

# Serial

---

## Getting Started with Your PCMCIA Serial Hardware and Software for Windows Me/9x



September 2000 Edition  
Part Number 321827C-01

## **Worldwide Technical Support and Product Information**

ni.com

### **National Instruments Corporate Headquarters**

11500 North Mopac Expressway Austin, Texas 78759-3504 USA Tel: 512 794 0100

### **Worldwide Offices**

Australia 03 9879 5166, Austria 0662 45 79 90 0, Belgium 02 757 00 20, Brazil 011 284 5011,  
Canada (Calgary) 403 274 9391, Canada (Ontario) 905 785 0085, Canada (Québec) 514 694 8521,  
China 0755 3904939, Denmark 45 76 26 00, Finland 09 725 725 11, France 01 48 14 24 24,  
Germany 089 741 31 30, Greece 30 1 42 96 427, Hong Kong 2645 3186, India 91805275406,  
Israel 03 6120092, Italy 02 413091, Japan 03 5472 2970, Korea 02 596 7456, Mexico (D.F.) 5 280 7625,  
Mexico (Monterrey) 8 357 7695, Netherlands 0348 433466, New Zealand 09 914 0488, Norway 32 27 73 00,  
Poland 0 22 528 94 06, Portugal 351 1 726 9011, Singapore 2265886, Spain 91 640 0085,  
Sweden 08 587 895 00, Switzerland 056 200 51 51, Taiwan 02 2528 7227, United Kingdom 01635 523545

For further support information, see the *Technical Support Resources* appendix. To comment on the documentation, send e-mail to [techpubs@ni.com](mailto:techpubs@ni.com)

© Copyright 1998, 2000 National Instruments Corporation. All rights reserved.

# Important Information

---

## Warranty

The serial hardware is 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 document 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

CVI™, LabVIEW™, National Instruments™, and ni.com™ are trademarks of National Instruments Corporation.

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

## WARNING REGARDING USE OF NATIONAL INSTRUMENTS PRODUCTS

(1) NATIONAL INSTRUMENTS PRODUCTS ARE NOT DESIGNED WITH COMPONENTS AND TESTING FOR A LEVEL OF RELIABILITY SUITABLE FOR USE IN OR IN CONNECTION WITH SURGICAL IMPLANTS OR AS CRITICAL COMPONENTS IN ANY LIFE SUPPORT SYSTEMS WHOSE FAILURE TO PERFORM CAN REASONABLY BE EXPECTED TO CAUSE SIGNIFICANT INJURY TO A HUMAN.

(2) IN ANY APPLICATION, INCLUDING THE ABOVE, RELIABILITY OF OPERATION OF THE SOFTWARE PRODUCTS CAN BE IMPAIRED BY ADVERSE FACTORS, INCLUDING BUT NOT LIMITED TO FLUCTUATIONS IN ELECTRICAL POWER SUPPLY, COMPUTER HARDWARE MALFUNCTIONS, COMPUTER OPERATING SYSTEM SOFTWARE FITNESS, FITNESS OF COMPILERS AND DEVELOPMENT SOFTWARE USED TO DEVELOP AN APPLICATION, INSTALLATION ERRORS, SOFTWARE AND HARDWARE COMPATIBILITY PROBLEMS, MALFUNCTIONS OR FAILURES OF ELECTRONIC MONITORING OR CONTROL DEVICES, TRANSIENT FAILURES OF ELECTRONIC SYSTEMS (HARDWARE AND/OR SOFTWARE), UNANTICIPATED USES OR MISUSES, OR ERRORS ON THE PART OF THE USER OR APPLICATIONS DESIGNER (ADVERSE FACTORS SUCH AS THESE ARE HEREAFTER COLLECTIVELY TERMED "SYSTEM FAILURES"). ANY APPLICATION WHERE A SYSTEM FAILURE WOULD CREATE A RISK OF HARM TO PROPERTY OR PERSONS (INCLUDING THE RISK OF BODILY INJURY AND DEATH) SHOULD NOT BE RELIANT SOLELY UPON ONE FORM OF ELECTRONIC SYSTEM DUE TO THE RISK OF SYSTEM FAILURE. TO AVOID DAMAGE, INJURY, OR DEATH, THE USER OR APPLICATION DESIGNER MUST TAKE REASONABLY PRUDENT STEPS TO PROTECT AGAINST SYSTEM FAILURES, INCLUDING BUT NOT LIMITED TO BACK-UP OR SHUT DOWN MECHANISMS. BECAUSE EACH END-USER SYSTEM IS CUSTOMIZED AND DIFFERS FROM NATIONAL INSTRUMENTS' TESTING PLATFORMS AND BECAUSE A USER OR APPLICATION DESIGNER MAY USE NATIONAL INSTRUMENTS PRODUCTS IN COMBINATION WITH OTHER PRODUCTS IN A MANNER NOT EVALUATED OR CONTEMPLATED BY NATIONAL INSTRUMENTS, THE USER OR APPLICATION DESIGNER IS ULTIMATELY RESPONSIBLE FOR VERIFYING AND VALIDATING THE SUITABILITY OF NATIONAL INSTRUMENTS PRODUCTS WHENEVER NATIONAL INSTRUMENTS PRODUCTS ARE INCORPORATED IN A SYSTEM OR APPLICATION, INCLUDING, WITHOUT LIMITATION, THE APPROPRIATE DESIGN, PROCESS AND SAFETY LEVEL OF SUCH SYSTEM OR APPLICATION.

# Compliance

---

## FCC/Canada Radio Frequency Interference Compliance\*

### Determining FCC Class

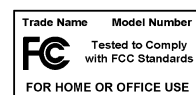
The Federal Communications Commission (FCC) has rules to protect wireless communications from interference. The FCC places digital electronics into two classes. These classes are known as Class A (for use in industrial-commercial locations only) or Class B (for use in residential or commercial locations). Depending on where it is operated, this product could be subject to restrictions in the FCC rules. (In Canada, the Department of Communications (DOC), of Industry Canada, regulates wireless interference in much the same way.)

Digital electronics emit weak signals during normal operation that can affect radio, television, or other wireless products. By examining the product you purchased, you can determine the FCC Class and therefore which of the two FCC/DOC Warnings apply in the following sections. (Some products may not be labeled at all for FCC; if so, the reader should then assume these are Class A devices.)

FCC Class A products only display a simple warning statement of one paragraph in length regarding interference and undesired operation. Most of our products are FCC Class A. The FCC rules have restrictions regarding the locations where FCC Class A products can be operated.

FCC Class B products display either a FCC ID code, starting with the letters **EXN**, or the FCC Class B compliance mark that appears as shown here on the right.

Consult the FCC web site <http://www.fcc.gov> for more information.



### FCC/DOC Warnings

This equipment generates and uses radio frequency energy and, if not installed and used in strict accordance with the instructions in this manual and the CE Mark Declaration of Conformity\*\*, may cause interference to radio and television reception. Classification requirements are the same for the Federal Communications Commission (FCC) and the Canadian Department of Communications (DOC).

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

### Class A

#### Federal Communications Commission

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. 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.

#### Canadian Department of Communications

This Class A digital apparatus meets all requirements of the Canadian Interference-Causing Equipment Regulations.

Cet appareil numérique de la classe A respecte toutes les exigences du Règlement sur le matériel brouilleur du Canada.

### Class B

#### Federal Communications Commission

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.

- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

## **Canadian Department of Communications**

This Class B digital apparatus meets all requirements of the Canadian Interference-Causing Equipment Regulations.

Cet appareil numérique de la classe B respecte toutes les exigences du Règlement sur le matériel brouilleur du Canada.

## **European Union - Compliance to EEC Directives**

Readers in the EU/EEC/EEA must refer to the Manufacturer's Declaration of Conformity (DoC) for information\*\* pertaining to the CE Mark compliance scheme. The Manufacturer includes a DoC for most every hardware product except for those bought for OEMs, if also available from an original manufacturer that also markets in the EU, or where compliance is not required as for electrically benign apparatus or cables.

\* Certain exemptions may apply in the USA, see FCC Rules §15.103 **Exempted devices**, and §15.105(c). Also available in sections of CFR 47.

\*\* The CE Mark Declaration of Conformity will contain important supplementary information and instructions for the user or installer.

# Contents

---

## About This Manual

Conventions .....	xi
Related Documentation.....	xii

## Chapter 1

### Introduction

How to Use This Manual .....	1-1
What You Need to Get Started .....	1-2
Optional Equipment .....	1-2
Serial Hardware Overview.....	1-2
NI-Serial Software Overview .....	1-3
Time-Saving Development Tools .....	1-4
Using Your Serial Hardware with LabVIEW and LabWindows/CVI.....	1-4

## Chapter 2

### Installation and Verification

Install the Software .....	2-1
Install the Hardware.....	2-3
Verify the Installation .....	2-4
Verify the Hardware Resources.....	2-6
Determine Which Physical Port Is Associated with COMx.....	2-7
Run the Diagnostic Test .....	2-7
Connect the Cables .....	2-8
Connecting Two-Wire Devices .....	2-9

## Chapter 3

### Configuration

View or Change Communication Port Settings .....	3-1
Port Settings Tab.....	3-2
Bits per Second.....	3-3
Data Bits .....	3-3
Parity.....	3-3
Stop Bits .....	3-3
Flow Control.....	3-3

Advanced Port Settings .....	3-3
Transceiver Mode .....	3-4
Use FIFO Buffers .....	3-4
Restore Defaults .....	3-4

## **Chapter 4**

### **Using Your Serial Hardware**

General Programming Requirements .....	4-1
Advanced Transceiver Control for the PCMCIA-485.....	4-1
Four-Wire Mode .....	4-2
Two-Wire Mode: $\overline{\text{DTR}}$ with Echo .....	4-2
Two-Wire Mode: $\overline{\text{DTR}}$ Controlled .....	4-2
Two-Wire Mode: $\overline{\text{TXRDY}}$ Auto Control.....	4-3
Setting the Transceiver Control Mode .....	4-3
Setting the Transceiver Mode with DeviceIoControl .....	4-4

## **Appendix A**

### **Serial Port Information**

## **Appendix B**

### **Uninstalling the Hardware and Software**

## **Appendix C**

### **Troubleshooting and Common Questions**

## **Appendix D**

### **Specifications**

## **Appendix E**

### **Technical Support Resources**

## **Glossary**

## **Index**

## Figures

Figure 2-1.	Add/Remove Programs Properties Dialog Box.....	2-1
Figure 2-2.	NI-Serial Setup Screen .....	2-2
Figure 2-3.	Inserting a PCMCIA Serial Card.....	2-3
Figure 2-4.	Device Manager Ports List for PCMCIA Serial Card Correctly Installed .....	2-5
Figure 2-5.	Device Manager Ports List for PCMCIA Serial Card Incorrectly Installed.....	2-6
Figure 2-6.	DB-9 Connector Pin Locations .....	2-8
Figure 3-1.	Port Settings Tab .....	3-2
Figure 3-2.	Advanced Port Settings Dialog Box.....	3-3
Figure A-1.	Typical Full-Duplex System .....	A-4
Figure A-2.	Typical Half-Duplex System.....	A-4
Figure A-3.	Multidrop Network Using Terminating Resistors.....	A-5
Figure A-4.	Straight-Through Cabling in a DTE-to-DCE Interface .....	A-6
Figure A-5.	Null-Modem Cabling in a DTE-to-DTE Interface .....	A-6
Figure B-1.	Selecting an Interface to Uninstall .....	B-2
Figure B-2.	Add/Remove Programs Properties Dialog Box.....	B-3
Figure B-3.	Successful Uninstallation .....	B-4
Figure C-1.	Ports List in Device Manager Tab.....	C-4
Figure C-2.	Multi-Function Adapter Class in the Device Manager .....	C-7

## Tables

Table 2-1.	DB-9 Pin Descriptions.....	2-8
Table 4-1.	Transceiver Control Modes .....	4-2
Table 4-2.	Transceiver Mode Control Bytes .....	4-3
Table 4-3.	DeviceIoControl Function Input Values .....	4-4
Table A-1.	RS-232, RS-422, and RS-485 Features .....	A-1
Table C-1.	Standard DOS-Based Addresses .....	C-5
Table D-1.	Physical Characteristics of the PCMCIA Serial Cards .....	D-1
Table D-2.	Environmental Characteristics of the Serial Hardware .....	D-2
Table D-3.	Software Characteristics.....	D-2



# About This Manual

---

This manual contains instructions to help you install and configure the National Instruments serial hardware and the NI-Serial software for Windows Me/9x. This manual includes information about the PCMCIA-232, PCMCIA-232/2, PCMCIA-232/4, PCMCIA-485, and PCMCIA-485/2.

This manual assumes that you are already familiar with Windows Me/9x.

## Conventions

---

The following conventions appear in this manual:

»

The » symbol leads you through nested menu items and dialog box options to a final action. The sequence **File»Page Setup»Options** directs you to pull down the **File** menu, select the **Page Setup** item, and select **Options** from the last dialog box.



This icon denotes a note, which alerts you to important information.

**bold**

Bold text denotes items that you must select or click on in the software, such as menu items and dialog box options. Bold text also denotes parameter names.

DTR

Signal names with an overscore, such as DTR, indicate that the signal is active low.

*italic*

Italic text denotes variables, emphasis, a cross reference, or an introduction to a key concept. This font also denotes text that is a placeholder for a word or value that you must supply.

`monospace`

Text in this font denotes text or characters that you should enter from the keyboard, sections of code, programming examples, and syntax examples. This font is also used for the proper names of disk drives, paths, directories, programs, subprograms, subroutines, device names, functions, operations, variables, filenames and extensions, and code excerpts.

`monospace bold`

Bold text in this font denotes the messages and responses that the computer automatically prints to the screen. This font also emphasizes lines of code that are different from the other examples.

## Related Documentation

---

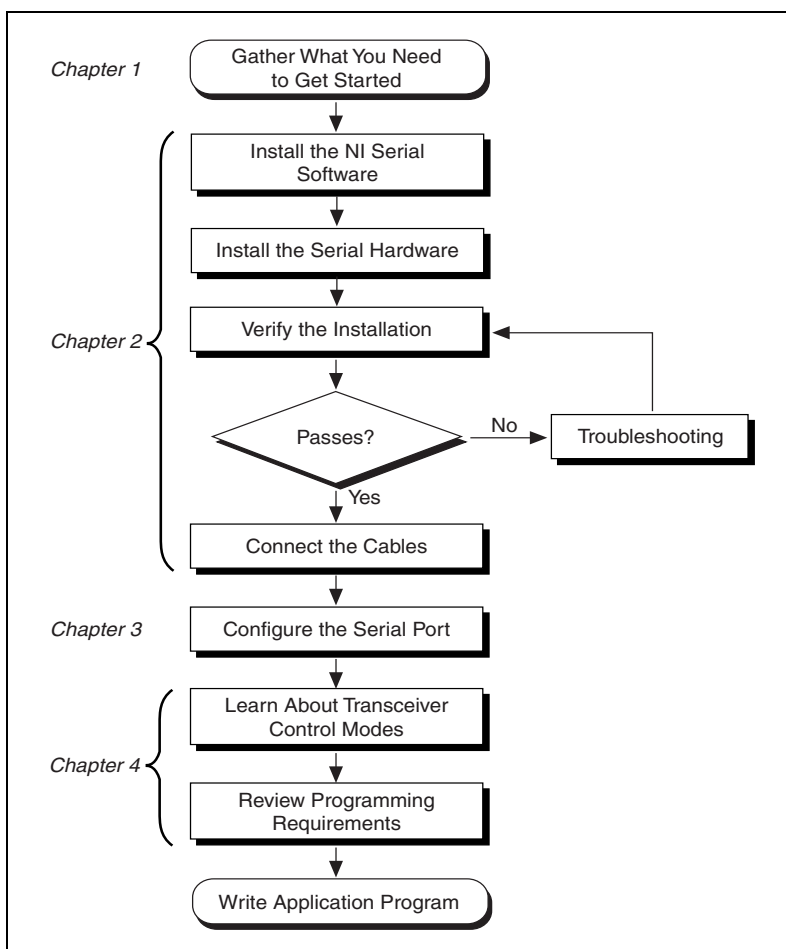
The following documents contain information that you might find helpful as you read this manual:

- ANSI/EIA-232-D Standard, *Interface Between Data Terminal Equipment and Data Circuit-Terminating Equipment Employing Serial Binary Data Interchange*
- EIA/RS-422-A Standard, *Electrical Characteristics of Balanced Voltage Digital Interface Circuits*
- EIA-485 Standard, *Standard for Electrical Characteristics of Generators and Receivers for Use in Balanced Digital Multipoint Systems*
- *Microsoft Win32 Software Developer Kit*, Online Documentation for *Win32 Overviews*, *Win32 Reference*, and *Microsoft Programmer's Guide to Windows 98/95*, Microsoft Corporation
- *NS16550AF Universal Asynchronous Receiver/Transmitter with FIFOs*, National Semiconductor
- *ST16C654 Asynchronous Receiver/Transmitter with FIFOs*, EXAR Corporation

# Introduction

This chapter explains how to use this manual, lists what you need to get started and optional equipment you can order, and briefly describes the serial hardware and the NI-Serial software.

## How to Use This Manual



## What You Need to Get Started

---

Before you install your serial hardware and the NI-Serial software, make sure you have all of the following items:

- Windows Me/9x installed on your computer
- One of the following serial interfaces, which is included in your kit:
  - PCMCIA-232 (one, two, or four port)
  - PCMCIA-485 (one or two port)
- CD, *NI-Serial Software for Windows 2000/NT/Me/9x*, which is included in your kit
- Adapter cable from PCMCIA card to DB-9 connector, which is included in your kit

## Optional Equipment

---

For more information about ordering the following optional equipment, contact National Instruments:

- DB-9 RS-485 termination connector
- RS-232 9-pin to 9-pin null modem cable
- RS-232 9-pin to 25-pin null modem cable
- RS-485 9-pin to 9-pin null modem cable

## Serial Hardware Overview

---

The serial hardware gives you a variety of solutions for serial communications. The PCMCIA-232 interfaces work with the RS-232 protocols, and the PCMCIA-485 interfaces work with the RS-422 and RS-485 protocols. You can use the PCMCIA-232 hardware for serial communication up to distances of 50 ft. You can connect the PCMCIA-485 hardware with up to 31 devices using serial cable lengths up to 4,000 ft.

The PCMCIA-232 is available with one, two, or four ports. The PCMCIA-485 is available in a one-port version or two-port version. All PCMCIA cards come with cables for each port that terminates in a standard DB-9 D-Sub connector. Throughout this manual, references to PCMCIA serial interfaces generally refer to all versions of the interfaces.

The PCMCIA-485 interfaces support four hardware transceiver control modes for reliable communication with two-wire and four-wire devices. Refer to Chapter 4, *Using Your Serial Hardware*, for more information about transceiver control modes.

All of the serial hardware uses standard 16550-compatible UARTs (Universal Asynchronous Receiver/Transmitters) for 100 percent compatibility with standard PC COM ports. The serial hardware contains FIFOs (First-In-First-Out buffers) for reduced susceptibility to interrupt latency and faster transmission rates. Full Plug and Play compatibility gives you the convenience of switchless configuration and installation. Refer to Appendix D, *Specifications*, for more information about the serial hardware specifications and operating conditions.

## NI-Serial Software Overview

---

The NI-Serial software for Windows Me/9x includes a native Windows device driver that provides full interrupt-driven, buffered I/O for multiple COM ports. You can obtain a maximum baud rate of 115.2 KBaud. You can also use any number of serial ports under Windows Me/9x. The NI-Serial software also includes a configuration utility, which is fully integrated into the Windows Me/9x Device Manager. For more information about software specifications, refer to Appendix D, *Specifications*.

The NI-Serial software includes the following components:

- Device driver
- Diagnostic utility
- Configuration utility
- Product manuals

The NI-Serial software supports all National Instruments serial hardware, including the AT, PCI, PXI/CompactPCI, and PCMCIA versions.

## Time-Saving Development Tools

---

Your kit includes the NI-Serial software for Windows Me/9x. In addition, you can order the LabWindows/CVI or LabVIEW software from National Instruments to speed your application development time and make it easier to communicate with your instruments.

LabVIEW is an easy-to-use, graphical programming environment you can use to acquire data from thousands of different instruments, including IEEE 488.2 devices, VXI devices, serial devices, PLCs, and plug-in data acquisition boards. After you have acquired raw data, you can convert it into meaningful results using the powerful data analysis routines in LabVIEW. LabVIEW also comes with hundreds of instrument drivers, which dramatically reduce software development time, because you do not have to spend time programming the low-level control of each instrument.

LabWindows/CVI is similar to LabVIEW, except that it combines an interactive, easy-to-use development approach with the programming power and flexibility of compiled ANSI C code.

For ordering information, or to request free demonstration software, contact National Instruments.

## Using Your Serial Hardware with LabVIEW and LabWindows/CVI

---

After you install your serial hardware and the NI-Serial software, you can use standard serial I/O functions in LabVIEW and LabWindows/CVI with your serial interface. If you already have LabVIEW or LabWindows/CVI and want to use it with your serial hardware, refer to your LabVIEW or LabWindows/CVI documentation for information about serial I/O functions.

---

# Installation and Verification

This chapter describes how to install the NI-Serial software, serial hardware, and cables, and how to verify the installation.

## Install the Software

---

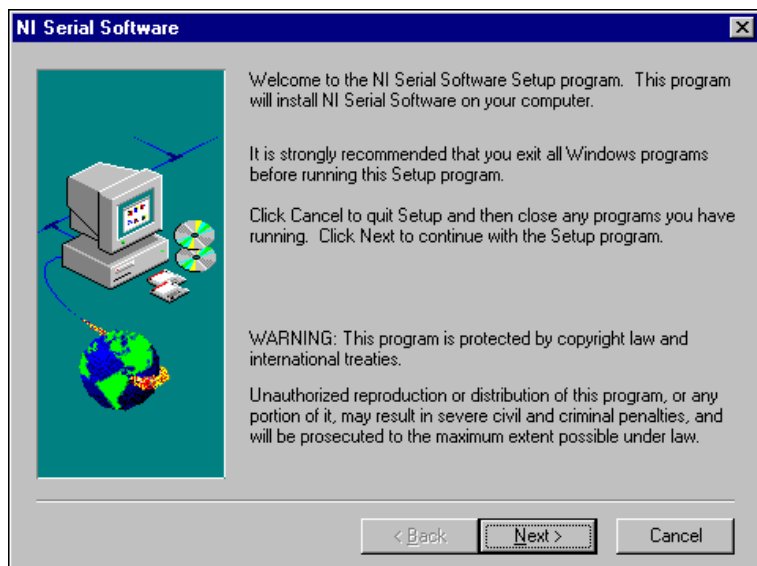
Install the NI-Serial software for Windows Me/9x before you install the hardware:

1. Select **Start»Settings»Control Panel**.
2. Double-click on the **Add/Remove Programs** icon in the Control Panel to launch the Add/Remove Programs applet. A dialog box similar to the one in Figure 2-1 appears.



**Figure 2-1.** Add/Remove Programs Properties Dialog Box

3. Click on the **Install** button.
4. When prompted, insert the *NI-Serial Software for Windows 2000/NT/Me/9x* CD and click on the **Next** button to proceed. The software installation wizard begins with the screen shown in Figure 2-2.



**Figure 2-2.** NI-Serial Setup Screen

The setup wizard guides you through the necessary steps to install the NI-Serial software. You can go back and change values where appropriate by clicking on the **Back** button. You can exit the setup where appropriate by clicking on the **Cancel** button.

5. If you need to install your hardware, or if this is your first time to install the NI-Serial software for Windows Me/9x, skip to the next section, [Install the Hardware](#). Otherwise, continue to Step 6.
6. If your hardware is already installed, restart Windows Me/9x.

Windows Me/9x should automatically detect your hardware and display one or more **New Hardware Found** dialog boxes. When you finish, continue to the [Verify the Installation](#) section.

If no **New Hardware Found** dialog box appears, refer to the section [Forcing Windows to Detect Your Hardware](#) in Appendix C, [Troubleshooting and Common Questions](#).



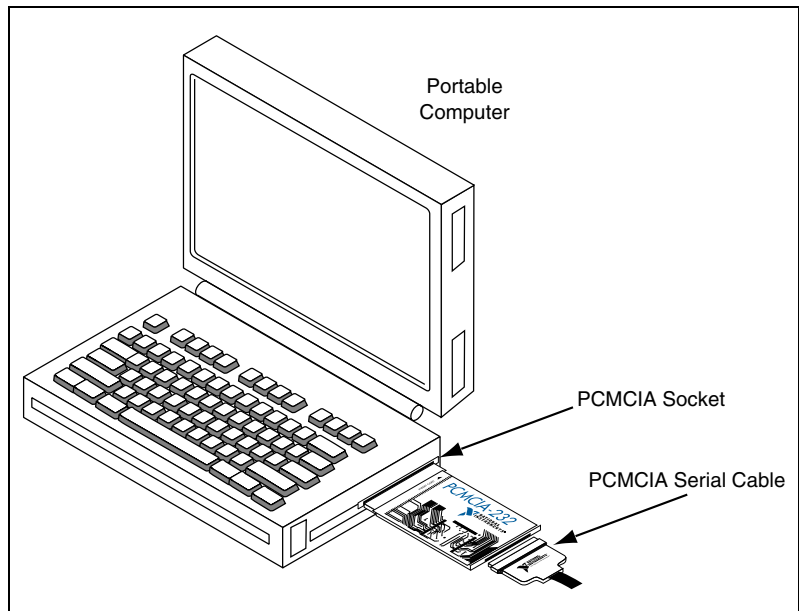
# Install the Hardware

To install the PCMCIA-232 or PCMCIA-485 in your computer, first start Windows Me/9x. If the card is not already inserted, insert it into a free PC Card (PCMCIA) socket the same way you insert a disk into a floppy drive. The card has no jumpers or switches to set, and you do not need to shut down the system before you insert it.

Windows Me/9x automatically displays one or more **New Hardware Found** dialog boxes when it detects and configures a PC Card.

If the **New Hardware Found** dialog box does not appear, refer to the [Forcing Windows to Detect Your Hardware](#) section in Appendix C, [Troubleshooting and Common Questions](#).

Figure 2-3 shows how to insert a PCMCIA serial card and how to connect the cable.



**Figure 2-3.** Inserting a PCMCIA Serial Card

The serial hardware installation is complete. Continue to the next section, [Verify the Installation](#).

## Verify the Installation

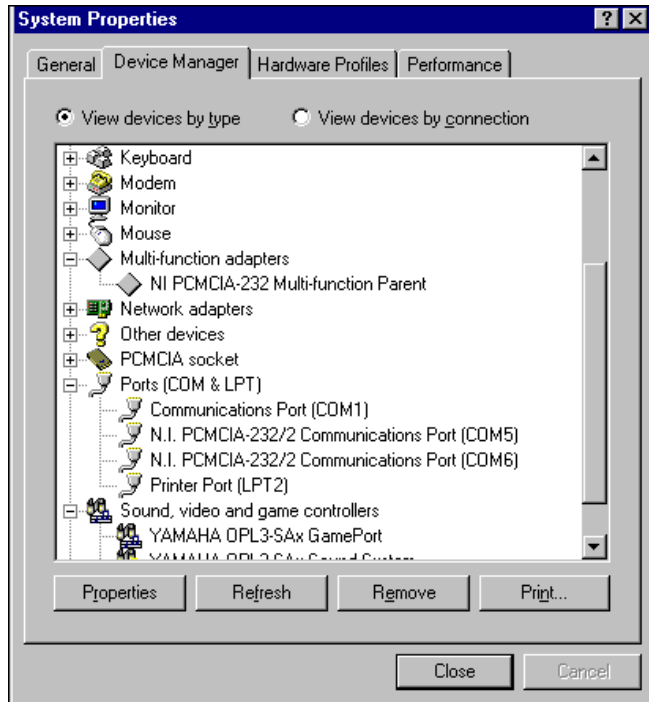
---

When you begin to verify the installation, keep in mind that the serial ports built into the computer are typically named from COM1 to COM4. Windows Me/9x typically issues port names to the ports on NI serial hardware starting with COM5, COM6, and so on. If not enough resources are available to assign to all the NI serial ports, the port with the higher COM $x$  name is left unconfigured.

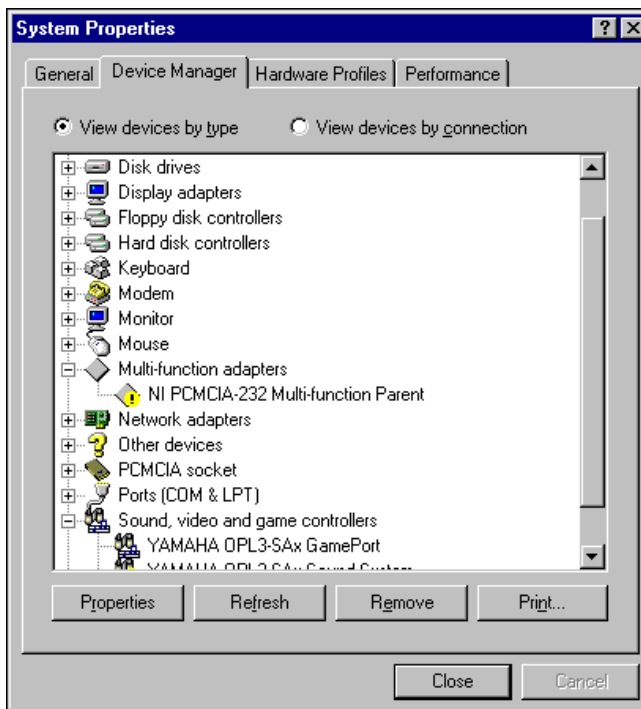
Double-click on the **System** icon in the Control Panel. In the **System Properties** window that appears, select the **Device Manager** tab, and click on the **View devices by type** button at the top of the page. If necessary, double-click on the **Ports (COM & LPT)** icon to view all of the ports.

The two-port and four-port PCMCIA cards are configured as devices belonging to the “multi-function adapters” class. The multi-function parent device is listed under the **Multi-function adapters** icon, and each child device is listed as a port under the **Ports (COM & LPT)** icon. If you have installed one of these interfaces, double-click on the **Multi-function adapters** icon as well as the **Ports (COM & LPT)** icon.

If a circled exclamation point appears through the port icon, the serial port is not installed properly. A problem with the port could have occurred because Windows Me/9x could not acquire resources for the port, or because an interrupt request resource conflict exists. If no circled exclamation point appears, the serial hardware is installed correctly. Figure 2-4 shows an example of NI serial hardware that is installed properly and Figure 2-5 shows an example of NI serial hardware that is *not* working properly.



**Figure 2-4.** Device Manager Ports List for PCMCIA Serial Card Correctly Installed



**Figure 2-5.** Device Manager Ports List for PCMCIA Serial Card Incorrectly Installed

## Verify the Hardware Resources

For every port of the newly installed serial interface, double-click on the name of the serial port in the Device Manager. Then, click on the **Resources** tab. If the resources were assigned correctly, the **Resources** page shows which resources are assigned to your serial ports.

If Windows Me/9x did not assign any resources to the port, the **Resources** page shows only a **Set Configuration Manually** button. Click on the **Set Configuration Manually** button and try to assign conflict-free resources to your serial hardware. For instructions, refer to the [Selecting Conflict-Free Resources](#) section of Appendix C, [Troubleshooting and Common Questions](#). You might have to free an interrupt request level so that one can be assigned to your serial hardware. For instructions, refer to the section [Freeing an Interrupt Request Level](#) in Appendix C, [Troubleshooting and Common Questions](#).

When you have finished verifying the hardware resources, proceed to the next section.

## Determine Which Physical Port Is Associated with COMx

For every port of the newly installed serial interface, double-click on the name of the serial port in the Device Manager. Then, click on the **Port Settings** tab. The top of this page displays the serial number of the serial hardware, and the physical port number starting at 1.

When you have finished verifying the driver installation and physical port, proceed to the next section.

## Run the Diagnostic Test

To verify and test the installation, run the diagnostic program that came with your NI-Serial software. The diagnostic program verifies that your serial driver is installed properly, that the configuration of your hardware does not conflict with anything else in your system, and that the serial driver can communicate with your hardware correctly.

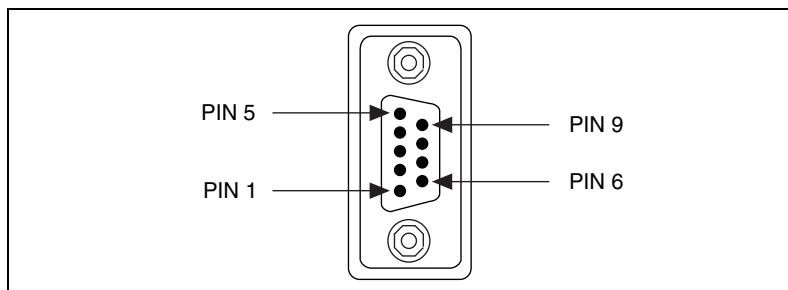
To run the test, select the **diagnostics** item under **Start»Programs»National Instruments Serial**.

If the diagnostic test completes with no failures, your serial hardware and software are installed properly. If it fails, refer to Appendix C, [Troubleshooting and Common Questions](#), for troubleshooting instructions.

## Connect the Cables

For the PCMCIA serial cards, you can use the standard DB-9 connector found on most serial cables.

Figure 2-6 and Table 2-1 give the pin locations and descriptions of the DB-9 connector, which is found on the PCMCIA serial cable.



**Figure 2-6.** DB-9 Connector Pin Locations

**Table 2-1.** DB-9 Pin Descriptions

DB-9 Pin	232 Signal	485 Signal
1	DCD*	GND
2	$\overline{\text{RXD}}$	CTS+ (HSI+)
3	$\overline{\text{TXD}}$	RTS+ (HSO+)
4	DTR*	RXD+
5	GND	RXD-
6	DSR*	CTS- (HSI-)
7	RTS	RTS- (HSO-)
8	CTS	TXD+
9	RI*	TXD-

\* These signals are not supported by the isolated 232 boards or ports 9-16 of the RS-232 sixteen-port board.

## Connecting Two-Wire Devices

---

The PCMCIA-485 boards are designed to work with either two- or four-wire devices. If you are using a two-wire device, refer to the device documentation for specific wiring instructions.

In general, half-duplex networks use a single twisted pair of wires for communication in both directions, so you must connect both the transmitter and the receiver at each end of the same pair of wires. For example, to connect an RS-485 data acquisition device to a port on your PCMCIA-485 using half-duplex communication, you need a single twisted pair of wires. At the PCMCIA-485, connect the TXD+ and RXD+ signals (pins 8 and 4 on a DB-9 connector) together and to one wire. Connect the other end of this wire to both the TXD+ and RXD+ signals on the data acquisition device. You use the same method to connect the TXD– and RXD– signals (pins 9 and 5 on a DB-9 connector) to the second wire.

For information about setting the transceiver mode for two-wire communication, refer to Chapter 4, *Using Your Serial Hardware*. For more information about duplex architectures, refer to Appendix A, *Serial Port Information*.

---

# Configuration

This chapter describes how to view or change the communication port settings.

---

## View or Change Communication Port Settings

---

The serial configuration utility is fully integrated with the Windows Me/9x Device Manager. You can use it to examine or modify the configuration of the serial port.

Follow these steps to configure the serial port. Repeat the configuration procedure for each serial port in your system.

1. Select **Start»Settings»Control Panel** and double-click on the **System** icon. The **System Properties** window appears.
2. Select the **Device Manager** tab, and click on the **View devices by type** button at the top of the page.
3. Double-click on the **Ports (COM & LPT)** icon.
4. Double-click on the name of the port you want to configure.

You can now view or change information about your serial port in the following ways:

- Click on the **Resources** tab to view information about the hardware resources assigned to the serial port.
- Click on the **Port Settings** tab to view information about the software configuration for the serial port. Refer to the next section, [Port Settings Tab](#), for more information.
- On the **Port Settings** tab, click on the **Advanced** button to change the RS-485 transceiver mode and to enable or disable the FIFOs on the serial hardware. Refer to the next section, [Port Settings Tab](#), for more information.



**Note** If you use two-wire  $\overline{\text{TXRDY}}$  mode, FIFOs must be enabled. Transceiver modes apply to RS-485 interfaces only. For more information about transceiver modes, refer to Chapter 4, [Using Your Serial Hardware](#).

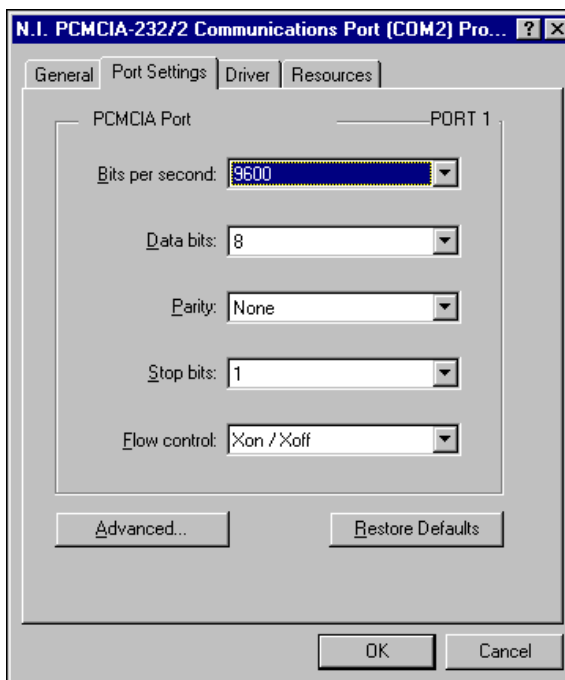


After you have selected the serial port parameters for the current port, click on the **OK** button to save the changes or click on the **Cancel** button to exit the dialog box without saving changes.

## Port Settings Tab

---

On the **Port Settings** tab, you can change any of the settings by clicking on the arrow button to the right of the setting. When you click on the arrow button, a list of valid values for that setting appears. Select the desired setting from the list. Figure 3-1 shows the **Port Settings** tab.



**Figure 3-1.** Port Settings Tab

The following sections describe the options you can set in the Device Manager **Port Settings** tab.

## Bits per Second

**Bits per second**, or baud rate, is the speed for a serial port.

## Data Bits

**Data bits** is the number of data bits in a single serial byte.

## Parity

**Parity** is the specification for even, odd, or no parity bits in each transmitted byte.

## Stop Bits

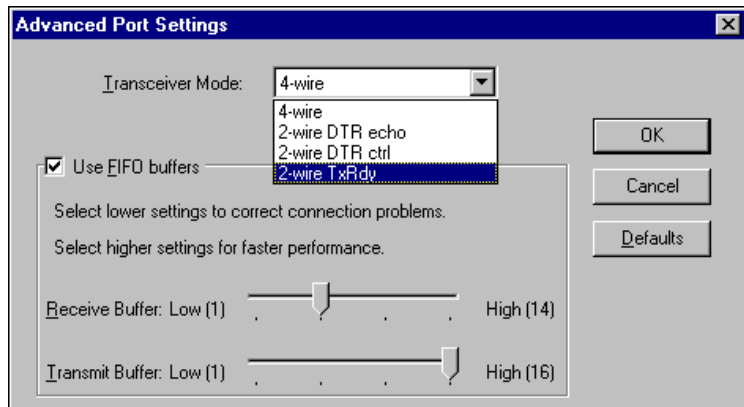
**Stop bits** is the number of terminating bits on the end of each transmitted serial byte.

## Flow Control

**Flow control** is a method for temporarily halting the stream of serial bytes to prevent overflow.

## Advanced Port Settings

You can view or change the advanced port settings by clicking on the **Advanced** button on the **Port Settings** tab. Figure 3-2 shows the **Advanced Port Settings** dialog box.



**Figure 3-2.** Advanced Port Settings Dialog Box

The following sections describe the advanced settings options.

## Transceiver Mode

This field shows the transceiver mode in use, and applies only to PCMCIA-485 boards. For more information about transceiver modes, refer to Chapter 4, *Using Your Serial Hardware*.

## Use FIFO Buffers

FIFO buffers are present on the 16550-compatible UARTs—one for the transmitter and one for the receiver. The **Receive Buffer** control sets the number of characters received in the FIFO before the PC is interrupted to read the data. The **Transmit Buffer** control sets the maximum number of bytes written to the FIFO in a block when the PC is interrupted to write the data. When you configure FIFO settings, consider the following points:

- You can select larger FIFO buffer sizes to reduce the number of interrupts your PC receives and therefore minimize system overhead.
- If transfer rates are high, you can lower the **Receive Buffer** value to prevent overrun errors due to interrupt latency.
- If your data transfer sizes are small and your **Receive Buffer** value is above your data sizes, your system is less efficient. Therefore, set the **Receive Buffer** value below your data sizes.



**Note** If you want your serial hardware ports to use the names COM1, COM2, COM3, or COM4, refer to the *Common Questions* section of Appendix C, *Troubleshooting and Common Questions*.

## Restore Defaults

To reset the fields to their default values at any time, click on the **Restore Defaults** button.

---

# Using Your Serial Hardware

This chapter describes how to set the hardware transceiver control mode for your RS-485 interfaces and lists some general programming requirements.

## General Programming Requirements

---

The NI-Serial software is fully integrated into the standard Windows Me/9x communications software. NI serial ports are used like any other Windows Me/9x communications (COM) port. Windows Me/9x has standard communication functions for use within either Win16 or Win32 applications.

When you develop your application, remember that you must use the standard Microsoft Windows serial communication functions. For information about Microsoft Windows serial communication functions, refer to the *Win32 Software Development Kit* and to the *Win32 Overviews and Win32 Reference* online help.

If you have LabVIEW or LabWindows/CVI and want to use it with your serial hardware, refer to your LabVIEW or LabWindows/CVI documentation for information about serial I/O functions.

## Advanced Transceiver Control for the PCMCIA-485

---



**Note** Transceiver modes apply only to the PCMCIA-485 boards.

The PCMCIA-485 boards support four modes of hardware transceiver control. You can use hardware flow control to enable and disable your transmitters and receivers so that they function on different bus topologies. Table 4-1 lists the status of the transmitters and receivers under each of the transceiver control modes.

**Table 4-1.** Transceiver Control Modes

Mode	Transmitter	Receiver
Four-wire mode	Always enabled	Always enabled
Two-wire mode: $\overline{\text{DTR}}$ with echo	Enabled with $\overline{\text{DTR}}$ unasserted	Always enabled
Two-wire mode: $\overline{\text{DTR}}$ controlled	Enabled with $\overline{\text{DTR}}$ unasserted	Enabled with $\overline{\text{DTR}}$ asserted
Two-wire mode: $\overline{\text{TXRDY}}$ auto control	Enabled with $\overline{\text{TXRDY}}$ asserted	Enabled with $\overline{\text{TXRDY}}$ unasserted



**Note** Signal names with an overscore, such as  $\overline{\text{DTR}}$ , indicate that the signal is active low.

## Four-Wire Mode

Use the four-wire mode for most full-duplex systems. In this mode, the transmitter and receiver are always enabled. This mode is the default.

## Two-Wire Mode: $\overline{\text{DTR}}$ with Echo

Use this mode in half-duplex systems where the  $\overline{\text{DTR}}$  (Data Terminal Ready) line must control the transmitter. In the  $\overline{\text{DTR}}$ -with-echo mode, the transmitter is tri-stated when the  $\overline{\text{DTR}}$  signal of the UART (Universal Asynchronous Receiver/Transmitter) is asserted. To transmit, your application must first clear the  $\overline{\text{DTR}}$  bit to enable the transmitter. After the data is fully transmitted, your application once again sets the  $\overline{\text{DTR}}$  bit to disable the transmitter. Because the receiver is always enabled in this mode, you not only receive packets from other devices, you also receive the packets sent from your transmitter.

## Two-Wire Mode: $\overline{\text{DTR}}$ Controlled

This mode is similar to the two-wire,  $\overline{\text{DTR}}$ -with-echo mode. Use this mode in half-duplex systems where the  $\overline{\text{DTR}}$  line must control the transmitter. Although this mode uses the same method as the  $\overline{\text{DTR}}$ -with-echo mode to control the transmitter, the hardware automatically disables the receiver whenever the transmitter is enabled. Thus, you do not receive the packets sent from your transmitter.

## Two-Wire Mode: $\overline{\text{TXRDY}}$ Auto Control

In this mode, the serial hardware transparently enables the transmitter and receiver in a two-wire system. Use this mode to remove the burden of flow control from your application. By connecting the transmitter to the  $\overline{\text{TXRDY}}$  (Transmit Ready) line, the hardware enables the transmitter for each byte to be transmitted. Also, the hardware disables the receiver whenever the transmitter is enabled, so you do not receive the packets sent from your transmitter.



**Note** When you are communicating with a two-wire device, National Instruments recommends that you use the two-wire  $\overline{\text{TXRDY}}$  auto control mode. Because this mode handles the transmitter/receiver enabling for a two-wire connection in your hardware, it reduces the software overhead required to perform this operation in your application program.

For more information about serial communication in two- or four-wire modes, refer to the [Serial Communication Issues](#) section in Appendix A, [Serial Port Information](#).

## Setting the Transceiver Control Mode

The recommended method for setting the transceiver control mode is with the serial configuration utility located in the Windows Me/9x Device Manager. For more information, refer to the [View or Change Communication Port Settings](#) section in Chapter 3, [Configuration](#). The mode you select in the Device Manager is automatically configured when you open a port on a serial interface. You can also set the hardware transceiver control mode from within a DOS application. For each port you want to control, write the control byte for the mode you want to select to the scratch register of the UART. Table 4-2 shows the control bytes for each mode.

**Table 4-2.** Transceiver Mode Control Bytes

Transceiver Mode	Control Byte
Four-wire mode	0x00
Two-wire mode: $\overline{\text{DTR}}$ with echo	0x01
Two-wire mode: $\overline{\text{DTR}}$ controlled	0x02
Two-wire mode: $\overline{\text{TXRDY}}$ auto control	0x03

The scratch register is located at offset 7 from the base address of the port. For example, if COM2 were located at base address 0x3F8, and you wanted to set the PCMCIA-485 board to two-wire mode with  $\overline{\text{DTR}}$  control, you would write a 0x02 to I/O address 0x3FF. The PCMCIA-485 board would immediately switch to the two-wire mode with  $\overline{\text{DTR}}$  control.

## Setting the Transceiver Mode with DeviceIoControl

The NI-Serial software extends the `DeviceIoControl` Windows function for programming the transceiver control mode. To program the transceiver control mode using `DeviceIoControl`, complete the following steps:

1. Add the following lines to your source code:
 

```
#include <winioctl.h>

#define IOCTL_SERIAL_SET_TRANSCEIVER_MODE
CTL_CODE(FILE_DEVICE_SERIAL_PORT, 37,
METHOD_BUFFERED, FILE_ANY_ACCESS)
```
2. Use the Win32 function `DeviceIoControl`, as follows:
  - a. Use the defined control code value listed in step 1 to set the transceiver mode.
  - b. Use the input buffer values (unsigned long) listed in Table 4-3 for programming different transceiver modes.

**Table 4-3.** DeviceIoControl Function Input Values

Transceiver Mode	DeviceIoControl Function Input Value
Four-wire mode	128
Two-wire mode: $\overline{\text{DTR}}$ with echo	129
Two-wire mode: $\overline{\text{DTR}}$ controlled	130
Two-wire mode: $\overline{\text{TXRDY}}$ auto control	131

For example, to set  $\overline{\text{TXRDY}}$  two-wire auto control mode, use the following code:

```
ULONG TranceiverMode = 131;
DeviceIoControl(hDevice,
IOCTL_SERIAL_SET_TRANSCEIVER_MODE, (PVOID)
&TranceiverMode, sizeof(ULONG), lpOutBuffer,
nOutBufferSize, lpBytesReturned, lpOverlapped);
```

# Serial Port Information

This appendix describes the RS-232, RS-422, and RS-485 standards and explains some of the issues involved with these types of serial communication.

Table A-1 lists the features of the RS-232, RS-422, and RS-485 standards.

**Table A-1.** RS-232, RS-422, and RS-485 Features

Feature	RS-232	RS-422	RS-485
Type of transmission lines	Single ended	Differential	Differential
Maximum number of drivers	1	1	32
Maximum number of receivers	1	10	32
Maximum cable length	50 ft	4,000 ft	4,000 ft
Maximum data rate	20 kbytes/s	10 Mbytes/s	10 Mbytes/s
Maximum CMV	$\pm 25$ V	$\pm 7$ V	+12 to $-7$ V
Driver output	5 to 25 V	2 to 6 V	1.5 to 6 V
Driver load	$> 3$ k $\Omega$	100 $\Omega$	60 $\Omega$

## RS-232

As specified in the ANSI/EIA-232-D Standard, *Interface Between Data Terminal Equipment and Data Circuit-Terminating Equipment Employing Serial Binary Data Interchange*, RS-232 standardizes serial communication between computers, and between computer terminals and modems. Most applications use the RS-232 standard to interface peripherals to personal computers. RS-232 uses transmission lines in which the state of each signal is represented by referencing the voltage level of a single line to ground. RS-232 was designed for serial communication up to



distances of 50 ft and with data rates up to 20 kbytes/s. However, because of improvements in line drivers and cabling, you can usually increase the actual performance of the bus past the limitations on speed and distance recommended in the specification.

## RS-422

---

As specified in the EIA/RS-422-A Standard, *Electrical Characteristics of Balanced Voltage Digital Interface Circuits*, RS-422 defines a serial interface much like RS-232. However, RS-422 uses balanced (or differential) transmission lines. Balanced transmission lines use two transmission lines for each signal. The state of each signal is represented, not by a voltage level on one line referenced to ground as in RS-232, but rather by the relative voltage of the two lines to each other. For example, the TX signal is carried on two wires, wire A and wire B. A logical 1 is represented by the voltage on line A being greater than the voltage on line B. A logical 0 is represented by the voltage on line A being less than the voltage on line B. Differential voltage transmission creates a signal that is more immune to noise as well as voltage loss due to transmission line effects. Thus, you can use RS-422 for longer distances (up to 4,000 ft) and greater transmission speeds (up to 10 Mbytes/s) than RS-232.

## RS-485

---

As specified in the EIA-485 Standard, *Standard for Electrical Characteristics of Generators and Receivers for Use in Balanced Digital Multipoint Systems*, RS-485 expands on the RS-422 standard by increasing the number of devices you can use from 10 to 32 and by working with half-duplex bus architectures. Unlike the RS-422 standard, RS-485 addresses the issue of using multiple transmitters on the same line. RS-485 defines the electrical characteristics necessary to ensure adequate signal voltages under maximum load, short-circuit protection. RS-485 can also withstand multiple drivers driving conflicting signals at the same time.

# Serial Communication Issues

---

This section explains some serial communication issues, including duplex architectures, termination methods, bias resistors, and types of connecting equipment.

## Duplex Architectures

*Duplex* refers to the means of bandwidth usage in a serial system. The two common means of bi-directional serial communication are full duplex and half duplex. Half-duplex communication involves a transmitter and a receiver connected to each end of the same wire or pair of wires. Because the same transmission line both sends and receives data, devices cannot send data in both directions at the same time. First, one device transmits over the wire(s) to the receiver of the second device. When the first device finishes transmitting, both devices switch the connections from their transmitter to their receiver, or vice versa. The device that was receiving data can then transmit over the line.

In full-duplex communication, the devices use a separate wire (or pair of wires) for simultaneous transmission in each direction. Thus, the devices do not switch between transmitting and receiving.

In a differential serial bus (such as RS-422 or RS-485), a half-duplex system transmits and receives over the same twisted pair of wires. Thus, half-duplex communication is often referred to as *two-wire* communications. Likewise, full-duplex communication is often referred to as *four-wire* communications, because the full-duplex system uses a separate pair of wires for communication in each direction.

## Full Duplex

A typical full-duplex multidrop bus architecture involves a master-slave protocol. Only one device, the master, can control access to the bus. All other devices are slaves. Slave devices must wait for the master to give them access to the bus. In a typical full-duplex system, one transmission line connects the bus master transmitter to all of the slave receivers. A second transmission line connects all of the slave transmitters to the bus master receiver. Because each transmission line has two separate wires, a full-duplex system is often referred to as a four-wire system. Figure A-1 shows a typical full-duplex system.

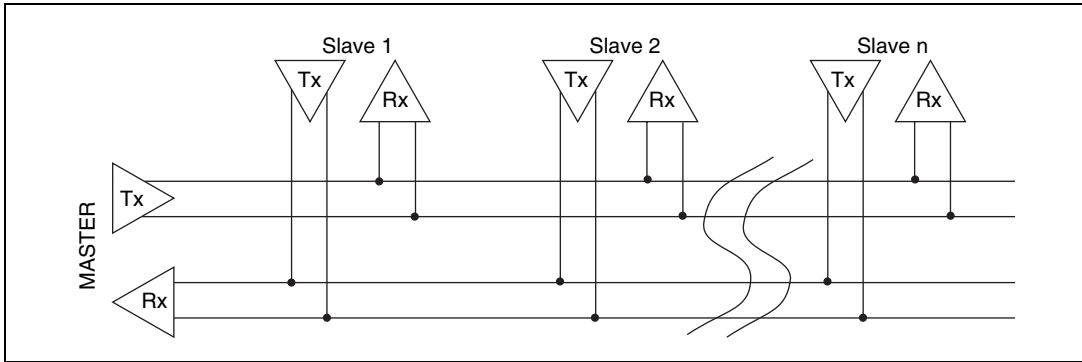


Figure A-1. Typical Full-Duplex System

## Half Duplex

A typical half-duplex multidrop bus architecture also involves a master-slave protocol. However, in a half-duplex system, all transmitters and receivers are connected to the same transmission line. A half-duplex system is often referred to as a two-wire system. Figure A-2 shows a typical half-duplex system.

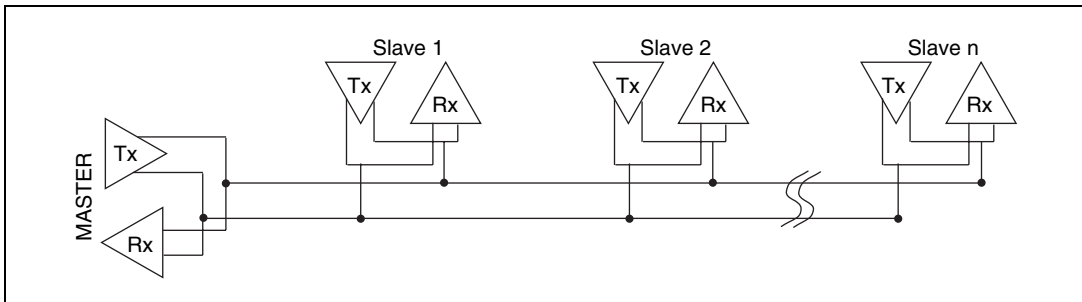
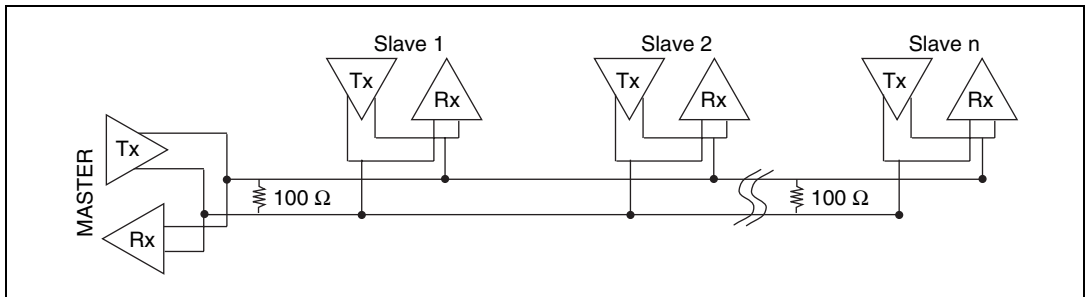


Figure A-2. Typical Half-Duplex System

## Termination

Because each differential pair of wires is a transmission line, you must properly terminate the line to prevent reflections. A common method of terminating a two-wire multidrop RS-485 network is to install terminating resistors at each end of the multidrop network. If you daisy-chained multiple instruments together, you need a terminating resistor at only the first and last instruments. The terminating resistor should match the characteristic impedance of the transmission line (typically 100 to 120  $\Omega$ ). You can order an optional DB-9 RS-485 termination connector that contains embedded terminating resistors for easy termination from National Instruments. For ordering information, contact National Instruments.

Figure A-3 shows a multidrop network using terminating resistors.



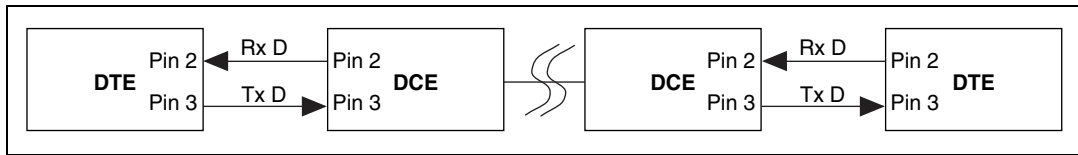
**Figure A-3.** Multidrop Network Using Terminating Resistors

## DTE vs. DCE

In the RS-232 specification, DTE (Data Terminal Equipment) and DCE (Data Communications Equipment)<sup>1</sup> refer to the types of equipment on either end of a serial connection. In general, DTE and DCE refer to computer equipment and modems, respectively. Because the RS-232 specification mainly involves connecting a DTE directly to a DCE and vice versa, the pinouts are defined so that cabling is simple. That is, a cable connected a computer to a modem by wiring pin 1 to pin 1, pin 2 to pin 2, and so on. This method is known as *straight-through* cabling.

Figure A-4 shows straight-through cabling in a DTE-to-DCE interface.

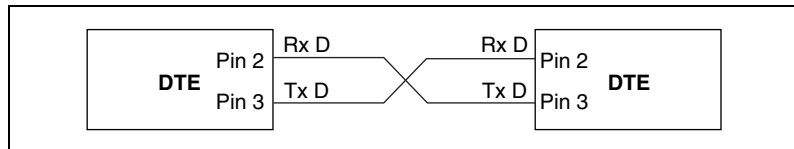
<sup>1</sup> In Revision D of the RS-232 specification, a DCE is a Data Circuit-Terminating Equipment.



**Figure A-4.** Straight-Through Cabling in a DTE-to-DCE Interface

Straight-through cabling is still the standard method to connect a modem to your PC. However, because many applications use serial communication to connect two or more DTEs without modems, the cabling becomes more complicated. If two DTEs are wired together using a straight-through cable, one transmitter is connected to the other transmitter, and one receiver is connected to the other receiver. In this setup, no transmissions can occur. Thus, these applications must use a cabling scheme that connects the transmitter on one device to the receiver on the other device and vice versa. This method is known as *null-modem* cabling, because it replaces the two modems that traditional RS-232 applications would require between the two DTEs. To communicate from one serial port to another, use a null-modem cable.

Figure A-5 shows null-modem cabling in a DTE-to-DTE interface.



**Figure A-5.** Null-Modem Cabling in a DTE-to-DTE Interface

# Uninstalling the Hardware and Software

This appendix describes how to uninstall your serial hardware and the NI-Serial software.

## Uninstall the Hardware

---

Before physically removing the serial hardware, you must remove the hardware information from the Windows Me/9x Device Manager.

Follow these steps to uninstall the hardware:

1. Double-click on the **System** icon under **Start»Settings»Control Panel**. The **System Properties** dialog box appears.
2. Select the **Device Manager** tab.
3. Click on the **View devices by type** button.
4. Double-click on the **Ports (COM & LPT)** icon.

5. Select the National Instruments port to remove from the list of ports, as shown in Figure B-1.

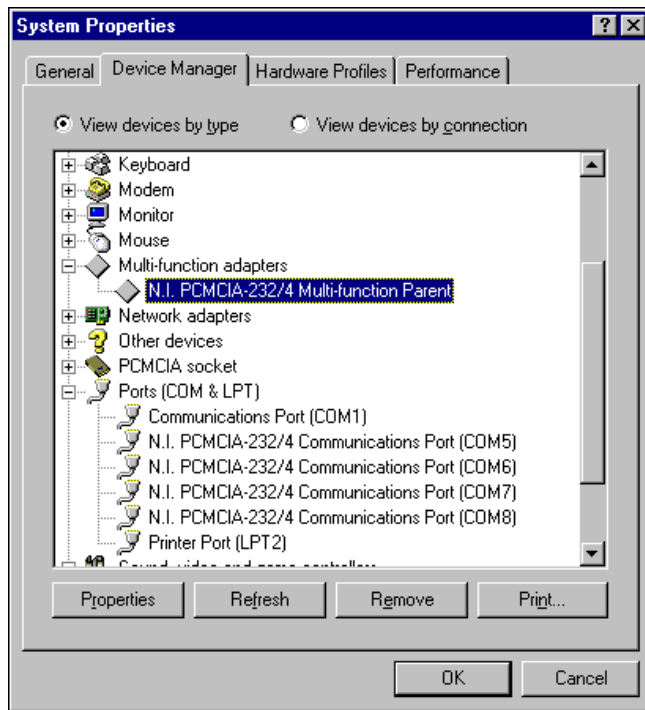


Figure B-1. Selecting an Interface to Uninstall

6. Click on the **Remove** button.



**Note** To remove ports on two-port and four-port PCMCIA cards from within the Device Manager, you must open the **Multi-function adapters** class, select the name of your interface, and click on the **Remove** button.

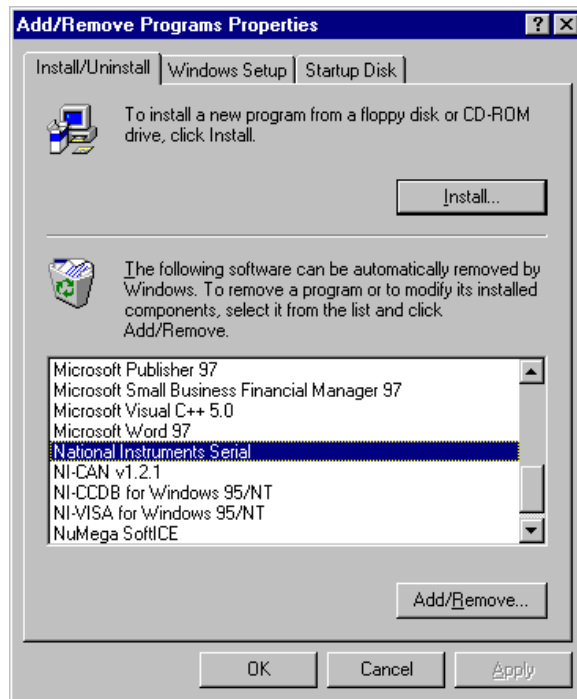
7. In the **Confirm Device Removal** dialog box, click on the **OK** button to remove this interface or click on **Cancel** to cancel your request.

# Uninstall the Software

Before uninstalling the software, remove all serial port information from the Windows Me/9x Device Manager, as described in the previous section. You do not need to shut down Windows Me/9x before uninstalling the software.

Complete the following steps to remove the NI-Serial software:

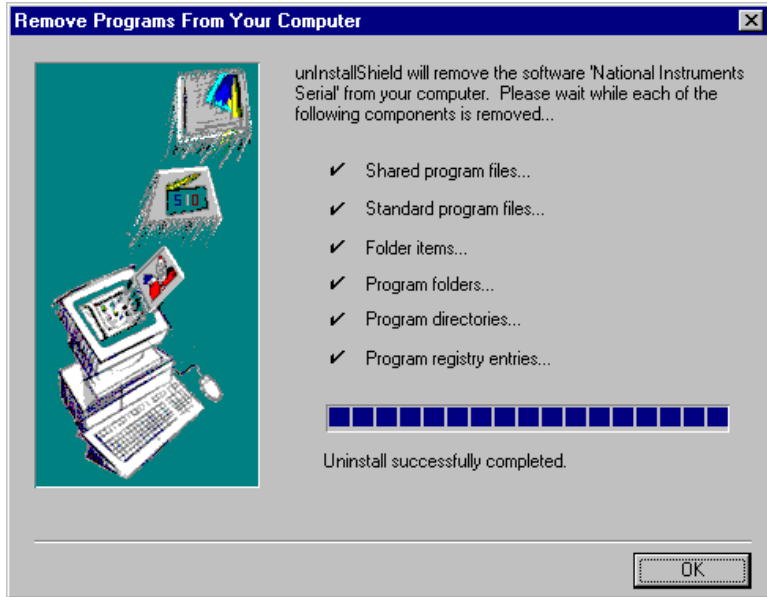
1. Select **Start»Settings»Control Panel**.
2. Double-click on the **Add/Remove Programs** icon in the Control Panel to launch the Add/Remove Programs applet. A dialog box similar to the one in Figure B-2 appears. This dialog box lists the software available for removal.



**Figure B-2.** Add/Remove Programs Properties Dialog Box



3. Select the serial software you want to remove, and click on the **Add/Remove** button. The uninstall program runs and removes all folders, programs, VxDs, DLLs, and registry entries associated with the NI-Serial software. Figure B-3 shows the results of a successful uninstallation.
4. Remove your PCMCIA card from your computer.



**Figure B-3.** Successful Uninstallation

If you want to reinstall the hardware and software, refer to Chapter 2, *Installation and Verification*.



---

# Troubleshooting and Common Questions

This appendix describes how to troubleshoot problems and answers some common questions.

## Freeing an Interrupt Request Level

---

To free an interrupt request level for your serial port, you must disable the device that is using the interrupt request level. To view the system-wide allocation of interrupt request resources and remove a device, complete the following steps:

1. Select **Start»Settings»Control Panel** and double-click on the **System** icon.
2. Click on the **Device Manager** tab.
3. Double-click on the **Computer** icon and click on the **View Resources** tab.
4. Click on the **Interrupt Request (IRQ)** button. Scan through the list of IRQ settings to determine which devices are using which interrupt request levels.
5. When you locate a device that you are not currently using, click on the **Cancel** button.
6. In the **Device Manager** tab, double-click on the icon for that device.
7. In the **Device usage** field, a checkmark appears to the left of the current configuration, usually **Original Configuration (Current)**. Click on the checkbox to remove the checkmark.
8. Click on the **Resources** tab and make sure that **Use automatic settings** is unchecked.
9. Repeat steps 7 through 9 for each unused device.
10. Click on the **OK** button to save your changes.
11. Restart Windows Me/9x to assign resources to your serial port. Then, complete step 6 in the *Verify the Installation* section of Chapter 2, *Installation and Verification*.

## Selecting Conflict-Free Resources

---

When the Device Manager indicates a conflict with another device, you can usually correct the problem by manually selecting conflict-free resources. To do so, complete the following steps:

1. Select **Start»Settings»Control Panel** and double-click on the **System** icon.
2. Click on the **Device Manager** tab and click on the **View devices by type** button.
3. Double-click on the **Multi-function adapters** icon.
4. From the **Multi-function adapters** list, double-click on the serial interface that you want to change.
5. Click on the **Resources** tab.



**Note** To change the resources of ports on two-port or four-port PCMCIA cards from within the Device Manager, you must open the **Multi-function adapters** class and double-click on the name of your interface. Use the **Resources** page to change the resources. Any resource changes are automatically reflected to the child devices.

6. Uncheck the **Use automatic settings** checkbox and click on the **Change Setting** button. If the system does not allow you to change the settings, select the configuration that gives you a conflict-free base address and interrupt level from the **Setting based on** list box.
7. Click on the **OK** button to save your changes.

## Troubleshooting Diagnostic Messages

---

This section lists possible Diagnostic error messages, along with solutions.

- If the **No National Instruments serial port found** error message appears, refer to Chapter 2, *Installation and Verification*, and follow these troubleshooting steps:
  1. Verify the hardware resources.
  2. In some versions of Windows Me/9x, the **Driver** tab in the Device Manager gives you the information about the driver installed. Verify that the National Instruments serial driver is installed for the port.
  3. If either the serial hardware or `niserial.vxd` file is missing, reinstall the hardware and software. You can find `niserial.vxd` in the `\Windows\system` directory.

- If the utility reports fewer ports installed than it should, refer to Chapter 2, *Installation and Verification*, and follow these troubleshooting steps:
  1. Verify the hardware resources.
  2. In some versions of Windows Me/9x, the **Driver** tab in the Device Manager gives you the information about the driver installed. Verify that the National Instruments serial driver is installed for the port.
  3. Check the hardware installation to make sure the correct number of boards/ports are installed.
- If the **I/O address test failed, Interrupt test cannot be performed** error message appears, verify the hardware resources as described in Chapter 2, *Installation and Verification*. If the test still fails, you probably have an I/O address conflict with legacy boards. Refer to the next section, *Resolving Resource Conflicts with Legacy Boards*.
- If the **Interrupt test failed** error message appears, verify the hardware resources as described in Chapter 2, *Installation and Verification*. If the test still fails, you probably have an interrupt level conflict with legacy boards. Refer to the next section, *Resolving Resource Conflicts with Legacy Boards*.

## Resolving Resource Conflicts with Legacy Boards

---

Resource conflicts typically occur if your system contains legacy boards that use resources that are not reserved properly. If a resource conflict exists, write down the resource that caused the conflict and refer to the Microsoft Windows Me/9x user's guide for instructions on how to use the Device Manager to reserve I/O and IRQ resources for legacy boards.

## Forcing Windows to Detect Your Hardware

---

If Windows does not detect your serial hardware correctly, or you have been using your serial hardware with the default Windows serial driver or an older version of the NI-Serial software, Windows does not display the **New Hardware Found** dialog box. To fix this problem, follow these steps:

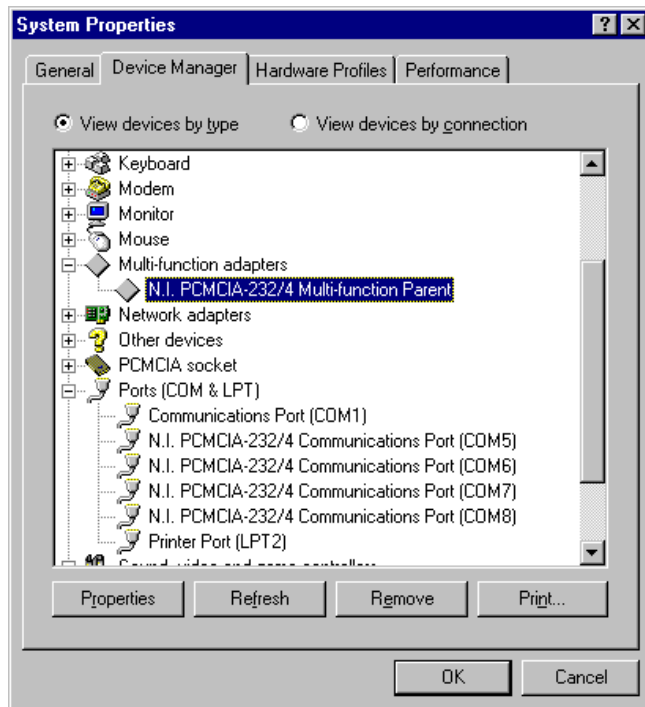
1. Select **Start»Settings»Control Panel**.
2. Double-click on the **System** icon.
3. Select the **Device Manager** tab.

4. Click on the **View devices by type** button.
5. Double-click on the **Ports (COM & LPT)** icon to display a list of all ports that Windows recognizes. If you have previously installed a PCMCIA serial card, double-click on **Other Devices**, where Windows lists the ports it does not recognize.



**Note** To remove ports on two- or four-port cards, you must open the **Multifunction adapter** class, select the name of the interface, and click on the **Remove** button.

6. Select a National Instruments port name from the list of ports that corresponds to a serial port (for example, COM 5, COM 6, COM 7, or COM 8). Figure C-1 shows the **Ports** list in the **Device Manager** with a port selected.



**Figure C-1.** Ports List in Device Manager Tab

7. Click on the **Remove** button, and click on **OK** to confirm the removal of the serial hardware information.
8. Repeat steps 6 and 7 until all serial port entries are removed. Then, click on **Refresh** to force Windows Me/9x to detect serial hardware and display the **New Hardware Found** dialog box.

# Common Questions

---

## How can I determine which type of serial hardware I have installed?

1. Select **Start»Settings»Control Panel** and double-click on the **System** icon.
2. Click on the **Device Manager** tab and click on the **View devices by type** button.
3. Double-click on the **Ports (COM & LPT)** icon to display a list of installed ports.

## How can I determine which version of the NI-Serial software I have installed?

The **Read Me** file gives the version of the software. To open the file, select **Start»Programs»National Instruments Serial»Read Me**.

## What do I do if the diagnostic test fails with an error?

Refer to the [Troubleshooting Diagnostic Messages](#) section earlier in this appendix.

## How can I determine which port is associated with COMx?

Refer to the section [Determine Which Physical Port Is Associated with COMx](#), in Chapter 2, *Installation and Verification*.

## How can I change the COM number of my National Instruments serial ports?

Table C-1 lists standard DOS-base addresses for serial ports.

**Table C-1.** Standard DOS-Based Addresses

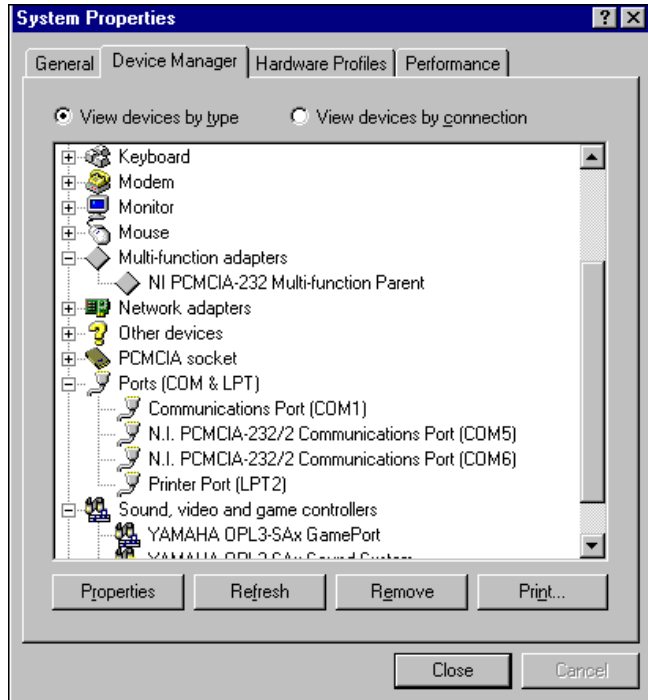
COM Port	Base Address
COM1	3f8
COM2	2f8
COM3	3e8
COM4	2e8

In most cases, Windows Me/9x does not assign names COM1 through COM4 to the serial hardware. Rather, it names the ports starting with COM5. If you assign any of the base addresses in Table D-1 to a National Instruments serial port, Windows Me/9x automatically changes the COM port name to the corresponding one listed in this table. You do not need to change the IRQ setting for this name change to occur. To change the base address, refer to the *Selecting Conflict-Free Resources* section earlier in this appendix.

### **How do I change the resources assigned to PCMCIA two-port or four-port serial cards?**

Because the two-port PCMCIA cards are single physical devices that support multiple ports, they are registered as “multi-function adapters” in the Device Manager. Each port is a child device listed under **Ports (COM & LPT)**. Windows Me/9x does not allow the resources of the multi-function child devices to be changed directly.

To change the resources of ports on two-port PCMCIA cards from within the Device Manager, you must open the **Multi-function adapters** class and double-click on the **Multi-function Parent** item that corresponds to your National Instruments serial interface. Use the **Resources** page to change the resources. Any resource changes are automatically reflected to the child devices. Figure C-2 shows the **Multi-function adapters** class in the Device Manager.



**Figure C-2.** Multi-Function Adapter Class in the Device Manager

### How do I remove information about PCMCIA two-port or four-port serial cards from the Device Manager?

Because the two-port PCMCIA cards are single physical devices that support multiple ports, they are registered as “multi-function adapters” in the Device Manager. Each port is a child device listed under **Ports (COM & LPT)**. Windows Me/9x does not allow multi-function child devices to be removed directly within the Device Manager. You must open the **Multi-function adapters** class as shown in Figure C-2, select your interface, and click on the **Remove** button.

### What is the maximum baud rate supported and how can I set it?

The maximum baud rate supported is 460.8 Kbaud for RS-485 and 115.2 Kbaud for RS-232. To set the baud rate, use the **SetCommState** Win32 function and pass the actual value of the baud rate in the **BaudRate** field of the **DCB** structure.



# Specifications

This appendix describes the characteristics of the serial hardware, the NI-Serial software, along with the recommended operating conditions.

## Hardware Specifications

**Table D-1.** Physical Characteristics of the PCMCIA Serial Cards

Characteristic	Specification
Dimensions	85.6 by 54.0 by 0.5 cm (3.370 by 2.126 by 0.197 in.)
I/O Connector	Special Cable with DB-9 Dsub Connector and Converter for PC Card
Power Requirement (from PCMCIA expansion slot)	
PCMCIA-232	+5 VDC 40 mA Typical
PCMCIA-485	+5 VDC 100 mA Typical
PCMCIA-232/2	+5 VDC 60 mA Typical
PCMCIA-485/2	+5 VDC 100 mA Typical
PCMCIA-232/4	+5 VDC 60 mA Typical

**Table D-2.** Environmental Characteristics of the Serial Hardware

Characteristic	Specification
Operating Environment Component Temperature Relative Humidity	0 to 70° C 10 to 90%, noncondensing
Storage Environment Temperature Relative Humidity	-40 to 120° C 5 to 90%, noncondensing
EMI	FCC Class A Verified

## Software Specifications

---

**Table D-3.** Software Characteristics

Characteristic	Specification
Maximum Serial Transfer Rate	115,200 bps*
* Actual speed may vary considerably from speed shown due to system and instrumentation capabilities.	

---

# Technical Support Resources

## Web Support

---

National Instruments Web support is your first stop for help in solving installation, configuration, and application problems and questions. Online problem-solving and diagnostic resources include frequently asked questions, knowledge bases, product-specific troubleshooting wizards, manuals, drivers, software updates, and more. Web support is available through the Technical Support section of [ni.com](http://ni.com)

## NI Developer Zone

---

The NI Developer Zone at [ni.com/zone](http://ni.com/zone) is the essential resource for building measurement and automation systems. At the NI Developer Zone, you can easily access the latest example programs, system configurators, tutorials, technical news, as well as a community of developers ready to share their own techniques.

## Customer Education

---

National Instruments provides a number of alternatives to satisfy your training needs, from self-paced tutorials, videos, and interactive CDs to instructor-led hands-on courses at locations around the world. Visit the Customer Education section of [ni.com](http://ni.com) for online course schedules, syllabi, training centers, and class registration.

## System Integration

---

If you have time constraints, limited in-house technical resources, or other dilemmas, you may prefer to employ consulting or system integration services. You can rely on the expertise available through our worldwide network of Alliance Program members. To find out more about our Alliance system integration solutions, visit the System Integration section of [ni.com](http://ni.com)

## Worldwide Support

---

National Instruments has offices located around the world to help address your support needs. You can access our branch office Web sites from the Worldwide Offices section of [ni.com](http://ni.com). Branch office Web sites provide up-to-date contact information, support phone numbers, e-mail addresses, and current events.

If you have searched the technical support resources on our Web site and still cannot find the answers you need, contact your local office or National Instruments corporate. Phone numbers for our worldwide offices are listed at the front of this manual.

# Glossary

---

Prefix	Meaning	Value
p-	pico-	$10^{-12}$
n-	nano-	$10^{-9}$
$\mu$ -	micro-	$10^{-6}$
m-	milli-	$10^{-3}$
k-	kilo-	$10^3$
M-	mega-	$10^6$
G-	giga-	$10^9$
t-	tera-	$10^{12}$

° degrees

$\Omega$  ohms

% percent

A amperes

ANSI American National Standards Institute

b bits

B bytes

baud bits per second

bps bits per second

C Celsius

COM Computer Output Microform; used in reference to a communication port

CTS clear to send

DB-xx subminiature D connector, where xx is the number of pins

DCD	data carrier detect
DCE	data communications equipment or data circuit-terminating equipment
DLL	dynamic link library
DMA	direct memory access
DSR	data set ready
DTE	data terminal equipment
$\overline{\text{DTR}}$	data terminal ready—the overscore denotes that the signal is active low
duplex	the means of bandwidth usage in a serial system
EIA	Electronic Industries Association
EMI	electromagnetic interference
FCC	Federal Communications Commission
FIFO	First-In-First-Out
ft	feet
GND	ground
HSI	handshake input
HSO	handshake output
Hz	Hertz
IEEE	Institute of Electrical and Electronic Engineers
in.	inches
I/O	input/output
IRQ	interrupt request
ISA	Industry Standard Architecture
Kbaud	kilobits per second
m	meters

MB	megabytes of memory
PC	personal computer
PCI	Peripheral Components Interconnect
RAM	random-access memory
RI	ring indicator
RTS	request to send
RX	receive
$\overline{\text{RXD}}$	receive data—the overscore denotes that the signal is active low
s	seconds
SCSI	Small Computer Systems Interface
TX	transmit
$\overline{\text{TXD}}$	transmit data—the overscore denotes that the signal is active low
$\overline{\text{TXRDY}}$	transmit ready—the overscore denotes that the signal is active low
UART	Universal Asynchronous Receiver/Transmitter
V	volts
VDC	volts direct current
VXI	VME eXtensions for Instrumentation
Win16	describes a 16-bit Windows application
Win32	describes a 32-bit Windows application

# Index

---

## A

- add/remove programs properties dialog box (figure), 2-1, B-3
- advanced port settings, 3-3
  - dialog box (figure), 3-3
  - transceiver mode, 3-4
  - using FIFO buffers, 3-4
- advanced transceiver control for the PCMCIA-485, 4-1
  - four-wire mode, 4-2
  - setting transceiver control mode, 4-3
  - transceiver mode control modes (table), 4-2
  - two-wire mode:  $\overline{\text{DTR}}$  controlled, 4-2
  - two-wire mode:  $\overline{\text{DTR}}$  with Echo, 4-2
  - two-wire mode:  $\overline{\text{TXRDY}}$  auto control, 4-3

## B

- bits per second, 3-3

## C

- cable connection, 2-8
- changing communication port settings, 3-1
- changing resources assigned to PCMCIA serial cards, C-6
- common questions, C-5
  - changing resources assigned to PCMCIA serial cards, C-6
  - determining port associated with COM $x$ , C-5
  - determining type of serial hardware installed, C-5
  - determining version of NI-Serial software installed, C-5
  - diagnostic test failure, C-5
  - maximum supported baud rate, C-7

- naming National Instruments serial ports, C-5
- removing information about PCMCIA serial cards from the device manager, C-7
- communication port settings, 3-1
  - viewing or changing, 3-1
- configuration, 3-1
- conflict-free resources, C-2
  - selecting, C-2
- connecting cables, 2-8
- connecting two-wire devices, 2-9
- conventions, *xi*
- customer education, E-1

## D

- data bits, 3-3
- DB-9
  - pin descriptions (table), 2-8
  - pin locations (figure), 2-8
- DCE vs. DTE, A-5
- default restoration, 3-4
- determining port associated with COM $x$ , 2-7, C-5
- determining type of serial hardware installed, C-5
- determining version of NI-Serial software installed, C-5
- development tools for saving time, 1-4
- device manager tab for PCMCIA serial card ports incorrectly installed (figure), 2-6
- device manager tab ports list (figure), 2-5, C-4
- DeviceloControl, 4-4
  - input values (table), 4-4
  - setting transceiver mode with, 4-4
- diagnostic messages, C-2
  - troubleshooting, C-2
- diagnostic test failure, C-5



DOS-based addresses (table), C-5  
DTE vs. DCE, A-5  
DTE-to-DCE interface, A-6  
    straight-through cabling in (figure), A-6  
DTE-to-DTE interface, A-6  
    null-modem cabling in (figure), A-6  
duplex architectures, A-3  
    full duplex, A-3  
    half duplex, A-4

## E

environmental characteristics, D-2

## F

FIFO buffers, 3-4  
flow control, 3-3  
forcing Windows to detect hardware, C-3  
four-wire mode, 4-2  
freeing an interrupt request level, C-1  
full duplex, A-3  
    full-duplex system (figure), A-4

## G

general programming requirements, 4-1  
getting started, 1-2  
glossary, G-1

## H

half duplex, A-4  
    half-duplex system (figure), A-4  
hardware installation, 2-3  
    getting started, 1-2  
    verification, 2-4  
hardware resources, verification, 2-6

## I

installation  
    determining port associated with  
        COMx, 2-7  
    getting started, 1-2  
    running the diagnostic test, 2-7  
    serial hardware, 2-3  
    serial software, 2-1  
    verification, 2-1  
    verifying hardware installation, 2-4  
    verifying hardware resources, 2-6  
interrupt request level, C-1  
    freeing, C-1  
introduction, 1-1

## L

LabVIEW, 1-4  
    using with serial hardware, 1-4  
LabWindows/CVI, 1-4  
    using with serial hardware, 1-4  
legacy boards, C-3  
    resolving resource conflicts with, C-3

## M

maximum supported baud rate, C-7  
multidrop network using terminating resistors  
    (figure), A-5  
multi-function adapter class in the Device  
    Manager (figure), C-7

## N

naming National Instruments serial ports, C-5  
    standard DOS-based addresses  
        (table), C-5  
National Instruments Web support, E-1  
NI Developer Zone, E-1

NI-Serial setup screen (figure), 2-2  
 null-modem cabling in a DTE-to-DTE  
 interface (figure), A-6

## O

optional equipment, 1-2  
 overview of serial hardware, 1-2  
 overview of serial software, 1-3

## P

parity, 3-3  
 PCMCIA-485
 

- advanced transceiver control for, 4-1
  - setting transceiver control mode, 4-3
- transceiver mode control modes  
 (table), 4-2
- two-wire mode:  $\overline{\text{DTR}}$  controlled, 4-2
- two-wire mode:  $\overline{\text{DTR}}$  with Echo, 4-2
- two-wire mode:  $\overline{\text{TXRDY}}$  auto  
 control, 4-3

physical characteristics, D-1  
 pin descriptions
 

- DB-9
  - connector pin descriptions  
 (table), 2-8
  - connector pin locations  
 (figure), 2-8

port settings tab, 3-2
 

- figure, 3-2
- options
  - advanced port settings, 3-3
  - bits per second, 3-3
  - data bits, 3-3
  - flow control, 3-3
  - parity, 3-3
  - stop bits, 3-3
- restore defaults button, 3-4

ports list in device manager tab (figure), C-4

## R

related documentation, *xii*  
 removing information about PCMCIA serial  
 cards from the device manager, C-7  
 resolving resource conflicts with legacy  
 boards, C-3  
 restoring defaults, 3-4  
 RS-232 standard, A-1
 

- features (table), A-1

 RS-422 standard, A-2
 

- features (table), A-1

 RS-485 standard, A-2
 

- features (table), A-1

 running the diagnostic test, 2-7

## S

selecting an interface to uninstall (figure), B-2  
 selecting conflict-free resources, C-2  
 serial communication issues, A-3
 

- DTE vs. DCE, A-5
- duplex architectures, A-3
- termination, A-5

serial hardware
 

- forcing Windows to detect, C-3
- general programming requirements, 4-1
- installation, 2-3
- overview, 1-2
- selecting an interface to uninstall  
 (figure), B-2
- uninstalling, B-1
- using, 4-1
- using with LabVIEW and  
 LabWindows/CVI, 1-4

serial port information, A-1  
 serial software
 

- add/remove programs properties dialog  
 box (figure), B-3
- installation, 2-1
- overview, 1-3

- successful uninstallation (figure), B-4
- uninstalling, B-3
- setting transceiver control mode, 4-3
  - setting with DeviceloControl, 4-4
- setup screen for NI-Serial software (figure), 2-2
- software installation, 2-1
  - getting started, 1-2
- software specifications, D-2
- specifications, D-1
  - environmental characteristics, D-2
  - physical characteristics, D-1
  - software specifications, D-2
- standard DOS-based addresses (table), C-5
- stop bits, 3-3
- straight-through cabling in a DTE-to-DCE interface (figure), A-6
- successful uninstallation (figure), B-4
- system integration, by National Instruments, E-1

## T

- technical support resources, E-1
- termination, A-5
  - multidrop network using terminating resistors (figure), A-5
- time-saving development tools, 1-4
- transceiver control modes (table), 4-2
- transceiver mode, 3-4
  - setting with DeviceloControl, 4-4
- transceiver mode control bytes (figure), 4-3
- troubleshooting, C-1
  - diagnostic messages, C-2
  - forcing Windows to detect your hardware, C-3
  - freeing an interrupt request level, C-1
  - resolving resource conflicts with legacy boards, C-3
  - selecting conflict-free resources, C-2

- two-wire devices, 2-9
  - connecting, 2-9
- typical full-duplex system (figure), A-4
- typical half-duplex system (figure), A-4

## U

- uninstalling serial hardware, B-1
- uninstalling serial software, B-3
- using FIFO buffers, 3-4
- using serial hardware, 4-1
- using this manual, 1-1

## V

- verifying hardware installation, 2-4
- verifying hardware resources, 2-6
- viewing or changing communication port settings, 3-1

## W

- Web support from National Instruments, E-1
- Win32 Overviews and Win32 Reference, 4-1
- Win32 Software Development Kit, 4-1
- worldwide technical support, E-2

## Free Manuals Download Website

<http://myh66.com>

<http://usermanuals.us>

<http://www.somanuals.com>

<http://www.4manuals.cc>

<http://www.manual-lib.com>

<http://www.404manual.com>

<http://www.luxmanual.com>

<http://aubethermostatmanual.com>

Golf course search by state

<http://golfingnear.com>

Email search by domain

<http://emailbydomain.com>

Auto manuals search

<http://auto.somanuals.com>

TV manuals search

<http://tv.somanuals.com>