

PARTNER ユーザーズ・マニュアル  
V800シリーズ  
『V850E2-TP個別編』

## ■ PARTNERのバージョンアップ

- 最新のPARTNER-V850E2-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-V850E2-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-V850E2-TP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

空白ページ

# 目次

はじめに.....	1
<b>1 起動.....</b>	<b>3</b>
1.1 PARTNER-V850E2-TPの環境設定.....	4
1.2 初期設定コマンド.....	7
1.3 起動時のエラーメッセージ.....	9
<b>2 チャイルドウィンドウ .....</b>	<b>11</b>
2.1 レジスタウィンドウ.....	12
2.1.1 レジスタウィンドウのショートカットキー .....	13
2.1.2 レジスタウィンドウのローカルメニュー .....	13
2.1.3 レジスタウィンドウでのマウス操作.....	14
2.2 ブレークウィンドウ.....	15
2.2.1 ブレークウィンドウのショートカットキー .....	15
2.2.2 ブレークウィンドウのローカルメニュー .....	16
2.2.3 ブレークウィンドウでのマウス操作.....	16
2.3 ヒストリウィンドウ.....	17
2.3.1 ヒストリウィンドウのショートカットキー .....	18
2.3.2 ヒストリウィンドウのローカルメニュー .....	18
2.4 I/Oウィンドウ .....	20
2.4.1 I/Oウィンドウのショートカットキー .....	20
2.4.2 I/Oウィンドウのローカルメニュー .....	21
2.4.3 I/Oウィンドウでのマウス操作 .....	21

## **3 ダイアログコマンド.....23**

<b>3.1 ダイアログボックス.....24</b>
3.1.1 アクセスブレークの設定ダイアログボックス .....24
3.1.2 トレース環境条件設定ダイアログボックス .....25
3.1.3 トレース制御条件設定ダイアログボックス .....27
3.1.4 トレースアドレス設定ダイアログボックス .....28
3.1.5 イベント設定ダイアログボックス .....29
3.1.6 イベントポイント設定ダイアログボックス .....30
3.1.7 CPU環境設定ダイアログボックス .....31
3.1.8 エミュレーションROM設定ダイアログ .....32
3.1.9 各種状態の設定ダイアログボックス .....33

## **4 リアルタイムトレース.....35**

<b>4.1 トレースの概要.....36</b>
4.1.1 ディレイカウント .....37
4.1.2 通常/完全モード .....37
4.1.3 トレースの開始条件 .....38
4.1.4 トリガ条件 .....39
4.1.5 トレースの停止条件 .....39
4.1.6 トレースの終了 .....40
4.1.7 強制ディレイモード .....40

### **4.2 タイムタグについて.....41**

4.2.1 タイムタグ表示の種類 .....41
4.2.2 タイムタグの注意 .....41

## **5 コマンドリファレンス.....43**

### **5.1 コマンド解説の規約.....44**

<b>5.2 機能別コマンド .....</b>	<b>45</b>
5.2.1 ハードウェアの初期化 .....	45
5.2.2 CPUリセット .....	45
5.2.3 環境設定コマンド .....	45
5.2.4 プログラムロード .....	46
5.2.5 ファイルリード/ライト .....	46
5.2.6 プログラム実行 .....	46
5.2.7 ブレークポイント .....	46
5.2.8 アクセスブレークポイント .....	46
5.2.9 コード表示 .....	47
5.2.10 レジスタ表示/変更 .....	47
5.2.11 システムレジスタ表示/変更 .....	47
5.2.12 メモリ表示/変更 .....	48
5.2.13 I/Oポート入力/出力 .....	48
5.2.14 シンボル表示/設定 .....	48
5.2.15 アセンブル .....	48
5.2.16 バックトレース .....	48
5.2.17 リアルタイムトレース .....	49
5.2.18 イベント .....	50
5.2.19 システムコール .....	50
5.2.20 式の表示 .....	50
5.2.21 Cのデータ参照/変更 .....	51
5.2.22 文字列の定義(Cの式評価) .....	51
5.2.23 システム制御 .....	51
5.2.24 基数変更 .....	52
5.2.25 ロギング/バッチ .....	52
5.2.26 オプション設定 .....	52

5.2.27 マクロコマンド .....	53
5.2.28 フラッシュメモリ管理 .....	53
<b>6 高級言語レベルデバッグ .....</b>	<b>55</b>
<b>6.1 GreenHills C.....</b>	<b>56</b>
6.1.1 コンパイルとリンク方法 .....	56
6.1.2 使用例 .....	56
6.1.3 MULTI2000での作成 .....	57
<b>6.2 GNU C(exeGCC) .....</b>	<b>59</b>
6.2.1 コンパイルとリンク方法 .....	59
6.2.2 使用例 .....	59
<b>6.3 NEC C .....</b>	<b>62</b>
6.3.1 コンパイルとリンク方法 .....	62
6.3.2 使用例 .....	63
<b>7 フラッシュメモリのサポート .....</b>	<b>65</b>
<b>7.1 対応デバイス.....</b>	<b>66</b>
<b>7.2 環境設定 .....</b>	<b>68</b>
<b>7.3 書き込み動作 .....</b>	<b>72</b>
<b>7.4 管理コマンド(ZF).....</b>	<b>74</b>
<b>付録 .....</b>	<b>75</b>
<b>レジスタ変数.....</b>	<b>76</b>
<b>エラーメッセージ .....</b>	<b>77</b>
<b>索引 .....</b>	<b>81</b>



空白ページ

# はじめに

このマニュアルは、『PARTNER ユーザーズ・マニュアルV800シリーズ共通編』で説明されていない『PARTNER-V850E2-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V800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。

## 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-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』、『10Base-TのLAN-BOX』、『PCIバス用インターフェースキット』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を確認してください。

空白ページ

# 1 起動

PARTNER-V850E2-TPを起動するには、PARTNER-V850E2-TPの環境設定が必要です。

この章では、PARTNER-V850E2-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-V850E2-TPの環境設定

### 起動オプションの設定



PARTNER-V850E2-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)



オプション文字列

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-V850E2-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(RPTV850E2T.KPJ)に書き込まれます。PARTNER-V850E2-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-V850E2-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(RPTV850E2T.CFG)、プロジェクトファイル(RPTV850E2T.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-V850E2-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指定されます。

### -Eオプション

コードウィンドウで表示/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複数指定できます。

### -SDオプション

コードウィ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

コードウィ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

アセンブラまたは各種Cのデバッグモードの指定をします。PARTNER-V850E2-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あるいは特定の機能が使用できることがあります。各処理系の使用方法は、『6 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)を指定します。このモードでPARTNER-V850E2-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 のデバッグモードを指定します。デバッグ情報は.OUTファイルから読み込みます。

**-XG**

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

**-XN**

NEC Cのデバッグモードを指定します。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:\$PARTNER\$BIN\$RPTV850E2T -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(RPTV850E2T.CFG)の内容以外で、PARTNER-V850E2-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(42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-V850E2-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。
ROM1 ~ ROM4	エミュレーションROMを設定します。
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-V850E2-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-V850E2-TP、ICEを初期化します。  
env !nmi0 !nmi1  
NMI0,NMII信号をマスクする、に設定します。  
rom1 100000 20000 1m 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、1M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100000H～11FFFFH  
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.outを読み込みます。
```

## 1.3 起動時のエラーメッセージ

PARTNER-V850E2-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。

## 起動時のエラーメッセージ



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 チャイルドウィンドウ

この章では、PARTNER-V850E2-TPのチャイルドウィンドウについて、表示や操作方法を説明します。

## 2.1 レジスタウィンドウ

レジスタウィンドウは、プログラムレジスタ(図 2)、システムレジスタ(図 3)を表示します。

レジスタ	
R0	00000000
R1	00000000
R2(hp)	00000000
R3(sp)	00000000
R4(gp)	00000000
R5(tp)	00000000
R6	00000000
R7	00000000
R8	00000000
R9	00000000
R10	00000000
R11	00000000

図 2 プログラムレジスタ表示例

レジスタ	
EIPC(0)	00000000
EIPSW(1)	00000000
FEPC(2)	00000000
FEPSW(3)	00000000
ECR(4)	00000000
PSW(5)	00000000
CTPC(16)	00000000
CTPSW(17)	00000000
DBPC(18)	00000000
DBPSW(19)	00000000
CTBP(20)	00000000

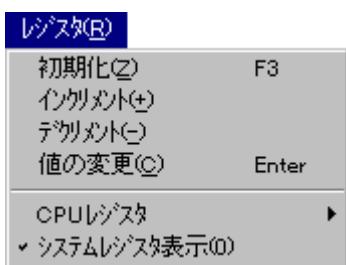
図 3 システムレジスタ表示例

### 2.1.1 レジスタウィンドウのショートカットキー

レジスタウィ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ィ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。

### 2.1.2 レジスタウィンドウのローカルメニュー

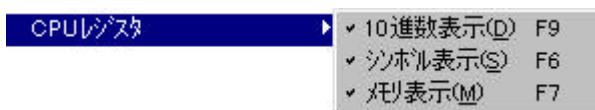


[レジスタ]メニューにはレジスタウィンドウの制御に関するコマンドが収められています。

メニュー項目	機能
初期化(Shift+F3)	カーソル位置のレジスタの値を0にします。
インクリメント(+)	カーソル位置のレジスタの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の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の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	プログラムレジスタ値の表示 / 非表示の切り替えをします。
システムレジスタ表示(O)	システムレジスタ値の表示 / 非表示の切り替えをします。

## レジスタウィンドウ

### プログラムレジスタ



メニュー項目	機能
10進数表示(D)	レジスタの値を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)	レジスタの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)	レジスタの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。

### 2.1.3 レジスタウィ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。

## 2.2 ブレークウィンドウ

ブレークウィンドウは、現在設定されているブレークポイントを表示します(図 4)。

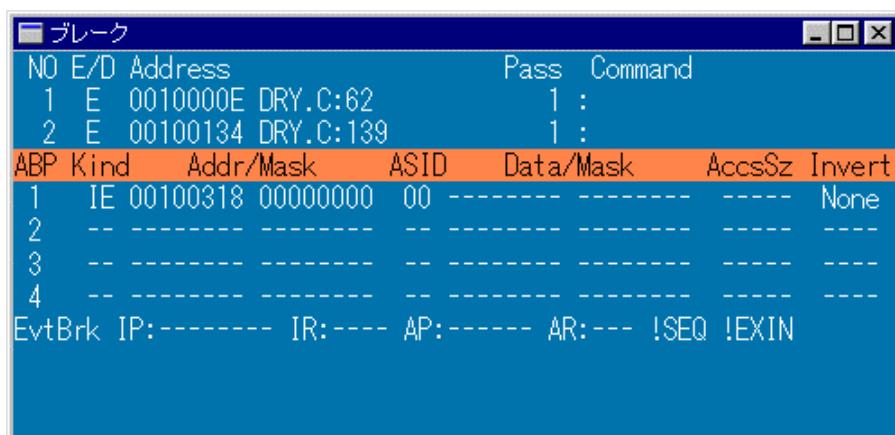


図 4 ブレークウィンドウ

### 2.2.1 ブレークウィンドウのショートカットキー

ブレークウィ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ィ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。
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。

### 2.2.2 ブレークウィンドウのローカルメニュー

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(B)...	F7
イベント設定(E)...	
イベントポイント(P)...	

[ブレーク]メニューにはブレークウィ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。
アクセスブレーク設定(B)	アクセス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。(『3.1.1 アクセスブレークの設定ダイアログボックス』24頁参照)
イベント設定(E)	イベントの条件を設定するダイアログボックスを表示します。(『3.1.5 イベント設定ダイアログボックス』29頁参照)
イベントポイント(P)	イベントポイントを設定するダイアログボックスを表示します。(『3.1.6 イベントポイント設定ダイアログボックス』30頁参照)

### 2.2.3 ブレークウィ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。

## 2.3 ヒストリウインドウ

ヒストリウインドウは、リアルタイムトレースメモリの内容を2通りの方法で表示します。内容を解析して逆アセンブル形式で表示(図5)する方法と、解析せずに内容を16進ダンプ形式で表示(図6)する方法があります。

Cycle ID:Address	Data/Code	Instruction	EXT	Stat
RAM_TEST.C:0100:		loop = 1 ;		
-1267 00:0000124C	EA01	MOV 1,r29 loop	1111	JMPD
		time= 000,000,000,034.1us		
RAM_TEST.C:0102:		fprintf(stdout, "%nHit Retrun Key >		
00:0000124E	0630B0C0	MOV -4F40,r16	1111	
00:00001254	8E100010	ADDI 10,r16,r17	1111	
00:00001258	3011	MOV r17,r6	1111	
00:0000125A	06274D40	MOV 4D40,r7	1111	
-1265 00:00001260	FF800BFE	JALR 1E5E fprintf ,r311111 JMP\$		
		time= 000,000,000,034.4us		

図5 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
-1267	12	82	0C	42	10	00	00									
-1266	02	00														
-1265	12	A0	50	62	10	00	00									
-1264	02	00														
-1263	12	85	0E	5E	10	00	00									
-1262	02	00														
-1261	12	A0	56	8E	10	00	00									
-1260	02	00														
-1259	12	85	0C	D0	20	00	00									
-1258	02	00														

図6 ヒストリ(ダンプ表示)

### 2.3.1 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。

### 2.3.2 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。
ジャンプ(J)	表示サイクル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(T)	ヒストリの実行 / 停止を切り替えます。
イベント設定(E)	イベントの条件を設定するダイアログボックスを表示します。(『3.1.5 イベント設定ダイアログボックス』29頁参照)
イベントポイント(P)	イベントポイントを設定するダイアログボックスを表示します。(『3.1.6 イベントポイント設定ダイアログボックス』30頁参照)

トレースアドレス(I)	トリガアドレス、データトレースアドレスを設定するダイアログボックスを表示します。(『3.1.4 トレースアドレス設定ダイアログボックス』28頁参照)
トレース環境(M)	トレース環境条件を設定するダイアログボックスを表示します。(『3.1.2 トレース環境条件設定ダイアログボックス』25頁参照)
トレース条件(B)	トレース制御条件を設定するダイアログボックスを表示します。(『3.1.3 トレース制御条件設定ダイアログボックス』27頁参照)

## 2.4 I/Oウィンドウ

I/Oウィンドウは、指定された内部周辺I/Oレジスタの表示と設定を行います(図 7)。

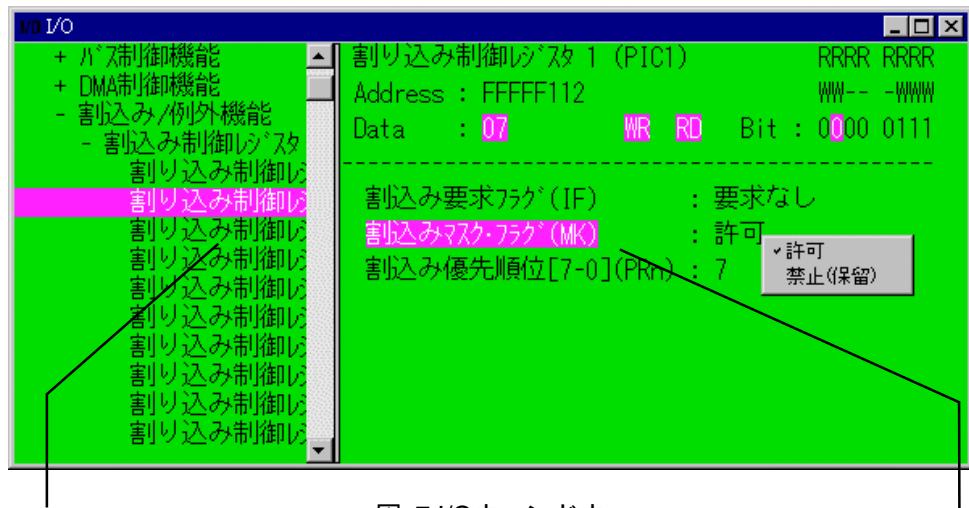


図 7 I/Oウィンドウ

アイテムウィンドウ

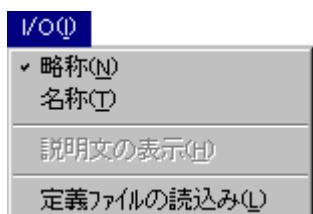
データウィンドウ

### 2.4.1 I/Oウィンドウのショートカットキー

I/Oウィ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I/Oウィ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
Tab	アイテム / データウィンドウの切り替えをします。
Enter	アイテムウィンドウの場合、階層構造の表示 / 非表示をします。 データウィンドウの場合、一つ下階層に移動します。
ESC・BS	データウィンドウの場合、一つ上階層に移動します。
	データウィンドウの場合、設定値を変更します。

## 2.4.2 I/Oウィンドウのローカルメニュー



[I/O]メニューにはI/Oウィ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
映射(N)	レジスタ名を表示します。
名称(I)	レジスタの機能名称を表示します。
説明文の表示(H)	レジスタの説明を表示します。
定義ファイルの読み込み(L)	定義ファイル(RPTxxxx.IO)からレジスタの情報を読み込みます。

## 2.4.3 I/Oウィンドウでのマウス操作

設定値の変更 :

変更したいデータの内容をクリックするとポップアップメニューが表示されます。メニュー項目を選択して値の変更、参照をします。

ポップアップメニューは、レジスタによって項目が異なります。

ローカルメニューの表示 :

I/Oウィンドウのアイテムウィンドウ内で右ボタンをクリックします。

クリップボードへのコピー :

I/Oウィンドウのアイテムウィンドウ内で、任意の位置で左ボタンを押し、ボタンを押したまま選択領域を指定します。ボタンを放すと選択されていた領域がクリップボードにコピーされます。

空白ページ

### 3 ダイアログコマンド

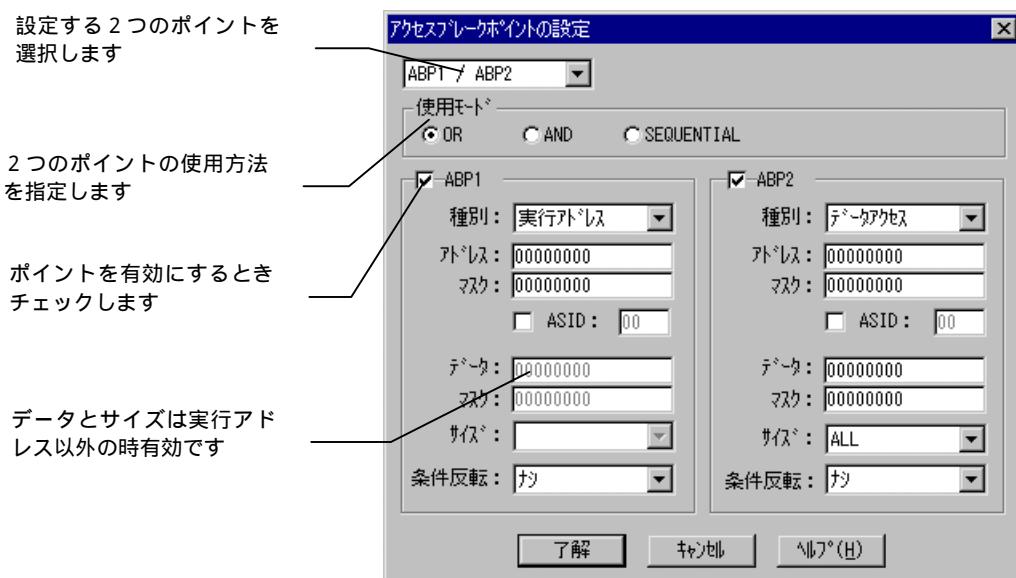
PARTNER-V850E2-TPには、ダイアログボックスやツールバーによってデバッグを行う方法と、コマンドウィンドウにコマンドを入力してデバッグを行う方法の2通りあります。この章では、PARTNER-V850E2-TPに依存したダイアログボックスについて説明します。

## 3.1 ダイアログボックス

### 3.1.1 アクセスブレークの設定ダイアログボックス

[ブレーク]-[アクセスブレーク]

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



アクセスブレークポイントは 4 ポイントあり、それぞれ単独またはABP1と2、ABP3と4の組み合わせで使用できます。

**使用モード**には次の 3 通りがあります。AFTERとANDのときは、両ポイントが有効である必要があります。

**OR** : それぞれを単独で使用します

**AFTER** : ABP2(4)の後にABP1(3)の発生でブレークします

**AND** : 同時に発生したときにブレークします。

この場合は、ABP1(3),ABP2(4)の種別は同じにしてください。

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。マスクでアドレスのマスク指定をすることができます。

ASIDを指定しない場合は、ASID値は比較されません。

データは、データ値条件を指定します。マスクでデータのマスク指定することができます。

サイズは、データのアクセスサイズ条件を指定します。

条件反転は、アドレス値、データ値の一致条件を反転する場合に指定します。

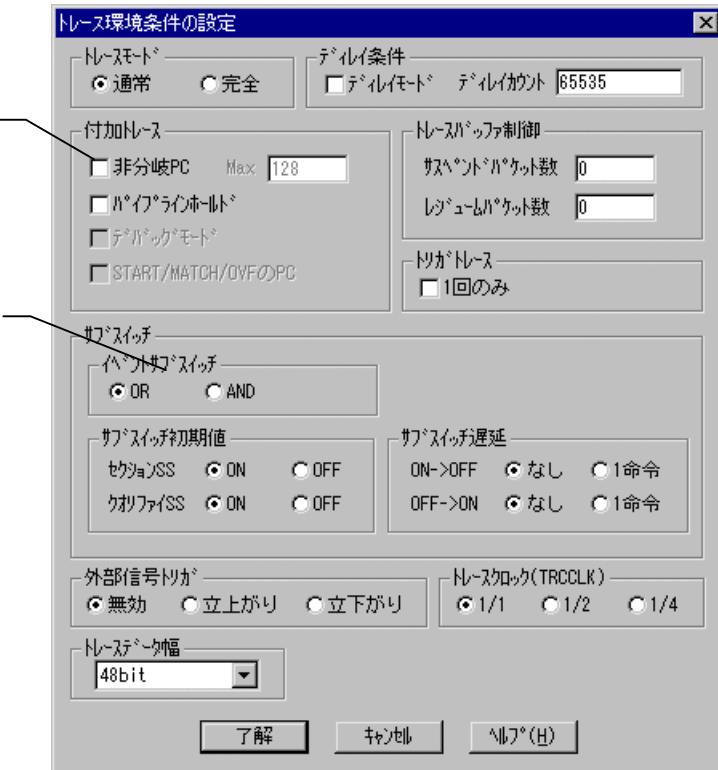
### 3.1.2 トレース環境条件設定ダイアログボックス

[ヒストリ]-[トレース環境]

[トレース環境条件設定]ダイアログボックスは、リアルタイムトレースの環境条件を設定します。

トレースデータとして取り込む付加情報を選択します。

サブスイッチ（トレース開始 / 停止）の制御を選択します。サブスイッチONで開始、OFFで停止です。



**トレースモード**は、トレースの取り込みモードを指定します。

**通常モード**はリアルタイムに実行内容をトレースしますが、CPU内のトレースバッファがフルになった場合に、トレースデータの取りこぼし（オーバーフロー）が発生します。

**完全モード**のときは、CPU内のトレースバッファがフルになった場合、トレースデータの取りこぼしがないようにするためにCPUの実行を一時停止します。その後、自動的に実行を再開します。

両モードでのトレースバッファへの書き込みの停止 / 再開は、トレースバッファ制御のパケット数で変更できます。

**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0～262143(0x3FFFF)の範囲で指定します。

**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。このモード中は、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）と外部入力端子、イベントユニットからのトリガポイントのOR条件です。

**付加トレース**は、トレースデータとして取り込む付加情報を選択します。  
**非分岐**PCIは、指定した命令数の命令が実行される間、トレース制御条件で指定したステータス条件が発生しない場合に、非分岐のトレースデータを取り込みます。  
**パイプラインホールド**は、完全モードのときに、CPUの実行を停止したことを示すトレースデータを取り込みます。

**トレースバッファ制御**は、トレースバッファへの書き込みの停止（サスPEND）／再開（レジューム）制御のパケット数を指定します。

**完全モード**は、トレースバッファにサスPENDパケット数以上のトレースデータが貯まつたら、CPU実行を一時停止します。レジュームパケット数以下になった場合にCPU実行を再開します。

**通常モード**では、サスPENDパケット数は無効です。31パケット以上のデータが貯まるとオーバーフローになります。レジュームパケット数以下になるとトレースの取り込みを再開します。

**トリガトレース**は、トリガポイントのトレースデータ取り込みを最初の発生1回／発生毎に行うかを選択します。

**サブスイッチ**は、CPUからのトレースデータの出力開始／停止の各制御を指定します。サブスイッチONで開始、OFFで停止です。

**イベントサブスイッチ**は、イベントユニットのセクションON/OFF要因とクオリファイ要因（区間内外）からのサブスイッチの生成条件を指定します。

クオリファイ要因のサブスイッチは、区間内でON、区間外でOFFになります。

**サブスイッチ初期値**は、トレース開始時におけるイベントユニットのセクション要因とクオリファイ要因のサブスイッチ状態の初期値を指定します。

**サブスイッチ遅延**は、サブスイッチが切り替わる遅延タイミングを指定します。

**外部信号**は、トレースの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。

**トレースクロック**は、トレースクロックの分周比を指定します。

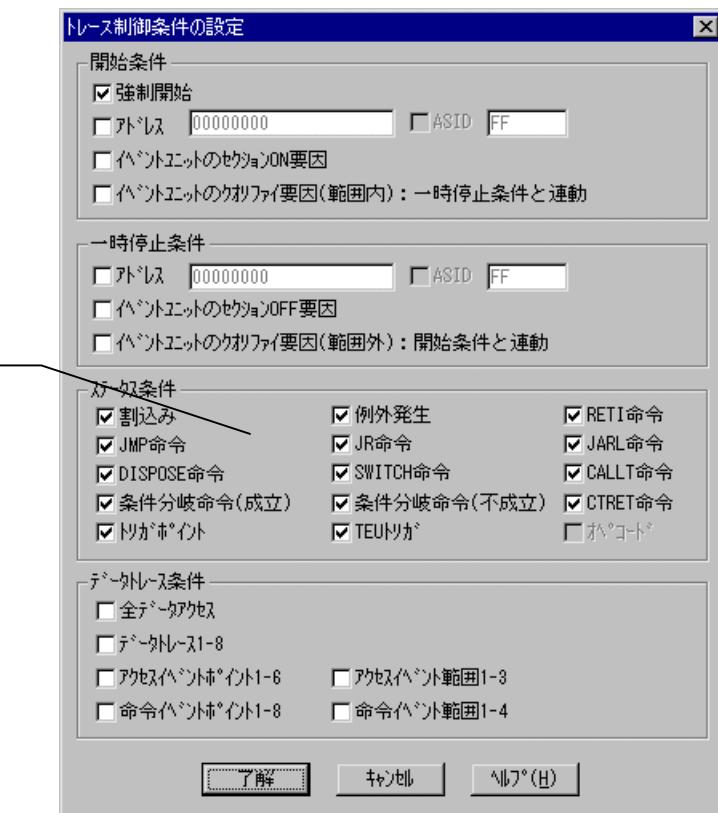
**トレースデータ幅**は、CPUの出力するトレースデータのビット幅を指定します。

### 3.1.3 トレース制御条件設定ダイアログボックス

[ヒストリ]-[トレース制御]

[トレース制御条件設定]ダイアログボックスでは、リアルタイムトレースの制御条件を設定します。

正しい解析のためすべてを  
チェックすることを推奨しま  
す



**開始条件**は、トレースデータの取り込み開始の条件を指定します。

実行アドレスとイベントユニットのトレース開始要因を選択します。

**強制開始**のときは、CPUを実行したときから取り込みが開始されます。

イベントユニットの開始条件を使用する場合は、『3.1.5 イベント設定ダイアログボックス』(29頁参照)、『3.1.6 イベントポイント設定ダイアログボックス』(30頁参照)でそれらの条件を設定する必要があります。

イベントユニットのクオリファイ要因を使用する場合は、区間内のトレースデータが取り込まれます。この場合は、一時停止条件のクオリファイ要因も選択されます。

**一時停止条件**は、トレースデータ取り込みの中止条件を指定します。

実行アドレスとイベントユニットのトレース停止要因を選択します。

中断後、開始条件が設定されている場合は、その条件によって再度トレースを開始します。

イベントユニットの開始条件を使用する場合は、『3.1.5 イベント設定ダイアログボックス』(29頁参照)、『3.1.6 イベントポイント設定ダイアログボックス』(30

頁参照)でそれらの条件を設定する必要があります。

イベントユニットのクオリファイ要因を使用する場合は、区間外のとき取り込みが停止します。この場合は、開始条件のクオリファイ要因も選択されます。

**ステータス条件**は、トレースデータとして取り込む実行アドレスのステータス条件とトリガ条件を指定します。TEUトリガはイベントユニットでのトリガ検出です。

これらの条件が少ない場合にはトレース解析ができないことがあります。そのため、トレース結果が正しく表示されない可能性があります。通常は、全ステータス条件を指定してください。

トリガポイントは『3.1.4 トレースアドレス設定ダイアログボックス』(28頁参照)で設定します。

**データトレース条件**は、トレースデータとして取り込むデータトレースの条件種別を指定します。

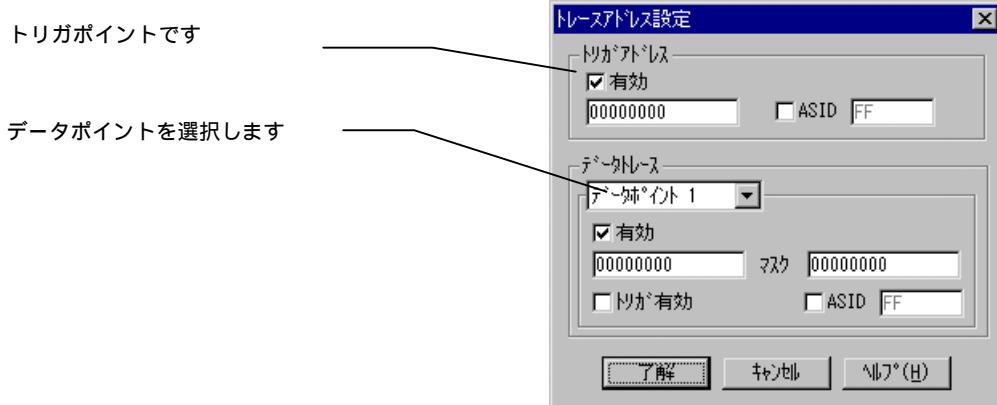
**全データアクセス**を選択した場合は、他のデータトレース条件は選択できません。この場合はデータアクセスすべてがトレースの対象となります。

データトレース 1 ~ 8 は『3.1.4 トレースアドレス設定ダイアログボックス』(28頁参照)で、各イベントポイント条件は『3.1.5 イベント設定ダイアログボックス』(29頁参照)、『3.1.6 イベントポイント設定ダイアログボックス』(30頁参照)で設定します。

### 3.1.4 トレースアドレス設定ダイアログボックス

[ヒストリ]-[トレースアドレス設定]

[トレースアドレス設定]ダイアログボックスは、リアルタイムトレースで使用するアドレスの条件を設定します。



**データトレース**は、データポイント 1 ~ 8までの設定ができます。

アクセスアドレスは、ビット 0 ~ 7のみをマスク指定することができます。

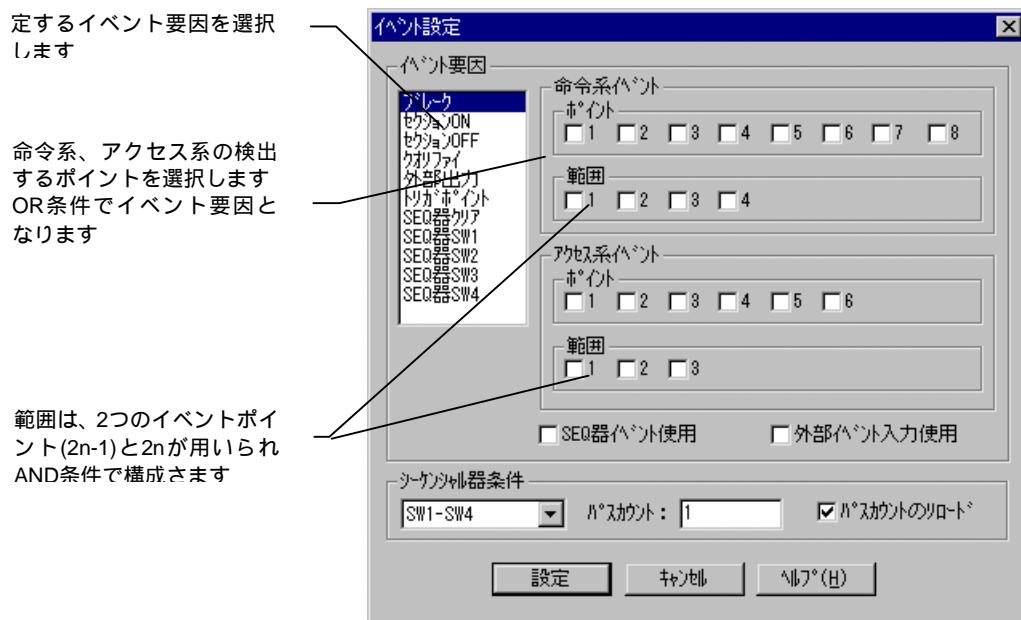
ASIDを指定しない場合は、ASID値は比較されません。

**トリガ有効**をチェックすることでトリガポイントに指定することができます。

### 3.1.5 イベント設定ダイアログボックス

[ブレーク]-[イベント設定]、[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は、各種イベント要因の条件を設定します。



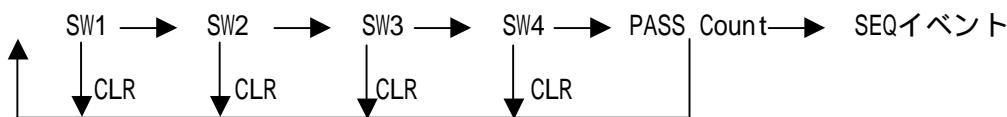
**範囲イベント**は、2つのポイントのAND条件で構成されます。そのため、2つのポイント条件が設定されている必要があります。ただし、アクセス系の場合、データ条件はポイント $2n-1$ の条件のみ有効です。

イベントポイントの条件は『3.1.6 イベントポイント設定ダイアログボックス』(30頁参照)で設定します。

イベント要因の**SEQ器イベント使用**は、シーケンシャル器の条件を設定します。

**シーケンシャル器条件**は、使用するスイッチの段数とパスカウントを指定します。シーケンシャル器は、4つのイベントが順番に発生したことを検出するものです。SW1～SW4のON条件が順番に発生したときに(SW1->SW2->SW3->SW4)、シーケンシャル器イベントが発生します。パスカウントは、順番に発生したときにカウントされ、指定回数の時にシーケンシャル器イベントになります。クリア要因(SEQ器CLR)によって、すべてのスイッチがクリアされます。

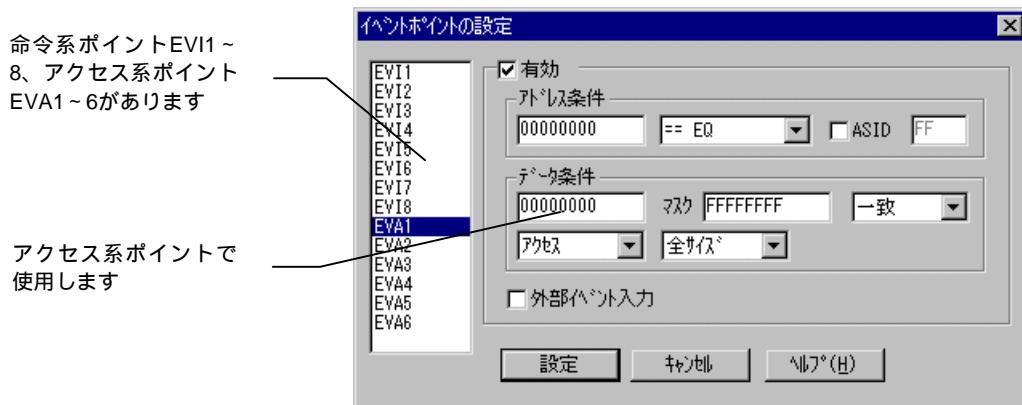
**パスカウント**は、1～0xFFFF (4095) の値を設定できます。指定段数のスイッチが順番にONになったときカウントされ、指定回数発生したときにイベントが発生します。



### 3.1.6 イベントポイント設定ダイアログボックス

[ブレーク]-[イベントポイント設定]、[ヒストリ]-[イベントポイント設定]

[イベントポイント設定]ダイアログボックスは、イベント要因の条件に使用される、イベントポイントの条件を設定します。



**アドレス条件**は、アドレス値と比較条件を指定します。命令系イベントポイントでは実行アドレス、アクセス系イベントポイントはデータアクセスアドレスを指定します。比較条件は次の関係があります。

設定アドレス == 実行 / データアドレス : EQ  
 設定アドレス > 実行 / データアドレス : LT  
 設定アドレス < 実行 / データアドレス : GT  
 設定アドレス != 実行 / データアドレス : NE  
 設定アドレス <= 実行 / データアドレス : GE  
 設定アドレス >= 実行 / データアドレス : LE

ONは比較しません（常にアドレス条件はON）。

これらのイベント検出は、命令系は実行後、アクセス系はアクセス後です。

ASIDを指定しない場合は、ASID値は比較されません。

**データ条件**は、アクセス系イベントポイントでの、データ値と次のデータ条件を指定します。データ値はマスク指定できます。

データ値の一致/不一致条件、データアクセスの種別、そしてデータのアクセスサイズを指定します。

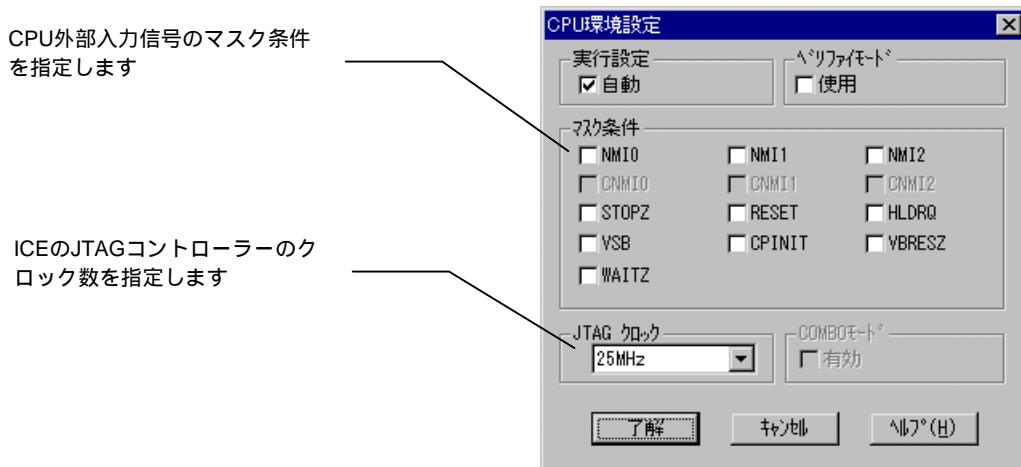
外部イベント入力は、OR条件です。

イベントポイントを使用するイベント要因の設定は、『3.1.5 イベント設定ダイアログボックス』(29頁)を参照してください。

### 3.1.7 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は、CPUの動作条件を設定します。



**実行設定**は、PATRNER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、行わないかの選択をします。

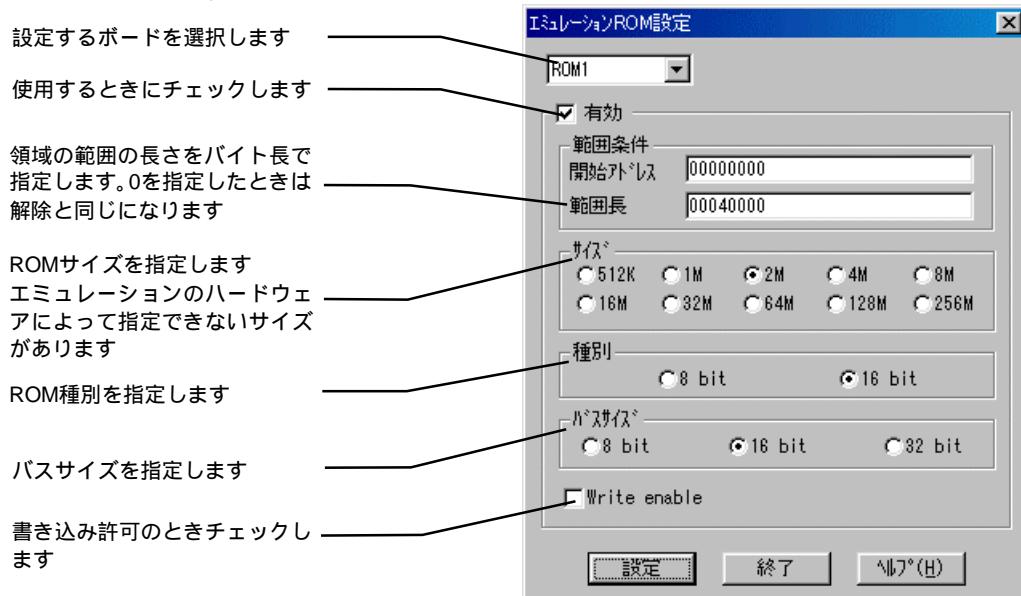
**ペリファイモード**は、メモリ書き込みを行う操作やコマンドのときに、書き込み後にメモリペリファイを行なうか、行わないかの選択をします。

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。

### 3.1.8 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の最下位のアドレスを指定し、**範囲長**は、エミュレートするROMの容量をバイト数(4バイトの境界)で入力してください。  
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、サイズの512kと種別の8bit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。

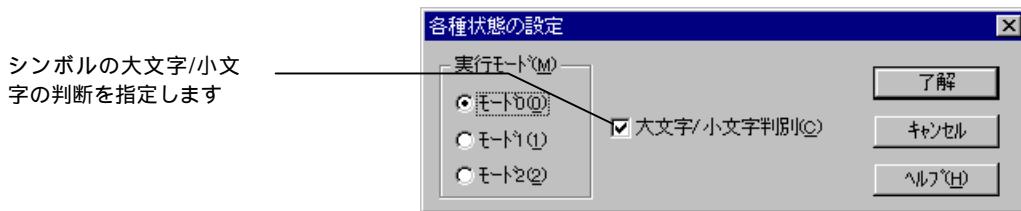
バスサイズにより複数枚のボードを使用します。そのため、**ボードの選択**と**バスサイズ**に次の関係があります。記載がないバスサイズは設定できません。

ボード選択	バスサイズ	ボードスロット位置	ボード選択	バスサイズ	ボードスロット位置
ボード1	8 bit	#3	ボード3	8 bit	#5
	16 bit	#3		16 bit	#5
	32 bit	#3+#4		32 bit	#5+#6
ボード2	8 bit	#4	ボード4	8 bit	#6
	16 bit	#4		16 bit	#6

### 3.1.9 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、PARTNER-V850E2-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-V850E2-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。

空白ページ

## 4 リアルタイムトレース

PARTNER-V850E2-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。

## 4.1 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。

トレースモード、トレース開始条件、トリガ条件、一時停止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 8、図 9のようになっています。

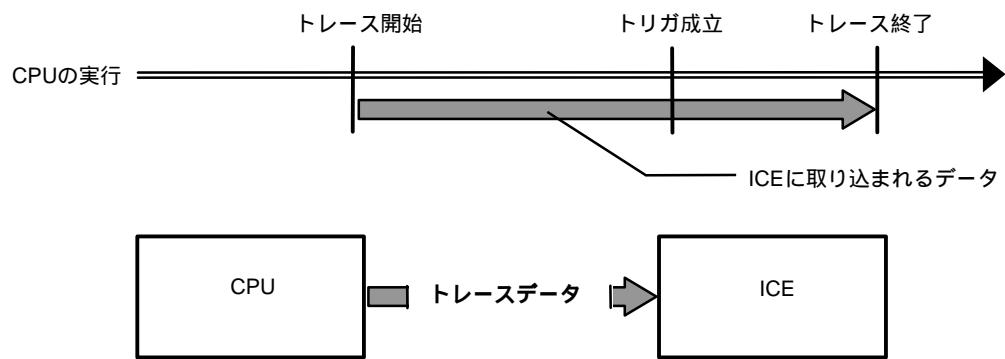


図 8 トレースデータ取り込みの流れ

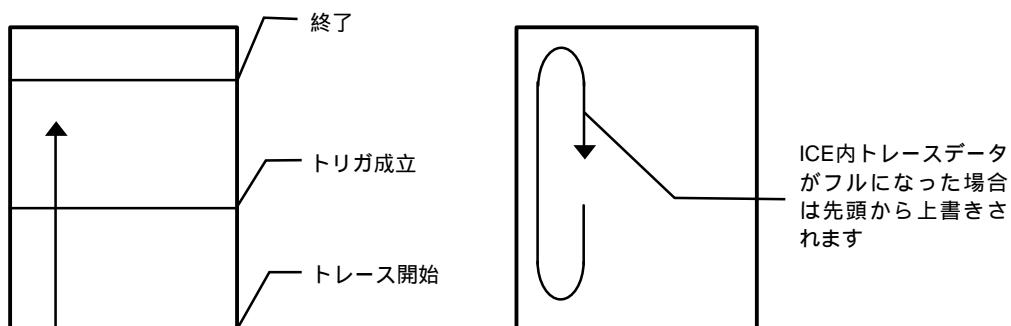


図 9 ICE内のトレースデータ

### 4.1.1 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 10)。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。

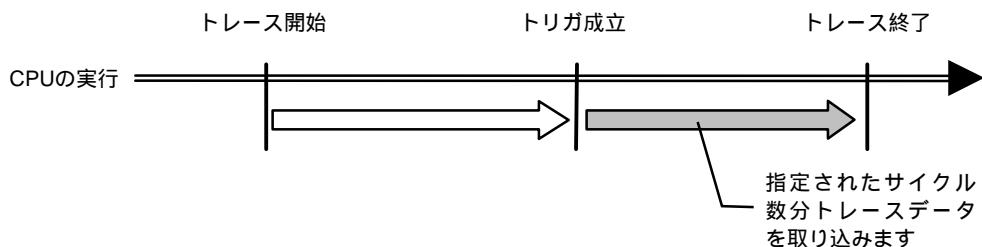


図 10 ディレイカウントの流れ

### 4.1.2 通常/完全モード

通常モードは、リアルタイムに実行内容をトレースするモードです。そのため、CPU内のトレースバッファがフルになった場合、トレースデータの取りこぼしが発生することがあります(図 11)。

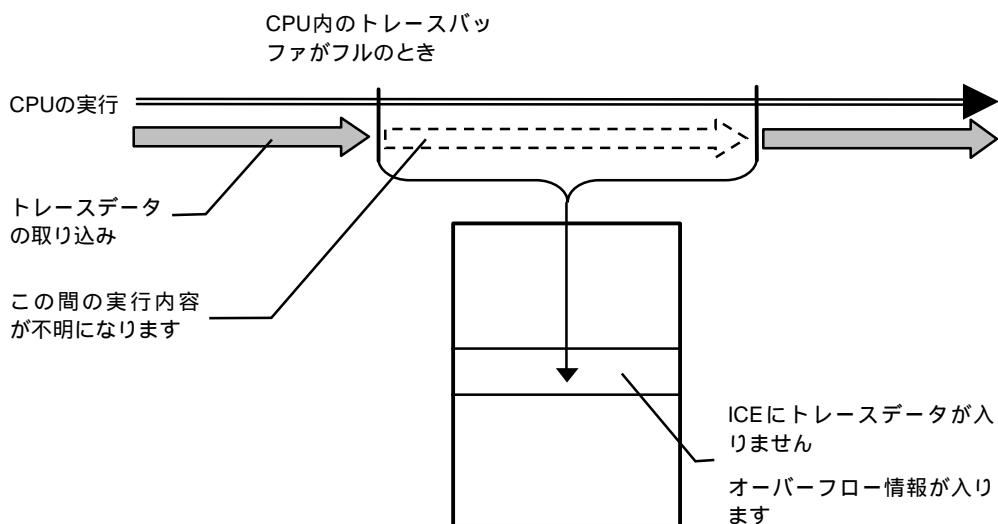


図 11 通常モード

完全モードは、トレースデータの取りこぼしがないようにするモードです。そのためCPU内のトレースバッファがフルになった場合に、CPUの実行を一時停止します。その後自動的に再開します(図12)。

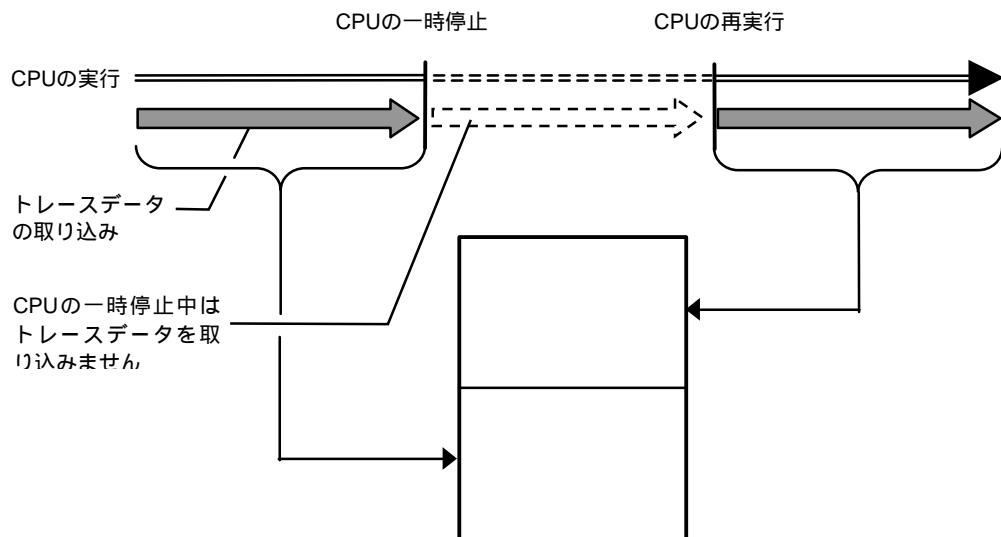


図 12 完全モード

#### 4.1.3 トレースの開始条件

開始条件は、トレースデータの取り込みを開始する条件です(図13)。

イベントユニットのトレース区間要因を使用するときは、開始条件の使用はできません。

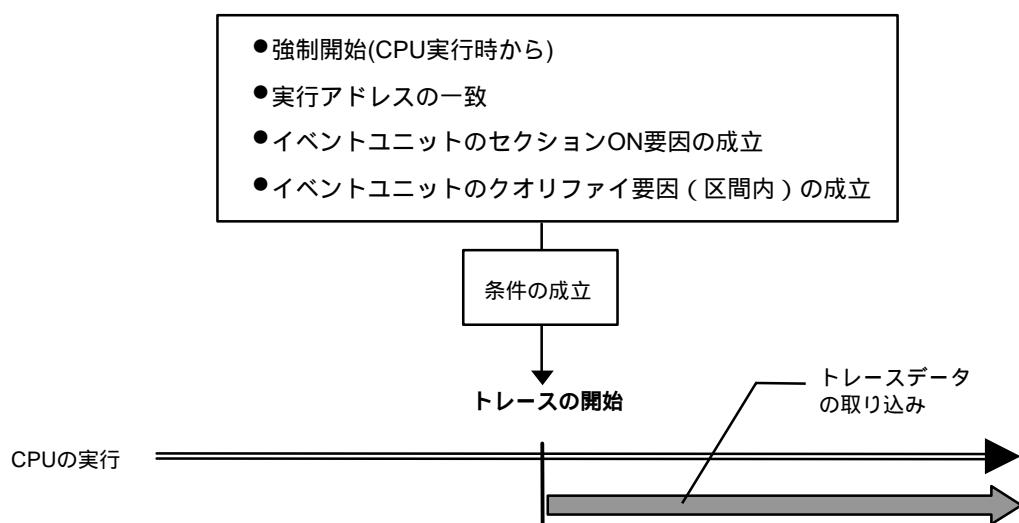
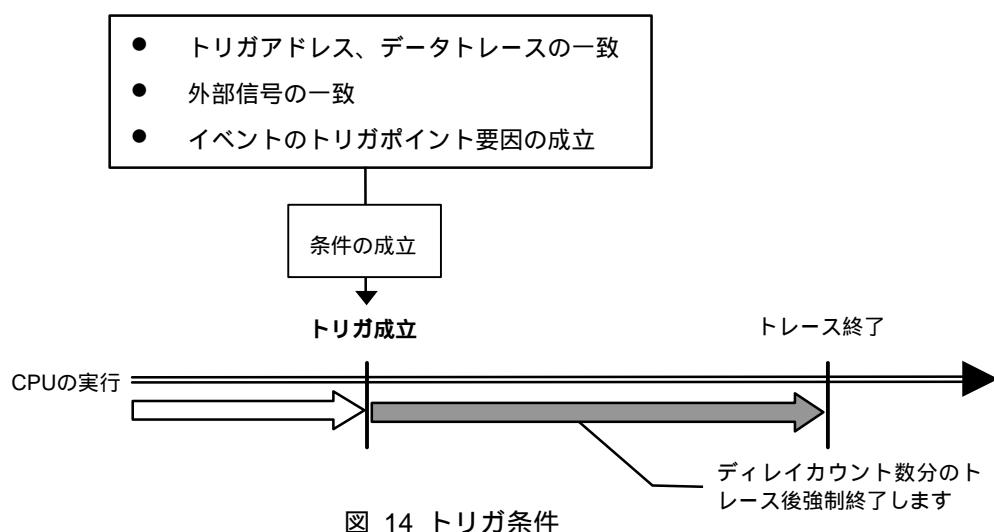


図 13 トレースの開始

#### 4.1.4 トリガ条件

ディレイカウントの起点となる条件です(図 14)。トリガ条件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。

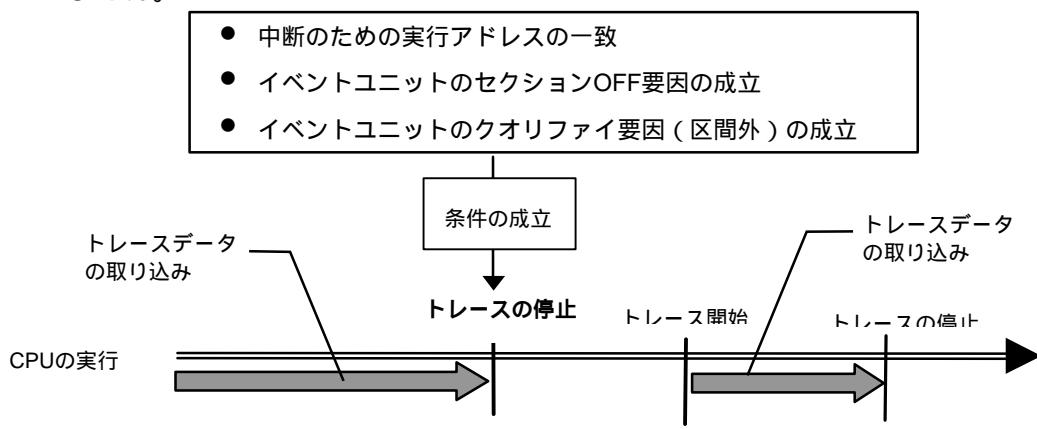


#### 4.1.5 トレースの停止条件

停止条件は、トレースデータの取り込みを一時中断する条件です。

停止後、開始条件が指定されている場合は、その条件によって再度トレースデータの取り込みを開始します(図 15)。

イベントユニットのトレース区間要因を使用するときは、開始条件の使用はできません。



### 4.1.6 トレースの終了

トレースの終了時は、以降のトレースデータの取り込みをしません。

停止条件とは違い、再度トレースを開始することはありません(図 16)。

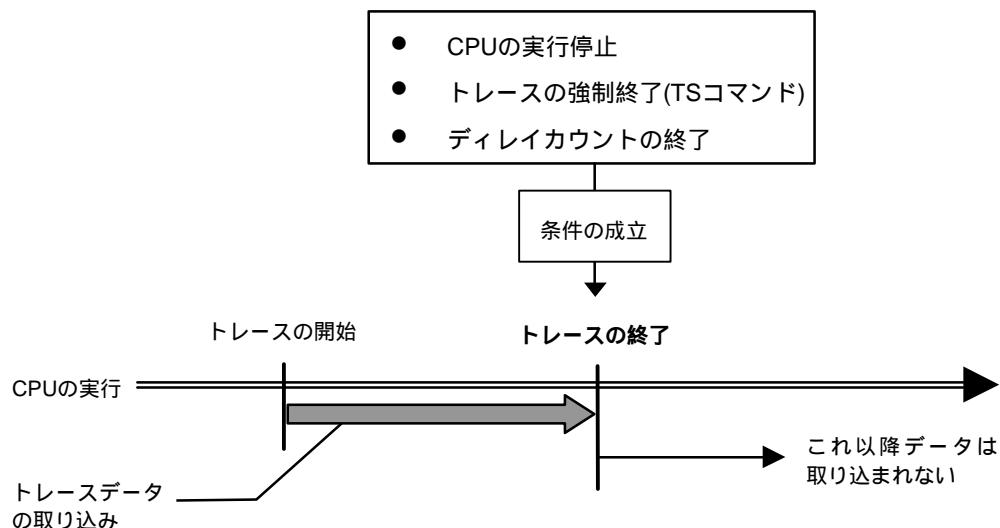


図 16 トレースの終了

### 4.1.7 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。このモード中はトリガ条件を無視します(図 17)。

この場合のトレース開始は、CPUの実行開始です。

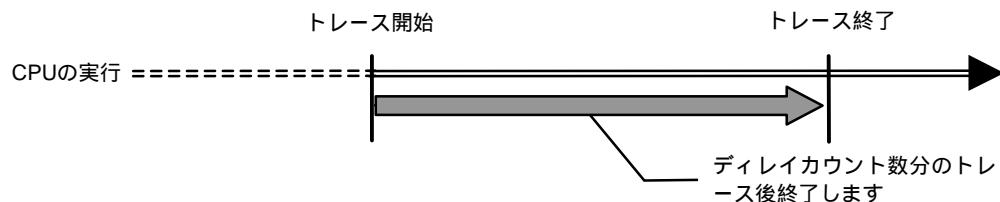


図 17 強制ディレイモード

## 4.2 タイムタグについて

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。

### 4.2.1 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。

### 4.2.2 タイムタグの注意

PARTNER-V850E2-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。

空白ページ

## 5 コマンドリファレンス

PARTNER-V850E2-TPのコマンドウィ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。

## 5.1 コマンド解説の規約

PARTNER-V850E2-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-V850E2-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。

## 5.2 機能別コマンド

### 5.2.1 ハードウェアの初期化

INIT PARTNER-V850E2-TPと ICE を初期化します。

### 5.2.2 CPUリセット

RESET CPU をリセットします。

### 5.2.3 環境設定コマンド

ENV [[!]AUTO] [[!]VERIFY] [JTAG{12|25|5M|2M|1M|500K|250K|100K}]  
 [[!]NMI{0|1|2}] [[!]RESET] [[!]HLDREQ] [[!]STOPZ] [[!]WAITZ]  
 [[!]VSB] [[!]CPINIT] [[!]VBRESZ]

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。

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

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。

#### 5.2.4 プログラムロード

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

#### 5.2.5 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。

#### 5.2.6 プログラム実行

T [回数] テ バ グ プ ロ グ ラ ムをトレース実行します。 (F8)

P [回数] テ バ 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] テ バ グ プ ロ グ ラ ムを実行します。 (F5,F7)  
/W:実行中コマンド 禁止

ESC テ バ グ プ ロ グ ラ ムを強制ブレークします。 (ESC)

#### 5.2.7 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。

#### 5.2.8 アクセスブレークポイント

BH ABP{1|2|3|4} アドレス [アドレスマスク [デ -タ [デ -タマスク]]]  
[(!)ASID [ASID 値]] [種別] [アクセスサイズ] [反転条件]  
アクセスブレークポイントの条件を設定します。

BH [OR1|AFTER1|AND1] [OR2|AFTER2|AND2]  
アクセスブレークポイントの使用モードを設定します。

BH !ABP{1|2|3|4} アクセスブレークポイントを解除します。

BH 現在のアクセスブレークポイントを表示します。

### 5.2.9 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。

### 5.2.10 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。

### 5.2.11 システムレジスタ表示/変更

STAT	システムレジスタ表示
STAT システムレジスタ名=データ	指定レジスタ値をデータに変更します。 <EIPC,EIPSW,FEPC,FEPSW,ECR,PSW,.....>

## 5.2.12 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ターンを検索します。

C 範囲,番地

指定範囲を指定番地と比較(コンペア)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。

## 5.2.13 I/Oポート入力/出力

P[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。

P0[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。

## 5.2.14 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。

## 5.2.15 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。

## 5.2.16 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。

## 5.2.17 リアルタイムトレース

TRC	現在のリアルタイムトレース制御条件を表示します。
TRC [![FORCE] [![SADDR アドレス [ASID [ASID 値]]] [![S_SEC] [![S_QLY], [![EADDR アドレス [ASID ASID 値]] [![E_SEC] [![E_QLY] リアルタイムトレースの開始条件、一時停止条件を設定します。	
TRC [![STSALL] [![EXP] [![INT] [![JMP] [![JR] [![JARL] [![RETI] [![IFD] [![IFE] [![CALLT] [![SWITCH] [![DISPOSE] [![CTRET]] [![TRG] [![TEUTRG] [![DTALL] [![DTTRC] [![EVAP] [![EVAR] [![EVEP] [![EVER] リアルタイムトレースの実行ステータス条件、データトレース条件を設定します。	
TENV	現在のリアルタイムトレース環境条件を表示します。
TENV [!]REAL[![DELAY][COUNT ディレイ値] [!NONBR NONBR MAX max 値] [![PHOLD] [![ONCE] [SUSPEND パケット数] [RESUME パケット数] [SSOR SSAND] [SS_ST_ON SS_ST_OFF] [QS_ST_ON QS_ST_OFF] [![SS_OFFDLY] [![SS_ONDLY] [NOEXT NEGA POSI] [TCLK1 TCLK2 TCLK4] [TDW4 TDW8 TDW16 TDW24 TDW48] リアルタイムトレース環境の各条件を設定します。	
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
TRG	現在のリアルタイムトレースのアドレス条件を表示します。
TRG [!]TRG [トリガアドレス] [ASID [ASID 値]] リアルタイムトレースのトリガアドレスを設定します。	
TRG [!]DTRC{1 2 3 4 5 6} [データアドレス [マスク]] [ASID [ASID 値]] [!]DTRG データトレース条件を設定します。	
TD[U]	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。'U'がついた場合は、逆アセンブル表示モードとなります。
<b>リアルタイムトレース表示モードの内部コマンド</b>	
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	タイムタグ表示の絶対表示と相対表示を切り替えます。

### 5.2.18 イベント

EV 現在の各イベント要因、シーケンシャル器の条件を表示します。

EV {BRK|SECON|SECOFF|QFY|TOUT|TRGP|SSW1|SSW2|SSW3|SSW4|SCLR }  
[!]IP1[!]IP2[!]IP3[!]IP4[!]IP5[!]IP6[!]IP7[!]IP8  
[!]AP1[!]AP2[!]AP3[!]AP4[!]AP5[!]AP6  
[!]IR1[!]IR2[!]IR3[!]IR4[!]AR1[!]AR2[!]AR3  
[!]SEQ[!]EXIN]

各イベント要因の条件を設定します。

EV [{S1|S2|S3|S4}][カント][!]RLE

シーケンシャル器の条件を設定します。

EP 各イベントポートの現在の条件を表示します。

EP [!]IP{1|2|3|4|5|6|7|8} アドレス [{EQ|LT|GT|ON|NE|GE|LE}]  
命令系イベントポートの条件を設定します。

EP [!]AP{1|2|3|4|5|6} アドレス [{EQ|LT|GT|ON|NE|GE|LE}] [データ [データマスク]]  
[{RD|WR|ACC}] [/B|/W|/D|/A] [{MT|UN}]  
アクセス系イベントポートの条件を設定します。

### 5.2.19 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ー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

### 5.2.20 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。

### 5.2.21 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。

### 5.2.22 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	ミニプロト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。

### 5.2.23 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

### 5.2.24 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。

### 5.2.25 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) フォーマット出力します。            |
| >        | フォーマットの中止(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。 |

### 5.2.26 オプション設定

- |                  |  |
|------------------|--|
| OPTION {ON OFF}  | 大小文字判別をします。(SHIFT+F10)<br>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 ウィンドウのかursor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 ウィンドウのかursor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。                           |

## 5.2.27 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	マクロを削除します。
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を読み込みます。
IF{ 式	C の if,elseif,else 文と同様な条件制御コマンドです。

## 5.2.28 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

空白ページ

## 6 高級言語レベルデバッグ

PARTNER-V850E2-TPでは、Green Hills、NEC(CA8x0)、およびGNU(exeGCC)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-V850E2-TPが必要とするデバッグ情報を出力する方法について説明します。

PARTNER-V850E2-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。

## 6.1 GreenHills C

PARTNER-V850E2-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。

### 6.1.1 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。

```
>ccv850e -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。

### 6.1.2 使用例

#### 1 ) コンパイル

```
>ccv850e -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”的オブジェクトファイルが作成されます。

## 2 ) リンク

```
>lx -map=sfile.map -o sfile.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

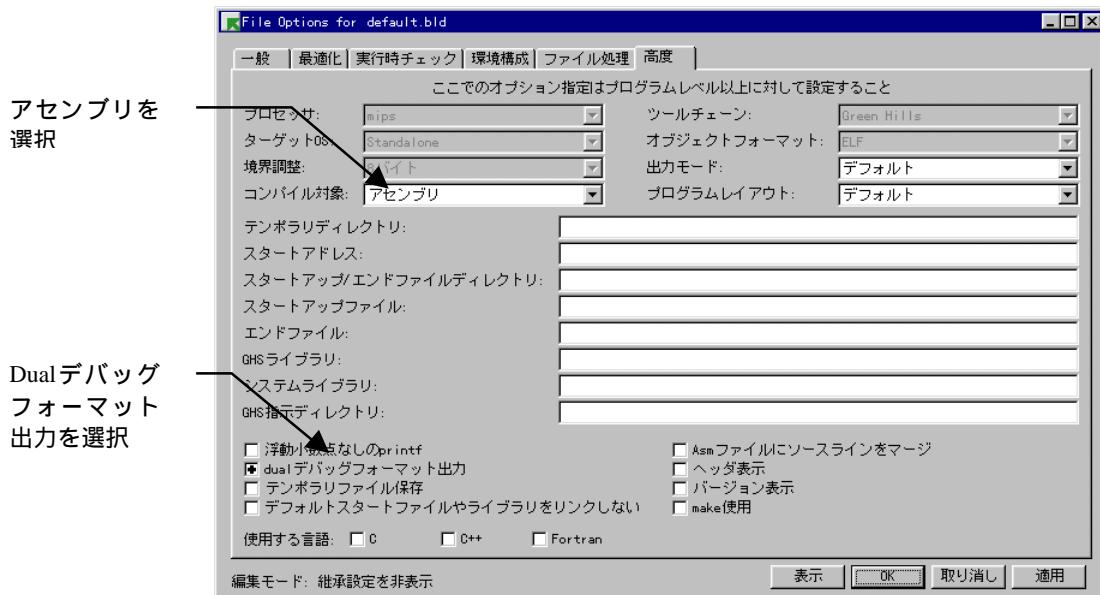
実行プログラムで指定される各セクションをメモリ空間のどの位置に配置するのかを決定します。

```
-m 3 -sec
{
    .text      0x000001000   :
    .rodata    :
    .syscall   :
    .secinfo   :
    .fixaddr   :
    .fixtype   :
    .data      0x03e00000   :
    .sdatabase align(4)   :
    .sdata     :
    .rosdata   :
    .sbss      :
    .bss       :
    .heap align(8) pad(0x4000)   :
    .stack align(8) pad(0x2000)   :
}
}
```

### 6.1.3 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。

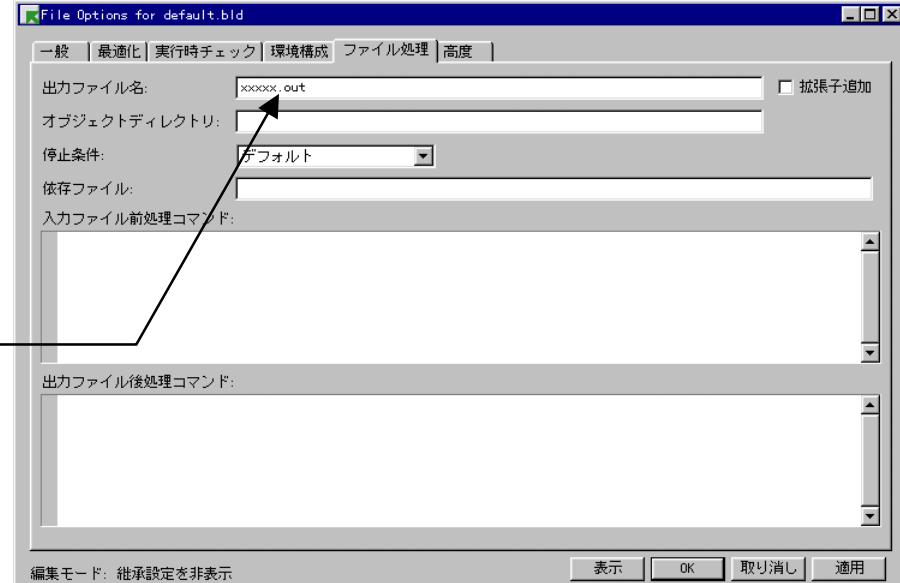
- ProjectメニューからFile Optionダイアログを表示します。
- 高度タブを表示します。
- dualデバッグフォーマット出力を選択します。
- コンパイル対象ドロップリストでアセンブリを選択します。



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。

## 6.2 GNU C(exeGCC)

PARTNER-V850E2-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。

### 6.2.1 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 -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。

### 6.2.2 使用例

#### 1 ) コンパイル

```
>gcc -g -m850e -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”的オブジェクトファイルが作成されます。

#### 2 ) リンク

```
>ld -Map sfile.map -Tchkpro.cmd -o sfile.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 ;
    __heep_size = 0x6000 ;
    __stack_init = 0x3e08000 ;
    .text 0x00001000 :
    {
        *(.jmptbl)
        *(.text)
    }
    .call_table ALIGN (4) :
    {
        *(.call_table)
        __ctbp = .;
        *(.call_table_data)
        *(.call_table_text)
    } = 0xff /* fill gaps with 0xff */
    .rodata ALIGN (4) :
    {
        *(.rodata)
        __erdata = .;
    }
    .data ALIGN (4) :
    {
        __data = .;
        *(.data)
        __edata = .;
    }
    .tdata ALIGN (4) : {
        __ep = .;
        *(.tbyte)
        *(.tcommon_byte)
        *(.tdata)
        *(.tbss)
        *(.tcommon)
    }
    .bss ALIGN (4) :
    {
        __fbss = .;
        *(.bss)
        *(COMMON)
        __end = .;
        __heep = .;
    }
}
```

( 次頁に続く )

```
/* DWARF debug sections.  
   Symbols in the .debug DWARF section are relative to the beginning of the  
   section so we begin .debug at 0.  It's not clear yet what needs to happen  
   for the others. */  
.debug          0 : { *(.debug) }  
.debug_srcinfo  0 : { *(.debug_srcinfo) }  
.debug_aranges  0 : { *(.debug_aranges) }  
.debug_pubnames 0 : { *(.debug_pubnames) }  
.debug_sfnames  0 : { *(.debug_sfnames) }  
.line           0 : { *(.line) }  
}
```

## 6.3 NEC C

PARTNER-V850E2-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。

### 6.3.1 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。

>ca850 -c -g ソースファイル名.c  
>ld850 -D リンクディレクティブ -o実行ファイル名.outソースファイル名.o -Llib -lc

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
リンクディレクティブ	リンクディレクティブファイル名

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。

### 6.3.2 使用例

#### 1 ) コンパイルとリンク

コンパイルとリンクはVSH環境のメイクツール(VMAKE)を利用します。

以下にmakefileの例を示します。

```
NECROOT = C:/nectools  
ca850 -g -c sfile.c  
ld850 -D dfile -o sfile.out lib\$crtn850.o sfile.o -Llib -lc
```

#### 2 ) リンクマップの例

```
TEXT : !LOAD ?RX V0x00001000 {  
    .text      = $PROGBITS ?AX;  
};  
  
DATA : !LOAD ?RW V0x03e00000 {  
    .data      = $PROGBITS ?AW;  
    .sdata     = $PROGBITS ?AWG;  
    .sbss      = $NOBITS ?AWG;  
    .bss       = $NOBITS ?AW;  
};  
    __tp_TEXT @ %TP_SYMBOL;  
    __gp_DATA @ %GP_SYMBOL &__tp_TEXT;
```

空白ページ

## 7 フラッシュメモリのサポート

PARTNER-V850E2-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。

## 7.1 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デバイス
  - 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) ポ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デバイス
  - 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) ポ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

上記の仕様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を次表に示します。

**【対応デバイス】**

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****

## 7.2 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(RPTV850E2T.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)

『7.1 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範囲、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
I32_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)

『7.1 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、これら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-V850E2-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           MAP省略時と同じ
FLASH_MEM 03000000,037FFFFF, 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 03000000,037FFFFF, 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      ←
```

## 7.3 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-V850E2-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フ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。

## 7.4 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フ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-V850E2-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	R0 レジスタ,,,R31レジスタ
_pc	PC レジスタ
_hp	R2 レジスタの別名
_sp	R3 レジスタの別名
_gp	R4 レジスタの別名
_tp	R5 レジスタの別名
_ep	R30レジスタの別名
_lp	R31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R0とR1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-V850E2-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。

## エラーメッセージ

---

### 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ぬか、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をリセットした後、再度初期化コマンドを実行してください。

# 索引

.

.CFG ..... 68, 69

## F

FLASH\_MEM ..... 68, 69

環境設定 ..... 68, 69

## G

GNU C ..... 59

コンパイルとリンク方法 ..... 59

使用例 ..... 59

Green Hills C ..... 56

コンパイルとリンク方法 ..... 56

使用例 ..... 56

## I

I/O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

INIT.MCR ..... 8

## M

MULTI2000での作成 ..... 57

## N

NEC C ..... 62

コンパイルとリンク方法 ..... 62

使用例 ..... 63

## P

PARTNER-V850E2-TPの環境設定 ..... 4

## あ

エラーメッセージ ..... 77

## か

起動 ..... 3

PARTNER-V850E2-TPの環境設定 ..... 4

起動時のエラーメッセージ ..... 9

初期設定コマンド ..... 7

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 ..... 5

高級言語レベルデバッグ ..... 55

GNU C ..... 59

Green Hills C ..... 56

NEC C ..... 62

コマンドリファレンス ..... 43

機能別コマンド .....	45	レジスタ表示/変更 .....	47
CPUリセット .....	45	ロギング/バッチ .....	52
Cのデータ参照/変更 .....	51	コマンド解説の規約 .....	44
I/Oポート入出力 .....	48	コンフィギュレーションファイル .....	68, 69
アクセスブレークポイント .....	46		
アセンブル .....	48	<b>さ</b>	
イベント .....	50		
オプション設定 .....	52	ショートカットキー	
環境設定コマンド .....	45	I/Oウインドウ .....	20
基数変更 .....	52	ヒストリウインドウ .....	18
コード表示 .....	47	ブレークウインドウ .....	15
式の表示 .....	50	レジスタウインドウ .....	13
システムコール .....	50		
システム制御 .....	51	初期設定コマンド	
システムレジスタ表示/変更 .....	47	起動時に必要なファイル .....	8
シンボル表示/設定 .....	48	初期化で使用するコマンド .....	7
ハードウェアの初期化 .....	45		
バックトレース .....	48	<b>た</b>	
ファイルのリード/ライト .....	46	ダイアログコマンド .....	23
フラッシュメモリ管理 .....	53	ダイアログボックス .....	24
ブレークポイント .....	46	CPU環境設定 .....	31
プログラム実行 .....	46	アクセスブレークの設定 .....	24
プログラムロード .....	46	イベントポイント設定 .....	30
マクロコマンド .....	53	イベント設定 .....	29
メモリ表示/変更 .....	48	エミュレーションROM設定 .....	32
文字列の定義 .....	51	トレースアドレス設定 .....	28
リアルタイムトレース .....	49	トレース環境条件設定 .....	25
		トレース制御条件設定 .....	27
		各種状態の設定 .....	33
		タイムタグについて .....	41

タイムタグの注意 .....	41	マウス操作 .....	16
タイムタグ表示の種類 .....	41	ローカルメニュー .....	16
チャイルドウインドウ .....	11	付録 .....	75
I/Oウインドウ .....	20	エラーメッセージ .....	77
ヒストリウインドウ .....	17	レジスタ変数 .....	76
ブレークウインドウ .....	15	<b>ま</b>	
レジスタウインドウ .....	12		
<b>は</b>			
はじめに .....	1	マウス操作	
ヒストリウインドウ		I/Oウインドウ .....	21
ショートカットキー .....	18	ブレークウインドウ .....	16
ローカルメニュー .....	18	レジスタウインドウ .....	14
ヒストリウインドウ .....	17	<b>ら</b>	
必要なセットアップ .....	1		
必要なハードウェア .....	1	リアルタイムトレース .....	35
フラッシュメモリのサポート		トレースの概要 .....	36
環境設定 .....	68	ディレイカウント .....	37
管理コマンド .....	74	トリガ条件 .....	39
イレーズ .....	74	トレースの開始条件 .....	38
クリア .....	74	トレースの停止(終了)条件 .....	39, 40
ステータス .....	74	強制ディレイモード .....	40
フラッシュ .....	74	通常/完全モード .....	37
書き込み動作 .....	72	レジスタウインドウ .....	12
対応デバイス .....	66	ショートカットキー .....	13
フラッシュメモリのサポート .....	65	マウス操作 .....	14
ブレークウインドウ .....	15	ローカルメニュー .....	13
ショートカットキー .....	15	レジスタ変数 .....	76

## 索引

---

ヒストリウインドウ .....	18	レジスタウインドウ .....	13
ブレークウインドウ .....	16		

**PARTNER ユーザーズ・マニュアル**  
**V800シリーズ**  
**『V850E2-TP個別編』**

第一版 発行日 2003年3月

株式会社 マイダス・ラボ

Copyright 2003 Midas lab Inc./Kyoto Micro Computer Co.,LTD.