

COMPREHENSIVE SERVICES

We offer competitive repair and calibration services, as well as easily accessible documentation and free downloadable resources.

SELL YOUR SURPLUS

We buy new, used, decommissioned, and surplus parts from every NI series. We work out the best solution to suit your individual needs.

 Sell For Cash  Get Credit  Receive a Trade-In Deal

OBSOLETE NI HARDWARE IN STOCK & READY TO SHIP

We stock **New**, **New Surplus**, **Refurbished**, and **Reconditioned** NI Hardware.



Bridging the gap between the manufacturer and your legacy test system.

 1-800-915-6216

 www.apexwaves.com

 sales@apexwaves.com

All trademarks, brands, and brand names are the property of their respective owners.

Request a Quote

 **CLICK HERE**

VME-MXI-2

Getting Started with Your VME-NB2040 and the NI-VXI™ Software for Macintosh



July 1994 Edition

Part Number 320414B-01

© Copyright 1992, 1994 National Instruments Corporation.
All Rights Reserved.

National Instruments Corporate Headquarters

6504 Bridge Point Parkway

Austin, TX 78730-5039

(512) 794-0100

Technical support fax: (800) 328-2203

(512) 794-5678

Branch Offices:

Australia (03) 879 9422, Austria (0662) 435986, Belgium 02/757.00.20, Canada (Ontario) (519) 622-9310,

Canada (Québec) (514) 694-8521, Denmark 45 76 26 00, Finland (90) 527 2321, France (1) 48 14 24 24,

Germany 089/741 31 30, Italy 02/48301892, Japan (03) 3788-1921, Netherlands 03480-33466, Norway 32-848400,

Spain (91) 640 0085, Sweden 08-730 49 70, Switzerland 056/20 51 51, U.K. 0635 523545

Limited Warranty

The National Instruments MXIbus boards and accessories are warranted against defects in materials and workmanship for a period of one year from the date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace equipment that proves to be defective during the warranty period. This warranty includes parts and labor.

The media on which you receive National Instruments software are warranted not to fail to execute programming instructions, due to defects in materials and workmanship, for a period of 90 days from date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace software media that do not execute programming instructions if National Instruments receives notice of such defects during the warranty period. National Instruments does not warrant that the operation of the software shall be uninterrupted or error free.

A Return Material Authorization (RMA) number must be obtained from the factory and clearly marked on the outside of the package before any equipment will be accepted for warranty work. National Instruments will pay the shipping costs of returning to the owner parts which are covered by warranty.

National Instruments believes that the information in this manual is accurate. The document has been carefully reviewed for technical accuracy. In the event that technical or typographical errors exist, National Instruments reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult National Instruments if errors are suspected. In no event shall National Instruments be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, NATIONAL INSTRUMENTS MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF NATIONAL INSTRUMENTS SHALL BE LIMITED TO THE AMOUNT THEREFORE PAID BY THE CUSTOMER. NATIONAL INSTRUMENTS WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. This limitation of the liability of National Instruments will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against National Instruments must be brought within one year after the cause of action accrues. National Instruments shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's failure to follow the National Instruments installation, operation, or maintenance instructions; owner's modification of the product; owner's abuse, misuse, or negligent acts; and power failure or surges, fire, flood, accident, actions of third parties, or other events outside reasonable control.

Copyright

Under the copyright laws, this publication may not be reproduced or transmitted in any form, electronic or mechanical, including photocopying, recording, storing in an information retrieval system, or translating, in whole or in part, without the prior written consent of National Instruments Corporation.

Trademarks

LabVIEW® and NI-VXI™ are trademarks of National Instruments Corporation.

Product and company names listed are trademarks or trade names of their respective companies.

Warning Regarding Medical and Clinical Use of National Instruments Products

National Instruments products are not designed with components and testing intended to ensure a level of reliability suitable for use in treatment and diagnosis of humans. Applications of National Instruments products involving medical or clinical treatment can create a potential for accidental injury caused by product failure, or by errors on the part of the user or application designer. Any use or application of National Instruments products for or involving medical or clinical treatment must be performed by properly trained and qualified medical personnel, and all traditional medical safeguards, equipment, and procedures that are appropriate in the particular situation to prevent serious injury or death should always continue to be used when National Instruments products are being used. National Instruments products are NOT intended to be a substitute for any form of established process, procedure, or equipment used to monitor or safeguard human health and safety in medical or clinical treatment.

FCC/DOC Radio Frequency Interference Compliance

This equipment generates and uses radio frequency energy and, if not installed and used in strict accordance with the instructions in this manual, may cause interference to radio and television reception. This equipment has been tested and found to comply with the following two regulatory agencies:

Federal Communications Commission

This device complies with Part 15 of the Federal Communications Commission (FCC) Rules for a Class A digital device. Operation is subject to the following two conditions:

1. This device may not cause harmful interference in commercial environments.
2. This device must accept any interference received, including interference that may cause undesired operation.

Canadian Department of Communications

This device complies with the limits for radio noise emissions from digital apparatus set out in the Radio Interference Regulations of the Canadian Department of Communications (DOC).

Le présent appareil numérique n'émet pas de bruits radioélectriques dépassant les limites applicables aux appareils numériques de classe A prescrites dans le règlement sur le brouillage radioélectrique édicté par le ministère des communications du Canada.

Instructions to Users

These regulations are designed to provide reasonable protection against harmful interference from the equipment to radio reception in commercial areas. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user will be required to correct the interference at his own expense.

There is no guarantee that interference will not occur in a particular installation. However, the chances of interference are much less if the equipment is installed and used according to this instruction manual.

If the equipment does cause interference to radio or television reception, which can be determined by turning the equipment on and off, one or more of the following suggestions may reduce or eliminate the problem.

- Operate the equipment and the receiver on different branches of your AC electrical system.
- Move the equipment away from the receiver with which it is interfering.
- Reorient or relocate the receiver's antenna.
- Be sure that the equipment is plugged into a grounded outlet and that the grounding has not been defeated with a cheater plug.

Notice to user: Changes or modifications not expressly approved by National Instruments could void the user's authority to operate the equipment under the FCC Rules.

If necessary, consult National Instruments or an experienced radio/television technician for additional suggestions. The following booklet prepared by the FCC may also be helpful: *How to Identify and Resolve Radio-TV Interference Problems*. This booklet is available from the U.S. Government Printing Office, Washington, DC 20402, Stock Number 004-000-00345-4.

Contents

About This Manual	ix
How to Use the Documentation Set.....	ix
Organization of This Manual	ix
Conventions Used in This Manual.....	x
Related Documentation.....	x
Customer Communication	x
Chapter 1	
Introduction	1-1
What You Need to Get Started	1-1
Optional Equipment	1-1
Optional Software	1-2
Chapter 2	
Hardware Configuration and Installation	2-1
Step 1. Unpack the NB-MXI and VME-MXI.....	2-1
Step 2. Install the NB-MXI Hardware	2-2
Step 3. Configure the VME-MXI Hardware.....	2-4
Front Panel Features.....	2-6
VMEbus System Controller.....	2-6
VME Address.....	2-7
VMEbus Request Level Selection	2-7
VMEbus Timeout Value	2-9
Step 4. Install the VME-MXI Hardware.....	2-10
Step 5. Connect the MXIbus Cable.....	2-11
Nonpolarized Cables.....	2-11
Polarized Cables.....	2-12
Chapter 3	
Software Installation and Configuration	3-1
The NI-VXI Software Package	3-1
Main Programs and Files	3-1
Installing the NI-VXI Software.....	3-4
Configuring the NI-VXI Software	3-4
Default Configurations.....	3-5
Using VXIedit.....	3-5
Configuration Editor	3-5
Logical Address Configuration.....	3-6
Bus Configuration.....	3-9
Device Configuration.....	3-10
Non-VXI Device Editor	3-11
Exiting VXIedit and Reinitializing the Hardware	3-11
Developing Your Application Program	3-12
Using the Interactive Utilities	3-12
Using the C Language Interface.....	3-12
Using LabVIEW	3-12

Appendix A
Specifications A-1

Appendix B
Customer Communication..... B-1

Glossary Glossary-1

Figures

Figure 1-1. VME-NB2040 Interface Kit..... 1-1

Figure 2-1 NB-MXI Installed in a Macintosh Computer..... 2-3

Figure 2-2 VME-MXI Parts Locator Diagram..... 2-4

Figure 2-3. VME-MXI with INTX Parts Locator Diagram..... 2-5

Figure 2-4. VMEbus System Controller Selection 2-6

Figure 2-5. VME-MXI VMEbus Requester Jumper Settings..... 2-8

Figure 2-6. VMEbus Timeout Value Selection..... 2-9

Figure 2-7. MXIbus Single-Ended Cable Configuration 2-11

Figure 2-8. MXIbus Dual-Ended Cable Configuration..... 2-12

Tables

Table 3-1. Logical Address Configuration Characteristics 3-6

Table 3-2. Bus Configuration Characteristics..... 3-9

Table 3-3. Device Configuration Characteristics 3-10

About This Manual

This manual contains instructions for installing and configuring the National Instruments VME-NB2040 interface kit. This manual is meant to be used with the *NI-VXI C Software Reference Manual for VME* and the *NI-VXI Text Utilities Reference Manual*.

How to Use the Documentation Set

Begin by reading this manual to guide you through the installation and configuration of the hardware and software. The software configuration requires that you first complete the installation and configuration of the hardware.

When you are familiar with the material in this manual, you can begin to use the *NI-VXI C Software Reference Manual for VME*. Chapter 1, *Introduction to NI-VXI for VME*, and Chapter 2, *Introduction to the NI-VXI Functions for VME*, present the concepts of using the NI-VXI software for programming VME systems and prepare you for detailed explanations of the NI-VXI functions. Study the descriptions of each function given in Chapters 3 through 9 to fully understand the purpose and syntax of each function. Refer to the *NI-VXI Text Utilities Reference Manual* to learn more about the `VICtext` and `VXIedit` programs.

Organization of This Manual

Getting Started with Your VME-NB2040 and the NI-VXI Software for Macintosh is organized as follows:

- Chapter 1, *Introduction*, describes the VME-NB2040 interface kit, lists what you need to get started, and lists optional equipment and software.
- Chapter 2, *Hardware Configuration and Installation*, contains the instructions to configure and install the VME-NB2040 interface kit.
- Chapter 3, *Software Installation and Configuration*, lists all the programs and files located on the NI-VXI distribution diskette, contains instructions for installing and configuring the NI-VXI software, and contains a description of the `VXIedit` configuration editor.
- Appendix A, *Specifications*, lists various module specifications of the NB-MXI and VME-MXI, such as physical dimensions and power requirements.
- Appendix B, *Customer Communication*, contains forms you can use to request help from National Instruments or to comment on our products and manuals.
- The *Glossary* contains an alphabetical list and description of terms used in this manual, including abbreviations, acronyms, metric prefixes, and symbols.

Conventions Used in This Manual

The following conventions are used to distinguish elements of text throughout this manual:

bold	Bold text denotes menus, menu items, dialog box buttons or options, or LEDs.
<i>italic</i>	Italic text denotes emphasis, a cross reference, or an introduction to a key concept.
<i>bold italic</i>	Bold italic text denotes a note, caution, or warning.
monospace	Text in this font denotes text or characters that are to be literally input from the keyboard, the proper names of disk drives, paths, directories, programs, subprograms, device names, filenames and extensions, and for statements and comments taken from program code.
bold monospace	Bold text in this font denotes the messages and responses that the computer automatically prints to the screen.
Macintosh NuBus	Macintosh NuBus refers to the Macintosh II and Quadra families of computers.

Related Documentation

The following documents contain information that you may find helpful as you read this manual:

- *IEEE Standard for a Versatile Backplane Bus: VMEbus*, ANSI/IEEE Standard 1014-1987
- *Multisystem Extension Interface Bus Specification*, Version 1.2
- *NI-VXI Software Reference Manual for C* (for VXI systems), National Instruments Corporation
- *VME-MXI User Manual*
- *VXI-1, VXIbus System Specification*, Rev. 1.4, VXIbus Consortium
- *VXI-6, VXIbus Mainframe Extender Specification*, Rev. 1.0, VXIbus Consortium

Customer Communication

National Instruments wants to receive your comments on our products and manuals. We are interested in the applications you develop using our products, and we want to help if you have problems with them. To make it easy for you to contact us, this manual contains comment and configuration forms for you to complete. These forms are in Appendix B, *Customer Communication*, at the end of this manual.

Chapter 1

Introduction

This chapter describes the VME-NB2040 interface kit, lists what you need to get started, and lists optional equipment and software.

The VME-NB2040 interface kit links any Apple Macintosh computer system with NuBus slots directly to the VMEbus. An Apple Macintosh computer equipped with a VME-NB2040 can function as a VME master and/or slave device. The VME-NB2040 makes your computer perform as though it were an embedded CPU plugged directly into the VME backplane or as though it has internal VME slots for plug-in boards. The driver software for the VME boards must, of course, be ported to run on the Macintosh computer with the VME-NB2040 kit.

What You Need to Get Started

- System 6.0.7 or higher installed on your Macintosh computer
- NB-MXI interface board
- Standard VME-MXI interface module
- or
- Enhanced VME-MXI interface module with INTX option
- NI-VXI distribution disk for the NB-MXI and the Macintosh
- MXIbus connector extender
- 2 m Type M1 MXIbus cable

Note: *The 2 m Type M1 MXIbus cable is included in your kit unless you specified otherwise in your order. You may have ordered your kit without this cable so that you could order a different type or length of MXIbus cable. Refer to the Optional Equipment section.*

Optional Equipment

Contact National Instruments for ordering information for any of the following optional equipment.

- VXI-MXI standard VXIbus mainframe extender kit
- VXI-MXI enhanced VXIbus mainframe extender kit
- VME-MXI standard VXIbus chassis extender kit
- VME-MXI enhanced VXIbus chassis extender kit
- MXIbus terminating pac (External)
- Type M1 MXIbus cables (straight point-to-point connectors)
- Type M2 MXIbus cables (straight point-to-right angle daisy-chain connectors)
- Type M3 MXIbus cables (right angle point-to-right angle daisy-chain connectors)

The Type M1, M2, and M3 MXIbus cables are available in the following lengths: 1 m, 2 m, 4 m, 8 m, and 20 m.

Optional Software

If you want to use the graphical programming and user interface capabilities of LabVIEW with your VME-NB2040, you must order the LabVIEW for Macintosh VXI Development System.. It contains the following components:

- LabVIEW for Macintosh Full Development System
- LabVIEW for Macintosh VXI Library
- LabVIEW for Macintosh VXI Instrument Library

Chapter 2

Hardware Configuration and Installation

This chapter contains the instructions to configure and install the VME-NB2040 interface kit. The instructions are given in the order that you should perform them. A summary of the steps is as follows:

1. Unpack the NB-MXI and VME-MXI hardware.
2. Install the NB-MXI hardware.
3. Configure the VME-MXI hardware.
4. Install the VME-MXI hardware.
5. Connect the MXIbus cable.

Step 1. Unpack the NB-MXI and VME-MXI

Follow these steps when unpacking your NB-MXI board and VME-MXI module:

1. Before attempting to configure or install the NB-MXI and VME-MXI, inspect the shipping container and its contents for damage. If damage appears to have been caused in shipment, file a claim with the carrier. Retain the packing material for possible inspection and/or for reshipment.
2. Verify that the pieces contained in the package you received match the kit parts list. *Do not* remove the boards from their plastic bags at this point.
3. Your NB-MXI board and VME-MXI module are shipped packaged in antistatic plastic bags to prevent electrostatic damage. Some of the circuitry on the NB-MXI and VME-MXI use CMOS technology and can be damaged by electrostatic discharge. Before removing the boards from their antistatic bags, touch the bags to a metal part of your computer chassis.
4. As you remove the NB-MXI and VME-MXI from their bags, be sure to handle them only by their edges. Avoid touching any of the IC components or connectors. Inspect them for loose components or any other sign of damage. Notify National Instruments if either board appears damaged in any way. *Do not* install equipment that appears to be damaged.

Step 2. Install the NB-MXI Hardware

Before attempting to install the NB-MXI, notice that some MXIbus cable connector hoods are slightly wider than most standard connector hoods and might interfere with other cables installed in adjacent NuBus slots. Normally, this will only be a problem if the cable connector hoods for the adjacent slots are also oversized. When choosing a NuBus slot in which to install the NB-MXI, verify that the MXIbus cable connector will not interfere with cables and connectors in other NuBus slots. If necessary, reposition the boards in the system to prevent cabling conflicts. It may also help to install the NB-MXI in one of the end slots so that you will only have to contend with the cable connectors of one other board.

If you cannot configure the NB-MXI to co-exist in an existing NuBus system by repositioning the boards, you can use one of the MXIbus cable options with a straight-point connector hood on the cable end that attaches to the NB-MXI. The straight-point connector hood is narrower than the MXIbus dual-connector arrangement and provides an easier fit for many system configurations. However, this approach requires that the NB-MXI be the first device in the MXIbus daisy-chain because a cable with a straight-point connector end cannot accept another MXIbus cable to propagate the bus. Remember that the first device in the MXIbus daisy-chain must also be configured as the MXIbus System Controller.

Also notice that most Macintosh computers have very narrow slot access openings. It may not be possible to properly install the MXIbus connector to the back of the NB-MXI in these machines. For this reason, National Instruments includes a MXIbus connector extender with your kit. The connector extender extends the MXIbus connector signals beyond the narrow slot opening so that you can use the standard MXIbus cables.

The following instructions are general installation instructions. Consult the user or technical reference manual of your computer for specific instructions and warnings.

1. Plug in your Macintosh computer before installing the NB-MXI. The plug grounds the computer and protects it from electrical damage while you are installing boards.

Warning: *To protect both yourself and the computer from electrical hazards, the computer should remain off until you are finished installing the board.*

2. Remove the top cover or access port of the Macintosh.
3. Locate and remove the bracket that covers the cut-out in the back panel of the Macintosh for the slot you have selected.
4. Before picking up the NB-MXI, touch the metal part of the power supply case inside the computer to discharge any static electricity that might be on your clothes or body.
5. Line up the NB-MXI with the MXIbus connector near the cut-out on the back panel and the other card edge lined up with the respective slot guide. Slowly push down on the middle of the NB-MXI until its connector is resting on the expansion slot receptacle. Using slow, evenly distributed pressure, press the NB-MXI straight down until it seats in the expansion slot.

6. Check the installation.
7. Replace the computer cover.
8. Insert the connector extender to the board MXIbus connector in the Macintosh computer. You will connect the MXIbus cable to this connector extender.

Figure 2-1 shows an NB-MXI installed in a Macintosh computer.

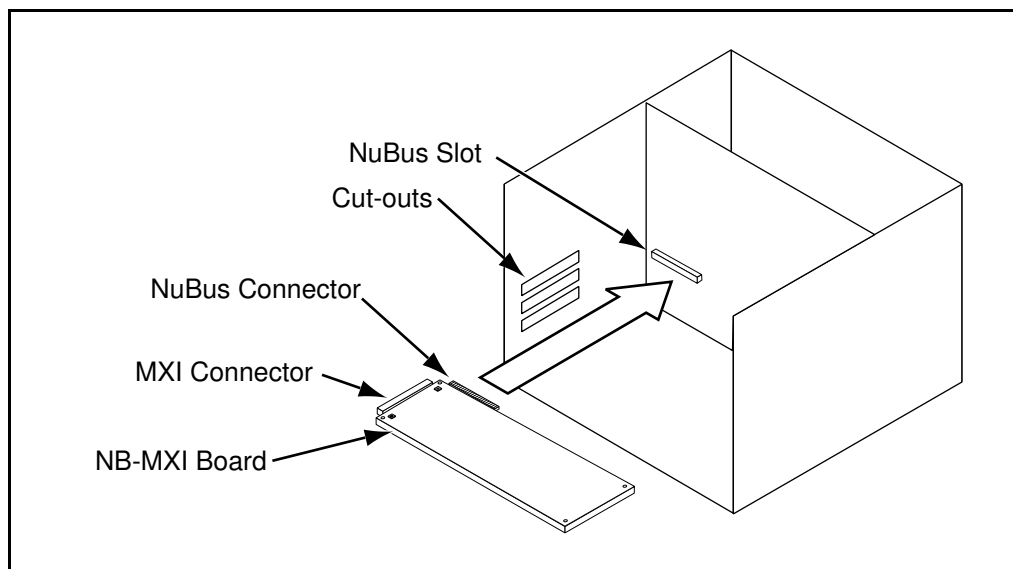


Figure 2-1. NB-MXI Installed in a Macintosh Computer

Step 3. Configure the VME-MXI Hardware

Figure 2-2 shows the location and factory default settings of the configuration switches and jumpers for a VME-MXI without the INTX daughter card option. The remainder of this chapter describes only those options that are user-configurable. Do not change the default settings of other jumpers and switches on the VME-MXI board unless you plan to install more than one VME-MXI in the same chassis. If this is the case, refer to the *VME-MXI User Manual* that came with your additional VME-MXI interface for more information.

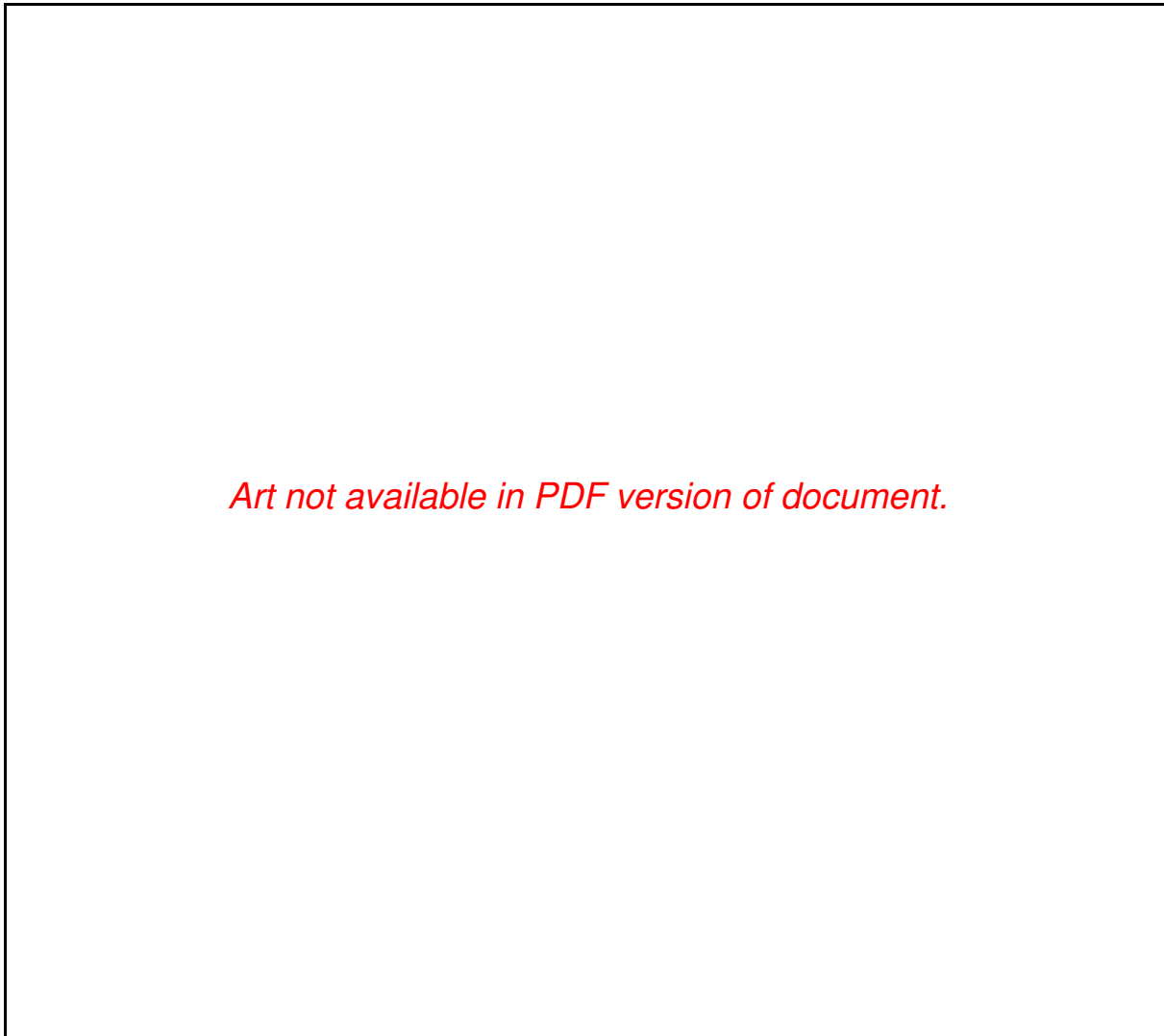


Figure 2-2. VME-MXI Parts Locator Diagram

Figure 2-3 shows the location and factory default settings of the configuration switches and jumpers for a VME-MXI with the INTX daughter card option. The options described in the rest of this chapter apply to VME-MXI modules with or without the INTX option. For more information about configuring modules with the INTX option, refer to the *VME-MXI User Manual*.

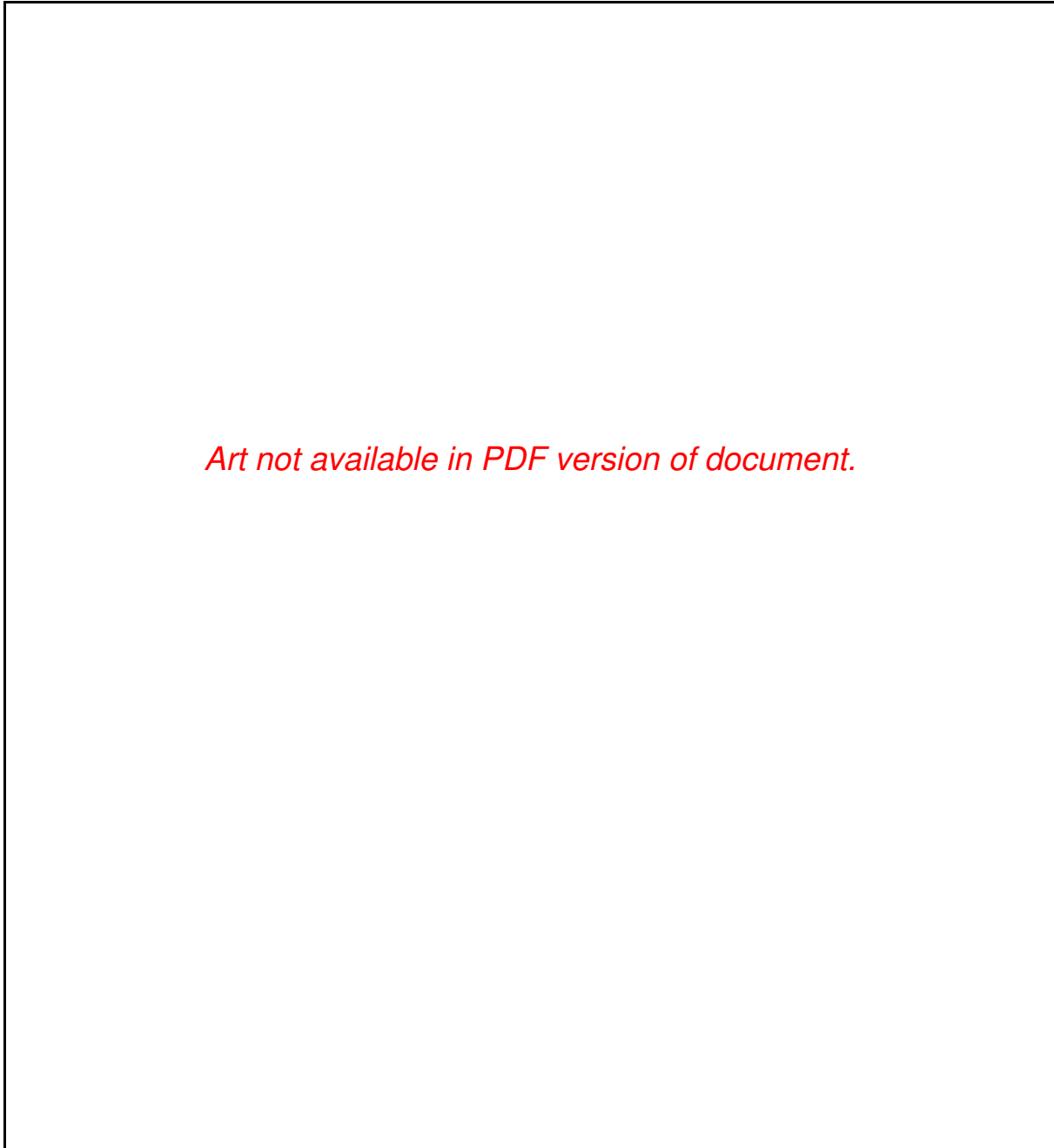


Figure 2-3. VME-MXI with INTX Parts Locator Diagram

Front Panel Features

The VME-MXI has the following front panel features:

- Three front panel LEDs
 - **FAILED** LED indicates that the VMEbus SYSFAIL line is asserted.
 - **VME ACCESS** LED indicates when the VME-MXI is accessed from the VMEbus.
 - **MXI ACCESS** LED indicates when the VME-MXI is accessed from the MXIbus.
- MXIbus connector
- System reset pushbutton
- INTX connector (if your VME-MXI includes the INTX daughter card option)

VMEbus System Controller

The VME-MXI is factory configured to be installed in Slot 1 of the VMEbus chassis as the VMEbus System Controller. If another device is already in Slot 1, you must decide which device will be the VMEbus System Controller and reconfigure the other device as non-VMEbus System Controller.

When the VME-MXI is the VMEbus System Controller, it has the VMEbus Data Transfer Bus Arbiter capability (PRI ARBITER) and it drives the 16 MHz VMEbus system clock. The VMEbus Data Transfer Bus Arbiter circuitry accepts bus requests on all four VMEbus request levels, prioritizes the requests, and grants the bus to the highest priority requester. The VMEbus system clock is driven by an onboard 16 MHz oscillator with a 50% \pm 5% duty cycle.

If you want to install the VME-MXI into a VXIbus mainframe, you can install it in any slot except Slot 0. The VME-MXI has VXIbus configuration registers, which makes it compatible with the VXIbus specification. However, the VME-MXI does not have the CLK10 and MODID circuitry required by a Slot 0 VXIbus device. Therefore, configure the VME-MXI as a non-VMEbus System Controller and install it in any other slot.

Figure 2-4(a) shows the default configuration setting for the VME-MXI installed as the VMEbus System Controller. To configure the VME-MXI as a non-VMEbus System Controller, change slide switch S5 as depicted in Figure 2-4(b).

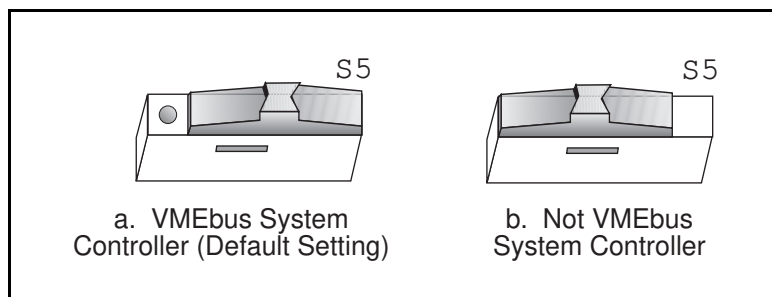


Figure 2-4. VMEbus System Controller Selection

VME Address

The NB-MXI and the VME-MXI use VME short (A16) address space for their configuration, communication, and status registers. Because both boards are not only VMEbus compatible but are also compatible with the VXibus specification, each board has an associated logical address. The logical address assigns a unique 64-byte block of address space in the upper 16 KB of A16 space to each board. The starting address of a device's address space can be calculated using the following formula:

starting address (A16 space) = (logical address * 40h) + C000h

The NB-MXI is responsible for configuring the system upon power-up or reset and is referred to as the System Resource Manager. Because the System Resource Manager is defined to have a logical address of 0, the NB-MXI will use A16 address space from C000h through C03Fh. The logical address of the NB-MXI is software configurable, but should not be changed when used with a VME system.

The VME-MXI is factory configured with a logical address of 1, meaning that it occupies A16 space in the range of C040h through C07Fh. The logical address of the VME-MXI is configurable via an onboard switch, but should not be changed unless you are using multiple VME-MXIs in your system. If this is the case, refer to the *VME-MXI User Manual* that came with your additional VME-MXIs for more information.

Ensure that no other VMEbus devices in your system occupy address space used by the NB-MXI or the VME-MXI. If they do, change the other VME devices so that they occupy unique portions of address space. It is recommended that any A16 devices have their operational registers in the lower three-quarters of A16 space (below C000h), leaving the upper quarter for logical addresses. This is required if the VME-MXI is part of a VXibus system. In addition, if you are installing the VME-MXI in a VXibus system, make sure that no other VXibus devices are statically configured at either Logical Address 0 or 1.

VMEbus Request Level Selection

The VME-MXI uses one of the four VMEbus request levels to request use of the VME Data Transfer Bus (DTB). The VME-MXI requests use of the DTB whenever an external MXibus device—such as a Macintosh computer with an NB-MXI interface—attempts a transfer that maps into the VMEbus chassis.

The VME-MXI is factory configured to use VMEbus request level 3. This is the highest priority request level and is suitable for most VMEbus systems. However, you can change the VME-MXI to use any of the other three request levels (0, 1, or 2) by changing the jumper configuration of the pin arrays at locations W2, W3, W4, W5, and W6. You may want to change request levels to change the priority of the VME-MXI request signal. For more information, refer to the VMEbus specification.

To change the VMEbus request level of the VME-MXI, rearrange the jumpers on the pin arrays as shown in Figure 2-5.

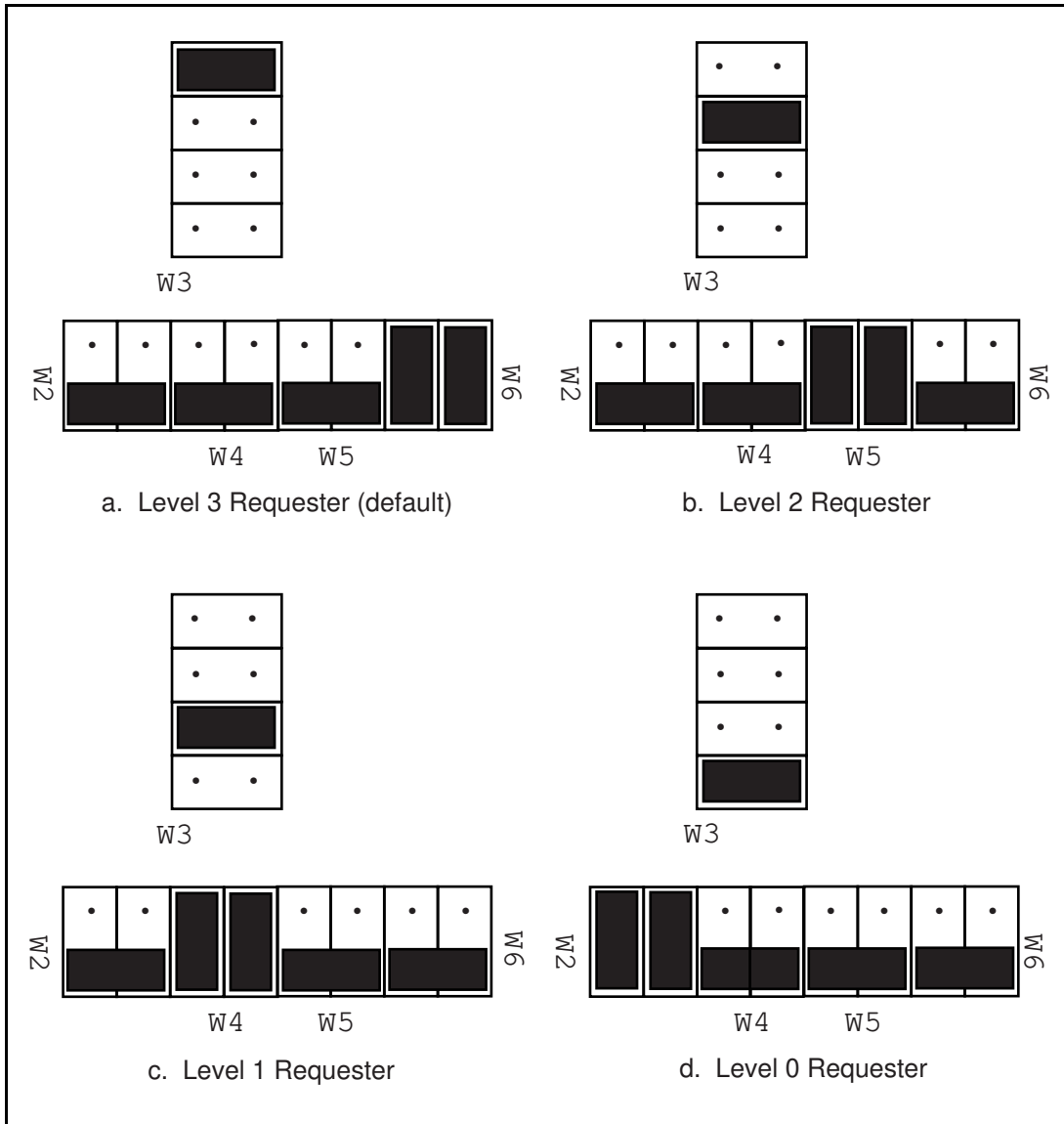


Figure 2-5. VME-MXI VMEbus Requester Jumper Settings

VMEbus Timeout Value

The VME-MXI contains the VMEbus Bus Timeout Unit (BTO) circuitry for the VMEbus. The BTO monitors the current bus cycle and asserts the bus error (BERR*) signal if either data strobe (DS1* or DS0*) remains active for a given amount of time.

The VMEbus system should have one, and only one, device that functions as the BTO Monitor on the VMEbus. The VME-MXI is factory configured to be the VMEbus System Controller (installed in Slot 1) and to provide the VMEbus BTO function. It is recommended that you use the VME-MXI as the VMEbus System Controller. If, however, you use another device as the VMEbus System Controller, it is recommended that you still retain the VME-MXI as the BTO Monitor, if possible. In this case, disable the BTO function on the System Controller and leave it enabled on the VME-MXI. If it is not possible to disable the BTO function on the System Controller, you can disable the BTO function on the VME-MXI instead. However, be sure that the BTO timeout value on your System Controller is sufficient to allow transfers to complete across the MXIbus. This is especially important for multiple-chassis systems or for systems with more than one VME-MXI.

The VMEbus BTO value can be disabled or set to 100, 200, or 400 μ s by rearranging the jumper selection at location W8, as shown in Figure 2-6.

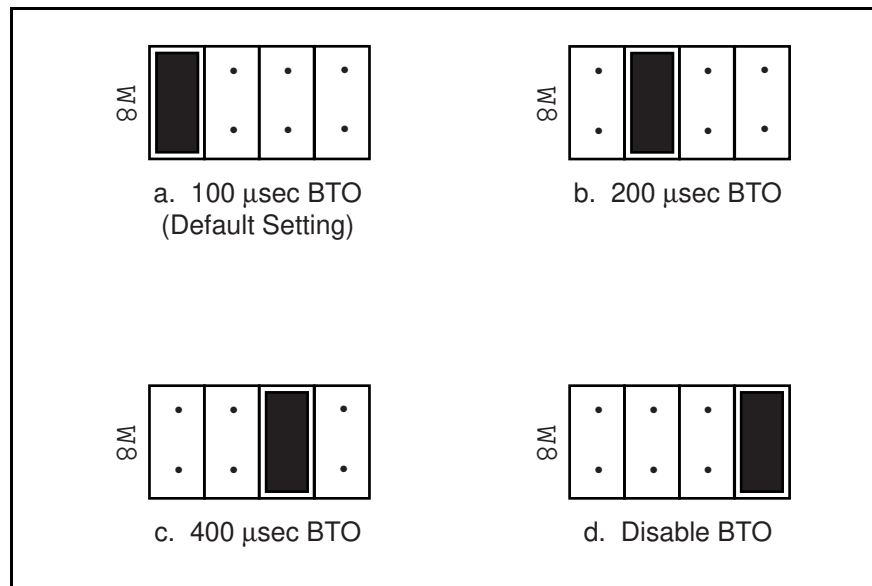


Figure 2-6. VMEbus Timeout Value Selection

Step 4. Install the VME-MXI Hardware

Given below are general installation instructions for the VME-MXI. Please read the following important warnings before installing the VME-MXI. Also consult the user manual or technical reference manual of your VMEbus chassis for specific instructions and warnings.

Warnings: *To protect both yourself and the chassis from electrical hazards, the chassis should remain off until you are finished installing the board.*

The VME-MXI can be installed into your VMEbus chassis either in Slot 1 when configured as the VMEbus System Controller, or it can be installed in any other slot except Slot 1 when configured as non-System Controller. Installing your VME-MXI into a slot that does not correspond with the jumper settings may result in damage to the VME-MXI, the VMEbus backplane, or both.

1. Plug in your VMEbus chassis before installing the VME-MXI. The plug grounds the chassis and protects it from electrical damage while you are installing boards.
2. Remove or open any doors or covers blocking access to the chassis slots.
3. Select the slot in the chassis that is correct for your system and insert the VME-MXI in the slot by aligning the top and bottom of the card with the card edge guides inside the chassis. Slowly push the VME-MXI straight into the slot until its plug connectors are resting on the backplane's receptacle connectors. Using slow evenly distributed pressure, press the VME-MXI straight in until it seats in the expansion slot. The front panel of the VME-MXI should be even with the front panel of the chassis.
4. Tighten the retaining screws on the top and bottom edges of the front panel.
5. Check the installation.
6. Connect the cables as described in the following section before restoring power.
7. Replace or close any doors or covers to the chassis.

Step 5. Connect the MXIbus Cable

There are two basic types of MXIbus cables. MXIbus cables can have either a single connector on each end or a single connector on one cable end and a double connector on the other end. Your VME-NB2040 kit comes standard with a cable with single connectors on each end.

Nonpolarized Cables

The cable with a single connector on each cable end is nonpolarized and may be installed with either end connected to either device. Be sure to tighten the screw locks to ensure proper pin connection. See Figure 2-7.

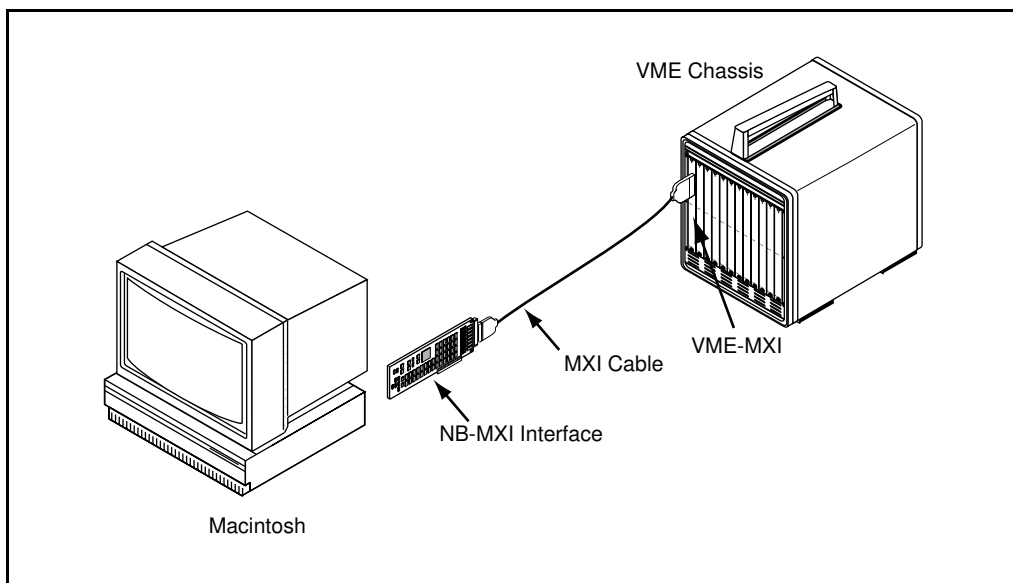


Figure 2-7. MXIbus Single-Ended Cable Configuration

Polarized Cables

If you are using a MXIbus cable with a single connector on one cable end and a double connector on the other end, it is a polarized cable that must be installed correctly for the system to function properly. Connect the end with the *single* connector to the NB-MXI and the end of the cable with the *double* connector to the VME-MXI. Be sure to tighten the screw locks to ensure proper pin connection. See Figure 2-8.

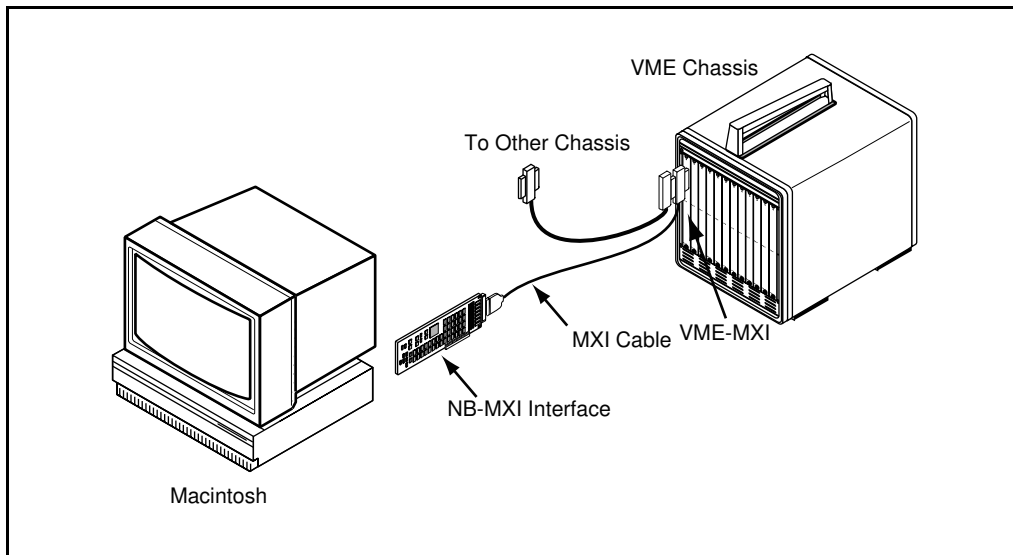


Figure 2-8. MXIbus Dual-Ended Cable Configuration

When the MXIbus cable is properly connected, turn on the VME chassis and then the Macintosh computer. After all devices are powered on and running properly, you may run the Resource Manager software utility to configure your system.

Chapter 3

Software Installation and Configuration

This chapter lists all the programs and files located on the NI-VXI distribution diskette, contains instructions for installing and configuring the NI-VXI software, and contains a description of the `VXIedit` configuration editor.

The NI-VXI software package is a comprehensive set of utility programs and driver software routines that you can use to configure your MXIbus-to-VME link and program your VME system. The NI-VXI software package was actually developed for the VXI-NB2040 kit for VXI systems. Because VXI is a superset of VME, and because the VME-MXI interface board for the VMEbus is compatible with the VXI-MXI interface board for the VXIbus, the NI-VXI software package contains comprehensive tools for VME systems. Some of the utility programs and driver software routines in the NI-VXI software package only apply to VXI systems and are not used in VME systems. Refer to the remainder of this chapter and to the *NI-VXI C Software Reference Manual for VME* to determine which of the NI-VXI software package utility programs and driver functions are used for VME, and which apply only to VXI.

The NI-VXI Software Package

The following files make up the NI-VXI software.

Main Programs and Files

The following are the main programs and files of the NI-VXI software package:

- `NI-VXI` is a folder containing the NI-VXI software for the NB-MXI.
- `NI NBMXI INIT` is a file that loads the NI-VXI device driver and initializes the NB-MXI board when the Macintosh computer with the NB-MXI is powered on or restarted. This program is copied into the `System` folder.

Note: *If your computer is running under System 7, you must ensure that the `NI NBMXI INIT` file is in the `Extensions` folder in the `System Folder`.*

- `VXI resource mgr files` is a folder containing the files used by the Resource Manager and the configuration files. This folder is also copied into the `System Folder`. Otherwise, make sure it is in the `System Folder`.

The NI-VXI folder contains the following files and subfolders:

- `README.DOC` is a file containing information about the NI-VXI software and the various files.
- `CLI` is a folder containing the THINK C language interface to the VXI device driver, user include files and some applications.
- `VXIinit` is the NB-MXI initialization program. This program initializes the board interrupts, Shared RAM, VXI register configurations, and bus configurations. Normally, you will not have to run this application because the NB-MXI is initialized when the Macintosh computer is powered on. Run this application only if you change the default configuration using `VXIedit` and you do not want to restart the NB-MXI.
- `Resource Manager` is the National Instruments multiple-mainframe Resource Manager. Run this program after you power-on or restart the VME chassis or the Macintosh computer containing the NB-MXI. The MPW tool is called `ResMan`.
- `VICtext` is a text-based interactive control program that executes functions you enter from the keyboard. It helps you learn the functions, program your VXI device, and develop and debug your application program.
- `VXIedit` is the VXI text-based resource editor program. You can use the Non-VXI Device Editor in this program to identify details about VME devices installed in your system. You must use this editor to instruct your system about the addresses your VME devices occupy. The Resource Manager can then use this configuration information to automatically *open* the hardware windows so that your Macintosh computer can access the VMEbus. This program also displays the system configuration information generated by the Resource Manager after it configures the link to the VMEbus. In VXI systems, you can also use this program to edit the model names of VXI devices and the manufacturer name and ID numbers.

The `VXI resource mgr files` folder contains the following files:

- `mfnameid.tbl` is a file that contains the data base of the manufacturer names and their ID numbers. This file is primarily used in VXI systems because VXI devices have manufacturer IDs. It is not used in VME systems.
- `model.tbl` is a file that contains the data base of the model names, manufacturer names, and the model codes numbers. This file is used only in VXI systems and is not used in VME systems.
- `device.tbl` is a file that contains the data base of the device names, manufacturer names, model names, and frame and slot associations for devices in the system. This file is used only in VXI systems and is not used in VME systems.
- `nonvxi.tbl` is a file that contains the data base for all non-VXI (VME) devices in the system. This file is used for VME systems and describes the address space and interrupt requirements of the VME devices in the system. The Resource Manager uses this file to determine how to open the windows to the VMEbus. Use the Non-VXI Device Editor to configure the address space requirements for your VME devices so that the Resource Manager can configure the hardware windows properly.

- `intcfg.tbl` is a file that contains the system interrupt configuration information. In VXI systems, interrupts can be assigned dynamically. In VME systems, however, interrupts are statically configured. This file also contains interrupt mapping information that can be used in a VME system but only in a multiple-mainframe situation.
- `trigcfg.tbl` is a file that contains the system trigger configuration information. This table file is not used in VME systems.
- `utilbus.tbl` is a file that contains the utility bus configuration information. This file is used in a VME system but only in a multiple-mainframe situation.
- `creg.tbl` is a file that contains device-dependent information to be written to the Control register of individual VXI devices by the Resource Manager.
- `vxila.cfg`, `vxibus.cfg`, and `vxidev.cfg` are files that contain the configuration information for the NB-MXI hardware.

The subfolder `CLI` contains the following files and folders:

- `cvxi.c` is the NI-VXI THINK C language interface to the NI-VXI driver.
- `Example Folder` is a folder containing various example programs that show how to use the NI-VXI software.
- `Include Folder` contains the following include files for the C language interface:
 - `nivxi.h` is the main header file containing the C prototypes for the NI-VXI functions.
 - `cvxi.h` is the include file used by `cvxi.c`.
 - `datasize.h` is a file that contains data size specifications.
 - `busacc.h` is a file that contains parameter and return values for the bus access functions.
 - `devinfo.h` is a file that contains parameter and return values for the device information and system configuration functions.
 - `vxoint.h` is a file that contains parameter and return values for the interrupt and signal functions.
 - `sysint.h` is a file that contains parameter and return values for the system interrupt functions.
 - `trig.h` is a file that contains parameter and return values for the trigger functions. This file is useful in VXI systems but is not applicable for VME systems.
 - `ws.h` is a file that contains parameter and return values for the Commander and Servant Word Serial functions. This file is useful in VXI systems but is not applicable for VME systems.

Your MPW folder will contain all the C library files after you install the NI-VXI software. In addition, NI-VXI installs versions of the VXI utilities that can run as MPW tools. You may need to adjust your environment variables so that you can access these files.

Installing the NI-VXI Software

Follow these instructions to install the NI-VXI software.

1. Insert the NI-VXI distribution disk and double-click on the **Install NI-VXI** icon.
2. The Installer window appears. You can install either all or only some of the NI-VXI files.
 - a. To install all of the files, drag the **All VXI files** icon to the appropriate hard disk.
 - b. To install only part of the software, click on **Show Other Installations** at the top of the Installer window. Select the items you want to install and drag the icons to the appropriate hard disk.
3. Restart your Macintosh when prompted to do so. The NI-VXI software is now installed.

Configuring the NI-VXI Software

Run the `VXIteDit` device configuration utility if you need to make any changes to the default software configurations. If you have not changed any of the default hardware settings, and the default software configurations are acceptable, you do not need to run the configuration utility. You can also run `VXIteDit` if you just want to examine the current software configuration.

Use the Non-VXI Device Editor from the main menu to configure your non-VXI (VME) devices. Be sure to specify a unique pseudo-logical address for each VME device. The pseudo-logical address can be any value between 256 and 511 inclusive. It is recommended that you start with 256 for your first device, and then give each subsequent device the next higher number. Assigning pseudo-logical addresses to your VME devices ensures that the Resource Manager program recognizes your VME devices when it configures your interface hardware. You can also enter manufacturer name, model name and other identification information for your VME devices. Be sure to configure the `InSystem` parameter to instruct the Resource Manager that this particular VME device is indeed installed in your system. If you choose, you can enter configuration information for numerous VME devices using the Non-VXI Device Editor.

Keep in mind that the Resource Manager will only use the configuration information for those devices that you indicated `InSystem`. When the Resource Manager executes, it displays information regarding the configuration of the interface hardware and your VME devices by referencing their assigned pseudo-logical addresses and device names.

You can enter configuration information for each VME device in your system either individually as unique VME devices, or collectively by considering the overall address space requirements of your VME devices as one logical VME device. By using `VXIteDit` to configure the address space requirements, the Resource Manager can automatically enable the hardware windows through which your computer can access the corresponding VME addresses where your VME devices reside. You can also run `VXIteDit` at any time to examine the software configurations.

You can also use this program to configure interrupts and the utility bus for single-mainframe or multiple-mainframe system configurations. The following sections describe how to modify

and/or view the configuration information for the NB-MXI. For more details on how to use `VXItedit`, refer to the *NI-VXI Text Utilities Reference Manual*.

Default Configurations

The NI-VXI software has factory default configurations. These default configurations match the factory default configurations of the hardware interface.

You must run `VXItedit` and use the Non-VXI Device Editor to configure the VME addresses you want to access from your computer. If you do not use `VXItedit` to make any configuration changes, the default characteristics of the software will remain in effect.

Using VXItedit

To run the VXI resource editor program, type `VXItedit`. The Configuration Editor, Non-VXI Device Editor, and Resource Manager Display are used in VME systems. Use the Configuration Editor to configure details of the NB-MXI hardware configuration. Use the Non-VXI Device Editor to configure the address requirements of VME devices in your VME system so that the Resource Manager program can automatically set up the MXI link. The Resource Manager Display lets you examine the status of the address space configuration after the Resource Manager program has executed. The Manufacturer Name Editor, Model Name Editor, Device Name Editor, and Trigger Configuration Editor are specific to VXI systems and are not used in VME systems (VXI is a superset of VME). The Interrupt Configuration Editor and Utility Bus Configuration Editor can be used in VME systems, but only in multiple-mainframe system configurations.

For further instructions on the use of the `VXItedit` program, refer to the *NI-VXI Text Utilities Reference Manual*.

Configuration Editor

Select the Configuration Editor from the main menu. The three configuration options available under the Configuration Editor are Logical Address Configuration, Bus Configuration, and Device Configuration.

Type `help` to obtain online assistance within any editor, including determining which commands are applicable to the editor you are using. Type `save` to apply any changes you have made, and then type `exit`. Saving the changes updates files `vxila.cfg`, `vxibus.cfg`, and `vxidev.cfg`. The `VXIinit` program uses these files to update the soft-configured settings, including board interrupts, shared RAM size, VXI register contents, and bus interface. If you want to exit without saving your changes, type `abort`. When you return to the main menu, you can enter another editor by selecting the number for that editor.

The following sections describe the various configuration options available under the Logical Address Configuration, Bus Configuration, and Device Configuration menus.

Logical Address Configuration

Table 3-1 describes the characteristics of the options available under the Logical Address Configuration Editor. Normally for VME systems, you do not have to modify the default configuration information.

Table 3-1. Logical Address Configuration Characteristics

Characteristic	Description	Default Value
Logical Address	An 8-bit number that uniquely identifies the NB-MXI. If it is set to 0, it is the VXIbus Resource Manager. Logical Addresses 0 through 254 specify a static device, while Logical Address 255 specifies a dynamic device, which is not an allowed option. Range = 0 to 254	0
Device Type	Indicates the classification of the NB-MXI. Affects only the contents of the Device Class field in the Device Type register. The functionality of the other registers does not change. 1 = Extended 2 = Message-Based 3 = Register-Based	2
Address Space	Indicates the addressing mode(s) of the NB-MXI's operational registers. 0 = A16/A24 1 = A16/A32 3 = A16 only	3

(continues)

Table 3-1. Logical Address Configuration Characteristics (Continued)

Characteristic	Description	Default Value
Shared Memory Size	<p>Indicates the size of VXI Shared RAM. The NB-MXI local RAM can be slave-accessed in either A24 or A32 space in D08, D16, and D32 data transfer modes.</p> <p>1 MB 2 MB 4 MB 8 MB 255 = Share all</p>	255
Shared Memory Buffer	<p>Indicates the size of the buffer to be allocated for Shared Memory. You must restart the Macintosh for this change to take effect. Size is in 64 KB blocks.</p> <p>1 = 64 KB 2 = 128 KB 3 = 292 KB</p>	0
Resource Manager Delay	<p>Time in seconds to wait after power-up or reset before configuring the VME hardware interface and checking for any VXIbus devices residing in the system.</p> <p>Range = 0 to 65535 seconds</p>	5 s
Servant Area Size	<p>Supplied when the NB-MXI receives the <i>Read Servant Area</i> command from the Resource Manager. The Servant area size is an 8-bit value (0 through 255) that indicates the NB-MXI Servant area. The Servant area begins at the logical address following the NB-MXI's logical address, and includes <i>N</i> contiguous logical addresses, where <i>N</i> is the value of the Servant area size. This field is valid only when the NB-MXI is configured as a Message-Based device.</p> <p><i>Not used in VME systems.</i></p> <p>Range = 0 to 255</p>	0

(continues)

Table 3-1. Logical Address Configuration Characteristics (Continued)

Characteristic	Description	Default Value
Protocol Register	<p>Copy of the Protocol register, indicating which protocols the NB-MXI supports. This field is valid only when the NB-MXI is configured as a Message-Based device. (Refer to the <i>VXibus System Specification</i>.)</p> <p><i>Not used in VME systems.</i></p>	0x0ff0
Response for <i>Read Protocol Command</i>	<p>The protocols that are supported, given in response when the NB-MXI receives the <i>Read Protocol</i> command from the Resource Manager. This field is valid only when the NB-MXI is configured as a Message-Based device. (Refer to the <i>VXibus System Specification</i>.)</p> <p><i>Not used in VME systems.</i></p>	0x8448
Number of Handlers	<p>The number of interrupt handlers that the NB-MXI supports.</p> <p>Range = 0 to 7</p>	1
Number of Interrupters	<p>The number of interrupters that the NB-MXI supports.</p> <p>Range = 0 to 7</p>	0

Bus Configuration

Table 3-2 describes the characteristics of the options available under the Bus Configuration Editor.

Table 3-2. Bus Configuration Characteristics

Characteristic	Description	Default Value
MXIbus System Controller	<p>A MXIbus link must have a single device that is responsible for MXIbus interrupt and bus arbitration as well as bus timeouts. This device is referred to as the <i>MXIbus System Controller</i> and is always the first device in the MXIbus daisy-chain. The NB-MXI is always designated as the MXIbus System Controller when used in this kit and should <i>not</i> be changed.</p> <p>0 = Not MXIbus System Controller 1 = MXIbus System Controller</p>	1
MXIbus Fair Request	<p>MXIbus fair requesters give all MXIbus devices equal opportunity to become bus masters and transfer data, regardless of their position within the MXIbus daisy-chain. MXIbus devices that are not fair requesters have a fixed priority scheme characterized by their relative position within the MXIbus daisy-chain. Devices closest to the MXIbus System Controller have the highest priority. The default configuration programs the NB-MXI to be an unfair requester and should be changed only if additional MXIbus devices are added to the system. Refer to the user manual included with your additional MXIbus devices for more information.</p> <p>0 = Not a MXIbus Fair Requester 1 = MXIbus Fair Requester</p>	0

Device Configuration

Table 3-3 describes the characteristics of the options available under the Device Configuration Editor.

Table 3-3. Device Configuration Characteristics

Characteristic	Description	Default Value
MXIbus System Controller Timeout	<p>When the NB-MXI is acting as the MXIbus System Controller, this field controls the amount of time the NB-MXI waits to assert BERR* on the MXIbus during any MXIbus cycle that does not complete before the timeout is reached.</p> <p>When the NB-MXI is <i>not</i> acting as the MXIbus System Controller, this field has no effect.</p> <p>0 = 100 μs 1 = 200 μs 2 = 400 μs 3 = 800 μs 4 = 1.8 ms 5 = 3.2 ms 6 = 6.4 ms 7 = 12.8 ms</p>	1

Non-VXI Device Editor

`VXItedit` contains a Non-VXI Device Editor that you can use to add or modify the address space and interrupt requirements for VME devices in your system. The NI-VXI software needs information about the attributes of the VME devices so that the Resource Manager can integrate them into the system. To select this editor from the `VXItedit` main menu, enter the option number or type in the editor name. For help with the Non-VXI Device Editor, type `help`. Refer also to the *NI-VXI Text Utilities Reference Manual* for more information about this editor.

You need to assign VME devices pseudo-logical addresses (between 256 and 511 inclusive), as well as device names, manufacturer names, and pseudo-model names. The `Frame` field should contain the logical address of the VME-MXI in the chassis. The default logical address is 1.

Type `list` to examine all the entries. Type `display` for a particular entry to examine all the fields of the particular entry. To edit any existing entry, type `modify` and give the index of the entry and the field to modify. The program will prompt you for the new field value. After you modify the entry, `VXItedit` sorts the list in alphabetical order. Make sure to set or clear the `InSystem` field of an entry, depending on whether that device is actually present in your system.

An easy way to create a new entry is to first duplicate a similar entry by typing `duplicate` and giving the index of the existing entry. After you duplicate an entry, modify its pseudo-logical address and the other address and interrupt requirements. Optionally, you can use the `add` command and enter all the values for a new entry.

You can delete any entry in the list by typing `delete` and giving the index of the entry. You can save the information you have modified by typing `save`, or type `abort` to ignore the modifications.

As an example, assume you have a VME memory device in your system that requires 64 kilobytes of memory in A24 space. Assign it a pseudo-logical address between 256 and 511. Set its `Frame` field to the logical address of the VME-MXI in the chassis (the default logical address for the VME-MXI in your kit is 1). Next, assign it a device name, manufacturer name, and model name. Finally, set its A24 memory base and size fields according to its memory requirements, and set the `InSystem` field to 1.

Run the Resource Manager to automatically configure the system. Verify that the Resource Manager display reflects all the changes you have made to accommodate the VME devices in your system.

Exiting VXItedit and Reinitializing the Hardware

To exit `VXItedit`, type `exit` or the number 11 when the main menu is displayed. If you changed any information, the program prompts you to save your changes before exiting the menu. Saving the configuration information updates the configuration files `vxila.cfg`, `vxibus.cfg`, and `vxidev.cfg`. Run `VXIinit` to reinitialize the hardware according to the new settings.

If you changed any of the software configuration settings from the default settings, record the new settings on the *VME-NB2040 Hardware and Software Configuration Form* in Appendix B.

Developing Your Application Program

After you have verified that the driver software has been installed successfully, you can proceed with development of your VME application software.

Using the Interactive Utilities

The interactive utilities described in the *NI-VXI Text Utilities Reference Manual* are very useful for developing applications. They are also useful for getting a feel for VXI communication protocols, configuring the VMEbus system, and debugging prototype instruments.

Using the C Language Interface

The file `cvxi.c` interfaces between your THINK C application program and the operating system entry points to the VXI driver. The functions available in `cvxi.c` are described in detail in the *NI-VXI C Software Reference Manual for VME*. To use these functions, you must compile `cvxi.c` and include the resulting object file during the link phase of each application compile operation.

The `DATASIZE.H` file defines the symbol `DIRECT`. If you want your application to use the Macintosh Device Manager instead of direct function calls, remove the definition of `DIRECT`.

The symbol `VXIMAC` must be defined for your application. It is recommended that you also define `__PROTOTYPES__` for full type checking of NI-VXI functions.

The `Example Folder` contains various example programs along with the sample project that show how to use various functions in the NI-VXI software and how to develop application programs using these functions.

Using LabVIEW

For information on developing VME application programs using LabVIEW, refer to the *LabVIEW VXI VI Reference Manual*.

To use LabVIEW to program your system, you must order the LabVIEW for Macintosh VXI Development System. Refer to the *Optional Software* section in Chapter 1, *Introduction*.

Appendix A

Specifications

This appendix lists various module specifications of the NB-MXI and VME-MXI, such as physical dimensions and power requirements.

NB-MXI

The following pages list the specifications for the NB-MXI module.

Capability Codes

MXIbus

Capability Code	Description
MA32	Master Mode A32, A24 and A16 addressing
MBLT	Master Mode block transfers
SA32	Slave Mode A32, A24, and A16 addressing
SBLT	Slave Mode block transfers
MD32	Master Mode D32, D16, and D08 data sizes
SD32	Slave Mode D32, D16, and D08 data sizes
SC	Optional MXIbus System Controller
FAIR	Can be a fair MXIbus requester
LOCK	Can lock the MXIbus for indivisible transfers
TERM	Can terminate the MXIbus

NuBus

Capability Code	Description
NBMD32	Master D32, D16, and D08 data sizes
NBSD32	Slave D32, D16, and D08 data sizes
AM	Can function as a NuBus alternate master
LOCK	Can lock the NuBus for indivisible transfers
INT	Can interrupt the NuBus

Electrical

Source	Typical	Direct Current (max)
+5 VDC	2.5 A	3.5 A

Environmental

Component Temperature	0° to 70° C (32° to 158° F) operating -55° to 150° C (-67° to 302° F) storage
Emissions	FCC Class A
Relative Humidity	0% to 95% noncondensing; operating 0% to 100% noncondensing; storage
Safety	Not applicable
Shock and Vibration	Not applicable

Physical

Board Size	Standard full-length NuBus PC-style board 12.858 in. by 4.0 in. (326.6 mm by 101.6 mm)
Connectors	Single fully implemented MXIbus connector
Slot Requirements	Single NuBus PC-style slot

Reliability

MTBF Contact Factory

Requirements

Memory space required 256 MB (Super Slot Space)

I/O space required 16 MB (Slot Space)

Timing

Master Mode

Transfer Type	Transfer Rate
Write	935 ns
Read	835 ns
Block Write	735 ns
Block Read	535 ns

Slave Mode

Transfer Type	Transfer Rate
Write	435 ns
Read	535 ns
Block Write	435 ns
Block Read	535 ns

Other

Daisy-Chain Delay 120 ns
(Passing GIN to GOUT or GOUT generation from System Controller)

VME-MXI

The following pages list the specifications for the VME-MXI module.

Capability Codes

VMEbus

Capability Code	Description
MA32, MA24, MA16	Master Mode A32, A24, and A16 addressing
SA32, SA24, SA16	Slave Mode A32, A24, and A16 addressing
MD32, MD16, MD08(EO)	Master Mode D32, D16, and D08 data sizes
SD32, SD16, SD08(EO)	Slave Mode D32, D16, and D08 data sizes
MBLOCK	Master Mode block transfers
SBLOCK	Slave Mode block transfers
MRMW	Master Mode Read/Modify/Write
SRMW	Slave Mode Read/Modify/Write
PRI	Prioritized arbitration
ROR	Release on Request bus requester
IH	Interrupt Handler
IR	Interrupt Requester
ROAK	Release on Acknowledge interrupter
BTO	Bus Timeout
SC	Optional VMEbus System Controller
IACK	IACK daisy-chain driver

MXIbus

Capability Code	Description
MA32, MA24, MA16	Master Mode A32, A24, and A16 addressing
SA32, SA24, SA16	Slave Mode A32, A24, and A16 addressing
MD32, MD16, MD08(EO)	Master Mode D32, D16, and D08 data sizes
SD32, SD16, SD08(EO)	Slave Mode D32, D16, and D08 data sizes
MBLOCK	Master Mode block transfers
SBLOCK	Slave Mode block transfers
SC	Optional MXIbus System Controller
FAIR	Optional MXIbus fair requester
TERM	Can accept MXIbus termination resistors
IH	Interrupt Handler
IR	Interrupt Requester

Electrical

Source	Typical	Maximum
+5 VDC	5.25 A	7.0 A

Environmental

Component temperature	0° to 70° C operating; -40° to 85° C storage
Relative humidity	10% to 90% noncondensing operating; 0% to 95% noncondensing storage
Emissions	FCC Class A
Safety	Not applicable
Shock and Vibration	Not applicable

Physical

Board size	9.187 in. by 6.299 in. 233.35 mm by 160 mm
Connectors	Single fully implemented MXIbus connector Single INTX connector (on boards equipped with optional INTX daughter card)
Slot Requirements	Single slot
VXI Keying Class	Class 1 TTL
Fully compatible with VMEbus specification	

Reliability

MTBF	Contact Factory
------	-----------------

Requirements

A16 Space	64 bytes
-----------	----------

Timing

Master Mode

Transfer Type	Transfer Rate
Write	675.5 ns
Read	625.5 ns
Block Write	320 ns
Block Read	279 ns

Slave Mode

Transfer Type	Transfer Rate
Write	381 ns
Read	381 ns
Block Write	238 ns
Block Read	238 ns

Other

Daisy-Chain Delay (Passing GIN to GOUT or GOUT generation from System Controller)	120 ns
--	--------

Appendix B

Customer Communication

For your convenience, this appendix contains forms to help you gather the information necessary to help us solve technical problems you might have as well as a form you can use to comment on the product documentation. Filling out a copy of the *Technical Support Form* before contacting National Instruments helps us help you better and faster.

National Instruments provides comprehensive technical assistance around the world. In the U.S. and Canada, applications engineers are available Monday through Friday from 8:00 a.m. to 6:00 p.m. (central time). In other countries, contact the nearest branch office. You may fax questions to us at any time.

Corporate Headquarters

(512) 795-8248

Technical support fax: (800) 328-2203
(512) 794-5678

Branch Offices	Phone Number	Fax Number
Australia	(03) 879 9422	(03) 879 9179
Austria	(0662) 435986	(0662) 437010-19
Belgium	02/757.00.20	02/757.03.11
Denmark	45 76 26 00	45 76 71 11
Finland	(90) 527 2321	(90) 502 2930
France	(1) 48 14 24 00	(1) 48 14 24 14
Germany	089/741 31 30	089/714 60 35
Italy	02/48301892	02/48301915
Japan	(03) 3788-1921	(03) 3788-1923
Netherlands	03480-33466	03480-30673
Norway	32-848400	32-848600
Spain	(91) 640 0085	(91) 640 0533
Sweden	08-730 49 70	08-730 43 70
Switzerland	056/20 51 51	056/20 51 55
U.K.	0635 523545	0635 523154

Technical Support Form

Photocopy this form and update it each time you make changes to your software or hardware, and use the completed copy of this form as a reference for your current configuration. Completing this form accurately before contacting National Instruments for technical support helps our applications engineers answer your questions more efficiently.

If you are using any National Instruments hardware or software products related to this problem, include the configuration forms from their user manuals. Use additional pages if necessary.

Name _____

Company _____

Address _____

Fax (____) _____ Phone (____) _____

Computer brand _____ Model _____ Processor _____

Operating system _____

Speed _____MHz RAM _____MB Display adapter _____

Mouse _____yes _____no Other adapters installed _____

Hard disk capacity _____MB Brand _____

Instruments used _____

National Instruments hardware product model _____ Revision _____

Configuration _____

National Instruments software product _____ Version _____

Configuration _____

The problem is _____

List any error messages _____

The following steps will reproduce the problem _____

VME-NB2040 Hardware and Software Configuration Form

Record the settings and revisions of your hardware and software on the line to the right of each item. Complete a new copy of this form each time you revise your software or hardware configuration, and use this form as a reference for your current configuration. Completing this form accurately before contacting National Instruments for technical support helps our applications engineers answer your questions more efficiently.

National Instruments Products

- NI-VXI Software Revision Number _____
(Disk Label: *NI-VXI Distribution Disk for the NB-MXI and the Macintosh*)
- Programming Language Interface Revision _____
- NB-MXI Logical Address _____
- NB-MXI Device Type _____
- NB-MXI Address Space _____
- NB-MXI Shared Memory Size _____
- NB-MXI Shared Memory Buffer _____
- NB-MXI Resource Manager Delay _____
- NB-MXI Servant Area Size _____
- NB-MXI Protocol Register _____
- NB-MXI Response for Servant-Side
Word Serial *Read Protocol* Query _____
- Number of NB-MXI Handlers _____
- Number of NB-MXI Interrupters _____
- MXIbus Fair Requester? _____
- MXIbus System Controller Timeout _____
- VME-MXI Installed in Slot 1? _____
- VME-MXI is VMEbus System Controller? _____
- VME-MXI Address _____
- VME-MXI VMEbus Request Level _____
- VMEbus BTO Timeout Value _____

- VME-MXI Hardware Revision
- NB-MXI Hardware Revision

Other Products

- Computer Make and Model
- Microprocessor
- Clock Frequency (Bus and Microprocessor)
- Total Memory in System
- Type of Video Board Installed
- Macintosh Operating System Version
- Programming Language Version
- Other Boards in System
- VMEbus Chassis Make and Model
- Other VME/VXibus Devices in System
- Static Logical Addresses of Other Devices

Documentation Comment Form

National Instruments encourages you to comment on the documentation supplied with our products. This information helps us provide quality products to meet your needs.

Title: **Getting Started with Your VME-NB2040 and the NI-VXI™ Software for Macintosh**

Edition Date: **July 1994**

Part Number: **320414B-01**

Please comment on the completeness, clarity, and organization of the manual.

If you find errors in the manual, please record the page numbers and describe the errors.

Thank you for your help.

Name _____

Title _____

Company _____

Address _____

Phone (_____) _____

Mail to: Technical Publications
National Instruments Corporation
6504 Bridge Point Parkway, MS 53-02
Austin, TX 78730-5039

Fax to: Technical Publications
National Instruments Corporation
MS 53-02
(512) 794-5678

Glossary

Prefix	Meaning	Value
n-	nano-	10^{-9}
μ -	micro-	10^{-6}
m-	milli-	10^{-3}
k-	kilo-	10^3
M-	mega-	10^6
g-	giga-	10^9

Symbols

- ° degrees
- % percent
- ± plus or minus

A

- A amperes
- A16 space VXIbus address space equivalent to the VME 64 KB *short* address space. In VXI, the upper 16 KB of A16 space is allocated for use by VXI devices configuration registers. This 16 KB region is referred to as VXI configuration space.
- A24 space VXIbus address space equivalent to the VME 16 MB *standard* address space.
- A32 space VXIbus address space equivalent to the VME 4 GB *extended* address space.
- address Character code that identifies a specific location (or series of locations) in memory.
- address modifier One of six signals in the VMEbus specification used by VMEbus masters to indicate the address space in which a data transfer is to take place.
- address space A set of 2^n memory locations differentiated from other such sets in VXI/VMEbus systems by six addressing lines known as address modifiers. n is the number of address lines required to uniquely specify a byte location in a given space. Valid numbers for n are 16, 24, and 32.

In VME/VXI, because there are six address modifiers, there are 64 possible address spaces.

ANSI American National Standards Institute

B

backplane An assembly, typically a printed circuit board, with 96-pin connectors and signal paths that bus the connector pins.

block-mode transfer An uninterrupted transfer of data elements in which the master sources only the first address at the beginning of the cycle. The slave is then responsible for incrementing the address on subsequent transfers so that the next element is transferred to or from the proper storage location. In VME, the data transfer may have no more than 256 elements; MXI does not have this restriction.

BTO unit Bus Timeout unit; a functional module that times the duration of each data transfer and terminates the cycle if the duration is excessive. Without the termination capability of this module, a bus master attempt to access a nonexistent slave could result in an indefinitely long wait for a slave response.

bus master A device that is capable of requesting the Data Transfer Bus (DTB) for the purpose of accessing a slave device.

C

C Celsius

CLK10 A 10-MHz differential ECL system clock used to synchronize trigger lines.

CMOS Complementary Metal Oxide Semiconductor; a process used in making chips.

Commander A Message-Based device which is also a bus master and can control one or more Servants.

D

daisy-chain A method of propagating signals along a bus, in which the devices are prioritized on the basis of their position on the bus.

Data Transfer Bus DTB; one of four buses on the VMEbus backplane. The DTB is used by a bus master to transfer binary data between itself and a slave device.

DIP Dual Inline Package

DTB	<i>See Data Transfer Bus.</i>
dynamic configuration	A method of automatically assigning logical addresses to VXIbus devices at system startup or other configuration times.
dynamically configured device	A device that has its logical address assigned by the Resource Manager. A VXI device initially responds at Logical Address 255 when its MODID line is asserted. A MXIbus device responds at Logical Address 255 during a priority select cycle. The Resource Manager subsequently assigns it a new logical address, which the device responds to until powered down.
E	
ECL	Emitter-Coupled Logic
embedded controller	An intelligent CPU (controller) interface plugged directly into the VXI backplane, giving it direct access to the VXIbus. It must have all of its required VXI interface capabilities built in.
F	
F	Fahrenheit
fair requester	A MXIbus master that will not arbitrate for the MXIbus after releasing it until it detects the bus request signal inactive. This ensures that all requesting devices will be granted use of the bus.
FCC	Federal Communications Commission
G	
GIN	Daisy-chain Grant In signal
GOUT	Daisy-chain Grant Out signal
H	
hex	Hexadecimal; the numbering system with base 16, using the digits 0 to 9 and letters A to F.
Hz	hertz; cycles per second.

I

IC	Integrated Circuit
IEEE	Institute of Electrical and Electronics Engineers
in.	inches
I/O	input/output; the techniques, media, and devices used to achieve communication between machines and users.
interrupt	A means for a device to request service from another device.
interrupt handler	A VMEbus functional module that detects interrupt requests generated by Interrupters and responds to those requests by requesting status and identify information.
INTX	Interrupt Timing and Extension

K

KB	1,024; kilobytes of memory
----	----------------------------

L

LabVIEW	Laboratory Virtual Instrument Engineering Workbench
LED	Light Emitting Diode
logical address	An 8-bit number that uniquely identifies each VXIbus device in a system. It defines the A16 register address of a device, and indicates Commander and Servant relationships.

M

MB	1,048,576; megabytes of memory
m	meters
master	A functional part of a MXI/VME/VXIbus device that initiates data transfers on the backplane. A transfer can be either a read or a write.
master-mode operation	A device is in master mode if it is performing a bus cycle which it initiated.
Message-Based device	An intelligent device that implements the defined VXIbus registers and communication protocols. These devices are able to use Word Serial Protocol to communicate with one another through communication registers.

MODID	Module Identification lines
MTBF	Mean Time Between Failure
MXIbus	Multisystem eXtension Interface Bus; a high-performance communication link that interconnects devices using round, flexible cables.
MXIbus System Controller	A functional module that has arbiter, daisy-chain driver, and MXIbus cycle timeout responsibility. Always the first device in the MXIbus daisy-chain.
N	
NI-VXI	The National Instruments bus interface software for VME/VXIbus systems.
P	
propagation	The transmission of signal through a computer system.
R	
Register-Based device	A Servant-only device that supports VXIbus configuration registers. Register-Based devices are typically controlled by Message-Based devices via device-dependent register reads and writes.
Resource Manager	A Message-Based Commander located at Logical Address 0, which provides configuration management services such as address map configuration, Commander and Servant mappings, and self-test and diagnostic management.
RM	See <i>Resource Manager</i> .
S	
s	seconds
Servant	A device controlled by a Commander; there are Message-Based and Register-Based Servants.
Shared Memory Protocol	A communication protocol that uses a block of memory that is accessible to both a client and a server. The memory block operates as a message buffer for communications.

slave	A functional part of a MXI/VME/VXIbus device that detects data transfer cycles initiated by a VMEbus master and responds to the transfers when the address specifies one of the device's registers.
slave-mode operation	A device is in slave mode if it is responding to a bus cycle.
Slot 0 device	A device configured for installation in Slot 0 of a VXIbus mainframe. This device is unique in the VXIbus system in that it performs the VMEbus System Controller functions, including clock sourcing and arbitration for data transfers across the backplane. Installing such a device into any other slot can damage the device, the VXIbus backplane, or both.
statically configured device	A device whose logical address cannot be set through software; that is, it is not dynamically configurable.

T

trigger	Either TTL or ECL lines used for intermodule communication.
TTL	Transistor-Transistor Logic

V

VDC	volts direct current
VI	virtual instrument. LabVIEW program; so called because it models the appearance and function of a physical instrument
VICtext	VXI Interactive Control Program, a part of the NI-VXI bus interface software package. Used to program VXI devices, and develop and debug VXI application programs.
VME	Versa Module Eurocard or IEEE 1014
VMEbus System Controller	A device configured for installation in Slot 1 of a VMEbus chassis. This device is unique in the VMEbus system in that it performs the VMEbus System Controller functions, including clock sourcing and arbitration for data transfers across the backplane. Installing such a device into any other slot can damage the device, the VMEbus backplane, or both.
VXIbus	VMEbus Extensions for Instrumentation
VXIteedit	VXI Resource Editor program, a part of the NI-VXI bus interface software package. Used to configure the system, edit the manufacturer name and ID numbers, edit the model names of VXI and non-VXI devices in the system, as well as the system interrupt configuration information, and display the system configuration information generated by the Resource Manager.

W

Word Serial
Protocol

The simplest required communication protocol supported by Message-Based devices in the VXIbus system. It utilizes the A16 communication registers to transfer data using a simple polling handshake method.