

お客様各位

---

## カタログ等資料中の旧社名の扱いについて

---

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

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

2010年4月1日  
ルネサスエレクトロニクス株式会社

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

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

## ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。  
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット  
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）  
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、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。

ユーザース・マニュアル

RENESAS

ID850QB Ver.3.20

統合デバッガ

操作編

---

対象デバイス

V850シリーズ

資料番号 U17964JJ1V0UM00 (第1版)

発行年月 March 2006 CP(K)

© NEC Electronics Corporation 2006

[メモ]

## 目次要約

第1章	概要	...	18
第2章	インストレーション	...	27
第3章	起動と終了	...	28
第4章	PM+との連携	...	36
第5章	デバッグ機能	...	42
第6章	ウインドウ・リファレンス	...	105
第7章	コマンド・リファレンス	...	287
付録A	拡張ウインドウ	...	337
付録B	入力規約	...	349
付録C	キー機能一覧	...	356
付録D	メッセージ	...	359
付録E	索引	...	387

Windows , Windows XP は米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

- 本資料に記載されている内容は2006年3月現在のものです、今後、予告なく変更することがあります。量産設計の際には最新の個別データ・シート等をご参照ください。
- 文書による当社の事前の承諾なしに本資料の転載複製を禁じます。当社は、本資料の誤りに関し、一切その責を負いません。
- 当社は、本資料に記載された当社製品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、一切その責を負いません。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。
- 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、一切その責を負いません。
- 当社は、当社製品の品質、信頼性の向上に努めておりますが、当社製品の不具合が完全に発生しないことを保証するものではありません。当社製品の不具合により生じた生命、身体および財産に対する損害の危険を最小限度にするために、冗長設計、延焼対策設計、誤動作防止設計等安全設計を行ってください。
- 当社は、当社製品の品質水準を「標準水準」、「特別水準」およびお客様に品質保証プログラムを指定していただく「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。

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

特別水準：輸送機器（自動車、電車、船舶等）、交通信号機器、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

特定水準：航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器、生命維持のための装置またはシステム等

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。意図されていない用途で当社製品の使用をお客様が希望する場合には、事前に当社販売窓口までお問い合わせください。

(注)

- (1) 本事項において使用されている「当社」とは、NECエレクトロニクス株式会社およびNECエレクトロニクス株式会社がその総株主の議決権の過半数を直接または間接に保有する会社をいう。
- (2) 本事項において使用されている「当社製品」とは、(1)において定義された当社の開発、製造製品をいう。

〔メモ〕

# はじめに

**対象者** このマニュアルは V850 シリーズの各製品の応用システムを設計、開発するユーザを対象としています。

**目的** このマニュアルは、次の構成に示す ID850QB の機能をユーザに理解していただくことを目的としています。

**構成** このマニュアルは、大きく分けて次の内容で構成しています。

- ・概要
- ・インストレーション
- ・起動と終了
- ・PM+との連携
- ・デバッグ機能
- ・ウィンドウ・リファレンス
- ・コマンド・リファレンス

**読み方** このマニュアルの読者には、電気、論理回路、マイクロコンピュータ、C 言語、アセンブラに関する一般知識を必要とします。

V850 シリーズのハードウェア機能を知りたいとき

各製品の**ユーザズ・マニュアル ハードウェア編**を参照してください。

V850 シリーズの命令機能を知りたいとき

V850ES **ユーザズ・マニュアル アーキテクチャ編** (U15943J) または

V850E1 **ユーザズ・マニュアル アーキテクチャ編** (U14559J) を参照してください。

**凡 例**

データ表記の重み	: 左が上位桁, 右が下位桁
注	: 本文中につけた注の説明
注意	: 気をつけて読んでいただきたい内容
備考	: 本文の補足説明
数の表記	: 2 進数 ...XXXX または XXXXB 10 進数...XXXX 16 進数...XXXXH

2 のべき数を示す接頭語 (アドレス空間, メモリ容量) :

K (キロ)	: $2^{10} = 1024$
M (メガ)	: $2^{20} = 1024^2$
G (ギガ)	: $2^{30} = 1024^3$

キーの表記 : このマニュアルでは、PC-9821 シリーズでのキー表記で説明していません。

**関連資料** このマニュアルを使用する場合は、次の資料もあわせてご覧ください。  
 関連資料は暫定版の場合がありますが、この資料では「暫定」の表示をしておりません。  
 あらかじめご了承ください。

**開発ツールに関する資料 (ユーザーズ・マニュアル)**

資料名	資料番号		
	和文	英文	
IE-V850E1-CD-NW (PCMCIA カード型オンチップ・デバッグ・エミュレータ)	U16647J	U16647E	
QB-V850ESSX2 (V850ES/SG2, V850ES/SJ2 用インサークキット・エミュレータ)	U17091J	U17091E	
QB-V850EIA4 (V850ES/IK1, V850E/IA3, V850E/IA4 用インサークキット・エミュレータ)	U17167J	U17167E	
QB-V850ESKX1 (V850ES/Kx1, V850ES/Kx1+用インサークキット・エミュレータ)	U17214J	U17214E	
CA850 Ver.2.70 C コンパイラ・パッケージ	操作編	U16932J	U16932E
	C 言語編	U16930J	U16930E
	アセンブリ言語編	U16931J	U16931E
	リンク・ディレクティブ編	U16933J	U16933E
CA850 Ver.3.00 C コンパイラ・パッケージ	操作編	U17293J	U17293E
	C 言語編	U17291J	U17291E
	アセンブリ言語編	U17292J	U17292E
	リンク・ディレクティブ編	U17294J	U17294E
PM+ Ver.6.00 プロジェクト・マネージャ	U17178J	U17178E	
ID850QB Ver.3.20 統合デバッガ	操作編	このマニュアル	U17964E
SM+ システム・シミュレータ	操作編	U18010J	U18010E
	ユーザ・オープン・インタフェース編	U17663J	U17663E
SM850 Ver.2.50 システム・シミュレータ	操作編	U16218J	U16218E
SM850 Ver.2.00 以上 システム・シミュレータ	外部部品ユーザ・オープン・インタフェース仕様編	U14873J	U14873E
RX850 Ver.3.20 リアルタイム OS	基礎編	U13430J	U13430E
	インストレーション編	U17419J	U17419E
	テクニカル編	U13431J	U13431E
	タスク・デバッグ編	U17420J	U17420E
RX850 Pro Ver.3.20 リアルタイム OS	基礎編	U13773J	U13773E
	インストレーション編	U17421J	U17421E
	テクニカル編	U13772J	U13772E
	タスク・デバッグ編	U17422J	U17422E
RX-NET ネットワーク・ライブラリ (TCP/IP) Ver.1.30	U15083J	-	
RX-NET ネットワーク・ライブラリ (PPP) Ver.1.30	U15303J	-	
RX-NET ネットワーク・ライブラリ (DNS) Ver.1.30	U15304J	-	
RX-NET ネットワーク・ライブラリ (DHCP) Ver.1.30	U15382J	-	
RX-NET ネットワーク・ライブラリ (SMTP)	U15505J	-	
RX-NET ネットワーク・ライブラリ (POP)	U15539J	-	
RX-NET Ver.1.10 ネットワーク・ライブラリ (telnet)	U16085J	-	
RX-NET Ver.1.00 ネットワーク・ライブラリ (FTP)	U15946J	-	
RX-NET Ver.1.00 ネットワーク・ライブラリ (WebServer)	U16294J	-	
AZ850 Ver.3.30 システム・パフォーマンス・アナライザ	U17423J	U17423E	
PG-FP4 フラッシュ・メモリ・プログラマ	U15260J	U15260E	
TW850 Ver.2.00 性能解析チューニング・ツール	U17241J	U17241E	

# 目次

- 第 1 章 概要 … 18
  - 1.1 特長 … 19
    - 1.1.1 新機能, 強化機能 … 19
    - 1.1.2 その他 … 20
  - 1.2 システム構成 … 21
  - 1.3 動作環境 … 24
    - 1.3.1 ハードウェア環境 … 24
    - 1.3.2 ソフトウェア環境 … 24
  - 1.4 デバッグ時の注意事項 … 25
    - 1.4.1 ソース・レベルのデバッグを行う場合 … 25
    - 1.4.2 セキュリティ ID [MINICUBE] [MINICUBE2] … 25
  - 1.5 GHS 社製コンパイラ使用に関して … 26
- 第 2 章 インストール … 27
  - 2.1 インストール … 27
  - 2.2 アンインストール … 27
- 第 3 章 起動と終了 … 28
  - 3.1 起動前の注意 [MINICUBE] [MINICUBE2] … 28
  - 3.2 起動オプションと引数の指定 … 29
    - 3.2.1 指定方法 … 29
    - 3.2.2 指定形式とオプション … 30
  - 3.3 起動方法 … 31
  - 3.4 終了方法 … 32
  - 3.5 起動時のエラー … 33
    - 3.5.1 IECUBE 接続時 … 33
    - 3.5.2 N-Wire CARD, MINICUBE 接続時 … 34
    - 3.5.3 MINICUBE2 接続時 … 35
- 第 4 章 PM+ との連携 … 36
  - 4.1 ビルド・モードの設定 … 37
  - 4.2 PM+ プロジェクトへのデバッグ登録 … 37
    - 4.2.1 デバッグ選択 … 37
    - 4.2.2 複数のロード・モジュール・ファイルをダウンロードする … 38
  - 4.3 PM+ から ID850QB を起動するには … 39
    - 4.3.1 デバッグ環境の再現 … 39
  - 4.4 オートロード … 40
    - 4.4.1 ソース修正によるオートロード … 40
    - 4.4.2 デバッグ起動によるオートロード … 41
- 第 5 章 デバッグ機能 … 42
  - 5.1 デバッグ環境の設定 … 43
    - 5.1.1 動作環境の設定 … 44
    - 5.1.2 オプションの設定 … 44
    - 5.1.3 マッピング設定 … 44
    - 5.1.4 外部メモリのアクセスに必要なレジスタの値の変更 … 45
  - 5.2 ダウンロード／アップロード機能 … 46
    - 5.2.1 ダウンロード … 47
    - 5.2.2 アップロード … 48
  - 5.3 ソース表示／逆アセンブル表示機能 … 49
    - 5.3.1 ソース表示 … 50
    - 5.3.2 逆アセンブル表示 … 50
    - 5.3.3 混合表示モード (ソース・テキスト・ウインドウ) … 51
    - 5.3.4 シンボル変換 … 52
  - 5.4 ブレーク機能 … 53
    - 5.4.1 ブレークの種類 … 54
    - 5.4.2 ブレーク・ポイントの設定 … 55

- 5.4.3 変数へのブレーク設定 … 56
- 5.4.4 ハードウェア・ブレークとソフトウェア・ブレーク … 57
- 5.4.5 フェイルセーフ・ブレーク機能 [IECUBE] … 58
- 5.5 プログラム実行機能 … 59
- 5.6 ウォッチ機能 … 61
  - 5.6.1 データ値の表示, 変更 … 62
  - 5.6.2 ローカル変数値の表示, 変更 … 62
  - 5.6.3 ウォッチ・データの登録, 削除 … 63
  - 5.6.4 ウォッチ・データの変更 … 63
  - 5.6.5 データ値の一時的表示, 変更 … 64
  - 5.6.6 バルーン・ウォッチ機能 … 64
  - 5.6.7 スタック・トレース表示機能 … 64
- 5.7 メモリ操作機能 … 65
  - 5.7.1 メモリ内容の表示, 変更 … 66
  - 5.7.2 アクセス・モニタ機能 [IECUBE] … 66
  - 5.7.3 メモリ内容の初期化, コピー, 比較 … 67
  - 5.7.4 フラッシュ・メモリへの書き込み機能 [MINICUBE] [MINICUBE2] … 67
- 5.8 レジスタ操作機能 … 68
  - 5.8.1 レジスタ内容の表示, 変更 … 69
  - 5.8.2 周辺 I/O レジスタ内容の表示, 変更 … 69
  - 5.8.3 I/O ポート内容の表示, 変更 … 70
- 5.9 タイマ機能 [IECUBE] … 71
  - 5.9.1 タイマ・イベント条件 … 72
  - 5.9.2 Run-Break イベント … 72
- 5.10 トレース機能 [IECUBE] … 73
  - 5.10.1 トレース・メモリ … 74
  - 5.10.2 トレース・データの設定 … 74
  - 5.10.3 トレース・データの確認 … 75
  - 5.10.4 混合表示モード (トレース・ウインドウ) … 76
  - 5.10.5 トレーサの動作 … 76
  - 5.10.6 条件トレースの設定 … 78
  - 5.10.7 DMA ポイント・トレース機能 … 78
- 5.11 カバレッジ測定機能 [IECUBE] … 79
  - 5.11.1 カバレッジ測定結果の表示 … 80
  - 5.11.2 カバレッジ測定範囲 … 80
  - 5.11.3 カバレッジ実行箇所の表示 … 81
  - 5.11.4 RRM 機能/トレース機能/カバレッジ機能間の排他使用に関して … 82
- 5.12 イベント機能 … 83
  - 5.12.1 イベント機能の利用 … 84
  - 5.12.2 イベントの作成 … 84
  - 5.12.3 各種イベント条件の設定 … 85
  - 5.12.4 各種イベント条件の有効イベント数 … 86
  - 5.12.5 イベントの管理 … 87
- 5.13 RRM 機能 … 89
  - 5.13.1 リアルタイム・モニタ機能 [IECUBE] … 90
  - 5.13.2 疑似リアルタイム・モニタ機能 (Break When Readout) … 91
- 5.14 DMM 機能 … 92
- 5.15 ロード/セーブ機能 … 93
  - 5.15.1 デバッグ環境 (プロジェクト・ファイル) … 94
  - 5.15.2 ウインドウの表示情報 (表示ファイル) … 95
  - 5.15.3 ウインドウの設定情報 (設定ファイル) … 96
- 5.16 ウインドウ共通機能 … 97
  - 5.16.1 アクティブ状態とスタティック状態 … 98
  - 5.16.2 ジャンプ機能 … 99
  - 5.16.3 トレース結果とウインドウの連結機能 [IECUBE] … 101
  - 5.16.4 ドラッグ & ドロップ機能 … 102
  - 5.16.5 注意事項 … 104

## 第 6 章 ウインドウ・リファレンス … 105

- 6.1 ウインドウ一覧 … 106
- 6.2 各ウインドウの説明 … 108
  - メイン・ウインドウ … 109
  - コンフィギュレーション・ダイアログ … 122

拡張オプション設定ダイアログ	130
フェイルセーフ・ブレーク設定ダイアログ	136
RRM 設定ダイアログ	138
フラッシュ・オプション設定ダイアログ	141
データ・フラッシュ・オプション設定ダイアログ	149
デバッグ・オプション設定ダイアログ	152
プロジェクト・ファイル・セーブ・ダイアログ	158
プロジェクト・ファイル・ロード・ダイアログ	159
ダウンロード・ダイアログ	160
アップロード・ダイアログ	163
ロード・モジュール一覧ダイアログ	165
ソース・テキスト・ウインドウ	167
ソース・サーチ・ダイアログ	172
ソース指定ダイアログ	174
逆アセンブル・ウインドウ	176
逆アセンブル・サーチ・ダイアログ	180
アドレス指定ダイアログ	182
シンボル変換ダイアログ	183
リスト・ウインドウ	185
ウォッチ・ウインドウ	188
クイック・ウォッチ・ダイアログ	193
ウォッチ登録ダイアログ	195
ウォッチ変更ダイアログ	198
ローカル変数ウインドウ	200
スタック・トレース・ウインドウ	202
メモリ・ウインドウ	205
メモリ・サーチ・ダイアログ	210
メモリ・フィル・ダイアログ	212
メモリ・コピー・ダイアログ	213
メモリ比較ダイアログ	214
メモリ比較結果ダイアログ	215
DMM ダイアログ	216
レジスタ・ウインドウ	218
レジスタ選択ダイアログ	221
IOR ウインドウ	222
IOR 選択ダイアログ	226
I/O ポート追加ダイアログ	228
タイマ・ダイアログ	230
タイマ測定結果ダイアログ	234
トレース・ウインドウ	236
トレース・サーチ・ダイアログ	241
トレース表示選択ダイアログ	245
フレーム指定ダイアログ	247
トレース・ダイアログ	249
ディレイ・カウント設定ダイアログ	252
コード・カバレッジ・ウインドウ	254
カバレッジ範囲選択ダイアログ	257
ソフトウェア・ブレーク・マネージャ	258
イベント・マネージャ	260
イベント・ダイアログ	264
イベント・リンク・ダイアログ	269
ブレーク・ダイアログ	273
表示ファイル・セーブ・ダイアログ	276
表示ファイル・ロード・ダイアログ	278
環境設定ファイル・セーブ・ダイアログ	280
環境設定ファイル・ロード・ダイアログ	281
リセット確認ダイアログ	282
終了確認ダイアログ	283
バージョン表示ダイアログ	284
コンソール・ウインドウ	285
ソース・ファイル選択ダイアログ	286

第7章 コマンド・リファレンス … 287

7.1 コマンド・ライン規約	288
7.2 コマンド一覧	288
7.3 エイリアス一覧	290
7.4 変数一覧	290
7.5 パッケージ一覧	291
7.6 キー・バインド	291
7.7 拡張ウインドウ	291
7.8 コールバック・プロシージャ	292
7.9 フック・プロシージャ	293
7.10 関連ファイル	294
7.11 注意事項	294
7.12 コマンドの説明	295
address	296
assemble	297
batch	298
breakpoint	299
cache	301
dbgexit	302
dbgopt	303
download	304
erase	305
extwin	306
finish	307
flop	308
go	309
help	310
hook	311
ie	312
inspect	313
jump	314
map	315
mdi	317
memory	318
module	319
next	320
refresh	321
register	322
reset	323
run	324
step	325
stop	326
tkcon	327
upload	328
version	329
watch	330
where	331
wish	332
xcoverage	333
xtime	334
xtrace	335
7.13 サンプル (電卓スクリプト)	336

付録 A 拡張ウインドウ	337
A.1 拡張ウインドウ概要	338
A.2 サンプル・ウインドウ一覧	338
A.3 サンプル・ウインドウの起動	338
A.4 各サンプル・ウインドウの説明	338
List ウインドウ	339
Grep ウインドウ	340
RRM ウインドウ	341
Hook ウインドウ	342
Memory Mapped I/O ウインドウ	344
Memory Mapped I/O 入力ダイアログ	346

Sym Inspect ウィンドウ … 347  
Run Break Time ウィンドウ … 348

付録 B 入力規約 … 349

B.1 使用可能文字 … 350

B.2 シンボル規定 … 351

B.3 数値規定 … 352

B.4 式と演算子に関する規定 … 353

B.5 ファイル名 … 355

付録 C キー機能一覧 … 356

付録 D メッセージ … 359

D.1 メッセージ表示形式 … 359

D.2 メッセージの種類 … 360

D.3 メッセージ一覧 … 361

付録 E 索引 … 387

# 図の目次

図番号 タイトル ページ

---

1-1	ID850QB	18
1-2	システム構成例 [IECUBE]	21
1-3	システム構成例 [MINICUBE]	22
1-4	システム構成例 [MINICUBE2]	23
3-1	起動オプションの設定 (例)	29
3-2	コンフィギュレーション・ダイアログ	31
3-3	終了確認ダイアログ	32
4-1	複数ファイルのダウンロード	38
5-1	ブレーク・ポイントの設定	55
5-2	変数へのブレーク設定	56
5-3	ソフトウェア・ブレークの管理	58
5-4	フェイルセーフ・ブレーク設定	58
5-5	実行ボタン	59
5-6	[実行]メニュー	59
5-7	ウォッチ・ウインドウ	62
5-8	ウォッチ表示形式の指定 (デバッグ・オプション設定ダイアログ)	62
5-9	ローカル変数ウインドウ	62
5-10	ウォッチ変更ダイアログ	63
5-11	クイック・ウォッチ・ダイアログ	64
5-12	スタック・トレース・ウインドウ	64
5-13	アクセス・モニタ機能 (メモリ・ウインドウ)	66
5-14	機能名/絶対名の切り替え	69
5-15	周辺 I/O レジスタ内容の表示	69
5-16	I/O ポートの登録	70
5-17	タイマ・イベントの設定, 表示 (タイマ・ダイアログ)	72
5-18	トレース・データの設定	74
5-19	トレース・データの確認	75
5-20	カバレッジ測定結果の表示	80
5-21	カバレッジ実行/未実行箇所の表示	81
5-22	各種イベント条件の設定方法	85
5-23	イベントの管理 (イベント・マネージャ)	87
5-24	RRM 設定ダイアログ	90
5-25	リアルタイム・モニタ機能のサンプリング間隔の指定	90
5-26	疑似リアルタイム・モニタ機能の指定	91
5-27	メモリ内容の書き換え (DMM ダイアログ)	92
6-1	メイン・ウインドウ	109
6-2	ツールバー (Picture only)	119
6-3	ツールバー (Picture and Text)	119
6-4	ステータスバー	119
6-5	コンフィギュレーション・ダイアログ	122
6-6	拡張オプション設定ダイアログ	130
6-7	フェイルセーフ・ブレーク設定ダイアログ	136
6-8	RRM 設定ダイアログ	138
6-9	フラッシュ・オプション設定ダイアログ	141
6-10	データ・フラッシュ・オプション設定ダイアログ	149
6-11	デバッグ・オプション設定ダイアログ	152
6-12	[Add Source path] ダイアログ	153
6-13	[Font] ダイアログ	154
6-14	プロジェクト・ファイル・セーブ・ダイアログ	158
6-15	プロジェクト・ファイル・ロード・ダイアログ	159
6-16	ダウンロード・ダイアログ	160
6-17	アップロード・ダイアログ	163
6-18	ロード・モジュール一覧ダイアログ	165
6-19	ソース・テキスト・ウインドウ	167
6-20	ソース・サーチ・ダイアログ	172

6-21	ソース指定ダイアログ	174
6-22	逆アセンブル・ウインドウ	176
6-23	逆アセンブル・サーチ・ダイアログ	180
6-24	アドレス指定ダイアログ (例: メモリ・ウインドウ)	182
6-25	シンボル変換ダイアログ	183
6-26	リスト・ウインドウ	185
6-27	ウォッチ・ウインドウ	188
6-28	クイック・ウォッチ・ダイアログ	193
6-29	ウォッチ登録ダイアログ	195
6-30	ウォッチ変更ダイアログ	198
6-31	ローカル変数ウインドウ	200
6-32	スタック・トレース・ウインドウ	202
6-33	メモリ・ウインドウ	205
6-34	メモリ・ウインドウ (RRM 機能選択時)	206
6-35	メモリ・サーチ・ダイアログ	210
6-36	メモリ・フィル・ダイアログ	212
6-37	メモリ・コピー・ダイアログ	213
6-38	メモリ比較ダイアログ	214
6-39	メモリ比較結果ダイアログ	215
6-40	DMM ダイアログ (例: Memory 選択時)	216
6-41	レジスタ・ウインドウ	218
6-42	レジスタ選択ダイアログ	221
6-43	IOR ウインドウ	222
6-44	周辺 I/O レジスタ選択ダイアログ	226
6-45	I/O ポート追加ダイアログ	228
6-46	タイマ・ダイアログ	230
6-47	タイマ測定結果ダイアログ	234
6-48	トレース・ウインドウ	236
6-49	トレース・サーチ・ダイアログ	241
6-50	トレース表示選択ダイアログ	245
6-51	フレーム指定ダイアログ	247
6-52	トレース・ダイアログ	249
6-53	ディレイ・カウント設定ダイアログ	252
6-54	コード・カバレッジ・ウインドウ	254
6-55	カバレッジ範囲選択ダイアログ	257
6-56	ソフトウエア・ブレーク・マネージャ	258
6-57	イベント・マネージャ (詳細表示モード)	260
6-58	イベント・ダイアログ	264
6-59	イベント・リンク・ダイアログ	269
6-60	ブレーク・ダイアログ	273
6-61	表示ファイル・セーブ・ダイアログ	276
6-62	表示ファイル・ロード・ダイアログ	278
6-63	環境設定ファイル・セーブ・ダイアログ	280
6-64	環境設定ファイル・ロード・ダイアログ	281
6-65	リセット確認ダイアログ	282
6-66	終了確認ダイアログ	283
6-67	バージョン表示ダイアログ	284
6-68	コンソール・ウインドウ	285
6-69	ソース・ファイル選択ダイアログ	286
7-1	実行画面	336
A-1	List ウインドウ	339
A-2	Grep ウインドウ	340
A-3	RRM ウインドウ	341
A-4	Hook ウインドウ	342
A-5	Memory Mapped I/O ウインドウ	344
A-6	Memory Mapped I/O 入力ダイアログ	346
A-7	Sym Inspect ウインドウ	347
A-8	RunBreakTime ウインドウ	348
D-1	エラー/ワーニング・ダイアログ	359

# 表の目次

表番号	タイトル	ページ
2-1	インストール	27
3-1	起動オプション	30
3-2	エラー・メッセージの出力パターン [IECUBE]	33
5-1	デバッグ機能一覧 (デバッグ操作の流れ)	42
5-2	マッピング属性の種類	44
5-3	ダウンロードが可能なファイルの種類	47
5-4	アップロードが可能なファイルの種類	48
5-5	表示可能なファイルの種類	50
5-6	シンボルでの指定方法	52
5-7	主なブレークの種類	54
5-8	ソフトウェア・ブレークの有効数	57
5-9	実行の種類	60
5-10	トレース・メモリ・サイズ	74
5-11	トレースの種類	76
5-12	トレーサ制御モードの種類	77
5-13	条件トレースの種類	78
5-14	コード・カバレッジ測定範囲	80
5-15	カバレッジ実行/未実行箇所の表示形式	81
5-16	各種イベント条件	84
5-17	各種イベント条件における有効イベント数	86
5-18	イベント・アイコン	88
5-19	リアルタイム・モニタ機能サンプリング可能領域	90
5-20	プロジェクト・ファイルの保存内容	94
5-21	表示ファイルの種類	95
5-22	設定ファイルの種類	96
5-23	ジャンプ元アドレスの詳細	99
5-24	ドラッグ & ドロップ機能の詳細 (行/アドレスの場合)	102
5-25	ドラッグ & ドロップ機能の詳細 (文字列の場合)	103
6-1	ウインドウ/ダイアログ一覧	106
6-2	CPU ステータス	120
6-3	IE ステータス	120
6-4	ブレーク要因	121
6-5	分周率と最大測定時間の関係 (タイムタグ・カウンタ (トレース))	131
6-6	採取するトレース・データの意味とトレース採取モードとの関係	132
6-7	分周率と最大測定時間の関係 (タイマ・カウンタ (タイマ))	133
6-8	フラッシュ・セルフ・プログラミング・エミュレーション対応デバイス	144
6-9	フラッシュ関数エミュレーション可否一覧 (Type1)	144
6-10	フラッシュ関数エミュレーション可否一覧 (Type3)	145
6-11	イベント設定状態	168
6-12	ウォッチ・ウインドウ入力形式	196
6-13	スコープで指定した場合の変数の扱い	196
6-14	測定可能値	232
6-15	アドレス条件の設定範囲 (トレース)	243
6-16	フレーム番号の指定形式	248
6-17	設定可能なイベント条件数	250
6-18	カバレッジ測定範囲 (詳細)	257
6-19	イベント詳細表示時のセパレータ	261
6-20	アドレス条件の設定範囲 (イベント)	266
6-21	イベント・リンク・ダイアログでのイベント設定数	271
6-22	条件設定エリアのイベント設定数	274
7-1	デバッガ制御コマンド一覧	288
7-2	コンソール/Tcl コマンド一覧	289
7-3	aliases.tcl ファイルの内容	290
7-4	変数一覧	290
7-5	パッケージ一覧	291

7-6	メッセージ ID	… 292
7-7	関連ファイル一覧	… 294
A-1	拡張ウィンドウのサンプル・ウィンドウ一覧	… 338
B-1	文字セット一覧	… 350
B-2	特殊文字一覧	… 350
B-3	数値の入力形式	… 352
B-4	演算子一覧	… 353
B-5	演算子の優先順位	… 354
B-6	進数の範囲	… 354
C-1	キー機能一覧	… 356
D-1	メッセージの種類	… 360



## 1.1 特長

ID850QB の特長を示します。

- [新機能, 強化機能](#)
- [その他](#)

### 1.1.1 新機能, 強化機能

#### (1) MINICUBE2 に対応

業界最小クラスのオンチップ・デバッグ・エミュレータ MINICUBE2 に対応しました。

#### (2) GHS 社製コンパイラ (DWARF2 形式) に対応

Green Hills Software 社製コンパイラに対応したロード・モジュール形式である GHS 拡張 DWARF2 に対応しました ([「1.5 GHS 社製コンパイラ使用に関して」](#) 参照)。

#### (3) リスト・ウインドウの追加

関数, 変数, シンボル, セクション, 割り込み要求名の一覧表示を行うリスト・ウインドウを追加しました ([「リスト・ウインドウ」](#) 参照)。

情報はタブごとに分けられており, タブごとに CSV 形式での保存が可能です。

#### (4) Data Not Equal に対応

イベント発生時のステータス条件に, データ条件 (リード/ライト) が不一致の場合 (Data Not Equal) を追加しました ([「イベント・ダイアログ」](#) 参照)。

#### (5) データ・フラッシュに対応

V850ES/Fx3 のデータ・フラッシュ・メモリへの書き込み/読み出しに対応しました。

また, EEPROM ライブラリ・エラー・エミュレーションに対応しました ([「データ・フラッシュ・オプション設定ダイアログ」](#) 参照)。

## 1.1.2 その他

### (1) インサーキット・エミュレータの機能を利用

インサーキット・エミュレータの持つ詳細なイベント設定機能を利用して、ブレーク・イベントの設定、ユーザ・プログラムの時間測定、トレース・データの収集などを行うことができます（「[5.12 イベント機能](#)」参照）。

### (2) オンチップ・デバッグをサポート [MINICUBE]

Nx85ET (RCU0+TEU+TRCU), Nx85E901 (RCU0), RCU1 のオンチップ・デバッグ・ユニット (RCU) によるデバッグ機能を提供します。

### (3) フラッシュ・メモリへの書き込み機能 [MINICUBE] [MINICUBE2]

通常のメモリ操作と同様のアクセス方法により、内蔵フラッシュ・メモリへの書き込み、およびロード・モジュールのダウンロードが可能です（「[5.7.4 フラッシュ・メモリへの書き込み機能 \[MINICUBE\] \[MINICUBE2\]](#)」参照）。

### (4) セキュリティ機能 [MINICUBE] [MINICUBE2]

セキュリティ・ユニット搭載品に対応し、内蔵 ROM, または内蔵フラッシュ・メモリに格納された ID コード認証を行います（「[コンフィギュレーション・ダイアログ](#)」の「[\(5\) ID Code \(ID コード入力エリア\) \[MINICUBE\] \[MINICUBE2\]](#)」参照）。

### (5) Tcl による機能拡張

Tcl/Tk (Tool Command Language) によるコマンド・ラインでのバッチ処理やフック処理、ユーザ独自のカスタム・ウィンドウの作成が可能です（「[第 7 章 コマンド・リファレンス](#)」, 「[付録 A 拡張ウィンドウ](#)」参照）  
最新の Tcl/Tk コア 8.4.12, および新コンソールに対応しています。

### (6) TIP または ToolLink による機能拡張

TIP (Tool Interface Protocol), または ToolLink に対応したタスク・デバッガ (RD), システム・パフォーマンス・アナライザ (AZ) などと連携することにより, リアルタイム OS (RX) を利用したユーザ・プログラムのデバッグ効率を飛躍的に向上させることが可能です。

## 1.2 システム構成

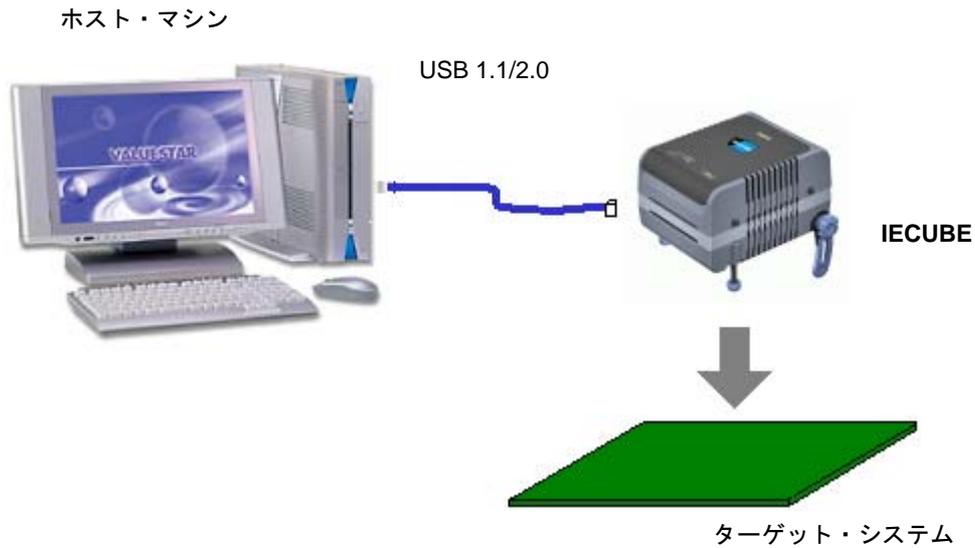
ID850QB は、次に挙げる 3 つのタイプのエミュレータと接続可能です。

これにより、V850 シリーズ用に開発されたユーザ・プログラム、およびターゲット・システムの快適なデバッグ環境を提供しています。

### (1) IECUBE

IECUBE は、USB ケーブルでホスト・マシンと接続することにより、ID850QB から操作します。

図 1-2 システム構成例 [IECUBE]



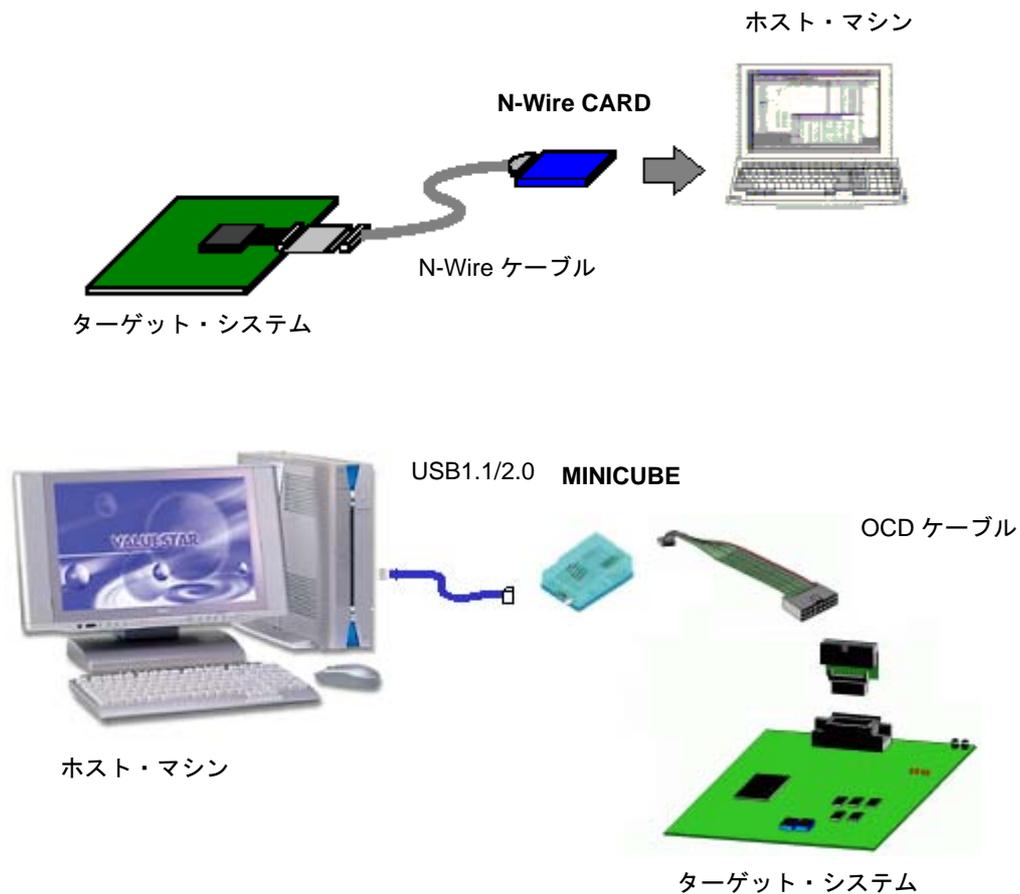
**(2) N-Wire CARD, MINICUBE**

N-Wire CARD, MINICUBE は、DCU (Debug Control Unit) を搭載した V850ES、および V850E1 を実装したターゲット・システムと接続することにより、デバッグ機能の提供が可能になります。

なお、N-Wire CARD は、PC カード型エミュレータであるため、ホスト・マシンに直接挿入することで、ID850QB から操作します。

MINICUBE は、USB ケーブルでホスト・マシンと接続することにより、ID850QB から操作します。

図 1 - 3 システム構成例 [MINICUBE]



**(3) MINICUBE2**

MINICUBE2 は、USB ケーブルでホスト・マシンと接続することにより、ID850QB から操作します。なお、MINICUBE2 は、オンチップ・デバッグ機能内蔵マイコンと接続することにより、デバッグ機能の提供が可能になります。

図 1 - 4 システム構成例 [MINICUBE2]



## 1.3 動作環境

この節では、動作環境に関する以下の項目について解説します。

- ハードウェア環境
- ソフトウェア環境

### 1.3.1 ハードウェア環境

#### (1) ホスト・マシン (対象 OS が動作するマシン)

CPU	Pentium II™ 400MHz 以上
メイン・メモリ	256M バイト以上

**注意** N-Wire CARD 接続時は、ノート PC での使用を想定しているため PC カード・スロット (TYPEII) 搭載品である必要があります。

#### (2) 対応インサーキット・エミュレータ

- IECUBE (QB-V850Exxxx)
- N-Wire CARD (IE-V850E1-CD-NW)
- MINICUBE (QB-V850MINI)
- MINICUBE2 (QB-MINI2)

### 1.3.2 ソフトウェア環境

#### (1) OS (下記のいずれか)

Windows<sup>(R)</sup>98, Windows<sup>(R)</sup>2000, Windows<sup>(R)</sup>Me, Windows<sup>(R)</sup>XP(Home Edition, Professional)

**注意** いずれの場合も最新の Service Pack がインストールされていることを推奨します。

#### (2) デバイス・ファイル (個別入手)

- 使用するターゲット・デバイスのデバイス・ファイル

**参考** 下記に示す NEC エレクトロニクス社の Web サイトから入手可能です (ODS)。

<http://www.necel.com/micro/>

#### (3) 対応ツール (NEC エレクトロニクス製)

- C コンパイラ・パッケージ CA850 (V3.10 以上)
- プロジェクト・マネージャ PM+ (V6.10 以上)
- システム・パフォーマンス・アナライザ AZ850 (V3.30 以上)
- 性能解析チューニング・ツール TW850 (V2.10 以上)

## 1.4 デバッグ時の注意事項

デバッグ時の注意事項を示します。

- [ソース・レベルのデバッグを行う場合](#)
- [セキュリティ ID \[MINICUBE\] \[MINICUBE2\]](#)

### 1.4.1 ソース・レベルのデバッグを行う場合

ソース・レベルのデバッグを行うオブジェクト・ファイルには、シンボル情報やその他デバッグを行うための情報（デバッグ情報）が含まれている必要があります。

このため、ソース・ファイルのコンパイル時には、以下の処理を行ってください。

#### (1) PM+ 使用時

ビルド・モードの選択時に [Debug Build] を指定

#### (2) PM+ 未使用時

-g オプションを追加

### 1.4.2 セキュリティ ID [MINICUBE] [MINICUBE2]

N-Wire CARD, MINICUBE, MINICUBE2 接続時に使用するオブジェクト・ファイルには、セキュリティ ID の情報が組み込まれている必要があります。

セキュリティ ID の設定に関しては「CA850 操作編」のユーザーズ・マニュアルを参照してください。

また、セキュリティ ID の詳細に関しては、N-Wire CARD, MINICUBE, MINICUBE2 のユーザーズ・マニュアルを参照してください。

なお、ID850QB 上からのセキュリティ ID (ID コード) の設定は、[コンフィギュレーション・ダイアログ](#)で行います。

## 1.5 GHS 社製コンパイラ使用に関して

ID850QB では、V3.20 より、GHS 拡張 DWARF2 形式に対応した Green Hills Software 社製コンパイラをサポートしています。

### (1) サポート・バージョン

- 統合開発環境 MULTI™ (V3.5.1, V4.0.5) (DWARF2 形式)

**注意** C++ 言語は未サポートです。また、V850E2, V850E2R コアは未サポートです。

### (2) デバッグ時の付加オプション (デバッグ・オプション)

- -G -dual\_debug

## 第2章 インストール

この章では、ID850QB のインストールに関する次の項目について解説します。

- [インストール](#)
- [アンインストール](#)

### 2.1 インストール

ID850QB の使用に際し、次のものをインストールする必要があります。

表 2-1 インストール

項目	方法
ID850QB	【ID850QB Disk を使用する場合】 自動実行されるインストーラに従いインストールします。 【ODS からダウンロードした場合】 ダウンロードした実行形式ファイルを実行し、インストーラに従いインストールします。
デバイス・ファイル	[スタート]メニュー→[プログラム]→[NEC Electronics Tools]→[デバイス ファイル インストーラ]を選択することにより起動する専用インストーラ "DFINST.exe" に従いインストールします。

### 2.2 アンインストール

アンインストールは、コントロール・パネルの [アプリケーションの追加と削除] (または [プログラムの追加と削除]) を用いて行ってください。

## 第3章 起動と終了

この章では、ID850QBの起動と終了に関する次の項目について解説します。

- [起動前の注意 \[MINICUBE\] \[MINICUBE2\]](#)
- [起動オプションと引数の指定](#)
- [起動方法](#)
- [終了方法](#)
- [起動時のエラー](#)

### 3.1 起動前の注意 [MINICUBE] [MINICUBE2]

N-Wire CARD, MINICUBE を接続している場合、または MINICUBE2 を接続している場合には、ID850QB の起動以前に、以下に示すツールを起動し、インサーキット・エミュレータとターゲット・システムが正常にデバッグできる状態であるかどうかを確認してください。

- N-Wire Checker **[MINICUBE]**
- OCD Checker **[MINICUBE2]**

**注意** インサーキット・エミュレータとターゲット・システムの接続、および電源の投入順序に関しては、N-Wire CARD, MINICUBE, または MINICUBE2 のユーザーズ・マニュアルを参照してください。誤った接続は、インサーキット・エミュレータ、およびターゲット・システムを破壊する恐れがあります。

## 3.2 起動オプションと引数の指定

ID850QB の起動オプションと引数を指定する際の手順を示します。

起動オプションと引数を指定することで、起動時のスクリプト・ファイル指定、およびプロジェクト・ファイル指定が可能です。

**参考** PM+ から ID850QB を起動する場合、起動オプションと引数の設定は PM+ の [ツール] メニューの [デバッグの設定 ...] で行います (「第4章 PM+ との連携」参照)。オプション欄にデバッグの起動オプションが設定できます。

### 3.2.1 指定方法

- 1) ID850QB のショートカットをデスクトップ上に作成します。  
ID850QB の実行ファイルは、インストールしたフォルダ内の bin フォルダにあります。
- 2) 作成したショートカットのプロパティを開き、[リンク先] に示される実行ファイル名の後に、オプション、引数を指定します (「3.2.2 指定形式とオプション」参照)。

図 3 - 1 起動オプションの設定 (例)



## 3.2.2 指定形式とオプション

### (1) 指定形式

```
id850g32.exe ?options?
id850g32.exe ?options? project
```

各オプションと引数はスペースで区切ります。文字列の大文字と小文字は区別しません。

?で囲まれた引数は省略可能です。

プロジェクト・ファイルを指定すると起動時にプロジェクト・ファイルを読み込みます。ただし、PM+ 起動中はプロジェクト・ファイルの指定を無視します。

なお、ファイル名、およびパス内にスペースがある場合には、プロジェクト・ファイル名、スクリプト・ファイル名を" "で囲んで指定してください（「例3」パス内にスペースがある場合の指定」参照）。

### (2) 指定オプション

指定できるオプションは次の通りです。

表 3 - 1 起動オプション

オプション	意味
/sc	ウインドウの背景色をシステム・カラーにする。
/script:script file name	起動時に実行するスクリプト・ファイルを指定する。

### (3) 指定例

#### 例 1) スクリプト・ファイルのみ指定

```
id850g32.exe /script:c:\work\script.tcl
```

#### 例 2) スクリプト・ファイルとプロジェクト・ファイルを指定

```
id850g32.exe /script:c:\work\script.tcl c:\work\project.prj
```

#### 例 3) パス内にスペースがある場合の指定

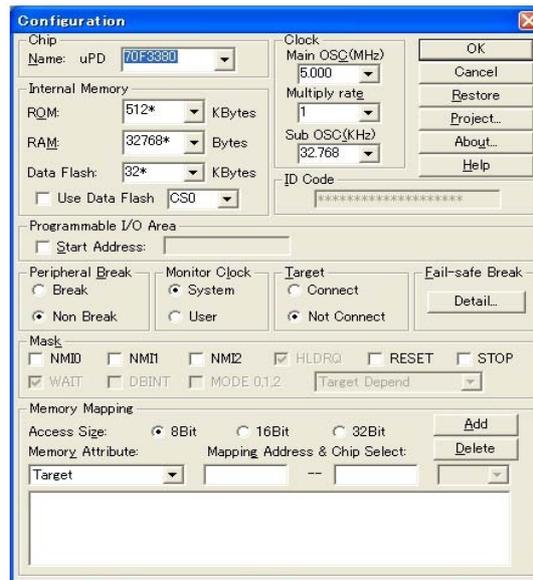
```
id850g32.exe /script:"c:\work folder\script.tcl" "c:\work folder\project.prj"
```

### 3.3 起動方法

- 1) ID850QB の起動は PM+, [スタート]メニュー, またはデスクトップ上に作成されたショートカットにより行います。なお, PM+ から起動する場合には, 「4.3 PM+ から ID850QB を起動するには」を参照してください。起動により, [コンフィギュレーション・ダイアログ](#)がオープンします。

**注意** この際, [コンフィギュレーション・ダイアログ](#)が表示されず, エラー・メッセージが表示された場合には, 「3.5 起動時のエラー」を参照し対処してください。

図 3-2 コンフィギュレーション・ダイアログ

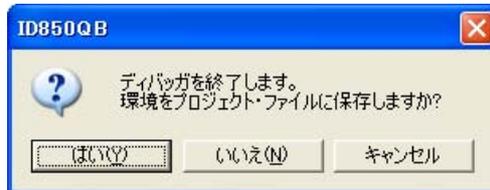


- 2) [コンフィギュレーション・ダイアログ](#)では ID850QB の動作環境に関する各種設定を行います。各項目の設定後, ダイアログ上の <OK> ボタンをクリックします。
- 3) [メイン・ウインドウ](#)がオープンし, ID850QB の操作が可能になります。デバッグ作業はメイン・ウインドウを中心に行います。

## 3.4 終了方法

- 1) ID850QB の **メイン・ウインドウ** で [ファイル] メニュー → [終了] を選択することにより、**終了確認ダイアログ** がオープンします（プログラム実行中に終了操作を行った場合は、実行停止確認のメッセージ・ボックスが表示されます）。

図 3-3 終了確認ダイアログ



- 2) 現在のデバッグ環境をプロジェクト・ファイルに保存したい場合は、<はい> ボタンをクリックします。<いいえ> ボタンをクリックした場合には、プロジェクト・ファイルには保存せず、すべてのウインドウがクローズし ID850QB が終了します。

### 3.5 起動時のエラー

以下に ID850QB の起動時に出力される可能性のあるエラー・メッセージを示します（出現順）。

これらのエラーが出力された場合には、「付録 D メッセージ」を参照してください。

**注意** 複数の IE を接続した場合の優先順位は以下のようになります。

IECUBE > MINICUBE > MINICUBE2 > N-Wire CARD

#### 3.5.1 IECUBE 接続時

ターゲットなどの接続状態と **コンフィギュレーション・ダイアログ** の設定状態によって出力されるエラー・メッセージのパターンは次の通りです。

表 3-2 エラー・メッセージの出力パターン [IECUBE]

エラー・メッセージ	コンフィギュレーション・ダイアログの [Target]		ターゲット		変換アダプタ		ターゲット電源	
	Connect	Not Connect	接続	未接続	有	無	ON	OFF
Ff606: ターゲットとの接続を確認し、ターゲットの電源を入れてください。	○							○
Wf607: 変換アダプタの接続を確認してください。		○		○		○		○
Ff608: ターゲットを外してください。		○	○					○
Ff609: ターゲットの電源を切り、ターゲットを外してください。		○					○	

F0100: インサーキット・エミュレータと通信できません。インターフェース・ボード用のデバイス・ドライバが正しくインストールされているか確認してください。
F0c43: エミュレータの接続ができません。
F0c70: DCU にアクセスできません。
F0c76: DCU アクセス開始時の初期状態異常です。[V850]
F0c77: DCU アクセス異常です。
Ff606: ターゲットとの接続を確認し、ターゲットの電源を入れてください。
Wf607: 変換アダプタの接続を確認してください。

Ff608: ターゲットを外してください。
A0105: デバイス・ファイル (file name) を正しく読めませんでした。
F0ca2: オンチップ・デバッグに対応していないデバイス・ファイルです。
F0ca4: IECUBE に対応していないデバイス・ファイルです。
F0c71: リセットができません。
F0c72: モニタ・メモリにアクセスできません。
F0c73: モニタ実行できません。
F0c74: CPU レジスタにアクセスできません。
F0c23: バス・ホールドが継続中です。
A0c01: レジスタのアクセス中に CPU からの応答がなくなりました。
A0c02: メモリのアクセス中に CPU からの応答がなくなりました。
F0c04: 外部空間にあるフラッシュ・メモリのデータベース・ファイルが見つかりません。
A01a0: エバチップから応答がありません。RESET, WAIT, HLDRQ などの信号やクロック信号に異常がないか確認してください。CPU から応答がありません。RESET, WAIT, HLDRQ などの信号やクロック信号に異常がないか確認してください。

### 3.5.2 N-Wire CARD, MINICUBE 接続時

F0100: インサーキット・エミュレータと通信できません。インタフェース・ボード用のデバイス・ドライバが正しくインストールされているか確認してください。
F0c43: エミュレータの接続ができません。
F03a0: ターゲットの電源が OFF です。
F0c70: DCU にアクセスできません。
F0c76: DCU アクセス開始時の初期状態異常です。[V850]
F0c77: DCU アクセス異常です。
A0105: デバイス・ファイル (file name) を正しく読めませんでした。
F0ca2: オンチップ・デバッグに対応していないデバイス・ファイルです。
F0ca3: デバイス・ファイルのオンチップ・デバッグ情報に未サポートの情報が含まれています。
F0c24: デバッグモードに移行できません。
F0c72: モニタ・メモリにアクセスできません。
F0c73: モニタ実行できません。
F0c74: CPU レジスタにアクセスできません。
F0c23: バス・ホールドが継続中です。
A0c01: レジスタのアクセス中に CPU からの応答がなくなりました。
A0c02: メモリのアクセス中に CPU からの応答がなくなりました。

F0c04: 外部空間にあるフラッシュ・メモリのデータベース・ファイルが見つかりません。

A01a0: エバチップから応答がありません。RESET, WAIT, HLDRQ などの信号やクロック信号に異常がないか確認してください。CPU から応答がありません。RESET, WAIT, HLDRQ などの信号やクロック信号に異常がないか確認してください。

### 3.5.3 MINICUBE2 接続時

F0100: インサーキット・エミュレータと通信できません。インタフェース・ボード用のデバイス・ドライバが正しくインストールされているか確認してください。

F0c43: エミュレータの接続ができません。

A01b2: エミュレータのファームウェアのバージョンが最新ではありません。ユーティリティで最新のファームウェアに更新してください。

A0105: デバイス・ファイル (file name) を正しく読めませんでした。

F03a0: ターゲットの電源が OFF です。

F0ca2: オンチップ・デバッグに対応していないデバイス・ファイルです。

F0ca3: デバイス・ファイルのオンチップ・デバッグ情報に未サポートの情報が含まれています。

F0ca1: モニタ・ファイルが見つかりませんでした。

F0c00: モニタ・ファイルの読み込みに失敗。

F0c71: リセットができません。

F02a3: リセットが継続中です。

F0c72: モニタ・メモリにアクセスできません。

F0c24: デバッグモードに移行できません。

F0c74: CPU レジスタにアクセスできません。

F0c73: モニタ実行できません。

F0c33: オンチップ・デバッグを使用禁止にしようとしました。

F0c34: オンチップ・デバッグで使用する予約領域への書き込みはできません。

A010a: デバッグとユーティリティの同時起動はできません。

A01a6: イグゼキュタ起動中です。

## 第 4 章 PM+ との連携

ID850QB では PM+（プロジェクト・マネージャ）との連携により、"プログラム作成→コンパイル→デバッグ→プログラムの修正"といった開発工程中の一連の作業を自動的に行うことができます。

この章では、PM+ との連携に関する次の項目について解説します。

なお、PM+ の機能詳細に関しては、PM+ のユーザーズ・マニュアルを参照してください。

- [ビルド・モードの設定](#)
- [PM+ プロジェクトへのデバッグ登録](#)
- [PM+ から ID850QB を起動するには](#)
- [オートロード](#)

**注意** Windows のコマンド・プロンプトを使用してロード・モジュール・ファイルを作成した場合、ID850QB と PM+ の連携機能は使用できません。

## 4.1 ビルド・モードの設定

PM+ 上で作成するロード・モジュール・ファイルを ID850QB 上でソース・レベル・デバッグする場合には、デバッグ情報を出力するビルドを行い、ロード・モジュール・ファイルを生成する必要があります。その設定は、PM+ 上で [Debug Build] を指定することにより行います。

## 4.2 PM+ プロジェクトへのデバッグ登録

PM+ ではプロジェクトごとに、使用するデバッガやダウンロードするロード・モジュール・ファイルを指定することができます。

### 4.2.1 デバッガ選択

デバッガの選択は、次のいずれかの方法で行います。

これにより、アクティブなプロジェクトのデバッガとして ID850QB が登録されます。また、PM+ のツールバーに ID850QB のアイコンが表示されます。

#### (1) 新規にワークスペースを作成する場合

- 1) PM+ 上の [ファイル] メニュー → [ワークスペースの新規作成 ...] を選択します。  
→ ウィザード形式のワークスペースの新規作成ダイアログがオープンします。
- 2) ウィザードにより、ワークスペースに必要な設定をすすめていくと、[デバッガの選択] ダイアログが表示されます。選択デバッガで ID850QB を指定してください。その他の設定項目に関する詳細は PM+ のユーザーズ・マニュアルを参照してください。

#### (2) 既存のワークスペースを使用する場合

- 1) PM+ 上の [ツール] メニュー → [デバッガの設定 ...] を選択します。  
→ [デバッガの設定] ダイアログがオープンします。
- 2) 選択デバッガで ID850QB を指定して、<OK> ボタンをクリックします。その他の設定項目に関する詳細は PM+ のユーザーズ・マニュアルを参照してください。

## 4.2.2 複数のロード・モジュール・ファイルをダウンロードする

ID850QB では、同じプロジェクト・グループ内の複数のロード・モジュール・ファイルのダウンロードが可能です。

複数のロード・モジュール・ファイルのダウンロードは、PM+ の [ デバッガの選択 ] ダイアログで指定することにより行います。

図 4 - 1 複数ファイルのダウンロード



なお、プロジェクト・グループに関する詳細は、PM+ のユーザーズ・マニュアルを参照してください。

**参考** 複数のロード・モジュール・ファイルがダウンロードされていることは、ID850QB の [ロード・モジュール一覧ダイアログ](#) で確認することができます。

**注意** 内蔵フラッシュ・メモリ搭載品では、[ デバッガの設定 ] ダイアログ上の [ デバッガにダウンロード時、シンボル・リセットを実行する ] をチェックした場合、ダウンロード前に内蔵フラッシュ・メモリの消去が行われます。

## 4.3 PM+ から ID850QB を起動するには

PM+ から ID850QB を起動するには、次の方法があります。

- PM+ のツールバー上の ID850QB 起動ボタンをクリックする。
- PM+ の [ビルド] メニュー → [デバッグ] を選択する。
- PM+ の [ビルド] メニュー → [ビルド→デバッグ] を選択する。
- PM+ の [ビルド] メニュー → [リビルド→デバッグ] を選択する。

現在、PM+ で使用しているプロジェクト・ファイルに ID850QB のデバッグ環境が保存されている場合には、保存されているデバッグ環境の状態で起動します。

PM+ で使用しているプロジェクト・ファイルに ID850QB のデバッグ環境が保存されていない場合には、[コンフィギュレーション・ダイアログ](#)が表示されます。このとき、デバイス種別 (Chip 名) を変更することはできません。

**注意** PM+ で多数のソース・ファイルをプロジェクトに登録した場合、ID850QB に登録可能なソース・パス長の上限を越え、ソース・ファイルを自動表示できない場合があります。

ソース・パス長の詳細は[デバッグ・オプション設定ダイアログ](#)の「(1) Source Path (ソース・パス指定エリア)」を参照してください。

### 4.3.1 デバッグ環境の再現

次に示す手順で、PM+ から ID850QB を起動時に前回のデバッグ環境を再現することができます。

- 1) PM+ で新規ワークスペース (プロジェクト・ファイル) (例: sample.prj) を作成します<sup>注</sup>。
- 2) PM+ から ID850QB を起動します。新規のプロジェクト・ファイルのため、ID850QB 単体での起動時と同様に、[コンフィギュレーション・ダイアログ](#)でデバイス種別 (Chip 名) 以外の項目を設定します。
- 3) ID850QB の[ダウンロード・ダイアログ](#)でデバッグ対象のロード・モジュール・ファイルをダウンロードします。
- 4) ID850QB でデバッグを行います。
- 5) ID850QB 終了時に、[終了確認ダイアログ](#)で <はい> ボタンをクリックし、ID850QB を終了します。  
→ PM+ のプロジェクト・ファイル (sample.prj) に ID850QB 終了時のデバッグ環境が保存されます (sample.prj へのデバッグ環境の保存は、ID850QB 終了時以外でもプロジェクト・ファイルの上書き保存により行うことができます)。
- 6) 次回、PM+ で sample.prj を読み込んで ID850QB を起動すると、プロジェクト・ファイル保存時のデバッグ環境が自動的に再現されます。

**注** ID850QB、および PM+ では、それぞれの環境情報をプロジェクト・ファイルに保存し参照します。ID850QB、および PM+ で扱うプロジェクト・ファイルの拡張子は prj です。なお、プロジェクト・ファイルで保存、再現される情報については、各製品のユーザーズ・マニュアルを参照してください。

## 4.4 オートロード

ID850QB を使用してデバッグを行っている際にバグなどを発見した場合、次の手順でソース・ファイルを修正することにより、コンパイルから再ダウンロードまでを自動的に実行することができます（「4.4.1 ソース修正によるオートロード」参照）。

また、ID850QB を起動した状態で、PM+ 上でコンパイル、およびリンク作業を行うことによっても、ロード・モジュールは ID850QB 上に再ダウンロードされます（「4.4.2 デバッガ起動によるオートロード」参照）。

**注意** PM+ で標準エディタ（idea-L）以外を使用する設定を行っている場合には、この処理を行うことはできません。

### 4.4.1 ソース修正によるオートロード

ソース修正によるオートロードは次の手順で行います。

- 1) 修正したいソース・ファイルを **ソース・テキスト・ウィンドウ** でオープンします。ID850QB で [ファイル] メニュー → [開く] を選択し、該当ファイルを指定してください（すでに、該当ファイルをソース・テキスト・ウィンドウ上にオープンしている場合は、そのウィンドウを最前面に表示します）。  
→該当ファイルがソース・テキスト・ウィンドウ上にオープンされます。
- 2) ID850QB で [編集] メニュー → [ソースの修正] を選択します。  
→エディタがオープンし、該当するソース・ファイルが読み込まれます。
- 3) エディタ上でソース・ファイルを修正します。
- 4) エディタを終了します。

**注意** ロード・モジュール・ファイルを自動的にダウンロードする際、CPU リセットは行いません。また、エディタの呼び出し時にオープンしていたデバッグ・ウィンドウと各イベント設定は復元されますが、ソース・ファイルの修正によって、以前使用していた行やシンボルがなくなった場合には、次のようになります。

- 変数表示していた変数は灰色表示になります。
- イベント条件は、イベント・マークが **黄色** 表示になります。
- ソフトウェア・ブレーク・ポイントは削除されます。

- 5) PM+ 上で [ビルド] メニュー → [ビルド → デバッグ]、または [ビルド] メニュー → [リビルド → デバッグ] を選択します。

## 4.4.2 デバッガ起動によるオートロード

ID850QB を起動した状態で、PM+ 上で次の操作を行った場合、自動的にロード・モジュールが ID850QB 上にダウンロードされます。

- PM+ 上で [ビルド] メニュー → [ビルド → デバッグ] を選択した場合
- PM+ 上で [ビルド] メニュー → [リビルド → デバッグ] を選択した場合

**参考** ダウンロード終了後、CPU リセットを行うかどうかの指定は PM+ の [ツール] メニューの [デバッガの設定 ...] で行います（デフォルトでは CPU リセットを行います）。

## 第5章 デバッグ機能

この章では、ID850QBのデバッグ機能に関して解説します。

表5-1 デバッグ機能一覧（デバッグ操作の流れ）

項目	参照先
デバッグ環境の設定	5.1 デバッグ環境の設定
ロード・モジュールのダウンロード	5.2 ダウンロード／アップロード機能
プログラムの表示、逆アセンブル結果表示	5.3 ソース表示／逆アセンブル表示機能
ブレークの設定	5.4 ブレーク機能
プログラムの実行	5.5 プログラム実行機能
変数値の確認	5.6 ウォッチ機能
メモリ内容の確認、編集	5.7 メモリ操作機能
レジスタ値の登録内容の確認、変更	5.8 レジスタ操作機能
実行時間の確認	5.9 タイマ機能 [IECUBE]
トレース・データの確認	5.10 トレース機能 [IECUBE]
カバレッジ測定結果の確認	5.11 カバレッジ測定機能 [IECUBE]
イベントの管理	5.12 イベント機能
RAM サンプリング	5.13 RRM 機能
DMM 機能	5.14 DMM 機能
デバッグ環境の保存、各ウインドウ状態の保存	5.15 ロード／セーブ機能
ジャンプ機能、トレース・ウインドウとの連結機能	5.16 ウインドウ共通機能

## 5.1 デバッグ環境の設定

デバッグ環境の設定に関する次の項目について解説します。

- [動作環境の設定](#)
- [オプションの設定](#)
- [マッピング設定](#)
- [外部メモリのアクセスに必要なレジスタの値の変更](#)

### 5.1.1 動作環境の設定

インサーキット・エミュレータの動作環境を設定するには、ID850QB 起動時に自動的にオープンする **コンフィギュレーション・ダイアログ** で行います。

すでにプロジェクト・ファイルが存在する場合には、<Project...> ボタンのクリックによりデバッグ環境の復元が可能です（「5.15.1 デバッグ環境（プロジェクト・ファイル）」参照）。

### 5.1.2 オプションの設定

下記の各種設定ダイアログで、デバッグに関する設定、およびインサーキット・エミュレータに関する設定等を行います。

- **コンフィギュレーション・ダイアログ**
- **拡張オプション設定ダイアログ**
- **フェイルセーフ・ブレイク設定ダイアログ**
- **RRM 設定ダイアログ**
- **フラッシュ・オプション設定ダイアログ**
- **データ・フラッシュ・オプション設定ダイアログ**
- **デバッグ・オプション設定ダイアログ**

### 5.1.3 マッピング設定

マッピング設定は **コンフィギュレーション・ダイアログ** で行います。

表 5 - 2 マッピング属性の種類

属性	意味
Emulation ROM [IECUBE] (メモリ・ボード搭載時)	エミュレーション ROM エミュレーション ROM に指定したメモリ領域は、対象デバイスに ROM を接続した場合と同等のメモリ領域となります。対象デバイスは、インサーキット・エミュレータ内のメモリに対してアクセスを行います。対象デバイスがこのメモリ領域に対して書き込みを行った場合、ライト・プロテクト・ブレイクが生じます。
Emulation RAM [IECUBE] (メモリ・ボード搭載時)	エミュレーション RAM エミュレーション ROM に指定したメモリ領域は、対象デバイスに RAM を接続した場合と同等のメモリ領域となります。対象デバイスは、インサーキット・エミュレータ内のメモリに対してアクセスを行います。
Target	ユーザ・エリア・マッピング ユーザ・エリア・マッピングに指定したメモリ領域はターゲット・システム上のメモリ、または CPU 内部に搭載されたメモリをアクセスする領域となります。
Target ROM [IECUBE]	ターゲット ROM ターゲット ROM に指定される領域は、フェイルセーフ・ブレイクのライト・プロテクトの対象となります（「 <b>フェイルセーフ・ブレイク設定ダイアログ</b> 」参照）。

属性	意味
I/O Protect	<p>I/O プロテクト領域 Target に指定した領域が、I/O プロテクト領域として指定可能です。 I/O プロテクト領域は、メモリ・ウインドウ上でマッピングされていない領域と同様に表示 (表示記号: ??) され、メモリ・ウインドウからこの領域に対する自由な読み込み／書き込みはできなくなります。このため、誤ったアクセスからの保護が可能になります。I/O プロテクト領域の値を読み込み／書き込みするには、IOR ウインドウ、またはウォッチ・ウインドウに登録する必要があります。</p>

#### 5.1.4 外部メモリのアクセスに必要なレジスタの値の変更

外部メモリに対してマッピングを行った際には、ダウンロードする前に、IOR ウインドウ、またはフック・プロセスージャを使用して、外部メモリのアクセスに必要なレジスタの値を変更する必要があります。

フック・プロセスージャを使用したレジスタ値の変更については、「7.9 フック・プロセスージャ」を参照してください。

なお、変更するレジスタに関しては、使用する品種のハードウェア・マニュアルを参照してください。

## 5.2 ダウンロード／アップロード機能

ID850QB では、表 5-3 ， 表 5-4 に示すファイルのダウンロード，およびアップロードが可能です。  
この項では次の項目について解説します。

- [ダウンロード](#)
- [アップロード](#)

## 5.2.1 ダウンロード

オブジェクト・ファイルのダウンロードは、[ダウンロード・ダイアログ](#)で行います。

デバッグ情報を持ったロード・モジュール・ファイルをダウンロードすることにより、自動的に該当ソース・テキスト・ファイル（[ソース・テキスト・ウインドウ](#)）が表示されます。

**参考** 複数のロード・モジュール・ファイルのダウンロードが可能です。

ロード済みファイルの確認は、[ファイル]メニュー→[ロードモジュール]の選択によりオープンする[ロード・モジュール一覧ダイアログ](#)で行います。

ダウンロード可能なファイル形式は次のとおりです。

表 5 - 3 ダウンロードが可能なファイルの種類

形式	拡張子
ロード・モジュール (ELF/CA850 (.out)) サード・パーティ・ロード・モジュール (ELF/GHS 拡張 DWARF2 (.out)) 注1	Load Module (*.out)
インテル・ヘキサ・フォーマット（標準, 拡張, 拡張リニア）注2	Hex Format (*.hex)
モトローラ・ヘキサ・フォーマット S タイプ	
拡張テック・ヘキサ・フォーマット (S0, S3, S7)	
バイナリ・データ	Binary Data (*.bin)
カバレッジ結果 [IECUBE]	Coverage (*.cvb)

注1 GHS 社製コンパイラ V3.5.1, V4.0.x に対応

注2 インテル・ヘキサ・フォーマットでは、アドレス 1M バイト以上のダウンロードが可能です。

**参考** \*.hex ファイルのフォーマットは自動判定されます。

## 5.2.2 アップロード

メモリ内容等のアップロードは、[アップロード・ダイアログ](#)で行います。

保存範囲の設定が可能です。

アップロード可能なファイル形式は次のとおりです。

表 5 - 4 アップロードが可能なファイルの種類

形式	拡張子
インテル・ヘキサ・フォーマット <sup>注</sup>	Hex Format (*.hex)
モトローラ・ヘキサ・フォーマット S タイプ (S0,S3,S7 -32 ビット・アドレス)	
拡張テック・ヘキサ・フォーマット	
バイナリ・データ	Binary Data (*.bin)
カバレッジ結果 [IECUBE]	Coverage (*.cvb)

**注** インテル・ヘキサ・フォーマットでは、標準（16 ビット・アドレス）、拡張（20 ビット・アドレス）、拡張リニア（32 ビット・アドレス）、アドレス 1M バイト以上のアップロードが可能です

**参考** \*.hex ファイルのフォーマットを指定して保存できます。

## 5.3 ソース表示／逆アセンブル表示機能

ソース・ファイルの表示は、[ソース・テキスト・ウインドウ](#)で行います。また、逆アセンブル表示、オンライン・アセンブルは[逆アセンブル・ウインドウ](#)で行います。

この項では次の項目について解説します。

- [ソース表示](#)
- [逆アセンブル表示](#)
- [混合表示モード（ソース・テキスト・ウインドウ）](#)
- [シンボル変換](#)

**参考** [ソース・テキスト・ウインドウ](#)、および[逆アセンブル・ウインドウ](#)上では、カバレッジ実行箇所の表示が可能です（「[5.11.3 カバレッジ実行箇所の表示](#)」参照）。

### 5.3.1 ソース表示

デバッグ情報を持ったロード・モジュール・ファイルをダウンロードすることにより、ソース・テキスト・ウィンドウ上に自動的に該当ソース・テキスト・ファイルが表示されます。

表示開始位置の変更は、[表示]メニュー→[移動...]の選択によりオープンするソース指定ダイアログで行います。

タブ・サイズや表示フォント等の表示に関する指定、およびソース・パスの指定は、デバッガ・オプション設定ダイアログで行います。検索は、<Search...> ボタンのクリックによりオープンするソース・サーチ・ダイアログで行います。検索結果はソース・テキスト・ウィンドウ上で反転表示されます。

表 5 - 5 表示可能なファイルの種類

ファイルの種類 (拡張子)	意味
Source (*.c, *.s)	ソース・ファイル (拡張子は、デバッガ・オプション設定ダイアログにて変更可能)
Text (*.txt)	テキスト・ファイル
All (*.*)	すべてのファイル

### 5.3.2 逆アセンブル表示

逆アセンブル表示は、逆アセンブル・ウィンドウで行います。

表示開始位置の変更は、[表示]メニュー→[移動...]の選択によりオープンするアドレス指定ダイアログで行います。

オフセット表示、およびレジスタ名表示の指定はデバッガ・オプション設定ダイアログで行います。

検索は、<Search...> ボタンのクリックによりオープンする逆アセンブル・サーチ・ダイアログで行います。検索結果は逆アセンブル・ウィンドウ上で反転表示されます。

### 5.3.3 混合表示モード（ソース・テキスト・ウインドウ）

ソース・テキスト・ウインドウでは、[表示]メニュー→[混合表示]を選択することにより、ソース・ファイルとあわせてプログラムの逆アセンブル表示が可能です。混合表示モードの表示内容は、表示ファイルとして保存可能です。

#### 通常表示

```

58      /* Timer Set */
59      TUM1 = 0x200;
60      CE1 = 1;
61      time_over = 0;

```

ソース・ファイルを表示するほか、一般的なテキスト・ファイルの内容が表示されます（デフォルト）。

#### 混合表示

```

58      /* Timer Set */
59      TUM1 = 0x200;
00000394      20660002      movea 0x200, r0, r12
00000398      606740f2      st.h r12, TUM1
60      CE1 = 1;
0000039c      c03f42f2      setl 0x7, TMC1
61      time_over = 0;
000003a0      440e0000      movhi 0x0, gp, r1
000003a4      61071184      st.w r0, -0x7hf0[r1]

```

表示するソース・ファイルの行にプログラム・コードが対応している場合、そのソース行に続いて逆アセンブル行が表示されます。逆アセンブル行では、アドレスのラベル、コード・データ、および逆アセンブルされたニモニックが表示されます（ニモニックの表示開始位置はタブ・サイズの設定値により調整されます）。

**注意 1** 混合表示モードは、ロード・モジュールがダウンロードされてシンボル情報が読み込まれている、かつ対応するソース・ファイルが表示されている場合のみ有効です。

**注意 2** 混合表示モード時にカーソル・キーによりスクロールを行った場合、余分なスクロールが発生することがあります。また、カーソル・キーでは、最終行までスクロールできない場合があります。

### 5.3.4 シンボル変換

シンボル変換ダイアログにより、指定した変数や関数のアドレス、およびシンボル値の表示が可能です。

シンボル変換は、ソース・テキスト・ウインドウ、逆アセンブル・ウインドウ上で変換したい文字列を選択し、コンテキスト・メニュー→[シンボル変換...]を選択することにより行います。

次表にシンボル指定方法を示します。

表 5 - 6 シンボルでの指定方法

変換の対象	指定方法
変数	var file#var (ファイル名を付けてスタティック変数を指定する場合) func#var (関数名を付けてスタティック変数を指定する場合) file#func#var (ファイル名, 関数名を付けてスタティック変数を指定する場合)
関数	func file#func (ファイル名を付けてスタティック関数を指定する場合)
ラベル	label file#label (ファイル名を付けてローカル・ラベルを指定する場合)
ソース・ファイルの行番号	file#no prog\$file#no
I/O ポート名	portname
I/O 名	I/O regname
レジスタ名	regname
PSW フラグ名	pswname

#### 備考 1 セパレータ "#"

ファイル名と変数、関数名、行番号とのセパレータとして使用します。

指定されたシンボルがスコープ内に見つからなかった場合、すべてのシンボル（スタティック変数、スタティック関数、ローカル・ラベル）を検索します。

#### 備考 2 セパレータ "\$"

複数のロード・モジュールを読み込んでいる場合に、ロード・モジュール名とファイル名、変数、関数名、およびシンボル名とのセパレータとして使用します。

デフォルトではシンボル名優先になっています。なお、一時的に優先順位を変えたい場合、シンボルの先頭に"\$"を付加することによりレジスタ名優先になります。

## 5.4 ブレーク機能

ブレーク機能とは、CPU によるユーザ・プログラムの実行、およびトレーサの動作を停止する機能です。この項では次の項目について解説します。

- [ブレークの種類](#)
- [ブレーク・ポイントの設定](#)
- [変数へのブレーク設定](#)
- [ハードウェア・ブレークとソフトウェア・ブレーク](#)
- [フェイルセーフ・ブレーク機能 \[IECUBE\]](#)

### 5.4.1 ブレークの種類

ブレークには次の種類があります。

表 5-7 主なブレークの種類

項目	内容
ハードウェア・ブレーク <sup>注1</sup> (イベント検出ブレーク)	設定されたブレーク・イベント条件を検出することにより、ユーザ・プログラムの実行を停止する機能。 →「5.4.2 ブレーク・ポイントの設定」参照
ソフトウェア・ブレーク <sup>注1</sup>	指定されたアドレスの命令をソフトウェア・ブレーク命令に書き換え、プログラム実行を停止する機能（「5.4.4 ハードウェア・ブレークとソフトウェア・ブレーク」参照）。 →「5.4.2 ブレーク・ポイントの設定」参照
[カーソル位置まで実行]によるブレーク <sup>注2</sup> (簡易ブレーク)	[実行]メニュー→[カーソル位置まで実行]の選択により実行されたユーザ・プログラムを、ソース・テキスト・ウィンドウ、または逆アセンブル・ウィンドウ上で指定されたアドレスを検出することにより停止する機能。
ステップ実行の条件成立によるブレーク	各コマンド（[ステップ・イン]、[ネクスト・オーバー]、[リターン・アウト]、[スローモーション]）の終了条件を満足することにより実行を停止する機能。
強制ブレーク	[実行]メニュー→[ストップ]、およびSTOPボタンの選択により実行を強制的に停止する機能。 すべての実行コマンドに対して有効です。
フェイルセーフ・ブレーク [IECUBE]	ユーザ・プログラムがメモリ、およびレジスタに対して禁止されていることを行った場合、強制的に実行を停止させる機能（「5.4.5 フェイルセーフ・ブレーク機能 [IECUBE]」参照）。 →「フェイルセーフ・ブレーク設定ダイアログ」参照
タイムアウト・ブレーク [IECUBE]	測定時間が指定されたタイムアウト時間を越えることにより、ユーザ・プログラムの実行を停止させる機能（「5.9 タイマ機能 [IECUBE]」参照）。 →「タイマ・ダイアログ」参照

**注1** [継続して実行]、[自動継続実行]、[カーソル位置から実行]、および[リスタート]実行に対して有効です（「表 5-9 実行の種類」参照）。

**注2** ユーザ・プログラムの実行停止後、この機能によるブレーク・ポイントは削除されます。  
この機能による実行中はカーソル位置以前に設定されているブレーク・イベントは発生しません。

## 5.4.2 ブレーク・ポイントの設定

ブレーク・ポイントは、ソース・テキスト・ウィンドウ、逆アセンブル・ウィンドウ上でワン・クリックすることにより、簡単に任意の場所への設定が可能です。

ブレーク・ポイントは、ブレーク・イベント条件として設定され、イベント機能を用いて管理されているため、設定数に制限があります（「5.12.4 各種イベント条件の有効イベント数」参照）。

### (1) ブレーク・ポイントの設定方法

ブレーク・ポイントは、' \* 'が表示されている行（プログラム・コードが存在している行）をクリックすることにより行います。

デフォルトではソフトウェア・ブレーク・ポイント（B）が設定されますが、コンテキスト・メニューの [ブレークポイント] を選択しクリックした場合には、ハードウェア・ブレーク・ポイント（B、またはB）が設定されます（「5.4.4 ハードウェア・ブレークとソフトウェア・ブレーク」参照）。

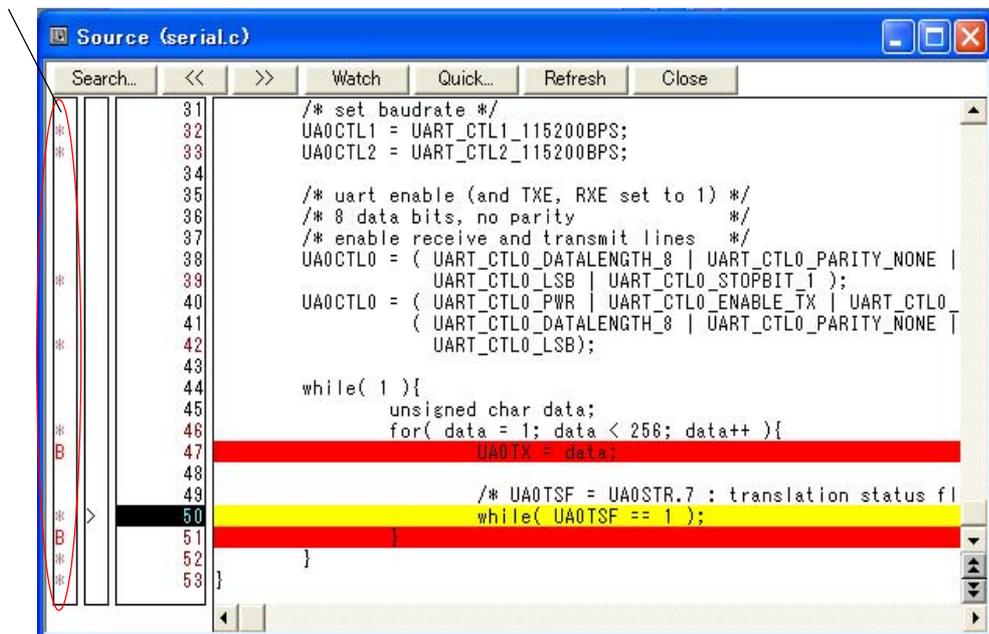
各種イベント、ブレークが設定されている行に対し設定を行った場合には、複数のイベント設定を示す 'A' が表示されます（「表 6 - 11 イベント設定状態」参照）。

**注意** 外部にマッピングした ROM エリアには、ソフトウェア・ブレーク・ポイントを設定／削除することはできません。

**参考** デフォルトで設定されるブレークは、拡張オプション設定ダイアログで変更可能です。

図 5 - 1 ブレーク・ポイントの設定

このエリア上の " \* "（プログラム・コード）をクリック



## (2) ブレーク・ポイントの削除方法

削除したいブレーク・ポイントの設定位置をクリックします。

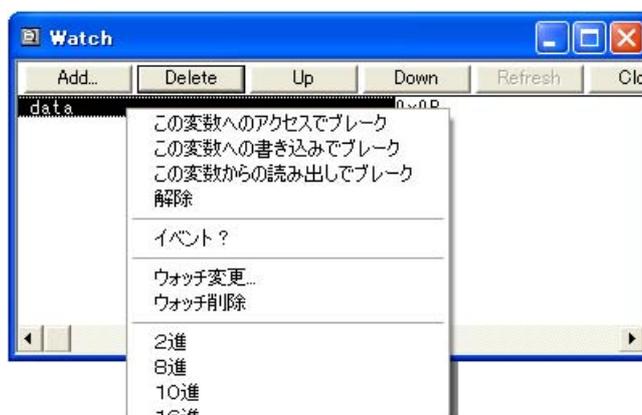
設定時と同様に、デフォルトではソフトウェア・ブレーク・ポイント (B) が削除されますが、コンテキスト・メニューの[ブレークポイント]を選択しクリックした場合には、ブレーク・ポイント(B, またはB)が削除されます。

削除の結果、他のイベントが残っていればそのイベントのマークが表示されます。

## 5.4.3 変数へのブレーク設定

ソース・テキスト・ウインドウ, ウォッチ・ウインドウでは、コンテキスト・メニューにより、変数へのアクセス・ブレーク設定 (アクセス・イベントを使用したブレーク・ポイントの設定) を簡単に行うことができます。

図 5 - 2 変数へのブレーク設定



## 5.4.4 ハードウェア・ブレークとソフトウェア・ブレーク

### (1) ハードウェア・ブレーク

ハードウェア・ブレークは、ハードウェアの資源を1つのイベント条件につき1つ使用し設定されるブレークです。

このため、ID850QB上では、ブレーク・イベント条件として「5.12 イベント機能」を用いて管理されています。

ブレーク・ポイントの有効数は品種により異なります（「5.12.4 各種イベント条件の有効イベント数」参照）。

### (2) ソフトウェア・ブレーク

ソフトウェア・ブレークは指定したアドレスの命令をソフトウェア・ブレーク命令に書き換えることにより設定されるブレークです。なお、外部ROM上への設定や変数のアクセス・タイミングでの停止などの指定はできません。

ソフトウェア・ブレーク・ポイントの有効数は以下のとおりです。

表 5-8 ソフトウェア・ブレークの有効数

接続 IE	有効数
[IECUBE]	2000
[MINICUBE] [MINICUBE2]	2000 <sup>注</sup>

**注** 内蔵ROM、および内蔵フラッシュ・メモリに対するソフトウェア・ブレークは、自動的にROMコレクション機能により設定されています。ROMコレクション機能により設定できる最大数は品種に依存します(4個)。

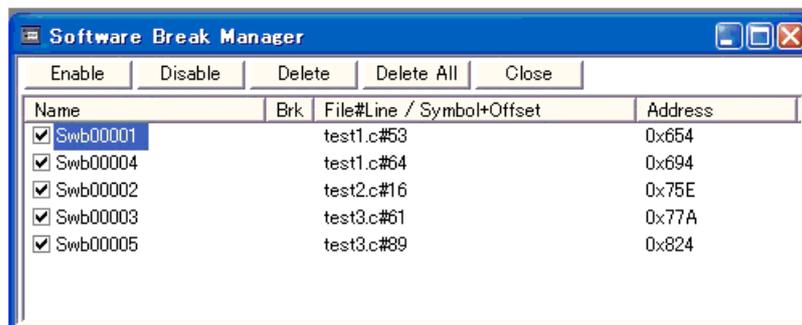
なお、ROMコレクション機能により設定されたソフトウェア・ブレークは、ターゲット・リセット、または内部リセットで一時的に無効になりますが、ブレークすると有効になります。

**注意** 疑似リアルタイム・モニタ機能 (Break When Readout) を有効にした場合、ソフトウェア・ブレーク・ポイントは設定できません。

また、有効なソフトウェア・ブレーク・ポイントを設定している場合には、ユーザ・プログラム実行中の書き込み (DMM 機能) はできません。[IECUBE]

ソフトウェア・ブレイクの管理は、ソフトウェア・ブレイク・マネージャで行います。

図 5 - 3 ソフトウェア・ブレイクの管理



### 5.4.5 フェイルセーフ・ブレイク機能 [IECUBE]

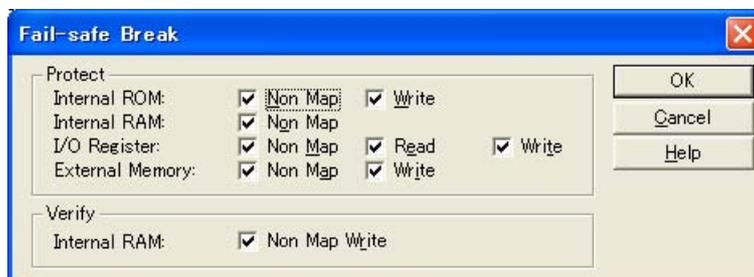
フェイルセーフ・ブレイクの設定は、フェイルセーフ・ブレイク設定ダイアログで行います。

チェック・ボックスによる個別の設定が可能です。

**参考** 内部 RAM 領域へのプロテクト設定は、ソフトウェアでのベリファイ処理により行っているため、ブレイク時に警告が表示されます。

IOR Illegal と内部 RAM のベリファイ・チェック時はステータスバーにアドレスが表示されます。

図 5 - 4 フェイルセーフ・ブレイク設定



## 5.5 プログラム実行機能

プログラム実行機能は、CPUによるユーザ・プログラムの実行とトレーサの動作を開始/停止する機能です。ユーザ・プログラムを実行することで、設定したブレーク・ポイント、または強制ブレークまでプログラム・カウンタ (PC) が進みます (「5.4 ブレーク機能」参照)。

**参考** ユーザ・プログラム実行中でも、トレース条件設定、およびタイマ条件設定が可能です (「[トレース・ダイアログ](#)」, 「[タイマ・ダイアログ](#)」参照)。**[IECUBE]**

ID850QBの実行機能には次の種類があり、操作は次図に示すツールバーの実行ボタン、または[実行]メニューにより行います。

図 5 - 5 実行ボタン



図 5 - 6 [実行]メニュー

実行(R)	イベント(N)	ブレイク(B)	シミュレーション(S)	ウィンドウ(W)	ヘルプ(H)
リスタート(R)		F4			
ストップ(S)		F2			
継続して実行(G)		F5			
ブレークせずに実行(P)		Ctrl+F5			
リターンアウト(E)		F7			
ステップイン(I)		F8			
ネクストオーバー(O)		F10			
カーソル位置から実行(A)		Shift+F6			
カーソル位置まで実行(M)		F6			
自動継続実行(Q)					
スローモーション(W)					
CPUバースト(B)		F3			

表 5 - 9 実行の種類

項目	内容
[リスタート] 	CPU リセット後、0 番地よりユーザ・プログラムを実行します。 ユーザ・プログラム実行前に CPU をリセットして [継続して実行] を実行した場合と同じ動作です。
[継続して実行] 	現在の PC レジスタで示されるアドレスからユーザ・プログラムを実行し、設定されているブレーク・イベント条件が成立するとユーザ・プログラムの実行を停止します。
[ブレークせずに実行] 	現在の PC レジスタで示されるアドレスからユーザ・プログラムを実行し、設定されているブレーク・ポイントを無視してユーザ・プログラムを実行します。
[リターン・アウト] 	呼び出し関数に戻るまでユーザ・プログラムを実行します。 C 言語で記述した関数が対象となります。
[ステップ・イン] 	ソース・モードの場合、現在の PC レジスタ値からソース・テキストの 1 行分をステップ実行し、各ウインドウの内容を更新します。 命令モードの場合、現在の PC レジスタ値から 1 命令を実行し、各ウインドウの内容を更新します。
[ネクスト・オーバー] 	jarl 命令によって呼び出された関数やサブルーチンを、1 ステップとみなしたネクスト・ステップ実行をします (jarl 命令を実行したときと同じネストになるまで、ステップ実行します)。jarl 命令以外の場合、[ステップ・イン] 実行時と同じ動作です。
[カーソル位置から実行]	指定したアドレスからユーザ・プログラムを実行し、設定されているブレーク・イベント条件が成立するとユーザ・プログラムの実行を停止します。
[カーソル位置まで実行]	現在の PC レジスタで示されるアドレスから、 <b>ソース・テキスト・ウインドウ</b> 、または <b>逆アセンブル・ウインドウ</b> 上の行/アドレス表示エリア内で選択されたアドレスまでユーザ・プログラムを実行し、ブレークします。 なお、この選択によるユーザ・プログラムの実行中は、現在設定されているブレーク・イベントは発生しません。
[自動継続実行]	現在の PC レジスタで示されるアドレスからユーザ・プログラムを実行し、設定されているブレーク・イベント条件が成立するとユーザ・プログラムを一旦停止し、各ウインドウの内容を更新したのち、再び停止しているアドレスからユーザ・プログラムを実行します。ユーザが [ストップ] を行うまで、この動作を繰り返します。
[スローモーション]	現在の PC レジスタ値で示されるアドレスからソース・モードの場合は 1 行分、命令モードの場合は 1 命令分のステップ実行を行い、そのつど各ウインドウの内容を更新します。ユーザが [ストップ] を行うまで、この動作を繰り返します。
[CPU リセット] 	CPU をリセットします。
[ストップ] 	プログラム実行を強制的に停止させます。

## 5.6 ウォッチ機能

この項では、ウォッチ機能に関する次の項目について説明します。

- [データ値の表示, 変更](#)
- [ローカル変数値の表示, 変更](#)
- [ウォッチ・データの登録, 削除](#)
- [ウォッチ・データの変更](#)
- [データ値の一時的表示, 変更](#)
- [バルーン・ウォッチ機能](#)
- [スタック・トレース表示機能](#)

### 5.6.1 データ値の表示, 変更

データ値の表示, 変更はウォッチ・ウィンドウで行います。

ウォッチ・データを登録することにより, データ値の推移が確認できます。

表示形式の指定はデバッガ・オプション設定ダイアログで行います。

図5-7 ウォッチ・ウィンドウ

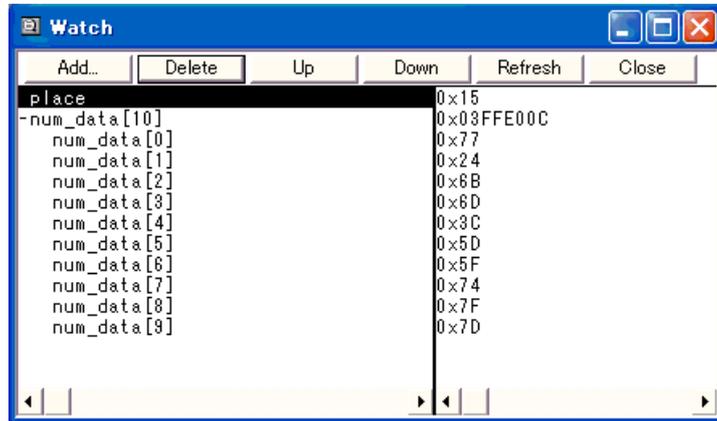


図5-8 ウォッチ表示形式の指定 (デバッガ・オプション設定ダイアログ)



### 5.6.2 ローカル変数値の表示, 変更

ローカル変数値の表示, 変更はローカル変数ウィンドウで行います。

このウィンドウでは, カレント関数内のローカル変数を自動的に表示します (変数の追加/削除は不可)。

図5-9 ローカル変数ウィンドウ



### 5.6.3 ウォッチ・データの登録, 削除

ウォッチ・ウィンドウへのデータ登録は、ソース・テキスト・ウィンドウ、または逆アセンブル・ウィンドウ上から可能です。各ウィンドウ上で変数やシンボル名を選択し <Watch> ボタンをクリックすることにより簡単に行えます。また、以下の方法でも登録可能です。

- 選択した変数やシンボル名を直接ウォッチ・ウィンドウ上にドロップ(「5.16.4 ドラッグ&ドロップ機能」参照)
- クイック・ウォッチ・ダイアログ、またはウォッチ登録ダイアログで <Add> ボタンをクリック

ウォッチ・データの削除は、変数名やシンボル名をクリックし (Shift キー, Ctrl キーによる複数選択も可能), <Delete> ボタンをクリックすることにより行います。ただし、配列の要素や構造体、共用体のメンバなど開かれた階層の行は削除することはできません。

### 5.6.4 ウォッチ・データの変更

ウォッチ・データの変更はウォッチ変更ダイアログで行います。

なお、シンボル名を変更した結果、すでに同名のシンボル名のデータが存在している場合も変更を許可します。

図 5 - 10 ウォッチ変更ダイアログ



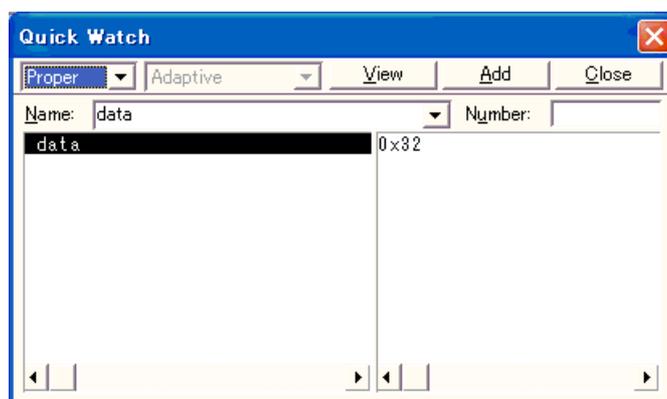
### 5.6.5 データ値の一時的表示, 変更

データ値の一時的表示, 変更にはクイック・ウォッチ・ダイアログを 사용합니다。

ソース・テキスト・ウィンドウ, または逆アセンブル・ウィンドウ上で変数やシンボル名を選択し <Quick...> ボタンをクリックすることにより, ウォッチ・データの登録が可能です。

このウィンドウ上では, 表示進数, 表示サイズ, および表示個数の変更が可能です。

図 5 - 11 クイック・ウォッチ・ダイアログ



### 5.6.6 バルーン・ウォッチ機能

ソース・テキスト・ウィンドウ, または逆アセンブル・ウィンドウ上では, 選択した変数上にマウス・カーソルを重ねることにより, 変数値がポップアップされます。

### 5.6.7 スタック・トレース表示機能

スタック・トレース・ウィンドウでは, 現在のユーザ・プログラムのスタック内容を表示します。

図 5 - 12 スタック・トレース・ウィンドウ



## 5.7 メモリ操作機能

この項では、メモリ操作に関する次の項目について解説します。

なお、ベリファイ・チェックの指定等は、[拡張オプション設定ダイアログ](#)で行います。

- [メモリ内容の表示, 変更](#)
- [アクセス・モニタ機能 \[IECUBE\]](#)
- [メモリ内容の初期化, コピー, 比較](#)
- [フラッシュ・メモリへの書き込み機能 \[MINICUBE\] \[MINICUBE2\]](#)

### 5.7.1 メモリ内容の表示, 変更

メモリ・ウィンドウ上では、ニモニック・コード、16進コード、およびASCIIコードを使用してメモリの内容の表示、および変更が可能です。検索は<Search...> ボタンのクリックによりオープンするメモリ・サーチ・ダイアログで行います。検索結果はメモリ・ウィンドウ上で反転表示されます。

表示開始位置の変更は、[表示]メニュー→[移動]の選択によりオープンするアドレス指定ダイアログで行います。

なお、プログラム実行中でもサンプリング範囲に割り当てられている変数、およびデータなどはリアルタイムに表示可能です（「5.13 RRM 機能」参照）。

### 5.7.2 アクセス・モニタ機能 [IECUBE]

アクセス・モニタ機能とは、RRM 機能のサンプリング範囲に対するアクセス状況（Read, Write, Read および Write）を色によりメモリ・ウィンドウ上に表示する機能です。

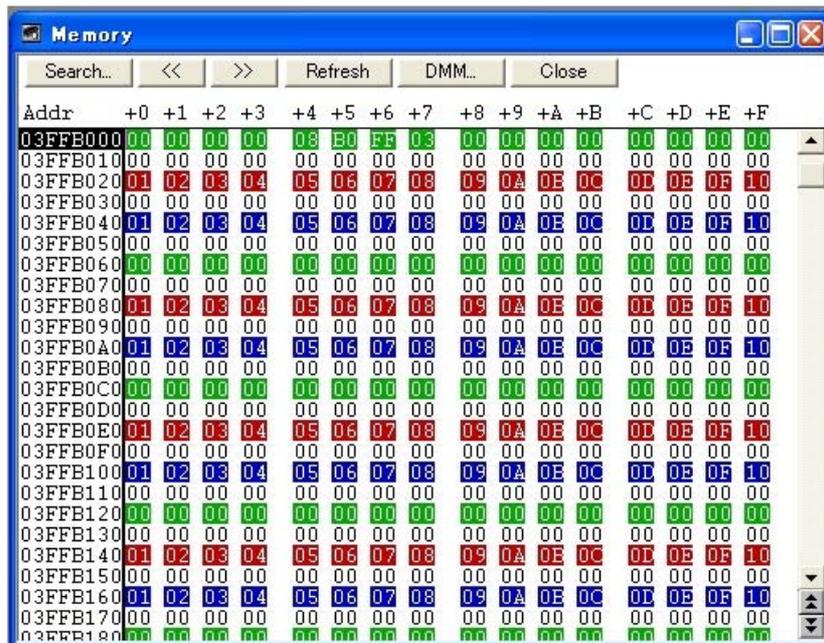
表示単位が "Byte" の場合のみアクセス・モニタ表示が可能です。また、アスキー表示エリアには色がつきません。

なお、[表示]メニュー→[アクセス状況表示]の選択により、アクセス状況の累積表示設定、およびアクセス状況表示のクリアが可能です。

**注意 1** プログラム実行中に DMA 経由で書き換えたメモリの値、およびデバッガから書き換えたメモリの値は、アクセス・モニタに表示できません。

**注意 2** この機能は、[オプション]メニュー→[RRM 機能]選択時のみ有効です。

図 5 - 13 アクセス・モニタ機能（メモリ・ウィンドウ）



### 5.7.3 メモリ内容の初期化, コピー, 比較

メモリ内容の初期化, コピー, 比較は, [編集]メニュー→[メモリ]→[初期化 ... / 複写 ... / 比較 ...] を選択することによりオープンする[メモリ・フィル・ダイアログ](#), [メモリ・コピー・ダイアログ](#), [メモリ比較ダイアログ](#)で行います。比較結果は[メモリ比較結果ダイアログ](#)に表示されます。

### 5.7.4 フラッシュ・メモリへの書き込み機能 [MINICUBE] [MINICUBE2]

ID850QB では, 通常のメモリ操作と同様のアクセス方法により, 内蔵フラッシュ・メモリへの書き込み, およびロード・モジュールのダウンロードが可能です。

[メモリ・ウインドウ](#), [逆アセンブル・ウインドウ](#), [ウォッチ・ウインドウ](#), [メモリ・フィル・ダイアログ](#), [メモリ・コピー・ダイアログ](#)からは, 内蔵フラッシュ・メモリ上のデータであることを意識することなく変更が可能です。また, 内蔵フラッシュ・メモリへのダウンロードは, [フラッシュ・セルフ・プログラミング機能](#)を利用することで実現しています ([「フラッシュ・オプション設定ダイアログ」](#), [「データ・フラッシュ・オプション設定ダイアログ」](#) 参照)。

**注意** ユーザ・プログラム実行中は内蔵フラッシュ・メモリへの書き込みはできません。

**参考** ID850QB では, 内蔵フラッシュ・メモリへのダウンロードを行ったあと余った領域の内容を消去します。

## 5.8 レジスタ操作機能

この項では、レジスタ操作に関する次の項目について解説します。

- [レジスタ内容の表示, 変更](#)
- [周辺 I/O レジスタ内容の表示, 変更](#)
- [I/O ポート内容の表示, 変更](#)

### 5.8.1 レジスタ内容の表示, 変更

レジスタ内容はレジスタ・ウィンドウで表示, 変更可能です。

レジスタ名称の表示切り替え (機能名/絶対名) は, デバッガ・オプション設定ダイアログで可能です。

**参考** 表示レジスタの選択は, レジスタ選択ダイアログで行います。

図5-14 機能名/絶対名の切り替え



### 5.8.2 周辺 I/O レジスタ内容の表示, 変更

周辺 I/O レジスタ内容は IOR ウィンドウで表示, 変更可能です。

表示開始位置の変更は, [表示]メニュー→[移動...]の選択によりオープンするアドレス指定ダイアログで行います。

表示レジスタの選択は, IOR 選択ダイアログで行います。

図5-15 周辺 I/O レジスタ内容の表示

The screenshot shows a window titled "IOR" with a table of I/O registers. The table has four columns: Name, Attribute, Address, and Value. The registers listed include PDL, PDLH, PDH, PCS, PCT, PCM, PCD, PMDL, PMDLL, PMDLH, PMDH, PMCS, PMCT, PMCM, PMCD, PMCDL, PMCDLH, PMCDLH, PMCDH, PMCCS, PMCCT, PMCCM, BSC, and WSWC.

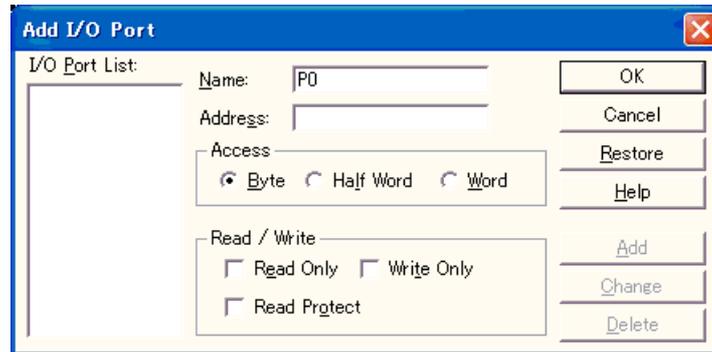
Name	Attribute	Address	Value
PDL	R/W 16	03FFF004	0000
PDLL	R/W 1.8	03FFF004	00
PDLH	R/W 1.8	03FFF005	00
PDH	R/W 1.8	03FFF006	00
PCS	R/W 1.8	03FFF008	00
PCT	R/W 1.8	03FFF00A	00
PCM	R/W 1.8	03FFF00C	00
PCD	R/W 1.8	03FFF00E	00
PMDL	R/W 16	03FFF024	FFFF
PMDLL	R/W 1.8	03FFF024	FF
PMDLH	R/W 1.8	03FFF025	FF
PMDH	R/W 1.8	03FFF026	FF
PMCS	R/W 1.8	03FFF028	FF
PMCT	R/W 1.8	03FFF02A	FF
PMCM	R/W 1.8	03FFF02C	FF
PMCD	R/W 1.8	03FFF02E	FF
PMCDL	R/W 16	03FFF044	0000
PMCDLH	R/W 1.8	03FFF044	00
PMCDLH	R/W 1.8	03FFF045	00
PMCDH	R/W 1.8	03FFF046	00
PMCCS	R/W 1.8	03FFF048	00
PMCCT	R/W 1.8	03FFF04A	00
PMCCM	R/W 1.8	03FFF04C	00
BSC	R/W 16	03FFF066	5555
WSWC	R/W 8	03FFF06E	77

### 5.8.3 I/O ポート内容の表示, 変更

ユーザ定義の I/O ポートは、**I/O ポート追加ダイアログ**で登録することにより、**IOR ウィンドウ**で表示、変更が可能になります。

なお、プログラマブル I/O レジスタに対応している品種では、**コンフィギュレーション・ダイアログ**で、プログラマブル I/O 領域使用の設定を行うことにより、プログラマブル I/O レジスタ内容の表示、変更が可能です。

図 5 - 16 I/O ポートの登録



## 5.9 タイマ機能 [IECUBE]

タイマ機能とは、ユーザ・プログラムの実行開始からブレークまでの実行時間 (Run-Break 時間)、またはタイマ・イベントを使用して、ユーザ・プログラム内の特定区間の実行時間を測定する機能です。

ID850QB のタイマ機能は、外部クロックを使用して測定しています。このため、測定可能時間は、「表 6 - 7 分周率と最大測定時間の関係 (タイマ・カウンタ (タイマ))」での設定により異なります。

なお、Run-Break 時間は、メイン・ウィンドウのステータスバーに表示されます。

この項では次の項目について解説します。

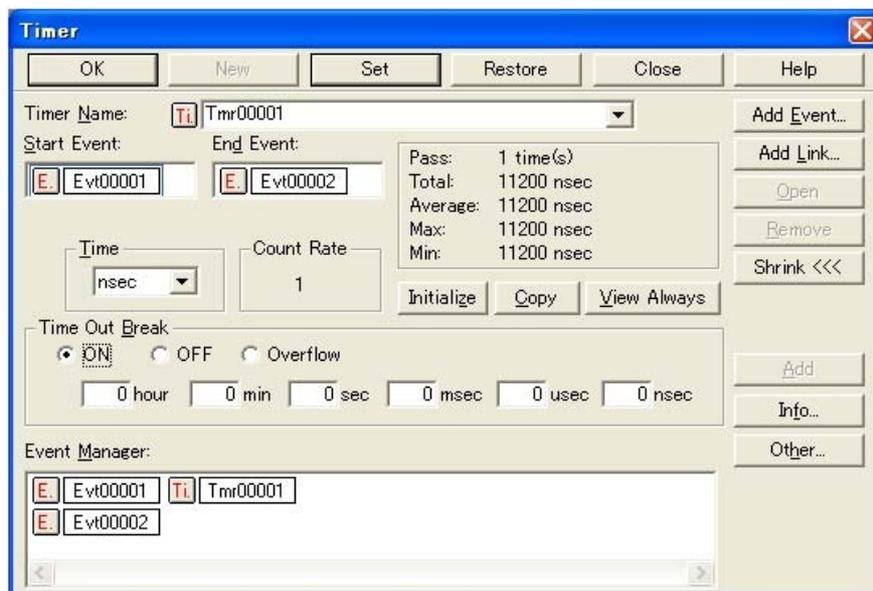
- タイマ・イベント条件
- Run-Break イベント

## 5.9.1 タイマ・イベント条件

時間測定を開始／終了するトリガを指定したものがタイマ・イベント条件です。タイマ・イベント条件は**タイマ・ダイアログ**上で設定します（「5.12 イベント機能」参照）。

ID850QB では、Time Out Break エリアで、タイムアウト・ブレークの設定が可能です。

図5-17 タイマ・イベントの設定，表示（タイマ・ダイアログ）



<View Always> ボタンにより、**タイマ測定結果ダイアログ**として常時表示しておくことが可能になります。

なお、プログラム実行中のタイマに対する操作は、[実行]メニュー→[タイマ開始／停止]により行います。

## 5.9.2 Run-Break イベント

Run-Break イベントとは、ユーザ・プログラムの実行開始からブレークまでの実行時間（Run-Break 時間）を測定するタイマ・イベント条件につけられたタイマ・イベント名です。あらかじめ登録されており、**タイマ・ダイアログ**で指定することにより、Run-Break 時間の表示が可能です。

なお、Run-Break 時間は、**メイン・ウィンドウ**のステータスバーにも表示されます。

Run-Break イベントは、同時に有効にできるタイマ・イベント数（「5.12.4 各種イベント条件の有効イベント数」参照）に含まれないため、タイマ・イベント条件の有効数にプラスして使用可能です。

## 5.10 トレース機能 [IECUBE]

トレース機能とは、ユーザ・プログラムの実行過程を示すデータの履歴をトレース・メモリに保存する機能です。なお、DMA の開始点、および終了点はトレース条件の如何に関わらず必ずトレースされます（「5.10.7 DMA ポイント・トレース機能」参照）。

この項では次の項目について解説します。

- トレース・メモリ
- トレース・データの設定
- トレース・データの確認
- 混合表示モード（トレース・ウインドウ）
- トレーサの動作
- 条件トレースの設定
- DMA ポイント・トレース機能

**注意** RRM 機能／トレース機能 [IECUBE] ／カバレッジ測定機能 [IECUBE] は排他使用の機能です（「5.11.4 RRM 機能／トレース機能／カバレッジ機能間の排他使用に関して」参照）。

各機能の切り替えは、[オプション]メニューで行います。

### 5.10.1 トレース・メモリ

ID850QB はリング構造のトレース・メモリを持っており、サイズ指定は**拡張オプション設定ダイアログ**で行います。

トレース・メモリの最大容量は以下のとおりです。

表 5 - 10 トレース・メモリ・サイズ

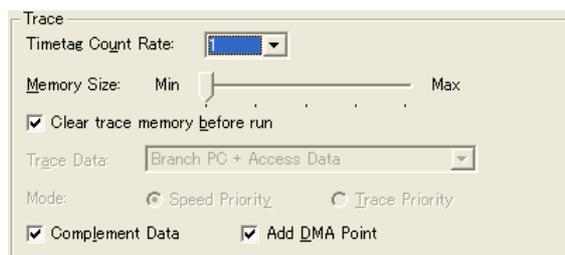
接続 IE	最大値
[IECUBE]	256K バイト

### 5.10.2 トレース・データの設定

採取するトレース・データの詳細設定は、**拡張オプション設定ダイアログ**で行います。

補完モードの選択（Complement Data エリアのチェック）により、ハードウェアではトレースできない分岐命令間の命令の補完表示が可能です。補完モード時は、ユーザ・プログラム実行中（トレーサ停止中）に内部 ROM 領域のアセンブル表示が可能です。

図 5 - 18 トレース・データの設定



### 5.10.3 トレース・データの確認

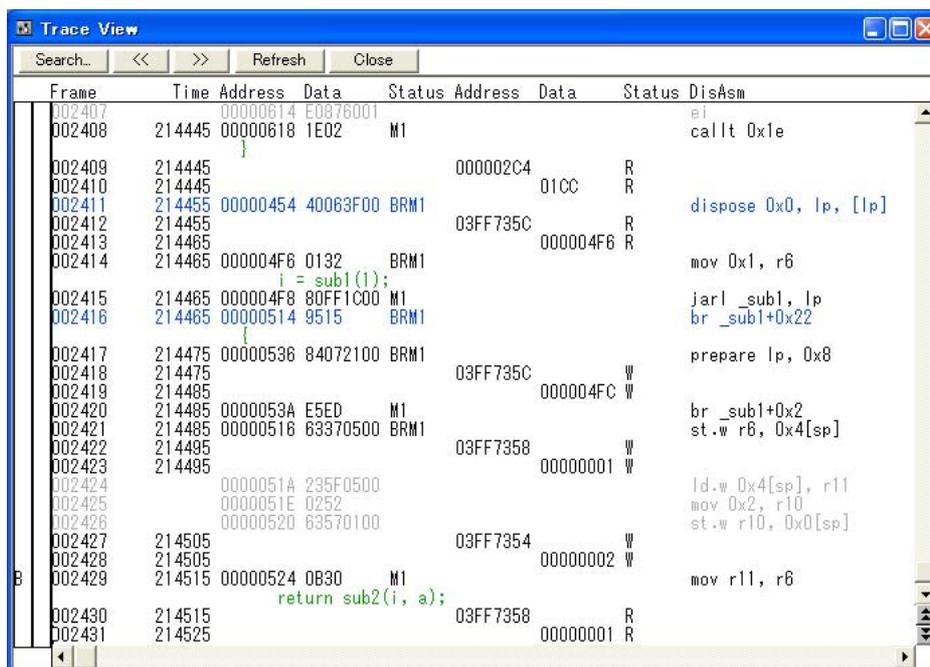
トレース・メモリに保存されたトレース・データはトレース・ウィンドウで確認できます。<Search...> ボタンのクリックによりオープンするトレース・サーチ・ダイアログで検索が可能です。

表示開始位置の変更は、[表示]メニュー→[移動]の選択によりオープンするフレーム指定ダイアログで行います。

なお、トレース・ウィンドウ内の表示項目はトレース表示選択ダイアログで選択可能です。

また、トレース・ウィンドウのタイム・タグ表示で使用するカウンタの分周率は、拡張オプション設定ダイアログで設定します。タイムタグ・カウンタ、カウント・レート、タイムタグ積算の設定は拡張オプション設定ダイアログで行います。

図 5 - 19 トレース・データの確認



### 5.10.4 混合表示モード（トレース・ウインドウ）

トレース・ウインドウでは、[表示]メニュー→[混合表示]を選択することにより、トレース結果と合わせてソース・ファイルの表示を行うことができます（混合表示モード）。

表示するプログラム・フェッチ・アドレスの行にプログラム・コードが対応している場合、そのトレース結果行の前にソース・ファイル行が表示されます。

ソース・ファイル行は、表示色が変更され緑色で強調表示されます。

Frame	Time	Address	Data	Status	Address	Data	Status	DisAsm
32757	3	000005A2	85058505	BRM1				br _main+0x21a
			while(1);					
32758	3	000005A2	85058505	BRM1				br _main+0x21a
			while(1);					
32759	3	000005A2	85058505	BRM1				br _main+0x21a
			while(1);					
32760	3	000005A2	85058505	BRM1				br _main+0x21a
			while(1);					

**注意** 混合表示モードは、ロード・モジュールがダウンロードされ、シンボル情報が読み込まれている時、フェッチ・アドレス、フェッチ・データ、フェッチ・ステータス、または逆アセンブル結果のいずれかが表示されている場合のみ有効です。

### 5.10.5 トレーサの動作

ユーザ・プログラムの実行形態、およびトレーサ制御モードでの設定により、トレーサの動作は次のように異なります。

なお、プログラム実行中のトレーサに対する操作は、[実行]メニュー→[トレーサ開始/停止]により行います。

#### (1) 実行時の動作

[実行]メニュー→[無条件トレース/条件トレース]の選択により、次のようにトレーサの動作が異なります。

表 5 - 11 トレースの種類

項目	内容
無条件トレース	ユーザ・プログラムの実行でトレースを開始し、ブレークの発生によりトレースを終了します。 この際、設定されているトレース・イベント条件は無視されます。
条件トレース	トレース・ダイアログで設定した条件によりトレースを開始/終了します（「5.10.6 条件トレースの設定」参照）。 ただし、その間にブレークが発生した場合は、その時点でトレースを終了します。

**(2) ステップ・イン実行時の動作**

1ステップ実行ごとにトレーサが動作し、1ステップ分のトレース・データが逐次トレース・メモリに追加されます。

**(3) ネクスト・オーバー実行時の動作**

ネクスト・オーバー実行の対象となる命令により、トレーサの動作が異なります。

**(a) jarl disp22,[lp] 命令**

jarl 命令と呼び出されたサブルーチンの内部のトレースを行います。

**(b) 上記以外の命令**

ステップ・イン実行時と同じ動作をします。

**(4) トレーサ制御モード**

トレーサ制御モードとして次の種類があります。設定は [ 実行 ] メニューから行います。

表 5 - 12 トレーサ制御モードの種類

モード	内容
ノン・ストップ	トレース・メモリを一周したら最古のフレームからオーバーライトします (デフォルト)。
フル・ストップ	トレース・メモリを一周したらトレーサを停止します。
フル・ブレーク	トレース・メモリを一周したらトレーサとプログラム実行を停止します。
ディレイ・トリガ・ストップ	ディレイ・トリガ・イベント発生時にディレイ・カウント・フレーム分トレースを取り、トレーサを停止します。

### 5.10.6 条件トレースの設定

トレース実行の開始／終了を示すトリガとなるイベント条件を使用して行うトレースが条件トレースです。

条件トレースの設定は、[トレース・ダイアログ](#)上でトレース・イベント条件を設定することにより行います（[「5.12 イベント機能」](#)参照）。

条件トレースを行うには、[実行]メニュー→[条件トレース]を有効にする必要があります。

条件トレースには次の種類があります。

表 5 - 13 条件トレースの種類

項目	内容, 設定方法
セクション・トレース	指定した条件と条件の間（特定の区間）だけ行うトレースです。 <a href="#">トレース・ダイアログ</a> でトレース開始イベントとトレース終了イベント条件を設定し、[実行]メニュー→[条件トレース]を有効にすることにより、セクション・トレースを行うことができます。
クオリファイ・トレース	条件が成立したそのときだけ行うトレースです。 クオリファイ・トレース・イベントとして複数のイベントを設定した場合には、それぞれのイベントが成立するごとにトレースを行います。 <a href="#">トレース・ダイアログ</a> でクオリファイ・トレース・イベント条件を設定し、[実行]メニュー→[条件トレース]を有効にすることにより、クオリファイ・トレースを行うことができます。
ディレイ・トリガ・トレース	条件が成立したその後、ディレイ・カウント分だけ行うトレースです。 <a href="#">トレース・ダイアログ</a> でディレイ・トリガ・イベント条件を設定し、 <a href="#">ディレイ・カウント設定ダイアログ</a> でディレイ・カウントを設定後、[実行]メニュー→[条件トレース]を有効にすることにより、ディレイ・トリガ・トレースを行うことができます。

### 5.10.7 DMA ポイント・トレース機能

DMA ポイント・トレース（Direct Memory Access Trace）は、通常のトレースより優先して行われます。

DMA ポイント・トレースによるアクセス・フレームは、[トレース・ウインドウ](#)に "M" マークが表示されます。

DMA ポイント・トレースの動作は以下のようになります。

#### (1) 無条件トレース時

通常の全トレースに加え、DMA の開始点と終了点は必ずトレースされます。

#### (2) セクション・トレース時

通常のセクション・トレースに加え、DMA の開始点と終了点は、セクション・トレースの区間外でも必ずトレースされます。

#### (3) クオリファイ・トレース時

通常のクオリファイ・トレースに加え、DMA の開始点と終了点は、クオリファイ・トレース条件未成立時にも必ずトレースされます。

## 5.11 カバレッジ測定機能 [IECUBE]

カバレッジ測定にはいくつかの種類がありますが、ID850QB では C0 カバレッジのコード・カバレッジ測定を行います。

C0 カバレッジ（命令網羅率）：コード内の全てのステートメントを少なくとも 1 回は実行

カバレッジ測定結果（カバレッジ・データ）のダウンロード／アップロードは、それぞれ[ダウンロード・ダイアログ](#)、[アップロード・ダイアログ](#)で行います。

この項では次の項目について解説します。

- [カバレッジ測定結果の表示](#)
- [カバレッジ測定範囲](#)
- [カバレッジ実行箇所の表示](#)
- [RRM 機能／トレース機能／カバレッジ機能間の排他使用に関して](#)

**注意 1** カバレッジ・ボード非搭載時には、この機能は無効です。

**注意 2** [RRM 機能／トレース機能 \[IECUBE\]](#)／[カバレッジ測定機能 \[IECUBE\]](#) は排他使用の機能です（「[5.11.4 RRM 機能／トレース機能／カバレッジ機能間の排他使用に関して](#)」参照）。

各機能の切り替えは、[\[オプション\]](#)メニューで行います。

### 5.11.1 カバレッジ測定結果の表示

カバレッジ測定結果は、[コード・カバレッジ・ウィンドウ](#)で確認できます。

[コード・カバレッジ・ウィンドウ](#)では、関数／セクション／割り込みハンドラ（ベクタ）別に測定結果が表示されます。なお、測定結果の更新はブレーク時に行われます（ユーザ・プログラム実行中の自動更新は行いません）。

[コード・カバレッジ・ウィンドウ](#)上の表示データは、CSV 形式での保存が可能です（「[5.15.2 ウィンドウの表示情報（表示ファイル）](#)」参照）。なお、選択タブごとに保存内容は変わります。

カバレッジ・データのクリアは、[\[オプション\]](#)メニューより行います。

図 5 - 20 カバレッジ測定結果の表示

The screenshot shows a window titled "Code Coverage" with a "Load Module:" dropdown set to "romp.out" and a "Total Coverage(%):" field showing "92.3". Below this is a table with columns: Name, Type, Status, Address, Size, Fetch, and Coverage(%). The table lists various functions and their coverage percentages.

Name	Type	Status	Address	Size	Fetch	Coverage(%)
RESET	nonmaskable	use	0	4	4	100.0
INTTP0CC0	maskable	use	0x170	4	4	100.0
NMI	nonmaskable	use	0x10	4	0	0.0
INTWDT2	nonmaskable	use	0x20	4	0	0.0
TRAP00	software	use	0x40	4	0	0.0
TRAP01	software	nonuse	0x40	8	0	0.0
TRAP02	software	nonuse	0x40	8	0	0.0
TRAP03	software	nonuse	0x40	8	0	0.0
TRAP04	software	nonuse	0x40	8	0	0.0
TRAP05	software	nonuse	0x40	8	0	0.0
TRAP06	software	nonuse	0x40	8	0	0.0
TRAP07	software	nonuse	0x40	8	0	0.0
TRAP08	software	nonuse	0x40	8	0	0.0
TRAP09	software	nonuse	0x40	8	0	0.0
TRAP0A	software	nonuse	0x40	8	0	0.0
TRAP0B	software	nonuse	0x40	8	0	0.0
TRAP0C	software	nonuse	0x40	8	0	0.0
TRAP0D	software	nonuse	0x40	8	0	0.0
TRAP0E	software	nonuse	0x40	8	0	0.0
TRAP0F	software	nonuse	0x40	8	0	0.0
TRAP10	software	use	0x50	4	0	0.0

### 5.11.2 カバレッジ測定範囲

カバレッジ測定範囲は以下のとおりです。

表 5 - 14 コード・カバレッジ測定範囲

接続 IE	コード・カバレッジ測定範囲
[IECUBE]	内蔵 ROM 空間 + 任意の 1M バイト（ <a href="#">カバレッジ範囲選択ダイアログ</a> で選択）

### 5.11.3 カバレッジ実行箇所の表示

ユーザ・プログラム内のカバレッジ実行／未実行箇所は、カバレッジ測定情報を元にソース・テキスト・ウィンドウ、逆アセンブル・ウィンドウ上に表示されます。

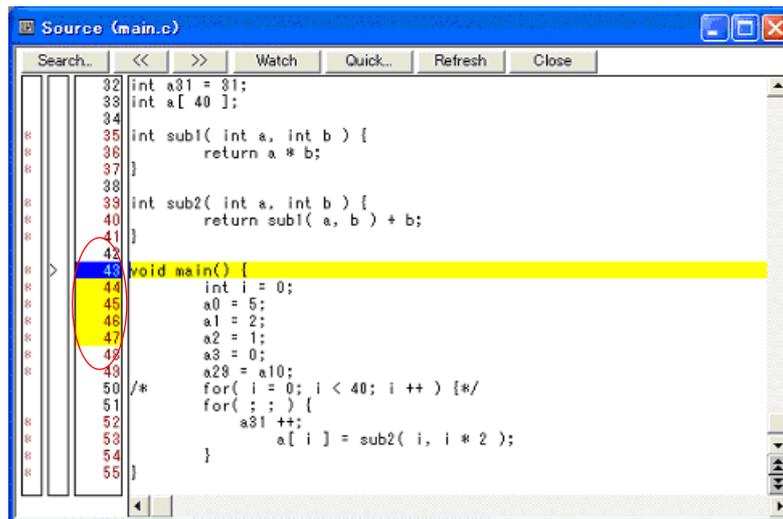
また、表示結果は各ソース・テキスト・ウィンドウ、逆アセンブル・ウィンドウの表示ファイルとして保存可能です（「5.15.2 ウィンドウの表示情報（表示ファイル）」）。

カバレッジ実行した行番号、またはアドレスの背景色が以下ようになります。表示ファイル内では、背景色が以下のマークに置き換えられ行番号、またはアドレスの横に付加されます。

表 5 - 15 カバレッジ実行／未実行箇所の表示形式

カバレッジ	ソース・テキスト・ウィンドウ		逆アセンブル・ウィンドウ	
	背景色	マーク	背景色	マーク
この行のコードを 100% で実行	黄色	'@'	黄色	'@'
この行のコードを 1 - 99% で実行	橙色	'+'	-	-
この行のコードを 0% 実行（未実行）	無色	-	無色	-

図 5 - 21 カバレッジ実行／未実行箇所の表示



#### 5.11.4 RRM 機能／トレース機能／カバレッジ機能間の排他使用に関して

RRM 機能／トレース機能 [IECUBE]／カバレッジ測定機能 [IECUBE] は排他使用の機能です。

このため、RRM 機能選択時にはトレース機能とカバレッジ機能が、トレース機能選択時には RRM 機能とカバレッジ機能が、カバレッジ機能選択時には RRM 機能とトレース機能が使用できなくなります。

各機能の切り替えは、[オプション]メニュー→[RRM 機能／トレース機能／カバレッジ機能]の選択により行います。

各機能は基本的には排他使用ですが、以下の操作は可能です。

- 各機能を切り替えてもカバレッジ・データはクリアされません。
- カバレッジ機能以外の機能を選択している場合でも、カバレッジ・データのダウンロード／アップロード、およびクリアは可能です。
- カバレッジ機能を選択している場合、All PC 固定のトレースが可能です（条件トレースは不可）。
- RRM 機能を選択している場合、**トレース・ウインドウ**に表示されるトレース・データは、分岐情報と**リアルタイム・モニタ機能 [IECUBE]**のサンプリング範囲内のアクセス情報のみになります（Branch PC 固定のトレース）。

## 5.12 イベント機能

イベントとは、「アドレス 0x1000 番地をフェッチした」、「アドレス 0x2000 番地にデータを書き込んだ」などのデバッグにおけるターゲット・システムの特定の状態を指しています。

ID850QB では、このようなイベントをブレーク、トレース等の各デバッグ機能のアクション・トリガとして利用しています。

この項では次の項目について解説します。

- イベント機能の利用
- イベントの作成
- 各種イベント条件の設定
- 各種イベント条件の有効イベント数
- イベントの管理

### 5.12.1 イベント機能の利用

イベント（イベント条件、およびイベント・リンク条件）は、各デバッグ機能を割り当てることで次表に示す各種イベント条件となります。これにより、各種イベント条件をデバッグ目的に準じて機能させることができます。

表 5 - 16 各種イベント条件

条件名	マーク	内容 →設定ダイアログ
ブレーク・イベント	B	ユーザ・プログラムの実行、またはトレーサの動作を停止させる際の条件（「5.4 ブレーク機能」参照） →ブレーク・ダイアログ
トレース・イベント [IECUBE]	T	ユーザ・プログラムの実行過程をトレース・メモリに保存する際の条件（「5.10 トレース機能 [IECUBE]」参照） →トレース・ダイアログ
タイマ・イベント [IECUBE]	Ti	時間測定の開始タイミングと終了タイミングを指定する際の条件（「5.9 タイマ機能 [IECUBE]」参照） →タイマ・ダイアログ

### 5.12.2 イベントの作成

イベントは、イベントという状態に個別に名前を付け、イベント条件、およびイベント・リンク条件として登録することで先に述べた各種イベント条件のアクション・トリガとして使用可能になります。

#### (1) イベント条件の作成、登録

イベント条件の作成は、通常イベント・ダイアログで行います。

このダイアログ上で、アドレス条件、ステータス条件、データ条件などを設定し、1つのイベント条件として名前を付け登録します。

簡易的な作成、登録方法として、ソース・テキスト・ウィンドウ、および逆アセンブル・ウィンドウ上でブレーク・ポイントを設定することにより生成されたイベント条件の利用も可能です（「5.4.2 ブレーク・ポイントの設定」参照）。

#### (2) イベント・リンク条件の作成、登録

イベント・リンク条件は、イベント条件に順序的な制約をもうけて単一のイベントとする条件で、指定順序どおりにユーザ・プログラムが実行した場合にのみ発生します。

イベント・リンク条件の作成は、イベント・リンク・ダイアログで行います。

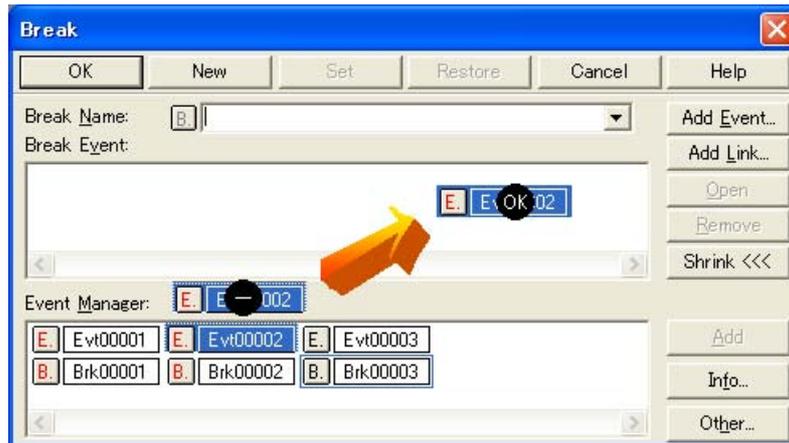
### 5.12.3 各種イベント条件の設定

表5-16 に示した各種イベント条件の作成は、それぞれの設定ダイアログ上において個別に行います。

#### (1) 各種イベント条件の設定方法

各種イベント条件の設定は、各設定ダイアログ上のイベント・マネージャ・エリア（またはイベント・マネージャ）に表示されているイベント条件、またはイベント・リンク条件のイベント・アイコンを選択し、設定したい条件エリアにドラッグ＆ドロップすることにより行います。

図5-22 各種イベント条件の設定方法



ドラッグ中のマウス・カーソルは、設定可能な条件エリア上で "OK" の形状となります。

作成した各種イベント条件は、設定ダイアログ上の <Set> ボタン、または <OK> ボタンをクリックすることにより、イベント・アイコンのマーク部分が赤くなり設定（有効化）されます。設定されることによりはじめて各種イベント条件としてのデバッグ・アクションが発生します。

#### (2) 選択モードでの設定（内容確認後の設定）

設定したい条件エリアにフォーカスを置き、<Add Event...> ボタン、または <Add Link...> ボタンをクリックすることにより、イベント・ダイアログ、イベント・リンク・ダイアログが選択モードでオープンします。ダイアログ上で設定する条件を選択することで該当イベントの詳細条件が表示されるため、内容確認を行ってからの条件設定が可能です。

#### (3) イベント・アイコンのコピー、移動

イベント条件設定エリア内では、ドラッグ＆ドロップによる以下の方法でイベント条件のコピー、移動が可能です。

- マウスだけでドロップした場合には、イベント条件を移動します。
- Ctrl キーを押しながらドロップした場合には、イベント条件をコピーします。

**(4) イベント・マネージャ・エリアでの操作**

設定したい条件エリアにフォーカスを置き、イベント・アイコンを選択した状態で <Add> ボタンをクリックすることでも各種イベント条件の設定が可能です。

**イベント設定内容表示**

イベントを選択し、<Open> ボタンをクリック、あるいはイベントをダブルクリックすることにより、選択したイベントに対応する設定ダイアログがオープンし、イベントの設定内容を表示することができます。

**削除**

イベントを選択し、<Remove/Delete> ボタンをクリック、あるいは Delete キーを押すことにより、イベントの削除が可能です。

**表示モード変更、並べ替え**

<Info...> ボタンをクリックすることにより、イベント・マネージャ・エリアの表示モードの変更や並べ替えが可能です。

**エリア非表示**

<Shrink<<<> ボタンにより、エリアの非表示が可能です。

**5.12.4 各種イベント条件の有効イベント数**

イベント条件、各種イベント条件は、それぞれ最大 256 個の条件を登録することができます。

1つのイベント条件、またはリンク・イベント条件をブレイク、トレース等複数の各種イベント条件として設定可能です。

ただし、同時に設定可能な（有効になる）各種イベント条件の個数には次のように制限があります。

このため、有効数を越えている場合や、使用しているイベント条件、またはイベント・リンク条件が同時に使用できる個数を越えてしまう場合には、一旦、設定した各種イベント条件を無効にした状態で登録する必要があります（「5.12.5 イベントの管理」参照）。

表 5 - 17 各種イベント条件における有効イベント数

接続 IE		イベント		イベント・リンク	ブレイク	トレース	タイマ
		実行	アクセス				
<b>[IECUBE]</b>		10 <sup>*a</sup>	6 <sup>*b</sup>	1	10+6	1	7 <sup>*g</sup>
<b>[MINICUBE]</b>	Nx85ET (RCU0+TEU+TRCU)	10 <sup>*a</sup>	4 <sup>*c</sup>	1 <sup>*d</sup>	10+4	-	-
	Nx85E901 (RCU0), RCU1	2 <sup>*e</sup>		1 <sup>*f</sup>	2 <sup>*e</sup>	-	-
<b>[MINICUBE2]</b>	デバッグ機能あり	2 <sup>*e</sup>		1 <sup>*f</sup>	2 <sup>*e</sup>	-	-
	デバッグ機能なし	-		-	-	-	-

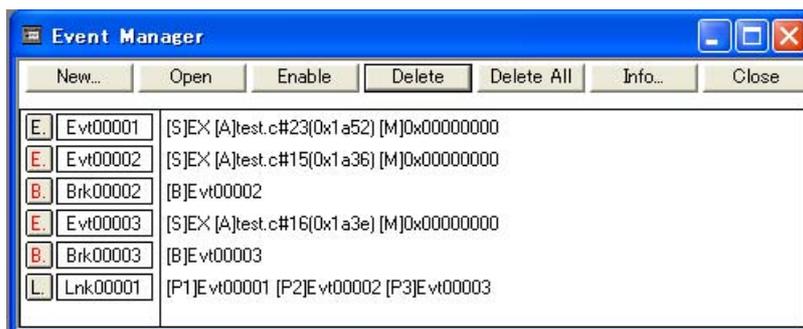
- \*a 実行前 2 個（ブレークにのみ使用可，アドレス範囲指定不可），実行後 8 個（アドレス範囲指定時は範囲指定により 2 個のイベントを使用するため 4 個）
- \*b アドレス範囲指定時は範囲指定により 2 個のイベントを使用するため 3 個。
- \*c アドレス範囲指定時は範囲指定により 2 個のイベントを使用するため 2 個。
- \*d Phase1 ~ Phase4 使用
- \*e アドレス範囲指定は不可
- \*f Phase1 と Phase2 にのみ設定可能
- \*g Start Event, または End Event に範囲アドレスを指定したイベント条件を設定することはできません。範囲アドレスを指定したイベント・リンク条件は設定可能です。

### 5.12.5 イベントの管理

すべてのイベントの管理は [イベント・マネージャ](#)で行います。

イベント・マネージャでは、[各種イベント条件](#)の表示（詳細表示），有効／無効の切り替え，削除等が可能です。

図 5 - 23 イベントの管理（イベント・マネージャ）



**(1) イベント・アイコン**

イベント・アイコンはイベントの種類を示すマークとイベント名とで構成されており、色はイベントの設定状態を示しています。

マーク部分をクリックすることにより、有効/無効の切り替えが可能です。

表5-18 イベント・アイコン

文字色	マーク	意味
赤	E.L.	イベント条件, イベント・リンク条件を使用している各種イベント条件が有効であることを示します。
	B.T.Ti.	各種イベント条件が有効であることを示します。 条件が成立することにより各種イベントが発生します。
黒	E.L.	イベント条件, イベント・リンク条件を使用している各種イベント条件が無効であることを示します。
	B.T.Ti.	各種イベント条件が無効であることを示します。 条件が成立してもイベントは発生しません。
黄	E.L.	イベント条件で指定しているシンボルが, 現在ロードしているプログラムでは認識できないために保留であることを示します。
	B.T.Ti.	各種イベント条件が保留であることを示します。 条件が成立してもイベントは発生しません。

## 5.13 RRM 機能

この項では、RRM 機能（リアルタイム・モニタ機能）に関する次の項目について解説します。

- [リアルタイム・モニタ機能 \[IECUBE\]](#)
- [疑似リアルタイム・モニタ機能 \(Break When Readout\)](#)

**注意** [RRM 機能](#)/[トレース機能 \[IECUBE\]](#)/[カバレッジ測定機能 \[IECUBE\]](#) は排他使用の機能です（「[5.11.4 RRM 機能](#)/[トレース機能](#)/[カバレッジ機能間の排他使用に関して](#)」参照）。

各機能の切り替えは、[オプション]メニューで行います。

### 5.13.1 リアルタイム・モニタ機能 [IECUBE]

リアルタイム・モニタ機能による読み込みが可能な領域は表 5 - 19 のとおりです。

この領域に割り当てられている変数、およびデータなどはウォッチ・ウインドウ、メモリ・ウインドウ上で常にリアルタイムな表示が可能です。

なお、サンプリング間隔は、拡張オプション設定ダイアログの " Redraw Interval " で指定可能です。

表 5 - 19 リアルタイム・モニタ機能サンプリング可能領域

接続 IE	サンプリング範囲
[IECUBE]	RRM 設定ダイアログでの指定領域

図 5 - 24 RRM 設定ダイアログ

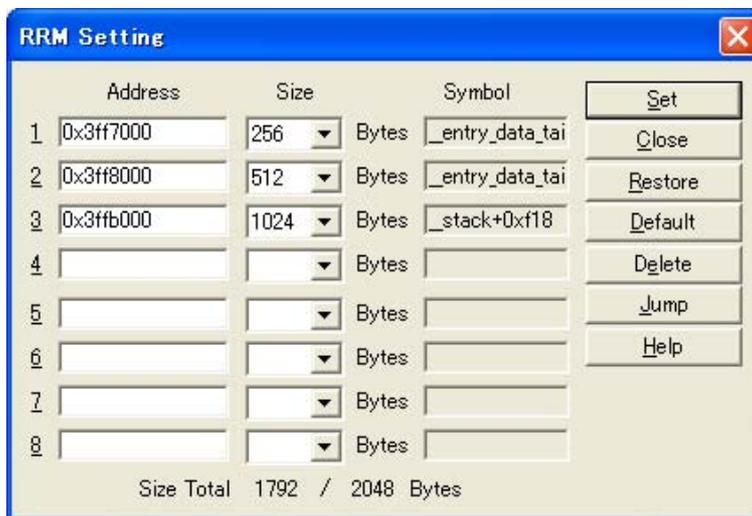


図 5 - 25 リアルタイム・モニタ機能のサンプリング間隔の指定



### 5.13.2 疑似リアルタイム・モニタ機能 (Break When Readout)

リアルタイム・モニタによる読み込みが不可能な領域に関しては、疑似リアルタイム・モニタ機能による読み込みが可能です。

疑似リアルタイム・モニタ機能では、ソフトウェア・エミュレーションによる読み込みを行うため、読み込み時にユーザ・プログラムの実行が一瞬ブレークします。

このため、この範囲に当てられている変数、およびデータなどはウォッチ・ウインドウ、メモリ・ウインドウ上でほぼリアルタイムな表示となります。

疑似リアルタイム・モニタ機能の ON / OFF、およびサンプリング範囲は、拡張オプション設定ダイアログの "Break When Readout" で指定します。

図 5 - 26 疑似リアルタイム・モニタ機能の指定



**注意** 疑似リアルタイム・モニタ機能とソフトウェア・ブレークは、排他使用の機能です。

疑似リアルタイム・モニタ機能を有効にした場合 ("Off" 以外を選択)、ソフトウェア・ブレーク・ポイントは設定できません。また、以前に設定した有効なソフトウェア・ブレーク・ポイントはすべて無効になります。

## 5.14 DMM 機能

DMM (Dynamic Memory Modification) 機能とは、ユーザ・プログラム実行中に、リアルタイムにメモリ (RAM) の内容を書き換える機能です。

DMM を行うには、メモリ・ウィンドウ、レジスタ・ウィンドウ、IOR ウィンドウ上の <DMM...> ボタンのクリックによりオープンする DMM ダイアログで、アドレスとデータを指定します。

**注意 1** この機能はソフトウェア・エミュレーションによって実現されているため、書き換えに際し一時的にユーザ・プログラムの実行が停止されます (疑似 DMM 機能)。

**注意 2** DMM 機能とソフトウェア・ブレークは、排他使用の機能です。

有効なソフトウェア・ブレーク・ポイントが設定されている場合、ユーザ・プログラム実行中の書き込み (DMM) はできません (エラーになります)。

図 5 - 27 メモリ内容の書き換え (DMM ダイアログ)



## 5.15 ロード／セーブ機能

ID850QB では、以下に示す各情報をファイルとして保存、およびロードが可能です。

これにより各情報の保存時の復元が可能です。

- デバッグ環境（プロジェクト・ファイル）
- ウィンドウの表示情報（表示ファイル）
- ウィンドウの設定情報（設定ファイル）

**参考** [ウィンドウ]メニュー→[スタティック]を選択することで、ウィンドウの状態保持を簡易的に行うことができます（「5.16.1 アクティブ状態とスタティック状態」参照）。

### 5.15.1 デバッグ環境（プロジェクト・ファイル）

プロジェクト・ファイル (\*.prj) は、デバッグ環境を記録するファイルです。

ある時点のデバッグ環境を保存することでプロジェクト・ファイルを作成し、このファイルを次回ロードすることにより、デバッグ環境の復元が可能です。

プロジェクト・ファイルのロード／セーブは、[プロジェクト・ファイル・セーブ・ダイアログ](#)、[プロジェクト・ファイル・ロード・ダイアログ](#)で行います。

起動時にプロジェクト・ファイルをロードする場合には、[コンフィギュレーション・ダイアログ](#)上の <Project...> ボタンから行います。

プロジェクト・ファイルへは、次の内容が保存されます。

表 5 - 20 プロジェクト・ファイルの保存内容

ウインドウ	保存内容
<a href="#">コンフィギュレーション・ダイアログ</a>	すべての項目（ターゲット・デバイス、クロック設定、端子マスク設定、マッピング情報）
メイン・ウインドウ	表示位置、ツールバー／ステータスバー／ボタン表示情報、実行モード情報
<a href="#">ダウンロード・ダイアログ</a>	ダウンロードするファイル情報
<a href="#">拡張オプション設定ダイアログ</a> <a href="#">デバッグ・オプション設定ダイアログ</a> <a href="#">フェイルセーフ・ブレーク設定ダイアログ</a> <a href="#">フラッシュ・オプション設定ダイアログ</a> <a href="#">データ・フラッシュ・オプション設定ダイアログ</a> <a href="#">RRM 設定ダイアログ</a>	設定情報
<a href="#">逆アセンブル・ウインドウ</a> <a href="#">メモリ・ウインドウ</a>	ウインドウの表示情報、表示開始アドレス情報
<a href="#">ソース・テキスト・ウインドウ</a> <a href="#">スタック・トレース・ウインドウ</a> <a href="#">IOR ウインドウ</a> <a href="#">ローカル変数ウインドウ</a> <a href="#">トレース・ウインドウ</a> <a href="#">コード・カバレッジ・ウインドウ</a> <a href="#">イベント・マネージャ</a> <a href="#">コンソール・ウインドウ</a> <a href="#">拡張ウインドウ</a> <a href="#">レジスタ・ウインドウ</a>	ウインドウの表示情報
<a href="#">イベント・ダイアログ</a> <a href="#">イベント・リンク・ダイアログ</a> <a href="#">ブレーク・ダイアログ</a> <a href="#">トレース・ダイアログ</a> <a href="#">タイマ・ダイアログ</a>	ウインドウの表示情報、各種イベント情報
<a href="#">リスト・ウインドウ</a>	ウインドウの表示位置
<a href="#">ウォッチ・ウインドウ</a>	ウインドウの表示情報 <sup>注</sup> 、ウォッチ登録情報
<a href="#">I/O ポート追加ダイアログ</a>	追加 I/O ポート情報
<a href="#">DMM ダイアログ</a>	DMM 情報
<a href="#">ディレイ・カウント設定ダイアログ</a>	ディレイ・カウント値

ウインドウ	保存内容
ソフトウェア・ブレイク・マネージャ	ウインドウの表示情報、ブレイク情報

注 構造体、配列等のポインタのメンバ表示の状態、およびメンバ個別の表示進数は保存されません。

## 5.15.2 ウインドウの表示情報（表示ファイル）

表示ファイルは、ウインドウの表示情報を記録しているファイルです。

表示ファイルは各ウインドウごとにロード／セーブ可能です。

表示ファイルをロードすることにより、参照用ウインドウ（スタティック状態のソース・テキスト・ウインドウ）がオープンし、保存時の表示情報が表示されます。

表示ファイルのロード／セーブは、表示ファイル・ロード・ダイアログ、表示ファイル・セーブ・ダイアログで行います。

表 5 - 21 表示ファイルの種類

ファイルの種類	カレント・ウインドウ名、ファイル名
Source Text (*.sww)	ソース・テキスト・ウインドウ <sup>注1</sup>
Assemble (*.dis)	逆アセンブル・ウインドウ <sup>注1</sup>
Memory (*.mem)	メモリ・ウインドウ
Watch (*.wch)	ウォッチ・ウインドウ
Register (*.rgw)	レジスタ・ウインドウ
I/O Register (*.ior)	IOR ウインドウ
Local Variable (*.loc)	ローカル変数ウインドウ
Stack Trace (*.stk)	スタック・トレース・ウインドウ
Trace (*.tvw)	トレース・ウインドウ
Code Coverage (*.csv)	コード・カバレッジ・ウインドウ (選択タブごとに保存)
List (*.csv)	リスト・ウインドウ (選択タブごとに保存)
Console (*.log)	コンソール・ウインドウ
All (*.*)	すべてのファイル
Source (*.c, *.s)	ソース・ファイル <sup>注2</sup>
Text (*.txt)	テキスト・ファイル

注1 表示ファイルの内容に、コード・カバレッジ測定結果（実行／未実行）のマークが付加されます（「表 5 - 15 カバレッジ実行／未実行箇所の表示形式」参照）。[IECUBE]

注2 ソース・ファイルの拡張子は拡張オプション設定ダイアログで変更可能です。

### 5.15.3 ウィンドウの設定情報（設定ファイル）

設定ファイルは、ウィンドウの設定情報（ウォッチ・データ設定、周辺 I/O レジスタ設定、およびイベント設定）を記録しているファイルです。

設定ファイルは、各ウィンドウごとにロード／セーブ可能です。

設定ファイルをロードすることにより、対象ウィンドウがオープンし保存時の設定情報が復元されます。

設定ファイルのロード／セーブは、[環境設定ファイル・ロード・ダイアログ](#)、[環境設定ファイル・セーブ・ダイアログ](#)で行います。

表 5 - 22 設定ファイルの種類

ファイルの種類	カレント・ウィンドウ名
Watch (*.wch) <sup>注</sup>	<a href="#">ウォッチ・ウィンドウ</a>
I/O Register (*.ior) <sup>注</sup>	<a href="#">IOR ウィンドウ</a>
Event (*.evn)	<a href="#">イベント・マネージャ</a>

**注** 変数値はロードされません。

## 5.16 ウィンドウ共通機能

対象ウィンドウは、次の共通機能を持ちます。

- [アクティブ状態とスタティック状態](#)
- [ジャンプ機能](#)
- [トレース結果とウィンドウの連結機能 \[IECUBE\]](#)
- [ドラッグ & ドロップ機能](#)
- [注意事項](#)

### 5.16.1 アクティブ状態とスタティック状態

下記のウィンドウは、[アクティブ状態](#)と[スタティック状態](#)という2つの状態を持っています。

- [ソース・テキスト・ウィンドウ](#)（シンボル情報を読み込んだソース・ファイル表示時）
- [逆アセンブル・ウィンドウ](#)
- [メモリ・ウィンドウ](#)

アクティブ状態のウィンドウは、1つしかオープンできないのに対し、スタティック状態のウィンドウは、複数オープンしておくことができるため、簡易的に、その時点でのウィンドウの状態を保持しておくことが可能です。状態の切り替えは、[ウィンドウ]メニュー→[アクティブ/スタティック]の選択により行います。

#### (1) アクティブ状態

アクティブ状態のウィンドウは、表示位置、内容ともに、常にカレント PC 値に連動して自動的に更新します。

また、[ジャンプ機能](#)のジャンプ先となるほか、[トレース・ウィンドウ](#)と連結させた場合、[トレース・ウィンドウ](#)と連動して表示内容を更新します。

アクティブ状態のウィンドウは1つのみ、オープンすることができます。

#### (2) スタティック状態

スタティック状態のウィンドウは、カレント PC 値に連動して、表示位置は移動しませんが、内容の更新は行いません。

なお、[ジャンプ機能](#)のジャンプ先とならないほか、[トレース・ウィンドウ](#)とも連結はしません。

すでに、アクティブ状態のウィンドウを表示している場合、次のウィンドウはスタティック状態としてオープンします。

スタティック状態のウィンドウは複数オープンすることができます。

## 5.16.2 ジャンプ機能

ジャンプ機能とは、カーソルの置かれた行／アドレスをジャンプ・ポインタとして、下記ウインドウのいずれかへジャンプし、ジャンプ先のウインドウを、ジャンプ・ポインタから表示する機能です。

- ソース・テキスト・ウインドウ
- 逆アセンブル・ウインドウ
- メモリ・ウインドウ

上記ウインドウ同士、またはトレース・ウインドウ、スタック・トレース・ウインドウ、イベント・マネージャ、レジスタ・ウインドウから上記ウインドウへのジャンプが可能です。

### (1) ジャンプ方法

次の方法でジャンプをします。

- 1) ジャンプ可能なウインドウで、ジャンプ・ポインタとしたい行／アドレスにカーソルを移動（イベント・マネージャでは、イベント・アイコンを選択）します。
- 2) [ジャンプ]メニューより、ジャンプしたいウインドウを選択します。

**注意** 対象ソース行にプログラム・コードが存在しない場合にはプログラム・コードの存在する上下どちらかの行の先頭アドレスが対象となります。

### (2) ジャンプ元アドレスの詳細

次に挙げるウインドウからジャンプする場合のジャンプ元アドレスの詳細を示します。

表 5 - 23 ジャンプ元アドレスの詳細

対象ウインドウ	ジャンプ・ポインタ詳細	
レジスタ・ウインドウからジャンプ	選択したレジスタ値	
メモリ・ウインドウからジャンプ	カーソル位置のアドレス値	
イベント・マネージャからジャンプ	選択したイベント・アイコンがイベント条件の場合、アドレス条件をジャンプ・ポインタとします。	
	アドレス条件がポイント設定の場合	指定アドレスにジャンプ
	アドレス条件が範囲設定の場合	下位アドレス（マスク指定がされている場合、マスク前のポイント・アドレス）にジャンプ
	アドレス条件がビット設定の場合	ビット位置のアドレスにジャンプ

対象ウインドウ	ジャンプ・ポインタ詳細	
スタック・トレース・ウインドウからジャンプ	カーソル位置のスタック・フレーム番号で示される関数をジャンプ・ポインタとします。	
	<b>カレント関数の場合</b>	
	ジャンプ先がソース・テキスト・ウインドウ	カレント PC 行にジャンプ
	上記以外	カレント PC アドレスにジャンプ
	<b>カレント関数以外の場合</b>	
	ジャンプ先がソース・テキスト・ウインドウ	ネストしている関数を呼び出す行にジャンプ
	上記以外	ネストしている関数を呼び出す命令の次のアドレスにジャンプ
トレース・ウインドウからジャンプ	<b>メモリ・ウインドウへのジャンプ</b>	
	カーソル位置がアクセス・アドレス、アクセス・データ、アクセス・ステータスの場合	アクセス・アドレス
	上記以外	フェッチ・アドレス
ソース・テキスト・ウインドウ、または逆アセンブル・ウインドウへのジャンプ	フェッチ・アドレス	

### 5.16.3 トレース結果とウインドウの連結機能 [IECUBE]

トレース・ウインドウと各ウインドウ（ソース・テキスト・ウインドウ、逆アセンブル・ウインドウ、メモリ・ウインドウ）を連結することにより、トレース・ウインドウ上のカーソル位置のアドレスをポインタとして、連結したウインドウ上で対応箇所を表示させることができます。

トレース・ウインドウ上でカーソル位置を移動すると、連結したウインドウ上の対応箇所が反転表示、またはカーソル位置で表示されます。

#### (1) 連結方法

次の方法で連結します。

- 1) **トレース・ウインドウ**をカレント・ウインドウにする。
- 2) [表示]メニュー → [ウインドウ連結]で連結先のウインドウを選択する。
- 3) **トレース・ウインドウ**のトレース結果表示エリアで連結元の行にカーソルを位置付ける。
- 4) 3) で選択した行のアドレスをポインタとして、2) で選択したウインドウの表示エリアで対応する部分が反転表示（、またはカーソル位置で表示）される。

**注意** **メモリ・ウインドウ**との連結では、トレース結果表示エリアでのカーソルの位置によって、連結元アドレスが次のように変わります。

- アクセス・アドレス, アクセス・データ, アクセス・ステータス→アクセス・アドレス
- 上記以外→フェッチ・アドレス

なお、**ソース・テキスト・ウインドウ**、または**逆アセンブル・ウインドウ**との連結では、常にフェッチ・アドレスをポインタとします。

## 5.16.4 ドラッグ & ドロップ機能

選択により反転表示している行番号、アドレス、およびテキストは、次の方法によって、ほかのウィンドウやエリア上へドラッグ & ドロップすることができます。

- 1) 選択されている行番号、アドレス、およびテキストをドラッグします。  
→マウス・カーソルの形状が矢印カーソルから '-' へ変化します。
- 2) ドロップ可能なウィンドウ、エリア上にドロップします。  
→ドロップ可能なウィンドウ、エリア上では、カーソルの形状が '-' から 'OK' へ変化します。

ドロップされたウィンドウでは、ドロップされたアドレス、行番号から求められるアドレスに対して動作を行います。例えばソース・テキスト・ウィンドウ上の変数をウォッチ・ウィンドウ上にドラッグ & ドロップすることにより、簡単に変数の登録が行えます。

### (1) ドラッグ & ドロップの詳細

ドロップ後の動作は、ドロップされたそれぞれのウィンドウやエリアで次のように異なります。

表 5 - 24 ドラッグ & ドロップ機能の詳細 (行/アドレスの場合)

ドロップ対象のウィンドウ/エリア	ドロップ後の動作
イベント・マネージャ、および各種イベント設定ダイアログのイベント・マネージャ・エリア	ドロップされた行番号やアドレスをアドレス条件として、実行イベント条件を自動生成します。 イベント条件名は、Evt00001, Evt00002... のように自動生成します。 パス・カウントは未指定になります。 アドレス条件は、最も近くにあるシンボルに対して、シンボル名 + オフセット値の形式で設定されます。
各種イベント設定ダイアログ中の条件設定エリア (アドレス、データ設定エリア以外)	ドロップされた行番号やアドレスをアドレス条件として、実行イベント条件を自動生成します。 更に、自動生成されたイベント条件をドロップされた各条件設定エリアに設定します。 イベント条件名は、Evt00001, Evt00002... のように自動生成します。 パス・カウントは未指定になります。 アドレス条件は、最も近くにあるシンボルに対して、シンボル名 + オフセット値の形式で設定されます。
各種イベント設定ダイアログ中の条件設定エリア (アドレス、データ設定エリア)	ドロップされたエリアへ行番号やアドレスのテキストが設定されます。 アドレス条件は、最も近くにあるシンボルに対して、シンボル名 + オフセット値の形式で設定されます。

表 5 - 25 ドラッグ &amp; ドロップ機能の詳細 (文字列の場合)

ドロップ対象のウインドウ/エリア	ドロップ後の動作	
イベント・マネージャ, および各種イベント設定ダイアログのイベント・マネージャ・エリア	ドロップされたテキストがシンボルとしてアドレス値へ変換可能である場合, 変換されたアドレス値をアドレス条件として, Access (R/W) ステータス (すべてのアクセス・ステータス), または Execute ステータスのイベント条件を自動生成します。 イベント条件名は, Evt00001, Evt00002... のように自動生成します。 データ条件とパス・カウントは未指定になります。 アドレス条件にはドロップされたテキストで設定されます。 生成されるイベント条件とシンボルの関係は, 次の通りです。	
	シンボル	ステータス
	変数	Access(R/W)
	関数	Execute
	データ・セクション内のシンボル	Access(R/W)
	コード・セクション内のシンボル	Execute
	その他	Access(R/W)
各種イベント設定ダイアログ中の条件設定エリア (アドレス, データ設定エリア以外)	ドロップされたテキストがシンボルとしてアドレス値へ変換可能である場合, 変換されたアドレス値をアドレス条件として, Access (R/W) ステータス (すべてのアクセス・ステータス), または Execute ステータスのイベント条件を自動生成します。 更に, 自動生成されたイベント条件を, ドロップされた各条件設定エリアに設定します。 イベント条件名は, Evt00001, Evt00002... のように自動生成します。 データ条件とパス・カウントは未指定になります。 アドレス条件にはドロップされたテキストで設定されます。 生成されるイベント条件とシンボルの関係は, 次の通りです。	
	シンボル	ステータス
	変数	Access(R/W)
	関数	Execute
	データ・セクション内のシンボル	Access(R/W)
	コード・セクション内のシンボル	Execute
	その他	Access(R/W)
各種イベント設定ダイアログ中の条件設定エリア (アドレス, データ設定エリア)	ドロップされたエリアへテキストが設定されます。	
ウォッチ・ウインドウ	ドロップされたテキストが, シンボルとして認識可能である場合, シンボルの内容を表示します。	

**参考** 各種イベント設定ダイアログは, 以下を指します。

- イベント・ダイアログ
- イベント・リンク・ダイアログ
- ブレーク・ダイアログ
- トレース・ダイアログ
- タイマ・ダイアログ

### 5.16.5 注意事項

- (1) ウィンドウ上の各エリアにおいて1行に表示できる文字数は319文字です。
- (2) 表示エリアの横幅が狭い場合、表示が乱れることがあります。ウィンドウの横幅を広げて調整してください。
- (3) <Refresh> ボタンのあるウィンドウでは、アクティブ時にカーソル位置を移動させると、再描画が正しく行われない場合があります。<Refresh> ボタンを押して再描画を行ってください。
- (4) F1 キーによりヘルプを表示させる場合、カーソルのあるウィンドウが対象になります。  
このため、起動直後などでトレース結果が全く表示されていない**トレース・ウィンドウ**では、ウィンドウ上にカーソルを置くことができないため、F1 キーを押してもヘルプが表示されない場合があります。この場合には、[ヘルプ]メニュー→[カレントウィンドウ]を選択し、ヘルプの表示を行ってください。
- (5) 自動継続実行中は、[実行]メニュー→[スローモーション]を選択しないでください。  
自動継続実行中、通常は、[実行]メニュー→[スローモーション]は淡色表示になりますが、選択できる瞬間があり、この間に[スローモーション]を選択すると、[実行]メニュー→[ストップ]を選択しても(STOP ボタンを押しても)プログラムが停止しなくなります。
- (6) イベント・アイコンなどをドラッグ中に、何らかの要因でアプリケーションの切り替えが発生した場合、ドラッグ状態のままドロップできなくなります。ESC キーでドラッグ状態を解除してからやり直してください。

## 第 6 章 ウィンドウ・リファレンス

この章では、ID850QB が持つウィンドウ／ダイアログの機能詳細について解説します。

- [ウィンドウ一覧](#)
- [各ウィンドウの説明](#)

## 6.1 ウィンドウ一覧

ID850QB のウィンドウ／ダイアログ一覧を示します。

表 6-1 ウィンドウ／ダイアログ一覧

ウィンドウ名	内容
メイン・ウィンドウ	ユーザ・プログラム実行制御, および各種ウィンドウのオープン
コンフィギュレーション・ダイアログ	動作環境の設定
拡張オプション設定ダイアログ	各種拡張オプションの設定
フェイルセーフ・ブレーク設定ダイアログ	フェイルセーフ・ブレークの設定
RRM 設定ダイアログ	RRM サンプリング範囲の設定
フラッシュ・オプション設定ダイアログ	フラッシュ・セルフ・プログラミング・エミュレーションの設定 [IECUBE]
データ・フラッシュ・オプション設定ダイアログ	データ・フラッシュ・エラー・エミュレーションの設定 [IECUBE]
デバッグ・オプション設定ダイアログ	その他各種オプションの設定
プロジェクト・ファイル・セーブ・ダイアログ	現在のデバッグ環境をプロジェクト・ファイルに保存
プロジェクト・ファイル・ロード・ダイアログ	デバッグ環境の復元
ダウンロード・ダイアログ	ダウンロード
アップロード・ダイアログ	アップロード
ロード・モジュール一覧ダイアログ	ダウンロード済みファイル名の表示
ソース・テキスト・ウィンドウ	ソース・ファイル, またはテキスト・ファイルの表示
ソース・サーチ・ダイアログ	ソース・テキスト・ウィンドウ内の検索
ソース指定ダイアログ	ソース・テキスト・ウィンドウに表示するファイルの指定, および表示開始位置の指定
逆アセンブル・ウィンドウ	ユーザ・プログラムの逆アセンブル表示
逆アセンブル・サーチ・ダイアログ	逆アセンブル・ウィンドウ内の検索
アドレス指定ダイアログ	メモリ・ウィンドウ, 逆アセンブル・ウィンドウ, IOR ウィンドウの表示開始アドレスの指定
シンボル変換ダイアログ	変数, 関数のアドレス, シンボルの値の表示
リスト・ウィンドウ	関数, 変数, シンボル, セクション, 割り込み要求名の一覧表示
ウォッチ・ウィンドウ	変数の表示
クイック・ウォッチ・ダイアログ	変数の値の一時的な表示
ウォッチ登録ダイアログ	ウォッチ・ウィンドウに表示する変数の登録
ウォッチ変更ダイアログ	ウォッチ・ウィンドウに表示する変数の変更
ローカル変数ウィンドウ	カレント関数内のローカル変数の表示
スタック・トレース・ウィンドウ	スタック内容の表示

ウィンドウ名	内容
メモリ・ウィンドウ	メモリ内容の表示
メモリ・サーチ・ダイアログ	メモリ・ウィンドウ内の検索
メモリ・フィル・ダイアログ	メモリ内容を指定データで埋める（初期化）
メモリ・コピー・ダイアログ	メモリのコピー
メモリ比較ダイアログ	メモリの比較
メモリ比較結果ダイアログ	メモリの比較結果表示
DMM ダイアログ	DMM を行うアドレスとデータの設定
レジスタ・ウィンドウ	レジスタ内容の表示
レジスタ選択ダイアログ	レジスタ・ウィンドウに表示するレジスタの選択
IOR ウィンドウ	IOR の内容表示
IOR 選択ダイアログ	IOR ウィンドウに表示する IOR や I/O ポートの選択
I/O ポート追加ダイアログ	IOR ウィンドウに表示する I/O ポートの登録
タイマ・ダイアログ	タイマ・イベント条件の登録, 設定 [IECUBE]
タイマ測定結果ダイアログ	実行時間測定結果表示 [IECUBE]
トレース・ウィンドウ	トレース結果表示 [IECUBE]
トレース・サーチ・ダイアログ	トレース・データの検索 [IECUBE]
トレース表示選択ダイアログ	トレース・ウィンドウ表示項目の選択 [IECUBE]
フレーム指定ダイアログ	トレース・ウィンドウの表示開始位置の指定 [IECUBE]
トレース・ダイアログ	トレース・イベント条件の登録, 設定 [IECUBE]
ディレイ・カウント設定ダイアログ	ディレイ・トリガ・トレースのディレイ・カウント設定 [IECUBE]
コード・カバレッジ・ウィンドウ	カバレッジ結果表示 [IECUBE]
カバレッジ範囲選択ダイアログ	カバレッジ測定範囲の選択 [IECUBE]
ソフトウェア・ブレーク・マネージャ	ソフトウェア・ブレークの管理
イベント・マネージャ	各種イベントの管理
イベント・ダイアログ	イベント条件の登録
イベント・リンク・ダイアログ	イベント・リンク条件の登録
ブレーク・ダイアログ	ブレーク・イベント条件の登録, 設定
表示ファイル・セーブ・ダイアログ	カレント・ウィンドウの表示情報を表示ファイルに保存
表示ファイル・ロード・ダイアログ	表示ファイルのロード
環境設定ファイル・セーブ・ダイアログ	カレント・ウィンドウの設定内容を設定ファイルに保存
環境設定ファイル・ロード・ダイアログ	設定ファイルのロード
リセット確認ダイアログ	ID850QB と CPU とシンボル情報の初期化
終了確認ダイアログ	ID850QB の終了
バージョン表示ダイアログ	バージョン表示

ウィンドウ名	内容
コンソール・ウィンドウ	コマンド入力ウィンドウ
ソース・ファイル選択ダイアログ	ファイル選択

## 6.2 各ウィンドウの説明

各ウィンドウ／ダイアログについて次のような形式で解説しています。

### ウィンドウ／ダイアログ名

ここでは、ウィンドウ／ダイアログの簡単な機能説明と注意事項を示します。

接続 IE により、無効になるウィンドウ／ダイアログがある場合には、ウィンドウ／ダイアログ名の右下に有効な接続 IE 名を明記します。

また、ウィンドウ／ダイアログの表示イメージを図として示します。

なお、関連項目を参照先として示しています。

### オープン方法

ここでは、ウィンドウ／ダイアログの主なオープン方法を示します。

### 各エリア説明

ここでは、ウィンドウ／ダイアログ内の各エリアの設定／表示項目の説明をします。

### コンテキスト・メニュー

ここでは、ウィンドウ上でのマウスの右クリックにより表示される、コンテキスト・メニューの項目について説明しています。コンテキスト・メニューからは、このウィンドウ上でよく使用する便利な機能が、ワン・アクションで行えます（ウィンドウのみ）。

### 関連操作

このウィンドウ／ダイアログでの操作方法を示します。

## メイン・ウィンドウ

ID850QB を起動して初期設定終了後、最初に自動的にオープンするウィンドウです。

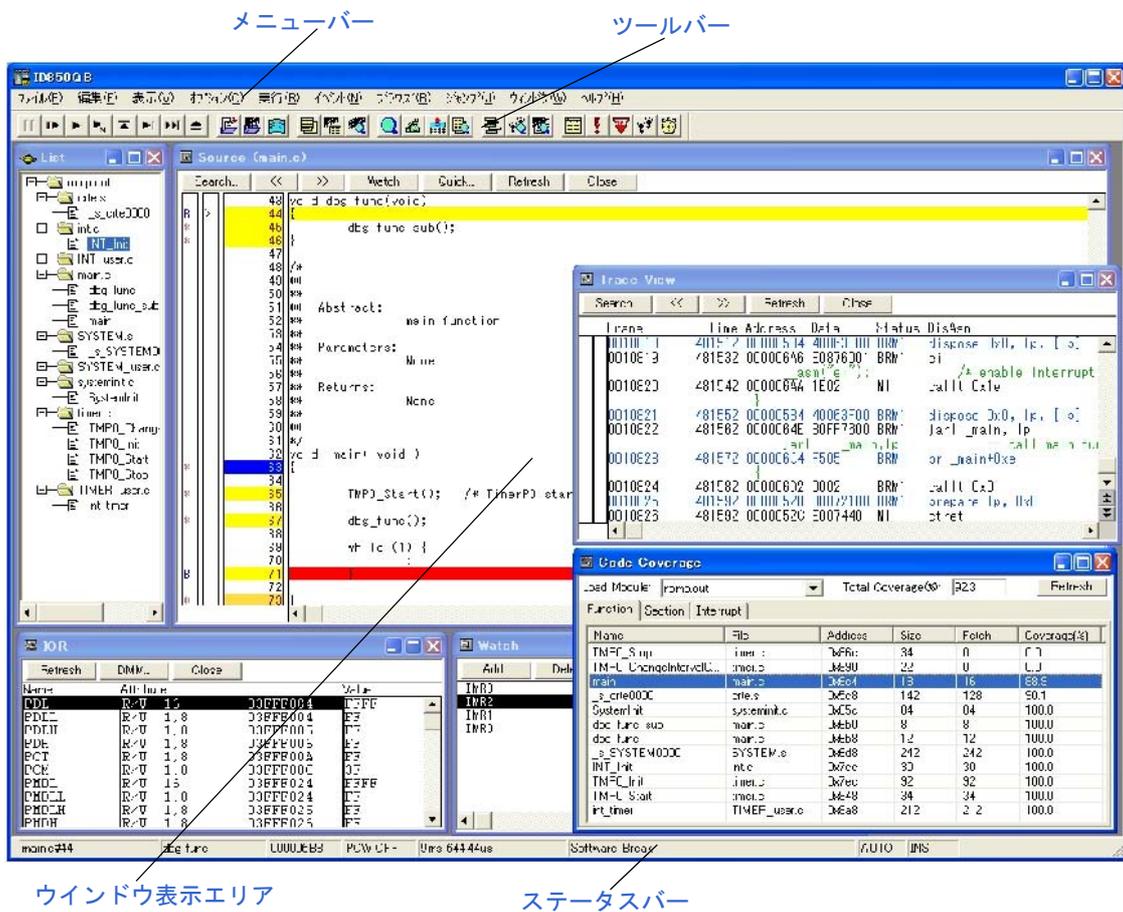
ID850QB では、このウィンドウを中心に各種ウィンドウを操作します（「6.1 ウィンドウ一覧」参照）。

ユーザ・プログラムの実行制御は、このウィンドウ上で行います。

ユーザ・プログラムの実行制御には、次の3つのモードがあります。

- ソース・モード（ソース・レベル・デバッグ）
- 命令モード（命令レベル・デバッグ）
- 自動モード（ソース・レベル/命令レベル自動切り替え）（デフォルト）

図 6-1 メイン・ウィンドウ



## メニューバー

- (1) [ファイル]メニュー
- (2) [編集]メニュー
- (3) [表示]メニュー
- (4) [オプション]メニュー
- (5) [実行]メニュー
- (6) [イベント]メニュー
- (7) [ブラウザ]メニュー
- (8) [ジャンプ]メニュー
- (9) [ウィンドウ]メニュー
- (10) [ヘルプ]メニュー

### (1) [ファイル]メニュー

開く ...	表示ファイル、またはソース・ファイルやテキスト・ファイルをロードします。 表示ファイル・ロード・ダイアログをオープンします。 ダイアログで選択したファイルの拡張子によって動作が異なります。
名前を付けて保存 ...	カレント・ウィンドウの表示内容をファイルに別名でセーブします。 表示ファイル・セーブ・ダイアログをオープンします。
閉じる	カレント・ウィンドウをクローズします。
ダウンロード ...	ファイルをダウンロードします。 ダウンロード・ダイアログをオープンします。
ロードモジュール ...	ダウンロード済みのファイル名を一覧表示します。 ロード・モジュール一覧ダイアログをオープンします。
アップロード ...	プログラムをアップロードします。 アップロード・ダイアログをオープンします。
プロジェクト	プロジェクト・ファイルを操作します。
開く ...	プロジェクト・ファイルをオープンします。 プロジェクト・ファイル・ロード・ダイアログをオープンします。
上書き保存	現在の状態を現在 ID850QB に読み込まれているプロジェクト・ファイルに上書きします。
名前を付けて保存 ...	現在の状態を指定するプロジェクト・ファイルにセーブします。 プロジェクト・ファイル・セーブ・ダイアログをオープンします。
環境	設定ファイルを操作します。
開く ...	設定ファイルをオープンします。 環境設定ファイル・ロード・ダイアログをオープンします。
名前を付けて保存 ...	現在のウィンドウの設定を設定ファイルにセーブします。 環境設定ファイル・セーブ・ダイアログをオープンします。
デバッガ・リセット ...	CPU、シンボル、および ID850QB の初期化を行います。 リセット確認ダイアログをオープンします。
終了	ID850QB を終了します（「3.4 終了方法」参照）。 終了確認ダイアログをオープンします。
(オープン・ファイル)	オープンしたファイル名の一覧を表示します。

## (2) [編集] メニュー

切り取り	選択した文字列を切り取ってクリップ・ボード・バッファに入れます。
コピー	選択した文字列をコピーしてクリップ・ボード・バッファに入れます。
貼り付け	クリップ・ボード・バッファの内容をテキスト・カーソル位置に貼り付けます。
書き込み	修正した内容をターゲットに書き込みます。
復元	修正した内容を取り消します。
メモリ	メモリ内容を操作します。
初期化 ...	メモリの初期化を行います。 メモリ・フィル・ダイアログをオープンします。
複写 ...	メモリのコピーを行います。 メモリ・コピー・ダイアログをオープンします。
比較 ...	メモリの比較を行います。 メモリ比較ダイアログをオープンします。
DMM...	ユーザ・プログラム実行中にメモリの内容をリアルタイムに書き換えます。 DMM ダイアログをオープンします。
ソースの修正	PM+ が動作している場合、アクティブなソース・テキスト・ウィンドウで表示しているソース・ファイルを、PM+ で指定されたエディタでオープンします。

## (3) [表示] メニュー

[表示] メニューには、共通部とアクティブなウィンドウに応じて追加される専用部があります。専用部に関しては、各ウィンドウを参照してください。

## (a) 共通部

検索 ...	検索を行います。 カレント・ウィンドウに応じて、各サーチ・ダイアログをオープンします。 <Search...> ボタンと同じ動作です。
移動 ...	表示位置を移動します。 カレント・ウィンドウに応じて、各指定ダイアログをオープンします。
クイック・ウォッチ ...	指定したデータの内容を一時的に表示します。 クイック・ウォッチ・ダイアログをオープンします。
ウォッチ登録 ...	指定したデータをウォッチ・ウィンドウに登録します。 ウォッチ登録ダイアログをオープンします。
ウォッチ追加	選択したデータをウォッチ・ウィンドウに追加します。 シンボルの場合は、デバッガ・オプション設定ダイアログの設定にしたがって追加します。
ウォッチ変更 ...	選択したウォッチ・ポイントの変数名文字列を変更します。 ウォッチ変更ダイアログをオープンします。 この項目は、ウォッチ・ウィンドウで変数を選択しているときのみ有効です。
ウォッチ削除	選択したウォッチ・ポイントをウォッチ・ウィンドウから削除します。 この項目は、ウォッチ・ウィンドウで変数を選択しているときのみ有効です。
シンボル変換 ...	指定した変数や関数のアドレス、指定したシンボルの値を表示します。 シンボル変換ダイアログをオープンします。

## (4) [オプション]メニュー

ツールバー	ツールバーの表示（デフォルト）／非表示を切り替えます。
ステータスバー	ステータスバーの表示（デフォルト）／非表示を切り替えます。
ボタン	各ウィンドウのボタンの表示（デフォルト）／非表示を切り替えます。
ソース・モード	ソース・レベル（行単位）でステップ実行します。
命令モード	命令レベル（命令単位）でステップ実行します。
自動モード	ソース・レベルのステップ実行と命令レベルのステップ実行を自動的に切り替えて、ステップ実行します（デフォルト）。 ソース・テキスト・ウィンドウがアクティブのときは、ソース・レベル（混合表示モード以外）、逆アセンブル・ウィンドウがアクティブのときは、命令レベルでステップ実行します。どちらのウィンドウもアクティブではないときは、ソース・レベルでステップ実行します。
コンフィギュレーション ...	環境設定を行います。 コンフィギュレーション・ダイアログをオープンします。
拡張オプション ...	拡張機能の設定を行います。 拡張オプション設定ダイアログをオープンします。
RRM 設定 ... [IECUBE]	RRM 機能のサンプリング範囲を設定します。 RRM 設定ダイアログをオープンします。
フラッシュ・オプション ... [IECUBE]	フラッシュ・セルフ・プログラミング・エミュレーションの設定を行います。 フラッシュ・オプション設定ダイアログをオープンします。
データ・フラッシュ・オプション ... [IECUBE]	データ・フラッシュ・エラー・エミュレーションの設定を行います。 データ・フラッシュ・オプション設定ダイアログをオープンします。
デバッガ・オプション ...	ID850QB のオプション設定を行います。 デバッガ・オプション設定ダイアログをオープンします。
I/O ポート追加 ...	ユーザ定義の I/O ポートを追加します。 I/O ポート追加ダイアログをオープンします。
トレース・クリア [IECUBE]	トレース・データをクリアします。 トレース・ウィンドウがアクティブな時のみ表示されます。
カバレッジ [IECUBE]	次のカバレッジ測定関連のダイアログをオープンします。
クリア [IECUBE]	カバレッジ測定結果をクリアします。
選択 ... [IECUBE]	1M バイト以上の空間のカバレッジ測定範囲の選択をします。 カバレッジ範囲選択ダイアログをオープンします。
RRM 機能 [IECUBE]	RRM 機能を選択します（「5.13 RRM 機能」参照）（デフォルト）。 この機能は、トレース機能、カバレッジ機能との排他使用です（「5.11.4 RRM 機能／トレース機能／カバレッジ機能間の排他使用に関して」参照）。 選択時、以下のメニューは選択不可となります。 [ トレーサ停止／開始 ], [ 無条件トレース／条件トレース ], [ トレーサ制御モード ]
トレース機能 [IECUBE]	トレース機能を選択します（「5.10 トレース機能 [IECUBE]」参照）。 この機能は、RRM 機能、カバレッジ機能との排他使用です（「5.11.4 RRM 機能／トレース機能／カバレッジ機能間の排他使用に関して」参照）。 選択時、以下のメニューは選択不可となります。 [ RRM 設定 ... ]

カバレッジ機能 [IECUBE]	カバレッジ機能を選択します（「5.11 カバレッジ測定機能 [IECUBE]」参照）。この機能は、RRM 機能、トレース機能との排他使用です（「5.11.4 RRM 機能／トレース機能／カバレッジ機能間の排他使用に関して」参照）。カバレッジ・ボード非搭載時は無効です。選択時、以下のメニューは選択不可となります。 [トレーサ停止／開始]、[無条件トレース／条件トレース]、[トレーサ制御モード]、[RRM 設定 ...]
------------------	---

## (5) [実行] メニュー

リスタート	CPU をリセットしてからプログラム実行します。  ボタンのクリック時と同様です。
ストップ	プログラム実行を強制的に停止させます。  ボタンのクリック時と同様です。
継続して実行	カレント PC からプログラムを実行します。  ボタンのクリック時と同様です。
ブレークせずに実行	設定されているブレーク・ポイント（ハード、ソフトとも）を無視してプログラムを実行します。  ボタンのクリック時と同様です。
リターン・アウト	呼び出し元に戻るまで実行します。  ボタンのクリック時と同様です。 <b>注意：</b> このコマンドは、C 言語で記述した関数が対象です。
ステップ・イン	プログラム内の命令を1つずつ実行します（ステップ実行）。関数、サブルーチンが呼び出されている場合には、その内部の命令を1つずつ実行します。  ボタンのクリック時と同様です。
ネクスト・オーバー	プログラム内の命令を1つずつ実行します（Next ステップ実行）。関数、サブルーチンが呼び出されている場合には、その内部の命令をステップ実行の対象から外します。  ボタンのクリック時と同様です。
カーソル位置から実行	<a href="#">ソース・テキスト・ウィンドウ</a> 、または <a href="#">逆アセンブル・ウィンドウ</a> のカーソル位置からプログラムを実行します。
カーソル位置まで実行	カレント PC から <a href="#">ソース・テキスト・ウィンドウ</a> 、または <a href="#">逆アセンブル・ウィンドウ</a> のカーソル位置までプログラムを実行します。
自動継続実行	プログラム実行をし続けます。ブレーク条件によりブレークした場合には、ウィンドウを更新してから、再度プログラム実行します。  ボタンを、ブレークするたびにクリックするのと同じ動作をします。
スローモーション	ステップ実行をし続けます。ステップ実行ごとにウィンドウを更新してから、再度ステップ実行します。  ボタンを、ブレークするたびにクリックするのと同じ動作をします。
CPU リセット	CPU をリセットします。  ボタンのクリック時と同様です。

PC 値変更	ソース・テキスト・ウィンドウ, または逆アセンブル・ウィンドウのカーソル位置のアドレスを PC に設定します。
ブレーク・ポイント	ソース・テキスト・ウィンドウ, または逆アセンブル・ウィンドウのカーソル位置にブレーク・ポイントを設定／削除します。
ソフトウェア・ブレーク・ポイント	ソース・テキスト・ウィンドウ, または逆アセンブル・ウィンドウのカーソル位置にソフトウェア・ブレーク・ポイントを設定／削除します。
すべてのブレーク・ポイントを削除	設定されているすべてのブレーク・ポイントを削除します。
無条件トレース [IECUBE]	無条件トレースを有効にして, プログラム実行中は常にトレースするように設定します (デフォルト)。この際, 設定されているトレース・イベント条件は無視されます。トレーサ起動中は条件トレースへ変更することはできません。
条件トレース [IECUBE]	条件トレースを有効にして, プログラム実行中はトレース・イベント条件にしたがってトレースするように設定します。トレーサ起動中は無条件トレースへ変更することはできません。
トレーサ制御モード [IECUBE]	トレーサ制御モードを設定します。

ノン・ストップ	トレース・メモリを一周したら最古のフレームからオーバーライトします (デフォルト)。
フル・ストップ	トレース・メモリを一周したらトレーサを停止します。
フル・ブレーク	トレース・メモリを一周したらトレーサとプログラム実行を停止します。
ディレイ・トリガ・ストップ	ディレイ・トリガ・イベント発生時にディレイ・カウント・フレーム分トレースを取り, トレーサを停止します。

タイマ開始／ タイマ停止 [IECUBE]	タイマ計測が停止中は開始させ, 動作中は停止させます。プログラム実行中でない時, およびタイマ・イベント未使用の時は無効です。プログラム実行を開始した直後はタイマ計測動作中となります。
トレーサ開始／ トレーサ停止 [IECUBE]	トレーサが停止中は開始させ, 動作中は停止させます。プログラム実行中でない時は無効です。プログラム実行を開始した直後はトレーサ動作中となります。

## (6) [ イベント ] メニュー

イベント・マネージャ	各種イベント条件の管理を行います。 イベント・マネージャをオープンします。  ボタンのクリック時と同様です。
ソフトウェア・ブレーク・マネージャ	ソフトウェア・ブレーク・イベント条件の管理を行います。 ソフトウェア・ブレーク・マネージャをオープンします。
イベント ...	イベント条件の登録を行います。 イベント・ダイアログをオープンします。  ボタンのクリック時と同様です。
イベントリンク ...	イベント・リンク条件の登録を行います。 イベント・リンク・ダイアログをオープンします。
ブレーク ...	ブレーク条件の登録と設定を行います。 ブレーク・ダイアログをオープンします。  ボタンのクリック時と同様です。

トレース ... [IECUBE]	トレース・イベント条件の登録と設定を行います。 トレース・ダイアログをオープンします。  ボタンのクリック時と同様です。
タイマ ... [IECUBE]	タイマ・イベント条件の登録と設定を行います。 タイマ・ダイアログをオープンします。  ボタンのクリック時と同様です。
ディレイ・カウント ... [IECUBE]	ディレイ・カウントの設定をします。 ディレイ・カウント設定ダイアログをオープンします。

## (7) [ブラウザ] メニュー

リスト	関数、変数、シンボル、セクション、割り込み要求名を一覧表示します。 リスト・ウィンドウをオープンします。
ソース	ソース・テキスト表示します。 ソース・テキスト・ウィンドウをオープンします。 すでにアクティブ状態のウィンドウが存在する場合は、スタティック状態でウィンドウをオープンします。  ボタンのクリック時と同様です。
アセンブル	アセンブル表示します。 逆アセンブル・ウィンドウをオープンします。 すでにアクティブ状態のウィンドウが存在する場合は、スタティック状態でウィンドウをオープンします。  ボタンのクリック時と同様です。
メモリ	メモリ内容を表示します。 メモリ・ウィンドウをオープンします。 すでにアクティブ状態のウィンドウが存在する場合は、スタティック状態でウィンドウをオープンします。  ボタンのクリック時と同様です。
ウォッチ	ウォッチ内容を表示します。 ウォッチ・ウィンドウをオープンします。  ボタンのクリック時と同様です。
レジスタ	レジスタ内容を表示します。 レジスタ・ウィンドウをオープンします。  ボタンのクリック時と同様です。
I/O レジスタ	周辺 I/O レジスタを表示します。 IOR ウィンドウをオープンします。  ボタンのクリック時と同様です。
ローカル変数	ローカル変数を表示します。 ローカル変数ウィンドウをオープンします。  ボタンのクリック時と同様です。
スタック	スタック内容を表示します。 スタック・トレース・ウィンドウをオープンします。  ボタンのクリック時と同様です。

トレース [IECUBE]	トレース結果を表示します。 トレース・ウィンドウをオープンします。  ボタンのクリック時と同様です。
コード・カバレッジ [IECUBE]	コード・カバレッジ測定結果を表示します。 コード・カバレッジ・ウィンドウをオープンします。  ボタンのクリック時と同様です。
コンソール	コンソール・ウィンドウをオープンします。
その他	その他のウィンドウを表示します（「付録 A 拡張ウィンドウ」参照）。 ユーザ定義のウィンドウ一覧を表示します。

## (8) [ジャンプ] メニュー

ソース	カレント・ウィンドウで選択しているデータ値をジャンプ先アドレスとして、該当するソース・テキストおよびソース行を表示します。 ただし、ジャンプ先アドレスに行情報がない場合には、ジャンプすることはできません。 ソース・テキスト・ウィンドウをオープンします。 アクティブ状態のソース・テキスト・ウィンドウが表示されている場合は、そのウィンドウを最前面（操作対象）に表示します。
アセンブル	カレント・ウィンドウで選択しているデータ値をジャンプ先アドレスとして、そのアドレスから逆アセンブル表示します。 逆アセンブル・ウィンドウをオープンします。 アクティブ状態の逆アセンブル・ウィンドウが表示されている場合は、そのウィンドウを最前面（操作対象）に表示します。
メモリ	カレント・ウィンドウで選択しているデータ値をジャンプ先アドレスとして、そのアドレスからメモリ内容を表示します。 メモリ・ウィンドウをオープンします。 アクティブ状態のメモリ・ウィンドウが表示されている場合は、そのウィンドウを最前面（操作対象）に表示します。

## (9) [ウィンドウ] メニュー

新しいウィンドウを開く	カレント・ウィンドウと同じ内容を表示する新しいウィンドウをオープンします。カレント・ウィンドウがソース・テキスト・ウィンドウ、逆アセンブル・ウィンドウ、メモリ・ウィンドウの場合のみ有効です。
重ねて表示	メイン・ウィンドウ内のウィンドウをカスケード表示にします。
並べて表示	メイン・ウィンドウ内のウィンドウをタイル表示にします。
アイコンの整列	メイン・ウィンドウ内のアイコンを再配置します。
すべてのウィンドウを閉じる	メイン・ウィンドウを除く、すべてのウィンドウをクローズします。
最新のデータに更新	ウィンドウの内容を最新のデータに更新します。
アクティブ	ウィンドウをアクティブ状態に切り替えます。
スタティック	ウィンドウをスタティック状態に切り替えます。
(オープンウィンドウ)	オープンしているウィンドウの一覧を表示します。 チェック・マークの付いているウィンドウがカレント・ウィンドウです。ウィンドウ名を選択することにより、選択したウィンドウをカレント・ウィンドウにします。

## (10) [ヘルプ]メニュー

ID850QB のヘルプ	製品のヘルプをオープンします。
コマンド・リファレンス	<a href="#">コマンド・リファレンス</a> のヘルプを表示します。
メイン・ウインドウ	メイン・ウインドウのヘルプを表示します。
カレント・ウインドウ	カレント・ウインドウのヘルプを表示します。
バージョン情報 ...	ID850QB のバージョンを表示します。 <a href="#">バージョン表示ダイアログ</a> をオープンします。

## ツールバー

## (1) 各ボタンの意味

## (2) ツールバーの操作

## (1) 各ボタンの意味

ツールバーの各ボタンの意味は、次の通りです。なお、ツールバーの各ボタン上にマウス・カーソルを置くと、数秒後にツール・ヒントがポップアップされます。

 Stop	ユーザ・プログラム実行を停止します。 [実行]メニュー→[ストップ]選択時と同様です。
 ReGo	CPU をリセットしてから、ユーザ・プログラム実行します。 [実行]メニュー→[リスタート]選択時と同様です。
 Go	CPU をリセットせずに、カレント PC からユーザ・プログラム実行します。 [実行]メニュー→[継続して実行]選択時と同様です。
 Go	設定されているブレーク・ポイントを無視して、ユーザ・プログラム実行します。 [実行]メニュー→[ブレークせずに実行]選択時と同様です。
 Ret	呼び出し元に戻るまで実行します。 [実行]メニュー→[リターン・アウト]選択時と同様です。 <b>注意：</b> このコマンドは、C 言語で記述した関数が対象です。
 Step	ステップ実行（プログラム内の命令を1つずつ実行）します。 関数、サブルーチンが呼び出されている場合には、その内部の命令を1つずつ実行します。 [実行]メニュー→[ステップ・イン]選択時と同様です。
 Over	Next ステップ実行（関数/コール文を1ステップとみなしプログラムを実行）します。 関数、サブルーチンが呼び出されている場合には、その内部の命令をステップ実行の対象から外します。 [実行]メニュー→[ネクスト・オーバー]選択時と同様です。
 Res	CPU をリセットします。 [実行]→[CPU リセット]選択時と同様です。
 Open	<a href="#">表示ファイル・ロード・ダイアログ</a> をオープンします。 [ファイル]メニュー→[開く ...]選択時と同様です。
 Load	<a href="#">ダウンロード・ダイアログ</a> をオープンします。 [ファイル]メニュー→[ダウンロード ...]選択時と同様です。
 Proj	<a href="#">プロジェクト・ファイル・ロード・ダイアログ</a> をオープンします。 [ファイル]メニュー→[プロジェクト]→[開く ...]選択時と同様です。

 Src	ソース・テキスト表示を行います。 ソース・テキスト・ウィンドウをオープンします。 [ブラウズ]メニュー→[ソース]選択時と同様です。
 Asm	アセンブル表示します。 逆アセンブル・ウィンドウをオープンします。 [ブラウズ]メニュー→[アセンブル]選択時と同様です。
 Mem	メモリ内容を表示します。 メモリ・ウィンドウをオープンします。 [ブラウズ]メニュー→[メモリ]選択時と同様です。
 Wch	ウォッチ内容を表示します。 ウォッチ・ウィンドウをオープンします。 [ブラウズ]メニュー→[ウォッチ]選択時と同様です。
 Reg	レジスタ内容を表示します。 レジスタ・ウィンドウをオープンします。 [ブラウズ]メニュー→[レジスタ]選択時と同様です。
 IOR	周辺 I/O レジスタの内容を表示します。 IOR ウィンドウをオープンします。 [ブラウズ]メニュー→[I/O レジスタ]選択時と同様です。
 Loc	ローカル変数内容を表示します。 ローカル変数ウィンドウをオープンします。 [ブラウズ]メニュー→[ローカル変数]選択時と同様です。
 Stk	スタック内容を表示します。 スタック・トレース・ウィンドウをオープンします。 [ブラウズ]メニュー→[スタック]選択時と同様です。
 TrW [IECUBE]	トレース結果を表示します。 トレース・ウィンドウをオープンします。 [ブラウズ]メニュー→[トレース]選択時と同様です。
 Cov [IECUBE]	コード・カバレッジ測定結果を表示します。 コード・カバレッジ・ウィンドウをオープンします。 [ブラウズ]メニュー→[コード・カバレッジ]選択時と同様です。
 Mgr	イベント・マネージャをオープンします。 [イベント]メニュー→[イベント・マネージャ]選択時と同様です。
 Evn	イベントの登録, 設定を行います。 イベント・ダイアログをオープンします。 [イベント]メニュー→[イベント ...]選択時と同様です。
 Brk	ブレーク・イベントの登録と設定を行います。 ブレーク・ダイアログをオープンします。 [イベント]メニュー→[ブレーク ...]選択時と同様です。
 Trc [IECUBE]	トレース・イベントの登録と設定を行います。 トレース・ダイアログをオープンします。 [イベント]メニュー→[トレース ...]選択時と同様です。
 Tim [IECUBE]	タイマ・イベントの登録と設定を行います。 タイマ・ダイアログをオープンします。 [イベント]メニュー→[タイマ ...]選択時と同様です。

## (2) ツールバーの操作

ツールバーは、[オプション]メニュー → [ツールバー]により、表示／非表示を選択できます。

ツールバーは、[デバッガ・オプション設定ダイアログ](#)での設定により次のように表示切り替えが可能です。

図 6 - 2 ツールバー (Picture only)



図 6 - 3 ツールバー (Picture and Text)



## ウィンドウ表示エリア

各種デバッグ・ウィンドウを表示するエリアです。

表示されたウィンドウは、このエリアの中でウィンドウ・サイズの変更、アイコン化などを行います。

## ステータスバー

ID850QB, およびインサーキット・エミュレータの状態を示すエリアです。

また、ユーザ・プログラム実行中には、ステータスバーの表示色が**赤色**に変わります。

ステータスバーは、[オプション]メニュー → [ステータスバー]により、表示／非表示を選択できます。

**注意** 画面の解像度が小さい場合（800 × 600 など）ステータスバーのすべてのステータスが表示されない場合があります。

図 6 - 4 ステータスバー



(1) プログラム名	PC 値で示されるプログラム・ファイル名を表示
ソース名	PC 値で示されるソース・ファイル名を表示
行番号	PC 値で示される行番号を表示
(2) 関数名	PC 値で示される関数名を表示
(3) PC 値	現在の PC 値を表示
(4) CPU ステータス	「表 6 - 2 CPU ステータス」参照
(5) IE ステータス	「表 6 - 3 IE ステータス」参照 (複数ある場合は '   ' で区切って表示)
(6) ブレーク要因	「表 6 - 4 ブレーク要因」参照

(7) STEP モード	ステップ実行モードを表示 [オプション]メニューで次のモードが選択されていることを表示 SRC..... ソース・モード INST..... 命令モード AUTO..... 自動モード
(8) キー入力モード	キー入力モードを表示 INS..... 挿入モード OVR..... 上書きモード ただし、メモリ・ウィンドウでは、OVR モード固定です。

表 6-2 CPU ステータス

表示	意味
HALT	ホールド・モード中
STOP/IDLE	ソフトウェア・ストップ、ハードウェア・ストップ、アイドルモード中
HOLD	バス・ホールド中
WAIT	ウェイト状態
RESET	リセット状態
POW OFF	ターゲットに電源が供給されていない

表 6-3 IE ステータス

表示	意味
RUN	ユーザ・プログラム実行中（ステータスバーの色が変化）
STEP	ステップ実行中
TRC	トレース実行中 [IECUBE]
TIM	タイマ動作中 [IECUBE]
COV	カバレッジ動作中 [IECUBE]
BREAK	ブレーク中
実行時間	ブレーク時にユーザ・プログラムの実行開始からブレーク発生までの時間（Run-Break 時間）の測定結果を表示 <sup>注</sup>
TIMER OVERFLOW	測定結果がオーバーフローした

**注** 最小 20n 秒から最大 195.2 時間（4K 分周時）程度まで測定可能です。[IECUBE]

最小 200n 秒から最大約 7 分まで測定可能です（DCK=10MHz）。[MINICUBE]

最小 100u 秒から最大 100 時間まで測定可能です。[MINICUBE2]

表6-4 ブレーク要因

表示	意味
Manual Break	強制ブレーク
Temporary Break	テンポラリ・ブレーク
Software Break	ソフトウェア・ブレーク
Trace Full Break	トレース・フルによるブレーク [IECUBE]
Non Map Break	ノン・マップ・エリアをアクセスした [IECUBE]
Write Protect	ライト・プロテクト領域に対してライトしようとした [IECUBE]
IOR Illegal	周辺 I/O レジスタに対してイリーガルなアクセスを行った [IECUBE]
Timer Over Break	実行時間オーバーを検出した [IECUBE]
Flash Macro Service	フラッシュ・マクロ・サービス中 [MINICUBE]
IRAM Write Protect (xxx xxx)	ブレーク時に IRAM ガード領域のベリファイ・チェックを行い、値が書き換わっていた [IECUBE] xxx xxx は該当アドレス、該当データ（複数ある場合は最初のみ表示）
Illegal Opcode Trap	不正命令例外発生によるブレーク [IECUBE]
Event Break: " イベント名 or イベント・リンク名 "	表示したイベント名、またはイベント・リンク名のイベントにより停止した

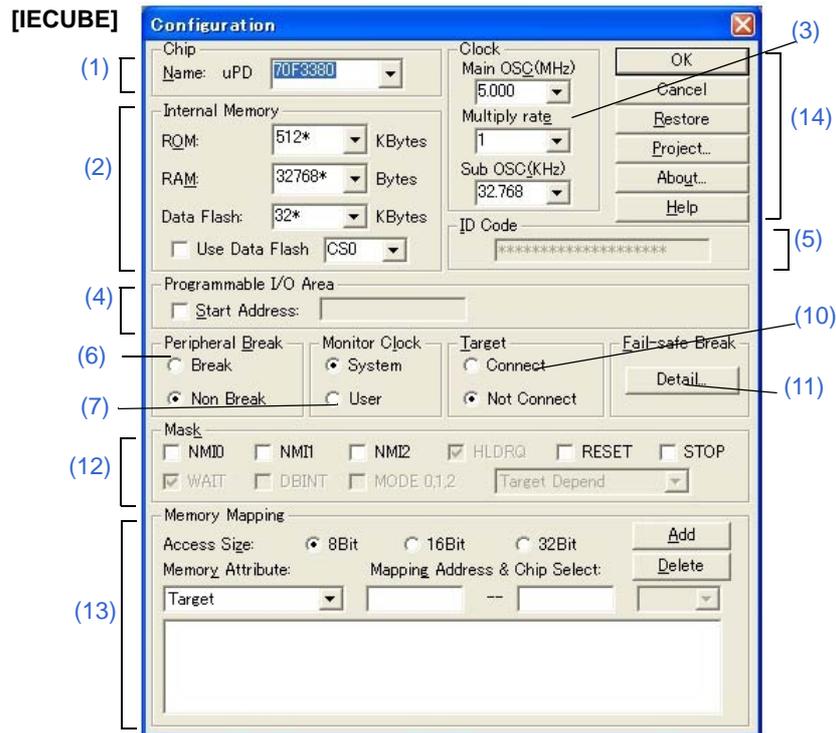
## コンフィギュレーション・ダイアログ

ID850QB の動作環境の表示と設定を行います（「5.1 デバッグ環境の設定」参照）。

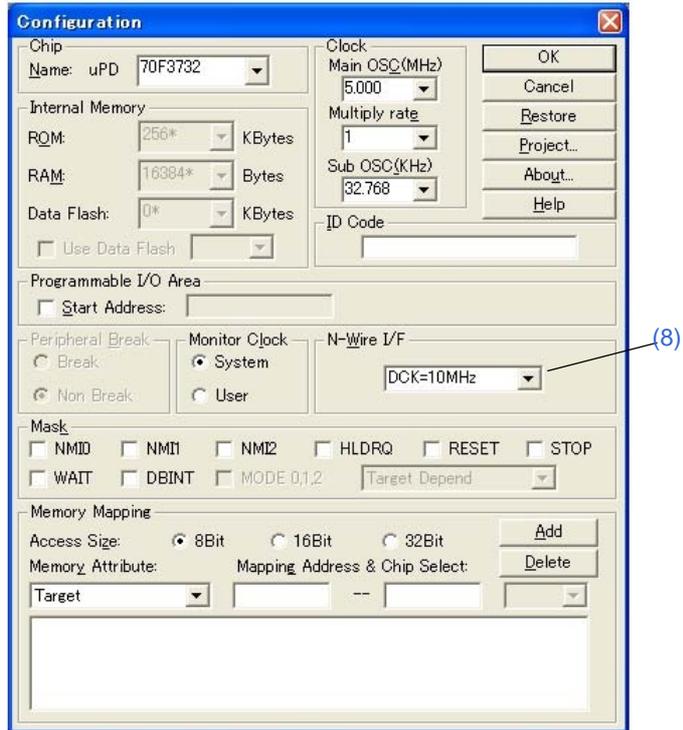
このダイアログは、ID850QB 起動後に自動的に表示されます。

なお、プロジェクト・ファイルを読み込む場合には、プロジェクト・ファイルを読み込んだ結果がダイアログ上に反映されます（「5.15.1 デバッグ環境（プロジェクト・ファイル）」参照）。

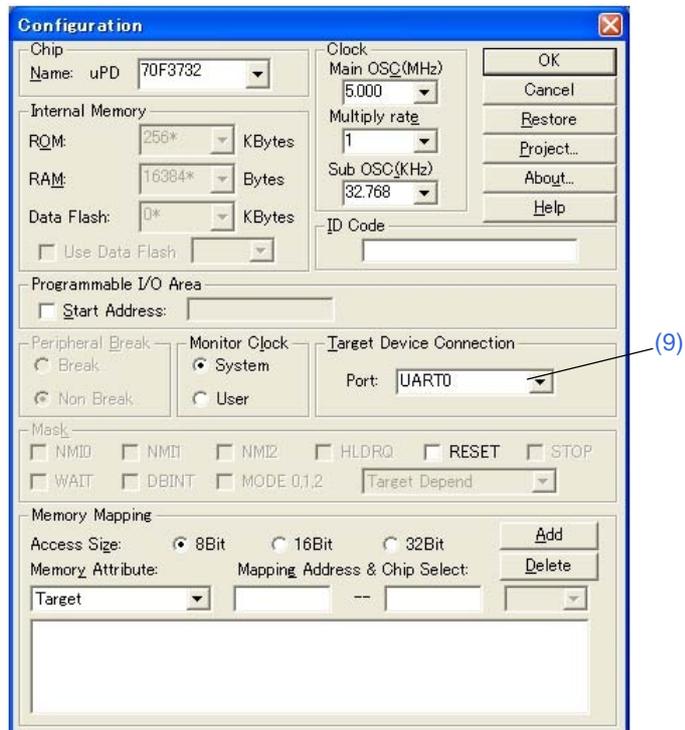
図 6-5 コンフィギュレーション・ダイアログ



[MINICUBE]



[MINICUBE2]



- オープン方法
- 各エリア説明

## オープン方法

(ID850QB を起動したとき自動的にオープン)

[オプション]メニュー→[コンフィギュレーション...]を選択

## 各エリア説明

### (1) Chip (CPU 選択エリア)

使用する品種名を選択するエリアです。指定は、ドロップダウン・リストからの選択により行います。

ドロップダウン・リストには、デバイス・ファイル・インストーラによりレジストリ登録された品種のみが表示されます。このエリアは起動時のみ指定可能です。

**参考** デフォルトでは前回起動時に選択した品種が表示されますが、その品種が登録されていない場合には、登録されている品種の先頭のみが表示されます。

### (2) Internal Memory (内部メモリ選択エリア)

下記に示す内部メモリの各サイズを設定します。

設定は、ドロップダウン・リストからの選択、またはキーボードからの入力により行います。

デフォルトのサイズは、"Chip"での選択によりデバイス・ファイルから取得され、表示されます(' \* '付きの値)。

**参考** このエリアは選択できません。デバイス・ファイルの値で固定になります。[MINICUBE] [MINICUBE2]

領域	意味	設定範囲
ROM:	内部 ROM サイズの設定	0, 8, 32, 64, 128, 256, 512, 1024 (K バイト)
RAM:	内部 RAM サイズの設定	4096, 12288, 28672, 61440 (バイト)
Data Flash:	データ・フラッシュ・メモリを設定します。(データ・フラッシュ・メモリ搭載品) データ・フラッシュ・メモリ領域を使用する場合、"Use Data Flash:" のチェックボックスをチェックします。 V850ES シリーズの場合、"チップ・セレクト・エリア"でデータ・フラッシュ・メモリをマッピングする際のチップ・セレクトを選択します。	
Use Data Flash:		
チップ・セレクト・エリア		

### (3) Clock (CPU クロック・ソース選択エリア)

メイン・クロック、サブ・クロックに関する設定を行います。

Main OSC (MHz)	メイン・クロックの通倍前の周波数を指定します。 ドロップダウン・リストからの選択 (5.000, 8.000, 13.500, 18.000 のいずれか)、または直接入力が可能です。
Multiply rate	メイン・クロックの通倍を指定します。 ドロップダウン・リストからの選択 (1 - 10 のいずれか)、または直接入力が可能です。
Sub OSC (KHz)	サブ・クロックの周波数を指定します (サブ・クロックのない品種では設定不可)。 ドロップダウン・リストからの選択、または直接入力が可能です。

**(4) Programmable I/O Area (プログラマブル I/O 領域アドレス設定エリア)**

プログラマブル I/O 領域の使用と開始アドレスを指定します。

" Chip " での選択品種がプログラマブル I/O 領域をサポートしている場合に限り指定可能です。プログラマブル I/O 領域を使用する場合には、チェック・ボックスをチェックし、プログラマブル I/O 領域の開始アドレスを入力します。なお、アドレスは 16K バイトでアラインされます。

**参考** " Chip " での選択品種がアドレス固定の拡張 I/O 領域を持つ品種である場合、このエリアの設定は自動的に行われます。

**(5) ID Code (ID コード入力エリア) [MINICUBE] [MINICUBE2]**

ID850QB から内蔵 ROM、または内蔵フラッシュ・メモリ上のコードを読み出す際の ID コードを入力します (ID コード認証)。

ID コードは 16 進数で 20 桁 (10 バイト) を入力します (デフォルトはすべて 'F')。

ID コードはレジストリに保存されます。なお、ID コードの入力に 3 回失敗すると ID850QB は強制終了します。

**参考 1** ROM レス品、または RSU (セキュリティ・ユニット) 未搭載品の場合、このエリアは設定不可です。

**参考 2** ID コード認証に関する詳細は、N-Wire CARD, MINICUBE, または MINICUBE2 のユーザーズ・マニュアルを参照してください。

**(6) Peripheral Break (ブレイク・モード選択エリア)**

ブレイク時のインサーキット・エミュレータの周辺エミュレーション機能の動作を選択します。

Break	周辺エミュレーション機能を停止させます。
Non Break	周辺エミュレーション機能を停止させません (デフォルト)。

**(7) Monitor Clock (モニタ動作クロック選択エリア) [IECUBE]**

ブレイク中のモニタ・プログラムの動作クロックを、サブ・クロックからメイン・クロックに切り替えるかどうか選択します。

サブ・クロックのない品種では設定不可です。

System	メイン・クロックに切り替えモニタ・プログラムを動作させます (デフォルト)。 <b>注意:</b> ID850QB は、PCC を操作してクロック切り替えを行いますが、メイン・クロックが停止している場合には切り替えを行いません。また、ブレイク時にメイン・クロックに切り替えた場合でも、ユーザ・プログラムに戻る際には、切り替える前の設定にクロックを戻します。
User	ユーザ・プログラムで設定したクロック選択のままモニタ・プログラムを動作させます。

**(8) N-Wire I/F (DCK クロック選択エリア) [MINICUBE]**

N-Wire CARD から、オンチップ・デバッグ・ユニット (DCU) へ供給するクロックを選択します。デフォルトは、10MHz のクロックを供給します。

**注意** 通常は 10MHz を選択してください。20MHz を選択した場合、ID850QB が起動できない場合があります。

DCK=10MHz	DCK は 10MHz になります (デフォルト)。 <b>参考:</b> "DCK=20MHz" を選択した場合に比べて、実行時間測定機能の計測時間の最大値は 2 倍に、分解能は 1/2 になります。
DCK=20MHz	DCK は 20MHz になります。

**(9) Target Device Connection (ターゲット・デバイス接続ポート設定エリア) [MINICUBE2]**

MINICUBE2 が、ターゲット・システム上のデバイスとシリアル通信を行う際に接続するポートを選択します。

MINICUBE2 では、通信インタフェースとして UART と CSI - H/S をサポートしています。

なお、選択可能なポートの種類は品種依存です。

**(10) Target (ターゲット選択エリア) [IECUBE]**

インサーキット・エミュレータにターゲットを接続するかどうかを指定します (「[表 3-2 エラー・メッセージの出カパターン \[IECUBE\]](#)」参照)。

Connect	接続する
Not Connect	接続しない

**参考** この設定は、不正な電源状態を検出するために使用しています。デフォルトの設定は、ターゲット・ボードの電源 (TVDD) 検出により決まります。

**(11) Fail-safe Break (フェイルセーフ・ブレイク選択エリア) [IECUBE]**

フェイルセーフ・ブレイク機能を選択するエリアです。

<Detail...> ボタンをクリックすることによりオープンする [フェイルセーフ・ブレイク設定ダイアログ](#) で、フェイルセーフ・ブレイク機能を個別に選択可能です。

**(12) Mask (マスク設定エリア)**

ターゲット・システムから送られてくる信号のマスク指定を行います。

マスク指定を行う端子をチェックしてください。

マスク指定を行った端子の信号はインサーキット・エミュレータには入力されません。

なお、端子のマスク指定は、デバッグ段階でターゲット・システムの動作が不安定なときのみ行ってください。

**参考 1** "RESET" のチェックにより、外部リセット、およびウォッチ・ドッグ・タイマなどにより発生する内部リセットがマスク可能です。この際、内部リセットのマスクが可能かどうかは品種依存となります。

**参考 2** TM タグに対応しているデバイス・ファイルを使用している場合には、Mode00 - Mode1F から選択可能です (表示される Mode はデバイス・ファイルの定義により決まります)。なお、IECUBE 接続時でターゲット接続時は、Target Depend が選択可能です。

**(13) Memory Mapping (マッピング設定エリア)**

マッピング設定を行います。

"Access Size" でメモリ・アクセス・サイズを選択、" Memory Attribute " でマッピング属性を指定し、" Mapping Address " でアドレス範囲を指定します。

<Add> ボタンをクリックすることによりマッピング設定され、リスト表示されます。

マッピング可能範囲は機種依存です。

Access Size	メモリ・アクセス・サイズを選択します。 これらの設定は、ID850QB のソフトウェア上のアクセス・サイズを指定するもので、外部バスのハードウェア上の動作は、MODE 端子の設定や周辺 I/O レジスタの設定に従います。	
	8Bit	ld.b 命令 / st.b 命令でメモリをアクセスします。
	16Bit	ld.h 命令 / st.h 命令でメモリをアクセスします。
	32Bit	ld.w 命令 / st.w 命令でメモリをアクセスします。

Memory Attribute	マッピング属性は次の種類が選択できます。用途に合わせて選択してください（「 <a href="#">表 5-2 マッピング属性の種類</a> 」参照）。	
	Emulation ROM [IECUBE]	（メモリ・ボード搭載時） インサーキット・エミュレータ代替 ROM を選択します。 マッピング単位：1M バイト
	Emulation RAM [IECUBE]	（メモリ・ボード搭載時） インサーキット・エミュレータ代替 RAM を選択します。 マッピング単位：1M バイト
	Target	ターゲット・メモリを選択します。 マッピング単位：1M バイト
	Target ROM [IECUBE]	ターゲット ROM を選択します。
	I/O Protect	I/O プロテクト領域を選択します。 この領域は、Target に指定した領域内に設定可能です。 I/O プロテクト領域に設定した領域は、 <a href="#">メモリ・ウィンドウ</a> 上でマッピングされていない領域と同様に表示（表示記号：??）され、 <a href="#">メモリ・ウィンドウ</a> からこの領域に対する自由な読み込み／書き込みはできなくなるため、誤ったアクセスからの保護が可能で <b>注</b> 。 マッピング単位：1 バイト
Mapping Address & Chip Select <b>注</b> [IECUBE]	マッピングするアドレスを指定します。 上位アドレス、下位アドレスをキーボードから入力します。 エミュレーション・メモリに指定する領域は、16M バイト（1M バイトのメモリが 16 バンク）から構成されるため、チップ・セレクト機能により、CS0 ~ CS7 のうち任意の箇所に割り当てることができます（ドロップダウン・リストにより選択）。 配置アドレスは、任意の 1 M バイト境界に割り当てることができます。 1 つの CS に複数のバンクを割り当てることができます。	
<Add> <Delete>	マッピングの設定、削除を行います。 <Add> ボタンのクリック時には、各エリアで指定されたマッピングが設定されリスト表示されます。マッピングを削除する場合には、リストから選択し、<Delete> ボタンをクリックします。	

**注** このエリアは、V850ES シリーズの場合、チップ・セレクトの割り当てが固定、もしくはチップ・セレクト機能がないため選択不可となります。IECUBE 接続時でも、オプション・ボード未装着時は選択不可です。

**注意 1** "I/O Protect" を設定した領域は、I/O ポートとして [IOR ウィンドウ](#)に登録するか、または [ウォッチ・ウィンドウ](#)に登録しない限り読み出されることはありません。読み出したい場合は、これらのウィンドウで強制読み込みを実行してください。

**注意 2** 外部メモリに対してマッピングを行った際には、外部メモリのアクセスに必要なレジスタの値を変更してください（「[5.1.4 外部メモリのアクセスに必要なレジスタの値の変更](#)」参照）。

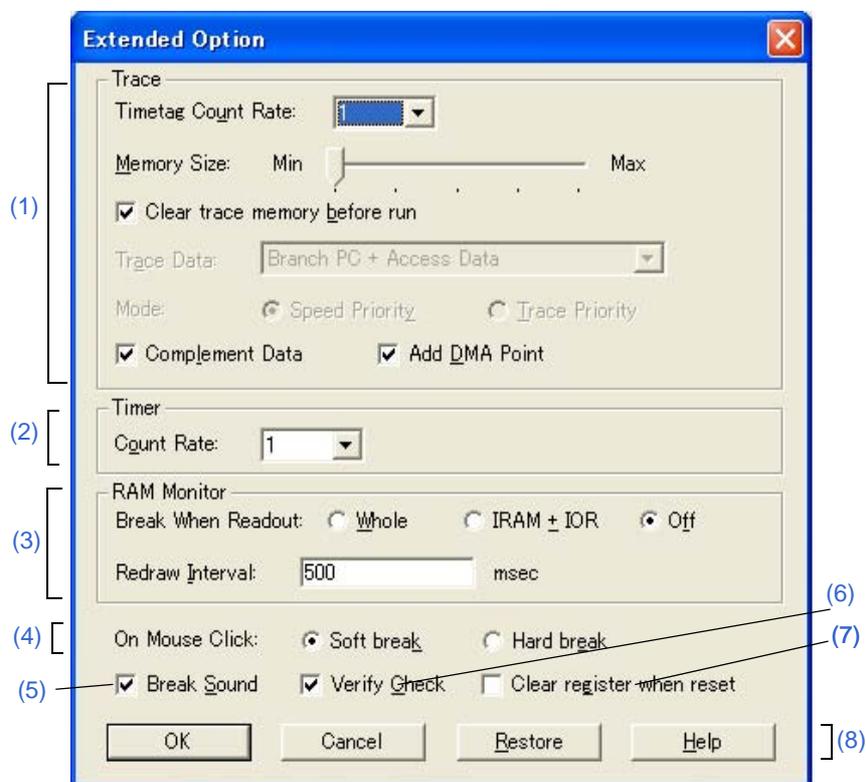
## (14) 機能ボタン

OK	現在設定した環境を有効にします。 環境を設定してこのダイアログをクローズします。
Cancel	変更を取り消して、このダイアログをクローズします。
Restore	このダイアログをオープンする前の環境設定状態に戻します。
Project...	<a href="#">プロジェクト・ファイル・ロード・ダイアログ</a> をオープンします。プロジェクト・ファイルのオープンや読み込み中にエラーが発生した場合、続行不可能なため、ID850QBを終了します。
About...	<a href="#">バージョン表示ダイアログ</a> をオープンします。
Help	このダイアログのオンライン・ヘルプを表示します。

## 拡張オプション設定ダイアログ

ID850QB の各種拡張オプションの表示と設定を行います（「5.1 デバッグ環境の設定」参照）。

図 6 - 6 拡張オプション設定ダイアログ



- オープン方法
- 各エリアの説明

### オープン方法

[オプション]メニュー → [拡張オプション...] を選択

## 各エリアの説明

### (1) Trace (トレース関連設定エリア) [IECUBE]

トレース関連の設定を行うエリアです (「5.10 トレース機能 [IECUBE]」参照)。

#### (a) Timetag Count Rate:

トレース・ウィンドウのタイムタグ表示で使用するカウンタの分周率を設定します。

分周率はドロップダウン・リストから選択します。

分周率を設定することにより、タイムタグで表示されるカウンタのカウント・アップに必要なクロック数が変更されます。

なお、タイムタグ・カウンタの分周率と最大測定時間の関係は、以下のようになります。

表 6-5 分周率と最大測定時間の関係 (タイムタグ・カウンタ (トレース))

分周率	分解能 (nsec)	測定可能最大時間	備考
TMCLK (1/1)	20	1.4 分	タイムタグ・カウンタ 32 ビット, 外部クロック 50MHz の 場合
TMCLK (1/2)	40	2.8 分	
TMCLK (1/4)	80	5.7 分	
TMCLK (1/8)	160	11.4 分	
TMCLK (1/16)	320	22.8 分	
TMCLK (1/32)	640	45.6 分	
TMCLK (1/64)	1280	1.5 時間	
TMCLK (1/128)	2560	3 時間	
TMCLK (1/256)	5120	6 時間	
TMCLK (1/512)	10240	12.2 時間	
TMCLK (1/1024)	20480	24.4 時間	
TMCLK (1/2048)	40960	48.8 時間	
TMCLK (1/4096)	81920	97.6 時間	

#### (b) Memory Size:

トレース・メモリ (バッファ) のサイズを設定します。

つまみをドラッグすることにより指定します。指定できるサイズは、小さい方から 8K (Min), 32K, 64K, 128K, 256K (Max) となっています。

**注意** 大きい値を設定するほど記録されるトレース・データ数は多くなりますが、トレース・データを読み出す際のレスポンスは悪くなります。

## (c) Clear trace memory before run

プログラム実行前にトレース・メモリをクリアする場合、チェックします。

## (d) Trace Data:

採取するトレース・データを選択します。

**備考** [オプション]メニュー→[RRM機能]選択時は、Branch PC 固定となります。

[オプション]メニュー→[カバレッジ機能]選択時は、All PC 固定となります。

表 6-6 採取するトレース・データの意味とトレース採取モードとの関係

項目	トレース範囲			
	Branch PC 分岐元と分岐 先の命令の PC 値を採取	All PC 全命令の PC 値を採取	Access PC アクセスを発生 させた命令 の PC 値を採 取	Access Data アクセス・ア ドレスとアク セス・データ を採取
Branch PC	○	-	-	-
All PC	○ <sup>*a</sup>			-
Access Data	-	-	-	○ <sup>*b</sup>
Branch PC + Access Data	○	-	-	○ <sup>*b</sup>
All PC + Access Data	○ <sup>*a</sup>			○ <sup>*b</sup>
Access Data + Access PC	-	-	○	○ <sup>*b</sup>
Branch PC + Access Data + Access PC	○	-	○	○ <sup>*b</sup>

<sup>\*a</sup> "All PC" を含むトレース・データ選択時は無条件トレースになり、同時にクオリファイ・トレースとセクション・トレースは設定できません（条件トレース設定は無効になります）。

<sup>\*b</sup> "Access Data" を含むトレース・データ選択時は、速度優先モード（(e) Mode: で Speed Priority を選択）の場合、内部 RAM 領域へのアクセスが 32 回連続するとデータの取りこぼしが発生する場合があります。

## (e) Mode:

トレース採取モードを指定します。

**備考** [オプション]メニュー→[RRM機能/カバレッジ機能]選択時は、"Speed Priority" 固定となります。

Speed Priority	速度（リアルタイム性）を優先してトレースを行うモードです。 このモードでは、採取するトレース・データにより、データの取りこぼしが発生する場合があります（「表 6-6 採取するトレース・データの意味とトレース採取モードとの関係」参照）。
Trace Priority	データ採取を優先してトレースを行うモードです（ノン・リアルタイム）。 全てのトレース・データを確実に採取するよう、データの取りこぼしが発生しうになると、内部的に CPU の実行パイプラインを一時的に止めて処理を行うため、ユーザ・プログラムに対するリアルタイム性は損なわれます。 (d) Trace Data: で Branch PC, または All PC を選択している場合、選択不可です。

## (f) Complement Data

トレース・データの補完表示を行う場合、チェックします（デフォルトはチェックあり）。

## (g) Add DMA Point

DMA ポイント・トレースを行う場合、チェックします（デフォルトはチェックあり）。

チェックにより、DMA の開始と終了フレームにマークが付きまます。

**(2) Timer (タイマ関連設定エリア) [IECUBE]**

タイマ・カウンタのレート値を設定します。

このエリアでの設定値が**タイマ・ダイアログ**の "Count Rate" に表示されます。

値は、ドロップダウン・リストから選択します。

タイマ・カウンタの分周率と最大測定時間の関係は以下のようになります。

表 6-7 分周率と最大測定時間の関係 (タイマ・カウンタ (タイマ))

分周率	分解能 (nsec)	測定可能最大時間	備考
TMCLK (1/1)	20	2.8 分	タイマ・カウンタ 33 ビット, 外部クロック 50MHz の 場合
TMCLK (1/2)	40	5.7 分	
TMCLK (1/4)	80	11.4 分	
TMCLK (1/8)	160	22.8 分	
TMCLK (1/16)	320	45.6 分	
TMCLK (1/32)	640	1.5 時間	
TMCLK (1/64)	1280	3 時間	
TMCLK (1/128)	2560	6 時間	
TMCLK (1/256)	5120	12.2 時間	
TMCLK (1/512)	10240	24.4 時間	
TMCLK (1/1024)	20480	48.8 時間	
TMCLK (1/2048)	40960	97.6 時間	
TMCLK (1/4096)	81920	195.2 時間	

**(3) RAM Monitor (RAM モニタ機能設定エリア)**

## (a) Break When Readout:

ユーザ・プログラムの実行を一瞬ブレークさせて RAM サンプリングを行う際の対象範囲を指定します  
(「5.13.2 疑似リアルタイム・モニタ機能 (Break When Readout)」参照)。

Whole	すべてのメモリ空間 <sup>注</sup> <b>注意：</b> メモリを読み出す範囲が広いため、オープンしているウィンドウが多い場合、ユーザ・プログラムを停止させている時間が長くなります。
IRAM + IOR	内蔵 RAM 領域と IOR 領域
Off	疑似リアルタイム・モニタ機能 (Break When Readout) を無効にします (デフォルト)。

**注** RRM 設定ダイアログでの指定範囲を除く。

## (b) Redraw Interval:

RAM サンプリングを行う際のサンプリング時間 (ミリ秒) を指定します。

サンプリング時間は、100 ミリ秒 (msec) 単位で 0 - 65500 まで指定できます。

0、または空欄を指定した場合はリアルタイム表示を行いません。

**(4) On Mouse Click: (デフォルト・ブレーク選択エリア)**

ソース・テキスト・ウィンドウ、または逆アセンブル・ウィンドウ上のポイント・マーク・エリアで、マウス・クリックによるブレークポイント設定を行った場合のデフォルト設定を行います (「5.4.2 ブレーク・ポイントの設定」参照)。

Soft break	ソフトウェア・ブレークを設定
Hard break	ハードウェア・ブレークを設定

**(5) Break Sound (ビーブ音指定エリア)**

ブレーク時にビーブ音を発生させる場合チェックします。

**(6) Verify Check (ベリファイ指定エリア)**

メモリ書き込み時のベリファイ・チェックを行う場合チェックします。

ベリファイ・チェックは、ダウンロード、メモリ・フィル、メモリ・コピーの実行時に行われます。また変数やデータなどをウォッチ・ウィンドウ、メモリ・ウィンドウで変更し、メモリに書き込む際にも行われます。

**注意** 内蔵フラッシュ・メモリへの書き込み (ダウンロードを含む) の際には、このエリアでのチェックの有無に関係なくベリファイ・チェックを行わず、常にフラッシュ・セルフ書き込みの内部ベリファイを行います (リード・ベリファイは行いません)。[MINICUBE]

**(7) Clear register when reset (レジスタクリア指定エリア)**

CPU リセット時にプログラム・レジスタ (r1 - r31) と EIPC, EIPSW, FEPC, FEPSW, CTPC, CTPSW, CTBP レジスタをクリアする場合、チェックします。

デフォルトはクリアしません。

**(8) 機能ボタン**

OK	設定した各種設定を有効にして、このダイアログをクローズします。
Cancel	変更を取り消して、このダイアログをクローズします。
Restore	このダイアログをオープンする前の設定状態に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

## フェイルセーフ・ブレイク設定ダイアログ

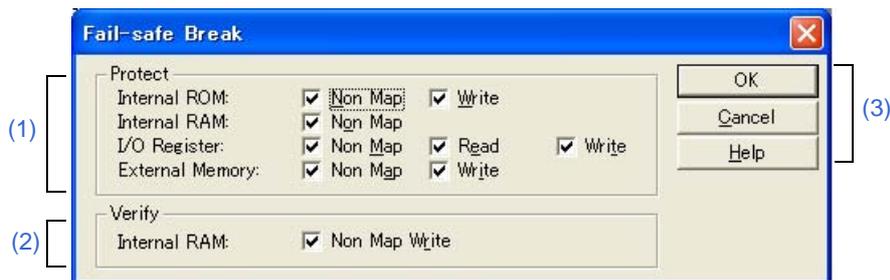
[IECUBE]

フェイルセーフ・ブレイクの設定を行います（「5.4.5 フェイルセーフ・ブレイク機能 [IECUBE]」参照）。

なお、プロジェクト・ファイルを読み込む場合には、プロジェクト・ファイルを読み込んだ結果がこのダイアログに反映されます。

**参考** フェイルセーフ・ブレイク機能の詳細に関しては、使用するインサーキット・エミュレータ、およびエミュレーション・ボードのユーザズ・マニュアルを参照してください。

図 6-7 フェイルセーフ・ブレイク設定ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

コンフィギュレーション・ダイアログの <Detail...> ボタンをクリック

### 各エリア説明

#### (1) Protect（プロテクト設定エリア）

フェイルセーフ・ブレイクのプロテクト設定を行います。

チェックした項目のフェイルセーフ・ブレイクをプロテクトします。

デフォルトでは全てチェック状態です。

Internal ROM:	内部 ROM 領域内のプロテクト設定を行います。	
	Non Map	アクセス禁止エリアへのアクセス
	Write	ライト禁止エリアへのライト
Internal RAM:	内部 RAM 領域内のプロテクト設定を行います。	
	Non Map	アクセス禁止エリアへのアクセス

I/O Register:	周辺 I/O レジスタ領域内のプロテクト設定を行います。	
	Non Map	アクセス禁止エリアへのアクセス
	Read	リード禁止エリアへのリード
	Write	ライト禁止エリアへのライト
External Memory:	外部メモリ領域内のプロテクト設定を行います。	
	Non Map	アクセス禁止エリアへのアクセス
	Write	ライト禁止エリアへのライト

**(2) Verify (ベリファイ設定エリア)**

フェイルセーフ・ブレークのベリファイ設定を行います。

チェックした項目へのアクセス時にベリファイ・チェックを行います。

Internal RAM:	内部 RAM 領域内のベリファイ・チェック設定を行います。	
	Non Map Write	ライト禁止エリアへのライト

**(3) 機能ボタン**

OK	設定を有効にして、このダイアログをクローズします。
Cancel	このダイアログをクローズします。
Help	このダイアログのオンライン・ヘルプを表示します。

## RRM 設定ダイアログ

[IECUBE]

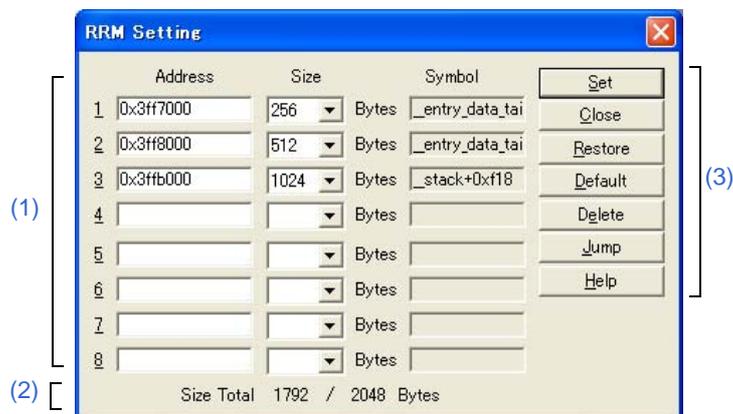
RRM 機能による読み込みを行うサンプリング範囲を設定します（「5.13 RRM 機能」参照）。

サンプリング範囲として、256 バイト単位で最大 8 箇所まで指定可能です。

なお、8 箇所指定したサイズの合計が 2048 バイトを越える値は設定できません。

**注意** [オプション]メニューで [RRM 機能] 以外の機能を選択している場合には、このダイアログはオープンできません。

図 6-8 RRM 設定ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

このダイアログはオープン方法により、オープン時の設定が次のように異なります。

(a) RRM 設定ダイアログから設定する場合

[オプション]メニュー→ [RRM 設定 ...] の選択によりオープンします。

この場合、" Address ", " Size " は手動で入力します。

(b) メモリ・ウィンドウから設定する場合

メモリ・ウィンドウをオープンし、ウィンドウ内のアドレスを選択後、コンテキスト・メニュー→ [RRM 設定 ...] を選択することによりオープンします。

この場合、空いている段の " Address " に選択したアドレス値が入り、" Size " には "256" が、" Symbol " にはアドレスをシンボル変換した値が表示されます。

**参考** 8 箇所指定したサイズの合計がすでに 2048 バイトを越えている場合には、ダイアログはオープンしますが、値の設定はできません。

## (c) ウォッチ・ウィンドウから設定する場合

ウォッチ・ウィンドウをオープンし、ウィンドウ内の変数を選択後、コンテキスト・メニュー→ [RRM 設定 ...] することによりオープンします。

この場合、空いている段の " Address " に変数をアドレス変換した値が入り、" Size " には "256" が " Symbol " には変数をシンボル変換した値が表示されます。

**参考** 8 箇所指定したサイズの合計がすでに 2048 バイトを越えている場合には、ダイアログはオープンしますが、値の設定はできません。

## 各エリア説明

### (1) サンプリング範囲設定エリア

Address	RRM 機能のサンプリング開始アドレスを指定します。 数値入力時のデフォルト進数は 16 進です。アドレス指定はシンボルや式での指定も可能です（「表 5-6 シンボルでの指定方法」参照）。 入力後、<Set> ボタンをクリックすることで設定が有効になります。
Size	" Address " からサンプリングする範囲を指定します。 選択できる値は 256, 512, 768, 1024, 1280, 1536, 1792, 2048 です。 ただし、1 から 8 までに指定した Size の合計が 2048 バイトを越える値は設定できません。
Symbol	" Address " で指定したアドレスのシンボルを表示します。 シンボル、またはシンボル+オフセットの形式で表示されます。 アドレスが確定されてない場合は空欄になります。

**参考** ID850QB では、設定を有効にする際、アドレスは 256 バイト単位でアラインされますが、アドレスが重複した場合には有効になりません。

### (2) Size Total (合計サイズ表示エリア)

" Size " で指定したサイズの合計を表示します。

合計が 2048 バイトを越えた場合、**赤色**で表示されます。

### (3) 機能ボタン

Set	指定したサンプリング範囲を確定します。
Close	このダイアログをクローズします。
Restore	このダイアログをオープンする前の設定状態に戻します。
Default	現在の設定をクリアし、1 段目の " Address " に内部 RAM の開始アドレスを、" Size " に "2048" を設定します。

Delete	フォーカスがある番号の設定を削除します。
Jump	<a href="#">メモリ・ウィンドウ</a> をオープンし、フォーカスがある番号の " Address " のアドレスを表示します。 ジャンプは、アクティブ状態の <a href="#">メモリ・ウィンドウ</a> が対象となります。 複数を開きたい場合には、メモリ・ウィンドウをスタティック状態にする必要があります（ <a href="#">「5.16.1 アクティブ状態とスタティック状態」</a> 参照）。
Help	このダイアログのオンライン・ヘルプを表示します。

## フラッシュ・オプション設定ダイアログ

[IECUBE]

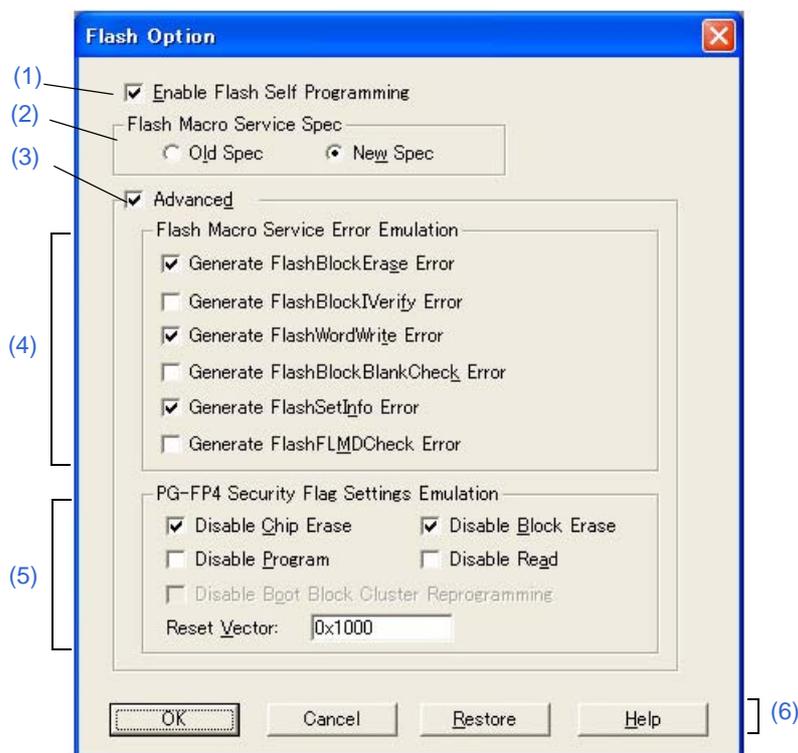
フラッシュ・セルフ・プログラミング・エミュレーション機能の設定を行います（コード・フラッシュ）。

なお、ユーザ・プログラム実行中は、このダイアログをオープンすることはできません。

対応デバイス一覧、フラッシュ関数エミュレーション可否一覧、および注意事項に関しては、「[フラッシュ・セルフ・プログラミング・エミュレーション特記事項](#)」を参照してください。

**注意** 使用デバイスがフラッシュ・メモリ搭載品でない場合、このダイアログをオープンすることはできません。

図6-9 フラッシュ・オプション設定ダイアログ



- [オープン方法](#)
- [各エリア説明](#)
- [フラッシュ・セルフ・プログラミング・エミュレーション特記事項](#)

### オープン方法

[オプション]メニュー→[フラッシュ・オプション...]を選択

## 各エリア説明

### (1) Enable Flash Programming (フラッシュ・セルフ・プログラミング・エミュレーション設定エリア)

チェックにより、フラッシュ・セルフ・プログラミング・エミュレーション機能が有効になります。  
これにより、(3) [Advanced \(拡張設定エリア\)](#) が設定可能です。  
デフォルトはチェックなしです。

### (2) Flash Macro Service Spec (フラッシュ・マクロ・サービス仕様選択エリア)

フラッシュ・セルフ・プログラミング・エミュレーション機能を有効にする場合、フラッシュ・マクロ・サービスの仕様を選択します。

このエリアはフラッシュ・メモリのプロセスが Type1 の場合に有効です。

"New Spec" を選んだ場合、(5) [PG-FP4 Security Flag Settings Emulation \(PG-FP4 セキュリティ・フラグ・エミュレーション設定エリア\)](#) の "Disable Read" と "Reset Vector" が有効になります。その際、"Reset vector" に設定するリセット・ベクタ・アドレスには、搭載フラッシュ・メモリ内のアドレスを入力してください。

Old Spec	フラッシュ・セルフ・プログラミング Ver.2.00 以前 (デフォルト)
New Spec	フラッシュ・セルフ・プログラミング Ver.3.00 以降

### (3) Advanced (拡張設定エリア)

チェックにより、フラッシュ・セルフ・プログラミング・エミュレーション機能の詳細設定が可能になります。  
これにより、(4) [Flash Macro Service Error Emulation \(フラッシュ・マクロ・サービス・エラー・エミュレーション設定エリア\)](#) と (5) [PG-FP4 Security Flag Settings Emulation \(PG-FP4 セキュリティ・フラグ・エミュレーション設定エリア\)](#) が設定可能になります。  
デフォルトはチェックなしです。

### (4) Flash Macro Service Error Emulation (フラッシュ・マクロ・サービス・エラー・エミュレーション設定エリア)

セルフ・ライブラリ関数の動作を設定します。

チェックにより、フラッシュ・メモリがダメージを受けた場合に返るエラー値を強制的に返すことができます (フラッシュ・メモリがダメージを受けた場合に返るエラーは通常のエミュレーションでは返ることがありません)。

設定できる項目は以下の通りです。全ての項目はデフォルトでチェックなしです。

Generate FlashBlockErase Error	FlashBlockErase 関数でエラーを返す
Generate FlashBlockIVerify Error	FlashBlockIVerify 関数でエラーを返す
Generate FlashWordWrite Error	FlashWordWrite 関数でエラーを返す
Generate FlashBlockBlankCheck Error	FlashBlockBlankCheck 関数でエラーを返す
Generate FlashSetInfo Error	FlashSetInfo 関数でエラーを返す
Generate FlashFLMDCheck Error <sup>注</sup>	FlashFLMDCheck 関数でエラーを返す

**注** IECUBE は、FLMD0 端子の値を読み出すことができません（通常、FlashFLMDCheck 関数は、FLMD0 端子にハイ・レベルが入力されているものとして常に正常終了を返しています）。  
このため、FlashFLMDCheck で FLMD0 端子にロウ・レベルが入力された場合のエラーを強制的に返したい時は、この "Generate FlashFLMDCheck error" をチェックして下さい。

#### (5) PG-FP4 Security Flag Settings Emulation (PG-FP4 セキュリティ・フラグ・エミュレーション設定エリア)

フラッシュ・ライタ PG-FP4 を用いてフラッシュ・メモリにセキュリティを設定した場合の、セキュリティ・フラグの初期値をエミュレーションします。

設定できる項目は以下の通りです。全ての項目はデフォルトでチェックなしです。

Disable Chip Erase	チップ消去禁止
Disable Block Erase	ブロック消去禁止
Disable Program	ライト禁止
Disable Read <sup>注1</sup>	リード禁止
Disable Boot Block Cluster Reprogramming <sup>注2</sup>	ブート領域書き換え禁止
Reset Vector: <sup>注1</sup>	リセット・ベクタ・アドレス設定

**注1** フラッシュ・メモリのプロセスが Type1 で、フラッシュ・セルフ・プログラミング Ver.3.00 以降の場合のみ有効

**注2** フラッシュ・メモリのプロセスが Type2, または Type3 の場合のみ有効

#### (6) 機能ボタン

OK	設定を有効にして、このダイアログをクローズします。
Cancel	このダイアログをクローズします。
Restore	このダイアログをオープンする前の設定状態に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

## フラッシュ・セルフ・プログラミング・エミュレーション特記事項

### (1) 対応デバイス一覧

以下に、フラッシュ・セルフ・プログラミング・エミュレーション機能の対応デバイスを記述します。

この情報は、2006年2月現在のものです。

表6-8 フラッシュ・セルフ・プログラミング・エミュレーション対応デバイス

フラッシュ・メモリのプロセス	デバイス
Type1	V850ES/Sx2, V850ES/Fx2, V850E/RS1, V850E/IA4, $\mu$ PD70F3229Y, V850ES/Hx2, V850ES/Jx2
Type3	V850ES/Kx1 (単電源フラッシュ・メモリ品のみ), V850ES/Kx1+, V850ES/Kx2

### (2) フラッシュ関数のエミュレーション可否一覧

以下に、ID850QBのフラッシュ・セルフ・プログラミング・エミュレーション機能におけるフラッシュ関数のエミュレーション可否、および制限事項を表記します（○：エミュレーション可能、△：制限付きでエミュレーション可能、×：エミュレーション不可）。

表6-9 フラッシュ関数エミュレーション可否一覧 (Type1)

フラッシュ関数名	機能概要、および制限	エミュレーション可否	
<b>Type1 (V850ES/Sx2, V850ES/Fx2, V850E/RS1, V850E/IA4, <math>\mu</math> PD70F3229Y, V850ES/Hx2, V850ES/Jx2)</b>			
FlashEnv	フラッシュ環境初期化/終了関数	○	
FlashBlockErase	1ブロック消去関数	○	
FlashWordWrite	1ワード分の書き込み関数 <b>(制限)</b> 第三引数にガード領域が指定されていた場合、意図しないアドレスでフェイル・セーフ・ブレークが発生します。	△	
FlashBlockIVerify	1ブロックの内部ベリファイ処理関数	○	
FlashBlockBlankCheck	1ブロックのブランク・チェック関数	○	
FlashGetInfo	フラッシュ情報取得関数		
	Option = 2	CPU番号、およびCPUで持っているブロック総数 <b>(制限)</b> CPU番号はコンフィギュレーションで設定したデバイス名称(4桁番号)が返却されます。	△
	Option = 3	セキュリティ情報	○
	Option = 4	ブート領域入れ替え情報取得 <b>(制限)</b> ブート領域入れ替え情報が反映されません。	△
	Option = 5 + ブロック番号	ブロックの最終アドレス取得	○

フラッシュ関数名	機能概要, および制限	エミュレーション可否
FlashSetInfo	フラッシュ情報設定関数 (制限) ブート領域入れ替え設定が無視されます。	△
FlashStatusCheck	直前に実行したフラッシュ関数の動作状況確認関数 (制限) FlashBlockErase 関数, および FlashBlockBlankCheck 関数で FE_BUSY から FE_OK になるタイミングが実デバイスと異なります。	△
FlashBootSwap	ブート領域のブロック入れ替え関数	×
FlashSetUserHandler	ユーザ割り込みハンドラ登録関数	○
FlashFLMDCheck	FLMD0 端子の状態チェック関数	○
FlashSetInfoEx	フラッシュ情報設定関数 (制限) ブート領域入れ替え設定が無視されます。	△注
FlashNWordRead	nワード分の読み出し関数 (制限) 第三引数にガード領域が指定されていた場合, 意図しないアドレスで フェイルセーフ・ブレイクが発生します。	△注

注 フラッシュ・セルフ・プログラミング Ver5.00 以降で追加された関数

表 6 - 10 フラッシュ関数エミュレーション可否一覧 (Type3)

フラッシュ関数名	機能概要, および制限	エミュレーション可否
<b>Type3 (V850ES/Kx1 (単電源フラッシュ・メモリ品のみ), V850ES/Kx1+, V850ES/Kx2)</b>		
FlashEnv	フラッシュ環境初期化/終了関数	○
FlashBlockErase	1 ブロック消去関数	○
FlashWordWrite	1ワード分の書き込み関数 (制限) 第三引数にガード領域が指定されていた場合, 意図しないアドレスで フェイルセーフ・ブレイクが発生します。	△
FlashBlockIVerify	1 ブロックの内部ベリファイ処理関数	○
FlashBlockBlankCheck	1 ブロックのブランク・チェック関数	○

フラッシュ関数名	機能概要, および制限		エミュレーション可否
FlashGetInfo	フラッシュ情報取得関数		
	Option = 2	CPU 番号, および CPU で持っているブロック総数 <b>(制限)</b> CPU 番号はデバイス・ファイルの名称 (4 桁番号) が返却されます。	△
	Option = 3	セキュリティ情報	○
	Option = 4	ブート領域入れ替え情報取得 <b>(制限)</b> ブート領域入れ替え情報が反映されません。	△
	Option = 5 + ブロック番号	ブロックの最終アドレス取得	○
FlashSetInfo	フラッシュ情報設定関数 <b>(制限)</b> ブート領域入れ替え設定が無視されます。		△
FlashBootSwap	ブート領域のブロック入れ替え関数		×
FlashFLMDCheck	FLMD0 端子の状態チェック関数		○
FlashWordRead	データ読み出し関数 <b>(制限)</b> 第三引数にガード領域が指定されていた場合, 意図しないアドレスでフェイルセーフ・ブレークが発生します。		△
FlashIVerify	内部ベリファイ関数 (EEPROM 専用)		×
FlashBlankCheck	ブランク・チェック関数 (EEPROM 専用)		×
EEPROM_Init	EEPROM 領域初期化関数 (EEPROM 専用)		×
EEPROM_Write	EEPROM 書き込み関数 (EEPROM 専用)		×
EEPROM_Read	EEPROM 読み出し関数 (EEPROM 専用)		×
EEPROM_Copy	EEPROM コピー関数 (EEPROM 専用)		×
EEPROM_VChK	EEPROM 有効領域チェック関数 (EEPROM 専用)		×
EEPROM_Erase	EEPROM 消去関数 (EEPROM 専用)		×

## (3) 制限事項一覧

以下に、フラッシュ・セルフ・プログラミング・エミュレーションを行う上での制限事項を表記します。

No.	制限事項
1	<p>以下の場合、フラッシュ・セルフ・プログラミング・エミュレーション機能を有効にできません。</p> <p>(a) 内蔵 ROM サイズをデフォルト・サイズ以外に設定した場合 回避策：<a href="#">コンフィギュレーション・ダイアログ</a>で内蔵 ROM サイズをデフォルト・サイズに戻してください。</p> <p>(b) 実行前ブレークを 2 個使用している場合 回避策：実行前ブレーク 1 つを無効、または削除してください。</p>
2	<p>フラッシュ・セルフ・プログラミング・エミュレーション機能を有効にすると、以下のデバッグ機能に制限が付きまます。</p> <p>(a) 内蔵 ROM と内蔵 RAM のサイズ変更はできません。</p> <p>(b) DMM 機能と疑似 RRM 機能は使用できません。</p> <p>(c) SP レジスタが適当な位置 (内蔵 RAM 等) に初期化される前にイベント等のブレークが発生した場合は、スタック・エリアに対する不正ブレーク要因となります。 この間にブレークが発生する可能性がある場合は、プログラム実行前に SP を適当な値にしてください。</p> <p>(d) <a href="#">拡張オプション設定ダイアログ</a>で "Clear register when reset" (リセット時のレジスタ・クリア) を選択すると、リセット・エミュレーションでフラッシュ・マクロ・サービスが変更した r3 レジスタの値がクリアされます。</p> <p>(e) ご使用の IECUBE で、下記制限事項が存在する場合、不正ブレークが発生する可能性があります。この場合、<a href="#">フェイルセーフ・ブレーク設定ダイアログ</a>で内蔵 RAM の Non Map のチェックをはずしてください。 - 内蔵 RAM でプログラム実行時のイリーガル・ブレーク制限事項</p>
3	<p>フラッシュ・セルフ・プログラミング・エミュレーション機能を有効に設定すると、0 番地から 4 バイト分は予約領域になり、0 番地に jr 0xffffd6 の 4 バイト命令が書き込まれます。 そのため、リセット・ベクタ 0 番地で使用の際には、スタートアップ・ルーチンは 4 番地から配置してください。また、フラッシュ・セルフ・プログラミングを有効から無効に設定すると、0 番地から 4 バイト分は 0 が書き込まれます。 リセット・ベクタ 0 番地以外で使用する際も 0 番地に分岐させるコードは記述しないでください。 FlashNWordRead, または FlashWordRead で予約領域を読み出したときは、0 の値が読めます。 実デバイスでも同一のプログラムを動作させるためには、以下に記述するコードを作成してください。</p> <pre data-bbox="277 1384 1134 1507"># RESET handler (0 番地の場合) .section      "RESET", text jr          __start          --jr 0xffffd6 に書き換えられる jr          __start</pre>
4	<p>リセット・ベクタ・ハンドリング指定アドレスに、0 番地を指定するとリセット・ベクタは 4 番地になります。0 番地以外を指定すると +4 されずに指定アドレスがリセット・ベクタになります。</p>
5	<p>FlashBlockErase() と FlashBlockBlankCheck() 後の FlashStatusCheck() の動作に関してエミュレーション時は、FlashStatusCheck() の返却値が FE_BUSY から FE_OK になるまでのタイミングが実デバイスと異なります。</p>
6	<p>FlashWordWrite, FlashWordRead, FlashNWordRead の第 3 引数に指定したアドレスがガード領域であった場合、不正なメモリ・アクセスとなり、意図しないアドレスでフェイル・セーフ・ブレークが発生します。 FlashWordWrite, FlashWordRead, FlashNWordRead で指定するアドレスを適切なアドレスに修正してください。</p>
7	<p><a href="#">フラッシュ・オプション設定ダイアログ</a>の各設定を有効にするためには、設定後、必ず CPU リセットして、再実行するようにしてください。 CPU リセットを行わず再実行した場合、設定が反映されない場合があります。</p>

No.	制限事項
8	デバッガのワーク領域として、最低 84(54H) バイトのスタックを確保してください。ブレーク時やフラッシュ書き換えエミュレーション処理時に最低 84 (54H) バイトのスタックを消費します。割り込みを許可する場合は、さらにデバッガのワーク領域として、84 (54H) バイトのスタックが必要です。また、多重割り込みの場合は、1 段ごとに 84 (54H) バイトのスタックを確保する必要があります。
9	CPU リセット時に内蔵 RAM の内容が破壊されます。通常、実デバイスにおいて、リセット後の内蔵 RAM データは保証しておらず、動作が異なる場合があります。
10	フラッシュ関数を仕様範囲外の方法で使用したり、サポートしていないフラッシュ関数を呼び出した場合、戻り値として "1" が返ります。

## データ・フラッシュ・オプション設定ダイアログ

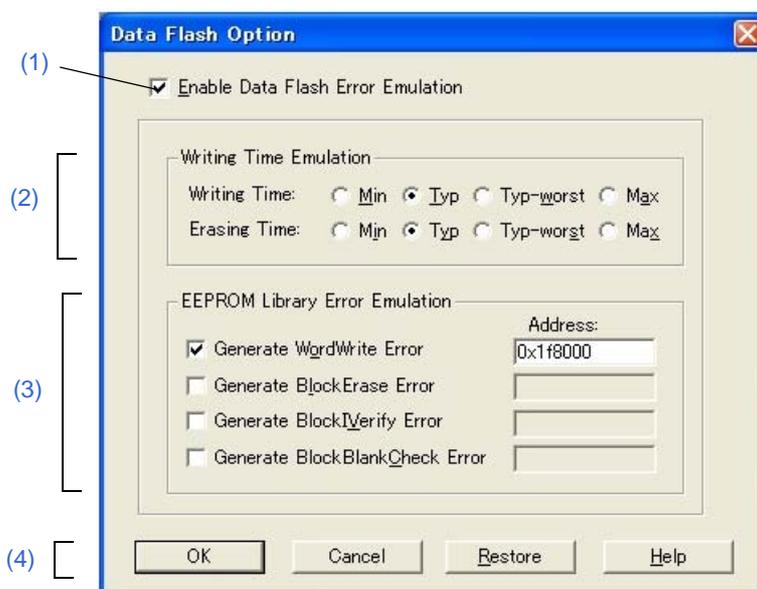
[IECUBE]

データ・フラッシュ・エラー・エミュレーション機能の設定を行います（データ・フラッシュ）。

なお、ユーザ・プログラム実行中は、このダイアログをオープンすることはできません。

**注意** 使用デバイスがデータ・フラッシュ・メモリ搭載品でない場合、このダイアログをオープンすることはできません。

図6-10 データ・フラッシュ・オプション設定ダイアログ



- オープン方法
- 各エリア説明
- 注意事項

### オープン方法

[オプション]メニュー → [データ・フラッシュ・オプション...] を選択

## 各エリア説明

### (1) Enable Data Flash Error Emulation (データ・フラッシュ・エラー・エミュレーション設定エリア)

チェックにより、データ・フラッシュ・エラー・エミュレーション機能が有効になります。

これにより、(2) Writing Time Emulation (書き込み／消去時間設定エリア) と (3) EEPROM Library Error Emulation (EEPROM ライブラリ・エラー・エミュレーション設定エリア) が設定可能になります。

デフォルトはチェックなしです。

### (2) Writing Time Emulation (書き込み／消去時間設定エリア)

データ・フラッシュ・メモリに対する書き込み時間、および消去時間を指定します。

書き込み時間 (Writing Time:), 消去時間 (Erasing Time:) とともに以下から選択します。

Min	リトライなし
Typ	フラッシュ・マクロ・スペック上の想定回数 (デフォルト)
Typ-worst	フラッシュ・マクロ・スペック上の想定最大回数
Max	リトライ回数最大

### (3) EEPROM Library Error Emulation (EEPROM ライブラリ・エラー・エミュレーション設定エリア)

EEPROM ライブラリ関数の動作を設定します。

チェックにより、通常のエミュレーションでは返ることのないエラー値を強制的に返すことができます。

項目をチェックした場合、エラーを発生させるデータ・フラッシュ・メモリ内のアドレスを指定します。

設定できる項目は以下の通りです。全ての項目はデフォルトでチェックなしです。

Generate WordWrite Error	WordWrite 関数でエラーを返す
Generate BlockErase Error	BlockErase 関数でエラーを返す
Generate BlockIVerify Error	BlockIVerify 関数でエラーを返す
Generate BlockBlankCheck Error	BlockBlankCheck 関数でエラーを返す

### (4) 機能ボタン

OK	設定を有効にして、このダイアログをクローズします。
Cancel	このダイアログをクローズします。
Restore	このダイアログをオープンする前の設定状態に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

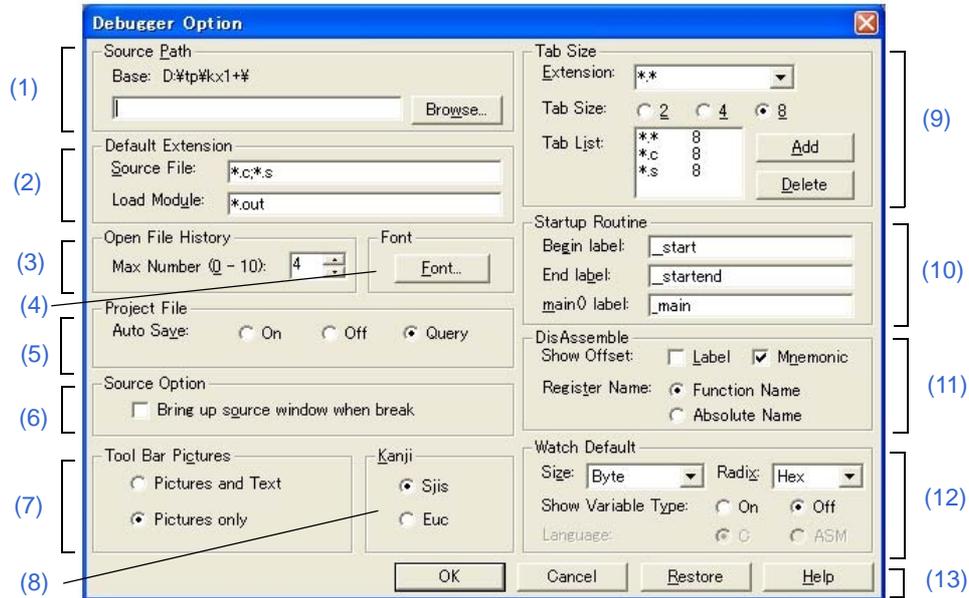
## 注意事項

- (1) [コンフィギュレーション・ダイアログ](#)で "Use Data Flash" をチェックしていない場合、このダイアログをオープンすることはできません。
- (2) "Enable Data Flash Emulation" をチェックした後で、[コンフィギュレーション・ダイアログ](#)で "Use Data Flash" のチェックをはずした場合、このダイアログも無効になります。
- (3) [RRM 設定ダイアログ](#)で、データ・フラッシュ・メモリに対して RRM 領域を設定した場合、ユーザ・プログラム実行中は、書き込んだ値は反映されません（データ・フラッシュ・メモリに対する書き込みは常にライブラリ経由であるため）。同様に、[メモリ・ウィンドウ](#)上でのアクセス・モニタ機能による色付け（Read/Write/Read および Write）も反映されません。
- (4) ユーザ・プログラム実行中に、[DMM ダイアログ](#)からデータ・フラッシュ・メモリへの書き込みはできません。

## デバッガ・オプション設定ダイアログ

ID850QB の各種オプションの表示と設定を行います。

図 6 - 11 デバッガ・オプション設定ダイアログ



- オープン方法
- 各エリア説明

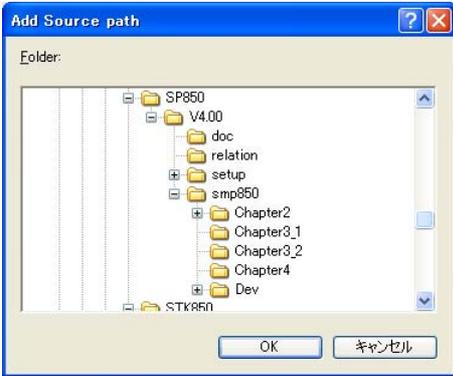
### オープン方法

[オプション]メニュー → [デバッガ・オプション...]を選択

## 各エリア説明

### (1) Source Path (ソース・パス指定エリア)

ソース・ファイルやテキスト・ファイルを検索するフォルダを指定するエリアです。

Base:	<p>相対パスの基準となるフォルダが表示されます。基準フォルダは、次の順序で決定されます。</p> <ol style="list-style-type: none"> <li>1) プロジェクト・ファイルをロードしたフォルダ</li> <li>2) 最後にロード・モジュールやヘキサ・ファイルをロードしたフォルダ</li> <li>3) Windows のカレント・フォルダ</li> </ol>
テキスト・エリア	<p>検索フォルダを指定します。指定は、&lt;Browse&gt; ボタンでの指定、または直接入力により行います。相対パスでの指定も可能です。&lt;Browse...&gt; ボタンをクリックすることにより、<a href="#">[Add Source path] ダイアログ</a>がオープンします。パスの区切りは、";" (セミコロン)、または"," (カンマ)で行います。ソース・パスに日本語を指定できます。</p> <p style="text-align: center;">図 6 - 12 [Add Source path] ダイアログ</p> 

**備考 1** ソース・パス内に";"(セミコロン), "," (カンマ)が存在する場合でも指定できます。

また、存在しないフォルダは追加できません。

**備考 2** このダイアログをオープンした直後は、基準フォルダを選択して開きます。

2 度目以降は前回選択したフォルダを選択してオープンします。

選択したフォルダが、すでにソース・パスに設定されている場合には追加を行いません。

**備考 3** ソース・パス長は、拡張子の区切りを含めて合計で 4095 文字まで設定可能です。

4095 文字を越える場合、4095 文字以内で有効なパスをソース・パスとして設定し、それ以降を切り捨てます。

**(2) Default Extension (デフォルト拡張子指定エリア)**

デフォルトの拡張子を指定するエリアです。

拡張子の区切りは, " " (ブランク), ";" (セミコロン), または "," (カンマ) で指定します。

Source File:	[ファイル]メニュー→[開く...]を選択して、 <a href="#">ソース・ファイル選択ダイアログ</a> を開いたときに表示されるソース・ファイルの拡張子を設定します。デフォルトは, "*.c, *.s" です。
Load Module:	<a href="#">ダウンロード・ダイアログ</a> をオープンした際に表示されるロード・モジュールの拡張子を設定します。デフォルトは, "*.out" です。

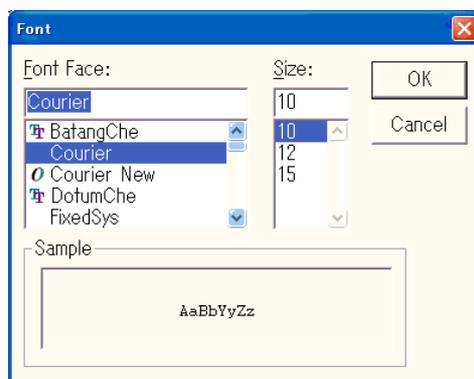
**(3) Open File History (オープン・ファイル・ヒストリ設定エリア)**

[ファイル]メニューの一番下欄に表示されるオープン・ファイルのヒストリ個数を設定するエリアです。0 から 10 までの数が指定可能です (デフォルト: 4)。0 を設定したときは、メニューにヒストリが表示されません。

**(4) Font (表示フォント設定エリア)**

<Font...> ボタンをクリックすることによりオープンする [\[Font\] ダイアログ](#)で、表示フォントやサイズを指定します。変更は、[ソース・テキスト・ウィンドウ](#)、[ウォッチ・ウィンドウ](#)、[クイック・ウォッチ・ダイアログ](#)、[ローカル変数ウィンドウ](#)、[スタック・トレース・ウィンドウ](#)に反映されます。

図 6 - 13 [Font] ダイアログ

**(5) Project File (プロジェクト・ファイル設定エリア)**

プロジェクト・ファイルの自動セーブの設定を行うエリアです ([「5.15.1 デバッグ環境 \(プロジェクト・ファイル\)」](#) 参照)。

Auto Save:	終了時のプロジェクト・ファイルの自動セーブの設定を行います。	
	On	自動セーブを行います。
	Off	自動セーブを行いません。
	Query	終了時に <a href="#">終了確認ダイアログ</a> を表示します (デフォルト)。

**(6) Source Option (ソース・オプション設定エリア)**

ブレーク時のソース・テキスト・ウィンドウの動作を設定します。

チェックにより、ブレーク時にアクティブなソース・テキスト・ウィンドウを最前面に表示します。

アクティブなソース・テキスト・ウィンドウがない場合、またはロード・モジュール・ファイル内にデバッグ情報がない場合にはアクティブな逆アセンブル・ウィンドウが最前面に表示されます。

**(7) Tool Bar Pictures (ツールバー表示ボタン設定エリア)**

ツールバーの表示ボタンを選択するエリアです (「(2) ツールバーの操作」参照)。

Pictures and Text	グラフィックと文字が表示されたボタンを表示します。
Pictures only	グラフィックのみのボタンを表示します (デフォルト)。

**(8) Kanji (漢字コード設定エリア)**

ソース・テキスト・ウィンドウ、トレース・ウィンドウに表示するファイルの漢字コードを選択するエリアです。

Sjis	漢字コードとして SJIS を使用 (デフォルト)
Euc	漢字コードとして EUC を使用

**(9) Tab Size (タブ・サイズ設定エリア)**

タブ・サイズを指定するエリアです。拡張子ごとに、その拡張子のファイルを表示する際のタブ・サイズを指定することができます。

Extension:	拡張子を指定します。 指定は、ドロップダウン・リストからの選択、または直接入力により行います。
Tab Size:	タブ・サイズを選択します。 タブ・コードをスペース何個分 (2 / 4 / 8) で表示するかをラジオ・ボタンにより選択します。
Tab List:	拡張子ごとの現在のタブ・サイズの設定が表示されています。
<Add>	" Extension: ", " Tab Size: " を指定し、<Add> ボタンをクリックすることでタブ・サイズの設定が可能です
<Delete>	" Tab List: " で選択し、このボタンをクリックすることで、設定が削除されます。

**(10) Startup Routine (スタートアップ・シンボル設定エリア)**

スタートアップ・ルーチンのテキスト領域（コード領域）の先頭アドレス、末尾アドレス、および表示開始シンボルをシンボルで指定します。

これにより、**ダウンロード・ダイアログ**でロード・モジュール形式のオブジェクト・ファイルをダウンロードした直後から、ソース・ファイルのオープンが可能になります（このとき、ID850QB は、PC が "Begin label:" と "End label:" の間にあった場合、"main() label:" からの表示を行っています）。

Begin label:	先頭アドレスのシンボルを指定 ( デフォルト : _start)
End label:	末尾アドレスのシンボルを指定 ( デフォルト : _startend)
main() label:	表示開始シンボルを指定 ( デフォルト : _main)

**注意 1** 指定されたシンボルが正しくない場合には、該当するソース・ファイルのアドレス範囲に PC が進むまでソース・ファイルをオープンすることができなくなります。また、スタートアップ・ルーチンをステップ実行でスキップすることはできません。

**注意 2** このエリアは必ず指定してください。空欄にするとダイアログのクローズができません。

**(11) DisAssemble (逆アセンブル表示設定エリア)**

逆アセンブル表示時の設定を行います。

Show Offset:	逆アセンブル表示時のオフセット表示（シンボル + オフセット）の有無を設定します。 オフセット表示を行わない場合、数値に一致するシンボルがあるときのみシンボルを表示し、一致するシンボルがないときは数値をそのまま 16 進数で表示します。	
	Label	Label 欄のオフセット表示する／しないを設定 デフォルトはチェックなしでオフセット表示しません。
	Mnemonic	Mnemonic 欄のオフセット表示する／しないを設定 デフォルトはチェックありでオフセット表示します。
Register Name:	逆アセンブル表示時のモニック中のレジスタ名の表示方法を選択します。	
	Function Name	機能名称、別名で表示（デフォルト）
	Absolute Name	絶対名称で表示

**(12) Watch Default (ウォッチ・シンボル設定エリア)**

ウォッチ・ウィンドウなどでウォッチするシンボルについての指定を行うエリアです。

Size: (デフォルトサイズ指定)	[ 適合 ] (Adaptive) を指定した場合のデータの表示サイズを選択します。	
Radix: (デフォルト進数指定)	[ 自動 ] (Proper) を指定した場合のデータの表示進数を選択します。 <b>注意:</b> ここでの設定は、ウォッチ・ウィンドウ上の配列変数 (またはラベル) の添字に対しても反映されます (デフォルトでは 16 進数になります)。なお、添字に関しては、設定変更以降にウォッチ・ウィンドウに登録されたウォッチ・データに対して反映されます。	
	Hex	16 進数表示 (デフォルト)
	Dec	10 進数表示
	Oct	8 進数表示
	Bin	2 進数表示
	String	文字列表示
Show Variable Type: (変数型の表示/非表示)	表示している変数の型の表示/非表示を選択します。	
	On	変数の型を表示する
	Off	変数の型を表示しない (デフォルト)
Language: (基数表示の指定)	表示している変数値の基数表示を選択します。	
	C	C 言語風の基数を表示 (デフォルト)
	ASM	選択不可

**(13) 機能ボタン**

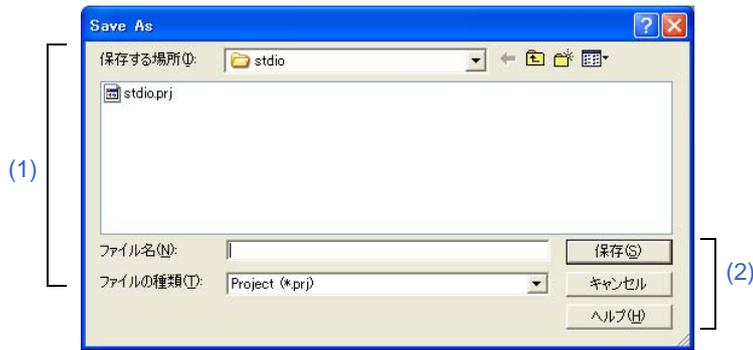
OK	設定した各種設定を有効にして、このダイアログをクローズします。
Cancel	変更を取り消して、このダイアログをクローズします。
Restore	このダイアログをオープンする前の設定状態に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

## プロジェクト・ファイル・セーブ・ダイアログ

現在のデバッグ環境をプロジェクト・ファイルに保存するダイアログです（「5.15.1 デバッグ環境（プロジェクト・ファイル）」参照）。

このダイアログではプロジェクト・ファイルを新規保存、またはファイル名を変更して保存します。

図 6 - 14 プロジェクト・ファイル・セーブ・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

[ファイル]メニュー→[プロジェクト]→[名前を付けて保存...]を選択

(以前にプロジェクト・ファイルをロード、またはセーブして同じファイル名で保存する場合は[ファイル]メニュー→[プロジェクト]→[上書き保存]を選択)

### 各エリア説明

#### (1) 保存ファイル設定エリア

保存する場所:	ファイル名を指定するエリアです。
ファイル名:	一覧から選択するか、直接入力により指定します。 指定可能文字数は拡張子を含めて 257 文字までです。
ファイルの種類:	保存するプロジェクト・ファイルの拡張子 (*.prj) を指定するエリアです。 拡張子を省略した場合には、デフォルトで .prj が付加されます。

#### (2) 機能ボタン

保存	選択したファイルにデバッグ環境をセーブします。 セーブ後、ダイアログをクローズします。
キャンセル	ファイルをセーブせず、このダイアログをクローズします。
ヘルプ	このダイアログのオンライン・ヘルプを表示します。

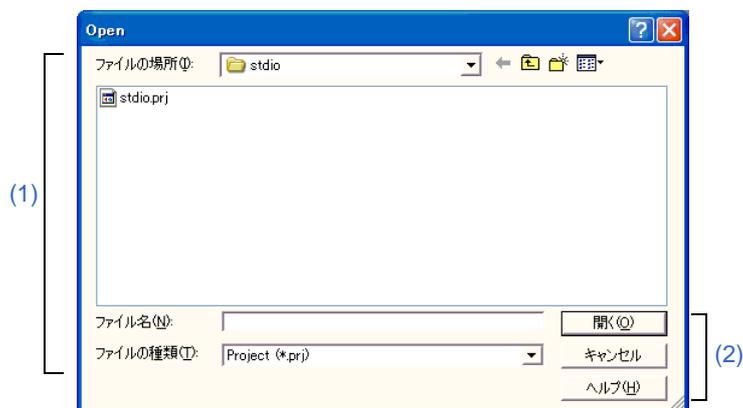
## プロジェクト・ファイル・ロード・ダイアログ

デバッグ環境を、指定したプロジェクト・ファイルに保存されたデバッグ環境を復元するダイアログです（「5.15.1 デバッグ環境（プロジェクト・ファイル）」参照）。

プロジェクト・ファイルのロード後、アクティブなソース・テキスト・ウィンドウがある場合、一番上に表示されます。

**注意** ID850QB 起動後に、起動時のターゲット・デバイスと異なる設定のプロジェクト・ファイルをロードした場合には、起動時に指定したターゲット・デバイスとなります。

図 6-15 プロジェクト・ファイル・ロード・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

 ボタンをクリック、または[ファイル]メニュー→[プロジェクト]→[開く...]を選択

### 各エリア説明

#### (1) ロード・ファイル設定エリア

ファイルの場所:	ロードするファイル名を指定します。
ファイル名:	一覧からの選択、または直接入力により指定します。 指定できる文字数は、拡張子を含めて 257 文字までです。
ファイルの種類:	ロードするファイルの拡張子 (*.prj) を指定します。

#### (2) 機能ボタン

開く	選択したファイルをロードします。ロード後、ダイアログをクローズします。
キャンセル	何もせずにこのダイアログをクローズします。
ヘルプ	このダイアログのオンライン・ヘルプを表示します。

## ダウンロード・ダイアログ

ダウンロードするファイル名、およびファイル形式を選択し、メモリ内容等をインサーキット・エミュレータ、およびターゲット・システムへダウンロードします（「5.2 ダウンロード／アップロード機能」参照）。

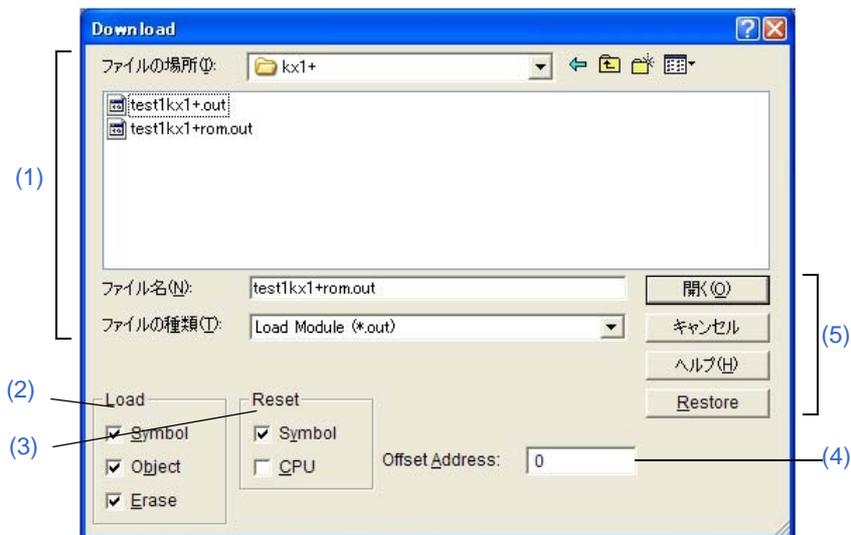
ロード・モジュール・ファイルをロードした場合には、該当するソース・ファイルを検索し、自動的にソース・テキスト・ウィンドウをオープンします。

**注意** ロード・モジュール・ファイル以外をロードした場合には、ソース・デバッグができません。

**参考 1** 内蔵フラッシュ・メモリへのダウンロードが可能です（「5.7.4 フラッシュ・メモリへの書き込み機能 [MINICUBE] [MINICUBE2]」参照）。[MINICUBE] [MINICUBE2]

**参考 2** ダウンロード中は、ダウンロードの進捗を表示するダイアログがオープンし、常にキャンセルを受け付けます。

図 6 - 16 ダウンロード・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法



ボタンをクリック、または[ファイル]メニュー→[ダウンロード...]を選択

## 各エリア説明

### (1) ロード・ファイル設定エリア

ファイルの場所：	ロードするファイル名を指定します。
ファイル名：	一覧からの選択、または直接入力により指定します。 指定できる文字数は、拡張子を含めて 257 文字までです。
ファイルの種類：	ロードするファイルの拡張子を指定します（「表 5-3 ダウンロードが可能なファイルの種類」参照）。 これらはデフォルトの拡張子であり、これら以外の拡張子も使用することができます。また、デバッグ・オプション設定ダイアログで表示ロード・モジュールのデフォルト拡張子を指定することもできます。

**参考** " "（ダブルクォーテーション）で区切ることにより複数ファイルの選択が可能です。

一覧から Shift キーや Ctrl キーを押しながら選択します。最大 20 個までダウンロード可能です。

### (2) Load（ロード条件設定エリア）

ロード条件を指定するエリアです。チェックした項目のロード条件を有効にします。

この指定は、ロード・モジュール形式のファイルをダウンロードする場合のみ有効です。

Symbol	シンボル情報を読み込むかどうかを指定します。 <b>注</b>
Object	オブジェクト情報を読み込むかどうかを指定します（HEX ファイルのロード時は、非チェック状態でもオブジェクト情報を読み込みます）。
Erase	選択不可 [IECUBE] ダウンロード前に内蔵フラッシュ・メモリを全消去するかどうかを指定します。 [MINICUBE]

**注** 複数のロード・モジュール・ファイルから構成されるプログラムをデバッグする場合などに、シンボル情報が不要なモジュールがあれば、" Symbol " のチェックを外します。これにより、メモリの使用量を軽減することができます。

### (3) Reset（リセット条件設定エリア）

リセット条件を指定するエリアです。チェックした項目のリセット条件を有効にします。

この指定は、ロード・モジュール形式のファイルをダウンロードする場合のみ有効です。

Symbol	シンボル情報をリセットするかどうかを指定します。 <b>注</b>
CPU	CPU リセットを行うかどうかを指定します。 （デフォルト：リセットしない）

**注** 複数のロード・モジュール・ファイルから構成されるプログラムをデバッグする場合には、シンボル情報をリセットせずに個々のロード・モジュール・ファイルをロードしてください。  
また、複数のロード・モジュール・ファイルをダウンロードする場合には、配置アドレスが重ならないようにしてください。

**(4) Offset Address: (オフセット・アドレス指定エリア)**

ロードするときのオフセット・アドレスを指定するエリアです。

なお、バイナリ・データの場合は、開始アドレスを指定します。

指定はシンボルや式でも可能です（「表 5-6 シンボルでの指定方法」参照）。

数値入力時のデフォルト進数は 16 進です。

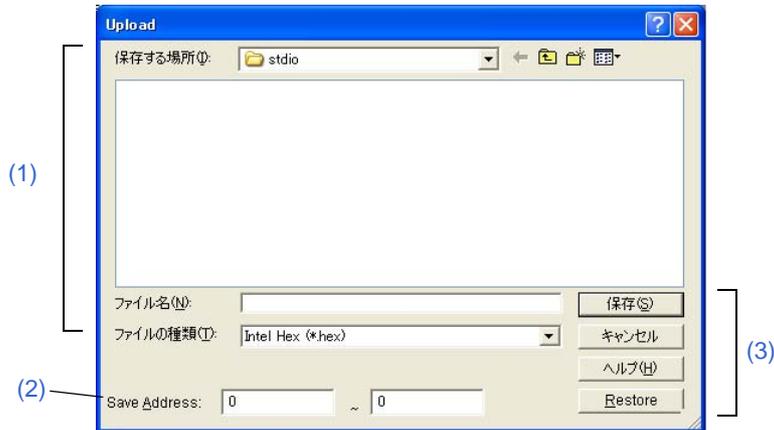
**(5) 機能ボタン**

開く	選択したファイルをロードします。ロード後、ダイアログをクローズします。
キャンセル	何もせずにこのダイアログをクローズします。
ヘルプ	このダイアログのオンライン・ヘルプを表示します。
Restore	入力したデータを元に戻します。

## アップロード・ダイアログ

保存するファイル名、およびファイル形式を設定しメモリ内容等をファイルへ保存します（「5.2 ダウンロード／アップロード機能」参照）。

図 6 - 17 アップロード・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

[ファイル]メニュー → [アップロード ...] を選択

### 各エリア説明

#### (1) アップロード・ファイル設定エリア

保存する場所：	ファイル名を指定するエリアです。一覧から選択するか、直接入力により指定します。
ファイル名：	指定可能文字数は拡張子を含めて 257 文字までです。
ファイルの種類：	保存するファイルの拡張子を指定するエリアです。ファイル名が確定した際の拡張子により、保存されるデータの形式（フォーマット）が決まります（「表 5 - 4 アップロードが可能なファイルの種類」参照）。

**参考** リスト以外の拡張子の使用も可能です。

**(2) Save Address: (アドレス範囲指定エリア)**

保存するアドレス範囲を指定するエリアです。

(カバレッジ・データ (\*.cvb) を選択した場合には全範囲が保存されるため、このエリアは設定不可になります。)

アドレス指定はシンボルや式での指定も可能です (「表 5-6 シンボルでの指定方法」参照)。

数値入力時のデフォルト進数は 16 進です。

**(3) 機能ボタン**

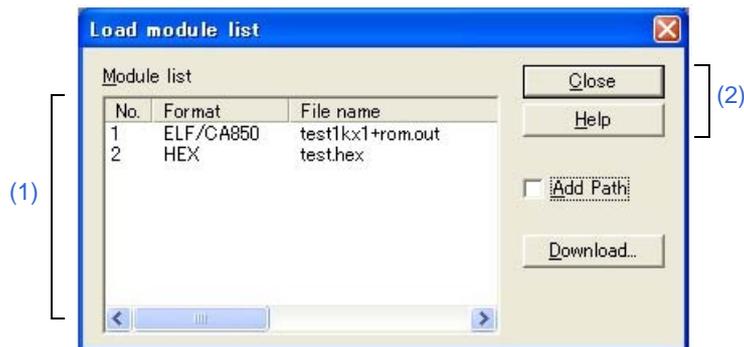
保存	設定に従いファイルに保存します。
キャンセル	何もせずにこのダイアログをクローズします。
ヘルプ	このダイアログのオンライン・ヘルプを表示します。
Restore	このダイアログをオープンする前の状態に戻します。

## ロード・モジュール一覧ダイアログ

ダウンロード済みのすべてのファイル名、ファイル・パス、およびファイル形式を一覧表示します（「5.2 ダウンロード/アップロード機能」参照）。

一覧表示されるファイル（カバレッジ・データ・ファイルは除く）は、プロジェクト・ファイルに保存され次回プロジェクト・ファイルを開いた際にダウンロードされます。また、<Download...> ボタンにより、[ダウンロード・ダイアログ](#)を開き、ファイルのダウンロードを行うことができます。

図 6 - 18 ロード・モジュール一覧ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

[ファイル]メニュー→[ロードモジュール...]を選択

## 各エリア説明

### (1) ロード・モジュール・ファイル表示エリア

Module list	ダウンロード済みのファイル名を表示するエリアです。		
	No.	読み込みを行った順番	
	Format	ファイル形式を表示	
		BIN	バイナリ・ファイル
		HEX	ヘキサ・ファイル
		COV	カバレッジ・データ・ファイル
		ELF/CA850	ロード・モジュール・ファイル (ELF/CA850)
		ELF/DWARF2	ロード・モジュール・ファイル (ELF/GHS 拡張 DWARF2)
		ELF	ロード・モジュール・ファイル (ELF シンボル情報なし)
		unknown	不明
File name	" Add Path " がチェックされているときはフル・パスで、チェックされていない場合にはファイル名のみを表示		
Add Path	ファイル名をパス付きで表示させる場合、チェックします。		
<Download...>	<a href="#">ダウンロード・ダイアログ</a> をオープンします。 新たにファイルをダウンロードすることができます。新たにダウンロードしたファイル名は、 <a href="#">ダウンロード・ダイアログ</a> をクローズした際、追加されます。		

**注意** リセット確認ダイアログでシンボル情報をリセットした場合、または[ダウンロード・ダイアログ](#)でシンボル情報をリセットしてダウンロードした場合には、それ以前にダウンロードされていたファイル名がクリアされます。

### (2) 機能ボタン

Close	このダイアログをクローズします。
Help	このダイアログのオンライン・ヘルプを表示します。

## ソース・テキスト・ウィンドウ

ソース・ファイル、およびテキスト・ファイルの表示を行います（「5.3 ソース表示／逆アセンブル表示機能」参照）。このウィンドウでは、ブレーク・ポイントの設定、カバレッジ実行箇所の表示、混合表示モード（ソース・テキスト・ウィンドウ）のほか、コンテキスト・メニュー、機能ボタン等によるさまざまな操作が可能です。

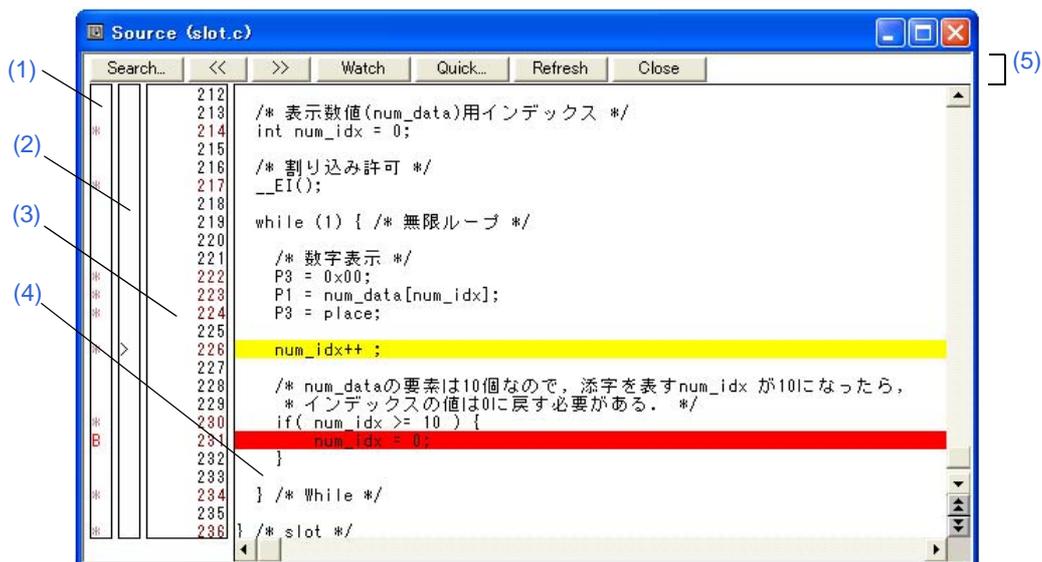
また、このウィンドウには、アクティブ状態とスタティック状態の2つの状態があり、アクティブ状態時には、トレース結果とウィンドウの連結機能 [IECUBE] を持ちます。また、ドラッグ & ドロップ機能により、ウィンドウ内の選択した項目を他ウィンドウで使用することができます（「5.16 ウィンドウ共通機能」参照）。

**注意 1** インクルード・ファイル内にプログラム・コードを記述し、複数のファイルでインクルードした場合には、行番号とアドレスとの対応関係が1対1になりません。このようなインクルード・ファイル内では、行番号とアドレスとの対応関係を使用する機能は正しく動作しません。

**注意 2** ダウンロード後に関数 "main" のあるソース・ファイル名がソース・パス内で見つからない場合、またはステップ実行中にソース・ファイルが見つからない場合、ID850QB は、ソース・ファイルを選択するダイアログをオープンし、探しているソース・ファイル名を表示し問い合わせます。このとき、<キャンセル> ボタンを押した場合には、そのファイル名を記憶し、以降 ID850QB を終了するまで記憶したソース・ファイルの問い合わせは行いません。

**参考** C 言語ソース・ファイル、およびアセンブリ言語ソース・ファイルとして表示できる行数は、65535 行までです。65535 行を越えるソース・ファイルは、分割してください。

図 6 - 19 ソース・テキスト・ウィンドウ



- オープン方法
- 各エリア説明
- [表示]メニュー（ソース・テキスト・ウィンドウ専用部）
- コンテキスト・メニュー

## オープン方法



ボタンをクリック，または[ブラウザ]メニュー→[ソース]を選択

(ロード・モジュール・ファイルをダウンロード後，該当ソース・ファイルがある場合には自動的にオープン)

## 各エリア説明

### (1) ポイント・マーク・エリア

イベント設定状態，プログラム・コード(\*)の表示，およびブレーク・ポイントの設定を行うエリアです。

プログラム・コード上をマウスでクリックすることにより，ブレーク・ポイントの設定／削除が可能です('\*'が表示されていない行では '\*' が表示されている上下どちらかの行に対して行われます)。

プログラム・コードは，ロード・モジュール・ファイルがダウンロードされシンボル情報が読み込まれている場合にのみ表示されます。

対応する行にイベントが設定されている場合，次表に示すマークが表示されます。'B'マークは，ブレーク・ポイントの種類や状態により表示色が異なります(このエリアでブレーク・ポイントを設定した場合，設定と同時に有効になります)。

表 6 - 11 イベント設定状態

マーク	意味
B (青)	ソフトウェア・ブレーク設定
B (赤)	有効なハードウェア・ブレーク (実行後) 設定
B (緑)	有効なハードウェア・ブレーク (実行前) 設定 注意：実行後のハードウェア・ブレークより優先して設定される
B (黒)	無効なハードウェア・ブレーク イベント・マネージャやブレーク・ダイアログで有効にすることができます。
E	イベント条件設定
L	イベント・リンク条件設定
T	トレース・イベント設定 [IECUBE]
Ti	タイマ・イベント設定 [IECUBE]
A	複数のイベント設定

**参考** イベントのアドレス条件が範囲アドレス指定の場合，範囲の下位アドレスが対象となります。  
また，アドレス条件のマスク指定は反映されません。

**(2) カレント PC マーク・エリア**

現在のカレント PC 値 (PC レジスタ値) を示すマーク '>' を表示するエリアです。

このマークをクリックすることにより、PC レジスタ値がポップアップ・ウィンドウで表示されます。

また、このエリアをダブルクリックすることにより指定した行までプログラムを実行します (「[\[カーソル位置まで実行\]](#)」参照)。

**(3) 行番号／アドレス表示エリア**

ソース・ファイルやテキスト・ファイルの行番号を表示するエリアです。**赤色**は対応するプログラム・コードが存在する行番号を、**黒色**は存在しない行番号を示します。なお、**混合表示モード** (ソース・テキスト・ウィンドウ) 時には、逆アセンブル表示のアドレスが灰色で表示されます。

また、コード・カバレッジ測定情報を元に、実行アドレスが色付きで表示されます (「[5.11.3 カバレッジ実行箇所](#)」参照)。

**(4) ソース・テキスト表示エリア**

ソース・ファイルやテキスト・ファイルを表示するエリアです。

次の行は強調表示されます。

カレント PC 行 ( <b>黄色</b> )	カレント PC 値に対応するソース行、または逆アセンブル行を示します (" > " 表示行)。 ブレーク時に、PC 位置に行番号情報がない場合には、 <b>逆アセンブル・ウィンドウ</b> に自動的にジャンプします。 <b>混合表示モード</b> (ソース・テキスト・ウィンドウ) 時は、逆アセンブル表示行のみ表示色の変更されます。
ブレーク・ポイント設定行 ( <b>赤色</b> )	有効なブレーク・ポイント設定行を示します。 <b>混合表示モード</b> (ソース・テキスト・ウィンドウ) 時は、逆アセンブル表示行のみ表示色の変更されます。

また、このエリアでは、カーソルの置かれた行 (プログラム・コードの先頭アドレス) / アドレスを対象として、次の機能を備えています。

- [\[カーソル位置から実行\]](#), [\[カーソル位置まで実行\]](#) 機能 (「[表 5-9 実行の種類](#)」参照)
- [ドラッグ & ドロップ機能](#)
- [コンテキスト・メニュー](#)

**注意** これら機能においては、ソース行にプログラム・コードが存在しない場合は、プログラム・コードが存在する上下どちらかの行の先頭アドレスが対象となります。

なお、次の場合にはこれらの機能を実行することができません (対応するメニューは選択不可)。

- ソース・ファイル以外を表示している場合
- ユーザ・プログラムが実行中の場合

## (5) 機能ボタン

Search...	ソース・サーチ・ダイアログをオープンし、文字列を検索します。 テキストが選択されていた場合はそのテキストを検索対象として、選択されていない場合には空欄の状態オープンされます。 検索方法は、ソース・サーチ・ダイアログ上で指定します。検索結果は、ソース・テキスト・ウィンドウ上で反転表示されます。 [表示]メニュー→[検索...]選択時と同様の動作です。
<<	ソース・サーチ・ダイアログで設定した検索条件に一致するテキスト、カーソル位置のアドレスから前方向(表示の上)に検索します。 検索中は<Stop>ボタンに変化します。
>>	ソース・サーチ・ダイアログで設定した検索条件に一致する内容を、カーソル位置のアドレスから後ろ方向(表示の下)に検索します。 検索中は<Stop>ボタンに変化します。
Stop (検索中)	検索を中断します。
Watch	選択した変数等をウォッチ・ウィンドウに追加します。ウォッチ・ウィンドウがオープンしていない場合はオープンします。 テキストが選択されていない場合、ウィンドウのオープンのみを行います。 [表示]メニュー→[ウォッチ追加]選択時と同様の動作です。
Quick...	選択した変数等の内容をクイック・ウォッチ・ダイアログで一時的に表示します。テキストが選択されていない場合、ダイアログのオープンのみを行います。 [表示]メニュー→[クイック・ウォッチ]選択時と同様の動作です。
Refresh	ウィンドウの内容を最新のデータに更新します。 ファイルの再読み込みは行いません。
Close	このウィンドウをクローズします。

## [表示]メニュー (ソース・テキスト・ウィンドウ専用部)

ソース・テキスト・ウィンドウのアクティブ時には、共通の[表示]メニューに以下の項目が追加されます。

ブレイク・イベント作成	選択している変数に対してブレイク・イベントを設定します。
この変数へのアクセスでブレイク	Read/Write アクセスのブレイク・イベントを設定します。
この変数への書き込みでブレイク	Write アクセスのブレイク・イベントを設定します。
この変数からの読み出しでブレイク	Read アクセスのブレイク・イベントを設定します。
解除	設定されているブレイク・イベントを削除します。
イベント内容の表示	カーソル位置の行、または選択した変数のイベント情報を表示します。 イベントが設定されている場合、イベント・ダイアログがオープンします。
混合表示	混合表示モード(ソース・テキスト・ウィンドウ)に切り替えます。

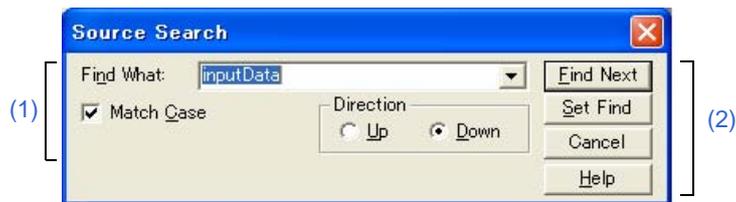
## コンテキスト・メニュー

移動 ...	表示位置を移動します。 ソース指定ダイアログをオープンします。
混合表示	混合表示モード (ソース・テキスト・ウィンドウ) に切り替えます。
ウォッチ登録 ...	指定したデータをウォッチ・ウィンドウに追加します。 ウォッチ登録ダイアログをオープンします。
シンボル変換 ...	指定した変数や関数のアドレス、またはシンボルの値を表示します。 シンボル変換ダイアログをオープンします。
この変数へのアクセスでブレーク	Read/Write アクセスのブレーク・イベントを設定します。
この変数への書き込みでブレーク	Write アクセスのブレーク・イベントを設定します。
この変数からの読み出しでブレーク	Read アクセスのブレーク・イベントを設定します。
解除	設定されているブレーク・イベントを削除します。
イベント内容の表示	カーソル位置の行、または選択した変数のイベント情報を表示します。 イベントが設定されている場合、イベント・ダイアログがオープンします。
カーソル位置まで実行	カレント PC からカーソル位置までプログラムを実行します (「表 5-7 主なブレークの種類」参照)
PC 値変更	カーソル位置のアドレスを PC に設定します。
ブレークポイント	カーソル位置のハードウェア・ブレーク・ポイントを設定/削除します。 備考: 実行前ブレーク (B) が優先して設定されます。
ソフトウェア・ブレークポイント	カーソル位置のブレーク・ポイントを設定/削除します。
アセンブル	カーソル位置のデータ値をジャンプ先アドレスとして、そのアドレスから逆アセンブル表示します (「5.16.2 ジャンプ機能」参照)。 逆アセンブル・ウィンドウをオープンします。 アクティブ状態の逆アセンブル・ウィンドウが表示されている場合は、そのウィンドウを最前面 (操作対象) に表示します。
メモリ	カーソル位置のデータ値をジャンプ先アドレスとして、そのアドレスからメモリ内容を表示します (「5.16.2 ジャンプ機能」参照)。 メモリ・ウィンドウをオープンします。 アクティブ状態のメモリ・ウィンドウが表示されている場合は、そのウィンドウを最前面 (操作対象) に表示します。

## ソース・サーチ・ダイアログ

ソース・テキスト・ウィンドウ内のファイル内容を検索します（「5.3.1 ソース表示」参照）。各項目を設定し、<Find Next> ボタンをクリックすることにより検索を開始します。また、<Set Find> ボタンをクリックすることで、ソース・テキスト・ウィンドウ上の方向ボタン（<<, >>）による検索が可能になります。

図 6 - 20 ソース・サーチ・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

ソース・テキスト・ウィンドウをカレント・ウィンドウにし、[表示]メニュー→[検索...]を選択、または同ウィンドウ上の<Search...> ボタンをクリック

### 各エリア説明

#### (1) サーチ条件指定エリア

Find What:	サーチ・データを指定するエリアです（最大 256 文字）。 デフォルトでは呼び出したウィンドウ内で選択した文字列が表示されますが、必要に応じて変更可能です。 入力履歴は最大 16 個まで残すことができます。	
Match Case	指定したサーチ・データの太文字／小文字を区別して検索する場合、チェックします。	
Direction	検索方向を選択するエリアです。	
	Up	前方検索。現在カーソルのある位置より前（表示の上）方向に検索します。
	Down	後方検索。現在カーソルのある位置より後ろ（表示の下）方向に検索します（デフォルト）。

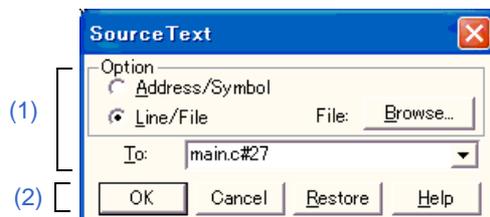
## (2) 機能ボタン

Find Next	設定に従い検索します。検索の結果、一致する文字列は反転表示されます。連続して検索する場合には、再度このボタンをクリックします。
Set Find	指定された条件を検索条件として設定し、ダイアログをクローズします。
Stop (検索中)	検索を中断します。
Cancel	このダイアログをクローズします。 (検索中は、このボタンが <Stop> ボタンに変化しています。)
Help	このダイアログのオンライン・ヘルプを表示します。

## ソース指定ダイアログ

ソース・テキスト・ウィンドウに表示するファイルの指定、および表示開始位置を指定します（「5.3.1 ソース表示」参照）。

図 6 - 21 ソース指定ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

ソース・テキスト・ウィンドウをカレント・ウィンドウにし、[表示]メニュー→[移動...]を選択

### 各エリア説明

#### (1) ソース・ファイル指定エリア

Option	移動先を指定する際の入力モードを選択するエリアです。	
	Address/Symbol	移動先をアドレス（またはシンボル）で指定する場合、選択します。
	Line/File	移動先を行番号（またはファイル名）で指定する場合、選択します。ファイル名の検索は <Browse...> ボタンで行います。
To:	表示ファイル名や表示アドレスを指定するエリアです。入力履歴は最大 16 個まで残すことができます。	
	" Address/Symbol " を選択した場合	表示を開始するアドレスを指定します。数値入力時のデフォルト進数は 16 進です。シンボルや式での指定も可能です（「表 5 - 6 シンボルでの指定方法」参照）。<OK> ボタンをクリックすることにより、指定されたアドレス値に対応するソース行が見えるようにソース・テキストが表示されます。
	" Line/File " を選択した場合	表示を開始する行番号（またはファイル名）を指定します。行番号の指定は [[パス名] ファイル名]# 行番号で行います。数値入力時のデフォルト進数は 10 進数です。ファイル名は省略可能です。ファイル名の指定はファイル名のみ、または絶対パスおよび相対パスで行います。ファイル名のみ、あるいは相対パスを指定した場合、デバッガ・オプション設定ダイアログで指定したソース・パス内のファイルが検索されます。<OK> ボタンをクリックすることにより、指定した行番号を先頭行に指定したファイルが表示されます。ファイル名の省略時には現在表示しているファイルが指定行から表示されます。また、行番号の省略時にはファイルの先頭行から表示されます。

## (2) 機能ボタン

OK	指定した位置からソース・テキスト表示を行います。
Cancel	このダイアログをクローズします。
Restore	このダイアログをオープンする前の状態に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

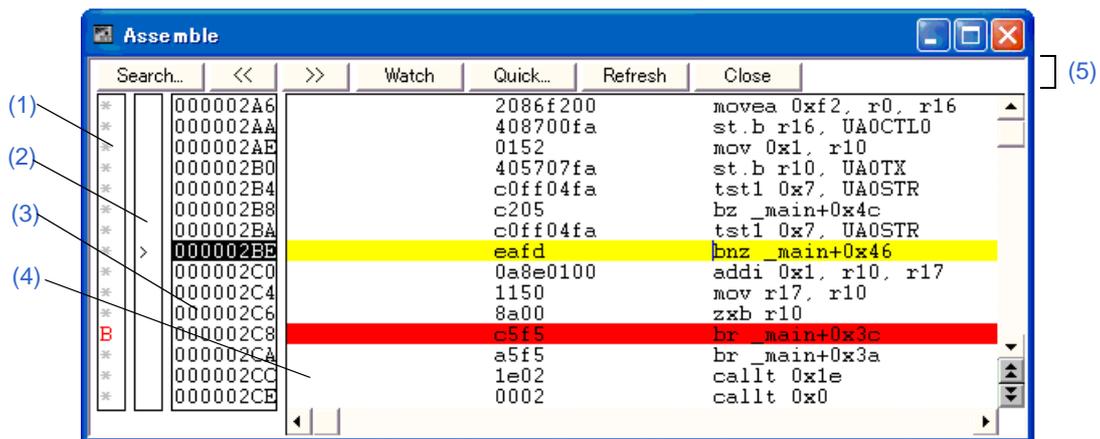
## 逆アセンブル・ウィンドウ

プログラムの逆アセンブル表示を行います。また、[オンライン・アセンブル](#)を行います（「[5.3 ソース表示／逆アセンブル表示機能](#)」参照）。オンライン・アセンブルの結果はメモリ・ウィンドウにも反映されます。

このウィンドウでは、[ブレーク・ポイントの設定](#)、[カバレッジ実行箇所の表示](#)のほか、[コンテキスト・メニュー](#)、[機能ボタン](#)等によるさまざまな操作が可能です。

このウィンドウには、[アクティブ状態とスタティック状態](#)の2つの状態があり、アクティブ状態時には、[トレース結果とウィンドウの連結機能 \[IECUBE\]](#)を持ちます。また、[ドラッグ & ドロップ機能](#)により、ウィンドウ内の選択した項目を他ウィンドウで使用することができます（「[5.16 ウィンドウ共通機能](#)」参照）。

図6-22 逆アセンブル・ウィンドウ



- オープン方法
- 各エリア説明
- [表示]メニュー（逆アセンブル・ウィンドウ専用部分）
- コンテキスト・メニュー
- 関連操作

## オープン方法



ボタンをクリック、または [ブラウズ] メニュー → [アセンブル] を選択

## 各エリア説明

### (1) ポイント・マーク・エリア

イベント設定状態の表示、およびブレーク・ポイントの設定を行うエリアです。

### (2) カレント PC マーク・エリア

現在のカレント PC 値 (PC レジスタ値) を示すマーク '>' を表示するエリアです。

このエリアをダブルクリックすることにより指定した行までユーザ・プログラムを実行します。(「[カーソル位置まで実行]」参照)。

### (3) アドレス指定エリア

逆アセンブル開始アドレスを表示するエリアです。

また、コード・カバレッジ測定情報を元に、実行アドレスが色付きで表示されます(「5.11.3 カバレッジ実行箇所を表示」参照)。

**注意** 最終アドレス (0xFFFFFFFF) は表示されません。

### (4) 逆アセンブル表示エリア

アドレスのラベル、コード・データ、および逆アセンブルされたニモニックを表示するエリアです。

ニモニック欄はオンライン・アセンブル可能です。

次の行は強調表示されます。

カレント PC 行 (黄色)	カレント PC 値に対応する行を示します (">" 表示行)。ブレーク時に、PC 位置に行番号情報がない場合には、逆アセンブル・ウィンドウに自動的にジャンプします。
ブレーク・ポイント設定行 (赤色)	有効なブレーク・ポイント設定行を示します。

また、このエリアでは次の機能を備えています。

- [カーソル位置から実行], [カーソル位置まで実行] 機能 (「表 5-9 実行の種類」参照)
- ドラッグ & ドロップ機能
- コンテキスト・メニュー

## (5) 機能ボタン

Search...	逆アセンブル・サーチ・ダイアログをオープンし、ニモニック文字列を検索します。検索方法は、逆アセンブル・サーチ・ダイアログで指定します。検索結果は、逆アセンブル・ウィンドウ上で反転表示されます。 [表示]メニュー→[検索...]選択時と同様の動作です。
<<	逆アセンブル・サーチ・ダイアログで設定した検索条件に一致する内容を、カーソル位置のアドレスから前方向（表示の上）に検索します。 検索中は<Stop>ボタンに変化します。
>>	逆アセンブル・サーチ・ダイアログで設定した検索条件に一致する内容を、カーソル位置のアドレスから後ろ方向（表示の下）に検索します。 検索中は<Stop>ボタンに変化します。
Stop（検索中）	検索を中断します。
Watch	(4) 逆アセンブル表示エリアで選択したシンボル等をウォッチ・ウィンドウに追加します。ウォッチ・ウィンドウがオープンしていない場合はオープンします。逆アセンブル表示エリアでテキストが選択されていない場合、ウォッチ・ウィンドウのオープンのみ行います。 [表示]メニュー→[ウォッチ追加]選択時と同様の動作です。
Quick...	(4) 逆アセンブル表示エリアで選択したシンボル等の内容をクイック・ウォッチ・ダイアログで一時的に表示します。逆アセンブル・テキスト表示エリアでテキストが選択されていない場合、クイック・ウォッチ・ダイアログのオープンのみ行います。 [表示]メニュー→[クイック・ウォッチ]選択時と同様の動作です。
Refresh	ウィンドウの内容を最新のデータに更新します。
Close	このウィンドウをクローズします。

## [表示]メニュー（逆アセンブル・ウィンドウ専用部分）

逆アセンブル・ウィンドウのアクティブ時には、共通の[表示]メニューに以下の項目が追加されます。

イベント内容の表示	カーソル位置のアドレスのイベント情報を表示します。 イベントが設定されている場合、イベント・ダイアログがオープンします。
-----------	---

## コンテキスト・メニュー

各メニュー項目は、クリックしたマウス・ポインタの位置ではなく、選択されている行、または項目に対して機能します（同名のメイン・メニューを選択した際と同じ動作）。

移動 ...	表示位置を移動します。 アドレス指定ダイアログをオープンします。
ウォッチ登録 ...	指定したデータをウォッチ・ウィンドウに追加します。 ウォッチ登録ダイアログをオープンします。
シンボル変換 ...	指定した変数や関数のアドレス、またはシンボルの値を表示します。 シンボル変換ダイアログをオープンします。
カーソル位置まで実行	カレント PC からカーソル位置までプログラムを実行します（「表 5-7 主なブレークの種類」参照）。
PC 値変更	カーソル位置のアドレスを PC に設定します。

ブレイクポイント	カーソル位置のハードウェア・ブレイク・ポイントを設定／削除します。 <b>備考</b> ：実行前ブレイク (B) が優先して設定されます。(Nx85ET)
ソフトウェア・ブレイクポイント	カーソル位置のブレイク・ポイントを設定／削除します。
ソース	カーソル位置のデータ値をジャンプ先アドレスとして、該当するソース・テキストおよびソース行を表示します (「5.16.2 ジャンプ機能」参照)。 ただし、ジャンプ先アドレスに行情報がない場合には、ジャンプすることはできません。 ソース・テキスト・ウィンドウをオープンします。 アクティブ状態のソース・テキスト・ウィンドウが表示されている場合は、そのウィンドウを最前面 (操作対象) に表示します。
メモリ	カーソル位置のデータ値をジャンプ先アドレスとして、そのアドレスからメモリ内容を表示します (「5.16.2 ジャンプ機能」参照)。 メモリ・ウィンドウをオープンします。 アクティブ状態のメモリ・ウィンドウが表示されている場合は、そのウィンドウを最前面 (操作対象) に表示します。

## 関連操作

### (1) オンライン・アセンブル

オンライン・アセンブルは、カーソルをニモニク欄へ移動し直接入力することにより行います (編集時の上書きモードと挿入モードは Insert キーで切り替え可能)。

ニモニク欄で変更を行いカーソルを別の行に移動させる際、変更内容がチェックされます。変更内容が不正である場合は、変更を行った行のコード・データが "\*" になります。

ニモニク欄で変更した内容は Enter キーでメモリに書き込まれます。Enter キーを押した場合も変更内容がチェックされます。1 行でも不正な行があると、変更内容はメモリに書き込まれません。変更内容の破棄は ESC キーで行います。

変更内容に不正がなかった場合は、Enter キーを押すことにより変更内容がメモリに書き込まれ、次の行へカーソルが移動します。

**注意** 変更した結果、変更前の命令のバイト数に対して、変更後の命令のバイト数が少ない場合は、残されたバイトが nop 命令に置き替わります。また、変更した結果、変更前の命令のバイト数に対して、変更後の命令のバイト数が多い場合は、次の命令を上書きします。この場合も残されたバイトは nop 命令に置き替わります。ソース行をまたがる場合でも同様に上書きされます。

## 逆アセンブル・サーチ・ダイアログ

逆アセンブル・ウィンドウの内容を検索します（「5.3.2 逆アセンブル表示」参照）。

入力文字列と逆アセンブラ文字列に含まれる連続した空白文字は、1つの空白文字として比較します。

各項目を設定し、<Find Next> ボタンをクリックすることにより検索を開始します。また、<Set Find> ボタンをクリックすることで、逆アセンブル・ウィンドウ上の方向ボタン (<<, >>) による検索が可能になります。

図 6 - 23 逆アセンブル・サーチ・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

逆アセンブル・ウィンドウをカレント・ウィンドウにし、[表示]メニュー→[検索...]を選択、または同ウィンドウ上の<Search...> ボタンをクリック

### 各エリア説明

#### (1) サーチ条件指定エリア

Find What:	サーチ・データを指定するエリアです（最大 256 文字）。 デフォルトでは呼び出したウィンドウ内で選択した文字列が表示されますが、必要に応じて変更可能です。 入力履歴は最大 16 個まで残すことができます。	
Match Case	指定したサーチ・データの太文字／小文字を区別して検索する場合、チェックします。	
Scan Whole Region	指定した範囲全体を検索する場合、チェックします。	
Direction	検索方向を選択するエリアです。	
	Up	前方検索。現在カーソルのある位置より前（表示の上）方向に検索します。
	Down	後方検索。現在カーソルのある位置より後ろ（表示の下）方向に検索します（デフォルト）。
Address:	検索するアドレスを指定するエリアです。 数値入力時のデフォルト進数は 16 進です。アドレス指定はシンボルや式での指定も可能です（「表 5 - 6 シンボルでの指定方法」参照）。	

## (2) 機能ボタン

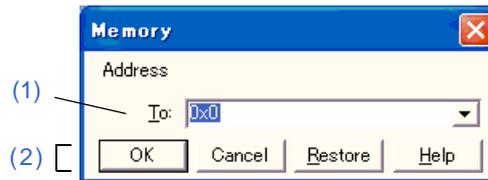
Find Next	設定に従い検索します。 検索の結果、一致する文字列は反転表示されます。 連続して検索する場合には、再度このボタンをクリックします。
Set Find	指定された条件を検索条件として設定し、ダイアログをクローズします。
Stop (検索中)	検索を中断します。
Close	このダイアログをクローズします。(検索中は、このボタンが <Stop> ボタンに変化しています。)
Help	このダイアログのオンライン・ヘルプを表示します。

## アドレス指定ダイアログ

次に示すウィンドウの表示開始アドレスを指定します。

- メモリ・ウィンドウ
- 逆アセンブル・ウィンドウ
- IOR ウィンドウ

図 6-24 アドレス指定ダイアログ (例: メモリ・ウィンドウ)



- オープン方法
- 各エリア説明

### オープン方法

対象ウィンドウをカレント・ウィンドウにし、[表示]メニュー→[移動...]を選択

### 各エリア説明

#### (1) アドレス指定エリア

Address	表示開始アドレスを指定します。	
	To:	デフォルトでは呼び出したウィンドウ内で選択した文字列、またはカレント PC 値等が表示されますが、必要に応じて変更可能です。数値入力時のデフォルト進数は 16 進です。アドレス指定はシンボルや式での指定も可能です (「表 5-6 シンボルでの指定方法」参照)。入力履歴は最大 16 個まで残すことができます。

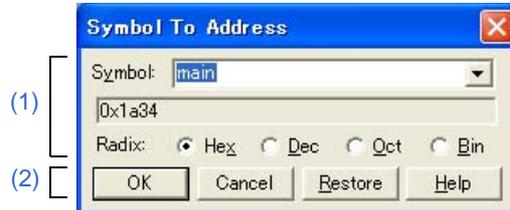
#### (2) 機能ボタン

OK	指定したアドレスから対象ウィンドウの表示を行います。
Cancel	このダイアログをクローズします。
Restore	このダイアログをオープンする前の状態に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

## シンボル変換ダイアログ

指定した変数や関数のアドレス、シンボルの値を表示します（「5.3.4 シンボル変換」参照）。

図 6 - 25 シンボル変換ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

[表示]メニュー→[シンボル変換...]を選択

### 各エリア説明

#### (1) シンボル変換エリア

Symbol:	変換する変数、関数名、シンボル名、または行番号等を指定するエリアです（「表 5 - 6 シンボルでの指定方法」参照）。このエリアを変更し <OK> ボタンをクリックすることにより、変換結果が下のエリアに表示されます。数値入力時のデフォルト進数は 10 進です。入力履歴は最大 16 個まで残すことができます。	
変換結果表示エリア	" Symbol: " で指定した変数、関数のアドレス、シンボルの値、行番号のアドレス、式の値等を表示するエリアです。なお、I/O ポート名、周辺 I/O レジスタ名を指定した場合にはアドレス値を、レジスタ名はレジスタ内容を、PSW フラグ名はフラグ値がそれぞれ表示されます。ビット・シンボルを指定した場合には、Address.bit の形式に変換されます。また、ビット・シンボルを含む式は指定することはできません。	
Radix:	変換結果の表示進数を選択するエリアです。	
	Hex	16 進数（デフォルト）
	Dec	10 進数
	Oct	8 進数
	Bin	2 進数

## (2) 機能ボタン

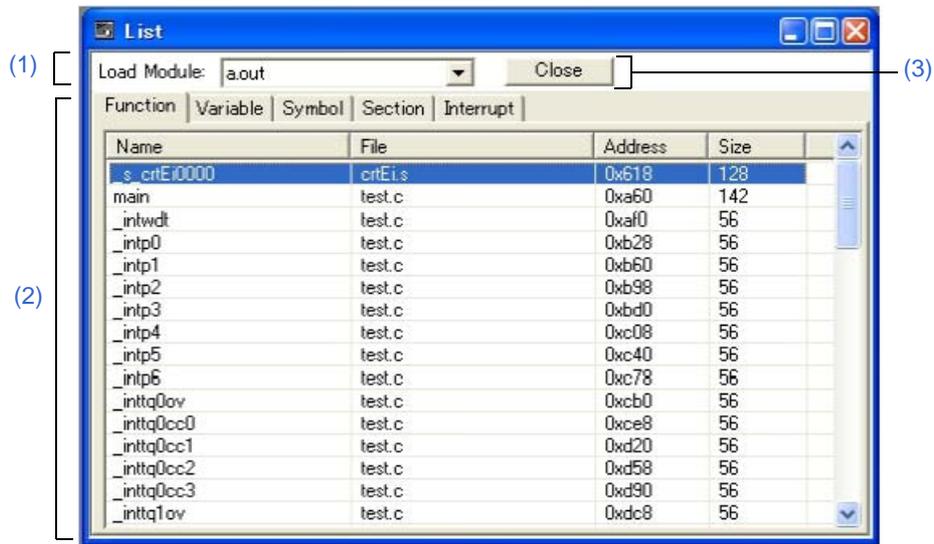
OK	" Symbol: " の内容を変更したときは、そのシンボルを変換します。変換後、" Symbol: " が未変更のときは、ダイアログをクローズします。
Cancel	このダイアログをクローズします。
Restore	入力したデータを元に戻します。 すでに <OK> ボタンを押している場合には、<OK> ボタンを押した直後の状態にデータを戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

## リスト・ウィンドウ

関数、変数、シンボル、セクション、割り込み要求名の一覧表示を行います。

リスト・ウィンドウ上の表示データは、CSV 形式での保存が可能です（「5.15.2 ウィンドウの表示情報（表示ファイル）」参照）。なお、選択タブごとに保存されます。

図 6 - 26 リスト・ウィンドウ



- オープン方法
- 各エリア説明
- コンテキスト・メニュー

### オープン方法

[ブラウザ]メニュー→[リスト]を選択

## 各エリア説明

### (1) Load Module:

ダウンロードしたロード・モジュール・ファイルを選択します。

ロード・モジュール・ファイルをダウンロードしていない場合は空欄になります。

### (2) 一覧表示エリア

タブ（関数／変数／シンボル／セクション／割り込み要求名）別に名称、サイズ、アドレスなどを一覧表示します。表示内容結果の更新は、ロード・モジュール・ファイルのダウンロード後に自動的に行われます。

ロード・モジュール・ファイルをダウンロードしていない場合は空欄になります。

#### (a) [Function] タブ選択時

Name	関数名 (アセンブラ・ソース・ファイルの場合は、ファイル単位に関数として表示)
File	関数が定義されているファイル名
Address	関数の開始アドレス
Size	関数のサイズ (単位: バイト)

#### (b) [Variable] タブ選択時

Name	変数名
File	変数が定義されているファイル名
Address	変数の開始アドレス
Size	変数のサイズ (単位: バイト)

#### (c) [Symbol] タブ選択時

Name	シンボル名
Address	シンボルのアドレス

#### (d) [Section] タブ選択時

Name	セクション名
Type	セクションのタイプ (code, data)
Address	セクションの開始アドレス
Size	セクションのサイズ (単位: バイト)

## (e) [Interrupt] タブ選択時

Name	割り込み要求名
Type	割り込みタイプ (nonmaskable, maskable, software, security id, flash mask option)
Status	プログラムでの使用状況 (use, nonuse) ---- : 不明
Address	割り込みハンドラの開始アドレス
Size	割り込みハンドラのサイズ (単位 : バイト) "Status" が "use" 以外の場合には、最大サイズ

また、このエリアは、選択行の開始アドレス値をジャンプ・ポインタとしてソース・テキスト・ウィンドウ、逆アセンブル・ウィンドウ、またはメモリ・ウィンドウにジャンプする機能を持っています。

ジャンプ先のウィンドウでは、ジャンプ・ポインタから表示されます。

ジャンプ機能は、ジャンプ元の行を選択し、[ジャンプ]メニュー→[ソース/アセンブル/メモリ]を選択することにより行います。また、ジャンプ元の行をダブルクリックするだけでも上記いずれかの適切なウィンドウへジャンプさせることができます。

**参考** 各列のタイトル（ラベル上）をクリックすることにより、表示アイテムがソートされます（昇順/降順はクリックにより切り替わります）。

また、各列の幅は変更可能ですが、保存することはできません。

## (3) 機能ボタン

Close	このウィンドウをクローズします。
-------	------------------

## コンテキスト・メニュー

ソース	カーソル位置のデータ値をジャンプ先アドレスとして、該当するソース・テキストおよびソース行を表示します（「5.16.2 ジャンプ機能」参照）。 ただし、ジャンプ先アドレスに行情報がない場合には、ジャンプすることはできません。 ソース・テキスト・ウィンドウをオープンします。 アクティブ状態のソース・テキスト・ウィンドウが表示されている場合は、そのウィンドウを最前面（操作対象）に表示します。
アセンブル	カーソル位置のデータ値をジャンプ先アドレスとして、そのアドレスから逆アセンブル表示します（「5.16.2 ジャンプ機能」参照）。 逆アセンブル・ウィンドウをオープンします。 アクティブ状態の逆アセンブル・ウィンドウが表示されている場合は、そのウィンドウを最前面（操作対象）に表示します。
メモリ	カーソル位置のデータ値をジャンプ先アドレスとして、そのアドレスから逆アセンブル表示します（「5.16.2 ジャンプ機能」参照）。 逆アセンブル・ウィンドウをオープンします。 アクティブ状態の逆アセンブル・ウィンドウが表示されている場合は、そのウィンドウを最前面（操作対象）に表示します。

## ウォッチ・ウィンドウ

指定されたウォッチ・データの表示、および変更を行うウィンドウです（「5.6 ウォッチ機能」参照）。

大域的なウォッチ・データ（グローバル変数やパブリック・シンボル等）は、メモリ・ウィンドウと同様にプログラム実行中でも表示可能です。

データ値の更新結果と書き換えは、メモリ・ウィンドウに反映されます。

ウォッチ・データの登録は、ソース・テキスト・ウィンドウ、または逆アセンブル・ウィンドウ上の <Watch...> ボタンにより行います（「5.6.3 ウォッチ・データの登録、削除」参照）。

このウィンドウでは、コンテキスト・メニューにより変数へのブレーク・ポイント設定が容易に行えます。

**注意 1** 同名のローカル変数とグローバル変数が存在する場合には、ローカル変数が優先されます。

**注意 2** ユーザ・プログラム実行中では、RRM 領域のバンク境界（256 バイト）をまたぐアクセスにおいては、上位アドレスの値が不正な表示になります。

このため、該当する変数値は、ウォッチ・ウィンドウ上で赤色に表示されます（この時、メモリ・ウィンドウ上での表示も不正になりますが、赤色表示にはなりません）。

上記の現象は、ユーザ・プログラム実行中のみです。ブレーク時には正常値が表示されます。[IECUBE]

**参考** ウォッチ・ウィンドウに表示可能な最大行数は 10000 行です。

図 6-27 ウォッチ・ウィンドウ



- オープン方法
- 各エリア説明
- [表示]メニュー（ウォッチ・ウィンドウ専用部分）
- コンテキスト・メニュー

### オープン方法



ボタンをクリック、または[ブラウザ]メニュー→[ウォッチ]を選択

## 各エリア説明

### (1) ウォッチ・データ表示／変更エリア

左側にはシンボル名を、右側にはデータ値を表示しています。

シンボル名表示エリア	<p>変数名、シンボル名、およびその型や構造体／共用体のタグ名を表示するエリアです。配列、ポインタ型変数、および構造体／共用体には先頭には "+" が表示され、ダブルクリックすることにより展開表示されます（先頭文字が "+" から "-" へ）。</p> <p>登録済みウォッチ・データの変更は、変更項目を選択し、<a href="#">コンテキスト・メニュー</a>→[ウォッチ変更...]を選択することによりオープンする<a href="#">ウォッチ変更ダイアログ</a>で行います。この際、配列の要素や、構造体／共用体のメンバ等、開かれた階層の行は変更できません。</p> <p>ウォッチ・ウィンドウから変数、またはシンボルにアクセス系のブレーク・ポイントを設定した場合、このエリアは、金色で強調表示されます。</p>	
	配列	"+"をダブルクリックすることにより、変数の全要素が配列変数の型に従い表示される。
	ポインタ型変数	"+"をダブルクリックすることにより、ポインタで示されるデータが表示される。
	構造体／共用体	"+"をダブルクリックすることにより、構造体／共用体の全メンバがメンバ変数の型に従い表示される。 ただし、構造体／共用体内に構造体／共用体が定義されている場合、内部の構造体／共用体の構造体／共用体名までが表示される。 内部の構造体／共用体も、同様に "+" 表示の展開が可能。
データ値表示／設定エリア	<p>データ値の表示と変更を行うエリアです。値は実行の停止とともに更新されます。値を保存しておきたい場合は、[ファイル]メニュー→[名前を付けて保存...]により保存可能です。なお、データ値の取得に失敗した場合は、空欄になります。</p> <p>値の変更は直接入力により行います。変更箇所は<b>赤色</b>で表示され Enter キーを押すことにより、ターゲット・メモリに書き込まれます。書き込み前の内容は ESC キーにより破棄できます。</p>	
	表示データ	内容、表示形式
	整数	16進数 ( <b>0x</b> xxxx) 10進数 (xxxx) 8進数 ( <b>0</b> xxxx) 2進数 ( <b>0b</b> xxxx)
	文字列	"文字列"
	列挙型	メンバ名
	スコープ指定されているデータ	指定されたスコープに従い表示
	浮動小数点型	単精度／倍精度に対応。入力／表示形式ともに次の通り。 [+ -] inf [+ -] nan [+ -] 整数部 e [+ -] 指数部 [+ -] 整数部 . 小数部 [e [+ -] 指数部
	"?"	スコープの変化や最適化コンパイル等により無効になったデータ

**参考 1** 配列の要素数が展開する時間が一定時間を越える程多い場合には、展開時に警告が表示されます。

**参考 2** 表示進数は、[コンテキスト・メニュー](#)により、変数ごとの変更が可能です。また整数の表示形式、および配列の要素数は、[デバッガ・オプション設定ダイアログ](#)により変更可能です。

## (2) 機能ボタン

Add...	ウォッチ登録ダイアログをオープンします。 ウォッチ登録ダイアログ上でウォッチ・データを指定し、<Add> ボタンをクリックすることにより、指定したウォッチ・データがウォッチ・ウィンドウに追加されます。
Delete	選択したウォッチ・データをウィンドウから削除します。
Up	選択行を 1 行上に移動します。
Down	選択行を 1 行下に移動します。
Refresh	ウィンドウの内容を最新のデータに更新します。
Close	このウィンドウをクローズします。

## [表示]メニュー（ウォッチ・ウィンドウ専用部分）

ウォッチ・ウィンドウのアクティブ時には、共通の [\[表示\]メニュー](#) に以下の項目が追加されます。  
作用対象は、選択している項目のみです。

ブレーク・イベント作成	選択している項目に対して以下のブレーク・イベントを設定します。
この変数へのアクセスでブレーク	選択している項目に対して Read / Write アクセスのブレーク・イベントを設定します。
この変数への書き込みでブレーク	選択している項目に対して Write アクセスのブレーク・イベントを設定します。
この変数からの読み出しでブレーク	選択している項目に対して Read アクセスのブレーク・イベントを設定します。
解除	選択項目に対して設定されているブレーク・イベントを削除します。
イベント内容の表示	選択した変数名のイベント情報を表示します。イベントが設定されている場合、 <a href="#">イベント・ダイアログ</a> をオープンします。
2 進	選択行を 2 進数表示
8 進	選択行を 8 進数表示
10 進	選択行を 10 進数表示
16 進	選択行を 16 進数表示
文字列	選択行を文字列表示
自動	変数ごとの既定値で表示 シンボルの場合は、 <a href="#">デバッガ・オプション設定ダイアログ</a> の設定にしたがって表示（デフォルト）
バイト	選択行を 8 ビット単位で表示
ハーフワード	選択行を 16 ビット単位で表示

ワード	選択行を 32 ビット単位で表示
適合	選択行の変数を変数ごとの既定値で表示（デフォルト） C 言語のシンボルの場合は、この項目のみ有効となります。 アセンブラのシンボルの場合は、 <a href="#">デバッグ・オプション設定ダイアログ</a> の設定に従って表示します。
アップ	選択行を 1 行上に移動
ダウン	選択行を 1 行下に移動
強制読み込み	読み込むと値が変化するために読み込み禁止にしている IOR、または <a href="#">I/O ポート追加ダイアログ</a> で追加した I/O ポート、I/O プロテクト領域のデータを強制的に読み込みます。

## コンテキスト・メニュー

各メニュー項目は、クリックしたマウス・ポインタの位置ではなく、選択されている行、または項目に対して機能します（同名のメイン・メニューを選択した際と同じ動作）。

この変数へのアクセスでブレーク	選択している項目に対して Read / Write アクセスのブレーク・イベントを設定します。
この変数への書き込みでブレーク	選択している項目に対して Write アクセスのブレーク・イベントを設定します。
この変数からの読み出しでブレーク	選択している項目に対して Read アクセスのブレーク・イベントを設定します。
解除	選択項目に対して設定されているブレーク・イベントを削除します。
RRM 設定 ...	RRM 機能のサンプリング範囲を設定します。 <a href="#">RRM 設定ダイアログ</a> をオープンします [IECUBE]
イベント内容の表示	選択した変数名のイベント情報を表示します。イベントが設定されている場合、 <a href="#">イベント・ダイアログ</a> をオープンします。
ウォッチ変更 ...	選択したウォッチ・データの変更を行います。 <a href="#">ウォッチ変更ダイアログ</a> をオープンします。
ウォッチ削除	選択したウォッチ・データをウィンドウから削除します。
2 進	選択行を 2 進数表示
8 進	選択行を 8 進数表示
10 進	選択行を 10 進数表示
16 進	選択行を 16 進数表示
文字列	選択行を文字列表示
自動	変数ごとの既定値で表示 シンボルの場合は、 <a href="#">デバッグ・オプション設定ダイアログ</a> の設定にしたがって表示（デフォルト）
バイト	選択行を 8 ビット単位で表示
ハーフワード	選択行を 16 ビット単位で表示
ワード	選択行を 32 ビット単位で表示

適合	選択行の変数を変数ごとの既定値で表示（デフォルト） C 言語のシンボルの場合は、この項目のみ有効となります。アセンブラのシンボルの場合は、 <a href="#">デバッガ・オプション設定ダイアログ</a> の設定に従い表示します。
アップ	選択行を 1 行上に移動
ダウン	選択行を 1 行下に移動

## クイック・ウォッチ・ダイアログ

指定されたウォッチ・データの一時的な表示、および変更を行います（「5.6 ウォッチ機能」参照）。

図 6 - 28 クイック・ウォッチ・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

ソース・テキスト・ウィンドウ／逆アセンブル・ウィンドウをカレント・ウィンドウにし、[表示]メニュー→[クイック・ウォッチ ...]を選択、または同ウィンドウ上の <Quick...> ボタンをクリック

### 各エリア説明

#### (1) 表示形式指定エリア

表示進数選択エリア	表示するウォッチ・データの表示進数を選択します。	
	Proper	変数の場合：変数ごとの規定値で表示 シンボルの場合：デバッガ・オプション設定ダイアログで設定された進数で表示
	Hex	16 進数表示
	Dec	10 進数表示
	Oct	8 進数表示
	Bin	2 進数表示
	String	文字列表示

表示サイズ選択エリア 	表示するウォッチ・データの表示サイズを選択します。 C 言語の変数やレジスタ等表示するサイズが固定である場合は選択不可です。	
	Adaptive	変数の場合：変数ごとの規定値で表示 シンボルの場合： <a href="#">デバッグ・オプション設定ダイアログ</a> で設定されたサイズで表示
	Byte	8 ビット単位で表示
	Half Word	16 ビット単位で表示
	Word	32 ビット単位で表示

**(2) ウォッチ・データ指定エリア**

Name:	ウォッチ・データを指定します。 デフォルトでは呼び出したウィンドウ内で選択した文字列が表示されますが、必要に応じて変更可能です。入力履歴は最大 16 個まで残すことができます。 このエリアの内容を変更し、<View> ボタンをクリックすることにより指定されたデータが下欄に表示されます。
Number:	表示するデータの個数を指定します（空欄、または 1 - 256）。 空欄を指定した場合は単純変数として、1 以上を指定した場合は配列変数として <a href="#">ウォッチ・ウィンドウ</a> に表示されます。 配列変数として表示された場合、データの先頭に "+" が表示され、ダブルクリックすることによりデータの全要素がデータの型に従い展開表示されます（データの先頭文字は "-" 表示に切り替わります）。 C 言語の変数やレジスタ等、表示する個数が固定である場合は指定が無効になります。

**(3) ウォッチ・データ表示エリア**

左側にはシンボル名を、右側にはデータ値を表示しています。

シンボル名表示エリア	ウォッチ・データ（変数名、シンボル名、およびその型や構造体／共用体のタグ名）を表示するエリアです（ <a href="#">ウォッチ・ウィンドウ参照</a> ）。 このエリアを編集することはできません。
データ値表示／設定エリア	データ値の表示と変更を行うエリアです（ <a href="#">ウォッチ・ウィンドウ参照</a> ）。

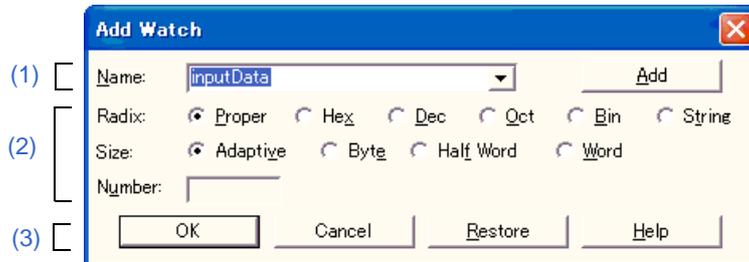
**(4) 機能ボタン**

View	<a href="#">(2) ウォッチ・データ指定エリア</a> で指定したデータを下欄に表示します。
Add	<a href="#">(2) ウォッチ・データ指定エリア</a> で指定したデータを <a href="#">ウォッチ・ウィンドウ</a> に登録します。
Close	このウィンドウをクローズします。ターゲット・メモリに書き込まれていない変更内容は破棄されます。

## ウォッチ登録ダイアログ

ウォッチ・ウィンドウに表示するウォッチ・データの登録を行います（「5.6 ウォッチ機能」参照）。同名のシンボル名のデータを複数登録することができます。

図 6 - 29 ウォッチ登録ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

[表示]メニュー → [ウォッチ登録...] を選択, またはウォッチ・ウィンドウ上の <Add...> ボタンをクリック

### 各エリア説明

#### (1) ウォッチ・データ登録エリア

Name:	ウォッチ・ウィンドウに登録するシンボルを指定します（「表 6 - 12 ウォッチ・ウィンドウ入力形式」参照）。デフォルトでは呼び出したウィンドウ内で選択した文字列が表示されますが、必要に応じて変更可能です。ただし、選択した文字列がない場合には、空欄になります。入力履歴は最大 16 個まで残すことができます。
<Add>	指定したデータをウォッチ・ウィンドウに追加します。このダイアログはオープンしたままです。

表 6 - 12 ウォッチ・ウィンドウ入力形式

- C 言語の変数名	
変数式 : 変数名	
変数式 [ 定数値   変数名 ]	配列の要素
変数式 . メンバ名	構造体 / 共用体の実体のメンバ
変数式 -> メンバ名	ポインタの指す構造体 / 共用体のメンバ
* 変数式	ポインタ変数の値
& 変数式	変数が配置されたアドレス
- レジスタ名	
- IOR 名, IOR ビット名	
- ラベル, および即値のアドレス	
- レジスタ名 . ビット	
- IOR 名 . ビット	
- ラベル名 . ビット, 即値のアドレス . ビット	
- スコープの指定	

スコープ指定した場合の扱いについては次の通りです。

表 6 - 13 スコープで指定した場合の変数の扱い

スコープ指定	プログラム名	ファイル名	関数名	変数名
<b>prog\$file#func#var</b>	prog	file	func	var
<b>prog\$file#var</b>	prog	file	グローバル	var
<b>prog\$func#var</b>	prog	グローバル	func	var
<b>prog\$var</b>	prog	グローバル	グローバル	var
<b>file#func#var</b>	カレント	file	func	var
<b>file#var</b>	カレント	file	グローバル	var
<b>func#var</b>	カレント	カレント	func	var
<b>var</b>	カレント	カレント	カレント	var

## (2) 表示形式変更エリア

Radix:	表示進数を選択します。	
	Proper	変数の場合：変数ごとの規定値で表示 シンボルの場合：デバッガ・オプション設定ダイアログで設定された進数で表示
	Hex	16 進数表示
	Dec	10 進数表示
	Oct	8 進数表示
	Bin	2 進数表示
	String	文字列表示
Size:	表示サイズを選択します。C 言語の変数やレジスタ等表示するサイズが固定である場合は選択不可です。	
	Adaptive	変数の場合：変数ごとの規定値で表示 シンボルの場合：デバッガ・オプション設定ダイアログで設定されたサイズで表示
	Byte	8 ビット単位で表示
	Half Word	16 ビット単位で表示
	Word	32 ビット単位で表示
Number:	<p>表示するデータの個数を指定します（空欄，または 1 - 256）。</p> <p>空欄を指定した場合は単純変数として，1 以上を指定した場合は配列変数として <b>ウォッチ・ウィンドウ</b> に表示されます。</p> <p>配列変数として表示された場合，データの先頭に "+" が表示され，ダブルクリックすることによりデータの全要素がデータの型に従い展開表示されます（データの先頭文字は "-" 表示に切り替わります）。</p> <p>C 言語の変数やレジスタ等，表示する個数が固定である場合は指定が無効になります。</p>	

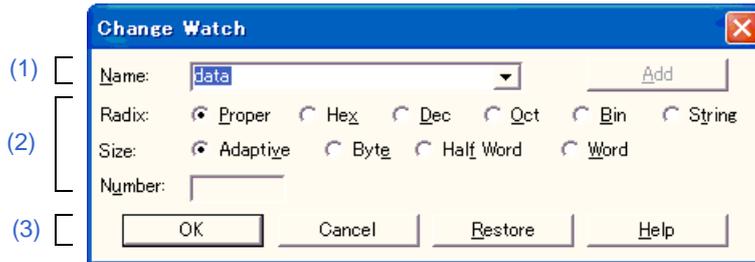
## (3) 機能ボタン

OK	指定したデータを <b>ウォッチ・ウィンドウ</b> に追加します。 このダイアログをクローズします。
Cancel	このダイアログをクローズします。
Restore	このダイアログをオープンする前の状態に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

## ウォッチ変更ダイアログ

ウォッチ・ウィンドウで選択されている行のデータを変更します（「5.6 ウォッチ機能」参照）。  
配列の要素や、構造体、共用体のメンバ等、開かれた階層の行を変更することはできません。  
データ変更により、選択されている行の内容が変更後のデータで置き換えられます。  
シンボル名を変更した結果、すでに同名のシンボル名のデータが存在しても変更を許可します。

図 6 - 30 ウォッチ変更ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

ウォッチ・ウィンドウをカレント・ウィンドウにし、[表示]メニュー→[ウォッチ変更...]を選択

### 各エリア説明

#### (1) ウォッチ・データ変更エリア

Name:	ウォッチ・ウィンドウで選択されている行のシンボル名を変更します（「表 6 - 12 ウォッチ・ウィンドウ入力形式」参照）。 シンボル名を変更した結果、すでに同名のシンボル名のデータが存在しても変更を許可します。 デフォルトでは呼び出したウィンドウ内で選択した文字列が表示されます。入力履歴は最大 16 個まで残すことができます。
<Add>	選択不可

## (2) 表示形式変更エリア

Radix:	ウォッチ・ウィンドウで選択されている行の表示進数を変更します	
	Proper	変数の場合：変数ごとの規定値で表示 シンボルの場合：デバッガ・オプション設定ダイアログで設定された進数で表示
	Hex	16 進数表示
	Dec	10 進数表示
	Oct	8 進数表示
	Bin	2 進数表示
Size:	ウォッチ・ウィンドウで選択されている行の表示サイズを変更します。C 言語の変数やレジスタ等表示するサイズが固定である場合は選択不可です。	
	Adaptive	変数の場合：変数ごとの規定値で表示 シンボルの場合：デバッガ・オプション設定ダイアログで設定されたサイズで表示
	Byte	8 ビット単位で表示
	Half Word	16 ビット単位で表示
	Word	32 ビット単位で表示
Number:	<p>ウォッチ・ウィンドウで選択されている行の表示データの個数を変更します（空欄、または 1 - 256）。</p> <p>空欄を指定した場合は単変数として、1 以上を指定した場合は配列変数としてウォッチ・ウィンドウに表示されます。</p> <p>配列変数として表示された場合、データの先頭に "+" が表示され、ダブルクリックすることによりデータの全要素がデータの型に従い展開表示されます（データの先頭文字は "-" 表示に切り替わります）。</p> <p>C 言語の変数やレジスタ等、表示する個数が固定である場合は指定が無効になります。</p>	

## (3) 機能ボタン

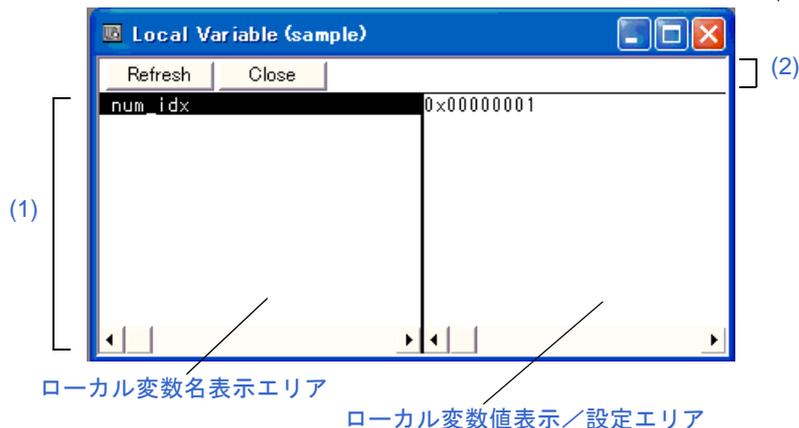
OK	ウォッチ・ウィンドウで選択されている行のデータを、指定されたデータに置き替え、このダイアログをクローズします。
Cancel	このダイアログをクローズします。
Restore	このダイアログをオープンする前の状態に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

## ローカル変数ウィンドウ

カレント関数内ローカル変数の表示、および値の変更を行います（「5.6.2 ローカル変数値の表示、変更」参照）。  
スタック・トレース・ウィンドウのジャンプ機能と連動し、ソース・テキスト・ウィンドウへジャンプした場合は、ジャンプした関数内のローカル変数を表示します。

このウィンドウでは、コンテキスト・メニュー、機能ボタン等によるさまざまな操作が可能です。

図 6 - 31 ローカル変数ウィンドウ



- オープン方法
- 各エリア説明
- [表示]メニュー（ローカル変数ウィンドウ専用部分）
- コンテキスト・メニュー

### オープン方法

 ボタンをクリック、または[ブラウザ]メニュー → [ローカル変数]を選択

### 各エリア説明

#### (1) ローカル変数表示/変数値変更エリア

ローカル変数名表示エリア	ローカル変数名を表示するエリアです。 表示形式はウォッチ・ウィンドウの「(1) ウォッチ・データ表示/変更エリア」と同じです。 Auto 変数、内部 Static 変数、Register 変数の表示が可能です。 このエリアは、カレント関数内のローカル変数が自動的に表示されるため、編集することはできません。
ローカル変数値表示/設定エリア	ローカル変数値の表示と変更を行うエリアです。 表示形式はウォッチ・ウィンドウの「(1) ウォッチ・データ表示/変更エリア」と同じです

**(2) 機能ボタン**

Refresh	ウィンドウの内容を最新のデータに更新します。
Close	このウィンドウをクローズします。

**[表示]メニュー（ローカル変数ウィンドウ専用部分）**

ローカル変数ウィンドウのアクティブ時には、共通の **[表示]メニュー** に以下の項目が追加されます。

2進	2進数表示
8進	8進数表示
10進	10進数表示
16進	16進数表示
文字列	文字列表示
自動	変数ごとの既定値で表示（デフォルト）

**コンテキスト・メニュー**

各メニュー項目は、クリックしたマウス・ポインタの位置ではなく、選択されている行、または項目に対して機能します（同名のメイン・メニューを選択した際と同じ動作）。

ウォッチ登録 ...	<a href="#">ウォッチ登録ダイアログ</a> をオープンします。
2進	選択行を2進数表示
8進	選択行を8進数表示
10進	選択行を10進数表示
16進	選択行を16進数表示
文字列	選択行を文字列表示
自動	変数ごとの既定値で表示 シンボルの場合は、 <a href="#">デバッガ・オプション設定ダイアログ</a> の設定にしたがって表示（デフォルト）

## スタック・トレース・ウィンドウ

現在のユーザ・プログラムのスタック内容の表示、および変更を行います（「5.6.7 スタック・トレース表示機能」参照）。

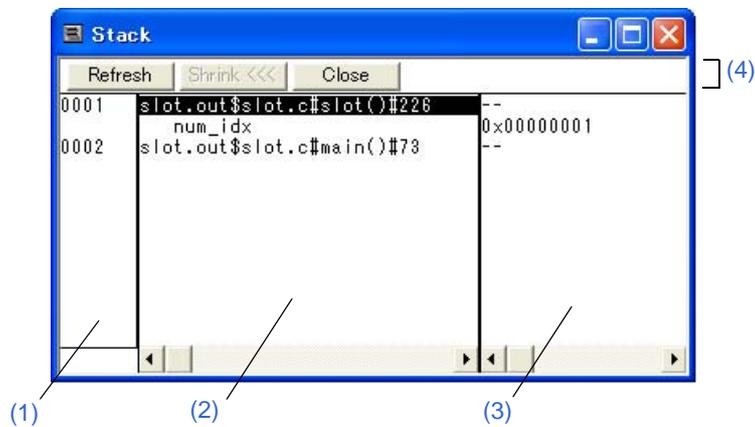
ジャンプ機能により、スタック内容に対応するウィンドウへのジャンプが可能です。

このウィンドウでは、コンテキスト・メニュー、機能ボタン等による操作が可能です。

**注意** スタック・トレース表示機能は、スタック・フレームを生成しない関数がある場合には、正しく表示しないことがあります。

**参考** 関数のプロローグ処理中、およびエピローグ処理中は [ERROR] を表示することがあります。

図 6 - 32 スタック・トレース・ウィンドウ



- オープン方法
- 各エリア説明
- [表示]メニュー（スタック・トレース・ウィンドウ専用部分）
- コンテキスト・メニュー

### オープン方法

 ボタンをクリック、または [ブラウザ]メニュー → [スタック] を選択

## 各エリア説明

### (1) スタック・フレーム番号表示エリア

スタック・フレーム番号を表示しています。

スタック・フレーム番号は1から始まる自然数で、スタックのネストが浅くなるほど大きな番号になります。つまり、ある関数に対してスタック番号が1つ大きい関数が、ある関数の呼び出し元の関数となります。

### (2) スタック・フレーム内容表示エリア

スタック・フレーム内容を表示するエリアです。

関数名、またはローカル変数名を表示します。ただし、このエリアを編集することはできません。

関数の場合	[プログラム名\$ファイル名#関数名(引数リスト)#行番号]と表示されます。 この行をダブルクリックすると、ジャンプ機能によりソース・テキスト・ウィンドウへジャンプします(ジャンプした関数内のローカル変数がローカル変数ウィンドウに表示されます)。関数にローカル変数がある場合は、ローカル変数を次行以降に表示します。
ローカル変数の場合	型と変数名が表示されます。 表示形式はウォッチ・ウィンドウの「(1)ウォッチ・データ表示/変更エリア」と同じです。なお、内部 Static 変数、Register 変数は表示しません。

### (3) スタック内容表示/設定エリア

スタック内容の表示、および変更を行うエリアです。

関数の場合	"--"が表示され、変更できません。
ローカル変数の場合	変数値を表示します。 表示形式はウォッチ・ウィンドウの「(1)ウォッチ・データ表示/変更エリア」と同じです。

### (4) 機能ボタン

Refresh	ウィンドウの内容を最新のデータに更新します。
Shrink <<<	選択した関数に対するローカル変数リストの展開表示を解除します。
Expand >>> (展開表示解除中)	選択した関数に対するローカル変数リストを展開表示します。
Close	このウィンドウをクローズします。

## [表示]メニュー（スタック・トレース・ウィンドウ専用部分）

スタック・トレース・ウィンドウのアクティブ時には、共通の [表示] メニューに以下の項目が追加されます。

2 進	2 進数表示
8 進	8 進数表示
10 進	10 進数表示
16 進	16 進数表示
文字列	文字列表示
自動	変数ごとの既定値で表示（デフォルト）

## コンテキスト・メニュー

2 進	2 進数表示
8 進	8 進数表示
10 進	10 進数表示
16 進	16 進数表示
文字列	文字列表示
自動	変数ごとの既定値で表示（デフォルト）
ソース	カーソル位置のデータ値をジャンプ先アドレスとして、該当するソース・テキストおよびソース行を表示します（「5.16.2 ジャンプ機能」参照）。 ソース・テキスト・ウィンドウをオープンします。 アクティブ状態のソース・テキスト・ウィンドウが表示されている場合は、そのウィンドウを最前面（操作対象）に表示します。
アセンブル	カーソル位置のデータ値をジャンプ先アドレスとして、そのアドレスから逆アセンブル表示します（「5.16.2 ジャンプ機能」参照）。 逆アセンブル・ウィンドウをオープンします。 アクティブ状態の逆アセンブル・ウィンドウが表示されている場合は、そのウィンドウを最前面（操作対象）に表示します。
メモリ	カーソル位置のデータ値をジャンプ先アドレスとして、そのアドレスからメモリ内容を表示します（「5.16.2 ジャンプ機能」参照）。 メモリ・ウィンドウをオープンします。 アクティブ状態のメモリ・ウィンドウが表示されている場合は、そのウィンドウを最前面（操作対象）に表示します。

## メモリ・ウィンドウ

メモリ内容の表示、および変更を行います（「5.7 メモリ操作機能」参照）。

このウィンドウでは、コンテキスト・メニュー、機能ボタン等によるさまざまな操作が可能です。

また、このウィンドウには、アクティブ状態とスタティック状態の2つの状態があり、アクティブ状態時には、トレース結果とウィンドウの連結機能 [IECUBE] とジャンプ機能を持ちます（「5.16 ウィンドウ共通機能」参照）。

**参考 1** メモリ内容へのアクセス状況（Read, Write, Read および Write）を色により表示可能です（「5.7.2 アクセス・モニタ機能 [IECUBE]」参照）。[IECUBE]

**参考 2** ウィンドウをオープンした際の表示開始位置は、次のようになります。

初回：RAM 領域の先頭アドレスから表示

2 回目以降：最後にアクティブ状態のウィンドウが閉じたアドレスから表示

（アクティブ状態のウィンドウが一度も閉じていない場合は、初回の表示開始位置から表示）

図 6 - 33 メモリ・ウィンドウ

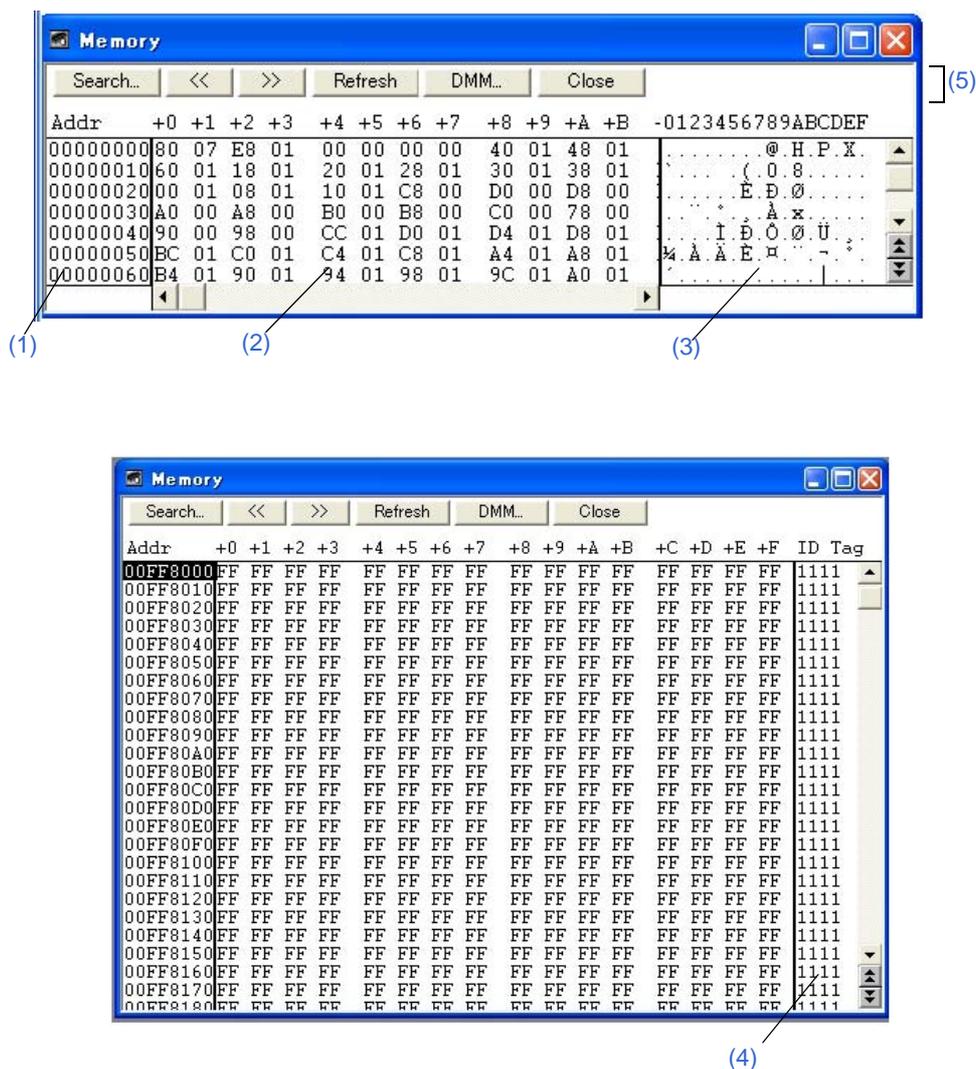
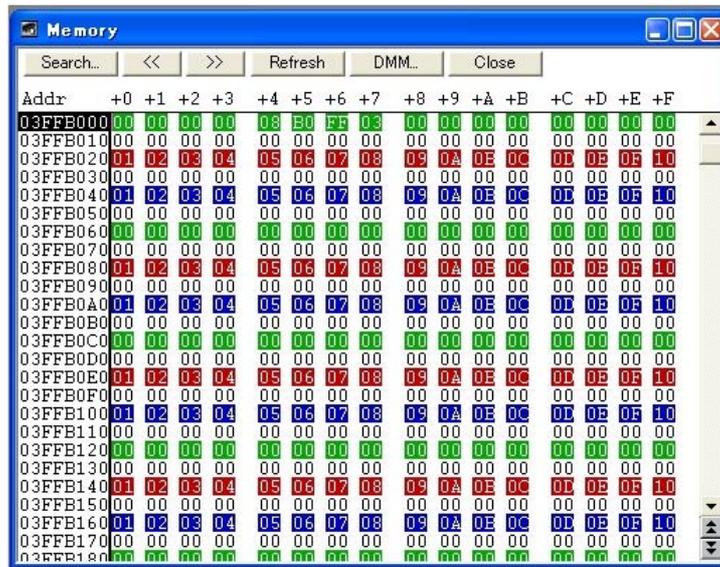


図6-34 メモリ・ウィンドウ (RRM 機能選択時)



- オープン方法
- 各エリア説明
- [表示]メニュー (メモリ・ウィンドウ専用部分)
- コンテキスト・メニュー

## オープン方法



ボタンをクリック, または [ブラウザ] メニュー → [メモリ ...] を選択

## 各エリア説明

### (1) Addr (アドレス表示エリア)

メモリ・アドレスを表示するエリアです。

このエリアは、マウスでクリックすることにより、任意の位置を選択することができます。選択されたアドレスは、反転表示されます。

**(2) +0 +1 +2.... (メモリ表示エリア)**

メモリ内容の表示と変更, およびアクセス状況表示を行うエリアです (「5.7.2 アクセス・モニタ機能 [IECUBE]」参照)。

表示		表示ファイル 保存時のマーク	意味
	緑	R	リード
	赤	W	ライト
	青	A	リード&ライト
00	無色	なし	測定対象外

メモリ内容の変更は直接入力により行います。変更箇所は赤色で表示され Enter キーを押すことにより、ターゲット・メモリに書き込まれます。書き込み前の内容は ESC キーにより破棄できます。一度に変更できるのは 256 バイトまでです。

**参考** ユーザ・プログラム実行中のメモリ内容の変更は、<DMM...> ボタンをクリックすることによりオープンする [DMM ダイアログ](#)で行います。

**(3) 0 1 2 3.... (アスキー表示エリア)**

メモリ内容のアスキー表示と変更を行うエリアです。

このエリアは、[表示]メニュー→[アスキー表示]を選択することにより表示されます。

メモリ内容をアスキー文字にて変更することができます。

変更方法は (2) +0 +1 +2.... (メモリ表示エリア) と同じです。

**注意** 表示アドレスを変更した場合、アスキー表示エリアのカーソル位置は同調しません。

**(4) ID Tag (ID タグ表示エリア)**

データ・フラッシュ・メモリの ID タグを表示するエリアです。

ID タグとは、データ・フラッシュ・メモリの 1 ワードに 1 ビットが割り当てられているパワー・フェイル検出用のビットで、"0" は、そのワードのデータが正常に書き込めたことを意味しています。

このエリアは、"0", または "1" の値で変更が可能です。

[表示]メニュー→[ID タグ表示]を選択することにより、表示の ON/OFF を切り替えられます (デフォルトは非表示)。このエリアは、(3) 0 1 2 3.... (アスキー表示エリア) と排他表示になります。

## (5) 機能ボタン

Search...	メモリ・サーチ・ダイアログをオープンして、メモリ表示内容の文字列、またはメモリ内容を検索します。 選択したデータ（メモリ値）が検索対象としてメモリ・サーチ・ダイアログに表示されます。データを指定しないで開いた場合、キー入力でデータ指定します。 検索の結果は、メモリ・ウィンドウ上で反転表示されます。
<<	メモリ・サーチ・ダイアログで設定した検索条件に一致するメモリ内容を、カーソル位置のアドレスから前方向（表示の上）に検索します。 検索中は <Stop> ボタンに変化します。
>>	メモリ・サーチ・ダイアログで設定した検索条件に一致するメモリ内容を、カーソル位置のアドレスから後ろ方向（表示の下）に検索します。 検索中は <Stop> ボタンに変化します。
Stop（検索中）	検索を中断します。
Refresh	ウィンドウの内容を最新のデータに更新します。
DMM...	DMM ダイアログをオープンします。
Close	このウィンドウをクローズします。

## [表示]メニュー（メモリ・ウィンドウ専用部分）

メモリ・ウィンドウのアクティブ時には、共通の[表示]メニューに以下の項目が追加されます。

2進	2進数表示
8進	8進数表示
10進	10進数表示
16進	16進数表示（デフォルト）
ニブル	4ビット単位で表示
バイト	8ビット単位で表示（デフォルト）
ハーフワード	16ビット単位で表示
ワード	32ビット単位で表示
アスキー表示	アスキー文字の表示／非表示を切り替えます。 チェックあり：表示 チェックなし：非表示（デフォルト）
IDタグ表示	データ・フラッシュ・メモリのIDタグの表示／非表示を切り替えます。 チェックあり：表示 チェックなし：非表示（デフォルト） アスキー表示との排他選択になります。
リトル・エンディアン	リトル・エンディアンで表示します（デフォルト）。
ビック・エンディアン	ビック・エンディアンで表示します。
アクセス・モニタ [IECUBE]	アクセス・モニタ機能に関する設定を行います。
クリア [IECUBE]	アクセス・モニタ機能による表示色をクリアします。
累積表示 [IECUBE]	アクセス状況（メモリ内容の変化）の累積表示をする／しないを切り替えます。 チェックあり：メモリ内容の変化を累積表示 チェックなし：前回更新時からのメモリ内容の変化のみを表示

## コンテキスト・メニュー

各メニュー項目は、クリックしたマウス・ポインタの位置ではなく、選択されている行、または項目に対して機能します（同名のメイン・メニューを選択した際と同じ動作）。

移動 ...	表示位置を移動します。 アドレス指定ダイアログをオープンします。
RRM 設定 ...	RRM 設定ダイアログをオープンします。[IECUBE]
2 進	2 進数表示
8 進	8 進数表示
10 進	10 進数表示
16 進	16 進数表示（デフォルト）
ニブル	4 ビット単位で表示
バイト	8 ビット単位で表示（デフォルト）
ハーフワード	16 ビット単位で表示
ワード	32 ビット単位で表示
アスキー表示	アスキー文字の表示／非表示を切り替えます。 チェックあり：表示 チェックなし：非表示（デフォルト）
ID タグ表示	データ・フラッシュ・メモリの ID タグ の表示 / 非表示を切り替えます。 チェックあり：表示 チェックなし：非表示（デフォルト） アスキー表示との排他選択になります。
アクセス・モニタの クリア [IECUBE]	アクセス・モニタ機能による表示色をクリアします。
累積表示 [IECUBE]	アクセス状況（メモリ内容の変化）の累積表示をする／しないを切り替えます。 チェックあり：メモリ内容の変化を累積表示 チェックなし：前回更新時からのメモリ内容の変化のみを表示

## メモリ・サーチ・ダイアログ

カーソルのあるメモリ・ウィンドウのメモリ内容を検索します（「5.7 メモリ操作機能」参照）。

メモリ・ウィンドウ内のメモリ表示エリアにカーソルがある場合は、指定されたデータをバイナリ・データ列と見なし、アスキー表示エリアではアスキー文字列とみなし、それぞれのエリア内を検索します。

各項目を設定し、<Find Next> ボタンをクリックすることにより検索を開始します。また、<Set Find> ボタンをクリックすることで、メモリ・ウィンドウ上の方向ボタン (<<, >>) による検索が可能になります。

**注意** ノンマップ領域、周辺 I/O レジスタ領域、I/O プロテクト領域は検索しません。

図 6 - 35 メモリ・サーチ・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

メモリ・ウィンドウをカレント・ウィンドウにし、[表示]メニュー → [検索...] を選択、または同ウィンドウ上の <Search...> ボタンをクリック

## 各エリア説明

### (1) サーチ条件指定エリア

Find What:	<p>サーチ・データを指定するエリアです (最大 256 文字)。          デフォルトでは呼び出したウィンドウ内で選択した文字列が表示されますが、必要に応じて変更可能です。          入力履歴は最大 16 個まで残すことができます。</p>	
	メモリ表示エリア 検索時	データは最大 16 個まで指定することができます。個々のデータは " 空白文字 " で区切って指定します。
	アスキー表示エリ ア検索時	データは最大 256 文字まで指定することができます。データ中の " 空白文字 " は空白文字として扱われます。
Unit:	<p>メモリ表示エリア検索時に、指定したサーチ・データの個々のデータを何ビットのデータと解釈して検索するかを選択するエリアです。</p>	
	Byte	8 ビット・データとして検索 (デフォルト)
	Half Word	16 ビット・データとして検索
	Word	32 ビット・データとして検索
Scan Whole Region	指定した範囲全体を検索する場合、チェックします。	
Direction	検索方向を選択するエリアです。	
	Up	前方検索。現在カーソルのある位置より前 (表示の上) 方向に検索します。
	Down	後方検索。現在カーソルのある位置より後ろ (表示の下) 方向に検索します (デフォルト)。
Address:	<p>検索するアドレス範囲を指定するエリアです。          数値入力時のデフォルト進数は 16 進です。アドレス指定はシンボルや式での指定も可能です (「表 5-6 シンボルでの指定方法」参照)。</p>	

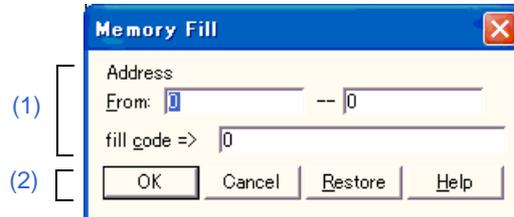
### (2) 機能ボタン

Find Next	<p>設定に従い検索します。          検索の結果、一致する文字列は反転表示されます。          連続して検索する場合には、再度このボタンをクリックします。</p>
Set Find	指定された条件を検索条件として設定し、ダイアログをクローズします。
Stop (検索中)	検索を中断します。
Close	このダイアログをクローズします。(検索中は、このボタンが <Stop> ボタンに変化しています。)
Help	このダイアログのオンライン・ヘルプを表示します。

## メモリ・フィル・ダイアログ

メモリ・ウィンドウのメモリ内容を指定データ (fill code) で埋めます (「5.7 メモリ操作機能」参照)。

図 6 - 36 メモリ・フィル・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

[編集]メニュー → [メモリ] → [初期化...] を選択

### 各エリア説明

#### (1) フィル範囲指定エリア

Address	フィル範囲, およびフィル・コードを指定するエリアです。	
From:	フィル範囲 (先頭アドレス -- 終了アドレス) を指定します。数値入力時のデフォルト進数は 16 進です。アドレス指定はシンボルや式での指定も可能です (「表 5 - 6 シンボルでの指定方法」参照)。	
fill code =>	" From: " で指定した範囲を埋める際に用いるデータ (fill code) を指定します。最大 16 個までのバイナリ・データ列 (バイト・データ列) での指定が可能です。個々のデータは " 空白文字 " で区切って指定します。	

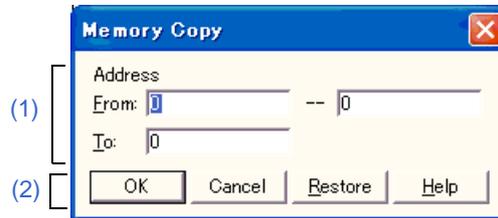
#### (2) 機能ボタン

OK	設定に従い指定データで埋めます。
Stop (処理中)	処理を中断します。
Cancel	このダイアログをクローズします。(処理中は, このボタンが <Stop> ボタンに変化しています。)
Restore	このダイアログをオープンする前の状態に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

## メモリ・コピー・ダイアログ

メモリ・ウィンドウのメモリ内容のコピーを行います（「5.7 メモリ操作機能」参照）。

図 6 - 37 メモリ・コピー・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

[編集]メニュー → [メモリ] → [複写 ...] を選択

### 各エリア説明

#### (1) コピー範囲指定エリア

Address	メモリ内容のコピー元、およびコピー先アドレスを指定するエリアです。数値入力時のデフォルト進数は 16 進です。アドレス指定はシンボルや式での指定も可能です（「表 5 - 6 シンボルでの指定方法」参照）。	
From:	コピー元のアドレス範囲（先頭アドレス -- 終了アドレス）を指定します。	
To:	コピー先の先頭アドレスを指定します。	

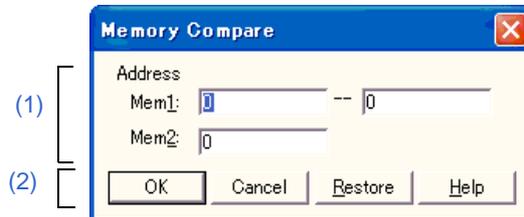
#### (2) 機能ボタン

OK	設定に従い、メモリのコピーを行います。
Stop（コピー中）	コピーを中断します。
Cancel	このダイアログをクローズします。（コピー中は、このボタンが <Stop> ボタンに変化しています。）
Restore	このダイアログをオープンする前の状態に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

## メモリ比較ダイアログ

メモリ・ウィンドウのメモリ内容の比較を行います（「5.7 メモリ操作機能」参照）。

図 6 - 38 メモリ比較ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

[編集]メニュー → [メモリ] → [比較...] を選択

### 各エリア説明

#### (1) 比較範囲指定エリア

Address	メモリ内容の比較元アドレス、および比較先アドレスを指定するエリアです。数値入力時のデフォルト進数は16進です。アドレス指定はシンボルや式での指定も可能です（「表 5 - 6 シンボルでの指定方法」参照）。	
Mem1:	比較元のアドレス範囲（先頭アドレス -- 終了アドレス）を指定します。	
Mem2:	比較先の先頭アドレスを指定します。	

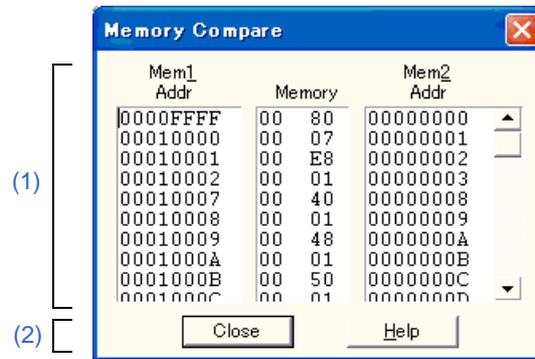
#### (2) 機能ボタン

OK	設定に従い、メモリの比較を行います。メモリを比較した結果、相違がなかった場合には、「Wf200: 違いはありませんでした。」のメッセージが表示されます。相違があった場合には、メモリ比較結果ダイアログがオープンします。
Stop（比較中）	比較を中断します。
Cancel	このダイアログをクローズします。（比較中は、このボタンが <Stop> ボタンに変化しています。）
Restore	このダイアログをオープンする前の状態に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

## メモリ比較結果ダイアログ

メモリ比較ダイアログにてメモリ比較を行った結果、メモリ内容に相違があったときに表示されるダイアログです（「5.7 メモリ操作機能」参照）。

図6-39 メモリ比較結果ダイアログ



- 各エリア説明

### 各エリア説明

#### (1) 比較結果表示エリア

メモリ比較結果を表示するエリアです。比較結果に違いのあった部分のみが表示されます。

Mem1 Addr	違いのあった比較元アドレスを表示しています。
Memory	違いのあったデータ内容を表示しています（左：比較元データ、右：比較先データ）。
Mem2 Addr	違いのあった比較先アドレスを表示しています。

#### (2) 機能ボタン

Close	このダイアログをクローズします。
Help	このダイアログのオンライン・ヘルプを表示します。

## DMM ダイアログ

DMM (Dynamic Memory Modification) を行うアドレスとデータを設定するダイアログです（「5.14 DMM 機能」参照）。

ユーザ・プログラム実行中に DMM 機能による書き換えを行います。

**注意** この機能はソフトウェア・エミュレーションによって実現されているため、書き換えの際、ユーザ・プログラムの実行が一瞬ブレイクします。（疑似 DMM 機能）。

図 6 - 40 DMM ダイアログ（例：Memory 選択時）



- オープン方法
- 各エリア説明

### オープン方法

[編集]メニュー→[DMM...]を選択、またはメモリ・ウィンドウ/レジスタ・ウィンドウ/ IOR ウィンドウ上の <DMM...> ボタンをクリック

### 各エリア説明

#### (1) DMM 対象選択エリア

DMM を行う対象を選択します。このオプション・ボタンの選択により、(2) DMM 設定エリアの表示が切り替わります。

Memory	メモリの DMM が可能
Register	レジスタの DMM が可能
IOR	IOR の DMM が可能

**参考** メモリ・ウィンドウ、レジスタ・ウィンドウ、IOR ウィンドウからオープンした場合は、対応するオプション・ボタンが選択された状態でこのダイアログがオープンされます。

**(2) DMM 設定エリア**

## (a) Memory 選択時

Memory Address:	データを書き込むメモリ・アドレスを指定するエリアです。数値入力時のデフォルト進数は 16 進です。アドレス指定はシンボルや式での指定も可能です（「表 5-6 シンボルでの指定方法」参照）。	
Write Data:	"Memory Address:" で指定したメモリ・アドレスに書き込むデータを指定するエリアです。	
Data Size:	"Write Data:" で指定したデータを書き込む際のデータ・サイズを選択するエリアです。	
	Byte:	8 ビット・データとして書き込む
	Half Word:	16 ビット・データとして書き込む
	Word:	32 ビット・データとして書き込む

## (b) Register 選択時

Register Name:	データを書き込むレジスタ名を指定するエリアです。大文字／小文字は区別しません。機能名称、および絶対名称での指定が可能です。
Write Data:	"Register Name:" で指定したレジスタに書き込むデータを指定するエリアです。

## (c) IOR 選択時

IOR Name:	データを書き込む IOR 名を指定するエリアです。大文字／小文字は区別しません。読み込み専用の IOR は変更できません。
Write Data:	"IOR Name:" で指定したレジスタに書き込むデータを指定するエリアです。

**(3) 機能ボタン**

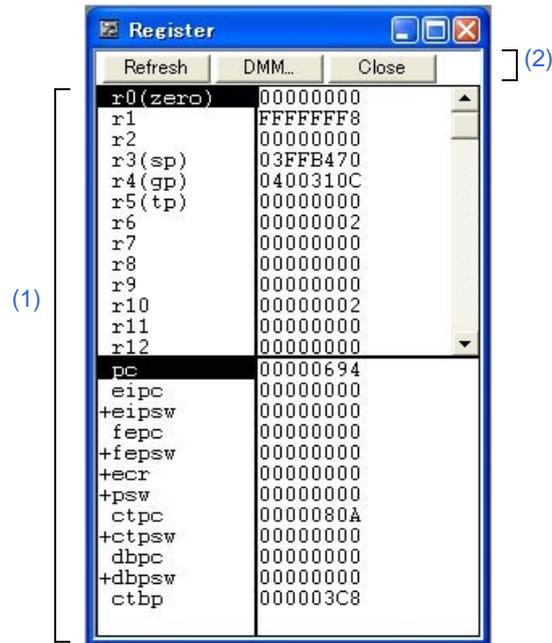
Set	設定に従い、データの書き込みを行います。
Close	このダイアログをクローズします。
Help	このダイアログのオンライン・ヘルプを表示します。

## レジスタ・ウィンドウ

レジスタ（プログラム・レジスタ／システム・レジスタ）の表示、および変更を行います（「5.8 レジスタ操作機能」参照）。

このウィンドウでは、コンテキスト・メニュー、機能ボタン等による操作が可能です。  
各エリアはジャンプ機能のジャンプ・ポインタになります。

図6-41 レジスタ・ウィンドウ



- オープン方法
- 各エリア説明
- [表示]メニュー（レジスタ・ウィンドウ専用部分）
- コンテキスト・メニュー

### オープン方法

 ボタンをクリック、または[ブラウザ]メニュー→[レジスタ]を選択。

## 各エリア説明

### (1) レジスタ値表示／変更エリア

左側にレジスタ名を、右側にレジスタ値を表示しています。

レジスタ値は直接入力により変更可能です。変更箇所は赤色で表示され Enter キーを押すことによりターゲット・メモリに書き込まれます。書き込み前の内容は ESC キーにより破棄できます。

なお、上段には、プログラム・レジスタを、下段には、システム・レジスタを表示しています。

プログラム・レジスタ表示／変更エリア	プログラム・レジスタの表示と値の変更を行うエリアです。
システム・レジスタ表示／変更エリア	システム・レジスタの表示と値の変更を行うエリアです。 "+" は展開表示可能なレジスタを示し、ダブルクリックすることによりフラグ名とフラグ値に展開表示されます ("+" → "-")。再度ダブルクリックすることにより展開表示が解除されます。

**注意** 不正な値 (0xFFFFFFFF を越える値) を入力したことによりレジスタのオーバーフローが生じた場合には、レジスタの値は、0xFFFFFFFF に変更されます。

### (2) 機能ボタン

Refresh	ウィンドウの内容を最新のデータに更新します。
DMM...	DMM ダイアログをオープンします。
Close	このウィンドウをクローズします。

## [表示]メニュー (レジスタ・ウィンドウ専用部分)

レジスタ・ウィンドウのアクティブ時には、共通の [表示] メニューに以下の項目が追加されます。

2 進	2 進数表示
8 進	8 進数表示
10 進	10 進数表示
16 進	16 進数表示
ピックアップ	レジスタ選択ダイアログで選択したレジスタをピックアップ表示します。
表示選択 ...	レジスタ選択ダイアログをオープンします。

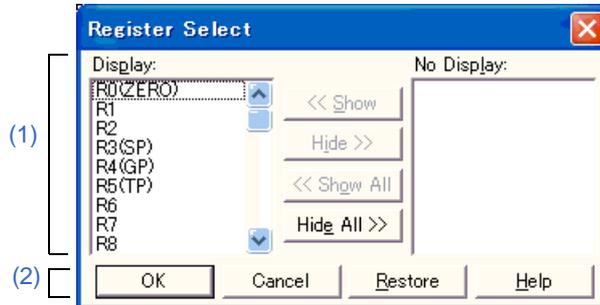
## コンテキスト・メニュー

ウォッチ登録 ...	選択文字列をウォッチ・ウィンドウに追加します。 <a href="#">ウォッチ登録ダイアログ</a> をオープンします。
2進	2進数表示
8進	8進数表示
10進	10進数表示
16進	16進数表示
ピックアップ	<a href="#">レジスタ選択ダイアログ</a> で選択したレジスタをピックアップ表示します。
表示選択 ...	<a href="#">レジスタ選択ダイアログ</a> をオープンします。

## レジスタ選択ダイアログ

レジスタ・ウィンドウに表示しないレジスタを選択します（「5.8 レジスタ操作機能」参照）。

図 6 - 42 レジスタ選択ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

レジスタ・ウィンドウをカレント・ウィンドウにし、[表示]メニュー→[表示選択...]を選択

### 各エリア説明

#### (1) 表示レジスタ選択エリア

Display:	レジスタ・ウィンドウに表示されているレジスタのリストです。	
ボタン	表示レジスタを変更する際、次のボタンをクリックします。 Ctrl キーや Shift キーを押しながらクリックすることで、複数レジスタの選択が可能です。	
	<< Show	" No Display: " のリストから選択されたレジスタを " Display: " へ移動
	Hide >>	" Display: " のリストから選択されたレジスタを " No Display: " へ移動
	<< Show All	すべてのレジスタを " Display: " へ移動
	Hide All >>	すべてのレジスタを " No Display: " へ移動
No Display:	レジスタ・ウィンドウに表示されていないレジスタのリストです。	

#### (2) 機能ボタン

OK	レジスタ・ウィンドウに、選択結果を反映しこのダイアログをクローズします。
Cancel	変更を取り消してこのダイアログをクローズします。
Restore	このダイアログをオープンする前の状態に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

## IOR ウィンドウ

周辺 I/O レジスタの内容、および I/O ポート追加ダイアログで登録した I/O ポート値の表示、変更を行うダイアログです（「5.8 レジスタ操作機能」参照）。

このウィンドウでは、[コンテキスト・メニュー](#)、[機能ボタン](#)等による操作が可能です。

**注意 1** 読み込み専用の周辺 I/O レジスタ、および I/O ポートの値は変更することができません。また、読み込み動作によってデバイスが動作してしまう周辺 I/O レジスタ、および I/O ポートは、読み込み保護の対象となり読み込みを行いません。このようなレジスタを読み込みたい場合には、該当レジスタを選択し、[コンテキスト・メニュー](#)から [強制読み込み] を実行してください。

**注意 2** ユーザ・プログラム実行中は、[拡張オプション設定ダイアログ](#)で設定したサンプリング時間毎に IOR の内容を更新します（「5.13 RRM 機能」参照）。ただし RAM モニタ機能が OFF の場合は更新しません。  
[IECUBE]

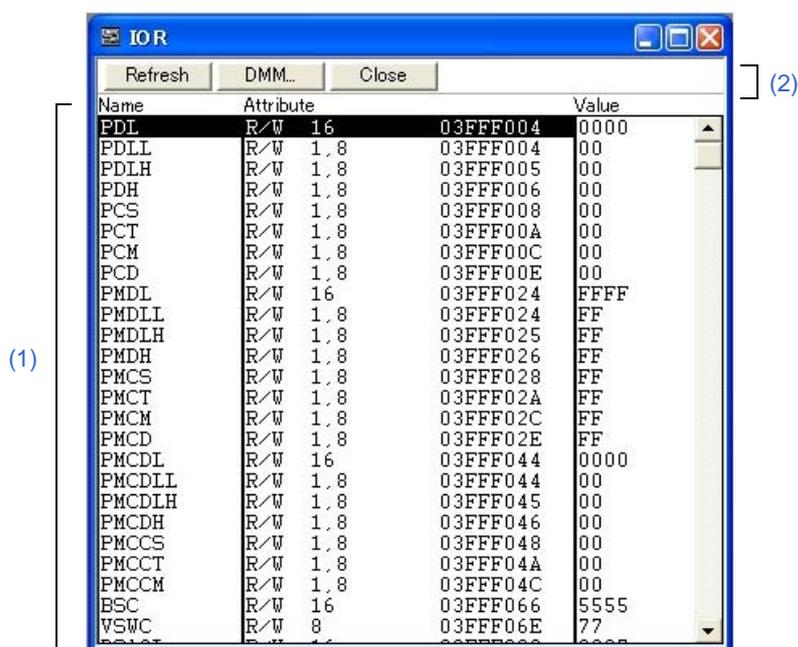
**参考 1** ウィンドウのオープン時の表示開始位置は、以下のようになります。

初回：最小アドレスの周辺 I/O レジスタから表示

2 回目以降：最後にウィンドウを閉じた際の先頭周辺 I/O レジスタから表示

**参考 2** プログラマブル I/O レジスタをサポートしている品種で、[コンフィギュレーション・ダイアログ](#)でプログラマブル I/O 領域を設定している場合、プログラマブル I/O レジスタ、および拡張周辺 I/O レジスタの表示も行います。I/O ポートアドレスが不定値となった場合、I/O ポート名は淡色表示となります。

図 6 - 43 IOR ウィンドウ



- オープン方法
- 各エリア説明
- [表示]メニュー (IOR ウィンドウ専用部分)
- コンテキスト・メニュー

## オープン方法



ボタンをクリック, または [ブラウザ] メニュー → [I/O レジスタ] を選択

## 各エリア説明

### (1) IOR 表示/変更エリア

Name	周辺 I/O レジスタ名や I/O ポート名を表示するエリアです。 I/O ポート・アドレスが不定値となった場合には淡色表示されます。	
Attribute	周辺 I/O レジスタ, I/O ポートの属性を表示するエリアです。 左から, リード/ライト属性, アクセス・タイプ, 絶対アドレスを示しています。 ビット周辺 I/O レジスタの場合には, ビット・オフセットも表示します。 [表示]メニュー→[属性]にて, このエリアの表示/非表示が選択できます。	
	リード/ライト属性	
	R	読み出しのみ可能
	W	書き込みのみ可能
	R/W	読み出し/書き込み可能
	*	読み込み動作によって, デバイスが動作してしまうのを防ぐために, エミュレーション・レジスタから読み出しを行うレジスタ。 直接周辺 I/O レジスタから読み出すには, [表示]メニュー→[強制読み込み]を実行してください。 また, 書き込み専用の周辺 I/O レジスタであってもエミュレーション・レジスタを介して, 値を読み出すことができます。 この機能はデバイスによりサポートされない場合があります。
	アクセス・タイプ	
	1	Bit アクセス可能
	8	Byte アクセス可能
	16	Half Word アクセス可能
	32	Word アクセス可能

Value	<p>周辺 I/O レジスタ、および I/O ポートの内容表示と変更を行うエリアです。値の変更は直接入力により行います。変更箇所は<b>赤色</b>で表示され Enter キーを押すことにより、ターゲット・メモリに書き込まれます。書き込み前の内容は ESC キーにより破棄できます。</p> <p>なお、読み出し専用の周辺 I/O レジスタや I/O ポートの場合、値を変更することはできません。</p> <p>読み込み保護された周辺 I/O レジスタ、および I/O ポートは、<a href="#">コンテキスト・メニュー</a> → [強制読み込み] を選択することで値の読み込みが可能です。</p> <p>属性によって、表示形式が次のように異なります。</p>	
	黒色表示	読み出し専用、または読み出し／書き込み可能
	--	書き込み専用
	**	読み出すと値が変化する周辺 I/O レジスタ、I/O ポート

## (2) 機能ボタン

Refresh	ウィンドウの内容を最新のデータに更新します。
DMM...	<a href="#">DMM ダイアログ</a> をオープンします。
Close	このウィンドウをクローズします。

## [表示] メニュー (IOR ウィンドウ専用部分)

IOR ウィンドウのアクティブ時には、共通の [\[表示\] メニュー](#) に以下の項目が追加されます。

2 進	2 進数表示
8 進	8 進数表示
10 進	10 進数表示
16 進	16 進数表示 (デフォルト)
名前順	アルファベット順に表示
アドレス順	アドレス順に表示 (デフォルト)
ソートしない	並べ替えをしない
属性	"Attribute" の表示／非表示を切り替えます。
ピックアップ	<a href="#">IOR 選択ダイアログ</a> で選択したレジスタをピックアップ表示します。
表示選択 ...	<a href="#">IOR 選択ダイアログ</a> をオープンします。
強制読み込み	読み込むと値が変化するために読み込み禁止にしている周辺 I/O レジスタ、または <a href="#">I/O ポート追加ダイアログ</a> で追加した I/O ポート、I/O プロテクト領域のデータを強制的に読み込みます。

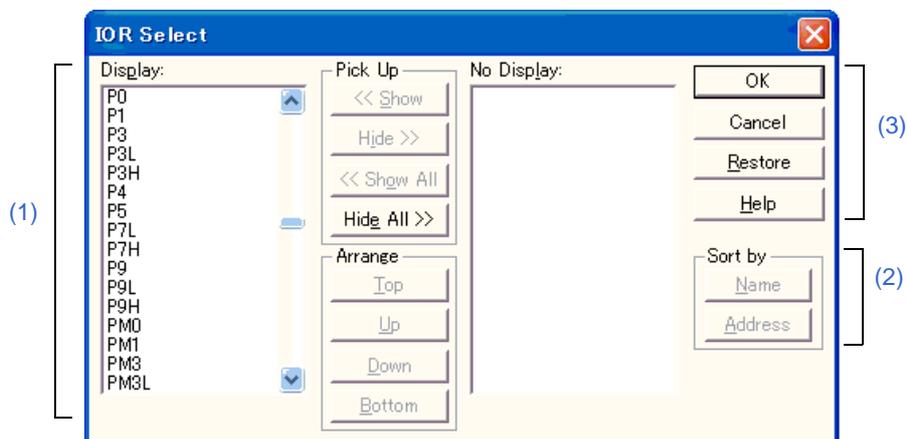
## コンテキスト・メニュー

移動 ...	アドレス指定ダイアログをオープンします。
ウォッチ登録 ...	ウォッチ登録ダイアログをオープンします。
I/O ポート追加 ...	I/O ポート追加ダイアログをオープンします。
2 進	2 進数表示
8 進	8 進数表示
10 進	10 進数表示
16 進	16 進数表示 (デフォルト)
名前順	アルファベット順に表示
アドレス順	アドレス順に表示 (デフォルト)
ソートしない	並べ替えをしない
属性	"Attribute" の表示/非表示を切り替えます。
ピックアップ	IOR 選択ダイアログで選択したレジスタをピックアップ表示します。
表示選択 ...	IOR 選択ダイアログをオープンします。
強制読み込み	読み込むと値が変化するために読み込み禁止にしている周辺 I/O レジスタ, または I/O ポート追加ダイアログで追加した I/O ポート, I/O プロテクト領域のデータを強制的に読み込みます。

## IOR 選択ダイアログ

IOR ウィンドウに表示しない周辺 I/O レジスタ, および I/O ポートを選択します(「5.8 レジスタ操作機能」参照)。また, 表示順序の指定を行います。

図 6 - 44 周辺 I/O レジスタ選択ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

IOR ウィンドウをカレント・ウィンドウにし, [表示]メニュー→[表示選択...]を選択

## 各エリア説明

### (1) 表示周辺 I/O レジスタ選択, 表示順序変更エリア

Display:	IOR ウィンドウに表示されているレジスタのリストです。Ctrl キーや Shift キーを押しながらクリックすることで、複数個の選択が可能です。	
Pick Up	表示周辺 I/O レジスタを変更する際、次のボタンをクリックします。	
	<< Show	" No Display: " のリストから選択されたレジスタを " Display: " へ移動
	Hide >>	" Display: " のリストから選択されたレジスタを " No Display: " へ移動
	<< Show All	すべてのレジスタを " Display: " へ移動
	Hide All >>	すべてのレジスタを " No Display: " へ移動
Arrange	" Display: " リスト内の表示順序を変更する際、次のボタンをクリックします。表示順序の変更は複数個同時には行えません。1 つずつ選択して行ってください。	
	Top	選択されたレジスタをリストの最上行に移動
	Up	選択されたレジスタを 1 行上に移動
	Down	選択されたレジスタを 1 行下に移動
	Bottom	選択されたレジスタをリストの最下行に移動
No Display:	IOR ウィンドウに表示されていないレジスタのリストです。	

### (2) 表示順序変更ボタン

Sort by	" No Display: " リスト内の表示順序を変更する際、次のボタンをクリックします。	
	Name	アルファベット順に表示
	Address	アドレス順に表示

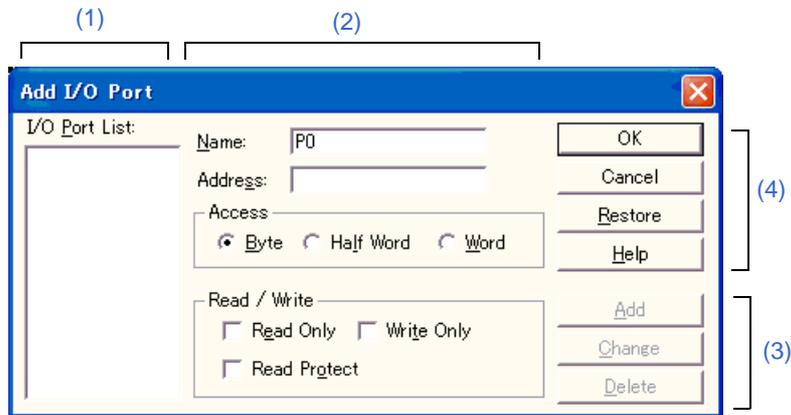
### (3) 機能ボタン

OK	IOR ウィンドウに選択結果を反映しこのダイアログをクローズします。
Cancel	変更を取り消してこのダイアログをクローズします。
Restore	このダイアログをオープンする前の状態に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

## I/O ポート追加ダイアログ

I/O ウィンドウに追加する I/O ポートの登録を行います（「5.8 レジスタ操作機能」参照）。

図 6 - 45 I/O ポート追加ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

[オプション]メニュー→[I/O ポート追加 ...]を選択

### 各エリア説明

#### (1) I/O Port List: (I/O ポート表示エリア)

現在登録されている I/O ポートの一覧を表示するエリアです。

新規登録によりこのリストに追加されます。登録済みの I/O ポートは (3) 追加, 削除ボタンにより, 変更/削除が可能です。

## (2) I/O ポート指定エリア

Name:	追加する I/O ポート名を入力するエリアです (最大 15 文字)。	
Address:	追加する I/O ポート・アドレスを指定するエリアです。 数値入力時のデフォルト進数は 16 進です。シンボルでの指定も可能です (「表 5-6 シンボルでの指定方法」参照)。 このエリアに指定可能なアドレスは、Target 領域、周辺 I/O レジスタ領域のいずれかです。	
Access	追加する I/O ポートのアクセス・サイズを次の中から選択します。	
	Byte	8 ビット (デフォルト)
	Half Word	16 ビット
	Word	32 ビット
Read / Write	追加する I/O ポートのアクセス属性を指定します。 デフォルトでは、すべて非チェック状態 (読み出し/書き込み共に許可) です。	
	Read Only	読み出し専用
	Write Only	書き込み専用
	Read Protect	読み出し保護

## (3) 追加, 削除ボタン

Add	指定したアドレスの I/O ポートを追加します。
Change	(1) I/O Port List: (I/O ポート表示エリア) で選択している I/O ポートの設定を変更します。
Delete	(1) I/O Port List: (I/O ポート表示エリア) で選択している I/O ポートを削除します。

## (4) 機能ボタン

OK	IOR ウィンドウに追加結果を反映し、このダイアログをクローズします。
Cancel	変更を取り消して、このダイアログをクローズします。
Restore	変更を元に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

## タイマ・ダイアログ

[IECUBE]

タイマ・イベント条件の登録，設定，および実行時間測定結果の表示を行います（「5.12 イベント機能」，「5.9 タイマ機能 [IECUBE]」参照）。

<View Always> ボタンをクリックすることにより，実行時間表示エリアをタイマ測定結果ダイアログとして常時表示させておくことが可能です。

タイマ・イベント条件の登録，設定は，このダイアログ上で各項目を設定し，<OK> ボタンをクリックすることにより行います（最大 256 個）。登録したタイマ・イベント条件は，イベント・マネージャで管理されます。

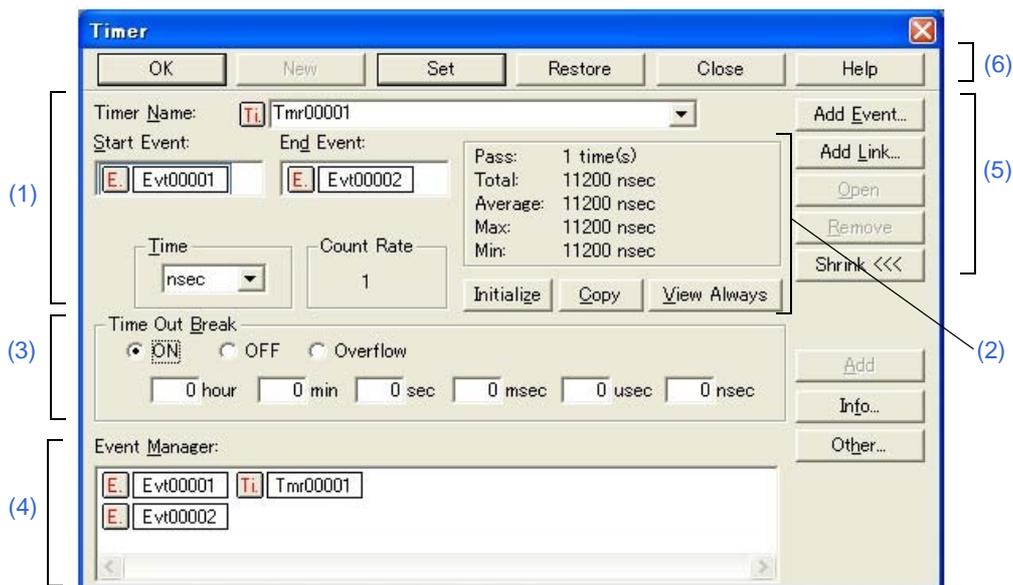
なお，同時に設定可能な（有効になる）タイマ・イベント条件数には制限があります（「5.12.4 各種イベント条件の有効イベント数」参照）。

実行時間測定結果の表示は，設定したタイマ・イベント条件を選択することにより表示されます。

**参考 1** 測定結果の表示内容は，ユーザ・プログラム実行中でも RRM 機能のサンプリング時間ごとに更新されます。

**参考 2** ユーザ・プログラム実行中でも，タイマ・イベント条件の設定／有効化／無効化／削除操作が可能です。

図 6 - 46 タイマ・ダイアログ



- オープン方法
- 各エリア説明
- 注意事項

### オープン方法



ボタンをクリック，または [ イベント ] メニュー → [ タイマ ... ] を選択

## 各エリア説明

### (1) タイマ・イベント条件設定エリア

Timer Name:	<p>タイマ・イベント名の設定を行います。 名前は最大8文字までの英数字を直接入力してください。 すでに登録済みのイベント条件の内容を表示するときは、ドロップダウン・リストから選択します。 ユーザ・プログラムの実行からブレイクまでを表示させたい場合には、"Run-Break"を指定してください（「5.9.2 Run-Break イベント」参照）。 このエリアの左側のマークは、イベント条件の使用状況を示します（「表5-18 イベント・アイコン」参照）。また、灰色のマークは、そのイベント条件が編集中で未登録の状態を示しています。 このマークをクリックすることにより、イベント条件の有効／無効を切り替えることができます。</p>
Start Event: End Event:	<p>時間測定開始条件用、停止条件用のイベント条件を設定します（区間測定）。 このエリアに登録できるイベント条件、およびイベント・リンク条件の個数は、開始条件、停止条件ともに1個です。 イベント条件の設定は、イベント・マネージャ・エリアから、設定するイベント・アイコンをドラッグして、このエリア上にドロップすることにより簡単に行えます。詳細は、「5.12.3 各種イベント条件の設定」を参照してください。</p>
Time	(2) 実行時間表示エリアの表示単位をドロップダウン・リストから選択します。
Count Rate	<p>タイマ・カウント・レート値（拡張オプション設定ダイアログの(2) Timer（タイマ関連設定エリア）[IECUBE]での設定値）を表示します（既存のタイマ・イベント条件の内容を表示している場合は、イベント作成時点でのタイマ・カウント・レート値になります。イベントの新規作成時、またはRun-Break イベント条件の内容を表示している場合には、現在のタイマ・カウント・レート値になります）。 このエリアは編集できません。 既存のタイマ・イベント条件のタイマ・カウント・レート値を変更したい場合には、拡張オプション設定ダイアログの(2) Timer（タイマ関連設定エリア）[IECUBE]の値を変更した後、タイマ・イベント条件を再設定する必要があります。また、イベント作成時点でのタイマ・カウント・レート値と拡張オプション設定ダイアログの(2) Timer（タイマ関連設定エリア）[IECUBE]での設定値が異なる場合、タイムアウト時間に影響するため、このエリアの値が赤色で表示されます（「注意事項」参照）。</p>

### (2) 実行時間表示エリア

ユーザ・プログラムの実行により、測定された結果を表示します。

Run-Break の場合は、"Total" のみが表示されます。

なお、カウンタのオーバーフローにより、測定結果が信頼できない項目に関しては、赤色で表示されます。

Pass:	通過回数
Total:	開始イベントと終了イベント条件で指定した測定区間の累積実行時間
Average:	平均実行時間
Max:	最大実行時間
Min:	最小実行時間
<Initialize>	測定結果をクリアします。
<Copy>	測定結果をクリップ・ボードにコピーします。
<View Always>	タイマ測定結果ダイアログをオープンします。

表 6 - 14 測定可能値

接続 IE	測定可能実行時間	測定可能実行回数
[IECUBE]	(Total, Max, Min は 33bit, 周波数は 50 MHz) 最大 2.8 分 (1 分周時/分解能 20 n 秒) 最大 195.2 時間 (4K 分周時/分解能 81920 n 秒)	16bit 最大 65535 回

**(3) Time Out Break (タイムアウト・ブレイク設定エリア)**

" Start Event:, End Event: " での区間測定時間 (タイマ・スタート・イベントの成立から, タイマ・エンド・イベントの成立の間の時間) に対するタイムアウト・ブレイクの設定を行います。

ON	区間測定時間がタイムアウト時間を越えた場合, タイムアウト・ブレイク (実行を終了) します。 下欄にタイムアウト時間を指定します。計測可能な最大時間までの値を指定可能です。
OFF	タイムアウト・ブレイクしません (デフォルト)。
Overflow	区間測定時間が測定可能な最大時間 (「表 6 - 14 測定可能値」参照) を越えた場合, タイムアウト・ブレイク (実行を終了) します。

**注意** Run-Break イベントの場合, "OFF" 固定となり, このエリアは無効です。

**(4) イベント・マネージャ・エリア**

登録されているイベント一覧を表示しています。

このエリア上のイベント・アイコンを各種イベント設定ダイアログのイベント設定エリアにドラッグ&ドロップすることにより, 容易に各種イベント条件の設定が行えます (「5.12.3 各種イベント条件の設定」参照)。

このエリアは, 各種イベント関連ダイアログで共通です (「ブレイク・ダイアログ」の「(2) イベント・マネージャ・エリア」参照)。

**(5) 機能ボタン (イベント条件の内容表示等)**

イベント条件設定エリアに置かれたイベント条件の内容表示/削除, およびイベント・マネージャ・エリアの表示/非表示等の操作を行います。

このエリアは, 各種イベント関連ダイアログで共通です (「ブレイク・ダイアログ」の「(3) 機能ボタン (イベント条件の内容表示等)」参照)。

**(6) 機能ボタン (イベント条件の登録/削除/有効化/無効化)**

イベントの登録/削除, および有効化/無効化等の操作を行うボタンです。指定したイベント条件のイベントは, <OK> (または <Set>) ボタンを押すことにより登録, 設定 (有効化) されます。

このエリアは, 各種イベント関連ダイアログで共通です。(「ブレイク・ダイアログ」の「(4) 機能ボタン (イベント条件の登録/削除/有効化/無効化)」参照)。

## 注意事項

IECUBE は、全タイマ・イベントでタイマ・カウント・レート値が共通であるため、イベント作成時のタイマ・カウント・レート ("Count Rate" の表示値) と現在のタイマ・カウント・レート値 (拡張オプション設定ダイアログの (2) Timer (タイマ関連設定エリア) [IECUBE] の設定) が異なる場合、(3) Time Out Break (タイムアウト・ブレイク設定エリア) で指定した時間がずれてしまいます。

### 例)

イベント作成時のタイマ・カウント・レート値が "16"、タイムアウト時間に "1sec" を指定したタイマ・イベントの場合

- (1) 現在のタイマ・カウント・レート値が "32" の時  
→タイマ・カウント・レート値が 2 倍となり、2 秒 で Timer Over Break が発生
- (2) 現在のタイマ・カウント・レート値が "4" の時  
→タイマ・カウント・レート値が 1/4 となり、250 ミリ秒 で Timer Over Break が発生

このため、上記のような状態で、タイムアウト・ブレイクの設定を行わないようにしてください (タイムアウト・ブレイク設定が "OFF"、または "Overflow" の場合は影響ありません)。

タイムアウト・ブレイクの設定を行う場合には、拡張オプション設定ダイアログの (2) Timer (タイマ関連設定エリア) [IECUBE] で、タイマ・カウント・レート値を変更し、タイマ・イベントを再設定してください。

## タイマ測定結果ダイアログ

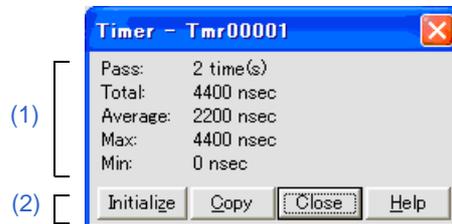
[IECUBE]

実行時間測定結果の表示を行います（「5.9 タイマ機能 [IECUBE]」参照）。

タイマ・ダイアログの <View Always> ボタンをクリックすることにより、タイマ・イベント条件 1 つに対して 1 つオープンされます（複数オープン可能）。同時にオープンできるのは 256 + 1（Run-Break イベント）個ですが、同時に測定できる数は「5.12.4 各種イベント条件の有効イベント数」+ 1（Run-Break イベント）個までとなります。

**参考** 測定結果の表示内容は、ユーザ・プログラム実行中でも RRM 機能のサンプリング時間ごとに更新されます。

図 6 - 47 タイマ測定結果ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

タイマ・ダイアログ上で、タイマ・イベント条件を選択し <View Always> ボタンをクリック

### 各エリア説明

#### (1) 実行時間表示エリア

タイマ・ダイアログと同様です。

Pass:	通過回数
Total:	開始イベントと終了イベント条件で指定した測定区間の累積実行時間
Average:	平均実行時間
Max:	最大実行時間
Min:	最小実行時間

## (2) 機能ボタン

Initialize	測定結果をクリアします。
Copy	測定結果をクリップ・ボードにテキスト形式でコピーします。
Close	このダイアログをクローズします。
Help	このダイアログのオンライン・ヘルプを表示します。

## トレース・ウィンドウ

[IECUBE]

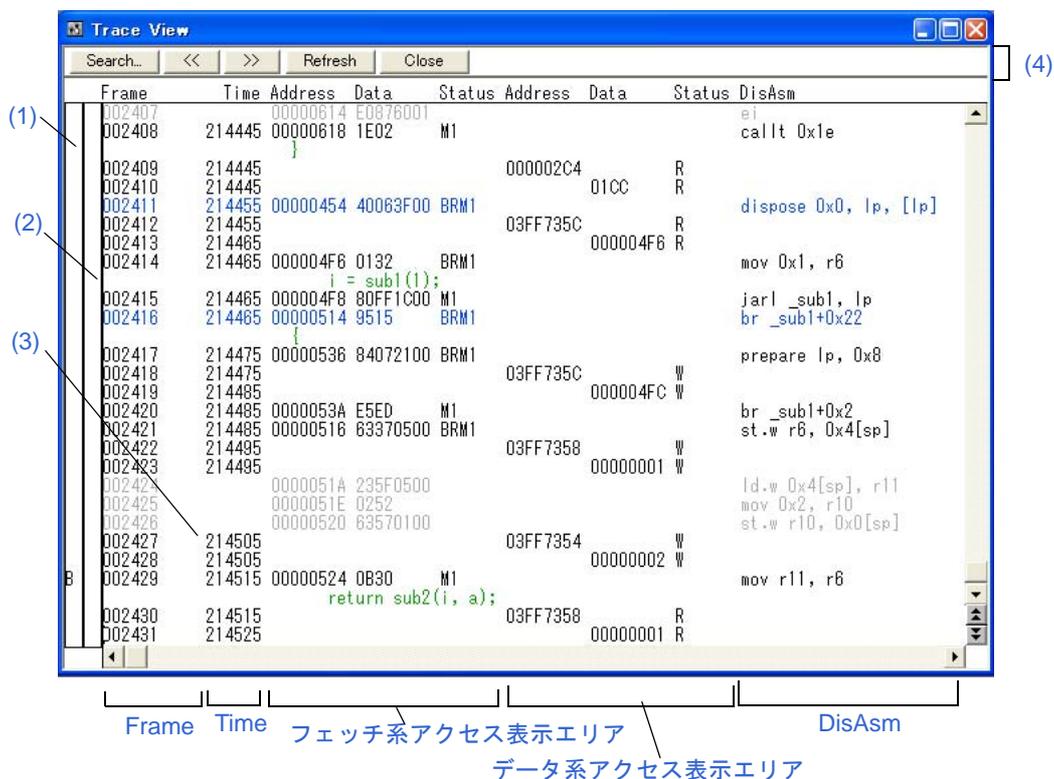
トレース結果を表示します（「5.10 トレース機能 [IECUBE]」参照）。

表示更新は、ブレーク時、またはステップ実行時に行われます。

このウィンドには、トレース結果と混合させたソース・ファイルの表示を行う混合表示モード（トレース・ウィンドウ）があります。また、他ウィンドウとの連結機能を持ちます（「5.16.3 トレース結果とウィンドウの連結機能 [IECUBE]」参照）。

コンテキスト・メニュー、機能ボタン等によるさまざまな操作が可能です。

図 6 - 48 トレース・ウィンドウ



- オープン方法
- 各エリア説明
- [表示]メニュー（トレース・ウィンドウ専用部分）
- コンテキスト・メニュー

### オープン方法

 ボタンをクリック、または[ブラウザ]メニュー→[トレース]を選択

## 各エリア説明

### (1) ポイント・マーク表示エリア

イベント設定状態を表示するエリアです。

対応するトレース・アドレスに、実行イベント、またはアクセス・イベントが設定されている場合、イベントの種類に対応するマークが表示されます。

なお、表示されるマークはトレース時のものではなくトレース結果表示時に設定されているイベントです。

### (2) トレーサ・モード表示エリア

トレーサ・モードの種類を表示するエリアです。

A	セクション・トレースとクオリファイ・トレースの開始と終了のフレーム
T	ディレイ・トリガ・フレーム
M	DMA ポイント・アクセス・フレーム (DMA の開始点、および終了点) (「5.10.7 DMA ポイント・トレース機能」参照)
N	データの取りこぼしが発生したフレーム

### (3) トレース結果表示エリア

トレース結果を表示するエリアです。

補完フレームは淡色で表示されます。なお、補完フレームでは、"Frame", "フェッチ系アクセス表示エリア", "DisAsm" のみが表示されます。

各エリアの表示／非表示は、[トレース表示選択ダイアログ](#)で選択できます。

Frame	トレース・フレーム番号を表示するエリアです。
Time	プログラムの実行開始から、各フレームの命令実行、またはメモリ・アクセスの要因が発生するまでにターゲット・チップが何クロックを要したかを表示するエリアです。 補完フレームでは表示されません。 表示内容は、 <a href="#">トレース表示選択ダイアログ</a> トレース表示選択ダイアログでクロック数表示／時間表示に切り替え可能です。 タイマ・カウンタの分周率と最大測定時間の関係は、「 <a href="#">表 6-5 分周率と最大測定時間の関係 (タイムタグ・カウンタ (トレース))</a> 」を参照してください。また、タイムタグと時間測定は、外部クロック (50MHz) で測定します。 <b>注意：</b> オーバーフローした場合は、タイムタグ最大値を赤色で表示します。
フェッチ系アクセス表示 エリア	プログラムのフェッチ結果を表示するエリアです。 補完フレームでは "Address", "Data" のみが表示されます。

Address	フェッチ・アドレスを表示	
Data	フェッチ・データを表示	
Status	ステータスの種類には次のものがあります。	
	BRM1	分岐後の最初の命令の1バイト目のフェッチ (BRM1 かつ M1 のフレームを含む) フェッチ・アドレスがシンボルの先頭の場合は、1行を青色で強調表示にします。
	M1	命令の1バイト目のフェッチ
	空白	次のいずれか - データ・アクセスのフレーム (最大3フレーム (RRM 機能選択時は最大2フレーム)) - 2命令同時実行の2命令目のフレーム (常に2フレーム) - 無効フレーム
	IF	割り込みが入ってキャンセルされたフェッチ・アドレス (ブレーク・ポイントを設定したアドレスも該当)
INFO	ディレイ・トリガ・トレース開始、セクション・トレース開始/終了、クオリファイ・トレース開始/終了時のフェッチ・アドレス	
データ系アクセス表示エリア	データのアクセス結果を表示するエリアです。 補完フレームでは表示されません。	
Address	アクセス・アドレスを表示	
Data	アクセス・データを表示	
Status	アクセス・ステータスを表示	
	R	データ・リード
	W	データ・ライト
DisAsm	逆アセンブル結果を表示するエリアです ("Status" が、BRM1, M1, または補完フレームのときのみ表示)。 アクセス・アドレスしか表示しないフレームにおいては、グローバル・シンボルを表示します。	

**参考 1** 6バイト長、8バイト長の命令コードを表示する場合は、先頭4バイトは1フレーム目に、残りバイトは2フレーム目、3フレーム目に表示されます。また、2命令が同時に実行された場合は、1フレーム表示が2行になり、下位アドレス側の命令コードが1行目に、上位アドレス側の命令コードが2行目に表示されます。

**参考 2** 2命令同時実行時の命令コードを表示する場合、一方の命令を1行目に表示し、同時に実行された命令を2行目に表示します。

**(4) 機能ボタン**

Search...	トレース・サーチ・ダイアログをオープンして、トレース結果の検索を行います。検索結果は、トレース・ウィンドウ上で反転表示されます。 [表示]メニュー→[検索...]選択時と同様の動作です。
<<	トレース・サーチ・ダイアログで設定した検索条件に一致するトレース結果を、カーソル位置のフレームから前方向（表示の上）に検索します。
>>	トレース・サーチ・ダイアログで設定した検索条件に一致するトレース結果を、カーソル位置のフレームから後ろ方向（表示の下）に検索します。
Refresh	ウィンドウの内容を最新のデータに更新します。
Close	このウィンドウをクローズします。

**[表示]メニュー（トレース・ウィンドウ専用部分）**

トレース・ウィンドウのアクティブ時には、共通の [表示]メニューに以下の項目が追加されます。

表示選択 ...	表示内容の選択を行います。 トレース表示選択ダイアログをオープンします。
混合表示	ソース表示の混合表示／非表示を切り替えます。 チェックあり：混合表示 チェックなし：非表示（デフォルト）
ウィンドウ連結	トレース・ウィンドウと次のウィンドウの連結を設定します（「5.16.3 トレース結果とウィンドウの連結機能 [IECUBE]」参照）。 チェックのあるウィンドウと連動します。
ソース	ソース・テキスト・ウィンドウと連結
アセンブル	逆アセンブル・ウィンドウと連結
メモリ	メモリ・ウィンドウと連結

**コンテキスト・メニュー**

移動 ...	表示位置を移動します。 フレーム指定ダイアログをオープンします。
トレース・クリア	トレース・データをクリアします。
表示選択 ...	表示内容の選択を行います。 トレース表示選択ダイアログをオープンします。
混合表示	ソース表示の混合表示／非表示を切り替えます。 チェックあり：混合表示 チェックなし：非表示（デフォルト）
ウィンドウ連結	トレース・ウィンドウと次のウィンドウの連結を設定します（「5.16.3 トレース結果とウィンドウの連結機能 [IECUBE]」参照）。
ソース	ソース・テキスト・ウィンドウと連結
アセンブル	逆アセンブル・ウィンドウと連結
メモリ	メモリ・ウィンドウと連結

ソース	<p>カーソル位置のデータ値をジャンプ先アドレスとして、該当するソース・テキストおよびソース行を表示します（「5.16.2 ジャンプ機能」参照）。</p> <p>ただし、ジャンプ先アドレスに行情報がない場合には、ジャンプすることはできません。</p> <p>ソース・テキスト・ウィンドウをオープンします。</p> <p>アクティブ状態のソース・テキスト・ウィンドウが表示されている場合は、そのウィンドウを最前面（操作対象）に表示します。</p>
アセンブル	<p>カーソル位置のデータ値をジャンプ先アドレスとして、そのアドレスから逆アセンブル表示します（「5.16.2 ジャンプ機能」参照）。</p> <p>逆アセンブル・ウィンドウをオープンします。</p> <p>アクティブ状態の逆アセンブル・ウィンドウが表示されている場合は、そのウィンドウを最前面（操作対象）に表示します。</p>
メモリ	<p>カーソル位置のデータ値をジャンプ先アドレスとして、そのアドレスからメモリ内容を表示します（「5.16.2 ジャンプ機能」参照）。</p> <p>メモリ・ウィンドウをオープンします。</p> <p>アクティブ状態のメモリ・ウィンドウが表示されている場合は、そのウィンドウを最前面（操作対象）に表示します。</p>

## トレース・サーチ・ダイアログ

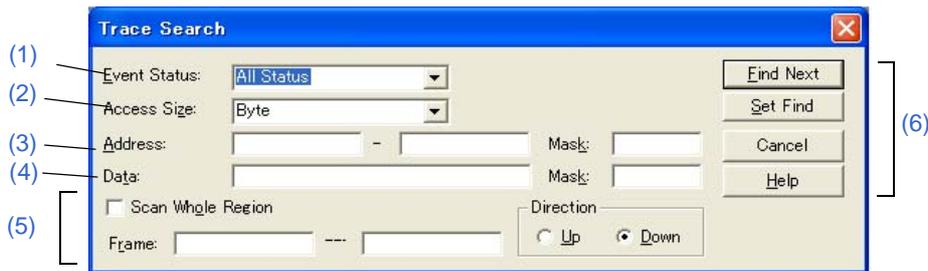
[IECUBE]

トレース・ウィンドウ上のトレース・データの検索を行います（「5.10 トレース機能 [IECUBE]」参照）。

各項目を設定し、<Find Next> ボタンをクリックすることにより、検索を開始します。

また、<Set Find> ボタンをクリックすることで、トレース・ウィンドウ上の方向ボタン（<<, >>）での検索が可能になります。

図 6 - 49 トレース・サーチ・ダイアログ



- [オープン方法](#)
- [各エリア説明](#)

### オープン方法

トレース・ウィンドウをカレント・ウィンドウにし、[表示]メニュー→[検索...]を選択、または<Search> ボタンをクリック

### 各エリア説明

#### (1) Event Status: (ステータス条件選択エリア)

ステータス条件の選択を行うエリアです。

省略した場合には、すべてのフレーム（All Status）を検索対象とします。

**参考** IF, INFO のフレームは検索対象外です。

All Status	すべてのフレーム（省略時選択）
M1 Fetch	M1 フェッチ（BRM1Fetch を含む）
R/W	データ・リード/ライト（R, W を含む）
Read	データ・リード
Write	データ・ライト

**(2) Access Size: (アクセス・サイズ条件設定エリア)**

アクセス・サイズ条件の選択を行うエリアです。

この選択により、トレース・データを検索する際のデータ条件のアクセス幅が決定されます。

Byte	データ条件を 8 ビット幅で検索 (8 ビット・アクセス時)
Half Word	データ条件を 16 ビット幅で検索 (16 ビット・アクセス時)
Word	データ条件を 32 ビット幅で検索 (32 ビット・アクセス時)
No Condition	アクセス・サイズの検索をしない ("Data" エリアは入力不可)
Bit	データ条件を 1 ビット幅で検索 (8 ビット・アクセス時) 注 1,2

**注 1** ステータス条件にアクセス・イベントを指定した場合、Bit の選択肢は表示されません。また、Bit を指定してもエラーとなります。

**注 2** この場合、データ条件を 1 ビット幅で検索しますが、インサーキット・エミュレータの動作上、ビットそのものへのアクセスは直接トレースされないため、ID850QB は内部的にアドレス条件、データ条件を次のように設定することで疑似的なビット・アクセスを検索しています。

(入力例)		(トレース検索の設定)
アドレス : 3FF7000.1		アドレス : 3FF7000
データ : 1	→	データ : 00000010
		マスク : 11111101B

そのため、同一アドレスの他のビットに対するアクセスや、同一アドレスの 8 ビット全体に対するアクセスでも、指定した [ アドレス . ビット ] の値が一致している場合には、指定したステータスに従ってトレース・データが検索されます。

**参考** アクセス・サイズ条件の設定を省略した場合は、アドレス条件およびデータ条件から自動的に判断され、次のように設定されます。

- アドレス条件がビット設定の場合、Bit
- データ条件が 8 ビットの場合、Byte
- データ条件が 16 ビットの場合、Half Word
- データ条件が 32 ビットの場合、Word
- データ条件の指定が省略された場合、No Condition

**(3) アドレス条件設定エリア**

アドレス条件を指定するエリアです（省略可）。

設定範囲は次のようになります。

表 6 - 15 アドレス条件の設定範囲（トレース）

設定範囲	条件
$0 \leq \text{アドレス値} \leq 0xFFFFFFFF$	なし
$0 \leq \text{マスク値} \leq 0xFFFFFFFF$	なし

Address:	アドレス条件のアドレス値（下位アドレス - 上位アドレス）を設定します（省略可）。 数値入力時のデフォルト進数は 16 進です。 シンボルでの指定も可能です（「表 5 - 6 シンボルでの指定方法」）。 次の 3 種類の設定が可能です。	
	ポイント設定	下位アドレスのみに値を設定するか、下位アドレスと上位アドレスに同じ値を設定します。
	範囲設定	下位アドレスと上位アドレスに値を設定します。マスク設定不可。
	ビット設定	下位アドレスのみに値を設定するか、下位アドレスと上位アドレスに同じ値を設定します。 値は " address.bit " の形式で指定します。マスク設定不可。 なお、ビット位置を示す bit の値は $0 \leq \text{bit} \leq 7$ でなければなりません。
Mask:	アドレス値に対するマスク値を設定します（" ポイント設定 " の場合のみ可）（省略可）。 マスク値が 1 となるビットは、アドレス値が 0 または 1 のどちらであっても構いません。	

**設定例 1) 0x4000 - 0x40FF までが条件と一致**

Address:	0x4000 - 0x4000
Mask:	0xFF

**設定例 2) 0x4000, 0x4001, 0x4100, 0x4101 が条件と一致**

Address:	0x4000 - 0x4000
Mask:	0x101

**(4) データ条件設定エリア**

データ条件を設定するエリアです（省略可）。

設定範囲は (2) Access Size: (アクセス・サイズ条件設定エリア) での指定により異なります（「イベント・ダイアログ」の「(5) データ条件設定エリア」参照）。

Data:	データ条件を設定するデータ値を指定します。数値入力時のデフォルト進数は 16 進です。シンボルでの指定も可能です（「表 5-6 シンボルでの指定方法」参照）。
Mask:	データ値に対するマスク値を設定します（省略可）。マスク設定を行うと、マスク値が 1 となるビットは、データ値が 0 または 1 のいずれであっても構いません。

**設定例 1) 0x4000 - 0x40FF までが条件と一致**

Data	0x4000
Mask	0xFF

**設定例 2) 0x4000, 0x4001, 0x4100, 0x4101 が条件と一致**

Data	0x4000
Mask	0x101

**(5) サーチ条件設定エリア**

Scan Whole Region	指定した範囲全体を検索する場合、チェックします。	
Frame:	検索するフレーム番号を指定するエリアです。数値入力時のデフォルト進数は 10 進です。フレーム番号の指定形式での指定も可能です。	
Direction	検索方向を選択するエリアです。	
	Up	前方検索 現在カーソルのある位置より前（表示の上）方向に検索します。
	Down	後方検索 現在カーソルのある位置より後ろ（表示の下）方向に検索します（デフォルト）。

**(6) 機能ボタン**

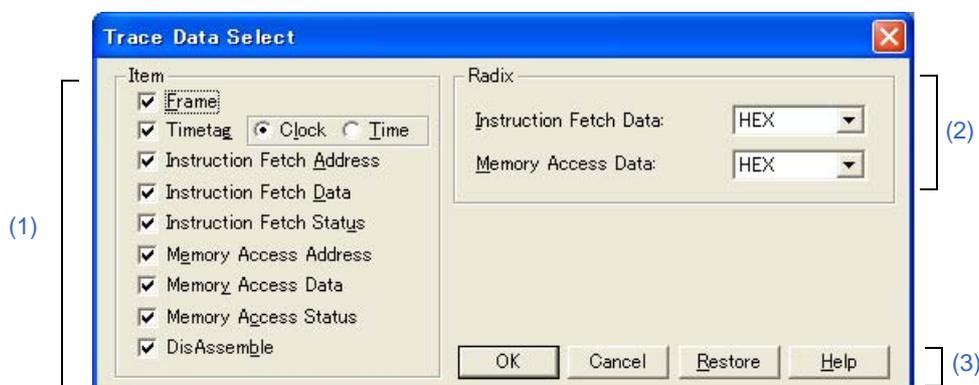
Find Next	指定したサーチ・データを条件に従い検索します。検索の結果、一致するフレームを反転表示します。連続して検索する場合には、再度このボタンを押します。
Set Find	指定された条件を検索条件として設定しダイアログをクローズします。
Cancel	このダイアログをクローズします。
Help	このダイアログのオンライン・ヘルプを表示します。

## トレース表示選択ダイアログ

[IECUBE]

トレース・ウィンドウに表示する項目を選択します（「5.10 トレース機能 [IECUBE]」参照）。

図 6 - 50 トレース表示選択ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

トレース・ウィンドウをカレント・ウィンドウにし、[表示]メニュー→[表示選択...]を選択

### 各エリア説明

#### (1) Item（トレース表示項目選択エリア）

トレース・ウィンドウに表示する項目を選択するエリアです。次の項目のチェックを外すことにより該当フィールドが非表示になります（「(3) トレース結果表示エリア」参照）。

Frame	Frame フィールド
Timetag	Time フィールド クロック表示（Clock）／時間表示（Time）の選択が可能です。
Instruction Fetch Address	Address（フェッチ系アクセス表示エリア）フィールド
Instruction Fetch Data	Data（フェッチ系アクセス表示エリア）フィールド
Instruction Fetch Status	Status（フェッチ系アクセス表示エリア）フィールド
Memory Access Address	Address（データ系アクセス表示エリア）フィールド
Memory Access Data	Data（データ系アクセス表示エリア）フィールド
Memory Access Status	Status（データ系アクセス表示エリア）フィールド
DisAssemble	DisAsm フィールド

**(2) Radix (トレース表示進数選択エリア)**

表示進数を選択するエリアです。次の表示フィールドに対する変更が可能です。

Instruction Fetch Data:	Data ( <a href="#">フェッチ系アクセス表示エリア</a> ) フィールド
Memory Access Data:	Data ( <a href="#">データ系アクセス表示エリア</a> ) フィールド

HEX	16 進表示 (デフォルト)
DEC	10 進数表示
OCT	8 進数表示
Bin	2 進数表示

**(3) 機能ボタン**

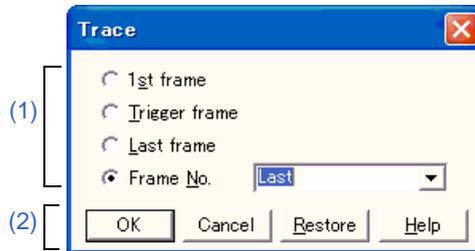
OK	<a href="#">トレース・ウィンドウ</a> に選択結果を反映します。
Cancel	このダイアログをクローズします。
Restore	変更を元に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

## フレーム指定ダイアログ

[IECUBE]

トレース・ウィンドウの表示開始位置を指定します（「5.10 トレース機能 [IECUBE]」参照）。

図 6 - 51 フレーム指定ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

トレース・ウィンドウをカレント・ウィンドウにし、[表示]メニュー → [移動...]を選択

### 各エリア説明

#### (1) フレーム選択エリア

移動先のフレームを次のうちから選択します。

1st frame	トレース・データの最初のフレームに移動させます。
Trigger frame	トレース・データ中のトリガ・フレームに移動させます。
Last frame	トレース・データの最後のフレームに移動させます。
Frame No.	指定したフレーム番号に移動させます。 指定は、ドロップダウン・リストからの選択、または直接入力により行います（「表 6 - 16 フレーム番号の指定形式」参照）。デフォルトでは、呼び出したウィンドウ内で選択した文字列、または "Last" が表示されます。 数値入力時のデフォルト進数は 10 進です。 0 を指定した場合、トレース・データの最初のフレームに移動します。 入力履歴は最大 16 個まで残すことができます。

表 6 - 16 フレーム番号の指定形式

指定	短縮形	内容
+ 数値	なし	カーソル位置のフレームから指定されたフレーム数（数値）だけ後ろ方向（表示の下方向）へ移動
- 数値	なし	カーソル位置のフレームから指定されたフレーム数（数値）だけ前方向（表示の上方向）へ移動
Top	O	トレース・データの最初のフレームに移動
First	S	" 1st frame " の指定と同様
Trigger	T	" Trigger frame " の指定と同様
Last	L	" Last frame " の指定と同様

**(2) 機能ボタン**

OK	指定した位置からトレース表示を行います。
Cancel	このダイアログをクローズします。
Restore	入力したデータを元に戻します。
Help	このダイアログのオンライン・ヘルプを表示します。

## トレース・ダイアログ

[IECUBE]

トレース・イベント条件の登録，設定，および表示を行います（「5.12 イベント機能」，「5.10 トレース機能 [IECUBE]」参照）。

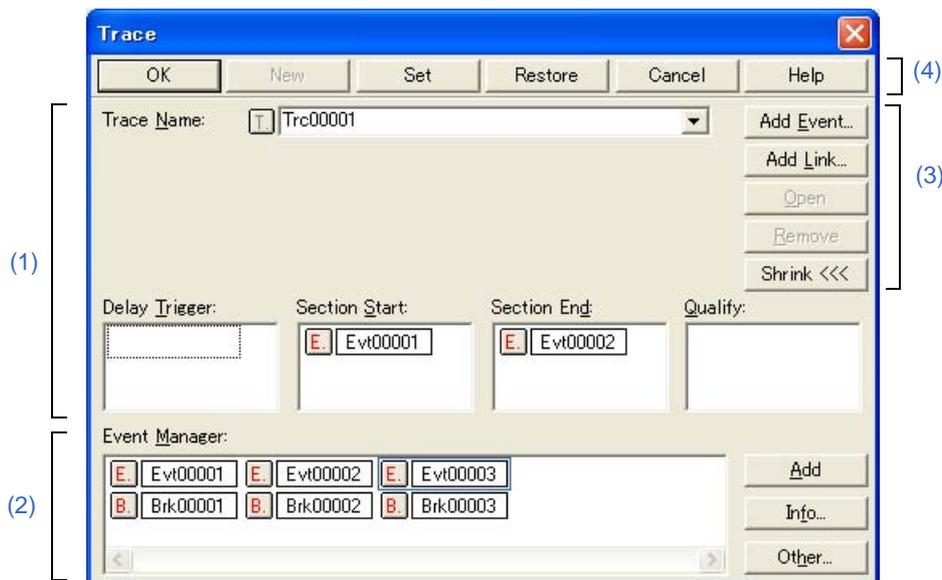
このダイアログ上では条件トレースを行う際のトレース・イベント条件を指定します（「表 5 - 13 条件トレースの種類」参照）。

トレース・イベント条件の登録，設定は，このダイアログ上で各項目を設定し，<OK> ボタンをクリックすることにより行います（最大 256 個）。登録したトレース・イベント条件は，イベント・マネージャで管理されます。

なお，同時に設定可能な（有効になる）トレース・イベント条件数には制限があります（「5.12.4 各種イベント条件の有効イベント数」参照）。

**参考** ユーザ・プログラム実行中でも，トレース・イベント条件の設定／有効化／無効化／削除操作は可能です。この場合，操作時にトレーサの動作を一瞬停止させます。

図 6 - 52 トレース・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

 ボタンをクリック，または [ イベント ] メニュー → [ トレース ... ] を選択

## 各エリア説明

### (1) トレース・イベント条件設定エリア

Trace Name:	トレース・イベント名の設定を行います。 名前は最大8文字までの英数字を直接入力してください。すでに登録済みのイベント条件の内容を表示するときは、ドロップダウン・リストから選択します。 このエリアの左側のマークは、イベント条件の使用状況を示します（「表5-18 イベント・アイコン」参照）。また、灰色のマークは、そのイベント条件が編集中で未登録の状態を示しています。このマークをクリックすることにより、イベント条件の有効/無効を切り替えることができます。
Qualify:	クオリファイ・トレース用のイベント条件を設定します（「5.10.6 条件トレースの設定」参照）。イベントを複数設定した場合には、それぞれイベントが成立するごとにトレースします。イベント条件の設定は、イベント・マネージャ・エリアから、設定するイベント・アイコンをドラッグし、このエリア上にドロップすることにより簡単に行えます（「5.12.3 各種イベント条件の設定」参照）。このエリアに設定できるイベント条件の個数は「表6-17 設定可能なイベント条件数」を参照してください。
Delay Trigger:	ディレイ・トリガ・イベント条件を設定します（「5.10.6 条件トレースの設定」参照）。イベント条件の設定は、イベント・マネージャ・エリアから、設定するイベント・アイコンをドラッグして、このエリア上にドロップすることにより簡単に行えます（「5.12.3 各種イベント条件の設定」参照）。このエリアに設定できるイベント条件の個数は「表6-17 設定可能なイベント条件数」を参照してください。
Section Start: Section End:	セクション・トレース開始用、終了用のイベント条件を設定します（「5.10.6 条件トレースの設定」参照）。 イベント条件の設定は、イベント・マネージャ・エリアから、設定するイベント・アイコンをドラッグして、このエリア上にドロップすることにより簡単に行えます（「5.12.3 各種イベント条件の設定」参照）。このエリアに設定できるイベント条件の個数は「表6-17 設定可能なイベント条件数」を参照してください。

表6-17 設定可能なイベント条件数

接続 IE	イベント条件 合計（実行/アクセス）	イベント・リンク条件
[IECUBE]	14 (8 <sup>*a</sup> /6)	1

\*a 実行後イベントのみ使用可

**(2) イベント・マネージャ・エリア**

登録されているイベント一覧を表示しています。

このエリア上のイベント・アイコンを各種イベント設定ダイアログのイベント設定エリアにドラッグ&ドロップすることにより、容易に各種イベント条件の設定が行えます（「5.12.3 各種イベント条件の設定」参照）。

このエリアは、各種イベント関連ダイアログで共通です（「ブレーク・ダイアログ」の「(2) イベント・マネージャ・エリア」参照）。

**(3) 機能ボタン（イベント条件の内容表示等）**

イベント条件設定エリアに置かれたイベント条件の内容表示／削除、およびイベント・マネージャ・エリアの表示／非表示等の操作を行います（「ブレーク・ダイアログ」の「(3) 機能ボタン（イベント条件の内容表示等）」参照）。

**(4) 機能ボタン（イベント条件の登録／削除／有効化／無効化）**

イベントの登録／削除、および有効化／無効化等の操作を行うボタンです。

指定したイベント条件のイベントは、<OK>（または<Set>）ボタンを押すことにより登録、設定（有効化）されます（「ブレーク・ダイアログ」の「(4) 機能ボタン（イベント条件の登録／削除／有効化／無効化）」参照）。

## ディレイ・カウント設定ダイアログ

[IECUBE]

ディレイ・カウントの設定、および表示を行います（「5.10 トレース機能 [IECUBE]」参照）。

ディレイ・カウントを設定することにより、**トレース・ダイアログ**で設定したディレイ・トリガ・イベント条件成立後、指定したディレイ・カウント分のトレースを行うことができます（「5.10.6 条件トレースの設定」参照）。

図 6 - 53 ディレイ・カウント設定ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

[ イベント ]メニュー→[ ディレイ・カウント ... ]を選択

### 各エリア説明

#### (1) ディレイ・カウント設定エリア

Delay Count	次の中から選択します。	
	FIRST	トリガ・ポインタをトレース・データの先頭に置き、全フレーム分をトレース後、トレーサを停止します。
	MIDDLE	トリガ・ポインタをトレース・データの中央に置き、全フレームの1/2 フレーム分をトレース後、トレーサを停止します。
	LAST	トリガ・ポインタをトレース・データの最後に置き、ただちにトレーサを停止します。

**注意** ディレイ・カウント値は、**拡張オプション設定ダイアログ**で指定したトレース・メモリ・サイズにより異なります。

例) トレース・メモリ・サイズが 256 K フレームの場合

FIRST: 256K - 1 フレーム

MIDDLE: (256K / 2) - 1 フレーム

LAST: 5 フレーム

## (2) 機能ボタン

OK	設定を有効にして、このダイアログをクローズします。
Restore	設定を元に戻します。
Cancel	このダイアログをクローズします。
Help	このダイアログのオンライン・ヘルプを表示します。

## コード・カバレッジ・ウィンドウ

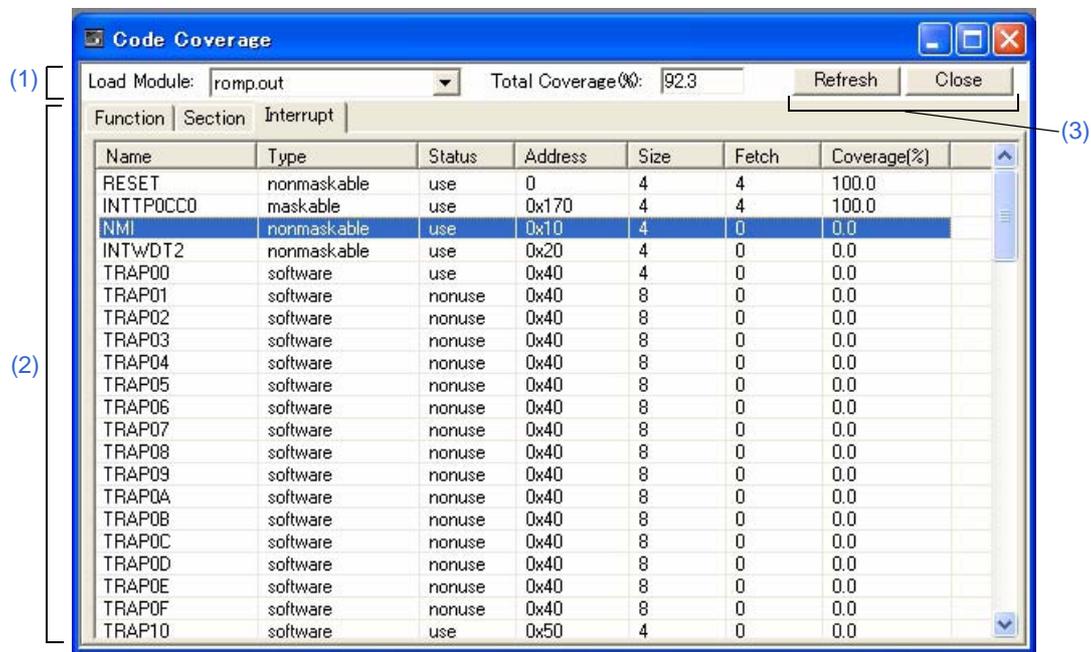
[IECUBE]

コード・カバレッジの測定結果表示を行います (C0 カバレッジ) (「5.11 カバレッジ測定機能 [IECUBE]」参照)。ユーザ・プログラムの実行、未実行の箇所の確認は、ソース・テキスト・ウィンドウ、逆アセンブル・ウィンドウで行います。

**注意** フラッシュ・セルフ・プログラミングのエミュレーションにより内蔵フラッシュ・メモリを書き換えた場合には、カバレッジ測定結果は不正確になります (「フラッシュ・オプション設定ダイアログ」参照)。

**参考** ID850QB 起動時にカバレッジ・データはクリアされます。

図 6 - 54 コード・カバレッジ・ウィンドウ



- オープン方法
- 各エリア説明
- コンテキスト・メニュー

### オープン方法

 ボタンをクリック、または [ブラウザ] メニュー → [コードカバレッジ] を選択

## 各エリア説明

### (1) カバレッジ情報表示エリア

Load Module:	ダウンロードしたロード・モジュール・ファイルを選択します。 ロード・モジュール・ファイルをダウンロードしていない場合は空欄になります。
Total Coverage (%):	コード・カバレッジを測定した領域の総網羅度を表示します。 <b>総網羅度 = 総関数実行 (フェッチ) サイズ / 総関数サイズ</b> で計算します (ただし、カバレッジ測定範囲外のセクションは除外)。 ロード・モジュール・ファイルをダウンロードしていない場合は空欄になります。

### (2) 測定結果表示エリア

タブ (関数 / セクション / 割り込みハンドラ) 別に測定結果を表示します。

カバレッジ測定結果の更新は、ブレーク時に自動的に行われます (ユーザ・プログラム実行中は更新されません)。ロード・モジュール・ファイルをダウンロードしていない場合は空欄になります。

また、このエリアは、選択行の開始アドレス値をジャンプ・ポインタとして**ソース・テキスト・ウィンドウ**、または**逆アセンブル・ウィンドウ**にジャンプする機能を持っています。ジャンプ先のウィンドウでは、ジャンプ・ポインタから表示されます。

ジャンプ機能は、ジャンプ元の行を選択し、[ジャンプ]メニュー→[ソース / アセンブル]を選択することにより行います。また、ジャンプ元の行をダブルクリックすることによってもジャンプさせることができます。

**参考** 各列のタイトル (ラベル上) をクリックすることにより、表示アイテムがソートされます (昇順 / 降順はクリックにより切り替わります)。

#### (a) [Function] タブ選択時

Name	関数名 (アセンブラ・ソース・ファイルの場合は、ファイル単位に関数として表示)
File	関数が定義されているファイル名
Address	関数の開始アドレス
Size	関数のサイズ (単位: バイト)
Fetch	実行 (フェッチ) したバイト数
Coverage (%)	関数の網羅度 (0 - 100 %) ----: カバレッジ測定範囲外の関数である場合

#### (b) [Section] タブ選択時

Name	セクション名
Type	セクションタイプ (code, data)
Address	セクションの開始アドレス
Size	セクションのサイズ (単位: バイト)
Fetch	実行 (フェッチ) したバイト数

Coverage [%]	セクションの網羅度 (0 - 100 %) ---- : カバレッジ測定範囲外のセクションである場合
--------------	---

## (c) [Interrupt] タブ選択時

Name	割り込み要求名
Type	割り込みタイプ (nonmaskable, maskable, software, security id, flash mask option)
Status	プログラムでの使用状況 ---- : 不明
Address	割り込みハンドラの開始アドレス
Size	割り込みハンドラのサイズ (単位 : バイト) "Status" が "use" 以外の場合には, 最大サイズ
Fetch	実行 (フェッチ) したバイト数
Coverage [%]	割り込みハンドラの網羅度 (0 - 100 %) ---- : カバレッジ測定範囲外の割り込みハンドラである場合

## (3) 機能ボタン

Refresh	ウィンドウの内容を最新のデータに更新します。
Close	このウィンドウをクローズします。

## コンテキスト・メニュー

ソース	カーソル位置のデータ値をジャンプ先アドレスとして、該当するソース・テキストおよびソース行を表示します (「5.16.2 ジャンプ機能」参照)。 ただし、ジャンプ先アドレスに行情報がない場合には、ジャンプすることはできません。 ソース・テキスト・ウィンドウをオープンします。 アクティブ状態のソース・テキスト・ウィンドウが表示されている場合は、そのウィンドウを最前面 (操作対象) に表示します。
アセンブル	カーソル位置のデータ値をジャンプ先アドレスとして、そのアドレスから逆アセンブル表示します (「5.16.2 ジャンプ機能」参照)。 逆アセンブル・ウィンドウをオープンします。 アクティブ状態の逆アセンブル・ウィンドウが表示されている場合は、そのウィンドウを最前面 (操作対象) に表示します。
クリア	カバレッジ測定結果をクリアします。
選択 ...	1M バイト以上の空間のカバレッジ測定範囲の選択をします。 カバレッジ範囲選択ダイアログをオープンします。

## カバレッジ範囲選択ダイアログ

[IECUBE]

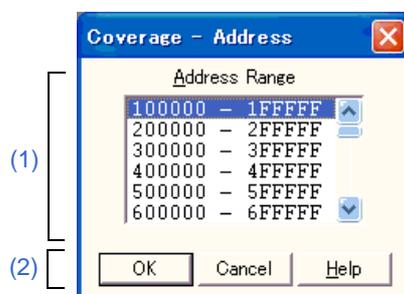
コード・カバレッジ・ウィンドウに表示するカバレッジ測定範囲の選択を行います（「5.11 カバレッジ測定機能 [IECUBE]」参照）。

測定可能な範囲は以下のとおりです。このダイアログでは、任意の 1M バイト分を選択可能です。

表 6 - 18 カバレッジ測定範囲（詳細）

接続 IE	コード・カバレッジ測定範囲
[IECUBE]	- アドレス 0x000000 - 0x0FFFFFF の 1M バイト空間（固定の測定領域） - アドレス 0x100000 - 0x3FFFFFF 番地内の任意の 1M バイト空間（このダイアログで選択することにより測定可能な領域）（デフォルト：0x3F00000 - 0x3FFFFFFF）

図 6 - 55 カバレッジ範囲選択ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

[オプション]メニュー→[カバレッジ]→[選択...]を選択

### 各エリア説明

#### (1) アドレス選択エリア

Address Range	カバレッジ測定を行う任意の 1M バイト空間を選択するエリアです。測定範囲の変更を行うことにより、以前に選択した範囲の測定結果（カバレッジ・データ）はクリアされますが、固定の測定領域である 0 - 0x0FFFFFF のカバレッジ・データはクリアされません。エリアを選択し <OK> ボタンをクリックすることによりカバレッジ測定範囲が決定されます。
---------------	--

#### (2) 機能ボタン

OK	選択に従いカバレッジ測定範囲を有効にします。
Cancel	このダイアログをクローズします。
Help	このダイアログのオンライン・ヘルプを表示します。

## ソフトウェア・ブレーク・マネージャ

ソフトウェア・ブレークの表示、有効／無効の切り替え、削除を行います（「5.4.4 ハードウェア・ブレークとソフトウェア・ブレーク」参照）。

なお、ソフトウェア・ブレーク・ポイントの設定は、このウィンドウ上では行えません。ソース・テキスト・ウィンドウ、または逆アセンブル・ウィンドウ上で行います（「5.4.2 ブレーク・ポイントの設定」参照）。

**注意 1** ソフトウェア・ブレーク・ポイントは、ユーザ・プログラム実行中の設定／削除が可能です。

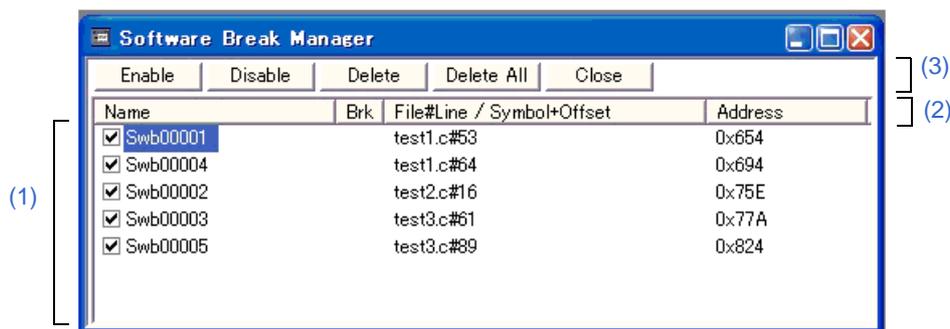
なお、ユーザ・プログラム実行中に設定／削除／有効／無効を行った場合には、ユーザ・プログラムを一瞬ブレークさせるため、警告が表示されます。

**注意 2** 疑似リアルタイム・モニタ機能 (Break When Readout) を有効にした場合、ソフトウェア・ブレーク・ポイントは設定できません。

また、有効なソフトウェア・ブレーク・ポイントを設定している場合には、ユーザ・プログラム実行中の書き込み (DMM 機能) はできません。

**参考** 各列のタイトル (ラベル上) をクリックすることにより、表示アイテムをソートします (昇順／降順はクリックにより切り替わります)。

図 6 - 56 ソフトウェア・ブレーク・マネージャ



- オープン方法
- 各エリア説明

### オープン方法

[イベント]メニュー→[ソフトウェア・ブレーク・マネージャ]を選択

## 各エリア説明

### (1) ブレーク情報表示エリア

Name	登録されているイベント名とそのイベントの有効／無効状態を表すチェック・ボックスを表示しています。 イベント名は、デフォルトで "Swb+[ 数字 ]" という形式で表示されていますがエクスプローラ・ライクに変更が可能です（最大半角英数字 256 文字分、日本語使用可能）。 イベントを無効にするには該当するイベント名のチェックを外します。新規作成時のイベントは有効です。 また、イベント名をダブルクリックすることにより、対応するソース行がある場合は、 <a href="#">ソース・テキスト・ウィンドウ</a> へ対応するソース行がない場合は、 <a href="#">逆アセンブル・ウィンドウ</a> へのジャンプが可能です。
Brk	カレント PC 位置に設定されているソフトウェア・ブレーク・イベントに対し '}' マークを表示しています。 これにより、ブレークの要因となったソフトウェア・ブレーク・イベントの特定が容易です。
File#Line / Symbol+Offset	ソフトウェア・ブレーク・イベントの設定場所を次の形式で表示しています。 <b>プログラム \$ ファイル名 # 行番号</b> （対応するソース行がある場合） <b>プログラム \$ ファイル名 # シンボル + オフセット</b> （対応するソース行がない場合） シンボルの再ダウンロードの際などにイベントの再評価を行う場合、これにもとづいて行われます。
Address	ソフトウェア・ブレーク・イベントが設定されているアドレスを表示するエリアです。

### (2) アイテム・ラベル・エリア

各アイテム・ラベルをクリックすることにより、イベント条件を並び替えます。昇順／降順はクリックすることにより切り替わります。

Name	文字列を辞書式の順序で比較（昇順／降順）
Brk	ソートなし
File#Line / Symbol+Offset	文字列を辞書式の順序で比較（昇順／降順）
Address	数値の大小で比較（昇順／降順）

### (3) 機能ボタン

Enable	選択しているイベントを有効にします。
Disable	選択しているイベントを無効にします。
Delete	選択しているイベントを削除します。
Delete All	設定されているすべてのソフトウェア・ブレーク・イベントを削除します。
Close	このウィンドウをクローズします。

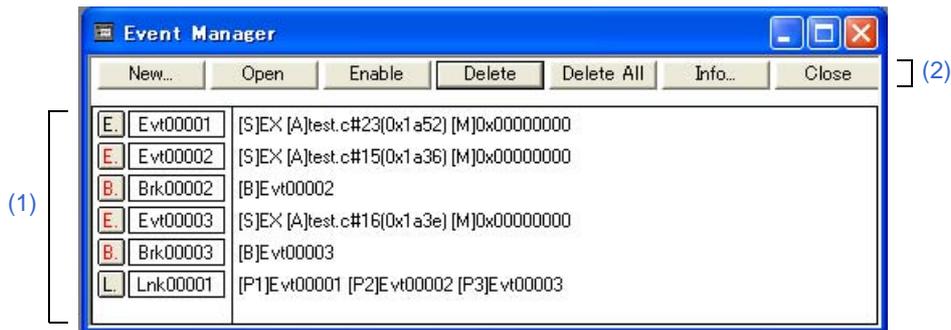
## イベント・マネージャ

各種イベント条件の表示（詳細表示）、有効／無効の切り替え、削除等のイベント条件の管理を行います（「5.12 イベント機能」参照）。

このウィンドウでは、コンテキスト・メニュー、機能ボタン等による操作が可能です。

イベント・アイコンはジャンプ機能のジャンプ・ポインタになります。

図 6 - 57 イベント・マネージャ（詳細表示モード）



- オープン方法
- 各エリア説明
- [表示]メニュー（イベント・マネージャ専用部分）
- コンテキスト・メニュー

### オープン方法

 ボタンをクリック、または[イベント]メニュー→[イベント・マネージャ]を選択

### 各エリア説明

#### (1) イベント表示エリア

登録されている各種イベント条件を一覧表示、および詳細表示するエリアです。

詳細表示は、コンテキスト・メニュー→[詳細表示]の選択により行います。

一覧表示時	 <p>イベント・アイコンを表示します（「表 5 - 18 イベント・アイコン」参照）。 イベント・アイコンは、ジャンプ・ポインタとなります（「5.16.2 ジャンプ機能」参照）。</p>
-------	--

詳細表示時	<table border="1"> <tr><td>E</td><td>Evt00001</td><td>[S]EX [A]est.c#23(0x1a52) [M]0x00000000</td></tr> <tr><td>E</td><td>Evt00002</td><td>[S]EX [A]est.c#15(0x1a36) [M]0x00000000</td></tr> <tr><td>B</td><td>Brk00002</td><td>[B]Evt00002</td></tr> <tr><td>E</td><td>Evt00003</td><td>[S]EX [A]est.c#16(0x1a3e) [M]0x00000000</td></tr> <tr><td>B</td><td>Brk00003</td><td>[B]Evt00003</td></tr> <tr><td>L</td><td>Lnk00001</td><td>[P1]Evt00001 [P2]Evt00002 [P3]Evt00003</td></tr> </table> <p>イベント内容の詳細を、キー情報をセパレータとして表示します（「表 6 - 19 イベント詳細表示時のセパレータ」参照）。</p>	E	Evt00001	[S]EX [A]est.c#23(0x1a52) [M]0x00000000	E	Evt00002	[S]EX [A]est.c#15(0x1a36) [M]0x00000000	B	Brk00002	[B]Evt00002	E	Evt00003	[S]EX [A]est.c#16(0x1a3e) [M]0x00000000	B	Brk00003	[B]Evt00003	L	Lnk00001	[P1]Evt00001 [P2]Evt00002 [P3]Evt00003
E	Evt00001	[S]EX [A]est.c#23(0x1a52) [M]0x00000000																	
E	Evt00002	[S]EX [A]est.c#15(0x1a36) [M]0x00000000																	
B	Brk00002	[B]Evt00002																	
E	Evt00003	[S]EX [A]est.c#16(0x1a3e) [M]0x00000000																	
B	Brk00003	[B]Evt00003																	
L	Lnk00001	[P1]Evt00001 [P2]Evt00002 [P3]Evt00003																	

表 6 - 19 イベント詳細表示時のセパレータ

キー情報	内容
<b>イベント条件の場合</b>	
[S]	ステータス条件
[Z]	アクセス・サイズ条件
[A]	アドレス条件 シンボルや式の場合：(実際のアドレス)
[D]	データ条件 シンボルや式の場合：(実際のアドレス)
[M]	マスク条件
<b>イベント・リンク条件の場合</b>	
[P1] - [P4]	n 段目のイベント・リンク条件
[D]	ディスエーブル条件
[P]	パス・カウント条件
<b>ブレーク条件の場合</b>	
[B]	ブレーク条件
<b>トレース条件の場合 [IECUBE]</b>	
[M]	トレーサ制御モード
[T]	ディレイ・トリガ条件
[D]	ディレイ・カウント
[S]	トレース開始条件
[E]	トレース終了条件
[Q]	クオリファイ・トレース条件
<b>タイマ条件の場合 [IECUBE]</b>	
[S]	タイマ測定開始条件
[E]	タイマ測定終了条件
[U]	タイマ測定単位
[B]	タイムアウト・ブレーク条件

## (2) 機能ボタン

OK	編集中のイベント条件があれば、自動的に登録してこのダイアログをクローズします。各種イベント条件は登録と同時に有効になります。	
New...	新規イベント種別選択ダイアログをオープンします。	
		各ボタンを押すことにより、各種イベント条件設定ダイアログを新規のイベント名が設定された状態でオープンします。各種イベント条件設定ダイアログのオープン後はこのダイアログはクローズされます。<Cancel>を押すことにより、イベント・マネージャに戻ります。
Set	各種イベント条件を登録します。イベント登録後、ダイアログはクローズしないため、続けて登録可能です。各種イベント条件は、登録と同時に有効になります。	
Open	選択しているイベント条件（1つ）に対応する各種イベント設定ダイアログをオープンします。各設定ダイアログには、選択したイベント条件の内容が表示されます。イベント・アイコンのダブルクリック、および Enter キーも同じ動作をします。	
Enable Disable	選択している各種イベント条件の有効（Enable）／無効（Disable）を切り替えます。ただし、イベント条件、イベント・リンク条件はこの対象となりません。イベント・アイコンのマーク部分をクリックすることでも同じ動作をします。	
Remove	選択しているイベント条件を削除します。イベント条件、またはイベント・リンク条件を削除する時、そのイベントが各種イベント条件として使用されている場合には、エラーとなり削除できません。	
Delete All	すべてのイベント条件を削除します。	
Info...	表示情報選択ダイアログをオープンします。表示モードの変更、イベント名の並べ替えを行います。	
		<ul style="list-style-type: none"> <li>&lt;Sort by Name&gt; ... イベントを名前順に並べ変えます。</li> <li>&lt;Sort by Kind&gt; ... イベントを種類順に並べ替えます。</li> <li>&lt;Unsort&gt; ... 並べ替えを行わず、登録順に表示します。</li> <li>&lt;Detail&gt; ... 詳細表示モードにします。</li> <li>&lt;Overview&gt; ... 一覧表示モードにします。</li> <li>&lt;Cancel&gt; ... このダイアログをクローズします（ESC キーと同様）。</li> </ul>
Close	このダイアログをクローズします。	

## [表示]メニュー（イベント・マネージャ専用部分）

イベント・マネージャのアクティブ時には、共通の [表示] メニューに、以下の項目が追加されます。

すべてのイベントを選択	登録されているすべてのイベントを選択します。
イベント削除	選択されているイベントを削除
名前順	アイコンをイベント名の順に表示
種類順	アイコンをイベント種類の順に表示
並べ替えなし	並べ替えをしない（デフォルト）
詳細表示	詳細表示
一覧表示	一覧表示（デフォルト）

## コンテキスト・メニュー

名前順	アイコンをイベント名の順に表示
種類順	アイコンをイベント種類の順に表示
並べ替えなし	並べ替えをしない（デフォルト）
詳細表示	詳細表示
一覧表示	一覧表示（デフォルト）
ソース	<p>選択イベント位置をジャンプ先アドレスとして、該当するソース・テキストおよびソース行を表示します（「5.16.2 ジャンプ機能」参照）。</p> <p>ただし、ジャンプ先アドレスに行情報がない場合には、ジャンプすることはできません。</p> <p>ソース・テキスト・ウィンドウをオープンします。</p> <p>アクティブ状態のソース・テキスト・ウィンドウが表示されている場合は、そのウィンドウを最前面（操作対象）に表示します。</p>
アセンブル	<p>選択イベント位置をジャンプ先アドレスとして、そのアドレスから逆アセンブル表示します（「5.16.2 ジャンプ機能」参照）。</p> <p>逆アセンブル・ウィンドウをオープンします。</p> <p>アクティブ状態の逆アセンブル・ウィンドウが表示されている場合は、そのウィンドウを最前面（操作対象）に表示します。</p>
メモリ	<p>選択イベント位置をジャンプ先アドレスとして、そのアドレスからメモリ内容を表示します（「5.16.2 ジャンプ機能」参照）。</p> <p>メモリ・ウィンドウをオープンします。</p> <p>アクティブ状態のメモリ・ウィンドウが表示されている場合は、そのウィンドウを最前面（操作対象）に表示します。</p>

## イベント・ダイアログ

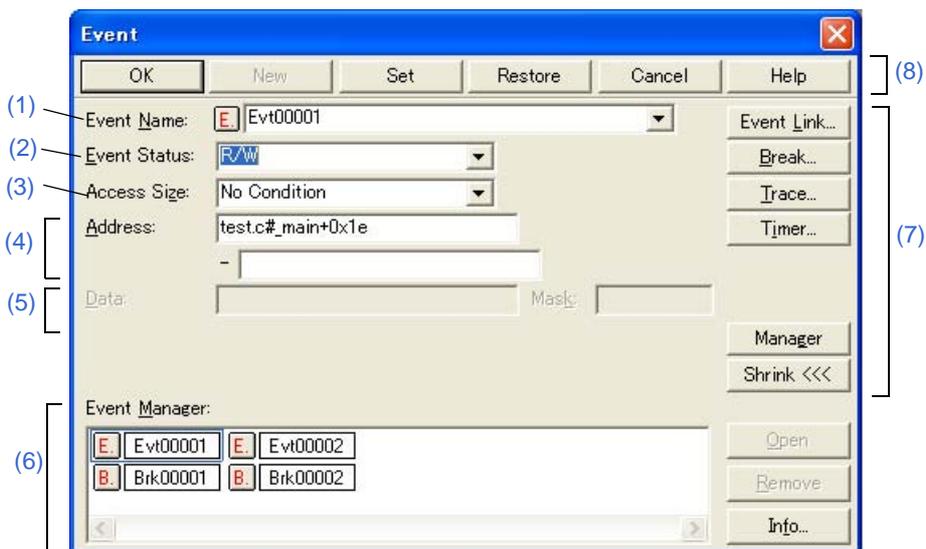
イベント条件の登録と表示を行います（「5.12 イベント機能」参照）。

イベント条件の登録は、このダイアログ上で各項目を設定し、<OK> ボタンをクリックすることにより行います。登録したイベント条件は、[イベント・マネージャ](#)で管理されます。

1つのイベント条件は複数の[各種イベント条件](#)として設定することができます。

ただし、設定できる（有効になる）各種イベント条件数には制限があります（「5.12.4 各種イベント条件の有効イベント数」参照）。

図 6 - 58 イベント・ダイアログ



- [オープン方法](#)
- [各エリア説明](#)

### オープン方法

#### ■通常モードでオープン

次の方法でイベント・ダイアログをオープンした場合、目的を限定せずにイベント条件を登録することができます。



ボタンをクリック、または[イベント]メニュー→[イベント...]を選択

#### ■選択モードでオープン

次の方法でイベント・ダイアログをオープンした場合、<OK> ボタンを押すと、呼び出し元の設定ダイアログのイベント条件として登録することができます（タイトル・バーに呼び出し元の設定ダイアログ名が表示されます）。

各種イベント設定ダイアログ上で <Add Event...> ボタンをクリック

## 各エリア説明

### (1) Event Name: (イベント名設定エリア)

イベント名の指定を行うエリアです。

指定は、最大8文字までの英数字を直接入力することにより行います。

作成済みのイベント条件の内容を表示する場合にはドロップダウン・リストから選択します。

選択モードでは、選択したイベント条件を、イベント・ダイアログを呼び出した元の設定ダイアログのイベント条件設定エリアに設定することができます。

このエリアの左側のマークは、イベント条件の使用状況を示します（「表5-18 イベント・アイコン」参照）。なお、灰色のE. マークは、そのイベント条件が編集集中で未登録の状態を示しています。

### (2) Event Status: (ステータス条件設定エリア)

ステータス条件の選択を行うエリアです。

この選択により実行イベントとアクセス・イベントの種別が決定されます（実行イベントを指定した場合 "Access Size:", "Data:", Mask: " は入力不可）。

指定できるステータス条件を次に示します。

実行イベント		
Execution	EX	プログラム実行
Before Execution	EX-B	プログラム実行（実行前ブレーク） <sup>注1</sup>
アクセス・イベント		
R/W	RW	データ・リード/ライト
Read	R	データ・リード
Write	W	データ・ライト
R/W(Data not Equal)	RWND	データ・リード/ライト (データ条件が不一致の時にイベントを発生) <sup>注2</sup>
Read(Data not Equal)	RND	データ・リード (データ条件が不一致の時にイベントを発生) <sup>注2</sup>
Write(Data not Equal)	WND	データ・ライト (データ条件が不一致の時にイベントを発生) <sup>注2</sup>

**注1** 複数個指定可能ですが、有効（Enable）にできるのはアクセス・イベントと合わせて2個までです。アドレス範囲は指定できません。ブレーク・イベント条件にのみ使用できます。

**注2** この場合、データ条件は省略不可です。またアクセス・サイズ条件に "No Condition", "Bit" を選択することはできません。

**(3) Access Size: (アクセス・サイズ条件設定エリア)**

アクセス・サイズ条件の選択を行うエリアです。

ドロップ・ダウンリストからの選択により、アクセス・イベントで検出する際のデータ条件のアクセス幅が決定されます。

アクセス・サイズの設定を省略した場合は、アドレス条件およびデータ条件から自動的に判断され、設定されます。

Byte	データ条件を 8 ビット幅で検出 (8 ビット・アクセス時)
Half Word	データ条件を 16 ビット幅で検出 (16 ビット・アクセス時)
Word	データ条件を 32 ビット幅で検出 (32 ビット・アクセス時)
No Condition	アクセス・サイズの検出をしない (Data エリアは入力不可)
Bit	データ条件を 1 ビット幅で検出 (8 ビット・アクセス時) 注

**注** この場合、データ条件を 1 ビット幅で検索しますが、インサーキット・エミュレータの動作上、ビットそのものへのアクセスは直接トレースされないため、ID850QB は内部的にアドレス条件、データ条件を次のように設定することで疑似的なビット・アクセスを検索しています。

(入力例)		(トレース検索の設定)
アドレス : 3FF7000.1		アドレス : 3FF7000
データ : 1	→	データ : 00000010
		マスク : 11111101B

そのため、同一アドレスの他のビットに対するアクセスや、同一アドレスの 8 ビット全体に対するアクセスでも、指定した [ アドレス . ビット ] の値が一致している場合には、指定したステータスに従ってトレース・データが検索されます。

**(4) アドレス条件設定エリア**

アドレス条件を指定するエリアです (省略可)。

設定範囲は次のようになります。

表 6 - 20 アドレス条件の設定範囲 (イベント)

接続 IE	設定範囲
[IECUBE] [MINICUBE]	$0 \leq \text{アドレス値} \leq 0xFFFFFFFF$

Address:	アドレス条件のアドレス値（下位アドレス - 上位アドレス）を設定します（省略可）。数値入力時のデフォルト進数は 16 進です。シンボルでの指定も可能です（「表 5-6 シンボルでの指定方法」）。次の 3 種類の設定が可能です。	
	ポイント設定	下位アドレスのみに値を設定するか、下位アドレスと上位アドレスに同じ値を設定します。
	範囲設定	下位アドレスと上位アドレスに値を設定します。マスク設定不可。
	ビット設定	下位アドレスのみに値を設定するか、下位アドレスと上位アドレスに同じ値を設定します。値は " address.bit " の形式で指定します。マスク設定不可。なお、ビット位置を示す bit の値は $0 \leq \text{bit} \leq 7$ でなければなりません。
Mask:	アドレス値に対するマスク値を設定します（" ポイント設定 " の場合のみ可）（省略可）。マスク値が 1 となるビットは、アドレス値が 0 または 1 のどちらであっても構いません。	

**注意** イベント設定では、物理アドレスとイメージ空間を区別しているため、28 ビットのアドレス指定をしてください

#### 設定例 1) 0x4000 - 0x40FF までが条件と一致

Address:	0x4000 - 0x4000
Mask:	0xFF

#### 設定例 2) 0x4000, 0x4001, 0x4100, 0x4101 が条件と一致

Address:	0x4000 - 0x4000
Mask:	0x101

#### (5) データ条件設定エリア

データ条件（データ値、マスク値）を設定するエリアです。

数値入力時のデフォルトの進数は 16 進です。

設定範囲は、" Access Size:" での指定により、次のようになります。

Data:	データ条件（データ値、マスク値）を設定するエリアです。数値入力時のデフォルトの進数は 16 進です。設定範囲は、" Access Size:" での指定により、次のようになります。	
	Byte	$0 \leq \text{データ値} \leq 0xFF$ $0 \leq \text{マスク値} \leq 0xFF$
	Half Word	$0 \leq \text{データ値} \leq 0xFFFF$ $0 \leq \text{マスク値} \leq 0xFFFF$
	Word	$0 \leq \text{データ値} \leq 0xFFFFFFFF$ $0 \leq \text{マスク値} \leq 0xFFFFFFFF$
	Bit	データ値 = 0, 1 マスク値 = 指定不可

Mask:	データ値に対するマスク値を設定します（省略可）。 マスク設定を行うと、マスク値が1となるビットは、データ値が0または1のいずれであっても構いません。
-------	---

### (6) イベント・マネージャ・エリア

Event manager:	登録されているイベント一覧を表示するエリアです（「表 5-18 イベント・アイコン」, 「(4) イベント・マネージャ・エリアでの操作」参照）。
<Open>	選択しているイベント条件（1つ）に対応する各種イベント設定ダイアログをオープンします。各設定ダイアログには、選択したイベント条件の内容が表示されます。イベント・アイコンのダブルクリックする、および Enter キーも同じ動作をします。
<Remove>	選択しているイベント条件を削除します。 イベント条件、またはイベント・リンク条件を削除する時、そのイベントが各種イベント条件として使用されている場合には、エラーとなり削除できません。
<Info...>	表示情報選択ダイアログをオープンします。 表示モードの変更、イベント名の並べ替えを行います。
	 <p>&lt;Sort by Name&gt; ... イベントを名前順に並べ変えます。          &lt;Sort by Kind&gt; ... イベントの種類順に並べ替えます。          &lt;Unsort&gt; ... 並べ替えを行わず、登録順に表示します。          &lt;Detail&gt; ... 詳細表示モードにします。          &lt;Overview&gt; ... 一覧表示モードにします。          &lt;Cancel&gt; ... このダイアログをクローズします（ESC キーと同様）。</p>

### (7) 機能ボタン（各種イベント条件への設定）

Event Link... Break... Trace... Timer...	各ボタンを押すことにより、各種イベント条件設定ダイアログを新規のイベント名が設定された状態でオープンします。
Manager...	イベント・マネージャをオープンします。
Expand >>> Shrink <<<	イベント・マネージャ・エリアの表示／非表示を切り替えます。 ダイアログのサイズが拡張／縮小されます。

### (8) 機能ボタン（イベント条件の登録／削除／有効化／無効化）

イベントの登録／削除、および有効化／無効化等の操作を行うボタンです。

指定したイベント条件のイベントは、<OK>（または<Set>）ボタンを押すことにより登録されます。

このボタンは、各種イベント関連ダイアログで共通です（「ブレーク・ダイアログ」の「(4) 機能ボタン（イベント条件の登録／削除／有効化／無効化）」参照）。

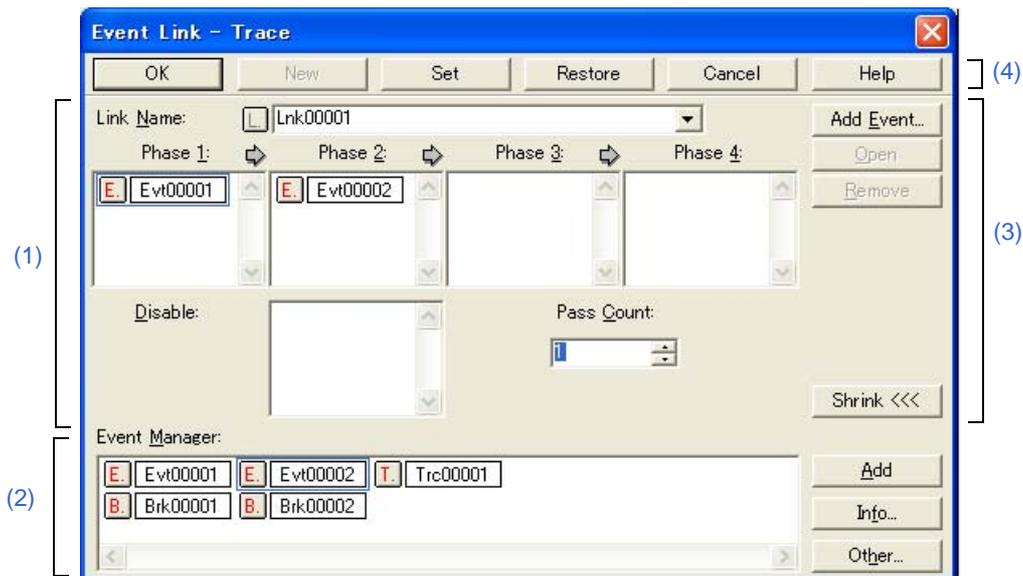
## イベント・リンク・ダイアログ

イベント・リンク条件の登録と表示を行います（「5.12 イベント機能」参照）。

イベント・リンク条件の登録は、このダイアログ上で各項目を設定し、<OK> ボタンをクリックすることにより行います（最大 256 個）。登録したイベント・リンク条件は、[イベント・マネージャ](#)で管理されます。

ただし、設定できる（有効になる）イベント・リンク条件には制限があります（「5.12.4 各種イベント条件の有効イベント数」参照）。

図 6 - 59 イベント・リンク・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

#### ■通常モードでオープン

次の方法でイベント・リンク・ダイアログをオープンした場合、目的を限定せずにイベント・リンク条件を登録することができます。

[イベント]メニュー→[イベントリンク ...]を選択

#### ■選択モードでオープン

次の方法でイベント・リンク・ダイアログをオープンした場合、<OK> ボタンを押すと、呼び出し元の設定ダイアログのイベント・リンク条件として登録することができます。

各種イベント設定ダイアログ上で <Add Link...> ボタンをクリック

（タイトル・バーに呼び出し元の設定ダイアログ名が表示）

## 各エリア説明

### (1) イベント・リンク条件設定エリア

Link Name:	<p>イベント・リンク名の設定を行います。 名前は最大8文字までの英数字を直接入力してください。 すでに作成してあるイベント・リンク条件の内容を表示するときは、ドロップダウン・リストから選択します。 選択モードでは、選択したイベント・リンク条件を、イベント・リンク・ダイアログを呼び出した元の設定ダイアログのイベント条件設定エリアに設定することができます。このエリアの左側のマークは、イベント・リンク条件の使用状況を示します（「表5-18 イベント・アイコン」参照）。また、灰色のLマークは、そのイベント・リンク条件が編集途中で未登録の状態を示しています。</p>
Phase1: Phase2: Phase3: Phase4:	<p>イベント条件およびイベント検出の順序を設定します。 イベント・リンクは4段まで設定可能です。ただし、途中でディスエーブル条件を検出した場合には、今まで成立したイベント条件は初期化され最初のイベント条件から検出を始めます。なお、リンク条件とディスエーブル条件が同時に検出された場合、ディスエーブル条件が優先されます。 順序は、Phase 1 → Phase 2 → Phase 3 → Phase 4 の順に設定します。なお、"Phase 4"まで設定する必要はなく、その場合には、最終Phaseに設定されたイベント条件を検出したところで、イベントを発生させます。また、"Phase1"のみにイベント条件を設定したり、同じイベント条件を複数のPhaseに設定することも可能です。 各エリアに設定できるイベント条件数、およびダイアログ全体で設定できるイベント条件の個数は、「表6-21 イベント・リンク・ダイアログでのイベント設定数」を参照してください。 イベント条件の設定は、イベント・マネージャ・エリアから、設定するイベント・アイコンをドラッグして、このエリア上にドロップすることにより簡単に行えます。詳細は、「5.12.3 各種イベント条件の設定」を参照してください。</p>
Disable:	<p>今まで成立したイベント条件を、無効にするイベント条件を設定します。 このエリアに設定できるイベント条件数、およびダイアログ全体で設定できるイベント条件の個数は、「表6-21 イベント・リンク・ダイアログでのイベント設定数」を参照してください。 イベント条件の設定は、イベント・マネージャ・エリアから、設定するイベント・アイコンをドラッグして、このエリア上にドロップすることにより簡単に行えます。詳細は、「5.12.3 各種イベント条件の設定」を参照してください。</p>
Pass Count: [IECUBE]	<p>パス・カウント条件（ユーザ・プログラム実行中に、このイベント・リンク条件が何回一致したら条件を成立させるか）を設定します。 省略時は1を設定したことになります（条件一致とともに条件成立）。</p>

表6-21 イベント・リンク・ダイアログでのイベント設定数

接続 IE		各エリア	合計（実行／アクセス）
[IECUBE]		10	14 (8/6)
[MINICUBE]	Nx85ET (RCU0+TEU+TRCU)	10 <sup>*a</sup>	12 (8/4)
	Nx85E901 (RCU0), RCU1	1 <sup>*b</sup>	2 <sup>*c</sup>
[MINICUBE2]	デバッグ機能あり	1 <sup>*b</sup>	2 <sup>*c</sup>
	デバッグ機能なし	-	-

\*a Phase1 - 4 に設定可能（ただし実行前イベントは設定不可）

\*b Phase1 と Phase2 にのみ設定可能（必ず2段に設定）

\*c 実行前イベントのみ

## (2) イベント・マネージャ・エリア

登録されているイベント一覧を表示しています。

このエリア上のイベント・アイコンを各種イベント設定ダイアログのイベント設定エリアにドラッグ&ドロップすることにより、容易に各種イベント条件の設定が行えます（「5.12.3 各種イベント条件の設定」参照）。

このエリアは、各種イベント関連ダイアログで共通です（「ブレーク・ダイアログ」の「(2) イベント・マネージャ・エリア」参照）。

## (3) 機能ボタン（イベント条件の内容表示等）

イベント条件設定エリアに置かれたイベント条件の内容表示／削除、およびイベント・マネージャ・エリアの表示／非表示等の操作を行います（「ブレーク・ダイアログ」の「(3) 機能ボタン（イベント条件の内容表示等）」参照）。

## (4) 機能ボタン（イベント条件の登録／削除／有効化／無効化）

イベントの登録／削除、および有効化／無効化等の操作を行うボタンです。

指定したイベント条件のイベントは、<OK>（または<Set>）ボタンを押すことにより登録されます。

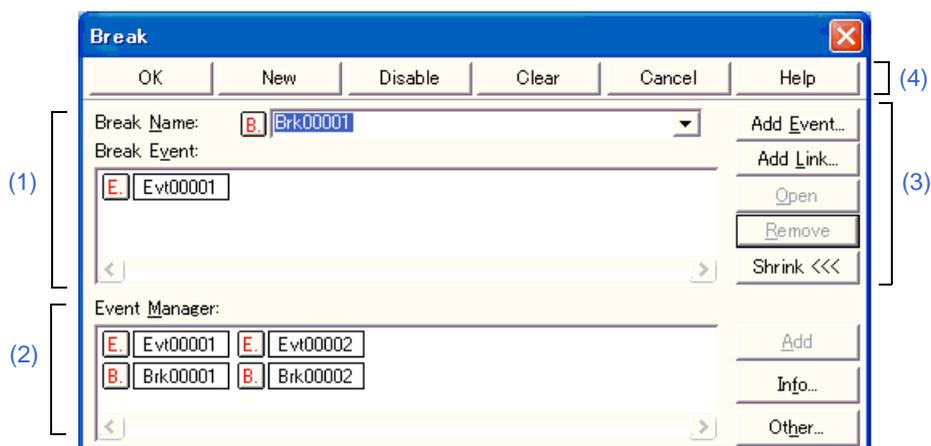
OK	編集中のイベント条件があれば、自動的に登録してこのダイアログをクローズします。 <b>選択モードの場合</b> イベント条件を選択して、呼び出し元の設定ダイアログ（タイトル・バーに表示）に戻ります。あらかじめ、そのダイアログをオープンしていた場合には、選択モードを通常モードに戻すだけで、このダイアログはクローズしません。それ以外の場合には、このダイアログをクローズします。
New	このダイアログのイベント条件を新規作成状態にします。 イベント名を自動生成して新規のイベント条件を用意します。
Set	各種イベント条件を登録します。イベント登録後、ダイアログはクローズしないため、続けて登録可能です。 <b>選択モードの場合</b> イベント条件を選択します。編集中のイベント条件があれば、自動的に登録して選択します。

Enable/Disable	選択している各種イベント条件の有効 (Enable) /無効 (Disable) を切り替えます。 ただし、イベント条件、イベント・リンク条件はこの対象となりません。 イベント・アイコンのマーク部分をクリックすることでも同じ動作をします。
Clear	イベント条件の内容をクリアします。
Restore	編集したイベント条件の内容を元に戻します。 未登録のイベント条件を表示している場合にはイベント名以外を空欄、あるいはデフォルトの設定にします。
Cancel Close	このダイアログをクローズします。 編集中のイベント条件であっても、登録せずにダイアログをクローズします。
Help	このダイアログのオンライン・ヘルプを表示します。

## ブレーク・ダイアログ

ブレーク・イベント条件の登録, 設定, および表示を行います (「5.12 イベント機能」, 「5.4 ブレーク機能」参照)。  
ブレーク・イベント条件の登録, 設定は, このダイアログ上で各項目を設定し, <OK> ボタンをクリックすることにより行います (最大 256 個)。登録したブレーク・イベント条件は, イベント・マネージャで管理されます。  
なお, 同時に設定可能な (有効になる) ブレーク・イベント条件数には制限があります (「5.12.4 各種イベント条件の有効イベント数」参照)。

図 6 - 60 ブレーク・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

 ボタンをクリック, または [ イベント ] メニュー → [ ブレーク ... ] を選択

### 各エリア説明

#### (1) ブレーク・イベント条件設定エリア

Break Name:	<p>ブレーク・イベント名の設定を行います。 名前は最大 8 文字までの英数字を直接入力してください。 すでに作成してあるイベント条件の内容を表示するには, ドロップダウン・リストから選択します。 このエリアの左側のマークは, イベント条件の使用状況を示します (「表 5 - 18 イベント・アイコン」参照)。また, 灰色のマークは, そのイベント条件が編集集中で未登録の状態を示しています。このマークをクリックすることにより, イベント条件の有効/無効を切り替えることができます。</p>
-------------	--

Break Event:	ブレイク用のイベント条件を設定します。 このエリアに設定できるイベント条件、イベント・リンク条件の個数は、「表 6 - 22 条件設定エリアのイベント設定数」を参照してください。 イベント条件の設定は、イベント・マネージャ・エリアから、設定するイベント・アイコンをドラッグして、このエリア上にドロップすることにより簡単に行えます。詳細は、「5.12.3 各種イベント条件の設定」を参照してください。
--------------	---

表 6 - 22 条件設定エリアのイベント設定数

接続 IE		合計イベント条件（実行前／実行後／アクセス）	イベント・リンク条件
[IECUBE]		16 (2 / 8 / 6)	1
[MINICUBE]	Nx85ET (RCU0+TEU+TRCU)	14 (2 / 8 / 4)	-
	Nx85E901 (RCU0), RCU1	2	-
[MINICUBE2]	デバッグ機能あり	2	-
	デバッグ機能なし	-	-

## (2) イベント・マネージャ・エリア

登録されているイベント一覧を表示しています。

このエリア上のイベント・アイコンを各種イベント設定ダイアログのイベント設定エリアにドラッグ&ドロップすることにより、容易に各種イベント条件の設定が行えます（「5.12.3 各種イベント条件の設定」参照）。

このエリアは、各種イベント関連ダイアログで共通です。

<Add>	イベント・マネージャ・エリアで選択しているイベント条件、イベント・リンク条件をフォーカスのある設定エリアに追加設定します。	
<Info...>		<p>&lt;Sort by Name&gt; ... イベントを名前順に並べ変えます。          &lt;Sort by Kind&gt; ... イベントの種類順に並べ替えます。          &lt;Unsort&gt; ... 並べ替えを行わず、登録順に表示します。          &lt;Detail&gt; ... 詳細表示モードにします。          &lt;Overview&gt; ... 一覧表示モードにします。          &lt;Cancel&gt; ... このダイアログをクローズします（ESC キーと同様）。</p>
<Other...>		<p>各ボタンを押すことにより、各種イベント条件設定ダイアログを新規のイベント名が設定された状態でオープンします。各種イベント条件設定ダイアログのオープン後はこのダイアログはクローズされます。          &lt;Manager...&gt; ボタンにより、イベント・マネージャをオープンします。          &lt;Cancel&gt; ボタンにより、このダイアログをクローズします。</p>

**(3) 機能ボタン（イベント条件の内容表示等）**

イベント条件設定エリアに置かれたイベント条件の内容表示／削除、およびイベント・マネージャ・エリアの表示／非表示等の操作を行います。

このボタンは、イベント関連ダイアログで共通です。

Add Event...	イベント・ダイアログを選択モードでオープンし、設定するイベント条件を選択、または新規作成します。追加設定されるエリアは、<Add Event...> ボタンを押した際に選択されていたエリアになります。
Add Link...	イベント・リンク・ダイアログを選択モードでオープンし、設定するイベント・リンク条件を選択、または新規作成します。追加設定されるエリアは、<Add Link...> ボタンを押した際に選択されていたエリアになります。
Open	選択しているイベント条件（1つ）に対応する各種イベント設定ダイアログをオープンします。各設定ダイアログには、選択したイベント条件の内容が表示されます。イベント・アイコンのダブルクリック、および Enter キーも同じ動作をします。
Remove Delete	選択しているイベント条件を削除します。 イベント条件、またはイベント・リンク条件を削除する時、そのイベントが各種イベント条件として使用されている場合には、エラーとなり削除できません。
Expand >>> Shrink <<<	イベント・マネージャ・エリアを表示／非表示します。 ダイアログのサイズが拡張／縮小されます。

**(4) 機能ボタン（イベント条件の登録／削除／有効化／無効化）**

イベントの登録／削除、および有効化／無効化等の操作を行うボタンです。

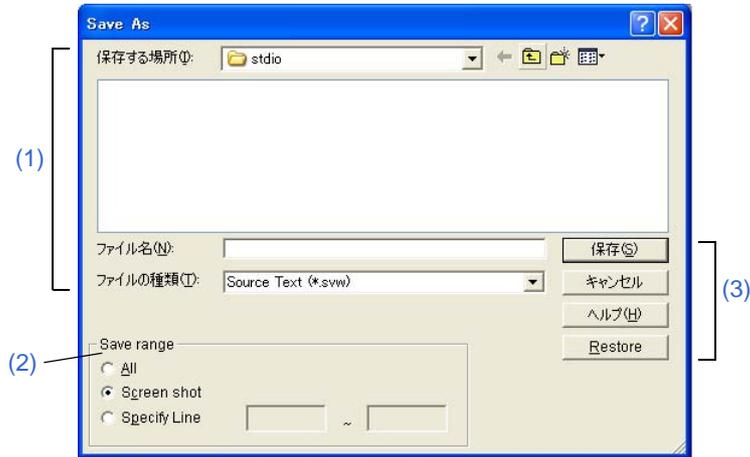
指定したイベント条件のイベントは、<OK>（または<Set>）ボタンを押すことにより登録、設定（有効化）されます。このボタンは、イベント関連ダイアログで共通です。

OK	編集中のイベント条件、各種イベント条件がある場合、それらを登録してこのダイアログをクローズします。 各種イベント条件の場合は、登録と同時に有効になります。
New...	このダイアログのイベント条件を新規作成状態にします。 イベント名を自動生成して新規のイベント条件を用意します。
Set	イベント条件、各種イベント条件を登録します。イベント登録後、ダイアログはクローズされないため、続けて登録可能です。 各種イベント条件の場合は、登録と同時に有効になります。
Enable/Disable	選択している各種イベント条件の有効（Enable）／無効（Disable）を切り替えます。 ただし、イベント条件、イベント・リンク条件はこの対象となりません。 イベント・アイコンのマーク部分をクリックすることでも同じ動作をします。
Clear	イベント条件の内容をクリアします。
Restore	編集したイベント条件の内容を元に戻します。 未登録のイベント条件を表示している場合にはイベント名以外を空欄、あるいはデフォルトの設定にします。
Cancel Close	このダイアログをクローズします。 編集中のイベント条件であっても、登録せずにダイアログをクローズします。
Help	このダイアログのオンライン・ヘルプを表示します。

## 表示ファイル・セーブ・ダイアログ

カレント・ウィンドウの表示情報を表示ファイルに保存するダイアログです（「5.15.2 ウィンドウの表示情報（表示ファイル）」参照）。

図 6 - 61 表示ファイル・セーブ・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

保存するウィンドウをカレント・ウィンドウにし、[ファイル]メニュー→[名前を付けて保存...]を選択

### 各エリア説明

#### (1) 保存ファイル設定エリア

保存する場所：	ファイル名を指定するエリアです。一覧から選択するか、直接入力により指定します。
ファイル名：	指定可能文字数は拡張子を含めて 257 文字までです。
ファイルの種類：	保存するファイルの種類（拡張子）を指定します（「表 5 - 21 表示ファイルの種類」参照）。カレント・ウィンドウに対応した拡張子が表示されます。

**(2) セーブ範囲指定エリア**

100 行／100 フレーム／256 バイト以上の範囲を指定した場合、メッセージ・ダイアログによりセーブ状況の把握が可能です。また、<Stop> ボタンによりセーブの中止も可能です。

Save range	セーブ範囲を指定するエリアです。 このエリアはカレント・ウィンドウが以下の場合、表示されます。 - ソース・テキスト・ウィンドウ - 逆アセンブル・ウィンドウ - メモリ・ウィンドウ - トレース・ウィンドウ	
	All	最初の行から最後の行まで、全領域を保存する場合、選択します。
	Screen shot	画面の最上行から最下行までの画面に見える領域を保存する場合、選択します。ただし、ソース・テキスト・ウィンドウが混合表示している場合は、画面に見える領域を含むソース行から保存されます。
	Specify Line Specify Frame Specify Address	保存する領域の開始行／終了行を指定する場合、選択します。 開始行／終了行を省略したときは、それぞれ、最初の行／最後の行を指定したとみなします。カレント・ウィンドウに応じて次のいずれかが表示されます。
	Specify Line	保存する行番号の範囲を指定します。数値入力時のデフォルトの進数は 10 進です。 ソース・テキスト・ウィンドウが混合表示モードのときは、指定した行の混合表示部分も一緒に保存されます。
	Specify Frame [IECUBE]	保存するフレームの範囲を指定します（「表 6-16 フレーム番号の指定形式」参照）。 数値入力時のデフォルトの進数は 10 進です。
	Specify Address	保存するアドレス範囲を指定します。 アドレス指定はシンボルや式での指定も可能です（「表 5-6 シンボルでの指定方法」参照）。 数値入力時のデフォルトの進数は 16 進です。

**(3) 機能ボタン**

保存	選択したファイルにカレント・ウィンドウの表示情報を保存します。セーブ後、ダイアログをクローズします。
キャンセル	何もせずにこのダイアログをクローズします。
ヘルプ	このダイアログのオンライン・ヘルプを表示します。
Restore	このダイアログをオープンする前の状態に戻します。

## 表示ファイル・ロード・ダイアログ

表示ファイルをロードするダイアログです（「5.15.2 ウィンドウの表示情報（表示ファイル）」参照）。

表示ファイルをロードすることにより、参照用ウィンドウ（スタティック状態のソース・テキスト・ウィンドウ）がオープンし、保存時の表示情報が表示されます。

ロードするファイルによって、オープンするウィンドウ／状態が次のように異なります。

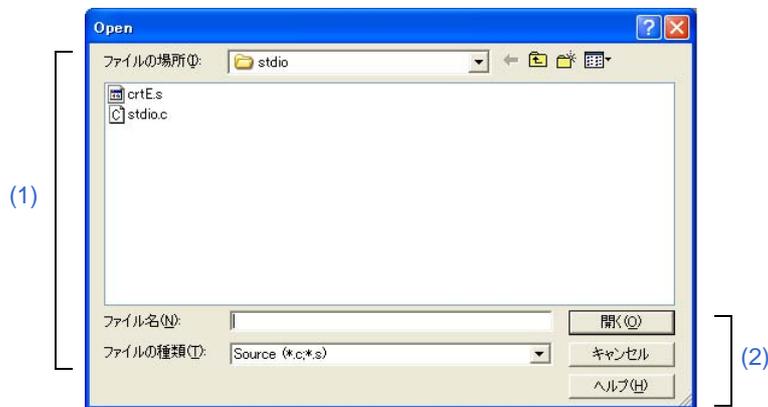
- シンボル情報が読み込まれているソース・ファイルをロードした場合

アクティブ状態のソース・テキスト・ウィンドウがあればスタティック状態で、なければアクティブ状態のソース・テキスト・ウィンドウとしてオープンされます。

- シンボル情報が読み込まれていないソース・ファイル、または表示ファイルをロードした場合

スタティック状態のソース・テキスト・ウィンドウにテキスト形式のファイルとしてオープンされます。

図 6 - 62 表示ファイル・ロード・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法



ボタンをクリック、または[ファイル]メニュー→[開く...]を選択

## 各エリア説明

### (1) ロード・ファイル設定エリア

ファイルの場所：	ロードするファイル名を指定します。一覧からの選択、または直接入力により指定します。
ファイル名：	指定できる文字数は、拡張子を含めて 257 文字までです。
ファイルの種類：	ロードするファイルの種類（拡張子）を指定します（「 <a href="#">表 5 - 21 表示ファイルの種類</a> 」参照）。

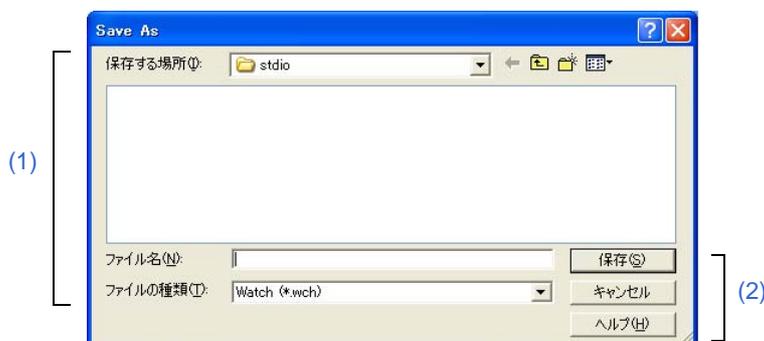
### (2) 機能ボタン

開く	選択したファイルをロードします。ロード後、ダイアログをクローズします。
キャンセル	何もせずにこのダイアログをクローズします。
ヘルプ	このダイアログのオンライン・ヘルプを表示します。

## 環境設定ファイル・セーブ・ダイアログ

カレント・ウィンドウの設定内容を設定ファイルに保存するダイアログです（「5.15.3 ウィンドウの設定情報（設定ファイル）」参照）。

図 6 - 63 環境設定ファイル・セーブ・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

保存するウィンドウをカレント・ウィンドウにし、[ファイル]メニュー→[環境]→[名前を付けて保存...]を選択

### 各エリア説明

#### (1) 保存ファイル設定エリア

保存する場所：	ファイル名を指定するエリアです。一覧から選択するか、直接入力により指定します。
ファイル名：	指定可能文字数は拡張子を含めて 257 文字までです。
ファイルの種類：	保存するファイルの種類（拡張子）を指定します（「表 5 - 22 設定ファイルの種類」参照）。カレント・ウィンドウに対応した拡張子が表示されます。

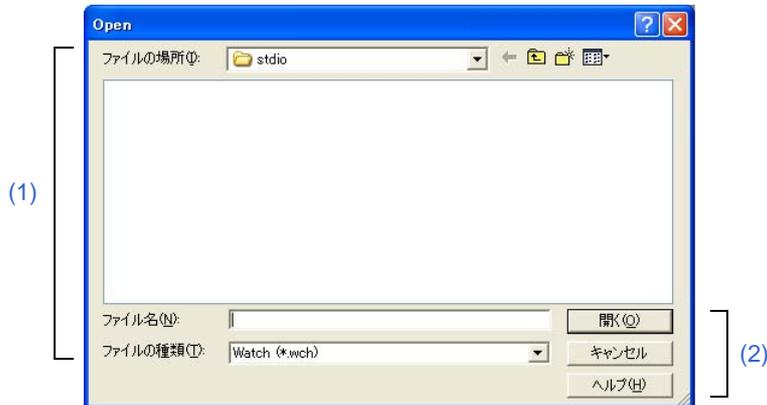
#### (2) 機能ボタン

保存	選択したファイルにカレント・ウィンドウの設定情報を保存します。セーブ後、ダイアログをクローズします。
キャンセル	何もせずにこのダイアログをクローズします。
ヘルプ	このダイアログのオンライン・ヘルプを表示します。

## 環境設定ファイル・ロード・ダイアログ

設定ファイルをロードするダイアログです（「5.15.3 ウィンドウの設定情報（設定ファイル）」参照）。  
設定ファイルをロードすることにより、対象ウィンドウがオープンし保存時の設定情報が復元されます。

図 6 - 64 環境設定ファイル・ロード・ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

[ファイル]メニュー → [環境] → [開く ...] を選択

### 各エリア説明

#### (1) ロード・ファイル設定エリア

ファイルの場所:	ロードするファイル名を指定します。一覧からの選択、または直接入力により指定します。
ファイル名:	指定できる文字数は、拡張子を含めて 257 文字までです。
ファイルの種類:	ロードするファイルの種類（拡張子）を指定します（「表 5 - 22 設定ファイルの種類」参照）。

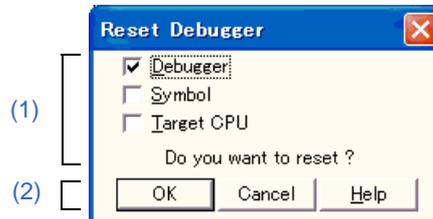
#### (2) 機能ボタン

開く	選択したファイルをロードします。ロード後、ダイアログをクローズします。
キャンセル	何もせずにこのダイアログをクローズします。
ヘルプ	このダイアログのオンライン・ヘルプを表示します。

## リセット確認ダイアログ

ID850QB 本体、CPU、およびシンボル情報の初期化を行います。

図 6 - 65 リセット確認ダイアログ



- [オープン方法](#)
- [各エリア説明](#)

### オープン方法

[ファイル]メニュー→[デバッガリセット...]を選択

### 各エリア説明

#### (1) リセット対象選択エリア

初期化する対象を指定するエリアです。チェックした項目の初期化を行います。

Debugger	ID850QB を初期化（デフォルト）
Symbol	シンボル情報を初期化
Target CPU	CPU を初期化

#### (2) 機能ボタン

OK	設定に従い初期化を行います。
Cancel	変更を取り消してこのダイアログをクローズします。
Help	このダイアログのオンライン・ヘルプを表示します。

## 終了確認ダイアログ

ID850QB 終了時、現在のデバッグ環境をプロジェクト・ファイルに保存してから終了するか、保存せずに終了するかを選択します（「5.15.1 デバッグ環境（プロジェクト・ファイル）」参照）。

デバッガ・オプション設定ダイアログで、終了確認ダイアログをオープンせずに ID850QB を終了するように設定することもできます。

図 6 - 66 終了確認ダイアログ



- オープン方法
- 機能ボタン

### オープン方法

- [ファイル]メニュー → [終了]を選択
- Windows を終了させるタスク・リストでアプリケーションを終了させる等の強制終了を行った時

### 機能ボタン

はい	現在のデバッグ環境をプロジェクト・ファイルに保存した後、すべてのウィンドウをクローズし、ID850QB を終了します。 プロジェクト・ファイル名が未定の場合には、プロジェクト・ファイル・セーブ・ダイアログがオープンします。また、プロジェクト・ファイル・セーブ・ダイアログで <キャンセル> ボタンを選択した際は、プロジェクト・ファイルの保存も ID850QB の終了も行いません。 (デバッグ・オペレーション中にプロジェクト・ファイルをロード、またはセーブした場合には、このボタンがデフォルトになります。)
いいえ	すべてのウィンドウをクローズし、ID850QB をクローズします。 (デバッグ・オペレーション中にプロジェクト・ファイルのロード、またはセーブをしなかった場合には、このボタンがデフォルトになります。)
キャンセル	何もせずにこのダイアログをクローズします。

## バージョン表示ダイアログ

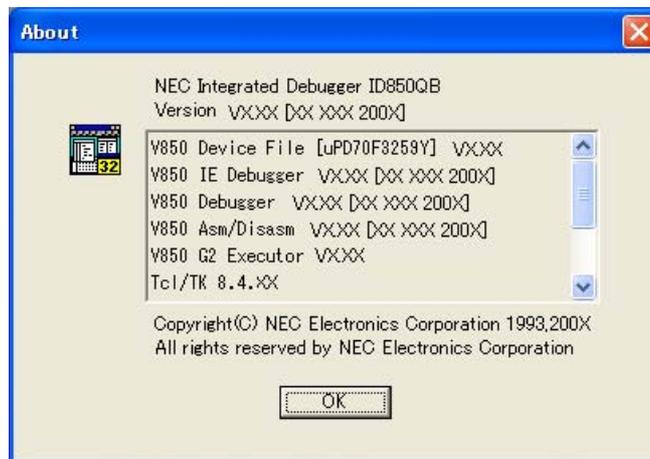
ID850QB のバージョン情報を表示します（西暦は 4 桁表示）。

**参考** バージョン情報は、ダイアログ上でコンテキスト・メニュー→ [すべて選択してコピー (&C)] を選択することにより、クリップ・ボード上にコピー可能です。

表示されるバージョン情報は次の通りです。

- 製品バージョン
- デバイス・ファイルのバージョン
- GUI のバージョン
- デバッガ DLL のバージョン
- アセンブラ DLL のバージョン
- エグゼキュータのバージョン
- Tcl/Tk のバージョン
- インサーキット・エミュレータのプロダクト ID とプロダクト・バージョン

図 6 - 67 バージョン表示ダイアログ



- オープン方法

### オープン方法

[ヘルプ]メニュー→[バージョン情報...]を選択

**参考** コンフィギュレーション・ダイアログ上の<About...>ボタンをクリックすることでもオープン可能です。

---

## コンソール・ウィンドウ

---

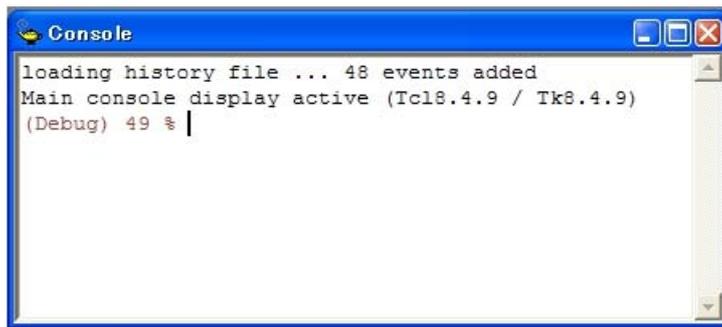
ID850QB を制御するためのコマンドを入力するウィンドウです。

キーバインドは、Emacs ライクとなっているため、コンソール・ウィンドウがアクティブ状態の時は、アクセラレータ・キーを受け付けません。ただし、F1 キーは、コンソール・ウィンドウのオンライン・ヘルプを表示します。

コンソール・ウィンドウが開いているときは、<OK> ボタンのみエラー・メッセージはコンソール・ウィンドウに表示されます。また、コンソール・ウィンドウを閉じた時に、コマンド履歴が保存されます。

なお、コマンド仕様に関しては、「[第7章 コマンド・リファレンス](#)」を参照してください。

図 6 - 68 コンソール・ウィンドウ



- [オープン方法](#)

### オープン方法

---

[ブラウザ]メニュー → [コンソール] を選択

## ソース・ファイル選択ダイアログ

ソース指定ダイアログへ設定するファイルを選択します。

**備考** 起動後、はじめてこのダイアログをオープンした場合には、ソース・パスで最初に指定しているフォルダを表示します。2 度目以降の場合には、以前に表示したフォルダを記憶して表示します。ただし、<キャンセル>した場合には記憶しません。

図 6 - 69 ソース・ファイル選択ダイアログ



- オープン方法
- 各エリア説明

### オープン方法

対象ダイアログ上の <Browse...> ボタンをクリック

### 各エリア説明

#### (1) オープン・ファイル設定エリア

ファイルの場所:	オープンするファイル名を指定します。一覧からの選択、または直接入力により指定します。
ファイル名:	指定できる文字数は、拡張子を含めて 257 文字までです。
ファイルの種類:	オープンするファイルの種類 (拡張子) を指定します (「表 5 - 5 表示可能なファイルの種類」参照)。

#### (2) 機能ボタン

開く	選択したファイルを設定します。設定後、ダイアログをクローズします。
キャンセル	何もせずにこのダイアログをクローズします。
ヘルプ	このダイアログのオンライン・ヘルプを表示します。

## 第7章 コマンド・リファレンス

この章では、ID850QB が持つコマンドの機能の詳細について解説します。

- [コマンド・ライン規約](#)
- [コマンド一覧](#)
- [エイリアス一覧](#)
- [変数一覧](#)
- [パッケージ一覧](#)
- [キー・バインド](#)
- [拡張ウィンドウ](#)
- [コールバック・プロシージャ](#)
- [フック・プロシージャ](#)
- [関連ファイル](#)
- [注意事項](#)
- [コマンドの説明](#)
- [サンプル \(電卓スクリプト\)](#)

## 7.1 コマンド・ライン規約

コマンド・ラインの指定には、次の規約があります。

- コマンド・ラインには、コマンド名、オプション、引数を指定します。
- 単語の区切りには、空白文字（スペース、またはタブ）を使います。
- 行末には、改行文字、またはセミコロンを使います。
- コマンド名とオプションは識別可能なところまで入力すると認識します。
- スクリプト内では、コマンド名をすべて入力しなければなりません。

### コマンド形式

```
command -options arg1 arg2 arg3 ...
```

## 7.2 コマンド一覧

表 7-1 デバッガ制御コマンド一覧

コマンド名	機能
address	アドレス式の評価
assemble	逆アセンブル／ライン・アセンブル
batch	バッチ実行（エコー付き）
breakpoint	ブレーク・ポイントの設定／削除
cache	キャッシュの設定 [MINICUBE]
dbgexit	ID850QB の終了
dbgopt	デバッガ・オプションの選択
download	ファイルのダウンロード
erase	内蔵フラッシュ・メモリの消去
extwin	拡張ウィンドウの作成
finish	関数から戻る
flop	内蔵フラッシュ・メモリに関連する操作
go	継続して実行
help	ヘルプの表示
hook	フックの設定
ie	IE レジスタの表示／設定
inspect	シンボル・インスペクト
jump	ウィンドウへのジャンプ
map	メモリ・マッピングの設定／削除
mdi	拡張ウィンドウの設定
memory	メモリの表示／設定

コマンド名	機能
<code>module</code>	ファイル一覧, 関数一覧の表示
<code>next</code>	プロシージャ・ステップ
<code>refresh</code>	ウインドウの再描画
<code>register</code>	レジスタ値, I/O レジスタ値の表示/設定
<code>reset</code>	リセット
<code>run</code>	CPU リセット & 実行
<code>step</code>	ステップ実行
<code>stop</code>	実行停止
<code>upload</code>	アップロード
<code>version</code>	バージョンの表示
<code>watch</code>	変数値の表示/設定
<code>where</code>	スタックのトレース
<code>wish</code>	Tclet の起動
<code>xcoverage</code>	カバレッジの操作 [IECUBE]
<code>xtime</code>	タイマの操作 [IECUBE]
<code>xtrace</code>	トレーサの操作 [IECUBE]

表 7-2 コンソール/Tcl コマンド一覧

コマンド名	機能
<code>alias</code>	別名作成
<code>cd</code>	ディレクトリ変更
<code>clear</code>	画面消去
<code>echo</code>	エコー
<code>exit</code>	閉じる/終了する
<code>history</code>	履歴表示
<code>ls</code>	ファイル表示
<code>pwd</code>	ディレクトリ確認
<code>source</code>	バッチ実行
<code>time</code>	コマンドの時間測定
<code>tkcon</code>	コンソール制御
<code>unalias</code>	別名削除
<code>which</code>	コマンド・パスまたは別名の表示
その他	Tcl / Tk 8.4 に準拠

## 7.3 エイリアス一覧

コマンドは、以下のファイル (bin/idtcl/aliases.tcl) にエイリアス定義することにより別名での指定が可能です。デフォルトでは、以下のように記述されています。

なお、このファイルは、エディタにより編集可能です。

表 7-3 aliases.tcl ファイルの内容

```
alias a assemble
alias b breakpoint
alias g go
alias i step -i
alias j jump
alias l download
alias m memory
alias n next
alias r run
alias s step
alias w watch
```

## 7.4 変数一覧

表 7-4 変数一覧

変数	機能
dcl (chip)	チップ名 read only
dcl (prjfile)	プロジェクト・ファイル名 read only
dcl (srcpath)	ソース・パス read only
dcl (ieid)	IE タイプ read only
dcl (iestat)	IE ステータス read only
dcl (bkstat)	ブレーク・ステータス read only
env (LANG)	言語
dcl_version	Dclバージョン read only

## 7.5 パッケージ一覧

表 7-5 パッケージ一覧

パッケージ	機能
tcltest	回帰テスト
cwind	ウインドウ自動制御
BWidget	ツールキット
tcllib	Tcl ライブラリ
mclistbox	マルチカラム・リストボックス
combobox	コンボ・ボックス

## 7.6 キー・バインド

- tcsh + Emacs ライク
- コマンド名の補完 [Tab]
- ファイル名の補完 [Tab]
- HTML ヘルプ [F1]

## 7.7 拡張ウインドウ

Tk を使うと拡張ウインドウを作成することができます。

拡張ウインドウは, '.' の代わりに '.dcl' をルートとして Widget を配置します。

以下のスクリプト・ファイルを bin/idtcl/tools/ に置くことにより, [ブラウザ]メニュー→[その他]選択時に拡張ウインドウが追加されます。

拡張ウインドウには拡張ウインドウ専用の mdi コマンドが追加されます。

```
# Sample.tcl
wm protocol .dcl WM_DELETE_WINDOW { exit }
mdi geometry 100 50
button .dcl.b -text Push -command exit
pack .dcl.b
```

**注意** 拡張ウインドウでは, MDI ウインドウの制約から Tk の menu コマンドを使うことはできません。

## 7.8 コールバック・プロシージャ

拡張ウィンドウは、非同期のメッセージによって呼び出される `dcl_asyncproc` プロシージャを持つことができます。

```
proc dcl_asyncproc {mid} {
  if {$mid == 19} {
    redraw
  }
}
```

`dcl_asyncproc` プロシージャの引数には、非同期メッセージ ID が渡されます。

メッセージ ID には以下のものがあります。

表 7-6 メッセージ ID

メッセージ ID	意味
9	コンフィギュレーション変更後
10	イベント登録後
11	イベント削除後
12	実行開始前
13	ブレーク後
14	CPU リセット後
15	ID850QB リセット後
17	拡張オプション変更後
18	デバッガ・オプション変更後
19	ダウンロード後
20	メモリ、またはレジスタ変更後
36	トレーサ開始前 [IECUBE]
37	トレーサ停止後 [IECUBE]
40	タイマ開始前 [IECUBE]
41	タイマ停止後 [IECUBE]
42	トレース・クリア後 [IECUBE]
45	シンボル・リセット後
46	RRM 機能からトレース機能、およびトレース機能から RRM 機能への変更後 [IECUBE]

## 7.9 フック・プロシージャ

フック・プロシージャを使用して、ID850QB にフックを設定できます。

フック・プロシージャには以下のものがあります。

- BeforeDownload (ダウンロード前のフック)
- AfterDownload (ダウンロード後のフック)
- AfterCpuReset (ブレーク中の CPU リセット後のフック)
- BeforeCpuRun (実行開始前のフック)
- AfterCpuStop (ブレーク後のフック)

フック・プロシージャを使用することで、プログラムのダウンロード前や CPU リセット後に、レジスタの値を変更することが可能です。

下記に手順の具体例を示します。なお、フックは ID850QB を終了させるまで有効となります。

### (1) [ID850QB 制御コマンドでフックを設定する場合]

- 1) エディタでスクリプト・ファイルを作成します。
- 2) ID850QB を起動し、[ブラウザ]メニュー→[コンソール]を選択し、[コンソール・ウインドウ](#)をオープンします。
- 3) ウインドウ上で、以下のようにスクリプト・ファイルを実行すると、スクリプト・ファイル内のフックが設定されます。

```
%hook test.tcl
```

### (2) [プロジェクト・ファイルのダウンロード時にフックを設定する場合]

- 1) エディタでスクリプト・ファイル<sup>注</sup>を作成します。
- 2) ID850QB を起動し、test.prj を読み込んでください。スクリプト・ファイル内のフックが設定されます。

```
proc BeforeDownload {} {
    register MM 0x7
    register PMC8 0xff
    register PMC9 0xff
    register PMCX 0xe0
}

proc AfterCpuReset {} {
    register MM 0x7
    register PMC8 0xff
    register PMC9 0xff
    register PMCX 0xe0
}
```

**注** スクリプト・ファイル名は、プロジェクト・ファイル名と同じにしてください。

例)

test.prj に対応したスクリプト・ファイルは test.tcl になります。

test.prj, test.pri, test.tcl は同じフォルダへ置いてください。

## 7.10 関連ファイル

表 7-7 関連ファイル一覧

ファイル名	機能
aliases.tcl	コンソールを開く時に実行します。 デフォルトの alias 等を設定します。
プロジェクト・ファイル名.tcl	プロジェクト・ファイルを開く時に実行します。 以下のフックが使えます。 BeforeDownload AfterDownload AfterCpuReset BeforeCpuRun AfterCpuStop
ロード・モジュール・ファイル名.tcl	ロード・モジュール・ファイルをダウンロードする時に実行します。 以下のフックが使えます。 BeforeDownload AfterDownload AfterCpuReset BeforeCpuRun AfterCpuStop

## 7.11 注意事項

- ファイル・パスのセパレータ文字はスラッシュ (/) を使います。
- コンソールが開いている時は、エラー・メッセージをコンソールに出力します。
- コマンドを強制終了させたい場合は、コンソールを閉じてください。
- 外部コマンド (DOS コマンド) の実行は、デフォルトで OFF になっています。

## 7.12 コマンドの説明

この節では、各コマンドについて次のような形式で解説します。

### コマンド名

---

名称を英語 - 日本語で記述します。

### 入力形式

---

コマンドの入力形式を記述します。

イタリック書体の引数は、ユーザが該当値を記述する引数であることを、"?" で囲まれた引数は省略可能であることを示しています。

なお、コマンド名とオプションは識別可能なところまで入力すると認識します。

### 別名

---

別名定義されている場合、そのコマンド名を示します。詳細は「[7.3 エイリアス一覧](#)」を参照してください。

### 機能説明

---

コマンドの機能を説明します。

### 使用例

---

コマンドの使用例を示します。

## address

---

---

address - アドレス式の評価

### 入力形式

---

**address** *expression*

### 機能説明

---

*expression* で指定したアドレス式をアドレスに変換します。

### 使用例

---

(IDCON) 1 % address main  
0xaa  
(IDCON) 2 % address main+1  
0xab

## assemble

assemble - 逆アセンブル/ライン・アセンブル

### 入力形式

**assemble** *?options? address ?code?*

### 機能説明

*code* で指定した文字列を、*address* で指定したアドレスからアセンブルします。

*address* に '.' を指定した場合は、直前のアセンブルの続きのアドレスと解釈します。

*code* を省略した場合は、*address* で指定したアドレスから逆アセンブルします。

*options* には、以下のものがあります。アセンブルの場合は無視します。

<b>-code</b>	命令コードも表示します。アセンブルの場合は無視します。
<b>-number</b> <i>number</i>	<i>number</i> 行表示します。アセンブルの場合は無視します。

### 使用例

```
(IDCON) 1 % assemble -n 5 main
0x000000aa B7      PUSH HL
0x000000ab B1      PUSH AX
0x000000ac 891C    MOVW AX,SP
0x000000ae D6      MOVW HL,AX
0x000000af A100    MOV A,#0H
(IDCON) 2 % assemble main mov a,b
(IDCON) 3 % assemble . mov a,b
```

## batch

---

---

batch - バッチ実行 ( エコー付き )

### 入力形式

---

**batch** *scriptname*

### 機能説明

---

*scriptname* で指定したファイルを画面に表示しながら一括実行します。

ネスト可能です。

### 使用例

---

```
(IDCON) 1 % clear  
(IDCON) 2 % batch bat_file.tcl  
(IDCON) 3 % tkcon save a:/log.txt
```

# breakpoint

breakpoint - ブレーク・ポイントの設定／削除

## 入力形式

**breakpoint** *?options? ?address1? ?address2?*

**breakpoint** -delete *brkno*

**breakpoint** -enable *brkno*

**breakpoint** -disable *brkno*

**breakpoint** -information

## 別名

b

## 機能説明

*options* と *address* で指定したブレーク・ポイントを操作します。

ブレーク・ポイントが正常に設定されるとブレーク・ポイント番号が返されます。

*options* には、以下のものがあります。

<b>-software</b>	ソフトウェア・ブレークを指定します。
<b>-hardware</b>	ハードウェア・ブレークを指定します (デフォルト)。
<b>-execute</b>	<i>address</i> 実行ブレークを設定します (デフォルト)。
<b>-beforeexecute</b>	<i>address</i> 実行前ブレークを設定します。
<b>-read</b>	<i>address</i> データ・リード・ブレークを設定します。
<b>-write</b>	<i>address</i> データ・ライト・ブレークを設定します。
<b>-access</b>	<i>address</i> データ・アクセス・ブレークを設定します。
<b>-size</b> <i>size</i>	アクセス・サイズ (8, 16, 32 のいずれか) を設定します (単位 : ビット)。[IECUBE]
<b>-data</b> <i>value</i>	データ条件を設定します。
<b>-datamask</b> <i>value</i>	データ・マスクを設定します。
<b>-information</b>	ブレーク・ポイントの一覧を表示します。
<b>-delete</b>	指定した番号のブレーク・ポイントを削除します。
<b>-disable</b>	指定した番号のブレーク・ポイントを無効にします。
<b>-enable</b>	指定した番号のブレーク・ポイントを有効にします。

## 使用例

---

(IDCON) 1 % breakpoint main

1

(IDCON) 2 % breakpoint -i

1 Brk00001 enable rammon.c#17

(IDCON) 3 % breakpoint -software sub

2

(IDCON) 4 % breakpoint -i

1 Brk00001 enable rammon.c#17

2 Brk00001 enable rammon.c#8

(IDCON) 5 % breakpoint -disable 2

(IDCON) 6 % breakpoint -i

1 Brk00001 enable rammon.c#17

2 Brk00001 disable rammon.c#8

(IDCON) 7 % breakpoint -delete 1

2 Brk00001 disable rammon.c#8

# cache

[MINICUBE]

cache - キャッシュの設定

## 入力形式

**cache****cache** *config* *?-icache itype?* *?-dcache dtype?***cache** *clear* *?-icache bool?* *?-dcache bool?*

## 機能説明

IキャッシュとDキャッシュの設定を行います。

サブコマンドに *config* を指定した際は、キャッシュ・タイプの設定を行います。

サブコマンドに *clear* を指定したときは、EXECによるキャッシュ・クリアを行うかどうかの設定を行います (デフォルトはクリア)。

サブコマンド以下を省略した場合は、現在の状態を表示します。

*itype* は、以下の中から選択します。

NB85E212	NB85E212 を使用
NB85E213	NB85E213 を使用
nouse	使用しない (デフォルト)

*dtype* は、以下の中から選択します。

NB85E252	NB85E252 を使用
NB85E263	NB85E263 を使用
nouse	使用しない (デフォルト)

*bool* は、以下の中から選択します。

0, false, off のいずれか	しない
1, true, on のいずれか	する

## 使用例

```
(IDCON) 1 % cache config -i NB85E212 -d NB85E252
(IDCON) 2 % cache
i-cache: NB85E212
d-cache: NB85E252
(IDCON) 3 % cache clear -i false
(IDCON) 4 % cache
i-cache: NB85E212 (persist)
d-cache: nouse
```

## dbgexit

---

---

dbgexit - ID850QB の終了

### 入力形式

---

`dbgexit ?options?`

### 機能説明

---

ID850QB を終了します。

*options* には以下のものがあります。

<code>-saveprj</code>	ID850QB 終了時にプロジェクトを保存します。
-----------------------	---------------------------

### 使用例

---

(IDCON) 1 % `dbgexit -saveprj`

---

## dbgopt

---

---

dbgopt - デバッガ・オプションの選択

---

### 入力形式

---

dbgopt *options* *?value?*

---

### 機能説明

---

デバッガのオプションを選択します。*options* には以下のものがあります。

<code>-function ?func?</code> <b>[IECUBE]</b>	RRM 機能／トレース機能／カバレッジ機能を切り替えます。 <i>func</i> を省略した場合、現在の機能を表示します。 <i>func</i> は、以下の中から選択します。	
	rrm	RRM 機能を選択します。(デフォルト)
	trace	トレース機能を選択します。
	coverage	カバレッジ機能を選択します。

---

### 使用例

---

(IDCON) 1 % dbgopt ?function trace

## download

download - ファイルのダウンロード

### 入力形式

`download ?options? filename ?offset?`

### 別名

I

### 機能説明

*filename* で指定したファイルを *options* にしたがってダウンロードします（ロード・モジュール形式のファイルと HEX 形式のファイルは自動認識されます）。

*offset* を指定すると *offset* だけアドレスをずらします（バイナリ形式のデータの場合は、*offset* にロード開始アドレスを指定します）。

<b>-binary</b>	バイナリ形式のデータをダウンロードします。
<b>-coverage</b>	カバレッジ・データをダウンロードします。[IECUBE]
<b>-append</b>	追加ダウンロードします。
<b>-nosymbol</b>	ダウンロードします。シンボル情報は読み込みません。
<b>-symbolonly</b>	シンボル情報のみを読み込みます。
<b>-erase</b>	ダウンロード前にフラッシュ・メモリを全消去します（フラッシュ・メモリ搭載品のみ）。[MINICUBE]
<b>-reset</b>	ダウンロード後に CPU リセットをします。
<b>-information</b>	ダウンロード情報を表示します。

### 使用例

(IDCON) 1 % download test.lmf

---

## erase

---

---

erase - フラッシュ・メモリの消去

---

### 入力形式

---

erase

---

### 機能説明

---

フラッシュ・メモリを消去します。

*options* には以下のものがあります。

すべてのオプションを省略した場合には、コード・フラッシュ・メモリを消去します。

<b>-code</b>	コード・フラッシュ・メモリを消去します。
<b>-data</b>	データ・フラッシュ・メモリを消去します。
<b>-external</b>	外部空間にあるフラッシュ・メモリを消去します。

---

### 使用例

---

(IDCON) 1 % erase -code -data

## extwin

---

---

extwin - 拡張ウィンドウの作成

### 入力形式

---

`extwin scriptfile`

### 機能説明

---

`scriptfile` で拡張ウィンドウを作成します。

### 使用例

---

(IDCON) 1 % extwin d:/foo.tcl

## finish

---

---

finish - 関数から戻る

### 入力形式

---

finish

### 機能説明

---

現在の関数を呼び出したプログラムに戻るまで実行します。

### 使用例

---

(IDCON) 1 % finish

# flop

[MINICUBE]

flop - フラッシュ・メモリに関連する操作

## 入力形式

**flop** -init

**flop** -user *clock*

## 機能説明

フラッシュ・メモリ書き込み時のクロック関連の設定を、デバイス・ファイルの設定に従うか、またはユーザーの設定に従うかを選択します。

<b>-init</b>	デバイス・ファイルに従い処理をします。 デバイス・ファイルに情報が無い場合は、内蔵フラッシュ・メモリ書き込み時にエラーになります。その場合は -user オプションを使用してください。
<b>-user <i>clock</i></b>	クロック発生機能の IOR を操作します。 <i>clock</i> に IOR の操作によって決まる CPU クロック (fCPU) の周波数 (MHz) を指定します。 この場合、フック・プロシージャで IOR の設定を行ってください。

## 使用例

(IDCON) 1 % flop -user 13.5

---

## go

---

---

go - 継続して実行

---

### 入力形式

---

go ?options?

---

### 別名

---

g

---

### 機能説明

---

プログラムの実行を継続します。-waitbreak を指定すると、プログラムが停止するのを待ちます。

options には、以下のものがあります。

-ignorebreak	ブレーク・ポイントを無視します。
-waitbreak	プログラムが停止するまで待ちます。

---

### 使用例

---

(IDCON) 1 % go -w

---

## help

---

---

help - ヘルプの表示

---

### 入力形式

---

help

---

### 機能説明

---

Dcl のヘルプを表示します。

---

### 使用例

---

(IDCON) 1 % help

For more information on a specific command, type HELP command-name

ASSOC Displays or modifies file extension associations.

AT Schedules commands and programs to run on a computer.

ATTRIB Displays or changes file attributes.

BREAK Sets or clears extended CTRL+C checking.

CACLS Displays or modifies access control lists (ACLs) of files.

CALL Calls one batch program from another.

CD Displays the name of or changes the current directory.

CHCP Displays or sets the active code page number.

CHDIR Displays the name of or changes the current directory.

CHKDSK Checks a disk and displays a status report.

CHKNTFS Displays or modifies the checking of disk at boot time.

CLS Clears the screen.

⋮  
⋮  
⋮

## hook

---

---

hook - フックの設定

### 入力形式

---

`hook scriptfile`

### 機能説明

---

*scriptfile* でフック用プロシージャを設定します。

フック設定はプロジェクト・ファイルのロード時と ID850QB リセット時に初期化されます。

### 使用例

---

(IDCON) 1 % hook d:/foo.tcl

## ie

---

---

ie - IE レジスタの表示／設定

### 入力形式

---

ie reg address ?value?

ie dcu address ?value?

### 機能説明

---

ie コマンドはインサーキット・エミュレータ依存です。

サブ・コマンドに reg を指定した際は、IE レジスタの参照と設定を行います。

サブ・コマンドに dcu を指定した際は、DCU レジスタの参照と設定を行います。

**注意** DCU レジスタの参照を行うとレジスタの値は0にリセットされます。

### 使用例

---

(IDCON) 1 % ie reg 0x100 1

(IDCON) 2 % ie dcu 0x100 1

## inspect

inspect - シンボル・インスペクト

### 入力形式

`inspect ?options? progname pattern`

### 機能説明

`progname` で指定したロード・モジュールのシンボルを `pattern` の正規表現で検索して表示します。

以下の正規表現が使用できます。

?	1文字にマッチ
*	0文字以上の文字にマッチ
[chars]	chars の文字にマッチ ([a-z0-9] 等の範囲指定も可能)
\x	文字 x にマッチ (? * [] \ の指定も可能)

`options` には以下のものがあります。

<b>-nocase</b>	アルファベットの大きい文字と小さい文字を区別しません。
<b>-address</b>	シンボルのアドレスとペアで表示します。

### 使用例

(IDCON) 1 % inspect test1.out {[a-z]\*}

---

## jump

---

---

jump - ウィンドウへのジャンプ

---

### 入力形式

---

**jump** -source -line *filename* ?*line*?

**jump** ?*options*? *address*

---

### 別名

---

j

---

### 機能説明

---

*options* で指定したウィンドウを表示します。

<b>-source</b>	<i>address</i> で指定したアドレスからソース・テキスト・ウィンドウを表示します。
<b>-assemble</b>	<i>address</i> で指定したアドレスから逆アセンブル・ウィンドウを表示します。
<b>-memory</b>	<i>address</i> で指定したアドレスからメモリ・ウィンドウを表示します。
<b>-line</b>	<i>line</i> で指定した行に移動します。
<b>-focus</b>	表示したウィンドウにフォーカスを移動します。

---

### 使用例

---

(IDCON) 1 % jump -s main

(IDCON) 2 % jump -s -l mainfile.c 10

(IDCON) 3 % jump -m array

## map

map - メモリ・マッピングの設定／削除

### 入力形式

```
map options address1 address2 ?accsize? ?cs?
```

### 機能説明

メモリ・マッピングの設定／削除／表示を行います。

**備考** `accsize` に 8, 16, 32 のいずれかのアクセス・サイズ（単位：バイト、デフォルトは 8）を指定します。

**注意** ID850QB でエミュレーション・メモリ（代替 ROM/RAM）をマッピングする場合には、`cs` に `cs0`, `cs1`, `cs2`, `cs3`, `cs4`, `cs5`, `cs6`, `cs7` のいずれかのチップ・セレクトを指定します。

ただし、V850ES シリーズの品種の場合には、チップ・セレクトの割り当てが固定、またはチップ・セレクト機能がないため省略可能です。また、`cs` を指定した場合には `accsize` は省略できません。

`options` には、以下のものがあります。

<b>-erom</b> [IECUBE]	代替 ROM をマッピングします（メモリ・ボード搭載時）。
<b>-eram</b> [IECUBE]	代替 RAM をマッピングします（メモリ・ボード搭載時）。
<b>-target</b>	ターゲット領域をマッピングします。
<b>-targetrom</b> [IECUBE]	ターゲット ROM 領域をマッピングします。
<b>-protect</b>	I/O プロテクト領域をマッピングします。
<b>-rrm</b>	RRM 領域の開始アドレスを設定します。 ユーザ・プログラム実行中に行った場合は CPU を一瞬止めて行います。 RRM 領域の 8 分割ができます。 以下の様に開始アドレスとサイズのペアをリスト形式で指定します。 {address size} {address size} {address size}...} size は 256, 512, 768, 1024, 1280, 1536, 1792, 2048 バイトのいずれかで、サイズの合計は 2048 バイトまでです。
<b>-clear</b>	すべてのマッピング設定を削除します。
<b>-information</b>	マッピング設定を参照します。

## 使用例

---

```
(IDCON) 1 % map -i  
1: 0 0x7fff 8 {IROM}  
2: 0x8000 0x87ff 8 {Target RRM}  
3: 0x8800 0x9fff 8 {Target}  
4: 0xa000 0xf7ff 8 {NonMap}  
5: 0xf800 0xfaff - {NonMap}  
6: 0xfb00 0xfedf 8 {Saddr}  
7: 0xfe0 0xfeff 8 {Register}  
8: 0xff00 0xffff 8 {IOR}  
(IDCON) 2 % map -erom 0x100000 0x10ffff  
(IDCON) 3 % map -c
```

## mdi

---

---

mdi - 拡張ウィンドウの設定

### 入力形式

---

`mdi geometry ?x y? width height`

`mdi title string`

### 機能説明

---

拡張ウィンドウのサイズとタイトル名を設定します。

拡張ウィンドウからのみ使用できます。

### 使用例

---

(IDCON) 1 % mdi geometry 0 0 100 100

(IDCON) 2 % mdi title foo

## memory

memory - メモリの表示／設定

### 入力形式

`memory ?options? address ?value?`

`memory ?options? -fill address1 address2 value`

`memory ?options? -copy address1 address2 address3`

### 別名

m

### 機能説明

`address` で指定したアドレスのメモリに `options` にしたがって `value` を設定します。

`value` を省略すると `address` で指定した アドレスのメモリの値を表示します。

`-fill` を指定すると `address1` から `address2` の間を `value` で充填します。

`-copy` を指定すると `address1` から `address2` の間を `address3` に複写します。

`options` には、以下のものがあります。

<code>-byte</code>	1 バイト単位で表示／設定します (デフォルト)。
<code>-halfword</code>	1 ハーフ・ワード単位で表示／設定します。
<code>-word</code>	1 ワード単位で表示／設定します。
<code>-fill</code>	データを充填します。
<code>-copy</code>	データを複写します。
<code>-noverify</code>	書き込み時にベリファイをしません。

**参考** ユーザ・プログラム実行中に以下の操作を行った場合には、CPU を一瞬止めて行っています。

- RRM 領域以外のメモリ参照
- メモリ設定

### 使用例

(IDCON) 1 % memory 100

0x10

(IDCON) 2 % memory 100 2

(IDCON) 3 % memory 100

0x02

(IDCON) 4 % memory -fill 0 1ff 0

## module

---

---

module - ファイル一覧, 関数一覧の表示

### 入力形式

---

`module progname ?filename?`

### 機能説明

---

*progname* で指定したロード・モジュールのファイル一覧／関数一覧を表示します。

*filename* を指定しない場合はファイルの一覧を表示します。

*filename* を指定した場合は指定したファイルの関数の一覧を表示します。

### 使用例

---

```
(IDCON) 1 % module rammon.lmf
1: rammon.c
(IDCON) 2 % module rammon.lmf rammon.c
1: rammon.c sub1
2: rammon.c main
```

---

## next

---

---

next - プロシージャ・ステップ

---

### 入力形式

---

next ?options?

---

### 別名

---

n

---

### 機能説明

---

プロシージャ・ステップ実行を行います。関数呼び出しを行っている場合は、関数実行後停止します。

options には、以下のものがあります。

<b>-source</b>	ソースの行単位で実行します (デフォルト)。
<b>-instruction</b>	命令単位で実行します。

---

### 使用例

---

(IDCON) 1 % next -i

(IDCON) 2 % next -s

## refresh

---

---

refresh - ウィンドウの再描画

### 入力形式

---

refresh

### 機能説明

---

ウィンドウを再描画し、最新のデータに更新します。

### 使用例

---

(IDCON) 1 % batch foo.tcl  
(IDCON) 2 % refresh

---

## register

---

---

register - レジスタ値, I/O レジスタ値の表示／設定

---

### 入力形式

---

`register ?options? regname ?value?`

---

### 機能説明

---

*regname* で指定したレジスタに *value* を設定します。

*value* を省略した場合は *regname* で指定したレジスタの値を表示します。

*options* には、以下のものがあります。

<code>-force</code>	強制読み込み, または書き込みを行います。
---------------------	-----------------------

**参考** ユーザ・プログラム実行中に以下の操作を行った場合には、CPU を一瞬止めて行います。

- ・レジスタの参照
- ・レジスタの設定

---

### 使用例

---

```
(IDCON) 1 % register pc
0x100
(IDCON) 2 % register pc 200
(IDCON) 3 % register pc
0x200
```

---

## reset

---

---

reset - リセット

---

### 入力形式

---

reset *?options?*

---

### 機能説明

---

ID850QB, CPU, シンボル, またはイベントをリセットします。

オプションを省略した場合は, CPU リセットをします。

*options* には, 以下のものがあります。

<b>-cpu</b>	CPU をリセットします (デフォルト)。
<b>-debugger</b>	ID850QB をリセットします。
<b>-symbol</b>	シンボルをリセットします。
<b>-event</b>	すべてのイベントとソフトウェア・ブレークをリセットします。

---

### 使用例

---

(IDCON) 1 % reset

---

## run

---

---

run - CPU リセット & 実行

---

### 入力形式

---

run *?options?*

---

### 別名

---

r

---

### 機能説明

---

プログラムをリセット後実行します。

-waitbreak を指定していなければ、プログラムの停止を待ちません。

*options* には、以下のものがあります。

<b>-waitbreak</b>	プログラムが停止するまで待ちます。
-------------------	-------------------

---

### 使用例

---

(IDCON) 1 % run

(IDCON) 2 % run -w

---

## step

---

---

step - ステップ実行

---

### 入力形式

---

**step** *?options?*

---

### 別名

---

**s** (ソース・ステップ (step -source))

**i** (命令ステップ (step -instruction))

---

### 機能説明

---

ステップ実行を行います。

関数呼び出しを行っている場合は、関数の先頭で停止します。

*options* には、以下のものがあります。

<b>-source</b>	ソースの行単位で実行します (デフォルト)。
<b>-instruction</b>	命令単位で実行します。

---

### 使用例

---

(IDCON) 1 % step -i

(IDCON) 2 % step -s

## stop

---

---

stop - 実行停止

### 入力形式

---

stop

### 機能説明

---

プログラムを強制的に停止します。

### 使用例

---

(IDCON) 1 % run  
(IDCON) 2 % stop

## tkcon

tkcon - コンソール制御

### 入力形式

tkcon cmd ?arg?

### 機能説明

コンソール・ウインドウを制御するコマンドです。

このコマンドはコンソール/Tcl コマンドの1つです（「表 7-2 コンソール/Tcl コマンド一覧」参照）。

<b>tkcon buffer ?size?</b>	コンソールの最大バッファ・サイズ（行数）を設定、参照します。バッファ・サイズを越えると最古の行から削除されます。
<b>tkcon close</b> <b>tkcon destroy</b>	コンソール・ウインドウを閉じます。
<b>tkcon font ?fontname?</b>	コンソール・ウインドウのフォントを設定、参照します。
<b>tkcon gets</b>	Stdin のような標準入力を行います。ダイアログがオープンします。
<b>tkcon history ?-newline?</b>	コマンドの履歴を表示します。
<b>tkcon save ?filename? ?type</b>	コンソール・ウインドウのバッファの内容をファイルに保存します。ファイル名、またはタイプを省略するとダイアログがオープンします。タイプは、all, history, stdin, stdout, stderr から選択します。
<b>tkcon version</b>	コンソールのバージョンを表示します。

### 使用例

(IDCON) 1 % tkcon save c:/temp/logfile.txt all

(IDCON) 2 % tkcon font {{MS ゴシック} 12}

---

## upload

---

---

upload - アップロード

---

### 入力形式

---

**upload** *?options? filename address1 address2*

**upload** -coverage *filename*

---

### 機能説明

---

指定したアドレス範囲のメモリ・データ、またはカバレッジ・データをファイルに保存します。

*options* には、以下のものがあります。

<b>-binary</b>	バイナリ形式で保存します。
<b>-coverage [IECUBE]</b>	カバレッジ・データを保存します。 カバレッジ・データの場合は、全範囲をファイルに保存します（開始アドレス／終了アドレスの指定は必要なし）。
<b>-intel</b>	インテル・ヘキサ形式で保存します（デフォルト）。
<b>-motorola</b>	モトローラ・ヘキサ形式で保存します。
<b>-tektronix</b>	テクトロニクス（テック）・ヘキサ形式で保存します。
<b>-force</b>	ファイルを上書きします。

---

### 使用例

---

(IDCON) 1 % upload -b foo.hex 0 0xffff

---

## version

---

---

version - バージョンの表示

---

### 入力形式

---

version

---

### 機能説明

---

ID850QB のバージョンを表示します。

---

### 使用例

---

```
(IDCON) 1 % version
GUI           : Vx.xx [XX XXXX 200X]
Devicefile   : V850 Device File [uPD703201] Vx.xx
Debugger     : V850 Debugger Vx.xx [XX XXXX 200X]
Executer     : V850 Executer Vx.xx [XXX XXXX 200X]
Monitor      : V850 Peripheral Vx.xx [XX XXXX 200X]
Assembler    : V850 Asm/Disasm Vx.xx [XX XXXX 200X]
Tcl/Tk       : 8.4.12
```

## watch

watch - 変数値の表示／設定

### 入力形式

`watch ?options? variable ?value?`

### 別名

w

### 機能説明

変数値の表示と設定をします。

*options* には、以下のものがあります。

<b>-binary</b>	値を 2 進数で表示します。
<b>-octal</b>	値を 8 進数で表示します。
<b>-decimal</b>	値を 10 進数で表示します。
<b>-hexdecimal</b>	値を 16 進数で表示します。
<b>-string</b>	値を文字列で表示します。
<b>-sizeof</b>	値の代わりに変数のサイズを 10 進数で表示します。
<b>-encoding <i>name</i></b>	文字列表示時のエンコーディングを指定します。 デフォルトでは、システムのエンコーディングを使います。 <i>name</i> (エンコーディング名) は Tcl の仕様に準拠します (shiftjis, euc-jp 等)。

### 使用例

```
(IDCON) 1 % watch var
0x10
(IDCON) 2 % watch -d var
16
(IDCON) 3 % watch array[0] 0xa
```

## where

---

---

where - スタックのトレース

### 入力形式

---

where

### 機能説明

---

スタックをバック・トレースします。

### 使用例

---

```
(IDCON) 1 % where  
1: test2.c#sub2(int i)#13  
2: test.c#num(int i)#71  
3: test.c#main()#82
```

## wish

---

---

wish - Tclet の起動

### 入力形式

---

**wish** *scriptname*

### 機能説明

---

Tk を使ったスクリプト (Tclet) を起動します。

Tclet で拡張ウィンドウを作成できます。

### 使用例

---

(IDCON) 1 % wish test.tcl

---

## xcoverage

[IECUBE]

---

xcoverage - カバレッジの操作

---

### 入力形式

**xcoverage** *option*

---

### 機能説明

カバレッジの操作をします。

*option* には、以下のものがあります。

<b>-clear</b>	カバレッジ・メモリをクリアします。
---------------	-------------------

---

### 使用例

(IDCON) 1 % xcoverage -clear

---

## xtime

[IECUBE]

---

xtime - タイマの操作

---

### 入力形式

xtime *option*

---

### 機能説明

タイマの操作をします。

*option* には、以下のものがあります。

<b>-start</b>	プログラム実行中にタイマを開始します。
<b>-stop</b>	プログラム実行中にタイマを停止します。
<b>-gobreak</b>	Go-Break 間の時間を nsec で表示します。

---

### 使用例

(IDCON) 1 % xtime -start

(IDCON) 2 % xtime -stop

## xtrace

[IECUBE]

xtrace - トレーサの操作

### 入力形式

**xtrace** -dump *?-append? frameno ?filename?***xtrace** -start**xtrace** -stop**xtrace** -clear**xtrace** -addup *?bool?***xtrace** -mode *?mode?***xtrace** -complement *?bool?*

### 機能説明

トレーサの操作をします。

*option* には、以下のものがあります。

<b>-start</b>	プログラム実行中にトレーサを開始します。
<b>-stop</b>	プログラム実行中にトレーサを停止します。
<b>-clear</b>	トレース・メモリをクリアします。
<b>-dump</b>	トレース・データをダンプします (デフォルト)。 ダンプ結果はコンソール・ウィンドウにリダイレクトされます。 ファイル名を指定した際は、ダンプ結果をファイルに書き込みます。
<b>-append</b>	ダンプ結果をファイルに追加します。
<b>-addup <i>?bool?</i></b>	タイム・タグを積算するか選択します。 <i>bool</i> を省略した場合、現在のモードを表示します。
<b>-mode <i>?mode?</i></b>	トレース制御モードを選択します (all, cond, nonstop, fullstop, fullbreak, delaystop, delaybreak, machine, event のいずれか)。 <i>mode</i> を省略した場合、現在のモードを表示します。
<b>-complement <i>?bool?</i></b>	トレース補完をするかどうかを選択します。 <i>bool</i> を省略した場合、現在のモードを表示します。

### 使用例

```
(IDCON) 1 % xtrace -start
(IDCON) 2 % xtrace -stop
(IDCON) 3 % xtrace -dump 3
_ 01685 2 000000BC M1 br _sub2+0x2
_ 01686 4 0000009A BRM1 st.w r6, 0x8[sp]
_ 01687 3 0000009E BRM1 st.w r0, 0x0[sp]
(IDCON) 4 % xtrace -clear
(IDCON) 5 % xtrace -addup true
```

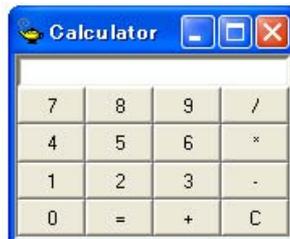
## 7.13 サンプル（電卓スクリプト）

次に、電卓スクリプトを記述した拡張ウィンドウのスクリプトとその実行画面を示します。

### 拡張ウィンドウのスクリプト

```
# Calculator.tcl
mdi geometry 100 100
set top .dcl
entry $top.e -relief sunken -textvariable v
frame $top.f -height 120 -width 120; pack $top.e -fill x; pack $top.f -fill both -expand 1
set i 0; set v {}; set r 0.25
foreach n {7 8 9 / 4 5 6 * 1 2 3 - 0 = + C} {
    if {$n == "=" || $n == "C"} {
        button $top.f.b$n -text $n
    } else {
        button $top.f.b$n -text $n -command "$top.e insert end $n"
    }
    place $top.f.b$n -relx [expr ($i%4)*$r] -rely [expr ($i/4)*$r] -relw $r -relh $r
    incr i
}
bind $top.f.bC <1> {$top.e delete 0 end}
bind $top.f.b= <1> {catch {expr $v} v}
```

図 7 - 1 実行画面



## 付録 A 拡張ウィンドウ

- 拡張ウィンドウ概要
- サンプル・ウィンドウ一覧
- サンプル・ウィンドウの起動
- 各サンプル・ウィンドウの説明

## A.1 拡張ウインドウ概要

ID850QB では、既存のウインドウに加えユーザ自身でカスタム・ウインドウの作成が可能です。

ID850QB には、Tcl (Tool Command Language) のインタプリタとデバッガ制御用のコマンドが組み込まれており、カスタム・ウインドウはこの Tcl により作成します。

ID850QB には、いくつかの拡張ウインドウのサンプルが同梱されています。

## A.2 サンプル・ウインドウ一覧

表 A-1 拡張ウインドウのサンプル・ウインドウ一覧

ウインドウ名	機能概要
List ウインドウ	ソース・ファイルと関数名の一覧を表示します。
Grep ウインドウ	文字列を検索します。
RRM ウインドウ	リアルタイム RAM モニタ用のメモリ・ウインドウです。
Hook ウインドウ	フック・プロシージャの設定を行います。
Memory Mapped I/O ウインドウ	指定したアドレスの書き込み、または読み込みを行います。
Sym Inspect ウインドウ	シンボルを正規表現で検索します。
Run Break Time ウインドウ	ユーザ・プログラムの実行 (Run) からブレーク (Break) までの時間を表示します。

## A.3 サンプル・ウインドウの起動

[ブラウザ]メニュー→[その他]→[List/Grep/RRM/Hook/Memory/SymInspect/RunBreakTime] を選択

**備考** これらサンプル・ウインドウの .tcl ファイルは以下のフォルダにインストールされています。

NEC Electronics Tools¥ID850QB¥Vx.xx¥bin¥dtcl¥tools

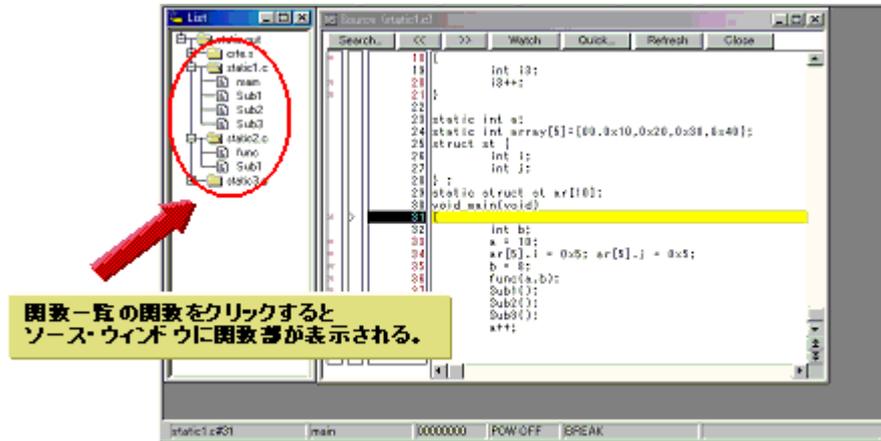
## A.4 各サンプル・ウインドウの説明

ID850QB では、以下のサンプル・ウインドウを用意しています。

## List ウィンドウ

ソース・ファイルと関数の一覧をツリー表示します。また、一覧表示をクリックすると対応したソースを表示します。

図 A - 1 List ウィンドウ

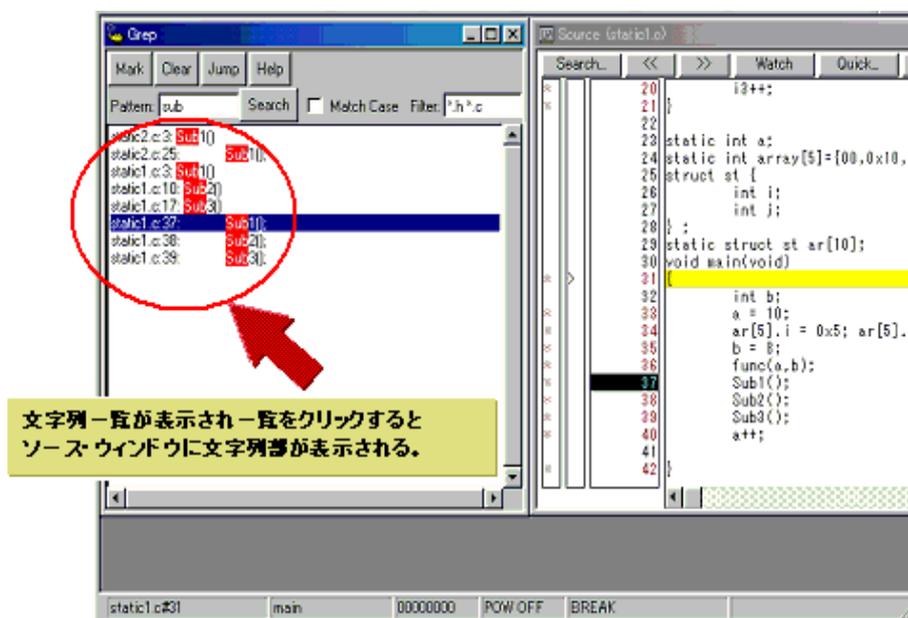


## Grep ウインドウ

ソース・パスのファイルから文字列検索を行います。

また、検索結果からクリックすると対応したソース・ファイルを表示します。

図 A - 2 Grep ウインドウ



オブジェクト	機能
Pattern	検索したい文字列を入力します。
<Mark> ボタン	検索した文字列をマークします。
<Clear> ボタン	マークをクリアします。
<Jump> ボタン	検索箇所にカーソルを置き、このボタンをクリックすることにより、対応したファイルを開きます。
Match Case	大文字／小文字の区別の有無を指定します。
Filter	検索するファイル・タイプを指定します。

## RRM ウィンドウ

RAM モニタ専用のウィンドウです。

RAM 領域にて、実行中に値が変化したアドレスの領域に色付けをし表示します。表示範囲は 1K バイトです。また、RAM を読み出す際にはユーザ・プログラムの実行が一瞬ブレークします。

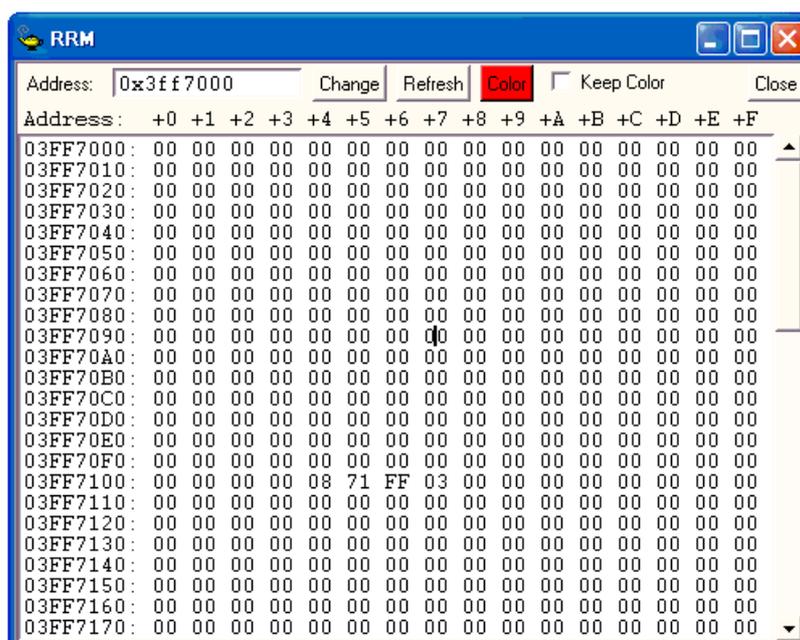
なお、このウィンドウ上では、ユーザ・プログラム実行中に RAM 領域の開始アドレス変更が可能です。

**注意** すべてのデータは、同一の時間には読み出されません（1K バイトをワード単位に分割して読み出しているため）。

**参考 1** 拡張オプション設定ダイアログで RAM モニタ機能 OFF の場合でも動作します。

**参考 2** サンプリング間隔は CPU の動作周波数に依存しますが、20MHz 動作時で 0.3 - 0.7 秒間隔程度になります。**[MINICUBE]**

図 A - 3 RRM ウィンドウ



オブジェクト	機能
Address	表示開始アドレスを入力します（自動的に 1K バイトアラインします）。
<Change> ボタン	開始アドレスの表示切り替えを行います。
<Refresh> ボタン	メモリの読み出しを行います。
<Color> ボタン	色のカスタマイズをします（デフォルトは赤）。
Keep Color	色表示の継続の有無を指定します。 チェックあり：一度でも値の変化があれば、ブレークするまで色表示を継続 チェックなし：値が変化しなければ色表示を消去（デフォルト）
<Close> ボタン	このウィンドウをクローズします。

## Hook ウインドウ

フック・プロシージャを使用し、デバッガにフックを設定するためのウインドウです。

フック・プロシージャを使用することで、ロード・モジュールのダウンロード前や CPU リセット後に、レジスタの値を変更することができます。

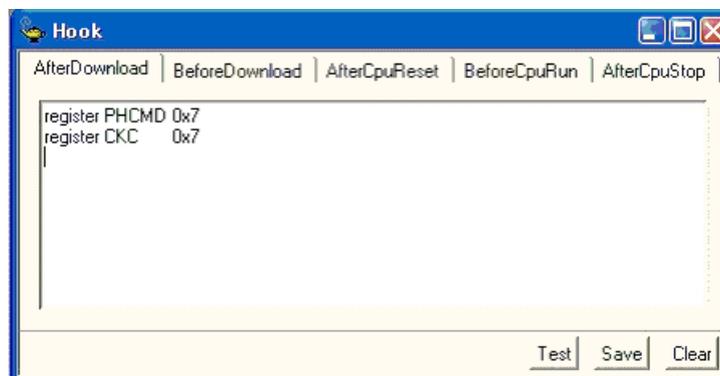
このウインドウでは、次の 5 つのタブでフック設定が可能です。

- [AfterDownload] タブ (ダウンロード後のフック)
- [BeforeDownload] タブ (ダウンロード前のフック)
- [AfterCpuReset] タブ (ブレイク中の CPU リセット後のフック)
- [BeforeCpuRun] タブ (実行開始前のフック)
- [AfterCpuStop] タブ (ブレイク後のフック)

**参考** 例えば、ロード・モジュールのダウンロード前に、[BeforeDownload] タブで IOR を設定することにより、ダウンロードを高速に行うことができます。また、外部メモリへのアクセスもこのタブを設定することで容易になります。

また、このウインドウの設定を、プロジェクトのあるフォルダに "プロジェクト・ファイル名 .tcl" で保存することにより、プロジェクトを開くときに同時に実行させることができます。

図 A - 4 Hook ウインドウ



オブジェクト	機能
[AfterDownload] タブ	ダウンロード後のフック。 ロード・モジュール・ファイルをダウンロードした後に、タブに記述したレジスタの値を指示した値に自動的に書き換えます。
[BeforeDownload] タブ	ダウンロード前のフック。 ロード・モジュール・ファイルをダウンロードする前に、タブに記述したレジスタの値を指示した値に自動的に書き換えます。
[AfterCpuReset] タブ	ブレイク中の CPU リセット後のフック。 CPU リセット後に、タブに記述したレジスタの値を指示した値に自動的に書き換えます。
[BeforeCpuRun] タブ	実行開始前のフック。 実行開始前にタブに記述したレジスタの値を指示した値に自動的に書き換えます。
[AfterCpuStop] タブ	ブレイク後のフック。 ブレイク後にタブに記述したレジスタの値を指示した値に自動的に書き換えます。
<Test> ボタン	すべてのタブに記述したコマンドをテスト実行します。
<Save> ボタン	すべてのタブの内容をファイルにセーブします。 プロジェクト・ファイルから ID850QB を起動した場合、"プロジェクト・ファイル名 .tcl" で保存します。
<Clear> ボタン	すべてのタブの記述をクリアします。

**注意** レジスタ名はプログラム・レジスタと周辺 I/O レジスタを指定します。

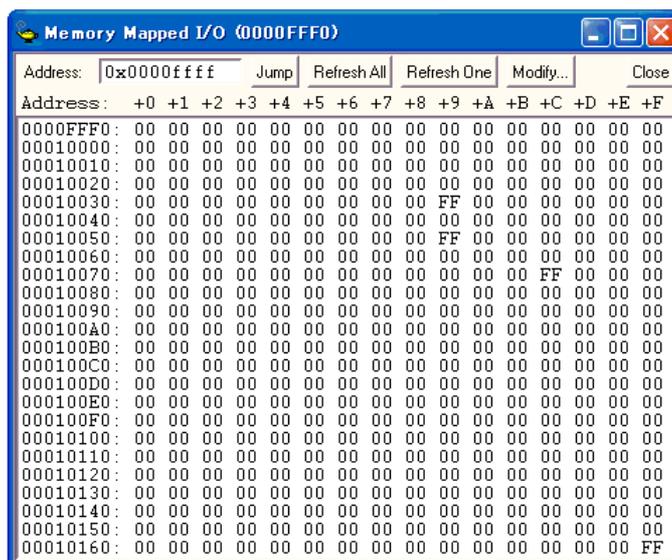
## Memory Mapped I/O ウインドウ

指定のアドレスに対してデータの読み込み、および書き込みを明示的に行います。

通常、メモリ・ウインドウ上で書き込みを行った場合、ID850QB が内部的にデータ内容を読み込みベリファイを行っています。また、メモリ・ウインドウ上ではスクロール操作をするだけでもメモリは読み込まれます。一方、このウインドウでは上記の動作は行われません。このため、任意のアドレスのみの読み込み、および書き込みをしたい場合このウインドウが有効です。

なお、ユーザ・プログラム実行中の書き込みは、一瞬ブレイクさせたあとに行っています。

図 A - 5 Memory Mapped I/O ウインドウ



オブジェクト	機能
Address:	表示したいアドレスを入力します。 Enter キーの押下、または <Jump> ボタンのクリックにより表示対象アドレスが変化します。この時、データ内容は読み出されないため、アドレス表示部には数値が表示されませんがデータ部には "XX" が表示されます。
<Jump> ボタン	Address 欄に入力したアドレスの表示を変更します。
<Refresh All> ボタン	表示しているすべての領域のデータを一回だけ読み込みます。 ノンマップ領域を読み込もうとした場合や、読み込み時にエラーが発生した場合には、データ部に "ZZ" が表示されます。
<Refresh One> ボタン	カーソルのあるアドレスのメモリ・データを一回だけ読み込みます。 読み込むデータ・サイズは表示形式に依存します。 ノンマップ領域を読み込もうとした場合や、読み込み時にエラーが発生した場合には、データ部に "ZZ" が表示されます。
<Modify...> ボタン	Memory Mapped I/O 入力ダイアログをオープンします。 カーソルの位置が Memory Mapped I/O 入力ダイアログに表示される入力アドレスとなります。Memory Mapped I/O 入力ダイアログがオープンしている時に本ウインドウのカーソル位置を変更し、このボタンをクリックすると、Memory Mapped I/O 入力ダイアログの Address も変化します。

オブジェクト	機能
<Close> ボタン	このウインドウをクローズします。
コンテキスト・メニュー	表示形式を Byte/HalfWord/Word のいずれかに変更します。

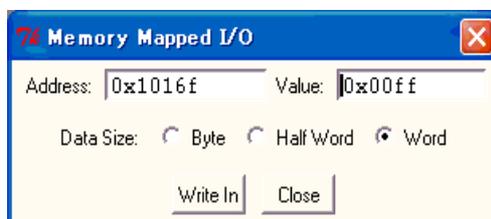
## Memory Mapped I/O 入力ダイアログ

Memory Mapped I/O ウインドウの <Modify...> ボタンをクリックすることによりオープンし、任意のアドレスへの書き込みを行います。

**注意 1** データを書き込む領域をメモリ・ウインドウやウォッチ・ウインドウで表示している場合、<Write in> ボタンのクリック後に、これらのウインドウのデータが読み込まれます。

**注意 2** "Data Size:" が、[コンフィギュレーション・ダイアログ](#)で指定されている "Access Size:" より小さい場合、ID850QB は、一度 "Access Size:" でデータを読み込んだ後、読み込んだデータの該当部分を変更し、変更したデータを "Access Size:" で書き込みます。

図 A - 6 Memory Mapped I/O 入力ダイアログ

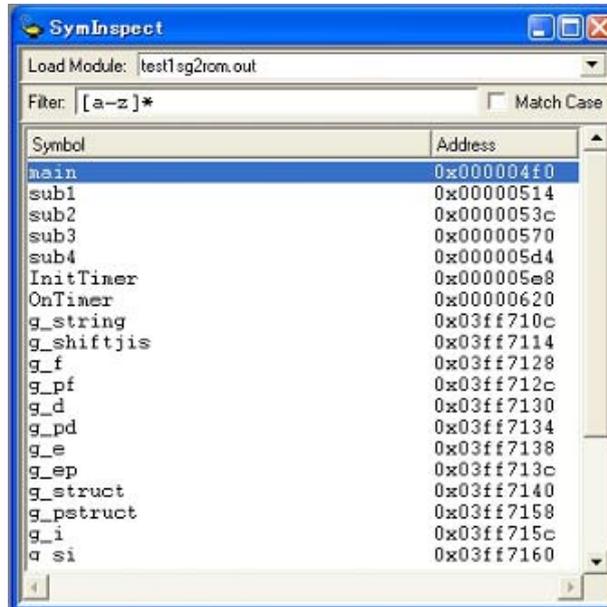


オブジェクト	機能
Address:	書き込みを行うアドレスを入力します。 デフォルトでは、 <a href="#">Memory Mapped I/O ウインドウ</a> でのカーソル位置のデータに対応するアドレスが表示されています。
Value:	書き込む値を入力します。
Data Size:	書き込みを行うデータ・サイズを選択します。 デフォルトでは、 <a href="#">Memory Mapped I/O ウインドウ</a> で指定されている表示サイズが選択されています。
<Write in> ボタン	指定したアドレス、データ・サイズでデータを書き込みます。
<Close> ボタン	このダイアログをクローズします。

## Sym Inspect ウインドウ

ロードしたロード・モジュール・ファイルのシンボルとアドレスの一覧を表示するウインドウです。  
シンボルを正規表現で検索します。

図 A - 7 Sym Inspect ウインドウ



オブジェクト	機能
Load Module:	ロード・モジュール・ファイルを選択します。
Filter:	シンボルを検索するための正規表現を指定します。
Match Case	Filter: で大文字, 小文字を区別する／しないを指定します。 チェックにより大文字／小文字を区別します。
Symbol	シンボルを表示します。クリックによりシンボル順にソートされます。
Address	アドレスを表示します。クリックによりアドレス順にソートされます。

コンテキスト・メニュー	機能
Copy	選択行のアドレスをクリップ・ボードにコピーします。
Jump to Source	選択行のアドレス (ソース・テキスト・ウインドウ) にジャンプします。
Jump to Assemble	選択行のアドレス (逆アセンブル・ウインドウ) にジャンプします。
Jump to Memory	選択行のアドレス (メモリ・ウインドウ) にジャンプします。

## Run Break Time ウインドウ

ユーザ・プログラムの実行開始 (Run) からブレーク (Break) までの時間を表示するウインドウです。長時間測定をする場合に有効です。Windows の時間機能を使用して実現しており、最小単位は秒です。

図 A - 8 RunBreakTime ウインドウ



オブジェクト	機能
<Clear> ボタン	時間表示をクリアします。

## 付録 B 入力規約

- 使用可能文字
- シンボル規定
- 数値規定
- 式と演算子に関する規定
- ファイル名

## B.1 使用可能文字

表 B - 1 文字セット一覧

分類	文字
英文字	大文字 : ABCDEFGHIJKLMNOPQRSTUVWXYZ 小文字 : abcdefghijklmnopqrstuvwxyz
数字	0123456789
英字相当文字	@ _

表 B - 2 特殊文字一覧

文字	名称	主な用途
(	左かっこ	演算順序の変更
)	右かっこ	演算順序の変更
+	プラス	加算演算子, または正符号
-	マイナス	減算演算子, または負符号
*	アスタリスク	乗算演算子, または間接参照演算子
/	スラッシュ	除算演算子
%	パーセント	剰余演算子
~	チルダ	補数演算子
	縦線	ビット和演算子
^	ハット	ビット差演算子
&	アンド	ビット積演算子, またはアドレス演算子
.	ピリオド	直接メンバ演算子, またはビット位置指定子
,	カンマ	オペランド間の区切り記号
[	左ブラケット	配列添字演算子, またはベース・レジスタ指定記号
]	右ブラケット	

## B.2 シンボル規定

- (1) シンボルは以下のいずれかの文字で構成されます。

A-Z, a-z, @, \_ (アンダバー), ?, 0-9

- (2) シンボルの先頭は、0-9の数字以外の文字でなければなりません。
- (3) 英大文字 (A-Z), 英小文字 (a-z) は区別されます。
- (4) シンボル名は最大 2048 文字です (2048 文字以上のシンボルを定義した場合、先頭から 2048 文字までが有効)。
- (5) シンボルは、ロード・モジュール・ファイルをロードすることにより定義されます。
- (6) シンボルには有効範囲ごとに次の種類があります。

グローバル・シンボル (アセンブリ言語, C 言語)

スタティック・シンボル (C 言語)  
 - ファイル内スタティック・シンボル  
 - 関数内スタティック・シンボル

ローカル・シンボル (C 言語)  
 - ファイル内ローカル・シンボル  
 - 関数内ローカル・シンボル  
 - ブロック内ローカル・シンボル

- (7) 使用する言語ごとに、次のシンボルが存在します。

**アセンブリ言語,**  
 ラベル名, ビット・シンボル名

**C 言語**  
 変数名 (ポインタ変数名, 列挙型変数名, 配列名, 構造体名, 共用体名を含む) 関数名, ラベル名  
 配列要素, 構造体要素, 共用体要素, ビット・フィールド (ただしシンボルが配列, 構造体, 共用体の場合)

- (8) シンボルはアドレスと数値の代わりに記述することができます。
- (9) シンボルの有効範囲は、アセンブル、またはコンパイルされた際のソース・デバッグ情報に基づいて決定されます。
- (10) グローバル・シンボルは、シンボル名だけを記述します。
- (11) ローカル・シンボルは、ファイル名と対にして表現します。

## B.3 数値規定

数値として次の種類が使用できます。それぞれ入力形式は次の通りです。

なお、サフィックス（太字），および 16 進数の英字は，大文字でも小文字でも構いません。また，先頭文字が A - F の場合，先頭に 0 をつけなければなりません。

ID850QB 上での入力欄は，デフォルトの進数に応じて，10 進数と 16 進数が切り替わります。

表 B - 3 数値の入力形式

数値	入力形式
2 進数	n <b>Y</b> n...n <b>Y</b> (n=0,1)
8 進数	n <b>O</b> n...n <b>O</b> (n=0,1,2,3,4,5,6,7) n <b>Q</b> n...n <b>Q</b> (n=0,1,2,3,4,5,6,7)
10 進数	n n...n n <b>T</b> n...n <b>T</b> (n=0,1,2,3,4,5,6,7,8,9)
16 進数	n n...n n <b>H</b> n...n <b>H</b> <b>0xn</b> <b>0xn</b> ...n (n=0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F)

## B.4 式と演算子に関する規定

### (1) 式

式とは、定数、レジスタ名、周辺 I/O レジスタ名、シンボル、およびこれらを演算子で結合したものをいいます。また、シンボルとして、周辺 I/O レジスタ名、ラベル名、関数名、変数名が記述された場合は、アドレスをシンボルの値として演算します。

なお、式を構成する演算子以外の要素を、項（定数、ラベル）といい、記述された左側から順に、第 1 項、第 2 項 ... と呼びます。

### (2) 演算子

演算子は C 言語の演算子から次に示すものが利用できます。

表 B - 4 演算子一覧

記号	意味	説明
<b>算術演算子</b>		
+	加算	第 1 項と第 2 項との値の和を返します。
-	減算	第 1 項と第 2 項との値の差を返します。
*	乗算	第 1 項と第 2 項との値の積を返します。
/	除算	第 1 項の値を第 2 項の値で割り、結果の整数部を返します。
MOD %	剰余	第 1 項の値を第 2 項の値で割り、結果の余りを返します。
- 符号	単項演算（負）	項の値の 2 の補数をとった値を返します。
+ 符号	単項演算（正）	項の値を返します。
<b>論理演算子</b>		
NOT ~	否定	項のビットごとの論理否定を取り、その値を返します。
AND &	論理積	第 1 項の値と第 2 項の値とのビットごとの論理積を取り、その値を返します。
OR 	論理和	第 1 項の値と第 2 項の値とのビットごとの論理和を取り、その値を返します。
XOR ^	排他的論理和	第 1 項の値と第 2 項の値とのビットごとの排他的論理和を取り、その値を返します。
<b>シフト演算子</b>		
SHR >>	右シフト	第 1 項の値を第 2 項の値で示す値（ビット数）分だけ右シフトし、その値を返します。上位ビットには、シフトされたビット数だけ 0 が挿入されます。
SHL <<	左シフト	第 1 項の値を第 2 項の値で示す値（ビット数）分だけ左シフトし、その値を返します。下位ビットには、シフトされたビット数だけ 0 が挿入されます。
<b>バイト分離演算子</b>		
HIGH	上位バイト	項の最下位 16 ビット中、上位 8 ビットを返します。
LOW	下位バイト	項の最下位 16 ビット中、下位 8 ビットを返します。

記号	意味	説明
<b>ワード分離演算子</b>		
HIGHW	上位ワード	項の 32 ビット中上位 16 ビットを返します。
LOWW	下位ワード	項の 32 ビット中下位 16 ビットを返します。
<b>その他</b>		
(	左かっこ	() 内の演算を () 外の演算に先立って行います。 '(' と ')' は常に対で使用されます。
)	右かっこ	

**(3) 演算の規則**

演算の順序は演算子の優先順序に従います。

表 B - 5 演算子の優先順位

優先順位	演算子
1 高い	(, )
2	+ 符号, - 符号, NOT, ~, HIGHT, LOW, HIGHW, LOWW
3	*, /, MOD, %, SHR, >>, SHL, <<
4	+, -
5	AND, &
6 低い	OR,  , XOR, ^

- 同じ優先順位の場合は、左から右へ演算を行います。
- () 内の演算は、() 外の演算を先に立って行います。
- 演算における各項は、すべて符号なしの 32 ビット・データとして扱われます。
- 演算結果は、すべて符号なしの 32 ビット・データとして扱われます。
- 演算中にオーバーフローした場合、下位 32 ビットを有効とし、オーバーフローの検出は行いません。

**(4) 項**

項に定数を記述する場合、次の数値を記述することができます。

表 B - 6 進数の範囲

進数	範囲
2 進数	0Y ≤ 数値 ≤ 11111111111111111111111111111111Y (32 桁)
8 進数	0O ≤ 数値 ≤ 3777777777O
10 進数	-2147483648 ≤ 数値 ≤ 4294967295 (負の 10 進数は内部で 2 の補数に変換します。)
16 進数	0H ≤ 数値 ≤ 0FFFFFFFH

## B.5 ファイル名

ファイル名に関しては以下の規約があります。

### (1) ソース・ファイル名とロード・モジュール・ファイル名に関して

ファイル名は、a-z, A-Z, 0-9, ., \_, +, - のいずれかの文字で構成されます。

ファイル名の先頭は、. 以外の文字でなければなりません。

ファイル名の先頭と最後に、. とスペースは使えません。

英大文字 (A - Z), 英小文字 (a - z) は区別されません。

ファイル名は、パスを含めて最大 259 文字です。

### (2) その他のファイル名に関して

Windows のファイル名規約に準拠します。

なお、ファイル名には次の文字は使えません。

`\/:*?"<>|;`

ファイル名の先頭と最後に . とスペースは使えません。

英大文字 (A - Z), 英小文字 (a - z) は区別されません。

ファイル名は、パスを含めて最大 259 文字です。

## 付録 C キー機能一覧

表 C - 1 キー機能一覧

キー	機能
BackSpace	カーソルの前 1 文字を削除し、カーソルは消した文字の位置に移動します。このとき、カーソルの後ろの文字列は前に詰められます。
Delete	<ul style="list-style-type: none"> <li>カーソル位置の 1 文字を削除し、あとの文字列を前に詰めます。</li> <li>イベント・マネージャ、各種イベント・ダイアログにて選択された各種イベント条件を削除します。</li> <li>ウォッチ・ウインドウにて、選択されたデータを削除します。</li> </ul>
Insert	ソース・テキスト・ウインドウ、逆アセンブル・ウインドウにて、挿入モード/上書きモードを切り替えます。ただし、メモリ・ウインドウとレジスタ・ウインドウ、IOR ウインドウにおいては、このキーは無効であり、入力モードは上書きモードだけとなります。
PrintScreen	画面表示全体をビット・マップ・イメージでクリップ・ボードに取り込みます (Windows の機能)。
Esc	<ul style="list-style-type: none"> <li>プルダウン・メニューをクローズします。</li> <li>モーダル・ダイアログをクローズします。</li> <li>入力データの Restore を行います。</li> </ul>
Alt	メニューバーにカーソルを移動します。
End	カーソルを行末へ移動します。
Home	カーソルを行頭へ移動します。
PageUp	画面を 1 画面分スクロール・アップします。カーソルも同時に画面の一番上へ移動します。
PageDown	画面を 1 画面分スクロール・ダウンします。カーソルも同時に画面の一番上へ移動します。
Space	空白 1 文字を挿入します。
Tab	カーソルを次の項目へ移動します。
↑	カーソルを上へ移動します。 カーソルが画面の一番上にある場合には、画面を 1 行ずつスクロール・ダウンします。
↓	カーソルを下へ移動します。 カーソルが画面の一番下にある場合には、画面を 1 行ずつスクロール・アップします。
→	カーソルを左へ移動します。 カーソルが画面の一番左にある場合には、画面を 1 項目分右へスクロールします。
←	カーソルを右へ移動します。 カーソルが画面の一番右にある場合には、画面を 1 項目分左へスクロールします。
Enter	<ul style="list-style-type: none"> <li>入力データを確定します。</li> <li>デフォルト・プッシュ・ボタンを押します。</li> </ul>
F1	ヘルプ・ウインドウをオープンします。
F2	プログラム実行を強制的に停止させます。 [実行]メニュー→[ストップ]選択時と同様の動作です。

キー	機能
F3	CPU をリセットします。[実行]メニュー→[CPU リセット] 選択時と同様の動作です。
F4	CPU をリセットしてからプログラム実行します。 [実行]メニュー→[リスタート] 選択時と同様の動作です。
F5	プログラム実行します。[実行]メニュー→[ 継続して実行 ] 選択時と同様の動作です。
F6	ソース・テキスト・ウインドウ, または逆アセンブル・ウインドウのカーソル位置までプログラムを実行します。 [実行]メニュー→[カーソル位置まで実行] 選択時と同様の動作です。
F7	呼び出し元に戻るまで実行します。 [実行]メニュー→[リターン・アウト] 選択時と同様の動作です。
F8	ステップ実行します。[実行]メニュー→[ステップ・イン] 選択時と同様の動作です。
F9	ソース・テキスト・ウインドウ, または逆アセンブル・ウインドウのカーソル位置にブレーク・ポイントを設定します。 [実行]メニュー→[ブレークポイント] 選択時と同様の動作です。
F10	Next ステップ実行します。 [実行]メニュー→[ネクスト・オーバー] 選択時と同様の動作です。
F11	ソフトウェア・ブレーク・ポイントの設定/削除を行います。 [実行]メニュー→[ソフトウェア・ブレークポイント]と同様の動作です。
Shift+End	選択範囲を行末まで広げます。
Shift+Home	選択範囲を行頭まで広げます。
Shift+ ←	選択範囲を一文字左へ広げます。
Shift+ →	選択範囲を一文字右へ広げます。
Shift+F6	ソース・テキスト・ウインドウ, または逆アセンブル・ウインドウのカーソル位置からプログラム実行します。 [実行]メニュー→[カーソル位置から実行] 選択時と同様の動作です。
Shift+F9	CPU をリセットします。 [実行]メニュー→[CPU リセット] 選択時と同様の動作です。
Ctrl+End	最終行を表示します。カーソルも同時に最終行の行末へ移動します。
Ctrl+Home	先頭行を表示します。カーソルも同時に先頭行の行頭へ移動します。
Ctrl+ ←	カーソルを1語左へ移動します。 カーソルが画面の一番左にある場合には, 画面を1項目分右へスクロールします。
Ctrl+ →	カーソルを1語右へ移動します。 カーソルが画面の一番右にある場合には, 画面を1項目分左へスクロールします。
Ctrl+F5	設定されているブレーク・ポイントを無視してプログラムを実行します。 [実行]メニュー→[ブレークせずに実行]と同じ機能です。
Ctrl+F9	ソース・テキスト・ウインドウ, または逆アセンブル・ウインドウのカーソル位置のアドレスをPCに設定します。 [実行]メニュー→[PC 値変更] 選択時と同様の動作です。
Ctrl+A	イベント・マネージャに登録されているすべてのイベントを選択します。 [表示]メニュー→[すべてのイベントを選択] 選択時と同様の動作です。
Ctrl+C	選択した文字列をコピーして, クリップ・ボード・バッファに入れます。

キー	機能
Ctrl+D	カレント・ウィンドウで選択しているデータ値をジャンプ先アドレスとして、そのアドレスから逆アセンブル表示します。逆アセンブル・ウィンドウをオープンします。 [ジャンプ]メニュー→[アセンブル]選択時と同様の動作です。
Ctrl+E	PM+ を動作している場合、アクティブなソース・テキスト・ウィンドウで表示しているソース・ファイルを、PM+ で指定されたエディタでオープンします。 [編集]メニュー→[ソースの修正]選択時と同様の動作です。
Ctrl+G	検索を行います。カレント・ウィンドウに応じて、各サーチ・ダイアログをオープンします。 [表示]メニュー→[検索...]選択時と同様の動作です。
Ctrl+J	表示位置を移動します。カレント・ウィンドウに応じて、各指定ダイアログをオープンします。 [表示]メニュー→[移動...]選択時と同様の動作です。
Ctrl+M	カレント・ウィンドウで選択しているデータ値をジャンプ先アドレスとして、そのアドレスからメモリ内容を表示します。メモリ・ウィンドウオープンします。 [ジャンプ]メニュー→[メモリ...]選択時と同様の動作です。
Ctrl+O	表示ファイルまたは、ソース・ファイルやテキスト・ファイルをロードします。表示ファイル・ロード・ダイアログをオープンします。ファイルの拡張子によって動作が異なります。 表示ファイルのとき：該当するウィンドウに表示します。 それ以外るとき：ソース・テキスト・ウィンドウに表示します。 [ファイル]メニュー→[開く...]選択時と同様の動作です。
Ctrl+S	カレント・ウィンドウの表示内容を表示ファイルにセーブします。 [ファイル]メニュー→[名前を付けて保存...]選択時と同様の動作です。
Ctrl+U	カレント・ウィンドウで選択しているデータ値をジャンプ先アドレスとして、該当するソース・テキスト、およびソース行を表示します。ソース・テキスト・ウィンドウをオープンします。 [ジャンプ]メニュー→[ソース]選択時と同様の動作です。
Ctrl+V	クリップ・ボード・バッファの内容をテキスト・カーソル位置に貼り付けます。
Ctrl+W	指定したデータの内容を一時的に表示します。 クイック・ウォッチ・ダイアログをオープンします。 [表示]メニュー→[クイック・ウォッチ...]選択時と同様の動作です。
Ctrl+X	選択した文字列を切り取ってクリップ・ボード・バッファに入れます。 [編集]メニュー→[切り取り...]選択時と同様の動作です。
Ctrl+Shift+ ←	選択範囲を 1 語左へ広げます。
Ctrl+Shift+ →	選択範囲を 1 語右へ広げます。

# 付録 D メッセージ

- メッセージ表示形式
- メッセージの種類
- メッセージ一覧

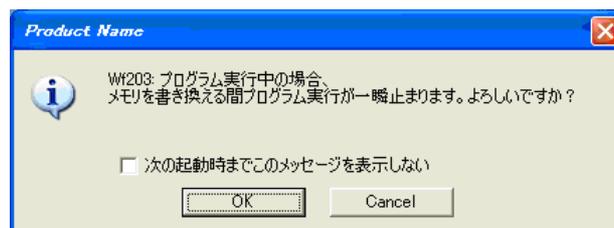
## D.1 メッセージ表示形式

メッセージは、エラー／ワーニング・ダイアログに出力されます。

エラー／ワーニング・ダイアログのオープン中に F1 キーを押すことにより、関連するオンライン・ヘルプが表示されます。

また、[次の起動時までこのメッセージを表示しない]チェック・ボックス付ダイアログのチェックを有効にすることにより、ID850QB 終了時まで該当メッセージは表示されなくなります。なお、ID850QB を起動しなおすことで、このメッセージは表示されるようになります（デフォルトはチェックなし）。

図 D - 1 エラー / ワーニング・ダイアログ



## D.2 メッセージの種類

ID850QB が出力するメッセージには次の種類があります。

表 D - 1 メッセージの種類

エラー種別	意味
A	Abort Error (致命的エラー) 処理を中断し、デバッグを終了します。 このエラーが発生した場合には、デバッグ作業を続行することはできません。
F	Fail (失敗) 処理を中断し、このときオープンしているウインドウ/ダイアログをクローズします。
W	Warning (警告) 処理は中断しますがこのときオープンしているウインドウ/ダイアログはクローズしません。

## D.3 メッセージ一覧

< X0000~ > < X1000~ > < X2000~ > < X3000~ > < X4000~ > < X5000~ > < X6000~ > < X7000~ > < X8000~ > < X9000~ > < Xa000~ > < Xb000~ > < Xc000~ > < Xd000~ > < Xe000~ > < Xf000~ >

### (1) X0000~

F0002: この機能はサポートしていません。
F0100: インサーキット・エミュレータと通信できません。インタフェース・ボード用のデバイス・ドライバが正しくインストールされているか確認してください。 1) エミュレータの電源、ケーブルの接続などを確認してください。 2) ドライバが正常にインストールされていない可能性があります。ドライバの再インストールをしてください。
A0101: イニシャライズ・ファイル (EXPC.INI) が見つかりません。
A0102: ホスト名が見つかりません。
F0103: インサーキット・エミュレータへの送信ができません。インサーキット・エミュレータの電源、ケーブルの接続、I/O アドレスの設定などを確認してください。
F0104: インサーキット・エミュレータから応答がありません。インサーキット・エミュレータの電源、ケーブルの接続、I/O アドレスの設定などを確認してください。
A0105: デバイス・ファイル (file name) を正しく読めませんでした。 1) 必要なファイルが壊れている可能性があります。デバイス・ファイルを再インストールしてください。
A0106: 受信データに異常がありました。 1) インサーキット・エミュレータの電源、ケーブルの接続、インタフェース・ボードの設定などを確認して、デバッグを再起動してください。
A0107: インサーキット・エミュレータとの通信でエラーが発生しました。
A0108: イニシャライズ・ファイル (EXPC.INI) を正しく読めません。
A0109: インサーキット・エミュレータと通信できません。デバッグを終了させ、インサーキット・エミュレータの電源、ケーブルの接続などを確認し、デバッグを再起動してください。 1) USB 通信に異常が起きた (パワー、ケーブルの切断等)、または IECUBE の故障の可能性がありません。[IECUBE]
A010a: デバッグとユーティリティの同時起動はできません。 1) QB-Programmer, MINICUBE2 自己診断ツール, OCD Checker が起動しています。これらのユーティリティをすべて終了させてください。 2) ID78K0-QB, ID78K0S-QB, または ID78K0R-QB が MINICUBE2 を使用しています。これらを終了させてください。 3) MULTI, または C-SPY が MINICUBE2 を使用しています。これらを終了させてください。

<p>A01a0: エバチップから応答がありません。RESET, WAIT, HLDRQ などの信号やクロック信号に異常がないか確認してください。</p> <p>CPU から応答がありません。RESET, WAIT, HLDRQ などの信号やクロック信号に異常がないか確認してください。</p> <p>1) HOLD 信号, WAIT 信号, クロック信号などを確認してください。IOR 値 (または SFR 値) は正しくない可能性があります。</p>
<p>A01a1: ie703000.ie を正しく読めませんでした。</p>
<p>A01a2: BK ボードが接続されていません。</p>
<p>A01a3: エミュレーション・ボードが接続されていません。</p>
<p>A01a4: インサーキット・エミュレータ内のボード構成に矛盾があります。</p>
<p>A01a5: POD/EM1 ボードが接続されていません。</p>
<p>A01a6: イグゼキュタ起動中です。</p> <p>1) MULTI, または C-SPY が MINICUBE2 を使用しています。これらを終了させてください。</p>
<p>A01a8: コンフィギュレーション・ファイル (LV8HW.INI) が見つかりません。</p>
<p>A01ad: インタフェース・ボード用のデバイス・ドライバを更新して下さい。</p> <p>1) デバイス・ドライバが古い可能性があります。最新のデバイス・ドライバをインストールしてください。</p>
<p>A01ae: コンフィギュレーション・ファイル (LV8HW.INI) を正しく読めません。</p>
<p>A01af: モニタ・コマンドの実行に失敗しました。</p>
<p>A01b0: モニタ・プログラムと通信できません。通信ポートが利用可能かどうか, CPU ボードの設定, ケーブルの種類などを確認してください。</p>
<p>A01b1: モニタ・プログラムと通信できません。デバッグを終了させ, CPU ボードの電源, ケーブルの接続などを確認し, デバッグを再起動してください。</p>
<p>A01b2: エミュレータのファームウェアのバージョンが最新ではありません。ユーティリティで最新のファームウェアに更新してください。</p> <p>1) MQB2UTL でファームウェアをアップデートしてください。</p>
<p>F0200: ベリファイ・エラーが起きました。メモリに書き込みができません。</p> <p>1) 外部メモリが設定されていないため, 外部メモリに対してアクセスができませんでした。ダウンロードする前に, <a href="#">IOR ウィンドウ</a>, または <a href="#">フック・プロシージャ</a> を使用して, 外部メモリのアクセスに必要なレジスタの値を変更してください。</p>
<p>F02a0: バス・ホールド中です。</p> <p>1) CPU がバス・ホールド状態です。デバッグのリセットを行ってください。</p>
<p>F02a2: 強制ブレークはできません。</p>

F02a3: リセットが継続中です。 1) リセット解除時に、MINICUBE2 の RESET 端子がハイ・レベルにならない場合に発生します。 以下の原因が考えられます。[MINICUBE2] ・ターゲット・システムとの接続に問題がある (RESET)。
F02d2: トレース・フレームのメモリが不足しています。
F0300: ユーザ・プログラム実行中です。
F0301: ユーザ・プログラム停止中です。
F0302: ユーザ・プログラム・トレース中です。
F0303: トレース測定は行われていません。
F0304: トレーサは OFF になっています。
F0306: トレース・ブロックはありません。
F0307: イベント条件がありません。
F0308: タイマ測定は行われていません。
F0309: トリガ・フレームはありません。
F030a: トレーサ停止中です。
F030b: 参照・削除しようとしたスナップ・イベントが未登録です。
F030c: 参照・削除しようとしたスタブ・イベントが未登録です。
F030d: タイマ動作中です。
F030e: メモリ・コピー範囲が重なっています。
F030f: トレースはすでに ON 状態です。
F0310: イベント条件が設定されていません。
F0311: 有効タイマ・イベント条件の最大個数を越えました。
F0312: このタイマ・イベントは設定されていません。
F0313: マッピング設定に誤りがあります。 1) <a href="#">コンフィギュレーション・ダイアログ</a> でマッピング設定を確認してください。また、外部メモリに対してマッピングを行った際には、ダウンロードする前に、 <a href="#">IOR ウィンドウ</a> 、または <a href="#">フック・プロシージャ</a> を使用して、外部メモリのアクセスに必要なレジスタの値を変更してください。
F0314: デレイ・トリガ条件が設定されているので、トレース・デレイ・モード以外は使用できません。
F0315: デレイ・トリガ条件は、トレース・デレイ・モード以外では設定できません。
F0316: 指定可能なマッピング数を越えました。
F03a0: ターゲットの電源が OFF です。 1) ターゲット・システムの電源を確認してください。インサーキット・エミュレータとターゲット・ボードの接続ケーブルを確認してください。または、ターゲット・ボードのコネクタに VDD 信号が来ているかを確認してください。
F03a1: ステップ実行中です。

F03a2: タイマ、トレーサ動作中です。
F03a3: 統合イベントと実行系イベントが混在しています。
F03d0: バッグ・トレース実行中です。
F03d1: バッグ・トレース停止中です。
F03d2: バッグ・トレース実行停止位置がバッグ・トレース可能な範囲を越えました。
F03d3: 統合イベントの Phase1 以外に Register ステータス、または、Memory ステータスは設定できません。
F03d4: バッグ・トレース情報がありません。
F03d5: 最後の命令でバッグ・ステップができません。
F0400: 指定した条件が不正です。 1) 使用しているインサーキット・エミュレータと <a href="#">コンフィギュレーション・ダイアログ</a> の設定が合っていない可能性があります。Chip 選択を確認してください。
F0401: タイマ測定結果がオーバーフローしました。
F0402: パス・カウントを設定したイベント条件の数が多すぎます。
F0403: アドレス範囲指定条件の最大個数を越えています。
F0404: 同時に使用可能なイベント条件の設定個数を越えています。
F0405: スナップ・イベント条件の最大個数を越えています。
F0406: スタブ・イベント条件の最大個数を越えています。
F0407: 初期化データ数が初期化領域を越えています。
F0408: 検索データが 16 バイトを越えています。
F0409: 検索データのサイズが検索範囲のサイズを越えています。
F040a: 統合イベント設定数の最大個数を越えました。
F04a0: ソフトウェア・ブレイク条件の最大個数を越えました。
F04a1: エミュレーション用のメモリが足りません。
F04a2: パス・サイズの区切りが最大数を越えました。
F04a3: 実行イベント条件の最大個数を越えました。
F04a4: パス・イベント条件の最大個数を越えました。
A0600: バッファの取得に失敗しました。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
A0601: オペレーティングシステムのリソースが不足しています。
F0b20: 使用できないイベントを使用しました。
F0b61: セクション・トレースイベント条件の最大個数を越えました。
F0b66: 実行前ブレイクとソフトウェアブレイクは同時に使用できません。 1) ソフトウェア・ブレイクの実装に実行前ブレイクを使用しているためです。[MINICUBE]
F0b80: ハードウェア異常のためリセットで停止しました。

F0c00: モニタ・ファイルの読み込みに失敗。 1) 必要なファイルが壊れている可能性がありますので、デバッグを再インストールしてください。
A0c01: レジスタのアクセス中に CPU からの応答がなくなりました。 1) クロック信号などを確認してください。レジスタ値が正しくない可能性があります。
A0c02: メモリのアクセス中に CPU からの応答がなくなりました。 1) HOLD 信号, WAIT 信号, クロック信号などを確認してください。メモリの値は正しくない可能性があります。
A0c03: I/O レジスタのアクセス中に CPU からの応答がなくなりました。 1) HOLD 信号, WAIT 信号, クロック信号などを確認してください。I/O レジスタ値は正しくない可能性があります。
F0c04: 外部空間にあるフラッシュ・メモリのデータベース・ファイルが見つかりません。
F0c20: ガード・エリア領域にアクセスしようとして失敗しました。
F0c21: メモリが NO READY 状態です。
W0c22: NO READY 状態をキャンセルしました。
F0c23: バス・ホールドが継続中です。 1) ターゲット・ボードの設定が正しいかどうかを確認してください。または HOLD 端子をマスクしてください。
F0c24: デバッグモードに移行できません。 1) クロック信号を確認してください。クロック停止, 低速クロックが考えられます。 2) リセット解除後に, モニタ・プログラムから応答がない場合に発生します。[MINICUBE2] 以下の原因が考えられます。 ・ターゲット・システムとの接続に問題がある (FLMD0)
F0c25: フラッシュ・マクロ・サービス用 ROM をアクセス, またはステップ実行しました。 1) 継続実行, または CPU リセットを行ってください。
F0c26: FLMD 端子が書き込み禁止状態です。 1) FLMD の状態が書き込み許可状態になっていません。FLMD0, FLMD1 端子の状態を確認してください。
F0c27: セキュリティ・フラグが書き込み禁止状態です。 1) フラッシュ・メモリのセキュリティ・フラグで書き込み禁止, ブロック消去禁止, チップ消去禁止になっています。フラッシュ・メモリに書き込むことはできません。
F0c28: 内部 RAM が不足しているためフラッシュ・メモリの書き込みができません。 1) 内部 RAM サイズが 4K バイト未満のため, フラッシュ・セルフ書き込みができません。
F0c29: フラッシュ・メモリのブランクチェックに失敗しました。
F0c2a: フラッシュ・メモリの消去に失敗しました。
F0c2b: フラッシュ・メモリの書き込みに失敗しました。
F0c2c: フラッシュ・メモリの内部ペリファイに失敗しました。
F0c2d: フラッシュ・メモリの書き込みに失敗しました。

F0c2e: フラッシュ・マクロ・サービスからの応答がありません。
F0c2f: フラッシュ・マクロ・サービスからの応答が正しくありません。
F0c30: フラッシュ I/O レジスタ操作禁止の設定を解除する必要があります。
F0c31: STOP モードが継続中です。強制ブレークはできません。STOP モードを解除するか CPU リセットして下さい。
F0c32: フラッシュ・メモリへの書き込みはシングル・チップ・モード 0 で行ってください。
F0c33: オンチップ・デバッグを使用禁止にしようとした。 1) ID コードの最上位ビットに 0 を書き込もうとした。 最上位ビットに 0 を書き込むとオンチップ・デバッグができるため、0 を書き込まないでください。[MINICUBE2]
F0c34: オンチップ・デバッグで使用する予約領域への書き込みはできません。 1) オンチップ・デバッグで使用する予約領域に許可されていない値を書き込もうとした。 予約領域は書き換えしないでください。[MINICUBE2] 2) リセット・エントリのコードを、モニタ・プログラムへ分岐するコードに書き換えることができませんでした。リセット・エントリのコードを指定されたコードに変更してください。[MINICUBE2]
F0c35: フラッシュ書き込み用のプログラムを内蔵 RAM に書き込めません。
F0c36: 内蔵 ROM サイズがデバイスのデフォルト・サイズになっていません。
F0c37: 電圧が低すぎてフラッシュ書き込みができません。
F0c38: 拡張モニタ領域が使用中のためモニタ・コードを書き込めません。
F0c39: RRM に失敗しました。
F0c3a: データ・フラッシュ領域の書き込みはサポートしていません。
F0c3b: データ・フラッシュ以外のフラッシュ環境のため、データ・フラッシュ・メモリの書き込みができません。
F0c3c: 外部空間にあるフラッシュ・メモリの情報が未設定です。
F0c3d: 外部空間にあるフラッシュ・メモリの消去に失敗しました。
F0c3e: 外部空間にあるフラッシュ・メモリの書き込みに失敗しました。
F0c40: 有効なイベント条件のステータスは変更できません。
F0c41: カバレッジ実行中です。
F0c42: デバッグ・モードへの移行に失敗しています。CPU リセットして下さい。
F0c43: エミュレータの接続ができません。 1) デスクトップ・パソコン使用で、PC カードを複数枚挿入している場合の切り替え設定ミスの可能性があります。設定を確認してください。または故障の可能性があります。[MINICUBE] 2) エミュレータの電源とケーブルの接続を確認してください。デスクトップ・パソコン使用で、PC カードを複数枚挿入している場合の切り替え設定ミスの可能性があります。設定を確認してください。[IECUBE] 3) エミュレータの電源とケーブルの接続を確認してください。[MINICUBE2]
F0c44: トレース・パケット・データの取りこぼしが発生しました。

F0c45: パワー OFF リセットエミュレーション中はプログラムの実行はできません。
F0c46: フラッシュ・セルフ・エミュレーション中は、内蔵 ROM サイズまたは内蔵 RAM サイズの変更と RAM モニタと DMM の機能を使用できません。
F0c47: ROM コレクション・エミュレーションは既に有効です。
F0c60: 実行前イベントはブレイク条件以外に設定できません。 1) 実行後イベントを使用してください。
F0c61: ハードウェア・ブレイクで使用できないイベント番号の登録はできません。
F0c62: ハードウェア・ブレイクで使用するイベント番号は使用できません。
F0c63: 統合イベント条件は設定できません。
F0c64: ROM エミュレーション RAM 領域は複数設定できません。
F0c67: フラッシュ・メモリのブロック間の書き込みはできません。
F0c68: ROM コレクション・エミュレーション可能な定形コードになっていません。
F0c69: データ・フラッシュ領域外のアドレスが指定されました。
F0c6a: 外部空間にあるフラッシュ・メモリ領域外のアドレスが指定されました。
F0c70: DCU にアクセスできません。 1) デバイス・ファイルの選択を間違えている可能性があります。 <a href="#">コンフィギュレーション・ダイアログ</a> の Chip 選択でターゲット・チップに対応したデバイス・ファイルを選択してください。チップの電源を確認してください。信号線 (DCK,DMS,DDI,DDO,DRSTZ) の接続を確認してください。N-Wire Checker の DCK wave form test でノイズ・レベルを確認してください。[MINICUBE] 2) IE が故障している可能性があります。[IECUBE]
F0c71: リセットができません。 1) クロック信号を確認してください。クロック停止、低速クロックが考えられます。 2) リセット時に、MINICUBE2 の RESET 端子がロウ・レベルにならない場合に発生します。 以下の原因が考えられます。[MINICUBE2] ・ターゲット・システムとの接続に問題がある (RESET)。
F0c72: モニタ・メモリにアクセスできません。 1) <a href="#">コンフィギュレーション・ダイアログ</a> の Main OSC の値を見直してください。また、N-Wire Checker の DCK wave form test でノイズ・レベルを確認してください。また、チップ内部に問題がある可能性もあります。[MINICUBE] 2) <a href="#">コンフィギュレーション・ダイアログ</a> の Main OSC の値を見直してください。それでも解決しない場合には、IE が故障している可能性があります。[IECUBE] 3) モニタ・プログラムの書き込みに失敗した場合に発生します。 以下の原因が考えられます。 ・ターゲット・システムとの接続に問題がある。(通信端子, FLMD0) ・ <a href="#">コンフィギュレーション・ダイアログ</a> の UART か CSI の選択が間違っている。 ・ <a href="#">コンフィギュレーション・ダイアログ</a> で入力した発振周波数 (Clock) とターゲット・システム上のデバイスの発振周波数が異なっている。[MINICUBE2]

<p>F0c73: モニタ実行できません。</p> <ol style="list-style-type: none"> <li>1) N-Wire Checker の DCK wave form test でノイズ・レベルを確認してください。または、チップ内部に問題がある可能性もあります。[MINICUBE]</li> <li>2) IE が故障している可能性があります。[IECUBE]</li> <li>3) ユーザ・プログラム実行中に、ID コードが変更された場合に発生します。 ユーザ・プログラムでは、ID コードの変更をしないでください。[MINICUBE2]</li> </ol>
<p>F0c74: CPU レジスタにアクセスできません。</p> <ol style="list-style-type: none"> <li>1) N-Wire Checker の DCK wave form test でノイズ・レベルを確認してください。または、チップ内部に問題がある可能性もあります。[MINICUBE]</li> <li>2) デバイス・ファイルの選択を間違えている可能性があります。コンフィギュレーション・ダイアログの Chip 選択でターゲット・チップに対応したデバイス・ファイルを選択してください。それでも解決しない場合には、IE が故障している可能性があります。[IECUBE]</li> <li>3) ターゲット・システム上のデバイスに対応していないモニタ・ファイルでデバッグを起動しようとしました。デバッグ上のデバイス選択が正しいかどうかを確認してください。[MINICUBE2]</li> </ol>
<p>F0c75: デバッグモードへの移行に失敗しています。CPU リセットして下さい。</p>
<p>F0c76: DCU アクセス開始時の初期状態異常です。[V850]</p> <ol style="list-style-type: none"> <li>1) DCU アクセス開始時の初期状態異常です（リセット状態で立ち上がらない）。 信号線 (DCK,DMS,DDI,DDO,DRSTZ) の接続を確認してください。N-Wire Checker の DCK wave form test でノイズ・レベルを確認してください。[MINICUBE]</li> <li>2) デバイス・ファイルの選択を間違えている可能性があります。コンフィギュレーション・ダイアログの Chip 選択でターゲット・チップに対応したデバイス・ファイルを選択してください。それでも解決しない場合には、IE が故障している可能性があります。[IECUBE]</li> </ol>
<p>F0c77: DCU アクセス異常です。</p> <ol style="list-style-type: none"> <li>1) DCU アクセス異常です（ベリファイ・エラー）。 信号線 (DCK,DMS,DDI,DDO,DRSTZ) の接続を確認してください。N-Wire Checker の DCK wave form test でノイズ・レベルを確認してください。[MINICUBE]</li> <li>2) IE が故障している可能性があります。[IECUBE]</li> </ol>
<p>F0c78: トレース・データの読み込みに失敗しました。</p>
<p>F0ca0: インサーキット・エミュレータと通信ができません。インサーキット・エミュレータの電源、ケーブルの接続、I/O アドレスの設定などを確認してください。</p>
<p>F0ca1: モニタ・ファイルが見つかりませんでした。</p> <ol style="list-style-type: none"> <li>1) 必要なファイルが壊れている可能性がありますので、デバッグを再インストールしてください。</li> <li>2) サポートされていないデバイスを選択した場合に発生します。対応デバイスかを確認してください。</li> </ol>

F0ca2: オンチップ・デバッグに対応していないデバイス・ファイルです。 1) オンチップ・デバッグ非対応のデバイス・ファイルで起動しようとした。デバイス・ファイルが古い可能性があります。最新のデバイス・ファイルをインストールしてください。[MINICUBE] [MINICUBE2] 2) IE が故障している可能性があります。[IECUBE]
F0ca3: デバイス・ファイルのオンチップ・デバッグ情報に未サポートの情報が含まれています。 1) デバイス・ファイルのオンチップ・デバッグ情報に不明フラグがあります。EXEC モジュールが古い可能性があります。最新の EXEC モジュールをインストールしてください。
F0ca4: IECUBE に対応していないデバイス・ファイルです。 1) IECUBE 非対応のデバイス・ファイルで起動しようとした。デバイス・ファイルが古い可能性があります。最新のデバイス・ファイルをインストールしてください。
F0caf: トレース・ブロックを越えた移動はできません。

## (2) X1000~

A1000: インサーキット・エミュレータの初期化に失敗しました。
A1001: 指定番号のエントリが存在しません。
A1002: 内部 RAM がリロケートできません。
F1003: リロケーションアドレスが不正です。
F1004: 指定条件が不正です。
A1005: 属性が無効です。
F1006: アドレスが不正です。
A1007: インサーキット・エミュレータの代替メモリが不足しています。
A1008: テーブル用のメモリが不足しています。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
A1009: すでに初期化済みです。
A100a: まだ初期化されていません。
F100b: ユーザ・プログラム実行中です。
F100c: バス・サイズの異なる領域に重複設定しようとした。
F100d: バス・サイズの定義可能な最大値を越えています。
F100e: バス・サイズの区切りが最大値を越えました。
W100f: ターゲットの電源が OFF です。
F1010: マップの範囲指定に誤りがあります。
F1011: 内蔵 ROM, RAM の設定に失敗しました。
F1012: この機能はサポートしていません。

F1013: 端子名が存在しません。
W1014: 値が登録されていません。
A1015: プログラマブル IOR がありません。
F1016: プログラマブル IOR は移動できません。 1) 必要なファイルが壊れている可能性がありますので、最新のデバイス・ファイルを再インストールしてください。
F1017: I/O Protect マッピングができるのは、ターゲット属性のみです。
F1018: 内蔵 ROM サイズが不正です。
F1019: 内蔵 ROM サイズまたは内蔵 RAM サイズが不正です。
F101d: データ・フラッシュ領域は移動できません。
F101e: データ・フラッシュ領域がありません。
A10ff: 通信エラーです。インサーキット・エミュレータとの通信ができません。
A1dbe: デバッグ内部でエラーが発生しました。

**(3) X2000~**

F2000: I/O レジスタ名が不正です。
A2001: アドレスが不正です。
F2002: ユーザ・プログラム実行中です。
F2003: I/O レジスタ数に誤りがあります。
F2004: ビット位置が不正です。
W2005: Read Protect 属性の I/O レジスタが指定されました。
F2006: 非公開の I/O レジスタが指定されました。
F2007: 書き込み禁止、または読み出し禁止の I/O レジスタが指定されました。
F2008: 指定された I/O レジスタは存在しません。
A2009: デバイス・ファイルが破損しているか、ファイルに誤りがあります。
F200a: I/O レジスタに対する設定値に誤りがあります。
A200b: コピーに失敗しました。
A200c: メモリが足りません。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
W200d: この I/O レジスタには初期値の設定がありません。
F200e: I/O レジスタ領域にアクセスしようとしてしました。
A20ff: 通信エラーです。インサーキット・エミュレータとの通信ができません。
A2222: 指定条件が不正です。

## (4) X3000~

F3000: マッピングされていないアドレスにアクセスしようとした。 1) プログラムの配置アドレスと、デバッガのアドレスの設定が合っていないと考えられます。コンパイル時のリンク・ディレクティブ・ファイルで指定したプログラムの配置アドレスにあわせて、 <a href="#">コンフィギュレーション・ダイアログ</a> で、外部メモリのマッピング設定をしてください。また、外部メモリに対してマッピングを行った際には、ダウンロードする前に、 <a href="#">IOR ウィンドウ</a> 、または <a href="#">フック・プロシージャ</a> を使用して、外部メモリのアクセスに必要なレジスタの値を変更してください。
F3001: メモリ内容が一致しませんでした。
F3002: 開始アドレスが不正です。
F3003: 終了アドレスが不正です。
F3004: 開始、終了アドレスともに不正です。
F3005: 指定条件が不正です。
F3006: ユーザ・プログラム実行中です。
F3007: ベリファイ・エラーです。
F3008: 指定条件がありません。
F3009: パラメータ・サイズがアクセス・サイズ・アライメントになっていません。
F300a: 指定アドレスがアクセス・サイズ・アライメントになっていません。
F300b: 指定アドレス (Source) がアクセス・サイズ・アライメントになっていません。
F300c: 指定アドレス (Destination) がアクセス・サイズ・アライメントになっていません。
F300d: 終了アドレスが不正です。
F300e: 指定範囲内でアクセス・サイズが異なります。
F300f: 指定範囲内 (Source) でアクセス・サイズが異なります。
F3010: 指定範囲内 (Destination) でアクセス・サイズが異なります。
F3011: 指定範囲 (Source) と指定範囲 (Destination) でアクセス・サイズが異なります。
A3012: メモリが足りません。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
F3013: DMM の書き込みに失敗しました。
F3014: マッピング領域を越えました。
F3015: 処理が中断されました。
F3016: この機能はサポートしていません。
A30ff: 通信エラーです。インサーキット・エミュレータとの通信ができません。

## (5) X4000~

F4000: 指定されたイベントは削除できません。 1) 指定されたイベントは他の条件で使用されているため、削除できません。他での使用を無効にしてから削除してください。
F4001: 指定したテーブル番号が不正です。
F4002: 開始アドレスが不正です。
F4003: 終了アドレスが不正です。
F4004: ステータスが不正です。
F4005: データが不正です。
F4006: すでに使用済みのイベント番号を指定しようとしてしました。
F4007: 同じ種類のイベント最大登録個数を越えています。
F4008: 指定したイベントは登録されていません。
F4009: データ・サイズが不正です。
F400a: モードが不正です。
F400b: 設定値が不正です。
F400c: セクション・トレース条件に統合イベント条件は使用できません。
F400d: 同一イベントを 32,767 個以上登録しようとしてしました。
F400e: イベント条件の指定がありません。
F400f: 統合イベント条件が不正です。
F4010: 指定した関数が見つかりません。
A4011: メモリが足りません。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
F4012: タイマが無効になっています。
W4013: アクセス・サイズとマッピングしているバス・サイズが異なります。
F4014: ソフトウェア・ブレークが使用できません。
F4015: アドレス範囲指定しているイベント条件は使用できません。
F4016: イベント条件の変更ができません。
F4017: 奇数番地のワード・アクセスはできません。
A4018: メモリが足りません。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
F4019: この機能はサポートしていません。
F401a: イベントがありません。
F401b: タグイベントは使用できません。

W401c: その領域にソフトウェア・ブレークは設定できません。
F401d: タイマの開始イベントと終了イベントを同じ設定にはできません。
F401e: トレース条件の最大個数を越えました。
F401f: パス・カウントは設定できません。
F4020: 実行前イベントではアドレス範囲を設定できません。
F4021: イベント条件の最大個数を越えました。
F4022: ソフトウェア・DMM 条件の最大個数を越えました。
F4023: リアルタイム・コール条件の最大個数を越えました。
F4024: ソフトウェア・ブレーク条件の最大個数を越えました。
F4025: スナップの条件が不完全です。
F4026: 統合イベント条件の Phase1 と Phase2 にはイベント条件を複数設定できません。
F4027: 内蔵 ROM 領域に設定できるソフトウェア・ブレーク条件の最大個数を越えました。
F4318: メモリバンクの設定が不正です。

**(6) X5000~**

A5000: デバイス・ファイルが不正です。
A5001: メモリが足りません。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
A5002: デバイス・ファイルのオープンに失敗しました。
A5003: デバイス・ファイルの読み込みに失敗しました。
A5004: デバイス・ファイルのクローズに失敗しました。
A5005: デバイス・ファイルの形式が違います。 1) 必要なファイルが壊れている可能性がありますので、デバイス・ファイルを再インストールしてください。
A5006: インサーキット・エミュレータの初期化に失敗しました。
A5007: デバイス・ファイルが壊れているか、ファイルに誤りがあります。
F5008: 指定したデバイス・ファイルをオープンできません。 1) 必要なファイルが壊れている可能性がありますので、デバイス・ファイルを再インストールしてください。
F5009: ie703000.ie をオープンできません。
F500a: デバイス・ファイルのバージョンが不正です。 1) 必要なファイルが壊れている可能性がありますので、デバイス・ファイルを再インストールしてください。
W500b: このデバイス・ファイルは内蔵 RAM をリロケートしません。

A500c: イニシャライズ・ファイル (EXPC.INI) を正しく読めません。
A500d: メモリが足りません。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
W500e: デバイス・ファイルに参照しようとしたタグ情報はありません。
A5300: デバイス・ファイルが不正です。
A5301: メモリが足りません。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
A5302: データベース・ファイルのオープンに失敗しました。 1) 必要なファイルが壊れている可能性がありますので、デバッグを再インストールしてください。デバイス・ファイルも再インストールしてください。
A5303: データベース・ファイルの読み込みに失敗しました。
A5304: データベース・ファイルのクローズに失敗しました。
A5305: データベース・ファイルの形式が違います。 1) 必要なファイルが壊れている可能性がありますので、デバッグを再インストールしてください。デバイス・ファイルも再インストールしてください。
A5306: データベース情報は初期化済です。
A5307: データベース情報がありません。
F5308: 指定したデータベース・ファイルをオープンできません。 1) 必要なファイルが壊れている可能性がありますので、デバッグを再インストールしてください。
F5309: データベース・ファイルのバージョンが不正です。 1) 必要なファイルが壊れている可能性がありますので、デバッグを再インストールしてください。デバイス・ファイルも再インストールしてください。

**(7) X6000~**

F6000: カレント関数が存在しません。
F6001: シンボル名が不正です。
F6002: 指定条件が不正です。
F6003: 関数名が不正です。
F6004: 出力バッファのサイズを越えています。
F6005: 式が不正です。

**(8) X7000~**

F7000: モード指定に誤りがあります。
F7001: ユーザ・プログラム実行中です。

F7002: ユーザ・プログラム停止中です。
F7003: トレース実行中です。
F7004: トレース・メモリが OFF になっています。
F7005: 関数リターン・アドレスがありません。ステップ実行が行えません。
W7010: ソース情報がありません。
W7011: ステップ実行結果不明。
A7012: メモリが足りません。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
A70fe: バス・ホールド中です。 1) CPU がバス・ホールド状態です。デバッグのリセットを行ってください。
A70ff: 通信エラーです。インサーキット・エミュレータとの通信ができません。
F7801: ステップ実行の終了待ち状態をキャンセルしました。
F7802: ステップ実行が中断しました。
F7f00: ステップ実行を強制終了しました。
F7f02: ステップ実行を中断しました。
A7f03: 実行／ステップ実行のキャンセルに失敗しました。
F7f04: マッピングされていない領域を実行しようとしてしました。
F7f05: この機能はサポートしていません。

**(9) X8000~**

F8000: 指定したファイルは見つかりませんでした。
F8001: 行番号が不正です。
F8002: カレント情報がセットされていません。
F8003: アドレスが不正です。
F8004: この機能はサポートしていません。

**(10) X9000~**

A9000: 指定したレジスタ・シンボルは存在しません。
A9001: 指定したレジスタ・シンボル ID は存在しません。
F9002: 設定値が不正です。
A9003: 指定条件が不正です。
A9004: レジスタのサイズが大きすぎます。
F9005: この機能はサポートしていません。

## (11) Xa000~

Fa001: 式が不正です。
Fa002: 開始／終了アドレスの指定が逆です。
Fa003: 指定したソース・パス情報が不正です。
Fa004: 式が長すぎます。
Aa005: メモリが足りません。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
Fa006: 引数が不正です。
Fa007: プログラム番号が不正です。
Fa008: ソース・パスがセットされていません。
Fa009: ファイルは見つかりませんでした。
Fa00a: ファイルのオープンに失敗しました。 1) ファイルが壊れているか、またはファイルが存在しません。ファイルを作成し直してください。
Aa00b: ファイルのクローズに失敗しました。
Aa00c: ファイルの読み込みに失敗しました。 1) ファイルが壊れているか、またはファイルが存在しません。ファイルを作成し直してください。
Fa00d: ロード・モジュールのソース・ファイルではありません。
Fa00e: 行番号が不正です。
Fa00f: 変数が存在しません。
Aa010: 通信エラーです。インサーキット・エミュレータとの通信ができません。
Fa011: レジスタにアクセスできません。
Fa012: メモリ（変数）にアクセスできません。
Aa013: ファイルの読み込みに失敗しました。
Fa014: バイナリ・ファイルを開こうとしました。
Fa015: テンポラリ・パスの取得に失敗しました。 1) ディスクがいっぱいです。不要なファイルを削除または移動して、ディスクの空き容量を増やしてください。
Fa016: テンポラリ・ファイルの作成に失敗しました。 1) ディスクがいっぱいです。不要なファイルを削除または移動して、ディスクの空き容量を増やしてください。
Fa017: テンポラリ・ファイルの削除に失敗しました。
Fa020: この機能はサポートしていません。
Fa021: レジスタに割り当てられたシンボルは指定できません。
Fa022: フォルダに使用できない文字が含まれているか、またはフォルダが存在しません。

## (12) Xb000~

Fb000: コマンド行が不正です。
Fb001: ロード・モジュール・ファイルにプログラム情報がありません。
Fb002: ファイルは見つかりませんでした。
Fb003: 関数が見つかりません。
Fb004: 選択した品種（チップ）と異なるロード・モジュールをロードしました。
Fb005: シンボルが見つかりません。 1) アドレスを求めることができませんでした。アドレス情報のある場所を指定してください。
Fb008: 式が不正です。
Ab009: メモリが足りません。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
Fb00a: ロード・モジュール・ファイル内に不正なシンボルがあります。
Fb00b: カレント・プログラムが存在しません。
Fb00c: カレント・ファイルが存在しません。
Ab00d: カレント関数が存在しません。
Ab00e: カレント行が存在しません。
Ab00f: タグが見つかりません。
Ab010: シンボル・テーブルのロードに失敗しました。
Ab011: 行番号が不正です。
Fb012: 行番号が大きすぎます。
Ab015: ファイルの読み込みに失敗しました。 1) ファイルが壊れているか、またはファイルが存在しません。ファイルを作成し直してください。
Ab016: ファイルのオープンに失敗しました。 1) ファイルが壊れているか、またはファイルが存在しません。ファイルを作成し直してください。
Ab017: ファイルの書き込みに失敗しました。 1) ファイルが壊れているか、またはファイルが存在しません。ファイルを作成し直してください。
Ab019: ファイルの読み込みに失敗しました。
Ab01a: ファイルのクローズに失敗しました。
Fb01b: ロード・モジュールのファイル名が長すぎます。
Ab01c: タスク種別のエントリ数が多すぎます。
Fb01d: アドレスが存在しません。
Wb01e: デバッグ情報がありません（Debug Build モードでコンパイルされていません）。
Fb01f: 構造体のメンバが見つかりません。
Fb020: 値が見つかりません。

Fb021: ロード・モジュール・ファイルにデバッグ情報とシンボル情報がありません。 1) シンボル情報が付加されたロード・モジュールを作成するには、Debug Build のビルド・モードでビルドを行ってください。
Fb022: 行番号が不正です。
Ab023: カレント・スタック・フレームがアクティブではありません。
Ab024: セクションが違います。
Fb026: 表示する配列が 4 次元を越えています。
Fb027: ファイルが途中までしかありません。 1) 指定したファイルが壊れている可能性があります。ファイルを作成し直してください。
Fb028: この機能はサポートしていません。
Fb029: アドレスが不正です。
Ab02a: 通信エラーです。インサーキット・エミュレータとの通信ができません。
Fb02b: 現在の PC 値では、スタック・トレースはできません。
Fb02c: 1 関数内の最大ブロック数を越えています。
Fb02d: 関数に渡した引数が不正です。
Fb02e: 指定したファイルがソース・パスから見つかりませんでした。 1) プログラムを停止した際、デバッガがソースを表示しようとしたときにソースが見つかりませんでした。デバッガ・オプション設定ダイアログでソースに対してパスが通っているか確認、もしくは、ソースが .out ファイルと同じディレクトリにあるか確認してください。また、エラー・メッセージが表示される場所で逆アセンブル・ウインドウを参照し、対応するソースにパスが通っているかどうか確認してください。
Fb02f: 最適化により情報が削除されました。
Ab030: モニタがタイム・アウトしました。 1) インサーキット・エミュレータの電源、ケーブルの接続、インタフェース・ボードの設定などを確認して、デバッガを再起動してください。
Ab031: すでにメモリ上に設定されています。
Ab032: スコープ外です。
Ab033: LP をストアしていません。
Fb034: 現在の PC 値からのリターン実行はできません。
Wb036: 現在の PC では指定された変数は参照できません。
Fb037: 行情報が多すぎます。
Fb038: 未対応のコンパイラバージョンのため、デバッグ情報のロードができません。 1) 最新のコンパイラでロード・モジュールを作成し直してください。
Ab039: デバッグ情報のロードに失敗しました。
Ab03a: セクション情報がありません。

Fb040: ロード・モジュール形式ではありません。 1) リンカの実出力ファイルではありません。リンカの実出力前のロード・モジュールではソース・デバッグはできません。リンカの実出力後のロード・モジュールを指定してください。
Ab041: ロード・モジュールに含まれるファイル数がデバッグで扱える上限値を越えました。
Wb042: SYM モジュールは初期化されていません。
Fb32e: 不正なポート番号です。
Fb32f: 不正なポート名です。
Fb330: 指定されたポートのビット位置が不正です。
Fb331: 指定されたインクリメント数が不正です。
Fb332: メモリ・バンク用ポートが未設定です。
Fb333: 不正なバンク番号が指定されました。
Fb334: メモリ・バンク用エリアが未設定です。
Wb335: シンボル情報が長すぎます。

**(13) Xc000~**

Fc001: ファイルのオープンに失敗しました。 1) ファイルが壊れているか、またはファイルが存在しません。ファイルを作成し直してください。
Ac002: ファイルのクローズに失敗しました。
Ac003: ファイルの読み込みに失敗しました。 1) ファイルが壊れているか、またはファイルが存在しません。ファイルを作成し直してください。
Ac004: ファイルの読み込みに失敗しました。
Fc005: 不正な形式のファイルを指定しました。
Fc006: 選択した品種（チップ）と異なるロード・モジュールをロードしました。
Fc007: ロード・モジュールの形式ではありません。 1) リンカの実出力ファイルではありません。リンカの実出力前のロード・モジュールではソース・デバッグはできません。リンカの実出力後のロード・モジュールを指定してください。
Fc008: 古いバージョンのロード・モジュール・ファイル（ELF）です。
Ac009: メモリが足りません。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
Fc00a: マッピングされていないアドレスにアクセスしようとしてしました。
Fc00b: ロード・モジュールがロードされていません。
Fc00c: 引数が不正です。
Fc00d: ユーザ・プログラムを実行中です。
Fc00e: ユーザ・プログラムのトレース中です。

Fc00f: 処理中です。
Ac010: 通信エラーです。インサーキット・エミュレータとの通信ができません。
Fc011: ロード・モジュール・ファイルの形式が違います。
Fc012: チェックサム・エラーが発生しました。
Fc013: アップロードするアドレス範囲が 1M バイトを越えています。
Fc014: ファイルの書き込みに失敗しました。 1) ファイルが壊れているか、またはファイルが存在しません。ファイルを作成し直してください。
Fc015: プログラム番号が不正です。
Fc016: ロード情報がいっぱいです。
Wc017: シンボル情報が重複しています。シンボルのリセットをしてください。
Fc018: ロード・モジュール形式ではありません。 1) リンカの実出力ファイルではありません。リンカの実出力前のロード・モジュールではソース・デバッグができません。リンカの実出力後のロード・モジュールを指定してください。
Fc019: メモリへの書き込みに失敗しました。
Wc01a: BSS 領域がマッピングされていない領域に割り当てられています。 1) プログラムを実行すると、ノンマップ・ブレークが発生する可能性があります。リンク・ディレクティブで内部 RAM に BSS 領域を割り当てるか、デバッガの <a href="#">コンフィギュレーション・ダイアログ</a> で BSS 領域にエミュレーション・メモリ、あるいはターゲット・メモリをマッピングしてください。
Fc01b: プログラマブル IOR のアドレスがありません。 1) 必要なファイルが壊れている可能性がありますので、デバッガを再インストールしてください。
Wc01c: プログラマブル IOR のアドレスが設定と異なります。 1) 必要なファイルが壊れている可能性がありますので、デバッガを再インストールしてください。
Wc01d: 選択した品種（チップ）と異なるロード・モジュールをロードしました。
Fc01e: フラッシュの消去はサポートしていません。
Fc100: この機能はサポートしていません。

**(14) Xd000~**

Ad000: デバッガ内部でエラーが発生しました。
Ad001: メモリが足りません。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
Ad002: イニシャライズ・ファイル（EXPC.INI）を正しく読めません。
Ad003: インサーキット・エミュレータにまだ接続されていません。
Fd004: ダイナミック・リンク・ライブラリが見つかりません。

## (15) Xe000~

Fe000: 引数が不正です。
Fe001: 開始アドレスが不正です。
Fe002: 終了アドレスが不正です。
Fe003: サイズが大き過ぎます。
Fe004: ファイルのオープンに失敗しました。 1) ファイルが壊れているか、またはファイルが存在しません。ファイルを作成し直してください。
Fe005: ファイルの読み込みに失敗しました。 1) ファイルが壊れているか、またはファイルが存在しません。ファイルを作成し直してください。
Fe006: ファイルの読み込みに失敗しました。
Fe007: ファイルの書き込みに失敗しました。 1) ファイルが壊れているか、またはファイルが存在しません。ファイルを作成し直してください。
Ae008: メモリが足りません。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
Fe009: ファイルの形式が違います。
Fe00a: ベリファイ・エラーです。
Fe010: この機能はサポートしていません。

## (16) Xf000~

Af000: メモリが足りません。 1) システムのメモリが不足しています。実行中のアプリケーションや、開いているファイルなどを閉じてください。
Ff000: メモリが足りません。
Ff001: [XXX] は見つかりませんでした。
Wf002: [XXX] は見つかりませんでした。最初から探しますか？
Wf003: 既にサーチ範囲を越えています。
Ff004: 不正なパラメータです。
Ff005: 不正な関数名です。
Ff006: 不正な数値です。
Ff007: 開始／終了アドレスの指定が逆です。
Ff008: 不正なシンボルまたはアドレス式です。
Ff009: [XXX] このファイルは不正な形式です。
Ff100: ディスクに書き込みができない一杯です。
Ff101: ファイルは見つかりませんでした。

Ff102:	ファイルは生成できませんでした。
Ff103:	旧版のプロジェクト・ファイルです。
Ff104:	不正なプロジェクト・ファイル形式です。
Ff105:	このプロジェクト・ファイルは [XXX] 用のファイルです。正しいファイルを指定してください。
Wf106:	プロジェクトの CPU が変更されています。設定を有効にするにはデバッグを一旦終了しなければなりません。終了しますか？
Wf107:	プロジェクトの CPU が変更されています。変更後の CPU で起動しますか？
Wf108:	選択した品種 [XXX] と異なるプロジェクト・ファイル [YYY] を開きました。品種を変更することはできませんが開きますか？
Wf109:	このバージョンのデバッグでは、プロジェクト・マネージャを使用できません。PM+ お使いください。
Wf200:	違いはありませんでした。
Ff201:	不正なメモリ・アドレス範囲です。
Ff202:	ベリファイ・エラーです。 1) 外部メモリが設定されていないため、外部メモリに対してアクセスができませんでした。ダウンロードする前に、 <b>IOR ウィンドウ</b> 、または <b>フック・プロシージャ</b> を使用して、外部メモリのアクセスに必要なレジスタの値を変更してください。
Wf203:	プログラム実行中の場合、メモリを書き換える間プログラム実行が一瞬止まります。よろしいですか？
Wf204:	有効なソフトウェア・ブレークが設定されているため、DMM はできません。
Wf300:	[XXX] の変更内容を保存しますか？
Ff301:	イベント条件で使用されているシンボルは評価できませんでした。
Wf302:	[XXX] を削除しますか？
Wf303:	[XXX] は編集中ですが、[YYY] を削除しますか？
Wf304:	[XXX] は編集中ですが、[YYY] を保存しますか？
Wf305:	[XXX] は既に存在しますが、置き換えますか？
Ff306:	名前が長過ぎます。
Ff307:	他の条件に同じ名前が使われています。
Ff308:	アドレスは省略できません。
Ff309:	不正なアドレス・マスクです。
Ff30a:	不正なデータ・マスクです。
Ff30b:	不正な外部プローブ・マスクです。
Ff30c:	不正な外部プローブ・データです。
Ff30d:	不正なパス・カウントです。
Ff30e:	不正なレジスタ名です。
Ff30f:	不正なレジスタ・バンクです。
Ff310:	不正なディレイ・カウントです。

Wf311: [XXX] イベント条件は 1 つしか有効にできませんが, [YYY] を有効にしますか?
Ff312: [XXX] は既に存在します。
Ff313: イベント番号は既に存在します。
Ff314: イベント名が設定されていません。
Ff315: [XXX] は既に存在します。
Ff316: 有効な [XXX] イベント条件の最大数を越えました。他の [YYY] イベント条件を無効にしてください。
Ff317: イベント条件の最大設定数を越えました。
Ff31e: 不正な開始アドレスです。
Ff31f: 不正な終了アドレスです。
Ff322: 不正な分解能です。
Ff323: 不正なタイムアウト・ブレーク設定です。
Ff324: Section と Qualify は同時に指定できません。
Wf325: プログラムが実行中です。プログラムを一瞬止めて設定しますか?
Wf326: プログラムが実行中です。プログラムを一瞬止めて解除しますか?
Ff327: RAM モニタが有効なため、ソフトウェア・ブレークを設定できません。 ハードウェア・ブレークをお使いください。
Ff350: 途中のフェーズにイベント条件が設定されていません。
Ff351: リンク条件とディスエーブル条件で同じイベント条件が使われています。
Ff352: イベント条件が設定されていません。
Ff357: Phase に AND 条件のイベントが設定されています。
Ff400: 不正なカバレッジ・アドレス範囲です。
Wf401: カバレッジをクリアしますか?
Ff500: 不正なシンボルです。
Ff501: 不正な値です。
Ff502: 不正なパラメータです。
Ff503: 表示できる変数の最大個数を越えました。
Ff504: この変数にブレークは設定できません。 1) 以下の変数にブレークは設定できません。 <ul style="list-style-type: none"> <li>・ローカル変数, スタティック変数</li> <li>・配列変数, 構造体/共用体のメンバー変数</li> <li>・レジスタ/周辺 I/O レジスタ</li> <li>・変数式</li> </ul>
Wf600: プロジェクト・ファイルを保存しますか?

<p>Wf601: ターゲットを接続している場合は、ターゲットの電源を ON にしてください。</p> <p>1) ターゲット・システムを接続していない場合は、そのままボタンをクリックして先に進んでください。</p>
<p>Wf602: MODE 端子をマスクするか、ターゲットを接続するようにしてください。</p>
<p>Ff603: ID コードが間違っています。</p> <p>1) 原因として以下のことが考えられます。[MINICUBE]</p> <ul style="list-style-type: none"> <li>・ ID コードが間違っている場合 → 正しい ID コードを入力してください。</li> <li>・ FLMD0 端子が High であるため、内蔵フラッシュ・メモリが書き込みモードになっている場合 → FLMD0 端子を Low にしてください。</li> <li>・ ID コード (0x79 番地の bit7) が 0 であるため、エミュレータ接続禁止モードになっている場合 → 内蔵フラッシュ・メモリを一旦 Erase(消去)してください。</li> </ul>
<p>Af604: ID コードが間違っています。デバッグを終了します。</p> <p>1) 原因として以下のことが考えられます。[MINICUBE]</p> <ul style="list-style-type: none"> <li>・ ID コードが間違っている場合 → 正しい ID コードを入力してください。</li> <li>・ FLMD0 端子が High であるため、内蔵フラッシュ・メモリが書き込みモードになっている場合 → FLMD0 端子を Low にしてください。</li> <li>・ ID コード (0x79 番地の bit7) が 0 であるため、エミュレータ接続禁止モードになっている場合 → 内蔵フラッシュ・メモリを一旦 Erase(消去)してください。</li> </ul>
<p>Ff605: ターゲットとの接続を確認してください。</p> <p>1) ターゲット・コネクタ (TC) の接続を確認してください。ターゲット・システムを接続しない場合は、<a href="#">コンフィギュレーション・ダイアログ</a>で Target の設定を見直してください。</p>
<p>Ff606: ターゲットとの接続を確認し、ターゲットの電源を入れてください。</p> <p>1) ターゲット・システムの電源を確認してください。ターゲット・システムを接続しない場合は、<a href="#">コンフィギュレーション・ダイアログ</a>で Target の設定を見直してください。</p>
<p>Wf607: 変換アダプタの接続を確認してください。</p> <p>1) 変換アダプタ (EA) の接続を確認してください。</p> <p>ターゲット・システムの接続をしない場合には、変換アダプタの装着を推奨しています。</p>
<p>Ff608: ターゲットを外してください。</p> <p>1) IE の内部電源からターゲット・システムに電源が流れ込む可能性があります。ターゲット・コネクタ (TC) と変換アダプタ (EA) の間を外してください。ターゲット・システムの接続をしない場合には、<a href="#">コンフィギュレーション・ダイアログ</a>での設定を見直してください。</p>
<p>Ff609: ターゲットの電源を切り、ターゲットを外してください。</p>
<p>Af60a: ID コードが間違っています。フラッシュ・メモリを消去しました。デバッグを終了します。</p> <p>1) このメッセージは、フラッシュ・メモリを消去するモードを設定していた場合で、アドレス 0x84 で ID 認証に失敗した時、フラッシュ・メモリ消去後に表示されます。[MINICUBE]</p>

Af60b:	ID コードが無効になっています。フラッシュ・メモリを消去しました。デバッグを終了します。 1) このメッセージは、パワーオフ・エミュレーション・モードでデバッグを使用中に、ターゲット・システム接続ケーブルを抜いてしまったりターゲット・システムの電源を落としたりした後にデバッグを起動した時、フラッシュ・メモリ消去後に表示されます。[MINICUBE]
Af60c:	ブレーク中にターゲット電源が Off になりました。
Wf60d:	ソースパスが 4095 文字を越えたので切り捨てました。
Wf700:	ロード・モジュール・ファイルをダウンロードしますか？
Wf701:	シンボル情報だけロードしますか？
Wf800:	メモリ・バンクの設定がされていません。
Wf801:	メモリ・バンクのアドレスはターゲット・メモリ内でなければなりません。
Ff802:	外部プローブが変更されたので、全てのイベント条件は削除されます。
Ff803:	このイベント条件のアドレスは不正です。
Ff804:	不正な PC 値です。
Ff805:	このアドレスにテンポラリ・ブレークを設定できません。
Ff806:	外部データはデバッグに使われています。
Ff900:	不正な I/O ポートです。
Ff901:	メモリ・マッピング違反です。 1) アドレスの指定が不正です。指定できるアドレスは、 <a href="#">I/O ポート追加ダイアログ</a> で確認してください。
Ff902:	不正なアクセス・サイズです。
Ff903:	不正なアクセス属性です。
Ff904:	同じ名前は登録できません。
Wf905:	[XXX] は既に存在しますが、置き換えますか？
Wf906:	[XXX] への変更内容を登録しますか？
Ffa00:	PC 位置のプログラムの [XXX] 関数が見つかりませんでした。 1) <a href="#">デバッグ・オプション設定ダイアログ</a> で main() label: に指定したシンボルが見つかりません。プログラムのメイン・ルーチンのシンボルを設定してください。デフォルトは、_main になっています。
Ffa01:	PC 位置の行番号が見つかりませんでした。 1) プログラム停止時のプログラム・カウンタ (PC) 値に対応するソース・ファイルが見つかりません。原因としては、以下のようなことが考えられます。 - ソース・ファイルがソース・パスの通っていない場所に存在する。 - ライブラリ、RX など、ソース・ファイルが存在していないところでプログラムを停止した。 - プログラムが暴走し、プログラムで使用していないアドレスへ実行が移り、そこで停止した。

<p>Wfb00: プログラムが実行中です。プログラム実行を止めますか？</p> <p>1) &lt;はい&gt; ボタンを選択した場合、ユーザ・プログラムの実行を停止してから、<a href="#">終了確認ダイアログ</a>を表示します。ただし、<a href="#">デバッガ・オプション設定ダイアログ</a>で終了確認ダイアログを表示しない設定にしている場合には、ID850QB を終了します。</p> <p>&lt;いいえ&gt; ボタンを選択した場合、ユーザ・プログラムの実行は停止せず、<a href="#">終了確認ダイアログ</a>も表示しません。ID850QB の終了も行いません。</p>
<p>Wfb01: ID コードの 0x79 番地の bit7 が 0 になっているため、以降 N-Wire エミュレータが使用禁止となります。このまま終了しますか？</p>
<p>Ffc00: Help ウィンドウが起動できません。ユーザーズ・マニュアルを参照して HTML Help 環境をインストールしてください。</p>
<p>Ffd00: [XXX] の指定に誤りがあります。</p>
<p>Ffe00: RRM の最大サイズを越えています。</p>
<p>Wfe01: RRM アドレスが重複しています。</p>
<p>Wfe0b: フラッシュ・セルフ・モードに移行します。現在のイベントは全てディスエーブルになります。よろしいですか？</p>
<p>Ffe0c: RAM モニタを有効にします。</p> <p>現在のソフトウェア・ブレイクは全て無効になります。</p> <p>よろしいですか？</p>
<p>Fffff: インタラプトされました。</p>

## 付録 E 索引

### A

AZ850 … 24

### C

CA850 … 24  
clock … 125

### D

DMA … 73  
DMM ダイアログ … 216

### G

-G -dual\_debug … 26  
-g オプション … 25  
GHS … 26

### I

I/O Protect … 45  
I/O プロテクト … 45  
I/O ポート追加ダイアログ … 228  
ID Tag … 207  
ID コード … 25  
IECUBE … 21  
Internal RAM … 44  
Internal ROM … 44  
IOR ウィンドウ … 222

### M

Mask … 127  
MINICUBE … 22  
MINICUBE2 … 23

### N

N-Wire CARD … 22  
N-Wire Checker … 28

### O

OCD Checker … 28

### P

PM+ … 24, 36

### R

RRM 設定ダイアログ … 138  
Run-Break イベント … 72

### T

Target … 44  
Target ROM … 44  
Tcl  
    address … 296  
    assemble … 297  
    batch … 298  
    breakpoint … 299  
    cache … 301  
    dbgexit … 302  
    dbgopt … 303  
    download … 304  
    erase … 305  
    extwin … 306  
    finish … 307  
    flop … 308  
    go … 309  
    help … 310  
    hook … 311  
    ie … 312  
    inspect … 313  
    jump … 314  
    map … 315  
    mdi … 317  
    memory … 318  
    module … 319  
    next … 320  
    refresh … 321  
    register … 322  
    reset … 323  
    run … 324  
    step … 325  
    stop … 326  
    tkcon … 327  
    upload … 328  
    version … 329  
    watch … 330  
    where … 331  
    wish … 332  
    xcoverage … 333  
    xtime … 334  
    xtrace … 335  
Tcl コマンド一覧 … 288  
TW850 … 24

### 【あ行】

アクセス・モニタ … 207  
アクセス・モニタ機能 … 66  
アクティブ状態とスタティック状態 … 98  
アスキー表示 … 207  
アップロード … 48  
アップロード・ダイアログ … 163  
アドレス指定ダイアログ … 182  
アンインストール … 27  
イベント管理 … 87  
イベント機能 … 83

イベント条件の設定 … 85  
 イベント詳細表示 … 261  
 イベント・アイコン … 88  
 イベント・ダイアログ … 264  
 イベント・マネージャ … 260  
 イベント・リンク・ダイアログ … 269  
 インサーキット・エミュレータ … 24  
 インストール … 27  
 ウィンドウ一覧 … 106  
 ウィンドウの連結 … 101  
 ウィンドウ・リファレンス … 105  
 ウォッチ機能 … 61  
 ウォッチ登録ダイアログ … 195  
 ウォッチ変更ダイアログ … 198  
 ウォッチ・ウィンドウ … 188  
 エラー … 359  
 演算子 … 353  
 オプション … 30

**【か行】**

拡張ウィンドウ … 337  
 拡張ウィンドウ (サンプル)  
   Grep ウィンドウ … 340  
   Hook ウィンドウ … 342  
   List ウィンドウ … 339  
   Memory Mapped I/O ウィンドウ … 344  
   RRM ウィンドウ … 341  
   Run Break Time ウィンドウ … 348  
   Sym Inspect ウィンドウ … 347  
 拡張オプション設定ダイアログ … 130  
 カバレッジ実行箇所の表示 … 81  
 カバレッジ測定機能 … 79  
 カバレッジ範囲選択ダイアログ … 257  
 環境設定ファイル・セーブ・ダイアログ … 280  
 環境設定ファイル・ロード・ダイアログ … 281  
 キー機能一覧 … 356  
 起動オプション … 29  
 起動前の注意 … 28  
 逆アセンブル・ウィンドウ … 176  
 逆アセンブル・サーチ・ダイアログ … 180  
 クイック・ウォッチ・ダイアログ … 193  
 クオリファイ・トレース … 78  
 コード・カバレッジ … 79  
 コード・カバレッジ・ウィンドウ … 254  
 コールバック・プロシージャ … 292  
 コマンド … 285  
 コマンド・リファレンス … 287  
 混合表示モード  
   ソース・テキスト・ウィンドウ … 51  
   トレース・ウィンドウ … 76  
 コンソール・ウィンドウ … 285  
 コンテキスト・メニュー … 108  
 コンフィギュレーション・ダイアログ … 122

**【さ行】**

式 … 353  
 ジャンプ機能 … 99  
 終了確認ダイアログ … 283  
 条件トレース … 76  
 進数の範囲 … 354  
 シンボル変換ダイアログ … 183

スクリプト・ファイル … 30  
 スタック・トレース表示機能 … 64  
 スタック・トレース・ウィンドウ … 202  
 ステータスバー … 119  
 セキュリティ ID … 25  
 セクション・トレース … 78  
 設定ファイル … 96  
 ソース指定ダイアログ … 174  
 ソース・サーチ・ダイアログ … 172  
 ソース・テキスト・ウィンドウ … 167  
 ソース・ファイル選択ダイアログ … 286  
 ソフトウェア・ブレイク・マネージャ … 258

**【た行】**

タイマ … 133  
 タイマ機能 … 71  
 タイマ測定結果ダイアログ … 234  
 タイマ・ダイアログ … 230  
 ダウンロード … 47  
 ダウンロード/アップロード機能 … 46  
 ダウンロード・ダイアログ … 160  
 注意事項 … 104  
 ツールバー … 117  
 ディレイ・カウント設定ダイアログ … 252  
 ディレイ・トリガ・トレース … 78  
 データ・フラッシュ・オプション設定ダイアログ … 149  
 データ・フラッシュ・メモリ … 124  
 デバッガ・オプション設定ダイアログ … 152  
 デバッグ環境の設定 … 43  
 デバッグ機能一覧 … 42  
 動作環境 … 24  
 ドラッグ & ドロップ機能 … 102  
 トレーサ制御モード … 77  
 トレース機能 … 73  
 トレースタイムタグ … 131  
 トレース表示選択ダイアログ … 245  
 トレース補完 … 74  
 トレース・ウィンドウ … 236  
 トレース・サーチ・ダイアログ … 241  
 トレース・ダイアログ … 249  
 トレース・データ … 132  
 トレース・メモリ … 74

**【な行】**

内部 RAM … 44  
 内部 ROM … 44  
 入力規約 … 349

**【は行】**

バージョン表示ダイアログ … 284  
 表示ファイル … 95  
 表示ファイル・セーブ・ダイアログ … 276  
 表示ファイル・ロード・ダイアログ … 278  
 ファイル名 … 355  
 フェイルセーフ・ブレイク … 126  
 フェイルセーフ・ブレイク設定ダイアログ … 136  
 フック・プロシージャ … 293  
 フラッシュ・オプション設定ダイアログ … 141  
 フラッシュ・メモリ … 67

**ブレーク**

- ブレークの種類 … 54
- ブレーク・ポイントの設定 … 55
- 変数へのブレーク設定 … 56
- ブレーク・ダイアログ … 273
- フレーム指定ダイアログ … 247
- プログラム実行機能 … 59
- プログラム・コード … 169
- プロジェクト・ファイル … 94, 283
- プロジェクト・ファイル・セーブ・ダイアログ … 158
- プロジェクト・ファイル・ロード・ダイアログ … 159
- 分周率と最大測定時間の関係（タイマ） … 133
- 分周率と最大測定時間（トレース） … 131
- ペリファイ・チェック … 134
- ポイント・マーク・エリア … 168, 177
- 補完モード … 74

**【ま行】**

- マッピング設定 … 44
- 右クリック・メニュー … 108
- 無条件トレース … 76
- メイン・ウインドウ … 109
- メッセージ … 359
- メッセージの種類 … 360
- メニューバー … 110
- メモリ操作機能 … 65
- メモリ比較結果ダイアログ … 215
- メモリ比較ダイアログ … 214
- メモリ・ウインドウ … 205
- メモリ・コピー・ダイアログ … 213
- メモリ・サーチ・ダイアログ … 210
- メモリ・フィル・ダイアログ … 212
- 文字セット … 350

**【や行】**

- 有効イベント数 … 86

**【ら行】**

- リスト・ウインドウ … 185
- リセット確認ダイアログ … 282
- レジスタ選択ダイアログ … 221
- レジスタ操作機能 … 68
- レジスタ・ウインドウ … 218
- ローカル変数ウインドウ … 200
- ロード／セーブ機能 … 93
- ロード・モジュール一覧ダイアログ … 165

## 【発行】

NECエレクトロニクス株式会社

〒211-8668 神奈川県川崎市中原区下沼部1753

電話（代表）：044(435)5111

お問い合わせ先

---

## 【ホームページ】

NECエレクトロニクスの情報がインターネットでご覧になれます。

URL(アドレス) <http://www.necel.co.jp/>

---

## 【営業関係，技術関係お問い合わせ先】

半導体ホットライン

（電話：午前 9:00～12:00，午後 1:00～5:00）

電話：044-435-9494

E-mail：info@necel.com

---

## 【資料請求先】

NECエレクトロニクスのホームページよりダウンロードいただくか，NECエレクトロニクスの販売特約店へお申し付けください。

---