

# **MJX330 for EJTAG2.5/2.6 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	03/04/23
1.01	● PDFファイル名変更 (MjxcdEjtg2.6J.pdf→Mj2CEj26J.pdf)	03/07/31
1.02	● レジスタコマンド修正 ● コンフィグレーション設定(MJX330設定画面) 変更 ● コマンド「変更～I/O ポート」を追加	03/08/20
1.03	● 誤字、脱字修正	03/09/03
1.04	● デバイス ドライバのインストール方法およびアンインストール方法を修正	03/10/28
1.10	● コンフィグレーションファイルの設定画面の変更	04/02/06
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、MJX330 for Ejtag2.5/2.6(以下、本文中は MJX330)をご購入いただきまして、誠にありがとうございます。

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、MJX330 の概略、各部の名称について記述しています。

## **第二章 デバイス ドライバのインストール**

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**

MJX330 とホストの接続方法、MJX330 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**

MJX330 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 MJX330 の環境設定**

MJX330 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**

MJX330 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 MJXDEBW コマンド**

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 MJX330 の概略 .....	9
1.3 各部の名称 .....	12
第二章 デバイス ドライバのインストール.....	14
2.1 デバイス ドライバのインストール手順 .....	14
2.1.1 Windows98/98SE の場合.....	14
2.1.2 Windows2000 の場合 .....	20
2.1.3 WindowsNT4.0 の場合 .....	26
2.1.4 WindowsXP の場合.....	31
2.2 デバイス ドライバのアンインストール手順.....	36
2.2.1 Windows98/98SE の場合.....	36
2.2.2 Windows2000 の場合 .....	38
2.2.3 WindowsNT4.0 の場合 .....	38
2.2.4 WindowsXP の場合.....	39
第三章 ハードウェアの接続.....	40
3.1 MJX330 とホストの接続 .....	40
3.2 MJX330 とターゲットの接続.....	41
第四章 ソフトウェアのインストール.....	42
第五章 MJX330 の環境設定 .....	49
第六章 ソフトウェアの起動と終了 .....	58
第七章 MJXDEBW コマンド .....	62
ABORT .....	66
BATCH .....	67
BP .....	68
BP/{C D E} .....	71
BP/{A R W} .....	72
BP/H .....	74
BP/S .....	75
CLEAR .....	76
CONFIG.....	77
DUMP .....	80
EXAMINE.....	81
FILL .....	82
FLASH .....	83

GO .....	85
HISTORY .....	86
INIT .....	87
JOURNAL .....	88
LOAD .....	89
MOVE .....	90
PIN .....	91
QUIT .....	92
REGISTER .....	93
REGISTER/T .....	95
SETLOAD .....	97
SETSAVE .....	98
STEP .....	99
TRACE .....	100
UNASM .....	101
VERSION .....	102
WAIT .....	103
第八章 MJXDEBW メニューコマンド .....	104
ファイル(F) ～ バッチ(B) .....	105
ファイル(F) ～ ジャーナル開始(S) .....	106
ファイル(F) ～ ジャーナル停止(E) .....	106
ファイル(F) ～ トレース結果(T) .....	107
エミュレーション(E) ～ 実行(G) .....	108
エミュレーション(E) ～ 再スタート(R) .....	108
エミュレーション(E) ～ 中断(B) .....	109
エミュレーション(E) ～ ステップ(S) .....	109
エミュレーション(E) ～ N-ステップ(N) .....	110
エミュレーション(E) ～ ダウンロード(L) .....	111
エミュレーション(E) ～ 初期化(I) .....	112
エミュレーション(E) ～ フラッシュメモリ(F) ～ 書込(W) .....	113
エミュレーション(E) ～ フラッシュメモリ(F) ～ 消去(E) .....	114
表示(V) ～ メモリ(M) .....	115
表示(V) ～ レジスタ(R) .....	117
表示(V) ～ 逆アセンブル(D) .....	118
表示(V) ～ トレース結果(T) .....	119
表示(V) ～ コマンド応答クリア(C) .....	120
表示(V) ～ オプション(O) ～ フォント(F) .....	121

表示(V) ～ オプション(O) ～ 表示(V).....	122
変更(M) ～ メモリ(M).....	123
変更(M) ～ I/O ポート(P).....	124
変更(M) ～ レジスタ(R).....	125
変更(M) ～ アセンブラ(A).....	126
変更(M) ～ フィル(F).....	127
変更(M) ～ 移動(M).....	128
設定(S) ～ ブレークポイント(B).....	129
設定(S) ～ トレース(T).....	131
設定(S) ～ 設定の読出(L).....	132
設定(S) ～ 設定の保存(S).....	133
設定(S) ～ コンフィグレーション(C) ～ 設定(S).....	134
設定(S) ～ コンフィグレーション ～ 表示.....	136
設定(S) ～ コンフィグレーション ～ 上書き保存(E).....	137
設定(S) ～ コンフィグレーション ～ 名前を付けて保存(A).....	138
ヘルプ(H) ～ トピックの検索(H).....	139
ヘルプ(H) ～ MJXDEBW のバージョン情報(A).....	140
第九章 高速ダウンロード.....	141
付録 A 仕様.....	142
付録 B ターゲット システムの制限事項.....	143
付録 C EJTAG コネクタ.....	144
付録 D レジスタ名一覧.....	146
付録 E MJX バイナリ ファイル.....	149
付録 F ターゲット システムへのプロービング.....	150
付録 G 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.....	152
付録 H 対応フラッシュ メモリー一覧.....	156
付録 I デバイス ドライバのトラブルシューティング.....	170
検索項目.....	176

# 第一章 概要

製品構成、MJX330 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 MJX330 for EJTAG2.5/2.6 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。

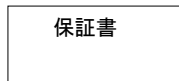
- ☐ MJX330 for EJTAG2.5/2.6 本体



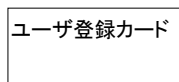
- ☐ プローブケーブル



- ☐ 保証書

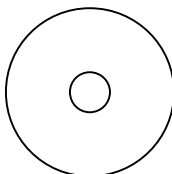


- ☐ ユーザー登録カード



**【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





## 1.2 MJX330 の概略

MJX330 for EJTAG2.5/2.6 は、EJTAG2.5/2.6 仕様の JTAG コネクタを備えた MIPS 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。

### MJX330 のおもな特長

- ターゲット システムの[EJTAG コネクタ](#)を使用するため、接続が簡単です。
- PC カード Type II で実現されているため、フィールドでの使用に適しています。
- 高速な CPU でも安定して動作します。
- 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。MJX330 が持つ機能をフルに使用できます。
- ハードウェアブレークポイント機能があります。
- ダウンロード速度 700K バイト/秒 \*1 (JTAG クロック:20MHz,MIPS32,Ejtag2.6)

### 構成

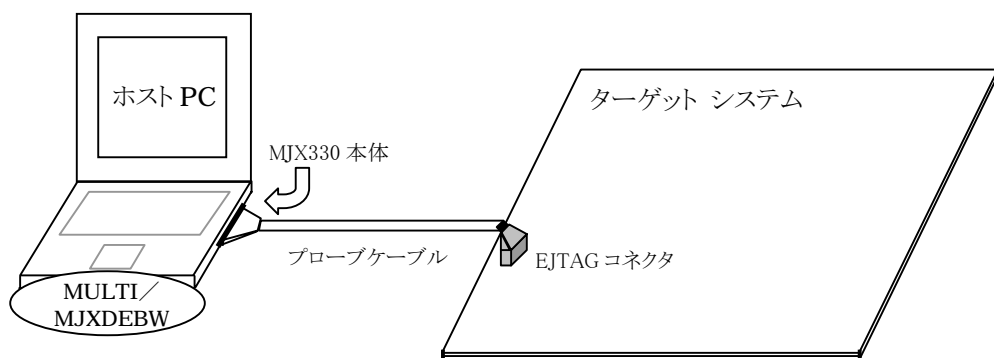


図 1-1 構成

\*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。

## 第一章 概要

MJX330 は図 1-1 の構成で動作します。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。プログラムはプローブケーブルを通して、ターゲット システム上の RAM へダウンロードされます。

MJX330 は、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](#)を設定することはできません。

### **MJX330 を使用する前の準備作業**

MJX330 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。「ハードウェアの接続」以外は、購入時に一度だけ行なえばよい作業です。

- [デバイスドライバのインストール](#)
- [ハードウェアの接続](#)
- [ソフトウェアのインストール](#)
- [MJX330 の環境設定](#)

準備作業が終了したら、[第六章](#)を参照して、MJX330 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、MJX330 独自の機能を使用するための MJXDEBW 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

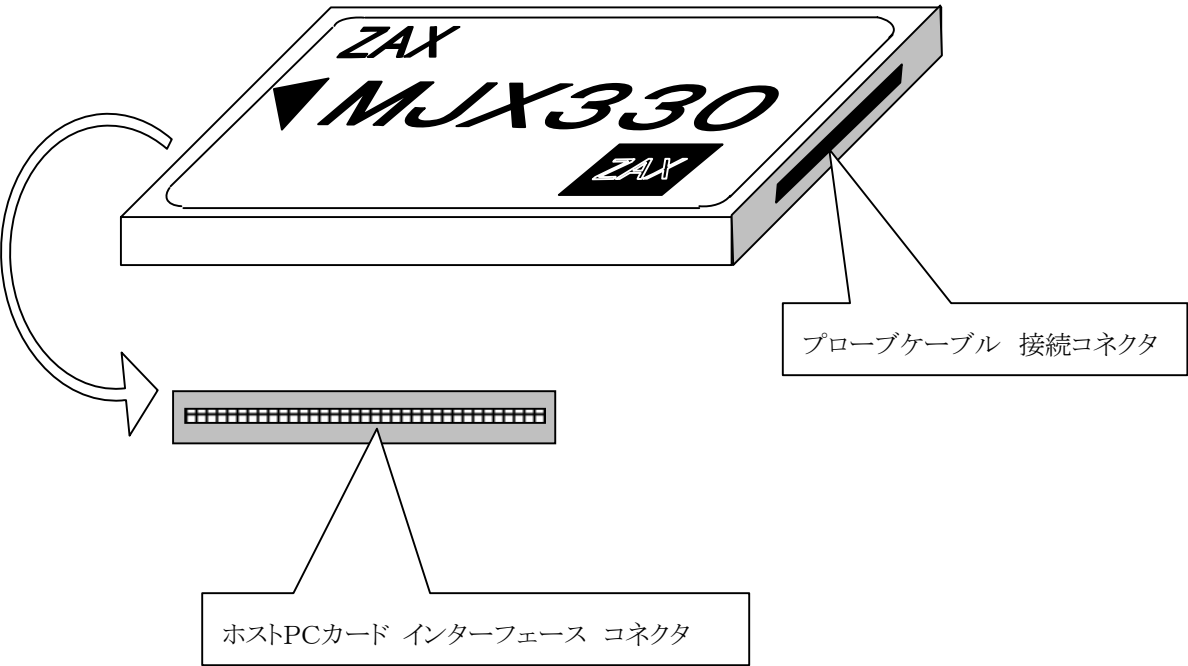
MULTI はいろいろな環境で実行することができる高級言語デバッガです。サーバープログラム MJXSERV を呼び出すことによって、MJX の環境で実行させることができますようになります。

### MJXDEBW について

MJXDEBW は MJXDEBW 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

MJX330 本体



ホストPCカード インターフェース コネクタ	ホストPCの PC カードスロットに挿入します。
プローブケーブル 接続コネクタ	プローブケーブルを接続します。

プローブケーブル



プローブケーブル

MJX330 と EJTAG を接続するケーブル

EJTAG コネクタ

ターゲット システムの EJTAG コネクタへ接続するコネクタ

ホストPC側コネクタ

MJX330 へ接続するコネクタ

## 第二章 デバイス ドライバのインストール<sup>\*1</sup>

【重要】 MJX330 はスタンバイには対応していません。ノートPCの消費電力設定をオフにしてご使用ください。

### 2.1 デバイス ドライバのインストール手順

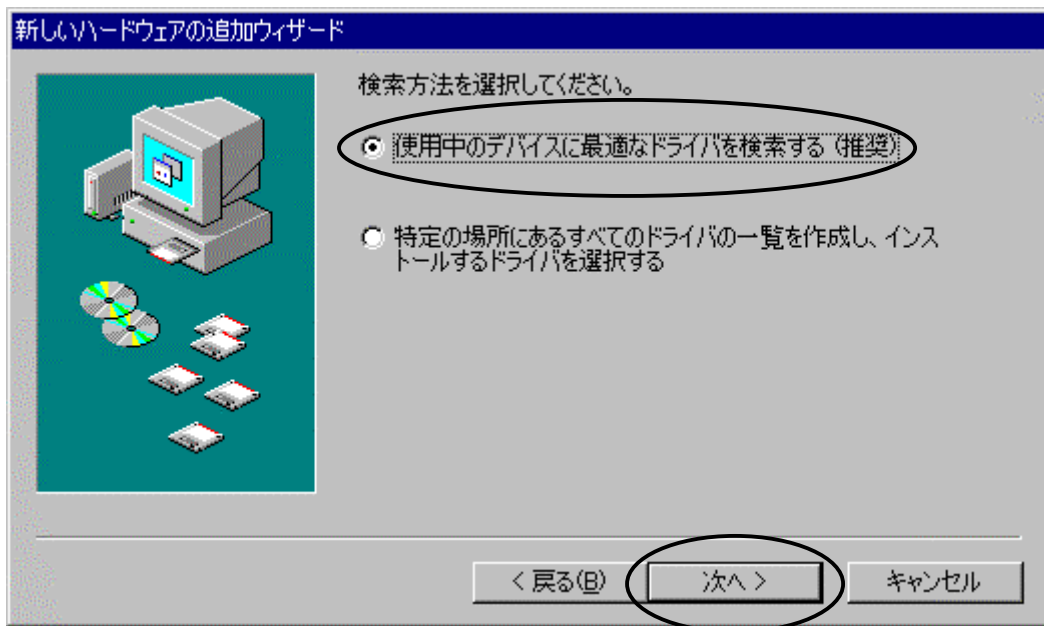
#### 2.1.1 Windows98/98SE の場合

ノートPCの電源オンの状態でPCカードを差すと、新しいハードウェアの追加ウィザードによりカードが検出され以下のダイアログが表示されますので、「次へ>」をクリックします。



<sup>\*1</sup>すでに PCMCIA I/F カードにより MJX440 をご使用のお客様はこの操作は不要です。

「使用中のデバイスに最適なドライバを検出する(推奨)」を選択し、「次へ>」をクリックします。



CD-ROMドライブに「MJX for MIPS(Ejtag2.5/2.6) Tools Disk」の CD-ROM をセットします。

「検索場所の指定(L)」を選択し、ドライバのあるディレクトリ D:¥DRIVER¥WIN9X (CD-ROM ドライブが D ドライブの場合)を指定します。

「次へ>」をクリックします。



ドライバ ファイルのあるディレクトリが正しい場合は、以下のダイアログが表示されますので、

「次へ」をクリックします。

「このデバイス用のドライバが見つかりませんでした。」と表示された場合は、「戻る(B)」をクリックし、ドライバ ファイルのある正しいディレクトリを指定します。





自動的にドライバがインストールされ、以下のダイアログが表示されます。

正しくインストールされた場合は「**ピポッ**」という音がしますので「**完了**」をクリックします。

「**ブー**」という音がした場合は正しくインストールされていないので、インストールを終了させてから、「付録I デバイス ドライバのトラブルシューティング」を参考にしてトラブルシュートを行なってください。



インストールが正しく行われたことを確認するために、コントロール パネルから「システム」をダブル クリックして開き、「デバイス マネージャ」タブをクリックします。

次に「ZDIF」クラスをクリックして、その下にある「ZDIF Interface Card [PCMCIA/PC Card]」のプロパティを表示します(ダブル クリックするか、「プロパティ(R)」をクリック)。



「リソース」タブをクリックし、「競合するデバイス」が「競合はありません。」になっていることを確認します。



## 第二章 デバイス ドライバのインストール

また、「I/O の範囲」が以下のいずれかになっていることを確認します。設定値は、動作環境によって異なります。

0220 – 023F

0260 – 027F

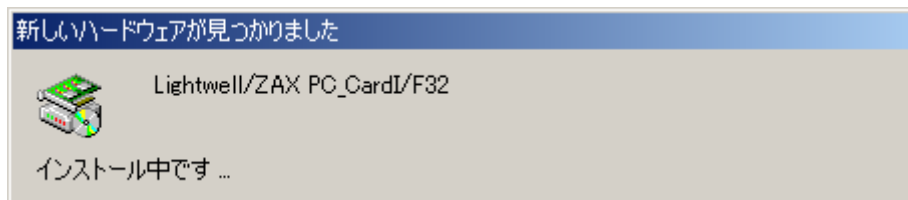
02E0 – 02FF

0320 – 033F

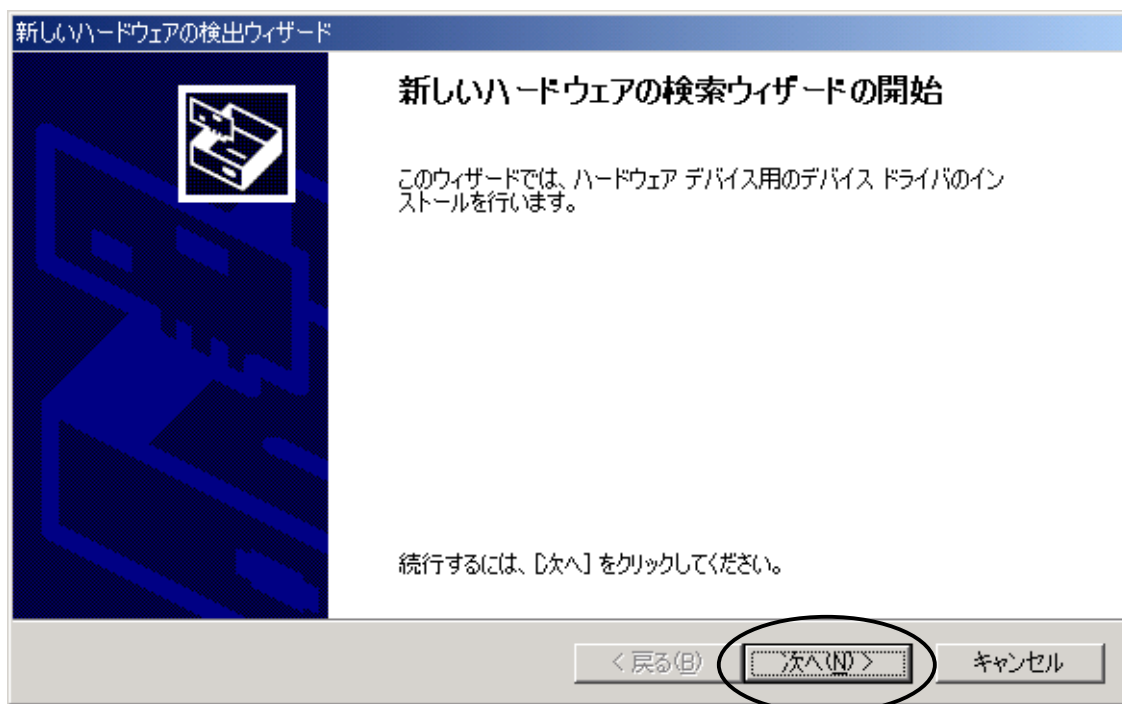
03E0 – 03FF

### 2.1.2 Windows2000 の場合

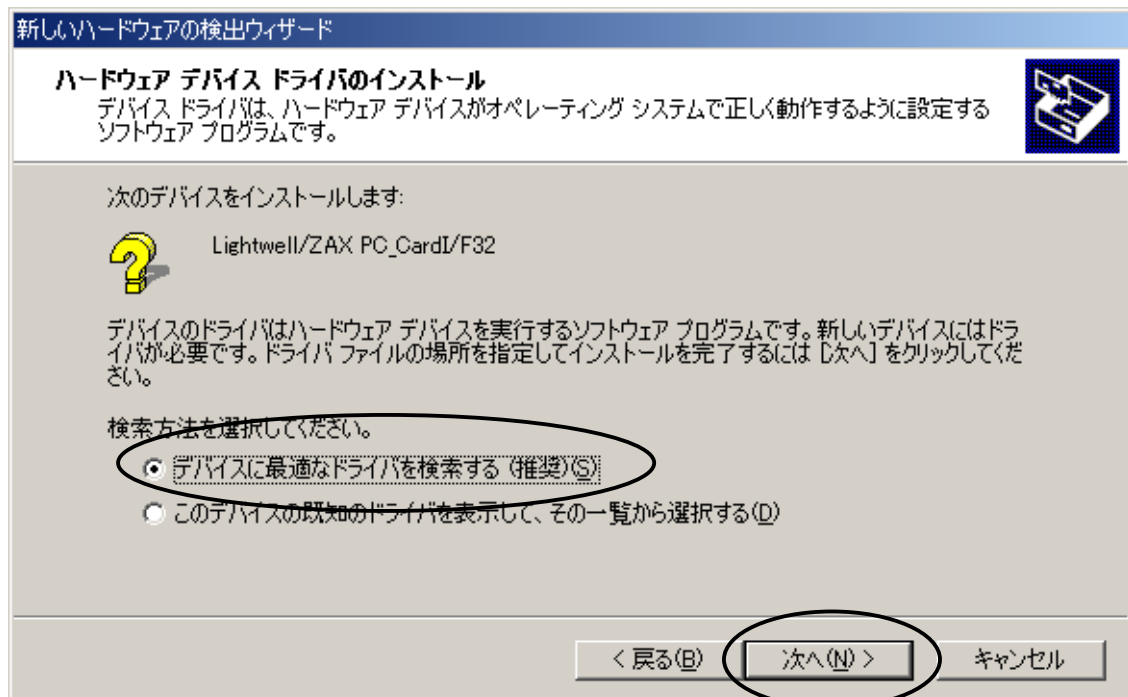
ノート PC の電源オンの状態で PC カードを差すと、ハードウェア ウィザードが起動します。



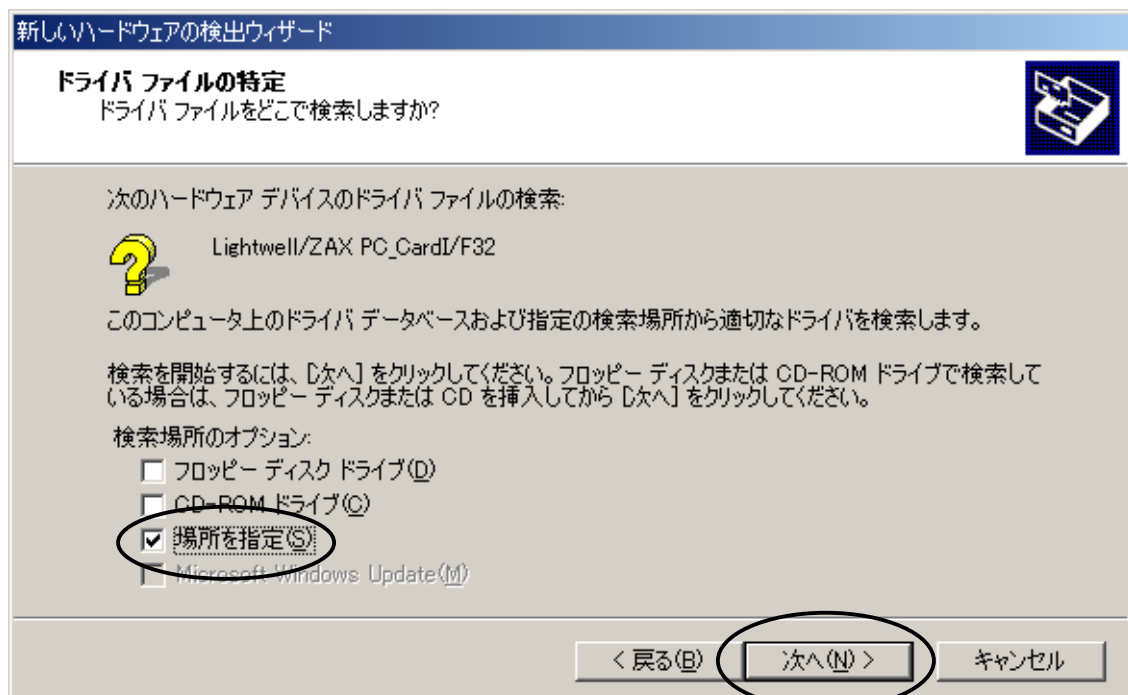
次のウィンドウが表示されたら、「次へ」をクリックしてください。



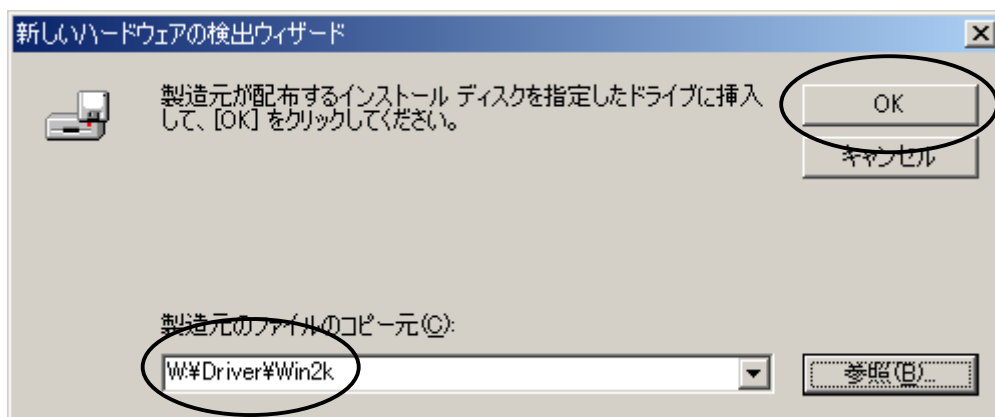
次のウィンドウが表示されたら、「デバイスに最適なドライバを検索する」を選択し、「次へ」をクリックしてください。



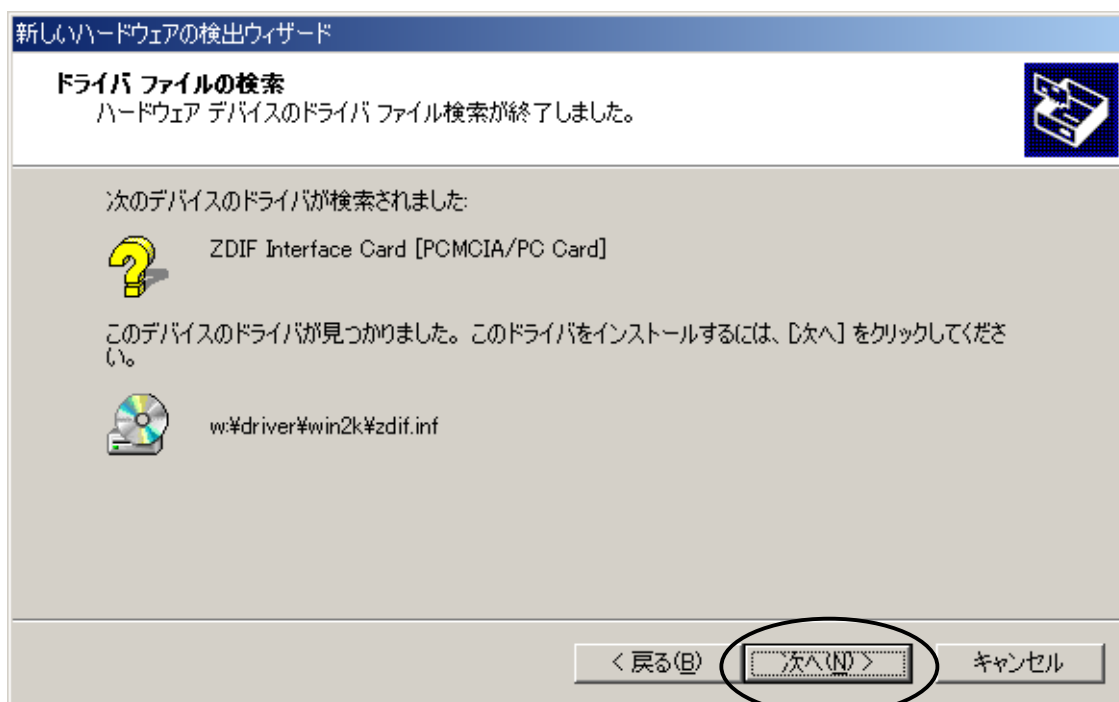
次のウィンドウが表示されたら、「場所を指定」を選択し、「次へ」をクリックしてください。



次のウィンドウが表示されたら、「MjX for MIPS(Ejtag2.5/2.6) Tools Disk」CD-ROM をドライブにセットしてください。次に、「検索場所の指定」を選択し、W:¥Driver¥Win2k (CD-ROMドライブが W の場合)と入力し、「OK」をクリックしてください。



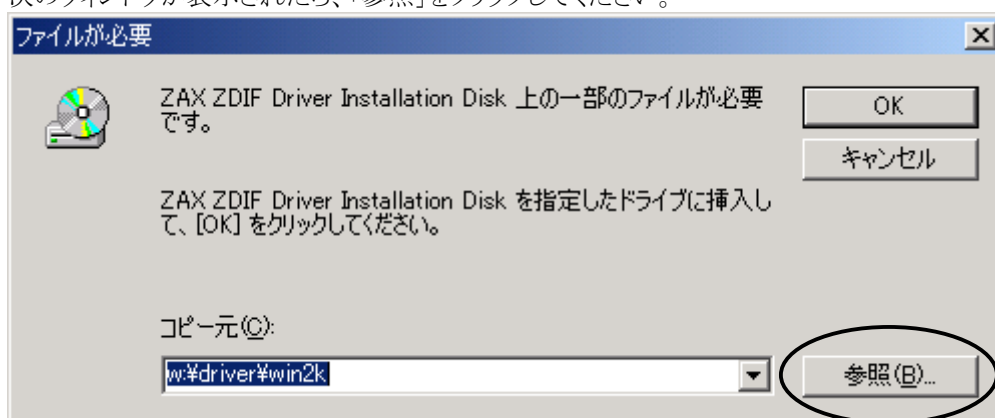
次のウィンドウが表示されたら、「OK」をクリックしてください。



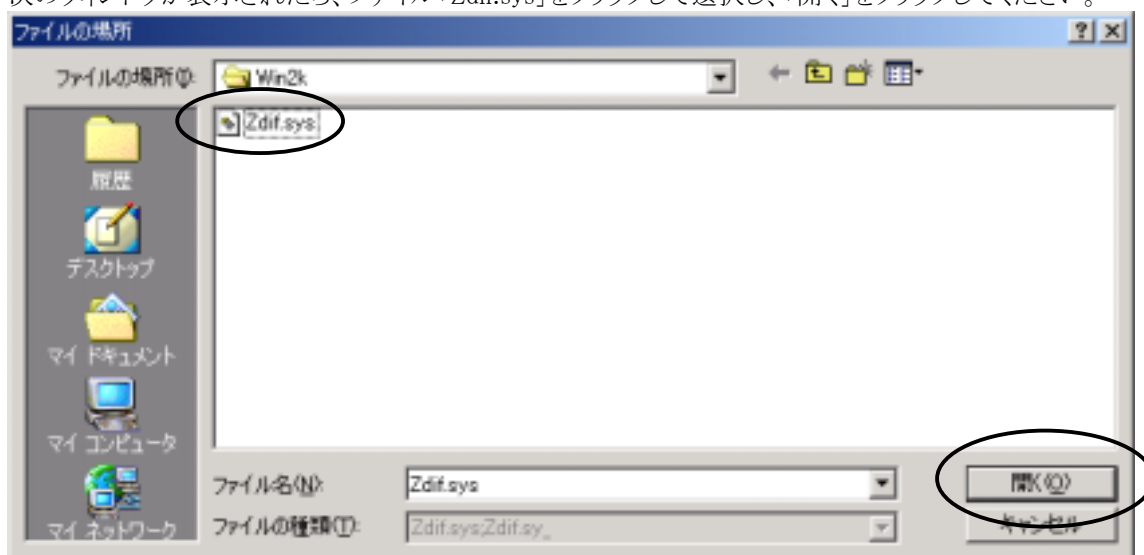
次のダイアログボックスが表示されたら、「OK」をクリックしてください。



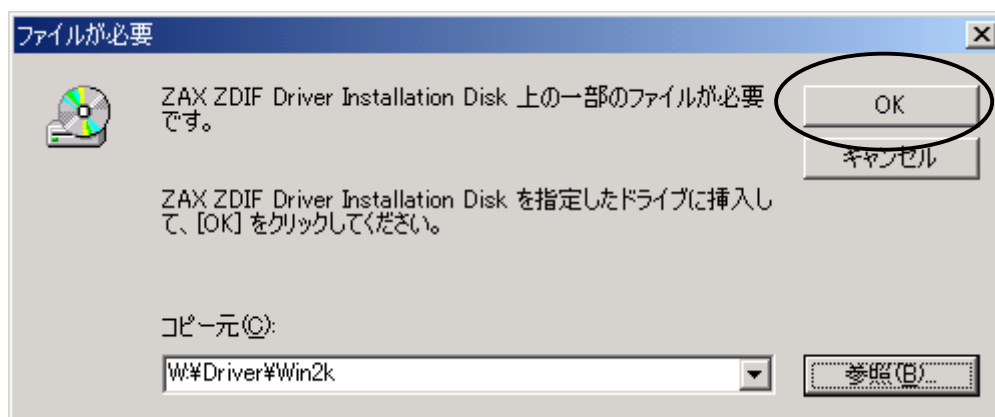
次のウィンドウが表示されたら、「参照」をクリックしてください。



次のウィンドウが表示されたら、ファイル「Zdif.sys」をクリックして選択し、「開く」をクリックしてください。



次のウィンドウが表示されたら、「OK」をクリックしてください。



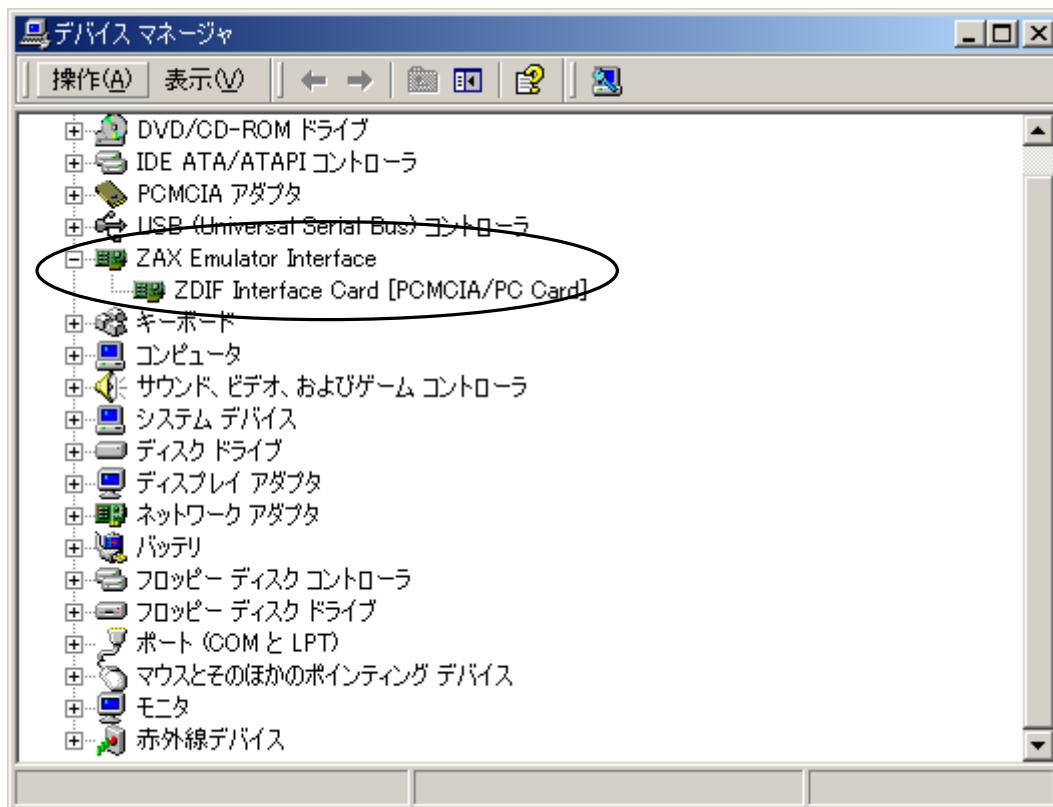
インストール完了を示す次のウィンドウが表示されたら、「完了」をクリックしてください。





ドライバのインストールが終了したら、次の手順でインストールが正常に終了したか確認することができます。

スタート メニュー → 設定 → コントロール パネル → システム → ハードウェア  
デバイス マネージャを選択  
「ZAX Emulator Interface」左の「+」をクリック



「ZDIF Interface Card [PCMCIA/PC Card]」が表示されれば正常です。

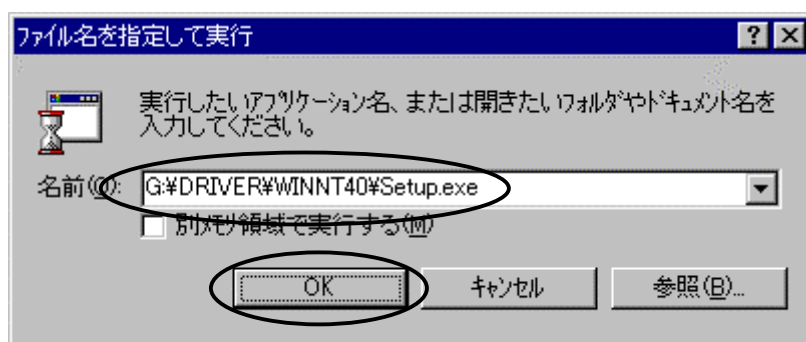
### 2.1.3 WindowsNT4.0 の場合

ノート PC の電源がオフしている状態で PC カードを差した後、電源をオンします。

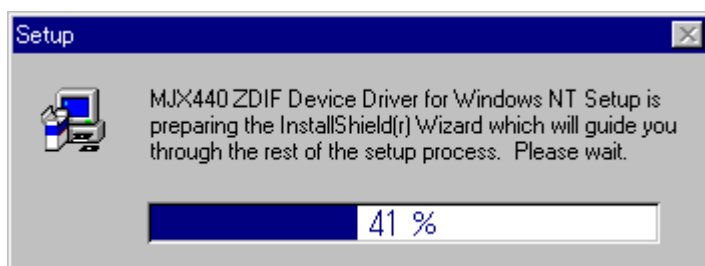
Windows NT4.0 が起動したら、CD-ROM ドライブに「*MJX for MIPS(Ejtag2.5/2.6) Tools Disk*」の CD-ROM をセットします。

「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。

「名前(O)」に **G:¥DRIVER¥WINNT4¥Setup.exe** (CD-ROM ドライブが G ドライブの場合)を指定し、「OK」をクリックします。

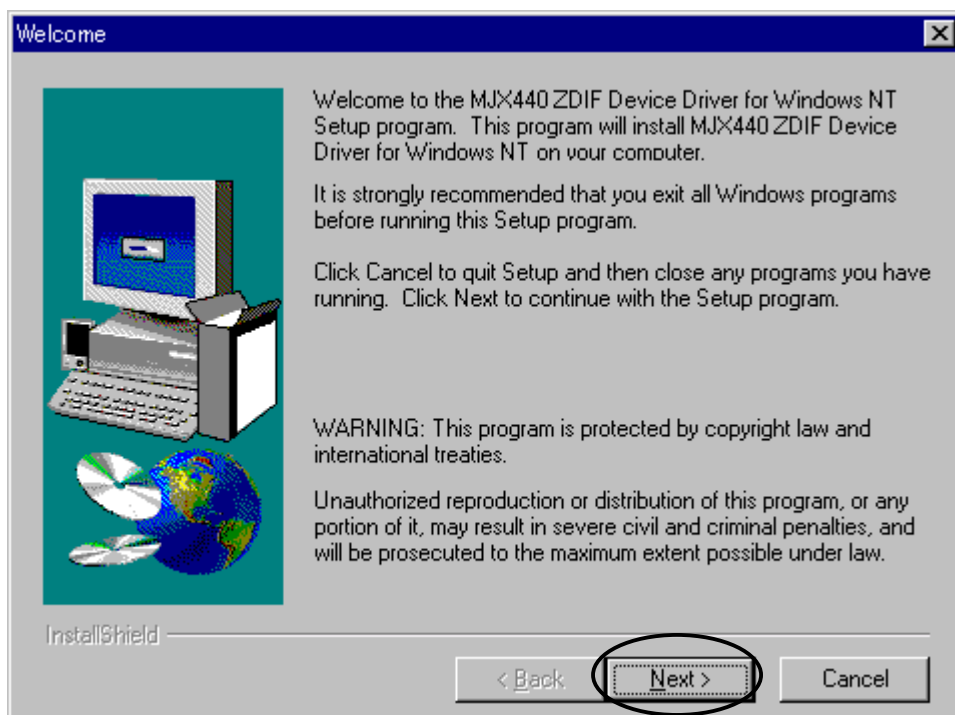


セットアップ プログラムが起動します。



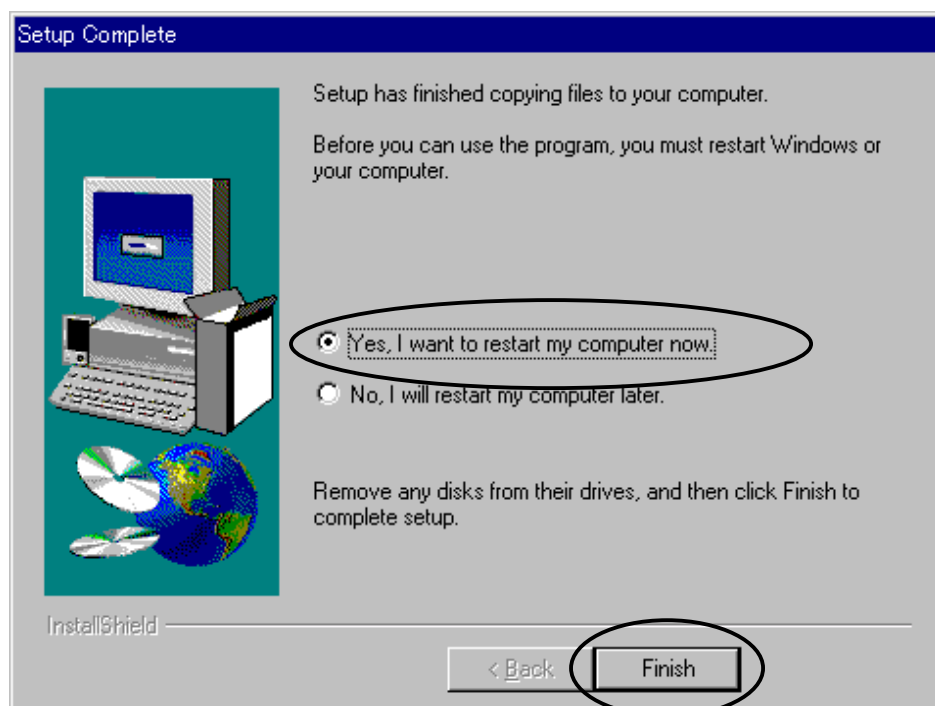
「Next >」をクリックします。

自動的にドライバがインストールされます。



ドライバのインストール後に以下のダイアログが表示されます。

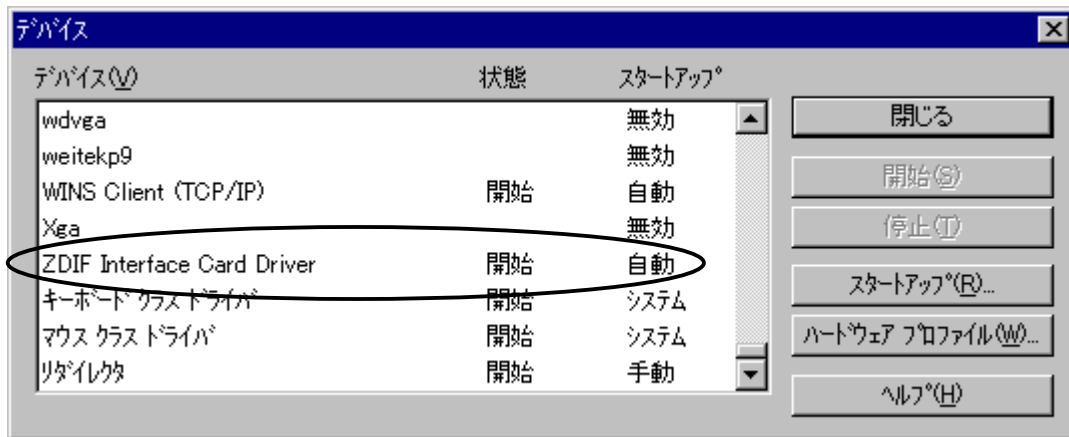
「Yes, I want to restart my computer now」を選択し、「Finish」をクリックします。



Windows NT4.0 が再起動します。

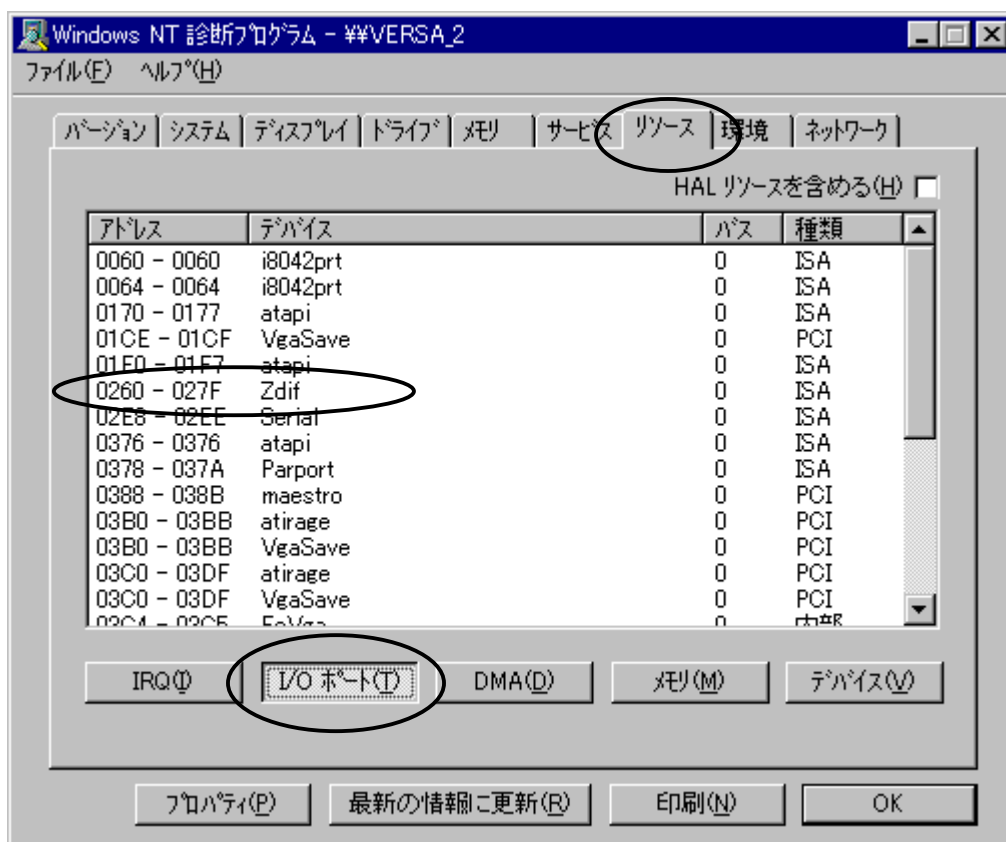
インストールが正しく行われたことを確認するために、コントロール パネルから「デバイス」をダブル クリックして開き、「ZDIF Interface Card Driver」を探します。

状態が「開始」に、スタートアップが「自動」になっていることを確認します。



続いて、「スタート」、「プログラム(P)」、「管理ツール (共通)」から「Windows NT 診断プログラム」を起動します。

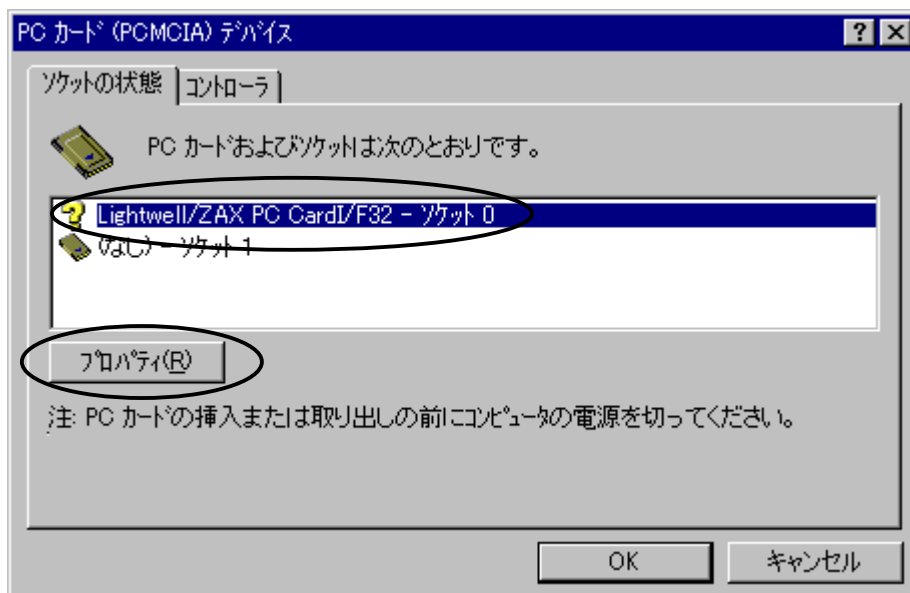
「リソース」タブをクリックし、「I/O ポート(T)」をクリックします。



「デバイス」で Zdif を探し、PC カードに割り当てられた I/O ポートが「アドレス」に表示されていることを確認します。アドレスの設定値は、動作環境によって異なります。

コントロール パネルで「PC カード(PCMCIA)」をダブル クリックして開きます。

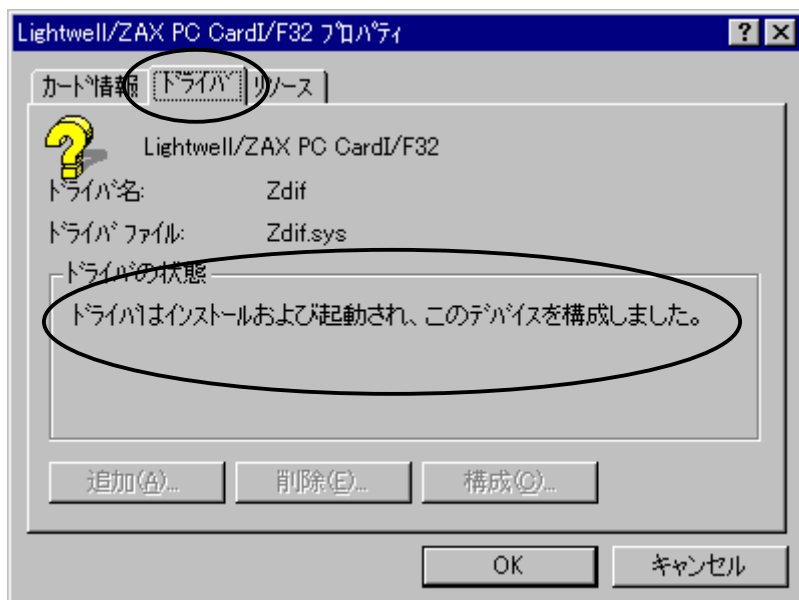
「Lightwell/ZAX PC Card I/F32」をクリックして選択し、「プロパティ(R)」ボタンをクリックします。



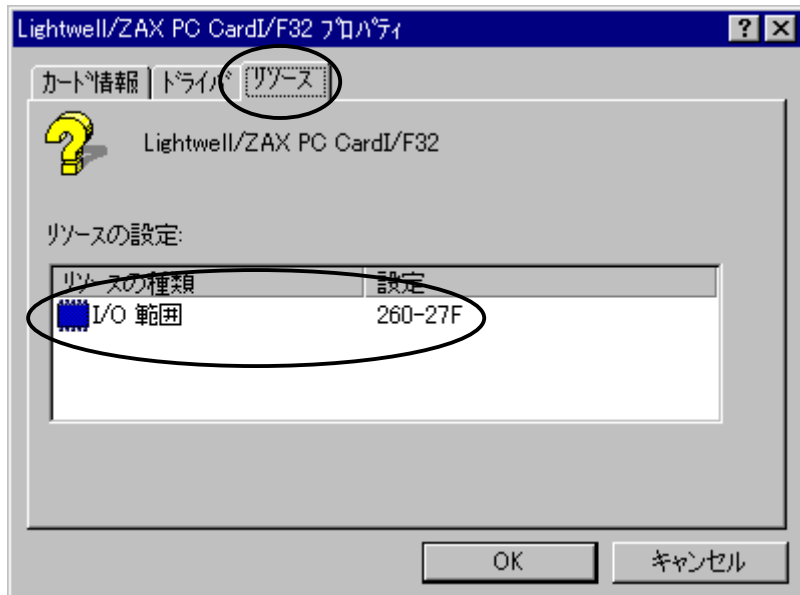
「カード情報」タブをクリックし、「デバイスの状態」が「デバイスは正常に動作しています。」になっていることを確認します。



「ドライバ」タブをクリックし、「ドライバの状態」が「ドライバはインストールおよび起動され、このデバイスを構成しました。」になっていることを確認します。



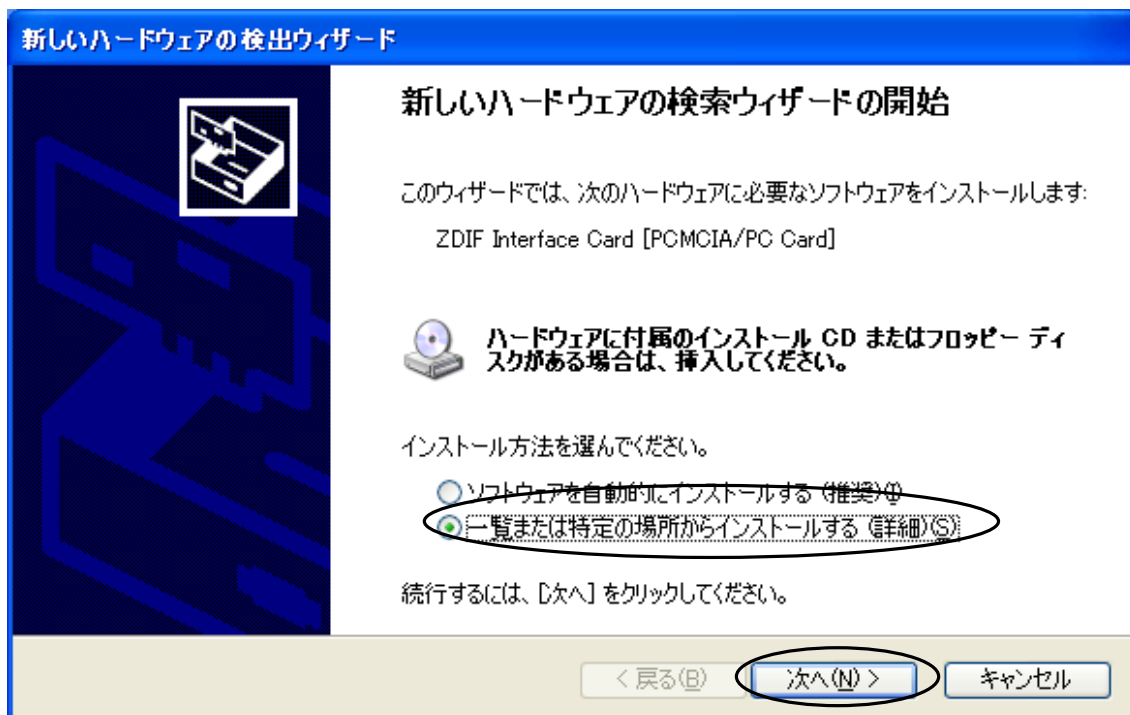
「リソース」タブをクリックし、「リソースの設定」、「I/O 範囲」が以下のいずれかになっていることを確認します。



0220 – 023F  
 0260 – 027F  
 02E0 – 02FF  
 0320 – 033F  
 03E0 – 03FF

## 2.1.4 WindowsXP の場合

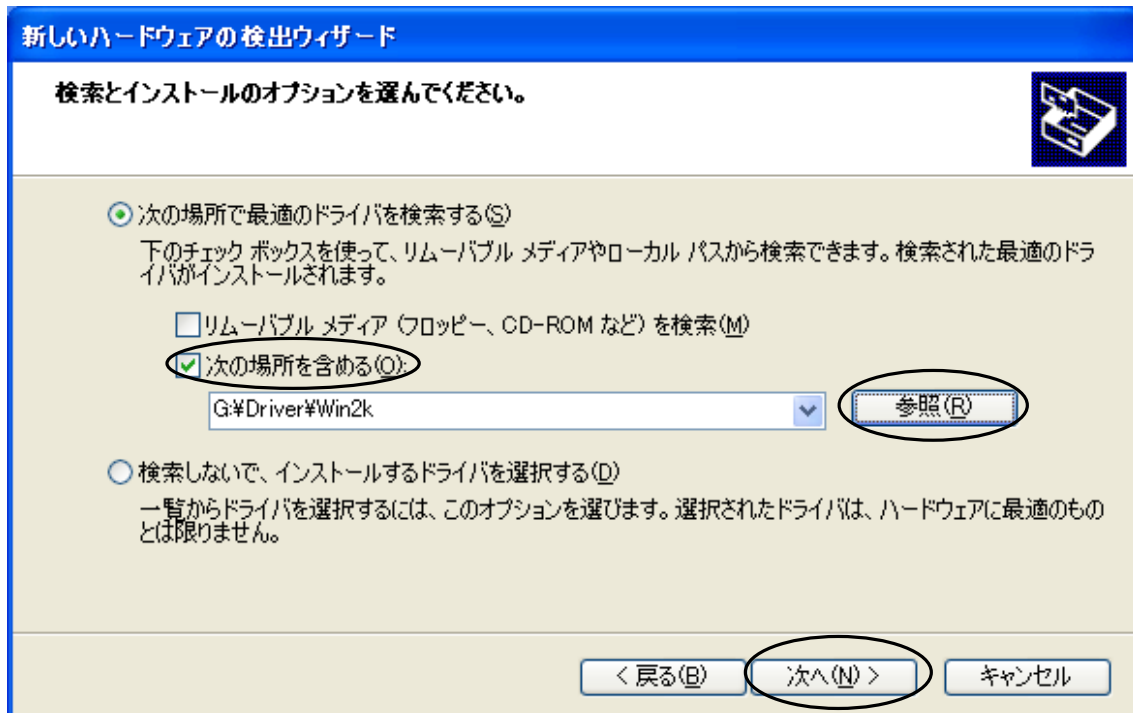
ノートPC の電源オンの状態で PC カードを差すと、ハードウェア ウィザードが起動します。  
次のウィンドウが表示されたら、「一覧または特定の場所からインストールする」を選んで、  
「次へ」をクリックしてください。



次のウィンドウが表示されたら、「MJX for MIPS(Ejtag2.5/2.6) Tools Disk」CD-ROM をドライブにセットしてください。

次に、「次の場所を含める」を選択し、参照ボタンをクリックして

G:\Driver\Win2k(CD-ROMドライブが G の場合)を選択してください。選択できたら「次へ」をクリックしてください。



次のダイアログボックスが表示されたら、「OK」をクリックしてください。

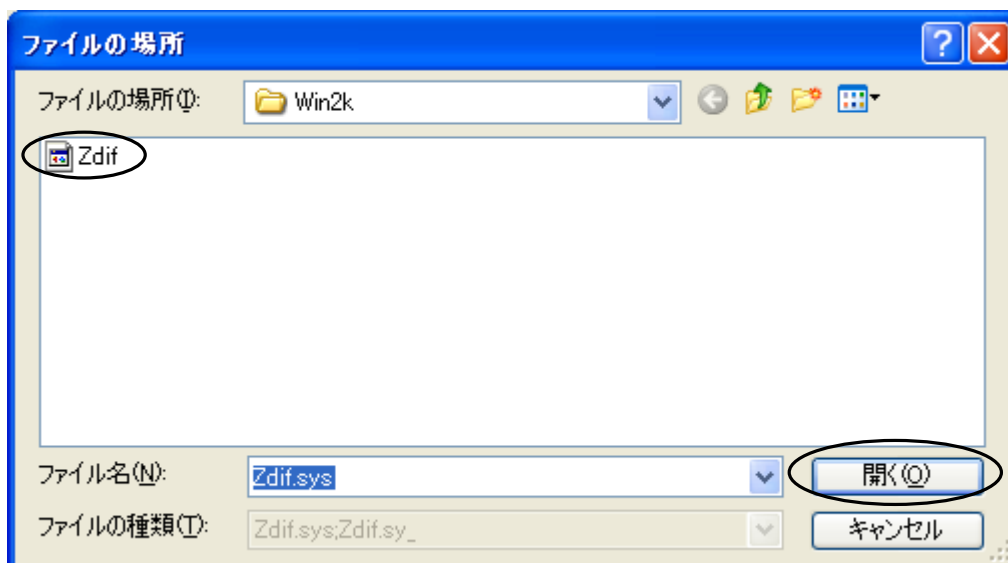




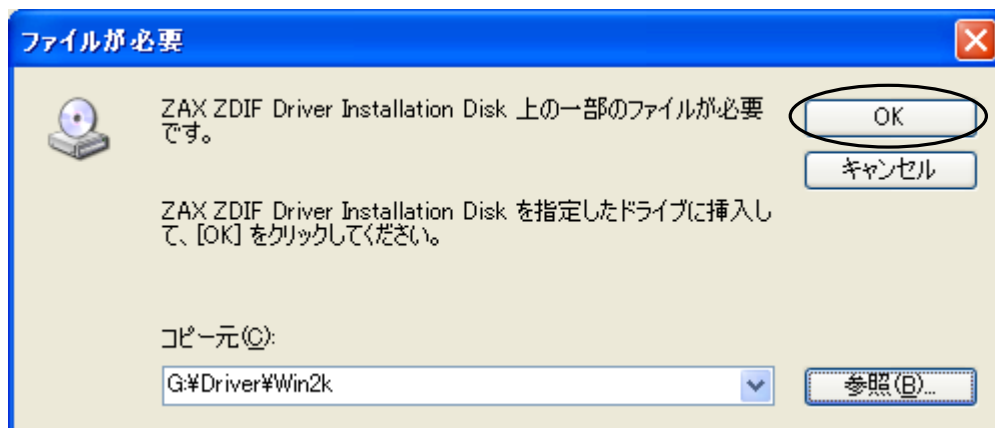
次のウィンドウが表示されたら、「参照」をクリックしてください。



次のウィンドウが表示されたら、ファイル「Zdif.sys」をクリックして選択し、「開く」をクリックしてください。



次のウィンドウが表示されたら、「OK」をクリックしてください。



インストール完了を示す次のウィンドウが表示されたら、「完了」をクリックしてください。

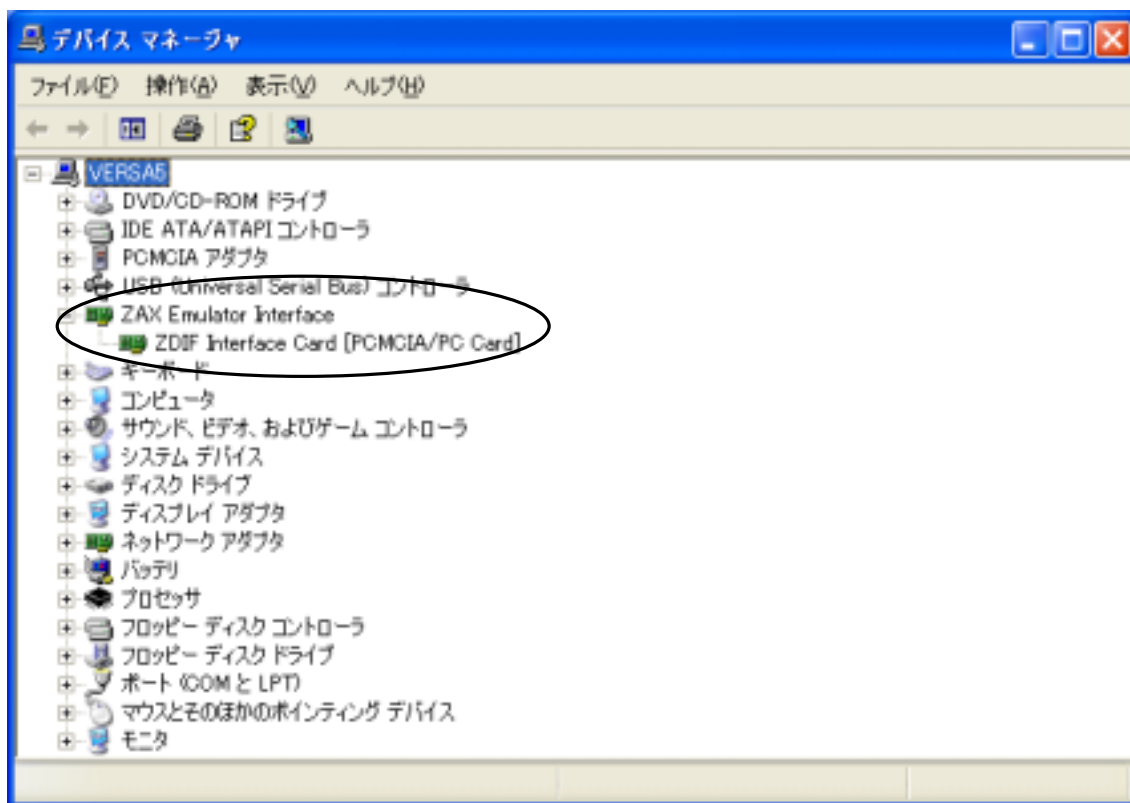


ドライバのインストールが終了したら、次の手順でインストールが正常に終了したか確認することができます。

スタートメニュー → マイ コンピュータ → システムタスク(システム情報を表示する)を選択して、システムプロパティの「ハードウェア」タブをクリックします。

デバイスマネージャを選択して、

「ZAX Emulator Interface」左の「+」をクリックします。



「ZDIF Interface Card [PCMCIA/PC Card]」が表示されれば正常です。

## 2.2 デバイス ドライバのアンインストール手順

### 2.2.1 Windows98/98SE の場合

- ①PC カードを差した状態で、コントロール パネルから「システム」をダブル クリックして開き、「デバイス マネージャ」タブをクリックします。
- ②次に「ZDIF」クラスをクリックして、その下にある「ZDIF Interface Card [PCMCIA/PC Card]」をクリックして選択し、「削除(E)」ボタンをクリックします。
- ③「デバイス削除の確認」ダイアログが表示されるので、「OK」ボタンをクリックします。

◆PC カードのデバイス ドライバの情報を完全に削除したい場合は、INF ファイルも削除します。

- ①エクスプローラを起動し「**表示(V)**」メニューから「**フォルダ オプション(O)...**」を選択し、「フォルダ オプション」ダイアログで「**表示**」タブをクリックします。
- ②「**登録されているファイルの拡張子は表示しない**」のチェックを外し(オフ)、「**全てのファイルを表示する**」をチェック有り(オン)にして、「**OK**」をクリックします。

これで INF ファイルが表示されるようになります。

- ③エクスプローラでシステム ディレクトリ(デフォルトは「C:\Windows」)の下の「**Inf\Other**」ディレクトリを開き、「**Zaxzdif.inf**」ファイルを削除します。

### 2.2.2 Windows2000 の場合

- ①PC カードを差した状態で、**コントロール パネル**から「**システム**」をダブル クリックして開き、「**ハードウェア**」タブから「**デバイス マネージャ**」ボタンをクリックします。
- ②「**ZAX Emulator Interface**」クラスをクリックして、その下にある「**ZDIF Interface Card [PCMCIA]**」をダブルクリックして表示したプロパティウインドウの「**ドライバ**」タブを選択します。そして「**削除(E)**」ボタンをクリックします。
- ③「デバイス削除の確認」ダイアログが表示されるので、「**OK**」ボタンをクリックします。

### 2.2.3 WindowsNT4.0 の場合

- ①PC カードを差した状態で、**コントロール パネル**から「**デバイス**」をダブル クリックして開き、「**ZDIF Interface Card Driver**」をクリックして選択します。
- ②続いて「**停止(T)**」ボタンをクリックします。
- ③「ZDIF Interface Card Driver デバイスを停止してもよろしいですか」というダイアログが表示されるので、「**はい(Y)**」をクリックします。

デバイスが停止し、ドライバがアンロードされます。

- ④**コントロール パネル**から「**アプリケーションの追加と削除**」をダブル クリックして開き、「**インストールと削除**」タブをクリックします。
- ⑤「**MJX440 ZDIF Device Driver for Windows NT**」をクリックして選択し、「**追加と削除(R)**」ボタンをクリックします。

- ⑥「**Confirm File Deletion**」ダイアログが表示されるので、「**はい(Y)**」をクリックします。

自動的にドライバがアンインストールされます。

- ⑦「**OK**」をクリックし、終了します。

- ⑧Windows NT4.0 をシャットダウンさせノート PC の電源をオフした後、PC カードを抜きます。

### 2.2.4 WindowsXP の場合

- ①PC カードを差した状態で、スタート → マイ コンピュータを選択して、システムタスクの「システム情報を表示する」をクリックして、システムプロパティウィンドウを表示します。
- ②システムプロパティウィンドウで「ハードウェア」タブを選択し、「デバイスマネージャ」をクリックしてデバイスマネージャウィンドウを表示します。
- ③「**ZAX Emulator Interface**」 クラスをクリックして、その下にある「**ZDIF Interface Card [PCMCIA]**」を選択して右クリックします。そして、「削除」を選択します。
- ④「デバイス削除の確認」ダイアログボックスが表示されるので、「**OK**」ボタンをクリックします。

## 第三章 ハードウェアの接続

MJX330 とホストの接続方法、MJX330 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。

### 3.1 MJX330 とホストの接続

#### MJX330 とホストの接続

プローブケーブルを MJX330 に接続し、MJX330 をホストの PC カードスロットに取り付けてください。

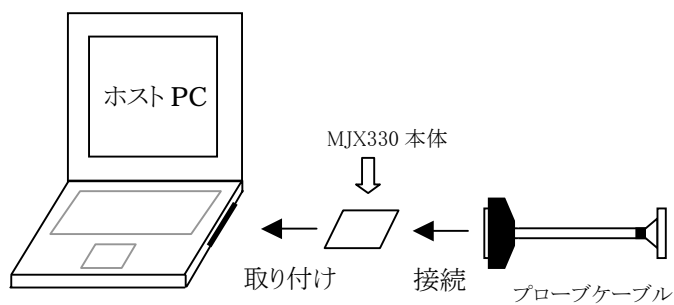


図 3-1 MJX330 とホストの接続

【注意】 プローブ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。

- ◆ PC 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。
- ◆ カード 2 枚を無理に差し込むと、PC カード スロット、および MJX330 カードのコネクタ部が壊れる場合があります。
- ◆ PC 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 MJX330 とターゲットの接続

### MJX330 とターゲットの接続

図 3-1 のとおり、MJX330 とプローブケーブルを接続した後、図 3-2 のように、ターゲット システムを接続してください。

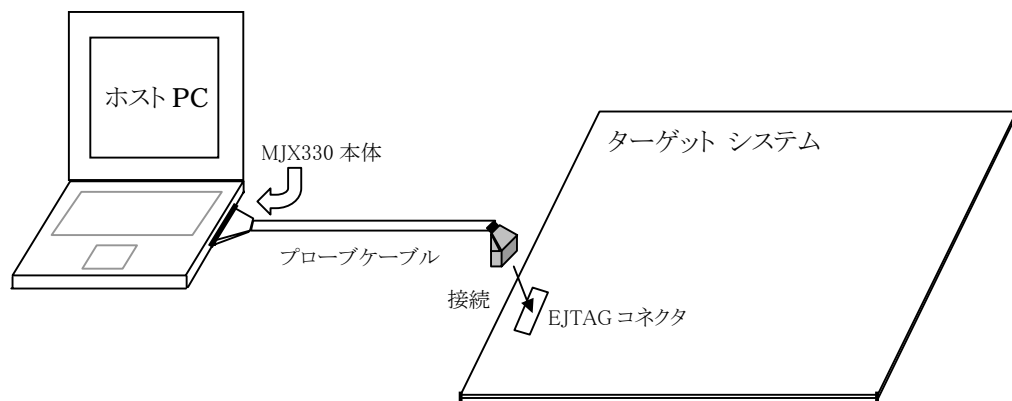


図 3-2 MJX330 とターゲットの接続

**【注意】** ターゲットシステムへプローブケーブルを接続するときは、必ずターゲットシステムの電源を切ってから行ってください。

**【注意】** プローブケーブルのコネクタには向きがあります。コネクタの1番ピン同士が合うように接続してください。プローブケーブルのコネクタには誤挿入防止ピンが挿入されています。稀にターゲットによって誤挿入防止ピンがあることによって接続できない場合があります。この場合は、誤挿入防止ピンを取り外してください。逆差しが可能になりますので充分注意して接続を行ってください。

## 第四章 ソフトウェアのインストール

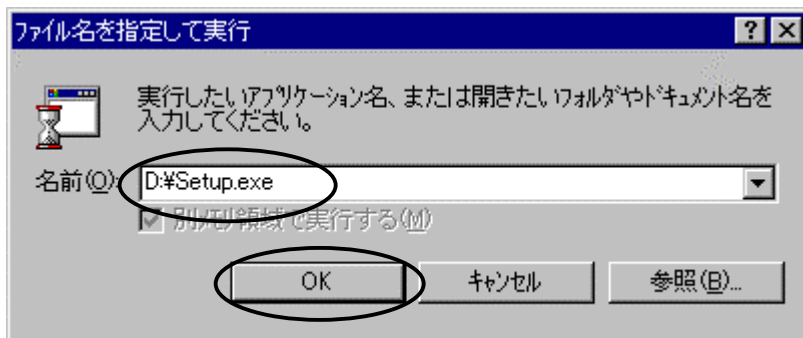
MJX330 を操作するソフトウェアのインストール方法について記述しています。

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。

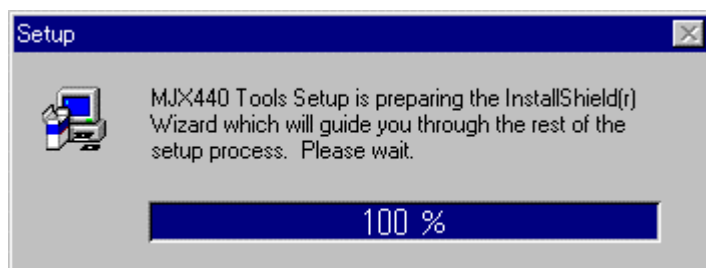
1. Green Hills Software 社の統合開発環境 MULTI をインストールします。
2. CD-ROMドライブに「*MJX for MIPS(Ejtag2.5/2.6) Tools Disk*」の CD-ROM をセットします。

「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。

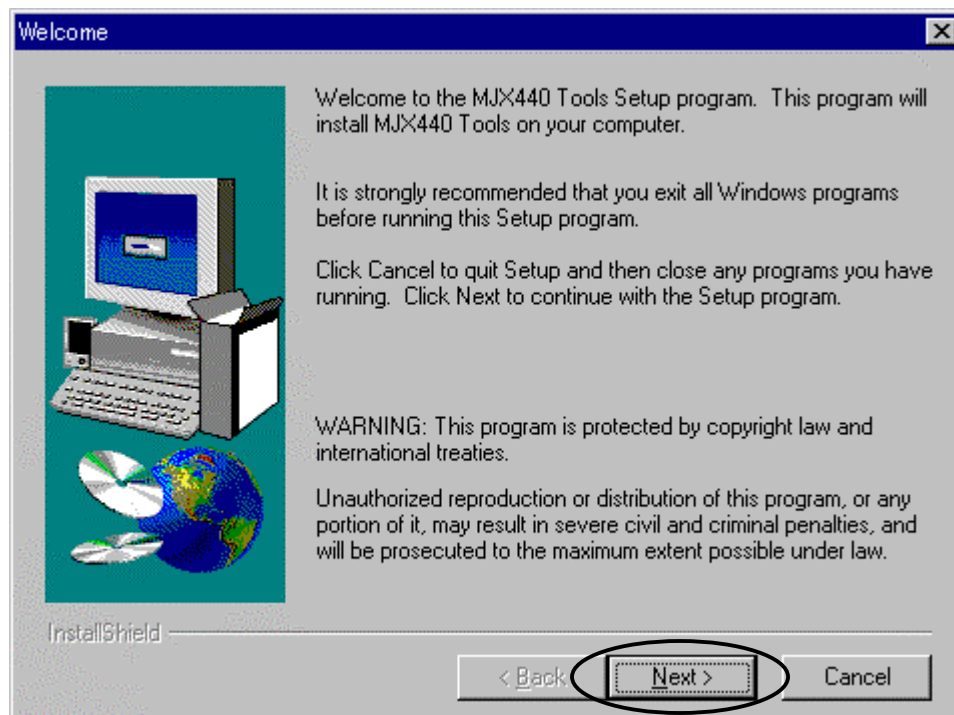
3. 「名前(O)」にセットアップ プログラム Setup.exe を指定し、「OK」をクリックします。(CD-ROMドライブが Dドライブの場合、「D:¥Setup.exe」)



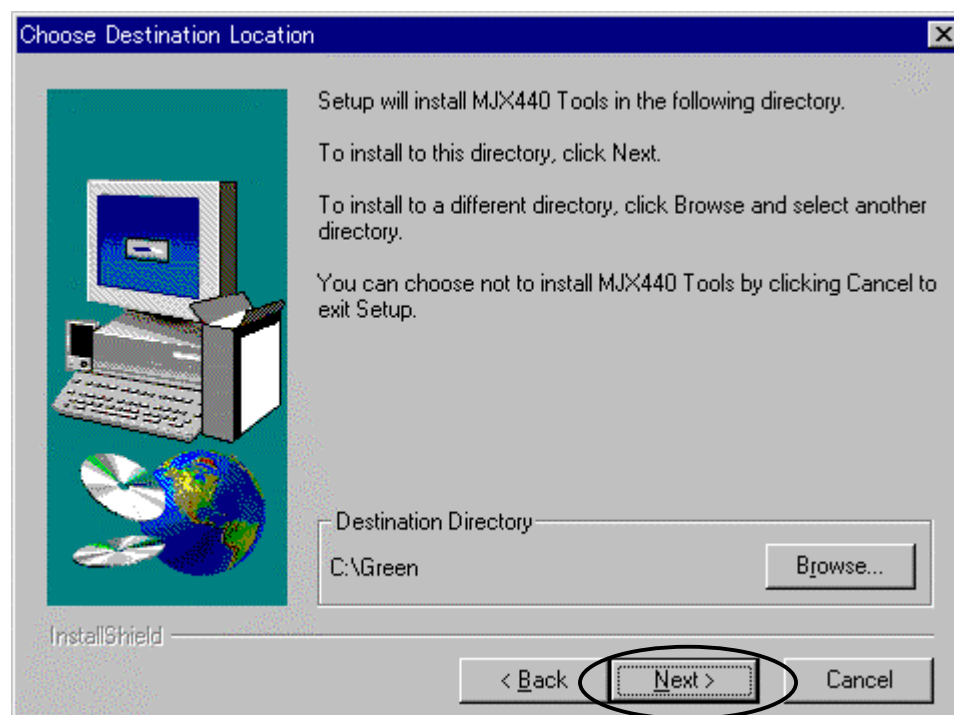
4. セットアップ プログラムが起動します。



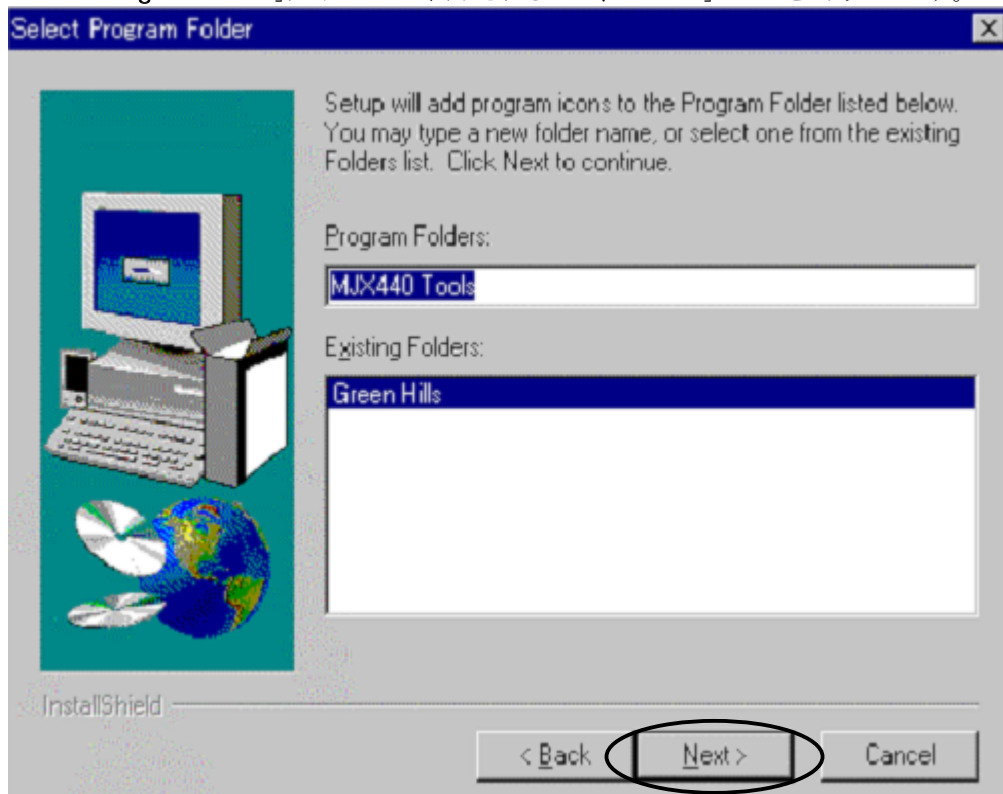
5. 「Welcome」ダイアログが表示されるので、「Next >」をクリックします。



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。

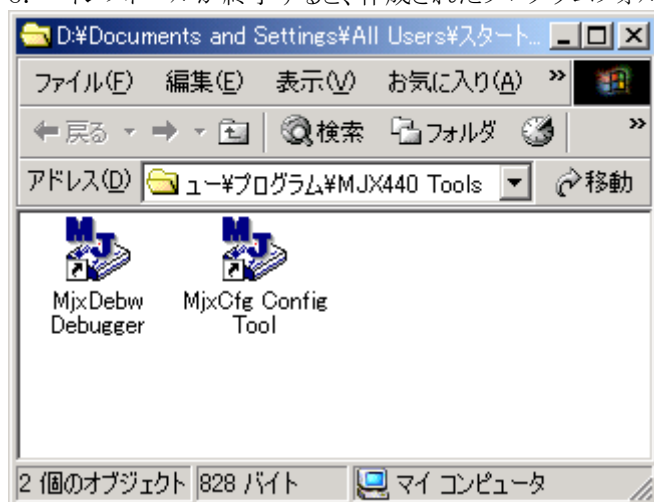


「Select Program Folder」ダイアログが表示されるので、「Next >」ボタンをクリックします。

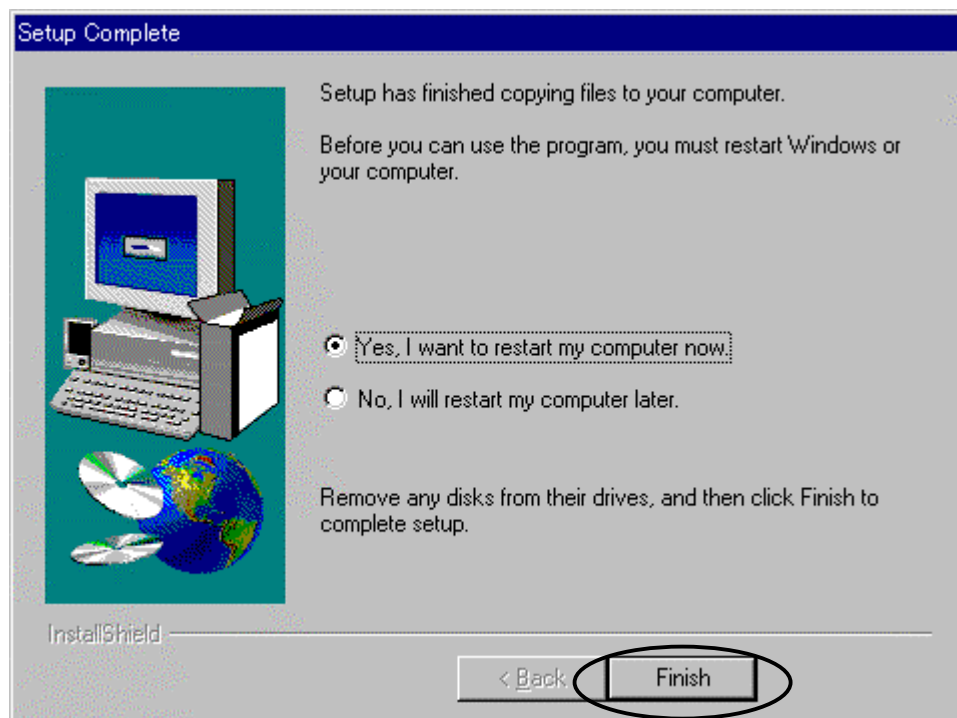


7. インストールが開始します。

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
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

この他にサブディレクトリにフラッシュメモリ書込に使用するデバイス定義ファイルや MJX330 for EJTAG2.5/2.6 の実行に必要なレジスタ定義ファイル、キャッシュ操作作用モニタのサンプルなどがインストールされます。

プロセスとハードウェアの構成は次のとおりです。

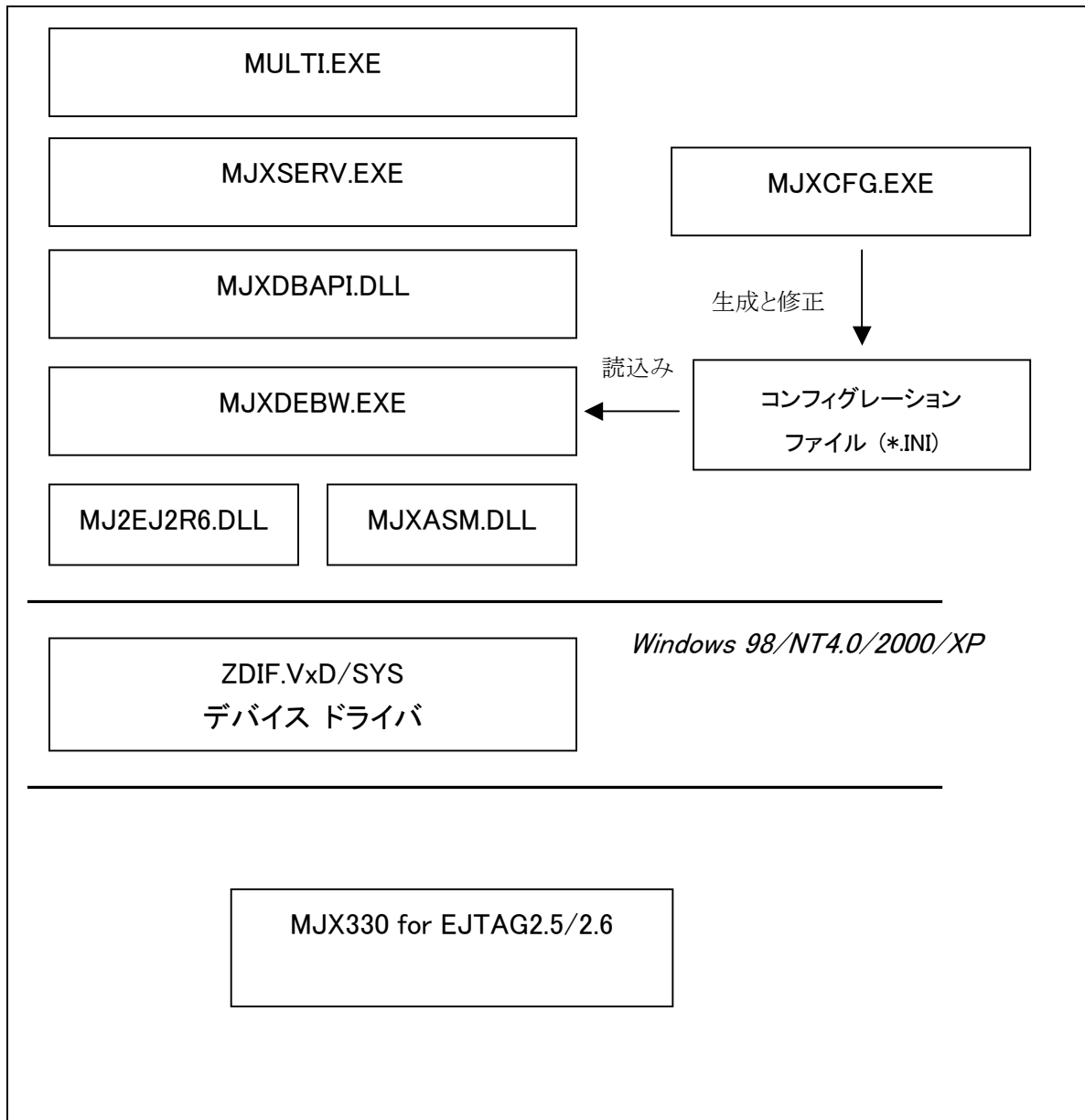


図 4-1 プロセスとハードウェアの構成

- 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。

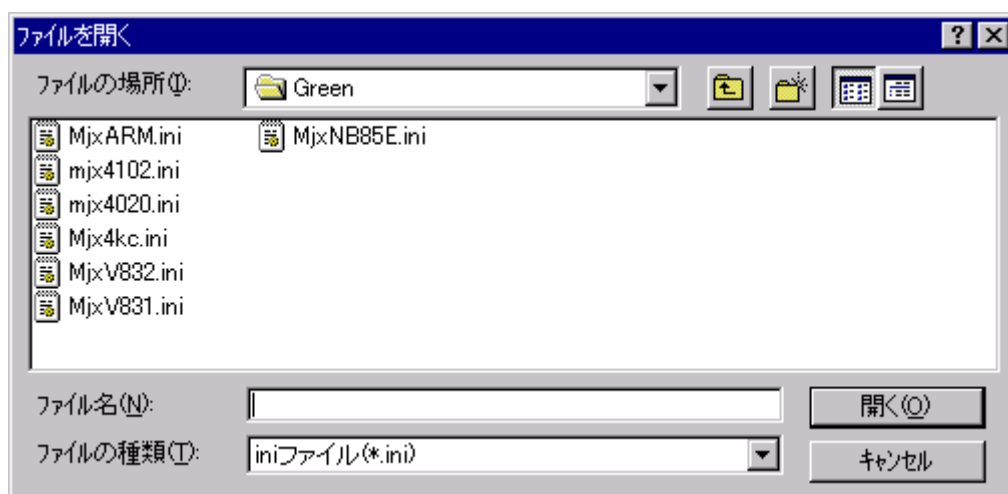


## 第五章 MJX330 の環境設定

MJX330 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。

MJX330 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。

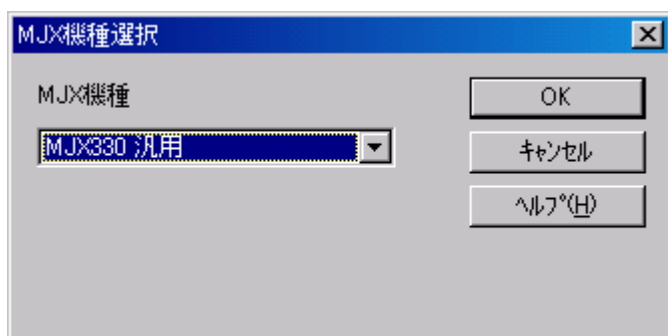
1. スタート メニュー
2. プログラム(P)
3. MJX Tools
4. MjxCfg Config Tool



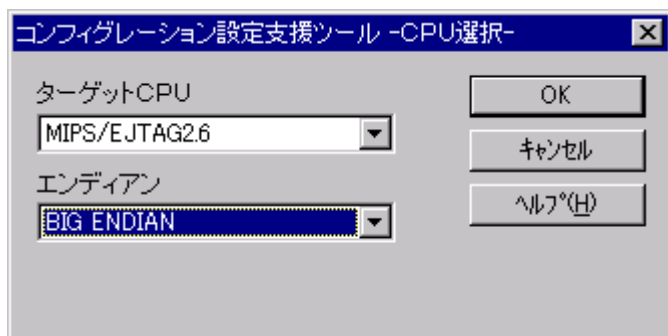
テンプレートとして **Mjx4kc.INI** を利用できます。

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「MJX330 汎用」を選択し、**OK** ボタンを押してください。

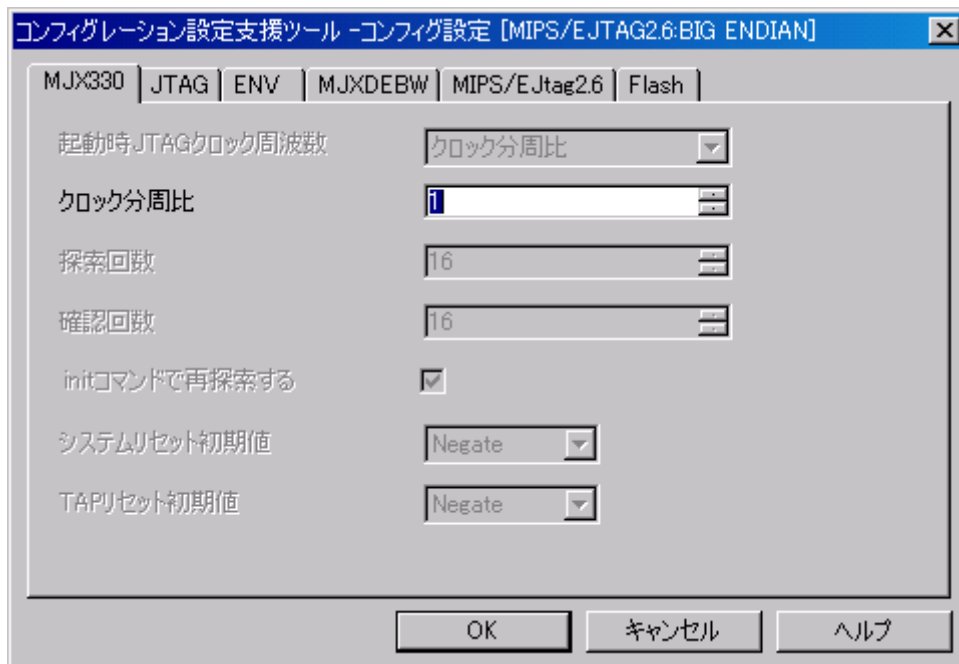


ターゲットCPU\*1とエンディアンを選択し、**OK** ボタンを押してください。

---

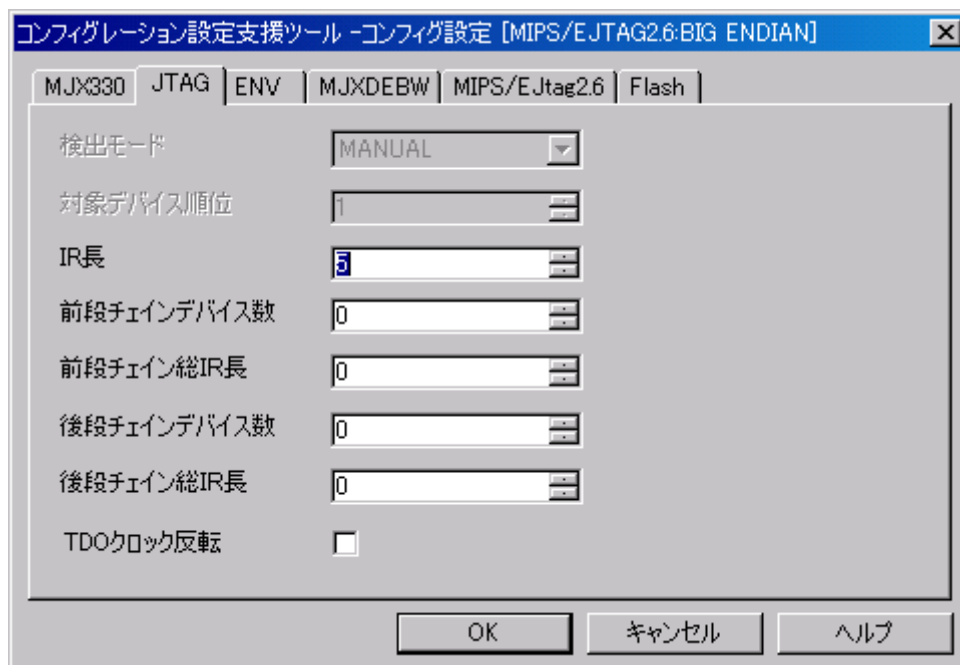
\*1 ターゲットCPUは EJTAG2.6 を指定してください。

MJX330 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。



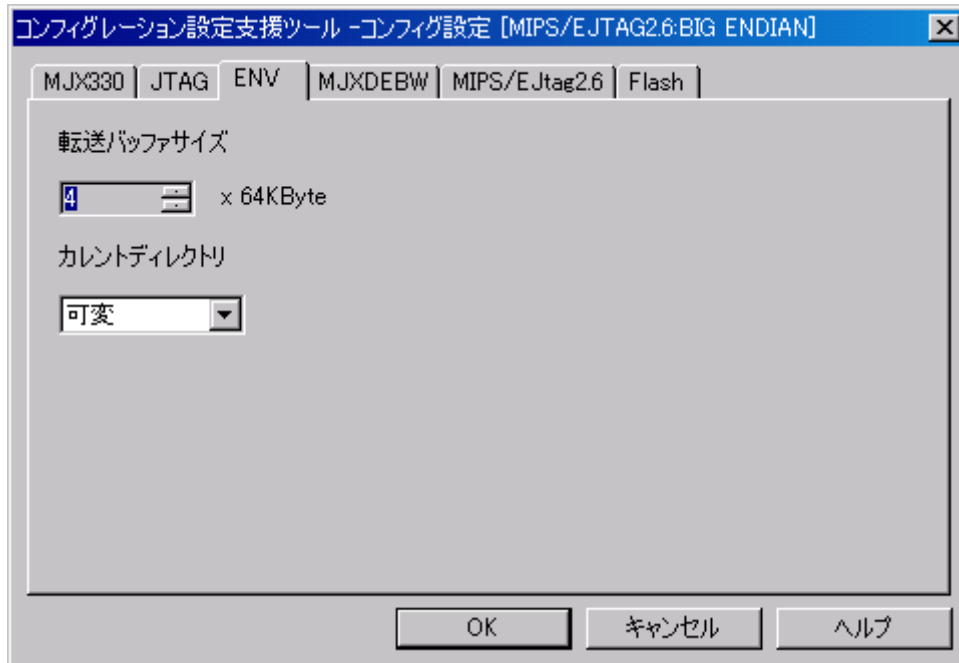
クロック分周比を設定してください。

JTAG 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。



JTAG インストラクションレジスタのビット長 (IR長)、前段／後段チェインデバイス数、前段／後段チェイン総IR長、TDO クロック反転を設定してください。

ENV 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。



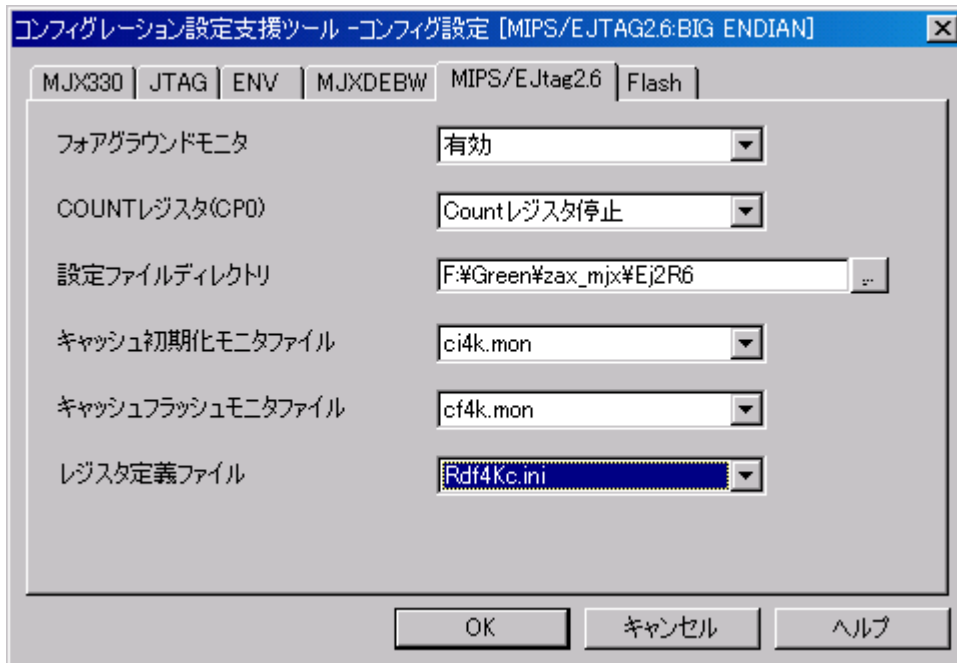
転送バッファサイズとカレントディレクトリを設定してください。

MJXDEBW 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。



リセット保持時間を設定してください。

MIPS/EJtag2.6 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、MJX330 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てください。また、設定ファイルディレクトリには、キャッシュ初期化モニタファイル、キャッシュフラッシュモニタファイル、レジスタ定義ファイルが保存されているディレクトリ名を指定してください。

**【注意】** インストール CD-ROM 付属のコンフィグレーションファイルサンプル Mjx4Kc.ini では設定ファイルディレクトリは C:\Green\zax\_mjx\Ej2R6 になっています。

インストール先を C:\Green から変更した場合は、設定ファイルディレクトリを  
<インストール先>\zax\_mjx\Ej2R6 へ変更してください。

**【注意】** キャッシュ初期化モニタファイル、キャッシュフラッシュモニタファイル、レジスタ定義ファイルにつきましてはインストール CD に 4Kc プロセッサ用のサンプルが含まれています。

サンプルファイルは

<インストール先>\zax\_mjx\Ej2R6 へインストールされます。

これらファイルのカスタマイズが必要なときは弊社営業部までお問い合わせください。

Flash 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。





## 第五章 MJX330 の環境設定

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

MJX330 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 MJX330 の環境設定」を行ってください。

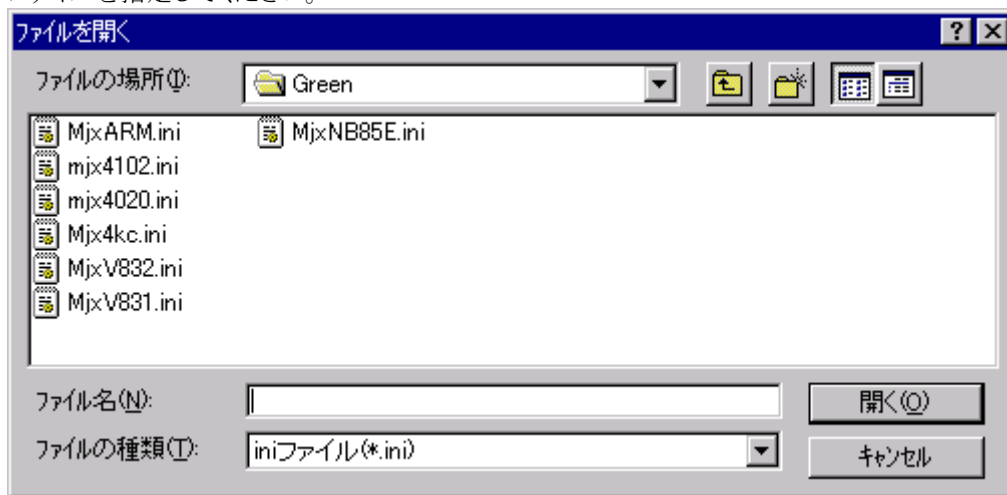
【注意】 MJX330 を操作するソフトウェアは、同時にひとつしか実行できません。

### MULTI を使用する場合

[MULTI](#)を使って MJX330 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、「[第五章 MJX330 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。<sup>\*2</sup>

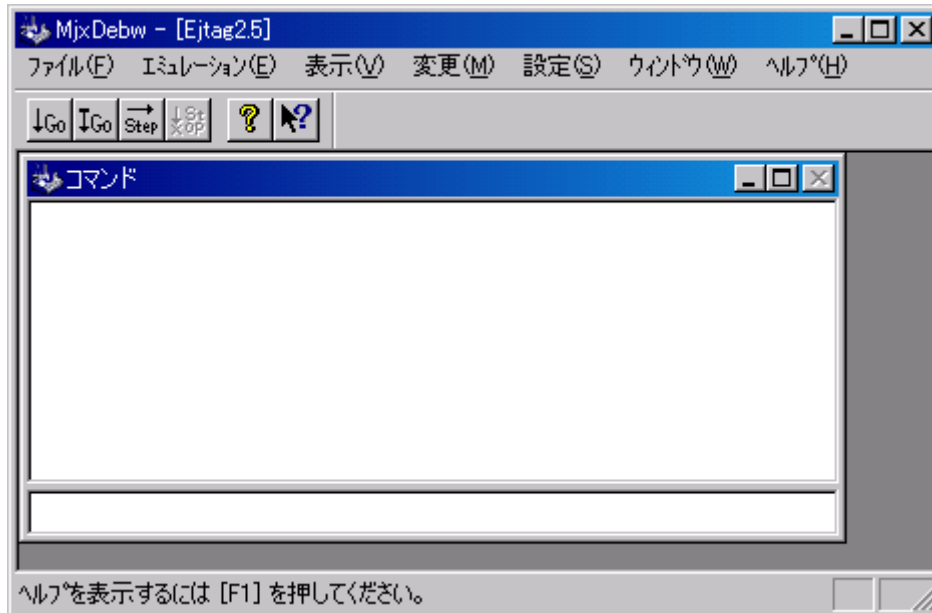


<sup>\*1</sup> ビルダから操作する場合は、サーバー名を `mjxserv` と指定し、**リモート** ボタンを押します。

<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、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 *Mjx4kc.ini* を指定します。

```
remote mjsxserv Mjx4kc.ini
```

終了するには、*quit* コマンドを入力してください。

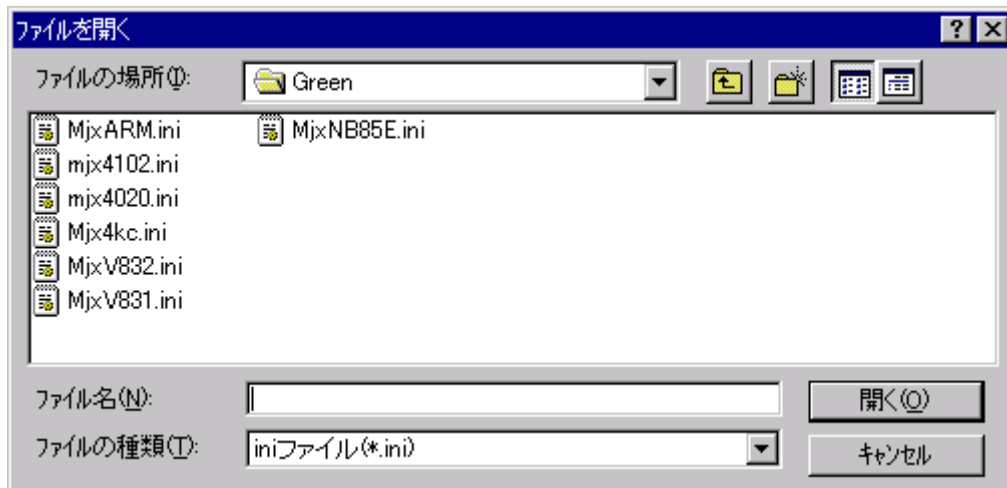
```
quit
```

### MJXDEBW を使用する場合

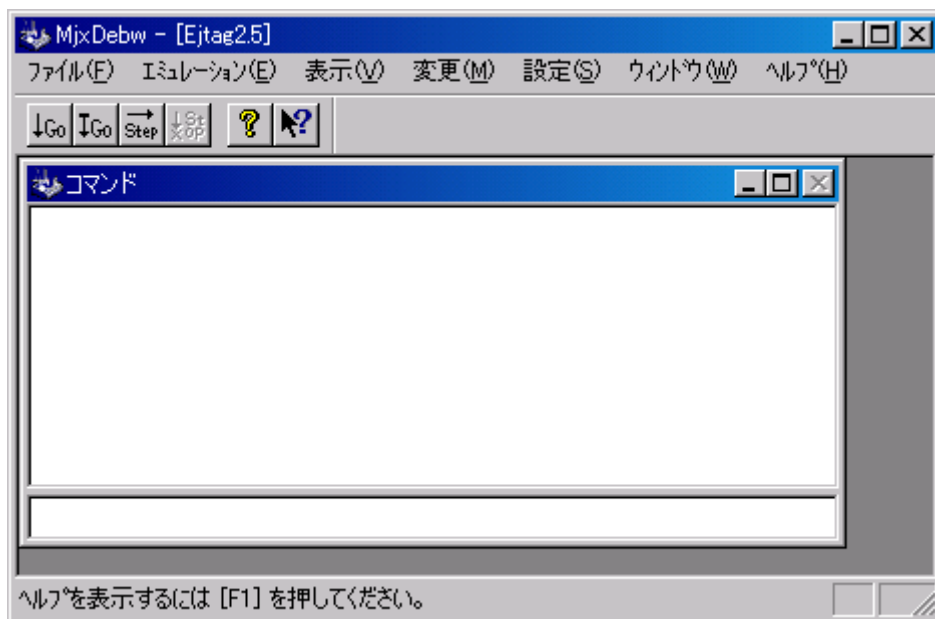
[MJXDEBW](#)を使ってMJX330 を操作するためには、スタート メニューから **MJXDEBW** を起動してください。  
手順は、次のとおりです。

1. スタート メニュー
2. プログラム(P)
3. MJX 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 *Mjx4kc.ini* を指定します。

```
MJXDEBW Mjx4kc.ini
```

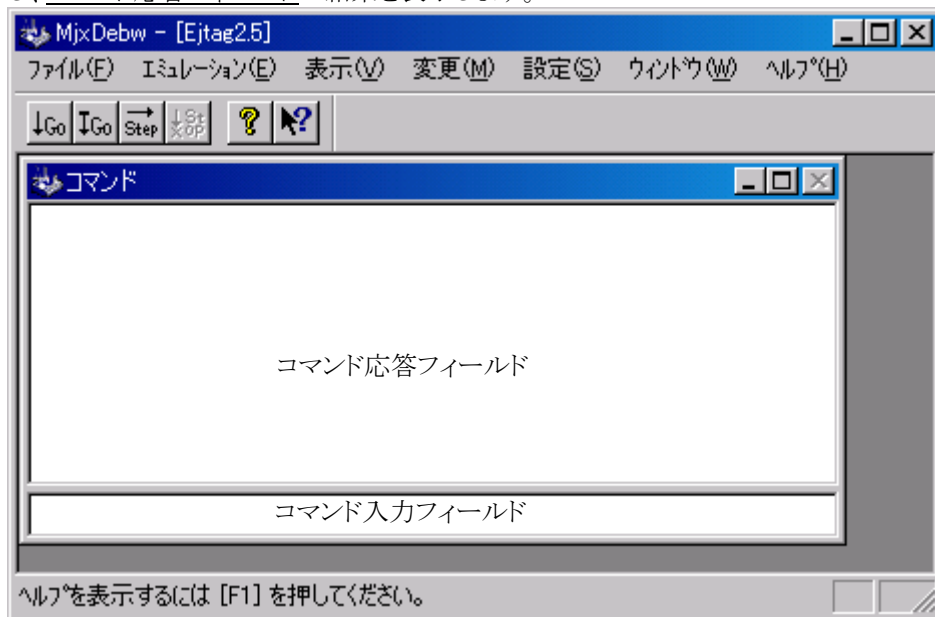
終了するためには、*quit* コマンドを入力してください。

```
quit
```

## 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。



- 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。

<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 MJXDEBW 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します
<a href="#"><u>BP/{C D E}</u></a>	ブレークポイントを解除/有効/無効にします
<a href="#"><u>BP/{A W R}</u></a>	データ アクセス ブレークポイントを設定します
<a href="#"><u>BP/H</u></a>	命令ハードウェア ブレークポイントを設定します
<a href="#"><u>BP/S</u></a>	命令ソフトウェア ブレークポイントを設定します
<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>	MJXDEBW 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>FLASH</u></a>	フラッシュ メモリの内容を書き換え、消去します
<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>	MJXDEBW 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>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>REGISTER/T</u></a>	TLB の内容を表示/変更します
<a href="#"><u>SETLOAD</u></a>	ブレークポイ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
<a href="#"><u>SETSAVE</u></a>	ブレークポイントおよびトレースの設定をファイルへ保存します
<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 トレースのトレース条件の表示/設定を行います
<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>	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

### 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	<a href="#">ABORT</a> と同じです
D	<a href="#">DUMP</a> と同じです
CL	<a href="#">CLEAR</a> と同じです
CO	<a href="#">CONFIG</a> と同じです
C	エラーです。CLEAR か CONFIG 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

[MULTI](#)を使用する場合、次の MJXDEBW コマンドは、無視されます。

[QUIT](#)      [MJXDEBW](#)の終了



また、次の MJXDEBW コマンドは、実行しないでください。

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

これらの MJXDEBW コマンドを実行した場合、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。命令ソフトウェア ブレークポイントの設定、ユーザ プログラムの実行、レジスタの変更は、MULTI のコマンドで行なうようにしてください。

MULTI 上でレジスタの変更をしても、ただちに[MJXDEBW](#)のレジスタは更新されません。MJXDEBW のレジスタは、ユーザ プログラムを実行する直前に更新されます。

## 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 コマンドを連続して実行します

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 ファイルの中の「;」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します

形式:

BP [{A|H|S}]                      設定されているブレークポイントを表示します。  
BP {N|AN|HN|SN}                  設定できるブレークポイントの数を表示します。

引数:

A              データ アクセス ブレークポイント指定  
H              命令ハードウェア ブレークポイント指定  
S              命令ソフトウェア ブレークポイント指定  
                A,H,S 省略時は、すべてのブレークポイント指定  
SN             命令ソフトウェア ブレークポイントの数を指定  
HN             命令ハードウェア ブレークポイントの数を指定  
AN             データ アクセス ブレークポイントの数を指定  
N              すべてのブレークポイントの数を指定

例:

BP                                  (すべてのブレークポイントを表示します)

補足:

- 「第八章 [設定\(S\) ～ ブレークポイント\(B\)...](#)」参照。

補足 1:

ブレークポイントは、次の形式で表示されます。

No.	Sts	Type	Addr	ASID	Addrmask	B	T	Size	Data	DataMask
0	ENA	SW	00000000							
1	ENA	INST	00001000	Glob	Disable	y	n			
2	ENA	ACCS	00002000	Glob	Disable	y	n	BYTE	Disable	Dis
3	DIS	LOAD	00005000	Glob	00000011	y	n	HALF	00004000	1001

表示される各フィールドの意味は、次のとおりです。

No.	ブレークポイントの ID 番号。(0 から始まる 10 進数)
Sts	ブレークポイントの有効/無効。 <b>ENA</b> 有効 <b>DIS</b> 無効
Type	ブレークポイントの種類。 <b>SW</b> ソフトウェア <b>INST</b> H/W 命令 <b>ACCS</b> H/W(R/W) <b>LOAD</b> H/W(Read) <b>STORE</b> H/W(Wr it e)
ADDR	アドレスを表示 (16 進数)。
ASID	ASID の有効/無効。 有効    ASID を表示 (16 進数) 無効 <b>Global</b>
AddrMask	アドレスマスクが有効のときに設定したアドレス。 有効    アドレスを表示 (16 進数) 無効 <b>Disable</b>
B	ブレークポイント リソースがブレークポイントとして利用されているか否か。 <b>y:</b> 利用されています。 <b>n:</b> 利用されていません。

T	ブレイクポイント リソースがトレース トリガとして再利用されているか否か。 y: 再利用されています。 n: 再利用されていません。
Size	アクセスサイズ。 BYTE BYTE アクセス HALF HALWORD アクセス WORD WORD アクセス DWORD DOUBLE WORD アクセス
Data	比較データを表示。 有効 比較データを表示(16進数) 無効 Disable
DataMask	バイト毎のデータマスク。 有効 1 無効 0 <例> BIT63-58: 0 BIT57-48: 0 BIT47-40: 0 BIT39-32: 0 BIT31-24: 1 BIT23-16: 0 BIT15-8 : 0 BIT7-0 : 1 のときバイトマスクは 00001001 (MIPS32 アーキテクチャの場合、4桁指定となります)

## BP/{C|D|E}

ブレークポイントを解除/有効/無効にします

形式:

BP/C {*num*|\*|A\*|H\*|S\*} ブレークポイントを解除します。  
 BP/D {*num*|\*|A\*|H\*|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。

引数:

*num*      ブレークポイント番号指定 (10 進数)  
 \*          すべてのブレークポイント指定  
 A\*        すべてのデータ アクセス ブレークポイント指定  
 H\*        すべての命令ハードウェア ブレークポイント指定  
 S\*        すべての命令ソフトウェア ブレークポイント指定

例:

BP/C S\*                      (すべての命令ソフトウェア ブレークポイントを解除します)  
 BP/D S\*                      (すべての命令ソフトウェア ブレークポイントを無効にします)  
 BP/E S\*                      (すべての命令ソフトウェア ブレークポイントを有効にします)

補足:

- 「第八章 [設定\(S\) ~ ブレークポイント\(B\)...](#)」参照。

## BP/{A|R|W}

データ アクセス ブレークポイントを設定します

形式:

BP/{A|R|W} *addr*, [*asid*], [*mask*], [*size*], [*value*], [*datamask*]

データ アクセス ブレークポイントを設定します。

引数:

/A	データ アクセス ブレークポイント指定
/R	データ読み込み ブレークポイント指定
/W	データ書き込み 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>asid</i>	ASID 比較値 (16 進数。省略時はグローバル)
<i>mask</i>	<i>addr</i> の無効ビット (省略時はマスクなし)
<i>size</i>	B: 8 ビット <i>value</i> 指定 H: 16 ビット <i>value</i> 指定 W: 32 ビット <i>value</i> 指定 D: 64 ビット <i>value</i> 指定 (省略時は W)
<i>value</i>	ブレークポイント データ指定 (16 進数。省略時はデータ比較なし)
<i>datamask</i>	<i>value</i> のマスクするバイトに 1 を立てる (4 または 8 桁 2 進数)

例:

BP/A 1000	(アドレス 0x1000 番地のデータ アクセスでブレーク)
BP/A F000, , 00FF	(アドレス 0xF000~0xF0FF 番地のデータ アクセスでブレーク) ASID 比較無効
BP/A 10000, , , W, 22	(アドレス 0x10000 番地のデータ アクセスで、データが 32 ビットで 0x00000022 のときブレーク)
BP/R 1000, , , B, 22	(アドレス 0x1000 番地の読み込みで、データが 8 ビットで 0x22 のときブレーク)
BP/R 1000, , , H, 4400, 0001	(アドレス 0x1000 番地の読み込みで、データが 16 ビットで 0x4400~0x44FF のときブレーク)
BP/W 1000, , , B, 22	(アドレス 0x1000 番地の書き込みで、データが 8 ビットで 0x22 のときブレーク)



補足:

- データ アクセス ブレークポイントは基本的には実行前ブレークです。ただし、ブレークポイントにデータ比較を指定した場合、ブレークポイントのオーバーランが発生する場合があります。(実行後ブレーク)  
 オーバーランが発生するかしないかは、CPU の実装に依存します。  
 オーバーランが発生したことは、ブレークポイント要因表示で知ることができます。
- データ読み込みブレークポイントでデータ比較をした場合でかつ、実行前ブレークとなるときは注意が必要です。このとき、ブレーク前にメモリからの読み出しが実行されますが、レジスタの更新は行われません。プログラム再開時に再度ロード命令が正しく実行されます。メモリアクセス動作が 2 回行われることになりますので、I/O ポート等に設定するときは十分に注意してください。
- *mask* は、マスクするビットを1に指定してください。
- *data mask* は、マスクするバイトを1に指定してください。
- トレース トリガとして、データ アクセス トリガを指定している場合は、設定できるブレークポイントの数がその分減ります。(TRACE/I コマンドを参照)
- ブレークポイントは、CPU の Debug Support Unit(DSU)の機能を使用しています。
- EJTAG2.5/2.6 で規定されているデバッグ機能(DSU)は、アクセスサイズによるデータブレーク条件設定をサポートしていません。  
 サイズ指定はアドレス下位ビットに応じた自動的なデータのシフトとマスクの付加に使用されます。  
 従って、以下のような制約がありますのでご注意ください。
  - (1) *size* は *value* が指定されている時のみ有効です。
  - (2) サイズ指定に関わらず、すべてのデータサイズによるデータアクセスでブレークが発生する可能性があります。
- 「第八章 [設定\(S\) ～ ブレークポイント\(B\)...](#)」参照。

## BP/H

命令ハードウェア ブレークポイントを設定します

形式:

BP/H *addr*, [*asid*], [*mask*]

命令ハードウェア ブレークポイントを設定します。

引数:

*addr*        ブレークポイント アドレス指定 (16 進数)  
*asid*        ASID 比較値 (16 進数。省略時はグローバル)  
*mask*        *addr* の無効ビット指定 (16 進数。省略時はマスクなし)

例:

BP/H F000                    (アドレス 0xF000 番地の命令を実行する直前にブレーク)  
BP/H F000, ,00FF            (アドレス 0xF000 番地～0xF0FF 番地の命令を実行する直前にブレーク)

補足:

- 命令ハードウェア ブレークポイントをは実行前ブレークです。
- *mask* は、マスクするビットを1に指定してください。
- 「第八章 [設定\(S\) ～ ブレークポイント\(B\)...](#)」参照。

## BP/S

命令ソフトウェア ブレークポイントを設定します

形式:

BP/S *addr*[,M16]      命令ソフトウェア ブレークポイントを設定します。

引数:

*addr*      ブレークポイント アドレス指定。(16 進数)

M16      MIPS16 命令の使用を指定<sup>\*1</sup>

例:

BP/S F0000000      (アドレス 0xF0000000 番地の命令実行直前にブレーク)

BP/S F0000000,M16      (アドレス 0xF0000000 番地の MIPS16 命令実行直前にブレーク)

補足:

- 命令ソフトウェア ブレークポイントは、RAM 領域に設定することができます。
- 設定可能な命令ソフトウェア ブレークポイントの数は 128 です。
- 「第八章 [設定\(S\) ~ ブレークポイント\(B\)...](#)」参照。

---

<sup>\*1</sup> 現在 MIPS16 は未対応です。

# 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:

- 「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 <i>item=value</i>	MJXDEBW の環境設定項目 <i>item</i> を <i>value</i> に設定します。 変更は直ちに行われ、初期化が必要な項目は初期化が行われます。
CONFIG/M <i>item=value</i>	MJXDEBW の環境設定項目 <i>item</i> を <i>value</i> に設定します。 初期化が不要な項目は直ちに設定が行われます。 初期化が必要な項目は <i>value</i> の設定は保留されます。 このときは保留された <i>item</i> の設定を実行するために別途 INIT コマンドを実行する必要があります。

引数:

<i>item</i>	環境設定項目、次のいずれかを指定
IRLEN	IR 長
PRE_DEVICE	前段チェインデバイス数
PRE_IRLEN	前段チェイン総 IR 長
PST_DEVICE	後段チェインデバイス数
PST_IRLEN	後段チェイン総 IR 長
INV_TDO	TDO クロック反転
JCLOCK	JTAG クロック
RST_HOLD	リセット保持時間
FGM	フォアグラウンドモニタ使用
COUNTDM	カウントレジスタ (CP0)

*value* 環境設定項目に設定する値

IRLEN の場合

4 から 16 デバッグ対象 CPU の JTAG インストラクションレジスタ(IR)のビット長を指定

PRE\_DEVICE の場合

0 から 15 JTAG チェインに複数のデバイスが接続されている場合、デバッグ対象 CPU よりも前方(TDI 側)に接続されているデバイスの総数を指定

PRE\_IRLEN の場合

0 から 255 JTAG チェインに複数のデバイスが接続されている場合、デバッグ対象 CPU よりも前方に接続されている各デバイスの IR 長の総和を指定

PST\_DEVICE の場合

0 から 15 JTAG チェインに複数のデバイスが接続されている場合、デバッグ対象 CPU よりも後方(TDO 側)に接続されているデバイスの総数を指定

PST\_IRLEN の場合

0 から 255 JTAG チェインに複数のデバイスが接続されている場合、デバッグ対象 CPU よりも後方に接続されている各デバイスの IR 長の総和を指定

INV\_TDO の場合

OFF TCK 立ち上がり(標準)

ON TCK 立ち下がり

JCLOCK の場合

1 から 4096 1~4096指定

RST\_HOLD の場合

1 から 60000 MJXDEBW プログラムがターゲット リセット時に JTAG システムリセット信号を保持する時間を指定 (ms単位)

FGM の場合

0 無効

1 有効

2 有効(高速)

COUNTDM の場合

0 停止する

1 CP0 の COUNT レジスタのカウントをそのまま継続する

例:

CONFIG	(MJXDEBW の環境を表示します)
CONFIG/S JCKLOCK=2	(JTAG クロックをベースクロックの2分周に設定します)
CONFIG/S FGM=0	(フォアグラウンドモニタの使用を禁止します)

補足:

- 設定 FGM = 2 は Etag2.5 では使用できません。
- 「第八章 [設定\(S\) ～ コンフィグレーション\(C\) ～ 設定\(S\)...](#)」参照。

## DUMP

メモリの内容を表示します

形式:

DUMP[/B|W|L|D] [*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
/D	64 ビット指定
<i>addr1</i>	メモリ表示開始アドレス (16 進数)
<i>addr2</i>	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/L 2000,20FF	(0x2000 から 0x20FF 番地のメモリを 32 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。
- MIPS32 アーキテクチャでは 64 ビット指定はできません。
- 「第八章 [表示\(V\) ~ メモリ\(M\)...](#)」参照。



## EXAMINE

メモリの内容を変更します

形式:

EXAMINE/{B|W|L|D} *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:

/B	8 ビット指定
/W	16 ビット指定
/L	32 ビット指定
/D	64 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:

- MIPS32 アーキテクチャでは 64bit 指定はできません。
- 「第八章 [表示\(V\) ~ メモリ\(M\)...](#)」参照。

# FILL

メモリの内容をフィルします

形式:

`FILL/{B|W|L|D} 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  
 /D      64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:

- MIPS32 アーキテクチャでは 64bit 指定はできません。
- 「第八章 [変更\(M\) ～ フィル\(F\)...](#)」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*, *offset*]]

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(既定値)

/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 デバイス名 (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

*offset*        ダウンロードファイルを指定された書き込みアドレスに、この値 (16 進数) を加算したアドレスにデータが書き込まれます (省略時は = 0)

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 G 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。
- 本機能が対応するフラッシュ メモリの一覧は、「[付録 H 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[/M16] *addr*                      アドレス *addr* からユーザ プログラムを実行します。

引数:

M16              Mips16e ASE モードでプログラムを開始します。<sup>\*1</sup>

*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 を使用している場合は、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。

---

<sup>\*1</sup> M16 は、現在サポートされていません。

## HISTORY

リアルタイム トレース結果を表示します

- ヒストリ コマンドは現在未対応です。

# 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:

/A	追記出力指定	
/W	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<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。

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 を実アドレス+0x2000 番地のメモリへダウ  
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。
- 「第八章 [エミュレーション\(E\) ～ ダウンロード\(L\)...](#)」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。

## MOVE

メモリの内容をブロック転送します

形式:

MOVE[/{B|W|L|D}] *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  
 /D      64 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/B 1000,10FF,2000    (0x1000 から 0x10FF 番地までのメモリをバイト単位で 0x2000 番地へブロック転送します)

補足:

- ビット指定省略時は、8 ビット指定が適用されます。
- 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。
- MIPS32 アーキテクチャでは 64bit 指定はできません。
- 「第八章 [変更\(M\) ～ 移動\(M\)...](#)」参照。

# 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*={ENABLE|DISABLE}   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

INTE      INT ピン イネーブル

NMIE      NMI   ピン   イネーブル

SRSTE SRST ピン イネーブル

ENABLE      有効にする

DISABLE      無効にする

例:

PIN (ピンの有効/無効の状態を表示します。)

PIN INTE=DISABLE (INT ピンを無効にします。)

# 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	すべてのレジスタの内容を表示します。
REGISTER/{G GPR}	すべての汎用レジスタの内容を表示します。
REGISTER/{O CP0}	すべてのコプロセッサ0レジスタの内容を表示します。
REGISTER/{F FPU}	すべての浮動小数点レジスタの内容を表示します。 表示形式はレジスタ定義ファイルに依存します。
REGISTER/FH	すべての浮動小数点レジスタの内容を 16 進数形式で表示します。
REGISTER/FS	すべての浮動小数点レジスタの内容を単精度浮動小数点形式で表示します。
REGISTER/FD	すべての浮動小数点レジスタの内容を倍精度浮動小数点形式で表示します。
REGISTER/FPS	すべての浮動小数点レジスタの内容をペアド単精度浮動小数点進数形式で表示します。
(REGISTER/{2 CP2}	コプロセッサ2レジスタの内容を表示します。)* <sup>1</sup>
REGISTER <i>RegName= val</i>	レジスタ <i>RegName</i> の内容を <i>val</i> に変更します。 (レジスタ名は自動検索します)
REGISTER/{G GPR} <i>RegName= val</i>	汎用レジスタ <i>RegName</i> の内容を <i>val</i> に変更します。
REGISTER/{O CP0} <i>RegName= val</i>	コプロセッサ0レジスタ <i>RegName</i> の内容を <i>val</i> に変更します。
REGISTER/{F FPU} <i>RegName= val</i>	浮動小数点レジスタ <i>RegName</i> の内容を <i>val</i> に変更します。(FGR: <i>val</i> =レジスタ定義ファイル依存, FCR: <i>val</i> =16 進数)
REGISTER/FH <i>RegName= val</i>	浮動小数点レジスタ <i>RegName</i> の内容を <i>val</i> に変更します。(FGR: <i>val</i> =16 進数, FCR: <i>val</i> =16 進数)
REGISTER/FS <i>RegName= val</i>	浮動小数点レジスタ <i>RegName</i> の内容を <i>val</i> に変更します。(FGR: <i>val</i> =単精度浮動小数点, FCR: <i>val</i> =16 進数)

\*<sup>1</sup> コプロセッサ2レジスタの内容表示は、現在サポートされていません。

REGISTER/FD *RegName=val* 浮動小数点レジスタ *RegName* の内容を *val* に変更します。(FGR: *val*=倍精度浮動小数点, FCR: *val*=16進数)

REGISTER/FPS *RegName=val1, val2* 浮動小数点レジスタ *RegName* の内容を *val1, val2* から Pairedsingle 形式に変換した値に変更します。  
(FPR: *val1, val2*=単精度浮動小数点, FCR: *val*=不可)

引数:

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

例:

REGISTER (すべてのレジスタの内容を表示します。)  
REGISTER/G (すべて汎用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。)  
REGISTER/CP0 STATUS=3000 (Status を 0x3000 に変更します。)  
REGISTER/F FCSR=0x1000 (FCSR を 0x1000 に変更します。)

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。
- 変更したレジスタ値は MJXDEBW プログラム内にバッファされ、アプリケーションプログラム実行前に CPU に書込みが行われます。
- FCCR(FCR25), FEXR(FCR26), FENR(FCR28) の各レジスタはレジスタ定義ファイルにより読み出し専用として定義されています。  
FPU 制御レジスタに新たな値を書き込む場合は FCSR(FCR31) へ値を設定してください。
- 「第八章 [表示\(V\) ～ レジスタ\(R\)](#)」参照。

形式:

*Index* 設定する TLB エントリのインデックスを指定します。

1KB\*1

4KB

16KB

64KB

256kB

1 MB

4MB

6MB

16MB

64MB

256MB

1A 論理アドレスを指定します。(16 進数)

**ASID** ASID を指定します。(16 進数)  
グローバル指定を行うときは文字'G'を指定します。

**PA0** 偶数ページの物理アドレスを指定します。(16 進数)

**C0** 偶数ページのキャッシュコヒーレンシ属性を指定します。  
0～7 の数値を指定します。

**DO** 偶数ページの Dirty ビットを指定します。**y** または **n** を指定します。

95

<i>V0</i>	偶数ページの Valid ビットを指定します。 <i>y</i> または <i>n</i> を指定します。
<i>PA1</i>	奇数ページの物理アドレスを指定します。(16 進数)
<i>C1</i>	奇数ページのキャッシュコヒーレンシ属性を指定します。0 ~ 7 の数値を指定します。
<i>D1</i>	奇数ページの Dirty ビットを指定します。 <i>y</i> または <i>n</i> を指定します。
<i>V1</i>	奇数ページの Valid ビットを指定します。 <i>y</i> または <i>n</i> を指定します。

例:

```
REGISTER/T           (すべての TLB エントリを表示します。)
REGISTER/T  I       (TLB レジスタを初期化します。)
REGISTER/T  0,1MB,0,G,0,3,y,y,100000,3,y,y
                    (TLB エントリ0へ値を設定します。)
```

補足:

- TLB 設定はコマンド実行後直ちに CPU レジスタに書込みが行われます。



## SETLOAD

ブレイクポイント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

ブレイクポイントの設定をファイルへ保存します

形式:

SETSAVE *filename*      ブレイクポイント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、実行しないでください。  
「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 トレースのトレース条件を表示/設定します

- トレース コマンドは現在未対応です。

## UNASM

メモリの内容を逆アセンブル表示します

形式:

UNASM[/E] [*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:

/E        Mips16e ASE モードで逆アセンブルします  
*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。

## 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 情報
	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 プログラムを強制停止します)

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ます (GO コマンドの実行後に発行します)。

## 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、レジスタ、逆アセンブル ウィンドウ、トレース結果ウィンドウ <sup>*1</sup> 。
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。

<sup>\*1</sup> トレース結果ウィンドウは現在未対応です。



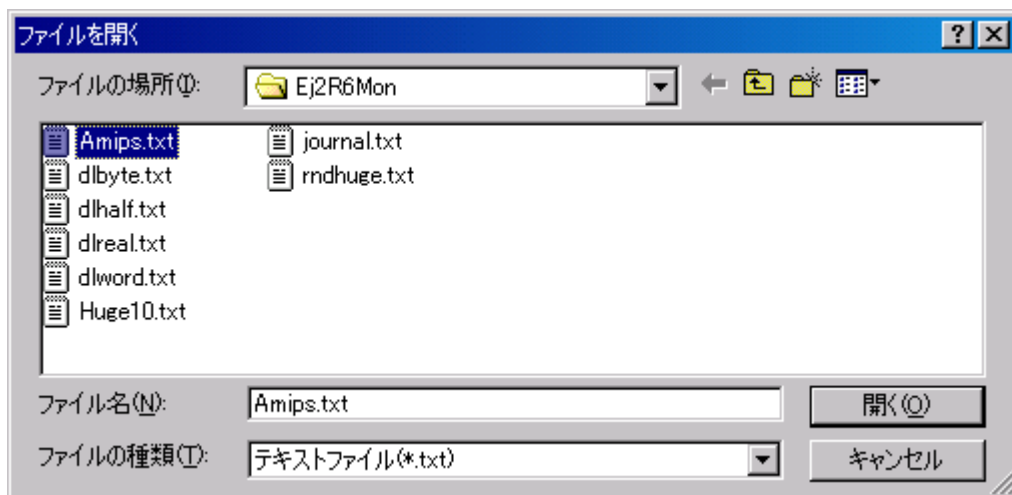
## ファイル(F) ～ バッチ(B)...

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)...

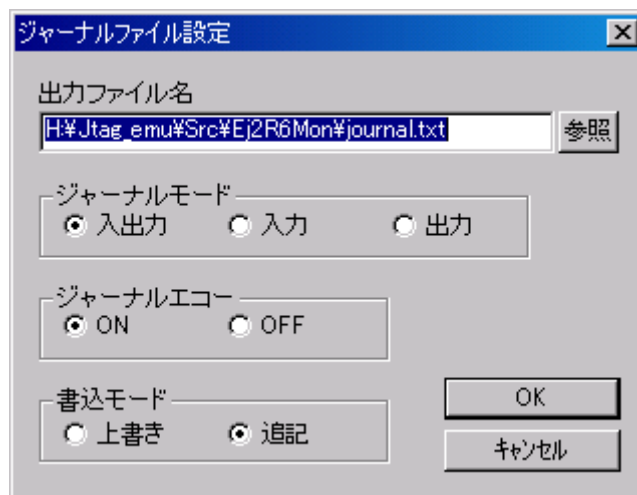
---

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

- 現在未対応です。

## エミュレーション(E) ～ 実行(G)...

---

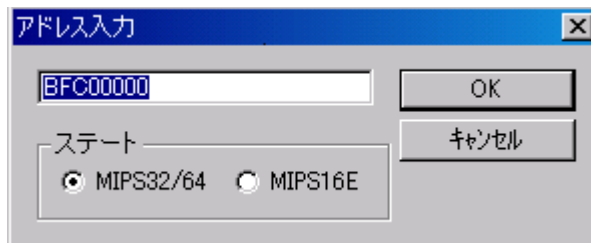
GO *addr*

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:

- 「第七章 [STEP](#)」参照。

## エミュレーション(E) ～ N-ステップ(N)...

---

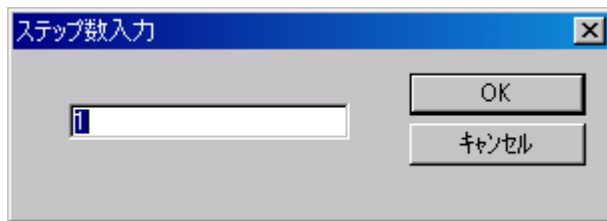
STEP *num*

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)...

---

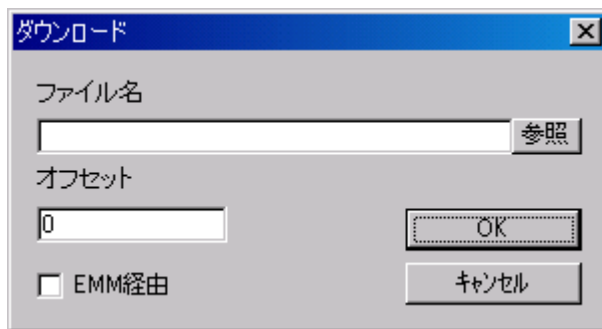
LOAD

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。
- MJX330 では EMM 経由の指定はできません。
- 「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:

MJX を再初期化します。

補足:

- 「第七章 [INIT](#)」参照。



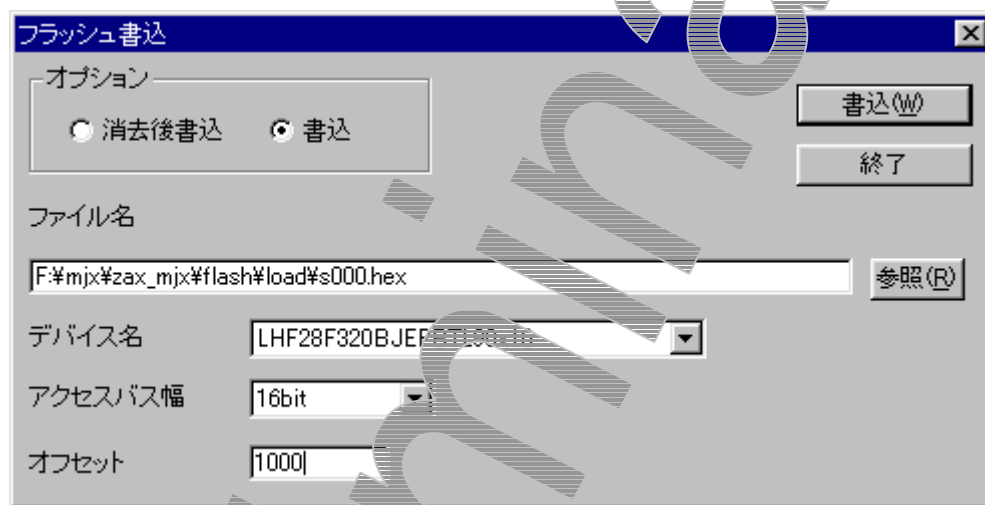
## エミュレーション(E) ～ フラッシュメモリ(F) ～ 書込(W)...

FLASH/WRITE

機能:

MJX バイナリ、S レコード、インテル HEX、または COFF 形式の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、オフセットを指定して、「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7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](#)」参照。

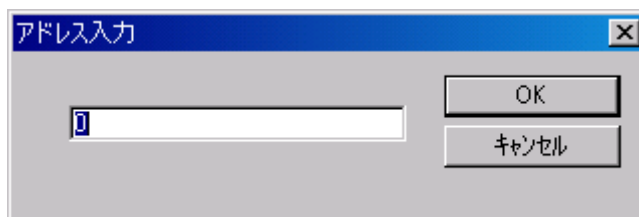
## 表示(V) ～ メモリ(M)...

DUMP

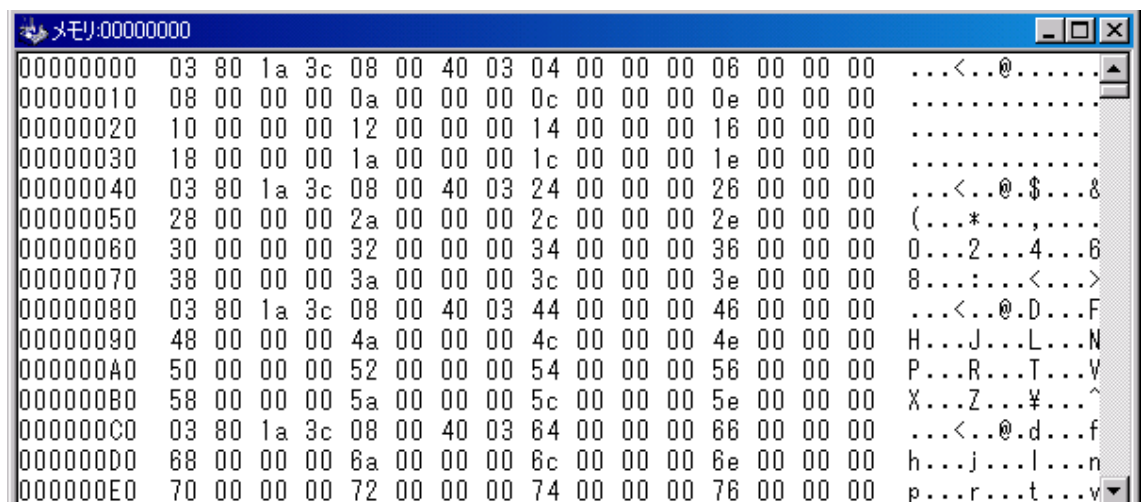
機能:

メモリの内容を表示する、メモリ ウィンドウを開きます。

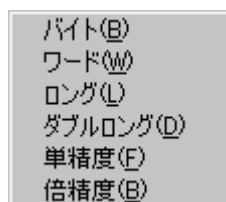
操作:



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。

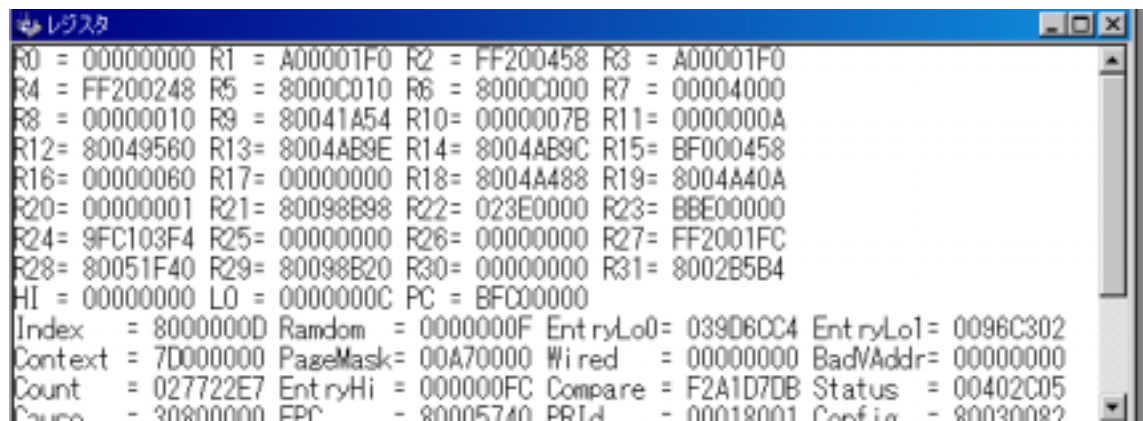
## 表示(V) ～ レジスタ(R)

### 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。

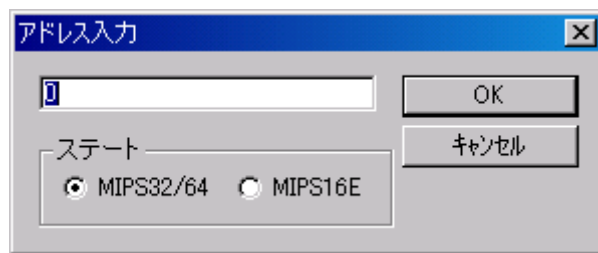
## 表示(V) ～ 逆アセンブル(D)...

UNASM

機能:

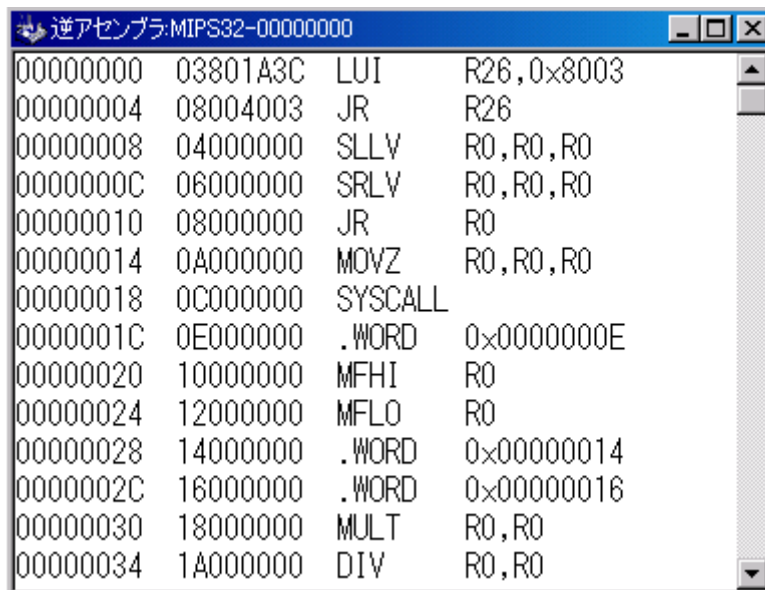
メモリの内容を逆アセンブル表示する、逆アセンブラ ウィンドウを開きます。

操作:



ダイアログ ボックスで、逆アセンブル表示を開始するアドレス(16 進数)、ステートを指定し、「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

- 現在未対応です。

## 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:

ウインドウに表示する文字のフォントを変更します。

操作:

フォント応答(O)...  
メモリ表示(M)...  
レジスタ(R)...  
逆アセンブル(D)...  
トレース結果(T)...

メニューで、フォントを変更したいウインドウを選びます。



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。

## 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。

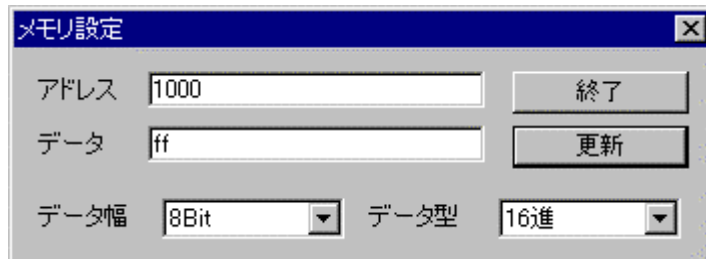
## 変更(M) ～ メモリ(M)...

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:

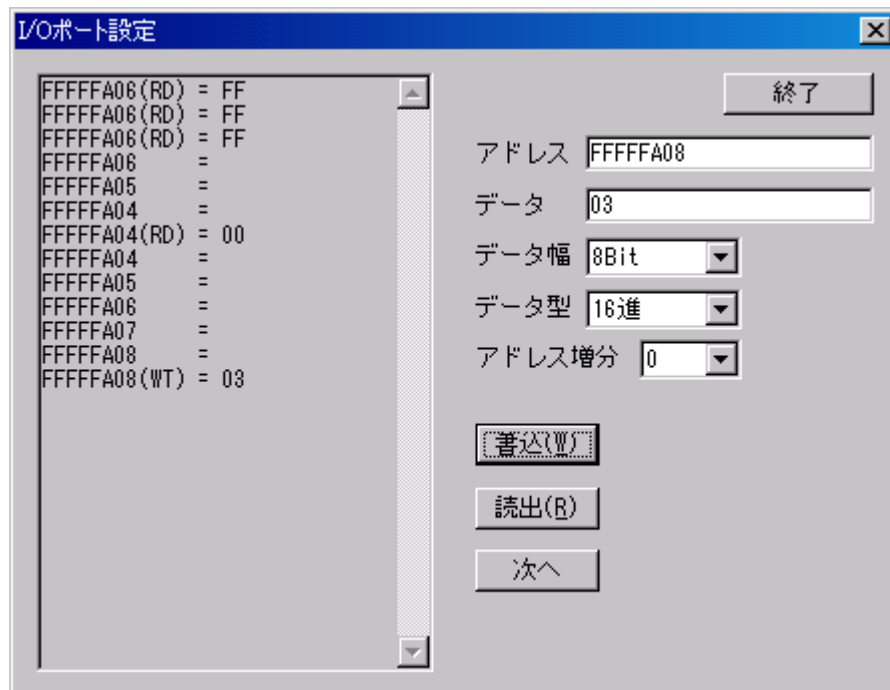
- 「第七章 [EXAMINE](#)」参照。

## 変更(M) ～ 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:

- MJX330 for EJTAG2.5/2.6 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)...

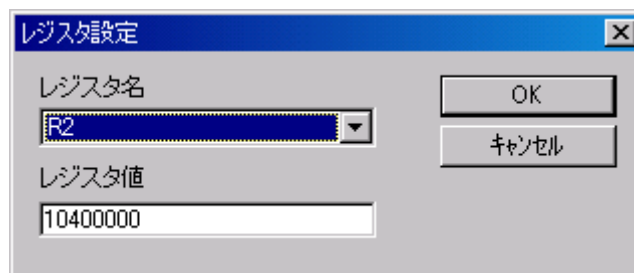
---

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:

- レジスタ設定ウインドウからは TLB の変更はできません。REGISTER コマンドを使用してください。
- 「第七章 [REGISTER](#)」参照。

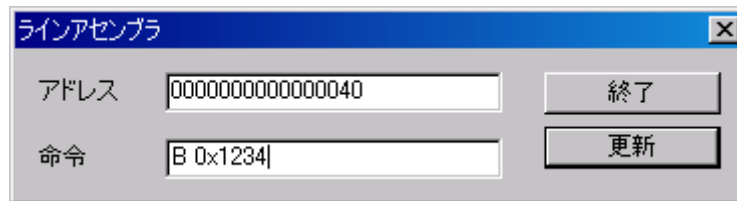
## 変更(M) ～ アセンブラ(A)...

---

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。

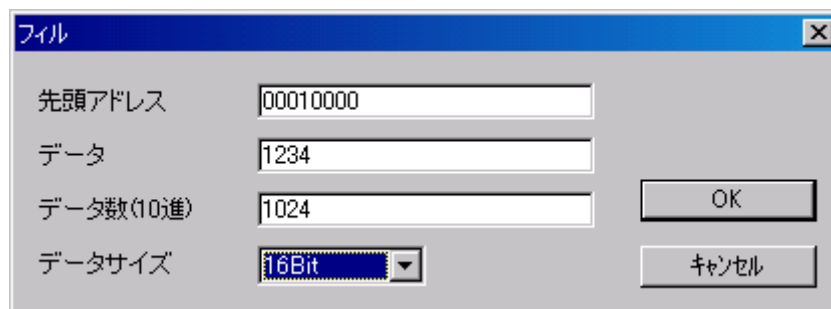
## 変更(M) ～ フィル(F)...

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。

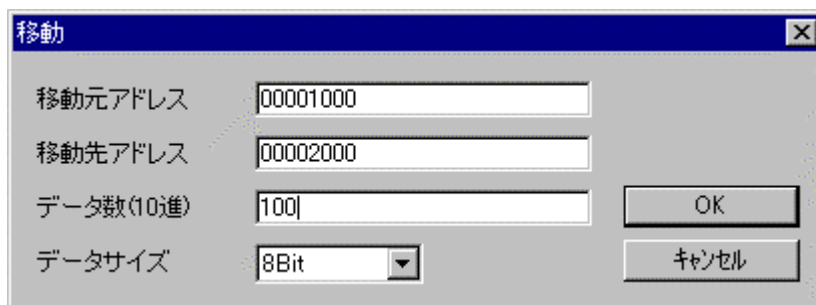
## 変更(M) ～ 移動(M)...

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

BP/S

BP/{A|W|R}

機能:

命令ソフトウェア、命令ハードウェア、アクセス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

ブレークタイプ:

アドレス:

☐ ASID有効:

☐ アドレスマスク有効:

データサイズ:

☐ データ比較有効:

データマスク: ☐ B83-56 ☐ B55-48 ☐ B47-40 ☐ B39-32  
☐ B31-24 ☐ B23-16 ☐ B15-8 ☐ B7-0

現在の設定

状...	タイプ	アドレス	A...	アド...	B..	T..	アクセ...	デー...	バ...
ENA	H/W(R/W)	00000000	00	-----	*	-	WORD	-----	----
ENA	H/W(Read)	00000000	00	-----	*	-	WORD	-----	----
DIS	H/W(Write)	00000000	00	-----	*	-	WORD	-----	----

表示選択:

最大H/W命令ブレーク数:  最大H/Wアクセスブレーク数:

「設定」フレームにある、ブレークタイプ、アドレス(16 進数)、ASID 有効、アドレスマスク有効、データサイズ、データ比較有効、データマスクを指定して「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。

「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「有効化」ボタンを押すと、ブレークポイントを有効化できます。同様の手順で「無効化」および「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。

「表示選択」ダイアログボックスは、「現在の設定」に表示するブレークポイントの種類(ブレークタイプ)を決定します。

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」フレーム内のデータはブレークタイプを選択すると、それ以下の項目は設定に必須な場合のみ入力できるようになっています。
- 登録、無効化、有効化、削除は、直ちにブレークポイントに設定されます。「キャンセル」ボタンを押しても設定のキャンセルはできません。
- 「第七章 [BP](#) ,[BP/H](#) , [BP/S](#) ,[BP/{A|R|W}](#)」参照。

## 設定(S) ～ トレース(T)...

---

TRACE

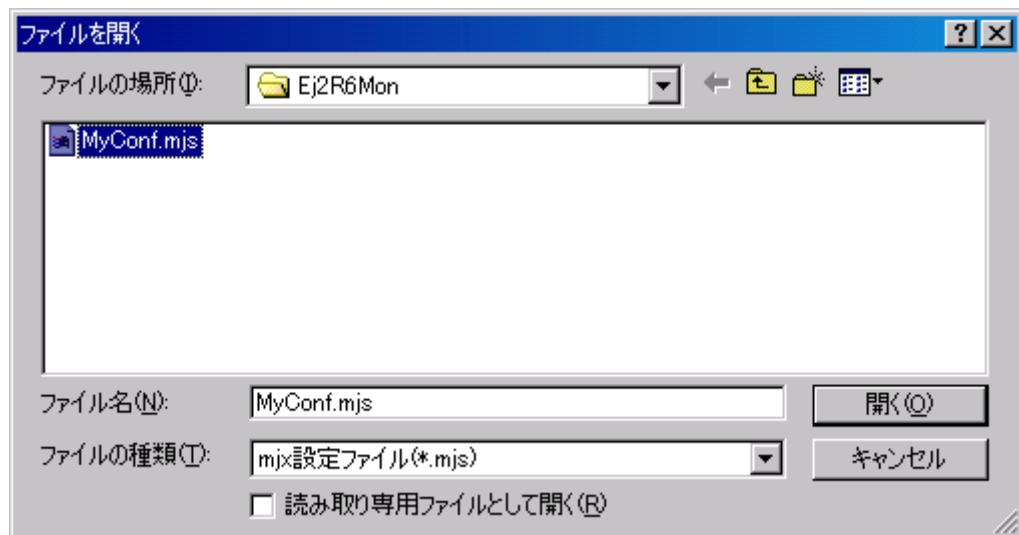
- 現在未対応です。

## 設定(S) ～ 設定の読出(L)...

---

機能:

ブレイクポイントの設定を、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 [SETLOAD](#)」参照。

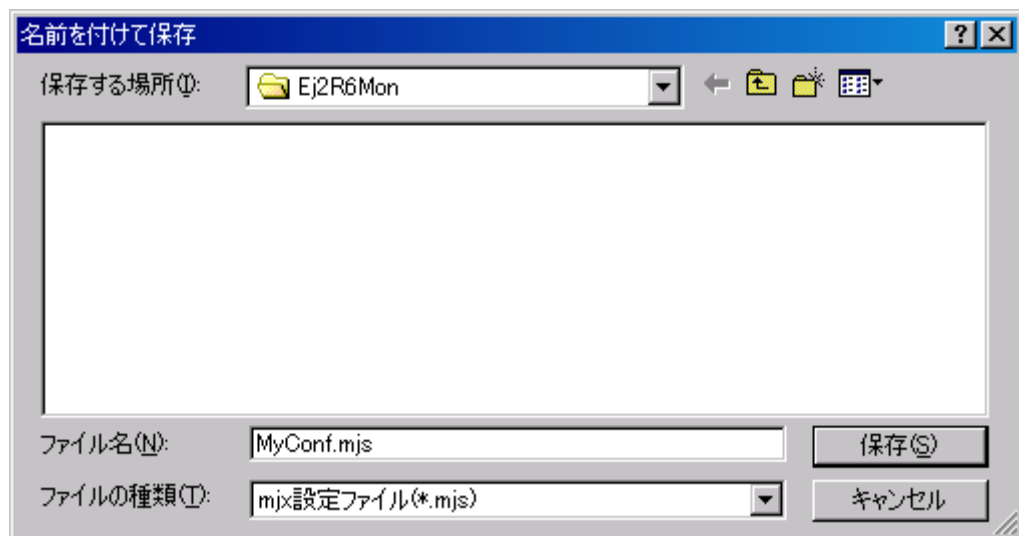
## 設定(S) ～ 設定の保存(S)...

---

機能:

ブレイクポイントの設定を、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、「保存」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。
- 「第七章 [SETSAVE](#)」参照。

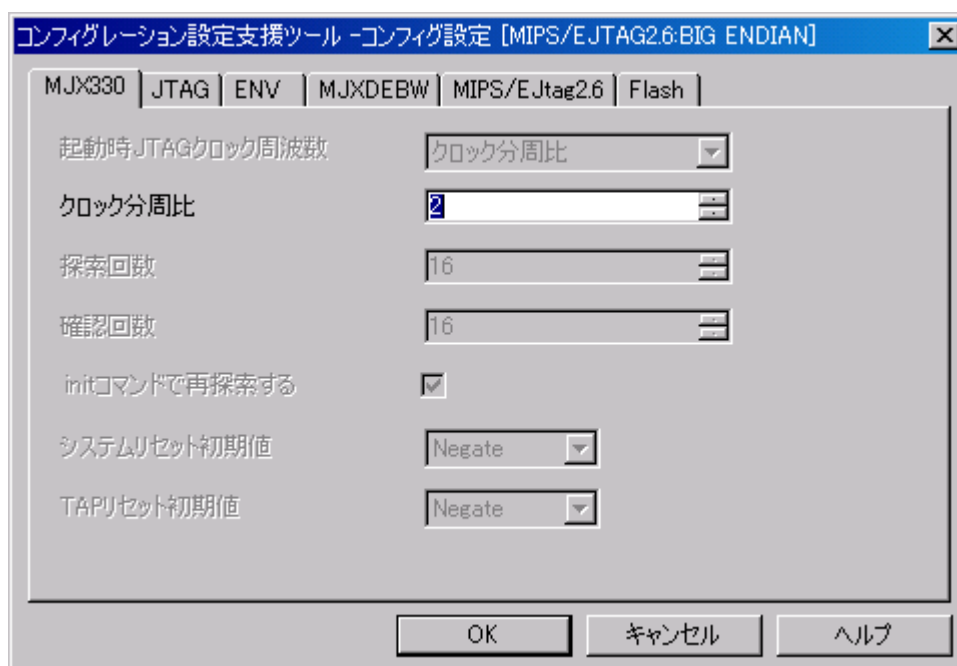
## 設定(S) ～ コンフィグレーション(C) ～ 設定(S)...

CONFIG

機能:

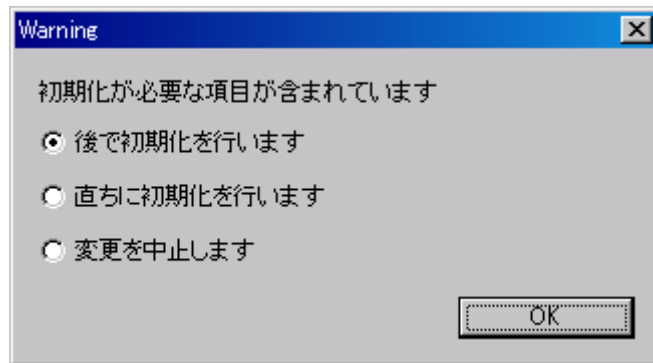
MJXDEBW の環境を変更します。

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。

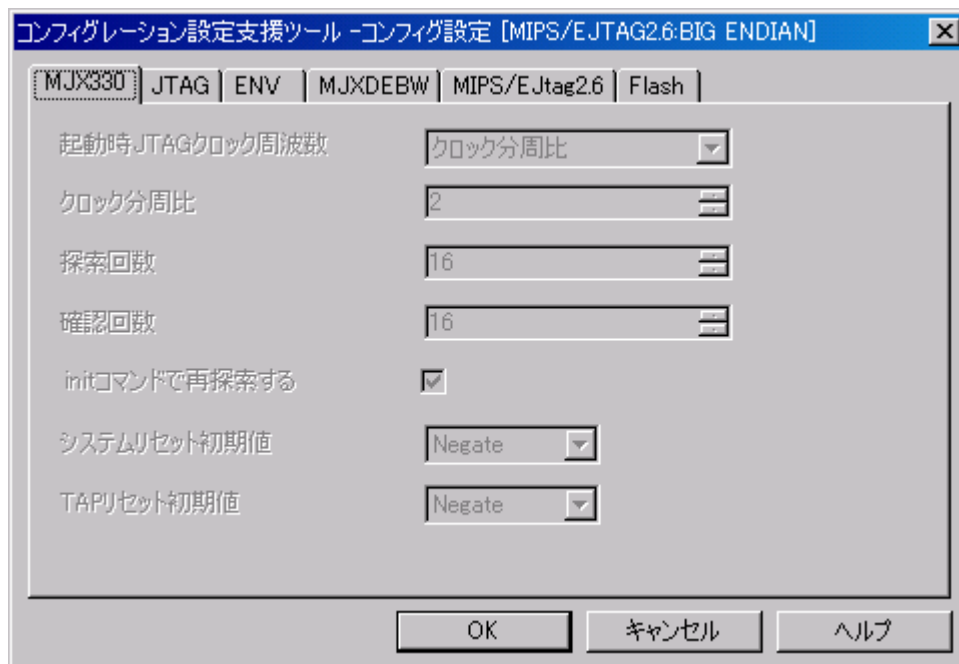
## 設定(S) ～ コンフィグレーション ～ 表示

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:

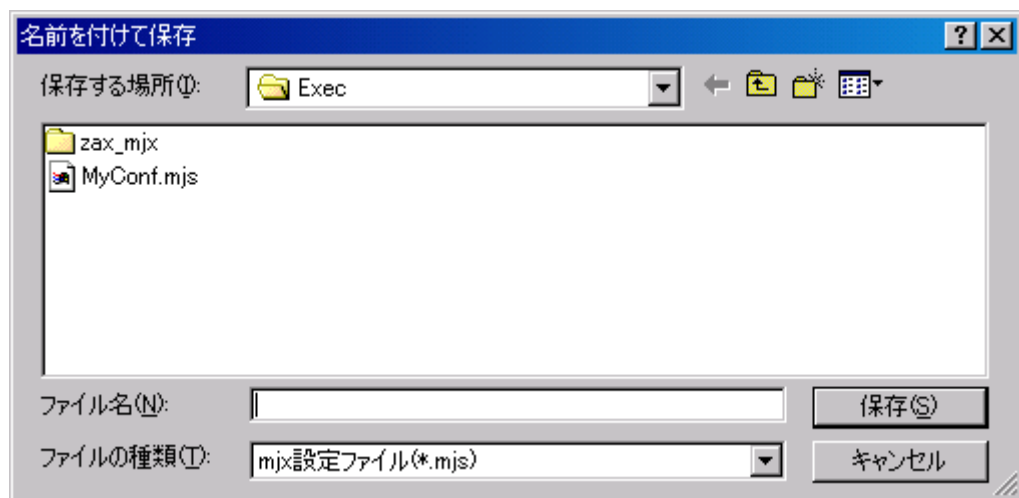
## 設定(S) ～ コンフィグレーション ～ 名前を付けて保存(A)

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。

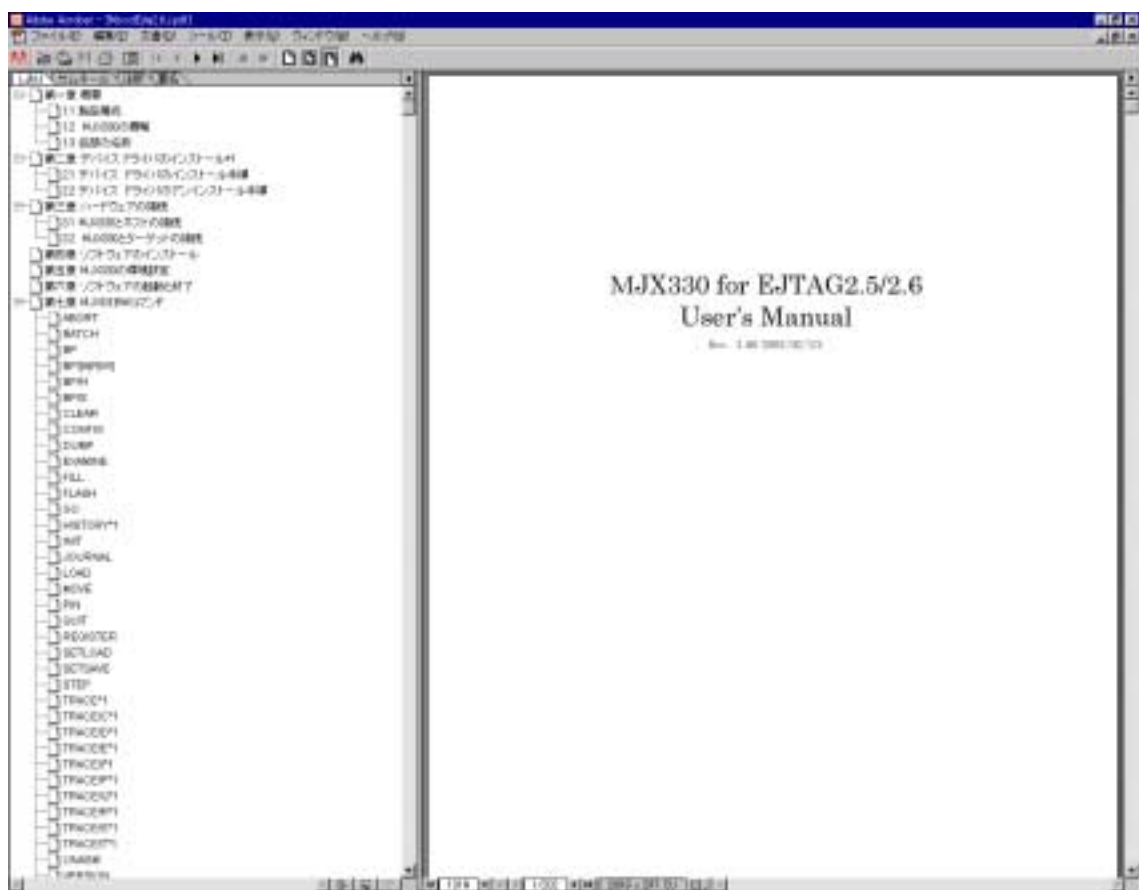
## ヘルプ(H) ～ トピックの検索(H)

機能:

MJX330 for Etag2.5/2.6 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)

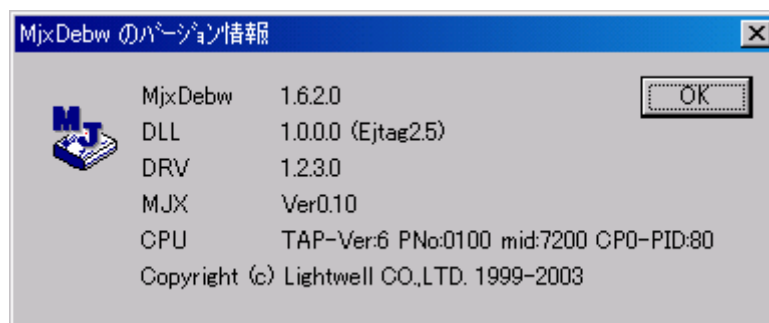
---

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。

- 700K バイト/秒 \*<sup>1</sup>(JTAG クロック:20MHz,MIPS32,Ejtag2.6)

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」\*<sup>2</sup> 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 バイナリ ファイル](#)は、MJXDEBW 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.mjx
```

---

\*<sup>1</sup> 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。

\*<sup>2</sup> Windows 2000/NT/XP に関しては「コマンドプロンプト」内から実行してください。

## 付録 A 仕様

本体寸法	PC カード TYPE II (85.6mm(縦)×54.0mm(横)×5.0mm(厚さ))
本体重量	33g
使用温度範囲	0℃～35℃
保存温度範囲	-10℃～55℃
周囲湿度範囲	30%～85%
インターフェース	PCMCIA
ターゲット インターフェース	<a href="#">EJTAG コネクタ</a>
対応 CPU	MIPS(EJTAG2.5/2.6)
ダウンロード速度	700K バイト/秒 *1 (JTAG クロック:20MHz,MIPS32,Ejtag2.6)
対応デバッグ	Green Hills <a href="#">MULTI</a>
ブレークポイント機能	命令        Max16*2
	アクセス    Max16*2
	ソフトウェア ×128
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

---

\*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。

\*2 CPU の実装により異なります。

## 付録 B ターゲット システムの制限事項

MJX330 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。

- EJTAG コネクタを実装している。

## 付録 C EJTAG コネクタ

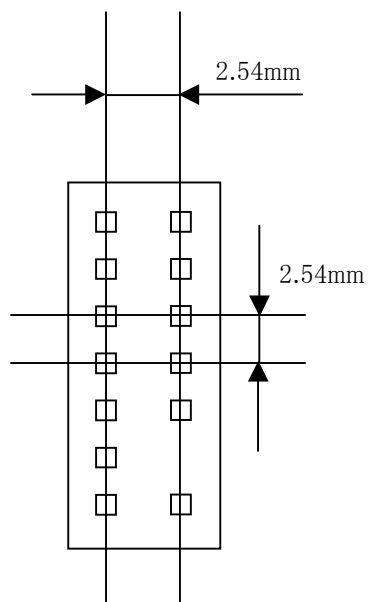
### ターゲットシステムの EJTAG コネクタ

ピンアサイン

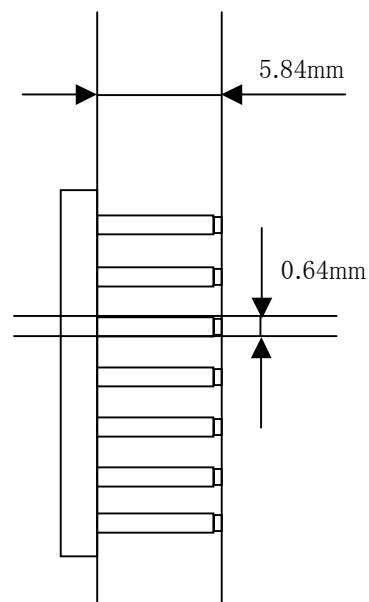
TRST	01	02	GND
TDI	03	04	GND
TDO	05	06	GND
TMS	07	08	GND
TCK	09	10	GND
RST	11	12	key
DINT	13	14	VIO

TOP VIEW

寸法図



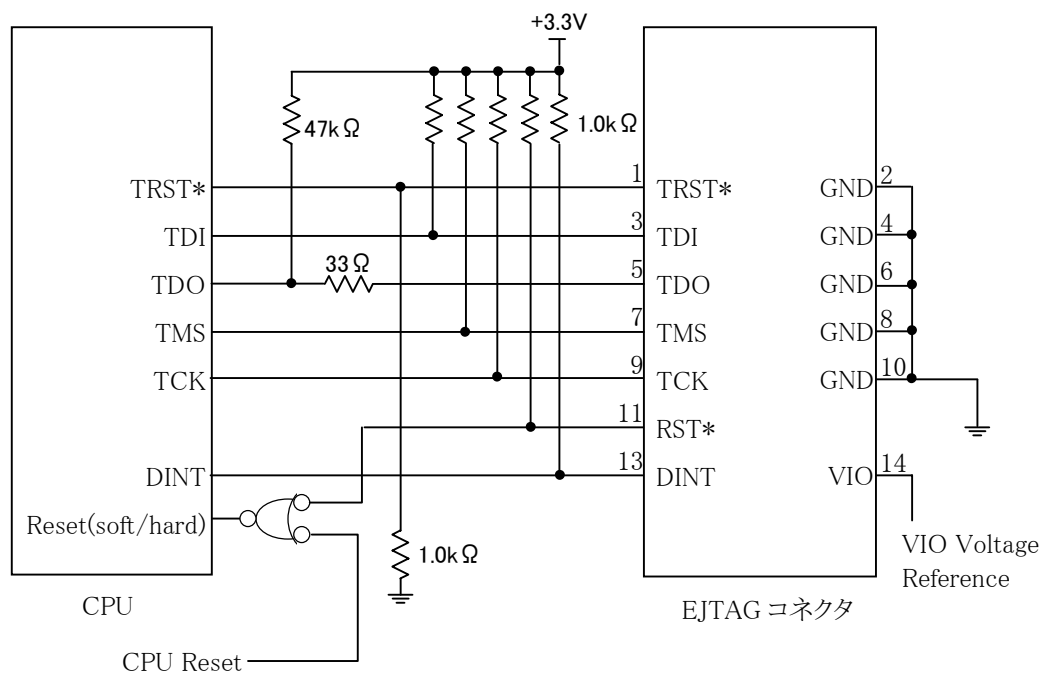
正面図



側面図



ターゲット システムの EJTAG コネクタ推奨回路



注意事項

- CPU～JTAG コネクタ間のパターンは、できるだけ短くしてください。(50mm 以下)
- TCK 信号のパターンは、GND でシールドしてください。
- MJX330 は、VIO 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。

## 付録 D レジスタ名一覧

レジスタ名はレジスタ定義ファイルによってユーザ固有に定義できます。サンプルのレジスタ定義ファイルでは以下のとおりです。

代表名      :   デフォルトのレジスタ名称<sup>\*1</sup>

エイリアス名:   ユーザ固有で付けられるレジスタの別名称

(汎用レジスタ)

代表名	エイリアス名	代表名	エイリアス名
R0	GPR0	R21	GPR21
R1	GPR1	R22	GPR22
R2	GPR2	R23	GPR23
R3	GPR3	R24	GPR24
R4	GPR4	R25	GPR25
R5	GPR5	R26	GPR26
R6	GPR6	R27	GPR27
R7	GPR7	R28	GPR28
R8	GPR8	R29	GPR29
R9	GPR9	R30	GPR30
R10	GPR10	R31	GPR31
R11	GPR11	HI	
R12	GPR12	LO	
R13	GPR13	PC	
R14	GPR14		
R15	GPR15		
R16	GPR16		
R17	GPR17		
R18	GPR18		
R19	GPR19		
R20	GPR20		

---

<sup>\*1</sup> 代表名は、レジスタ表示コマンドで使います。代表名およびエイリアス名はレジスタ設定コマンドで使います。エイリアス名は最大3個まで指定できます。レジスタ名の重複検査は行われないので注意してください。

付録 D レジスタ名一覧

(コプロセッサ 0 レジスタ)

代表名	エイリアス名	代表名	エイリアス名
Index		TagLo	
Ramdom		DataLo	
EntryLo0		TagHi	
EntryLo1		DataHi	
Context		ErrorEPC	
PageMask			
PageGrain			
Wired			
HWREna			
BadVAddr			
Count			
EntryHi			
Compare			
Status			
IntCtl			
SRSCtl			
SRSTMap			
Cause			
EPC			
PRId			
EBase			
Config			
Config1			
Config2			
Config3			
LLAddr			
WatchLo			
WatchHi			
XContext			
PerfCntC[n]			
PerfCntD[n]			
ErrCtl			
CacheErr			

【注意】CPU により実装されていないレジスタも含まれています。

(浮動小数点レジスタ)

代表名	エイリアス名	代表名	エイリアス名
FGR0	F0	FIR	FCR0
FGR1	F1	FCCR	FCR25
FGR2	F2	FEXR	FCR26
FGR3	F3	FENR	FCR28
FGR4	F4	FCSR	FCR31
FGR5	F5		
FGR6	F6		
FGR7	F7		
FGR8	F8		
FGR9	F9		
FGR10	F10		
FGR11	F11		
FGR12	F12		
FGR13	F13		
FGR14	F14		
FGR15	F15		
FGR16	F16		
FGR17	F17		
FGR18	F18		
FGR19	F19		
FGR20	F20		
FGR21	F21		
FGR22	F22		
FGR23	F23		
FGR24	F24		
FGR25	F25		
FGR26	F26		
FGR27	F27		
FGR28	F28		
FGR29	F29		
FGR30	F30		
FGR31	F31		

## 付録 E 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)

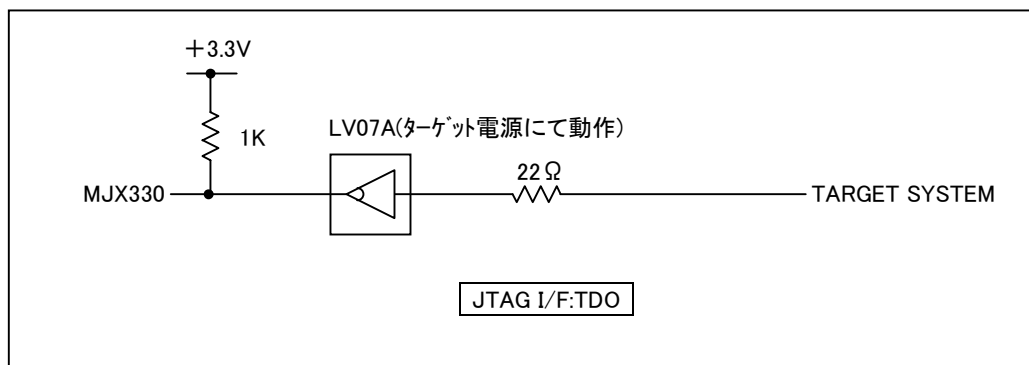
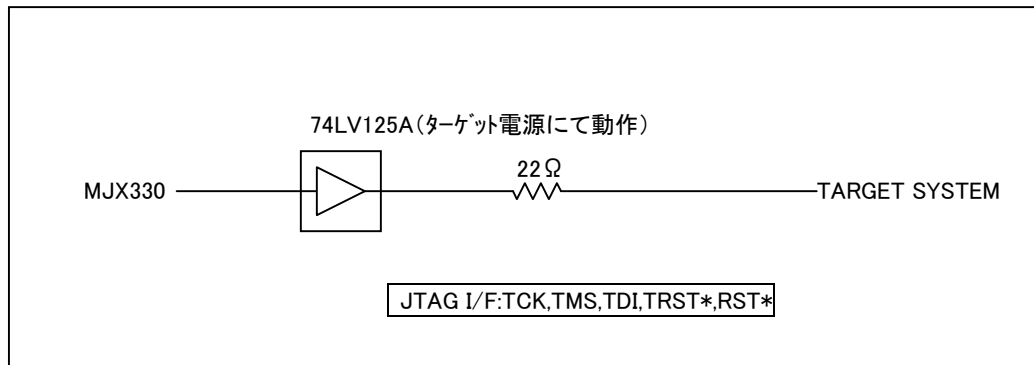
データ (可変、データ バイト長)

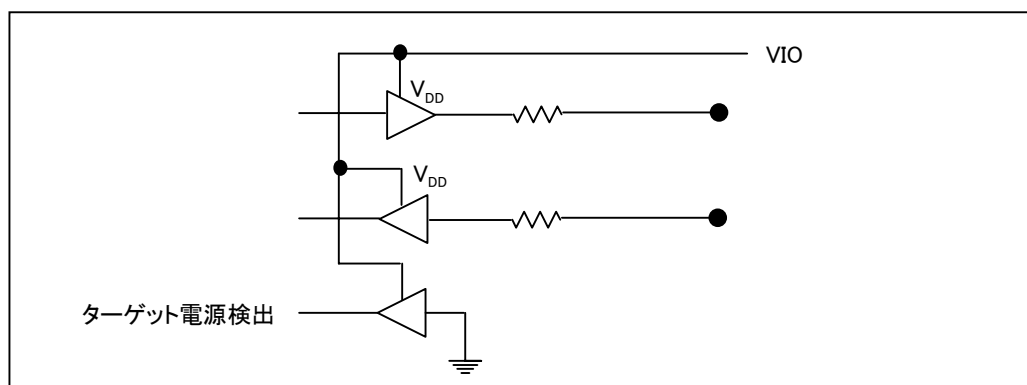
XX	XX	XX	XX	XX	XX	XX	...	XX	XX	XX	XX	XX	XX	XX	XX
----	----	----	----	----	----	----	-----	----	----	----	----	----	----	----	----

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 F ターゲット システムへのプロービング





## 付録 G 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(第7行以降)、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 <sup>(注 1)</sup>	チップ消去機能無フラグ <sup>(注 2)</sup>
2	デバイス コード <sup>(注 3)</sup>	
3	アクセス モード <sup>(注 4)</sup>	コマンド・インターフェース <sup>(注 5)</sup>
4	コマンド入力 第1バス サイクル 書き込み アドレス <sup>(注 6)</sup>	
5	コマンド入力 第2バス サイクル 書き込み アドレス <sup>(注 6)</sup>	
6	セクタ数 <sup>(注 7)</sup>	
7	第 1 ~ n セクタ サイズ <sup>(注 8)</sup>	セクタ数 <sup>(注 8)</sup>
	...	
	...	
-	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類を入力します。

0: JEDEC 標準型コマンド互換 (AMD タイプ)

1: インテル系の SPI (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 AM29F160B (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
書込み	16 ビット	555	AA	2AA	55	555	A0	PA	PD		—	—	—
	8 ビット	AAA		555		AAA							
チップ 消去	16 ビット	555	AA	2AA	55	555	80	555	AA	2AA	55	555	10
	8 ビット	AAA		555		AAA		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		AAA		555			

表の「書込み」シーケンスにしたがって、指定アドレスにデータを出力することで、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いたい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(555 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、その後に、0x10000 のセクタが 15 連続している構成の場合に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こと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 構成のみなので、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 H 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
AM29DL102CB	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
AM29LV010B	128Kx8	3V
AM29LV010D	512Kx8	3V
AM29LV081B	1Mx8	3V
AM29LV015D	2Mx8	3V
AM29LV013C	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	1Mx8	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
AT49BV2048	128Kx16	3V
AT49LV2048	128Kx16	3V
AT49BV2048A	256Kx8/128x16	3V
AT49LV2048A	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	5V
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
AT49BV161	2Mx8/1Mx16	3V
AT49BV161T	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
M29W160DT	16Mx8/8Mx16	3V
M29W160DB	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
MBM29DL000BC	512Kx8/256Kx16	3V
MBM29DL000TC	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	4Mx8/2Mx16	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
28F160F3T	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
LH28F160S3T-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
LH28F320BFHE-PTTL70	2Mx16	3V
LH28F320BFE-PTTL80	2Mx16	3V
LH28F320BFHE-PTTL80	2Mx16	3V
LH28F320BFE-PTTL80	2Mx16	3V
LH28F320BFHE-PTTL80	2Mx16	3V
LH28F320BFHE-PTTLZ1	2Mx16	3V

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
LH28F640BFB-PTTL90	4Mx16	3V
LH28F640BFN-PTTLZ2	4Mx16	3V
LH28F800BJE-PTTL90	1Mx8/512Kx16	3V
LH28F800BJB-PTTL10	1Mx8/512Kx16	3V



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-BTL90	1Mx8/512Kx16	3V
LH28F800BVN-BTL10	1Mx8/512Kx16	3V
LH28F800BVB-BTL90	1Mx8/512Kx16	3V
LH28F800BVB-BTL10	1Mx8/512Kx16	3V
LH28F800BVE-TV85	1Mx8/512Kx16	5V
LH28F800BVN-BV10	1Mx8/512Kx16	5V
LH28F800BVHB-BV85	1Mx8/512Kx16	5V

## 付録 I デバイス ドライバのトラブルシューティング

### I-1 PC カードを差すとホスト PC がハングアップする場合 (Windows 95/98/98SE)

一部のノート PC<sup>\*1</sup> では、ドライバのインストールのために PC カードを差すと、ハングアップしてしまうことがあります。

そのような場合は一旦強制的に電源をオフし、電源オフの状態で PC カードを差してから、電源をオンしてください。

その後、「2.1 デバイス ドライバのインストール手順 [2.1.1 Window98/98SE の場合](#)」にしたがってドライバをインストールしてください。

---

<sup>\*1</sup> SONY VAIO PCG-505 でこの現象を確認しています

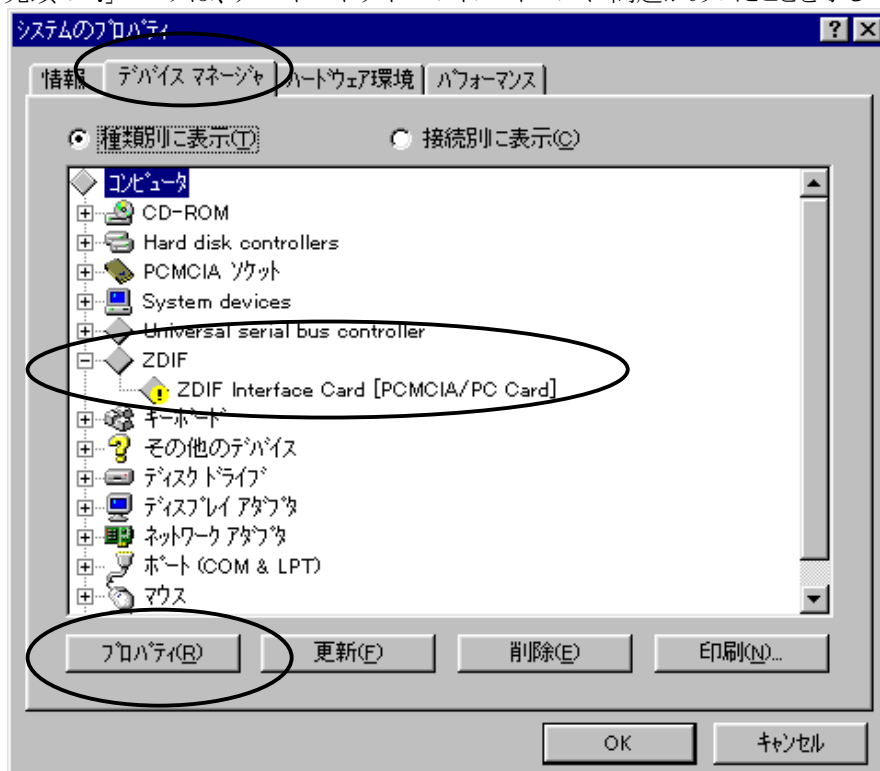
## I-2 リソースの競合、またはリソースの空きがない場合 (Windows 98/98SE)

PC カードのデバイス ドライバのインストール終了時に「ビポッ」ではなく「ブー」という音がした場合は、ドライバのインストールに失敗しています。

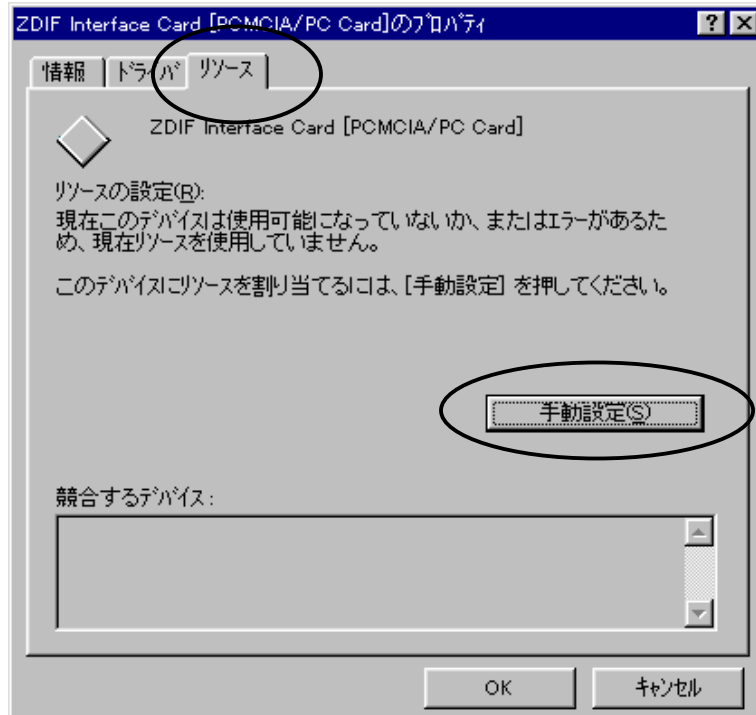
まずコントロール パネルから「システム」をダブル クリックして開き、「デバイス マネージャ」タブをクリックします。

次に「ZDIF」クラスをクリックして、その下にある「ZDIF Interface Card [PCMCIA/PC Card]」のプロパティを表示します(ダブル クリックするか、「プロパティ(R)」をクリック)。

先頭の「!」マークは、デバイス ドライバのインストールに問題があったことを示しています。



「リソース」タブをクリックすると、以下のように表示されますので、「**手動設定(S)**」をクリックします。



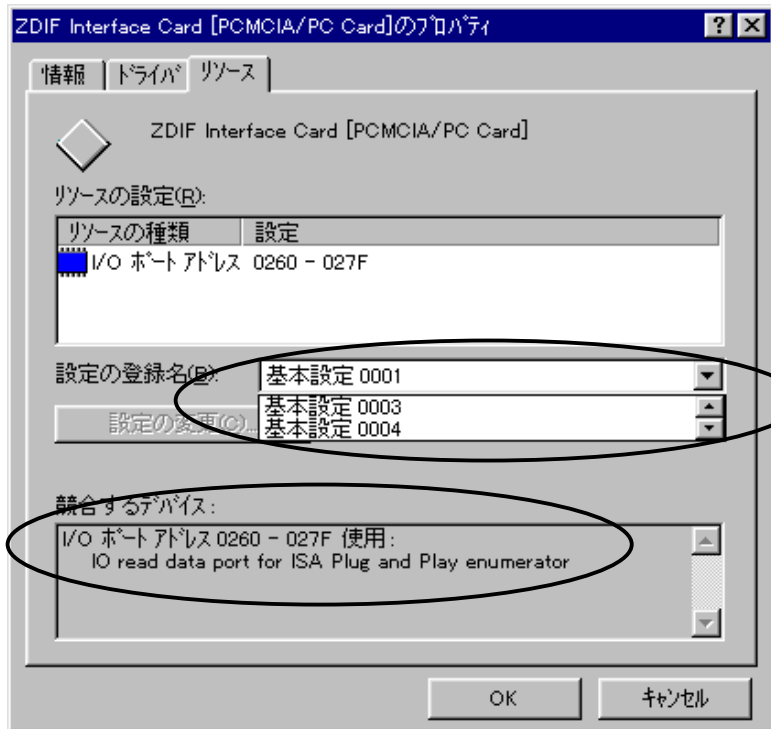
「自動設定(U)」がオフ(チェック オフ)になっていること、「競合するデバイス」が「競合はありません。」ではなく他デバイスがリソースを使用(競合)していることを確認します。



「設定の登録名(B)」で「基本設定 000」から「基本設定 004」までを順に選択してみます。

「競合するデバイス」に「競合はありません。」と表示されるものがあればこれを使用します。

その場合、Plug & Play 機構による自動設定ではなく、手動による設定を行ったこととなりますので、ワーニングが表示されますが、これは無視して使用します。



「基本設定 000」から「基本設定 004」までの全ての設定で競合が発生している場合は、競合している他デバイスのリソースを変更して空にしない限り、PC カードを使用することはできません。

PC カードが使用するリソース(I/O ポート)は以下のうちのどれか一つです。

0220 - 023F = 基本設定 000

0260 - 027F = 基本設定 001

02E0 - 02FF = 基本設定 002

0320 - 033F = 基本設定 003

03E0 - 03FF = 基本設定 004

競合している他デバイスのリソース(I/O ポート)も変更できない場合は、PC カードを使用することはできません。

### I-3 割り込みを使用しない PC カードを受け付けないホスト PC の場合 (Windows 98/98SE)

PC カードのデバイス ドライバのインストール終了時に「ブー」という音がしてインストールに失敗したものの、「リソースの競合、またはリソースの空きがない場合」のパターンには当てはまらない（リソースの競合がない、またはドライバがデバイス マネージャに登録されていない等）場合は、以下の手順を試してみてください。

一部のノート PC<sup>\*1</sup> では、割り込みを使用しない PC カードを正しく認識できないことがあります。  
この場合は一旦、ドライバのアンインストールを行った後、”擬似的に割り込みを使用するように設定したインストール”を行います。

まず、「[2.2 デバイス ドライバのアンインストール手順](#)」を参考にして、デバイス ドライバを削除します。  
INF ファイルも削除してください。

一旦ノート PC をシャットダウンして電源オフし、PC カードを抜きます。

次に、「[2.1 デバイス ドライバのインストール手順](#)」にしたがってドライバを再インストールしますがその際、ドライバ ファイルのあるディレクトリには

D:¥Driver¥Win9x¥ThinkPad （CD-ROMドライブが D の場合）

を指定してください。

このディレクトリに含まれるドライバ ファイルは「D:¥Driver¥Win9x」と同じですが、PC カードが割り込みを使用するように擬似的に設定する INF ファイルが入っています。

なお、このインストールが正しく行われるためには、割り込みが最低 1 つは空いている必要があります。

---

<sup>\*1</sup> IBM ThinkPad シリーズの一部旧機種でこの現象を確認しています。

## I-4 ハードウェア ウィザードが起動せず、デバイス ドライバをインストールできない場合(Windows 98/98SE)

ノート PC に PC カードを差しても全く認識されず、ハードウェア ウィザードも起動しない場合があります

\*1。

この場合は PC カードを使用することはできません。

詳細は弊社サポートまでお問い合わせください。

## I-5 リソースの空きがない場合(WindowsNT4.0)

PC カードは以下のリソース(I/O ポート)のうち、いずれか一つ空いているものを使用します。

0220 – 023F

0260 – 027F

02E0 – 02FF

0320 – 033F

03E0 – 03FF

これらのリソースが全て他デバイスで使用されていて空きがない場合は、競合している他デバイスのリソースを変更して空きにしない限り、PC カードを使用することはできません。

---

\*1 IBM ThinkPad 380ED 2635 7AJ でこの問題が報告されています。

## 検索項目

### [MJX330 について](#)

[MJX330 のおもな特長](#)

[構成](#)

[MJX330 を使用する前の準備作業](#)

### [MULTI について](#)

### [MJXDEBW について](#)

[デバイス ドライバのインストール](#)

[ハードウェアの接続](#)

[MJX330 とホストの接続](#)

[MJX330 とターゲットの接続](#)

### [MJX330 を操作するソフトウェアのインストール](#)

[MJX330 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](#)

### [MJX330 の環境設定](#)

[コンフィグ支援ツール MJXCFG](#)

### [MJX330 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](#)

### [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メニューコマンドの一覧](#)

### [高速ダウンロード](#)

[次ページ](#)



[仕様](#)

[本体寸法](#)

[本体重量](#)

[使用温度範囲](#)

[保存温度範囲](#)

[周囲湿度範囲](#)

[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](#)

[制限事項](#)

[ターゲット システムの制限事項](#)

[EJTAG コネクタ](#)

[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](#)

---

**MJX330 について**

< 関連項目 >

- ・[MJX330 のおもな特長](#)
- ・[構成](#)
- ・[MJX330 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 >

- [MJX330 とホストの接続](#)
- [MJX330 とターゲットの接続](#)

[検索項目の先頭ページ](#)

---

---

**MJX330 を操作するソフトウェアのインストール**

< 関連項目 >

- [MJX330 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](#)

---

---

**MJX330 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](#)

---

---

## 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](#)

---