

# MJX440 for V831/V832 User's Manual

Rev.2.12 2004/03/02

# 変更履歴

バージョン	変更内容	変更日
2.01	● 用語を修正 (パラレル I/F ケーブル →パラレル インターフェース ケーブル)	02/01/08
2.02	● pdfファイル名を変更 (Mjxv831J.pdf → MjxV83XJ.pdf)	03/07/15
2.10	● コンフィグレーションファイルの設定画面の変更	04/02/23
2.11	● 誤字修正	04/02/27
2.12	● 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。
- MS-DOS、Windows 95、Windows 98、Windows NT は、Microsoft の登録商標です。
- MULTI は、Green Hills Software 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 3 月

この度は、MJX440 for V831/V832(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。  
させていただきます。

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 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
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 .....	40
第七章 MJX440 コマンド .....	44
ABORT .....	47
BATCH .....	48
BP .....	49
CLEAR .....	51
CONFIG .....	52
DUMP .....	53
EXAMINE .....	54
FILL .....	55
GO .....	56
HISTORY .....	57
INIT .....	59
JOURNAL .....	60
LOAD .....	61
MOVE .....	62
PIN .....	63
PORT .....	64
QUIT .....	65
REGISTER .....	66
STEP .....	67
TRACE .....	68
UNASM .....	71

VERSION.....	72
WAIT .....	73
XPIN .....	74
第八章 MJXDEBW メニューコマンド.....	75
ファイル(F) ～ バッチ(B).....	76
ファイル(F) ～ ジャーナル開始(S).....	77
ファイル(F) ～ ジャーナル停止(E) .....	77
ファイル(F) ～ トレース結果(T).....	78
エミュレーション(E) ～ 実行(G).....	79
エミュレーション(E) ～ 再スタート(R).....	79
エミュレーション(E) ～ 中断(B).....	80
エミュレーション(E) ～ ステップ(S).....	80
エミュレーション(E) ～ N-ステップ(N).....	80
エミュレーション(E) ～ ダウンロード(L).....	82
エミュレーション(E) ～ 初期化(I).....	83
表示(V) ～ メモリ(M).....	84
表示(V) ～ レジスタ(R).....	85
表示(V) ～ 逆アセンブル(D).....	86
表示(V) ～ トレース結果(T).....	87
表示(V) ～ コマンド応答クリア(C).....	92
表示(V) ～ オプション(O) ～ フォント(F) .....	93
表示(V) ～ オプション(O) ～ 表示(V).....	94
変更(M) ～ メモリ(M).....	95
変更(M) ～ レジスタ(R).....	96
変更(M) ～ アセンブラ(A).....	97
変更(M) ～ フィル(F).....	98
変更(M) ～ 移動(M).....	99
設定(S) ～ ブレークポイント(B).....	100
設定(S) ～ トレース(T).....	102
設定(S) ～ 設定の読出(L).....	104
設定(S) ～ 設定の保存(S).....	105
設定(S) ～ コンフィグレーション(C) ～ 設定(S)...	106
設定(S) ～ コンフィグレーション(C) ～ 表示(V)...	108
設定(S) ～ コンフィグレーション ～ 上書き保存(E) .....	109
設定(S) ～ コンフィグレーション ～ 名前を付けて保存(A).....	110
ヘルプ(H) ～ トピックの検索(H).....	111

ヘルプ(H) ~ MJXDEBW のバージョン情報(A).....	112
第九章 高速ダウンロード.....	113
付録 A 仕様 .....	114
付録 B ターゲット システムの制限事項 .....	115
付録 C N-Wire コネクタ.....	116
付録 D ROM プローブ .....	118
付録 E 対応 ROM ピンアサイン .....	121
付録 F LED.....	125
付録 G レジスタ名一覧.....	126
付録 H MJX バイナリ ファイル.....	127
付録 I ターゲット システムへのプロービング .....	128

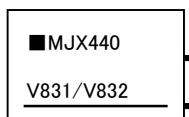
# 第一章 概要

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 V831/V832 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 V831/V832 本体



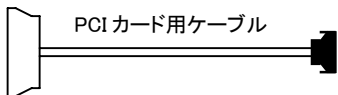
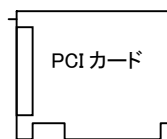
- ☐ AC アダプタ



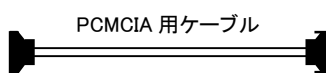
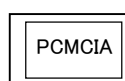
- ☐ AC コード



- ☐ ホスト インターフェース (PCI カード、または PCMCIA カード)と  
ホスト インターフェース ケーブル\*1



または



- ☐ N-Wire ケーブル



---

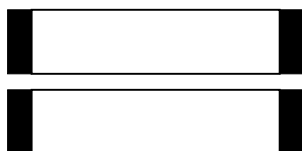
\*1 出荷時には、PCI カード用、または PCMCIA カード用のどちらか一方が含まれています。



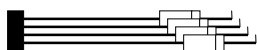
- ☐ ROM プローブ\*1



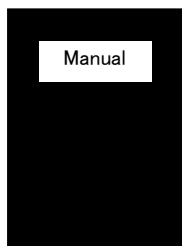
- ☐ ROM プローブ ケーブル 2本



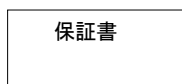
- ☐ 外部トリガ ケーブル



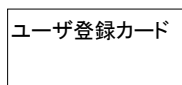
- ☐ ユーザーズ マニュアル



- ☐ 保証書

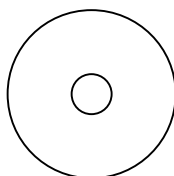


- ☐ ユーザー登録カード



**【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、次ページ「表 1-1 出荷時の ROM プローブ」を参照してください。

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

表 1-1 出荷時の ROM プローブ

## 1.2 MJX440 の概略

MJX440 は、N-Wire コネクタを備えた V831 または V832 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。
- [MJX440 コマンド](#)が使用できます。MJX440 が持っているハードウェア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 440K バイト/秒、ROM インサーキット接続時 4M バイト/秒)

### 最小構成

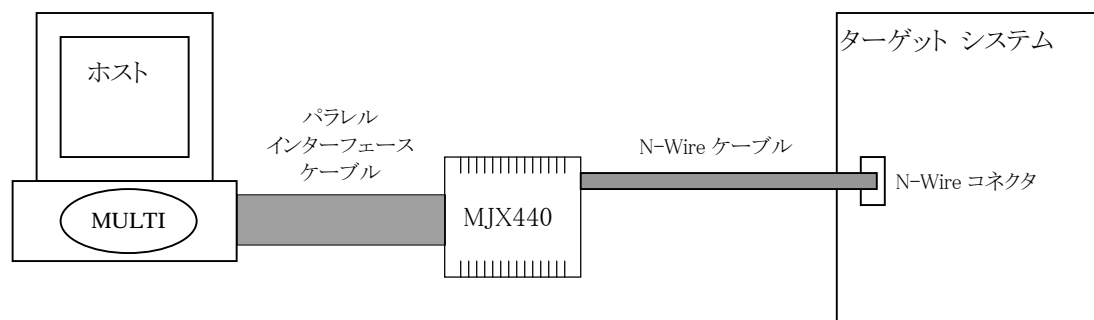


図 1-1 最小構成

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

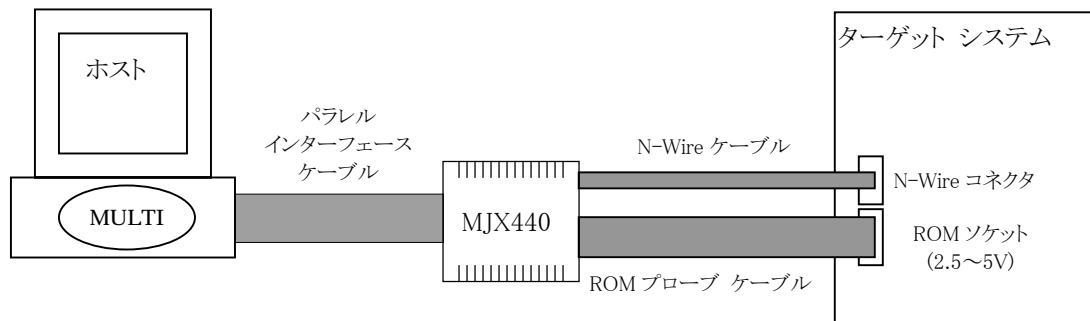


図 1-2 ROM インサーキット接続

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

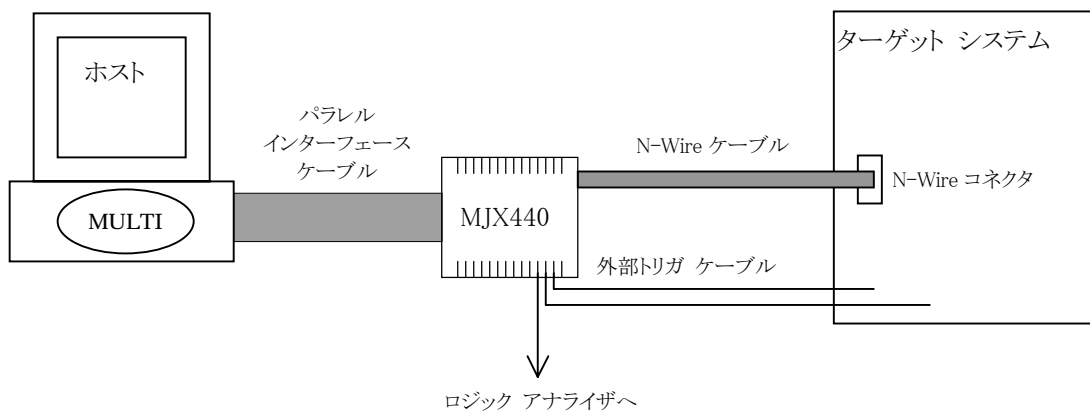


図 1-3 外部トリガ ケーブル接続

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

- 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)
- 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 メモリに残す。(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)
- MJX440 コマンドで操作できる、汎用信号をターゲット システムで使用する(出力)

### **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なってください。これらは、購入時に一度だけ行なえばよい作業です。

- [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](#)を参照してください。

高速ダウンロードを行ないたい場合は、[第九章](#)を参照してください。

### MULTI について

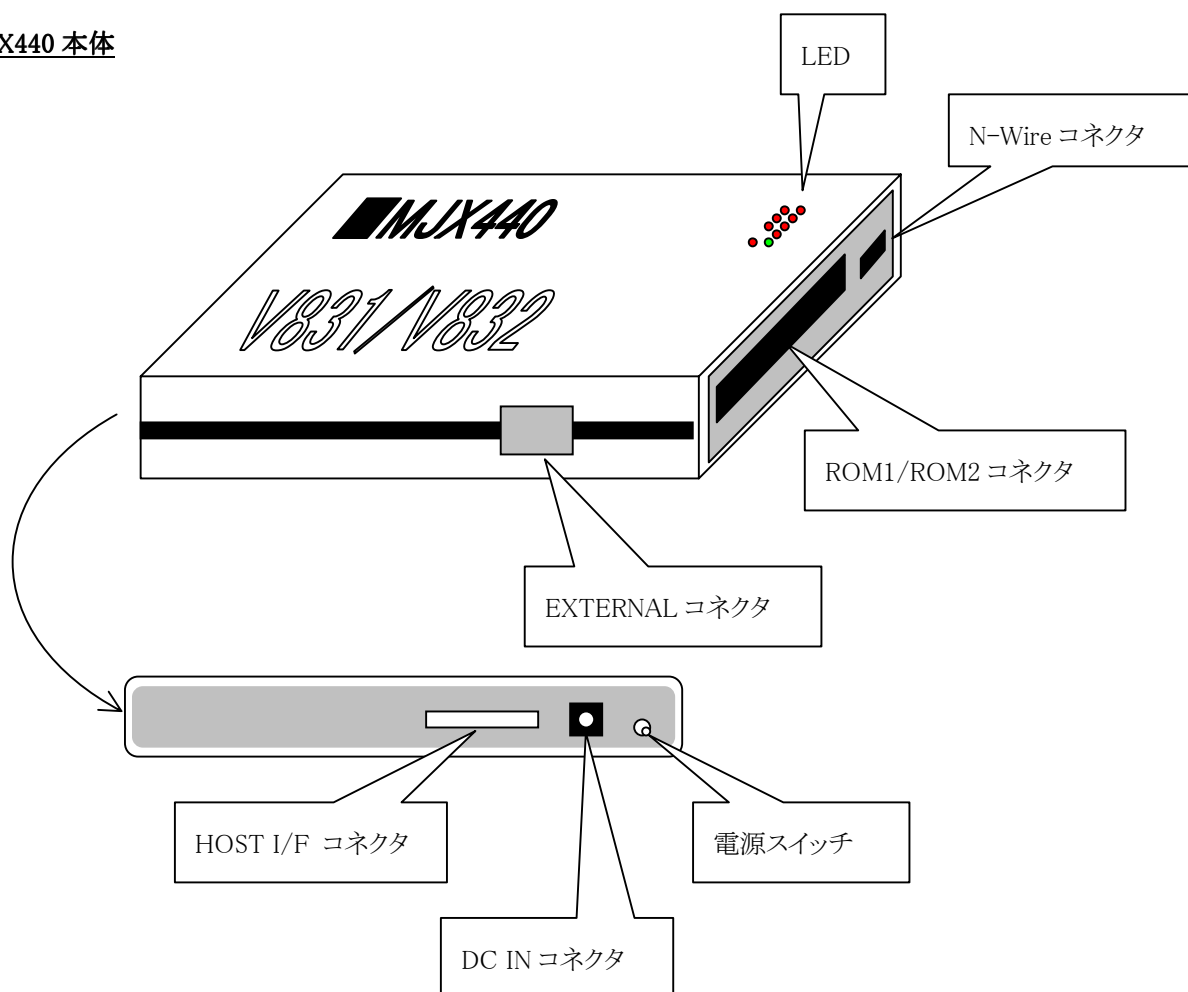
MULTI はいろいろな環境で実行することができる高級言語デバッガです。サーバプログラム MJXSERV を呼び出すことによって、MJX440 の環境で実行させることができますようになります。

### 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。

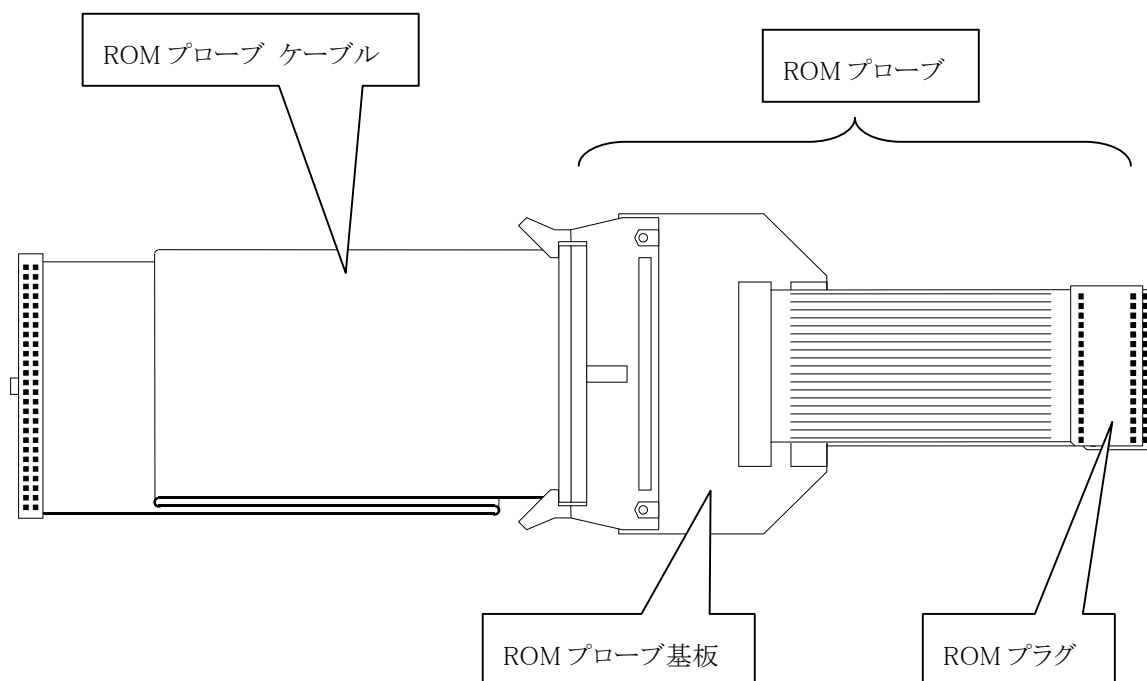
EXTERNAL コネクタ

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 front cover of a manual. It has a solid black background. In the center, there is a white rectangular box containing the title text in black. The text is arranged in four lines: 'MJX440', 'Host Interface', 'Installation', and 'Manual'.

### 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 と他の機器を接続は、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、ホスト インターフェイス ケーブルで接続してください。

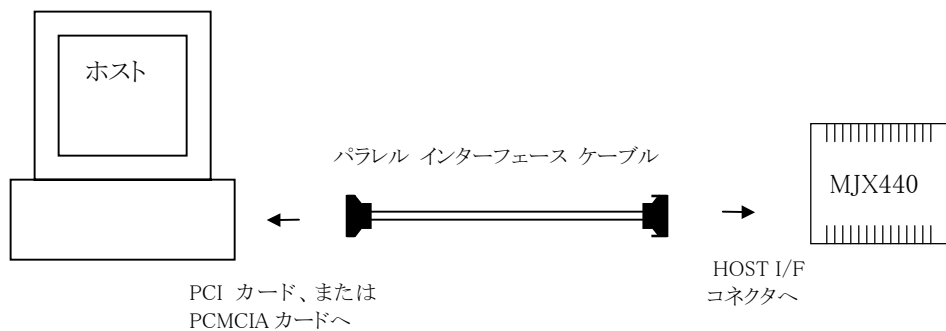


図 3-1 MJX440 とホストの接続

**【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

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。

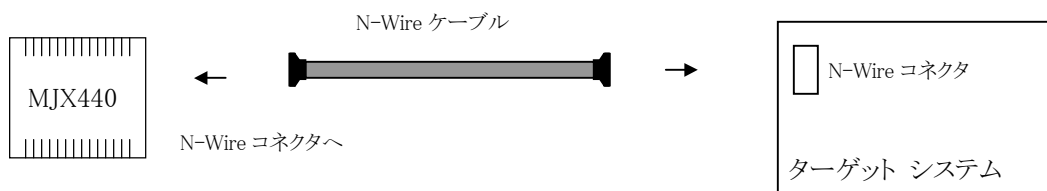


図 3-2 N-Wire ケーブルの接続

**【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、添付されているすべての ROM プローブを、ROM プローブ ケーブルに接続してください。

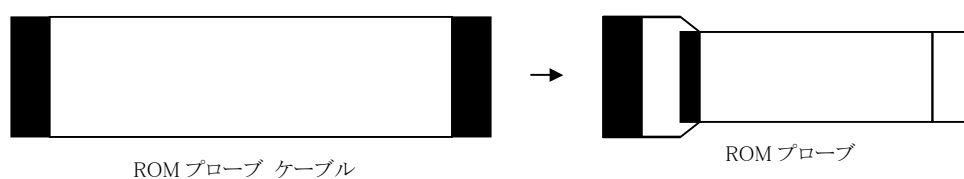


図 3-3 ROM プローブと ROM プローブ ケーブルの接続 1

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。

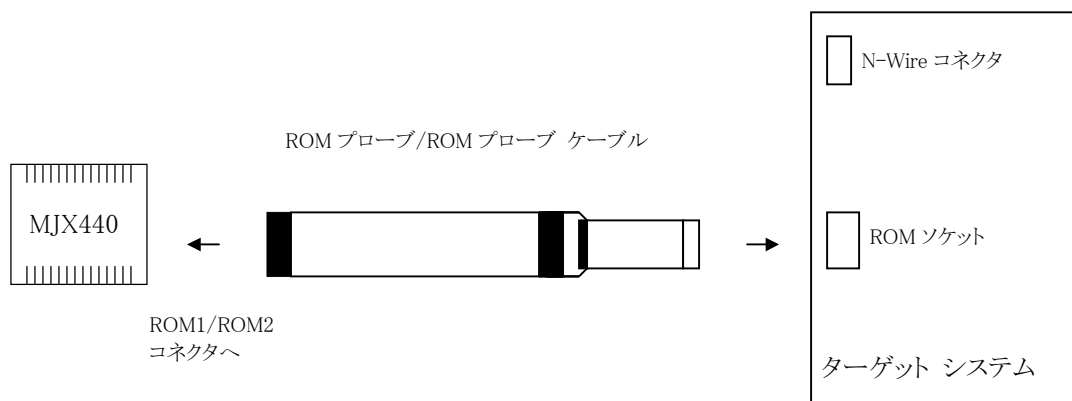


図 3-4 ROM プローブと ROM プローブ ケーブルの接続 2

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。

- ROM のデータ バス幅
- ROM の個数
- ROM 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)

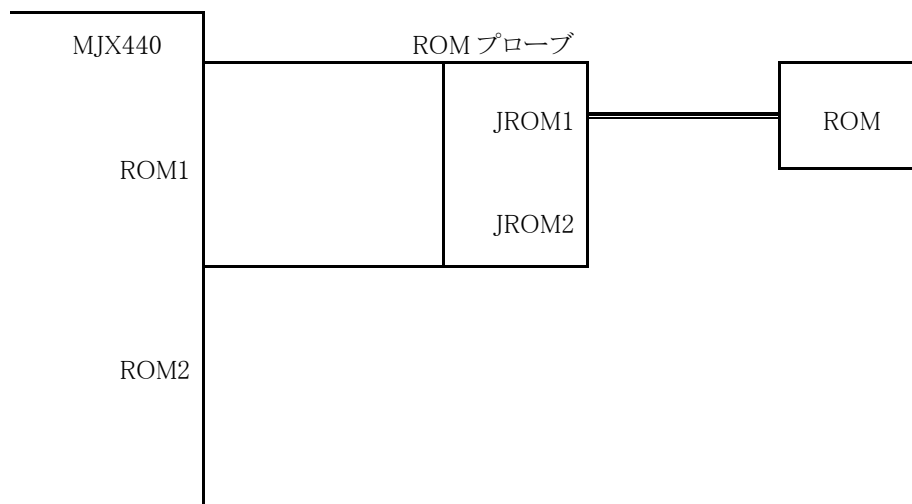


図 3-5-1 ROM プローブの接続 1

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、ROM2)

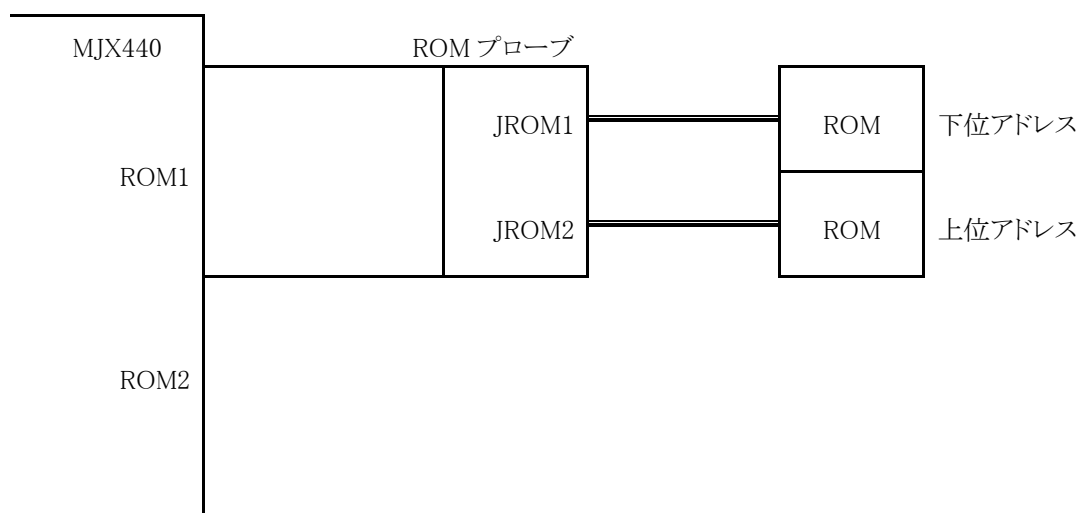


図 3-5-2 ROM プローブの接続 2

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、ROM3、ROM4)

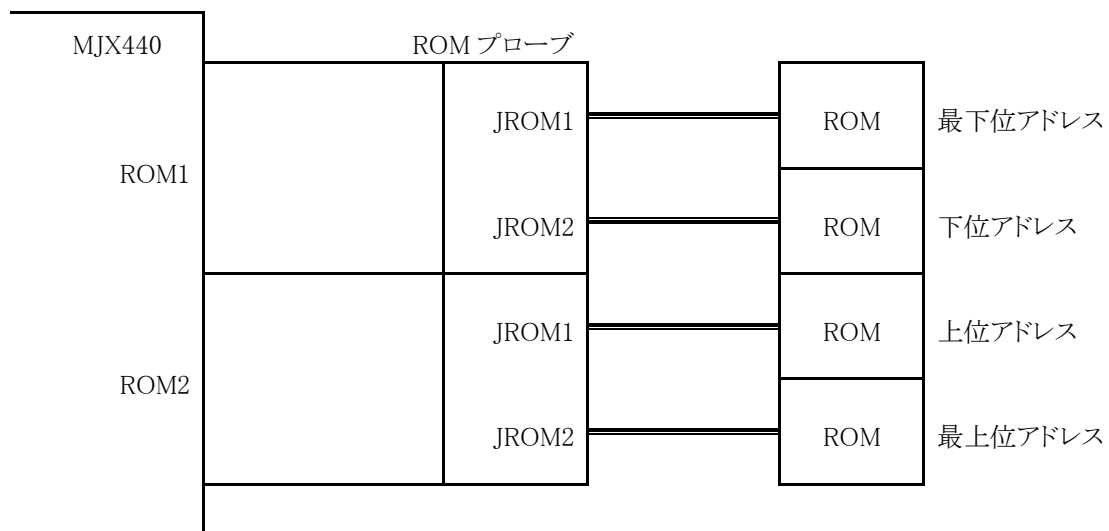


図 3-5-3 ROM プローブの接続 3

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、ROM2)

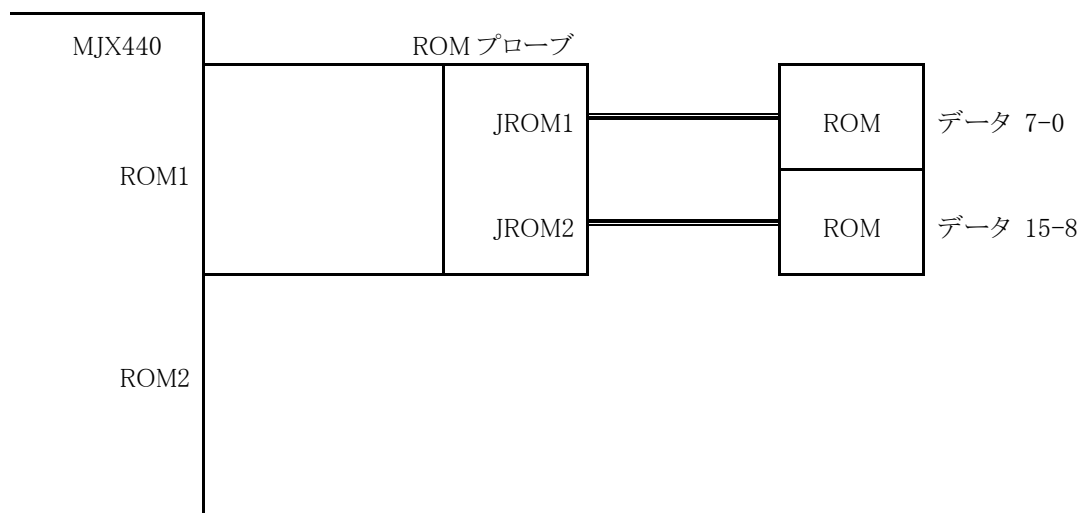


図 3-5-4 ROM プローブの接続 4

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、ROM3、ROM4)

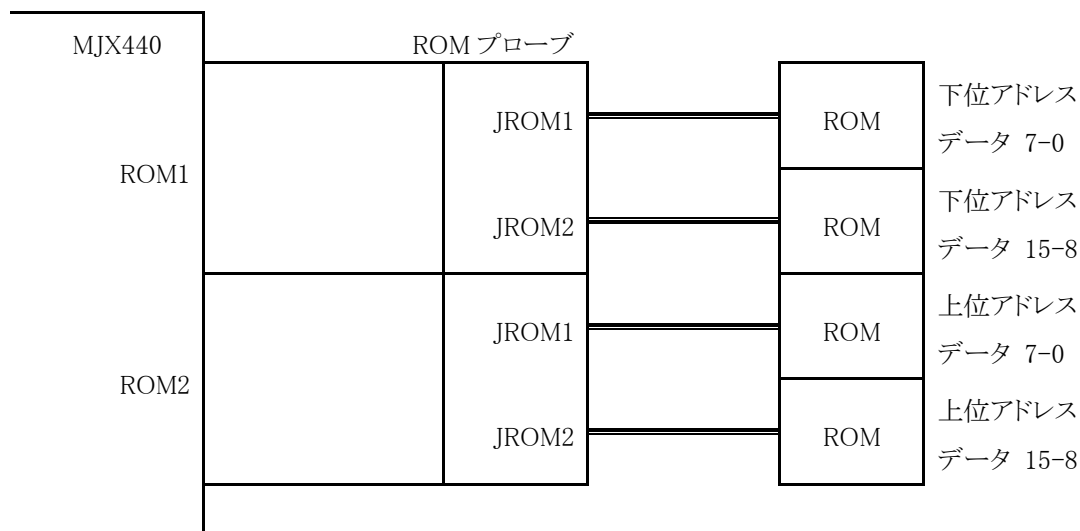


図 3-5-5 ROM プローブの接続 5

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、ROM3、ROM4)

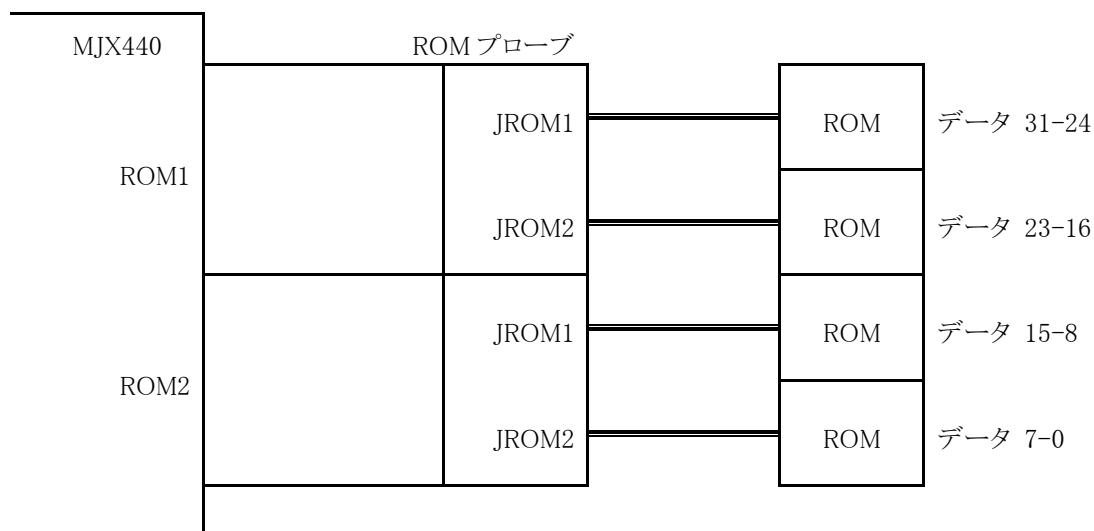


図 3-5-6 ROM プローブの接続 6



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)

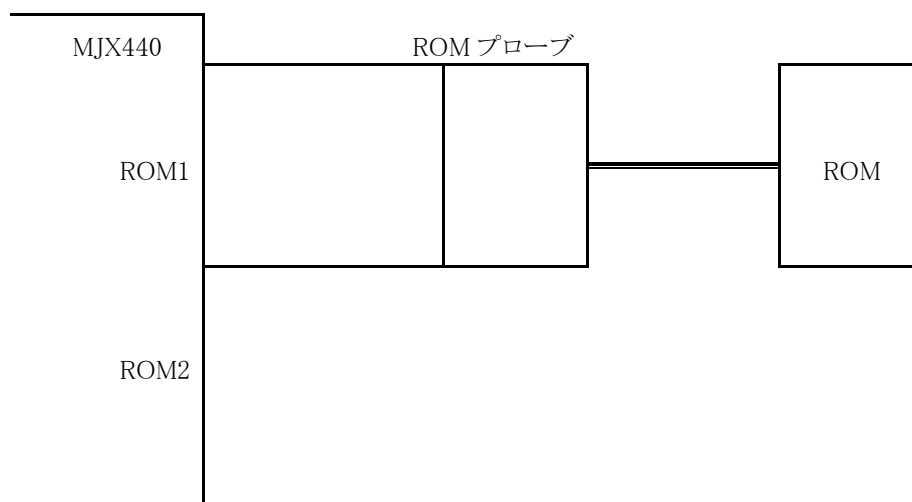


図 3-5-7 ROM プロブの接続 7

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)

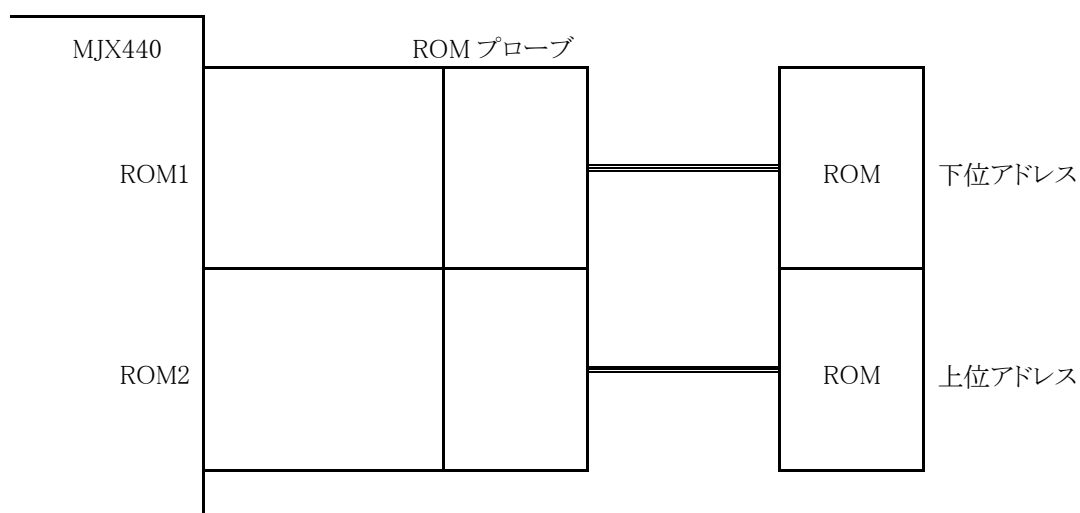


図 3-5-8 ROM プロブの接続 8

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)

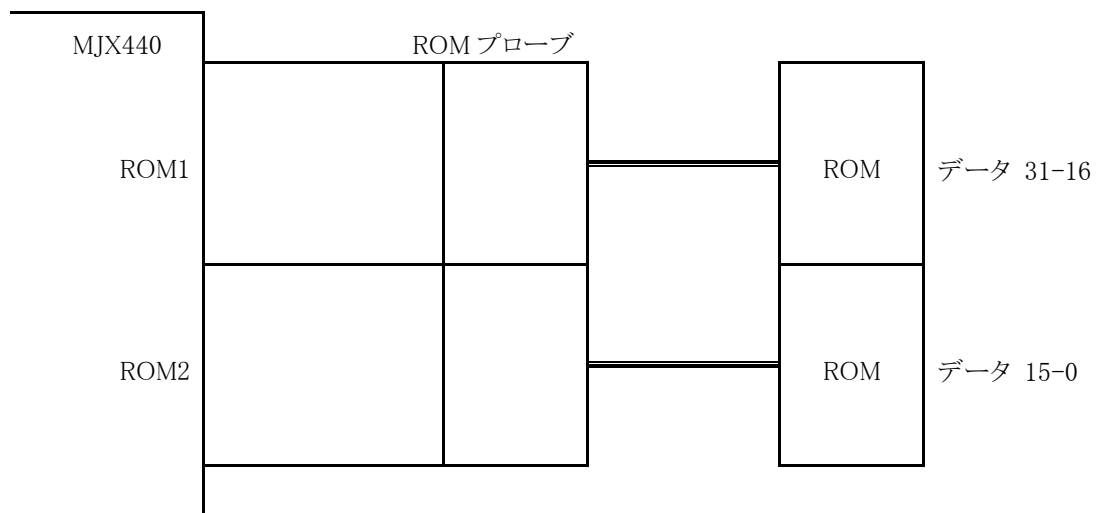


図 3-5-9 ROM プローブの接続 9

### 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、MJX440 の EXTERNAL コネクタと、ターゲット システムの信号やロジック アナライザを、外部トリガ ケーブルで接続してください。

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。

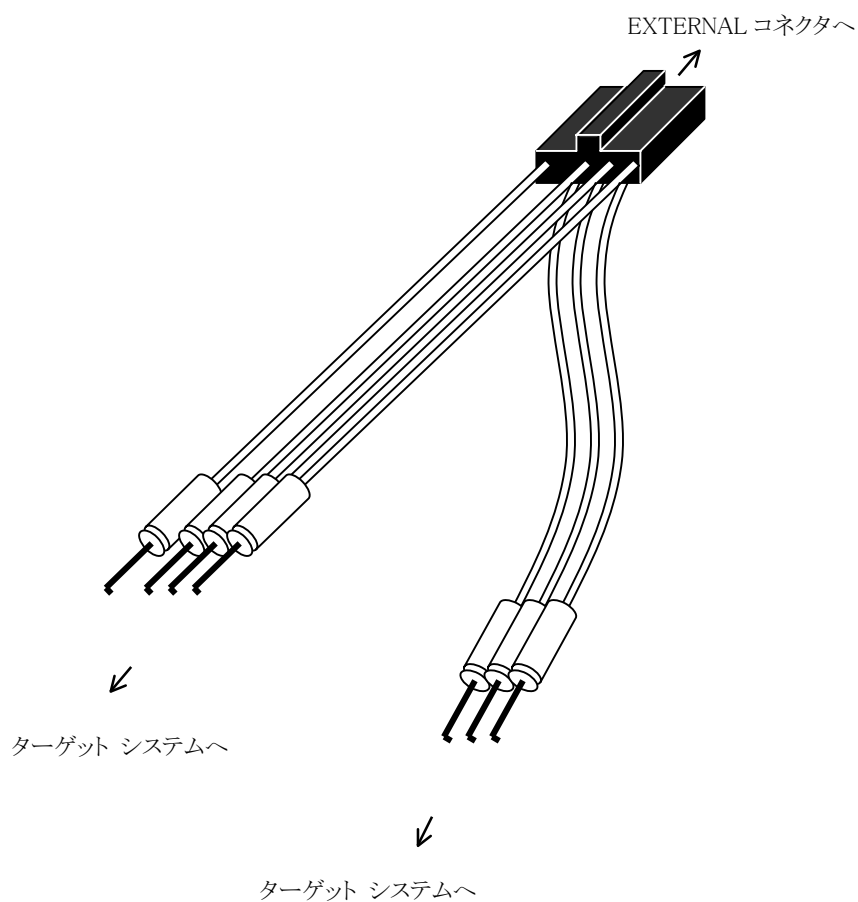


図 3-5 外部トリガ ケーブルの接続

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。

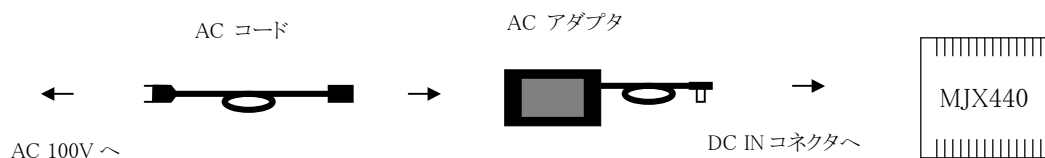
信号名	色	入出力	機能
EXTIN1	クリップ 黄 ケーブル 茶	TTL 入力	接続した信号が HIGH レベルの場合、対応する LED が点灯します。 <sup>*1</sup> また、EXTIN1 と EXTIN2 については、リアルタイム トレースにも、信号の状態が記録されます。
EXTIN2	クリップ 黄 ケーブル 赤	TTL 入力	
EXTIN3	クリップ 黄 ケーブル 橙	TTL 入力	
EXTOUT1	クリップ 赤 ケーブル 茶	3.3V O.D.出力 <sup>*2</sup>	MJX440 コマンドの XPIN コマンドで、LOW または HIGH レベルを出力します。
EXTOUT2	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。

<sup>\*1</sup> 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。

<sup>\*2</sup> 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
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

**【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

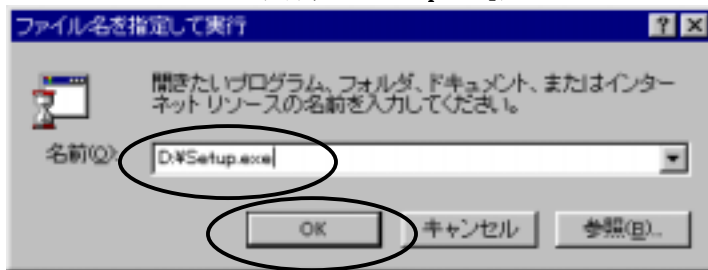
MJX440 を操作するソフトウェアのインストール方法について記述しています。

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。

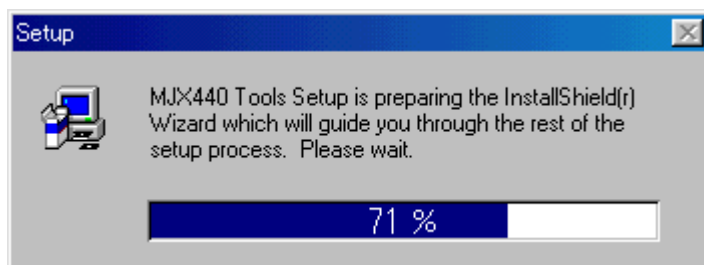
1. Green Hills Software 社の統合開発環境 MULTI をインストールします。
2. CD-ROMドライブに「MJX440 for V831/V832 Tools」の CD-ROM をセットします。

ここで、セットアップ プログラムが自動的に起動された場合は、3. と 4. の手順は必要ありません。

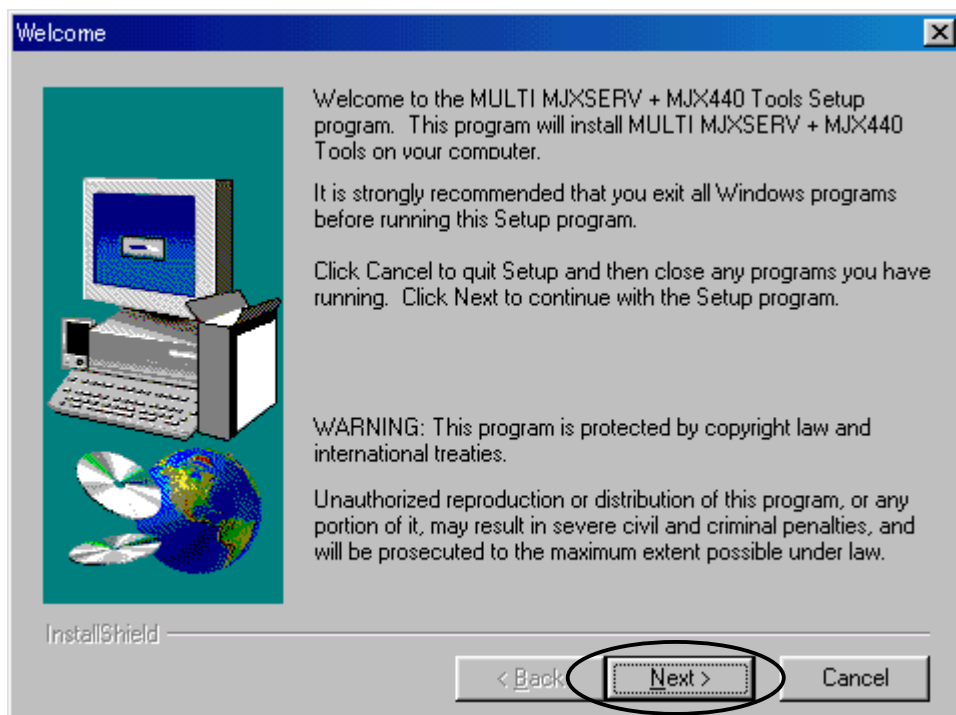
3. 「スタート」、「ファイル名を指定して実行(R)...」の順に選択し、「ファイル名を指定して実行」ダイアログを表示します。
4. 「名前(O)」にセットアップ プログラム Setup.exe を指定し、「OK」をクリックします。(CD-ROM ドライブが D ドライブの場合、「D:¥Setup.exe」)



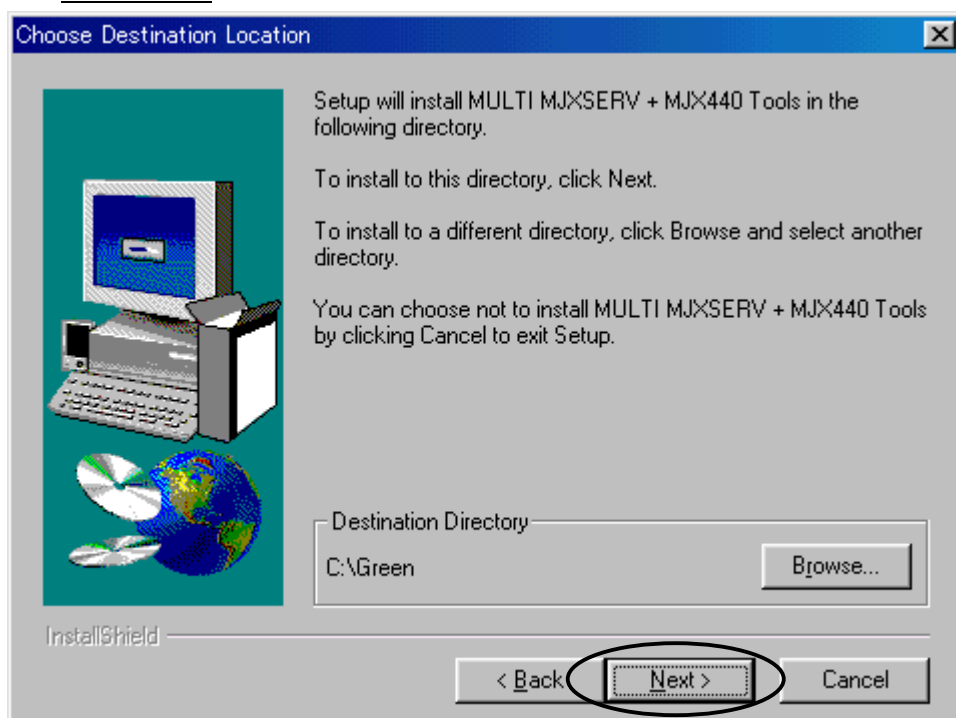
5. セットアップ プログラムが起動します。



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

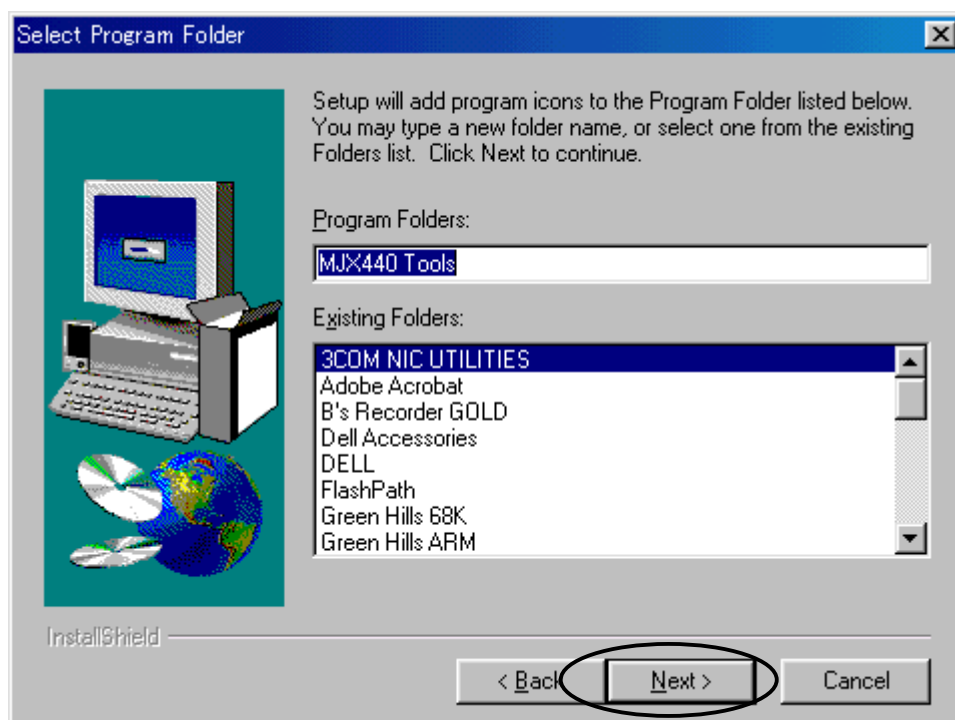


7. 「Choose Destination Location」ダイアログが表示されるので、「Browse...」ボタンをクリックしてインストール先を指定します。先にインストールした MULTI と同じディレクトリを、インストール先に指定してください。(デフォルトは「C:\¥Green」です)



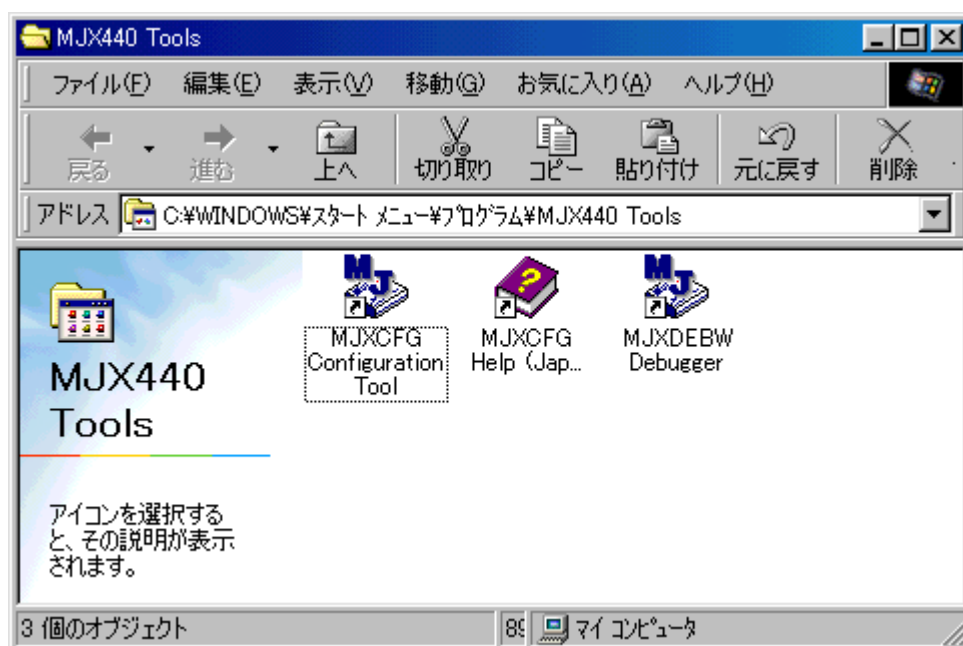
- 「Next >」をクリックします。

「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。



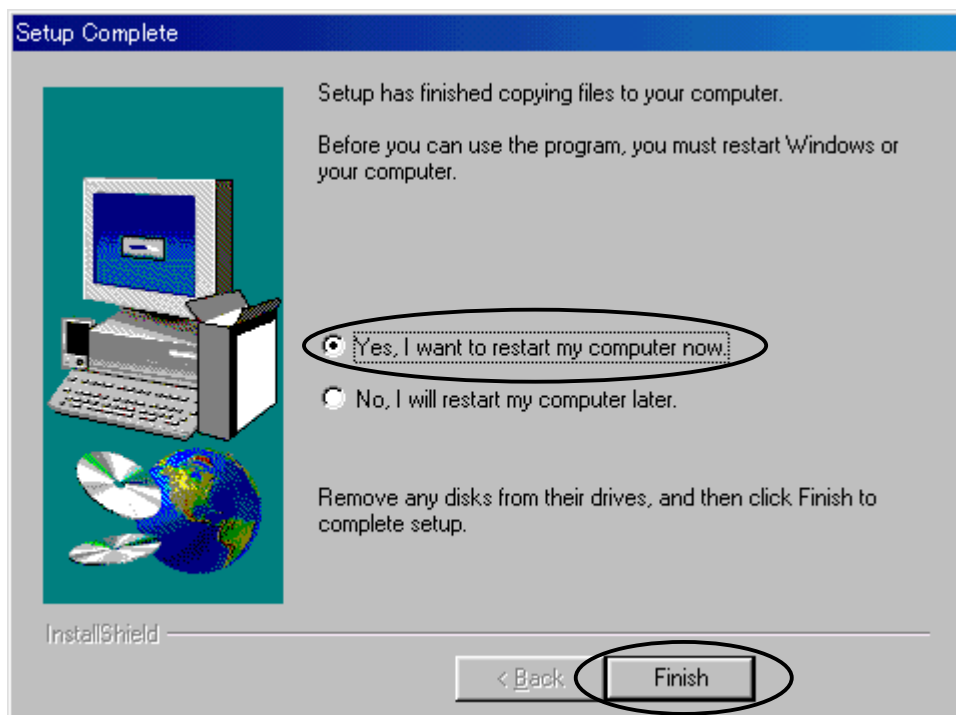
8. インストールが開始します。

9. 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。





10. 「**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。



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)
MJXDBAPI.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
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

ソフトウェアのファイル構成は次のとおりです。

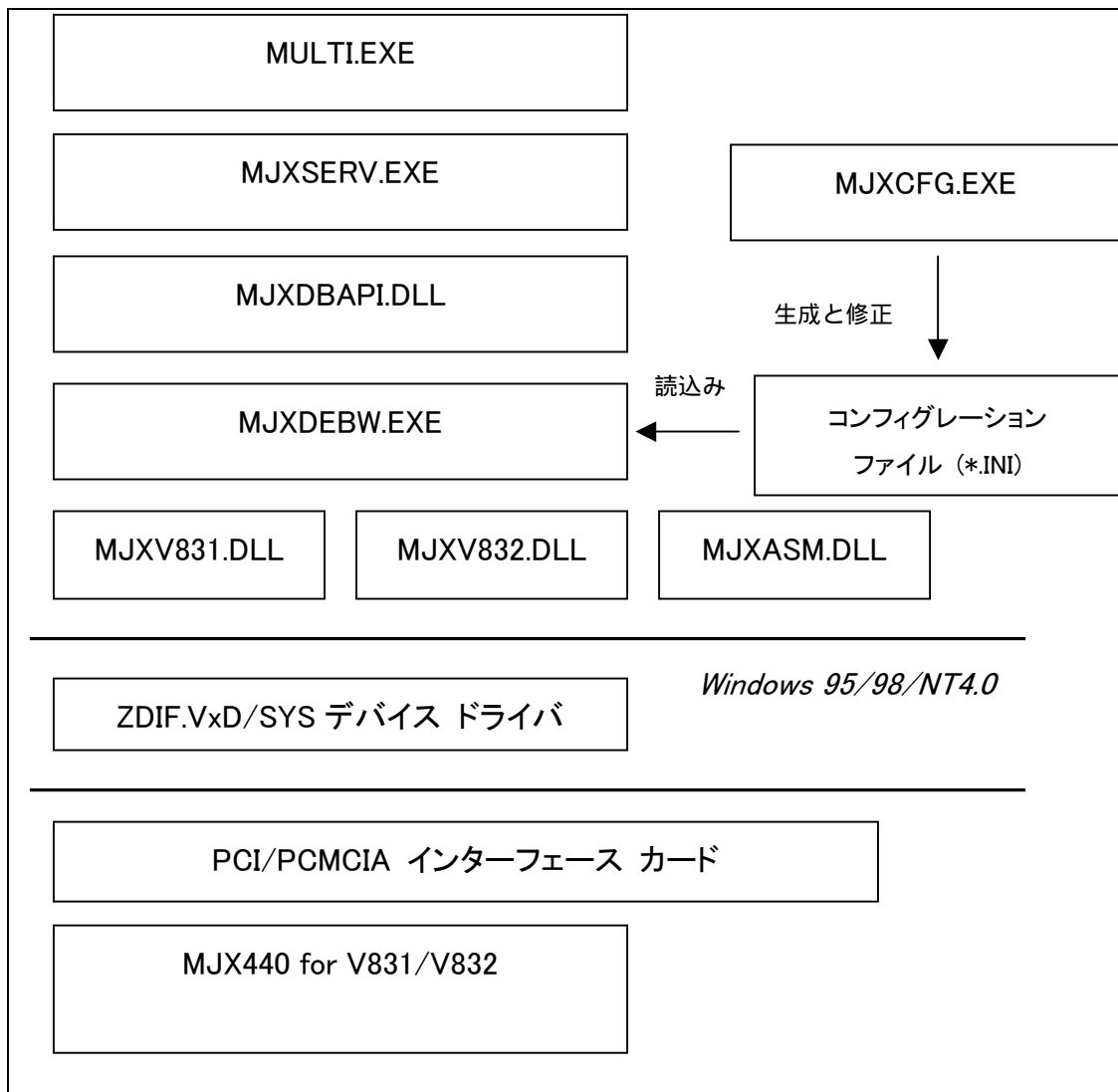


図 4-1 ソフトウェアのファイル構成

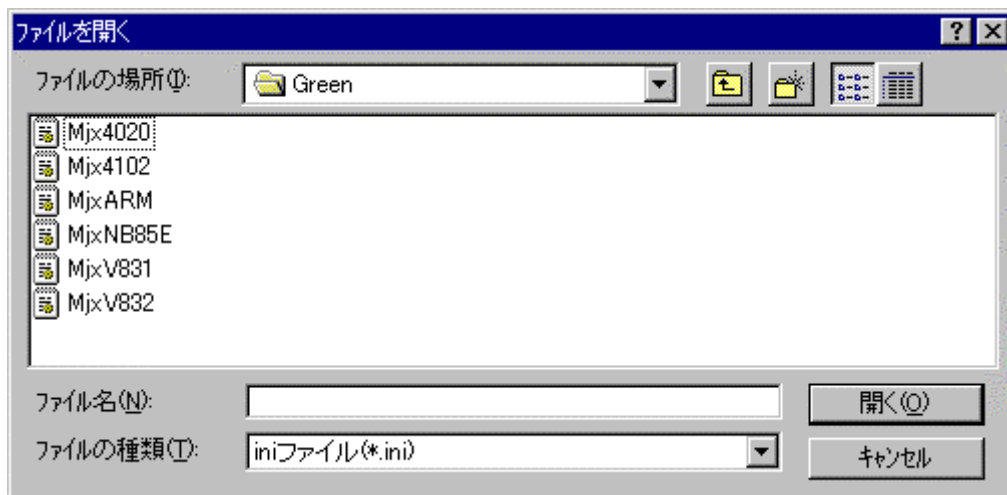
- MULTI.EXE は MJXSERV.EXE を子プロセスとして呼び出します。
- MJXSERV.EXE は MJXDBAPI.DLL を経由して MJXDEBW.EXE を子プロセスとして呼び出します。
- MJXDEBW.EXE は MDXSERV.EXE の子プロセスとしてだけでなく、単体でも動作できます。

## 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。

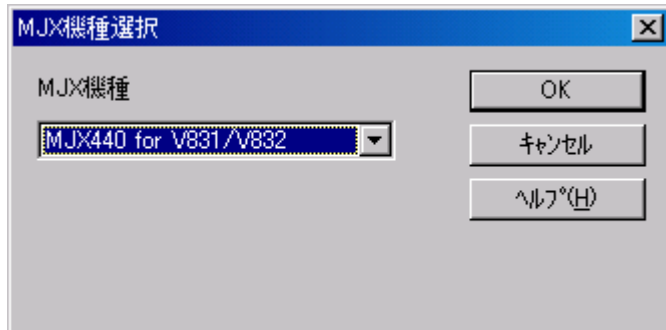
1. スタート メニュー
2. プログラム(P)
3. MJX Tools
4. MjxCfg Config Tool



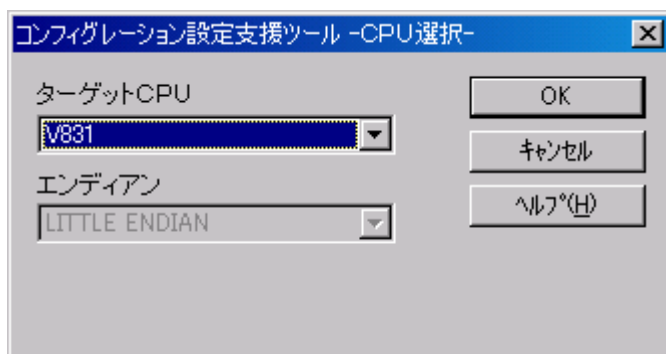
環境設定を格納しておくコンフィグレーション ファイル名を指定します。

ターゲット システムの CPU に応じて、**MJXV831.INI** または **MJXV832.INI** を指定してください。

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「MJX330for V831/V832」を選択し、**OK** ボタンを押してください。



ターゲットCPU\*1を選択し、**OK** ボタンを押してください。

---

\*1 ターゲットCPUは V831,V832 のいずれかを指定してください。

ターゲット CPU が V832 の場合、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 (MjxCfgJ.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。

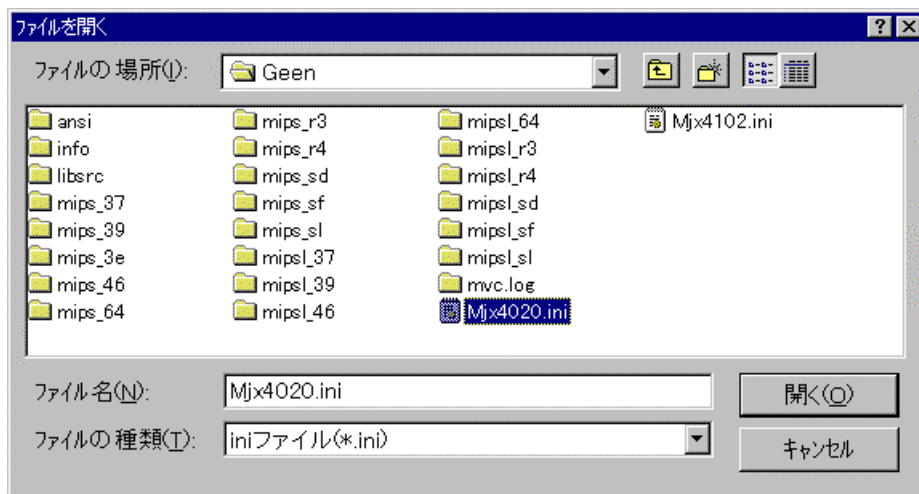
【注意】 MJX440 を操作するソフトウェアは、同時にひとつしか実行できません。

### 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。<sup>\*1</sup>

```
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、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。<sup>\*2</sup>



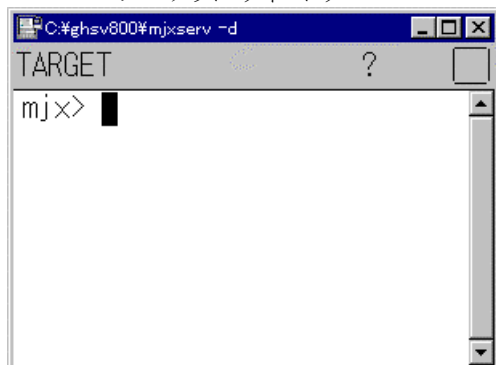
<sup>\*1</sup> ビルダから操作する場合は、サーバ名を `mjxserv` と指定し、**リモート** ボタンを押します。

<sup>\*2</sup> 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、MULTIのターゲット ウィンドウとMJXDEBW のウィンドウが開きます。

MULTI のターゲット ウィンドウ



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 **MJX832.INI** を指定します。

```
remote mjxserv mjx832.ini
```

終了するためには、*quit* コマンドを入力してください。

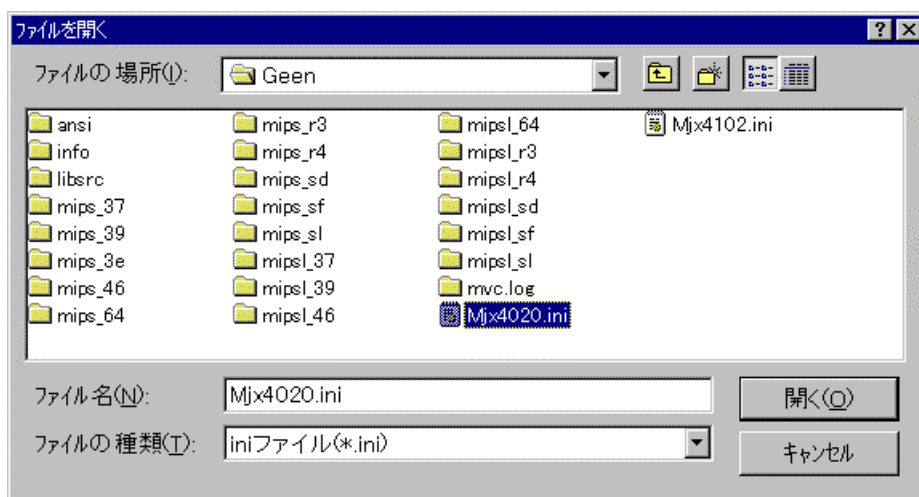
```
quit
```

## MJXDEBW の場合

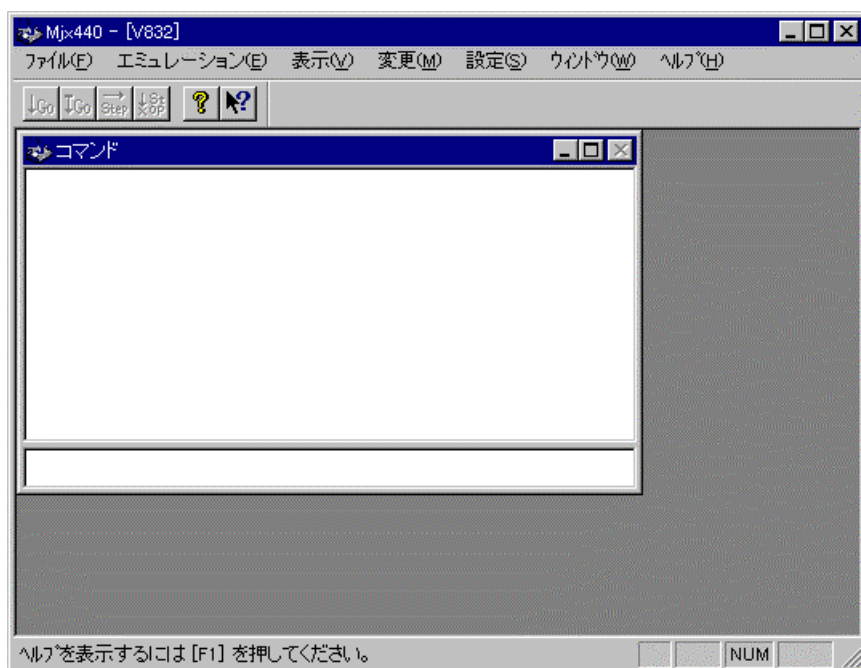
[MJXDEBW](#)を使って MJX440 を操作するためには、スタート メニューから **MJXDEBW** を起動してください。  
手順は、次のとおりです。

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。



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。



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 **MJX832.INI** を指定します。

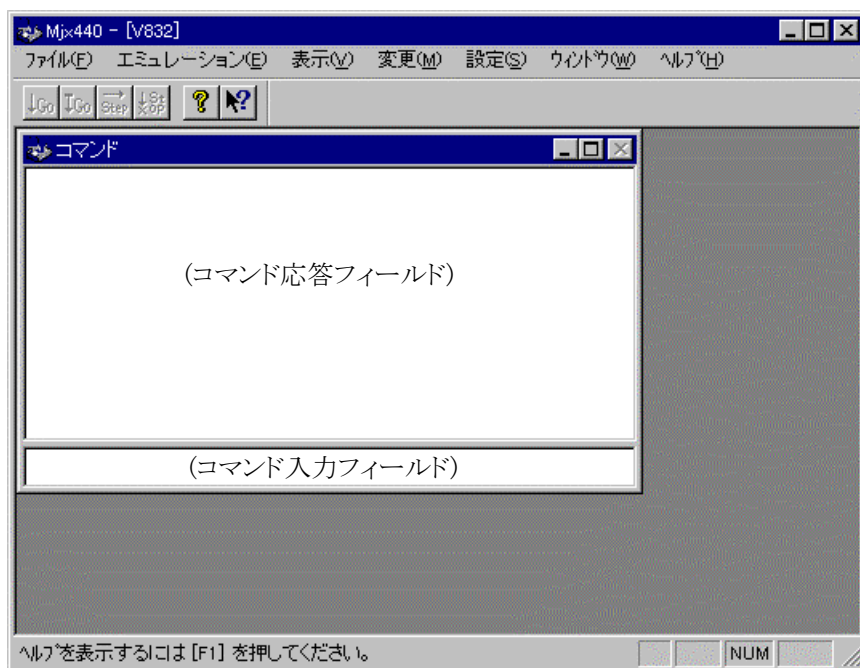
```
MJXDEBW mjsx832.ini
```

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 MJX440 コマンド

MJX440 コマンドは、MJXDEBW のコマンド入力フィールドから入力できるコマンドです。



MJXDEBW ウィンドウ内の、コマンド サブ ウィンドウには、次の機能があります。

- 下位側は、MJX440 コマンドを入力するためのコマンド入力フィールドです。
- 上位側は、MJX440 コマンドの実行結果を表示するコマンド応答フィールドです。

MJX440 コマンドの一覧は、次のとおりです。

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

## 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。

CON	CONFIG と同じです
C	CONFIG と同じです
HE	HELP と同じです
HI	HISTORY と同じです
H	エラーです。HELP か HISTORY の区別ができません

## 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。

GO

REGISTER (レジスタの変更のみ)

STEP

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:

# BATCH

バッチ ファイルに記述されたMJX440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、MJX440 コマンドとして実行します。

引数:

*file* MJX440 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。



## BP

ブレークポイントを表示/設定/解除します

形式:

BP [A H S]	ブレークポイントを表示します。
BP/C <i>num</i> * A* H* S*	ブレークポイントを解除します。
BP/D <i>num</i> * A* H* S*	ブレークポイントを無効にします。
BP/E <i>num</i> * A* H* S*	ブレークポイントを有効にします。
BP/A <i>addr</i> [, <i>space</i> ]	アクセス ブレークポイントを設定します。
BP/H <i>addr</i>	ハードウェア ブレークポイントを表示します。
BP/S <i>addr</i>	ソフトウェア ブレークポイントを表示します。

引数:

<i>num</i>	ブレークポイント番号指定
A	アクセス ブレークポイント指定
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
*	すべてブレークポイント指定
A*	すべてのアクセス 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
<i>addr</i>	ブレークポイント アドレス指定 (16 進数)
<i>space</i>	M がメモリ空間指定、I は I/O 空間指定 (省略時は M)

例:

> BP	(ブレークポイントを表示します)
> BP/C *	(すべてのブレークポイントを解除します)
> BP/D H*	(ハードウェア ブレークポイントを無効にします)
> BP/E A*	(アクセス ブレー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アクセス ブレークポイントを設定します)
> BP/A 1000, I	(0x1000 番地の I/O 空間にアクセス ブレークポイントを設定します)
> BP/H 2000	(0x2000 番地にハードウェア ブレークポイントを設定します)
> BP/S 3000	(0x3000 番地にソフトウェア ブレークポイントを設定します)

補足:

- アクセス ブレークポイントは、最大 4 個まで設定できます。
- ハードウェア ブレークポイントは、命令上に最大 2 個まで設定できます。
- ソフトウェア ブレークポイントは、命令上に最大 128 個まで設定できます。

# 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:

# CONFIG

MJX440 の環境を表示/変更します

形式:

CONFIG                      現在設定されている MJX440 の環境を表示します。  
CONFIG/S *item=value*      MJX440 の環境設定項目 *item* を *value* に設定します。

引数:

<i>item</i>	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
EPCAVAIL	EPC の使用～割り込みルーチンのトレース表示のとき
<i>value</i>	環境設定項目に設定する値、
12.5 または 25	JCLOCK の場合
16 進数アドレス	TOPADDR の場合
0 または 1	EPCAVAIL の場合 (0=無効 1=有効)

例:

> CONFIG	(MJX440 の環境を表示します)
> CONFIG JCLOCK=12.5	(JTAG クロックを 12.5MHz に設定します)
> CONFIG TOPADDR=20000	(ROM の先頭アドレスを 0x20000 に設定します)
> CONFIG EPCAVAIL=0	(割り込みルーチンのトレース表示のとき EPC を使用しない)

補足:

- EPCAVAIL は、V831 のときのみ有効です。

# 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。
- サイズ省略時は、前回のサイズが適用されます。
- 内蔵命令 RAM の表示はできません。

# EXAMINE

メモリの内容を変更します

形式:

EXAMINE[ /B|/W|/L] *addr*=*data*

アドレス *addr* のメモリをデータ *data* に変更します。

引数:

/B	8 ビット指定
/W	16 ビット指定
/L	32 ビット指定
<i>addr</i>	メモリ変更開始アドレス (16 進数)
<i>data</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。)
> EXAMINE/L 2000          (0x2000 番地からのメモリの内容を、対話型で変更します)
00002000 00000000 11223344
00002004 00000000 55667788
00002008 00000000 .      (ピリオドのみで終了します)
```

補足:

- 対話型の場合、ピリオドのみの入力で終了します。
- サイズ省略時は、前回のサイズが適用されます。
- 内蔵命令 RAM の変更はできません。

# 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 0x000 でフィルします)

補足:

- 内蔵命令 RAM のフィルはできません。
- 8M バイト以上のフィルはできません。(制限事項)

# 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:

- 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。
- プログラムがブレークポイントで停止しない場合は、スペース キーを押してください。強制停止できます。



# 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/D [ <i>start</i> [, <i>end</i> ]]	パケット番号 <i>start</i> から <i>end</i> 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。
HISTORY/P [ <i>start</i> [, <i>end</i> ]]	パケット番号 <i>start</i> から <i>end</i> 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。

引数:

<i>start</i>	表示開始パケット番号 (10 進数)
<i>end</i>	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/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)
> 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)

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 トレース トリガ点  
エンド トリガ トレース トリガ点  
ミッド トリガ トレース トリガ点  
インナー トリガトレース開始トリガ点

- (V831 逆アセンブル表示の制限) アドレスが確定できないパケットについては、パケット番号を指定しても、逆アセンブル表示されません。

# INIT

MJX440 を再初期化します

形式:

INIT

MJX440 を再初期化します。

補足:

- CPU もリセットされます。

# 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:

# 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<sup>\*1</sup>、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。

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。
- 内蔵命令 RAM へのダウンロードフィルはできません。
- `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。

---

<sup>\*1</sup> 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
<i>addr1</i>	転送元メモリ開始アドレス (16 進数)
<i>addr2</i>	転送元メモリ終了アドレス (16 進数)
<i>addr3</i>	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:

- 内蔵命令 RAM のブロック転送はできません。
- 8M バイト以上のブロック転送はできません。(制限事項)

# PIN

ピンを有効/無効にします

形式:

PIN                                      ピンの有効/無効の状態を表示します。  
PIN *pinname*=EN|DI                  ピン *pinname* を有効または無効にします。

引数:

*pinname*    ピン名、次のいずれかを指定

INTP00

INTP01

INTP02

INTP03

INTP10

INTP11

INTP12

INTP13

NMI

HLDRQ

RESET

EN            有効にする

DI            無効にする

例:

> PIN                                      (ピンの有効/無効の状態を表示します。)

> PIN INTP11=DI                      (ピン INTP11 を無効にします。)

補足:

# PORT

I/O ポートの内容を表示/変更します

形式:

PORT[/B|/W|/L] *addr*      アドレス *addr* の I/O ポートの内容を表示します。

PORT[/B|/W|/L] *addr*=*data*      アドレス *addr* の I/O ポートをデータ *data* に変更します。

引数:

/B      8 ビット指定

/W      16 ビット指定

/L      32 ビット指定

*addr*      I/O ポートアドレス (16 進数)

*data*      I/O ポート変更データ (16 進数)

例:

> PORT/L 2000      (0x2000 番地からの I/O ポートの内容を表示します。)

> PORT/W 1000=55      (0x1000 番地の I/O ポートを 16 ビットデータ 0x0055 に変更します)

補足:



# 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。

# REGISTER

レジスタの内容を表示/変更します

形式:

REGISTER                      レジスタの内容を表示します。  
REGISTER *reg*=*data*        レジスタ *reg* をデータ *data* に変更します。

引数:

*reg*        レジスタ名指定  
            「[付録 G レジスタ名一覧](#)」を参照してください  
*data*       レジスタ変更データ (16 進数)

例:

> REGISTER                      (レジスタの内容を表示します。)  
> REGISTER PC=2000            (PC を 0x2000 に変更します)

補足:

# 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:

# TRACE

リアルタイム トレースのモードを表示/設定します

形式:

TRACE	トレース条件をすべて表示します。
TRACE/A <i>addr</i> [ , <i>space</i> ]	データ トレースを設定します。
TRACE/C <i>addr</i>	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。
TRACE/M <i>mode</i> [ , <i>mask</i> ]	トレース モードを設定します。
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>taddr</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>taddr</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>taddr</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>length</i> ]	実行トレース条件をインナー トリガに設定します。(V832 のみ)
TRACE/I DI	実行トレース条件を中止します。
TRACE/R	トレース条件をすべて無効にします。

引数:

<i>addr</i>	データ トレースをするアクセス アドレス指定 (16 進数)
<i>space</i>	M がメモリ空間指定、I は I/O 空間指定 (省略時は M)
<i>mode</i>	トレース モード指定、次のいずれかを指定
RT	通常モード
FL	完全モード
<i>mask</i>	トレース要因指定、次のビットごとに 16 進数で指定 (省略時は 1FF0)
0010	例外発生
0020	割り込み発生
0040	条件分岐命令による分岐
0080	PC 相対分岐命令による分岐
0100	JAL 命令による分岐
0200	RETI 命令による分岐 (分岐先 PC)
0400	RETI 命令による分岐 (分岐元 PC)
0800	レジスタ間接分岐命令による分岐 (分岐先 PC)
1000	レジスタ間接分岐命令による分岐 (分岐元 PC)
<i>length</i>	トレースするワード長指定 (10 進数、省略時 131072)

*taddr*    トレース トリガのアドレス指定 (16 進数)  
*saddr*    トレースを開始するトレース トリガのアドレス指定 (16 進数)  
*eaddr*    トレースを中止するトレース トリガのアドレス指定 (16 進数)

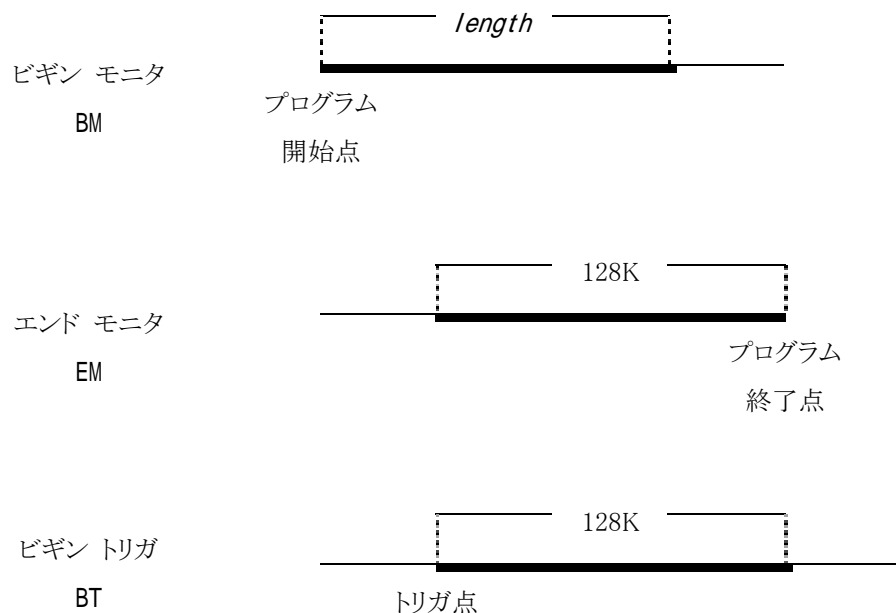
例:

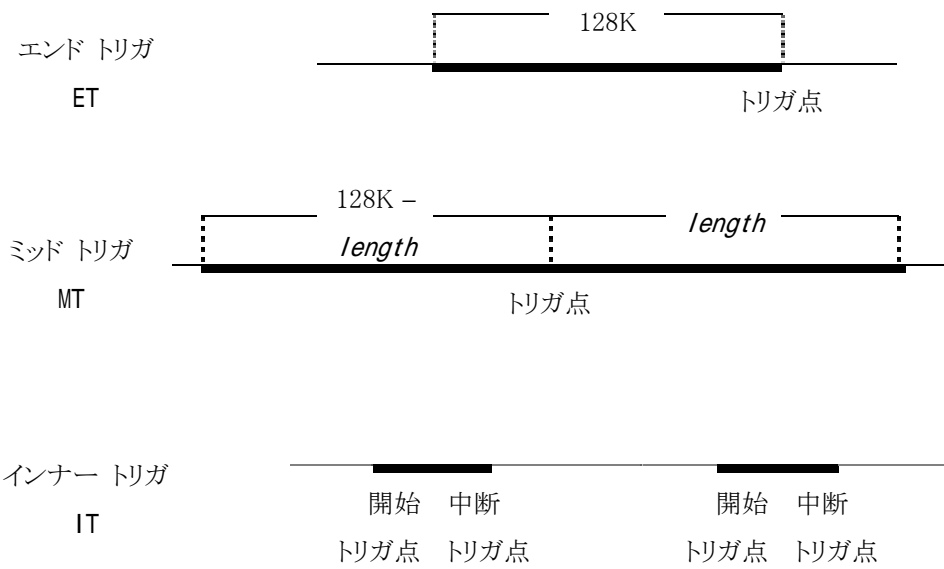
> TRACE                    (トレース条件をすべて表示します。)  
> TRACE/A 1000            (0x1000 番地のアクセスをトレースする、データ トレースを設定します。)  
> TRACE/C \*                (すべてのデータ トレースを解除します。)  
> TRACE/M FL              (トレース モードを完全モードに設定します。)  
> TRACE/I BT,1000         (実行トレース条件をビギン トリガに、トレース トリガのアドレスを 0x1000 に設定します。)  
> TRACE/R                  (トレース条件をすべて無効にします。)

補足:

- 完全トレース モード FL では、データ トレースは無効になります。
- 実行する命令によって、トレース メモリに格納できるパケット数が変わります。

トレース範囲:





# 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。
- 内蔵命令 RAM の逆アセンブル表示はできません。

# VERSION

ソフトウェアのバージョンを表示します

形式:

VERSION

ソフトウェアのバージョンを表示します



# 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。

XP IN

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 <i>ch, level</i>	外部トリガ ケーブルの出力信号 <i>ch</i> をレベル <i>level</i> 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 [MJX440 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)	メモリとレジスタ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)...

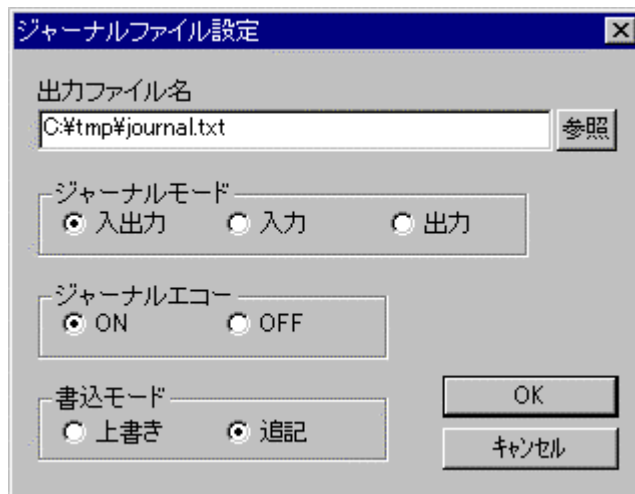
---

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)...

---

JOURNAL

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)...

---

G0 *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 [G0](#)」参照。

## エミュレーション(E) ～ 再スタート(R)

---

G0

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 [G0](#)」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)...

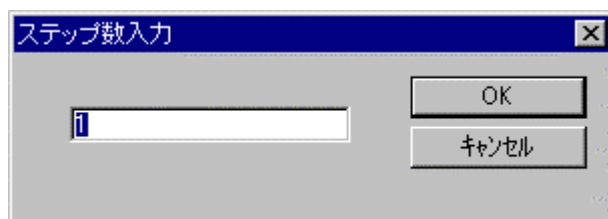
---

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:

- 「第七章 [STEP](#)」参照。

## エミュレーション(E) ～ ダウンロード(L)...

---

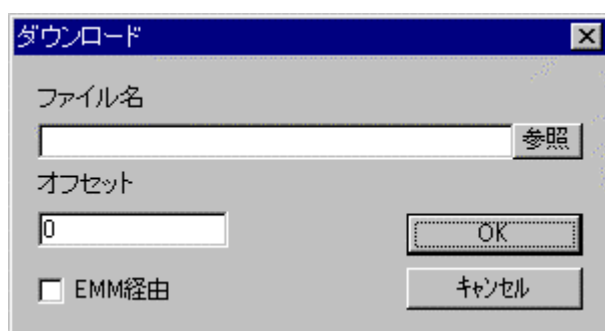
LOAD

Ctrl+L

機能:

MJX バイナリ<sup>\*1</sup>、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。
- 「第七章 [LOAD](#)」参照。

---

<sup>\*1</sup> 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:

[MJXDEBW](#)を再初期化します。

補足:

- 「第七章 [INIT](#)」参照。

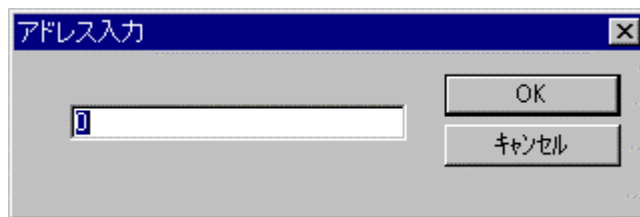
## 表示(V) ～ メモリ(M)...

DUMP

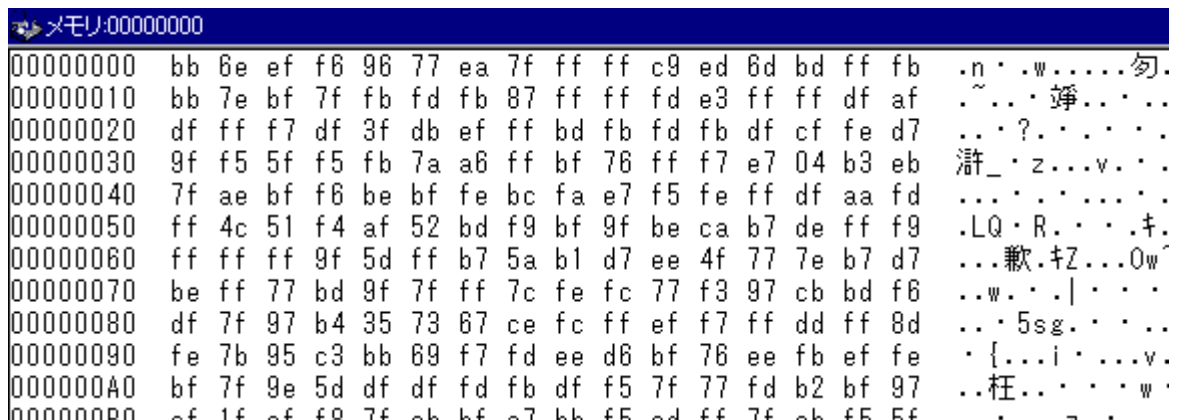
機能:

メモリの内容を表示する、メモリ ウィンドウを開きます。

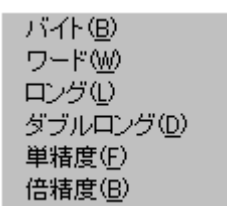
操作:



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。



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。

## 表示(V) ～ レジスタ(R)

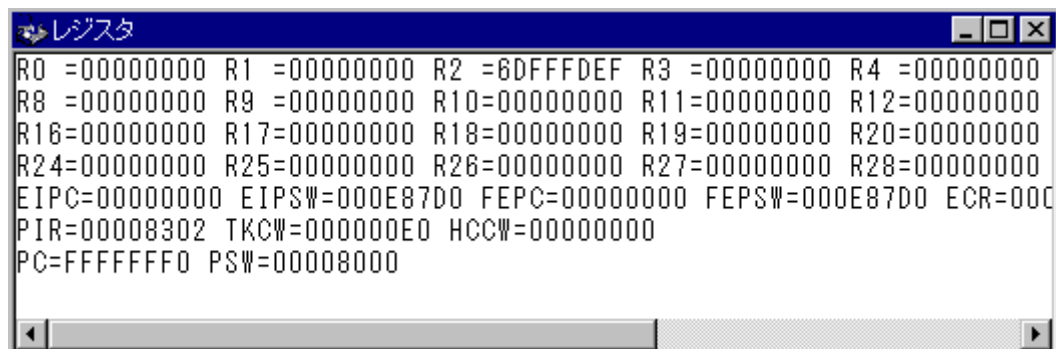
---

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:



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

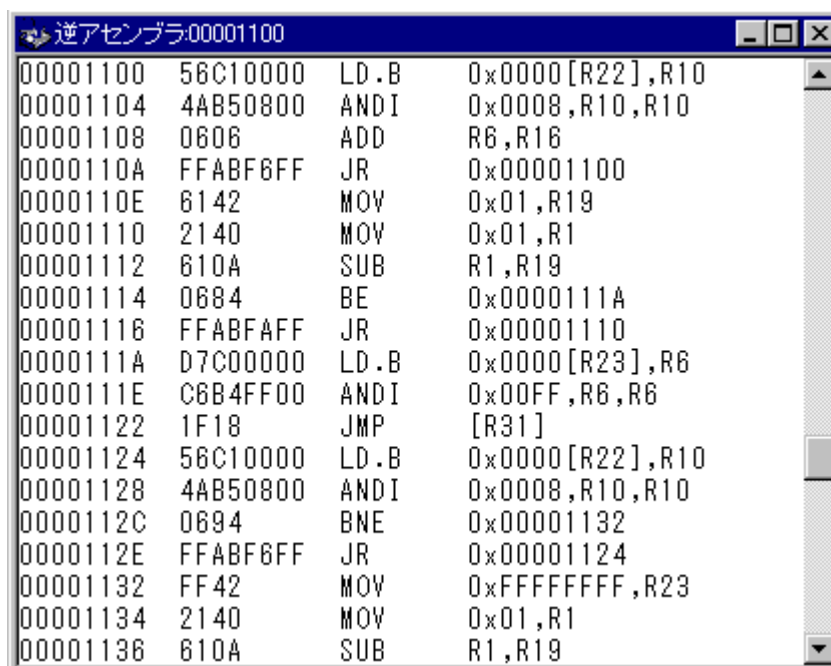
機能:

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、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。

## 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:

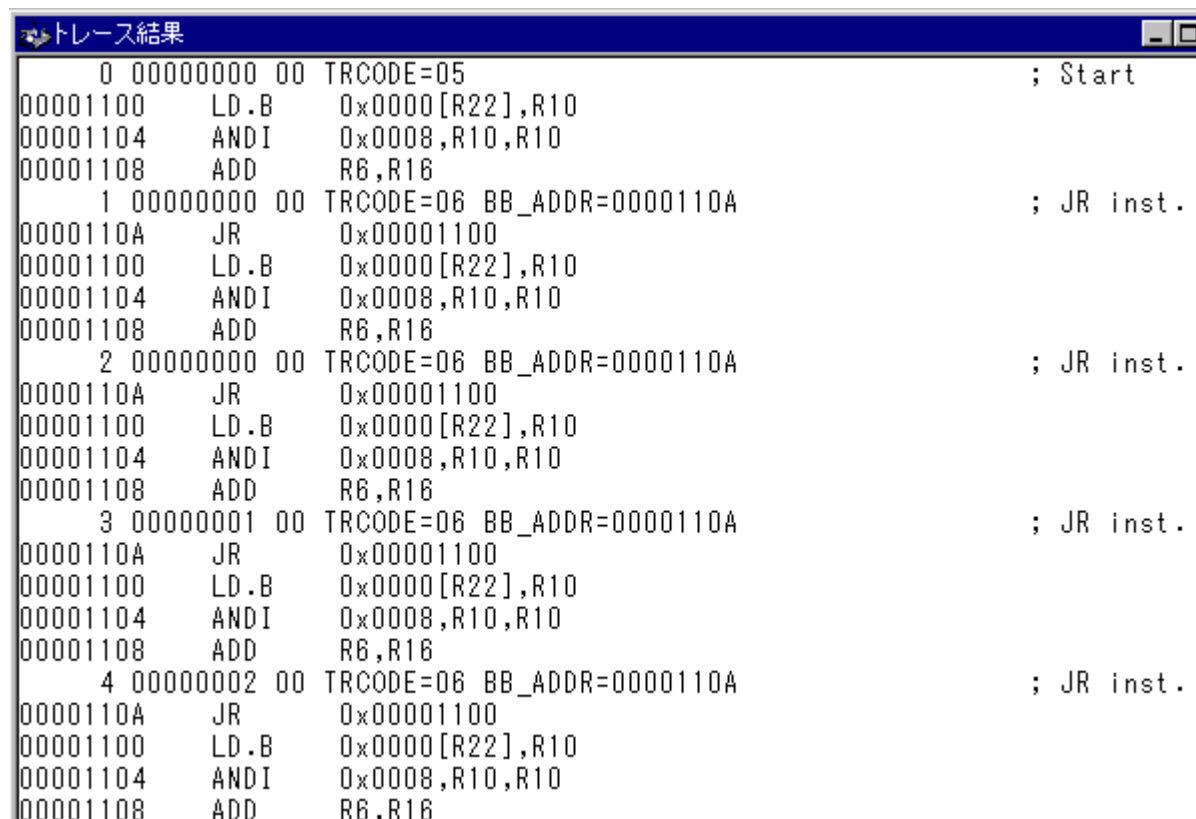


トレース結果出力条件

実行トレース条件	ビギンモニター	OK
パケット範囲	0->0	キャンセル
表示開始パケット番号	0	
表示モード	パケット	

ダイアログボックスで、表示開始サイクル、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。



0	00000000	00	TRCODE=05		; Start
00001100	LD.B		0x0000[R22],R10		
00001104	ANDI		0x0008,R10,R10		
00001108	ADD		R6,R16		
1	00000000	00	TRCODE=06 BB_ADDR=0000110A		; JR inst.
0000110A	JR		0x00001100		
00001100	LD.B		0x0000[R22],R10		
00001104	ANDI		0x0008,R10,R10		
00001108	ADD		R6,R16		
2	00000000	00	TRCODE=06 BB_ADDR=0000110A		; JR inst.
0000110A	JR		0x00001100		
00001100	LD.B		0x0000[R22],R10		
00001104	ANDI		0x0008,R10,R10		
00001108	ADD		R6,R16		
3	00000001	00	TRCODE=06 BB_ADDR=0000110A		; JR inst.
0000110A	JR		0x00001100		
00001100	LD.B		0x0000[R22],R10		
00001104	ANDI		0x0008,R10,R10		
00001108	ADD		R6,R16		
4	00000002	00	TRCODE=06 BB_ADDR=0000110A		; JR inst.
0000110A	JR		0x00001100		
00001100	LD.B		0x0000[R22],R10		
00001104	ANDI		0x0008,R10,R10		
00001108	ADD		R6,R16		

(逆アセンブラ表示の例)

- 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。

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	トリガー点
エンド トリガー	トリガー点
ミッド トリガー	トリガー点

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。

## EJTagトレース結果表示フォーマット

## サイクル表示フォーマット

88



#### (1)サイクル番号

CPU サイクルの番号を示します。

零点はトレースモードにより異なります。

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

ビギントリガー - トリガー点

エンドトリガー - トリガー点

ミッドトリガー - トリガー点

#### (2)外部入力信号

外部入力 EXT11,EXT12 および EXT13 信号のレベルを示します。

CPU サイクルとの関連により外部信号の遷移状態が判ります。

外部入力信号はトレースクロックに同期して捕捉されます。

トレースクロック = CPU クロック/2

トレースクロック = CPU クロック/4

の場合は、外部入力信号は 2 サイクルまたは 4 サイクル毎に表示されます。

#### (3)PCST

CPU の実行状態を示す簡単な文字列を表示します。

PCST コード、シンボル、表示文字列と CPU 実行状態を以下の表に示します。

PCST コード	シンボル	表示	CPU 実行状態
111	STL	Pipeline Stall	パイプラインストール
110	JMP	Jump instruction	分岐命令実行および PC アドレス出力の開始
101	BRT	Branch taken	直接分岐命令の実行(PC アドレス出力なし)
100	EXP	Exception	例外発生および例外コード出力開始
011	SEQ	Sequential	非分岐命令実行
010	TST	Trigger at stall	トリガー発生(パイプラインストール中)
001	TSQ	Trigger at exec	トリガー発生(非分岐命令実行中)
000	DBM	Debug mode	デバッグモード

#### (4)アドレス情報表示領域

##### (i)PCST が JMP の場合

分岐命令による分岐先のアドレスと ASID(ASID トレースが有効な場合)が表示されます。

JMP ステータスに続くアドレス出力中に再度 JMP ステータスが発生した場合、

先行する JMP ステータスに伴うアドレス出力は中止されます。

この場合、下位の有効なビット数が表示されます。分岐先アドレスの決定は出来ません。

##### (ii)PCST が EXP の場合

例外コードと例外ベクタアドレスが表示されます。

EXP ステータスに続く例外コード出力中に再度 EXP ステータスが発生した場合、  
先行する EXP ステータスに伴う例外コード出力は中止されます。

この場合、例外ベクタアドレスの表示は行いません。

例外コードは 32bitCPU の場合 3bit、64bitCPU の場合 4bit です。

(iii)その他の場合

アドレス情報はありません。

## 2 アセンブラ表示

### アセンブラ表示フォーマット

```
____1 12345678 9ABCDEF0 XXXXXXXXXXXXXXXX
|           |           |           |
|           |           |           |
|           |           |           +----- アセンブラニーモニック
|           |           +----- 機械語
|           +----- PC アドレス
+----- サイクル番号
```

#### (1) サイクル番号

CPU サイクルの番号を示します。

零点はトレースモードにより異なります。

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

ビギントリガー - トリガー点

エンドトリガー - トリガー点

ミッドトリガー - トリガー点

#### (2) PC アドレス

命令のアドレスを示します。

#### (3) 機械語

命令の機械語コードを示します。

#### (4) アセンブラニーモニック

アセンブラニーモニックを示します。

### 注意事項

#### (1) アドレス喪失

JMP ステータスに続くアドレス出力中に再度発生した JMP ステータスにより

先行する JMP ステータスに伴うアドレス出力が中止された場合、アドレスの決定は出来ません。

再度完全なアドレスが得られるまで、アドレス喪失状態となりこの間の

アセンブラ表示は出来ません。

中止された例外コード出力に付いても同様です。

#### (2) 命令-PCST ミスマッチ

アセンブラ表示解析中に PCST と実際の命令(解析したアドレスに従う)が合わなかった場合、

すなわち一方が分岐命令を示しているのもう一方が非分岐命令を示しているときは、

トレース捕捉回路の異常と考え、トレース結果アセンブラ表示は中止されます。

## 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。

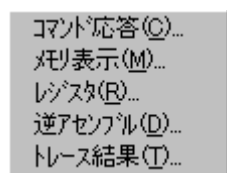
## 表示(V) ～ オプション(O) ～ フォント(F)

---

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)...

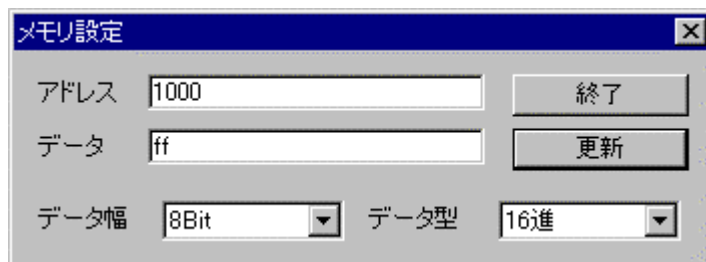
---

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。

## 変更(M) ～ レジスタ(R)...

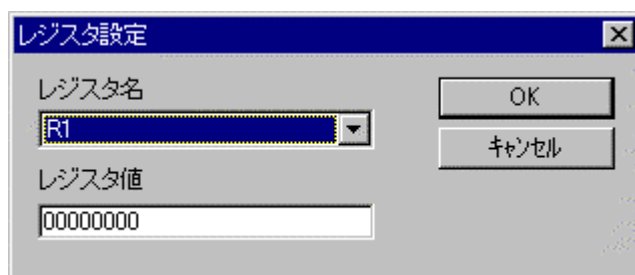
---

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。



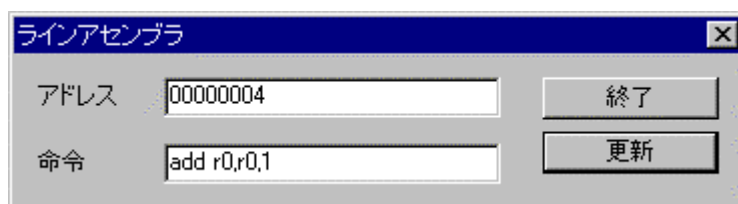
## 変更(M) ～ アセンブラ(A)...

---

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。

## 変更(M) ～ フィル(F)...

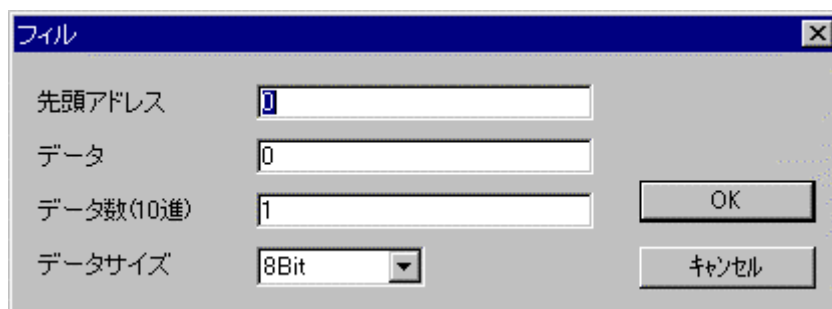
---

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。

## 変更(M) ～ 移動(M)...

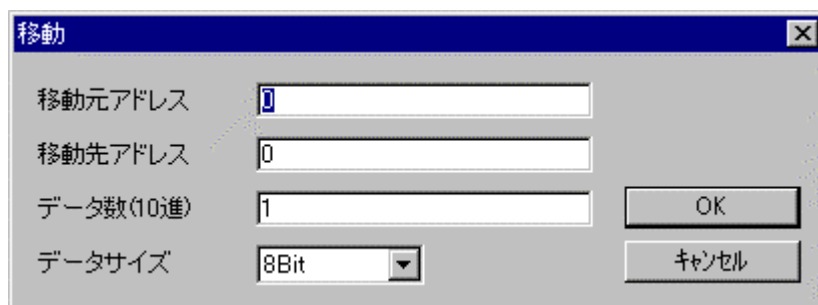
---

MOVE

機能:

メモリの内容をブロック転送します。

操作:



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。

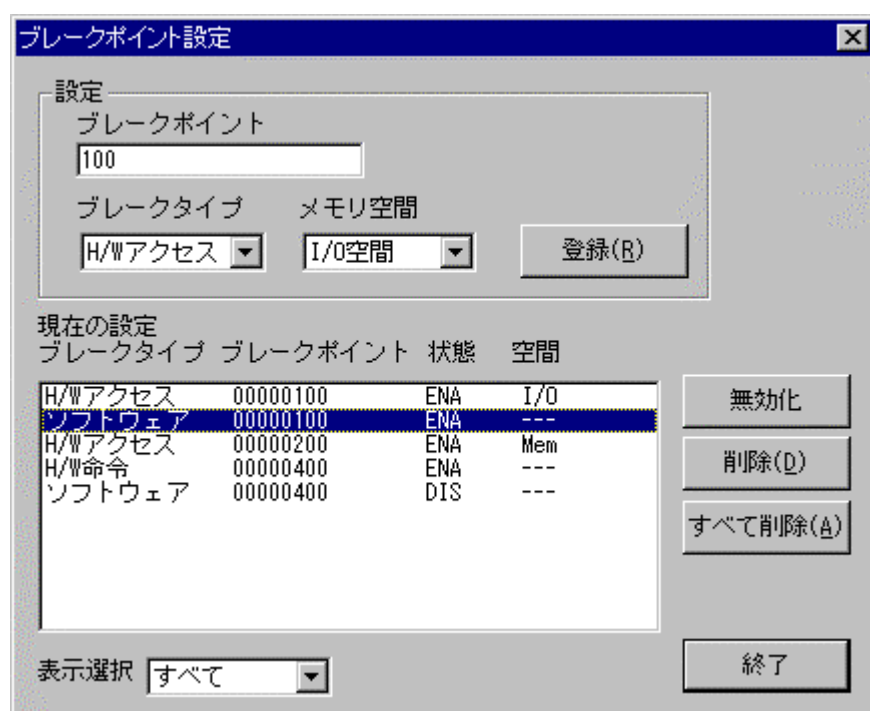
## 設定(S) ～ ブレークポイント(B)...

BP

機能:

ブレークポイントを設定します

操作:



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。

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。

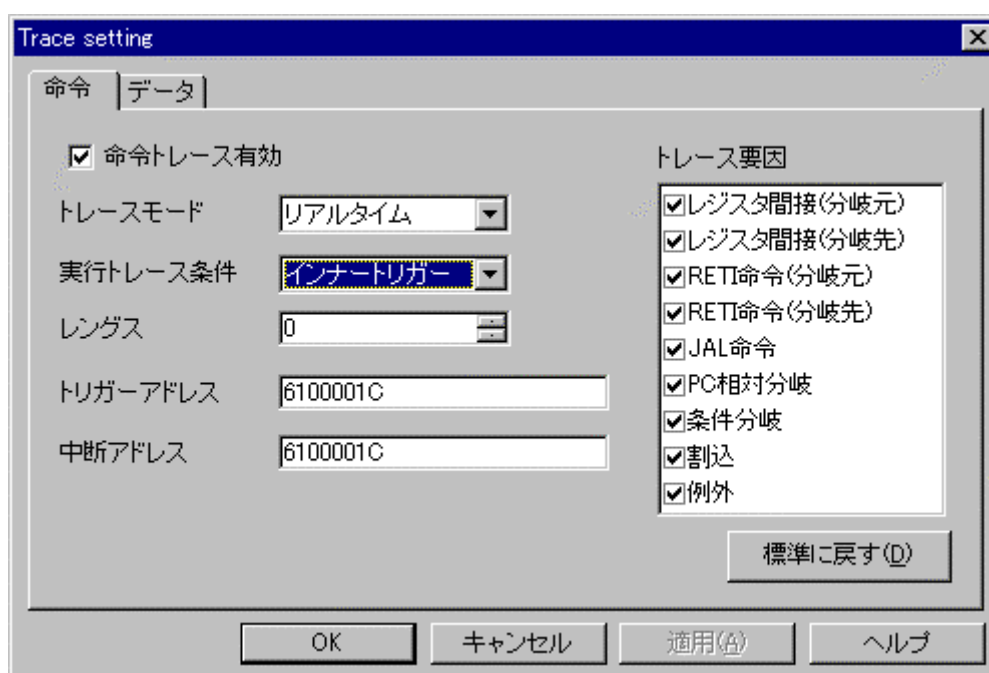
## 設定(S) ～ トレース(T)...

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: 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。

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  
インナートリガー

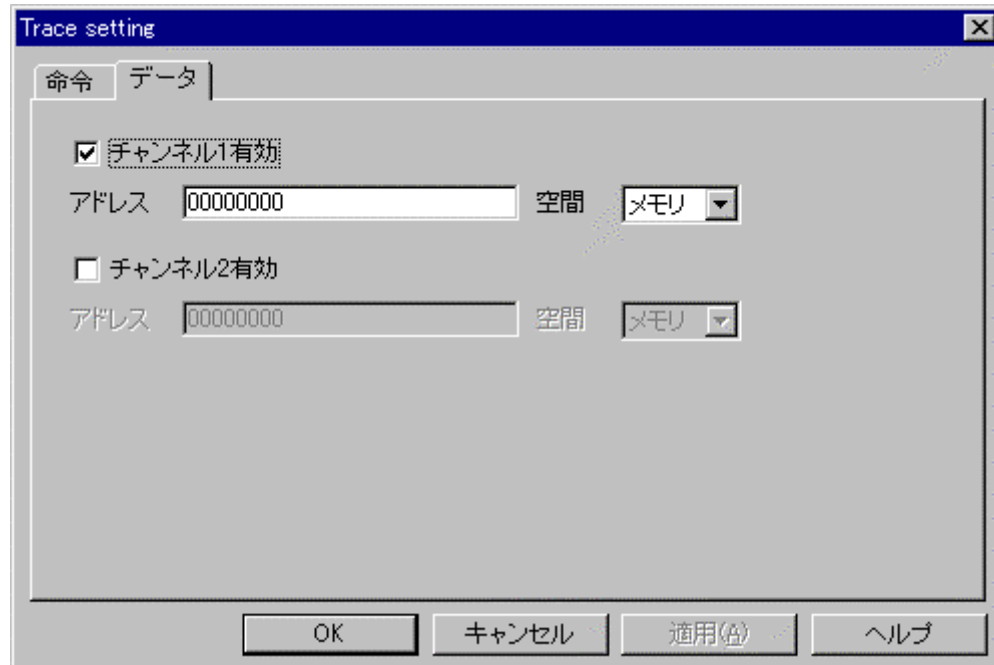
レンジ ス トレースバッファの長さを指定します。長さの単位は CPU クロックです。「トリガーモ

ード」がビギンモニター／ミッドトリガー／インナートリガーのとき設定できます。「レングス」に設定できる値は 1～131071 です。(両エッジクロックトレースの場合)

トリガーアドレス      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 トリガーの場合指定できます。

中断アドレス          中断アドレス情報を設定します。「トレース モード」がインナートリガーの場合設定できます。

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、メモリ、I/O のいずれかを選びます。

補足：

- 「第七章 [TRACE](#)」参照。

## 設定(S) ～ 設定の読出(L)...

---

SETLOAD

機能:

ブレイクポイントの設定とトレースの設定を、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。
- 「第七章 [JOURNAL](#)」参照。



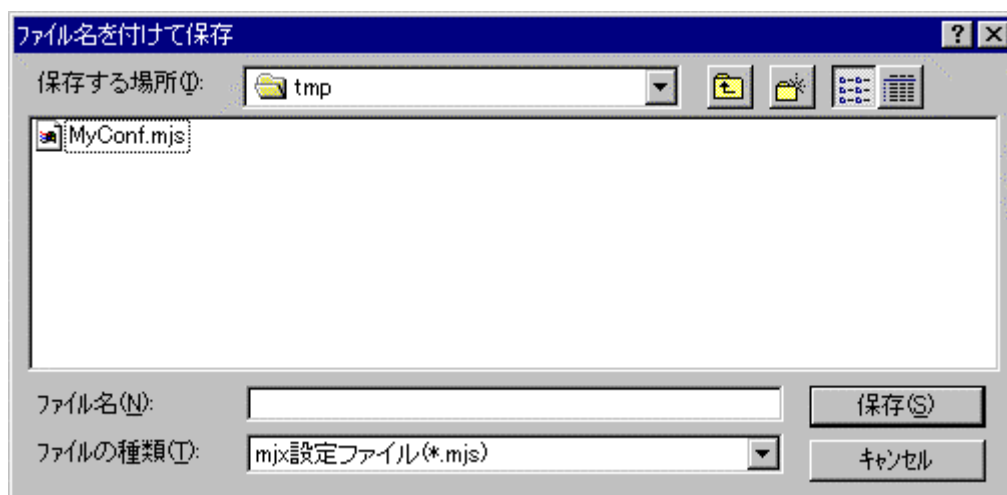
## 設定(S) ～ 設定の保存(S)...

SETSAVE

機能:

ブレイクポイントの設定とトレースの設定を、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:

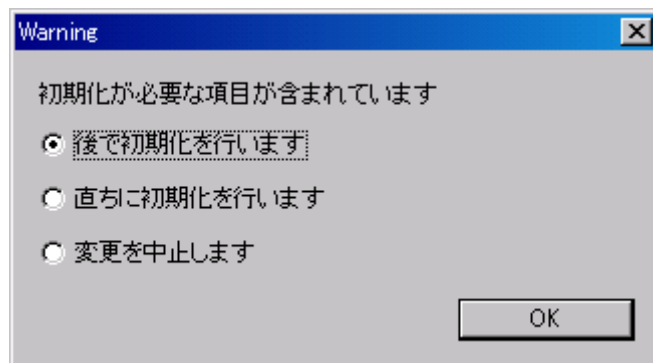
[MJXDEBW](#)の環境を設定します。

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)

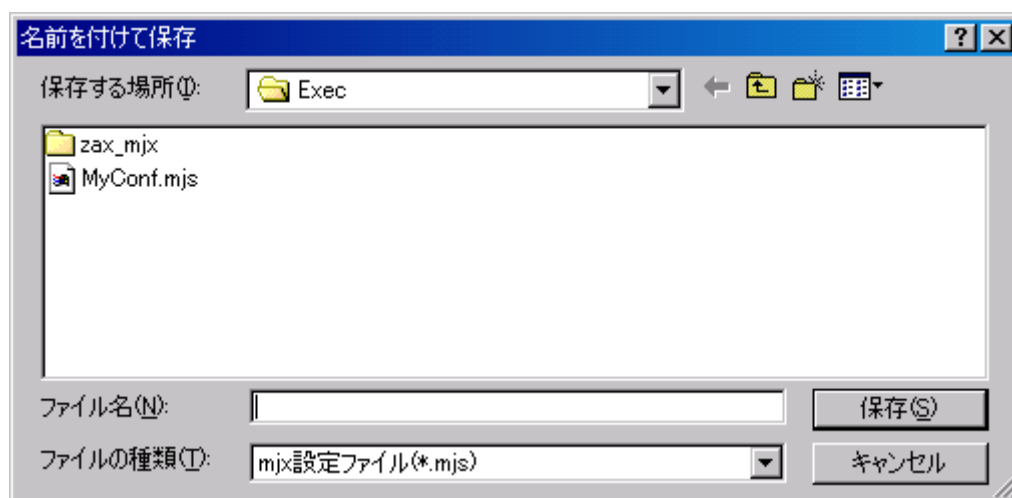
---

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)

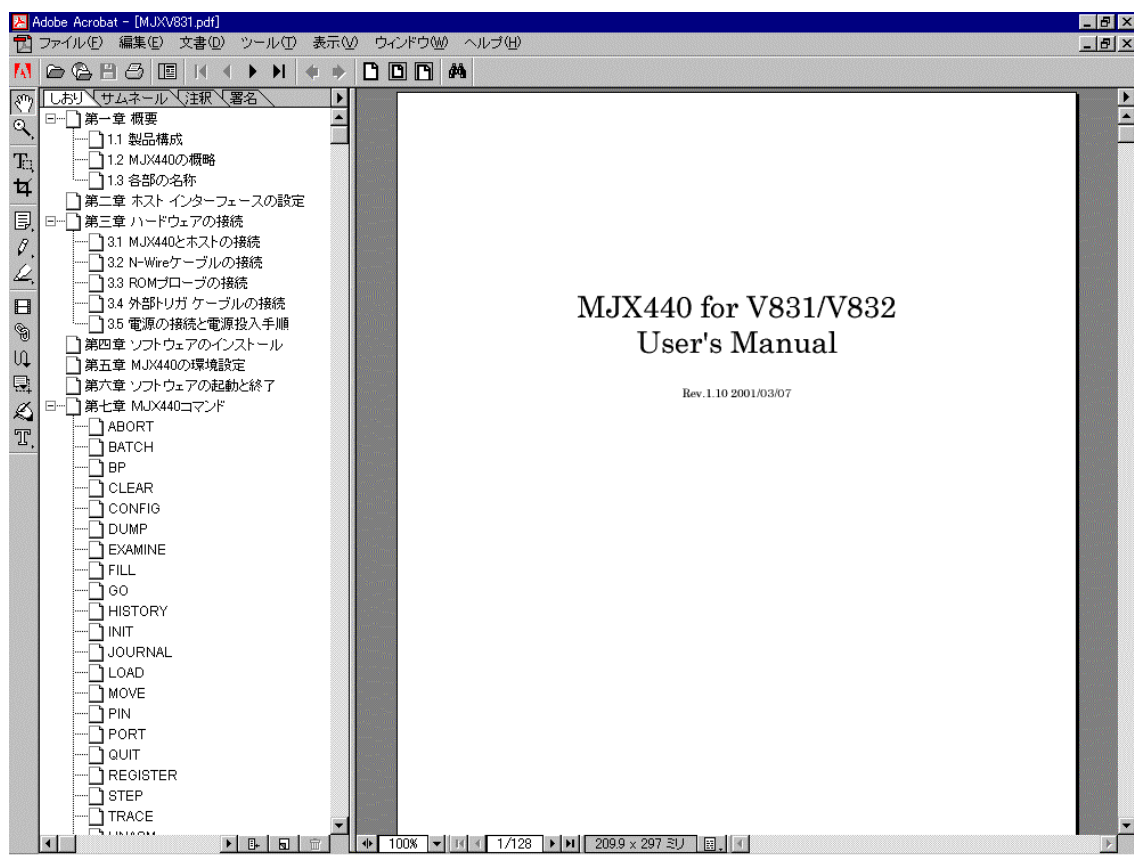
---

機能：

MJX440 for V831/V832 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。

- N-Wire ケーブル接続時、440K バイト/秒 (RAM 領域)
- ROM インサーキット接続時、4M バイト/秒 (ROM 領域)

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 .mjx 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 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。

```
mjx> load myfile.mjx
```

## 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℃～35℃
保存温度範囲	-10℃～55℃
周囲湿度範囲	30%～85%
対応 ROM	<a href="#">「付録 D ROM プローブ」</a> 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 (16M バイトまで増設可)
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 カード、または PCMCIA カード)
ターゲット インターフェース	<a href="#">N-Wire コネクタ</a> ROM ソケット
対応 CPU	V831、V832
ダウンロード速度	440K バイト/秒 (N-Wire)、 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
ブレークポイント機能	命令フェッチ      x 2 ソフトウェア      x 100 メモリ アクセス      x 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 <a href="#">トレース条件</a> 6 種 タイム スタンプ      32bit
制限事項	<a href="#">「付録 B ターゲット システムの制限事項」</a> 参照

## 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。
- ターゲット システムのバス クロックより、N-Wire クロック(DCK 信号)の方が遅い。

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 が複数個実装されている場合、すべての ROM のアドレス信号が同一信号である。

## 付録 C N-Wire コネクタ

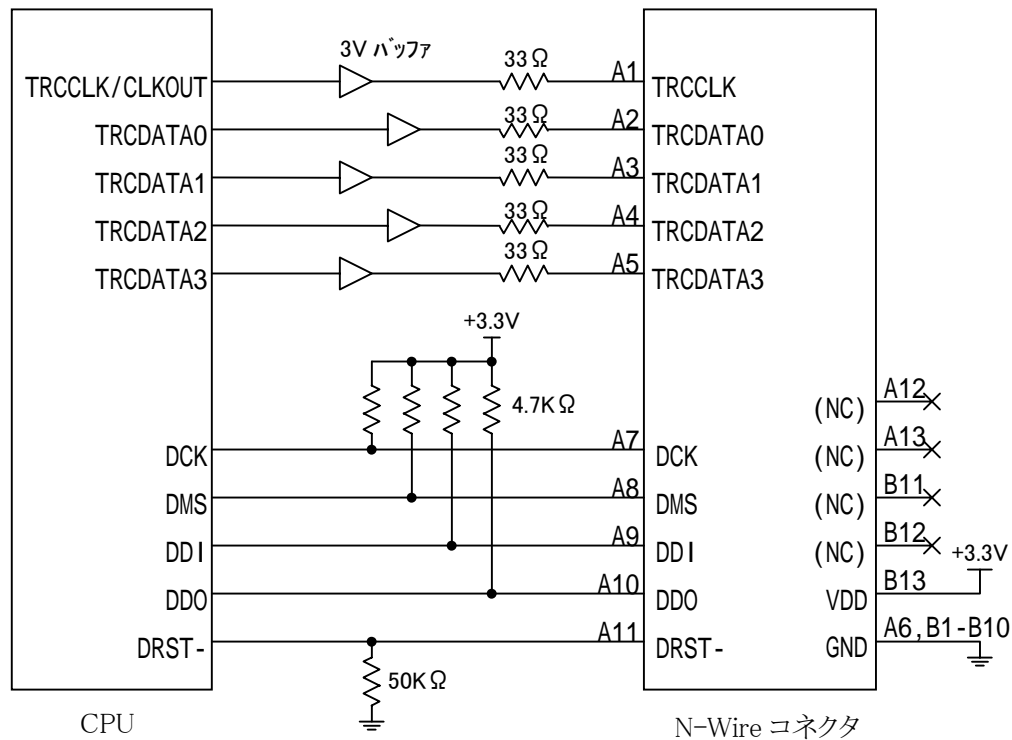
ピンアサイン

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

## 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。
- MJX400 は、B13 VDD を電源の状態検出のみに使用しています。したがって、余分な電源容量は必要ありません。

## 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

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

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

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)	$\mu$ 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

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

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

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
N-Wire (赤)	N-Wire コネクタの電源が供給されているとき点灯
EXI1 (赤)	<a href="#">外部トリガ ケーブル</a> の EXTIN1 信号が HIGH レベルのとき点灯
EXI2 (赤)	<a href="#">外部トリガ ケーブル</a> の EXTIN2 信号が HIGH レベルのとき点灯
EXI3 (赤)	<a href="#">外部トリガ ケーブル</a> 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

R0	R16	PC
R1	R17	PSW
R2	R18	EIPC
R3	R19	EIPSW
R4	R20	FEPC
R5	R21	FEPSW
R6	R22	ECR
R7	R23	PIR
R8	R24	TKCW
R9	R25	HCCW
R10	R26	
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)

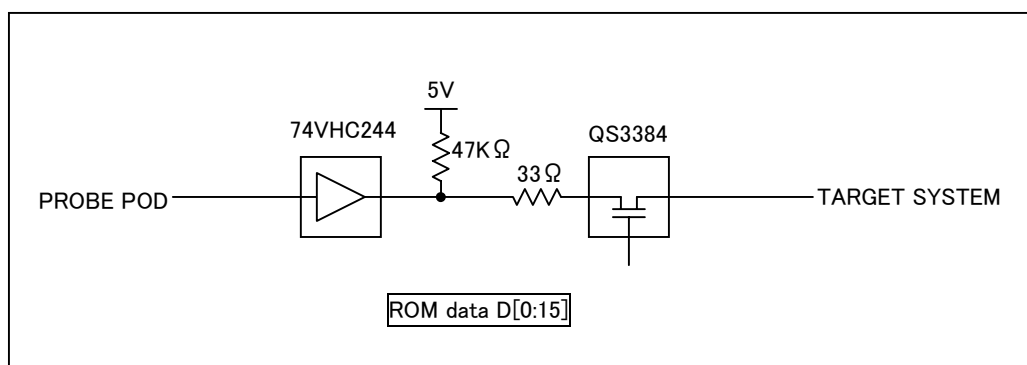
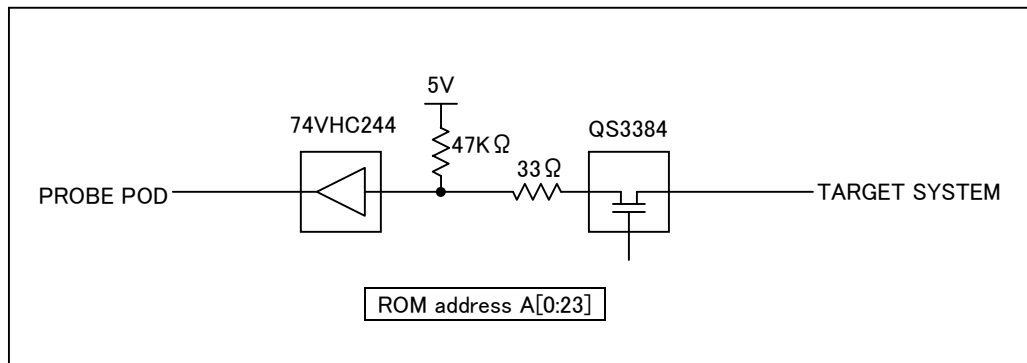
データ (可変、データ バイト長)

XX	XX	XX	XX	XX	XX	XX	...	XX	XX	XX	XX	XX	XX	XX	XX
----	----	----	----	----	----	----	-----	----	----	----	----	----	----	----	----

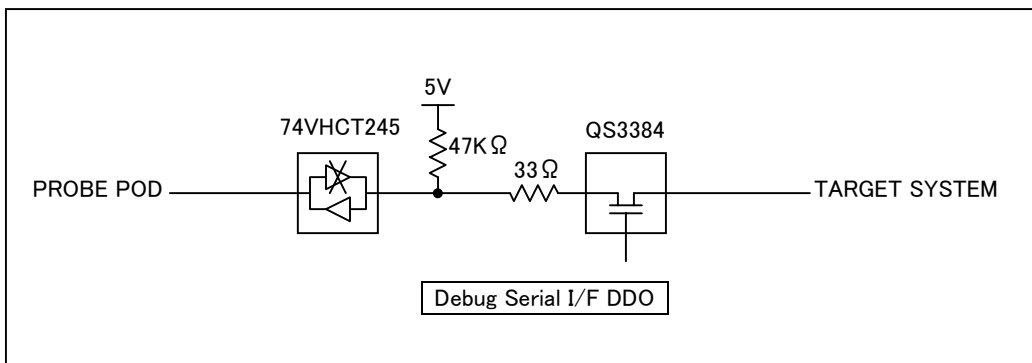
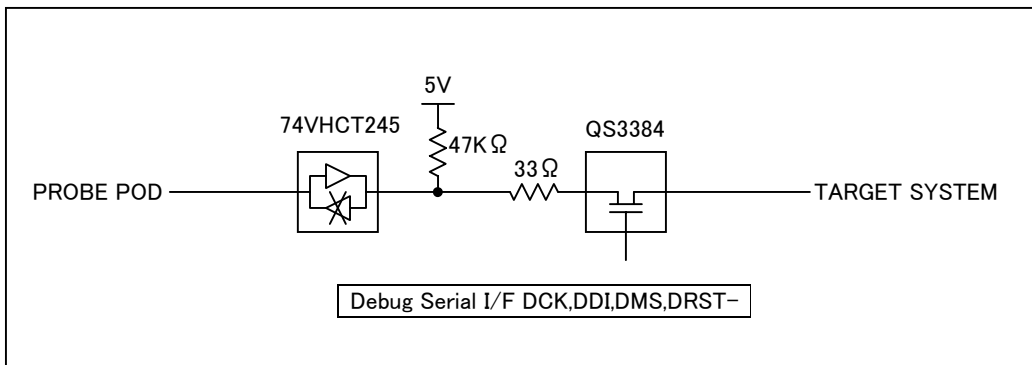
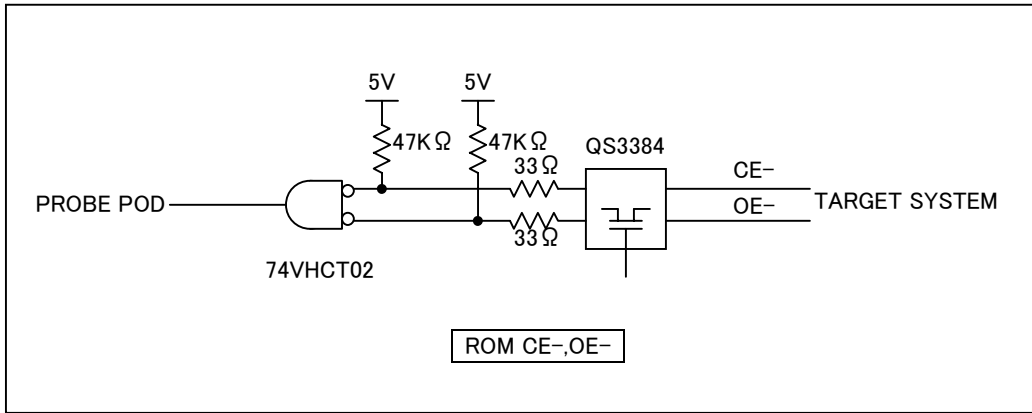
終了ヘッダ (16 バイト長)

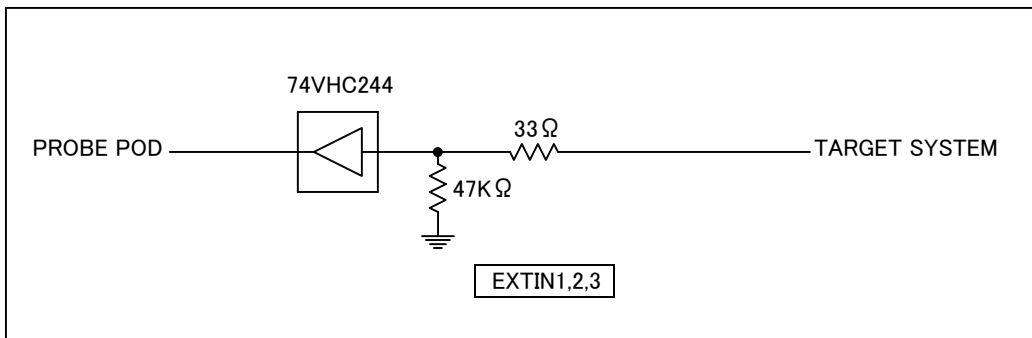
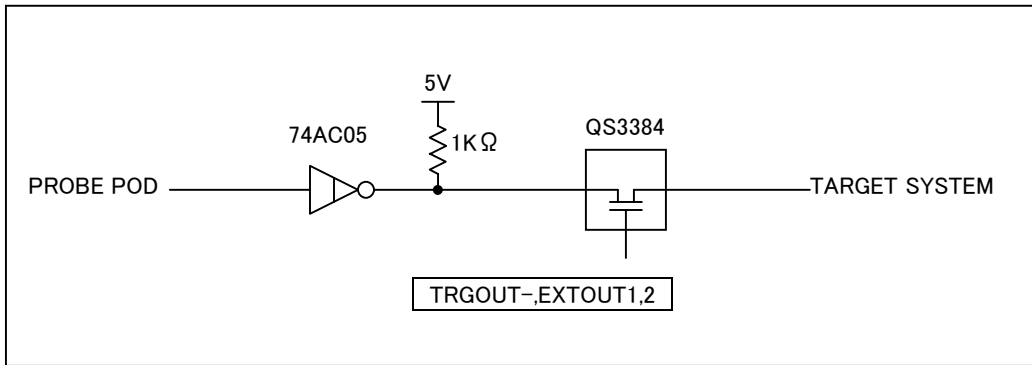
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
'M'	'J'	'1'	00	00	00	00	00	00	00	00	00	00	00	00	00

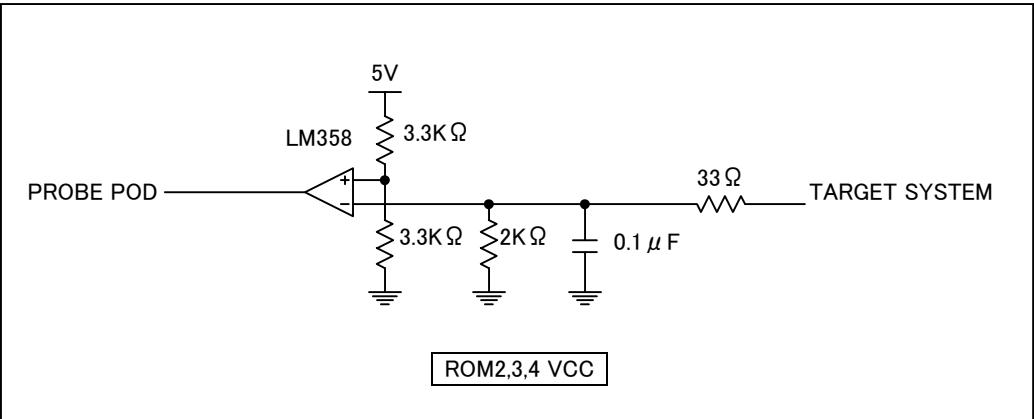
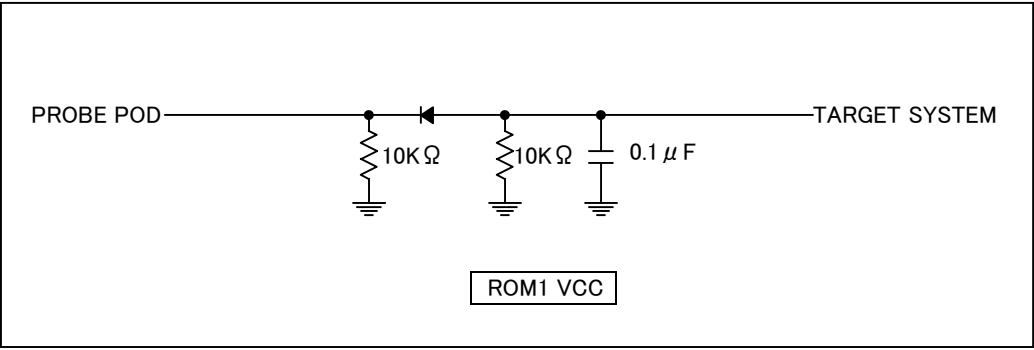
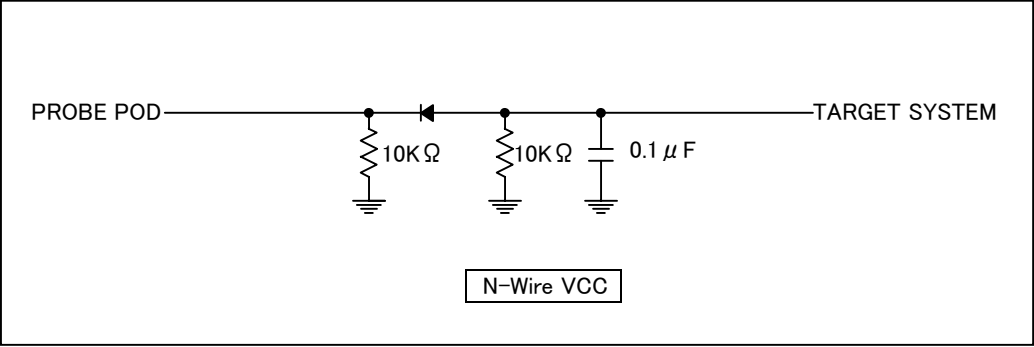
## 付録I ターゲット システムへのプロービング



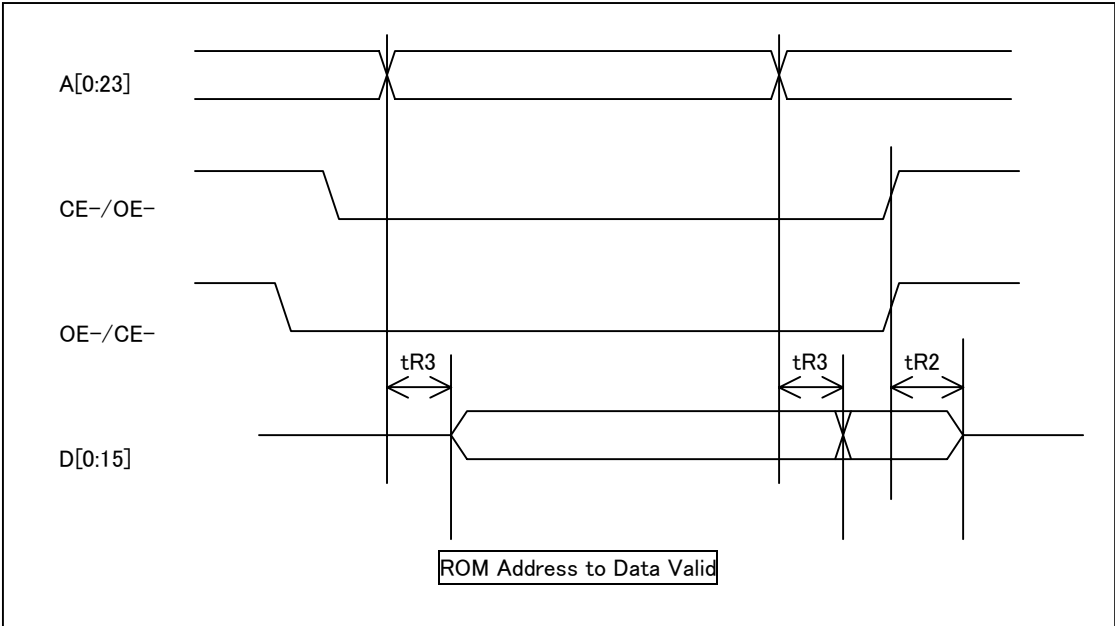
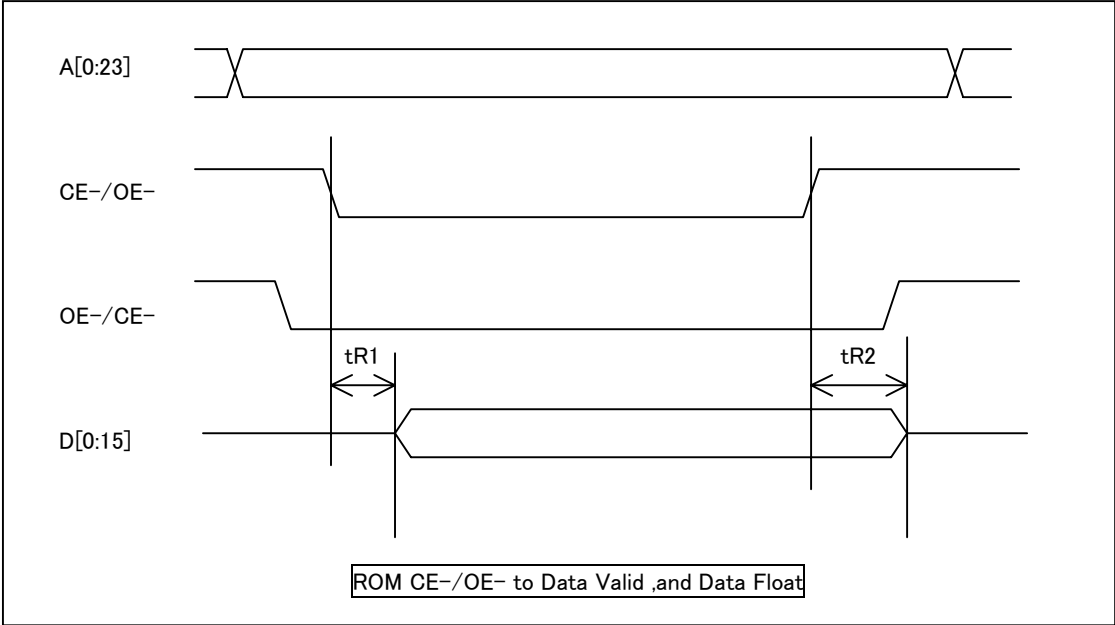








符号	项 目	PROBE POD(TYP)
tR1	CE-/OE- to Data Valid Delay	30nS
tR2	CE-/OE- to Data Float Delay	30nS
tR3	Address to Data Valid Delay	50nS



符号	项 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	

