

SuperH™ 対応 JTAG デバッガ

Xross Finder

JTAG DEBUGGER

ユーザーズマニュアル

第 6.0 版



ALPHA PROJECT Co., LTD

ご使用になる前に

このたびは XrossFinder をお買い上げいただき誠にありがとうございます。
本製品をお役立て頂くために、このマニュアルを十分お読みいただき、正しくお使い下さい。
今後共、弊社製品をご愛顧賜りますよう宜しくお願いいたします。

梱包内容

本製品は、下記の品より構成されております。梱包内容をご確認のうえ、万が一、不足しているものがあればお買い上げの販売店までご連絡ください。

梱包内容			
●XrossFinder 本体	1 台	●マニュアル、ソフトウェアCD	1 枚
●H-UDI ケーブル	1 本	●EXT-TRG/RESET-OUT ハーネス	各 1 本
●USB ケーブル	1 本	●保証書	1 枚
		●コントロールソフト使用ライセンス	1 枚

■本製品の内容及び仕様は予告なしに変更されることがありますのでご了承ください。

取り扱い上の注意



- 本製品には、民生用の一般電子部品が使用されています。宇宙、航空、医療、原子力、運輸、交通、各種安全装置など人命、事故に関わる特別な品質、信頼性が要求される用途でのご使用はご遠慮ください。
- 極端な高温下や低温下、または振動の激しい環境での使用はご遠慮ください。
- 水中、高湿度、油の多い環境でのご使用はご遠慮ください。
- 腐食性ガス、可燃性ガス等の環境中でのご使用はご遠慮ください。
- 定格を越える電源・電圧を加えないでください。

- ノイズの多い環境での動作は保証しかねますのでご了承ください。
- 発煙や発火、異常な発熱があった場合には、すぐに電源を切ってください。
- 本書に記載される製品および技術のうち、「外国為替および外国貿易法」に定める規制貨物等（技術）に該当するものを輸出または国外に持ち出す場合には同法に基づく輸出許可が必要です。
- 本製品に付属するソフトウェア等の著作権は（株）アルファプロジェクトが保有しております。これらを無断で転用、掲載、譲渡、配布することは禁止します。

保証

- 本製品は万全の注意を払って製作されていますが、万一初期不良品であった場合、お買い上げ頂いた販売店へ保証書を添えてご返却ください。
- 万が一、本製品を使用して事故または損失が発生した場合、弊社では一切その責を負いません。
- 保証内容、免責等につきましては、添付の保証書をご覧ください。
- 本製品は仕様範囲を越える条件において使用された場合については、動作は保証されません。
- 製品を改造した場合、保証は一切適用されません。
- 他社製品との接続互換性および相性問題は保証いたしません。

目 次

1. 概要	6
1-1. 特徴	6
1-2. 構成	6
1-3. 対応 CPU	7
1-3-1. SH-2、SH-2A 対応 CPU	7
1-3-2. SH-3、SH-4、SH-4A 対応 CPU	8
1-4. 対応コンパイラ	9
1-5. コントロールソフト動作環境	9
1-6. 機能概要	10
1-7. 本体各部の説明	11
1-7-1. 本体前面部	11
1-7-2. 本体背面部	11
2. インストール	12
2-1. コントロールソフトのインストール	12
2-2. USBドライバのインストール	15
2-3. ライセンスキーの登録	18
2-4. アンインストール	19
2-4-1. コントロールソフトの削除	19
2-4-2. USBドライバの削除	20
3. デバッグの基本手順と操作	21
3-1. ユーザプログラムを作成する	22
3-1-1. デバッグ情報の形式	22
3-1-2. 最適化オプションについて	22
3-1-3. RAM 上でのプログラム実行	22
3-1-4. ROM 上でのプログラム実行	22
3-2. コマンドファイルを作成する	23
3-2-1. ターゲットボード初期設定	23
3-2-2. コマンド形式	23
3-3. ターゲットと本体を接続する	24
3-4. ターゲットの起動と初期設定	26
3-4-1. はじめて接続する場合の初期設定	26
3-4-2. 2回目以降の起動の場合	27
3-4-3. 接続が失敗した場合には	27

3-5. ユーザプログラム（デバッグ情報）を読み込む	28
3-5-1. RAM 上で実行する場合	28
3-5-2. ROM 上で実行する場合	30
3-6. プログラムを実行、停止する.....	32
3-6-1. 「Execution Toolbar」での操作	32
3-6-2. メニューやキーボードでの操作.....	32
3-7. ウィンドウを表示する	33
3-7-1. 「View Tool Bar」から表示.....	33
3-7-2. メニューやキーボードでの操作.....	33
3-8. ソースファイルを見る	34
3-9. 別のソースファイルを見る	35
3-9-1. 「Functions」ウィンドウの項目をクリックする	35
3-9-2. 「Call Stack」ウィンドウの項目をクリックする	35
3-9-3. 「Breakpoints」ウィンドウのポップアップメニューから選択する	35
3-9-4. 実行が停止した場合.....	36
3-10. ブレークポイントを設定する.....	37
3-10-1. 「Source」ウィンドウで設定	37
3-10-2. 「Functions」ウィンドウで設定	37
3-10-3. 「Breakpoints」ウィンドウで設定.....	37
3-11. CPU レジスタの値を見る、変更する.....	39
3-11-1. CPU レジスタの値を見る.....	39
3-11-2. CPU レジスタの値を変更する	39
3-12. メモリの内容を見る、変更する	40
3-12-1. メモリ内容を表示する.....	40
3-12-2. メモリ内容を変更する.....	41
3-13. 変数の値を見る、変更する.....	42
3-13-1. 「Watch」ウィンドウへの変数名を登録する	42
3-13-2. 変数の値を見る	43
3-13-3. 変数の値を変更する	44
3-14. 作業を保存する、復元する.....	45
3-14-1. 作業を保存する	45
3-14-2. 作業を復元する	45
3-15. メモリのファイル保存、ファイル読み込み	47
3-15-1. メモリの内容をファイルへ保存.....	47

3-15-2. ファイルの内容をメモリへ読み込む.....	48
4. メニューと各ウィンドウの機能と操作	49
4-1. ウィンドウ構成.....	49
4-2. メニュー.....	50
4-2-1. 「File」メニュー [Alt+F].....	50
4-2-2. 「Setup」メニュー [Alt+S].....	50
4-2-3. 「Execution」メニュー [Alt+X].....	53
4-2-4. 「View」メニュー [Alt+V].....	54
4-2-5. 「Window」メニュー [Alt+W].....	55
4-2-6. 「Help」メニュー [Alt+H].....	55
4-3. ツールバー.....	56
4-3-1. 「File toolbar」.....	56
4-3-2. 「Execution toolbar」.....	56
4-3-3. 「View toolbar」.....	56
4-3-4. 「Status bar」.....	56
4-4. ウィンドウ.....	57
4-4-1. 「Source」ウィンドウ.....	57
4-4-2. 「Breakpoints」ウィンドウ.....	61
4-4-3. 「Memory」ウィンドウ.....	64
4-4-4. 「Registers」ウィンドウ.....	66
4-4-5. 「Watch」ウィンドウ.....	68
4-4-6. 「Functions」ウィンドウ.....	72
4-4-7. 「Call Stack」ウィンドウ.....	74
4-4-8. 「Log」ウィンドウ.....	76
4-4-9. エラーメッセージ.....	78
5. 技術資料	80
5-1. 外部トリガ入力回路.....	80
5-2. リセット出力回路.....	80
5-3. H-UDI 接続参考図.....	81
5-4. 機能・仕様.....	81
5-5. 注意事項.....	81
6. サポートのご案内	82

1. 概要

1-1. 特徴

XrossFinder は、Renesas SuperH RISC engine ファミリのマイクロコンピュータを使用したシステムの開発をソフト／ハードウェアの両面からサポートする開発支援装置です。XrossFinder 本体は、H-UDI インターフェースを経由しユーザターゲットシステムに接続され、ホストコンピュータよりユーザプログラムをダウンロードしデバッグを行います。

- 小型・軽量・電源不要 : デバッガ本体は小型軽量。USB バスパワー動作のため電源は不要です。
- コントロールソフト標準添付 : Windows ベースにて C/C++ ソースコードデバッグが可能です。
- ハード/ソフトブレークをサポート : ターゲット CPU の持つ機能を利用しハードウェア・ブレークが可能です。
- 外部トリガ入力ブレーク機能 : ターゲットからのトリガ信号によりブレークが可能です。
- フラッシュメモリ書き込みサポート : FlashWriter EX(別売)により、フラッシュメモリへの書き込みが可能です。

1-2. 構成

XrossFinder は以下の機器およびソフトウェアから構成されています。

- XrossFinder 本体
ホストマシンとターゲットボードを接続するためのユニットです。ホストマシンとは USB で、ターゲットボードとは H-UDI で接続します。
- コントロールソフト
ホストマシンより、ターゲットボード上のユーザプログラム実行や情報表示などの制御を行います。

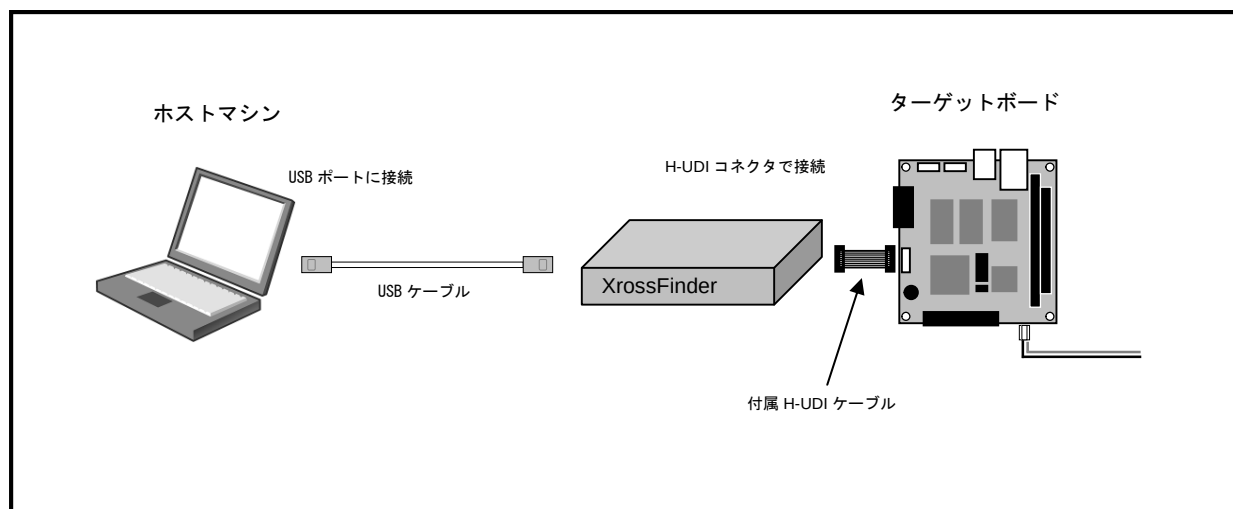


図 1-2 XrossFinder を使用したデバッグ環境の構成例

1-3. 対応 CPU

1-3-1. SH-2、SH-2A 対応 CPU

2010/07 現在

	シリーズ名	CPU	弊社製 CPU ボード	対応
SH-2 SH-2A	SH7040 シリーズ	SH7047	AP-SH2F-5A	○
	SH7137 シリーズ	SH7136	AP-SH2F-11A	○
		SH7137	AP-SH2F-12A	○
	SH7144 シリーズ	SH7144	AP-SH2F-7A, SF-7144F	○
		SH7145	AP-SH2F-6A, MS104-SH2	○
	SH7146 シリーズ※3	SH7146	AP-SH2F-8A	○
		SH7149	AP-SH2F-9A	○
	SH7080 シリーズ※2	SH7083	—	○
		SH7084	—	○
		SH7085	AP-SH2F-10A	○
		SH7086	—	○
	SH7200 シリーズ	SH7201	—	○
		SH7203	—	○
		SH7206	—	○
	SH7210 シリーズ	SH7211	AP-SH2A-0A	○
	SH7216 シリーズ	SH7216	AP-SH2A-4A	○
	SH7280 シリーズ	SH7243	—	○
		SH7285	—	○
		SH7286	AP-SH2A-2A	○
	SH7260 シリーズ	SH7261	—	○
		SH7262	—	○
		SH7263	AP-SH2A-1A	○
		SH7264	—	○
	SH7606 シリーズ	SH7606	—	○
	SH7670 シリーズ	SH7670	AP-SH2A-3A	○
		SH7671	—	○
	SH-Ether シリーズ	SH7619	μ ST-SH2	○
	SH2A-Dual シリーズ	SH7205	AP-SH2AD-0A	○※4
		SH7265	—	○※4
SH-TINY	SH/TINY シリーズ	SH7124F	—	○
		SH7125F ※1	STK-7125	○

※1：SH7125 デバッグ MCU ボードにも対応

※2：SH7080 シリーズは、CPU の種別が 2 種類ありますが、CPU の製品型名の先頭 3 文字（“R5E” もしくは “R5F”）が一致するものを選択してください。

※3：SH7146 シリーズは、CPU の種別が R（R マスク）と A（A マスク）の 2 種類ありますが、CPU の製品型名の 7146、7149 の後ろにある文字と一致するものを選択してください。

※4：SH2A-Dual へ対応するための追加機能につきましては、アプリケーションノート「AN803 XrossFinder ユーザーズマニュアル補足」を参照してください。

1-3-2. SH-3、SH-4、SH-4A 対応 CPU

2010/07 現在

	シリーズ名	CPU	弊社製 CPU ボード	対応
SH-3	SH7700 シリーズ	SH7705	—	○
		SH7706	—	○
		SH7709A	AP-SH3-1A	○
		SH7709S	AP-SH3-2A	○
		SH7710	—	○
		SH7712	—	○
		SH7727	AP-SH3D-2A	○
		SH7729	AP-SH3D-0A	○
		SH7729R	AP-SH3D-1A	○
SH-4 SH-4A	SH7730 シリーズ	SH7730	AP-SH4A-1A	○
	SH7750 シリーズ	SH7750	AP-SH4-0A	○
		SH7750R	AP-SH4-1A, MS104-SH4	○
		SH7751R	—	○
		SH7760	—	○
	SH7780 シリーズ	SH7763	AP-SH4A-2A	○
		SH7764	MS104-SH4AG	○
		SH7780	AP-SH4A-0A	○
		SH7785	AP-SH4A-3A	○
	SH7450 シリーズ	SH7450	—	○
		SH7451	—	○

1-4. 対応コンパイラ

下記コンパイラの ELF/DWARF 形式デバッグ情報フォーマットに対応しています。

- 「SuperH RISC engine C/C++コンパイラ」(バージョン6以降)
- GNU-C/C++コンパイラ (KPIT GNUSH 推奨: 導入方法は弊社 HP 技術コラムを参照して下さい)

1-5. コントロールソフト動作環境

コントロールソフトの動作に必要なホストマシンの推奨環境は以下の通りです。

パーソナルコンピュータ	PC/AT互換機
OS	Windows 2000/XP/Vista
メモリ	256Mバイト以上を推奨
ハードディスク	20Mバイト以上の空き領域
CDドライブ	CD-ROM読み込み可能なドライブ (インストール時のみ)
表示	800×600 以上
USBポート	USB 1.1 または 2.0 1ポート

1-6. 機能概要

以下に XrossFinder のデバッグ機能一覧を示します。

機能名称	機能内容	操作 WINDOW
プログラムダウンロード	ユーザプログラムをターゲットの RAM にダウンロードします。	メイン
ソースリスト表示	プログラムを C / アセンブリ / C + アセンブリ混在で表示します。	ソース
関数リストの表示	関数のリストをツリー表示します。	ファンクション
作業環境の読み込み／保存	ブレークポイント等の現在の作業環境を保存、読み込みします。	メイン
ユーザプログラムの実行／停止	ユーザプログラムの実行／停止を行います。	メイン
ステップ実行	ユーザプログラムのステートメントをステップ実行させます。	メイン
ハードウェアリセット	ターゲットボードにリセット信号を出力し、プログラムカウンタをユーザプログラムの先頭にします。	メイン
ブレークポイント設定	ブレークポイント、ブレーク条件を指定して任意の点でプログラムを停止させます。ソース上とアドレスで指定可能です。	ソース ブレーク
外部トリガブレーク設定	ターゲット側からの信号による外部トリガブレークが指定可能です。	メイン
レジスタ表示	レジスタの内容を表示します。	レジスタ
レジスタ書込み	レジスタの内容を任意の値に書き換えます。	レジスタ
メモリ編集	ターゲット上のメモリ内容を表示、編集します。サイズ、表示形式、シンボル指定が可能です。	メモリ
メモリ設定	ターゲット上のメモリ内容を任意の値に書き換えます。アドレス指定とシンボル指定が可能です。	メイン
関数呼出表示	関数の呼出階層を表示します。	コールフレーム
ウォッチ	指定された特定のメモリ領域をウォッチします。シンボル指定が可能です。	ウォッチ ソース
ログ表示	操作及びターゲット間の処理のログを表示します。	ログ

1-7. 本体各部の説明

1-7-1. 本体前面部



・ EXT-TRG 端子

外部トリガ入力。
EXT-TRG ハーネスを接続します。

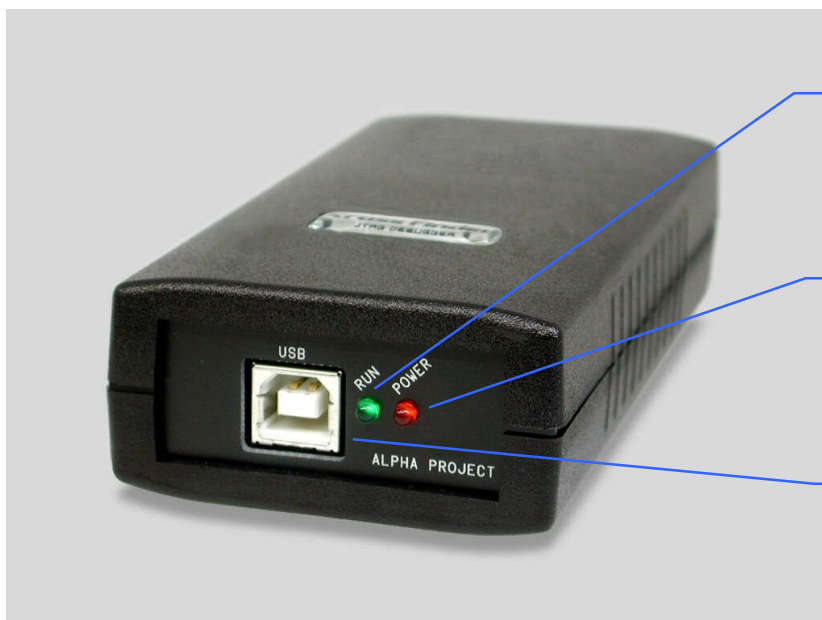
・ RESET-OUT 端子

RESET 信号出力。
RESET ハーネスを接続します。

・ H-UDI コネクタ

H-UDI ケーブルを接続します。

1-7-2. 本体背面部



・ RUN インジケータ

実行 LED。
ユーザプログラム実行中に点灯します。

・ POWER インジケータ

電源 LED。
USB コネクタを接続し電源が供給されると点灯します。

・ USB コネクタ

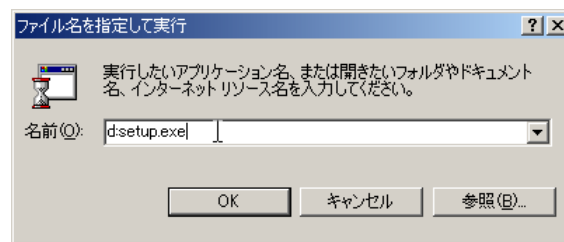
USB ケーブルを接続します。

2. インストール

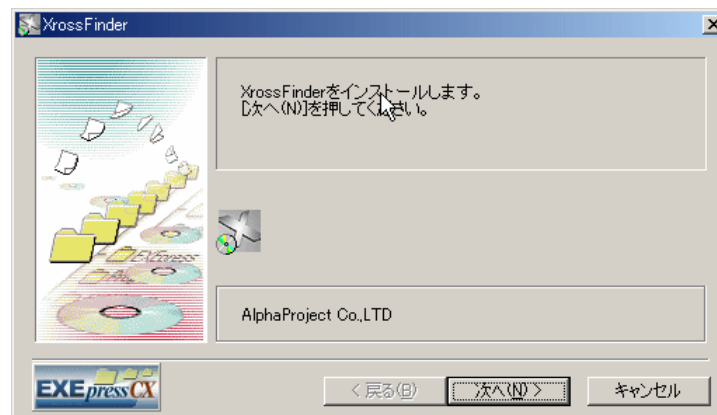
インストールを開始するにあたり、ホストマシン、ターゲット、インストール CD、デバッグ本体、付属 H-UDI ケーブル、USB ケーブルをご準備下さい。(以降 Windows2000 環境での例となります)

2-1. コントロールソフトのインストール

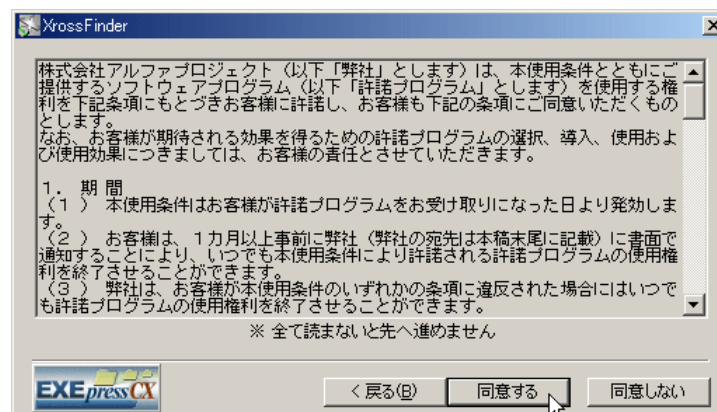
- 1) ホストマシンを起動し OS を立ち上げます。[スタート]メニューから、[ファイル名を指定して実行]を選び、インストールディスク内の SETUP.EXE を指定して実行します。



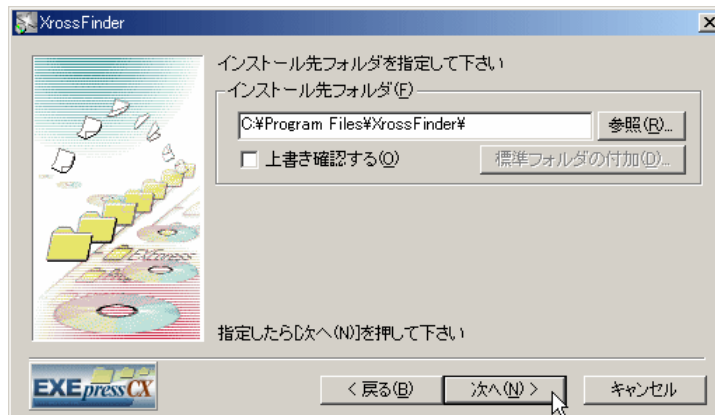
- 2) インストール画面が表示されます。「次へ」のボタンをクリックして下さい。



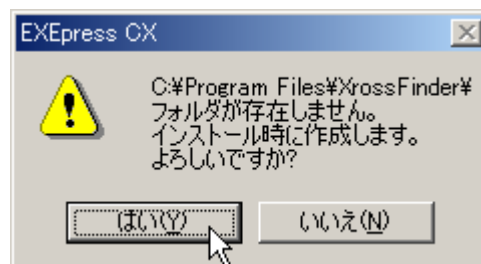
- 3) ソフトウェア使用許諾をお読み頂き、同意いただければ「同意する」ボタンをクリックして下さい。



- 4) インストール先のフォルダを指定します。表示されている場所であれば、「次へ」のボタンをクリックして下さい。変更する場合は参照ボタンで任意の場所を指定してください。



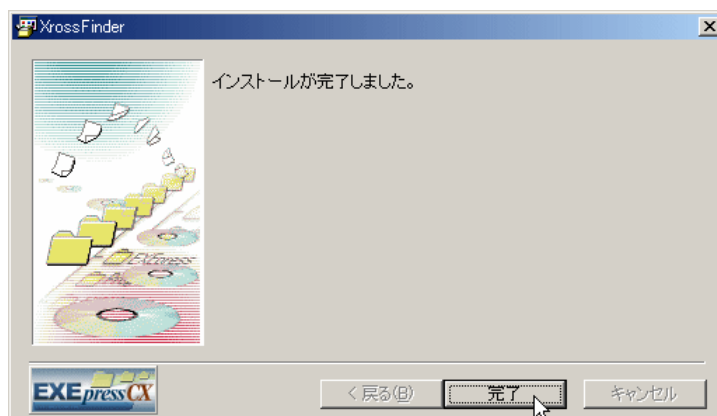
- 5) インストールのフォルダを新規作成します。よろしければ、「はい」のボタンをクリックして下さい。



- 6) インストール情報の再確認し、よろしければ「次へ」のボタンをクリックして下さい。インストールが開始されます。修正する場合は「戻る」ボタンをクリックして下さい。

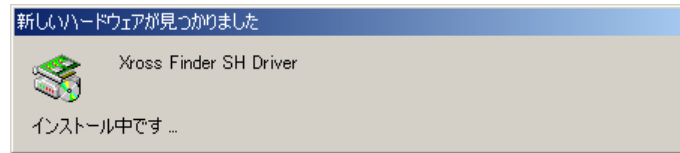


- 7) インストールが終了したら、完了ボタンをクリックして下さい。以上でコントロールソフトのインストールは終了です。

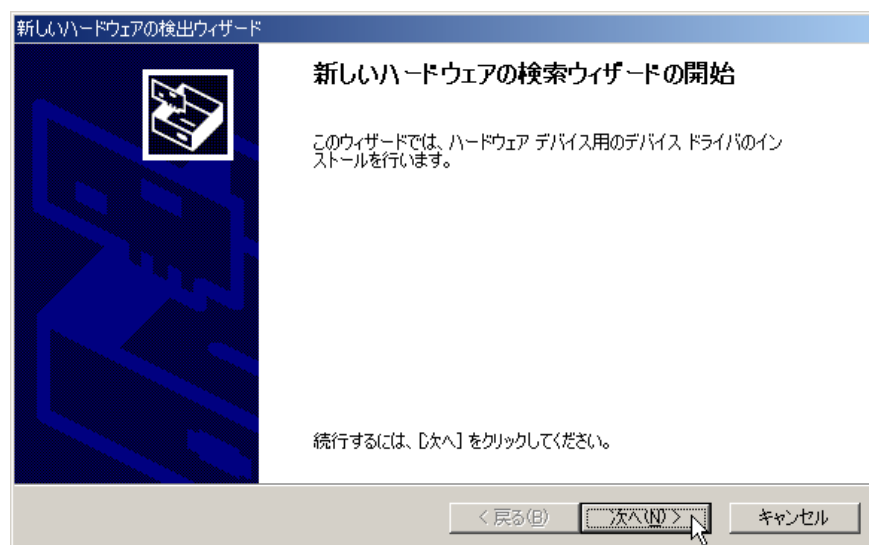


2-2. USBドライバのインストール

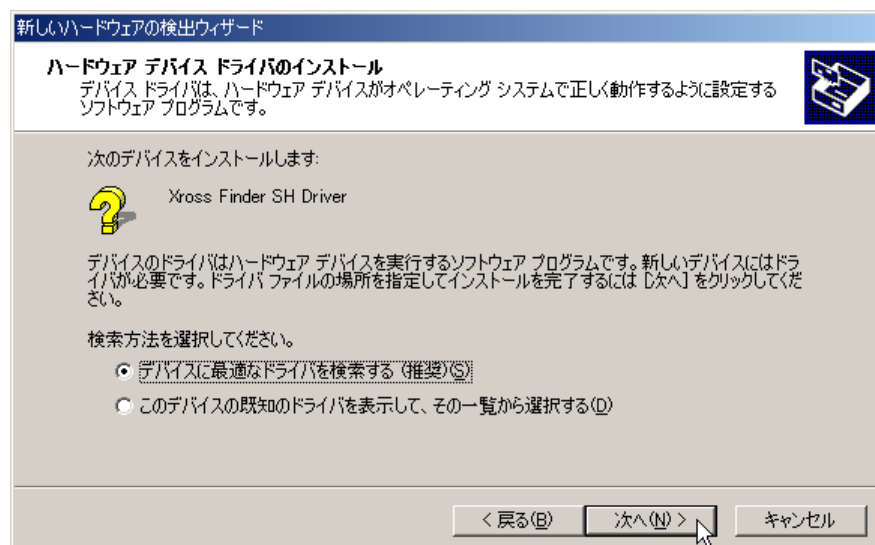
- 1) XrossFinder 本体とホストマシンをUSBケーブルで接続します。「新しいハードウェアが見つかりました」と表示されます。



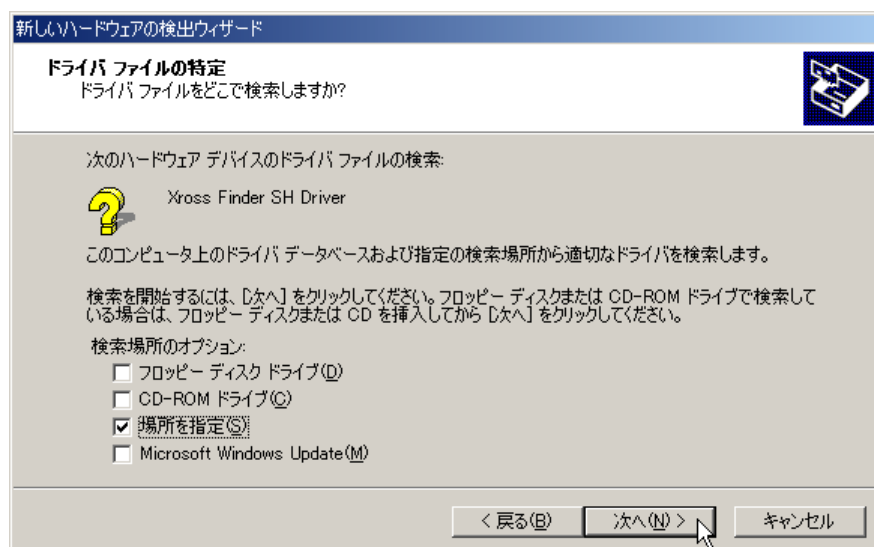
- 2) ハードウェア検出のウィザードが起動しますので、「次へ(N)」ボタンを押します。



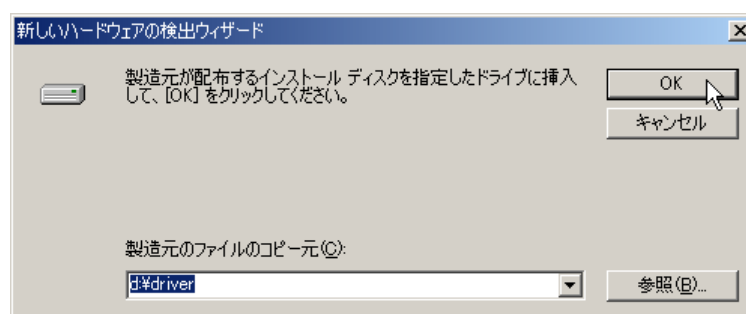
- 3) 「デバイスに最適なドライバを検索する」を選択し「次へ(N)」ボタンを押します。



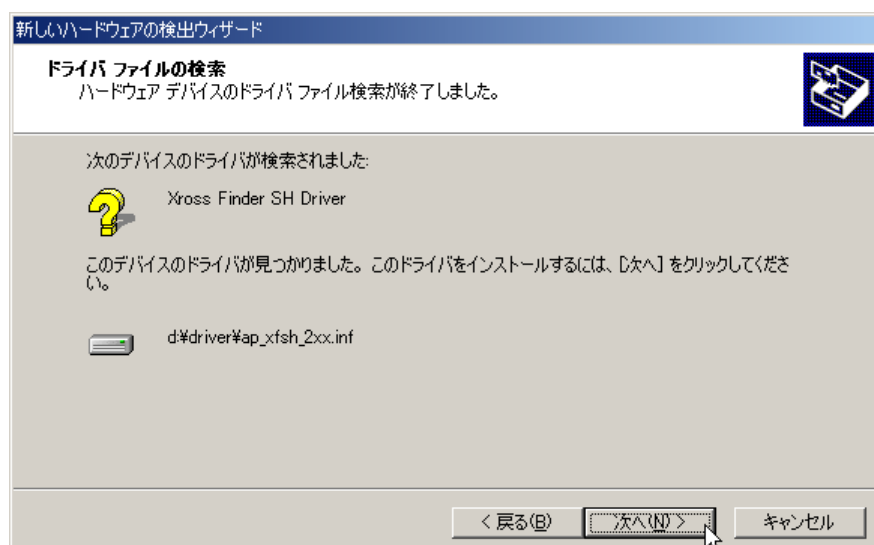
- 4) ドライバファイルの検索は、「場所を指定」を選択し、「次へ(N)」ボタンを押します。



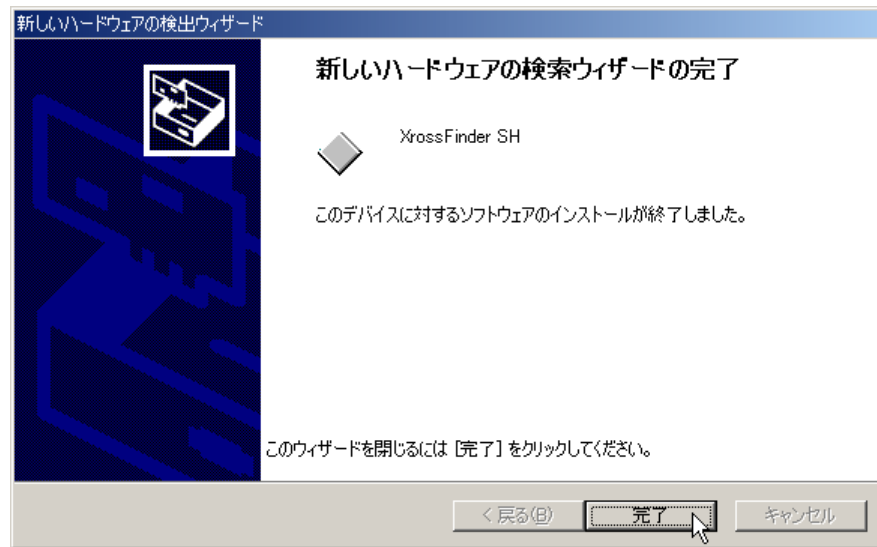
- 5) インストール CD のドライバのパスを指定し、「OK」ボタンを押します。



- 6) ドライバが検索されたら、「次へ(N)」ボタンを押します。



- 7) インストールが終了したら、「完了」ボタンを押します。以上でドライバのインストールは完了です。

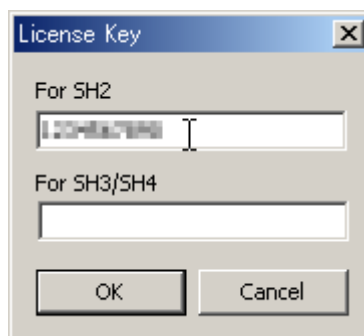


2-3. ライセンスキーの登録

- 1) 「スタート」メニューから、「プログラム(P)」→「AlphaProject」を選び、「XrossFinder」フォルダより XrossFinder(License)を実行します。



- 2) ライセンスキー入力ウィンドウが表示されますので、「コントロールソフト使用ライセンス」に記載されているライセンスキーを入力して下さい。(ライセンスは SH-2 と SH-3/SH-4 の 2 種類あります)

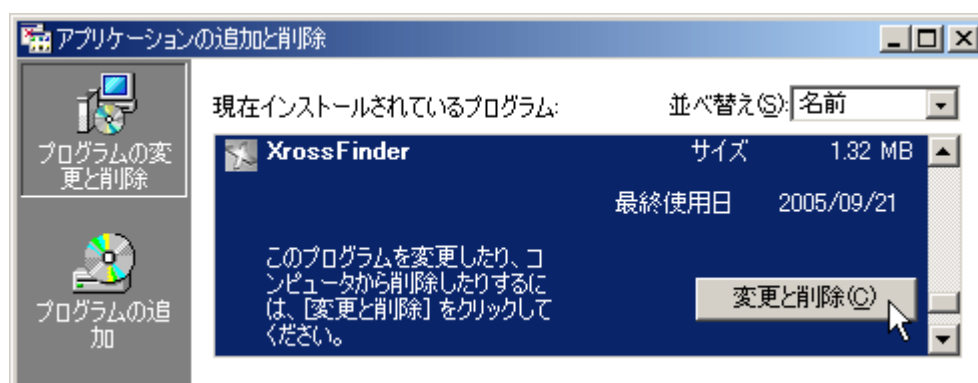


- 3) ライセンスキーの入力が終わりましたら、「OK」ボタンを押します。以上でライセンスの登録は終了です。新たにライセンスを登録する場合以外は、再度入力する必要はありません。

2-4. アンインストール

2-4-1. コントロールソフトの削除

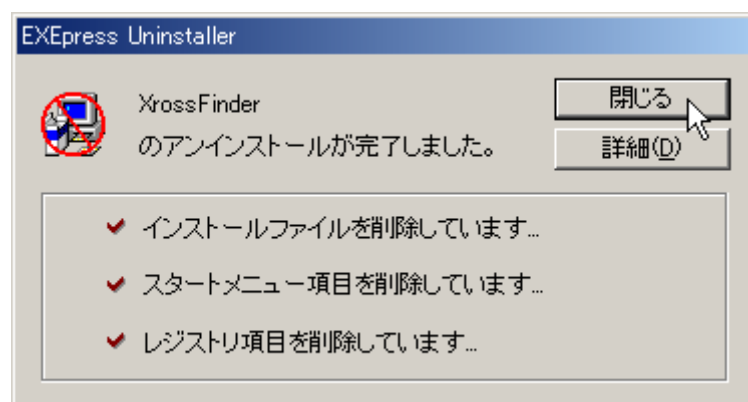
- 1) 「スタート」メニューから、「設定(S)」→「コントロールパネル(C)」を選び、「アプリケーションの追加と削除」を実行します。
- 2) 「現在インストールされているプログラム」から XrossFinder の項目を選択し、「変更と削除(C)」のボタンをクリックします。



- 3) アンインストールの確認が表示されますので「はい(Y)」を押すとアンインストールが開始されます。

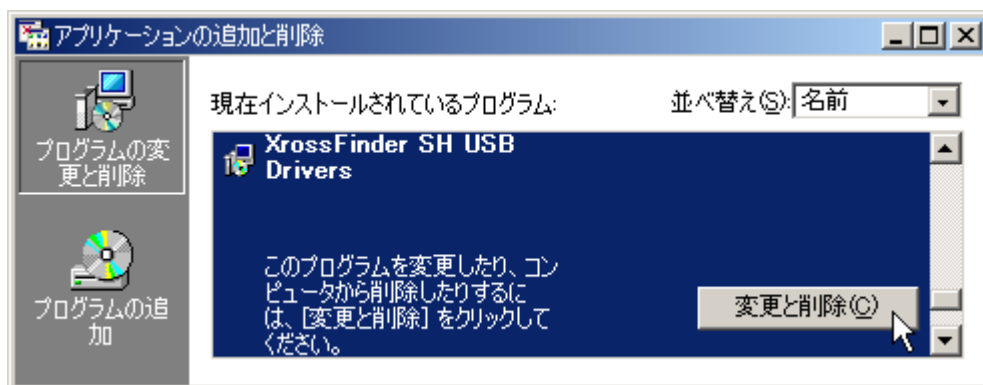


- 4) 「アンインストールが完了しました」と表示されたら、「閉じる」ボタンを押します。以上でアンインストールは終了です。

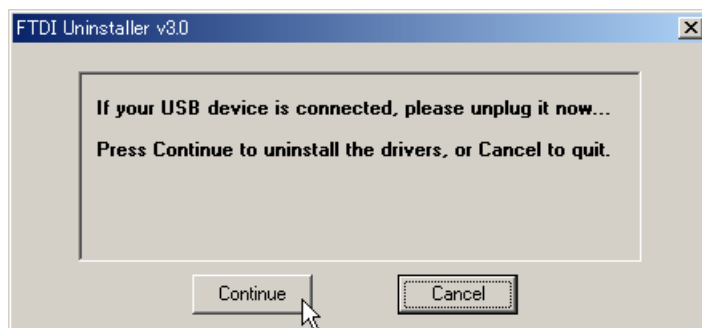


2-4-2. USBドライバの削除

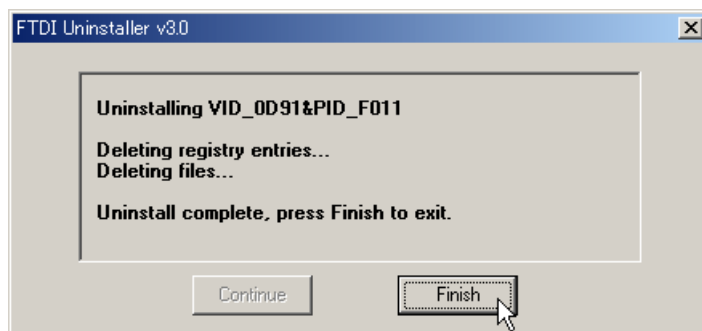
- 1) 「スタート」メニューから、「設定(S)」→「コントロールパネル(C)」を選び、「アプリケーションの追加と削除」を実行します。
- 2) 「現在インストールされているプログラム」から XrossFinder の項目を選択し、「変更と削除(C)」のボタンをクリックします。



- 3) XrossFinder 本体の USB ケーブルを外します。「Continue」を押すとアンインストールが開始されます。



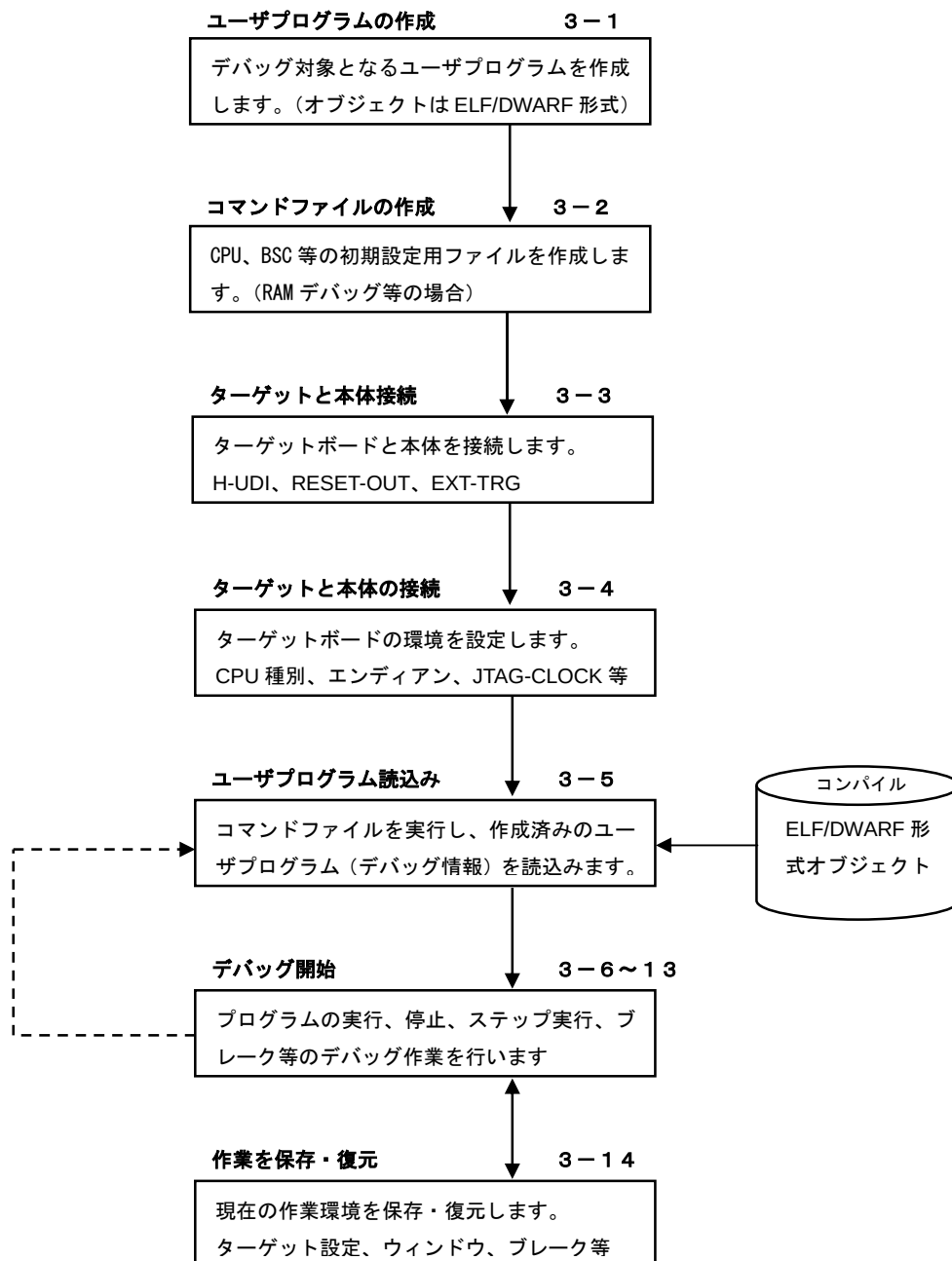
- 4) 「Uninstall Complete」と表示されたら、「Finish」ボタンを押します。以上でアンインストールは終了です。



3. デバッグの基本手順と操作

この章では、デバッグでよく使われる機能を中心に、デバッグ操作の概要を解説致します。
コントロールソフト各部の詳しい操作については、次章にまとめてありますのでご参照ください。

デバッグ開始までの手順は以下のようになります。



3-1. ユーザプログラムを作成する

3-1-1. デバッグ情報の形式

- ユーザプログラムのコンパイル／リンク オプションに **ELF/DWARF 形式** を指定してください。
 - XrossFinder は ELF/DWARF 形式のデバッグ情報をサポートしています。
 - コンパイル後のユーザプログラム オブジェクトの拡張子を下記に示します。
Renesas 社コンパイラ : *.abs 、 GNU-GCC コンパイラ : *.out、*.x、*.elf

3-1-2. 最適化オプションについて

- 最適化オプションを指定せずにコンパイルすることをお勧めします。
 - 最適化オプションを指定すると、ソースコードと出力コード間で、実行の順番や変数の割付けが対応しなくなる傾向があります。
 - このためステップ実行で予期しない動作や、変数の値が表示されない等の現象が生じ、ソースレベルデバッグが困難になる恐れがあります。

3-1-3. RAM 上でのプログラム実行

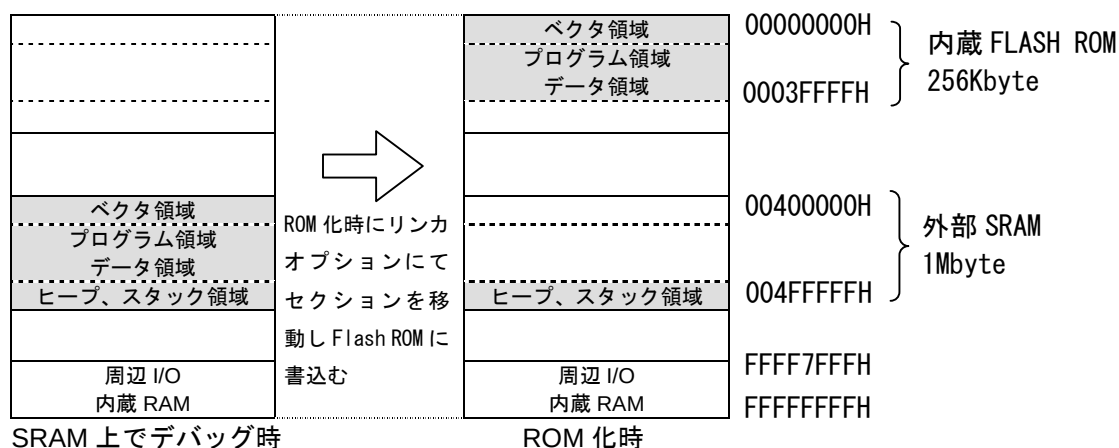
- 最終的に ROM 化されるプログラムも、デバッグ時には RAM にダウンロードしてデバッグする方法をお勧め致します。
 - RAM 上のデバッグ時にはハードウェア、ソフトウェア・ブレークが使用できます。
 - プログラム転送時間の短縮、ROM 書込み回数が減らせます。
 - CPU、メモリ、BUS コントローラ等の初期化の為にコマンドファイルを使用します。(次章参照)
 - リンカオプション等でプログラム配置が変更できるよう開発環境を構築しましょう。

3-1-4. ROM 上でのプログラム実行

- ROM 化された製品レベルでのプログラムもデバッグが可能です。
 - ROM 上のデバッグ時にはハードウェア・ブレークが使用できます。

メモリマップ例)

弊社製 CPU ボード AP-SH2F-7A (内蔵 ROM 有効モード) の場合



3-2. コマンドファイルを作成する

コマンドファイルはターゲットの CPU レジスタ、BUS コントローラ、メモリ等の初期設定を行うためのコマンドを記述するテキストファイルです。RAM 上でデバッグする場合、あらかじめコマンドファイルを実行しユーザプログラムをターゲット RAM にダウンロードします。

3-2-1. ターゲットボード初期設定

- コマンドファイルを使いターゲットボードの初期設定をします。
 - RAM上でデバッグする場合、あらかじめバス、メモリ等の初期設定が必要です。
 - プログラムカウンタやスタックポインタの初期値も設定することができます。

3-2-2. コマンド形式

- コマンドファイルには、メモリや CPU レジスタへ値を書込むためのコマンド列が記述されます。
- テキストファイルなので一般のエディタで作成できます。
 - コマンドは記述順に実行されます。
 - コマンドは下記形式で記述します。

[書式] コマンド [ロケーション] 値

↓

値 (16 進表記)

↓

アドレス (16 進表記) またはレジスタ名 (CPU のレジスタ名)

コマンド	意 味	動 作
WL	Write Long	指定アドレスにロング(4バイト)値を書込む
WW	Write Word	指定アドレスにワード(2バイト)値を書込む
WB	Write Byte	指定アドレスにバイト(1バイト)値を書込む
WR	Write Registers	指定レジスタに値を書込む
WT	Wait Time	指定時間ウエイトする (単位:mS、最大値:5000)

下記に記述例を記載します。ファイル名は任意、拡張子は .xfc にて保存して下さい。

例) sample.xfc

```
// コメント
WL FF800000 0000000C // FF800000H 番地に 0000000CH を書込む
// ↓行をあけても良い

//                ↓桁数はデータサイズに合わせなくても良い
WW FF00001C 808      // FF00001CH 番地に 0808H を書込む

//                ↓桁数が多い場合最上位桁は無視される
WR R0  AFFFFFFF      // CPU のレジスタ R0 に FFFFFFFFH をセットする

WT 100                // 100mS ウェイトする

// 小文字でも良い
wr pc 0c000000        // CPU のレジスタ PC に 0C000000H をセットする
```

弊社製 CPU ボード用のコマンドファイルのサンプルは、弊社 HP よりダウンロード可能ですのでご利用下さい。

3-3. ターゲットと本体を接続する

1) ホストマシンと本体の接続



- ・ 本体とホストマシンを付属の USB ケーブルで接続します。
- ・ POWER インジケータが点灯します。（RUN インジケータはターゲット CPU が実行されている時に点灯します。）

2) H-UDI ケーブルを接続する



3) RESET-OUT ハーネスを接続する



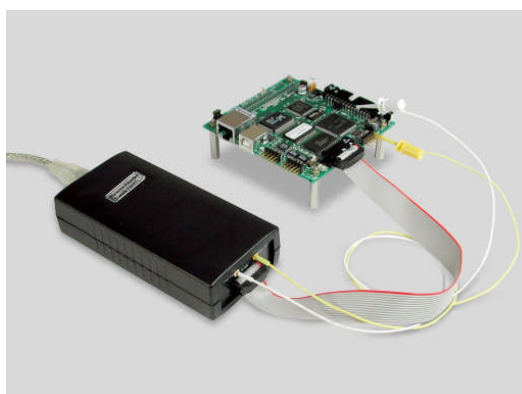
- ・ 付属の RESET-OUT ハーネスを本体に接続します。
- ・ ターゲットボードの予期せぬ動作を防ぐため、接続することをお勧めします。
- ・ CPU 以外の周辺回路もリセットしたい場合にも有効です。

4) EXT-TRG（外部トリガ入力）ハーネスを接続する



- ・ 付属の EXT-TRG ハーネスを本体に接続します。
- ・ ターゲットボードからの入力信号のエッジをトリガとして、デバッグ中のユーザプログラムの実行を強制停止させたい場合に使用します。

5) 本体とターゲットボードを接続する



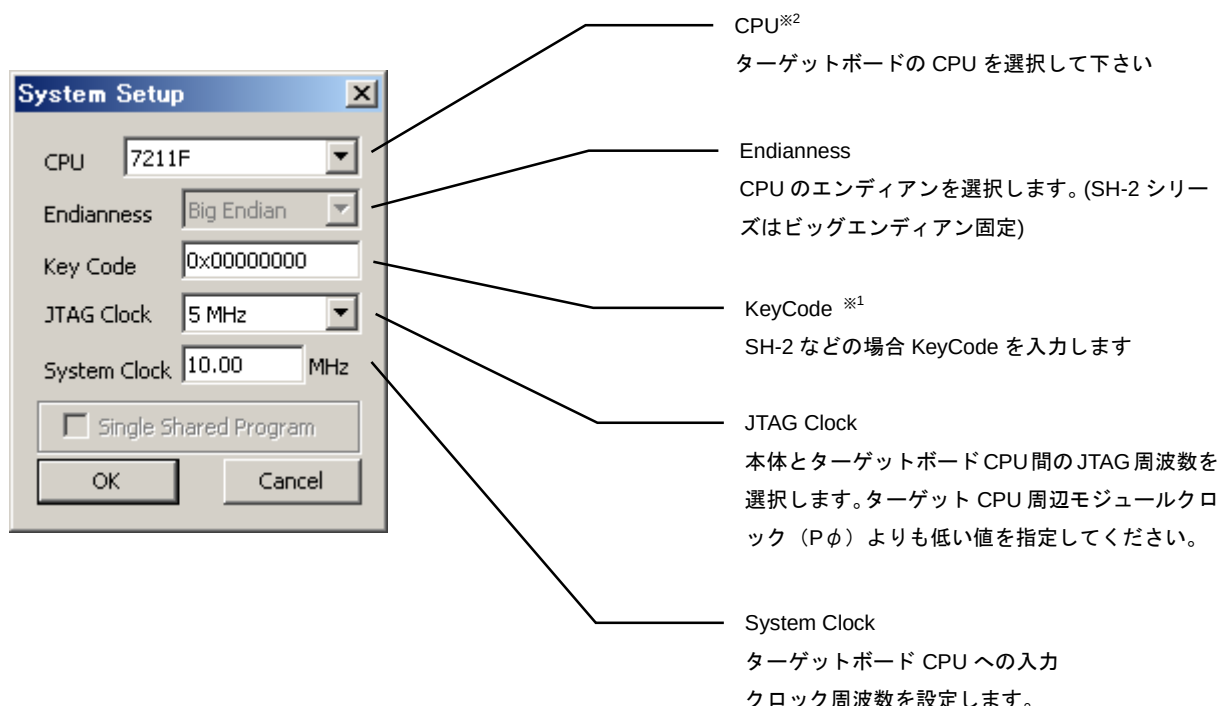
- ① ターゲットボードの電源を切った状態で行ってください。
- ② ターゲットボードと本体を付属の H-UDI ケーブルで接続します。
- ③ EXT-TRG（外部トリガ入力）は、外部トリガ機能を使用したいターゲットボード上の信号に接続します。
(後述の「External Trigger setup」ならびに「技術資料」を参照して下さい)
- ④ RESET-OUT をターゲットボードのリセット入力に接続してください。
(RESET-OUT 端子はオープンコレクタ出力です。ターゲット側でプルアップが必要です。後述の技術資料を参照して下さい)

3-4. ターゲットの起動と初期設定

ターゲットボードを初めて接続しコントロールソフトを起動するには、事前にターゲットボード CPU の初期設定が必要になります。

3-4-1. はじめて接続する場合の初期設定

- 1) ターゲットボードの電源を投入します。
- 2) 「スタート」メニューから、「プログラム(P)」→「AlphaProject」を選び、「XrossFinder」フォルダより XrossFinder (Setup)を実行します。
- 3) System Setup 設定ダイアログが表示されます。下記の各項目を設定してください。



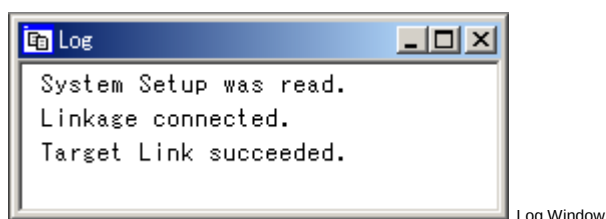
NOTE

※1 : Key Code(SH-2 シリーズの SH7144F/SH7145F 等の場合)

セキュリティ用の Key Code を入力します。この値は、ターゲットボード CPU と H-UDI 接続を行う際に必要となります。ターゲットボードに書込まれている値と一致しない場合、内蔵 FlashROM は消去されます。詳しくはアプリケーションノートをご参照ください。

※2 : CPU の選択肢に括弧のついた同じ型番のものががあります。これらにつきましては、「1-3. 対応 CPU」をご参照ください。

- 4) 以上を設定しダイアログの OK ボタンを押すとターゲットボード CPU との接続を開始します。
- 5) 接続が成功すると、「Log」ウィンドウには右図のように表示されます。
接続が失敗しエラーメッセージが表示された場合、次節を参照してください。



3-4-2. 2回目以降の起動の場合

一度設定を行えば、同じターゲットボードであれば次回以降の設定は必要ありません。
次回からは下記の手順にてコントロールソフトを起動します。

- 1) ホストマシンと本体、本体とターゲットボードの接続を確認します。
- 2) ターゲットボードの電源を投入します。
- 3) タスクバーのスタートボタンでメニューを表示し、「プログラム」を選択します。
- 4) サブメニューから「XrossFinder」を選択し、表示されたサブメニューから「XrossFinder」を選択します。

3-4-3. 接続が失敗した場合には

- 「USB Not Found」「USB Open Error」「USB Transmit Error」のメッセージが表示された場合。

- ✓ USB ケーブルは正しく接続されていますか？

接続を再確認して下さい。

1. 設定ダイアログが表示されている場合「Cancel」ボタンを押し、設定を中止する。
2. XrossFinder コントロールソフトの「File」メニューから「Exit」を選び、終了する。
3. ターゲットボードの電源を切る。
4. USB ケーブルをいったん引き抜き、再び接続したあと、コントロールソフトの起動を再び行う。

- ✓ USB ドライバが正しくインストールされていますか？

以下の手順で確認して下さい。（Windows2000 の場合）

1. タスクバー→スタートボタン→設定→コントロールパネル→システムで「システムのプロパティ」を表示。
2. 「ハードウェア」タブをクリックし、「デバイスマネージャ」ボタンをクリックする。
3. ツリー表示された項目の中の「USB コントローラ」を展開し「XrossFinder SH」が存在しているか確認する。
ドライバが無い場合は、再度インストールをして下さい。

- 「Debug target Error」「Target Time Out」のメッセージが表示された場合。

- ✓ ターゲットボードの電源は入っていますか？

- ✓ ターゲットボードのディップスイッチなど、各種設定は正しいですか？

- ✓ H-UDI ケーブルが接続されていますか？

上記いずれかに問題があった場合、適切な処理後コントロールソフトの「Setup」メニューから「System」を選択し、設定ダイアログの内容を確認して OK ボタンを押します。

- ✓ 「JTAG Clock」設定値の変更

「JTAG Clock」の値を変更することで接続できることがあります。コントロールソフトの「Setup」メニューから「System」を選択し、設定ダイアログの「JTAG Clock」を一段階小さな値に設定してください。

- ✓ RESET-OUT を接続してみる

ターゲットボードに RESET-OUT ケーブルをつなぐことで接続できることがあります。

1. XrossFinder コントロールソフトの「File」メニューから「Exit」を選び終了する。
2. ターゲットボードの電源を切る。
3. RESET-OUT ケーブルを接続し、コントロールソフトの起動を再び行う。

- 「XXXX Version not matched.」のメッセージが表示された場合。

- ✓ 関連ファイルまたはプログラムのバージョンが一致なくなっています。

弊社ホームページで最新版を確認し更新してください。

3-5. ユーザプログラム（デバッグ情報）を読み込む

ユーザプログラムの実行環境（RAM または ROM）により、読み込み手順は異なります。
最初にコントロールソフトを起動します。

（各ウィンドウのサイズ、位置は適宜変更して下さい。）

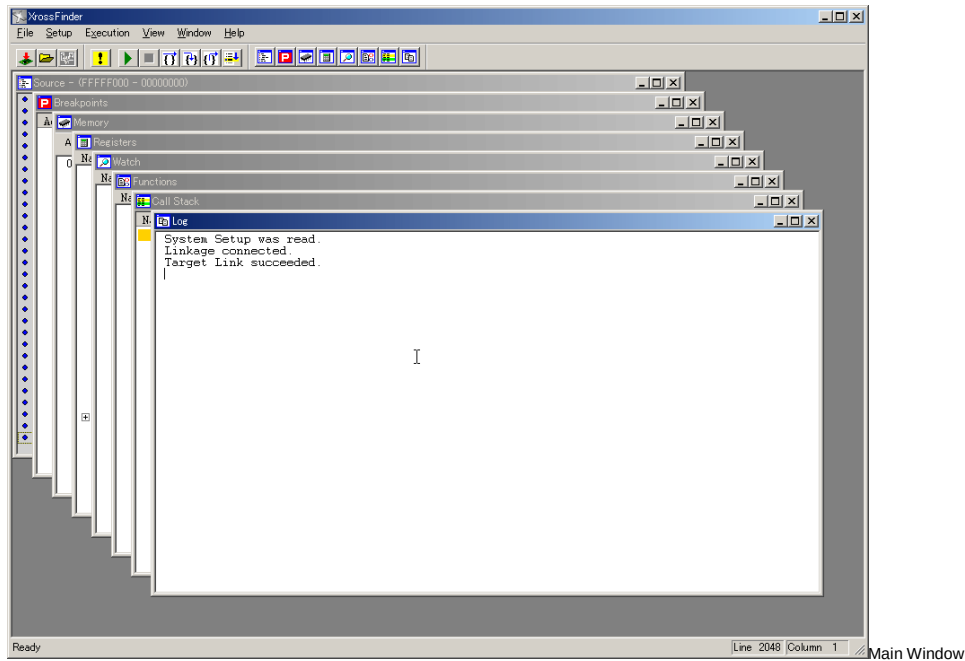
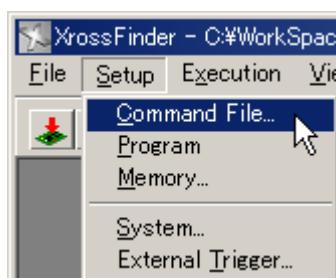


図 3-5 コントロールソフト起動画面

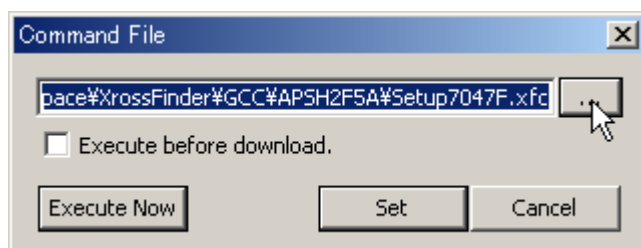
3-5-1. RAM 上で実行する場合

- 1) RAM 上へプログラムを転送する前にコマンドファイルを実行します。



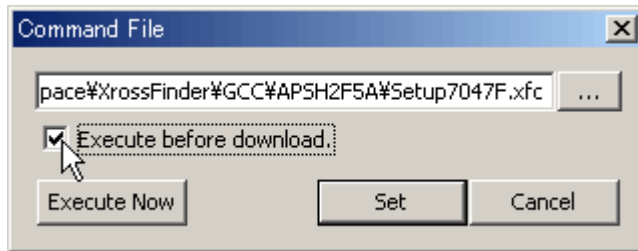
・「Setup」メニューの「Command...」を選択します。

- 2) コマンドファイルを選択します。



・テキストボックスでコマンドファイルのパスを入力するか、「...」ボタンのクリックで表示されるファイルダイアログでコマンドファイルを選択します。

- 3) “Execute before download”チェックボタンを有効にします。



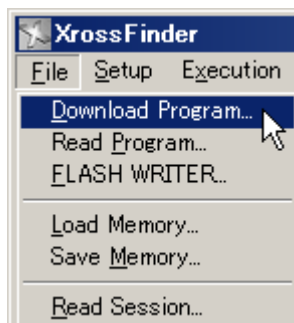
・チェックボタンを有効にすると RAM 上へプログラムを転送する前にコマンドファイルが実行されます。

- 4) 「Set」ボタンで設定を終了します。ターゲットの変更が無ければ、次回以降設定の必要はありません。

その場で実行を確認したい場合は「Execute Now」ボタンをクリックします。

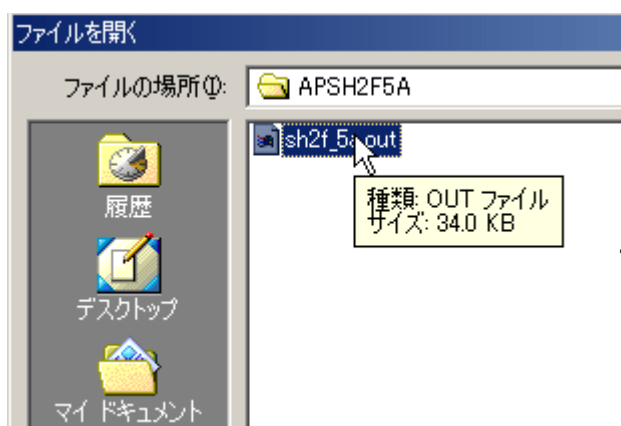
コマンドファイルの記述に問題があった場合は、「Command File Error at line XX.」と表示されますのでお使いのエディタ等で修正してください。

- 5) RAM 上へ転送するユーザプログラムファイルを選択します。



・「File」メニューの「Download Program...」を選択するか、「File Tool

Bar」の「Download To Target」ボタン  をクリックします。



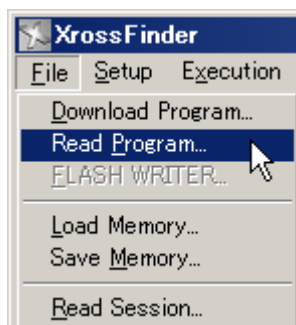
・ファイルダイアログでユーザプログラムファイルを選択します。

- 6) ユーザプログラムファイル読み込まれ、続いて自動的にコマンドファイルの実行とユーザプログラムが転送されます。

3-5-2. ROM 上で実行する場合

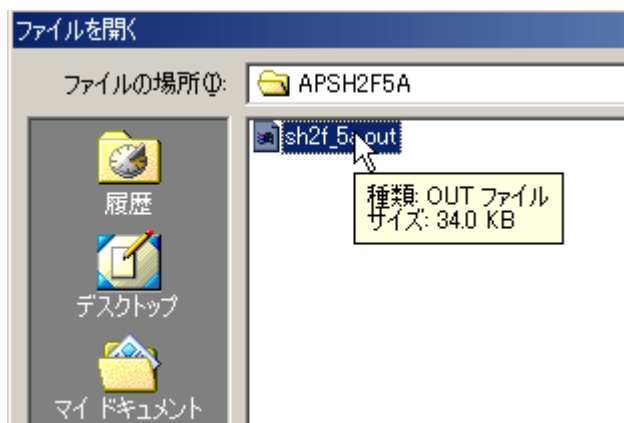
ROM 上でデバッグを行う場合は、あらかじめ読み込ませるユーザプログラム（デバッグ情報）と対応したプログラムが ROM に書込まれている必要があります。

- 1) ユーザプログラムファイルを選択し読み込みます。



・「File」メニューの「Read Program...」を選択するか、「File Tool Bar」

の「Read Debug Informations」ボタン  をクリックします。



・ファイルダイアログでユーザプログラムファイルを選択します。

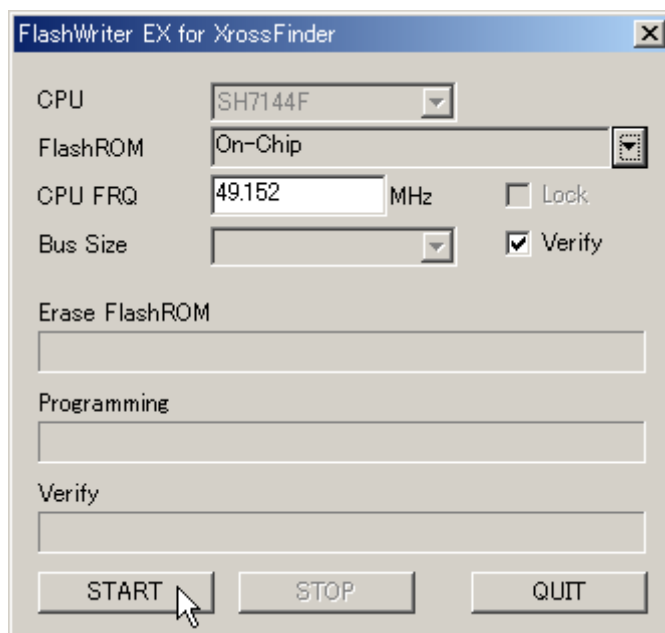
- 2) 読み込んだユーザプログラムと対応したプログラムが既に ROM に書込まれている場合はこれで終了です。

新しいプログラムコードで Flash ROM を書き込みしたい場合には、別売の FlashWriter EX でプログラムコードをターゲット ROM に書込みます。FlashWriter EX は独立したアプリケーションですが、XrossFinder から起動でき、シームレスに作業ができます。

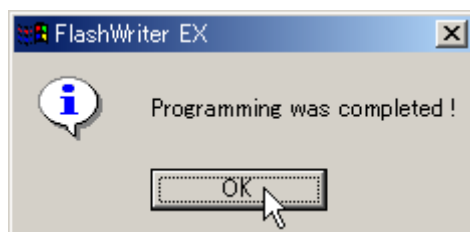
続いて FlashWriter EX でフラッシュ ROM に書き込みを実行する場合の例を記載します。

(SH7144F 内蔵フラッシュ ROM に書込む場合)

- 3) 「File Tool Bar」  の「Flash Writer EX」ボタン  をクリックし、Flash Writer EX^{※1} を起動します。



- 4) FlashWriter EX のダイアログが表示されます。フラッシュ ROM を選択、CPU クロック周波数を設定し「Start」ボタンをクリックすると FlashROM に書き込みを開始します。
- 5) 書き込みが完了したら「OK」ボタンを押し FlashWriter EX を終了します。
詳細は Flash Writer EX のマニュアルをご参照ください。



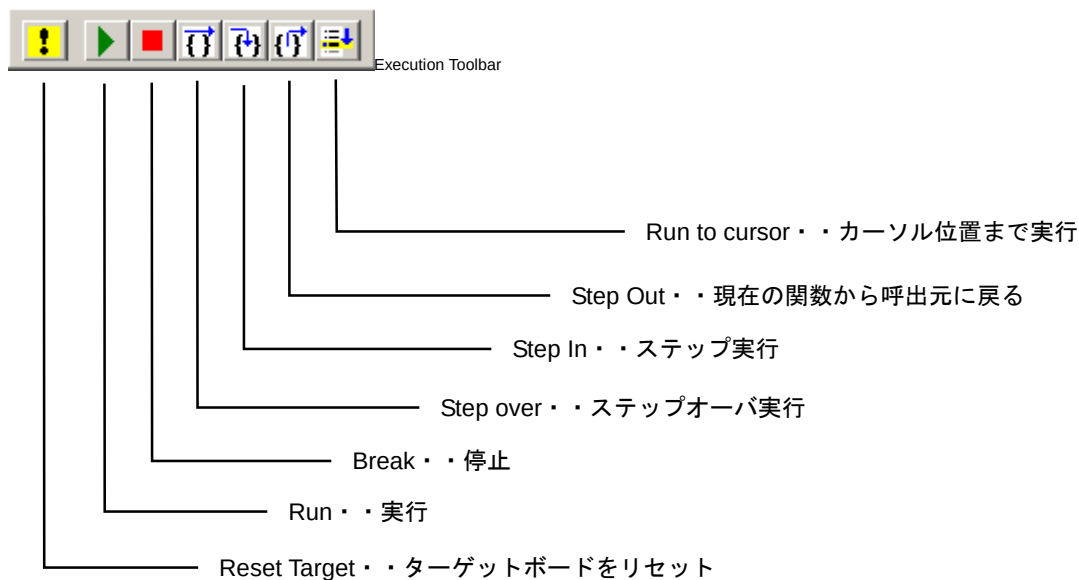
NOTE

- ※ 1 : XrossFinder から Flash Writer EX を初めて起動する場合、「Flash Writer EX」ボタン  のクリックでファイルダイアログが表示されるので、Flash Writer EX のインストールディレクトリにある FWEX.exe を指定してください。
- ※ 2 : SH2 の場合、KeyCode が 0xFFFFFFFF の状態では書き込みはできません。
- ※ 3 : FlashWriter EX で生成される書き込みモジュールに対応致しました。

3-6. プログラムを実行、停止する

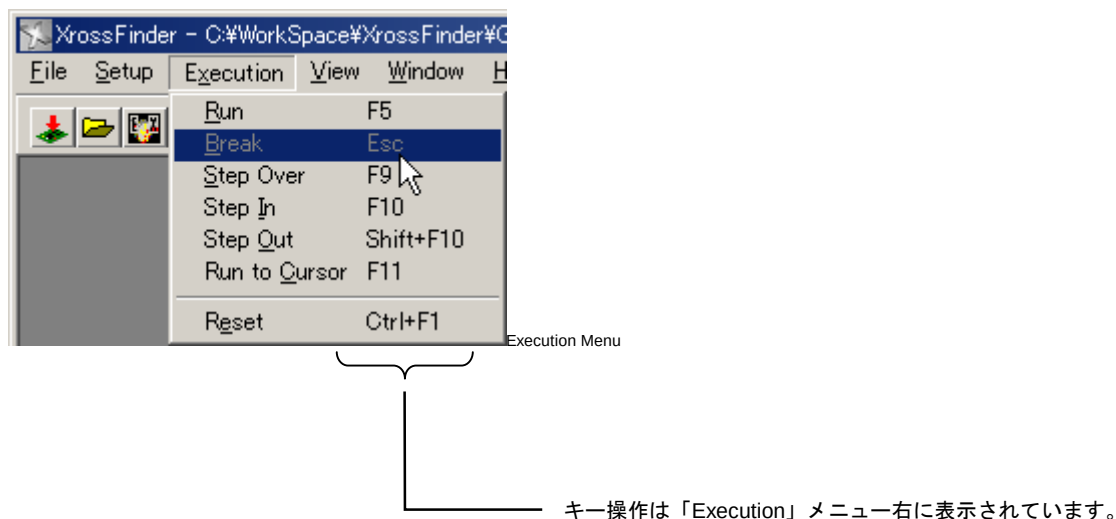
3-6-1. 「Execution Toolbar」での操作

ユーザプログラムの実行と停止は「Execution Toolbar」にあるボタンで行います。



3-6-2. メニューやキーボードでの操作

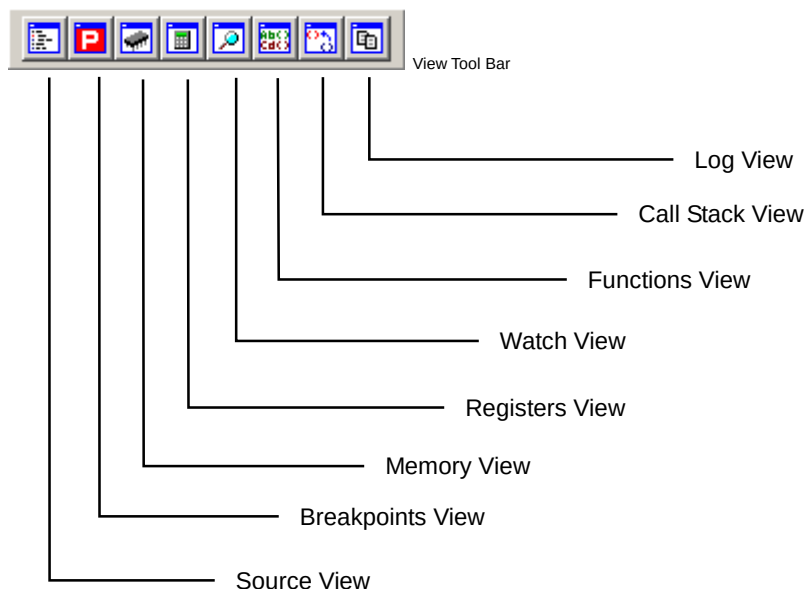
「Execution」メニューやキーボードでも操作できます。



3-7. ウィンドウを表示する

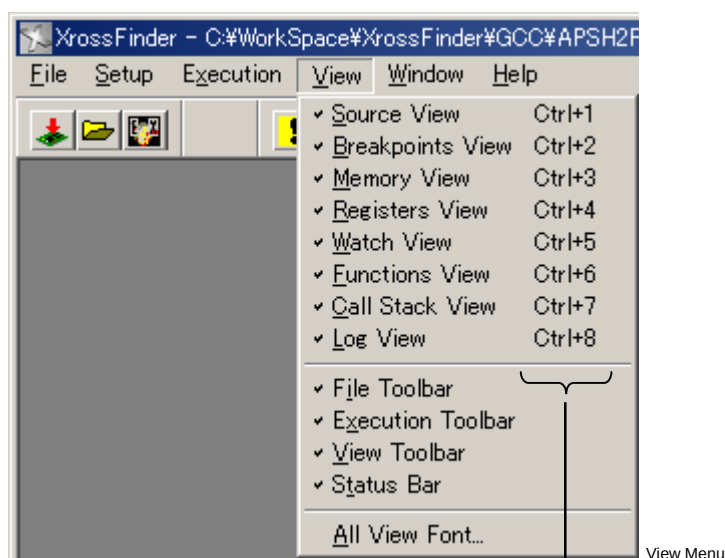
3-7-1. 「View Tool Bar」 から表示

各ウィンドウは、「View Tool Bar」にあるボタンで表示させることができます。



3-7-2. メニューやキーボードでの操作

「View」メニューやキーボードの Ctrl+'数字キー' でも操作できます。



キー操作は「View」メニュー右に表示されています。

3-8. ソースファイルを見る

- ソースファイルは「Source」ウィンドウに表示されます。

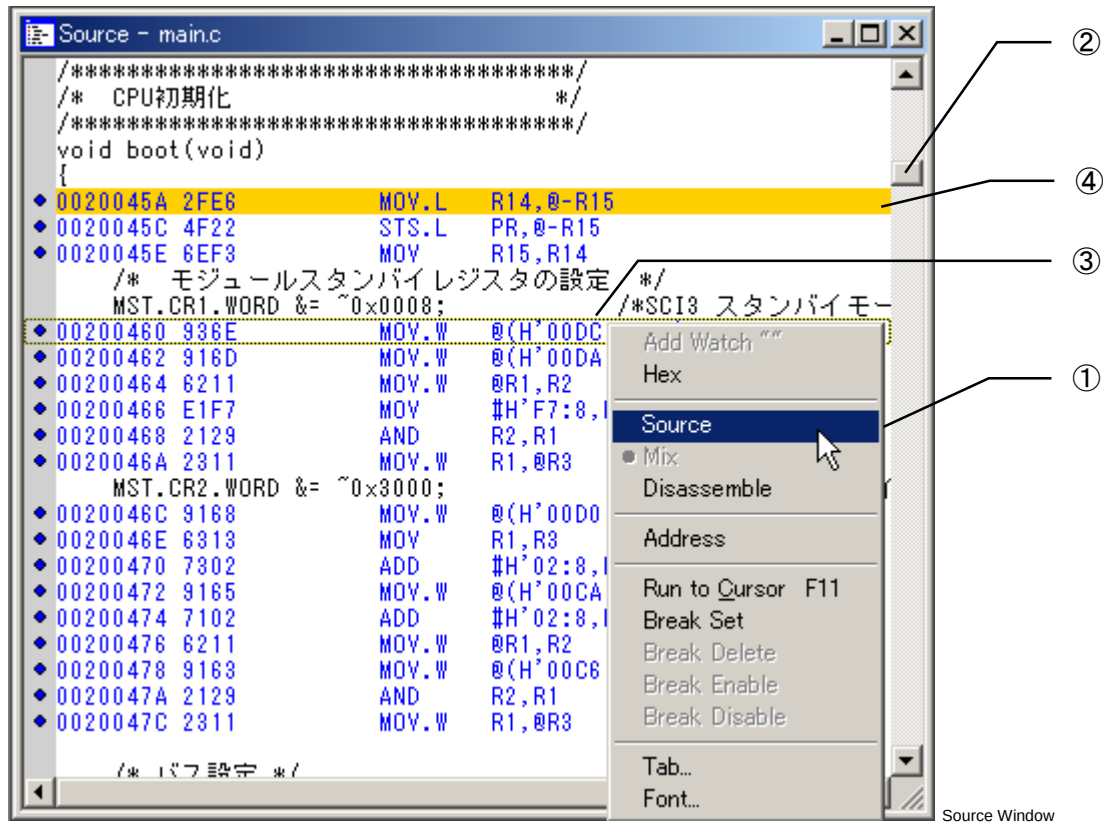


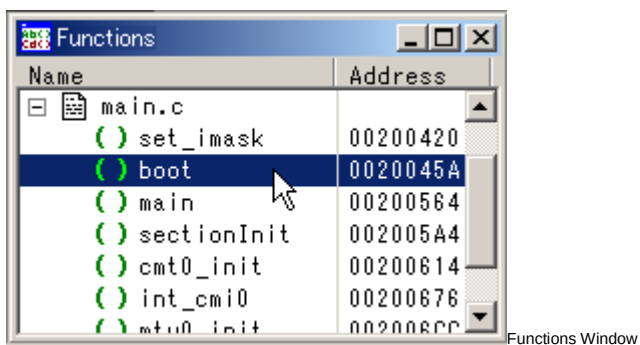
図 3-8 ソースウィンドウ

- ① ウィンドウ内を右クリックして表れるポップアップメニューで表示モードを変更できます。
 - ・ Source : ソースコード表示
 - ・ Mix : ソース・逆アセンブリ表示
 - ・ Disassemble : 逆アセンブリ表示
- ② ソースファイル内の移動はスクロールバー、キーボード矢印キー、ページキーの操作で行います。
- ③ 行を囲む点線枠は、その行が選択されていることを示すカーソルです。
- ④ オレンジ色の行は、現在停止しているアドレスです。

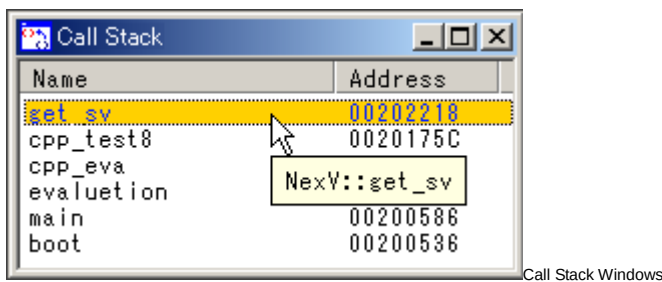
3-9. 別のソースファイルを見る

現在表示されているソースファイル以外のファイルを見るには、いくつかの方法があります。

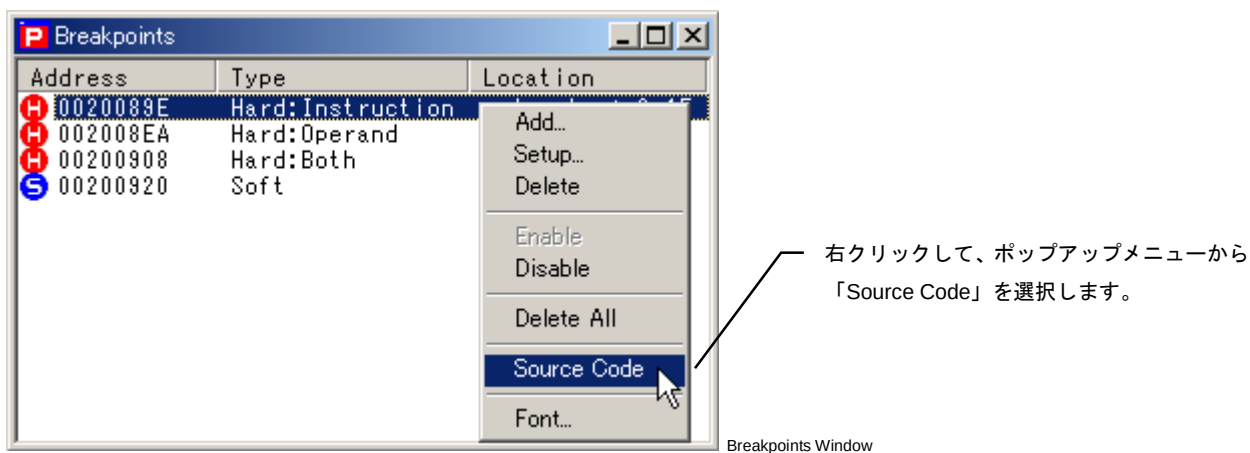
3-9-1. 「Functions」ウィンドウの項目をクリックする



3-9-2. 「Call Stack」ウィンドウの項目をクリックする



3-9-3. 「Breakpoints」ウィンドウのポップアップメニューから選択する



3-9-4. 実行が停止した場合

プログラムの実行が停止した場合には、停止アドレスに対応するソースファイルを表示します。

NOTE

※：開発環境のディレクトリを移動させた場合などには、XrossFinder はソースファイルを見つけられないことがあります。
その場合、ファイルダイアログが表示されますので、対応するソースファイルを指定してください。

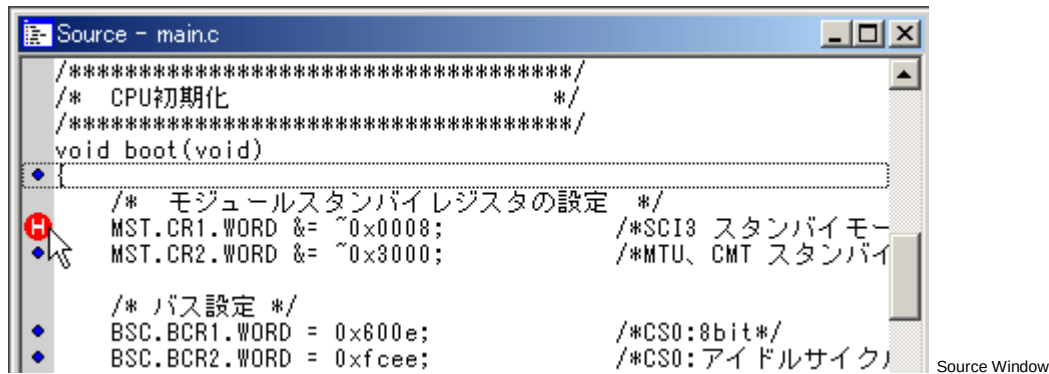
※ソースファイルの表示履歴機能：ソースファイルの表示履歴を記憶しています。Alt+左矢印で16段階まで表示を戻すことができます。

3-10. ブレークポイントを設定する

ブレークポイントを設定するには、いくつかの方法があります。

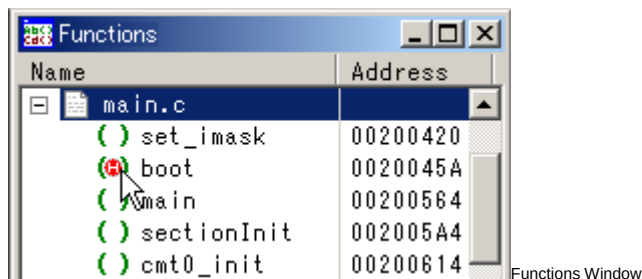
3-10-1. 「Source」ウィンドウで設定

「Source」ウィンドウの左端の青色ドットをクリックします。



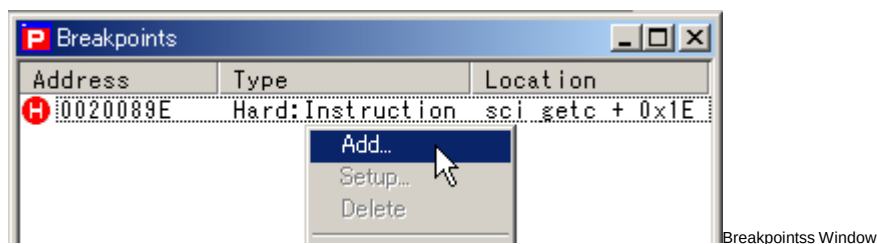
3-10-2. 「Functions」ウィンドウで設定

「Functions」ウィンドウで関数名の左にある括弧アイコンをクリックします。



3-10-3. 「Breakpoints」ウィンドウで設定

「Breakpoints」ウィンドウを右クリックし、ポップアップメニューから「Add」を選択して、ダイアログで設定します。



各ウィンドウにおいて、ブレークポイントの状態（有効/無効）を変更するには、キーボードの Enter キーや Space キーを使います。

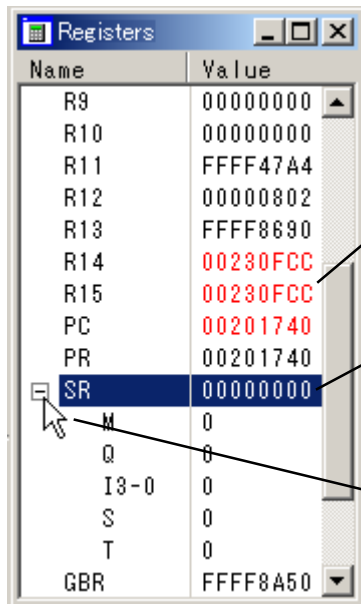
NOTE

- ※1: ブレークポイントには、ハードウェア・ブレークとソフトウェア・ブレークがあります。ハードウェア・ブレークは、CPUの種類によって同時に有効状態にできる数に制限があります。ソフトウェア・ブレークはプログラムの実行コードを書き換える為、ROM上のプログラムには使用できません。XrossFinderではハードウェア・ブレークに空きがあればこれを優先して使用します。
- ※2: 「Run to cursor」機能を使用すると、ブレークポイントと同様にソースコードの特定の位置で実行を停止させることができます。「Source」ウィンドウで停止させたい行にカーソルを移動し、「Execution Tool Bar」にある「Run to cursor」ボタンをクリックするか、右クリックのポップアップメニュー選択により操作できます。「Execution」メニューの「Run to cursor」またはF11キー、または右クリックのポップアップメニューから選択することでも操作できます。
- ※3: ブレークポイントの設定および「Run to cursor」は、ユーザプログラムが実行中でも使用することができます。ただし、プログラムの実行をわずかな時間中断させるため、時間に厳格な処理の実行中に使用するのは注意が必要です。

3-1-1. CPU レジスタの値を見る、変更する

3-1-1-1. CPU レジスタの値を見る

CPU の各レジスタの値は「Registers」ウィンドウに表示されます。



Registers Window

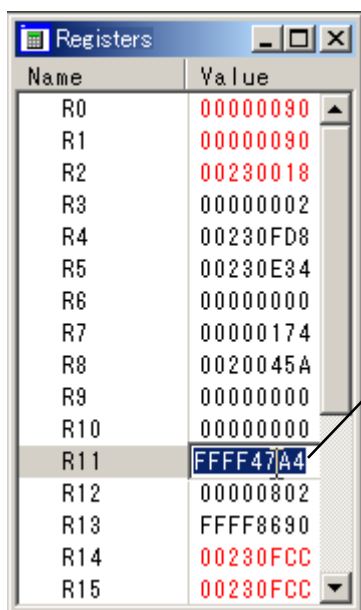
実行が停止するごとに、変化した値は赤色で表示されます。

ステータスレジスタは、ビットグループごとのツリー構造でも表示されます。レジスタ数の多い CPU では一部のレジスタグループもツリー構造で表示されます。

ツリーを展開するには、[+]マークをクリックするか、選択した状態で矢印キー[→]を押します。

3-1-1-2. CPU レジスタの値を変更する

ウィンドウ内で CPU レジスタの値が変更できます。



Registers Window

レジスタを選択し、クリックするか Enter キーを押します。入力値を確定するには Enter キーを押します。入力をキャンセルするには Escape キーを押します。

右クリックでポップアップメニューの「Modify」を選択することでも入力できます。

3-12. メモリの内容を見る、変更する

ターゲットボードのメモリ内容を表示、変更するには「Memory」ウィンドウを使います。

3-12-1. メモリ内容を表示する

ウィンドウ上部にあるバーの各入力ボックスに必要な項目を設定します。
TAB キーを使うことで各ボックスを移動できます。

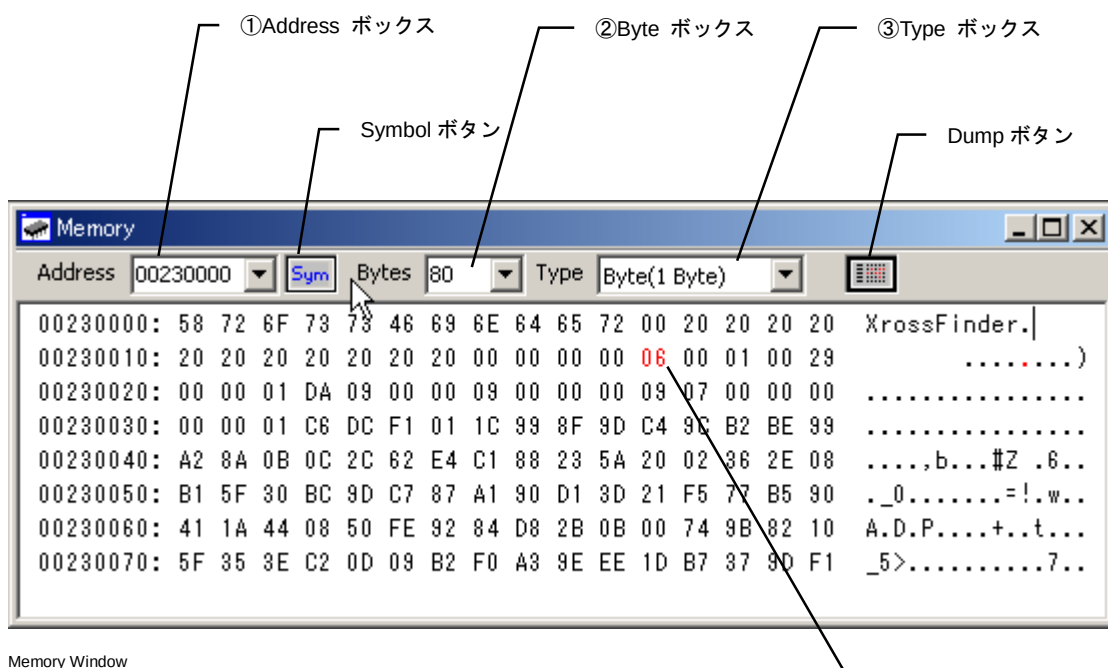


図 3-12 メモリウィンドウ

実行が停止するごとに、変化した値は赤色で表示されます。

① 「Address」ボックス

表示したいメモリ領域の先頭番地（16 進）を入力します。

「Symbol」ボタンを押し、リストからアドレスを選択することもできます。指定したい項目をダブルクリックするか、選択して Enter キーを押すことで自動的にアドレスが入力されます。

② 「Bytes」ボックス

表示するサイズ（16 進バイト数）を入力します。

登録済み項目から選択するか、キーボードから入力します。

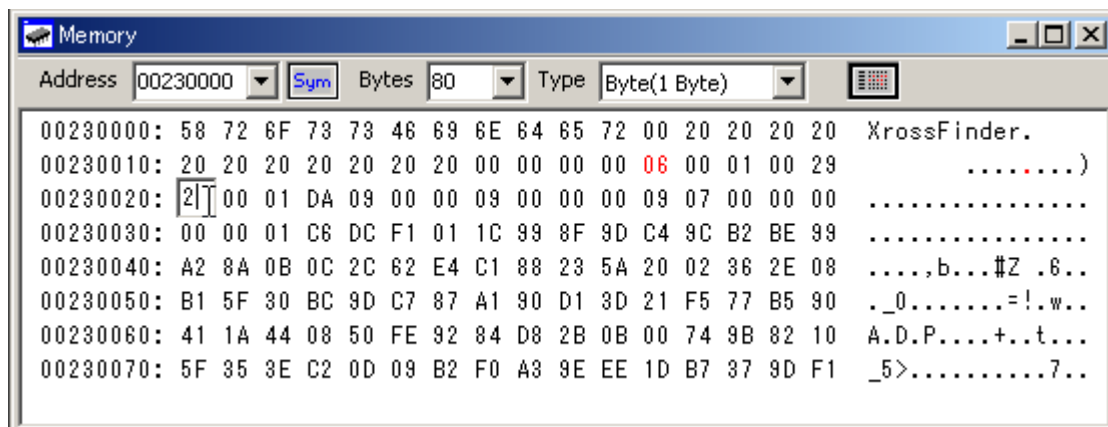
③ 「Type」ボックス

表示形式を選択します。

必要な項目を設定し、いずれかのボックスで Enter キーを押すか、バーの右端にある「Dump」ボタンをクリックします。

3-12-2. メモリ内容を変更する

変更したいデータの場所へカーソルを移動し、Enter キーを押し値を入力してください。
値を確定するには Enter キーを押します。入力をキャンセルするには Escape キーを押します。



Memory Window

NOTE

- ※ 1 : 変更したいデータへカーソルを移動させ、入力が可能な形式のキー（たとえば 16 進表示なら '0' - '9'、'A' - 'F'、'a' - 'f'）を押しても編集が可能です。
- ※ 2 : 指定範囲のメモリ領域を固定のデータ値で埋め尽くすには、「Setup」メニューの「Memory」の選択で表示される「Memory Fill」ダイアログから行います。
- ※ 3 : キャッシュメモリを搭載した CPU の場合、キャッシュ有効時に外部メモリとの整合が取れない場合があります。

3-1-3. 変数の値を見る、変更する

C/C++ソースレベルでの変数の値を表示、変更するには「Watch」ウィンドウを使います。

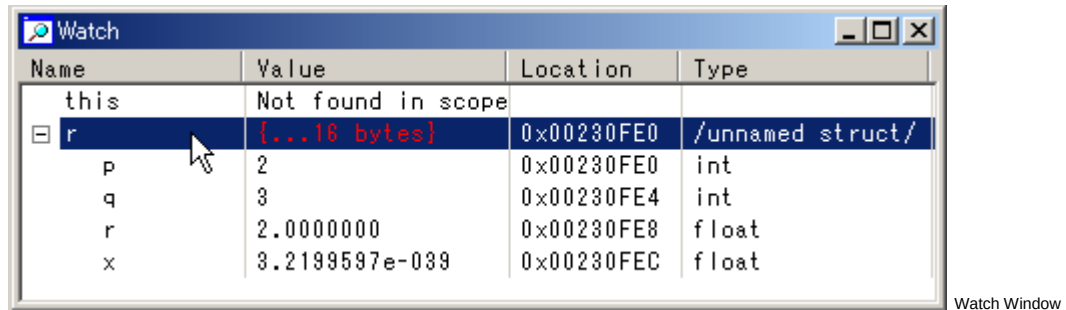
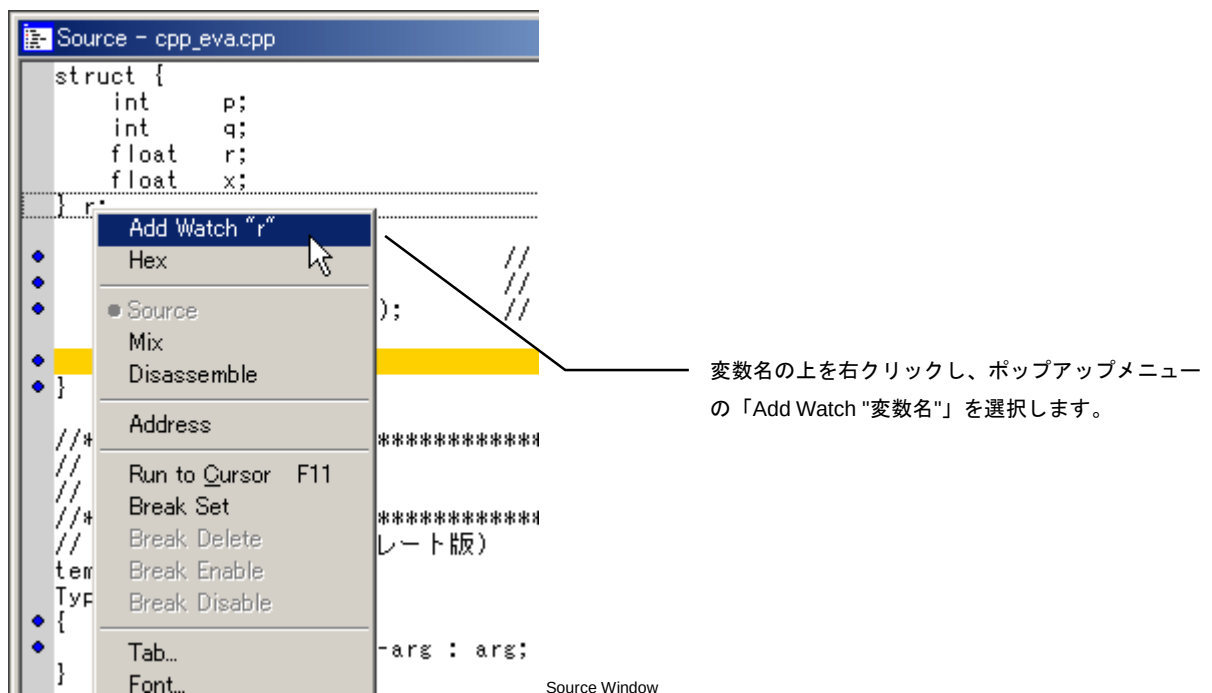


図 3-1-3 ウォッチウィンドウ

3-1-3-1. 「Watch」ウィンドウへの変数名を登録する

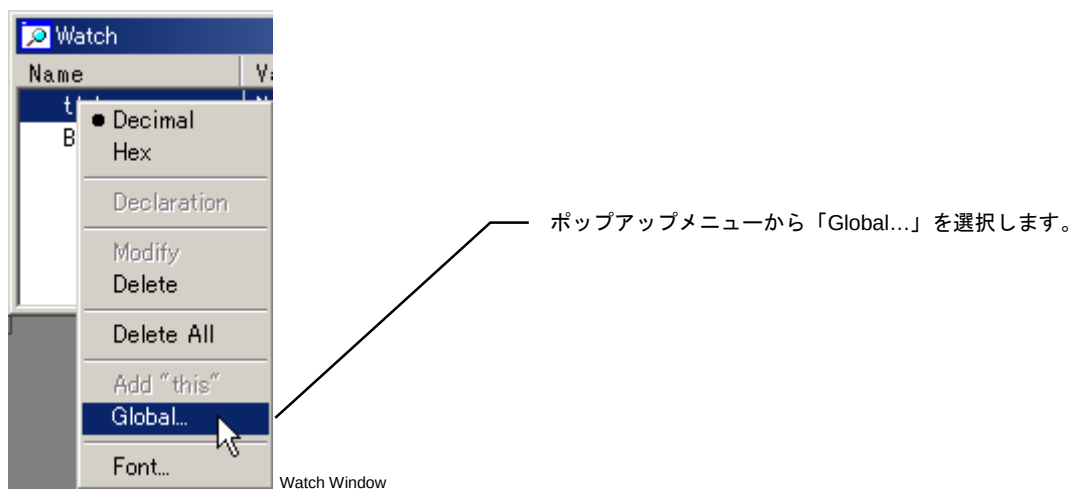
- ① 「Source」ウィンドウから登録する場合
「Source」ウィンドウ上の変数名を右クリックします。



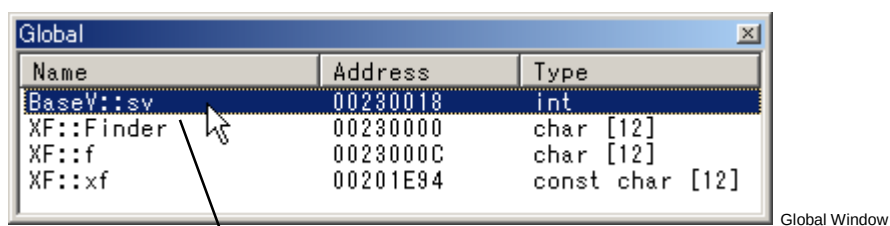
登録された変数は「Watch」ウィンドウに表示されます。

② 「Global」 ウィンドウから登録する場合。

a) 「Watch」 ウィンドウを右クリックします。



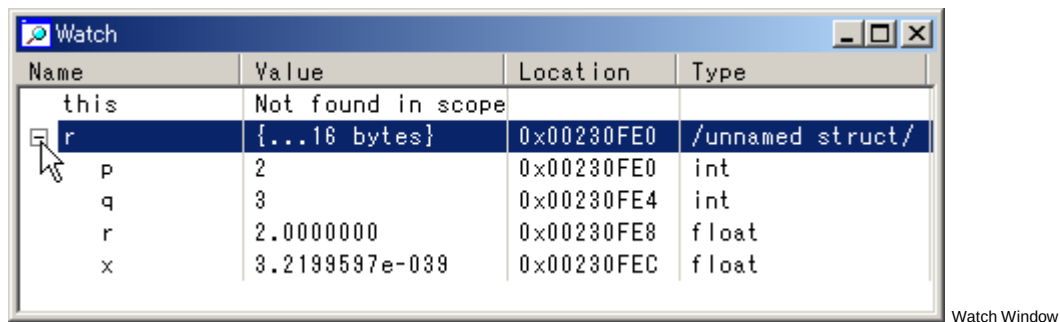
b) グローバル変数(または静的変数)を選択します。



登録したい項目をダブルクリックするか、選択して Enter キーを押すと「Watch」ウィンドウに登録されます。

3-13-2. 変数の値を見る

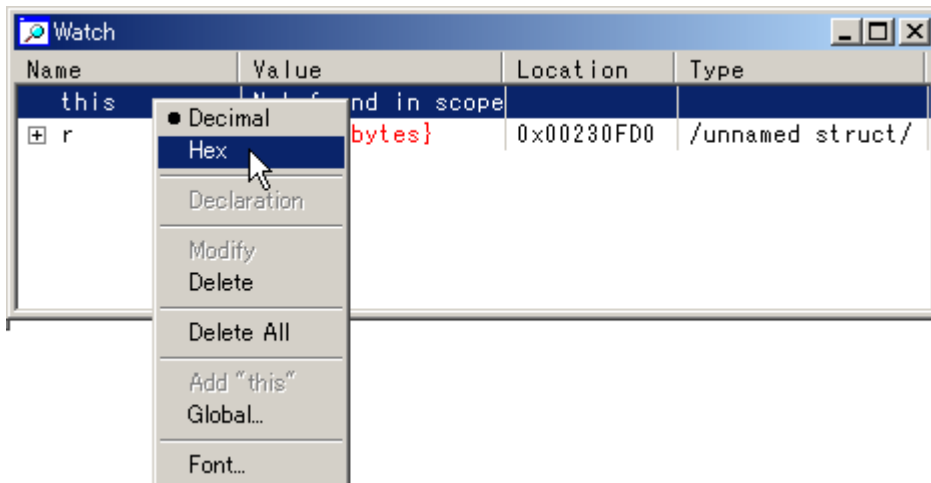
a) 構造体表示



構造体要素はツリー構造で表示されます。

ツリーを展開するには、項目の左端にある[+]マークをクリックするか、選択した状態で矢印キー[→]を押します。

b) 表示形式の変更



Watch Window

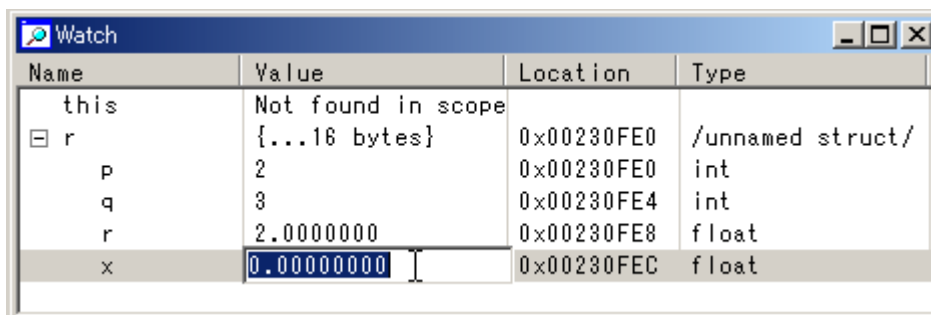
変更したい項目を右クリックし、ポップアップメニューの「Hex」または「Decimal」を選択して切り替えます。

通常、変数の値は型に従った形式で表示されています。

選択した項目がツリー構造の場合、「子」項目すべての表示を切り替えます。

変数の値は、実行が停止するごとに状態が変化すると赤色で表示されます。

3-13-3. 変数の値を変更する



Watch Window

1) 変更したい値をクリックするか Enter キーを押します。

2) 入力可能状態になりますので、値を入力します。

項目を右クリックし、ポップアップメニューの「Modify」を選択することでも入力可能になります。

3) 値を確定するには Enter キーを押します。キャンセルするには Escape キーを押します。

NOTE

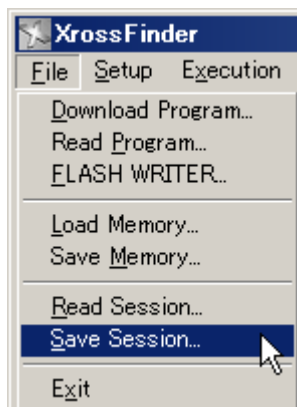
※1: 「Source」ウィンドウで変数名を左クリックすると、値を見ることが出来る場合もあります。

※2: XrossFinder では、プログラムが停止した位置での「変数スコープ」に従い、名前によって変数を探索しています。たとえば、ある関数内のローカル変数が、グローバル変数と同じ名前であれば、関数内のローカル変数の値を表示します。ただし、コンパイル時の最適化の影響などでデバッグ情報の中の変数情報が十分でない場合は、この規則が当てはまらないこともあります。

※3: XrossFinder の「Watch」は「Call Stack」と連動しています。「Call Stack」ウィンドウでは関数の呼出し階層を表示していますが、ここでの関数階層の選択を変更すると、「変数スコープ」を移動させることができます。たとえば、現在プログラムがある関数内で停止しているとすると、その関数を呼出ししているひとつ上の階層のローカル変数を参照することもできます。

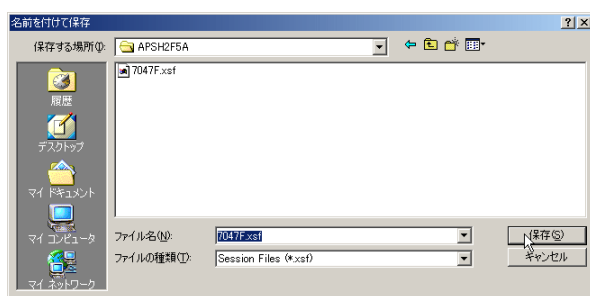
3-14. 作業を保存する、復元する

3-14-1. 作業を保存する



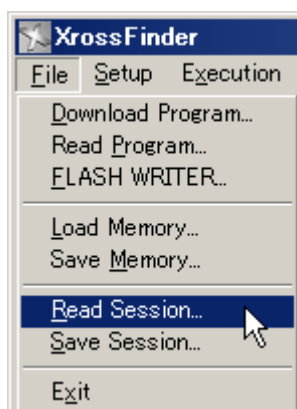
File Menu

・「Save Session」を選択します。



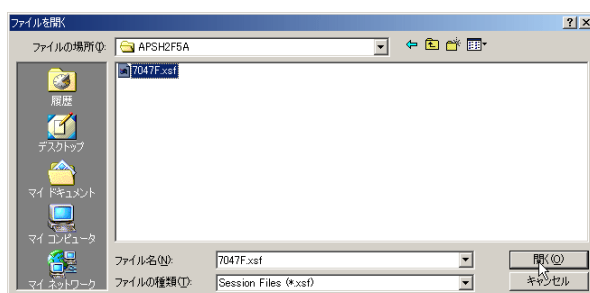
・名前を付けて保存します。

3-14-2. 作業を復元する



File Menu

・「Read Session」を選択します。



・ファイルを選択して読み込みます。

「Save Session...」および「Read Session...」は現在の作業状況を保存、復元します。これにより、中断した作業にスムーズに復帰することができます。保存されるのは以下の情報です。

- ① ターゲット設定などの各種設定。※¹
- ② 各ウィンドウの表示状態やフォント。※¹
- ③ ユーザプログラムファイルの場所および読み込み。
- ④ 「Breakpoints」ウィンドウのブレークポイント、「Watch」ウィンドウの変数名、などの登録内容。

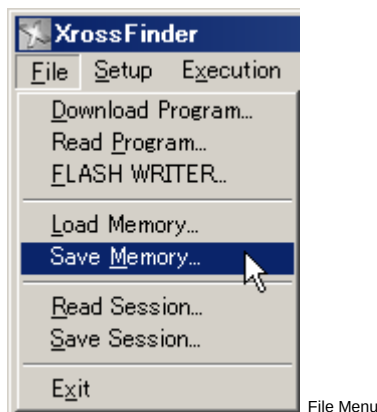
NOTE

※1：①、②の項目については、コントロールソフトの起動・終了時に自動的に保存・復元が行われます。

3-15. メモリのファイル保存、ファイル読み込み

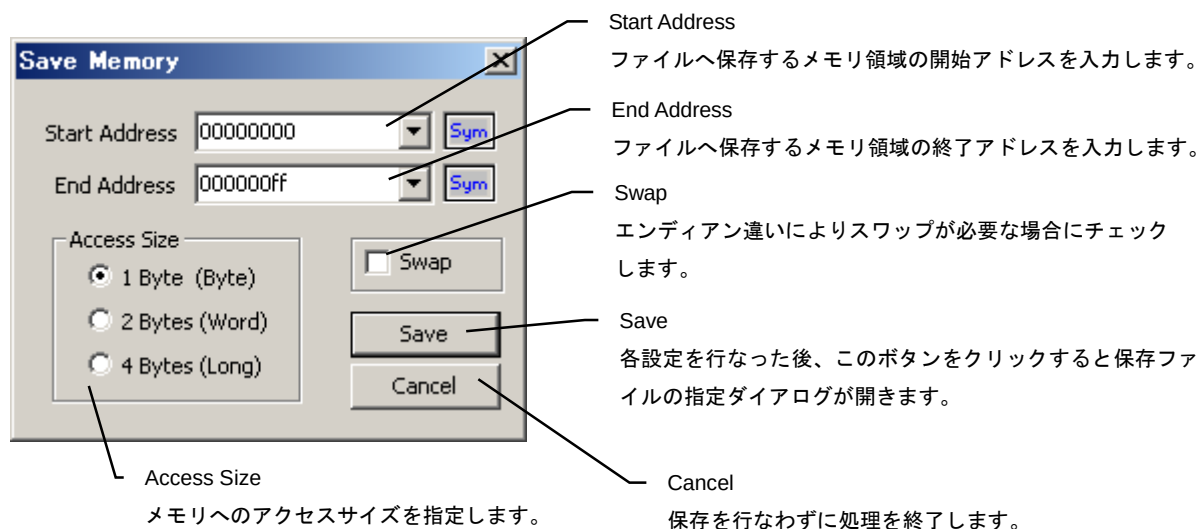
3-15-1. メモリの内容をファイルへ保存

指定したメモリ領域のデータをファイルへ保存します。

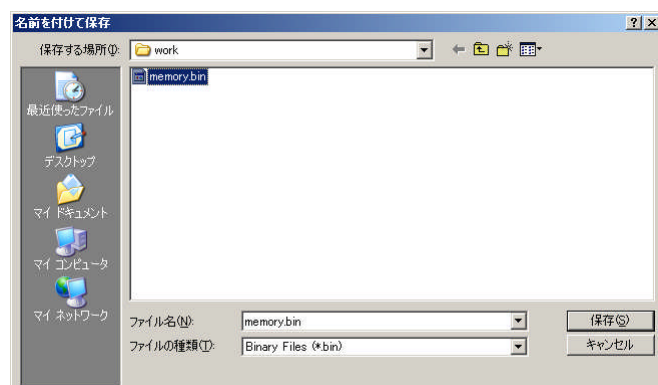


・「Save Memory」を選択します。

Save Memory ダイアログが開きますので、各設定を行い「Save」をクリックします。

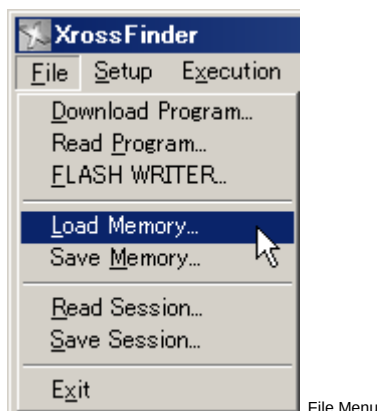


保存ファイルの指定ダイアログが開きますので、ファイル名を指定して、「保存」をクリックしてください。



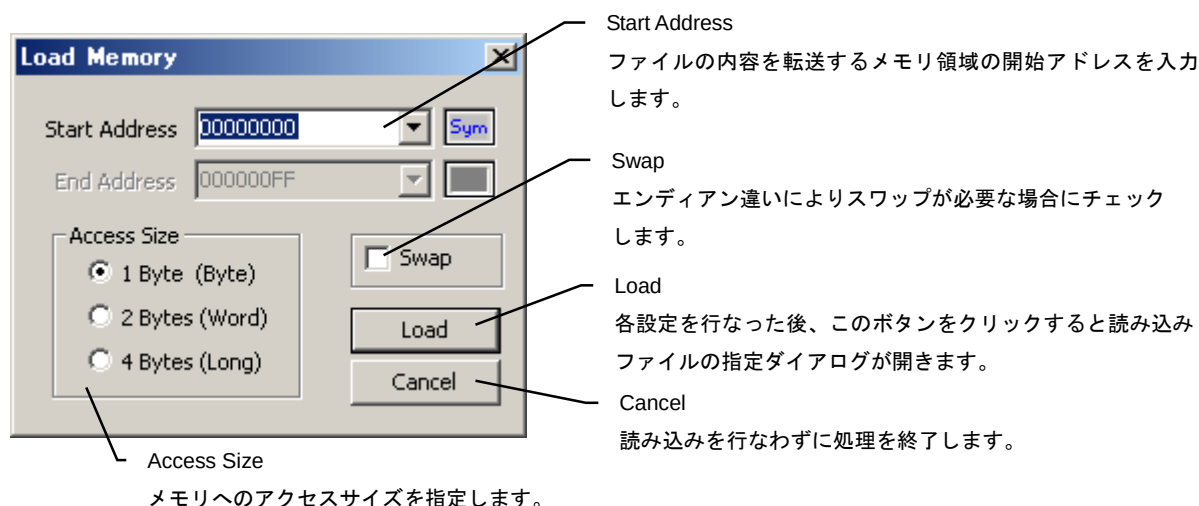
3-15-2. ファイルの内容をメモリへ読み込む

ファイルの内容を指定したメモリ領域へ転送します。

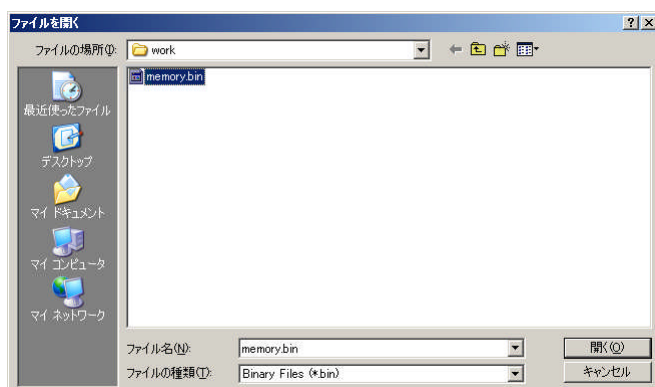


・「Load Memory」を選択します。

Load Memory ダイアログが開きますので、各設定を行い「Load」をクリックします。



読み込みファイルの指定ダイアログが開きますので、ファイル名を指定して、「開く」をクリックしてください。



4. メニューと各ウィンドウの機能と操作

4-1. ウィンドウ構成

コントロールソフトの主なウィンドウには以下のものがあります。

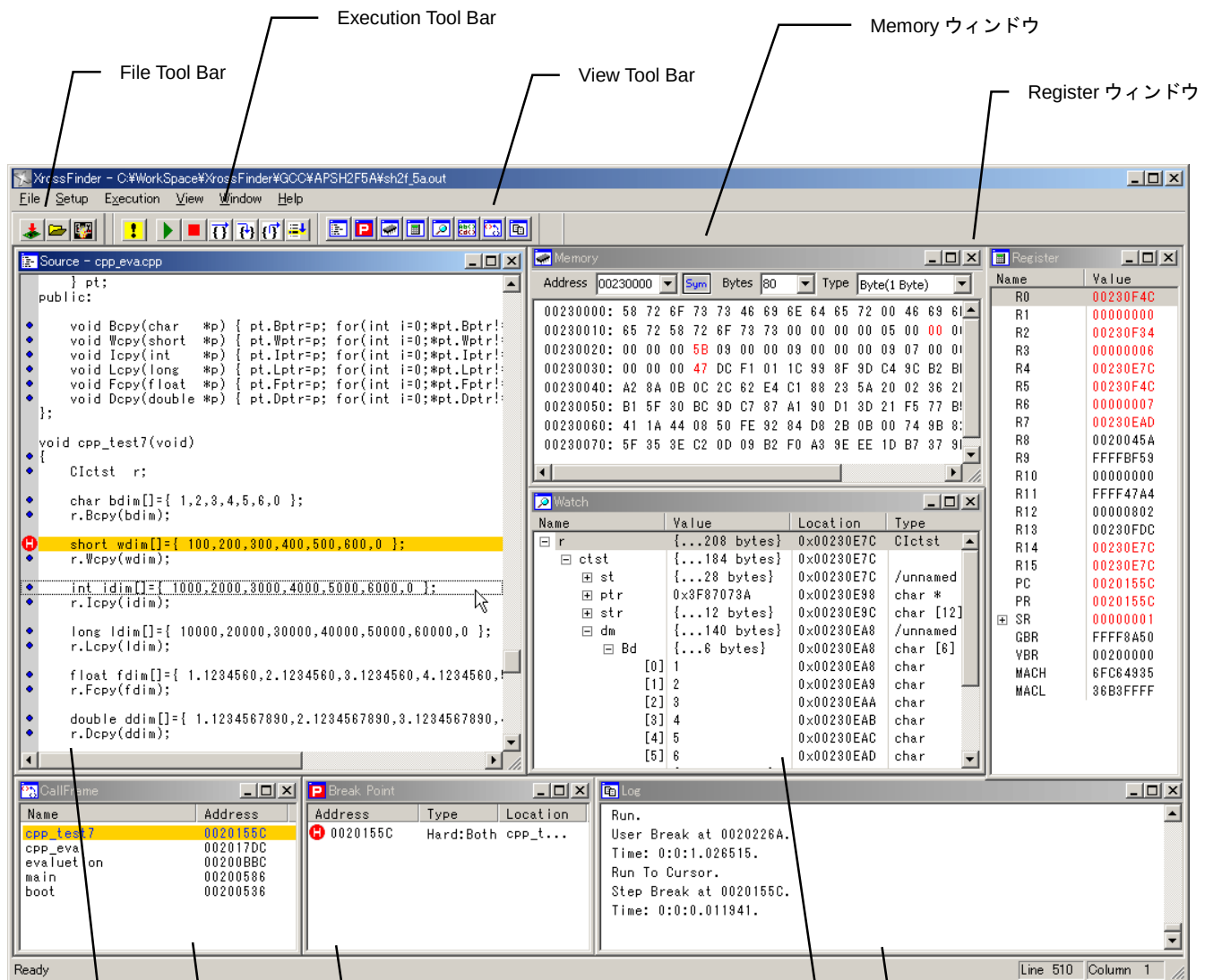


図 4-1 メインウィンドウ画面

BreakPoints ウィンドウ

Log ウィンドウ

Call Stack ウィンドウ

Watch ウィンドウ

Source ウィンドウ

4-2. メニュー

メニューバーに、各コマンドがグループ化されています。以降に各コマンドの説明を記述します。

4-2-1. 「File」メニュー [Alt+F]

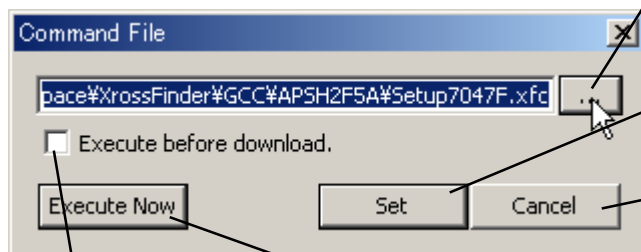
メニュー項目	ショートカット	機能概要
「Download Program...」 [D] 	—	ユーザプログラムファイルのデバッグ情報を読み込み、コマンドファイル実行し、プログラムコードをターゲットメモリへダウンロードします。
「Read Program...」 [P] 	—	ユーザプログラムファイルのデバッグ情報を読み込みます。
「FLSH WRITER...」 [F] 	—	Flash Writer EX（別売）を起動します。
「Load Memory...」 [L]	—	指定したファイルの内容をターゲットメモリへ転送します。
「Save Memory...」 [M]	—	ターゲットメモリの内容をファイルへ出力します。
「Read Session...」 [R]	—	指定された Session ファイルを読み込み、作業状況を復元します。
「Save Session...」 [S]	—	現在の作業状況を、Session ファイルに書き込みます。
「Exit」 [X]	—	コントロールソフトの実行を終了します。 ターゲット実行中の場合は強制的に停止します。

4-2-2. 「Setup」メニュー [Alt+S]

メニュー項目	ショートカット	機能概要
「Command File...」 [C]	—	「Command File」ダイアログを表示します
「Program」 [P]	—	ユーザプログラムファイルのプログラムコードをターゲットに転送します。 「Command File」ダイアログの「Execute before download」がチェックされていれば、転送前にコマンドファイルを実行します。
「Memory...」 [M]	—	「Memory Fill」ダイアログを表示します。
「System...」 [S]	—	「System setup」ダイアログを表示します。
「External Trigger...」 [T]	—	「External Trigger setup」ダイアログを表示します。

■ 「Command File」ダイアログ

コマンドファイルに関する設定を変更します。



・ エディットボックス
コマンドファイルのパス名を入力します。
右側のボタン[...]でファイルダイアログによる指定も可能です。

・ 「Set」ボタン
設定を有効にし、ダイアログを終了します。

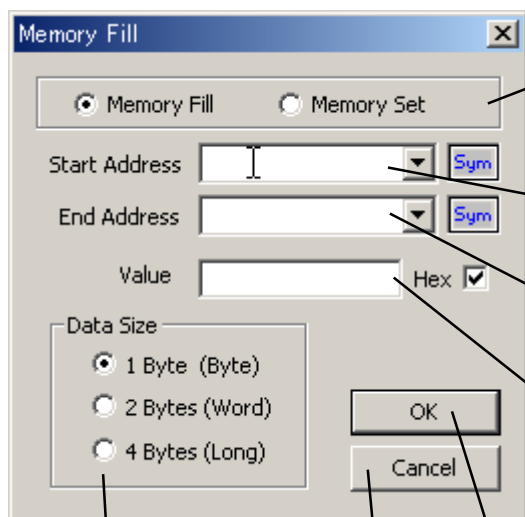
・ 「Cancel」ボタン
設定を無効とし、ダイアログを終了します。

・ 「Execute Now」ボタン
設定を有効にし、ダイアログを終了した後、コマンドファイルを実行します。

・ 「Execute before download」スイッチ
チェックが有効であれば、プログラムコード転送前にコマンドファイルを実行します

■ 「Memory Fill」 ダイアログ

指定したメモリ範囲に、値の書き込みを行います。



・ 「Memory Fill/Memory Set」 切替えスイッチ
書き込みアドレス範囲を指定します。

・ 「Start Address」※¹
開始アドレスを 16 進で入力します。

・ 「End Address」※¹
終了アドレスを 16 進で入力します。

・ 「Value」
書き込む値を入力します。
「Hex」チェックボックスが有効な場合は 16 進表記指定、無効な場合は 10 進です。
値は「Data Size」で指定されるサイズにゼロ拡張されます。

・ 「OK」 ボタン
設定内容に従って書き込みを実行します。

・ 「Cancel」 ボタン
設定を無効にしダイアログを終了します。

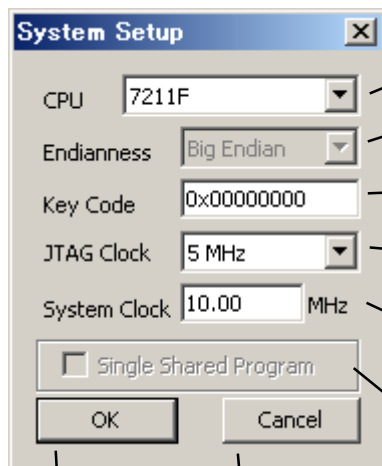
・ 「Data Size」
書き込むデータのサイズを指定します。

NOTE

※1 : 「Start Address」、「End Address」 右側にあるボタンで開く「Symbol」ウィンドウでアドレス指定もできます。
指定したい項目をダブルクリックするか、選択して Enter キーを押すことで、自動的にアドレスが入力されます。

■ 「System setup」 ダイアログ

ターゲットシステムの設定をします。



・ 「OK」 ボタン

設定を有効にしターゲットと接続を行います。

・ 「Cancel」 ボタン

設定を無効としダイアログを終了します。

・ 「CPU」

ターゲットシステムの CPU を指定します。

・ 「Endianness」

ターゲット CPU に設定されたエンディアンを指定します。
(SH-2 シリーズ等はビッグエンディアン固定)。

・ 「Key Code」 (SH7144/SH7145 等の場合)
セキュリティのための Key Code を指定します。

・ 「JTAG Clock」

本体とターゲット CPU 間の JTAG 周波数を選択します。
CPU 周辺モジュールクロック (Pφ) よりも低い値でなければなりません。

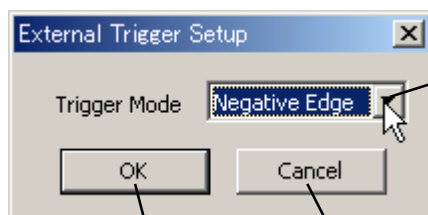
・ 「System Clock」

ターゲットボード CPU への入力クロック周波数を設定します。

・ 「Single Shared Program」
デュアルコア用の設定です。

■ 「External Trigger setup」 ダイアログ

外部トリガ入力の設定をします。XrossFinder 本体の外部トリガ入力に接続したターゲットボードからの入力信号の条件により、デバッグ中のユーザプログラムの実行を強制停止することができます。



・ 「Trigger Mode」 ※1,※2

入力信号のトリガ条件を選択します。

① 「None」: トリガ無効

② 「Negative Edge」: 立ち下がりエッジでトリガ有効

③ 「Positive Edge」: 立ち上がりエッジでトリガ有効

④ 「Both」: 立ち下がり/立ち上がり両エッジでトリガ有効

・ 「Cancel」 ボタン

設定を無効としダイアログを終了します。

・ 「OK」 ボタン

設定を有効にしダイアログを終了します。

NOTE

※1: 外部トリガが機能するには、本体の「EXT-TRG」とターゲットボードが適切に接続されていなければなりません。

詳しくは後述の「技術資料」を参照してください。

※2: 外部トリガがかかってからプログラムが停止するまでには、わずかな時間の遅れが生じます。

4-2-3. 「Execution」メニュー [Alt+X]

メニュー項目	ショートカット	機能概要
「Run」 [R] 	[F5]	現在のプログラムカウンタから、ターゲット CPU の実行を開始します。
「Break」 [B] 	[Escape]	実行を強制的に停止します。
「Step Over」 [S] 	[F9]	「Source」 ウィンドウに表示されている 1 ステートメントを実行後、停止します。 サブルーチン呼出の場合、サブルーチン実行後停止します。
「Step In」 [I] 	[F10]	「Source」 ウィンドウに表示されている 1 ステートメントを実行後、停止します。 サブルーチン呼出の場合、サブルーチン内の最初のステートメント手前で停止します。
「Step Out」 [O] 	[Shift+F10]	現在のサブルーチン実行後リターンし停止します。 デバッグ情報による戻りアドレスが特定できない場合は、アセンブリコード単位で連続ステップ実行し停止します。
「Run to cursor」 [C] 	[F11]	「Source」 ウィンドウで選択されている行の手前まで実行します。 ・「Source」 ウィンドウ左の青色ドットがある行が有効となります。 ・選択した行が、処理の流れの中にないと停止しません。 ・他の方法で停止した場合、「Run to cursor」は解除されます。 ・「Run to cursor」はプログラムの実行中でも機能します。ただし、プログラムの実行をわずかな時間ですが中断させるため、時間に厳格な処理の実行中には使用しないほうが良いでしょう。
「Reset」 [E] 	[Ctrl+F1]	デバッグ接続による CPU 内部リセットを発行し、同時に XrossFinder 本体の「RESET-OUT」端子からリセットパルスを出力します。 リセット発行後、リセットアドレスで停止します。

NOTE ステップ実行について

※1: ステップ実行による実行・停止の動作単位を「ステートメント」と定義します。「ステートメント」単位は、「Source」ウィンドウの表示モードにより異なります。また、ウィンドウ左側に「1ステートメント」を示す青色ドットが表示されます。

・「Source」モード時・・・・・・・・・・ ソースコード行が最小の単位

・「Mix」、「Disassemble」モード時・・ アセンブリコードが最小の単位

※2: 「Source」モードでの「ステートメント」はデバッグ情報に依存しています。コンパイル時に最適化オプションが指定されている場合にはソースコード行と出力コードが対応しなくなる傾向があり、ステップ実行時の進行に関しても予期しない動作を示すことがあります。

※3: 「Source」モードでの「ステートメント」実行とは、内部処理的には次の「ステートメント」を構成する最初の命令の手前まで、アセンブリコード単位での連続ステップ実行となります。このためデバッグ情報のない関数を「Step In」で呼出した場合、しばらく実行が停止しないことがあります。

※4: 「Mix」および「Disassemble」モード時のアセンブリコード単位ステップ実行においても、CPU 特性によっては複数のコードが実行されてしまう場合があります。

※5: CPU によっては、SLEEP 命令をステップ実行するとタイムアウトメッセージが表示され、実行が停止しない場合があります。この場合には「Break」を試してください。

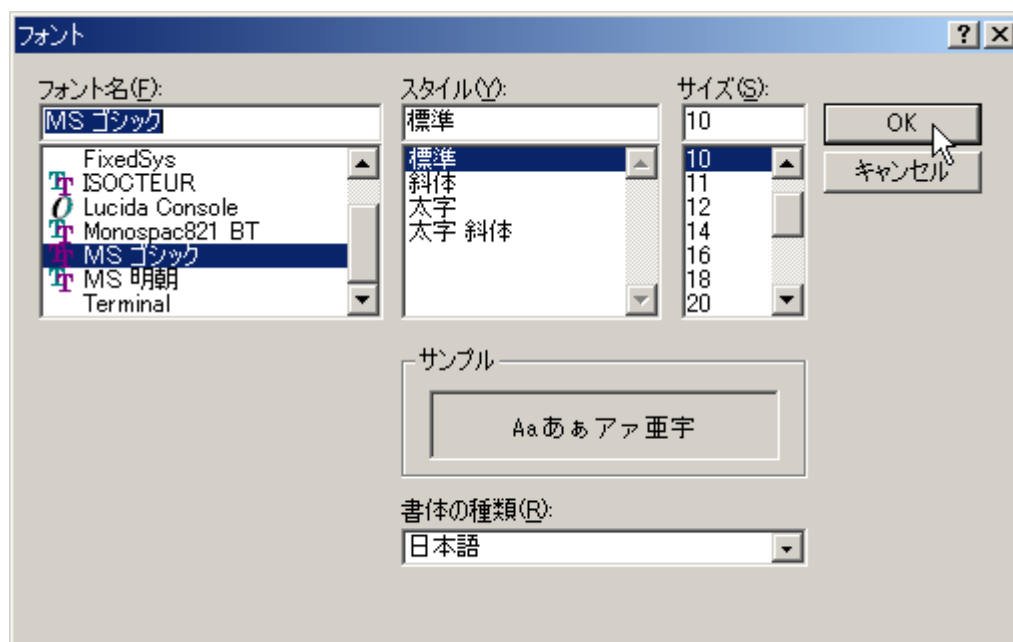
4-2-4. 「View」メニュー [Alt+V]

メニュー項目	ショートカット	機能概要
「Source View」 [S]	[Ctrl+1]	「Source」ウィンドウを表示します。
「Breakpoints View」 [B]	[Ctrl+2]	「Breakpoints」ウィンドウを表示します。
「Memory View」 [M]	[Ctrl+3]	「Memory」ウィンドウを表示します。
「Registers View」 [R]	[Ctrl+4]	「Registers」ウィンドウを表示します。
「Watch View」 [W]	[Ctrl+5]	「Watch」ウィンドウを表示します。
「Functions View」 [F]	[Ctrl+6]	「Functions」ウィンドウを表示します。
「Call Stack View」 [C]	[Ctrl+7]	「Call Stack」ウィンドウを表示します。
「Log View」 [L]	[Ctrl+8]	「Log」ウィンドウを表示します。
「File Toolbar」 [I]	—	「File Toolbar」の表示・非表示を切り替えます。
「Execution Toolbar」 [X]	—	「Execution Toolbar」の表示・非表示を切り替えます。
「View Toolbar」 [V]	—	「View Toolbar」の表示・非表示を切り替えます。
「Status bar」 [T]	—	「Status bar」の表示・非表示を切り替えます。
「All View Font...」 [A]	—	フォント設定ダイアログを表示します。

■ フォント設定ダイアログ

Windows 標準のフォント設定ダイアログです。

デフォルトでは FixedSys Font に設定されています。

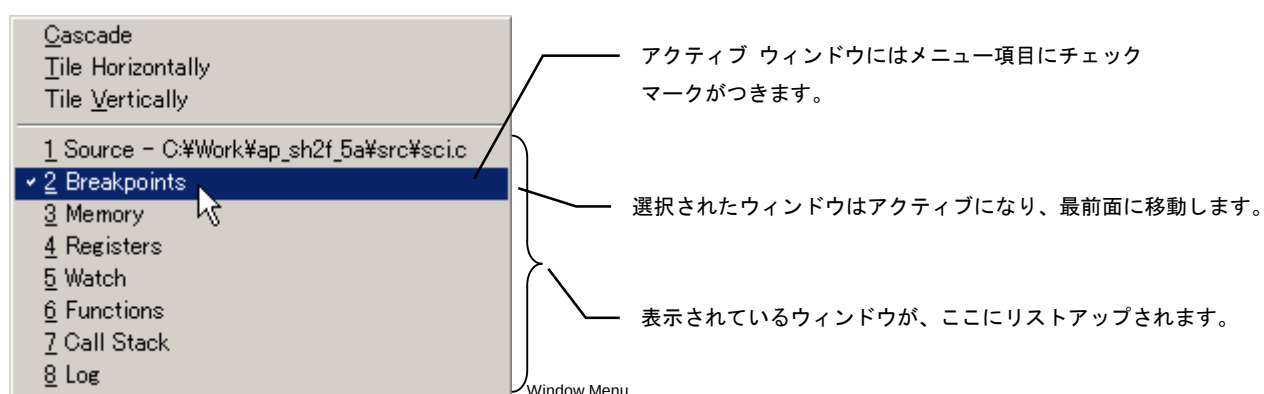


- ダイアログを「OK」で終了すると、設定内容はすべてのウィンドウにただちに適用されます。
- 各ウィンドウに対し個別にフォント設定するには、それぞれのウィンドウを右クリックして表示されるポップアップウィンドウの「Font...」を選択します。

4-2-5. 「Window」メニュー [Alt+W]

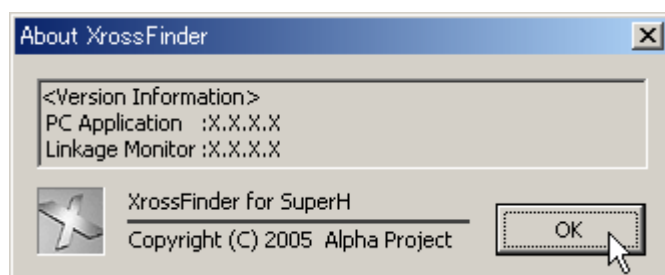
メニュー項目	ショートカット	機能概要
「Cascade」 [C]	—	ウィンドウを重ねて表示します。
「Tile Horizontally」 [T]	—	ウィンドウを横幅重視で並べて表示します。
「Tile Vertically」 [V]	—	ウィンドウを縦幅重視で並べて表示します。

■ 表示ウィンドウ一覧



4-2-6. 「Help」メニュー [Alt+H]

メニュー項目	ショートカット	機能概要
「About XrossFinder...」 [A]	—	XrossFinder のバージョン情報などを表示します。
「XrossFinder Home Page...」 [X]	—	弊社サポートページにジャンプします。



4-3. ツールバー

4-3-1. 「File toolbar」



「File」メニューの項目と同じ機能をすばやく操作するための、ボタンが並べられたツールバーです。このツールバーは移動や非表示にすることができます。各ボタンの機能は「File」メニューの解説を参照ください。

4-3-2. 「Execution toolbar」



「Execution」メニューの項目と同じ機能をすばやく操作するための、ボタンが並べられたツールバーです。このツールバーは移動や非表示にすることができます。各ボタンの機能は「Execution」メニューの解説を参照ください。

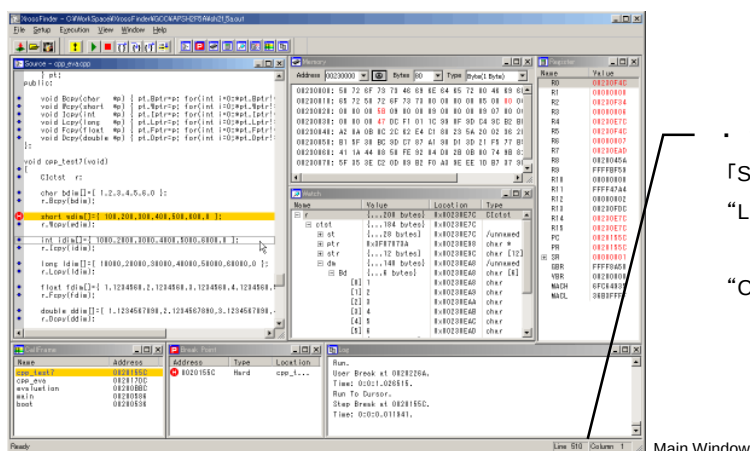
4-3-3. 「View toolbar」



「View」メニューの項目と同じ機能をすばやく操作するための、ボタンが並べられたツールバーです。このツールバーは移動や非表示にすることができます。各ボタンの機能は「View」メニューの解説を参照ください。

4-3-4. 「Status bar」

XrossFinder の状態を表示します。ステータスバーは「View」メニューより非表示にすることができます。



- ・ ステータス表示

“READY” — ターゲット停止状態

“RUNNING” — ターゲット実行状態

マウスカーソルの移動などに応じて、メニュー項目やツールバーボタンの説明を表示します。

- ・ カーソル位置表示

「Source」ウィンドウのカーソルの位置を表示します。

“Line” — 行番号

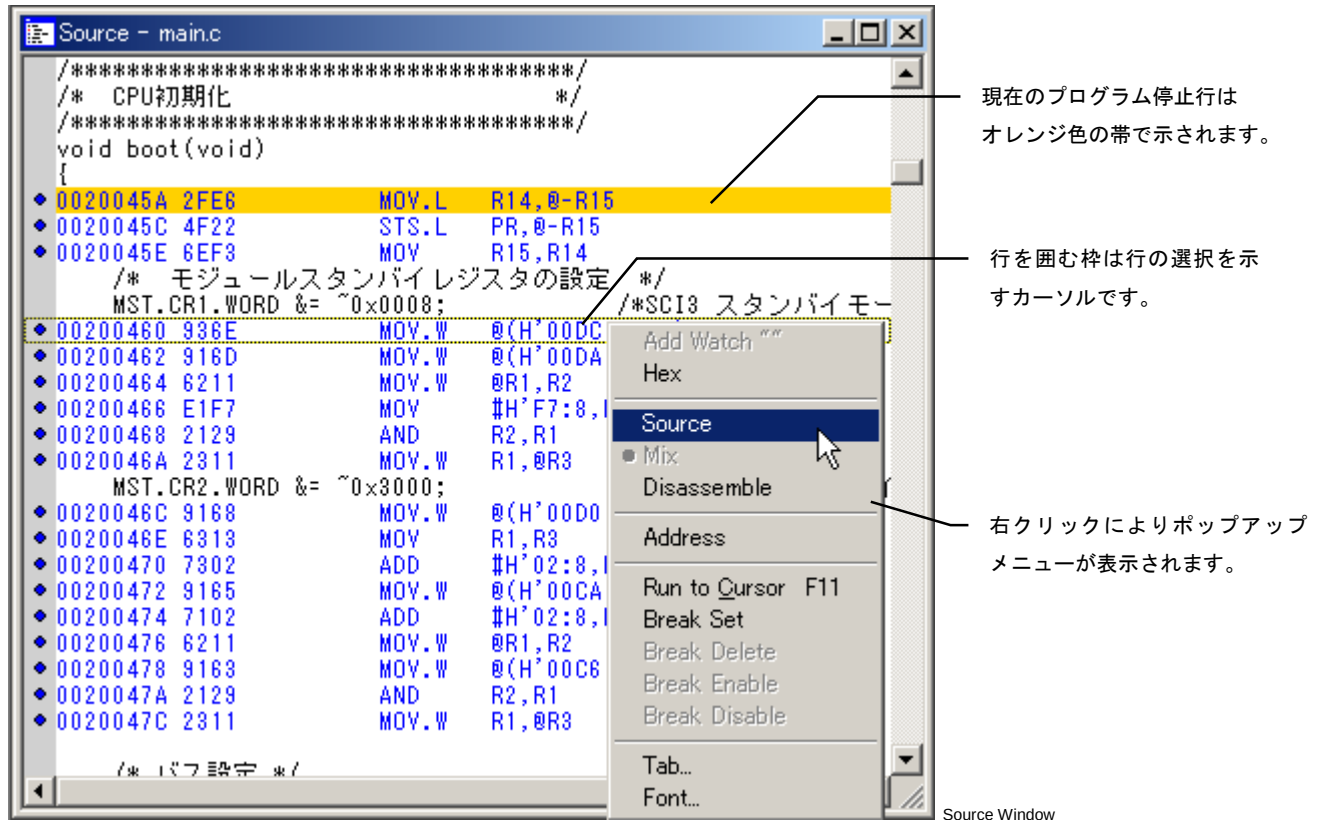
(「Mix」表示モードの場合、ソースコードファイルでの行番号を表示します。)

“Column” — ウィンドウ左端の列番号

4-4. ウィンドウ

4-4-1. 「Source」ウィンドウ

ソースウィンドウは、Cソースファイル表示、アセンブリソースファイル表示、混在表示が選択できます。ソースウィンドウ上でブレークポイントの設定や変数の表示などの操作ができます。



■ マウス操作

ボタン操作	位置	機能
左クリック	行の左の青ドット	ブレークポイントの設定／削除
	コード表示領域の変数名	変数の値の表示
	コード表示領域	カーソルの移動
左クリック 左ドラッグ	スクロールバー（縦・横）	コード表示領域のスクロール（縦・横）
左ドラッグ	コード表示領域	カーソルの移動とスクロール（縦・横）
右クリック	ウィンドウ内部全域	ポップアップメニューの表示
	コード表示領域の変数名	ポップアップメニューの表示（Watch 登録）

■ キー操作

キー	機 能
上・下矢印	カーソルの上・下への移動とスクロール
左・右矢印	コード領域の左・右へのスクロール
Page Up/Down	上・下へのページスクロールとカーソルの移動
Home	ファイルの先頭行へのカーソルの移動とスクロール
End	ファイルの最終行へのカーソルの移動とスクロール
Ctrl + 上・下矢印	カーソルを移動しない行スクロール
Ctrl + 左・右矢印	コード領域の左・右への最大スクロール
Enter	カーソル行へのブレークポイントの設定／削除
Delete	カーソル行に設定されたブレークポイントの削除
Space	カーソル行に設定されたブレークポイントの無効化／有効化
Alt+左矢印	ソースコード表示履歴を前に戻す。
Alt+右矢印	ソースコード表示履歴を次に進める。
Alt+Home	ソースコード表示履歴を最初に戻す。
Alt+End	ソースコード表示履歴を最後に進める。
Alt+上矢印	表示モードを Source→Disassemble...の順に変更する。
Alt+下矢印	表示モードを Source→Mix...の順に変更する。

■ ポップアップメニュー

項目名	機 能
Add Watch "変数名"	[Watch]ウィンドウへの登録
Hex	「Source」ウィンドウ上の変数値表示の形式の切替
Source	表示モードを C/C++ソースコードモードにする
Mix	表示モードを C/C++ソースコードと逆アセンブリコードの混合モードにする
Disassemble	表示モードを逆アセンブリコードモードにする
Address...	表示アドレス指定ダイアログの表示
Run to cursor	選択されているカーソル行の手前まで実行
Break Set	カーソル行へのブレークポイントの設定
Break Delete	カーソル行に設定されたブレークポイントの削除
Break Enable	カーソルに設定されたブレークポイントの有効化
Break Disable	カーソルに設定されたブレークポイントの無効化
Tab...	タブ設定ダイアログの表示
Font...	フォント設定ダイアログの表示

■ ソースコード表示

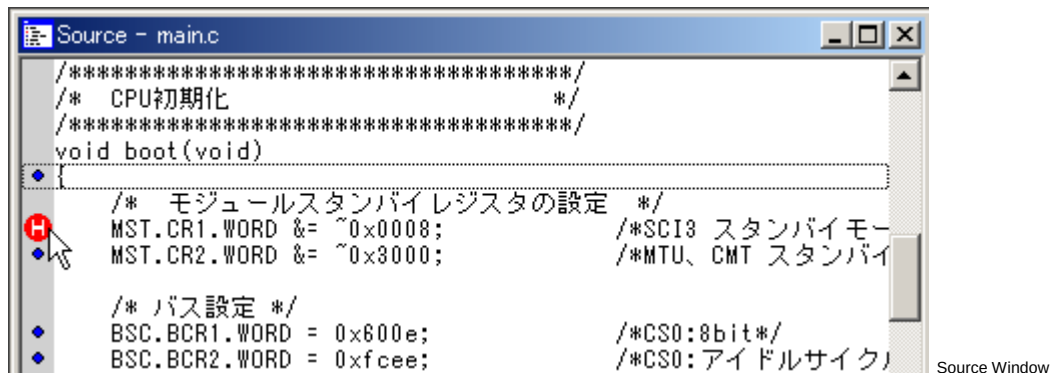
- ① プログラムコードを表示します。C/C++ソースコード、ターゲットメモリの逆アセンブリコード、ソースコードと逆アセンブリコードを対応させた混合コードを切り替えて表示できます。
- ② デバッグ情報にないメモリ領域は逆アセンブリコードで表示します。
- ③ ターゲットの実行が停止すると、自動的に停止アドレスに対応するソースコードを表示します。
- ④ 任意のソースファイルや領域を表示させるには下記の方法があります。
 - 「Functions」ウィンドウで項目を選択する。
 - 「Call Stack」ウィンドウで階層を選択する。
 - 「Breakpoints」ウィンドウの項目を選択しポップアップメニューで指定する。
 - 「Address」ダイアログでアドレスを指定する。
- ⑤ 「Call Stack」ウィンドウで関数呼出し階層を移動した場合、移動先の行が緑色の帯で表示されます。

NOTE

※1：C/C++ソースコードのファイル位置はデバッグ情報内のパス情報により特定されます。開発環境のディレクトリの移動などによりファイルが見つけれない場合、ファイルダイアログを表示してユーザによる指定を求めます。

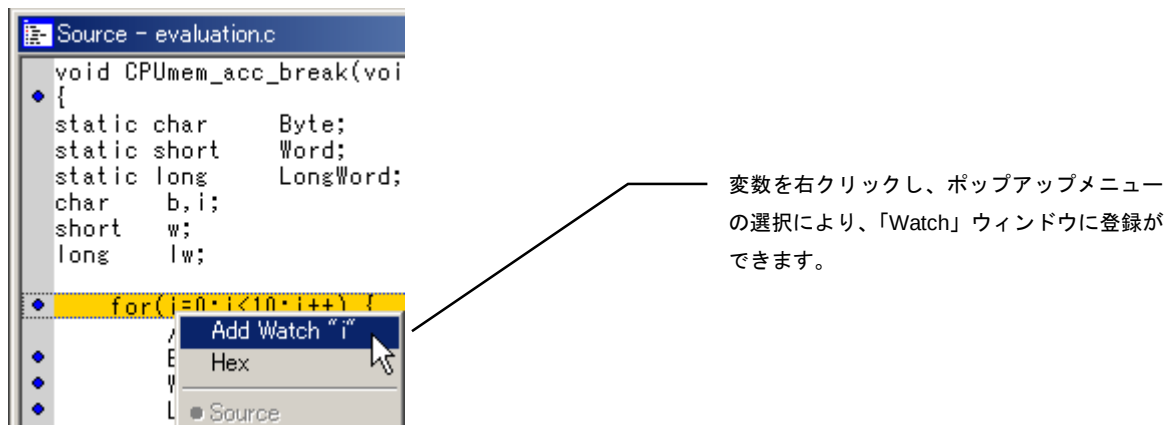
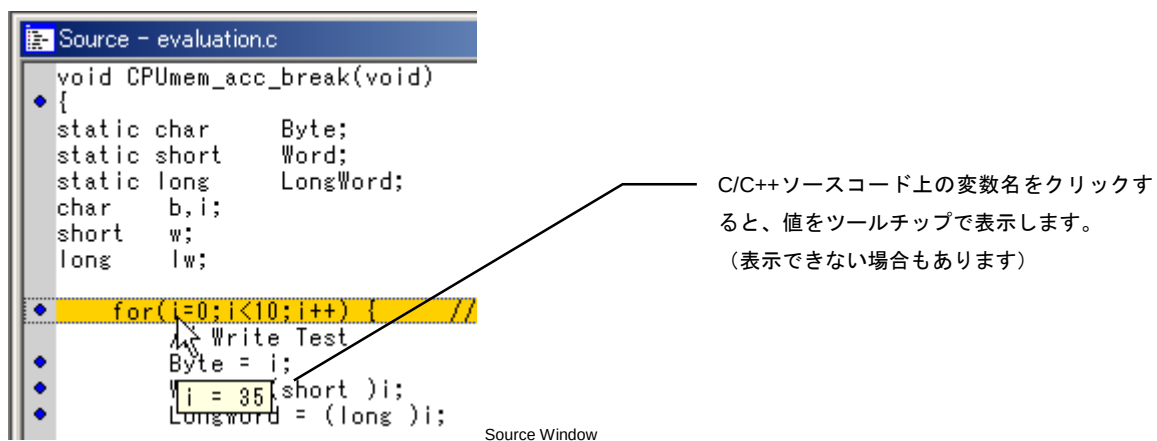
※2：ターゲット実行中に、他のソースコードファイルを表示する場合、逆アセンブリコードは表示できません。

■ ブレークポイント設定

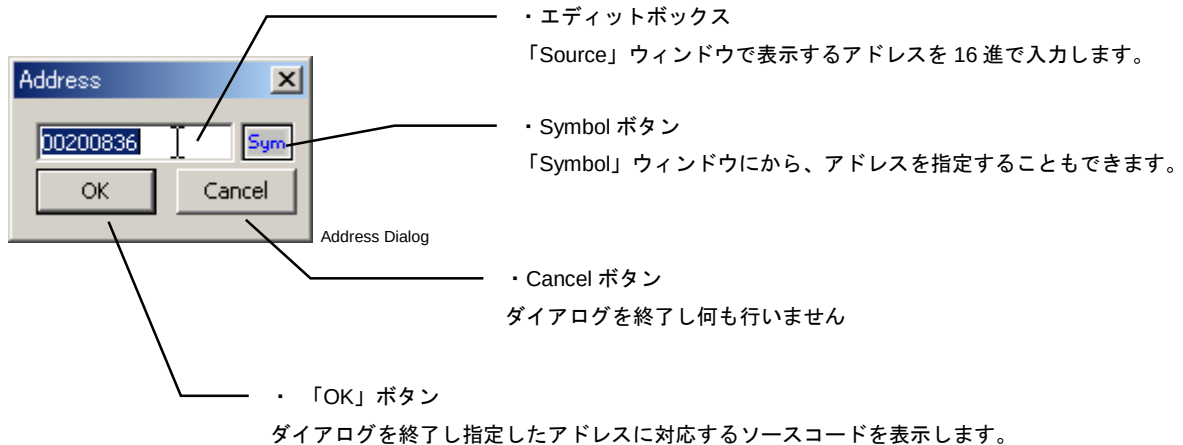


- ① マウスまたはキーボードの操作で、プログラムコード行にブレークポイントを設定します。
- ② ブレークポイント設定ができるのは、行の左に青色ドットがある位置です。
- ③ ブレークポイントの詳細は、「Breakpoints」ウィンドウの解説を参照してください。

■ 変数の値 表示と「Watch」ウィンドウへの登録

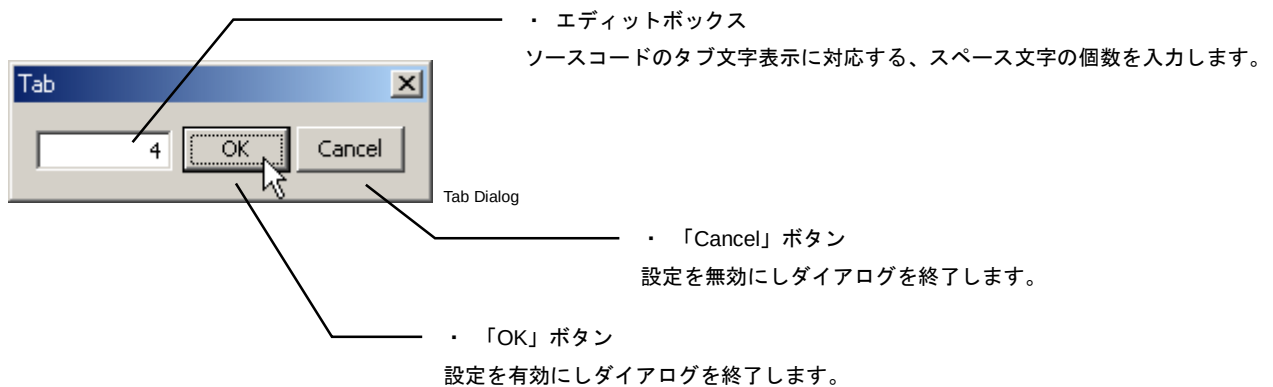


■ 「Address」 ダイアログ



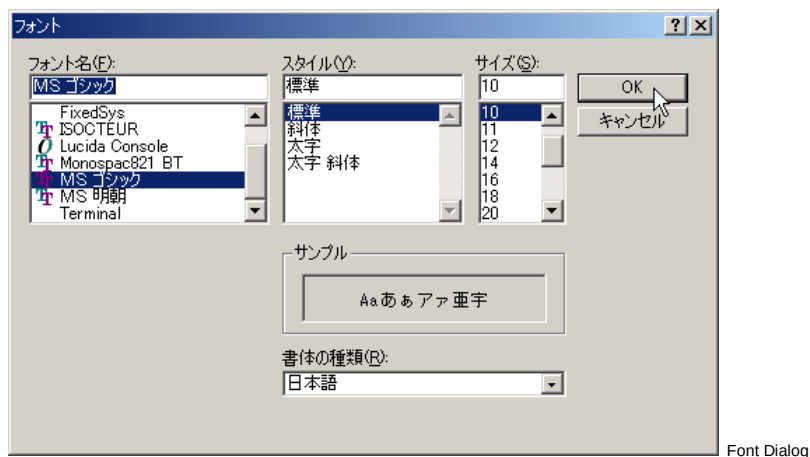
■ 「Tab」 ダイアログ

ソースコードファイルのタブ文字の表示数を指定します。



■ フォント設定ダイアログ

Windows 標準のフォント設定ダイアログです。



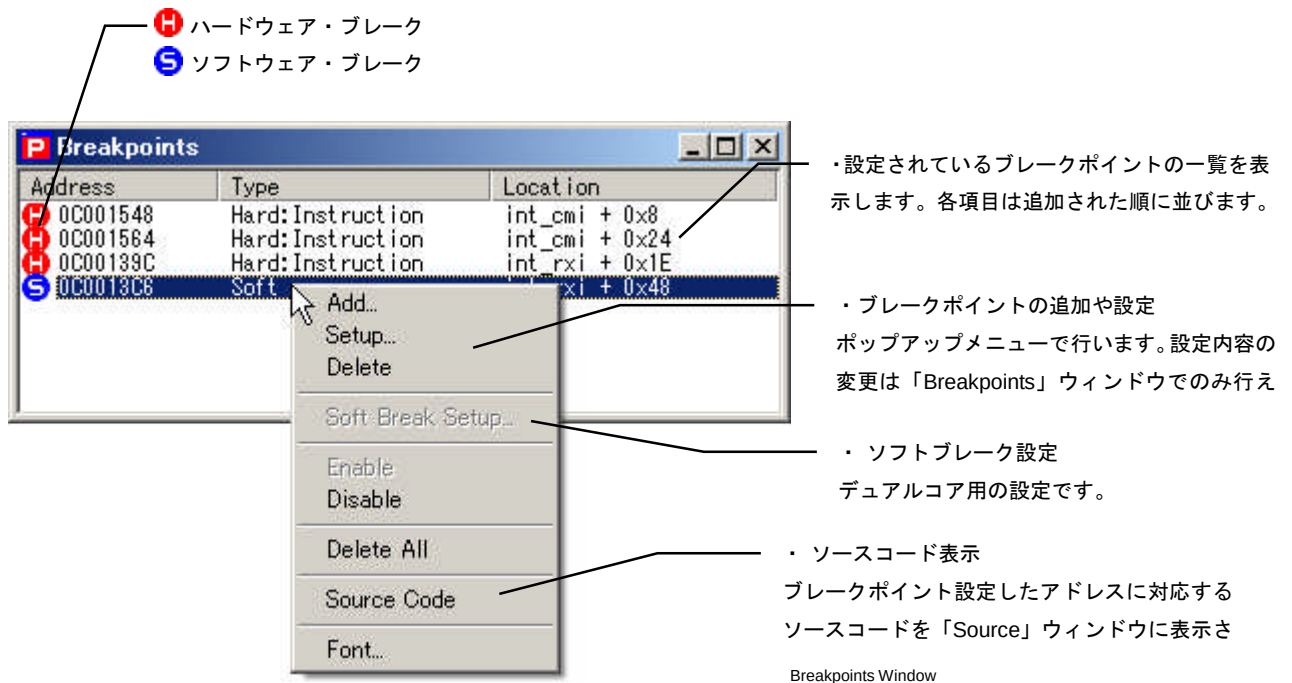
- ・ ダイアログを「OK」で終了すると、設定内容は「Source」ウィンドウにただちに適用されます。

NOTE

※1: すべてのウィンドウに対し一括してフォントを変更するには、「View」メニューの「All View Font...」で設定します。

4-4-2. 「Breakpoints」ウィンドウ

ブレークポイントの表示・設定ができます。



■ ブレークポイントの表示と設定

- ① ユーザプログラムがブレークポイントで停止した場合、対応するブレークポイント項目がオレンジ色で表示されます。
- ② ブレークポイントには、**H**ハードウェア・ブレークと**S**ソフトウェア・ブレークがあります。ハードウェア・ブレークは、同時に有効状態にできる数に制限があります。この数は CPU の種類によって異なります。一方、ソフトウェア・ブレークはプログラムの実行コードを書き換える必要があるため、ROM 上のプログラムに対しては使用できません。XrossFinder ではハードウェア・ブレークに空きがあれば、これを優先して使用します。

NOTE

- ※1: ブレークポイントの設定は、ユーザプログラム実行中でも可能です。ただし、プログラムの実行をわずかな時間ですが中断させるため、時間に厳格な処理の実行中に使用するのには注意が必要です。
- ※2: 遅延分岐命令のスロット命令にはブレークポイントを設定しないで下さい。CPU によっては予期しない動作を示すことがあります。
- ※3: CPU の種類によってはハードウェア・ブレークのオペランドブレーク指定の数に制限があります。

■ マウス操作

ボタン操作	位 置	機 能
左クリック	項目の左のアイコン	ブレイクポイントの有効化／無効化
	項目	項目選択
左ドラッグ	ヘッダーの境界	各欄（カラム）の幅の変更
左ダブルクリック	項目	項目の設定変更ダイアログの表示
左クリック 左ドラッグ	スクロールバー（縦・横）	表示領域のスクロール（縦・横）
右クリック	項目以外	ポップアップメニューの表示
	項目	ポップアップメニューの表示（項目についての操作）

■ キー操作

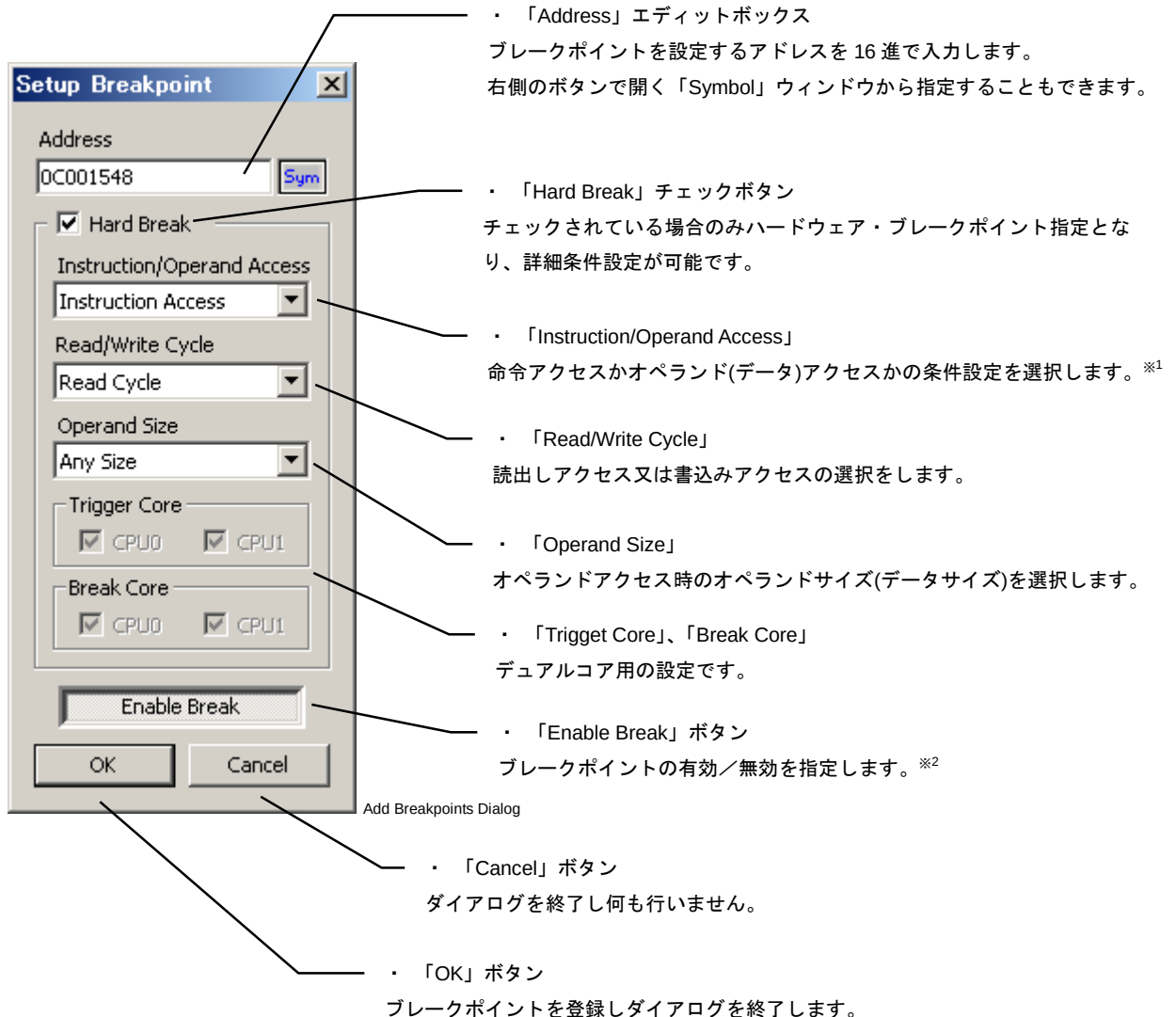
キー	機 能
上・下矢印	項目選択の移動とスクロール
左・右矢印	表示領域の左・右へのスクロール
Ctrl + 左・右矢印	表示領域の左・右への最大スクロール
Page Up/Down	上・下へのページスクロールと選択の移動
Home	先頭項目への選択の移動とスクロール
End	最後尾項目へのカーソルの移動とスクロール
Enter	選択項目の設定変更ダイアログの表示
Delete	選択項目の削除
Space	選択項目の削除の無効化／有効化

■ ポップアップメニュー

項目名	機 能
Add	新規追加のための設定ダイアログの表示
Setup...	選択項目の設定変更ダイアログの表示
Delete	選択項目の削除
Enable	選択項目の有効化
Disable	選択項目の無効化
Delete All	全項目の削除
Source Code	選択項目のアドレスに対応するソースコードの表示
Font...	フォント設定ダイアログを表示

■ 「Add/Setup BreakPoint」 ダイアログ

新規追加または設定変更時に、パラメータを設定するためのダイアログです。



NOTE

※1：オペランドを指定した場合、CPU によっては停止位置がずれることがあります

※2：ハードウェアブレークポイント数には制限があり、CPU の種類により有効数は異なります。有効数を超えた場合「Enable Break」ボタンはグレイ表示されますが、無効状態でも登録は可能です。

■ フォント設定ダイアログ

Windows 標準のフォント設定ダイアログです。

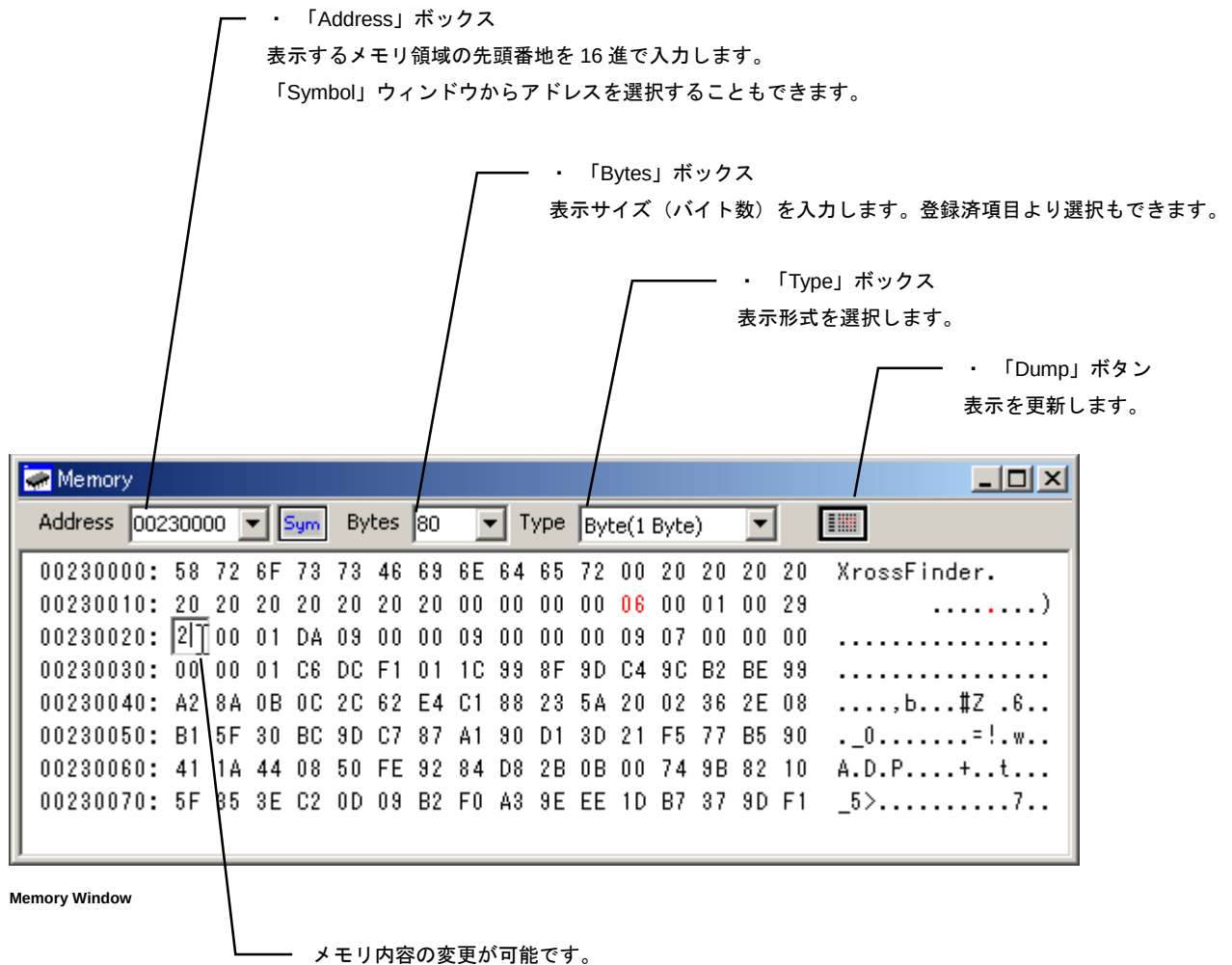
- ・ ダイアログを「OK」で終了すると、設定内容は「Breakpoints」ウィンドウにただちに適用されます。

NOTE

※1：すべてのウィンドウに対し一括してフォント設定するには、「View」メニューの「All View Font...」を選択します。

4-4-3. 「Memory」 ウィンドウ

メモリの内容を表示・変更ができます。



■ メモリ内容の表示

- ① 表示形式が選択できます。
- ② ターゲットの実行が停止すると、変化したデータは赤色で表示されます。
- ③ メモリ内容を表示するには、ウィンドウ上部の入力ボックスに必要な項目を設定し、いずれかのボックスで Enter キーを押すか「Dump」ボタンを押します。TAB キーを使うことで各ボックスを移動できます。

■ メモリ内容の変更

- ① 変更したいデータに編集カーソル(キャレット)を移動させ、Enter キーを押すと入力可能になります。
- ② 入力が可能な形式のキー（たとえば 16 進表示なら '0' - '9'、'A' - 'F'、'a' - 'f'）を押しても編集ができます。

■ マウス操作

ボタン操作	位置	機能
左クリック	データ表示領域	編集カーソル（キャレット）の移動
右クリック	データ表示領域	ポップアップメニューの表示
左ドラッグ	データ表示領域	表示内容のテキスト選択
左ダブルクリック	データ表示箇所	1 データのテキスト選択

■ キー操作

キー	機能
上・下矢印	編集カーソル（キャレット）の上・下移動とスクロール
左・右矢印	編集カーソル（キャレット）の上・下移動とスクロール
Ctrl + 左・右矢印	データ単位での編集カーソル（キャレット）の移動
Enter （データ表示箇所）	データ入力状態への移行
表示形式での入力可能キー	データ入力状態への移行と編集カーソル位置の文字の上書き
Escape （入力状態時）	入力のキャンセルとデータ入力状態の終了
Shift + 矢印	テキスト選択
Ctrl + A	全表示内容のテキスト選択
Ctrl + C	テキスト選択範囲のクリップボードへのコピー （他のアプリへ貼り付けができます）

■ ポップアップメニュー

項目名	機能
Copy	テキスト選択範囲のクリップボードへのコピー （他のアプリへ貼り付けができます）
Font...	フォント設定ダイアログを表示

■ フォント設定ダイアログ

Windows 標準のフォント設定ダイアログです。

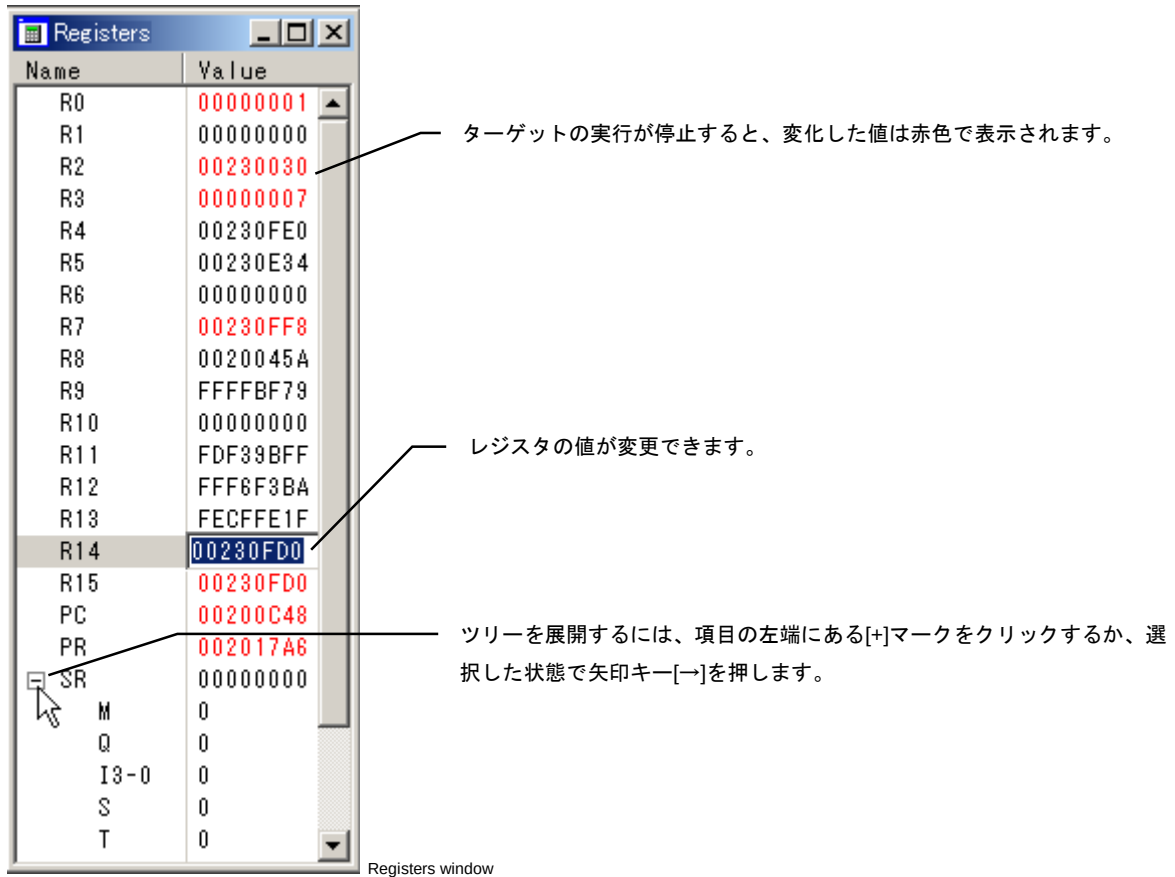
- ・ ダイアログを「OK」で終了すると、設定内容は「Memory」ウィンドウにただちに適用されます。

NOTE

※1: すべてのウィンドウに対し一括してフォント設定するには、「View」メニューの「All View Font...」を選択します。

4-4-4. 「Registers」ウィンドウ

ターゲット CPU のレジスタの値を表示します



■ レジスタ値の表示

- ① レジスタ表示値はすべて 16 進表現です。
- ② ステータスレジスタの各ビット(グループ)はツリー表示されます。レジスタ数の多い CPU では一部のレジスタグループもツリー構造で表示されます。

■ レジスタ値の変更

- ① 変更したい項目を選択し、値の表示部分をクリックするか、Enter キーを押すか、またはポップアップメニューの「Modify」を選択すると、入力可能状態になります。入力は 16 進表記で行います。
- ② 値を確定するには Enter キーを押します。入力をキャンセルするには Escape キーを押します。

■ マウス操作

ボタン操作	位置	機能
左クリック	項目の[+]/[-]	ツリーの展開／縮小(折り畳み)
	項目	項目選択
	値表示部	値入力状態への移行
左ドラッグ	ヘッダーの境界	各欄（カラム）の幅の変更
左クリック 左ドラッグ	スクロールバー（縦）	表示領域のスクロール（縦）
右クリック	項目	項目選択および ポップアップメニューの表示（項目についての操作）
右クリック	項目以外の表示領域	ポップアップメニューの表示（項目についての操作）

■ キー操作

キー	機能
上・下矢印	項目選択の移動とスクロール
Ctrl + 上・下矢印	項目選択を移動しないスクロール
左矢印	ツリーの圧縮または親項目への選択移動
右矢印	ツリーの展開、または最初の子項目への選択移動
Page Up/Down	上・下へのページスクロールと選択の移動
Ctrl+Page Up/Down	項目選択を移動しないページスクロール
Home	先頭項目への選択の移動とスクロール
End	最後尾項目へのカーソルの移動とスクロール
Enter	選択項目の値入力状態への移行
Escape (入力状態時)	入力のキャンセルとデータ入力状態の終了

■ ポップアップメニュー

項目名	機能
Modify	選択項目の値入力状態への移行
Font...	フォント設定ダイアログを表示

■ フォント設定ダイアログ

Windows 標準のフォント設定ダイアログです。

- ・ ダイアログを「OK」で終了すると、設定内容は「Registers」ウィンドウにただちに適用されます。

NOTE

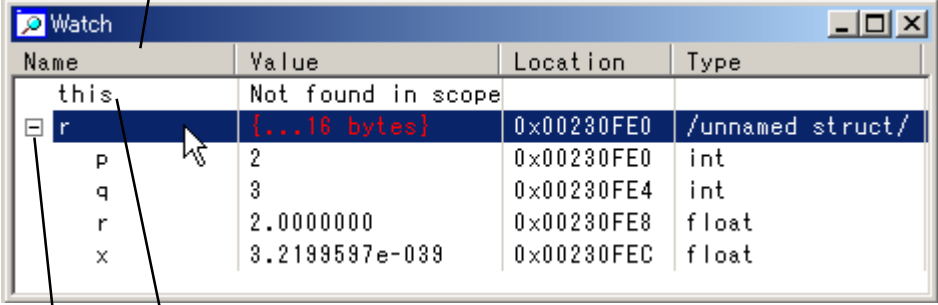
※1: すべてのウィンドウに対し一括してフォント設定するには、「View」メニューの「All View Font...」を選択します。

4-4-5. 「Watch」 ウィンドウ

変数の表示・登録・変更ができます。

■ 変数の表示

変数情報として、「Name」(変数名)、「Value」(変数値)、「Location」(割り付け場所)、「Type」(変数型)が表示されます。



C++ソースコードのクラスメンバ関数内では、暗黙の引き数"this"を登録することができます。

構造体などの場合、その各要素はツリー構造で表示されます。

- ① 変数の値は、変数の型に従った形式で表示されますが、16進表現に切り替えることもできます。この切り替えは、選択したツリー項目の「子」項目すべてが対象になります。たとえば配列変数の全要素の表示を一度に変更することもできます。
- ② 各変数の値は、実行が停止するごとに、状態が変化すると赤色で表示されます。

NOTE

※1: ビットフィールドの場合、ビット位置が「Location」欄に、ビットサイズが「Type」欄に、以下の形式で表示されます。

<表現形式>

「Location」 L:D-B
「Type」 T:S

<記号の意味>

L=ロケーション (アドレスまたはレジスタ名)
D=ビットフィールドが割り付けられているデータのサイズ(ビット数)
B=ビットフィールドが割り付けられているデータの最上位ビットから数えたビット位置
T=ビットフィールドの型
S=ビットフィールドのサイズ(ビット数)

<例>

アドレス 0x00004000 の 32 ビットデータのビット 27 からの 3 ビットに割り付けられた場合
「Location」 0x00004000:32-4
「Type」 unsigned int:3

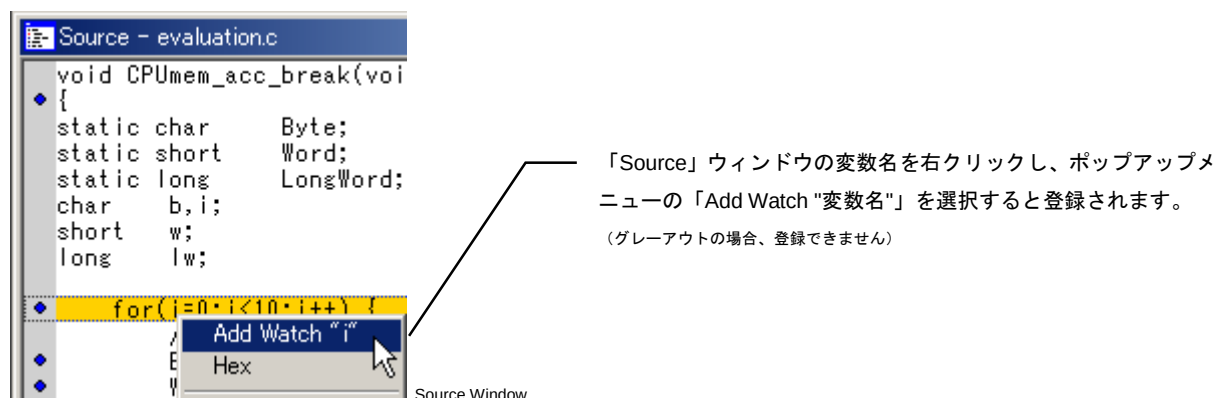
また「Value」欄には、型に従ってゼロ拡張または符号拡張された値が表示されます。

※2 : XrossFinder では、プログラムが停止した位置での「変数スコープ」に従い、名前によって変数を探索しています。たとえば、ある関数内のローカル変数が、グローバル変数と同じ名前であれば、関数内のローカル変数の値を表示します。ただし、コンパイル時の最適化の影響などでデバッグ情報の中の変数情報が十分でない場合は、この規則が当てはまらない場合もあります。XrossFinder がどの変数を参照しているかはポップアップメニューの「Declaration」で確かめることができます。

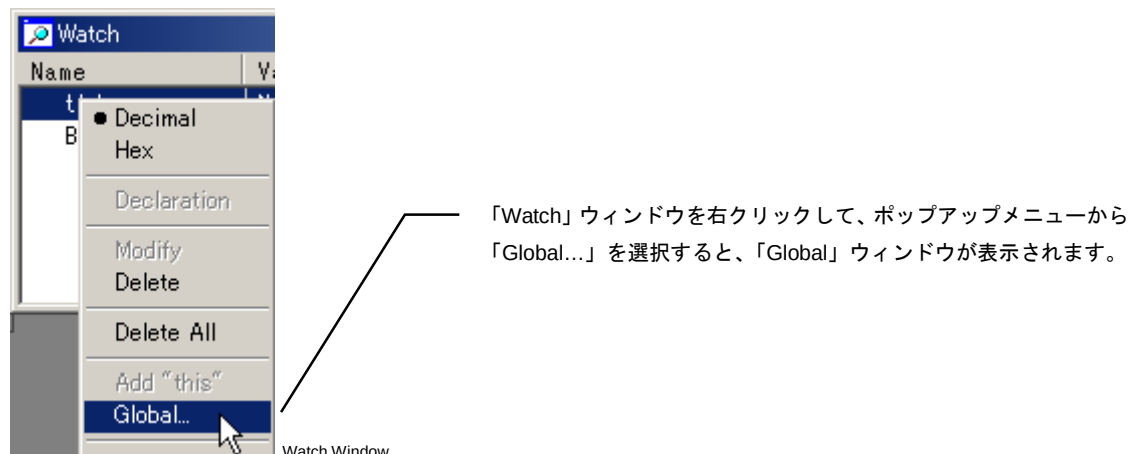
※3 : XrossFinder の「Watch」は「Call Stack」と連動しています。「Call Stack」ウィンドウでは関数の呼出し階層を表示していますが、ここでの関数階層の選択を変更すると「変数スコープ」を移動させることができます。たとえば、現在プログラムがある関数内で停止しているとすると、その関数を呼出しているひとつ上の階層のローカル変数を参照することもできます。CPU レジスタに割り付けられたローカル変数の値が退避されている場合、退避場所が「Location」の欄に(Saved at ...)と表示されます。

■ 変数名の登録

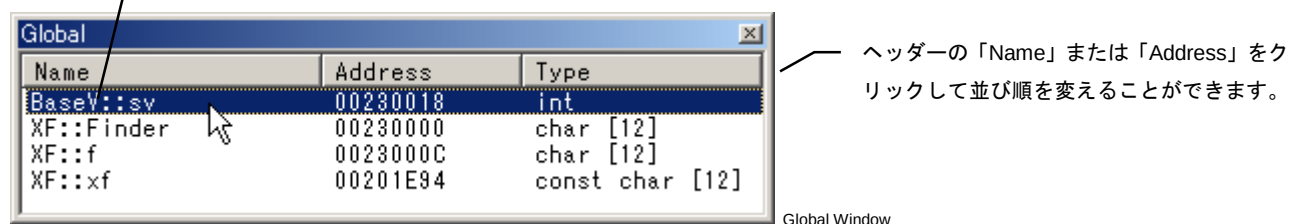
- ① 「Watch」ウィンドウへの変数名の登録は、「Source」ウィンドウから行います。



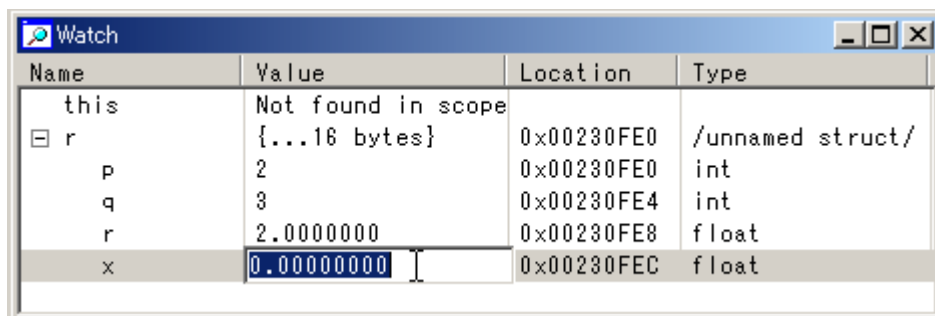
- ② グローバル変数(または静的変数)は、「Global」ウィンドウから登録できます。



登録したい変数の項目をダブルクリックするか、選択して Enter キーを押すと「Watch」ウィンドウに登録されます。
「Global」ウィンドウをキャンセルするには右上の [X] ボタンをクリックするか Escape キーを押します。



■ 変数値の変更



Watch Window

- ① 変更したい項目を選択し、値の表示部分をクリックするか、Enter キーを押すか、またはポップアップメニューの「Modify」を選択すると、入力可能状態になります。
- ② 入力は現在の表現形式か 16 進表現で行います。16 進の場合"0x"を先頭に付加します。
- ③ 値を確定するには Enter キーを押します。入力をキャンセルするには Escape キーを押します。

■ マウス操作

ボタン操作	位置	機能
左クリック	項目の[+]/[-]	ツリーの展開／縮小(折り畳み)
	項目	項目選択
	値表示部	値入力状態への移行
左ドラッグ	ヘッダーの境界	各欄（カラム）の幅の変更
左ダブルクリック	変数名	ツリーの展開／縮小(折り畳み)
左クリック 左ドラッグ	スクロールバー（縦）	表示領域のスクロール（縦）
右クリック	項目	項目選択および ポップアップメニューの表示（項目についての操作）
右クリック	項目以外の表示領域	ポップアップメニューの表示（項目についての操作）

■ キー操作

キー	機能
上・下矢印	項目選択の移動とスクロール
Ctrl + 上・下矢印	項目選択を移動しないスクロール
左矢印	ツリーの縮小(折り畳み)、または親項目への選択移動
右矢印	ツリーの展開、または最初の子項目への選択移動
Page Up/Down	上・下へのページスクロールと選択の移動
Ctrl+Page Up/Down	項目選択を移動しないページスクロール
Home	先頭項目への選択の移動とスクロール
End	最後尾項目へのカーソルの移動とスクロール
Enter	選択項目の値入力状態への移行
Escape (入力状態時)	入力のキャンセルとデータ入力状態の終了

■ ポップアップメニュー

項目名	機 能
Decimal	変数型に従った値表現への切り替え（選択項目およびその子項目）
Hex	16進表現への切り替え（選択項目およびその子項目）
Declaration	変数宣言の場所を「Source」ウィンドウに表示（親項目のみ）
Modify	選択項目の値入力状態への移行
Delete	選択項目の削除（親項目のみ）
Delete All	全項目の削除
Add "this"	"this"の登録
Global...	「Global」ウィンドウを表示
Font...	フォント設定ダイアログを表示

■ フォント設定ダイアログ

Windows 標準のフォント設定ダイアログです。

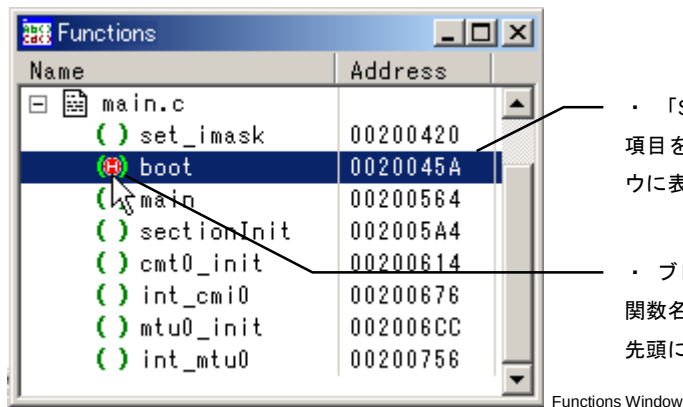
- ・ ダイアログを「OK」で終了すると、設定内容は「Watch」ウィンドウにただちに適用されます。

NOTE

※1：すべてのウィンドウに対し一括してフォント設定するには、「View」メニューの「All View Font...」を選択します。

4-4-6. 「Functions」 ウィンドウ

デバッグ情報に基づいた関数のリストを表示します。



・ 「Source」 ウィンドウの表示移動

項目を選択すると、対応するファイル／関数を「Source」ウィンドウに表示します。

・ ブレークポイントの設定

関数名の()括弧をクリックするか、Enter キーの入力により、関数の先頭にブレークポイントを設定できます。

■ 関数リストの表示

- ① 関数はファイル(コンパイル単位)ごとにグループ分けされツリー表示します。ここにはインクルードされたファイルにより定義された関数も含まれます。
- ② 関数がグローバル（他のコンパイル単位から参照可能）であれば、関数名の左の丸括弧は緑色で表示します。そうでない場合は紫色で表示します。
- ③ マウスがファイル名の上にあるとき、ファイルのフルパスをツールチップで表示します。
- ④ マウスが関数名の上にあるとき、グローバル名（C++での記述）をツールチップで表示します。

■ マウス操作

ボタン操作	位置	機能
左クリック	ファイルの[+]/[-]	ツリーの展開／縮小(折り畳み)
	関数の丸括弧アイコン	ブレークポイントの設定／削除
	項目	項目選択および対応するソースコードの表示
左ドラッグ	ヘッダーの境界	各欄（カラム）の幅の変更
左ダブルクリック	ファイル名	ツリーの展開／縮小(折り畳み)
左クリック 左ドラッグ	スクロールバー（縦）	表示領域のスクロール（縦）
右クリック	項目	項目選択および ポップアップメニューの表示（項目についての操作）
右クリック	項目以外の表示領域	ポップアップメニューの表示（項目についての操作）

■ キー操作

キー	機 能
上・下矢印	項目選択の移動とスクロール
Ctrl + 上・下矢印	項目選択を移動しないスクロール
左矢印	ツリーの縮小(折り畳み)、または親項目への選択移動
右矢印	ツリーの展開、または最初の子(関数)項目への選択移動
Page Up/Down	上・下へのページスクロールと選択の移動
Ctrl+Page Up/Down	項目選択を移動しないページスクロール
Home	先頭項目への選択の移動とスクロール
End	最後尾項目へのカーソルの移動とスクロール
Enter	選択関数先頭へのブレイクポイントの設定／削除
Delete	選択関数先頭に設定されたブレイクポイントの削除
Space	選択関数先頭に設定されたブレイクポイントの無効化／有効化
Tab	対応するソースコードの再表示

■ ポップアップメニュー

項目名	機 能
Break Set	選択関数先頭へのブレイクポイントの設定
Break Delete	選択関数先頭に設定されたブレイクポイントの削除
Break Enable	選択関数先頭に設定されたブレイクポイントの有効化
Break Disable	選択関数先頭に設定されたブレイクポイントの無効化
Font...	フォント設定ダイアログの表示

■ フォント設定ダイアログ

Windows 標準のフォント設定ダイアログです。

- ・ ダイアログを「OK」で終了すると、設定内容は「Functions」ウィンドウにただちに適用されます。

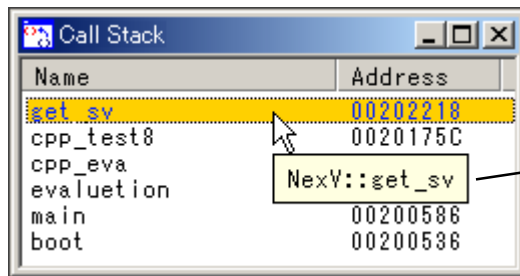
NOTE

※1: すべてのウィンドウに対し一括してフォント設定するには、「View」メニューの「All View Font...」を選択します。

4-4-7. 「Call Stack」 ウィンドウ

スタックの内容とデバッグ情報とに基づいた関数呼出し階層のリストを表示します。

■ 関数呼出し階層の表示



Call Stack Window

現在停止している関数名が一番上に表示され、以降は呼出し元の関数名が順に並べられます。

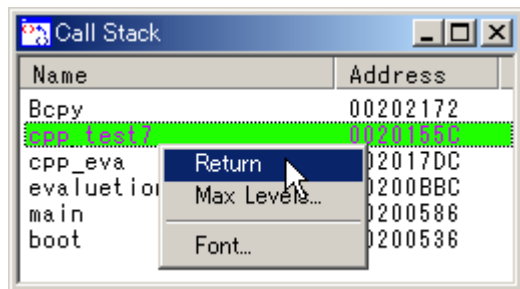
マウスに関数名の上に移動するとグローバル名 (C++でのスコープ名付き) をツールチップで表示します。

関数の表示アドレスは、一番上の項目が現在停止アドレス、それ以外は各関数への戻りアドレスです。

■ 関数呼出し階層の移動

- ① 関数名を選択することで階層の移動を行い、「Source」ウィンドウには、対応するソースコードが表示されます。
- ② 現在停止している関数を選択するとオレンジ色で表示され、その他は緑色で表示されます。
- ③ 関数名の選択を変更すると、「Watch」ウィンドウも連動し変数スコープの移動を行います。

■ 関数呼出し階層へのリターン

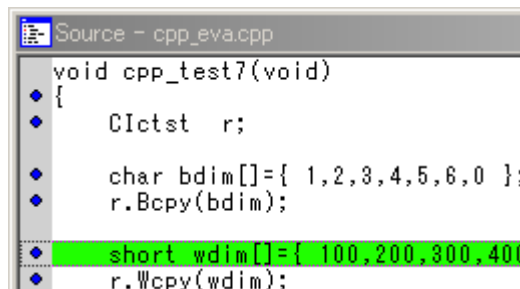


Call Stack Window

戻りたい位置の関数名を右クリックし、ポップアップメニューより「Return」を選択します。※¹



右クリックに連動し「Source」ウィンドウのソースコードが切り替わります。



Source Window

「Return」実行停止後は、オレンジ色に変化します。

NOTE

※¹ : 再起呼出しのように階層リストに同じ戻りアドレスが複数あると、指定した階層まで「Return」できない場合があります。

■ マウス操作

ボタン操作	位置	機能
左クリック	項目	項目選択および「Source」ウィンドウでの階層の表示
左ドラッグ	ヘッダーの境界	各欄（カラム）の幅の変更
左クリック 左ドラッグ	スクロールバー（縦・横）	表示領域のスクロール（縦・横）
右クリック	項目以外	ポップアップメニューの表示
	項目	ポップアップメニューの表示（項目についての操作）

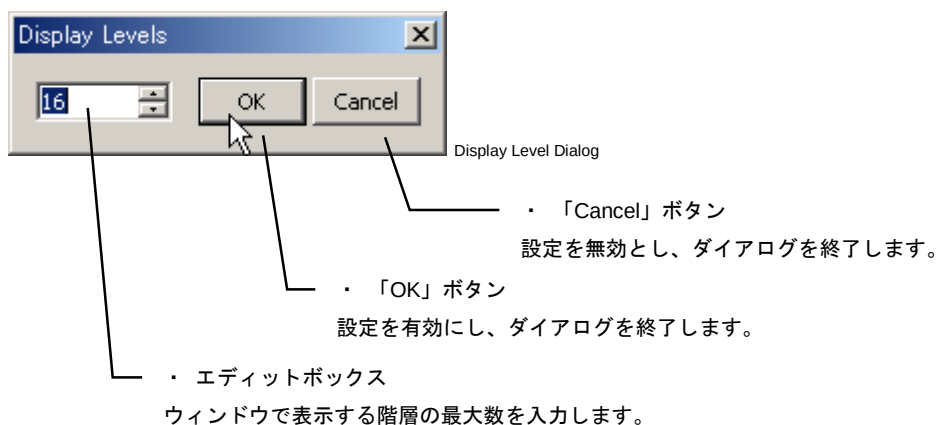
■ キー操作

キー	機能
上・下矢印	項目選択の移動とスクロール
左・右矢印	表示領域の左・右へのスクロール
Ctrl + 左・右矢印	表示領域の左・右への最大スクロール
Page Up/Down	上・下へのページスクロールと選択の移動
Home	先頭項目への選択の移動とスクロール
End	最後尾項目へのカーソルの移動とスクロール
Tab	「Source」ウィンドウでの階層の再表示

■ ポップアップメニュー

項目名	機能
Return	選択階層までの「Step Out」
Max Levels	「Display Levels」ダイアログの表示
Font...	フォント設定ダイアログの表示

■ 「Display Levels」ダイアログ



■ フォント設定ダイアログ

Windows 標準のフォント設定ダイアログです。

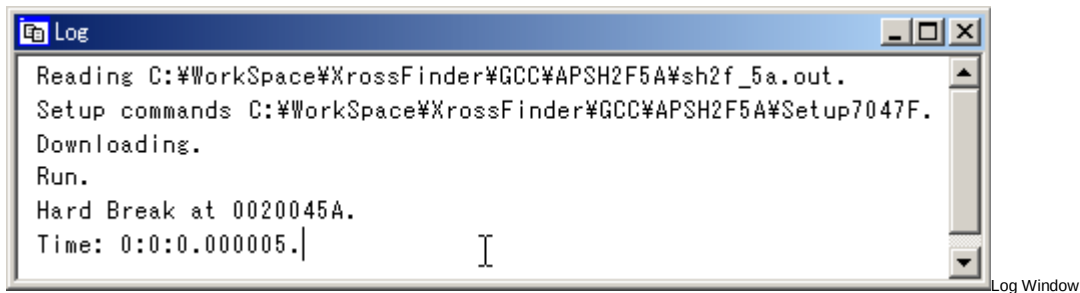
- ・ ダイアログを「OK」で終了すると、設定内容は「Call Stack」ウィンドウにただちに適用されます。

NOTE

※1: すべてのウィンドウに対し一括してフォント設定するには、「View」メニューの「All View Font...」を選択します。

4-4-8. 「Log」ウィンドウ

ユーザプログラムの処理内容や結果を表示します。



■ 処理内容の表示

- ① ユーザプログラムの読み込み、転送、実行などの処理内容や結果を表示します。
- ② 実時間実行の場合には実行時間を表示します。表示形式は以下の通りです。

<表現形式>

Time : H : M : S

<記号の意味>

H = 時、D = 分、S = 秒 — 小数点以下6桁まで表示 (μ Sec 単位)

NOTE

※1: 実行時間は XrossFinder 本体内のタイマーで計測していますので、必ずしも正確ではありません。

■ マウス操作

ボタン操作	位置	機能
左クリック	表示領域	編集カーソル（キャレット）の移動
右クリック	表示領域	ポップアップメニューの表示
左ドラッグ	表示領域	表示内容のテキスト選択
左ダブルクリック	表示箇所	単語のテキスト選択

■ キー操作

キー	機能
上・下矢印	編集カーソル（キャレット）の上・下移動とスクロール
左・右矢印	編集カーソル（キャレット）の上・下移動とスクロール
Ctrl + 左・右矢印	単語単位での編集カーソル（キャレット）の移動
Shift + 矢印	テキスト選択
Ctrl + A	全表示内容のテキスト選択
Ctrl + C	テキスト選択範囲のクリップボードへのコピー (他のアプリへ貼り付けができます)

■ ポップアップメニュー

項目名	機 能
Copy	テキスト選択範囲のクリップボードへのコピー (他のアプリへ貼り付けができます)
Clear	表示内容の全消去
Font...	フォント設定ダイアログを表示

■ フォント設定ダイアログ

Windows 標準のフォント設定ダイアログです。

- ・ダイアログを「OK」で終了すると、設定内容は「Log」ウィンドウにただちに適用されます。

NOTE

※1: すべてのウィンドウに対し一括してフォント設定するには、「View」メニューの「All View Font...」を選択します。

4-4-9. エラーメッセージ

ウィンドウやダイアログに表示されるエラーメッセージを記載します。

エラーメッセージ	内 容 / 処 置
All Hard Breaks are in using	すべてのハードウェア・ブレークが使用されています。 他のハードウェア・ブレークを無効にするか削除してください。
All Hard Breaks which this setting is possible are in using	CPU の種類によっては、ハードウェア・ブレークの Instruction/Operand Access 設定が制限されます。同じ設定を使用している他のハードウェア・ブレークを無効にするか削除して下さい。
Break already exist	ブレークポイントは既に登録されています。
Command File Error at line xxx	コマンドファイルの記述に誤りがあります。 エラー表示された行の記述を修正して下さい。
Command File was not read	コマンドファイルは読み込まれませんでした。
CPU not supported	ターゲット CPU はサポートされていません。
Debug Target Error	ターゲットでエラーが発生しました。 ターゲットの電源/Dip SW、H-UDI ケーブル接続を再確認して下さい。
Debug Target Time Out	ターゲットの接続でタイムアウトが発生しました。 ターゲットの電源/Dip SW、H-UDI ケーブル接続を再確認して下さい。
Endianness Mismatched	プログラムファイルと「System Setup」のインディアン設定が一致しません。 プログラムファイルと「System Setup」のインディアンを一致させて下さい。
File was not found	指定したファイルが見つかりません。
File Read Error	指定したファイルの読み込みに失敗しました。
File Write Error	指定したファイルの書き込みに失敗しました。
Illegal Address Value	指定したアドレスは無効です。
Illegal Value	指定した値は無効です。
Illegal Value Expression	指定した表記は無効です。
Initial File Format Error	初期設定ファイルの記述に誤りがあります。
Initial File not found	初期設定ファイルが見つかりません。
Initial File Parameter Error	初期設定ファイルのパラメータが間違っています。
Initial File Read Error	初期設定ファイルの読み込みに失敗しました。
Invalid Address	変数アドレスが無効です
License Key is incorrect	ライセンスキーが間違っています。
Linkage Error	ターゲットとの接続に失敗しました。 ターゲットの電源/Dip SW、H-UDI ケーブル接続を再確認して下さい。
Linkage Time Out	ターゲットの接続でタイムアウトが発生しました。 ターゲットの電源/Dip SW、H-UDI ケーブル接続を再確認して下さい。
Linkage Version not matched	ソフトウェアのバージョンが異なっています。 最新版をダウンロードしバージョンアップをして下さい。
Memory Empty	メモリの確保に失敗しました。 他のアプリケーションを終了してください。
Not Available	変数情報が使用できません
Not found in scope	変数情報がスコープ内に見つかりません
Not Supported	デバッグ情報形式などがサポートされていません
Processing was canceled	処理は中断されました。
Soft Break can not be set	ソフトウェアブレークポイントの設定が出来ません。 ROM 上でソフトウェア・ブレークは使用できません。
Soft Break has no vacancy	ソフトウェアブレークポイントの空きがありません。 他のソフトウェアブレークポイントを削除して下さい。

Target Flash Memory Erase Error	ターゲットのフラッシュメモリの消去に失敗しました。
Target has been reset or powered off	ターゲットの電源が入っていないかリセットがかかっています。 ターゲットボードを再確認して下さい。
Target Moinitor Version not matched	ソフトウェアのバージョンが異なります。 最新版をダウンロードしバージョンアップをして下さい。
The Key Code was not matched and the On-Chip Flash Memory was erased	ターゲットCPUと「System Setup」のキーコードが一致しません。 内蔵フラッシュメモリは強制的に消去されました。
This application program is not licensed	ターゲットCPUの使用ライセンスが登録されていません。 ライセンスを登録して下さい。
Two or more USB Drivers exist	2つ以上のXrossFinderが接続されています。 接続は1台のみとして下さい。
Unknown Error	予期せぬエラー。
USB Driver was not found	USBポートのオープンに失敗しました。 USBケーブルの接続、USBドライバの登録を確認して下さい。
USB Port Open Error	USBポートのオープンに失敗しました。 USBケーブルの接続、USBドライバの登録、または他のアプリケーションが ドライバ使用中か確認して下さい。
USB Receive Error	USBポートからの受信に失敗しました。 USBケーブルの接続、USBドライバの登録を確認して下さい。
USB Transmit Error	USBポートへの送信に失敗しました。 USBケーブルの接続、USBドライバの登録を確認して下さい。
Using Key Code Value 0xFFFFFFFF erases the On-Chip Flash Memory	「System Setup」のキーコードに0xFFFFFFFFを使用すると内蔵フラッシュ が消去されます。
Value Undefined	変数情報が未定義です。
Verify Error	ベリファイに失敗しました。
Watch already exist	ウォッチに登録済みです。

5. 技術資料

5-1. 外部トリガ入力回路

外部トリガ入力は、ターゲットボードからの入力信号のエッジをトリガとして、デバッグ中のユーザプログラムの実行を強制停止させたい場合に使用します。

XrossFinder 本体の外部トリガ入力回路を図 5-1 に記載致します。

入力信号は 3.3-5V の範囲内として下さい。

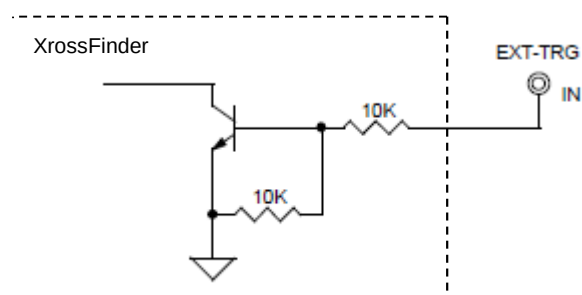


図 5-1 外部トリガ入力回路

5-2. リセット出力回路

XrossFinder 本体のリセット出力回路を図 5-2 に記載致します。

出力はオープンコレクタとなっていますのでターゲットボード側でプルアップ等の処理をして下さい。

また、ターゲットボードのリセット回路出力もオープンコレクタ相当である必要があります。

ターゲットボードの予期せぬ動作を防ぐため接続することをお勧め致します。

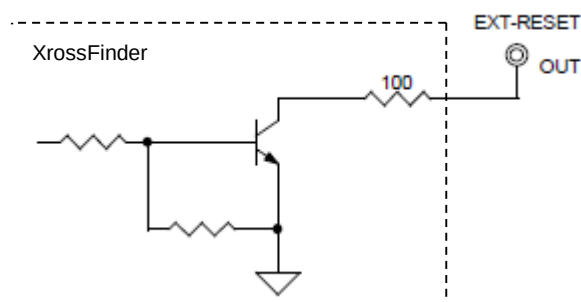


図 5-2 リセット出力回路

5-3. H-UDI 接続参考図

H-UDI 接続参考図につきましては、アプリケーションノート「AN802 H-UDI コネクタ接続例」を参照していただけますようお願いいたします。

5-4. 機能・仕様

デバグ機能	
実行	実行/停止/ステップイン/ステップオーバー/ステップリターン/カーソル位置まで実行 停止時に実行時間表示(μS単位)
ブレークポイント	ソフトウェアブレークポイント 256 点 ハードウェアブレークポイント 1~9 点 (CPU 依存)
リセット	ソフトウェア CPU リセット ハードウェアリセット (RESET-OUT 端子出力)
外部トリガ	EXT-TRG 端子の信号によりユーザプログラムの停止
メモリ	メモリ内容の表示、編集
ソース表示	C/アセンブリ/C++アセンブリ混在表示
レジスタ	CPU レジスタの表示、編集
ウォッチ	指定されたメモリ領域のウォッチ、編集
関数リスト	関数リストのツリー表示
関数呼出	関数の呼出階層の表示
コントロールソフト動作環境	
パーソナルコンピュータ	PC/AT 互換機
OS	Windows 2000/XP
メモリ	256M バイト以上推奨
ハードディスク	20M バイト以上の空き領域
CD ドライブ	CD-ROM 読み込み可能なドライブ (インストール時のみ)
表示	800x600 以上
USB ポート	USB1.1 または 2.0
対応コンパイラ	ルネサス社製「SuperH RISC engine C/C++コンパイラ」(バージョン 6 以降) GNU-C/C++コンパイラ (KPIT GNUSH 推奨) 上記コンパイラの ELF/DWARF 形式にのみ対応
本体仕様	
ホスト PC I/F	USB1.1 インターフェース
ターゲット I/F	H-UDI (3.3V または 5V)
電源	USB バスパワーにて供給 (5V)
消費電量	約 80mA
重量	約 100g
外形寸法	60mm (W) × 120mm (D) × 30mm (H)
使用環境	0~40℃ (結露無き事)

5-5. 注意事項

- ・ XrossFinder 本体は、1 台のホストコンピュータに対し複数台の接続はできません。
- ・ コントロールソフトの多重起動はできません。
- ・ 本体の電源を投入したまま、ターゲットボードを取り替えないで下さい。
- ・ ターゲットボードの電源を入れた状態で、本体の電源を投入(USB コネクタの接続)しないで下さい。

6. サポートのご案内

●製品サポート

製品サポートは別紙サポート規定に準じます。万が一、不具合や問題を発見された場合には、お手数ですが弊社サポートまでご連絡下さい。

以下の内容に該当するお問い合わせにつきましては受け付けておりませんのであらかじめご了承ください。

- 本製品の回路動作に関するご質問
- ユーザ回路の設計方法やその動作についてのご質問
- 関連ツールの操作指導
- その他、製品の仕様範囲外の質問やお客様の技術によって解決されるべき問題

●弊社ホームページのご利用について

お客様にお役立ていただける情報を弊社ページに掲載しておりますので、是非ご利用下さい。

弊社ホームページアドレス <http://www.apnet.co.jp>

エンジニアリングサービスのご案内

弊社製品をベースとしたカスタム品やシステム開発を承っております。
お客様の仕様に合わせて、設計から OEM 供給まで一貫したサービスを提供いたします。
詳しくは、弊社営業窓口までお問い合わせください。

営業案内窓口

■ TEL	053-401-0033 (代表)
■ E-MAIL	sales@apnet.co.jp

改定履歴

版 数	日 付	改 定 内 容
1.0	2005/11/10	・ 初版
2.0	2006/03/27	・ 対応 CPU の追加 (1.3 章) SH7083F,SH7084F,SH7085F,SH7086F,SH7124F,SH7125F,SH7146F,SH7149F,SH7727 ・ 「System Clock」: CPU への入力クロック周波数の設定を追加 (3-4-1,4-2-2 章) ・ 「Breakpoints」 ウィンドウの Type 表示の変更。 HardBreak の設定時、Instruction/Operand の Access 設定を表示
3.0	2006/07	・ 対応 CPU の追加 (1.3 章) SH7780,SH7211,SH7206
4.0	2006/09	・ 対応 CPU の追加 (1.3 章) SH7760 ・ CPU 接続例の追加 (5.3 章)
5.0	2007/03	・ 対応 CPU の追加 (1.3 章) SH7706,SH7619, SH7125 デバッグ MCU ボード, SH7083,SH7084,SH7085,SH7086 の量産版 ・ CPU 接続例の追加/修正 (5.3 章) ・ 「Sytem setup」 ダイアログに「SH7080 シリーズ」についてを追加 (4.2 章)
5.1	2007/04	・ コマンドファイルのコマンド形式にウエイト機能追加(3.2.2 章)
5.2	2007/09	・ 対応 CPU の追加 (1.3 章) SH7201、SH7203、SH7261、SH7263、SH7730 ・ コントロールソフトのインストール手順修正 (2.1 章)
5.3	2008/01	・ 対応 CPU の追加 (1.3 章) SH7606、SH7785 ・ 「デバッグ情報の形式」の GNU-GCC コンパイラに*.x、*.elf 拡張子を追加 (3.1 章) ・ CPU 接続例の追加/修正 (5.3 章)
5.4	2008/06	・ 対応 CPU の追加 (1.3 章) SH7286、SH7764 ・ CPU 接続例の追加/修正 (5.3 章)
5.5	2008/11	・ 対応 CPU の追加 (1.3 章) SH7136、SH7137、SH7670、SH7671、SH7751R、SH7763 ・ CPU 接続例の追加/修正 (5.3 章) ・ 対応 OS を追加 (1.5 章)
5.6	2009/02	・ 対応 CPU の追加 (1.3 章) SH7146 A マスク品、SH7147 A マスク品、SH7205、SH7265 ・ デュアルコア対応のための修正 ・ H-U-D-I コネクタ接続例をアプリケーションノートとして提供するため削除 (5.2 章)
5.7	2009/06	・ 対応 CPU の追加 (1.3 章) SH7243、SH7264、SH7285
5.8	2009/11	・ 対応 CPU の追加 (1.3 章) SH7216、SH7705
5.9	2010/04	・ 対応 CPU の追加 (1.3 章) SH7262、SH7710 ・ メモリのファイル保存、ファイル読み込み追加 (3.15 章)
6.0	2010/07	・ 対応 CPU の追加 (1.3 章) SH7712、SH7450、SH7451

本文書について

- ・本文書の著作権は株式会社アルファプロジェクトが保有します。
- ・本文書の内容を無断で転載することは一切禁止します。
- ・本文書の内容は、将来予告なしに変更されることがあります。
- ・本文書の内容については、万全を期して作成いたしましたが、万一ご不審な点、誤りなどお気付きの点がありましたら弊社までご連絡下さい。
- ・本文書の内容に基づき、アプリケーションを運用した結果、万一損害が発生しても、弊社では一切責任を負いませんのでご了承下さい。

商標について

- ・Windows®の正式名称は Microsoft®Windows®Operating System です。
Microsoft、Windows は、米国 Microsoft Corporation.の米国およびその他の国における商標または登録商標です。
Windows®Vista、Windows®XP、Windows®2000 Professiona は、米国 Microsoft Corporation.の商品名称です。
本文書では下記のように省略して記載している場合がございます。ご了承ください。
Windows®Vista は Windows Vista もしくは WinVista
Windows®XP は Windows XP もしくは WinXP
Windows®2000 Professional は Windows 2000 もしくは Win2000
- ・SuperH は、ルネサスエレクトロニクス株式会社の登録商標、商標または商品名称です。
- ・その他の会社名、製品名は、各社の登録商標または商標です。



株式会社アルファプロジェクト
〒431-3114
静岡県浜松市東区積志町 834
<http://www.apnet.co.jp>
E-MAIL : query@apnet.co.jp