

MJX440 for V831/V832 User's Manual

Rev.1.00 07/05/1999

Notes

- The copying of this manual, either wholly or in part, without the consent of Lightwell Corporation is prohibited.
- Lightwell disclaims any responsibility for any effects resulting from the use of this product.
- The specifications for this product and the contents of this manual are subject to change without notice.
- MS-DOS, Windows are registered trademarks of Microsoft.
- MULTI is a registered trademark of Green Hills Software Corporation.

©1999, Lightwell Corporation. All rights reserved.

Printed in Japan

Address: 5-20-12, Ogikubo, Suginami-ku, Tokyo, 167-0051, Japan

Telephone: 03-3392-3331

Fax: 03-3393-3878

E-mail: ZAXSupport@lightwell.co.jp

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

July 1999

Congratulations on your recent purchase of MJX440 for V831/V832("MJX440").

This manual is composed of the following contents:

Chapter 1. Overview

This chapter describes the product configuration, provides an overview of MJX440, and explains the nomenclature of the various components.

Chapter 2. Setting the Parallel Interface

This chapter describes how to mount the parallel interface (a PCI card or a PCMCIA card) and how to install a device driver for it.

Chapter 3. Connecting the Hardware

This chapter describes how to connect the MJX440 to a host as well as how to connect the MJX440 to a target system.

Chapter 4. Installing the Software

This chapter describes how to install the software for the operation of MJX440.

Chapter 5. Setting the Environment for MJX440

This chapter describes how to set the environment before using the MJX440.

Chapter 6. Starting and Ending Software

This chapter describes how to start and end the software for the operation of the MJX440.

Chapter 7. MJX440 Commands

This chapter describes how to use the various MJX440 commands.

Chapter 8. Rapid Downloading

This chapter describes the procedure for effecting rapid downloading.

Appendixes

The Appendixes provide additional technical information, such as specifications and restrictions that are applicable to the target system.

CONTENTS

Chapter 1. Overview	5
1.1 Product Makeup	5
1.2 Overview of MJX440	8
1.3 Nomenclature.....	12
Chapter 2. Setting the Parallel Interface.....	14
2.1 Installing the Parallel Interface	14
2.2 Installing Device Drivers.....	15
2.3 Removing/Uninstalling a Device Driver	45
Chapter 3. Connecting the Hardware.....	64
3.1 Connecting the MJX440 to the Host	64
3.2 Connecting an N-Wire Cable.....	65
3.3 Connecting ROM Probes	66
3.4 Connecting an External Trigger Cable.....	73
3.5 Connecting the Power Supply and Turning the Power On.....	75
Chapter 4. Installing the Software	76
Chapter 5. Setting the MJX440 Environment	85
Chapter 6. Starting and Terminating Software.....	87
Chapter 7. MJX440 Commands	89
Chapter 8. Rapid Downloading	118
Appendix A. Specifications	119
Appendix B. Limits on Target Systems.....	120
Appendix C. N-Wire Connector	121
Appendix D. ROM Probes	123
Appendix E. Corresponding ROM Pin Assignment.....	126
Appendix F. LEDs.....	130
Appendix G. List of Registers.....	131
Appendix H. MJX Binary File	132
Appendix I. Probing of the Target System	133
Appendix J. Troubleshooting.....	139

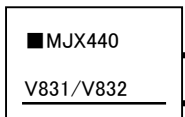
Chapter 1. Overview

This chapter describes the product configuration, provides an overview of MJX440, and explains the nomenclature of the various components.

1.1 Product Makeup

When shipped, each MJX440 for V831/V832 package contains the items indicated below. If you find any items missing, please contact Lightwell Corporation.

- MJX440 for V831/V832 system unit



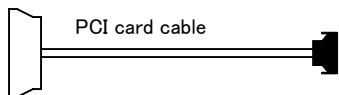
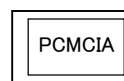
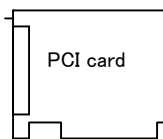
- AC adapter



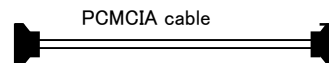
- AC cord



- Parallel interface (PCI card or PCMCIA card) and a parallel interface cable *¹



or



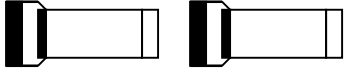
- N-Wire cable



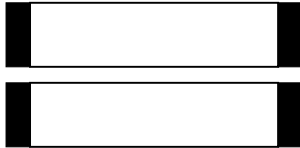
*¹ When shipped, the package contains a cord for either a PCI card or PCMCIA card, but not both.

Chapter 1. Overview

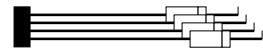
- ROM probe *1



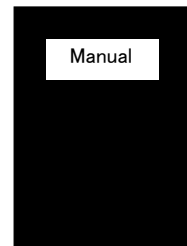
- Two ROM probe cables



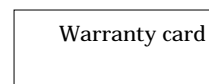
- External trigger cable



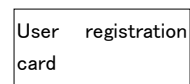
- User's manual



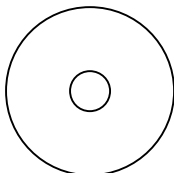
- Warranty card



- User registration card



- CD-ROM



*1 The type of a ROM probe included in the package depends on the type of ROM used in the system. For details, see "Table 1-1 Factory-shipped ROM probes" on the following page.

Chapter 1. Overview

Following is a list of ROM probes that are included in the initial package, which depend on the type of ROM used in the product:

ROM type	ROM probe	
	ROM probe board	ROM plug
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

Table 1-1 Factory-shipped ROM probes

1.2 Overview of MJX440

MJX440 is an N-Wire connector equipped developmental aid device for the debugging of a V831 or V832 target system.

MJX440: Its Principal Features

- Because it uses the target system's N-Wire connector, MJX440 allows ready connection to the target system.
- MJX440 operates stably even when working with a high-speed CPU.
- MJX440 permits any in-circuit connection to the ROM and can debug programs running on ROM.
- MJX440 can be used in conjunction with the high-level language Debugger MULTI.
- The use of MJX440 commands permits the full use of the hardware on which MJX440 is based.
- MJX440 supports a real-time tracing function.
- MJX440 supports a hardware breakpoint function.
- MJX440 permits the rapid downloading of programs (440 kbytes/sec for N-Wire connection; 4Mbytes/sec for ROM in-circuit connection)

Minimum configuration

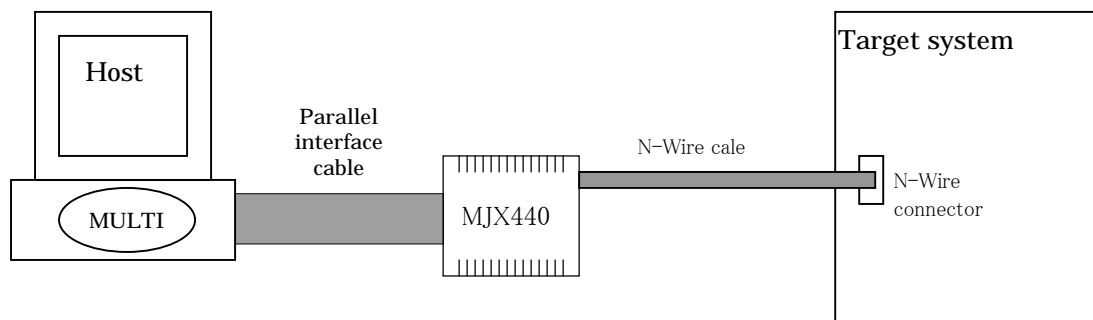


Figure 1-1. Minimum configuration

MJX440 operates in the minimum configuration shown in Figure 1-1. When used in its minimum configuration, MJX440 can debug programs that are stored in the RAM of the target system. In this operating mode, the program to be debugged is downloaded onto the RAM of the target system through the N-Wire cable.

In the minimum configuration, MJX440 can execute programs stored in the ROM, but it cannot download

Chapter 1. Overview

programs onto the ROM region or set software breakpoints. Debugging a ROM program requires a ROM in-circuit connection of MJX440.

ROM in-circuit connection

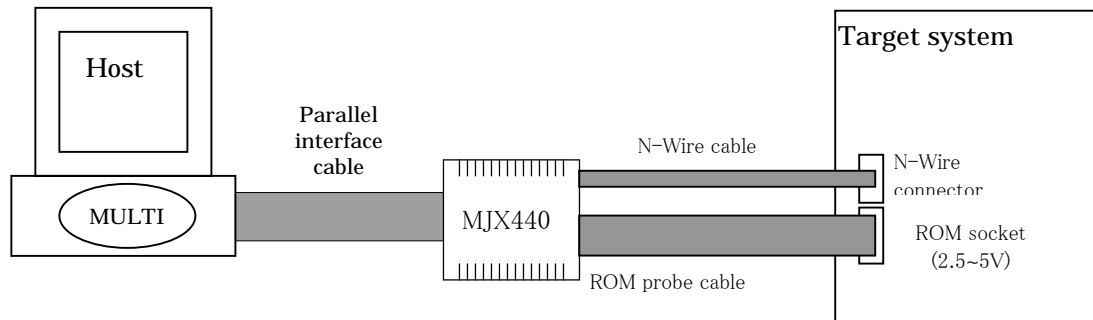


Figure 1-2. ROM in-circuit connection

Under a ROM in-circuit connection as shown in Figure 1-2, MJX440 can also debug programs that are stored in ROM. In this operating mode, the program to be debugged and stored in ROM is downloaded onto the emulation memory in the MJX440 rather than onto the RAM for the target system.

In this case, the ROM voltage is automatically recognized within the 2.5 ~ 5V range.

External trigger cable connection

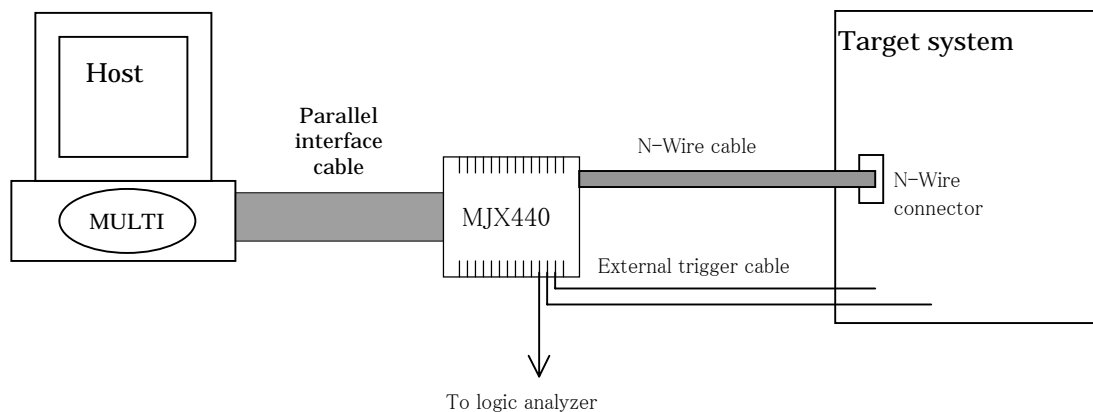


Figure 1-3. External trigger cable connection

Chapter 1. Overview

As shown in Figure 1-3, an external trigger cable can be used to implement the following functionality:

- Displaying target system signal status information on the LED on the MJX440 (input)
- Storing target system signal status information in the realtime trace memory (input)
- Using the trace trigger as a logic analyzer trigger signal (output)
- Allowing the use of genetic signals, eligible for MJX440 command operation, on the target system (output)

Preliminary to using the MJX440

Before using the MJX440, you need to perform the preliminary steps described below, which can be performed by referring to Chapters 2 through 5. These steps need to be performed only once after the MJX440 is purchased.

- Setting the parallel interface
- Connecting the hardware
- Installing the software
- Setting the environment for the MJX440

Once these preliminary steps are completed, refer to Chapter 6 to start the software (MULTI or MJXDEB) for operating the MJX440. Normal startup of the software indicates that the preliminary steps were successfully completed. If the software fails to start properly, check to see whether there was an error in performing the preliminary steps.

For a description of how to use the compiler or MULTI, see the respective manuals and release notes. An explanation of MJX440 commands for using functions specific to MJX440 is given in Chapter 7.

Refer to Chapter 8 for a description of how to effect a rapid downloading.

Chapter 1. Overview

About MULTI

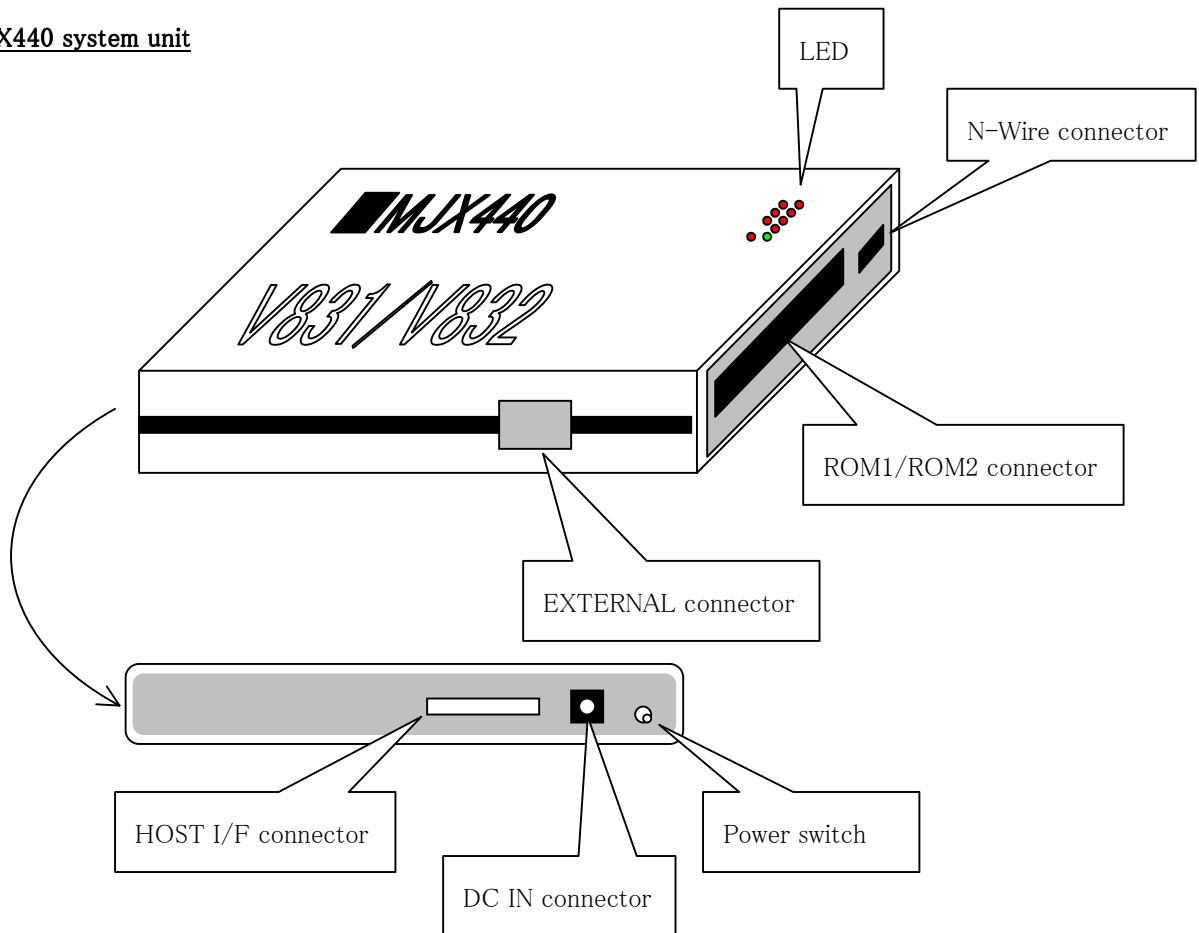
MULTI is a high-level language debugger that can be executed under various environments. You can run MULTI under the MJX440 environment by calling the server program MJXSERV.

About MJXDEB

MJXDEB refers to a quick debugger that supports MJX440 commands only. MJXDEB may be a suitable choice when no high-level language debugging is needed or when the target system is to be checked by using the batch processing function.

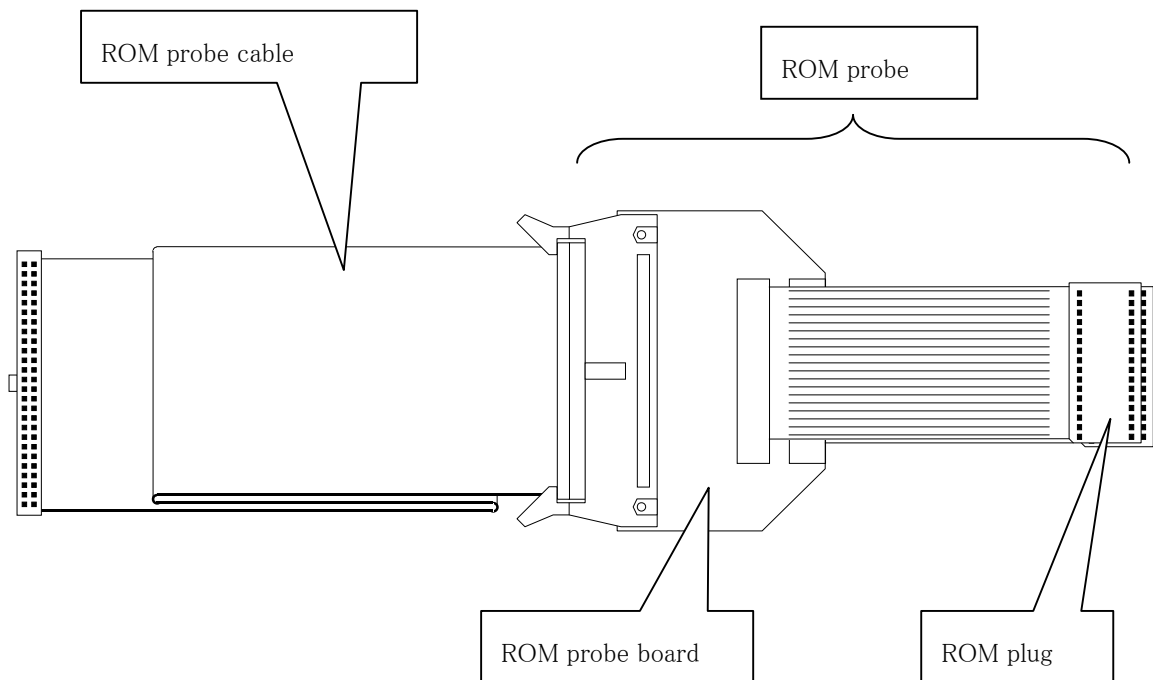
1.3 Nomenclature

MJX440 system unit



Power switch	This switch turn the power on and off.
DC IN connector	Connects the AC adapter.
HOST I/F connector	Connects the parallel interface cable.
N-Wire connector	Connects the N-Wire cable.
ROM1/ROM2connector	Connects the ROM probe cable. The top part is a ROM2 connector, the bottom part the ROM2 connector.
EXTERNALconnector	Connects the external trigger cable.
LED	Displays power and connector plug-in status information.

ROM probe



ROM probe cable

This cable connects the MJX440 to the ROM probe.

ROM probe

This refers to all probes that are connected to the target system ROM.

ROM probe board

This is the circuit board part of the ROM probe (with jumper settings).

ROM plug

This is a unit by which a device is connected to the ROM socket of the target system.

Chapter 2. Setting the Parallel Interface

This chapter describes how to install the parallel interface (a PCI or PCMCIA card) as well as how to install the device driver for it.

2.1 Installing the Parallel Interface

PCI card

Install the PCI card in the PCI expansion slot of the host system by performing the procedures indicated below (refer to the host system manual for details):

1. Close all software on the host system; turn off the power for the host system.
2. Remove the cover from the host system unit.
3. Remove the cover from the expansion slot.
4. Install the PC card in the expansion slot.
5. Replace the system unit cover.
6. Go to Section 2.2, "Installing Device Drivers".

PCMCIA card

Install the PCMCIA card in the PCMCIA expansion slot of the host system by performing the procedures indicated below (refer to the host system manual for details):

1. Close all software on the host system; turn off the power for the host system. ^{*1}
2. Insert the PCMCIA card into the slot.
3. Go to Section 2.2, "Installing Device Drivers".

^{*1} If your system is running under Windows 95/Windows 98, you can insert the PCMCIA card while the power is still on, in which case the first step of turning off the power for the host system in Section 2.2, "Installing Device Drivers", can be skipped.

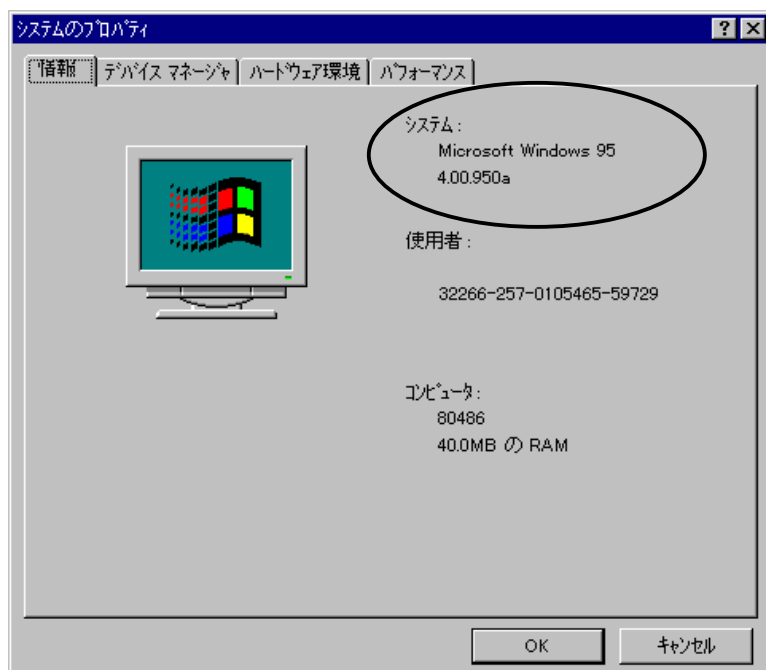
2.2 Installing Device Drivers

The installation of device drivers is divided into the following sections: *1

- Device drive installation procedure (PCMCIAcard, Windows 98)
- Device drive installation procedure (PCCard, Windows 98)
- Device drive installation procedure (PCMCIAcard, Windows 95 OSR2)
- Device drive installation procedure (PCCard, Windows 95 OSR2)
- Device drive installation procedure (PCMCIAcard, Windows 95 OSR1)
- Device drive installation procedure (PCCard, Windows 95 OSR1)
- Device drive installation procedure (PCMCIAcard, Windows NT4.0)
- Device drive installation procedure (PCCard, Windows NT4.0)

Versions of Windows 95 required different installation procedures.

Check the version of your Windows by double-clicking on **System** in the **Control Panel** (see the next page).



*1 MJX440 does not support Windows 3.1, 3.11, Windows NT3.5, or 3.51.

Chapter 2. Setting the Parallel Interface

In this manual, the following versions are referred to as “**Windows 95 OSR1**”:

Microsoft Windows 95 4.00.950

Microsoft Windows 95 4.00.950a

Similarly, the following version is referred to as “**Windows 95 OSR2**”:

Microsoft Windows 95 4.00.950b

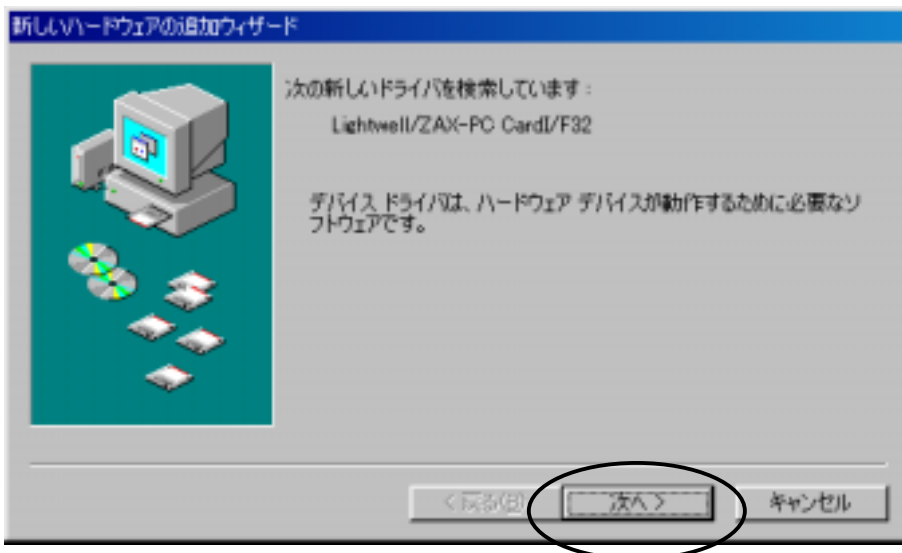
Please refer to the section of the manual that matches your environment.

Note that the device driver for the MJX440 is named “**ZDIF Interface Card Driver**”.

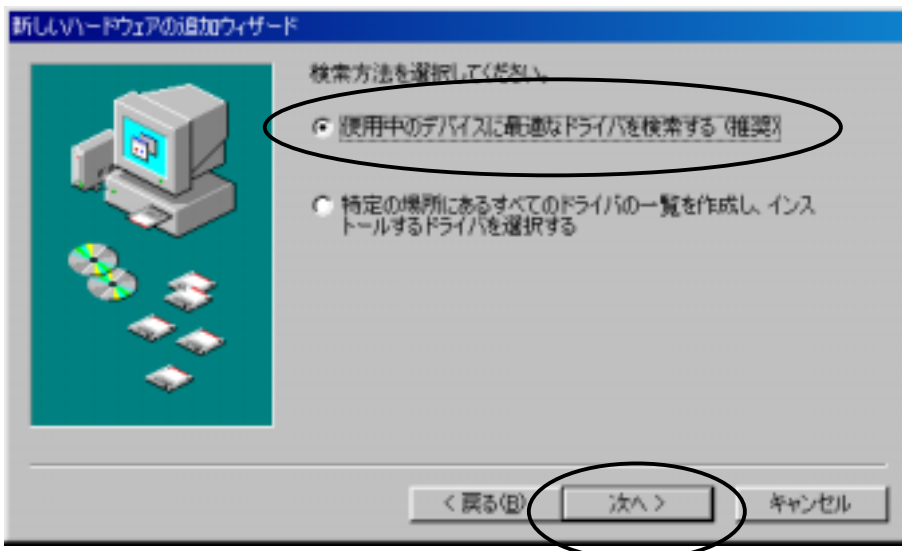
Chapter 2. Setting the Parallel Interface

Device driver installation procedure (PCMCIA card, Windows 98)

When the PCMCIA card is inserted with the power for the notebook PC turned on, the Add New Hardware Wizard detects the card and displays the following dialog, at which point you should click on “Next>”:



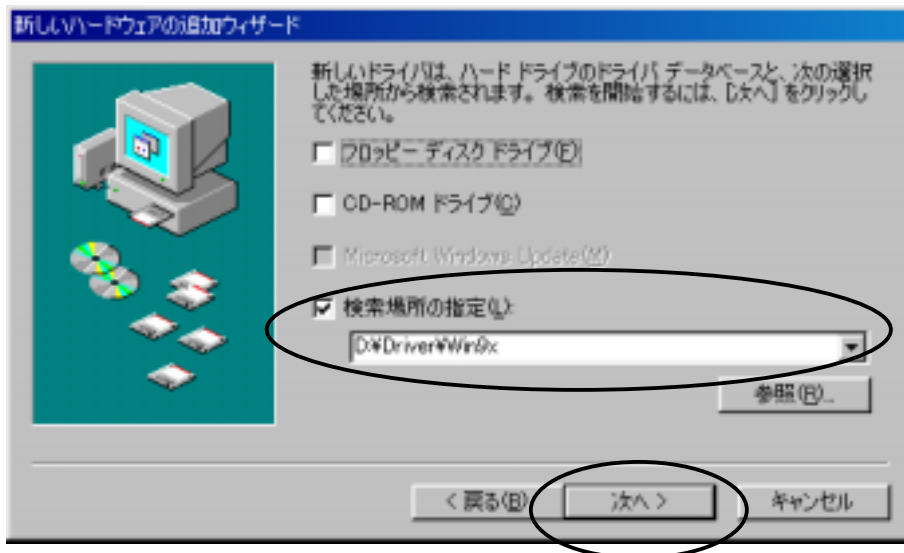
Select “Detect the appropriate driver for your device (recommended)” and then click on “Next>”.



Chapter 2. Setting the Parallel Interface

Load the CD-ROM labeled “*MJX440 for V831/V832 Tools Disk*” in the CD-ROM drive. Select “**Browse (L)**” to specify the directory where the driver is located, e.g., “**D:\Driver\Win9x**” if the CD-ROM drive is drive D: (depending on your specific PC).

Click on “**Next>**”.



If the directory contains the drive file, the following dialog appears, at which point click on “**Next>**”:

If a message “**A drive for this device was not found**” message appears, click on “**<Back (B)**” to specify the correct directory where the drive file is located.

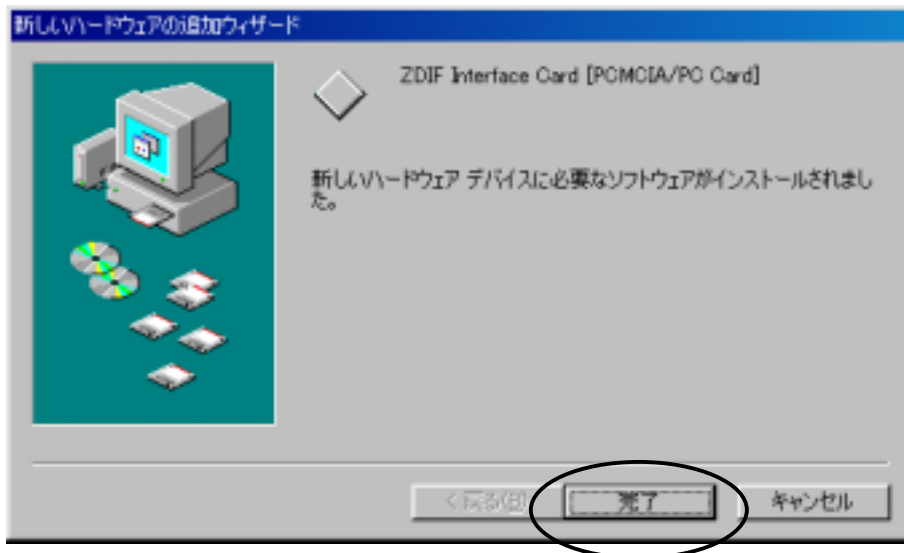


Chapter 2. Setting the Parallel Interface

The driver is automatically installed; the dialog shown below appear.

A “ping” sound goes off when the drive is correctly installed, at which point click on “**Finish**”.

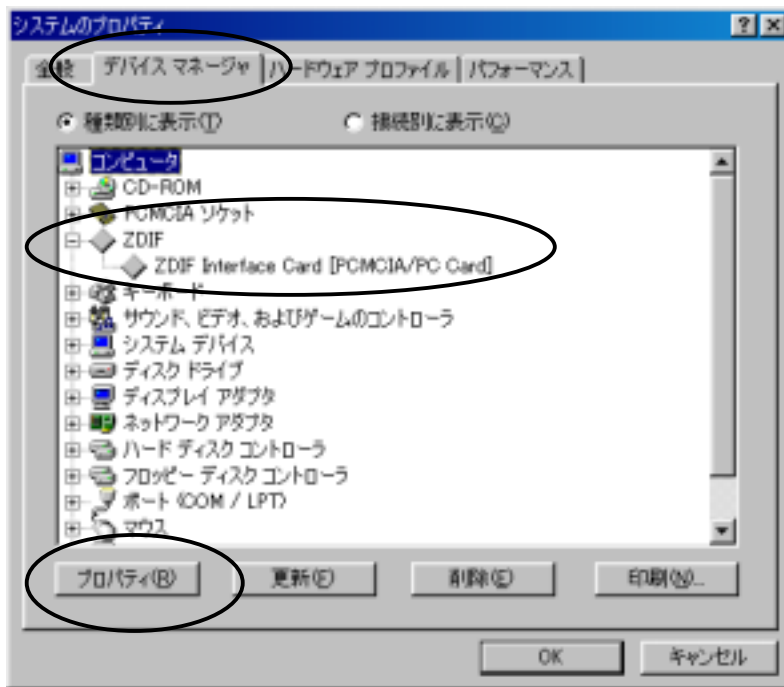
A buzzer sound indicates an installation failure. If this happens, close the installation process and refer to Appendix J, “Troubleshooting”, to resolve the problem.



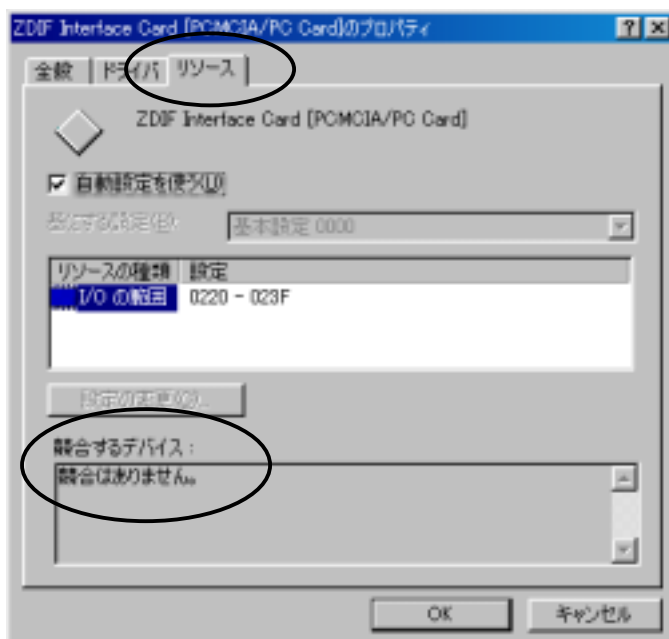
Chapter 2. Setting the Parallel Interface

To confirm that the device driver was installed correctly, double-click and open **System** in the **Control Panel**, and click on the **Device Manager** tab.

Click on “ZDIF” class to display the properties of the “ZDIF Interface Card [PCMCIA/PC Card]” below (either double-click or click on “Properties (R)” .



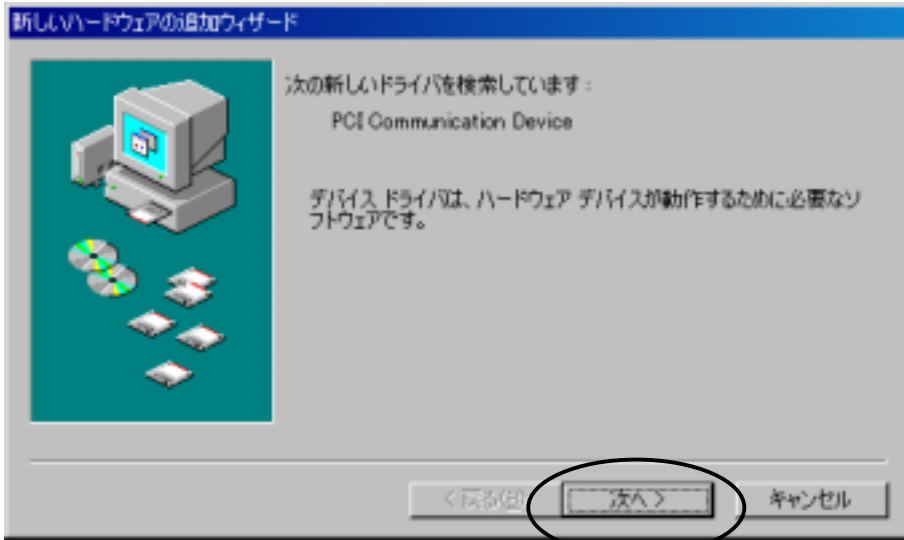
Click on “Resources” tab to make sure that the “Conflicting devices” item indicates “No conflict”.



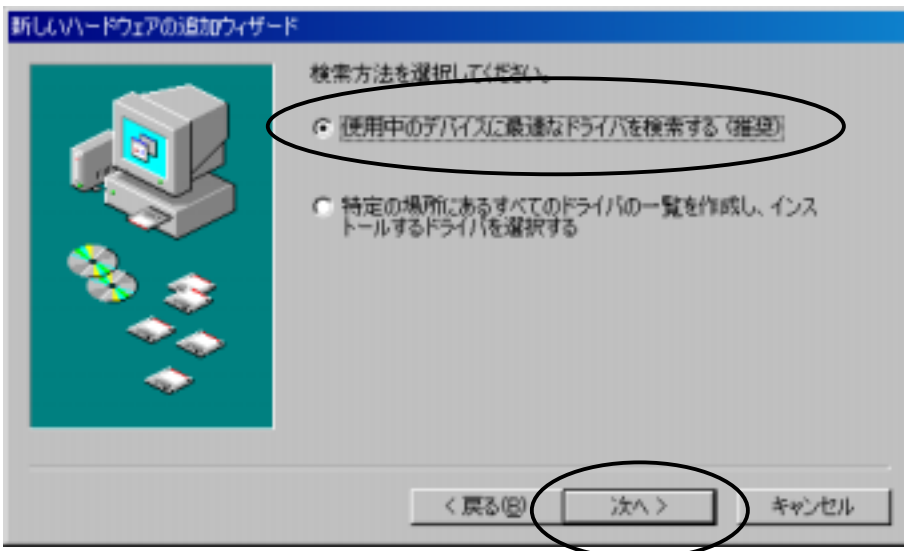
Chapter 2. Setting the Parallel Interface

Device driver installation procedure (PCI card, Windows 98)

When the PCI card is inserted and the power for the host system is turned on, the Add New Hardware Wizard detects the card and displays the following dialog, at which point you should click on “Next>”:



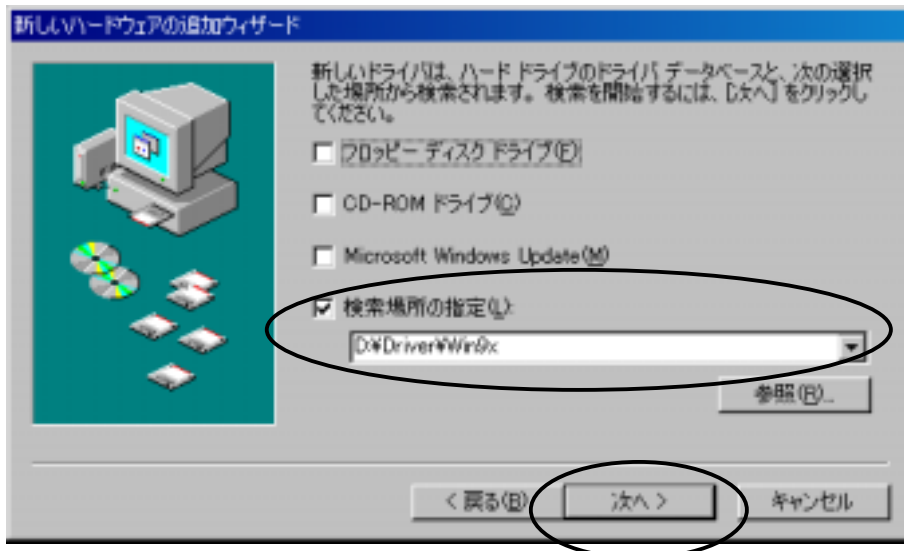
Select “Detect the appropriate driver for your device (recommended)” and then click on “Next>”.



Chapter 2. Setting the Parallel Interface

Load the CD-ROM labeled “*MJX440 for V831/V832 Tools Disk*” in the CD-ROM drive. Select “**Browse (L)**” to specify the directory where the driver is located, e.g., “**D:\Driver\Win9x**” if the CD-ROM drive is drive D: (depending on your specific PC).

Click on “**Next>**”.



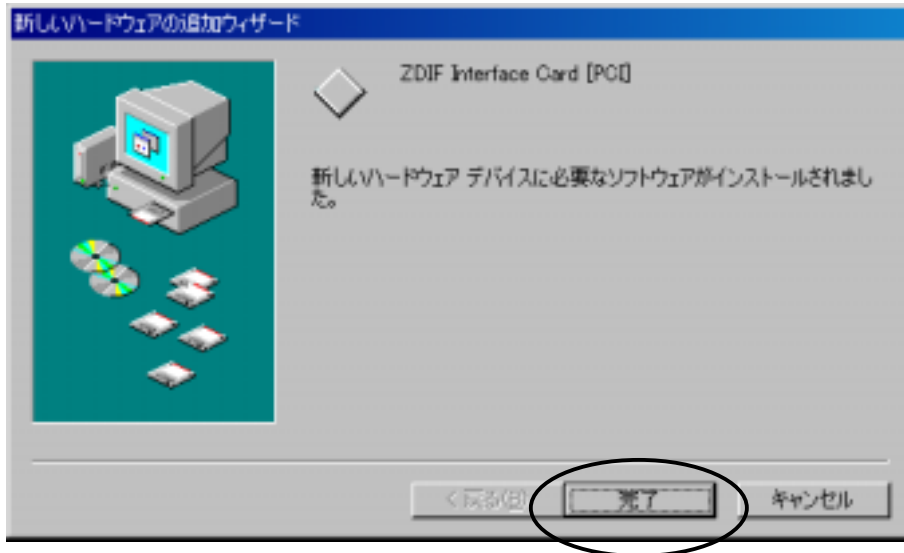
If the directory contains the drive file, the following dialog appears, at which point click on “**Next>**”:

If a message “**A drive for this device was not found**” message appears, click on “**<Back (B)**” to specify the correct directory where the drive file is located.



Chapter 2. Setting the Parallel Interface

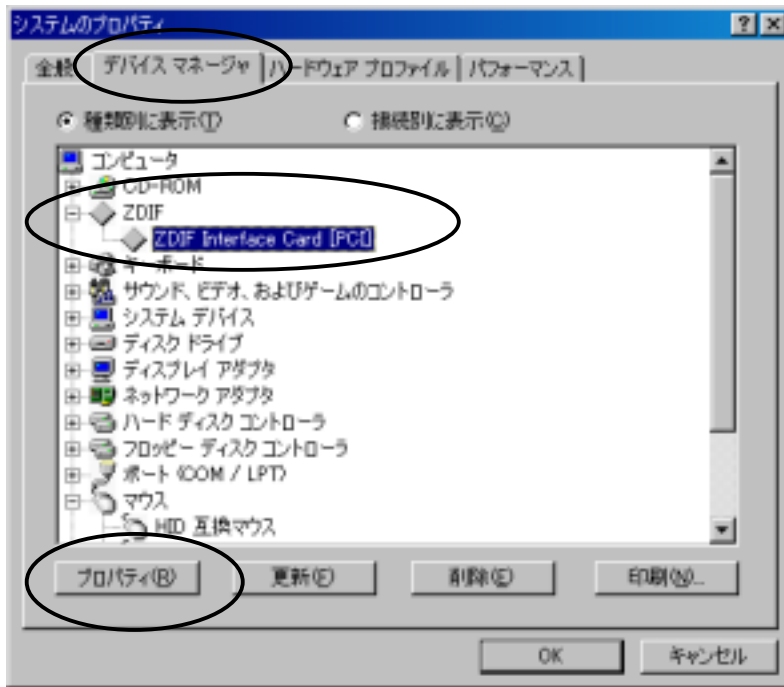
The driver is automatically installed. When the following dialog appears, end the process by clicking on **Finish**:



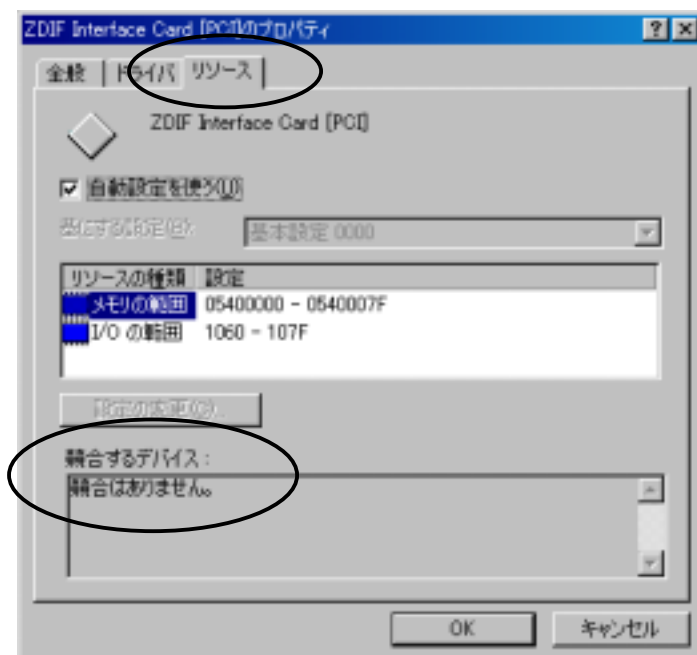
Chapter 2. Setting the Parallel Interface

To confirm that the device driver was installed correctly, double-click and open **System** in the **Control Panel**, and click on the **Device Manager** tab.

Click on “ZDIF” class to display the properties of the “ZDIF Interface Card [PCMCIA/PC Card]” below (either double-click or click on “Properties (R)”).



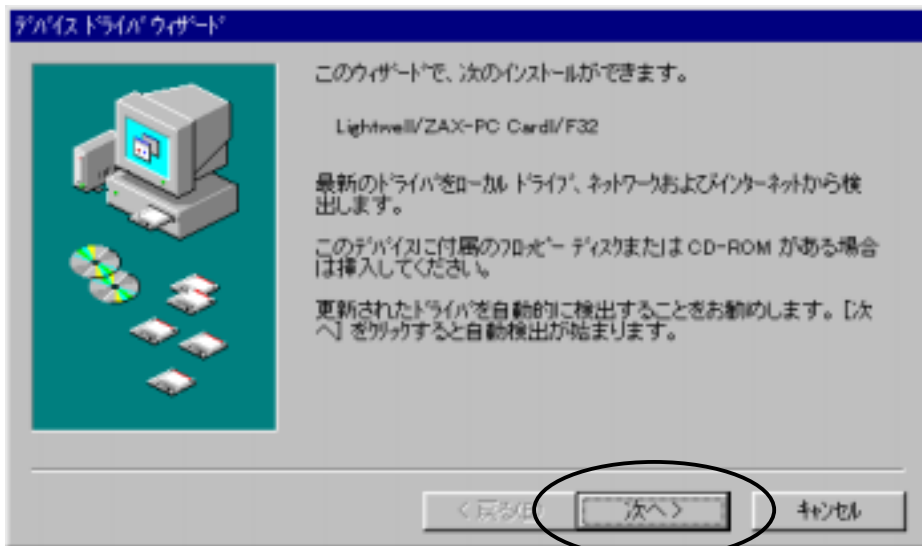
Click on “Resources” tab to make sure that the “Conflicting devices” item indicates “No conflict”.



Chapter 2. Setting the Parallel Interface

Device driver installation procedure (PCMCIA card, Windows 95 OSR2)

When the PCMCIA card is inserted with the power for the notebook PC turned on, the Add New Hardware Wizard detects the card and displays the following dialog, at which point you should click on “Next>”:



The system automatically searches the A: drive for the device driver. Failing to find it, the system displays the dialog shown below.

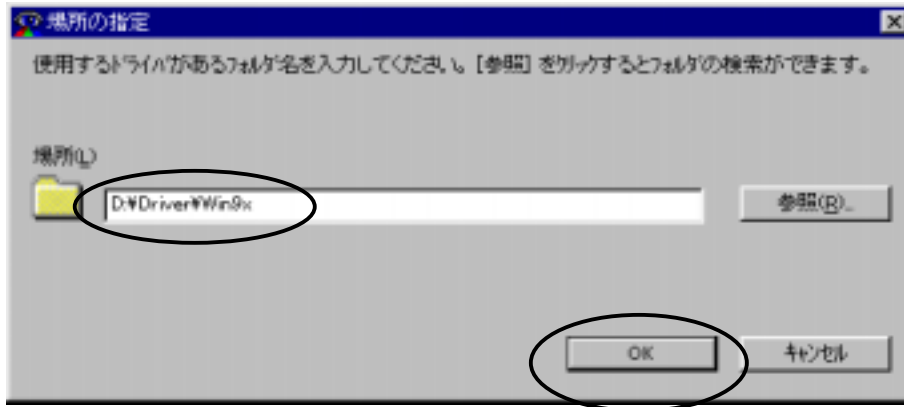
Click on “Browse (O) ...”.



Chapter 2. Setting the Parallel Interface

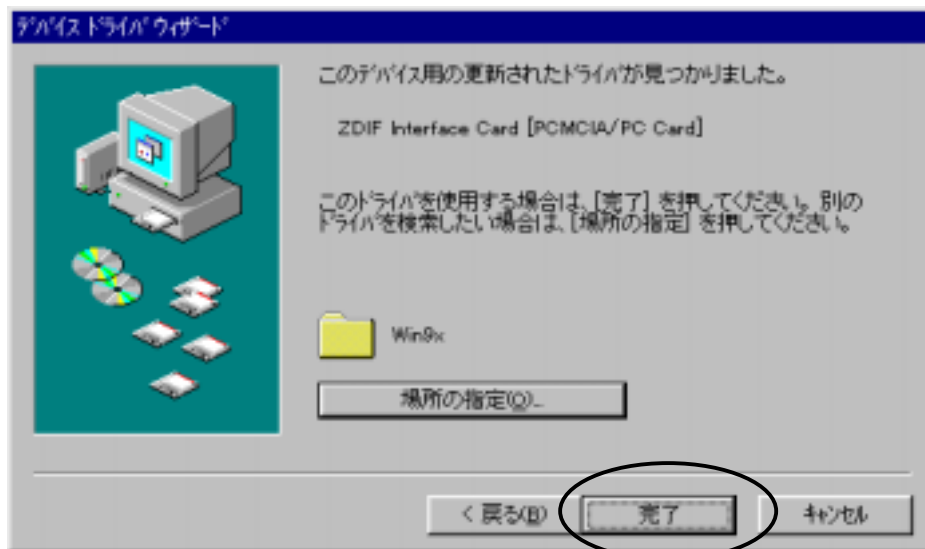
Load the CD-ROM labeled “*MJX440 for V831/V832 Tools Disk*” in the CD-ROM drive. Select “**Browse (L)**” to specify the directory where the driver is located, e.g., “**D:\Driver\Win9x**” if the CD-ROM drive is drive D: (depending on your specific PC).

Click on “**OK**”.



The driver file is detected; the dialog shown below appears.

Click on “**Finish**”.



Chapter 2. Setting the Parallel Interface

If the following message appears, click on “OK” to go to the next step:



Respecify the directory “D:¥Driver¥Win9x” where the driver file is located; click on “OK”.



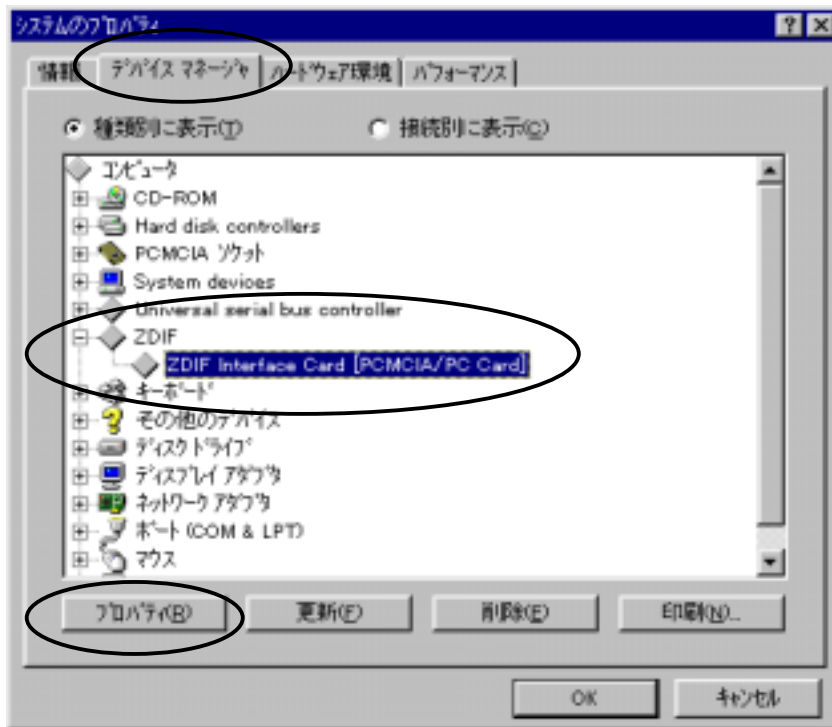
A “ping” sound goes off when the drive is correctly installed.

A buzzer sound indicates an installation failure. If this happens, close the installation process and refer to Appendix J, “Troubleshooting”, to resolve the problem.

Chapter 2. Setting the Parallel Interface

To confirm that the device driver was installed correctly, double-click and open **System** in the **Control Panel**, and click on the **Device Manager** tab.

Click on “ZDIF” class to display the properties of the “ZDIF Interface Card [PCMCIA/PC Card]” below (either double-click or click on “Properties (R)”).



Click on “Resources” tab to make sure that the “Conflicting devices” item indicates “No conflict”.



Chapter 2. Setting the Parallel Interface

Device driver installation procedure (PCI card, Windows 95 OSR2)

When the PCI card is inserted and the power for the host system is turned on, the Add New Hardware Wizard detects the card and displays the following dialog, at which point you should click on “Next>”:



The system automatically searches the A: drive for the device driver. Failing to find it, the system displays the dialog shown below.

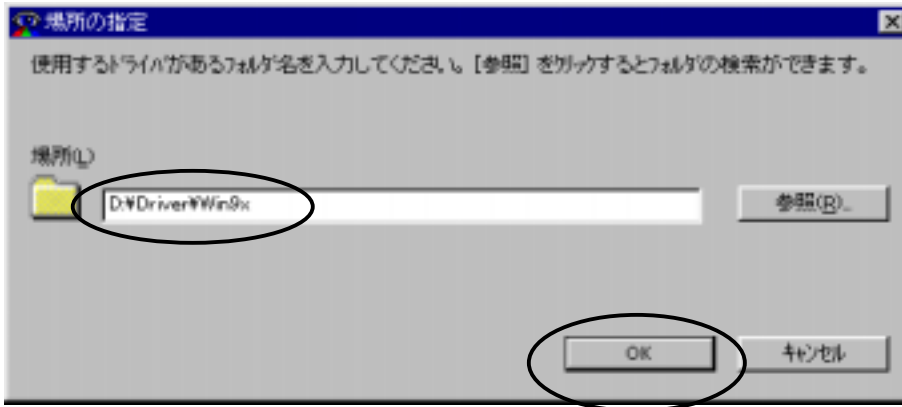
Click on “Browse (O) ...”.



Chapter 2. Setting the Parallel Interface

Load the CD-ROM labeled “*MJX440 for V831/V832 Tools Disk*” in the CD-ROM drive. Select “**Browse (L)**” to specify the directory where the driver is located, e.g., “**D:\Driver\Win9x**” if the CD-ROM drive is drive D: (depending on your specific PC).

Click on “**OK**”.



The driver file is detected; the dialog shown below appears.

Click on “**Finish**”.



The driver is automatically installed.

Chapter 2. Setting the Parallel Interface

If the following message appears, click on “OK” to go to the next step:



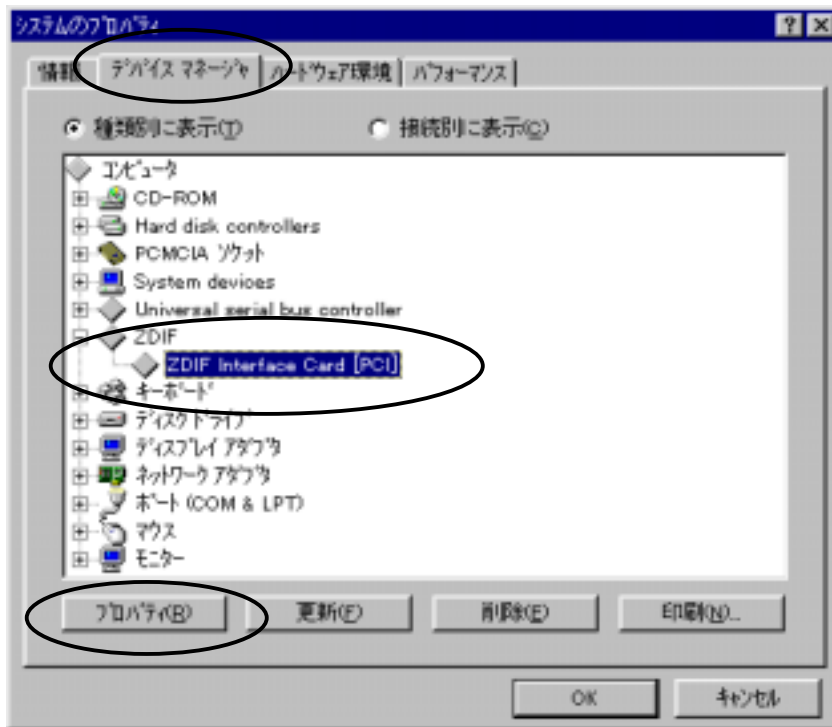
Respecify the directory “D:¥Driver¥Win9x” where the driver file is located; click on “OK”.



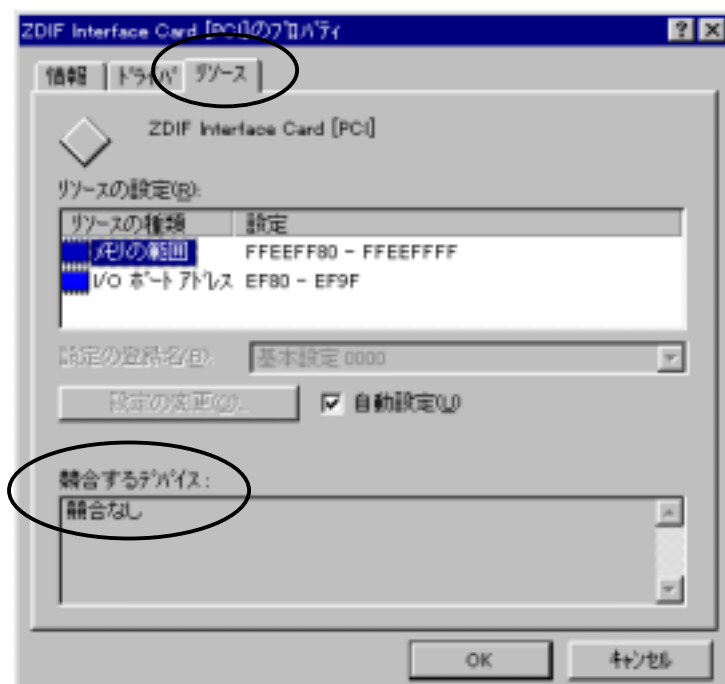
Chapter 2. Setting the Parallel Interface

To confirm that the device driver was installed correctly, double-click and open **System** in the **Control Panel**, and click on the **Device Manager** tab.

Click on “ZDIF” class to display the properties of the “ZDIF Interface Card [PCMCIA/PC Card” below (either double-click or click on “Properties (R)”).



Click on “Resources” tab to make sure that the “Conflicting devices” item indicates “No conflict”.

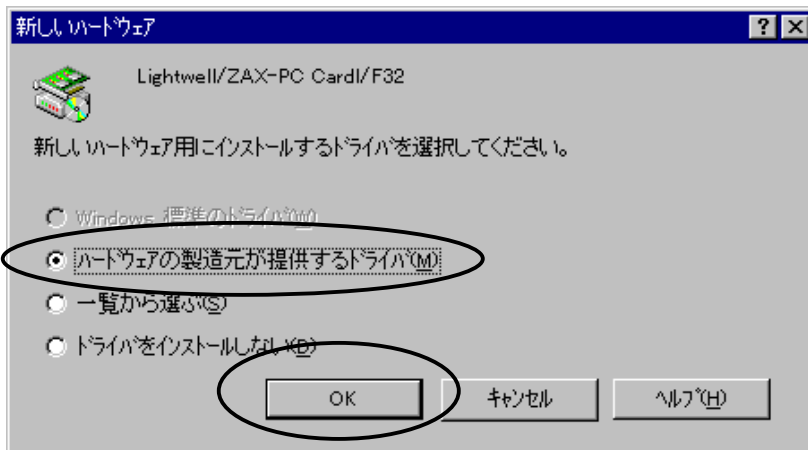


Chapter 2. Setting the Parallel Interface

Device driver installation procedure (PCMCIA card, Windows 95 OSR1)

When the PCMCIA card is inserted with the power for the notebook PC turned on, the Add New Hardware Wizard detects the card and displays the dialog shown below.

Select “**Hardware vendor-supplied driver (M)**”, click on “**OK**”.

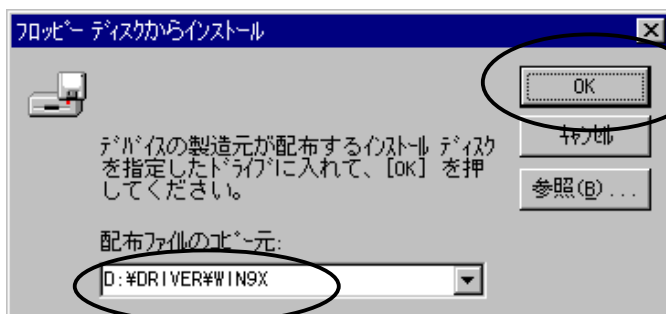


Load the CD-ROM labeled “*MJX440 for V831/V832 Tools Disk*” in the CD-ROM drive.

For “**Copy distribution file from**”, specify the directory where the driver is located.

For example, specify “**D:¥Driver¥Win9x**” if the CD-ROM drive is drive D: (depending on your specific PC).

Click on “**OK**”.



The driver is automatically installed. A “ping” sound goes off when the drive is correctly installed, at which point click on “**Finish**”.

A buzzer sound indicates an installation failure. If this happens, close the installation process and refer to Appendix J, “Troubleshooting”, to resolve the problem.

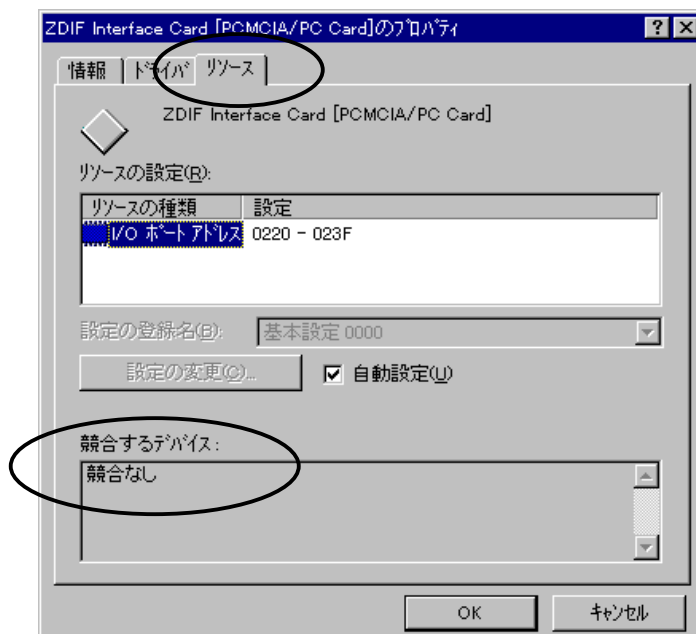
Chapter 2. Setting the Parallel Interface

To confirm that the device driver was installed correctly, double-click and open **System** in the **Control Panel**, and click on the **Device Manager** tab.

Click on “**ZDIF**” class to display the properties of the “**ZDIF Interface Card [PCMCIA/PC Card]**” below (either double-click or click on “Properties (R)” .



Click on “**Resources**” tab to make sure that the “**Conflicting devices**” item indicates “**No conflict**”.

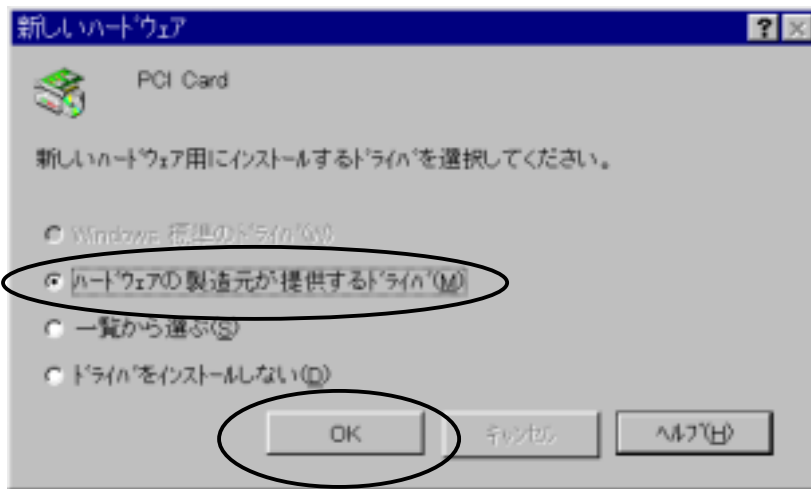


Chapter 2. Setting the Parallel Interface

Device driver installation procedure (PCI card, Windows 95 OSR1)

When the PCI card is inserted and the power for the host system is turned on, the Add New Hardware Wizard detects the card and displays the dialog shown below.

Select “**Hardware vendor-supplied driver (M)**”, click on “**OK**”.

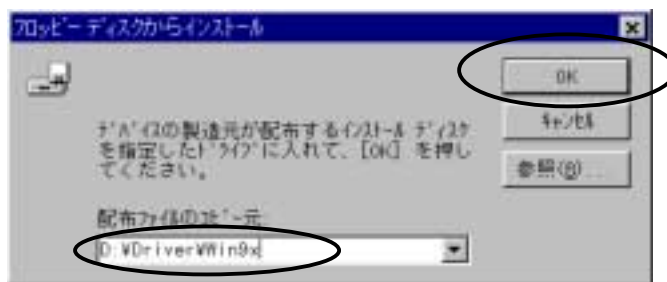


Load the CD-ROM labeled “*MJX440 for V831/V832 Tools Disk*” in the CD-ROM drive.

For “**Copy distribution file from**”, specify the directory where the driver is located.

For example, specify “**D:¥Driver¥Win9x**” if the CD-ROM drive is drive D: (depending on your specific PC).

Click on “**OK**”.



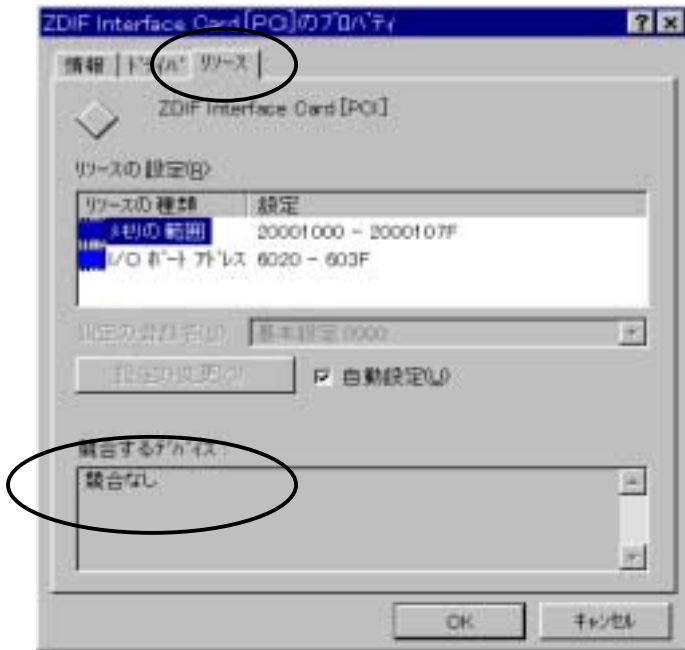
The driver is automatically installed.

Chapter 2. Setting the Parallel Interface

To confirm that the device driver was installed correctly, double-click and open **System** in the **Control Panel**, and click on the **Device Manager** tab.

Click on “**ZDIF**” class to display the properties of the “**ZDIF Interface Card [PCMCIA/PC Card]**” below (either double-click or click on “Properties (R)” .

Click on “**Resources**” tab to make sure that the “Conflicting devices” item indicates “**No conflict**”.



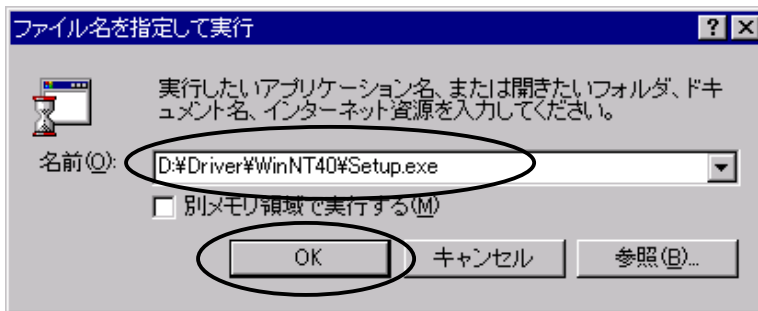
Chapter 2. Setting the Parallel Interface

Device driver installation procedure (PCMCIA card, Windows NT4.0)

With the power for the notebook PC turned off, insert the PCMCIA card, and then turn on the power. When Windows NT4.0 comes up, load the CD-ROM labeled “*MJX440 for V831/V832 Tools Disk*” in the CD-ROM drive.

Select “**Start**” and “**Run (R)**”, and display the “**Run**” dialog.

In “**Open (O)**”, specify “**D:¥Driver¥WinNT40¥Setup.exe**” (assuming that your CD-ROM drive is D: drive), and click on “**OK**”.

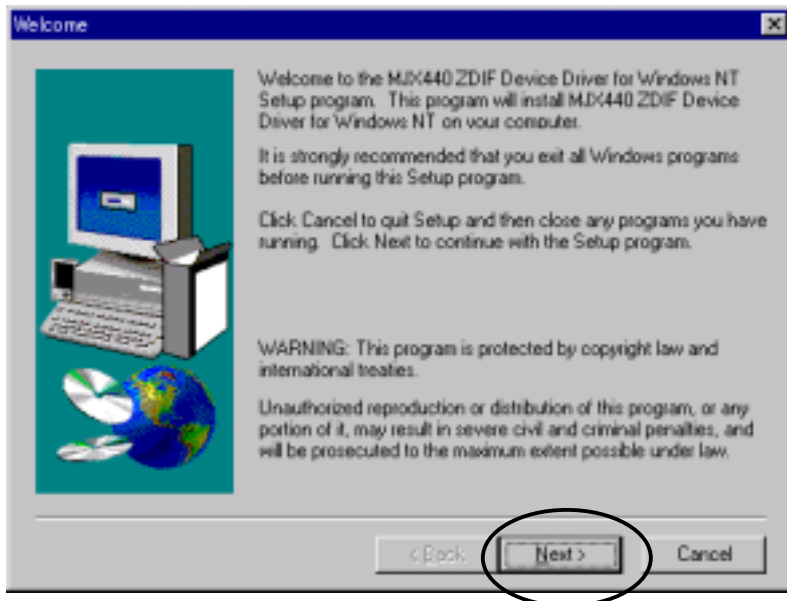


The setup program starts.



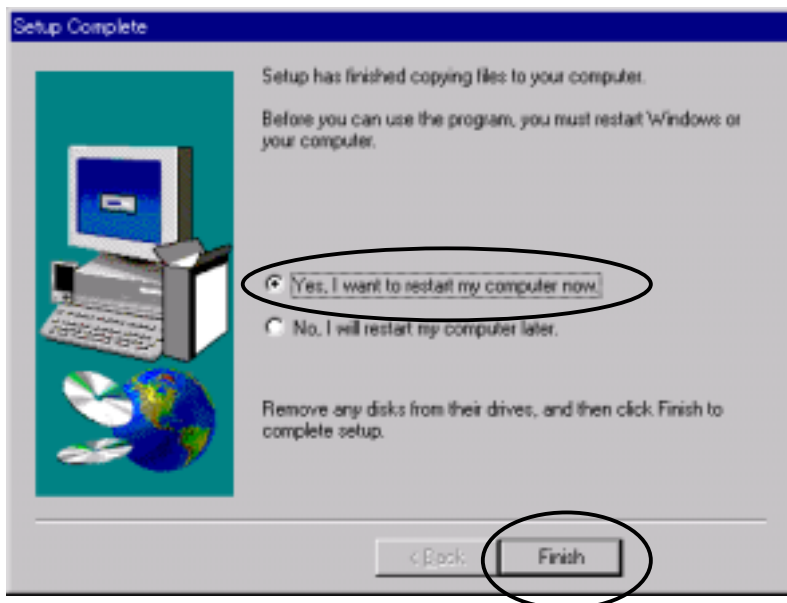
Chapter 2. Setting the Parallel Interface

Click on “Next >”, which automatically installs the driver.



The dialog shown below appears after the driver is installed.

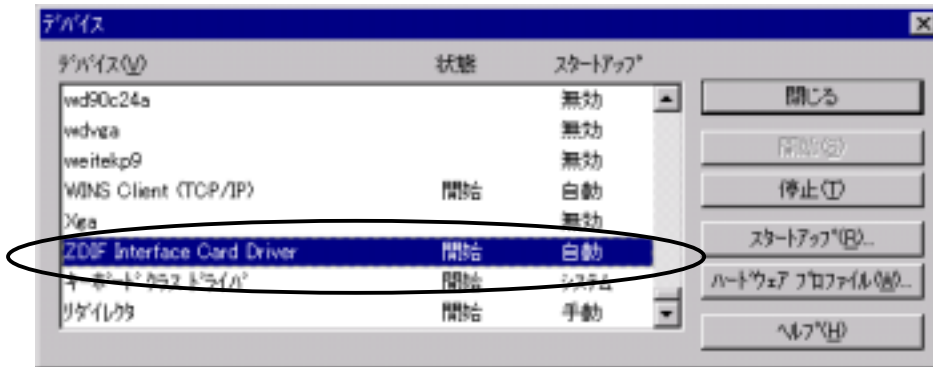
Select “Yes, I want to restart my computer now”, click on “Finish”.



Windows NT4.0 restarts.

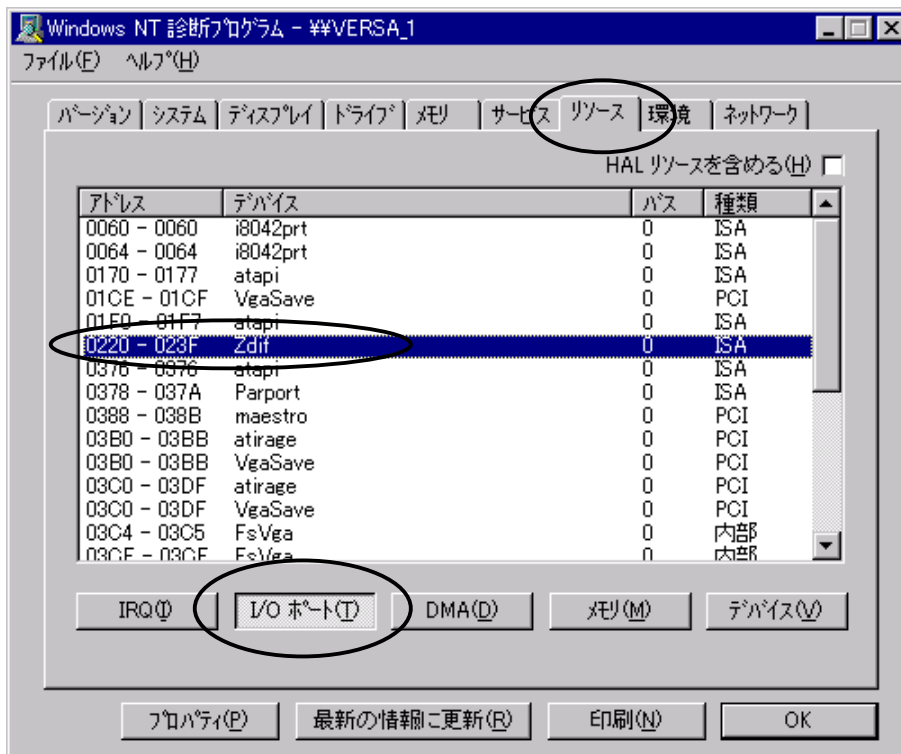
Chapter 2. Setting the Parallel Interface

To confirm that the driver was installed correctly, open “**Devices**” by double-clicking on it in the **Control Panel**. Look for the “**ZDIF Interface Card Driver**”. Make sure that the status column indicates “**Start**” and the Startup column, “**Startup**”.



Select “**Start**”, “**Program (P)**”, and “**Control tool (common)**” to launch the “**Windows NT Diagnostic Program**”.

Click on the “**Resources**” tab, and then click on “**I/O port (T)**”.

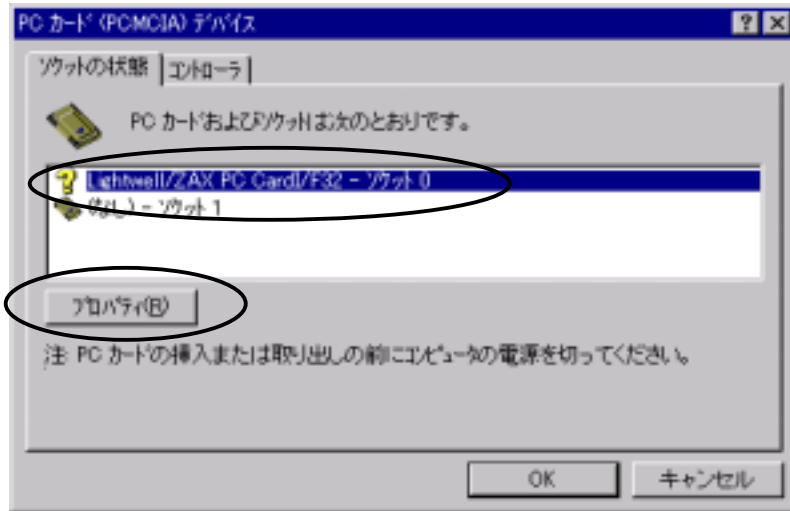


Look for Zdif under “**Device**”. Confirm that the I/O port assigned to the PCMCIA card is displayed under “**Address**”.

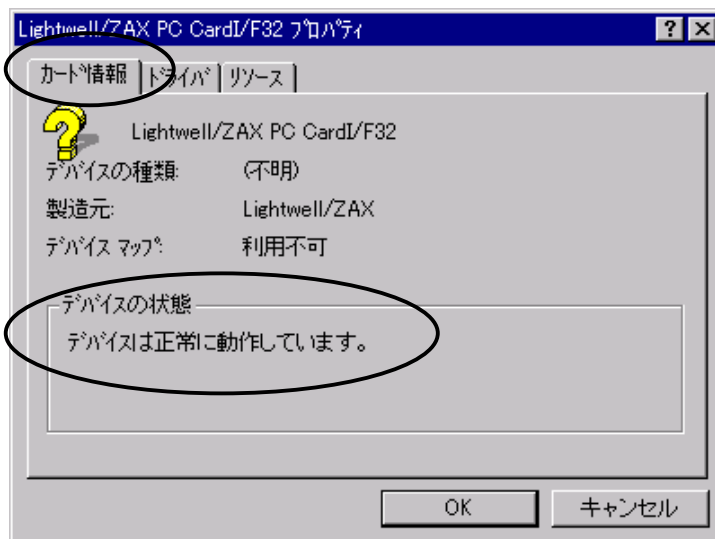
Chapter 2. Setting the Parallel Interface

In the Control Panel, open “PCcard (PCMCIA)” by double-clicking on it.

Click and select “Lightwell/ZAX PC Card I/F32”, click on the “Properties (R)” button.

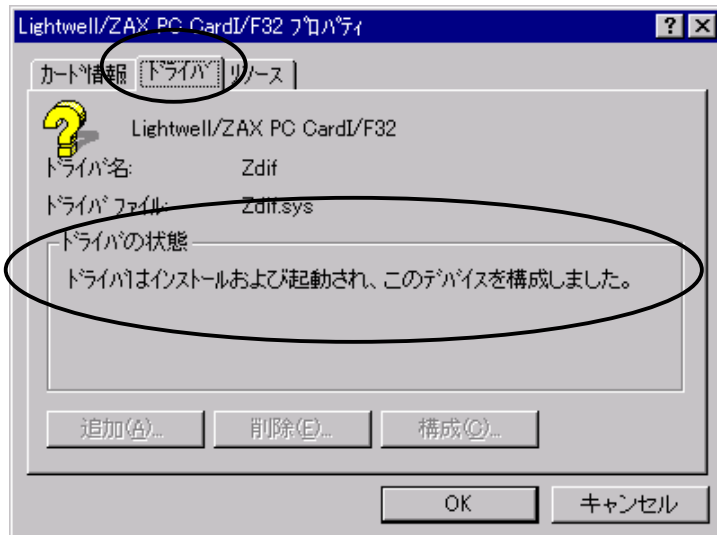


Double click on “Card information” tab to confirm that “Device status” indicates “Device is operating normally”.



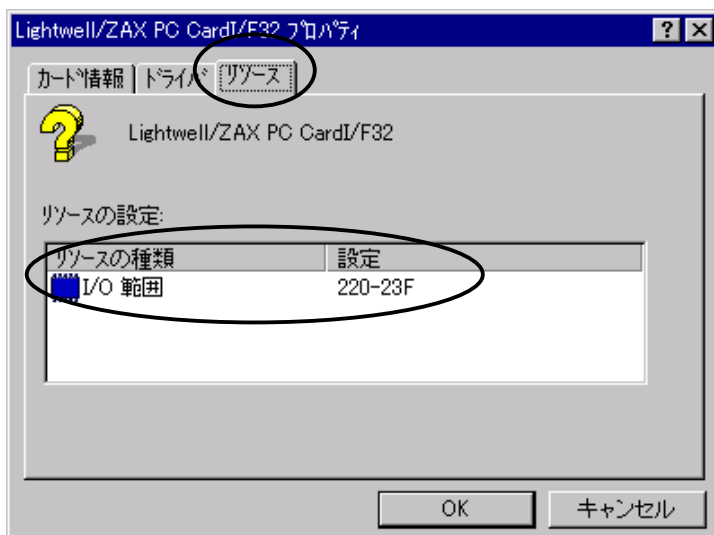
Chapter 2. Setting the Parallel Interface

Click on the “Driver” tab. Confirm that “Driver status” indicates “The driver was installed and started, the device was configured”.



Click on the “Resources” tab. Confirm that the “Resource settings” and the “I/O range” match one of the values listed below:

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



Chapter 2. Setting the Parallel Interface

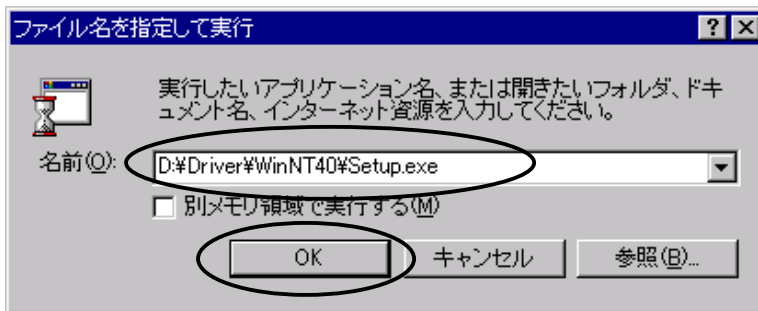
Device driver installation procedure (PCI card, Windows NT4.0)

Insert the PCI card, and then turn on the power for the host system.

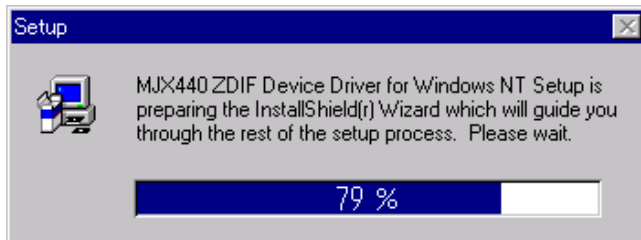
When Windows NT4.0 comes up, load the CD-ROM labeled “*MJX440 for V831/V832 Tools Disk*” in the CD-ROM drive.

Select “**Start**” and “**Run (R)**”, and display the “**Run**” dialog.

In “**Open (O)**”, specify “Setup.exe” (assuming that your CD-ROM drive is D: drive , specify “**D:¥Driver¥WinNT40¥Setup.exe**”, and then click on “**OK**”.

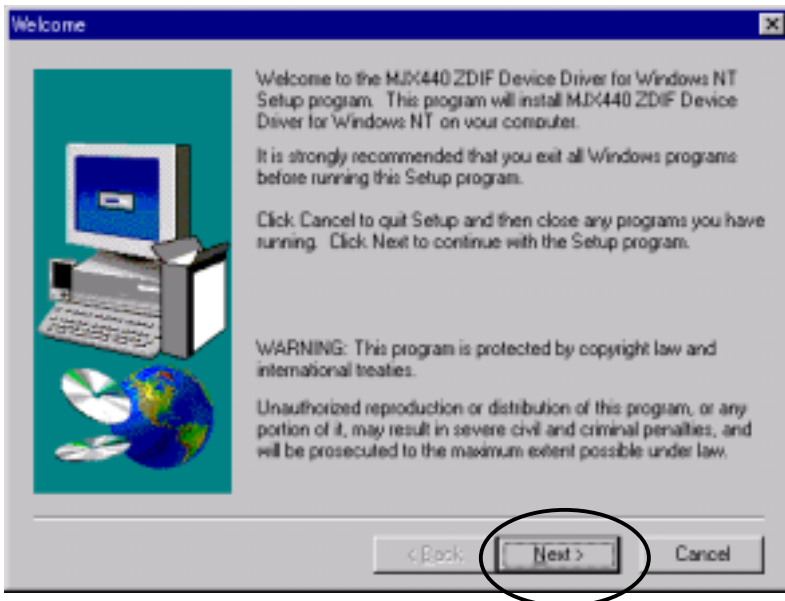


The setup program starts



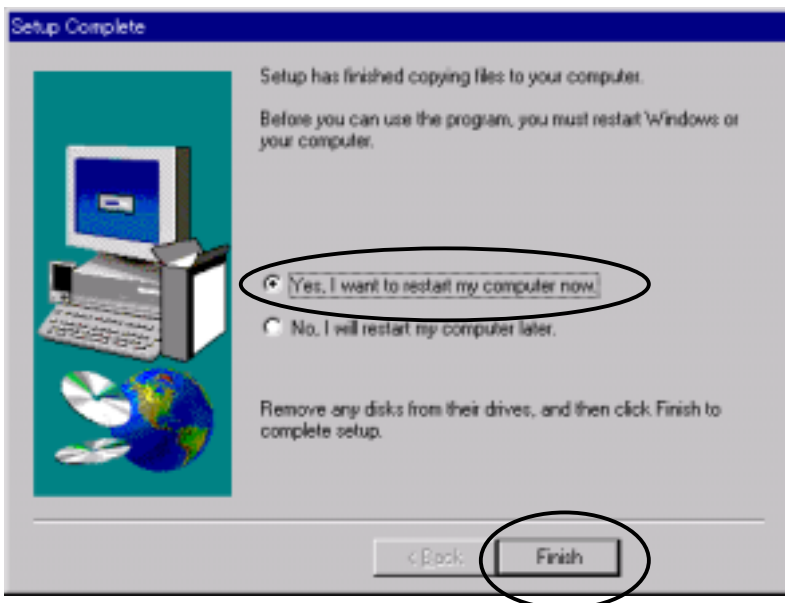
Chapter 2. Setting the Parallel Interface

Click on “Next >”, which automatically installs the driver.



The dialog shown below appears after the driver is installed.

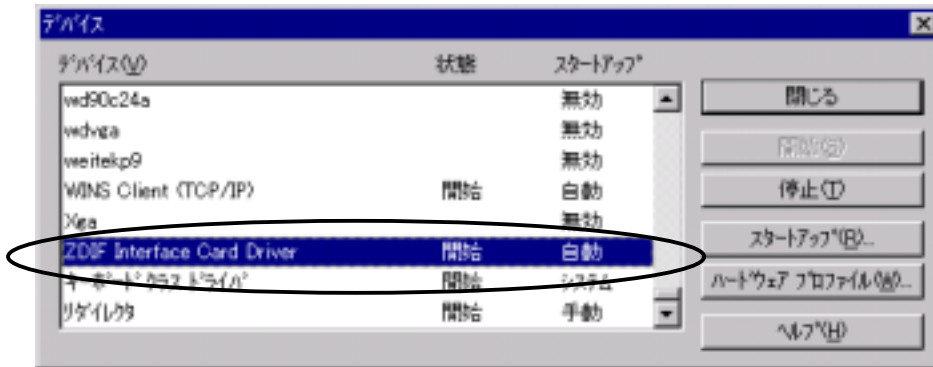
Select “Yes, I want to restart my computer now”, click on “Finish”.



Windows NT4.0 restarts.

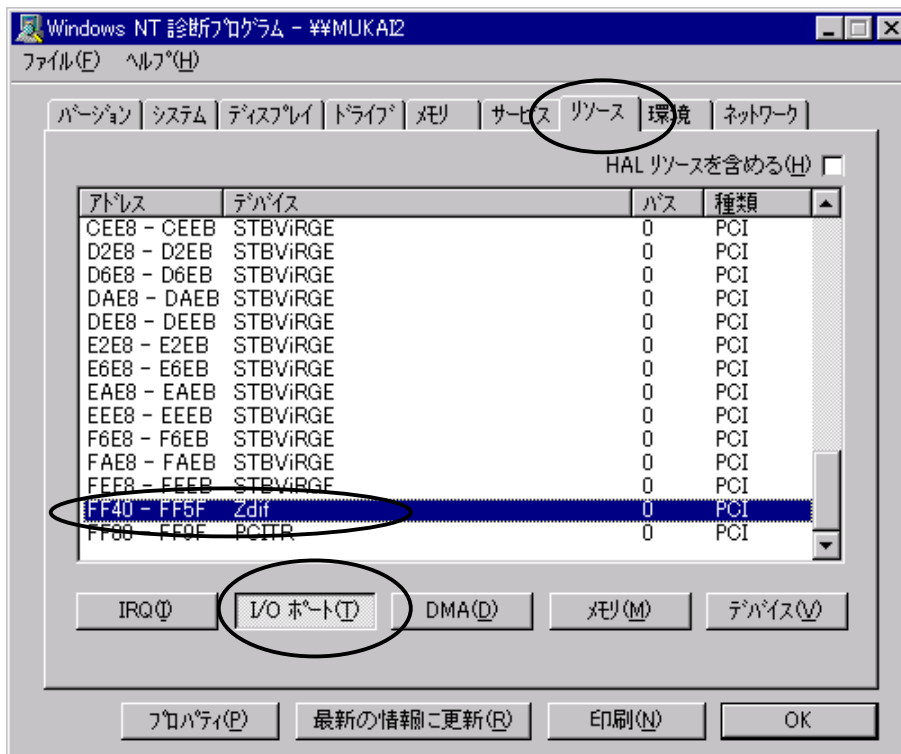
Chapter 2. Setting the Parallel Interface

To confirm that the driver was installed correctly, open “**Devices**” by double-clicking on it in the **Control Panel**. Look for the “**ZDIF Interface Card Driver**”. Make sure that the status column indicates “**Start**” and the Startup column, “**Startup**”.



Select “**Start**”, “**Program (P)**”, and “**Control tool (common)**” to launch the “**Windows NT Diagnostic Program**”.

Click on the “**Resources**” tab, and then click on “**I/O port (T)**”.



Look for Zdif under “**Device**”. Confirm that the I/O port assigned to the PCI card is displayed under “**Address**”.

2.3 Removing/Uninstalling a Device Driver

【Note】 A device driver should be removed/uninstalled only when necessary.

The device driver remove/uninstall procedure is divided into the following sections:

- Device driver removal/uninstallation procedure (PCMCIAcard, Windows 98)
- Device driver removal/uninstallation procedure (PCI card, Windows 98)
- Device driver removal/uninstallation procedure (PCMCIAcard, Windows 95 OSR2)
- Device driver removal/uninstallation procedure (PCICard, Windows 95 OSR2)
- Device driver removal/uninstallation procedure (PCMCIAcard, Windows 95 OSR1)
- Device driver removal/uninstallation procedure (PCICard, Windows 95 OSR1)
- Device driver removal/uninstallation procedure (PCMCIAcard, Windows NT4.0)
- Device driver removal/uninstallation procedure (PCI card, Windows NT4.0)

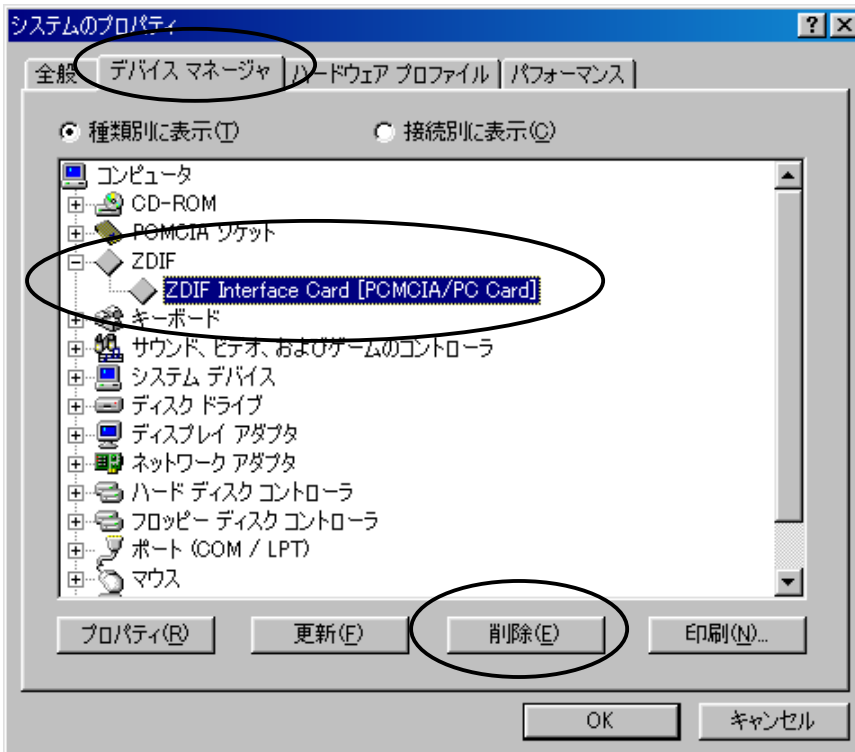
Please refer to the section of the manual that matches your environment.

Chapter 2. Setting the Parallel Interface

Removing/uninstalling device drivers (PCMCIA card, Windows 98)

With the PCMCIA card still inserted, on the **Control Panel**, double-click and open “**System**”, and then click on the “**Device Manager**” tab.

Click on the “**ZDIF**” class, click and select “**ZDIF Interface Card [PCMCIA/PC Card]**” that is located below it, and then click on the “**Remove (E)**” button.



When the “**Confirm Device Removal**” dialog appears, click on the “**OK**” button.



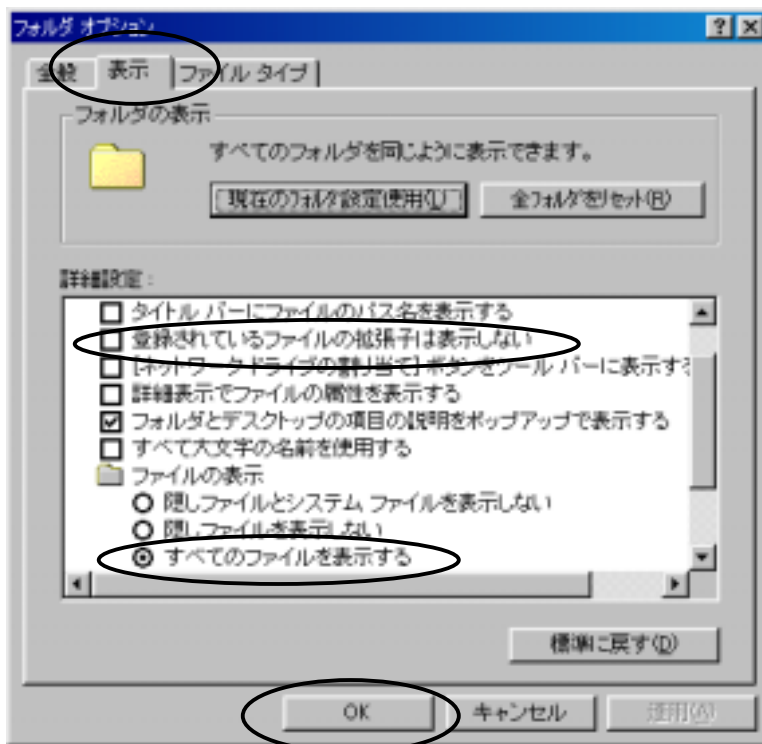
Chapter 2. Setting the Parallel Interface

If you want to completely delete the device driver information for the PCMCIA card, you should also delete the associated INF file.

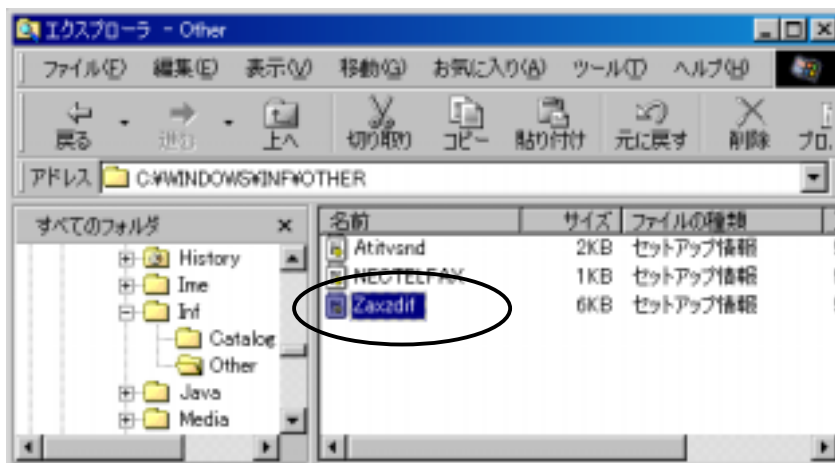
To do this, start **Windows Explorer** and select “**Folder Options (O) ...**” in the “**View (V)**” menu. Double-click on the “**View**” tab on the “**Folder Options**” dialog

Remove the checkmark (off) from “**Hide file extensions for known files**”, check off “**Show all files**” (on), and then click on “**OK**”.

This displays INF files.



Use **Explorer** to open the “**Inf\Other**” directory under the system directory (default --- “**C:\Windows**”), and delete the “**Zaxzdif.inf**” file.

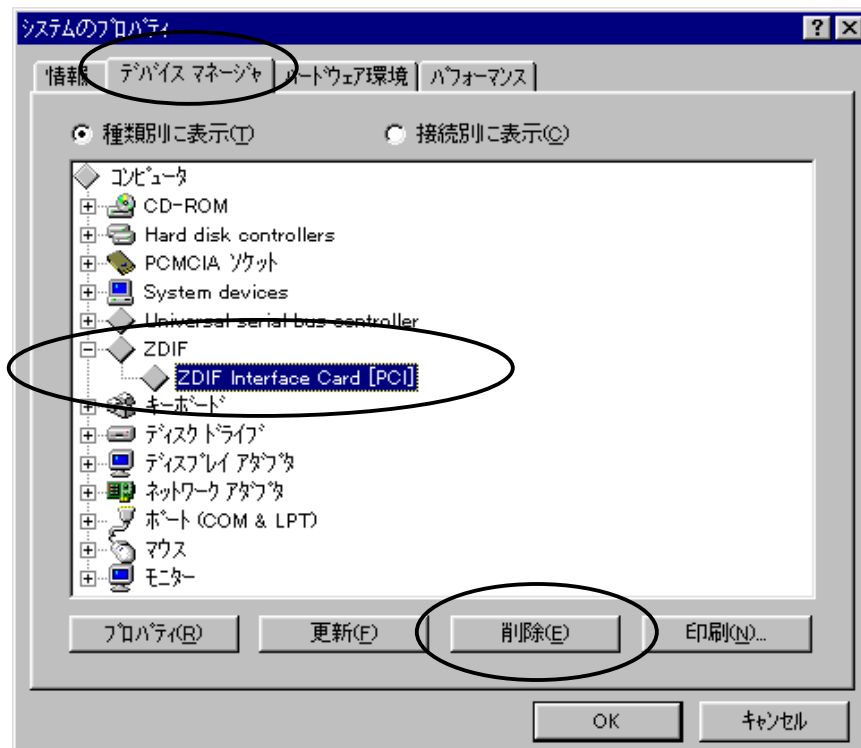


Chapter 2. Setting the Parallel Interface

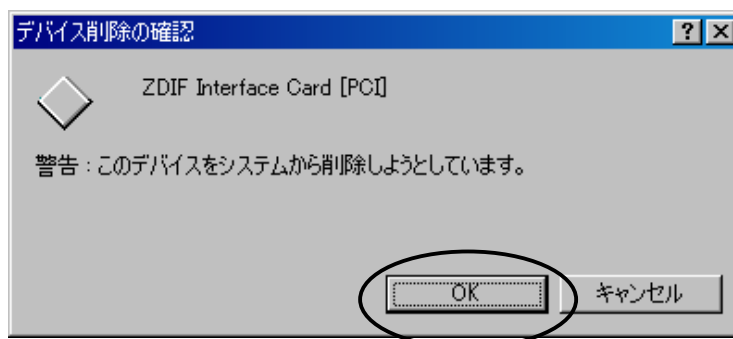
Removing/uninstalling device drivers (PCI card, Windows 98)

With the PCI card still inserted, on the **Control Panel**, double-click and open “**System**”, and then click on the “**Device Manager**” tab.

Click on the “**ZDIF**” class, click and select “**ZDIF Interface Card [PCI]**” that is located below it, and then click on the “**Remove (E)**” button.



When the “**Confirm Device Removal**” dialog appears, click on the “**OK**” button.



Chapter 2. Setting the Parallel Interface

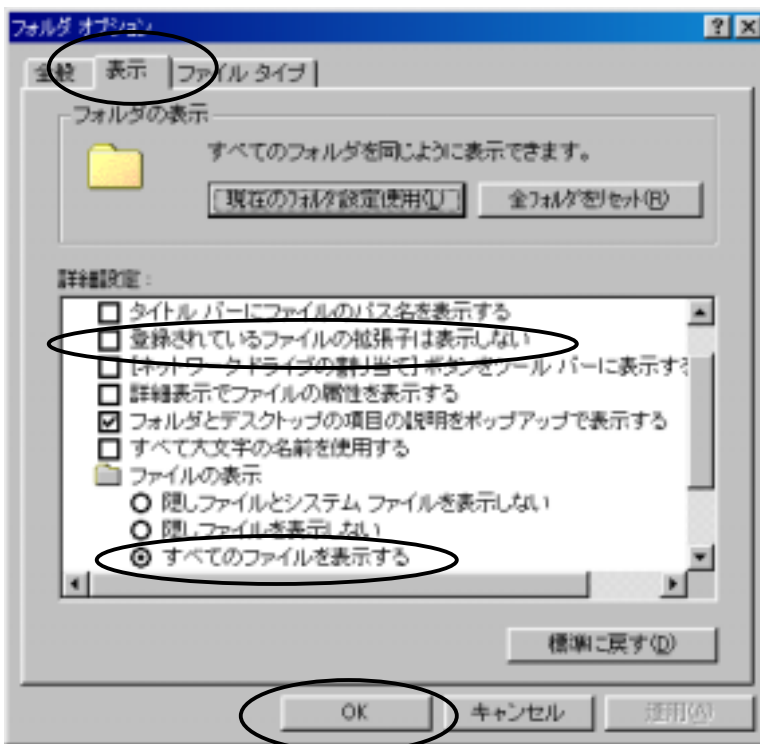
If you want to completely delete the device driver information for the PCMCIA card, you should also delete the associated INF file.

To do this, start **Windows Explorer** and select “**Folder Options (O) ...**” in the “**View (V)**” menu.

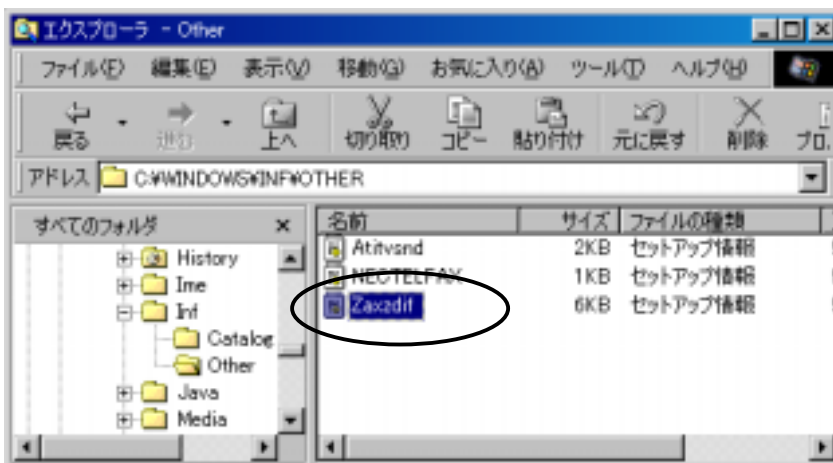
Double-click on the “**View**” tab on the “**Folder Options**” dialog

Remove the checkmark (off) from “**Hide file extensions for known files**”, check off “**Show all files**” (on), and then click on “**OK**”.

This displays INF files.



Use **Explorer** to open the “**Inf\Other**” directory under the system directory (default -- “**C:\Windows**”), and delete the “**Zaxzdif.inf**” file.



Chapter 2. Setting the Parallel Interface

Removing/uninstalling device drivers (PCMCIA card, Windows 95 OSR2)

See “Removing/uninstalling device drivers (PCMCIA card, Windows 98)”.

Chapter 2. Setting the Parallel Interface

Removing/uninstalling device drivers (PCI card, Windows 95 OSR2)

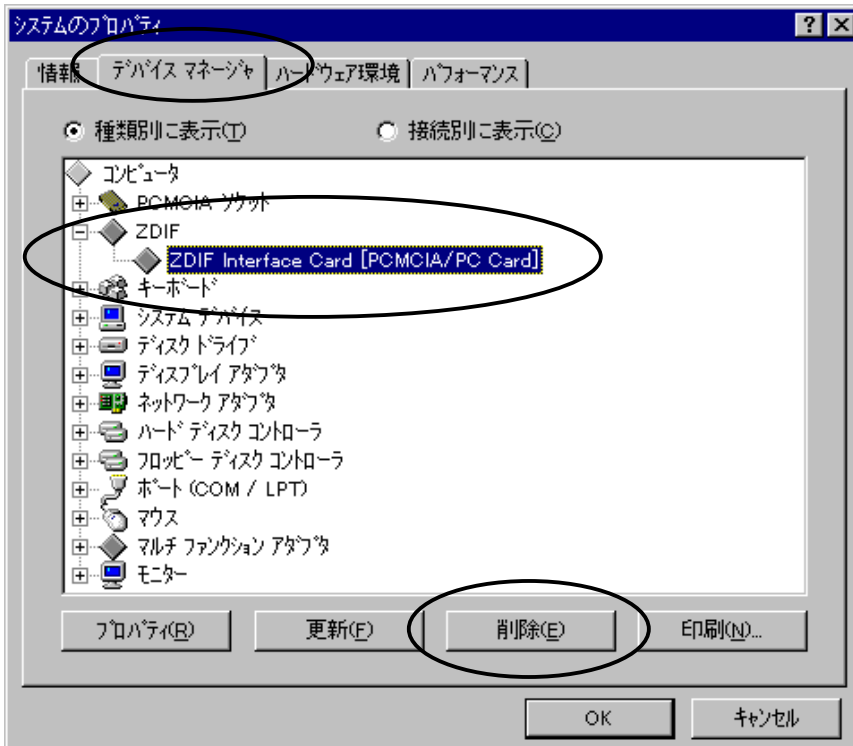
See “Removing/uninstalling device drivers (PCI_card, Windows 98)”.

Chapter 2. Setting the Parallel Interface

Removing/uninstalling device drivers (PCMCIA card, Windows 95 OSR1)

With the PCMCIA card still inserted, on the **Control Panel**, double-click and open “**System**”, and then click on the “**Device Manager**” tab.

Click on the “**ZDIF**” class, click and select “**ZDIF Interface Card [PCMCIA/PC Card]**” that is located below it, and then click on the “**Remove (E)**” button.



When the “**Confirm Device Removal**” dialog appears, click on the “**OK**” button.



Chapter 2. Setting the Parallel Interface

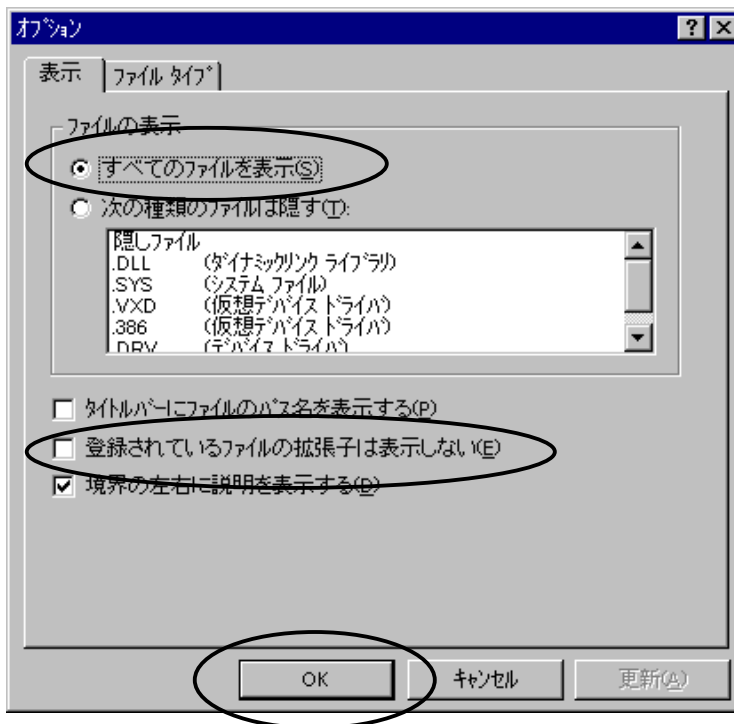
If you want to completely delete the device driver information for the PCMCIA card, you should also delete the associated INF file.

To do this, start **Windows Explorer** and select “**Folder Options (O) ...**” in the “**View (V)**” menu.

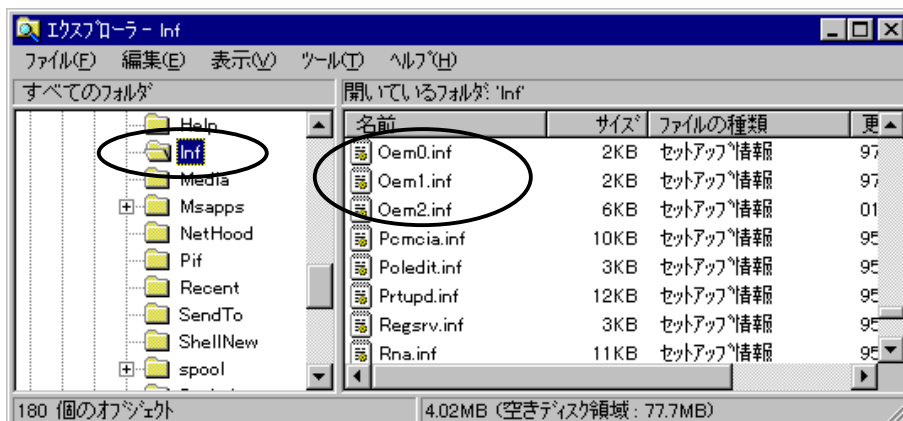
Double-click on the “**View**” tab on the “**Folder Options**” dialog

Remove the checkmark (off) from “**Hide file extensions for known files**”, check off “**Show all files**” (on), and then click on “**OK**”.

This displays INF files.

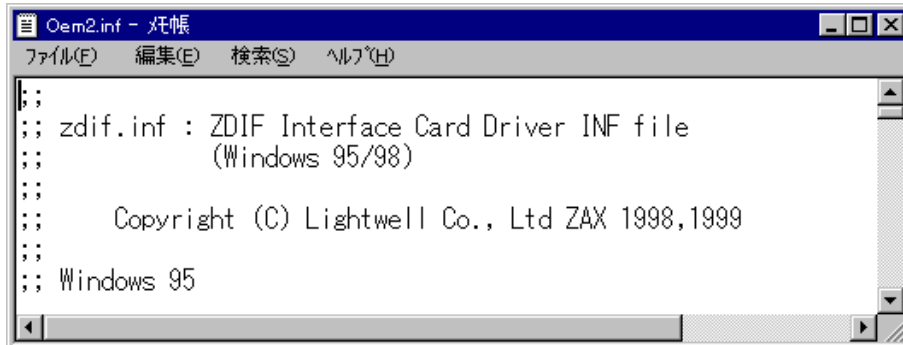


Use **Explorer** to open the “**Inf**” directory under the system directory (default --- “**C:\Windows**”), and look for an “**OemN.inf**” file (where N denotes a numeric value 0 or greater).



Chapter 2. Setting the Parallel Interface

Open the files named "OemN.inf" (where N denotes a numeric value 0 or greater) one by one; delete any file that contains the following comment at the beginning of the file:



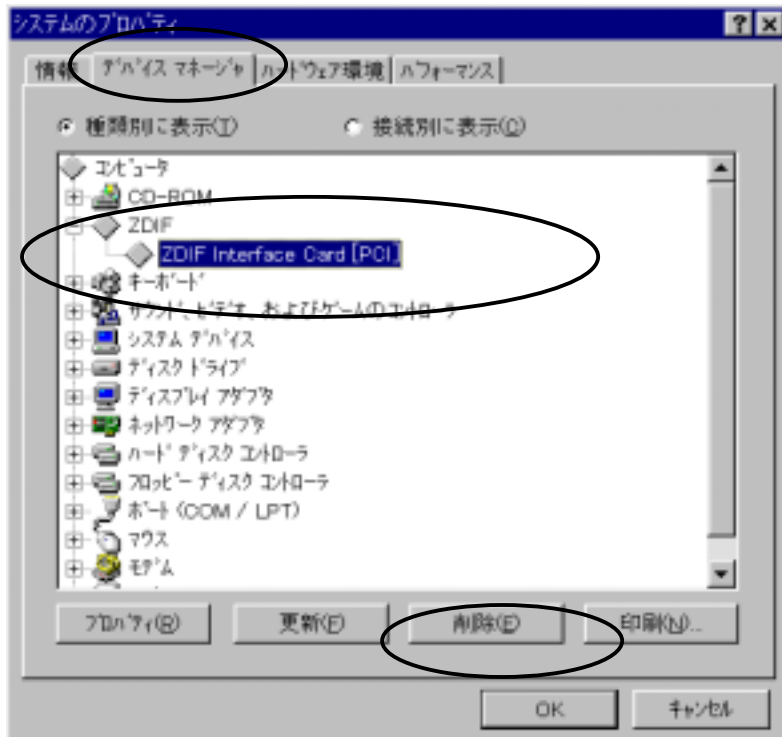
```
; ;  
; ; zdif.inf : ZDIF Interface Card Driver INF file  
; ; (Windows 95/98)  
; ;  
; ; Copyright (C) Lightwell Co., Ltd ZAX 1998,1999  
; ;  
; ; Windows 95  
; ;
```

Chapter 2. Setting the Parallel Interface

Removing/uninstalling device drivers (PCI card, Windows 95 OSR1)

With the PCI card still inserted, on the **Control Panel**, double-click and open “**System**”, and then click on the “**Device Manager**” tab.

Click on the “**ZDIF**” class, click and select “**ZDIF Interface Card [PCI]**” that is located below it, and then click on the “**Remove (E)**” button.



When the “**Confirm Device Removal**” dialog appears, click on the “**OK**” button.



Chapter 2. Setting the Parallel Interface

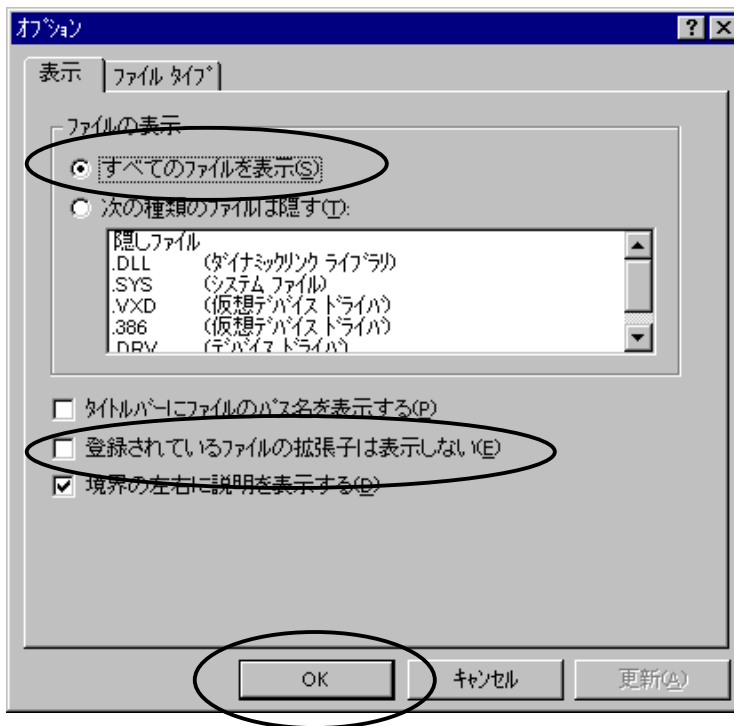
If you want to completely delete the device driver information for the PCMCIA card, you should also delete the associated INF file.

To do this, start **Windows Explorer** and select “**Folder Options (O) ...**” in the “**View (V)**” menu.

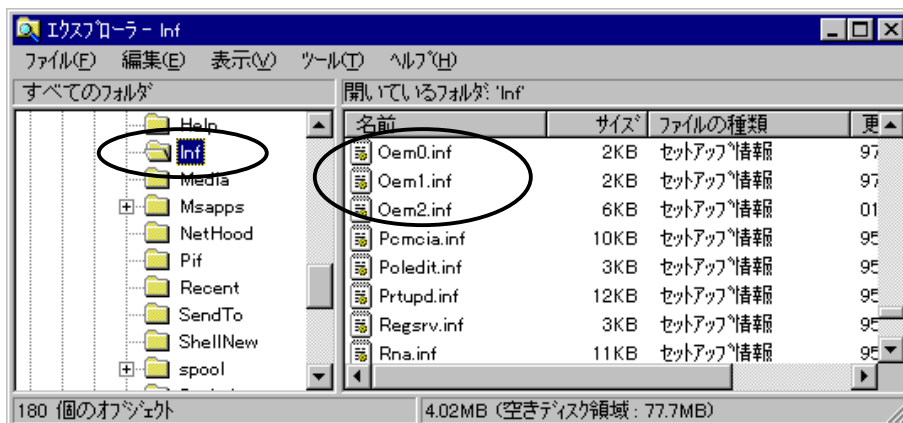
Double-click on the “**View**” tab on the “**Folder Options**” dialog

Remove the checkmark (off) from “**Hide file extensions for known files**”, check off “**Show all files**” (on), and then click on “**OK**”.

This displays INF files.

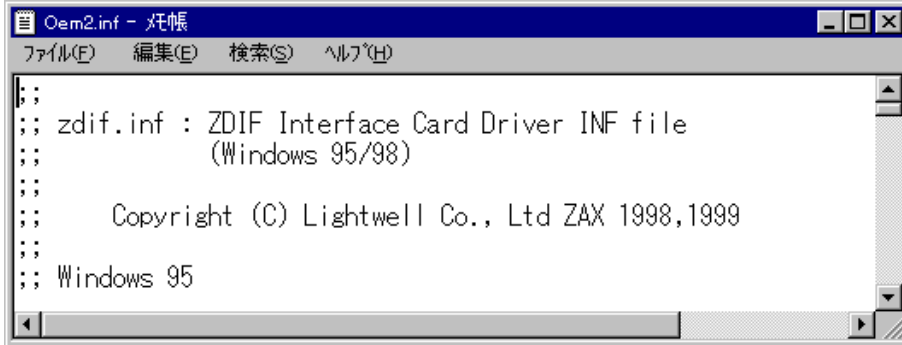


Use **Explorer** to open the “**Inf**” directory under the system directory (default --- “**C:\Windows**”), and look for an “**OemN.inf**” file (where N denotes a numeric value 0 or greater).



Chapter 2. Setting the Parallel Interface

Open the files named "OemN.inf" (where N denotes a numeric value 0 or greater) one by one; delete any file that contains the following comment at the beginning of the file:



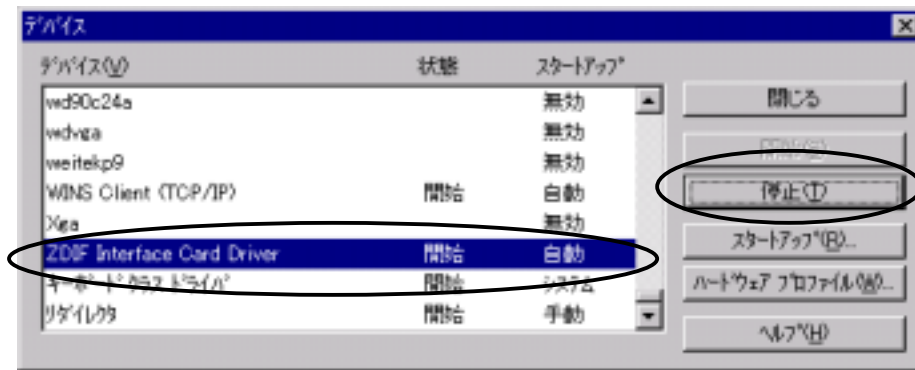
```
;;  
;; zdif.inf : ZDIF Interface Card Driver INF file  
;; (Windows 95/98)  
;;  
;; Copyright (C) Lightwell Co., Ltd ZAX 1998,1999  
;;  
;;  
;; Windows 95  
;;
```

Chapter 2. Setting the Parallel Interface

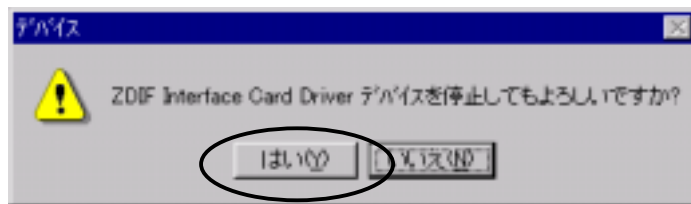
Removing/uninstalling device drivers (PCMCIA card, Windows NT4.0)

With the PCMCIA card still inserted, on the **Control Panel**, double-click and open “**Devices**”. Then click and select “**ZDIF Interface Card Driver**”.

Next, click on the “**Stop (T)**” button.



When the following dialog appears, click on “**Yes (Y)**”:



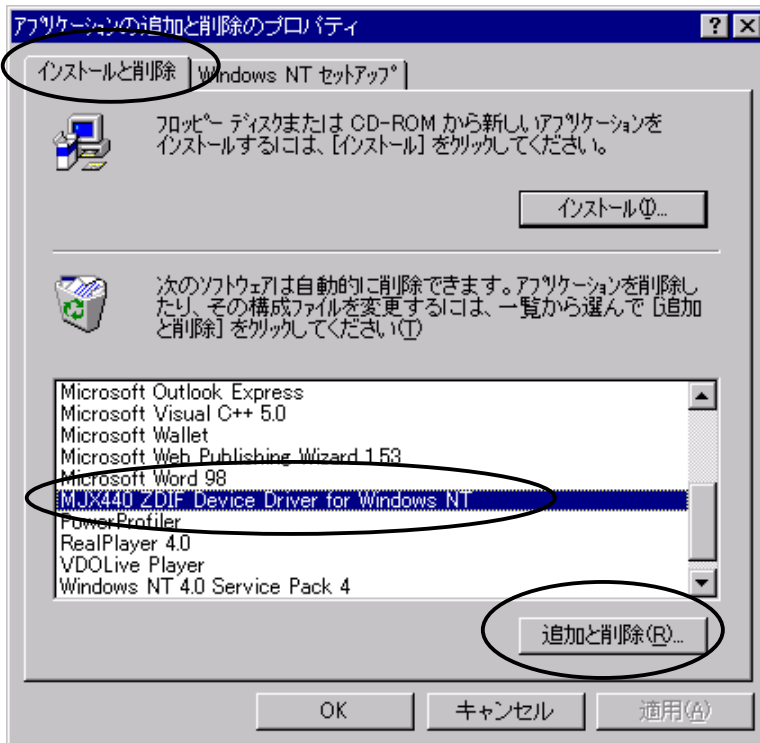
The device halts and the driver is unloaded.



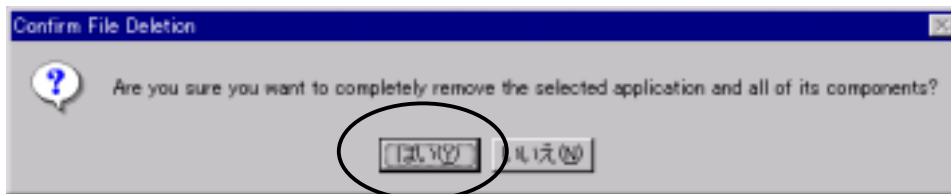
Chapter 2. Setting the Parallel Interface

On the **Control Panel**, double-click and open “**Add/Remove Programs**”, and click on the “**Install/Uninstall**” tab.

Select “**MJX440 ZDIF Device Driver for Windows NT**” by clicking on it and click on the “**Add/Remove**” button.



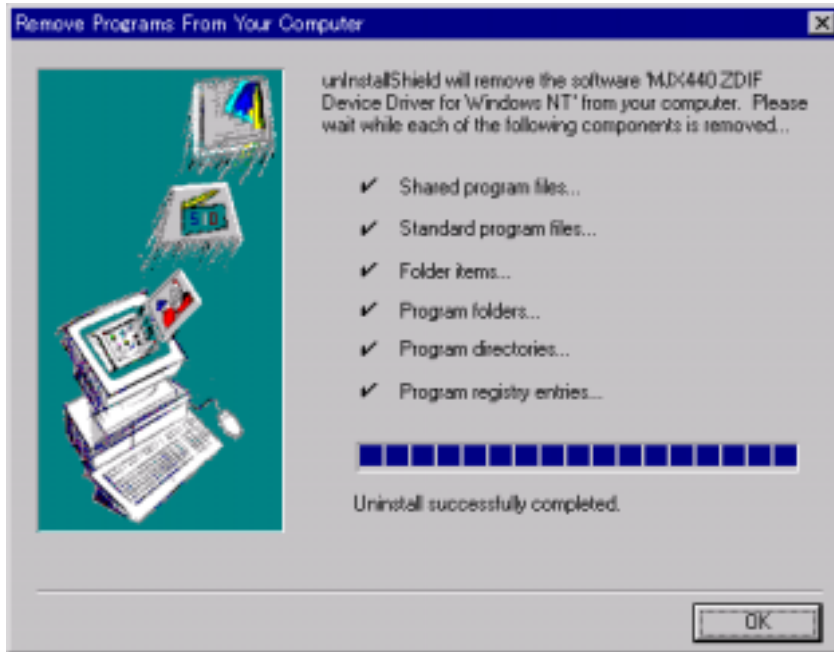
When the “**Confirm File Deletion**” dialog appears, click on “**Yes (Y)**”.



Chapter 2. Setting the Parallel Interface

The driver is automatically uninstalled.

Click on "OK" to exit.



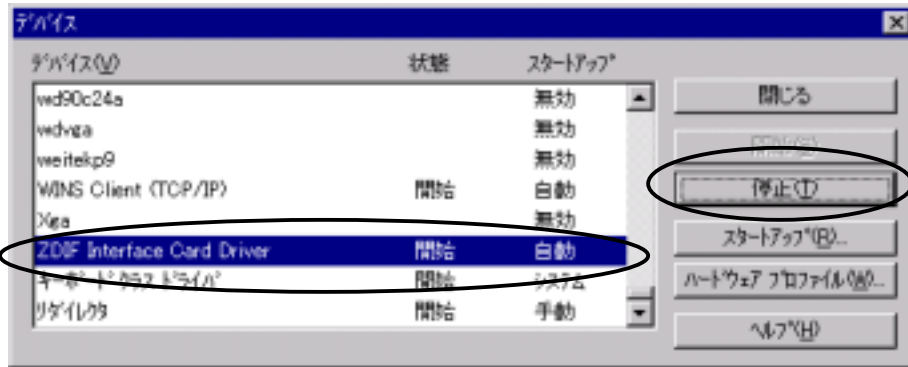
Shut Windows NT4.0 down, turn off the power for the notebook PC, and then remove the PCMCIA card.

Chapter 2. Setting the Parallel Interface

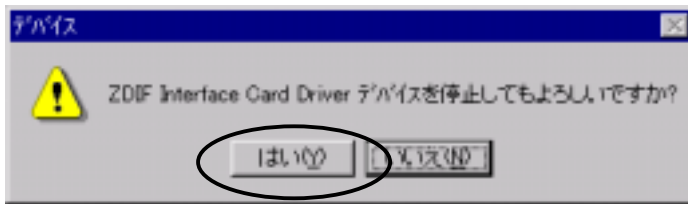
Removing/uninstalling device drivers (PCI card, Windows NT4.0)

With the PCI card still inserted, on the **Control Panel**, double-click and open “**Devices**”. Then click and select “**ZDIF Interface Card Driver**”.

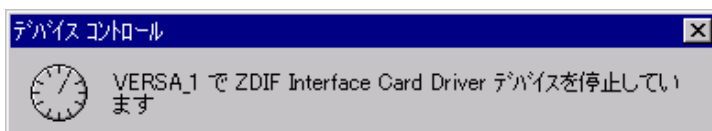
Next, click on the “**Stop (T)**” button.



When the following dialog appears, click on “**Yes (Y)**”:



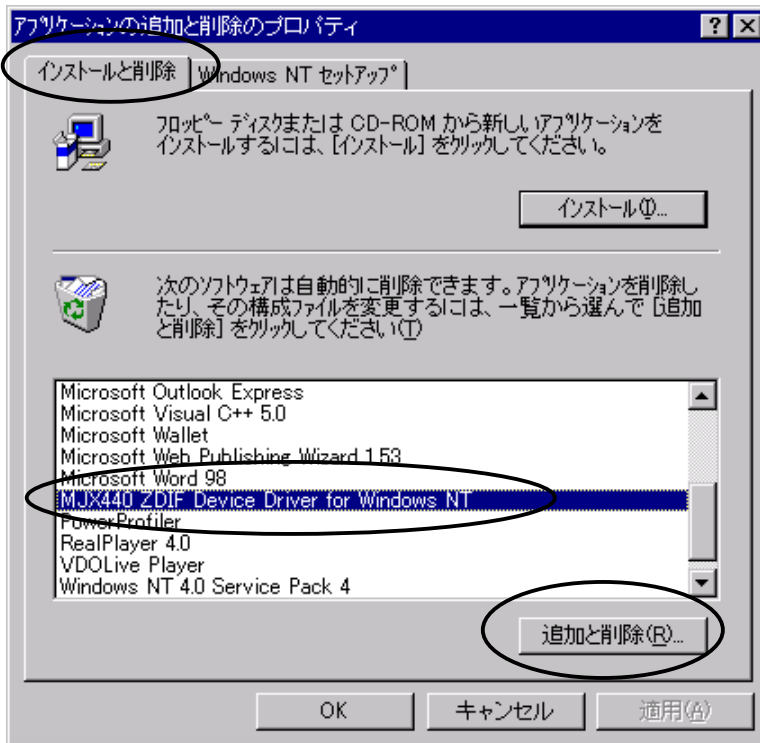
The device halts and the driver is unloaded.



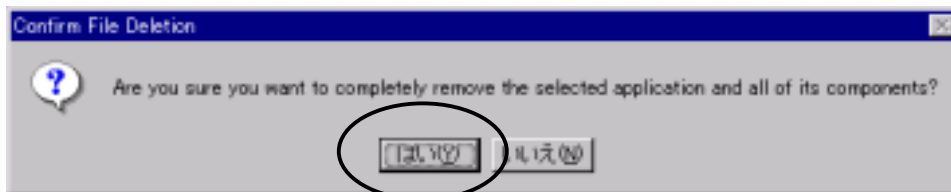
Chapter 2. Setting the Parallel Interface

On the **Control Panel**, double-click and open “**Add/Remove Programs**”, and click on the “**Install/Uninstall**” tab.

Select “**MJX440 ZDIF Device Driver for Windows NT**” by clicking on it and click on the “**Add/Remove**” button.



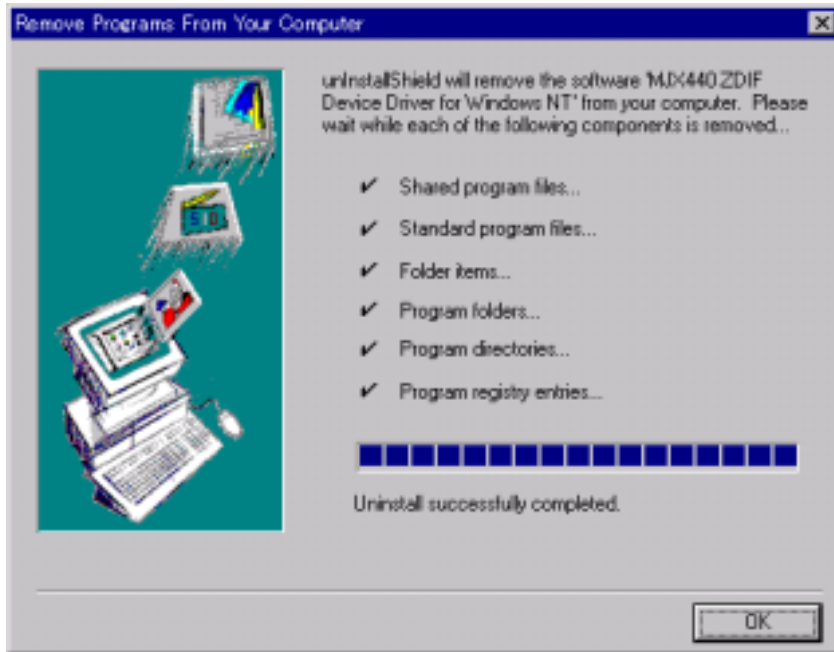
When the “**Confirm File Deletion**” dialog appears, click on “**Yes (Y)**”.



Chapter 2. Setting the Parallel Interface

The driver is automatically uninstalled.

Click on "OK" to exit.



Shut Windows NT4.0 down, turn off the power for the notebook PC, and then remove the PCI card.

Chapter 3. Connecting the Hardware

This chapter describes how to connect the MJX440 to the host as well as how to connect the MJX440 to the target system.

【Important】 Before connecting the MJX440 to other devices, be sure to turn off the power for the devices.

3.1 Connecting the MJX440 to the Host

Connect the PCI or PCMCIA card installed in the host system to the HOST I/F connector of the MJX440, using a parallel interface cable.

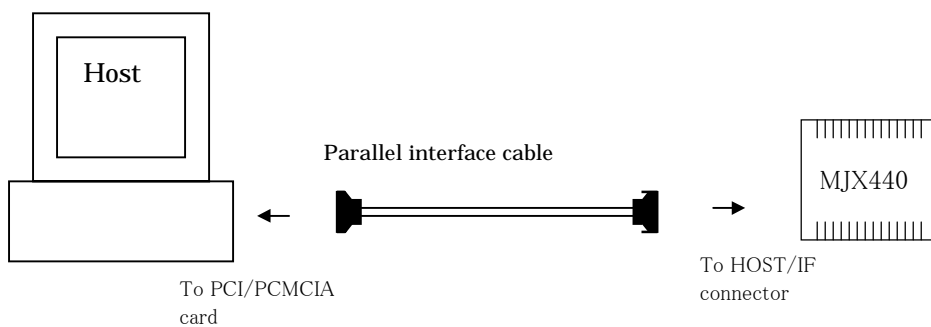


Figure 3-1. Connecting the MJX440 to the host

【Notes】 The following limitations apply due to the thickness of the connector that connects the PCMCIA cable to the card:

- ◆ Even when there are two PCMCIA card slots, only one card can be used in some cases. There are also cases where a card can be inserted only into the bottom slot.
- ◆ An attempt to force the insertion of two cards can damage the PCMCIA card slot and the PCMCIA card connector part.

3.2 Connecting an N-Wire Cable

Use an N-Wire cable to connect the N-Wire cable for the MJX440 to the N-Wire connector on the target system.

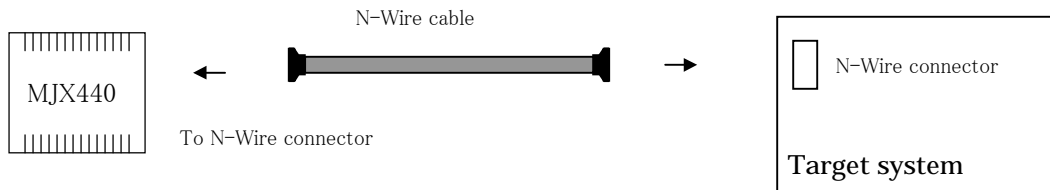


Figure 3-2. Connecting the N-Wire cable

【Note】 To ensure the proper orientation of the connectors, make sure that the \triangle marks on the connectors match.

Chapte 3. Connecting the Hardware

How to connect a ROM probe depends on the following characteristics of the target system:

- ROM data bus width
- Number of ROM chips
- ROM access bus width

In Figures 3-5-1 through 3-5-9, select the connection diagram that matches your target system and connect the ROM probes to the ROM probe cables according to the diagram.

【Note】 Be sure to turn off both the MJX440 and the target system.

【Note】 Be careful not to plug in the ROM probe in reverse.

【Note】 Distinguish the two ROM plugs that can be connected to ROM probes J-101A and J-104A by using the silk-printed characters (JROM1 or JROM2) that can be found on the ROM probe board.

Chapte 3. Connecting the Hardware

One 8-bit bus width ROM chip and an 8-bit ROM access bus width:
(LED lit: ROM1)

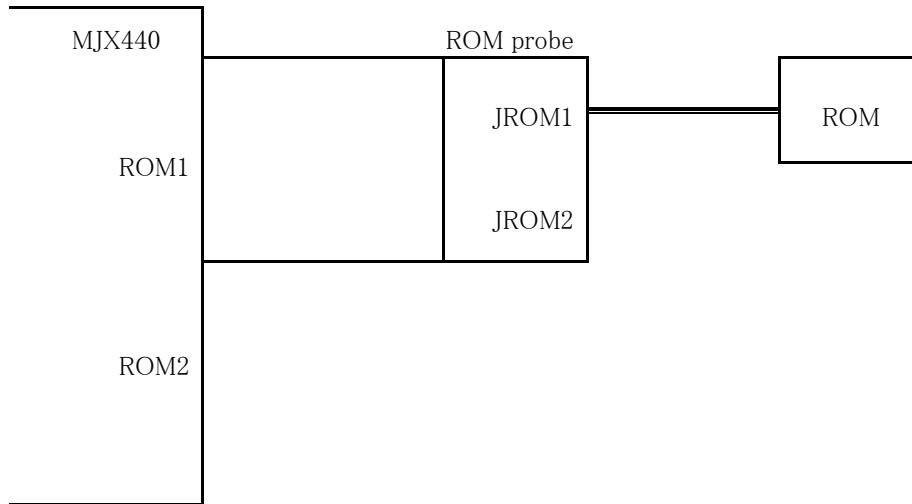


Figure 3-5-1. Connecting a ROM probe (1)

Two 8-bit bus width ROM chips and an 8-bit ROM access bus width:
(LEDs lit: ROM1, ROM2)

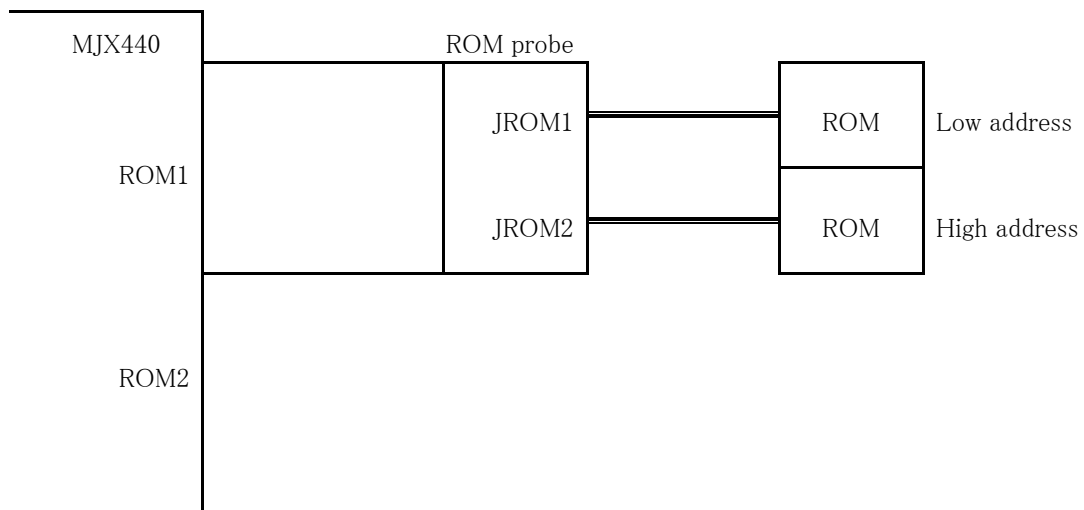


Figure 3-5-2. Connecting a ROM probe (2)

Chapte 3. Connecting the Hardware

Four 8-bit bus width ROM chips and an 8-bit ROM access bus width:
 (LEDs lit: ROM1, ROM2, ROM3, ROM4)

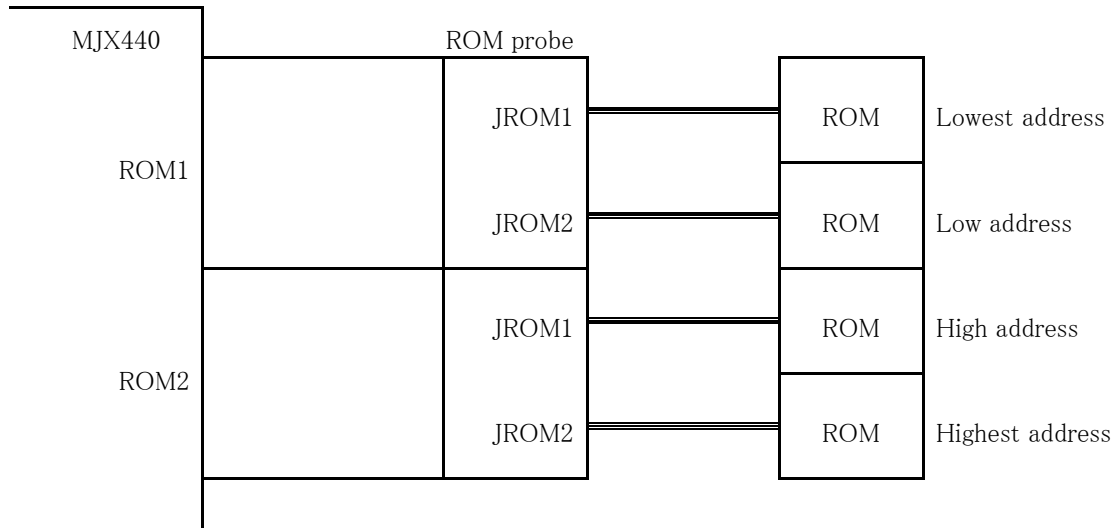


Figure 3-5-3. Connecting a ROM probe (3)

Two 8-bit bus width ROM chips and a 16-bit ROM access bus width:
 (LEDs lit: ROM1, ROM2)

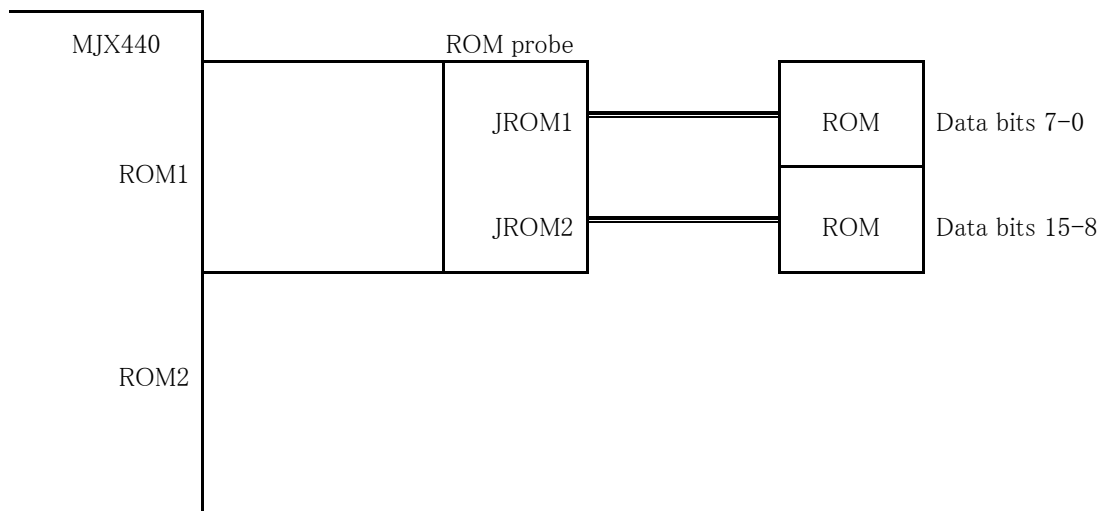


Figure 3-5-4. Connecting a ROM probe (4)

Chapte 3. Connecting the Hardware

Four 8-bit bus width ROM chips and a 16-bit ROM access bus width:
 (LEDs lit: ROM1, ROM2, ROM3, ROM4)

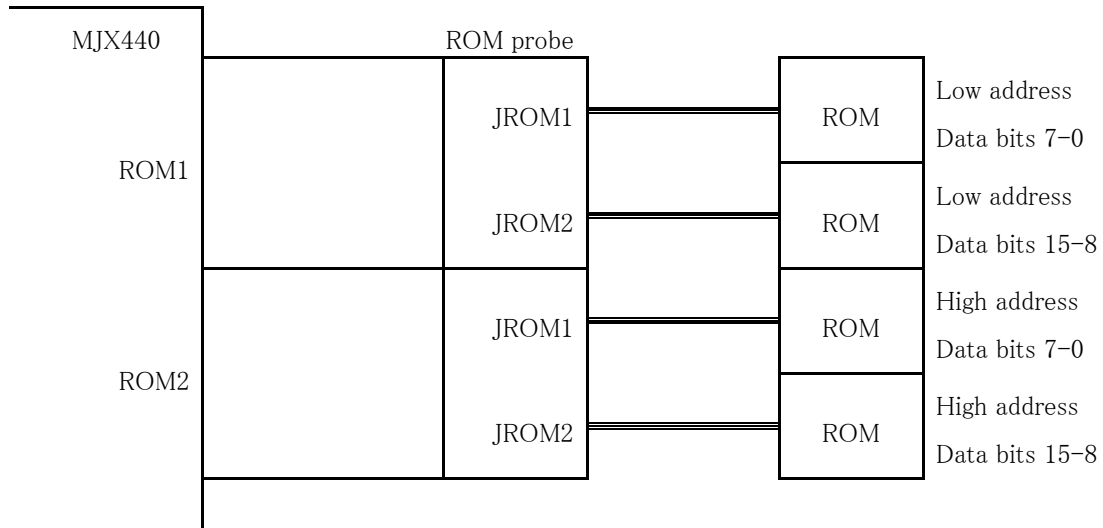


Figure 3-5-5. Connecting a ROM probe (5)

Four 8-bit bus width ROM chips and a 32-bit ROM access bus width:
 (LEDs lit: ROM1, ROM2, ROM3, ROM4)

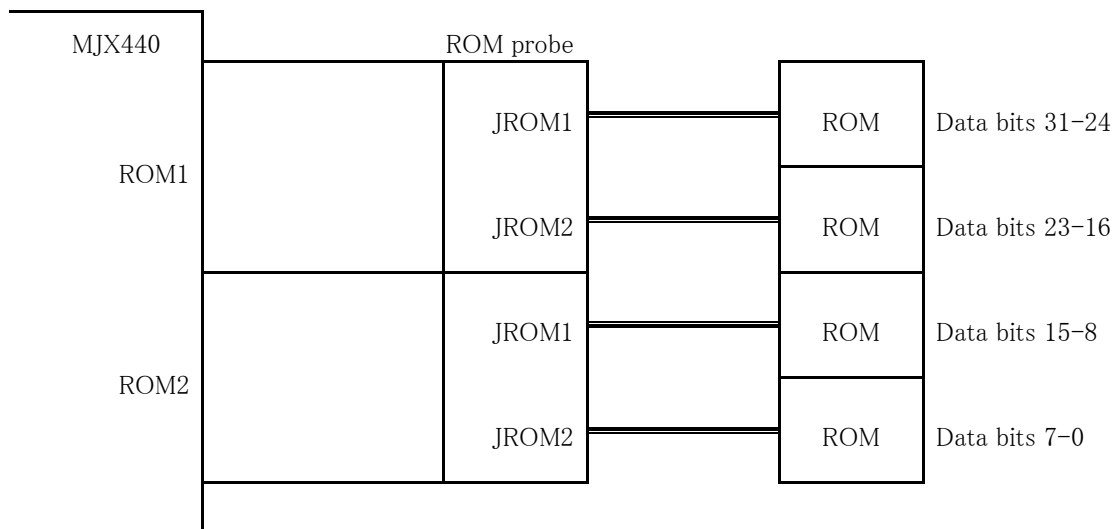


Figure 3-5-6. Connecting a ROM probe (6)

Chapte 3. Connecting the Hardware

One 16-bit bus width ROM chip and a 16-bit ROM access bus width:
(LED lit: ROM1)

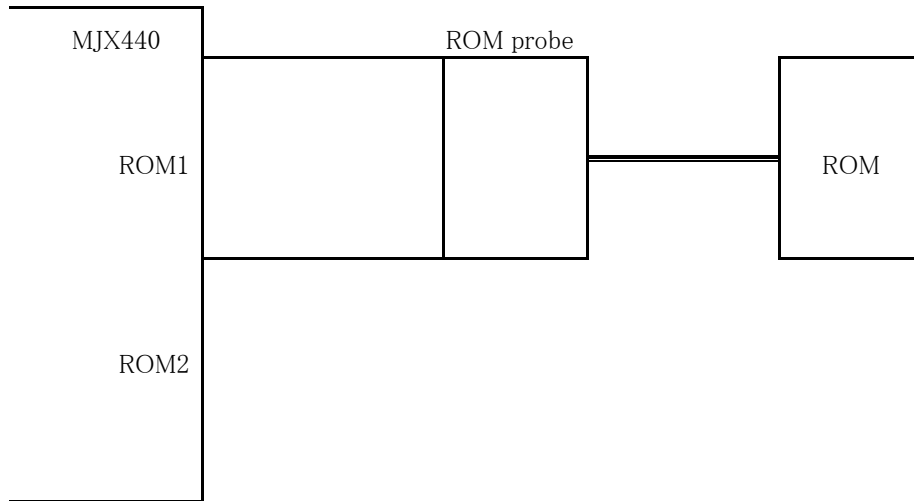


Figure 3-5-7. Connecting a ROM probe (7)

Two 16-bit bus width ROM chips and a 16-bit ROM access bus width:
(LEDs lit: ROM1, ROM2)

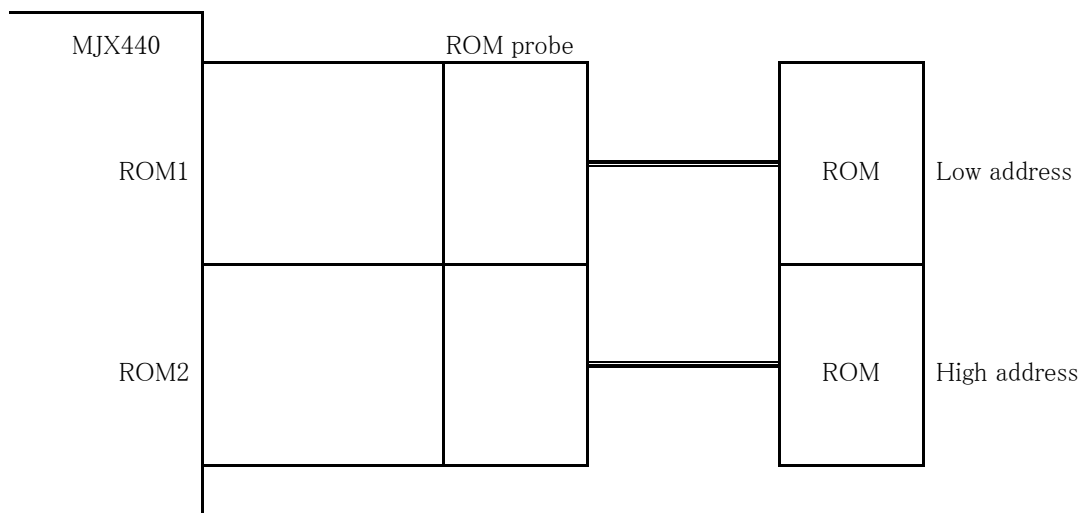


Figure 3-5-8. Connecting a ROM probe (8)

Chapte 3. Connecting the Hardware

Two 16-bit bus width ROM chips and a 32-bit ROM access bus width:
(LEDs lit: ROM1, ROM2)

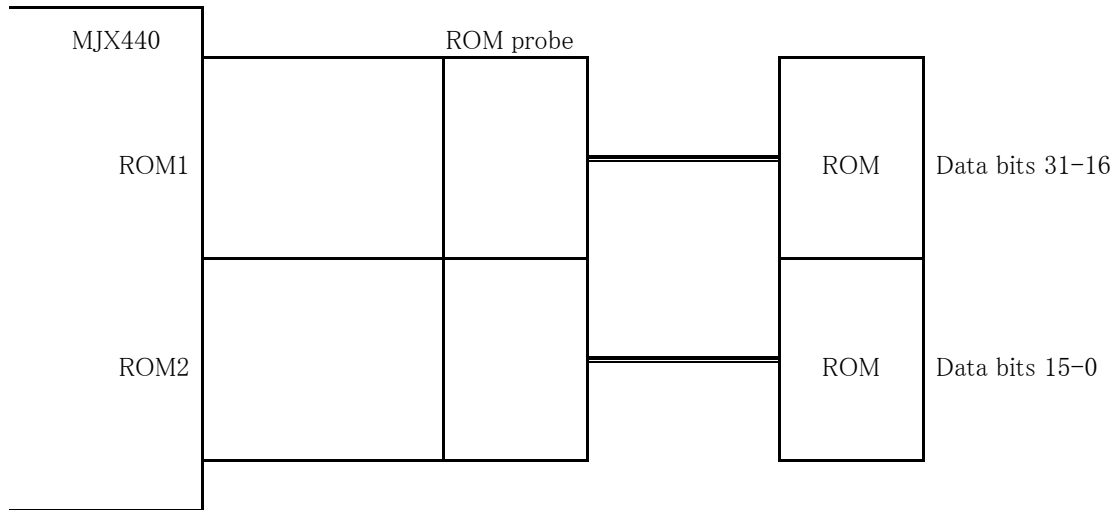


Figure 3-5-9. Connecting a ROM probe (9)

3.4 Connecting an External Trigger Cable

When displaying the status of target system signals on the LED or using MJX440-output trace trigger signals as input into the logic analyzer, connect the EXTERNAL connector from the MJX440 to the target system signals, and to the logic analyzer, using an external trigger cable.

An external trigger cable needs to be connected only when any of the functions mentioned above is used.

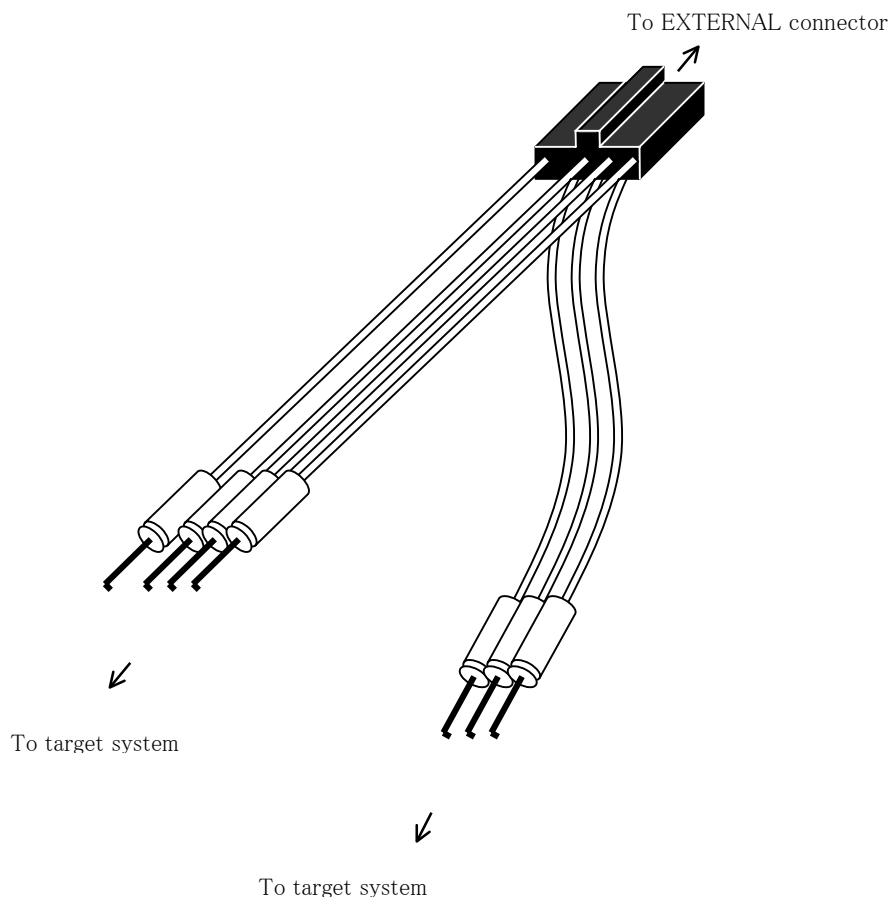


Figure 3-5. Connecting an external trigger cable

Chapte 3. Connecting the Hardware

Details of the signals passing through the external trigger cable are shown below, where signals are color-coded in terms of clip and cable colors:

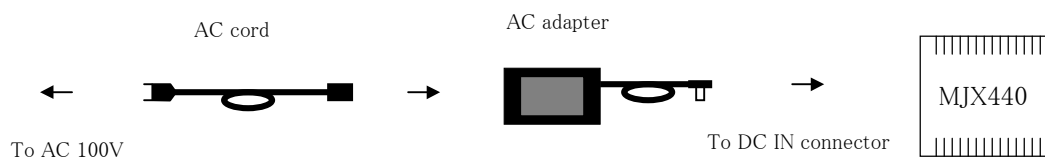
Signal	Color	I/O	Functionality
EXTIN1	Clip: yellow Cable: brown	TTL input	If the connected signal is at the HIGH level, the corresponding LED is lit.* ¹ For the signals EXTIN1 and EXTIN2, the signal status is also recorded during realtime tracing.
EXTIN2	Clip: yellow Cable: red	TTL input	
EXTIN3	Clip: yellow Cable: orange	TTL input	
EXTOUT1	Clip: red Cable: brown	3.3V O.D. output* ²	Generates either the LOW or HIGH level when asserted by the XPIN command of the MJX440 command set.
EXTOUT2	Clip: red Cable: red	3.3V O.D. output	
TRGOUT-	Clip: green Cable: green	3.3V O.D. output	Generates LOW level signals following a trace trigger. This signal can be used as trigger input into the logic analyzer.
GND	Clip: black Cable: black		This signal is connected to the GND outlet of the target system.

*¹ On the MJX440 system unit, signal names are indicated in abbreviation, e.g., EXTIN3→EXI 3.

*² "O.D." stands for "open drain".

3.5 Connecting the Power Supply and Turning the Power On

Aftr connecting all cables, connect the AD cord and the AD adapter to the MJX440, making sure that the power switch for the MJX440 is off.



Turn on the devices in the following sequence:

1. Host
2. MJX440
3. Target system

Similarly, turn off the devices in the following sequence:

1. Target system
2. MJX440
3. Host

【Important】 Turning the power on or off in the incorrect sequence can damage the equipment.

【Important】 Do not connect or disconnect any of the devices when turning the power on.

Chapter 4. Installing the Software

This chapter describes how to install the software for the operation of the MJX440.

- Installing MULTI + MJXSERV, MJXDEB
- Installing the MJXDEB

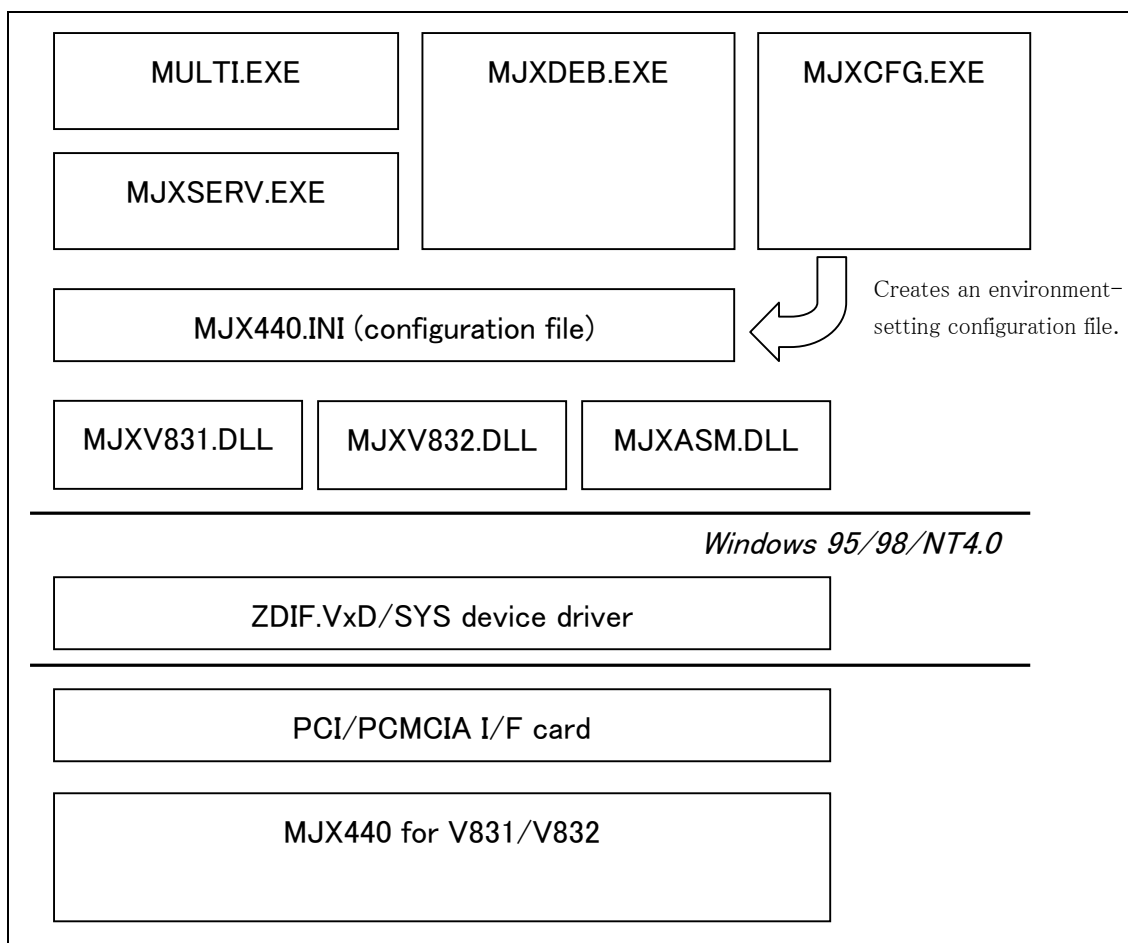
If you purchased **Green Hills Software's integrated development environment MULTI together with MJXDEB, perform the**

MULTI + MJXSERV, MJXDEB installation procedure.

When installing solely MJXDEB, perform the

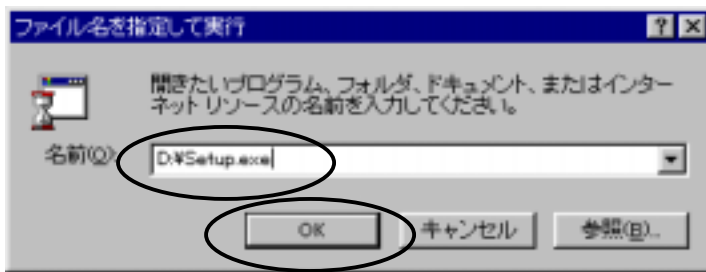
MJXDEB installation procedure.

The MULTI + MJXSERV, MJXDEB option is configured as follows:

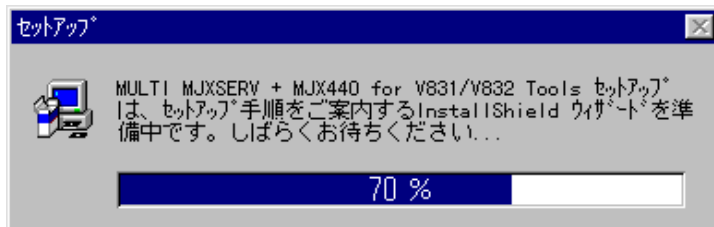


MULTI + MJXSERV, MJXDEB installation

1. Install Green Hills Software's integrated development environment MULTI.
2. Then, install the server program MJXSERV for MULTI.
Load the CD-ROM labeled "MULTI MJXSERV + MJX440 for V831/V832 Tools Disk" in the CD-ROM drive.
Select "Start" and "Run (R) ..." to display the "Run" dialog.
Select "Start" and "Run (R) ..." to display the "Run" dialog.
3. In "Open (O)", specify the setup program Setup.exe (assuming that your CD-ROM drive is D: drive, specify "D:\Setup.exe"). Click on "OK".



4. The setup program starts.



5. When the following dialog appears, click on “Next (N)”:

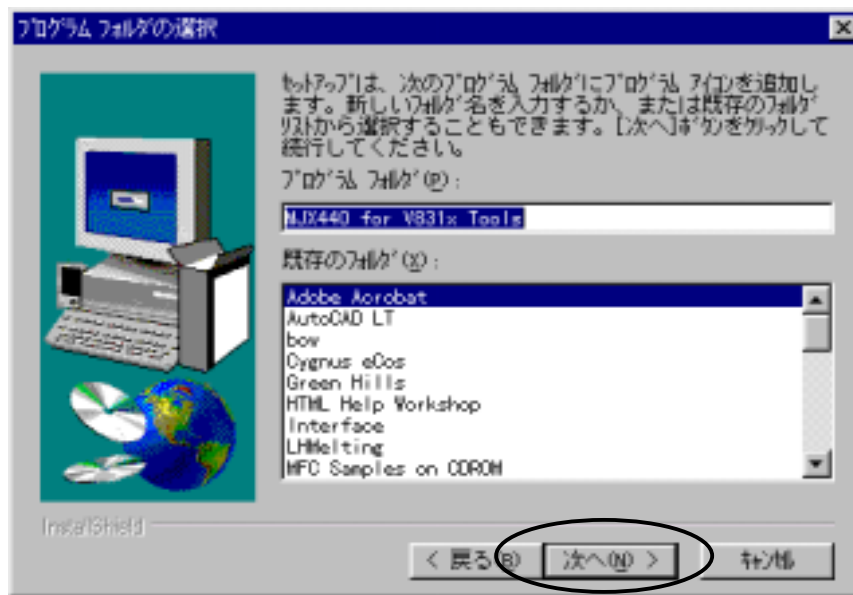


6. When the “Select Destination Directory/Folder” dialog appears, specify the destination by clicking on the “Browse (R)...” button.
Be sure to specify the directly in which MULTI was previously installed.
(Default directory: “C:¥Green”)



Click on “Next (N) >”.

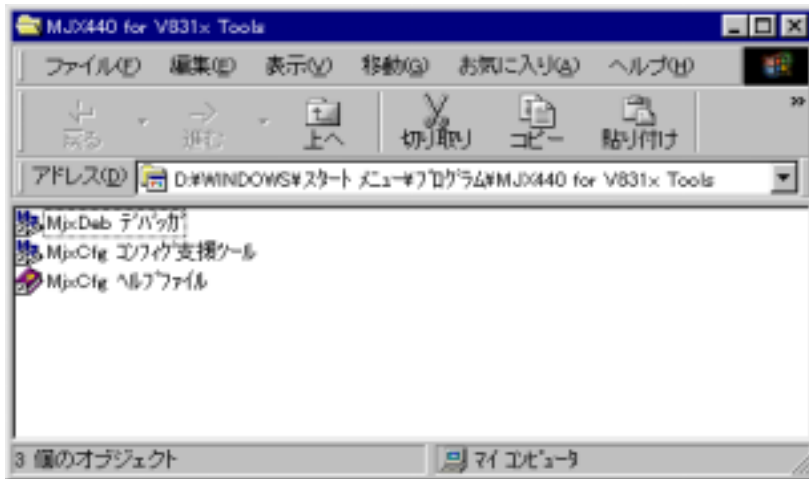
- When the “Select Program Folder” dialog appears, click on the “Next (N) >” button.



When the “Copying file” dialog appears, click on the “Next (N) >” button, which starts the installation process.



8. The installation finishes, displaying program folder
 MJX440 for V83x Tools
 that has been created.

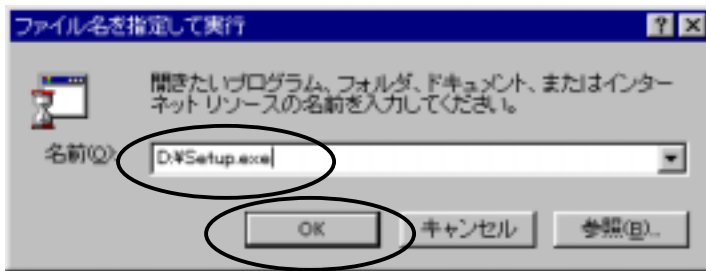


This process installs the following files:

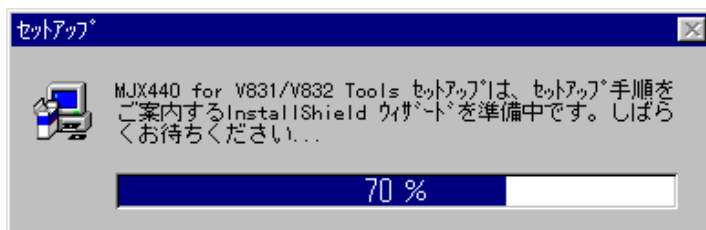
MJXSERV.EXE	MULTI server for MJX440 for V831/V832
MJXASM.DLL	V831/V832 assembler/disassembler library
MJXV831.DLL	V831 library
MJXV832.DLL	V832 library
MJXDEB.EXE	Quick debugger MJXDEB
MJXUSAGE.TXT	MJXDEB help file
MJX440.ICO	MJXDEB icon
MJXCVT.EXE	MJX binary file conversion program MJXCVT
MJXCFG.EXE	Configuration support tool MJXCFG (for creating configuration files)
MJXCFG.HLP	MJXCFG help file
MJXCFG.GID	“
MJXCFG.CNT	“
MSVCRT.DLL	Microsoft Visual C++ runtime library

MJXDEB installation

1. Load the CD-ROM labeled “MJX440 for V831/V832 Tools Disk” in the CD-ROM drive.
Select “Start” and “Run (R) ...” to display the “Run” dialog.
2. In “Open (O)”, specify the setup program Setup.exe (assuming that your CD-ROM drive is D: drive, specify “D:¥Setup.exe”). Click on “OK”.



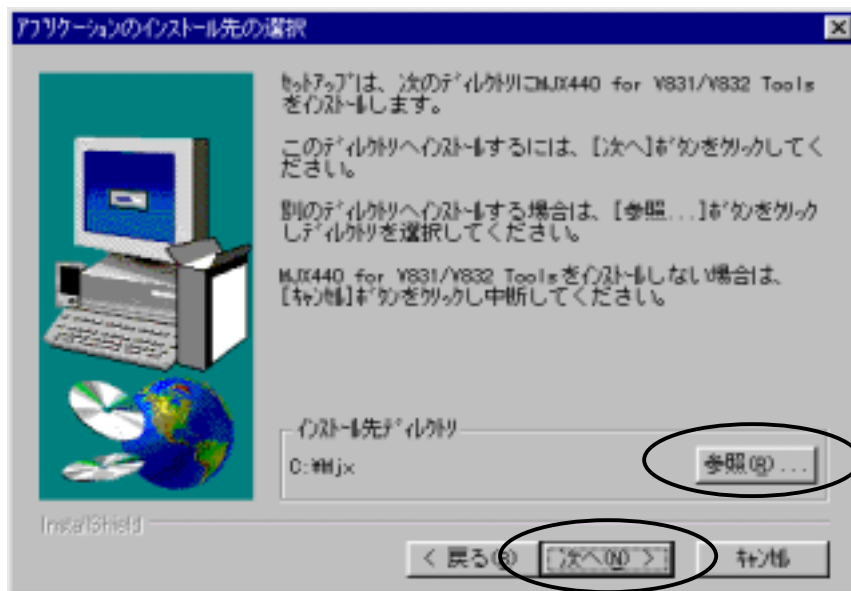
3. The setup program starts.



4. When the following dialog appears, click on “Next (N)>”:

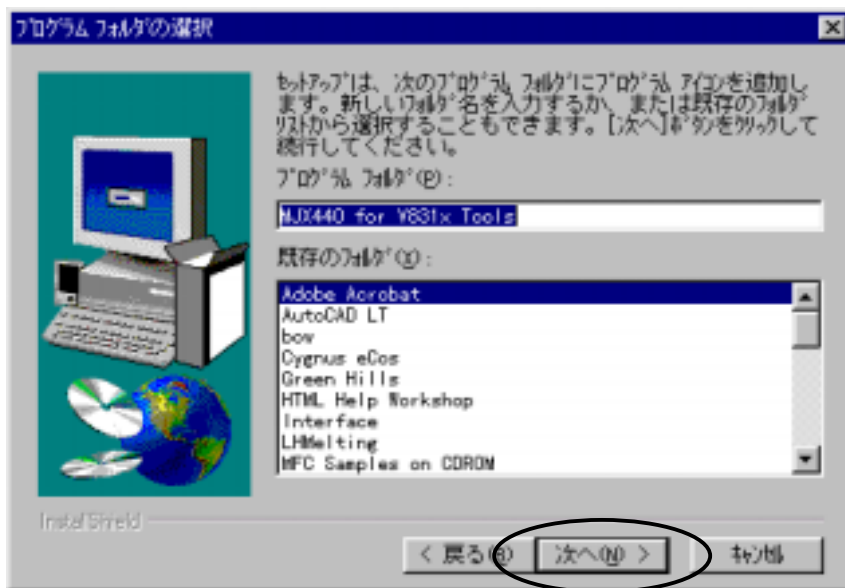


5. When the “Select Destination Directory/Folder” dialog appears, specify the destination by clicking on the “Browse (R)...” button.
(Default directory: “C:¥Mjx”)



Click on “Next (N) >”.

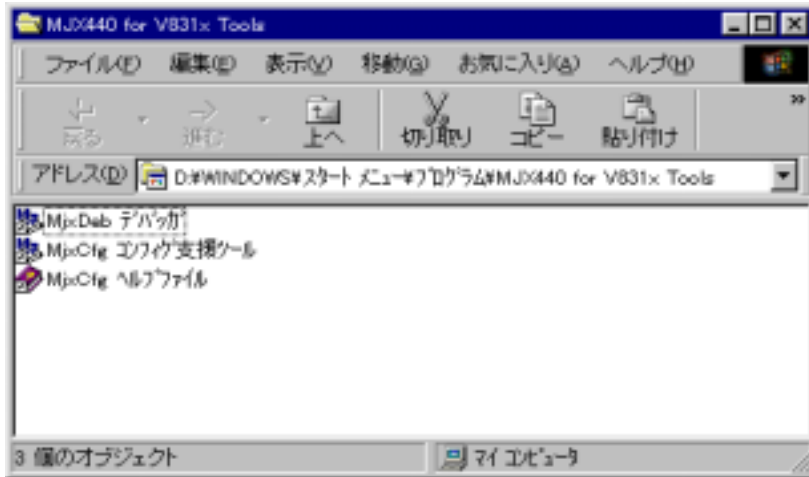
- When the “Select Program Folder” dialog appears, click on the “Next (N) >” button.



- When the “Copying file” dialog appears, click on the “Next (N) >” button, which starts the installation process.



9. The installation finishes, displaying program folder
 MJX440 for V83x Tools
 that has been created.



This process installs the following files:

MJXASM.DLL	V831/V832 assembler/disassembler library
MJXV831.DLL	V831 library
MJXV832.DLL	V832 library
MJXDEB.EXE	Quick debugger MJXDEB
MJXUSAGE.TXT	MJXDEB help file
MJX440.ICO	MJXDEB icon
MJXCVT.EXE	MJX binary file conversion program MJXCVT
MJXCFG.EXE	Configuration support tool MJXCFG (for creating configuration files)
MJXCFG.HLP	MJXCFG help file
MJXCFG.GID	“
MJXCFG.CNT	“
MSVCRT.DLL	Microsoft Visual C++ runtime library

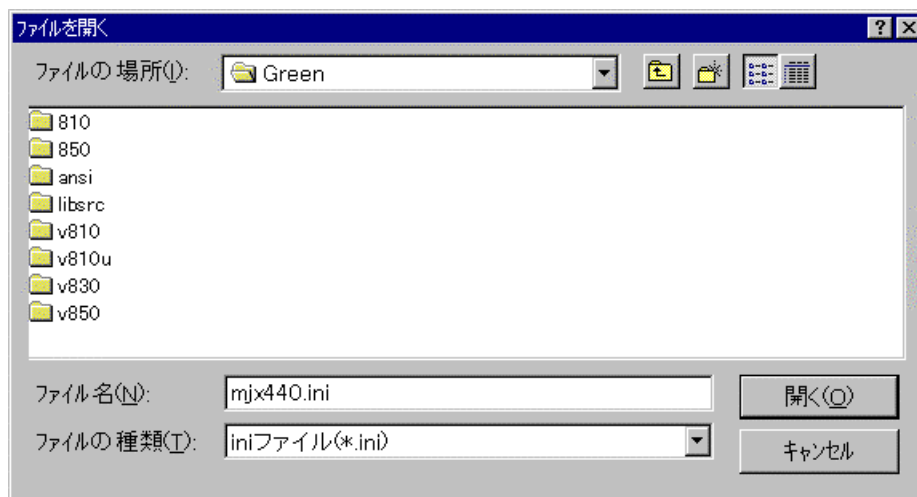
This set of files differs from the files installed under “MULTI + MJXSERV, MJXDEB installation” in that it does not include the MJXSERV.EXE file (MULTI server for MJX440 for V831/V832).

Chapter 5. Setting the MJX440 Environment

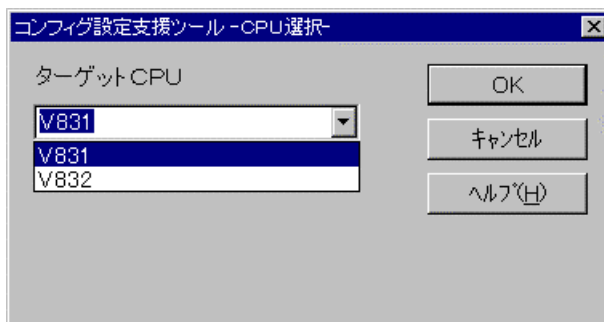
This chapter describes how to set the environment that is necessary for the operation of the MJX440.

The environment for the MJX440 can be set by using the configuration support tool **MJXCFG**. After turning on all the devices, launch MJXCFG from the start menu according to the following procedures:

1. Start menu
2. Program (P)
3. MJX440 for V83X Tools
4. MJXCFG

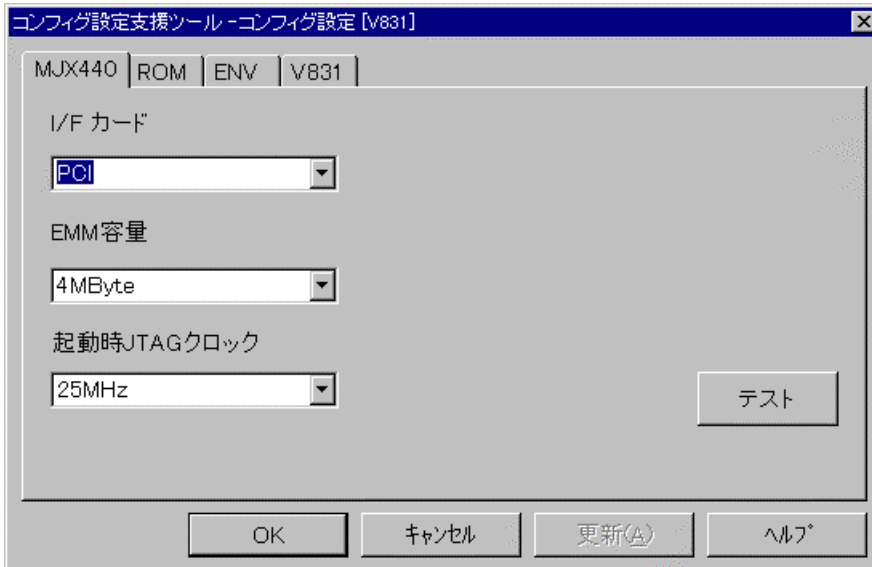


Then, specify the configuration file in which the environmental parameters are to be stored. You should normally specify MJX440.INI.

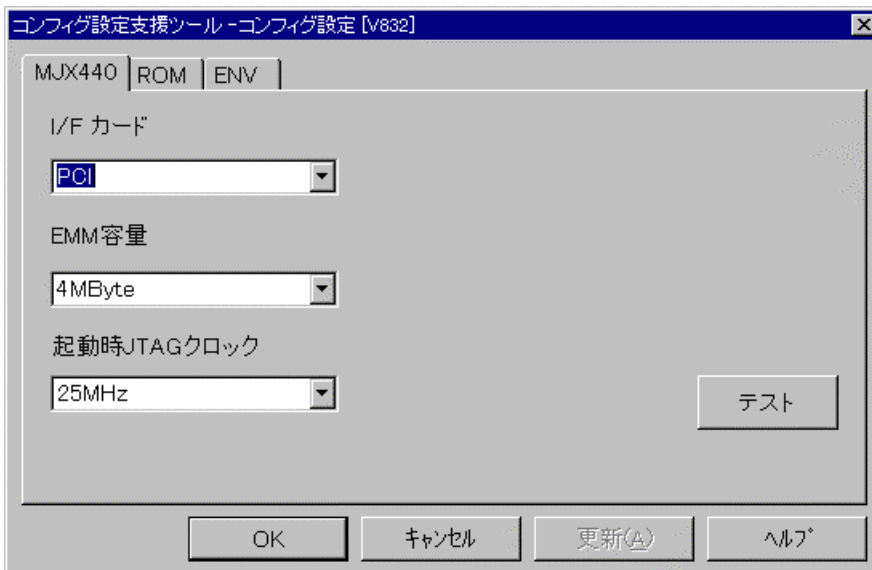


Select the target system CPU.

The following dialog box appears if the target CPU is V831:



Similarly, the following dialog box appears if the target CPU is V832:



When the dialog box appears, set the appropriate fields according to the MJX440 and the target system environment to be used, and then press the **OK** button. See Help information for details of the fields.

Pressing the **Test** button in the MJX440 tag and the ROM tag button causes the system to test the compatibility between the fields that are set, generating an error message in the event of an error.

Chapter 6. Starting and Terminating Software

This chapter describes how to start and terminate software for the operation of the MJX440.

【Note】 Only one software program for the operation of the MJX440 can be executed at a time.

MULTI

To operate the MJX440 using MULTI, after starting the MULTI system, use the following command to remote-connect the MJXSERV. *¹; the target window opens upon successful connection:

```
remote mjxserv
```

In this case, the MJX440.INI file is used as a configuration file. Any other configuration file should be specified explicitly. The following command, for example, uses the MJX832.INI configuration file:

```
remote mjxserv mjx832.ini
```

To terminate the program, enter the *quit* command:

```
quit
```

MJX440 commands can be entered from the target window when the *mjx>* prompt is displayed:

```
mjx> pin
```

*¹ To operate the MJX440 from a builder, specify the server name *mjxserv* and press the **REMOTE** button.

MJXDEB

To operate the MJX440 using MJXDEB, launch **MJXDEB** from the start menu by performing the following sequence of actions:

1. Start menu
2. Program (P)
3. MJX440 for V83X Tools
4. MJXDEB

The default configuration file used is **MJX440.INI**. To use any other configuration file, specify the desired configuration file in the argument. The following command, for example, uses the file **MJX832.INI**:

```
mjxdeb mjx832.ini
```

This command can be entered either from the MS-DOS prompt or by modifying the properties of the shortcut in the start menu.

To terminate the program, simply enter the *q* command:

```
> q
```


Chapter 7. MJX440 Commands

This chapter describes how to use MJX440 commands.

MJX440 commands are commands that are supported by MJXDEB. Most of MJX440 commands can be executed from within the MULTI target window from which MJXSERV is remote-connected.

Following is a list of available MJX440 commands:

ASM	Modifies memory contents in Assembly language.
BATCH	Executes the MJX440 commands that are coded in a batch file.
BP	Displays, sets, and resets breakpoints.
CONFIG	Displays and modifies the MJX440 configuration.
DUMP	Displays memory contents.
EXAMINE	Modifies memory contents.
FILL	Fills memory contents.
GO	Executes a user program.
HELP	Displays help messages.
HISTORY	Displays the results of a realtime tracing.
INIT	Reinitializes the MJX440.
JOURNAL	Writes the results of execution of a command to a file.
LOAD	Downloads a file into memory.
MOVE	Block-transfers memory contents.
PIN	Enables/disables a pin.
PORT	Displays/modifies the contents of an I/O port.
QUIT	Terminates MJXDEB.
REGISTER	Displays/modifies the contents of a register.
STEP	Executes a user program in steps.
TRACE	Displays/sets the realtime trace mode.
UNASM	Disassembles and displays memory contents.
VERSION	Displays software version information.
XPIN	Displays/sets the status of signals serviced by an external trigger cable.

Abbreviation of command names

The name of a command can be shortened to any length as long as it remains distinct from all other commands.

CON	This means the same as CONFIG .
C	This means the same as CONFIG .
HE	This means the same as HELP .
HI	This means the same as HISTORY .
H	This abbreviation, unable to distinguish between HELP and HISTORY , is illegal.

Notes on using MULTI

The following command is ignored in the MULTI target window:

QUIT

Whenever possible, the use of the following command in the MULTI target window should be avoided:

GO
REGISTER (For modifying a register only)
STEP

Execution of these commands can compromise the compatibility between MULTI and MJXSERV. Therefore, functions such as controlling the execution of a user program or overwriting a register should be performed using a MULTI command.

BATCH

Executes the MJX440 commands that are coded in a batch file.

Format:

BATCH *file* Reads the batch file *file* line by line and executes the contents of the file as MJX440 commands.

Arguments:

file Name of the batch file in which MJX440 commands are coded (a text file).

Example:

> **BAT** INIT.TXT (Executes the **INIT.TXT** file as a batch file.)

Remarks:

- The **BATCH** command can be nested up to four nesting levels.
- The **QUIT** in a batch file serves to terminate the **BATCH** command only; it does not terminate **MJXDEB**.

BP

Displays, sets, and resets breakpoints.

Format:

BP [A H S]	Displays a breakpoint.
BP/C <i>num</i>* A* H* S*	Clears a breakpoint.
BP/D <i>num</i>* A* H* S*	Disables a breakpoint.
BP/E <i>num</i>* A* H* S*	Enables a breakpoint.
BP/A <i>addr</i>[,<i>space</i>]	Sets an access breakpoint.
BP/H <i>addr</i>	Displays a hardware breakpoint.
BP/S <i>addr</i>	Displays a software breakpoint.

Arguments:

<i>num</i>	Specifies a breakpoint number.
A	Specifies an access breakpoint.
H	Specifies a hardware breakpoint.
S	Specifies a software breakpoint.
*	Specifies all breakpoints.
A*	Specifies all access breakpoints.
H*	Specifies all hardware breakpoints.
S*	Specifies all software breakpoints.
<i>addr</i>	Specifies a breakpoint address (hexadecimal).
<i>space</i>	M is a memory space specification; I is an I/O space specification (default: M).

Examples:

> BP	(Displays a breakpoint.)
> BP/C *	(Clears all breakpoints.)
> BP/D H*	(Disables a hardware breakpoint.)
> BP/E A*	(Enables an access breakpoint.)
> BP/A 1000	(Assigns an access breakpoint at address 0x1000 in memory space.)
> BP/A 1000,I	(Assigns an access breakpoint at address 0x1000 in I/O space.)
> BP/H 2000	(Assigns a hardware breakpoint at address 0x2000.)
> BP/S 3000	(Assigns a software breakpoint at address 0x3000.)

Remarks:

- A maximum of 4 access breakpoints can be set.
- A maximum of 2 hardware breakpoints can be set in an instruction.
- A maximum of 128 software breakpoints can be set in an instruction.

CONFIG

Displays and modifies the MJX440 configuration.

Format:

CONFIG Displays the current MJX440 configuration.
CONFIG/S *item*=*value* Assigns MJX440 configuration item *item* to *value*.

Arguments:

item Specifies one of the following configuration items:

JCLOCK	JTAG clock
TOPADDR	ROM starting address
EPCAVAIL	Availability of EPC -- for displaying interrupt routine trace information

value Values assigned to configuration items

12.5 or 25	JCLOCK
Hexadecimal address	TOPADDR
0 or 1	EPCAVAIL (0=disable, 1=disable) (SIC)

Examples:

```
> CONFIG (Displays MJX440 configuration information.)
> CONFIG JCLOCK=12.5 (Sets the JTAG clock at 12.5MHz.)
> CONFIG TOPADDR=20000 (Sets the ROM starting address at 0x20000.)
> CONFIG EPCAVAIL=0 (Do not use EPC for displaying interrupt routine trace information.)
```

Remarks:

- The EPCAVAIL option is valid only with V831.

DUMP

Displays memory contents.

Format:

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

Displays memory contents from addresses *addr1* through *addr2*.

Arguments:

- /B Specifies 8 bits.
- /W Specifies 16 bits.
- /L Specifies 32 bits.
- addr1* Starting memory display address (hexadecimal)
- addr2* Ending memory display address (hexadecimal)

Examples:

- > DUMP/B 1000 (Displays 64-byte memory starting at address 0x1000 in increments of 8 bits.)
- > DUMP/W 2000,20FF (Displays memory contents from address 0x2000 through address 0x20FF in increments of 16 bits.)
- > DUMP (Displays the continuation of the previous DUMP command.)

Remarks:

- Displays 64 bytes of memory contents if *addr2* is omitted.
- Displays the continuation of the previous DUMP command if *addr1* is omitted.
- If a size is omitted, the size specified in the previous command execution is applied.
- This command does not display the contents of built-in instruction RAM.

EXAMINE

Modifies memory contents.

Format:

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

Interactively changes the memory contents beginning at address *addr*.

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

Changes the memory contents at address *addr* to data *data*.

Arguments:

- /B** Specifies 8 bits.
- /W** Specifies 16 bits.
- /L** Specifies 32 bits.
- addr*** Starting memory change address (hexadecimal)
- data*** Memory change data (hexadecimal)

Examples:

- > **EXAMINE/B 1000=55** (Changes the memory contents at address 0x1000 to 8-bit data 0x55.)
 - > **EXAMINE/W 3000=1,2,3** (Changes the memory contents starting from address 0x3000 to 16-bit data 0x0001, 0x0002, and 0x0003.)
 - > **EXAMINE/L 2000** (Interactively changes the memory contents starting from address 0x2000.)
- ```
00002000 00000000 11223344
00002004 00000000 55667788
00002008 00000000 . (Terminates when a period is encountered.)
```

Remarks:

- In the interactive mode, this command terminates when a period is entered.
- If a size is omitted, the size specified in the previous command execution is applied.
- This command does not change the contents of built-in instruction RAM

# FILL

Fills memory contents.

Format:

`FILL[/B|/W|/L] addr1,addr2,data`Fills the memory from address *addr1* through address *addr2* with data *data*.

Arguments:

`/B` Specifies 8 bits.  
`/W` Specifies 16 bits.  
`/L` Specifies 32 bits.  
*addr1* Memory fill starting address (hexadecimal)  
*addr2* Memory fill ending address (hexadecimal)  
*data* Fill data (hexadecimal)

Examples:

> `FILL/B 0,3FF,FF` (Fills the memory from 0x0 through 0x3FF with 8-bit data 0xFF.)  
 > `FILL/W 1000,1FFF,0` (Fills the memory from 0x1000 through 0x1FFF with 16-bit data 0x000.)

Remarks:

- This command cannot fill built-in instruction RAM.
- This command cannot fill more than 8M bytes of memory (a limit item).

# GO

Executes a user program.

Format:

`GO [addr]`                      Executes a user program from address *addr*

Arguments:

*addr*      User program starting address (hexadecimal)

Examples:

> `GO 1000`                      (Executes the user program starting from address 0x1000.)

> `GO`                              (Executes a user program from the current PC.)

Remarks:

- This command should not be executed from within a MULTI target window.
- When the argument *addr* is omitted, the command executes the user program from the PC.
- If the program being executed by this command fails to stop at a breakpoint, it can still be stopped by pressing the space key.

# HELP

Displays help messages.

Format:

|                 |                                                     |
|-----------------|-----------------------------------------------------|
| HELP            | Displays a list of MJX440 commands.                 |
| HELP <i>cmd</i> | Displays a help message on the command <i>cmd</i> . |

Arguments:

*cmd* MJX440 command

Examples:

|            |                                            |
|------------|--------------------------------------------|
| > HELP     | (Displays a list of MJX440 commands.)      |
| > HELP ASM | (Displays a help message on ASM commands.) |

Remarks:

- Command help messages can also be displayed by entering the string “c ommand name ?”.
- |         |                                            |
|---------|--------------------------------------------|
| > ASM ? | (Displays a help message on ASM commands.) |
|---------|--------------------------------------------|

# HISTORY

Displays the results of a realtime tracing.

Format:

|                                                  |                                                                                                                  |
|--------------------------------------------------|------------------------------------------------------------------------------------------------------------------|
| <b>HISTORY</b>                                   | Displays a range of packet numbers in which realtime trace results are stored.                                   |
| <b>HISTORY/D</b> [ <i>start</i> [, <i>end</i> ]] | Displays the results of realtime tracing from packet number <i>start</i> through <i>end</i> in disassembling.    |
| <b>HISTORY/P</b> [ <i>start</i> [, <i>end</i> ]] | Displays the results of realtime tracing from packet number <i>start</i> through <i>end</i> in units of packets. |

Arguments:

|                     |                                          |
|---------------------|------------------------------------------|
| <b><i>start</i></b> | Starting display packet number (decimal) |
| <b><i>end</i></b>   | Ending display packet number (decimal)   |

Examples:

|                          |                                                                                                   |
|--------------------------|---------------------------------------------------------------------------------------------------|
| > <b>HISTORY</b>         | (Displays a range of packet numbers in which realtime trace results are stored.)                  |
| > <b>HISTORY/D 0,70</b>  | (Displays the results fo realtime tracing from packet numbers 0 through 70 in disassembling.)     |
| > <b>HISTORY/P -10,0</b> | (Displays the results of realtime tracing from packet numbers -10 through 0 in units of packets.) |

Remarks:

- Displays 16 packets if *end* is omitted.
- Displays thje continuation of the previous **HISTORY** command if *start* is omitted.

- Specify *start and end* in terms of offset values, for which the packet number 0 is the starting value. Depending on the trace conditions that are employed, the packet number 0 corresponds with the following program location point:

Begin monitor: program starting point

End monitor: program end point

Begin trigger: trace trigger point

End trigger: trace trigger point

Mid-trigger: trace trigger point

Inner trigger: trace starting trigger point

(Limitation on V831 disassembly display): Packets with undefined addresses cannot be displayed in a disassembled form even when their packet numbers are specified.

# INIT

Reinitializes the MJX440.

Format:

**INIT** Reinitializes the MJX440.

Remarks:

- This command also resets the CPU.

# JOURNAL

Writes the results of execution of a command to a file.

Format:

**JOURNAL**[/A/W] *file*[,*mode*][,*echo*]

Writes the results of command execution to file *file*.

**JOURNAL**/E

Terminates file output and closes the file.

Arguments:

|                      |                                                    |
|----------------------|----------------------------------------------------|
| <b>/A</b>            | Append output specification                        |
| <b>/W</b>            | New output specification (default)                 |
| <b><i>file</i></b>   | Output file name specification                     |
| <b><i>mode</i></b>   | Output mode specification                          |
| <b>IN</b>            | Writes commands only.                              |
| <b>OUT</b>           | Writes command execution results only.             |
| <b>ALL (default)</b> | Writes both command and command execution results. |
| <b><i>echo</i></b>   | Echo mode specification                            |
| <b>OFF</b>           | No screen display of file output                   |
| <b>ON (default)</b>  | Screen display of file output                      |

Examples:

- > **JOURNAL TEST.TXT** (Writes command execution results to the TEST.TXT file.)
- > **JOURNAL/E** (Terminates the file output and closes the file.)
- > **JOURNAL/A TEST.TXT** (Appends command execution results to the TEST.TXT file.)

Remarks:



# LOAD

Downloads a file into memory.

Format:

**LOAD** *file*[,*offset*] Downloads into memory file *file* of MJX binary\*<sup>1</sup>, S-record, Intel hex, or COFF format.

**LOAD/R** *file*[,*offset*] Down loads into memory MJX binary file *file* through emulation memory.

Arguments:

*file* File to be downloaded

*offset* Offset address (default)

Examples:

> **LOAD** PROG1.ABS (Downloads the file PROG1.ABS into memory.)

> **LOAD** PROG1.ABS,2000 (Downloads the file PROG1.ABS into memory.)

> **LOAD/R** PROG2.MJX (Downloads the file PROG2.MJX into memory through emulation memory.)

Remarks:

- The file format is automatically recognized.
- If a file name extension is omitted, the extension **.mjx** is supplied by default.
- This command cannot download a file into built-in instruction RAM.
- The **LOAD/R** command may not be able to download an MJX binary file correctly if the file contains records that point to emulation memory.

---

\*<sup>1</sup> For a description of MJX binary files, see Chapter 8, “Rapid Downloading”.

# MOVE

Block-transfers memory contents.

Format:

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

Block-transfers memory contents from addresses *addr1* through *addr2* to address *addr3*.

Arguments:

- /B** Specifies 8 bits.
- /W** Specifies 16 bits.
- /L** Specifies 32 bits.
- addr1*** Source memory starting address (hexadecimal)
- addr2*** Source memory ending address (hexadecimal)
- addr3*** Destination memory address (hexadecimal)

Example:

> **MOVE 1000,10FF,2000** (Block-transfers memory contents from the addresses 0x1000 through 0x10FF to the address 0x2000.)

Remarks:

- This command cannot block transfer the contents of built-in instruction RAM.
- This command cannot block transfer memory contents exceeding 8M bytes (a limit item).

# PIN

Enables/disables a pin.

Format:

**PIN** Displays pin enabled/disabled status.  
**PIN *pinname*=EN|DI** Enables or disables pin *pinname*.

Arguments:

*pinname* (pin name); specify one of the following pins:

**INTP00**

**INTP01**

**INTP02**

**INTP03**

**INTP10**

**INTP11**

**INTP12**

**INTP13**

**NMI**

**HLDRQ**

**RESET**

**EN** Enable

**DI** Disable

Examples:

> **PIN** (Displays the pin enabled/disabled status.)

> **PIN INTP11=DI** (Disables the pin **INT11**.)

Remarks:

# PORT

Displays/modifies the contents of an I/O port.

Format:

**PORT[/B|/W|/L] *addr***      Displays the contents of the I/O port at address *addr* .  
**PORT[/B|/W|/L] *addr=data***      Converts the contents of the I/O port at address *addr* to data *data*.

Arguments:

**/B**      Specifies 8 bits.  
**/W**      Specifies 16 bits.  
**/L**      Specifies 32 bits.  
***addr***    I/O port address (hexadecimal)  
***data***    I/O port change data (hexadecimal)

Examples:

> **PORT/L 2000**      (Displays the contents of the I/O port starting at the address 0x2000.)  
> **PORT/W 1000=55**      (Converts the I/O port at the address 0x1000 to 16-bit data at 0x0055.)

Remarks:

# QUIT

Terminates MJXDEB.

Format:

**QUIT** Terminates MJXDEB.

Remarks:

- The **QUIT** command in a batch file terminates the **BATCH** command only; it does not terminate **MJXDEB**.

# REGISTER

Displays/modifies the contents of a register.

Format:

**REGISTER**                                    Displays the contents of a register.  
**REGISTER *reg*=*data***                    Changes the contents of register *reg* to data *data*.

Arguments:

***reg***            Register name specification  
                  See Appendix G, "List of Register Names".  
***data***           Register change data (hexadecimal)

Examples:

> **REGISTER**                                    (Displays the contents of a register.)  
> **REGISTER PC=2000**                    (Changes the PC to 0x2000.)

Remarks:

# STEP

Executes a user program in steps.

Format:

STEP [*num*]                      Executes a user program by *num* steps.

Arguments:

*num*            Step count specification (decimal; default: 1)

Examples:

> STEP                              (Executes a user program by 1 step.)  
> STEP 10                            (Executes a user program by 10 steps.)

Remarks:

# TRACE

Displays/sets the realtime trace mode.

Format:

|                                                            |                                                                   |
|------------------------------------------------------------|-------------------------------------------------------------------|
| TRACE                                                      | Displays all trace conditions.                                    |
| TRACE/A <i>addr</i> [, <i>space</i> ]                      | Sets data tracing.                                                |
| TRACE/C <i>addr</i>                                        | Clears data tracing.                                              |
| TRACE/C *                                                  | Clears all data tracing.                                          |
| TRACE/M <i>mode</i> [, <i>mask</i> ]                       | Sets the trace mode.                                              |
| TRACE/I BM[, <i>length</i> ]                               | Sets execution trace conditions on the begin monitor.             |
| TRACE/I EM                                                 | Sets execution trace conditions on the end monitor.               |
| TRACE/I BT, <i>taddr</i>                                   | Sets execution trace conditions on the begin trigger.             |
| TRACE/I ET, <i>taddr</i>                                   | Sets execution trace conditions on the end trigger.               |
| TRACE/I MT, <i>taddr</i>                                   | Sets execution trace conditions on the mid trigger.               |
| TRACE/I IT, <i>saddr</i> , <i>eaddr</i> [, <i>length</i> ] | Sets execution trace conditions on the inner trigger (V832 only). |
| TRACE/I DI                                                 | Suspends execution trace conditions.                              |
| TRACE/R                                                    | Disables all trace conditions.                                    |

Arguments:

|              |                                                                                         |
|--------------|-----------------------------------------------------------------------------------------|
| <i>addr</i>  | Access address specification for data tracing (hexadecimal)                             |
| <i>space</i> | <b>M: memory space specification; I: I/O spece specification (default: M)</b>           |
| <i>mode</i>  | Trace mode specification; specify one of the following:                                 |
| RT           | Regular mode                                                                            |
| FL           | Full mode                                                                               |
| <i>mask</i>  | Trace cause specification; specify in hexadecimal in the following bits (default: 1FF0) |
| 0010         | Exception event                                                                         |
| 0020         | Interrupt event                                                                         |
| 0040         | Branching by a conditional branch instruction                                           |
| 0080         | Branching by a relative PC branch instruction                                           |
| 0100         | Branching by the JAL instruction                                                        |
| 0200         | Branching by the RETI instruction (branch to PC)                                        |
| 0400         | Branching by the RETI instruction (branch from PC)                                      |



**0800**                    Branching by register indirect branch instruction (branch to PC)

**1000**                    Branching by register indirect branch instruction (branch from PC)

*length*    Tracing word length specification (decimal; default: 131072)

*taddr*      Tracing trigger address specification (hexadecimal)

*saddr*      Trace-initiating trigger address specification (hexadecimal)

*eaddr*      Trace-suspending trigger address specification (hexadecimal)

Examples:

> TRACE                    (Displays all trace conditions.)

> TRACE/A 1000            (Sets data tracing that traces access to the address 0x1000.)

> TRACE/C \*                (Clears all data tracing.)

> TRACE/M FL              (Sets the trace mode in the full mode.)

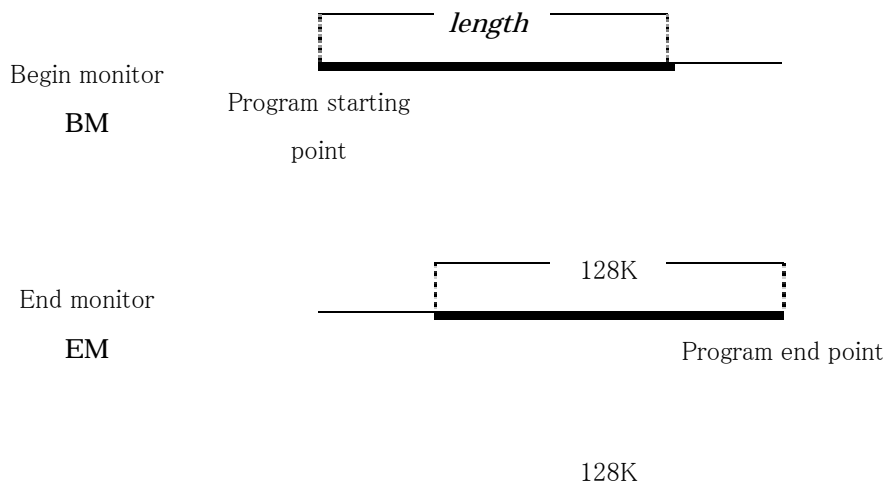
> TRACE/I BT,1000        (Sets execution trace conditions on the begin trigger; sets the address of the trace trigger at 0x1000.)

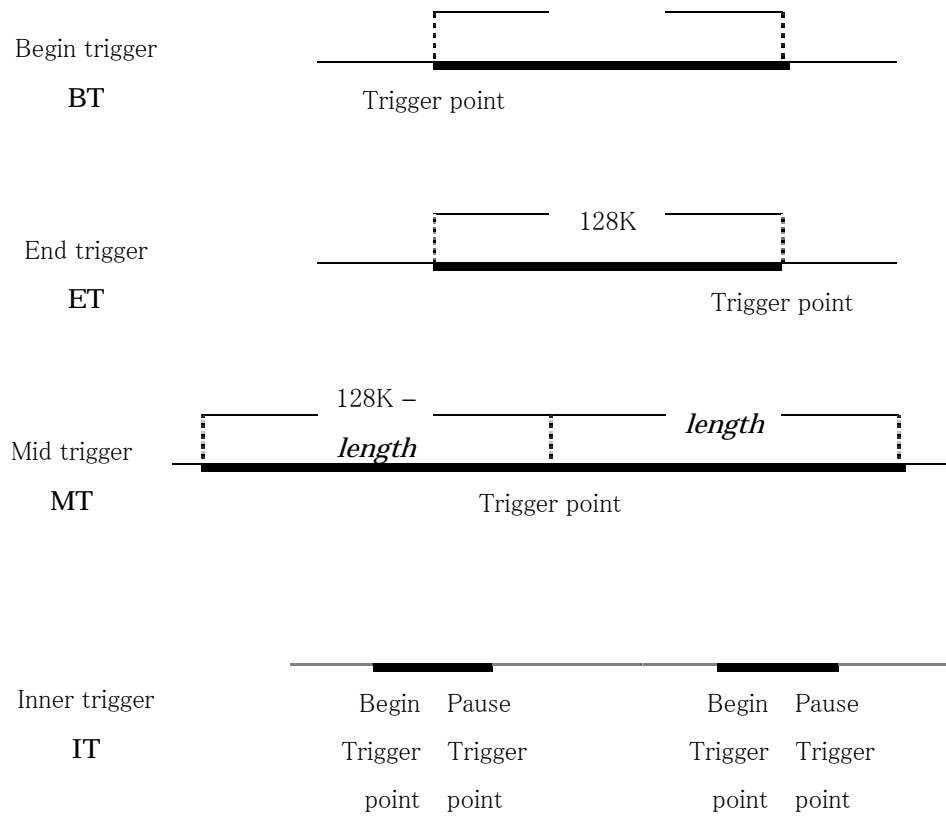
> TRACE/R                 (Disables all trace conditions.)

Remarks:

- The full trace mode FL disables the data tracing.
- The number of packets that can be stored in trace memory depends on the specific instruction that is executed.

Tracing range:





# UNASM

Disassembles and displays memory contents.

Format:

**UNASM** [*addr* [, *count*]]      Displays the memory contents including *count* instructions starting from address *addr*, in the disassembled form.

Arguments:

*addr*      Disassembled display memory starting address (hexadecimal)  
*count*      Number of disassembled instructions for display (decimal; default: 16)

Examples:

> **UNASM 1000**      (Displays the memory contents including 16 instructions starting from the address 0x1000 in the disassembled form.)  
> **UNASM**      (Displays the continuation of the previous UNASM command.)

Remarks:

- Displays memory contents including 16 instructions in the disassembled form when *count* is omitted.
- Displays the continuation of the previous UNASM command when *addr* is omitted.
- This command cannot display the contents of built-in instruction RAM in the disassembled form.

# VERSION

Displays software version information.

Format:

VERSION

Displays software version information.

## XPIN

Displays/sets the status of signals serviced by an external trigger cable.

Format:

**XPIN** Displays the status of signals serviced by an external trigger cable.

**XPIN *ch,level*** Sets output signal *ch* from the external trigger cable to *level*.

Arguments:

*ch* Pin specification for external trigger cable output signals

1 EXTOUT1

2 EXTOUT2

*level* Output pint level specification

0 LOW level

1 HIGH level

Examples:

> **XPIN** (Displays the status of signals serviced by an external trigger cable.)

> **XPIN 1,0** (Sets the external trigger cable output signal EXTOUT1 to the LOW level.)

Remarks:

- The output signals are negative logic.

## Chapter 8. Rapid Downloading

This chapter describes the procedures for effecting rapid downloading.

By creating an MJX binary file, you can rapidly download programs at the following rates:

- N-Wire cable connection: 440Kbytes/second (RAM region)
- ROM in-circuit connection: 4Mbytes/second (ROM region)

MJX binary files can be created by using the file conversion program MJXCVT, which converts S-records into an MJX binary file. A description of how to use this program, executed from the MS-DOS prompt, follows:

### Using MJXCVT

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

-o offset      Add the offset address to the output file.

infile        Input file name

outfile       Output file name (default: *infile* with the extension .mjx)

### Downloading an MJX binary file

The MJX binary file created by MJXCVT can be downloaded by issuing the LOAD command in the MJX440 command set:

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

## Appendix A. Specifications

|                             |                                                                                                                               |
|-----------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| System unit dimensions      | 31mm(H) x 149mm(W) x 186mm(D)                                                                                                 |
| Weight                      | 450g                                                                                                                          |
| Power supply (AC adapter)   | Input: AC100~240V 50Hz/60Hz<br>Output: DC 5V $\pm$ 5% 3.0A                                                                    |
| ROM probe cable             | 300mm                                                                                                                         |
| External trigger cable      | 300mm                                                                                                                         |
| Operating temperature range | 0°C~35°C                                                                                                                      |
| Storage temperature range   | -10°C~55°C                                                                                                                    |
| Ambient humidity range      | 30%~85%                                                                                                                       |
| Compatible ROM              | See Appendix D, "ROM Probes".                                                                                                 |
| Number of ROM chips         | 1, 2, and 4 8-bit chips; 1 and 2 16-bit chips                                                                                 |
| Emulation memory size       | 4Mbytes (16Mb maximum)                                                                                                        |
| Access time                 | 50nsec from a CS                                                                                                              |
| Interface                   | Parallel (PCI or PCMCIA card)                                                                                                 |
| Target interface            | N-Wire connector<br>ROM socket                                                                                                |
| Supported CPU               | V831, V832                                                                                                                    |
| Downloading rate            | 440Kbytes/sec (N-Wire),<br>4Mbytes/sec (ROM in-circuit connection)                                                            |
| Supported debugger          | Green Hills MULTI                                                                                                             |
| Breakpoint function         | Instruction fetch   x 2<br>Software           x 100<br>Memory access     x 4                                                  |
| Trace function              | Trace memory:     48bit $\times$ 128K<br>Trace clock:       66MHz max<br>Trace conditions: 6 sets<br>Time stamp:        32bit |
| Limits                      | See Appendix B, "Limits on Target Systems".                                                                                   |

## Appendix B. Limits on Target Systems

In order to use the MJX440, be sure that your target system meets the following conditions:

- The target system is equipped with connectors in compliance with N-Wire specifications.
- The N-Wire clock (the DCK signal) is slower than the bus clock in the target system.

In order to use a target system with a ROM in-circuit connection, the target system must meet the following conditions:

- The target system is equipped with a ROM socket.
- The target system contains ROM in a bank configuration.
- If multiple ROM chips are installed, all ROM address signals in the target system are identical.



## Appendix C. N-Wire Connector

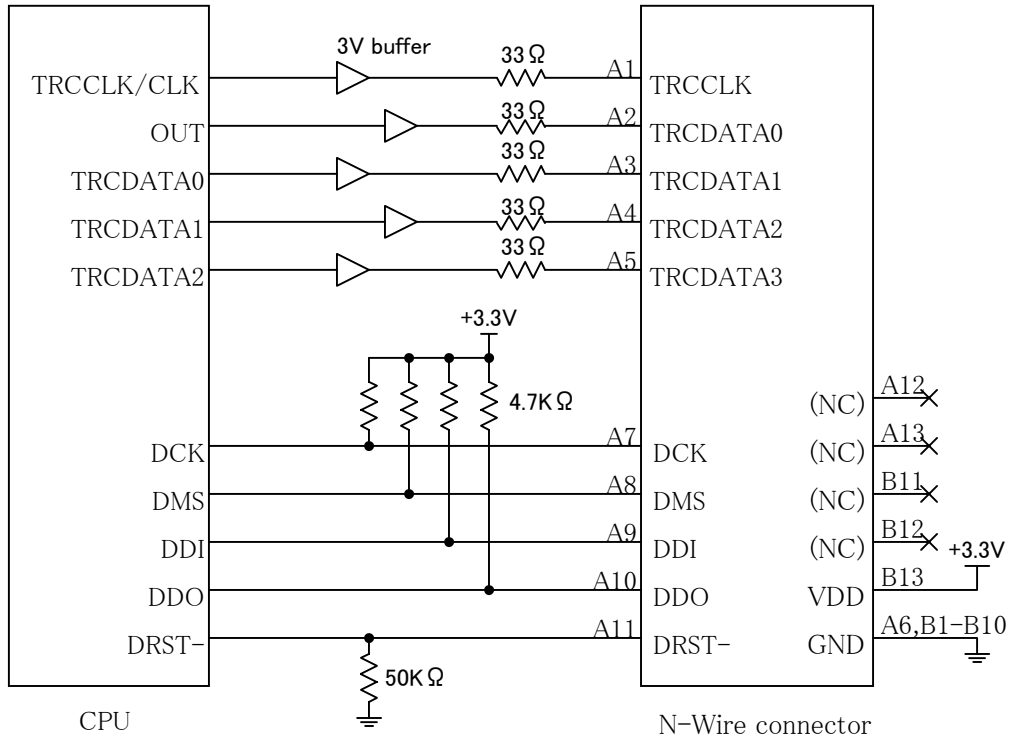
### Pin assignment

|          |     |     |     |
|----------|-----|-----|-----|
| 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 |

### Recommended connectors

- KELCorporation, 8830E-026-170S (straight)
- KELCorporation, 8830E-026-170L (right angle)

Recommended circuitry



Notes

- The patterns between the CPU and the N-Wire connector should be as short as possible (less than 100 mm).
- The patterns between TRCCLK/CLKOUT~TRCCLK and between DCK~DCK should be GND-shielded.
- The MJX400 uses B13 VDD solely for detecting the power supply status; therefore, it does not require any greater power capacity.

Recommended 3V buffer products:

- TI Corporation, SN74LVC541A
- Toshiba, TC74LCX541F

## Appendix D. ROM Probes

ROM probe board J-101A jumper settings

| ROM plug          | Jumper settings                      | ROM size                       | Supported ROM                                                                                                                            | ROM maker                                                                                         |
|-------------------|--------------------------------------|--------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------|
| 32pin<br>(27010)  | JMP1 1-2 shorted<br>JMP2 2-3 shorted | 128K x 8bit<br>(0x20000byte)   | HN27C101AG<br>$\mu$ PD27C1001AD<br>TC571000D<br>TC571000AD<br>TC57H1000AD<br>M5M27C101K<br>MBM27C1001-nnZ<br>27010<br>27C010<br>Am27C010 | Hitachi<br>NEC<br>Toshiba<br>Toshiba<br>Toshiba<br>Mitsubishi<br>Fujitsu<br>intel<br>intel<br>AMD |
| 32pin<br>(27020)  | JMP1 1-2 shorted<br>JMP2 1-2 shorted | 256K x 8bit<br>(0x40000byte)   | $\mu$ PD27C2001D<br>M5M27C201K<br>Am27C020                                                                                               | NEC<br>Mitsubishi<br>AMD                                                                          |
| 32pin<br>(27040)  | JMP1 1-2 shorted<br>JMP2 1-2 shorted | 512K x 8bit<br>(0x80000byte)   | HN27C4001G<br>$\mu$ PD27C4001DZ<br>TC574000D<br>TC574000DI<br>M5M27C401K<br>MBM27C4001-nnZ<br>27040<br>Am27C040                          | Hitachi<br>NEC<br>Toshiba<br>Toshiba<br>Mitsubishi<br>Fujitsu<br>intel<br>AMD                     |
| 32pin<br>(27080)  | JMP1 1-2 shorted<br>JMP2 1-2 shorted | 1024K x 8bit<br>(0x100000byte) | Am27C080                                                                                                                                 | AMD                                                                                               |
| 32pin<br>(271000) | JMP1 2-3 shorted<br>JMP2 2-3 shorted | 128K x 8bit                    | HN27C301AG<br>$\mu$ PD27C1000AD<br>TC571001D<br>TC571001AD<br>TC57H1001AD<br>M5M27C100K<br>MBM27C1000-nnZ                                | Hitachi<br>NEC<br>Toshiba<br>Toshiba<br>Toshiba<br>Mitsubishi<br>Fujitsu                          |

Appendix D. ROM Probes

ROM probe board J-102A jumper settings

| ROM plug                  | Jumper settings  | ROM size                        | Supported ROM               | ROM maker      |
|---------------------------|------------------|---------------------------------|-----------------------------|----------------|
| 40pin<br>(27C4000 16bit)  | JMP1 1-2 shorted | 256K x 16bit<br>(0x80000byte)   | HN27C4000G<br>Am27C400      | Hitachi<br>AMD |
| 42pin<br>(27C8000 16bit)  | JMP1 2-3 shorted | 512K x 16bit<br>(0x100000byte)  | $\mu$ PD27C8000<br>Am27C800 | NEC<br>AMD     |
| 42pin<br>(27C16000 16bit) | JMP1 1-2 shorted | 1024K x 16bit<br>(0x200000byte) |                             |                |

ROM probe board J-103A jumper settings

| ROM plug          | Jumper settings  | ROM size                      | Supported ROM                                                                                                                       | ROM maker                                                                       |
|-------------------|------------------|-------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| 40pin<br>(271024) | JMP1 2-3 shorted | 64K x 16bit<br>(0x20000byte)  | HN27C1024HG<br>$\mu$ PD27C1024D<br>$\mu$ PD27C1024AD<br>TC57H1024D<br>TC57H1024AD<br>MBM27C1024-nnZ<br>27210<br>27C210<br>Am27C1024 | Hitachi<br>NEC<br>NEC<br>Toshiba<br>Toshiba<br>Fujitsu<br>intel<br>intel<br>AMD |
| 40pin<br>(272048) | JMP1 1-2 shorted | 128K x 16bit<br>(0x40000byte) | Am27C2048                                                                                                                           | AMD                                                                             |
| 40pin<br>(274096) | JMP1 1-2 shorted | 256K x 16bit<br>(0x80000byte) | HN27C4096G<br>HN27C4096HG<br>HN27C4096AG<br>HN27C4096AHG<br>TC574096D<br>MBM27C4096-nnZ<br>27240<br>Am27C4096                       | Hitachi<br>Hitachi<br>Hitachi<br>Hitachi<br>Toshiba<br>Fujitsu<br>intel<br>AMD  |

Appendix D. ROM Probes

ROM probe board J-104A jumper settings

| ROM plug                 | Jumper settings  | ROM size                       | Supported ROM               | ROM maker      |
|--------------------------|------------------|--------------------------------|-----------------------------|----------------|
| 40pin<br>(27C4000 8bit)  | JMP1 1-2 shorted | 512K x 8bit<br>(0x80000byte)   | HN27C4000G<br>Am27C400      | Hitachi<br>AMD |
| 42pin<br>(27C8000 8bit)  | JMP1 2-3 shorted | 1024K x 8bit<br>(0x100000byte) | $\mu$ PD27C8000<br>Am27C800 | NEC<br>AMD     |
| 42pin<br>(27C16000 8bit) | JMP1 1-2 shorted | 2048K x 8bit<br>(0x200000byte) |                             |                |

## Appendix E. Corresponding ROM Pin Assignment

|      |    |    |      |
|------|----|----|------|
| 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

Appendix E. Corresponding ROM Pin Assignment

|      |    |    |      |
|------|----|----|------|
| 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

Appendix E. Corresponding ROM Pin Assignment

|       |    |    |           |
|-------|----|----|-----------|
| 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



Appendix E. Corresponding ROM Pin Assignment

|       |    |    |      |
|-------|----|----|------|
| 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

## Appendix F. LEDs

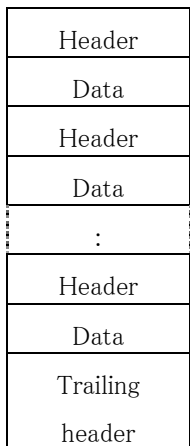
|              |                                                                |
|--------------|----------------------------------------------------------------|
| PWR (green)  | Lit when power is on                                           |
| N-Wire (red) | Lit when power to N-Wire connector is on                       |
| EXI1 (red)   | Lit when external trigger cable EXTIN1 signal is at HIGH level |
| EXI2 (red)   | Lit when external trigger cable EXTIN2 signal is at HIGH level |
| EXI3 (red)   | Lit when external trigger cable EXTIN3 signal is at HIGH level |
| ROM1 (red)   | See Figures 3-5-1 through 3-5-9.                               |
| ROM2 (red)   | Same as the above                                              |
| ROM3 (red)   | Same as the above                                              |
| ROM4 (red)   | Same as the above                                              |

## Appendix G. List of Registers

|     |     |       |
|-----|-----|-------|
| 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 |       |

# Appendix H. MJX Binary File

Binary file organization



Header (16 bytes)

|     |     |     |    |      |      |      |    |    |    |    |      |      |      |      |    |
|-----|-----|-----|----|------|------|------|----|----|----|----|------|------|------|------|----|
| 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: data byte length (MSB)

len2: data byte length

len3: data byte length (LSB)

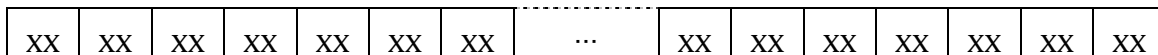
adr1: logical address (MSB)

adr2: logical address

adr3: logical address

adr4: logical address (LSB)

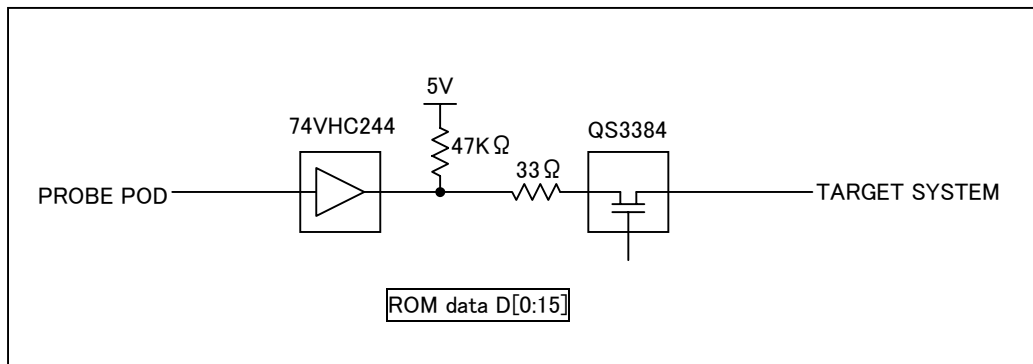
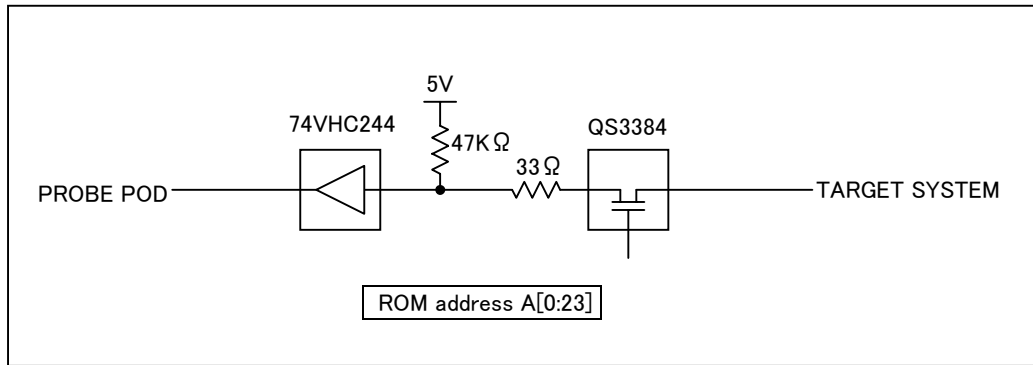
Data (variable data byte length)



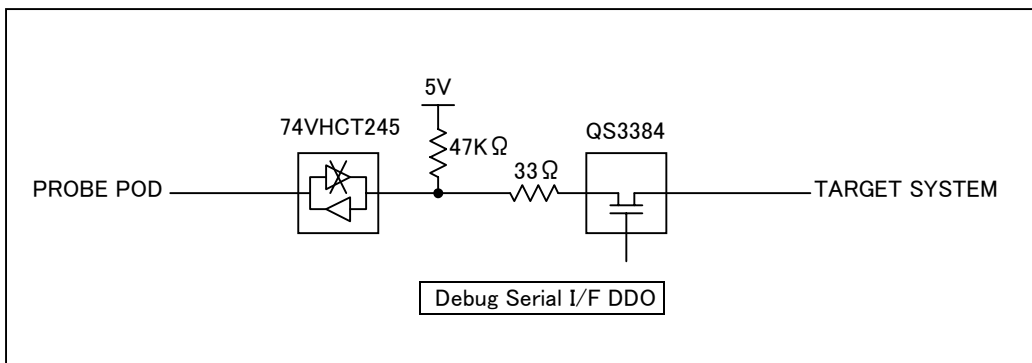
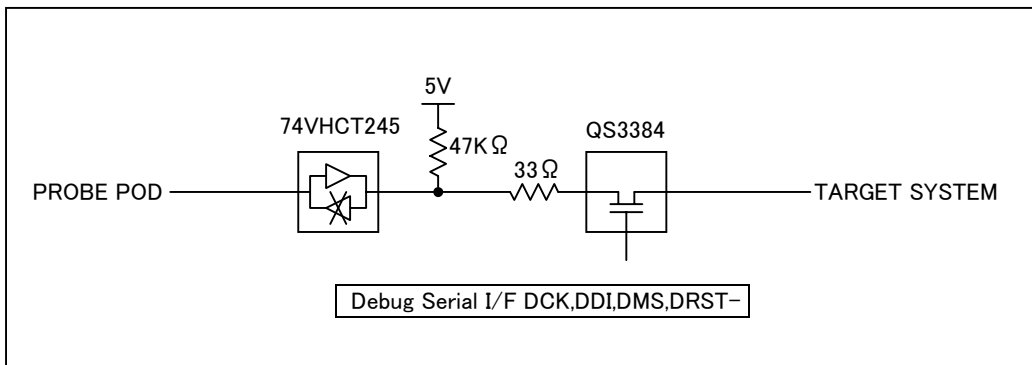
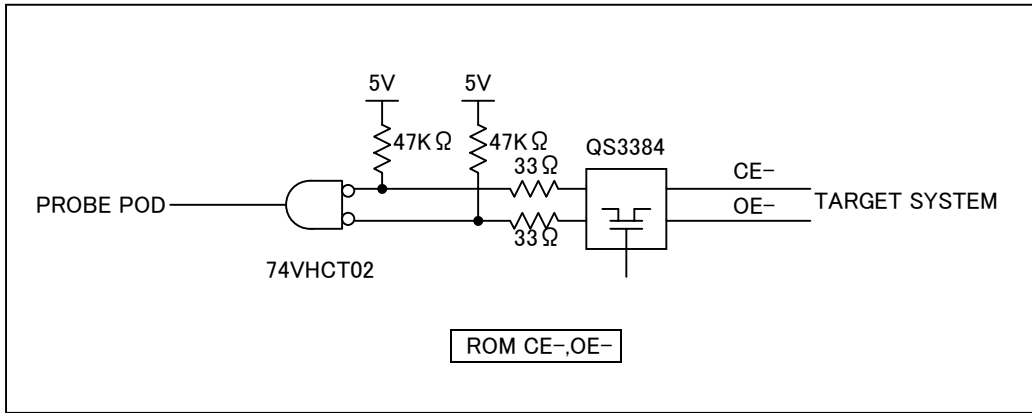
Trailing header (16 bytes)

|     |     |     |    |    |    |    |    |    |    |    |    |    |    |    |    |
|-----|-----|-----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 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 |

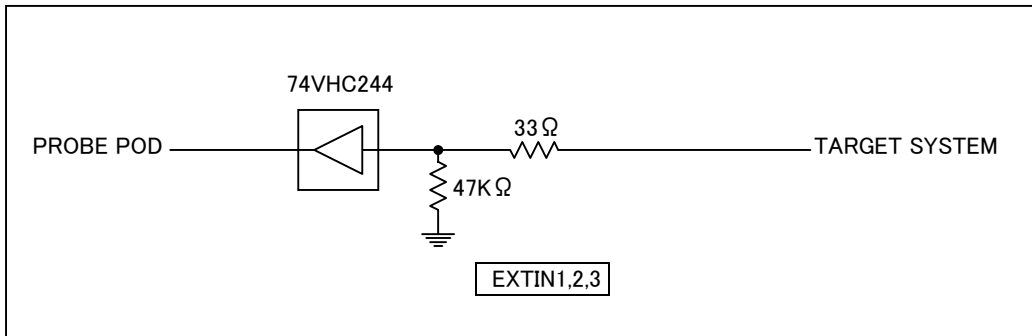
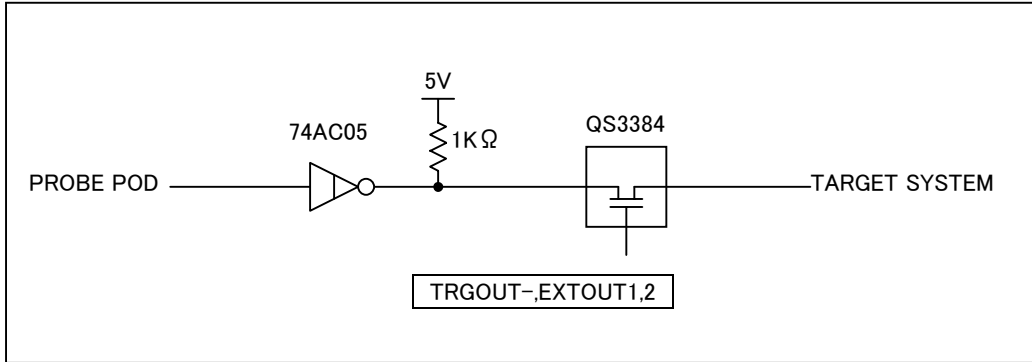
## Appendix I. Probing of the Target System



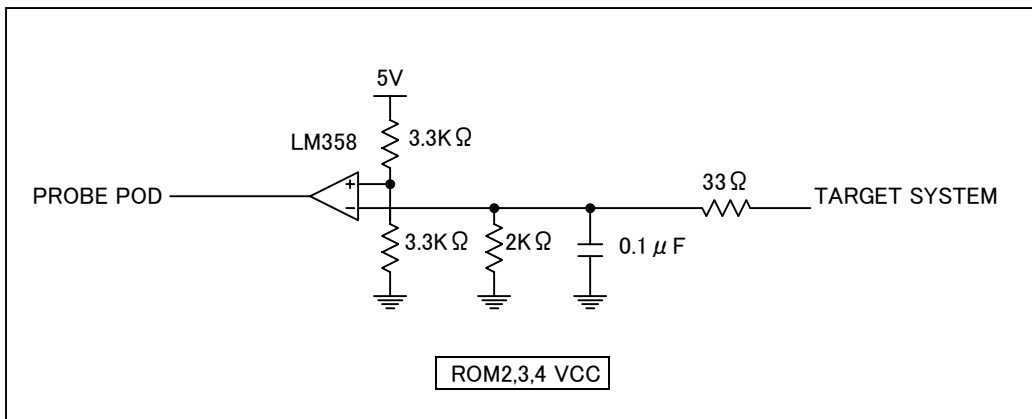
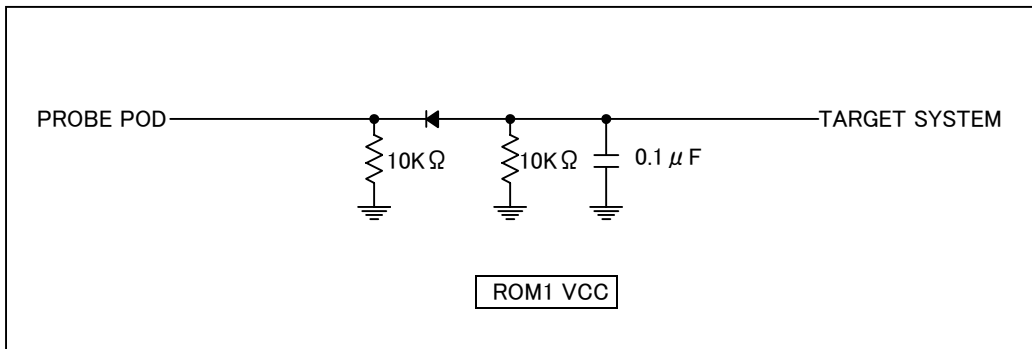
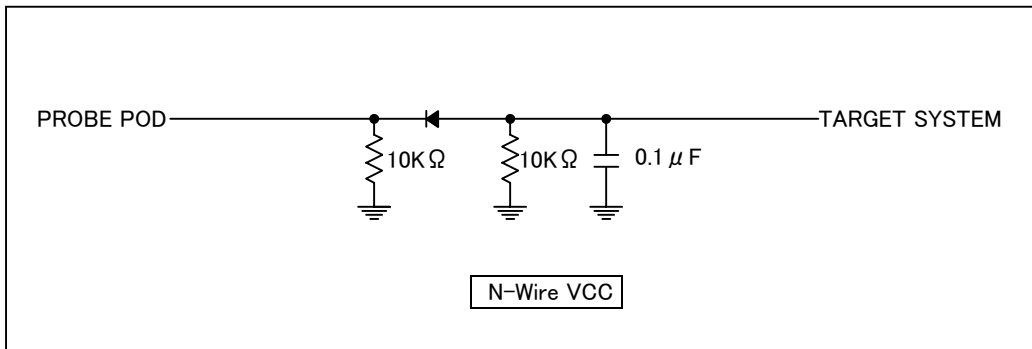
Appendix I. Probing of the Target System



Appendix I. Probing of the Target System



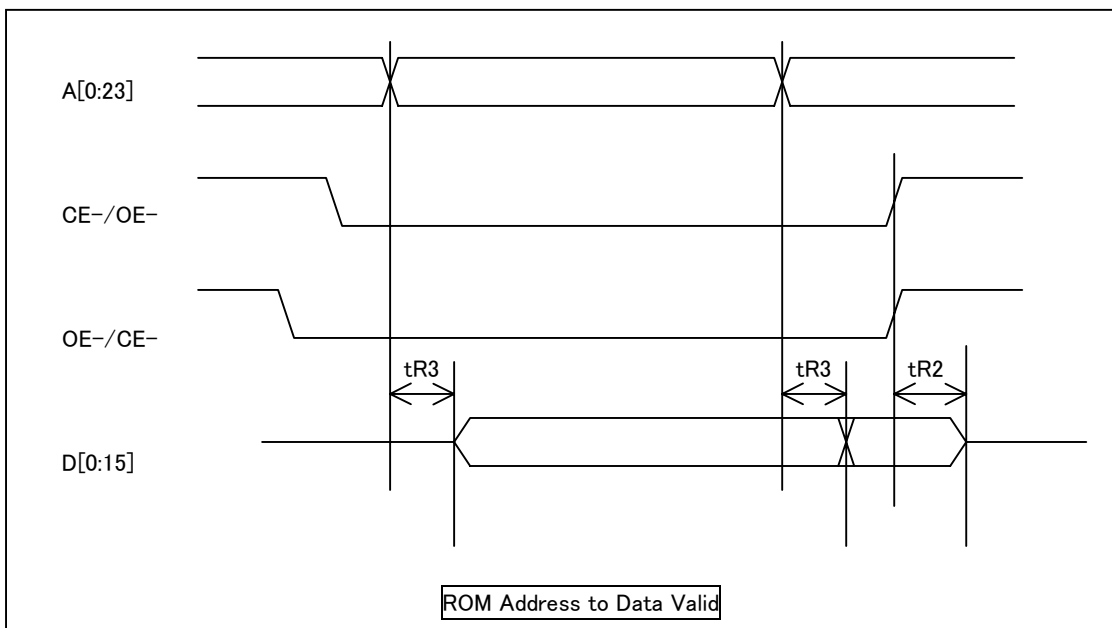
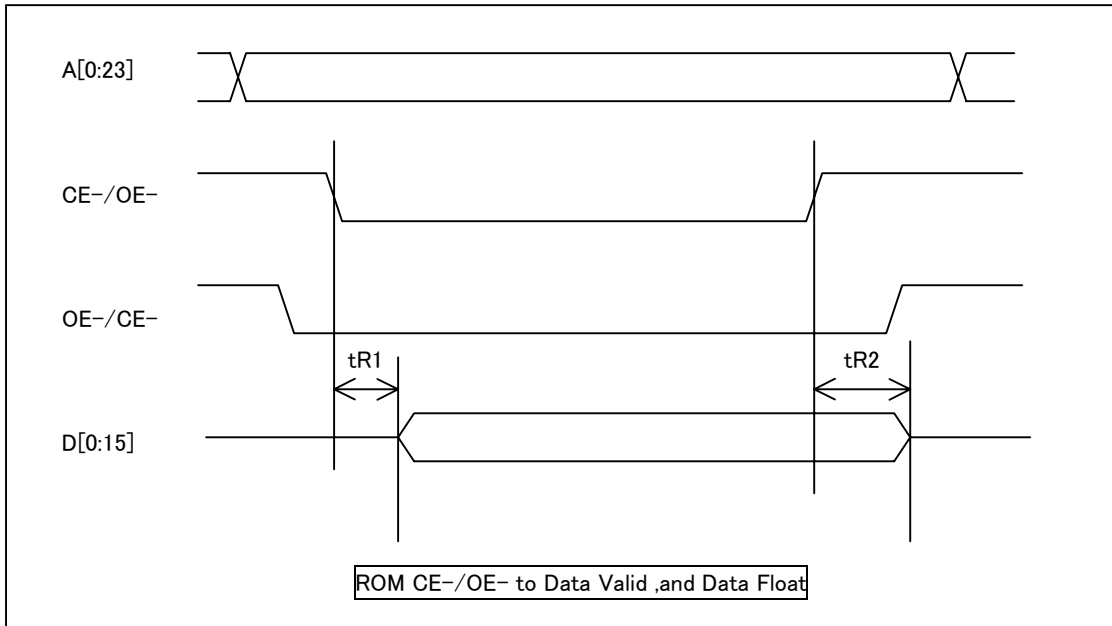
Appendix I. Probing of the Target System





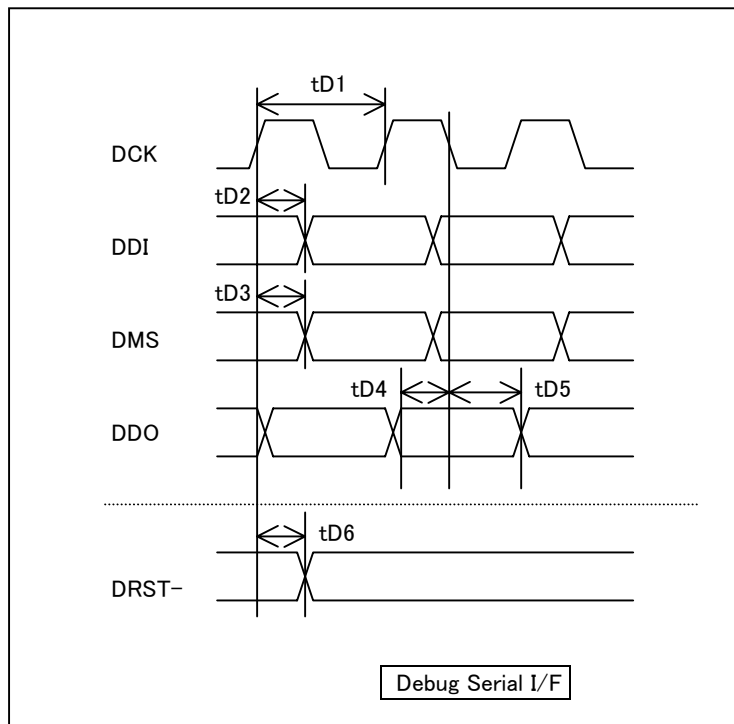
Appendix I. Probing of the Target System

| Code | Item                        | 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           |



Appendix I. Probing of the Target System

| Code | Item               | 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  |     |



## Appendix J. Troubleshooting

### PCMCIA card, Windows 95/98

- The notebook PC freezes when the PCMCIA card is inserted.
- Resource contention or no free resources
- A notebook PC that does not accept PCMCIA cards that do not use interrupts
- The Hardware Wizard does not start; the driver cannot be installed.
- The notebook PC fails to start up when a PCMCIA card from another manufacturer is used simultaneously with a Lightwell card.

### PCMCIA card, Windows NT4.0

- No free resources

### PCI card, Windows NT4.0

- The system freezes when either MULTI or the MJXDEB debugger is started.

PCMCIA card, Windows 95/98

The notebook PC freezes when the PCMCIA card is inserted.

Some notebook PCs <sup>\*1</sup> are known to freeze when the PCMCIA card is inserted as part of the driver installation process.

If this problem occurs, turn off the power, insert the PCMCIA card in the power-off state, and then turn the power back on.

After that, install the driver by referring to Section 2.2, “**Installing a Device Driver**”.

---

<sup>\*1</sup> The occurrence of this problem has been confirmed with the SONY VAIO PCG-505.

PCMCIA card, Windows 95/98

Resource contention or no free resources

If a buzzer sound goes off instead of a ping sound upon completion of a PCMCIA card device driver installation process, it is an indication that the installation of the driver failed.

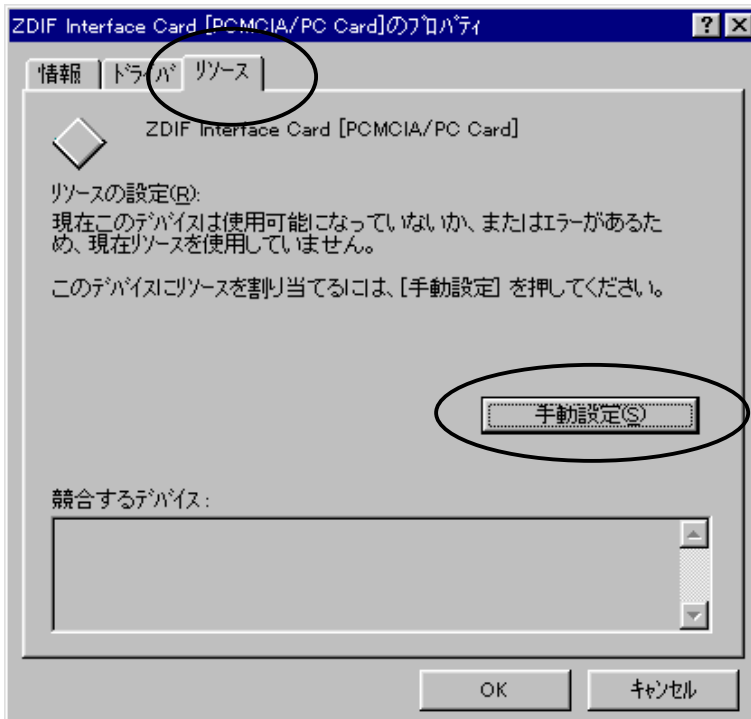
If this problem occurs, open “System” by double-clicking on it on the **Control Panel**, and then click on “Device Manager”.

Then, click on the “ZDIF” class to display the properties of “ZDIF Interface Card [PCMCIA/PC Card]” down below (either double-click or click on “Properties (R)”).

The “!” mark at the beginning of the line item indicates that there was a problem during the device driver installation process.



Clicking on the “Resources” tab the following display, at which time click on “Manual setting (S)” .



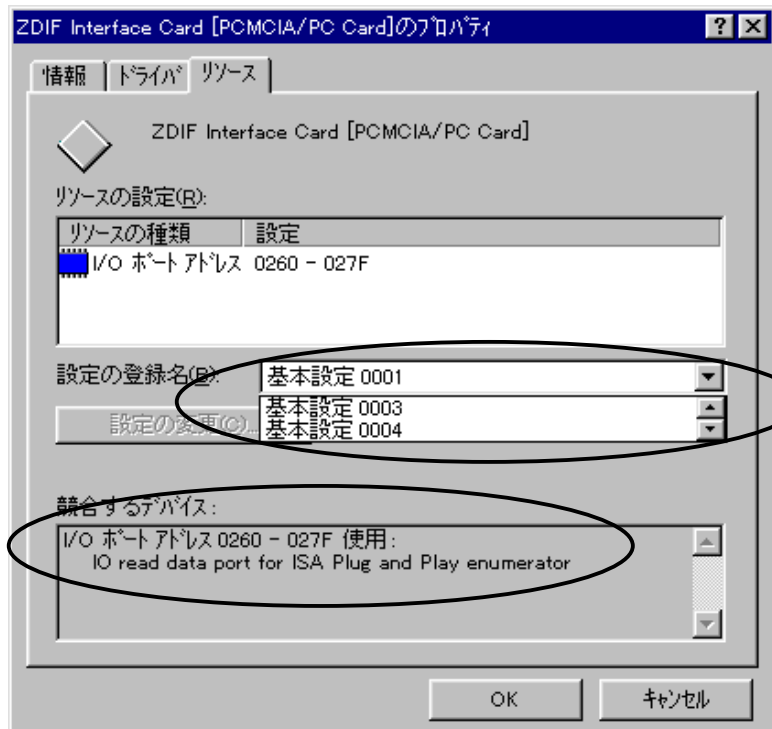
Confirm that “Auto setting (U)” is off (checked off) and that “Conflicting devices” shows that another device is using the resource (competing), rather than showing “No conflict”.



In “Settings name (B)”, select “Basic settings 000” through “Basic settings 004” in sequence.

If a “Conflicting device” indicates “No conflict”, use that device.

This amounts to performing manual setting instead of relying on Plug & Play auto-setting, which generates a warning. This warning can be ignored.



If a conflict exists in all settings from “Basic settings 000” through “Basic settings 004”, the PCMCIA card cannot be used unless the resources for other contending devices are freed.

The PCMCIA card uses one of the following resources (I/O ports):

- 0220 – 023F =basic settings 000
- 0260 – 027F = basic settings 001
- 02E0 – 02FF = basic settings 002
- 0320 – 033F = basic settings 003
- 03E0 – 03FF = basic settings 004

In situations where the resources (I/O ports) for contending devices cannot be modified, the PCMCIA card cannot be used.

**PCMCIA card, Windows 95/98**

**A notebook PC that does not accept PCMCIA cards that do not use interrupts**

If a buzzer sound is generated upon installation of a PCMCIA card device driver, indicating an installation failure but the problem does not fit in the “**Resource contention or no free resources**” category (no resource contention or the drive is not registered with Device Manager), try the following procedures:

Some notebook PCs <sup>\*1</sup> fail to properly recognize PCMCIA cards that do not use interrupts.

In this case, install the driver, and then perform an installation process where the use of interrupts by emulation is set”.

First, delete the device driver by referring to Section 2.3, “**Deleting/Uninstalling a Device Driver**” . Be sure to delete the associated INF file also.

Shut the notebook PC down and unplug the PCMCIA card.

Next, re-install the driver according to Section 2.2, “**Installing a Device Driver**”, specifying

D:\Driver\Win9x\ThinkPad

as the directory in which the driver file is located.

The driver file stored in this directory, while being otherwise identical to the “D:\Driver\Win9x” file, contains an INF file that sets the use of interrupts by the PCMCIA card by emulation.

Note that for this installation process to proceed correctly, there must be at least one free interrupt.

---

\*1 This problem has been confirmed to occur in older models in the IBM ThinkPad series.



**PCMCIA card, Windows 95/98**

**The Hardware Wizard does not start up; the driver cannot be installed.**

Some notebook PCs fail to recognize the PCMCIA card at all when it is inserted, causing the Hardware Wizard to fail to start up \*1.

In such a case, the PCMCIA card cannot be used.

If this problem occurs in your system, please contact the Technical Support Office at Lightwell.

---

\*1 This problem has been reported in conjunction with the IBM ThinkPad 380ED 2635 7AJ.

**PCMCIA card, Windows 95/98**

**The notebook PC fails to start when a Lightwell PCMCIA card is used together with a PCMCIA card made by another vendor.**

Some notebook PCs fail to start when a Lightwell PCMCIA card is used in combination with a PCMCIA card made by another vendor. \*1

Specifically, the following problem is known to occur:

- ① The PC contains an installed PCMCIA card driver.
- ② The PC contains an installed driver for the PCMCIA card made by another vendor.
- ③ The notebook PC is turned on and started with two PCMCIA cards inserted.
- ④ The notebook PC freezed during the startup process.

This problem can be avoided as follows:

- ◆ Make sure that your operating system is Windows 98.
- ◆ When starting the noebook PC, make sure that only the other PCMCIA card is inserted.  
Insert the Lightwell PCMCIA card after Windows 95 has started. (Sic – Windows 98 vs. 95)

---

\*1 This problem has been reported to occur when the following compination of components is used:

Notebook PC: NEC PC-98NX Aile NX AL20C  
OS: Windows 95 OSR2  
PCMCIA card: MelcoCorp., LAN card LPC-TX-CB  
Lightwell, PCMCIA card

**PCMCIA card, Windows NT4.0**

**No free resources**

The PCMCIA card uses one of the following resources (I/O ports) if it is available:

0220 – 023F

0260 – 027F

02E0 – 02FF

0320 – 033F

03E0 – 03FF

If all these resources are being used, the PCMCIA card cannot be used unless one of them is freed and made available.

**PCI card, Windows NT4.0**

**The system freezes when either MULTI or the MJXDEB debugger is started.**

In some cases, making access to the MJX440 via the PCI card after starting either MULTI or the MJXDEB debugger causes the entire system to freeze.\*<sup>1</sup>

This problem can be avoided by modifying BIOS settings as follows:

- ① Restart Windows NT4.0 and bring up the BIOS setting mode during the startup process.
- ② In the menu item for indicating whether the operating system is plug and play capable, specify "No".
- ③ Exit the BIOS setting mode to start Windows NT4.0.

---

\*<sup>1</sup> This problem has been reported to occur in the following models:  
NEC PC98 MateNX DA30D/S  
+ Windows NT 4.0 (+ Service Pack3)

[TRANSLATOR'S NOTES (please delete this page after reading it)

It appears that the following items are errors in the original Japanese manual:

On p. 95, "(0=disable, 1=disable)"

On p. 149, Windows 98 vs. Windows 95

The items in question are indicated by providing "sic" notes.]