

MJX440 for NB85E User's Manual

Rev. 1.12 2004/03/02

変更履歴

バージョン	変更内容	変更日
1.00	● 新規作成	02/03/18
1.01	● 付録「V850E/ME2 内蔵命令 RAM の取り扱い」を追加 ● コマンド「変更～I/O ポート」を追加	03/08/20
1.02	● 誤字、脱字修正	03/09/03
1.10	● コンフィグレーションファイルの設定画面の変更	04/02/09
1.11	● 誤字修正	04/02/27
1.12	● 誤字修正	04/03/02

ご注意

- 本マニュアルの一部または全部を無断で複製することはできません。
- 本製品を運用した結果の影響については、いかなる責任も負いません。
- 本製品の仕様および本マニュアルの内容は予告なく変更することがあります。
- Windows は、米国 Microsoft 社の登録商標です。
- MULTI は、米国 Green Hills Software 社の登録商標です。

©1999-2004 Lightwell Co., Ltd. All rights reserved.

Printed in Japan

住所: 〒167-0051 東京都杉並区荻窪 5-20-12

TEL: 03-3392-3331

FAX: 03-3393-3878

E-mail: ZAXSupport@lightwell.co.jp

URL: <http://www.lightwell.co.jp/ZAX/>

2004 年 03 月

この度は、MJX440 for NB85E(以下、本文中は MJX440)をご購入いただきまして、誠にありがとうございます。
ます。

本マニュアルの内容は、次のとおりです。

第一章 概要

製品構成、MJX440 の概略、各部の名称について記述しています。

第二章 ホスト インターフェースの設定

ホストインターフェース(PCI/ PCMCIA カード)の取り付け方法と、
設定方法、デバイスドライバのインストール方法について記述しています。

第三章 ハードウェアの接続

MJX440 とホストの接続方法、MJX440 とターゲット システムの接続方法について記述しています。

第四章 ソフトウェアのインストール

MJX440 を操作するソフトウェアのインストール方法について記述しています。

第五章 MJX440 の環境設定

MJX440 を使用する前に必要な、環境設定の方法について記述しています。

第六章 ソフトウェアの起動と終了

MJX440 を操作するソフトウェアの起動方法と終了方法について記述しています。

第七章 MJX440 コマンド

MJX440 コマンドの使い方について記述しています。

第八章 MJXDEBW メニューコマンド

MJXDEBW メニューコマンドの使い方について記述しています。

第九章 高速ダウンロード

高速ダウンロードを行なうための手順について記述しています。

付録

仕様、ターゲット システムの制限事項などの技術情報について記述しています。

もくじ

第一章 概要.....	8
1.1 製品構成.....	8
1.2 MJX440 の概略.....	11
1.3 各部の名称.....	15
第二章 ホスト インターフェースの設定	17
第三章 ハードウェアの接続.....	18
3.1 MJX440 とホストの接続.....	18
3.2 N-Wire ケーブルの接続.....	19
3.3 ROM プローブの接続.....	20
3.4 外部トリガ ケーブルの接続.....	27
3.5 電源の接続と電源投入手順.....	29
第四章 ソフトウェアのインストール.....	30
第五章 MJX440 の環境設定	37
第六章 ソフトウェアの起動と終了	39
第七章 MJXDEBW コマンド.....	43
ABORT	46
BATCH	47
BP	48
CLEAR	50
CONFIG.....	51
DUMP	52
EVENT	53
EXAMINE.....	56
FILL	57
FLASH	58
GO.....	60
HISTORY.....	61
INIT	62
JOURNAL.....	63
LOAD	64
MOVE	65
OUTTRIG.....	66
PIN	67
QUIT	68
REGISTER.....	69
SETLOAD.....	70

SETSAVE.....	71
STEP	72
TRACE	73
TRACE/A.....	74
TRACE/C.....	75
TRACE/M.....	76
TRACE/I.....	78
TRACE/R.....	81
UNASM	82
VERSION.....	83
WAIT	84
XPIN	85
第八章 MJXDEBW メニューコマンド	86
ファイル(F) ~ バッチ(B).....	87
ファイル(F) ~ ジャーナル開始(S).....	88
ファイル(F) ~ ジャーナル停止(E).....	88
ファイル(F) ~ トレース結果(T).....	89
エミュレーション(E) ~ 実行(G).....	90
エミュレーション(E) ~ 再スタート(R).....	90
エミュレーション(E) ~ 中断(B).....	91
エミュレーション(E) ~ ステップ(S).....	91
エミュレーション(E) ~ N-ステップ(N).....	92
エミュレーション(E) ~ ダウンロード(L).....	93
エミュレーション(E) ~ 初期化(I).....	94
エミュレーション(E) ~ フラッシュメモリ(F) ~ 書込(W).....	95
エミュレーション(E) ~ フラッシュメモリ(F) ~ 消去(E).....	96
表示(V) ~ メモリ(M).....	97
表示(V) ~ レジスタ(R)	99
表示(V) ~ 逆アセンブル(D).....	100
表示(V) ~ トレース結果(T).....	101
表示(V) ~ コマンド応答クリア(C)	107
表示(V) ~ オプション(O) ~ フォント(F)	108
表示(V) ~ オプション(O) ~ 表示(V).....	109
変更(M) ~ メモリ(M).....	110
変更(M) ~ I/O ポート(P).....	111
変更(M) ~ レジスタ(R).....	112
変更(M) ~ アセンブラ(A).....	113

変更(M) ~ フィル(F).....	114
変更(M) ~ 移動(M).....	115
設定(S) ~ ブレークポイント(B).....	116
設定(S) ~ トレース(T).....	118
設定(S) ~ イベント(E).....	122
設定(S) ~ 設定の読出(L).....	125
設定(S) ~ 設定の保存(S).....	126
設定(S) ~ コンフィグレーション(C) ~ 設定(S).....	127
設定(S) ~ コンフィグレーション(C) ~ 表示(V).....	129
設定(S) ~ コンフィグレーション ~ 上書き保存(E).....	130
設定(S) ~ コンフィグレーション ~ 名前を付けて保存(A).....	131
ヘルプ(H) ~ トピックの検索(H).....	132
ヘルプ(H) ~ MJXDEBW のバージョン情報(A).....	133
第九章 高速ダウンロード.....	134
付録 A 仕様.....	135
付録 B ターゲット システムの制限事項.....	136
付録 C N-Wireコネクタ.....	137
付録 D ROMプローブ.....	139
付録 E 対応 ROMピンアサイン.....	145
付録 F LED.....	149
付録 G レジスタ名一覧.....	150
付録 H MJX バイナリ ファイル.....	151
付録 I ターゲット システムへのプロービング.....	152
付録 J フラッシュ メモリ デバイス情報定義 ファイル レイアウト.....	158
付録 K 対応フラッシュ メモリー覧.....	162
付録 L V850/ME2 内蔵命令 RAM の取り扱い.....	176
検索項目.....	177

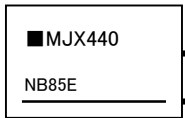
第一章 概要

製品構成、MJX440 の概略、各部の名称について記述しています。

1.1 製品構成

出荷時の MJX440 for NB85E パッケージの中には、次のものが含まれています。万一、欠品がございましたら、弊社までご連絡ください。

- MJX440 for NB85E 本体



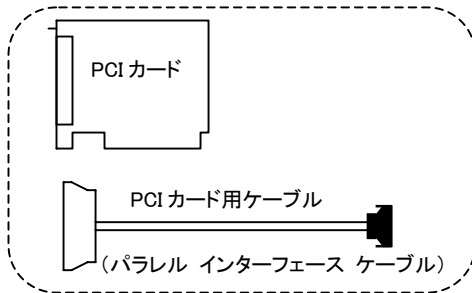
- AC アダプタ



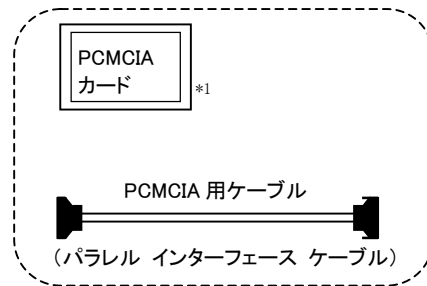
- AC コード



- ホスト インターフェース



または



- N-Wire ケーブル



*1 カードの正しい名称は「PC カード」です。

PCMCIA(Personal Computer Memory Card International Association)は PC カードの標準化団体名ですが、このマニュアルでは PC カードを全て「PCMCIA カード」と表記しています。

第一章 概要

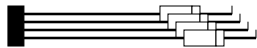
- ROMプローブ*1 2本(J-101A、J-102A、J-103AまたはJ-104A 各2本)



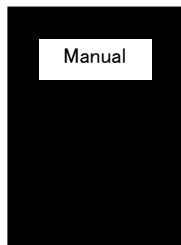
- ROMプローブ ケーブル2 *1本



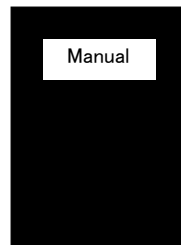
- 外部トリガ ケーブル 1種類



- マニュアル2冊

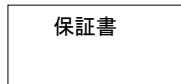


MJX440 for NB85E
User's Manual

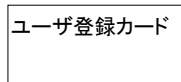


MJX440
HostInterface
Installation Manual

- 保証書

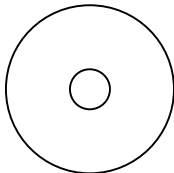


- ユーザー登録カード



【重要】 ユーザー登録カードは、必要事項をご記入の上、弊社までご返送ください。

- CD-ROM



*1 ROMの種類によって、出荷時に含まれているROMプローブが異なります。詳しくは、「表 1-1 出荷時のROMプローブ」を参照してください。
ただしエミュレーションメモリなしモデルにはROMプローブおよびROMプローブケーブルは添付されません。

第一章 概要

出荷時のROMプローブは、次のとおりです。ROMの種類によって異なります。

ROMの種類	ROMプローブ*1	
	ROMプローブ基板	ROMプラグ*2
27010(8bit) 27020(8bit) 27040(8bit) 27080(8bit) 271000(8bit)	J-101A×2本	32pin×4*3
27C4000(16bit)	J-102A×2本	40pin×2
27C8000(16bit) 27C16000(16bit)	J-102A×2本	42pin×2
271024(16bit) 272048(16bit) 274096(16bit)	J-103A×2本	40pin×2
27C4000(8bit)	J-104A×2本	40pin×4*3
27C8000(8bit) 27C16000(8bit)	J-104A×2本	42pin×4*3

表 1-1 出荷時のROMプローブ

*1 ROMプローブの詳細な図は「1.3 各部の名称」、「ROMプローブ」および「付録D ROMプローブ」を参照してください。

*2 ROMのピン数と接続可能な最大ROM個数です。実際の接続数はターゲットシステムにより異なります。

*3 ROMプローブ基板J-101A、J-104AはROMプラグが2本あるため、同時に2つに接続出来ます。

1.2 MJX440 の概略

MJX440 for NB85Eは、N-Wire コネクタを備えたNB85E コアを搭載した ASIC プロセッサのターゲット システムをデバッグするための、開発支援装置です。

MJX440 のおもな特長

- ターゲット システムの[N-Wire コネクタ](#)を使用するため、接続が簡単です。
- 高速な CPU でも安定して動作します。
- 任意で、[ROM インサーキット接続](#)ができます。ROM 上のプログラムもデバッグできます。
- 高級言語デバッガ[MULTI](#)が使用できます。
- MJXDEBW デバッガ上で[MJXDEBW コマンド](#)が使用できます。MJX440 が持つ機能をフルに使用できます。
- [リアルタイムトレース](#)機能があります。
- ハードウェアブレークポイント機能があります。
- プログラムのダウンロードが高速です。

N-Wire 接続時 480K バイト/秒 *1 (JTAG クロック 25MHz の場合)

ROM インサーキット接続時 4M バイト/秒

最小構成

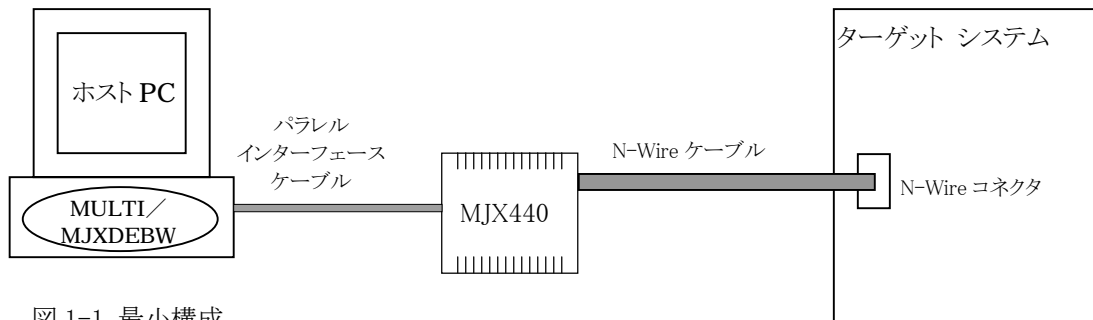


図 1-1 最小構成

*1 ダウンロードの速さは JTAG クロックを高く設定する程、高速になります。

MJX440 は図 1-1 の最小構成で動作します。最小構成で使用した場合は、ターゲット システムの RAM 上のプログラムをデバッグすることができます。プログラムは N-Wire ケーブルをとおして、ターゲット システム上の RAM へダウンロードされます。

最小構成の場合、ROM 上のプログラムを実行することはできませんが、ROM 領域にプログラムをダウンロードしたり、[ソフトウェア ブレークポイント](#)を設定することができません。ROM 上のプログラムをデバッグする場合は、さらに ROM インサーキット接続する必要があります。

ROM インサーキット接続

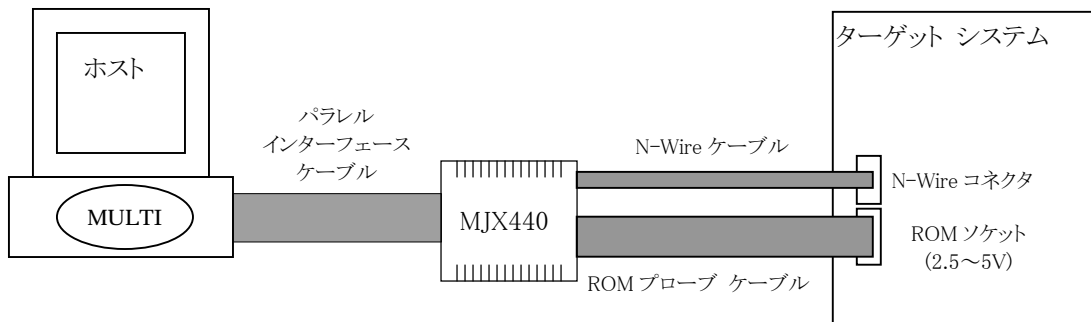


図 1-2 ROM インサーキット接続

図 1-2 のように、ROM インサーキット接続した場合は、ROM 上のプログラムもデバッグできるようになります。ROM 上のプログラムは、ターゲット システムの RAM ではなく、MJX440 内のエミュレーション メモリへダウンロードされます。

また、ROM の電圧は、2.5~5V の範囲内で自動認識されます。

外部トリガ ケーブル接続

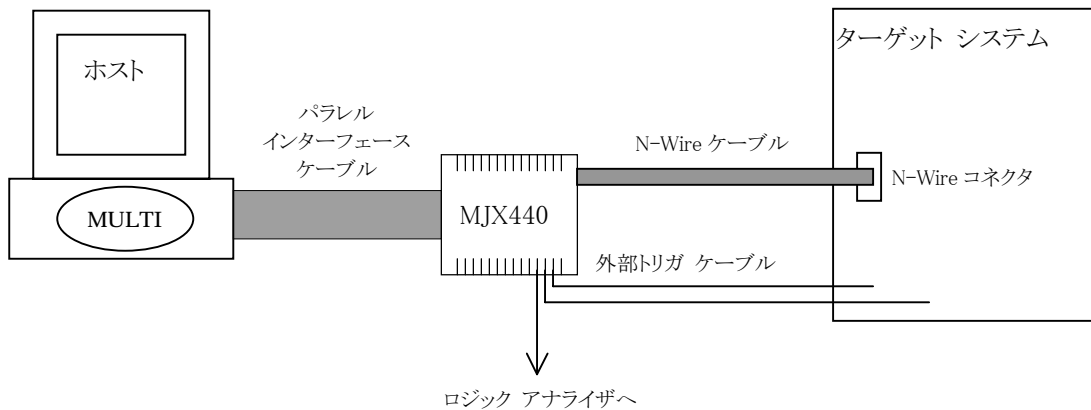


図 1-3 外部トリガ ケーブル接続

図 1-3 のように、[外部トリガケーブル](#)を接続し、次の機能を実現させることができます。

- MJX440 コマンドから操作できる汎用信号をターゲット システムに出力する。(出力)
- ターゲット システムの信号の状態を、MJX440 上の[LED](#)に表示させる。(入力)
- トレース トリガを、ロジック アナライザのトリガ信号として使用する。(出力)
- ターゲット システムの信号の状態を、リアルタイム トレース情報として記録する。(入力)
- 外部入力信号でターゲット プログラムをブレイクする。(入力)

MJX440 を使用する前の準備作業

MJX440 を使用する前には、次の準備作業が必要です。第二章から第五章までを参照して行ってください。「ハードウェアの接続」以外は、購入時に一度だけ行えばよい作業です。

- [ホスト インターフェースの設定](#)
- [ハードウェアの接続](#)
- [ソフトウェアのインストール](#)
- [MJX440 の環境設定](#)

準備作業が終了したら、[第六章](#)を参照して、MJX440 を操作するソフトウェア([MULTI](#)または[MJXDEBW](#))を起動してください。ソフトウェアが正常に起動できれば、準備作業は終了です。ソフトウェアが正常に起動できなかった場合は、準備作業に誤りがないか、確認してください。

コンパイラや MULTI の使い方については、それぞれのマニュアルとリリース ノートを参照してください。また、MJX440 独自の機能を使用するための MJX440 コマンドは、[第七章](#)および[第八章](#)を参照してください。

高速ダウンロードについては、[第九章](#)を参照してください。

MULTI について

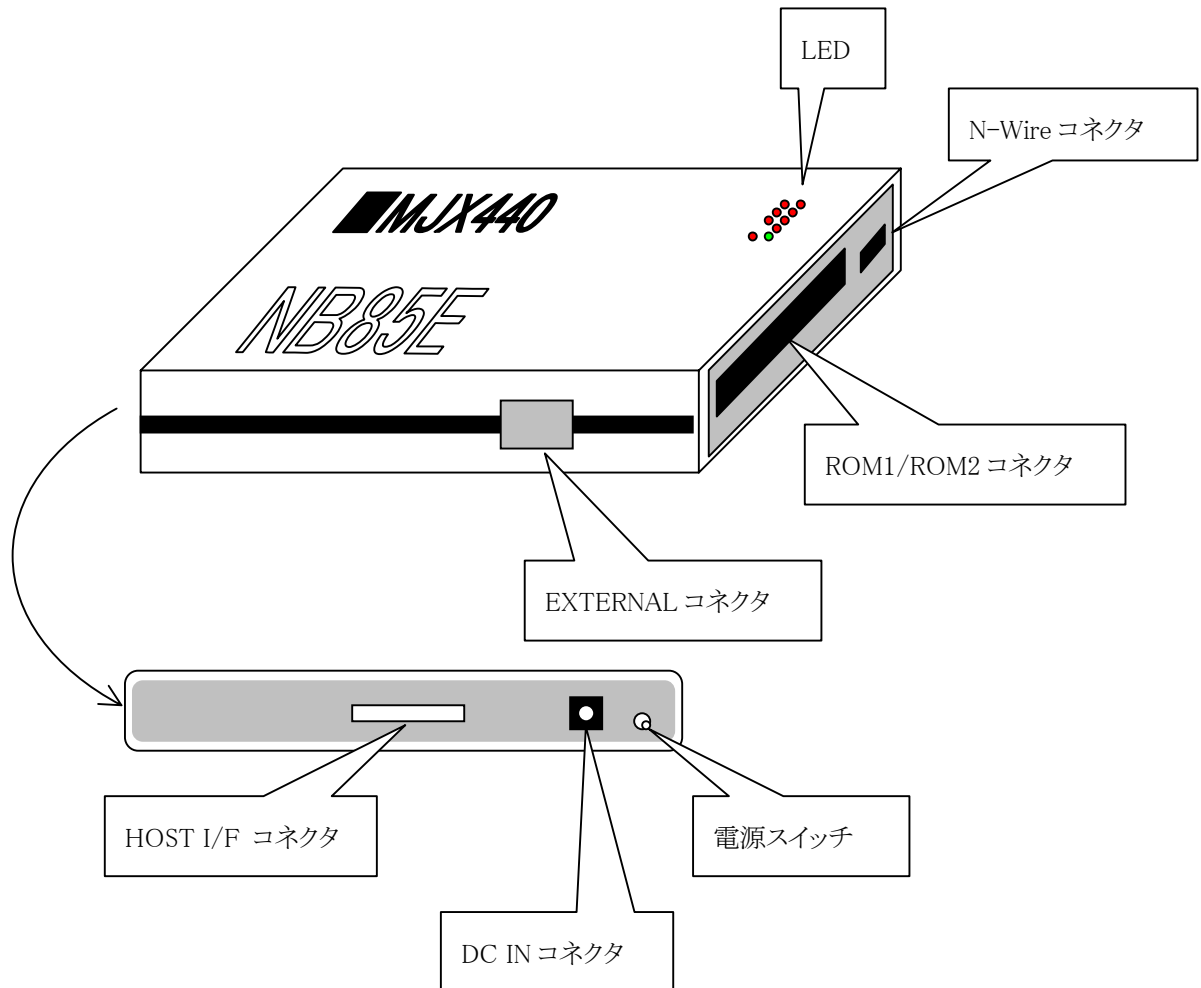
MULTI はいろいろな環境で実行することができる高級言語デバッガです。サーバープログラム MJXSERV を呼び出すことによって、MJX440 の環境で実行させることができますようになります。

MJXDEBW について

MJXDEBW は MJX440 コマンドのみをサポートする簡易デバッガです。高級言語デバッグをしない場合や、バッチ処理機能を使って、ターゲット システムの検査をする場合などに使用することができます。

1.3 各部の名称

MJX440 本体



電源スイッチ

ON で電源投入、OFF で電源切断します。

DC IN コネクタ

AC アダプタを接続します。

HOST I/F コネクタ

ホスト インターフェース ケーブルを接続します。

N-Wire コネクタ

N-Wire ケーブルを接続します。

ROM1/ROM2 コネクタ

ROMプローブ ケーブルを接続します。上部がROM1、下部がROM2 コネクタです。

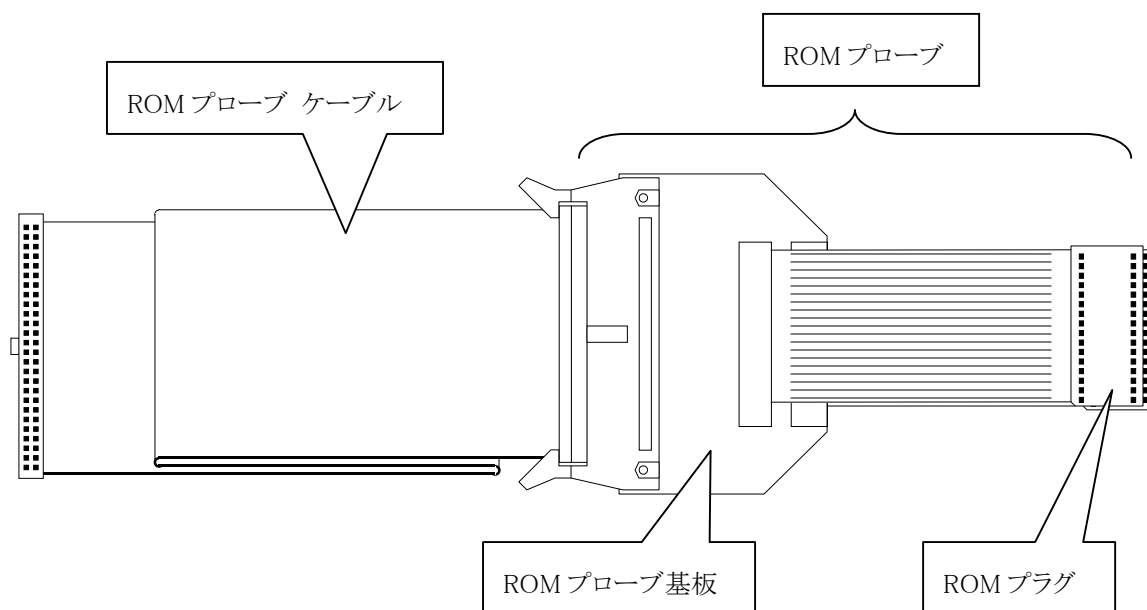
EXTERNAL コネクタ

外部トリガ ケーブルを接続します。

[LED](#)

電源やコネクタの接続状態を表示します。

ROM プローブ



- | | |
|---------------|---|
| ROM プローブ ケーブル | MJX440 と ROM プローブ を接続するケーブル |
| ROM プローブ | ターゲット システムの ROM へ接続するプローブ全体の総称 |
| ROM プローブ 基板 | ROM プローブの基板部分 (ジャンパ設定 があります) |
| ROM プラグ | ターゲット システムの ROM ソケットへ接続する部分 |

第二章 ホスト インターフェースの設定

ホスト インターフェース(PCIカード、またはPCMCIAカード)の取り付け方法と、そのデバイス ドライバのインストール方法については、次の別冊マニュアルを参照してください。

The image shows the cover of a manual. It has a black background with a white rectangular box in the center. Inside the box, the text "MJX440 Host Interface Installation Manual" is written in a black serif font, centered and stacked in three lines.

MJX440
Host Interface
Installation Manual

第三章 ハードウェアの接続

MJX440 とホストの接続方法、MJX440 とターゲット システムの接続方法について記述しています。

【重要】 MJX440 と他の機器との接続は、必ず機器の電源を切ってから行なってください。

3.1 MJX440 とホストの接続

ホストに取り付けた PCI カード、または PCMCIA カードと、MJX440 の [HOST I/F コネクタ](#) を、パラレル インターフェース ケーブルで接続してください。

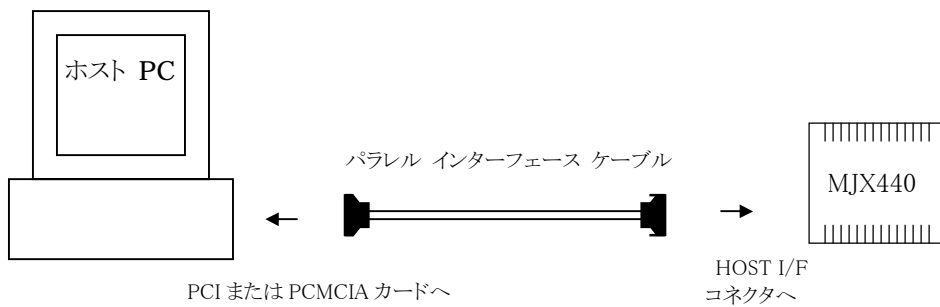


図 3-1 MJX440 とホストの接続 / パラレル接続

【注意】 PCMCIA 用ケーブルのカードとの接続コネクタの厚みにより、以下の制限事項があります。

- ◆ PCMCIA カード スロットが 2 つある場合でも、カードを 1 枚しか使用できない場合があります。また、下のスロットにしかカードを差せない場合があります。
- ◆ カード 2 枚を無理に差し込むと、PCMCIA カード スロット、および PCMCIA カードのコネクタ部が壊れる場合があります。
- ◆ PCMCIA カード スロットが 1 つだけの機種では、使用できない場合があります。

3.2 N-Wire ケーブルの接続

N-Wire プロープ

MJX440 の [N-Wire コネクタ](#)と、ターゲット システム上の N-Wire コネクタを、N-Wire ケーブルで接続してください。

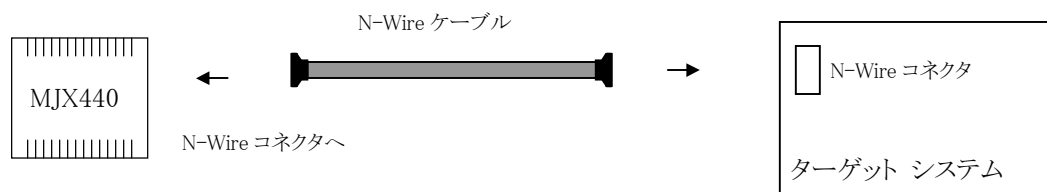


図 3-2 N-Wire ケーブルの接続

【注意】 コネクタには向きがあります。コネクタの△マーク同士を合わせるように、接続してください。

3.3 ROM プローブの接続

ROM プローブの接続は、ROM インサーキット接続をする場合のみ必要な作業です。

はじめに、ROM の種類に合わせて、ROM プローブ基板のジャンパを設定してください。詳しい設定方法は、「[付録D ROMプローブ](#)」を参照してください。

つぎに、添付されている ROM プローブを、ROM プローブ ケーブルに接続してください。

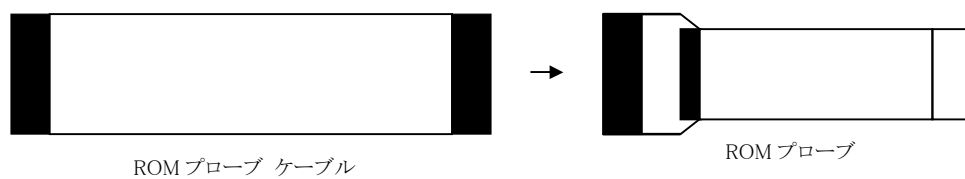


図 3-3 ROM プローブと ROM プローブ ケーブルの接続 1

つぎに、ROM プローブをターゲット システムの ROM ソケットへ接続し、もう一方の ROM プローブ ケーブルを MJX440 の ROM1/ROM2 コネクタへ接続します。^{*1}

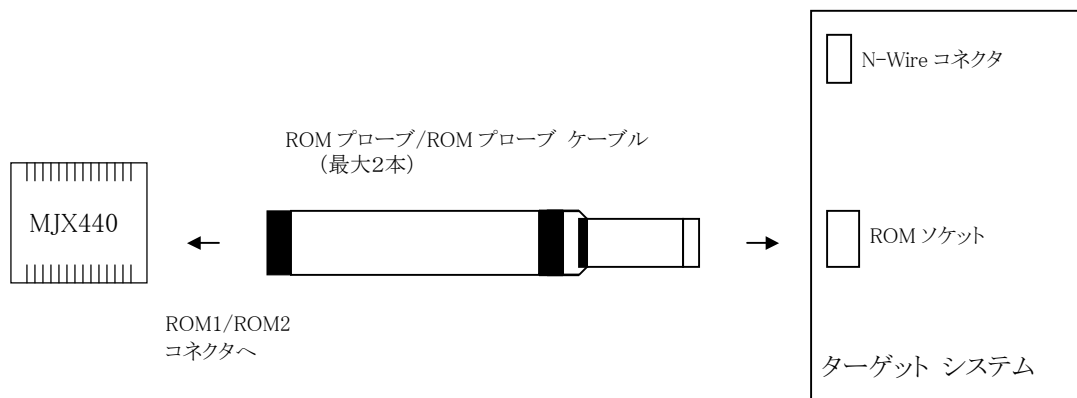


図 3-4 ROM プローブと ROM プローブ ケーブルの接続 2

^{*1} 必要な ROM プローブ本数は使用する ROM のタイプと個数により異なります。

第三章 ハードウェアの接続

ROMプローブの接続方法は、ターゲット システムの次の項目によって異なります。

- ROM1個あたりのデータ バス幅
- ROM の個数
- ROM 全体でのアクセス バス幅

図 3-5-1～3-5-9 の中から、ターゲット システムと一致する接続図をひとつ選び、図にしたがって ROMプローブとROMプローブ ケーブルを接続してください。

【注意】 MJX440 およびターゲット システムの電源を切ってから行なってください。

【注意】 ROMプローブを逆差ししないでください。

【注意】 ROMプローブ J-101AとJ-104A に接続できる二つの ROM プラグは、ROMプローブ基板上のシルク印刷文字(JROM1 または JROM2)で区別してください。

バス幅 8 ビット ROM 1 個、ROM アクセス バス幅 8 ビットの場合
(点灯 LED: ROM1)

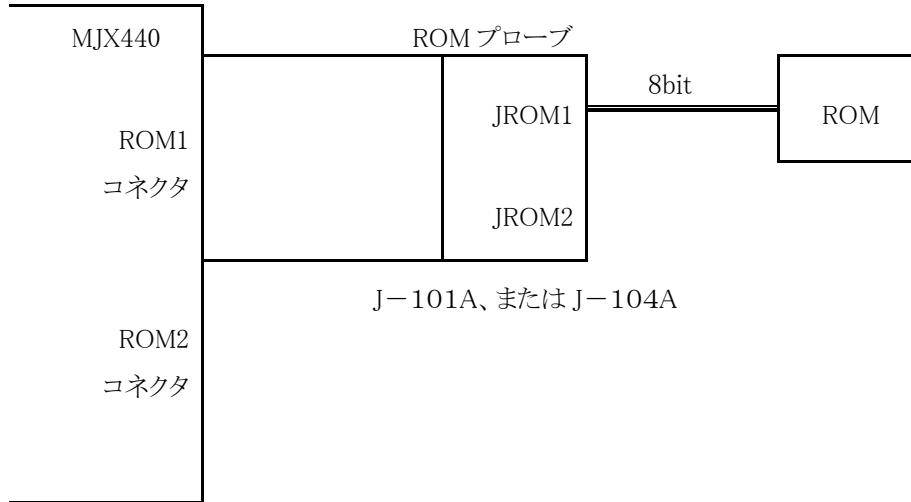


図 3-5-1 ROM プローブの接続 1

バス幅 8 ビット ROM 2 個、ROM アクセス バス幅 8 ビットの場合
(点灯 LED: ROM1)

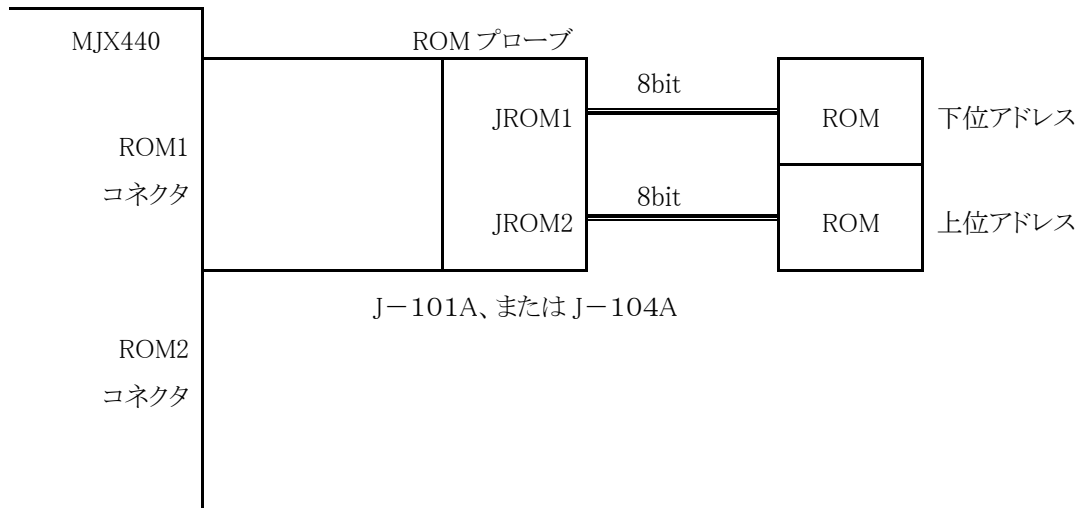


図 3-5-2 ROM プローブの接続 2

バス幅 8 ビット ROM 4 個、ROM アクセス バス幅 8 ビットの場合
 (点灯 LED: ROM1、ROM2)

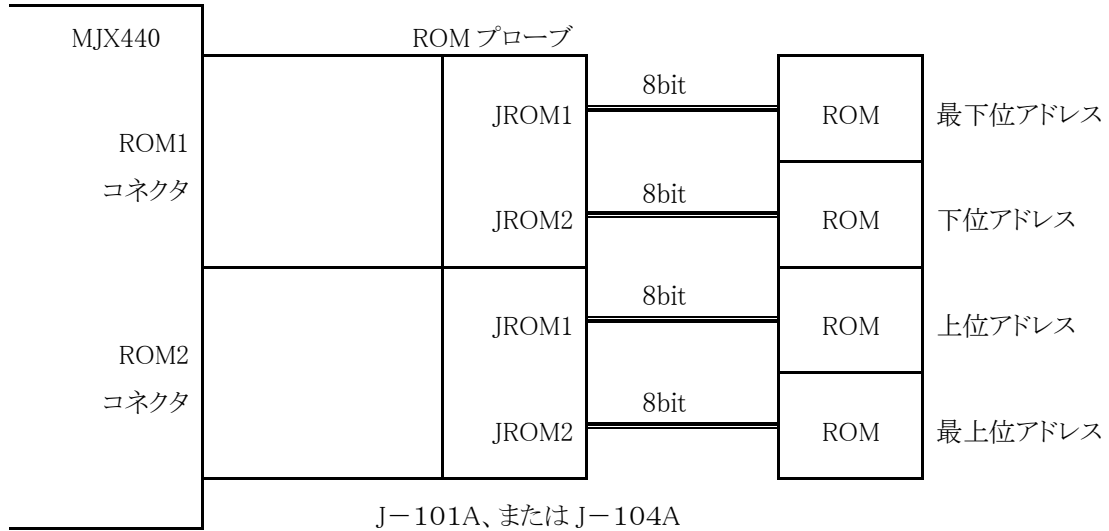


図 3-5-3 ROM プロブの接続 3

バス幅 8 ビット ROM 2 個、ROM アクセス バス幅 16 ビットの場合
 (点灯 LED: ROM1)

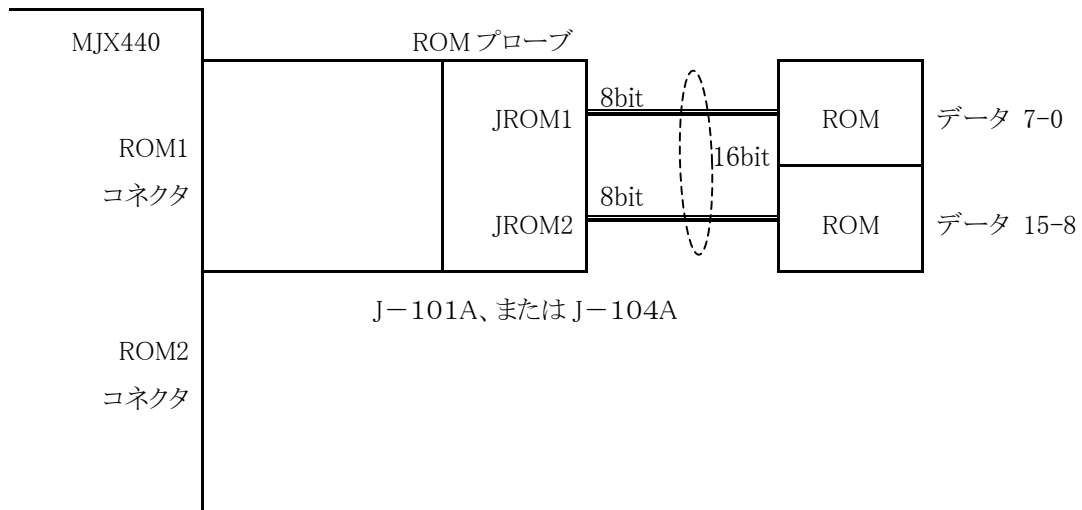


図 3-5-4 ROM プロブの接続 4

バス幅 8 ビット ROM 4 個、ROM アクセス バス幅 16 ビットの場合
 (点灯 LED: ROM1、ROM2)

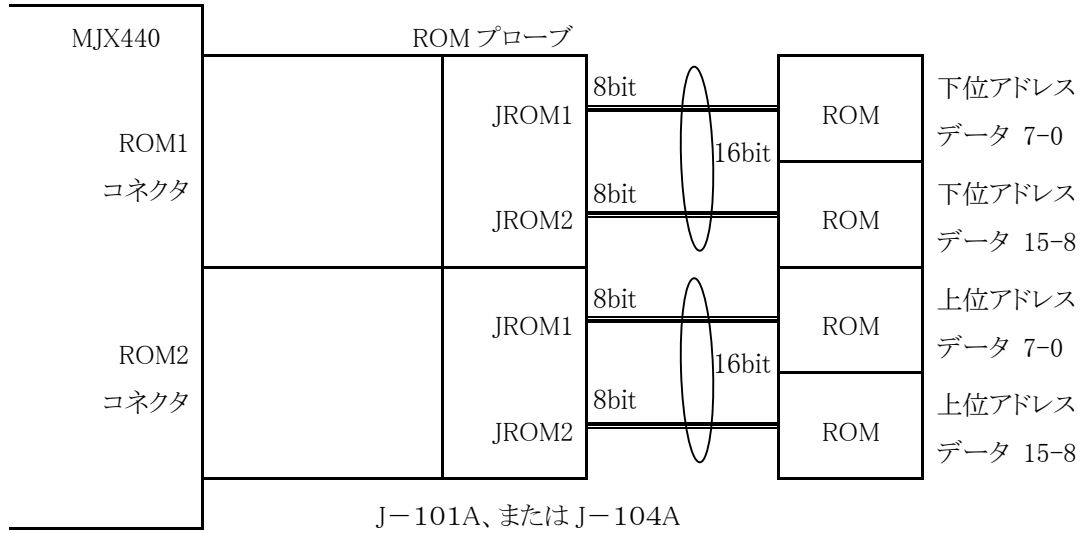


図 3-5-5 ROM プローブの接続 5

バス幅 8 ビット ROM 4 個、ROM アクセス バス幅 32 ビットの場合
 (点灯 LED: ROM1、ROM2)

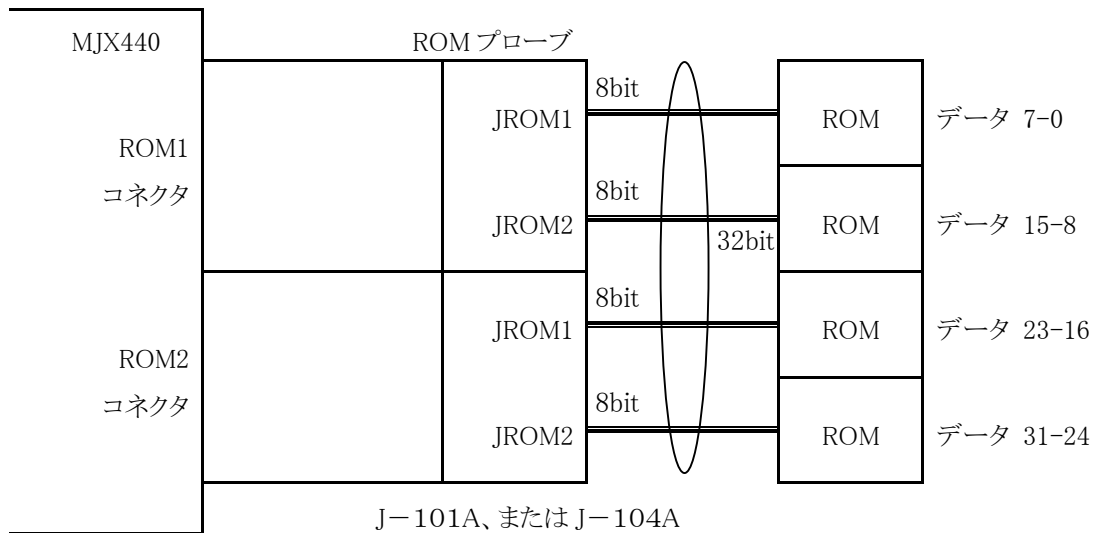


図 3-5-6 ROM プローブの接続 6

バス幅 16ビット ROM 1 個、ROM アクセス バス幅 16ビットの場合
 (点灯 LED: ROM1)

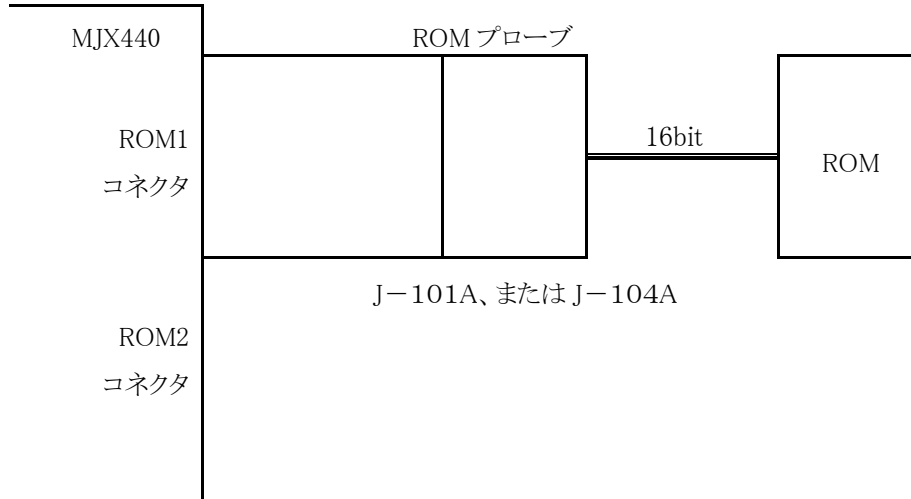


図 3-5-7 ROM プローブの接続 7

バス幅 16ビット ROM 2 個、ROM アクセス バス幅 16ビットの場合
 (点灯 LED: ROM1、ROM2)

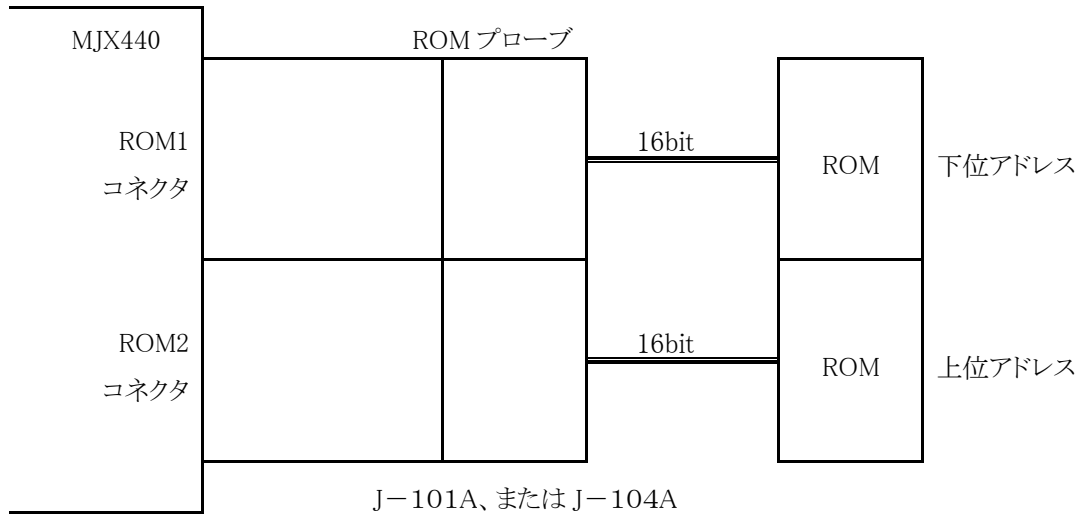


図 3-5-8 ROM プローブの接続 8

バス幅 16ビット ROM 2 個、ROM アクセス バス幅 32ビットの場合
(点灯 LED: ROM1、ROM2)

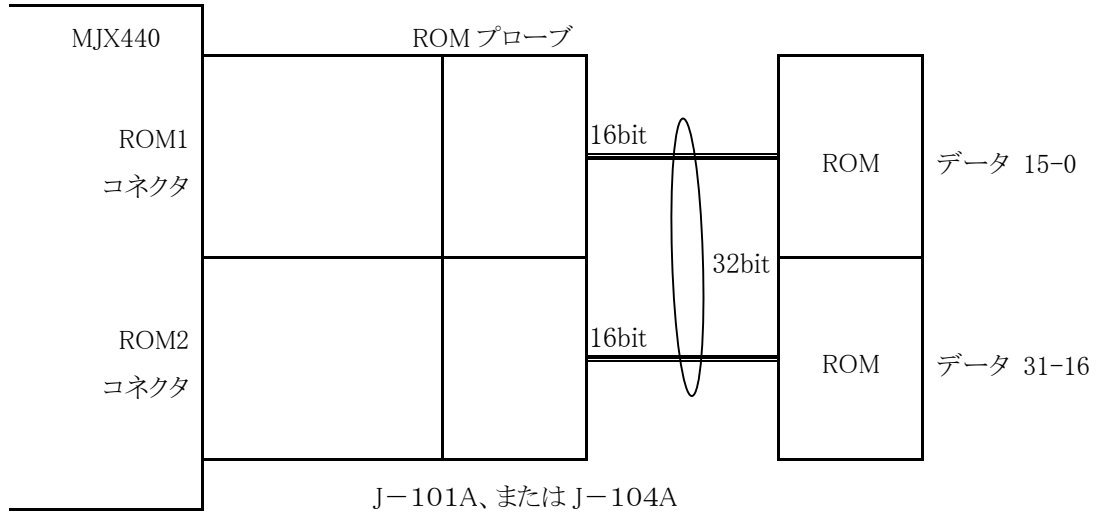


図 3-5-9 ROM プローブの接続 9

3.4 外部トリガ ケーブルの接続

ターゲット システムの信号の状態を LED に表示させたり、MJX440 が出力するトレース トリガ信号をロジック アナライザの入力として使用する場合は、外部トリガ ケーブルを使用します。これらの機能を使用しない場合は、外部トリガ ケーブルを接続する必要はありません。

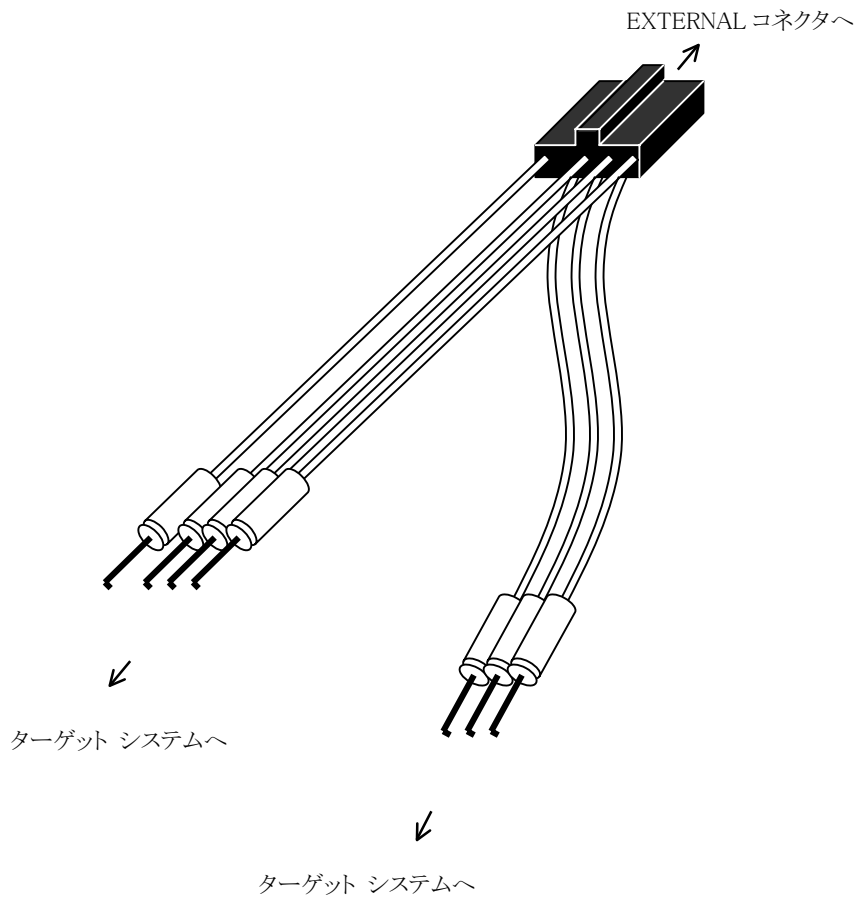


図 3-5 外部トリガ ケーブルの接続

第三章 ハードウェアの接続

外部トリガ ケーブルの信号の詳細は、次のとおりです。信号は、クリップとケーブルの色で区別します。

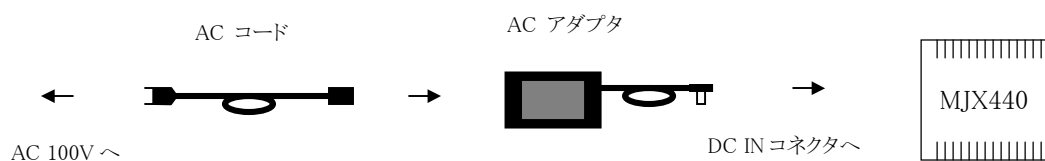
信号名	色	入出力	機能
EXTIN1	クリップ 黄 ケーブル 茶	TTL 入力	接続した信号が HIGH レベルの場合、対応する LED が点灯します。 ^{*1} また、EXTIN1 と EXTIN2 については、リアルタイム トレースにも、信号の状態が記録されます。
EXTIN2	クリップ 黄 ケーブル 赤	TTL 入力	
EXTIN3	クリップ 黄 ケーブル 橙	TTL 入力	
EXTOUT1	クリップ 赤 ケーブル 茶	3.3V O.D.出力 ^{*2}	MJX440 コマンドの XPIN コマンドで、LOW または HIGH レベルを出力します。
EXTOUT2	クリップ 赤 ケーブル 赤	3.3V O.D.出力	
TRGOUT-	クリップ 緑 ケーブル 緑	3.3V O.D.出力	トレース トリガ以降 LOW レベルを出力します。ロジック アナライザのトリガ入力として使用できます。
GND	クリップ 黒 ケーブル 黒		ターゲット システムの GND に接続します。

^{*1} MJX440 本体上の信号名は、EXTIN3→EXI 3 のように省略して表記されています。

^{*2} O.D.は、オープン ドレインの略です。

3.5 電源の接続と電源投入手順

すべてのケーブルの接続が終了したら、AC アダプタと AC コードを接続してください。このとき、MJX440 の電源スイッチが OFF の状態になっていることを確認してください。



機器の電源投入は、次の手順で行なってください。

1. ホスト PC
2. MJX440
3. ターゲット システム

機器の電源切断は、次の手順で行なってください。

1. ターゲット システム
2. MJX440
3. ホスト PC

【重要】 電源投入および切断の手順を間違えると、機器が破壊される場合があります。

【重要】 電源投入時には、機器の接続および着脱をしないでください。

第四章 ソフトウェアのインストール

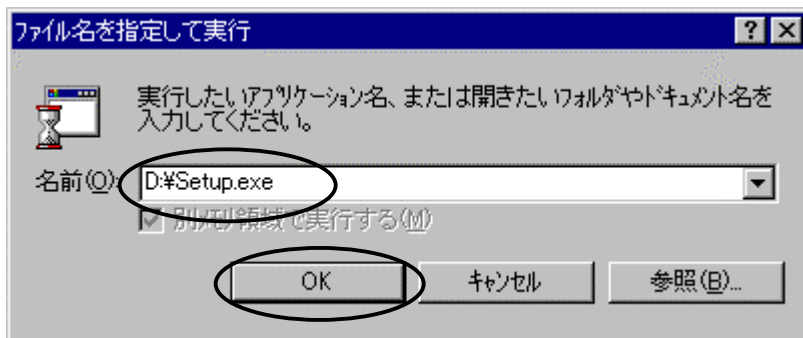
MJX440 を操作するソフトウェアのインストール方法について記述しています。

インストールは、以下の手順にしたがって行ってください。

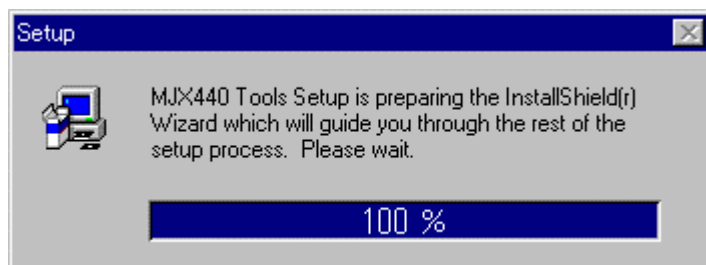
1. Green Hills Software 社の統合開発環境 MULTI をインストールします。
2. CD-ROMドライブに「MJX for NB85E Tools Disk」の CD-ROM をセットします。

「スタート」、「ファイル名を指定して実行(R)...」の順に選択し、「ファイル名を指定して実行」ダイアログを表示します。

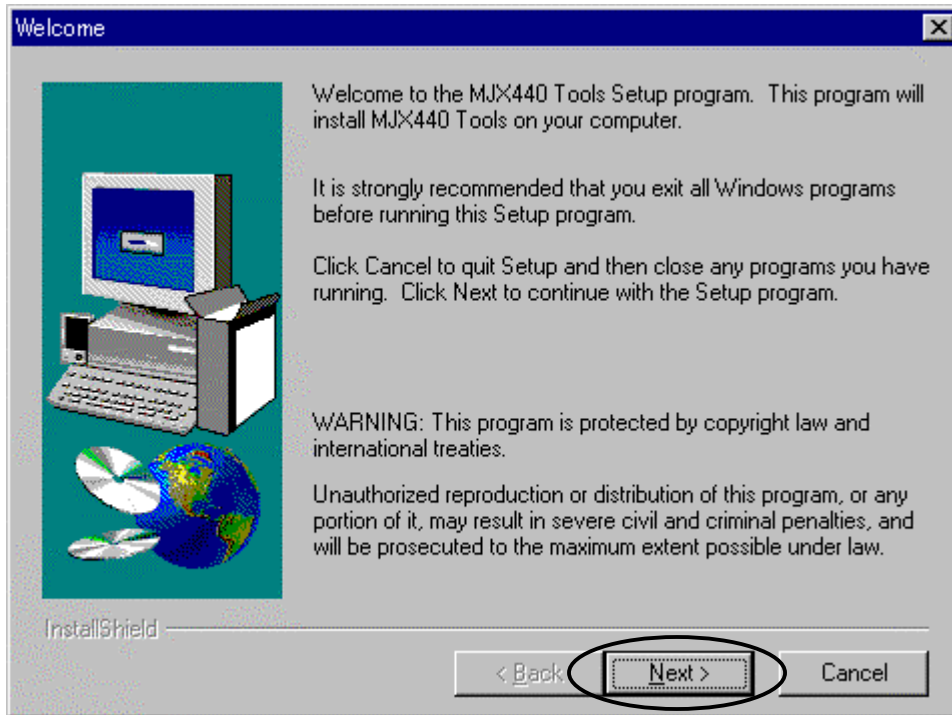
3. 「名前(O)」にセットアップ プログラム Setup.exe を指定し、「OK」をクリックします。(CD-ROMドライブが Dドライブの場合、「D:¥Setup.exe」)



4. セットアップ プログラムが起動します。

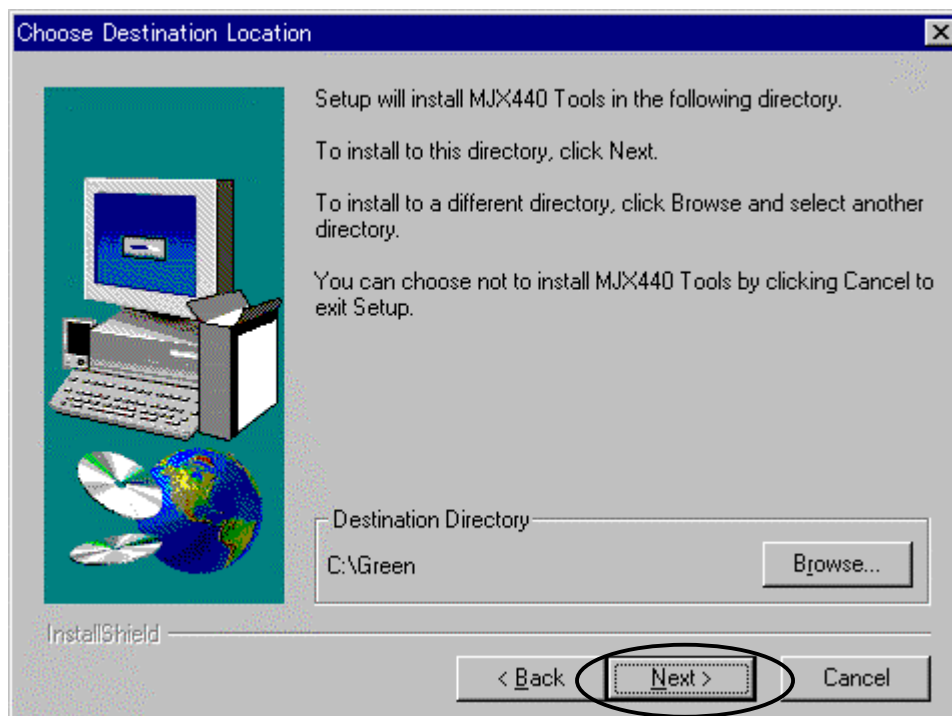


5. 「Welcome」ダイアログが表示されるので、「Next >」をクリックします。

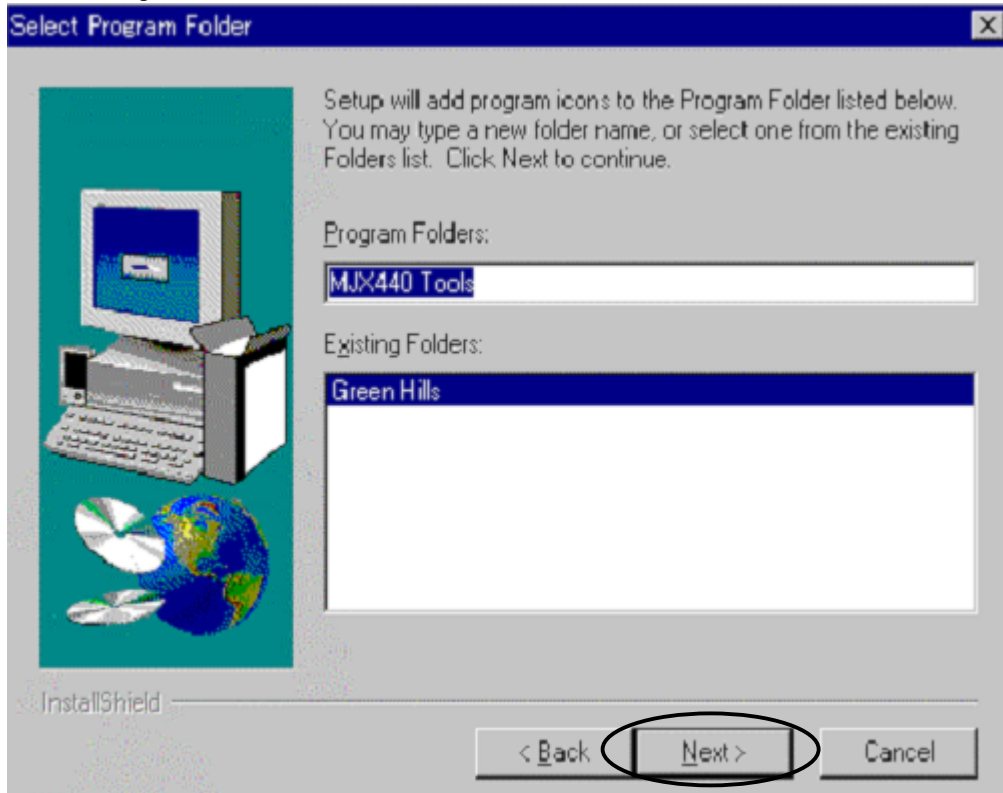


6. 「Choose Destination Location」ダイアログが表示されるので、「Browse...」ボタンをクリックしてインストール先を指定します。先にインストールしたMULTIと同じディレクトリを、インストール先に指定してください。(デフォルトは「C:\Green」です)

「Next >」をクリックします。

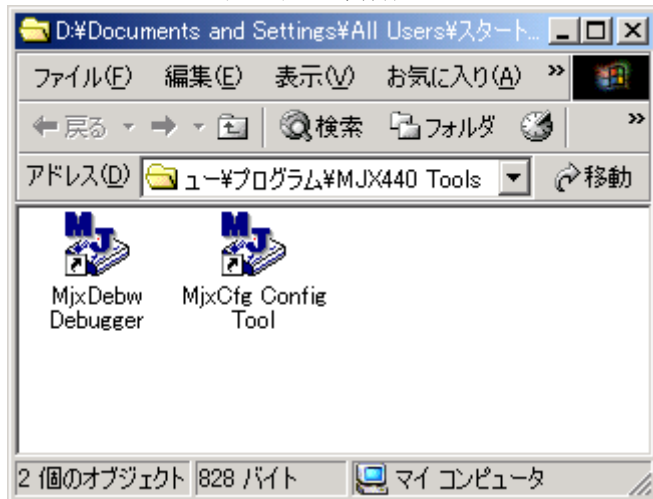


「Select Program Folder」ダイアログが表示されるので、「Next >」ボタンをクリックします。

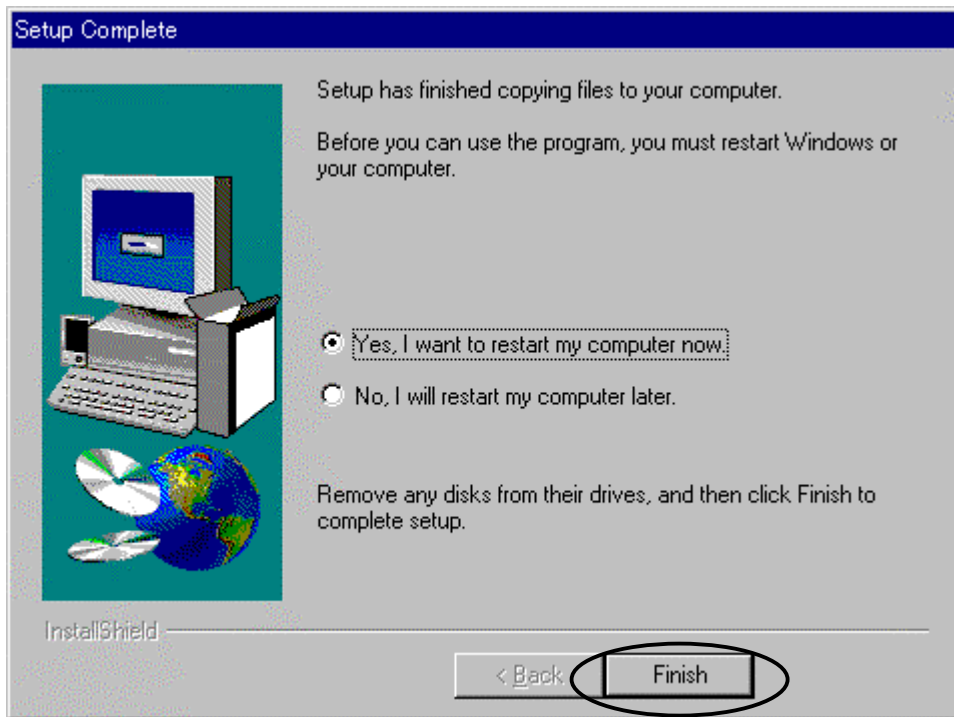


7. インストールが開始します。

8. インストールが終了すると、作成されたプログラムフォルダが表示されます。



9. 「Setup Complete」ダイアログが表示されるので、「Finish」ボタンをクリックし、システムを再起動します。



【重要】 インストーラはシステムディレクトリ*1 にライブラリMSVCRT.DLLをインストールします。
古いライブラリはMSVCRTxx.DLL (xxはバージョン番号) のようにリネームされます。

*1 システムディレクトリのデフォルトは、Windows9xの場合はC:\Windows\System、WindowsNT4.0およびWindows2000では、C:\Winnt\System32です。

インストールされるファイルは次のとおりです。

MJXDEBW.EXE	簡易デバッガ プログラム MJXDEBW
MJXCFG.EXE	コンフィグ支援ツール MJXCFG (コンフィグレーション ファイル作成用)
MJXSERV.EXE	MJX 用 MULTI サーバー プログラム MJXSERV
MJXCVT.EXE	MJX バイナリ ファイル変換プログラム MJXCVT
WNTIX.EXE	インストール用プログラム (使用しない)
MJX4020.INI	MJXDEBW 用コンフィグレーション ファイルのサンプル (CW4020)
MJX4102.INI	MJXDEBW 用コンフィグレーション ファイルのサンプル (TR4102)
MJXARM.INI	MJXDEBW 用コンフィグレーション ファイルのサンプル (ARM9)
MJXNB85E.INI	MJXDEBW 用コンフィグレーション ファイルのサンプル (NB85E)
MJXV831.INI	MJXDEBW 用コンフィグレーション ファイルのサンプル (V831)
MJXV832.INI	MJXDEBW 用コンフィグレーション ファイルのサンプル (V832)
Mjx4kc.INI	MJXDEBW 用コンフィグレーション ファイルのサンプル (MIPS4kc)
MjxVR55.INI	MJXDEBW 用コンフィグレーション ファイルのサンプル (VR5500)
MJXDDBAPI.DLL	MJXSERV-MJXDEBW 通信ライブラリ
MJXASM.DLL	アセンブラ/逆アセンブラ ライブラリ
MJXARM.DLL	ARM ライブラリ
MJXEJTAG.DLL	TR4102/CW4020 ライブラリ
MJXNB85E.DLL	NB85E ライブラリ
MJXV831.DLL	V831 ライブラリ
MJXV832.DLL	V832 ライブラリ
MJ2Ej2R6. DLL	EJTAG2.5/2.6 ライブラリ
Mj2VR55.DLL	V5500 ライブラリ
MSVCP60.DLL	Microsoft Visual C++ ランタイム ライブラリ
MSVCRT.DLL	Microsoft Visual C++ ランタイム ライブラリ
CfgResource.DLL	コンフィグリソース ライブラリ
MjxARMJ.PDF	ARM 和文ヘルプのファイル
MjxEJtgJ.PDF	CW4020/TR4102 和文ヘルプのファイル
MjxEJtgE.PDF	CW4020/TR4102 英文ヘルプのファイル
MjxNB85J.PDF	NB85E 和文ヘルプのファイル
MjxCfgJ.PDF	コンフィグレーションヘルプ画面のファイル
MjxiFJ.PDF	ホストインターフェース 和文ヘルプのファイル
MjxiFE.PDF	ホストインターフェース 英文ヘルプのファイル
MjxV83XJ.PDF	V831/V832 和文ヘルプのファイル
MjxV83XE.PDF	V831/V832 英文ヘルプのファイル
MJX440.ICO	MJX440 アイコン ファイル

第四章 ソフトウェアのインストール

MjxCARMJ.PDF	MJX330 for ARM 和文ヘルプのファイル
Mj2CARMJ.PDF	MJX330 for ARM (汎用品) 和文ヘルプのファイル
Mj2CEj26J.PDF	MJX330 for EJTAG2.5/2.6 和文ヘルプのファイル
MjxCNB85J.PDF	MJX330 for NB85E 和文ヘルプのファイル
Mj2CVR5J.PDF	MJX330 for VR5500 和文ヘルプのファイル

プロセスとハードウェアの構成は次のとおりです。

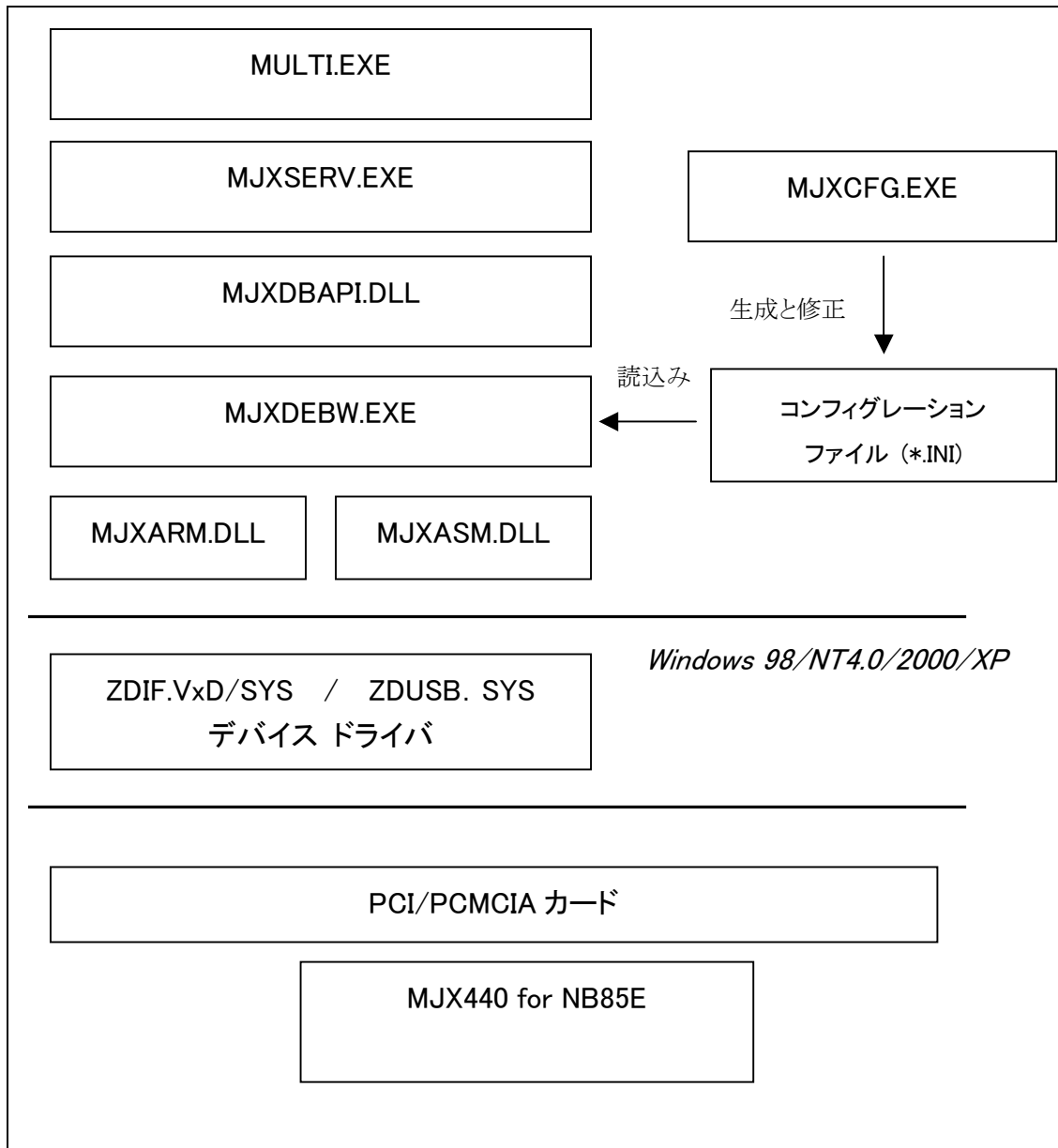


図 4-1 プロセスとハードウェアの構成/Windows

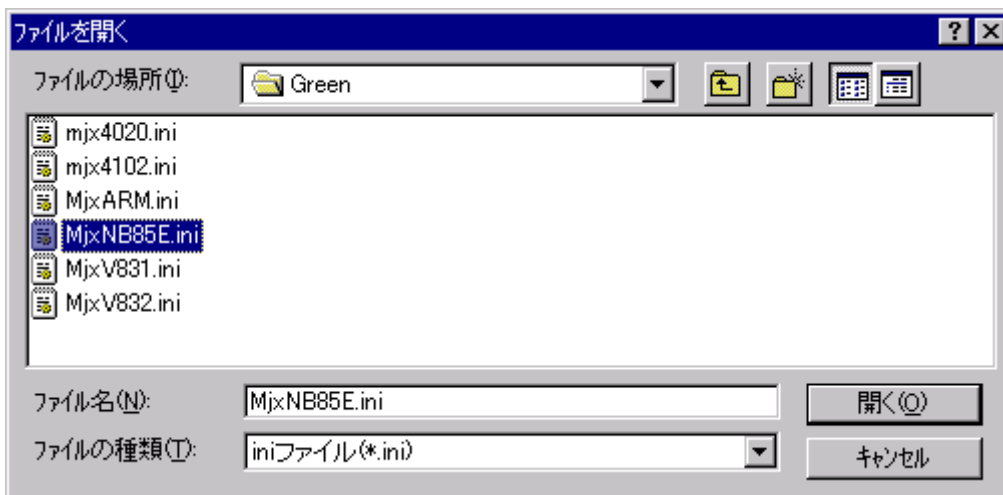
- MULTI.EXE は MJXSERV.EXE を子プロセスとして呼び出します。
- MJXSERV.EXE は MJXDBAPI.DLL を経由して MJXDEBW.EXE を子プロセスとして呼び出します。
- MJXDEBW.EXE は MDXSERV.EXE の子プロセスとしてだけでなく、単体でも動作できます。

第五章 MJX440 の環境設定

MJX440 を使用する前に必要な、環境設定の方法について記述しています。

MJX440 の環境設定は、コンフィグ支援ツール MJXCFG で行ないます。すべての機器の電源を投入した後、スタートメニューから MJXCFG を起動してください。手順は、次のとおりです。

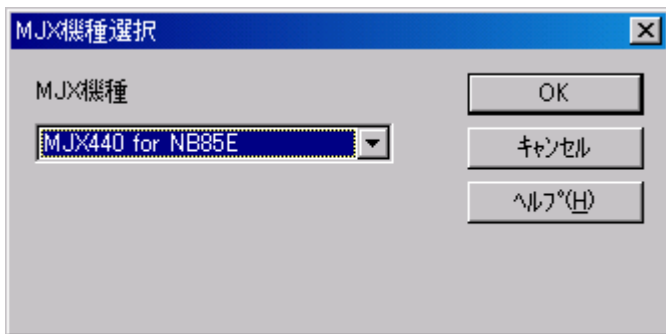
1. スタートメニュー
2. プログラム(P)
3. MJX Tools
4. MjxCfg Config Tool



テンプレートとして MJXNB85E.INI を利用できます。

新たにコンフィグレーションファイルを作成する場合は、新しいファイル名を指定します。

新規のコンフィグレーション ファイルを指定した場合は、次のダイアログ ボックスが表示されます。



MJX の機種から「MJX440 for NB85E」を選択し、**OK**ボタンを押してください。

コンフィグレーションファイル指定し、NB85E タブをクリックすると、次のダイアログ ボックスが表示されます。



ダイアログ ボックスが表示されたら、MJX440 やターゲット システムの環境に合わせてフィールドを設定し、**OK**ボタンを押してください。

コンフィグレーション設定項目の詳細は「MJX440/MJX330 MjxCfg User's Manual (Mjxcfg.PDF)」をご覧ください。

第六章 ソフトウェアの起動と終了

MJX440 を操作するソフトウェアの起動方法と終了方法について記述しています。

【注意】 ソフトウェアを起動する前には必ず「第五章 MJX440 の環境設定」を行ってください。

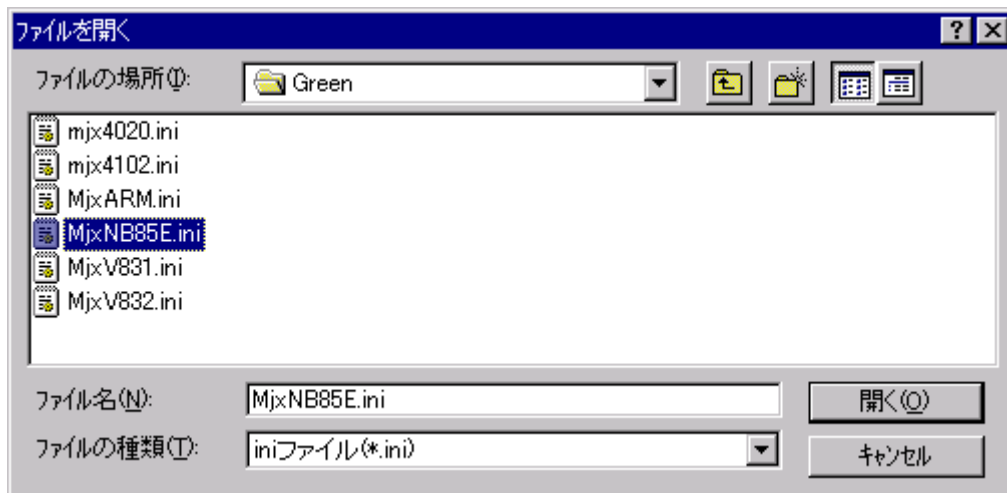
【注意】 MJX440 を操作するソフトウェアは、同時にひとつしか実行できません。

MULTI を使用する場合

MULTI を使って MJX440 を操作するためには、MULTI 本体を起動した後、次のコマンドで MJXSERV をリモート接続してください。*1

```
remote mjxserv
```

上記コマンドを入力すると、コンフィグレーション ファイルの入力をするための、ダイアログ ボックスが開きます。ここで、「[第五章 MJX440の環境設定](#)」で作成したターゲット システム用のコンフィグレーション ファイルを指定してください。*2

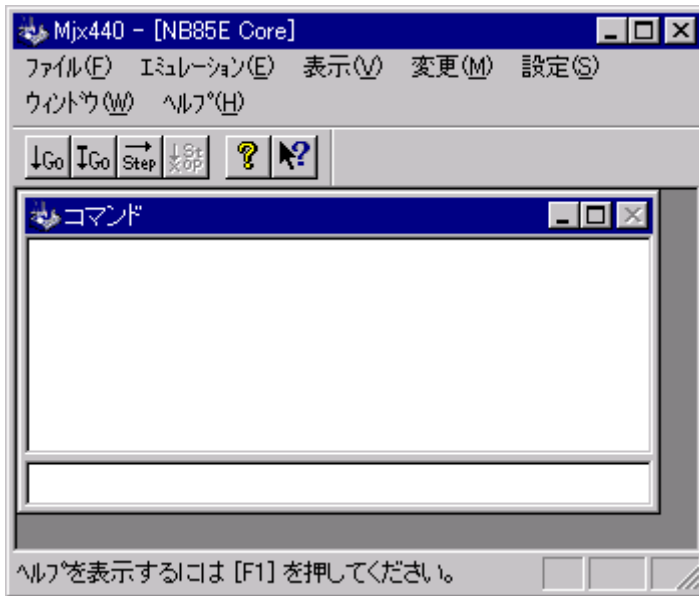


*1 ビルダから操作する場合は、サーバー名を mjxserv と指定し、**リモート** ボタンを押します。

*2 コンフィグレーション ファイルを指定するダイアログ ボックスで、ファイル名の指定に十数秒以上の時間がかかると、MULTI は「Server Message Timed Out, Terminate Connection?」というメッセージを表示します。このメッセージが表示された場合は、必ず「いいえ」のボタンを押してください。

正常にリモート接続できた場合は、MJXDEBW のウィンドウが開きます。

MJXDEBW のウィンドウ



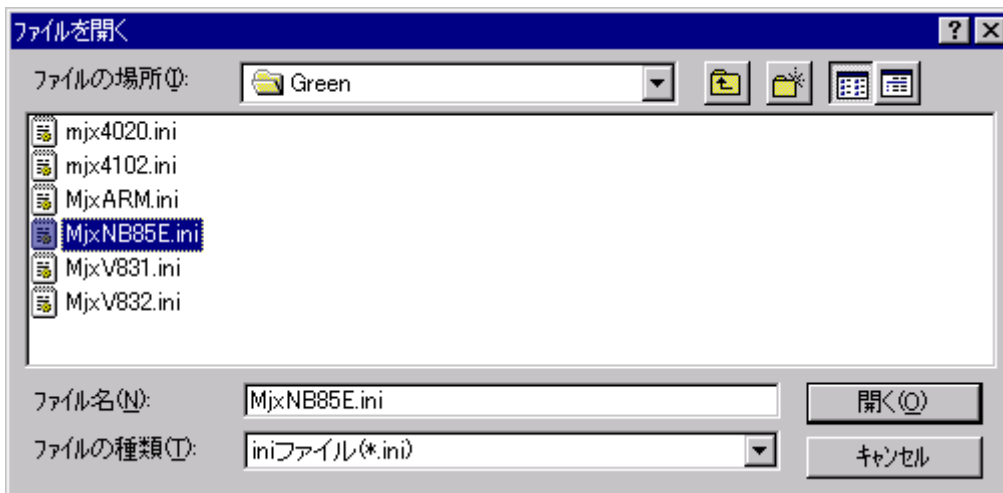
コンフィグレーションファイルは、MJXSERV の引数として指定することもできます。次のコマンドは、コンフィグレーション ファイルとして *MjxNB85E.ini* を指定します。

```
remote mjxserv MjxNB85E.ini
```


MJXDEBW を使用する場合

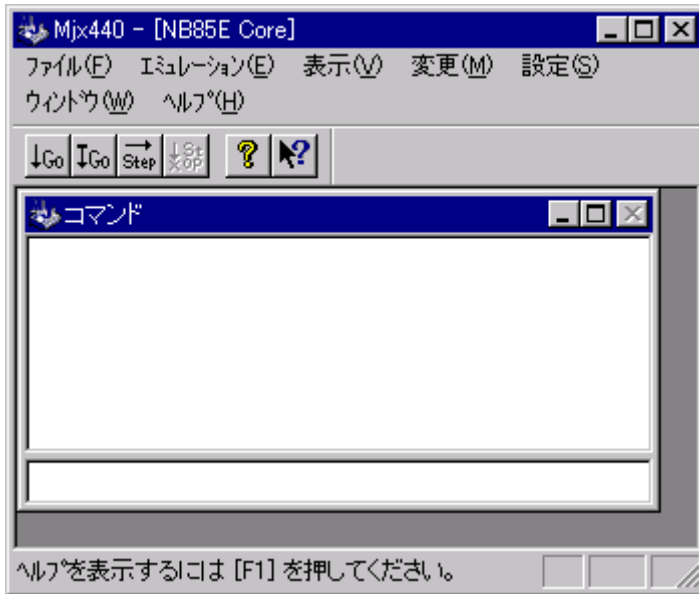
[MJXDEBW](#)を使ってMJX440 を操作するためには、スタート メニューから **MJXDEBW** を起動してください。
手順は、次のとおりです。

1. スタート メニュー
2. プログラム(P)
3. MJX440 Tools
4. MJXDEBW Debugger



MJXDEBW が起動されると、コンフィグレーション ファイルの入力をするための、ダイアログ ボックスが開きます。ここで、ターゲット システム用のコンフィグレーション ファイルを指定してください。

MJXDEBW が正常に起動できた場合は、MJXDEBW のウィンドウが開きます。



コンフィグレーション ファイルは、MJXDEBW の引数として指定することもできます。次のコマンドは、コンフィグレーション ファイルとして *MjxNB85E.ini* を指定します。

```
MJXDEBW MjxNB85E.ini
```

終了するためには、*q* コマンドを入力してください。

```
> q
```

第七章 MJXDEBW コマンド

この章ではデバッガ MJXDEBW のコマンドの使い方について記述しています。

MJXDEBW コマンドは、コマンド ウィンドウ内で実行するコマンドです。コマンド入力フィールドから入力し、コマンド応答フィールドへ結果を表示します。

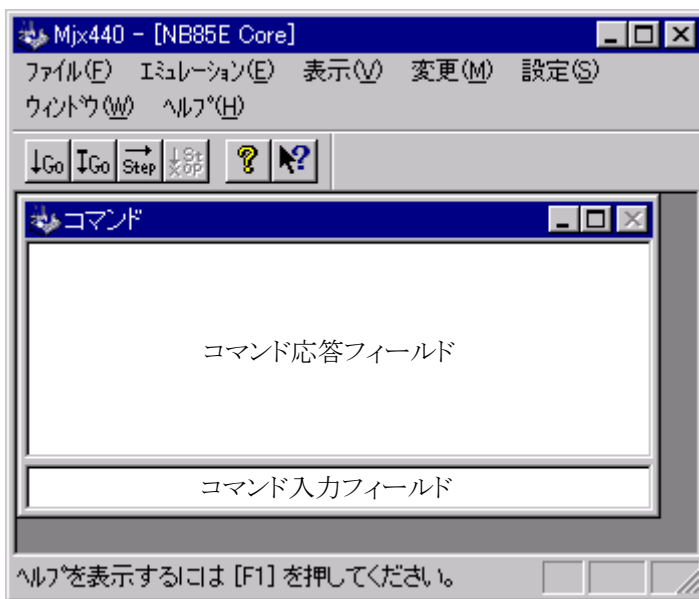


図 7-1 MJXDEBW 画面

- コマンド応答フィールドに表示できる行数は、「[表示\(V\) ~ オプション\(O\) ~ 表示\(V\)...](#)」メニューによって変更することができます。
- コマンド入力フィールドは履歴 バッファを持っています。以前に入力したコマンドは、上下のカーソル キーによって呼び出すことができます。保持するコマンドの行数は、「[表示\(V\) ~ オプション\(O\) ~ 表示\(V\)...](#)」メニューによって変更することができます。

MJXDEBW コマンドの一覧は、次のとおりです。

<u>ABORT</u>	ユーザ プログラムを強制停止します
<u>BATCH</u>	バッチ ファイルに記述された MJXDEBW コマンドを実行します
<u>BP</u>	ブレークポイントを表示/設定/解除します
<u>CLEAR</u>	コマンド応答フィールドをクリアします
<u>CONFIG</u>	MJX440 の環境を表示/変更します
<u>DUMP</u>	メモリの内容を表示します
<u>EVENT</u>	イベントを表示/設定します
<u>EXAMINE</u>	メモリの内容を変更します
<u>FILL</u>	メモリの内容をフィルします
<u>FLASH</u>	フラッシュ メモリの内容を書き換え、消去します*1
<u>GO</u>	ユーザ プログラムを実行します
<u>HISTORY</u>	リアルタイム トレース結果を表示します
<u>INIT</u>	MJX440 を再初期化します
<u>JOURNAL</u>	コマンド実行結果をファイルへ出力します
<u>LOAD</u>	ファイルをメモリへダウンロードします
<u>MOVE</u>	メモリの内容をブロック転送します
<u>OUTTRIG</u>	トリガ出力イベントを設定/解除します
<u>PIN</u>	ピンを有効/無効にします
<u>QUIT</u>	MJXDEBW を終了します
<u>REGISTER</u>	レジスタの内容を表示/変更します
<u>SETLOAD</u>	ブレークポイント、イベントおよびトレースの設定をファイルから読み出します
<u>SETSAVE</u>	ブレークポイント、イベントおよびトレースの設定をファイルへ保存します
<u>STEP</u>	ユーザ プログラムをステップ実行します
<u>TRACE</u>	リアルタイム トレースのモードを表示/設定します
<u>UNASM</u>	メモリの内容を逆アセンブル表示します
<u>VERSION</u>	ソフトウェアのバージョンを表示します
<u>WAIT</u>	ユーザ プログラムが停止するまで待ちます
<u>XPIN</u>	外部トリガ ケーブルの信号の状態を表示/設定します

*1 MJX440 for NB85E では、この機能は対応していません。

コマンド名の省略

コマンド名は、他のコマンド名と区別できる長さまで、省略することができます。

AB	ABORT と同じです
CO	CONFIG と同じです
CL	CLEAR と同じです
C	エラーです。CONFIG か CLEAR の区別ができません

コマンドライン表記のルール

コマンドラインは以下のルールに従って表記します。

[]	:省略可能な引数
斜体文字	:変数となる引数
標準文字	:固定の文字引数
{aa bb cc}	: で区切られた文字列を選択
[xxx...]	:xxx が繰り返し指定が可能であることを示す
△	:スペース

表示簡略化のため行末に連続する ,(カンマ)は省略可能です。

MULTIを使用する場合の注意

次のコマンドは、MULTI のターゲット ウィンドウ内では無視されます。

QUIT

次のコマンドは、できれば MULTI のターゲット ウィンドウ内では実行しないでください。

BP/S	命令ソフトウェア ブレークポイントを設定
GO	ユーザ プログラムの実行
STEP	ユーザ プログラムのステップ実行
REG <i>reg=val</i>	レジスタの変更

これらのコマンドはを実行した場合、それ以降、MULTIとMJXSERV間の整合性がとれなくなる危険性があります。ユーザ プログラムの実行制御やレジスタの書き換えは、MULTIのコマンドで行なうようにしてください。

ABORT

ユーザ プログラムを強制停止します

形式:

ABORT

実行中のユーザ プログラムを強制的に停止します。

引数:

例:

GO

(ユーザ プログラムを実行します)

ABORT

(ユーザ プログラムを強制停止します)

補足:

- 「第八章 [エミュレーション\(E\) ～ 中断\(B\)](#)」参照。

BATCH

バッチ ファイルに記述された MJXDEBW コマンドを実行します

形式:

`BATCH file` バッチ ファイル *file* を一行ずつ読み込み、MJXDEBW コマンドとして実行します。

引数:

file MJXDEBW コマンドが記述されたバッチ ファイル名 (テキスト ファイル)

例:

`BAT INIT.TXT` (INIT.TXT ファイルをバッチ ファイルとして実行します)

補足:

- BATCH コマンドは、4 レベルまでネスティングすることができます。
- バッチファイル中の「;」で始まる行は、コメントになります。
- バッチ ファイルの中の QUIT コマンドは、BATCH コマンドのみを終了し、MJXDEBW は終了しません。
- 「第八章 [ファイル\(F\) ~ バッチ\(B\)...](#)」参照。

BP

ブレークポイントを表示/設定/解除します

形式:

BP	[{H S}]	全ハードウェア(H)/ソフトウェア命令(S)ブレークポイントを表示します。
BP/C	{num}* A* W* R* H* S*	ブレークポイントを解除します。
BP/D	{num}S*	ブレークポイントを無効にします。
BP/E	{num}S*	ブレークポイントを有効にします。
BP/{A W R}	addr, [addrmask],[data],[datamask],[size],[asid]	RW アクセス/WRITE アクセス/READ アクセス ブレークポイントを設定します。
BP/H	addr,[addrmask],[asid]	ハードウェア ブレークポイントを設定します。
BP/V	eventname	イベント ブレークポイントを設定します。
BP/N	eventname	イベント ブレークポイントを解除します。
BP/S	addr	ソフトウェア ブレークポイントを設定します。

引数:

H	ハードウェア ブレークポイント指定
S	ソフトウェア ブレークポイント指定
C	ブレークポイント解除
D	ブレークポイント無効
E	ブレークポイント有効
A	アクセス ブレークポイント指定
W	WRITE アクセス ブレークポイント指定
R	READ アクセス ブレークポイント指定
num	ブレークポイント番号指定(10進数)
*	すべてブレークポイント指定
A*	すべての RW アクセス ブレークポイント指定
W*	すべての WRITE アクセス ブレークポイント指定
R*	すべての READ アクセス ブレークポイント指定
H*	すべてのハードウェア ブレークポイント指定
S*	すべてのソフトウェア ブレークポイント指定
addr	アクセスブレークポイント アドレス指定 (16 進数)
addrmask	アクセスブレークポイント アドレスマスク指定 (16 進数)
data	アクセスデータ条件 (省略時はデータ条件の比較なし)

第七章 MJXDEBWコマンド

<i>datamask</i>	アクセスデータ条件のマスク（省略時はマスクなし）
<i>size</i>	データサイズ {ANY BYTE HALF WORD}から選択。（選択2文字目以降は省略可）
<i>asid</i>	設定アドレスの ASID 条件（省略時は ASID 条件の比較なし）
<i>eventname</i>	EVENT コマンドで設定したイベント識別名

例:

BP	(ブレイクポイントを表示します)
BP/C *	(すべてのブレイクポイントを解除します)
BP/D S*	(全ソフトウェア ブレイクポイントを無効にします)
BP/E S*	(全ソフトウェア ブレイクポイントを有効にします)
BP/A 1000	(0x1000 番地のメモリ空間にRWアクセス ブレイクポイントを設定します)
BP/H 2000	(0x2000 番地にハードウェア ブレイクポイントを設定します)
BP/S 3000	(0x3000 番地にソフトウェア ブレイクポイントを設定します)
BP/V abc	(識別名 abc のイベント ブレイクポイントを設定します)
BP/N abc	(識別名 abc のイベント ブレイクポイントを解除します)
BP/S 1000	(現在とステートと同じメモリ領域のアドレス 0x1000 番地に命令ソフトウェア ブレイクポイントを設定します)

補足:

- アクセス/ハードウェア ブレイクポイントは、最大 2 個まで設定できます。
- ソフトウェア ブレイクポイントは、命令上に最大 128 個まで設定できます。
- ミスアライメントアクセスに対してはデータ条件は無効になります。
- データサイズが ANY の場合、データ条件は無効になります。
- データ条件を省略した場合、データマスク条件は無効になります。
- ソフトウェア ブレイクポイントを設定すると、指定したアドレスの命令を実行する直前にユーザプログラムを停止することができます。
- ソフトウェア ブレイクポイントは、RAM 領域に設定することができます。
- 「第八章 [設定\(S\) ~ ブレイクポイント\(B\)...](#)」

CLEAR

コマンド応答フィールドをクリアします

形式:

CLEAR

コマンド応答フィールドの表示をすべて消去します。

引数:

例:

CLEAR

補足:

- 「第八章 [表示\(V\) ~ コマンド応答クリア\(C\)](#)」参照。

CONFIG

MJXDEBW の環境を表示/変更します

形式:

CONFIG 現在設定されている変更可能な MJXDEBW の環境設定を表示します。

CONFIG/S *item=value* MJXDEBW の環境設定項目 *item* を *value* に設定します。

引数:

item 環境設定項目、次のいずれかを指定

JCLOCK	JTAG クロック
TOPADDR	ROM の先頭アドレス
TRCCLK	トレースクロック分周比
DBINT	EXTIN3 ピンの機能選択

value 環境設定項目に設定する値、

25 または 12.5	JCLOCK の場合
16 進数アドレス	TOPADDR の場合
1 または 2	TRCCLK の場合
DIS,LEAD または TRAIL	DBINT の場合
DIS	EXTIN3 ピンを強制停止信号として使用しない
LEAD	EXTIN3 ピンの立ち上がりを強制停止信号として使用する
TRAIL	EXTIN3 ピンの立ち下がりを強制停止信号として使用する

例:

CONFIG (MJXDEBW の環境を表示します)

CONFIG/S JCLOCK=12.5 (JTAG クロックを 12.5MHz に設定します)

CONFIG/S TOPADDR=20000 (ROM の先頭アドレスを 0x20000 に設定します)

CONFIG/S TRCCLK =1 (トレースクロックを 1 分周に指定します)

CONFIG/S DBINT = DIS (EXTIN3 ピンを強制停止信号として使用しません)

補足:

- トレースクロックは CPU クロック1分周または2分周の選択ができます。MJX440の最大トレース周波数(66MHz)を超えない値を設定してください。

DUMP

メモリの内容を表示します

形式:

DUMP[/{B|W|L}] [*addr1*, [*addr2*]]

アドレス *addr1* から *addr2* までのメモリの内容を表示します。

引数:

/B 8ビット指定

/W 16ビット指定

/L 32ビット指定

addr1 メモリ表示開始アドレス (16進数)

addr2 メモリ表示終了アドレス (16進数)

例:

DUMP/B 1000 (0x1000番地から64バイトのメモリを8ビットで表示します)

DUMP/W 2000,20FF (0x2000から0x20FF番地のメモリを16ビットで表示します)

DUMP (前回のDUMPコマンドの続きを表示します)

補足:

- *addr2*省略時は、64バイト分のメモリを表示します。
- *addr1*省略時は、前回のDUMPコマンドの続きを表示します。
- ビット指定省略時は、前回のサイズが適用されます。
- 「第八章 [表示\(V\) ~ メモリ\(M\)...](#)」参照。
- 「付録L [V850/ME2内蔵命令RAMの取り扱い](#)」参照

EVENT

イベント機能構成と設定されている全イベントを表示／設定します

形式:

- EVENT イベント機能構成と設定されている全イベントを表示します。
- EVENT/A IP, *eventname*, *addr1* [, *asid*]
 命令アドレスイベントを設定します。
- EVENT/A IR, *eventname*, *Baddr1*, *Eaddr1* [, *asid*]
 命令アドレス範囲イベントを設定します。
- EVENT/A {AP|WP|RP}, *eventname*, *addr2r*, [*data*], [*datamask*], [*size*] [, *asid*]
 RW アクセス(AP)、WRITE アクセス(WP)、READ アクセス(RP)アドレス イベントを設定します。
- EVENT/A {AR|WR|RR}, *eventname*, *Baddr2*, *Eaddr2*, [*data*], [*datamask*], [*size*] [, *asid*]
 RW アクセス範囲(AR)、WRITE アクセス範囲(WR)、READ アクセス範囲(RR)アドレス イベントを設定します。
- EVENT/D *eventname* イベントを抹消します。
- EVENT/Q *eventname*, *step* [, *count*]
 シーケンシャルイベントを設定します。
 パスカウントを設定できます。
- EVENT/S {1|2|3|4|C}, *eventname*
 シーケンス器 スイッチ イベント(SW1,SW2,SW3,SW4,クリア)へイベントを割り当てます。
- EVENT/R {1|2|3|4|C}, *eventname*
 シーケンス器 スイッチ イベント(SW1,SW2,SW3,SW4,クリア)イベント割当を開放します。

引数:

<i>eventname</i>	イベント識別子
<i>addr1</i>	命令アドレス イベント設定アドレス (16 進数)
<i>Baddr1</i>	命令アドレス範囲イベント開始設定アドレス(16 進数)
<i>Eaddr1</i>	命令アドレス範囲イベント終了設定アドレス(16 進数)
<i>addr2</i>	アクセス ブレークポイント設定アドレス (16 進数)
<i>Baddr2</i>	アクセス アドレス範囲イベント開始設定アドレス(16 進数)
<i>Eaddr2</i>	アクセス アドレス範囲イベント終了設定アドレス(16 進数)

第七章 MJXDEBWコマンド

<i>data</i>	アクセス データ条件 (省略時はデータ条件の比較なし)
<i>datamask</i>	アクセス データ条件マスク (省略時は全ビットマスクする。データ条件省略時は無効)
<i>size</i>	データサイズを以下の中から選択 (2文字目以降は省略可) ANY BYTE HALF WORD
<i>asid</i>	設定アドレス範囲の ASID 条件 (省略時は ASID 条件の比較なし)
<i>step</i>	シーケンス器段数 (1~4)
<i>count</i>	パスカウント (省略時は 1)

例:

EVENT/A	IP,abc,1000	(0x1000 番地に命令アドレスイベントをイベント名 abc で登録します)
EVENT/A	IR,bcd,1000,3000	(0x1000 番地から 0x3000 番地に命令アドレス範囲イベントをイベント名 bcd で登録します)
EVENT/A	RP,cde,1000	(0x1000 番地をアクセスブレイクポイントとし、READ アクセスアドレスイベントをイベント名 cde で登録します)
EVENT/D	abc	(イベント名 abc を抹消します)
EVENT/Q	def	(シーケンシャル イベントをイベント名 def として設定します)
EVENT/S	1,abc	(イベント名 abc をシーケンス器スイッチイベント(SW1)に割り当てます)
EVENT/R	1,abc	(イベント名 abc をシーケンス器スイッチイベント(SW1)から割当を開放します)

補足:

- ミスアライメントアクセスに対してはデータ条件は無効になります。
- データサイズが ANY の場合、データ条件は無効になります。
- データ条件を省略した場合、データマスク条件は無効になります。
- イベントを抹消する場合、チャンネル割当が行われている場合は、チャンネルが開放されません。シーケンシャル イベントの場合は、シーケンシャル器が無効になります。

- シーケンシャル イベントの設定は、「EVENT/A」コマンドでシーケンス器スイッチイベント割り当てを行わなければなりません。

シーケンス器 スイッチ イベントの割り当ては、

シーケンス器段数が1段のときは SW4 が有効になります。

シーケンス器段数が2段のときは SW4,SW3 が有効になります。

シーケンス器段数が3段のときは SW4,SW3,SW2 が有効になります。

シーケンス器段数が4段のときは SW4,SW3,SW2,SW1 が有効になります。

このコマンドでは実際にイベントチャンネルの割り当てが行われます。

- イベントは、有効／無効あわせて最大 128 個のイベントを設定できます。
- 関連コマンド [TRACE](#), [BP](#), [OUTTRIG](#)
- 「第八章 [設定\(S\) ~ イベント\(E\)...](#)」参照。

1 イベント識別名のルール

イベント識別名はカンマ、スペース、タブを除く印字可能半角文字から構成します。
最大 32 文字まで指定できます。

2 イベントチャンネルの占有:

イベントが設定できるハードウェア(イベントチャンネル)は有限です。

「EVENT/S [IP|AP|WP|RP|IR|AR|WR|RR]」コマンドでは各イベントはハードウェアのイベントチャンネルに対して割り当ては行われません。

BP/V

EVENT/S

TRACE/I [BE|EE|ME|IE]

OUTTRIG/S

コマンドが発行された時点でハードウェアのイベントチャンネルに対して割り当てが行われません。イベントチャンネルは命令系とアクセス系に分かれており、各チャンネル数は

EVENT イベント機能構成表示コマンドで知ることができます。

範囲イベントはイベントチャンネルを2チャンネル占有します。

EXAMINE

メモリの内容を変更します

形式:

EXAMINE/{B|W|L} *addr*=*data1*[,*data2*...]

アドレス *addr* のメモリをデータ *data1*[,*data2*...]に変更します。

引数:

<i>/B</i>	8ビット指定
<i>/W</i>	16ビット指定
<i>/L</i>	32ビット指定
<i>addr</i>	メモリ変更開始アドレス (16進数)
<i>data1, data2</i>	メモリ変更データ (16進数)

例:

EXAMINE/B 1000=55	(0x1000 番地のメモリを 8 ビットデータ 0x55 に変更します)
EXAMINE/W 3000=1,2,3	(0x3000 番地からのメモリを 16 ビットデータ 0x0001、0x0002、0x0003 に変更します。)

補足:

- 「第八章 [表示\(V\) ~ メモリ\(M\)...](#)」参照。
- 「付録 L [V850/ME2 内蔵命令 RAM の取り扱い](#)」参照。

FILL

メモリの内容を変更します

形式:

`FILL/{B|W|L} addr1,addr2,data`

アドレス *addr1* から *addr2* までのメモリを、データ *data* でフィルします。

引数:

`/B` 8ビット指定
`/W` 16ビット指定
`/L` 32ビット指定
addr1 メモリ フィル開始アドレス (16進数)
addr2 メモリ フィル終了アドレス (16進数)
data フィル データ (16進数)

例:

`FILL/B 0,3FF,FF` (0x0 から 0x3FF 番地のメモリを 8ビットデータ 0xFF でフィルします)
`FILL/W 1000,1FFF,0` (0x1000 から 0x1FFF 番地のメモリを 16ビットデータ 0x0000 でフィルします)

補足:

- 8M バイト以上のフィルはできません。(制限事項)
- 「第八章 [変更\(M\) ~ フィル\(F\)...](#)」参照。
- 「付録 L [V850/ME2 内蔵命令 RAM の取り扱い](#)」参照。

FLASH

フラッシュ メモリの内容を書き換え、消去します

形式:

FLASH[/WRITE|/UPDATE] *file_name* [, [*device_type*, *access_bus_width*]]

フラッシュ メモリの内容を書き換えます。

FLASH[/ERASEONE|/ERASEALL] *addr* [, [*device_type*, *access_bus_width*]]

フラッシュ メモリの内容を消去します。

/WRITE フラッシュ メモリにデータを書き込みます (書き込み)

/UPDATE フラッシュ メモリのデータを更新します (セクタ単位で消去後、書き込み)

/ERASEONE フラッシュ メモリの内容をセクタ単位に消去します

/ERASEALL フラッシュ メモリの内容を全て(チップ単位に)消去します

引数:

file_name ダウンロードするファイル名 (書き込みデータ)

addr 消去アドレス (16 進数で指定。絶対アドレスです。先頭からのオフセットではありません)

device_type フラッシュ デバイスを (省略時は、コンフィグレーション設定値を使用)

access_bus_width アクセス バス幅 (省略時は、コンフィグレーション設定値を使用)

1 1 バイト アクセス

2 2 バイト アクセス

4 4 バイト アクセス

例:

FLASH/WRITE load.hex, 2 (load.hex ファイルの内容をコンフィグレーション設定値で指定されたフラッシュ デバイスへ 2 バイト単位で書き込む)

FLASH/ERASEONE BFC00040, AM29F080B

(AM29F080B の BFC00040 (16 進) のアドレスに該当するセクタを消去)

補足:

- オプション省略時は、/WRITE を既定値とします。
- ファイル形式 (MJX バイナリ、S レコード、インテル HEX、COFF 形式) は自動認識されます。
- {MjxDebw.exe の格納ディレクトリ}\zax_mjx\flash\device の下に{device_type}.dat ファイルを作成して、デバイス情報定義ファイルが存在する必要があります。
ファイル レイアウトは、「[付録 J フラッシュ メモリ デバイス情報定義ファイル レイアウト](#)」を参照してください。
- ソフトウェア ブレークポイントは全てクリアされます。
- ロックビットのセット・リセットの機能があるフラッシュ デバイスでは、ロックビットがクリアされている必要があります。
- 本機能に対応するフラッシュ メモリの一覧は、「[付録 K 対応フラッシュ メモリ一覧](#)」にあります。
- 本機能はターゲット ボード上の RAM に書き込みプログラム本体、フラッシュ メモリ デバイス情報、更新データ (セクタ単位) をロードして実行されます。
実行に必要なメモリ容量は、18 K バイト + (フラッシュ デバイスの全セクタ数 x 8 バイト) + 更新されるセクタ (複数あるときはそのうちの最大) サイズです。
本機能を使用する場合、事前にRAMを使用可能にする必要があります。(手順はCPU 及びターゲットボードに依存します。)
- 本機能を使用する場合、フラッシュ領域に対するキャッシュ設定を無効にする必要があります。
- 「[第八章 MJXDEBW メニューコマンド](#)」参照。

GO

ユーザ プログラムを実行します

形式:

GO [*addr*] アドレス *addr* からユーザ プログラムを実行します。

引数:

addr ユーザ プログラム開始アドレス (16 進数)

例:

GO 1000 (0x1000 番地からユーザ プログラムを実行します)

GO (現在の PC からユーザ プログラムを実行します)

補足:

- ユーザ プログラムがブレークポイントなどで停止しない場合、WAIT あるいは ABORT コマンドで強制終了させます。(タイムアウトはありません。)
- MULTI のターゲット ウィンドウ内では、実行しないでください。
- *addr* 省略時は、現在の PC から実行します。
- 「第八章 [エミュレーション\(E\) ~ 実行\(G\)...](#)」参照。
- 「第八章 [エミュレーション\(E\) ~ 再スタート\(R\)](#)」参照。

HISTORY

リアルタイム トレース結果を表示します

形式:

- HISTORY** リアルタイム トレース結果が格納されている、パケット番号の範囲を表示します。
- HISTORY/P** [*start*[,*end*]] パケット番号 *start* から *end* までのリアルタイム トレース結果を、パケット単位で表示します。
- HISTORY/D** [*start*[,*end*]] パケット番号 *start* から *end* までのリアルタイム トレース結果を、逆アセンブルで表示します。

引数:

- start* 表示開始パケット番号 (10 進数)
- end* 表示終了パケット番号 (10 進数)

例:

- HISTORY** (リアルタイム トレース結果が格納されている、パケット番号の範囲を表示します)
- HISTORY/P** -10,0 (パケット番号-10から0までのリアルタイム トレース結果を、パケット単位で表示します)
- HISTORY/D** 0,70 (パケット番号0から70までのリアルタイム トレース結果を、逆アセンブルで表示します)

補足:

- *end*省略時は、16 パケット分を表示します。
 - *start*省略時は、前回の **HISTORY** コマンドの続きを表示します。
 - *start*と *end*は、パケット番号 0 を起点とするオフセットで指定します。
トレース条件により、パケット番号 0 の点は、次のように異なります
- | | |
|------------------|------------|
| ビギン モニタ | プログラム開始点 |
| エンド モニタ | プログラム終了点 |
| ビギン トリガ/ビギンイベント | トレース トリガ点 |
| エンド トリガ/エンドイベント | トレース トリガ点 |
| ミッド トリガ/ミッドイベント | トレース トリガ点 |
| インナートリガ/インナーイベント | トレース開始トリガ点 |
- 「第八章 [表示\(V\) ~ トレース結果\(T\)...](#)」参照。

INIT

MJX を再初期化します

形式:

INIT

MJX を再初期化します。

補足:

- ターゲットシステムの CPU もリセットされます。
- 「第八章 [エミュレーション\(E\) ~ 初期化\(I\)](#)」参照。

JOURNAL

コマンド実行結果をファイルへ出力します

形式:

JOURNAL[/{A|W}] *file*[, *mode*[, *echo*]]

コマンド実行結果をファイル *file* へ出力するようにします。

JOURNAL/E

ファイルへの出力を終了し、ファイルをクローズします。

引数:

<i>/A</i>	追記出力指定
<i>/W</i>	新規出力指定 (省略時)
<i>file</i>	出力ファイル名指定
<i>mode</i>	出力モード指定
IN	コマンドのみ出力
OUT	コマンド実行結果のみ出力
ALL (省略時)	コマンドとコマンド実行結果を出力
<i>echo</i>	エコー モード指定
OFF	ファイルへ出力する内容は画面に表示しない
ON (省略時)	ファイルへ出力する内容を画面にも表示する

例:

JOURNAL TEST.TXT	(コマンド実行結果を TEST.TXT ファイルへ出力するようにします)
JOURNAL/E	(ファイルへの出力を終了し、ファイルをクローズします)
JOURNAL/A TEST.TXT	(コマンド実行結果をファイル TEST.TXT へ追記出力するようにします)

補足:

- */W* を指定し、かつ同名のファイルが存在する場合、そのファイルの内容は失われます。
- 「第八章 [ファイル\(F\) ~ ジャーナル開始\(S\)...](#)」参照。
- 「第八章 [ファイル\(F\) ~ ジャーナル停止\(E\)](#)」参照。

LOAD

ファイルをメモリへダウンロードします

形式:

LOAD *file* [, *offset*] MJX バイナリ*¹、S レコード、インテルヘキサ、または COFF 形式のファイル *file* をメモリへダウンロードします。

LOAD/R *file* [, *offset*] MJX バイナリ ファイル *file* をエミュレーション メモリ経由でメモリへダウンロードします。より高速にダウンロードを行うことができます。

引数:

file ダウンロードするファイル名
offset オフセット アドレス (省略時 0)

例:

LOAD PROG1.ABS (ファイル PROG1.ABS をメモリへダウンロードします)

LOAD PROG1.ABS,2000 (ファイル PROG1.ABS をメモリへダウンロードします)

LOAD/R PROG2.MJX (ファイル PROG2.MJX をエミュレーション メモリ経由でメモリへダウンロードします)

補足:

- ファイル形式は、自動認識されます。
- ファイル名の拡張子を省略した場合は、.mjx が補完されます。
- LOAD/R コマンド時、MJX バイナリ ファイルの中に、エミュレーション メモリを指すレコードがある場合、正しくダウンロードできません。
- /R 指定でダウンロードを行った場合エミュレーションメモリの内容は破壊されます。
- 「第八章 [エミュレーション\(E\) ~ ダウンロード\(L\)...](#)」参照。
- 「付録 L [V850/ME2 内蔵命令 RAM の取り扱い](#)」参照。

*1 MJX バイナリ ファイルについては「第八章 高速ダウンロード」を参照してください。

MOVE

メモリの内容をブロック転送します

形式:

MOVE[/{B|W|L}] *addr1*,*addr2*,*addr3*

アドレス *addr1* から *addr2* までのメモリを *addr3* へブロック転送します。

引数:

/B 8ビット指定
 /W 16ビット指定
 /L 32ビット指定
addr1 転送元メモリ開始アドレス (16進数)
addr2 転送元メモリ終了アドレス (16進数)
addr3 転送先メモリ アドレス (16進数)

例:

MOVE 1000,10FF,2000 (0x1000 から 0x10FF 番地のメモリを 0x2000 番地へブロック転送します)

補足:

- 8M バイト以上のブロック転送はできません。(制限事項)
- サイズ省略時は 8 ビット設定が選択されます。
- 転送元領域と転送先領域が重なっている場合の結果は不定です。
- 「第八章 [変更\(M\) ~ 移動\(M\)...](#)」参照。

OUTTRIG

トリガ出力イベントを設定／解除します

形式:

OUTTRIG/S *eventname* トリガ出力イベントを設定します。
OUTTRIG/R *eventname* トリガ出力イベントを解除します。

引数:

eventname イベント識別名

例:

OUTTRIG/S abc (イベント abc をトリガ出力イベントとして設定します)
OUTTRIG/R abc (イベント abc をトリガ出力イベントから解除します)

補足:

- トリガ出力のイベントは、ターゲットCPUにイベントトリガ出力機能が実装されている場合のみ有効です。
- 「第八章 [設定\(S\) ~ イベント\(E\)...](#)」参照。

QUIT

MJXDEBW を終了します

形式:

QUIT MJXDEBW を終了します。

補足:

- バッチ ファイルの中の QUIT コマンドは、BATCH コマンドのみを終了し、MJXDEBW は終了しません。

SETLOAD

ブレイクポイント、イベントおよびトレースの設定をファイルから読み出します

形式:

SETLOAD *filename* ブレイクポイント、イベントおよびトレースの設定をファイルから読み出します。

引数:

filename 読み出すファイル名

例:

SETLOAD abc.mjs (abc.mjs ファイルを読み出します)

補足:

- 「第八章 [設定\(S\) ～ 設定の読出\(L\)...](#)」参照。

SETSAVE

ブレイクポイント、イベントおよびトレースの設定をファイルから保存します

形式:

SETSAVE *filename* ブレイクポイント、イベントおよびトレースの設定をファイルへ
保存します。

引数:

filename 保存するファイル名

例:

SETSAVE abc.mjs (abc.mjs ファイルを保存します)

補足:

- 同じ名前のファイルが存在する場合、上書きします。
- 「第八章 [設定\(S\) ～ 設定の保存\(S\)...](#)」参照。

STEP

ユーザ プログラムをステップ実行します

形式:

STEP [*num*] ユーザ プログラムを *num* 回ステップ実行します

引数:

num ステップ実行回数指定 (10 進数、省略時は 1)

例:

STEP (ユーザ プログラムを 1 回ステップ実行します。)

STEP 10 (ユーザ プログラムを 10 回ステップ実行します。)

補足:

- MULTI を使用している場合は、レジスタを変更しないでください。
- 「第八章 [エミュレーション\(E\) ~ ステップ\(S\)](#)」参照。
- 「第八章 [エミュレーション\(E\) ~ N-ステップ\(N\)...](#)」参照。

TRACE

リアルタイム トレースの全トレース条件を表示します

形式:

TRACE トレース条件をすべて表示します。

引数:

例:

TRACE (トレース条件をすべて表示します)

補足:

- 「第八章 [設定\(S\) ~ トレース\(T\)...](#)」参照。

TRACE/A

RW/WRITE/READ データ トレースポイントを設定します

形式:

TRACE/A {A|W|R}, *addr*[, *addrmask* [, *asid*]]

RW データ(A),WRITE データ(W),READ(R)データ トレースポイントを設定します。

引数:

<i>addr</i>	データ トレースポイント設定アドレス(16 進数)
<i>addrmask</i>	データ トレースポイント設定アドレスマスク (16 進数。省略時はマスクなし)
<i>asid</i>	設定アドレスの ASID 条件(省略時は ASID 条件の比較なし)

例:

TRACE/A W,1000 (0x1000 番地への WRITE アクセスをデータトレースポイントとして設定します)

補足:

- 命令トレース停止時にはデータトレースはエンドモニタモードで実行します。データ トレースポイントは最大 2 点まで設定できます。
- *addr* の有効範囲は A27-A2 です。
- *addrmask* の有効範囲は A9-A2 です。
- 「第八章 [設定\(S\) ~ トレース\(T\)...](#)」参照。

TRACE/C

データ トレースポイントを解除します

形式:

TRACE/C *addr* データ トレースを解除します。
TRACE/C * すべてのデータ トレースを解除します。

引数:

addr データ トレースポイント設定アドレス(16 進数)
* すべてのデータ トレースポイント

例:

TRACE/C * (すべてのデータ トレースポイントを解除します。)

補足:

- 「第八章 [設定\(S\) ~ トレース\(T\)...](#)」参照。

TRACE/M

トレースモードおよびトレース要因を設定します

形式:

TRACE/M *mode*[, *mask*] トレースモードおよびトレース要因の設定をします

引数:

<i>mode</i>	<p>トレースモードを指定します。</p> <p>RT: 通常モード</p> <p>FL: 完全モード</p>
<i>mask</i>	<p>トレースするトレース要因をビット毎に16進数で指定します。</p> <p>800000:CTRET 命令(分岐元 PC)</p> <p>000800:CTRET 命令(分岐先 PC)</p> <p>400000:DISPOSE 命令(分岐元 PC)</p> <p>000400:DISPOSE 命令(分岐先 PC)</p> <p>200000:SWITCH 命令(分岐元 PC)</p> <p>000200:SWITCH 命令(分岐先 PC)</p> <p>100000:CALLT 命令(分岐元 PC)</p> <p>000100:CALLT 命令(分岐先 PC)</p> <p>080000:条件分岐命令(条件成立)による分岐(分岐元 PC)</p> <p>000080:条件分岐命令(条件成立)による分岐(分岐先 PC)</p> <p>040000:条件分岐命令(条件不成立)による分岐(分岐元 PC)</p> <p>000040:条件分岐命令(条件不成立)による分岐(分岐先 PC)</p> <p>020000:JARL 命令(分岐元 PC)</p> <p>000020:JARL 命令(分岐先 PC)</p> <p>010000:JR 命令(分岐元 PC)</p> <p>000010:JR 命令(分岐先 PC)</p> <p>008000:JMP 命令(分岐元 PC)</p> <p>000008:JMP 命令(分岐先 PC)</p> <p>004000:RETI 命令(分岐元 PC)</p> <p>000004:RETI 命令(分岐先 PC)</p> <p>002000:例外発生(分岐元 PC)</p> <p>000002:例外発生(分岐先 PC)</p>

第七章 MJXDEBWコマンド

001000:割り込み発生(分岐元 PC)
000001:割り込み発生(分岐先 PC)
省略時は標準設定(下記参照)になります。

<標準設定>

000800:CTRET 命令(分岐先 PC)
000400:DISPOSE 命令(分岐先 PC)
000200:SWITCH 命令(分岐先 PC)
000100:CALLT 命令(分岐先 PC)
080000:条件分岐命令(条件成立)による分岐(分岐元 PC)
040000:条件分岐命令(条件不成立)による分岐(分岐元 PC)
000008:JMP 命令(分岐先 PC)
000004:RETI 命令(分岐先 PC)
002000:例外発生(分岐元 PC)
000002:例外発生(分岐先 PC)
001000:割り込み発生(分岐元 PC)
000001:割り込み発生(分岐先 PC)

例:

TRACE/M RT, 008000 (トレース モードを通常モードに、トレース要因を JMP 命令(分岐元PC)に設定します)

補足:

- 完全トレース モードでは、データ トレースポイントは無効になります。
- 標準設定で設定されるトレース要因が1つでも欠けると逆アセンブル モードでのトレース結果表示はできません。
- トレース要因を増やすとリアルタイムモード時、オーバーフローが発生しやすくなりますが、逆アセンブラ表示時のオーバーフロー発生による実行アドレス喪失からの復帰が早くなります。
- 「第八章 [設定\(S\) ~ トレース\(T\)...](#)」参照。

TRACE / I

実行トレース条件を設定します

形式:

TRACE / I	BM, <i>length</i>	実行トレース条件をビギン モニタに設定します。
TRACE / I	EM	実行トレース条件をエンド モニタに設定します。
TRACE / I	BT, <i>addr</i> [, <i>asid</i>]	実行トレース条件をビギン トリガに設定します。
TRACE / I	ET, <i>addr</i> [, <i>asid</i>]	実行トレース条件をエンド トリガに設定します。
TRACE / I	MT, <i>addr</i> , <i>length</i> [, <i>asid</i>]	実行トレース条件をミッド トリガに設定します。
TRACE / I	IT, <i>Saddr</i> , <i>Eaddr</i> [, [<i>Sasid</i>], [<i>Easid</i>]]	実行トレース条件をインナートリガに設定します。
TRACE / I	BE, <i>eventname1</i> [, <i>eventname2</i> ...]	実行トレース条件をビギン イベントに設定します。
TRACE / I	EE, <i>eventname1</i> [, <i>eventname2</i> ...]	実行トレース条件をエンド イベントに設定します。
TRACE / I	ME, <i>length</i> , <i>eventname1</i> [, <i>eventname2</i> ...]	実行トレース条件をミッド イベントに設定します。
TRACE / I	IE, <i>Seventname</i> , <i>Eventname</i>	実行トレース条件をインナー イベントに設定します。
TRACE / I	DI	実行トレース条件を中止します。

引数:

<i>addr</i>	トリガ アドレスを指定
<i>asid</i>	トリガ アドレスの ASID 条件 (省略時は ASID 条件なし)
<i>Saddr</i>	インナートリガ条件のトレース開始アドレスを指定
<i>Eaddr</i>	インナートリガ条件のトレース中断アドレスを指定
<i>Sasid</i>	インナートリガ条件のトレース開始アドレスの ASID 条件を指定 (省略時は ASID 条件の比較なし)
<i>Easid</i>	インナートリガ条件のトレース中断アドレスの ASID 条件を指定 (省略時は <i>Sasid</i> と同じ)
<i>eventnameN</i>	トリガ イベントを指定
<i>SeventnameN</i>	インナー イベント条件のトレース開始イベントを指定
<i>EventnameN</i>	インナー イベント条件のトレース中断イベントを指定
<i>length</i>	トレース ワード長を指定 (省略時は最大 131072 ワード)

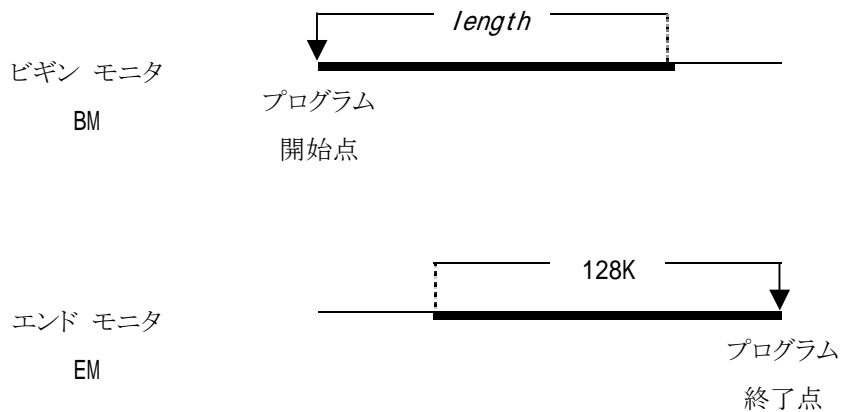
例:

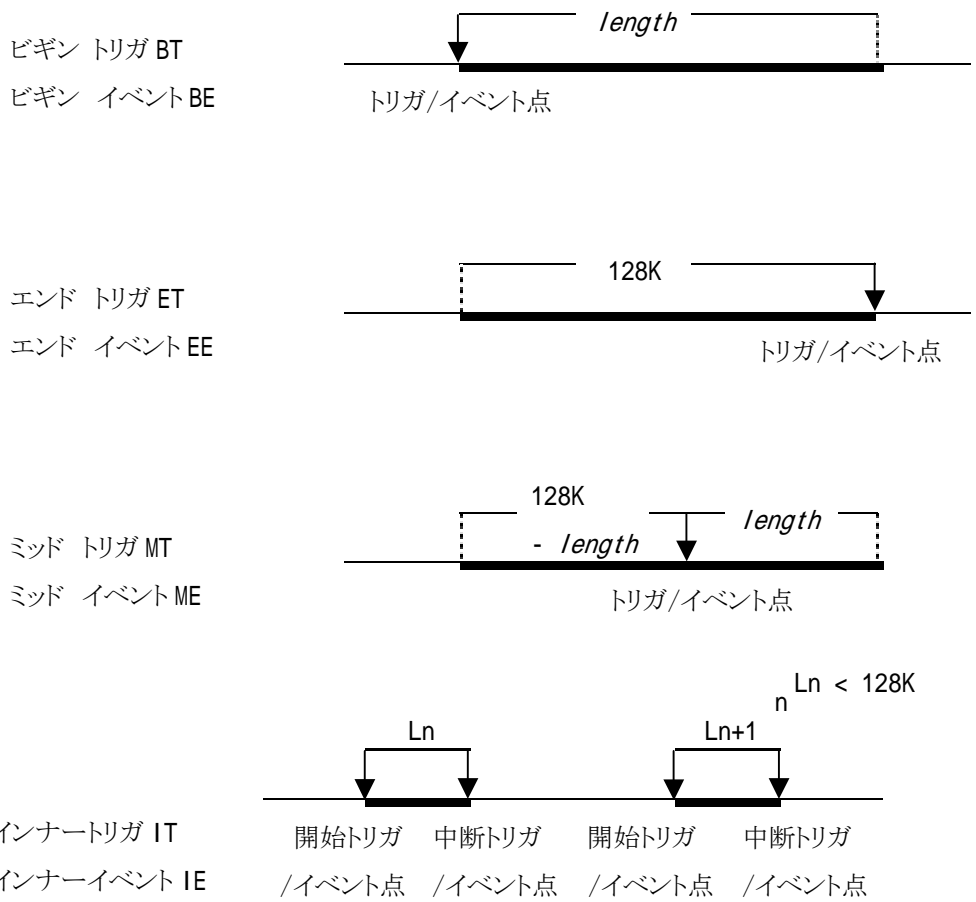
TRACE/I BT, 1000 (実行トレース条件をビギン トリガに、トレース トリガのアドレスを 0x1000 に設定します)

補足:

- MJX440 のトレース バッファの長さは 128K ワードです。1ワードに N-Wire からのトレースデータを8bit 入れることができます。
N-Wire のトレース パケットは可変長であり、指定長のトレース バッファに何パケット入るかは予測できません。
- トレース スタートと分岐が同時に発生した場合、分岐パケットが出ません。
この場合、トレース 逆アセンブラ表示が実行アドレス喪失状態になります。
トレース スタートと分岐命令が重ならないようなアドレスを推奨します。
また、トレース スタートと割り込みが重ならないように制御することはできません。
この場合、トレース 逆アセンブラ表示が正しく行われません。
トレース スタートアドレスは各実行トレース条件ごとに以下のようになっています。
 - ・IT, IE : トレース開始アドレス/トレース開始イベント
 - ・その他 : プログラム開始アドレス
- BT, ET, MT, BE, EE, ME の各実行トレース条件ではトリガ/イベント点で MJX440 の EXTERNAL コネクタの TRGOUT 端子から LOW レベルが出力されます。
IT, IE では、TRGOUT は出ません。
- BE, EE, ME モードではトリガ イベントを複数設定できます。最初に発生したイベントがトリガになります。また、IE モードでもイベントの複数設定が可能ですが、コマンドラインでは入力することができません。イベント設定ダイアログボックスで設定してください。
- 「第八章 [設定\(S\) ~ トレース\(T\)...](#)」参照。

トレース範囲:





TRACE/R

全トレース条件を無効にします

形式:

TRACE/R トレース条件をすべて無効にします。

引数:

例:

TRACE/R (トレース条件をすべて無効にします)

補足:

- 「第八章 [設定\(S\) ~ トレース\(T\)...](#)」参照。

UNASM

メモリの内容を逆アセンブル表示します

形式:

UNASM [*addr* [, *count*]] アドレス *addr* から *count* 命令分のメモリの内容を逆アセンブル表示します。

引数:

addr 逆アセンブル表示メモリ開始アドレス (16 進数)
count 逆アセンブル表示命令数 (10 進数、省略時 16)

例:

UNASM 1000 (0x1000 番地から 16 命令分のメモリを逆アセンブル表示します)
UNASM (前回の UNASM コマンドの続きを表示します)

補足:

- *count* 省略時は、16 命令分のメモリを逆アセンブル表示します。
- *addr* 省略時は、前回の UNASM コマンドの続きを表示します。
- 「第八章 [表示\(V\) ~ 逆アセンブル\(D\)...](#)」参照。
- 「付録 L [V850/ME2 内蔵命令 RAM の取り扱い](#)」参照。

VERSION

ソフトウェアのバージョンを表示します

形式:

VERSION	次のバージョン情報を表示します。
MjxDebw	MJXDEBW.EXE ソフトウェアのバージョン
DLL	CPU 依存 DLL ライブラリのバージョン
DRV	デバイス ドライバのバージョン
MJX	エミュレータのバージョン
CPU	CPU の種類
	DCU のバージョン

例:

VERSION

補足:

- 「第八章 [ヘルプ\(H\) ~ MJXDEBW のバージョン情報\(A\)](#)」参照。

WAIT

ユーザ プログラムが停止するまで待ちます

形式:

WAIT [*time*] ユーザ プログラムが停止するまで待ちます。指定時間内に停止しない場合は、ユーザ プログラムを強制停止します。

引数:

time 待ち時間 (10 進数、単位ミリ秒、省略時は∞)

例:

GO (ユーザ プログラムを実行します)

WAIT 1000 (ユーザ プログラムが停止するまで待ちます。1 秒以内に停止しない場合は、ユーザ プログラム強制停止します)

補足:

- 待ち時間はミリ秒単位で指定できますが、実際の精度は百ミリ秒程度です。
- バッチファイル内で使用してください。

XPIN

外部トリガ ケーブルの信号の状態を表示/設定します

形式:

XPIN 外部トリガ ケーブルの信号の状態を表示します。
 XPIN *ch, level* 外部トリガ ケーブルの出力信号 *ch* をレベル *level* に設定します。

引数:

<i>ch</i>	外部トリガ ケーブルの出力信号のピンの指定
1	EXTOUT1
2	EXTOUT2
<i>level</i>	出力ピンのレベル指定
0	LOW レベル
1	HIGH レベル

例:

XPIN (外部トリガ ケーブルの信号の状態を表示します。)
 XPIN 1,0 (外部トリガ ケーブルの出力信号EXTOUT1をLOWレベルに設定します。)

補足:

- 出力信号は負論理です。

第八章 MJXDEBW メニューコマンド

この章ではデバッガ MJXDEBW のメニュー コマンドの使い方について記述しています。

メニュー コマンドを使用すると、「第七章 [MJXDEBW コマンド](#)」に相当することを、メニューから実行させることができます。各メニューには、次のようにコマンドが割り当てられています。

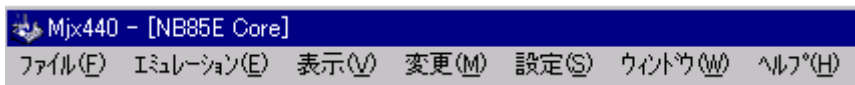


図8-1 MJXDEBW のメニューバー

ファイル(F)	ファイル メニューには、ファイルからコマンドを読み込みバッチ処理を行なうコマンド、コマンドの実行結果をファイルに残すジャーナル処理のコマンドなどが割り当てられています。
エミュレーション(E)	エミュレーション メニューには、ユーザ プログラムの実行制御、プログラムのダウンロード、初期化コマンドが割り当てられています。
表示(V)	表示メニューには、次のウィンドを表示するためのコマンドが含まれています メモリ ウィンドウ、レジスタ ウィンドウ、逆アセンブル ウィンドウ、トレース結果ウィンドウ。
変更(M)	メモリ/レジスタなどを変更するためのコマンドが割り当てられています。
設定(S)	ブレークポイントやトレース モードを設定するためのコマンドが割り当てられています。
ウィンドウ(W)	ウィンドウの表示形式を変更するためのコマンドが割り当てられています。 (Windows 準拠)
ヘルプ(H)	MJXDEBW のバージョン情報を表示するためのコマンドが割り当てられています。

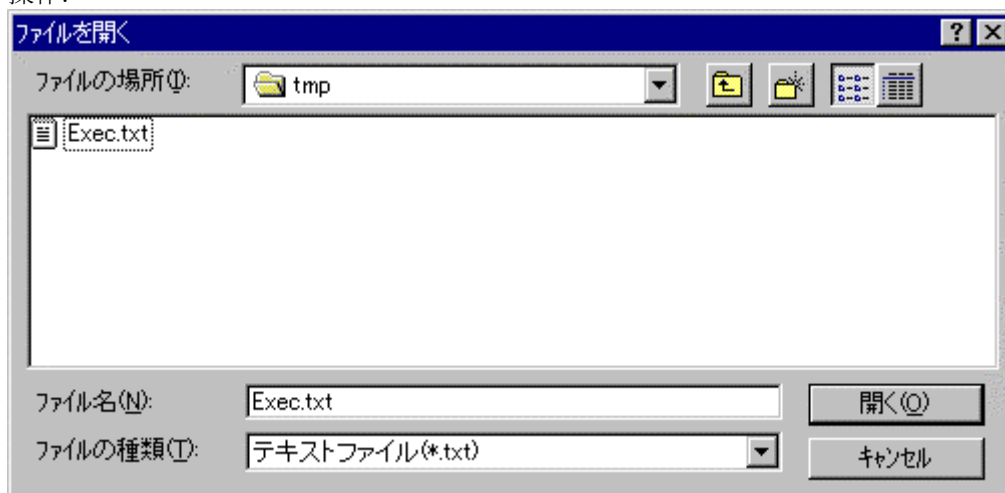
ファイル(F) ~ バッチ(B)...

BATCH

機能:

バッチ ファイルに記述された MJXDEBW コマンドを連続して実行します。

操作:



ダイアログ ボックスで、ファイル名を指定し、「開く」ボタンを押してください。

補足:

- 「第七章 [BATCH](#)」参照。

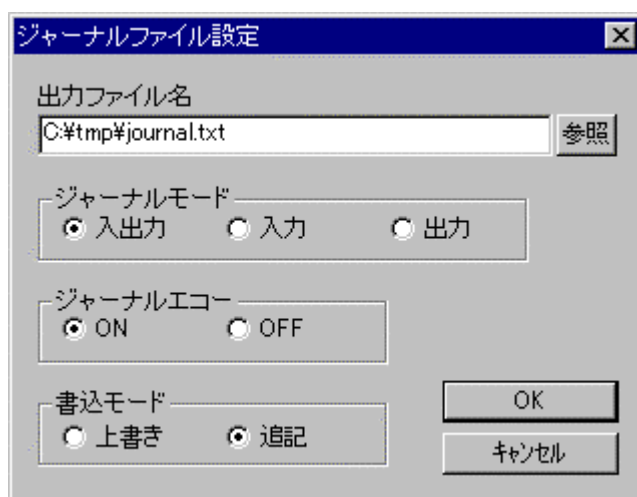
ファイル(F) ～ ジャーナル開始(S)...

JOURNAL

機能:

コマンド実行結果をファイルへ出力する、ジャーナル機能を開始します。

操作:



ダイアログ ボックスで、出力ファイル名、ジャーナル モード、ジャーナル エコー、書き込みモードを指定し、「OK」ボタンを押してください。

補足:

- 「第七章 [JOURNAL](#)」参照。

ファイル(F) ～ ジャーナル停止(E)

JOURNAL/E

機能:

ジャーナル機能を終了します。

補足:

- 「第七章 [JOURNAL](#)」参照。

ファイル(F) ~ トレース結果(T)...

HISTORY

機能:

トレース結果をファイルへ出力します。

補足:

- 現在未対応の機能です。

エミュレーション(E) ~ 実行(G)...

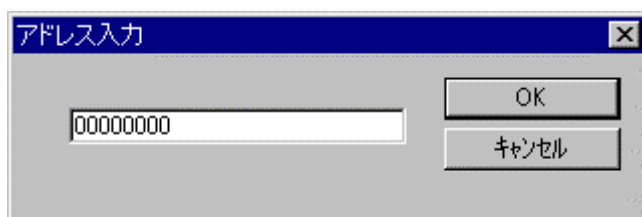
GO *addr*

Ctrl+G

機能:

プログラムの実行を開始するアドレスを指定してユーザ プログラムを実行します。

操作:



ダイアログ ボックスで、プログラムの実行を開始するアドレス(16 進数)を指定し、「OK」ボタンを押してください。

補足:

- 「第七章 [GO](#)」参照。

エミュレーション(E) ~ 再スタート(R)

GO

F5

機能:

現在の PC のアドレスからユーザ プログラムを実行します。

補足:

- 「第七章 [GO](#)」参照。

エミュレーション(E) ~ 中断(B)

ABORT
Ctrl+B

機能:

ユーザ プログラムを強制停止します。

補足:

- 「第七章 [ABORT](#)」参照。

エミュレーション(E) ~ ステップ(S)

STEP
F10

機能:

ユーザ プログラムをステップ実行します。

補足:

- 「第五章 [STEP](#)」参照。

エミュレーション(E) ~ N-ステップ(N)...

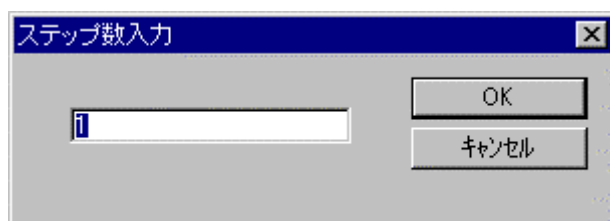
STEP *num*

Ctrl+S

機能:

ユーザ プログラムを指定回数分ステップ実行します。

操作:



ダイアログ ボックスで、ステップ実行回数(10 進数)を指定し、「OK」ボタンを押してください。

補足:

- 「第五章 [STEP](#)」参照。

エミュレーション(E) ~ ダウンロード(L)...

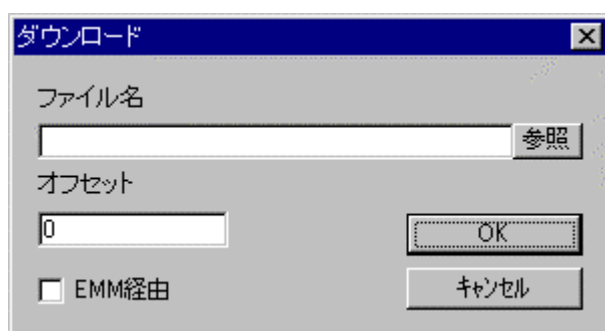
LOAD

Ctrl+L

機能:

MJX バイナリ^{*1}、S レコード、インテル HEX、または COFF 形式のファイル *file* をメモリへダウンロードします。

操作:



ダイアログ ボックスで、ファイル名、オフセット、EMM 経由を指定し、「OK」ボタンを押してください。

補足:

- ファイル形式は、自動認識されます。
- EMM 経由をONにするとエミュレーションメモリの内容は破壊されますがより高速にダウンロードが実行できます。
- 「第七章 [LOAD](#)」参照。
- 「付録 L [V850/ME2 内蔵命令 RAM の取り扱い](#)」参照。

^{*1} MJX バイナリ ファイルの作成方法については「第九章 高速ダウンロード」を参照してください。

エミュレーション(E) ～ 初期化(I)

INIT

機能:

MJX を再初期化します。

補足:

- 「第七章 [INIT](#)」参照。

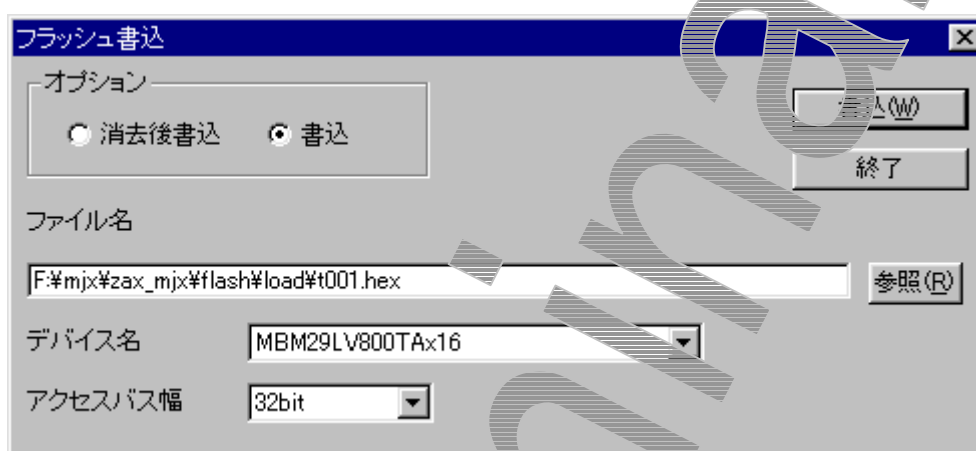
エミュレーション(E) ~ フラッシュメモリ(F) ~ 書込(W)...

FLASH WRITE

機能:

MJX バイナリ、Sレコード、インテル HEX、または COFF 形式ファイルをフラッシュメモリへ書き込みます。

操作:



ダイアログボックスで、ファイル名、デバイス名、アクセスバス幅を指定して、「書込」ボタンを押してください。

補足:

- ファイル形式は自動認識されます。
- 「[第七章 FLASH](#)」参照。

エミュレーション(E) ~ フラッシュメモリ(F) ~ 消去(E)...

FLASH/ERASE ONE

FLASE/ERASE ALL

機能:

指定された消去アドレスに該当するフラッシュメモリをセクタ消去、またはチップ消去します。

操作:



ダイアログボックスで消去アドレス、デバイス名、アクセスバス幅を指定して、「消去」ボタンを押してください。

補足:

- 「第七章 [FLASH](#)を照。

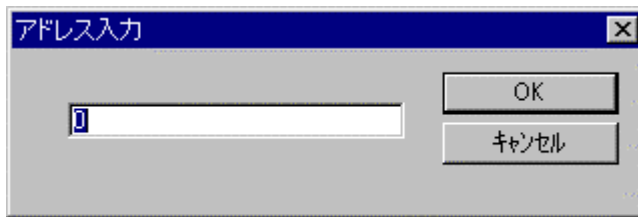
表示(V) ~ メモリ(M)...

DUMP

機能:

メモリの内容を表示する、メモリ ウィンドウを開きます。

操作:



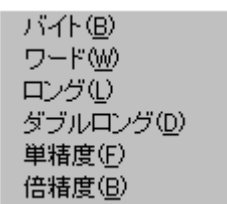
ダイアログ ボックスで、メモリ表示を開始するアドレス(16 進数)を指定し、「OK」ボタンを押してください。

```

メモリ:00000000
00000000  bb 6e ef f6 96 77 ea 7f ff ff c9 ed 6d bd ff fb  .n .w . . . . 匂.
00000010  bb 7e bf 7f fb fd fb 87 ff ff fd e3 ff ff df af  .~ . . . 蟬 . . . .
00000020  df ff f7 df 3f db ef ff bd fb fd fb df cf fe d7  . . . ? . . . . .
00000030  9f f5 5f f5 fb 7a a6 ff bf 76 ff f7 e7 04 b3 eb  澁 _ . z . . . v . .
00000040  7f ae bf f6 be bf fe bc fa e7 f5 fe ff df aa fd  . . . . . . . . .
00000050  ff 4c 51 f4 af 52 bd f9 bf 9f be ca b7 de ff f9  .LQ . R . . . . .キ.
00000060  ff ff ff 9f 5d ff b7 5a b1 d7 ee 4f 77 7e b7 d7  ... 歎 .キZ ... 0w
00000070  be ff 77 bd 9f 7f ff 7c fe fc 77 f3 97 cb bd f6  ..w . . . | . . . .
00000080  df 7f 97 b4 35 73 67 ce fc ff ef f7 ff dd ff 8d  ... 5sg . . . . .
00000090  fe 7b 95 c3 bb 69 f7 fd ee d6 bf 76 ee fb ef fe  . { . . . i . . . . v .
000000A0  bf 7f 9e 5d df df fd fb df f5 7f 77 fd b2 bf 97  ..枉 . . . . .w .
000000B0  cf 1f af f8 7f eb bf a7 bb f5 ed ff 7f eb f5 5f  . . . . .

```

新しくメモリ ウィンドウが表示されます。



メモリ ウィンドウ内で、マウスの右ボタンを押すと、ポップアップ メニューが表示され、表示形式を変更することができます。

補足:

- メモリ ウィンドウは複数開くことができます。開くことができるウィンドウ個数は、「[表示\(V\) ~ オプション\(O\) ~ 表示\(V\)...](#)」メニューで変更することができます。
- 「第七章 [DUMP](#)」参照。
- 「付録L [V850/ME2 内蔵命令 RAM の取り扱い](#)」参照。

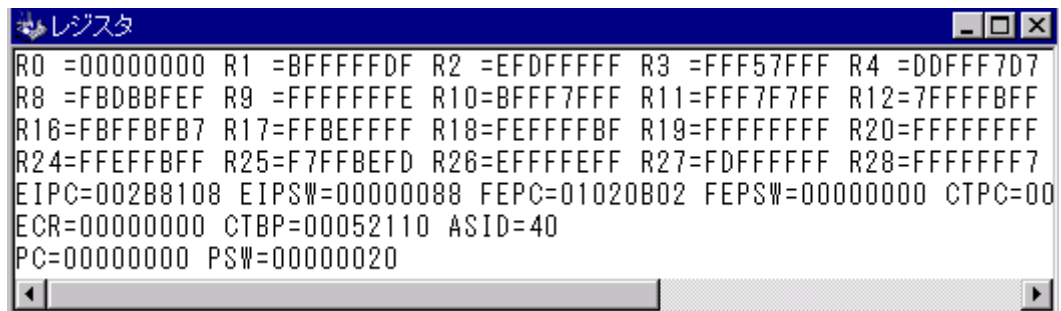
表示(V) ~ レジスタ(R)

REGISTER

機能:

レジスタの内容を表示する、レジスタ ウィンドウを開きます。

操作:



```
レジスタ
R0 =00000000 R1 =BFFFFFFD R2 =EFDFFFFFF R3 =FFF57FFF R4 =DDFFF7D7
R8 =FBDBBFEB R9 =FFFFFFFFE R10=BFFF7FFF R11=FFF7F7FF R12=7FFFFBFF
R16=FBFFBFB7 R17=FFBEFFFF R18=FEFFFFBF R19=FFFFFFFF R20=FFFFFFFF
R24=FFEFFBFF R25=F7FFBEFD R26=EFFFEFF R27=FDFFFFFF R28=FFFFFFF7
EIPC=002B8108 EIPSW=00000088 FEPC=01020B02 FEPSW=00000000 CTPC=00
ECR=00000000 CTBP=00052110 ASID=40
PC=00000000 PSW=00000020
```

新しくレジスタ ウィンドウが表示されます。

補足:

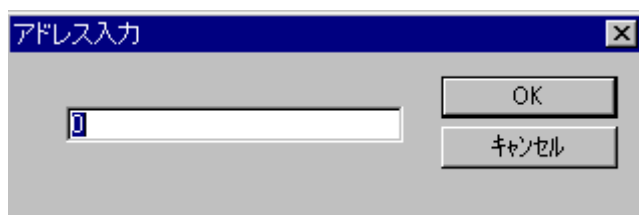
- 「第七章 [REGISTER](#)」参照。

表示(V) ~ 逆アセンブル(D)...

UNASM

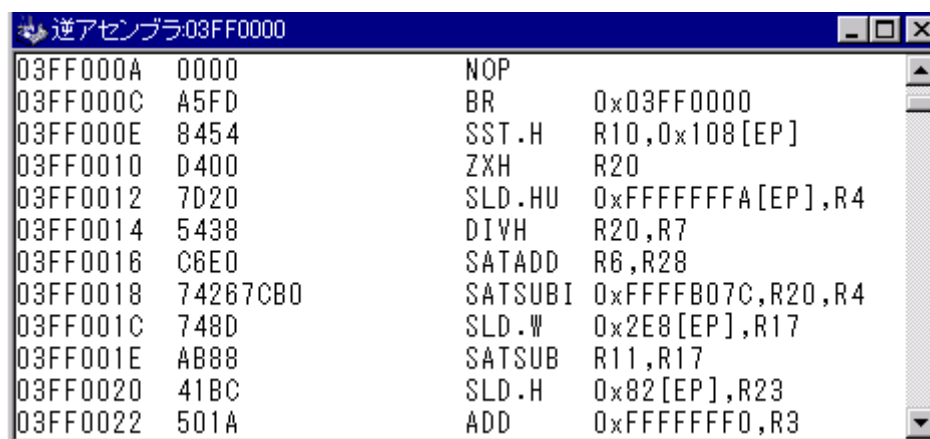
機能:

メモリの内容を逆アセンブル表示する、逆アセンブラ ウィンドウを開きます。



操作:

ダイアログ ボックスで、逆アセンブル表示を開始するアドレス(16 進数)を指定し、「OK」ボタンを押してください。



新しく逆アセンブラ ウィンドウが表示されます。

補足:

- 逆アセンブラ ウィンドウは複数開くことができます。開くことができるウィンドウ個数は、「[表示\(V\) ~ オプション\(O\) ~ 表示\(V\)...](#)」メニューで変更することができます。
- 「第七章 [UNASM](#)」参照。
- 「付録 L [V850/ME2 内蔵命令 RAM の取り扱い](#)」参照。

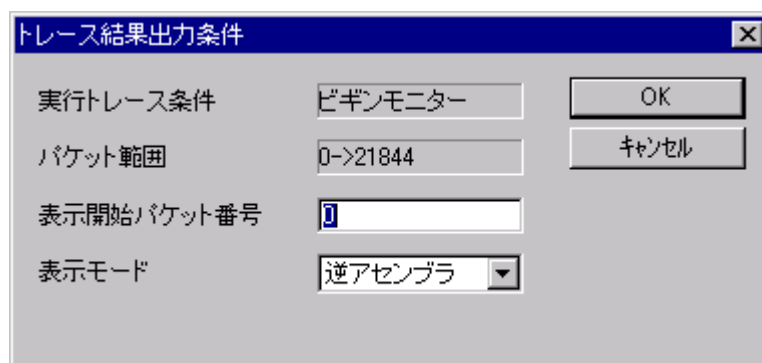
表示(V) ~ トレース結果(T)...

HISTORY

機能:

リアルタイム トレース結果の内容を表示する、トレース結果ウィンドウを開きます。

操作:



ダイアログボックスで、表示開始パケット番号、表示モードを指定し、「OK」ボタンを押してください。トレース結果ウィンドウが表示されます。

(逆アセンブラ表示の例)

```

トレース結果
03FF001E  NOP
03FF0020  NOP
      1 00000003 00 TRCODE=0A PC=03FF0022 ASID=00 JINF=1; BccT(pre)
03FF0024  BR      0x03FF0000
03FF0000  NOP
03FF0002  ST.W   R2,0x0000[R1]
03FF0006  NOP
03FF0008  NOP
03FF000A  NOP
03FF000C  NOP
03FF000E  LD.W   0x0000[R1],R2
03FF0012  NOP
03FF0014  NOP
03FF0016  NOP
03FF0018  NOP
03FF001A  ADDI   0x0001,R2,R2
03FF001E  NOP
03FF0020  NOP
      2 00000006 00 TRCODE=0A PC=03FF0022 ASID=00 JINF=1; BccT(pre)

```

補足:

- 実行トレース条件については、「第八章 [設定\(S\) ~ トレース\(T\)...](#)」参照。
- 「第七章 [HISTORY](#)」参照。

サイクル範囲では、トレース バッファ メモリへ取り込まれたサイクルの範囲を表示します。サイクルの 0 点は「実行トレース条件」により異なります。

ビギン モニター	プログラム開始点
エンド モニター	プログラム終了点(トレース中止点)
ビギン トリガー/イベント	トリガー点
エンド トリガー/イベント	トリガー点
ミッド トリガー/イベント	トリガー点
インナートリガー/イベント	トレース開始 トリガー点

表示開始サイクルでは、トレース結果表示ウィンドウの最上行に表示するサイクルを指定します。

表示モードでは、トレース結果表示モードを指定します。次の中から選択します。

- パケット トレース出力をパケット単位で表示します。
- 逆アセンブラ トレース出力をパケット表示と共に命令コードとアセンブラ ニーモニックで表示します。

トレース パケット 表示フォーマット

[トレースパケットフォーマット共通部分]

DDDDDD XXXXXXXX CC TRCODE=XX ZZZZZZZZZZZZZZ
 ① ② ③④ ⑤ ⑥

- ① 行番号 6 桁の 10 進整数
- ② タイムスタンプ 8 桁の 16 進数 *1
- ③ 0 または 1 EXTIN1 信号の状態
- ④ 0 または 1 EXTIN2 信号の状態
- ⑤ トレース要因コード 2 桁の 16 進数
- ⑥ トレース要因毎に異なる個別表示部分

[トレースパケットフォーマット個別部分]

TRCODE=01 OVF=C SSW=C; Start
 ① ② ③

- ① 0 または 1 オーバーフローフラグ
- ② 0 または 1 トレースサブスイッチ状態 *2
- ③ トレーススタートポイント検出

TRCODE=02 OVF=C SCF=C SSW=C; Match Point
 ① ② ③ ④

- ① 0 または 1 オーバーフローフラグ
- ② 0 または 1 スタートパケット競合フラグ*3
- ③ 0 または 1 トレースサブスイッチ状態 *2
- ④ マッチポイント検出

TRCODE=03; Overflow
 ①

- ① トレースデータオーバーフロー発生

TRCODE=08 PC=XXXXXXXX ASID=XX JINF=C; YYYYYY
 ① ② ③ ④

- ① 分岐先アドレス 8 桁の 16 進数
- ② ASID 2 桁の 16 進数
- ③ 0 または 1 分岐元パケット付加情報 *4
- ④ 分岐先 PC トレース要因(割込、例外、分岐命令による分岐(分岐先 PC)、強制スタート)*5

TRCODE=09 PC=XXXXXXXX ASID=XX JINF=C; YYYYYY/WWWWW
 ① ② ③ ④ ⑤

- ① 分岐先/元アドレス 8 桁の 16 進数
- ② ASID 2 桁の 16 進数
- ③ 0 または 1 分岐元パケット付加情報 *4
- ④ 分岐先 PC トレース要因(割込、例外、分岐命令による分岐)*6
- ⑤ 分岐元 PC トレース要因(割込、例外、分岐命令による分岐、強制スタート)*5

TRCODE=0A PC=XXXXXXXX ASID=XX JINF=C; YYYYYY
 ① ② ③ ④

- ① 分岐元アドレス 8 桁の 16 進数
- ② ASID 2 桁の 16 進数
- ③ 0 または 1 分岐元パケット付加情報 *4
- ④ 分岐元 PC トレース要因(割込、例外、分岐命令による分岐(分岐元 PC))*6

TRCODE=0C XXXXXXXX XXXXXXXX TRG=C; Data Write
 ① ② ③ ④

- ① アクセスアドレス 8 桁の 16 進数
- ② データ値 2 桁,4 桁または 8 桁の 16 進数
- ③ 0 または 1 トリガポイント番号 *7
- ④ データトレースポイントへのライトアクセス検出

TRCODE=0D XXXXXXXX XXXXXXXX TRG=C; Data Read
 ① ② ③ ④

- ① アクセスアドレス 8 桁の 16 進数
- ② データ値 2 桁,4 桁または 8 桁の 16 進数
- ③ 0 または 1 トリガポイント番号 *7
- ④ データトレースポイントへのリードアクセス検出

***1** タイムスタンプ

1MHz で動作するフリーランカウンタの値です。
二つのタイムスタンプの差分を計算することでプログラム実行時間を計測することができます。

***2** トレースサブスイッチ状態

インナートリガまたはインナーイベントモードでトレースを実行すると
トレース開始点でトレースが再開されるときに SSW=1 になります。
トレースサブスイッチは MJXDEBW プログラムが制御を行うので、ユーザからは操作できません。

***3** スタートパケット競合フラグ

スタートパケットとマッチポイントパケットが競合した場合、スタートパケットは破棄されマッチポイント
パケットには競合を示す SCF=1 が出力されます。
なお Mjxdebw ではスタートポイントとマッチポイントは Mjxdebw プログラムから自動的に設定します
ので、ユーザからは操作できません。

***4** 分岐元パケット付加情報

V850E は2命令同時実行が可能です。
JINF=1 の場合、実際の分岐元アドレスは表示される(アドレス+2)となります。
なお TRCODE=08 の場合は分岐先アドレスが表示されるので JINF は意味を持ちません。

***5** 分岐先 PC トレース要因では以下の文字情報が表示されます。

INT(pst)	: 割り込み(NMI/INT)
EXCEPT(pst)	: 例外、トラップ命令(,DBTRP 命令)
RETI(pst)	: RETI 命令(,DBRET 命令)
JMP(pst)	: JMP 命令
JR(pst)	: JR 命令
JARL(pst)	: JARL 命令
BccF(pst)	: 条件付分岐命令 Bcond(条件不成立時)
BccT(pst)	: 条件付分岐命令 Bcond(条件成立時)
CALLT(pst)	: CALLT 命令
SWITCH(pst)	: SEITCH 命令
DISPOSE(pst)	: DISPOSE 命令
CTRET(pst)	: CRET 命令
Force Start	: 強制スタート
Unknown	: 未定義の分岐要因を示す

*6 分岐元 PCトレース要因では以下の文字情報が表示されます。

INT(pst)	: 割り込み(NMI/INT)
EXCEPT(pre)	: 例外、トラップ命令(DBTRP 命令)
RETI(pre)	: RETI 命令(DBRET 命令)
JMP(pre)	: JMP 命令
JR(pre)	: JR 命令
JARL(pre)	: JARL 命令
BccF(pre)	: 条件付分岐命令 Bcond(条件不成立時)
BccT(pre)	: 条件付分岐命令 Bcond(条件成立時)
CALLT(pre)	: CALLT 命令
SWITCH(pre)	: SEITCH 命令
DISPOSE(pre)	: DISPOSE 命令
CTRET(pre)	: CRET 命令
Unknown	: 未定義の分岐要因を示す

*7 トリガポイント番号

トリガ番号については MJXDEBW プログラムが自動的に0または1を選択します。
ユーザからは操作できません。

表示(V) ~ コマンド応答クリア(C)

CLEAR

機能:

コマンド応答フィールドをクリアします。

補足:

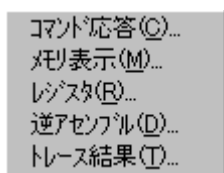
- 「第七章 [CLEAR](#)」参照。

表示(V) ~ オプション(O) ~ フォント(F)

機能:

ウィンドウに表示する文字のフォントを変更します。

操作:



メニューで、フォントを変更したいウィンドウを選びます。



ダイアログ ボックスで、フォント、スタイル、サイズを指定し、「OK」ボタンを押してください。

補足:



表示(V) ~ オプション(O) ~ 表示(V)...

機能:

コマンド ウィンドウ、メモリ ウィンドウ、逆アセンブル ウィンドウの属性を指定します。

操作:



ダイアログ ボックスで、各ウィンドウの属性を指定し、「OK」ボタンを押してください。

補足:

-

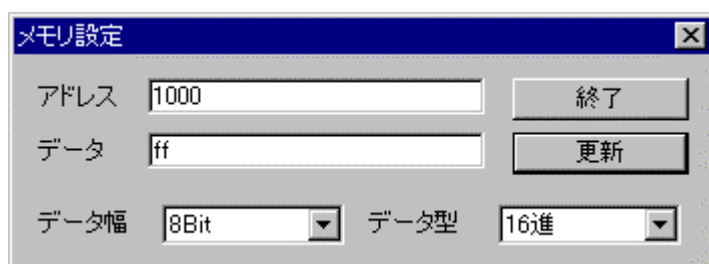
変更(M) ~ メモリ(M)...

EXAMINE

機能:

メモリの内容を変更します。

操作:



ダイアログ ボックスで、変更するメモリのアドレス(16 進数)、データ、データ幅、データ型を指定し、「更新」ボタンを押してください。

「更新」ボタンを押すとメモリが変更され、ダイアログのアドレスが自動的に進みます。続けてメモリを変更する場合は、同様の手順を繰り返してください。

メモリの変更が終了したら、「終了」ボタンを押してください。

補足:

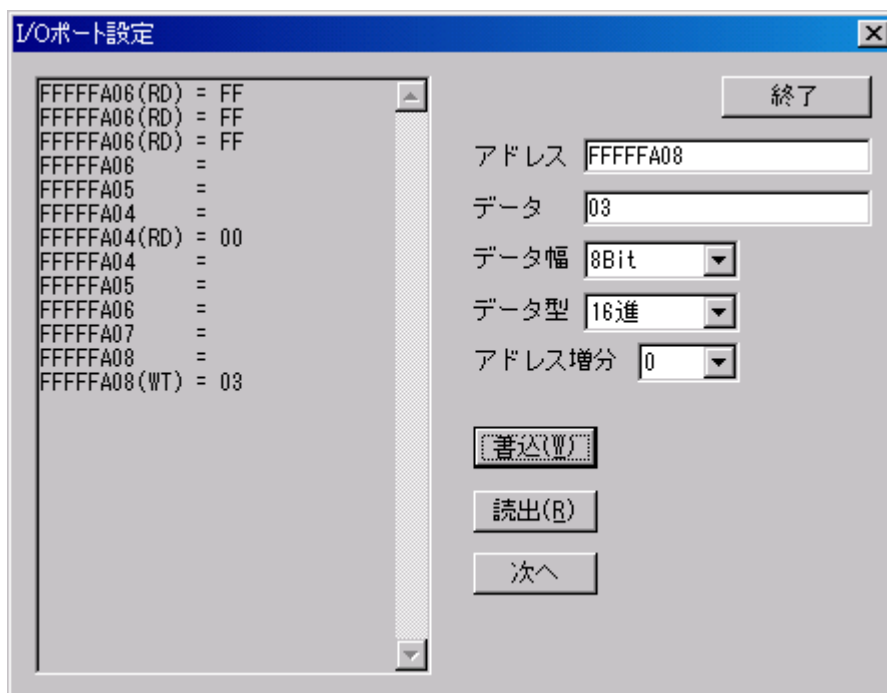
- 「第七章 [EXAMINE](#)」参照。
- 「付録 L [V850/ME2 内蔵命令 RAM の取り扱い](#)」参照。

変更(M) ~ I/O ポート(P)...

機能:

I/O ポートの内容を変更します。

操作:



ダイアログ ボックスで、変更する I/O ポートの アドレス(16 進数)、データ、データ幅、データ型、アドレス増分を指定し、「書込」または「読出」ボタンを押してください。

「次へ」ボタンを押すと I/O ポートアクセスを行わずにアドレスの増減ができます。

補足:

- MJX440 for NB85E ではメモリマップド I/O に対してアクセスを行います。

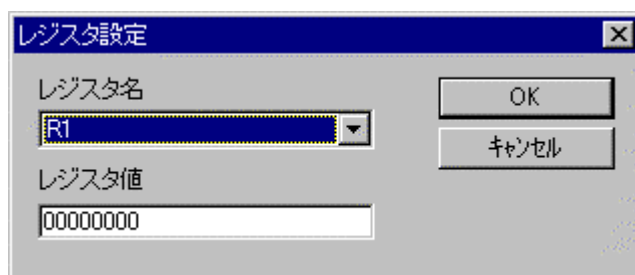
変更(M) ~ レジスタ(R)...

REGISTER

機能:

レジスタの内容を変更します。

操作:



ダイアログ ボックスで、レジスタ名、レジスタ値を指定し、「OK」ボタンを押してください。

補足:

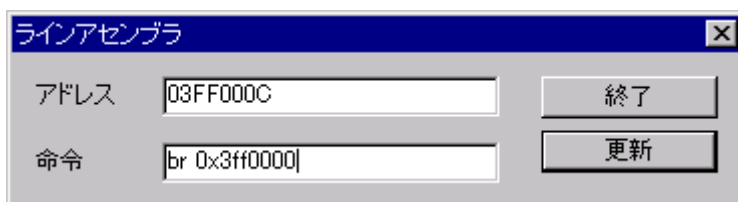
- 「第七章 [REGISTER](#)」参照。

変更(M) ~ アセンブラ(A)...

機能:

メモリの内容をアセンブラで変更します。

操作:



ダイアログ ボックスで、変更するメモリのアドレス(16 進数)、アセンブラの命令を指定し、「更新」ボタンを押してください。

「更新」ボタンを押すとメモリが変更され、ダイアログのアドレスが自動的に進みます。続けてメモリを変更する場合は、同様の手順を繰り返してください。

メモリの変更が終了したら、「終了」ボタンを押してください。

補足:

- シンボルは入力できません。
- 16 進数は、0x1234 のように先頭に「0x」を付加してください。
- 「付録 L [V850/ME2 内蔵命令 RAM の取り扱い](#)」参照。

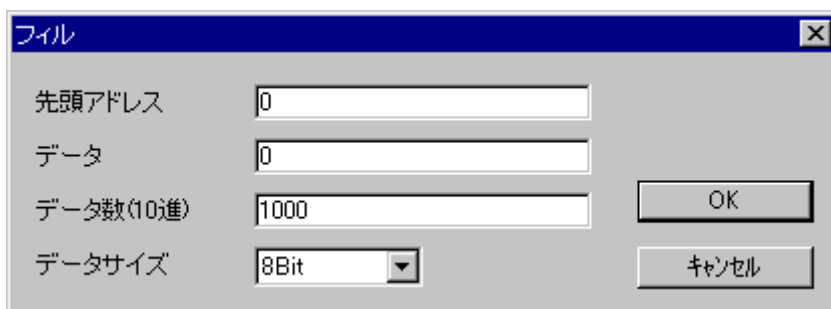
変更(M) ~ フィル(F)...

FILL

機能:

メモリの内容をフィルします。

操作:



ダイアログ ボックスで、変更するメモリの先頭アドレス(16 進数)、データ(16 進数)、データ数(10 進数)、データ サイズを指定し、「OK」ボタンを押してください。

補足:

- 「第七章 [FILL](#)」参照。
- 「付録 L [V850/ME2 内蔵命令 RAM の取り扱い](#)」参照。

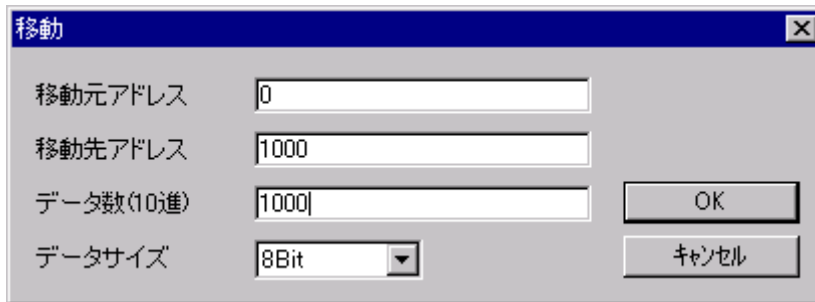
変更(M) ~ 移動(M)...

MOVE

機能:

メモリの内容をブロック転送します。

操作:



移動元アドレス	0
移動先アドレス	1000
データ数(10進)	1000
データサイズ	8Bit

OK
キャンセル

ダイアログ ボックスで、変更するメモリの移動元アドレス(16 進数)、移動先アドレス(16 進数)、データ数(10 進数)、データ サイズを指定し、「OK」ボタンを押してください。

補足:

- 「第七章 [MOVE](#)」参照。

設定(S) ~ ブレークポイント(B)...

BP

機能:

ブレークポイントを設定します

操作:

ブレークポイント設定

設定

ブレークポイント: ブレークタイプ: ソフトウェア データサイズ: ANY

アドレスマスク有効

データ比較有効

データマスク有効

ASID有効

登録(R) 終了

無効化 削除(D) すべて削除(A)

現在の設定

ブレークタイプ	ブレークポイント	アドレスマスク	データ	データマスク	サイズ	ASID	状態
Software	03FF0000	-----	-----	-----	----	--	ENA
H/WInst	03FF1000	-----	-----	-----	----	--	ENA
H/WAccess(R/W)	03FF1500	-----	-----	-----	ANY	--	ENA
H/WAccess(R/W)	03FF1500	-----	-----	-----	BYTE	--	DIS

表示選択:

イベントブレーク

hd001
sw001

変更(M)

設定フレームにあるブレークタイプを選択し、ブレークポイント(16 進数)を設定します。「登録」ボタンを押すと、ブレークポイントを設定できます。

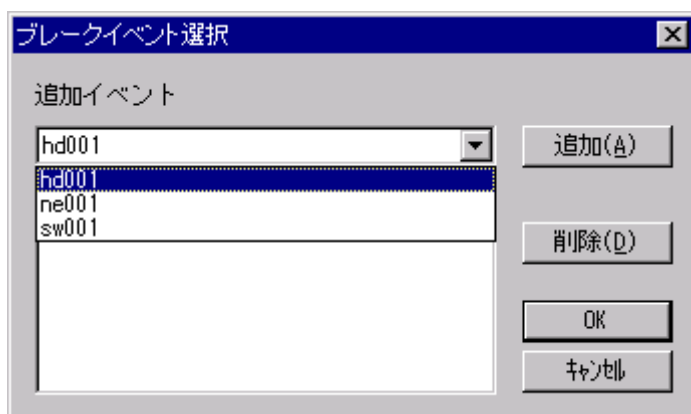
設定されたブレークポイントは、「現在の設定」に表示されます。

表示選択を選択すると、現在の設定の表示が変更されます。

現在の設定の中からひとつのブレークポイントを選択し「無効化」ボタンを押すと、ブレークポイントが無効化できます。同様の手順で「有効化」と「削除」ができます。

「すべて削除」ボタンを押すと、ブレークポイントをすべて削除します。

設定フレームにあるイベントブレークの「変更」ボタンを押すと、以下のようなブレークイベント選択画面が表示されます。



追加イベントからイベント名を選択して、「追加」—「OK」ボタンを押してください。

ブレークポイントの設定が終了したら、「終了」ボタンを押してください。

補足：

- 設定フレーム内の登録は、ブレークタイプを選択すると、それ以下の項目は設定必要な場合のみ入力できるようになっています。
- 「第七章 [BP](#)」参照。
- トレース トリガとして再利用したブレークポイントは削除できません。
トレース トリガを無効にしてから削除してください。

設定(S) ~ トレース(T)...

TRACE

機能:

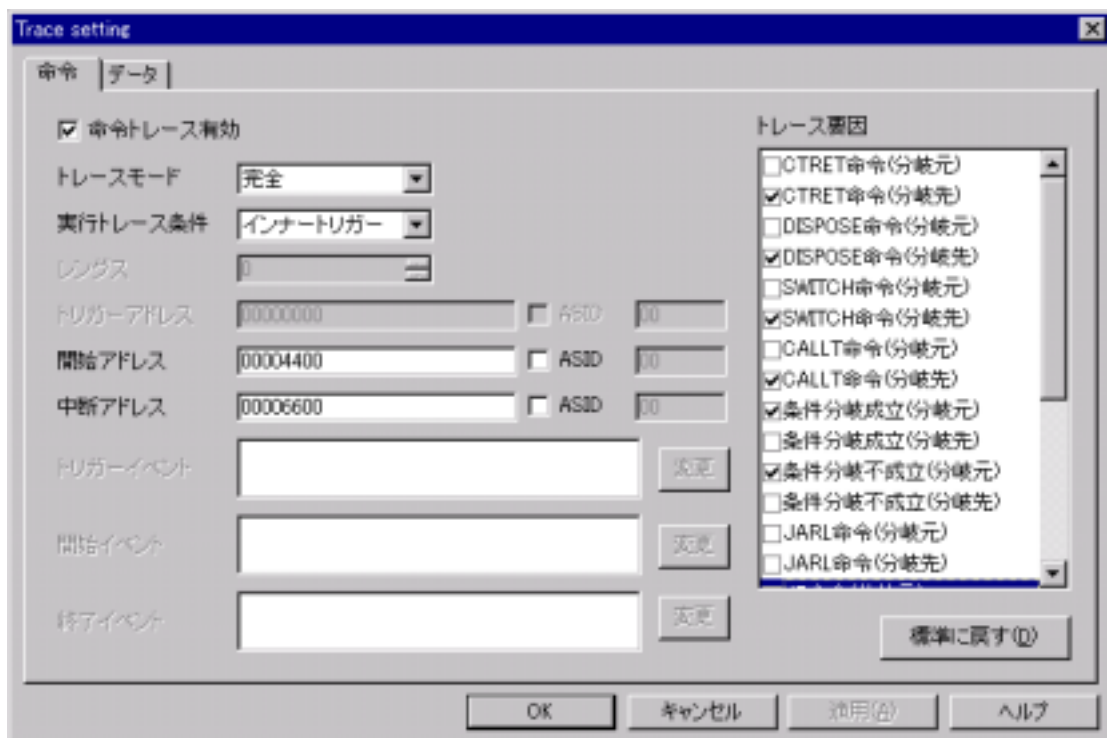
リアルタイム トレースのトレース モードを表示/設定します。

操作:

(1) 命令トレース条件の設定

ダイアログボックスの命令タブで、トレースモードを設定し、「OK」ボタンを押してください。

トレースモードの設定は次のとおりです。



命令トレース有効

チェックするとトレース全体が有効になります。

トレースモード

トレースモードは命令トレースのモードを指定します。トレースモードは以下から選択します。

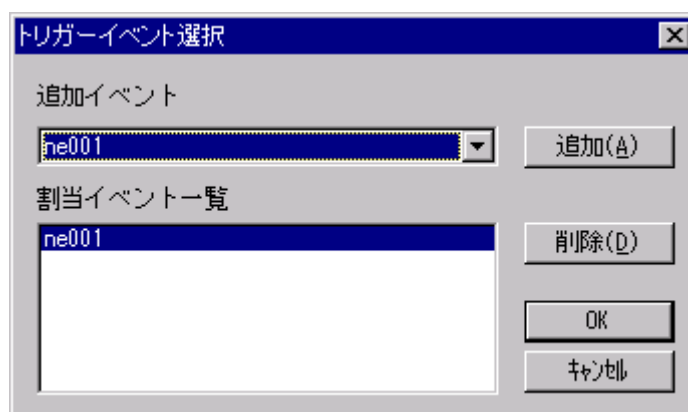
完全モード

通常モード

トレース条件	<p>トレース条件を選択します。トレース条件は、以下の中から選択します。</p> <ul style="list-style-type: none">ビギン モニターエンド モニタービギン トリガーエンド トリガーミッド トリガーインナートリガビギン イベントエンド イベントミッド イベントインナーイベント
レンジス	<p>トレースバッファの長さを指定します。長さの単位はトレースワードです。「トリガーモード」がビギンモニター／ミッドトリガー／インナートリガ／ミッドイベント／インナーイベントのとき設定できます。「レンジス」に設定できる値は 1～131071 です。</p>
トリガーアドレス	<p>トリガー アドレス情報を設定します。「トレース モード」がビギン トリガー／ミッド トリガー／エンド トリガー／インナートリガの場合指定できます。</p>
中断アドレス	<p>中断アドレス情報を設定します。「トレース モード」がインナートリガの場合設定できます。</p>
ASID	<p>チェックを有効にすると ASID 条件が有効になります。</p>
トレース要因	<p>チェックを有効にすると、トレース要因を設定できます。</p>

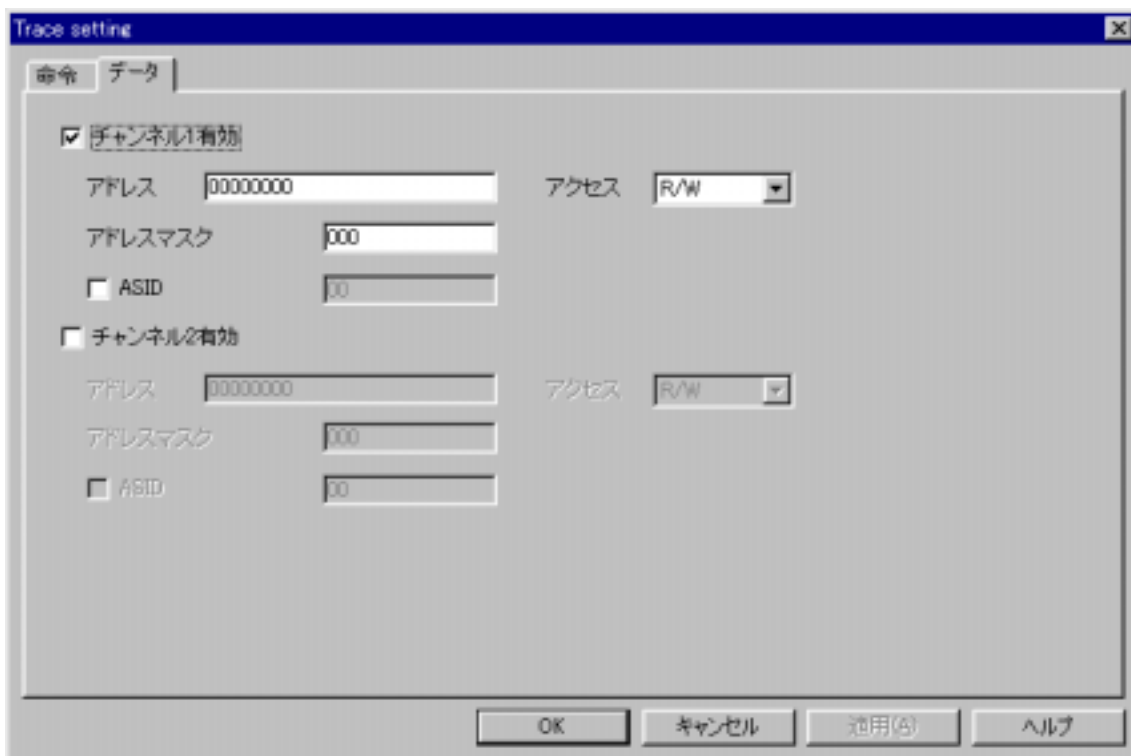
トリガーイベント	トリガーイベントを設定します。
開始イベント	インナーイベント条件のトレース開始イベントを指定します。
終了イベント	インナーイベント条件のトレース中断イベントを指定します。
イベント設定ウィンドウ	トリガーイベント、開始イベント、終了イベントの「変更」ボタンを押すと、下のようなイベント設定ウィンドウが表示されます。「追加イベント」からイベントを選択して「追加」ボタンを押します。イベントの設定を解除したい場合は、「割当イベント一覧」から削除するイベントを選択して、「削除」ボタンを押します。

<トリガーイベントのイベント設定ウィンドウ>



(2) データトレース条件の設定

ダイアログボックスのデータタブで、チャンネル有効を設定し、「OK」ボタンを押してください。



チャンネルの設定は次のとおりです

チャンネル有効 有効にしたいチャンネル番号のチェックボックスを選択します。
 選択したチャンネルのアドレス(16 進数)を指定しアクセスデータを選択します。必要に応じてアドレスマスク(16 進数)、ASID 条件の有効/無効を決定します。

補足:

- 「第七章 [TRACE](#)」参照。

設定(S) ~ イベント(E)...

EVENT

機能:

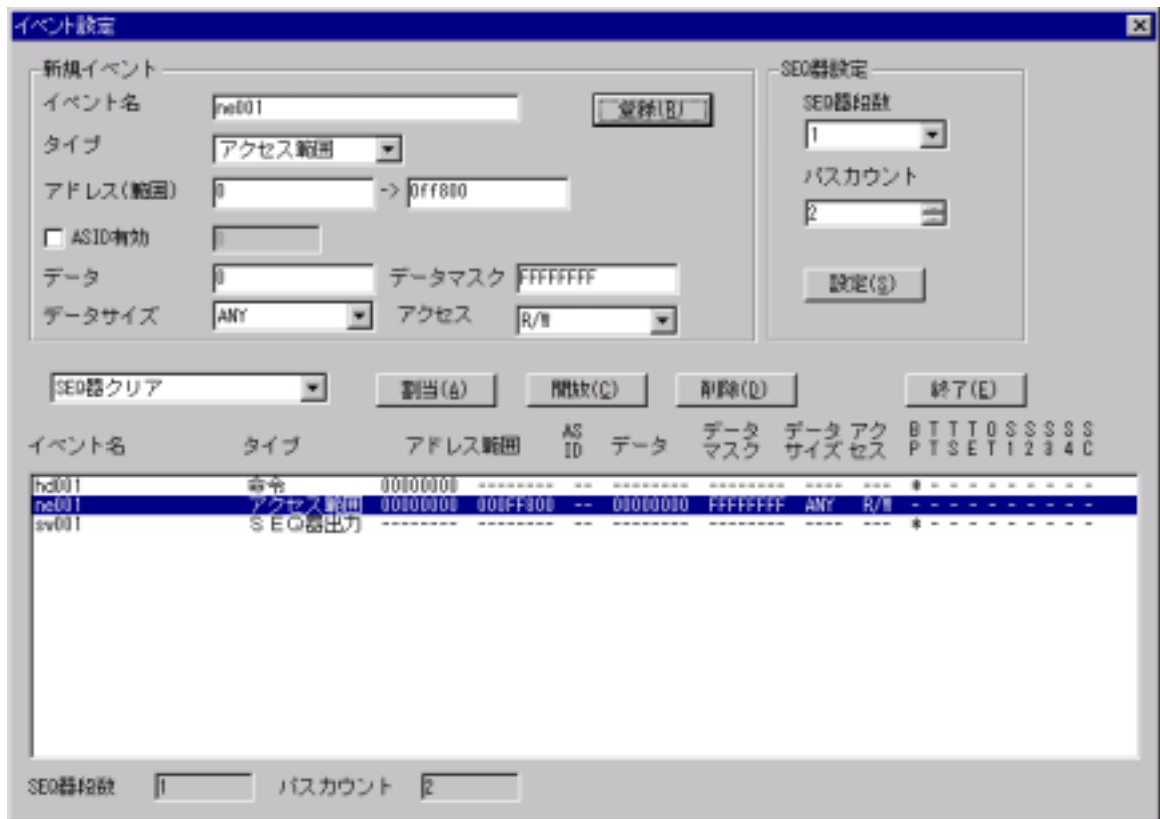
イベント機能構成と設定されている全イベントを表示します。

操作:

ダイアログボックスの「新規イベント」フレームを設定し、「登録」ボタンを押してください。

作成したイベントの詳細が表示されます。

「新規イベント」の詳細設定は、次のとおりです。



イベント名 イベント識別名を指定します。イベント識別名は、カンマ、スペース、タブを除く印字可能半角文字を指定します。

タイプ イベントの内容を選択します。



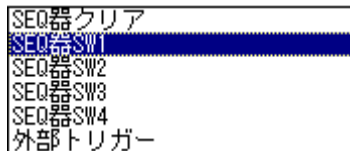
アドレス範囲 イベント開始アドレス(16進数)を設定します。

ASID 有効 ASID 条件の有効/無効を決定します。

シーケンス器を使用する場合は、「SEQ 器設定」フレームの SEQ 器段数とパスカウントを設定してください。

シーケンス器スイッチイベントを割り当てる場合は、イベント一覧ウィンドウから対象となるイベントの詳細を選択し、コンボボックスから、シーケンス器スイッチイベントを選択して「割当」ボタンを押してください。

(シーケンス器スイッチイベント)



「開放」ボタンを押すと、選択されているシーケンス器スイッチイベントが開放されます。「削除」ボタンを押すと、選択されたイベントが削除されます。

同様にして外部トリガー出力イベントの割り当てを行うことができます。

チャンネル割当が行われているイベントは*マークで表示が行われます。

割当対象の略称は以下のとおりです。

BP	:ブレイクポイント
TT	:トレース トリガ ポイント
TS	:トレース 開始ポイント
TE	:トレース 終了ポイント
OT	:外部トリガ出力ポイント
S1	:SEQ 器スイッチ1
S2	:SEQ 器スイッチ 2
S3	:SEQ 器スイッチ 3
S4	:SEQ 器スイッチ 4
SC	:SEQ 器クリアポイント

終了する場合は、「終了」ボタンを押してください。

補足:

- 「第七章 [EVENT](#)」参照。

設定(S) ～ 設定の読出(L)...

SETLOAD

機能:

ブレイクポイント、イベントおよびトレースの設定を、ファイルから読み込みます。

操作:



ダイアログ ボックスで、ファイル名を指定し、「開く」ボタンを押してください。

補足:

- 「[設定\(S\) ～ 設定の保存\(S\)...](#)」メニューで作成されたファイルを指定してください。

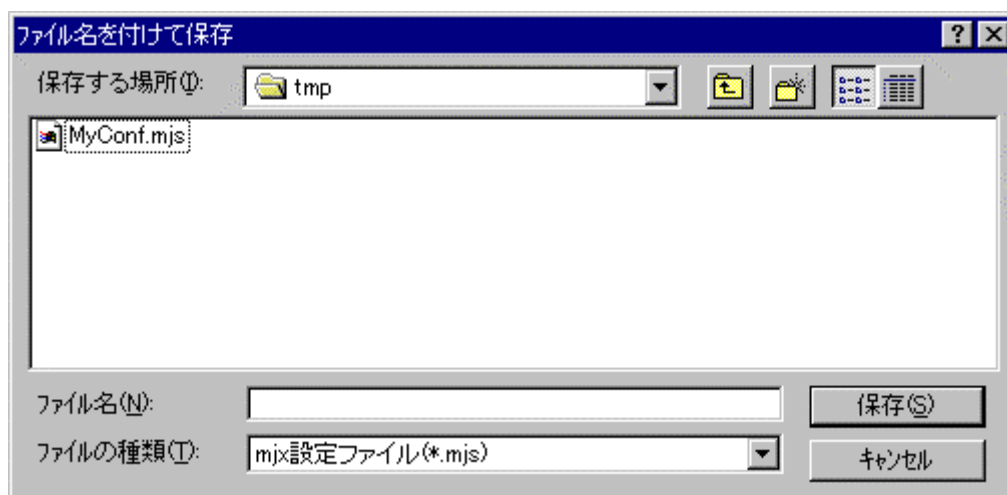
設定(S) ~ 設定の保存(S)...

SETSAVE

機能:

ブレイクポイント、イベントおよびトレースの設定をファイルへ保存します。

操作:



ダイアログ ボックスで、ファイル名を指定し、「開く」ボタンを押してください。

補足:

- 保存したファイルは「[設定\(S\) ~ 設定の読出\(L\)...](#)」メニューで指定することができます。

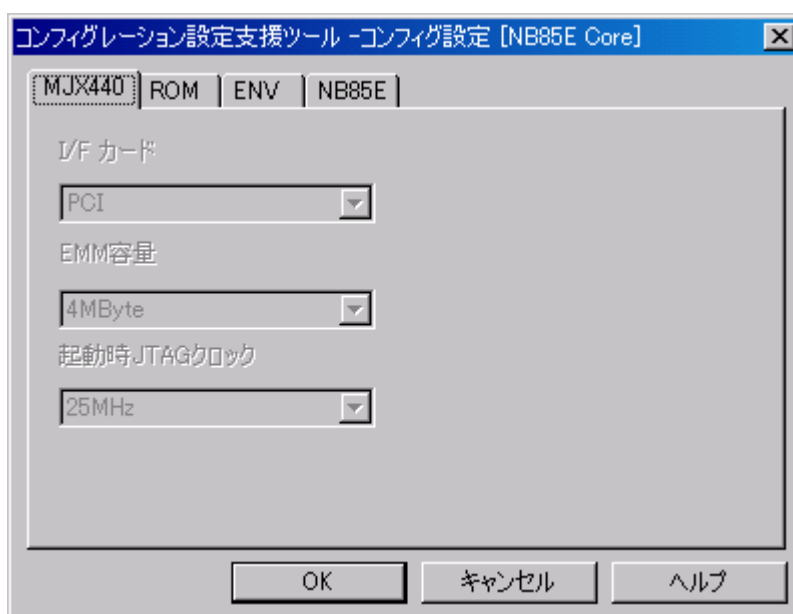
設定(S) ~ コンフィグレーション(C) ~ 設定(S)...

CONFIG

機能:

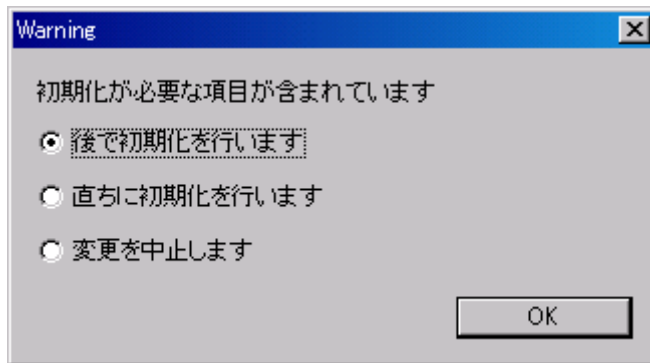
MJXDEBW の環境を設定します。

操作:



コンフィグレーション設定支援ツールが起動します。

変更する項目を設定してください。



初期化が必要な項目が変更された場合、確認ダイアログが表示されます。
適切な処置を選択し「OK」ボタンを押してください。

補足：

- MJXDEBW 起動中に変更できない項目は反転表示されます。
- 「第七章 [CONFIG](#)」参照。

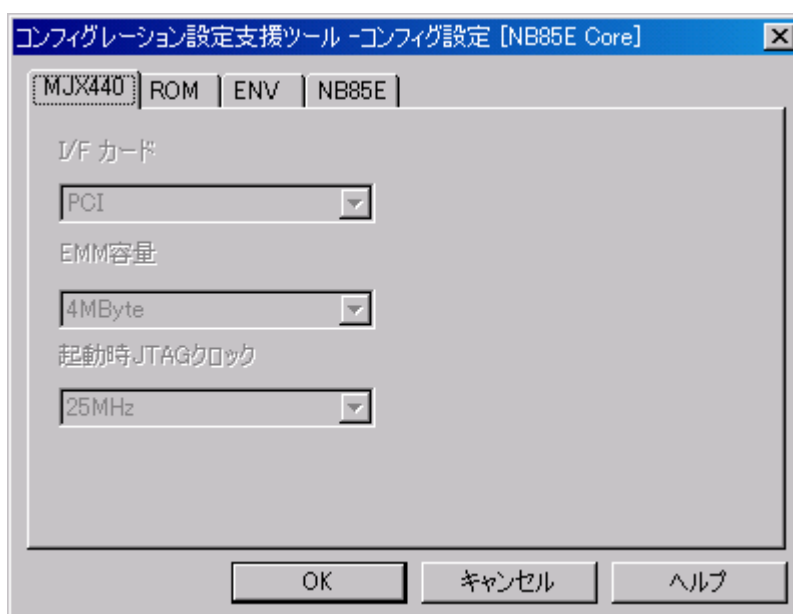
設定(S) ~ コンフィグレーション(C) ~ 表示(V)...

CONFIG

機能:

MJXDEBW の環境を表示します。

操作:



補足:

- 「第七章 [CONFIG](#)」参照。

設定(S) ~ コンフィグレーション ~ 上書き保存(E)

CONFIG

機能:

現在のコンフィグレーションの設定を MJXDEBW 起動時に指定したコンフィグレーションファイルへ上書き保存します。

操作:

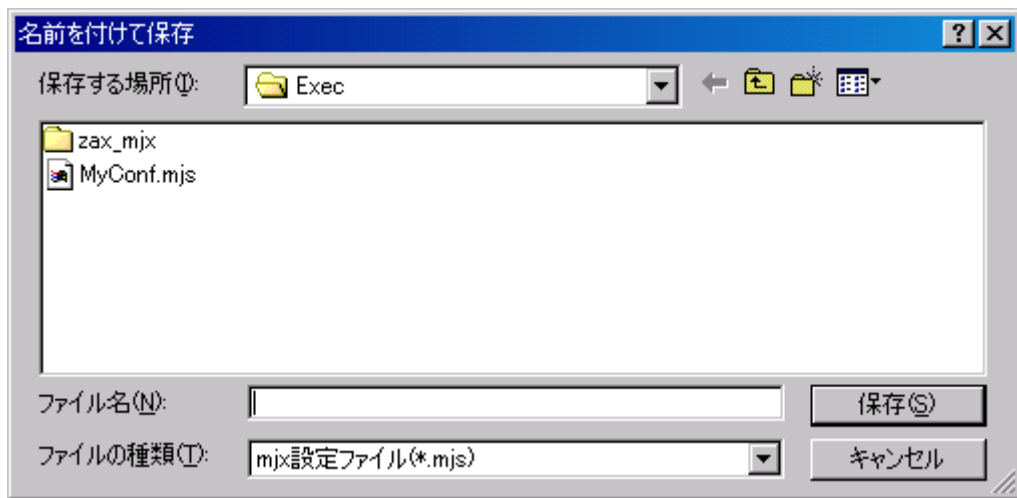
設定(S) ~ コンフィグレーション ~ 名前を付けて保存(A)

CONFIG

機能:

現在のコンフィグレーションの設定を新たなコンフィグレーションファイルへ保存します。

操作:



ダイアログボックスでファイル名を指定し、「保存」ボタンを押してください。

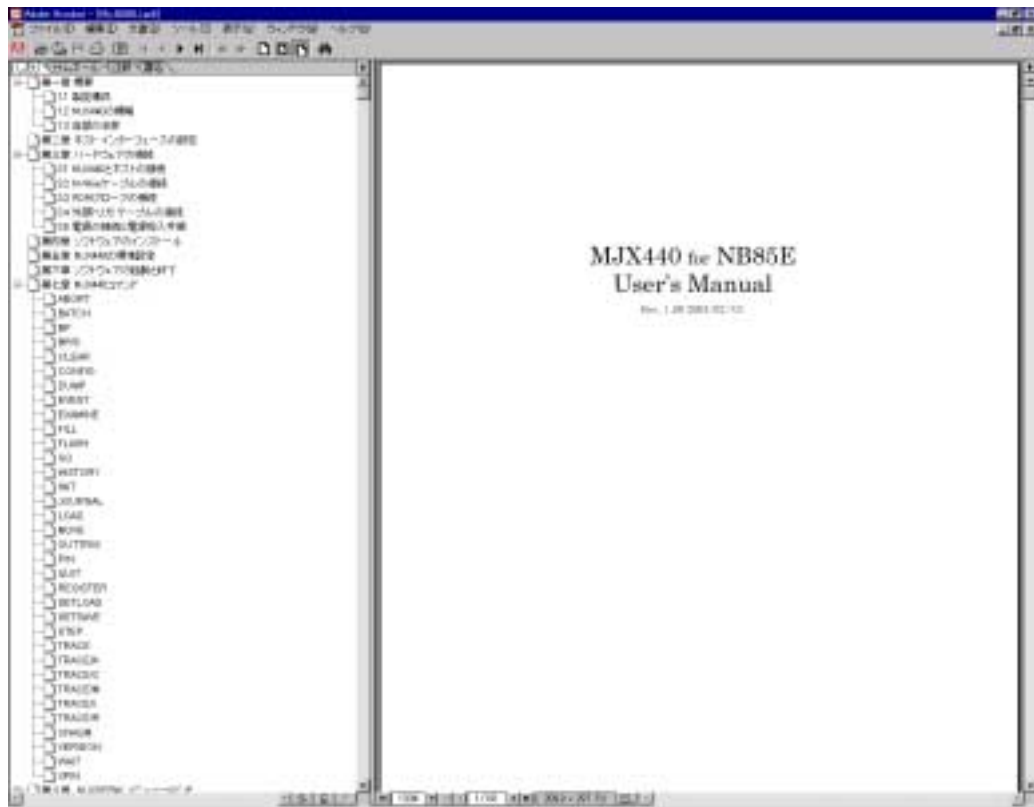
ヘルプ(H) ~ トピックの検索(H)

機能:

MJX440 for NB85E に関するマニュアル画面が表示されます。

操作:

マニュアル画面は「しおり」またはしおりの最後に記述されている「検索項目」のシートから参照してください。



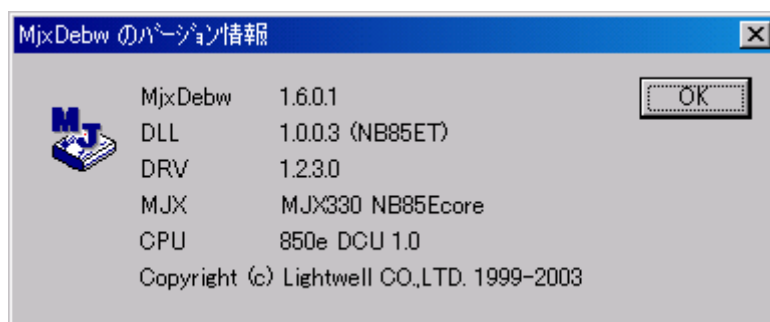
ヘルプ(H) ~ MJXDEBW のバージョン情報(A)

VERSION

機能:

バージョンを表示します

操作:



補足:

- 「第七章 [VERSION](#)」参照。

第九章 高速ダウンロード

高速ダウンロードを行なうための手順について記述しています。

[MJX バイナリ ファイル](#)を作成することによって、プログラムを高速でダウンロードすることができます。およそのダウンロード速度は、次のとおりです。(PCI カード使用の時)

- N-Wire ケーブル接続時、480K バイト/秒*1 (JTAG クロック 25MHz)
- ROM インサーキット接続時、4M バイト/秒

MJX バイナリ ファイルは、ファイル変換プログラム MJXCVT を使用して作成します。MJXCVT は、S レコード ファイルを MJX バイナリ ファイルへ変換するプログラムです。使用方法は、次のとおりです。MS-DOS プロンプト内から実行してください。

MJXCVT の使用方法

```
mjxcvt [-o offset] infile [outfile]
```

-o offset 出力ファイルにオフセット アドレスを加算する

infile 入力ファイル名

outfile 出力ファイル名 (省略時は、infile の拡張子を .mjk にしたファイル名)

MJX バイナリ ファイルのダウンロード

MJXCVT で作成された[MJX バイナリ ファイル](#)は、MJX440 コマンドの LOAD コマンドでダウンロードすることができます。

```
load myfile.mjk
```

付録 A 仕様

本体寸法	31mm(高さ) x 149mm(幅) x 186mm(奥行き)
本体重量	450g
電源 (AC アダプタ)	入力 AC100~240V 50Hz/60Hz 出力 DC 5V±5% 3.0A
ROM プローブ ケーブル	300mm
外部トリガ ケーブル	300mm
使用温度範囲	0°C~35°C
保存温度範囲	-10°C~55°C
周囲湿度範囲	30%~85%
対応 ROM	「付録 D ROM プローブ」 参照
対応 ROM 個数	8ビット 1 個、2 個、4 個 16ビット 1 個、2 個
エミュレーション メモリ容量	4M バイト(標準)
アクセス タイム	CS から 50n 秒
インターフェース	パラレル (PCI、PCMCIA カード)
ターゲット インターフェース	N-Wire コネクタ ROM ソケット
対応 CPU	NB85E Core
ダウンロード速度	480K バイト/秒*1 (N-Wire 接続、JTAG クロック25MHzの場合) 4M バイト/秒 (ROM インサーキット接続時)
対応デバッグ	Green Hills MULTI (対応 OS は Windows98/NT4.0/2000/XP)
ブレイクポイント機能	ハードウェア × 2 ソフトウェア × 128 イベントブレイク 命令 最大 8 アクセス 最大 4
トレース機能	トレース メモリ 48bit × 128K トレース クロック 66MHz max (CPU に依存) トレース条件 10 種 タイムスタンプ 32bit
制限事項	「付録 B ターゲット システムの制限事項」 参照

*1 ダウンロードの速さは JTAG クロックを高く設定する程、高速になります。

付録 B ターゲット システムの制限事項

MJX440 を使用するためには、ターゲット システムが次の条件を満足している必要があります。

- N-Wire 仕様のコネクタを実装している。

さらに、ROM インサーキット接続するためには、次の条件を満足している必要があります。

- ROM ソケットが実装されている。
- ROM がバンク化されていない。
- ROM が複数個実行されている場合、すべての ROM のアドレス信号が同じか、ROM のアドレスが連続している。

付録 C N-Wireコネクタ

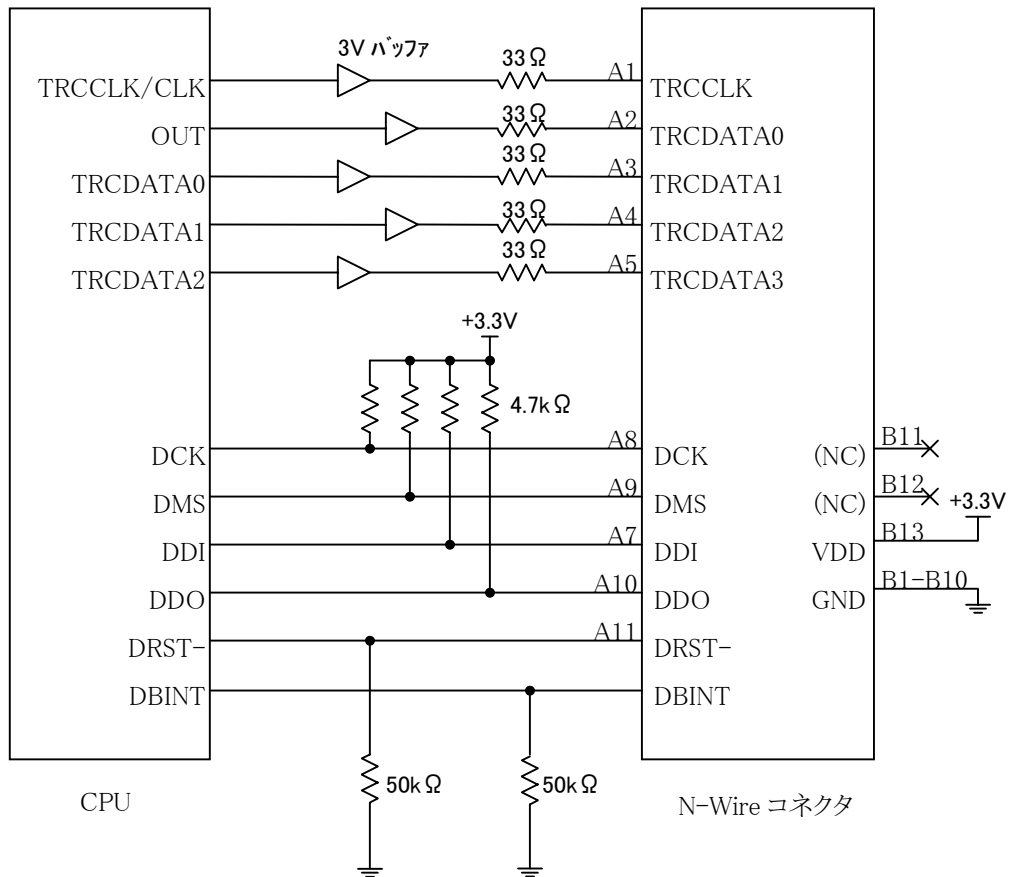
ピンアサイン

TRCCLK	A01	B01	GND
TRCDATA0	A02	B02	GND
TRCDATA1	A03	B03	GND
TRCDATA2	A04	B04	GND
TRCDATA3	A05	B05	GND
TRCEND	A06	B06	GND
DDI	A07	B07	GND
DCK	A08	B08	GND
DMS	A09	B09	GND
DDO	A10	B10	GND
DRST-	A11	B11	NC
PORT0	A12	B12	NC
PORT1	A13	B13	VDD

推奨コネクタ

- KEL 社製 8830E-026-170S (ストレート)
- KEL 社製 8830E-026-170L (ライトアングル)

ターゲット システムの推奨回路



注意事項

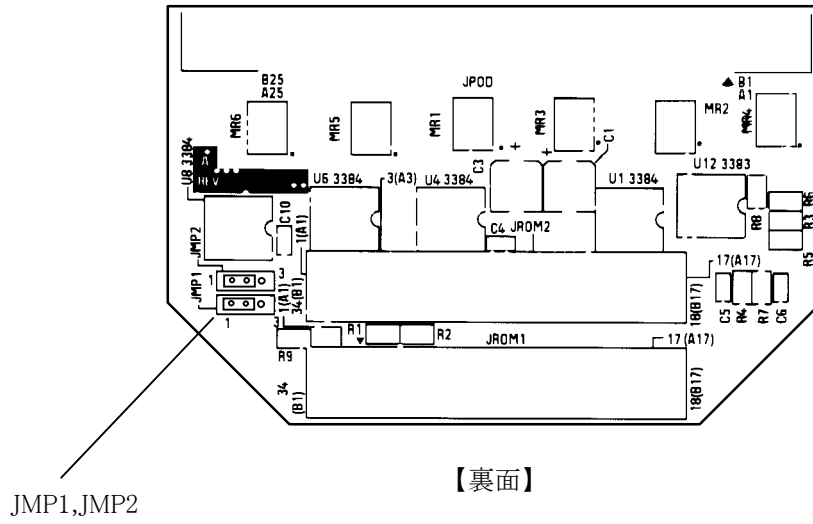
- CPU～N-Wire コネクタ間のパターンは、できるだけ短くしてください。(100mm 以下)
- TRCCLK/CLKOUT～TRCCLK 間と DCK～DCK 間のパターンは、GND でシールドしてください。
- MJX440 は、B13 VDD をターゲットプロービングバッファの電源に使用しています。

3V バッファ推奨品

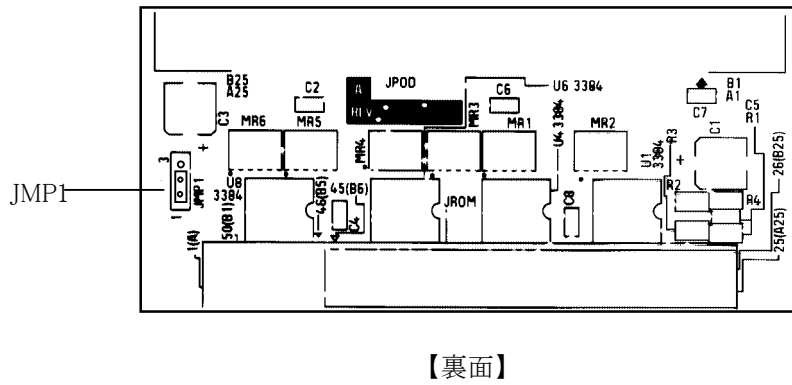
- TI 社製 SN74LVC541A
- 東芝社製 TC74LCX541F

付録D ROMプローブ

J-101A (32pin-8bit ROM用)

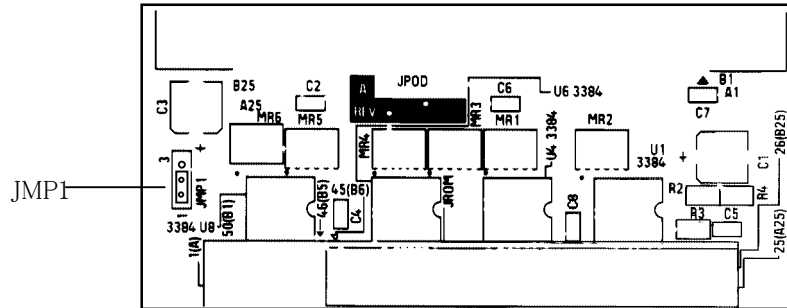


J-102A (40/42pin-16bit ROM用 - バスサイジング 16bit 固定)



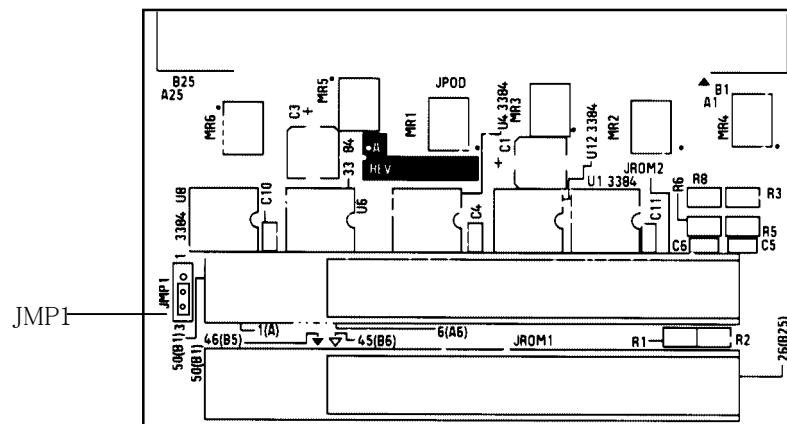
付録D ROMプロンプ

J-103A (40/42pin-16bit ROM用 - バスサイジング機能無し)



【裏面】

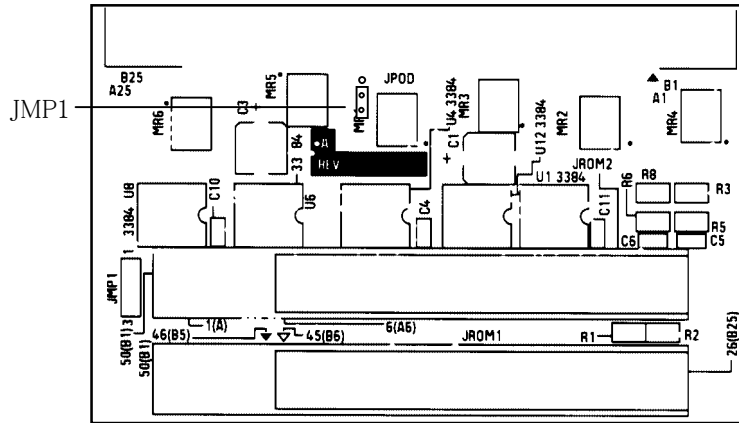
J-104A (40/42pin-16bit ROM用 - バスサイジング8bit固定)



【裏面】

付録D ROMプロンプ

J104(40/42pin-16bit ROM用)



【裏面】

付録 D ROM プローブ

ROM プローブ基板 J-101A ジャンパ設定:

ROM プラグ	ジャンパ設定	ROM 容量	対応 ROM	ROM メーカー
32pin (27010)	JMP1 1-2 ショート JMP2 2-3 ショート	128K x 8bit (0x20000byte)	HN27C101AG μ PD27C1001AD TC571000D TC571000AD TC57H1000AD M5M27C101K MBM27C1001-nnZ 27010 27C010 Am27C010	Hitachi NEC Toshiba Toshiba Toshiba Mitsubishi Fujitsu intel intel AMD
32pin (27020)	JMP1 1-2 ショート JMP2 1-2 ショート	256K x 8bit (0x40000byte)	μ PD27C2001D M5M27C201K Am27C020	NEC Mitsubishi AMD
32pin (27040)	JMP1 1-2 ショート JMP2 1-2 ショート	512K x 8bit (0x80000byte)	HN27C4001G μ PD27C4001DZ TC574000D TC574000DI M5M27C401K MBM27C4001-nnZ 27040 Am27C040	Hitachi NEC Toshiba Toshiba Mitsubishi Fujitsu intel AMD
32pin (27080)	JMP1 1-2 ショート JMP2 1-2 ショート	1024K x 8bit (0x100000byte)	Am27C080	AMD
32pin (271000)	JMP1 2-3 ショート JMP2 2-3 ショート	128K x 8bit	HN27C301AG μ PD27C1000AD TC571001D TC571001AD TC57H1001AD M5M27C100K MBM27C1000-nnZ	Hitachi NEC Toshiba Toshiba Toshiba Mitsubishi Fujitsu

付録 D ROM プローブ

ROM プローブ基板 J-102A ジャンパ設定

ROM プラグ	ジャンパ設定	ROM 容量	対応 ROM	ROM メーカー
40pin (27C4000 16bit)	JMP1 1-2 ショート	256K x 16bit (0x80000byte)	HN27C4000G Am27C400	Hitachi AMD
42pin (27C8000 16bit)	JMP1 2-3 ショート	512K x 16bit (0x100000byte)	μ PD27C8000 Am27C800	NEC AMD
42pin (27C16000 16bit)	JMP1 1-2 ショート	1024K x 16bit (0x200000byte)		

ROM プローブ基板 J-103A ジャンパ設定

ROM プラグ	ジャンパ設定	ROM 容量	対応 ROM	ROM メーカー
40pin (271024)	JMP1 2-3 ショート	64K x 16bit (0x20000byte)	HN27C1024HG μ PD27C1024D μ PD27C1024AD TC57H1024D TC57H1024AD MBM27C1024-nnZ 27210 27C210 Am27C1024	Hitachi NEC NEC Toshiba Toshiba Fujitsu intel intel AMD
40pin (272048)	JMP1 1-2 ショート	128K x 16bit (0x40000byte)	Am27C2048	AMD
40pin (274096)	JMP1 1-2 ショート	256K x 16bit (0x80000byte)	HN27C4096G HN27C4096HG HN27C4096AG HN27C4096AHG TC574096D MBM27C4096-nnZ 27240 Am27C4096	Hitachi Hitachi Hitachi Hitachi Toshiba Fujitsu intel AMD

付録D ROMプローブ

ROMプローブ基板 J-104A ジャンパ設定

ROM プラグ	ジャンパ設定	ROM 容量	対応 ROM	ROM メーカー
40pin (27C4000 8bit)	JMP1 1-2 ショート	512K x 8bit (0x80000byte)	HN27C4000G Am27C400	Hitachi AMD
42pin (27C8000 8bit)	JMP1 2-3 ショート	1024K x 8bit (0x100000byte)	μ PD27C8000 Am27C800	NEC AMD
42pin (27C16000 8bit)	JMP1 1-2 ショート	2048K x 8bit (0x200000byte)		

付録 E 対応 ROMピンアサイン

Vpp	1	32	Vcc
A16	2	31	PGM*
A15	3	30	NC
A12	4	29	A14
A7	5	28	A13
A6	6	27	A8
A5	7	26	A9
A4	8	25	A11
A3	9	24	OE*
A2	10	23	A10
A1	11	22	CE*
A0	12	21	I/O7
I/O0	13	20	I/O6
I/O1	14	19	I/O5
I/O2	15	18	I/O4
Vss	16	17	I/O3

J-101A 27010

Vpp	1	32	Vcc
A16	2	31	PGM*
A15	3	30	A17
A12	4	29	A14
A7	5	28	A13
A6	6	27	A8
A5	7	26	A9
A4	8	25	A11
A3	9	24	OE*
A2	10	23	A10
A1	11	22	CE*
A0	12	21	I/O7
I/O0	13	20	I/O6
I/O1	14	19	I/O5
I/O2	15	18	I/O4
Vss	16	17	I/O3

J-101A 27020

Vpp	1	32	Vcc
A16	2	31	A18
A15	3	30	A17
A12	4	29	A14
A7	5	28	A13
A6	6	27	A8
A5	7	26	A9
A4	8	25	A11
A3	9	24	OE*
A2	10	23	A10
A1	11	22	CE*
A0	12	21	I/O7
I/O0	13	20	I/O6
I/O1	14	19	I/O5
I/O2	15	18	I/O4
Vss	16	17	I/O3

J-101A 27040

A19	1	32	Vcc
A16	2	31	A18
A15	3	30	A17
A12	4	29	A14
A7	5	28	A13
A6	6	27	A8
A5	7	26	A9
A4	8	25	A11
A3	9	24	OE*
A2	10	23	A10
A1	11	22	CE*
A0	12	21	I/O7
I/O0	13	20	I/O6
I/O1	14	19	I/O5
I/O2	15	18	I/O4
Vss	16	17	I/O3

J-101A 27080

付録E 対応 ROMピンアサイン

Vpp	1	32	Vcc
OE*	2	31	PGM*
A15	3	30	NC
A12	4	29	A14
A7	5	28	A13
A6	6	27	A8
A5	7	26	A9
A4	8	25	A11
A3	9	24	A16
A2	10	23	A10
A1	11	22	CE*
A0	12	21	I/O7
I/O0	13	20	I/O6
I/O1	14	19	I/O5
I/O2	15	18	I/O4
Vss	16	17	I/O3

J-101A 271000

A17	1	40	A8
A7	2	39	A9
A6	3	38	A10
A5	4	37	A11
A4	5	36	A12
A3	6	35	A13
A2	7	34	A14
A1	8	33	A15
A0	9	32	A16
CE*	10	31	BYTE*/Vpp
Vss	11	30	Vss
OE*	12	29	I/O15/A-1
I/O0	13	28	I/O7
I/O8	14	27	I/O14
I/O1	15	26	I/O6
I/O9	16	25	I/O13
I/O2	17	24	I/O5
I/O10	18	23	I/O12
I/O3	19	22	I/O4
I/O11	20	21	Vcc

J-102A/J-104A
27C4000

A18	1	42	NC
A17	2	41	A8
A7	3	40	A9
A6	4	39	A10
A5	5	38	A11
A4	6	37	A12
A3	7	36	A13
A2	8	35	A14
A1	9	34	A15
A0	10	33	A16
CE*	11	32	BYTE*/Vpp
Vss	12	31	Vss
OE*	13	30	I/O15/A-1
I/O0	14	29	I/O7
I/O8	15	28	I/O14
I/O1	16	27	I/O6
I/O9	17	26	I/O13
I/O2	18	25	I/O5
I/O10	19	24	I/O12
I/O3	20	23	I/O4
I/O11	21	22	Vcc

J-102A/J-104A
27C8000

付録E 対応 ROMピンアサイン

A18	1	42	A19
A17	2	41	A8
A7	3	40	A9
A6	4	39	A10
A5	5	38	A11
A4	6	37	A12
A3	7	36	A13
A2	8	35	A14
A1	9	34	A15
A0	10	33	A16
CE*	11	32	BYTE*/Vpp
Vss	12	31	Vss
OE*	13	30	I/O15/A-1
I/O0	14	29	I/O7
I/O8	15	28	I/O14
I/O1	16	27	I/O6
I/O9	17	26	I/O13
I/O2	18	25	I/O5
I/O10	19	24	I/O12
I/O3	20	23	I/O4
I/O11	21	22	Vcc

J-102A/J-104A
27C16000

Vpp	1	40	Vcc
CE*	2	39	PGM*
I/O15	3	38	NC
I/O14	4	37	A15
I/O13	5	36	A14
I/O12	6	35	A13
I/O11	7	34	A12
I/O10	8	33	A11
I/O9	9	32	A10
I/O8	10	31	A9
Vss	11	30	Vss
I/O7	12	29	A8
I/O6	13	28	A7
I/O5	14	27	A6
I/O4	15	26	A5
I/O3	16	25	A4
I/O2	17	24	A3
I/O1	18	23	A2
I/O0	19	22	A1
OE*	20	21	A0

J-103A 271024

Vpp	1	40	Vcc
CE*	2	39	PGM*
I/O15	3	38	A16
I/O14	4	37	A15
I/O13	5	36	A14
I/O12	6	35	A13
I/O11	7	34	A12
I/O10	8	33	A11
I/O9	9	32	A10
I/O8	10	31	A9
Vss	11	30	Vss
I/O7	12	29	A8
I/O6	13	28	A7
I/O5	14	27	A6
I/O4	15	26	A5
I/O3	16	25	A4
I/O2	17	24	A3
I/O1	18	23	A2
I/O0	19	22	A1
OE*	20	21	A0

J-103A 272048

付録E 対応 ROMピンアサイン

Vpp	1	40	Vcc
CE*	2	39	A17
I/O15	3	38	A16
I/O14	4	37	A15
I/O13	5	36	A14
I/O12	6	35	A13
I/O11	7	34	A12
I/O10	8	33	A11
I/O9	9	32	A10
I/O8	10	31	A9
Vss	11	30	Vss
I/O7	12	29	A8
I/O6	13	28	A7
I/O5	14	27	A6
I/O4	15	26	A5
I/O3	16	25	A4
I/O2	17	24	A3
I/O1	18	23	A2
I/O0	19	22	A1
OE*	20	21	A0

J-103A 274096

付録 F LED

PWR (緑)	電源が供給されているとき点灯
JTAG (赤)	N-Wire コネクタの電源が供給されているとき点灯
EXI1 (赤)	外部トリガ ケーブルの EXTIN1 信号が HIGH レベルのとき点灯
EXI2 (赤)	外部トリガ ケーブルの EXTIN2 信号が HIGH レベルのとき点灯
EXI3 (赤)	外部トリガ ケーブルの EXTIN3 信号が HIGH レベルのとき点灯
ROM1 (赤)	図 3-5-1～3-5-9 を参照
ROM2 (赤)	同上
ROM3 (赤)	同上
ROM4 (赤)	同上

付録 G レジスタ名一覧

(汎用レジスタ)

R0	R16	PC
R1	R17	PSW
R2	R18	EIPC
R3	R19	EIPSW
R4	R20	FEPC
R5	R21	FEPSW
R6	R22	CTPC
R7	R23	CTPSW
R8	R24	ECR
R9	R25	CTBP
R10	R26	ASID
R11	R27	
R12	R28	
R13	R29	
R14	R30	
R15	R31	

付録 H MJX バイナリ ファイル

バイナリ ファイル構成

ヘッダ
データ
ヘッダ
データ
⋮
ヘッダ
データ
終了ヘッダ

ヘッダ (16 バイト長)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
'M'	'J'	'1'	00	len1	len2	len3	00	00	00	00	adr1	adr2	adr3	adr4	00

len1: データ バイト長 (MSB)

len2: データ バイト長

len3: データ バイト長 (LSB)

adr1: 論理アドレス (MSB)

adr2: 論理アドレス

adr3: 論理アドレス

adr4: 論理アドレス (LSB)

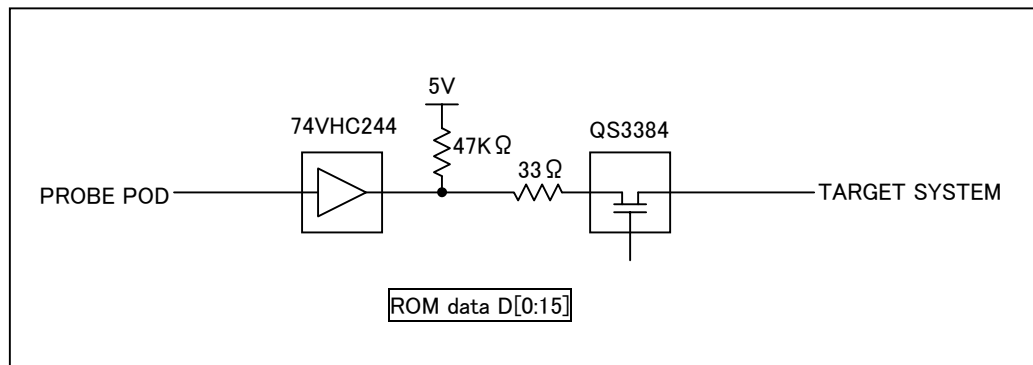
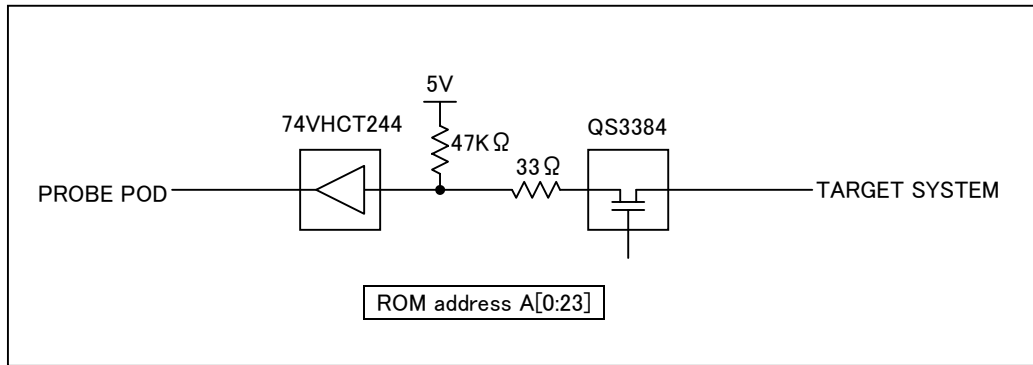
データ (可変、データ バイト長)

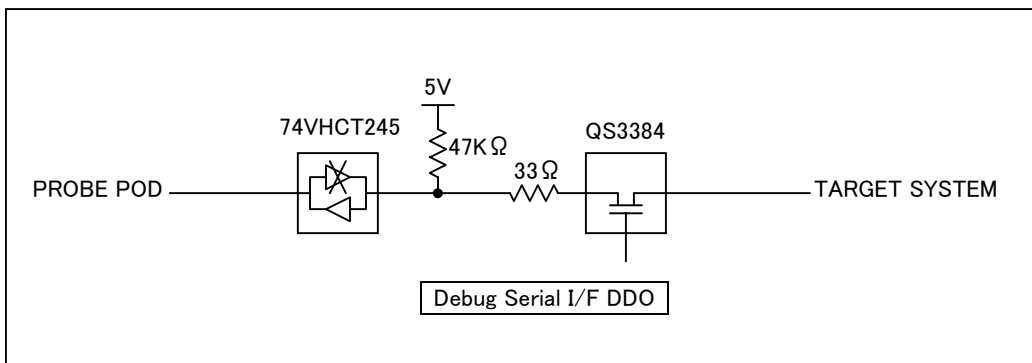
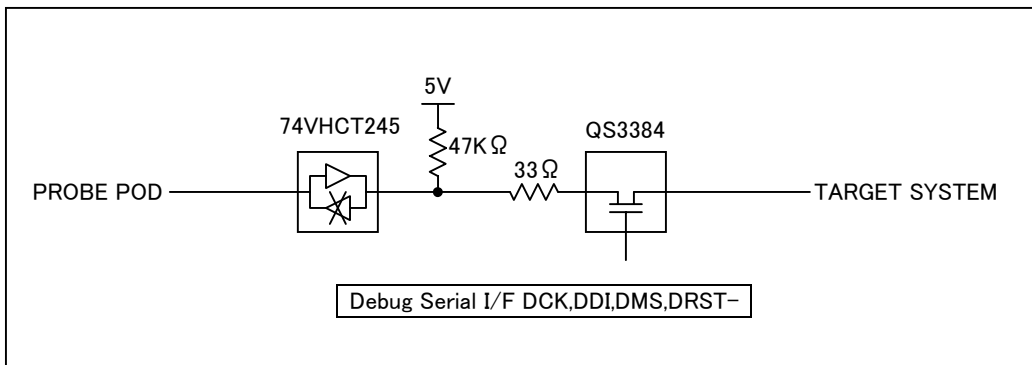
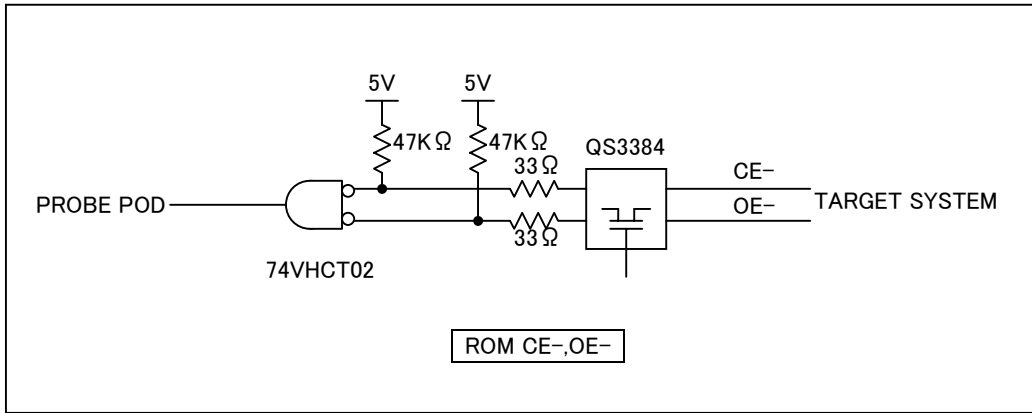
XX	XX	XX	XX	XX	XX	XX	⋮	XX	XX	XX	XX	XX	XX	XX	XX
----	----	----	----	----	----	----	---	----	----	----	----	----	----	----	----

終了ヘッダ (16 バイト長)

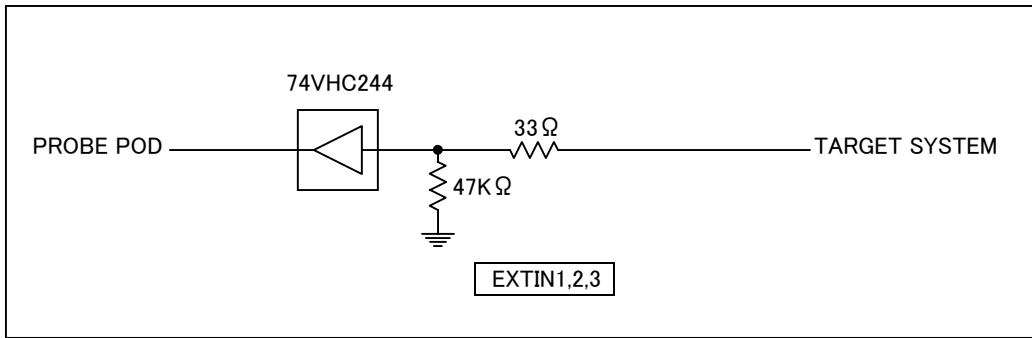
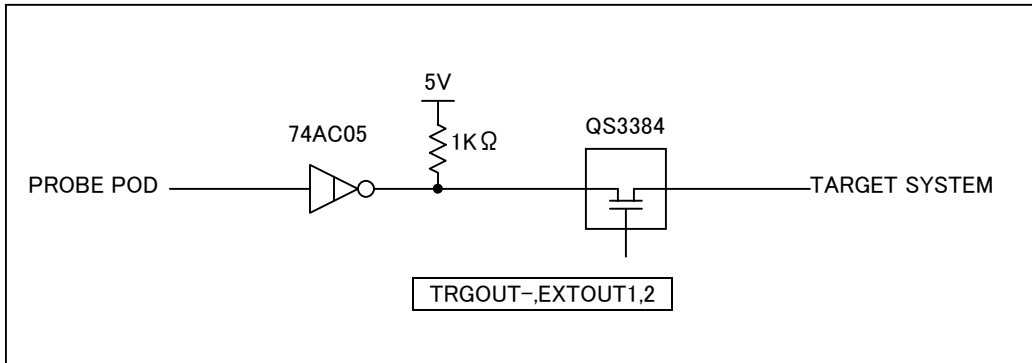
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
'M'	'J'	'1'	00	00	00	00	00	00	00	00	00	00	00	00	00

付録I ターゲット システムへのプロービング

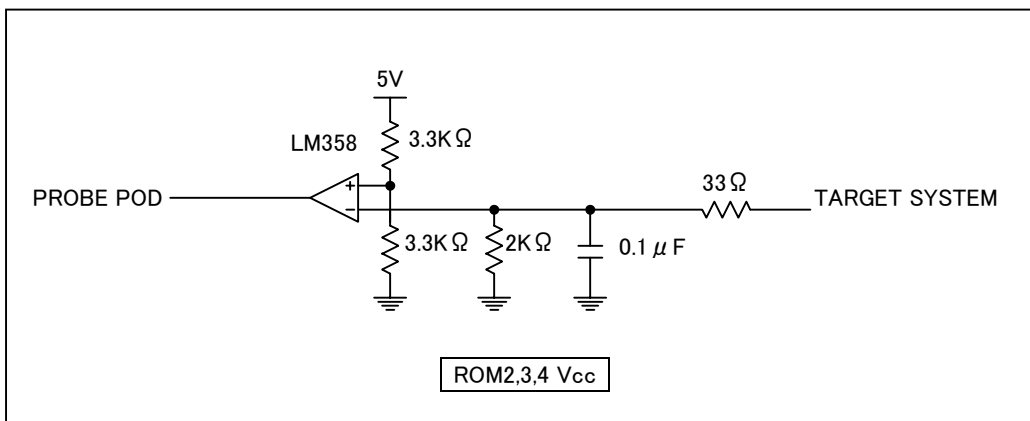
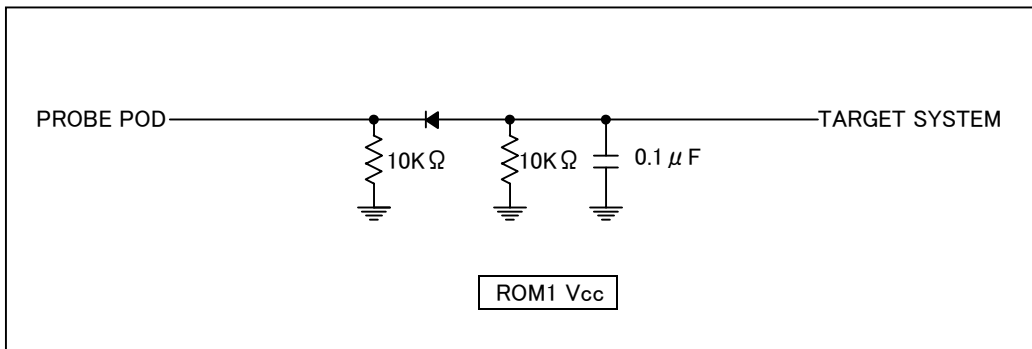
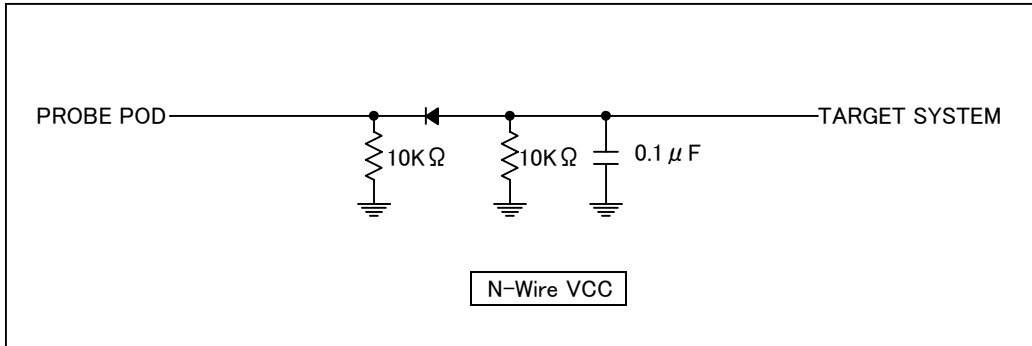




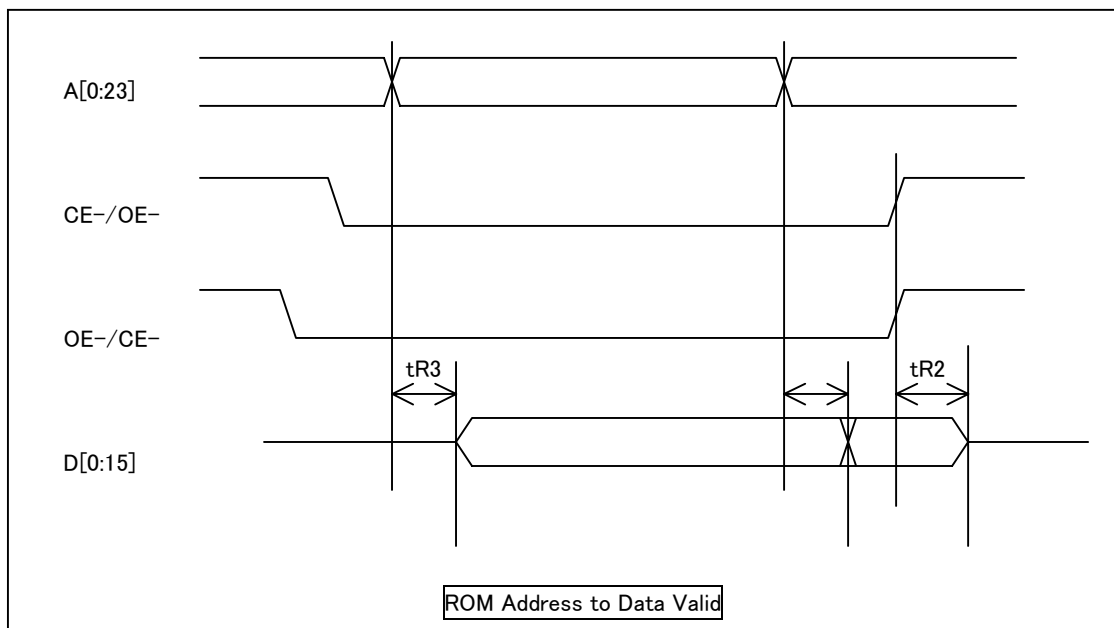
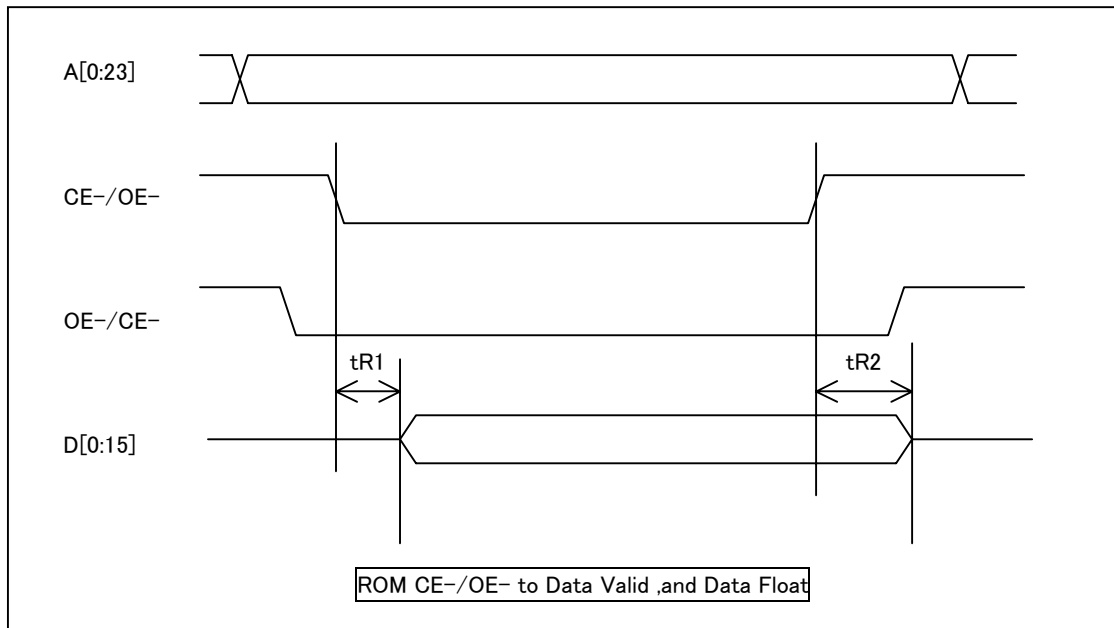
付録I ターゲット システムへのプロービング



付録I ターゲット システムへのプロービング

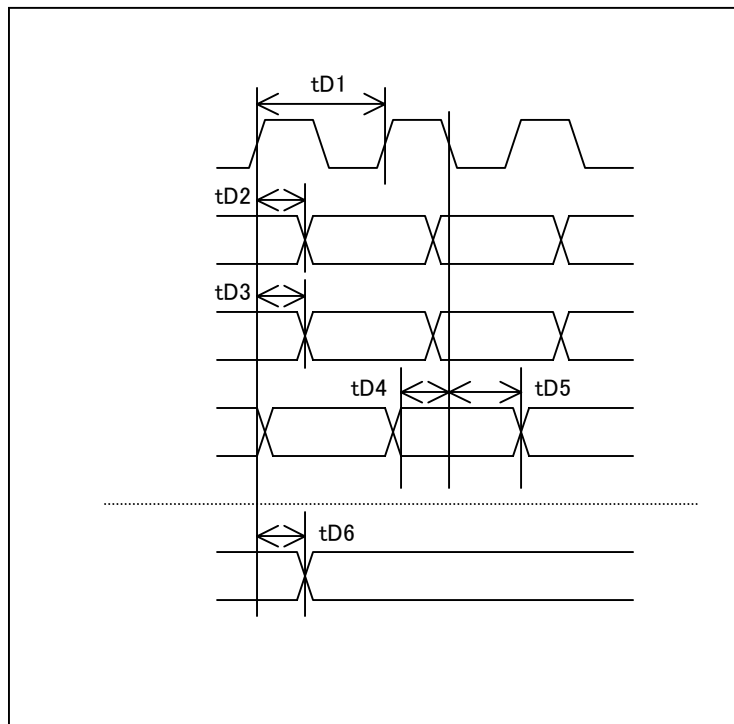


符号	項 目	PROBE POD(TYP)
tR1	CE-/OE- to Data Valid Delay	50nS
tR2	CE-/OE- to Data Float Delay	30nS
tR3	Address to Data Valid Delay	50nS



付録I ターゲット システムへのプロービング

符号	項 目	PROBE POD		
		MIN	TYP	MAX
tD1	DCK Clock Period		40	
tD2	DDI Valid Delay	5	12	35
tD3	DMS Valid Delay	5	12	35
tD4	DDO in Setup Time	20		
tD5	DDO in Hold Time	0		
tD6	DRST- Active Delay		12	



付録 J フラッシュ メモリ デバイス情報定義 ファイル レイアウト

{MjxDebw.exe の格納ディレクトリ}\zax_mjx\flash\device の下に、{型式}×{構成}.dat のファイル名で作成します。

ファイル名の例は

{型式}×{構成}.dat ファイル名の例

【型式、構成の例】

型式	構成
AM29DS323DT	2M×16
AM29DS163DB	2M×8 / 1M×16

- 型式 AM29DS323DT、構成 2M×16 の場合
AM29DS323DT には 16bit 構成しかないので
AM29DS323DT.dat
となります。
- 型式 AM29DS163DB、構成 2M×8 / 1M×16 の場合
AM29DS163DB には 8bit 構成と 16bit 構成の 2 タイプがあるので
AM29DS163DBx8.dat
AM29DS163DBx16.dat
の 2 ファイルになります。

ファイル レイアウトは下記の通りです。一行に複数の項目を記述するときは、項目を空白文字で区切ります。

行	項目	
1	マニファクチャ コード ^(注1)	チップ消去機能無フラグ ^(注2)
2	デバイス コード ^(注3)	
3	アクセス モード ^(注4)	コマンド・インターフェース ^(注5)
4	コマンド入力 第1バス サイクル 書き込み アドレス ^(注6)	
5	コマンド入力 第2バス サイクル 書き込み アドレス ^(注6)	
6	セクタ数 ^(注7)	
7	第 1 ~ n セクタ サイズ ^(注8)	セクタ数 ^(注9)
	...	
	...	
-	第 i ~ 最後のセクタ サイズ	セクタ数

注1: 16 進数で入力します。

注2: チップ消去機能がないときは 1、あるときは 0 もしくは、空白を入力します。

注3: 16 進数で入力します。

注4: 00: アクセス モードが 8 ビット

10: アクセス モードが 8/16 ビット選択可で、8 ビット アクセス

11: アクセス モードが 8/16 ビット選択可で、16 ビット アクセス

注5: コマンド・インターフェースの種別を入力します。

0: JEDEC 標準型コマンドと互換 (AMD タイプ)

1: インテル系でUI (コマンド・ユーザ・インターフェース) を使用 (インテル タイプ)

注6: コマンド(セクタ イレース、チップ イレース、プログラムなど)を入力するときのコマンド入力アドレスを 16 進数で指定します。

インテル系のデバイスへの書き込み・消去では使用しません。0 を入力しておいてください。次ページに、AMD AM29F160DB (2 M x 8-Bit、または 1M x 16-Bit 構成)での例を示します。

コマンド		第 1 バス サイクル		第 2 バス サイクル		第 3 バス サイクル		第 4 バス サイクル		第 5 バス サイクル		第 6 バス サイクル	
		アドレス	データ	アドレス	データ	アドレス	データ	アドレス	データ	アドレス	データ	アドレス	データ
書き込み	16 ビット	555	AA	2AA	55	555	A0	PA	PD	555	55	—	—
	8 ビット	AAA		555		AAA		—		—			
チップ 消去	16 ビット	555	AA	2AA	55	555	80	555	AA	555	55	555	10
	8 ビット	AAA		555		AAA		555		AAA			
セクタ 消去	16 ビット	555	AA	2AA	55	555	80	555	AA	2AA	55	SA	30
	8 ビット	AAA		55		AAA		555		555			

表の「書き込み」シーケンスにしたがって、指定アドレスにデータを入力することで、フラッシュ メモリへデータを書き込みすることができます。

PA、PD は実際に書き込みを行うアドレスとデータです。また SA は消去したいセクタのアドレスです。

デバイス情報定義ファイルの「コマンド入力 第1バス サイクル 書き込み アドレス」には表の「第1バス サイクル アドレス(55 または AAA)」を、「コマンド入力 第2バス サイクル 書き込み アドレス」には「第2バス サイクル アドレス(2AA または 555)」を指定します。

16 ビット モードの場合には、16 ビット アドレスング モードで入力します。

詳しくは、各デバイスのデータ シートを参照してください。

注 7: セクタの総数を 10 進数で入力します。

注 8: セクタ サイズとセクタ数を入力します。セクタ サイズは 16 進数で、セクタ数は 10 進数で指定します。

セクタ数が 1 の場合は、指定を省略できます。

セクタ サイズはバイト単位で指定します。

例えば、サイズ 0x2000 のセクタが 8 個連続していて、その後、0x10000 のセクタが 15 連続している構成の場合は次のように指定します。

```
2000 8
10000 15
```


以下に AMD AM29F160DB の デバイス情報定義ファイルのサンプルを示します。

2Mx8-Bit 構成: AM29F160DBx8.dat、1Mx16-Bit 構成: AM29F160DBx16.dat。

AM29F160DB x8. dat	AM29F160DB x16. dat	注
0 1	0 1	マニファクチャ コード
d 8	2 2 d 8	デバイス コード
1 0	1 1	アクセス モード
a a a	5 5 5	第 1 バス サイクル 書き込み アドレス
5 5 5	2 a a	第 2 バス サイクル 書き込み アドレス
3 5	3 5	セクタ総数
4 0 0 0	4 0 0 0	第 1 セクタ サイズ
2 0 0 0	2 0 0 0	第 2 セクタ サイズ
2 0 0 0	2 0 0 0	セクタ数を省略 (= 1)
8 0 0 0	8 0 0 0	
1 0 0 0 0 3 1	1 0 0 0 0 3 1	第 3 - 35 セクタ サイズ

チップ消去機能無フラグ、および、コマンド・インターフェースの項は既定値を使用するというで入力していません (0: 機能有、0: JEDEC 互換)。

次に インテル 28F640K3 の デバイス情報定義ファイルのサンプルを示します。

28F640K3.dat (16-Bit 構成であるので、ファイル名に"x16" はつきません。)

28F640K3. dat	注
8 9 1	マニファクチャ コード、チップ消去機能無フラグ
8 8 0 1	デバイス コード
1 1 1	アクセス モード、コマンド・インターフェース
0	第 1 バス サイクル 書き込み アドレス
0	第 2 バス サイクル 書き込み アドレス
6 4	セクタ総数
2 0 0 0 0 6 4	第 1-64 セクタ サイズ

チップ消去機能無フラグに 1、コマンド・インターフェースの項にも 1 が設定されています。

第 1、2 バス サイクル 書き込み アドレスの項にはいずれも 0 を設定しています。

付録 K 対応フラッシュ メモリー一覧

1. AMD

型式	構成	電源電圧
AM29BDS323DT	2Mx16	1.8V
AM29DS163DB	2Mx8/1Mx16	1.8V
AM29DS163DT	2Mx8/1Mx16	1.8V
AM29DS323DB	4Mx8/2Mx16	1.8V
AM29DS323DT	4Mx8/2Mx16	1.8V
AM29SL800CB	1Mx8/512Kx16	1.8V
AM29SL800CT	1Mx8/512Kx16	1.8V
AM29SL160CB	2Mx8/1Mx16	1.8V
AM29SL160CT	2Mx8/1Mx16	1.8V
AM29DL400BB	512Kx8/256Kx16	3V
AM29DL400BT	512Kx8/256Kx16	3V
AM29DL800BB	1Mx8/512Kx16	3V
AM29DL800BT	1Mx8/512Kx16	3V
AM29DL161DB	2Mx8/1Mx16	3V
AM29DL161DT	2Mx8/1Mx16	3V
AM29DL162DB	2Mx8/1Mx16	3V
AM29DL162DT	2Mx8/1Mx16	3V
AM29DL163DB	2Mx8/1Mx16	3V
AM29DL163DT	2Mx8/1Mx16	3V
AM29DL164DB	2Mx8/1Mx16	3V
AM29DL164DT	2Mx8/1Mx16	3V
AM29DL322DB	4Mx8/2Mx16	3V
AM29DL322DT	4Mx8/2Mx16	3V
AM29DL323DB	4Mx8/2Mx16	3V
AM29DL323DT	4Mx8/2Mx16	3V
AM29DL324DB	4Mx8/2Mx16	3V
AM29DL324DT	4Mx8/2Mx16	3V
AM29DL802CB	512Kx16	3V

続き

型式	構成	電源電圧
AM29BL162CB	1Mx16	3V
AM29PL160CB	2Mx8/1Mx16	3V
AM29LV200BB	256Kx8/128Kx16	3V
AM29LV200BT	256Kx8/128Kx16	3V
AM29LV400BB	512Kx8/256Kx16	3V
AM29LV400BT	512Kx8/256Kx16	3V
AM29LV800BB	1Mx8/512Kx16	3V
AM29LV800BT	1Mx8/512Kx16	3V
AM29LV160BB	2Mx8/1Mx16	3V
AM29LV160BT	2Mx8/1Mx16	3V
AM29LV160DB	2Mx8/1Mx16	3V
AM29LV160DT	2Mx8/1Mx16	3V
AM29LV320DB	4Mx8/2Mx16	3V
AM29LV320DT	4Mx8/2Mx16	3V
AM29LV640D	4Mx16	3V
AM29LV641D	4Mx16	3V
AM29LV001BB	128Kx8	3V
AM29LV001BT	128Kx8	3V
AM29LV002BB	256Kx8	3V
AM29LV002BT	256Kx8	3V
AM29LV004BB	512Kx8	3V
AM29LV004BT	512Kx8	3V
AM29LV008BB	1Mx8	3V
AM29LV008BT	1Mx8	3V
AM29LV116DB	2Mx8	3V
AM29LV116DT	2Mx8	3V
AM29LV010BB	128Kx8	3V
AM29LV040B	512Kx8	3V
AM29LV081B	1Mx8	3V
AM29LV017D	2Mx8	3V
AM29LV033C	4Mx8	3V
AM29LV065D	8Mx8	3V

続き

型式	構成	電源電圧
AM29F002BB	256Kx8	5V
AM29F002BT	256Kx8	5V
AM29F002NBB	256Kx8	5V
AM29F002NBT	256Kx8	5V
AM29F004BB	512Kx8	5V
AM29F004BT	512Kx8	5V
AM29F200BB	256Kx8/128Kx16	5V
AM29F200BT	256Kx8/128Kx16	5V
AM29F400BB	512Kx8/256Kx16	5V
AM29F400BT	512Kx8/256Kx16	5V
AM29F800BB	1Mx8/512Kx16	5V
AM29F800BT	1Mx8/512Kx16	5V
AM29F160DB	2Mx8/1Mx16	5V
AM29F160DT	2Mx8/1Mx16	5V
AM29F010B	128Kx8	5V
AM29F040B	512Kx8	5V
AM29F080B	1Mx8	5V
AM29F016D	2Mx8	5V
AM29F017D	2Mx8	5V
AM29F032B	4Mx8	5V

2. ATMEL

型式	構成	電源電圧
AT49F001	128Kx8	5V
AT49F001T	128Kx8	5V
AT49F001N	128Kx8	5V
AT49F001NT	128Kx8	5V
AT49BV001	128Kx8	3V
AT49BV001T	128Kx8	3V
AT49BV001N	128Kx8	3V
AT49BV001NT	128Kx8	3V
AT49LV001	128Kx8	3V
AT49LV001T	128Kx8	3V
AT49LV001N	128Kx8	3V
AT49LV001NT	128Kx8	3V
AT49F002	256Kx8	5V
AT49F002T	256Kx8	5V
AT49F002N	256Kx8	5V
AT49F002NT	256Kx8	5V
AT49BV002	256Kx8	3V
AT49BV002T	256Kx8	3V
AT49BV002N	256Kx8	3V
AT49BV002NT	256Kx8	3V
AT49LV002	256Kx8	3V
AT49LV002T	256Kx8	3V
AT49LV002N	256Kx8	3V
AT49LV002NT	256Kx8	3V
AT49F2048	128Kx16	5V
AT49F2048A	256Kx8/128x16	5V
AT49LV2048	128Kx16	3V
AT49LV2048A	256Kx8/128x16	3V
AT49BV2048A	256Kx8/128x16	3V
AT49F4096A	512Kx8/256Kx16	5V

続き

型式	構成	電源電圧
AT49BV4096A	512Kx8/256Kx16	3V
AT49LV4096A	512Kx8/256Kx16	3V
AT49BV4096	256Kx16	3V
AT49LV4096	256Kx16	3V
AT49F4096	256Kx16	5V
AT49F008A	1Mx8	5V
AT49F008AT	1Mx8	5V
AT49F8192A	1Mx8/512Kx16	5V
AT49F8192AT	1Mx8/512Kx16	5V
AT49BV008A	1Mx8	3V
AT49BV008AT	1Mx8	3V
AT49BV8192A	1Mx8/512Kx16	3V
AT49BV8192AT	1Mx8/512Kx16	3V
AT49LV8192A	1Mx8/512Kx16	3V
AT49F8192	512Kx16	5V
AT49F8192T	512Kx16	3V
AT49F8011	1Mx8/512Kx16	5V
AT49F8011T	1Mx8/512Kx16	5V
AT49BV8192	512Kx16	3V
AT49BV8192T	512Kx16	3V
AT49LV8192	512Kx16	3V
AT49LV8192T	512Kx16	3V
AT49BV8011	1Mx8/512Kx16	3V
AT49BV8011T	1Mx8/512Kx16	3V
AT49LV8011	1Mx8/512Kx16	3V
AT49LV8011T	1Mx8/512Kx16	3V
AT49BV160	2Mx8/1Mx16	3V
AT49BV160T	2Mx8/1Mx16	3V
AT49LV160	2Mx8/1Mx16	3V
AT49LV160T	2Mx8/1Mx16	3V

続き

型式	構成	電源電圧
AT49LV161	2Mx8/1Mx16	3V
AT49LV161T	2Mx8/1Mx16	3V
AT49BV1604	2Mx8/1Mx16	3V
AT49BV1604T	2Mx8/1Mx16	3V
AT49BV1614	2Mx8/1Mx16	3V
AT49BV1614T	2Mx8/1Mx16	3V
AT49BV1604A	2Mx8/1Mx16	3V
AT49BV1604AT	2Mx8/1Mx16	3V
AT49BV1614A	2Mx8/1Mx16	3V
AT49BV1614AT	2Mx8/1Mx16	3V
AT49LV1614A	2Mx8/1Mx16	3V
AT49LV1614AT	2Mx8/1Mx16	3V
AT49BV320	4Mx8/2Mx16	3V
AT49BV320T	4Mx8/2Mx16	3V
AT49BV321	4Mx8/2Mx16	3V
AT49BV321T	4Mx8/2Mx16	3V
AT49LV320	4Mx8/2Mx16	3V
AT49LV320T	4Mx8/2Mx16	3V
AT49LV321	4Mx8/2Mx16	3V
AT49LV321T	4Mx8/2Mx16	3V

3. ST

型式	構成	電源電圧
M29F010B	1Mx8	5V
M29F102BB	1Mx16	5V
M29F002BT	2Mx8	5V
M29F002BB	2Mx8	5V
M29F002BNT	2Mx8	5V
M29F200BT	2Mx8/1Mx16	5V
M29F200BB	2Mx8/1Mx16	5V
M29F040B	4Mx8	5V
M29F400BT	4Mx8/2Mx16	5V
M29F400BB	4Mx8/2Mx16	5V
M29F080A	8Mx8	5V
M29F800AT	8Mx8/4Mx16	5V
M29F800AB	8Mx8/4Mx16	5V
M29F016D	16Mx8	5V
M29W010B	1Mx8	3V
M29W102BT	1Mx16	3V
M29W102BB	1Mx16	3V
M29W022BT	2Mx8	3V
M29W022BB	2Mx8	3V
M29W200BT	2Mx8/1Mx16	3V
M29W200BB	2Mx8/1Mx16	3V
M29W004BT	4Mx8	3V
M29W004BB	4Mx8	3V
M29W040B	4Mx8	3V
M29W400BT	4Mx8/2Mx16	3V
M29W400BB	4Mx8/2Mx16	3V
M29W008AT	8Mx8	3V
M29W008AB	8Mx8	3V
M29W800AT	8Mx8/4Mx16	3V
M29W800AB	8Mx8/4Mx16	3V
M29W1600AT	16Mx8/8Mx16	3V
M29W1600B	16Mx8/8Mx16	3V

4. 富士通

型式	構成	電源電圧
MBM29F002BC	256Kx8	5V
MBM29F002TC	256Kx8	5V
MBM29F200BC	256Kx8/128Kx16	5V
MBM29F200TC	256Kx8/128Kx16	5V
MBM29F040C	512Kx8	5V
MBM29F004BC	512Kx8	5V
MBM29F004TC	512Kx8	5V
MBM29F400BC	512Kx8/256Kx16	5V
MBM29F400TC	512Kx8/256Kx16	5V
MBM29F080A	1Mx8	5V
MBM29F800BA	1Mx8/512Kx16	5V
MBM29F800TA	1Mx8/512Kx16	5V
MBM29F017A	2Mx8	5V
MBM29F016A	2Mx8	5V
MBM29F160BE	2Mx8/1Mx16	5V
MBM29F160TE	2Mx8/1Mx16	5V
MBM29F033C	4Mx8	5V
MBM29LV002BC	256Kx8	3V
MBM29LV002TC	256Kx8	3V
MBM29LV200BC	256Kx8/128Kx16	3V
MBM29LV200TC	256Kx8/128Kx16	3V
MBM29LV004BC	512Kx8	3V
MBM29LV004TC	512Kx8	3V
MBM29LV400BC	512Kx8/256Kx16	3V
MBM29LV400TC	512Kx8/256Kx16	3V
MBM29DL400BC	512Kx8/256Kx16	3V
MBM29DL400TC	512Kx8/256Kx16	3V
MBM29LV008BA	1Mx8	3V
MBM29LV008TA	1Mx8	3V
MBM29LV080A	1Mx8	3V
MBM29LV800BA	1Mx8/512Kx16	3V
MBM29LV800TA	1Mx8/512Kx16	3V

続き

型式	構成	電源電圧
MBM29DL800BA	1Mx8/512Kx16	3V
MBM29DL800TA	1Mx8/512Kx16	3V
MBM29LV800BE	1Mx8/512Kx16	3V
MBM29LV800TE	1Mx8/512Kx16	3V
MBM29LV016B	2Mx8	3V
MBM29LV016T	2Mx8	3V
MBM29LV017	2Mx8	3V
MBM29LV160B	2Mx8/1Mx16	3V
MBM29LV160T	2Mx8/1Mx16	3V
MBM29PL160BD	2Mx8/1Mx16	3V
MBM29PL160TD	2Mx8/1Mx16	3V
MBM29DL161BD	2Mx8/1Mx16	3V
MBM29DL161TD	2Mx8/1Mx16	3V
MBM29DL162BD	2Mx8/1Mx16	3V

5. INTEL

型式	構成	電源電圧
28F008SA	1Mx8	5V
28F008B3T	1Mx8	3V
28F008B3B	1Mx8	3V
28F800B3T	512Kx16	3V
28F800B3B	512Kx16	3V
28F016B3T	2Mx8	3V
28F016B3B	2Mx8	3V
28F160B3T	1Mx16	3V
28F160B3B	1Mx16	3V
28F004S5	512Kx8	5V
28F008S5	1Mx8	5V
28F016S5	2Mx8	5V
28F016SA	2Mx8	5V
28F004S3	512Kx8	3V
28F008S3	1Mx8	3V
28F016S3	2Mx8	3V
28F004B5T	512Kx8	5V
28F004B5B	512Kx8	5V
28F200B5T	256Kx8/128Kx16	5V
28F200B5B	256Kx8/128Kx16	5V
28F400B5T	512Kx8/256Kx16	5V
28F400B5B	512Kx8/256Kx16	5V
28F800B5T	1Mx8/512Kx16	5V
28F800B5B	1Mx8/512Kx16	5V
28F320J5	1Mx8/2Mx16	5V
28F640J5	8Mx8/4Mx16	5V
28F160S3	2Mx8/1Mx16	3V
28F320S3	4Mx8/2Mx16	3V
28F160S5	2Mx8/1Mx16	5V
28F320S5	4Mx8/2Mx16	5V
28F800F3T	512Kx16	3V
28F800F3B	512Kx16	3V
28F160B3T	1Mx16	3V

続き

型式	構成	電源電圧
28F160F3B	1Mx16	3V
28F800C3T	512Kx16	3V
28F800C3B	512Kx16	3V
28F160C3T	1Mx16	3V
28F160C3B	1Mx16	3V
28F320C3T	2Mx16	3V
28F320C3B	2Mx16	3V
28F640C3T	4Mx16	3V
28F640C3B	4Mx16	3V
28F320J3A	4Mx8/2Mx16	3V
28F640J3A	8Mx8/4Mx16	3V
28F128J3A	16Mx8/8Mx16	3V
28F640K3	4Mx16	3V
28F128K3	8Mx16	3V
28F256K3	16Mx16	3V
28F640K18	4Mx16	1.8V
28F128K18	8Mx16	1.8V
28F256K18	16Mx16	1.8V

6. シャープ

型式	構成	電源電圧
LH28F004BVT-TL85	512Kx8	Smart Voltage
LH28F004BVT-BL85	512Kx8	Smart Voltage
LH28F008SCN-L12	1Mx8	Smart Voltage
LH28F008SCT-L85	1Mx8	Smart Voltage
LH28F008SCHT-L85	1Mx8	Smart Voltage
LH28F008SCHT-L12	1Mx8	Smart Voltage
LH28F008BJT-BTLZ1	1Mx8	3V
LH28F016SCT-L95	2Mx8	Smart Voltage
LH28F016SCN-L12	2Mx8	Smart Voltage
LH28F016SCT-L12	2Mx8	Smart Voltage
LH28F016SCHT-L95	2Mx8	Smart Voltage
LH28F160BJE-TTL90	2Mx8/1Mx16	3V
LH28F160BJHE-TTL90	2Mx8/1Mx16	3V
LH28F160BJE-BTL90	2Mx8/1Mx16	3V
LH28F160BJHE-BTL90	2Mx8/1Mx16	3V
LH28F160BJB-TTL90	2Mx8/1Mx16	3V
LH28F160BJHB-TTL90	2Mx8/1Mx16	3V
LH28F160BJB-BTL90	2Mx8/1Mx16	3V
LH28F160BJHB-BTL90	2Mx8/1Mx16	3V
LH28F160BJHB-BTL90	1Mx16	3V
LH28F160S3NS-L10	2Mx8/1Mx16	3V
LH28F160S3HT-L10A	2Mx8/1Mx16	3V
LH28F160S3HT-L10A	2Mx8/1Mx16	3V
LH28F160S3HB-L10A	2Mx8/1Mx16	3V
LH28F160S3B-L10A	2Mx8/1Mx16	3V
LH28F160S3HNS-L10	2Mx8/1Mx16	3V
LH28F320BFE-PTTL70	2Mx16	3V
LH28F320BFE-PTTL70	2Mx16	3V
LH28F320BFE-PTTL80	2Mx16	3V
LH28F320BFHE-PTTL80	2Mx16	3V
LH28F320BFE-PBTL80	2Mx16	3V
LH28F320BFHE-PBTL80	2Mx16	3V
LH28F320BFHE-PTTLZ1	2Mx16	3V

付録K 対応フラッシュ メモリ一覧

続き

型式	構成	電源電圧
LH28F320BFHE-PBTLZ2	2Mx16	3V
LH28F320BFN-PTTLZH	2Mx16	3V
LH28F320BJE-PTTL90	4Mx8/2Mx16	3V
LH28F320BJHE-PTTL90	4Mx8/2Mx16	3V
LH28F320BJE-PBTL90	4Mx8/2Mx16	3V
LH28F320BJHE-PBTL90	4Mx8/2Mx16	3V
LH28F320BJB-PTTL90	4Mx8/2Mx16	3V
LH28F320BJHB-PTTL90	4Mx8/2Mx16	3V
LH28F320BJB-PBTL90	4Mx8/2Mx16	3V
LH28F320BJHB-PBTL90	4Mx8/2Mx16	3V
LH28F320BJD-TTL80	2Mx16	3V
LH28F400BVE-TL85	512Kx8/256Kx16	Smart Voltage
LH28F400BVE-TL12	512Kx8/256Kx16	Smart Voltage
LH28F400BVHE-TL85	512Kx8/256Kx16	Smart Voltage
LH28F400BVHE-TL12	512Kx8/256Kx16	Smart Voltage
LH28F400BVN-TL85	512Kx8/256Kx16	Smart Voltage
LH28F400BVN-TL12	512Kx8/256Kx16	Smart Voltage
LH28F400BVE-BL85	512Kx8/256Kx16	Smart Voltage
LH28F400BVE-BL12	512Kx8/256Kx16	Smart Voltage
LH28F400BVHE-BL12	512Kx8/256Kx16	Smart Voltage
LH28F400BVN-BL85	512Kx8/256Kx16	Smart Voltage
LH28F640BFE-PTTL80	4Mx16	3V
LH28F640BFHE-PTTL80	4Mx16	3V
LH28F640BFE-PBTL80	4Mx16	3V
LH28F640BFHE-PBTL80	4Mx16	3V
LH28F640BFE-PTTL90	4Mx16	3V
LH28F640BFHE-PTTL90	4Mx16	3V
LH28F640BFE-PBTL90	4Mx16	3V
LH28F640BFHE-PBTL90	4Mx16	3V
LH28F640BFHE-PTTL90	4Mx16	3V
LH28F640BFN-PTTLZ2	4Mx16	3V
LH28F500BJE-PTTL90	1Mx8/512Kx16	3V
LH28F500BJE-PTTL10	1Mx8/512Kx16	3V

付録K 対応フラッシュ メモリー一覧

続き

型式	構成	電源電圧
LH28F800BJHE-PTTL90	1Mx8/512Kx16	3V
LH28F800BJE-PBTL90	1Mx8/512Kx16	3V
LH28F800BJHE-PBTL90	1Mx8/512Kx16	3V
LH28F800BJHE-PTTL10	1Mx8/512Kx16	3V
LH28F800BJE-PBTL10	1Mx8/512Kx16	3V
LH28F800BJHE-PBTL10	1Mx8/512Kx16	3V
LH28F800BJB-PTTL90	1Mx8/512Kx16	3V
LH28F800BJB-PTTL10	1Mx8/512Kx16	3V
LH28F800BJHB-PTTL90	1Mx8/512Kx16	3V
LH28F800BJHB-PTTL10	1Mx8/512Kx16	3V
LH28F800BJB-PBTL90	1Mx8/512Kx16	3V
LH28F800BJB-PBTL10	1Mx8/512Kx16	3V
LH28F800BJHB-PBTL90	1Mx8/512Kx16	3V
LH28F800BJHB-PBTL10	1Mx8/512Kx16	3V
LH28F800BJHG-PBTL1FZ5	512Kx16	3V
LH28F800BVE-TTL90	1Mx8/512Kx16	3V
LH28F800BVE-TTL10	1Mx8/512Kx16	3V
LH28F800BVHE-TTL10	1Mx8/512Kx16	3V
LH28F800BVN-TTL90	1Mx8/512Kx16	3V
LH28F800BVN-TTL10	1Mx8/512Kx16	3V
LH28F800BVB-TTL90	1Mx8/512Kx16	3V
LH28F800BVB-TTL10	1Mx8/512Kx16	3V
LH28F800BVHB-TTL10	1Mx8/512Kx16	3V
LH28F800BVE-BTL90	1Mx8/512Kx16	3V
LH28F800BVE-BTL10	1Mx8/512Kx16	3V
LH28F800BVHE-BTL10	1Mx8/512Kx16	3V
LH28F800BVN-TTL90	1Mx8/512Kx16	3V
LH28F800BVN-BTL10	1Mx8/512Kx16	3V
LH28F800BVB-BTL90	1Mx8/512Kx16	3V
LH28F800BVB-BTL10	1Mx8/512Kx16	3V
LH28F800BVHB-BV85	1Mx8/512Kx16	5V
LH28F800BVN-BV10	1Mx8/512Kx16	5V
LH28F800BVHB-BV85	1Mx8/512Kx16	5V

付録 L V850/ME2 内蔵命令 RAM の取り扱い

V850E/ME2 では内蔵命令 RAM に対する書込許可/読出許可は周辺 I/O レジスタ内蔵命令 RAM モード・レジスタ (IRAMM) により制御されます。

RESET 後には内蔵命令 RAM はライト・モードになっていますので内蔵命令 RAM 領域からの読出は不正な値で行われます。MJXDEBW デバッガからいったん DOWNLOAD や EXAMINE コマンドで内蔵命令 RAM 領域への書込みが行われると MJXDEBW デバッガは内蔵命令 RAM 領域をリード・モードへ変更します。この後、内蔵命令 RAM 領域へのアクセスは MJXDEBW デバッガにより自動的に IRAMM レジスタの操作が行われます。通常時はリード・モードになっています。ブートストラップルーチン等をデバッグする場合はご注意ください。

検索項目

[MJX440 について](#)

[MJX440 のおもな特長](#)

[最小構成](#)

[ROM インサーキット接続](#)

[外部トリガ ケーブル接続](#)

[MJX440 を使用する前の準備作業](#)

[MULTI について](#)

[MJXDEBW について](#)

[ホスト インターフェースの設定](#)

[ハードウェアの接続](#)

[MJX440 とホストの接続](#)

[ROM プローブの接続](#)

[外部トリガ ケーブルの接続](#)

[電源の接続と電源投入手順](#)

[MJX440 を操作するソフトウェアのインストール](#)

[MJX440 を操作するソフトウェアのインストール方法](#)

[インストールされるファイル](#)

[ソフトウェアのファイル構成](#)

[MJX440 の環境設定](#)

[コンフィグ支援ツール MJXCFG](#)

[MJX440 を操作するソフトウェアの起動と終了](#)

[MULTI を使用する場合](#)

[MJXDEBW を使用する場合](#)

[次ページ](#)

[MJXDEBW コマンド](#)

[MJXDEBW コマンドの一覧](#)

[コマンド名の省略](#)

[MULTIを使用する場合の注意](#)

[MJXDEBW メニューコマンド](#)

[MJXDEBW メニューコマンド一覧](#)

[高速ダウンロード](#)

[仕様](#)

[本体寸法](#)

[本体重量](#)

[電源 \(AC アダプタ\)](#)

[ROM プローブ ケーブル](#)

[外部トリガ ケーブル](#)

[使用温度範囲](#)

[保存温度範囲](#)

[周囲湿度範囲](#)

[対応 ROM](#)

[対応 ROM 個数](#)

[エミュレーション メモリ容量](#)

[アクセス タイム](#)

[インターフェース](#)

[ターゲット インターフェース](#)

[対応 CPU](#)

[ダウンロード速度](#)

[対応デバッガ](#)

[ブレークポイント機能](#)

[トレース機能](#)

[制限事項](#)

[前ページ](#)

[次ページ](#)

[ターゲット システムの制限事項](#)

[N-Wire コネクタ](#)

[ROM プローブ基板](#)

[ROM プローブ基板 J-101A ジャンパ設定](#)

[ROM プローブ基板 J-102A ジャンパ設定](#)

[ROM プローブ基板 J-103A ジャンパ設定](#)

[ROM プローブ基板 J-104A ジャンパ設定](#)

[対応 ROM ピンアサイン](#)

[J-101A 27010](#)

[J-101A 27020](#)

[J-101A 27040](#)

[J-101A 27080](#)

[J-101A 271000](#)

[J-102A/J-104A/27C4000](#)

[J-102A/J-104A/27C8000](#)

[J-102A/J-104A/27C16000](#)

[J-103A 271024](#)

[J-103A 272048](#)

[J-103A 274096](#)

[前ページ](#)

[次ページ](#)

[LED](#)

[レジスタ名一覧](#)

[MJX バイナリ ファイル](#)

[バイナリ ファイル構成](#)

[ヘッダ](#)

[データ](#)

[終了ヘッダ](#)

[ターゲット システムへのプロービング](#)

[前ページ](#)

MJX440 について

< 関連項目 >

- [MJX440 のおもな特長](#)
- [最小構成](#)
- [ROM インサーキット接続](#)
- [外部トリガ ケーブル接続](#)
- [MJX440 を使用する前の準備作業](#)

[検索項目の先頭ページ](#)

ハードウェアの接続

< 関連項目 >

- [MJX440 とホストの接続](#)
- [ROMプローブの接続](#)
- [外部トリガ ケーブルの接続](#)
- [電源の接続と電源投入手順](#)

[検索項目の先頭ページ](#)

MJX440 とホストの接続

< 関連項目 >

・[パラレル接続](#)

[検索項目の先頭ページ](#)

N-Wire プローブの接続

<関連項目>

• [N-Wire プローブ](#)

[検索項目の先頭ページ](#)

ROM プローブの接続

<関連項目>

- [ROMプローブの接続方法](#)

[検索項目の先頭ページ](#)

ROM プローブの接続方法

<関連項目>

- [バス幅 8 ビット ROM 1 個、ROM アクセス バス幅 8 ビットの場合](#)
- [バス幅 8 ビット ROM 2 個、ROM アクセス バス幅 8 ビットの場合](#)
- [バス幅 8 ビット ROM 4 個、ROM アクセス バス幅 8 ビットの場合](#)
- [バス幅 8 ビット ROM 2 個、ROM アクセス バス幅 16 ビットの場合](#)
- [バス幅 8 ビット ROM 4 個、ROM アクセス バス幅 16 ビットの場合](#)
- [バス幅 8 ビット ROM 4 個、ROM アクセス バス幅 32 ビットの場合](#)
- [バス幅 16 ビット ROM 1 個、ROM アクセス バス幅 16 ビットの場合](#)
- [バス幅 16 ビット ROM 2 個、ROM アクセス バス幅 16 ビットの場合](#)
- [バス幅 16 ビット ROM 2 個、ROM アクセス バス幅 32 ビットの場合](#)
- [バス幅 16 ビット ROM 4 個、ROM アクセス バス幅 32 ビットの場合](#)

[検索項目の先頭ページ](#)

MJX440 を操作するソフトウェアのインストール

< 関連項目 >

- [MJX440 を操作するソフトウェアのインストール方法](#)
- [インストールされるファイル](#)
- [ソフトウェアのファイル構成](#)

[検索項目の先頭ページ](#)

MJX440 を操作するソフトウェアの起動と終了

< 関連項目 >

- [MULTI を使用する場合](#)
- [MJXDEBW を使用する場合](#)

[検索項目の先頭ページ](#)

MJXDEBW コマンド

<関連項目>

- [MJXDEBW コマンドの一覧](#)
- [コマンド名の省略](#)
- [MULTI を使用する場合の注意](#)

[検索項目の先頭ページ](#)

MJXDEBW メニューコマンド

<関連項目>

• [MJXDEBW メニューコマンドの一覧](#)

[検索項目の先頭ページ](#)

検索項目

N_Wire コネクタ

<関連項目>

• [N_Wire コネクタ](#)

[検索項目の先頭ページ](#)

ROMプローブ基板

<関連項目>

- [ROMプローブ基板 J-101A ジャンパ設定](#)
- [ROMプローブ基板 J-102A ジャンパ設定](#)
- [ROMプローブ基板 J-103A ジャンパ設定](#)
- [ROMプローブ基板 J-104A ジャンパ設定](#)

[検索項目の先頭ページ](#)

対応 ROM ピンアサイン

< 関連項目 >

- [J-101A 27010](#)
- [J-101A 27020](#)
- [J-101A 27040](#)
- [J-101A 27080](#)
- [J-101A 271000](#)
- [J-102A/J-104A/27C4000](#)
- [J-102A/J-104A/27C8000](#)
- [J-102A/J-104A/27C16000](#)
- [J-103A 271024](#)
- [J-103A 272048](#)
- [J-103A 274096](#)

[検索項目の先頭ページ](#)

MJX バイナリ ファイル

<関連項目>

• [バイナリ ファイル構成](#)

• [ヘッダ](#)

• [データ](#)

• [終了ヘッダ](#)

[検索項目の先頭ページ](#)
