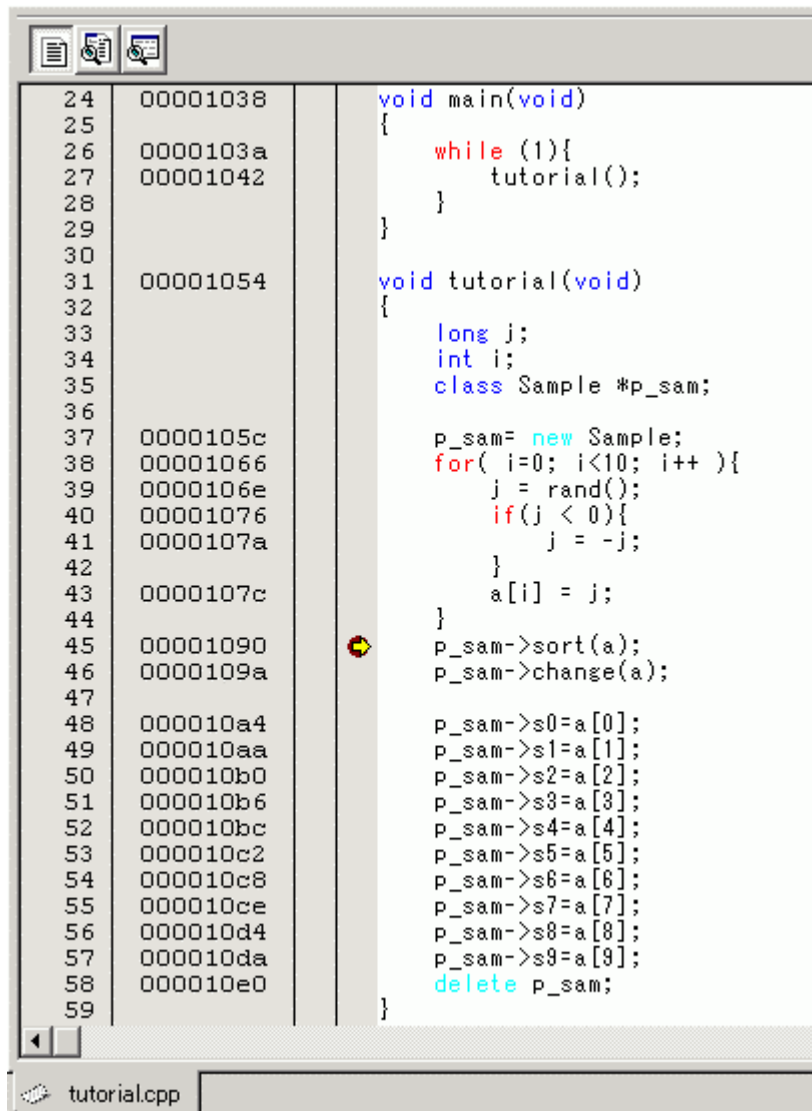



図4.8 [エディタ]ウィンドウ (ブレーク状態)

4 チュートリアル

[ステータス]ウィンドウで最後に発生したブレイクの要因が確認できます。

[表示 -> CPU -> ステータス]を選択するか、[ステータスの表示]ツールバーボタンをクリックしてください。
[ステータス]ウィンドウが表示されますので、[Platform]シートを開いて Cause of last break のステータスを確認してください。

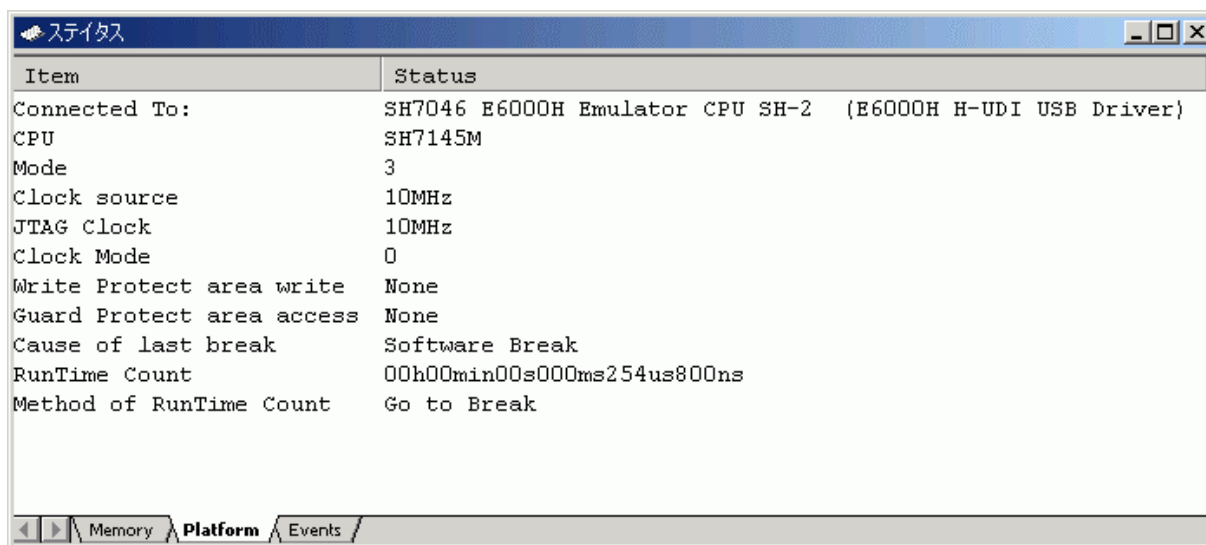



図4.9 ステータスウィンドウ

【注】 本ウィンドウで表示される内容は、製品ごとに異なります。各製品の表示内容については、「3 デバッグ」またはオンラインヘルプを参照してください。

4.7 ブレイクポイントの確認

設定したすべてのブレイクポイントは、[イベントポイント]ウィンドウで確認することができます。

[表示 -> コード -> イベントポイント]を選択するか、[イベントポイント]ツールバーボタンをクリックすると、[イベントポイント]ウィンドウが表示されます。[Software]シートを開きます。

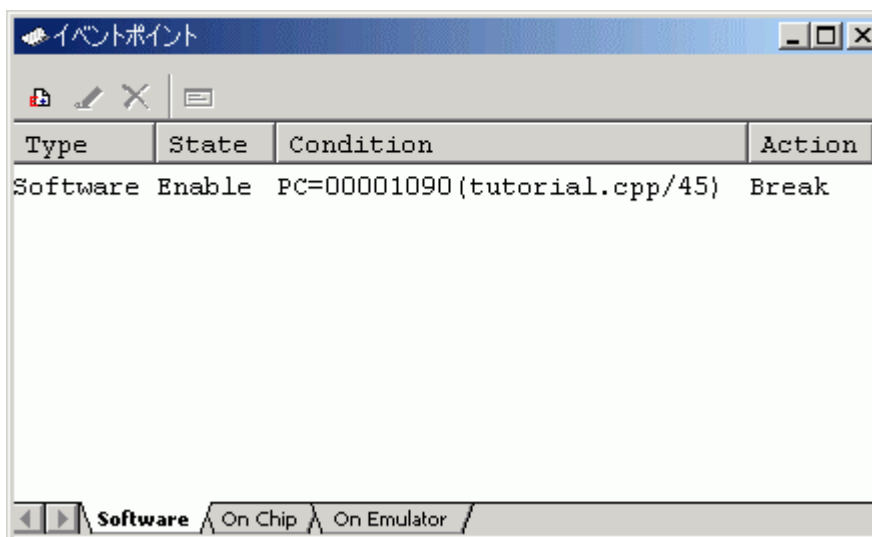


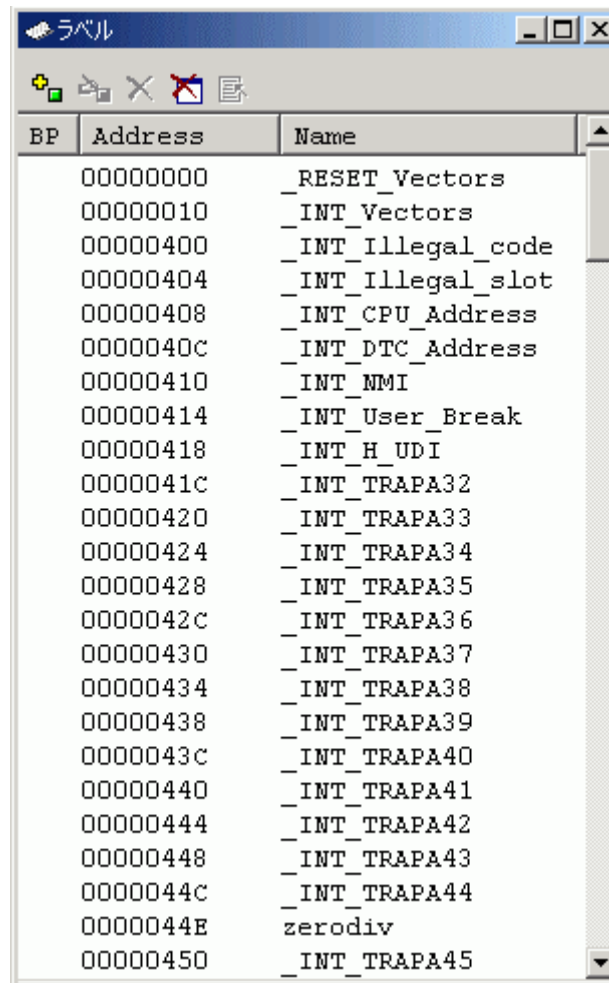
図4.10 イベントポイントウィンドウ

マウスの右ボタンで[イベントポイント]ウィンドウをクリックすると開くポップアップメニューにより、ブレイクポイントの設定 / 変更、新しいブレイクポイントの定義、およびブレイクポイントの削除、有効 / 無効の選択ができます。

4.8 シンボルの参照

[ラベル]ウィンドウを使ってモジュール内のシンボル情報を表示させることができます。

[表示]メニューから[シンボル]サブメニューを選択し、[ラベル]を選択してください。[ラベル]ウィンドウが表示され、モジュール内のシンボル情報が参照できます。




The screenshot shows a window titled 'ラベル' (Labels) with a toolbar and a table of symbols. The table has three columns: 'BP', 'Address', and 'Name'. The symbols listed are:

BP	Address	Name
	00000000	_RESET_Vectors
	00000010	_INT_Vectors
	00000400	_INT_Illegal_code
	00000404	_INT_Illegal_slot
	00000408	_INT_CPU_Address
	0000040C	_INT_DTC_Address
	00000410	_INT_NMI
	00000414	_INT_User_Break
	00000418	_INT_H_UDI
	0000041C	_INT_TRAPA32
	00000420	_INT_TRAPA33
	00000424	_INT_TRAPA34
	00000428	_INT_TRAPA35
	0000042C	_INT_TRAPA36
	00000430	_INT_TRAPA37
	00000434	_INT_TRAPA38
	00000438	_INT_TRAPA39
	0000043C	_INT_TRAPA40
	00000440	_INT_TRAPA41
	00000444	_INT_TRAPA42
	00000448	_INT_TRAPA43
	0000044C	_INT_TRAPA44
	0000044E	zerodiv
	00000450	_INT_TRAPA45

図4.11 ラベルウィンドウ

4.9 メモリ内容の確認

ラベル名を指定することによって、ラベルが登録されているメモリの内容を[メモリ]ウィンドウで確認することができます。たとえば、以下のように、バイトサイズで_mainに対応するメモリ内容を確認します。

[表示 -> CPU -> メモリ...]を選択するか、[メモリ]ツールバーボタンをクリックして、[表示開始アドレス]ダイアログボックスを開いてください。

[表示開始アドレス]エディットボックスに"_main"を入力してください。

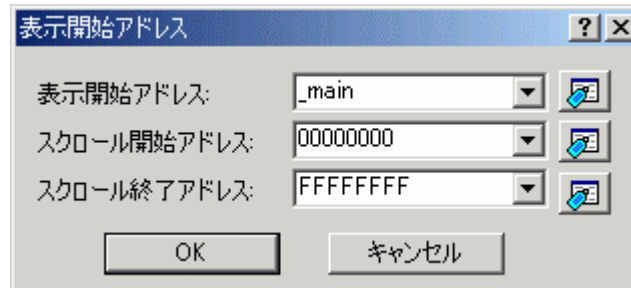


図4.12 表示開始アドレスダイアログボックス

[OK]ボタンをクリックしてください。指定されたメモリ領域を示す[メモリ]ウィンドウが表示されます。

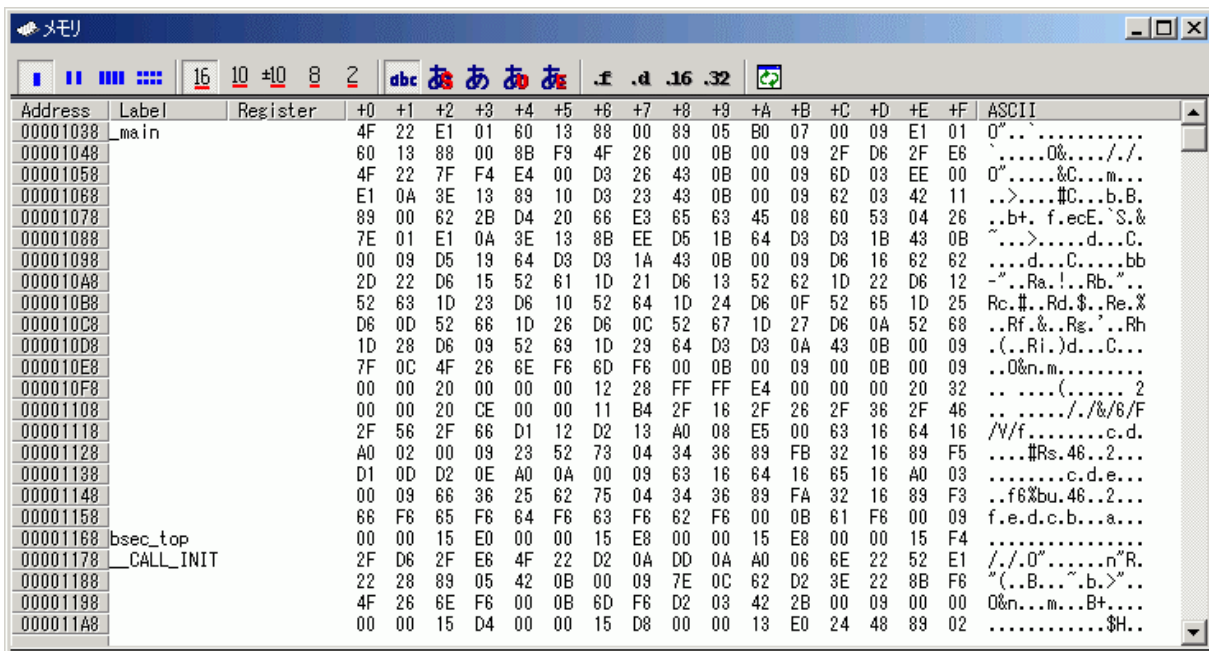


図4.13 メモリウィンドウ

4.10 変数の参照

プログラムをステップ処理するとき、プログラムで使われる変数の値が変化することを確認できます。たとえば、以下の手順で、プログラムの初めに宣言した long 型の配列 a を見ることができます。

[エディタ]ウィンドウに表示されている配列 a の左側をクリックし、カーソルを置いてください。マウスの右ボタンで[インスタントウォッチ]を選択してください。

以下のダイアログボックスが表示されます。

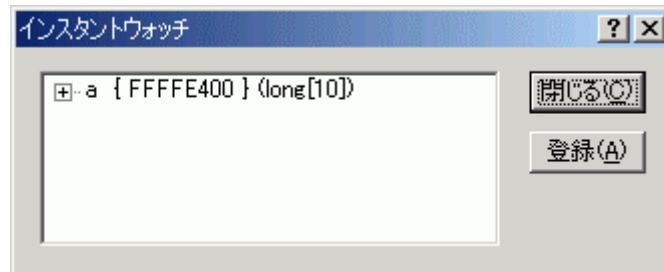


図4.14 インスタントウォッチダイアログボックス

[登録]ボタンをクリックして、[ウォッチ]ウィンドウに変数を加えてください。

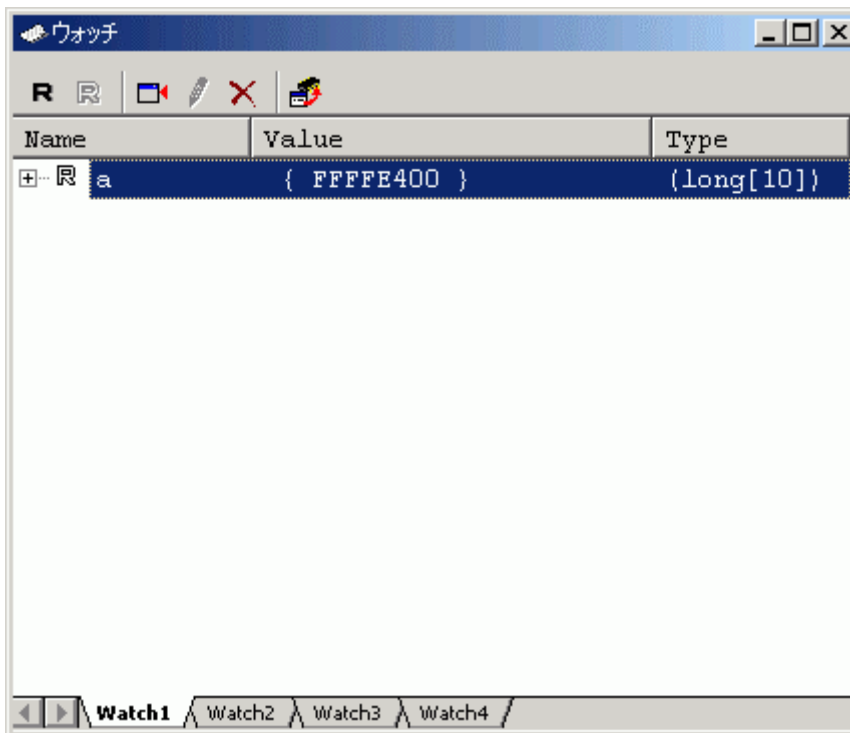


図4.15 ウォッチウィンドウ (配列の表示)

また、変数名を指定して、[ウォッチ]ウィンドウに変数を加えることもできます。マウスの右ボタンで[ウォッチ]ウィンドウをクリックし、ポップアップメニューから[シンボル登録]を選択してください。

4 チュートリアル

以下のダイアログボックスが表示されます。



図4.16 シンボル登録ダイアログボックス

[変数または式]エディットボックスに変数 `i` を入力し、[OK]ボタンをクリックします。

[ウォッチ]ウィンドウに、`int` 型の変数 `i` が表示されます。

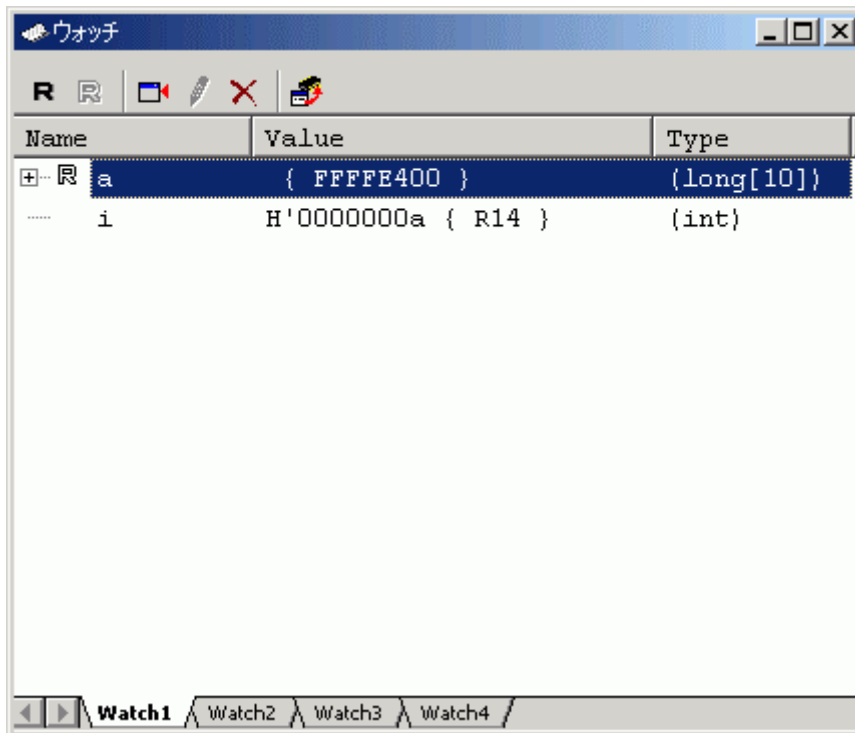


図4.17 ウォッチウィンドウ (変数の表示)

[ウォッチ]ウィンドウの配列 a の左側にある"+"マークをクリックし、配列 a の各要素を参照することができます。

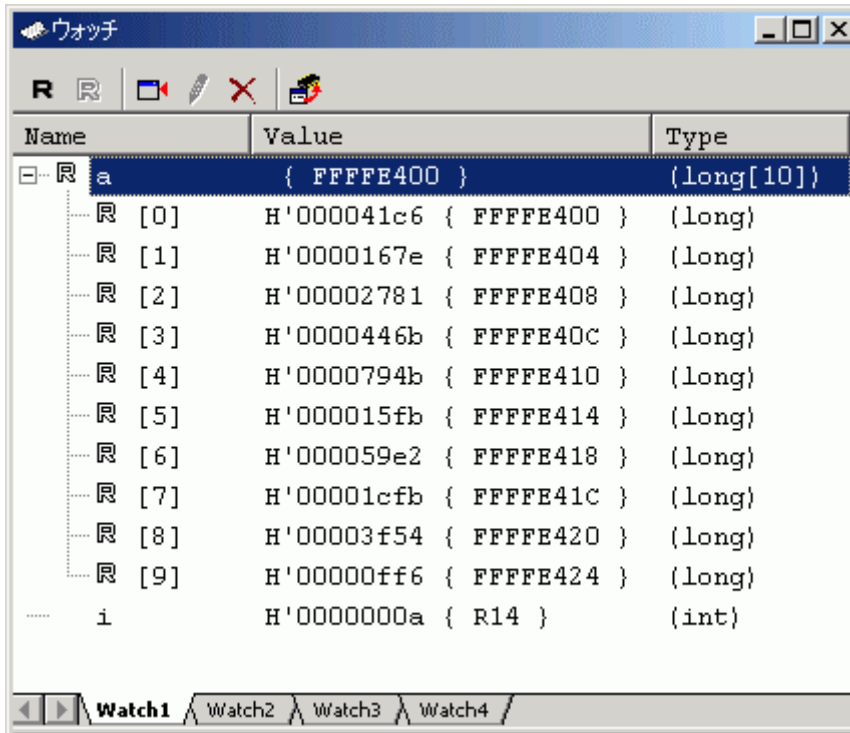



図4.18 ウォッチウィンドウ (配列要素の表示)

4.11 ローカル変数の表示

[ローカル]ウィンドウを使って関数内のローカル変数を表示させることができます。例として、tutorial 関数のローカル変数を調べます。

この関数は、3つのローカル変数 `j`、`i`、`p_sam` を宣言します。

[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  をクリックすると、[ローカル]ウィンドウが表示されます。

[ローカル]ウィンドウには、現在のプログラムカウンタ (PC) が指している関数のローカル変数とその値が表示されます。

関数内にローカル変数が存在しない場合、[ローカル]ウィンドウに何も表示されません。

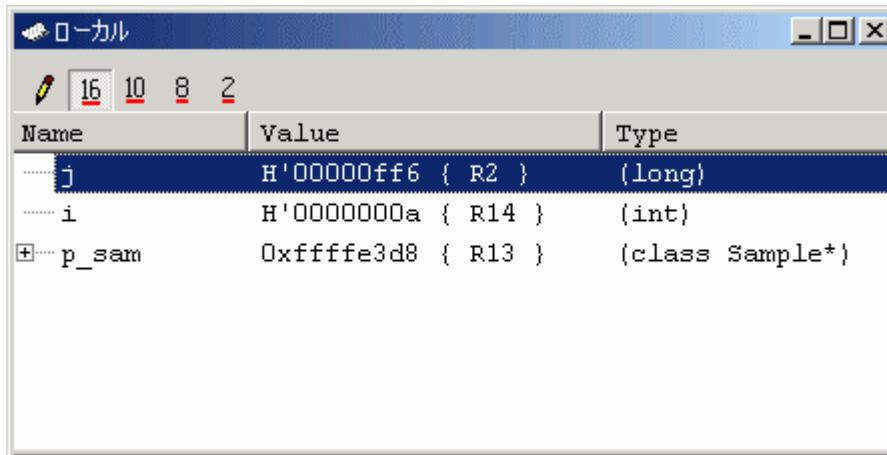


図4.19 ローカルウィンドウ

[ローカル]ウィンドウのクラスインスタンス `p_sam` の左側にある "+" マークをクリックし、クラスインスタンス `p_sam` の構成要素を表示させてください。

`sort` 関数実行前と実行後のクラスインスタンス `p_sam` の要素を参照し、ランダムデータが降順にソートされていることを確認してください。

4.12 プログラムのステップ実行

High-performance Embedded Workshop は、プログラムのデバッグに有効な各種のステップコマンドを備えています。

表4.1 ステップオプション

項番	コマンド	説明
1	ステップイン	各ステートメントを実行します (関数内のステートメントを含む)。
2	ステップオーバ	関数コールを1ステップとして、ステップ実行します。
3	ステップアウト	関数を抜け出し、関数を呼び出したプログラムの次のステートメントで停止します。
4	ステップ...	指定した速度で指定回数分ステップ実行します。

4.12.1 ステップインコマンドの実行

[ステップイン]コマンドはコール関数の中に入り、コール関数の先頭のステートメントで停止します。

sort 関数の中に入るために、[デバッグ]メニューから[ステップイン]を選択するか、またはツールバーの[ステップイン]ボタンをクリックしてください。



図4.20 ステップインボタン

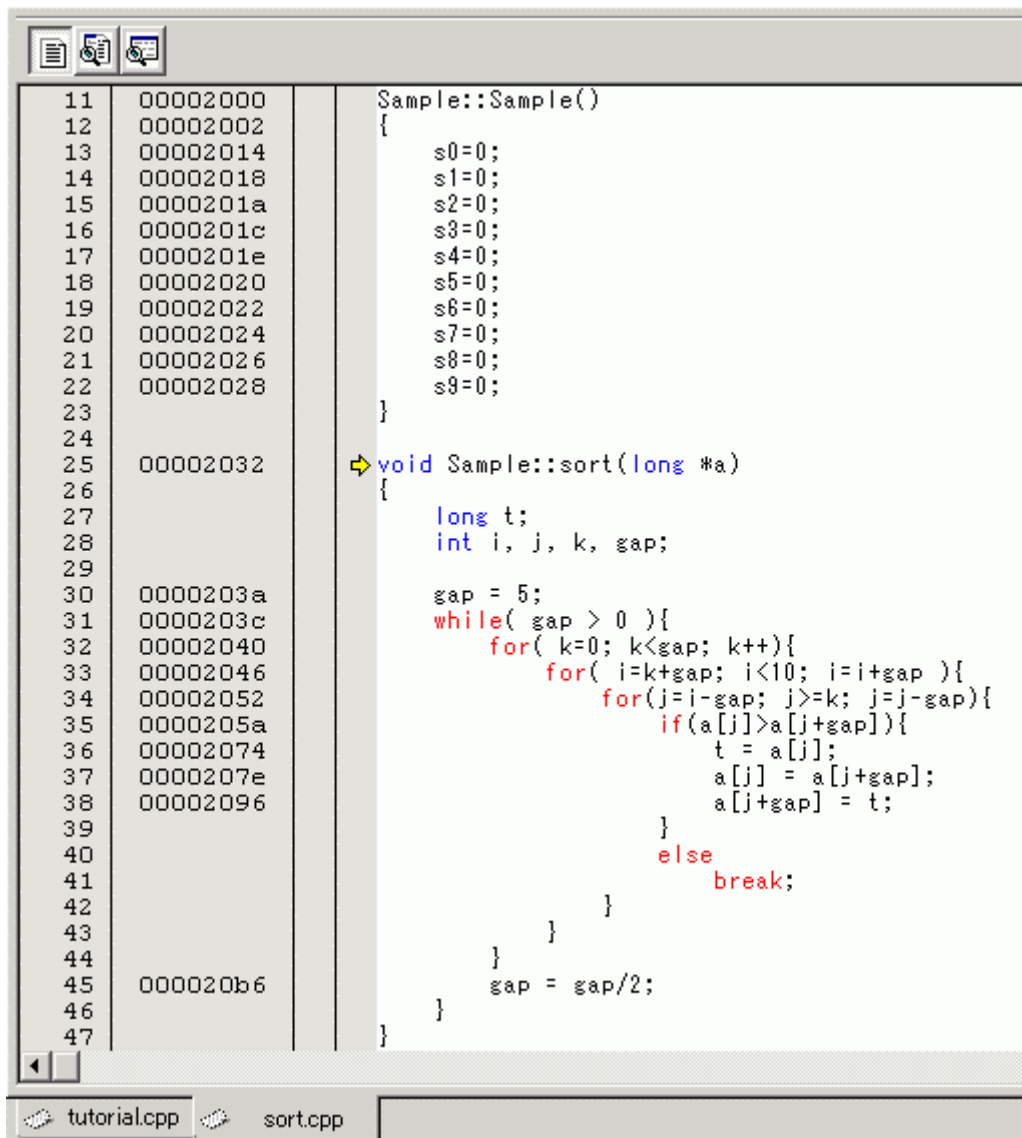


図4.21 [エディタ]ウィンドウ (ステップイン)

[エディタ]ウィンドウの強調表示が、sort 関数の先頭のステートメントに移動します。

4.12.2 ステップアウトコマンドの実行

[ステップアウト]コマンドはコール関数の中から抜け出し、コール元プログラムの次のステートメントで停止します。

sort 関数の中から抜け出すために、[デバッグ]メニューから[ステップアウト]を選択するか、またはツールバーの[ステップアウト]ボタンをクリックしてください。



図4.22 ステップアウトボタン

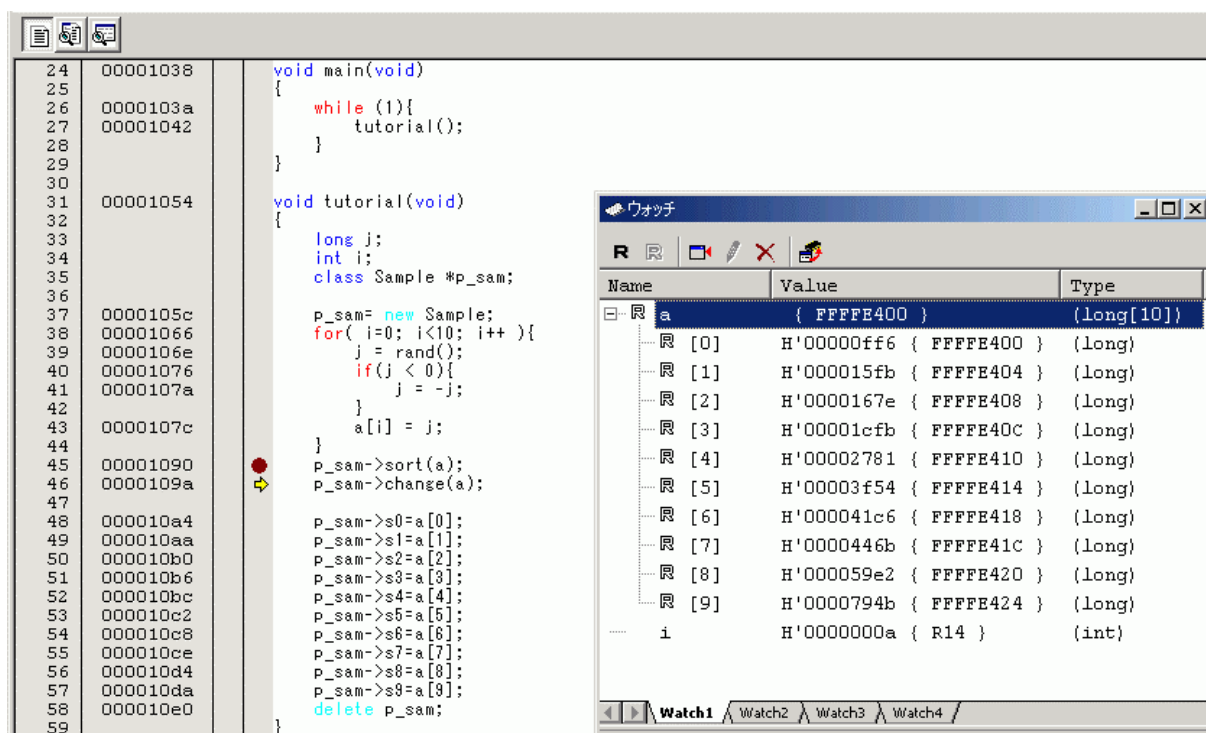


図4.23 High-performance Embedded Workshop ウィンドウ (ステップアウト)

[ウォッチ]ウィンドウに表示された変数 `a` のデータが昇順にソートされます。

4.12.3 ステップオーバーコマンドの実行

[ステップオーバー]コマンドは関数コールを1ステップとして実行して、メインプログラムの次のステートメントで停止します。

change 関数中のステートメントを一度にステップ実行するために、[デバッグ]メニューから[ステップオーバー]を選択するか、またはツールバーの[ステップオーバー]ボタンをクリックしてください。



図4.24 ステップオーバーボタン

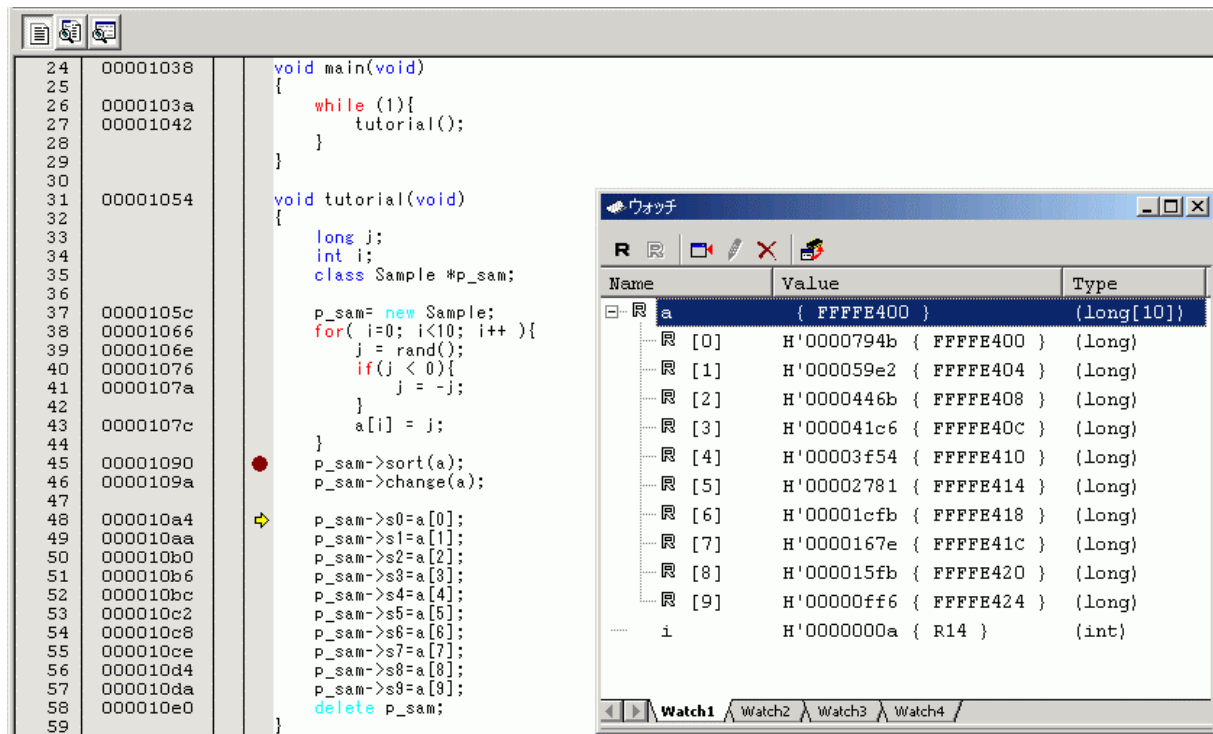


図4.25 High-performance Embedded Workshop ウィンドウ (ステップオーバー)

[ウォッチ]ウィンドウに表示された変数 `a` のデータが降順にソートされます。

4.13 プログラムの強制ブレーク

High-performance Embedded Workshop は、プログラムを強制的にブレークすることができます。

ブレークポイントをすべて解除してください。

tutorial 関数の残り部分を実行するために、[デバッグ]メニューから[実行]を選択するか、ツールバー上の[実行]ボタンを選択してください。



図4.26 実行ボタン

プログラムは無限ループ処理を実行していますので、強制ブレークするために、[デバッグ]メニューから[プログラムの停止]を選択するか、ツールバー上の[停止]ボタンを実行してください。



図4.27 停止ボタン

4.14 ターゲットマイコンのリセット

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。

ターゲットマイコンのリセットを行うには、[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。



図4.28 CPU リセットボタン

また、リセット直後からプログラムを実行する場合には、[デバッグ]メニューから[リセット後実行]を選択するか、ツールバー上の[リセット後実行]ボタンを選択してください。



図4.29 リセット後実行ボタン

【注】 本チュートリアルプログラムは、リセットベクタからプログラムを実行しても問題なく実行できるように作成しています。

4.15 ブレーク機能


E6000H エミュレータは、ソフトウェアブレーク機能、オンチップブレーク機能およびオンエミュレータブレーク機能をもっています。

ソフトウェアブレークポイントの設定、オンチップブレーク設定およびオンエミュレータブレーク設定は[イベントポイント]ウィンドウでそれぞれ行うことができます。

以下にブレーク機能の概要と設定方法について説明します。

4.15.1 ソフトウェアブレーク機能

E6000H エミュレータは、最大 255 ポイントまでソフトウェアブレークポイントを設定することができます。

[表示 -> コード -> イベントポイント]を選択するか、[イベントポイント]ツールバーボタンをクリックすると、[イベントポイント]ウィンドウが表示されます。[Software]シートを開きます。

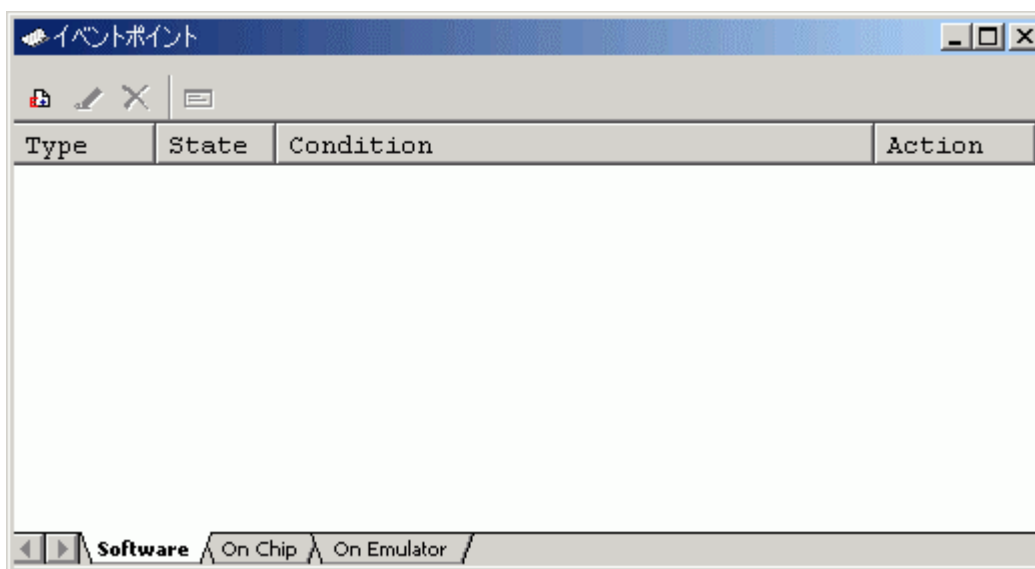


図4.30 イベントポイントウィンドウ（ソフトウェアブレーク設定前）

4 チュートリアル

マウスの右ボタンで[イベントポイント]ウィンドウをクリックし、ポップアップメニューから[追加...]を選択してください。[ブレークポイントプロパティ]ダイアログボックス([ソフトウェアブレーク]ページ)が表示されます。

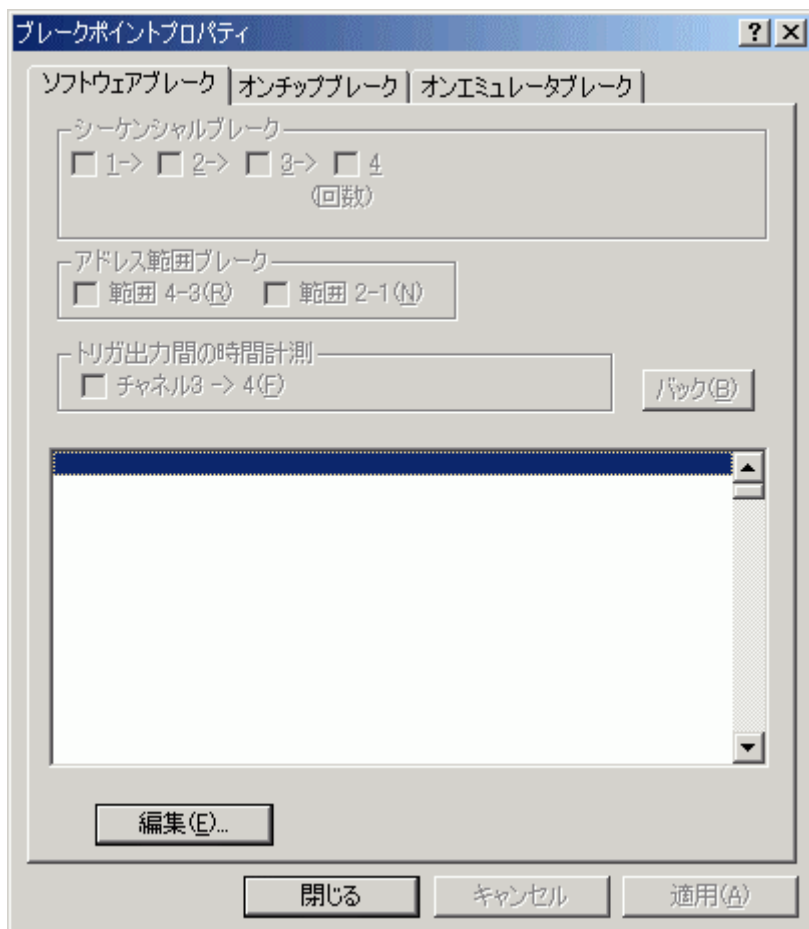


図4.31 ブレークポイントプロパティダイアログボックス

[編集...]ボタンをクリックしてください。[ソフトウェアブレイク]ダイアログボックスが表示されます。

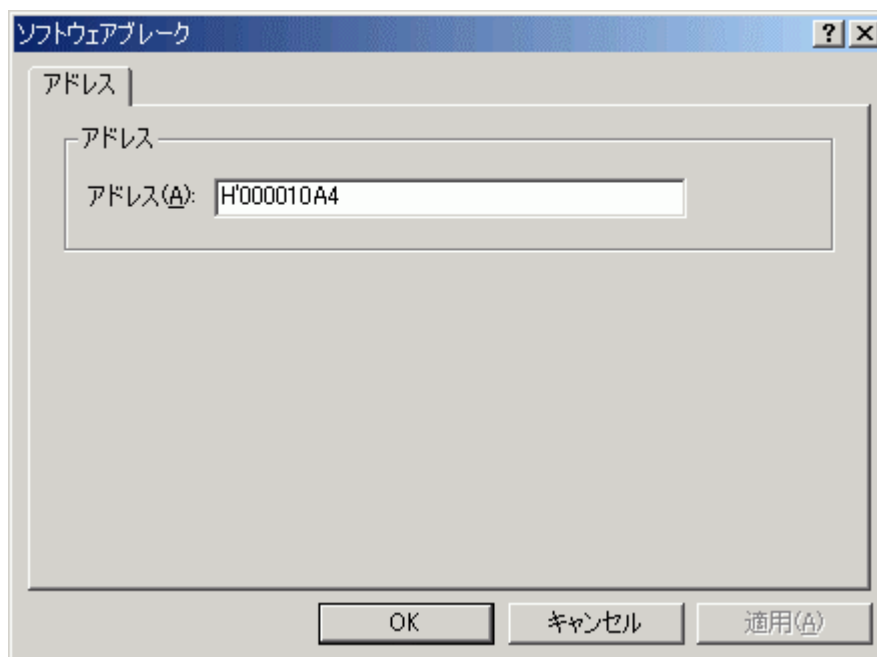


図4.32 ソフトウェアブレイクダイアログボックス

[アドレス]エディットボックスに tutorial 関数内の「p-sam->s0=a[0];」と記述されている行のアドレスを[Source]ウィンドウで参照し入力してください。本例では H'000010A4 を入力します。

【注】 本ダイアログボックスは、製品ごとに異なります。各製品の内容については、「3 デバッグ」またはオンラインヘルプを参照してください。

[OK]ボタンをクリックしてください。[ブレイクポイントプロパティ]ダイアログボックスの[閉じる]ボタンを押してください。

[イベントポイント]ウィンドウには、設定されたソフトウェアブレイクポイントが表示されます。

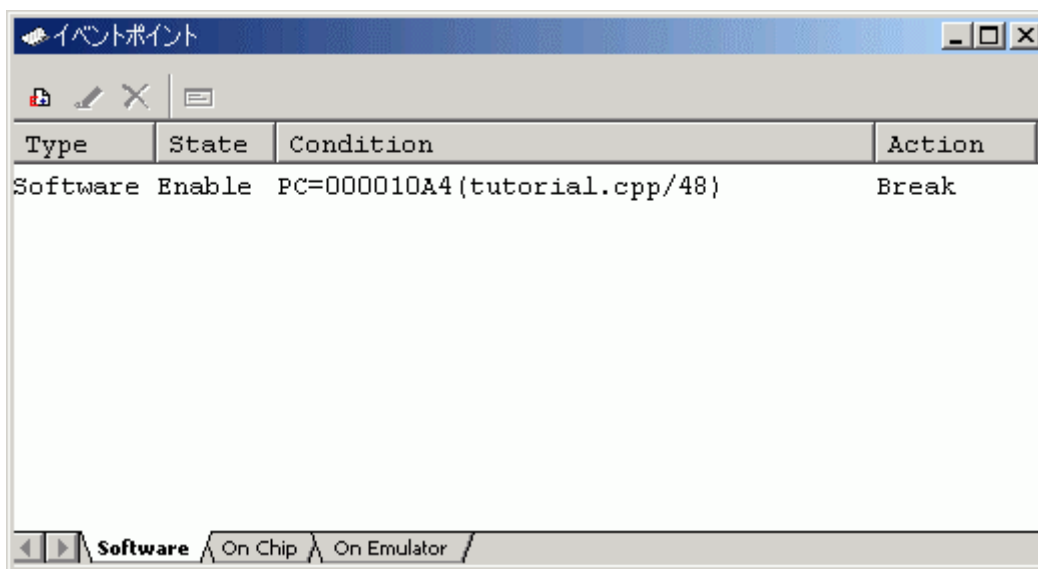


図4.33 イベントポイントウィンドウ (ソフトウェアブレイク設定時)

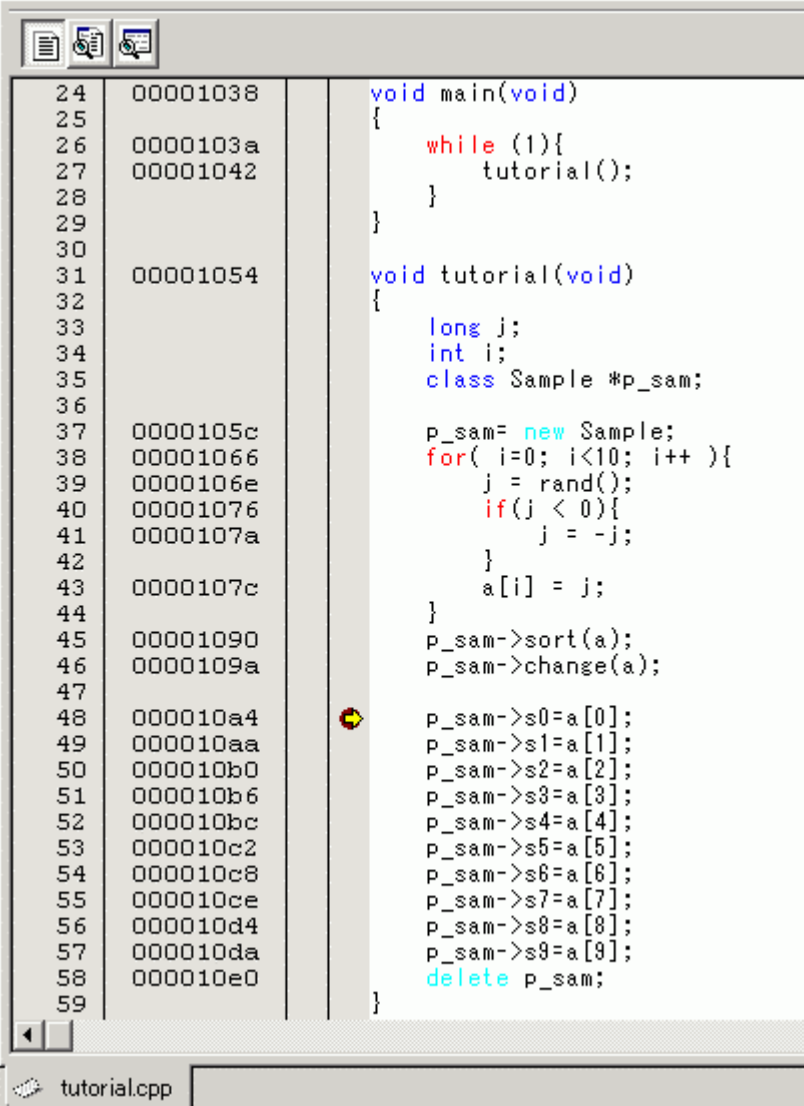
【注】 本ウィンドウで表示される内容は、製品ごとに異なります。各製品の表示内容については、「3 デバッグ」またはオンラインヘルプを参照してください。

[イベントポイント]ウィンドウを閉じてください。

4 チュートリアル

チュートリアルプログラムをブレークポイントで停止させるため、[デバッグ]メニューから[リセット後実行]を選択してください。

設定したブレークポイントまで、プログラムを実行して停止します。

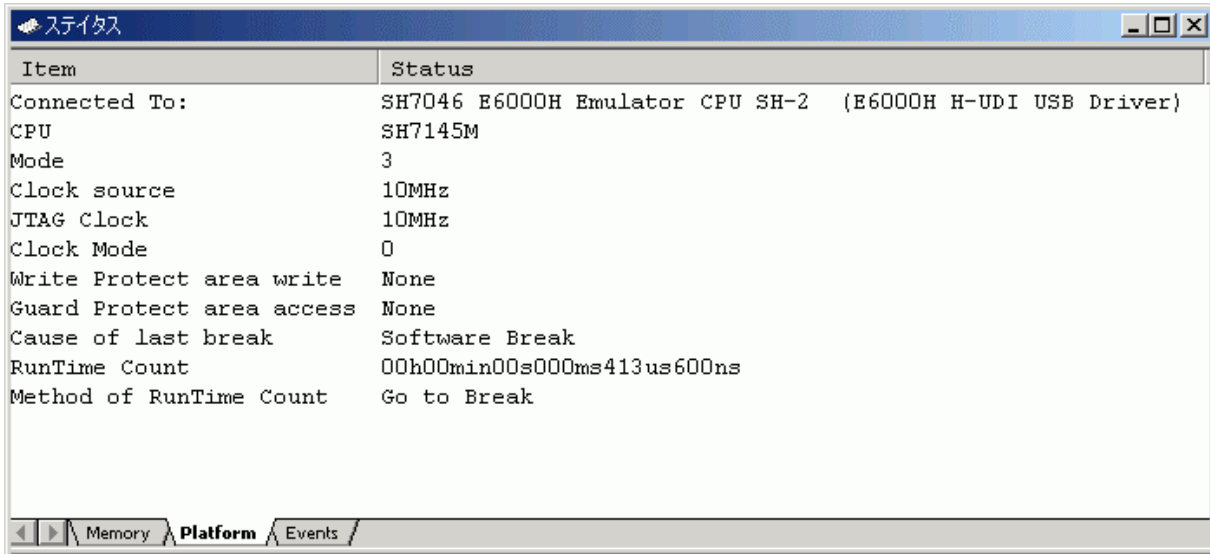


```
24 00001038 void main(void)
25
26 0000103a {
27 00001042     while (1){
28         tutorial();
29     }
30
31 00001054 void tutorial(void)
32 {
33     long j;
34     int i;
35     class Sample *p_sam;
36
37 0000105c     p_sam= new Sample;
38 00001066     for( i=0; i<10; i++ ){
39 0000106e         j = rand();
40 00001076         if(j < 0){
41 0000107a             j = -j;
42         }
43 0000107c         a[i] = j;
44     }
45 00001090     p_sam->sort(a);
46 0000109a     p_sam->change(a);
47
48 000010a4     p_sam->s0=a[0];
49 000010aa     p_sam->s1=a[1];
50 000010b0     p_sam->s2=a[2];
51 000010b6     p_sam->s3=a[3];
52 000010bc     p_sam->s4=a[4];
53 000010c2     p_sam->s5=a[5];
54 000010c8     p_sam->s6=a[6];
55 000010ce     p_sam->s7=a[7];
56 000010d4     p_sam->s8=a[8];
57 000010da     p_sam->s9=a[9];
58 000010e0     delete p_sam;
59 }
```

tutorial.cpp

図4.34 実行停止時の[エディタ]ウィンドウ（ソフトウェアブレーク）

[ステータス]ウィンドウの表示内容は、以下のようになります。



The screenshot shows a window titled 'ステータス' (Status) with a table of system parameters. The table has two columns: 'Item' and 'Status'. The data is as follows:

Item	Status
Connected To:	SH7046 E6000H Emulator CPU SH-2 (E6000H H-UDI USB Driver)
CPU	SH7145M
Mode	3
Clock source	10MHz
JTAG Clock	10MHz
Clock Mode	0
Write Protect area write	None
Guard Protect area access	None
Cause of last break	Software Break
RunTime Count	00h00min00s000ms413us600ns
Method of RunTime Count	Go to Break

At the bottom of the window, there are navigation tabs: '<|> Memory Platform Events /', with 'Platform' currently selected.


図4.35 ステータスウィンドウの表示内容（ソフトウェアブレイク）

【注】 本ウィンドウで表示される内容は、製品ごとに異なります。各製品の表示内容については、「3 デバッグ」またはオンラインヘルプを参照してください。

4.15.2 オンチップブ레이크機能

オンチップブ레이크のチャンネル4を用いて、指定した条件が5回成立した場合にブ레이크する方法を説明します。

【注】 成立回数を設定できるチャンネルは、製品ごとに異なります。各製品の内容については、「3 デバッグ」またはオンラインヘルプを参照してください。

[表示 -> コード -> イベントポイント]を選択するか、[イベントポイント]ツールバーボタンをクリックして、[イベントポイント]ウィンドウを表示してください。

先ほど設定したソフトウェアブ레이크ポイントを削除します。マウスの右ボタンで[Software]シートをクリックすることによって開くポップアップメニューから[すべてを削除]を選択し、設定されているソフトウェアブ레이크ポイントをすべて解除してください。

次はオンチップブ레이크を設定します。[イベントポイント]ウィンドウの[On Chip]タブをクリックしてください。

マウスの右ボタンで[イベントポイント]ウィンドウをクリックし、ポップアップメニューから[追加...]を選択してください。[ブ레이크ポイントプロパティ]ダイアログボックス([オンチップブ레이크]ページ)が表示されます。

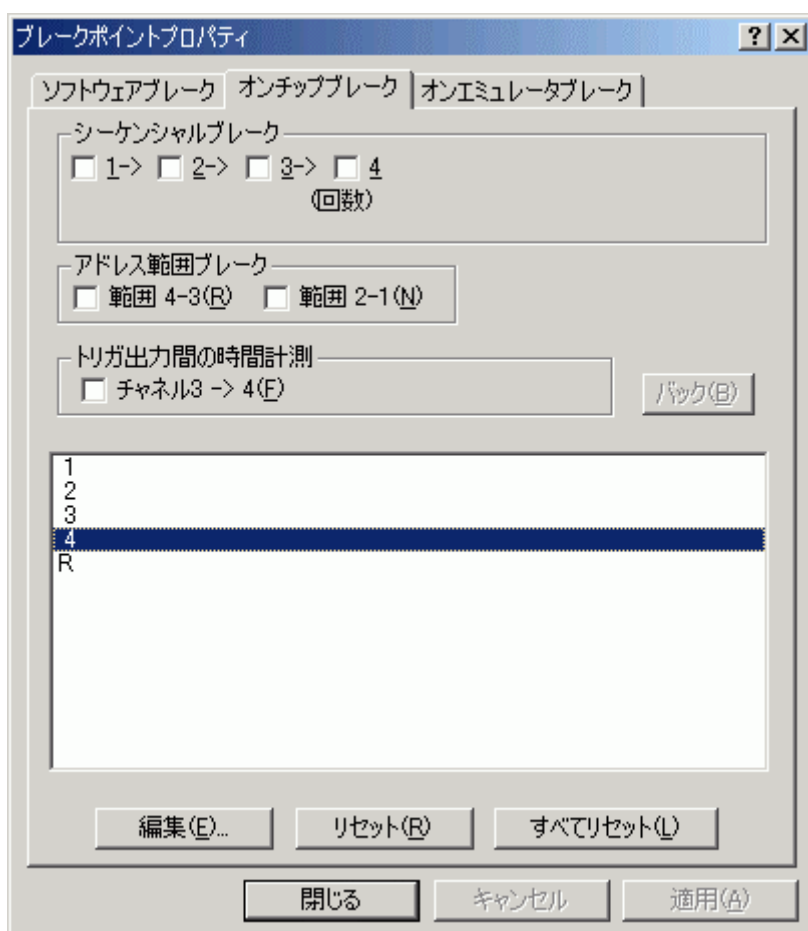


図4.36 ブ레이크ポイントプロパティダイアログボックス（オンチップブ레이크ページ）

リストボックスにて[4]を選択し、[編集...]ボタンをクリックしてください。[オンチップブ레이크 チャンネル 4]ダイアログボックスが表示されます。

[アドレス]ページの各グループボックスを以下のように設定します。

- [無効]チェックボックスのチェックを解除します。
- [アドレス]エディットボックスに tutorial 関数内の「a[i]=j;」と記述されている行のアドレスを[エディタ]ウィンドウで参照し入力してください。本例では H'0000107C を入力します。

[回数]ページの各グループボックスを以下のように設定します。

- [無効]チェックボックスのチェックを解除します。
- [回数]エディットボックスに D'5 を入力します。

【注】 本ダイアログボックスは、製品ごとに異なります。各製品の内容については、「3 デバッグ」またはオンラインヘルプを参照してください。

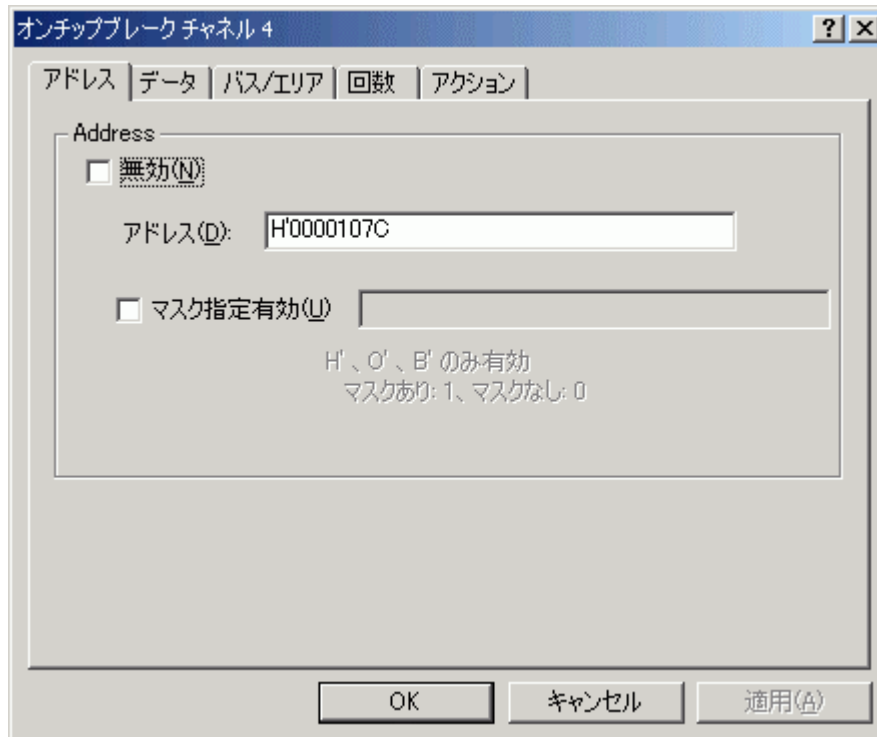


図4.37 オンチップブレイク チャンネル 4 ダイアログボックス

[OK]ボタンをクリックしてください。[ブレイクポイントプロパティ]ダイアログボックスの[閉じる]ボタンをクリックしてください。

[イベントポイント]ウィンドウには、設定されたオンチップブレイクポイントが表示されます。

【注】 本ウィンドウで表示される内容は、製品ごとに異なります。各製品の表示内容については、「3 デバッグ」またはオンラインヘルプを参照してください。

[イベントポイント]ウィンドウを閉じてください。

チュートリアルプログラムをオンチップブレイクポイントで停止させるため、[デバッグ]メニューから[リセット後実行]を選択してください。設定したブレイクポイントまで、プログラムを実行して停止します。


停止要因は[ステータス]ウィンドウの[Cause of last break]にて確認することができます。

[ウォッチ]ウィンドウから変数 i の値を参照します。変数 i の値は 4 となっており、イベントが 5 回成立したことによってブレイクが発生したことがわかります。

設定したイベントポイントを解除します。マウスの右ボタンで[イベントポイント]ウィンドウをクリックすることによって開くポップアップメニューから[すべてを削除]を選択し、設定されているイベントポイントをすべて解除してください。

4.16 トレース機能

E6000H エミュレータのトレース機能がもつリアルタイムトレースバッファでは、128k までのバスサイクルを保持でき、実行中は常に更新されます。リアルタイムトレースバッファの内容は[トレース]ウィンドウに表示されます。

[表示 -> コード -> トレース]を選択するか、[トレース]ツールバーボタンをクリックしてください。
[トレース]ウィンドウが表示されます。

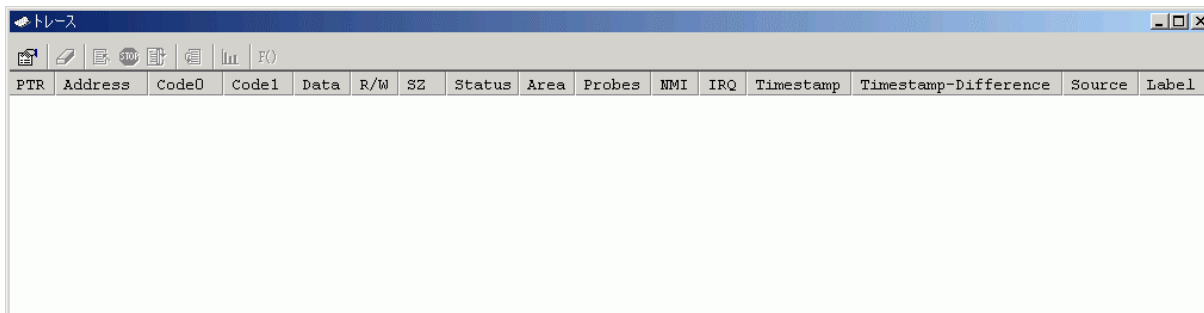


図4.38 トレースウィンドウ

[トレース]ウィンドウにトレース情報が表示されている場合は、マウスの右ボタンで[トレース]ウィンドウをクリックすることによって開くポップアップメニューから[クリア]を選択し、トレース情報をクリアします。

以下にトレース機能の概要と設定方法について説明します。

4.16.1 フリートレースによるトレース情報の表示

フリートレース機能は、ユーザプログラムの実行を開始した時点からブレークするまでのトレース情報を連続的に取得します。

- (1) すべてのブレーク条件を解除してください。マウスの右ボタンで[トレース]ウィンドウをクリックすることによって開くポップアップメニューから[設定...]を選択してください。[トレース取得プロパティ]ダイアログボックスが表示されます。[フリートレース]チェックボックスがチェックされていることを確認してください。[閉じる]ボタンをクリックしてください。

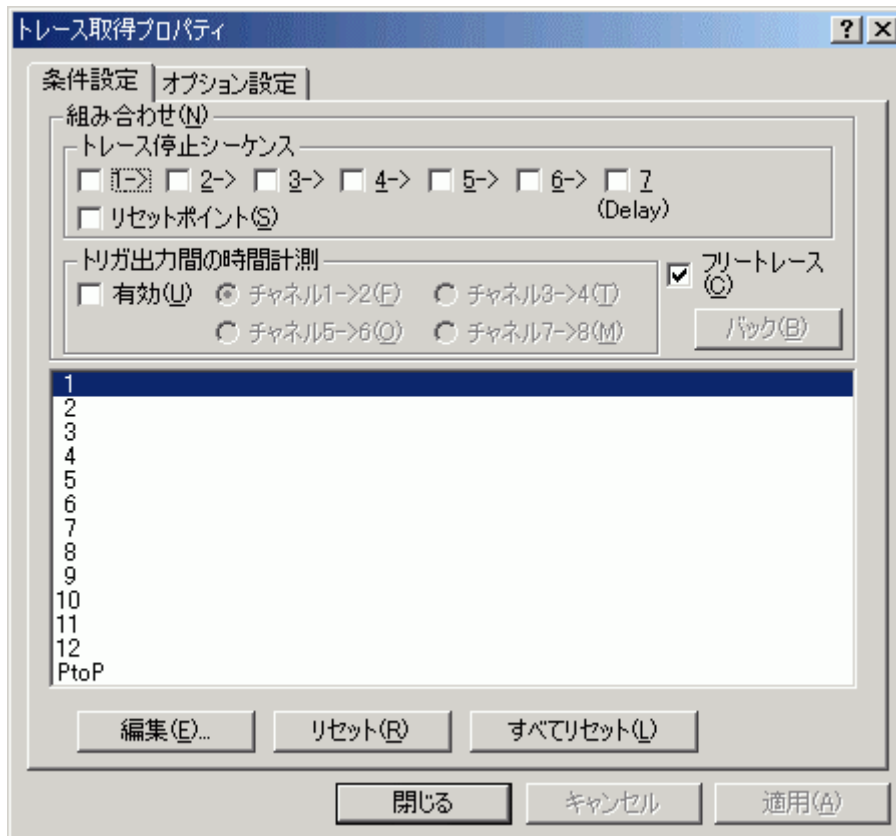


図4.39 トレース取得プロパティダイアログボックス (フリートレース)

- (2) tutorial 関数内の「p-sam->s0=a[0];」と記述されている行にソフトウェアブレークを設定してください。(「4.15.1 ソフトウェアブレーク機能」参照)
- (3) [デバッグ]メニューから[リセット後実行]を選択してください。ブレークにより処理が停止し、[トレース]ウィンドウにブレークするまでのトレース情報が表示されます。

PTR	Address	Code0	Code1	Data	R/W	SZ	Status	Area	Probes	NMI	IRQ	Timestamp	Timeste
-00012	0000212c	ADD	#H'2C,R15	MOV.L @R15+,R9	RD	L	IF	ROM	1111		1	0000h00...	0000h0C
-00011							OTH		1111		1	0000h00...	0000h0C
-00010	00002130	RTS		NOP	RD	L	IF	ROM	1111		1	0000h00...	0000h0C
-00009							OTH		1111		1	0000h00...	0000h0C
-00008	ffffffd0				RD	L	DAT	RAM	1111		1	0000h00...	0000h0C
-00007	000010a4	.DATA.W	H'0000		RD	L	IF	ROM	1111		1	0000h00...	0000h0C
-00006							OTH		1111		1	0000h00...	0000h0C
-00005	000010a8			2d22d615	RD	L	IF	ROM	1111		1	0000h00...	0000h0C
-00004							OTH		1111		1	0000h00...	0000h0C
-00003							OTH		1111		1	0000h00...	0000h0C
-00002									1111		1	0000h00...	0000h0C
-00001									1111		1	0000h00...	0000h0C
+00000									1111		1	0000h00...	0000h0C

図4.40 トレースウィンドウ (フリートレース)

4.16.2 トレース停止によるトレース情報の表示

トレース停止機能は、指定した条件が成立するとトレース情報の取得を停止します。ユーザプログラムをブレークさせずに、トレース情報からプログラムの流れを確認することができます。

- (1) ブレーク条件が設定されている場合はすべて解除してください。
- (2) [トレース取得プロパティ]ダイアログボックス-[条件設定]ページのリストボックスから[1]を選択し、[編集...]ボタンをクリックしてください。[トレース取得条件 チャンネル 1]ダイアログボックスが表示されます。[アクション]ページの[条件成立後の処理]グループボックスから[トレース停止]ラジオボタンを選択してください。

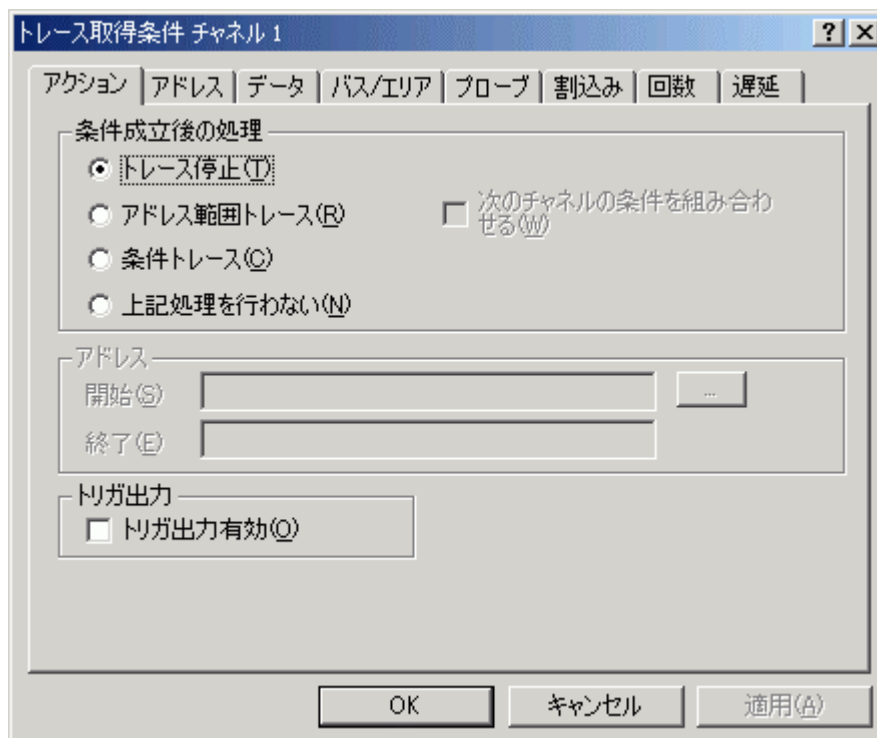


図4.41 トレース取得条件 チャンネル 1 ダイアログボックス (トレース停止)

(3)条件のアドレスを設定します。[トレース取得条件 チャンネル 1]ダイアログボックス-[アドレス]ページの[無効]チェックボックスを解除し、[開始]エディットボックスに tutorial 関数内の「a[i]=j;」と記述されている行のアドレスを[エディタ]ウィンドウで参照し入力してください。本例では H'0000107C を入力します。これでアドレスの指定は終了です。[OK]ボタンをクリックし、[トレース取得条件 チャンネル 1]ダイアログボックスを閉じます。

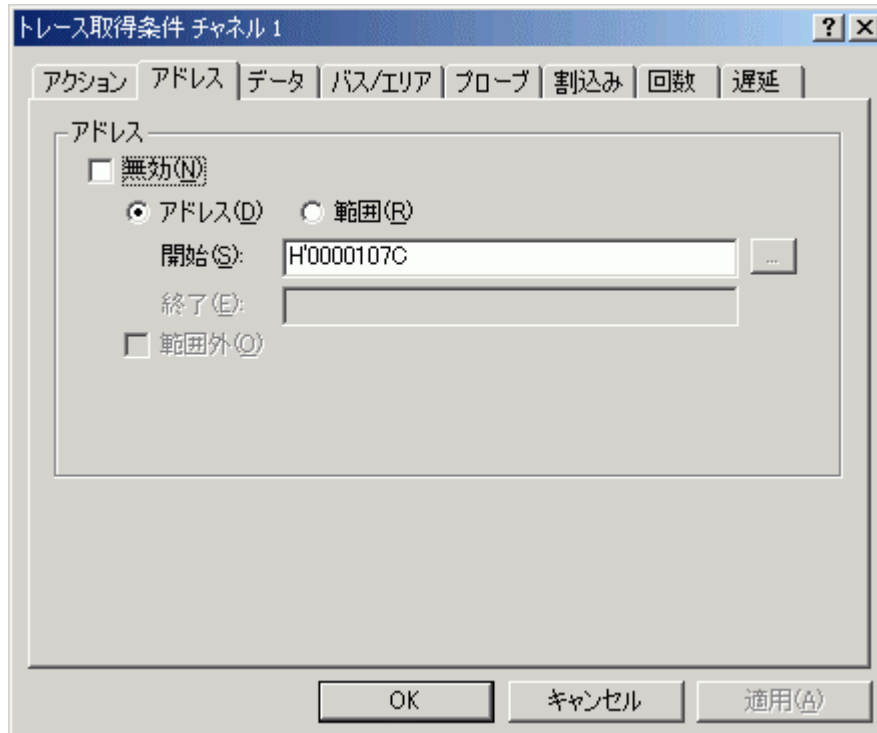


図4.42 トレース取得条件 チャンネル 1 ダイアログボックス ([アドレス]ページ)

4 チュートリアル

(4) [トレース取得プロパティ]ダイアログボックス-[条件設定]ページのリストボックスに設定した内容が表示されます。[トレース取得プロパティ]ダイアログボックスの[閉じる]ボタンをクリックしてください。

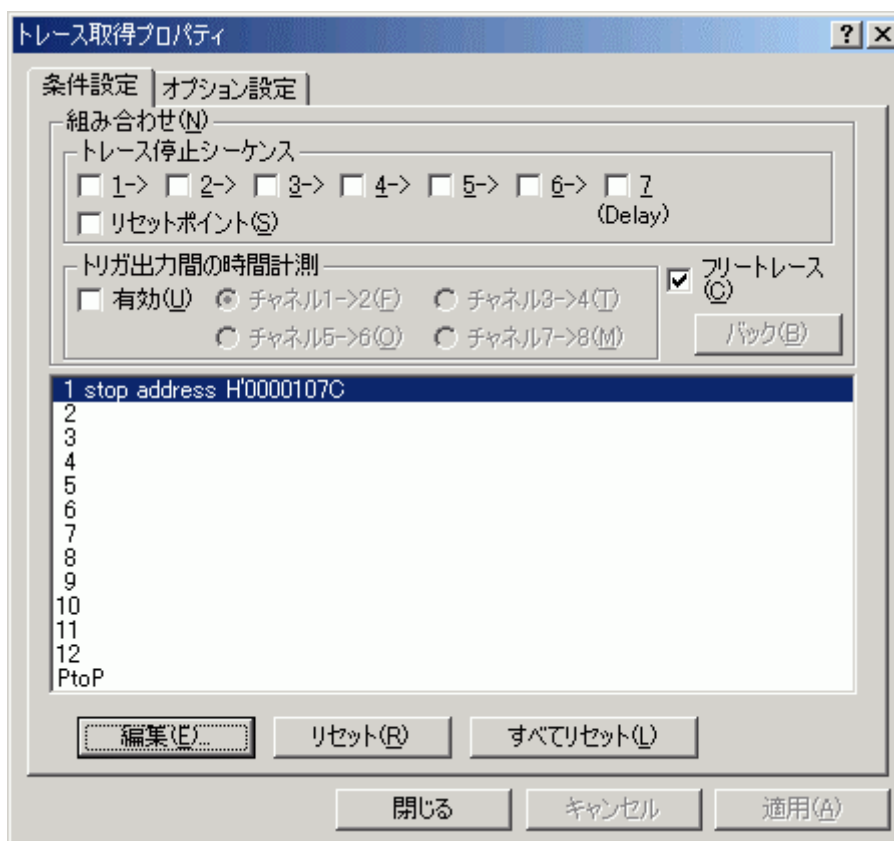


図4.43 トレース取得プロパティダイアログボックス (トレース停止)

(5) [デバッグ]メニューから[リセット後実行]を選択してください。トレース停止条件が成立し、[トレース]ウィンドウに以下の内容が表示されます。

PTR	Address	Code0	Code1	Data	R/W	SZ	Status	Area	Probes	NMI	IRQ	Timestamp	Time	
-00012	00001074	MOV	R0, R2	CMP/P2 R2	62034211	RD	L	IF	ROM	1111	1	0000h00...	0000h0C	
-00011								OTH		1111	1	0000h00...	0000h0C	
-00010	ffffffd0				d529f317	RD	L	DAT	RAM	1111	1	0000h00...	0000h0C	
-00009	00001078	BT	@H'107c:8		8900622b	RD	L	IF	ROM	1111	1	0000h00...	0000h0C	
-00008								OTH		1111	1	0000h00...	0000h0C	
-00007	0000107c				d42066e3	RD	L	IF	ROM	1111	1	0000h00...	0000h0C	
-00006	0000107c	MOV.L	@(H'00...	MOV	R14, R6	d42066e3	RD	L	IF	ROM	1111	1	0000h00...	0000h0C
-00005								OTH		1111	1	0000h00...	0000h0C	
-00004	00001080	MOV	R6, R5	SHLL2	R5	65634508	RD	L	IF	ROM	1111	1	0000h00...	0000h0C
-00003	00001100				ffffe400	RD	L	DAT	ROM	1111	1	0000h00...	0000h0C	
-00002	00001084	MOV	R5, R0	MOV.L	R2, @(R...	60530426	RD	L	IF	ROM	1111	1	0000h00...	0000h0C
-00001								OTH		1111	1	0000h00...	0000h0C	
+00000	00001088				7e01e10a	RD	L	IF	ROM	1111	1	0000h00...	0000h0C	

図4.44 トレースウィンドウ (トレースストップ)

4.16.3 条件トレースによるトレース情報の表示

条件トレース機能は、指定した条件が成立した箇所のみをトレース取得します。特定のアドレス（グローバル変数やメモリマップド I/O など）へのリード/ライトに着目したプログラムの解析に役立ちます。

- (1) ユーザプログラムが実行中の場合は、[デバッグ]メニューから[プログラムの停止]を選択し、実行を停止させてください。
- (2) ブレーク条件が設定されている場合はすべて解除してください。[トレース取得プロパティ]ダイアログボックス-[条件設定]ページの[フリートレース]チェックボックスのチェックを解除してください。（解除しない場合はフリートレースになります。）
- (3) [トレース取得プロパティ]ダイアログボックス-[条件設定]ページのリストボックスから[1]を選択し、[編集...]ボタンをクリックしてください。[トレース取得条件 チャンネル 1]ダイアログボックスが表示されます。[アクション]ページの[条件成立後の処理]グループボックスから[条件トレース]ラジオボタンを選択してください。
- (4) 条件のアドレスを設定します。[トレース取得条件 チャンネル 1]ダイアログボックス-[アドレス]ページの[無効]チェックボックスを解除し、[開始]エディットボックスに a[0]のアドレスを[ウォッチ]ウィンドウで参照し入力してください。本例では H'FFFFFFE400 を入力します。これでアドレスの指定は終了です。[OK]ボタンをクリックし、[トレース取得条件 チャンネル 1]ダイアログボックスを閉じます。
- (5) [トレース取得プロパティ]ダイアログボックス-[条件設定]ページのリストボックスに設定した内容が表示されます。[トレース取得プロパティ]ダイアログボックスの[閉じる]ボタンをクリックしてください。
- (6) tutorial 関数内の「delete p_sam;」と記述されている行のアドレス（本例では H'000010E0）にソフトウェアブレークを設定します。（「4.15.1 ソフトウェアブレーク機能」参照）
- (7) [デバッグ]メニューから[リセット後実行]を選択してください。ブレークにより処理が停止し、[トレース]ウィンドウに以下の内容が表示されます。

PTR	Address	Code0	Code1	Data	R/W	SZ	Status	Area	Probes	NMI	IRQ	Timestamp	Timesta
-00012	ffffe400			000015fb	RD	L	DAT	RAM	1111		1	0000h00...	0000h0C
-00011	ffffe400			00000ff6	WR	L	DAT	RAM	1111		1	0000h00...	0000h0C
-00010									1111		1	0000h00...	0000h0C
-00009									1111		1	0000h00...	0000h0C
-00008	ffffe400			00000ff6	RD	L	DAT	RAM	1111		1	0000h00...	0000h0C
-00007	ffffe400			00000ff6	RD	L	DAT	RAM	1111		1	0000h00...	0000h0C
-00006									1111		1	0000h00...	0000h0C
-00005									1111		1	0000h00...	0000h0C
-00004	ffffe400			00000ff6	RD	L	DAT	RAM	1111		1	0000h00...	0000h0C
-00003	ffffe400			0000794b	WR	L	DAT	RAM	1111		1	0000h00...	0000h0C
-00002									1111		1	0000h00...	0000h0C
-00001									1111		1	0000h00...	0000h0C
+00000	ffffe400			0000794b	RD	L	DAT	RAM	1111		1	0000h00...	0000h0C

図4.45 トレースウィンドウ（条件トレース）

4.16.4 統計

取得したトレース情報から、内蔵 RAM に対する書き込み回数を収集します。

- (1) ブレーク条件が設定されている場合はすべて解除してください。[トレース取得プロパティ]ダイアログボックス-[条件設定]ページの[すべてリセット]をクリックしてトレース条件を解除してください。[トレース取得プロパティ]ダイアログボックス-[条件設定]ページの[フリートレース]チェックボックスのチェックを入れてください。
- (2) tutorial 関数内の「p-sam->s0=a[0];」記述されている行のアドレス（本例では H'000010A4）でブレークするように設定します。（「4.15.1 ソフトウェアブレーク機能」参照）
- (3) [デバッグ]メニューから[リセット後実行]を選択してください。ブレークにより処理が停止し、[トレース]ウィンドウにトレース情報が表示されます。
- (4) マウスの右ボタンで[トレース]ウィンドウをクリックすることによって開くポップアップメニューから[統計...]を選択してください。トレースデータのロードを示すダイアログが表示された後に、[統計]ダイアログボックスが表示されます。

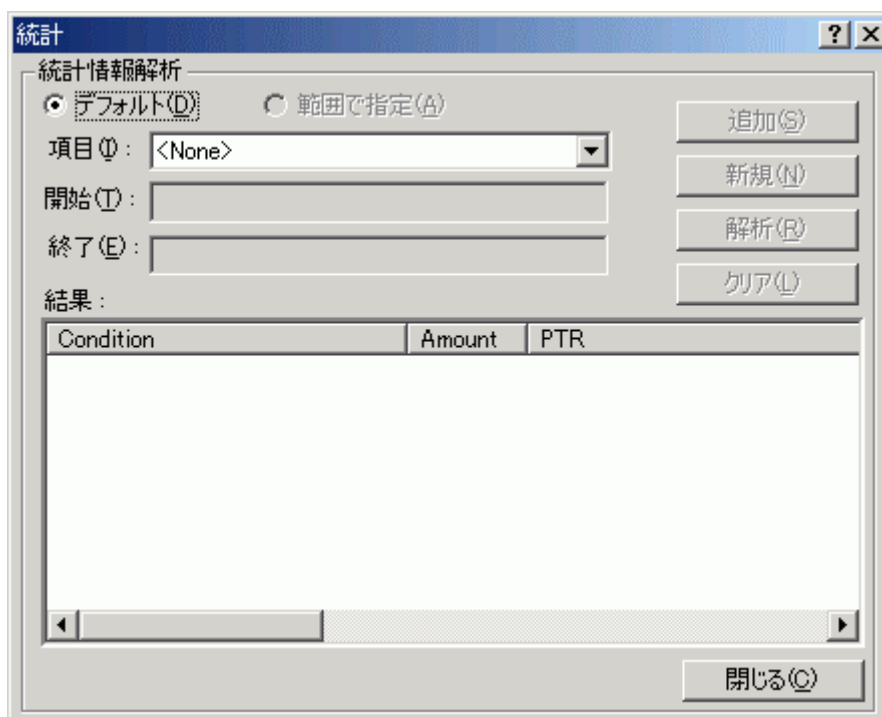


図4.46 統計ダイアログボックス

- (4) [項目]ドロップダウンリストボックスから R/W を選択し、[開始]エディットボックスに WR を入力します。入力終了後、[新規]ボタンをクリックすると[結果]リストボックスの[Condition]列に "R/W=WR" と表示されます。

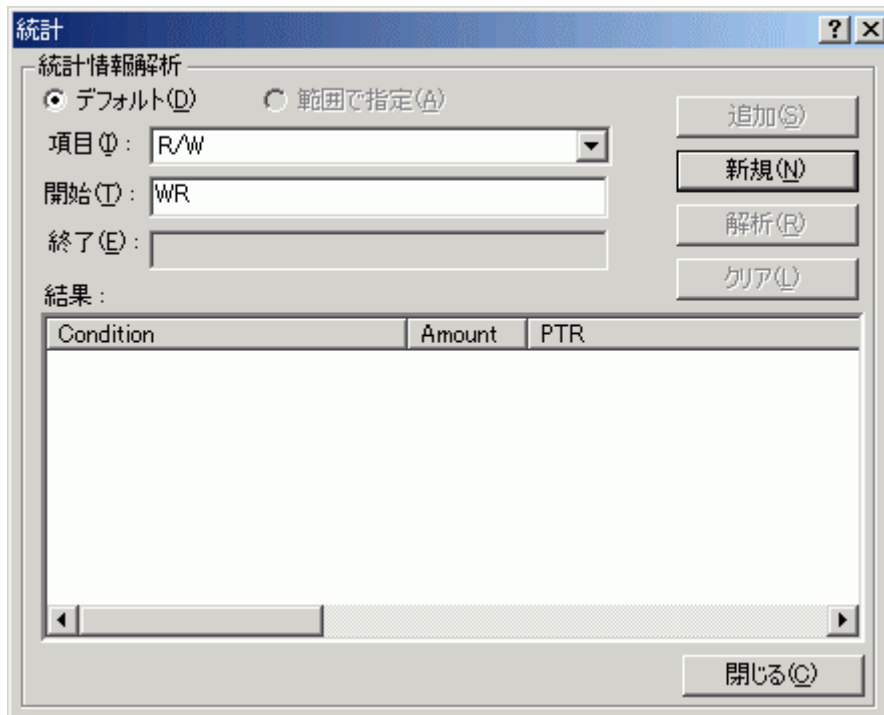


図4.47 統計ダイアログボックス (新規条件)

- (5) 続いて[項目]ドロップダウンリストボックスから Area を選択し、[開始]エディットボックスに RAM を入力します。入力終了後、[追加]ボタンをクリックすると[結果]リストボックスの[Condition]列に表示されていた "R/W=WR" に条件が追加され、"R/W=WR & Area=RAM" と表示されます。これで条件の入力は終了です。

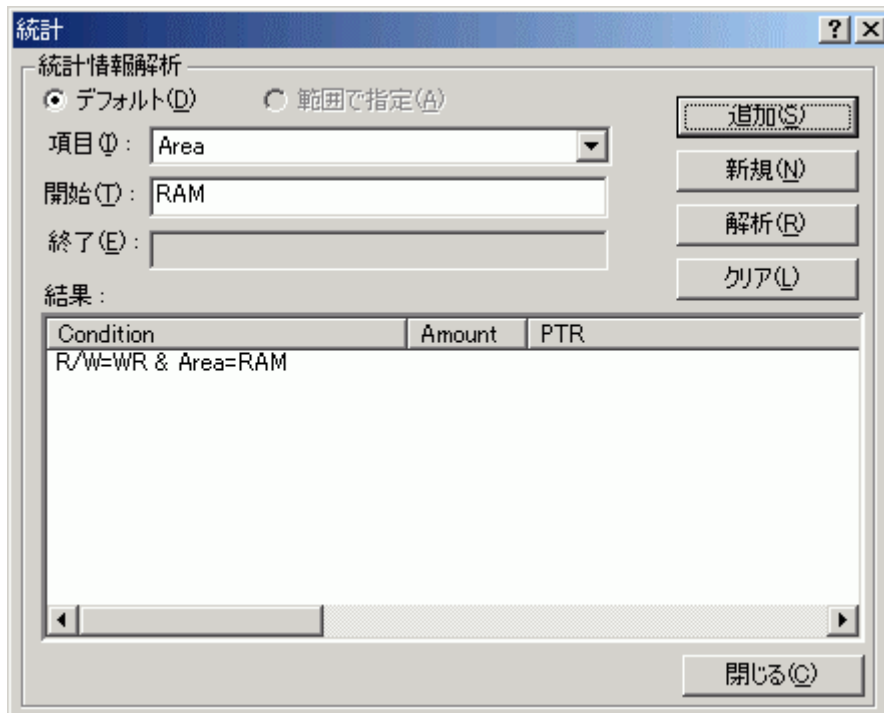


図4.48 統計ダイアログボックス (条件追加)

4 チュートリアル

(6) 入力した条件による解析を行います。[解析]ボタンをクリックしてください。条件に該当する件数とそのPTRが表示されます。

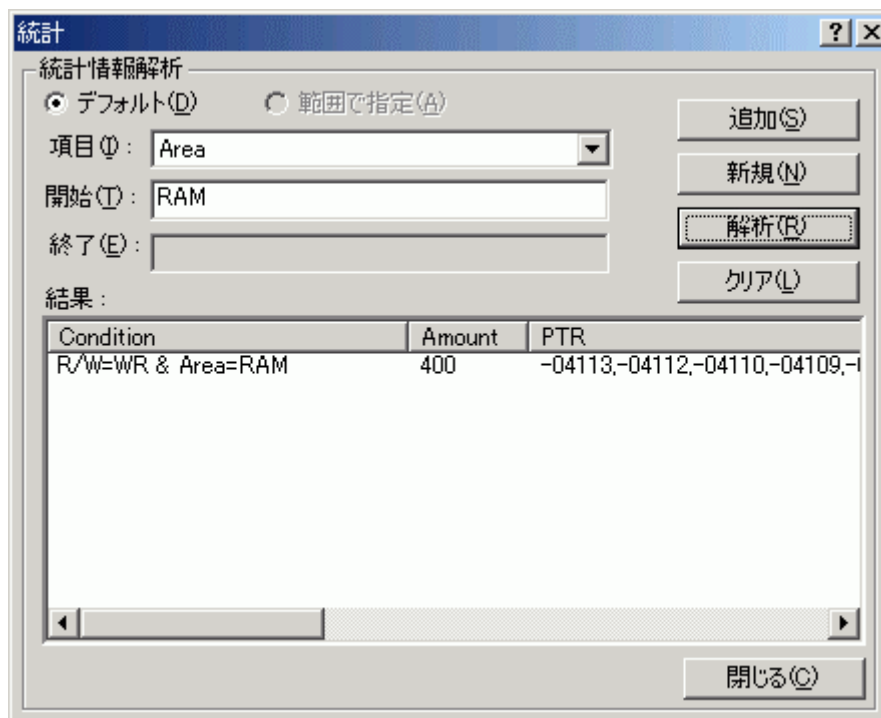


図4.49 統計ダイアログボックス（解析結果）

(7) [閉じる]ボタンをクリックし、[統計]ダイアログボックスを閉じます。

(8) 設定したイベントポイントを解除し、トレース情報をクリアします。マウスの右ボタンで[イベントポイント]ウィンドウをクリックすることによって開くポップアップメニューから[すべてを削除]を選択し、設定されているイベントポイントをすべて解除します。マウスの右ボタンで[トレース]ウィンドウをクリックすることによって開くポップアップメニューから[クリア]を選択し、トレース情報をクリアします。

4.16.5 関数コール

取得したトレース情報から、関数をコールしている情報のみを収集します。

- (1) tutorial 関数内の「p-sam->s0=a[0];」と記述されている行のアドレス（本例では H'000010A4）でブレークするように設定します。（「4.15.1 ソフトウェアブレーク機能」参照）
- (2) [デバッグ]メニューから[リセット後実行]を選択してください。ブレークにより処理が停止し、[トレース]ウィンドウにトレース情報が表示されます。
- (3) マウスの右ボタンで[トレース]ウィンドウをクリックすることによって開くポップアップメニューから[関数コール...]を選択してください。[関数コール箇所の表示]ダイアログボックスが表示されます。

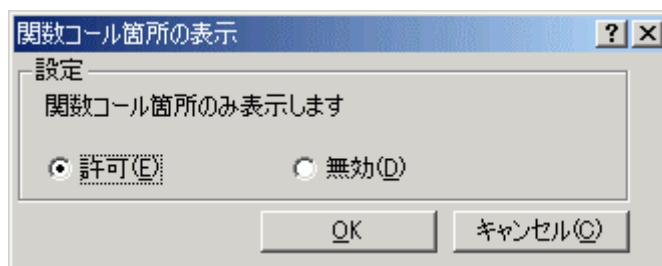


図4.50 関数コール箇所の表示ダイアログボックス

- (4) [許可]ラジオボタンをクリックし、[OK]ボタンをクリックしてください。[トレース]ウィンドウの表示が関数をコールしている情報の変更に変わります。

PTR	Address	Data	R/W	SZ	Status	Area	Probes	NMI	IRQ	Source	Label
-04125	00000800	d20ee410	RD	L	IF	ROM	1111		1	set_vbr({(void *)}((char *)&IW...	_PowerON_Reset_PC
-02333	00001038	4f22e101	RD	L	IF	ROM	1111		1	void main(void)	_main
-02325	00001054	2fd62fe6	RD	L	IF	ROM	1111		1	void tutorial(void)	tutorial()
-02314	00002000	4f222448	RD	L	IF	ROM	1111		1	Sample::Sample()	Sample::Sample()
-02205	00001000	7ffcd60b	RD	L	IF	ROM	1111		1	char *sbrk(unsigned long size) ...	_sbrk
-01588	00002032	00092fc6	RD	L	IF	ROM	1111		1	void Sample::sort(long *a)	Sample::sort(long *)
-00439	000020ce	00092f96	RD	L	IF	ROM	1111		1	void Sample::change(long *a)	Sample::change(long *)

図4.51 トレースウィンドウ（関数コール）

- (5) [トレース]ウィンドウの表示を元の状態に戻します。（3）の手順にて[関数コール箇所の表示]ダイアログボックスを表示し、[無効]ラジオボタンをクリック、その後[OK]ボタンをクリックします。
- (6) 設定したイベントポイントを解除し、トレース情報をクリアします。マウスの右ボタンで[イベントポイント]ウィンドウをクリックすることによって開くポップアップメニューから[すべてを削除]を選択し、設定されているイベントポイントをすべて解除します。マウスの右ボタンで[トレース]ウィンドウをクリックすることによって開くポップアップメニューから[クリア]を選択し、トレース情報をクリアします。

4.17 スタックトレース機能

E6000H エミュレータでは、スタック情報を用いて、関数呼び出し履歴を表示します。

- 【注】
1. 本機能は、Dwarf2 形式のデバッグ情報をもったロードモジュールをロードした場合のみ使用できます。Dwarf2 形式のデバッグ情報をもったロードモジュールは、H8S,H8/300 C/C++コンパイラ V4.0 以降でサポートしています。
 2. 本機能の詳細については、オンラインヘルプを参照してください。

4 チュートリアル

sort 関数内の行の[S/W ブレークポイント]カラムをダブルクリックして、ソフトウェアブレークポイントを設定してください。

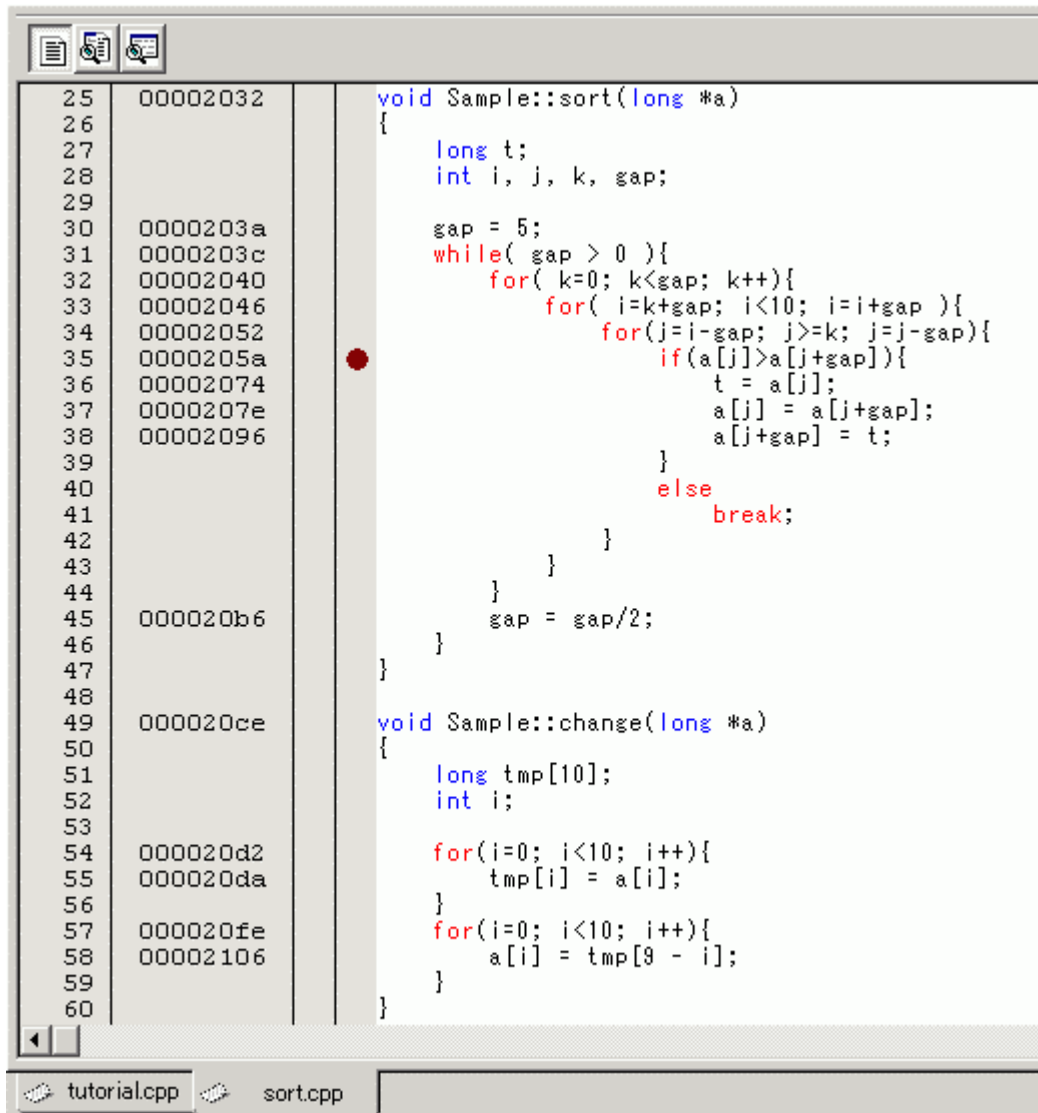
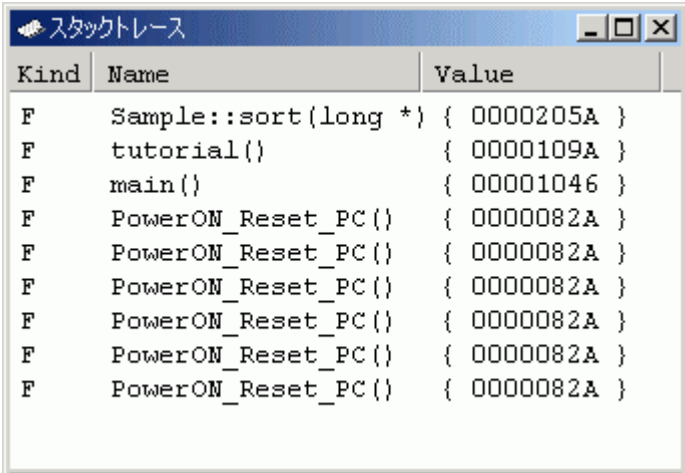


図4.52 [エディタ]ウィンドウ (ソフトウェアブレークポイントの設定)

[デバッグ]メニューから[リセット後実行]を選択してください。

ブレーク後、[表示]メニューから[コード]サブメニューを選択し、[スタックトレース]を選択し[スタックトレース]ウィンドウを開いてください。



Kind	Name	Value
F	Sample::sort(long *)	{ 0000205A }
F	tutorial()	{ 0000109A }
F	main()	{ 00001046 }
F	PowerON_Reset_PC()	{ 0000082A }
F	PowerON_Reset_PC()	{ 0000082A }
F	PowerON_Reset_PC()	{ 0000082A }
F	PowerON_Reset_PC()	{ 0000082A }
F	PowerON_Reset_PC()	{ 0000082A }
F	PowerON_Reset_PC()	{ 0000082A }

図4.53 スタックトレースウィンドウ

現在 PC が sort()関数内にあり、sort()関数は、tutorial()関数からコールされていることがわかります。

sort 関数内の行の[S/W ブレークポイント]カラムを再度ダブルクリックして、ソフトウェアブレークポイントを解除します。

4.18 パフォーマンス測定機能

E6000H エミュレータには、チップのパフォーマンスを測定する機能として、以下に示すモードがあります。

- 指定範囲内時間測定
- 指定アドレス間時間測定
- 指定アドレス範囲間時間測定
- 領域アクセス回数測定
- 指定範囲内コール回数測定

本チュートリアルでは「指定範囲内時間測定」の設定方法について説明します。

4.18.1 指定範囲内時間測定

- (1) [表示]メニューから[パフォーマンス]サブメニューを選択し、[パフォーマンス解析]を選択します。[パフォーマンス解析方式の選択]ダイアログボックスが表示されます。

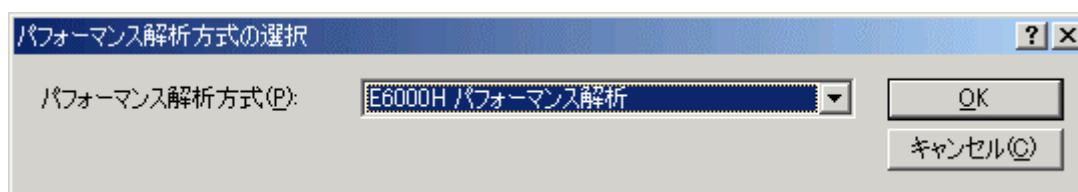


図4.54 パフォーマンス解析方式の選択ダイアログボックス

- (2) [パフォーマンス解析方式の選択]ダイアログボックスの[パフォーマンス解析方式]ドロップダウンリストボックスから"E6000H パフォーマンス解析"を選択し、[OK]ボタンをクリックします。[パフォーマンス解析]ウィンドウが表示されます。

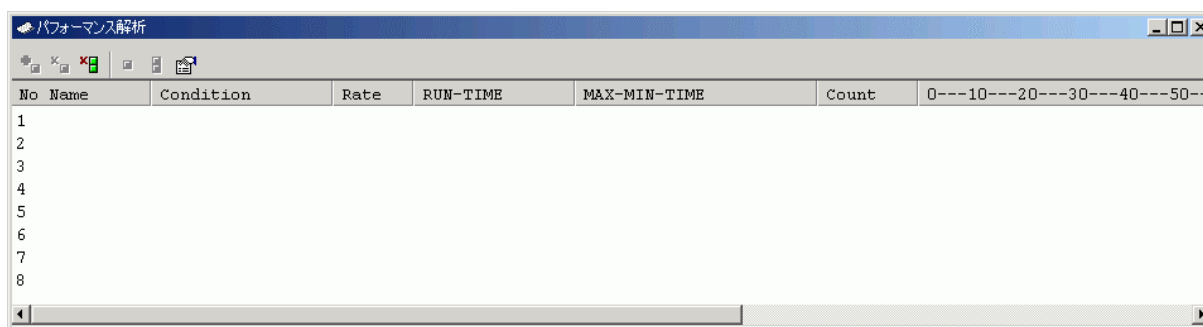


図4.55 パフォーマンス解析ウィンドウ

(3) [パフォーマンス解析]ウィンドウの[No]列が"1"の行を選択し、マウスの右ボタンをクリックすることによって開くポップアップメニューから[設定...]を選択してください。[パフォーマンス解析プロパティ]ダイアログボックスが表示されます。

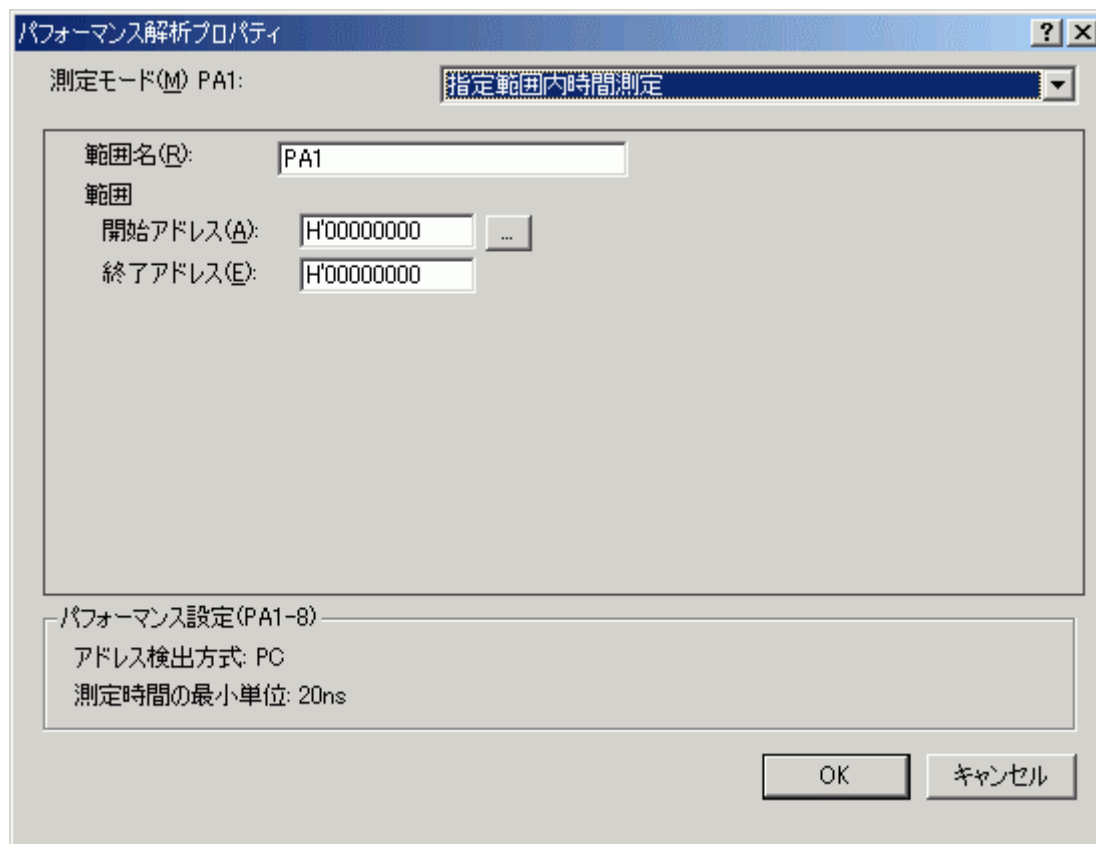


図4.56 パフォーマンス解析プロパティダイアログボックス

(4) [測定モード]ドロップダウンリストボックスから、モードとして指定範囲内時間測定を選択します。

(5) パラメータの設定は以下とします。

- [範囲名]エディットボックスに sort と入力します。
- [開始アドレス]エディットボックスの右側の[...]ボタンをクリックし、[関数範囲入力]ダイアログボックスを表示します。[関数範囲入力]ダイアログボックスの[関数名]エディットボックスに関数名"Sample::sort (long *)"を入力し、[OK]ボタンをクリックします。[開始アドレス]エディットボックスと[終了アドレス]エディットボックスに指定した関数のアドレスが設定されます。

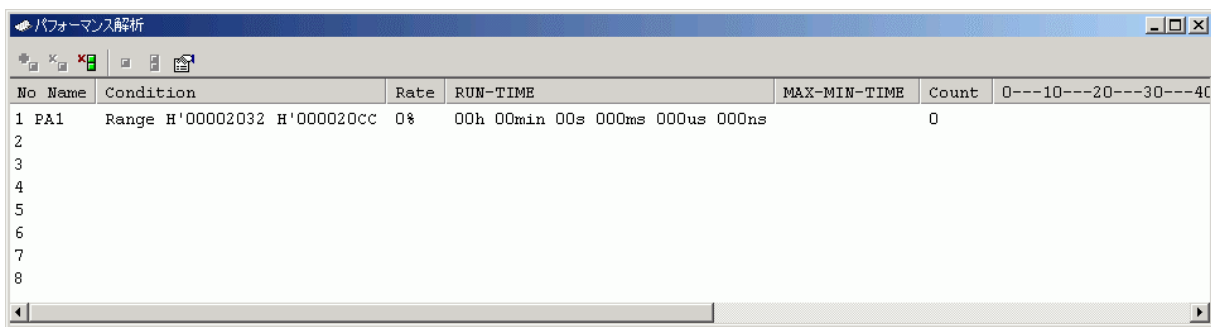


図4.57 関数範囲入力ダイアログボックス

【注】 [関数範囲入力]ダイアログボックスにより算出されるアドレスは参考値です。場合により関数の終了アドレスが異なる場合があります。
[逆アセンブリ]ウィンドウにより関数の最終命令を確認し、[終了アドレス]の設定値を最終命令のアドレスに補正してください。(一般的に関数の最終命令は RTS 命令となります。)
アドレス入力ではアドレス値以外にラベル名および式の指定も可能です。

4 チュートリアル

(6) [OK]ボタンをクリックします。[パフォーマンス解析]ウィンドウの[No]列 1 に設定した内容が表示されます。これで指定範囲内時間測定の設定は終了です。

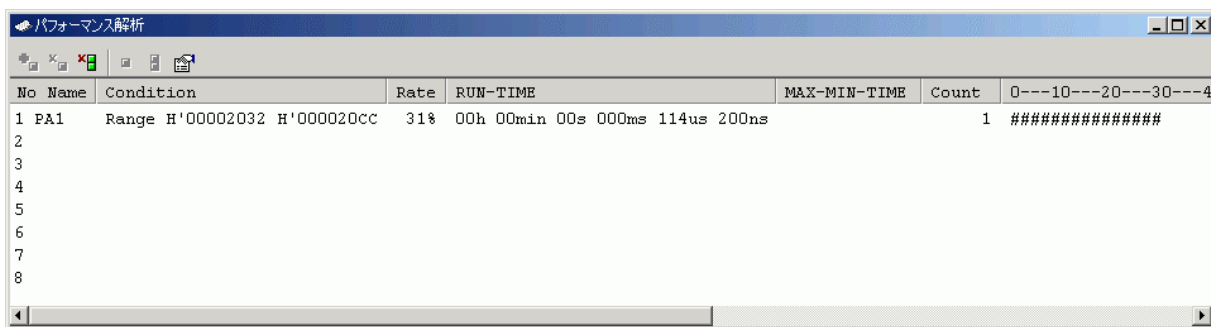


No	Name	Condition	Rate	RUN-TIME	MAX-MIN-TIME	Count	0---10---20---30---40
1	PA1	Range H'00002032 H'000020cc	0%	00h 00min 00s 000ms 000us 000ns		0	
2							
3							
4							
5							
6							
7							
8							

図4.58 パフォーマンス解析ウィンドウ（設定完了時）

(7) tutorial 関数内の「p-sam->change(a);」と記述されている行のアドレス（本例ではH'0000109A）にソフトウェアブレークポイントによるブレークを設定します。（「4.15.1 ソフトウェアブレーク機能」参照）

(8) [デバッグ]メニューから[リセット後実行]を選択してください。ブレークにより処理が停止し、[パフォーマンス解析]ウィンドウに以下の内容が表示されます。[Count]列の値が1となっており、sort 関数が1回実行されたことと、実行時間が確認できます。本チュートリアルでは、最小時間測定幅を20nsで設定しています。最小時間測定幅の変更は[コンフィグレーションプロパティ]ダイアログボックスにて行います。



No	Name	Condition	Rate	RUN-TIME	MAX-MIN-TIME	Count	0---10---20---30---40
1	PA1	Range H'00002032 H'000020cc	31%	00h 00min 00s 000ms 114us 200ns		1	#####
2							
3							
4							
5							
6							
7							
8							

図4.59 パフォーマンス解析ウィンドウ（結果表示）

(9) パフォーマンスの設定をクリアし、イベントポイントを解除します。マウスの右ボタンで[パフォーマンス解析]ウィンドウをクリックすることによって開くポップアップメニューから[全てリセット]を選択し、設定されている設定をすべてクリアします。マウスの右ボタンで[イベントポイント]ウィンドウをクリックすることによって開くポップアップメニューから[すべてを削除]を選択し、設定されているイベントポイントをすべて解除します。

4.19 モニタ機能

E6000H エミュレータには、指定したアドレスのメモリ内容を、ユーザプログラム実行中にモニタすることができます。

ここでは tutorial 関数の変数 a が割り当てられたアドレス近辺の内容をモニタします。

- (1) [表示]メニューから[CPU]サブメニューを選択、さらに[モニタ]サブメニューを選択し、[モニタ設定...]を選択します。[Monitor Setting]ダイアログボックスが表示されます。

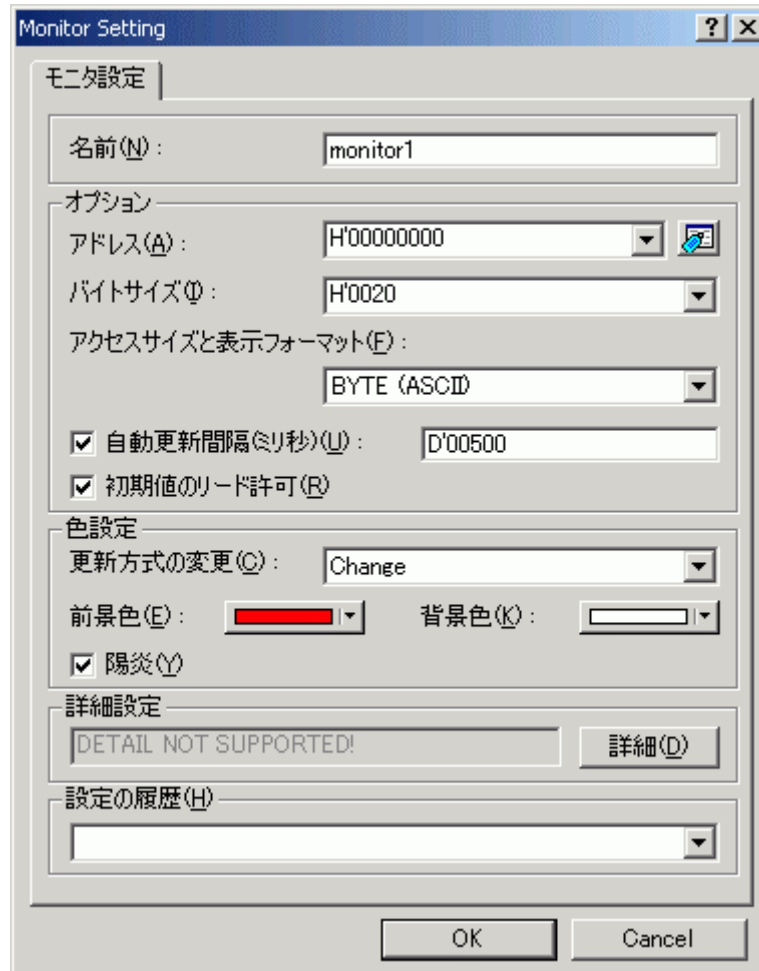


図4.60 Monitor Setting ダイアログボックス

4 チュートリアル

(2) [モニタ設定]ページの内容を以下のように設定します。

- [名前]エディットボックスに monitor1 と入力します。
- [オプション]グループボックスの各パラメータを以下の設定とします。
 - [アドレス]エディットボックスに tutorial 関数で定義している変数 a が割りついているアドレスを [ローカル]ウィンドウで参照し入力してください。本例では H'FFFE400 を入力します。
 - [バイトサイズ]ドロップダウンリストボックスに H'20 と入力します。
 - [アクセスサイズと表示フォーマット]ドロップダウンリストボックスから BYTE (ASCII) を選択します。
 - [自動更新間隔 (ミリ秒)]のチェックボックスをチェックし、エディットボックスに D'00500 と入力します。
 - [初期値のリード許可]チェックボックスをチェックします。
- [色指定]グループボックスの各パラメータを以下の設定とします。
 - [更新方式の変更]ドロップダウンリストボックスから Change を選択します。
 - [前景色]ドロップダウンリストボックスを赤に、[背景色]ドロップダウンリストボックスを白に選択します。
 - [陽炎]チェックボックスをチェックします。

【注】 前景色および背景色の設定はご使用のオペレーティングシステムにより使用できない場合があります。

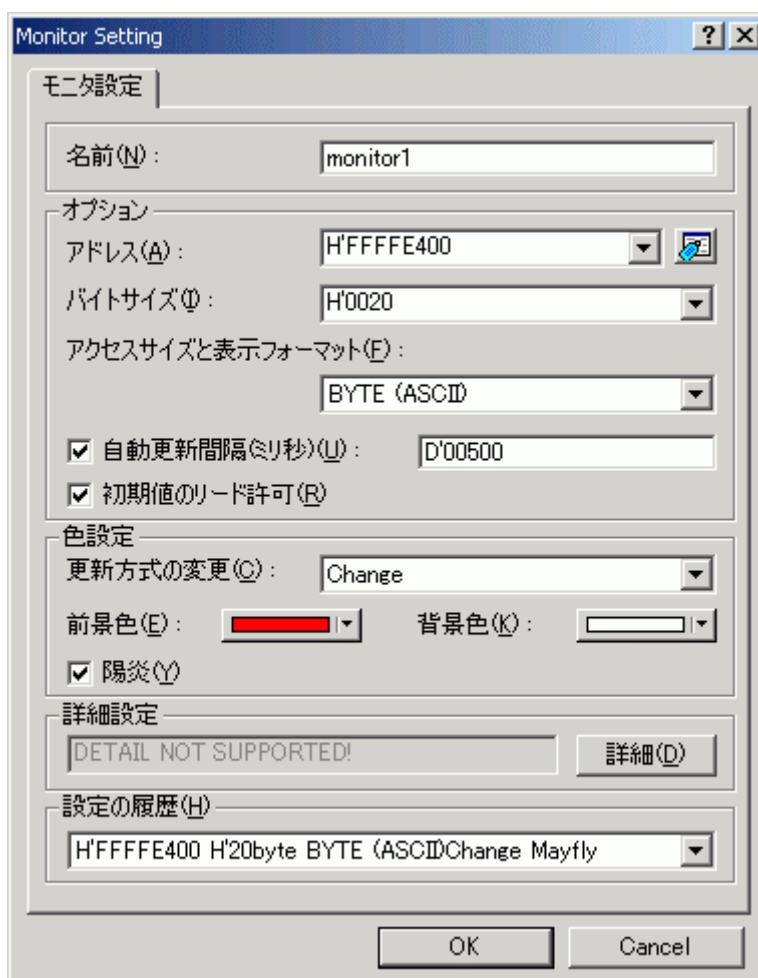


図4.61 Monitor Setting ダイアログボックス (設定完了時)

(3) [OK]ボタンをクリックします。[モニタ]ウィンドウが開きます。

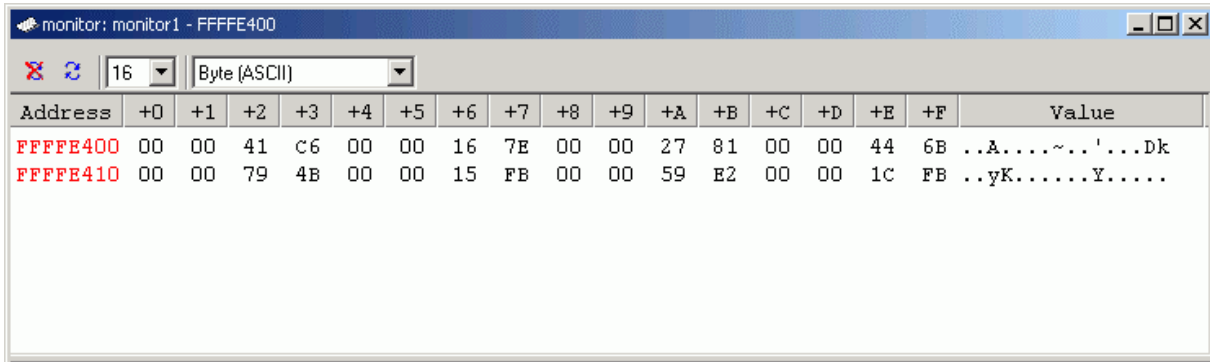


図4.62 モニタウィンドウ (実行前)

(4) [デバッグ]メニューから[リセット後実行]を選択してください。処理が実行されることによりメモリの内容が更新された場合は、更新されたメモリの値が赤 ([モニタ設定]ページの[前景色]ドロップダウンリストボックスと[背景色]ドロップダウンリストボックスで設定した色) に変更されます。また、更新が行われないメモリの値および、更新後一定間隔更新が行われなかったメモリの値は黒で表示されます。

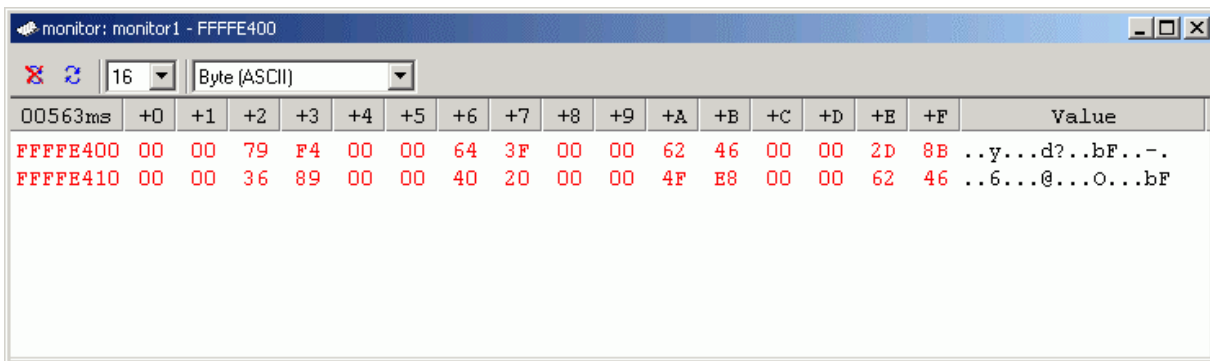


図4.63 モニタウィンドウ (実行中)

(5) [モニタ]ウィンドウの表示状況が確認できましたら、[デバッグ]メニューから[プログラムの停止]を選択し、実行を停止します。

4.20 さてつぎは？

このチュートリアルでは、E6000H エミュレータのいくつかの主な特徴と、High-performance Embedded Workshop の使い方を紹介しました。

E6000H エミュレータで提供されるエミュレーション機能を使用することによって、高度なデバッグを行うことができます。それによって、ハードウェアとソフトウェアの問題が発生する条件を正確に分離し、識別すると、それらの問題点を効果的に調査することができます。

5. 本製品固有のソフトウェア仕様と注意事項

この章では、E6000H エミュレータに関するソフトウェア仕様と注意事項について説明します。

5.1 対応ハードウェアについて

本エミュレータソフトウェアは SH7046 E6000H エミュレータ(HS7046EPH60H)に対応したソフトウェアです。

5.2 デバッグプラットフォームについて

本エミュレータで選択可能なデバッグプラットフォームは以下の通りです。

選択したデバッグプラットフォームによりエミュレーション可能なデバイスが異なります。

表5.1 選択可能ターゲット

デバッグプラットフォーム	備 考
SH7046 E6000H Emulator CPU SH-2	SH7046、SH7047、SH7144 シリーズのターゲットマイコンのエミュレーションを行います。

5.3 メモリ内容の表示と変更方法について

5.3.1 実行中のメモリ内容の表示と変更

本エミュレータでは、ユーザプログラム実行中のメモリ内容の表示と変更について以下の3つのアクセス方法があります。

表5.2 メモリ内容の表示と変更タイプ

アクセス方法	概 要	停止時間	表示	変更
モニタ機能	ユーザプログラムの実行を停止せずにメモリ内容の表示を自動更新します。	なし	可	不可
パラレルアクセス機能	ユーザプログラムの実行を一時的に停止します。	小	可	可
ショートブレーク機能	ユーザプログラムの実行を一時的に停止します。	大	可	可

また、それぞれのアクセス方法には、以下の特徴があります。

表5.3 メモリ内容の表示と変更タイプ

アクセス方法	対象ウィンドウ	対象メモリ領域
モニタ機能	[モニタ]ウィンドウ、モニタ機能を使用した [ウォッチ]ウィンドウのリアルタイム更新表示	ユーザプログラムが使用可能な全領域のうち指定した8ポイントまたは合計256バイトまでの領域。
パラレルアクセス機能	[モニタ]ウィンドウ以外のメモリ内容の表示を行うウィンドウ。	内蔵 ROM、エミュレーションメモリ。
ショートブレーク機能	[モニタ]ウィンドウ以外のメモリ内容の表示を行うウィンドウ。	内蔵 RAM、内蔵 I/O、ユーザメモリ。

- 【注】
- 1 [ウォッチ]ウィンドウにて、通常のデータリードによるリアルタイム更新を設定した場合、設定した対象メモリ領域によりアクセス方法が異なります。
 - 2 スリープモードおよびスタンバイモード中に、パラレルアクセス機能およびショートブレーク機能によるメモリ内容の表示と変更はできません。
 - 3 パラレルアクセス機能およびショートブレーク機能を使用した場合はリアルタイム性がなくなります。

5.3.2 パラレルアクセス機能停止時間参考値

ユーザプログラム実行中のメモリ内容の表示および変更のためのプログラム停止時間の参考値を示します。

表 5.4 停止時間の参考値

アクセス方法	条 件		停止時間
ショートブレーク	リード	内蔵 RAM への 256 バイト表示	3.5ms
		内蔵 RAM への 1 ロングワード リード	1.8ms
	ライト	内蔵 RAM への 1 ロングワード ライト	1.8ms
パラレルアクセス	リード	エミュレーションメモリに対するロングワード アクセスによるリード	220ns
	ライト	エミュレーションメモリに対するロングワード アクセスによるライト	220ns
Auto update Memory	ユーザプログラムの実行を停止しません。		

表 5.5 測定環境

項 目	設定値	
E6000H 設定	システムクロック (φ)	10.0MHz
	クロックモード	クロックモード 3
	JTAG クロック	10.0MHz
使用ホスト PC	CPU	Pentium® 1.5GHz
	PC インタフェース	PCI インタフェース

5.3.3 モニタ機能

- モニタ機能は最大 8 ポイントまたは合計 256 バイトまでの指定が可能です。
- 1 チャンネル 32 バイトのチャンネル 8 本のハード機構によりモニタ機能を実現しています。各チャンネルは 32 バイト境界で設定する必要があるため、32 バイト境界をまたぐ場合は、2 本のチャンネルを使用します。このため複数の 32 バイト境界をまたぐポイントを指定した場合は、設定できる合計が 256 バイト未満になります。
- ユーザプログラム実行中に、モニタ機能の設定または変更を行った場合、ユーザプログラムのリアルタイム性はなくなります。
- ユーザプログラム実行中に、[アクセスサイズと表示フォーマット]の変更を行った場合、ユーザプログラムのリアルタイム性はなくなります。

5.4 プログラムの実行について

5.4.1 ステップ実行

- ステップの実行では、ブレークが無効になります。ただしトリガは出力されます。

5.5 イベント機能について

5.5.1 ソフトウェアブ레이크

- ソフトウェアブ레이크ポイントは命令を置き換えることにより実現しますので、RAM(エミュレーションRAMを含む)領域にのみ設定できます。また、次に示すアドレスには、指定できません。

メモリ内容が H'0000 であるアドレス

- ソフトウェアブ레이크ポイントのアドレスの内容は、ユーザプログラムで変更しないでください。
- ユーザプログラム実行中にソフトウェアブ레이크ポイントを設定したアドレスの内容はブ레이크命令に置き換えられています。
- ソフトウェアブ레이크ポイントの設定数と [プログラム実行] ダイアログボックスの [テンポラリ PC ブ레이크ポイント] の設定数の合計は、最大 255 個です。したがって、ソフトウェアブ레이크ポイントを 255 個設定した状態では、[プログラム実行] ダイアログボックスの [テンポラリ PC ブ레이크ポイント] での指定は無効となります。ソフトウェアブ레이크ポイントと [プログラム実行] ダイアログボックスの [テンポラリ PC ブ레이크ポイント] は、設定数の合計が 255 個未満で使用してください。
- 遅延分岐命令の直後 (スロット命令) にはブ레이크を設定しないでください。ブ레이크を設定した場合、遅延分岐命令を実行した時点でスロット不当命令割り込みが発生し、ブ레이크しません。

5.5.2 オンチップブ레이크

- 条件成立回数はチャンネル 4 のみ指定可能です。
- リセットポイントはシーケンシャルブ레이크を有効にした場合のみ意味を持ちます。
- シーケンシャルブ레이크とブ레이크の範囲指定 ([アドレス範囲ブ레이크]) を同時に設定した場合はシーケンシャルブ레이크が有効になります。
- アドレス条件およびデータ条件は、アドレスバスおよびデータバスの値が一致したバスサイクルを検索します。

ロングワードアクセス

1 回のバスサイクルでロングワードデータをリード/ライトします。
データ条件はロングワードデータの指定のみ有効となります。アドレス条件は、4 の倍数のみ有効となります。アクセスサイズ条件はロングワードの指定のみ有効となります。

ワードアクセス

1 回のバスサイクルでワードデータをリード/ライトします。アドレス条件は、2 の倍数のみ有効となります。アクセスサイズ条件はワードの指定のみ有効となります。

バイトアクセス

1 回のバスサイクルでバイトデータをリード/ライトします。データ条件はバイトデータの指定のみ有効となります。アドレス条件は偶数と奇数どちらも有効となります。

5.5.3 オンエミュレータブ레이크

条件成立からブ레이크が発生するまで数サイクルかかります。

条件成立からブ레이크が発生するまで数サイクルかかります。

アドレスバス条件およびデータバス条件は、アドレスバスおよびデータバスの値が一致したバスサイクルを検索します。したがって、次のことを考慮して検索条件を設定してください。

- 32ビットバス領域

ロングワードアクセス

1 回のバスサイクルでロングワードデータをリード/ライトします。
データ条件はロングワードデータの指定のみ有効となります。アドレス条件は、4 の倍数のみ有効となります。

ワードアクセス

1 回のバスサイクルでワードデータをリード/ライトします。データ条件はワードデータの指定のみ有効となります。アドレス条件は 2 の倍数のみ有効となります。

バイトアクセス

1 回のバスサイクルでバイトデータをリード/ライトします。データ条件はバイトデータの指定のみ有効となります。アドレス条件は偶数と奇数どちらも有効となります。

5 本製品固有のソフトウェア仕様と注意事項

- 16ビットバス領域

ロングワードアクセス

この領域をロングでアクセスすると2回のワードサイクルでデータをリード/ライトします。データ条件はワードデータの指定のみ有効となります。また、アドレス条件は、2の倍数のみ有効となります。

ワードアクセス

1回のバスサイクルでワードデータをリード/ライトします。データ条件はワードデータの指定のみ有効となります。アドレス条件は2の倍数のみ有効となります。

バイトアクセス

1回のバスサイクルでバイトデータをリード/ライトします。データ条件はバイトデータの指定のみ有効となります。アドレス条件は偶数と奇数どちらも有効となります。

- 8ビットバス領域

この領域は全てバイトアクセスになります。(ロングワードでアクセス時は4回、ワードでアクセス時は2回のバイトアクセスとなります。)アドレス条件は、偶数と奇数どちらも有効となりますがデータ条件はバイトデータ指定のみ有効となります。

5.6 トレース機能について

5.6.1 トレース表示内容

- [トレース取得プロパティ]ダイアログボックスの[計測時間単位]設定をユーザクロック(システムクロック)にした場合、タイムスタンプ表示は32ビット16進表示になります。

5.6.2 トレース取得条件

- 条件成立からトレース停止が発生するまで数サイクルかかります。
- シーケンシャルトレース停止の各通過ポイントおよびリセット条件の成立間隔は、6バスサイクル以上あけてください。
- [トレース取得プロパティ]ダイアログボックスの[トリガ出力間の時間計測]で[チャンネル1->2]、[チャンネル3->4]、[チャンネル5->6]を選択した場合、測定スタート - 測定エンドを6バスサイクル以上離して設定してください。
- [トレース取得プロパティ]ダイアログボックスの[トリガ出力間の時間計測]で[チャンネル7->8]を選択した場合、測定スタート - 測定エンドを15バスサイクル以上離して設定してください。
- トレース停止モード条件は実行開始から6バスサイクル以上離して成立するように設定してください。
- 設定したアドレス条件を実行開始PCとしてユーザプログラムを実行すると正常にシーケンシャルブレーク/トレースストップしない場合があります。設定したアドレス条件の次の命令を実行開始PCとしてユーザプログラムを実行してください。
- パフォーマンス測定機能のチャンネル1を使用している場合は、2点間トレース測定は設定できません。

5.6.3 トレース情報の検索

- [General]ページにおいて、検索範囲入力時、[開始ポイント]オプションに検索を終了するPTRの値、[終了ポイント]オプションに検索を開始するPTRの値を指定することも可能です。
- [トレース取得プロパティ]ダイアログボックスの[計測時間単位]設定をユーザクロック(システムクロック)にした場合、タイムスタンプの検索はできません。

5.6.4 トレース情報のフィルタ

- トレース情報をフィルタリングした場合、保存する範囲の指定はできません。フィルタリングした結果[トレース]ウィンドウに表示されたトレース情報すべてを保存します。保存する範囲を指定したい場合は[Trace Filter]ダイアログボックスの[General]ページよりフィルタ範囲を指定してください。
- [トレース取得プロパティ]ダイアログボックスの[計測時間単位]設定をユーザクロック(システムクロック)にした場合、タイムスタンプのフィルタはできません。

5.7 モニタ機能について

- 前景色および背景色の設定はご使用のオペレーティングシステムにより使用できない場合があります。

5.8 パフォーマンス測定機能について

5.8.1 測定誤差について

パフォーマンス測定機能では、以下の測定誤差が発生します。

- ± 1 分解能の誤差（分解能が 20ns の場合は、 ± 20 ns の誤差が発生します。）
 ± 1 分解能の誤差が発生するケースとしては、ユーザプログラム実行開始時、ユーザプログラム実行停止時（ブレーク）、開始条件成立時、および終了条件成立時があります。
- 測定用の水晶発振モジュールの周波数安定度： $\pm 0.01\%$

5.8.2 留意事項

- それぞれの測定モードにおいて、終了条件成立から次の開始条件成立までの時間間隔が、1 分解能より大きくなるようにしてください。この間隔が、1 分解能未満の場合は、終了条件成立から次の開始条件成立までの時間も計測に含まれます。
- [指定範囲内時間測定]は指定範囲からアドレスがはずれた場合を終了条件成立とします。これに対し、[指定アドレス間時間測定]および、[指定アドレス範囲間時間測定]は、指定条件が成立した時点で終了条件成立とします。このためこれらに同じアドレスを指定した場合、[指定範囲内時間測定]の計測結果は、[指定アドレス間時間測定]および、[指定アドレス範囲間時間測定]より大きくなります。
- 各測定は、プリフェッチサイクルのアドレスバスの値を用いて行っています。分岐命令または遅延スロット命令の次の命令に隣接したアドレスに終了アドレス条件を指定すると正しく測定が行えません。あらかじめバストレースの表示で、分岐命令をプリフェッチしたサイクル以降の動作を確認し、分岐により実行されないプリフェッチサイクルのアドレスを終了アドレスに設定してください。
- 2 点間トレースを設定している場合は、パフォーマンス測定機能のチャンネル 1 は設定できません。
- パフォーマンス測定機能の分解能は[コンフィグレーションプロパティ]ダイアログボックスの[時間計測単位]にて設定します。分解能にクロックカウンタを設定した場合は、[RUN-TIME]および [MAX-MIN-TIME]の値はクロックカウンタ値(16 進数表示)になります。測定用カウンタは 24 ビットあり、最大測定時間は[時間計測単位]の設定値により下記の時間になります。

[Timer Resolution]の設定値	最大測定時間
52us	約 14 分
1.6us	約 26 秒
20ns	約 0.33 秒

- 領域アクセス回数測定および指定範囲内コール回数測定の最大測定回数は 65535 になります。

5.9 プロファイル機能について

- 最適化リンケージエディタが出力するスタック使用量情報ファイル(拡張子は".sni")がない場合、プロファイルデータの測定時に実行された関数のみを表示します。スタック使用量情報ファイルについては、最適化リンケージエディタのマニュアルを参照してください。
- スタックサイズは実際の値とは異なります。関数の呼び出し経路における目安としてください。また、最適化リンケージエディタが出力するスタック使用量情報ファイル(拡張子は".sni")がない場合には、スタックサイズを表示しません。
- プロファイラ機能使用時、イベント機能のソフトウェアブレークとエミュレータブレーク、およびオンチップブレークのチャンネル1からチャンネル4は使用できません。
- プロファイラ機能使用時、実行時のパラレルアクセス機能は使用できません。
- 本機能は、プログラム実行途中で内部的にブレークを発生させて実現していますので、リアルタイム性はありません。また、測定には誤差があります。
- [Cycles]項目の表示は、実行時間の測定用カウンタの値(10進数表示)になります。実行時間の測定用カウンタの分解能は、[コンフィグレーションプロパティ]ダイアログボックスの[時間計測単位]にて設定します。各関数の実行時間は、下記の計算式になります。

実行時間 = [Cycles]項目の値 × [時間計測単位]の値

- 本機能を有効にする時はあらかじめ、[コンフィグレーションプロパティ]ダイアログボックスにて[プログラム実行中のメモリアクセス許可]を無効にする必要があります。

5.10 入力形式について

5.10.1 マスク入力

アドレスバス条件およびデータバス条件では、マスク入力が指定可能です。アドレスの指定では、1ビット、3ビットまたは4ビット単位でマスク指定を行うことができます。マスクしたビットはどんな値でも条件が成立することになります。

アドレスバス条件でマスクを指定する場合は、[マスク指定有効]エリアにマスク値を指定します。

データ条件でマスクを指定する場合も同様に、[マスク指定有効]エリアにマスク値を指定します。

マスク指定の方法は、値の入力時に無視したい桁と同じビットに1を代入し、指定します。マスク指定の例を示します。

表 5.6 アドレスマスク指定

項番	入力値	マスク単位	使用例	マスク位置
1	2進数	1ビット	B'00000111	ビット2~0をマスク
2	8進数	3ビット	O'000017	ビット3~0をマスク
3	16進数	4ビット	H'07FF	ビット10~0をマスク

5.11 チュートリアルプログラムについて

5.11.1 チュートリアルプログラムの操作についての注意事項

本製品を使用して「4 チュートリアル」の説明に沿って High-performance Embedded Workshop を操作する場合は、下記の内容にて操作を追加または変更してください。

4.6 プログラムの実行	実行を行う前に、ターゲットマイコンをリセットしてください。ターゲットマイコンのリセット方法は「4.14 ターゲットマイコンのリセット」を参照してください。
4.19 ステップオーバーコマンドの実行	本製品ではステップアウト実行後の停止位置が sort 関数の呼び出し行になります。ステップオーバーコマンドを実行する前に、一度ステップインコマンドを実行し停止位置を change 関数の呼び出し行に移動してください。

5.12 メモリマップ機能について

5.12.1 エミュレーションメモリ

- (1) 本エミュレータはエリアコントロールを図5.1および図5.2のメモリブロックに分けて管理しています。ユーザシステム上のメモリとエミュレータのエミュレーションメモリを1ブロックに共存させることはできません。
- (2) エミュレーションメモリ使用時は、エミュレーションメモリ割り付けエリアにWAITステートサイクルが50MHz動作時、1サイクル以上必要です。エミュレーションメモリをご使用になる際には、以下の表を参照してバスステートコントローラにウェイト数を設定してください。

外部動作周波数	必要ウェイト数
50MHz	1 ウェイト以上
40MHz 以下	0 ウェイト

5.12.2 メモリマップ制御

メモリマップは、CS 空間ごとに 128k バイト、1M バイトまたは 2M バイトのブロックのいずれかに分割されています。アクセス属性の変更は各ブロック単位に設定することができます。各 CS 図空間で使用できるブロックのサイズは図 5.1 または図 5.2 を参照してください。

本エミュレータに搭載しているエミュレーションメモリは 4MB です。4MB のエミュレーションメモリを 128k バイト (16 個) および 1M バイト (2 個) に分割して使用することができます。

エミュレーションメモリの設定も、アクセス属性と同様にブロック単位 (図 5.1 または図 5.2) に設定することができます。1M バイトまたは 2M バイトのブロックに、複数個の 128k バイトのエミュレーションメモリを設定することができます。ただし、割り付けたブロックの残りのエリアは使用することができなくなります。1M バイトのエミュレーションメモリを 1 つだけ 2M バイトのブロックに割り付けた場合も同様です。

5 本製品固有のソフトウェア仕様と注意事項

内蔵ROM無効モード時メモリマップ

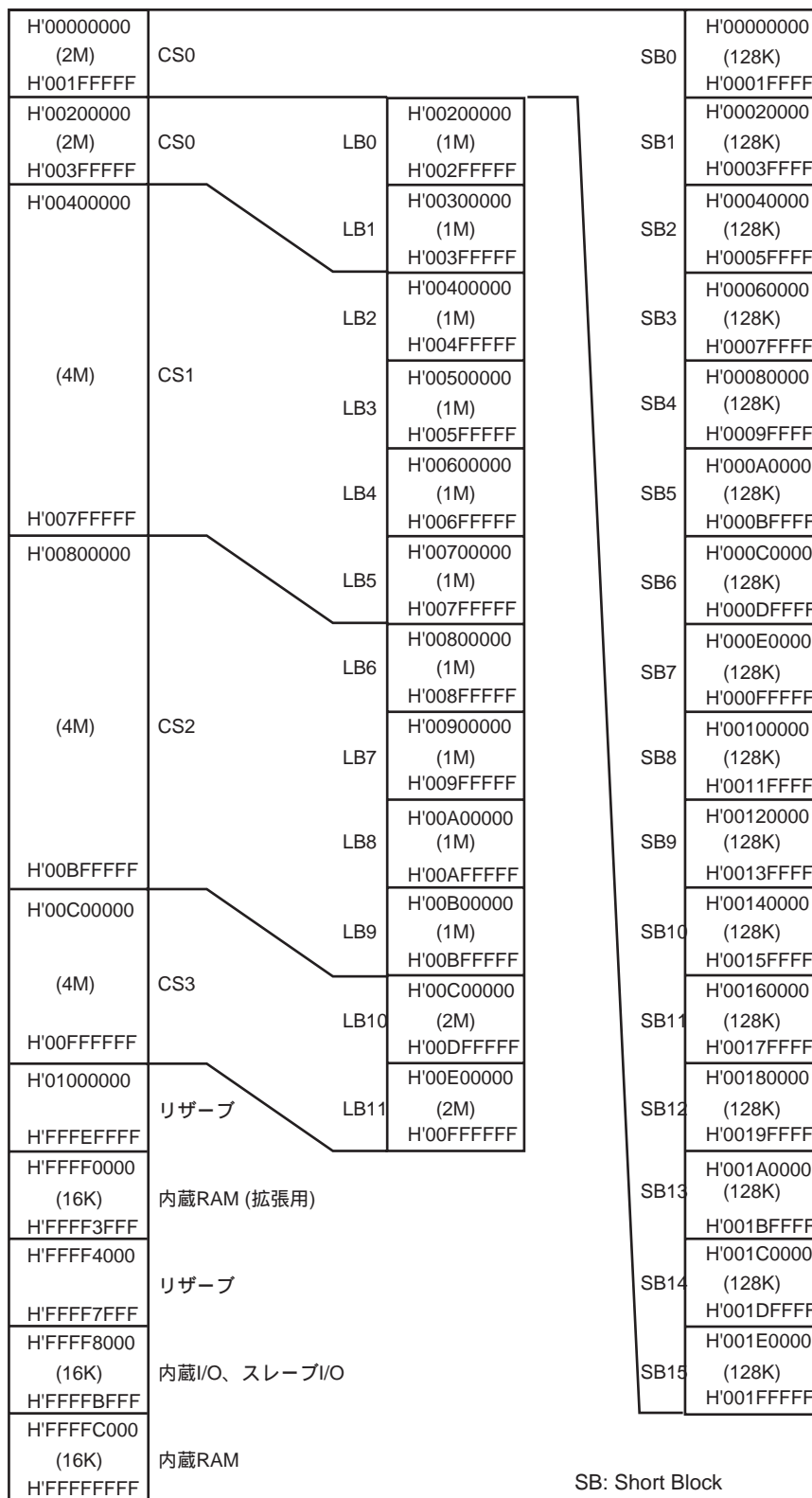


図 5.1 内蔵 ROM 無効モード時メモリマップ

内蔵ROM有効モード時メモリマップ

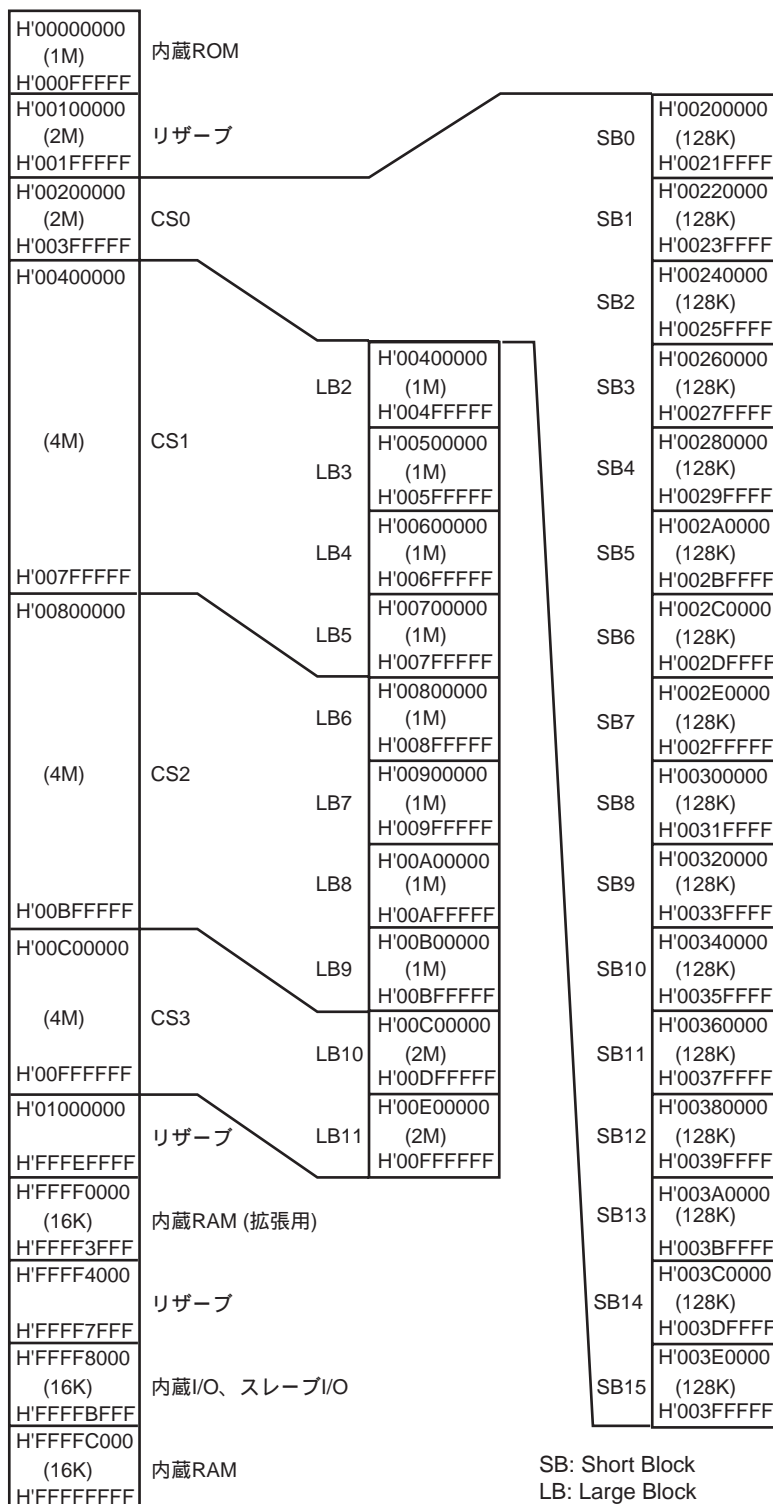


図 5.2. 内蔵 ROM 有効モード時メモリマップ

6. エラーメッセージ

6.1 E6000H のエラーメッセージ

6.1.1 起動時のエラーメッセージ

エミュレータの起動時にエラーが発生した場合、E6000H からエラー出力されます。High-performance Embedded Workshop では、専用のメッセージダイアログボックスに出力します。

表 6.1 に E6000H エミュレータ起動時のエラーメッセージを示します。

表6.1 起動時のエラーメッセージ

エラーメッセージ	内容と対策
There is no configuration file.	E6000H エミュレータの起動に必要な、Config file がありません。 High-performance Embedded Workshop を終了し、High-performance Embedded Workshop をインストールし直してください。さらにユーザーケーブルを接続し直して E6000H エミュレータの電源を入れて、High-performance Embedded Workshop を再起動してください。本現象が解決できない場合は、発生状況をご連絡ください。
The contents of the configuration file are incorrect.	E6000H エミュレータの起動に必要な、Config file が不正です。 High-performance Embedded Workshop を終了し、High-performance Embedded Workshop をインストールし直してください。さらにユーザーケーブルを接続し直して E6000H エミュレータの電源を入れて、High-performance Embedded Workshop を再起動してください。本現象が解決できない場合は、発生状況をご連絡ください。
Main Board not Support (XX XX XX) Emulator is switched off or not connected	E6000H エミュレータの電源が入っていないか、ユーザーケーブルが接続されていません。 High-performance Embedded Workshop を終了し、ユーザーケーブルを接続し直して E6000H エミュレータの電源を入れて、High-performance Embedded Workshop を再起動してください。本現象が解決できない場合は、発生状況をご連絡ください。
Emulation Board not Support (XX XX XX) Emulator is switched off or not connected	E6000H エミュレータの電源が入っていないか、ユーザーケーブルが接続されていません。 High-performance Embedded Workshop を終了し、ユーザーケーブルを接続し直して E6000H エミュレータの電源を入れて、High-performance Embedded Workshop を再起動してください。本現象が解決できない場合は、発生状況をご連絡ください。
EVA chip Board not Support (XX XX XX) Emulator is switched off or not connected	E6000H エミュレータの電源が入っていないか、ユーザーケーブルが接続されていません。 High-performance Embedded Workshop を終了し、ユーザーケーブルを接続し直して E6000H エミュレータの電源を入れて、High-performance Embedded Workshop を再起動してください。本現象が解決できない場合は、発生状況をご連絡ください。
Can't initialize G/A registers	E6000H エミュレータ本体の初期化に失敗しました。High-performance Embedded Workshop を終了し、ユーザーケーブルを接続し直して E6000H エミュレータの電源を入れて、High-performance Embedded Workshop を再起動してください。本現象が解決できない場合は、発生状況をご連絡ください。
There is no effective clock source	有効なクロックソースが見つかりません。有効なクロックソースをを正しく設定し直してください。
This mode can not specify	ターゲットボードのモード端子状態が異常です。動作モード 4 で起動します。モード端子を正しく設定し直してください。
Can't find firmware file Firmware open Error Firmware Download Error Firmware Name Error	E6000H エミュレータの起動に必要なファイルに異常があります。 High-performance Embedded Workshop を終了し、High-performance Embedded Workshop をインストールし直してください。さらにユーザーケーブルを接続し直して E6000H エミュレータの電源を入れて、High-performance Embedded Workshop を再起動してください。本現象が解決できない場合は、発生状況をご連絡ください。
Failed to receive a firmware initialization command.	E6000H エミュレータのファームウェアの起動に失敗しました。 High-performance Embedded Workshop を終了し、ユーザーケーブルを接続し直して E6000H エミュレータの電源を入れて、High-performance Embedded Workshop を再起動してください。本現象が解決できない場合は、発生状況をご連絡ください。
Target system is Vcc down	Vcc が設定したしきい値以下です。
JTAG Timeout Svrval Error JTAG Packet Receive Error	High-performance Embedded Workshop を終了し、ユーザーケーブルを接続し直して E6000H エミュレータの電源を入れて、High-performance Embedded Workshop を再起動してください。

6.1.2 実行中のエラーメッセージ

エミュレーションの実行中にエラーが発生した場合、E6000H からエラー出力されます。High-performance Embedded Workshop では、専用のメッセージダイアログボックスに出力するほか、ステータスバーにメッセージが出力されます。

表 6.2にメッセージダイアログボックスに出力される E6000H エミュレーションの実行中のエラーメッセージを示します。





















表6.2 メッセージダイアログボックスに出力される実行中のエラーメッセージ

エラーメッセージ	内容と対策
Communication DLL error. Communication Timeout error.	E6000H エミュレータの電源が落ちた、もしくは通信障害が発生しています。High-performance Embedded Workshop を終了し、ユーザーケーブルを接続し直して E6000H エミュレータの電源を入れて、High-performance Embedded Workshop を再起動してください。本現象が解決できない場合は、発生状況をご連絡ください。
Parallel Access Error	パラレルアクセスでエラーが発生しました。一度ブレイクするまで、パラレルアクセスは禁止になります。

付録 A メニュー一覧

GUIメニューの一覧を表 A.1に示します。

表 A-1 GUI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	備考	
表示	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	
	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+U		[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
		ステータス	Ctrl+U		[ステータス]ウィンドウを表示します
		拡張モニタ			[拡張モニタ]ウィンドウを表示します
		モニタ	設定...	Shift+Ctrl+E	
		ウィンドウの選択...			[モニタ]ウィンドウの一覧表示、および追加 / 編集等を行う[ウィンドウの選択]ダイアログボックスを表示します
	シンボル	ラベル	Shift+Ctrl+A		[ラベル]ウィンドウを表示します
		ウォッチ	Ctrl+W		[ウォッチ]ウィンドウを表示します
		ローカル	Shift+Ctrl+W		[ローカル]ウィンドウを表示します
	コード	イベントポイント	Ctrl+E		[イベントポイント]ウィンドウを表示します
		トレース	Ctrl+T		[トレース]ウィンドウを表示します
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	
グラフィック	画像...	Shift+Ctrl+G		[画像]ウィンドウを表示します	
	波形...	Shift+Ctrl+V		[波形]ウィンドウを表示します	
パフォーマンス	パフォーマンス解析	Shift+Ctrl+P		[パフォーマンス解析]ウィンドウを表示します	
	プロファイル	Shift+Ctrl+F		[プロファイル]ウィンドウを表示します	

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	備考	
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			ターゲットハードウェアをリセットし、PC をリセットベクタアドレスに設定します	
	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		ターゲットハードウェアをリセットし、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 からテキストカーソルの位置までユーザプログラムを実行します	
	カーソル位置を PC 値に設定			テキストカーソルの位置に PC を設定します	
	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 位置表示	Shift+Ctrl+Y		現在の PC がある[エディタ] または[逆アセンブリ] ウィンドウをオープンします	
	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一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			ソースライン一行だけをステップ実行します
	プログラムの停止	Esc		ユーザプログラムの実行を停止します	
	初期化			本エミュレータを切断し、再接続します	
	接続			本エミュレータを接続します	
	接続解除			本エミュレータを切断します	
	メモリの保存			任意のアドレス領域を保存します	
	メモリのペリファイ			アドレス領域を検証します	
	オーバーレイの構成...			オーバーレイ使用時の優先セクショングループの設定を行います	
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		

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	備考	
基本設定	カスタマイズ...			High-performance Embedded Workshop 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...			High-performance Embedded Workshop 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			数値の表示 / 入力時の基数のデフォルト設定を 16 進数とします
		10 進数			数値の表示 / 入力時の基数のデフォルト設定を 10 進数とします
		8 進数			数値の表示 / 入力時の基数のデフォルト設定を 8 進数とします
		2 進数			数値の表示 / 入力時の基数のデフォルト設定を 2 進数とします
エミュレータ	システム...			本エミュレータの設定を行う[コンフィグレーションプロパティ]ダイアログボックスを表示します	
	メモリリソース...			本エミュレータのメモリマップの表示、および設定を行う[メモリマップ]ダイアログボックスを表示します	

付録 B コマンドライン一覧

コマンド一覧を表 B.1 に示します。

表 B.1 コマンド一覧

項番	コマンド名	短縮形	説明
1	!	-	コメント
2	ADD_FILE	AF	カレントプロジェクトにファイルを追加します
3	ANALYSIS	AN	性能分析機能の有効化 / 無効化
4	ANALYSIS_RANGE	AR	性能分析範囲の設定、表示
5	ANALYSIS_RANGE_DELETE	AD	性能分析範囲の削除
6	ASSEMBLE	AS	アセンブルの実行
7	ASSERT	-	コンディションのチェック
8	AUTO_COMPLETE	AC	オートコンプリート機能の有効化 / 無効化
9	BREAKPOINT_ONCHIP	BC	オンチップブレークポイントの表示、シーケンシャルブレークの設定および PtoP 時間測定の設定
10	BREAKPOINT_ONCHIPn	BCn	オンチップブレークポイントの各チャンネル設定
11	BREAKPOINT_ONCHIP_CLEAR	BCC	オンチップブレークポイントの削除
12	BREAKPOINT_ONCHIP_ENABLE	BCE	オンチップブレークポイントの有効 / 無効の切り換え
13	BREAKPOINT_ONEMULATOR	BE	オンエミュレータブレークポイントの表示
14	BREAKPOINT_ONEMULATORn	BE n	オンエミュレータブレークポイントの各チャンネル設定
15	BREAKPOINT_ONEMULATOR_CLEAR	BEC	オンエミュレータブレークポイントの削除
16	BREAKPOINT_ONEMULATOR_ENABLE	BEE	オンエミュレータブレークポイントの有効 / 無効の切り換え
17	BREAKPOINT_SOFTWARE	BS	ソフトウェアブレークポイントの設定
18	BREAKPOINT_SOFTWARE_CLEAR	BSC	ソフトウェアブレークポイントの削除
19	BREAKPOINT_SOFTWARE_ENABLE	BSE	ソフトウェアブレークポイントの有効 / 無効の切り換え
20	BUILD	BU	カレントプロジェクトのビルド処理を開始します
21	BUILD_ALL	BL	カレントプロジェクトのすべてのビルド処理を開始します
22	CHANGE_CONFIGURATION	CC	コンフィギュレーションの設定
23	CHANGE_PROJECT	CP	プロジェクトの設定
24	CHANGE_SESSION	CS	セッションの設定
25	CLOSE_WORKSPACE	CW	ワークスペースを閉じる
26	CLOCK_MODE	CKM	クロックモードの設定および表示
27	CONFIGURE_PLATFORM	CPF	プラットフォームの設定
28	COVERAGE_DISPLAY	CVD	カバレッジ情報の表示
29	COVERAGE_CLEAR	CVC	カバレッジ情報の初期化
30	COVERAGE_SET	CVS	カバレッジ情報の設定・設定内容の変更
31	DEFAULT_OBJECT_FORMAT	DO	デフォルトオブジェクト (プログラム) フォーマットの設定
32	DEVICE_TYPE	DE	デバイスタイプの選択
33	DISASSEMBLE	DA	逆アセンブル表示
34	EMULATOR_CLOCK	ECK	ターゲットマイコンの入力クロック設定
35	ERASE	ER	コマンドラインウィンドウの内容クリア
36	EVALUATE	EV	式の計算
37	EXMONITOR_DISPLAY	EXMD	拡張モニタの内容表示
38	EXMONITOR_SET	EXMS	拡張モニタ項目の表示 / 非表示の切り換え
39	EXMONITOR_SETRATE	EXMSR	実行中およびブレーク中の拡張モニタ更新時間の設定
40	FILE_LOAD	FL	オブジェクト (プログラム) ファイルのロード
41	FILE_SAVE	FS	メモリ内容のファイルセーブ
42	FILE_UNLOAD	FU	ファイルをアンロードします
43	FILE_VERIFY	FV	ファイル内容とメモリ内容の比較

付録 B コマンドライン一覧

項番	コマンド名	短縮形	説明
44	GENERATE_MAKE_FILE	GM	High-performance Embedded Workshop 外でビルドするための make ファイルを生成します
45	GO	GO	ユーザプログラムの実行
46	GO_RESET	GR	リセットベクタからのユーザプログラムの実行
47	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
48	HALT	HA	ユーザプログラムの停止
49	HELP	HE	コマンドラインのヘルプ表示
50	INITIALIZE	IN	デバッグプラットフォームの初期化
51	JTAG_CLOCK	JCK	JTAG クロック (TCK) の設定および表示
52	LOG	LO	ロギングファイルの操作
53	MAP_DISPLAY	MA	メモリマッピング情報の表示
54	MAP_SET	MS	メモリマッピングの設定
55	MEMORY_COMPARE	MC	メモリ内容の比較
56	MEMORY_DISPLAY	MD	メモリ内容の表示
57	MEMORY_EDIT	ME	メモリ内容の変更
58	MEMORY_FILL	MF	指定データによるメモリ内容の一括変更
59	MEMORY_FIND	MI	メモリ範囲内でデータを検索します
60	MEMORY_MOVE	MV	メモリブロックの移動
61	MEMORY_TEST	MT	メモリブロックのテスト
62	MODE	MO	MCU モードを設定または表示
63	MONITOR_CLEAR	MOC	モニタポイントの削除
64	MONITOR_DISPLAY	MOD	モニタ内容の表示
65	MONITOR_REFRESH	MOR	モニタ内容に自動更新制御
66	MONITOR_SET	MOS	モニタポイントの設定と表示
67	OPEN_WORKSPACE	OW	ワークスペースのオープン
68	PINSEL	PSL	ピンセレクト設定内容の表示
69	PINSEL_xxxx	PSLxxxx	ピンセレクトの設定 (xxxx : 端子名)
70	PROFILE	PR	プロファイルの有効化/無効化
71	PROFILE_DISPLAY	PD	プロファイル情報の表示
72	PROFILE_SAVE	PS	プロファイル情報のファイル出力
73	QUIT	QU	High-performance Embedded Workshop の終了
74	RADIX	RA	入力基数の設定
75	REFRESH	RF	メモリ関連ウィンドウの更新
76	REGISTER_DISPLAY	RD	CPU レジスタ値の表示
77	REGISTER_SET	RS	CPU レジスタ値の設定
78	REMOVE_FILE	REM	指定ファイルをカレントプロジェクトから削除します
79	RESET	RE	CPU のリセット
80	SAVE_SESSION	SE	現在のセッションをセーブ
81	SLEEP	-	コマンド実行の遅延
82	STEP	ST	ステップ実行 (命令単位またはソース行単位)
83	STEP_MODE	SM	ステップモードの選択
84	STEP_OUT	SP	PC 位置の関数を終了するまでのステップ実行
85	STEP_OVER	SO	ステップオーバー実行
86	STEP_RATE	SR	ステップ実行速度の設定、表示
87	SUBMIT	SU	エミュレータコマンドファイルの実行
88	SYMBOL_ADD	SA	シンボルの設定
89	SYMBOL_CLEAR	SC	シンボルの削除
90	SYMBOL_LOAD	SL	シンボル情報ファイルのロード
91	SYMBOL_SAVE	SS	シンボル情報のファイルセーブ
92	SYMBOL_VIEW	SV	シンボルの表示
93	STATUS	STS	[ステータス] ウィンドウの内容表示
94	SAVE_WORKSPACE	SW	現在のワークスペースの保存
95	TCL	-	TCL の有効 / 無効の切り換え

項番	コマンド名	短縮形	説明
96	TIMER	TI	実行時間測定タイマの分解能の設定、表示
97	TOOL_INFORMATION	TO	現在登録されているツールの情報をファイルへ出力
98	TRACE	TR	トレース情報の表示
99	TRACE_ACQUISITION	TA	トレース取得条件の設定、表示
100	TRACE_ACQUISITIONn	TAn	トレース取得条件の各チャンネルおよび PtoP ポイントの設定
101	TRACE_ACQUISITION_CLEAR	TAC	トレース取得条件の削除
102	TRACE_BINARY_COMPARE	TBC	トレースバイナリファイルと現在のトレース情報の比較
103	TRACE_BINARY_SAVE	TBV	トレース情報をバイナリファイルに保存
104	TRACE_FILTER	TF	トレース情報のフィルタ
105	TRACE_SAVE	TV	トレース情報をテキストファイルに保存
106	TRACE_STATISTIC	TST	統計情報解析の実行
107	UPDATE_ALL_DEPENDENCIES	UD	カレントプロジェクトの依存関係を更新します
108	USER_SIGNALS	US	ユーザシグナル情報の有効/無効の切り換え
109	WATCH_ADD	WA	Watch アイテムの追加
110	WATCH_AUTO_UPDATE	WU	Watch アイテムの自動更新の設定または解除
111	WATCH_DELETE	WD	Watch アイテムの削除
112	WATCH_DISPLAY	WI	ウォッチウィンドウの内容の表示
113	WATCH_EDIT	WE	Watch アイテムの値の編集
114	WATCH_EXPAND	WX	Watch アイテムの展開または縮小
115	WATCH_RADIX	WR	Watch アイテムの表示基数の変更
116	WATCH_SAVE	WS	ウォッチウィンドウの表示内容をファイルに保存

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ください。

ルネサスマイクロコンピュータ開発環境システム
ユーザーズマニュアル
SH7046 E6000H エミュレータ

発行年月日 2005年10月21日 Rev.4.00
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
グローバルストラテジックコミュニケーション本部
カスタマサポート部

営業お問合せ窓口
株式会社ルネサス販売



<http://www.renesas.com>

本			社	〒100-0004	千代田区大手町2-6-2 (日本ビル)	(03) 5201-5350
京			社	〒212-0058	川崎市幸区鹿島田890-12 (新川崎三井ビル)	(044) 549-1662
西	浜	支	社	〒190-0023	立川市柴崎町2-2-23 (第二高島ビル2F)	(042) 524-8701
東	東	支	社	〒980-0013	仙台市青葉区花京院1-1-20 (花京院スクエア13F)	(022) 221-1351
い	北	支	店	〒970-8026	いわき市平小太郎町4-9 (平小太郎ビル)	(0246) 22-3222
茨	わ	支	店	〒312-0034	ひたちなか市堀口832-2 (日立システムプラザ勝田1F)	(029) 271-9411
新	城	支	店	〒950-0087	新潟市東大通1-4-2 (新潟三井物産ビル3F)	(025) 241-4361
松	潟	支	社	〒390-0815	松本市深志1-2-11 (昭和ビル7F)	(0263) 33-6622
中	本	支	社	〒460-0008	名古屋市中区栄4-2-29 (名古屋広小路ブレイス)	(052) 249-3330
関	西	支	社	〒541-0044	大阪市中央区伏見町4-1-1 (明治安田生命大阪御堂筋ビル)	(06) 6233-9500
北	陸	支	社	〒920-0031	金沢市広岡3-1-1 (金沢パークビル8F)	(076) 233-5980
広	島	支	店	〒730-0036	広島市中区袋町5-25 (広島袋町ビルディング8F)	(082) 244-2570
鳥	取	支	店	〒680-0822	鳥取市今町2-251 (日本生命鳥取駅前ビル)	(0857) 21-1915
九	州	支	社	〒812-0011	福岡市博多区博多駅前2-17-1 (ヒロカネビル本館5F)	(092) 481-7695

■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：コンタクトセンタ E-Mail: csc@renesas.com

SH7046 E6000H エミュレータ
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

RJJ10J1381-0400