

PARTNER ユーザーズ・マニュアル  
VRシリーズ  
『VR5400-TP個別編』

## ■ PARTNERのバージョンアップ

- 最新のPARTNER-VR5400-TPは、以下のサイトよりダウンロードできます。

<http://www.midas.co.jp/products/download/program/partner.htm>

## ■ ご注意

- PARTNER-VR5400-TP(プログラム及びマニュアル)に関する著作権は株式会社マイダス・ラボ及び京都マイクロコンピュータ株式会社が所有します。
- 本プログラム及びマニュアルは著作権法で保護されており、弊社の文書による許可が無い限り複製、転載、改変等できません。
- お客様に設定される使用権は、PARTNER-VR5400-TP用をお客様が登録する1台のシステムにおいてのみ使用する権利を意味します。
- 本製品は、万全の注意を持って作製されていますが、ご利用になった結果については、販売会社、株式会社マイダス・ラボ及び京都マイクロコンピュータ株式会社は一切の責任を負いかねますのでご了承ください。
- 本プログラム及びマニュアルに記載されている事柄は、予告なく変更されることがあります。
- 本プログラムの変更により、このマニュアルに記載している画面のイメージと実際の画面が異なる場合があります。

## ■ 商標について

- MS-Windows、Windows、MS、MS-DOSは米国マイクロソフト・コーポレーションの商標です。
- そのほか本書で取り上げるプログラム名、システム名、CPU名などは、一般に各メーカーの商標です。

改訂履歴

日付 Y / M / D	Rev	内 容
1999.05.17	1.00	初版
1999.12.08	1.10	エミュレーションROMダウグ、ROMコマンドのサイズに32M～256M追加 デバッグ情報領域の単位、省略値変更
2000.01.13	1.11	誤記訂正(P39)
2000.03.06	1.20	『8 フラッシュメモリのサポート』追加 RTE-1000-TP対応 ヒストリウンドウ、ヒストリメニュー、CPU環境設定ダウグ、ENVコマンド、TDコマンド 変更、タイムタグの追加 マスクテーラの分離 イペント設定ダウグ、EVコマンド変更 トレースモード設定にトレース有効/無効を追加
2001.05.20	1.21	download site 情報の変更
2002.01.10	1.30	RTE-2000-TP対応の追加 エミュレーションROMダウグ、ROMコマンドの変更 ROM1～ROM4コマンド追加
2002.11.20	1.31	以下の項を追加 7.1.3 MULTI2000での作成 1.1 PARTNER-VR5400-TPの環境設定-RTEオプション
2003.05.16	1.32	フラッシュメモリの対応デバイスにIntelタイプを追加 フラッシュメモリの環境設定に『領域設定書式2(FLASH_MEM2)』を追加

空白ページ

# 目次

はじめに ..... 1

## 1 起動 ..... 3

1.1 PARTNER-VR5400-TPの環境設定 ..... 4

1.2 初期設定コマンド ..... 8

1.3 起動時のエラーメッセージ ..... 10

## 2 メインウィンドウ補足 ..... 13

2.1 メニュー ..... 14

2.1.1 実行メニュー ..... 14

## 3 チャイルドウィンドウ ..... 15

3.1 レジスタウインドウ ..... 16

3.1.1 レジスタウインドウのショートカットキー ..... 17

3.1.2 レジスタウインドウのローカルメニュー ..... 18

3.1.3 レジスタウインドウでのマウス操作 ..... 19

3.2 ブレークウインドウ ..... 20

3.2.1 ブレークウインドウのショートカットキー ..... 20

3.2.2 ブレークウインドウのローカルメニュー ..... 21

3.2.3 ブレークウインドウでのマウス操作 ..... 21

3.3 ヒストリウインドウ ..... 22

3.3.1 ヒストリウインドウのショートカットキー ..... 23

3.3.2 ヒストリウインドウのローカルメニュー ..... 23

## 4 ダイアログコマンド ..... 26

## 目次

---

4.1 ダイアログボックス.....	27
4.1.1 トレースモード設定ダイアログボックス .....	27
4.1.2 イベント設定ダイアログボックス .....	28
4.1.3 CPU環境設定ダイアログボックス .....	29
4.1.4 エミュレーションROM設定ダイアログ .....	30
4.1.5 イベントブレークの設定ダイアログボックス .....	31
4.1.6 TLBの設定ダイアログボックス .....	31
4.1.7 キャッシュフラッシュダイアログボックス .....	32
4.1.8 各種状態の設定ダイアログボックス .....	32
<b>5 リアルタイムトレース .....</b>	<b>34</b>
5.1 リアルタイムトレースの概要 .....	35
5.1.1 ディレイカウント .....	36
5.1.2 トレースの開始 .....	36
5.1.3 トリガ条件 .....	37
5.1.4 トレースの終了 .....	37
5.1.5 強制ディレイモード .....	38
5.2 タイムタグについて .....	39
5.2.1 タイムタグ表示の種類 .....	39
5.2.2 タイムタグの注意 .....	39
<b>6 コマンドリファレンス .....</b>	<b>40</b>
6.1 コマンド解説の規約 .....	41
6.2 機能別コマンド .....	42
6.2.1 ハードウェアの初期化 .....	42
6.2.2 CPUリセット .....	42
6.2.3 環境設定コマンド .....	42

6.2.4 プログラムロード .....	43
6.2.5 ファイルリード/ライト .....	43
6.2.6 プログラム実行 .....	44
6.2.7 ブレークポイント .....	44
6.2.8 イベントブレークポイント .....	44
6.2.9 コード表示 .....	45
6.2.10 レジスタ表示/変更 .....	45
6.2.11 システムレジスタ表示/変更 .....	45
6.2.12 メモリ表示/変更 .....	46
6.2.13 I/Oポート入力/出力 .....	46
6.2.14 シンボル表示/設定 .....	46
6.2.15 アセンブル .....	46
6.2.16 バックトレース .....	46
6.2.17 リアルタイムトレース .....	47
6.2.18 キャッシュフラッシュ/初期化 .....	47
6.2.19 Cのデータ参照/変更 .....	48
6.2.20 文字列の定義(Cの式評価) .....	48
6.2.21 システム制御 .....	48
6.2.22 システムコール .....	49
6.2.23 式の表示 .....	49
6.2.24 基数変更 .....	49
6.2.25 ロギング/バッチ .....	49
6.2.26 オプション設定 .....	50
6.2.27 マクロコマンド .....	50
6.2.28 フラッシュメモリ管理 .....	51
<b>7 高級言語レベルデバッグ .....</b>	<b>52</b>

## 目次

---

<b>7.1 GreenHills C.....</b>	<b>53</b>
7.1.1 コンパイルとリンク方法.....	53
7.1.2 使用例.....	53
7.1.3 MULTI2000での作成 .....	54
<b>7.2 GNU C(exeGCC) .....</b>	<b>56</b>
7.2.1 コンパイルとリンク方法.....	56
7.2.2 使用例.....	56
<b>7.3 CodeWarrior C.....</b>	<b>58</b>
7.3.1 コンパイルとリンク方法.....	58
7.3.2 使用例.....	59
<b>8 フラッシュメモリのサポート .....</b>	<b>62</b>
8.1 対応デバイス.....	63
8.2 環境設定 .....	65
8.3 書き込み動作.....	69
8.4 管理コマンド(ZF).....	71
<b>付録 .....</b>	<b>72</b>
レジスタ変数.....	73
エラーメッセージ .....	74

# はじめに

このマニュアルは、『PARTNER-VR5400-TP』に依存した部分の操作方法について説明しています。

その他の操作方法は『PARTNER ユーザーズ・マニュアルVRシリーズ共通編』で説明されていますので合わせて参照してください。

なお、特に説明がない場合は、RTE-200-TP、RTE-1000-TP、RTE-2000-TPで共通です。

## 必要なハードウェア

- インターフェースキット

パーソナルコンピュータとRTE-200-TP、RTE-1000-TPまたはRTE-2000-TPを接続するためのインターフェースキットが必要です。インターフェースキットには、『PC Cardインターフェースキット』、『PC98 Desk top PC用インターフェースキット』、『DOS/V Desk Top PC用インターフェースキット』、『PCIバス用インターフェースキット』、『10Base-TのLAN-BOX』等があります。

## 必要なセットアップ

- RTE for Win32

RTE for Win32のインストールマニュアルを参照してDLLのインストールを行い、使用するICEの種類と接続ポートを設定し、ハードウェアの接続確認を実施してください。

空白ページ

# 1 起動

PARTNER-VR5400-TPを起動するには、PARTNER-VR5400-TPの環境を設定する必要があります。

この章では、PARTNER-VR5400-TPの起動に関するオプションの設定方法を説明します。

## 1.1 PARTNER-VR5400-TPの環境設定

### 起動オプションの設定



PARTNER-VR5400-TPの起動オプションを指定します。起動オプションでは、動作モードやメモリの使用状態を変更します。(図 1)



図 1 [起動オプション]ダイアログボックス

オプション文字列の個所には、ダイアログで入力した内容が文字列で表示されます。PARTNER-VR5400-TPの起動オプションは通常、RPTSETUPの[起動オプション]ダイアログボックスで指定します。指定された起動オプションはプロジェクトファイル(RPTVR5400T.KPJ)に書き込まれます。PARTNER-VR5400-TPの起動オプションを以下に示します。

#### -Bオプション

デバッグ情報およびマクロ領域のサイズ指定を行います。デバッグ情報量枠にはグローバル,ローカルシンボル,行番号情報,関数/変数の属性情報などの多くの情報が格納されます。

#### **-B[サイズD][,サイズM]**

**サイズ D** デバッグ情報を登録する領域のサイズ(1K バイト単位)  
省略時には 1024K バイトの領域を確保します。

**サイズ M** マクロ定義情報を保存する領域のサイズ(1K バイト単位)

省略時には 32K バイトの領域を確保します。

#### -Dオプション

PARTNER-VR5400-TPのカレントディレクトリを指定します。カレントディレクトリにはコンフィギュレーションファイル(RPTVR5400T.CFG), プロジェクトファイル(RPTVR5400T.KPJ)が存在する必要があります。

#### **-Dディレクトリ**

ディレクトリがPARTNER-VR5400-TP起動時のカレントディレクトリになります。このオプションが省略された場合は[プロパティ]ダイアログボックスの[作業フォルダ]に指定されているディレクトリ、[コマンドライン]に指定されている実行プログラムの存在するディレクトリの順にディレクトリが指定されます。

#### -Eオプション

コードウインドウで表示/参照するソースファイルの拡張子のデフォルトを追加指定します。

#### **-E拡張子**

拡張子がソースファイルのデフォルトの拡張子となります。拡張子にはピリオドを除く拡張子部分を指定します。このオプションが省略された場合はfile.Cまたはfile.ASMがデフォルトの指定になります。拡張子指定はセミコロン(;)で区切ることにより行えます。

#### -SDオプション

コードウインドウに参照するソースファイルのあるディレクトリを指定します。ディレクトリ指定はセミコロン(;)で区切ることにより複数指定できます。

#### **-SDディレクトリ**

ディレクトリがソースファイルのあるディレクトリになります。このオプションが省略された場合は-Dオプションで指定したディレクトリになります。

#### -TABオプション

コードウインドウのソース表示をするときのタブのサイズを指定します。エディタでタブサイズを変更したファイルを表示するのに便利な機能です。

#### **-TABタブサイズ**

タブサイズで指定される数にタブを合わせます。このオプションを省略した場合のタブサイズは8に設定されます。

### -Xオプション

アセンブラーまたは各種Cのデバッグモードの指定をします。PARTNER-VR5400-TPでは多くの処理系をサポートしているために各処理系を明確にする必要があります。この指定を正しく行っていない場合には、正常なデバッグができないかあるいは特定の機能が使用できないことがあります。各処理系の使用方法は、『7 高級言語レベルデバッグ』、『各処理系のマニュアル』を参照してください。

#### **-X**

アセンブラーのデバッグモード(デフォルト)の指定です。このモードでPARTNER-VR5400-TPを起動するとC関連(スタックのバケットレースやローカル変数などに関連した)コマンドは利用できません。

#### **-XC**

Green Hills のデバッグモード指定をします。デバッグ情報は.OUTファイルから読み込みます。

#### **-XG**

exeGCC のデバッグモード指定をします。デバッグ情報は.OUTファイルから読み込みます。

#### **-XCW**

Code Warrior Cのデバッグモード指定をします。デバッグ情報は.OUTファイルから読み込みます。

### -RTEオプション

使用するRTEのチャンネル番号を指定します。

#### **-RTEチャンネル番号**

チャンネル番号には0～3を指定します。RTEのチャンネル機能を使用しない場合は0を指定してください。

### @オプション

-Dオプション以外のオプションを@で指定したファイルから読み込むことが可能です。

#### **【例】**

コマンドライン

C:¥RPTVR54xx¥BIN¥RPTV54xx -DC:¥SAMPLE @TEST. INF

ファイル内容(TEST.INF)

-B1024 -SDC:¥SAMPLE¥SOURCE -XC

## 1.2 初期設定コマンド

コンフィギュレーションファイル(RPTVR5400T.CFG)の内容以外で、RTE-VR5400-TPの動作環境を予め設定しておく必要がある場合は、次のコマンドで初期化することができます。

各コマンドについては、コマンドリファレンス(34頁参照)、または、ハードウェア・ユーザーズ・マニュアルを参照してください。

### 初期化で使用するコマンド

INIT	PARTNER-VR5400-TPおよびICEの初期化をします。
ENV	各種入力信号のマスクなどのCPUの各種環境を設定します。
ROM	エミュレーションROMを設定します(RTE-1000-TP用)。
ROM1 ~ ROM4	エミュレーションROMを設定します(RTE-2000-TP用)。
NC	非メモリキャッシュの領域を設定します。
NCD	非メモリキャッシュの領域を解除します。
NROM	エミュレーションメモリ割付除外領域を設定します。
NROMD	エミュレーションメモリ割付除外領域を解除します。

これらのコマンドはコマンドウィンドウで入力しますが、PARTNER-VR5400-TPの起動時に自動実行されるINIT.MCRファイルに記述しておくと便利です。

## INIT.MCRでの初期化例

INIT.MCRファイルに、以下の例のような初期化コマンドを記述しておくことで、起動毎の初期化を自動で行うことができます。

```
init
    PARTNER-VR5400-TP、ICEを初期化します。
env !nmi
    NMI信号をマスクする、に設定します。
rom BFC00000 80000 4m rom16 bus16
    エミュレーションするROMを16ビットROM、16ビットバス、4Mビット
    サイズに、そしてエミュレーションアドレス範囲をBFC00000H ~
    BFC7FFFFH番地に設定します。
nc 20000 1000
    20000番地から1000バイトの領域をメモリキャッシュの除外エリアに
    指定します。
L sample.out
    デバッグを行うプログラムsample.exeを読み込みます。
```

ROMコマンドはRTE-1000-TPの例です。

## 1.3 起動時のエラーメッセージ

PARTNER-VR5400-TPが起動時にエラーを検出した際に表示するメッセージを示します。エラー要因を取り除いてから再起動してください。



RTEの電源が入っていないか、ハードウェアの不良が考えられます。電源、接続ケーブル等を確認してください。その後、ChkRTEプログラムを実行してください。



他のアプリケーションがRTEを使用しています。または、異常終了によりRTEのコントロールプログラムは終了していません。その場合にはWindowsを再起動してください。



初期化中のRTEから応答がありません。ターゲットシステムが接続されている場合、プロセッサが動作できない可能性があります。障害を取り除いてから起動してください。



ホストカードまたはPCカードがインストールされていない可能性があります。正しくインストールされていることを確認しChkRTEプログラムを実行してください。



RTEが接続されていないか、RTEの電源が入っていないことが考えられます。電源、接続ケーブルなどを確認してください。その後、ChkRTEプログラムを実行してください。



システムの初期化ができません。RTEに電源が入っていない可能性があります。電源の接続を確認し、ChkRTEプログラムを実行してください。



何らかの異常終了によりRTEのコントロールプログラムは終了していません。Windowsを再起動するか、RTEのコントロールプログラムを強制終了してください。



RTEインストールに不備があり必要なDLLが見つかりません。RTEのSetUp CD-ROMで再インストールしてください。その後、必ずChkRTEプログラムを実行してください。



## 2 メインウインドウ補足

この章では、共通編と一部異なる個所を説明します。

## 2.1 メニュー

### 2.1.1 実行メニュー

プログラム実行(G)	F5
カム(C)	F7
トレース(I)	F8
ステップ(S)	F10
リターン実行(U)	Ctrl+F5
ブレークポイント(P)	F9
強制ブレーク(B)	ESC
キャッシュフラッシュ(F)...	
エミュレーションROM設定(R)...	
CPU環境設定(V)...	

[実行]メニューにはユーザプログラムの実行 / 停止に関するコマンドが収められています。

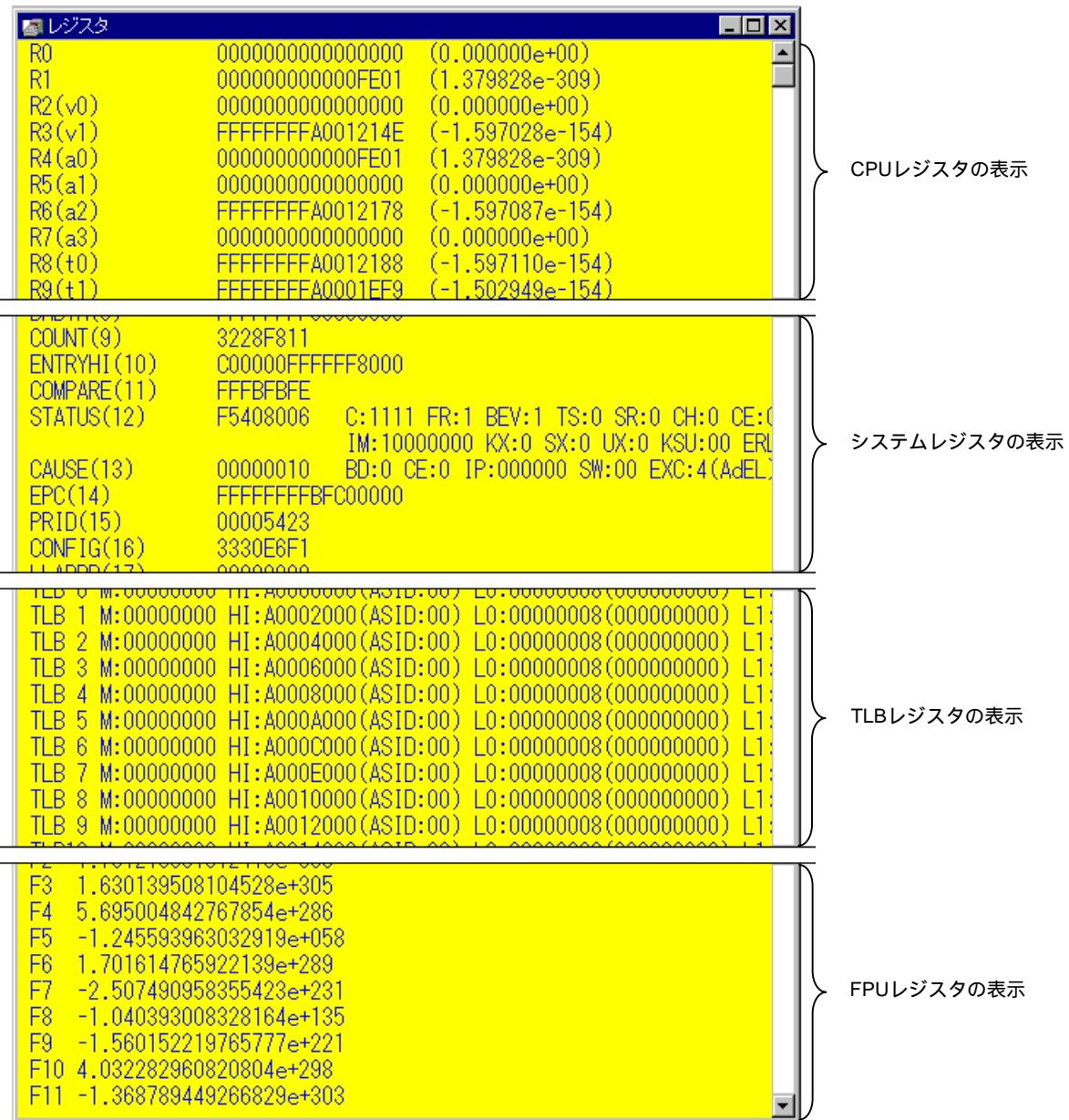
メニュー項目	機能
プログラム実行(G)	ユーザプログラムを現在のPCから実行します
カム(C)	ユーザプログラムを現在のPCからカーソルのある行まで実行します。
トレース(I)	ソース行単位、機械語単位のトレース実行をします。
ステップ(S)	ソース行単位、機械語単位のステップ実行をします。
リターン実行(U)	現在の関数の実行を終了(リターン)して呼び出し関数に戻るまで実行します。
ブレークポイント(P)	ブレークポイントの設定 / 解除をします。
強制ブレーク(B)	強制的にユーザプログラムを中断し、PARTNER-VR5400-TPに制御を戻します。(コマンド/マクロの連続実行中は連続実行を中断します)
キャッシュフラッシュ(F)	キャッシュのフラッシュ、初期化を行います。(『4.1.7 キャッシュフラッシュダイアログボックス』31頁参照)
エミュレーションROM設定(M)	エミュレーションROM領域を設定します。(『4.1.4 エミュレーションROM設定ダイアログ』30頁参照)
CPU環境設定(V)	CPUの環境値を設定します。(『4.1.3 CPU環境設定ダイアログボックス』29頁参照)

## 3 チャイルドウインドウ

この章では、PARTNER-VR5400-TPの機能に依存したチャイルドウインドウについて、表示や操作方法を説明します。

### 3.1 レジスタウインドウ

レジスタウインドウは、CPUレジスタ、CP0レジスタ、TLBレジスタ、FPUレジスタを表示します(図 2)。



The screenshot shows the 'Registers' window with four distinct sections:

- CPUレジスタの表示 (CPU Registers):** Contains a list of CPU registers (R0-R9) with their values and descriptions. For example, R0 is 0000000000000000 (0.000000e+00).
- システムレジスタの表示 (System Registers):** Contains a list of system registers (COUNT(9), ENTRYHI(10), COMPARE(11), STATUS(12), CAUSE(13), EPC(14), PRID(15), CONFIG(16)) with their values and descriptions.
- TLBレジスタの表示 (TLB Registers):** Contains a list of TLB entries (TLB 0 to TLB 9) with their values and descriptions. Each entry includes fields for M (Memory Address), HI (High Address), ASID (Address Space Identifier), L0 (Low Address), and L1 (Low Address).
- FPUレジスタの表示 (FPU Registers):** Contains a list of FPU registers (F2-F11) with their values and descriptions.

図 2 各レジスタ表示例

### 3.1.1 レジスタウインドウのショートカットキー

レジスタウインドウでは以下のショートカットキーにより各種コマンドが実行できます。このショートカットキーはレジスタウインドウにフォーカスがある場合に適用されます。

キー操作	機能
F3	カーソル位置のレジスタまたはフラグの初期化をします。
F6	レジスタ値のシンボル表示をします。
F7	レジスタ値が指しているメモリの内容を表示します。
F9	レジスタ値を10進数で表示します。
Enter	カーソル位置のレジスタ値またはフラグの値を変更するダイアログボックスを表示します。

### 3.1.2 レジスタウインドウのローカルメニュー



[レジスタ]メニューにはレジスタウインドウの制御に関するコマンドが収められています。

メニュー項目	機能
初期化(②)	カーソル位置のレジスタ値を0にします。
インクリメント(±)	カーソル位置のレジスタ値をインクリメントします。
デクリメント(-)	カーソル位置のレジスタ値をデクリメントします。
値の変更(C)	カーソル位置のレジスタ値を変更するダイアログボックスを表示します。
CPUレジスタ	CPUレジスタ値の表示方法を設定します。
CP0レジスタ	CP0レジスタ値の表示 / 非表示の切り替えをします。
TLBレジスタ	TLBレジスタ値の表示 / 非表示の切り替えをします。
FPUレジスタ	FPUレジスタ値の表示方法を設定します。
64ビット表示	レジスタの値の64ビット表示を設定します。

## CPUレジスタ



メニュー項目	機能
10進数表示(D)	レジスタ値を10進数で表示 / 非表示の切り替えをします。
シンボル表示(S)	レジスタ値のシンボルを表示 / 非表示の切り替えをします。
メモリ表示(M)	レジスタ値が指しているメモリ内容の表示 / 非表示の切り替えをします。

## FPUレジスタ



メニュー項目	機能
表示(F)	FPUレジスタ値の表示 / 非表示の切り替えします。
倍精度切替(X)	FPUレジスタ値の単精度表示 / 倍精度表示の切り替えをします。

## 3.1.3 レジスタウインドウでのマウス操作

レジスタまたはフラグ部分を左ダブルクリックすると、値変更を行うダイアログボックスが表示されます。

## 3.2 ブレークウインドウ

ブレークウインドウは、現在設定されているブレークポイントを表示します(図 3)。

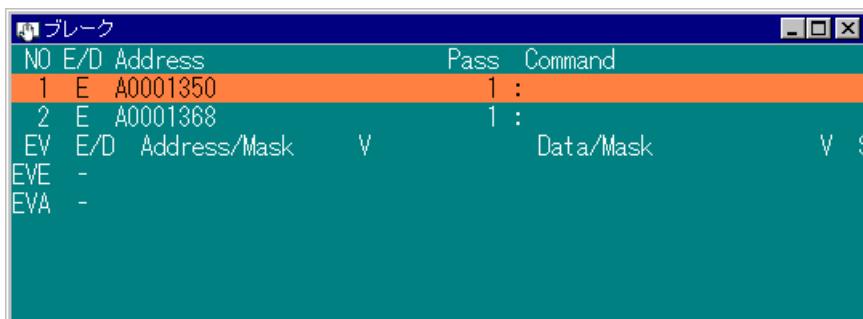


図 3 ブレークウインドウ

### 3.2.1 ブレークウインドウのショートカットキー

ブレークウインドウでは以下のショートカットキーにより各種コマンドが実行できます。このショートカットキーはブレークウインドウにフォーカスがある場合に適用されます。

キー操作	機能
F3	全てのブレークポイントを削除します。
F4	現在のブレークポイントの設定を保存します。
F6	F4で保存したブレークポイントの設定を呼び出します。
F7	アクセスブレークの設定をします。 (『4.1.5 イベントブレークの設定ダイアログボックス』31頁参照)
F9	現在選択されているブレークポイントの許可/禁止を切り替えます。
Enter	
INS	ブレークポイントの設定ダイアログボックスを表示します。
DEL	現在選択されているブレークポイントを削除します。

### 3.2.2 ブレークウインドウのローカルメニュー

ブレーク(B)	
ブレーク設定(A)...	Ins
禁止(D)	F9,Enter
削除(C)	Del
全て削除(L)	F3
設定のセーブ(S)...	F4
設定のロード(L)...	F6
アクセスブレーク(E)...	F7

[ブレーク]メニューにはブレークウインドウの制御に関するコマンドが収められています。

メニュー項目	機能
ブレーク設定(A)	新規にブレークポイントを設定するダイアログボックスを表示します。
禁止(D)/許可(E)	カーソルのあるブレークポイントの許可 / 禁止の切り替えをします。
削除(C)	カーソルのあるブレークポイントを削除します。
全て削除(L)	全てのブレークポイントを削除します。
設定のセーブ(S)	現在のブレークポイントの設定を保管します。
設定のロード(L)	以前保管されていたブレークポイントの設定をロードします。
イベントブレーク(E)	イベントブレーク設定ダイアログが設定されます。 (『4.1.5 イベントブレークの設定ダイアログボックス』31頁参照)

### 3.2.3 ブレークウインドウでのマウス操作

ブレークポイント部分を左ダブルクリックすると、ブレークポイントの有効/無効を切り替えます。

### 3.3 ヒストリウンドウ

ヒストリウンドウは、リアルタイムトレースメモリの内容を逆アセンブル表示(図 4)または、16進ダンプ表示(図 5)で表示します。

RTE-100-TPでは逆アセンブル形式のタイムタグ値は表示されません。

Cycle	Address	Stat	EXT	Data/Code	Instruction	
-12347	BFC05FD8	TPC	1111	34040001	ORI	r4, r0, 1
				time=	000,000,106,713.3us	
-12336	BFC05FDC	NSEQ	1111	0FF01491	JAL	BFC05244
				time=	000,000,106,714.7us	
	BFC05FEO		1111	00000000	NOP	
-12333	BFC05244	NSEQ	1111	27BDFFF0	ADDIU	r29, r29, -10
				time=	000,000,106,721.2us	
	BFC05248		1111	AFB000C	SW	r31, C(r29)
	BFC0524C		1111	AFB10004	SW	r17, 4(r29)
	BFC05250		1111	AFB00000	SW	r16, 0(r29)

図 4 ヒストリ(逆アセンブル表示)

Cycle	Cnt	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
-12347	11	48	DF	50	CF	BF	F0										
-12336	03	82	00														
-12333	03	88	00														
-12330	03	83	00														
-12327	03	85	00														
-12324	03	8A	00														
-12321	11	44	EF	50	CF	BF	F0										
-12310	03	83	00														
-12307	03	83	00														
-12304	03	82	00														

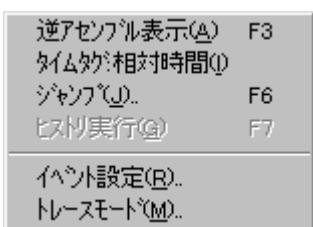
図 5 ヒストリ(ダンプ表示)

### 3.3.1 ヒストリウインドウのショートカットキー

ヒストリウインドウでは以下のショートカットキーにより各種コマンドが実行できます。このショートカットキーはヒストリウインドウにフォーカスがある場合に適用されます。

キー操作	機能
F3	表示形式を16進ダンプ/逆アセンブルに切り替えます。
F6	表示するフレーム番号のダイアログボックスを表示します。
F7	ユーザプログラム実行中、リアルタイムトレースの実行/停止を切り替えます。

### 3.3.2 ヒストリウインドウのローカルメニュー



[ヒストリ]メニューにはヒストリウインドウの制御に関するコマンドが収められています。

メニュー項目	機能
ダンプ表示(D)/逆アセンブル表示(A)	ヒストリの表示形式をダンプと逆アセンブルに切り替えます。
相対時間(I)/絶対時間(I)	タイムタグの表示形式を切り替えます。 RTE-100-TPでは使用できません。
ジャンプ(J)	表示フレームを指定するダイアログボックスを表示します。
ヒストリ実行(S)/ヒストリストップ(I)	ヒストリの実行 / 停止を切り替えます。
イベント設定(R)	トリガアドレス、データトレースアドレスを指定するダイアログボックスを表示します。(『4.1.2 イベント設定ダイアログボックス』28頁参照)
トレースモード(M)	トレースモードを指定するダイアログボックスを表示します。(『4.1.1 トレースモード設定ダイアログボックス』27頁参照)



空白ページ

## 4 ダイアログコマンド

PARTNER-VR5400-TPには、ダイアログボックスやツールバーによってデバッグを行う方法と、コマンドウインドウにダイアログコマンドを入力してデバッグを行う方法の2通りあります。

## 4.1 ダイアログボックス

PARTNER-VR5400-TPに依存したダイアログボックスの説明をします。

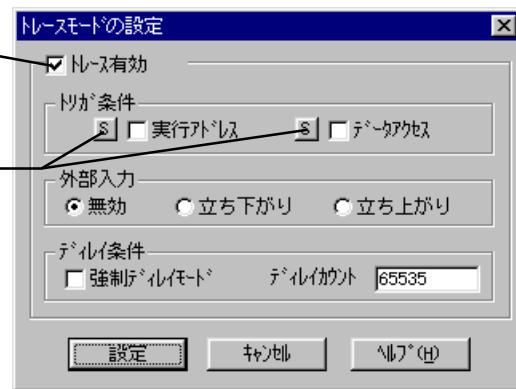
### 4.1.1 トレースモード設定ダイアログボックス

[ヒストリ]-[トレースモード]

[トレースモード設定]ダイアログボックスでは、リアルタイムトレースの条件を設定します。

リアルタイムトレースを使用するときチェックします

イベント設定ダイアログボックスを表示します



トリガ条件は、実行アドレスとデータアクセスのイベントを選択します。イベントは、『4.1.2 イベント設定ダイアログボックス』(28頁)で設定します。

ディレイ条件は、トリガ成立後に取り込むサイクル数(ディレイカウント)を0~131071(0x1FFFF)の範囲で指定します。

強制ディレイモードは、トレース開始後、ディレイカウント数分のトレースをした時点で強制的にトレースを終了するモードです。このモードは、トリガアドレス(トリガポイント)は、無視されます。

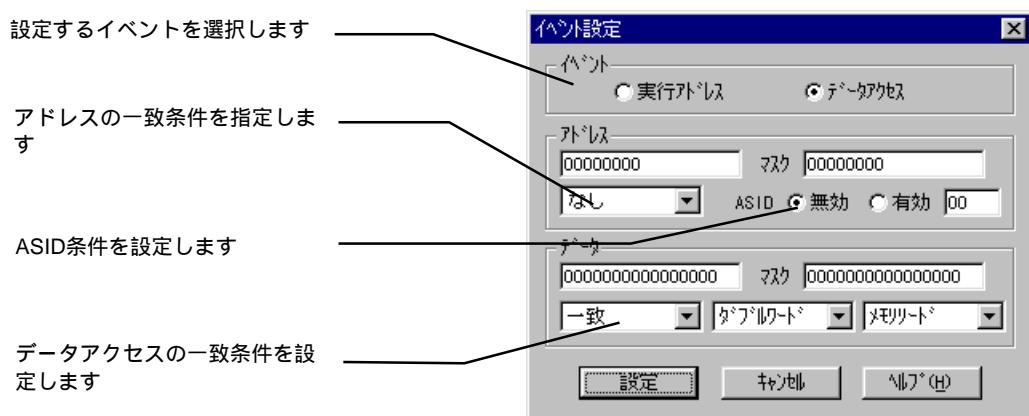
トリガは、トリガポイント(トリガアドレス)と外部信号端子、イベントユニットからのトリガポイントOR条件です。

外部入力は、トレーストリガとして使用する外部入力端子の条件を指定します。

## 4.1.2 イベント設定ダイアログボックス

[ヒストリ]-[イベント設定]

[イベント設定]ダイアログボックスでは、リアルタイムトレース、イベントブレークで使用するイベント条件を設定します。



実行アドレスの場合は、アドレス、アドレスの一致条件の指定、ASIDのみが有効になります。また、実行アドレスイベントは『ブレークポイント設定ダイアログボックス』(共通編参照)で、ROM領域へのブレークポイントに使用します。そのため、すでにブレークポイントとしてが設定されている場合は変更することができません。

アドレスは、実行アドレス/アクセスアドレス値を指定します。マスク指定することができます。

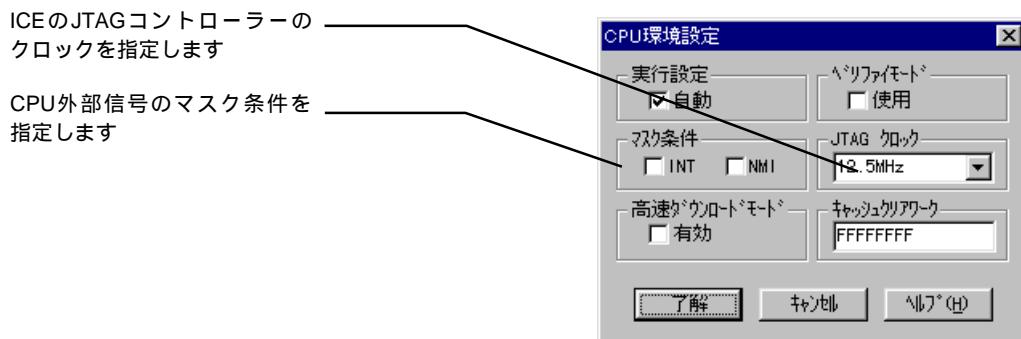
データは、値データ条件を指定します。マスク指定することができます。サイズはデータサイズで指定します。アドレスに奇数番地を指定した場合と、アドレスの最下位ビットをマスクした場合には、データサイズにバイトを選択できません。

アクセスサイクルは、データのアクセスサイクルを指定します。

### 4.1.3 CPU環境設定ダイアログボックス

#### [実行]-[CPU環境設定]

[CPU環境設定]ダイアログボックスでは、CPUの動作条件を設定します。



**実行設定**は、PARTNERおよびICEの内部処理のために、CPUをブレークさせた場合に、再実行を自動で行うか行わないかを選択します。

**ペリファイモード**は、メモリ書き込みを行う操作やコマンドのときに、書き込み後にメモリペリファイを行なうか、行わないかを選択します。

ペリファイは、最初のエラーがあったアドレスで終了します。

**高速ダウンロード**は、エミュレーションROMを使用してメモリの書き込みを高速で行なうか、行わないかを設定します。

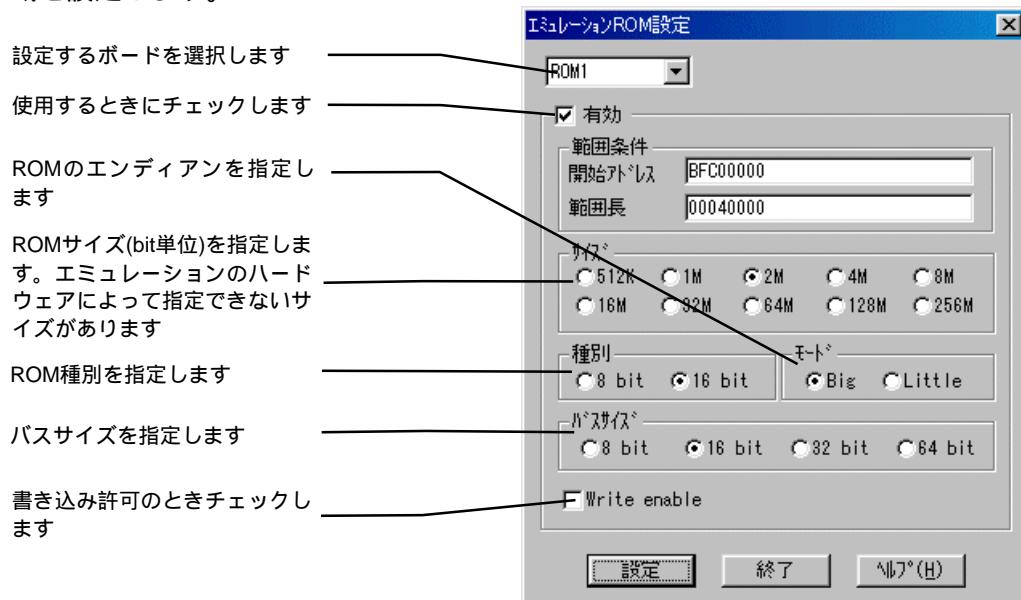
このモードは、128バイトを超えるメモリ書き込みに対して作用します。なお、エミュレーションROM領域を設定していない場合、またはエミュレーションROM領域がキャッシュ領域の場合は使用できません。使用するエミュレーションROM領域データは変更されず、保持されます。

**キャッシュクリアワーク**は、キャッシュをクリア(フラッシュ/初期化)するためのワーク領域の先頭アドレスを設定します。ワーク領域は128バイトの領域です。キャッシュ領域内を指定することはできません。キャッシュのクリアはCPUの実行時、またはCACHEコマンドで実行されます。設定された領域のデータは保証されませんので、空き領域を設定してください。

## 4.1.4 エミュレーションROM設定ダイアログ

[実行]-[エミュレーションROM設定]

[エミュレーションROM設定]ダイアログボックスは、エミュレーションROMの領域を設定します。



開始アドレスはエミュレートするROMの最下位アドレスを指定し、範囲長へは、エミュレートするROMの容量をバイト数(8バイトの境界)で入力します。

範囲長に0を指定した場合は、解除(使用しない)と同じになります。

16ビットのROMを使用する場合、種別の8bitとサイズの512Kは指定できません。

設定ボタンを押したときにのみ指定した内容が設定されます。終了ボタンでダイアログを終了します。

RTE-1000-TPの場合は、ボードの選択、Write Enableとバスサイズの64bitは指定できません。

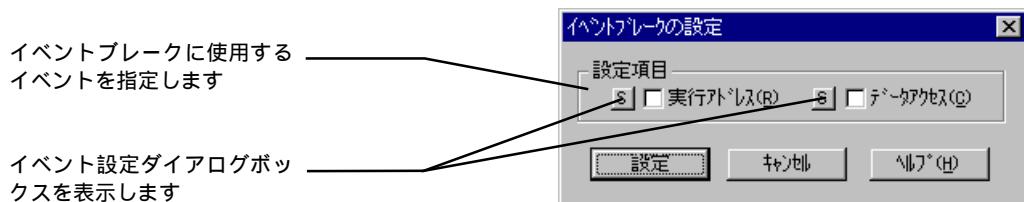
RTE-2000-TPの場合は、バスサイズにより複数枚のボードを使用します。そのため、ボードの選択とバスサイズに次の関係があります。記載がないバスサイズは設定できません。

ポート選択	バスサイズ	ポートスロット位置	ポート選択	バスサイズ	ポートスロット位置
ポート1	8 bit	#3	ポート3	8 bit	#5
	16 bit	#3		16 bit	#5
	32 bit	#3+#4		32 bit	#5+#6
	64 bit	#3+#4+#5+#6	ポート4	8 bit	#6
ポート2	8 bit	#4		16 bit	#6
	16 bit	#4			

### 4.1.5 イベントブレークの設定ダイアログボックス

[ブレーク]-[イベントブレーク]

[イベントブレークの設定]ダイアログボックスは、イベントブレークポイントを設定します。

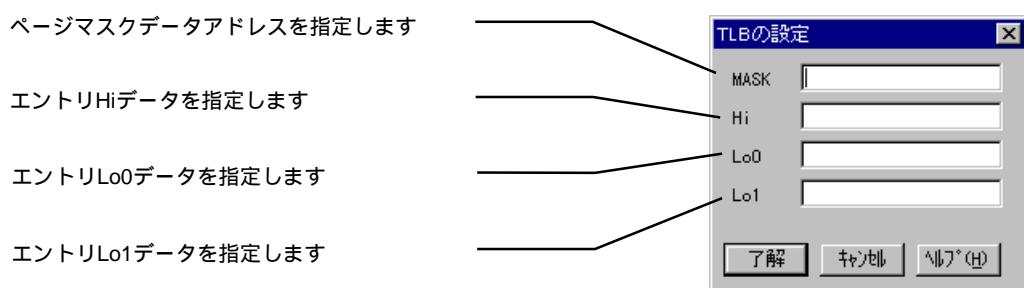


各イベントは、『4.1.2 イベント設定ダイアログボックス』(28頁)で設定します。実行アドレスイベントは、『ブレークポイント設定ダイアログ』でも使用されます。ROM領域にブレークポイントを設定したときに、実行アドレスのイベントブレークが設定されます。そのため、すでにROM領域にブレークポイントが設定されている場合は実行アドレスのイベントブレークは変更できません。

### 4.1.6 TLBの設定ダイアログボックス

[レジスタ]-[値の変更]

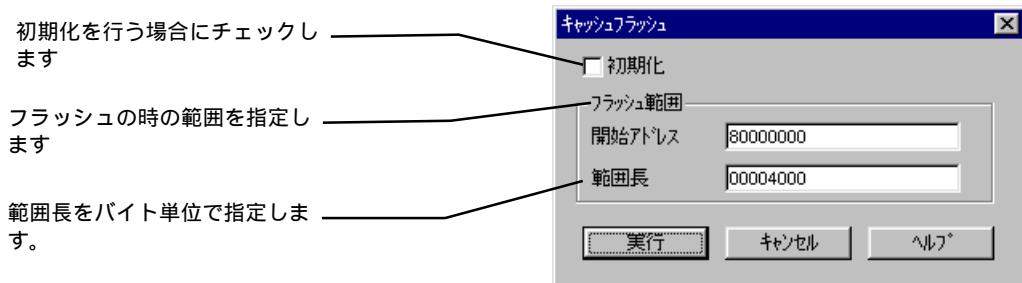
[TLBの設定]ダイアログボックスは、TLBエントリを設定します。



### 4.1.7 キャッシュフラッシュダイアログボックス

[実行]-[キャッシュフラッシュ]

[キャッシュフラッシュ]ダイアログボックスでは、キャッシュのクリア(フラッシュ、初期化)を行います。



初期化を選択した場合は、**フラッシュ範囲**は無視されます。

フラッシュ範囲の範囲長を 0 に指定した場合は、初期化になります。

フラッシュは命令、データの両キャッシュラインを無効にします。データキャッシュについてはライトバックをします。

初期化は、命令/データキャッシュのキャッシュタグ、キャッシュラインをすべて無効にします。

### 4.1.8 各種状態の設定ダイアログボックス

[設定]-[オプション設定]

[各種状態の設定]ダイアログボックスでは、PARTNER-VR5400-TPの各種状態を設定します。



実行モードは現在作用しません。将来の拡張のために用意しています。

**実行モード**はユーザプログラムの実行時の状態を制御します。  
PARTNER-VR5400-TPでは、特別な制御を行っていません。デフォルト(モード0)に設定します。

**大文字 / 小文字判別**は関数名や変数名など登録されたシンボルの大文字 / 小文字の判別をするか否かを指定します。

マーク(チェック)すると大文字 / 小文字の判別をします。マークを解除すると判別を行いません。

マークした場合は、シンボルを使用できる操作に関して、登録されたシンボルと同様に正しく大文字 / 小文字で指定します。正しく指定されていない場合は、シンボルを参照できません。

マークを解除した場合は、大文字 / 小文字に関係なく指定された文字列に該当するシンボルを参照します。

## 5 リアルタイムトレース

PARTNER-VR5400-TPはCPUの実行内容をリアルタイムにトレースすることができます。

この章ではリアルタイムトレース機能について説明します。

## 5.1 リアルタイムトレースの概要

リアルタイムトレースは、CPUから出力された実行内容（トレースデータ）を、実行ごとにICE内のトレースバッファに書き込みます。この内容は、ヒストリウィンドウで見ることができます。

トリガ条件、ディレイ条件によってトレースバッファへの取り込みを制御することができます。

トレースデータ取り込みの流れについては、図 6、図 7のようになっています。

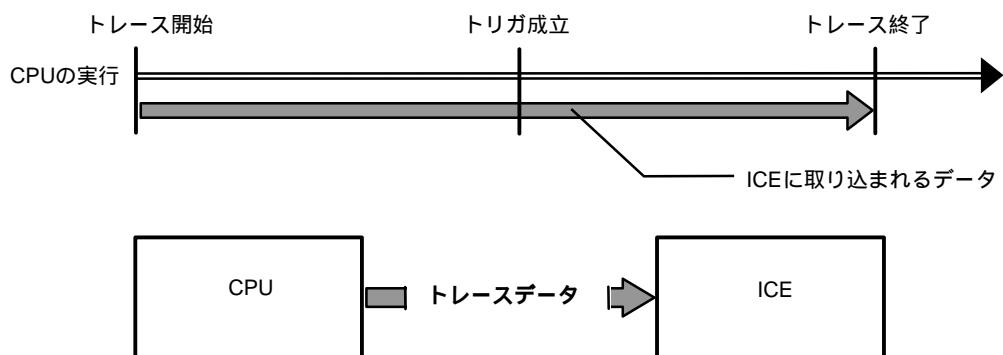


図 6 トレースデータ取り込みの流れ

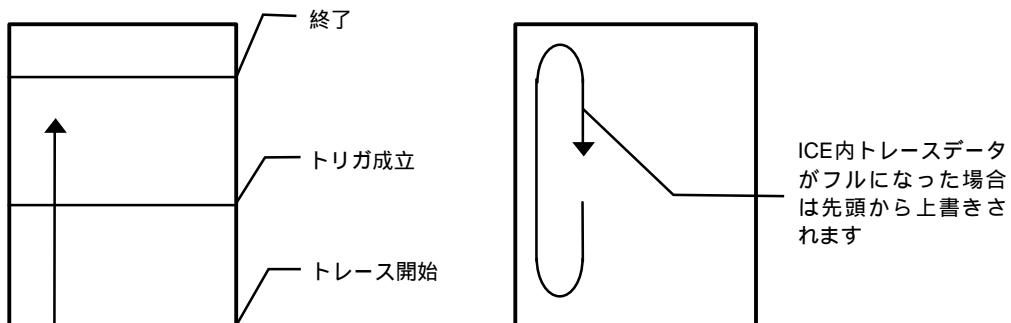


図 7 ICE内のトレースデータ

### 5.1.1 ディレイカウント

ディレイカウントは、トリガ成立後に取り込むサイクル数です(図 8)。

サイクル数は、CPUの実行内容により異なります。1サイクルが1実行単位ではありません。

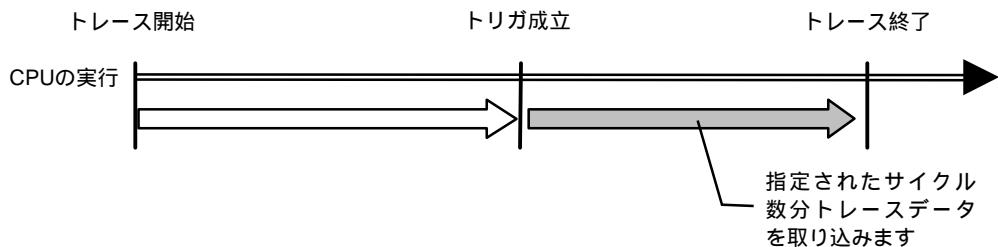


図 8 ディレイカウントの流れ

### 5.1.2 トレースの開始

トレースデータの取り込みは、CPUの実行とともに行われます(図 9)。



図 9 トレースの開始

### 5.1.3 トリガ条件

トリガはディレイカウントの起点となる条件です(図10)。トリガを設定することにより、条件前後の実行内容を見ることができます。

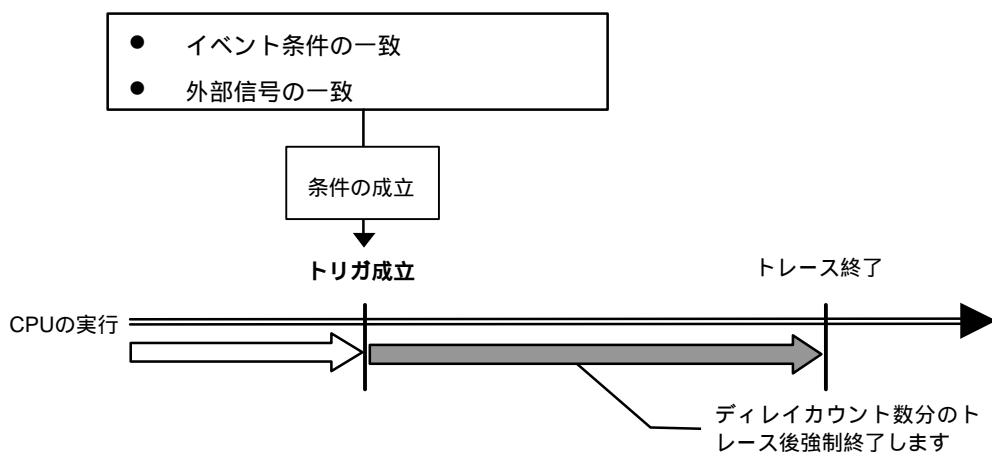


図 10 トリガ条件

### 5.1.4 トレースの終了

トレースデータの取り込みを終了します(図11)。

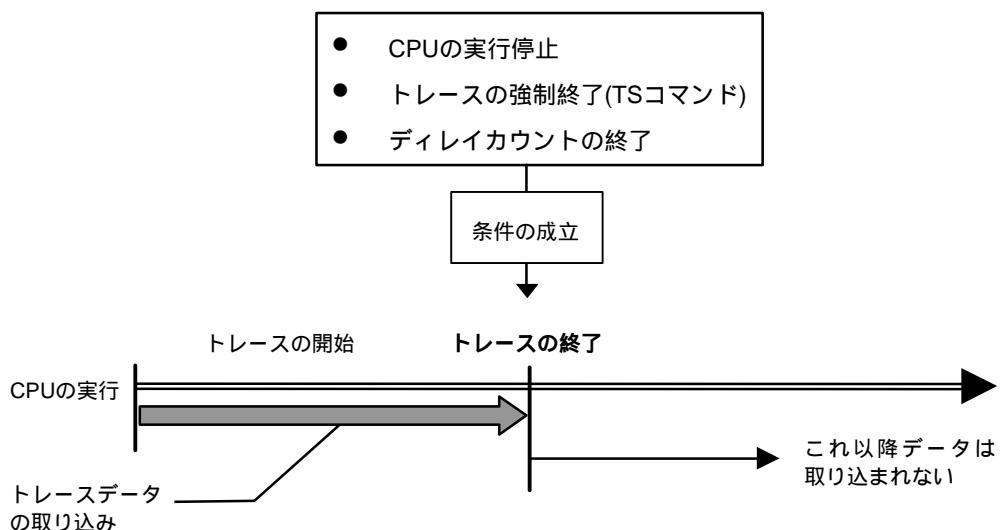


図 11 トレースの終了

### 5.1.5 強制ディレイモード

強制ディレイモードは、トレース開始後、指定されたディレイカウント(サイクル数)分取り込んだ時点で強制的にトレースを終了します。このモード中、トリガ条件は無視されます(図 12)。

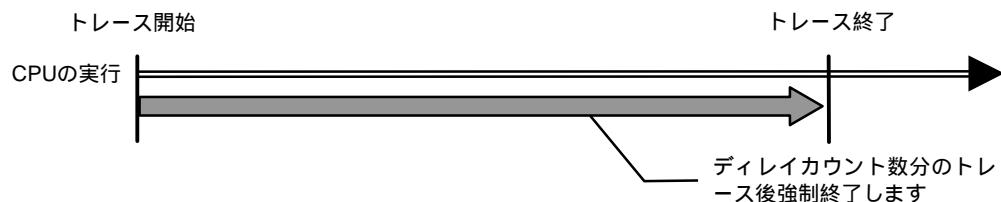


図 12 強制ディレイモード

## 5.2 タイムタグについて

RTE-1000-TP、RTE-2000-TPでは、トレースデータ毎に時間データもトレースバッファに記録されます。これにより、実行時間の目安を得ることができます。タイムタグの内容は、ヒストリウィンドウで見ることができます。

### 5.2.1 タイムタグ表示の種類

タイムタグの表示には、絶対時間表示と相対時間表示の2種類あります。

**絶対時間表示**：トレースバッファの最も古いサイクルの時間を0として、そこからの処理時間を表示します。

最も古いサイクル（記録されている最初のサイクル）がICEの内部処理である場合、そのサイクルは表示されないため、時間が0のサイクルが表示されないことがあります。

**相対時間表示**：前サイクルの時間との差を表示します。

なお、タイムタグが表示されないサイクルもあります。サイクル間の実行命令を補完している場合、タイムタグデータが正しく記録されていない場合は表示されません。

### 5.2.2 タイムタグの注意

PARTNER-VR5400-TPおよびICEは、内部処理のためにユーザープログラムを暗黙の内にブレークし実行することがあります。そのような場合、それらの処理時間もタイムタグに含まれてしまいます。

また、上記の場合を含めて、トレースバッファにブレーク、または実行の前後が記録されている場合は、内部処理のトレースデータは表示しません。そのため、ブレークまたは実行前後では、絶対時間表示と相対時間表示の時間が一致しない場合があります。

## 6 コマンドリファレンス

PARTNER-VR5400-TPのコマンドウインドウに入力可能なコマンドの説明をします。詳しい内容(入力例など)はオンラインヘルプを参照してください。

## 6.1 コマンド解説の規約

PARTNER-VR5400-TPのコマンドは、コマンド名とパラメータリストによって構成されます。また、パラメータは省略できる場合もあります。省略できるパラメータに関しては鉤括弧([...])で示します。2通り以上の選択要素がある場合には中括弧({...})と|で要素内容を記述します。パラメータが省略された場合は、PARTNER-VR5400-TPの初期値や以前実行したコマンドの続きの値などが用いられます。

## 6.2 機能別コマンド

### 6.2.1 ハードウェアの初期化

INIT

PARTNER-VR5400-TP と ICE を初期化します。

### 6.2.2 CPUリセット

RESET

ターゲット CPU をリセットします。

### 6.2.3 環境設定コマンド

ENV [!]AUTO][!]NMI][!]INT][!]VERIFY]

[JTAG{25|12|5M|2M|1M|500K|250K|100K }][!]HI][ワ-ケアドレス]

各種入力信号のマスクなどの CPU の各種環境を設定します。

斜体のパラメータは RTE-1000/2000-TP のみ

ROM [アドレス[範囲]] [ROM8|ROM16][BUS8|BUS816|BUS32]  
[512K|1M|2M|4M|8M|16M|32M|64M|128M|256M][/B|/L]

エミュレーション ROM の種類、アドレス範囲を設定します。

RTE-200/1000-TP 用です。

ROM1 ~ ROM4 [アドレス[範囲]] [ROM8|ROM16][BUS8|BUS816|BUS32|BUS64]  
[512K|1M|2M|4M|8M|16M|32M|64M|128M|256M][/B|/L][!]WREN]

エミュレーション ROM の種類、アドレス範囲を設定します。

RTE-2000-TP 用です。

NC [アドレス 範囲] 非メモリキャッシュの領域を設定します。

NCD リスト 非メモリキャッシュの領域を解除します。

NROM [アドレス 範囲] エミュレーションメモリ割付除外領域の参照と設定します。

NROMD リスト エミュレーションメモリ割付除外領域を解除します。

ACCS [32|64] メモリアクセス時のアクセスサイズを設定します。

#### 6.2.4 プログラムロード

L [ファイル名] テ' バ' グ' プ' ロ' ムとテ' バ' グ' 情報を読み込みます。

#### 6.2.5 ファイルリード/ライト

RD ファイル名, 番地 指定ファイルを指定番地に書き込みます。

WR ファイル名, 範囲 範囲指定されたメモリ内容を指定ファイルに書き込みます。

### 6.2.6 プログラム実行

T [回数]	デバッグプログラムをトレース実行します。 (F8)
P [回数]	デバッグプログラムをステップ実行します。 (F10)
G [=実行番地][,ブレーク番地][,/W]	
G@ [,/W]	デバッグプログラムを実行します。 (F5,F7) /W:実行中コマンド禁止
SC	デバッグプログラムを強制ブレークします。 (ESC)

### 6.2.7 ブレークポイント

BP [番地[,回数[,コマンド]]]	ブレークポイントを指定番地に設定します。 (F9)
BC リスト	リストで指定されたブレークポイントを解除します。
BD リスト	リストで指定されたブレークポイントを無効にします。
BE リスト	リストで指定されたブレークポイントを有効にします。

### 6.2.8 イベントブレークポイント

BH [イベント]	イベントブレークポイントを設定します。
BHD [イベント]	指定されたイベントブレークポイントを無効にします。

### 6.2.9 コード表示

V[.][ファイル名:][行]	指定ファイルの指定行をコードウインドウに表示します。
V 関数名	指定関数のソースファイルをコードウインドウに表示します。
U [番地]	コードウインドウに指定番地から逆アセンブル表示を行います。
UPUSH [番地]	現在の表示アドレスをアドレススタック(8段の内部スタック)に PUSH して指定番地から逆アセンブル表示します。
UPOP	最後に UPUSH されたアドレスから逆アセンブル表示してアドレススタックを POP します。
UEND	最後に UPUSH されたアドレスから逆アセンブル表示します。

### 6.2.10 レジスタ表示/変更

R	レジスタの値を表示します。
_レジスタ=式	指定レジスタを式の値に変更します。
R レジスタ	レジスタ値の変更をします。

### 6.2.11 システムレジスタ表示/変更

CPO	CPO レジスタ表示
CPO CPO レジスタ名(番号)=データ	指定 CPO レジスタを変更します。
TLB [レジスタ番号][,レジスタ番号(E)]	TLB レジスタ表示
TLB レジスタ番号,マスク,エントリ Hi,エントリ Lo0,エントリ Lo1	指定 TLB レジスタを変更します。
FPU	FPU レジスタ表示
FPU[D] FPU レジスタ名(番号)=データ	指定 FPU レジスタを変更します。FPUD は 64 ビット実数表示をします。

## 6.2.12 メモリ表示/変更

D[型]範囲[,回数][,基数]

範囲内のメモリ内容を型書式と基数指定にしたがって表示します。

E[型] 番地	指定番地より型指定された形式でメモリ内容を変更します。
F[型] 範囲,リスト	指定範囲を型指定された形式でリストの値でフィル(Fill)します。
S[型] 範囲,リスト	指定範囲を型指定された形式でリストのメモリバッファを検索します。
C 範囲,番地	指定範囲を指定番地と比較(コバア)します。
M 範囲,番地	指定範囲を指定番地にロック移動します。

## 6.2.13 I/Oポート入力/出力

PI[型] 番地[,/C] 型指定された形式で指定アドレスの内容を表示します。

PO[型] 番地,データ[,/C]  
型指定された形式で指定アドレスにデータを出力します。

## 6.2.14 シンボル表示/設定

X[シンボル名] シンボル名(指定がなければすべてのシンボル)を表示します。

[.]名前=アドレス 名前のシンボルを指定アドレスで登録(変更)します。

## 6.2.15 アセンブル

A 番地 指定番地よりアセンブルしてメモリに直接展開します。

## 6.2.16 バックトレース

K C の関数のバックトレース表示を行います。

## 6.2.17 リアルタイムトレース

TRC	現在のリアルタイムトレースモードを表示します。
TRC [テイクイ値] [[!]DELAY] [[!]EVA] [[!]EVE][NOEXT NEGA POSI] [[!]TRON]	リアルタイムトレースのモードを設定します。
TS	リアルタイムトレースの停止
TG	リアルタイムトレースの再開
EV	現在のイベント設定を表示します。
EV {E A} [アドレス アドレスマスク] [アドレス条件], [データ値 データマスク] [データ条件] [データサイズ] [データアクセス] [AISD 条件]	イベントを設定します。
TD	リアルタイムトレースの内容を表示するモードに入ります。 リアルタイムトレース表示モードの内部コマンド
B	表示開始フレームをトレースメモリの先頭に移動します。
D [フレームアドレス S][,フレームアドレス E]	表示モードをダンプ表示に変更し、表示します。
U [フレームアドレス S][,フレームアドレス E]	表示モードを逆アセンブル表示に変更し、表示します。
T	RTE-1000/2000-TP のとき、タイムタグ表示の絶対表示と相対表示を切り替えます。

## 6.2.18 キャッシュフラッシュ/初期化

CAHE [アドレス[範囲長]]	キャッシュフラッシュを行います。
CAHE INIT	キャッシュ制御回路の初期化を行います。

### 6.2.19 Cのデータ参照/変更

INS C の式[, 関数] C の式を評価してインサートウインドウに表示します。 (F6, Ctrl+I)

W? C の式 C の式をウォッチウインドウに登録します。 (Shift+F7, Ctrl+W)

W[型] 番地[, 範囲][, 基数]  
番地と範囲で指定されたメモリ内容をウォッチウインドウに登録します。

Y リスト リストで指定されたウォッチ行を削除します。

VAL C の式[, 関数]

? C の式[, 関数]  
C の式を評価して表示します。

### 6.2.20 文字列の定義(Cの式評価)

DEF 文字列 1 文字列 2  
#DEFINE 文字列 1 文字列 2  
ミニプロセッサのための文字列を定義します。

DEF 現在登録されている文字列定義内容の表示をします。

DEF \* 現在登録されているすべての文字列定義を無効にします。

### 6.2.21 システム制御

EXIT PARTNER を終了します。

Q PARTNER を終了します。

HELP ヘルプ表示を行います。

VER PARTNER のバージョン表示をします。

MAP 設定されたメモリマップを表示します。

!! コマンドラインヒストリの表示をします。

! 文字列 コマンドラインヒストリの文字列サーチをします。

& スルーコマンドへの移行/終了

& スルーコマンド スルーコマンドの実行

## 6.2.22 システムコール

SYSC アドレス	アドレスで指定されたアドレスをエントリポイントとしてシステムコールを ON します。
SYSC OFF	システムコール機能を OFF します。
SYSC	システムコール機能の状態表示

## 6.2.23 式の表示

H 式	式の値を 8,10,16 進数, ASCII および実数で表示します。
H 式 1, 式 2	式 1 と式 2 の和と差の値を表示します。
PRINTF 書式[, パラメータ]	
PF 書式[, パラメータ]	C の関数の printf() と同様なフォーマット表示をします。

## 6.2.24 基数変更

N 基数	入力基数を 10 進数または 16 進数に設定します。
------	-----------------------------

## 6.2.25 ロギング/バッチ

> ファイル名	コマンド ウィンドウの表示/入力をファイルに出力します。
>> ファイル名	指定ファイルへのアpend (APPEND) ロギング 出力します。
>	ロギング の中断(ログファイルのクローズ)をします。
< ファイル名	コマンド ウィンドウの入力をファイルから行います。ESC キーでバッチ処理の中断が可能です。

## 6.2.26 オプション設定

OPTION {ON OFF}	大小文字判別をします。(SHIFT+F10) 画面コントロール/その他
CLS	コマンド ウィンドウのクリアをします。
HOME	コマンド ウィンドウのカーソルをホームポジションへ移動します。
LOCATE X 座標, Y 座標	コマンド ウィンドウのカーソルを指定した位置へ移動します。
LALL	マクロでの表示出力指定をします。
SALL	マクロでの表示出力抑制指定をします。
LIST	コマンド ウィンドウの表示出力指定をします。
NLIST	コマンド ウィンドウの表示出力抑制指定をします。
BEL	ベルを鳴らします。
TIME	現在時刻(時:分:秒)を表示します。
WAIT	一時停止します。
PROMPT 文字	プロンプトを指定された文字に変更します。
*	コメント行指定をします。

## 6.2.27 マクロコマンド

{ マクロ名	マクロ名でマクロ本体を登録します。
DO{ } WHILE 式	C の do..while 文と同様なマクロコマンド
FOR{ }	C の for 文と同様なマクロコマンド
WHILE{ 式	C の while 文と同様なマクロコマンド
REPEAT{ パラメータ	リピートマクロコマンド
BREAK	マクロから抜け出します。
KILL マクロ名	マクロの削除をします。
MLIST [マクロ名]	マクロの内容を表示します。
MLIST >ファイル名	現在登録されているすべてのマクロを指定ファイルに書き込みます。
<ファイル名	指定されたマクロファイルからマクロ読み込みをします。
IF{ 式	C の if, elseif, else 文と同様な条件制御コマンドです。

### 6.2.28 フラッシュメモリ管理

ZF Stat[,アドレス]	フラッシュメモリの状態表示
ZF Erase, アドレス	指定のアドレスを含むフラッシュメモリの全領域消去
ZF Flush	未書き込みデータの書き込み
ZF Clear	未書き込みデータの破棄

## 7 高級言語レベルデバッグ

PARTNER-VR5400-TPでは、Green Hills、GNU(exeGCC)、およびCodeWarriorのC言語をサポートしています。

これらの処理系を使ったROM化についての詳しい内容は各処理系のマニュアルを参考にしてください。ここでは、PARTNER-VR5400-TPが必要とするデバッグ情報を出力する方法を説明します。

PARTNER-VR5400-TPはデバッグ対象となるプログラムを読み込む時、同時にそのプログラムのデバッグ情報を読み込みます。したがって、Cコンパイラあるいはアセンブラーからデバッグ情報が正しく生成されていなければ正常なデバッグができません。

## 7.1 GreenHills C

PARTNER-VR5400-TPはELFファイルフォーマットの実行ファイルを読み込みます。実行ファイルの拡張子は”.OUT”です。

### 7.1.1 コンパイルとリンク方法

C言語で作成したモジュールのコンパイル時には、コマンドラインに-g -XDWARF オプションを付加します。このオプションを付加してコンパイルしたモジュールはソースレベルデバッグが可能です。

さらに、-oオプションで実行ファイルのファイル名を指定する場合は、拡張子を .out にします。

```
> ccmipe -cpu=r5000 -g -XDWARF ソースファイル名 -o実行ファイル名.out
ソースファイル名      ソースファイル名
実行ファイル名      実行ファイル名
```

ビルダーを使用してC言語で作成したモジュールのコンパイル時には、以下に示すビルドオプションを付加してください。

```
:dwarf=true
:debuglevel=plain
:outputname=実行ファイル名.out
```

この結果、ELFファイルフォーマット(.OUT)の実行ファイルが作成されます。

なお、オプティマイズ関連のオプションを設定した場合には、正しくデバッグできないことがありますので、オプティマイズなしの設定にしてください。

### 7.1.2 使用例

#### 1 ) コンパイル

```
> ccmipe -cpu=r5000 -g -XDWARF -c @cmpfiles
cmpfiles      コンパイル対象ソースファイルのリストファイル
              (複数のソースファイル名称を指定)
```

この結果、拡張子”.o”のオブジェクトファイルが作成されます。

## 2 ) リンク

```
>lx -map=chkpro.map -o chkpro.out @chkpro.lx @lnkfiles
  chkpro.lx    リンカに引渡すオプション定義(リンクマップ)ファイル
  lnkfiles     リンク対象オブジェクトファイルのリストファイル
                (複数のオブジェクトファイル名称を指定)
```

この結果、ELFファイルフォーマット(.OUT)の実行ファイルが作成されます。

## 3 ) リンクマップの例

実行プログラムで指定される各セクションをメモリ空間のどの位置に配置するのかを決定します。

```
-m 3 -sec
{
  .text      0x80010000      :
  .rodata    align(8)        :
  .syscall   :               :
  .secinfo   :               :
  .fixaddr   :               :
  .fixtype   :               :
  .data      align(8)        :
  .sdatabase align(8)        :
  .sdata     :               :
  .rosdata   :               :
  .sbss      align(8)        :
  .bss       align(8)        :
  .heap      align(8) pad(0x4000) :
  .stack     align(8) pad(0x2000) :
```

### 7.1.3 MULTI2000での作成

MULTI2000を使用して作成する場合、次の操作でデバッグ情報を出力してください。

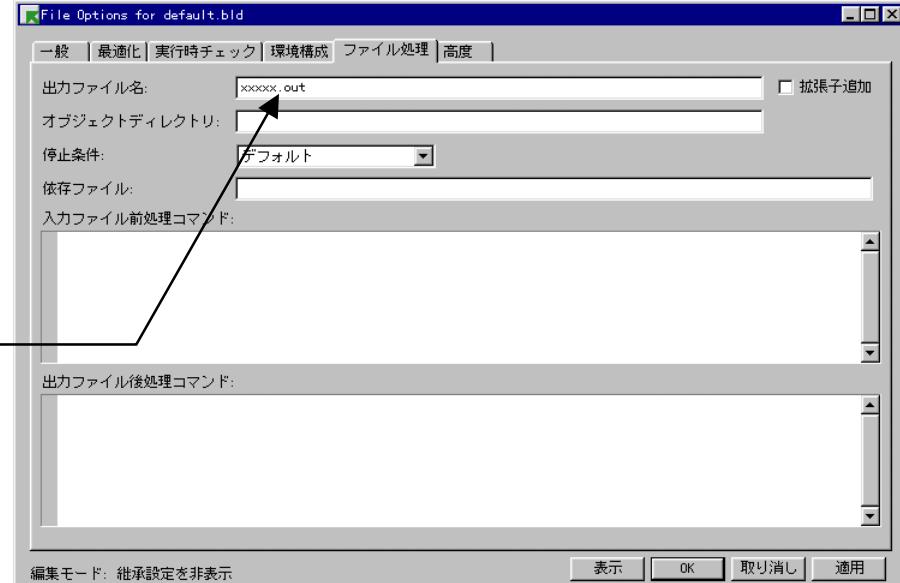
- ProjectメニューからFile Optionダイアログを表示します。
- 高度タブを表示します。
- dualデバッグフォーマット出力を選択します。
- コンパイル対象ドロップリストでアセンブリを選択します。



また、次の操作で出力ファイルの拡張子を”.out”にしてください。

- ・ ファイル処理タブを表示します。
- ・ 拡張子”.out”のファイル名を指定します。

出力ファイルの拡張子をoutにします



コマンドラインで使用する場合は、 -G -dual\_debug -noobj -o出力ファイル.out を追加指定してください。

## 7.2 GNU C(exeGCC)

PARTNER-VR5400-TPはGNU Cの実行ファイルを読み込みます。実行ファイルの拡張子は”.OUT”です。

### 7.2.1 コンパイルとリンク方法

C言語で作成したモジュールのコンパイル時には、-gオプションを付加してください。このオプションを付加してコンパイルしたモジュールはソースレベルデバッグが可能です。さらに、-oオプションで実行ファイルのファイル名を指定する場合は、拡張子を.outにしてください。以下にコンパイル及びリンク時の基本的な指定方法を明記します。

```
>gcc -mips3 -g ソースファイル名 -o実行ファイル名.out -Tコマンドファイル名
ソースファイル名      ソースファイル名
実行ファイル名      実行ファイル名
コマンドファイル名      リンク用コマンドファイル
```

この結果、COFFフォーマット(.OUT)の実行ファイルが作成されます。

なお、オプティマイズ関連のオプションを設定した場合には、正しくデバッグできないことがありますので、オプティマイズなしの設定にしてください。

### 7.2.2 使用例

#### 1 ) コンパイル

```
>gcc -g -mips3 -c @mpfiles
mpfiles      コンパイル対象ソースファイルのリストファイル
              (複数のソースファイル名称を指定)
```

この結果、拡張子”.o”のオブジェクトファイルが作成されます。

#### 2 ) リンク

```
>ld -Map chkpro.map -Tchkpro.cmd -o chkpro.out @lnkfiles
chkpro.cmd  リンカに引渡すオプション定義(リンクマップ)ファイル
lnkfiles    リンク対象オブジェクトファイルのリストファイル
              (複数のオブジェクトファイル名称を指定)
```

この結果、COFFファイルフォーマット(.OUT)の実行ファイルが作成されます。

### 3 ) リンクマップの例

実行プログラムで指定される各セクションをメモリ空間のどの位置に配置するのかを決定します。

```
SECTIONS
{
    _xfer = 0 ;
    _stack_init = 0x8001ffff ;
    _heep_size = 0x4000 ;
    .text 0x80001000 :
    {
        *(.text)
    }
    .rodata . :
    {
        *(.rodata)
        _erdata = . ;
    }
    .data :
    {
        _data = . ;
        *(.data)
        _edata = . ;
    }
    .gp = . :
    .lit8 . : {
        *(.lit4)
    }
    .sbss . :
    {
        _fdss = . ;
        *(.sbss)
        *(.scommon)
    }
    .bss . :
    {
        *(.bss)
        *(COMMON)
        _end = . ;
        _heep = . ;
    }
}
```

## 7.3 CodeWarrior C

PARTNER-VR5400-TPはELFフォーマットの実行ファイルを読み込みます。実行ファイルの拡張子は”.OUT”です。

【注意】PARTNER-VR5400-TPの起動時に設定するデバッグモード設定オプションには、”-XCW”(CodeWarrior Cのデバッグモード)を選択してください。

### 7.3.1 コンパイルとリンク方法

C言語で作成したモジュールのコンパイルは、ビルド・ツール(CodeWarriorIDE)で行います。

ビルド・ツールでのプロジェクト作成時に、ターゲット設定オプションに関して以下の設定にします。

デバッグフォーマット : DWARF

オプティマイズレベル : 0

さらに、プロジェクトに必要な各種の環境設定(HeapSize, StackSize, Section Mappings...)および必要なモジュールの登録をします。

この結果、ELFフォーマット(.OUT)の実行ファイルが作成されます。

なお、オプティマイズ関連のオプションを設定した場合には、正しくデバッグできないことがありますので、オプティマイズなしの設定にしてください。

### 7.3.2 使用例

#### 1 ) コンパイルとリンク

ビルド・ツール(CodeWarriorIDE)で行います。

#### 2 ) リンクマップの例

実行プログラムで指定される各セクションをメモリ空間のどの位置に配置するのかを決定します。

```
MEMORY {
    .text      (RX) :  ORIGIN = 0xa0020000, LENGTH = 0x40000
    .data      (RW) :  ORIGIN = 0xa0060000, LENGTH = 0x20000
    .init      (RW) :  ORIGIN = AFTER(.text, .data), LENGTH = 0x10000
}
SECTIONS {
    ..code :
    {
        _ftext = .;
        *      (.text)
        .= ALIGN(0x8);
        _etext = .;
        *      (.rodata)
        .= ALIGN(0x4);
    } > .text
    ..code_cpp:
    {
        *      (.vtables)      # Only needed for C++
        *      (.exception)    # Only needed for C++
        .= ALIGN(0x4);
        __exception_table_start__ = .;
        EXCEPTION           # Only needed in C++
        __exception_table_end__ = .;

    } >> .text
    ..data :
    {
        *      (.data)
        .= ALIGN(0x8);
        _fsdata = .;
        _gp = _fsdata + 0x00008000;
        *      (.sdata)
        .= ALIGN(0x8);
        _fsbss = .;
        *      (.sbss)
        .= ALIGN(0x8);
        _esbss = .;
        _fbss = .;
        *      (.bss)
    }
}
```

( 次頁に続く )

```
        . = ALIGN(0x8);
        _ebss = .;
        *           (COMMON)
} > .data
..tmp :
{
    __static_init = .;
    STATICINIT
    . = ALIGN(8);
} > .init
_stack_addr = . + 0x4000;
_heap_addr = _stack_addr;
_heap_end = _heap_addr + 400*1024;
}
```

空白ページ

## 8 フラッシュメモリのサポート

PARTNER-VR5400-TPでは、メモリアクセス系のダイアログコマンド( E, F, M, L, RDなどのコマンド ) やウィンドウコマンドで、フラッシュメモリを意識することなく書き込みが行えます。

ZFコマンドは、フラッシュメモリの制御や状態表示のためのコマンドです。

実行制御系のダイアログコマンド( G, T, Pなどのコマンド ) やウィンドウコマンドでは、フラッシュメモリはROMとして取り扱います。

したがって、ブレークポイントの制御は、ハードウェア・ブレーク制御になります。

## 8.1 対応デバイス

書き込みに対応するデバイスは、以下の仕様を満たすものに限られます。

1 . AMD製またはその互換品で以下の項目に適合すること

- 1) 単一電源による電気的消去及び書き込み可能なデバイスであること
- 2) JEDEC標準コマンドで消去および書き込み制御が可能であること
- 3) ブートセクタ以外のセクタサイズが64KBであること
- 4) ボトムブートタイプのセクタ構成は以下の仕様であること  
[16KB x 1セクタ] + [8KB x 2セクタ] + [32KB x 1セクタ] + [64KB x nセクタ]
- 5) トップブートタイプのセクタ構成は以下の仕様であること  
[64KB x nセクタ] + [32KB x 1セクタ] + [8KB x 2セクタ] + [16KB x 1セクタ]
- 6) セクタ保護(ロックプロテクト)がなされていないこと
- 7) CPUに8/16/32ビットのデータバスで接続されていること
- 8) デバイスの空間がバンク化されていたり一部の空間がマッピングされていないなどの特殊な回路構成でないこと(デバイスの全空間がCPUのメモリ空間にリニアにマッピングされていること)

2 . Intel製またはその互換品で以下の項目に適合すること

- 1) 単一電源による電気的消去及び書き込み可能なデバイスであること
- 2) ブートセクタ以外のセクタサイズが64/128/256KBであること
- 3) ボトムブートタイプのセクタ構成は以下の仕様であること  
[8KB x 8セクタ] + [64/128/256KB x nセクタ]
- 4) トップブートタイプのセクタ構成は以下の仕様であること  
[64/128/256KB x nセクタ] + [8KB x 8セクタ]
- 5) セクタ保護(ロックプロテクト)がなされていないこと
- 6) CPUに8/16/32ビットのデータバスで接続されていること
- 7) デバイスの空間がバンク化されていたり一部の空間がマッピングされていないなどの特殊な回路構成でないこと(デバイスの全空間がCPUのメモリ空間にリニアにマッピングされていること)

3 . 上記 1 , 2 仕様とセクタサイズ、セクタ構成のみが異なるデバイス

上記の項目を満たさないデバイスは、ダイアログコマンドやウィンドウコマンドでの書き込みはできませんが、読み込みやユーザプログラムでのアクセスは可能

です。現時点で 1 と 2 の仕様を満たすデバイス例を次表に示します。

### 【対応デバイス例】

AMD	Am29F040, Am29F400, Am29F800, Am29LV004, Am29LV400
ST	M29F040, M29F400, M29W008, M29W800
富士通	MBM29F040, MBM29F400, MBM29LV800, MBM29LV008, MBM29LV160
東芝	TC58FVT400, TC58FVB400, TC58FVT800, TC58FVT160
INTEL	28F***C3, 28F***w18, 28F***W30, 28F***K**, 28F***J3*
日立	HMB32S4, HMT32S4
三菱	M***B33****, M***T33****

## 8.2 環境設定

ダイアログコマンドやウィンドウコマンドで書き込みを行うためには、コンフィギュレーションファイル(RPTVR5400T.CFG)にフラッシュメモリの領域設定が必要になります。設定には、セクタ構成によって2種類(FLASH\_MEM、FLASH\_MEM2)があります。

### 領域設定書式1(FLASH\_MEM)

『8.1 対応デバイス』の1または2の仕様を満たすデバイスの場合に使用します。ターゲットボードに実装されているデバイスのアドレス、タイプおよびバス幅を設定します。この書式は、FLASH\_MEM2の設定と合わせて最大8個まで設定できます。複数定義する場合は昇順に定義してください。

書式： FLASH\_MEM 開始アドレス, 終了アドレス, メモリタイプ,  
CPU/バス幅 [, FLASH/バス幅]

開始アドレスと終了アドレスには、デバイスがマッピングされているメモリ領域の範囲を指定します。

メモリタイプには、デバイスの種類によって以下の何れかを指定します。

メモリタイプ	デバイスタイプ	セクタ構成
TOP	AMDタイプ	トップブート
BOTTOM	AMDタイプ	ボトムブート
NO	AMDタイプ	ブートブロックなし
I32W_TOP	INTELタイプ、ライトバッファ有	トップブート 32Kword(64KB)/セクタ
I32W_BOTTOM	INTELタイプ、ライトバッファ有	ボトムブート 32Kword(64KB)/セクタ
I32W_NO	INTELタイプ、ライトバッファ有	ブートブロックなし 32Kword(64KB)/セクタ
I32_TOP	INTELタイプ	トップブート 32Kword(64KB)/セクタ
I32_BOTTOM	INTELタイプ	ボトムブート 32Kword(64KB)/セクタ
I32_NO	INTELタイプ	ブートブロックなし 32Kword(64KB)/セクタ
I64_NO	INTELタイプ	ブートブロックなし 64Kword(128KB)/セクタ
I128_NO	INTELタイプ	ブートブロックなし 128Kword(256KB)/セクタ

ライトバッファは、コマンド41Hで制御できるものでなければなりません。

CPUバス幅には、CPUとの接続バス幅で8/16/32bitいずれかのバス幅を指定します。たとえば、8bitバスのデバイス2個を組み合わせて16bitバスでCPUに接続した場合は16を指定します。

FLASHバス幅には、デバイスのデータバス幅で8/16bitいずれかのバス幅を指定します。省略された場合には、CPUバス幅と同じになります。但し、CPUバス幅が32の場合には、8または16を必ず指定してください。

#### 領域設定書式2(FLASH\_MEM2)

『8.1 対応デバイス』の3の仕様を満たすデバイスの場合に使用します。ターゲットボードに実装されているデバイスのアドレス、タイプおよびバス幅を設定します。この書式は、FLASH\_MEMの設定と合わせて最大8個まで設定できます。複数定義する場合は昇順に定義してください。

書式 : FLASH\_MEM2 開始アドレス, メモリタイプ, セクタ情報,  
CPUバス幅 [, FLASHバス幅]

開始アドレスは、デバイスがマッピングされているメモリアドレスを指定します。

メモリタイプには、デバイスの種類によって以下の何れかを指定します。

メモリタイプ	デバイスタイプ
0	AMDタイプ
1	INTELタイプ
1001	INTELタイプ、ライトバッファ有 ライトバッファは、コマンド41H で制御できるものでなければなりません

セクタ情報には、デバイスのセクタ構成を次の書式で指定します。

セクタサイズ:セクタ数[,セクタサイズ:セクタ数[,セクタサイズ:セクタ数, …]]

セクタサイズ:セクタ数でデバイスのセクタ仕様を記述します。

セクタサイズ:セクタ数で1セットとして128セットまで指定できます。

セクタサイズ 1セクタのサイズ(16進数)を指定します。

セクタ数 連続するセクタ数(10進数)を指定します。

CPUバス幅には、CPUとの接続バス幅で8/16/32bitいずれかのバス幅を指定します。たとえば、8bitバスのデバイス 2 個を組み合わせて16bitバスでCPUに接続した場合は16を指定します。

FLASHバス幅には、デバイスのデータバス幅で8/16bitいずれかのバス幅を指定します。省略された場合には、CPUバス幅と同じになります。但し、CPUバス幅が32の場合には、8または16を必ず指定してください。

なお、この書式を定義する場合は、MAP書式(『共通編 3.3章を参照』)と整合性を保つ必要があり、次の条件を満たさなければなりません。

- 1 ) MAP書式の後にこの書式を記述すること
- 2 ) MAP書式で設定した領域内にこの書式で設定する領域が含まれること
- 3 ) 1 つのMAP書式で設定した領域内にこの書式で定義する領域が含まれること

上記の条件を満たさない場合は、PARTNER-VR5400-TPの起動時にエラーメッセージが表示され起動しません。

## コンフィギュレーションファイル定義例

---

正しい設定 1 - 1 : 単一ブロックの設定

MAP 00000000, FFFFFFFF  
FLASH\_MEM 03000000,037FFFF, NO, 32, 8

正しい設定 1 - 2 : 単一ブロックの設定

MAP 00000000, FFFFFFFF  
FLASH\_MEM2 00000000,0,2000:8,10000:126,2000:8,16,16

正しい設定 2 - 1 : 複数ブロックの設定

MAP 00000000, 003FFFFF  
MAP 00400000, 007FFFFF  
MAP 00800000, 00FFFFFF  
MAP 02000000, 02FFFFFF  
MAP 03000000, 037FFFFF  
MAP 03C00000, 03FFEFFF  
FLASH\_MEM 02000000, 027FFFFF, NO, 32, 8  
FLASH\_MEM 03000000, 037FFFFF, NO, 32, 8

正しい設定 2 - 2 : 複数ブロックの設定

MAP 00000000, 003FFFFF  
MAP 00400000, 007FFFFF  
MAP 00800000, 00FFFFFF  
MAP 02000000, 02FFFFFF  
MAP 03000000, 037FFFFF  
MAP 03C00000, 03FFEFFF  
FLASH\_MEM2 00000000,0, 10000:128,32,8  
FLASH\_MEM2 01000000,0, 10000:128,32,8

誤った設定1 : MAP領域外

MAP 00000000, 033FFFFF  
FLASH\_MEM 03000000,037FFFF, NO, 32, 8 ←

誤った設定 2 : 複数MAPに架かる

MAP 00000000, 003FFFFF  
MAP 00400000, 007FFFFF  
MAP 00800000, 00FFFFFF  
MAP 02000000, 02FFFFFF  
MAP 03000000, 037FFFFF  
MAP 03C00000, 03FFEFFF  
FLASH\_MEM 02000000, 027FFFFF, NO, 32, 8  
FLASH\_MEM 02800000, 037FFFFF, NO, 32, 8 ←

## 8.3 書き込み動作

ダイアログコマンドやウィンドウコマンドで書き込みのための特別な操作は必要としません。通常の操作でフラッシュメモリを意識することなく書き込みできます。

但し、フラッシュメモリはその特性からメモリ内容を書き換えるためにセクタ単位で一度電気的に消去し、その後書き込みをする必要があります。書き込みには数秒の時間を要します。

したがって、PARTNER-VR5400-TPでは、処理速度の改善や書き込み回数の削減を目的にデバッガ内部に管理バッファを設けて、フラッシュメモリの書き込み動作を以下に示す仕組みで行います。

- 1 ) ダイアログコマンドやウィンドウコマンドで書き込みが発生
- 2 ) 該当するアドレスを含むセクタ内容を管理バッファに読み込む
- 3 ) 書き換えられたアドレスに相当する管理バッファの内容を更新する
- 4 ) 以下のタイミングで管理バッファをセクタ単位で実際に書き込む

ユーザプログラムの実行開始時

ロックでの書き込みを伴う処理の終了時

ZFコマンドで明示的に管理バッファをフラッシュした時

その他、内部処理で必要な場合

管理バッファのフラッシュ動作が開始するとコマンドウィンドウやステータスバーに以下のようなメッセージが表示されます。

### 【メッセージ表示例】

- 1 ) Flash ROM [42000000-42003FFF] Sect.Erase. /\* セクタ消去中 \*/  
表示されているセクタアドレスを消去中です。
- 2 ) Flash ROM [42000000-42003FFF] Sect.Write. /\* セクタ書き込み中 \*/  
表示されているセクタアドレスにデータを書き込み中です。
- 3 ) Flash ROM [42000000-420FFFFF] CHIP-Erase. /\* チップ全体の消去中 \*/  
表示されているアドレスのチップ全体を消去中です。
- 4 ) Flash ROM Device Error! /\* 消去・書き込み失敗 \*/  
消去または書き込みに失敗しました。環境ファイルの設定間違いや回路不良やデバイス不良が原因です。また、セクタ保護(ライトプロテクト)がなされている場合も書き込みに失敗します。

## 読み書き動作の技術的補足

---

フラッシュメモリの読み書き動作を理解するために、デバッガ内部に設けられた管理バッファについて説明します。

管理バッファはフラッシュメモリの容量と同じサイズのメモリが確保され、その状態をセクタ単位で次表のように管理します。

バッファ状態	意味
Invalid	バッファ内のデータは全て無効（バッファ未使用） 初期値
Valid	バッファ内にはフラッシュメモリと同じデータが格納
Dirty	バッファ内のデータが更新されフラッシュメモリには未書き込み

ダイアログコマンドやウィンドウコマンドでフラッシュメモリの領域に対する読み書きが行われると、次のように状態が遷移します。

**読み込み要求が発生した場合 :**

該当セクタの状態がDirtyの場合にのみ、管理バッファからデータを読み込みます。その他の状態では、直接フラッシュメモリから読み込みます。この要求では、状態遷移は発生しません。

**書き込み要求が発生した場合 :**

該当セクタの状態がInvalidの場合にフラッシュメモリのデータを管理バッファに読み込み、状態をValidに変更します。その後、データを管理バッファに書き込み状態をDirtyに変更します。

**フラッシュ要求が発生した場合 :**

Dirtyのセクタを対象にバッファからラッシュメモリへ書き込み、状態をValidに変更します。

**クリア要求が発生した場合 :**

セクタ状態を無条件にInvalidに変更します。この時にDirtyのセクタが存在してもフラッシュメモリへの書き込みは行いません。

**消去要求が発生した場合 :**

セクタ状態を無条件にInvalidに変更します。この時にDirtyのセクタが存在してもフラッシュメモリへの書き込みは行いません。

## 8.4 管理コマンド(ZF)

フラッシュメモリの制御や管理バッファの状態表示に用意されたZFコマンドについて説明します。

**ステータス :** 書式 ZF Stat[, アドレス]

指定されたアドレスが含まれるデバイスの管理バッファの状態を表示します。アドレスが省略された場合は、全てのデバイスに対する管理バッファの状態を表示します。

**イレーズ :** 書式 ZF Erase, アドレス

指定されたアドレスが含まれるデバイス全体の消去をします。アドレスを省略することはできません。

消去を実行すると、管理バッファのセクタ状態を無条件にInvalidにします。Dirtyのセクタが存在した場合、そのデータを破棄します。

**フラッシュ :** 書式 ZF Flush

Dirtyのセクタを対象に管理バッファからラッシュメモリへ書き込みを行います。書き込みが完了したセクタは、Validにします。

**クリア :** 書式 ZF Clear

管理バッファのセクタ状態を無条件にInvalidにします。

Dirtyのセクタが存在した場合、そのデータを破棄します。

# 付録

## レジスタ変数

PARTNER-VR5400-TPが使用可能なレジスタ変数は次の通りです。

レジスタ疑似変数	レジスタ
_r0,,,_r31	\$0 レジスタ,,,,\$31 レジスタ
_pc	PC レジスタ
_at	\$1 レジスタの別名
_v0,_v1	\$2,\$3 レジスタの別名
_a0,,,_a3	\$4,,,\$7 レジスタの別名
_t0,,,_t7	\$8,,,\$15 レジスタの別名
_s0,,,_s7	\$16,,,\$23 レジスタの別名
_t8,_t9	\$24,\$25 レジスタの別名
_k0,_k1	\$26,\$27 レジスタの別名
_gp	\$28 レジスタの別名
_sp	\$29 レジスタの別名
_s8	\$30 レジスタの別名
_ra	\$31 レジスタの別名

```
>while{ _R0!=_R1      /* R0レジスタとR1レジスタの内容比較 */
? T                      /* トレース実行コマンド */
?}                      /* マクロ終わり */
>
```

上記の例では\$0と\$1のレジスタ値が同じになるまでT(トレース)コマンドを実行します。レジスタと同名のシンボルがある場合にも、レジスタが優先されます。

## エラーメッセージ

PARTNER-VR5400-TPのエラーメッセージについて以下に説明します。

### コマンドエラー

指定されたコマンドがPARTNERの内部コマンドおよびマクロコマンドとして認識できません。

### アドレス指定が誤りです

アドレス入力部に不適当なアドレスが指定された場合や、スタートアドレスとエンドアドレスが逆転している場合に発生します。また、シンボル登録されていないシンボル名が使用された場合にも発生します。

### データ指定が誤りです

データ入力部に不適当なデータが指定された場合や、範囲外のデータが指定された場合に発生します。

### コマンドのフォーマットが不正です

コマンドの入力書式、パラメータの指定方法や数に間違이があります。

### Verifyエラー

メモリに正しくデータが書き込めませんでした。メモリの実装されていないアドレスやROM領域に書き込み動作をしたときに発生します。

### マクロ内でのマクロの定義はできません

マクロの定義(登録)は、PARTNERのコマンドレベルで行ってください。マクロコマンド内での定義はできません。

### マクロ内でのマクロ削除はできません

マクロの削除は、PARTNERのコマンドレベルで行ってください。マクロコマンド内の削除はできません。

### マクロ名が内部コマンドと重複しています

マクロコマンド定義を行おうとしたマクロコマンド名がPARTNERの内部コマンドと重複しています。違うマクロコマンド名で定義してください。

マクロバッファがいっぱいです

マクロコマンド定義のためのバッファがいっぱい、または定義したマクロコマンドの数が多すぎます。RPTSETUPを使用してマクロバッファサイズを広げてPARTNERを起動してください。

マクロ定義が正しく終了していません{ }

マクロコマンドの定義で中括弧( { } )の数が一致していません。

マクロが16レベル以上入れ子になっています

マクロの入れ子(ネスティング)は最大15レベルまで許されます。

指定された設定がありません

ブレークポイント、ウォッチ等の指定番号の設定が存在しません。

ブレークポイントが設定できません(最大15ポイント)

ブレークポイントを15点越えて設定しようとしました。ブレークポイントは最大15点まで指定できます。

ウォッチの設定ができません

ウォッチを16点越えて設定しようとしました。ウォッチ設定は最大16点まで指定できます。

ウォッチの指定が重複しています

Wコマンドでウォッチ登録しようとした内容はすでにウォッチ登録されています。

ファイルフォーマットが不正です

ロードしようとした実行形式のファイルのフォーマットが異常です。RPTSETUPで指定したデバッグモードとロードしようとするファイルが一致していません。

指定ファイルがありません

コマンド行で指定されたファイルが見つかりません。

指定ファイルがオープンできません

指定されたファイルがオープンできません。そのファイルが存在するか確認してください。また、ファイルをオープンしすぎている場合にも発生します。

**ファイルがクリエートできません**

ディスクがいっぱい、ファイルを同時にオープンしすぎています。不要なファイルをクローズするなどしてください。

**ディスクがいっぱいです**

指定されたディスクの空き容量が足りません。

**デバッグ情報がありません**

Lコマンドでプログラムをロードするとき、デバッグ情報がプログラムファイル内にありませんでした。

**デバッグ情報のフォーマットが不正です**

デバッグ情報のフォーマットが不正です。RPTSETUPで指定したデバッグモードとロードしようとするファイルを確認してください。

**デバッグ情報領域がいっぱいです(起動時の-Bオプション参照)**

デバッグ情報登録用のバッファに空き領域がありません。RPTSETUPで指定したデバッグ情報バッファサイズを拡大してください。

**ローカルシンボルの設定はできません**

既に登録されているローカルシンボルと同名のグローバルシンボルを登録しようとします。

**Cの変数が見つかりません**

?コマンドやVALコマンドで指定したCの変数が見つかりません。

**Cの式計算エラー**

?コマンドやVALコマンドでのCの式でエラーがあります。

**副作用のある演算子は使用できません**

?コマンドなどで副作用のある演算子(=, +=, -=等)は使用できません。副作用のある演算子はVALコマンドで使用してください。

### PARTNERの内部エラー

PARTNERの内部処理で不都合が生じた場合に表示されます。このエラーは通常起こらないように設計されています。このエラーが発生し、かつ再現性がある場合には、販売会社にご連絡ください。

### ターゲット実行中は使用できません

指定したコマンドは、ユーザプログラム実行中には使用できません。ブレークしてからコマンドを実行してください。

### モニタプログラムが正しく実行できません

CPUを制御するためのモニタプログラムが実行できないか、正常に終了しない状態です。初期化コマンドを実行してください。初期化コマンドの実行でこのエラーが表示される場合はターゲットハードウェアのCPUリセットした後、再度初期化コマンドを実行してください。



# 索引

.

.CFG ..... 65, 66

C

Code Warrior C ..... 58

  コンパイルとリンク方法 ..... 58

  使用例 ..... 59

F

FLASH\_MEM ..... 65

  環境設定 ..... 65

FLASH\_MEM2 ..... 66

  環境設定 ..... 66

G

GNU C ..... 56

  コンパイルとリンク方法 ..... 56

  使用例 ..... 56

Green Hills C ..... 52, 53

  コンパイルとリンク方法 ..... 53

  使用例 ..... 53

I

INIT.MCR ..... 9

M

MULTI2000での作成 ..... 54

P

PARTNER-VR5400-TPの環境設定 ..... 4

あ

エラーメッセージ ..... 74

か

起動 ..... 3

PARTNER-VR5400-TPの環境設定 ..... 4

  起動時のエラーメッセージ ..... 10

  初期設定コマンド ..... 8

  起動オプションの設定 ..... 4

  @オプション ..... 6

  -Bオプション ..... 4

  -Dオプション ..... 5

  -Eオプション ..... 5, 6

  -SDオプション ..... 5

  -TABオプション ..... 5

  -Xオプション ..... 6

  高級言語レベルデバッグ ..... 52

  Code Warrior C ..... 58

  GNU C ..... 56

  Green Hills C ..... 53

  コマンドリファレンス ..... 40

  機能別コマンド ..... 42

CPUリセット	42	ロギング/パッチ	49
Cのデータ参照/変更	48	コマンド解説の規約	41
I/Oポート入力/出力	46	コンフィギュレーションファイル	65, 66
アセンブル	46	<b>さ</b>	
イベントブレークポイント	44	ショートカットキー	
オプション設定	50	ヒストリウインドウ	23
環境設定コマンド	42	ブレークウインドウ	20
基数変更	49	レジスタウインドウ	17
キャッシュフラッシュ	47	初期設定コマンド	
コード表示	45	INIT.MCRでの初期化例	9
式の表示	49	初期化で使用するコマンド	8
システムコール	49	<b>た</b>	
システム制御	48	ダイアログコマンド	26
システムレジスタ表示/変更	45	ダイアログボックス	27
シンボル表示/設定	46	CPU環境設定	29
ハードウェアの初期化	42	TLBの設定	31
バックトレース	46	イベントブレークの設定	31
ファイルリード/ライト	43	イベント設定	28
フラッシュメモリ管理	51	エミュレーションROM設定	30
ブレークポイント	44	キャッシュフラッシュ	32
プログラム実行	44	トレースモード設定	27
プログラムロード	43	各種状態の設定	32
マクロコマンド	50	タイムタグについて	39
メモリ表示/変更	46	タイムタグの注意	39
文字列の定義(Cの式評価)	48	タイムタグ表示の種類	39
リアルタイムトレース	47	チャイルドウインドウ	15
レジスタ表示/変更	45		

ヒストリウインドウ	22
ブレークウインドウ	20
レジスタウインドウ	16
<b>は</b>	
はじめに	1
ヒストリウインドウ	
ショートカットキー	23
ローカルメニュー	23
ヒストリウインドウ	22
必要なセットアップ	1
必要なハードウェア	1
フラッシュメモリのサポート	
環境設定	65
管理コマンド	71
イレーズ	71
クリア	71
ステータス	71
フラッシュ	71
書き込み動作	69
対応デバイス	63
フラッシュメモリのサポート	62
ブレークウインドウ	20
ショートカットキー	20
マウス操作	21
ローカルメニュー	21
付録	72

エラーメッセージ	74
レジスタ変数	73

**ま**

## マウス操作

ブレークウインドウ	21
レジスタウインドウ	19
メインウインドウ補足	13
メニュー	14
実行メニュー	14

**ら**

リアルタイムトレース	34
トレースの概要	35
ディレイカウント	36
トリガ条件	37
トレースの開始条件	36
トレースの停止(終了)条件	37
強制ディレイモード	38
レジスタウインドウ	16
ショートカットキー	17
マウス操作	19
ローカルメニュー	18
レジスタ変数	73
ローカルメニュー	
ヒストリウインドウ	23
ブレークウインドウ	21
レジスタウインドウ	18

空白ページ

PARTNER ユーザーズ・マニュアル  
VRシリーズ  
『VR5400-TP個別編』

第二版 発行日 2002年1月  
株式会社 マイダス・ラボ

Copyright 2002 Midas lab Inc./Kyoto Micro Computer Co.,LTD.