

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事務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、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）
特定水準： 航空機器、航空宇宙機器、海中継機器、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。

お客様各位

資料中の「日立製作所」、「日立XX」等名称の株式会社ルネサス テクノロジへの変更について

2003年4月1日を以って三菱電機株式会社及び株式会社日立製作所のマイコン、ロジック、アナログ、ディスクリット半導体、及びDRAMを除くメモリ(フラッシュメモリ・SRAM等)を含む半導体事業は株式会社ルネサス テクノロジに承継されました。従いまして、本資料中には「日立製作所」、「株式会社日立製作所」、「日立半導体」、「日立XX」といった表記が残っておりますが、これらの表記は全て「株式会社ルネサス テクノロジ」に変更されておりますのでご理解の程お願い致します。尚、会社商標・ロゴ・コーポレートステートメント以外の内容については一切変更しておりませんので資料としての内容更新ではありません。

ルネサステクノロジ ホームページ (<http://www.renesas.com>)

2003年4月1日
株式会社ルネサス テクノロジ
カスタマサポート部

ご注意

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、その他お気づ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。

H8/3802シリーズ用 E6000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

HS3800EPI60HJ

ご注意

- 1 本書に記載の製品及び技術のうち「外国為替及び外国貿易法」に基づき安全保障貿易管理関連貨物・技術に該当するものを輸出する場合、または国外に持ち出す場合は日本国政府の許可が必要です。
- 2 本書に記載された情報の使用に際して、弊社もしくは第三者の特許権、著作権、商標権、その他の知的所有権等の権利に対する保証または実施権の許諾を行うものではありません。また本書に記載された情報を使用した事により第三者の知的所有権等の権利に関わる問題が生じた場合、弊社はその責を負いませんので予めご了承ください。
- 3 製品及び製品仕様は予告無く変更する場合がありますので、最終的な設計、ご購入、ご使用に際しましては、事前に最新の製品規格または仕様書をお求めになりご確認ください。
- 4 弊社は品質・信頼性の向上に努めておりますが、宇宙、航空、原子力、燃焼制御、運輸、交通、各種安全装置、ライフサポート関連の医療機器等のように、特別な品質・信頼性が要求され、その故障や誤動作が直接人命を脅かしたり、人体に危害を及ぼす恐れのある用途にご使用をお考えのお客様は、事前に弊社営業担当迄ご相談をお願い致します。
- 5 設計に際しては、特に最大定格、動作電源電圧範囲、放熱特性、実装条件及びその他諸条件につきましては、弊社保証範囲内でご使用いただきますようお願い致します。
保証値を越えてご使用された場合の故障及び事故につきましては、弊社はその責を負いません。また保証値内のご使用であっても半導体製品について通常予測される故障発生率、故障モードをご考慮の上、弊社製品の動作が原因でご使用機器が人身事故、火災事故、その他の拡大損害を生じないようにフェールセーフ等のシステム上の対策を講じて頂きますようお願い致します。
- 6 本製品は耐放射線設計をしておりません。
- 7 本書の一部または全部を弊社の文書による承認なしに転載または複製することを堅くお断り致します。
- 8 本書をはじめ弊社半導体についてのお問い合わせ、ご相談は弊社営業担当迄お願い致します。

重要事項

- 当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。

エミュレータとは：

ここでいうエミュレータとは、株式会社日立製作所（以下、「日立」という。）が製作した次の製品を指します。

（１）E6000 エミュレータ本体、（２）ユーザシステムインタフェースケーブル、（３）PC インタフェースボード（４）オプションボード

お客様のユーザシステム及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：

当エミュレータは、日立マイクロコンピュータ H8/3802 シリーズ（以下、MCU と略します）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、当エミュレータを正しく使用してください。この目的以外に当エミュレータを使用することを堅くお断りします。

使用制限：

当エミュレータは、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

このような目的で当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：

日立は、当エミュレータのデザイン、機能および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。

エミュレータを使う人は：

当エミュレータは、ユーザズマニュアルをよく読み、理解した人のみが使用してください。

特に、当エミュレータを初めて使用する人は、当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おすすめします。

保証の範囲：

日立は、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）日立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です。

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、日立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日立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：

このユーザズマニュアルおよび当エミュレータは著作権で保護されており、すべての権利は日立に帰属しています。このユーザ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、日立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：

このユーザズマニュアルの一部の図は、実物と異っていることがあります。

予測できる危険の限界：

日立は、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、このユーザズマニュアルと当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、当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

- 当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。起こり得る危害又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

危険は、回避しないと、死亡又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

警告は、回避しないと、死亡又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

注意は、回避しないと、軽傷又は中程度の傷害を招くことがある潜在的に危険な状況を示します。



注意

安全警告記号の付かない注意は、回避しないと、財物損傷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。

▲ 警告

1. 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、日立または日立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、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を破壊する可能性があります。
 3. エミュレータまたはユーザシステムのパワーオン時、エミュレータとユーザシステムインタフェースケーブルおよびユーザシステムインタフェースケーブルとユーザシステム上の IC ソケット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を破壊する可能性があります。
 4. ユーザシステムインタフェースケーブル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。
-

まえがき

本書について

本エミュレータ (HS3800EPI60H) は、以下のデバイス (MCU) をサポートします。なお、本文中はデバイス名は表記せず、MCU のみ表記します。

サポートデバイス	MCU
H8/3800、3801、3802	H8/3802 シリーズ

本書は、H8/3802 シリーズマイクロコンピュータ用の E6000 エミュレータのセットアップと使用方法を説明します。本書はデバッグプラットフォームのマニュアルです。

「1 はじめに」では、E6000 エミュレータの主なエミュレーション機能の概要と、E6000 エミュレータの制御ソフトウェアである日立デバッグインタフェース (以降、HDI と呼びます) の機能を簡単に紹介します。

「3 ハードウェア」は、E6000 エミュレータとユーザシステムの接続方法、およびハードウェア詳細について記載します。

「4 チュートリアル」は、簡単な C プログラムのロードとデバッグの方法を示しながら、E6000 エミュレータの主な特長を紹介します。チュートリアルプログラムはディスクで提供されます。したがって、チュートリアルプログラムを実行することによって、システムの動作を直接理解できます。

「5 H8/3802 シリーズ E6000 エミュレータ HDI の機能」は、本 E6000 エミュレータ専用の HDI の特長を記載します。

想定

本書は、読者に MS-DOS[®] および Windows[®] プログラムの実行および使用の手順に関する知識があるものと想定して話を進めます。

関連マニュアル

- 日立デバッグインタフェースユーザズマニュアル
- ユーザシステムインタフェースケーブル取扱い説明書
- PC インタフェースボード取扱い説明書
 - ISAバスインタフェースボード (HS6000EII01HJ)
 - PCIバスインタフェースボード (HS6000EIC01HJ, HS6000EIC02HJ)
 - PCMCIAインタフェースカード (HS6000EIP01HJ)
 - E6000エミュレータ、E8000エミュレータ用LANアダプタ (HS6000ELN01HJ)

Microsoft[®]、MS-DOS[®]、および Windows[®] は米国マイクロソフトコーポレーションの米国及びその他の国における登録商標です。

IBM PC は米国 IBM 社の商標です。

本マニュアルは動作環境を IBM PC 上の英語版 Microsoft Windows[®] 98 として記述しています。

目次

第1章 はじめに

1.1	デバッグの特長	1
1.1.1	ブレイクポイント	1
1.1.2	トレース	1
1.1.3	実行時間測定	1
1.2	イベント検出システム (CES: Complex Event System)	2
1.2.1	イベントチャンネル	2
1.2.2	範囲チャンネル	2
1.2.3	ブレイク	2
1.2.4	イベント間実行時間測定	2
1.3	ハードウェアの特長	3
1.3.1	メモリ	3
1.3.2	動作電圧および動作周波数	3
1.3.3	エミュレーションクロック	3
1.3.4	外部プローブ	3
1.3.5	使用環境条件	4
1.3.6	外形寸法と質量	4

第2章 セットアップ

2.1	パッケージ内容	5
2.2	PC インタフェースボードのセットアップ	5
2.2.1	Windows® 98 での PC インタフェースボードのセットアップ	5
2.3	Windows NT® 4.0 での PC インタフェースボードの設定	8
2.4	HDI のインストール	9
2.4.1	インストールの詳細	14
2.5	システムのチェック	15
2.6	さてつぎは?	16
2.7	HDI のアンインストール	16
2.8	トラブルシューティング	19
2.8.1	接続不良	19
2.8.2	通信不良	20

第3章 ハードウェア

3.1	ユーザシステムへの接続	21
3.1.1	ユーザシステムインタフェースケーブル先端部とユーザシステムの接続例	21
3.1.2	ユーザシステムインタフェースケーブル本体部と E6000 エミュレータの接続	22
3.1.3	ユーザシステムインタフェースケーブル本体部と先端部の接続	22

3.2	電源供給	22
3.2.1	AC 電源アダプタ	22
3.2.2	極性	23
3.2.3	電源モニタ回路	23
3.3	ハードウェアインタフェース	23
3.3.1	信号保護	23
3.3.2	ユーザシステムインタフェース回路	23
3.3.3	クロック発振器	24
3.3.4	外部プローブ/トリガ出力	25
3.3.5	電源フォロウ回路	25
3.4	MCU と E6000 エミュレータの相違点	26
3.4.1	A/D コンバータ	26
3.4.2	未使用領域のアクセス	26
3.4.3	Go Reset コマンドによるプログラム実行	27
第 4 章 チュートリアル		
4.1	はじめに	29
4.1.1	概要	29
4.2	チュートリアルプログラムの動作	29
4.3	HDI の実行	32
4.3.1	ターゲットプラットフォームの選択	32
4.3.2	メニュー	33
4.4	E6000 エミュレータのセットアップ	34
4.4.1	プラットフォームの構成	34
4.4.2	メモリマッピング	35
4.5	チュートリアルプログラムのダウンロード	37
4.5.1	オブジェクトファイルのダウンロード	37
4.5.2	プログラムリストの表示	38
4.6	ブレークポイントの使い方	39
4.6.1	プログラムブレークポイントの設定	39
4.6.2	プログラムの実行	39
4.6.3	レジスタ内容の参照	40
4.6.4	ブレークポイントの確認	41
4.7	メモリと変数の表示	42
4.7.1	メモリを表示する	42
4.7.2	変数を表示する	42
4.8	プログラムのステップ実行	44
4.8.1	シングルステップ	44
4.8.2	関数全体のステップ実行	46
4.8.3	ローカル変数の表示	47
4.9	イベント検出システムの使用方法	49
4.9.1	イベント検出システムによる ハードウェアブレークポイントの設定	49

4.10	トレースバッファの使い方	50
4.10.1	トレースバッファの表示	50
4.10.2	トレースフィルタの設定	51
4.11	セッションの保存	53
4.11.1	さてつぎは?	53

第5章 H8/3802 シリーズ用 E6000 エミュレータ HDI の機能

5.1	コンフィグレーションダイアログボックス	57
5.2	ブレイクポイント	58
5.2.1	プログラムブレイクポイントを設定する	58
5.3	イベント検出システム	59
5.3.1	[General]	60
5.3.2	[Bus / Area]	60
5.3.3	[Signals]	61
5.3.4	[Action]	62
5.3.5	イベントシーケンス	62
5.3.6	イベントの前提条件	63
5.3.7	イベントをリセットする	64
5.4	メモリマッピングダイアログボックス	65
5.5	トレースウインドウ	67
5.5.1	[Filter]	67
5.5.2	[Find]	67
5.5.3	[Cycle]	67
5.5.4	[Pattern]	68
5.5.5	[General]	68
5.5.6	[Bus / Area]	69
5.5.7	[Signals]	69
5.6	トレース制御	70
5.6.1	[General]	70
5.6.2	[Stop]	71
5.6.3	[Delayed stop]	71
5.7	コマンドライン	72

第6章 コマンドライン機能

6.1	BREAKPOINT / EVENT	75
6.1.1	プログラムブレイクポイント	75
6.1.2	アクセスブレイクポイント	75
6.1.3	範囲ブレイクポイント	75
6.1.4	オプション	75

6.2	BREAKPOINT_CLEAR / EVENT_CLEAR.....	77
6.3	BREAKPOINT_DISPLAY / EVENT_DISPLAY.....	77
6.4	BREAKPOINT_ENABLE / EVENT_ENABLE.....	77
6.5	BREAKPOINT_SEQUENCE / EVENT_SEQUENCE.....	78
6.6	CLOCK	78
6.7	DEVICE_TYPE.....	79
6.8	MAP_SET	79
6.9	MODE	80
6.10	TEST_EMULATOR	80
6.11	TIMER.....	80
6.12	TRACE_ACQUISITION.....	81
6.13	TRACE_COMPARE.....	81
6.14	TRACE_SAVE.....	81
6.15	TRACE_SEARCH	82
6.16	USER_SIGNALS	82
6.17	REFRESH.....	82
第7章 故障解析		
7.1	テストプログラムを実行するための システムセットアップ	83
7.2	テストプログラムによる故障解析.....	83
7.3	エラー発生時の処理	87

図表目次

図目次

図 2-1	Computer Properties ダイアログ(設定前)	6
図 2-2	Edit Resource Setting ダイアログ	7
図 2-3	Computer Properties ダイアログ(設定後)	7
図 2-4	HDI インストールディスクの選択画面	10
図 2-5	HDI インストーラ起動画面	10
図 2-6	Read Me ファイル画面	10
図 2-7	HDI インストールディスクの選択画面	11
図 2-8	バックアップ指定画面	11
図 2-9	バックアップディレクトリの指定画面	12
図 2-10	HDI インストール中の画面 (1)	12
図 2-11	ディスク入れ替え要求画面 (1)	12
図 2-12	HDI インストール中の画面 (2)	13
図 2-13	ディスク入れ替え要求画面 (2)	13
図 2-14	ホストインタフェース選択要求画面	13
図 2-15	アイコンのプログラムグループの指定画面	14
図 2-16	HDI のプログラムグループ	14
図 2-17	HDI アイコン	15
図 2-18	HDI 起動中のステータスバー表示	15
図 2-19	HDI の起動画面	16
図 2-20	Start メニュー (アンインストーラ)	17
図 2-21	Select Uninstall Method ダイアログボックス	17
図 2-22	Perform Rollback ダイアログボックス	18
図 2-23	Perform Uninstall ダイアログボックス	19
図 2-24	エラーメッセージ (1)	19
図 2-25	エラーメッセージ (2)	20
図 3-1	E6000 コネクタの位置	21
図 3-2	ユーザシステムインタフェースケーブルの接続	21
図 3-3	ネジの締め付け順序	22
図 3-4	ユーザシステムインタフェースケーブル外観図	22
図 3-5	電源プラグ	23
図 3-6	クロック発振回路	25

図 3-7	外部プローブコネクタ	25
図 3-8	外部プローブインタフェース回路	25
図 3-9	ユーザシステムと E6000 との Vcc の関係	26
図 4-1	HDI 起動メニュー	32
図 4-2	プラットフォームの選択	33
図 4-3	HDI ウィンドウ画面	33
図 4-4	Configuration ダイアログボックス	34
図 4-5	Memory Mapping ダイアログボックス	35
図 4-6	Edit Memory Mapping ダイアログボックス	36
図 4-7	System Status ウィンドウ (Memory シート)	36
図 4-8	Open ダイアログボックス (オブジェクトファイルの選択)	37
図 4-9	HDI ダイアログボックス	37
図 4-10	Open ダイアログボックス (ソースファイルの選択)	38
図 4-11	ソースプログラム画面	38
図 4-12	ブレークポイントの設定	39
図 4-13	ステートメントの強調表示	40
図 4-14	System Status ウィンドウ (Platform シート)	40
図 4-15	Register Window 画面	41
図 4-16	レジスタ値の編集	41
図 4-17	Breakpoints ウィンドウ	41
図 4-18	Memory Window の設定	42
図 4-19	Memory ウィンドウ (Byte)	42
図 4-20	Watch 画面	43
図 4-21	Watch 画面 (シンボル拡張)	43
図 4-22	Add Watch ダイアログボックス	43
図 4-23	Watch ウィンドウ (変数の追加)	44
図 4-24	Step In 実行後のプログラムウインドウの表示 (1)	45
図 4-25	Step In 実行後のプログラムウインドウの表示 (2)	45
図 4-26	Step Out 実行後のプログラムウインドウ画面	46
図 4-27	Step In 実行後のプログラムウインドウの画面 (3)	46
図 4-28	Step Over 実行後のプログラムウインドウの画面	47
図 4-29	Step In 実行後のプログラムウインドウ画面 (4)	47
図 4-30	ローカル変数の表示	48
図 4-31	ローカル変数の表示 (配列要素の表示)	48
図 4-32	ブレークポイントの追加	49
図 4-33	Breakpoint Window 画面 (追加後)	50
図 4-34	ブレークポイントによるプログラムの停止	50
図 4-35	Trace Window 画面	51

図 4-36	Trace Filter ダイアログボックス	51
図 4-37	Bus / Area の設定	52
図 4-38	Trace Window 画面 (トレースフィルタ指定)	53
図 5-1	Configuration ダイアログボックス	57
図 5-2	Breakpoints 画面	58
図 5-3	ブレークポイントの設定	59
図 5-4	イベントの設定 (General)	60
図 5-5	イベントの設定 (Bus / Area)	61
図 5-6	イベントの設定 (Signals)	61
図 5-7	イベントの設定 (Action)	62
図 5-8	イベントシーケンス画面 (1)	63
図 5-9	イベントシーケンス画面 (2)	64
図 5-10	イベントシーケンス画面 (3)	65
図 5-11	Memory Mapping 画面	65
図 5-12	メモリマッピングの変更	66
図 5-13	Trace Window 画面	67
図 5-14	Trace Filter 画面 (General) (1)	68
図 5-15	Trace Filter 画面 (General) (2)	68
図 5-16	Trace Filter 画面 (Bus / Area)	69
図 5-17	Trace Filter 画面 (Signals)	69
図 5-18	Trace Acquisition 画面 (General)	70
図 5-19	Trace Acquisition 画面 (Stop)	71
図 5-20	Trace Acquisition 画面 (Delayed Stop)	72
図 5-21	Command Line Window 画面	72

表目次

表 1-1	メモリタイプの定義	3
表 1-2	動作電圧および動作周波数	3
表 1-3	エミュレーションクロック一覧	3
表 1-4	E6000 使用環境条件	4
表 1-5	外形寸法および質量	4
表 2-1	PC インタフェースボードのメモリ領域	6
表 3-1	MCU と E6000 エミュレータの相違	26
表 4-1	コンフィグレーションオプションの設定例	35
表 4-2	メモリタイプの定義	35
表 4-3	メモリタイプオプション	35
表 4-4	プログラムステップオプション	44
表 5-1	HDI のメニューとマニュアルの対応表	55

表 5-2	コンフィグレーションダイアログボックス	57
表 5-3	イベントチャンネルとオプションの設定	59
表 5-4	イベントアクション	62
表 5-5	メモリタイプの定義	66
表 5-6	アクセスタイプの定義	66
表 6-1	HDI コマンドライン機能とマニュアルの対応表	73
表 6-1	HDI コマンドライン機能とマニュアルの対応表 (つづき)	74
表 6-2	MCU バスステータス	76
表 6-3	ブレークポイント削除の指定	77
表 6-4	ブレークポイント有効 / 無効	78
表 6-5	クロックパラメータ	79
表 6-6	MCU モードのパラメータ	80
表 6-7	タイマコマンド	80
表 6-8	ユーザ信号コマンド	82

1. はじめに

E6000 エミュレータは、日立 MCU をサポートする高性能リアルタイムインサーキットエミュレータです。本 E6000 エミュレータは H8/3802 シリーズマイクロコントローラ用のプログラムの開発とデバッグができます。

E6000 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で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。

E6000 エミュレータは、Windows ベースのインタフェースプログラムである HDI とともに動作します。HDI は、E6000 エミュレータハードウェアを制御し、豊富なコマンドを提供します。

1.1 デバッグの特長

1.1.1 ブレークポイント

E6000 エミュレータは、強力なハードウェアブレークおよびプログラムブレークを備えているので、ソフトウェアとユーザシステムのデバッグを効率よく実行できます。

ハードウェアブレークポイント

イベント検出システムのイベントチャンネルと範囲チャンネルを使って、最大 12 箇所のブレークポイントが設定できます。ハードウェアブレークポイントに関しては、「1.2 イベント検出システム (CES)」を参照してください。

プログラムブレークポイント (PC ブレークポイント)

最大 256 のプログラムブレークポイントが設定できます。プログラムブレークポイントは、ユーザ命令を BREAK 命令で置き換えることによって設定されます。

1.1.2 トレース

E6000 エミュレータは、強力なリアルタイムトレース機能を備えていますので、MCU の動作を詳細に調べることができます。リアルタイムトレースバッファは、32768 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。バッファはローリングバッファとして構成され、エミュレーションを中断することなく、トレースを中断しトレース内容を表示することができます。

トレースバッファ内の取得データは、デバッグを容易にするためにソースプログラムおよびアセンブリ言語の両方で表示されます。ただし、トレースフィルタリングが行われた場合は、アセンブリ言語だけが表示されます。

トレースバッファは、すべてのバスサイクルあるいは選択されたサイクルだけを記憶するように制御されます。イベント検出システムを使って所望のトレース制御を選択します。詳細は、以下の「1.2 イベント検出システム」を参照してください。

すべてのバスサイクルを記憶しておいて、選択されたサイクルだけを見ることも可能です。これをトレースフィルタリングといいます。

1.1.3 実行時間測定

E6000 エミュレータによって、総実行時間の測定、またはイベント検出システムで指定されたイベント間の実行時間の測定ができます。タイマの分解能は以下のいずれかの値に設定できます。

20ns, 125ns, 250ns, 500ns, 1 μ s, 2 μ s, 4 μ s, 8 μ s, 16 μ s

1. はじめに

測定可能な最大時間は、分解能 20ns で 6 時、分解能 16 μ s で約 200 日間です。

1.2 イベント検出システム (CES: Complex Event System)

実際のデバッグの大部分において、デバッグしようとするプログラムの不具合またはハードウェアの不具合は、限定された状況においてのみ、発生します。たとえば、あるハードウェアエラーは、メモリの特定の領域がアクセスされた時のみ発生します。簡単なプログラムブレークポイントを使ってその問題を調べ上げるのは、非常に困難です。

E6000 エミュレータは、調べたい条件を正確に記述できるシステム (イベント検出システム) を備えています。これによって、MCU 信号の指定された組み合わせのイベントを定義できます。

イベント検出システムは、E6000 エミュレータのトレース、ブレーク、およびイベント間実行時間測定機能を制御します。

1.2.1 イベントチャネル

イベントチャネルによって、指定されたイベントの発生を検出できます。イベントは以下の項目の組み合わせで定義できます。

- ・ アドレスまたはアドレス範囲
- ・ アドレス範囲外
- ・ マスク条件指定付きデータ
- ・ リードまたはライト
- ・ MCUアクセスタイプ (命令プリフェッチ、データフェッチなど)
- ・ MCUアクセス領域 (内蔵ROM、内蔵RAMなど)
- ・ 4つの外部プローブ信号の値
- ・ イベントの発生回数
- ・ イベントの発生後のディレイサイクル数

また、最大 8 イベントがシーケンスで組み合わせられます。それぞれのイベントは、シーケンスにおける前のイベントの発生によって起動、あるいは停止します。たとえば、内蔵 RAM の指定された領域がアクセスされた後で I/O レジスタが書き込まれたときというブレーク条件を設定できます。

1.2.2 範囲チャネル

範囲チャネルは、以下の項目の組み合わせで定義できます。

- ・ アドレスまたはアドレス範囲
- ・ マスク条件指定付きデータ
- ・ リードまたはライト
- ・ MCUアクセスタイプ (命令プリフェッチ、データフェッチなど)
- ・ MCUアクセス領域 (内蔵ROM、内蔵RAMなど)
- ・ 4つの外部プローブ信号の値
- ・ イベントの発生後のディレイサイクル数

イベント検出システムは、E6000 エミュレータの以下の機能を制御するために使われます。

1.2.3 ブレーク

指定されたイベントまたはイベントのシーケンスが発生したときに、プログラム実行を停止します。たとえば、プログラムがあるアドレスからデータ読み出し後、あるアドレスにデータを書き込んだときに実行を停止するように、ブレークを設定できます。また、ブレークは 65535 バスサイクルまで任意に遅らせることができます。

1.2.4 イベント間実行時間測定

2 つのイベントを設定し、最初のイベントの発生と 2 番目のイベントの発生間のプログラムの実

行時間を測定できます。

1.3 ハードウェアの特長

1.3.1 メモリ

E6000 エミュレータは、エミュレーションメモリとして内蔵 ROM/内蔵 RAM 用メモリを標準装備しています。

エミュレーションメモリは、MCU アドレス空間に 1 バイト単位で割り付けできます。各メモリブロックは、[Configure Map...] コマンドを使って、ユーザシステム上のメモリに指定でき、それぞれの場合で、リードライトアクセス、リードオンリアクセス、またはアクセス禁止を指定できます。

エミュレーションメモリの各々のメモリタイプの定義を表 1-1 に示します。

表 1-1 メモリタイプの定義

メモリタイプ	説明
オンチップ	MCU 内蔵メモリ
エミュレータ	エミュレーションボード上のメモリ

メモリアドレスの指定されたブロックの内容は、[Memory...] コマンドを使って表示されます。メモリの内容はいつでも（プログラム実行中であっても）変更でき、その結果は、他の関連するウィンドウにすぐに反映されます。

1.3.2 動作電圧および動作周波数

以下に本 E6000 エミュレータがサポートする MCU の動作電圧および動作周波数仕様について示します。MCU によっては低電圧動作、高周波数動作を保証しないものがありますので注意してください。

表 1-2 動作電圧および動作周波数

No.	MCU 型名	動作電圧(V)	動作周波数範囲 fosc(MHz)
1	H8/3802 シリーズ	1.8-5.5	1.0-4.0
		2.7-5.5	1.0-10.0
		4.5-5.5	1.0-16.0

1.3.3 エミュレーションクロック

システムクロックとサブクロックは表 1-3 に示すような周波数に設定できます。

表 1-3 エミュレーションクロック一覧

エミュレータ	エミュレーションクロック	ターゲット MCU	周波数設定値
HS3800EPI60H	システムクロック	H8/3802	8MHz,2MHz,0.5MHz およびターゲットクロック ÷ 2
	サブクロック	H8/3802	32.768kHz,38.4kHz,307.2kHz, およびターゲットサブクロック

1.3.4 外部プローブ

ユーザシステム上の任意の信号をブレイクもしくはトレースに使うために、E6000 エミュレータには外部プローブが接続できます。外部プローブの信号はローまたはハイレベルに応じて、イベント検出システムの条件として設定できます。

1. はじめに

1.3.5 使用環境条件

表 1-4 E6000 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%RH（結露なし） 非動作時：35～80%RH（結露なし）
3	周囲ガス	腐食性ガスのないこと
4	AC 入力電源	電圧：100～240V AC 周波数：50/60Hz 電流：Max. 0.6A
5	ユーザ Vcc (UVcc)	電圧：1.8～5.5V

1.3.6 外形寸法と質量

表 1-5 外形寸法および質量

項番	項目	仕様
1	外形寸法	219 x 160 x 54 (mm)
2	質量	約 970 (g)

2. セットアップ

本章は、PC インタフェースボードを使用した E6000 エミュレータのセットアップ方法、および HDI と共に使うための E6000 エミュレータの準備方法を述べます。

本章は以下の方法について説明します。

- ・ PCインタフェースボード（別売）のセットアップ
- ・ E6000エミュレータのセットアップ
- ・ HDIソフトウェアのインストールとそれを使ったシステムの正しい動作のチェック

2.1 パッケージ内容

E6000 エミュレータには、以下の構成部品が梱包されています。

- ・ E6000エミュレータ
- ・ AC電源アダプタ 5V 5A（ACケーブル付き）
- ・ HDIインストールディスク（HS3800EPI60SF）
- ・ テストプログラムディスク（HS3800EVI60SF）
- ・ 外部プローブ
- ・ E6000 H8/3802シリーズ用エミュレータユーザーズマニュアル（本マニュアル）
- ・ 日立デバッグインタフェースユーザーズマニュアル

セットアップの前に、上記の構成部品がすべてそろっていることを確認してください。

ホストコンピュータには IBM PC または、IBM PC 互換機が使用できます。Windows® 98 または Windows NT®が必要です。

2.2 PC インタフェースボードのセットアップ

E6000 エミュレータは、PC インタフェースボード（HS6000EII01H）を使って HDI と通信します。はじめに、PC インタフェースボードを PC に差し込む必要があります。

PC インタフェースボードはメモリマップドボードであり、差し込む前に、PC インタフェースボードが使うメモリ領域を確保しなければなりません。これによって、他のプログラムが不用意に PC インタフェースボードを使ってしまうことを防止できます。

PC インタフェースボードに割り当てたメモリ領域が、他のボードに割り当てた領域と重ならないようにします。もしも重なると、PC インタフェースボードと E6000 エミュレータは正しく動作しません。出荷時には、PC インタフェースボードのメモリ領域は H'D0000 から H'D3FFF に割り当ててあります。

2.2.1 Windows® 98 での PC インタフェースボードのセットアップ

- ・ Windows® 98を起動します。
- ・ [My Computer] アイコンをマウス右ボタンでクリックし、ポップアップメニューから [Properties] を選択します。
System Properties ダイアログが表示されます。
- ・ Device Managerパネルの [Computer] アイコンをダブルクリックし、Computer Propertiesダイアログを開きます。
- ・ View Resourcesパネルの [Memory] をクリックし、メモリのリソースを表示します。

2. セットアップ

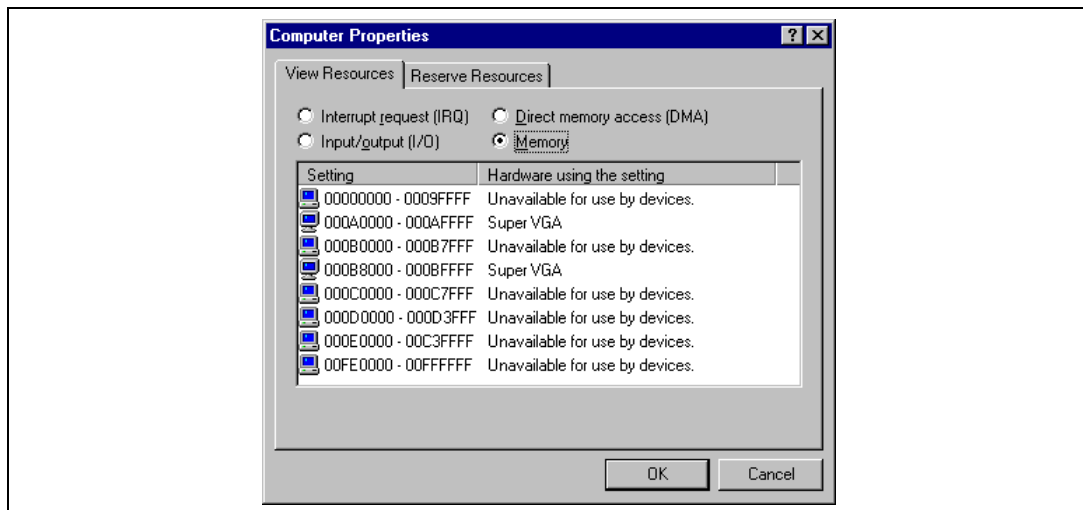


図 2-1 Computer Properties ダイアログ(設定前)

ここにリストされていないメモリ領域が、PC インタフェースボード用に使用できます。下の表は、PC インタフェースボードのリアパネルのスイッチによって指定されるアドレスを示しています。これらのメモリ領域の中で、Computer Properties ダイアログでリストされていないメモリ領域を選択してください。たとえば、H'D8000 から H'DBFFF の領域を選択すると、対応するスイッチ番号は 6 になります。

表 2-1 PC インタフェースボードのメモリ領域

メモリ領域	スイッチ
H'C0000 ~ H'C3FFF	0
H'C4000 ~ H'C7FFF	1
H'C8000 ~ H'CBFFF	2
H'CC000 ~ H'CFFFF	3
H'D0000 ~ H'D3FFF (出荷時の設定)	4
H'D4000 ~ H'D7FFF	5
H'D8000 ~ H'DBFFF	6
H'DC000 ~ H'DFFFF	7
H'E0000 ~ H'E3FFF	8
H'E4000 ~ H'E7FFF	9
H'E8000 ~ H'EBFFF	A
H'EC000 ~ H'EFFFF	B

選択したメモリ領域を Windows[®] 98 が使用しないよう、以下の手順で登録します。

- Reserve Resources パネルの [Memory] をクリックし、[Add] をクリックします。

Edit Resource Setting ダイアログが表示されます。

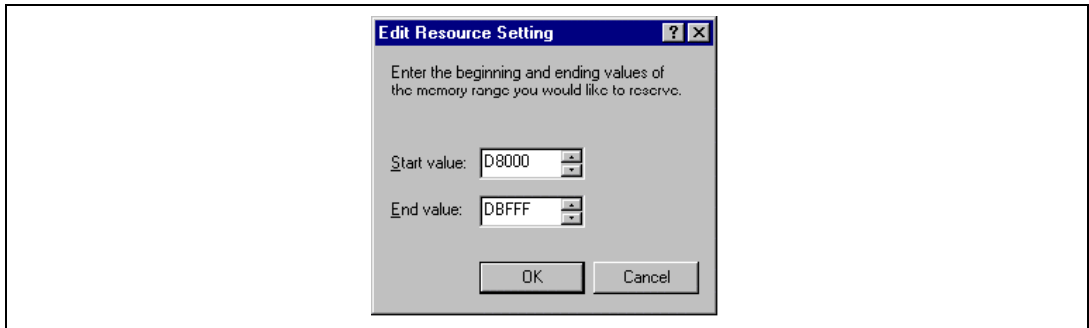


図 2-2 Edit Resource Setting ダイアログ

- ・ 選択したメモリ領域の [Start value] [End value] を入力してください。
- ・ PCをリスタートせずシャットダウンし、電源スイッチを切ってください。
- ・ 小型のマイナスインプラを使って、PCインタフェースボードのリアパネルのスイッチを回し、選択したメモリ領域に対応するスイッチ番号を矢印が差すようにしてください。
- ・ PCのカバーを取り外し、未使用のISAバススロットにPCインタフェースボードを差し込んでください。
- ・ PCのカバーを取り付けてください。
- ・ PCインタフェースボードとE6000エミュレータの“ PC IF ” コネクタの間にPCインタフェースケーブルを接続してください。各プラグはカチッと音がするまでしっかりと差し込んでください。
- ・ PCの電源スイッチを入れてください。
- ・ Computer Propertiesダイアログで選択したメモリ領域が、System Reservedとリストされていることを確認してください。

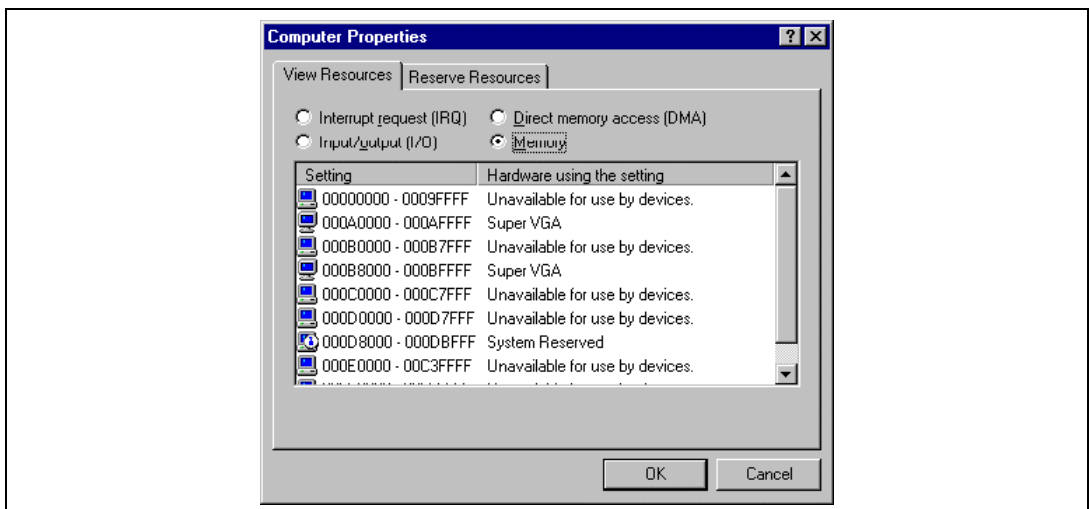


図 2-3 Computer Properties ダイアログ(設定後)

2. セットアップ

2.3 Windows NT[®] 4.0 での PC インタフェースボードの設定

PC インタフェースボードは ISA バススロットを使用しますので、ISA バススロットをサポートしていない PC では使用することはできません。

ISA ボードのインストール方法については、お使いの「PC に付属のマニュアル」を参照してください。ここでは一般的な方法を述べます。

(1) Windows NT[®]実行

Start/Programs/Administrative Tools (Common)/Windows NT Diagnostics を実行してください。

'Resource' タブの'Memory'ボタンをクリックし、使用されている上位メモリ領域を以下に記録してください。

#	Start	End	#	Start	End	#	Start	End
0			4			8		
1			5			9		
2			6			A		
3			7			B		

- Windows NT[®]をシャットダウンしてください。

(2) PC をセットアップモードで起動

セットアップモードについては、お使いの「PC に付属のマニュアル」を参照してください。

- 使用されている上位メモリ領域をチェックしてください。

#	Start	End	#	Start	End	#	Start	End
0			4			8		
1			5			9		
2			6			A		
3			7			B		

(これは、上で得た Windows NT[®]の値と同じはずです)

PC インタフェースボードの設定を登録してください。使用するメモリ領域は、他のデバイスにいつも使用されず、かつ PC インタフェースボードのスイッチ位置の一つと同じにする必要があります。

スイッチ位置：

#	Start	End	#	Start	End	#	Start	End
0	C0000	C3FFF	4	D0000	D3FFF	8	E0000	E3FFF
1	C4000	C7FFF	5	D4000	D7FFF	9	E4000	E7FFF
2	C8000	CBFFF	6	D8000	DBFFF	A	E8000	EBFFF
3	CC000	FFFFF	7	DC000	DFFFF	B	EC000	FFFFF

ご使用の PC に Intel P&P BIOS ディスクが付属の場合は以下の手順で設定します。

- ・ Intel P&P BIOS ディスクで PC を起動します。
'View/System Resources'で使用している上位メモリ領域をチェックします。
- ・ 'Configure/Add Card/Others...'で'Unlisted Card'を追加します。
- ・ .CFG ファイルがありませんので、次のダイアログボックスでは No と答えます。
- ・ 'Configure Unlisted Card'ダイアログボックスで、'Memory [hex]'リストボックスに移動します。
- ・ 'Add Memory...'ボタンを使用し、'Specify Memory'ダイアログボックスを表示します。
- ・ 他のデバイスにいつも使用されず、かつ PC インタフェースボードのスイッチ位置の一つと同じメモリ領域を入力します。

- ・ ファイルをセーブします。
- ・ 終了します。
- ・ PC をリスタートせずシャットダウンし、電源スイッチを切ってください。
- ・ 小型のマイナスインスクリバを使って、PC インタフェースボードのリアパネルのスイッチを回し、選択したメモリ領域に対応するスイッチ番号を矢印が差すようにしてください。
- ・ PC のカバーを取り外し、未使用の ISA バススロットに PC インタフェースボードを差し込んでください。
- ・ PC のカバーを取り付けてください。
- ・ PC インタフェースボードと E6000 エミュレータの “ PC IF ” コネクタの間に PC インタフェースケーブルを接続してください。各プラグはカチッと音がするまでしっかりと差し込んでください。
- ・ PC の電源スイッチを入れてください。

(3) アドミニストレータモードで Windows NT[®]を実行

- ・ HDI のインストールを行ってください。
- ・ Start/Programs/Hdi/Setup ISA bus board を実行してください。（もしも DOS プロンプトが開けなかった場合は、DOS プロンプトを開き HDI をインストールしたディレクトリに移動して SETUPISA.EXE を実行してください。）

2.4 HDI のインストール

最初に以下のように、インストールディスクを使って HDI を PC 上にインストールしてください。

- ・ Windows を起動してください（まだ動作していない場合）。
- ・ 起動中の他のアプリケーションをすべて閉じてください。
- ・ HDI インストールディスク (#1/3) を PC のフロッピーディスクドライブに挿入してください。
- ・ [Start] メニューから [Run] を選択してください。
- ・ “ A:/setup.exe ” とタイプし、[OK] をクリックしてください。

2. セットアップ

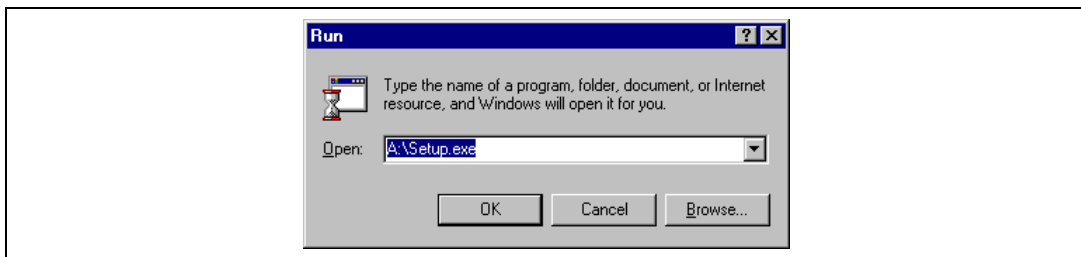


図 2-4 HDI インストールディスクの選択画面

HDI インストーラが動作し、以下の [Welcome!] が表示されます。

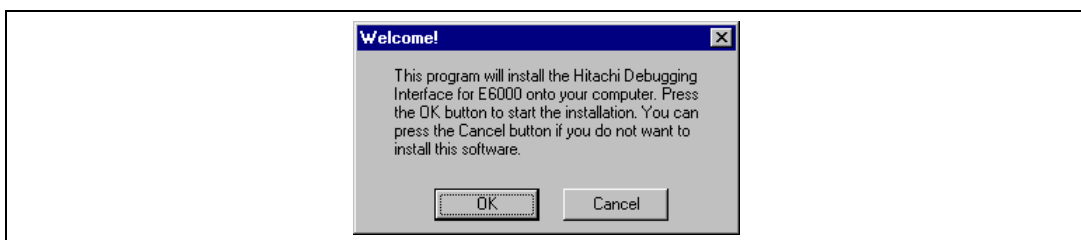


図 2-5 HDI インストーラ起動画面

- ・ [OK] をクリックし、インストールを続行してください。
以下のダイアログボックスが、インストールしている HDI のバージョンの [Read Me] ファイルを表示します。

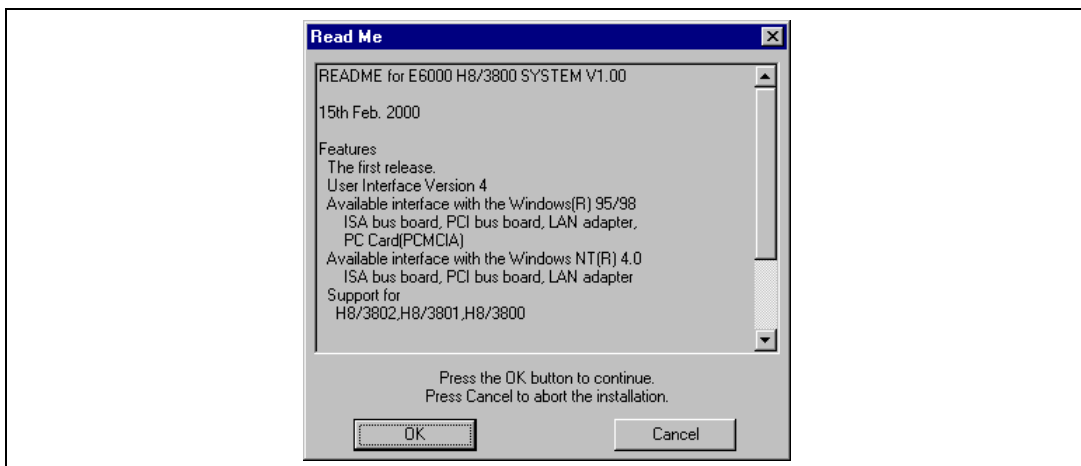


図 2-6 Read Me ファイル画面

- ・ インストールに関する重要な情報として [Read Me] ファイルをチェックし、[OK] をクリックしてください。
以下のダイアログボックスによって、HDI をインストールするディレクトリを選択できます。

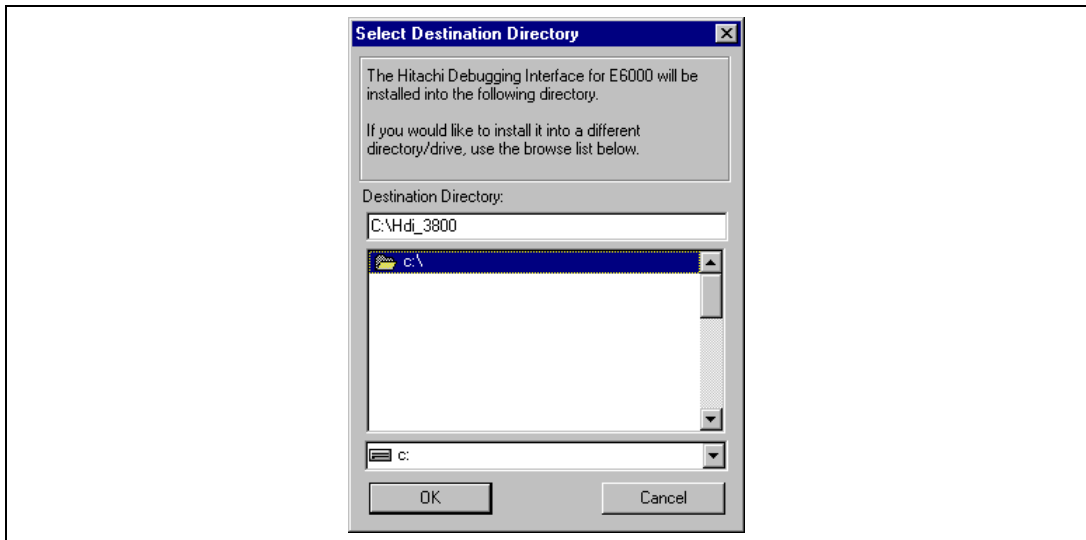


図 2-7 HDI インストールディスクの選択画面

- ・ [OK] をクリックしてデフォルトディレクトリにインストールするか、あるいは別のディレクトリを指定して [OK] をクリックしてください。
以下のダイアログボックスが、インストールによって置き換わるファイルのバックアップを取るかどうかを尋ねます。

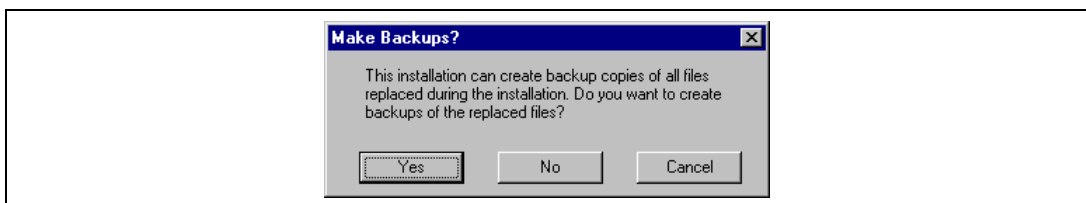


図 2-8 バックアップ指定画面

- ・ インストールによって置き換わるファイルを保存するには [Yes] をクリックしてください。バックアップを取る必要がなければ [No] をクリックしてください。
[Yes] を選択すると、以下のダイアログボックスによって、バックアップディレクトリを指定できます。

2. セットアップ



図 2-9 バックアップディレクトリの指定画面

【注】 バックアップされるファイルがない場合、バックアップディレクトリを指定しても、バックアップディレクトリが作成されない場合があります。

- ・ 使用するディレクトリを指定し、[OK]をクリックしてください。インストーラは、指定されたディレクトリにHDIファイルをコピーします。

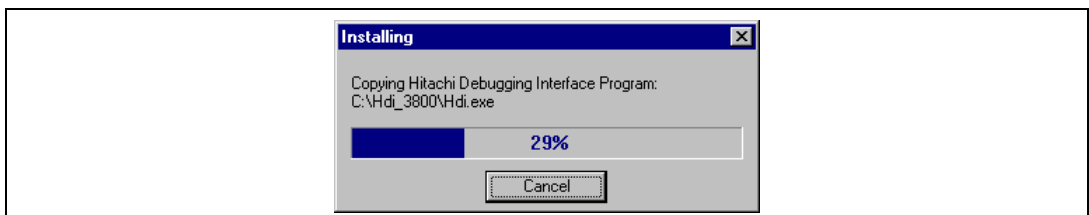


図 2-10HDI インストール中の画面(1)

1 枚目のディスク (#1/3) のインストールが完了すると以下のメッセージが表示されますので、2 枚目のディスク (#2/3) をフロッピーディスクドライブに挿入し [OK] をクリックしてください。



図 2-11 ディスク入れ替え要求画面(1)

インストーラは、指定されたディレクトリに HDI ファイルをコピーします。

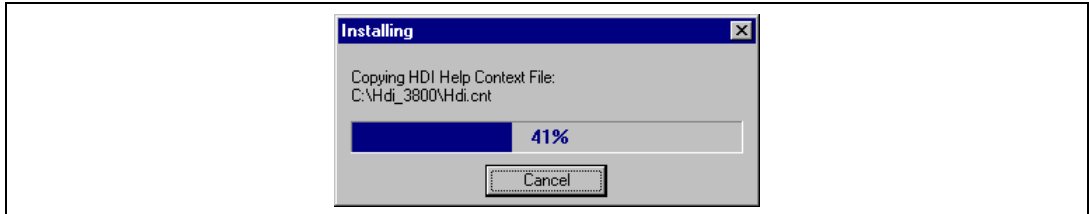


図 2-12 HDI インストール中の画面 (2)

2 枚目のディスク (#2/3) のインストールが完了すると以下のメッセージが表示されますので、3 枚目のディスク (#3/3) をフロッピーディスクドライブに挿入し [OK] をクリックしてください。



図 2-13 ディスク入れ替え要求画面 (2)

以下のダイアログボックスによって、使用するホストインタフェースを選択してください。

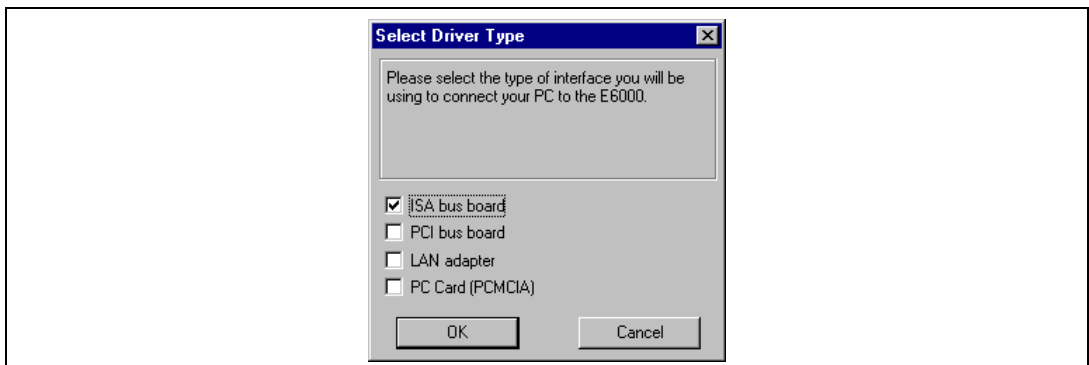


図 2-14 ホストインタフェース選択要求画面

以下のダイアログボックスによって、HDI アイコンのスタートメニューに登録するグループを指定できます。

2. セットアップ

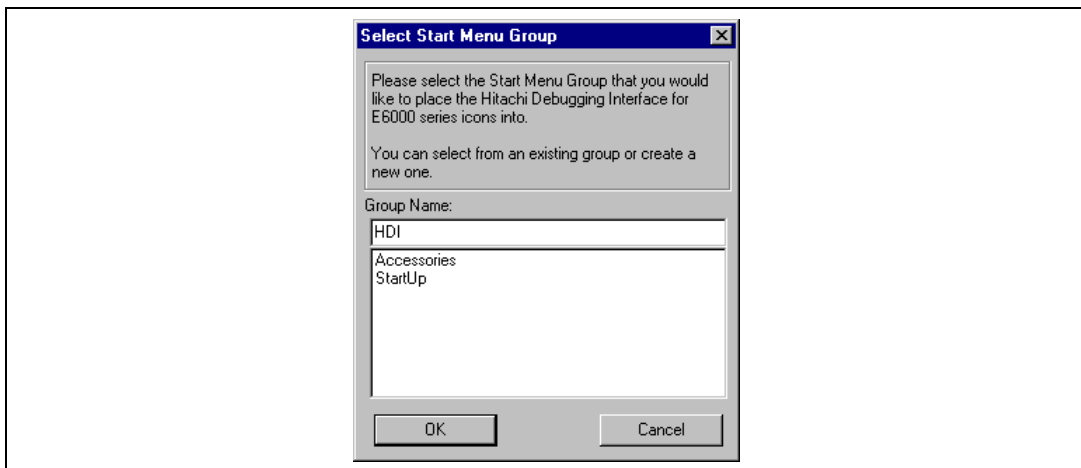


図 2-15 アイコンのプログラムグループの指定画面

- ・ 表示されているグループから選択するか、または新しいグループ名を入力し、[OK] をクリックしてください。

2.4.1 インストールの詳細

インストーラは、指定されたスタートメニューに以下のアイコンを生成します(デフォルト:HDI)。

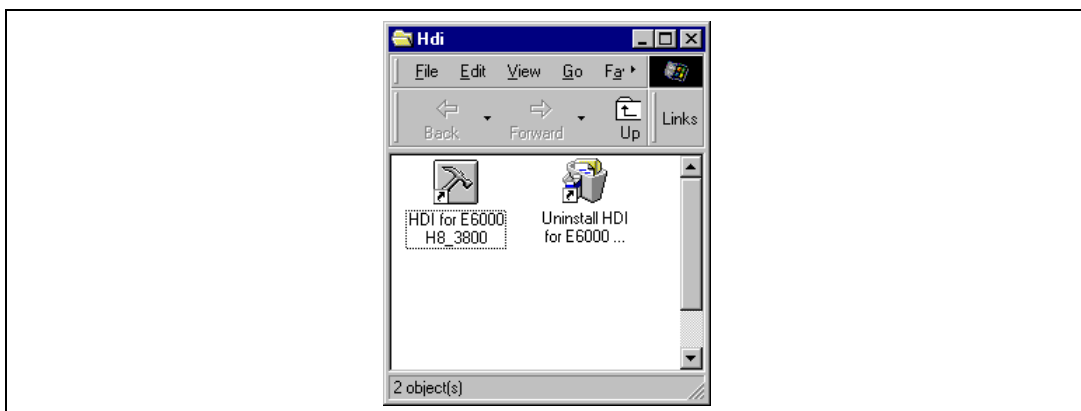


図 2-16 HDI のプログラムグループ

これらのアイコンは以下の機能を備えています。

[HDI for E6000 H8_3800] は HDI プログラムです。

[Uninstall HDI for E6000 H8_3800] は、HDI のアンインストール時に、HDI とその関連ファイルを削除するのに使います。

2.5 システムのチェック

次に、HDI を実行し、E6000 エミュレータが正しく動作することをチェックします。

- ・ E6000エミュレータの電源スイッチを入れ、パワーLEDランプが赤く点灯することを確認してください。
- ・ スタートメニューから [HDI for E6000 H8_3800] を選択してください。

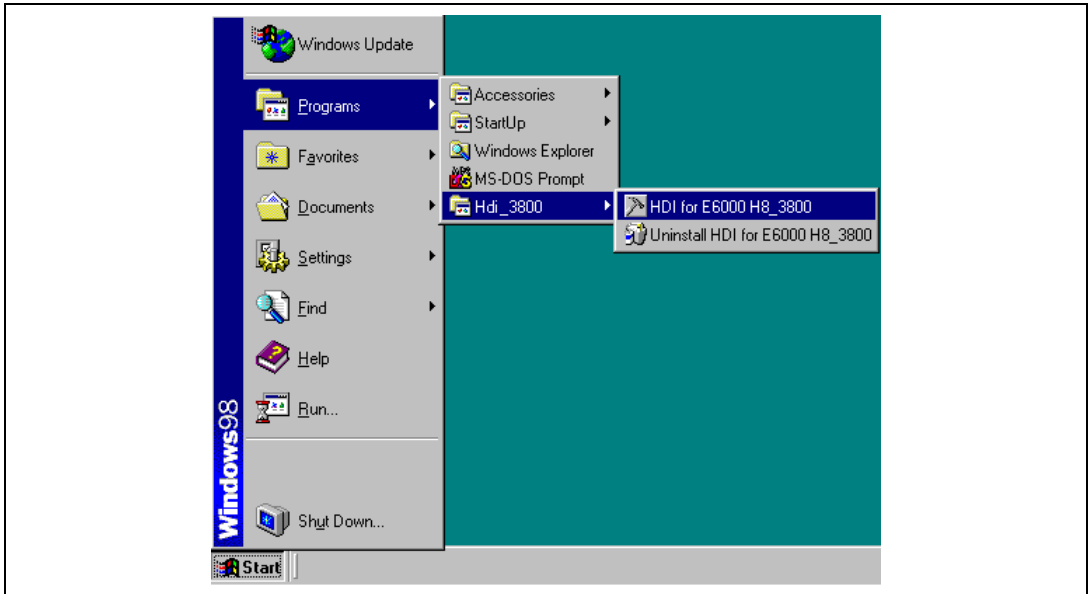


図 2-17 HDI アイコン

HDI のウィンドウが表示され、以下のメッセージがウィンドウの下にステータスバーで現われます。

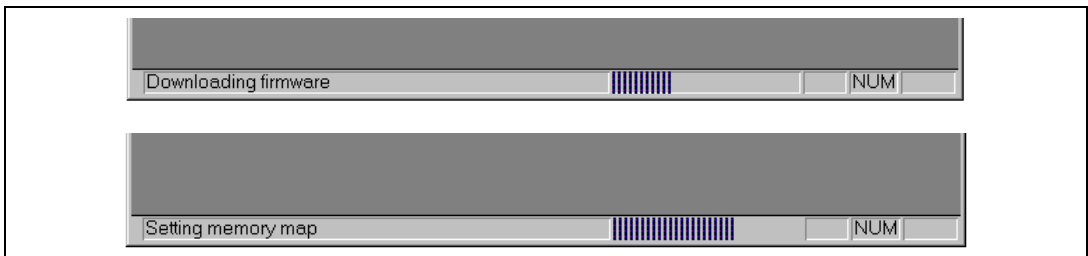


図 2-18 HDI 起動中のステータスバー表示

最後に、全てが正しく設定されたことを示すために、ステータスバーが"Link up"を表示し、HDI のウィンドウが以下のように表示されます。

2. セットアップ

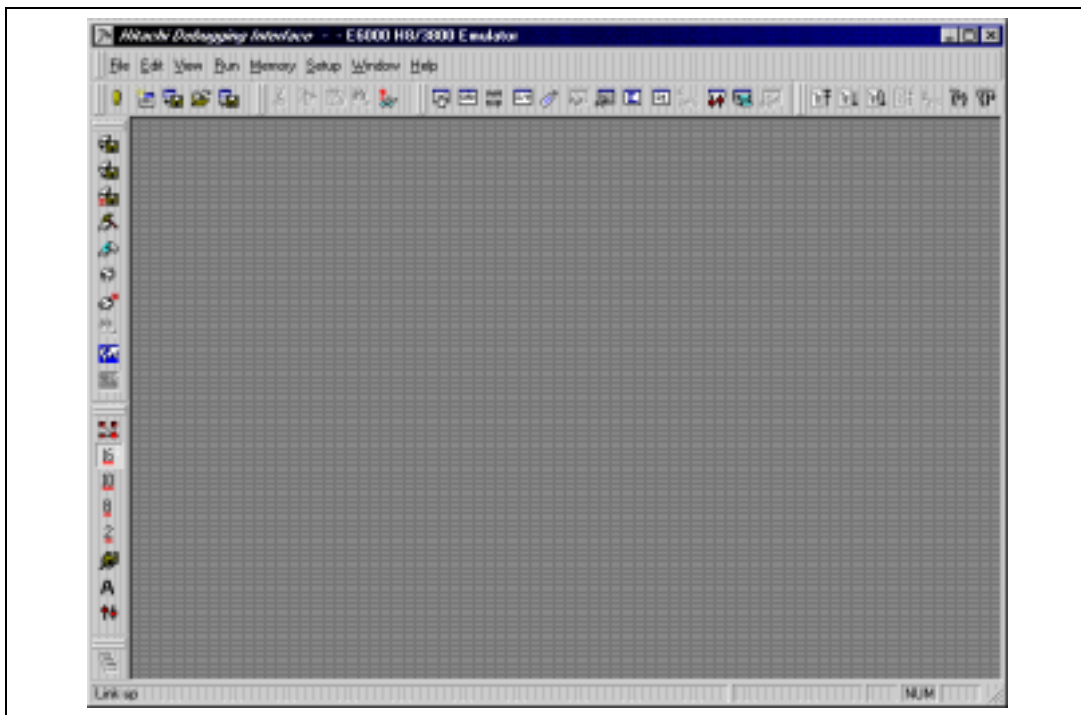


図 2-19 HDI の起動画面

2.6 さてつぎは？

これで E6000 エミュレータは正しくセットアップできました。E6000 エミュレータの主な機能に慣れるために、「4. チュートリアル」の章にしたがって操作してください。そして MCU のプログラムの開発とデバッグを行うために E6000 エミュレータの使い方を覚えてください。

2.7 HDI のアンインストール

Windows® 98 上で HDI のアンインストールをする例を説明します。

- ・ [Start] メニューから [Uninstall HDI for E6000 H8/3800] を選択してください。

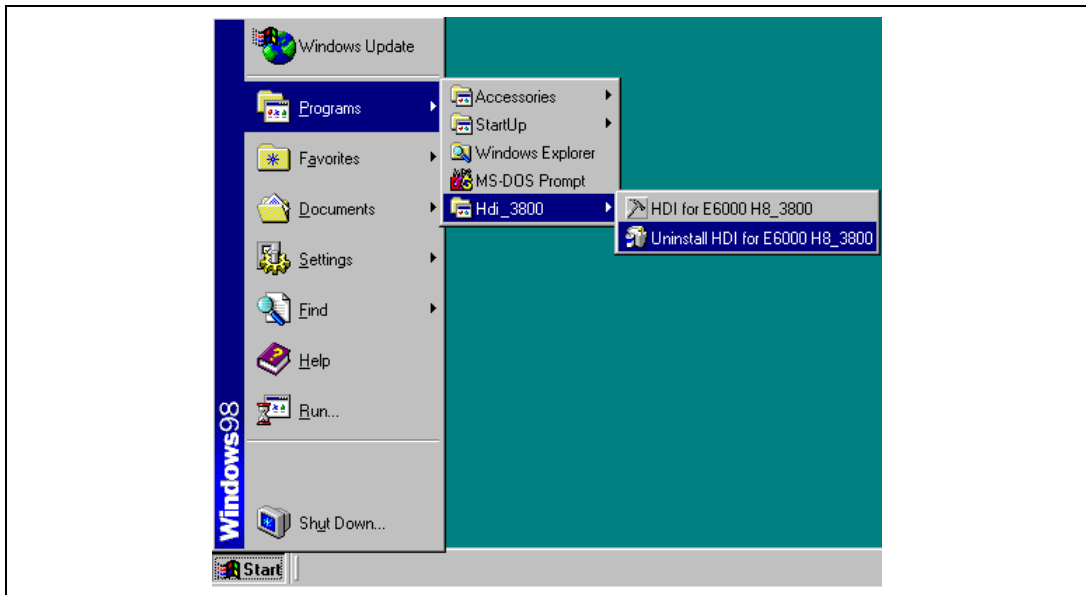


図 2-20 Start メニュー (アンインストーラ)

アンインストーラが起動し、以下のダイアログボックスが表示されます。

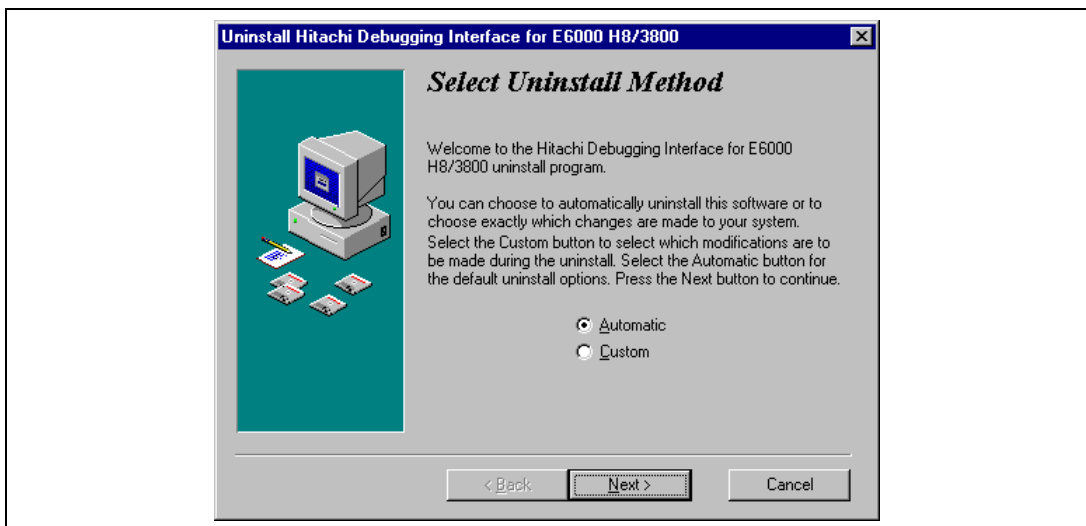


図 2-21 Select Uninstall Method ダイアログボックス

- ・ 自動アンインストールを行う場合は、[Automatic] オプションボタンを選択し、[Next] ボタンをクリックしてください。
- ・ 削除するファイルを選択する場合は、[Custom] オプションボタンを選択し、[Next] ボタンをクリックしてください。
- ・ アンインストールを中断する場合は [Cancel] ボタンをクリックしてください。

2. セットアップ

インストール時に、バックアップを行った場合はロールバックを確認する以下のダイアログボックスが表示されます。

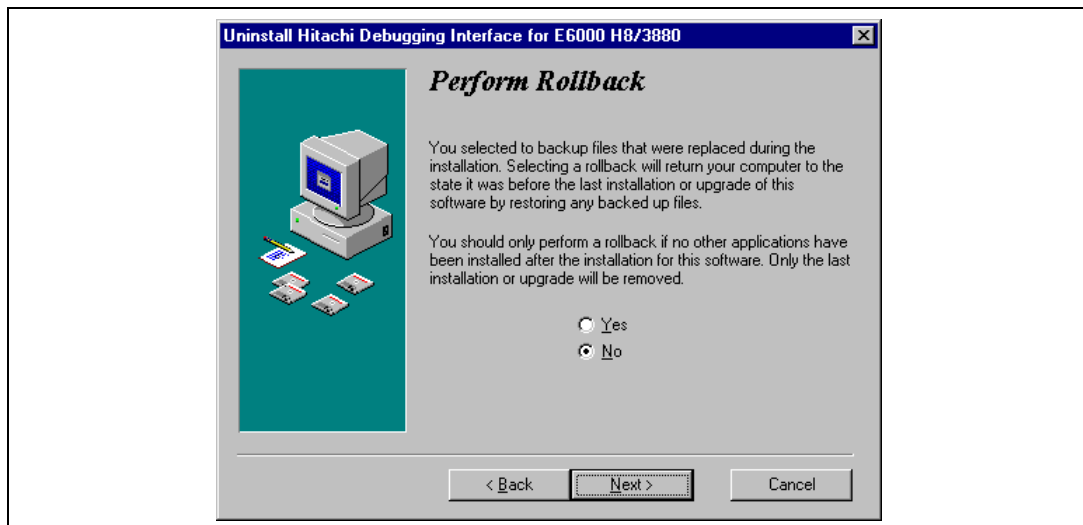


図 2-22 Perform Rollback ダイアログボックス

- ・ ロールバックを行う場合は、[Yes] オプションボタンを選択し、[Next] ボタンをクリックしてください。
- ・ ロールバックを行わない場合は、[No] オプションボタンを選択し、[Next] ボタンをクリックしてください。
- ・ アンインストールを中断する場合は [Cancel] ボタンをクリックしてください。
- ・ Select Uninstall Methodダイアログボックスに戻る場合は、[Back] ボタンをクリックしてください。

【注】 ロールバックにより、インストール時にバックアップされたファイルを元に戻すことができます。

【注】 インストール時にバックアップを行っていない場合、またはバックアップしたファイルが存在しない場合は、Perform Rollback ダイアログボックスは表示されません。

アンインストールの開始を確認するためのダイアログボックスが表示されます。

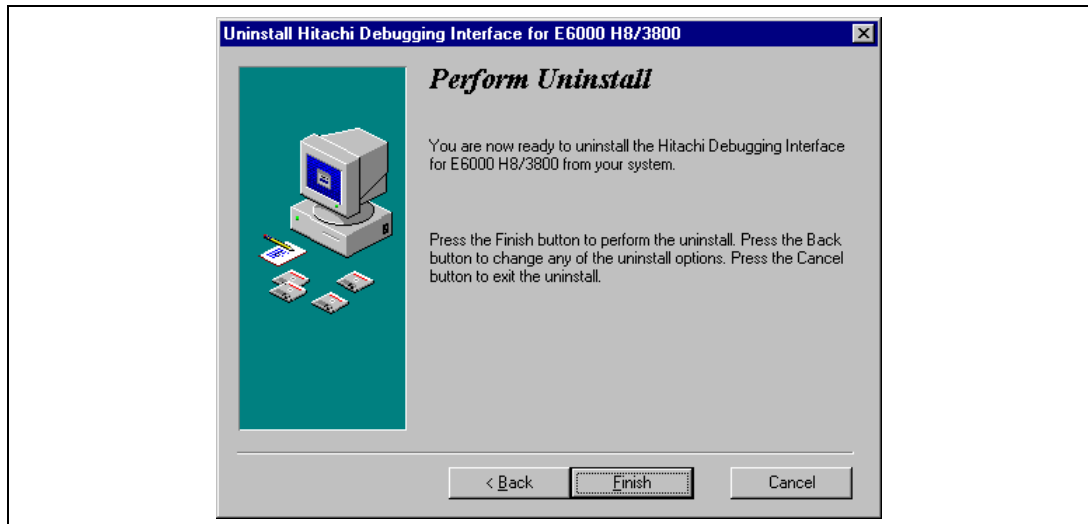


図 2-23 Perform Uninstall ダイアログボックス

- ・ アンインストールを開始する場合は、[Finish] ボタンをクリックしてください。
- ・ アンインストールを中断する場合は、[Cancel] ボタンをクリックしてください。
- ・ Select Uninstall Methodダイアログボックスに戻る場合は、[Back] ボタンをクリックしてください。
アンインストールが完了すると、インストーラで作成したディレクトリやファイルが削除されます。

【注】 HDI インストールディレクトリ内にユーザがファイルやサブディレクトリを作成した場合は、そのディレクトリとファイルは削除されません。

【注】 ロールバックを行わなかった場合は、バックアップディレクトリとバックアップファイルは削除されません。

2.8 トラブルシューティング

2.8.1 接続不良

イニシャライズ中に以下のメッセージボックスが現われた場合、PC インタフェースボードはE6000 エミュレータを認識できていません。

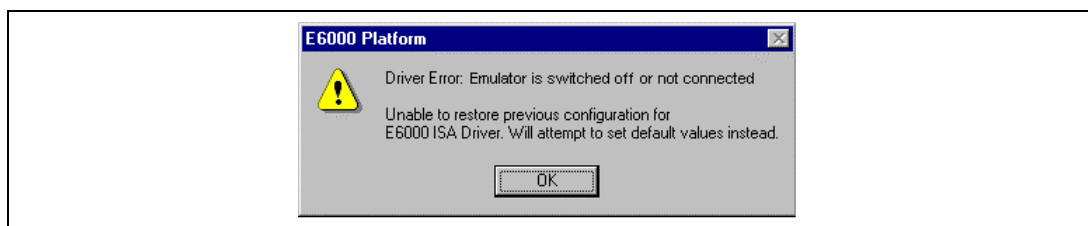


図 2-24 エラーメッセージ (1)

考えられる原因としては以下のようなものがあります。

- ・ 添付のAC電源アダプタがE6000エミュレータに接続されていないか、またはE6000エミュレ

2. セットアップ

ータの電源スイッチが入っていません。E6000エミュレータのパワーLEDを確認してください。

- ・ PCインタフェースケーブルが、PCインタフェースボードとE6000エミュレータの間で正しく接続されていません。

2.8.2 通信不良

以下のメッセージが表示されると、HDI が E6000 エミュレータを正しくセットアップできていません。



図 2-25 エラーメッセージ (2)

考えられる原因としては以下のようなものがあります。

- ・ CONFIG.SYSファイルに確保されたメモリ領域と、PCインタフェースボード上のリヤパネルのスイッチ設定が異なります。
- ・ 選択されたメモリ領域が別のアプリケーションで使われています。
- ・ 「2.2 PCインタフェースボードのセットアップ」「2.3 PCインタフェースボードの設定」に従って設定を見直してください。

3. ハードウェア

本章は、E6000 エミュレータをユーザシステムに接続する方法を説明します。

3.1 ユーザシステムへの接続

E6000 エミュレータをユーザシステムへ接続するには、以下の手順に従ってください。

- ・ユーザシステムインタフェースケーブル先端部をユーザシステムへ接続する。
- ・ユーザシステムインタフェースケーブルのケーブル本体部を E6000 エミュレータへ接続する。
- ・ケーブル本体部を先端部へ接続する。

これら手順の詳細については、「ユーザシステムインタフェースケーブル添付の取扱い説明書」を参照してください。

以下に、E6000 エミュレータのコネクタを示します。

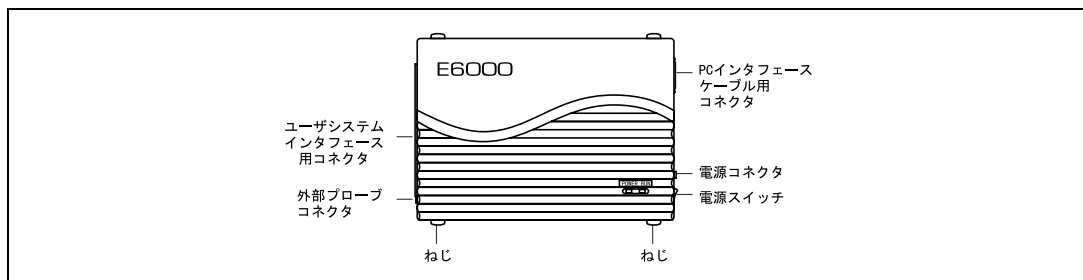


図 3-1 E6000 コネクタの位置

3.1.1 ユーザシステムインタフェースケーブル先端部とユーザシステムの接続例

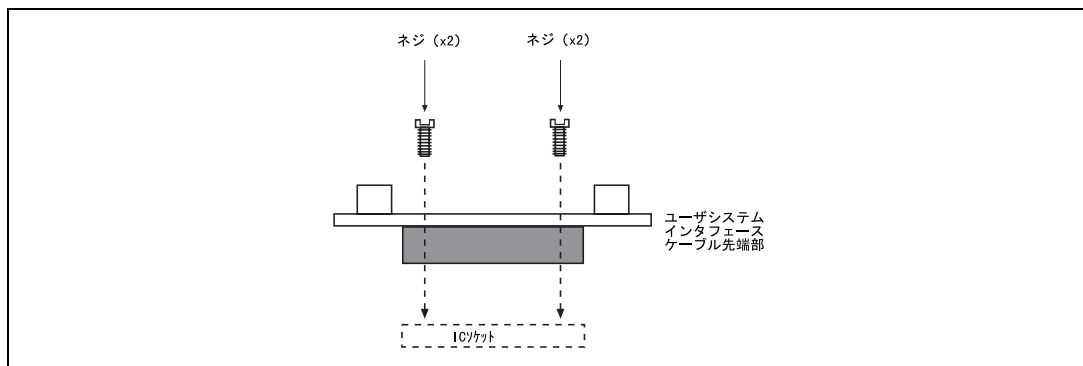


図 3-2 ユーザシステムインタフェースケーブルの接続

- ・接続前に、必ず、E6000エミュレータ、ユーザシステムの電源を切ってください。

3. ハードウェア

- ・ ユーザシステムインタフェースケーブル先端部をユーザシステム上のソケットに挿入してください。

【注】 QFP パッケージによっては、ユーザシステムインタフェースケーブル先端部の向きにかかわらず、ソケットに差し込むことができます。挿入の際には、E6000 エミュレータ側とソケットの 1 ピンの位置を必ず一致させてください。

- ・ 付属のねじを使って、ユーザシステムインタフェースケーブルにユーザシステムインタフェースケーブル先端部とソケットをねじ留めしてください。以下に示す順番で、ねじを徐々に締め付けてください。

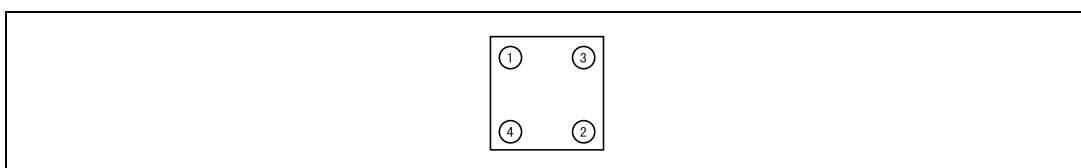


図 3-3 ネジの締め付け順序

【注】 ねじを締め付け過ぎないように注意してください。ユーザシステムの接続不良やユーザシステムインタフェースケーブル先端部が壊れる原因となります。QFP ソケットに半田付け用固定金具が付いている場合は、これを使用して、E6000 エミュレータとユーザシステムの接続を強めることができます。

3.1.2 ユーザシステムインタフェースケーブル本体部と E6000 エミュレータの接続

ユーザシステムインタフェースケーブル本体部ケーブルを E6000 エミュレータに接続してください。ケーブルは、まっすぐに、確実に接続されるまで押し込んでください。

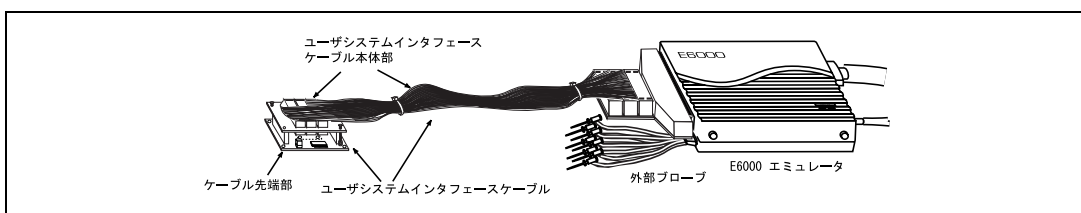


図 3-4 ユーザシステムインタフェースケーブル外観図

3.1.3 ユーザシステムインタフェースケーブル本体部と先端部の接続

ユーザシステムインタフェースケーブル本体部をユーザシステムに接続されている先端部に接続してください。

3.2 電源供給

3.2.1 AC 電源アダプタ

E6000 エミュレータに付属の AC 電源アダプタを常に使用してください。

3.2.2 極性

以下に電源プラグの極性を示します。

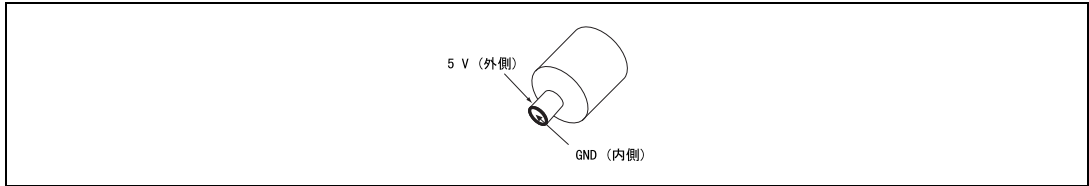


図 3-5 電源プラグ

3.2.3 電源モニタ回路

E6000 エミュレータには、ユーザシステムの電源モニタ回路があり、4.75V 以上の電源が供給されるとパワー LED が赤く点灯します。パワー LED が消えている場合は、E6000 エミュレータの電源レベルをチェックしてください。電源電圧が 4.75V 未満の場合、E6000 エミュレータに必要な電源が供給されません。

【注】 必ず E6000 エミュレータに付属の AC 電源アダプタを使用してください。

3.3 ハードウェアインタフェース

E6000 エミュレータのユーザシステムインタフェース信号は、バッファなしに直接エミュレータ上のエバチップに接続されています。

3.3.1 信号保護

ユーザシステムインタフェース信号は、ダイオードによって、過大/過小電圧から保護されています。ただし、AVcc とアナログポートには、この保護回路はありません。

アナログポート以外のポートには、プルアップ抵抗が接続されています。

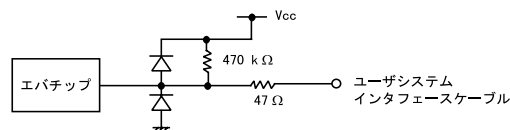
なお、E6000 はユーザシステムインタフェースケーブル先端部の信号を監視することにより、ユーザシステムが接続されているかどうかを判断しています。

3.3.2 ユーザシステムインタフェース回路

E6000 エミュレータのユーザインタフェースには、プルアップ抵抗が入っており信号の遅れが生じます。また、プルアップ抵抗により信号がハイインピーダンス状態でもハイレベルになります。このことを考慮してユーザシステムのハードウェアを調整してください。また、ユーザシステムインタフェースケーブルによる信号の遅れは約 3ns です。

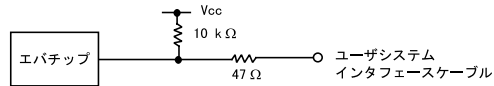
以下にユーザインタフェース信号回路を示します。

(1) 以下に記述のない信号



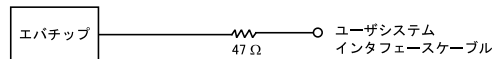
3. ハードウェア

(2) OSC1, X1

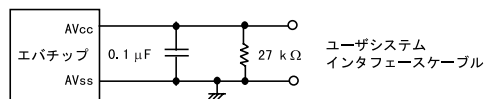


(3) P50/ $\overline{WKP0}$ /SEG1 ~ P57/ $\overline{WKP7}$ /SEG8

P60/SEG9 ~ P67/SEG16
P70/SEG17 ~ P77/SEG24
P80/SEG25 ~ P87/SEG32/CL1
PC0/COMP0 ~ PC3/COMP3
PB0/AN0 ~ PB7/AN7

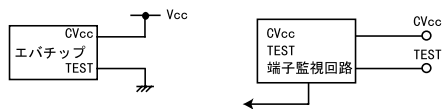


(4) AVcc, AVss

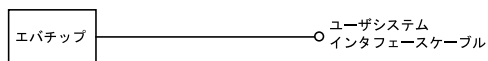


(5) CVcc, TEST

CVcc が GND に接続されている場合、または TEST が Vcc レベルに接続されている場合は、HDI 起動時にメッセージが出力されます。



(6) V0, V1, V2, V3



3.3.3 クロック発振器

以下の図は、ユーザーシステムインタフェースケーブル先端部に構成された発振回路を示します。この回路は、水晶発振子を 1 ~ 16MHz の範囲で発振するように設計されています。詳細については「各ユーザーシステムインタフェースケーブルマニュアル」を参照してください。

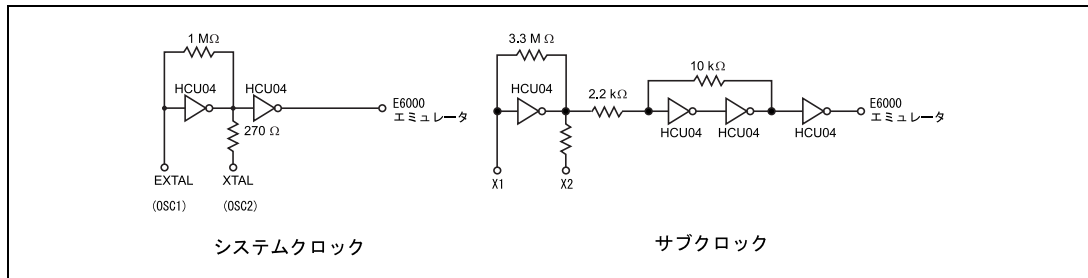


図 3-6 クロック発振回路

3.3.4 外部プローブ/トリガ出力

E6000 エミュレータ筐体側面にある EXT のマークが記された 8 ピンコネクタ (ユーザインタフェースコネクタの横) に E6000 エミュレータ付属の外部プローブを接続してください。

外部プローブは入力 4 本とトリガ出力 2 本を備えています。

以下にこのコネクタのピン配置を示します。

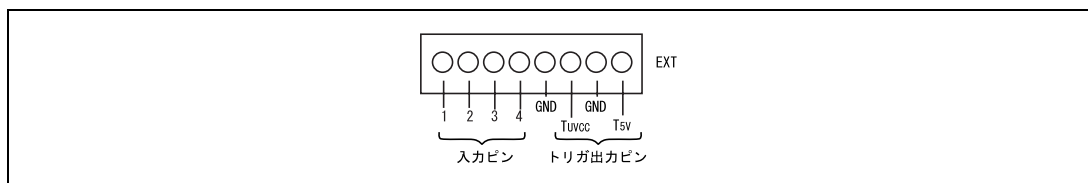


図 3-7 外部プローブコネクタ

以下に外部プローブのインタフェース回路を示します。

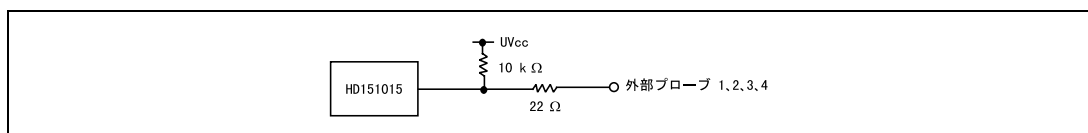


図 3-8 外部プローブインタフェース回路

トリガ出力はイベントチャンネル 8 によって出力されるアクティブロー信号です。トリガ出力は T5V (プローブ色: 白色、2.5V ~ 5.0V の範囲でユーザシステムの電圧レベルに依存しません。) または TUVcc (プローブ色: 黄色、ユーザシステム電源電圧) レベルの 2 つあります。

なお、トリガ出力 TUVcc については、ユーザシステム電源電圧 1.8V にての評価は行なえません (2.0V ~ 5.0V の範囲としてください)。

3.3.5 電源フォロワ回路

E6000 エミュレータに搭載されてる電圧フォロワ回路は、ユーザシステムの電圧レベルをモニタしています。E6000 エミュレータの電源はユーザシステムの電源レベルを生成し E6000 エミュレータ内に供給しているため、MCU 電源がユーザシステムから供給されることはありません。

E6000 エミュレータにユーザシステムインタフェースケーブルが接続されていないと、E6000 エミュレータ上の MCU は 5V で動作し、ユーザシステムインタフェースケーブルが接続されている場合は、ユーザシステムの電源電圧と同レベルの電圧で動作します。ユーザシステム Vcc が MCU の動作電圧よりも低い場合であっても、E6000 エミュレータは供給電圧を UVcc に一致させます。したがって 5V 以下で動作している場合は、エミュレーションクロックの周波数が各 Vcc における最高

3. ハードウェア

動作周波数を超えないように注意してください。

E6000 エミュレータコンフィグレーションダイアログボックスを使って、[User VCC Threshold] を 5V から 0V の範囲で設定できます。ユーザ Vcc がその値よりも下がった場合、システムステータスウィンドウの [User System Status] には [Down] が表示されます。User Vcc Threshold 電源レベルよりも高い場合は [OK] が表示されます。また、ユーザシステムインタフェースケーブル未接続時は、E6000 の Vcc レベルは 5V になります。

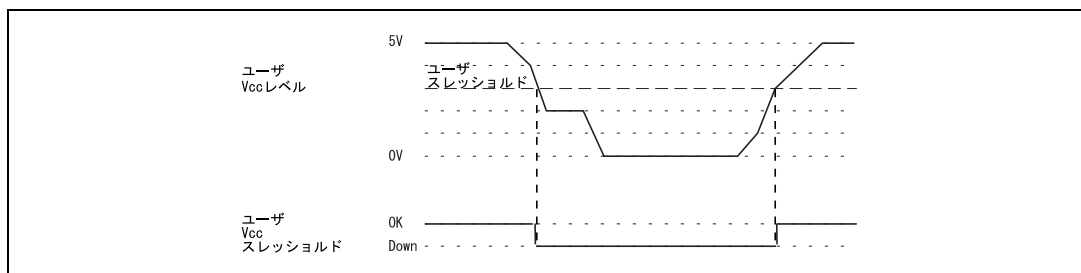


図 3-9 ユーザシステムと E6000 との Vcc の関係

3.4 MCU と E6000 エミュレータの相違点

E6000 エミュレータの電源投入後およびコマンドリセット後の、MCU と E6000 エミュレータのレジスタの初期値の相違を以下に示します。

表 3-1 MCU と E6000 エミュレータの相違

状態	レジスタ	E6000	エミュレータMCU
電源投入後	PC	不定	リセットベクタ値
	R0 to R6	0000	不定
	R7(SP)	0010	不定
	CCR	I マスクは 1 その他は不定	I マスクは 1 その他は不定
リセットコマンド後	PC	リセットベクタ値	リセットベクタ値
	R0 to R6	不定	不定
	R7 (SP)	0010	不定
	CCR	I マスクは 1 その他は不定	I マスクは 1 その他は不定

E6000 エミュレータの I/O ポート上の保護回路の詳細については、「3.3 ハードウェアインタフェース」を参照してください。

3.4.1 A/D コンバータ

ユーザシステムインタフェースケーブルで接続されているため、A/D 変換の精度は、MCU のハードウェアマニュアルに記載の精度より劣下します。

3.4.2 未使用領域のアクセス

未使用領域 H'FF80 ~ H'FF8F はエミュレータシステムで使用しています。MAP 設定でエミュレータに割り付けた場合、正常動作できませんので、本領域は使用しないでください。

3.4.3 Go Reset コマンドによるプログラム実行

Go Reset コマンドを用いてプログラムを実行する際、E6000 は約 500 μ s のリセット信号をエバチップに入力します。実行時間測定結果には、このリセット信号入力時間が加算されます。

3. ハードウェア

4. チュートリアル

本章では、E6000 エミュレータの主な特長を HDI の操作例に従って説明します。

チュートリアルでは、E6000 エミュレータ上のエミュレーションメモリを使用して実行しますので、E6000 エミュレータをユーザシステムに接続する必要はありません。

4.1 はじめに

このチュートリアルは、簡単な C プログラムで作成されています。

本章を読む前に、

- 「2 セットアップ」に従って、E6000 エミュレータを HDI で起動してください。このチュートリアルを使用するために E6000 エミュレータをユーザシステムに接続する必要はありません。
- MCU のアーキテクチャと命令セットについてよく理解してください。詳しくは、「H8/3802 シリーズのハードウェアマニュアル」を参照してください。

4.1.1 概要

チュートリアルは、NAME (アルファベット順)、AGE (昇順)、ID (昇順) を並び換えるプログラムです。ソースプログラム (tutorial.C) および Sysrof フォーマットのオブジェクトファイル (tutorial.abs) は、HDI のインストールディスク中に用意されています。

4.2 チュートリアルプログラムの動作

プログラムの最初の部分は、include するファイルの宣言です。

```
#include <machine.h>
#include "string.h"
```

次の部分は、プログラムの中で使われる定数、構造体および関数の初期値の定義です。

```
#define NAME    (short)0
#define AGE     (short)1
#define ID     (short)2
#define LENGTH  8

struct namelist {
    char    name[LENGTH];
    short  age;
    long   idcode;
};

struct namelist section1[] = {
    "Naoko", 17, 1234,
    "Midori", 22, 8888,
```

4. チュートリアル

```
    "Rie",    19, 7777,  
    "Eri",    20, 9999,  
    "Kyoko",  26, 3333,  
    "",       0,  0  
};
```

```
int count;
```

```
void sort();
```

main 関数は次のとおりです。

```
main( )  
{  
    count = 0;  
    for ( ; ; ){  
        sort(section1, NAME);  
        count++;  
        sort(section1, AGE);  
        count++;  
        sort(section1, ID);  
        count++;  
    }  
}
```

残りの部分は main 関数から呼び出される関数です。

```
void sort(list, key)  
struct namelist list[];  
short key;  
{  
    short i,j,k;  
    long min;  
    char *name;  
    struct namelist worklist;  
  
    switch(key){  
        case NAME :  
            for (i = 0 ; *list[i].name != 0 ; i++){
```

```
        name = list[i].name;
        k = i;
        for (j = i+1 ; *list[j].name != 0 ; j++){
            if (strcmp(list[j].name , name) < 0){
                name = list[j].name;
                k = j;
            }
        }
        worklist = list[i];
        list[i] = list[k];
        list[k] = worklist;
    }
    break;

case AGE :
    for (i = 0 ; list[i].age != 0 ; i++){
        min = list[i].age;
        k = i;
        for (j = i+1 ; list[j].age != 0 ; j++){
            if (list[j].age < min){
                min = list[j].age;
                k = j;
            }
        }
        worklist = list[i];
        list[i] = list[k];
        list[k] = worklist;
    }
    break;

case ID :
    for (i = 0 ; list[i].idcode != 0 ; i++){
        min = list[i].idcode;
        k = i;
        for (j = i+1 ; list[j].idcode != 0 ; j++){
            if (list[j].idcode < min){
                min = list[j].idcode;
            }
        }
    }
}
```

4. チュートリアル

```
        k = j;
    }
}
worklist = list[i];
list[i] = list[k];
list[k] = worklist;
}
break;
}
}
```

4.3 HDI の実行

HDI を実行するには、Start/Programs/Hdi_3800/[HDI for E6000 H8_3800]をクリックしてください。

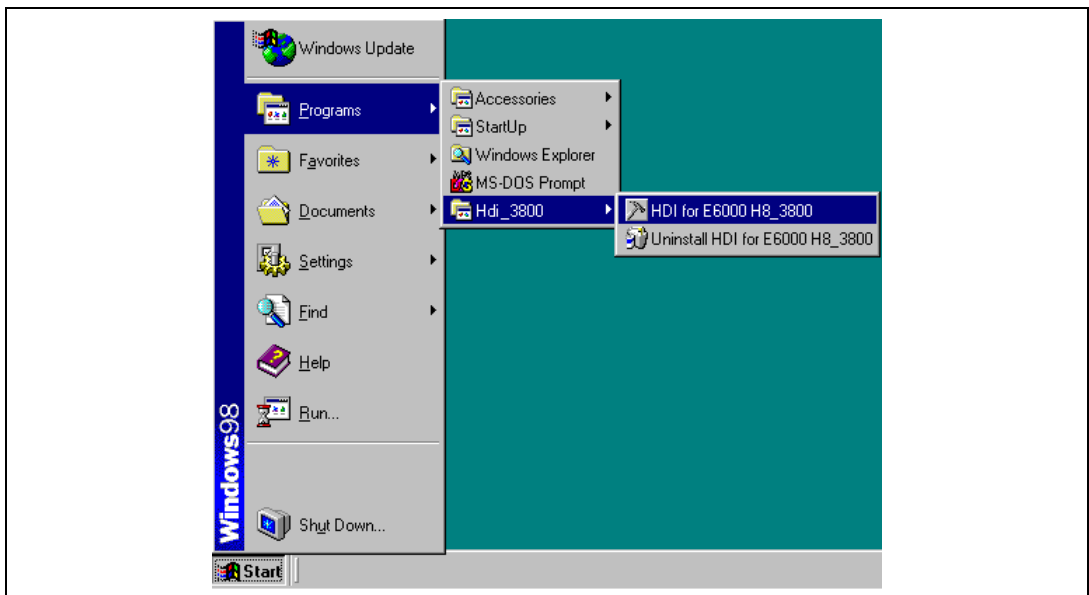


図 4-1 HDI 起動メニュー

4.3.1 ターゲットプラットフォームの選択

HDI は複数のターゲットプラットフォームをサポートする拡張機能があります。複数のプラットフォーム用にシステムがセットアップされると、使用するプラットフォームを選択する必要があります。

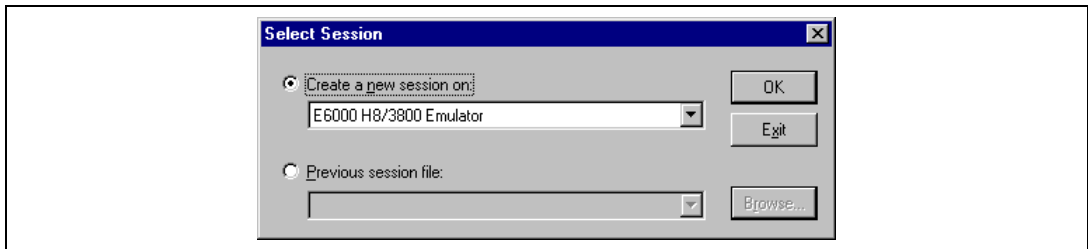


図 4-2 プラットホームの選択

- このチュートリアルでは、E6000 H8/3800 Emulatorを選択し、[OK] をクリックしてください。

[File] メニューから [New Session...] を選択すれば、いつでもターゲットプラットフォームを変更できます。

E6000 エミュレータが正しくセットアップされていれば、ステータスバーの [Link up] メッセージと共に、HDI ウィンドウが表示されます。以下にウィンドウの主な機能を示します。

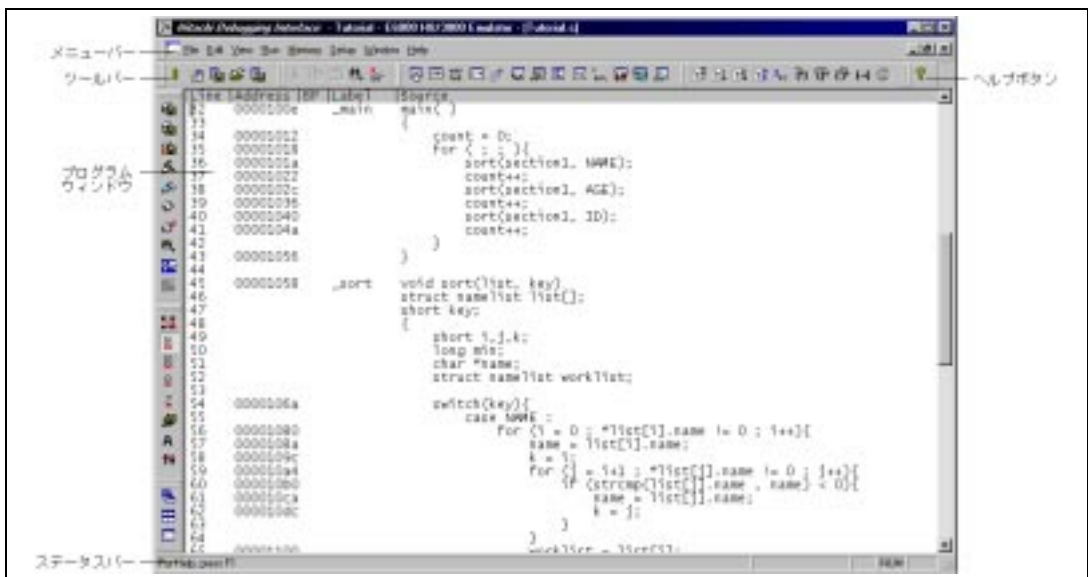


図 4-3 HDI ウィンドウ画面

HDI の主な機能については、「日立デバッグインタフェースユーザズマニュアル」をご覧ください。

4.3.2 メニュー

メニューバーには、E6000 エミュレータの環境設定または HDI のデバッグ機能を使用するためのコマンドがあります。（E6000 エミュレータをセットアップし HDI を使うための HDI コマンドへのアクセスを示します。）

ツールバー

よく使うメニューコマンドのショートカットとして便利なボタンです。

プログラムウインドウ

デバッグしているソースプログラムなどを表示します。

4. チュートリアル

ステータスバー

E6000 エミュレータの状態、例えばダウンロードの進捗状況や実行モードにおけるアドレスバスの状態を示します。

ヘルプボタン

HDI の使い方やコマンド構成についてのヘルプ画面を表示します。

4.4 E6000 エミュレータのセットアップ

E6000 エミュレータにプログラムをダウンロードする前に、E6000 に対象 MCU 条件を設定しなければなりません。以下の項目を設定する必要があります。

- ・ デバイスタイプ
- ・ 動作モード
- ・ 動作クロック
- ・ ユーザ信号
- ・ メモリマップ

以下に、チュートリアルプログラム用に E6000 エミュレータを設定する方法について述べます。

4.4.1 プラットフォームの構成

- ・ 選択したプラットフォームに固有の設定をするために、[Setup] メニューから [Configure Platform...] 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。

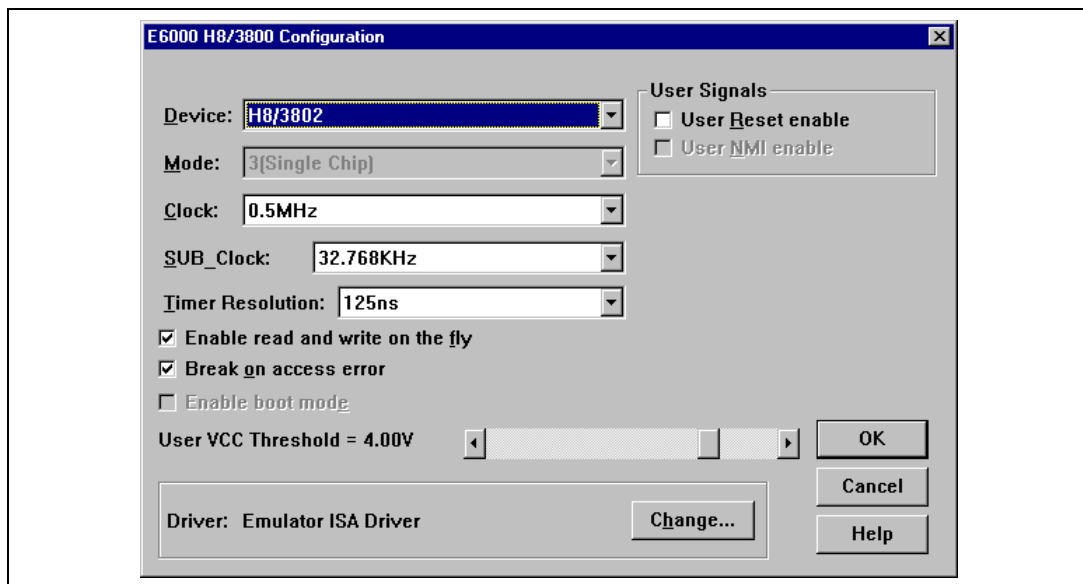


図 4-4 Configuration ダイアログボックス

- ・ オプションを以下のように設定してください。


表 4-1 コンフィグレーションオプションの設定例

オプション	設定値
デバイス (Device)	H8/3802
モード (Mode)	3 (シングルチップモード) 変更不可
動作クロック (Clock)	0.5MHz
タイマ分解能 (Timer Resolution)	125ns
ユーザシステムの電圧レベル (User Vcc Threshold)	4.00V
その他のオプション	イネーブル

- ・ [OK] をクリックしてターゲットコンフィグレーションを更新してください。

4.4.2 メモリマッピング

Configuration ダイアログボックスでデバイスおよびモードを選択すると、HDI は自動的に選択したデバイスおよびモードに合わせたマップの割り付けを行います。

- ・ 現在のメモリマップを表示するには、[Memory] メニューから [Configure Map] を選択するか、またはツールバーの [Memory Map] ボタンをクリックしてください。
[Memory Mapping] ダイアログボックスが以下のように表示されます。

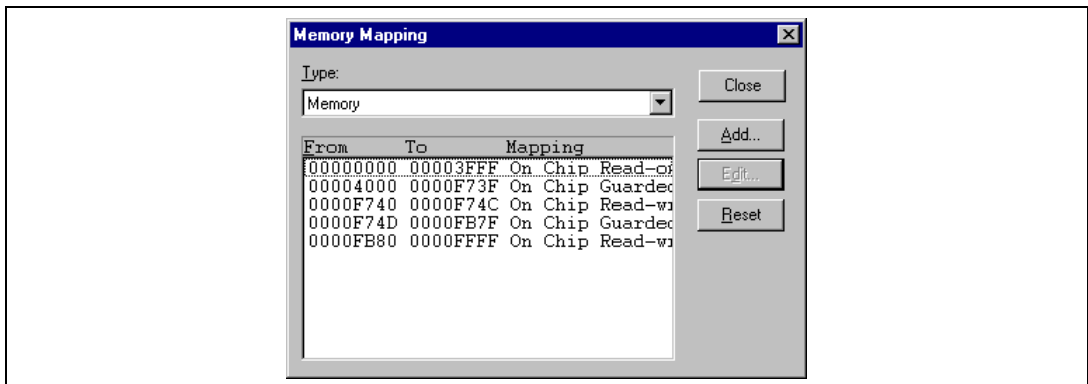


図 4-5 Memory Mapping ダイアログボックス

E6000 エミュレータメモリは、以下のタイプがあります。

表 4-2 メモリタイプの定義

メモリタイプ	説明
オンチップ (On Chip)	MCU 内蔵メモリをアクセスします。
エミュレータ (Emulator)	エミュレーションメモリをアクセスします。

また、アクセス制限については以下の 3 つのタイプがあります。

表 4-3 メモリタイプオプション

アクセスタイプ	説明
Read-Write	RAM
Read-Only	ROM
Guarded	アクセス不可

4. チュートリアル

本チュートリアルでは、デフォルトのマッピングを使用します。以下のように割り付け状態を見ることもできます。

- マッピング設定を変更する場合は、対象の設定値を選択して [Edit] ボタンをクリックするか、または対象のマッピング設定行をダブルクリックしてください。
ここでは、 [Memory Mapping] ダイアログボックスの On Chip Read-only の箇所をダブルクリックしてください。
[Edit Memory Mapping] ダイアログボックスが表示されます。

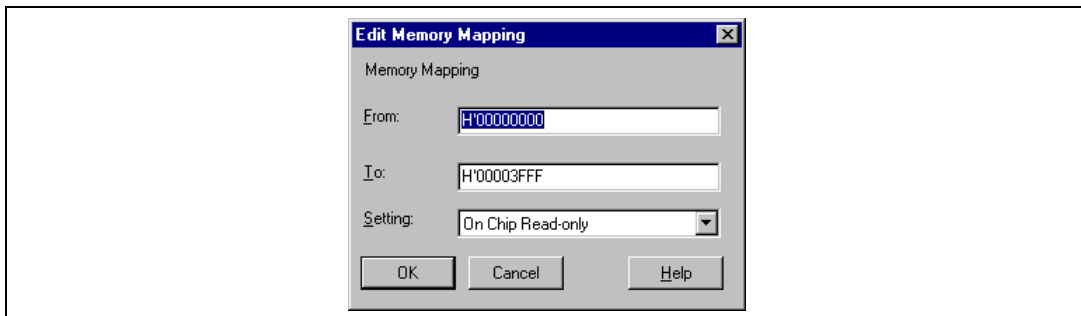



図 4-6 Edit Memory Mapping ダイアログボックス

- [OK] をクリックして、ダイアログボックスを閉じてください。
デバイスのマッピング情報を表示するには [View] メニューから [Status] を選択するか、またはツールバーの [Status] ボタン をクリックし System Status ウィンドウを開き、Memory シートを選択してください。デバイスのマッピング情報が以下のように表示されます。

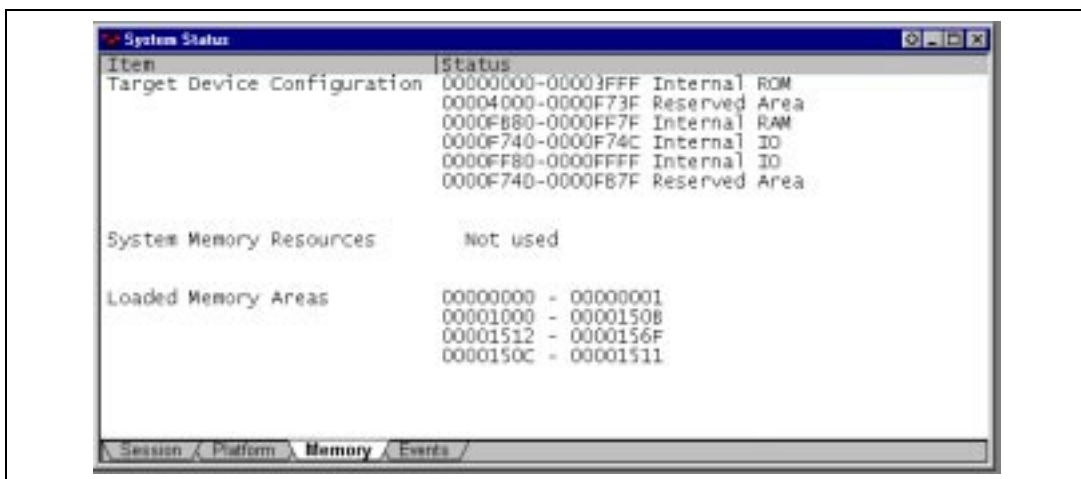


図 4-7 System Status ウィンドウ (Memory シート)


【注】 メモリマップは対象 MCU によって異なります

4.5 チュートリアルプログラムのダウンロード

E6000 エミュレータを上記のようにセットアップした後、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します。

4.5.1 オブジェクトファイルのダウンロード

最初に、以下のように Sysprof フォーマットオブジェクトファイルをロードしてください。

- ・ [File] メニューから [Load Program...] を選択するか、またはツールバーの [Load Program] ボタンをクリックしてください。

Load Program ダイアログボックスが開きます。

- ・ [Browse...] ボタンをクリックし、OpenダイアログボックスよりTutorialディレクトリの下
のTutorial.absファイルを選択した後、[Open] ボタンをクリックしてください。
Load Programダイアログボックスに戻りますので、さらに [Open] ボタンをクリックしファイルのダウンロードを開始してください。

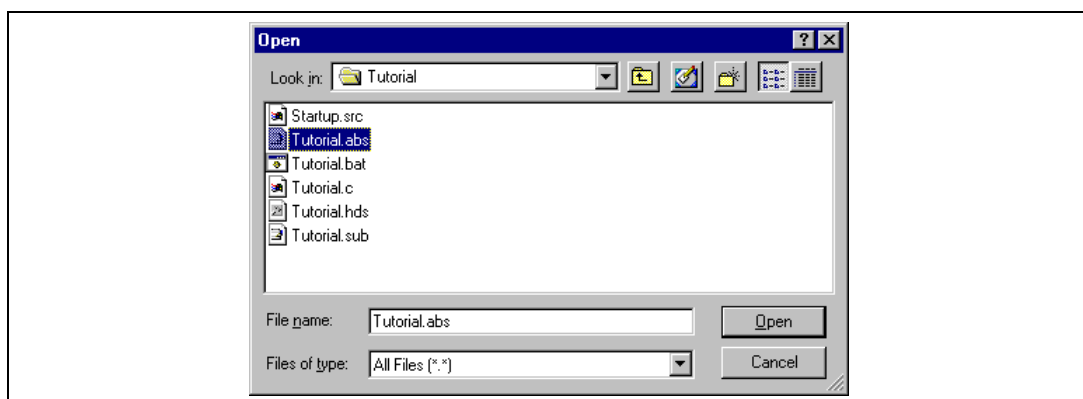


図 4-8 Open ダイアログボックス (オブジェクトファイルの選択)

ファイルがロードされると、以下のダイアログボックスにプログラムコードが書き込まれたメモリエリアに関する情報が表示されます。

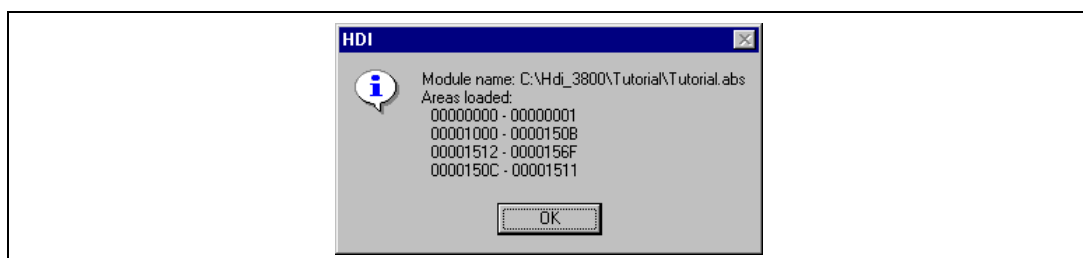


図 4-9 HDI ダイアログボックス

- ・ [OK] をクリックしてください。
プログラムは内蔵 ROM 領域にロードされました。

4. チュートリアル

4.5.2 プログラムリストの表示

HDI では、プログラムリストをソースコードやアセンブラニーモニックで表示することができます。

- ・ [View] メニューから [Source...] を選択するか、またはツールバーの [Program Source] ボタンをクリックしてください。
ロードしたオブジェクトファイルに対応する C ソースファイルを選ぶ必要があります。

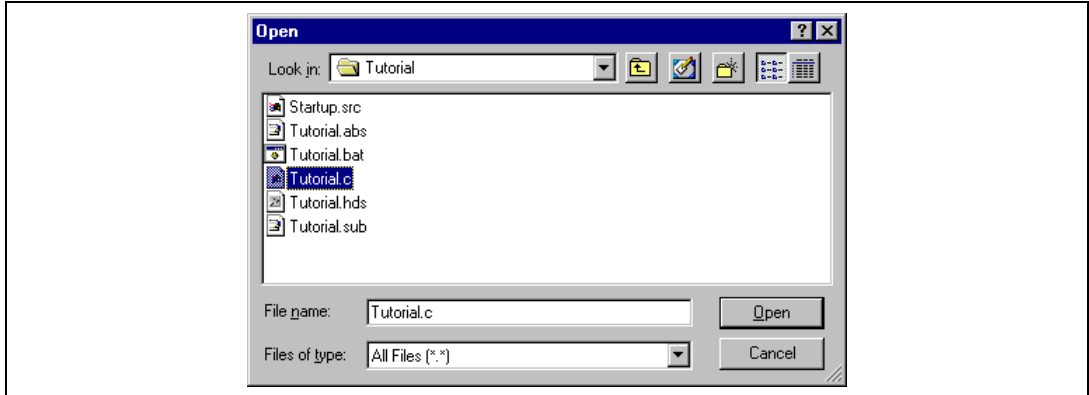


図 4-10 Open ダイアログボックス (ソースファイルの選択)

- ・ [tutorial.c] を選択し、[Open] をクリックしてプログラムウインドウを表示してください。

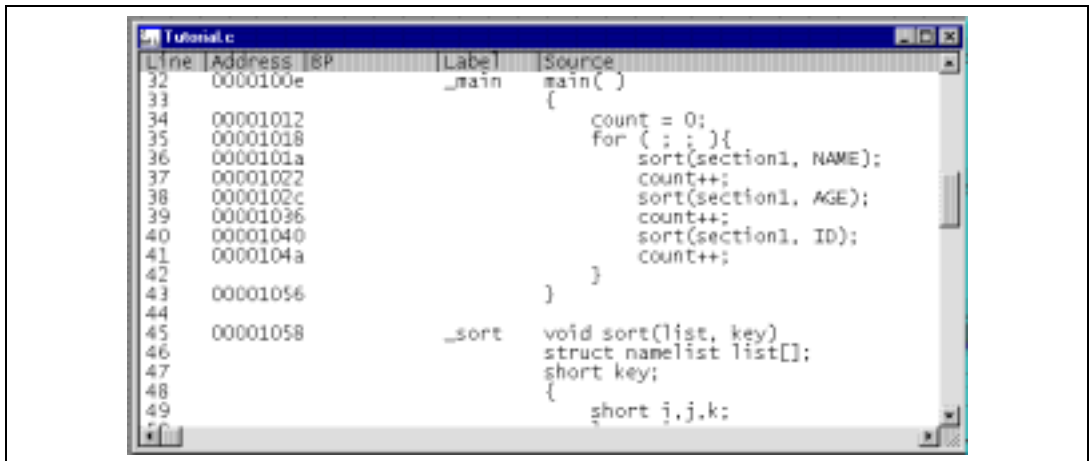


図 4-11 ソースプログラム画面

- ・ 必要ならば、[Setup] メニューの [Customise] サブメニューから [Font...] オプションを選択し、ホストPCに合ったフォントとサイズを選択してください。
プログラムウインドウを、最初に開いたときはメイン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プログラムをスクロールし、定義文等を見ることができます。

4.6 ブ레이크ポイントの使い方

最も簡単なデバッグ機能のひとつにプログラムの特定の箇所に達したときに実行を停止できるプログラムブレークポイントがあります。この機能を使用することによりプログラムが停止した時のMCUやメモリの状態を調べることができます。

4.6.1 プログラムブレークポイントの設定

プログラムウィンドウによって、プログラムのあらゆるポイントにブレークポイントを簡単に設定できます。たとえば、以下のようにしてアドレス H'1036 にブレークポイントを設定します。

- ・ H'1036番地を含むラインの [BP] カラムをダブルクリックしてください。

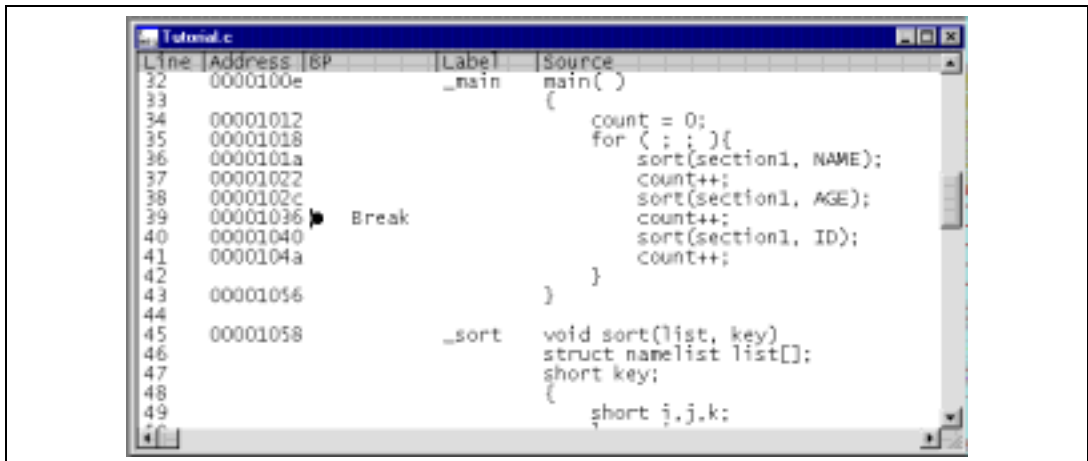



図 4-12 ブ레이크ポイントの設定

その位置に “ Break ” が表示され、そのアドレスにプログラムブレークポイントが設定されたことを示します。また、本章では実行しませんが、さらにダブルクリックしていくことによりイベント間実行測定のイベント設定 (“ +Timer ” で測定開始、“ -Timer ” で測定終了)、Point to Point トレース制御の設定 (“ +Trace ” でトレース開始、“ -Trace ” でトレース停止)およびトレースストップの設定 (“ TrStop ” でトレースストップ)ができます。これらはダブルクリックすることにより、以下のような順序でサイクリックに設定できます。

“ Blank ” “ Break ” “ +Timer ” “ -Timer ” “ +Trace ” “ -Trace ” “ TrStop ” “ Blank ” ...
 または
 “ -Trace ”

4.6.2 プログラムの実行

リセットベクタで指定されているアドレスからプログラムを実行するには、

- ・ [Run] メニューから [Reset Go] を選択するか、またはツールバーの [Reset Go] ボタン  をクリックしてください。

プログラムはブレークポイントを設定したところまで実行し、プログラムが停止した位置を示すためにプログラムウィンドウ中でステートメントが強調表示されます。

4. チュートリアル

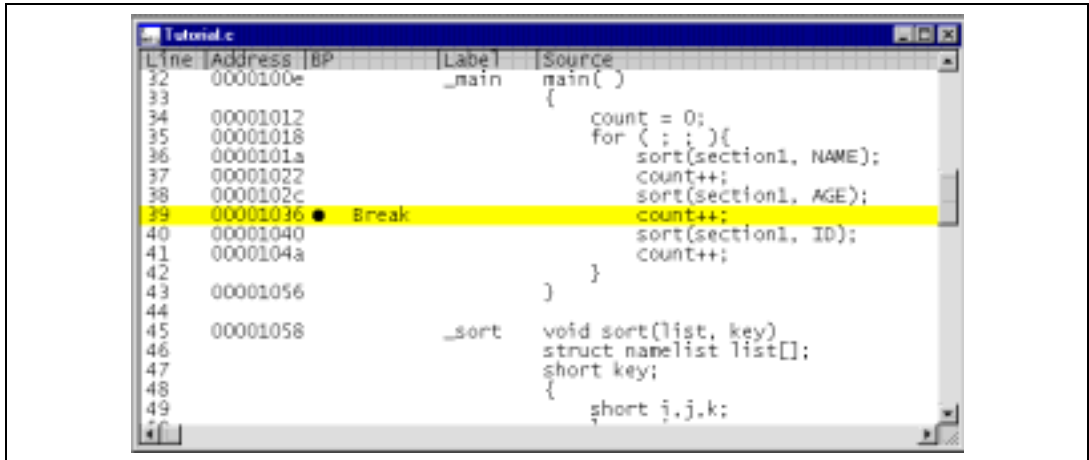



図 4-13 ステートメントの強調表示

[Break=Soft Ware Breakpoint] メッセージがステータスバーに表示され、ブレイクの原因を示します。

また、System Status ウィンドウでも最後のブレイクの原因が確認できます。

- [View] メニューから [Status] を選択するか、またはツールバーの [Status] ボタンをクリックして、System Statusウィンドウを開きPlatformシートを選択してください。

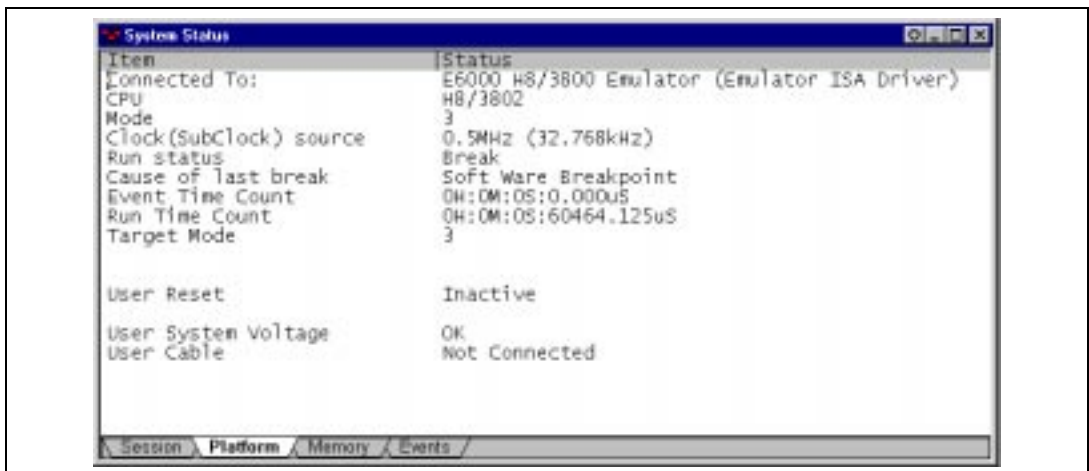



図 4-14 System Status ウィンドウ (Platform シート)

[Cause of last break]のラインは、ブレイクの原因がプログラムブレイクであることを示しています。

4.6.3 レジスタ内容の参照

プログラムが停止している間に、MCU レジスタの内容を参照できます。それらは Registers ウィンドウに表示されます。

- [View] メニューから [Registers] を選択するか、またはツールバーの [CPU Registers] ボタンをクリックしてください。

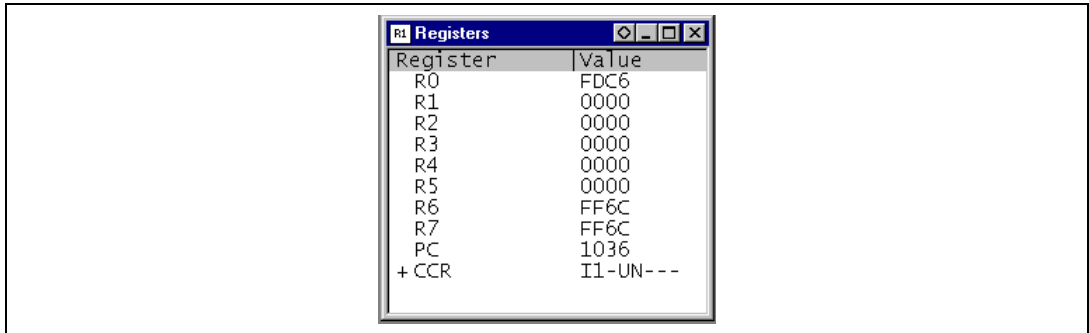


図 4-15 Register Window 画面

プログラムカウンタ PC の値は強調表示されたステートメント H'1036 になっています。

【注】 その他のレジスタの値は上に示すものとは異なることがあります。

レジスタの値は Registers ウィンドウで変更できます。

- ・ PCの値を変えるには、Registers ウィンドウで [PC] に対応する[Value]カラムをダブルクリックしてください。

以下のダイアログボックスによって値を編集できます。

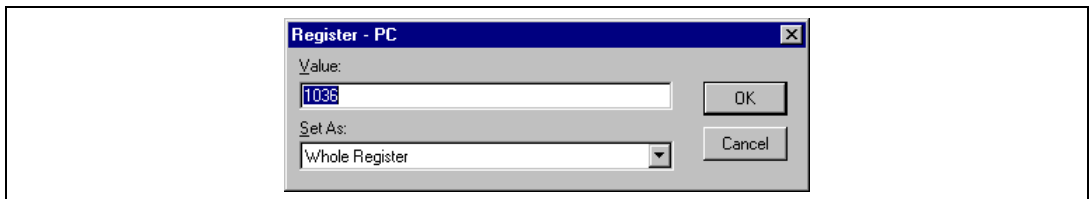




図 4-16 レジスタ値の編集

- ・ 値をH'102C（前のステートメントのアドレス）に変更し、[OK] をクリックしてください。強調表示されたバーがプログラムウィンドウの前のステートメントに移動し、新しいプログラムカウンタの値を示します。
- ・ [Run] メニューから [Go] を選択するか、またはツールバーの [Go] ボタンをクリックし、ブレークポイントまでの実行を再開してください。

4.6.4 ブレークポイントの確認

プログラムに設定した全てのブレークポイントの一覧をブレークポイントウィンドウで見ることができます。

- ・ [View] メニューから [Breakpoints] を選択するか、またはツールバーの [Breakpoints] ボタンをクリックしてください。

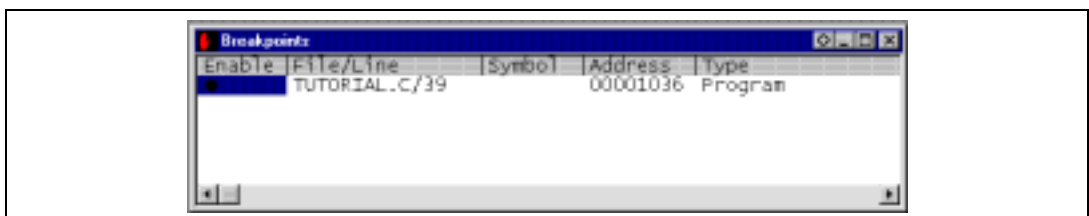


図 4-17 Breakpoints ウィンドウ

4. チュートリアル

Breakpoints ウィンドウによって、ブレークポイントの許可または禁止、新しいブレークポイントの設定、およびブレークポイントの削除ができます。

次へ進む前に、以下のようにブレークポイントを削除してください。

- Breakpoints ウィンドウのブレークポイントを強調表示し、[Delete] をクリックしてください。
- Breakpoints ウィンドウを閉じてください。


4.7 メモリと変数の表示

メモリ領域の内容を参照することにより、またはプログラム中で使われる変数の値を表示することによって、プログラムの動作をモニタできます。

4.7.1 メモリを表示する

メモリブロックの内容を Memory ウィンドウで見ることができます。

たとえば、Byte で構造体 [section1] に対応したメモリを見る場合：

- [View] メニューから [Memory...] を選択するか、またはツールバーの [Memory] ボタン  をクリックしてください。
- [Address] フィールドに section1 を入力し、[Format] を Byte に設定してください。

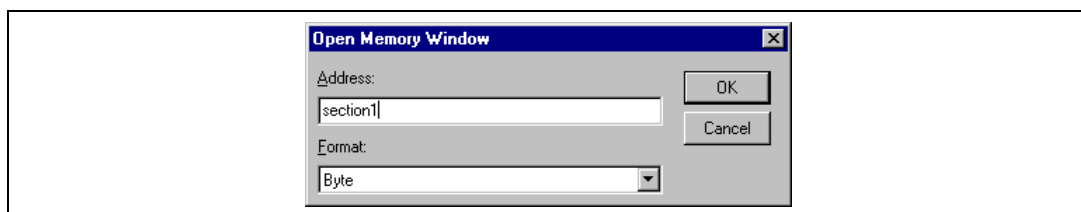


図 4-18 Memory Window の設定

- [OK] をクリックして、指定されたメモリ領域を示す Memory ウィンドウを開いてください。

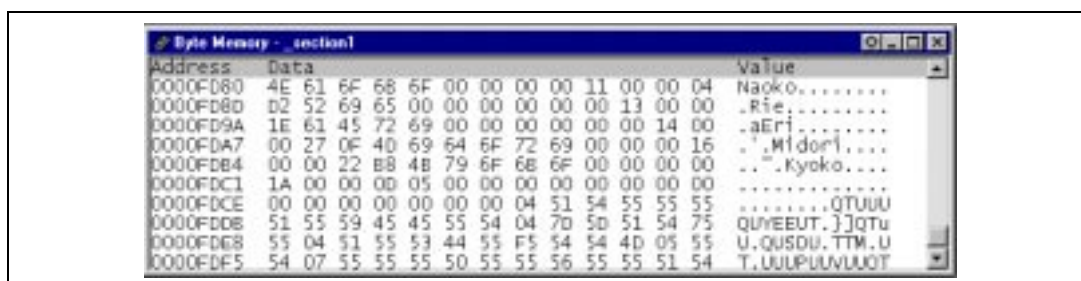


図 4-19 Memory ウィンドウ (Byte)

- [section1] 列の内容をモニタできるように、Memory ウィンドウを開いたままの状態にしてください。

4.7.2 変数を表示する

プログラムをステップ処理するとき、プログラムで使用される変数の値を見ることができ、期待した様にそれらが変化することを確認できます。

たとえば以下の手順で、プログラムの始めに宣言した構造体変数 “ section1 ” を見ることができま

す。

- ・ `sort (section1, ID);`のラインが見えるように、プログラムウィンドウをスクロールアップしてください。
- ・ プログラムウィンドウの`section1`の左にカーソルを置くようにクリックしてください。
- ・ マウスの右ボタンでプログラムウィンドウをクリックし、ポップアップメニューを表示し、ポップアップメニューより [Add Watch] を選択してください。

Watch ウィンドウに変数が表示されます。

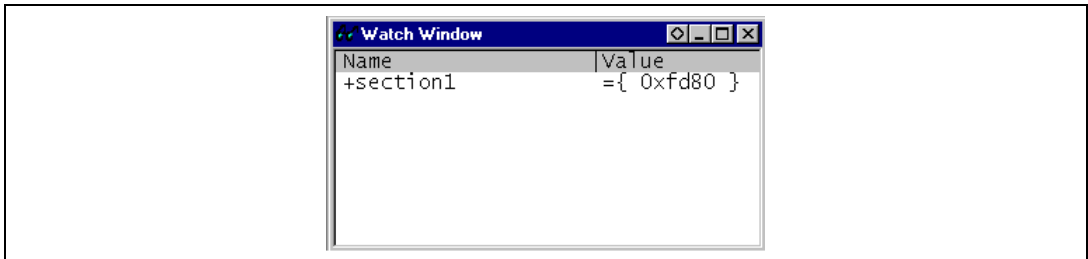


図 4-20 Watch 画面

Watch ウィンドウのシンボル `Section1` の左にある "+" をダブルクリックし、シンボルを拡張して各配列の要素を見ることができます。

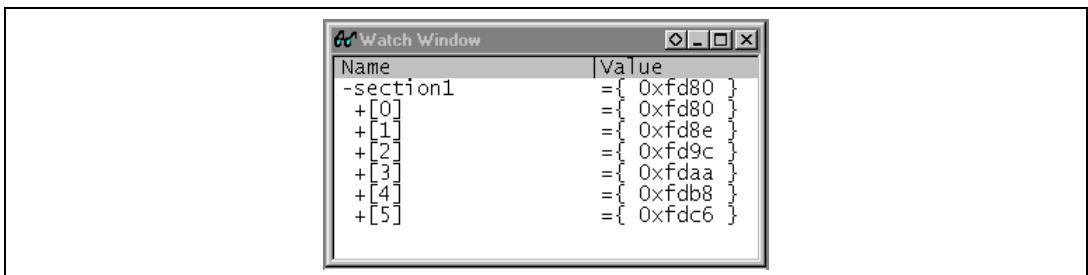


図 4-21 Watch 画面 (シンボル拡張)

また、変数名を指定して、Watch ウィンドウに変数を追加することもできます。

- ・ マウスの右ボタンでWatchウィンドウをクリックし、ポップアップメニューから [Add Watch...] を選択してください。
- ・ 変数名 "count" を入力し、 [OK] ボタンをクリックしてください。

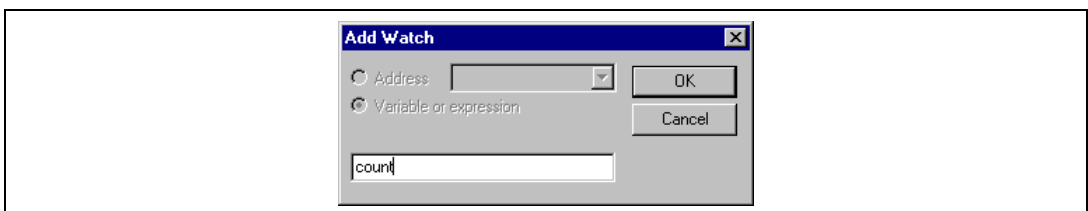


図 4-22 Add Watch ダイアログボックス

Watch ウィンドウに `int` 型の変数 "count" が追加されます。

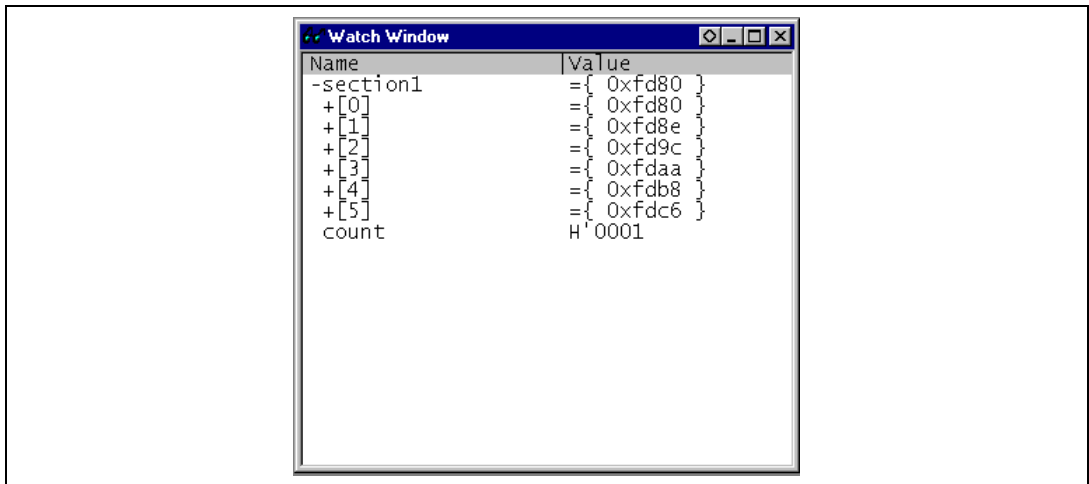


図 4-23 Watch ウィンドウ (変数の追加)

4.8 プログラムのステップ実行

E6000 エミュレータは、プログラムのシングルステップにおけるオプションを備えており、命令やステートメントを一度に実行します。表 4.4 に示すようなステップオプションがあります。

表 4-4 プログラムステップオプション

コマンド	説明
Step in	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)。
Step Over	呼び出された関数の全ステートメントを実行します。
Step out	関数を抜け出し、関数を呼び出したプログラムにおける次のステートメントで停止します。
Step...	指定したステートメント数ステップ実行します。

4.8.1 シングルステップ

- ・ PC=H'1036にブレークポイントを設定してください。
- ・ さらに [Step In] コマンドによって、sortファンクションコールまでプログラムを実行してください。sort(section1, ID)のステートメントが強調表示されます。

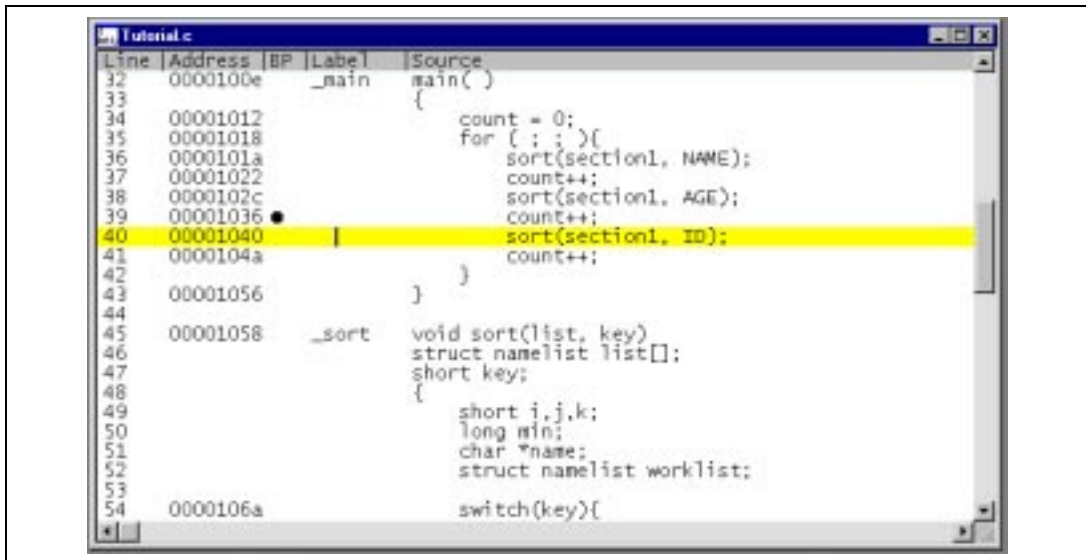



図 4-24 Step In 実行後のプログラムウインドウの表示 (1)

- sort中のステートメントをステップ実行するために [Run] メニューから [Step In] を選択するか、またはツールバーの [Step In] ボタン  をクリックしてください。

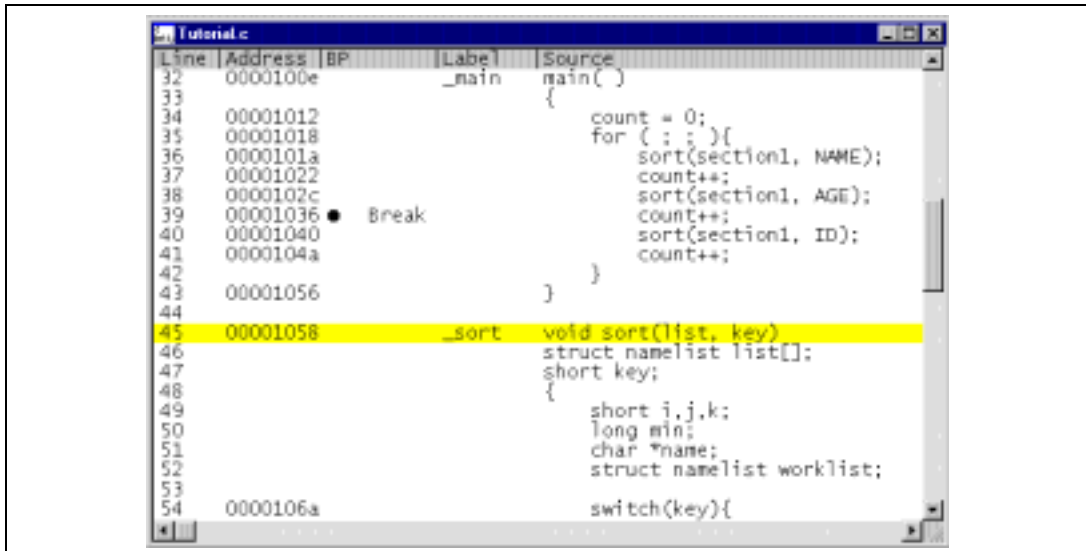



図 4-25 Step In 実行後のプログラムウインドウの表示 (2)

- [Run] メニューから [Step Out] を選択するか、またはツールバーの [Step Out] ボタン  をクリックして関数を抜け出し、メイン関数内の次のステートメントに戻ってください。アドレス H'104a が強調表示されます。

4. チュートリアル

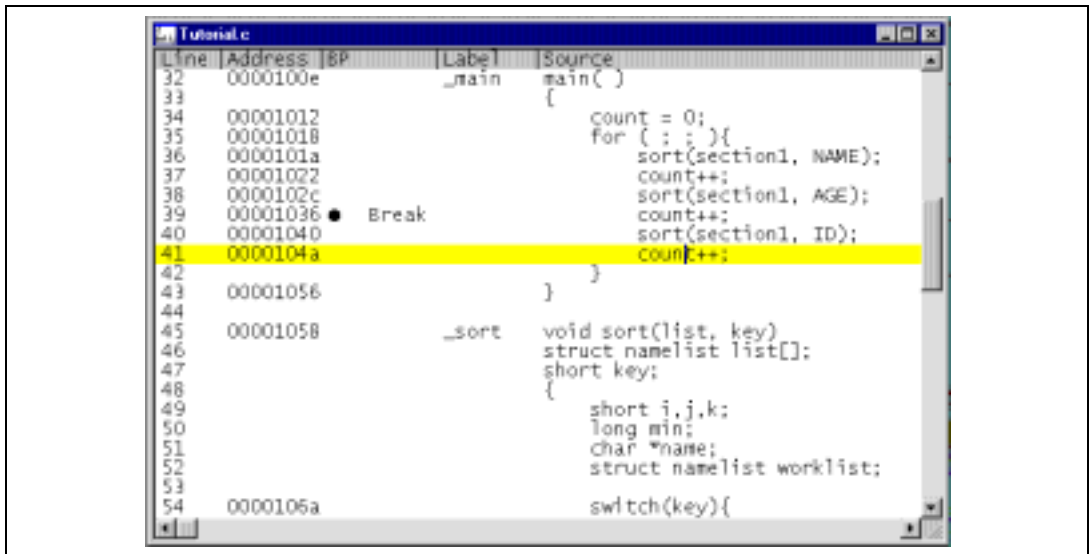


図 4-26 Step Out 実行後のプログラムウィンドウ画面

- さらに2回 [Step In] コマンドによってsortファンクションコールまでプログラムを実行してください。

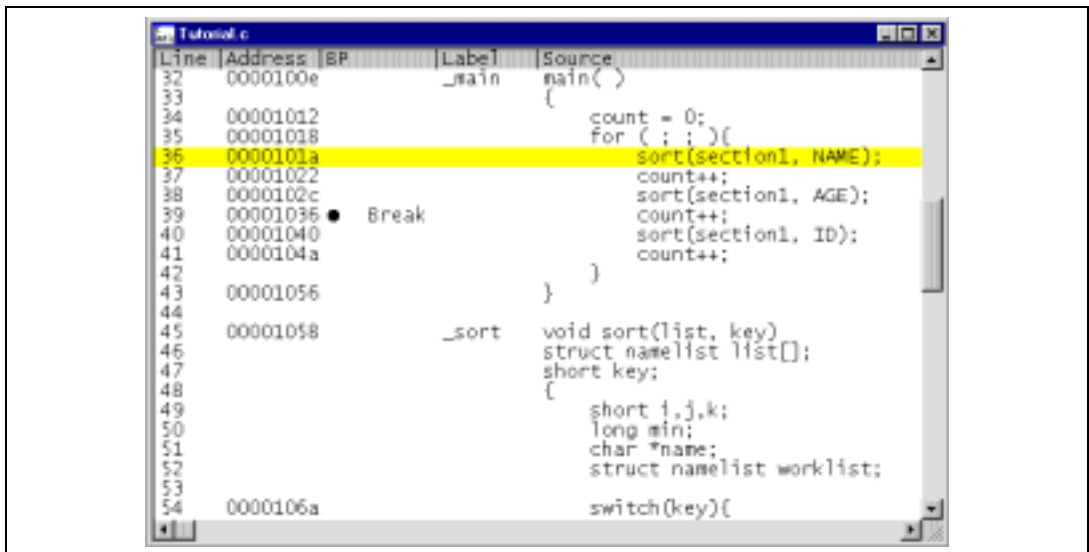



図 4-27 Step In 実行後のプログラムウィンドウの画面 (3)

4.8.2 関数全体のステップ実行

[Step Over] コマンドは、関数本体をシングルステップすることなく実行し、メインプログラムの中の次のステートメントで停止します。

- [Run] メニューから [Step Over] を選択するか、またはツールバーの [Step Over] ボタン  をクリックしてください。

プログラムは sort 関数を実行し、次のアドレス H'1022 で停止します。

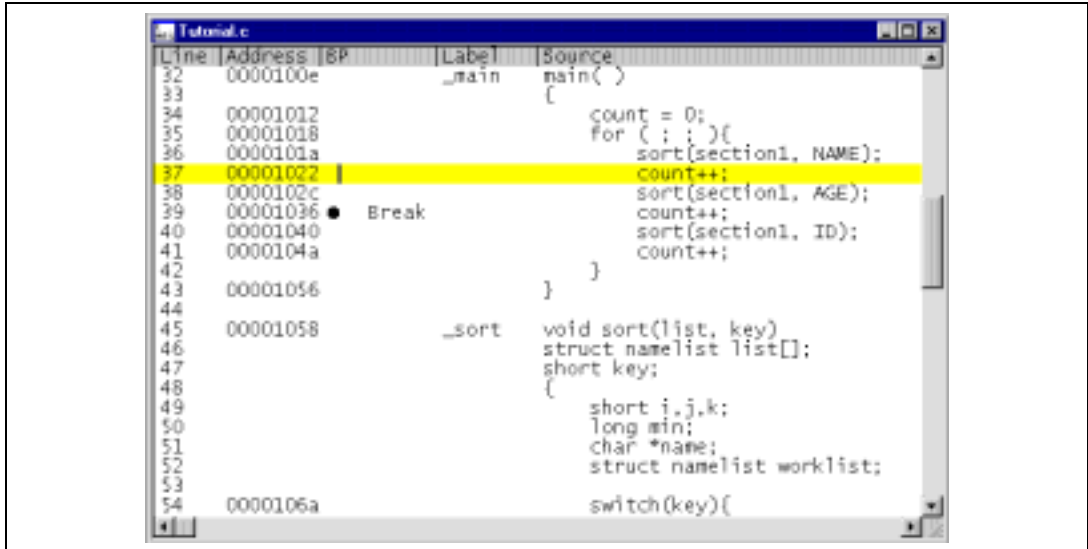



図 4-28 Step Over 実行後のプログラムウインドウの画面

4.8.3 ローカル変数の表示

Locals 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、sort 関数のローカル変数を調べます。

- ・ [Run] メニューから [Step In] を選択するか、またはツールバーの [Step In] ボタン  を 2回クリックして、sort関数の実行を開始してください。

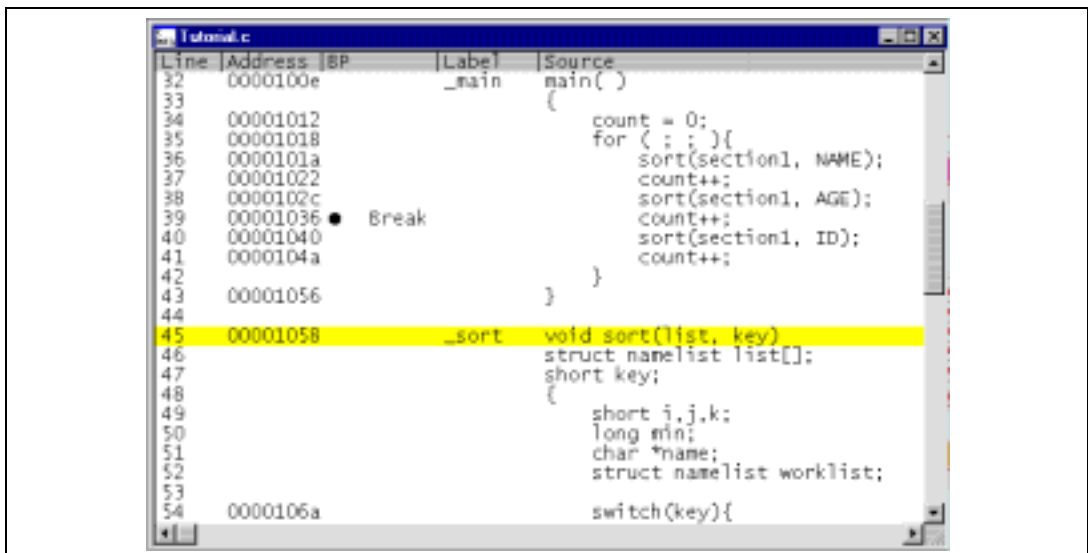




図 4-29 Step In 実行後のプログラムウインドウ画面 (4)

4. チュートリアル

- [View] メニューから [Locals] を選択するか、またはツールバーの [Locals] ボタンをクリックして、Localsウインドウを開いてください。
最初は、ローカル変数宣言が行われていないため、[Locals] ウインドウは何も表示しません。
- [Run] メニューから [Step In] を選択するか、またはツールバーの [Step In] ボタンを11回クリックしてステップ実行を行ってください。
[Locals] ウインドウは、ローカル変数とその値を表示します。

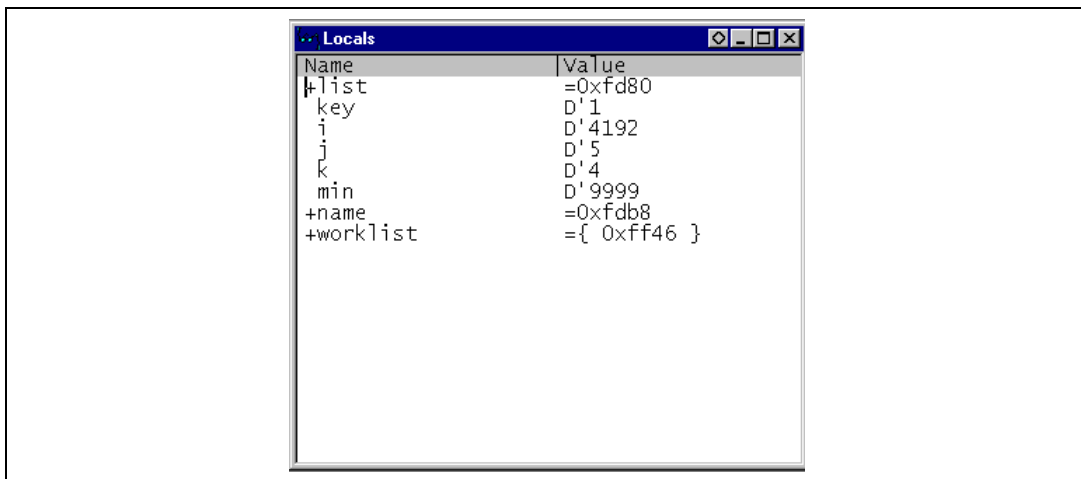


図 4-30 ローカル変数の表示

- Localsウインドウの変数worklistの前にあるシンボル + をダブルクリックし、配列worklistの構成要素を表示させてください。

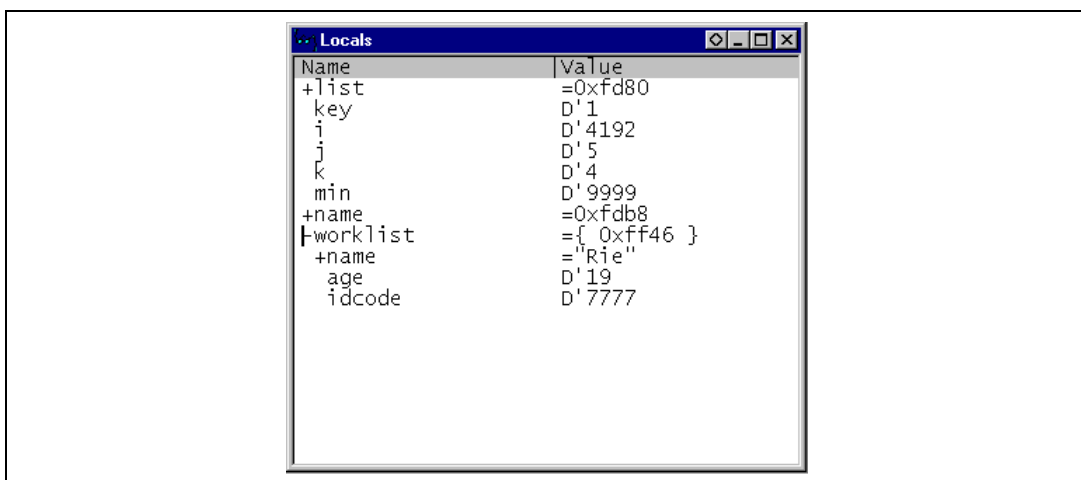



図 4-31 ローカル変数の表示 (配列要素の表示)

- [Run] メニューから [Step Out] を選択するか、またはツールバーの [Step Out] ボタンをクリックして、メインプログラムに戻ってください。


4.9 イベント検出システムの使用方法

本チュートリアルでは、Memory ウィンドウでメモリ領域の内容を見ること、あるいは Watch ウィンドウおよび Locals ウィンドウで変数の値を見ることによって、プログラムの動作をモニタしてきました。

しかしプログラムの動作は非常に複雑なため、メモリ領域をモニタしたり、変数を見たりできないことがあります。E6000 エミュレータのイベント検出システムを使用することによりプログラムが H'109E をアクセスした時を検出することができます。

4.9.1 イベント検出システムによる ハードウェアブレークポイントの設定

イベント検出システムを使用したハードウェアブレークポイントを設定して、以下のようにプログラムの一部をモニタしてください。

- [View] メニューから [Breakpoint] を選択するか、またはツールバーの [Breakpoint] ボタンをクリックして、Breakpoints ウィンドウを表示してください。
- 新しいブレークポイントを設定するため、Breakpoints ウィンドウ内でマウスの右ボタンをクリックし [Add...] を選択してください。

以下のダイアログボックスが現われ、ブレークポイントの属性を設定できます。

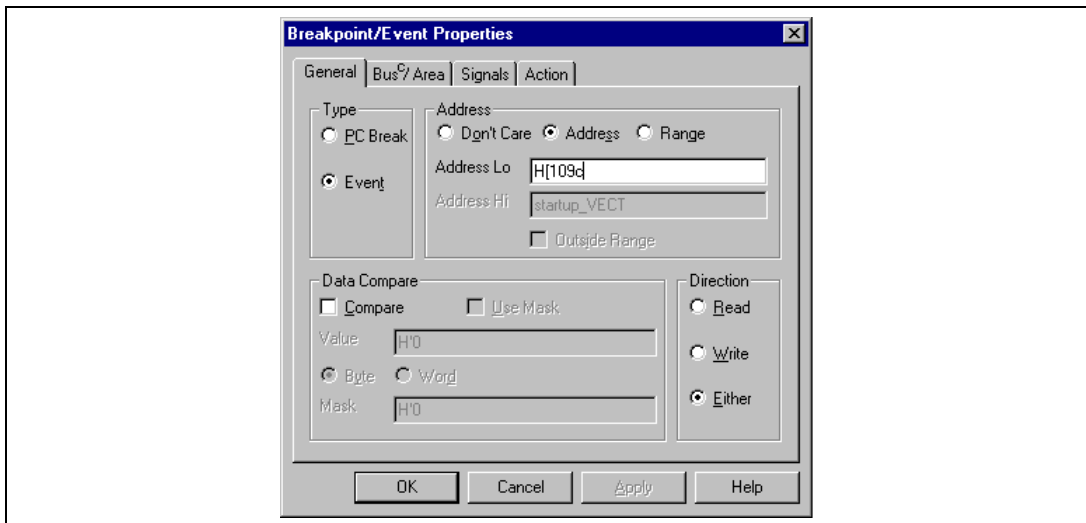


図 4-32 ブレークポイントの追加

- [Type] 選択を [Event] にし、イベント条件として [Address Lo] ボックスにアドレス H'109c を入力してください。
- [OK] をクリックしてブレークポイントを設定してください。
これによって、アドレス H'109c がアクセス（読み出しまたは書き込み）されたときにブレークします。

Breakpoints ウィンドウは、設定された新しいハードウェアブレークポイントを表示します。

4. チュートリアル

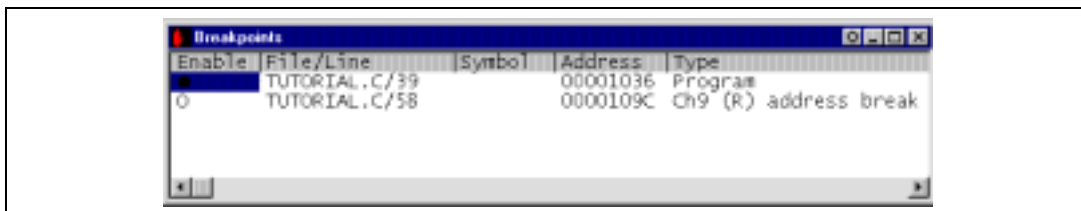



図 4-33 Breakpoint Window 画面（追加後）

- ・ [Run] メニューから [Go] を選択するか、あるいはツールバーの [Go] ボタンをクリックして、プログラムを現在の位置から実行してください。
アドレス H'109c で実行が停止します。

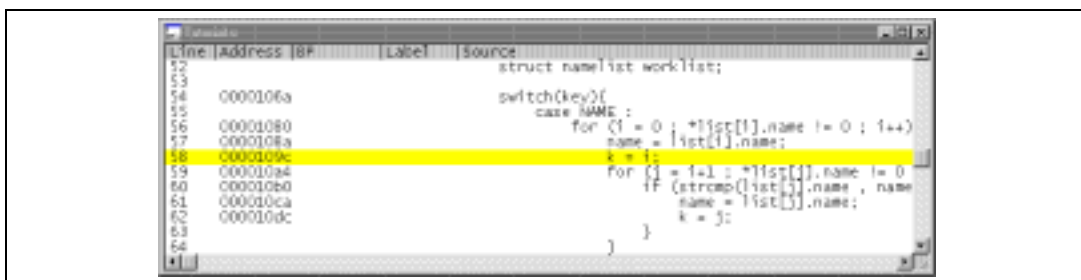


図 4-34 ブレークポイントによるプログラムの停止


ステータスバーには、“BREAK = Event Break”と表示し、イベント条件の一致によってブレークが発生したことを示します。

4.10 トレースバッファの使い方

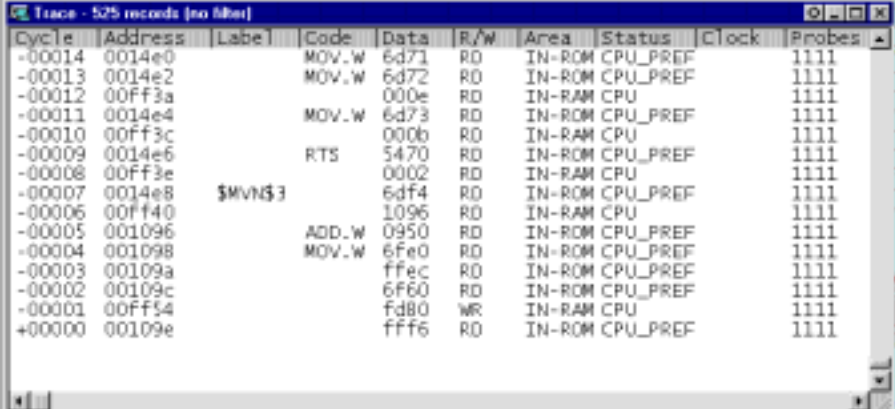
MCU の動作を確認するため、指定されたイベントの直前までの MCU サイクルはトレースバッファに記録されています。

4.10.1 トレースバッファの表示

プログラムのアクセスアドレスを指定し、トレースバッファ内の MCU サイクルを調べることによって、どのようなアクセスが起こったかを知ることができます。

- ・ [View] メニューから [Trace] を選択するか、あるいはツールバーの [Trace] ボタンをクリックして、Trace ウィンドウを開いてください。

必要ならば、最後の数サイクルが見えるようにウィンドウをスクロールダウンしてください。Trace ウィンドウが以下のように表示されます。



Cycle	Address	Label	Code	Data	R/W	Area	Status	Clock	Probes
-00014	0014e0		MOV.W	6d71	RD	IN-ROM CPU_PREF		1111	
-00013	0014e2		MOV.W	6d72	RD	IN-ROM CPU_PREF		1111	
-00012	00ff3a			000e	RD	IN-RAM CPU		1111	
-00011	0014e4		MOV.W	6d73	RD	IN-ROM CPU_PREF		1111	
-00010	00ff3c			000b	RD	IN-RAM CPU		1111	
-00009	0014e6		RTS	5470	RD	IN-ROM CPU_PREF		1111	
-00008	00ff3e			0002	RD	IN-RAM CPU		1111	
-00007	0014e8	\$MVN\$3		6df4	RD	IN-ROM CPU_PREF		1111	
-00006	00ff40			1096	RD	IN-RAM CPU		1111	
-00005	001096		ADD.W	0950	RD	IN-ROM CPU_PREF		1111	
-00004	001098		MOV.W	6Fe0	RD	IN-ROM CPU_PREF		1111	
-00003	00109a			ffec	RD	IN-ROM CPU_PREF		1111	
-00002	00109c			6F60	RD	IN-ROM CPU_PREF		1111	
-00001	00ff54			fd80	WR	IN-RAM CPU		1111	
+00000	00109e			fff6	RD	IN-ROM CPU_PREF		1111	

図 4-35 Trace Window 画面

- 必要ならば、タイトルバーのすぐ下のラベルの横にあるカラムディバイダをドラッグして、カラムの幅を調節してください。

【注】 H8/3802 シリーズでは clock count には何も表示されません。

ソフトウェアブレークで停止した場合、Code カラムに “Data ”、Data カラムに “ 5770 ” が表示されます。

4.10.2 トレースフィルタの設定

現在 Trace ウィンドウは、すべての MCU サイクルを表示しています。

- マウスの右ボタンで Trace ウィンドウをクリックし、ポップアップメニューより [Clear] を選択して、現在のトレースバッファを削除してください。
- 同様に [Filter...] を選択して、Trace Filterダイアログボックスを表示してください。

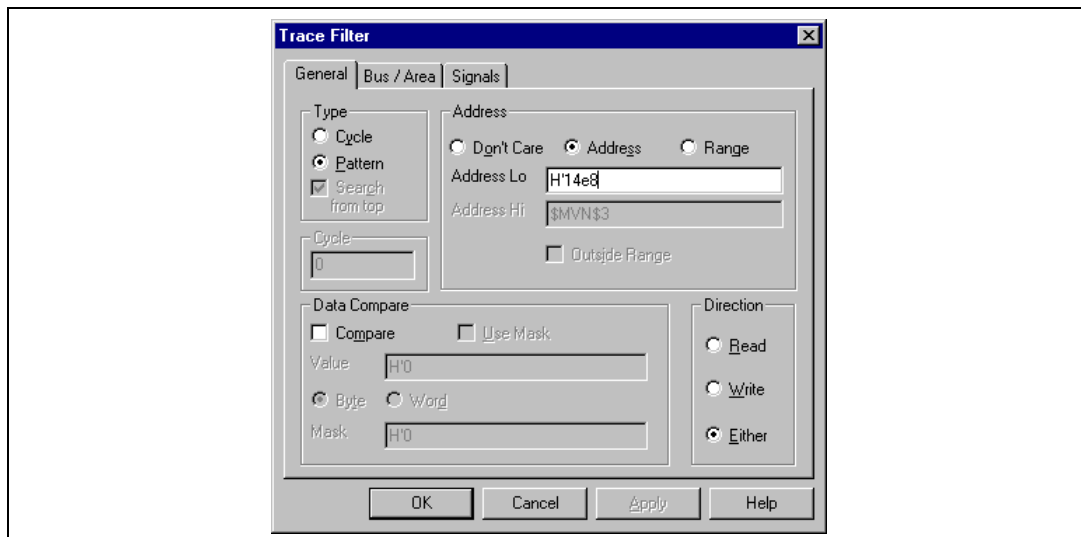


図 4-36 Trace Filter ダイアログボックス

4. チュートリアル

これによって、トレースバッファに表示されるサイクルを限定するためのフィルタ条件を設定できます

- 必要ならば、[General] をクリックして、[General] パネルを表示してください。
- [Type] セクションで [Pattern] タイプを選択してください。
- [Address] セクションで、[Address] をクリックし、[Address Lo] フィールドに H'14e8 と入力してください。
- [Bus / Area] をクリックし、[Bus / Area] パネルを表示してください。
- [Don't Care] チェックボックスのチェックをはずし、[Bus State] を [CPU_PREFETCH] に設定してください。

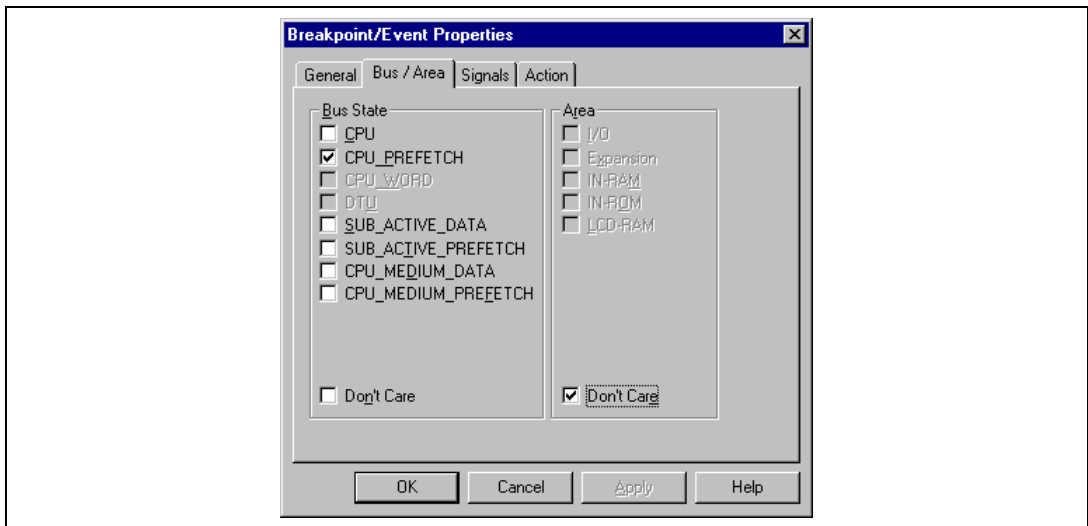


図 4-37 Bus / Area の設定

- [OK] をクリックして、トレースフィルタを保存してください。
- [View] メニューから [Breakpoints] を選択して、ブレークポイントウィンドウを開き、全てのブレークポイントを削除してください。
- [Run] メニューから [Go] を選択して、プログラムの終わりまで実行してください。
- [Run] メニューから [Halt] を選択して、トレースバッファを見ることができるように行を停止してください。

トレースウィンドウには、MCU が H'14e8 番地をアクセスしたサイクルだけが表示されます。

Cycle	Address	Label	Code	Data	R/W	Area	Status	Clock	Probes	HWI	Source
-01516	0014e8	\$MvNS1	6df4	RD	IN-ROM CPU_PREF	1111					
-01487	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-01452	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-01383	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-01336	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-01301	0014e8	\$MvNS1	6df4	RD	IN-ROM CPU_PREF	1111					
-01291	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-01175	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-01091	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-01080	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-00914	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-00903	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-00710	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-00693	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-00644	0014e8	\$MvNS1	6df4	RD	IN-ROM CPU_PREF	1111					
-00609	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-00599	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-00413	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-00399	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-00388	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-00222	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-00211	0014e8	\$MvNS2	6df4	RD	IN-ROM CPU_PREF	1111					
-0001E	0014e8	\$MvNS1	6df4	RD	IN-ROM CPU_PREF	1111					

図 4-38 Trace Window 画面 (トレースフィルタ指定)

4.11 セッションの保存

終了する前に、次のデバッグセッションで同じ E6000 エミュレータと HDI コンフィグレーションを使用して再開できるように、セッションを保存しておく和良好的でしょう。

- ・ [File] メニューから [Save Session] を選択してください。
- ・ [File] メニューから [Exit] を選択して、HDIを終了してください。

4.11.1 さてつぎは？

このチュートリアルは、E6000 エミュレータのいくつかの主な特長と、HDI の使用方法を紹介しました。E6000 エミュレータで提供されるエミュレーションツールを組み合わせることによって、非常に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、識別することにより、それらの問題点を効果的に調査することができます。

HDI の使用方法に関する詳細については、別に発行されている「日立デバッグインタフェースユーザーズマニュアル」を参照してください。

5. H8/3802 シリーズ用 E6000 エミュレータ HDI の機能

本章は、H8/3802 シリーズ専用の HDI の特徴に関する情報について述べます。あらゆるターゲットに共通する HDI の一般的な特長に関しては、別に発行されている「日立デバッグインタフェースユーザズマニュアル」を参照してください。以下に HDI メニューと「日立デバッグインタフェースユーザズマニュアル」(HDI マニュアル) および本マニュアルに記載する項目の対応表を示します。

表 5-1 HDI のメニューとマニュアルの対応表

メニューバー	プルダウン メニュー	HDI マニュアル	本マニュアル
File Menu	New Session...		
	Load Session...		
	Save Session		4.11
	Save Session As...		
	Load Program...		4.5
	Initialise		
	Exit		
Edit Menu	Cut		
	Copy		
	Paste		
	Find...		
	Evaluate...		
View Menu	Breakpoints		4.6.4, 4.9.1, 5.2, 5.3
	Command Line		5.7
	Disassembly...		
	I/O Area		
	Labels		
	Locals		4.8.3
	Memory...		4.7.1
	Performance Analysis		
	Registers		4.6.3
	Source...		4.5
	Status		4.6.2
	Trace		4.10, 5.5
	Watch		4.6

表 5-1 HDI のメニューとマニュアルの対応表 (つづき)

メニューバー	プルダウン メニュー	HDI マニュアル	本マニュアル
Run Menu	Reset CPU		
	Go		4.6.2
	Reset GO		4.6.2
	Go to Cursor		
	Set PC To Cursor		
	Run...		
	Step In		4.8
	Step Over		4.8
	Step Out		4.8
	Step...		
	Halt		
Memory Menu	Refresh		
	Load...		
	Save...		
	Verify...		
	Test...		
	File...		
	Copy...		
	Compare...		
	Configure Map...		4.4.2
	Configure Overlay...		
Setup Menu	Status bar		
	Options...		
	Radix		
	Customise		
	Configure Platform...		4.4.1, 5.1
Window Menu	Cascade		
	Tile		
	Arrange Icons		
	Close All		
Help Menu	Index		
	Using Help		
	Search for Help on		
	About HDI		

〃

5.1 コンフィグレーションダイアログボックス

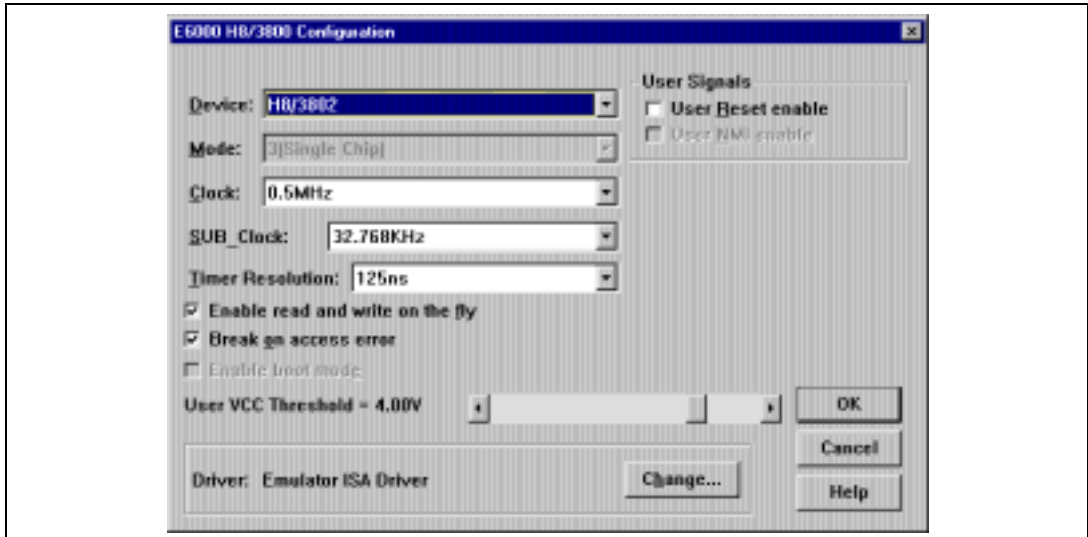


図 5-1 Configuration ダイアログボックス

Configuration ダイアログボックスによって、E6000 エミュレータをセットアップします。

Configuration ダイアログボックスを表示するには、[Setup] メニューから [Configure Platform...] を選択してください。

以下の表は、Configuration ダイアログボックスで表わされるオプションについて説明します。

表 5-2 コンフィグレーションダイアログボックス

オプション	説明
デバイス (Device)	エミュレーションする MCU を指定します。
モード (Mode)	MCU の動作モードを指定します。動作モードは 3 に固定されています。
エミュレーションクロック (Clock)	MCU クロックの速度を指定します。H8/3802 シリーズは、8MHz、2MHz、0.5MHz、または Target/2 が選択できます。MCU のサブクロックを指定します。32.768kHz、38.4kHz、307.2kHz、または Target が選択できます。
タイマ分解能 (timer Resolution)	実行時間の測定に使用する最小時間を指定します。20ns、125ns、250ns、500ns、1 μ s、2 μ s、4 μ s、8 μ s、16 μ s のいずれかの値に設定できます。
ユーザ信号 (User Signals)	ユーザシステムからのリセット信号を有効または無効にできます。ボックスをチェックすると、信号が有効になります。
プログラム実行中のリードライトの許可 (Enable read and write on the fly)	プログラム実行中でのユーザメモリのリードライトを有効または無効にできます。
ブートモード (Enable boot mode)	MCU のフラッシュメモリへのブートプログラミング動作を有効にします。(H8/3802 シリーズは使用できません。)
アクセスエラーでのブレーク (Break on access error)	不当なアクセス (ROM 領域へのライト等) ブレークを有効または無効にできます。チェックしなければ、ROM へのあらゆる書き込みおよびアクセス禁止された領域へのアクセスが無視されます。
ユーザ Vcc スレッシュホールド (User VCC Threshold = x.xxV)	ユーザシステム電圧レベルを監視します。スレッシュホールドによって設定された値より低くなると、ユーザ Vcc が低いことをシステムステータスウインドウによってユーザに知らせます。

5.2 ブレークポイント

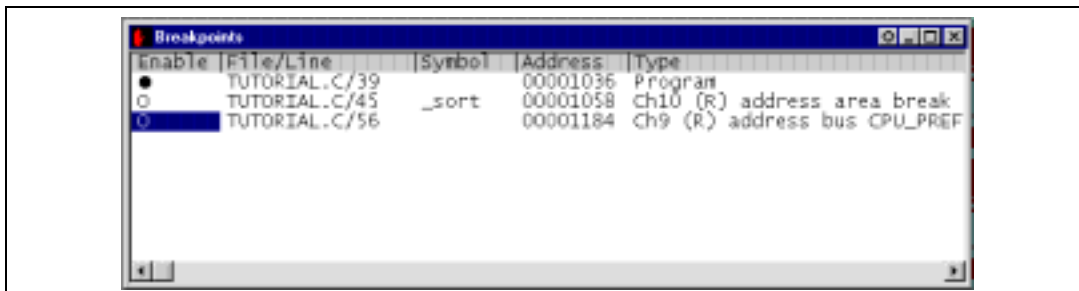


図 5-2 Breakpoints 画面

Breakpoints ウィンドウは設定された全てのブレークポイントの一覧を表示します。

Breakpoints ウィンドウを表示するには、[View] メニューから [Breakpoints] を選択してください。

現状のブレークポイントを変更するには、それをダブルクリックするか、あるいは [Breakpoints] リストの中から選択した後、ポップアップメニューより [Edit...] を選択してください。

ブレークポイントを許可または禁止するには、[Breakpoints] リストの中から選択した後、ポップアップメニューより [Disable/Enable] を選択してください。ブレークポイントが許可されると、[Enable] コラムの中に表示されます。

ブレークポイントを削除するには、ブレークポイントリストの中から選択した後、ポップアップメニューより [Delete] を選択してください。全てのブレークポイントを削除するには、[Delete All] を選択してください。

新しいブレークポイントを設定するには、ポップアップメニューより [Add...] を選択して、[Breakpoint/Event Properties] ダイアログボックスを表示してください。そして、付け加えたいブレークポイントの条件を設定してください。

[Breakpoints/Event Properties] ダイアログボックスの詳細に関しては、以下の「5.3 イベント検出システム」を参照してください。

5.2.1 プログラムブレークポイントを設定する

プログラムブレークポイントを設定するには、[Type] を [PC Break] に設定し、[Address Lo] フィールドにブレークポイントのアドレスを入力してください。

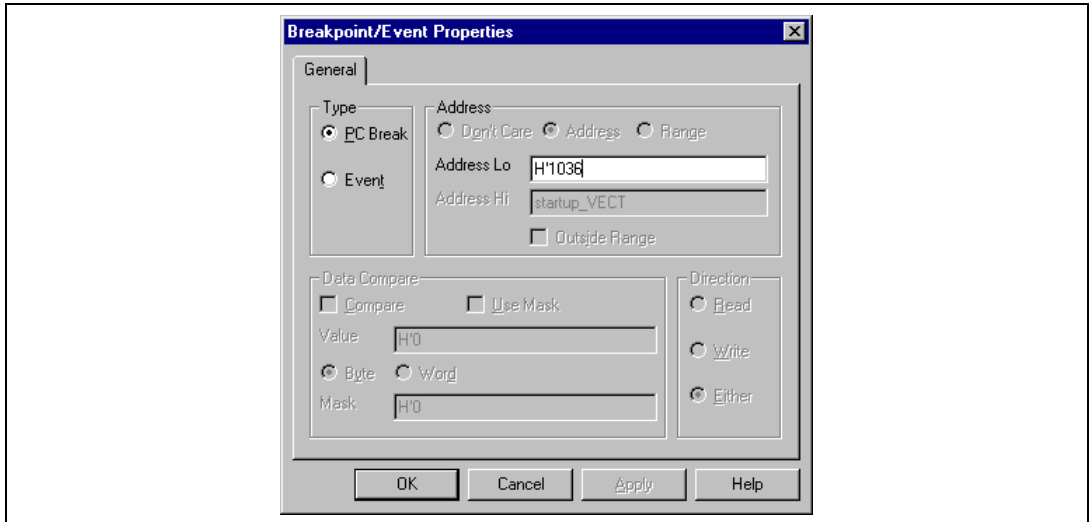


図 5-3 ブレークポイントの設定

あるいは、プログラムウィンドウの [Break] カラムをダブルクリックしてください。

5.3 イベント検出システム

イベント検出システムに、MCU 信号の状態を指定したイベントを設定できます。このイベントは E6000 エミュレータのトレース、ブレーク、およびイベント間実行時間測定機能を制御します。

イベント検出システムはイベントチャンネルおよび範囲チャンネルで構成し、指定されたイベントがいつ発生したかを検出できます。また、最大 8 イベントによるシーケンス指定も可能です。この場合、それぞれのイベントは、シーケンスにおける前のイベントの発生によって起動、あるいは停止します。

以下の表は、イベントチャンネルおよび範囲チャンネルで指定できる条件を示します。

表 5-3 イベントチャンネルとオプションの設定

オプション	イベントチャンネル	範囲チャンネル
アドレスまたはアドレス範囲内		
アドレス範囲外		
データの値 (マスク機能付)		
リードまたはライト		
アクセスタイプ (命令プリフェッチなど)		
アクセス領域 (内蔵 ROM、内蔵 RAM など)		
外部プローブの値		
イベントの一致回数		
シーケンスでの組み合わせが可能		

[Breakpoint/Event Properties] ダイアログボックスによって、各イベントをブレーク、トレース、もしくはイベント間実行時間測定の内、どれに使用するか設定します。

イベントをブレークに設定するには、[Type] を [Event] に設定してください。すると、[Breakpoint/Event Properties] ダイアログボックスは、4 つのオプションパネルを表わし、ブレークポイントで使うイベントの全ての条件 (General, Bus / Area, Signals, Action) を設定できます。

5.3.1 [General]

[General] 属性パネルによって、イベントチャンネルのアドレスとデータアクセス条件を設定できます。

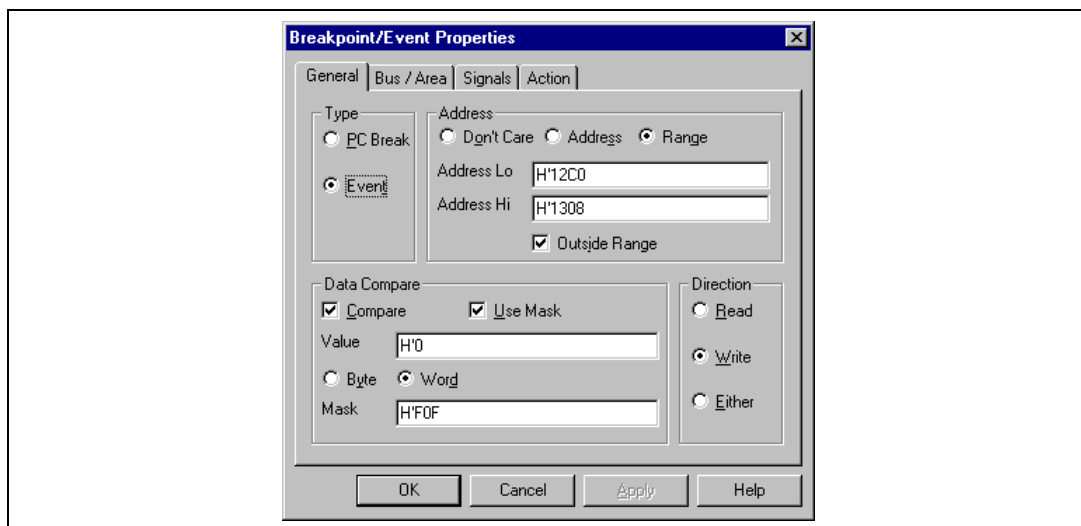


図 5-4 イベントの設定 (General)

アドレス

アドレスまたはアドレスの範囲がアクセスされたときに、チャンネルを起動できます。指定された範囲外のアドレスへアクセスするとチャンネルを起動するように指定するには、[Outside Range] を選択してください。

データコンペア

設定した値とデータバスの値が一致したときにチャンネルを起動できます。データの特定のビットをマスクする場合は、[Use Mask] を選択してください。

ディレクション

リード、ライト、またはリードライトアクセス時にチャンネルが起動できます。

5.3.2 [Bus / Area]

特定のバスステートまたはアクセスされたメモリ領域でチャンネルを起動できます。

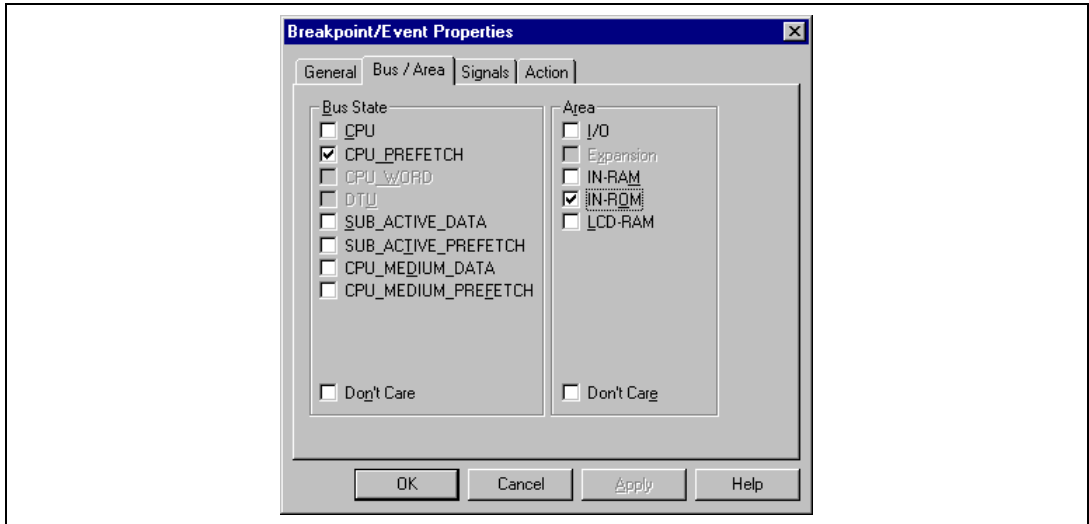


図 5-5 イベントの設定 (Bus / Area)

5.3.3 [Signals]

4つの外部プローブ信号の特定の組み合わせでイベントを起動するように指定します。

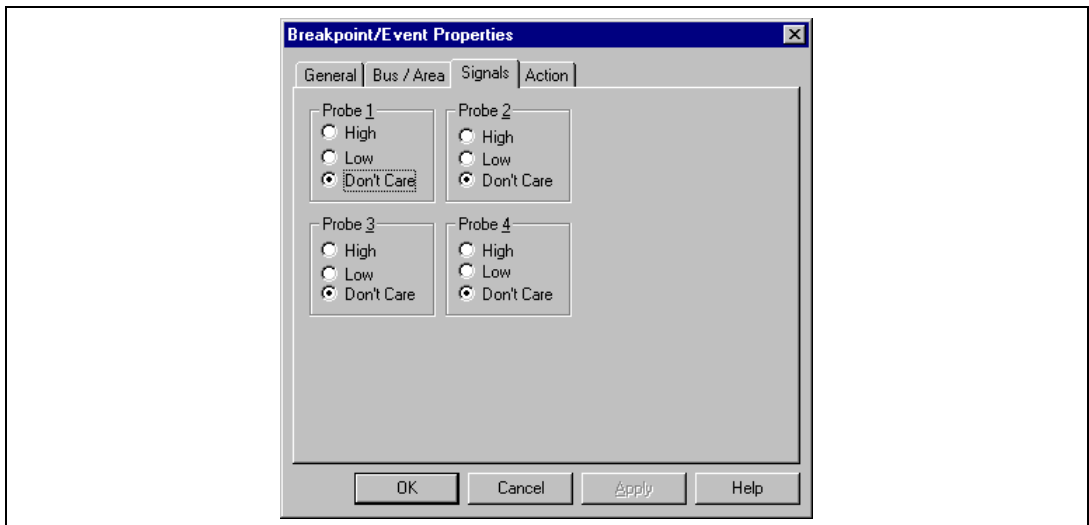


図 5-6 イベントの設定 (Signals)

5.3.4 [Action]

イベントが起動されるときアクションを指定します。

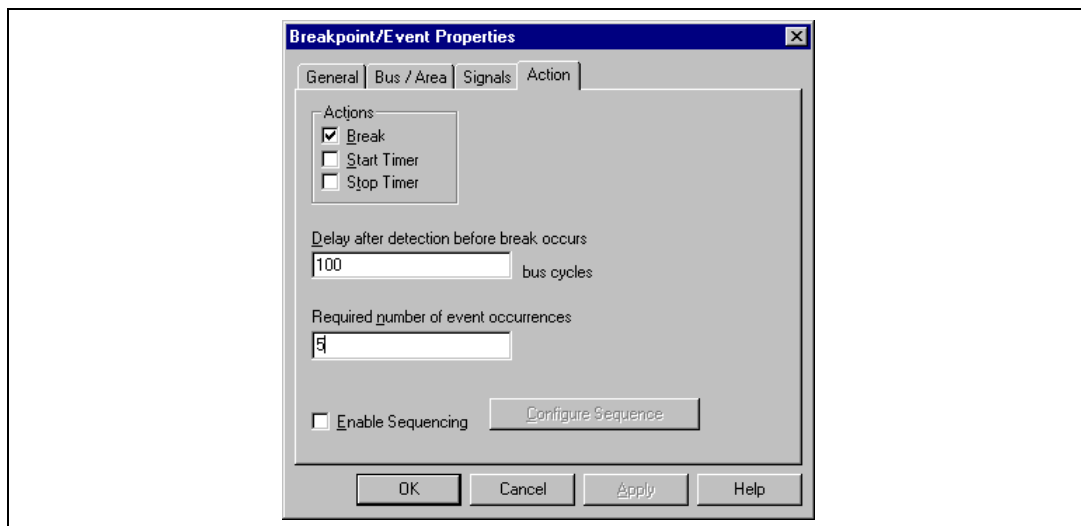


図 5-7 イベントの設定 (Action)

表 5-4 のアクションが指定できます。

表 5-4 イベントアクション

アクション	説明
ブレーク	プログラムの実行を停止します。
タイマ開始	イベント間実行時間測定タイマを開始します。タイマ分解能については、「5.1 コンフィグレーションダイアログボックス」を参照してください。
タイマ停止	イベント間実行時間測定タイマを停止します。

イベントが起動された後、指定された数のバスサイクルだけチャンネルの起動を遅らせるには、[Delay after detection before break occurs] フィールドにバスサイクル数を入力してください。
 指定された回数だけイベントが起動されるまでチャンネルの起動を遅らせるには、[Required number of event occurrences] フィールドに必要なイベント発生数を入力してください。
 イベントのシーケンスを生成するには、シーケンスの一部を形成する全てのイベントの [Enable Sequencing] オプションを選択してください。

5.3.5 イベントシーケンス

シーケンスを構成するには、[Action] で [Enable Sequencing] をチェックし、[Configure Sequence] ボタンをクリックしてください。[Event Sequencing] ダイアログボックスが表示されます。

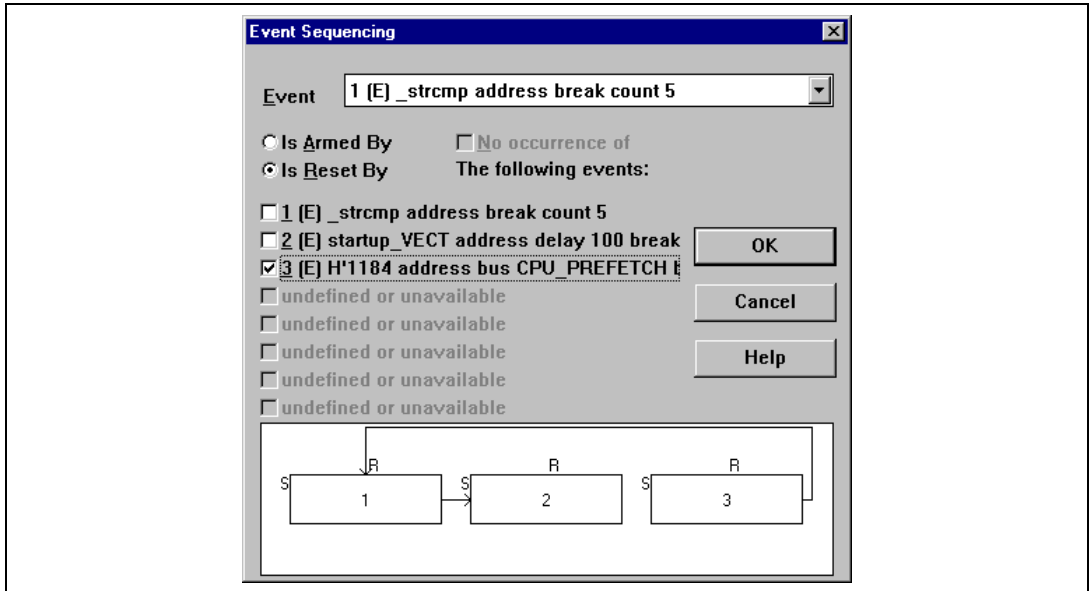


図 5-8 イベントシーケンス画面 (1)

このダイアログボックスによって、シーケンスの各イベントにおいて、イベントの前提条件となるイベント、またはリセットするイベントを1つ以上指定できます。

[Event] ドロップダウンリストボックスから、構成したいイベントを選択してください。これによって、[Enable Sequencing] で指定されたあらゆるイベントが選択できます。

現在選択されているイベントにおいて、[Is Armed By] をクリックして、イベントの前提条件となるイベントを設定してください。

同様に、[Is Reset By] をクリックして、イベントをリセットするイベントを設定してください。

5.3.6 イベントの前提条件

たとえば、4つのアドレスのシーケンスが発生した時にだけ起動されるイベントシーケンスを定義するには、以下のように設定します。

4 is armed by 3.

3 is armed by 2.

2 is armed by 1.

[Event Sequencing] ダイアログボックスは、定義したシーケンスを図式表示します。

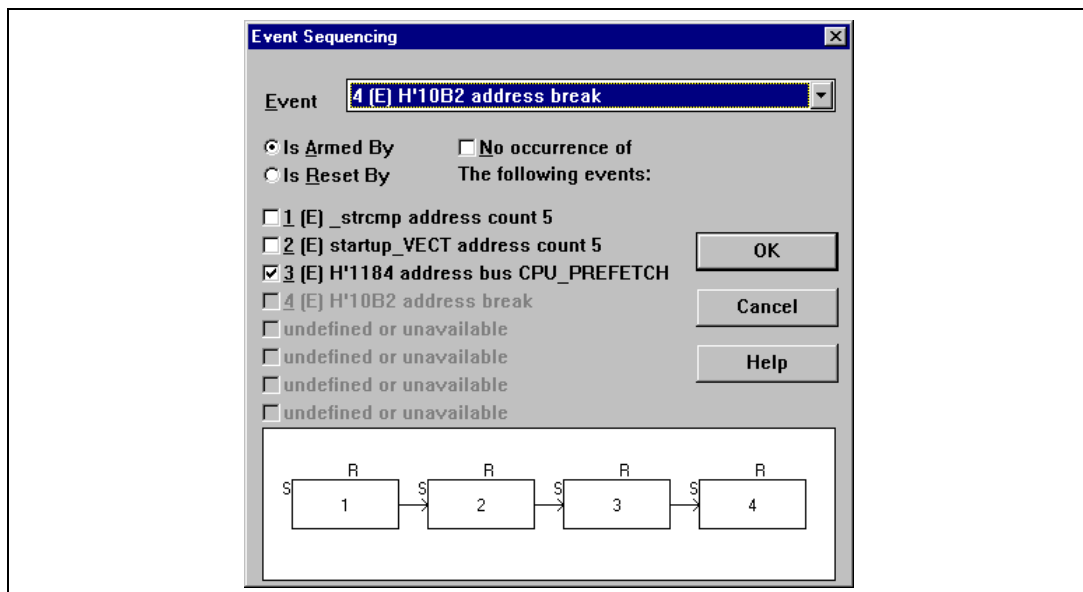


図 5-9 イベントシーケンス画面 (2)

シーケンスを定義するときは、シーケンスの最後のイベントだけを [Break] として定義してください。

5.3.7 イベントをリセットする

イベントが、シーケンスの中の別のイベントによってリセットされるように、指定することもできます。たとえば、イベント 2 の後にイベント 3 が続き、その後にイベント 4 が続いて、イベント 1 はその間には発生しない（もし、この間にイベント 1 が発生した場合は、前提条件のイベント発生がクリアされる場合）という条件でブレークが発生するようにするには、イベントシーケンスを以下のように設定してください。

- 4 is armed by 3 and reset by 1.
- 3 is armed by 2 and reset by 1.
- 2 is reset by 1.
- 1 is reset by 1.

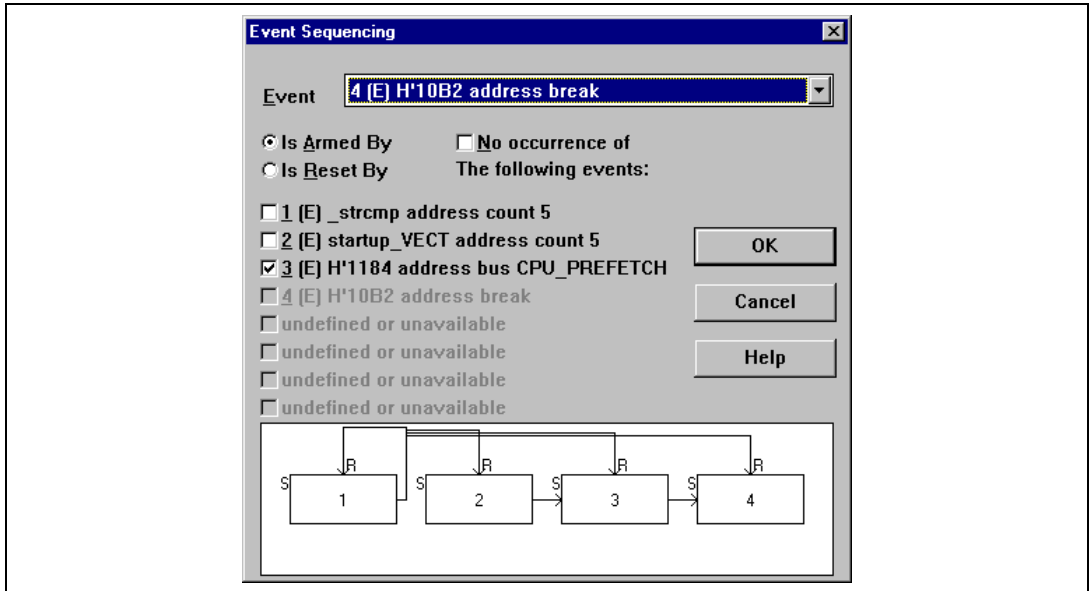


図 5-10 イベントシーケンス画面 (3)

5.4 メモリマッピングダイアログボックス

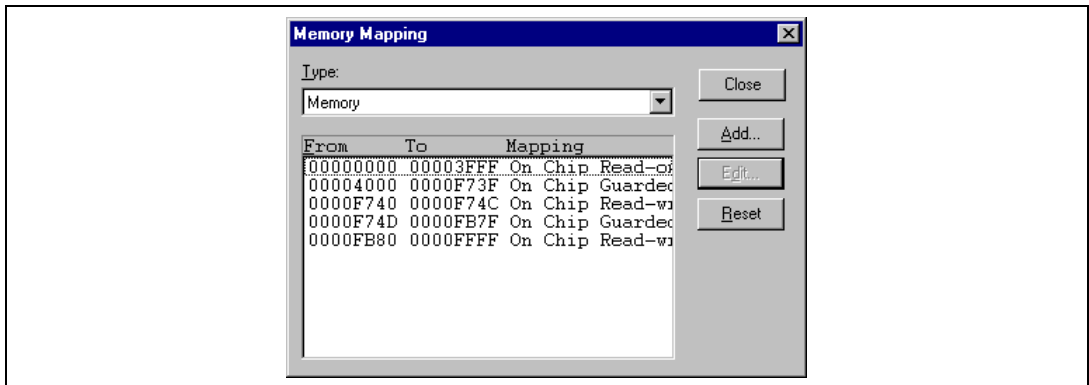


図 5-11 Memory Mapping 画面

メモリマッピングダイアログボックスは、エミュレーションメモリのマッピングの表示および編集ができます。

メモリマッピングウインドウを表示するには、[Memory] メニューから [Configure Map...] を選択してください。

メモリマッピングを編集するには、該当メモリブロックをダブルクリックしてください。あるいは、メモリマッピングリストの中から選択し、[Edit] をクリックしてください。以下のダイアログボックスがメモリブロックの現在の設定を示します。

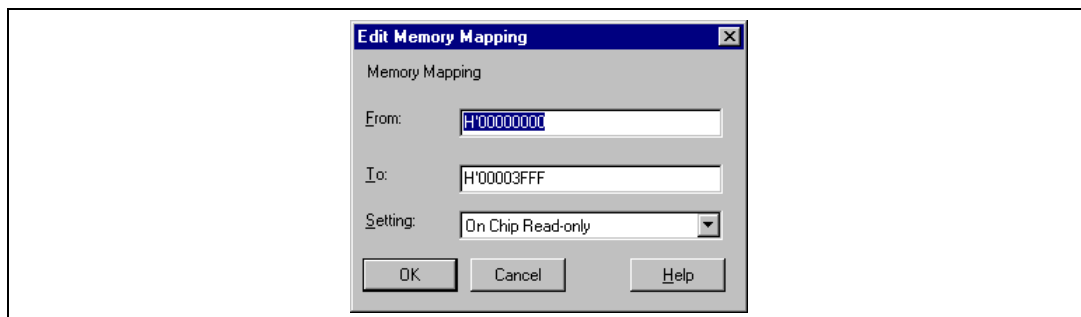


図 5-12 メモリマッピングの変更

[From] フィールドと [To] フィールドに、メモリブロックのアドレスの範囲を指定してください。そして、[Setting] ドロップダウンリストボックスで、メモリのタイプを選択してください。以下のオプションが選択できます。

表 5-5 メモリタイプの定義

メモリ	説明
オンチップ	MCU 内蔵メモリをアクセスします。
エミュレータ	エミュレーションメモリをアクセスします。

オプションにおいて、以下の 3 つのアクセスタイプから 1 つを指定できます。

表 5-6 アクセスタイプの定義

アクセスタイプ	説明
Read-Write	RAM
Read-Only	ROM
Guarded	アクセス不可

選択した MCU タイプとモードのデフォルトマッピングにメモリマッピングをリセットするには、[Reset] をクリックしてください。

5.5 トレースウインドウ

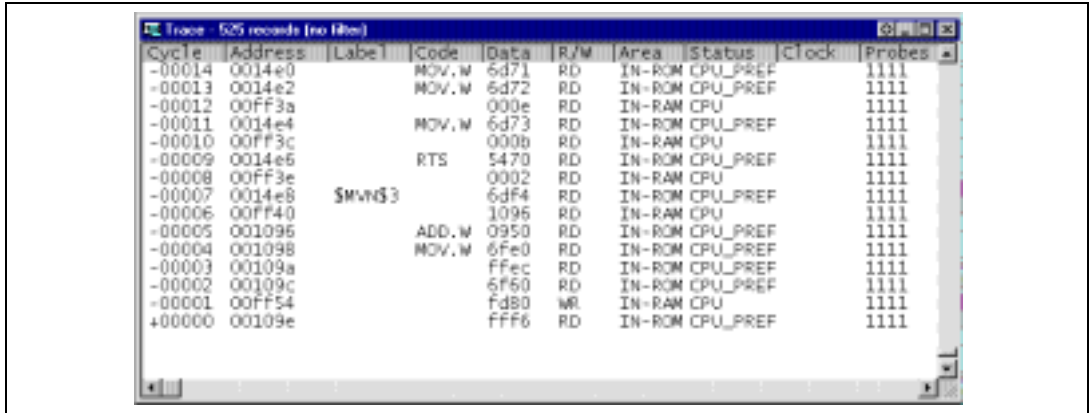


図 5-13 Trace Window 画面

トレースウインドウは、トレースバッファの内容を表示します。

トレースウインドウを表示するには、[View] メニューから [Trace] を選択してください。

トレースバッファの記憶データは、デバッグを容易にするためにソースプログラムおよびアセンブリ言語の両方で表示されます。ただし、トレースフィルタリングが使われた場合は、アセンブリ言語だけが表示されます。

本エミュレータでは、何も表示しません。

4本の外部プローブの値を Probes カラムに表示します。右から順に外部プローブ番号 1、2、3、4 となり、High を 1、Low を 0 と表示します。

[NMI] カラムは、本エミュレータでは、何も表示しません。

[Clear] をクリックすることによりトレースバッファを削除できます。また [Save] をクリックして、トレースバッファの内容をファイルに保存できます。

デフォルトでは、トレースバッファはあらゆる実行サイクルを収集し、最後の 32768 サイクルを保存します。また、指定されたサイクルだけを表示するフィルタ機能を設定できます。

5.5.1 [Filter]

フィルタを定義するには、トレースウインドウのポップアップメニューより [Filter...] を選択してください。

5.5.2 [Find]

トレースバッファの特定のトレースデータを検索するには、ポップアップメニューより [Find...] を選択してください。見つけたいトレースを指定するため、同じダイアログボックスが現われます。

5.5.3 [Cycle]

フィルタとして特定のサイクルを指定するには、[Type] を [Cycle] に設定し、[Cycle] ボックスにサイクル番号を入力してください。

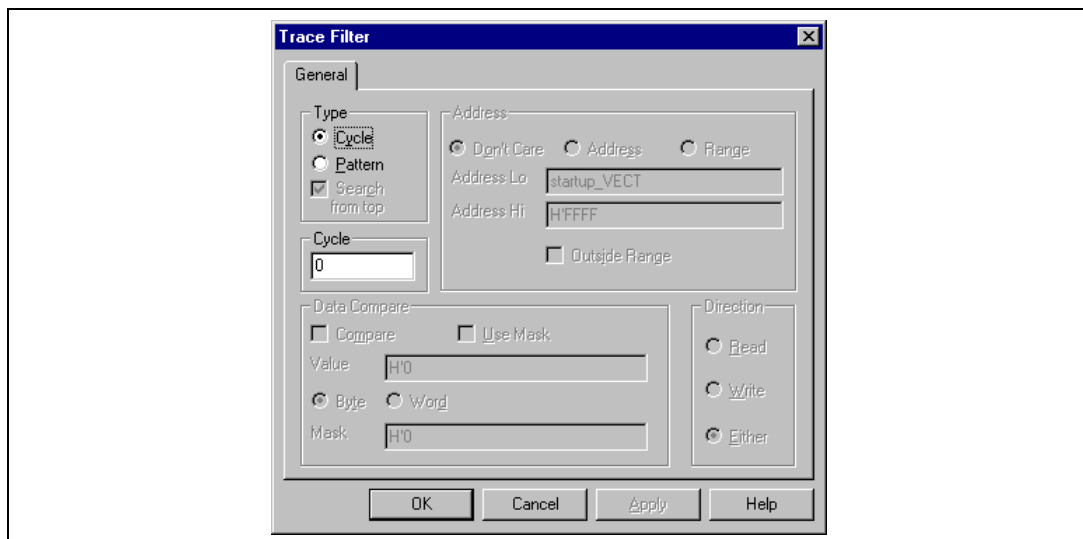


図 5-14 Trace Filter 画面 (General) (1)

5.5.4 [Pattern]

フィルタパターンを入力するには、[Type] を [Pattern] に設定し、必要な値を指定してください。

[Trace Filter] ダイアログボックスが 3 つのオプションパネルを示しますので、[General]、[Bus / Area]、および [Signals] のうちのどのサイクルを表示するかを指定できます。

5.5.5 [General]

[General] パネルによって、表示されるサイクルのアドレスとデータのアクセス条件を設定できます。

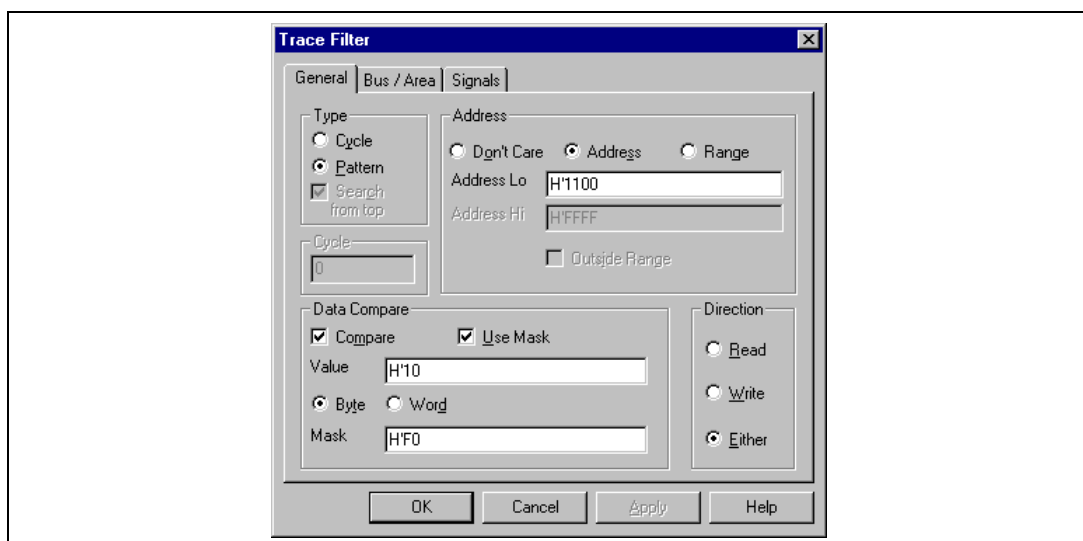


図 5-15 Trace Filter 画面 (General) (2)

5.5.6 [Bus / Area]

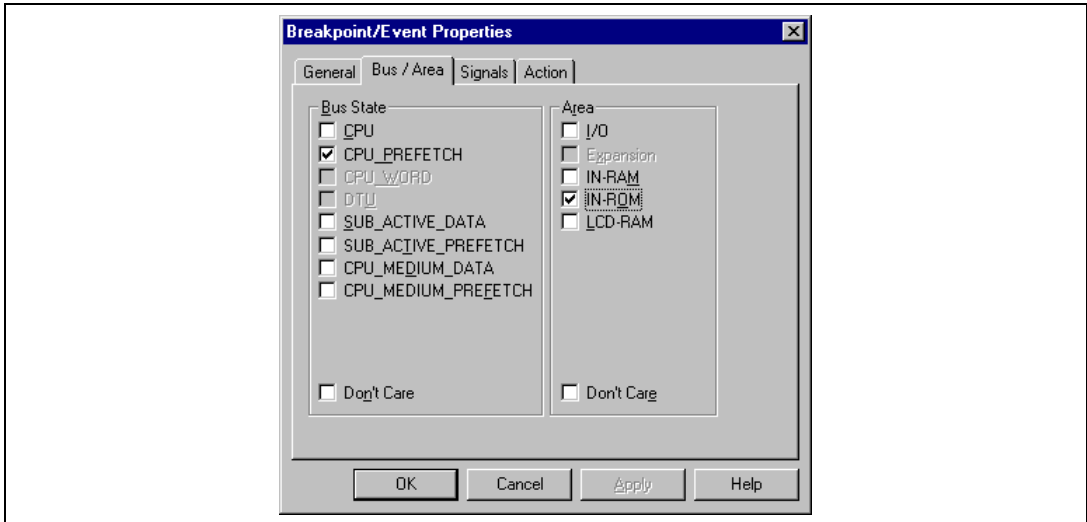


図 5-16 Trace Filter 画面 (Bus / Area)

5.5.7 [Signals]

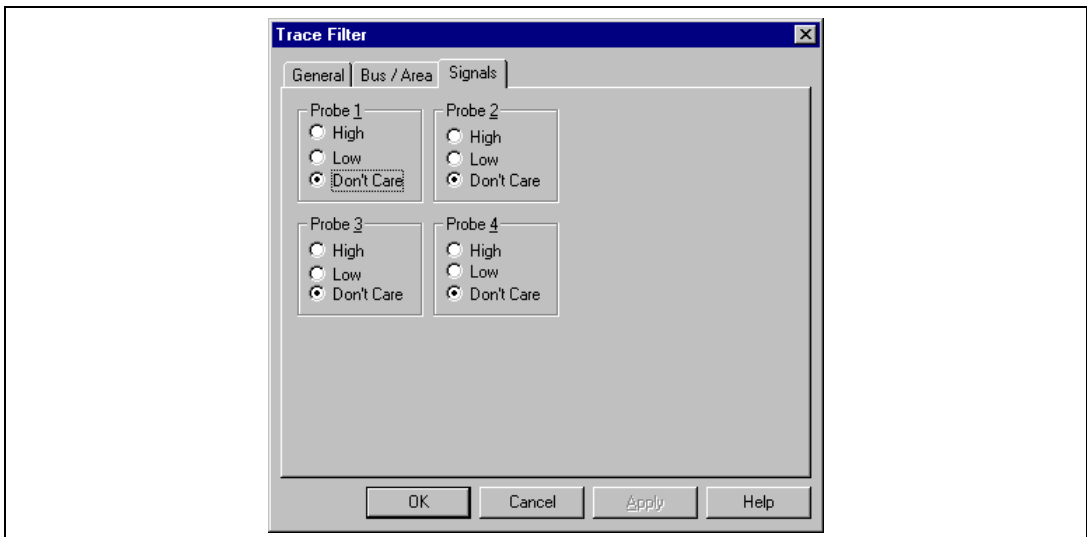


図 5-17 Trace Filter 画面 (Signals)

5.6 トレース制御

バッファは、すべてのバスサイクルあるいは選択されたサイクルだけを記憶するためにセットアップされます。トレース制御を指定するには、トレースウインドウ中の [Acquisition] をクリックしてください。

Trace Acquisition ダイアログボックスは以下のパネルを表示しますので、トレース制御条件を指定できます。

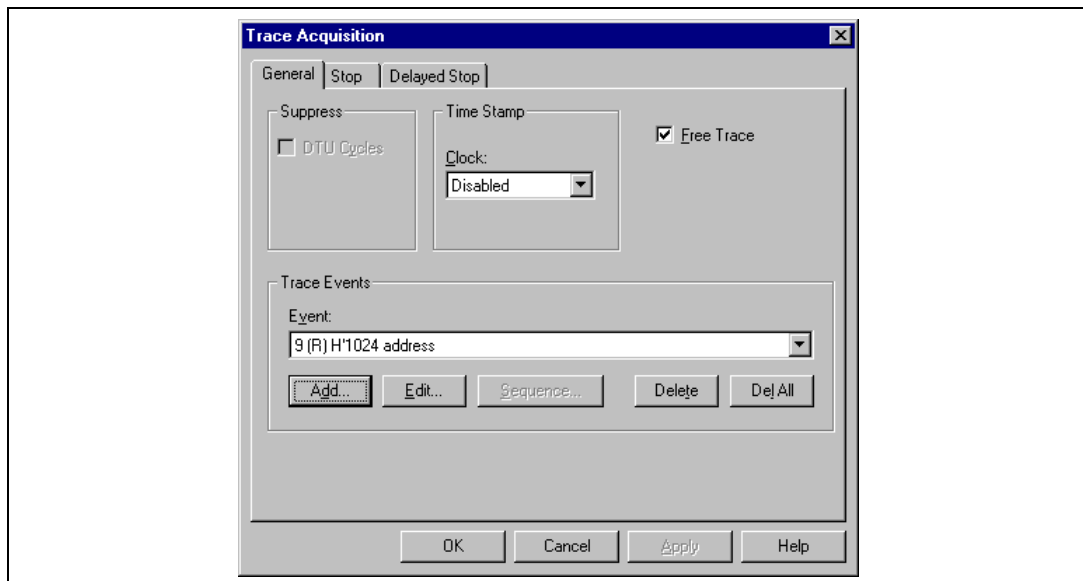


図 5-18 Trace Acquisition 画面 (General)

5.6.1 [General]

[Suppress] スペース中の [DTU Cycles] チェックボックスは、無効となります。

[Time Stamp] でクロックを指定することで、MCU の動作サイクルに対応した実行時間をトレースバッファに表示させることができます。ただし、実行時間表示を指定した場合は、以下の情報は表示されません。

- ・ Area, ・ Status, ・ Probes

また、乗除算命令の実行も正しく表示されません。

[Free Trace] チェックボックスをチェックして、すべてのトレース制御条件を禁止してください。これは、条件を削除することなくトレース制御を一時的に禁止します。[Free Trace] をチェックすると、トレースストップ、トレースディレイストップおよび [Suppress] セクションで指定された条件を除くすべてのバスサイクルを取得できます。

[General] パネルの [Trace Events] セクションによって、トレース制御のイベント、およびイベントシーケンスを定義できます。

イベントドロップダウンリストボックスが、現在設定されているすべてのイベントを表示します。

新しいイベントを加えるには、[Add...] をクリックし、[Breakpoint/Event Properties] ダイアログボックスにイベントの内容を入力してください。利用できるオプションの詳細な情報については、「5.3 イベント検出システム」を参照してください。

イベントを編集するには、[Event] リストの中のイベントを選択し、[Edit...] をクリックしてください。

イベントのシーケンスを設定するには、[Sequence...] をクリックしてください。このオプションは、一つ以上のイベントが設定され、[Enable Sequencing] が選択されているときのみ使えます。

イベントを削除するには、[Event] リストの中のイベントを選択し、[Delete] をクリックしてください。また、すべてのトレースイベントを削除するには、[Del All] をクリックしてください。

5.6.2 [Stop]

指定したイベントが発生した時にトレース取得を停止できます。

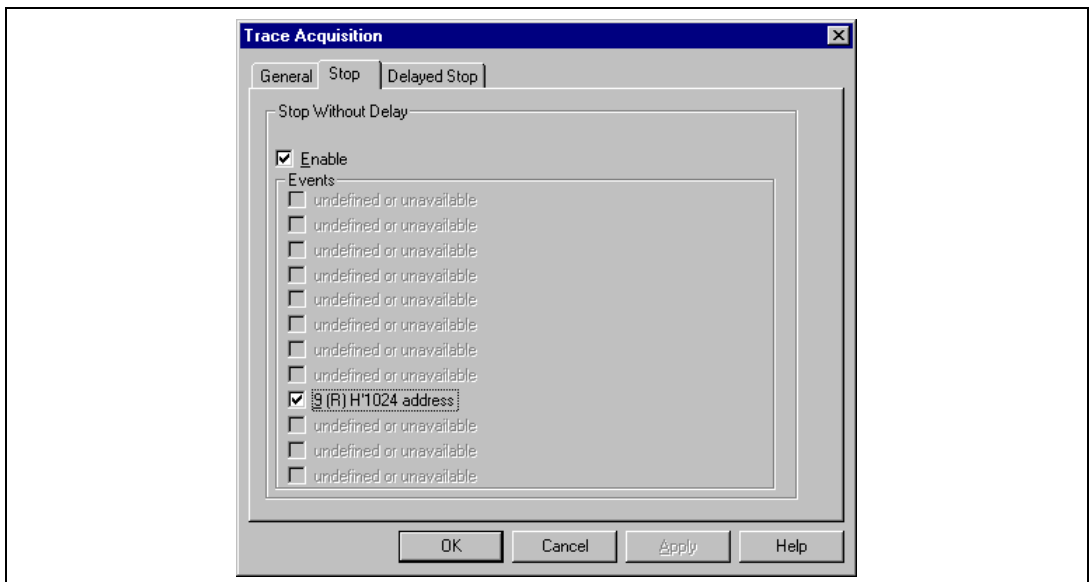


図 5-19 Trace Acquisition 画面 (Stop)

5.6.3 [Delayed stop]

指定したイベントが発生した後、指定されたサイクル数後にトレース制御を停止できます。

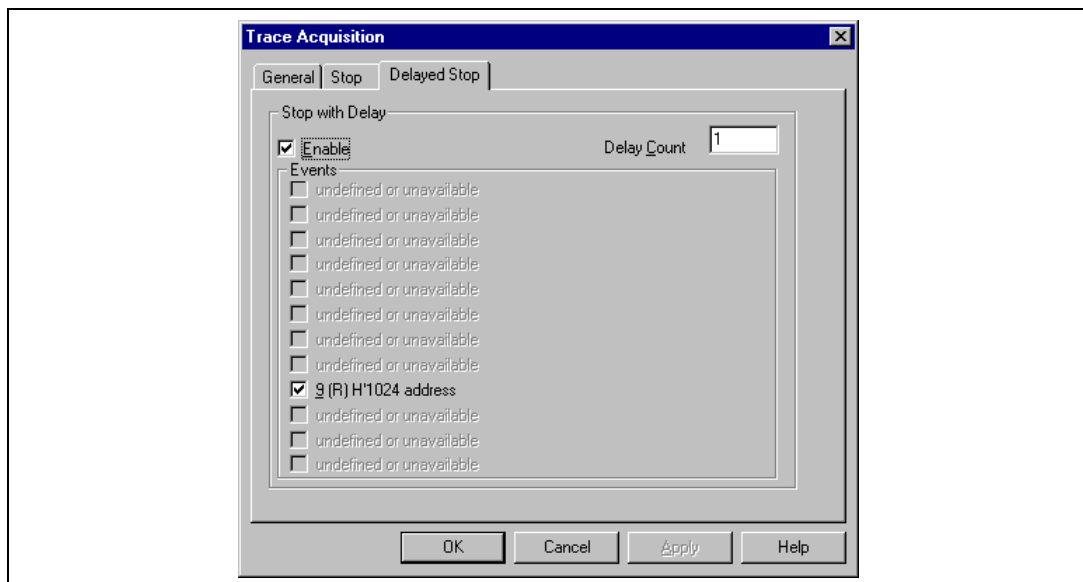


図 5-20 Trace Acquisition 画面 (Delayed Stop)

5.7 コマンドライン

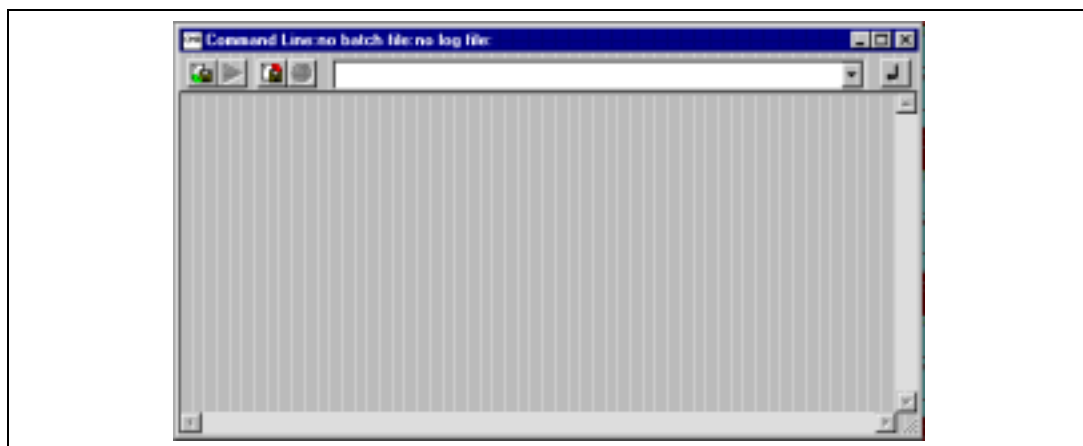


図 5-21 Command Line Window 画面

効率よくデバッグを行うために、コマンドラインウィンドウを使用してコマンドの入力と実行ができます。コマンドラインウィンドウを表示するには、[View]メニューから [Command Line] を選択してください。

MCU 専用のコマンドライン機能の詳細については、「6 コマンドライン機能」を参照してください。

6. コマンドライン機能

本章では、MCU 専用のコマンドライン機能について説明します。その他のコマンドライン機能については、「日立デバッグインタフェースユーザズマニュアル」を参照してください。以下にコマンドライン機能と HDI マニュアルおよび本マニュアルに記載する項目の対応表を示します。

表 6-1 HDI コマンドライン機能とマニュアルの対応表

コマンド名	短縮形	HDI マニュアル	本 マニュアル	説明
!				コメント
ACCESS	AC			不当アクセスに対する動作の設定
ANALYSIS_RANGE	AR			性能分析範囲の設定と表示
ANALYSIS_RANGE_DELETE	AD			性能分析範囲の解除
ANALYSIS	AN			性能分析機能の有効化 / 無効化
ASSEMBLE	AS			アセンブルの実行
ASSERT				コンディションのチェック
BREAKPOINT / EVENT	BP, EN		6.1	ブレークポイント / イベントの設定
BREAKPOINT_CLEAR	BC		6.2	ブレークポイント / イベントの解除
EVENT_CLEAR	EC			
BREAKPOINT_DISPLAY	BD		6.3	ブレークポイント / イベントの表示
EVENT_DISPLAY	ED			
BREAKPOINT_ENABLE	BE		6.4	ブレークポイント / イベントの有効化 / 無効化
EVENT_ENABLE	EE			
BREAKPOINT_SEQUENCE	BS		6.5	シーケンスの定義および解除
EVENT_SEQUENCE	ES			
CLOCK	CK		6.6	エミュレータの CPU クロック時間の設定
DEVICE_TYPE	DE		6.7	エミュレータのデバイスタイプの選択
DISASSEMBLE	DA			逆アセンブル表示
ERASE	ER			コマンドウィンドウの内容のクリア
EVALUATE	EV			式の計算
FILE_LOAD	FL			オブジェクト(プログラム)ファイルのロード
FILE_SAVE	FS			メモリ内容のファイルセーブ
FILE_VERIFY	FV			ファイル内容とメモリ内容の比較
GO	GO			ユーザープログラムの実行
GO_RESET	GR			リセットベクタからのユーザープログラムの実行
GO_TILL	GT			デンプラリブレークポイントまでのユーザープログラムの実行
HALT	HA			ユーザープログラムの停止
HELP	HE			コマンドラインまたはコマンドに対するヘルプ表示

6. コマンドライン機能

表 6-1 HDI コマンドライン機能とマニュアルの対応表 (つづき)

コマンド名	短縮形	HDI マニュアル	本 マニュアル	説明
INITIALISE	IN			プラットフォームの初期化
INTERRUPTS	IR			プラットフォームの割り込み処理の有効化 / 無効化 (E6000 エミュレータではサポートしません)
LOG	LO			ロギングファイルの操作
MAP_DISPLAY	MA			メモリマッピング情報の表示
MAP_SET	MS		6.8	メモリマッピングの設定
MEMORY_DISPLAY	MD			メモリ内容の表示
MEMORY_EDIT	ME			メモリ内容の変更
MEMORY_FILL	MF			指定データによるメモリ内容の一括変更
MEMORY_MOVE	MV			メモリブロックの移動
MEMORY_TEST	MT			メモリブロックのテスト
MODE	MO		6.9	CPU モードの設定と表示
QUIT	QU			HDI の終了
RADIX	RA			入カラディックスの設定
REFRESH	RF		6.17	メモリ関連ウィンドウの更新
REGISTER_DISPLAY	RD			CPU レジスタ値の表示
REGISTER_SET	RS			CPU レジスタ値の設定
RESET	RE			CPU のリセット
SLEEP				コマンド実行の遅延
STEP	ST			ステップ実行 (命令単位またはソース行単位)
STEP_OVER	SO			ステップオーバー実行
STEP_RATE	SR			複数ステップ実行
STEP_OUT	SP			PC 位置の関数を終了するまでのステップ実行
SUBMIT	SU			エミュレータコマンドファイルの実行
SYMBOL_ADD	SA			シンボルの追加
SYMBOL_CLEAR	SC			シンボルの削除
SYMBOL_LOAD	SL			シンボル情報ファイルのロード
SYMBOL_SAVE	SS			シンボル情報のファイルセーブ
SYMBOL_VIEW	SV			シンボルの表示
TEST_EMULATOR	TE		6.10	エミュレータハードウェアのテスト
TIMER	TI		6.11	実行時間測定タイム分解能の表示、設定
TRACE	TR			トレース情報の表示
TRACE_ACQUISITION	TA		6.12	トレース取得情報の設定と表示
TRACE_COMPARE	TC		6.13	トレース情報の比較
TRACE_SAVE	TV		6.14	トレース情報の保存
TRACE_SEARCH	TS		6.15	トレース情報の検索
USER_SIGNALS	US		6.16	ユーザーシグナル情報の有効化 / 無効化

6.1 BREAKPOINT / EVENT

省略形： BP, EN

ブレークポイントを設定します。ブレークポイントには以下の3種類があり、それぞれ別のフォーマットが用意されています。

- ・ プログラムブレークポイント
- ・ アクセスブレークポイント
- ・ 範囲ブレークポイント

6.1.1 プログラムブレークポイント

フォーマット : bp program address
 : bp p address

指定したアドレスにプログラムブレークポイントを設定します。

6.1.2 アクセスブレークポイント

フォーマット : bp access address [options]
 : bp a address [options]

オプション:

```
<options> = [<dataopts>] [read|write] [<signalopts>] [<busopts>] [<areaopts>] [<actionopts>]
[<count <countval>] [<delay <delayval>] [<channel <channelno>]
<dataopts> = data <data> [mask <mask>] [byte|word]
<signalopts> = signal ((1|2|3|4) (high|low))+
<busopts> = bus (cpu | cpupre | sadata | sapre| cpumdata | cpumpre)+
<areaopts> = area (io|iram|irom|lcdram)+
<actionopts> = action (trace | none | break((timer (start|stop))))+
<channelno> = 1..12
```

アクセスブレークポイントを設定すると、指定したアドレスを指定した方法で MCU がアクセスしたときに、ブレークが発生します。

6.1.3 範囲ブレークポイント

フォーマット: bp range [outside] <address low> <address hi> [<options>]

オプションは、アクセスブレークポイントのオプションと同じです。

範囲ブレークポイントを設定すると、指定したアドレス範囲または範囲外を MCU がアクセスしたときに、ブレークが発生します。

6.1.4 オプション

data <data> [mask <mask>] [byte | word]

データ比較を行いません。マスクを指定すると、マスクで0にセットされているビットに対応するデータビットは、比較されません。

例: data h'20 mask h'fff0 word

データバスの上位12ビットがh'002の場合、イベントが発生します。

このオプションを省略した場合、データ比較は行ないません。

signal ((1 | 2 | 3 | 4) (high | low)) +

指定した外部プローブ信号が指定した状態のとき、イベントが発生します。

例: signal 1 high 3 low

外部プローブ信号1がハイレベルでかつ外部信号3がローレベルのとき、イベントが発生します。(その他の信号レベルはチェックしません)

このオプションを省略した場合、信号レベルは無視されます。

bus (cpu | cpupre | sadata | sapre| cpumdata | cpumpre) +

6. コマンドライン機能

MCU のバスが指定した状態のとき、イベントが発生します。

表 6-2 MCU バスステータス

イベント	MCU のステータス
cpu	CPU がアクティブモードの時のデータアクセスサイクルを示します。
cpupre	CPU がアクティブモードの時の命令プリフェッチサイクルを示します。
sadata	CPU がサブアクティブモードの時のデータアクセスサイクルを示します。
sapre	CPU がサブアクティブモードの時の命令プリフェッチサイクルを示します。
cpumdata	CPU が中速アクティブモードの時のデータアクセスサイクルを示します。
cpumpre	CPU が中速アクティブモードの時の命令プリフェッチサイクルを示します。

例： bus cpu cpupre

バス状態が、プリフェッチまたはデータアクセスのとき、イベントが発生します。

このオプションを省略した場合、バス状態は無視されます。

area (io | iram | irom | lcdram)+

bus オプションと同様に、指定した領域をアクセスしたとき、イベントが発生します。

例： area irom iram

内蔵ROMまたは内蔵RAMアクセスのとき、イベントが発生します。

lcdram は MCU の LCDRAM 領域のアクセスを示します。

このオプションを省略した場合、アクセス領域は無視されます。

action (trace | none | break | (timer (start | stop)))+

イベントが検出されたときの動作を指定します。

デフォルトの動作は、ブレーク発生です。その他の動作として、イベント間実行時間測定の開始と停止を選択することができます。(タイマは1つしかありません)

count <countval>

イベント通過回数を設定します。(10進数)

delay <delayval>

イベントが検出されてから動作が始まるまでのディレイサイクルを、バスサイクル単位で指定します。(10進数)

channel 1..12

イベント検出システムチャンネル番号を指定します。イベント発生の順序を指定する場合、このチャンネル番号を参照して指定します。(詳細は、「6.5 event_sequence」を参照してください。)チャンネル1~8はイベント検出用、チャンネル9~12は範囲検出用チャンネルです。

使用例

en access 100	アクセスブレークポイントをアドレス 100 に設定
bp p 110	プログラムブレークポイントをアドレス 110 に設定
en access 100 data 55 byte	アクセスブレークポイントをアドレス 100、データ 55 に設定
bp range 12 45	範囲ブレークポイントをアドレス 12 からアドレス 45 の範囲に設定
bp range outside 60 89	範囲ブレークポイントを、アドレス 60 からアドレス 89 以外の範囲がアクセスされたときにブレークするように設定
bp a 200 read	アクセスブレークポイントをアドレス 200 のリードサイクルに設定
bp a 500 write	アクセスブレークポイントをアドレス 500 のライトサイクルに設定
bp a 100 read channel 8	アクセスブレークポイントをアドレス 100 のリードサイク

ルに設定。チャンネル8を指定しているので、イベントが発生した時に外部プローブからトリガ信号が出力されます。

6.2 BREAKPOINT_CLEAR / EVENT_CLEAR

省略形： BC, EC

ユーザが設定したブレークポイントを削除します。

表 6-3 ブレークポイント削除の指定

種類 (キーワード)	説明 (ブレークポイントの種類)
program <address>	指定したプログラムブレークポイントの削除
access <address> <options>	指定したアクセスブレークポイントの削除
range <address> <options>	指定した範囲ブレークポイントの削除
all	全ブレークポイントの削除
all trace	全トレースイベントの削除
channel 1 ~ 12	指定したチャンネル番号のイベントの削除

オプションは、BREAKPOINT / EVENT コマンドと同様です。イベントを特定するのに必要最小限のオプションを指定してください。

使用例

bc p 256	アドレス 256 のプログラムブレークポイントを削除
event_clear chan 5	チャンネル番号を指定してイベントを削除
bc all	全ブレークポイントを削除

6.3 BREAKPOINT_DISPLAY / EVENT_DISPLAY

省略形： BD, ED

現在設定されているブレークポイントと、その有効 / 無効を表示します。トレースイベントは “ trace ” と表示します。

使用例

bd	設定されている全ブレークポイントと、その有効 / 無効状態を表示します。
----	--------------------------------------

6.4 BREAKPOINT_ENABLE / EVENT_ENABLE

省略形： BE, EE

指定したブレークポイントまたは全ブレークポイントを有効 / 無効にします。

表 6-4 ブレークポイント有効/無効

パラメータ	種類 (キーワード)	説明
1	true false	ブレークポイントを有効にします。 ブレークポイントを無効にします。
2	all program <address> access <address> <options> range <address1> <address 2> <options> channel 1..12	全ブレークポイント プログラムブレークポイント アクセスブレークポイント 範囲ブレークポイント 指定したチャンネル番号のイベントを有効または無効にします。

BREAKPOINT / EVENT コマンドで設定するのと同様に、オプションを指定して、イベントを正確に特定することができます。

使用例

be true all	全ブレークポイントを有効
be false all	全ブレークポイントを無効
be false p 256	アドレス 256 のプログラムブレークポイントを無効
be true access 12	アドレス 12 のアクセスブレークポイントを有効
be false chan 1	チャンネル 1 のイベント検出を無効

6.5 BREAKPOINT_SEQUENCE / EVENT_SEQUENCE

省略形：BS, ES

フォーマット

```
bs <channel> [armed_by [not] <chan1> <chan2> ...]
                    [armed_by off]
                    [reset_by <chan1> <chan2> ...]
                    [reset_by off]
```

別のイベントの前提条件となる event またはリセット条件となるイベントを指定します。

使用例

```
bs 1 armed_by 2 3
```

イベント 2 または 3 は、イベント 1 発生的前提条件となります。設定できる番号は、イベント検出用のチャンネル 1~8 です。チャンネル番号は、event コマンドのチャンネルオプションで、イベントに割り当てられます。

```
bs 2 reset_by 4
```

イベント 2 は、イベント 4 が発生したときにリセットされます。

キーワード Off は、他のイベントによる前提/リセット条件を無効にします。これにより、各イベントは、独立して検出・発生します。

6.6 CLOCK

省略形：CK

システムクロック()およびサブクロック(w)の指定、表示を行いません。パラメータを省略すると、現在のシステムクロックを表示します。システムクロックを変更すると、E6000 エミュレータはリセットされます。MCU のシステムクロック()は、(OSC1,OSC2) 入力、1/2 の周波数に

なります。

表 6-5 クロックパラメータ

パラメータ	種類 (キーワード)	説明 (エミュレーションクロック)
1	05 2 8 t2	0.5 MHz 内部クロック 2MHz 内部クロック 8MHz 内部クロック ターゲット/2
2	sub 32k sub 38k sub 307k sub t	32.768kHz 内部サブクロック(w) 38.4kHz 内部サブクロック(w) 307.2kHz 内部サブクロック(w) ターゲットサブクロック

ユーザシステムのクロックは、ユーザシステムの Vcc が供給されているときだけ選択可能です。

ck 現在のエミュレーションクロックを表示
ck 2 sub 32k システムクロックとして 2MHz 内部クロック、サブクロックとして 32.768kHz 内部クロックを指定

本エミュレータでサブクロック(32.768kHz および 38.4kHz)動作中にブレークが検出された場合、そのクロックにて動作、表示反応してしまうため、サブクロック評価時は、38.4kHz の 8 倍周波数 307.2kHz を選択していただき評価することをお勧めします。

- 【注】
1. ターゲットシステムクロックはユーザシステムの Vcc が供給されている場合にのみ選択可能です。
 2. ターゲット MCU (H8/3802 シリーズ) は、内部サブクロック(w)として、307.2kHz を選択することはできません。

6.7 DEVICE_TYPE

省略形：DE

ターゲット MCU を指定、または現在の設定を表示します。

使用例

de MCU タイプの表示
de h8/3802 H8/3802 を指定

6.8 MAP_SET

省略形：MS

エミュレーションメモリマップを設定します。

フォーマット

ms <start> <end> (internal | on-chip) (none | read-only | guarded)

使用例

ms 8000 F73F internal アドレス H'8000 から H'F73F に内部のアクセス可能メモリを割り付けます

【注】 on-chip メモリ (例 内蔵 RAM、内蔵 ROM、I/O またはリザーブエリア) は、デバイス内

6. コマンドライン機能

蔵のメモリです。基本的にはメモリマップの属性変更はできません。
ただし、H'EE00 ~ H'F73F, H'F760 ~ H'F77F, H'FF80 ~ H'FF8F を除くリザーブエリアのみ
Internal を指定することでエミュレーションメモリの設定が可能です。

6.9 MODE

省略形：MO

MCU モードの設定、表示を行ないます。

表 6-6 MCU モードのパラメータ

パラメータ	種類 (キーワード)	説明 (モード)
1	3	3 (シングルチップモード)

MCU ではモード 3 に固定です。

使用例

mode

現在のモードを表示

mode 3

モード 3 に設定、メモリマップを再設定

6.10 TEST_EMULATOR

省略形：TE

E6000 エミュレータのハードウェアとエミュレーションメモリ領域のテストを行ないます。本コマンド実行後は、E6000 エミュレータシステムを初期化してください。

使用例

te

E6000 エミュレータのテスト

6.11 TIMER

省略形：TI

実行時間測定タイマの分解能の表示、変更を行ないます。これにより、実行時間測定およびイベント間実行時間測定のタイマの分解能が設定できます。

表 6-7 タイマコマンド

コマンド	説明
ti	タイマ分解能の表示
ti <timer resolution>	タイマ分解能の設定

タイマ分解能：20ns、125ns、250ns、500ns、1 μ s、2 μ s、4 μ s、8 μ s、16 μ s

使用例

ti 20

タイマ分解能を 20ns に設定

ti 250ns

タイマ分解能を 250ns に設定

ti 8

タイマ分解能を 8 μ s に設定

ti 16us

タイマ分解能を 16 μ s に設定

6.12 TRACE_ACQUISITION

省略形： TA

トレース取得オプションの設定、表示を行いません。

フォーマット

```
TA [<suppress>] [<freetrace>] [<timestamp>] [<stop>] [<stopdelay>]
  [<range>][<default>]
```

```
<suppress> = suppress dtu (true|false)      ( MCU では使用できません。 )
<freetrace> = freetrace (true|false)
<timestamp> = timestamp (disable | 125ns | 250ns | 500ns | 1us | 2us | 4us | 8us | 16us | 100us )
<stop> = stop ( disable | event <1 ~ 12> )
<stopdelay> = stopdelay ( disable | event <1 ~ 12> [count <count>] )
<range> = range <1 ~ 4> ( disable |
  ptop <startaddr> <stopaddr> [cyclic] |
  range <1 ~ 12> |
  event <1 ~ 8> <1 ~ 8> [cyclic] )
<default> = default
```

使用例

```
ta                               全トレース制御オプションを表示
ta stop event 1 2                チャンネル 1 もしくは 2 のどちらかのイベントが発生したとき
                                にトレース取得を止める
ta stopdelay event 1 2          チャンネル 1 もしくは 2 のどちらかのイベントが発生したとき
count 100                       に 100 バスサイクルトレース取得後にトレース取得を止める
ta timestamp 500ns              トレースのタイムスタンプ機能を有効にし、タイムスタンプ
                                機能の分解能を 500nS に設定
ta range 2 event 4 5 cyclic      イベント 4 でトレース開始し、イベント 5 でトレース停止す
                                る条件をトレース制御 2 に登録する、また、サイクル指定が
                                あるので条件が成立する毎にトレースを取得する
```

6.13 TRACE_COMPARE

省略形： TC

[trace_save] で保存したトレースファイルと現在のトレース結果を比較します。

```
trace_compare <filename>
```

6.14 TRACE_SAVE

省略形： TV

トレースした情報を、バイナリ形式のファイルに保存します。保存したデータは、別のトレース結果と [trace_compare] コマンドで比較することができます。

```
trace_save <filename>
```

6.15 TRACE_SEARCH

省略形： TS

トレース結果を検索します。 [trace find] ダイアログボックスと同じ方法でトレース結果を検索できます。

フォーマット

```
TS [<address>][<dataopts>] [<signalopts>] [<busopts>] [<areaopts>]
  [<directionopts>] [<timestampopts>] [<fromopts>]
```

```
<address>      = address <address> [to <address>]
<dataopts>    = data <data> [mask <mask>] [byte|word]
<signalopts>  = signal <sig><sig><sig><sig>
  <sig>        = (1|0|x)                1 = high, 0 = low, x = don't care
<busopts>    = bus (cpu | cpupre | sadata | sapre| cpumdata| cpumpre )+
<areaopts>    = area ( io | iram | irom | lcdram )+
<directionopts> = dir (read | write | either)
<timestampopts> = time <start> [ to <stop>]
  <start> and <stop> should be in format 0s:0000.000
<fromopts>    = from <record>
```

使用例

```
ts address 104 data 55aa w      アドレスが 104 番地、データが 55aa のワードアクセスのトレースサイクルを検索する
ts area irom                   ROM エリアアクセスのトレースサイクルを検索する
```

6.16 USER_SIGNALS

省略形： US

ユーザ信号 (Reset) の入力を有効 / 無効にします。パラメータを省略すると、Reset 信号の有効 / 無効状態を表示します。

表 6-8 ユーザ信号コマンド

種類	説明
us	ユーザ信号状態を表示
us enable reset	指定された信号を有効
us disable reset	指定された信号を無効

6.17 REFRESH

省略形： RF

メモリ関連ウィンドウを更新します。

7. 故障解析

本章では、E6000 エミュレータ用テストプログラムによる故障解析の手順について示します。

7.1 テストプログラムを実行するためのシステムセットアップ

- (1) テスト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ケーブルおよびユーザシステムを接続しないでください。
 - E6000 エミュレータ(HS3800EPI60H)
 - PC
 - E6000 PC インタフェースボード (本マニュアルでは、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枚をご用意ください。)
 - ISAバスインタフェースボード (HS6000EII01H)
 - PCIバスインタフェースボード (HS6000EIC01HまたはHS6000EIC02H)
 - PCMCIAインタフェースカード (HS6000EIP01H)
- (2) 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。
- (3) PCインタフェースケーブルをE6000エミュレータ本体に接続してください。
- (4) E6000エミュレータ本体に、付属のACアダプタを接続してください。
- (5) PCを起動し、DOSプロンプトのコマンド入力待ち状態にしてください。
- (6) E6000エミュレータ本体の電源をオンにしてください。

7.2 テストプログラムによる故障解析

E6000 エミュレータに添付されているテストプログラム用フロッピーディスク(HS3800EVI60SF)をPCに挿入し、カレントディレクトリをA: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。

- (1) ISAバスインタフェースボード (HS6000EII01H)
>A:TM3800 -ISA (RET)
- (2) PCIバスインタフェースボード (HS6000EIC01HまたはHS6000EIC02H)
>A:TM3800 -PCI (RET)
- (3) PCMCIAインタフェースカード (HS6000EIP01H)
>A:TM3800 -PCCD (RET)

カレントディレクトリを A: に移動しない状態で C:¥>A:TM3800 -ISA (RET) のように他のカレントディレクトリからテストプログラムを起動した場合はテストプログラムが正しく動作しません。必ず A:にカレントディレクトリを移動してテストプログラムを実行してください。

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 (PC 166MHz, Windows®95 PCMCIA インタフェースカード使用時にテスト時間は約 6 分です)。

7. 故障解析

E6000 H8/3800 EMULATION BOARD Tests Vx.x
Hitachi Ltd (2000)
SIMM module fitted? (1. None 2. 1MB 3. 4MB): 1

Searching for interface cardOK

Checking emulator is connectedOK

Emulator Board Information:

Main Board ID	H'1
Emulation Board ID	H' 15
SIMM	No SIMM module inserted

01) Testing Main Board Register :

IDR0 Register OK
PAGE Register OK
CES G/A Register OK
IDR1 Register OK

02) Testing Dual-Port RAM :

Decode Test OK
Marching Test OK

03) Testing Firmware RAM :

Decode Test. page range H'700 - H'71f OK
Marching Test. page range H'700 - H'71f OK

04) Testing Trace RAM :

Decode Test. page range H'000 - H'04f OK
Marching Test. page range H'000 - H'04f OK

05) Testing Mapping RAM :

Decode Test. page range H'200 - H'27f OK
Marching Test. page range H'200 - H'27f OK

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。

SIMM メモリを使用しないため"1"を入力
してください。

ホスト PC に PC 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。
また、ISA バスインタフェースボードを接
続している場合は、アドレスを表示しま
す。値はアドレス設定値によって変わ
ります (PCI, PCMCIA ではアドレスは表示
しません)

ホスト PC と E6000 エミュレータが正しく
接続されていることを示します。

E6000 エミュレータ(下基板) の ID 番号
で、常に 1 を示します。

E6000 エミュレータ(上基板)の ID 番号で、
常に 15 を示します。

オプションのメモリボードの有無を示
します。

E6000 エミュレータ上のレジスタのチェッ
ク結果(正常終了)を示します。

E6000 エミュレータ上の Dual-Port RAM の
デコードテスト、マーチングテストチェ
ック結果 (正常終了) を示します。

E6000 エミュレータ上の Firm RAM のデコ
ードテストチェック結果(正常終了)を示
します。

E6000 エミュレータ上の Firm RAM のマー
チングテストチェック結果(正常終了)を示
します。

E6000 エミュレータ上の Trace RAM のデ
コードテストチェック結果(正常終了)を示
します。

E6000 エミュレータ上の Trace RAM のマ
ーチングテストチェック結果(正常終了)を
示します。

E6000 エミュレータ上の Mapping RAM の
デコードテストチェック結果(正常終了)を
示します。

E6000 エミュレータ上の Mapping RAM の
マーチングテストチェック結果(正常終了)
を示します。

- 06) Testing Internal ROM and RAM :
- Setting up, please wait..
 - Decode Test [0x0000 - 0xff7f] OK
 - Marching Test [0x0000 - 0xff7f] OK
- 07) Testing STEP Operation :
- Setting up, please wait..
 - Step Operation OK
- 08) Testing Key Break :
- Setting up, please wait..
 - Key Break OK
- 09) Testing Emulation RAM Hardware Break :
- Setting up, please wait..
 - GRD Break OK
 - WPT Break OK
- 10) Testing Internal ROM Write-Protect :
- Setting up, please wait..
 - Write-Protect OK
- 11) Testing Hardware Break :
- Setting up, please wait..
 - A)Break Point Initialised OK
 - B)Event Detectors CES channel 1-12 OK
 - C)Test Sequencing 1 OK
 - D)Check Range Break OK
 - E)Range Break Test for Data OK
- 12) Testing Emulation RAM Trace :
- Setting up, please wait..
 - A)Free Trace Test OK
 - B)Range Trace Test OK
 - C)Point to Point Trace Test OK
 - D)Start and Stop Event Trace Test OK
 - Setting up, please wait..
 - E)Time STAMP Trace Test
 - Time STAMP Trace Test 1 OK
 - Time STAMP Trace Test 2 OK
 - Time STAMP Trace Test 3 OK
- 13) Testing Runtime counter :
- Setting up, please wait..
 - Testing Internal Clock = 8.00MHz OK
 - Testing Internal Clock = 2.00MHz OK
 - Testing Internal Clock = 0.5MHz OK
 - Testing Internal SubClock = 32.768kHz OK

内蔵 ROM および RAM のデコードテスト、マーチングテストチェック結果(正常終了)を示します。

ステップ実行制御回路のチェック結果(正常終了)を示します。

強制ブレーク制御回路のチェック結果(正常終了)を示します。

不当アクセスブレーク制御回路のチェック結果(正常終了)を示します。

内蔵 ROM 領域に対する書き込み禁止制御回路のチェック結果(正常終了)を示します。

ハードウェアブレーク制御回路のチェック結果(正常終了)を示します。

トレース制御回路のチェック結果(正常終了)を示します。

実行時間測定カウンタのチェック結果(正常終了)を示します。

7. 故障解析

14) Testing Emulation Monitor :

Setting up, please wait..

A)A15-A0 (MONIT1OE:D7-D0) TEST. OK

B)ST2 to ST0 (MONIT2E:D6-D4,
MONIT0E:D2-D0) OK

C)BRKACK (MONIT0E:D4) TEST..... OK

D)CNN (MONIT2E:D1) TEST OK

E)IF (MONIT2E:D7) TEST OK

F)WINDOW (MONIT2O:D1) TEST OK

Tests run for xH:xM:xS

0 total errors

Tests passed, emulator functioning correctly

エミュレーションモニタ制御回路のチェック結果(正常終了)を示します。

テスト時間を示します。

エラー発生数の合計を示します。

テストプログラムにより正常動作が確認されたことを示します。

本テストプログラムは不具合を検出すると ERROR を表示してプログラムの実行を中止します。この場合、エミュレータハードウェアの故障が考えられます。発生したエラー内容の詳細を当社の購入営業担当までご連絡ください。

7.3 エラー発生時の処理

E6000 エミュレータをご使用中に動作エラーが発生した場合は、お手数ですが下記故障症状調査書に症状をご記入のうえ、担当営業まで FAX でご連絡いただく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 _____
 ご担当者名 _____ 様
 TEL _____
 FAX _____

1) 不具合発生製品型名およびシステム構成

- a) E6000エミュレータ(HS3800EPI60H) 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)
- b) PCインタフェースボード
 型式 HS _____ H、 シリアルNo. _____、 レビジョン _____
 (基板上に捺印表示しています)
- c) ユーザシステムインタフェースケーブル
 型式 HS _____ H、 シリアルNo. _____、 レビジョン _____
 (基板上に捺印表示しています)
- d) HDI(HS3800EPI60SF) バージョンV _____
 (FDにVx.xxと表示しています)
- e) ご使用になっているPC
 メーカー名 _____、 型式 _____
 使用OS (Windows®95, Windows®98, WindowsNT® いずれかに)

2) ターゲットシステムの使用条件

- a) デバッグ対象マイコン型名： H8/ _____
- b) ターゲットシステム電圧： _____ V
- c) 使用クロック： (貸出しクロック、外部クロック入力 いずれかに)
- d) 動作周波数： _____ MHz

3) エラー発生状況

- a~cのいずれかに をつけ、内容を記載してください。
- a) HDIがLink upしない
 (エラーメッセージ： _____)
- b) テストプログラムでエラーが発生
 (テスト番号： _____ エラーメッセージ： _____)
- c) デバッグ中にエラーが発生
 下記の各項目についてエラー内容を記載してください。

7. 故障解析

- 4) メモリのデータ化けは発生していますか？ (はい、いいえ いずれかに)
- a) データ化けを起こしている箇所のMemory Mapping設定
(ROM, RAM, I/O, Emulator, その他 _____ いずれかに)
 - b) データ化けを起こしている箇所はMemory Windowでリード/ライトできますか？
(はい、いいえ いずれかに)
- 5) ターゲットシステムに対して入出力できない特定の信号はありますか？
(はい、いいえ いずれかに)
- a) 信号名： _____、ピン番号： _____
 - b) 信号レベル異常： (High固定、Low固定、中間レベル いずれかに)
- 6) 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。

H8/3802 シリーズ用 E6000 エミュレータ ユーザズマニュアル



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

ADJ-702-290