

お客様各位

カタログ等資料中の旧社名の扱いについて

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

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

2010年4月1日
ルネサスエレクトロニクス株式会社

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

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

ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りが無いことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

注 1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。



ユーザーズ・マニュアル

IE-75000-R/IE-75001-R

インサーキット・エミュレータ

資料番号 U15337JJ3V0UM00 (第3版)
(旧資料番号 EEU - 846B)
発行年月 February 2001 N CP(K)

© NEC Corporation 1992

[メ モ]

この装置は、第一種情報装置（商工業地域において使用されるべき情報装置）で商工業地域での電波妨害禁止を目的とした情報処理装置等電波障害自主規制協議会（VCCI）基準に適合しております。

したがって、住宅地域，またはその隣接した地域で使用すると，ラジオ，テレビジョン受信機などに受信障害を与えることがあります。

ユーザズ・マニュアルに従って正しく取り扱いをしてください。

PC DOSは，米国IBM Corp. の商標です。

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

- **本資料の内容は予告なく変更することがありますので，最新のものであることをご確認の上ご使用ください。**
- 文書による当社の承諾なしに本資料の転載複製を禁じます。
- 本資料に記載された製品の使用もしくは本資料に記載の情報の使用に際して，当社は当社もしくは第三者の知的財産権その他の権利に対する保証または実施権の許諾を行うものではありません。上記使用に起因する第三者所有の権利にかかわる問題が発生した場合，当社はその責を負うものではありませんのでご了承ください。
- 本資料に記載された回路，ソフトウェア，及びこれらに付随する情報は，半導体製品の動作例，応用例を説明するためのものです。従って，これら回路・ソフトウェア・情報をお客様の機器に使用される場合には，お客様の責任において機器設計をしてください。これらの使用に起因するお客様もしくは第三者の損害に対して，当社は一切その責を負いません。

本版で改訂された主な箇所

箇所	内容
全般	ディバグ対象デバイスの開発完了(μ PD75036, 116H, 117H, 312B, 316B, P336)
p. 43	第5章 操作概要 5.2.1 (5) セットアップ・ファイルの設定 注意を修正
p. 123	第7章 ファームウェアROMのバージョンについて ファームウェアROM (Ver.1.4), コントロール・プログラム (Ver. 1.1) の開発完了
p. 165 p. 166 p. 361	第8章 コマンド説明 8.4.6 (2) の [使用例] (c) オペランド以下を省略したとき 注意を追加 (d) 前回は設定したイベント条件を取り消したいとき 追加 8.4.45 [使用例] (a) トレーサ起動, 停止条件の指定 修正
p. 398 p. 398, 399	付録B エラー・メッセージ一覧表 エラー・メッセージNo.49の説明を修正 エラー・メッセージNo.50-52を追加
前版p. 401-404	付録C 使用上の注意 No.4, 13, 15を削除

本文欄外の★印は、本版で改訂された主な箇所を示しています。

巻末にアンケート・コーナーを設けております。このドキュメントに対するご意見をお気軽にお寄せください。

はじめに

対象者 このマニュアルは、4ビット・シングルチップ・マイクロコンピュータ75Xシリーズを採用し、IE-75001-Rによりシステム・デバッグを行うエンジニアを対象としています。

目的 このマニュアルは、IE-75001-Rの持つ各種デバッグ機能を理解していただくことを目的とします。

構成 このマニュアルは大きく分けて次の内容で構成しています。

- ・システム構成
- ・機能概要
- ・コマンド説明

読み方 このマニュアルを読むエンジニアは、75Xシリーズのデバッグ対象デバイスの機能、使用方法に熟知し、またデバッグの知識があることを前提としています。

IE-75000-Rをご使用の場合は、このマニュアルのIE-75001-RをIE-75000-Rと読み替えてください（第1章 概説参照）。

一通りIE-75001-Rの機能、操作方法を理解しようとするとき
→目次に従って読んでください。

基本仕様、動作環境を理解しようとするとき
→「第1章 概説」から「第4章 システム構成」を読んでください。

基本的な操作手順、機能を理解しようとするとき
→「第5章 操作概要」から「第6章 基本機能の使い方」を読んでください。

具体的なコマンドの種類、機能、入力フォーマットを理解しようとするとき
→「第8章 コマンド説明」を読んでください。

IE-75001-Rのシリアル・インタフェース、またはパラレル・インタフェースを使用して周辺装置を接続する場合
→「第9章 その他の機能」を読んでください。

用語について

このマニュアルの中で使用する用語について、その意味を下表に示します。

用語	意味
エミュレーション・デバイス	エミュレータ内で対象デバイスのエミュレーションを行っているデバイスの総称です。 エミュレーションCPUを含みます。
エミュレーションCPU	エミュレータ内で、ユーザが作成したプログラムを実行しているCPU部分です。
対象デバイス	エミュレーションの対象となっているデバイスです（本チップ）。
ターゲット・プログラム	ディバグの対象となるプログラムです（ユーザの作ったプログラム）。
ターゲット・システム	ディバグの対象となるシステムです（ユーザの作ったシステム）。 ターゲット・プログラムおよびユーザの作成したハードウェアを含みます。 狭義にはハードウェアのみを指します。

凡 例 データ表記の重み : 左が上位桁, 右が下位桁

内の表記 : モニタ画面に表示される内容や入力したコマンド

入力キーの記述 : XXXXX はキー入力を示します。

はリターン・キー

ESC はエスケープ・キー

注 : 本文中に付けた注の説明

注意 : 特に気をつけて読んでいただきたい内容

備考 : 本文の補足説明

数の表記 : 2進数 $XXXXY$

8進数 $XXXXQ$

10進数 $XXXXT$

16進数 $XXXXH$

関連資料一覧

資料名			資料番号
IE-75000-R-EM ユーザーズ・マニュアル			EEU-673
IE-75617-R-EM ユーザーズ・マニュアル (暫定)			EEU-840
PG-1500 ユーザーズ・マニュアル			EEU-651
RA75X アセンブラ・パッケージ ユーザーズ・マニュアル	操作編	PC-9800シリーズ (MS-DOS) ベース IBM PCシリーズ (PC DOS) ベース	EEU-731
	言語編	—	EEU-730
PG-1500コントローラ ユーザーズ・マニュアル			EEU-704

注意 上記関連ドキュメントは予告なしに内容を変更することがあります。設計などには必ず最新の資料をご使用ください。

目 次

第1章 概 説 … 1
1.1 システム構成 … 1
1.2 基本仕様 … 2
1.3 コントロール／トレース・ボードとブレーク・ボードの構成 … 4
1.3.1 コントロール／トレース・ボード・ブロック … 4
1.3.2 ブレーク・ボード・ブロック … 6
第2章 各部の名称 … 9
2.1 本体各部の名称 … 9
2.2 付 属 品 … 14
第3章 設 置 … 15
3.1 設 置 … 15
3.2 RS-232-Cモード設定 … 17
3.3 コントロール／トレース・ボードの設定 … 20
第4章 システム構成 … 23
4.1 ホスト・マシンとの接続 … 24
4.1.1 PC-9800シリーズとの接続 … 24
4.1.2 IBM PCシリーズとの接続 … 26
4.2 PGシリーズとの接続 … 29
4.2.1 PG-1500との接続 … 29
4.2.2 PG-1000との接続 … 31
第5章 操作概要 … 35
5.1 システムの動作環境 … 35
5.2 操作手順 … 39
5.2.1 システムの起動手順 … 40
5.2.2 プログラムのデバッグ手順 … 47
5.2.3 システムの終結手順 … 48
第6章 基本機能の使い方 … 49
6.1 システムの動作モードとコマンド入力 … 50
6.2 基本機能の使い方 … 52
6.2.1 クロックの選択機能 … 52
6.2.2 リセット機能 … 53
6.2.3 カバレッジ機能 … 53

- 6.2.4 ロード機能 … 54
- 6.2.5 エミュレーション実行機能 … 56
- 6.2.6 ブレーク機能 … 63
- 6.2.7 トレース機能 … 66
- 6.2.8 チェック機能 … 74
- 6.2.9 イベント設定, 検出機能 … 77
- 6.2.10 レジスタ操作機能 … 82
- 6.2.11 メモリ操作機能 … 84
- 6.2.12 セーブ機能 … 86
- 6.2.13 システム終了機能 … 87
- 6.2.14 その他の機能 … 88
- 6.3 ディバグの基本手順 … 91**
- 6.4 基本機能の使用例 … 96**
 - 6.4.1 IE-75001-Rの初期設定 … 97
 - 6.4.2 ディバグ環境の設定 … 98
 - 6.4.3 ターゲット・プログラムの実行 … 100
 - 6.4.4 実行結果の確認 … 110
 - 6.4.5 プログラムの修正 … 118
 - 6.4.6 ディバグの終了 … 120
 - 6.4.7 システムの終結 … 122

第7章 ファームウェアROMのバージョンについて … 123

- 7.1 追加機能について … 123
- 7.2 バージョンによる機能の違い … 123

第8章 コマンド説明 … 125

- 8.1 コマンド表記法 … 126**
 - 8.1.1 コマンド形式 … 126
 - 8.1.2 コマンド構成要素の説明 … 129
- 8.2 数値, シンボル, 式の記述仕様 … 135**
 - 8.2.1 数値の記述仕様 … 135
 - 8.2.2 特殊数値の記述仕様 (X記述) … 137
 - 8.2.3 シンボルの記述仕様 … 138
 - 8.2.4 式の記述仕様 … 140
- 8.3 特殊キーの使用法 … 141**
- 8.4 コマンド説明 … 142**
 - 8.4.1 アセンブル・コマンド (ASM) … 142
 - 8.4.2 データ・メモリ・バンク設定コマンド (BNK) … 145
 - 8.4.3 データ・メモリ・イベント条件設定コマンド (BRA) … 147
 - 8.4.4 イベント情報表示コマンド (BRK) … 153
 - 8.4.5 イベント・モード設定コマンド (BRM) … 156
 - 8.4.6 プログラム・メモリ・イベント条件設定コマンド (BRS) … 160
 - 8.4.7 チェック・ポイント設定コマンド (CHK) … 167
 - 8.4.8 クロック選択コマンド (CLK) … 172
 - 8.4.9 コマンド・ファイル作成コマンド (COM) … 174

8.4.10	カバレッジ測定結果操作コマンド	(CVD) … 179
8.4.11	カバレッジ測定範囲設定コマンド	(CVM) … 183
8.4.12	逆アセンブル・コマンド	(DAS) … 189
8.4.13	ディレクトリ表示コマンド	(DIR) … 191
8.4.14	イベント検出点設定コマンド	(DLY) … 192
8.4.15	DOSコマンド実行コマンド	(DOS) … 195
8.4.16	コントロール・プログラム終了コマンド	(EXT) … 197
8.4.17	コマンド・ヒストリ表示コマンド	(HIS) … 198
8.4.18	ヘルプ・コマンド	(HLP) … 200
8.4.19	ロード・コマンド	(LOD) … 203
8.4.20	出力デバイス・リダイレクト・コマンド	(LST) … 206
8.4.21	演算コマンド	(MAT) … 210
8.4.22	プログラム・メモリ操作コマンド	(MEM) … 211
8.4.23	チャンネル2モード設定コマンド	(MOD) … 229
8.4.24	外部センス・クリップ・モード設定コマンド	(OUT) … 232
8.4.25	バス回数設定コマンド	(PAS) … 238
8.4.26	端末モード設定コマンド	(PGM) … 239
8.4.27	データ・メモリ操作コマンド	(RAM) … 250
8.4.28	汎用レジスタ操作コマンド	(REG) … 268
8.4.29	リセット・コマンド	(RES) … 275
8.4.30	エミュレーション実行コマンド	(RUN) … 277
8.4.31	セーブ・コマンド	(SAV) … 290
8.4.32	エミュレーション・デバイス・モード切り替えコマンド	(SET) … 293
8.4.33	特殊レジスタ操作コマンド	(SPR) … 295
8.4.34	入力デバイス・リダイレクト・コマンド	(STR) … 302
8.4.35	リアルタイム・エミュレーション中止コマンド	(STP) … 306
8.4.36	対象デバイス選択コマンド	(STS) … 309
8.4.37	シンボル操作コマンド	(SYM) … 313
8.4.38	システム再起動コマンド	(SYS) … 330
8.4.39	トレース表示コマンド	(TRD) … 332
8.4.40	トレース・データ検索条件設定コマンド	(TRF) … 346
8.4.41	トレーサ起動コマンド	(TRG) … 350
8.4.42	トレース・モード指定コマンド	(TRM) … 351
8.4.43	トレース・ポインタ操作コマンド	(TRP) … 353
8.4.44	クオリファイ・トレース条件設定コマンド	(TRX) … 356
8.4.45	セクション・トレース条件設定コマンド	(TRY) … 359
8.4.46	オブジェクト・ベリファイ・コマンド	(VRY) … 363

第9章 その他の機能 … 365

9.1 IE-75001-RのRS-232-Cインタフェース機能概要 … 365

- 9.1.1 RS-232-Cインタフェースの信号線 … 366
- 9.1.2 ターミナル・モードとモデム・モード … 367
- 9.1.3 RTSの設定 … 368
- 9.1.4 ソフトウェア・ハンドシェークとハードウェア・ハンドシェーク … 369
- 9.1.5 チャンネル1とチャンネル2の機能 … 372

9.2 IE-75001-Rの平行・インタフェースの機能概要 … 373

- 9.2.1 パラレル・インタフェースの信号線 … 374
- 9.2.2 チャンネル3, チャンネル4の機能（高速ダウン・ロード・モード） … 375
- 9.2.3 パラレル・インタフェース回路 … 376
- 9.2.4 高速ダウン・ロード・モードのタイミング … 377

付録A コマンド一覧表 … 379

付録B エラー・メッセージ一覧表 … 395

付録C 使用上の注意 … 401

- C.1 使用上の注意概要 … 401**
- C.2 使用上の注意の詳細説明 … 402**

写真の目次

写真番号	タイトル, ページ
2-1	IE-75001-Rの外観と各部名称 … 9
2-2	IE-75001-Rの正面 … 10
2-3	IE-75001-Rの裏側 … 10
2-4	IE-75001-Rの側面 … 11
2-5	RS-232-C設定部の拡大図 … 11
2-6	ボード位置 … 12
2-7	ブレーク・ボード (IE-75000-R-BK) … 13
2-8	コントロール/トレース・ボード … 13
3-1	IE-75001-Rの裏側 (AC IN) … 16
3-2	IE-75001-Rの側面 (インタフェース・ケーブルの接続) … 16
3-3	IE-75001-Rの内部 … 21

図 の 目 次 (1/2)

図番号	タイトル, ページ
1-1	IE-75001-Rシステム構成 … 1
1-2	コントロール/トレース・ボード・ブロック図 … 5
1-3	ブレーク・ボード・ブロック図 … 7
3-1	エミュレーション・プローブの接続 … 15
3-2	RS-232-Cの各モード・スイッチ … 17
3-3	出荷時の各モード設定 … 18
3-4	J1, J2ケーブル接続図 … 21
4-1	チャンネル1の設定 (PC-9800シリーズとの接続) … 24
4-2	チャンネル1の設定 (IBM PCシリーズとの接続) … 26
4-3	アシンクロナス・コミュニケーション・アダプタの設定 … 27
4-4	IBM PCとのRS-232-C接続 … 28
4-5	チャンネル2の設定 (PG-1500との接続) … 30
4-6	チャンネル2の設定 (PG-1000との接続) … 31
4-7	PG-1000の設定 (底面DIPスイッチ) … 32
5-1	システムの動作環境 … 36
5-2	操作手順概略フロー・チャート … 39
6-1	プロンプト表示とシステム動作状態の例(1) … 51
6-2	プロンプト表示とシステム動作状態の例(2) … 57
6-3	プロンプト表示とシステム動作状態の例(3) … 59
6-4	プロンプト表示とシステム動作状態の例(4) … 61
6-5	プロシージャ実行の概念図 … 62
6-6	トレースの概念図 … 67
6-7	チェック・データを含んだ場合のトレース・メモリ … 76
6-8	イベント検出点の設定 … 80
6-9	イベント検出の概念図 … 81
6-10	デバッグの基本手順フロー・チャート … 91
8-1	イベント条件の設定と検出手順 … 157
8-2	トリガ・フレームのポイント … 193
8-3	1ステップ実行モード … 278

図 の 目 次 (2/2)

図番号	タイトル, ページ
8-4	トレース・ポイント図 … 344
8-5	トレーサの状態図 … 356
8-6	トレーサの状態図 … 359
9-1	ターミナル／モデム・モード切り替え回路図 … 367
9-2	パラレル・インタフェースのピン配置 … 374
9-3	パラレル・インタフェース回路 … 376
9-4	高速ダウン・ロード・モードのタイミング … 377

表 の 目 次 (1/2)

表番号	タイトル, ページ
1-1	基本仕様 … 2
3-1	キャラクタ仕様 … 19
3-2	コントロール/トレース・ボードのジャンパ設定 (出荷時) … 20
4-1	チャンネル1の設定 (PC-9800シリーズとの接続) … 24
4-2	PC-9800シリーズの設定 … 25
4-3	PC-9800シリーズとの接続 … 25
4-4	チャンネル1の設定 (IBM PCシリーズとの接続) … 26
4-5	IBM PCの設定 … 26
4-6	IBM PCとの接続 … 28
4-7	PG-1500との接続 … 29
4-8	チャンネル2の設定 (PG-1500との接続) … 29
4-9	PG-1500の設定 … 30
4-10	チャンネル2の設定 (PG-1000との接続) … 31
4-11	PG-1000の設定 … 32
4-12	PG-1000との接続 … 33
5-1	ハードウェア環境 … 37
5-2	ソフトウェア環境 … 38
5-3	ホスト・マシンのシリアル・チャンネルの設定 … 40
6-1	ホスト・マシンからダウン・ロードされるファイル … 54
6-2	ロード・コマンドの種類と機能 … 55
6-3	トレース項目と指定値範囲 … 70
6-4	トレース・データ表示の説明 … 71
6-5	トレース・データの範囲指定 … 72
6-6	ホスト・マシンにセーブできるファイル … 86
6-7	セーブ・コマンドの種類と機能 … 87
7-1	ファームウェアROMと、コントロール・プログラムのバージョンの違いによる動作 … 123
8-1	コマンド一覧 … 129
8-2	各数値における最小値, 最大値の指定例 … 136

表 の 目 次 (2/2)

表番号	タイトル, ページ
8-3	PG-1500のコマンド一覧 … 241
8-4	PG-1000のコマンド一覧 … 244
8-5	ページの概念 … 332
8-6	option表 … 333
8-7	メニュー・モード時の入力と動作 … 334
8-8	option指定個数表 … 336
8-9	チェック・ポイント・データの種別 … 339
9-1	RS-232-Cインタフェースの信号線 … 366
9-2	ピン番号とRTS … 366
9-3	RTSの設定 … 368
9-4	ソフトウェア・ハンドシェークの接続 … 369
9-5	ハードウェア・ハンドシェークの接続 … 370
9-6	チャンネル1とチャンネル2の機能 … 372
9-7	パラレル・インタフェースの信号表 … 374

第1章 概 説

IE-75001-Rは、4ビット・シングルチップ・マイクロコンピュータ、75Xシリーズを用いたハードウェア、またはソフトウェアを効率的にデバッグするための開発支援装置です。IE-75001-R単体だけでは動作しません。必ず別売のエミュレーション・ボード (IE-75000-R-EMまたはIE-75617-R-EM^注) を接続してお使いください。なお、IE-75000-RにはIE-75000-R-EMが内蔵されています。

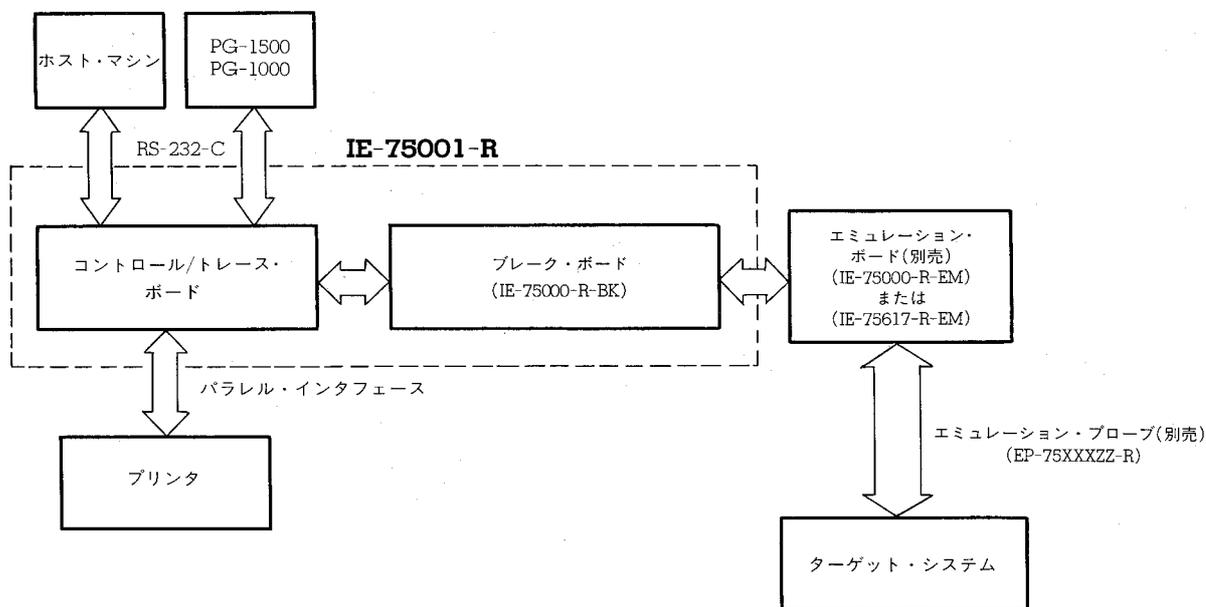
この章ではIE-75001-Rのシステム構成および基本仕様について説明します。

注 開発中

1.1 システム構成

IE-75001-Rのシステム構成は次のようになっています。

図 1-1 IE-75001-Rシステム構成



1.2 基本仕様

表 1-1 基本仕様 (1/2)

項 目		内 容	
エミュレーション対象デバイス		μPD75×××	
動作周波数		32 kHz-6 MHz (CPUクロック)	
メモリ容量		プログラム・メモリ 64 Kバイト データ・メモリ 4 Kニブル	
エミュレーション機能		ノンブレイクによるリアルタイム実行 (RUN N) ブレイクポイント付きリアルタイム実行 (RUN B) シングル・ステップによるCPUトレース実行 (RUN T)	
イベント 検出 条件	アクセス系検出 (BRA1~BRA4)	指定フル・イベントのアクセスを検出する。 (アドレス, データ, ステータス, 外部信号)×4ポイント(アドレス・マスク可能) うち2ポイントはアドレス・パーティション可能。	
	フェッチ系検出 (BRS1, BRS2)	7ポイント・パラレル・フェッチまたは4ポイント・シーケンシャル・フェッチ (BRS1), 1ポイント・フェッチ (BRS2)と外部信号とのAND設定可	
バス・カウント		ブレイク用イベント条件の検出回数をカウント(1-255回)。ただし, 複数のイベント条件 を同時に検出した場合は1回としてカウントする。	
イベント・ディレイ機能		トレース・フレームに対するイベント検出点の設定 FIRST/MIDDLE/LAST	
強制ブレイク		マニュアル・ブレイク, ガード・ブレイク (データ・メモリ, SPR, レジスタ, スタック)	
ト レ ー ス 機 能	トレース・モード	トレース条件	クロック・サイクル
		全トレース: 無条件にトレースする。	マシン・サイクル: 全トレース時のみ有効 (ポート・トレース可 能) システム・クロックごとにトレースする。
		セクション・トレース: イネーブル条件でトレース開始。 ディスエーブル条件でトレース終了。	イベント・サイクル: 全モード時有効 (ポート・トレース不可能) フェッチ, リード, ライト, 割り込みが起 こったときのみ必要なバスをトレースする。
	クオリファイ・トレース: クオリファイ条件成立間だけトレース を行う。		
トレース容量	49ビット×2Kワード		
データ表示	インストラクション表示, 検索表示, チェック・ポイント表示, 2進表示 (ポート, 外部データ)		

表 1-1 基本仕様 (2/2)

項 目	内 容
アクティブ・モード	エミュレーション・デバイス動作中に実現可能な機能 イベント検出条件の参照/変更 イベント・ディレイの参照/変更 トレース・データ表示 トレース・モードの参照/変更 トレーサ再起動
イベント出力	ブレイク条件成立時にトリガ信号を出力する (アクティブ・ロウ・レベル)
外部インタフェース	RS-232-C (300-19200 bps) : 2チャンネル CH1 : コンソールおよびホスト・マシン接続用 CH2 : PROMプログラマ用, VAX用 セントロニクスI/F : 2チャンネル セントロニクス入力 : オブジェクト・ダウン・ロード用 セントロニクス出力 : セントロニクス入力をスルーで出力
カバレッジ機能	C0カバレッジ機能, カバレッジ・エリアの設定可能
チェック機能	イベント成立時に指定レジスタ・メモリSPRをトレーサに書き込む (合計MAX.5個)。実行は一時的に停止するがすぐに自動的に再開する。
アクセス系データ出力	データ・メモリSPRの2アドレス (各ニブル) のR/Wデータをリアルタイムに外部センス・クリップへ出力可能。
環境条件のロード/セーブ	デバッグ環境のロード/セーブが可能。データ・メモリも可能。
ファミリー展開	エミュレーション・プローブを交換することにより, ファミリ内のすべての製品に対応が可能。各対象デバイスとエミュレーション・プローブの対応については, IE-75000-R-EMユーザーズ・マニュアル および IE-75617-R-EMユーザーズ・マニュアル(暫) を参照。
外形寸法	奥行き : 370 mm, 横幅 : 160 mm, 高さ : 283 mm
重 量	約8.5 kg
入力電圧	AC100/120 V (50/60 Hz) } 入力自動切替方式 AC200/240 V (50/60 Hz) }
入力電流 (TYP.)	AC100 V 2A, AC200 V 1A (50/60 Hz)
使用温度範囲	10~40°C
保存温度範囲	-20~+45°C (結露しないこと)
周囲湿度範囲	0~90 %RH
設置場所	設置場所は, ゴミや, チリなどの少ない場所を選びます。 また空気取り入れ口付近には障害物を置かないようにしてください。

1.3 コントロール／トレース・ボードとブレイク・ボードの構成

IE-75001-Rの中核である、コントロール／トレース・ボードとブレイク・ボード (IE-75000-R-BK) の各ブロックについて説明します。

なお、エミュレーション・ボードのブロックについては、**IE-75000-R-EM ユーザーズ・マニュアル** および**IE-75617-R-EM ユーザーズ・マニュアル(暫定)**を参照してください。

1.3.1 コントロール／トレース・ボード・ブロック

コントロール／トレース・ボード・ブロックの構成は次のとおりです (図1-2 参照)。

(1) **ドライバ・コントロール**

ドライバ・モジュールとのインタフェースです。

(2) **トレースRAM**

トレース・データを、バス・サイクルごとにイベント検出点まで2047ステップの最新データを保持します。

(3) **メモリ・バンク・セクタ**

バンク切り替えによりROM, DRAMユニット, トレースRAMを選択します。

(4) **シリアル・インタフェース**

RS-232-C仕様のインタフェースが2チャンネルあります。

(5) **パラレル・インタフェース**

高速ダウン・ロードとスルー出力の2チャンネルがあります。

(6) **I/Oセクタ**

バンク切り替えによりシリアル・インタフェースとパラレル・インタフェースそしてドライバ・コントロールを選択します。

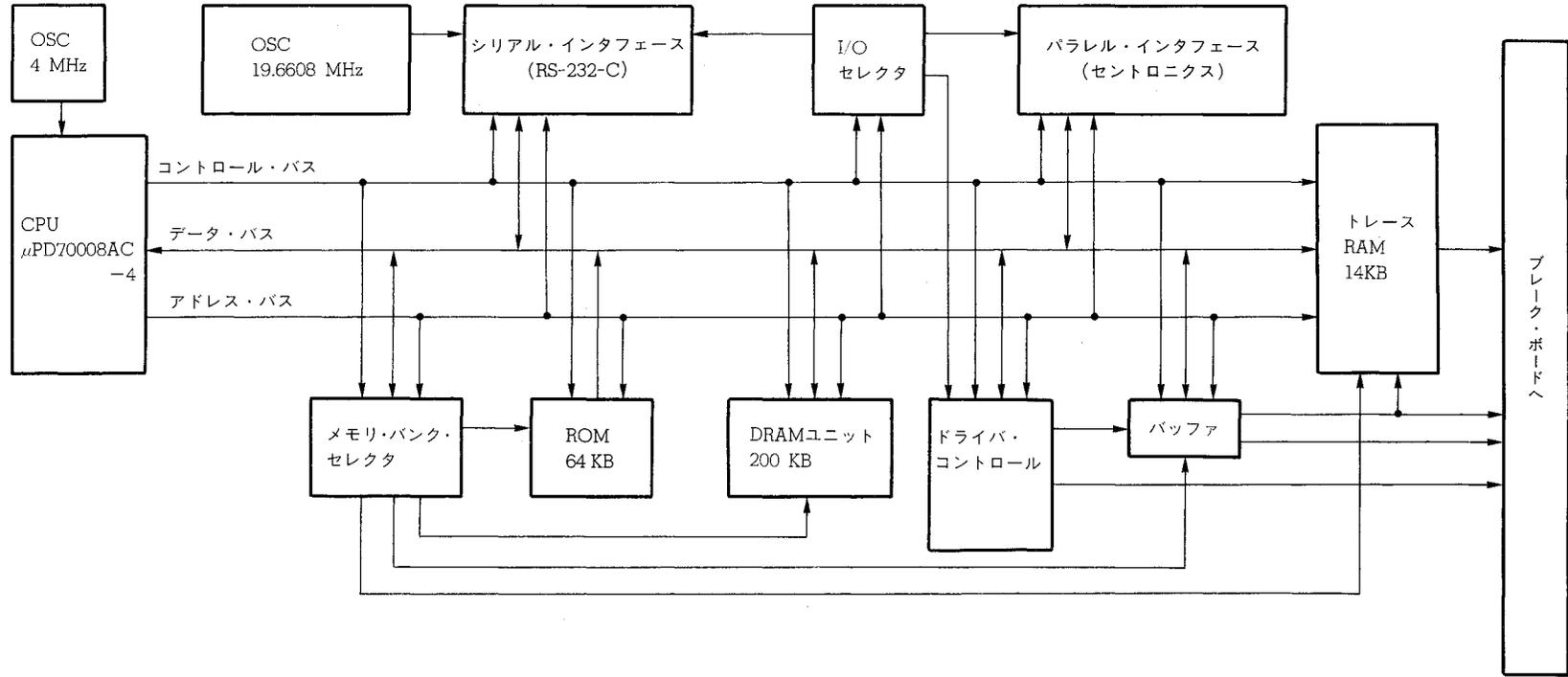
(7) **DRAMユニット**

200 Kバイトのメモリにシンボル用192 Kバイト, プログラム用8 Kバイトのワーク・エリアを持っています。

(8) **ROM**

64 KバイトのROMにIE-75001-Rを起動させるプログラムが入っています。

図 1-2 コントロール/トレース・ボード・ブロック図



1.3.2 ブレーク・ボード・ブロック

ブレーク・ボード・ブロックの構成は次の通りです（図 1-3 参照）。

(1) ブレーク・コントロール

ブレーク条件をコントロールしている部分です。IE-75001-Rは豊富なブレーク機能を持っています。各イベント条件を組み合わせることにより多彩なブレーク条件を設定することができます。

(2) トレース・コントロール

トレース条件をコントロールしている部分です。IE-75001-RはCPUの実行状態を記憶しておくトレース機能を持っています。各イベント条件を組み合わせることにより多彩なトレース条件を設定することができます。

(3) オルタネート・メモリ

スーパーバイザCPUとエミュレーションCPUとが交互にコミュニケーションを行うためのメモリです。

(4) プログラム・メモリ/データ・メモリ

IE-75001-Rは、64 Kバイトのプログラム・メモリと2 Kバイトのデータ・メモリを持っています。ターゲット・システムが開発されていない場合でも、これらのメモリを使用してソフト・ディバグなどを行うことができます。

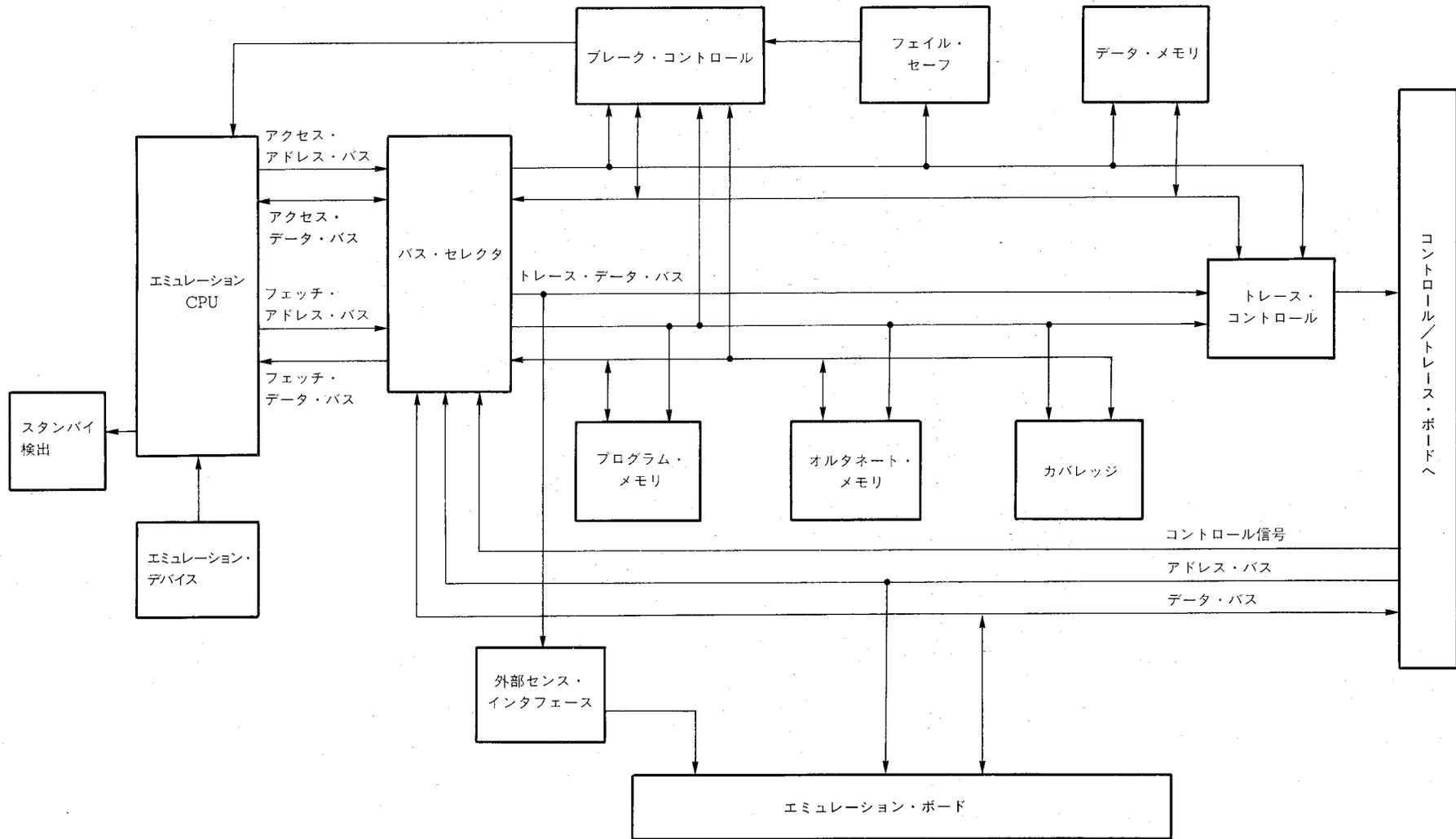
(5) フェール・セーフ

デバイスが持たない内部RAM, SPR, レジスタ, スタックに対して保護などを行う回路です。

(6) カバレッジ

エミュレーション中に実行されたターゲット・プログラムの測定を行い、実行箇所と未実行箇所を表示することができます。

図 1-3 ブレーク・ボード・ブロック図



第2章 各部の名称

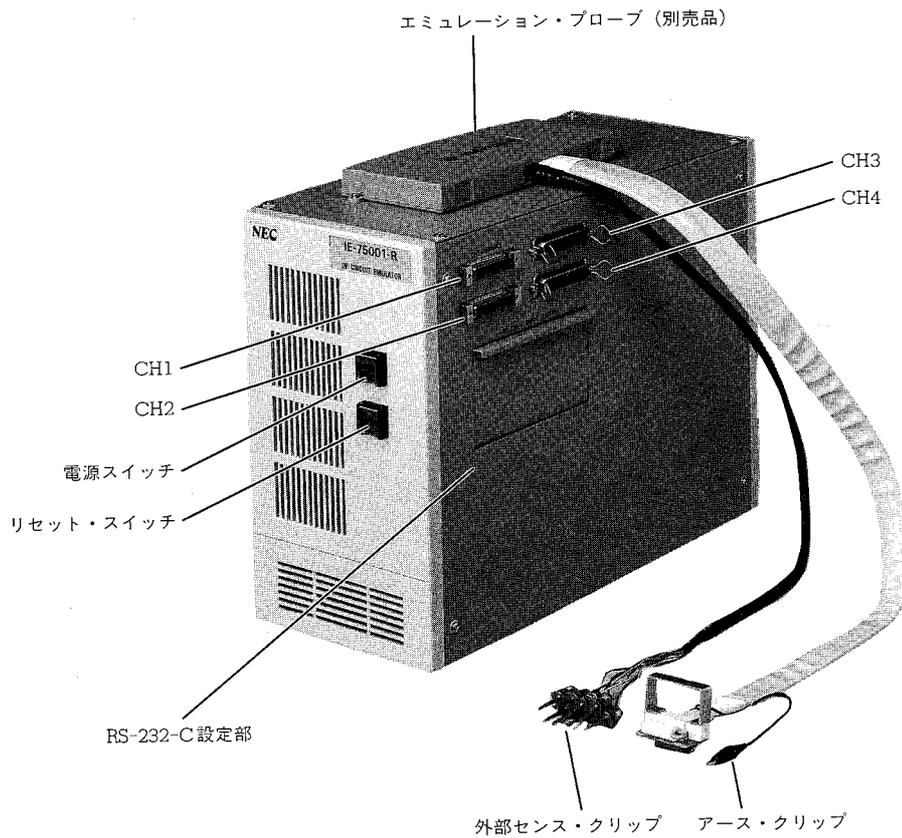
この章では、IE-75001-Rの本体各部および付属品などの名称を紹介します。

梱包箱の中にはIE-75001-R本体と付属品梱包箱が入っています。万一、不足や破損などがありましたら、当社の特約店までご連絡ください。

2.1 本体各部の名称

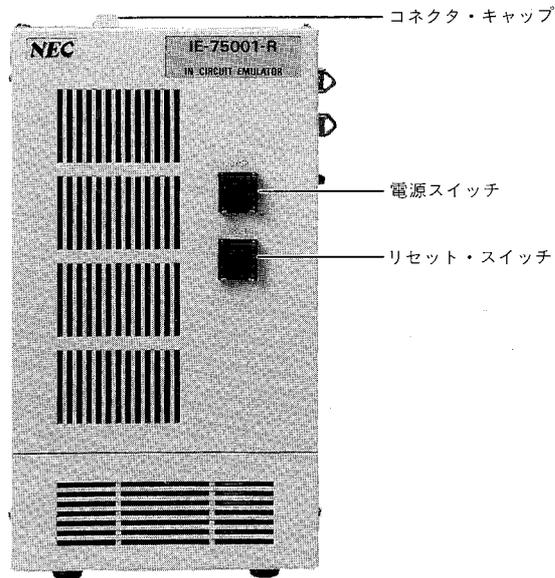
(1) 外観と各部名称

写真 2-1 IE-75001-Rの外観と各部名称



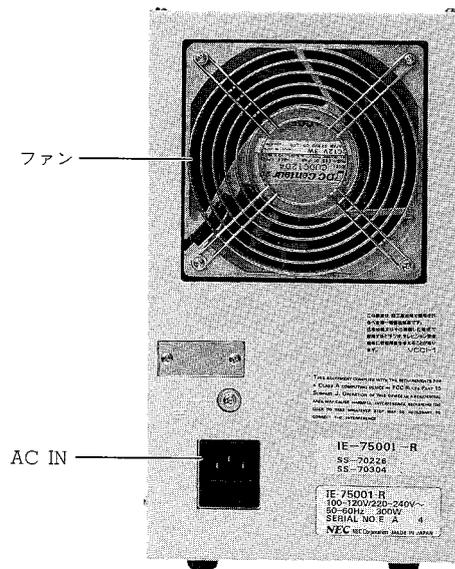
(2) 正面

写真 2-2 IE-75001-Rの正面



(3) 裏側

写真 2-3 IE-75001-Rの裏側



(4) 側面

写真 2-4 IE-75001-Rの側面

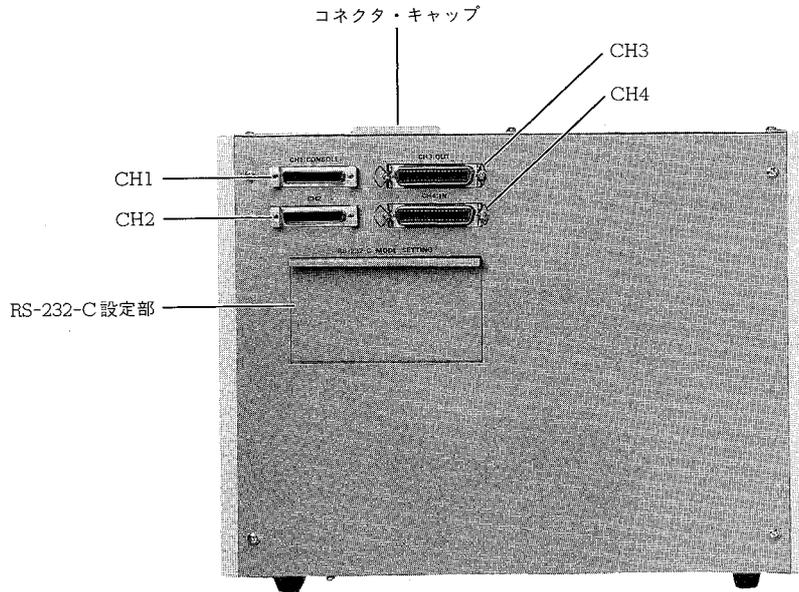
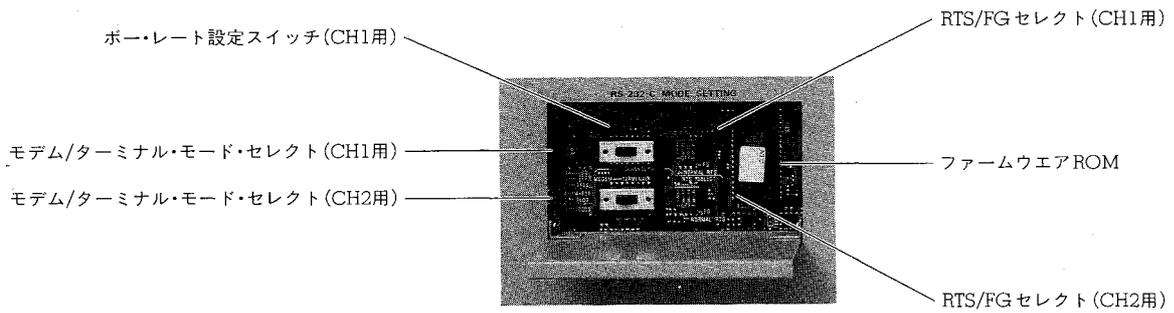


写真 2-5 RS-232-C設定部の拡大図



⑤ ボード

IE-75001-Rの中には次の2枚のボードが入っています。

- (a) ブレーク・ボード 1枚
(IE-75000-R-BK)
- (b) コントロール／トレース・ボード 1枚 (IE-75001-Rに固定)

本体上面のネジ6箇所を外し、フタを開けて確認してください。

写真 2-6 ボード位置

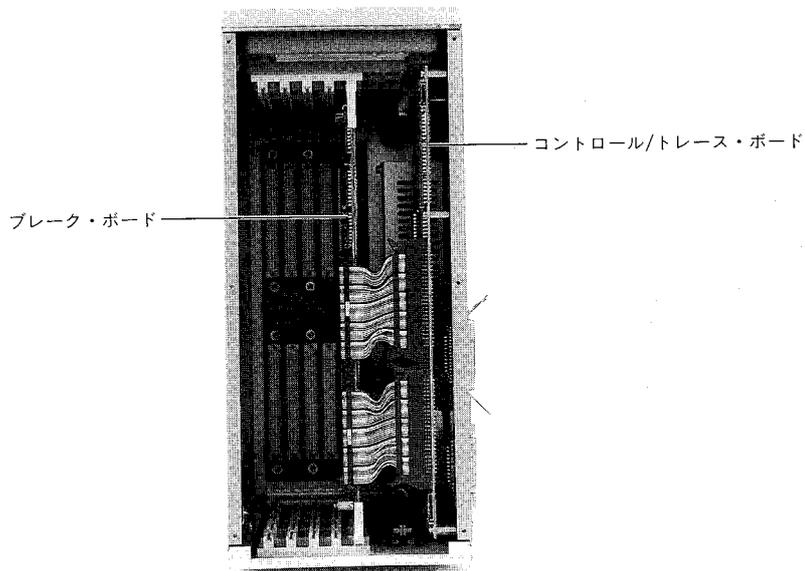


写真 2-7 ブレーク・ボード (IE-75000-R-BK)

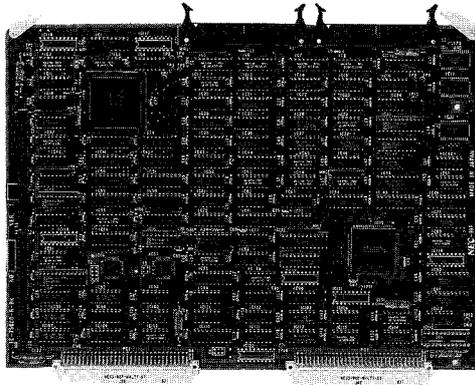
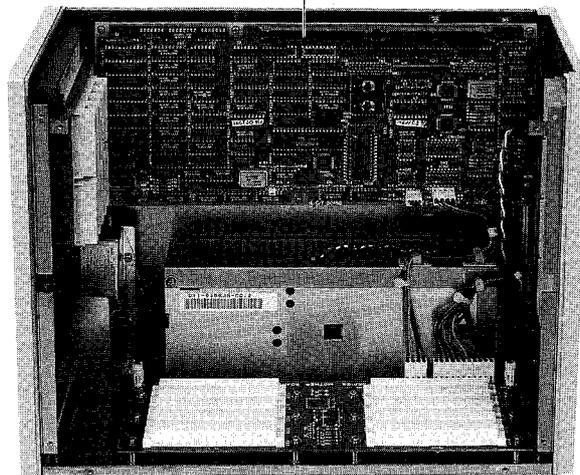


写真 2-8 コントロール/トレース・ボード

コントロール/トレース・ボード (IE-75001-Rに固定)



2.2 付属品

付属品梱包箱の中には次のものが入っています。確認してください。

- | | |
|--|----|
| (a) IE-75000-R/IE-75001-R ユーザーズ・マニュアル(このマニュアル) | 1冊 |
| (b) AC100V用電源ケーブル | 1本 |
| (c) AC200V用電源ケーブル | 1本 |
| (d) RS-232-C インタフェース・ケーブル | 1本 |
| (e) アース・リード・ケーブル | 1本 |
| (f) スペア・ヒューズ | 1個 |
| (g) ACアダプタ | 1個 |
| (h) 添付品リスト | 1通 |
| (i) 保証書 | 1通 |
| (j) 梱包明細書 | 1通 |

第3章 設 置

この章では、IE-75001-Rをエミュレーション・ボードなどと接続し、RS-232-Cの各モード設定を行う方法について解説します。

3.1 設 置

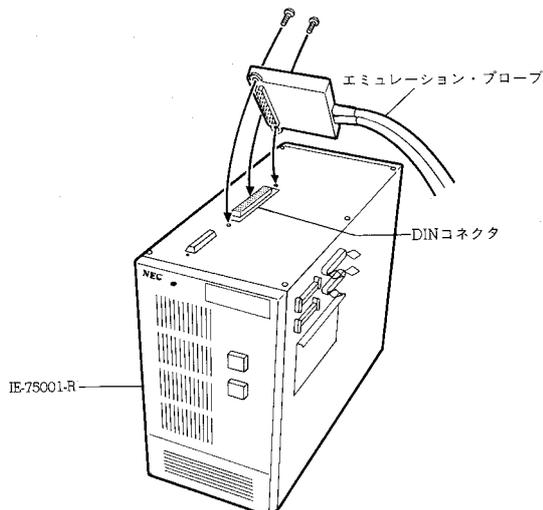
エミュレーション・ボード、エミュレーション・プローブ (EP-75XXXZZ-R, 別売) および付属品ケーブルをIE-75001-R本体に接続します。エミュレーション・プローブは、ターゲット・システムで使用している75Xシリーズのパッケージに合わせてお使いください。

(1) エミュレーション・ボードおよびエミュレーション・プローブの接続方法

- ① IE-75001-Rのネジを外してフタを開け、IE-75000-R-BKをスロットから抜き取ります。
- ② 別売のIE-75000-R-EMまたはIE-75617-R-EMとIE-75000-R-BKをネジ留めします。
- ③ ②で接続した2枚のボードに、エミュレーション・プローブのアダプタ・ボードを接続します。
- ④ ③で接続した3枚のボードを、IE-75001-R筐体の中にインストールします。
- ⑤ 筐体のフタを閉め、ネジを留めます。なお、コネクタ・キャップは接続するエミュレーション・プローブの種類に応じて、取り外してください。
- ⑥ エミュレーション・プローブをコネクタに接続し、ネジでしっかり留めます。

注意 接続方法を間違えますと、IE-75001-R本体が破壊されることがあります。なお、接続の詳細については、エミュレーション・ボード (IE-75000-R-EMまたはIE-75617-R-EM) および、各エミュレーション・プローブのユーザーズ・マニュアルを参照してください。

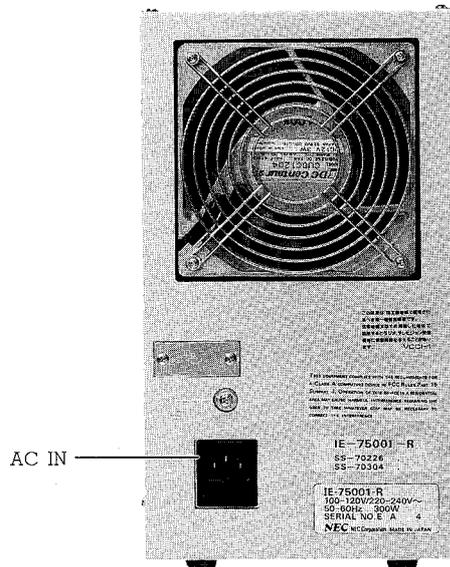
図 3-1 エミュレーション・プローブの接続



(2) 電源ケーブルの接続方法

IE-75001-R本体裏側のAC INに電源ケーブルを差し込んでください。

写真 3-1 IE-75001-Rの裏側 (AC IN)



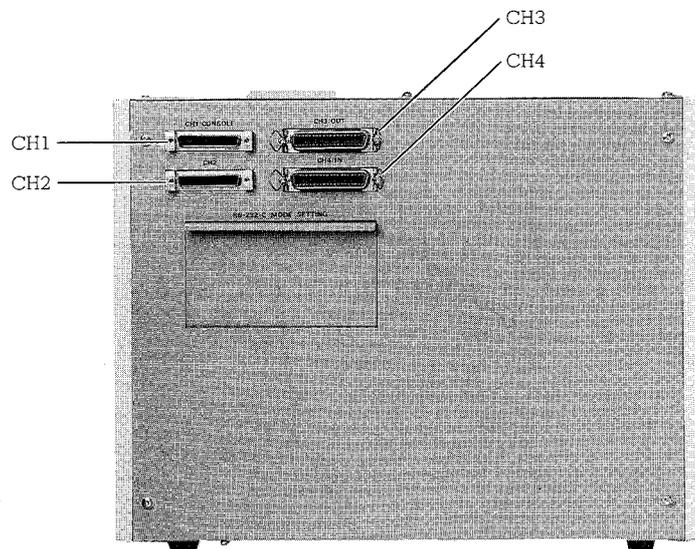
(3) RS-232-Cインタフェース・ケーブルの接続方法

IE-75001-R本体側面のCH1 (ホスト・マシン接続時) またはCH2 (PG-1500, 1000接続時) のコネクタに差し込みます。

(4) パラレル・インタフェース・ケーブルの接続方法

IE-75001-R本体側面のCH3 (プリンタ接続時) またはCH4 (ホスト・マシン接続時) のコネクタに差し込みます。

写真 3-2 IE-75001-Rの側面 (インタフェース・ケーブルの接続)



3.2 RS-232-Cモード設定

RS-232-Cモード・スイッチは、IE-75001-R本体の側面カバーの中にあります。設定を行う場合は、側面カバーを開いて行ってください。

図 3-2 RS-232-Cの各モード・スイッチ

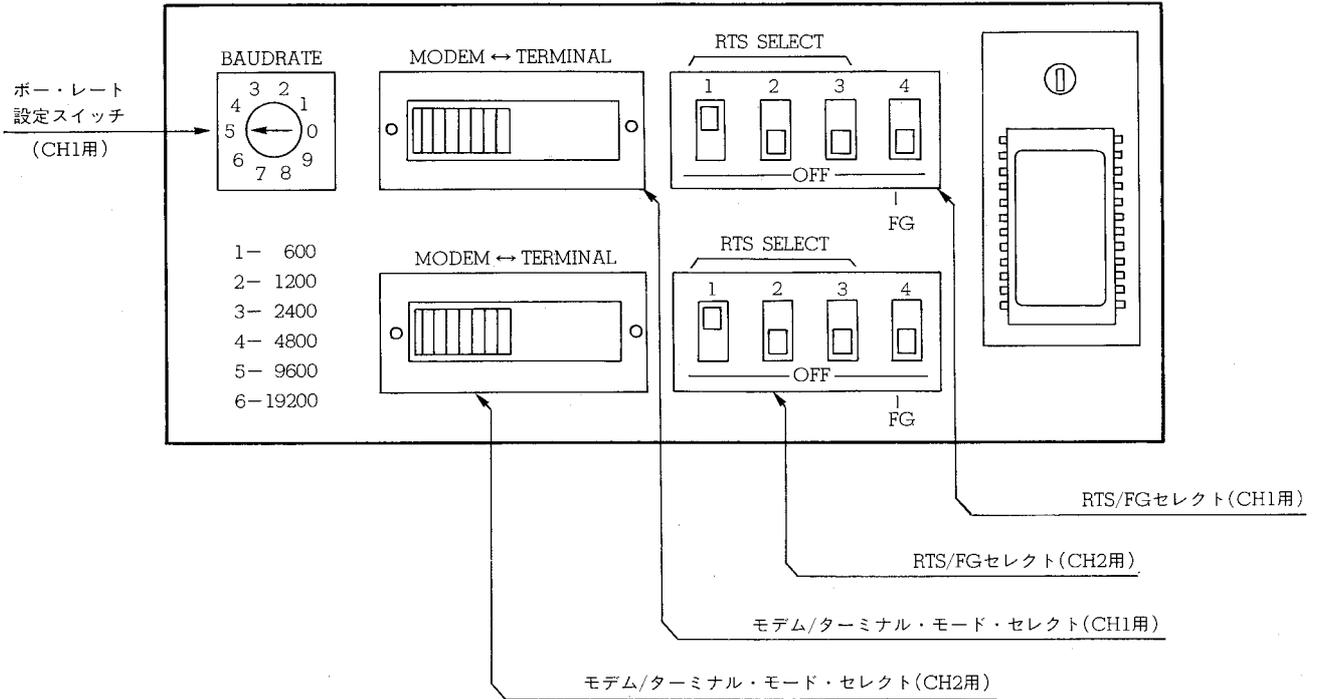
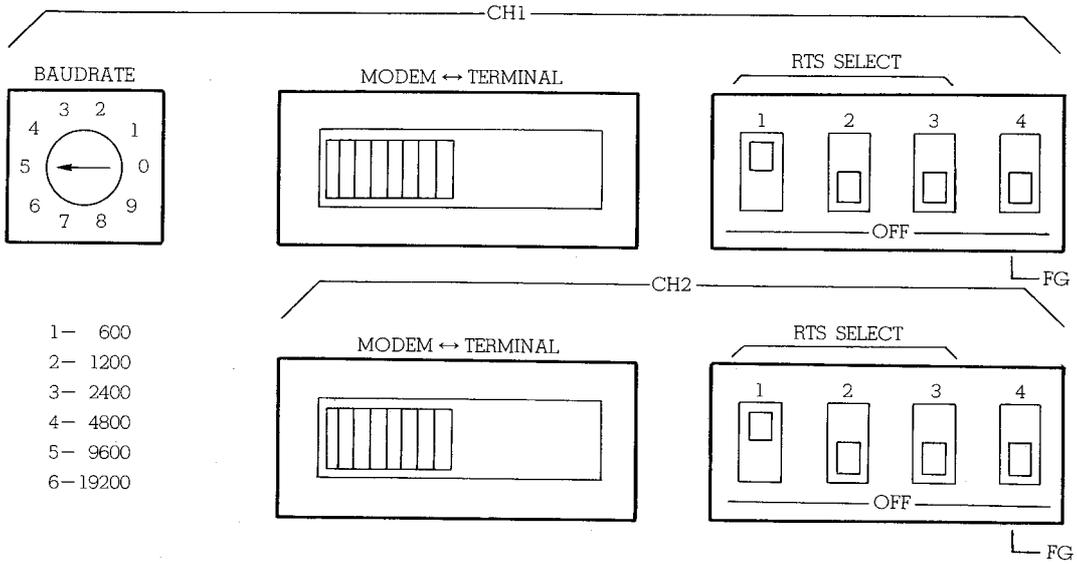


図 3-3 出荷時の各モード設定



モデム/ターミナル・モードの設定

スライド・スイッチを使用しており、側面パネルに向かって左側にスライドするとモデム・モードに、右側にスライドするとターミナル・モードになります。
出荷時は、**モデム・モード**に設定してあります。

RTSの設定

DIPスイッチを使用しており、上側にスライドするとON、下側にスライドするとOFFとなります。
RTSは1-3番スイッチで設定します。
出荷時は、**1番：ON, 2, 3番：OFF**に設定してあります。

FG (フレーム・グラウンド) の設定

DIPスイッチを使用しており、上側にスライドするとON、下側にスライドするとOFFになります。
FGは、4番スイッチで設定します。
出荷時は**4番：OFF (FGとSGがオープン状態)**に設定してあります。

ボー・レートの設定

マイクロDIPスイッチを使用しています。

スイッチ・ポジションは、0-9まで10ポジションあり“7, 8, 9”は使用しません。

なお、このスイッチはCH1のボー・レート設定用で、出荷時は“5” (9600 bps) に設定してあります。

CH2のボー・レートの設定は“MOD”コマンドで設定します。

詳しくは、第8章 コマンドの説明 を参照してください。

キャラクタ仕様の設定

(表 3-1 参照)

CH1は設定の変更はできません。

CH2は“MOD”コマンドにより設定します。

表 3-1 キャラクタ仕様

キャラクタ仕様	CH1 (不変)	CH2 (コマンドによる)
キャラクタ長	8ビット	7/8ビット
パリティ・ビット	なし	偶数/奇数/なし
ストップ・ビット長	2ビット	1/2ビット

3.3 コントロール／トレース・ボードの設定

コントロール／トレース・ボードの出荷時のジャンパ設定は次のとおりです。ジャンパはすべて出荷時のままにしておいてください。

通常の使用においては設定の変更を行う必要はありません。

表 3-2 コントロール／トレース・ボードのジャンパ設定（出荷時）

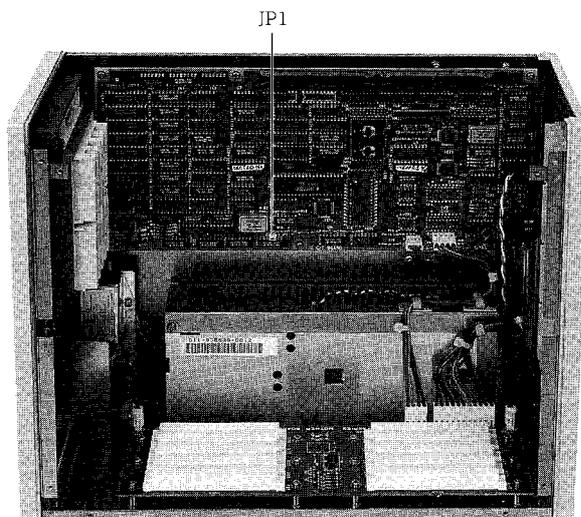
ジャンパNo.	設 定
JP1	1-2 ショート

注意 出荷時以外の設定にしますと正常に動作しません。

備考 コントロール/トレース・ボードとIE-75001-Rは次の手順で接続されています。

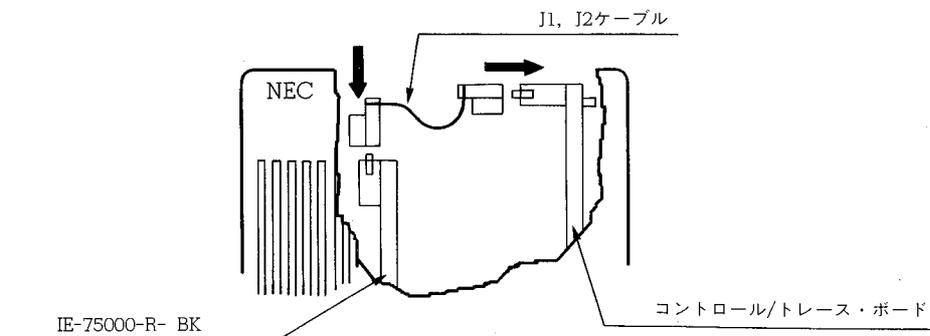
- ① IE-75001-R本体上面のネジ6箇所を外してフタを開けます。
- ② IE-75000-R-BKとコントロール/トレース・ボードを接続しているケーブル (J1, J2ケーブル) を外します。
- ③ スロットにある基板はすべて抜き取ります (ボードの両端のカード・プラーを手前に引き抜きます)。

写真 3-3 IE-75001-Rの内部



- ④ J1, J2ケーブルを再び取り付けるときは、下図のように接続します。

図 3-4 J1, J2ケーブル接続図



第4章 システム構成

この章では、IE-75001-Rと次に示す周辺装置の接続方法について述べます。周辺装置との接続の際には、本章を必ずお読みください。なお、ここではIE-75001-Rにはすでに、IE-75000-R-EMが接続されているものとして説明しています。

ホスト・マシン	PC-9800シリーズ IBM PCシリーズ
PROMプログラマ	PG-1500 PG-1000 (廃品種のため、新規のご購入はできません)

PC-9800シリーズ

PC-9800シリーズは別売のコントロール・プログラムをMS-DOS™上で動作させることにより、ソフトウェア開発からハードウェアを含む総合評価までの一貫した開発環境を提供します。

IBM PCシリーズ

IBM PCシリーズは別売のコントロール・プログラムをPC DOS™上で動作させることにより、ソフトウェア開発からハードウェアを含む総合評価までの一貫した開発環境を提供します。

PG-1500

PG-1500は、256 Kビットから、4 Mビットまでの代表的PROMのプログラミングが可能なPROMプログラマです。別売のソケット・アダプタを使用することにより、NECのシングルチップ・マイコンに内蔵されているPROMのプログラミングが可能です。

さらにPG-1500は、キー・パネル・スイッチとシリアル・インタフェースを持っていますので、スタンドアローン・タイプのPROMプログラマとして動作できます。またシリアル・インタフェースに接続したコンソールを通じて、リモート動作も可能です。

IE-75001-Rと接続する場合は、汎用のRS-232-Cインタフェース・ケーブルを使用してください。

PG-1000

PG-1000はオプションのパーソナル・モジュールを交換することにより、PROM内蔵シングルチップ・マイコンやバイポーラPROMのプログラミングが可能なPROMプログラマです。キー・パネル・スイッチとシリアル・インタフェースを持っていますので、スタンドアローン・タイプのPROMプログラマとして動作が可能です。またシリアル・インタフェースに接続したコンソールを通じて、リモート動作も可能です。

IE-75001-Rと接続する場合は、PG-1000付属のRS-232-Cインタフェース・ケーブルを使用してください。

4.1 ホスト・マシンとの接続

4.1.1 PC-9800シリーズとの接続

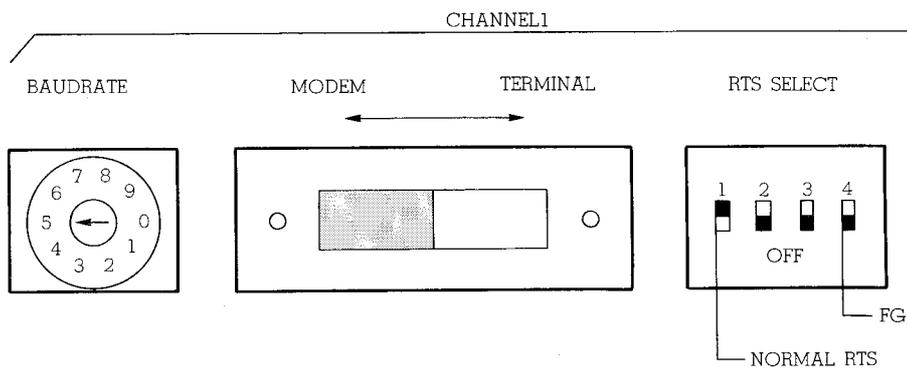
PC-9800シリーズとIE-75001-Rとの接続は、次に示す順番で行ってください。

- ① PC-9800シリーズおよびIE-75001-Rの電源は切った状態で設定を行います。
- ② IE-75001-Rのチャンネル1を次のように設定してください。

表 4-1 チャンネル1の設定 (PC-9800シリーズとの接続)

設定項目	設定内容
ボー・レート	9600 bps
モード	モデム・モード
RTS	標準RTS

図 4-1 チャンネル1の設定 (PC-9800シリーズとの接続)



- ③ PC-9800シリーズを次のように設定します。

表 4-2 PC-9800シリーズの設定

設定項目	設定内容
ボー・レート	9600 bps
キャラクタ長	8ビット
パリティ・チェック	なし
ストップ・ビット	2ビット
X-ON/X-OFF	NONE

- ④ PC-9800シリーズ本体の背面にある標準RS-232-CチャンネルとIE-75001-RのCH1をIE-75001-R付属のRS-232-Cケーブルで接続してください。また、パラレル・インタフェースを使用する場合は、IE-75001-RのCH4とPC-9800シリーズ本体の背面にあるプリンタ用コネクタをPC-9800用のプリンタ・ケーブルで接続してください。

表 4-3 PC-9800シリーズとの接続

IE-75001-R	接 続	PC-9800シリーズ本体
CH1	RS-232-Cケーブル ←—————→	標準RS-232-Cチャンネル
CH4	プリンタ・ケーブル ←—————	プリンタ用コネクタ

- ⑤ PC-9800シリーズ, IE-75001-Rの順番に電源を投入してください。なお, 電源を切るときにはIE-75001-R, PC-9800シリーズの順番に行ってください。

4.1.2 IBM PCシリーズとの接続

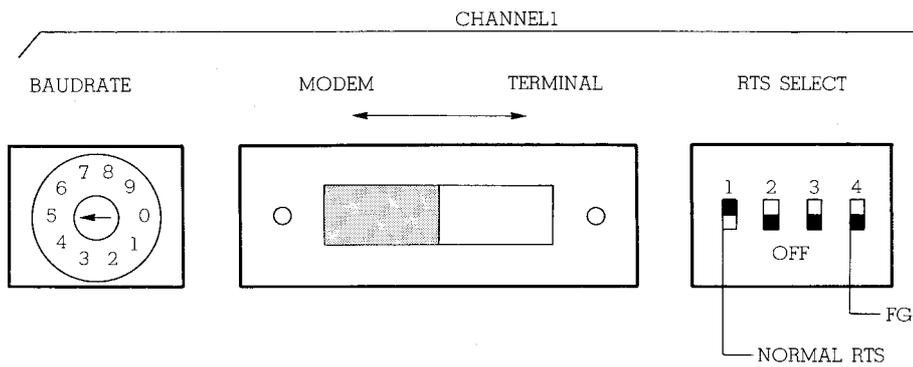
IBM PCシリーズとIE-75001-Rの接続は、次に示す順番で行ってください。

- ① IBM PCおよびIE-75001-Rの電源は切った状態で設定を行います。
- ② IE-75001-Rのチャンネル1を次のように設定してください。

表 4-4 チャンネル1の設定 (IBM PCシリーズとの接続)

設定項目	設定内容
ボー・レート	9600 bps
モード	モデム・モード
RTS	標準RTS

図 4-2 チャンネル1の設定 (IBM PCシリーズとの接続)



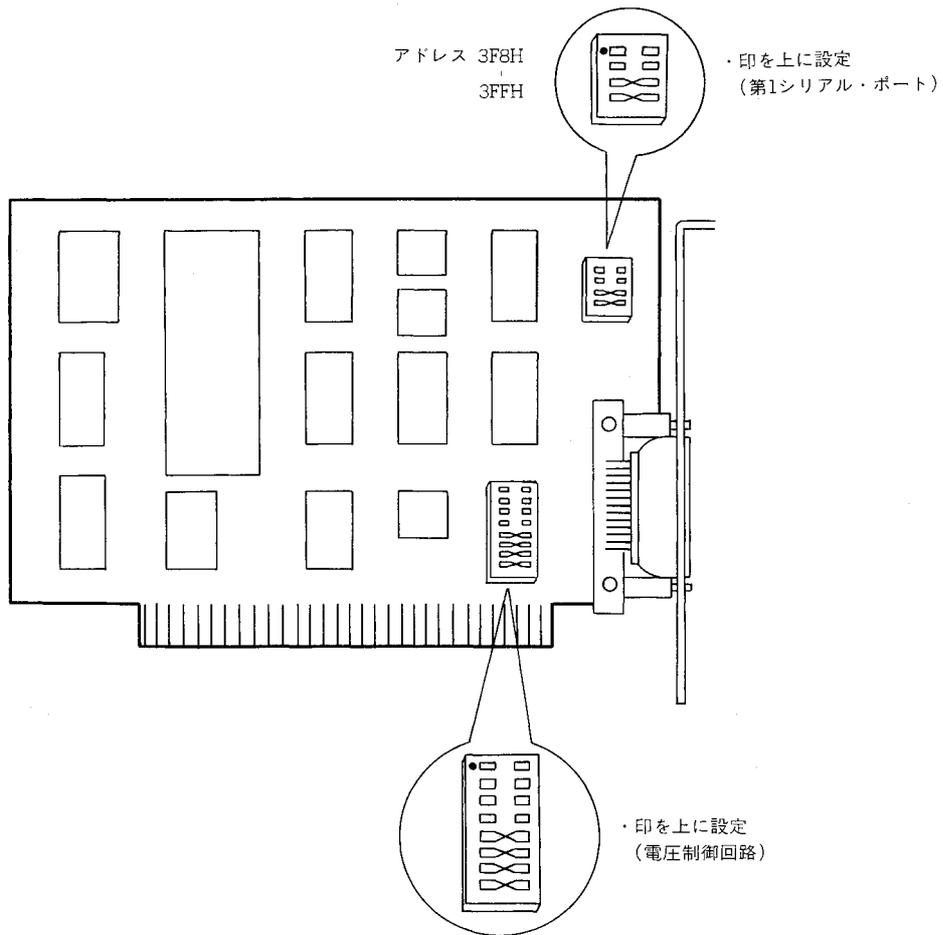
- ③ IBM PCを次のように設定してください。

表 4-5 IBM PCの設定

設定項目	設定内容
ボー・レート	9600 bps
パリティ	なし(NONE)
データ・ビット	8ビット
ストップ・ビット	2ビット

- ④ 図4-3に示すようにIBM PC内に挿入されているアシンクロナス・コミュニケーション・アダプタの設定を行ってください。本コントロール・プログラムは第1 (No.0) のシリアル・ポートのみサポートします。

図4-3 アシンクロナス・コミュニケーション・アダプタの設定



- ⑤ IBM PCオプションのアシクロナス・コミュニケーション・アダプタのRS-232-CチャンネルとIE-75001-RのCH1を**IBM PC用RS-232-Cケーブル**で接続してください（図4-4参照）。また、パラレル・インタフェースを使用する場合は、IE-75001-RのCH4とIBM PC本体の背面にあるプリンタ用コネクタを**IBM PC用のプリンタ・ケーブル**で接続します。

図4-4 IBM PCとのRS-232-C接続

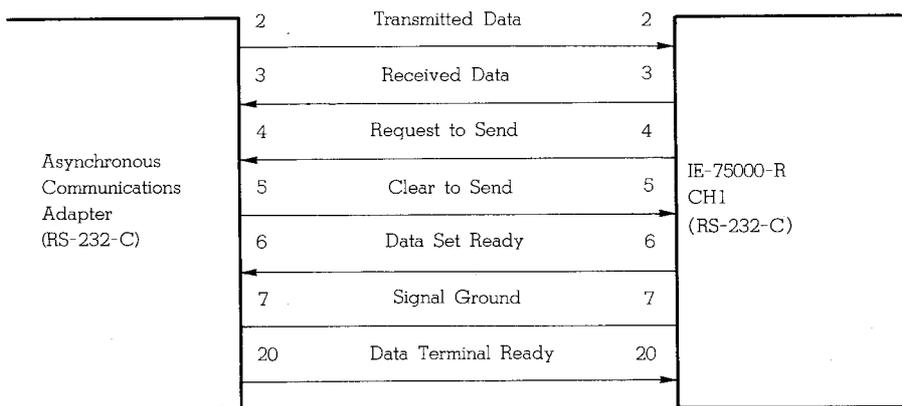
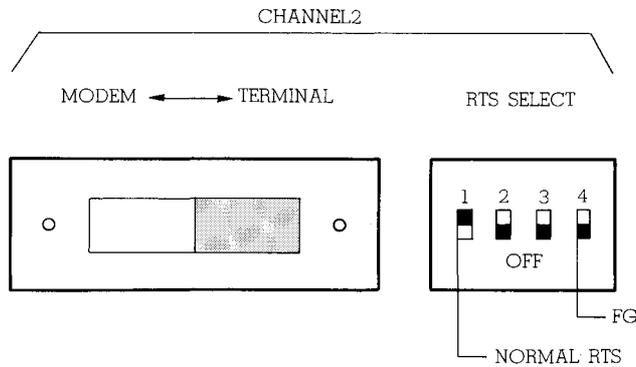


表4-6 IBM PCとの接続

IE-75001-R	接 続	IBM PC本体
CH1	RS-232-Cケーブル ←————→	アシクロナス・コミュニケーション・アダプタのRS-232-Cチャンネル
CH4	プリンタ・ケーブル	プリンタ用コネクタ

- ⑥ IBM PC, IE-75001-Rの順番に電源を投入してください。なお、電源を切るときにはIE-75001-R, IBM PCの順番に行ってください。

図 4-5 チャンネル2の設定 (PG-1500との接続)



- ⑤ PG-1500の設定を、キー入力により行います。詳しい設定方法は**PG-1500 ユーザーズ・マニュアル**を参照してください。

表 4-9 PG-1500の設定

設定項目	設定内容	表示の記号	パラメータ
ボー・レート	1200 2400 4800 <input type="checkbox"/> 9600 19200 (bps)	BR	9600
パリティ	<input type="checkbox"/> NON(なし) EVN (偶数) ODD (奇数)	P	NON
X-ON/X-OFF制御	<input type="checkbox"/> ON OFF	XN	OFF
ビット構成	7 <input type="checkbox"/> 8	B	8
ストップ・ビット	1 <input type="checkbox"/> 2	SB	2
プリチェック	ON <input type="checkbox"/> OFF	PC	OFF

備考 設定内容の欄で、 で囲んであるものが、出荷時の初期値です。

4.2.2 PG-1000との接続

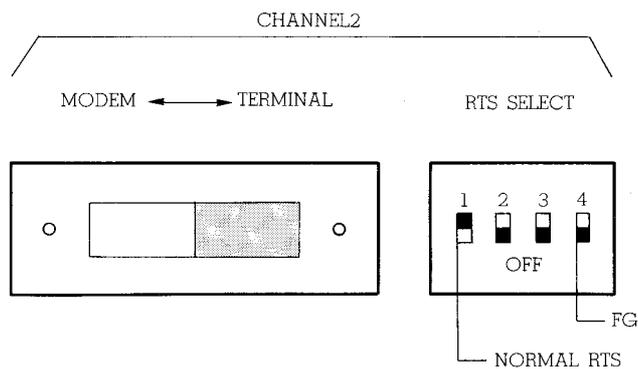
IE-75001-RにPG-1000を接続する場合、次の手順で行ってください。

- ① IE-75001-R, PG-1000の電源を切ります。
- ② IE-75001-Rのチャンネル2の設定を次のようにします。ハンドシェーク方式, ポー・レートおよびキャラクタ仕様の設定は **8.4.23 チャンネル2・モード設定コマンド (MOD)** を参照してください。

表 4-10 チャンネル2の設定 (PG-1000との接続)

設定項目	設定内容	
モード切り替え	ターミナル・モード	
フレーム・グラウンド	4番ピン OFF	
RTSセレクト	1番ピン ON	
	2番ピン, 3番ピン OFF	
ハンドシェーク方法	CHAR (コマンド設定)	
ポー・レート	9600	
キャラクタ仕様	データ長	8
	パリティ・ビット	なし
	ストップ・ビット	2

図 4-6 チャンネル2の設定 (PG-1000との接続)



- ③ PG-1000の設定は、本体の底面にある4連DIPスイッチと8連DIPスイッチにより行います。次のように各DIPスイッチを設定してください。

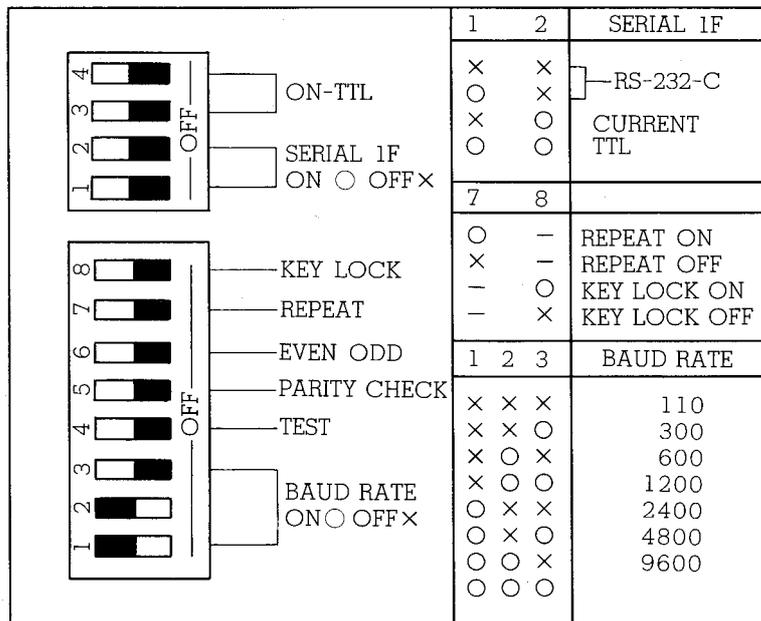
表 4-11 PG-1000の設定

設定項目	スイッチNo.	スイッチの設定
シリアル・インタフェース・モード	4連スイッチ 1~4	1~4 : ON
ポー・レート	8連スイッチ 1~3	1, 2 : ON 3 : OFF
パリティ・チェック	5, 6	5 : OFF 6 : ON/OFF注

注 ON/OFFどちらでもかまいません。

注意 ビット4, 7, 8は絶対にONにしないでください。

図 4-7 PG-1000の設定 (底面DIPスイッチ)



- ④ IE-75001-RのCH2とPG-1000のシリアル・インタフェース・コネクタを**PG-1000**付属のケーブルを用いて接続します。

注意 必ず**PG-1000**付属のケーブルを用いて接続してください。

表 4-12 PG-1000との接続

IE-75001-R	接 続	PG-1000
CH2	専用RS-232-Cケーブル ←—————→	シリアル・インタフェース ・コネクタ

- ⑤ PG-1000, IE-75001-Rの順序で電源を入れます。

第5章 操作概要

この章では、第4章までに説明した各周辺装置の設定が完了したことを前提に、IE-75001-Rシステムの動作環境と電源投入からシステムの終結、電源の切断までの、一連の操作手順について説明します。

5.1 システムの動作環境

IE-75001-Rは、ホスト・マシン（PC-9800シリーズ、IBM PCシリーズ）と接続し、ホスト・マシン上で動作するコントロール・プログラム（IE75000.COM）によって制御されます（図5-1 参照）。

図 5-1 システムの動作環境

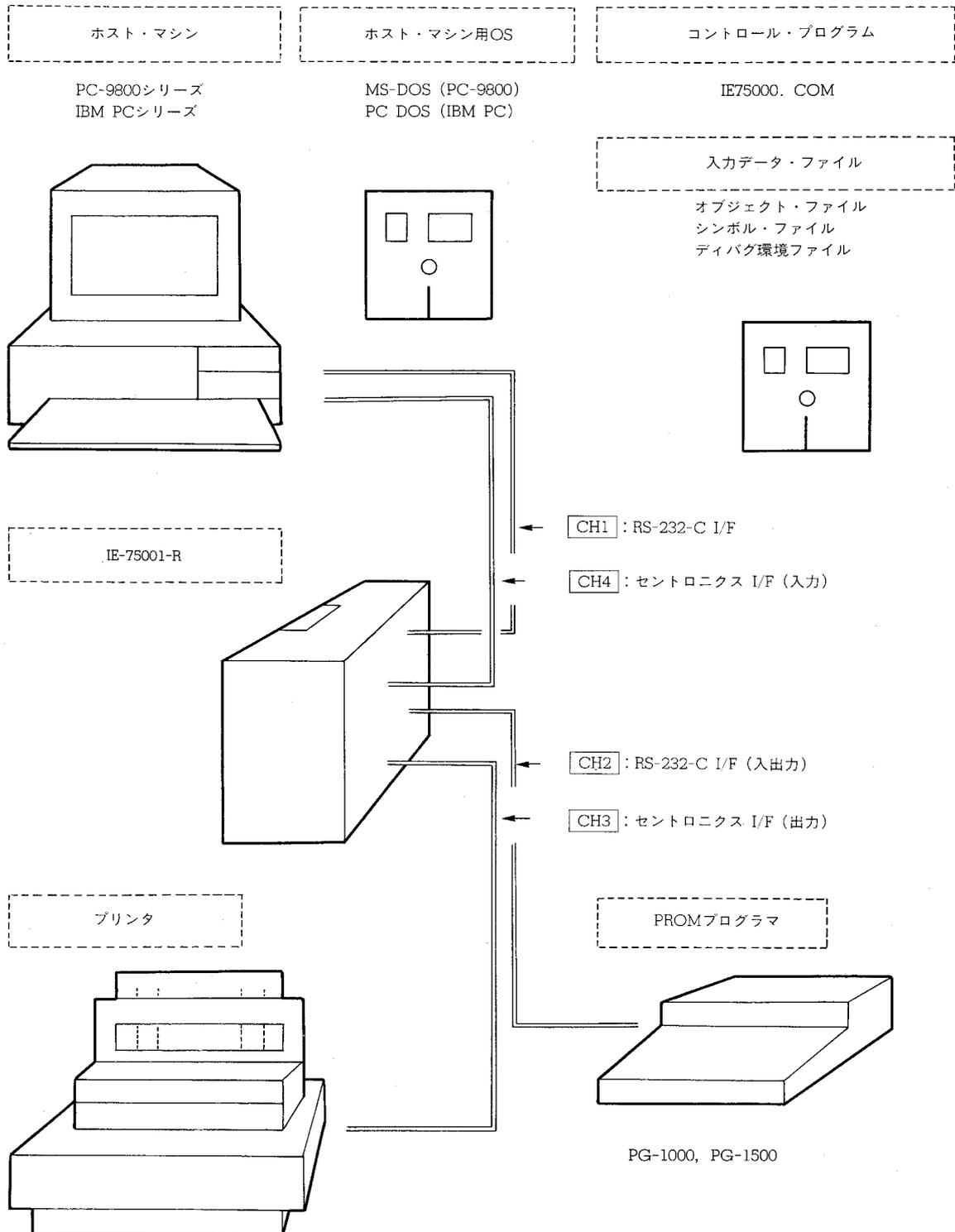


表 5-1 ハードウェア環境

使用機器名称	チャンネル, 周辺装置	用 途
IE-75001-R	CH1(入出力用) (RS-232-C インタフェース)	ホスト・マシンに接続します。 ホスト・マシンとIE-75001-R相互間の, 制御データ授受に使用します。 またオブジェクト・ファイルなどの, 通常ダウン・ロードやアップ・ロードにも使用します。
	CH2(入出力用) (RS-232-C インタフェース)	PROMプログラマ(PGシリーズなど)に接続します。 IE-75001-RとPROMプログラマ相互間の, 制御データ授受に使用します。
	CH3(出力専用) (セントロニクス・インタフェース)	プリンタなどに接続します。 ホスト・マシンがプリンタ出力を行うときに使用します。 CH4(セントロニクス・インタフェース)の入力データをスルー(透過)出力することができます。
	CH4(入力専用) (セントロニクス・インタフェース)	ホスト・マシンに接続します。 ホスト・マシンからオブジェクト・ファイルなどを高速ダウン・ロードするときに使用します。
ホスト・マシン	キーボード	コマンド入力など, すべての操作に使用します。
PC-9800 シリーズ (LT, XTを除く) または IBM PC シリーズ	FDD	次の用途で使用します。 ① OSのロード ② コントロール・プログラムのロード ③ データ・ファイルのロード/セーブ ・オブジェクト・ファイル ・シンボル・ファイル ・ディバグ環境ファイル

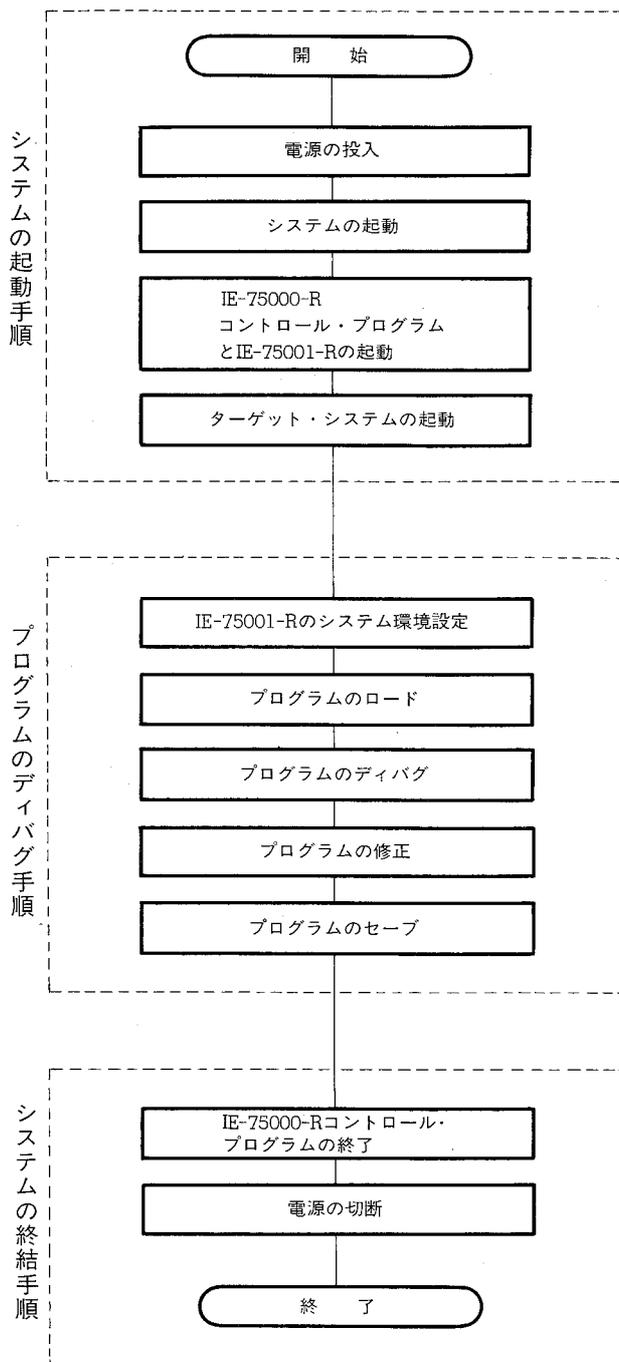
表 5-2 ソフトウェア環境

使用ソフトウェア名称	用 途
MS-DOS(Ver 3.30~)	PC-9800シリーズ(LT, XTを除く)を, ホスト・マシンとして使用する場合のOSです。
PC DOS(Ver 3.10~)	IBM PCシリーズを, ホスト・マシンとして使用する場合のOSです。
IE-75001-R コントロール・プログラム (ファイル名: IE75000.COM)	IE-75001-Rを制御するためのソフトウェアです。 ホスト・コンピュータ上で動作します。
オブジェクト・ファイル (ファイル名: ××××××××.HEX)	ターゲット・プログラムのオブジェクト・コードを格納するファイルです。
シンボル・ファイル (ファイル名: ××××××××.SYM)	ターゲット・プログラムに対して, ユーザが定義したシンボルを格納するファイルです。 注意 このファイルはなくてもかまいませんが, そのときシンボルの処理はできません。
ディバグ・環境ファイル (ファイル名: ××××××××.DBG)	ディバグ環境の情報を格納するファイルです。 このファイルによって, 次のコマンドを環境設定することができます。 BNK, BRA1, BRA2, BRA3, BRA4, BRK, BRM, BRS1, BRS2, CHK, CLK, DLY, MOD, OUT, PAS, PGM, REG, SET, STS, TRF, TRM, TRX, TRY 注意 このファイルはなくてもかまいませんが, そのとき環境設定はできません。

5.2 操作手順

IE-75001-Rシステムの一連の操作手順概略は次のようになります。

図 5-2 操作手順概略フロー・チャート



5.2.1 システムの起動手順

(1) 電源の投入

次の手順で、電源の投入を行います。電源の投入は主電源から末端装置への順序で行い、切断はこれと逆の順序で行います。

- ① ホスト・マシンの電源を投入します。
- ② IE-75001-Rの電源を投入します。
- ③ PGシリーズが接続されている場合は、PGシリーズの電源を投入します。
- ④ プリンタが接続されている場合は、プリンタの電源を投入します。

(2) システムの起動

システムの起動は、MS-DOS (PC-9800シリーズ用のOS) を使用する場合と、PC DOS (IBM PCシリーズ用のOS) を使用する場合^注とは、操作が異なります。

① OSの起動

ホスト・マシンのOSを起動してください。ドライブの選定は任意です。

② ホスト・マシンのシリアル・チャンネル設定

ホスト・マシンのシリアル・チャンネルを次表の条件に設定します。シリアル・チャンネルの設定に使うコマンドは次ページに示すとおり、MS-DOSとPC DOSでは異なります。

表 5-3 ホスト・マシンのシリアル・チャンネルの設定

設定項目	設定内容
ボー・レート	9600 bps
キャラクタ長	8ビット
パリティ	なし
ストップ・ビット	2ビット
X-ON/X-OFF	いずれでもよい

^注 PC DOSを使用する場合、システムを起動するときは必ずCONFIG.SYSにANSI.SYSを組み込んでください。

備考 モニタ画面内の入力キーの記述は次のとおりです。

XXXXXXXX : キー入力を示します。

 : リターン・キー

 : エスケープ・キー

^ : コントロール・キー ( キー)

^X :  キーと  キーのダブル・タッチ

MS-DOSでの設定

SPEEDコマンドを使用して、シリアル・チャンネルを設定します。

<pre>A> <u>SPEED</u> SPEED Version X.X RS232C-0 XXXX BITS-X PARITY-XXX STOP-X XXXX - <u>RS232C-0 9600 BITS-8 PARITY-NONE STOP-2</u> A> <u>^C</u></pre>	<p>①</p> <p>②</p>
--	-------------------

- ① SPEEDコマンドを入力します。
- ② シリアル・チャンネルを設定します。

注意 MS-DOSを使用する場合、システムを起動するときは必ずCONFIG.SYSにRSDRV.SYSを組み込んでください。

PC DOSでの設定

MODEコマンドを使用して、シリアル・チャンネルを設定します。

<pre>A> <u>MODE COM1 : mode</u> <u>COM1 : XXXX.X.X.X-</u></pre>	<p>①</p> <p>②</p>
--	-------------------

- ① MODEコマンドを入力します。
- ② シリアル・チャンネルを設定します。

③ IE-75000-Rコントロール・プログラムとIE-75001-Rの起動

IE-75000-Rコントロール・プログラム (IE75000.COM) を起動するとともにIE-75001-Rの起動を次の手順で行います。この操作で、ホスト・マシンとIE-75001-Rが論理的に接続されます。

- ① IE-75000-Rコントロール・プログラム (IE75000.COM, IE75000.OV1, IE75000.OV2, IE75000.OV3, IE75000.OV4) を、ホスト・マシンの任意のフロッピー・ディスク・ドライブにセットします。
- ② カレント・ドライブの指定を、コントロール・プログラムをセットしたドライブに変更します。
- ③ コントロール・プログラム名称 (IE75000) を入力します。

MS-DOSでの指定画面

A>B: <input type="checkbox"/>	①
B>IE75000 <input type="checkbox"/>	②
IE-75000 CONTROLLER (PC-9800 SERIES) Vx.x [DD Mmm YY] Copyright(C) YYYY by NEC Corporation	③
IE-75000/1-R Monitor V1.4 [1 Aug 92] Copyright(C) 1989, 1992 by NEC Corporation	④

- ① ドライブを変更します。
- ② コントロール・プログラム名称 (IE75000) を入力します。
- ③ IE75000のバージョンNo.を示します。
- ④ IE-75001-Rのモニタ・バージョンNo.を示します。

PC DOSでの指定画面

A>B: <input type="checkbox"/>	①
B>IE75000 <input type="checkbox"/>	②
IE-75000 CONTROLLER (IBM PC SERIES) Vx.x [DD Mmm YY] Copyright(C) 1989 by NEC Corporation	③
IE-75000/1-R Monitor V1.4 [1 Aug 92] Copyright(C) 1989, 1992 by NEC Corporation	④

- ① ドライブを変更します。
- ② コントロール・プログラム名称 (IE75000) を入力します。

- ③ IE75000のバージョンNo.を示します。
- ④ IE-75001-Rのモニタ・バージョンNo.を示します。

(4) ターゲット・システムの起動

ターゲット・システムを接続してデバッグするときは、次の手順でターゲット・システムを起動します。この操作によって、IE-75001-Rとターゲット・システムが論理的に接続されます。まず、ターゲット・システムの電源を投入し、次にキー入力を行います。

```

Self check ok

Target CPU      uPD75104/104A/106
Program Memory  0-FFFFH
Data Memory     00H-13FH, F80-FFFFH
Memory Bank     0-1, 15
Register Bank   0-3
Power on target system (Y/N) Y
  Create new set up mode (Y or N) : Y
  Do you use high speed down load mode? (Y/N) = N
  Lod object file name = TEST C S
brk : 0 >
    
```

- ① IE-75000-R-EMのDIPスイッチ (SW1) で設定された対象デバイスを選択した状態で起動します。ここでは、 μ PD75104, 104A, 106が設定された状態になっています。
- ② “Y” を入力します。
- ③ “Y” を入力します。
- ④ “N” を入力します。

(5) セットアップ・ファイルの設定

セットアップ・ファイルの設定および作成を行います。

注意 セットアップ・ファイルの設定は、コントロール・プログラムのVer.1.1, ファームウェアROMのVer.1.4の組み合わせで使用してください。コントロール・プログラムのVer.1.1は、ファームウェアROMのVer.1.0-1.3でも使用できますが、自動で設定できるのは高速ダウン・ロードの指定のみとなります。
そのため、コントロール・プログラムのVer.1.1を使用する場合には必ず、ファームウェアROMのVer.1.4を使用してください。

★

(a) セットアップ・ファイルを作成するとき

Create new set up mode (Y or N) : <u>Y</u>	①注
Do you use high speed down load mode?(Y/N) = <u>N</u>	②
Lod object file name = <u>TEST C S</u>	③

- ① “Y”を入力して、セットアップ・ファイルを作ります。
- ② 高速ダウン・ロードの選択をします。ここでは、“N”を入力し、高速ダウン・ロードを選択していません。
- ③ ロードするオブジェクト・ファイル名とオプション（オブジェクト・ファイル（.HEX）とシンボル・ファイル（.SYM）のロードを選択する）を入力します。このオプションは省略しないでください。省略すると、(b)のような自動設定ができなくなります。

注 セットアップ・ファイル（SETUP.STR）がカレント・ディレクトリにない場合は、①は省略され、自動的にセットアップ・ファイルを生成します。

(b) セットアップ・ファイルから自動的に実行するとき

Create new set up mode (Y or N) : <u>N</u>	①
Do you use high speed down load mode?(Y/N) = N	②
Lod object file name = TEST C S	③

- ① “N”を入力します。
- ② セットアップ・ファイルの内容で自動的に設定されます。
- ③ //

(c) セットアップ・ファイルから自動設定されない場合

オブジェクト・ファイル名のオプションを入力せずに、セットアップ・ファイルを作ってしまうと、オブジェクト・ファイルをロードするとき自動設定されません。

Create new set up mode (Y or N) : <u>N</u>	①
Do you use high speed down load mode? (Y/N) = N	②
Lod object file name = TEST	③
Debug condition load (Y/N)? <u>N</u>	④

- ① “N”を入力します。
- ② セットアップ・ファイルの内容で自動的に設定されます。
- ③ //
- ④ 自動設定されません。デバッグ環境ファイルの入力を指定してください。

(6) 高速ダウン・ロードの指定

ホスト・マシンのオブジェクト・ファイルからプログラムを高速ダウン・ロードする場合は、高速ダウン・ロード指定をします。

```

Do you use high speed down load mode? (Y/N) Y①
brk:0>②
    
```

- ① “Y” を入力します^注。
- ② プロンプトとしてbrk:0>が表示されブレーク・モードとなります。

注 この場合、IE-75001-Rとホスト・マシンは、セントロニクス・インタフェースで接続されている必要があります。

(7) オブジェクト・ファイルのロード

ディバグするオブジェクト・ファイル名を入力します。

```

Lod object file name=TEST C S①注
    
```

- ① ロードするオブジェクト・ファイル名とオプションを入力します。このとき、拡張子(.HEX)は入力しないでください。

注 オブジェクト・ファイル名 $\begin{bmatrix} C \\ D \\ S \end{bmatrix}$ で入力します。

- C : オブジェクト・コードのみロード
- D : ディバグ環境のみロード
- S : シンボルのみロード

省略: 上記3ファイルすべてをロード (ディバグ環境のロード選択になります)。

オブジェクト・ファイルをロードしないとき

```

Lod object file name=①
brk:0>②
    
```

- ① キーを入力します。
- ② プロンプトとしてbrk:0>が表示され、ブレーク・モードとなります。

(8) システムの起動完了

前述の一連の操作により、プロンプトの表示がbrk:n>（ブレイク・モード）となり、システムの起動が完了します。この時点から、IE-75000-Rコントロール・プログラムの豊富なコマンドを使用し、目的とするプログラムのデバッグを行うことができます。

5.2.2 プログラムのディバグ手順

システムの起動が完了しますと、IE-75000-Rコントロール・プログラムの各種コマンドを使用してプログラムのディバグを行うことができます。以下はディバグ手順の概要です。

基本的なコマンドの使用方法については、第6章 基本機能の使い方において説明します。

(1) IEのシステム環境設定

IE-75001-Rでディバグを行うために、次の操作によりシステム環境を設定します。

- 対象デバイスをリセット
- クロックの選定
- ディバグ環境ファイルのロード（プログラム／シンボル・ファイルと一括ロードが可能）

(2) プログラムのロード

プログラムのディバグを行うためには、ディバグ対象となるプログラムをIE-75001-Rにロードしなければなりません。この方法には、次の二通りがあります。

- ホスト・マシンのオブジェクト・ファイルからのロード
- PGシリーズ（CH2に接続）からのロード

(3) プログラムのディバグ

豊富に用意されたコマンドを使用し、プログラムのバグ調査とその修正を行います。

(4) プログラムのセーブ

プログラムのディバグを完了させ、これを対象媒体にセーブします。セーブはロードと同様、次の二通りの方法があります。

- ホスト・マシンのオブジェクト・ファイルにセーブ
- PGシリーズ（CH2に接続）にセーブ

5.2.3 システムの終結手順

(1) IE-75000-Rコントロール・プログラムの終結

すべてのディバグ処理を完了したのちEXTコマンドで、IE-75000-Rコントロール・プログラムの終結をし、制御をOSに戻します。

(2) 電源の切断

電源の切断は、投入時と逆の順序で行います。末端の装置から主電源へ行うのが原則です。

- ① フロッピー・ディスクなどの媒体を、すべて取り外します。
- ② ターゲット・システムの電源を切断します。
- ③ プリンタが接続されている場合は、プリンタの電源を切断します。
- ④ PGシリーズが接続されている場合は、PGシリーズの電源を切断します。
- ⑤ IE-75001-Rの電源を切断します。
- ⑥ ホスト・マシンの電源を切断します。

第6章 基本機能の使い方

この章ではIE-75001-Rが持つ、主要なデバッグ機能とその使い方について、基本的な事項を説明します。

はじめてIE-75001-Rを扱われる方は、**6.1**–**6.2**節で基本的な機能を理解し、**6.3**節デバッグの基本手順にそって操作をしてください。なお、**6.4**節にはIE-75001-RにIE-75000-R-EMを接続した場合の基本機能の使用例を載せています。

6.1 システムの動作モードとコマンド入力

システム動作モードは、「ターゲット・プログラムの実行（エミュレーション）機能」および「トレース機能」が働いているか否か、システムの動作状態を示すものです。

システム動作モードには、次に示す3種類があります。またコマンドの入力は、システム動作モードによって制約を受けます。

ブレーク・モード (brk : n>)

「ターゲット・プログラムの実行（エミュレーション）機能」と「トレース機能」が、ともに停止している状態です。

エミュレーション・モード (emu : n>)

「ターゲット・プログラムの実行（エミュレーション）機能」は働いているが、「トレース機能」が停止している状態です。このモードは、ターゲット・プログラムの実行を停止してはならないときに使用されます。

トレース・モード (trc : n>)

「ターゲット・プログラムの実行（エミュレーション）機能」と「トレース機能」が、ともに働いている状態です。

(1) プロンプト表示とシステム動作モード

システムの動作モードは、モニタが画面に表示する「プロンプト」によって、識別することができます。またプロンプトが表示されていない状態では、コマンドは受け付けられません。

システム動作モード	プロンプト表示	CPU動作	トレーサ動作
ブレーク・モード	brk : n>	停止	停止
エミュレーション・モード	emu : n>	動作	停止
トレース・モード	trc : n>	動作	動作

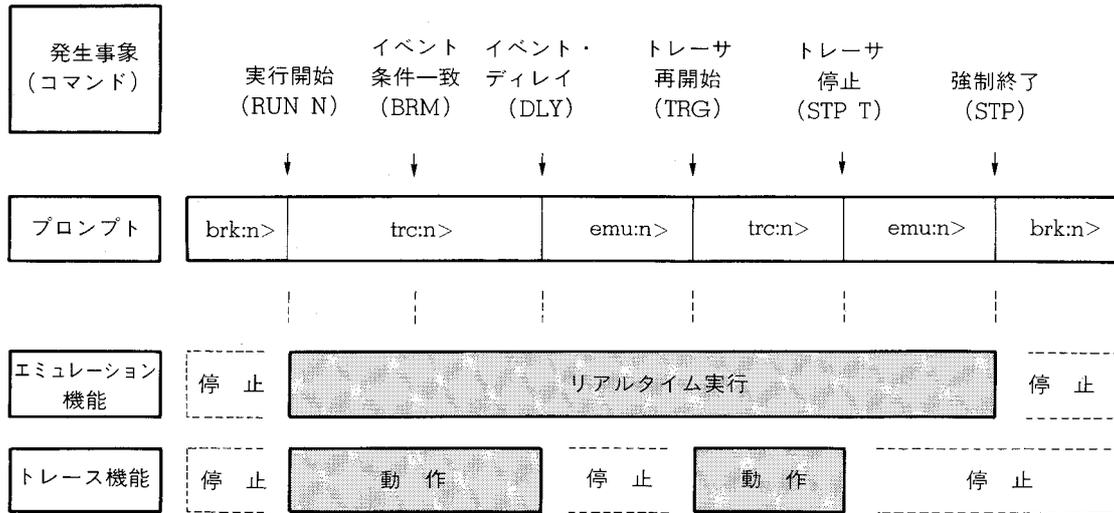
：「システム動作モード」を示します。各システム動作モードと使用可能コマンドは8.1.2を参照してください。

n : ホスト・マシンの「シリアル・ポート番号」を示します。

② プロンプト表示とシステム動作状態の相関関係

プロンプト表示と、エミュレーション・デバイスおよびトレースの機能についての相関関係の一例を示すと、次のようになります。

図 6-1 プロンプト表示とシステム動作状態の例(1)



6.2 基本機能の使い方

本節では、IE-75001-Rに用意されている豊富なデバッグ機能の中から、基本的な機能について説明します。

説明の順序は、6.3節 デバッグ基本手順の記述にそった形で行います。本節で機能の概略を理解したのちに、デバッグ基本手順を見ながら一連の操作をすると理解しやすいでしょう。

6.2.1 クロックの選択機能

クロック選択機能は、対象デバイスに対して供給するクロック・ソースを指定する機能です。クロック・ソースとして、次の2種類を選択することができます。

IE-75001-R 内部の固定クロック	(CLK Iコマンド)
ターゲット・システム側のクロック	(CLK Uコマンド)

起動時は内部の固定クロック (4.19 MHz) が選択されます。

ターゲット・システム側のクロックを選択するときは、エミュレーション・ボードにも設定が必要です。詳しくはエミュレーション・ボードのユーザズ・マニュアルを参照してください。

エミュレーション・ボードでクロックの設定を行わないで“CLK U”のコマンドを実行すると“E-CPU ERROR”のメッセージが出ます。

```
PORT ××H, ××H DATA ××
××
E-CPU ERROR !!
```

注意 クロック選択時は、RESコマンドが内部で発行されます。

6.2.2 リセット機能

リセット機能は、IE-75001-Rのシステム全体、またはエミュレーション・デバイスのリセットを行う機能です。

IE-75001-Rシステムのリセット (RES Hコマンド)

IE-75001-Rのシステム全体を、リセットします。IE-75001-R起動時の動作環境状態に戻ります。

エミュレーション・デバイスのリセット (RES コマンド)

エミュレーション・デバイスのみを、リセットします。トレース・メモリ、カバレッジ・メモリはクリアされます。

プログラムの先頭からエミュレーションを実行するときは、必ずこのRESコマンドを実行してください。

6.2.3 カバレッジ機能

カバレッジ機能は、RUNコマンドによって、ターゲット・プログラムの実行（リアルタイム・エミュレーション）を行った際に、その実行部分に対し識別子を付加する機能です。

これによってユーザは、ターゲット・プログラムのどの部分が実行済みであり、どの部分が未実行であるか、容易に判別できます。

カバレッジ機能を制御するための機能には次のものがあります。

カバレッジ測定範囲の設定機能

- カバレッジ測定範囲の設定、追加 (CVM Aコマンド)
- カバレッジ測定範囲の解除 (CVM Kコマンド)
- カバレッジ測定範囲の表示 (CVM Dコマンド)

カバレッジ測定結果の操作機能

- カバレッジ測定結果の削除 (CVD Kコマンド)
- カバレッジ測定結果の表示 (CVD Dコマンド)

6.2.4 ロード機能

ロード機能は、IE-75001-Rの各チャンネルに接続された装置から、ターゲット・プログラムをIE-75001-Rのメモリ上に展開する機能です。ホスト・マシンからは、ディバグ環境情報、シンボル・テーブルをロードすることもできます。

接 続 装 置	IE-75001-Rのチャンネル	ロードの種類
ホスト・マシン (PC-9800シリーズ, IBM PCシリーズ)	CH1 (入出力用)	標準ダウン・ロード 標準アップ・ロード
PROMプログラマ (PG-1000, PG-1500)	CH2 (入出力用)	標準ダウン・ロード 標準アップ・ロード
ホスト・マシン	CH4 (入力専用)	高速ダウン・ロード注

注 高速ダウン・ロードを行う場合は、システム起動時に高速ダウン・ロードを指定します。

詳細は**8.4.19 ロード・コマンド (LOD)**を参照してください。

(1) ホスト・マシンからのロード

ホスト・マシンに接続されたディスク装置から、LODコマンドを用いオブジェクト、ディバグ環境、シンボルの各ファイル内容を、IE-75001-R上のメモリに展開します。

通常、ホスト・マシンからのロードは、チャンネル1 (CH1) を介して、標準ダウン・ロードを行いますが、システム起動時に「高速ダウン・ロード」を指定する場合には、チャンネル4 (CH4) を介して、高速ダウン・ロードを行います。

ファイルの概要、ロード機能に用いるコマンドとその機能は次のとおりです。

表 6-1 ホスト・マシンからダウン・ロードされるファイル

ファイル	概 要
オブジェクト・ファイル (ファイル名: ×××××××.HEX)	ターゲット・プログラムのオブジェクト・コード (インテル標準HEX形式) を格納するファイルです。
シンボル・ファイル (ファイル名: ×××××××.SYM)	ターゲット・プログラムに対して、ユーザが定義したシンボルを格納するファイルです。
ディバグ環境ファイル (ファイル名: ×××××××.DBG)	ディバグ環境の情報が格納されているファイルです。 本ファイルによって、以下のコマンドを環境設定することができます。 BRA1, BRA2, BRA3, BRA4, BRK, BRM, BRS1, BRS2, CHK, CLK, DLY, MOD, OUT, PAS, PGM, REG, STS, TRF, TRM, TRX, TRY

注意1. シンボル・ファイルはなくてもかまいませんが、このときシンボルの処理はできません。アペンド・シンボルのロードは、**SYM L**コマンドを使用して行います。

2. デバッグ環境ファイルはなくてもかまいませんが、このとき環境設定はできません。

表 6-2 ロード・コマンドの種類と機能

コマンド	機 能
LOD file	オブジェクト・ディバッグ環境、シンボルの各ファイルの内容を一括ロードできます。
LOD file C	オブジェクト・ファイルのみをロードします。 備考 オブジェクト・ファイルがロードされた時点で、カバレッジ測定範囲はターゲット・プログラムのアドレス範囲が無条件で設定されます。
LOD file D	ディバッグ環境ファイルのみをロードします。
LOD file S	シンボル・ファイルのみをロードします。

② PROMプログラマからのロード

チャンネル2 (CH2) に接続された、PROMプログラマ (PGシリーズ) から、オブジェクト・コード (インテル標準HEX形式) を、IE-75001-Rのメモリ上に展開します。

PROMプログラマからロードするには、次の環境設定が必要です。

① チャンネル2モードの設定

MODコマンドを使用してシリアル・チャンネル2 (CH2) の通信動作状態を設定します。

備考 ハードウェアによる設定については、第9章を参照してください。

② PGモード (端末モード) の設定

PGMコマンドを使用してシリアル・チャンネル2 (CH2) とPROMプログラマとのインタフェースをとり、PGモード (端末モード) を確立します。この状態で、PROMプログラマからのオブジェクト・コードのアップ・ロード、ダウン・ロードが可能となります。

③ 端末モードの終了: ^Z (CTRL+Z) を入力します。

6.2.5 エミュレーション実行機能

エミュレーション実行機能は、対象デバイスによる「ターゲット・プログラムの実行（エミュレーション）」および「トレース」を開始する機能です。

エミュレーションの実行形態により、機能は次のように分類されます。

リアルタイム実行機能

- ノンブレイク・リアルタイム実行機能 (RUN Nコマンド)
- ブレイク条件付きリアルタイム実行機能 (RUN Bコマンド)

ノンリアルタイム実行機能

- ステップ実行機能 (RUN Tコマンド)

(1) リアルタイム実行機能

リアルタイム実行機能には、ターゲット・プログラムの実行停止（ブレイク）を指定しない「ノンブレイク・リアルタイム実行」と、ターゲット・プログラムの実行停止を指定する、「ブレイク条件付きリアルタイム実行」とがあります。

(a) ノンブレイク・リアルタイム実行機能 (RUN Nコマンド)

ノンブレイク・リアルタイム実行は、指定されたアドレスからターゲット・プログラムの実行およびトレースを開始し、指定されたイベントの検出によってトレースのみを停止します。

この時点で、システムの動作モードは、エミュレーション・モード (emu:n>) となります。

(i) トレースの停止条件

トレースの停止は、通常、次のレジスタに設定されている「イベント検出条件」によって行います。

- イベント・モード・レジスタ (BRMコマンド)

BRMコマンドには、イベント条件レジスタのいずれか1個以上が指定されていなければなりません。

- イベント条件レジスタ (BRA1-BRA4, BRS1-BRS2)

また、イベント検出の詳細条件は、イベント条件レジスタに設定されていなければなりません。イベント検出の条件設定方法については、**6.2.9 イベント設定, 検出機能**を参照してください。

トレースを停止するには、このほかに強制終了する方法があります。これについては(iii), (iv)項で説明します。

(ii) **トレースの再開 (TRGコマンド)**

エミュレーション・モードにおいてトレースを再開します。

(iii) **マニュアルによるトレースの停止 (STP Tコマンド)**

トレース・モード (trc:n>) では, STP Tコマンドによってトレースのみを停止することができます。

(iv) **強制終了**

ターゲット・プログラムの実行 (エミュレーション) およびトレースを停止する機能として, 強制終了 (マニュアル・ブレーク/フェール・セーフ・ブレーク) があります。

- ・ マニュアル・ブレーク (STPコマンド)
- ・ フェール・セーフ・ブレーク

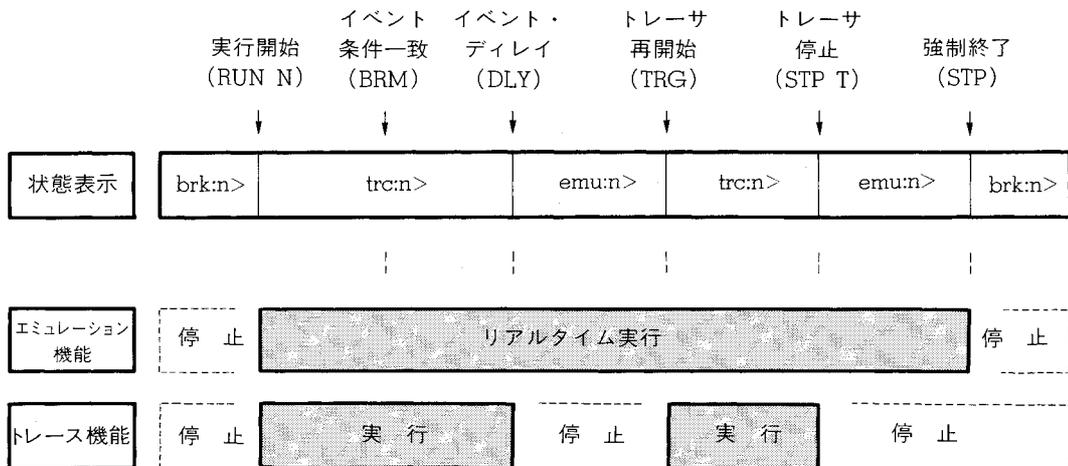
ターゲット・プログラムが, 対象デバイスに存在しない「DMEM (データ・メモリ), SPR (特殊レジスタ), レジスタ領域, スタック領域」をアクセスしたとき。

強制終了 (強制ブレーク) の詳細については, **6.2.6 ブレーク機能**を参照してください。

(v) **システム動作状態相関関係**

ノンブレーク・リアルタイム実行での, 「状態表示 (プロンプト表示)», 「CPUの動作», 「トレーサの動作」の相関関係は, 次の図のようになります。

図 6-2 プロンプト表示とシステム動作状態の例(2)



(b) ブレーク条件付きリアルタイム実行機能 (RUN Bコマンド)

ブレーク条件付きリアルタイム実行は、指定されたアドレスからターゲット・プログラムの実行を開始し、指定されたイベント条件を検出することによって、エミュレーション・デバイスとトレーサの双方を停止します。

また、イベント検出によってエミュレーション・デバイスおよびトレーサが停止すると、自動的に「1ステップ実行モード」となります。

(i) イベント検出によるブレーク

ブレークの発生は、イベント・モード・レジスタ (BRM) およびイベント条件レジスタ (BRA1-BRA4, BRS1-BRS2) に設定された条件に基づいて行います。

イベント検出の条件設定方法については、**6.2.9 イベント設定, 検出機能**を参照してください。

(ii) 強制終了によるブレーク

イベント検出によるブレークのほかに、ターゲット・プログラムの実行 (エミュレーション) およびトレースを停止する機能として、強制終了 (マニュアル・ブレーク/フェール・セーフ・ブレーク) があります。

- マニュアル・ブレーク (STP)
- フェール・セーフ・ブレーク

ターゲット・プログラムが、対象デバイスに存在しない「DMEM (データ・メモリ), SPR (特殊レジスタ), レジスタ領域, スタック領域」をアクセスしたとき。

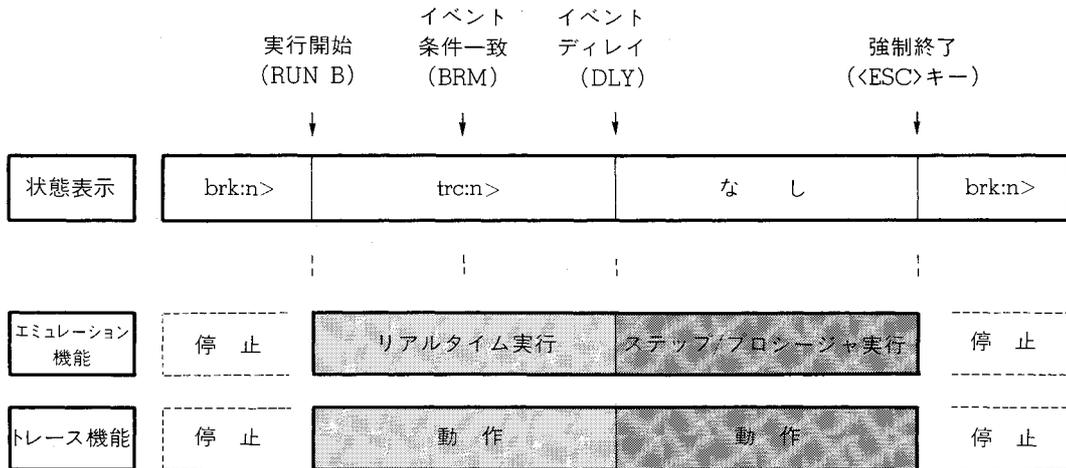
(iii) 1ステップ実行とプロシージャ実行

本機能については、ノンリアルタイム実行機能を参照してください。

(iv) システム動作状態相関関係

ブレーク条件付きリアルタイム実行での、「状態表示」、「CPUの動作」、「トレーサの動作」の相関関係は、次の図のようになります。

図 6-3 プロンプト表示とシステム動作状態の例(3)



(2) ノンリアルタイム実行機能

ノンリアルタイム実行機能には、1命令ごとにターゲット・プログラムの実行を停止し、内部レジスタの表示/検出、実行トレースを行う「ステップ実行」と、実行を開始したルーチンよりも深いネスト・レベルのルーチンに対して、内部レジスタの表示/検出、実行トレースを行わない「プロシージャ実行」の2種類があります。

(a) ステップ実行機能 (RUN Tコマンド)

ステップ実行は、ターゲット・プログラムを指定されたアドレスから1命令実行し、内部レジスタの表示や検出、および実行トレースを行います。

またステップ実行は、指定された「条件式」が成立するまで、あるいは「指定命令数」の実行が完了するまで断続的に続けられます。

ステップ実行終了後は、「1ステップ実行モード」となります。

(i) 1ステップ実行モード

1ステップ実行モードでは、<cr>キーを入力することによって、1ステップごとにターゲット・プログラムを実行することができます。

1ステップ実行中は、「トレース・データ表示指定 (TRD)」、「レジスタ表示 (REG)」の有無にかかわらず、無条件に「トレース・データ内容」、「レジスタ内容」および「逆アセンブル結果」の表示を行います。

また、1ステップ実行モードにおいては「プロシージャ実行」を行うことができます。

注意 1ステップ実行を行うと、その直前までに書き込まれたトレース・データは、トレース・メモリから消去されます。

(ii) 強制終了

ターゲット・プログラムの実行およびトレースを停止する機能として、強制終了 (マニュアル・ブレーク/フェール・セーフ・ブレーク) があります。

• マニュアル・ブレーク

<ESC> キーの入力が行われたとき。

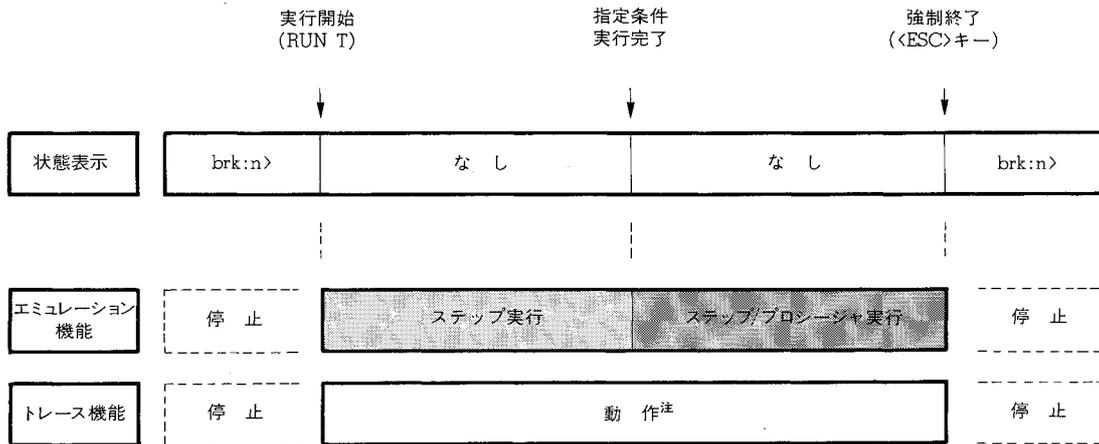
• フェール・セーフ・ブレーク

ターゲット・プログラムが、対象デバイスに存在しない「DMEM (データ・メモリ)」、SPR (特殊レジスタ)、レジスタ領域、スタック領域」をアクセスしたとき。

(iii) システム動作状態相関関係

ステップ実行での、「状態表示」、「CPUの動作」、「トレーサの動作」の相関関係は、下図のようになります。

図 6-4 プロンプト表示とシステム動作状態の例(4)



注 トレース・メモリは1ステップ実行ごとに消去されます。

(b) プロシージャ実行機能 (RUN Tコマンド)

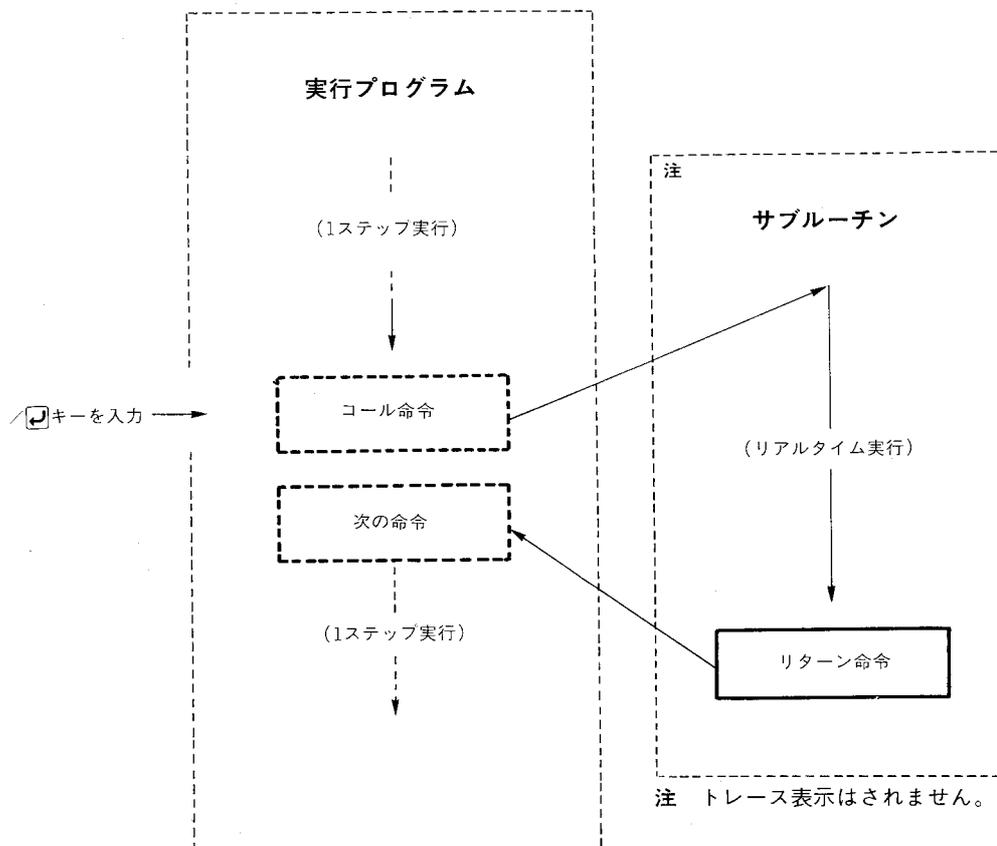
プロシージャ実行機能は、「1ステップ実行モード」においてのみ、使用できる機能です。

プロシージャ実行機能は、実行を開始したルーチンよりも深いネスト・レベルのルーチンに対して、リアルタイム実行（イベント検出やトレースを行わない）を行う機能です。

プロシージャ実行の操作は、次の手順で行います。

- ① <cr> キーを入力しながら、1ステップ実行を行っていきます。
- ② コール命令 (CALL, CALLA, CALLF, CALLTのいずれか) が表示され、/キーと <cr> キーを入力すると、プロシージャ実行に入ります。
- ③ コール命令の分岐先サブルーチンが、プロシージャ実行 (リアルタイム実行) されます。
- ④ サブルーチン内のリターン命令によって、サブルーチンから元の実行プログラムに戻ります。戻り先は、コール命令の次の命令です。
- ⑤ コール命令の次の命令から、再び1ステップ実行モードになります。

図 6-5 プロシージャ実行の概念図



6.2.6 ブレーク機能

ブレーク機能は、エミュレーション・デバイスによる「ターゲット・プログラムの実行（エミュレーション）」および「トレース」を停止する機能です。

ブレーク機能には、大別すると次に示す3種類があります。

イベント検出ブレーク

マニュアル・ブレーク

フェール・セーフ・ブレーク

これらの「ブレーク機能」と「エミュレーション実行機能」との関係を、次に示します。

○：有効 ×：無効

	イベント検出 ブレーク	マニュアル・ ブレーク	フェール・セーフ ・ブレーク
ノンブレーク・リアルタイム実行 (RUN Nコマンド)	×	○	○
ブレーク条件付きリアルタイム実行 (RUN Bコマンド)	○	○	○
ステップ実行（ノンリアルタイム） (RUN Tコマンド)	×	○	○

(1) イベント検出ブレーク

イベント検出ブレークは、指定されたイベント条件を検出することによって、ターゲット・プログラムの実行を停止する機能です。

このブレークは、ブレーク条件付きリアルタイム実行機能 (RUN Bコマンド) に対してのみ有効となります。

イベント検出の条件は、それぞれ「イベント・モード・レジスタ」および「イベント条件レジスタ」に設定されていなければなりません。

(i) イベント・モード・レジスタ (BRM)

BRMには、次に示すイベント条件レジスタ名が指定されている必要があります。

BRA1, BRA2, BRA3, BRA4, BRS1, BRS2

BRMに登録 (選定) された、イベント条件レジスタの内容がブレークの条件となります。

なお、初期設定時はBRS1になっています。

(ii) イベント条件レジスタ (BRA1-BRA4, BRS1-BRS2)

イベント条件レジスタには、イベント検出を行うための詳細条件が設定されている必要があります。

BRA1-BRA4 : データ・メモリ・アクセスに関する事柄を、イベント検出の条件として設定します。

BRS1-BRS2 : プログラム・メモリ・アクセスに関する事柄を、イベント検出の条件として設定します。

イベント検出の条件設定方法については、**6.2.9 イベント設定, 検出機能**を参照してください。

(2) マニュアル・ブレーク

マニュアル・ブレークは、ユーザがリアルタイム・エミュレーションの実行およびトレースの実行を、強制的に停止（強制ブレーク）させるための機能です。強制ブレーク機能は、すべての実行機能に対して有効となります。

マニュアル・ブレークには、次の2種類があります。

強制ブレーク (STP)

STPコマンドが入力されると、エミュレーション・デバイスおよびトレーサの動作を停止します。STPコマンドによる強制ブレークは、リアルタイム実行機能においてのみ有効となります。

強制ブレーク (RES)

RESコマンドが入力されると、エミュレーション・デバイスおよびトレーサの動作を停止します。

注意 エミュレーション・デバイスは、ターゲット・プログラムの実行を停止すると同時にリセットされます。

(3) フェール・セーフ・ブレーク

フェール・セーフ・ブレークは、エミュレーション・デバイスが異常動作をした場合に、IE-75001-Rが、これを強制的に停止させる機能です。この強制ブレーク機能は、すべての実行機能に対して有効となります。

フェール・セーフ・ブレークは、次の1種類です。

不正アクセス・ブレーク

ターゲット・プログラムが、対象デバイスに存在しないDMEM（データ・メモリ）、SPR（特殊レジスタ）、レジスタ領域、スタック領域を、アクセスした場合に強制的に停止します。

6.2.7 トレース機能

トレース機能は、ターゲット・プログラムのトレース（追跡）を行った際、プログラム・メモリ、データ・メモリ、I/Oポート、外部センス・クリップなどの状態に関する情報を、IE-75001-Rが持つ「トレース・メモリ」に書き込んでいく機能です。

トレース・メモリに書き込まれたデータは、「TRDコマンド」によって画面上に表示して見ることができます。

トレースの実行およびトレース表示に関する主な機能には、次のものがあります。

トレースの動作

- ノンブレイク・リアルタイム実行時の動作 (RUN Nコマンド)
- ブレイク条件付きリアルタイム実行時の動作 (RUN Bコマンド)
- ステップ実行時の動作 (RUN Tコマンド)

トレース条件の設定機能

- クオリファイ・トレース条件の指定 (TRXコマンド)
- セクション・トレース条件の指定 (TRYコマンド)
- トレース・モードの指定 (TRMコマンド)

トレース・データの表示, ポインタ操作機能

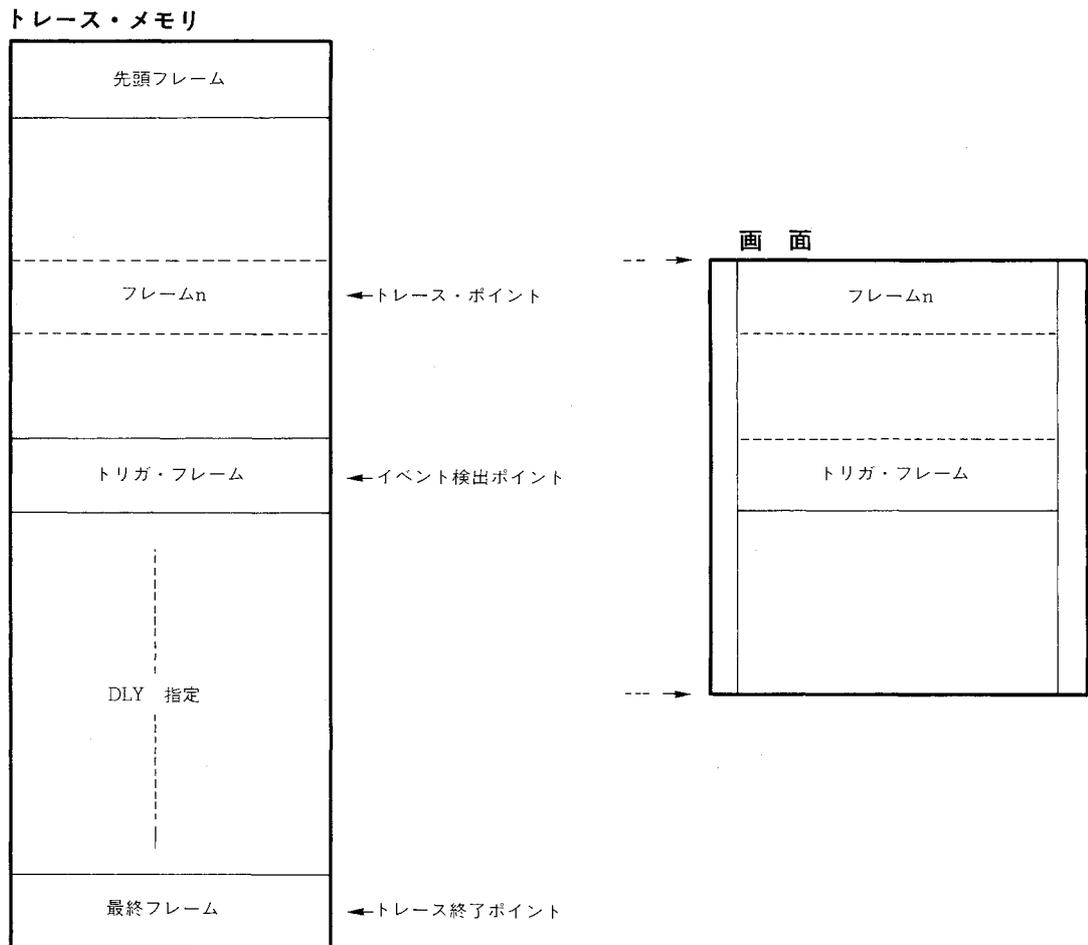
- トレース・データ検索条件の設定 (TRFコマンド)
- トレース・データの表示 (TRDコマンド)
- トレース・ポインタの操作 (TRPコマンド)

(1) トレースの実行とトレース・メモリの関係

トレースは、先頭フレームから最終フレームに向かって、トレース情報を記録していく機能です。最終フレームまで記録すると、再び先頭フレームの位置から記録を続けます。

RUN Bコマンド、**DLY M**コマンドを指定したときを例にとったトレースの概念図を次に示します。

図 6-6 トレースの概念図



備考 DLY Mコマンドは、トリガ・フレームをトレース・メモリの中央へ設定するコマンドです。

② トレースの動作

トレーサは、実行形態によって異なった動作をします。

(a) ノンブレイク・リアルタイム実行時の動作

トレーサは、RUN Nコマンドの入力でトレースを開始し、BRMコマンドで指定されたイベント条件が成立すると、さらにDLYコマンドで指定されたディレイ・ポイントまでトレースを行って、トレース動作を終了します。

(b) ブレイク条件付きリアルタイム実行時の動作

トレーサは、RUN Bコマンドの入力でトレースを開始し、BRMコマンドで指定されたイベント条件が成立すると、さらにDLYコマンドで指定されたディレイ・ポイントまでトレースを行って、トレース動作を中断します。

このとき、ターゲット・プログラムの実行（エミュレーション・デバイスの実行）も同時に停止し、1ステップ実行モードになります。

1ステップ実行モードでの操作とトレース実行は、次のようになります。

入力：次の1命令を実行し、その命令に関するトレース・データを画面に表示します。

注意 1ステップ実行を行うと、その直前までに書き込まれたトレース・データは、トレース・メモリから消去されます。

／ 入力：プロシージャ実行します。

ESC 入力：1ステップ実行モードを終了します。

(c) ステップ実行時の動作

RUN Tコマンドでは1ステップ実行ごとにトレーサが動きますが、1ステップ分のトレース情報しか記録しません。また、1ステップ実行ごとにトレース・データはトレース・メモリから消去されます。

③ トレース条件の設定機能

次のコマンドによって、トレースを行うときの条件を指定することができます。これらの指定がないときには、「無条件トレース」すなわちターゲット・プログラムの1命令ごとに、トレース情報が記録されます。

(a) クオリファイ・トレースの指定 (TRXコマンド)

指定されたプログラム・アドレスで、あるいは指定されたデータ・メモリをアクセスしたときに、トレースを行う指定です。指定する条件は、次のイベント条件レジスタに設定しておく必要があります。

BRA1-BRA4 : データ・メモリのアクセス条件

BRS1-BRS2 : プログラム・アドレス

イベント条件レジスタへの設定方法は、**8.4.3 データ・メモリ・イベント条件設定コマンド (BRA)**、**8.4.6 プログラム・メモリ・イベント条件設定コマンド (BRS)**を参照してください。

(b) セクション・トレースの指定 (TRYコマンド)

指定されたイネーブル条件でトレースを開始し、指定されたディスエーブル条件でトレースを停止する指定です。即ち範囲指定トレースの指定です。

TRY E [BR?][BR?][BR?][BR?][BR?][BR?] : イネーブル条件設定

TRY D [BR?][BR?][BR?][BR?][BR?][BR?] : ディスエーブル条件設定

イネーブルおよびディスエーブル条件は、イベント条件レジスタに設定されている必要があります。

BR? : BRA1-BRA4 : データ・メモリのアクセス条件

BRS1-BRS2 : プログラム・アドレス

(c) トレース・モードの指定 (TRMコマンド)

クオリファイ・トレース、セクション・トレース、無条件トレースのどのトレースを行うか指定します。本コマンドで指定されたトレース・モードが、最終的に有効となります。

TRM TRX : クオリファイ・トレース

TRM TRY : セクション・トレース

TRM ALL : 無条件トレース

(4) トレース・データの表示, ポインタ操作機能

次のコマンドによって, トレース・データの表示, および表示を行う際の検索条件を, 指定することができます。

(a) トレース・データ検索条件の設定 (TRFコマンド)

トレース・データの検索条件を指定することができます。トレース・データ表示コマンドで, 次が指定されたとき有効となります。

TRD[ALL] \$Fコマンド

TRD[ALL] \$Qコマンド

検索条件は, 次表の全項目あるいは任意の項目を選択して, 指定することができます。

表 6-3 トレース項目と指定値範囲

指定項目	内 容	指定値範囲
PA	プログラム・メモリ・アドレス	0-FFFFH
PD	プログラム・メモリ・データ PAで示された内容 (命令コード)	0-FFH
MA	データ・メモリ・アドレス	0-FFFFH
MD	データ・メモリ・データ MAで示された内容 (データ)	0-FFH
MRW	データ・メモリ・アクセス・ステータス MRD : リード MWR : ライト MRW : リード・モディファイ・ライト NC : すべてのリード・ライト	同 左
Pn	I/Oポート・データ	0-FH
EXT	外部センス・クリップ・データ	0-FFH

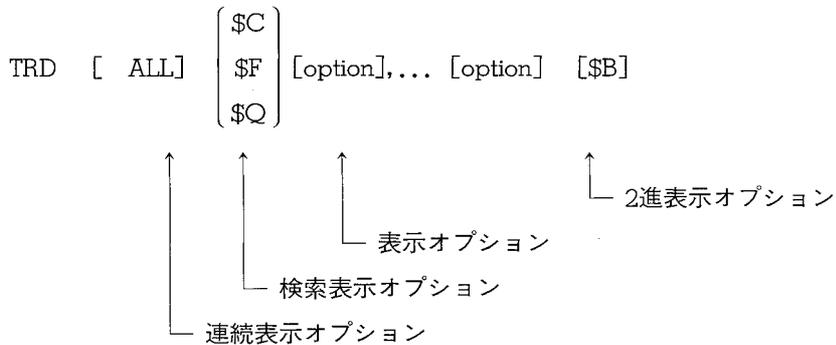
(b) トレース・データの表示 (TRDコマンド)

トレース・メモリに書き込まれているトレース・データを、画面に表示します。表示する項目は、次のとおりです。

表 6-4 トレース・データ表示の説明

指定項目	内 容	値, 範囲
Frame	フレームNo. トレース・メモリのフレーム番号	0-7FFH
PA	プログラム・メモリ・アドレス	0-FFFFH
PD	プログラム・メモリ・データ PAで示された内容 (命令コード)	最大3バイト
MA	データ・メモリ・アドレス	0-FFFH
MD	データ・メモリ・データ MAで示された内容 (データ)	0-FH(4ビット) 0-FFH(8ビット)
MRW	データ・メモリ・アクセス・ステータス MRD : リード MWR : ライト MRW : リード・モディファイ・ライト	—
Pn	I/Oポート・データ	0-FH
EXT	外部センス・クリップ・データ	0-FFH
Label	レーベル PAをシンボル変換したレーベル	8文字
Mnemonic	ニモニック・コード PD (命令コード) を, 逆アセンブルした コード	24文字

トレース・データ表示では、オプションを指定することによって、表示の方法や表示の内容を選択することができます。



(i) 全データ表示指定 (TRD ALLコマンド)

全トレース・データを表示して、コマンドを終了します。

(ii) ページ単位表示指定 (TRDコマンド)

カレント・フレームから、画面1ページ分 (12行) を表示して、メニュー・モードになります。この状態で、次のコマンドを使用して表示を行うことができます。

表 6-5 トレース・データの範囲指定

コマンド	機能概要
L	トレース中止フレームを基準にして、表示をします。
F	トレース先頭フレームを基準にして、表示をします。
+, <cr>	カレント・ページの次のページを表示します。
-	カレント・ページの前のページを表示します。
T	トリガ・フレーム (イベント条件検出のフレーム) を基準にして、表示をします。
.	TRDコマンドを終了します

注 「T」, 「.」のコマンド以外は、「検索表示オプション」の指定 (\$C, \$Q, \$F) によって機能が修飾されます。詳細は、第8章 コマンド説明を参照してください。

(iii) 検索表示条件指定

TRDコマンドで、「検索表示オプション (\$Q, \$F)」が指定されたときには、TRFコマンドで指定された、検索条件が有効となります。

(c) **トレース・ポイントの操作 (TRP コマンド)**

トレース・ポイントの操作機能は、トレース・ポイントを指定された位置に位置付けます。位置付けの方法は、次の5種類があります。

移動数の指定 (TRP word コマンド)

トレース・ポイントを、指定された数だけ移動します。指定の範囲は、±1から±7FF(16進)まで可能です。

先頭フレームの指定 (TRP F コマンド)

トレース・ポイントを、先頭フレーム (最も古いフレーム) に位置付けます。

最終フレームの指定 (TRP L コマンド)

トレース・ポイントを、最終フレーム (最も新しいフレーム) に位置付けます。

トリガ・フレームの指定 (TRP T コマンド)

トレース・ポイントを、トリガ・フレームに位置付けます。トリガ・フレームが存在しない場合、本コマンドは意味を持ちません。なおエミュレーション実行後のトレース・ポイントは、トリガ・フレームを指しています。ただし、トリガ・フレームが存在しない場合は先頭フレーム (もっとも古いフレーム) を指しています。

トレース・フレームの絶対番地指定

トレース・ポイントを指定された絶対番地に移動します。指定範囲は現在トレースされているトレース範囲のみです。

6.2.8 チェック機能

チェック機能とは、リアルタイム実行中に指定されたチェック・ポイントを検出すると、リアルタイム実行を一時中断し、指定された情報をチェック・データとしてトレース・メモリに出力する機能です。トレース・メモリへの情報出力を完了すると、リアルタイム実行は再開されます。

汎用レジスタ内容のチェック・データ出力 (CHK BR? …BR? REGコマンド)

特殊レジスタ内容のチェック・データ出力 (CHK BR?…BR? SPRコマンド)

データ・メモリ内容のチェック・データ出力 (CHK BR?…BR? addrコマンド)

(1) 汎用レジスタ内容のチェック・データ出力 (CHK BR?…BR? REGコマンド)

指定されたイベント条件レジスタの内容 (チェック・ポイント) を検出すると、汎用レジスタの内容をチェック、データとしてトレース・メモリに書き込みます。

(a) チェック・ポイントの選択

次のイベント条件レジスタを、チェック・ポイントとして指定します。

BRA1, BRA2, BRA3, BRA4	: データ・メモリ・アクセス・ポイント
BRS1, BRS2	: プログラム・メモリ・アクセス・ポイント
OFF	: チェック機能を停止

注意 チェック・ポイントの設定は、BRA1-BRA4, BRS1-BRS2のコマンドを使用して設定します。

(b) チェック・データの指定

レジスタ代表名 (REG) を指定します。次のレジスタ内容がチェック・データとして出力されます。

XA, HL, DE, BC, XA', HL', DE', BC'	: ペア・レジスタ
SP (含むSBS), PC, RBS, MBS	: コントロール・レジスタ
CY, RBE, MBE, ISTO, IST1	: フラグ

(2) 特殊レジスタ内容のチェック・データ出力 (**CHK BR? ...BR? SPR...SPR**コマンド)

指定されたイベント条件レジスタの内容 (チェック・ポイント) を検出すると、指定された特殊レジスタの内容をチェック・データとしてトレース・メモリに書き込みます。

(a) チェック・ポイントの選択

(1)項に同じ

(b) チェック・データの指定

最高5個までの、特殊レジスタを指定します。特殊レジスタは、デバイスによって異なります。

(3) データ・メモリ内容のチェック・データ出力 (**CHK BR? ...BR? addr...addr**)

指定されたイベント条件レジスタの内容 (チェック・ポイント) を検出すると、指定されたデータ・メモリ・アドレスの内容をチェック・データとしてトレース・メモリに書き込みます。

(a) チェック・ポイントの選択

(1)項に同じ

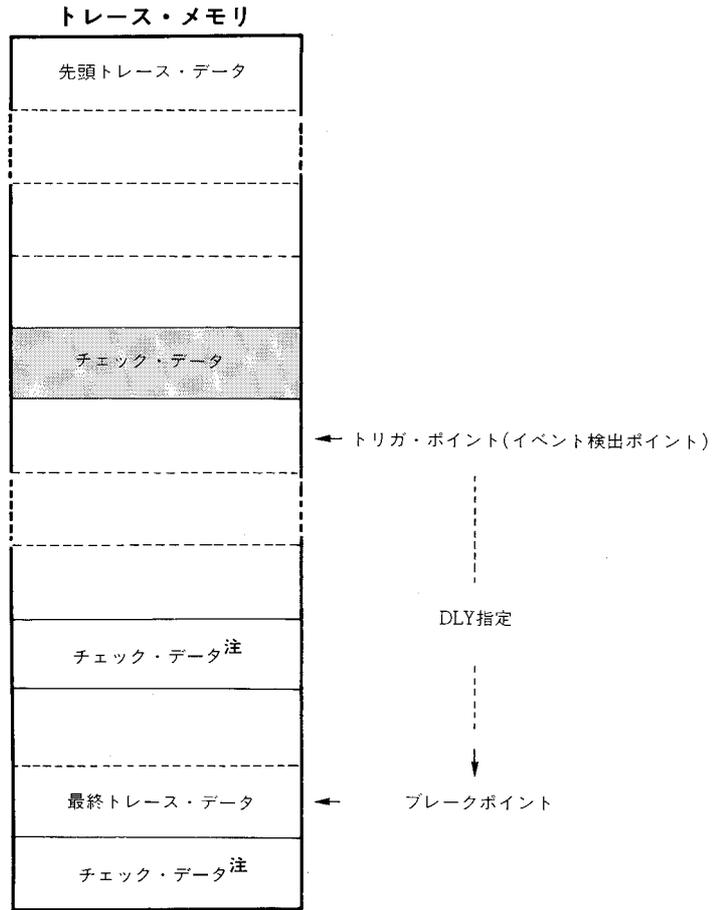
(b) チェック・データの指定

データ・メモリ・アドレスを最高5ポイントまで指定します。

(4) チェック・データの出カタイミング

DLY中(イベント検出からブレークまでの部分)、チェック・データは出力されません(図6-7参照)。

図6-7 チェック・データを含んだ場合のトレース・メモリ



注 この部分のチェック・データは出力されません。

6.2.9 イベント設定, 検出機能

イベント設定, 検出機能は, エミュレーション・デバイスによる「ターゲット・プログラムの実行」やトレーサによる「トレース動作」を停止するための「条件を設定」したり, 設定された条件に従って「イベントを検出」する機能です。

イベント条件設定, 検出機能には, 次に示す3種類があります。

イベント検出条件の設定機能

データ・メモリ・イベント条件の設定 (BRA1-BRA4コマンド)

プログラム・メモリ・イベント条件の設定 (BRS1-2コマンド)

イベント検出条件の統合機能

イベント条件の統合 (BRMコマンド)

イベント検出のディレイ機能

イベント検出点の設定 (DLYコマンド)

(1) イベント検出条件の設定機能

イベント検出条件の設定機能は, 「ターゲット・プログラムの実行」や「トレース」を停止するための条件を, 「イベント条件レジスタに設定」する機能です。イベント検出条件の設定機能で設定されたイベント検出条件は, イベント検出条件の統合機能によって, イベント・モード・レジスタに設定されないと, 最終的に有効となりません。詳細は, (2) イベント検出条件の統合機能(BRMコマンド)を参照してください。

イベント検出条件を設定する機能には, 次の2種類があります。

(a) データ・メモリ・イベント条件設定機能 (BRA1-BRA4コマンド)

ターゲット・プログラムが, 指定されたデータ・メモリをアクセスしたとき, または外部センス・クリップに対してデータの入力があったとき, これをイベント検出条件として, データ・メモリ・イベント条件レジスタに, 設定することができます。

(i) データ・メモリ・イベント条件レジスタ

データ・メモリ・イベント条件レジスタは, 次の4個があります。BRA1-BRA4コマンドを使用して, 最高4個の条件を設定することができます。

(ii) イベント条件

イベント検出条件には、次のステータスを設定できます (8.4.3参照)。

MA : データ・メモリ・アドレス
V : データ・バリュー
C : アクセス・ステータス
E : 外部センス・クリップ・データ

(b) プログラム・メモリ・イベント条件設定機能 (BRS1-BRS2コマンド)

ターゲット・プログラムが、指定されたプログラム・メモリ・アドレスの命令を実行したとき(エミュレーション・デバイスによってアクセスされたとき)、およびそのときに外部センス・クリップに対してデータの入力があったとき、これをイベント検出の条件としてプログラム・メモリ・イベント条件レジスタに設定することができます。

(i) プログラム・メモリ・イベント条件レジスタ

プログラム・メモリ・イベント条件レジスタは、BRS1, BRS2の2つがあります。BRS1, BRS2コマンドを使用して、最大2個の条件を設定することができます。

(ii) イベント条件

イベント検出条件には、次の項目を設定できます。

PA : プログラム・メモリ・アドレス
E : 外部センス・クリップ・データ

(2) イベント検出条件の統合機能 (BRMコマンド)

イベント検出条件の統合機能は、イベント検出条件の設定機能で設定されたイベント検出条件を、イベント・モード・レジスタに統合し、イベント検出を最終的に有効とする機能です。

初期設定時は、BRS1になっています。

(i) イベント・モード・レジスタ

イベント・モード・レジスタは、BRM1個のみ存在します。

(ii) イベント条件レジスタ

イベント・モード・レジスタに登録できるイベント条件レジスタは、次の6個です。

データ・メモリ・イベント条件レジスタ : BRA1, BRA2, BRA3, BRA4

プログラム・メモリ・イベント条件レジスタ : BRS1, BRS2

③ イベント検出点の設定機能

イベント検出点の設定機能は、イベント・モード設定コマンド（BRMコマンド）によって最終的に決定されたイベント検出点に対して、実行動作のディレイ（遅延調整）を設定する機能です。

イベント検出点（トリガ・ポイント）を、トレース・メモリのどこに置くか、次の3種類を指定します。

なお、初期設定時は“L”に指定されています。

(i) **F (First)** 指定

イベント検出点を、トレース・メモリの、最初（先頭フレーム）に置きます。

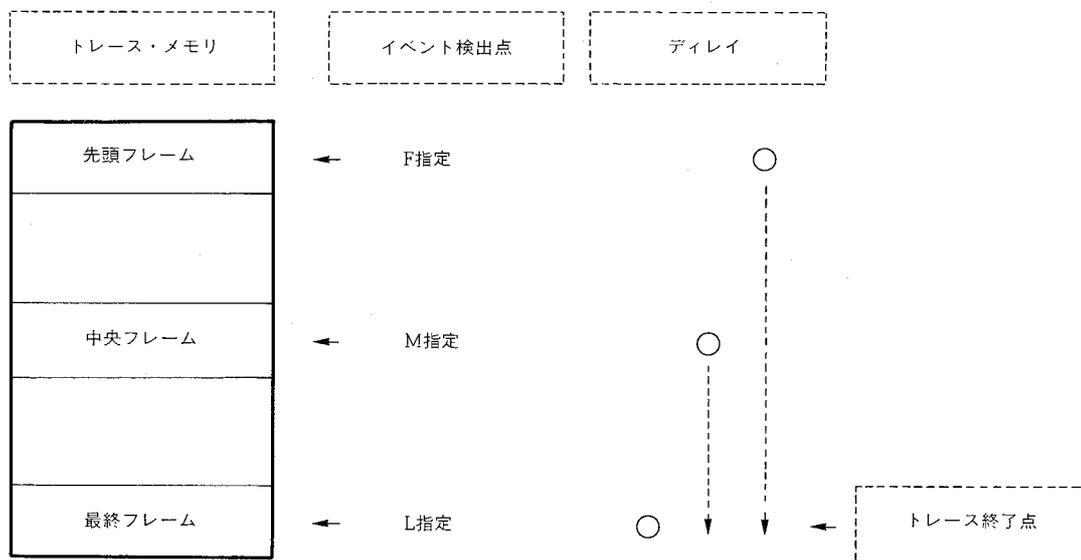
(ii) **M (Middle)** 指定

イベント検出点を、トレース・メモリの、中央に置きます。

(iii) **L (Last)** 指定

イベント検出点を、トレース・メモリの、最後に置きます。

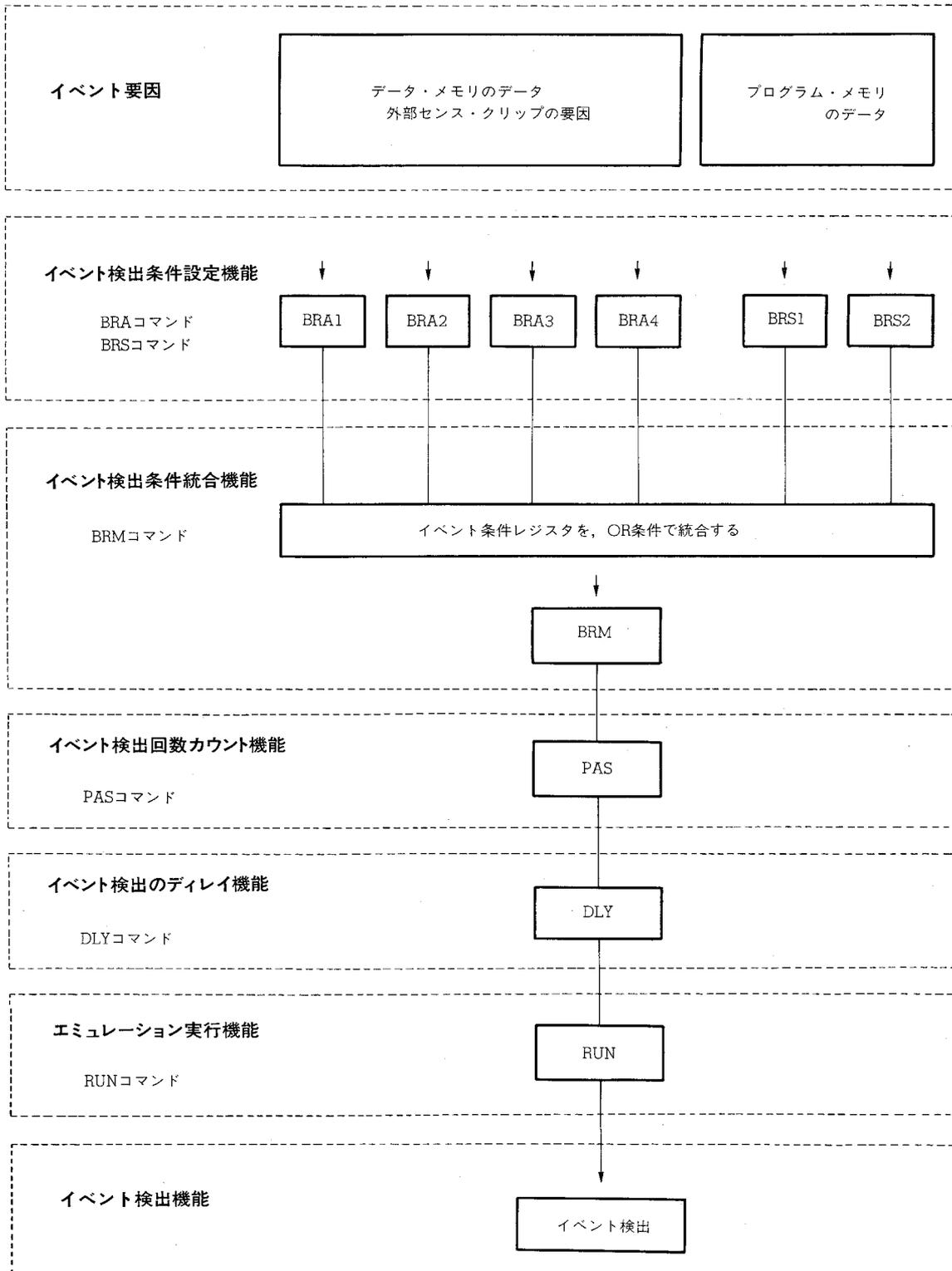
図 6-8 イベント検出点の設定



(4) イベント検出の概念

イベント条件の設定からイベント検出までの概念を、次に示します。

図 6-9 イベント検出の概念図



6.2.10 レジスタ操作機能

レジスタ操作機能は、汎用レジスタおよび特殊レジスタの内容表示および内容変更を行う機能です。主な機能を次に示します。

汎用レジスタの操作機能

- レジスタ内容の表示 (REG Dコマンド)
- レジスタ内容の変更 (REG Cコマンド)

特殊レジスタの操作機能

- レジスタ内容の表示 (SPR Dコマンド)
- レジスタ内容の変更 (SPR Cコマンド)

(1) 汎用レジスタ操作機能

制御レジスタおよび汎用レジスタについて、内容の表示および変更を行う機能です。

制御レジスタ PC, SP, PSW, RBS, MBS

汎用レジスタ XA, HL, DE, BC, XA', HL', DE', BC', X, A, H, L, D, E, B, C

PSWについては、次に示すPSWフラグ名で表示や変更を行います。レジスタは対象デバイスにより異なります。

PSWフラグ名 CY, RBE, MBE, IST1, IST0

(a) レジスタ内容の表示 (REG Dコマンド)

カレント・レジスタ・バンクあるいは、対象デバイスの全レジスタ・バンクのレジスタ内容を表示します。

カレント・レジスタ・バンク：RBS (レジスタ・バンク・セクタ) によって指定されたレジスタ・バンク

(b) レジスタ内容の変更 (REG Cコマンド)

カレント・レジスタ・バンクのレジスタ内容を変更します。

(2) 特殊レジスタの操作機能

対象デバイスのI/O領域(以降特殊レジスタと呼ぶ)の内容表示および内容変更を行う機能です。レジスタ操作は、次に示すグループ名、レジスタ名によって行うことができます。

グループ名：75Xシリーズにおける特殊レジスタは、データ・メモリの0F80H-0FFFH番地に割り当てられています。

SPRコマンドでは、特殊レジスタが割り当てられた、アドレス上位8ビットをグループ名として指定することができます。

グループ名は、対象デバイスにより異なります。

レジスタ名：特殊レジスタ名は、対象デバイスにより異なります。

(a) レジスタ内容の表示 (SPR Dコマンド)

指定レジスタ、グループ内の全レジスタ、全特殊レジスタの内容を表示します。

(b) レジスタ内容の変更 (SPR Cコマンド)

指定レジスタ、グループ内の全レジスタ、全特殊レジスタ内容を変更します。

6.2.11 メモリ操作機能

メモリ操作機能は、ニモニック・コードあるいは16進コードを使用して、データ・メモリやプログラム・メモリの内容を、表示したり変更する機能です。

ニモニック・コードによるメモリ操作機能

- プログラム・メモリ内容のニモニック・コード表示 (DASコマンド)
- ニモニック・コードによるプログラム・メモリ内容の変更 (ASMコマンド)

16進コードによるメモリ操作機能

- プログラム・メモリの操作 (MEMコマンド)
- データ・メモリの操作 (RAMコマンド)

(1) ニモニック・コードによるメモリ操作機能

プログラム・メモリ上にあるターゲット・プログラム (オブジェクト・コード) をニモニック・コードで表示または変更する機能です。

(a) プログラム・メモリ内容のニモニック・コード表示 (DASコマンド)

プログラム・メモリ上のオブジェクト・コード (16進コード) をニモニック・コードで表示します。

(b) ニモニック・コードによるプログラム・メモリ内容の変更 (ASMコマンド)

ニモニック・コードを入力することによって、プログラム・メモリ上のターゲット・プログラムを変更 (修正, 追加) することができます。

② 16進コードによるメモリ操作機能

16進コードを使用して、プログラム・メモリやデータ・メモリの内容表示、あるいは内容変更などを行う機能です。

(a) プログラム・メモリの操作

プログラム・メモリの操作機能には、次のものがあります。

メモリ内容の変更	(MEM Cコマンド)
メモリ内容の表示	(MEM Dコマンド)
メモリ内容のテスト	(MEM Eコマンド)
メモリ内容のイニシャライズ	(MEM Fコマンド)
メモリ内容のサーチ	(MEM Gコマンド)
メモリ内容のコピー	(MEM Mコマンド)
メモリ内容の交換	(MEM Xコマンド)
メモリ内容の比較	(MEM Vコマンド)

(b) データ・メモリの操作

データ・メモリ操作機能には、次のものがあります。

メモリ内容の変更	(RAM Cコマンド)
メモリ内容の表示	(RAM Dコマンド)
メモリ内容のテスト	(RAM Eコマンド)
メモリ内容のイニシャライズ	(RAM Fコマンド)
メモリ内容のサーチ	(RAM Gコマンド)
メモリ内容のコピー	(RAM Mコマンド)
メモリ内容の交換	(RAM Xコマンド)
メモリ内容の比較	(RAM Vコマンド)
ファイルからのデータ・メモリへのロード	(RAM Lコマンド)
データ・メモリからファイルへのセーブ	(RAM Sコマンド)

6.2.12 セーブ機能

セーブ機能は、IE-75001-R上に展開されているターゲット・プログラムを各チャンネルに接続された装置に格納する機能です。

ホスト・マシンに対しては、デバッグ環境情報をセーブすることもできます。

接続装置には次の種類があります。

接続装置	IE-75001-Rのチャンネル	用途
ホスト・マシン (PC-9800シリーズ, IBM PCシリーズ)	CH1 (入出力用)	標準ダウン・ロード用 標準アップ・ロード用
PROMプログラマ (PG-1000, 1500)	CH2 (入出力用)	標準ダウン・ロード用 標準アップ・ロード用

(1) ホスト・マシンへのセーブ

IE-75001-R上のオブジェクト・コード、デバッグ環境情報を、ホスト・マシンに接続されたディスク装置のファイルに格納します。

ファイルの概要、セーブ機能に用いるコマンドとその機能は次表のとおりです。

表 6-6 ホスト・マシンにセーブできるファイル

ファイル	概要
オブジェクト・ファイル (ファイル名: ××××××××.HEX)	ターゲット・プログラムのオブジェクト・コード (インテル標準HEX形式) が格納されているファイルです。
シンボル・ファイル (ファイル名: ××××××××.SYM)	ターゲット・プログラムに対して、ユーザが定義したシンボルを格納するファイルです。
デバッグ環境ファイル (ファイル名: ××××××××.DBG)	デバッグ環境の情報が格納されているファイルです。 本ファイルによって、以下のコマンドを環境設定することができます。 BRA1, BRA2, BRA3, BRA4, BRK, BRM, BRS1, BRS2, CHK, CLK, DLY, MOD, OUT, PAS, PGM, REG, STS, TRF, TRM, TRX, TRY

注意1. シンボル・ファイルはなくてもかまいませんが、このときシンボルの処理はできません。

アペンド・シンボルのロードは、**SYM L**コマンドを使用して行います。

2. デバッグ環境ファイルはなくてもかまいませんが、このとき環境設定はできません。

表 6-7 セーブ・コマンドの種類と機能

ファイル	機 能
SAV file	オブジェクト・コード、デバッグ環境情報をオブジェクト・ファイル、デバッグ環境ファイルにセーブします。
SAV file C	オブジェクト・コードのみを、オブジェクト・ファイルにセーブします。
SAV file D	デバッグ環境情報のみを、デバッグ環境ファイルにセーブします。

② PROMプログラマへのセーブ

IE-75001-Rのメモリ上のオブジェクト・コードを、チャンネル2 (CH2) に接続されているPROMプログラマ (PGシリーズ) へ格納します。

6.2.13 システム終了機能

システム終了機能は、IE-75001-Rのシステムを終結し、OSに制御を戻す機能です。IE-75001-Rのシステムを終結する場合には、必ずEXTコマンドを使用してください。

6.2.14 その他の機能

前項までに説明した以外の機能について、用途別に概要を示します。詳細は第8章 コマンド説明を参照してください。

(1) シンボル操作機能

75Xシリーズで使用される各種シンボルの追加、削除、表示などの、シンボル操作を行う機能です。シンボル操作機能には、次のものがあります。

アペンド・シンボルの定義	(SYM Aコマンド)
アペンド・シンボル値の変更	(SYM Cコマンド)
全シンボルの表示 (アペンド、パブリック、ローカルの各シンボル)	(SYM Dコマンド)
アペンド・シンボルの削除	(SYM Eコマンド)
全シンボルの削除	(SYM Kコマンド)
アペンド・シンボルのロード	(SYM Lコマンド)
アペンド・シンボルのセーブ	(SYM Sコマンド)
カレント・モジュール名の指定	(SYM Mコマンド)

(2) コマンド・ファイル関係の操作機能

コマンド・ファイルのオープン、クローズ、入出力など、コマンド・ファイルに関する操作を行う機能です。

(a) 入力装置のリダイレクト (STR d:fileコマンド)

ホスト・マシンに接続されているディスク装置上のファイルから、コマンドおよびデータを入力する機能です。

入力対象ファイル

- コマンド・ファイル作成コマンド (COM) で作成されたファイル
- エディタで作成された、コマンドおよびデータ

入力開始、終了方法

- 入力の一時的停止 : ^L (CTRL+L)
- 一時的停止の解除 : ^L (コマンド入力待ち状態に限る)
- 入力の終了 : ^K (CTRL+K)

(b) 出力装置のリダイレクト (LSTコマンド)

コマンドの実行結果を指定された装置に出力するために、装置をオープンしたりクローズしたりする機能です。次の4種類の機能があります。

ファイル出力装置とファイルのオープン (LST d:fileコマンド)

リスト出力装置のオープン (LST LST:コマンド)

ファイルまたはリスト出力装置のクローズ (LST CON:コマンド)

出力の開始と停止方法

- 出力開始: ^P (CTRL+P)
 - 出力停止: ^P
- } 繰り返し押すと、交互に状態が変わる。

(c) コマンド・ファイル作成機能 (COMコマンド)

COMコマンド入力時点以降のコマンドおよびデータを、ファイルあるいはリスト出力装置に出力するためにオープンしたり、オープンされた装置をクローズしたりする機能です。

次に示す4種類の機能があります。

コマンド・ファイルのオープン (COM d:fileコマンド)

リスト装置のオープン (COM LST:コマンド)

コマンド・ファイルのクローズ (COM CON:コマンド)

出力の開始と停止方法

- 出力開始: ^O (CTRL+O)
 - 出力停止: ^O
- } 繰り返し押すと、交互に状態が変わる。

(d) コマンド・ヒストリの表示 (HISコマンド)

現在のコマンド・ヒストリ内容を、画面に表示 (20行) したり再実行したりする機能です。

指定行からの再実行: !n (n:行番号)

直前行の再実行 :!!

③ その他

(a) **DOS実行機能 (DOSコマンド)**

IE-75000-Rコントロール・プログラムから、OSに制御を渡す機能です。DOSコマンドを実行後はOSのコマンドを使用して作業ができます。

OSから再びIE-75000-Rコントロール・プログラムに戻るには、OSのEXITコマンドを使用します。

(b) **ヘルプ機能 (HLPコマンド)**

コマンドの入力形式や入力パラメータの説明など、使用方法について表示する機能です。

(c) **ディレクトリ表示機能 (DIRコマンド)**

指定されたディスク装置のファイル上の、ディレクトリを表示する機能です。

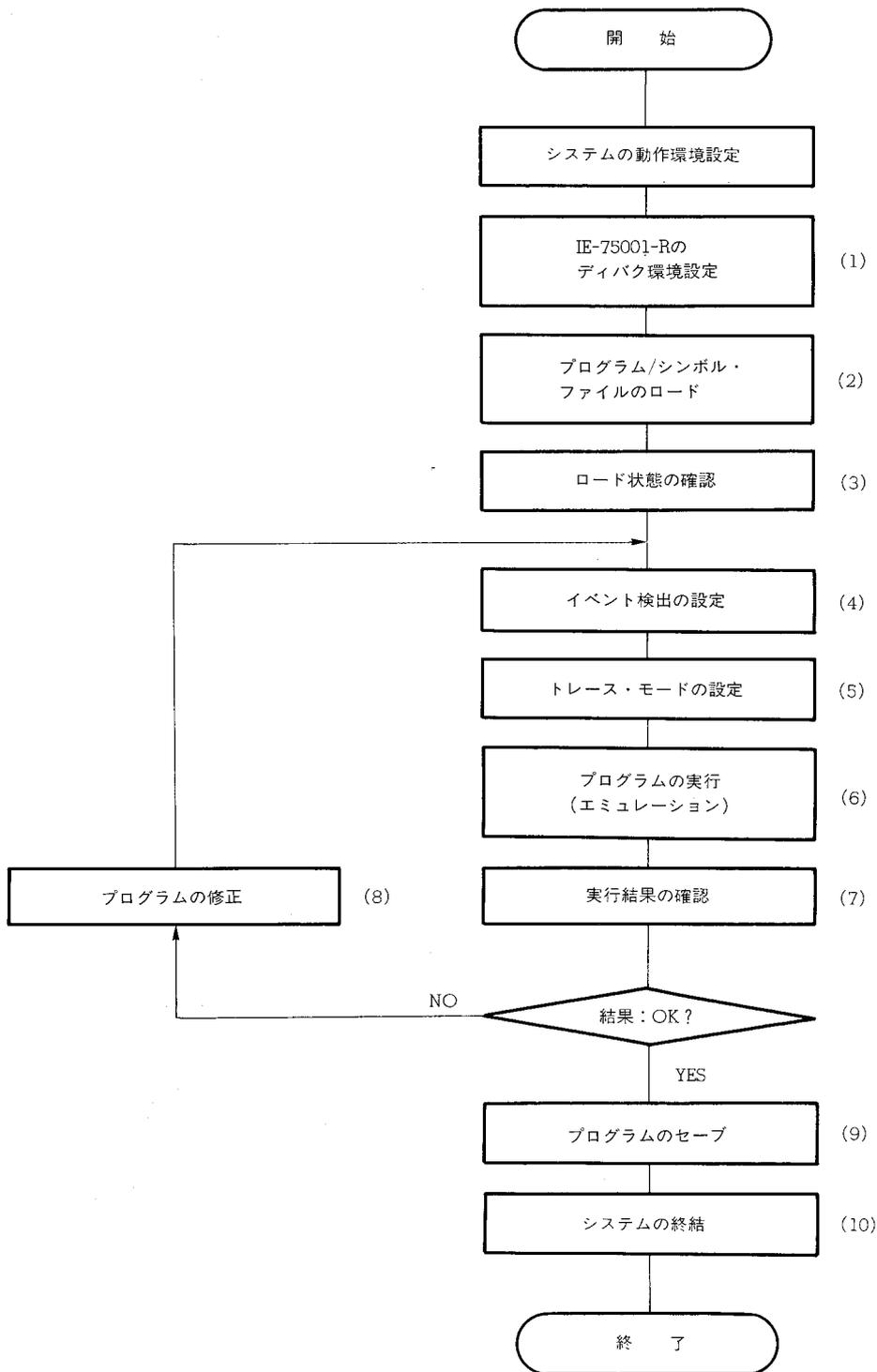
(d) **演算コマンド (MATコマンド)**

条件式を入力し、演算結果を画面に表示する機能です。

6.3 ディバグの基本手順

本節では次に示すフロー・チャートに従ってディバグの基本手順について説明します。

図 6-10 ディバグの基本手順フロー・チャート



(1) IE-75001-Rのディバグ環境設定

システムの起動が完了すると、ブレイク・モードとなり一連のコマンドの入力が可能となります。
ディバグ作業に先立って、次のコマンドを使用しディバグ環境の設定を行います。

CLK : クロックの選定
OUT : イベント・トリガ信号の出力指定
LOD file D : ホスト・マシンのファイルから、ディバグ環境情報をロード（プログラムのロード時に、一括ロードが可能）

(2) プログラム／シンボルのロード

ディバグの対象となるプログラムおよびシンボルをロードします。

(a) プログラムをロードするときは、次のコマンドを使用します。

LOD file C : ホスト・マシン上のオブジェクト・ファイルからロード
PGM : PROMプログラマ（PGシリーズ）からロード

(b) シンボルをロードするときは、以下のコマンドを使用します。

LOD file S : ホスト・マシン上のシンボル・ファイルからロード
SYM L : アペンド・シンボルをロード

(3) ロード後の状態確認

ディバグ環境情報、プログラムおよびシンボルのロードが完了したら、この状態が正常かどうかを次のコマンドを用いて確認することができます。

DAS : プログラムの逆アセンブルを行って確認
RAM : データ・メモリの内容を確認
SYM D : シンボルの確認
VRY : プログラム・ロード直後の、ファイルとメモリ内容の比較検証

(4) イベント検出条件の設定

ディバグの初期段階では、設定したポイントごとにイベントの検出を行って、ケース・バイ・ケースでディバグ作業を進めていきます。

イベント検出には、次のコマンドを用います。

BRA	: データ・メモリに関する検出条件の設定
BRS	: プログラム・メモリに関する検出条件の設定
PAS	: パス・カウンタの設定
BRM	: BRA, BRSなどで設定された検出条件を、プログラムの実行時に有効とするかどうか、最終的に決定
DLY	: イベント検出のディレイ

(5) トレース・モードの設定

ディバグで必要とするトレース情報を、トレース領域(メモリ)に出力するよう指定します。このデータは実行結果の確認において使用されます。

トレース方法の指定は、次のコマンドを用いて行います。

TRX	: クオリファイ・トレース時の条件設定
TRY	: セクション・トレース時の条件設定
TRM ALL	: 無条件トレース・モードの設定
TRM TRX	: クオリファイ・トレース・モードの設定
TRM TRY	: セクション・トレース・モードの設定

(6) プログラムの実行

ディバグ対象となるプログラムを実行(エミュレーション)します。実行の形態には、次の3種類があります。

RUN N	: ノンブレイクのリアルタイム実行
RUN B	: ブレイク付きのリアルタイム実行
RUN T	: ステップ実行

(7) 実行結果の確認

イベント検出によって、対象デバイスおよびトレーサが停止したなら、次のコマンドを用いて実行結果の確認を行います。

REG D	: 汎用レジスタの内容確認
SPR D	: 特殊レジスタの内容確認
MEM D	: プログラム・メモリの内容確認
RAM D	: データ・メモリの内容確認
TRD	: トレース・データの内容確認
TRF	: トレース・データ検索条件の設定
TRP	: トレース・ポインタの操作

(8) プログラムの修正

プログラムに修正を加える必要が生じた場合には、次のコマンドを用いてプログラムやデータの内容を変更します。

ただし、大幅な修正を必要とする場合には、ソース・プログラムの修正を行うほうが効率的です。

ASM	: ニモニック・コードによる、プログラムおよびデータの修正
MEM C	: 16進コードによるプログラム・メモリ内容の修正
RAM C	: 16進コードによるデータ・メモリ内容の修正

(9) プログラム/シンボル/ディバグ環境情報のセーブ

プログラムのディバグ作業を完了したなら、プログラム、シンボル、ディバグ環境情報をセーブします。

ホスト・マシンのファイルには、プログラム、ディバグ環境情報の一括セーブができますが、以下は個別セーブの例です。

- (a) プログラムをセーブするときは、次のコマンドを使用します。

SAV file C : ホスト・マシン上のオブジェクト・ファイルへセーブ
PGM : PROMプログラム (PGシリーズ) へセーブ

- (b) アペンド・シンボルをセーブするときは、次のコマンドを使用します。

SYM S : ホスト・マシン上のアペンド・シンボル・ファイルへアペンド・シンボル
をセーブ

- (c) ディバグ環境情報をセーブするときは、次のコマンドを使用します。

SAV file D : ホスト・マシン上のディバグ環境情報ファイルへセーブ

(10) システムの終結

すべてのディバグ作業を完了させ、システムの終結を行います。これにより制御はOSに戻され、ホスト・マシンとIE-75001-Rは論理的に切り離されます。

なお、継続してほかのプログラムをディバグする場合には、(1) ディバグ環境の設定から行います。システムの終結には次のコマンドを用います。

EXT : IE-75000-Rコントロール・プログラムの終了

6.4 基本機能の使用例

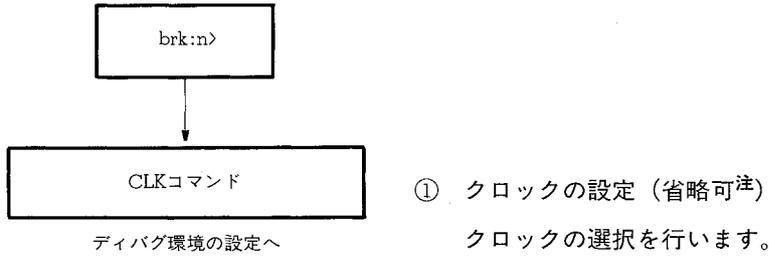
この節では、IE-75001-Rの初期設定からシステムの終結処理まで、順を追いながら基本的な機能の使い方について説明していきます。

はじめてIE-75001-Rを使用される方は、この節の説明と使用例を見ながら操作してみてください。一通りの操作と基本的機能が理解できます。

6.4.1 IE-75001-Rの初期設定

システムの起動が完了すると、ブレーク・モード (brk:n>) となります。この時点で、IE-75001-Rの初期設定を行います。

(a) IE-75001-Rの初期設定手順



注 省略すると内蔵クロック (4.19 MHz) でエミュレーションを行います。

(b) 操作の例

システムの起動を完了し、IE-75001-Rの初期設定を行った例です。

```

brk:0>CLK [enter] ①
IE ②
brk:0>CLK U [enter] ③
brk:0>CLK [enter] ④
User
brk:0>
  
```

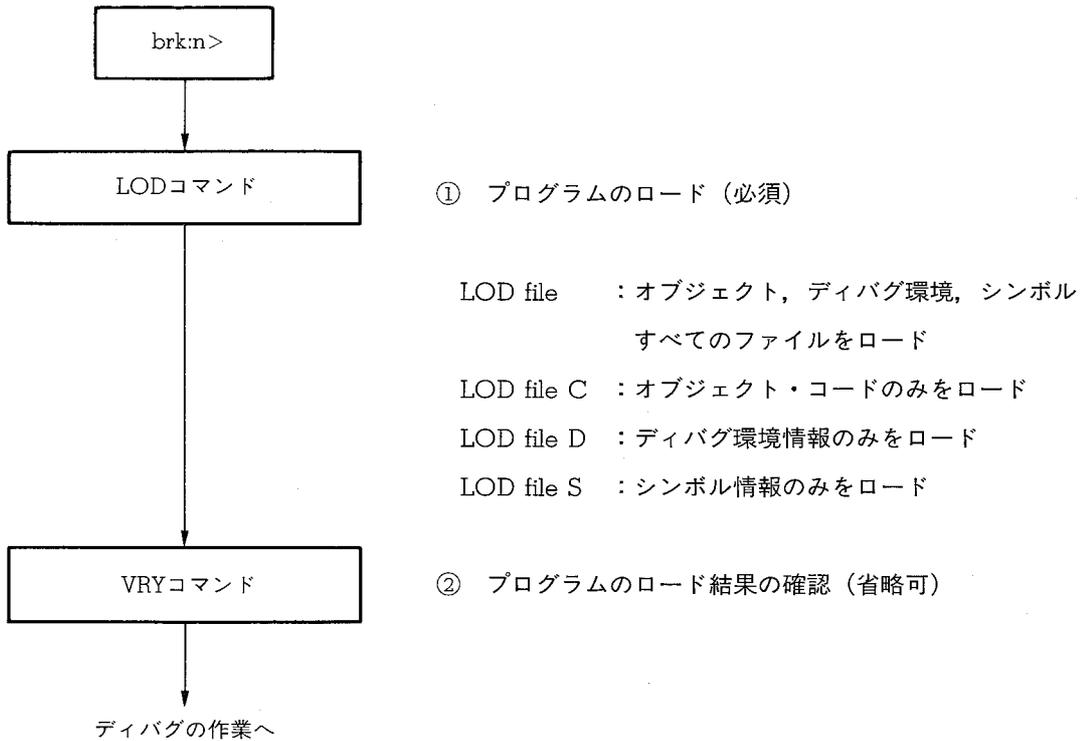
- ① 現在選択されているクロックの確認
- ② 現在選択されているクロックの表示
- ③ ターゲット・システム側のクロックに変更^注
- ④ 変更した内容の確認

注 ターゲット・システム側のクロックに変更する場合は、エミュレーション・ボードの設定変更も必要です。

6.4.2 デバッグ環境の設定

IE-75001-Rの初期設定を完了させ、ターゲット・プログラムのロードなどデバッグ環境の設定を行います。

(a) デバッグ環境の設定手順



(b) 操作の例

ホスト・マシンから、IE-75001-Rのチャンネル1 (CH1) またはチャンネル2 (CH2) へ、オブジェクト・ファイル、デバッグ環境ファイル、シンボル・ファイルを一括ロードした例です。

```
brk : 0>
brk : 0>
brk : 0>LOD B : SAMPLE ①
  Debug condition load (Y/N) ? Y ②
  Debug condition load complete ③
  object load complete ④
  symbol table loading
  HEIKIN      load complete
  TASU       load complete ⑤
  WARU       load complete

brk : 0>VRY B : SAMPLE.HEX ⑥
  object verify complete ⑦
brk : 0>■
```

- ① 一括ロードの指定 (ファイル名に拡張子は不要です)
- ② デバッグ環境のロード判断とロード指示
- ③ デバッグ環境のロード完了表示
- ④ オブジェクトのロード完了表示
- ⑤ シンボルのロード完了表示
- ⑥ オブジェクト・ファイルとロード結果の比較
- ⑦ 比較結果の正常終了表示

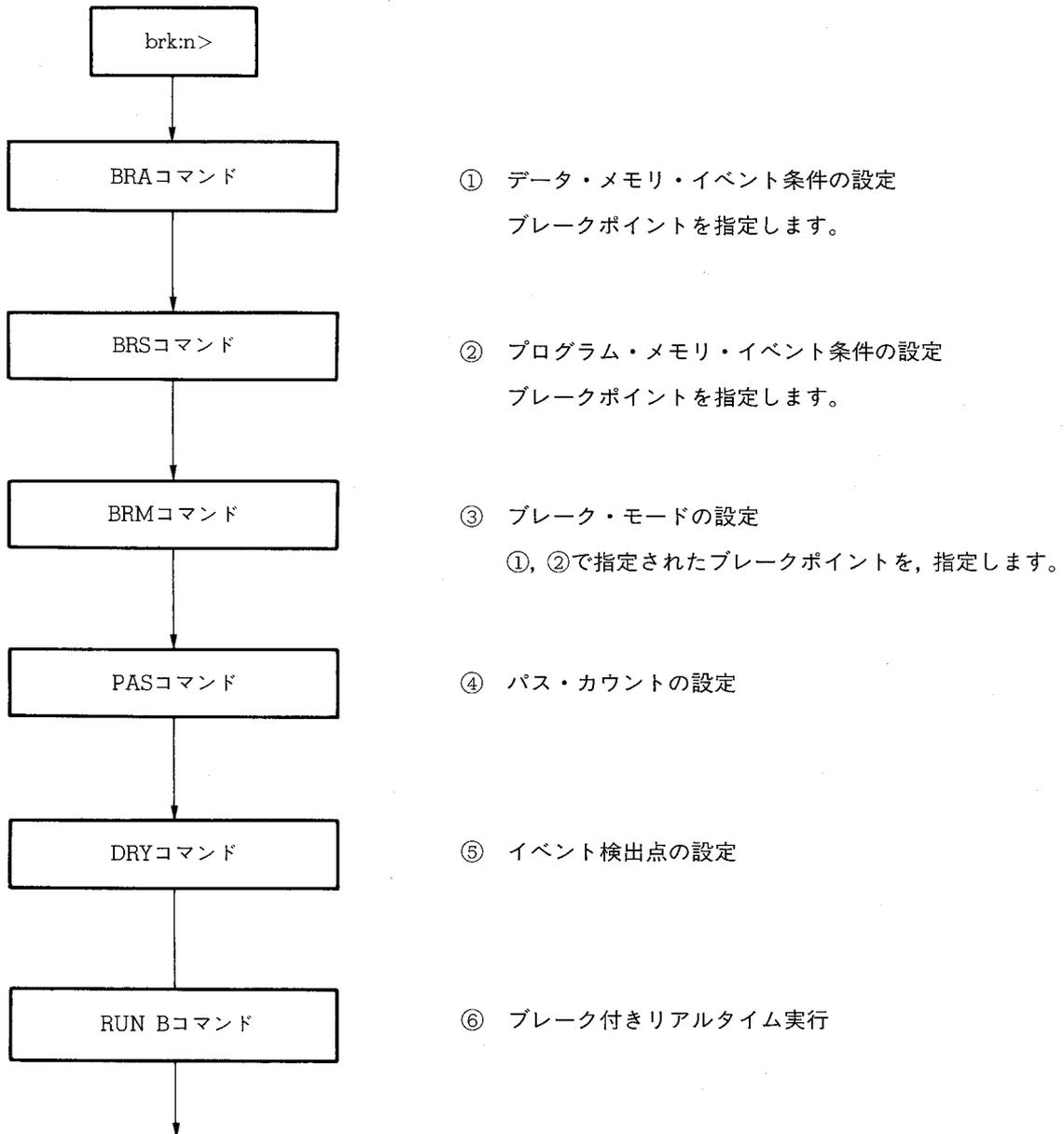
6.4.3 ターゲット・プログラムの実行

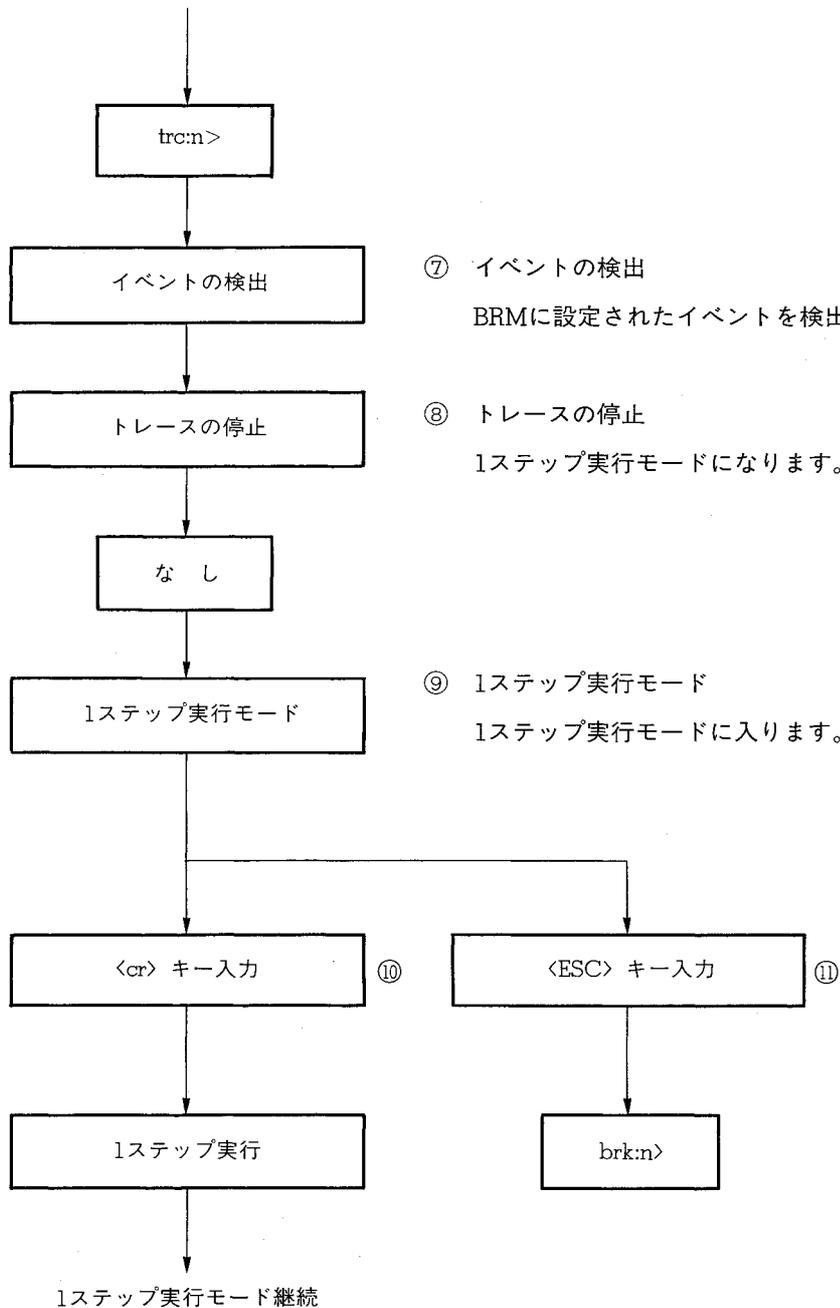
初期設定およびディバグ環境の設定が完了すると、いよいよ本来の作業であるターゲット・プログラムのディバグに入ります。

ここでは、ターゲット・プログラムの実行について説明します。

(1) 無条件トレース指定によるターゲット・プログラムの実行例

(a) 操作の手順





⑦ イベントの検出
BRMに設定されたイベントを検出します。

⑧ トレースの停止
1ステップ実行モードになります。

⑨ 1ステップ実行モード
1ステップ実行モードに入ります。

⑩ 1ステップ実行
 <cr> キーが入力されると、1ステップを実行します。そのまま、1ステップ実行モードが継続されます。
 1ステップ実行では、1命令ごとのトレース情報を画面に表示します。

⑪ <ESC>キーが入力されると、ブレーク・モードとなります。

注意 1ステップ実行を行うと、直前までにトレース・メモリに書き込まれた情報は消去されます。

(b) 操作の例

データ・メモリ・アドレスの、0100H番地、0101H番地がアクセスされたとき、またはプログラム・メモリの、2000H番地の命令が実行されたときにブレークするよう、指定した例です。

```

brk : 0> BRA 1 MA=0100 V=00001111Y C=W8 ①
brk : 0> BRA 2 MA=0101 V=0000XXXXY C=R8 ②
brk : 0> BRS 1 P 2000 ③
brk : 0> BRM BRA1 BRA2 BRS1 ④
brk : 0> DLY M ⑤
brk : 0> RUN B 100 ⑥
Emulation start at 100 ⑦
trc : 0> ■ ⑧
BRS1 break terminated ⑨
XA HL DE BC XA' HL' DE' BC' RBS MBS RBE CY ISTO SP PC } 注
00 00 00 00 00 00 00 00 0 0 0 0 0 000 0000
One step emulation standby ⑩
    
```

注 対象デバイスにより、存在しないレジスタは--で表示されます。

- ① イベント検出条件として、次の内容をBRA1に設定

データ・メモリ・アドレス：0100H番地 (16進)
 データ・メモリ内容 (値)：00001111Y (2進)
 メモリ・アクセス条件：有効ビット=8ビット
 I/Oモード=ライト, リード・モディファイ・ライト

- ② イベント検出条件として、次の内容をBRA2に設定

データ・メモリ・アドレス：0101H番地 (16進)
 データ・メモリ内容 (値)：0000XXXXY (2進マスク指定)
 メモリ・アクセス条件：有効ビット=8ビット
 I/Oモード=ライト, リード・モディファイ・ライト

備考 X=不定データ, Y=2進数表現

- ③ イベント検出条件として、次の内容をBRS1に設定

プログラム・メモリ・アドレス：2000H番地（16進）

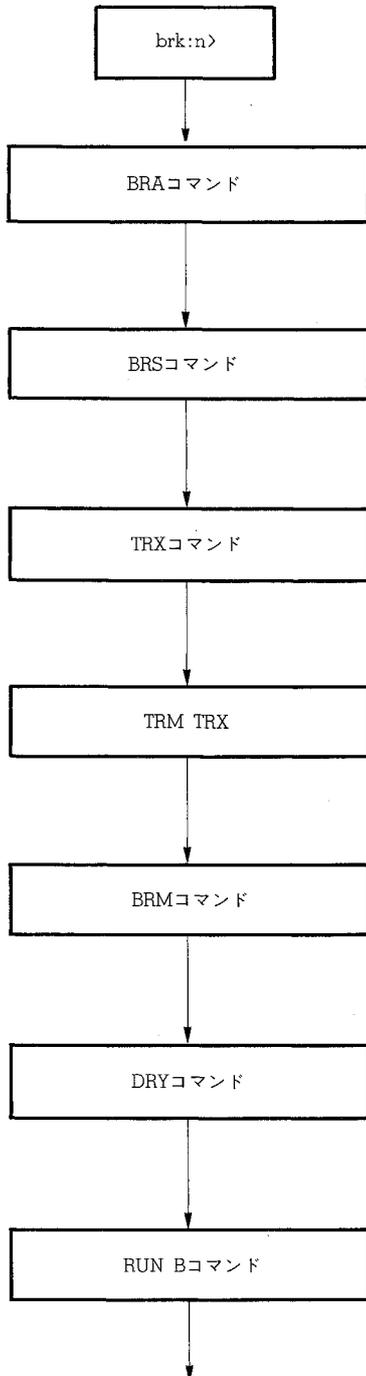
- ④ 次のイベント検出条件内容をBRMに設定

BRA1, BRA2, BRS1

- ⑤ イベント検出点をトレース・メモリの「中央フレーム」に設定
- ⑥ プログラム・アドレスの100H番地から、エミュレーションを開始するよう指定
- ⑦ プログラム・アドレスの100H番地から、エミュレーションを開始した表示
- ⑧ トレース・モードでコマンド待ちの状態
- ⑨ BRS1の要因（プログラム・アドレスの2000H番地）でブレイクしたことを示す表示
- ⑩ 1ステップ実行モードに入った表示

② クオリファイ・トレース指定によるターゲット・プログラムの実行例

(a) 操作の手順



- ① データ・メモリ・イベント条件の設定
ブレークポイントおよびトレース・ポイント，データ値などを指定します。
- ② プログラム・メモリ・イベント条件の設定
ブレークポイントおよびトレース・ポイントを指定します。
- ③ クオリファイ・トレース条件の設定
①，②で指定されたトレース・ポイントを指定します。
- ④ トレース・モードの設定
TRX（クオリファイ・トレース）を指定します。
- ⑤ ブレーク・モードの設定
①，②で指定されたブレークポイントを指定します。
- ⑥ イベント検出点の設定
- ⑦ ブレーク付きリアルタイム実行

(b) 操作の例

データ・メモリ・アドレスの0100H番地, 0101H番地がアクセスされたとき, またはプログラム・メモリの, 1000H番地の命令が実行されたときだけトレース・データを取り, 3000H番地の命令が実行されたときにブレークするよう, 指定する例です。

```

brk : 0> BRA 1 MA=0100 V=00001111Y C=W8 ①
brk : 0> BRA 2 MA=0101 V=0000XXXXY C=R8 ②
brk : 0> BRS 1 P 1000 ③
brk : 0> BRS 2 3000 ④
brk : 0> TRX BRA1 BRA2 BRS1 ⑤
brk : 0> TRM TRX ⑥
brk : 0> BRM BRS2 ⑦
brk : 0> DLY M ⑧
brk : 0> RUN B 100 ⑨
Emulation start at 100 ⑩
trc : 0> ■ ⑪
BRS2 break terminated ⑫
XA HL DE BC XA' HL' DE' BC' RBS MBS RBE CY ISTO SP PC } 注
00 00 00 00 00 00 00 00 0 0 0 0 0 000 0000
One step emulation standby ⑬
    
```

注 対象デバイスにより, 存在しないレジスタは--で表示されます。

① イベント検出条件として, 次の内容をBRA1に設定

データ・メモリ・アドレス : 0100H番地 (16進)

データ・メモリ内容 (値) : 00001111Y (2進)

メモリ・アクセス条件 : 有効ビット=8ビット

I/Oモード=ライト, リード・モディファイ・ライト

備考 Y : 2進数表示

- ② イベント検出条件として、次の内容をBRA2に設定

データ・メモリ・アドレス：0101H番地（16進）

データ・メモリ内容（値）：0000XXXXY（2進マスク指定）

メモリ・アクセス条件：有効ビット=8ビット

I/Oモード=リード，リード・モディファイ・ライト

備考 X：不定データ，Y：2進数表示

- ③ イベント検出条件として、次の内容をBRS1に設定

プログラム・メモリ・アドレス：1000H番地（16進）

- ④ イベント検出条件として、次の内容をBRS2に設定

プログラム・メモリ・アドレス：3000H番地（16進）

- ⑤ クオリファイ・トレースのトレース・ポイントとして次の内容をTRXに設定

BRA1, BRA2, BRS1

- ⑥ クオリファイ・トレースを設定

- ⑦ イベント検出条件の統合、次の内容をBRMに設定

BRS2

- ⑧ イベント検出点を、トレース・メモリの「中央フレーム」に設定

- ⑨ プログラム・アドレスの100H番地から、エミュレーションを開始するよう指定

- ⑩ プログラム・アドレスの100H番地から、エミュレーションを開始した表示

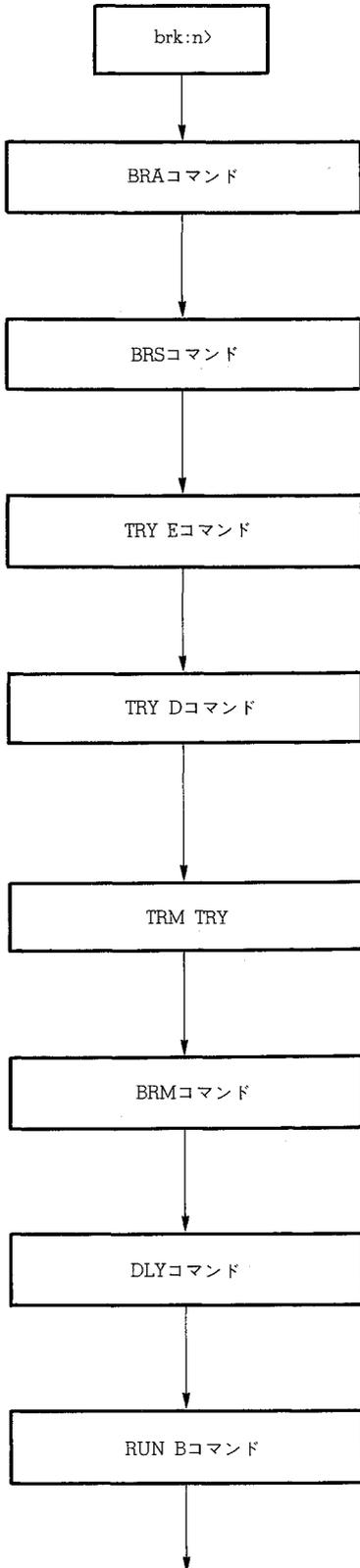
- ⑪ トレース・モードでコマンド待ちの状態

- ⑫ BRS2の要因（プログラム・アドレスの3000H番地）で、ブレイクしたことを示す表示

- ⑬ 1ステップ実行モードに入った表示

③ セクション・トレース指定によるターゲット・プログラムの実行例

(a) 操作の手順



- ① データ・メモリ・イベント条件の設定
ブレークポイントおよびトレース・ポイント, データ値などを指定します。
- ② プログラム・メモリ・イベント条件の設定
ブレークポイントおよびトレース・ポイントを指定します。
- ③ トレース・イネーブル・ポイントの設定
①, ②で指定されたトレース・ポイントを指定します。
- ④ トレース・ディスエーブル・ポイントの設定
①, ②で指定されたトレース・ポイントを指定します。
- ⑤ トレース・モードの設定
TRY (セクション・トレース) を指定します。
- ⑥ ブレーク・モードの設定
①, ②で指定されたブレークポイントを指定します。
- ⑦ イベント検出点の設定
- ⑧ ブレーク付きリアルタイム実行

以下, 無条件トレースに同じ

(b) 操作の例

データ・メモリ・アドレスの0100H番地がアクセスされたときからトレースを開始し、データ・メモリ・アドレスの0200H番地がアクセスされたときにトレースを停止し、1000H番地の命令が実行されたときにブレークするように指定した例です。

```

brk : 0 > BRA 1 MA=0100 V=00001111Y C=W8 ①
brk : 0 > BRA 2 MA=0200 V=0000XXXXY C=R8 ②
brk : 0 > BRS 1 P 1000 ③
brk : 0 > TRY E BRA1 ④
brk : 0 > TRY D BRA2 ⑤
brk : 0 > TRM TRY ⑥
brk : 0 > BRM BRS1 ⑦
brk : 0 > DLY M ⑧
brk : 0 > RUN B 100 ⑨
Emulation start at 100 ⑩
trc : 0 > ■ ⑪
BRS1 break terminated ⑫
XA HL DE BC XA' HL' DE' BC' RBS MBS RBE CY ISTO SP PC
00 00 00 00 00 00 00 00 0 0 0 0 0 000 0000 } 注
One step emulation standby ⑬
    
```

注 対象デバイスにより，存在しないレジスタは--で表示されます。

① イベント検出条件として，次の内容をBRA1に設定

データ・メモリ・アドレス：0100H番地（16進）

データ・メモリ内容（値）：00001111Y（2進）

メモリ・アクセス条件：有効ビット=8ビット

I/Oモード=ライト，リード・モディファイ・ライト

- ② イベント検出条件として、次の内容をBRA2に設定

データ・メモリ・アドレス：0200H番地（16進）

データ・メモリ内容（値）：0000XXXXY（2進マスク指定）

メモリ・アクセス条件：有効ビット=8ビット

I/Oモード=リード，リード・モディファイ・ライト

備考 X：不定データ，Y：2進数表示

- ③ イベント検出条件として、次の内容をBRS1に設定

プログラム・メモリ・アドレス：1000H番地（16進）

- ④ セクション・トレースのトレース開始点として、次の内容をTRY Eに設定

BRA1

- ⑤ セクション・トレースのトレース停止点として、次の内容をTRY Dに設定

BRA2

- ⑥ セクション・トレースを設定

- ⑦ イベント検出条件の統合、次の内容をBRMに設定

BRS1

- ⑧ イベント検出点を、トレース・メモリの中央フレームに設定

- ⑨ プログラム・アドレスの100H番地から、エミュレーションを開始するよう指定

- ⑩ プログラム・アドレスの100H番地から、エミュレーションを開始した表示

- ⑪ トレース・モードでコマンド待ちの状態

- ⑫ BRS1の要因（プログラム・アドレスの1000H番地）で、ブレークしたことを示す表示

- ⑬ 1ステップ実行モードに入った表示

6.4.4 実行結果の確認

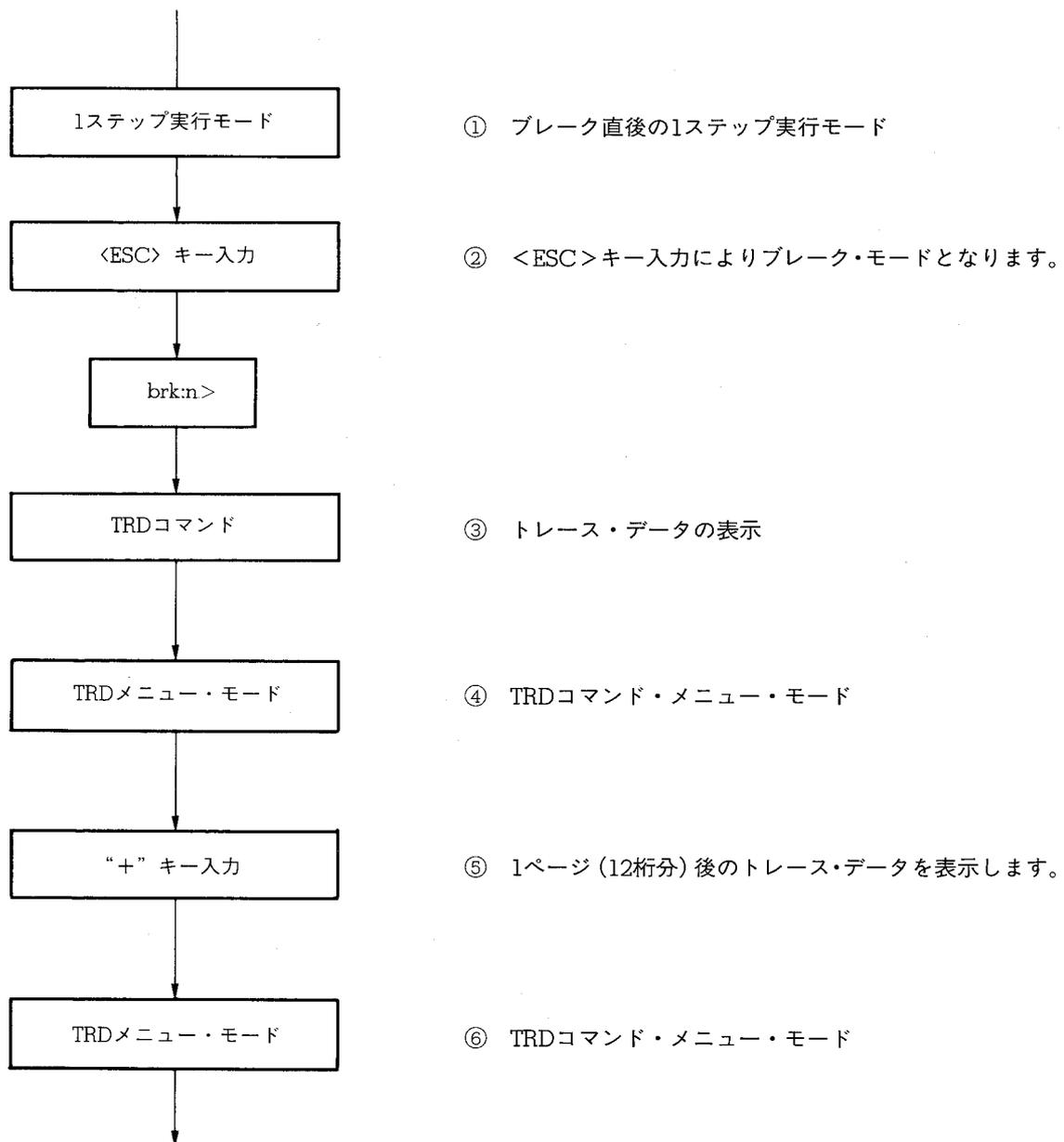
ターゲット・プログラムを実行し、イベントを検出してブレーク状態になったのち、トレース・データ、レジスタ値、特殊レジスタ値、メモリの内容を調べて実行結果の確認を行います。

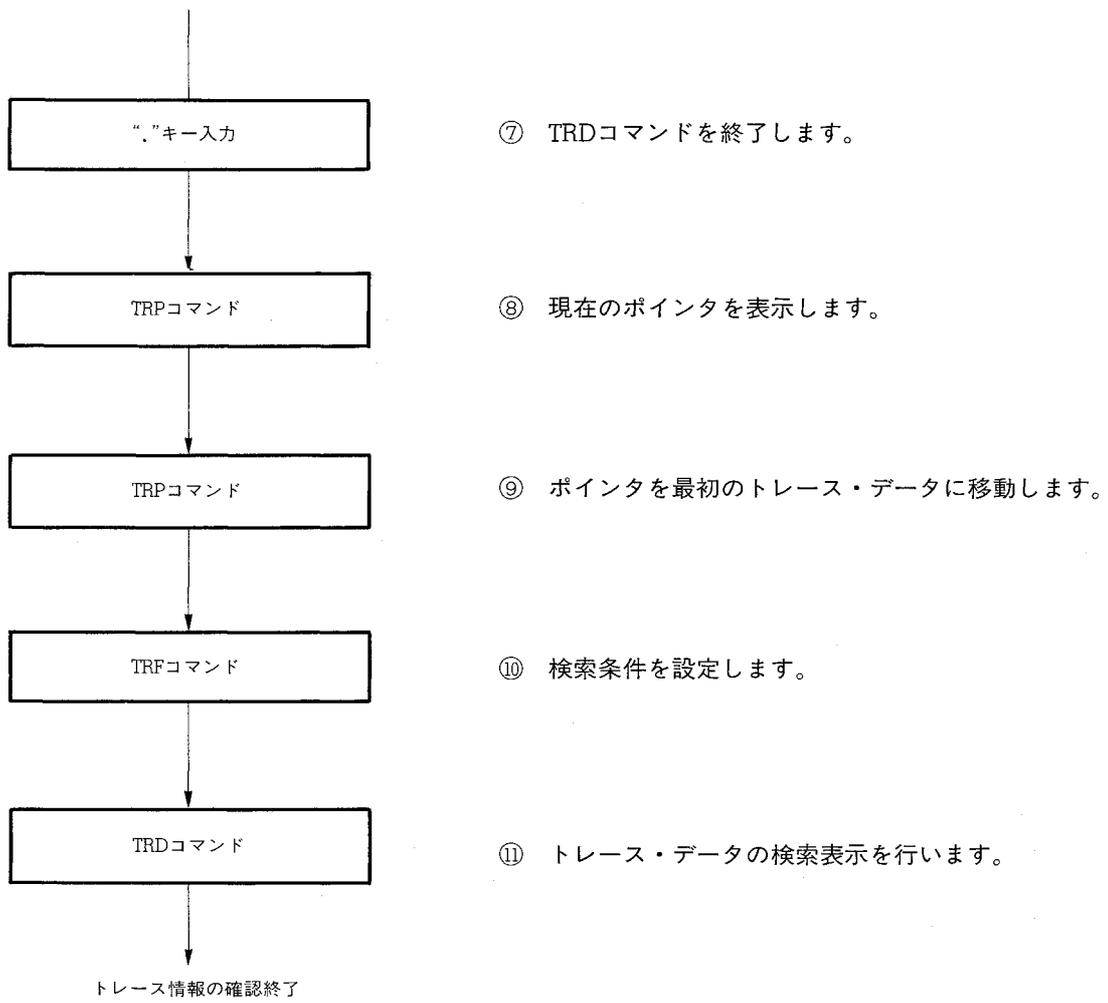
(1) トレース・データの確認

RUN Bコマンドでターゲット・プログラムの実行を開始し、イベントを検出後、トレース情報を確認する場合の手順を示します。イベント検出後、1ステップ実行モードになるまでの手順は、6.4.

3 ターゲット・プログラムの実行を参照してください。

(a) 操作の手順





(b) 操作の例

```

One step emulation standby ESC
brk : 0>TRD DMEM

Frame PA PD MA MD MRW Label Mnemonic
00C 0104 8A 002 02 MRW INCS HL
00D 0105 AA 4A --- -- --- SKE HA, HL
00E 002 02 MRD
00F 0107 AB 01 02 --- -- --- BR 10102H
010 --- -- ---
011 --- -- ---
012 0102 89 11 --- -- --- MOV XA, #0011H
013 000 11 MWR
014 0104 8A 002 03 MRW INCS HL
015 0105 AA 4A --- -- --- SKE XA, HL
016 002 03 MRD
017 0107 AB 01 02 --- -- --- BR 10102H
Total frame=1A1 (L/F/T+/cr/-/Frame No./.) ? +
Frame PA PD MA MD MRW Label Mnemonic
018 --- -- ---
019 --- -- ---
01A 0102 89 11 --- -- --- MOV XA, #0011H
01B 000 11 MWR
01C 0104 8A 002 04 MRW INCS HL
01D 0105 AA 4A --- -- --- SKE XA, HL
01E 002 04 MRD
01F 0107 AB 01 02 --- -- --- BR 10102H
020 --- -- ---
021 --- -- ---
022 0102 89 11 --- -- --- MOV XA, #0011H
023 000 11 MWR
Total frame=1A1 (L/F/T+/cr/-/Frame No./.) ? .
brk : 0>TRP
Total frame number =1A1
Display frame pointer =024
brk : 0>TRF MA=0 MD=11H MRW=MWR
brk : 0>TRD ALL $Q DMEM
    
```

- ①
- ②
- ③
- ④
- ⑤
- ⑥
- ⑦
- ⑧
- ⑨

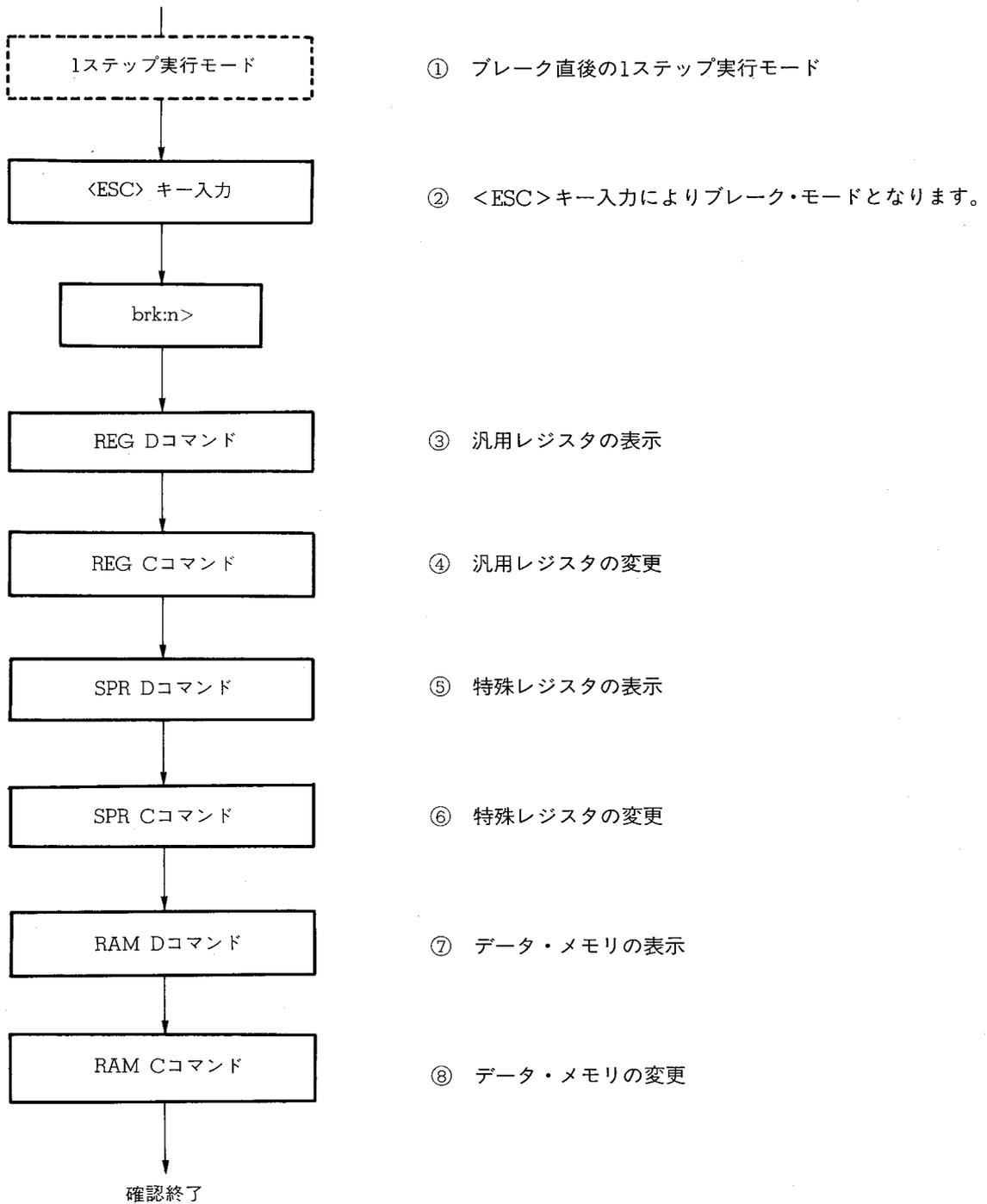
Frame	PA	PD	MA	MD	MRW	Label	Mnemonic
!002	0102	89	11	---	--	---	MOV XA, #0011H
003				000	11	MWR	
!00A	0102	89	11	---	--	---	MOV XA, #0011H
00B				000	11	MWR	
!012	0102	89	11	---	--	---	MOV XA, #0011H
013				000	11	MWR	
!01A	0102	89	11	---	--	---	MOV XA, #0011H
01B				000	11	MWR	
!022	0102	89	11	---	--	---	MOV XA, #0011H
023				000	11	MWR	
!02A	0102	89	11	---	--	---	MOV XA, #0011H
02B				000	11	MWR	
!032	0102	89	11	---	--	---	MOV XA, #0011H
033				000	11	MWR	
!03A	0102	89	11	---	--	---	MOV XA, #0011H
03B				000	11	MWR	
!042	0102	89	11	---	--	---	MOV XA, #0011H
043				000	11	MWR	
!04A	0102	89	11	---	--	---	MOV XA, #0011H
04B				000	11	MWR	
!052	0102	89	11	---	--	---	MOV XA, #0011H
053				000	11	MWR	
!05A	0102	89	11	---	--	---	MOV XA, #0011H
05B				000	11	MWR	
!062	0102	89	11	---	--	---	MOV XA, #0011H
063				000	11	MWR	
!06A	0102	89	11	---	--	---	MOV XA, #0011H
06B				000	11	MWR	
!072	0102	89	11	---	--	---	MOV XA, #0011H
073				000	11	MWR	
!07A	0102	89	11	---	--	---	MOV XA, #0011H
07B				000	11	MWR	
!082	0102	89	11	---	--	---	MOV XA, #0011H
083				000	11	MWR	
TOA3	0217	AB	02 04	---	--	---	BR !0204H

- ① 1ステップ実行モードを<ESC>キーにより終了
- ② トレース・データを確認するために、表示指定を実行
- ③ 現在のポインタより、1ページ表示し、メニュー・モードの状態
 - ・エミュレーション実行直後のトレース・ポインタは、トリガ・フレームを指しています。TRDコマンドではトリガ・フレームの前後5行を表示します。
 - ・トリガ・フレームが存在しない場合は、先頭のフレームに位置付けられます。
- ④ 次のページも確認するために、メニュー・モードで“+”を入力
- ⑤ TRDコマンドのメニュー・モードを終了する場合は、“.”を入力
- ⑥ 現在のポインタを確認
- ⑦ ポインタは、前回に表示したフレームの次を表示
- ⑧ データ・メモリ・アドレス0H番地に、011H番地をライトしたフレームを捜すために、検索条件を設定
- ⑨ 先頭のフレームより、検索条件に一致したフレームだけを表示

(2) レジスタ値，特殊レジスタ値，メモリ内容の確認

RUN Bコマンドでターゲット・プログラムの実行を開始し，イベントを検出後，レジスタ値，特殊レジスタ値，メモリ内容を確認し，必要があれば変更する場合の手順を示します。

(a) 操作の手順



(b) 操作の例

brk : 0>REG D	①
XA HL DE BC XA' HL' DE' BC' RBS MBS RBE MBE CY IST1 ISTO SP PC	②注
00 11 22 33 44 55 66 77 0 0 0 0 0 0 0 0 000 0178	
brk : 0>REG C	③
XA 00 = 11	④
HL 11 = 22	
DE 22 = 33	⑤注
BC 33 =	
XA' 44 =	
HL' 55 =	
DE' 66 =	
BC' 77 =	
RBS 0 =	
MBS 0 =	
SP 000 =	
PC 0178 =	
brk : 0>SPR D F8X	⑥
(F8X)	
SBS=1 BTM.33=.1 BTM=W ₀ BT=00 DSPM=W ₀ DIMS=W ₀	⑦
DIGS=W ₀ KSF.3=.0 KSF=2	
brk : 0>SPR C BTM	⑧
BTM W ₀ =1	⑨
brk : 0>RAM D 100,11F \$B	⑩
ADDR +0 +2 +4 +6 +8 +A +C +E +10+12+14+16+18+1A+1C+1E	⑪
0100 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F	
brk : 0>RAM C 100 \$B	⑫
0100 00=10	⑬
0102 01=11	
0104 02=12	
0106 03=.	⑭

注 対象デバイスにより、存在しないレジスタは--で表示されます。

- ① レジスタの内容を確認するために、表示指定の実行
- ② カレント・レジスタ・バンクの全汎用レジスタ、制御レジスタ、PSWフラグを表示
- ③ レジスタの内容変更を指定
- ④ レジスタの内容を変更

XAレジスタの内容を00Hから11Hに変更します。

HLレジスタの内容を11Hから22Hに変更します。

DEレジスタの内容を22Hから33Hに変更します。

- ⑤ 変更しないレジスタの場合は、〈cr〉キーを入力
- ⑥ 特殊レジスタの内容を確認するために表示指定の実行
- ⑦ グループ名“F8X”が指定されたので、そのグループに属す特殊レジスタを表示
- ⑧ 変更したい特殊レジスタを指定
- ⑨ 特殊レジスタBTMの内容を1に設定
- ⑩ データ・メモリの内容を確認するために、0100H番地からバイト単位で16バイト表示
- ⑪ データ・メモリの0100H番地からバイト単位で16バイト分のデータを表示
- ⑫ データ・メモリの0100H番地から0105H番地までの内容をバイト単位で変更する指定
- ⑬ データ・メモリの内容を変更

0100H番地 00Hから10Hへ変更します。

0102H番地 01Hから11Hへ変更します。

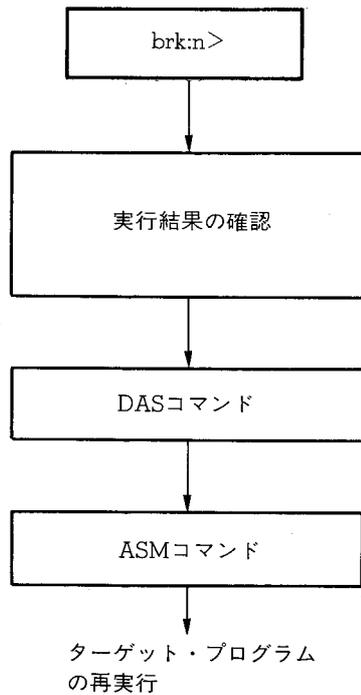
0104H番地 02Hから12Hへ変更します。

- ⑭ データ・メモリの変更を終了

6.4.5 プログラムの修正

ターゲット・プログラムを実行し、トレース・データ、レジスタの値、メモリの内容などで実行結果を調べて、プログラムに誤りがあることが分かった場合、プログラムの修正を行います。

(a) プログラムの修正手順



- ① トレース・データ、レジスタ値、メモリ内容などを確認します。

6.4.4 実行結果の確認を参照してください。

- ② 逆アセンブルの指定
プログラム・メモリを逆アセンブルしプログラムの確認を行います。
- ③ アセンブル指定
プログラムの修正を行います。

(b) 操作の例

プログラム・メモリ・アドレスの0100H番地から逆アセンブルを行い、0104H番地の命令を修正した例です。

```

brk : 0>DAS 100␣ ①
PA   Object      Mnemonic      ②
0100 AA 10        MOV          @HL, XA
0102 89 34        MOV          XA, #0034H
0104 89 56        MOV          XA, #0056H
0106 10          GET1        0020H
0107 IF          GET1        003EH
0108 89 00        MOV          XA, #0000H
010A 92 80        MOV          0080H, XA
010C 99 10        SEL          MBO
010E A3 30        MOV          A, 0030H
0110 49          PUSH        XA

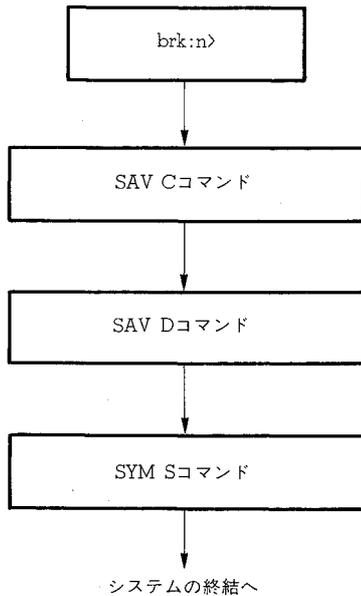
brk : 0>ASM 104␣ ③
0104 89 56        MOV          XA, #0056H      =MOV XA, #0078H␣ ④
      89 78
0106 10          GET1        0020H          =END␣ ⑤⑥
brk : 0>■
    
```

- ① プログラム内容を確認するために、0100H番地からプログラム・メモリの逆アセンブルを実行
- ② 指定されたアドレスから17バイト目のアドレスを含む命令までを逆アセンブルして表示
- ③ 0104H番地を修正するために、0104H番地からアセンブルの指定
- ④ 現在、メモリに設定されているコードと、逆アセンブル・イメージが表示され、0104H番地の命令をモニタにより修正
- ⑤ 修正した命令のオブジェクト・コードを表示
- ⑥ 次の命令のコードと逆アセンブル・イメージが表示され、目的の命令の修正が完了したので、アセンブルの終了“END␣”を入力

6.4.6 ディバグの終了

ディバグが一通り終了した場合、または途中でやめる場合、終了したときと同じ状態でディバグを再開できるようにするため、ディバグにより変更されたプログラム、ディバグ環境、および追加シンボルをセーブします。

(a) セーブの手順



① プログラムをオブジェクト・ファイルへセーブします。

② ディバグ環境情報を、ディバグ環境情報ファイルへセーブします。

③ アペンド・シンボルをアペンド・シンボル・ファイルへセーブします。

①②は、この例では分割してセーブしていますが、SAVコマンドにより一括してセーブすることもできます。

(b) 操作の例

IE-75001-Rから、ホスト・マシンへ、オブジェクト、デバッグ環境、アペンド・シンボルをセーブした例です。

```
brk : 0>SAV SAMPLE C ①
object save complete ②
brk : 0>SAV SAMPLE D ③
debug data save complete ④
brk : 0>SYM S ⑤
brk : 0>■
```

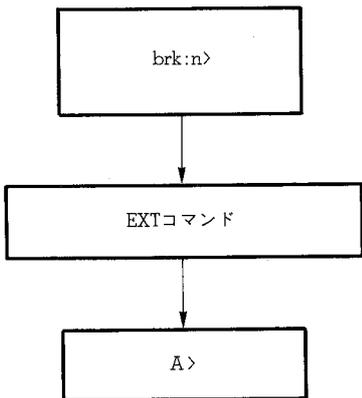
- ① プログラム・メモリの0H-OFFFFHの範囲をホスト・マシンのカレント・ドライブに“SAMPLE.HEX” というファイル名でセーブ
- ② オブジェクトのセーブが完了したことを表示
- ③ デバッグ環境情報をホスト・マシンのカレント・ドライブに “SAMPLE.DBG” というファイル名でセーブ
- ④ デバッグ環境情報のセーブが完了したことを表示
- ⑤ アペンド・シンボルをホスト・マシンのカレント・ドライブに “IE75000.SYM” というファイル名でセーブ

6.4.7 システムの終結

すべてのデバッグ作業を完了させ、IE-75000-Rコントロール・プログラムを終了しシステムの終結を行います。

業務を終了した際には、必ずシステムの終結を行ってください。

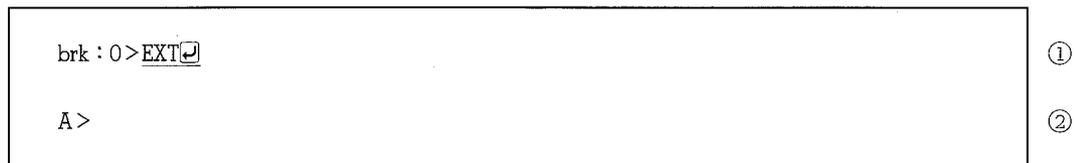
(a) システムの終結手順



- ① IE-75000-Rコントロール・プログラムの終了
IE-75000-Rコントロール・プログラムを終了し、制御をOS (DOS) に戻します。
- ② OSに制御が戻ります。

(b) 操作の例

EXTコマンドによって、IE-75000-Rコントロール・プログラムを終了し、OSに制御を戻した例です。



- ① IE-75000-Rコントロール・プログラムを終了
- ② OSに制御が戻り、OSがプロンプトを表示

第7章 ファームウェアROMのバージョンについて

7.1 追加機能について

1993年2月にIE-75001-RのファームウェアROMをバージョン・アップ (Ver.1.4) しました。同じく2月にバージョン・アップ (Ver.1.1) したコントロール・プログラムと合わせて使うと、次の機能が新たに使えるようになります。詳細については第8章 コマンド説明の各コマンドの説明を参照してください。

★

- ① 起動時の設定にセットアップ・ファイルの設定が追加され、2回目以降は自動で立ち上げ可能
- ② BRAコマンドのアドレス指定時に、NOT条件が設定可能
- ③ BRAコマンドのデータ値指定時に、NOT条件が設定可能
- ④ 対話形式でもBRSコマンドの指定が可能
- ⑤ BNKコマンドが、エミュレーション・モードでも使用可能
- ⑥ TRPコマンドで絶対アドレスの入力可能

7.2 バージョンによる機能の違い

ファームウェアROMとコントロール・プログラムのそれぞれのバージョンによってIE-75001-Rの機能や動作に違いがあります。

表 7-1 ファームウェアROMと、コントロール・プログラムのバージョンの違いによる動作

★

ファームウェアROM / コントロール・プログラム	Ver. 1.0-1.3	Ver.1.4
Ver.1.0	動作します	動作しません注1
Ver.1.1	動作します	動作します注2

注1. Ver.1.1のコントロール・プログラムを使用してください。

2. 追加機能を使用できます。

第8章 コマンド説明

この章では、IE-75001-Rで使用される全コマンドについて、その表記法、特殊キーの使用法などの使用条件および機能の詳細を説明します(画面表示は、IE-75001-RにIE-75000-R-EMを接続した場合を例にしています)。

なおコマンドの説明は、索引を配慮してアルファベット順に構成してあります。

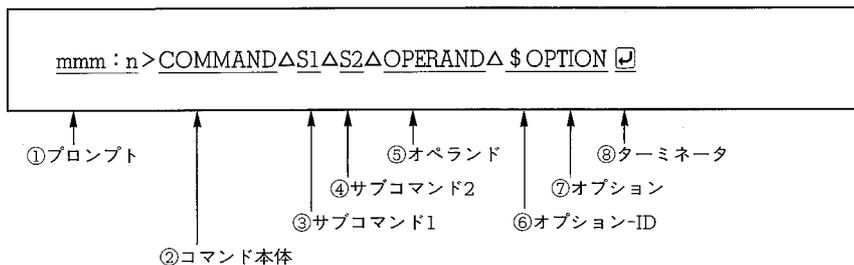
	ASM	A
BRS, BRM, BRK, BRA, BNK		B
CVM, CVD, COM, CLK, CHK		C
DOS, DLY, DIR, DAS		D
	EXT	E
	HLP, HIS	H
	LST, LOD	L
	MOD, MEM, MAT	M
	OUT	O
	PGM, PAS	P
RUN, RES, REG, RAM		R
SYS, SYM, STS, STP, STR, SPR, SET, SAV		S
TRY, TRX, TRP, TRM, TRG, TRF, TRD		T
	VRY	V

8.1 コマンド表記法

8.1.1 コマンド形式

(1) コマンド入力形式

コマンドの一般的な入力形式は次のとおりです。入力は1行単位で行います。



① プロンプト

モニタが表示するプロンプトで、コマンドの入力を促すものです。表示内容は、システムの動作状態を示しています。

mmm : システム動作モード

n : ホスト・マシンのシリアル・ポート番号

プロンプトの表示	システム動作モード
brk : n >	ブレーク・モード
emu : n >	エミュレーション・モード
trc : n >	トレース・モード

② コマンド本体

コマンドの本体で、その機能を3文字の省略形で表現しています。

入力時に省略することはできません。

③ サブコマンド1

コマンド本体の機能を修飾します。

サブコマンド1以下は、入力を省略することもできます。

④ サブコマンド2

コマンド本体およびサブコマンド1の機能を修飾します。サブコマンド2は、一部のコマンドに適用されるもので、一般形式では存在しません。

⑤ オペランド

コマンドの機能を実現する際の、具体的な数値や文字列を指定します。

⑥ オプション-ID

オプション入力を行うときの識別子です。オプションを行うときには、必ずこの識別子"\$"を先行して入力します。

⑦ オプション

オペランドの修飾をします。

⑧ ターミネータ

コマンド入力の終了を示します。

(2) コマンド記述一般則

記 述	説 明
英大文字	原則として、コマンド説明に記述されているとおりに入力することを示します。 例 メイン・コマンド、オプションなど
英小文字	原則として、ユーザが判断または選択して該当数値などを、入力することを示します。 例 オペランドの入力
{文字列}	{ }の中に記述されている文字列のどれかを、選択することを示します。
[文字列]	[]の中に記述されている文字列の入力が、省略できることを示します。
(文字列)	()の中に記述されている文字列は、サブコマンドおよびオペランドの説明です。
△	スペース入力することを示します。コマンドの各要素を区切るセパレータとして使用されます。
¥	IBM PCではバック・スラッシュ “\” を入力します。

8.1.2 コマンド構成要素の説明

(1) メイン・コマンド

コマンドの本体で、その機能を3文字の省略形で表現しています。コマンド名で記述されているとおりに入力してください。省略することはできません。

表8-1 コマンド一覧 (1/2)

○：使用可， ×：使用不可

項番	コマンド名	機 能	システム動作モード		
			brk	emu	trc
1	ASM	アセンブル	○	×	×
2	BNK	データ・メモリ・バンク設定	○	○	×
3	BRA	データ・メモリ・イベント条件設定	○	○	×
4	BRK	イベント情報表示	○	○	×
5	BRM	イベント・モード設定	○	○	×
6	BRS	プログラム・メモリ・イベント条件設定	○	○	×
7	CHK	チェック・ポイント設定	○	○	×
8	CLK	クロック選択	○	×	×
9	COM	コマンド・ファイル作成	○	○	○
10	CVD	カバレッジ測定結果操作	○	×	×
11	CVM	カバレッジ測定範囲設定	○	×	×
12	DAS	逆アセンブル	○	×	×
13	DIR	ディレクトリ表示	○	○	○
14	DLY	イベント検出点設定	○	○	×
15	DOS	OS制御に移す	○	○	○
16	EXT	コントロール・プログラム終了	○	×	×
17	HIS	コマンド・ヒストリ表示	○	○	○
18	HLP	ヘルプ	○	○	○
19	LOD	ロード	○	×	×
20	LST	出力デバイス・リダイレクト	○	○	○
21	MAT	演算	○	○	○
22	MEM	プログラム・メモリ操作	○	×	×
23	MOD	チャンネル2・モード設定	○	×	×
24	OUT	外部センス・クリップ・モード設定	○	○	×
25	PAS	バス・カウント設定	○	○	×
26	PGM	端末モード	○	×	×
27	RAM	データ・メモリ操作	○	×	×

表8-1 コマンド一覧 (2/2)

○：使用可, ×：使用不可

項番	コマンド名	機 能	システム動作モード		
			brk	emu	trc
28	REG	汎用レジスタ操作	○	×	×
29	RES	リセット	○	○	○
30	RUN	エミュレーション実行	○	×	×
31	SAV	セーブ	○	×	×
32	SET	エミュレーション・デバイス・モード切り替え	○	×	×
33	SPR	特殊レジスタ操作	○	×	×
34	STR	入力デバイス・リダイレクト	○	○	○
35	STP	リアルタイム・エミュレーション中止	×	○	○
	STP T	リアルタイム・トレース停止	×	×	○
36	STS	ディバグ対象デバイス選択	○	×	×
37	SYM	シンボル操作	○	×	×
38	SYS	システム再起動	○	×	×
39	TRD	トレース表示	○	○	×
40	TRF	トレース・データ検索条件設定	○	○	×
41	TRG	トレーサ起動	×	○	×
42	TRM	トレース・モード指定	○	○	×
43	TRP	トレース・ポインタ操作	○	○	×
44	TRX	クオリファイ・トレース条件指定	○	○	×
45	TRY	セクション・トレース条件指定	○	○	×
46	VRY	オブジェクト・ベリファイ	○	×	×

② サブコマンド

コマンド本体の機能を修飾します。サブコマンドには、サブコマンド1と2があります。サブコマンド1以下は、入力を省略することもできます。

(a) サブコマンド1

記述要素	意味	記述要素	意味
A	追加	M	転送, 移動
B	条件付き	N	条件なし
C	変更, チップ	R	読み取り
D	表示	S	サイズ・ステップ
E	テスト	T	ステップ
F	初期化, サーチ	U	ユーザ
G	サーチ	V	比較
H	ハードウェア	W	書き込み
I	IE-75001-R	X	交換
K	消去	Z	ゼロ・チェック
n	レジスタ番号, バンク番号		

(b) サブコマンド2

記述要素	意味	記述要素	意味
P	パラレル	S	シーケンシャル

③ オペランド

コマンドの機能を実現する際の、具体的な数値や文字列を指定します。

(a) 変数入力

ユーザが、記述要素に従って、その数値や式などを作り出して入力します。

記述要素	意味
addr	演算式入力可能なアドレス
addrx	"X" 記述可能なアドレス
addrb	ビット指定可能なアドレス
bit	1ビットの数値
command	コマンド名
count	10進数の数値
data	8ビットの数値
expression	式, 条件式
file	ファイル名
group	グループ名
module name	シンボルのモジュール名
option	オプション
partition	"," で区切られたアドレス範囲, または "X" 記述によるアドレス範囲
parameter list	スペースで区切られたパラメータの列
register	レジスタ名
string	"," で区切られた, 文字やデータの列 (data-string)
symbol	シンボル名
word	16ビットの数値

(b) 定数入力

ユーザが、各コマンドで指定された記述要素を、そのままの形式で入力します。次にその一例を示します。

記述要素	意味
BR ?	レジスタ名 (BRA1-BRA4, BRS1-BRS2)
REG	汎用レジスタの代表名
SPR	特殊レジスタの代表名
TRD	トレース表示
LST	リスト
CON	クローズ
F	First
M	Middle
L	Last
C	オブジェクト・コード
D	ディバグ環境
S	シンボル・ファイル

(c) オペランド・プロンプト

コマンドによっては、下記のようなキーを促すプロンプトが表示されます。その一例を以下に示します。詳細は各コマンド説明で記述します。

記述要素(プロンプト)	意味
C=	ステータス
MA=	データ・メモリのアドレス
MD=	データ・メモリのデータ
MRW=	データ・メモリのアクセス・ステータス
E=	外部センス・クリップのデータ
EXT=	外部センス・クリップのデータ
PA=	プログラム・メモリ・アドレス
PD=	プログラム・メモリ・データ
Pn=	I/Oポートnのデータ
V=	バリュー(値)

(4) オプション

オペランドを修飾する、付加情報です。記述要素そのままの形式で入力します。

記述要素	意味
R	レジスタの表示あり
B	バイト指定(データ・メモリ操作コマンド)
	ビット指定(トレース・コマンド)
N	ニブル指定
I	イベント・サイクル指定
M	マシン・サイクル指定

8.2 数値，シンボル，式の記述仕様

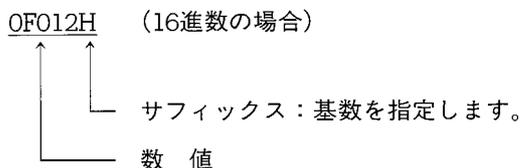
IE-75001-Rのコマンド入力時（主としてオペランド）に使用される「数値」，「シンボル」，「式」の記述規則について説明します。

8.2.1 数値の記述仕様

数値記述は，直接数値を記述します。数値と，その末尾にサフィックスを記述することによって，16進，10進，8進，2進数で表現することができます。

また数値の表現は，シンボル表現，あるいは式表現で置き換えることもできます。

(1) 数値の表現



- (a) 数値の最上位桁は，数字（0-9）を記述します。
- (b) 数値の前に符号（+，-）を付加することができます。この場合，符号の次の文字は数字（0-9）でなければなりません。
- (c) 数値の最上位桁が数字（0-9）でないときは，シンボルとして処理します。

(2) 数値の大きさ

数値は，コマンドごとに規定された大きさに応じて，次の4種類で扱うことができます。

16ビット，8ビット，4ビット，1ビット

③ 基 数

数値の最後に以下に示す4種類の符号を付加することで、基数を指定することができます。

H (16進数), T (10進数), Q (8進数), Y (2進数)

- (a) 通常、基数は各コマンドごとにその機能に応じて規定されています。
- (b) 基数を示す符号が省略された場合、規定された基数で数値は処理されます。
- (c) 規定外の基数を用いて数値を入力する場合、基数を示す符号を付加します。

表 8-2 各数値における最小値, 最大値の指定例

基 数	最小値	最 大 値			
	ビット共通	16ビット	8ビット	4ビット	2ビット
16進数	0H	0FFFFH	0FFH	0FH	1H
10進数	0T	65535T	255T	15T	1T
8進数	0Q	177777Q	377Q	17Q	1Q
2進数	0Y	1111111111111111Y	11111111Y	1111Y	1Y

備考 0および1は、基数を示す符号を入力しなくてもかまいません。

8.2.2 特殊数値の記述仕様 (X記述)

特殊数値は、数値表現の特別な場合を表します。「数値表現」「シンボル表現」および「式表現」は、ただ1つの数値を表します。これに対して特殊数値表現は、複数の数値の集まりを表します。

特殊数値を記述する場合、任意の数値を表現するXを使用します。Xは、16進数では0H-FHに、8進数では0Q-7Qに、2進数では0Yと1Yに、それぞれ対応します。ただし、10進数のX表現はできません。

特殊数値の記述仕様には、次の2種類があります。

数値範囲の記述

マスク・データの記述

(1) 数値範囲の記述

最下位から連続して上位桁へXを記述すると、Xで示された数値の最小から最大までの範囲をとることができます。

記述には次の点に注意してください。

- ・Xが連続していないときは、マスク・データの指定とみなします。
- ・最上位桁のXの前は「数字」でなければなりません。数字がないときは、その前に0 (零) を付けてください。0を付けないと、シンボルとして処理されます。

例 数値範囲の記述 (16ビット・データの場合)

```
OXXXXXH    → 0H-0FFFFH
OXXXXXXXXQ  → 0Q-177777Q
OXXXXXXXXXY → 0Y-11111111Y
```

(2) マスク・データの記述

数値の任意の桁位置にXを使用することによって、マスク・データを記述することができます。X桁のXの前は「数字」でなければなりません。数字がないときは、その前に0 (零) を付けてください。

例 マスク・データの記述

```
OX00H      → 0000H, 0100H, 0200H, 0300H, 0400H, 0500H, 0600H, 0700H,
             0800H, 0900H, 0A00H, 0B00H, 0C00H, 0D00H, 0E00H, 0F00H
01X1Q      → 0101Q, 0111Q, 0121Q, 0131Q, 0141Q, 0151Q, 0161Q, 0171Q
1X1010X1Y  → 10101001Y, 1101001Y, 10101011Y, 11101011Y
```

8.2.3 シンボルの記述仕様

シンボルには、ローカル・シンボルとパブリック・シンボルなどがあります。通常これらのシンボルは、シンボル・テーブル・ファイルをロードすることで定義されます。

以下にシンボルについての規約、使用方法、シンボル数などを説明します。

(1) シンボル名

シンボル名を構成する文字は、次に示す文字のみとなります。

A-Z, a-z, @, ?, _, 0-9

使用にあたっては次の点に注意してください。

- 先頭の文字は、数字0-9以外でなければなりません。
- 英小文字 (a-z) は、英大文字 (A-Z) として扱います。
- シンボル名の有効文字数は、最大8文字となります。
- 8文字以上の文字が記述された場合、先頭から8文字が有効となります。

(2) シンボル値

シンボル値は、16ビット・データとして扱われます。

(3) モジュール名

シンボルを記述する場合に付記するモジュール名の記述仕様は、次の3種類のシンボルで異なります。

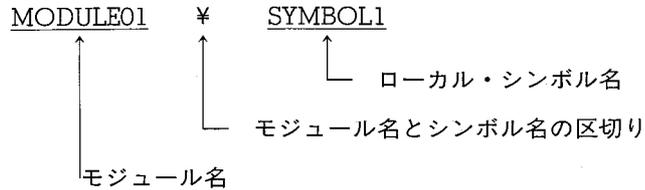
(a) パブリック・シンボルのモジュール名記述仕様

パブリック・シンボルを記述する場合、シンボル名のみ記述します。モジュール名 (PUBLIC ¥) を記述する必要はありません。

備考 “¥” は、IBM PCシリーズをホストとして使用した場合、“\” となります。

(b) ローカル・シンボルのモジュール名記述仕様

ローカル・シンボルを記述する場合、次に示すようにシンボル名の前にモジュール名を付加します。



カレント・モジュールが指定 (SYM Mコマンド) されている場合、モジュール名を省略してシンボル名のみ記述します。

備考 パブリック・モジュールに同一シンボル名が存在する場合、パブリック・シンボルが選択されます。詳細は**8.4.37 シンボル名操作コマンド (SYM)** を参照してください。

(c) IEシンボルのモジュール名記述仕様

IEシンボルを記述する場合、シンボル名のみ記述します。モジュール名 (IESYMBOL ¥) を記述する必要はありません。

(4) 最大シンボル数

IE-75001-Rで記述 (登録) 可能な最大シンボル数は、次のとおりです。

(a) シンボル・テーブル・ファイル

シンボル・テーブル・ファイルのシンボルは、約7000シンボルまで登録できます。

(b) IEシンボル

IEシンボルは、約7000シンボルまで登録できます。

8.2.4 式の記述仕様

式の表現は、数値と数値、シンボルとシンボル、あるいは数値とシンボルを演算子で結合した形で行います。また式表現は、数値表現のかわりに使用できます。

(1) 演算子と式の記述

式には、次の演算子が使用できます。

演算子	優先順位	備考
()	↑ 高	最大32レベルまで可能
* /	優先順位	
+ -		
AND		
OR XOR	↓ 低	

使用にあたっては次の点に注意してください。

- ・カッコの使用は、最大32レベルまでとなります。
- ・「数値、シンボル」と「AND、OR、XOR」の各演算子を記述する場合には、1つ以上の「スペース」が、演算子の前後に必要です。
- ・「数値、シンボル」と「(,), *, /, +, -」の各演算子を記述する場合には、スペースを挿入する必要はありません。

(2) 演算処理

演算は、すべて16ビットの整数で行われます。演算の中間結果、最終結果が16ビット以上になった場合には、17ビット目より上位は切り捨てます。

演算は、「数値、シンボル」に対してのみ有効であり、「予約語あるいは特殊数値」に対しての演算は、エラーとなります。

8.3 特殊キーの使用法

IE-75001-Rで使用される特殊キーを次に示します。

(1) ダブル・タッチ入力のキー

キー入力	記述形式	機能
CTRL +A	^ A	挿入モード・トグル・スイッチ
CTRL +C	^ C	コントロール・プログラム強制終了
CTRL +H	^ H	一文字削除
CTRL +I	^ I	スペースと同じ
CTRL +J	^ J	行入力の終了
CTRL +K	^ K	STRコマンドの強制終了
CTRL +L	^ L	STRコマンドの一時停止／再開
CTRL +M	^ M	行入力の終了
CTRL +O	^ O	COMコマンドの出力スイッチ
CTRL +P	^ P	LSTコマンドの出力スイッチ
CTRL +Q	^ Q	^ Sによる一時停止の解除
CTRL +S	^ S	一時停止
CTRL +X	^ X	一行削除

(2) シングル・タッチ入力のキー

キー入力	記述形式	機能
BS	<BS>	一文字削除
	<cr>	行入力の終了
DEL		一文字削除
ESC	<ESC>	コマンド実行の終了
TAB	<TAB>	スペースと同じ
	;	コメントの開始
	!	履歴の呼び出し
	←	カーソル左移動
	→	カーソル右移動

8.4 コマンド説明

8.4.1 アセンブル・コマンド (ASM)

入力形式	
形式1	ASM [Δ addr]
基数	addr : H
brk : n >	<input type="radio"/>
emu : n >	<input type="checkbox"/>
trc : n >	<input type="checkbox"/>

【機能】

ASMコマンドは、指定されたプログラム・アドレスから、75Xシリーズのニモニック・コードで入力されたプログラムをオブジェクト・コードに変換（アセンブル）して、プログラム・メモリに書き込みます。

すなわち、煩わしい機械語を使用せずに、ニモニック・コードを用いて、プログラムの変更や修正を容易に行うことができます。

なお、初期設定時は、addrの値は0000Hです。

(a) アセンブラの動作

- ① 指定されたプログラム・アドレスからオブジェクト・コードを取り出し、75Xシリーズのニモニック・コードに変換（ディスアセンブル）して表示します。
- ② プロンプト（“=”）を表示して、ニモニック・コードによるソース・プログラムの入力を待ちます。
- ③ 入力されたニモニック・コードを、オブジェクト・コードに変換（アセンブル）してプログラム・メモリに書き込みます。同時に次の行に、変換したオブジェクト・コードを表示します。

(b) アセンブラへの入力データ

アセンブラに対する入力は、次表に示す2種類のアセンブラ制御命令と、ディバグ対象となるデバイスのニモニック・コードです。

命 令	機 能
ORG△addr	アセンブルするアドレスを, addrで指定されたアドレスに変更します。
END	アセンブル・コマンドの処理を終了します。
ニモニック・コード	<p>ディバグの対象となるデバイスのニモニック・コードを入力できます。</p> <p>① デバイスの種別は, STSコマンドで指定されたものです。</p> <p>② I/O (入出力) 命令を使用する場合は, I/Oアドレスを, I/O予約語で指定しておくことができます。</p>

A

(c) アセンブラのエラー処理

入力されたニモニック・コードにエラーがあった場合には, 次のメッセージを表示し再入力待ちとなります。

メッセージ: “***** Error!! *****”

【指定方法】

オペランド

addr : アセンブルを開始する, プログラム・メモリのアドレスを指定します。

アドレス指定範囲: 0-0FFFFH

注意 アドレスの範囲は対象デバイスによって異なります。

【使用規約】

本アセンブラは, オブジェクト・コードのリンクス・チェックを行いません。したがって, 1バイト命令を2バイト命令に変更するような場合には, 次命令の頭部1バイトが塗り潰されますので, 残りバイトをNOP命令で埋めるなどの処置が必要です。

【使用例】

プログラム・アドレスの1000H番地から、ニモニック・コードを用いてプログラムの変更を行う例です。

```
brk:0>ASM 1000H①  
  
Addr Code Label Mnem. Operand  
1000 60 NOP =MOV X, #0②  
1000 9A 09  
1002 60 NOP =END④  
  
brk:0>■
```

- ① 1000H番地からアセンブルを開始する指定
- ② 1000H番地の内容を表示し、NOPをMOV X, #0に変更
- ③ 変更内容（オブジェクト・コード）の表示
- ④ 次命令、1002H番地の内容を表示し、アセンブルを終了する指定

8.4.2 データ・メモリ・バンク設定コマンド (BNK)

入力形式	
形式1	BNK [Δn] ($0 \leq n \leq 0FH$)
基数	n : H

brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="checkbox"/>	×
-----------	-----------------------	-----------	-----------------------	-----------	--------------------------	---

【機能】

BNKコマンドは、BNKの内容を、指定されたバンク番号に変更します。BNKは、逆アセンブル時に、オペランドのシンボル変換を行う場合のデータ・メモリ・アドレスの上位4ビットを指定するものです。

なお、初期設定時は、nは00Hに設定されています。

(a) BNKの参照

BNKの内容は、アセンブル・コマンド (ASM) およびディスアセンブル・コマンド (DAS) によってのみ参照され、リアルタイム実行には影響を与えません。

(b) データ・メモリのアクセス・アドレス決定

75Xシリーズでは、データ・メモリをアクセスする際のアドレスを、次に示す2つの要素を加えた値 (12ビット) で決定します。

- 命令のオペランドの値 (下位8ビット)
- 命令実行時のメモリ・バンク・レジスタ (MBS) の値 (上位4ビット)

(c) データ・メモリ・バンク・アドレスの表示

オペランドが省略された場合には、現在設定されているデータ・メモリ・バンク・アドレスを表示します。

(d) BNKの値

エミュレーション実行後、BNKの値は次のように設定されます。

- MBE=0のとき→BNK=0
- MBE=1のとき→BNK=MBSの値

【指定方法】

オペランド

n : データ・メモリのバンク・アドレスを、16進数値で指定します。

指定範囲： $0 \leq n \leq 0FH$

省略時 : 現在の、データ・メモリのバンク・アドレスを表示します。

【使用例】

データ・メモリのバンク・アドレスを、0から1に変更する例です。

brk : 0 > <u>BNK</u> <input type="checkbox"/>	①
00	②
brk : 0 > <u>BNK 1</u> <input type="checkbox"/>	③

- ① 現在のバンク・アドレスを確認
- ② 現在のバンク・アドレスの表示
- ③ バンク・アドレス0を1に変更

8.4.3 データ・メモリ・イベント条件設定コマンド (BRA1-BRA4)

入力形式 形式1 $BRA[\Delta n] \left[\Delta MA = \begin{cases} \text{addr} \\ \text{partition} \end{cases} \right] \left[\Delta V = \text{data1} \right] \left[\Delta C = \text{status} \right] \left[\Delta E = \text{data2} \right]$ <div style="margin-left: 100px;"> \uparrow $(1 \leq n \leq 4)$ </div>						
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10%; padding: 2px;">基数</td> <td style="padding: 2px;">addr : H</td> <td style="padding: 2px;">partition : H</td> <td style="padding: 2px;">data1 : Y</td> <td style="padding: 2px;">data2 : Y</td> </tr> </table>	基数	addr : H	partition : H	data1 : Y	data2 : Y	
基数	addr : H	partition : H	data1 : Y	data2 : Y		
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 25%; padding: 2px;">brk : n ></td> <td style="width: 5%; padding: 2px; text-align: center;">○</td> <td style="width: 25%; padding: 2px;">emu : n ></td> <td style="width: 5%; padding: 2px; text-align: center;">○</td> <td style="width: 25%; padding: 2px;">trc : n ></td> <td style="width: 5%; padding: 2px; text-align: center;">×</td> </tr> </table>	brk : n >	○	emu : n >	○	trc : n >	×
brk : n >	○	emu : n >	○	trc : n >	×	

【機能】

BRAコマンドは、各種イベント条件を設定するコマンドです。たとえば、プログラムがBRAコマンドで指定された条件でデータ・メモリをアクセスしたとき、ブレークを発生させたり、トレース・データを出力させたりすることができます。

(a) イベント条件の設定個数

イベント条件の設定は、1コマンドに対し1条件で、最高4個（データ・メモリ・イベント条件レジスタの個数分）まで指定することができます。

また、本コマンドで設定された条件は、イベント・モード設定コマンド (BRM) によって選定された時点で最終的に有効となります。

【指定方法】

サブコマンド

n : データ・メモリ・イベント条件レジスタの番号を指定します。
 レジスタ番号指定範囲：1-4

オペランド

$MA = \begin{matrix} \text{addr}X \\ \text{partition} \end{matrix}$: データ・メモリのアドレスを指定します。なお、アドレスの範囲は対象デバイスによって異なります。アドレスの直前に “/” をつけることにより、指定したアドレス以外をアクセスすると、ブレークを起こすように設定します。BRA1, BRA2が指定されたときには、アドレス・マスク指定およびパーティション指定ができます。

BRA3, BRA4が指定されたときには、アドレス・マスクのみ指定ができます。

アドレス指定範囲：0-OFFFH

マスク指定範囲 : 0-OXXXH

省略時 : 直前に設定された値がとられます。

初期値 : 0XXXXH (16進数マスク指定)

V=data1 : データの値を、次の形式で指定します。なお、データ指定の前に“/”をつけることにより、指定したデータ値以外をアクセスすると、ブレークを起すように設定します。

データ値指定範囲 : 0-OFFH

マスク指定範囲 : 0-OXXXXXXXXY

省略時 : 直前に設定された値がとられます。

初期値 : OXXXXXXXXY
(2進数マスク指定)

備考 イベント条件として有効となるのは、「status」で指定された“ビット数分”です。

例 “R4”と指定すると、下位4ビットのみが有効となります。

C=status : データ・メモリへのアクセス条件を指定します。

指定形式	アクセスの条件	
	有効ビット数	対象I/Oモード
R4	4ビット	リード, リード・モディファイ・ライト
W4	4ビット	ライト, リード・モディファイ・ライト
RW4	4ビット	リード, ライト, リード・モディファイ・ライト
R8	8ビット	リード, リード・モディファイ・ライト
W8	8ビット	ライト, リード・モディファイ・ライト
RW8	8ビット	リード, ライト, リード・モディファイ・ライト
SP	スタック・ポインタで指されたアドレスのデータ	

省略時 : 直前に設定された値がとられます。

初期値 : RW4

E=data2 : 外部センス・クリップからの入力データ値を、次の形式で指定します。なお、データ指定の前に“/”をつけることにより、指定したデータ値以外を入力すると、ブレークを起すように設定します。

データ値指定範囲 : 0-OFFH

マスク指定範囲 : 0-OXXXXXXXXY

省略時 : 直前に設定された値がとられます。

初期値 : OXXXXXXXXY (2進数マスク指定)

【使用規約】

コマンドで設定した条件でイベントを発生させるには、あらかじめイベント・モード設定コマンド (BRM) によって、データ・メモリ・イベント・レジスタ (BRA1-BRA4) を選定してなければなりません。詳細は、**8.4.5 イベント・モード設定コマンド (BRM)**を参照してください。

- 注意1.** スキップ命令実行後の次の命令にブレーク条件を設定した場合、そのブレーク条件が満たされたとしても、ブレークはしません (無視されます)。

例 Loop: MOV A, @HL

.

.

.

INCS A

BR Loop

MOV X, A ←このポイントにブレーク条件を設定してもブレークしません。

MOV XA, @HL ←この例の場合には、このポイントにブレーク条件を設定してください。

2. ブレーク付きリアルタイム実行 (RUN B) の場合、次のブレーク条件により数命令実行後、ブレークします。
- ①R4, R8をブレーク条件とした場合、ブレーク条件成立後、次の命令を実行し、ブレークします。
 - ②W4, W8をブレーク条件とした場合、ブレーク条件成立後、次の命令が1バイトの場合、2命令実行後ブレークします。同様に次の命令が2バイトまたは3バイトの場合、1命令実行後ブレークします。

【使用例】

(a) イベント条件の設定

データ・メモリ・アドレスの、0100H番地から0103H番地がアクセスされたときに、イベントの検出を行うよう、BRA1-BRA4に条件設定を行う例です。

brk : 0>BRA 1 MA=0100 V=00001111Y C=W8	①
brk : 0>BRA 2 MA=0101 V=0000XXXXY C=R8	②
brk : 0>BRA 3 MA=0102 V=0AH C=W8	③
brk : 0>BRA 4 MA=0103 V=010H C=W8	④
brk : 0>BRM BRA1 BRA2 BRA3 BRA4	⑤
brk : 0>RUN B	⑥

① イベント検出条件として、次の内容をBRA1に設定

データ・メモリ・アドレス：0100H番地（16進）

データ・メモリ内容（値）：00001111Y（2進）

メモリ・アクセス条件：有効ビット=8ビット

I/Oモード=ライト，リード・モディファイ・ライト

② イベント検出条件として、次の内容をBRA2に設定

データ・メモリ・アドレス：0101H番地（16進）

データ・メモリ内容（値）：0000XXXXY（2進マスク指定）

メモリ・アクセス条件：有効ビット=8ビット

I/Oモード=リード，リード・モディファイ・ライト

③ イベント検出条件として、次の内容をBRA3に設定

データ・メモリ・アドレス：0102H番地（16進）

データ・メモリ内容（値）：0AH（16進）

メモリ・アクセス条件：有効ビット=8ビット

I/Oモード=ライト，リード・モディファイ・ライト

- ④ イベント検出条件として、次の内容をBRA4に設定

データ・メモリ・アドレス：0103H番地（16進）

データ・メモリ内容（値）：010H（16進）

メモリ・アクセス条件：有効ビット=8ビット

I/Oモード=ライト，リード・モディファイ・ライト

- ⑤ BRA1-BRA4に設定したイベント条件を，イベント・モード・レジスタ（BRM）に設定（この指定で最終的にイベント条件が有効となる）
- ⑥ 現在のプログラム・アドレスから，エミュレーションを開始するよう指定

(b) アドレスNOT指定

```
brk:0>BRA 1 MA=/100H V=0XXXXXXXXY C=RW4 E=0XXXXXXXXY
brk:0>
```

①

- ① アドレス100番地以外に対する，4ビット・アクセスでブレークの設定をします。

(c) データ値NOT指定

```
brk:0>BRA 1 MA=100H V=/0H C=RW4 E=0XXXXXXXXY
brk:0>
```

①

- ① アドレス100番地に，データ値0以外に対する4ビット・アクセスでブレークの設定をします。

(d) 外部センス・クリップの値のNOT指定

```
brk:0>BRA 1 MA=100H V=0H C=RW4 E=/OFFH
brk:0>
```

①

- ① アドレス100番地に，データ値0に対する4ビット・アクセスで，外部センス・クリップの値がOFFH以外でブレークの設定をします。

(e) サブコマンド以下を省略したとき

サブコマンド以下を省略すると、BRA1-BRA4の設定内容が表示されます。

```
brk : 0>BA⏎ ①
  
BRA1 : MA=OXXXXH      V=OXXXXXXXXXY  C=R8  E=OXXXXXXXXXY  ②
BRA2 : MA=OXXXXH      V=OXXXXXXXXXY  C=R8  E=OXXXXXXXXXY
BRA3 : MA=OXXXXH      V=OXXXXXXXXXY  C=R8  E=OXXXXXXXXXY
BRA4 : MA=OXXXXH      V=OXXXXXXXXXY  C=R8  E=OXXXXXXXXXY
```

- ① サブコマンド以下を省略指定
- ② BRA1-BRA4の内容を表示

(f) オペランド以下を省略したとき

オペランド以下を省略するとメニュー・モードとなり、対話形式でイベント条件の設定を行うことができます。

```
brk : 0>BRA 1⏎ ①
  
MA 0H, 0FFFH=OXXXXH⏎ ②
  
V 10001Y=⏎ ③
  
4bit D-mem Read      (R4) ④
4bit D-mem Write     (W4)
8bit D-mem Read      (R8)
8bit D-mem Write     (W8)
4bit D-mem Read/Write (RW4)
8bit D-mem Read/Write (RW8)
Stack Pointer Access (SP)
  
C W8=R8⏎ ⑤
  
E OXXXXXXXXXY=⏎ ⑥
  
brk : 0>
```

- ① オペランド以下を省略指定
- ② メモリ・アドレスをマスク指定に変更
- ③ データ値の変更なし
- ④ ステータスのガイダンス表示
- ⑤ ステータス：W8をR8に変更
- ⑥ 外部センス・クリップ・データ値の変更なし

8.4.4 イベント情報表示コマンド (BRK)

入力形式
形式1 BRK

brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input checked="" type="checkbox"/>
-----------	-----------------------	-----------	-----------------------	-----------	-------------------------------------

【機能】

BRKコマンドは、リアルタイム・エミュレーション実行における、直前の「ブレーク要因」や「トレース中断要因」および「イベント条件レジスタ」の内容を表示します。

(a) 動作モードと表示内容

システムの動作モードによって、表示されるイベント情報の内容が異なります。

エミュレーション・モード：トレース中断要因を表示します。

ブレーク・モード ：ブレーク要因を表示します。

(b) 表示対象レジスタ

次のイベント条件レジスタの内容を表示します。

BRA1, BRA2, BRA3, BRA4, BRS1, BRS2

(c) ガード・ブレーク

IE-75001-Rには、対象デバイスが持つ機能以外のところを実行した場合、強制的にブレークする機能を持っています。

- ① GDM：ガード・データ・メモリ
指定領域以外のデータ・メモリをアクセスした場合
- ② GDSP：ガード・スタック・ポインタ
指定領域以外のスタック・ポインタをアクセスした場合
- ③ GDR：ガード・レジスタ
指定領域以外のレジスタをアクセスした場合
- ④ GDIO：ガードI/O
指定領域以外のモード・レジスタI/O領域をアクセスした場合

【指定方法】

コマンド本体のみ入力します。

【使用例】

(a) ブレーク要因の表示

ブレーク・モードにおいて、BRA2に設定されたイベント条件によって、ブレークしたときの要因を表示した例です。

```

brk : 0>BRK①

Event Register Information
(ena) BRA1 : MA = 123, 346H          V = 0XXXXXXXXY C = RW8 E = 0XXXXXXXXY ②
(dis) BRA2 : MA = LABEL001, LABEL002 V = 1Y          C = R4  E = 11Y
(dis) BRA3 : MA = 0XXXX1XXXXXXXXY    V = 1XXY        C = W8  E = 1XXY
(dis) BRA4 : MA = 0XXXXH             V = 0XXXXXXXXY C = RW4 E = 0XXXXXXXXY
(dis) BRS1 : Sequential              ③
      PA = LABEL003                  E = 0XXXXXXXXY
      PA = 0XXXX1XXXXXXXXXXXXX1      E = 0XXXXXXXXY
      PA = 0XXXXH                    E = 11Y
(dis) BRS2 : PA = 0XXXXH             E = 0XXXXXXXXY ④

BREAK CAUSED BY BRA1 ⑤
    
```

- ① イベント情報表示の指定
- ② データ・メモリ・イベント・レジスタ (BRA1-BRA4) 内容の表示
- ③ プログラム・メモリ・イベント・レジスタ (BRS1) 内容の表示

S (Sequential) 指定のとき : 最大4ポイントの設定条件を表示

P (Parallel) 指定のとき : 最大7ポイントの設定条件を表示

- ④ プログラム・メモリ・イベント・レジスタ (BRS2) 内容の表示
イベント条件の表示は、1ポイントのみです。
- ⑤ ブレーク要因の表示です。イベント条件レジスタ名、ブレーク要因を表示

レジスタ : BRA1, BRA2, BRA3, BRA4, BRS1, BRS2

要因 : GDM, GDSP, GDR, GDIO, STP

各イベント条件レジスタがイベント検出の条件として、有効であるか否か (BRMに設定されているか否か) を表示します。

(ena) : 検出条件が有効 (BRMに設定されている)

(dis) : 検出条件が無効 (BRMに設定されていない)

(b) トレース中断要因の表示

エミュレーション・モードにおいて、BRS1に設定されたイベント条件によって、ブレイクしたときの要因を表示した例です。

```

emu : 0>BRK☑ ①

Event Register Information
(dis) BRA1 : MA = 123, 346H          V = 0XXXXXXXXY C = RW8 E = 0XXXXXXXXY ②
(dis) BRA2 : MA = LABEL001, LABEL002 V = 1Y          C = R4  E = 11Y
(dis) BRA3 : MA = 0XXXX1XXXXXXXXY    V = 1XXY        C = W8  E = 1XXY
(dis) BRA4 : MA = 0XXXXH             V = 0XXXXXXXXY C = RW4 E = 0XXXXXXXXY
(ena) BRS1 : Sequential ③
      PA = LABEL003             E = 0XXXXXXXXY
      PA = 0XXXX1XXXXXXXXXXXXY E = 0XXXXXXXXY
      PA = 0XXXXH              E = 11Y
(dis) BRS2 : PA = 0XXXXH          E = 0XXXXXXXXY ④

TRACER STOPPED BY BRS1 ⑤
    
```

- ① イベント情報表示の指定
- ② データ・メモリ・イベント・レジスタ (BRA1-BRA4) 内容の表示
- ③ プログラム・メモリ・イベント・レジスタ (BRS1) 内容の表示

S (Sequential) 指定のとき : 最大4ポイントの設定条件を表示します。

P (Parallel) 指定のとき : 最大7ポイントの設定条件を表示します。

- ④ プログラム・メモリ・イベント・レジスタ (BRS2) 内容の表示
イベント条件の表示は、1ポイントのみです。
- ⑤ トレース中断要因の表示です。イベント条件レジスタ名、トレース中断要因を表示

レジスタ : BRA1, BRA2, BRA3, BRA4, BRS1, BRS2

要因 : GDM, GDSP, GDR, GDIO, STPT

各イベント条件レジスタがイベント検出の条件として、有効であるか否か (BRMに設定されているか否か) を表示します。

(ena) : 検出条件が有効 (BRMに設定されている)

(dis) : 検出条件が無効 (BRMに設定されていない)

8.4.5 イベント・モード設定コマンド (BRM)

入力形式

形式1 BRM [Δ BR?] [Δ BR?]

brk:n>	<input type="radio"/>	emu:n>	<input type="radio"/>	trc:n>	<input type="checkbox"/>
--------	-----------------------	--------	-----------------------	--------	--------------------------

【機能】

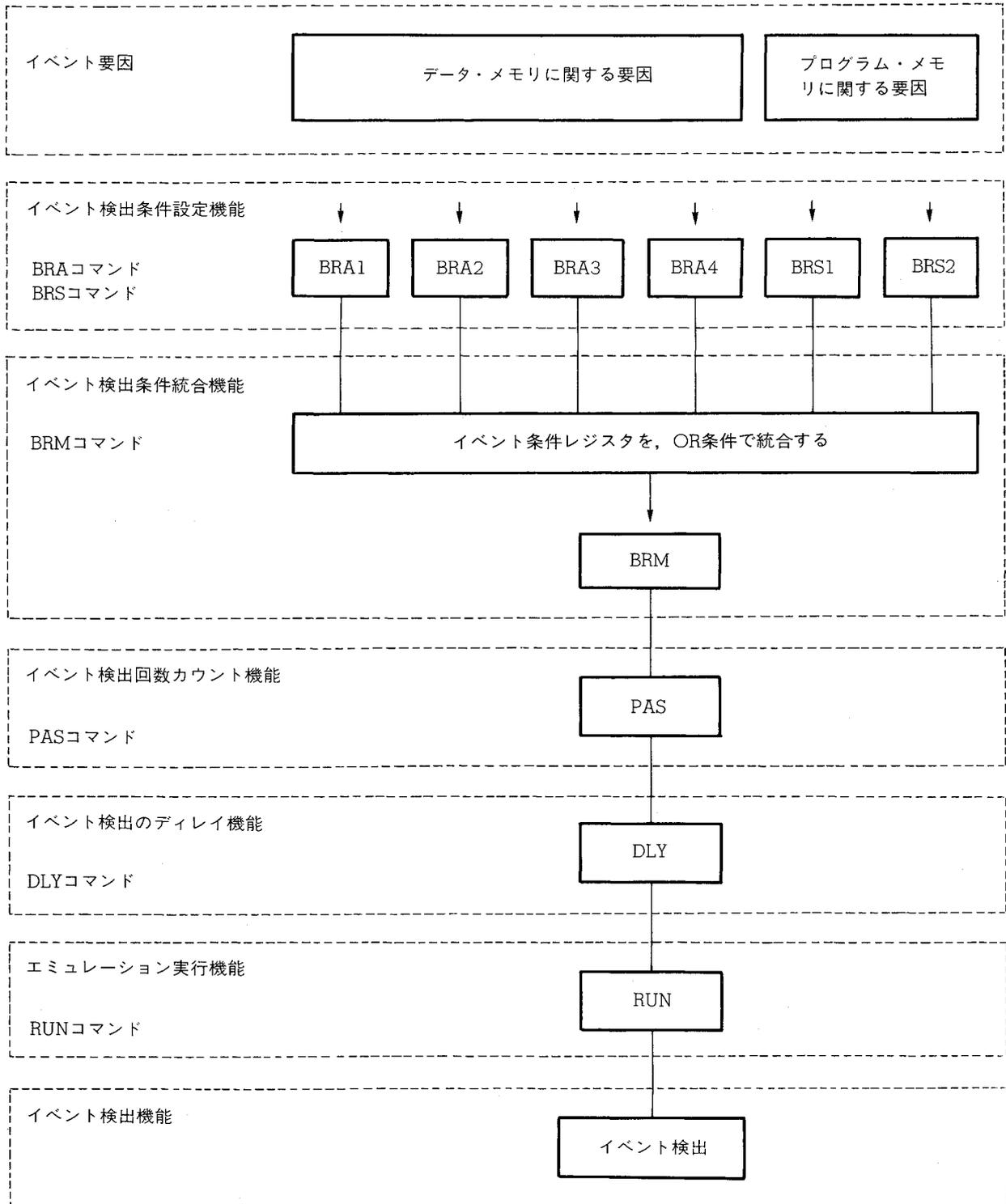
BRMコマンドは、イベント条件レジスタに設定されたイベント条件を、最終的なイベント条件としてイベント・モード・レジスタに統合します。

すなわち、このコマンドで指定されたイベント条件レジスタの内容が、トレースやブレークのイベント検出条件として有効となります。

なお、初期設定時は、BRS1が指定されています。

イベント検出条件の設定からイベント検出までの、一般的な操作手順と概念を示すと、次の図のようになります。

図 8-1 イベント条件の設定と検出手順



- (a) オペランドが省略された場合には、現在イベント・モード・レジスタ (BRM) に選定されているイベント条件レジスタ名を表示します。
- (b) OUTコマンドでBRA3, BRA4が指定されている場合、BRMコマンドでBRA3, BRA4を指定することはできません。

【指定方法】

オペランド

BR? : 次のイベント条件レジスタのうち、最大6個までを指定します。

BRA1, BRA2, BRA3, BRA4, BRS1, BRS2, OFF

備考 “OFF” が指定された場合は、イベント検出条件は無効となりリアルタイム実行には影響を与えません。

初期値 : BRS1

省略時 : オペランドが省略された場合は、現在イベント・モード・レジスタ (BRM) に設定されているイベント条件レジスタ名を表示します。

【使用例】

イベント条件レジスタの設定

データ・メモリ・イベント・レジスタ (BRA1-BRA3) と、プログラム・メモリ・イベント・レジスタ (BRS1) に設定された4個のイベント条件を、イベント・モード・レジスタ (BRM) に選定した例です。

```
brk:0>BRA 1 MA=0100 V=000001111Y C=W8
```

①

```
brk:0>BRA 2 MA=0101 V=00000XXXXY C=R8
```

②

```
brk:0>BRA 3 MA=0102 V=0AH C=W8
```

③

```
brk:0>BRS 1 P 02000H 02500H,XXXX0000Y
```

④

```
brk:0>BRM BRA1 BRA2 BRA3 BRS1
```

⑤

```
brk:0>BRM
```

⑥

```
BRA1 BRA2 BRA3 BRS1
```

⑦

```
brk:0>RUN B
```

⑧

- ① BRA1にイベント条件を設定
- ② BRA2にイベント条件を設定
- ③ BRA3にイベント条件を設定
- ④ BRS1にイベント条件を設定
- ⑤ BRA1-BRA3, BRS1に設定されたすべてのイベント条件を, BRMに選定
- ⑥ 選定内容の表示指定 (確認)
- ⑦ 選定内容の表示
- ⑧ エミュレーションの実行指定

8.4.6 プログラム・メモリ・イベント条件設定コマンド (BRS)

入力形式	
形式1	BRS [$\Delta 1$] $\left[\begin{array}{l} \Delta S \\ \Delta P \end{array} \right]$ [Δaddrx [, data]] [Δaddrx [, data]]
形式2	BRS $\Delta 2$ [Δaddrx [, data]]
基数	addrx : H data : Y

【機能概要】

BRSコマンドは、指定されたプログラム・メモリ・アドレスの命令を実行したとき、あるいは外部センス・クリップからデータの入力があったとき、これをイベントの発生とするための条件を設定します。

たとえばプログラムが、本コマンドで指定されたアドレスを通過したときに、ブレークを発生させたりトレース・データを出力させたりすることができます。

【使用規約】

- 以下に示す命令にブレーク条件を設定した場合、ブレーク条件の次の命令を実行したあとブレークします。また、このとき以下に示す命令が連続している場合、以下に示す命令以外の命令を実行したあとブレークします。

```
SET1   OFBX   (EI OFBX)
CLR1   OFBX   (DI OFBX)
SEL    OMBX
```

```
例 ①CLR1  OFB2.3  ←ブレーク条件設定箇所
    MOV   A, #0H  ←実際にブレークする箇所

    ②SET1  OFB4.3  ←ブレーク条件設定箇所
    SET1  OFB5.0
    SEL   OMB0
    MOV   A, #0H  ←実際にブレークする箇所
```

- スキップされる命令にブレーク条件を設定した場合、その命令がスキップされ実行されなくてもブレークします。

(1) 多重イベント設定

形式1	BRS [$\Delta 1$] <table border="1" style="display: inline-table; vertical-align: middle; margin: 0 10px;"><tr><td style="padding: 2px 5px;">ΔS</td></tr><tr><td style="padding: 2px 5px;">ΔP</td></tr></table> [Δaddrx [, data]]... ... [Δaddrx [, data]]	ΔS	ΔP			
ΔS						
ΔP						
基数	addrx : H data : Y					
brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="checkbox"/>	×

【機能】

サブコマンド1で指定された番号を持つ「イベント条件レジスタ」に、オペランドで指定された「プログラム・メモリ・アドレス」、「外部センス・クリップからの入力データ値」をイベント検出の条件として設定します。

イベントの検出は、サブコマンド2で指定された順序で行います。またイベント条件の設定は、シーケンシャル・イベントの場合、最大4個、パラレル・イベントの場合、最大7個の多重設定ができます。

【指定方法】

(a) サブコマンド1

プログラム・メモリ・イベント条件レジスタの番号を指定します。“1”を指定します。

(b) サブコマンド2

イベント検出の検出順序を指定します。

S : シーケンシャル・イベント検出指定

オペランドで指定された順序でプログラムが実行されたとき、イベントの検出となります。

P : パラレル・イベント検出指定

プログラムが、オペランドで指定されたいずれかの条件を実行したときに、イベントの検出となります。

(c) オペランド

イベント検出のポイントを指定します。指定できる個数は、シーケンシャル検出とパラレル検出とは異なります。

addrx : イベント検出を行いたいプログラムのアドレスを指定します。

アドレス指定範囲 : 0-OFFFFH

マスク指定範囲 : 0-OXXXXXH

指定個数 : 最大4ポイント

(シーケンシャル指定時)

最大7ポイント

(パラレル指定時)

data : 外部センス・クリップからの入力データ値を指定します。

データ値指定範囲 : 0-OFFH

マスク指定範囲 : 0-OXXXXXXXXY

指定個数 : 最大4ポイント

(シーケンシャル指定時)

最大7ポイント

(パラレル指定時)

(2) 単一イベント設定

形式2		BRS [$\Delta 2$] [Δ addrx [, data]]	
基数	addrx : H	data : Y	
brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>
		trc : n >	<input type="checkbox"/>

【機能】

サブコマンド1で指定された番号の「イベント条件レジスタ」に、オペランドで指定された「プログラム・メモリ・アドレス」、「外部センス・クリップからの入力データ値」をイベントの検出の条件として設定します。

またイベント条件の設定は、単一設定です。本形式は、セクション・トレースを行う場合の「開始アドレス」、「終了アドレス」の設定に使用されます。

【指定方法】

(a) サブコマンド

プログラム・メモリ・イベント条件レジスタの番号を指定します。

"2" を指定します。

(b) オペランド

イベント検出のポイントを指定します。指定できる個数は1個です。

addrx : イベント検出を行いたいプログラムのアドレスを指定します。

アドレス指定範囲 : 0-OFFFFH

マスク指定範囲 : 0-OXXXXXY

指定個数 : 1ポイント

data : 外部センス・クリップからの入力データ値を指定します。

アドレス指定範囲 : 0-OFFH

マスク指定範囲 : 0-OXXXXXXXXXY

指定個数 : 1ポイント

【使用例】

(a) 多重イベントの指定

プログラム・メモリ・イベント・レジスタ (BRS1) に、ブレーク条件を多重イベント指定した例です。

```
brk : 0> BRS 1 P 2000H 2500H, 0XXXX0000Y ①
brk : 0> BRM BRS1 ②
brk : 0> BRS ③
BRS1 : Parallel
        PA=2000H           E=XXXXXXXXXY
        PA=2500H           E=XXXX0000Y
BRS2 : PA=0XXXXH          E=XXXXXXXXXY
brk : 0> RUN B ④
```

- ① BRS1にイベント条件を設定
- ② BRS1に設定されたイベント条件をBRMに選定 (検出条件の確定)
- ③ 選定内容の表示指定 (確認)
- ④ エミュレーションの実行指定

(b) 単一イベントの指定

プログラム・メモリ・イベント・レジスタ (BRS1, BRS2) に、セクション・トレースの開始ポイントと終了ポイントを設定した例です。

brk : 0>BRS 1 S 2000H	①
brk : 0>BRS 2 3000H	②
brk : 0>TRY E BRS1	③
brk : 0>TRY D BRS2	④
brk : 0>TRM TRY	⑤
brk : 0>RUN N	⑥

- ① BRS1に開始アドレスを設定
- ② BRS2に終了アドレスを設定
- ③ BRS1をトレース有効点（開始ポイント）に指定
- ④ BRS2をトレース無効点（終了ポイント）に指定
- ⑤ セクション・トレースを指定
- ⑥ エミュレーションの実行指定

(c) オペランド以下を省略したとき

オペランド以下を省略すると、メニュー・モードとなり、対話形式でイベント条件の設定を行うことができます。

brk : 0>BRS 1	①
PARALLEL (P)/SEQUENTIAL (S) P=P	②
PA, EXT OXXXX, OXXXXXXXXY =300, 11100011	③
PA, EXT OXXXX, OXXXXXXXXY =400	④
PA, EXT OXXXX, OXXXXXXXXY =.	④
brk : 0>	

- ① オペランド以下を省略指定します。
- ② パラレル・イベント検出を選択します。
- ③ ブレークポイントを設定します。
- ④ ドット (.) 入力で終了します。

注意 リターンのみへの入力は、前の設定値が有効となります。Cで、そのラインをクリア
 します。

★

```
brk : 0>BRS 1①
PARALLEL (P)/SEQUENTIAL (S) P=S②
PA, EXT 0300, 011100011Y =3FF②
PA, EXT 0400, 0XXXXXXXXY =400, 00001111②
PA, EXT 0XXXX, 0XXXXXXXXY =410②
PA, EXT 0XXXX, 0XXXXXXXXY =400②
brk : 0>
```

- ① シーケンシャル・イベント検出を選択
- ② ブレークポイントを設定します。

```
brk : 0>BRS 2①
PA, EXT 0XXXX, 0XXXXXXXXY =500, 01010101②
brk : 0>
```

- ① オペランド以下を省略指定
- ② イベント条件を設定

★

(d) 前回に設定したイベント条件を取り消したいとき

前に設定したイベント条件を取り消すことができます。

```
brk : 0>BRS 1①
PARALLEL (P)/SEQUENTIAL (S) S=S②
PA, EXT 10H, 0XXXXXXXXY =C②
PA, EXT 20H, 0XXXXXXXXY =②
PA, EXT 20H, 0XXXXXXXXY =.②
brk : 0>BRS ②
BRS1 : Sequential
PA = 20H E = 0XXXXXXXXY
PA = 30H E = 0XXXXXXXXY
PA = 10H E = 0XXXXXXXXY
PA = 0XXXX E = 0XXXXXXXXY
BRS2 : PA = 0XXXX E = 0XXXXXXXXY
brk : 0>
```

- ① シーケンシャル・イベント検出を選択
- ② 取り消したいイベント条件でC^②を入力

8.4.7 チェック・ポイント設定コマンド (CHK)

入力形式	
形式1	CHK [Δ BR?][Δ BR?][Δ BR?][Δ BR?][Δ BR?][Δ BR?][Δ REG]
形式2	CHK [Δ BR?][Δ BR?][Δ BR?][Δ BR?][Δ BR?][Δ BR?][Δ SPR][Δ SPR][Δ SPR][Δ SPR] [Δ SPR]
形式3	CHK [Δ BR?][Δ BR?][Δ BR?][Δ BR?][Δ BR?][Δ BR?][Δ addr][Δ addr][Δ addr][Δ addr] [Δ addr]
	<div style="display: flex; justify-content: space-around; border-top: 1px dashed black; border-bottom: 1px dashed black; padding: 5px 0;"> オペランド1 オペランド2 </div>
基数	addr : H

【機能概要】

CHKコマンドは、オペランド1で選定された、イベント条件レジスタの内容(チェック・ポイントが設定されている)を検出すると、リアルタイム実行(RUNコマンドによる)を一時的に中断し、オペランド2で指定されたチェック・データ(汎用レジスタ、特殊レジスタ、メモリのいずれかの内容)をトレース・メモリに書き込みます。このあと、リアルタイム実行を再開します。

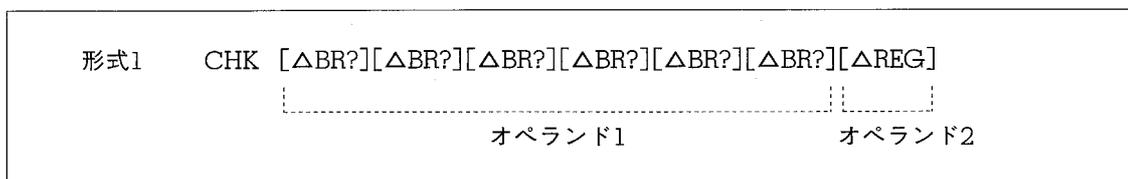
- (a) リアルタイム実行のDLY中(イベント検出からブレイクまでの間)では、CHKコマンドで指定したチェック条件を満足しても、チェック・データはトレース・メモリに書き込まれません。
- (b) オペランドが省略されたときには、現在選定されているチェック・ポイントとチェック・データが表示されます。
- (c) トレース・メモリに書き込まれたデータは、トレース・データ表示コマンド(TRD)によって確認することができます。
- (d) OUTコマンドでBRA3, BRA4が指定されている場合、CHKコマンドでBRA3, BRA4を指定することはできません。

【使用規約】

- (a) チェック・ポイントは、あらかじめイベント条件レジスタに設定しておく必要があります。
- (b) CHKコマンドで設定された条件と割り込みが同時に発生した場合、CHKコマンドが優先的に処理され、その後割り込みが処理されます。CHKコマンドの条件が連続している場合は、連続したCHKコマンドを処理したあと、割り込みが処理されます。

備考 画面の表示イメージは、8.4.39 トレース表示コマンド (TRD) を参照してください。

(1) 汎用レジスタ内容のチェック・データ出力指定



brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	×
-----------	-----------------------	-----------	-----------------------	-----------	---

【機能】

オペランド1で選定されたイベント検出条件のいずれかを満たすとき、汎用レジスタの内容を、チェック・データとしてトレース・メモリに書き込みます。

【指定方法】

(a) オペランド1

チェック・ポイントが設定されている、イベント条件レジスタを選定します。

BR? : イベント条件レジスタ名を、最大6個 (6ポイント) まで指定します。

BRA1, BRA2, BRA3, BRA4, BRS1, BRS2, OFF

(b) オペランド2

チェック・データ出力の対象となる、汎用レジスタの代表名を指定します。

REG : "REG" と指定します。

次のレジスタおよびフラグの内容が、チェック・データ出力の対象となります。

ただし、レジスタは対象デバイスによって異なります。

XA, HL, DE, BC (ペア・レジスタ)

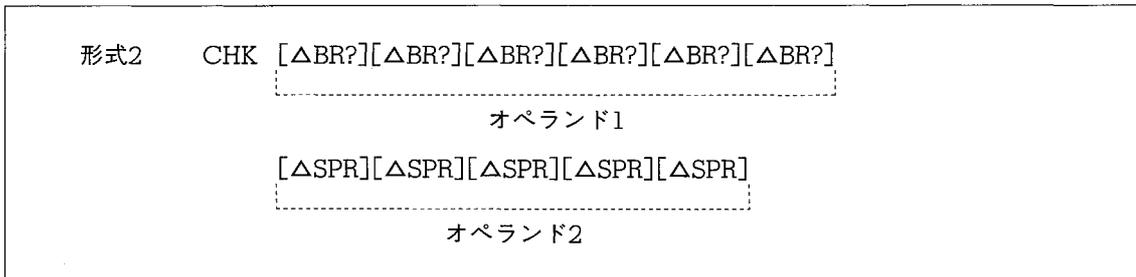
XA', HL', DE', BC'

SP (含むSBS), PC,

RBS (コントロール・レジスタ), MBS

CY, RBE, MBE, IST0 (フラグ), IST1

(2) 特殊レジスタ内容のチェック・データ出力指定



brk:n>	<input type="radio"/>	emu:n>	<input type="radio"/>	trc:n>	<input type="checkbox"/>
--------	-----------------------	--------	-----------------------	--------	--------------------------

【機能】

オペランド1で選定されたイベント検出条件のいずれかを満たすとき、オペランド2で指定された特殊レジスタの内容を、チェック・データとしてトレース・メモリに書き込みます。

【指定方法】

(a) オペランド1

チェック・ポイントが設定されている、イベント条件レジスタを選定します。

BR? : イベント条件レジスタ名を、最大6個（6ポイント）まで指定します。

BRA1, BRA2, BRA3, BRA4, BRS1, BRS2, OFF

(b) オペランド2

チェック・データ出力の対象となる特殊レジスタを指定します。

SPR : 特殊レジスタ名を、最大5個まで指定します。

③ データ・メモリ内容のチェック・データ出力指定

形式3 CHK [ΔBR?][ΔBR?][ΔBR?][ΔBR?][ΔBR?][ΔBR?] <div style="text-align: center; border-top: 1px dashed black; border-bottom: 1px dashed black; width: 300px; margin: 0 auto 10px auto;"> オペランド1 </div> [Δaddr][Δaddr][Δaddr][Δaddr][Δaddr] <div style="text-align: center; border-top: 1px dashed black; border-bottom: 1px dashed black; width: 300px; margin: 0 auto 10px auto;"> オペランド2 </div>
基数 addr : H

brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	×
-----------	-----------------------	-----------	-----------------------	-----------	---

【機能】

オペランド1で選定されたイベント検出条件のいずれかを満たすとき、オペランド2で指定されたデータ・メモリの内容を、チェック・データとしてトレース・メモリに書き込みます。

【指定方法】

(a) オペランド1

チェック・ポイントが設定されている、イベント条件レジスタを選定します。

BR? : イベント条件レジスタ名を、最大6個 (6ポイント) まで指定します。

BRA1, BRA2, BRA3, BRA4, BRS1, BRS2, OFF

(b) オペランド2

チェック・データ出力の対象となるデータ・メモリ・アドレスを指定します。

addr : データ・メモリのアドレスを、最大5個まで指定します。

マスク指定はできません。

アドレス指定範囲：対象デバイスにより異なります。

【使用例】

(a) 汎用レジスタ内容のチェック・データ出力指定

BRA1, BRA2をチェック・ポイントとして選定し、このときの汎用レジスタの内容をチェック・データとして、トレース・メモリに書き込むように指定した例です。

```
brk:0>BRA 1 MA=0100H V=11110000Y C=W8
```

①

```
brk:0>BRA 2 MA=0101H V=00001111Y C=W8
```

②

```
brk:0>CHK BRA1 BRA2 REG
```

③

```
brk:0>RUN N
```

④

- ① BRA1にチェック・ポイントを設定
- ② BRA2にチェック・ポイントを設定
- ③ BRA1, BRA2を選定し、汎用レジスタの内容出力を指定
- ④ エミュレーション開始指定

(b) データ・メモリ内容のチェック・データ出力指定

BRS1をチェック・ポイントとして選定し、このときのデータ・メモリの内容をチェック・データとして、トレース・メモリに書き込むように指定した例です。

```
brk:0>CHK
```

①

```
BRA 1 BRA2 REG
```

②

```
brk:0>BRS 1 P 02000H 02500H
```

③

```
brk:0>CHK BRS1 0100H 0200H 0123H
```

④

```
brk:0>RUN N
```

⑤

- ① 現選定チェック・ポイントの確認
- ② 現選定チェック・ポイントの表示
- ③ BRS1にチェック・ポイントを設定
- ④ BRS1を選定し、データ・メモリの内容出力を指定
- ⑤ エミュレーションの実行開始指定

8.4.8 クロック選択コマンド (CLK)

入力形式
形式1 CLK $\begin{bmatrix} \Delta U \\ \Delta I \end{bmatrix}$

brk : n >	<input type="radio"/>	emu : n >	<input type="checkbox"/>	trc : n >	<input type="checkbox"/>
-----------	-----------------------	-----------	--------------------------	-----------	--------------------------

【機能】

CLKコマンドは、オペランドで指定された、ターゲット・システム側のクロックまたはIE-75001-R側のクロックをクロック・ソースとして選択します。

オペランドが省略されたときには、現在選択されているクロックが表示されます。

注意 ターゲット・システム側のクロックを選択する場合は、必ずエミュレーション・ボードの部品台 (SX1, SX2) の設定をしてください。設定を行わないと、クロックがエミュレーション・デバイスに供給されません。

ユーザ・クロックの設定方法については、IE-75000-R-EM ユーザーズ・マニュアル (EEU-673) を参照してください。

【指定方法】

オペランド

次の形式で、どれをクロック・ソースとするか指定します。

U : ターゲット・システム側のクロックをクロック・ソースとする。

I : IE-75001-R側のクロックをクロック・ソースとする。

【使用規約】

クロック・ソース選択時は、自動的にRESコマンドが内部で実行されます。

【使用例】

クロック・ソースの変更

現在設定されているIE-75001-R側のクロックを、ターゲット・システム側のクロックに変更した例です。

```
brk:0>CLK  ①  
  
IE ②  
  
brk:0>CLK U  ③  
  
brk:0>CLK  ④  
  
User
```

- ① 現在選択されているクロックの確認
- ② 現在選択されているクロックの表示
- ③ ターゲット・システム側のクロックに変更
- ④ 変更した内容の確認

8.4.9 コマンド・ファイル作成コマンド (COM)

入力形式

形式1	COM [Δ d : file]
形式2	COM [Δ LST :]
形式3	COM [Δ CON :]

【機能概要】

COMコマンドは、オペランドで指定された、フロッピー・ディスク装置上のファイルまたはリスト出力装置を、コマンド・ファイル作成のためにオープンします。また、オープンされたこれらの装置やファイルをクローズします。

コマンド・ファイルに対する実際の出力開始および停止は、次の操作で行います。

① コマンド・ファイルの出力開始

コマンド入力待ちの状態では、“ \wedge O (CTRL+O)” キーを押します。この行からのコマンドがファイルまたはリストに出力されます。

② コマンド・ファイルの出力停止

コマンド・ファイルを出力中、かつコマンド入力待ちの状態では、“ \wedge O” キーを押します。この行からのコマンドは、ファイルまたはリストへの出力が停止されます。

(1) フロッピー・ディスク装置ファイルのオープン指定

形式1 COM [Δ d : file]

brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="radio"/>
-----------	-----------------------	-----------	-----------------------	-----------	-----------------------

【機能】

オペランドで指定された、フロッピー・ディスク装置上のファイルを、コマンド・ファイル作成のためにオープンします。

ファイル名が登録済みの場合には、ファイルの属性指定によって次の処理を行います。

(a) DIR属性かつR/W属性のとき

次のメッセージを表示して、登録済みのファイルを消去し新たにコマンド・ファイルを作成するか、または登録済みのファイルをそのまま残すか、ユーザに選択させます。

File already exists. Delete ? (Y or N) : _____

Y : 登録済みのファイルを消去して、新たにコマンド・ファイルを作成する。

N : 登録済みのファイルをそのまま残す。

(b) SYS属性またはR/O属性のとき

次のメッセージを表示して、コマンドを無視します。

File already exists.

【指定方法】

オペランド

オープンするフロッピー・ディスク装置番号およびファイル名を指定します。

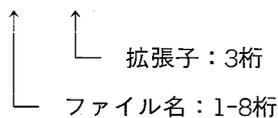
d: : フロッピー・ディスク装置番号を指定します。

A: フロッピー・ディスク装置-A

B: フロッピー・ディスク装置-B

file : ファイル名を指定します。

XXXXXXXX.YYY



(第1文字目は英字)

(2) リスト装置のオープン指定

形式2 COM [ΔLST:]

brk : n > emu : n > trc : n >

【機 能】

オペランドで指定されたリスト装置を、コマンドやデータを出力するためにオープンします。

リスト装置が他の処理で使用中の場合には、次のメッセージを表示してコマンドを無視します。

List device is used by other process.

【指定方法】

オペランド

“LST:” と指定します。

③ コマンド・ファイル出力装置のクローズ指定

形式3	COM [ΔCON:]
-----	-------------

brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="radio"/>
-----------	-----------------------	-----------	-----------------------	-----------	-----------------------

【機 能】

オープンされている、フロッピー・ディスク装置上のファイルあるいはリスト出力装置を、クローズします。

【指定方法】

オペランド

“CON :” と指定します。

【使用例】

コマンド・ファイルをフロッピー・ディスク装置上のファイルに作成する例です。

下図の②, ③, ④, ⑤, ⑥が, コマンド・ファイルに出力されます。

brk : 0 > <u>COM B : SAMPLE. STR</u> <input type="checkbox"/>	①
brk : 0 > <u>MEM F 0, 1FF 1, 2, 3, 4, 5, 6, 7</u> <input type="checkbox"/>	②注
brk : 0 > <u>MEM D 010X</u> <input type="checkbox"/>	③
0100 01 02 03 04 05 06 07...	
brk : 0 > <u>BRA 1 MA=100</u> <input type="checkbox"/>	④
brk : 0 > <u>BRM BRA1</u> <input type="checkbox"/>	⑤
brk : 0 > <u>TRM ALL \$M</u> <input type="checkbox"/>	⑥
brk : 0 > ■	⑦注

① 装置 : B, ファイル名 : SAMPLE. STR をオープン

② “^O (CTRL+O)” を入力後, “MEM F 0, 1FF 1, 2, 3, 4, 5, 6, 7 ” を入力

③ }
 ④ } ②-⑥ と実行結果を, コマンド・ファイルに出力
 ⑤ }
 ⑥ }

⑦ “^O” を入力し, SAMPLE. STR をクローズ

注 “^O” は画面上では表示されません。

8.4.10 カバレッジ測定結果操作コマンド (CVD)

入力方式	
形式1	CVD [Δ D] [partition]
形式2	CVD [Δ K]
基数	partition : H

【機能概要】

CVDコマンドは、カバレッジの測定結果を表示したり、測定結果を削除（初期化：カバレッジ測定実行前の状態に戻すこと）することができます。

カバレッジ測定の結果は、ターゲット・プログラムのエミュレーション実行部分と未実行部分を以下の識別子によって示します。

識別子表示	意 味	64バイト単位で表示時の優先順位
"X"	カバレッジ測定範囲外での実行を示す	1
"*"	カバレッジ測定範囲内での実行を示す	3
."	カバレッジ測定範囲内での未実行を示す	2
" Δ "	カバレッジ測定範囲外での未実行を示す	4

備考 カバレッジ測定範囲の設定については、**8.4.11 カバレッジ測定範囲設定コマンド(CVM)**を参照してください。

【使用規約】

カバレッジの測定は、「カバレッジ測定範囲の設定コマンド (CVM)」によって設定された範囲が対象となります。したがって、カバレッジの測定を行うためには、事前にこの設定を行う必要があります。

(1) カバレッジ測定結果の表示指定

形式1 CVD [Δ D] [partition]	
基数	partition : H

brk : n >	○	emu : n >	×	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

オペランドで指定された、開始アドレスから終了アドレスで示される、プログラム・メモリのカバレッジ測定結果を表示します。

このときの表示は、1バイト単位で行われます。すなわち、オブジェクト・コードの1バイトを識別子1個 (X, *, ., Δ のいずれか1個) に対応させて表示します。

- (a) サブコマンド以下が省略された場合には、プログラム・メモリの0-0FFFFH番地までの測定結果を表示します。このときの表示は、64バイト単位で行われます。すなわちオブジェクト・コードの64バイトを識別子1個に対応させて表示します。

- (b) オペランド以下が省略された場合にも、上記と同様の働きをします。

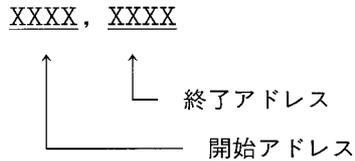
【指定方法】

(a) サブコマンド

D : "D" と指定します。

(b) オペランド

partition : カバレッジ測定結果の表示範囲を、開始アドレスと終了アドレスで指定します。



② カバレッジ測定結果の削除指定

形式2 CVD ΔK

【機 能】

カバレッジ測定結果の識別を消去し、ターゲット・プログラムのエミュレーションが未実行の状態に戻します。

すなわち、測定識別子の「X」を「Δ」、「*」を「.」に変更します。

【指定方法】

サブコマンド

K: "K" と指定します。

【使用例】

(a) カバレッジ測定結果の範囲表示指定

開始アドレス (100H番地) と終了アドレス (1FFH番地) の範囲で示される、プログラム・メモリの測定結果を、1バイト単位で表示した例です。

```

brk:0>CVD D 100,1FF①
addr  0          10 ..... 30 ②
0100 *****③
0140 *****
0180 *****.....
01C0 .....④

brk:0>■
```

- ① 100H-1FF H番地を表示指定
- ② 以下、測定結果を1バイト単位で表示
- ③ 実行済みの表示
- ④ 18A番地まで実行されたことが確認できる(それ以降は、未実行)

(b) カバレッジ測定結果の全メモリ表示指定

サブコマンド以下を省略し、全プログラム・メモリの測定結果を、64バイト単位で表示した例です。

```
brk:0>CVD  ①
  

  addr  000   100   200   300   400.....FOO ②
  0000  *****
  1000  ... *****.....
  2000  .....
  .
  .
  .
  F000
  

brk:0>■
```

- ① サブコマンド以下を省略して指定
- ② 以下、測定結果を64バイト単位で表示
(0番地からのOFFF番地までと、1100番地から12FF番地が実行されたことが確認できる)

(c) カバレッジ測定結果の削除指定

カバレッジ測定結果の削除を行った例です。測定識別子の実行済み (*) を未実行 (.) に変更します。

```
brk : 0>CVD K ①
  

brk : 0>■
```

- ① カバレッジ測定結果の削除を指定

8.4.11 カバレッジ測定範囲設定コマンド (CVM)

入力形式	
形式1	CVM [Δ A] [partition]
形式2	CVM [Δ D]
形式3	CVM [Δ K] [partition]
基数	partition : H

【機能概要】

CVMコマンドは、カバレッジ測定の範囲設定、範囲の表示、あるいは範囲設定の解除を行うものです。

IE-75001-Rが持つカバレッジ機能は、COカバレッジと呼ばれます。これは、RUNコマンドによって、ターゲット・プログラムのリアルタイム・エミュレーション実行を行った際に、その実行部分に対し識別子を付加する機能です。これによってユーザは、ターゲット・プログラムのどの部分が実行済みであり、どの部分が未実行あるいは指定範囲外であるか、容易に判別できます。

識別子表示	意味
"X"	カバレッジ測定範囲外での実行を示す
"*"	カバレッジ測定範囲内での実行を示す
"."	カバレッジ測定範囲内での未実行を示す
" Δ "	カバレッジ測定範囲外での未実行を示す

備考 カバレッジ測定結果の内容を確認するには、CVMコマンドを使用して行います。詳細は、**8.4.10 カバレッジ測定結果操作コマンド (CVD)** を参照してください。

(1) カバレッジ測定範囲の設定, 追加

形式1 CVM [ΔA] [partition]					
基数	partition : H				
brk : n >	<input type="radio"/>	emu : n >	<input type="checkbox"/>	trc : n >	<input type="checkbox"/>

【機能】

オペランドで指定された, 開始アドレスと終了アドレスで示されるプログラム・メモリの範囲を, カバレッジの測定範囲として設定または追加します。

サブコマンド以下が省略された場合には, **CVM D** (カバレッジ測定範囲の表示コマンドと, 同一の機能を果たします。

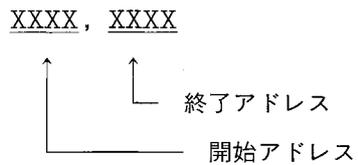
【指定方法】

(a) サブコマンド

A : "A" と指定します。

(b) オペランド

partition : カバレッジ測定範囲を, 開始アドレスと終了アドレスで指定します。



② カバレッジ測定範囲の表示

形式2 CVM [Δ D]

brk : n >	<input type="radio"/>	emu : n >	<input type="checkbox"/>	trc : n >	<input type="checkbox"/>
-----------	-----------------------	-----------	--------------------------	-----------	--------------------------

【機 能】

カバレッジ測定範囲の表示を行います。

オブジェクトのロード直後は、ロードされた領域がカバレッジの測定範囲に設定されています。

【指定方法】

サブコマンド

D: "D" と指定します。

(3) カバレッジ測定範囲の解除

形式3 CVM [Δ K] [partition]	
基数	partition : H

brk : n >	○	emu : n >	×	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

オペランドで指定された、開始アドレスと終了アドレスで示されるプログラム・メモリの範囲を、カバレッジの測定範囲から解除します。

オペランドが省略された場合には、全プログラム・メモリ (0-0FFFFH番地) をカバレッジ測定範囲から解除します。

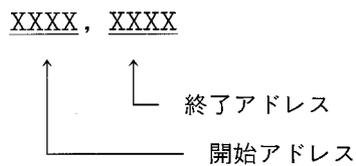
【指定方法】

(a) サブコマンド

K : "K" と指定します。

(b) オペランド

partition : カバレッジ測定の解除範囲を、開始アドレスと終了アドレスで指定します。



【使用例】

(a) カバレッジ測定範囲の設定

開始アドレス (0H番地) と終了アドレス (1FFF番地) の範囲で示されるプログラム・メモリを、カバレッジ測定範囲に設定し、さらに3000-3FFFH, 5500-5FFFHの範囲を追加する例です。

```
brk:0>CVM A 0, 1FFF
```

①

```
brk:0>CVM A 3000, 3FFF
```

②

```
brk:0>CVM A 5500, 5FFF
```

③

```
brk:0>■
```

- ① 0H-1FFF番地を測定範囲に指定
- ② 3000-3FFF番地を測定範囲に追加
- ③ 5500-5FFF番地を測定範囲に追加

(b) カバレッジ測定範囲の表示指定

前述(a)で設定したカバレッジ測定範囲を、表示して確認する例です。

```
brk:0>CVM D
```

①

```
0000,1FFF 3000, 3FFF 5500, 5FFF
```

②

```
brk:0>■
```

- ① カバレッジ測定範囲の表示指定
- ② カバレッジ測定範囲の表示

(c) カバレッジ測定範囲の解除

前述(a)で設定した範囲を，削除しながら確認をする例です。

```
brk : 0 > CVM K 3000, 3FFF
```

①

```
brk : 0 > CVM D
```

②

```
0000, 1FFF 5500, 5FFF
```

③

```
brk : 0 > CVM K
```

④

```
brk : 0 > CVM D
```

⑤

```
nothing
```

⑥

```
brk : 0 > ■
```

- ① 3000-3FFFH番地を測定範囲から削除
- ② カバレッジ測定範囲の表示を指定
- ③ 現在設定されている範囲を表示
- ④ 全メモリを，測定範囲から削除
- ⑤ カバレッジ測定範囲を表示し確認
- ⑥ 設定範囲がないときの表示

8.4.12 逆アセンブル・コマンド (DAS)

入力形式	
形式1	DAS [Δ addr Δ partition]
基数	addr : H partition : H

brk : n >	<input type="radio"/>	emu : n >	<input type="checkbox"/>	trc : n >	<input type="checkbox"/>
-----------	-----------------------	-----------	--------------------------	-----------	--------------------------

【機能】

DASコマンドは、指定されたプログラム・メモリ上のオブジェクト・コードを75Xシリーズのニモニックに変換（ディスアセンブル）して、画面に表示します。

ユーザは、本コマンドとアセンブル・コマンドを使用することにより、煩わしい機械語の判読から解放され、プログラムの変更や修正を容易に行うことができます。

(a) 命令セットの逆アセンブル

逆アセンブル可能な、「命令セット」および「SPR予約語名」は、対象デバイスによって異なります。

(b) データの逆アセンブル

データ・メモリを操作する命令を逆アセンブルする場合、次の条件を満足しているときのみ、シンボル変換を行います。

- ① データ属性 ("D") のシンボルがあること。
- ② 上記シンボルの「上位4ビットの値」が、「BNKコマンド」で指定された値と等しいこと。
- ③ さらに、そのシンボルの「下位8ビットの値」が、当該命令のオペランドで指定されたデータ・メモリ・アドレスと同じであること。

(c) アドレス指定の場合は、指定アドレスから逆アセンブルを行い、表示します。

(d) パーティション指定の場合は、開始アドレスから終了アドレスで示される範囲の逆アセンブルを行い、表示します。

(e) オペランドを省略した場合は、直前に逆アセンブルが終了した次の命令から、17バイト目のアドレスを含む命令までを逆アセンブルして、表示します。

D

【指定方法】

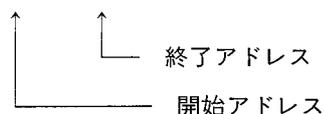
オペランド

addr : 逆アセンブルの開始アドレスを指定します。

アドレス指定範囲：0-OFFFFH

partition : 逆アセンブルの範囲を、開始アドレスと終了アドレスで指定します。

XXXX, YYYY (XXXX < YYYY)



XXXX : 0-OFFFFH

YYYY : 0-OFFFFH

【使用例】

プログラム・メモリの100H番地から、逆アセンブルを行う例です。

```
brk:0>DAS 100①
Addr Code Label Mnem. Operand
0100 9A09 MOV X, #0000H
0102 71 MOV A, #0001H
0103 9A2F MOV B, #0002H
0105 9A3E MOV C, #0003H
0107 9A4D MOV D, #0004H
0109 9A5C MOV E, #0005H
010B 9A6B MOV H, #0006H
010D 9A7A MOV L, #0007H
010F 60 NOP
0110 60 NOP
brk:0>■
```

- ① プログラム・メモリ100H番地から逆アセンブルを指定
- ② 100H番地から逆アセンブルの結果を表示

8.4.13 ディレクトリ表示コマンド (DIR)

入力形式

形式1 DIR [Δ file]brk : n > emu : n > trc : n >

【機能】

DIRコマンドは、カレント・ディスク・ドライブ上の全ディレクトリまたはファイル上のディレクトリを画面に表示します。

オペランドが省略された場合は、カレント・ドライブ上の全ディレクトリを表示します。

【指定方法】

オペランド

file : ファイル名を指定します。

【使用例】

カレント・ドライブ上のディレクトリを表示した例です。

brk:0>DIR

Directory=A: ¥SS

```

<          >  <..          >  COMMAND.COM  SAMPLE1.EXE  SAMPLE2.EXE
IE75000 HLP  IE75000  EXE  ASM01.DAT  ASM02.DAT  DEMO1.EXE
SAMPLE3.EXE  SAMPLE.MAC  ERROR.EXE  ERRSUB.EXE  DUMP.EXE
DISPLAY.EXE  EXTERNAL.MAC  FILE  LST

```

brk:0>■

8.4.14 イベント検出点設定コマンド (DLY)

入力形式	
形式1	DLY [ΔF] [ΔM] [ΔL]

brk:n>	<input type="radio"/>	emu:n>	<input type="radio"/>	trc:n>	<input type="checkbox"/>
--------	-----------------------	--------	-----------------------	--------	--------------------------

【機能】

DLYコマンドは、リアルタイム・エミュレーション実行中に、イベント条件検出を行ったときの、トレース停止タイミングを設定します。

イベント検出点（トリガ・ポイント）とトレース・メモリに出力されるトレース情報の停止タイミングの関係は、次の3種類であり、この中からユーザが選択することができます。

なお、初期設定は、L (Last) に設定されています。

(a) F (First) 設定

イベント検出点を、トレース・メモリの最初に設定します。

イベントの検出を行うと、直前までに出力されたトレース・データは捨てられ、イベント検出時点のトレース・データ（トリガ・フレーム）をトレース・メモリの最初に出力します。以後トレース・データの出力を継続し、トレース・メモリが満杯になった時点でトレースを停止します。

ユーザは、トリガ・フレームから以降のトレース・データを確認することができます。

(b) M (Middle) 設定

イベント検出点を、トレース・メモリの中央に設定します。

イベント検出点のトレース・データ（トリガ・フレーム）がトレース・メモリの中央に設定され、残りの半分のトレース・メモリがフルになったときトレースを停止します。

ユーザは、イベント検出点を中心として、その前後のトレース・データを確認することができます。

(c) L (Last) 設定

イベント検出点を、トレース・メモリの最後に設定します。

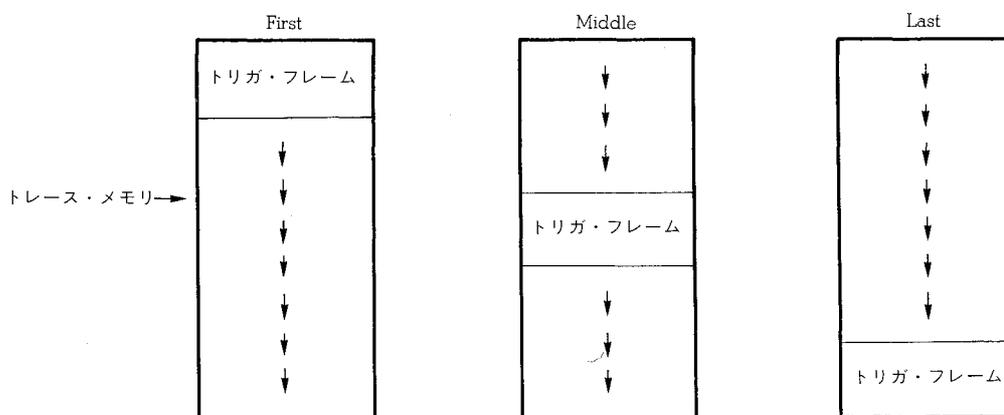
イベントの検出を行うと、イベント検出時点のトレース・データ（トリガ・フレーム）をトレース・メモリに出力して、直ちにトレースを停止します。

そしてトリガ・フレームがトレース・メモリの最後に出力された形になります。

ユーザは、トリガ・フレームから以前のトレース・データを確認することができます。

以上を図示すると、次のようになります。

図 8-2 トリガ・フレームのポイント



(d) オペランドが省略された場合

オペランドが省略された場合は、イベント検出点の設定状態を表示します。

【指定方法】

オペランド

F：イベント検出点を、トレース・メモリの最初に設定

M：イベント検出点を、トレース・メモリの中央に設定

L：イベント検出点を、トレース・メモリの最後に設定

【使用例】

現在設定されているイベント検出点を表示して確認したのちに、これをトレース・メモリの中央に設定し直して、その内容を表示する例です。

```
brk : 0 > DLY  ①  
  
FIRST ②  
  
brk : 0 > DLY M  ③  
  
brk : 0 > DLY  ④  
  
MIDDLE ⑤  
  
brk : 0 > ■
```

- ① 設定内容の表示指定
- ② 現在設定されている内容の表示
- ③ イベント検出点を、トレース・メモリの中央に設定
- ④ 設定内容の表示指定
- ⑤ 現在設定されている内容の表示

8.4.15 DOSコマンド実行コマンド (DOS)

入力形式
形式1 DOS

brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="radio"/>
-----------	-----------------------	-----------	-----------------------	-----------	-----------------------

【機能】

DOSコマンドは、IE-75000-Rコントロール・プログラムを終了させることなく、一時的に制御をOS (DOS) に移します。これによって、OSの持つ各種のコマンド機能を使用することができます。

OSから再びIE-75000-Rコントロール・プログラムに制御を戻すには、OSの「EXITコマンド」を使用して行います。

【指定方法】

コマンド本体のみ入力します。

【使用規約】

本コマンドでOSに制御がわたっているときは、フロッピー・ディスク等の脱着を行わないでください。内容が破壊される場合があります。

本機能は、COMMAND.COMを実行することで実現する機能です。COMMAND.COMは、COM-SPECのデータをもとに最初に検索し、見つからない場合はPATHのデータを参照し、COMMAND.COMを検索します。

検索で見つかった場合は、そのCOMMAND.COMを使用します。見つからない場合は本機能を実現することはできません。

【使用例】

DOSコマンドを用いOSへ制御を移し、各種OSコマンドを使用したあと、再びIE-75000-Rコントロール・プログラムに制御を戻す例です。

```
brk : 0>DOS①  
  
A>TIME②  
:  
:  
A>EXIT③  
  
brk : 0>④
```

- ① OSに制御を移す指定
- ② OSのプロンプトが表示されて、OSのコマンドが使用可能
- ③ OSから、IE-75000-Rコントロール・プログラムに制御を戻す指定
- ④ IE-75000-Rコントロール・プログラムに制御が戻り、IE-75000-Rコントロール・プログラムのプロンプトを表示

8.4.16 コントロール・プログラム終了コマンド (EXT)

入力形式
形式1 EXT

brk : n >	○	emu : n >	×	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

EXTコマンドは、IE-75000-Rコントロール・プログラムの処理を終了し、OS (DOS) に制御を渡します。

業務を終了する場合には、必ず本コマンドを使用します。

【指定方法】

コマンド本体のみ入力します。

【使用規約】

- (a) 業務を終了する場合には、必ず本コマンドを使用してください。
- (b) 本コマンドは、ブレーク・モード以外では使用できません。

【使用例】

デバッグ作業をすべて完了し、OSに制御を戻す例です。

brk : 0 > EXT	①
A > ■	②

- ① OS (DOS) に制御を戻す指定
- ② OSに制御が戻り、OSがプロンプトを表示

E

8.4.17 コマンド・ヒストリ表示コマンド (HIS)

入力形式
形式1 HIS

brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="radio"/>
-----------	-----------------------	-----------	-----------------------	-----------	-----------------------

【機能】

HISコマンドは、コマンド・ヒストリ・メモリに保存されている、コマンドの表示およびその再実行を行います。

(a) コマンドの表示

ヒストリ・メモリに記憶されているコマンドについて、古いものから最新のものまでを順番に番号を付加しながら表示します。

なお、ヒストリ・メモリには、最新のコマンドを含め20行分のコマンドまでさかのぼって保存されています。したがって、最大20行まで表示できます。

(b) コマンドの再実行

ヒストリ・メモリに保存されているコマンドは、呼び出したあと、再実行することができます。その方法は次のとおりです。

(i) 番号指定による呼び出し

特定コマンドを呼び出す場合には、次の形式で行います。

ln <cr> (n : コマンド番号)

(ii) 最新コマンドの呼び出し

最新コマンドを呼び出す場合は、次の形式で行います。

!! <cr>

(iii) コマンドの再実行

呼び出されたコマンドが表示されると、カーソルがそのコマンド行の後尾に表示されます。

ここで<cr>を入力します。

【指定方法】

コマンド本体のみ入力します。

【使用例】

(a) コマンド・履歴の表示

コマンド・履歴・メモリに保存されているコマンドを表示する例です。

```
brk:0>HIS①
1  LOD TEST②
2  CLK
   :
   :
20 HIS
brk:0>■
```

- ① コマンド・履歴表示指定
- ② コマンド番号を付加し、コマンドを表示（20行目が最新）

(b) コマンドの再実行

上述(a)で確認後、コマンド番号=2の「CLKコマンド」を呼び出して再実行し、クロックの状態を確認する例です。

```
brk:0>!2①
CLK②
IE③
brk:0>■
```

- ① コマンド番号2の呼び出し指定
- ② <cr>を入力し、CLKコマンドを再実行
- ③ クロックの状態表示

8.4.18 ヘルプ・コマンド (HLP)

入力形式

形式1 HLP [Δ command]

brk:n>	<input type="radio"/>	emu:n>	<input type="radio"/>	trc:n>	<input type="radio"/>
--------	-----------------------	--------	-----------------------	--------	-----------------------

【機能】

HLPコマンドは、コマンドの一覧、コマンドの使用方法を表示します。

本コマンドを実行すると、ヘルプ・モードのコマンド入力待ち“HLP>”になり、このモード内でコマンドを入力すると、そのコマンドの使用方法を表示します。

ヘルプ・モードから、通常のコマンド待ちにする場合は、<cr>のみを入力します。

【指定方法】

オペランド

command : 表示するコマンド名を指定します。

省略時 : コマンドの一覧を表示します。

【使用規約】

ヘルプ・モード時に入力されたコマンドは、実行しません。

【使用例】

(a) ヘルプ・コマンドの使用例

ディレクトリ・コマンドの使用方法を表示する例です。

```
brk : 0>HLP DIR①

(Directory Dump Command)

DIR[_d : ][file name.ext]          display file directory

    d          =drive number          (omission=default drive)
                use character A-P

    file name  =file name is organized by 8 characters

    ext        =extend character is organized by 3 characters
                use character of file name and ext
                (0-9, A-Z, Special character omit = < > . , ; : = []

HLP>■②
```

① “DIR” コマンドの使用例を表示する指定

② ヘルプ・モード

(b) コマンド一覧の表示

IE-75001-Rで使用できるコマンドの一覧を表示します。

```
brk : 0>HLP①

Command Table
ASM      BNK      BRA      BRK      BRS
BRM      CHK      CLK      COM      CVD
CVM      DAS      DIR      DLY      DOS
EXT      HIS      HLP      LOD      LST
MAT      MEM      MOD      OUT      PAS
PGM      RAM      REG      RES      RUN
SAV      SET      SPR      STR      STP
STS      SYM      TRD      TRF      TRG
TRM      TRP      TRX      TRY      VRY

HLP>■
```

① コマンド一覧の表示指定

(c) コマンドの使用方法の表示

ヘルプ・モードでコマンドを入力するとコマンドの使用方法が表示されます。

```
HLP>DIR①
(Directory Dump Command)
DIR[_d : ][file name.ext]      display file directory
    d      =drive number      (omission=default drive)
           use character A-P
file name =file name is organized by 8 characters
ext       =extend character is organized by 3 characters
           use character of file name and ext
           (0-9, A-Z, Special character omit = < > . , ; : = [ ]
HLP>■
```

① ヘルプ・モードにおけるコマンド入力

8.4.19 ロード・コマンド (LOD)

入力形式
形式1 LOD△file [△ module¥----] [△ option]

brk : n >	<input type="radio"/>	emu : n >	<input type="checkbox"/>	trc : n >	<input type="checkbox"/>
-----------	-----------------------	-----------	--------------------------	-----------	--------------------------

【機能】

LODコマンドは、指定ファイル名のオブジェクト・ファイル、デバッグ環境ファイル、シンボル・ファイルをホスト・マシンからダウン・ロードします。

ダウン・ロードには、次の2とおりの方式があります。

ロード方法	内 容	選択方法
高速ダウン・ロード	ホスト・マシンのセントロニクスI/F出力からIE側のセントロニクスI/F入力(CH4)へダウン・ロードします。	IE-75001-Rの電源を投入し、コントロール・プログラムの起動後、指定するとき、高速ダウン・ロード・モードを選択します。
通常ダウン・ロード	ホスト・マシンのRS-232-C I/FチャンネルからIE側のRS-232-C I/Fチャンネル(CH1)へダウン・ロードします。	IE-75001-Rの電源を投入し、コントロール・プログラムの起動後、指定するとき、高速ダウン・ロード・モードを選択しません。

【指定方法】

オペランド

- file : オブジェクト・ファイル, シンボル・ファイル, デバッグ環境ファイルの拡張子を除いたファイル名
- module ¥ : シンボル・ファイルをロードする場合のモード名
- option : "C", "D", "S" のいずれか1つ

option	機 能
C	オペランドで指定したファイル名で拡張子 ".HEX" のオブジェクト・ファイルをロードします。
D	オペランドで指定したファイル名で拡張子 ".DBG" のデバッグ環境ファイルをロードします。 環境ファイルにより, 次のコマンドが環境設定されます。 BNK, BRA1, BRA2, BRA3, BRA4, BRK, BRM, BRS1, BRS2, CHK, CLK, DLY, MOD, OUT, PAS, PGM, REG, SET, STS, TRF, TRM, TRX, TRY
S	オペランドで指定したファイル名で拡張子 ".SYM" のシンボル・ファイルをロードします。 指定ファイル名のあとにモジュール名を指定した場合, 指定モジュールのシンボル・ファイルのみをロードします。
省略時	オプション "C", "D", "S" の処理をすべて行います。

【使用規約】

オブジェクト・ファイルをロードした場合, オブジェクトの格納アドレスは, 自動的にカバレッジ測定範囲に設定されます (8.4.11 カバレッジ測定範囲設定コマンド (CVM) 参照)。

【使用例】

(a) 各ファイルの同時ロード指定

オブジェクト・ファイル、デバッグ環境ファイル、シンボル・ファイルを同時にロードする例です。

```
brk : 0>LOD SAMPLE   
  
Debug condition load (Y/N) ? Y   
Debug condition load complete  
object load complete  
symbol table loading  
PUBLIC load complete  
MOD01 load complete  
MOD02 load complete  
  
brk : 0>■
```

(b) 指定モジュールのファイルのロード指定

オブジェクト・ファイル、デバッグ環境ファイルおよび指定モジュールのシンボル・ファイルをロードする例です。

```
brk : 0>LOD SAMPLE PUBLIC ¥, MOD02 ¥   
  
Debug condition load (Y/N) ? Y   
Debug condition load complete  
object load complete  
symbol table loading  
PUBLIC load complete  
MOD02 load complete  
  
brk : 0>■
```

(c) 環境ファイルのロード指定

環境ファイルのみをロードする例です。

```
brk : 0>LOD SAMPLE D   
  
debug condition load complete  
  
brk : 0>■
```

8.4.20 出力デバイス・リダイレクト・コマンド (LST)

入力形式

形式1	LST [Δ d : file]
形式2	LST [Δ LST :]
形式3	LST [Δ CON :]

【機能概要】

LSTコマンドは、コマンドの実行結果をコンソールとともに指定された出力デバイスに出力するために、ファイルあるいはリスト装置をオープンします。

出力デバイスに対する実際の出力開始および停止は、次の操作で行います。

(a) デバイス出力開始

コマンド入力待ち状態で、“ \wedge P (**CTRL**+P)” キーを押します。この行からのコマンドの実行結果が、指定デバイスに出力されます。

(b) デバイス出力停止

指定デバイスに出力中、かつコマンド入力待ちの状態で“ \wedge P” キーを押します。この行からのコマンドの実行結果は、出力が停止されます。

(1) ファイルのオープン指定

形式1 LST [Δ d : file]

brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="radio"/>
-----------	-----------------------	-----------	-----------------------	-----------	-----------------------

【機能】

オペランドで指定されたファイルを，リダイレクト・ファイル作成のため，オープンします。
 ファイル名が登録済みの場合には，ファイルの属性指定によって次の処理を行います。

(a) DIR属性かつR/W属性の場合

次のメッセージを表示して，登録済みのファイルを消去し新たにリダイレクト・ファイルを作成するか，または登録済みのファイルをそのまま残すか，ユーザが選択します。

File already exists. Delete? (Y or N) : _____

Y : 登録済みのファイルを消去して，新たにリダイレクト・ファイルを作成します。

N : 登録済みのファイルをそのまま残します。

(b) SYS属性またはR/O属性のとき

次のメッセージを表示して，コマンドを無視します。

File already exists.

【指定方法】

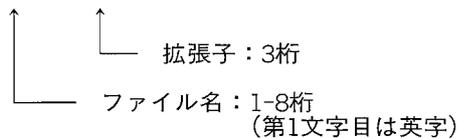
オペランド

オープンするドライブ番号とファイル名を指定します。

d : ドライブ番号を指定します。

file : ファイル名を指定します。

xxxxxxxxx.yyy



② リスト装置のオープン指定

形式2 LST [Δ LST :]

brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="radio"/>
-----------	-----------------------	-----------	-----------------------	-----------	-----------------------

【機 能】

オペランドで指定されたリスト装置を、コマンドの実行結果を出力するためにオープンします。

リスト装置が他の処理で使用中の場合には、次のメッセージを表示してコマンドを無視します。

List device is used by other process.

【指定方法】

オペランド

“LST :” と指定します。

③ リダイレクト・ファイル出力装置のクローズ指定

形式3 LST [Δ CON :]

brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="radio"/>
-----------	-----------------------	-----------	-----------------------	-----------	-----------------------

【機 能】

オープンされているファイル、あるいはリスト出力装置を、クローズします。

【指定方法】

オペランド

“CON :” と指定します。

【使用例】

ファイルを出力デバイスとしてオープンする例です。次の②～⑦がSAMPLE.TXTというファイルに出力されます。

brk : 0> <u>LST B : SAMPLE. TXT</u> <input type="checkbox"/>	①
brk : 0> <u>MEM F 0, 1FF 1, 2, 3, 4, 5, 6, 7, 8</u> <input type="checkbox"/>	②注
brk : 0> <u>MEM D 080X</u> <input type="checkbox"/>	③
0800 01 02 03 04 05 06 07...	④
brk : 0> <u>BRA A=100</u> <input type="checkbox"/>	⑤
brk : 0> <u>BRM BRA1</u> <input type="checkbox"/>	⑥
brk : 0> <u>TRM ALL \$M</u> <input type="checkbox"/>	⑦
brk : 0>■	⑧注

- ① 装置 : B, ファイル名 : SAMPLE. TXTをオープン
- ② “ ^ P (**CTRL**+P)” を入力後, “MEM F 0, 1FF 1, 2, 3, 4, 5, 6, 7, 8” を入力
- ③ } ②-⑦と実行結果を, SAMPLE. TXTに出力
- ④ } ②-⑦と実行結果を, SAMPLE. TXTに出力
- ⑤ } ②-⑦と実行結果を, SAMPLE. TXTに出力
- ⑥ } ②-⑦と実行結果を, SAMPLE. TXTに出力
- ⑦ } ②-⑦と実行結果を, SAMPLE. TXTに出力
- ⑧ “ ^ P” を入力し, SAMPLE. TXTをクローズ

注 “ ^ P” は画面上では表示されません。

8.4.21 演算コマンド (MAT)

入力形式
形式1 MAT△ expression

brk : n> emu : n> trc : n>

【機能】

MATコマンドは、オペランドに記述された式表現を評価し、その結果を16進数、10進数、8進数、2進数の順で表示します。

【指定方法】

オペランド

expression : word, byte, symbolを、次に示す演算子で結合して記述します。

()	高
* /	↑
+ -	優先順位
AND	↓
OR XOR	低

【使用規約】

演算は、すべて16ビットの整数で行われます。演算結果が16ビットをオーバした場合は、下位の16ビットだけが有効となります。

【使用例】

MATコマンドを用いた演算の例です。

```
brk : 0>MAT 5H+7H AND 17Q  ①
      0CH, 12T, 14Q, 1100Y ②
brk : 0>■
```

- ① 式の入力
- ② 演算結果の表示

8.4.22 プログラム・メモリ操作コマンド (MEM)

入力形式			
形式1	MEM	ΔC	[Δaddr]
形式2	MEM	[ΔD	[$\left\{ \begin{array}{l} \Delta\text{addr} \\ \Delta\text{partition} \end{array} \right\}$]
形式3	MEM	ΔE	[Δpartition]
形式4	MEM	Δ	$\begin{bmatrix} F \\ G \end{bmatrix}$ Δpartition Δdata-string
形式5	MEM	Δ	$\begin{bmatrix} M \\ X \\ V \end{bmatrix}$ Δpartition Δaddr
基数	addr : H	partition : H	data-string : H

【機能概要】

MEMコマンドは、プログラム・メモリのメモリ内容の変更、表示、テスト、イニシャライズ、サーチ、コピー、交換、比較を8ビット単位で行います。

(1) メモリ内容の変更

形式1	MEMΔC [addr]				
基数	addr : H				
brk : n >	<input type="radio"/>	emu : n >	<input type="checkbox"/>	trc : n >	<input type="checkbox"/>

【機能】

プログラム・メモリの内容を変更します。

【指定方法】

オペランド

addr : プログラム・メモリの変更開始アドレスを指定します。

マスク指定はできません。

アドレス指定範囲 : 0-0FFFFH

省略時 : 前回に終了したアドレスがとられます。

初期値 : 0H

【使用例】

100H番地からのメモリ内容を変更する例です。

```

brk : 0>MEM C 100①

0100 00 11②
0101 11 22②
0102 22 33②
0103 33 ②
0104 44 55②
0105 55 66②
0106 66 77②
0107 77 88②
0108 88 .③

brk : 0>MEM C④

0108 88 99⑤
0109 99 0AA⑥
010A AA 0BB⑥
010B BB 0CC⑥
010C CC .⑦

brk : 0>■

```

- ① メモリの変更開始アドレス
- ② 変更値の指定 (103番地は未変更)
- ③ “. ②”を入力しメモリ内容の変更終了
- ④ メモリ内容の変更開始アドレスを省略
- ⑤ 前回のメモリ内容の変更を終了したアドレス
- ⑥ 変更値の指定
- ⑦ “. ②”を入力しメモリ内容の変更終了

【指定方法】

オペランド

addr : プログラム・メモリの内容を表示したいアドレスまたはアドレス範囲を指定します。

partition : アドレス範囲は、パーティション指定とマスク指定ができます。

パーティション指定の場合は、開始アドレスと終了アドレスを“, ”で区切ります。

アドレス指定範囲 : 0-OFFFFH

マスク指定範囲 : 0-OXXXXH

パーティション指定 : 指定された範囲が表示されます。

開始アドレスのみ指定 : 指定されたアドレスから11行分指定します。

省略時 : 直前に終了したアドレス+1から11行分表示します。

初期値 : 0H

【使用例】

100H番地からのメモリ内容を表示した例です。

```

brk : 0>MEM D 100, 17F①

0100  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F .....
0110  30 31 32 33 34 35 36 38 39 3A 3B 3C 3D 3E 3F 0123456789;<=>?
0120  40 41 42 43 44 45 46 47 48 49 4A 4B 4C 4D 4E 4F @ABCDEFGHIJKLMNO
0130  50 51 52 53 54 55 56 57 58 59 5A 5B 5C 5D 5E 5F PQRSTUVWXYZabcde
      .
      .
      .
0170  20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F !"#%&'()*+,-./

brk : 0>MEM D 108②

0108
0110  30 31 32 33 34 35 36 37 38 39 3A 3B 3C 3D 3E 3F 0123456789;<=>?
      .
      .
      .
01A0  A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 AA AB AC AD AE AF .....

brk : 0>MEM D③

01B0  B0 B1 B2 B3 B4 B5 B6 B7 B8 B9 BA BB BC BD BE BF .....
01C0  C0 C1 C2 C3 C4 C5 C6 C7 C8 C9 CA CB CC CD CE CF .....
01D0  D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 DA DB DC DD DE DF .....
      .
      .
      .
0250  30 31 32 33 34 35 36 37 38 39 3A 3B 3C 3D 3E 3F 0123456789;<=>?

brk : 0>■
    
```

- ① メモリの表示開始/終了アドレス指定
- ② メモリの表示開始アドレスだけの指定
- ③ メモリの表示アドレスを省略

③ プログラム・メモリのテスト

形式3 MEMΔE [Δpartition]					
基数	partition : H				
brk : n >	<input type="radio"/>	emu : n >	<input type="checkbox"/>	trc : n >	<input type="checkbox"/>

【機能】

テスト範囲で指定されたプログラム・メモリのテストを行います。

- (a) テスト範囲を省略した場合は、0-0FFFFH番地のプログラム・メモリ領域をテストします。
- (b) メモリ・テストを実行しても、テストしたメモリのデータは保存されます。
- (c) メモリ・テストで異常を検出した場合、異常を検出したアドレス以降のテストは行いません。

【指定方法】

オペランド

partition : プログラム・メモリのテスト範囲を指定します。

テスト範囲は、パーティション指定と、マスク指定ができます。

パーティション指定の場合は、開始アドレスと終了アドレスを“,”で区切ります。

アドレスの指定範囲 : 0-0FFFFH

マスクの指定範囲 : 0-0XXXXH

省略時 : 0-0FFFFH番地のプログラム・メモリ領域をテストします。

【使用例】

(a) 0H-0FFFH番地のプログラム・メモリのテストを行う例

brk : 0>MEM E 0XXX	①
Complete	②
brk : 0>■	

- ① テストするメモリの番地指定
- ② 結果の表示

(b) 1000H-1FFFH番地のテストをして152AH番地で異常を検出する例

brk : 0>MEM E 1000, 1FFF	①
152A	②
brk : 0>■	

- ① テストするメモリの番地指定
- ② 結果の表示

(4) メモリのイニシャライズ

形式4 MEM△F△ partition △ data-string	
基数	partition : H data-string : H
brk : n>	<input type="radio"/> emu : n> × trc : n> ×

【機能】

イニシャライズ範囲で指定された範囲に、指定されたイニシャライズ・データを設定します。

【指定方法】

オペランド

- partition : メモリのイニシャライズ範囲を指定します。
 イニシャライズ範囲は、パーティション指定と、マスク指定ができます。
 パーティション指定の場合は、開始アドレスと終了アドレスを“,”で区切ります。
 アドレスの指定範囲：0-0FFFFH
 マスクの指定範囲：0-OXXXXH
- data-string : イニシャライズ・データを指定します。
 最大10個までのデータ列を指定でき、データとデータは“,”で区切ります。
 データは、マスク・データの指定ができます。データ列は最大10個までです。

【使用例】

(a) 100H-1FFHまでを1, 2, 3, 4, 5, 6, 7, 8, 9, 0のデータ列でイニシャライズする例

```
brk : 0>MEM F 100, 1FF 1, 2, 3, 4, 5, 6, 7, 8, 9, 0 ①
brk : 0>MEM D 100, 1FF ②

0100 01 02 03 04 05 06 07 08 09 00 01 02 03 04 05 06 .....
0110 07 08 09 00 01 02 03 04 05 06 07 08 09 00 01 02 .....
0120 03 04 05 06 07 08 09 00 01 02 03 04 05 06 07 08 .....
      .
      .
      .
01E0 05 06 07 08 09 00 01 02 03 04 05 06 07 08 09 00 .....
01F0 01 02 03 04 05 06 07 08 09 00 01 02 03 04 05 06 .....

brk : 0>■
```

- ① イニシャライズ範囲, データ指定
- ② イニシャライズ後のメモリ内容の表示指定

(b) マスク・データによるイニシャライズの例

```
brk : 0>MEM F 100, 11F 3X ①
brk : 0>MEM D 100, 13F ②

0100 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 1234567890123456
0110 37 38 39 30 31 32 33 34 35 36 37 38 39 30 31 32 7890123456789012
0120 03 04 05 06 07 08 09 00 01 02 03 04 05 06 07 08 .....
0130 09 00 01 02 03 04 05 06 07 08 09 00 01 02 03 04 .....

brk : 0>■
```

- ① イニシャライズ範囲, マスク・データ指定
- ② イニシャライズ後のメモリ内容表示指定

備考 マスク・データ “3X” でイニシャライズした場合は, この例のように下位4ビットは変化しません。

(5) メモリ内容のサーチ

形式4		MEM△G△ partition △ data-string	
基数	partition : H	data-string : H	
brk : n >	○	emu : n >	×
		trc : n >	×

【機能】

メモリのサーチ範囲で指定された範囲内から、サーチ・データで指定されるデータ列をサーチします。

- (a) データ列を検出した場合は、検出アドレスを表示します。
 サーチ範囲内で複数のデータがあった場合は、すべての検出アドレスを表示します。
- (b) サーチ・データは最大10個までで、マスク指定もできます。

【指定方法】

オペランド

- partition : メモリのサーチ範囲を指定します。
 サーチ範囲は、パーティション指定と、マスク指定ができます。
 パーティション指定の場合は、開始アドレスと終了アドレスを“,”で区切ります。
 アドレスの指定範囲 : 0-OFFFFH
 マスクの指定範囲 : 0-OXXXXH
- data-string : サーチ・データを指定します。
 最大10個までのデータ列を指定でき、データとデータは“,”で区切ります。

【使用例】

100H-1FFHのメモリ内容が次のようなとき、サーチをする例です。

0100	31	32	33	34	35	36	37	38	39	30	31	32	33	34	35	36
0110	37	38	39	30	31	32	33	34	35	36	37	38	39	30	31	32
0120	03	04	05	06	07	08	09	00	01	02	03	04	05	06	07	08
0130	09	00	01	02	03	04	05	06	07	08	09	00	01	02	03	04
0140	05	06	07	08	09	00	01	02	03	04	05	06	07	08	09	00
0150	01	02	03	04	05	06	07	08	09	00	01	02	03	04	05	06
0160	07	08	09	00	01	02	03	04	05	06	07	08	09	00	01	02
0170	03	04	05	06	07	08	09	00	01	02	03	04	05	06	07	08
0180	09	00	01	02	03	04	05	06	07	08	09	00	01	02	03	04
0190	05	06	07	08	09	00	01	02	03	04	05	06	07	08	09	00
01A0	01	02	03	04	05	06	07	08	09	00	01	02	03	04	05	06
01B0	07	08	09	00	01	02	03	04	05	06	07	08	09	00	01	02
01C0	03	04	05	06	07	08	09	00	01	02	03	04	05	06	07	08
01D0	09	00	01	02	03	04	05	06	07	08	09	00	01	02	03	04
01E0	05	06	07	08	09	00	01	02	03	04	05	06	07	08	09	00
01F0	01	02	03	04	05	06	07	08	09	00	01	02	03	04	05	06

```
brk : 0>MEM G 100, 1FF 30, 31, 32, 
  

0109
0113
011D
  

brk : 0>■
```

①
②

- ① 100H-1FFHの範囲から30, 31, 32の連続したデータ列をサーチ
- ② データ列を検出したアドレス

(6) メモリ内容のコピー

形式5		MEM△M△ partition △ addr	
基数	partition : H	addr : H	
brk : n >	<input type="radio"/>	emu : n >	<input type="checkbox"/>
		trc : n >	<input type="checkbox"/>

【機能】

コピー元範囲で指定された範囲のメモリ内容を、コピー先アドレスで指定されるアドレス以降にコピーします。

- (a) コピー元範囲内のアドレスと、コピー先アドレスは重複してもかまいません。
- (b) コピー先アドレスは、1個のアドレスのみ指定ができます。

【指定方法】

オペランド

partition : コピー元範囲を指定します。

コピー元範囲はパーティション指定と、マスク指定ができます。

パーティション指定の場合は、開始アドレスと終了アドレスを“,”で区切ります。

アドレスの指定範囲 : 0-0FFFFH

マスクの指定範囲 : 0-0XXXXH

addr : コピー先アドレスを1個指定します。

アドレスの指定範囲 : 0-0FFFFH

【使用例】

100H-13FHまでを180H以降にコピーする例です。

```

brk : 0>MEM M 100, 13F 180
brk : 0>MEM D 100, 1FF

0100 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 1234567890123456
0110 37 38 39 30 31 32 33 34 35 36 37 38 39 30 31 32 7890123456789012
0120 03 04 05 06 07 08 09 00 01 02 03 04 05 06 07 08 .....
0130 09 00 01 02 03 04 05 06 07 08 09 00 01 02 03 04 .....
0140 05 06 07 08 09 00 01 02 03 04 05 06 07 08 09 00 .....
0150 01 02 03 04 05 06 07 08 09 00 01 02 03 04 05 06 .....
0160 07 08 09 00 01 02 03 04 05 06 07 08 09 00 01 02 .....
0170 03 04 05 06 07 08 09 00 01 02 03 04 05 06 07 08 .....
0180 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 1234567890123456
0190 37 38 39 30 31 32 33 34 35 36 37 38 39 30 31 32 7890123456789012
01A0 03 04 05 06 07 08 09 00 01 02 03 04 05 06 07 08 .....
01B0 09 00 01 02 03 04 05 06 07 08 09 00 01 02 03 04 .....
01C0 05 06 07 08 09 00 01 02 03 04 05 06 07 08 09 00 .....
01D0 01 02 03 04 05 06 07 08 09 00 01 02 03 04 05 06 .....
01E0 07 08 09 00 01 02 03 04 05 06 07 08 09 00 01 02 .....
01F0 03 04 05 06 07 08 09 00 01 02 03 04 05 06 07 08 .....

brk : 0>■
    
```

- ① コピー元範囲，コピー先アドレスの指定
- ② コピー後のメモリ内容の表示指定（③のメモリ内容が④にコピーされたことを確認できる）

(7) メモリ内容の交換

形式5		MEM△X△ partition △ addr	
基数	partition : H	addr : H	
brk : n >	○	emu : n >	×
		trc : n >	×

【機能】

メモリの交換範囲で指定されたメモリ内容と、メモリの交換先アドレスで指定されたアドレス以降のメモリ内容を交換します。

注意1. 交換範囲と交換先アドレスが重複してはいけません。

2. 交換先アドレスは、1個のアドレスのみ指定ができます。

【指定方法】

オペランド

partition : メモリの交換範囲を指定します。

交換範囲は、パーティション指定と、マスク指定ができます。

パーティション指定の場合は、開始アドレスと終了アドレスを“,”で区切り、マスクの指定はできません。

アドレスの指定範囲 : 0-0FFFFH

マスクの指定範囲 : 0-0XXXXH

addr : 交換先アドレスを1個指定します。

アドレスの指定範囲 : 0-0FFFFH

【使用例】

100H-17FHのメモリ内容が次のようになっているとき、メモリの内容を交換する例です。

```

0100 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36
0110 37 38 39 30 31 32 33 34 35 36 37 38 39 30 31 32
0120 03 04 05 06 07 08 09 00 01 02 03 04 05 06 07 08
0130 09 00 01 02 03 04 05 06 07 08 09 00 01 02 03 04
0140 05 06 07 08 09 00 01 02 03 04 05 06 07 08 09 00
0150 01 02 03 04 05 06 07 08 09 00 01 02 03 04 05 06
0160 07 08 09 00 01 02 03 04 05 06 07 08 09 00 01 02
0170 03 04 05 06 07 08 09 00 01 02 03 04 05 06 07 08
    
```

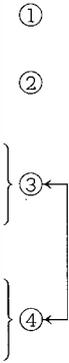
```

brk : 0>MEM X 100, 13F 140␣ ①

brk : 0>MEM D 100, 17F␣ ②

0100 05 06 07 08 09 00 01 02 03 04 05 06 07 08 09 00 .....
0110 01 02 03 04 05 06 07 08 09 00 01 02 03 04 05 06 .....
0120 07 08 09 00 01 02 03 04 05 06 07 08 09 00 01 02 .....
0130 03 04 05 06 07 08 09 00 01 02 03 04 05 06 07 08 .....
0140 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 1234567890123456
0150 37 38 39 30 31 32 33 34 35 36 37 38 39 30 31 32 7890123456789012
0160 03 04 05 06 07 08 09 00 01 02 03 04 05 06 07 08 .....
0170 09 00 01 02 03 04 05 06 07 08 09 00 01 02 03 04 .....

brk : 0>■
    
```



- ① 100H-13FHまでを140H以降と交換
- ② 交換後のメモリ内容の表示指定 (③と④のメモリ内容が交換されたことを確認できる)

⑧ メモリ内容の比較

形式5		MEM△V△ partition △ addr	
基数	partition : H	addr : H	
brk : n >	○	emu : n >	×
		trc : n >	×

【機能】

比較範囲で指定された範囲の内容と、比較先アドレスで指定されたアドレス以降のメモリ内容を比較します。

- 備考1. 比較範囲内のアドレスと、比較先アドレスは重複してもかまいません。
2. 比較した結果、メモリ内容が一致しない場合は、一致しないアドレスとデータを表示します。
 3. 比較は、比較範囲の終了まで続けます。

【指定方法】

オペランド

partition : メモリの比較範囲を指定します。

比較範囲は、パーティション指定と、マスク指定ができます。

パーティション指定の場合は、開始アドレスと終了アドレスを“,”で区切り、マスクの指定はできません。

アドレスの指定範囲 : 0-OFFFFH

マスクの指定範囲 : 0-OXXXXH

addr : 比較先アドレスを1個指定します。

アドレスの指定範囲 : 0-OFFFFH

【使用例】

100H番地以降のメモリ内容が次のようになっているとき、メモリの内容を比較する例です。

```

0100 05 06 07 08 09 00 01 02 03 04 05 06 07 08 09 00
0110 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36
0120 37 38 39 30 31 32 33 34 35 36 37 38 39 30 31 32
0130 03 04 05 06 07 08 09 00 01 02 03 04 05 06 07 08
0140 09 00 01 02 03 04 05 06 07 08 09 00 01 02 03 04
0150 05 06 07 08 09 00 01 02 03 04 05 06 07 08 09 01
0160 31 32 33 34 35 41 37 38 39 30 31 32 33 34 35 36
0170 37 38 39 30 31 32 33 34 35 36 37 38 39 30 31 32
0180 03 04 05 06 07 08 09 00 01 02 03 04 05 06 07 08
0190 09 00 01 02 2F 04 05 06 07 08 09 00 01 02 03 04
    
```

```

brk : 0> MEM V 100, 14F 150
    ①
    010F 00 015F 01
    0115 36 0165 41
    0144 03 0194 2F
    ②
brk : 0> ■
    
```

- ① 100H-14FHまでを150H以降と比較
- ② 比較した結果の表示

【指定方法】

オペランド

MODE : ハンドシェーク・モードの選択を行います。

CHAR : ハードウェア・ハンドシェーク・モード

FLOW : ソフトウェア・ハンドシェーク・モード

初期値 : ソフトウェア・ハンドシェーク・モード

BAUD : ボー・レートを, 以下の中から1個だけ選択を行います。

19200, 9600, 4800, 2400, 1200, 600, 300

初期値 : 9600

LONG : キャラクタ長の選択を行います。

7 : 7ビット・キャラクタ長

8 : 8ビット・キャラクタ長

初期値 : 8ビット・キャラクタ長

PAR : パリティ・ビットの選択を行います。

NON : パリティ・ビットなし

EVEN : 偶数パリティ・ビット

ODD : 奇数パリティ・ビット

初期値 : パリティ・ビットなし

STOP : ストップ・ビット長の選択を行います。

1 : ストップ・ビット長1

2 : ストップ・ビット長2

初期値 : ストップ・ビット長2

省略 : 動作状態の設定を対話形式で行います。

【使用例】

(a) チャネル2モード設定

MODE, BAUD, LONG, PAR, STOPの指定を行う例です。

```
brk:0>MOD MODE=CHAR BAUD=4800 LONG=8 PAR=NON STOP=2
brk:0>■
```

①

- ① 1キャラクタ・ハンドシェイク，ボー・レートは，4800ボー，キャラクタ長は8ビット，パリティ・ビットなし，ストップ・ビットは2ビットに設定

(b) チャネル2モード設定（対話形式）

オペランドを省略すると動作状態の設定を対話形式で行うことができます。

```
brk:0>MOD
Mode Char=FLOW
Baud 4800=9600
Long 8=
Par NON=EVEN
Stop 2=1
brk:0>■
```

①

②

③

④

⑤

⑥

- ① 対話形式でチャネル2の動作状態を設定
 ② バッファ制御モードに変更
 ③ ボー・レートを9600ボーに変更
 ④ キャラクタ長は変更しない
 ⑤ 偶数パリティ・ビットに変更
 ⑥ ストップ・ビット長を1に変更

8.4.24 外部センス・クリップ・モード設定コマンド (OUT)

入力形式

形式1	OUT [Δ OFF]
形式2	OUT [Δ ON Δ \$T]
形式3	OUT [Δ ON Δ BRA? [Δ BRA?]]

【機能】

OUTコマンドは、外部センス・クリップのモードを設定します。外部センス・クリップは次に示す使い方ができます。

(a) OFF

外部センス・クリップの8本すべてを入力して使用します。

(b) ON \$T

外部センス・クリップのうちの7本（ビット1-ビット7）を入力，1本（ビット0）を外部イベント出力として使用します。

(c) ON BRA? [BRA?] BRA? : BRA3 OR BRA4

外部センス・クリップを4本ずつの2組に分け（ビット0-ビット3，ビット4-ビット7），BRA3および，BRA4で設定されたデータ・メモリ・アドレスへのアクセス・データの出力に使用します。

注意 外部センス・クリップのモード設定コマンドは、外部センス・クリップをどのように使用するかを設定するだけであり、外部センス・クリップを接続した端子の用途が、設定と合っているかのチェックは行いません。したがって、接続には十分な注意が必要です。なお、外部センス・クリップ入出力レベルは、TTLレベルです。

(1) 外部センス・クリップ入力モード設定

形式1	OUT [Δ OFF]
-----	---------------------

brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="checkbox"/>	×
-----------	-----------------------	-----------	-----------------------	-----------	--------------------------	---

【機能】

外部センス・クリップの8本すべてを入力として使用するよう設定します。

- (a) 入力値はトレーサに書き込まれますので、TRDコマンドにより値の変化をトレースすることができます。
- (b) イベント・レジスタの設定時に、外部信号レベルを設定することにより、イベント条件として使用することができます。

【指定方法】

オペランド

“OFF” と指定します。

省略時：現在の外部センス・クリップのモードが表示されます。

② 外部センス・クリップ入出力モード設定

形式2	OUT [△ON△\$T]
-----	---------------

brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="checkbox"/>
-----------	-----------------------	-----------	-----------------------	-----------	--------------------------

【機能】

外部センス・クリップのうちの7本（ビット1-ビット7）を入力として使用し、1本（ビット0）を出力として使用するよう設定します。

- (a) 入力値はトレーサに書き込まれますので、TRDコマンドにより、値の変化をトレースすることができます。
- (b) 出力（ビット0）は外部イベント出力として使用することができます^注。

注 このとき、ビット0はオープン・コレクタ出力となるため、プルアップ抵抗を接続してください。

【指定方法】

(a) オペランド1

“ON”と指定します。

(b) オプション

“\$T”と指定します。

③ 外部センス・クリップ出力モード設定

形式3 OUT [△ON△BRA? [△BRA?]]

brk : n >	○	emu : n >	○	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

外部センス・クリップを4本ずつの2組（ビット0-ビット3, ビット4-ビット7）に分け、BRA3および、BRA4で指定されたデータ・メモリ・アドレスへのアクセス・データの出力に使用するように設定します。

- (a) 外部センス・クリップとBRAの対応を次に示します。

BRA3 : ビット4-ビット7

BRA4 : ビット0-ビット3

- (b) BRA3, BRA4がOUTコマンド以外のコマンド (BRM, TRX, TRY, CHK) で設定されている場合、OUTコマンドではBRA3, BRA4を指定できません。逆に、BRA3, BRA4がOUTコマンドで設定されている場合、他のコマンドではBRA3, BRA4を指定できません。

【指定方法】

- (a) オペランド1

"ON" と指定します。

- (b) オペランド2

BRA? : BRA3またはBRA4を指定します。省略はできません。

【使用規約】

本コマンドを実行する前に、BRA3, BRA4の設定を行っておく必要があります。

OUTコマンドのEXTクリップの初期値は、OUTコマンドを設定した時点の指定メモリの値になります。

【使用例】

(a) 外部センス・クリップの出力モードの表示

オペランドを省略すると現在の設定内容が表示されます。

brk : 0 > <u>OUT</u> <input type="checkbox"/>	①
OFF	
brk : 0 > <u>OUT</u> <input type="checkbox"/>	②
ON \$T	
brk : 0 > <u>OUT</u> <input type="checkbox"/>	③
ON BRA3	

- ① OFFが設定されている場合
- ② ONで\$Tが設定されている場合
- ③ ONでBRA3が設定されている場合

(b) 外部センス・クリップのモード指定

入力, 入出力, 出力の各モード指定を行う例です。

brk : 0 > OUT OFF <input type="checkbox"/>	①
WARNING ALL BIT INPUT MODE	
brk : 0 > OUT ON \$T <input type="checkbox"/>	②
WARNING BIT0 OUTPUT MODE	
brk : 0 > OUT ON BRA3 <input type="checkbox"/>	③
WARNING BIT4-7 OUTPUT MODE	
brk : 0 > OUT ON BRA4 <input type="checkbox"/>	④
WARNING BIT0-3 OUTPUT MODE	
brk : 0 > OUT ON BRA3 BRA4 <input type="checkbox"/>	⑤
WARNING ALL BIT OUTPUT MODE	
brk : 0 > OUT ON BRA4 <input type="checkbox"/>	⑥
USE BRA4 OTHERS COMMAND (BRM, TRY, CHIC)	

- ① 入力モードを指定する場合
- ② 入出力モードを指定する場合
- ③ 出力モードでBRA3を指定する場合
- ④ 出力モードでBRA4を指定する場合
- ⑤ 出力モードでBRA3とBRA4を指定する場合
- ⑥ BRA4がすでに他のコマンドで使用されている場合

8.4.25 パス回数設定コマンド (PAS)

入力形式	
形式1	PAS [Δ count]
基数	count : T
brk : n >	<input type="radio"/>
emu : n >	<input type="radio"/>
trc : n >	<input type="checkbox"/>
	×

【機能】

PASコマンドは、イベント・モード・レジスタ (BRM) で統合されたイベント検出の、パス回数を設定します。オペランド省略時は、現在の設定値を表示します。

【指定方法】

オペランド

count : パス回数を10進で指定します。

パス回数範囲 : 1-255

初期値 : 1

省略時 : 現在の設定値を表示します。

【使用例】

パス回数を5回に設定する例です。

brk : 0 > PAS 5 <input type="checkbox"/>	①
brk : 0 > PAS <input type="checkbox"/>	②
5	③
brk : 0 > ■	

- ① パス回数の指定
- ② パス回数を省略
- ③ 現在の設定値を表示

8.4.26 端末モード設定コマンド (PGM)

入力形式	
形式1	PGM
形式2	PGM△C

【機能概要】

PGMコマンドは、IE-75001-RのCH2に接続したPROMプログラマ (PG-1000, PG-1500) を遠隔操作するために、モード設定します。

備考 MODコマンドを使用したPGシリーズの接続方法 (4.2 PGシリーズとの接続および、各PGシリーズのユーザーズ・マニュアルも参照してください。)

(1) PG-1500との接続

- ① IE-75001-Rのチャンネル2 (CH2) とPG-1500をRS-232-Cのストレート・ケーブルで接続します。
- ② IE-75001-Rのチャンネル2はMODコマンド, PG-1500はMODEキーにより, RS-232-Cのインタフェースを設定します。
- ③ IE-75001-Rの側面にあるRS-232-C設定部のフタを開け, CH2用モデム/ターミナル・モード・セレクトのスライド・スイッチをターミナル・モード側に設定します。
- ④ 次のキーの順で, PG-1500のパネル操作を行います。

(Power ON)→RESET→FUNCTION→REMOTE→SET

(2) PG-1000との接続

- ① IE-75001-Rのチャンネル2 (CH2) とPG-1000をRS-232-Cのストレート・ケーブルで接続します。
- ② IE-75001-Rのチャンネル2はMODコマンド, PG-1000はMODEキーにより, RS-232-Cのインタフェースを設定します。
- ③ IE-75001-Rの側面にあるRS-232-C設定部のフタを開け, CH2用モデム/ターミナル・モード・セレクトのスライド・スイッチをターミナル・モード側に設定します。
- ④ 次のキーの順で, PG-1000のパネル操作を行います。

(Power ON)→RESET→DEVICE SELECT→A-ENT→SERIAL

(1) 端末モード設定

形式1	PGM
-----	-----

brk : n >	○	emu : n >	×	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

IE-75001-RのCH2とPROMプログラマ（PG-1000, PG-1500）との通信制御を行います。

本コマンドを実行すると、PGMモードになります。CH2とPG-1000, PG-1500を接続することにより、PG-1000, PG-1500からオブジェクト・コードのアップ/ダウン・ロードができるようになります。PGMモードを終了するときは、“^Z”を入力します。

【指定方法】

コマンド本体のみ入力します。

【使用例】

(a) PG-1500と接続する場合/切り離す場合

```

brk : 0 > PGM  ①
Beginning of PGM mode
PG > ②
.
.
.
PG > CTRL Z ③
Exit PGM mode (Y/N)  ④
brk : 0 > ■ ⑤
    
```

- ① PGMコマンドにより、PG-1500と接続します。
- ② PG-1500と接続された状態（“PG>”の表示）で、PG-1500のコマンド^注により、操作を行います。
- ③ } “^Z (**CTRL**+Z)”と“Y^注”の入力により、PG-1500と切り離されます。
- ④ }
- ⑤ PGMモードから切り離されます。

注 PG-1500を接続した場合の操作コマンドは次頁のとおりです。

表 8-3 PG-1500のコマンド一覧

コマンド	形 式	機 能
RR	PG>RR R_S_ADR, R_E_ADR, PG_S_ADR, CONV	PROMの内容のリード
RS	PG>RS sub sub=C/R/A	デバイスの選択
RV	PG>RV R_S_ADR, R_E_ADR, PG_S_ADR, CONV	PROMとPG内メモリとの比較
RW	PG>RW R_S_ADR, R_E_ADR, PG_S_ADR, CONV	PROMへの書き込み
RZ	PG>RZ	PROM消去チェック
MC	PG>MC PG_S_ADR	PG内メモリ内容の変更
MD	PG>MD PG_S_ADR, PG_E_ADR	PG内メモリ内容の表示
MF	PG>MF PG_S_ADR, PG_E_ADR, INT_DATA	PG内メモリの初期化
LI	PG>LI	IE-75001-RからPGへのデータ転送
SI	PG>SI PG_S_ADR, PG_E_ADR	PGからIE-75001-Rへのデータ転送
??	PG>??	コマンド・ヘルプ

詳細は、PG-1500のユーザーズ・マニュアルを参照してください。

- 備考
- ・ PG_S_ADR : PGスタート・アドレス
 - ・ PG_E_ADR : PGエンド・アドレス
 - ・ R_S_ADR : PROMスタート・アドレス
 - ・ R_E_ADR : PROMエンド・アドレス
 - ・ CONV : アドレス分割指定
 - ・ INT_DATA : 初期化データ

(b) PG-1500とオブジェクト・コードのアップ・ロードの方法 (LIコマンド)

IE-75001-Rのマッピングされたメモリ空間の内容をPG-1500のバッファへ転送します。転送範囲を省略した場合は、マッピングされている全メモリ空間を転送します。途中で転送を中止する場合は、〈ESC〉キーを入力します。

```
PG>LI①
    Partition=YYYY, ZZZZ②
PG>■
```

① YYYYには、IE-75001-Rメモリの転送開始アドレスを、ZZZZにはIE-75001-Rのメモリ転送終了アドレスを入力します。

(c) PG-1500とオブジェクト・コードのダウン・ロードの方法 (SIコマンド)

PG-1500のバッファの内容をIE-75001-Rのマッピングされたメモリ空間に転送します。IE-75001-Rのロード・バイアスは省略できません。途中で転送を中止する場合は〈ESC〉キーを入力します。

```
PG>SI XXXX, YYYY①
    Bias=ZZZZ②
    complete
PG>■
```

① XXXXには、PG-1500のバッファの転送開始アドレスを、YYYYはPG-1500のバッファの転送終了アドレスを入力します。

② ZZZZには、IE-75001-Rのロード・バイアスを入力します。

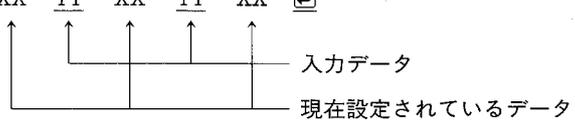
(d) PG-1000と接続する場合／切り離す場合

brk : 0>PGM	①
Beginning of PGM mode	
*I	②
*■	③
·	
·	
* CTRL Z	④
Exit PGM mode (Y/N) Y	⑤
brk : 0>■	⑥

- ① PGMコマンドにより、PG-1000と接続します。
- ② “I” の入力により、PG-1000とのインテリジェント・モードに移行します。なお、“I”の文字は表示されません。
- ③ この状態 (“*” の表示) で、PG-1000のコマンド注の入力により、操作を行います。
- ④ } “Z (CTRL+Z)” と “Y” の入力により、PG-1000と切り離されます。
- ⑤ }
- ⑥ PGMモードから切り離されます。

注 PG-1000を接続した場合の操作コマンドは次頁のとおりです。

表 8-4 PG-1000のコマンド一覧

コマンド	形式	コマンドの機能
A	* <u>A</u> s, e, r _☐	パラメータの設定
E	* <u>E</u> r _☐	PGのバッファのデータを変更 (形式は次のとおりです) * <u>E</u> r _☐ r XX- YY- XX- YY- XX- ☐  データ入力形式 ・データを入力する (16進以外を入力するとその時点で, "?" を表示) ・スペース・キーの入力 (データ変更なし)
F	* <u>F</u> r, re, d _☐	PGのバッファをdで初期化
I	* <u>I</u> _☐	インテリジェント・モード (ディスプレイヘエコー・バックするモード) へ移行
O	* <u>O</u> r, re _☐	PGのバッファの内容を表示 表示形式→r 00 00 00 00 00 00 …… 00
R	* <u>R</u> r, re _☐	PROMの内容をPGのバッファへ転送
S	* <u>S</u> _☐	PROMの選択
T	* <u>T</u> _☐	トランジェント・モード (ディスプレイヘエコー・バックしないモード) へ移行
V	* <u>V</u> s, e, r _☐	PROMの内容とPGのバッファの内容を比較する →データが違うときは "?" が表示される
W	* <u>W</u> s, e, r _☐	PGのバッファの内容をPROMに書き込む →書き込みエラー時は "?" が表示される
Y	* <u>Y</u> _☐	現在設定されているパラメータを表示
Z	* <u>Z</u> _☐	PROMにデータが書き込まれているかどうかをチェック →データが書き込まれている場合にのみ "?" が表示される (データがない場合は, 何も表示されない)
L	* <u>L</u> bias _☐ partition= i, ie _☐	IE-75001-Rの, マッピングされているi番地からie番地までのデータを, PGのバッファのi+bias番地からie+bias番地まで転送する
P	* <u>P</u> r, re _☐ Bias=i _☐	PGのバッファのr番地からre番地までのデータを, IE-75001-Rのマッピングされているr+i番地からre+l番地まで転送する

詳細は, PG-1000 ユーザーズ・マニュアルを参照してください。

- 備考
- ・ s : PROM開始番地
 - ・ e : PROM終了番地
 - ・ r : PGバッファ開始番地
 - ・ r e : PGバッファ終了番地
 - ・ i : IE-75001-R開始番地
 - ・ i e : IE-75001-R終了番地
 - ・ d : データ

注意 エラー発生条件

s>e

r>re

i>ie

16進以外を入力

(e) PG-1000とオブジェクト・コードのアップ・ロードの方法 (Lコマンド)

IE-75001-Rのマッピングされたメモリ空間の内容をPG-1000バッファへ転送します。転送範囲を省略した場合は、マッピングされている全メモリ空間を転送します。途中で転送を中止する場合は、〈ESC〉キーを入力します。

<pre>*LXXX␣</pre>	①
<pre>Partition=YYYY, ZZZZ␣</pre>	②
<pre>*■</pre>	

- ① XXXには、PG-1000のロード・バイアスを入力します。
- ② YYYYには、IE-75001-Rメモリの転送開始アドレスを、ZZZZはIE-75001-Rのメモリの転送終了アドレスを入力します。

(f) **PG-1000とオブジェクト・コードのダウン・ロードの方法 (Pコマンド)**

PG-1000のバッファの内容をIE-75001-Rのマッピングされたメモリ空間に転送します。IE-75001-Rのロード・バイアスは省略できません。途中で転送を中止する場合は〈ESC〉キーを入力します。

<pre>*PXXXX, YYYY␣</pre> <pre> Bias=ZZZZ␣</pre> <pre> complete</pre> <pre>*■</pre>	<p>①</p> <p>②</p>
--	-------------------

- ① XXXXには、PG-1000のバッファの転送開始アドレスを、YYYYにはPG-1000のバッファの転送終了アドレスを入力します。
- ② ZZZZには、IE-75001-Rのロード・バイアスを入力します。

② カレント制御キャラクタ変更

形式2 PGM△C

brk : n >	○	emu : n >	×	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

PGMモードにおける制御キャラクタを対話形式で任意のキャラクタに変更することができます。

(a) 任意のキャラクタとして次の16種類のキャラクタが使用できます。

^ A (CTRL)+A (01H)
 ^ B (CTRL)+B (02H)
 ^ E (CTRL)+E (05H)
 ^ F (CTRL)+F (06H)
 ^ G (CTRL)+G (07H)
 ^ N (CTRL)+N (0EH)
 ^ O (CTRL)+O (0FH)
 ^ P (CTRL)+P (10H)
 ^ R (CTRL)+R (12H)
 ^ T (CTRL)+T (14H)
 ^ U (CTRL)+U (15H)
 ^ V (CTRL)+V (16H)
 ^ W (CTRL)+W (17H)
 ^ X (CTRL)+X (18H)
 ^ Y (CTRL)+Y (19H)
 ^ Z (CTRL)+Z (1AH)

(b) 制御キャラクタの変更に問題がなければPGMモードになります。制御キャラクタ変更後はPGMコマンドを実行する必要はありません。

(c) 同一のキャラクタを複数の機能の制御キャラクタとして設定することはできません。

(d) キーあるいは“^H”入力の場合は、初期値に変換します。

(e) <ESC>キー入力による中断の場合は、それまでに変換したキャラクタを無効にします。

【指定方法】

オペランド

“C” を指定します。

【使用例】

(a) キャラクタの変更例

キャラクタの変更方法および`DEL`キーの使い方などを説明します。

```
brk : 0>PGM C  ①

Termination of "PGM"      ... ^Z
Beginning of  "HEX LOAD"   ... ^A
Beginning of  "HEX SAVE"   ... ^E
Termination of "SYM LOAD"  ... ^N
Termination of "LOAD"      ... ^B
Termination of "SAVE"      ... ^F
Break of     "LOAD/SAVE"   ... ^W

Termination of "PGM"
      :
Beginning of  PGM mode ④
```

① カレント制御キャラクタ変更コマンド

② 初期値を表示

③ Termination of "PGM" の現在、設定されている値を表示し、入力待ちになります。設定を変更したい場合は、"`CTRL`+設定したいキャラクタ"を入力し、`Enter`で確定します。また、初期値の設定にしたいときは`DEL`キー（または"`CTRL`+H"）を入力後、`Enter`で確定します。確定後、画面は新しく設定されたキャラクタを表示します。なお、変更の必要がない場合は`Enter`のみ入力します。

Termination of "PGM" の設定が終了すると、次のBeginning of "HEX LOAD" が、画面の一番下に出てきます。最初の行と同じように設定してください。Beginning of "HEX LOAD" の設定が終了すると、次のBeginning of "HEX SAVE" が画面の一番下に出てきます。

このように1つの設定が終了すると次の行が出てきます。引き続き、Termination of "SYM LOAD", Termination of "LOAD", Termination of "SAVE", Break of "LOAD/SAVE" を設定してください。

④ 入力に問題がない場合、Beginning of PGM modeというメッセージを表示し、PGMモードになります。

注意 ③の設定の途中に`ESC`キーで中断すると、変更した部分は無効となります。

(b) 誤って、同一のキャラクタを複数の制御キャラクタとして設定した場合の例

```

brk : 0>PGM C①

Termination of "PGM"      ... ^Z
Beginning   of "HEX LOAD" ... ^A
Beginning   of "HEX SAVE" ... ^E
Beginning   of "SYM LOAD" ... ^N
Termination of "LOAD"     ... ^B
Termination of "SAVE"     ... ^F
Break      of "LOAD/SAVE" ... ^W

Termination of "PGM"      ... ^A②
      :
Termination of "PGM"      ... ^A---Multi define
Beginning   of "HEX LOAD" ... ^Z
Beginning   of "HEX SAVE" ... ^W
Beginning   of "SYM LOAD" ... ^E
Termination of "LOAD"     ... ^A---Multi define
Termination of "SAVE"     ... ^N
Break      of "LOAD/SAVE" ... ^B

Termination of "PGM"      ... ^A■④
    
```

- ① カレント制御キャラクタ変換コマンド
- ② デフォルト値を表示
- ③ 制御キャラクタの設定（前項参照）。
- ④ キャラクタの設定に間違いがあると（ここでは、"**CTRL**+A"を2箇所設定しています）、PGMモードにならずにキャラクタ変更の入力待ちの状態に戻ります。正しく設定し直してください。

8.4.27 データ・メモリ操作コマンド (RAM)

入力形式	
形式1	$\text{RAM}\Delta\text{C} \left[\left\{ \begin{array}{l} \Delta\text{addr} \left[\left\{ \begin{array}{l} \Delta \$\text{B} \\ \Delta \$\text{N} \end{array} \right\} \right] \\ \Delta\text{addrb} \end{array} \right\} \right]$
形式2	$\text{RAM} [\Delta\text{D} \left[\left\{ \begin{array}{l} \left\{ \begin{array}{l} \Delta\text{addr} \\ \Delta\text{partition} \end{array} \right\} \left\{ \begin{array}{l} \Delta\text{B} \\ \Delta\text{N} \end{array} \right\} \\ \Delta\text{addrb} \end{array} \right\} \right]]$
形式3	$\text{RAM}\Delta\text{E} [\Delta\text{partition}]$
形式4	$\text{RAM} \left\{ \begin{array}{l} \Delta\text{F} \\ \Delta\text{G} \end{array} \right\} \Delta\text{partition}\Delta\text{data-string} \left[\left\{ \begin{array}{l} \Delta \$\text{B} \\ \Delta \$\text{N} \end{array} \right\} \right]$
形式5	$\text{RAM} \left\{ \begin{array}{l} \Delta\text{M} \\ \Delta\text{X} \\ \Delta\text{V} \end{array} \right\} \Delta\text{partition}\Delta\text{addr} \left[\left\{ \begin{array}{l} \Delta \$\text{B} \\ \Delta \$\text{N} \end{array} \right\} \right]$
形式6	$\text{RAM}\Delta\text{L}$
形式7	$\text{RAM}\Delta\text{S} [\Delta\text{partition}] [, \text{partition}] \cdots [, \text{partition}]$
基数	addr : H addrb : H partition : H data-string : H

【機能概要】

RAMコマンドは、データ・メモリの変更、表示、テスト、イニシャライズ、サーチ、コピー、交換、比較、ロード、セーブを行います。

【使用規約】

- (a) コマンドの中で指定できるアドレスの範囲は、対象デバイスにより異なります。ディバグ対象デバイスの設定は“STS”コマンドで行います。
- (b) “RES”コマンドでエミュレーション・デバイスをリセットすると、データ・メモリの内容は不定になります。この動作は、対象デバイスと同じですが、注意が必要です。

【使用例】

(a) バイト型でデータ・メモリの内容を変更する場合

データ・メモリの内容を0100H番地よりバイト単位で変更する例です。

```

brk:0>RAM C 0100 $B ①
0100 00 =11 ②
0102 11 =22 ②
0104 22 =33 ②
0106 33 =44 ②
0108 44 =55 ②
010A 55 =. ③

brk:0>RAM C $B ④
010A 55 =66 ⑤
010C 66 =77 ⑤
010E 77 =88 ⑤
0110 88 =. ⑥

brk:0>■
    
```

- ① メモリ内容変更コマンドの指定
- ② 変更値の入力
- ③ “. ”を入力してメモリ内容の変更終了
- ④ 変更開始アドレス省略指定
- ⑤ 直前の変更終了アドレス (パワーオン時は0番地)
- ⑥ “. ”を入力してメモリ内容の変更終了

(b) ニブル型でデータ・メモリ内容を変更する場合

データ・メモリの内容を0100H番地よりニブル単位で変更する例です。

```

brk:0>RAM C 0100 $N ①
0100 1 = 2 ②
0101 1 = 2 ②
0102 2 = 3 ②
0103 2 = 3 ②
0104 3 = 4 ②
0105 3 = . ③

brk:0>RAM C $B ④
0105 3 = 4 ⑤
0106 4 = 5 ⑤
0107 4 = 5 ⑤
0108 5 = . ⑥

brk:0>■

```

- ① コマンドの指定
- ② 変更する値を入力
- ③ “. ”を入力してメモリ内容の変更終了
- ④ 変更開始アドレス, およびオプション省略指定
- ⑤ 直前の変更終了アドレス (パワーオン時は0番地)
- ⑥ “. ”を入力してメモリ内容の変更終了

(c) 指定された番地の内容をビット単位で変更する場合

データ・メモリの0100H番地をビット1の値を変更する例です。

```

brk:0>RAM C 100. 1 ①
0100.1 1 = 0 ②

brk:0>■

```

- ① コマンドの指定
- ② 変更する値を入力

(2) データ・メモリ内容の表示

形式2	RAM	[Δ D	[$\left\{ \begin{array}{l} \Delta\text{addr} \\ \Delta\text{partition} \end{array} \right\} \left[\left\{ \begin{array}{l} \Delta \$B \\ \Delta \$N \end{array} \right\} \right]$]]
基数	addr : H	partition : H	addrb : H			
brk : n>	○	emu : n>	×	trc : n>	×	

【機能】

データ・メモリの表示アドレスとして指定されたアドレスの内容を、指定された型（バイト，ニブル）で表示します。また，データ・メモリ内容は，ビット単位でも表示することができます。

【指定方法】

オペランド

addr partition : データ・メモリ内容を表示したいアドレスまたはアドレス範囲を指定します。

アドレス範囲は，パーティション指定とマスク指定ができます。

パーティション指定の場合は，開始アドレスと終了アドレスを“,”で区切ります。

アドレス指定範囲 : 対象デバイスにより異なります。

開始アドレスのみ指定 : 指定されたアドレスから11行分表示します。

省略時 : 直前に設定されたアドレス+1から11行分表示します。

初期値 : 0H

\$B : バイト型で表示します。

この場合，アドレスは偶数アドレスしか指定できません。

\$N : ニブル型で表示します（省略時）。

addrb : データ・メモリを表示したいアドレスと，ビット番号を指定します。

ビット属性のシンボルは指定できません。

形式 : アドレス,ビット番号

アドレスは指定範囲は対象デバイスにより異なります。

ビットは0-3

【使用例】

(a) バイト型でデータ・メモリの内容を表示する場合

データ・メモリの内容を0100H番地よりバイト単位で表示する例です。

```
brk : 0>RAM D 100,17F $BQ

ADDR  +0 +2 +4 +6 +8 +A +C +E +10+12+14+16+18 +1A+1C+1E
0100   00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F .....
0120   .....
.
.
0160   .....

brk : 0>■
```

(b) ニブル型でデータ・メモリの内容を表示する場合

データ・メモリの内容を0100H番地よりニブル単位で表示する例です。

```
brk : 0>RAM D 100,17F $NQ

ADDR  +0+1+2+3+4+5+6+7+8+9+A+B+C+D+E+F
0100   0 0 1 0 2 0 3 0 4 0 5 0 6 0 7 0
0110   .....
0120
0130
0140
0150
0160
0170   .....

brk : 0>■
```

(c) 指定された番地の内容をビット単位で表示する場合

データ・メモリの0100H番地をビット1の値を表示する例です。

```
brk : 0>RAM D 100. 1Q

0100. 1 0

brk : 0>■
```

③ データ・メモリ内容のテスト

形式3		RAM		△E [△partition]	
基数	partition : H				
brk : n >	○	emu : n >	×	trc : n >	×

【機能】

テスト範囲で指定されたデータ・メモリの内容をテストします。

テスト範囲を省略した場合は、対象デバイスの全データ・メモリ領域をテストします。

【指定方法】

オペランド

partition : データ・メモリのテスト範囲を指定します。

テスト範囲は、パーティション指定と、マスク指定ができます。

パーティション指定の場合は、開始アドレスと終了アドレスを“,”で区切ります。

アドレス指定範囲：対象デバイスにより異なります。

省略時 : 対象デバイスの全データ・メモリ領域をテストします。

初期値 : OH

【使用規約】

- (a) メモリ・テストをしても、テストしたメモリのデータは保存されます。
- (b) メモリ・テストで異常を検出した場合、異常を検出したアドレス以降のテストは行いません。

【使用例】

- (a) データ・メモリの0-100H番地をテストする例

```
brk:0>RAM E 0, 100
```

```
Complete
```

```
brk:0>■
```

- (b) パーティションを省略して対象デバイスの全データ・メモリ領域をテストする例

```
brk:0>RAM E
```

```
110
```

```
brk:0>■
```

①

②

- ① 対象デバイスの全データ・メモリのテスト
- ② 異常を検出したアドレスを表示

備考 テストは、異常を検出したアドレスで中止されます。

(4) データ・メモリ内容のイニシャライズ

形式4		RAM△F△partition△string-data	$\left\{ \begin{array}{l} \Delta \$B \\ \Delta \$N \end{array} \right\}$
基数	partition : H	data-string : H	
brk : n>	○	emu : n>	×
		trc : n>	×

【機能】

イニシャライズ範囲として指定されたアドレス範囲に、指定された型で、指定されたデータ列を設定します。

【指定方法】

オペランド

- partition : データ・メモリのイニシャライズ範囲を指定します。
イニシャライズ範囲は、パーティション指定と、マスク指定ができます。
パーティション指定の場合は、開始アドレスと終了アドレスを“,”で区切ります。
アドレス指定範囲：対象デバイスにより異なります。
- data-string : イニシャライズ・データを指定します。
最大10個までのデータを指定でき、データとデータは“,”で区切ります。
データは、マスク・データの指定ができます。
- \$B : バイト型でイニシャライズを行います。
この場合、アドレスは偶数アドレスしか指定できません。
- \$N : ニブル型でイニシャライズを行います (省略時)。
この場合、イニシャライズ・データのマスク指定はできません。

【使用例】

データ・メモリの100H-1FFH番地を0-9までのデータ列で初期化する例です。

```
brk : 0>RAM F 100, 1FF 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, □ ①
brk : 0>■
```

- ① partition : 100H-1FFH番地
データ列 : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
型 : 省略 (ニブル型)

(5) データ・メモリのサーチ

形式4	RAM△G△partition△data-string[$\left\{ \begin{array}{l} \Delta \$B \\ \Delta \$N \end{array} \right\}$]
基数	partition : H data-string : H

brk : n >	○	emu : n >	×	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

サーチ範囲で指定された範囲内から、指定された型でデータ列をサーチします。

- (a) データ列を検出した場合は、検出アドレスを表示します。サーチ範囲内で複数のデータがあった場合は、すべての検出アドレスを表示します。
- (b) サーチ・データは最大10個までで、\$B指定のときだけマスク指定もできます。

【指定方法】

オペランド

- partition : データ・メモリのサーチ範囲を指定します。
 サーチ範囲は、パーティション指定と、マスク指定ができます。
 パーティション指定の場合は、開始アドレスと終了アドレスを“,”で区切ります。
 アドレス指定範囲：対象デバイスにより異なります。
- data-string : 最大10個までのデータを指定でき、データとデータは“,”で区切ります。
 データは、\$Bのときだけマスク・データの指定ができます。
- \$B : バイト型でサーチを行います。
 この場合、アドレスは偶数アドレスしか指定できません。
- \$N : ニブル型でサーチを行います (省略時)。

【使用例】

データ・メモリの100H-1FFH番地から0, 1, 2, 3, 4, 5, 6, 7, 8, 9のデータ列をサーチする例です。

```
brk : 0> RAM G 100 1FF 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 ①  
  
0109  
0113 ②  
011D  
  
brk : 0> ■
```

- ① partition : 100H-1FFH番地
data : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
型 : 省略 (ニブル型)
- ② データ列を検出したアドレスを表示

(6) データ・メモリのコピー

形式5	RAM△M△partition△addr[{ △\$B } { △\$N }]
基数	partition : H addr : H

brk : n >	○	emu : n >	×	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

コピー元範囲で指定された範囲を、コピー先アドレスから指定された型でコピーします。

- (a) コピー元範囲内のアドレスと、コピー先アドレスは重複してもかまいません。
- (b) コピー先アドレスは、1個のアドレスのみ指定ができます。

【指定方法】

オペランド

partition : コピー元範囲を指定します。

コピー元範囲は、パーティション指定と、マスク指定ができます。

パーティション指定の場合は、開始アドレスと終了アドレスを“,”で区切ります。

アドレス指定範囲：対象デバイスにより異なります。

addr : コピー先アドレスを1個指定します。

アドレス指定範囲：対象デバイスにより異なります。

\$B : バイト型でコピーを行います。

この場合アドレスは、偶数アドレスしか指定できません。

\$N : ニブル型でコピーを行います (省略時)。

【使用例】

データ・メモリの100H-13FH番地を、180H番地以降へコピーする例です。

brk : 0 > RAM M 100, 13F 180	①
brk : 0 > ■	

- ① partition : 100H-13FH番地
- data : 180H番地
- 型 : 省略 (ニブル型)

(7) データ・メモリの交換

形式5	RAM△X△partition△addr	[{ Δ\$B } { Δ\$N }]
基数	partition : H	addr : H

brk : n >	○	emu : n >	×	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

交換範囲として指定された範囲の内容と、交換先アドレスとして指定されたアドレス以降の内容を交換します。

- 注意1. 交換範囲と交換先アドレスが重複してはいけません。
- 2. 交換先アドレスは、1個のアドレスのみ指定できます。

【指定方法】

オペランド

partition : 交換範囲を指定します。

交換範囲は、パーティション指定と、マスク指定ができます。

パーティション指定の場合は、開始アドレスと終了アドレスを “,” で区切ります。

アドレス指定範囲：対象デバイスにより異なります。

addr : 交換先アドレスを1個指定します。

アドレス指定範囲：対象デバイスにより異なります。

\$B : バイト型でデータ交換を行います。

この場合、アドレスは偶数アドレスしか指定できません。

\$N : ニブル型でデータ交換を行います (省略時)。

【使用例】

データ・メモリの100H-1FFHの番地の領域のデータを200H番地から始まる領域のデータと交換する例です。

brk : 0 > RAM X 100, 1FF 200	①
------------------------------	---

- ① partition : 100H-1FFH番地
- addr : 200H番地
- 型 : 省略 (ニブル型)

(8) データ・メモリ内容の比較

形式5	RAM△V△partition△addr[$\left\{ \begin{array}{l} \Delta \$B \\ \Delta \$N \end{array} \right\}$]
基数	partition : H addr : H
brk : n >	○
emu : n >	×
trc : n >	×

【機能】

比較範囲として指定された範囲の内容と、比較先アドレスとして指定されたアドレス以降のメモリ内容を比較します。

- (a) 比較範囲内のアドレスと、比較先アドレスは重複してもかまいません。
- (b) 比較した結果、メモリ内容が一致しない場合は、一致しないアドレスとデータを表示します。
- (c) 比較は、比較範囲の終了まで続けます。

【指定方法】

オペランド

- partition : データ・メモリの比較範囲を指定します。
 比較範囲は、パーティション指定と、マスク指定ができます。
 パーティション指定の場合は、開始アドレスと、終了アドレスを“, ”で区切ります。
 アドレス指定範囲は対象デバイスにより異なります。
- addr : 比較先アドレスを1個指定します。
 パーティション指定と、マスク指定はできません。
 アドレス指定範囲は対象デバイスにより異なります。
- \$B : バイト型でデータ比較を行います。
 この場合、アドレスは偶数アドレスしか指定できません。
- \$N : ニブル型でデータ比較を行います (省略時)。

【使用例】

データ・メモリ100H-14FH番地を150H番地以降と比較する例です。

```
brk : 0>RAM V 100, 14F 150 ①  
  
010F 00 015F 01  
0115 36 0165 41  
0144 03 0194 2F  
  
brk : 0>■ ②
```

- ① partition : 100H-14FH番地
addr : 150H番地
型 : 省略 (ニブル型)

- ② 比較した結果, 一致しなかったアドレスとデータの表示

⑨ データ・メモリのロード

形式6 RAMΔL

brk : n >	○	emu : n >	×	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

カレント・ドライブの“IE75000.RAM”という名前のファイルを、データ・メモリにロードします。

- (a) ロードするロケーションは、“RAMΔS” コマンドで指定した領域にロードされます。
- (b) ファイル“IE75000.RAM”が、カレント・ドライブにない場合、またはそのファイル・フォーマットが異常な場合はエラーとなります。

【使用例】

正常にロードされた例です。

<pre>brk : 0 > <u>RAM L</u> ①</pre>
<pre>Object load complete ②</pre>
<pre>brk : 0 > ■</pre>

- ① データ・メモリのロード指定
- ② 終了状態の表示

(10) データ・メモリのセーブ

形式7	RAMΔS	[Δpartition]	[, partition]	[, partition]	[, partition]	[, partition]
基数	partition : H					
brk : n >	<input type="radio"/>	emu : n >	<input type="checkbox"/>	trc : n >	<input type="checkbox"/>	

【機能】

データ・メモリの最大5箇所のセーブ範囲を、カレント・ドライブに、“IE75000.RAM” というファイル名でセーブします。

- (a) アドレスが省略された場合は、全データ・メモリをセーブします。
- (b) すでに、カレント・ドライブに、“IE75000.RAM” というファイルがあった場合、次のメッセージを表示します。

File already exists. Delete? (Y or N) :

Y : ファイルを削除し、新しいファイルをオープンします。

N : ファイルはそのまま残します。

【指定方法】

オペランド

partition : データ・メモリのセーブしたいアドレス範囲を指定します。

アドレス範囲は、パーティション指定と、マスク指定ができます。

パーティション指定の場合は、開始アドレスと終了アドレスを“,”で区切ります。

アドレス指定範囲：対象デバイスにより異なります。

【使用例】

(a) データ・メモリの0H-1FFH番地をセーブする例

```
brk:0>RAM S 0, 1FF  ①
Object save complete ②
brk:0>■
```

- ① データ・メモリ・セーブの指定
- ② 終了状態の表示

(b) オペランドを省略して対象デバイスの全データ・メモリ領域をセーブする例

```
brk:0>RAM S  ①
File already exists. Delete? (Y or N) : Y  ②
Object save complete
brk:0>■ ③
```

- ① オペランドを省略した指定
- ② “Y ” を入力
すでにカレント・ドライブ中に、“IE75000.RAM” というファイルが存在する場合の処理です。
- ③ 終了状態の表示

8.4.28 汎用レジスタ操作コマンド (REG)

入力形式

形式1 REG Δ C [Δ register]形式2 REG [Δ D [{ Δ ALL
 Δ register }]]

【機能概要】

REGコマンドは、汎用レジスタ、および制御レジスタの変更、表示を行います。PSWに関しては、PSWフラグ名で変更、表示ができます。対象デバイスにより、変更、表示可能なレジスタが異なります。

(1) レジスタの変更

形式1 REGΔC[Δregister]

brk : n >	○	emu : n >	×	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

汎用レジスタ、および制御レジスタの変更を行います。

変更は、RBS（レジスタ・バンク・セレクト）が指すレジスタ・バンクの、指定レジスタ名に対して行われます。

(a) 汎用レジスタ、または制御レジスタが指定された場合は、指定したレジスタの変更ができません。

ただし、“PSW”を指定した場合は、フラグ単位で先頭から順番に変更でき、フラグ名を指定した場合には、指定したフラグのみが変更されます。

(b) レジスタ名を省略した場合は、汎用レジスタ、および制御レジスタを順番に変更できます。

【指定方法】

オペランド

register : 次のレジスタ、およびフラグの中から指定します。

制御レジスタ : PC, SP, PSW, RBS, MBS

汎用レジスタ : XA, HL, DE, BC, XA', HL', DE', BC', X, A, H, L, D, E,
B, C

PSWフラグ : CY, RBE, MBE, ISTO, IST1

省略時 : 汎用レジスタ、および制御レジスタを順番に指定できます。

【使用例】

(a) レジスタの変更

レジスタ名が省略された場合は、汎用レジスタおよび制御レジスタを順番に変更します。

```
brk : 0>REG C  ①

XA 00 = 11  ②
HL 11 = 22  ②
DE 22 = 33  ②
BC 33 =  ③
XA' 44 =  ③
HL' 55 =  ③
DE' 66 =  ③
BC' 77 =  ③
RBS 0 =  ③
MBS 0 =  ③
SP 000 =  注 ③
PC 0000 =  ③

brk : 0>■
```

- ① すべての汎用レジスタおよびPSWを除く制御レジスタの変更
- ② 変更値の設定
- ③ 変更なし

注 対象デバイスにより3/2バイト・スタックがあります。

(b) "PSW" を指定する例

制御レジスタのうち"PSW"を指定する場合は、PSWフラグ単位で変更できます。

```
brk : 0>REG C PSW  ①

RBE 0 = 1  ②
MBE 0 = 1  ②
CY 0 = 1  ②
IST1 0 =  ③
IST0 0 =  ③

brk : 0>■
```

- ① すべてのPSWフラグの変更
- ② 変更値の設定
- ③ 変更なし

(c) レジスタ名が指定された場合

レジスタ名に、汎用レジスタ、あるいは制御レジスタが指定された場合、指定したレジスタを変更できます。

```
brk:0>REG C DE
```

```
DE 00=11
```

```
brk:0>
```

①

① DEレジスタの変更

② レジスタの表示

形式2	$\text{REG} [\Delta D \left[\left\{ \begin{array}{l} \Delta \text{ALL} \\ \Delta \text{register} \end{array} \right\} \right]]$
-----	---

brk : n >	○	emu : n >	×	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

汎用レジスタ、および制御レジスタの表示を行います。

表示は、カレント・レジスタ・バンクの指定したレジスタに対して行われます。

- (a) オペランドに“ALL”を指定した場合には、対象レジスタの全レジスタ・バンクの汎用レジスタ、制御レジスタ、PSWフラグの表示を行います。
- (b) オペランドに“PSW”を指定した場合には、PSWフラグの表示を行います。
- (c) オペランドを省略した場合には、カレント・レジスタ・バンクの全汎用レジスタ制御レジスタ、PSWフラグの表示を行います。

【指定方法】

オペランド

- ALL : 対象レジスタの全レジスタ・バンクの汎用レジスタ、制御レジスタ、PSWフラグの表示を行います。
- register : 次のレジスタ、およびフラグの中から指定します。
 - 制御レジスタ : PC, SP, PSW, RBS, MBS
 - 汎用レジスタ : XA, HL, DE, BC, XA', HL', DE', BC', X, A, H, L, D, E, B, C
 - PSWフラグ : CY, RBE, MBE, IST0, IST1
- 省略時 : カレント・レジスタ・バンクの全汎用レジスタ、制御レジスタ、PSWフラグの表示を行います。

【使用例】

(a) オペランドを省略した場合

オペランドを省略した場合は、カレント・レジスタ・バンクの全汎用レジスタ、制御レジスタ、PSWフラグの表示を行います。

```
brk : 0>REG DⓂ

XA HL DE BC XA' HL' DE' BC' RBS MBS RBE MBE CY IST1 IST0 SP PC
00 00 00 00 00 00 00 00 0 0 0 0 0 0 0 000 0000

brk : 0>■
```

備考 対象デバイスにより存在しないレジスタは--で表示されます。

(b) レジスタ名に“PSW”を指定した場合

レジスタ名に“PSW”を指定した場合には、全PSWフラグの表示を行います。

```
brk : 0>REG D PSWⓂ

RBE MBE CY IST1 IST0
0 0 0 0 0

brk : 0>■
```

(c) オペランドにレジスタ名を指定した場合

```
brk : 0>REG D PCⓂ

PC 0000

brk : 0>■
```

(d) オペランドに“ALL”を指定した場合

オペランドに“ALL”を指定した場合には、対象デバイスの全レジスタ・バンクの汎用レジスタ、制御レジスタ、PSWフラグの表示を行います。

```
brk : 0>REG D ALL
```

RBS	MBS	RBE	MBE	CY	IST1	IST0	SP	PC
0	0	0	0	0	0	0	000	0000
	XA	HL	DE	BC	XA'	HL'	DE'	BC'
RB0	00	00	00	00	00	00	00	00
RB1	00	00	00	00	00	00	00	00
	.							
	.							
	.							
RB14	00	00	00	00	00	00	00	00
RB15	00	00	00	00	00	00	00	00

- ① すべてのレジスタ・バンクの内容表示を指定
- ② 表示 (対象デバイスにより異なる)

8.4.29 リセット・コマンド (RES)

入力形式
形式1 RES [Δ H]

brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="radio"/>
-----------	-----------------------	-----------	-----------------------	-----------	-----------------------

【機能】

RESコマンドは、エミュレーション・デバイスのリセット、IE-75001-Rのシステム・リセットを行います。

【指定方法】

オペランド

H : IE-75001-Rのシステム・リセットを行います。

省略時 : エミュレーション・デバイスのリセットを行います。

【使用例】

(a) オペランドに“H”を指定してIE-75001-Rのシステム・リセットを行う例

```
brk:0>RES H①
IE-75000/1-R Monitor V1.4 [1 Aug 92]
Copyright (C) 1989, 1992 by NEC Corporation

Self check ok

Target CPU      uPD75108/108A/108F/P108/P108B/112/112F/116/116F/P116②
Program Memory 0-FFFFH
Data Memory    00H-1FFH, F80H-FFFH
Memory Bank    0-1, 15
Register Bank   0-3
Power on target system (Y/N) Y③
Do you use high speed down load mode? (Y/N)=Y③
```

- ① IE-75001-Rのシステム・リセット
- ② 対象デバイスの表示
- ③ 高速ダウン・ロード・モードを使用

(b) エミュレーション・デバイスのリセット例

オペランド省略時は、エミュレーション・デバイスのリセットを行います。

```
brk : 0> RES
```

```
brk : 0>
```

①

① オペランドを省略指定

8.4.30 エミュレーション実行コマンド (RUN)

入力形式	
形式1	RUN△N [△addr]
形式2	RUN△B [△addr] .
形式3	RUN△T [△addr] $\left\{ \begin{array}{l} , \text{expression} \\ , \text{word} \end{array} \right\}$ [△TRD] [△REG]
基数	addr : H

【機能概要】

RUNコマンドは、エミュレーション・デバイスによるプログラムの実行を開始します。
サブコマンド1を指定することにより、3種類の実行機能を選択することができます。

(a) RUN△N：ノンブレイク・リアルタイム実行

指定されたアドレスからターゲット・プログラムの実行およびトレースを開始し、イベント検出によりトレーサのみ停止します。

エミュレーション・デバイスは停止しません。

(b) RUN△B：ブレイク条件付きリアルタイム実行

指定されたアドレスからユーザ・プログラムの実行およびトレースを開始し、イベント検出によりエミュレーション・デバイスとトレーサの双方を停止します。

(c) RUN△T：ステップ実行

指定されたアドレスからターゲット・プログラムのステップ実行を開始し、指定命令数の実行または、指定条件の一致によりエミュレーション・デバイスとトレーサの双方を停止します。

RUN△BおよびRUN△Tでは、コマンド実行後、イベント検出や指定命令数の実行、または指定条件の一致によりエミュレーション・デバイスとトレーサの双方が停止すると1ステップ実行モード注となります (図8-3参照)。

注 1ステップ実行モード

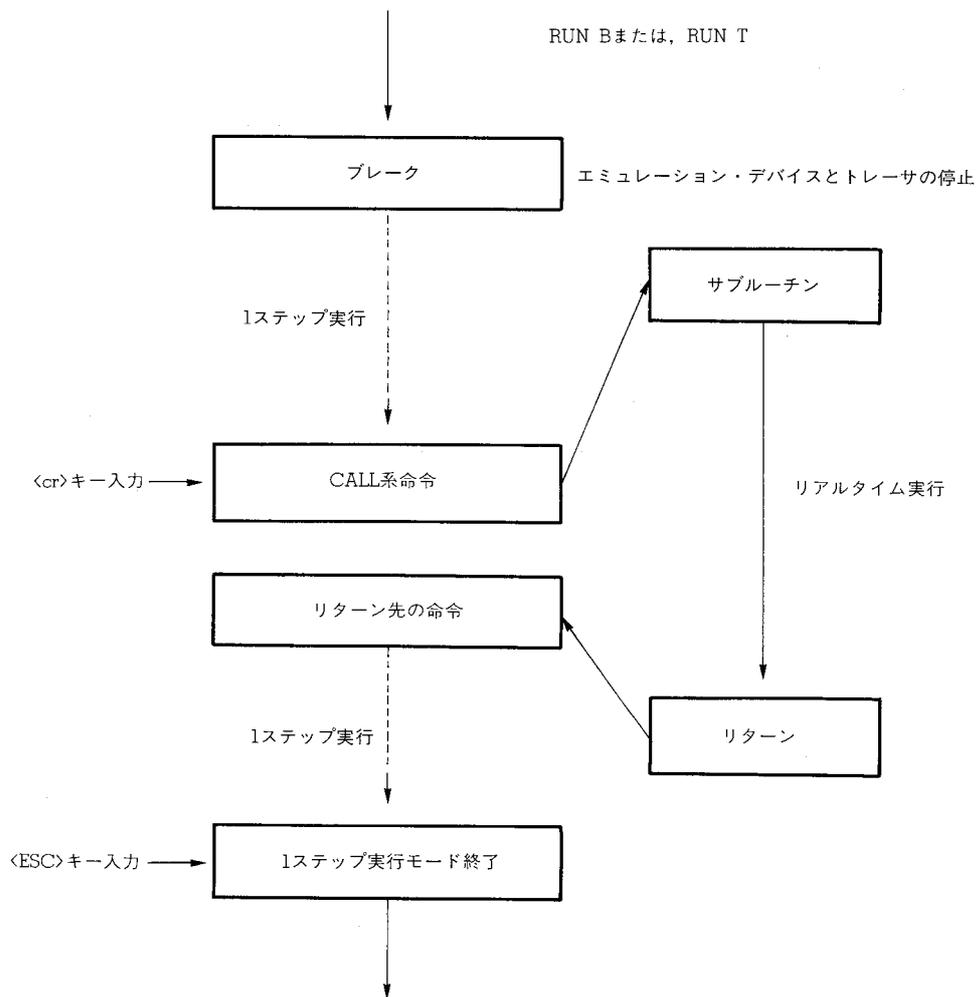
1ステップ実行モードでは、〈cr〉（リターン）キーの入力によりターゲット・プログラムを1ステップずつ実行します。1ステップ実行するごとに実行した命令の逆アセンブル表示とカレント・バンクのレジスタ値の表示を行います。また、CALL, CALLA, CALLF, CALLT命令の実行後に / 〈cr〉（スラッシュ・リターン）を入力すると、CALL系命令の分岐先のサブルーチンをリアルタイム実行し、サブルーチンからリターンした次の命令を実行して、1ステップ実行モードになります。リアルタイム実行中は何も表示されません。

〈ESC〉キーの入力で1ステップで実行モードを終了し、コマンド入力待ちになります。

【使用規約】

1ステップ実行時には、割り込み要求があった場合、要求フラグ (IRQx) はセットされますが、割り込みは実行されません。1ステップ実行を行うと、それまでのトレース・データは初期化されます。1ステップ実行では、STOP, HALT状態になりません。

図 8-3 1ステップ実行モード



1ステップ実行モードを終了するには〈ESC〉キーを入力します。

(I) ノンブレイク・リアルタイム実行

形式1	RUN△N [△addr]				
基数	addr : H				
brk : n >	○	emu : n >	×	trc : n >	×

【機能】

オペランドで指定されたアドレスからターゲット・プログラムの実行およびトレースを開始し、イベント検出またはSTP Tコマンドによりトレーサのみを停止します。

エミュレーション・デバイスは、フェール・セーフ・ブレイクの発生または、STPコマンドによる強制終了指定以外では停止しません。

- (a) オペランドが省略された場合、現在のPCの値が実行開始アドレスになります。
- (b) トレーサおよびエミュレーション・デバイスを強制的に停止させるには、次に示す方法があります。

	現在のモード	
	trc : n >	emu : n >
STP コマンド	<ul style="list-style-type: none"> ・トレーサとエミュレーション・デバイスの双方が停止する。 ・brk : n >モードになる。 	<ul style="list-style-type: none"> ・エミュレーション・デバイスが停止する。 ・brk : n >モードになる。
STP T コマンド	<ul style="list-style-type: none"> ・トレーサが停止する。 ・エミュレーション・デバイスは停止しない。 ・emu : n >モードになる。 	<ul style="list-style-type: none"> ・使用できない。

- (c) イベント検出または、STP Tコマンドでトレーサが停止した場合、トレーサの停止要因を表示します。
 - (i) イベント検出
BRA1, BRA2, BRA3, BRA4, BRS1, BRS2
 - (ii) トレーサ停止指定
STP T
- (d) フェール・セーフ・ブレイクまたは、STPコマンドによる強制終了指定でエミュレーション・デバイスが停止した場合、エミュレーション・デバイスの停止要因とカレント・バンクのレジスタ値を表示します。
 - (i) フェール・セーフ・ブレイク
GDM, GDIO, GDR, GDSP

(ii) 強制終了指定

STP

【指定方法】

オペランド

ターゲット・プログラム実行の開始アドレスを指定します。

addr : ターゲット・プログラム実行開始アドレス

アドレス指定範囲：0-0FFFFH

【使用例】

(a) ノンブレイク・リアルタイム実行

```

brk : 0>RUN N 100①
      Emulation Start at 0100
trc : 0>■②
       tracer stop
emu : 0>■③
emu : 0>STP④
      STP break terminated
      XA HL DE BC XA' HL' BC' RBS MBS RBS MBE CY IST1 IST0 SP PC
      00 00 00 00 00 00 00 0 0 0 0 0 0 0 0 000 0000
brk : 0>■
brk : 0>RUN N 100⑤
      Emulation Start at 0100
trc : 0>■⑥
       tracer stop
emu : 0>■⑦
emu : 0>TRG⑧
      Tracer Start
trc : 0>■
  
```

- ① トレース・モード・コマンド待ち
- ② トレーサ停止要因の表示 "BRA1-BRA4", "BRS1", "BRS1", "STPT"
- ③ エミュレーション・モード・コマンド待ち
- ④ "STP" コマンドによるブレイク
- ⑤ トレース・モード・コマンド待ち
- ⑥ トレーサ停止要因の表示 "BRA1-BRA4", "BRS1", "BRS1", "STPT"
- ⑦ "TRG" コマンドによるトレーサの再スタート
- ⑧ トレース・モード・コマンド待ち

(b) "RUN N" の強制ブレークの表示例

```

trc : 0>■

```

①

```

    break terminated

```

②

XA	HL	DE	BC	XA'	HL'	BC'	RBS	MBS	RBS	MBE	CY	IST1	IST0	SP	PC
00	00	00	00	00	00	00	0	0	0	0	0	0	0	000	0000

```

brk : 0>■

```

- ① トレース・モード・コマンド待ち
- ② 強制ブレーク要因の表示
"GDM", "GDIO", "GDR", "GDSP"

② ブレーク条件付きリアルタイム実行

形式2 RUN△B [△addr]					
基数	addr : H				
brk : n >	<input type="radio"/>	emu : n >	<input type="checkbox"/>	trc : n >	<input type="checkbox"/>

【機能】

オペランドで指定されたアドレスからターゲット・プログラムの実行を開始し、イベント検出によりエミュレーション・デバイスとトレーサの双方を停止します。フェール・セーフ・ブレークの発生、またはSTPコマンドによる強制終了指定でもエミュレーション・デバイスとトレーサの双方を停止します。

- (a) オペランドが省略された場合、現在のPCの値が実行開始アドレスになります。
- (b) エミュレーション・デバイスの実行を強制的に停止させるには、STPコマンドを入力します。
STPコマンドにより、トレーサとエミュレーション・デバイスの双方が停止し、“trc : n >”モードから“brk : n >”モードになります。
- (c) エミュレーション・デバイスが停止した場合、停止要因とカレント・バンクのレジスタ値を表示します。
 - (i) イベント検出
BRA1, BRA2, BRA3, BRA4, BRS1, BRS2
 - (ii) 強制終了指定
STP T
 - (iii) フェール・セーフ・ブレーク
GDM, GDIO, GDR, GDSP
- (d) イベントの検出でエミュレーション・デバイスが停止すると、停止要因とカレント・バンクのレジスタ値を表示し、1ステップ実行モードとなります。

【指定方法】

オペランド

ターゲット・プログラム実行の開始アドレスを指定します。

addr : ターゲット・プログラム実行開始アドレス
アドレス指定範囲 : 0-0FFFFH

【使用例】

(a) "RUN B" におけるブレークと1ステップ実行の表示例

```

brk : 0> RUN B 100  ①
Emulation Start at 0100

trc : 0>  ②

 break terminated ③

XA HL DE BC XA' HL' BC' RBS MBS RBS MBE CY IST1 IST0 SP PC
00 00 00 00 00 00 00 0 0 0 0 0 0 0 0 000 0000

One Step emulation standby  ④

 ⑤

XA HL DE BC XA' HL' BC' RBS MBS RBS MBE CY IST1 IST0 SP PC
00 00 00 00 00 00 00 0 0 0 0 0 0 0 0 000 0000

One Step emulation standby _____ ⑥

brk : 0> 
    
```

- ① 100番地から実行
- ② トレース・モード・コマンド待ち
- ③ ブレーク要因の表示
"BRA1-BRA4", "BRS1-BRS2"
- ④ <cr> キーを入力
- ⑤ "TRD" コマンドと同様なトレース表示
- ⑥ <ESC> キーを入力

(b) "RUN B" における強制ブレークの表示例

```

trc : 0>■

```

①

```

    break terminated

```

②

XA	HL	DE	BC	XA'	HL'	BC'	RBS	MBS	RBS	MBE	CY	IST1	IST0	SP	PC
00	00	00	00	00	00	00	0	0	0	0	0	0	0	000	0000

```

brk : 0>■

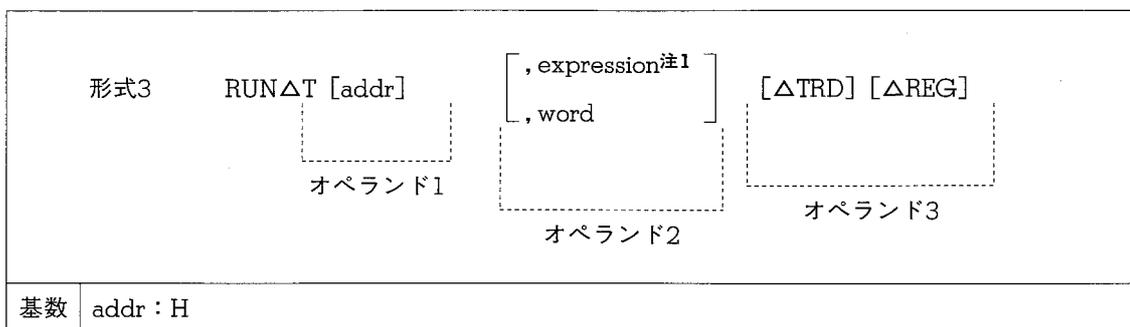
```

① トレース・モード・コマンド待ち

② 強制ブレーク要因の表示

"GDM", "GDIO", "GDR", "GDSP"

③ ステップ実行



brk : n>	○	emu : n>	×	trc : n>	×
----------	---	----------	---	----------	---

注1. expression : register^{注2}

=	(マスク・データ ^{注3}
>	ビット・データ
<	4ビット・データ
=>	8ビット・データ
>=	16ビット・データ
=<	
<=	
><	
<>	

2. register : X, A, H, L, D, E, B, C, XA, HL, DE, BC, XA', HL', DE', BC',
PC, SP, RBS, MBS, CY, RBE, MBE, IST1, IST0

3. マスク・データ : マスク・データの場合 “=”, “><”, “<>” 以外の条件は使用できません。

【機能】

オペランド1で指定されたアドレスから、オペランド2で指定された条件式が成立するまで、または指定ステップ数と同数だけターゲット・プログラムをステップ実行します。

ステップ実行中はオペランド3の指定により実行した命令の逆アセンブル表示、およびカレント・バンクのレジスタ値の表示を行います。

使用上の注意点

- オペランド1が省略された場合、現在のPCの値が実行開始アドレスになります。
- オペランド2が省略された場合、1ステップだけ実行します。
- オペランド3が省略された場合、ステップ実行中の実行した命令の逆アセンブル表示、およびカレント・バンクのレジスタ値の表示は行いません。
- レジスタ条件の一致、または指定ステップ数の実行によりエミュレーション・デバイスの実行が停止すると、カレント・バンクのレジスタ値を表示し、1ステップ実行モードとなります。
- ステップ実行はリアルタイム実行ではありません。したがって、BRM, TRMの指定はステップ実行中は無効となります。また、ステップ実行ではトレーサがクリアされますので、ステップ実行後、TRDコマンドによりトレース・データを実行することはできません。
- ステップ実行中にフェール・セーフ・ブ레이크が発生した場合、または<ESC>キーの入力により強制終了指定を行った場合、停止要因とカレント・バンクのレジスタ値を表示してbrk:n>モードになります。
 - (i) フェール・セーフ・ブ레이크
GDM, GDIO, GDR, GDSP
 - (ii) 強制終了指定
ESC

【指定方法】

(a) オペランド1

ステップ実行の開始アドレスを指定します。

addr : ステップ実行開始アドレス

アドレス指定範囲: 0-0FFFFH

省略時: 現在のPCの値がステップ実行開始アドレスとなります。

(b) オペランド2

ステップ実行の終了条件を指定します。

expression: レジスタの値によってステップ実行を終了させたい場合に、レジスタ名と値の条件を指定します。

register : X, A, H, L, D, E, B, C, XA, HL, DE, BC, XA', HL',
DE', BC', PC, SP, RBS, MBS, CY, RBE, MBE, IST1, IST0

備考 指定できるレジスタは対象デバイスにより異なります。

比較演算子: =, >, <, =>, >=, =<, ><, <>

比較値 : マスク・データ
ビット・データ
4ビット・データ
8ビット・データ
16ビット・データ

備考 マスク・データの場合 "=", "><", "<>" 以外の比較演算子は使用できません。

word : 決められたステップ数の実行によりステップ実行を終了させたい場合に、ステップ数を指定します。

ステップ数指定範囲: 1-FFFFH

省略時 : ステップ数に1が指定されたとみなします。

(c) オペランド3

ステップ実行中に表示する情報を指定します。情報は1ステップごとに表示されます。

TRD : 実行した命令の逆アセンブル表示を行います。

REG : カレント・バンクのレジスタ値を表示します。

省略時 : 何も表示しません。

【使用例】

トレース実行の表示の例です。

```

brk : 0>RUN T 100, A=1 TRD REG①
Emulation Start at 0100

②
XA HL DE BC XA' HL' BC' RBS MBS RBS MBE CY IST1 ISTO SP PC
00 00 00 00 00 00 00 0 0 0 0 0 0 0 000 0000
.
.
.
.
Terminated
One Step emulation standby③

④
.
.
強制ブレーク時

⑤
break terminated

XA HL DE BC XA' HL' BC' RBS MBS RBS MBE CY IST1 ISTO SP PC
00 00 00 00 00 00 00 0 0 0 0 0 0 0 000 0000

brk : 0>■
    
```

- ① Aレジスタ=1までトレース実行
- ② 1命令分のトレースと逆アセンブル
 “TRD” コマンドと同様なトレースを表示します。
 Aレジスタ=1または強制ブレーク条件が成立するまで繰り返します。
- ③ 条件成立時 (Aレジスタ=1) , <cr> キーを入力
- ④ “RUN B” と同様な1ステップ実行モード
- ⑤ 強制ブレーク要因の表示
 “ESC”, “GDM”, “GDIO”, “GDR”, “GDSP”

8.4.31 セーブ・コマンド (SAV)

入力形式	形式1	SAV Δ	[d:] file	[Δ partition]-----[Δ partition]	[{ Δ C } { Δ D }]
			オペランド1	オペランド2 (最大5つ)	オペランド3
基数	partition : H				

brk : n >	<input type="radio"/>	emu : n >	<input type="checkbox"/>	trc : n >	<input type="checkbox"/>
-----------	-----------------------	-----------	--------------------------	-----------	--------------------------

【機能】

SAVコマンドは、IE-75001-RのCH1と接続したホスト・マシンに、オペランド1で指定したファイル名でオブジェクト・コードおよびデバッグ環境のセーブを行います。

- (a) オブジェクト・コードは、“filename.HEX”というファイル名で、オペランド2で指定されたプログラム・メモリの領域を、インテル標準HEX形式でホスト・マシンにセーブします
- (b) デバッグ環境は“filename.DBG”というファイル名で、コマンドの設定値をホスト・マシンにセーブします。デバッグ環境として設定値をセーブするコマンドを次に示します。

BRA1, BRA2, BRA3, BRA4, BRK, BRM, BRS1, BRS2, CHK, CLK, DLY, MOD, OUT, PAS, PGM, REG, STS, TRF, TRM, TRX, TRY

- (c) オペランド2が省略された場合、0-OFFFFHのプログラム・メモリをセーブします。
- (d) オペランド3を指定することにより、オブジェクト・コードだけ、あるいはデバッグ環境だけをセーブすることができます。オペランド3を省略すると、オブジェクト・コードとデバッグ環境の両方をセーブします。
- (e) ファイル名が登録済みの場合には、ファイルの属性指定によって次の処理を行います。

- (i) DIR属性かつR/W属性のとき

次のメッセージを表示し、登録済みのファイルを消去して新たにファイルを作成するか、または登録済みのファイルをそのまま残すかをユーザに選択させます。

File already exists. Delete? (Y or N) : _

Y : 登録済みのファイルを消去して、新たにファイルを作成する。

N : 登録済みのファイルをそのまま残す。

- (ii) SYS属性かつR/O属性のとき

次のメッセージを表示して、コマンドを無視します。

File already exists.

【指定方法】

(a) オペランド

オブジェクト・コードまたは、デバッグ環境をセーブするファイルの名前を指定します。

file : 拡張子を除いたファイル名

(b) オペランド2

partition : プログラム・メモリのセーブしたい領域を指定します。

アドレス指定形式 : “開始アドレス, 終了アドレス”

アドレス指定範囲 : 0-0FFFFH

指定回数 : 最大5個

省略時 : “0-0FFFFH” が指定されたとみなし64 Kバイトのプログラム・メモリがセーブされます。

(c) オペランド3

セーブするデータの種別を指定します。

C : オブジェクト・コードだけをセーブします。

D : デバッグ環境だけをセーブします。

省略時 : オブジェクト・コード, デバッグ環境の両方をセーブします。

【使用例】

(a) アドレスを指定しないでセーブする場合

プログラム・メモリのカバレッジ測定範囲（ロードしたオブジェクト・ファイルと同じ範囲）およびデバッグ環境を、ホストのカレント・ドライブに“SAMPLE.HEX”，“SAMPLE.DBG”というファイル名でそれぞれセーブする例です。

```
brk : 0>SAV SAMPLE C
Object save complete
debug date save complete
brk : 0>■
```

(b) アドレスを指定してセーブする場合

プログラム・メモリの0-1FFFHと3000H-30FFHの領域をホストのカレント・ドライブに“SAMPLE.HEX”というファイル名でセーブする例です。

```
brk : 0>SAV SAMPLE 0, 1FFF 3000, 30FF C
File already exists. Delete? (Y or N) : Y
Object save complete
brk : 0>■
```

①

① “SAMPLE.HEX” がすでに存在する場合

(c) “SAMPLE.HEX” がすでに存在する場合

“SAMPLE.HEX” がSYS属性もしくは、R/O属性ですすでに存在する場合の例です。

```
brk : 0>SAV SAMPLE C
File already exists
```

8.4.32 エミュレーション・デバイス・モード切り替えコマンド (SET)

入力形式

$$\text{形式1} \quad \text{SET} \left[\left\{ \begin{array}{l} \Delta\text{STACK} \\ \Delta\text{SLWAIT} \end{array} \right\} \left\{ \begin{array}{l} \Delta\text{OFF} \\ \Delta\text{ON} \end{array} \right\} \right]$$

brk : n >	<input type="radio"/>	emu : n >	<input type="checkbox"/>	trc : n >	<input type="checkbox"/>
-----------	-----------------------	-----------	--------------------------	-----------	--------------------------

【機能】

SETコマンドは、エミュレーション・デバイスの割り込み時のスタック・バイト数の切り替え、およびエミュレーション・デバイスのRESET後のウェイト時間の切り替えを行います。オペランドが省略された場合は、STACKおよびSLWAITの現在の設定されている値が、OFFまたはONで表示されます。

【指定方法】

(a) オペランド1

スタック・バイト数のセットか、ウェイト時間のセットかを指定します。

STACK : エミュレーション・デバイスの割り込み時のスタック・バイト数の指定

SLWAIT : エミュレーション・デバイスのRESET後のウェイト時間の指定

(b) オペランド2

STACKまたはSLWAITの値を設定します。

STACKの場合

OFF : 2バイト・スタックに設定します。

ON : 3バイト・スタックに設定します。

STSコマンドまたはIE-75000-R-EM(エミュレーション・ボード)上のDIPスイッチ(SW1)により、対象デバイスを μ PD75217, 218, P218, 236, 237, 238, P238, P336, 517, 518, P518 (内部ROM24/32バイトのデバイス)とした場合、3バイト・スタックが自動的に選択されます。上記以外は2バイト・スタックです。

SLWAITの場合

OFF : ウェイト時間を31.25 msに設定します (デフォルト)。

ON : ウェイト時間を7.81 msに設定します。

【使用規約】

本コマンドで設定された値は、リセット・コマンドの実行後、有効となります。

【使用例】

(a) オペランドを省略したとき

オペランドを省略すると現在の設定内容が表示されます。

```
brk : 0>SET  
  
STACK : OFF  
SLWAIT : OFF  
  
brk : 0>■
```

(b) エミュレーション・デバイス・モード切り替え

STACK, SLWAITの設定を行い、エミュレーション・デバイスのモードを切り替える例です。

```
brk : 0>SET STACK ON  
  
brk : 0>SET SLWAIT OFF  
  
brk : 0>RES
```

①
②
③

- ① スタックを3バイトに設定
- ② ウェイト時間を31.25 msに設定
- ③ RESコマンドの実行（実行後、モードが切り替わる）

8.4.33 特殊レジスタ操作コマンド (SPR)

入力形式	
形式1	$\text{SPR}\Delta\text{C} \left[\left\{ \begin{array}{l} \Delta\text{group} \\ \Delta\text{register} \end{array} \right\} \right]$
形式2	$\text{SPR} [\Delta\text{D} \left[\left\{ \begin{array}{l} \Delta\text{group} \\ \Delta\text{register} \end{array} \right\} \right]]$

【機能概要】

SPRコマンドは、対象デバイスのI/O領域（以降、特殊レジスタと呼びます）の変更、表示を行います。

変更、表示を行える特殊レジスタは、対象デバイスにより異なります。実際には、STSコマンドで設定した対象デバイスが持つ特殊レジスタとなります。

(a) グループ名

75Xシリーズでは、特殊レジスタは、データ・メモリの0F80H-0FFFH番地にマッピングされます。“SPR”コマンドでは、このマッピング・アドレスの上位8ビットを特殊レジスタのグループ名として指定することができます。グループ名で特殊レジスタを指定した場合、グループに属す特殊レジスタを一括して変更、表示を行うことができます。

“F8X”：データ・メモリの0F80H-0F8FHにマッピングされている特殊レジスタの変更、表示を行う。

使用できるグループ名は対象デバイスにより異なります。

(b) レジスタ名

特殊レジスタ名を指定して変更、表示を行うことができます。特殊レジスタ名は、対象デバイスにより異なります。

(1) 特殊レジスタの変更

形式1 $\text{SPR}\Delta\text{C} \left[\begin{array}{l} \Delta\text{group} \\ \Delta\text{register} \end{array} \right]$
--

brk : n >	○	emu : n >	×	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

レジスタ名、またはグループ名で指定した特殊レジスタの変更を行います。

- (a) レジスタ名を指定した場合、指定した特殊レジスタだけを変更します。
- (b) グループ名で指定した場合、グループに属すすべての特殊レジスタを変更します。
- (c) グループ名およびレジスタ名を省略した場合、全特殊レジスタをグループ単位で変更します。
- (d) グループ単位に変更を行う場合、途中で変更を終了するには<ESC>キーを入力します。また、途中で変更する必要がない場合、次のレジスタへ進むには<cr>キーを入力します。
- (e) 変更できる特殊レジスタは、Write onlyおよび、Read/Write特殊レジスタです。
- (f) Write only, Read only特殊レジスタの識別
 - Write onlyの特殊レジスタは“Wo”と表示されます。
 - Read onlyの特殊レジスタは“Ro”と表示されます。

【指定方法】

(a) サブコマンド

“C”と指定します。

(b) オペランド

group : 特殊レジスタがマッピングされているデータ・メモリ・アドレス (0F80HからOFFFH) の上位8ビットをグループ名として指定します。

備考 指定できるグループ名は対象デバイスにより異なります。

register : 特殊レジスタ名を指定します。

省略時 : オペランドを省略すると、全特殊レジスタをグループ単位で変更する指定となります。

【使用例】

(a) グループ名指定

グループ名を指定して特殊レジスタを変更する例です。

```
brk:0>SPR C F8X①
(F8X)
SBS      2=1②
BTM.3    .0=1②
BTM      Wo=③
BT       00=Ro④
DSPM     Wo=②
DIMS     Wo=②
DIGS     Wo=②
KSF.3    .0=Ro②
KSF      2=Ro②
brk:0>■
```

- ① グループ名指定による特殊レジスタの変更OF80H-OF8FH番地の特殊レジスタ
- ② Read/Write特殊レジスタ値の変更
- ③ Write onlyの特殊レジスタは“Wo”と表示
- ④ Read onlyの特殊レジスタは“Ro”と表示

(b) レジスタ名指定

レジスタ名を指定して特殊レジスタを変更する例です。

```
brk:0>SPR C BTM①
BTM      Wo=1②
brk:0>■
brk:0>SPR C BT②
BT       00=Ro②
brk:0>■
```

- ① レジスタ名指定による特殊レジスタの変更
- ② Read onlyの特殊レジスタを変更しようとした場合“Ro”と表示され、変更不可

(c) オペランドを省略するとき

オペランドを省略すると全特殊レジスタの変更指定となります。

```
brk:0>SPR C
(F8X)
SBS          1=0
BTM.3       .1=0
.
.
.

(F9X)
TPGM.3      .0=
TPGM        Wo=0
.
.
.
```

①

②

① グループ名の表示

② グループ名の表示

【使用例】

(a) グループ名指定

グループ名を指定して特殊レジスタを表示する例です。

```
brk : 0>SPR D F8X①
(F8X)
SBS=1    BTM.3=.1    BTM=Wo    BT=00    DSPM=Wo    DIMS=Wo
DIGS=Wo  KSF.3=.0    KSF=2
brk : 0>■
```

- ① レジスタ名指定による特殊レジスタの表示
- ② レジスタ名と値を表示

(b) レジスタ名指定

レジスタ名を指定し特殊レジスタを表示する例です。

```
brk : 0>SPR D BT①
BT      00
brk : 0>■
brk : 0>SPR D BTM②
BTM     Wo
brk : 0>■
```

- ① レジスタ名指定による特殊レジスタの表示
- ② Write onlyの特殊レジスタの表示

(c) オペランドを省略するとき

オペランドを省略すると全特殊レジスタの表示指定となります。

```

brk:0>SPR D①
[F8X]
      SBS=1      BTM.3=.1      BTM=Wo      BT=00      DSPM=Wo      DIMS=Wo②
      DIGS=Wo     KSF.3=.0      KSF=2

[F9X]
      TPGM.3=.0   TPGM=Wo     MODL=00     MODH=00     WM=Wo

[FAX]
      .          .....

[FBX1]
      .          .....

[FBX2]
      .
      .

[FEX]
      .
      .

[FFX]
      .
      .          .....

brk:0>■
    
```

- ① 全特殊レジスタの表示指定
- ② 特殊レジスタ名=値

8.4.34 入力デバイス・リダイレクト・コマンド (STR)

入力形式

形式1 STR△ [d:] file [△parameter list]

brk : n >	○	emu : n >	×	trc : n >	○
-----------	---	-----------	---	-----------	---

【機能】

STRコマンドは、このコマンド以降のコマンドおよびデータをホスト・マシンの指定ファイルから入力します。この際、本コマンドで実パラメータを指定することにより、ファイル中の仮パラメータに置き換えることができます。

- (a) ファイルからのコマンドおよびデータの入力を一時中断する場合は、 \hat{L} キーを入力します。
- (b) 再会する場合は、コマンド待ち状態で \hat{L} キーを入力します。
- (c) 終了する場合は、 \hat{K} キーを入力します。

【指定方法】

オペランド

[d:]file : コマンド、およびデータが格納されているファイルの名前を指定します。ドライブ番号は省略することができます。

parameter list : 実パラメータを指定します。

実パラメータは、最大4個まで指定することができます。

【使用規約】

入力ファイルの作成

本コマンドで使用できるファイルの形式は、COMコマンドで作成されたファイル、あるいはエディタ等によってコマンド、およびデータの入力形式で作成されたものに限りません。また仮パラメータを使用する場合のファイルは、エディタ等で作成してください。

仮パラメータは、\$0, \$1, \$2, \$3と記述します。

仮パラメータの\$とアセンブラを指定する\$を区別するため、相対アドレスを指定する場合は、\$\$と記述してください。

【使用例】

- (a) "SAMPLE.STR" というファイルから、コマンドを入力する例

```

brk : 0>STR b : SAMPLE.STR①
brk : 0>LOD SAMPLE
object load complete
symbol table loading
PUBLIC      load complete
MOD00      load complete
MOD01      load complete
MOD02      load complete
MOD03      load complete
MOD04      load complete
MOD05      load complete
brk : 0>MEM D 0X
0000  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F .....
brk : 0>MEM F OXX 00
brk : 0>MEM D OXX
0000  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0010  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
      :
00F0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
brk : 0>■

```

- ① ドライブBの "SAMPLE.STR" を入力ファイルに指定

ファイルからのコマンドおよびデータの入力は、ファイルの終了を検出した場合に終了します。ファイルが終了した場合、その後の入力はキーボードからの入力となります。上記のファイルの内容を次に示します。

```

A>TYPE b : SAMPLE.STR①
LOD SAMPLE
MEM D 0X
MEM F OXX 00
MEM D OXX
A>■

```

(b) 仮パラメータを記述したファイルから、コマンドを入力する例

```

brk : 0>STR SAMPLE.STR OXX OXX①
brk : 0>LOD SAMPLE
object load complete
symbol table loading
PUBLIC      load complete
MOD00      load complete
MOD01      load complete
MOD02      load complete
MOD03      load complete
MOD04      load complete
MOD05      load complete
brk : 0>MEM D OX②
0000  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F .....
brk : 0>MEM F OXX 00②
brk : 0>MEM D OXX②
0000  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0010  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
      :
00F0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....

brk : 0>■
    
```

- ① パラメータを指定
- ② 仮パラメータを展開

パラメータが指定された場合は、ファイル中に仮パラメータが指定されていなければなりません。仮パラメータが指定されていない場合は、実パラメータは無視されます。

上記例のファイルの内容を次に示します。

```

A>TYPE b : SAMPLE.STR①
LOD SAMPLE
MEM D OX
MEM F $0 00
MEM D $1
A>■②
    
```

- ① 仮パラメータ1
- ② 仮パラメータ2

(c) 指定したファイルが存在しない場合

```
brk : 0>STR b : SAMPLE. STR
```

```
B : SAMPLE. STR file not found
```

```
brk : 0>■
```

①

① ファイルが見つからない場合のメッセージ

8.4.35 リアルタイム・エミュレーション中止コマンド (STP)

入力形式

形式1 STP

形式2 STP△T

【機能概要】

STPコマンドは、エミュレータ中のエミュレーション・デバイスの実行またはトレーサの動作を中止します。

【使用規約】

RUN△Bの場合は、STP△Tを使用することはできません。

(1) エミュレーション・デバイスとトレーサの停止

形式1	STP
-----	-----

brk : n >	×	emu : n >	○	trc : n >	○
-----------	---	-----------	---	-----------	---

【機能】

エミュレーション・デバイスとトレーサの双方を停止します。

本コマンド実行後は、“brk : n >” モードになります。

【指定方法】

コマンド本体のみ入力します。

【使用例】

STPコマンドにより、リアルタイム・エミュレーションを中止する場合の例です。

<pre>brk : 0 > RUN N 100</pre> <p>Emulation start at 100</p> <pre>trc : 0 > STP</pre> <p>STP break terminated</p> <pre>XA HL DE BC XA' HL' BC' RBS MBS RBE MBE CY IST1 IST0 SP PC 00 00 00 00 00 00 00 0 0 0 0 0 0 0 0 000 0000</pre> <pre>brk : 0 > ■</pre>	<p>①</p> <p>②</p> <p>③</p> <p>④</p>
---	-------------------------------------

- ① エミュレーション開始指定
- ② エミュレーション・モードでのSTPコマンドによるブレイク指定
- ③ STPコマンドにより停止したことを表示
- ④ レジスタ表示

② トレーサの停止

形式2	STP△T
-----	-------

brk : n >	×	emu : n >	×	trc : n >	○
-----------	---	-----------	---	-----------	---

【機能】

トレーサのみを停止します。

本コマンド実行後は、“emu : n >” モードになります。

【指定方法】

メイン・コマンドおよびオペランドの入力のみです。

【使用例】

STP△Tコマンドにより、トレーサを停止した場合の例です。

```
brk : 0 > RUN N 200
```

```
Emulation start at 0200
```

```
trc : 0 > STP T
```

```
STP T tracer stop
```

```
emu : 0 > ■
```

①

②

① エミュレーション開始指定

② トレーサ停止指定

8.4.36 対象デバイス選択コマンド (STS)

入力形式	
形式1	STS△C
形式2	STS [△D]

【機能概要】

STSコマンドは、IE-75001-Rでディバグする対象デバイスを選択、あるいは現在選択している対象デバイスの表示を示します。

STS△C：対象デバイスの選択

STS△D：対象デバイスの表示

(a) 本コマンドで対象デバイスを選択することに伴い、次に示す項目が変更されます。

- ① ガード・アドレス（データ・メモリ、SFR、レジスタ、スタック領域）
- ② REGコマンドで操作可能なレジスタ名
- ③ TRDコマンドで参照可能なI/Oポート名
- ④ SPRコマンドで操作可能な特殊レジスタ名
- ⑤ ASMおよびDASコマンドで使用可能な命令セット
- ⑥ RAMコマンドで操作可能なデータ・メモリ領域

(b) パワーオン時には、 μ PD75108, 108A, 108F, P108, P108B, 112, 112F, 116, 116F, P116が設定された状態になっています（IE-75000-R-EMを接続した場合）。

(1) 対象デバイスの選択

形式1	STSΔC
-----	-------

brk : n >	<input type="radio"/>	emu : n >	×	trc : n >	×
-----------	-----------------------	-----------	---	-----------	---

【機能】

対象デバイスの選択を行います。

本コマンドを入力すると、デバイス一覧が表示されます。このデバイス一覧の中より、対象デバイスの番号を選択することができます。

なお、対象デバイスの表示は2画面に分かれています。STSΔC実行後、<cr>で画面が切り替わります。

【指定方法】

コマンド本体およびサブコマンドのみ入力します。

注意 本コマンドで対象デバイスの番号を入力せずに '.'のみを入力した場合は、設定変更は行われず現状の対象デバイスのままとまります。

【使用例】

デバッグ対象デバイスとして μ PD75512, 516, P516を選択する例です。

```

brk : 0>STS C
(1) Target CPU uPD75004/006/008/P008
(2) Target CPU uPD75028
(3) Target CPU uPD75036/P036
(4) Target CPU uPD75048/P048
(5) Target CPU uPD75064/066/068/P068
(6) Target CPU uPD75104/104A/106
(7) Target CPU uPD75108/108A/108F/P108/P108B/112/112F/116/116F/P116
(8) Target CPU uPD75116H
(9) Target CPU uPD75117H/P117H
(10) Target CPU uPD75206
(11) Target CPU uPD75208/CG208
(12) Target CPU uPD75212A/216A/CG216A/P216A
(13) Target CPU uPD75217
(14) Target CPU uPD75218/P218
(15) Target CPU uPD75236
(16) Target CPU uPD75237/238/P238
(17) Target CPU uPD75268
(18) Target CPU uPD75304/304B/306/306B/308/308B/P308/312/316/P316
(19) Target CPU uPD75312B/316B/P316A/P316B
(20) Target CPU uPD75328/P328
(21) Target CPU uPD75336/P336
(22) Target CPU uPD75402/402A/P402

Target CPU No.7 (cr : next page/. : end) =

```

- ① コマンド入力
- ② 対象デバイスの一覧表示
- ③ 画面を次ページに切り替え

```

(23) Target CPU uPD75512/516/P516
(24) Target CPU uPD75517/518/P518

Target CPU No.7 (cr : next page/. : end) = 23

```

- ① 対象デバイスを(23) μ PD75512, 516, P516に設定

備考1. μ PD75P048, 064, 066, 068, P068, P117H, 218, P316Bは開発中です。★

2. μ PD75402は保守品です（新規のご購入はできません）。

② デバッグ対象デバイスの表示

形式1 STS [Δ D]

brk : n> ○ emu : n> × trc : n> ×

【機能】

デバッグ対象デバイスの表示を行います。

表示される項目は、以下のとおりです。

- ① デバッグ対象デバイス名
- ② プログラム・メモリ領域アドレス
- ③ データ・メモリ領域アドレス
- ④ データ・メモリ・バンク範囲
- ⑤ レジスタ・バンク範囲

【指定方法】

コマンド本体およびサブコマンドのみ入力します。

【使用例】

デバッグ対象デバイスの表示を行う例です。

```
brk : 0>STS D0 ①
Target CPU      μPD75108/108A/108F/P108/P108B/112/112F/116/116F/P116
Program Memory  0-FFFFH
Data Memory     00H-1FFH, F80H-FFFH ②
Memory Bank    0-1, 15
Register Bank   0-3
brk : 0>■
```

- ① デバッグ対象デバイスの表示指定
- ② 対象デバイスの情報を表示

8.4.37 シンボル操作コマンド (SYM)

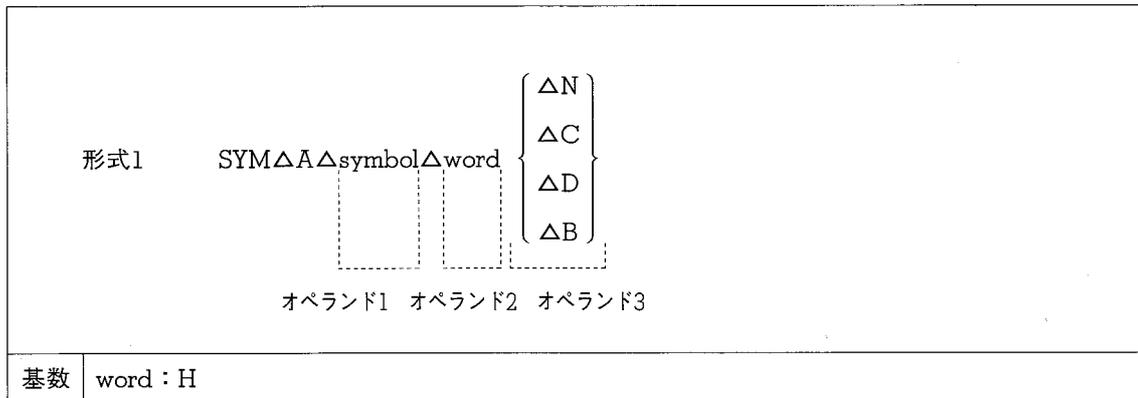
入力形式	
形式1	$\text{SYM}\Delta A\Delta\text{symbol}\Delta\text{word} \left\{ \begin{array}{l} \Delta N \\ \Delta C \\ \Delta D \\ \Delta B \end{array} \right\}$
形式2	$\text{SYM}\Delta C\Delta\text{symbol}\Delta\text{word}$
形式3	$\text{SYM} [\Delta D [\Delta\text{module name}\text{¥}]]$
形式4	$\text{SYM}\Delta E [\Delta\text{symbol}]$
形式5	$\text{SYM} \left[\left\{ \begin{array}{l} \Delta K \\ \Delta L \\ \Delta S \\ \Delta M \end{array} \right\} \right]$
<p>注 ¥はIBM PCシリーズをホスト・マシンとした場合\となります。</p>	
基数	word : H

【機能概要】

SYMコマンドでは、サブコマンドの指定により以下に掲げる操作ができます。

- SYMΔA : アPEND・シンボルの定義
- SYMΔC : アPEND・シンボルのシンボル値の変更
- SYMΔD : シンボルの表示
- SYMΔE : アPEND・シンボルの削除
- SYMΔK : 全シンボルの削除
- SYMΔL : アPEND・シンボルのロード
- SYMΔS : アPEND・シンボルのセーブ
- SYMΔM : カレント・モジュールの指定／表示／変更

(1) アペンド・シンボルの定義



brk : n >	<input type="radio"/>	emu : n >	<input checked="" type="checkbox"/>	trc : n >	<input checked="" type="checkbox"/>
-----------	-----------------------	-----------	-------------------------------------	-----------	-------------------------------------

【機能】

アペンド・シンボルの定義を行います。

本コマンドで定義されたシンボルに対しては、IESYMBOLというモジュール名が付けられます。

シンボル・タイプは、オペランド3で指定します。

N : ナンバ・タイプのアペンド・シンボルを定義

C : コード・タイプのアペンド・シンボルを定義

D : データ・タイプのアペンド・シンボルを定義

B : ビット・タイプのアペンド・シンボルを定義

【指定方法】

(a) オペランド1

symbol : 追加したいシンボル名を指定します。

注意 すでに定義されているアペンド・シンボルと同一名称のシンボルを定義することはできません。

(b) オペランド2

word : シンボル値を指定します。

指定範囲: 0-0FFFFH

(c) オペランド3

シンボル・タイプを指定します。

指定形式	説明
N	ナンバ・タイプのアペンド・シンボルを定義します。
C	コード・タイプのアペンド・シンボルを定義します。
D	データ・タイプのアペンド・シンボルを定義します。
B	ビット・タイプのアペンド・シンボルを定義します。

備考 IE-75001-RのSYMコマンドでは、予約語と同じシンボルを登録した場合、シンボルが重複されているかチェックは行っていません。アペンド・シンボルに予約語を使用した場合、ASM, DASコマンドでは、予約語ではなく、アペンド・シンボルとして登録したシンボルおよびシンボル・タイプを使用します。

【使用例】

(a) アペンド・シンボルの定義

シンボル値が1000の“SYMBOL01”を、コード・タイプで指定する例

```
brk:0>SYM A SYMBOL01 1000 C
```

```
brk:0>■
```

①

① コード・タイプのアペンド・シンボルとして定義

(b) ビット・タイプでシンボル値が100.3の“SYMBOL02”を定義した例

```
brk:0>SYM A SYMBOL02 100.3 B
```

```
brk:0>■
```

①

① “B”を指定することにより、ビット・タイプのアペンド・シンボルとして定義

(c) 予約語をアペンド・シンボルとして定義した例

```

brk : 0>SYM A HL 55 N ①
brk : 0>ASM 1000H ②

Addr Code Label Mnem. Operand
1000 60 NOP =MOV HL,#00H ③
***** Error !! *****
1000 60 NOP =. ④

brk : 0>SYM E ④
brk : 0>ASM 1000H ⑤

Addr Code Label Mnem. Operand
1000 60 NOP =MOV HL,#00H ⑥
1000 8B 00
1002 60 NOP =. ⑥

brk : 0>■
    
```

- ① “HL”（ペア・レジスタと同一名称）というシンボルを定義
- ② 1000H番地からアセンブルを開始する指定（ASMコマンド参照）
- ③ “MOV HL, #00H” と入力するとエラー
- ④ “HL” というシンボルを削除（SYM Eコマンド参照）
- ⑤ 再度，1000H番地からアセンブルを開始する指定
- ⑥ アセンブル

(2) アペンド・シンボルのシンボル値の変更

形式2		SYM△C△symbol△word			
基数	word : H				
brk : n >	○	emu : n >	×	trc : n >	×

【機能】

SYM△Aコマンドによって定義されたアペンド・シンボルのシンボル値を変更します。

【指定方法】

オペランド

symbol : シンボル名を指定します。

注意 **SYM△A**コマンドにより、すでに定義済みのシンボル名以外を指定することはできません。

word : 変更したいシンボル値を指定します。

指定範囲 : 0-0FFFFH

【使用例】

(a) アペンド・シンボルのシンボル値の変更を行った場合

brk : 0 > <u>SYM A SYMBOL01 1000 C</u> <input type="checkbox"/>	①
brk : 0 > <u>SYM C SYMBOL01 2000</u> <input type="checkbox"/>	②
brk : 0 > ■	

① シンボル値が1000のSYMBOL01の定義

② シンボル値を1000から2000へ変更

(b) 指定したアペンド・シンボルが定義されていない場合の例

```
brk : 0>SYM C SYMBOL99 1000①  
  
Symbol not found ②  
  
brk : 0>■
```

- ① SYMBOL99のシンボル値の変更指定
- ② シンボルが定義されていない場合のメッセージ

③ シンボルの表示

形式3 SYM[△D [△module name ¥]]

注 ¥は、IBM PCシリーズをホスト・マシンとした場合\となります。

brk : n>	○	emu : n>	×	trc : n>	×
----------	---	----------	---	----------	---

【機能】

「アペンド・シンボル」、「パブリック・シンボル」および「モジュールのローカル・シンボル」のシンボル名、シンボル値、シンボル・タイプの表示を行います。

オペランドに“module name ¥”を指定した場合、指定されたモジュール名のパブリック・シンボルおよびローカル・シンボルが表示されます。

オペランドが省略された場合は、定義されているすべてのシンボルが表示されます。なお、この場合に限り、サブコマンド“D”を省略することが可能です。

【指定方法】

オペランド

表示を行いたいモジュール名を指定します。

module name ¥ : モジュール名指定

指定されたモジュール名のパブリック・シンボルおよびローカル・シンボルが表示されます。

省略時 : 定義されているすべてのシンボルが表示されます。

【使用例】

(a) アペンド・シンボルが定義されている場合の例

```
brk : 0>SYM D①
  

module : IESYMBOL②
{1000 : SYMBOL01(C)} {2000.2 : SYMBOL02(B)} {3000 : SYMBOL03(C)}
{1100 : SYMBOL04(C)} {4000 : SYMBOL05(C)} {2100 : SYMBOL08(C)}
module : MODULE1③
{0000 : XSYM0001(C)} {0100 : XSYM0002(C)} {0200 : XSYM0007(C)}
{0400 : XSYM0005(C)}
module : MODULE2④
{0A00 : LSYM0001(C)} {0B00 : LSYM0002(C)}
.
.
module : MODULE3⑤
.
.
.
brk : 0>■
```

- ① 全シンボルの表示指定
- ② アペンド・シンボルの表示
- ③ モジュール名 "MODULE1" のローカル・シンボルの表示
- ④ モジュール名 "MODULE2" のローカル・シンボルの表示
- ⑤ モジュール名 "MODULE3" のローカル・シンボルの表示

(b) アペンド・シンボルが定義されていない場合の例

```

brk : 0>SYM D 
  

module : IESYMBOL
module : MODULE1
[0000 : XSYM0001(C)] [0100 : XSYM0002(C)] [0200 : SYM0007(C)]
[0400 : XSYM0005(C)]
module : MODULE2
[0A00 : LSYM0001(C)] [0B00 : LSYM0002(C)]
.
.
module : MODULE3
.
.
.
brk : 0>■

```

①

① 全シンボルの表示指定

(c) モジュール名 "MODULE2" を指定した場合

モジュール名 "MODULE2" のローカル・シンボルだけが表示されます。

```

brk : 0>SYM D MODULE2 
  

module : MODULE2
[0A00 : LSYM001(C)] [0B00 : LSYM002(C)]
.
.
brk : 0>■

```

①

① モジュール名指定

(4) アペンド・シンボルの削除

形式4 SYM△E [△symbol]

brk : n >	○	emu : n >	×	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

指定したアペンド・シンボルの削除を行います。

オペランドが省略された場合は、SYM△Aコマンドで定義されたアペンド・シンボルをすべて削除します。

【指定方法】

オペランド

symbol : 削除したいシンボル名を指定します。

注意 SYM△Aコマンドにより、すでに定義済みのシンボル名以外を指定することはできません。

省略時 : SYM△Aコマンドで定義されたアペンド・シンボルをすべて削除します。

【使用例】

(a) 削除したいシンボル名を指定した場合

brk : 0 > <u>SYM E SYMBOL01</u>	①
brk : 0 > ■	

① “SYMBOL01” の削除指定

(b) オペランドが省略された場合

brk : 0 > <u>SYM E</u>	①
brk : 0 > ■	

① すべてのアペンド・シンボルを削除

⑤ 全シンボルの削除

形式5	SYMΔK
-----	-------

brk:n>	○	emu:n>	×	trc:n>	×
--------	---	--------	---	--------	---

【機能】

登録されている「アペンド・シンボル」、「パブリック・シンボル」、および「モジュールのローカル・シンボル」をすべて削除します。

【指定方法】

コマンド本体およびサブコマンドのみ入力します。

【使用例】

全シンボルの削除を行った例

brk:0>SYM K

brk:0>■

①

① 全シンボルの削除指定

(6) アペンド・シンボルのロード

形式5 SYMΔL

brk : n> ○ emu : n> × trc : n> ×

【機能】

アペンド・シンボル・ファイルからアペンド・シンボルをロードします。

アペンド・シンボル・ファイルのファイル名は、“IE75000.SYM”というファイル名が自動的に設定されます。また、ドライブ・ユニットは、カレント・ドライブが指定されます。

【指定方法】

コマンド本体およびサブコマンドのみ入力します。

【使用例】

(a) アペンド・シンボルのロードを行った例

brk : 0>SYM L^①

brk : 0>■

① アペンド・シンボル・ファイル “IE75000.SYM” をロード

(b) カレント・ドライブに “IE75000.SYM” が存在しない場合

brk : 0>SYM L^①

IE75000.SYM file not found

brk : 0>■

① “IE75000.SYM” がない場合のメッセージ

(7) アペンド・シンボルのセーブ

形式5 SYM△S

brk:n> ○ emu:n> × trc:n> ×

【機能】

アペンド・シンボル・ファイルへアペンド・シンボルをセーブします。

アペンド・シンボル・ファイルのファイル名は、“IE75000.SYM” というファイル名が自動的に設定されます。また、ドライブ・ユニットは、カレント・ドライブが指定されます。

【指定方法】

コマンド本体およびサブコマンドのみ入力します。

【使用例】

(a) アペンド・シンボルのセーブを行った例

brk:0>SYM S[ⓐ]

brk:0>■

①

① “IE75000.SYM” というファイル名で登録

(b) アペンド・シンボルが定義されていない場合

brk:0>SYM S[ⓐ]

no symbol of append

brk:0>■

①

① アペンド・シンボルが定義されていない場合のメッセージ

(c) "IE75000.SYM" というファイルがすでに存在している場合

```
brk:0>SYM S␣
```

```
File already exists. Delete ? (Y or N) Y␣
```

```
brk:0>■
```

①

- ① "IE75000.SYM" というファイルがすでに存在している場合に、上記のようなメッセージを表示

このとき、"Y[␣]"を入力すると、すでに存在しているファイルを削除し、新たに"IE75000.SYM"を作成します。また、"Y[␣]"以外が入力された場合、コマンドは無視されます。ただし、すでに存在するファイルの属性がDIR属性かつR/W属性の場合だけです。

(d) "IE75000.SYM" というファイルを作成できない場合

```
brk:0>SYM S␣
```

```
File already exists.
```

```
brk:0>■
```

①

- ① ファイルの属性がSYS属性、あるいはR/O属性の場合は、上記のようなメッセージを表示
この場合、コマンドは無視されます。

⑧ カレント・モジュールの指定／表示／変更

形式5	SYM△M
-----	-------

brk : n >	<input type="radio"/>	emu : n >	<input type="checkbox"/>	trc : n >	<input type="checkbox"/>
-----------	-----------------------	-----------	--------------------------	-----------	--------------------------

【機能】

設定されているカレント・モジュール名の指定／表示／変更をします。

カレント・ドライブが設定されている場合、カレント・モジュール内のシンボルは、モジュール名を省略して記述できます。

他のモジュールに属するローカル・シンボルを記述する場合、ローカル・シンボル名の前に、モジュール名を付加します。

“ALL SYM”，各モジュール名，“¥”（設定なし）の設定が可能です。初期値は“ALL SYM”に設定されています。

ASM, BRA, BRK, BRS, DAS, TRD, TRFの各コマンドでは、シンボル表示が可能です。この表示は、SYM△Mコマンドの指定によって行います。

シンボル種類 \ モジュール指定	モジュール指定	ALL SYM指定	モジュール指定なし (¥指定)
カレントのパブリック・シンボル	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
カレントのローカル・シンボル	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>
カレント以外のパブリック・シンボル	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
カレント以外のローカル・シンボル	<input type="checkbox"/>	<input type="radio"/>	<input type="checkbox"/>

備考 ALL SYM指定を行った場合、アドレスが同じローカル・シンボルの場合は、どちらか一方の表示になります。

【指定方法】

コマンド本体およびサブコマンドのみ入力します。

【使用例】

(a) カレント・モジュールとして“MOD01”を指定するとき

カレント・モジュールとして，“MOD01”を指定し、引き続きカレント・モジュールを“MOD01”から“MOD02”に変更した例です。

```
brk:0>SYM M␣
ALL SYM =MOD01¥␣
```

①

```
brk:0>SYM M␣
MOD01¥ =MOD02¥␣
```

②

```
brk:0>■
```

- ① “MOD01”をカレント・モジュールに指定
- ② カレント・モジュールを“MOD02”に変更

(b) カレント・モジュールとして，“ALL SYM”を指定するとき

```
brk:0>SYM M K␣
```

①

- ① カレント・モジュールを“ALL SYM”に指定します。
ALL SYM指定に設定され、SYM指定なしの現在と同じシンボル表示になります。

(c) カレント・モジュールを“指定なし”にするとき

```
brk:0>SYM M␣
```

①

```
ALL SYM=¥␣
```

- ① モジュール指定なしに変更。イコール(=)のあとに“¥”を入力してください。

8.4.38 システム再起動コマンド (SYS)

入力形式

形式1 SYS

brk : n >	○	emu : n >	×	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

SYSコマンドは、IE75000 (コントロール・プログラム) の再起動を行います。

【使用例】

(a) システムの再起動を行う例

```
brk : 0 > SYS  ①
IE-75000/1-R Monitor V1.4 [1 Aug 92]
Copyright (C) 1989, 1992 by NEC Corporation

Self check ok

Target CPU      uPD75104/104A/106
Program Memory  0-FFFFH
Data Memory     00H-13FH, F80H-FFFFH
Memory Bank     0-1, 15
Register Bank   0-3
Power on target system (Y/N) Y 
Create new set up mode (Y or N) : Y 
Do you use high speed down load mode ? (Y/N) = N 
Lod object file name = TEST C S 
brk : 0 >
```

① システムの再起動

(b) 暴走し、ハングアップしたときの回避方法

```

PORT XXH, XXH DATA XX
XX
E-CPU ERROR !!

  RESET

IE-75000/1-R Monitor V1.4 [1 Aug 92]
Copyright (C)1989, 1992 by NEC Corporation

Self check ok

Target CPU      uPD75104/104A/106
Program Memory  0-FFFFH
Data Memory     00H-13FH, F80H-FFFH
Memory Bank     0-1, 15
Register Bank   0-3
Power on target system (Y/N) Y
Create new set up mode (Y or N) : Y
Do you use high speed down load mode ? (Y/N) = N
Lod object file name = TEST C S
brk : 0>SYS

IE-75000/1-R Monitor V1.4 [1 Aug 92]
Copyright (C)1989, 1992 by NEC Corporation

Self check ok

Target CPU      uPD75104/104A/106
Program Memory  0-FFFFH
Data Memory     00H-13FH, F80H-FFFH
Memory Bank     0-1, 15
Register Bank   0-3
Power on target system (Y/N) Y
Create new set up mode (Y or N) : Y
Do you use high speed down load mode ? (Y/N) = N
Lod object file name = TEST C S
brk : 0>■

```

① IE-75001-R本体の前面パネルのRESETスイッチを押します。

備考 RESETスイッチを押すと、IE-75001-Rのモニタ・プログラムが起動します。このときプロンプトは“brk : ”となります。この状態で“SYS”を入力することにより、IE75000 (コントロール・プログラム) が再起動し、メッセージが表示され、“brk : n”のプロンプトに戻ります。

8.4.39 トレース表示コマンド (TRD)

入力形式

形式1 TRD [ΔALL] [Δ\$C] [Δoption [, option] [, option]
 Δ\$F . . . [, option] [Δ\$B]]
 Δ\$Q]]

オペランド1 オペランド2 オペランド3 オペランド4

注意 オペランド1の“ALL”を指定する際は、オペランド2では“\$F”を指定することはできません。

brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="checkbox"/>
-----------	-----------------------	-----------	-----------------------	-----------	--------------------------

【機能】

TRDコマンドは、トレース・データを表示します。
 表示形式は、オペランドの組み合わせにより変えることができます。

(a) トレース・データの表示量

オペランドで“ALL”を指定すると、トレース・データすべてを表示します。検索条件 (“\$C”, “\$Q”) が指定されている場合は、検索条件に一致するデータすべてを連続的に表示します。“ALL”を省略すると、現在のポイントから1ページ分のトレース・データを表示します。ページの概念は、検索表示オプションの指定状況により変化します。

表 8-5 ページの概念

検索表示 オプション	1ページのトレース・データ行
なし	12行
\$C	1行
\$Q	12行
\$F	11行

備考 BRMコマンドの条件成立フレーム (トリガ・フレーム) を表示する際は、1ページは11行となります。

(b) トレース・データの検索

オペランドで "\$C", "\$Q", "\$F" を指定することにより、トレース・データの検索表示を行います。

省略すると、検索を行わず、トレース・データすべてを表示します。

\$C : CHKコマンドで設定されたチェック・ポイントのトレース・データ (以後、チェック・ポイント・データという) および、BRMコマンドの条件成立フレーム (以後、トリガ・フレームという) を検索表示します。

\$Q : TRFコマンドで設定された検索条件成立フレーム (以後、!フレームという) およびトリガ・フレームを検索表示します。

\$F : !フレームおよびトリガ・フレームを検索し、前後5フレームを表示します。

!フレームを表示する場合は、行の先頭に "!" を付加します。

トリガ・フレームを表示する場合は、行の先頭に "T" を付加します。

(c) トレース・データの追加表示

通常、本コマンドでは、フレームNO., プログラム・メモリ・アドレス, 命令コード, ラベル, ニモニックを表示します。

optionを指定することにより、データ・メモリへのアクセス状態, 外部センス・クリップ, I/Oポートを追加して表示することができます (表8-6 参照)。

また, "\$B" を指定すると, optionで "EXT" またはI/Oポート名を指定したときに, その値を2進数で表示します。通常は16進表示です。

表8-6 option表

指定形式	説明
"P0", "P1" "P2"-"P15"	I/Oポートのトレース・データの表示指定です。 ただし、イベント・サイクル・トレース・モードの場合は指定できません (8.4.42 トレース・モード指定コマンド (TRM) を参照してください)。 また、指定可能なI/Oポート名は対象デバイスによって異なります。
"EXT"	外部センス・クリップのトレース・データの表示指定です。
"DMEM"	データ・メモリへアクセスした際のトレース・データの表示指定です。 データ・メモリのアクセス・アドレス データ・メモリのアクセス・データ データ・メモリへのリード/ライト状態 が表示されます

備考 オプションは順不同で指定することができますが、指定可能な数は、限定されます (【使用規約】の(a)を参照してください)。

T

(d) メニュー・モード

オペランドで“ALL”を省略すると、メニュー・モードになります。

検索表示を行わない場合は、フレーム総数を表示します。

“\$C”指定の場合は、チェック・ポイント・データ数を表示します。

“\$Q”, “\$F”指定の場合は、!フレーム数を表示します。

メニュー・モードのキー入力待ち状態における入力と動作の関係は次のとおりです。

表 8-7 メニュー・モード時の入力と動作

入力	検索表示 オプション	動 作
“L”	なし	最後のページを表示します。
	\$C	最後のチェック・ポイント・データを表示します。
	\$Q	最後の!フレーム・ページを表示します。
	\$F	最後の!フレームを中心とするページを表示します。
“F”	なし	最初のページを表示します。
	\$C	最初のチェック・ポイント・データを表示します。
	\$Q	最初の!フレーム・ページを表示します。
	\$F	最初の!フレームを中心とするページを表示します。
“T”	共通	トリガ・フレームを中心とするページを表示します。
“+” または <cr>	なし	直前に表示したページの次のページを表示します。
	\$C	直前に表示したチェック・ポイント・データの次のチェック・ポイント・データを表示します。
	\$Q	直前に表示した!フレーム・ページの次のページを表示します。
	\$F	直前に表示したページの次のトリガ・フレームまたは!フレームを中心とするページを表示します。
“-”	なし	直前に表示したページの前のページを表示します。
	\$C	直前に表示したチェック・ポイント・データの前のチェック・ポイント・データを表示します。
	\$Q	直前に表示した!フレーム・ページの前のページを表示します。
	\$F	直前に表示したページの前のトリガ・フレームまたは!フレームを中心とするページを表示します。
“.”	共通	トレース表示コマンドを終了します。

【指定方法】

(a) オペランド1

連続表示オプションの指定を行います。

ALL : トレース・データ内で、検索条件に一致するデータすべての表示を指定します。

表示終了後はコマンド待ちになります。

省略時：現在のポインタから1ページ表示し、メニュー・モードとなります。

(b) オペランド2

検索表示オプションの指定を行います。

\$C : チェック・ポイント・データおよびトリガ・フレームを検索し表示するモードです。

\$Q : !フレームおよびトリガ・フレームを検索し、表示するモードです。

\$F : !フレームおよびトリガ・フレームを検索し、前後5フレームを表示するモードです。

省略時：非サーチ・モード（検索は行わない）となります。

(c) オペランド3

追加表示オプションの指定を行います。

option : 表示したいオプションを表7-6の中から自由に組み合わせて、追加することができます。ただし、指定可能な数は限られています（【使用規約】の(a)を参照してください）。

省略時：追加表示オプションの表示は行われません。

(d) オペランド4

2進数表示オプションの指定を行います。

\$B : \$Bを指定すると、optionでI/OポートまたはEXTを指定したときに、その値を2進数で表示します。

省略時：16進数で表示されます。

【使用規約】

(a) I/Oポートの表示指定可能な数

本コマンドでI/Oポートのトレース・データを表示する場合、オペランド部のI/Oポート指定以外の表示指定（"DMEM", "EXT" あるいは, "\$B"）により、一度に画面上に表示できるI/Oポート・トレース・データの数が増減します。

表 8-8 option指定個数表

"DMEM" 指定	"Pn" 指定	"EXT" 指定	"\$B" 指定
—	"Pn"・・・6個	—	—
	"Pn"・・・5個	"EXT"	
	"Pn"・・・5個	—	"\$B"
	"Pn"・・・3個	"EXT"	
"DMEM"	"Pn"・・・3個	—	—
	"Pn"・・・2個	"EXT"	
	"Pn"・・・2個	—	"\$B"
	"Pn"・・・1個	"EXT"	

(b) "ALL" を指定した場合

連続表示オプション "ALL" を指定した場合、検索表示オプションとして "\$F" を指定することはできません。

【使用例】

(a) イベント・サイクルごとのトレース・データをすべて表示する例

```
brk:0>TRD ALL␣

```

Frame	PA	PD	Label	Mnemonic
000	0100	99 10		SEL MBO
001	0102	89 00		MOV XA, #0000H
003	0104	8B 30		MOV HL, #0030H
005	0106	8A		INCS HL
007	0107	AA 5A		MOV XA, HL

```
brk:0>■
```

(b) optionに "DMEM" を指定して、イベント・サイクルごとのトレース・データをすべて表示する例

```
brk:0>TRD ALL DMEM␣

```

Frame	PA	PD	MA	MD	MRW	Label	Mnemonic
000	0100	99 10	---	--	---		SEL MBO
001	0102	89 00	---	--	---		MOV XA, #0000H
002			000	00	MWR		
003	0104	8B 30	---	--	---		MOV HL, #0030H
004			002	30	MWR		
005	0106	8A	---	--	---		INCS HL
006			002	31	MWR		
007	0107	AA 5A	---	--	---		MOV XA, HL
008			000	31	MWR		

```
brk:0>■
```

(c) マシン・サイクルごとのトレース・データをすべて表示した例

```
brk:0>TRD ALL␣

Frame PA PD Label Mnemonic
000 0100 99 10 SEL MBO
002 0102 89 00 MOV XA, #0000H
004 0104 8B 30 MOV HL, #0030H
006 0106 8A INCS HL
007 0107 AA 5A MOV XA, HL

brk:0>■
```

(d) optionに“DMEM”を指定して、マシン・サイクルごとのトレース・データをすべて表示する例

```
brk:0>TRD ALL DMEM␣

Frame PA PD MA MD MRD Label Mnemonic
000 0100 99 10 --- -- --- SEL MBO
001 --- -- ---
002 0102 89 00 --- -- --- MOV XA, #0000H
003 000 00 MWR
004 0104 8B 30 --- -- --- MOV HL, #0030H
005 002 30 MWR
006 0106 8A 002 31 MRW INCS HL
007 0107 AA 5A --- -- ---
000 31 MWR MOV XA, HL

brk:0>■
```

(e) チェック・ポイント・データが存在している場合の表示例

```
brk : 0 > TRD ALL 

Frame  PA    PD    Label    Mnemonic
000  0100  99 10          SEL     MBO
002  0102  89 00          MOV     XA, #0000H
004  ***CHK REG*** XA=00 HL=00 DE=00 BC=00
005  ***CHK REG*** XA'=00 HL'=00 DE'=00 BC'=00 RBS=0 MBS=0
006  ***CHK REG*** RBE=0 MBE=0 CY=0 IST1=0 IST0=0 SP=000 PC=000
007  0104  8B 30          MOV     HL, #0030H
      .
      .
      .

brk : 0 > ■
```

Frame No.004-006には、チェック・ポイント・データが存在していることが確認できます。
 チェック・ポイント・データは、次のように種別表示されます。

表 8-9 チェック・ポイント・データの種別

指 定 種 別	表 示 形 式
"REG" 指定	***CHK REG***
"special register" 指定	***CHK SPR***
"データ・メモリ" 指定	***CHK MEM***

備考 チェック・ポイントの設定法は、**8.4.7 チェック・ポイント設定コマンド (CHK)**
 を参照してください。

(f) optionにI/Oポート指定した場合のトレース・データを表示する場合

```
brk : 0>TRD ALL P0, P1, P2
```

Frame	PA	PD	P0	P1	P2	Label	Mnemonic
000	0109	8A	F	F	F		INCS HL
001	010A	AA 10	F	F	F		MOV @HL, XA
002			F	F	F		MOV
003	010C	89 34	F	F	F		MOV XA, #0034H
004			F	F	F		

```
brk : 0>
```

(g) 2進表示でオプション "\$B" を指定し、追加表示オプションとして "EXT" とI/Oポート名を指定した場合の表示例

```
brk : 0>TRD ALL P0, P1, EXT $B
```

Frame	PA	PD	EXT	P0	P1	Label	Mnemonic
000	0109	8A	11111111	1111	1111		INCS HL
001	010A	AA 10	11111111	1111	1111		MOV @HL, XA
002			11111111	1111	1111		
003	010C	89 34	11111111	1111	1111		MOV XA, #0034H
004			11111111	1111	1111		

```
brk : 0>
```

(h) 連続オプション "ALL" を省略し、トレース・データを1ページ表示する例

```
brk : 0>TRD DMEM[]

Frame PA      PD      MA      MD MRW      Label  Mnemonic
018 0102      89 11    ---    --    ---          MOV   XA, #0011H
019                000      11 MWR
01A ***CHKREG*** XA=11      HL=02    DE=34    BC=00
01B ***CHKREG*** XA'=00      HL'=00    DE'=00    BC'=02 RBS=0 MBS=0
01C ***CHKREG*** RBE=1      MBE=0    CY=0     IST1=0 IST0=1 SP=0F2 PC=0104
01D 0104      8A      002     03 MRW          INCS   HL
01E 0105      AA 4A    ---     --    ---          SKE   XA, HL
01F                002     03 MRD
020 0107      AB 01 02 ---     --    ---          BR    !0102H
021                ---     --    ---
022                ---     --    ---
023 0102      89 11    ---     --    ---          MOV   XA, #0011H

Total frame=0D9          (L/F/T/+/cr/-/Frame No./.) ? ■
```

(i) 検索表示オプション "\$C" を指定して、トレース・データを1ページ表示する例

```
brk : 0>TRD $C DMEM[]

Frame PA      PD      MA      MD MRW      Label  Mnemonic
00F ***CHK REG*** XA=00      HL=00    DE=00    BC=00

Total frame=107      CHK. frame=017      (L/F/T/+/cr/-/Frame No./.)? ■
```

(j) 検索表示オプション "\$Q" を指定して、トレース・データを1ページ表示する例

```
brk : 0>TRD $Q DMEM
```

Frame	PA	PD	MA	MD	MRW	Label	Mnemonic
I025	1A08	89 34	---	--	---		MOV XA, #0034
I026			000	34	MWR		
I02A	1A08	89 34	---	--	---		MOV XA, #0034
I02F			000	34	MWR		
I030	1A08	89 34	---	--	---		MOV XA, #0034
I034			000	34	MWR		
I035	1A08	89 34	---	--	---		MOV XA, #0034
I039			000	34	MWR		
I03A	1A08	89 34	---	--	---		MOV XA, #0034
I03E			000	34	MWR		
I03D	1A08	89 34	---	--	---		MOV XA, #0034
I043			000	34	MWR		

Total frame=0d9 lframe=023 (L/F/T/+/cr/-/Frame No./.) ? ■

(k) 検索表示オプション "\$F" を指定して、トレース・データを1ページ表示する例

```
brk : 0>TRD $F DMEM
```

Frame	PA	PD	MA	MD	MRW	Label	Mnemonic
078			---	--	---		
079	1A04	9D 88	---	--	---		SET1 IRQBT
07A			FB8	3	MRW		
07B	1A06	9D 82	---	--	---		EI
07C			FB2	8	MRW		
I07D	1A08	89 34	---	--	---		MOV XA, #0034H
07E			000	34	MRW		
07F	1A0A	8B 30	---	--	---		MOV HL, #0030H
080			002	30	MWR		
081	1A0C	8A	002	31	MRW		INCS HL
082	1A0D	AA 5A	---	--	---		MOV XA, HL

Total frame=107 lframe=023 (L/F/T/+/cr/-/Frame No./.) ? ■

(1) トレース・データを指すポインタの動作

マシン・サイクルごとのトレース・データを表示する例です。

```

brk : 0>TRP①

Total frame number=107
Display frame number=018

brk : 0>TRP C②

brk : 0>TRD DMEM③

Frame   PA      PD  MA   MD MRW   Label  Mnemonic
00C           ---  --  ---
00D  1A00   9C B2  ---  --  ---          DI
00E           FB2   0  MRW
00F *** CHK REG ***XA=00  HL=00  DE=00 BC=00
010 *** CHK REG ***XA'=00  HL'=00  DE'=00 BC'=00 RBS=0 MBS=0
011 *** CHK REG ***RBE=0  MBE=0  CY=0  IST1=0  IST0=0  SP=000 PC = 0000
012  1A02   9D 98  ---  --  ---          SET1  IEET
013           FB8   3  MRW
014  1A04   9D 88  ---  --  ---          SET1  IRQBT
015           FB8   3  MRW
016 ***CHKREG*** XA=00  HL=00  DE=00 BC=00
017 ***CHKREG*** XA'=00  HL'=00  DE'=00 BC'=00 RBS=0 MBS=0

Total frame=107 (L/F/T/+/cr/-/Frame No./.)?16④

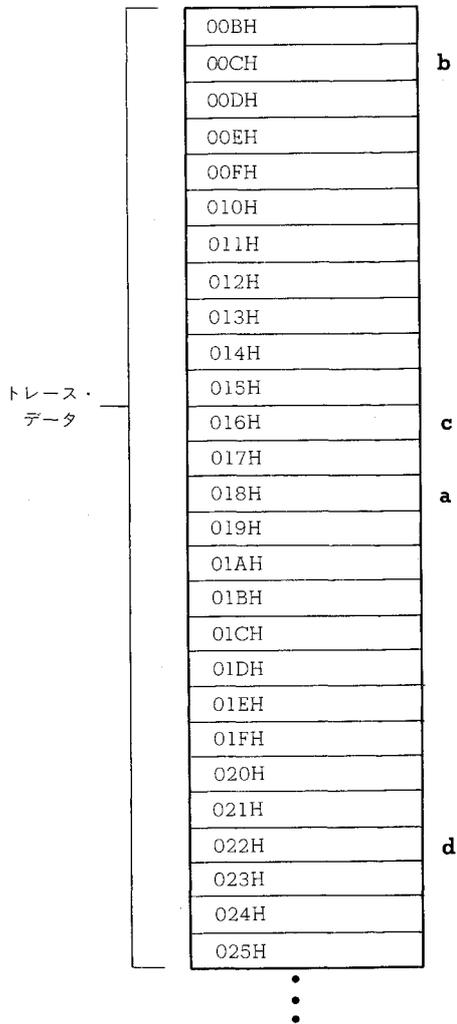
Frame   PA      PD  MA      MRW   Label  Mnemonic
016 *** CHK REG ***XA=00  HL=00  DE=00 BC=00 RBS=0 MBS=0
017 *** CHK REG ***XA'=00  HL'=00  DE'=00 BC'=00 IST0=0 SP=000 PC=000
018 *** CHK REG ***RBE=0  MBE=0  CY=0  IST1=0
019  1A06   9D B2  ---  --  ---          EI
01A           FB2   8  MRW
01B  1A08   89 34  ---  --  ---          MOV XA, #0034H
01C           000  34  MWR
01D >           0EA  12  MWR
01E           0E8  0A  MWR
01F           0E6  1A  MWR
020 2000   89 55  ---  --  ---          MOV XA, #0055H
021           000  55  MWR

Total frame=107 (L/F/T/+/cr/-/Frame No./.)?■⑤
    
```

- ① 現在のポインタの表示指定
- ② ポインタの移動指定
- ③ トレース・データの表示指定
- ④ メニュー・モードによる指定
- ⑤ メニュー・モードの入力待ち状態

前ページの例で、ポインタの動作を次の図に示します。

図8-4 トレース・ポインタ図



現ポインタは、“TRP” コマンド①により、a (018H) にあることがわかります。

“TRP” コマンド②により、1ページ分前bに移動させます。

“TRD DMEM” ③により、bより1ページ分 (017Hまで) 表示します (このときポインタはaに移動します)。表示後は連続表示オプション (“ALL”) の指定がないので、メニュー・モードとなります。

④のように、メニュー・モードで “16” 直接フレーム・ナンバを指定すると、ポインタはc (016H) に移動し、cより1ページ分 (021Hまで) 表示します (表示後のポインタはdにあります)。

フレーム・ナンバ01DHには “>” が表示されています。これは、割り込みがかかったことを示します。

(m) スキップ命令を表示する例

スキップ命令によりスキップされた命令は、プログラム・アドレスが “[]” で囲まれて表示されます。スキップされる命令はトレースされますが実行はしません。

Frame	PA	PD	Label	Mnemonic
00E	1002	CF		DECS B
00F	1003	9A 07		SKE B,#0000H
011	(1005)	*****	SKIP *****	
013	1008	A3 30		MOV A,0030H

8.4.40 トレース・データ検索条件設定コマンド (TRF)

入力形式								
形式1	TRF [Δ PA = $\left(\begin{array}{c} \text{[addrX} \\ \text{partition1} \end{array} \right)$] [Δ PD = data1] [Δ MA = $\left(\begin{array}{c} \text{[addrX} \\ \text{partition2} \end{array} \right)$]] $[\Delta$ MD = data2] [Δ MRW = status] [Δ Pn = data3] $[\Delta$ EXT = data4]							
基数	addr1-2 : H partition1-2 : H data1-4 : H							
<table border="1"> <tr> <td>brk : n ></td> <td><input type="radio"/></td> <td>emu : n ></td> <td><input type="radio"/></td> <td>trc : n ></td> <td><input type="checkbox"/></td> <td>×</td> </tr> </table>		brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="checkbox"/>	×
brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="checkbox"/>	×		

【機能】

TRFコマンドは、トレース・データの検索条件を設定します。

検索条件の設定は、コマンド入力時1ラインで設定する方法と、対話形式で設定する方法があります。

対話形式で設定する場合、現在の検索条件の表示後、各条件別に順次設定します。

TRF<cr>と入力した場合、現在設定されている検索条件を表示し、対話形式の設定モードとなります。

【指定方法】

なお、本コマンドにより設定した検索条件は、TRDコマンドで検索表示する際に使用されます。詳細は**8.4.39 トレース表示コマンド (TRD)** を参照してください。

オペランド

PA=addrX partition1 : プログラム・メモリの検索アドレスを指定します。アドレス範囲はマスク指定とパーティション指定ができます。

アドレス指定範囲 : 0-0FFFFH

マスク指定範囲 : 0-OXXXXH

省略時 : 条件として設定されません。

初期値 : 0XXXXH

PD=data1 : プログラム・メモリの検索データを指定します。

データ値指定範囲 : 0-0FFH

マスク指定範囲 : 0-OXXH

省略時 : 条件として設定されません。

初期値 : OXXH

MA=addrX : データ・メモリ・アクセスの検索アドレスを指定します。なお、アドレス
partition2 の範囲は対象デバイスによって異なります。

アドレス指定範囲 : 0-OFFFH

マスク指定範囲 : 0-OXXXH

省略時 : 条件として設定されません。

初期値 : 0XXXH

MD=data2 : データ・メモリ・アクセスの検索データを指定します。

データ値指定範囲 : 0-OFFH

マスク指定範囲 : 0-OXXH

省略時 : 条件として設定されません。

初期値 : 0XXH

MRW=status : データ・メモリへのアクセス状態検索ステータスを指定します。

指定形式	アクセス状態
MRD	メモリ・リード
MWR	メモリ・ライト
MRW	メモリ・リード・モディファイ・ライト
NC	すべてのリード・ライト

省略時 : 条件として設定されません。

初期値 : NC

Pn=data3 : I/Oポートの検索データを指定します。なお、指定可能なI/Oポート名は対
象デバイスによって異なります。

データ値指定範囲 : 0-OFH

マスク指定範囲 : 0-OXH

省略時 : 条件として設定されません。

初期値 : P0-P15まで、すべてOXH

EXT=data4 : 外部センス・クリップの検索データを指定します。

データ値指定範囲 : 0-OFFH

マスク指定範囲 : 0-OXXH

省略時 : 条件として設定されません。

初期値 : 0XXH

省略時 : オペランドが省略された場合は、現在、設定されている検索条件を表示し、
対話形式の設定モードとなります。

【使用規約】

I/Oポートの検索データの指定は、イベントごとのサイクル・モード時には、無効となります。また、TRXトレース時にプログラム・フェッチ系（PA, PD）とデータ・メモリ・アクセス系（MA, MD, MRW）を同時に検索条件として設定すると、検索することができません。

【使用例】**(a) 検索条件の設定**

プログラム・メモリ・アドレスの0H番地から1FFFH番地を操作したトレース・データを検索するように、設定を行う例です。

```
brk : 0> TRF PA=0, 1FFF 
```

```
brk : 0> ■
```

(b) オペランド以下を省略したとき

オペランド以下を省略するとメニュー・モードとなり、対話形式で検索条件の設定を行うことができます。

```

brk:0>TRF①
PA 0, 1FFF = OXXX②
PD OXXH = ③
MA OXXXH = ③
MD OXXH = ③
MRW NC = ③
P0 OX = ③
P1 OX = ③
P2 OX = ③
P3 OX = ③
P4 OX = ③
P5 OX = ③
P6 OX = ③
P7 OX = ③
P8 OX = ③
P9 OX = ③
P10 OX = ③
P11 OX = ③
P12 OX = ③
P13 OX = ③
P14 OX = ③
P15 OX = ③
EXT OXX = ③

brk:0>■

```

①

②

③

注

- ① メニュー・モード指定
- ② 新しくOXXXHを設定
- ③ 現在のデータは変更しない。

注 対象デバイスによって表示が異なります。

8.4.41 トレーサ起動コマンド (TRG)

入力形式

形式1 TRG

brk:n>	×	emu:n>	○	trc:n>	×
--------	---	--------	---	--------	---

【機能】

トレーサ起動コマンドは、トレーサの再起動を行います。

トレーサの再起動は、エミュレーション・デバイスが動作中で、トレーサのみ停止している状態（エミュレーション・モード）で行います。

本コマンドが実行されると、トレース・モードとなります。

【指定方法】

コマンド本体のみを入力します。

【使用規約】

本コマンドは、ブ레이크・モードあるいは、トレース・モードでは使用できません。

【使用例】

トレーサを再起動する例です。

```
emu:0>TRG
```

```
Tracer start
```

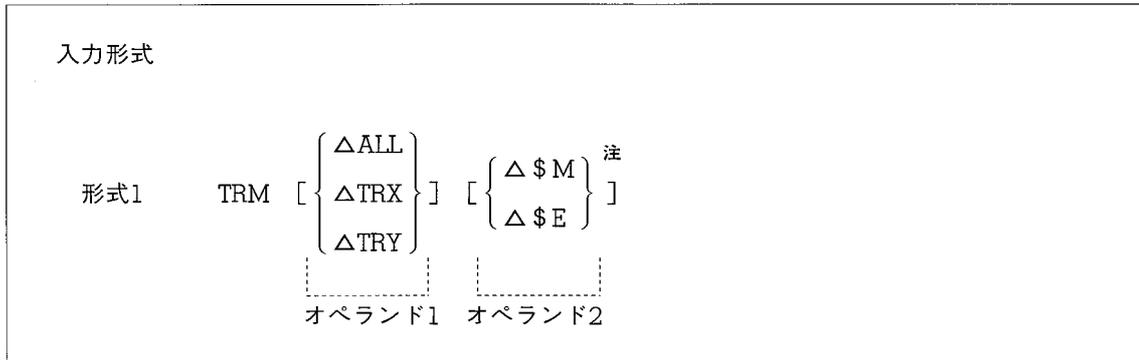
```
trc:0>■
```

①

① トレーサの再起動指定

トレーサの再起動は、プロンプトにより確認することができます。

8.4.42 トレース・モード指定コマンド (TRM)



brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="checkbox"/>
-----------	-----------------------	-----------	-----------------------	-----------	--------------------------

注 オペランドで“TRX”あるいは、“TRY”を指定するときは、オペランド2では、“\$M”を指定することはできません。

【機能】

TRMコマンドは、リアルタイム・トレーサのトレース条件モードおよびトレース・サイクル・モードを指定します。

- (a) “ALL”を指定するとエミュレーション中のプログラムの実行を指定されたサイクルごと (“ Δ \$M”または“ Δ \$E”) に、無条件トレースします。
- (b) “TRX”あるいは“TRY”を指定すると、それぞれのコマンドで設定された条件に従い、イベント・サイクルごとにトレースします。詳細は8.4.44 クオリファイ・トレース条件設定コマンド (TRX) , 8.4.45 セクション・トレース条件設定コマンド (TRY) を参照してください。
- (c) TRM<cr>と入力した場合、現在のトレース・モードを表示します。

【指定方法】

(a) オペランド1

トレース条件モードの指定を行います。

ALL : 無条件トレース指定。

エミュレーション中、リアルタイムでトレーサが起動します。

TRX : クオリファイ・トレース指定

トレース・ポイントのイベント条件が成立時にかぎり、トレースします。詳細はTRXコマンドを参照してください。

TRY : セクション・トレース指定

トレース開始ポイントのイベント条件成立時から、トレース終了ポイントのイベント条件成立時までをトレースします。詳細は**8.4.45 セクション・トレース条件設定コマンド (TRY)** を参照してください。

初期値: ALL

(b) オペランド2

トレース・サイクル・モードの指定を行います。

\$M : マシン・サイクルごとのトレース・モード

1マシン・サイクルごとに、プログラム・アドレス、データ・メモリ・バス、リード・ライ
ト信号、ポート、外部センス・クリップ信号の状態をトレースします。

\$E : イベント・サイクルごとのトレース・モード

イベント発生時ごとに、必要な情報だけをトレースします。ただし、ポートのトレースは
行いません。

初期値: \$M

【使用規約】

トレース・サイクル・モードが、マシン・サイクルごとのトレース・モード (“\$M”) の場合は、
トレース条件として、クオリファイ・トレース (“TRX”) とセクション・トレース (“TRY”) を
指定することはできません。

【使用例】

マシン・サイクルで無条件トレースを行うように設定を行う例です。

brk: 0>TRM	①
TRX \$E	
brk: 0>TRM ALL \$M	②
brk: 0>TRM	
ALL \$M	③

- ① 現在のトレース・モードを表示
- ② トレース・モードの指定
- ③ 現在のトレース・モードを表示

8.4.43 トレース・ポインタ操作コマンド (TRP)

入力形式	
形式1	TRP [$\left\{ \begin{array}{l} \Delta\text{word} \\ \Delta F \\ \Delta L \\ \Delta T \end{array} \right\}$]
基数	word : H
brk : n >	<input type="radio"/>
emu : n >	<input type="radio"/>
trc : n >	<input type="checkbox"/>
	×

【機能】

TRPコマンドは、トレース・ポインタを任意のトレース・メモリに移動します。

(a) トレース・ポインタの移動/表示

TRP F/L<cr>と入力した場合、トレース・ポインタをトレース・メモリの最初/最後に移動します。

TRP T<cr>と入力した場合、トレース・ポインタをBRMコマンドの条件成立フレーム（トリガ・フレーム）に移動します。ただし、トリガ・フレームが存在しない場合は、本コマンドは意味を持ちません。

オペランドでトレース・ポインタの移動数を指定した場合、現在のポインタから±7FFHの範囲で、ポインタを移動します。

TRP<cr>と入力した場合、トレース・メモリに記憶されているトレース・フレーム総数と、現在のトレース・ポインタ値を表示します。

(b) 本コマンドとTRDコマンドとの関係

TRDコマンドでは、本コマンドにより指定したポインタから表示を行います。ただし、連続表示オプション“ALL”が省略された場合に限りです。

本コマンドで“T”を指定した場合、TRDコマンドではトリガ・フレームを中心に11行表示を行います。

また、検索表示オプション“\$C”、“\$Q”、“\$F”を指定した場合は、検索表示を行うため、現在のトレース・ポインタ値と実際に表示されるポインタ値との間にずれが生じる可能性があります。詳細は8.4.39 トレース表示コマンド (TRD) を参照してください。

(c) 絶対番地指定

トレース・データがあるときのみ、絶対アドレスの入力ができます。

【指定方法】

オペランド

word : 現在のトレース・ポインタからの移動数を指定します。

ポインタ移動数: ±1H-7FFH

F : トレース・ポインタをトレース・メモリの最初に移動します。

L : トレース・ポインタをトレース・メモリの最後に移動します。

T : トレース・ポインタをトリガ・フレームに移動します。

トリガ・フレームが存在しない場合は、本コマンドは意味を持ちません。

省略時: オペランドが省略された場合は、現在トレースされているデータの総数およびトレース・ポインタを16進で表示します。

【使用例】

(a) トレース・ポインタの移動

```
brk : 0 > TRP 20
```

①

```
brk : 0 > TRP L
```

②

```
brk : 0 > ■
```

① トレース・ポインタを20H進める指定

② トレース・ポインタをトレース・メモリの最後に移動

(b) オペランドを省略したとき

オペランドが省略された場合は、現在トレースされているデータの総数と、トレース・ポインタが表示されます。

```
brk : 0 > TRP 
  
Total frame number    = 200
  
Display frame pointer = 012
  
brk : 0 > 
```

①
②

- ① トレースされたフレームの総数を表示
- ② トレース・ポインタ値を表示

(c) 絶対番地指定

```
brk : 0 > TRP 050H 
```

①

- ① 表示ポインタ050Hを指定します。このコマンドは、トレース・データがあるときのみ有効です。

8.4.44 クオリファイ・トレース条件設定コマンド (TRX)

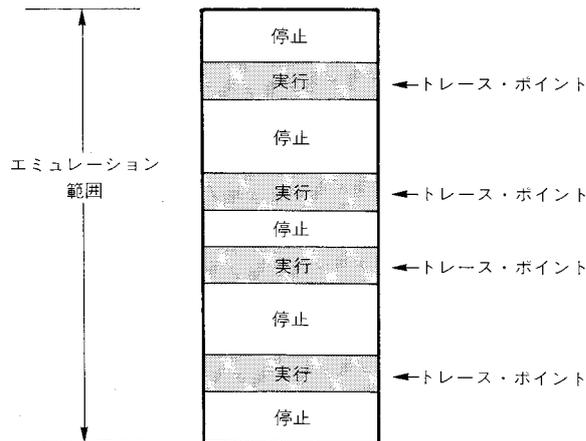
入力形式
形式1 TRX [Δ BR?][Δ BR?][Δ BR?][Δ BR?][Δ BR?][Δ BR?]

brk : n >	<input type="radio"/>	emu : n >	<input type="radio"/>	trc : n >	<input type="checkbox"/>
-----------	-----------------------	-----------	-----------------------	-----------	--------------------------

【機能】

TRXコマンドは、イベント条件レジスタに設定されたイベント条件を、クオリファイ・トレースのイベント条件として指定します。

図 8-5 トレーサの状態図



クオリファイ・トレースとは、必要な情報だけを限定トレースするもので、本コマンドではイベント条件レジスタに設定されたイベント条件をトレース・ポイント条件として指定します。

図 8-5 のように、トレース・ポイント条件成立時にかぎり、トレーサにトレース・データが書き込まれます。

TRX<cr>と入力した場合、現在の設定されている条件を表示します。

OUTコマンドでBRA3, BRA4が設定されている場合、TRXコマンドでBRA3, BRA4を設定することはできません。

【指定方法】

オペランド

BR? : 次のイベント条件レジスタのうち、最大6個までを指定します。

BRA1, BRA2, BRA3, BRA4

BRS1, BRS2, OFF

備考 “OFF” が指定された場合は、トレーサは起動しません。

初期値：OFF

省略時：オペランドが省略された場合は、現在の設定されている条件を表示します。

【使用規約】

本コマンドで設定した条件でクオリファイ・トレースするためには、あらかじめトレース・モード指定コマンド (TRM) でクオリファイ指定 (“TRX”) を選定しなければなりません。クオリファイ・トレース情報には、スキップ動作が行われても、スキップを行ったという表示 “**** *SKIP****” は出ません。詳細は8.4.42 トレース・モード指定コマンド (TRM) を参照してください。

【使用例】

(a) クオリファイ・トレース条件の指定

データ・メモリ・アドレスの0100H番地がアクセスされたときにトレーサにトレース・データを書き込むように指定を行った例です。

```
brk:0>BRA 1 MA=0100
```

①

```
brk:0>TRX BRA1
```

②

```
brk:0>TRM TRX
```

③

```
brk:0>RUN N
```

④

- ① BRA1にデータ・メモリ・アドレスを設定
- ② クオリファイ・トレース条件としてBRA1を指定
- ③ クオリファイ・トレースを指定
- ④ エミュレーションの実行指定

(b) オペランドを省略したとき

オペランドを省略すると、現在の設定内容が表示されます。

```
brk : 0>TRX   
  
BRA1  
  
brk : 0>■
```

①

① クオリファイ・トレース条件としてBRA1を設定

8.4.45 セクション・トレース条件設定コマンド (TRY)

入力形式

形式1 TRY[{ ΔE }][$\Delta BR?$]

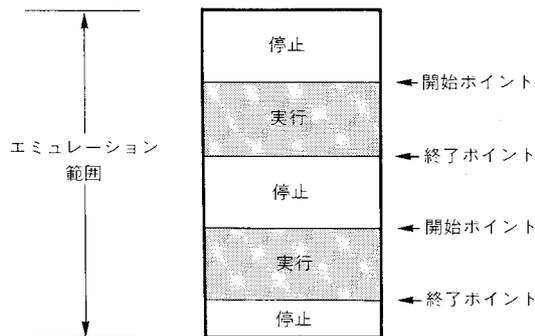
ΔD

brk : n >	○	emu : n >	○	trc : n >	×
-----------	---	-----------	---	-----------	---

【機能】

TRYコマンドは、イベント条件レジスタに設定されたイベント条件をセクション・トレースのイベント条件として指定します。

図 8-6 トレーサの状態図



セクション・トレースとは、部分的にトレースするもので、本コマンドではイベント条件レジスタに設定されたイベント条件を、開始ポイント条件、終了ポイント条件として指定します。

図 8-6 のように、開始ポイント条件成立時にトレーサは起動し、終了ポイント条件成立時にトレーサは停止します。この間、トレーサにトレース・データが書き込まれます。

TRY<cr>と入力した場合、現在の設定されている条件を表示します。

OUTコマンドでBRA3, BRA4が設定されている場合、TRYコマンドでBRA3, BRA4を設定することはできません。

【指定方法】

(a) サブコマンド

E : 開始ポイントの条件を指定します。

オペランドで指定されたイベント条件レジスタのいずれかが条件を満たしたときに、トレーサが起動します。

D : 終了ポイントの条件を指定します。

オペランドで指定されたイベント条件レジスタのいずれかが条件を満たしたときに、トレーサが停止します。

(b) オペランド

BR? : 次のイベント条件レジスタの中から指定を行います。ただし、データ・メモリ・アクセス系 (BRA1-BRA4) とプログラム・フェッチ系 (BRS1-BRS2) を混在して指定することはできません。

BRA1, BRA2, BRA3, BRA4

BRS1, BRS2, OFF

備考 “OFF” が指定された場合は、開始ポイント条件あるいは終了ポイント条件として無効となります。

初期値：開始ポイント条件→OFF

終了ポイント条件→OFF

【使用規約】

- (a) 本コマンドで設定した条件でトレーサを起動、停止させるには、あらかじめトレース・モード指定コマンド (TRM) でセクション指定 (“TRY”) を選定しなければなりません。詳細は、**8.4.42** **トレース・モード指定コマンド (TRM)** を参照してください。
- (b) 開始ポイントの条件として、データ・メモリ・アクセス系 (BRA1-BRA4) を指定した場合、終了ポイントの条件はデータ・メモリ・アクセス系 (BRA1-BRA4) を指定してください。
- (c) また、開始ポイントの条件として、プログラム・フェッチ系 (BRS1-BRS2) を指定した場合、終了ポイントの条件はプログラム・フェッチ系 (BRS1-BRS2) を指定してください。
- (d) プログラム・フェッチ系 (BRS1-BRS2) を終了ポイントに指定した場合、最後のトレース表示だけが次の場合正常に表示されない場合があります。
 - ・終了ポイントの命令が2バイトまたは3バイト命令の場合

上記以外の指定を行った場合はトレーサの停止条件は無効となります。

【使用例】

(a) トレーサ起動, 停止条件の指定

セクション・トレースの開始ポイント, 終了ポイントとしてプログラム・メモリ・イベント・レジスタ (BRS1, BRS2) を設定する例です。

この例ではプログラム・アドレスが2000H番地でトレーサがスタートし, 3000H番地でストップします。

brk : 0 > <u>BRS 1 P 2000H</u>	①	
brk : 0 > <u>BRS 2 3000H</u>	②	
brk : 0 > <u>TRY E BRS1</u>	③	
brk : 0 > <u>TRY D BRS2</u>	④	
brk : 0 > <u>TRM TRY</u>	⑤	
brk : 0 > <u>BRM OFF</u>	⑥	★
brk : 0 > <u>RUN N</u>	⑦	

- ① BRS1に開始アドレスを設定
- ② BRS2に終了アドレスを設定
- ③ BRS1をトレース有効点 (開始ポイント) に指定
- ④ BRS2をトレース無効点 (終了ポイント) に指定
- ⑤ セクション・トレースを指定
- ⑥ ブレーク条件なしを設定
- ⑦ エミュレーションの実行指定

(b) サブコマンド以下を省略するとき

サブコマンド以下を省略すると, 現在の設定内容が表示されます。

brk : 0 > <u>TRY</u>	
Enable condition : BRS1	①
Disable condition : BRS2	②
brk : 0 > ■	

- ① トレース開始ポイントとしてBRS1を設定
- ② トレース終了ポイントとしてBRS2を設定

(c) 開始ポイント/終了ポイント条件の変更

開始ポイント条件/終了ポイント条件として、それぞれ別系列のイベント条件レジスタを設定する例です。

```
brk : 0 > TRY  ①  
  
Enable condition : OFF  
Disable condition : OFF  
  
brk : 0 > TRY E BRA1  ②  
  
brk : 0 > TRY D BRS1  ③  
  
TRY condition unmatched ④  
  
brk:0 > ■
```

- ① 現在の設定内容の表示指定
- ② 開始ポイント条件としてデータ・メモリ・アクセス系のイベント条件レジスタを設定
- ③ 終了ポイント条件としてプログラム・フェッチ系のイベント条件レジスタを設定
- ④ 別系列のイベント条件レジスタを設定した時点でワーニング・メッセージを表示
この場合、設定はされますが、本来の意味を果たしません。

8.4.46 オブジェクト・ベリファイ・コマンド (VRY)

入力形式

形式1 VRY Δ [d:] file

brk:n>	○	emu:n>	×	trc:n>	×
--------	---	--------	---	--------	---

【機能】

VRYコマンドは、オブジェクト・ファイルとメモリ内容を比較します。比較した結果がファイルとメモリで異なる場合は、そのアドレスとデータが表示されます。

【指定方法】

オペランド

[d:]file: オブジェクト・コードが格納されているファイルの名前を指定します。
ドライブ番号は省略することができます。

【使用例】

(a) 比較結果が正常であった場合

brk:0>VRY SAMPLE.HEX^①object verify complete^②brk:0>■^③

- ① SAMPLE.HEXというファイルとメモリ内容を比較
- ② 正常終了した場合のメッセージ
- ③ コマンド入力待ち

(b) 比較結果に異常があった場合

```
brk : 0> VRY SAMPLE. HEX  ①
object verify ②
Address File Memory
0123 00 01
1234 FF FE
brk : 0> ■
```

- ① SAMPLE. HEXというファイルとメモリ内容を比較
- ② 比較した結果に異常があった場合のメッセージ

(c) 指定したファイルが存在しない場合

```
brk : 0> VRY SAMPLE. HEX  ①
object SAMPLE. HEX not found
brk : 0> ■
```

- ① SAMPLE. HEXというファイル名が見つからない場合のメッセージ

第9章 その他の機能

9.1 IE-75001-RのRS-232-Cインタフェース機能概要

この節では、IE-75001-RのRS-232-Cインタフェース（チャンネル1，チャンネル2）について説明します。特にこの節を読まなくても、第4章 システム構成に従って、ホスト・マシンあるいはPROMプログラマと接続すれば正しく動作します。チャンネル1，チャンネル2の機能や仕様を詳細に知りたい場合はこの節をお読みください。

9.1.1 RS-232-Cインタフェースの信号線

ここでは、ターミナル・モードの機器を「ターミナル」、モデム・モードの機器を「モデム」と省略します。

IE-75001-Rで使用している信号線を表9-1にまとめます。

表9-1 RS-232-Cインタフェースの信号線

No.	系	信号名	機能	方向		ピン番号
				モデム	ターミナル	
1	グラウンド	Frame Ground	FG	保安用グラウンド		1
		Signal Ground	SG	信号用グラウンド		7
2	データ	Transmitted Data	TxD	ターミナルからモデムへの送信データ線	←	2
		Received Data	RxD	モデムからターミナルへの送信データ線	→	3
3	スタティック・ハンドシェーク	Data Set Ready	DSR	モデム活動状態線	→	6
		Data Terminal Ready	DTR	ターミナル活動状態線	←	20
4	ダイナミック・ハンドシェーク	Request To Send	RTS	ターミナルからモデムへのデータ送信許可信号線	←	4注
		Clear To Send	CTS	モデムからターミナルへの送信許可信号線	→	5

注 詳細については、9.1.3 RTSの設定を参照してください。

表9-2 ピン番号とRTS

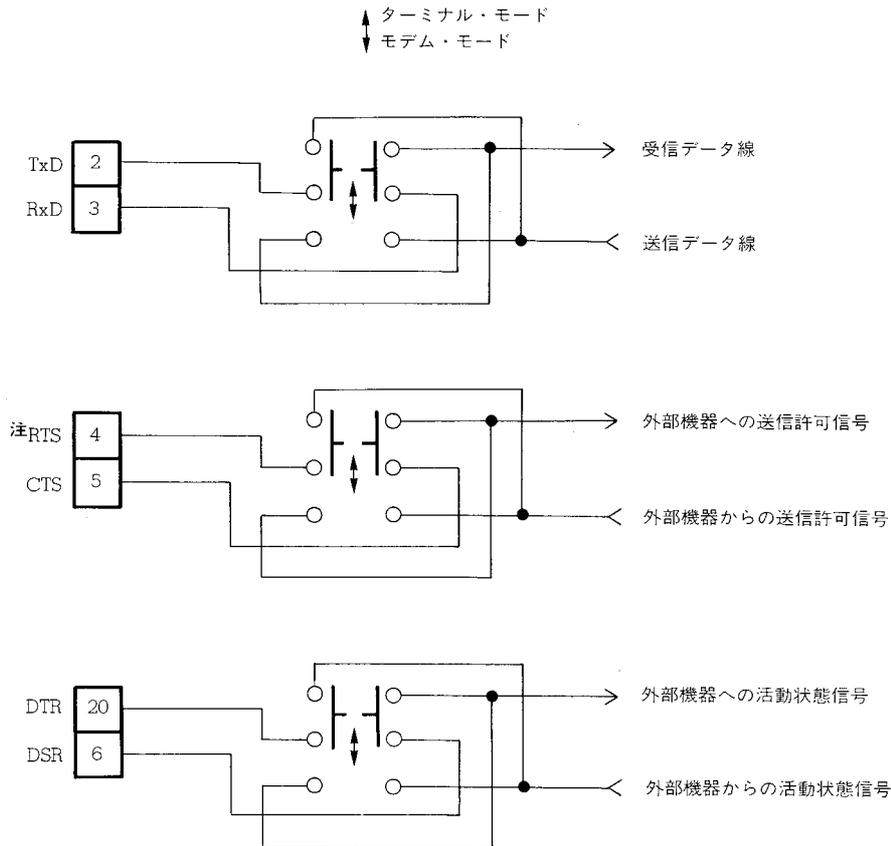
ピン番号	信号名	接続する機器
4	RTS N	ホスト・マシン, PROMプログラマ

9.1.2 ターミナル・モードとモデム・モード

ターミナル・モードとモデム・モードの切り替えは、データ系、スタティック・ハンドシェイク系、ダイナミック・ハンドシェイク系の各信号の向きを変えることにより行われます。

以下に、スライド・スイッチによりどのように切り替わるのか具体的な回路図を示します。

図 9-1 ターミナル/モデム・モード切り替え回路図



注 9.1.3 RTSの設定を参照してください。

注意 RS-232-Cインタフェースでは、必ず“ターミナル・モードの機器”と“モデム・モードの機器”を接続してください。

“ターミナル・モードの機器”同士あるいは、“モデム・モードの機器”同士は、絶対に接続しないでください。

もし、そのような接続をしますと、出力信号同士が衝突して、どちらかのインタフェース・ドライバが破壊されることがあります。

9.1.3 RTSの設定

RTSは、RS-232-Cインタフェースでは4番ピンになっていますが、装置によっては4番ピン以外にRTSと同じ機能を持つ信号線があります。

このような装置とハードウェア・ハンドシェイクをとるために、次のようにRTSを設定することができます。

表 9-3 RTSの設定

選択する RTS名	RTS, FGの設定				接続する機種
	1	2	3	4注2	
RTS N注1	ON	OFF	OFF	OFF	ホスト・マシン, PROMプログラマ

注1. 通常はこの設定にしておきます。

2. DIPスイッチの4番スイッチは、SG（信号グラウンド）とFG（フレーム・グラウンド）を共通にする（ON）か、オープンにする（OFF）かを設定します。通常はオープン（OFF）にしておきます。

9.1.4 ソフトウェア・ハンドシェークとハードウェア・ハンドシェーク

(1) ソフトウェア・ハンドシェーク（フロー制御）

(a) ソフトウェア・ハンドシェークを行うための接続

次の図のように正しく接続を行ってください。

表 9-4 ソフトウェア・ハンドシェークの接続

ピン 番号	系	モデム側	方向	ターミナル側
		信号名		信号名
1	グラウンド	FG	—	FG
7		SG	—	SG
2	データ	TxD	←	TxD
3		RxD	→	RxD

(b) ソフトウェア・ハンドシェークの方法

受信データを蓄えておくバッファ（256バイト）を持っています。このバッファの1/2まで受信データが蓄えられると、データ転送の中断を要求するために、Ctrl-Sを送信先へ発行します。そして、バッファ内のデータが1/3まで減るとデータ転送の再開を要求するために、Ctrl-Qを送信先へ発行します。

ただし、Ctrl-Sを発行して実際にデータ転送が中断されるまでにバッファの容量を越える受信データが転送されますとデータの取りこぼしが生じる可能性があります。

(2) ハードウェア・ハンドシェーク (CHAR: 1キャラクタ・ハンドシェーク)

(a) ハードウェア・ハンドシェークを行うための接続

次表に示すように信号を接続してください。信号が正しく接続されないとデータ転送が正常に行われません。

表 9-5 ハードウェア・ハンドシェークの接続

ピン 番号	系	モデム側	方向	ターミナル側
		信号名		信号名
1	グラウンド	FG	—	FG
7		SG	—	SG
2	データ	TxD	←	TxD
3		RxD	→	RxD
6	スタティック・	DSR	→	DSR
20	ハンドシェーク	DTR	←	DTR
4	ダイナミック・	RTS	←	RTS
5	ハンドシェーク	CTS	→	CTS

(b) ハードウェア・ハンドシェークの方法

ソフトウェア・ハンドシェークと異なり、バッファを使用しません。(a)の信号でデータの送受信をすべて制御します。そのため、信号が正しく接続されていないと正常なハードウェア・ハンドシェークが行われません。

(i) データ送信時

RTS, DTRがともにアクティブであれば、RxDにデータの送信を行います。

(ii) データ受信時

DSRを常にアクティブにしておきます。データ受信OKならばCTSをアクティブに、OKでなければインアクティブにします。このように1バイトずつデータの転送を制御します。

(3) チャネル1のハンドシェーク方式 (ハードウェア/ソフトウェア兼用)

チャネル1では、ハードウェア・ハンドシェークとソフトウェア・ハンドシェークの両方を利用してハンドシェークを行います。

基本的にはソフトウェア・ハンドシェークでデータの送受信を行い、ハードウェア・ハンドシェークでデータの取りこぼしがないように調整をします。

(4) チャンネル2のハンドシェーク方式

チャンネル2ではハードウェア・ハンドシェークとソフトウェア・ハンドシェークのどちらか注をコマンドで設定します。

各ハンドシェークを行う前に以下の点に注意してください。

(a) ハードウェア・ハンドシェークを選択した場合

ハンドシェーク信号（DSR, DTR, RTS, CTS）を接続せずにハンドシェークを行うと正常に行われません。ハードウェア・ハンドシェークを選択した場合は、必ずハンドシェーク信号を接続してください。

(b) ソフトウェア・ハンドシェークを選択した場合

受信データがシリアル・データを蓄えておくバッファの容量を越えてしまった場合、データの取りこぼしが生じる可能性があります（本項(1)(b)ソフトウェア・ハンドシェークの方法を参照してください）。

注 8.4.23 チャンネル2・モード設定コマンド（MOD）を参照してください。

9.1.5 チャンネル1とチャンネル2の機能

チャンネル1とチャンネル2の機能は次表のとおりです。

表 9-6 チャンネル1とチャンネル2の機能

項 目	チャンネル1	チャンネル2	
モード切り替え	ターミナル/モデム・モード (スイッチ切り替え)	←	
ボー・レート	300, 600, 1200, 2400, 4800, 9600, 19200 bps (スイッチ切り替え)	← 注 (ソフトウェア切り替え)	
ハンドシェーク方式	ハードウェア (1キャラクタ), ソフトウェア (フロー制御) 兼用	ハードウェア (1キャラクタ) ま たはソフトウェア (フロー制御) 注 (ソフトウェア切り替え)	
キ ャ ラ ク タ 仕 様	キャラクタ長	8ビット ただし最上位ビット (MSB) は 出力時0, 入力時無視	7ビットまたは8ビット (ソフト ウェア切り替え) 注 ただし8ビット指定時に最上位ビ ット (MSB) は, 出力時0で入力 時無視
	パリティ・ビット	なし	偶数パリティ/奇数パリティ/ パリティなし 注 (ソフトウェア切り替え)
	ストップ・ビット長	2ビット	1ビットまたは2ビット 注 (ソフトウェア切り替え)

注 ソフトウェア切り替えは8.4.23 チャンネル2・モード設定コマンド (MOD) "MOD" コマンドを参照してください。

9.2 IE-75001-Rの平行・インタフェースの機能概要

この節では、IE-75001-Rの平行・インタフェース（チャンネル3とチャンネル4）について述べます。

IE-75001-Rには、8ビット・平行・インタフェースがあり、入力データおよびインタフェース・コントロール信号は、すべてTTLレベルとなっています。また、インタフェース回路はセントロニクス準拠となっています。

チャンネル3、チャンネル4の機能を詳細に知りたい場合は、この節をお読みください。

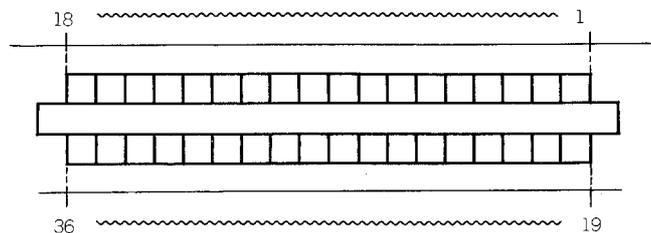
9.2.1 パラレル・インタフェースの信号線

パラレル・インタフェースの入力信号と端子機能を次に示します。

表 9-7 パラレル・インタフェースの信号表

ピン 番号	信号名	方 向		機 能
		チャンネル3	チャンネル4	
1	$\overline{\text{STB}}$	出力	入力	データを読み込むためのストロブ・パルス
2	DATA 0	出力	入力	パラレル・データ 0
3	DATA 1	出力	入力	パラレル・データ 1
4	DATA 2	出力	入力	パラレル・データ 2
5	DATA 3	出力	入力	パラレル・データ 3
6	DATA 4	出力	入力	パラレル・データ 4
7	DATA 5	出力	入力	パラレル・データ 5
8	DATA 6	出力	入力	パラレル・データ 6
9	DATA 7	出力	入力	パラレル・データ 7
10	$\overline{\text{ACK}}$	入力	出力	データの入力完了時に出力される
11	BUSY	入力	出力	データの受け取り不可の信号
19~30 33	GND	—	—	信号グランド
12	PE	—	入力	未使用 (3.3 k Ω の抵抗を介して+5 Vにプルアップ)
32	$\overline{\text{ERROR}}$	—	入力	未使用 (3.3 k Ω の抵抗を介して+5 Vにプルアップ)

図 9-2 パラレル・インタフェースのピン配置



9.2.2 チャンネル3, チャンネル4の機能 (高速ダウン・ロード・モード)

IE-75001-Rの起動時に、高速ダウン・ロード・モード (チャンネル3, チャンネル4) を選択すると、次のファイルを、ホスト・マシンからパラレル・インタフェースを通して高速ダウン・ロードの実行がで
きます。

オブジェクト・ファイル

シンボル・ファイル

ディバグ環境ファイル

ロード・コマンド以外を使用してダウン・ロードを行う場合は、すべてチャンネル4からチャンネル3へス
ルーされます。たとえばMS-DOSのPRINTコマンドを使ってリスト出力する場合、プリンタを再度PC-
9800に接続しなくても、チャンネル3に接続されたプリンタより出力することができます。

また、ロード・コマンド実行時に、他のコマンドによって、パラレル・インタフェース (リスト出力
等) が使用されている場合は、

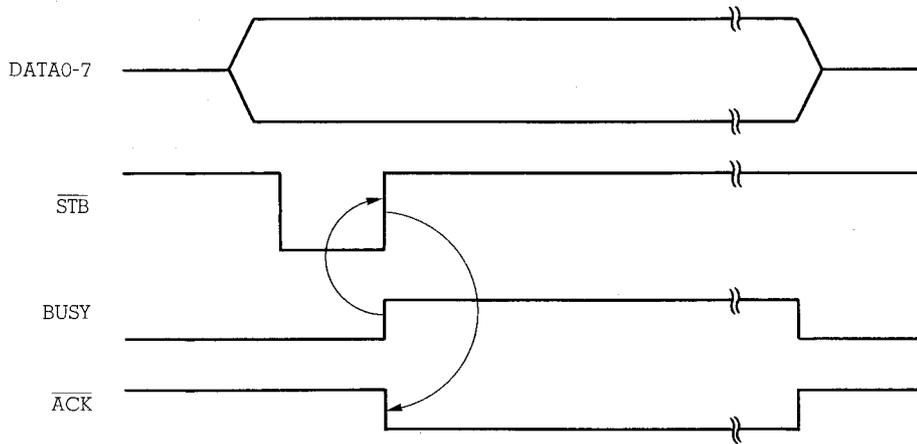
Select Serial Interface

が表示され、シリアル・インタフェースのチャンネル1より、ダウン・ロードが実行されます。

9.2.4 高速ダウン・ロード・モードのタイミング

高速ダウン・ロード・モードのタイミングを次の図で示します。

図9-4 高速ダウン・ロード・モードのタイミング



付録A コマンド一覧表

コマンドの種類	コマンド 本 体	サブコマンド	オ ペ ラ ン ド	動 作 状 態			
				brk :	emu :	trc :	
アセンブル	ASM	なし	[addr]	addr アセンブルの開始アドレス	○	×	×
バンク設定	BNK	なし	[n] $n : 0 \leq n \leq 0FH$	n メモリ・バンク選択レジスタ値	○	○	×
データ・メモリ・ イベント条件設定	BRA	$\left. \begin{array}{l} 1 \\ 2 \\ 3 \\ 4 \end{array} \right\}$	$[MA = \left. \begin{array}{l} \text{addr} \\ \text{partition} \end{array} \right\}] [V = \text{data1}] [C = \text{status}] [E = \text{data2}]$ <small>注1</small> <small>注2</small>	<div style="border: 1px dashed black; padding: 5px;"> addr データ・メモリ・アドレス partition データ・メモリ・アドレス範囲 data1 データ値 status アクセス条件 data2 外部センス・クリップからの入力データ値 </div>	○	○	×
			<small>注1.</small> BRA1, BRA2 アドレス, マスク指定およびパーティション指定ができます。 BRA3, BRA4 アドレス, マスクのみ指定ができます。 <small>2.</small> $\left. \begin{array}{l} R4 \\ W4 \\ RW4 \\ R8 \\ W8 \\ RW8 \\ SP \end{array} \right\}$				
イベント情報表示	BRK	なし	なし	注意 本コマンドは、トレース・モードでは使用できません。	○	○	×

コマンドの種類	コマンド 本 体	サブコマンド	オ ペ ラ ン ド		動 作 状 態		
					brk :	emu :	trc :
コマンド・ファイル 作成	COM	なし	[{ [d:]file } [LST : CON :]]	[d:]file コマンド・ファイルとしてファイルをオープンします LST: コマンド・ファイルとしてリスト装置をオープンします CON: コマンド・ファイルをクローズします	○	○	○
カバレッジ測定結果 の表示	CVD	D	[partition]	partition 表示範囲	○	×	×
カバレッジ測定結果 の削除	CVD	K	なし		○	×	×
カバレッジ測定範囲 の追加	CVM	A	[partition]	partition 追加範囲	○	×	×
カバレッジ測定範囲 の表示	CVM	D	なし		○	×	×
カバレッジ測定範囲 の指定解除	CVM	K	[partition]	partition 指定解除範囲	○	×	×
逆アセンブル	DAS	なし	[{ addr } [partition]]	addr 逆アセンブルの開始アドレス partition 逆アセンブルのアドレス範囲	○	×	×

コマンドの種類	コマンド 本 体	サブコマンド	オ ペ ラ ン ド	動 作 状 態		
				brk :	emu :	trc :
ディレクトリ表示	DIR	なし	[[d:]file]	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
イベント検出点設定	DLY	なし	<div style="border: 1px dashed black; padding: 5px;"> <p>F 指定条件成立後、トレース・フルまでトレーサ停止を遅らせる M 指定条件成立後、トレーサの全フレームの半分がデータで埋まるまでトレーサを遅らせる L 指定条件成立後、即、トレーサを停止する</p> <p> { F M L } </p> <p> TRIGGER POINT F M L TRACE MEMORY -----*-> </p> </div>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
DOS	DOS	なし	なし	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
IE-75000-Rコントロ ール・プログラム	EXT	なし	なし	<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
コマンド・ヒストリ 表示	HIS	なし	なし	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
ヘルプ	HLP	なし	[command]	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

コマンドの種類	コマンド 本 体	サブコマンド	オ ペ ラ ン ド	動 作 状 態			
				brk :	emu :	trc :	
オブジェクト/シンボル/ディバグ環境のロード	LOD	なし	$[[d:]file[\overset{\text{注}}{\text{module}\$...}][\left\{\begin{matrix} C \\ D \\ S \end{matrix}\right\}]]$ <div style="border: 1px dashed black; padding: 5px; margin-top: 10px;"> [d:]file ファイル名 module シンボル・ファイルのモジュール名 C オブジェクト指定 D ディバグ環境指定 S シンボル・ファイル指定 </div> <p style="text-align: center; margin-top: 10px;">注 IBM PCシリーズをホストとした場合\となります。</p>	○	×	×	
出力デバイス・リダイレクト	LST	なし	$\left[\begin{matrix} [d:]file \\ LST: \\ CON: \end{matrix} \right]$ <div style="border: 1px dashed black; padding: 5px; margin-top: 10px;"> [d:]file ファイルを出力デバイスとしてオープンする LST: リスト装置出力デバイスとしてオープンする CON: オープンされている出力デバイスをクローズする </div>	○	○	○	
演算	MAT	なし	expression expression 式	○	○	○	
プログラム・メモリの操作	メモリ内容の変更	MEM	C	[addr] addr メモリの変更開始アドレス	○	×	×
	メモリ内容表示	MEM	D	$\left[\begin{matrix} addr \\ partition \end{matrix} \right]$ <div style="border: 1px dashed black; padding: 5px; margin-top: 10px;"> addr メモリの表示開始アドレス partition メモリの表示開始/終了アドレス </div>	○	×	×
	メモリのテスト	MEM	E	[partition] partition テストするアドレスの範囲	○	×	×
	メモリのイニシャライズ	MEM	F	partition data-string partition メモリのイニシャライズ範囲 data-string イニシャライズ・データ	○	×	×

コマンドの種類		コマンド 本 体	サブコマンド	オ ペ ラ ン ド	動 作 状 態			
					brk :	emu :	trc :	
プログラム・メモリの操作	メモリ内容のサーチ	MEM	G	partition data-string	<div style="border: 1px dashed black; padding: 2px;"> partition メモリのサーチ範囲 data-string サーチ・データ </div>	○	×	×
	メモリ内容のコピー	MEM	M	partition addr	<div style="border: 1px dashed black; padding: 2px;"> partition コピー元範囲 addr コピー先アドレス </div>	○	×	×
	メモリ内容の交換	MEM	X	partition addr	<div style="border: 1px dashed black; padding: 2px;"> partition メモリの交換範囲 addr 交換先アドレス </div>	○	×	×
	メモリ内容の比較	MEM	V	partition addr	<div style="border: 1px dashed black; padding: 2px;"> partition メモリの比較範囲 addr 比較先アドレス </div>	○	×	×
チャンネル2モード設定	MOD	なし	$[\text{MODE} = \left\{ \begin{array}{l} \text{CHAR} \\ \text{FLOW} \end{array} \right\}] [\text{BAUD} = \left\{ \begin{array}{l} 19200 \\ 9600 \\ 2400 \\ 1200 \\ 600 \\ 300 \end{array} \right\}] [\text{LONG} = \left\{ \begin{array}{l} 7 \\ 8 \end{array} \right\}] [\text{PAR} = \left\{ \begin{array}{l} \text{OFF} \\ \text{EVEN} \\ \text{ODD} \end{array} \right\}] [\text{STOP} = \left\{ \begin{array}{l} 1 \\ 2 \end{array} \right\}]$	<div style="display: flex; justify-content: space-around; text-align: center;"> <div style="border: 1px dashed black; padding: 2px;">ハンドシェーク・モード</div> <div style="border: 1px dashed black; padding: 2px;">ボ-レート</div> <div style="border: 1px dashed black; padding: 2px;">キャラクタ長</div> <div style="border: 1px dashed black; padding: 2px;">パリティ・ビット</div> <div style="border: 1px dashed black; padding: 2px;">ストップ・ビット</div> </div>	○	×	×	
外部センス・クリップ・モード	OUT	なし	$\left[\left\{ \begin{array}{l} \text{OFF} \\ \text{ON } \$T \\ \text{ON BRA?} \end{array} \right\} \right]$	<div style="border: 1px dashed black; padding: 5px;"> OFF 外部信号トレース (8本) と外部信号レベル検出を行う ON \$T 外部信号トレース (7本) と外部イベント出力 (第0ビット) を行う ON BRA? BRA3, BRA4が示すD-memデータ (8ビット) を外部出力し, これのトレースも行う </div>	○	○	×	

コマンドの種類	コマンド 本 体	サブコマンド	オ ペ ラ ン ド	動 作 状 態						
				brk :	emu :	trc :				
バス・カウント設定	PAS	なし	[count]	count バス回数			○	○	×	
端末モード設定	PGM	なし	[C]	C 制御キャラクタの変更			○	×	×	
データ・メモリの操作	データ・メモリ 内容の変更	RAM	C	$\left[\left\{ \begin{array}{l} \text{addr} [\left\{ \begin{array}{l} \$B \\ \$N \end{array} \right\}] \right\} \right]$ addrb	addr 変更開始アドレス addrb 変更アドレス \$B バイト単位 \$N ニブル単位			○	×	×
	データ・メモリ 内容の表示	RAM	D	$\left[\left\{ \begin{array}{l} \text{addr} \\ \text{partition} \end{array} \right\} \left[\left\{ \begin{array}{l} \$B \\ \$N \end{array} \right\} \right] \right]$ addrb	addr 表示開始アドレス partition 表示するデータ・メモリ範囲 addrb 表示アドレス \$B バイト単位 \$N ニブル単位			○	×	×
	データ・メモリ 内容のテスト	RAM	E	[partition]	partition テストするデータ・メモリ範囲			○	×	×
	データ・メモリ 内容のイニシャ ライズ	RAM	F	partition data-string $\left[\left\{ \begin{array}{l} \$B \\ \$N \end{array} \right\} \right]$	partition データ・メモリのイニシャライズ範囲 data-string イニシャライズ・データ			○	×	×
	データ・メモリ 内容のサーチ	RAM	G	partition data-string $\left[\left\{ \begin{array}{l} \$B \\ \$N \end{array} \right\} \right]$	partition データ・メモリのサーチ範囲 data-string サーチ・データ			○	×	×

コマンドの種類		コマンド 本 体	サブコマンド	オ ペ ラ ン ド	動 作 状 態			
					brk :	emu :	trc :	
データ・メモリの操作	データ・メモリ 内容のコピー	RAM	M	partition addr [{ \$B } { \$N }]	partition コピー元範囲 addr コピー先アドレス	○	×	×
	データ・メモリ 内容の交換	RAM	X	partition addr [{ \$B } { \$N }]	partition メモリの交換範囲 addr 交換先アドレス	○	×	×
	データ・メモリ 内容の比較	RAM	V	partition addr [{ \$B } { \$N }]	partition データ・メモリの比較範囲 addr 比較先アドレス	○	×	×
	データ・メモリ のロード	RAM	L	なし		○	×	×
	データ・メモリ のセーブ	RAM	S	[partition][, partition]・・・[, partition] 最大5個	partition セーブするデータ・メモリの範囲	○	×	×
レジスタ操作	レジスタの変更	REG	C	[register]	register レジスタ名	○	×	×
	レジスタの表示	REG	D	[{ ALL } { register }]	ALL 全レジスタ・バンクの全レジスタ指定 register レジスタ名	○	×	×
リセット		RES	なし	[H]	H IE-75001-Rのリセット	○	○	○

コマンドの種類	コマンド 本 体	サブコマンド	オ ペ ラ ン ド	動 作 状 態				
				brk :	emu :	trc :		
エミュレーション実行	ブレーク条件なし リアルタイム実行	RUN	N	[addr]	addr 実行開始アドレス	○	×	×
	ブレーク条件付き リアルタイム実行	RUN	B	[addr]	addr 実行開始アドレス	○	×	×
	トレース実行	RUN	T	[addr][^{注1} {, expression } {, word }][TRD][REG]	addr 実行スタート・アドレス word トレース実行ステップ数 TRD トレース表示指定 REG レジスタ表示指定	○	×	×

注1. register { = } { マスク・データ }
 { > } { ビット・データ }
 { < } { 4ビット・データ }
 { => } { 8ビット・データ }
 { >= } { 16ビット・データ }
 { =< }
 { <= }
 { >< }
 { <> }

注2. マスク・データの指定の場合 “=”, “><”, “<>” 以外の条件は、使用できません。

コマンドの種類		コマンド 本 体	サブコマンド	オ ペ ラ ン ド	動 作 状 態			
					brk :	emu :	trc :	
セーブ		SAV	なし	<p style="text-align: center;">注</p> <p>[d:]file [partition] [{ C } { D }]</p> <p style="text-align: center;">注 5つまで指定可能</p>	[d:]file ファイル名 partition プログラム・メモリ領域 C オブジェクトのみ D デバッグ環境のみ	○	×	×
エミュレーション・ デバイス・モード 切り替え		SET	なし	<p>[{ STACK } { OFF } { SLWAIT } { ON }]</p> <p>STACK 対象デバイスの割り込み時のスタック・バイト数の指定 OFF:2バイト ON:3バイト SLWAIT 対象デバイスのRESET後のウェイト時間の指定 OFF:31.25 ms ON:7.81 ms</p>	STACK 対象デバイスの割り込み時のスタック・バイト数の指定 OFF:2バイト ON:3バイト SLWAIT 対象デバイスのRESET後のウェイト時間の指定 OFF:31.25 ms ON:7.81 ms	○	×	×
特殊レジスタ操作	特殊レジスタの 変更	SPR	C	<p>[{ group } { register }]</p>	group グループ名 register レジスタ名	○	×	×
	特殊レジスタの 表示	SPR	D	<p>[{ group } { register }]</p>	group グループ名 register レジスタ名	○	×	×
入力デバイス・リダイレクト		STR	なし	<p>[d:]file [parameter list]</p>	[d:]file ファイル名 parameter list 実パラメータ・リスト	○	×	○
リアルタイム・ エミュレーション 中止		STP	なし			×	○	○
				[T]	T トレーサの停止	×	×	○

コマンドの種類		コマンド 本 体	サブコマンド	オ ペ ラ ン ド	動 作 状 態																										
					brk :	emu :	trc :																								
対象デバイス 選択	対象デバイスの 選択	STS	C	なし	○	×	×																								
	対象デバイスの 表示	STS	D	なし	○	×	×																								
シン ボ ル 操 作	アペンド・シン ボルの定義	SYM	A	symbol word <table style="display: inline-table; vertical-align: middle;"> <tr><td>{</td><td>N</td><td>}</td></tr> <tr><td></td><td>C</td><td></td></tr> <tr><td></td><td>D</td><td></td></tr> <tr><td></td><td>B</td><td>}</td></tr> </table> <div style="border: 1px dashed black; padding: 5px; margin-top: 10px;"> <table> <tr><td>symbol</td><td>シンボル名</td></tr> <tr><td>word</td><td>シンボル値</td></tr> <tr><td>N</td><td>Number属性のアペンド・シンボルを定義</td></tr> <tr><td>C</td><td>Code //</td></tr> <tr><td>D</td><td>Date //</td></tr> <tr><td>B</td><td>Bit //</td></tr> </table> </div>	{	N	}		C			D			B	}	symbol	シンボル名	word	シンボル値	N	Number属性のアペンド・シンボルを定義	C	Code //	D	Date //	B	Bit //	○	×	×
	{	N	}																												
		C																													
		D																													
	B	}																													
symbol	シンボル名																														
word	シンボル値																														
N	Number属性のアペンド・シンボルを定義																														
C	Code //																														
D	Date //																														
B	Bit //																														
アペンド・シン ボルのシンボル 値の変更	SYM	C	symbol word	<div style="border: 1px dashed black; padding: 5px; margin-top: 10px;"> <table> <tr><td>symbol</td><td>シンボル名</td></tr> <tr><td>word</td><td>シンボル値</td></tr> </table> </div>	symbol	シンボル名	word	シンボル値	○	×	×																				
symbol	シンボル名																														
word	シンボル値																														
シンボル (アペ ンド/パブリッ ク/モジュール) の表示	SYM	D	注 [module name¥] 注 IBM PCシリーズをホストとした場合\となります。	<div style="border: 1px dashed black; padding: 5px; margin-top: 10px;"> module name¥ モジュール名 </div>	○	×	×																								
アペンド・シン ボルの削除	SYM	E	symbol	<div style="border: 1px dashed black; padding: 5px; margin-top: 10px;"> symbol シンボル名 </div>	○	×	×																								

コマンドの種類		コマンド 本 体	サブコマンド	オ ペ ラ ン ド	動 作 状 態		
					brk :	emu :	trc :
シ ン ボ ル 操 作	全シンボルの削 除	SYM	K	なし	○	×	×
	アペンド・シン ボルのロード	SYM	L	なし	○	×	×
	アペンド・シン ボルのセーブ	SYM	S	なし	○	×	×
	カレント・モジ ュール名の指定	SYM	M	なし	○	×	×
システム再起動		SYS	なし	なし	○	×	×
トレース表示		TRD	なし	$[ALL] \left\{ \begin{array}{l} \$C \\ \$F \\ \$Q \end{array} \right\} [\text{option } [, \text{option}] [, \text{option}] \cdots [, \text{option}] [\Delta \$B]]$ <div style="border: 1px dashed black; padding: 5px; margin: 10px 0;"> <p>注</p> <p>ALL 全トレース・データを表示して、コマンドを終了する。</p> <p>\$C チェックポイント、およびトリガ・フレームを検索し、表示する。</p> <p>\$F "TRF" で指定した検索条件、およびトリガ・フレームを検索し、前後5行を表示する。</p> <p>\$Q "TRF" で指定した検索条件、およびトリガ・フレームを検索し、表示する。</p> </div> <p>注 "P0", "P1", ... "P15" I/Oポート・トレース・データの表示</p> <p>"EXT" 外部センス・クリップ・データの表示</p> <p>"DMEM" D-memトレース・データ (MA, MD, MRW) の表示</p>	○	○	×

コマンドの種類	コマンド 本 体	サブコマンド	オペランド	動作状態					
				brk :	emu :	trc :			
トレース・データ検索条件設定	TRF	なし	$[PA = \left\{ \begin{array}{l} \text{addrX} \\ \text{partition1} \end{array} \right\}][PD = \text{data1}][MA = \left\{ \begin{array}{l} \text{addrX} \\ \text{partition2} \end{array} \right\}][MD = \text{data2}][MRW = \text{status}][Pn = \text{data3}][EXT = \text{data4}]$ <div style="border: 1px dashed black; padding: 5px; margin-top: 10px;"> <p>addrX P-memアドレス検索データ</p> <p>partition1 P-memアドレス検索データ範囲</p> <p>data1 P-memデータ検索データ</p> <p>addrX D-memアドレス検索データ</p> <p>partition2 D-memアドレス検索データ範囲</p> <p>data2 D-memデータ</p> <p>status D-memアクセス・ステータス検索データ</p> <p>"MRD" : メモリ・リード</p> <p>"MWR" : メモリ・ライト</p> <p>"MRW" : メモリ・リード・モディファイ・ライト</p> <p>data3 I/Oポート検索データ</p> <p>data4 外部センス・クリップ検索データ</p> </div>	○	○	×			
トレーサ起動	TRG	なし	なし	×	○	×			
トレース・モード指定	TRM	なし	$\left[\left\{ \begin{array}{l} \text{ALL} \\ \text{TRX} \\ \text{TRY} \end{array} \right\} \right] \left[\left\{ \begin{array}{l} \$M \\ \$E \end{array} \right\} \right]$ <div style="border: 1px dashed black; padding: 5px; margin-top: 10px;"> <p>"ALL" 無条件トレースを指定する</p> <p>"TRX" クオリファイ・トレースを指定する</p> <p>"TRY" セクション・トレースを指定する</p> <p>"\$M" マシン・サイクル・トレースを指定する</p> <p>"\$E" イベント・サイクル・トレースを指定する</p> </div>	○	○	×			
トレース・ポインタ操作	TRP	なし	$\left[\left\{ \begin{array}{l} \text{word} \\ \text{F} \\ \text{L} \\ \text{T} \end{array} \right\} \right]$ <div style="border: 1px dashed black; padding: 5px; margin-top: 10px;"> <p>word ポインタの移動数</p> <p>F ポインタを先頭に置く</p> <p>L ポインタを最後に置く</p> <p>T ポインタをトリガ・フレームに置く</p> </div>	○	○	×			
クオリファイ・トレース条件指定	TRX	なし	[BR?][BR?][BR?][BR?][BR?][BR?]	BR? イベント条件レジスタ名指定			○	○	×

コマンドの種類	コマンド 本 体	サブコマンド	オ ペ ラ ン ド	動 作 状 態		
				brk :	emu :	trc :
セクション・トレー ス条件指定	TRY	{ D E }	[BR?][BR?][BR?][BR?][BR?][BR?] <div style="border: 1px dashed black; padding: 2px; display: inline-block; margin-left: 200px;">BR? イベント条件レジスタ名指定</div>	○	○	×
オブジェクト・ベリ ファイ	VRY	なし	[d:]file <div style="border: 1px dashed black; padding: 2px; display: inline-block; margin-left: 200px;">[d:]file ファイル名</div>	○	×	×

付録B エラー・メッセージ一覧表

No.	エラー・メッセージ, 内容説明	
1	メッセージ	Unrecognized command
	説明	コマンド・キー・ワードが正しくない。
2	メッセージ	Command format error
	説明	コマンド・キー・ワードは正しいが、オペランドが正しくない。
3	メッセージ	Command/Data too long
	説明	128文字以上のコマンド, またはデータが入力された。
4	メッセージ	Input data error
	説明	入力したデータが正しくない。
5	メッセージ	System mode command
	説明	スタンドアローン・モードでシステム・モードのコマンドを入力した。
6	メッセージ	Check sum error
	説明	オブジェクトのロード/セーブ時にチェック・サム・エラーを検出した。
7	メッセージ	Bad character
	説明	オブジェクトのロード/セーブ時に正しくない文字を検出した。
8	メッセージ	aborted
	説明	オブジェクトのロード/セーブ中に中断キーが入力された。
9	メッセージ	Warning double define
	説明	LODコマンドで, 同一モジュール名が複数回指定された。
10	メッセージ	Bad file entry
	説明	ファイル名の記述が正しくない。
11	メッセージ	File overflow
	説明	LODコマンドで, 入力可能なシンボル・ファイル数をオーバした。
12	メッセージ	Illegal record
	説明	LODコマンドで, シンボル・テーブル・ファイルのレコード形式が正しくない。
13	メッセージ	load failed
	説明	LODコマンドで, シンボル, あるいはオブジェクトのコード中にエラーを検出した。
14	メッセージ	module overflow
	説明	LODコマンドで, 入力できるモジュール数をオーバした。
15	メッセージ	Not loaded symbol
	説明	シンボルがロードされていない。

No.	エラー・メッセージ, 内容説明	
16	メッセージ	Not found module record
	説明	LODコマンドで, 指定されたモジュール名がシンボル・テーブル・ファイルに存在しない。
17	メッセージ	file not found
	説明	指定されたファイル名が存在しない。
18	メッセージ	no symbol of append
	説明	SYM_D, SYM_Sコマンドで, アペンド・シンボルは存在しない。
19	メッセージ	Can not execute HLP command !
	説明	カレント・ディレクトリ上にヘルプ・ファイル (IE75000. HLP), ヘルプ・オーバーレイ・ファイル (IE75000. OV2) が存在しない。
20	メッセージ	No. HLP file on the default drive
	説明	HLPコマンド実行時, カレント・ディレクトリ上にヘルプ・ファイル (IE75000. HLP), ヘルプ・オーバーレイ・ファイル (IE75000. OV2) が見つからなかった
21	メッセージ	Keyword Error
	説明	HLPコマンドで, コマンド・キー・ワードが正しくない。
22	メッセージ	Can not command abbreviation !
	説明	カレント・ディレクトリ上に省略形のオーバーレイ・ファイル (IE75000. OV1) が存在しない。
23	メッセージ	File already exists.
	説明	ファイルの属性がSYS属性, あるいはR/O属性のファイルに対して同一名のファイルを新たに作成しようとした。
24	メッセージ	Reserved file name
	説明	IE-75000-Rコントロール・プログラムが使う, 予約されたファイル名を指定した。
25	メッセージ	File name is used by other process
	説明	すでにオープン済みのファイル名を指定した。
26	メッセージ	Can not closeファイル名
	説明	表示されたファイルのクローズが正常にできなかった。
27	メッセージ	Disk write error ファイル名
	説明	表示されたファイルの書き込みで異常を見つけた。
28	メッセージ	Disk read error ファイル名
	説明	表示されたファイルの読み込みで異常を見つけた。
29	メッセージ	Can not open ファイル名
	説明	指定されたファイルがオープンできなかった。
30	メッセージ	File make error ファイル名
	説明	表示されたファイルを作成できなかった。

No.	エラー・メッセージ, 内容説明	
31	メッセージ	Can not close ファイル名. Cancel ××× command
	説明	×××のコマンド実行中, 表示されたファイルのクローズが正常にできなかった (×××はSTR, LST, COMの各コマンド)。
32	メッセージ	Disk write error ファイル名. Cancel ××× command
	説明	×××のコマンド実行中, 表示されたファイルの書き込みで異常を見つけた (×××は, LST, COMの各コマンド)。
33	メッセージ	Disk read error ファイル名. Cancel STR command
	説明	STRコマンド実行中, 表示されたファイルの読み込みで異常を見つけた。
34	メッセージ	List device is used by other process
	説明	他の処理がリスト装置を使っている (COMコマンドとLSTコマンドの両方でリスト装置を指定した場合)。
35	メッセージ	Append symbol file not found
	説明	SYM_Lコマンドで, アペンド・シンボル・ファイル (IE75000.SYM) がカレント・ディレクトリ上に存在しなかった。
36	メッセージ	Illegal append symbol file
	説明	SYM_Lコマンドで, アペンド・シンボル・ファイルの形式が正しくない。
37	メッセージ	Communication error
	説明	IE-75001-Rとホスト・マシンの通信が正常にできなかった。
38	メッセージ	Not found memories
	説明	外部メモリが指定されたのにメモリが使用できない。
39	メッセージ	Non map area access !
	説明	ASMコマンド実行中, マッピングされていないメモリにアクセスしようとした。
40	メッセージ	Assemble area over !
	説明	ASMコマンドで, アクセスできるメモリの範囲を越えた。
41	メッセージ	Disassemble area over !
	説明	DASコマンドで, アクセスできるメモリの範囲を越えた。
42	メッセージ	Caution !
	説明	ジェネリックなオブジェクトが生成された。または注意を要する。
43	メッセージ	Error !
	説明	オブジェクト・コードを生成できない。またはあきらかにエラーである。
44	メッセージ	Warning !
	説明	オブジェクトの生成はできるが, 正しい動作は望めない。

No.	エラー・メッセージ, 内容説明	
45	メッセージ	Unexecutable command
	説明	エミュレーション中に実行できないコマンドを入力した, あるいはエミュレーション中にしか実行できないコマンドをブレーク中に入力した。
46	メッセージ	Non trace data
	説明	TRDコマンドで, トレース・データがない。
47	メッセージ	Not found
	説明	TRDコマンドで, 指定したフレームがない。
48	メッセージ	Trigger frame not found
	説明	TRDの促進入力時, またはTRPのトリガ・フレーム指定を入力したがトリガ・フレームがなかった。
★ 49	メッセージ	PORT ××H, ××H DATA ×× ×× E-CPU ERROR !!
	説明	エミュレーション・デバイスが正常に動作しない ^注 。 <ul style="list-style-type: none"> ●ユーザ・クロックを使っている場合 エミュレーション・ボード上のユーザ・クロックの設定がきちんとされているか確認してください。 ●内部クロックを使っている場合 プローブを通して, 過電流, 過電圧がエミュレーション・デバイスにかかっている可能性があります。 負荷原因を取り除いてください。
★ 50	メッセージ	object load aborted
	説明	ハイ・スピード・ダウン・ロードを選択しているにもかかわらず, セントロニクスの接続がされていない。
★ 51	メッセージ	異常レコード aborted
	説明	モジュール名の先頭に0-9の数字を使用した。 モジュール名を変更してください。

注 CLK Uコマンドを実行後, このメッセージが表示された場合は, 十分な振幅, ゲインを持つユーザ・クロックや外部クロックが, エミュレーション・デバイスに供給されていません。クロックの波形などの確認をしてください。また, この操作以外で発生した場合は, 当社または特約店にご連絡ください。

No.	エラー・メッセージ, 内容説明	
52	メッセージ	No connect Abort (Y or N) ;
	説明	ホスト・マシンとIE-75000-R, IE-75001-Rがつかない。 RS-232-Cの設定を確認してください。 RS232. DRVがデバイス・ドライバに組み込んでいるか確認してください (PC-9800の場合)。 筐体にエミュレーション・ボード, ブレーク・ボードが接続されているか確認してください。 ブレーク・ボードとJ1, J2のケーブルが接続されているか確認してください。

★

付録C 使用上の注意

C.1 使用上の注意概要

No.	注意点の概要
1	CHKコマンドで設定した箇所では割り込みがかからない。
2	DLY中はCHKデータのトレースは行われない。
3	ブレークしないでスリップする命令がある。
4	1ステップ・コマンド実行時は、STOP、HALT状態にならない。
5	BRAコマンドで設定したブレーク条件を実行すると1-2命令スリップする。
6	クオリファイ・トレースでは、スキップされる命令のトレース表示が正常にされない。
7	スキップされる命令にブレークを設定した場合、その命令がスキップされて実行されなくてもブレークする。
8	1ステップ実行時には割り込みがかからない。
9	セクション・トレースによるトレース結果で、最終行の表示が実際の実行結果と異なる。
10	デバイスの仕様通りに動作しない命令がある。
11	トレース結果が実際の実行結果と異なる特殊レジスタ、汎用レジスタがある。
12	μ PD75028, 036, P036, 048, P048 ^注 , 064 ^注 , 066 ^注 , 068 ^注 , P068 ^注 を使用する場合、サブ・システムのユーザ・クロックが使用できない。

注 開発中

備考 表のNo.は、次頁以降の詳細説明の番号と対応しています。

C.2 使用上の注意の詳細説明

1. CHKコマンドで設定した箇所では割り込みがかからない。

CHKコマンドで設定した条件と同じ箇所では割り込みが発生した場合、次の命令で割り込みが受け付けられます。また、CHKコマンドで設定した条件が連続している場合、その最終条件の次で割り込みが受け付けられます。

2. DLY中は、CHKデータのトレースは行わない。

トレース機能において、DLY期間中にCHKコマンドで設定した条件が満足されてもトレース・データとして取り込みません。

3. ブレークしないでスリップする命令がある。

次に示す命令にブレーク条件を設定した場合、その命令を実行後、すぐブレークしないで次の命令でブレークします（スリップします）。

また、次に示す命令が連続した場合、最後の命令を実行後、すぐブレークしないで次の命令でブレークします。

SET1	FBX.X	(EI FBX)	} FBX.X, MBXの領域をMOVなどの命令でアクセス したときも同様
CLR1	FBX.X	(DI FBX)	
SEL	MBX		

4. 1ステップ・コマンド実行時は、STOP, HALT状態にはならない。

1ステップ・コマンド実行時は、STOP, HALT命令は実行後直ちに解除されるためSTOP, HALTの各状態を保持しません。

5. BRAコマンドで設定したブレーク条件を実行すると1-2命令スリップする。

(1) BRAコマンドでイベント条件としてメモリのライト条件を設定した場合

次のようにブレークがスリップされます。

- ① ブレーク条件が設定されたところの次の命令が1バイト命令の場合
→ ブレーク条件が設定された所の次の命令を2つ実行してブレークします。

例

```
MOV @HL, A ←ここでブレーク条件成立
MOV XA, BC
MOV DE, XA ←この命令を実行してブレーク
```

- ② ブレーク条件が設定された所の次の命令が2バイトおよび3バイト命令の場合
→ ブレーク条件が設定された所の次の命令を1つ実行してブレークします。

例

```
MOV @HL, XA ←ここでブレーク条件成立
MOV DE, XA ←この命令を実行してブレーク
```

② **BRA**コマンドでイベント条件としてメモリのリード条件を設定した場合

次の命令を実行後、ブレークします（命令のバイト数に関係しません）。

6. **クオリファイ・トレースでは、スキップされる命令のトレース表示が正常にされない。**
クオリファイ・トレースでは、実際はスキップ動作が行われてもスキップしたという表示がされません。
7. **スキップされる命令にブレークを設定した場合、その命令がスキップされて実行されなくてもブレークする。**
ブレーク条件で設定した命令が実際はスキップされ、実行されていないとしてもブレークします。
8. **1ステップ実行には割り込みがかからない。**
1ステップ実行に割り込み要求が発生した場合、要求フラグ (IRQ×××) はセットされますが、割り込みは実行されません（要求フラグはそのまま保持されます）。
9. **セクション・トレースによるトレース結果で最終行の表示が実際の実行結果と異なる。**
BRSをセクション・トレースのディスエーブル条件とした場合、ディスエーブル条件に設定した命令が2バイトおよび3バイト命令であると、そのトレース表示が正常に表示されません。
10. **デバイスの仕様通りに動作しない命令がある。**

次の命令は、仕様と異なる動作をします。

XCH A, @HL	} これらの命令において、次の特殊レジスタのアドレスを指定した場合、不定動作となります（デバイスも同様に不定動作となります）。
XCH A, @HL+	
XCH A, @HL-	
XCH A, @rpal	
XCH XA, @HL	
XCH A, mem	
XCH XA, mem	

11. トレース結果が実際の実行結果と異なる特殊レジスタ、汎用レジスタがある(実行は正常に行われている)。

次の特殊レジスタのリード・データは、実際のデータとトレース・データが異なり、不定値がトレースされます。

- BS (F82H)
- PSW (FB0H)

POP命令でデータを復帰させてからMOV命令等でデータが更新されるまでの間に汎用レジスタの値をリードした場合、実際の汎用レジスタ値とトレース・データの値は異なり、不定値がトレースされます(実際の値はあっています)。

12. μ PD75028, 036, P036, 048, P048, 064, 066, 068, P068を使用する場合、サブシステムのユーザ・クロックが使用できない。

内蔵のサブシステム・クロックをお使いください。

[メ モ]

— お問い合わせ先 —

【技術的なお問い合わせ先】

NEC半導体テクニカルホットライン
(電話：午前 9:00～12:00，午後 1:00～5:00)

電話 : 044-435-9494
FAX : 044-435-9608
E-mail : s-info@saed.tmg.nec.co.jp

【営業関係お問い合わせ先】

第一販売事業部

東京 (03)3798-6106, 6107,
6108

名古屋 (052)222-2375

大阪 (06)6945-3178, 3200,
3208, 3212

仙台 (022)267-8740

郡山 (024)923-5591

千葉 (043)238-8116

第二販売事業部

東京 (03)3798-6110, 6111,
6112

立川 (042)526-5981, 6167

松本 (0263)35-1662

静岡 (054)254-4794

金沢 (076)232-7303

松山 (089)945-4149

第三販売事業部

東京 (03)3798-6151, 6155, 6586,
1622, 1623, 6156

水戸 (029)226-1702

広島 (082)242-5504

高崎 (027)326-1303

鳥取 (0857)27-5313

太田 (0276)46-4014

名古屋 (052)222-2170, 2190

福岡 (092)261-2806

【資料の請求先】

上記営業関係お問い合わせ先またはNEC特約店へお申しつけください。

【インターネット電子デバイス・ニュース】

NECエレクトロニクスデバイスの情報がインターネットでご覧になれます。

URL(アドレス)

<http://www.ic.nec.co.jp/>

アンケート記入のお願い

お手数ですが、このドキュメントに対するご意見をお寄せください。今後のドキュメント作成の参考にさせていただきます。

[ドキュメント名] IE-75000-R/IE-75001-R ユーザーズ・マニュアル
(U15337JJ3V0UM00 (第3版))

[お名前など] (さしつかえのない範囲で)

御社名(学校名, その他) ()
ご住所 ()
お電話番号 ()
お仕事の内容 ()
お名前 ()

1. ご評価(各欄に)をご記入ください)

項 目	大変良い	良 い	普 通	悪 い	大変悪い
全体の構成					
説明内容					
用語解説					
調べやすさ					
デザイン, 字の大きさなど					
その他()					
()					

2. わかりやすい所(第 章, 第 章, 第 章, 第 章, その他)
理由 []

3. わかりにくい所(第 章, 第 章, 第 章, 第 章, その他)
理由 []

4. ご意見, ご要望

5. このドキュメントをお届けしたのは
NEC販売員, 特約店販売員, その他()

ご協力ありがとうございました。

下記あてにFAXで送信いただくか, 最寄りの販売員にコピーをお渡ししてください。

日本電気(株)NECエレクトロニクス
半導体テクニカルホットライン

FAX: (044) 435-9608

2000.6