

**Agilent Technologies**  
**USB/LAN/GPIB Interfaces**

**Connectivity Guide**



**Agilent Technologies**

## Notices

© Agilent Technologies, Inc. 2003-2006

No part of this manual may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Agilent Technologies, Inc. as governed by United States and international copyright laws.

### Manual Part Number

E2094-90009

### Edition

Fourth edition, May 2006

Agilent Technologies, Inc.  
815 14th St. SW  
Loveland, CO 80537 USA

### Trademark Information

Visual Studio is a registered trademark of Microsoft Corporation in the United States and other countries.

Windows NT is a U.S. registered trademark of Microsoft Corporation.

Windows and MS Windows are U.S. registered trademarks of Microsoft Corporation.

### Software Revision

This guide is valid for Revisions 14.xx of the Agilent IO Libraries Suite software, where xx refers to minor revisions of the software that do not affect the technical accuracy of this guide.

## Warranty

**The material contained in this document is provided “as is,” and is subject to being changed, without notice, in future editions. Further, to the maximum extent permitted by applicable law, Agilent disclaims all warranties, either express or implied, with regard to this manual and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Agilent shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or of any information contained herein. Should Agilent and the user have a separate written agreement with warranty terms covering the material in this document that conflict with these terms, the warranty terms in the separate agreement shall control.**

## Technology Licenses

The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license.

## Restricted Rights Legend

If software is for use in the performance of a U.S. Government prime contract or subcontract, Software is delivered and licensed as “Commercial computer software” as defined in DFAR 252.227-7014 (June 1995), or as a “commercial item” as defined in FAR 2.101(a) or as “Restricted computer software” as defined in FAR 52.227-19 (June 1987) or any equivalent agency regulation or contract clause. Use, duplication or disclosure of Software is subject to Agilent Technologies’ standard commercial license terms, and non-DOD Departments and Agencies of the U.S. Government will receive no greater than Restricted Rights as defined in FAR 52.227-19(c)(1-2) (June 1987). U.S. Government users will receive no greater than Limited Rights as defined in FAR 52.227-14 (June 1987) or DFAR 252.227-7015 (b)(2) (November 1995), as applicable in any technical data.

## Safety Notices

### CAUTION

A **CAUTION** notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in damage to the product or loss of important data. Do not proceed beyond a **CAUTION** notice until the indicated conditions are fully understood and met.

### WARNING

A **WARNING** notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in personal injury or death. Do not proceed beyond a **WARNING** notice until the indicated conditions are fully understood and met.

# Contents

<b>1</b>	<b>Getting Started.....</b>	<b>9</b>
	Using This Guide .....	10
	Steps to Get Started .....	11
<b>2</b>	<b>Connecting Instruments to LAN .....</b>	<b>13</b>
	LAN Quick Start .....	14
	Step 1: Install I/O Software on Your PC .....	17
	Step 2: Select LAN Network Type .....	18
	Typical Site LAN Networks .....	19
	Typical Private LAN Networks .....	20
	Step 3: Gather Network Information .....	21
	Gather Site LAN Network Information .....	22
	Define Private LAN Network Parameters .....	24
	Step 4: Connect Your Instruments .....	26
	Connect Instruments to Site LAN .....	27
	Connect Devices to Private LAN .....	29
	Step 5: Configure Your Instruments .....	31
	Check Your Web Browser. ....	32
	Set/Change LAN Parameters .....	32
	Step 6: Run Agilent Connection Expert .....	35
	Run Agilent Connection Expert .....	36
	Add a LAN interface .....	36
	Configure a LAN Interface .....	37
	Communicate with Instruments Using Interactive IO .....	47
	Communicate with Instruments Using Telnet .....	51

<b>3</b>	<b>Connecting Instruments to USB</b>	<b>53</b>
	USB Quick Start	54
	Step 1: Install I/O Software on Your PC	58
	Step 2: Connect Instruments to USB	59
	USB Interface Overview	60
	Steps to Connect USB Instruments	60
	Step 3: Identify Instruments Automatically	62
	Step 4: Communicate with Instruments	66
	Steps to Check Identification Parameters	66
	(Optional) Step 5: Program Your Instruments	69
<b>4</b>	<b>Connecting Instruments to GPIB</b>	<b>71</b>
	GPIB Quick Start	72
	Step 1: Install I/O Software on Your PC	76
	Step 2: Install GPIB Cards in Your PC	77
	Steps to Install a GPIB Card in Your PC	78
	Step 3: Connect Instruments to GPIB Card	82
	Steps to Connect Instruments to GPIB Cards	82
	Step 4: Configure GPIB Interface Cards	86
	Steps to Configure GPIB Interface Cards	86
	Step 5: Communicate with Instruments	91
	Communicating Using Interactive IO	92
	Step 6: Program Your Instruments	95
<b>5</b>	<b>Programming Your Instruments</b>	<b>97</b>
	Programming Overview	98

Obtaining Instrument Drivers .....	99
Using Agilent IO Libraries Suite Sample Code .....	100
Using VISA COM in Other Visual Basic Projects ...	101
Addressing Instruments .....	102
Addressing Instruments via LAN .....	103
Addressing Instruments via USB .....	109
Addressing Instruments via GPIB .....	110
Sample Programs .....	114
<b>6 Installing I/O Software .....</b>	<b>123</b>
System Requirements for Agilent IO Libraries Suite	124
Installing the IO Libraries Suite Software .....	125
Installation Troubleshooting .....	132
Selecting Application Software .....	133
Keeping Your Software Up To Date .....	134
Using Agilent 488 in Multi-Vendor Systems .....	135
Using Agilent VISA in Side-by-Side Mode .....	136
<b>7 TCP/IP Network Basics .....</b>	<b>139</b>
LAN Interface Overview .....	140
Typical Network Topologies .....	140
LAN Hardware Architecture .....	142
TCP/IP Protocols .....	145
The TCP/IP Network Model .....	145
The Network Interface Layer .....	147
The Internet Layer .....	148
The Transport Layer .....	150
The Application Layer .....	151
IP Addressing .....	155

IP Address Classes .....	155
Subnets and Subnet Masks .....	157
Local and Remote Networks .....	158
IP Address Configuration Methods .....	160
Configuration Methods Overview .....	160
Dynamic Host Configuration Protocol (DHCP) .....	161
Auto-IP/ZEROCONF .....	161
Duplicate IP Address Detection .....	162
Device Hostname Services .....	164
Device Hostname Services Overview .....	164
Dynamic DNS Naming .....	165
RFC NetBIOS Naming .....	166
Static DNS Naming .....	167
Configuring Your PC for LAN Operation .....	169
Checking PC Settings .....	169
Installing Network Interface Cards .....	170
Installing TCP/IP on Your PC .....	170
Setting PC IP Address .....	172
Setting PC Hostname .....	174
<b>8 Troubleshooting Guidelines .....</b>	<b>177</b>
Troubleshooting LAN Interfaces .....	178
LAN Troubleshooting Overview .....	178
Communication Checks .....	181
Web Browser Checks .....	184
Troubleshooting USB Interfaces .....	188
USB Troubleshooting Overview .....	188
USB Hardware Checks .....	190
USB Software Checks .....	191
Troubleshooting GPIB Interfaces .....	195

	GPIB Troubleshooting Overview .....	195
	GPIB Hardware Checks .....	196
	GPIB Software Checks .....	199
<b>9</b>	<b>Guide Information .....</b>	<b>207</b>
	Guide Contents .....	208
	Related Documentation .....	209
	Accessing an Electronic Copy of This Guide .....	210
	Contacting Agilent .....	211
	<b>Glossary .....</b>	<b>213</b>







# 1 Getting Started

This chapter includes:

- [Using This Guide](#)
- [Steps to Get Started](#)

## NOTE

This guide does not describe LAN networks that include a gateway, such as the Agilent E5810A LAN/GPIB Gateway for Windows. See the applicable gateway documentation for information on gateway systems.

Also, this guide does not provide a detailed description of LAN, USB, or GPIB interfaces or TCP/IP networks. Consult standard reference texts for this information.

If you need to contact Agilent, see [Contacting Agilent](#) for addresses.

---



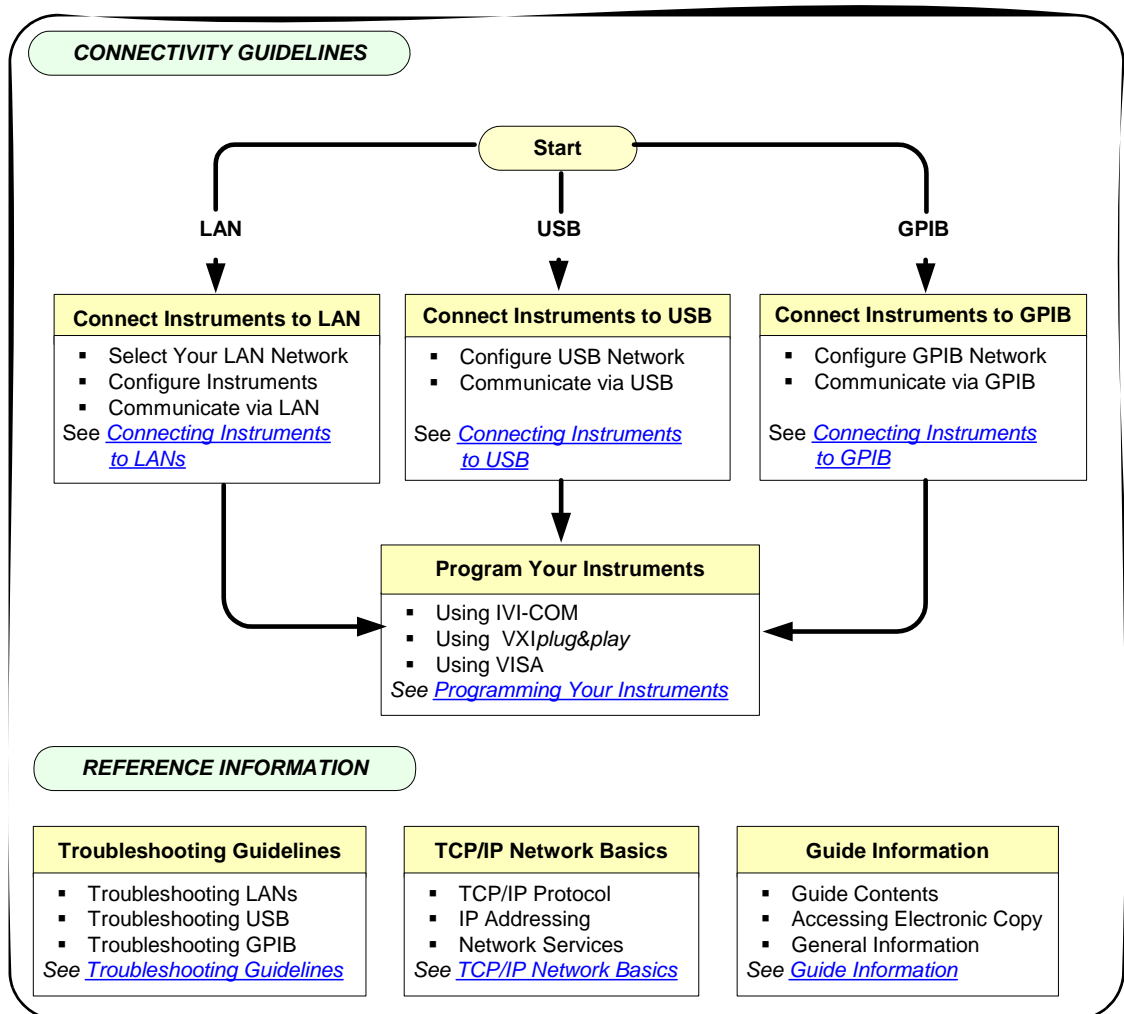
## Using This Guide

This *Agilent Connectivity Guide* shows how to connect instruments to USB, LAN, and GPIB interfaces and how to configure and troubleshoot these interfaces on PCs with Windows 2000 or Windows XP operating systems. A summary of the guide contents follows.

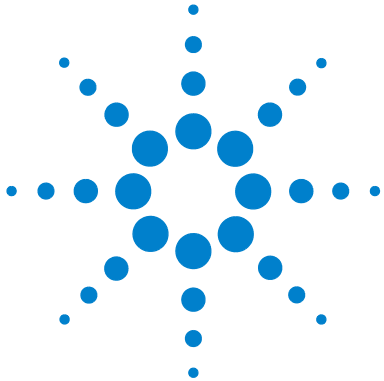
Section	Description
<b>CONNECTIVITY GUIDELINES</b>	
<i>Getting Started</i>	Shows steps to get started and how to contact Agilent
<i>Connecting Instruments to LANs</i>	Gives guidelines to connect instruments to LANs
<i>Connecting Instruments to USB</i>	Gives guidelines to connect instruments to USB
<i>Connecting Instruments to GPIB</i>	Gives guidelines to connect instruments to GPIB
<i>Programming Your Instruments</i>	Gives guidelines to program instruments via LAN/USB/GPIB
<b>REFERENCE INFORMATION</b>	
<i>Installing I/O Software</i>	Shows how to install I/O software on your PC
<i>Troubleshooting Guidelines</i>	Troubleshooting guidelines for LAN, USB, and GPIB Interfaces
<i>TCP/IP Network Basics</i>	Provides an overview of TCP/IP networks
<i>Guide Information</i>	Lists general information for this guide
<i>Glossary</i>	Defines some of the terms used in this guide
<i>Index</i>	Index of selected items in this guide

## Steps to Get Started

This figure shows suggested steps to get started setting up LAN, USB, or GPIB interfaces and the applicable sections of this guide. Some guidelines follow this figure to select the interface you want to set up.



- **Connect Instruments to LAN.** A Local Area Network (LAN) provides instrument connectivity over distances and allows sequential sharing of instruments among multiple PCs. A LAN is the recommended method to connect instruments together in applications such as new test systems. If you select LAN connections, go to [Connecting Instruments to LAN](#).
- **Connect Instruments to USB.** Universal Serial Bus (USB) is a quick and easy way to connect instruments to PCs on a benchtop. Typically, you can use USB to connect a single instrument to a PC. If you select USB connections, go to [Connecting Instruments to USB](#).
- **Connect Instruments to GPIB.** If you select General Purpose Interface Bus (GPIB) connections, go to [Connecting Instruments to GPIB](#).



## 2 Connecting Instruments to LAN

This chapter includes:

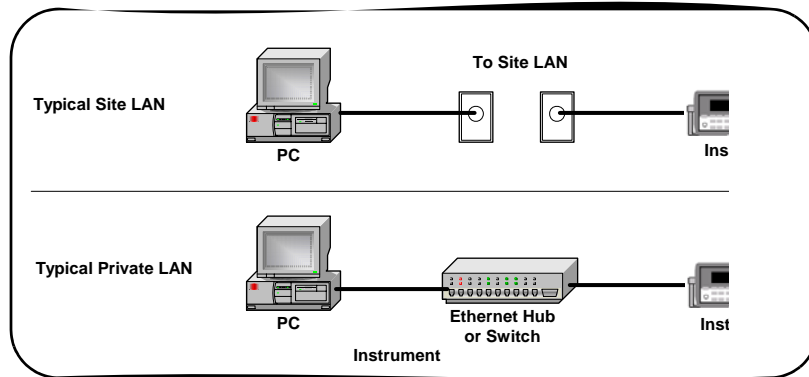
- LAN Quick Start
- Step 1: Install I/O Software on Your PC
- Step 2: Select LAN Network Type
- Step 3: Gather Network Information
- Step 4: Connect Your Instruments
- Step 5: Configure Your Instruments
- Step 6: Run Agilent Connection Expert
  - Add a LAN interface
  - Add and Configure a LAN Instrument
  - Communicate with Instruments Using Interactive IO
  - Communicate with Instruments Using Telnet



## LAN Quick Start

This section shows suggested steps to help you quickly get started connecting and configuring your LAN-enabled instruments for site LAN or private LAN operation.

**What are Site LANs and Private LANs?** A **site LAN network** is defined as a local area network (LAN) in which LAN-enabled instruments and Windows PCs are connected to a site LAN (workgroup LAN, Intranet, or enterprise LAN) via (optional) routers, hubs, and/or switches. A **private LAN network** is defined as a local area network (LAN) in which LAN-enabled instruments and Windows PCs are **NOT** connected to a site LAN.



**Figure 1** Example Site and Private LAN Networks

**Getting Started** To get started, you may want to copy the following overview of steps to use as a guide as you set up your instruments for site LAN or private LAN operation. For additional information:

- [TCP/IP Network Basics](#) for an introduction to TCP/IP networks
- [Troubleshooting LAN Interfaces](#) for LAN troubleshooting tips
- [Glossary](#) for a definition of some LAN terms
- [Contacting Agilent](#) if you need to contact Agilent



### 5 Configure Your Instruments

As required, manually configure your instruments using the instrument's Web pages or front panel.

You can use the **Configuring Your Instrument** Web page to configure instruments.



### 6 Run Agilent Connection Expert

Run Agilent Connection Expert; add and configure the LAN interface (optional); add a LAN instrument; communicate with your instruments using Interactive IO; and program your instruments (optional).

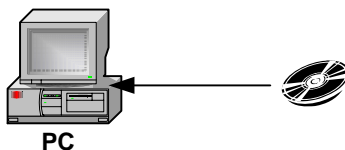




## Step 1: Install I/O Software on Your PC

### 1 Install I/O Software on Your PC

Install Agilent IO Libraries Suite from a CD or the Web.



**What's in This Step?** This step gives guidelines to install applicable I/O software on your PC. Go to [Installing I/O Software](#) for details. After the I/O software is installed, go to [Step 2: Select LAN Network Type](#).

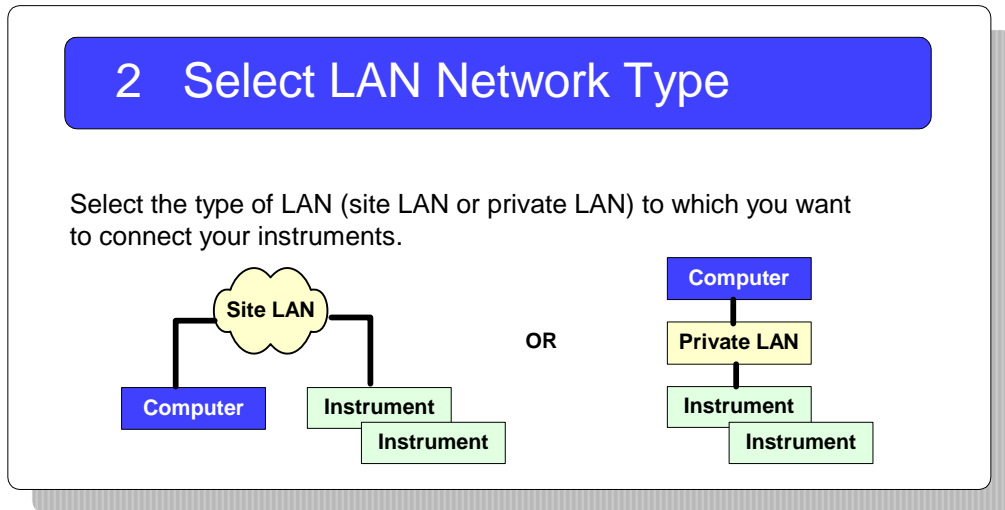
#### NOTE

Agilent IO Libraries Suite supports the following operating systems:

- Microsoft Windows 2000
- Microsoft Windows XP

If you wish to use an earlier version of Windows, you must download a compatible version of the Agilent IO Libraries from <http://www.agilent.com/find/iolib>

## Step 2: Select LAN Network Type

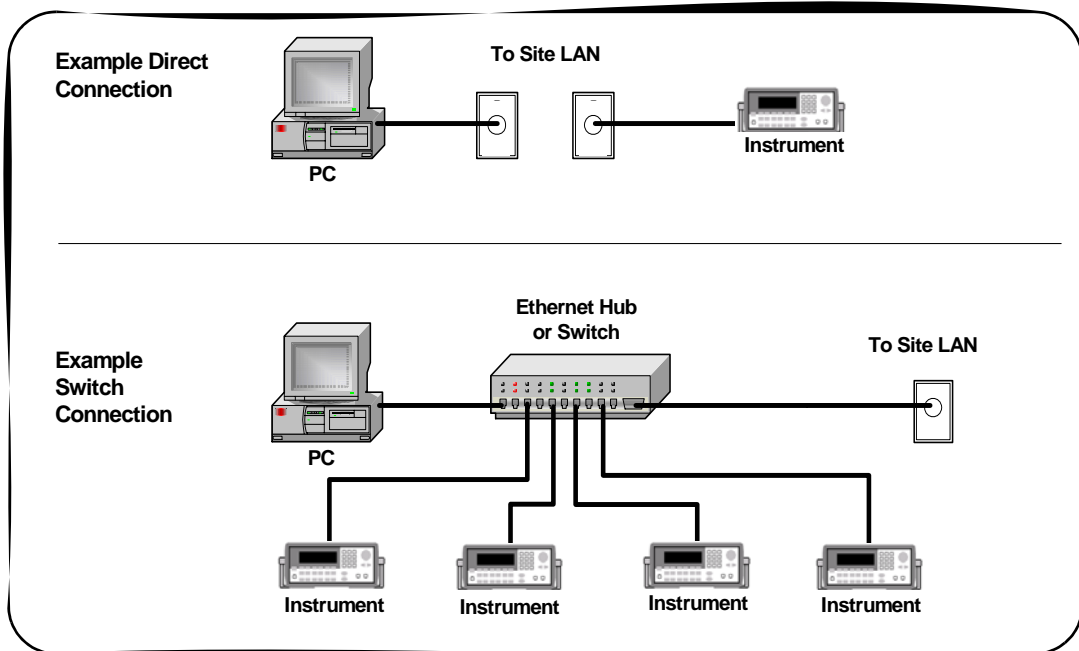


**What's in This Step?** The first step is to determine whether you want your instruments and your PC to be connected to a site LAN or to a private LAN. This step shows some examples of site LAN and private LAN networks. Select site LAN or private LAN connections, based on your requirements. Then, go to [Step 3: Gather Network Information](#).

- For examples of typical site LAN networks, go to [Typical Site LAN Networks](#)
- For examples of typical private LAN networks, go to [Typical Private LAN Networks](#)

## Typical Site LAN Networks

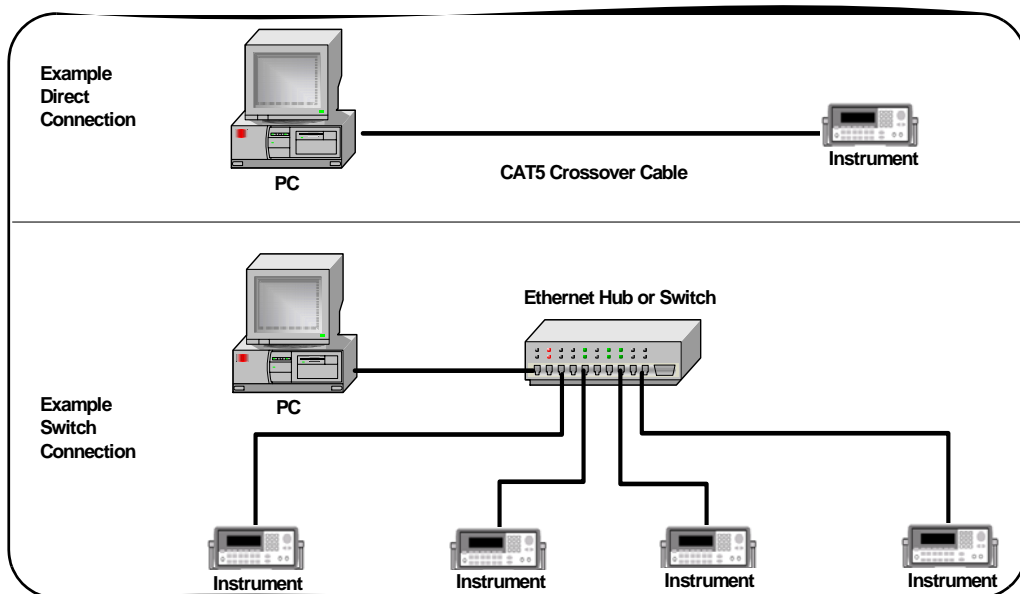
In this guide, a **site LAN network** is defined as a local area network (LAN) in which LAN-enabled instruments and Windows PCs are connected to a site LAN (workgroup LAN, Intranet, or enterprise LAN) via (optional) routers, hubs, and/or switches.



**Figure 2** Example Site LAN Networks

## Typical Private LAN Networks

In this guide, a **private LAN network** is defined as a local area network (LAN) in which LAN-enabled instruments and Windows PCs are **NOT** connected to a site LAN (workgroup LAN, Intranet, or enterprise LAN). This figure shows some example private LAN networks.



**Figure 3** Example Private LAN Networks

**NOTE**

For a private LAN you, the designer of the LAN, are the “System Administrator” for the LAN and are responsible for defining all private LAN parameters.

## Step 3: Gather Network Information

### 3 Gather Network Information

Gather network parameters for site LAN connections or define private LAN parameters.

Site LAN Network Information Card	
<b>Instrument Information (Completed by Instrument User)</b> (Serial Number and Ethernet (MAC) Hardware Address usually on label on instrument)	
Instrument Serial Number:	_____
Ethernet (MAC) Hardware Address:	_____

**What's in This Step?** When you have selected the type of LAN to connect your instrument to, the next step is to gather information for your site LAN or to define parameters for your private LAN. When you have gathered the necessary information, go to [Step 4: Connect Your Instruments](#).

- For steps to gather site LAN network information, go to [Gather Site LAN Network Information](#).
- For steps to select private LAN network parameters, go to [Define Private LAN Network Parameters](#).

## Gather Site LAN Network Information

Suggested steps follow to gather information for instrument connections to a site LAN.

- 1 Does the Site LAN Support DHCP?** Find out if the site LAN supports Dynamic Host Configuration Protocol (DHCP). In general, if the site LAN supports DHCP, you can connect your instruments to the site LAN without contacting your IT department.

However, if your local policy requires it, contact your System Administrator in your Information Technology (IT) department and tell him/her you want to connect instruments to the LAN.

- 2 Select Automatic Configuration.** If the site LAN supports DHCP, IP addresses for the instruments are automatically assigned by the network. If you do not require fixed (static) IP addresses for your instruments, you do **not** need to complete the *Site LAN Network Information Card*. Go to [Step 4: Connect Your Instruments](#).

**OR**

**Select Manual Configuration.** If the site LAN does not support DHCP or if you want to set a fixed (static) IP address for each instrument, the System Administrator must assign the static IP addresses and you must manually configure your instruments.

- 3 Enter Instrument Information.** If the LAN does not support DHCP or if you selected manual configuration, make a copy of the *Site LAN Network Information Card* for each instrument to be connected to the site LAN. Record the instrument serial number and hardware (MAC) address on the **Instrument Information** section of the card.
- 4 Give the Cards to the System Administrator.** Give the partially completed card(s) to the System Administrator and ask him/her to complete the **Site LAN Information** part of each card and to return the card(s) to you. Then, go to [Step 4: Connect Your Instruments](#).

<b>Site LAN Network Information Card</b>	
<b>Instrument Information (Completed by Instrument User)</b>	
(Serial Number and Ethernet (MAC) Hardware Address usually on label on instrument)	
Instrument Serial Number:	_____
Ethernet (MAC) Hardware Address:	_____
<b>Site LAN Information (Completed by System Administrator)</b>	
Does the Network Support DHCP?	Yes ___ No ___
If No, provide:	IP Address (Static): _____
	Subnet Mask: _____
	Default Gateway: _____
Does the Network Support DNS?	Yes ___ No ___
If Yes, provide:	Instrument Host Name: _____
	Instrument Domain Name: _____
	DNS Server IP Address: _____
Will You Allow Universal Plug&Play to be Enabled?	Yes ___ No ___

## Define Private LAN Network Parameters

Suggested steps follow to define parameters for a private LAN.

### NOTE

For a private LAN you, the designer of the LAN, are the “System Administrator” for the LAN and are responsible for defining all private LAN parameters. However, you may want to check with the System Administrator in your Information Technology (IT) department for guidelines on designing your private LAN.

---

- 1 Does the LAN Support DHCP?** In general, private LANs with a router support DHCP but typically do not support Dynamic DNS. Determine if the LAN supports DHCP.

### NOTE

Most Agilent products and most PCs will automatically choose an IP address via auto-IP, if a DHCP server is not present.

---

- 2 Complete the Network Information Card.** Make a separate copy of the *Private LAN Network Information Card* for each instrument to be connected to the private LAN. For each card, define and record the network, PC settings, and instrument settings you want for the LAN. As required, see [TCP/IP Network Basics](#) or the [Glossary](#) for a definition of terms.
- 3 Where to Go Next.** When you have completed the *Private LAN Network Information Cards*, go to [Step 4: Connect Your Instruments](#).

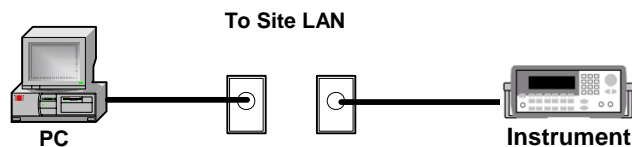


Private LAN Network Information Card	
<b>N E T W O R K</b>	DHCP Enabled:            Yes ___ No ___ Dynamic DNS Enabled:    Yes ___ No ___ UPnP Enabled OK:        Yes ___ No ___ Subnet Mask:             _____ DNS Server IP Address:   _____
<b>P C</b>	Hardware Address: _____ IP Address:                _____ Subnet Mask:             _____ DNS Server:               _____ Host Name:                _____ Domain Name:             _____
<b>I N S T R U M E N T</b>	Instrument Serial Number: _____ Ethernet (MAC) Hardware Address: _____ IP Address:                _____ Subnet Mask:             _____ DNS Server:               _____ Host Name:                _____ Domain Name:             _____ DHCP:                    ON___            OFF___ Auto IP:                  ON___            OFF___            N/A___ UPnP:                    Enabled___      Disabled___      N/A___

## Step 4: Connect Your Instruments

### 4 Connect Your Instruments

Connect your instruments to the site LAN or private LAN you selected.



**What's in This Step?** This section gives guidelines to connect your instruments and LAN interface devices such as routers, hubs, or switches to site LANs or to private LANs. When you have connected your instruments, go to [Step 5: Configure Your Instruments](#).

To connect instruments to a site LAN, go to [Connect Instruments to Site LAN](#).

- To connect instruments to a private LAN, go to [Connect Devices to Private LAN](#).

## Connect Instruments to Site LAN

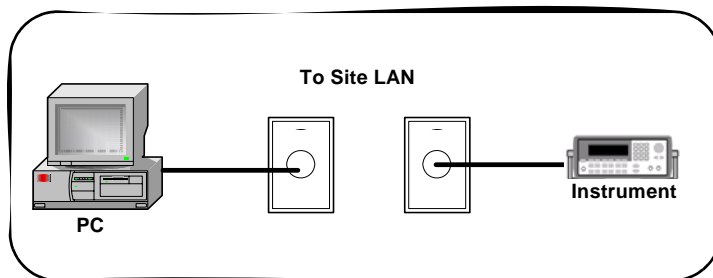
This section shows typical ways you can connect LAN-enabled instruments and interface devices such as routers, hubs, and switches to a site LAN.

Two example network connections follow that you can use as templates to connect your instruments to the site LAN. Modify the connections as required for your application. When you have completed the instrument and device connections, go to [Step 5: Configure Your Instruments](#).

### NOTE

The example connection steps assume the PC is configured for LAN operation and connected to an existing site LAN. If this is not the case and you need to configure the PC, see [Configuring Your PC for LAN Operation](#).

### Example: Direct Connection (Site LAN)



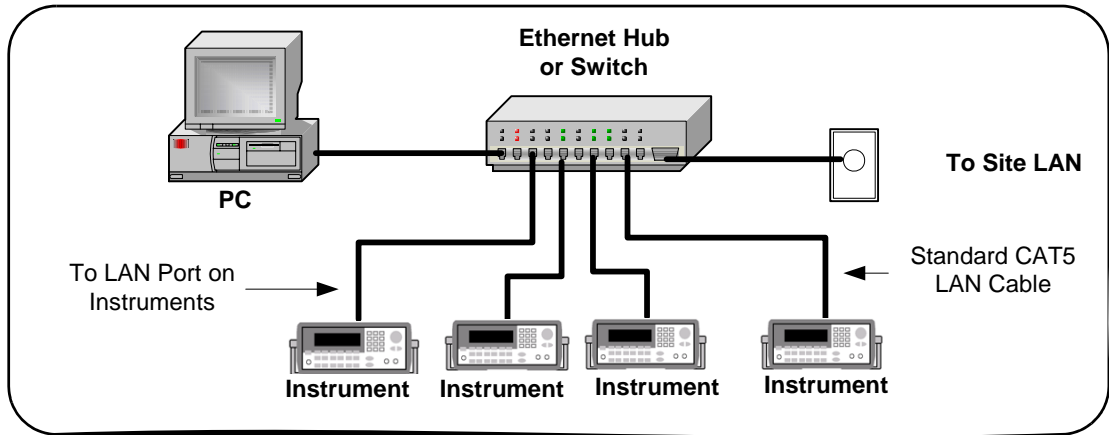
**Figure 4** Direct Connection of Instruments to a Site LAN

- 1 Turn Power OFF.** Remove power to the instrument.
- 2 Connect Instrument to the LAN.** Use a standard CAT5 LAN cable to connect the instrument to the site LAN.
- 3 Turn Power ON.** Apply power to the instrument and verify proper instrument power-on sequence. See the instrument's *User's Guide* for information.

## 2 Connecting Instruments to LAN

### Example: Switch Connections (Site LAN)

This figure shows one way to connect instruments to a site LAN via an Ethernet hub or switch.



**Figure 5** Connecting Instruments to a Site LAN via a Hub or Switch

- 1 Turn Power OFF.** Remove power from the instruments.
- 2 Connect Instruments to Hub/Switch.** If you use a hub or switch, use standard CAT5 LAN cables to connect each instrument to the hub or switch. See the hub/switch documentation for connections.
- 3 Turn Power ON.** Apply power to the hub/switch (if not already ON) and then to the instruments and verify proper power-on sequence. See the instrument's *User's Guide* for instrument information.

## Connect Devices to Private LAN

This section shows typical ways you can connect LAN-enabled instruments and interface devices such as routers, hubs, and switches to a private LAN, assuming your PC is properly configured for LAN operation.

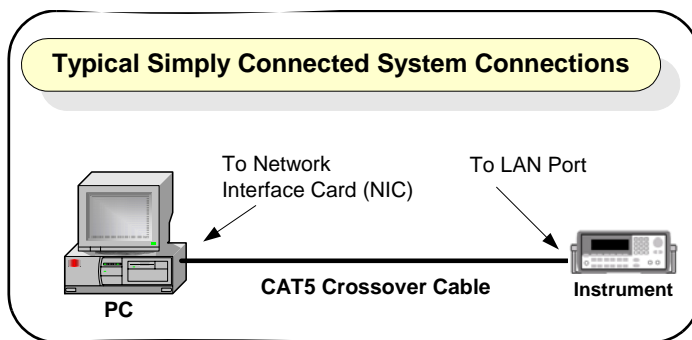
Two example network connections follow that you can use as templates to connect your instruments to a private LAN. Modify the connections as required for your application. When you have completed the instrument and device connections, go to [Step 5: Configure Your Instruments](#).

### NOTE

The example connection steps assume the PC is properly configured for LAN operation. If this is not the case and you need to configure the PC, see [Configuring Your PC for LAN Operation](#).

### Example: Direct Connections (Private LAN)

This figure shows one way to directly connect a Windows PC with a LAN-enabled instrument. Suggested connection steps follow.



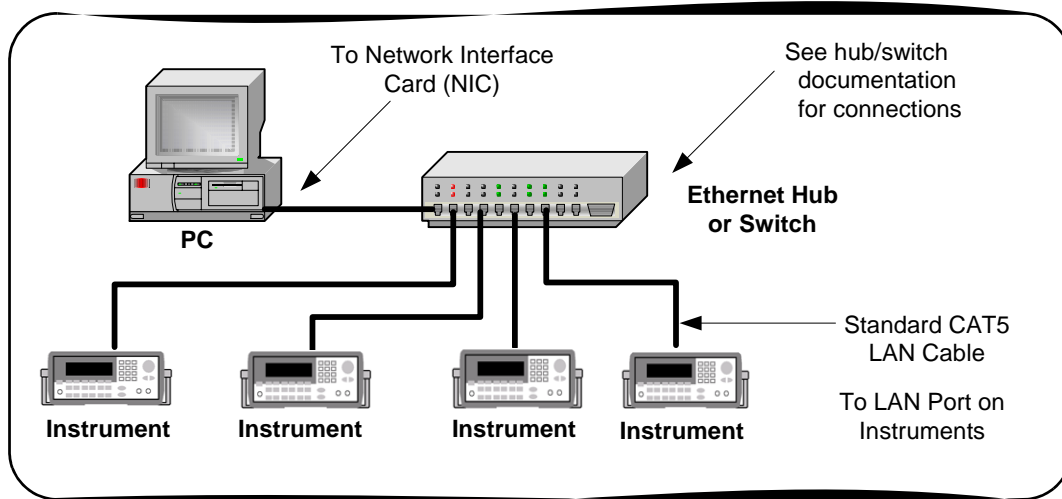
**Figure 6** Typical Direct Connection of Instrument to PC

- 1 Turn Power OFF.** Remove power from the instrument.
- 2 Connect Instrument to the LAN.** Use a standard CAT5 crossover cable to connect the instrument to the LAN.

- 3 Turn Power ON.** Apply power to the instrument and verify proper instrument power-on sequence. See the instrument's *User's Guide* for information on instrument power-on sequences.

### Example: Switch Connections (Private LAN)

This figure shows one way to connect a Windows PC and LAN-enabled instruments via an Ethernet hub or switch. Suggested connection steps follow.



**Figure 7** Example Connections via a Hub or Switch (Private LAN)

- 1 Turn Power OFF.** Remove power to instruments and the hub/switch.
- 2 Connect Instruments to Hub or Switch.** Use standard CAT5 LAN cables to connect each instrument to the hub or switch. See the hub or switch documentation for connection information.
- 3 Turn Power ON.** Apply power to the hub/switch (if not already ON) and then to the instruments and verify proper power-on sequence. See the instrument's *User's Guide* for information on instrument power-on sequences.

## Step 5: Configure Your Instruments

### 5 Configure Your Instruments

As required, manually configure your instruments using the instrument's Web pages or front panel.

You can use the **Configuring Your Instrument** Web page to configure instruments.

Configuring your 20 MHz FunctionArbitrary Waveform Generator

Note: You must click "Save" before changes to parameters become effective. Parameters marked with an asterisk(\*) also require that the instrument be rebooted before changes take effect.

Unit Edit Save Refresh Defaults Factory Defaults

Parameter	Configured Value	Factory Default
IP Address	192.168.1.100	192.168.1.100
Subnet Mask	255.255.255.0	255.255.255.0
Default Gateway	192.168.1.1	192.168.1.1

**What's in This Step?** This step shows how to manually set TCP/IP parameters for LAN-enabled instruments, using the instrument's Web Pages or front panel.

- If your LAN supports DHCP **and** you selected automatic configuration for your instruments, **skip this step** and go to [Step 6: Run Agilent Connection Expert](#).
- If the LAN does not support DHCP or you want to make configuration changes, you can use the steps in this section to configure your instruments. Then, go to [Step 6: Run Agilent Connection Expert](#).

## Check Your Web Browser.

If your instrument is Web-enabled, the instrument includes a Web Server you can access using a supported Web Browser (Internet Explorer 5.0 or higher). You can then use your Web Browser and the instrument's Web Pages to view/modify network configuration parameters as required.

### NOTE

If your instrument is not Web-enabled or you cannot access the instrument at its IP address, you must set the TCP/IP parameters from the instrument's front panel.

---

## Set/Change LAN Parameters

- 1 Display the Instrument Welcome Page.** To display an instrument's **Welcome Page**, determine the instrument's IP address from the front panel display. Then, open your Web Browser. From the Web Browser address line, type '**http://<Instrument IP Address>**', where **<Instrument IP Address>** is the IP address displayed on the front panel, and press **Enter** to display the instrument's **Welcome Page**.

For example, if the instrument's current IP address is 169.254.3.2 (as displayed on the instrument's front panel), typing **http://169.254.3.2** and pressing **Enter** displays the instrument's **Welcome Page**. The following figure shows a typical display for the Agilent 33220A.




Welcome to your **Web-Enabled 20 MHz Function/Arbitrary Waveform Generator**

**Information about this Web-Enabled Instrument**

<b>Instrument:</b>	33220A 20 MHz Function/Arbitrary Waveform Generator
<b>Serial Number:</b>	US 0123456
<b>Description:</b>	Agilent 33220A (US 0123456)
<b>Hostname:</b>	a-33220a-23456.ltd.agilent.com
<b>IP Address:</b>	156.140.105.104
<b>VISA TCP/IP Connect String:</b>	TCPIP0::a-33220a-23456::INSTR

[Advanced information](#)



**2 View Current Configuration Page.** To view the current configuration, click the **View and Modify Configuration** icon to display the **Current Configuration Page**. An example display for the 33220A follows.

**Current Configuration of 20 MHz Function/Arbitrary Waveform Generator**

[Modify Configuration](#)

Parameter	Currently in use
DHCP:	ON
IP Address:	156.140.105.104
Subnet Mask:	255.255.254.0
Default Gateway:	156.140.104.1
DHCP Server:	156.140.104.102
DNS Server:	156.140.2.11
Hostname:	a-33220a-23456
Domain:	ltd.agilent.com
Description:	Agilent 33220A (US 0123456)
Lan Keepalive Timeout:	1800

**3 Display the Configuring the Instrument Page.** From the **Current Configuration Page**, click the **Modify Configuration**

box to display the **Configuring the Instrument Page**. You can then set/change instrument parameters as required

- 4 **Set/Change TCP/IP Parameters.** This figure shows a partial **Configuring your 20MHz Function/Arbitrary Waveform Generator Page**. For example, to change the settings for the 33220A IP address and default gateway from the values shown in the **Configured Value** column to those shown in the **Edit Configuration** column, type the desired values in the **Edit Configuration** column. Then click **Save**. Click **Reboot 33220A** to make the changes effective.

**NOTE**

Since the IP Address, Subnet Mask, and Default Gateway parameters are marked with an asterisk (\*), you must click **Save** and then click **Reboot 33220A** to make the changes effective.

The **Configured Value** for the IP Address, Subnet Mask and Default Gateway are NOT necessarily the values currently set for the instrument. These values will be used during boot if DHCP is OFF or is unavailable.

**Configuring your 20 MHz Function/Arbitrary Waveform Generator**

*Note: You must click "Save" before changes to parameters become effective. Parameters marked with an asterisk(\*) also require that the instrument be rebooted before changes take effect.*

Parameter	Configured Value	Edit Configuration
DHCP: *	ON	<input type="radio"/> OFF <input checked="" type="radio"/> ON
These LAN parameters will be used during boot if DHCP is OFF or unavailable		
IP Address: *	156.140.104.232	<input type="text" value="156.140.104.232"/>
Subnet Mask: *	255.255.254.0	<input type="text" value="255.255.254.0"/>
Default Gateway: *	156.140.104.1	<input type="text" value="156.140.104.1"/>

## Step 6: Run Agilent Connection Expert

### 6 Run Agilent Connection Expert

Run Agilent Connection Expert; add and configure the LAN interface (optional); add a LAN instrument; communicate with your instruments using Interactive IO; and program your instruments (optional).



**What's in This Step?** This step shows how to

- Run Agilent Connection Expert
- Add and configure a LAN interface (optional)
- Add a LAN instrument
- Communicate with your instruments
- Program your instruments (optional)

## Run Agilent Connection Expert

Click the Agilent IO Control icon in your Microsoft notification area,



and select **Agilent Connection Expert**, or click **Refresh All** in the explorer pane if Connection Expert is already running.

## Add a LAN interface

One LAN interface labeled LAN (TCPIP0) appears in the explorer pane by default. If there is not a LAN interface available, or you wish to connect different instruments via LAN with different connection parameters (such as different connect timeouts), then take the following steps:

- 1 Click **Add Interface** on the Connection Expert toolbar to create an additional LAN interface.
- 2 When the **Manually Add an Interface** box appears with LAN interface highlighted, click **Add**.
- 3 When the **LAN Interface** configuration box appears, modify the properties as desired, then click **OK**.

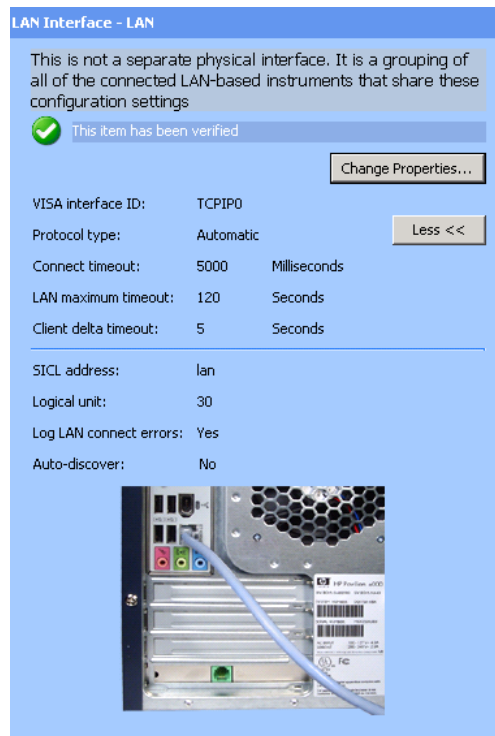
### NOTE

The required Auto-discover setting for LAN interfaces is **off**. To prevent disrupting your network traffic, Connection Expert does not automatically query each bus on the LAN to detect the presence of new devices.

## Configure a LAN Interface

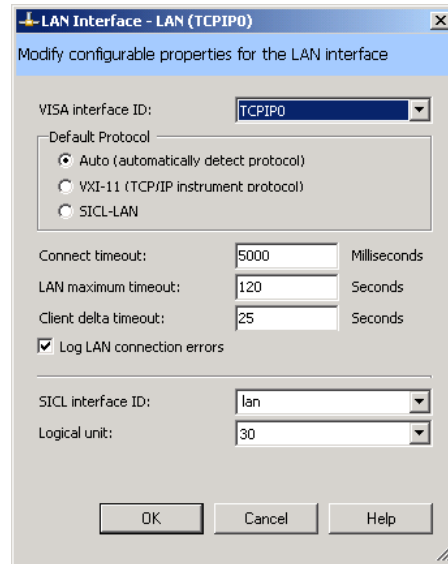
When you select a LAN interface in the explorer pane, the properties pane displays the current properties for that interface. The properties that you are most likely to need are displayed at the top of the properties pane. If you see a button labeled More, you may click it to see additional properties, as shown in the graphic below.

- 1 Click **More** to view the full property list.



**Figure 8** LAN Interface Properties Pane

- 2 Select a **LAN** interface in the explorer pane. Click the **Change Properties...** button in the properties pane. The LAN Interface dialog box appears, as shown below.



**Figure 9** LAN Interface Configuration Dialog Box

- 3 Change the properties and click **OK**. Your changes will appear in the properties pane.

**VISA Interface ID.** A symbolic name that is used to uniquely identify this interface. The VISA interface ID combines the interface type and a numeric identifier. For example, TCPIP0 is the default VISA interface ID for a TCPIP interface.

**Protocol type.** The protocol type to be used with the LAN client software on this PC. The Agilent IO Libraries Suite supports three protocol choices: Automatic (automatically detect protocol), VXI-11 (TCP/IP Instrument protocol), and SICL-LAN protocol.

**Default selection:** Automatic

**Connect timeout.** The time, in milliseconds, that the PC will wait when attempting to connect to a LAN instrument. Default value: 5000 milliseconds

**LAN maximum timeout:** The actual timeout value used when a test program specifies a timeout value of infinity. Default value: 120 seconds

**Client delta timeout.** The incremental time that is added to the timeout value specified in a test program to allow for the additional time required to transfer information over a LAN connection. Default value: 25 seconds

**SICL interface ID.** The unique name that SICL programs use to identify this interface.

**Logical unit.** A number used to uniquely identify an interface in Agilent VEE and SICL applications. The logical unit number is an integer in the range of 0-99.

**NOTE**

The logical unit may be used in place of the SICL interface ID in Agilent VEE and SICL applications. The logical unit is not used by VISA.

**Log LAN connect errors.** You can set Connection Expert to log LAN connection error information in your Windows Event Viewer or Event Log. Default setting: Yes

**Auto-discover.** Auto-discover is not shown in the Change Properties dialog box, because it is required to be Off (No) for a LAN interface.

## Add and Configure a LAN Instrument

When an instrument resides on your local area network (LAN), there are several ways to add the instrument to your test system configuration. You may also use any of these methods to make a change to an existing instrument.

### NOTE

Connection Expert only discovers LAN instruments automatically in your *local subnet* to prevent disrupting your network traffic. In practical terms, your *local subnet* is defined as being on your side of the nearest router. To communicate with LAN instruments that are *remote* (on the other side of the router), you must be able to specify a Hostname or IP address in the LAN Instrument dialog box.

### Connecting an instrument on your LAN local subnet

When an instrument is on your LAN local subnet, the easiest way to add the instrument is to request that Connection Expert discover the instrument. This eliminates the need to specify the IP address or the instrument hostname.

### Connecting an instrument *outside* your LAN local subnet

When an instrument resides outside your LAN local subnet, you may specify the instrument by either the IP address or the instrument hostname.

Connection Expert can discover any server on the local subnet that uses either the VXI-11 or the SICL-LAN protocol.

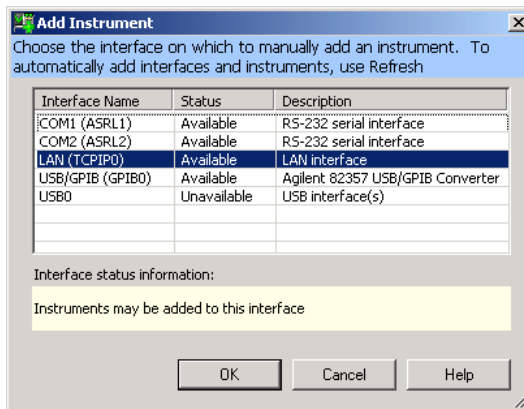
After Connection Expert discovers a remote instrument on the LAN, it performs the same bus addressing procedures as it would for a local instrument. If Connection Expert can open a connection to an instrument, it considers that the instrument has been “discovered” (even though Connection Expert does not send or receive any information).

To discover an instrument on your local subnet,

- 1 Select the LAN interface node in the explorer pane, **LAN (TCPIP0)** for this example. Right-click to get a menu. Then click **Add Instrument**. (Alternatively, you could click



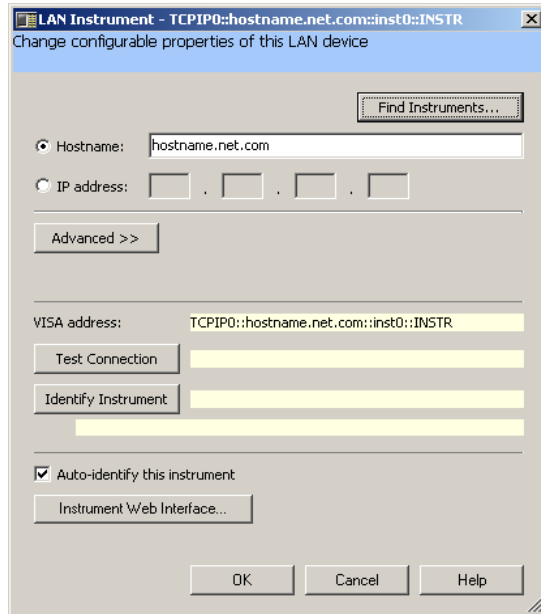
the **Add Instrument** button on the tool bar, the Task Guide, or the I/O Configuration menu.) The Add Instrument dialog box appears, as shown.



**Figure 10** The Add Instrument Dialog Box

- 2 Leaving the LAN interface selected, click **OK**. The LAN instrument configuration dialog box appears, as shown.

## 2 Connecting Instruments to LAN



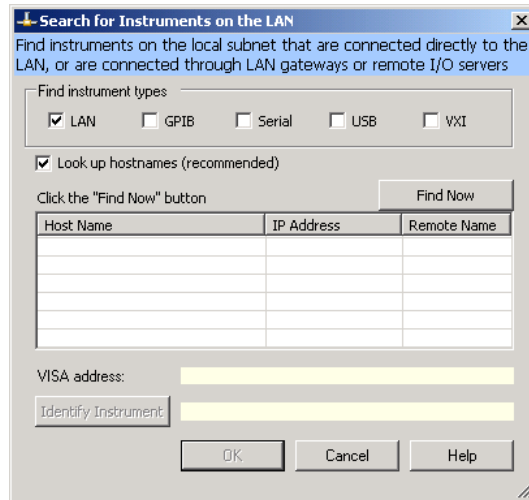
**Figure 11** LAN Instrument Configuration Dialog Box

Click **Find Instruments...**

### NOTE

**Find Instruments** only searches in your local subnet. In practical terms, your **local subnet** is defined as instruments on your side of the nearest router. To communicate with LAN instruments that are remote (on the other side of the router), you must be able to specify a **Hostname** or **IP address** in the LAN Instrument dialog box above.

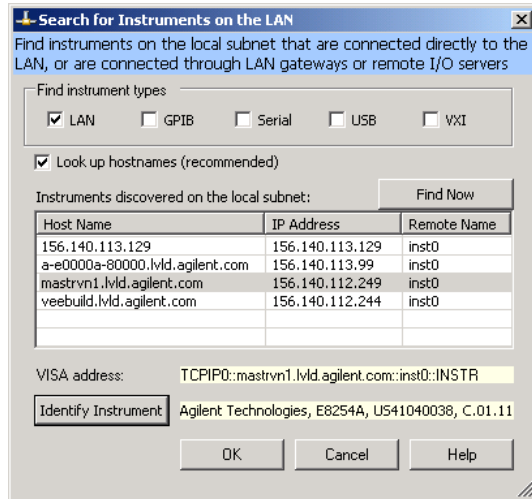
- 3 When the **Search for instruments on the LAN** dialog box appears (as shown below), leave **LAN** selected, and click **Find Now**.



**Figure 12** Search for Instruments on the LAN Dialog Box

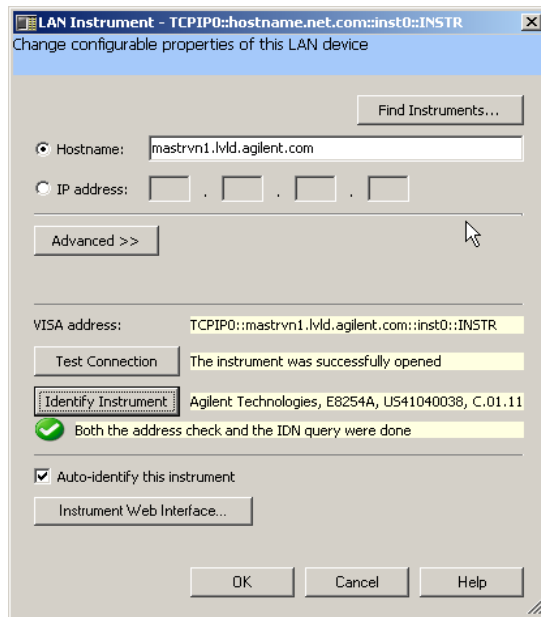
## 2 Connecting Instruments to LAN

- 4 When the LAN instruments on the subnet are discovered as shown below, select the one of interest, click **Identify Instrument** (if desired), then **OK**.



**Figure 13** Discovery of LAN Instruments on the Subnet

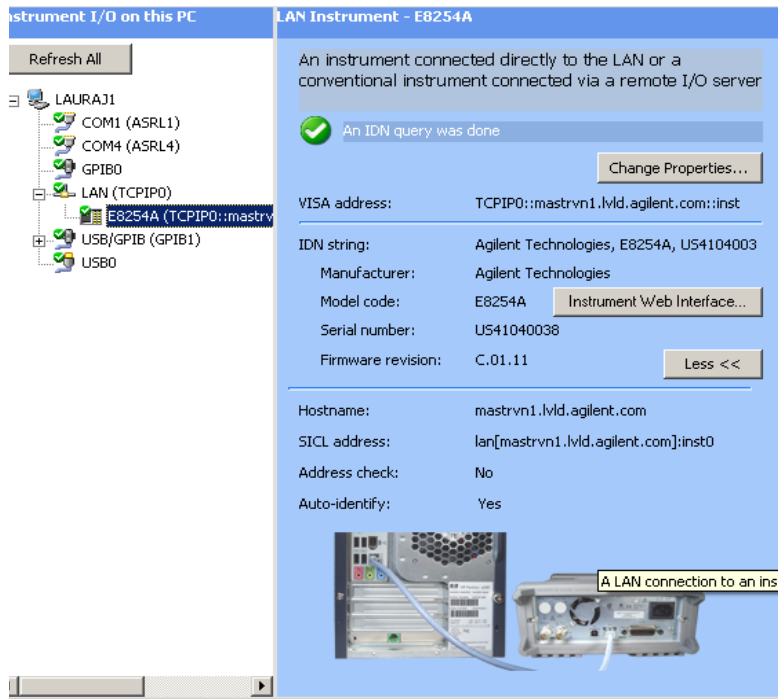
- 5 Click **OK** when you are satisfied that the correct instrument has been located and selected. You now see the LAN Instrument configuration dialog box displayed including information from your selected instrument, as shown below.



**Figure 14** Configuration Properties for Selected Instrument

Automatically, you have the IP address, the Hostname, a default Remote name, the VISA address, a verified test connection. The instrument's identity has been verified. Click **OK** to return to the main Connection Expert window. Now you see the LAN instrument added to the explorer pane and its properties enumerated in the properties pane, as shown.

## 2 Connecting Instruments to LAN



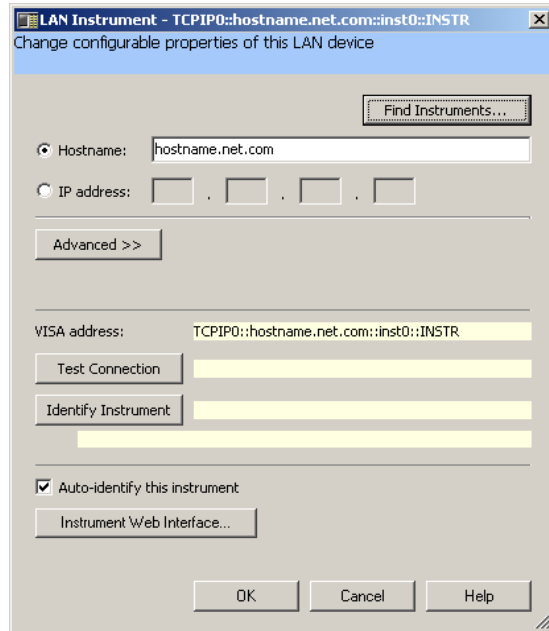
**Figure 15** Newly Added LAN Instrument in Connection Expert

To add a LAN instrument by specifying the hostname or IP address

- 1 In the explorer pane, click the LAN interface node where you want to add the instrument.
- 2 Right-click the interface node, and select **Add Instrument** from the drop-down menu
- 3 When the **Add Instrument** dialog box appears with the LAN interface highlighted, click **OK**.
- 4 When the **LAN Instrument** configuration dialog box appears, as shown below, select and edit either the **IP address** or the **Hostname**. With either choice, you will also need to enter the remote name for the instrument.

*Optional tests:* Click **Test Connection**. Click **Identify Instrument**.

Click **OK**.



**Figure 16** Editing Properties in the LAN Instrument Dialog Box

## Communicate with Instruments Using Interactive IO

You can use the Interactive IO utility within Connection Expert, VISA Assistant, or the Telnet utility to verify communication with instruments via the LAN.

This section gives guidelines to communicate with your instruments using Interactive IO.

### NOTE

Communication with installed LAN instruments was established in [Run Agilent Connection Expert](#) if the instruments were visible and verified in the Connection Expert explorer view. Therefore, this is an optional step you can use to verify communication with instruments.

---

Interactive IO is a software utility that lets you interactively send commands to instruments and see their responses without writing any program code.

You can use Interactive IO to quickly verify connectivity to your instrument, to troubleshoot communication problems, to learn the instrument's command set, and to rapidly prototype commands and check the instrument's responses before writing code.

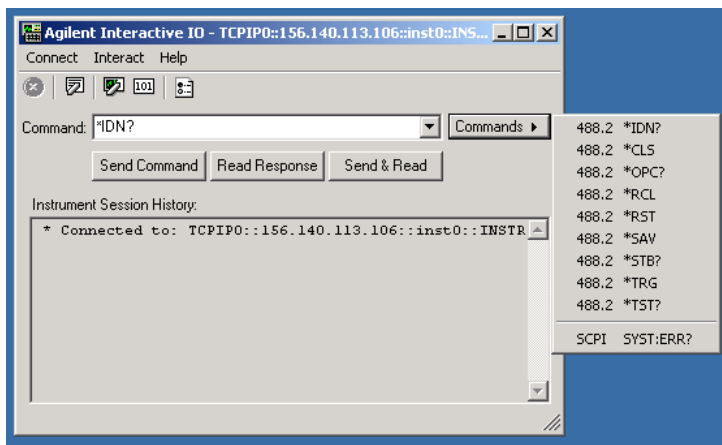
Start Interactive IO from within Connection Expert, either by clicking the *Send commands to this instrument* task in the task guide, or, on the Connection Expert menu bar, by clicking **Tools > Interactive IO**.

### Example using Interactive IO

- 1 Select the desired instrument in the explorer view. Click **Tools > Interactive IO**. Interactive IO gives you a number



of common IEEE 488.2 and SCPI commands for communicating with instruments, as shown below.



**Figure 17** Common 488.2 and SCPI Commands in Interactive IO

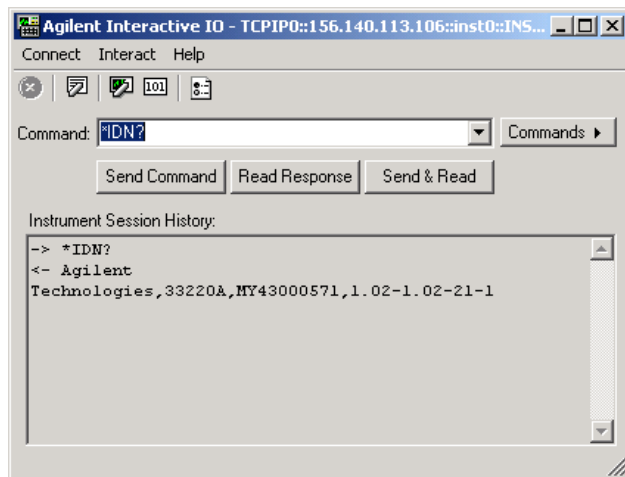
You can also type in instrument-specific commands (from your instrument's programming guide). The common 488.2 commands and their meanings are listed in the table below.

488.2 Commands	Meaning
*IDN?	The Identification query returns manufacturer, model, serial number, and firmware level or equivalent.
*CLS	The Clear Status command clears status data structures, and forces the device to the Operation Complete query idle state.
*OPC?	The Operation Complete query places an ASCII character "1" into the device's Output Queue when all pending selected device operations have been finished.
*RCL	The Recall command restores the current settings of a device from a copy stored in local memory.
*RST	The Reset command performs a device reset, which sets the device-specific functions to a known state that is independent of the past-use history of the device.

## 2 Connecting Instruments to LAN

<b>*SAV</b>	The Save command stores the current settings of the device in local memory.
<b>*STB?</b>	The Read Status Byte query allows the programmer to read the status byte and master Summary Status bit.
<b>*TRG</b>	The Trigger command signals the transition of the Device Trigger (DT) function to the Device Trigger Active State (DTAS).
<b>*TST?</b>	The Self-test query returns zero to indicate the test completed with no errors. A return value not equal to zero indicates the test is not completed or completed with errors.
<b>SCPI Commands</b>	
<b>SYST:ERR?</b>	The queue query message is a request for the next entry from the instrument's error/event queue. Items in this queue contain an integer in the range [-32768, 32767] denoting an error/event number and associated descriptive text. Negative numbers are reserved by the SCPI standard; positive numbers are instrument-dependent. A value of zero indicates that no error or event has occurred.

2 Open the **Commands** menu and select a command. The example below shows a \*IDN? command selected.



**Figure 18** \*IDN? Command Example in Interactive IO

- 3 Click **Send & Read** to send the command and receive a response. The results are displayed in the **Instrument Session History**.

Alternatively, you can click the **Send Command** and **Read Response** buttons, when you desire, to control the time gap between these commands.

Interactive IO has a default timeout value set (5000 ms.), which may not be long enough for your particular application. To change that default, go to **Interact > Options**, edit the **Timeout** value, and click **OK**.

## Communicate with Instruments Using Telnet

You can use the Interactive IO utility within Connection Expert, VISA Assistant, or the Telnet utility to verify communication with instruments via the LAN.

This section gives guidelines to communicate with your instruments using the Telnet utility.

### NOTE

Communication with installed LAN instruments was established in [Run Agilent Connection Expert](#) if the instruments were visible and verified in the Connection Expert explorer view. Therefore, this is an optional step you can use to verify communication with instruments.

- 1 Display the instrument's Welcome Page as described on [page 32](#).
- 2 Find the SCPI Telnet Port for this instrument. For many Agilent instruments, you can find this port number by clicking **Advanced information** on the **Welcome** page. For most Agilent products, the SCPI Telnet Port is 5024.
- 3 Open a DOS window. (On Windows XP, click **Start > All Programs > Accessories > Command Prompt**.)
- 4 Optionally, test your connection by typing:

```
ping XXX.XXX.XXX.XXX
```

where XXX.XXX.XXX.XXX is the IP address of the product. If the connection is established, you will see several status lines indicating replies from the instrument to the **ping** command. Otherwise, you will see the message **Request timed out**.

- 5 Run the Microsoft Telnet Client. (On Windows XP, click **Start > Run** and type

```
telnet
```

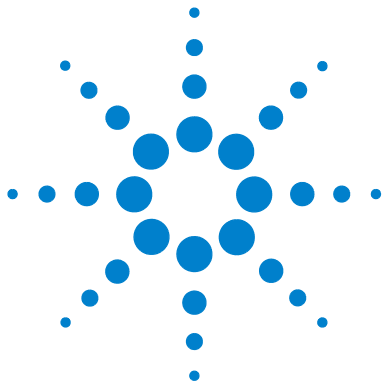
- 6 At the Telnet command line, type:

```
open XXX.XXX.XXX.XXX PPPP
```

where XXX.XXX.XXX.XXX is the instrument's IP address and **PPPP** is the port (5024 for most Agilent products). You should see a response from the instrument, such as:

**Welcome to Agilent's 33220A Waveform Generator**

- 7 Enter commands for the product. CTRL-C is typically a device clear; check your instrument documentation for other commands.
- 8 Close the Telnet window to close the connection to the instrument and exit Telnet.



## 3 Connecting Instruments to USB

This chapter includes:

- USB Quick Start
- Step 1: Install I/O Software on Your PC
- Step 2: Connect Instruments to USB
- Step 3: Identify Instruments Automatically
- Step 4: Communicate with Instruments
- (Optional) Step 5: Program Your Instruments



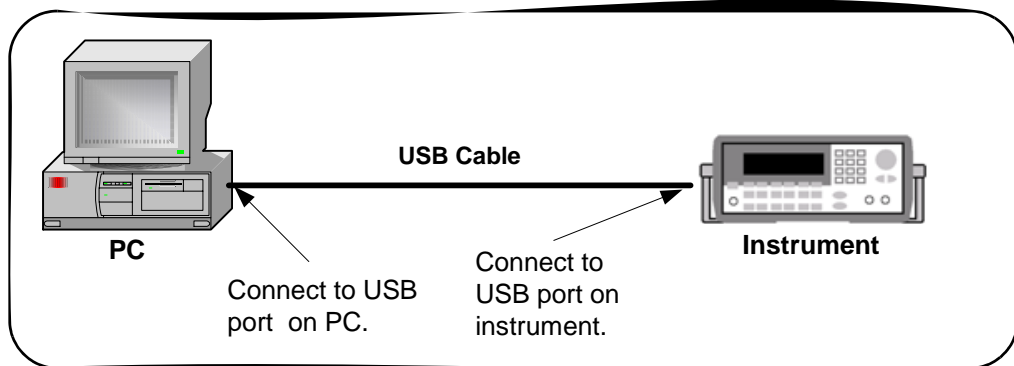
## USB Quick Start

This section shows suggested steps to help you quickly get started connecting USB-enabled instruments to the Universal Serial Bus (USB).

**NOTE**

Optionally, a USB hub may be connected between the PC and USB instrument(s). However, this configuration is not described in this guide. See your USB hub documentation if you use a USB hub.

**Typical USB Interface System** In this guide, a **USB interface system** is defined as a system in which USB-enabled instruments are connected via a USB cable to a USB 1.1 port in a Windows PC or to a USB 2.0 port in a Windows XP PC (with Service Packs 1 or 2). This figure shows a typical USB interface system with a PC and a USB instrument connected via a USB cable.



**Figure 19** Typical USB Interface System

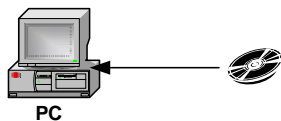
**Getting Started** To get started, you may want to print the figures for Steps 1 - 5 to use as a guide as you set up your USB system. See the associated step if you need more details.

### Other Information You Can Use

- [Troubleshooting USB Interfaces](#) for USB troubleshooting tips
- [Glossary](#) for a definition of some USB terms
- [Contacting Agilent](#)

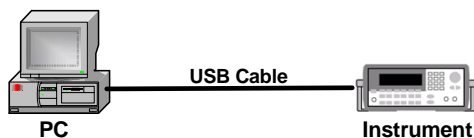
## 1 Install I/O Software on Your PC

Install Agilent IO Libraries Suite from a CD or the Web.



## 2 Connect Instruments to USB

Use USB cables to connect your instruments to the USB port(s) on your PC, or use the Agilent 82357A USB/GPIB Interface Converter.



## 3 Identify Instruments Automatically

With Connection Expert running, you can discover and identify your USB instruments automatically.

After the instruments are powered and connected to the PC, **Connection Expert** executes identification queries automatically.

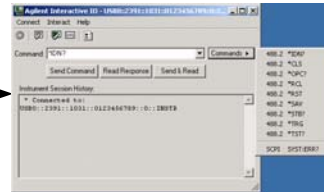




## 4 Communicate with Instruments (Optional)

Use **Interactive IO** within **Connection Expert** to send commands and read responses via the USB interface.

Use **Interactive IO** to send commands such as 488.2 `"*IDN?"` to your instruments.



## 5 Program Your Instruments (Optional)

This is an optional step. As desired, you can program instruments via the USB interface using applications such as IVI-COM, VXI*npn*, or VISA.

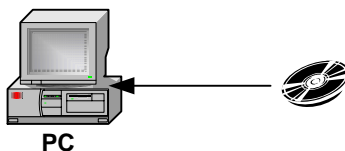
VISA: `viOpen (...,"UsbDevice1",...)`  
 SICL: `iopen ("UsbDevice1")`

VISA Alias

## Step 1: Install I/O Software on Your PC

### 1 Install I/O Software on Your PC

Install Agilent IO Libraries Suite from a CD or the Web.



**What's in This Step?** Install Agilent IO Libraries Suite and other I/O software on your PC before you connect instruments to USB. See [Installing I/O Software](#) for details. Then, go to [Step 2: Connect Instruments to USB](#)

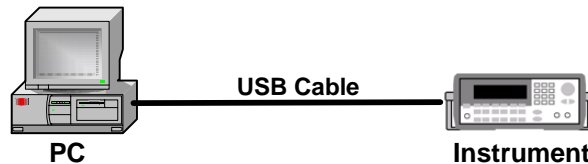
#### NOTE

Version M.01.00 or later of the Agilent IO Libraries is required for USB. This procedure documents the use of Agilent IO Libraries Suite 14.0.

## Step 2: Connect Instruments to USB

### 2 Connect Instruments to USB

Use USB cables to connect your instruments to the USB port(s) on your PC, or use the Agilent 82357A USB/GPIB Interface Converter.



**What's in This Step?** This step provides a USB interface overview, shows how to connect USB Instruments using the USB ports on your PC, or by using the Agilent 82357A USB/GPIB Interface Converter.

## USB Interface Overview

When a USB device that implements the USBTMC or USBTMC-USB488 protocol is plugged into the computer, the Agilent IO Libraries Suite automatically detects the device. It is important to note that only USBTMC and USBTMC-USB488 devices are recognized by the IO Libraries Suite. Other USB devices such as printers, scanners, and storage devices are not recognized and cannot be programmed with the Agilent IO Libraries Suite. “USB instruments” in the remainder of this section refers to devices that implement the USBTMC or USBTMC-USB488 protocol.

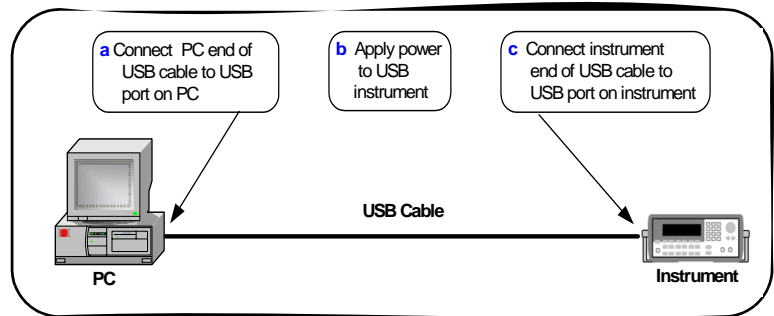
**Connecting an Instrument via USB or Converter.** The Universal Serial Bus (USB) is a quick and easy way to connect instruments to PCs on a benchtop. Some newer instruments have USB interfaces built in; for those that do not, you can use the **Agilent 82357A USB/GPIB Interface Converter** to connect GPIB instruments to a USB port on your PC. You can use USB to connect a single instrument to a PC; or, with the 82357A, you can connect up to 14 GPIB instruments.

### NOTE

Do not confuse the **Agilent 82357 USB/GPIB Interface Converter** with a USBTMC device. The 82357 is automatically configured as a GPIB interface, not as a USBTMC device, when it is plugged into the computer.

## Steps to Connect USB Instruments

Use the steps in this figure to directly connect USB instruments to a Windows PC. If you have not yet installed I/O software on your PC, go to [Installing I/O Software](#) and install the software BEFORE you connect USB instruments to your PC. When you have made the connections for your system, go to [Step 3: Identify Instruments Automatically](#).



**Figure 20** Connecting a USB Instrument to a PC

**NOTE**

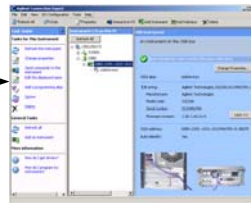
If you have not yet installed I/O software on your PC, go to [Step 1: Install I/O Software on Your PC](#) and install the software BEFORE you connect USB instruments to your PC. Be sure to use a USB 2.0-compliant USB cable, even if you are using USB 1.1 operation.

## Step 3: Identify Instruments Automatically

### 3 Identify Instruments Automatically

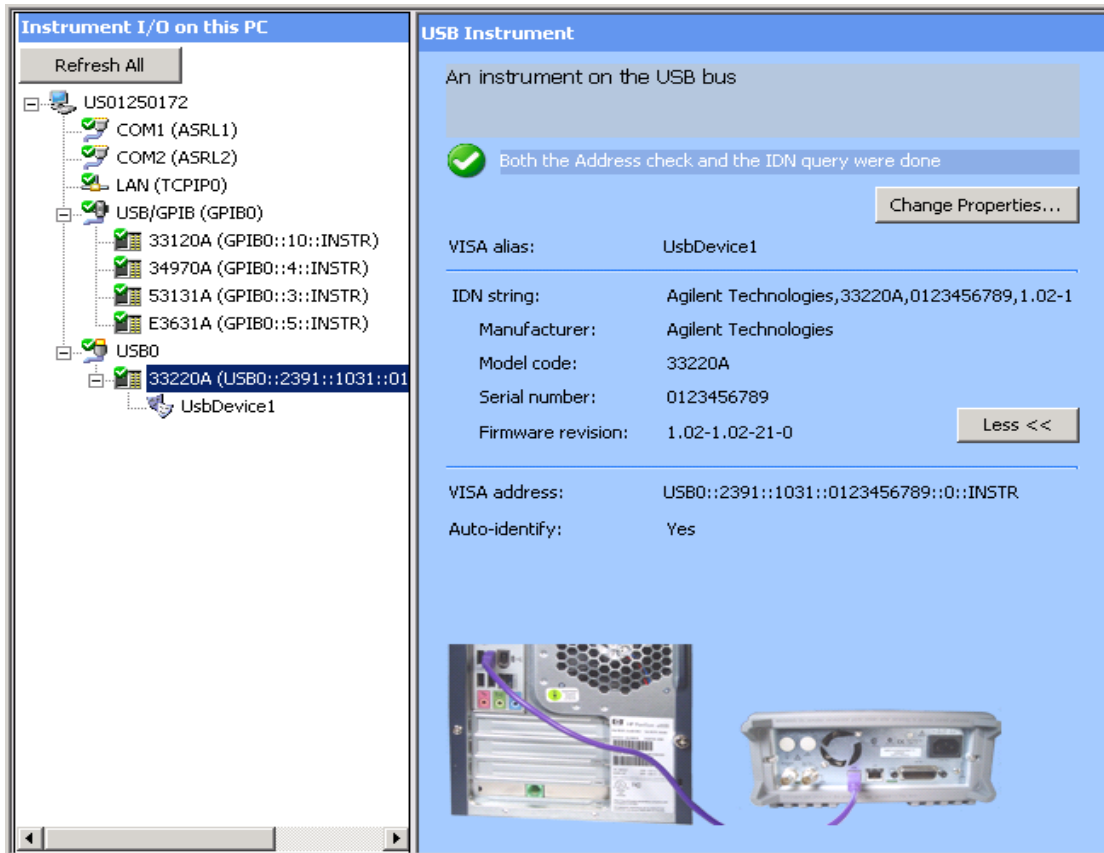
With Connection Expert running, you can discover and identify your USB instruments automatically.

After the instruments are powered and connected to the PC, Connection Expert executes identification queries automatically.



**What's in This Step?** This step provides an overview of how Connection Expert automatically identifies USB instruments, and how you can assign VISA aliases, or programming aliases, to address those instruments in a more convenient way.

**Automatic Identification of USB Devices.** When a USB instrument is plugged into the PC, the IO Libraries Suite adds a USB interface, a USB instrument, and a VISA alias (**UsbDevice1**, in this case) to the **Instrument I/O on this PC** explorer tree. Note that the USB/GPIB converter, **USB/GPIB (GPIB0)**, looks different from the USB interface.



**Figure 21** Appearance of the USB Converter vs. USB Interface

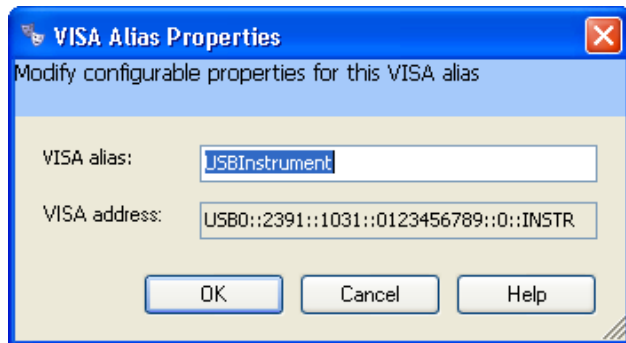
Both a listener check and an identification query are done automatically, as you can see on the right side in the **USB Instrument** pane next to the green-white check mark symbol.

#### NOTE

When you apply power, a **Found New Hardware** wizard may appear from the Windows operating system. Just step through the wizard clicking **Next**. (You don't need to insert a CD, since you have already installed the I/O software.) You may see two wizards when you connect a USB/GPIB interface converter. Don't worry; step through both. Then return to this section.

The instrument's identification parameters can also be seen, such as the IDN string, the manufacturer, the model code, the serial number, the firmware revision as well as pictures of the instrument and PC rear connection panels.

**Assigning a VISA Alias.** Connection Expert automatically assigns a default alias name. You can modify a VISA alias by right-clicking on the USB instrument in the Instrument IO pane and selecting **Add VISA alias** from the drop-down menu. A **VISA Alias Properties** dialog box appears.



**Figure 22** VISA Alias Dialog Box



You can change this VISA alias name to one of your own choosing before clicking **OK**.

**NOTE**

A VISA alias is a name of your choosing, which you assign to a device and use in your programs. Once assigned, the alias is a synonym for the device's VISA address, so you can use it to open a VISA session (using the **viOpen** function) and to get resource information (using **viParseRsrc** or **viParseRsrcEx**).

Using VISA aliases in your programs, rather than VISA addresses, provides two significant advantages:

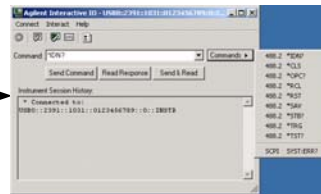
- **Portability.** If you program using aliases, you can run your program on a new test system, whose instruments are at different addresses, simply by using Connection Expert to create the same aliases on the new system as you created on your development system. Similarly, you can move or replace instruments without changing or recompiling test code, simply by changing the alias definitions in Connection Expert.
- **Readability.** Your programs will be much easier to read and understand if, for example, your multimeter is called “myDMM” instead of “GPIB2::14::8::INSTR”. This is particularly important in the case of USB instruments, whose VISA addresses are typically long and cumbersome, containing the instrument's serial number among other information.

## Step 4: Communicate with Instruments

### 4 Communicate with Instruments (Optional)

Use **Interactive IO** within **Connection Expert** to send commands and read responses via the USB interface.

Use **Interactive IO** to send commands such as 488.2 `"*IDN?"` to your instruments.



**What's in This Step?** This step shows how to use Interactive IO to send 488.2 commands to your USB instruments and read responses. Next, go to [\(Optional\) Step 5: Program Your Instruments](#).

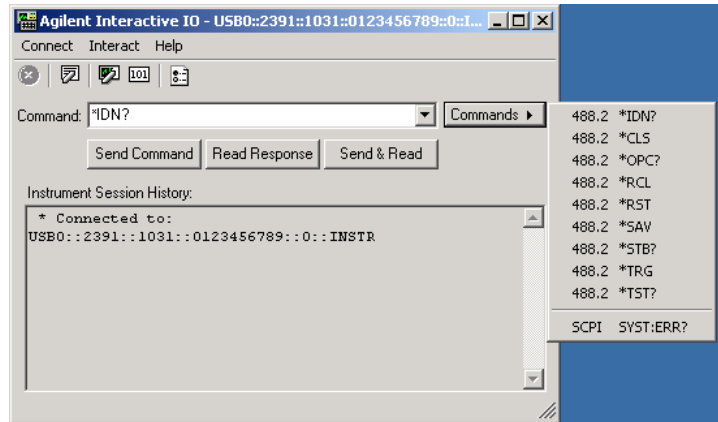
#### NOTE

Communication with installed instruments was established in [Step 2: Connect Instruments to USB](#). Therefore, this is an optional step you can use to verify communication with instruments.

## Steps to Check Identification Parameters

- 1 Highlight the USB instrument in the **Instrument I/O on this PC** pane. Then click the **Interactive IO** icon on the toolbar. (If you are using a USB/GPIB converter, then you would be highlighting a GPIB instrument.)

The **Interactive IO** dialog box appears. The **Commands** menu is selected, as shown below.



**Figure 23** Interactive IO with 488.2 Commands Displayed

- 2 Select a command, or enter one in the **Command** area.
- 3 Click **Send & Read** to send the command to the instrument and read a response. Alternatively, you may click **Send Command** and **Read Response**, as you desire.

The Instrument Session History area will display the results.

#### NOTE

If you are getting a timeout when sending and receiving commands, you can change your **Timeout** default setting of 500 ms in the **Interact > Options** dialog box.

488.2 Commands	Meaning
*IDN?	The Identification query returns manufacturer, model, serial number, and firmware level or equivalent.
*CLS	The Clear Status command clears status data structures, and forces the device to the Operation Complete query idle state.
*OPC?	The Operation Complete query places an ASCII character "1" into the device's Output Queue when all pending selected device operations have been finished.
*RCL	The Recall command restores the current settings of a device from a copy stored in local memory.
*RST	The Reset command performs a device reset, which sets the device-specific functions to a known state that is independent of the past-use history of the device.
*SAV	The Save command stores the current settings of the device in local memory.
*STB?	The Read Status Byte query allows the programmer to read the status byte and master Summary Status bit.
*TRG	The Trigger command signals the transition of the Device Trigger (DT) function to the Device Trigger Active State (DTAS).
*TST?	The Self-test query returns zero to indicate the test completed with no errors. A return value not equal to zero indicates the test is not completed or completed with errors.
SCPI Commands	
SYST:ERR?	The queue query message is a request for the next entry from the instrument's error/event queue. Items in this queue contain an integer in the range [-32768, 32767] denoting an error/event number and associated descriptive text. Negative numbers are reserved by the SCPI standard; positive numbers are instrument-dependent. A value of zero indicates that no error or event has occurred.

**NOTE**

Some instruments do not recognize any/all 488.2 commands.

## (Optional) Step 5: Program Your Instruments

### 5 Program Your Instruments (Optional)

This is an optional step. As desired, you can program instruments via the USB interface using applications such as IVI-COM, *VXIplug&play*, or VISA.

VISA: viOpen (...,"UsbDevice1",...)  
SICL: iopen ("UsbDevice1")

VISA Alias



**What's in This Step?** This is an optional step that involves communicating programmatically with your instruments, using I/O software such as IVI-COM or *VXIplug&play* instrument drivers, or I/O libraries, such as VISA.

### **3 Connecting Instruments to USB**



## 4 Connecting Instruments to GPIB

This chapter includes:

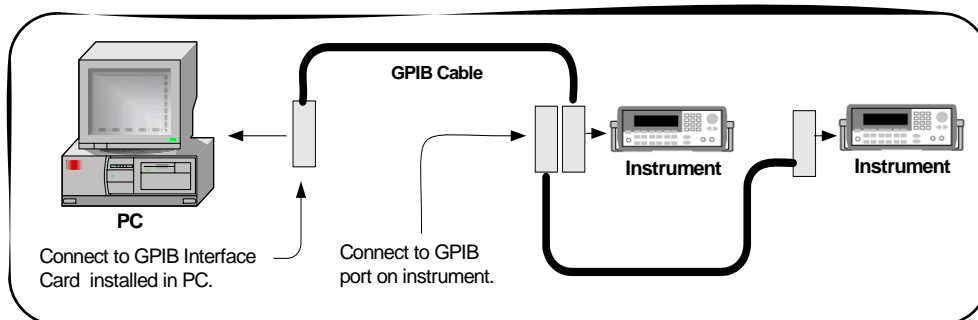
- GPIB Quick Start
- Step 1: Install I/O Software on Your PC
- Step 2: Install GPIB Cards in Your PC
- Step 3: Connect Instruments to GPIB Card
- (Optional) Step 4: Configure GPIB Interface Cards
- (Optional) Step 5: Communicate with Instruments
- (Optional) Step 6: Program Your Instruments



## GPIB Quick Start

This section shows suggested steps to help you quickly get started connecting GPIB instruments to the General Purpose Interface Bus (GPIB).

**Typical GPIB Interface System** In this guide, a **GPIB interface system** is defined as a system in which GPIB instruments are connected to a GPIB interface card in a Windows PC via GPIB cables. (Refer to **Installing I/O Software** for a Windows support matrix.) This figure shows a typical GPIB interface system with a PC and two GPIB instruments connected via GPIB cables.



**Figure 24** Typical GPIB Interface System

**Getting Started** To get started, you may want to print the figures for Steps 1 - 6 to use as a guide as you set up your GPIB system.

### NOTE

You can save time connecting GPIB instruments by using the Agilent 82357A USB/GPIB Interface Converter. (Then you won't need a GPIB card.) For more information, go to [www.agilent.com/find/gpib](http://www.agilent.com/find/gpib).



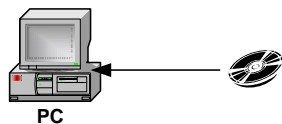
### Other Information You Can Use

- [Troubleshooting GPIB Interfaces](#) for GPIB troubleshooting tips
- [Glossary](#) for a definition of some GPIB terms
- [Contacting Agilent](#)

## 4 Connecting Instruments to GPIB

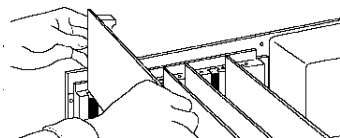
### 1 Install I/O Software on Your PC

Install Agilent IO Libraries Suite from a CD or the Web.



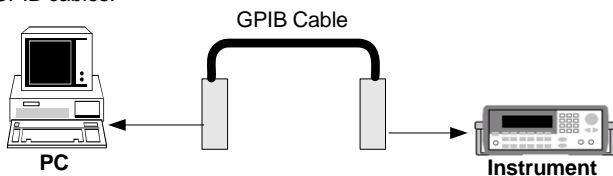
### 2 Install GPIB Cards in Your PC

As required, install one or more GPIB interface cards in your PC. (Or, connect the 82357A GPIB/USB Converter cable to a USB port on your PC.)



### 3 Connect Instruments to GPIB Card

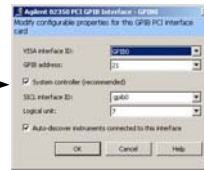
Connect your instruments to the installed GPIB interface card(s) using GPIB cables.



## 4 Configure GPIB Interface Cards (Optional)

Use **Agilent Connection Expert** to configure installed GPIB interface card parameters.

You can use the **GPIB Interface Properties** dialog box to configure GPIB interface cards installed in your PC.



## 5 Communicate With Instruments (Optional)

You can use Interactive IO to verify communication with instruments.



## 6 Program Your Instruments (Optional)

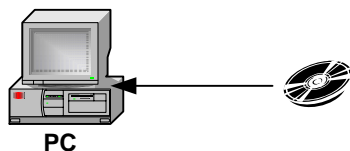
As desired, you can program instruments via the GPIB interface using Agilent VEE, Agilent T&M Toolkit 2.0, Visual Basic 6, Visual Basic.NET, or many other programming environments.

VISA: viopen (... "GPIB0::5::INSTR"...)

## Step 1: Install I/O Software on Your PC

### 1 Install I/O Software on Your PC

Install Agilent IO Libraries Suite from a CD or the Web.

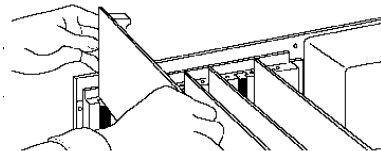


**What's in This Step?** Before you connect your instruments to GPIB, install Agilent IO Libraries Suite software on your PC. See [Installing I/O Software](#) for details. Then, go to [Step 2: Install GPIB Cards in Your PC](#).

## Step 2: Install GPIB Cards in Your PC

### 2 Install GPIB Cards in Your PC

As required, install one or more GPIB interface cards in your PC. (Or, connect the 82357A GPIB/USB Converter cable to a USB port on your PC.)



**What's in This Step?** This step shows how to install GPIB interface cards (such as an Agilent 82350 PCI GPIB Interface for Windows) in your PC.

#### NOTE

If you have not yet installed I/O software on your PC, go to [Step 1: Install I/O Software on Your PC](#) and install the software BEFORE you install GPIB Interface Cards in your PC.

#### NOTE

If you are using the 82357A GPIB/USB Converter, just plug your cable into the USB port on the PC and connect it to the GPIB port on the instrument, and continue to [Step 5: Communicate with Instruments](#).

## Steps to Install a GPIB Card in Your PC

Example steps follow to install an Agilent 82350 PCI GPIB Interface Card for Windows in your PC. Modify the steps as required if you install a different GPIB interface card in your PC. When you have installed all required GPIB Interface Cards, go to [Step 3: Connect Instruments to GPIB Card](#).

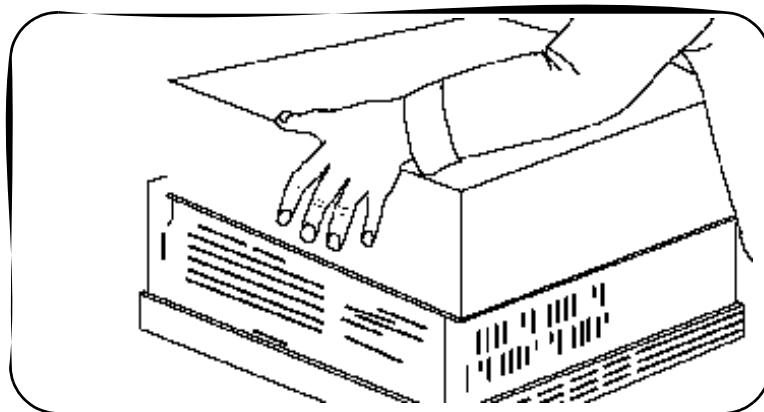
### CAUTION

To reduce the risk of damaging a GPIB Interface Card, only handle the card by the sheet metal frame or by its edges.

- 1 Record the card serial number.** Remove the card from its anti-static bag and record the serial number for future reference. Save the anti-static bag so you can protect the card if you need to remove it from the PC.

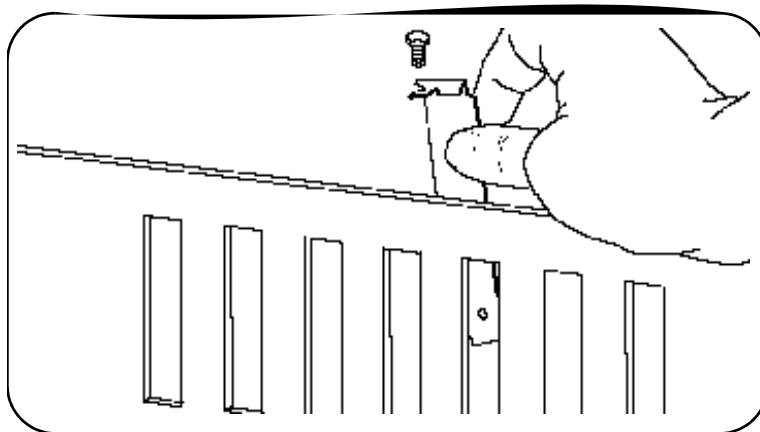
For example, the 82350 serial number is located on the white serial number label on the card. The label contains 24 characters, with the last 8 characters (plus US) being the 82350 Serial Number. Thus, if the numbers on the serial number label are 82350-66511-4224-01-42300113, since the last 8 characters are 42300113, the 82350 Serial Number is US42300113.

- 2 Remove the PC cover.** Remove power from the PC and from all of its peripherals. Then, remove the power cord from the PC. Unlock and remove the cover from the PC to allow access to the I/O slots. See your PC documentation for instructions.



**Figure 25** Removing the PC Cover

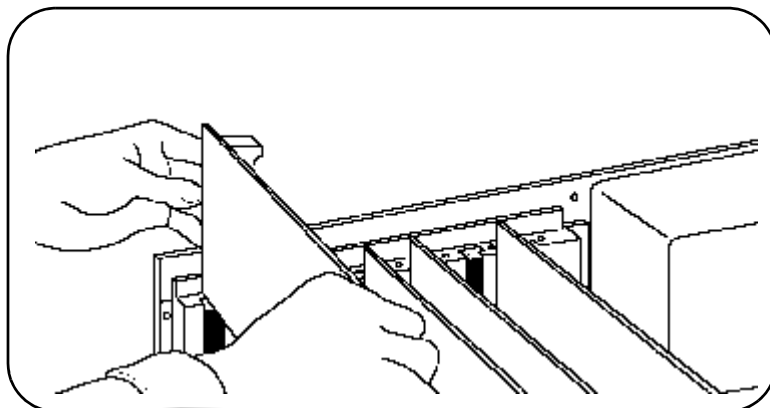
**3 Remove the cover plate.** Remove one of the PC back panel cover plates. The 82350B is a 5V PCI card and will not fit in a 3.3V PCI slot or in an EISA or ISA slot. Choose a 5V PCI slot that will give adequate clearance for the GPIB connector. If you install more than one 82350, you may want to install the cards so there is at least one empty slot between every two 82350s.



**Figure 26** Removing a Cover Plate

- 4 Install the 82350.** Insert the edge connector: Insert the 82350 Interface Card edge connector into the PCI expansion slot connector of the PC. Make sure the interface is fully seated by pushing firmly on the top edge of the card with the palm of your hand. The GPIB connector should extend through the back panel opening to allow GPIB cable connection.

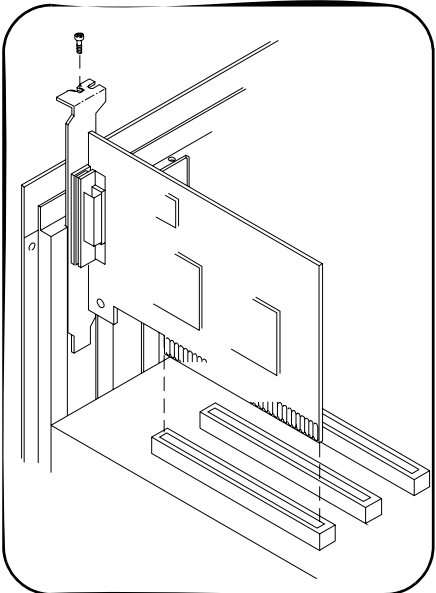
When inserting the 82350, be sure to hold the card by its edges. Also, be careful with the metal faceplate around the GPIB connector as the faceplate can be bent.



**Figure 27** Installing the GPIB Card

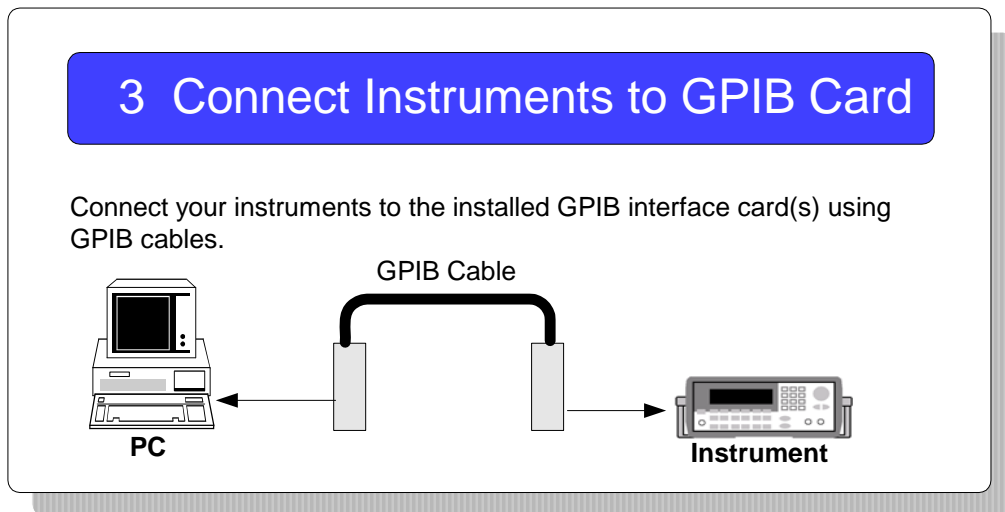
- 5 Replace the Cover Plate Screw.** This will hold the 82350 in place. Save the blank cover plate for use if the 82350 is later removed. Replace the PC cover(s) as described in your PC documentation.





**Figure 28** Replacing the Cover Plate Screw

## Step 3: Connect Instruments to GPIB Card



**What's in This Step?** This step gives guidelines to connect GPIB instruments to a GPIB Interface Card (such as an Agilent 82350) installed in your PC by using GPIB cables. When you have made the connections for your system, go to [Step 4: Configure GPIB Interface Cards \(Optional\)](#).

### Steps to Connect Instruments to GPIB Cards

- 1 Review Connection Guidelines.** The recommended method for connecting a GPIB system is linear with the system controller (PC) at one end of the system. However, a GPIB system can also be connected together in a star, linear, or a combination configuration as long as the total number

of devices on the system is  $\leq 15$  and these guidelines are followed:

- To minimize stress on connector mountings, no more than three cable connector blocks should be stacked on top of one another. The GPIB connector screws should be finger-tightened only.
- Minimize cable length as much as possible. All system devices must have tri-state drivers and must be powered on. Turning devices on or off while a system is running may cause faulty operation.
- For operation with data transfer rates  $< 500$  Kbytes/sec., the total allowed length of all GPIB cables is  $\leq 2$  meters times the number of devices connected together, up to a maximum of 20 meters.
- For operation with data transfer rates  $> 500$  Kbytes/sec., the total allowed length of all GPIB cables is  $\leq 1$  meter times the number of devices connected together, up to a maximum of 15 meters.
- The cable length between adjacent devices is not critical as long as the system meets the overall restriction. GPIB bus extenders are available that allow operation over much greater distances.

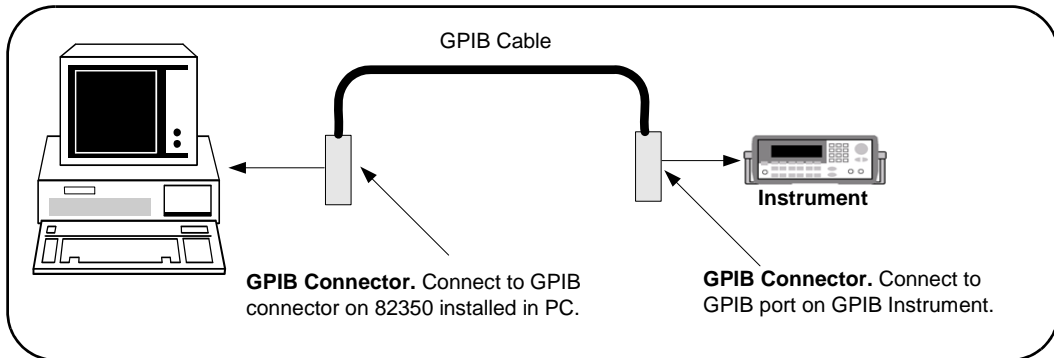
**NOTE**

For assistance selecting and buying cables, go to <http://www.agilent.com/find/gpib> and click **GPIB Cables and GPIB Adapters**.

- 
- 2 Connect GPIB Cables to the GPIB Interface Card.** Connect a separate GPIB cable to each installed GPIB Interface Card. Tighten the GPIB connector screws finger-tight only. (The screwdriver slots are for removal purposes only.) Two example connections follow to connect a single GPIB instrument or to connect multiple GPIB instruments.

### Example: Connecting a Single GPIB Instrument

This figure shows connections from a single GPIB instrument to the GPIB connector of an Agilent 82350 GPIB Interface Card installed in your PC.



**Figure 29** Connecting a Single GPIB Instrument

You may want to record the primary GPIB address of the attached instrument for future programming use. Or you can use Connection Expert to create programming aliases that are meaningful to you. After making the connections, reconnect the PC power cord and apply power to the PC and to attached peripherals/instruments.

### CAUTION

To avoid damage to the connectors, only finger-tighten the connectors.

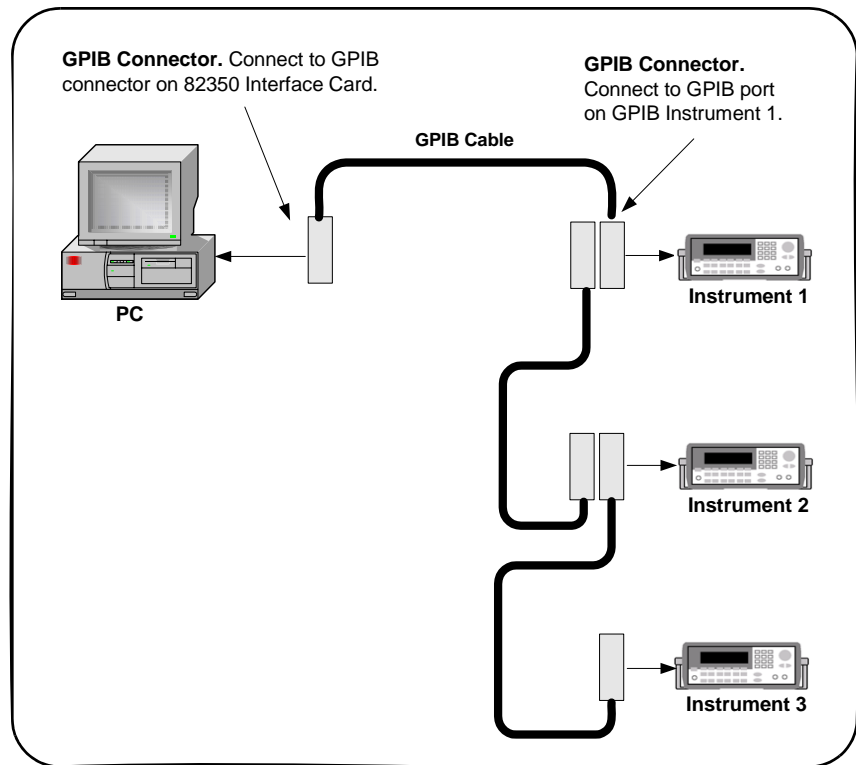
### Example: Connecting Multiple GPIB Instruments

This figure shows one way to connect three GPIB instruments to an Agilent 82350 GPIB interface card. You may want to record the primary GPIB address of each attached instrument for future programming use; or you can use Connection Expert to create programming aliases that

are meaningful to you. After making the connections, reconnect the PC power cord and apply power to the PC and attached peripherals/instruments.

**CAUTION**

*To avoid damage to the connectors, only finger-tighten the connectors.*



**Figure 30** Connecting Three GPIB Instruments

## Step 4: Configure GPIB Interface Cards

### 4 Configure GPIB Interface Cards (Optional)

Use **Agilent Connection Expert** to configure installed GPIB interface card parameters.

You can use the **GPIB Interface Properties** dialog box to configure GPIB interface cards installed in your PC.



**What's in This Step?** This step shows suggested actions to use Connection Expert to configure a Windows operating system for a PC that has an Agilent 82350 GPIB Interface Card (or equivalent) installed. The default configuration is done automatically. Click **Refresh All** to update it at any time. This section shows how to verify or change configuration parameters.

### Steps to Configure GPIB Interface Cards

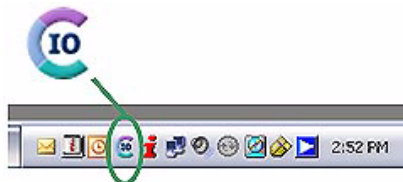
- 1 **Apply Power.** Apply power to the PC and to the installed GPIB instruments. As Windows starts again, a **Found New Hardware Wizard** may start. This figure shows a typical

Windows 2000 display. The display may be different for other operating systems.



**Figure 31** Found New Hardware Wizard Example

- 2 Install Configuration Files.** Click **Next>** to accept the default suggestions. (Make sure that you have installed Agilent IO Libraries Suite first. You will not need a CD.) Click **Finish** to complete the installation.
- 3 Open Connection Expert.** Click the **Agilent IO Control** icon,



and then click **Agilent Connection Expert**. When the main screen appears, you will see a map of the system connections in the **Instrument I/O on this PC** pane (also called “explorer pane”). *If you see your interface and instruments in the explorer pane, you are ready to go!*

## 4 Connecting Instruments to GPIB

### NOTE

For a description of Connection Expert and the Agilent IO Libraries Suite, see the *Agilent IO Libraries Suite Getting Started Guide*. This guide is on the *Automation-Ready CD with Agilent IO Libraries Suite* and is available from the IO Control **Documentation** menu.

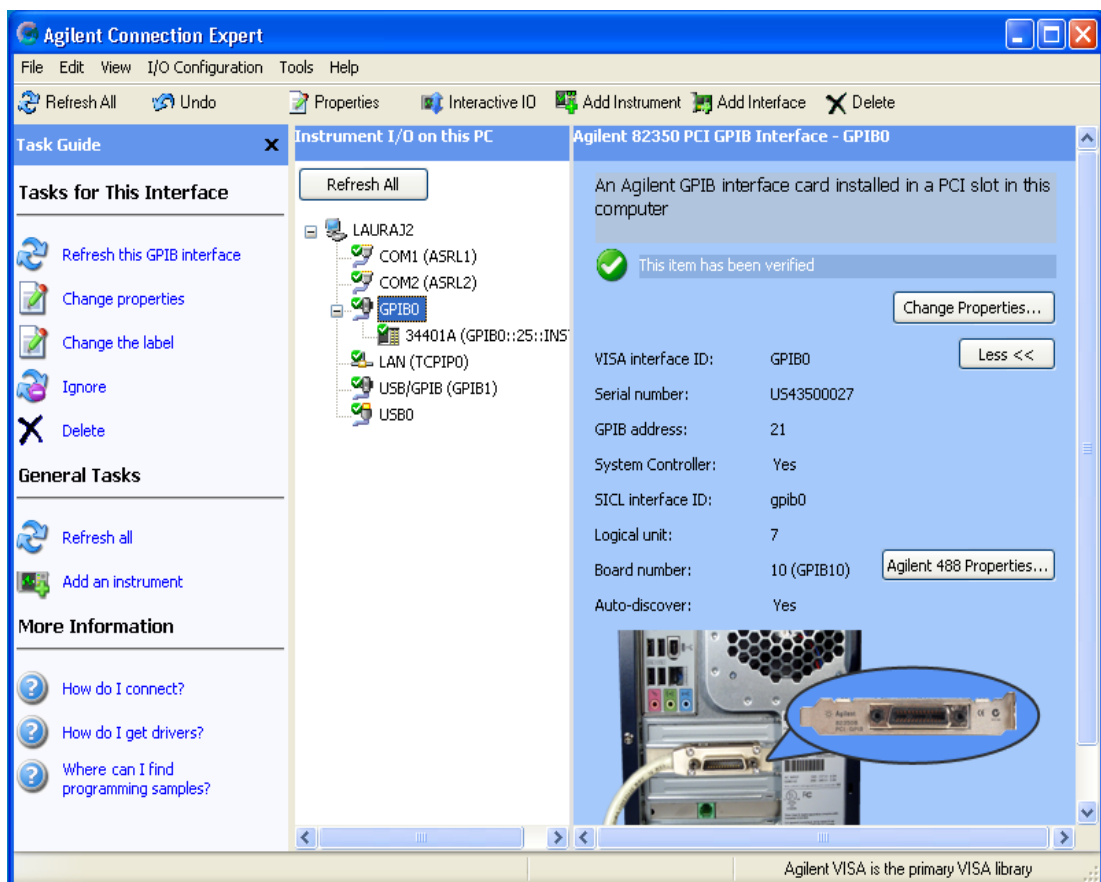
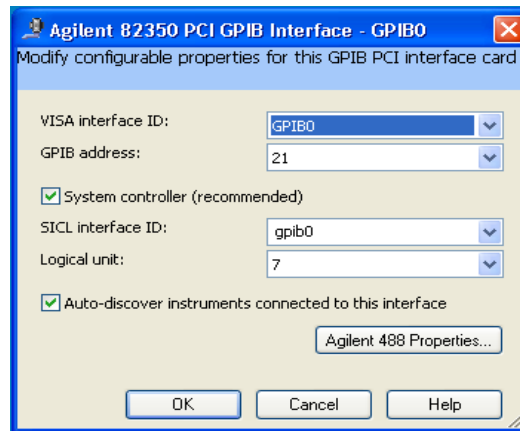


Figure 32 Agilent Connection Expert



When you select a particular instrument in the explorer pane, you see the parameters of the instrument in the properties pane on the right – such as the product number, serial number, VISA address, GPIB address, and firmware revisions. The green check mark icon at the top signifies that a listener check and IDN query were performed automatically.

- 4 **Configure GPIB Card Parameters.** With the GPIB interface selected in the explorer pane, click **Change Properties...**



**Figure 33** The 82350 Configuration Dialog Box

Set the GPIB interface properties as required. Also, verify that this is the System Controller for the GPIB to which it is attached (this is the typical operating mode). (See the [System Controller](#) discussion below.) If you plan to program your instruments using the Agilent 488 API, or to run NI-488.2-compatible programs with Agilent interface hardware (such as GPIB cards), click the button labeled **Agilent 488 Properties...** to enable the Agilent 488 library (if it is not already enabled), to set the board number, or to change other properties as needed. Details of each interface parameter are available in this dialog box's online Help.

**5 Change/Accept the Configuration Values.** When the configuration values displayed are acceptable to you, click the **OK** button.

**Repeat Steps for Other Interfaces.** If you have installed more than one GPIB interface in your test system, repeat these steps for the remaining interfaces. Then go to [Step 5: Communicate with Instruments \(Optional\)](#).

### System Controller

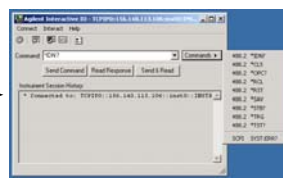
The System Controller setting determines whether this GPIB interface controls which bus devices talk and which bus devices listen. If several devices exist on a bus, be sure each has a unique GPIB bus address and only one device is the system controller (it is usually the device installed in the computer). Each GPIB interface has its own independent bus. Thus, each interface may be a System Controller as long as it is not chained together with other GPIB interfaces. However, two or more System Controllers on the same bus will cause the bus to be inoperative.

Note that the 82357A USB/GPIB Interface Converter will always be System Controller for its bus.

## Step 5: Communicate with Instruments

### 5 Communicate With Instruments (Optional)

You can use Interactive IO to verify communication with instruments.



**What's in This Step?** If your instruments show up in the explorer pane as verified, communication has been established. This section shows how to further verify instruments and connections using Interactive IO. When you have verified communication with your instruments (as desired), go to [Step 6: Program Your Instruments \(Optional\)](#).

## Communicating Using Interactive IO

Interactive IO is a utility within Agilent IO Libraries Suite that communicates with instruments. It can be used to send and receive strings to instruments, which support formatted I/O.

### NOTE

**Timeout Errors:** If you are getting timeout errors when sending commands and waiting for results, you can lengthen the timeout in the **Interact > Options** menu, which is set at 5000 ms by default.

### NOTE

**When to Use VISA Assistant:** VISA Assistant is still installed as part of the Agilent IO Libraries Suite, but, for most users, it has been replaced by the Interactive IO utility. Interactive IO provides a simpler, more intuitive way to send commands to instruments and read the results.

Some capabilities of VISA Assistant are not yet available in the Interactive IO utility. These include:

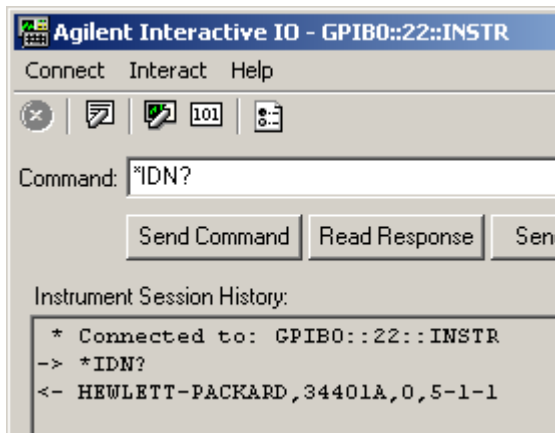
- Memory I/O for VXI and GPIB-VXI interfaces
- Reading and writing of VISA attributes
- Configuration of *VXIplug&play* drivers

If you need these capabilities, you can start the VISA Assistant utility by clicking the **Agilent IO Control > VISA Assistant**, or by clicking **Tools > VISA Assistant** on the Connection Expert menu bar.

### Example using Interactive IO to control an Agilent 34401A

- 1 Select an instrument in the Connection Expert explorer pane.
- 2 Click the Interactive IO icon below the menu bar.
- 3 Open the **Commands** menu and select a command. The example below shows a **\*IDN?** command has been chosen. (See the table below for commands and meanings. You can also type an instrument-specific command.)
- 4 Click **Send & Read** to send the selected command and have the instrument return a result. Your selections are displayed in the **Instrument Session History** area, as shown in

this example below. (You can also click the **Send Command** and **Read Response** buttons separately for additional control over these actions.)



**Figure 34** Interactive IO Utility with Instrument Session History

488.2 Commands	Meaning
*IDN?	The Identification query returns manufacturer, model, serial number, and firmware level or equivalent.
*CLS	The Clear Status command clears status data structures, and forces the device to the Operation Complete query idle state.
*OPC?	The Operation Complete query places an ASCII character "1" into the device's Output Queue when all pending selected device operations have been finished.
*RCL	The Recall command restores the current settings of a device from a copy stored in local memory.
*RST	The Reset command performs a device reset, which sets the device-specific functions to a known state that is independent of the past-use history of the device.
*SAV	The Save command stores the current settings of the device in local memory.

## 4 Connecting Instruments to GPIB

<b>*STB?</b>	The Read Status Byte query allows the programmer to read the status byte and master Summary Status bit.
<b>*TRG</b>	The Trigger command signals the transition of the Device Trigger (DT) function to the Device Trigger Active State (DTAS).
<b>*TST?</b>	The Self-test query returns zero to indicate the test completed with no errors. A return value not equal to zero indicates the test is not completed or completed with errors.
<b>SCPI Commands</b>	
<b>SYST:ERR?</b>	The queue query message is a request for the next entry from the instrument's error/event queue. Items in this queue contain an integer in the range [-32768, 32767] denoting an error/event number and associated descriptive text. Negative numbers are reserved by the SCPI standard; positive numbers are instrument-dependent. A value of zero indicates that no error or event has occurred.

## Step 6: Program Your Instruments

### 6 Program Your Instruments (Optional)

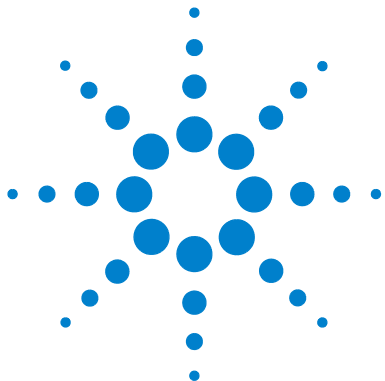
As desired, you can program instruments via the GPIB interface using Agilent VEE, Agilent T&M Toolkit 2.0, Visual Basic 6, Visual Basic.NET, or many other programming environments.

```
VISA: viopen (... "GPIB0::5::INSTR" ...)
```

**What's in This Step?** *This is an optional step.* After you have set up your GPIB system, as desired you can program GPIB instruments from your PC via the GPIB interface. See [Programming Your Instruments](#) for guidelines and examples.







## 5 Programming Your Instruments

This chapter includes:

- Programming Overview
- Addressing Instruments
- Sample Programs



## Programming Overview

This section provides an overview of programming instruments via LAN, USB, and GPIB interfaces. Sample programs are included to demonstrate generating a simple sine wave on an Agilent 33220A 20 MHz Function/Arbitrary Waveform Generator using Standard Commands for Programmable Instruments (SCPI).

The sample programs are written in Microsoft Visual Basic 6.0, Visual C++, and C# using Agilent VISA COM and VISA I/O.

### NOTE

The programs in this guide are protected by copyright.

Copyright © 2003- 2005 Agilent Technologies, Inc.

You have a royalty-free right to use, modify, reproduce and distribute the programs listed in this guide (and/or any modified version) in any way you find useful, provided that you agree that Agilent has no warranty, obligations, or liability for any program content.

Agilent Technologies provides programming examples for illustration only. All example programs assume you are familiar with the programming language being demonstrated and the tools used to create and debug procedures. Agilent support engineers can help explain the functionality of Agilent software components and associated commands, but they will not modify these samples to provide added functionality or construct procedures to meet your specific needs.

### NOTE

If you plan to program your instruments using the Agilent 488 API, or to run NI-488.2-compatible programs with Agilent interface hardware (such as GPIB cards), you may need to explicitly enable the Agilent 488 library. To do this, in Agilent Connection Expert click **Tools > Agilent 488...** and select the check box labeled **Enable Agilent GPIB cards for 488 programs**.

## Obtaining Instrument Drivers

Selected combinations of program environments and I/O are included in the sample programs in this guide. You will need to acquire the programming environment independently of your I/O and instrument software.

In many cases, you receive two CDs with your purchase of an Agilent hardware or software product: one is the *Automation-Ready CD with Agilent IO Libraries Suite* and the other is the product-specific CD.

VISA, VISA COM, SICL, and Agilent 488 I/O libraries are available as part of the Agilent IO Libraries Suite, on the *Automation-Ready CD*. The product CD typically contains drivers and/or sample programs that increase the ease of use of programming or support your instrument's measurement applications.

IVI-COM, *VXIplug&play*, and other types of instrument drivers for the Agilent 33220A and other instruments may be obtained at <http://www.agilent.com/find/drivers>. To download a driver, you will be asked to register on the Agilent Developer Network (ADN) free of charge.

The types of drivers on ADN include:

- IVI-COM drivers and components
- *VXIplug&play* drivers
- IntuiLink connectivity software (Agilent IntuiLink Toolbars, which provide access to instruments in Microsoft Excel and Microsoft Word)
- VEE panel drivers
- D-SCPI drivers
- SCPI drivers

**NOTE**

USB ports ONLY work properly on Windows XP when the latest Microsoft USB drivers from Windows XP Service Pack 1 or 2 have been properly installed. In addition, if using USB on Windows 2000, Agilent recommends you install the latest Service Pack for Windows 2000.

Upgrading your Windows XP PC to Service Pack 1 will probably not update the USB drivers automatically. See [www.agilent.com/find/connectivity](http://www.agilent.com/find/connectivity) to verify if correct USB drivers are installed and to install correct drivers as required.

---

## Using Agilent IO Libraries Suite Sample Code

The *Automation-Ready CD with Agilent IO Libraries Suite* provides easy instructions for connecting, configuring, and verifying your instruments with IO Libraries Suite. Go to [Chapter 6](#), “Installing I/O Software”, for a more detailed discussion.

**NOTE**

To obtain the Agilent IO Libraries Suite, go to [www.agilent.com/find/iolib](http://www.agilent.com/find/iolib). If you own any Agilent instrumentation or software products, you can download the IO Libraries Suite free of charge.

---

When you're ready to start programming, IO Libraries Suite provides a number of sample programs for a variety of instruments. To access these programs from the Windows Start menu, click **Start > Programs > Agilent IO Libraries Suite > Program Samples**. Sample programs are provided in various languages, including Microsoft C, C#, Excel, Visual Basic 6.0, and Visual Basic .NET.

To use a sample program:

- 1 Select a program for your instrument or a similar instrument. Open the program in your programming environment.
- 2 Replace the instrument addressing in the program with your instrument's VISA address or VISA alias.

- 3 If necessary, replace the command strings in the program with commands for your instrument.
- 4 Run the program and debug, if necessary.

## Using VISA COM in Other Visual Basic Projects

Some of the sample programs in this chapter use Agilent VISA COM. To use VISA COM in another Visual Basic project, perform these steps:

- 1 **Set the Reference.** Set the reference to include the libraries in the **Project > References** menu:
  - “VISA COM 3.0 Type Library” corresponds to GlobMgr.dll
  - “VISA COM 488.2 Formatted I/O 1.0” corresponds to BasicFormattedIO.dll
- 2 **Create Formatted I/O Reference.** Create the formatted I/O reference with a statement such as:  
`Dim Fgen As VisaComLib.FormattedIO488`
- 3 **Create the New Object.** Use  
`Set Fgen = New VisaComLib.FormattedIO488`  
to create the formatted I/O object.

For additional information on VISA COM usage, see the *VISA COM Help* (click the IO Control, then click **Documentation > VISA COM Help**).

## Addressing Instruments

This section gives guidelines to address instruments, including:

- [Addressing Instruments via LAN](#)
- [Addressing Instruments via USB](#)
- [Addressing Instruments via GPIB](#)

### NOTE

This chapter discusses addressing for VISA and VISA COM. This form of addressing is also used with IVI-COM and VXIplug&play instrument drivers. This chapter also includes some discussion of SICL addressing.

For a full discussion of SICL addressing, see the *Agilent SICL User's Guide*. For a discussion of addressing for the Agilent 488 API, see the *Agilent 488 Online Help*. Both of these documents are available from the IO Control's **Documentation** menu.

---

To simplify the addressing in your programs, you can use VISA aliases. A VISA alias is a name of your choosing, which you assign to a device and use in your programs. Once assigned, the alias is a synonym for the device's VISA address, so you can use it to open a VISA session (using the **viOpen** function) and to get resource information (using **viParseRsrc** or **viParseRsrcEx**).

Agilent Connection Expert allows you to define VISA aliases for the devices in your test system.

**Why use VISA aliases?** Using VISA aliases in your programs, rather than VISA addresses, provides two significant advantages:

- **Portability.** If you program using aliases, you can run your program on a new test system, whose instruments are at different addresses, simply by using Connection Expert to create the same aliases on the new system as you created on your development system. Similarly, you can move or replace instruments without changing or recompiling test code, simply by changing the alias definitions in Connection Expert.
- **Readability.** Your programs will be much easier to read and understand if, for example, your multimeter is called “myDMM” instead of “GPIB2::14::8::INSTR”. This is particularly important in the case of USB instruments, whose VISA addresses are typically long and cumbersome, containing the instrument's serial number among other information.

**Default aliases for USB devices.** Connection Expert gives an alias to each instrument you connect via USB, because VISA addresses for USB devices are long and cumbersome. Connection Expert assigns aliases **UsbDevice1**, **UsbDevice2**, etc.; you can change these names by selecting the alias in the explorer view and clicking the **Change...** button.

**Using aliases with SICL and with Agilent 488.** When you create a VISA alias, Connection Expert automatically creates SICL and Agilent 488 aliases of the same name. You can use a SICL alias in place of a SICL address in the **iopen** function call, or an Agilent 488 alias in the **ibfind** function call, with the same advantages described above.

## Addressing Instruments via LAN

To address instruments via the LAN, you must first check the default configuration of the LAN interface using Agilent Connection Expert. A summary of the applicable LAN networking protocols follows. See “[LAN Interface Overview](#)” on page 140 for a description of LAN hardware, software, and network protocols.

## LAN Networking Protocols

There are two LAN networking protocols provided with the Agilent IO Libraries Suite software. You can use one or both of these protocols when configuring your systems to use VISA and SICL over LAN.

- **SICL-LAN Protocol** is a networking protocol developed by Agilent that is compatible with all VISA LAN products. This LAN networking protocol is the default choice in the Agilent IO Libraries Suite configuration when configuring the LAN interface. The SICL-LAN protocol on Windows supports VISA operations over LAN to GPIB interfaces.
- **VXI-11 (TCP/IP Instrument Protocol)** is a networking protocol developed by the VXIbus Consortium based on the SICL-LAN Protocol that permits interoperability of LAN software from different vendors who meet the VXIbus Consortium standards.

When using either of these networking protocols, the LAN software uses the TCP/IP protocol suite to pass messages between the LAN device and the server. The server accepts device I/O requests over the network from the device and then proceeds to execute them.

By default, the LAN supports both protocols by automatically detecting the protocol the server is using. When a VISA **viOpen** or SICL **iopen** call is performed, the LAN interface driver first tries to connect using the SICL-LAN protocol. If that fails, the driver will try to connect using the VXI-11 protocol.

If you want to control the protocol used, you can configure more than one LAN interface using Connection Expert and set each interface to a different protocol. The protocol used will then depend on the interface you are connecting through.



In VISA, the protocol used is determined by the configuration settings and cannot be changed programmatically. In SICL, you can override the configuration settings programmatically by specifying the protocol in the `iopen` string.

### Example: Using Network Protocols in SICL

Some examples using VXI-11 and SICL-LAN protocol follow. Note that SICL names are case-sensitive.

- `iopen("lan[machineName]:gpib0,1")` will use the protocol type configured in Connection Expert. If **Automatic** is configured, SICL-LAN protocol will be attempted. If SICL-LAN is not supported, VXI-11 protocol will be used.
- `iopen("lan;auto[machineName]:gpib0,1")` will automatically select the protocol (SICL-LAN if available and VXI-11 otherwise.)
- `iopen("lan;sicl-lan[machineName]:gpib0,1")` will use SICL-LAN protocol.
- `iopen("lan;vxi-11[machineName]:gpib0,1")` will use VXI-11 protocol.
- The Agilent IO Libraries Suite also supports TCP/IP socket reads and writes. To open a socket session, use `iopen("lan,socketNbr[machineName]")`. For example, `iopen("lan,7777[machineName]")` will open a socket connection for socket number 7777 on '`machineName`'.

### Configuring LAN Interfaces

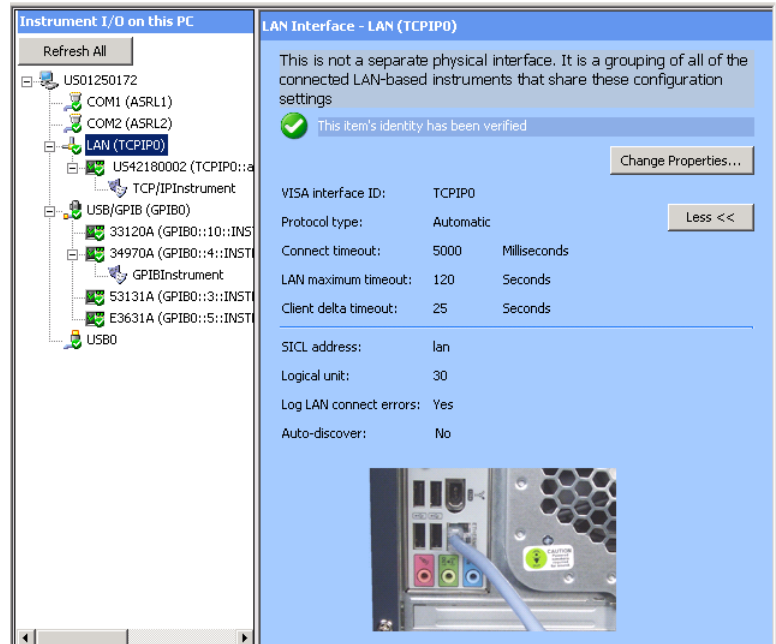
When you have configured LAN interfaces, you can then use the interface name specified during configuration in a VISA `viOpen` call in your program. A summary of the steps to configure a LAN interface follows.

#### NOTE

A single LAN interface can be used by any number of devices. The main reason for having more than one LAN interface configured is that you want to use different protocols (such as SICL-LAN and VXI-11) with different devices. You can also specify different timeout durations for different LAN interfaces.

- 1 To check the configuration of your LAN interfaces using Connection Expert, click the Agilent IO Control icon (blue **IO** icon in the Windows notification area) and click **Agilent Connection Expert**. When the main screen appears, highlight the LAN interface in the explorer (center) pane – **LAN (TCPIPO)** in this example – and check the configuration in

the **LAN Interface** properties pane on the right. To alter any default properties, click **Change Properties....**



**Figure 35** LAN Interface Properties Pane (on right)

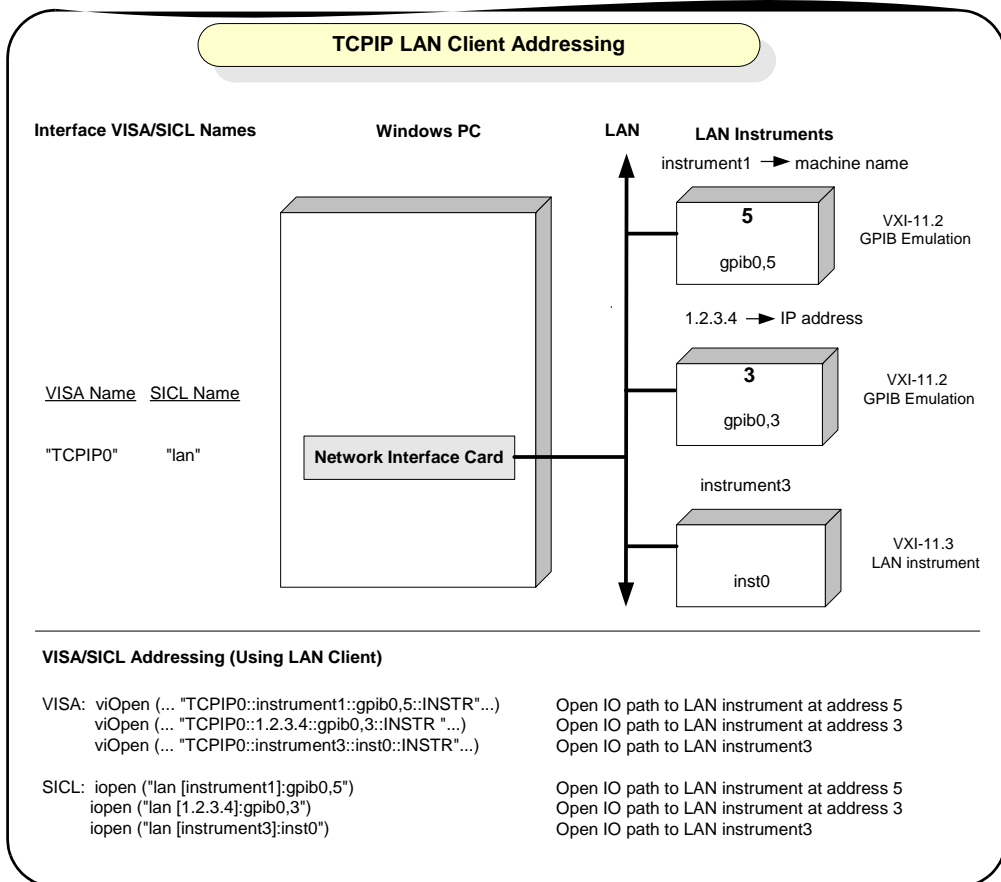
- When the **LAN Interface** dialog box appears, you can change the interface's properties, such as the VISA interface ID, logical unit number, timeout durations, as required. Here you can select the protocol type (Auto, VXI-11, or SICL-LAN) and choose whether to log LAN connection errors. When you finish setting properties, click **OK**.

### Example: Addressing LAN Instruments Using VISA/SICL

The LAN interface system in this figure consists of a Windows PC with a LAN (NIC) card and three LAN instruments. **instrument1** and **instrument2** are VXI-11.2 (GPIB Emulation) instruments, and **instrument3** is a VXI-11.3 LAN instrument.

For this system, Connection Expert has been used to assign the LAN card a VISA interface ID of “TCPIP0” and a SICL interface ID of “lan”. For the addressing examples, **instrument1** has been addressed by its machine name, **instrument2** by its IP address, and **instrument3** by its remote instrument name (*inst0*).

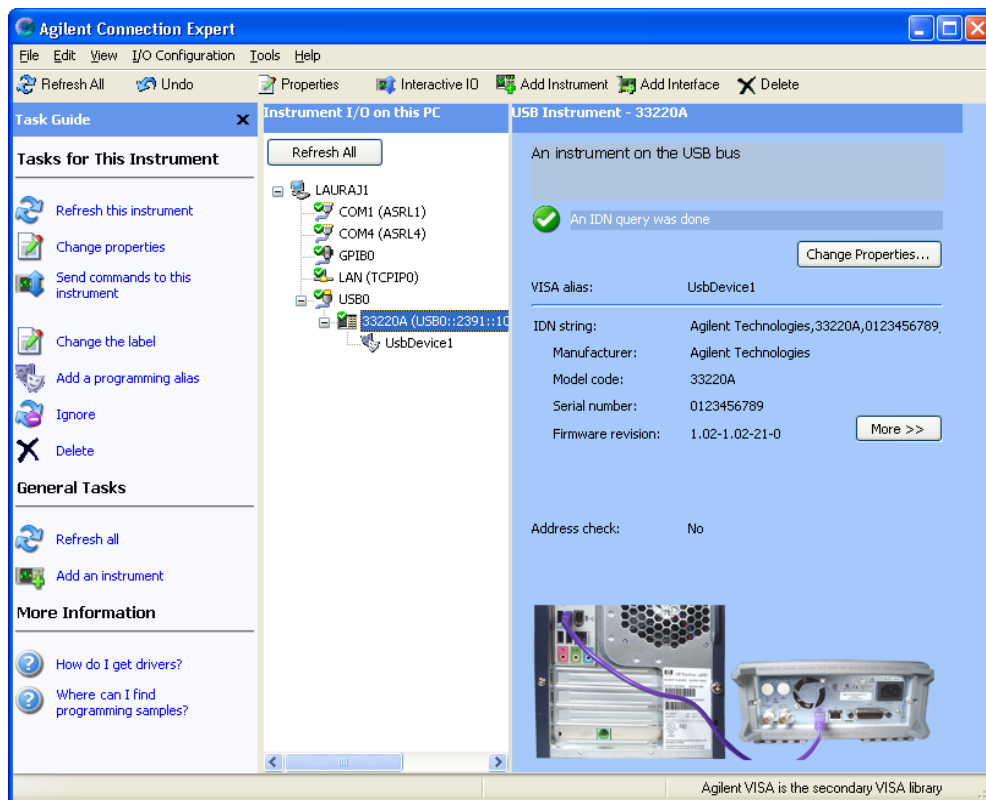
Since unique names have been assigned by Connection Expert, you can now use the VISA **viOpen** command or the SICL **iopen** command to open the I/O paths to the GPIB instruments as shown in the figure.



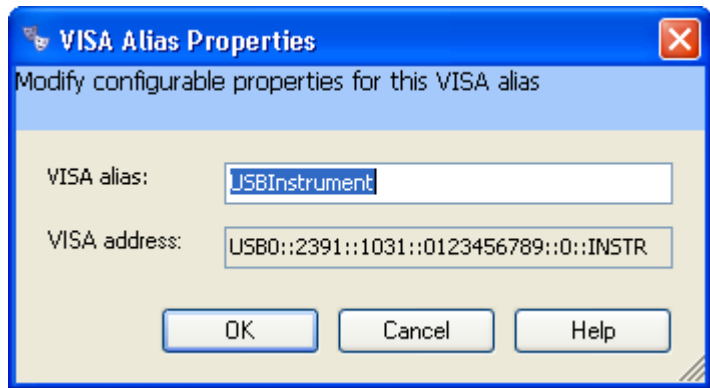
## Addressing Instruments via USB

As desired, you can use a VISA alias to address instruments via USB. The VISA alias associates a symbolic name with a specific instrument. You can use it as a VISA *rsrcName* or SICL *address* instead of using the full address (or resource descriptor), which may include the instrument's vendor ID, product ID, serial number, and interface ID, etc. See the following steps to set, add, delete or change a VISA alias.

- 1 **Start Connection Expert.** Click the blue **IO** icon (on the Windows taskbar) and select **Agilent Connection Expert** to display Connection Expert's main window.



- 1 Connection Expert automatically configures a USB interface, discovers any USB devices, and assigns default VISA alias names, as shown in the figure above.
- 2 **Display the Changing VISA Alias Properties dialog box.** Highlight the USB alias name in the explorer (center) pane. Then click **Change Properties...** in the **VISA Alias Properties** pane on the right to display the **VISA Alias** dialog box. Change the default VISA alias name to something meaningful.



**Figure 36** Changing VISA Alias Properties Dialog Box

The VISA address of a USB instrument is based on the instrument's vendor ID, product ID, serial number, and VISA interface ID.

- 3 Click **OK**.

## Addressing Instruments via GPIB

To address instruments via a GPIB interface (such as an 82350 GPIB interface or 82357 USB/GPIB interface converter), first check the default configuration of the interface using Connection Expert.

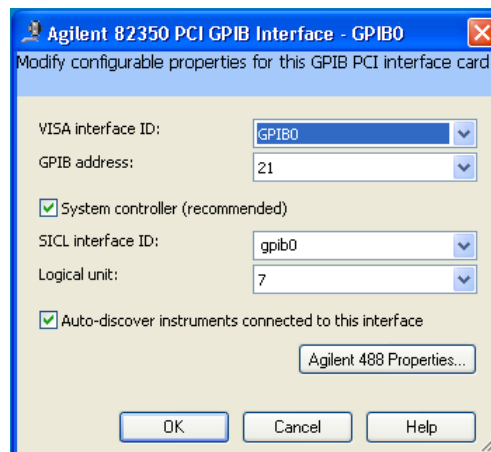
## Steps to Configure a GPIB Interface

These steps are written for the 82350 PCI GPIB interface card. Modify the steps as required if you use another type of supported GPIB interface. The *Agilent IO Libraries Suite Online Help* includes a description of configurable parameters for each supported interface type.

### NOTE

If you plan to program your instruments using the Agilent 488 API, or to run NI-488.2-compatible programs with Agilent interface hardware (such as GPIB cards), you may need to enable the Agilent 488 library. To do this, click **Tools > Agilent 488...**, then select the check box labeled **Enable Agilent GPIB cards for 488 programs**.

**Configure GPIB Card Parameters.** With the GPIB interface selected in the instrument explorer pane, click **Change Properties...** in the **Agilent 82350 PCI GPIB Interface** properties pane on the right. The PCI GPIB configuration dialog box appears, as shown below.



**Figure 37** The 82350 Configuration Dialog Box

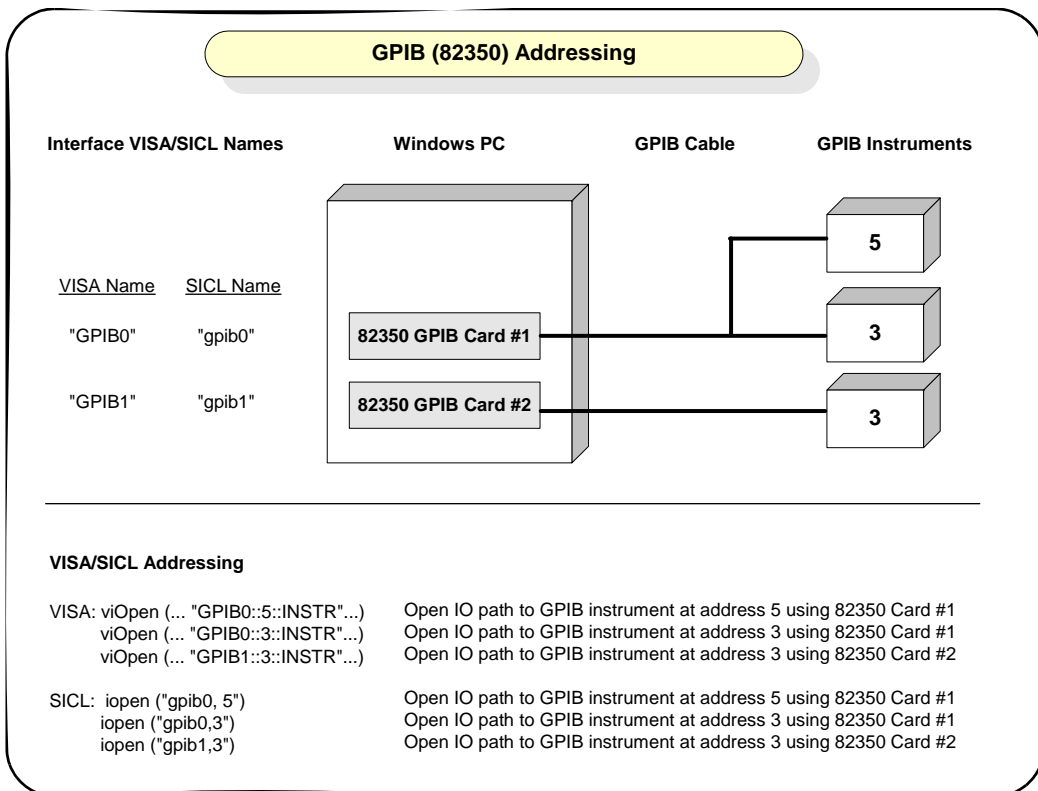
## 5 Programming Your Instruments

- 1 Set the GPIB properties as required, referring to the online Help for specific information on each property as needed. Also, verify that this is the System Controller for the GPIB to which it is attached (this is the typical operating mode). (See “System Controller” on page 90.) Then, click the **OK** button.
- 2 **Change/Accept the Configuration Values.** When the configuration values displayed are acceptable to you, click the **OK** button.

**Repeat Steps for Other Interfaces.** If you have installed more than one GPIB interface in your test system, repeat the above steps for the remaining interfaces.

### Example: Addressing GPIB Instruments Using VISA/SICL

This figure shows example VISA and SICL addressing for GPIB instruments connected to a PC via a GPIB interface.





The GPIB interface system consists of a Windows PC with two 82350 GPIB cards connected to three GPIB instruments via GPIB cables. For this system, Connection Expert has been used to assign GPIB card #1 a VISA interface ID of “GPIB0” and a SICL interface ID of “gpib0”.

Connection Expert has also been used to assign GPIB card #2 a VISA interface ID of “GPIB1” and a SICL interface ID of “gpib1”. With these names assigned to the interfaces, the VISA/SICL addressing is as shown in the figure.

Since unique names have been assigned by Connection Expert, you can use the VISA **viOpen** command to open the I/O paths to the GPIB instruments as shown in this figure. Or, you can use the SICL **iopen** command to open the I/O paths shown.

## Sample Programs

Sample programs follow to demonstrate sine waveform generation for an Agilent 33220A 20 Mhz Function/Arbitrary Waveform Generator. The sample programs are essentially repeated to show the same functionality with different programming environments and I/O. Agilent IO Libraries Suite provides a number of additional sample programs for a variety of instruments. To access other sample programs from the Windows Start menu, click **Start > Programs > Agilent IO Libraries Suite > Programming Samples**.

The example programs in this section are:

- [Example: Simple Sine Waveform \(Visual Basic and VISA COM\)](#)
- [Example: Simple Sine Waveform \(C# and VISA COM\)](#)
- [Example: Simple Sine Waveform \(Visual C++ and VISA\)](#)

### Example: Simple Sine Waveform (Visual Basic and VISA COM)

This program selects the function as “sine,” and then sets the frequency, amplitude and offset of the waveform for an Agilent 33220A. The program is written in Visual Basic 6.0 and uses VISA COM.

```
Private Sub cmdSimpleSine_Click()

Dim io_mgr As VisaComLib.ResourceManager
Dim Fgen As VisaComLib.FormattedIO488

Set io_mgr = New AgilentRMLib.SRMClS
Set Fgen = New VisaComLib.FormattedIO488
Set Fgen.IO = io_mgr.Open(txtIO.Text)

On Error GoTo MyError
```

```
' This program sets up a waveform by selecting the waveshape
' and adjusting the frequency, amplitude, and offset.
```

```
With Fgen
```

```
.WriteString "*RST"           ' Reset the function generator
.IO.Clear                     ' Clear errors and status registers
```

```
.WriteString "FUNCTION SINusoid" ' Select waveshape
' Other options are SQUARE, RAMP, PULSE, NOISE, DC, and USER
.WriteString "OUTPUT:LOAD 50"    ' Set the load impedance in Ohms
' (50 Ohms default)
```

```
' May also be INFINITY, as when using oscilloscope or DMM
```

```
.WriteString "FREQUENCY 2500"    ' Set the frequency.
.WriteString "VOLTage 1.2"       ' Set the amplitude in Vpp.
' Also see VOLTage:UNIT
.WriteString "VOLTage:OFFSET 0.4" ' Set the offset in Volts
```

```
' Voltage may also be set as VOLTage:HIGH and VOLTage:LOW for
' low level and high level
```

```
.WriteString "OUTPUT ON"        ' Turn on the instrument output
```

```
End With
```

```
Exit Sub
```

```
MyError:
```

```
txtError = Err.Description & vbCrLf
```

```
Resume Next
```

```
End Sub
```

**Example: Simple Sine Waveform (C# and VISA COM)**

This program selects the function as “sine,” and then sets the frequency, amplitude and offset of the waveform for an Agilent 33220A. The program is written in C# and VISA COM.

**NOTE**

“VISA COM 3.0 Type Library” corresponds to GlobMgr.dll.

Once you have selected C# in Visual Studio, click **Project > Add Reference...** . Select the **COM** tab. Scroll down the list of components presented, and select **VISA COM 3.0 Type Library**.

```
using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;

namespace WindowsApplication1
{
    /// <summary>
    /// Summary description for Form1.
    /// </summary>
    public class Form1 : System.Windows.Forms.Form
    {
        private System.Windows.Forms.Button button1;
        private System.Windows.Forms.TextBox USBInstrument;
        /// <summary>
        /// Required designer variable. "USBInstrument" is a VISA alias
        /// for the 33220A Function/Arbitrary Waveform Generator.
        /// </summary>
        private System.ComponentModel.Container components = null;
        public Form1()
        {
            //
            // Required for Windows Form Designer support
            //
            InitializeComponent();
        }
    }
}
```

```

    //
    // TODO: Add any constructor code
    // after InitializeComponent call
    //
}

/// <summary>
/// Clean up any resources being used.
/// </summary>
protected override void Dispose( bool disposing )
{
    if( disposing )
    {
        if (components != null)
        {
            components.Dispose();
        }
    }
    base.Dispose( disposing );
}

#region Windows Form Designer generated code
/// <summary>
/// Required method for Designer support -
/// do not modify
/// the contents of this method with the
/// code editor.
/// </summary>
private void InitializeComponent()
{
    this.button1 = new System.Windows.Forms.Button();
    this.USBInstrument = new System.Windows.Forms.TextBox();
    this.SuspendLayout();
    //
    // button1
    //
    this.button1.Location = new System.Drawing.Point(56, 96);
    this.button1.Name = "button1";
}

```

## 5 Programming Your Instruments

```
        this.button1.TabIndex = 0;
        this.button1.Text = "button1";
        this.button1.Click += new
System.EventHandler(this.button1_Click);
        //
        // textBox1
        //
        this.USBInstrument.Location = new System.Drawing.Point(88, 136);
        this.USBInstrument.Name = "USBInstrument";
        this.USBInstrument.TabIndex = 1;
        this.USBInstrument.Text = "USBInstrument";
        this.USBInstrument.TextChanged += new
System.EventHandler(this.USBInstrument_TextChanged);
        //
        // Form1
        //
        this.AutoScaleBaseSize = new System.Drawing.Size(5, 13);
        this.ClientSize = new System.Drawing.Size(292, 273);
        this.Controls.Add(this.USBInstrument);
        this.Controls.Add(this.button1);
        this.Name = "Form1";
        this.Text = "Form1";
        this.Load += new System.EventHandler(this.Form1_Load);
        this.ResumeLayout(false);

    }
#endregion

/// <summary>
/// The main entry point for the application.
/// </summary>
[STAThread]
static void Main()
{
    Application.Run(new Form1());
}

private void Form1_Load(object sender, System.EventArgs e)
{
}
}
```

```

private void USBInstrument_TextChanged(object sender,
    System.EventArgs e)
{
}

private void button1_Click(object sender, System.EventArgs e)
{
    Ivi.Visa.Interop.ResourceManager io_mgr=new
        Ivi.Visa.Interop.ResourceManagerClass();
    Ivi.Visa.Interop.FormattedIO488 fgen=new
        Ivi.Visa.Interop.FormattedIO488Class();

    fgen.IO=(Ivi.Visa.Interop.IMessage)io_mgr.Open(USBInstrument.Text,Ivi.
    Visa.Interop.AccessMode.NO_LOCK,2000,"");
    fgen.WriteString("*rst",true);
    fgen.IO.Clear();
    fgen.WriteString("FUNCTION SINusoid",true); //Select waveshape
    //
    //Other options are SQUARE, RAMP, PULSE, NOISE, DC, and USER
    //
    fgen.WriteString("OUTPUT:LOAD 50",true);
    //Set the load impedance in Ohms
    //(50 ohms default)
    //
    //May also be INFINITY, as when using oscilloscope or DMM
    //
    fgen.WriteString("FREQUENCY 2500",true); //Set the frequency
    fgen.WriteString("VOLTAGE 1.2",true); //Set the amplitude in
        //Vpp.
        //Also see VOLTAGE:UNIT
    fgen.WriteString("VOLTAGE:OFFSET 0.4",true);
    //Set the offset in Volts
    //Voltage may also be set as VOLTAGE:HIGH and VOLTAGE:LOW for
    //low level and high level.
    //
    fgen.WriteString("OUTPUT ON",true);
    //Turn on the instrument output
}
}
}

```

**Example: Simple Sine Waveform (Visual C++ and VISA)**

This example program is intended for use with Microsoft Visual C++ 6.0 and Agilent VISA. This program uses the arbitrary waveform function to download and output a square wave pulse with a calculated rise time and fall time. The waveform consists of 4000 points downloaded to the function generator as ASCII data.

```
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <visa.h>

#define CHECK(func)\
do {\
    ViStatus _s = (func);\
    if (_s < 0)\
    {\
        fprintf(stderr, "Error: %s returned %d\n", #func, _s);\
        exit(0);\
    }\
} while (0)

/* Specify the default address */
#define DEFAULT_LOGICAL_ADDRESS "GPIB0::10::INSTR"

void main()
{
    ViSession    viDefaultRM, Instrument;
    ViRsrc       TxtAddress = DEFAULT_LOGICAL_ADDRESS;
    ViUInt32     actual;
    char SCPIcmd[10000];
    unsigned short i;

    CHECK(viOpenDefaultRM(&viDefaultRM));
    CHECK(viOpen(viDefaultRM, TxtAddress, VI_NULL, VI_NULL,
    &Instrument));

    /* Specify long seconds timeout for waveform download */
    CHECK(viSetAttribute(Instrument, VI_ATTR_TMO_VALUE, 4000));
```



```

    strcpy(SCPICmd, "*RST\n"); /* Reset the function generator */
    CHECK(viWrite(Instrument, SCPICmd, (ViUInt32)strlen(SCPICmd),
&actual));
    strcpy(SCPICmd, "*CLS\n"); /* Clear errors and status registers */
    CHECK(viWrite(Instrument, SCPICmd, (ViUInt32)strlen(SCPICmd),
&actual));

    /* Compute waveform */

    fprintf(stderr, "Computing Waveform...\n");

    strcpy(SCPICmd, "DATA VOLATILE");
    for(i = 1; i <= 5; i++) /* Set rise time (5 points) */
        sprintf(SCPICmd, "%s,%3.1f", SCPICmd, (double)(i - 1)/5);

    for(i = 6; i <= 205; i++)
        strcat(SCPICmd, ",1"); /* Set pulse width (200 points) */

    for(i = 206; i <= 210; i++) /* Set fall time (5 points) */
        sprintf(SCPICmd, "%s,%3.1f", SCPICmd, (double)(210 - i)/5);

    for(i = 211; i <= 4000; i++)
        strcat(SCPICmd, ",0"); /* Set remaining points to zero */

    strcat(SCPICmd, "\n");

    /* Download data points to volatile memory */

    fprintf(stderr, "Downloading Arb...\n");

    CHECK(viWrite(Instrument, SCPICmd, (ViUInt32)strlen(SCPICmd),
&actual));

    fprintf(stderr, "Download Complete\n");

    /* Set up arbitrary waveform and output */

    strcpy(SCPICmd, "DATA:COPY PULSE, VOLATILE\n"); /* Copy arb to
non-volatile memory */
    CHECK(viWrite(Instrument, SCPICmd, (ViUInt32)strlen(SCPICmd),

```

## 5 Programming Your Instruments

```
&actual));
    strcpy(SCPIcmd, "FUNCTION:USER PULSE\n"); /* Select the active
arb waveform */
    CHECK(viWrite(Instrument, SCPIcmd, (ViUInt32)strlen(SCPIcmd),
&actual));
    strcpy(SCPIcmd, "FUNCTION:SHAPE USER\n"); /* Output the selected
arb waveform */
    CHECK(viWrite(Instrument, SCPIcmd, (ViUInt32)strlen(SCPIcmd),
&actual));

    strcpy(SCPIcmd, "OUTPUT:LOAD 50\n"); /* Output termination is 50 Ohms
*/
    CHECK(viWrite(Instrument, SCPIcmd, (ViUInt32)strlen(SCPIcmd),
&actual));
    strcpy(SCPIcmd, "FREQUENCY 5000;VOLTAGE 5\n"); /* Output frequency is
5 kHz @ 5 Vpp */
    CHECK(viWrite(Instrument, SCPIcmd, (ViUInt32)strlen(SCPIcmd),
&actual));
    strcpy(SCPIcmd, "OUTPUT ON\n"); /* Enable output */
    CHECK(viWrite(Instrument, SCPIcmd, (ViUInt32)strlen(SCPIcmd),
&actual));

    CHECK(viClose(Instrument));
    CHECK(viClose(viDefaultRM));
}
```



## 6 Installing I/O Software

This chapter includes:

- System Requirements for Agilent IO Libraries Suite
- Installing the IO Libraries Suite Software
- Installation Troubleshooting
- Selecting Application Software
- Modifying, Repairing, or Removing IO Libraries Suite
- Keeping Your Software Up To Date
- Using Agilent VISA in Side-by-Side Mode

### NOTE

This chapter assumes that you are installing Agilent IO Libraries Suite from the *Automation-Ready CD*. If you are installing from the Web, double-click the downloaded executable to extract files and launch the setup program; then proceed with the software installation as documented in this chapter.



## System Requirements for Agilent IO Libraries Suite

- **Processor:** 450 MHz Pentium II or higher required, 800 MHz recommended
- **Operating system:** One of the following Microsoft Windows versions:
  - Windows XP Professional or Home Edition, Service Pack 1 or later
  - Windows 2000 Professional, Service Pack 4 or later
- **Browser:** Microsoft Internet Explorer 5.01 or greater
- **Available memory:** 128 MB (256 MB or greater recommended)
- **Available disk space:**
  - 225 MB required for installation:
    - 160 MB for Microsoft .NET Framework
    - 65 MB for Agilent IO Libraries Suite
  - 175 MB required for operation:
    - 110 MB for Microsoft .NET Framework
    - 65 MB for Agilent IO Libraries Suite
- **Video:** Super VGA (800x600) 256 colors or more

If possible, you should use the current version of the Agilent IO Libraries Suite. This version supports the newest interfaces and operating systems, and has the most advanced features. However, you may need an earlier version of the IO Libraries Suite to support an older interface or operating system. For example, Agilent IO Libraries Suite 14.0 is needed to support Windows 98SE or Windows Me.

If you have the *Automation-Ready CD* with Agilent IO Libraries Suite 14.1, you can install version 14.0 directly from the CD-ROM. In Windows Explorer, navigate to the folder called **Previous Versions** on the CD-ROM. Double-click the file **IOLibSuite14\_0.exe** to install version 14.0.

If you need a version earlier than 14.0, or if you do not have the Automation-Ready CD, go to <http://www.agilent.com/find/iolib> to locate the version you need.

## Installing the IO Libraries Suite Software

**NOTE**

You must have Administrator privileges to install Agilent IO Libraries Suite and to run Connection Expert.

---

**NOTE****Every IO Libraries Suite installation is a “Full installation”**

In past versions of IO Libraries, there was an option to install IO Libraries as either a Full version or a Runtime version. Beginning with IO Libraries Suite 14.0, the installation process provides a **Full installation** for all users (i.e. there is no longer a Runtime installation option). This means that every installation now includes the development components of the Agilent IO Libraries Suite software, which allow you to develop your own I/O programs. If you wish to minimize the size of the installed software, select a **Custom** installation; when you get to the **Select Features** dialog, do not select the manuals and the sample programs.

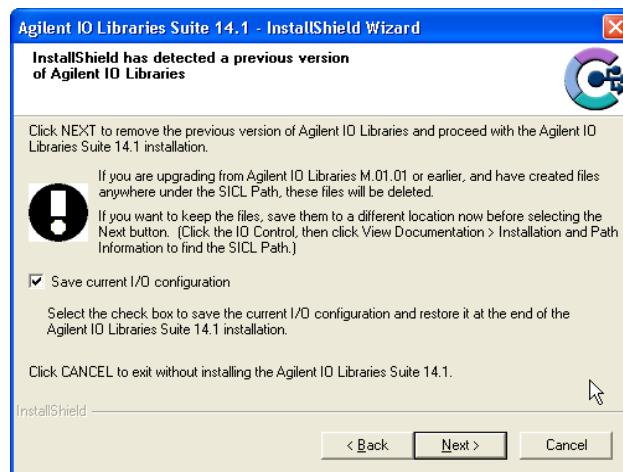
---

This section describes how to install Agilent IO Libraries Suite on your PC. The process is as follows:

- 1 Verify that your PC meets the minimum system requirements. (See “System Requirements for Agilent IO Libraries Suite” on page 124.)
- 2 If you are upgrading to IO Libraries Suite from a previous version of IO Libraries, you must remove the instruments and interfaces listed below before you upgrade your software. This step is necessary in order for these devices to obtain the correct drivers to work with Agilent IO Libraries Suite.
  - a Disconnect any USB instruments from your PC.
  - b Disconnect any Agilent 82357 USB/GPIB interface converters from your PC.
  - c Disconnect any Agilent E8491 IEEE 1394 PC Link to VXI interfaces from your PC.
- 3 Close all other applications on your PC.
- 4 Insert the *Automation-Ready CD with Agilent IO Libraries Suite* into the CD-ROM drive of your PC.

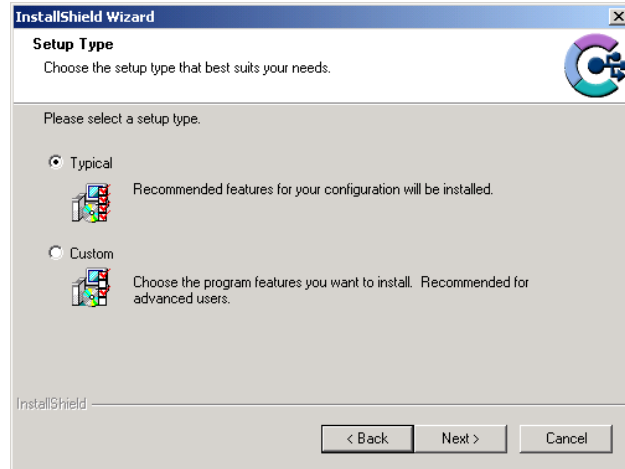
- Wait a few seconds for the auto-run window to appear.
  - If the auto-run window does not appear automatically,
    - Click **Start > Run...**
    - Type `<drive>:autorun\auto.exe`, where `<drive>` is your CD drive letter.
- 5** When the auto-run window appears, click **Install Software** once, and wait for the InstallShield Wizard to appear.
- Optional: If you want more information before you begin the installation, click **Connect in 3 Steps** and/or **Readme**.
- The *Three-Step Connection Procedure* explains how to connect an instrument to your PC quickly and easily, using the IO Libraries Suite.
  - The Read Me file includes last-minute information about the IO Libraries Suite software.
- 6** When the InstallShield Wizard appears, click **Next >** to begin the IO Libraries Suite software installation.
- 7** Read the License Agreement(s). If you accept the terms, click the radio button labeled **I accept the terms of the license agreement** and then click **Next >** to continue.
- 8** If you are upgrading to this version of IO Libraries Suite from a previous version of IO Libraries or IO Libraries Suite, you may see a dialog box similar to [Figure 38](#). Notice that you *must* allow the previous version to be removed if you want to install the current version of IO Libraries Suite.
- a** If you are upgrading from Agilent IO Libraries version M or earlier, *and* you have created files in the IO Libraries installation folder or any subfolder (for example, if you modified IO Libraries sample code and saved it in the original samples folder) these files will be deleted when you proceed with the upgrade. *If you wish to keep these files, you must copy them to a safe location at this time.*
  - b** Select the check box if you want to save your current I/O configuration.

- If you select this option, interfaces and instruments that were either discovered by, or manually added to, either Connection Expert or the old IO Config utility will appear when you run Connection Expert.
  - If you do not select this option, Connection Expert will display only the interfaces and instruments that it discovers automatically. Serial and LAN instruments will not appear; you will have to add them manually.
- c Click **Next >** to proceed with the installation of IO Libraries Suite.



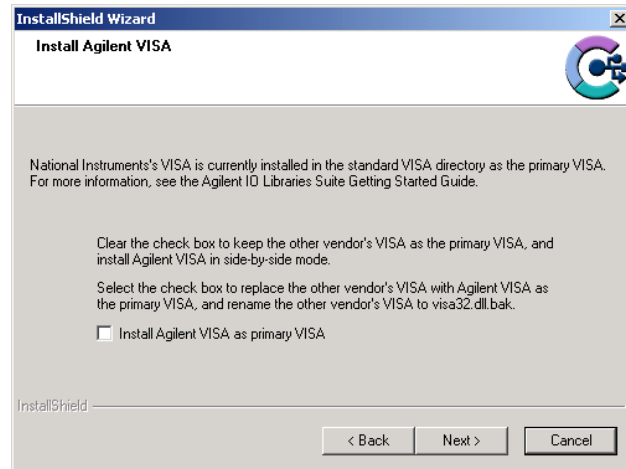
**Figure 38** Upgrading from Previous Version of IO Libraries

- 9 When the InstallShield Wizard **Setup Type** dialog box appears, as shown, select **Typical** or **Custom**, then click **OK**. The **Typical** setup installs the recommended features for your configuration in standard locations on your PC. In a **Typical** setup, if another vendor's VISA software is already installed on this PC, Agilent VISA is installed as secondary. If no other vendor's VISA software is found on this PC, Agilent VISA is installed as primary. (See "Using Agilent VISA in Side-by-Side Mode" on page 136.)



**Figure 39** Choosing Setup Type

**10** If you chose the **Custom** setup *and* you have another vendor's VISA implementation installed on your PC, then you will see the dialog box below, which asks whether you want to install Agilent VISA as the primary VISA.



**Figure 40** Agilent VISA as Primary VISA



If you do not have another vendor's VISA installed, you will see a similar dialog box, but the check box will be selected, indicating that the default is to install Agilent VISA as primary.

For details on this topic, see [Using Agilent VISA in Side-by-Side Mode](#) on page 136.

- 11** If you chose a **Custom** setup, you will now see the **Choose Destination Location** dialog.

From this dialog, you may select specific destination folders for:

- The IO Libraries Suite software
- Agilent VISA (This choice is enabled only if you have not previously installed *VXIPlug&Play* drivers or other VISA components.)
- IVI shared components (This choice is enabled only if you have not previously installed IVI shared components or IVI drivers.)

#### NOTE

The Agilent VISA and IVI shared components folders chosen during this installation will affect the installation location of any related instrument drivers that you install later. If you have already installed such drivers, the related IO Libraries Suite components will be installed in the folders you chose at that time, and these choices will now be disabled.

- 12** If you chose a **Custom** setup, you will now see the **Select Features** dialog box.

- Click on any feature in the list to see the feature description and the space requirements for the selected set of features. It is recommended that you install the manuals and sample programs if you plan to program with the Agilent IO Libraries; however, you may omit them to save space.
- Select the check box for each feature to be installed. Clear the check box for each feature to omit.
- When you are done selecting features, click **Next >**.

**13** The **Start Copying Files** dialog appears; click **Next>** to begin copying files.

- If the Microsoft .NET Framework version 1.1 has not previously been installed on your PC, you will see the .NET Framework files being copied onto your computer. The Microsoft .NET Framework provides necessary infrastructure for Agilent IO Libraries Suite utilities, as well as for .NET programming tools and many other applications. You will only need to install the .NET Framework once, even if you remove and reinstall or upgrade your IO Libraries Suite.
- If the Microsoft .NET Framework Service Pack 1 has not previously been installed on your PC, you will see the Service Pack files being copied onto your computer. This Service Pack is necessary for the proper operation of Agilent IO Libraries Suite.
- If the IVI (Interchangeable Virtual Instrument) Shared Components have not previously been installed on your PC, you will now see them being copied onto your computer. The IVI Shared Components provides necessary infrastructure for Agilent IO Libraries Suite utilities, as well as for IVI-COM instrument drivers.

**14** After the files have been copied, you may see a dialog asking you to restart your PC. This occurs only if you have certain I/O hardware configurations. If you choose not to reboot at this time, you should reboot before running Connection Expert.

**15** When the **InstallShield Wizard Complete** dialog appears,

- a Select the check boxes for the actions that you want to do next.
  - Select the **Read Me** file check box if you want to view the ReadMe file.
  - Select the **Launch Agilent Connection Expert** check box if you want to run Connection Expert now. Run Connection Expert if you are ready to connect, configure, and verify your instruments. If you need to install hardware, such as a GPIB card, then wait to

run Connection Expert after you have done that installation.

**b** Click **Finish**.

- 16** If you are installing Agilent IO Libraries Suite along with another instrument, interface, or software package such as Agilent VEE or Agilent T&M Toolkit, there may be another CD with additional software (drivers, sample programs, etc.). If you want to install this additional software,
- Insert the CD into the CD drive on your PC.
  - Follow the instructions that came with the CD or the hardware or software product.
- 17** Re-connect any USB instruments or E8491 IEEE-1394 FireWire to VXI interfaces that you may have disconnected in [step 2](#). Install any new hardware.
- 18** If Connection Expert is already running, click the **Refresh All** button to identify any hardware you have just installed or re-connected.

If Connection Expert is not already running, run it now to verify your I/O configuration: In the Windows taskbar notification area, click the **IO** icon (IO Control), then click **Agilent Connection Expert**.

- a** Locate your interfaces and instruments in the explorer pane. Click on them to see their properties (displayed in the right-hand pane). Observe their state, also displayed in the right-hand pane: if communication to the interface or instrument has been successfully established, it will be in the Verified state, denoted by a green check mark.
- b** Change the default I/O configuration (if necessary) by clicking **Change Properties...** in the property pane of the interface or instrument you wish to configure.

#### NOTE

If you plan to program your GPIB instruments using the Agilent 488 API, or to run NI-488.2-compatible programs with Agilent interface hardware (such as GPIB cards), you may need to enable the Agilent 488 library. To do this, click **Tools > Agilent 488...**, then select the check box labeled **Enable Agilent GPIB cards for 488 programs**.

- c If you would like to test your connections manually, select your instrument, right-click and choose **Send Commands To This Instrument**.

## Installation Troubleshooting

If you encounter problems while installing the IO Libraries Suite, the following steps may help.

- 1 Close or Cancel all InstallShield Wizards and other Agilent IO Libraries Suite windows. Exit any other applications on your system.
- 2 Browse to the `autorun` folder of your Automation-Ready CD and double-click to run `auto.exe`. This restarts the installation process.
- 3 If you see the standard InstallShield Wizard, step through the installation process as described in this chapter.
- 4 If you see **Modify**, **Repair**, and **Remove** options, select **Repair**. This will reinstall all installed features of IO Libraries Suite. If this does not solve the problem, restart the installation again, select **Remove**, and then reinstall the product.

### “Error 1935” When Installing Microsoft .NET Framework

The Agilent IO Libraries Suite utilities require the Microsoft .NET Framework; if the .NET Framework is not already installed on your PC, the IO Libraries Suite installation process will attempt to install it. There is a known issue that can cause you to receive an “Error 1935” error message during this .NET installation.

For steps to resolve this error, go to <http://www.microsoft.com> and search on “Error 1935”. Follow the instructions given by Microsoft to resolve the error.

## Selecting Application Software

This table gives guidelines for using additional I/O software.

Use This Software:	When You Want to:
<b>Instrument Web Browser (<i>Product Design or Product Characterization</i>)</b>	
Provides a “soft front panel” displayed on your Web browser that you can use to control the instrument (Web-enabled instruments only).	Remotely communicate with instruments from your PC, but do not need to program the instruments. (The IO Libraries are not required.)
<b>Agilent IntuiLink (<i>Product Design or Product Characterization</i>)</b>	
Provides a “soft front panel” specific to an instrument that you can use to control instruments.	Remotely communicate with instruments from your PC, but do not need to program the instruments (The IO Libraries are required.)
<b>IVI-COM Drivers (<i>Product Characterization or Product Test</i>)</b>	
IVI-COM drivers are implemented as COM (Microsoft Common Object Model) objects, and are therefore optimized and recommended for use in Microsoft Visual Studio and Visual Studio .NET.	IVI-COM drivers that implement standard instrument-class interfaces provide syntactical interchangeability; this means that you may be able to replace an instrument in your test system with another, similar instrument with fewer changes or even no changes to your test program, depending on your use of instrument-specific interfaces.
<b>VXIplug&amp;play Drivers (<i>Product Characterization or Product Test</i>)</b>	
VXIplug&play drivers (also known as Plug&Play or Universal Instrument Drivers) conform to a set of system-level standards produced by the VXIplug&play Systems Alliance. These standards apply to instrument drivers, soft front panels, installation packages, documentation, technical support and application development environments. VXIplug&play drivers are widely used and based on the VISA API.	VXIplug&play drivers allow you to develop application programs that are portable across many computer platforms and I/O interfaces.

## Modifying, Repairing, or Removing IO Libraries Suite

- 1 Click **Start > Settings > Control Panel > Add/Remove Programs**
- 2 Select **Agilent IO Libraries Suite**.
- 3 Click **Change/Remove**
- 4 Select the action you want to take:

**Modify:** Adds, replaces, or removes selected features. You can change your installation from primary to side-by-side or from side-by-side to primary by using this option. You can also use this to install manuals or sample code if you did not initially install them.

**Repair:** Reinstalls all features.

**Remove:** Removes the product from your PC.

## Keeping Your Software Up To Date

Web resource for the latest IO Libraries Suite software:

[www.agilent.com/find/iolib](http://www.agilent.com/find/iolib)

When you update the IO Libraries Suite software, you may choose to preserve your test system configuration. This includes the instruments, instrument aliases, interfaces, and all related properties.

**Web resource for the latest instrument drivers:**

[www.agilent.com/find/drivers](http://www.agilent.com/find/drivers)

This Web site includes many types of instrument drivers, including IVI-COM and *VXIplug&play* drivers.

## Using Agilent 488 in Multi-Vendor Systems

Agilent IO Libraries Suite includes the Agilent 488 library, provided for compatibility with National Instruments' NI-488.2 and other vendors' 488 libraries. Agilent 488 allows you to use your NI-488.2 compatible programs with Agilent GPIB interfaces and other vendors' interfaces (even simultaneously).

If your test system includes only Agilent GPIB interface hardware and software, you do not need to take any special action to use Agilent 488.

If your system includes NI-488.2 software from National Instruments, or any other vendor's compatible implementation, you may need to explicitly enable Agilent 488. To do this, click **Tools > Agilent 488..**, then select the check box labeled **Enable Agilent GPIB cards for 488 programs.**

### NOTE

If you install Agilent IO Libraries Suite on a PC that has third-party 488 library software (such as NI-488.2) already installed, *or* if you subsequently install such third-party software, Agilent 488 will automatically be disabled. If you want to use Agilent 488, you must re-enable it after installing any third-party 488 library.

Although Agilent 488 does not conflict with NI-488.2 and both libraries can be used simultaneously, some National Instruments software may display error messages when Agilent 488 is enabled; for example, you may see the message **Missing or Damaged GPIB-32.DLL** when you attempt to use National Instruments Measurement & Automation Explorer to change the properties of an NI GPIB interface. If this occurs:

- 1 Disable Agilent 488: in Connection Expert, click **Tools > Agilent 488...** and clear (unselect) the check box labeled **Enable Agilent GPIB cards for 488 programs.**
- 2 Perform the operation that caused the error (such as changing the properties of the NI interface).

- 3 Re-enable Agilent 488: in Connection Expert, click **Tools > Agilent 488...** and select the check box labeled **Enable Agilent GPIB cards for 488 programs**.

(In a few cases, you may be prompted to restart your PC before you can successfully enable or disable Agilent 488.)

## Using Agilent VISA in Side-by-Side Mode

Agilent IO Libraries Suite includes an implementation of VISA (Virtual Instrument Software Architecture), one of three available application programming instruments for instrument control. The VISA standard requires that `visa32.dll`, the dynamic-link library that implements the VISA interface, be installed in prescribed locations; therefore, only one installed implementation (the **primary** implementation) can fully comply with the standard at any one time. Because of this, you must take special care if you want to use Agilent IO Libraries Suite alongside any other implementation of VISA, such as National Instruments' NI-VISA.

When you have both Agilent VISA and another vendor's VISA installed on the same system, you will need to decide whether to let your programs use the primary VISA or whether to explicitly direct them to use the Agilent VISA implementation.

### NOTE

In general, you do not need to install NI-VISA to use National Instruments GPIB cards and devices. You can install National Instruments' NI-488.2 as a driver for these devices, and avoid the complications of side-by-side operations.

Agilent IO Libraries Suite supports the option to install Agilent VISA in side-by-side mode, which allows Agilent VISA to be used simultaneously with another vendor's VISA implementation. In side-by-side mode, Agilent VISA is installed only in a secondary location and therefore does not overwrite another vendor's VISA if it is present on the computer.



In general, non-Agilent VISA implementations do not support operation as the secondary VISA (in side-by-side mode). This means that you must operate Agilent VISA in side-by-side mode as the secondary VISA, and the other vendor's VISA as primary.

### **About primary and secondary VISA:**

- VISA programs normally use `visa32.dll`, which is installed in the Windows system directory. This is the primary VISA, as defined by the VISA standard.
- Agilent VISA is implemented in `agvisa32.dll`, which is also installed in the Windows system directory.
- If Agilent VISA is installed as primary, the Agilent version of `visa32.dll` is installed in the Windows system directory. It simply forwards all VISA calls to `agvisa32.dll`.
- If you install Agilent VISA as primary, all files needed to support side-by-side mode are also installed. This means that if you subsequently install another vendor's VISA as primary, you can still use Agilent VISA as secondary.

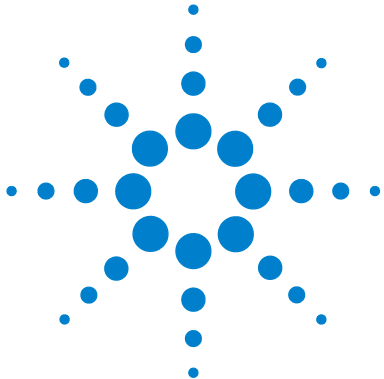
If Agilent VISA is installed in side-by-side mode (as the secondary VISA):

- `agvisa32.dll` is installed, but Agilent's `visa32.dll` is not. Thus, it will not overwrite another vendor's `visa32.dll` residing in the standard location.
- Agilent support files are not installed in the `bin`, `include` and `lib` directories of the VISA path (e.g. `c:\program files\visa\winnt`), since they would overwrite the other vendor's support files. Copies of the Agilent versions of the VISA support files reside in the `agbin`, `include` and `lib` subdirectories under `<VISA path>\agvisa`. (These files are installed even when Agilent VISA is primary.)

- The IO Control menu and the status bar at the bottom of the Connection Expert window will contain a message indicating whether Agilent VISA is installed as primary, installed as secondary, or installed as primary but overwritten by another vendor's VISA. In the latter case, Agilent VISA will operate identically to a secondary installation.

VISA programs that are linked to the standard VISA DLL (e.g. C:\WINDOWS\system32\visa32.dll) will use the primary VISA. However, if a VISA program is linked with `agvisa32.lib` or dynamically loads `agvisa32.dll`, it will always use Agilent VISA (regardless of whether Agilent VISA is primary or secondary). Addressing is identical regardless of whether you are using the primary or secondary VISA in your program.

See the *Agilent IO Libraries Suite Online Help* for more information about side-by-side VISA operation.



## 7 TCP/IP Network Basics

This chapter includes:

- LAN Interface Overview
- TCP/IP Protocols
- IP Addressing
  
- IP Address Configuration Methods
- Device Hostname Services
- Configuring Your PC for LAN Operation

### NOTE

The information in this section is a summary of TCP/IP networks and LANs and is not intended to be a complete discussion of the subject. Consult standard reference texts for further details on TCP/IP and LANs.

---



## LAN Interface Overview

This section provides an overview of Local Area Networks (LANs) that use Transmission Control Protocol/Internet Protocol (TCP/IP), including:

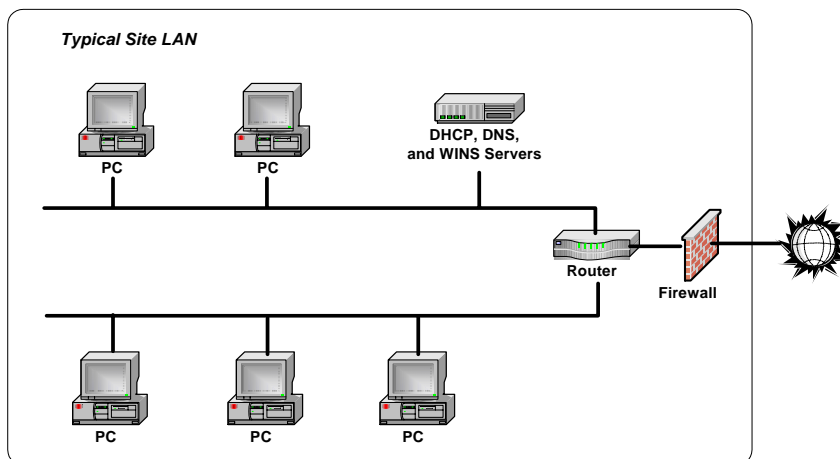
- Typical Network Topologies
- LAN Hardware Architecture

### Typical Network Topologies

LANs using TCP/IP can be divided into two categories: Site LANs and Private LANs.

#### Site LAN Topology

A Site LAN can be a workgroup LAN, Intranet, or enterprise (corporate) LAN. Typically, a site LAN includes several PCs, routers, and servers. The LAN network is usually administered and controlled by a **system administrator** (or **network administrator**) who installs, monitors, and troubleshoots the network.



**Figure 41** Portion of a Typical Site LAN

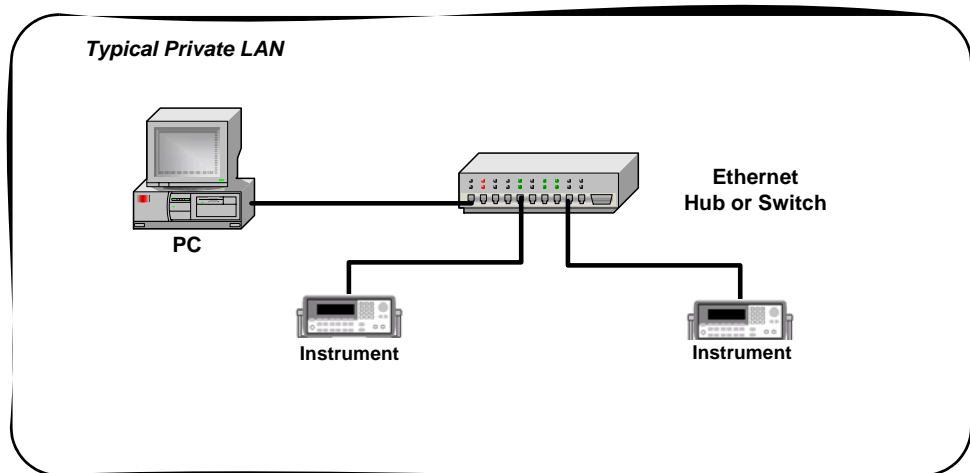
Within the site LAN, every device (PC, router, server, etc.) is called a **host**. A host is any device on a network that has a TCP/IP address. TCP/IP addresses are called **IP addresses**, and each device on the network must have a unique IP address. A typical IP address for a device is 156.140.105.50. In general, communication within the site LAN and to private LANs is behind a **firewall**.

Communication among devices on the site LAN is controlled by **routers**. A router is a host that interfaces with other networks and can move data (called packets) from one network to another or can move data within the network, depending on the IP addresses of the hosts that are communicating with each other.

The **servers** on the site LAN can supply addressing and naming services, such as Dynamic Host Configuration Protocol (DHCP) to automatically assign IP addresses to hosts on the network or Domain Name Service (DNS) or Windows Internet Naming Service (WINS) that allow hostnames to be associated with their IP addresses.

### **Private LAN Topology**

A private LAN (also called a local or isolated network) can operate as a local network (not connected to any other network) or can be connected to a site LAN or to another local network. This figure shows an example private LAN with three hosts (PC and two instruments). Communication between the PC and the instruments is via a **switch** or **hub**. You can also connect a single instrument to a PC by using a **crossover cable**.



**Figure 42** Typical Private LAN

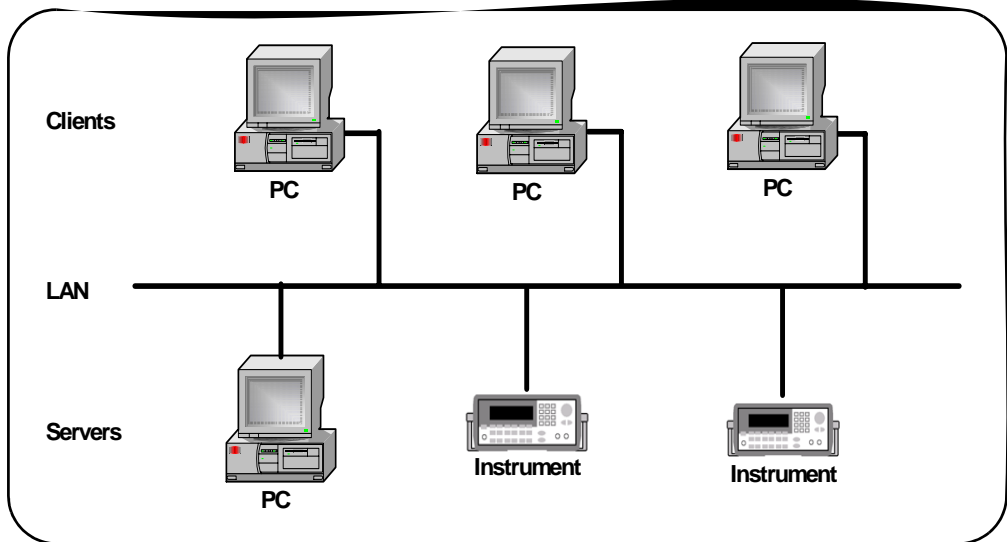
## LAN Hardware Architecture

A LAN is a way to extend the control of instrumentation beyond the limits of typical instrument interfaces. You can communicate with instruments using a web browser, the Telnet utility, or sockets. However, to program (send SCPI commands) over the LAN, you must first configure the LAN interface using Agilent Connection Expert, provided with the Agilent IO Libraries Suite.

### Client-Server Model

The LAN software uses the **client-server model** of computing. Client-server computing refers to a model where an application (the **client**) does not perform all necessary tasks of the application itself. Instead, the client makes requests of another device (the **server**) for certain services.

A LAN interface, such as a PC, makes requests over the network to a server, such as a PC or a LAN-enabled instrument. For example, in this figure the Server PC and the instruments can act as LAN servers. Once the LAN server has completed the requested operation on the instrument or device, the LAN server sends a reply to the LAN interface. This reply contains requested data and status information that indicates whether or not the operation was successful.

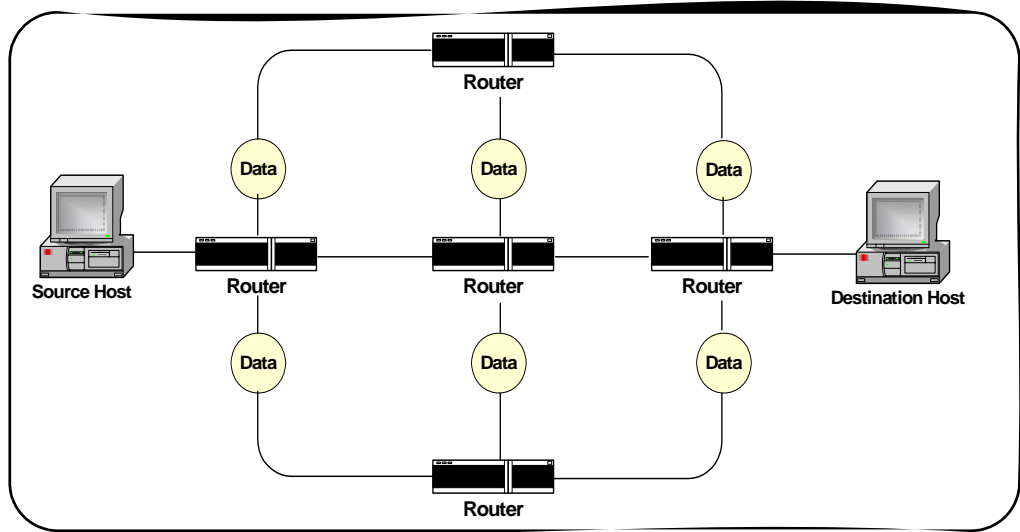


**Figure 43** Client-Server Model

### Packet Switching

A TCP/IP network is a **packet-switched** network. In this type of network, the computer that is sending the data (source host) breaks the data into smaller segments, called **packets**.

Each packet is individually addressed and is sent to the destination (destination host.) The destination host then reassembles the packets into the original message. Each packet can be sent from source to destination in any of several routes. The **routers** in the network control the paths of the packets.



**Figure 44** Example of Packet-Switched Network



## TCP/IP Protocols

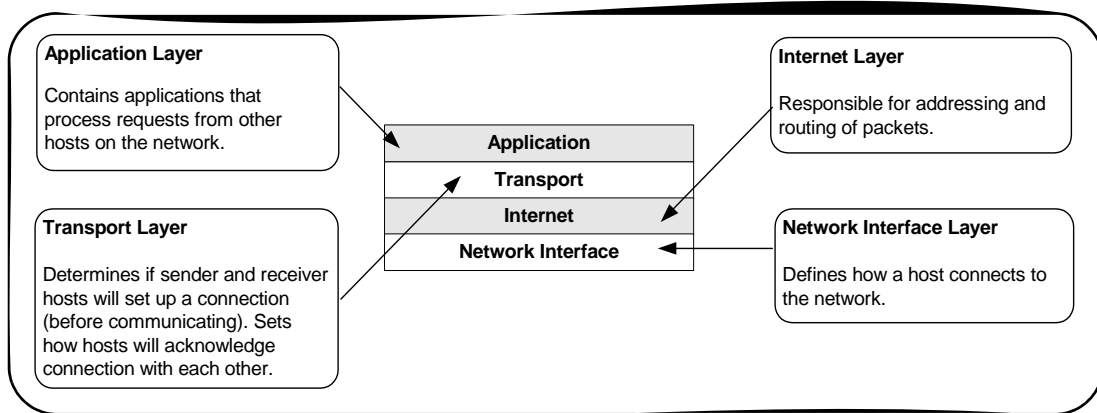
This section summarizes protocols for LAN networks that use TCP/IP for communication between hosts (devices such as computers, printers, and instruments), including:

- The TCP/IP Network Model
- The Network Interface Layer
- The Internet Layer
- The Transport Layer
- The Application Layer

### The TCP/IP Network Model

The TCP/IP network model is based on **protocols** and an associated set of layers that control the actions of the network. A **protocol** is a rule or set of rules and standards for communication and data transfer between **hosts** on a network. A **host** is any device on the network (such as a computer, server, printer, etc.) that has a TCP/IP address. To exchange data on the network, the source and destination hosts must agree on the protocol and each host must have a unique (TCP/IP) address. When protocols are grouped together, they form a **protocol suite** and work together as a **protocol stack**.

**TCP/IP Layers** The TCP/IP network model is based on the Department of Defense (DoD) model.



**Figure 45** Functions of the TCP/IP Layers

**TCP/IP Protocol Overview** This table summarizes the components of each layer and shows the applicable Request for Comments (RFC) for each component.

Layer	Component Name	Description	Specification
Internet	IP - Internet Protocol	The lowest level protocol by which data are sent from one computer to another on the Internet.	RFC 791* RFC 2460**
	ICMP - Internet Control Message Protocol	A message control and error-reporting protocol between IP devices on a network.	RFC 792
	ARP - Address Resolution Protocol	A protocol for translating IP addresses into Ethernet addresses.	RFC 826
Transport	TCP - Transmission Control Protocol	A connection-oriented protocol running over IP that reliably delivers data.	RFC 793
	UDP - User Datagram Protocol	A datagram-oriented protocol running over IP	RFC 768

Layer	Component Name	Description	Specification
Application	DNS - Domain Name System	A protocol for translating a Hostname into an IP address.	RFC 1034 and RFC 1035
	DHCP - Dynamic Host Configuration Protocol	A protocol for obtaining TCP/IP parameters from a central server.	RFC 2131 and RFC 2132
	FTP - File Transfer Protocol	A protocol that defines how files are transferred from one host to another on a network.	RFC 959
	HTTP - Hypertext Transfer Protocol	A protocol that defines how files are exchanged on the Internet	RFC 1945 and RFC 2068

\* = IPv4 (current standard), \*\* = IPv6 (proposed new standard)

## The Network Interface Layer

The lowest layer in the TCP/IP stack is the Network Interface Layer. The primary responsibility of this layer is to define how a host device (computer, instrument, etc.) connects to the network. The Network Interface Layer acts as a host's connection (interface) to the network. There are no TCP/IP protocols associated with the Network Interface layer.

The Network Interface Layer is used to send and receive packets. At the Network Interface Layer, a header that contains addressing information is applied to each packet. A part of this header is the host **hardware address**. The hardware address must be unique to the device and does not change during the life of the device.

The hardware address, also called the Media Access Control (MAC) Address, Ethernet Address, Physical Address, or Network Interface Card (NIC) Address, is a 12-digit hexadecimal address. A typical hardware address is 00:30:D3:00:00:23, where the first six digits represent the manufacturer of the device and the last six digits represent the serial number assigned to the device.

### NOTE

For a TCP/IP packet to be delivered to a device, the packet **must** contain the destination device's hardware address.

As packets are sent through the network, each host on the network looks at the packet to see if the packet is addressed to the host's hardware address. If not, the host ignores the packet.

### The Internet Layer

The Internet Layer of the TCP/IP model contains the protocols responsible for addressing and routing of packets. The Internet Layer includes several protocols, including:

- Internet Protocol (IP)
- Address Resolution Protocol (ARP)
- Internet Control Message Protocol (ICMP)

For TCP/IP communications to be successful, the packet examined by the Network Interface Layer must include a hardware address. As the packet moves up to the Internet Layer, it also must include an **IP address**. The Internet Layer provides the protocols to determine the hardware address for routing the packet to its destination.

**Internet Protocol (IP)** Internet Protocol (IP) is responsible for determining the source and destination IP addresses of every packet sent on the network. Typically, IP addresses are assigned by a **System Administrator**. An IP address is assigned to each host on a network and each host must have a unique IP address.

In contrast to a hardware address that refers to a physical network interface card, an IP address is a **logical address** that is assigned to the host. The IP address can be changed, since it refers only to the host. A typical IP address is 167.155.21.45. See [IP Addressing](#) for a description of IP addresses.

**Address Resolution Protocol (ARP)** Address Resolution Protocol (ARP) is a protocol used to **resolve** (translate) a **logical address** to a physical (hardware) address. ARP is

used when a source host wants to communicate with a destination host, but has only the IP address. After the hardware address is resolved, ARP maintains that information for a short period of time.

**Internet Control Message Protocol (ICMP)** Internet Control Message Protocol (ICMP) is primarily used to send error messages, perform diagnostics, and control data flow. You can also use ICMP with the Packet InterNet Groper (**Ping**) utility. To ping another host from a Command Prompt, type `ping <ip_address>`.

#### Example: Using ping for Echo Request

For example, you could use the Ping utility to send ICMP echo request packets to the destination host and request the destination host return these packets. If the packets are returned, you can assume the connection is good. If the packets are not returned, a connectivity problem exists. This figure shows an example return from pinging a computer at IP address 156.140.72.1.

```
C:\>ping 156.140.72.1

Pinging 156.140.72.1 with 32 bytes of data:

Reply from 156.140.72.1: bytes=32 time<10ms TTL=254
Reply from 156.140.72.1: bytes=32 time<10ms TTL=254
Reply from 156.140.72.1: bytes=32 time<10ms TTL=254
Reply from 156.140.72.1: bytes=32 time<10ms TTL=254

Ping statistics for 156.140.72.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

**Figure 46** Example Using Ping Utility

## The Transport Layer

The Transport Layer of the TCP/IP model contains the protocols responsible for addressing and routing of packets. The Transport Layer determines if the sender and receiver will establish a connection before communicating and how often acknowledgements of the connection are sent. The protocols at the Transport Layer deliver data to and receive data from the Transport Layer protocols of other hosts. Hosts can be on the same local network or on a remote network. The Transport Layer has two protocols:

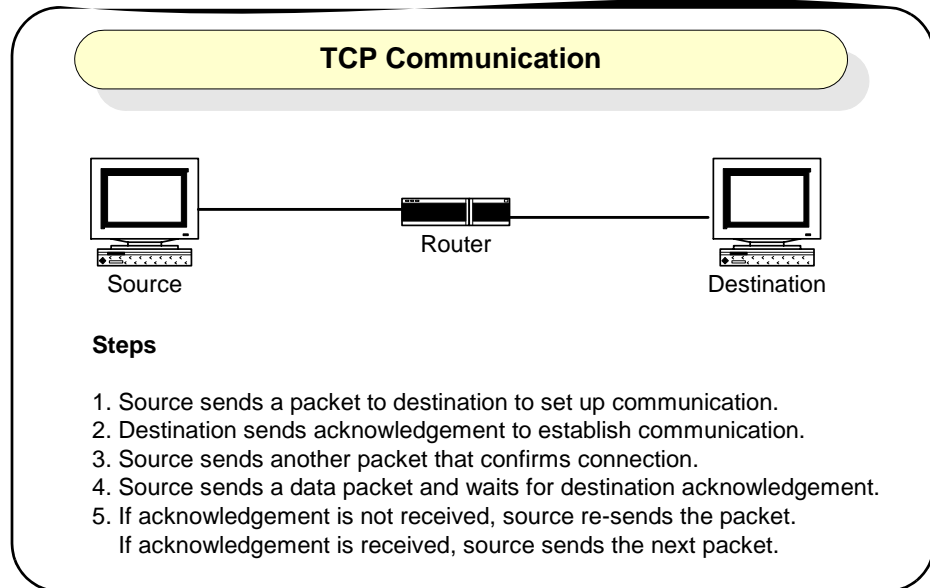
- [Transmission Control Protocol \(TCP\)](#)
- [User Datagram Protocol \(UDP\)](#)

**Transmission Control Protocol (TCP)** Transmission Control Protocol (TCP) is the protocol that connects the sending host and the receiving host to each other. TCP provides the connection as packets are moved between hosts having the conversation. Every packet has a TCP header that includes **sequence numbers**, acknowledgement numbers, address information, and other information. If packets get out of order enroute from the source to the destination, the sequence number allows the packets to be reassembled in the correct order.

As each packet is sent from the source host, an acknowledgement of receipt is sent by the destination host within a specified time. If the acknowledgement is not sent within this time, the sender re-sends the packet.

If the receiver gets the packet in a damaged condition, the packet is discarded and the receiver sends no acknowledgement. In this case, since an acknowledgement was not received in the specified time, the sender re-sends

the packet. This figure shows a network in which the source and destination are using TCP and summarizes the steps to use TCP for communication between hosts.



**User Datagram Protocol (UDP)** User Datagram Protocol (UDP) is used for connectionless, non-guaranteed communication. Unlike TDP, UDP does not set up a connection and does not use acknowledgements. Instead, UDP sends out packets without waiting for acknowledgement from the destination.

## The Application Layer

The Application Layer is the layer where requests for data or services are processed. Applications at this layer are waiting for requests to process and all applications are “listening” at their respective **port**. The Application Layer has two protocols:

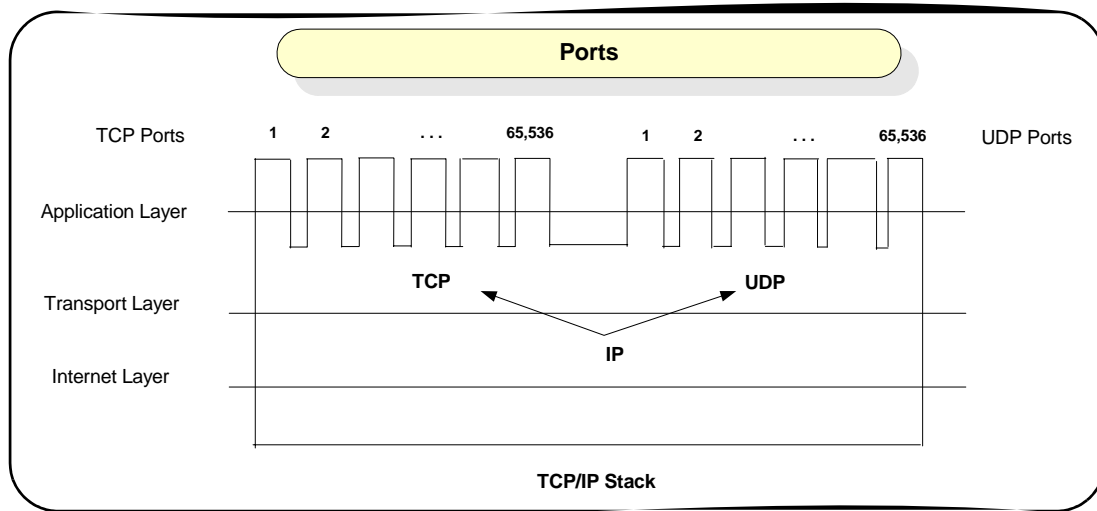
- File Transfer Protocol (FTP)

- Hypertext Transfer Protocol (HTTP)

**NOTE**

The Application Layer is not where an Internet browser, spreadsheet, etc. interact. Applications running at the Application Layer interact with the browser, spreadsheet, etc. applications.

**Ports** This figure shows the TCP/IP protocol stack. In the stack, TCP and UDP each have access to 65,536 ports at the Application Layer. As a packet moves up the stack, IP directs the packet to either a TCP port or to a UDP port. Since all applications are listening at their respective ports, when the packet arrives at the appropriate port, it can be processed correctly.

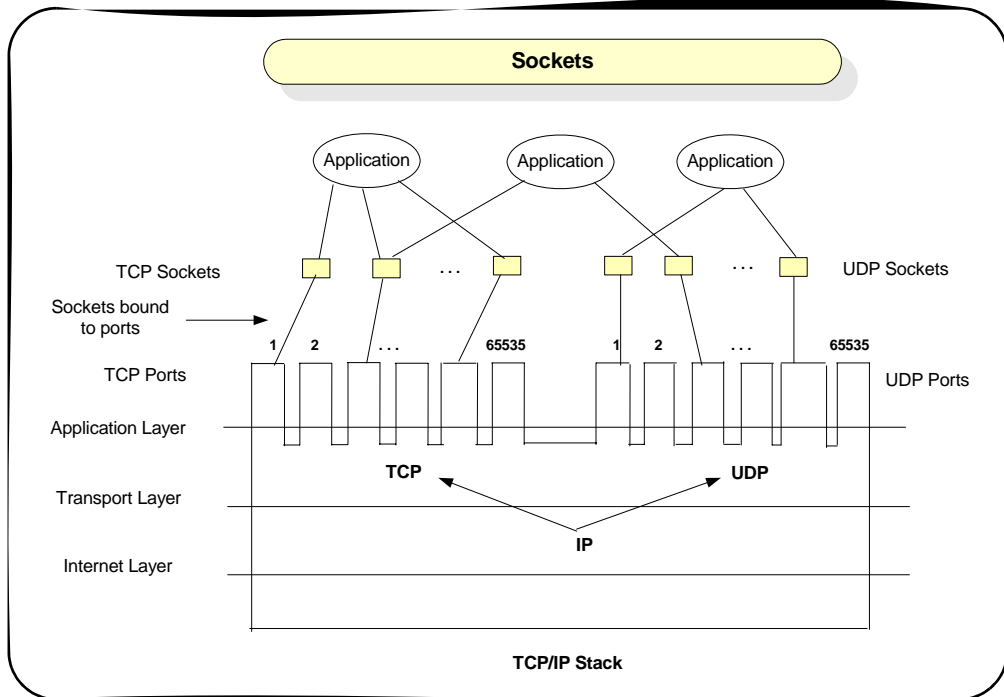


**Figure 47** TCP/IP Protocol Stack

**Sockets** For TCP/IP protocol, when a packet is delivered to a specific IP address, it is passed up to TCP or UDP and then to the appropriate host. This process forms a funnel through the TCP/IP stack, called a **socket**. A socket is uniquely defined by the IP address, the end-to-end protocol (TCP or UDP), and the port number.



This figure shows typical socket architecture. When a socket is first created, it has an associated protocol but not an IP address or port number. A socket must be **bound** to a port number before it can receive messages from a remote application.



**Figure 48** Typical Socket Architecture

**File Transfer Protocol (FTP)** File Transfer Protocol (FTP) is an application used to transfer files from a host to another host and store the files on the requesting host. In an FTP session, one host (the client) requests a file and the other host (the server) transfers a copy of the file to the client. The two hosts thus establish a **client/server** relationship. File transfer can be in text or binary format.

An FTP client can use a word processor, FTP command line utility or FTP command server to request a file from the server. The command-line FTP client application requires you to know FTP commands, while an FTP command

interpreter allows you to connect to an FTP server without using FTP commands. You can login to FTP using anonymous so that no password is required.

**Hypertext Transfer Protocol (HTTP)** Hypertext Transfer Protocol (HTTP) is an application used to transfer files from one host (the server) to another host (the client) and to display the files at the requesting host. The HTTP application runs on a Web server, listens for requests at a TCP port (usually port 80 for requests), and sends files back to the requestor.

The requesting host displays the files on a web browser, such as Internet Explorer. A client makes an HTTP request by typing a **Uniform Resource Locator (URL)** in the address line of the web browser or by clicking a **hyperlink** on a page that is displayed on the web browser.

# IP Addressing

This section describes IP addressing for TCP/IP networks, including:

- [IP Address Classes](#)
- [Subnets and Subnet Masks](#)
- [Local and Remote Networks](#)
- [IP Address Configuration Methods](#)

## IP Address Classes

Each host on a TCP/IP network must have a unique address. This address is called the **IP address** and consists of a **network portion** and a **host portion**. The network portion and host portion of an IP address are determined by the **subnet mask**.

Each IP address consists of four decimal numbers separated by periods. However, TCP/IP uses the 32-bit binary equivalent of the IP address. For example, the decimal value and binary equivalent of IP address 14.230.26.116 is:

14.230.26.116 = 0000 0110.1110 0110.0001 1010.0111 0100

IP addresses are divided into five classes: Class A, Class B, Class C, Class D, and Class E. This table summarizes standard Class A, Class B, and Class C IP addresses. In the table, the **Range** is the range of values for the first 8 bits of the IP address, regardless of class. In the **Binary Equivalent** column, N = a network bit and H = a host bit.

Class	Range	Network Bits	Binary Equivalent
A	1 - 127	8	<b>NNNN NNNN</b> .HHHH HHHH.HHHH HHHH.HHHH HHHH
B	128 - 191	16	<b>NNNN NNNN.NNNN NNNN</b> .HHHH HHHH.HHHH HHHH
C	192 - 223	24	<b>NNNN NNNN.NNNN NNNN.NNNN NNNN</b> .HHHH HHHH

**NOTE**

Dot-notation addresses ("nnn.nnn.nnn.nnn" where "nnn" is a byte value) such as IP addresses must be expressed with care, as most web software on the PC will interpret byte values with leading zeros as octal numbers. Thus, "255.255.020.011" is actually equivalent to decimal "255.255.16.9" rather than "255.255.20.11" because ".020" is interpreted as "16" expressed in octal and ".011" as "9". To avoid confusion, use only decimal expressions of byte values (0 to 255), with **no leading zeros**.

For example, the Agilent 33220A assumes that all IP addresses and other dot-notation addresses are expressed as decimal byte values and strips all leading zeros from these byte values. Thus, trying to enter "255.255.020.011" in the IP address field, it becomes "255.255.20.11" (a purely decimal expression). You should enter exactly the same expression, "255.255.20.11" in your PC web software to address the instrument. Do not use "255.255.020.011" as the PC will interpret that address differently due to the leading zeros.

**Example: Class A, B, and C IP Addresses**

This table shows three example IP addresses, where the bold part of the binary equivalent is the network portion of the address and the non-bold portion is the host portion of the address. For a Class A address, the first 8 bits are the network portion and the remaining 24 bits are the host portion of the address.

For a Class B IP address, the first 16 bits are the network portion and the remaining 16 bits are the host portion of the address. For a Class C IP address, the first 24 bits are the network portion and the remaining 8 bits are the host portion of the address.

Class	IP Address	Binary Equivalent
A	54.16.23.1	<b>0011 0110.</b> 0001 0000. 0001 0111. 0000 0001
B	154.16.23.1	<b>1001 0110. 0001 0000.</b> 0001 0111. 0000 0001
C	204.16.23.1	<b>1100 1100. 0001 0000. 0001 0111.</b> 0000 0001

## Subnets and Subnet Masks

As noted, an IP address consists of a network portion and a host portion. A **subnet mask** is a number that looks like an IP address that shows IP how many bits are used for the network portion of the IP address by “masking” the network portion of the IP address. Every IP address must have a subnet mask. You can use standard or custom subnet masks. To see how subnet masks are used, we will first define a **subnet**.

**What is a Subnet?** Large networks can be divided by a System administrator by creating a number of smaller networks, called **subnets**. The larger network is divided into subnets by using **routers** (also called **default gateways**). All devices on one side of the router form a separate subnet. Each subnet must have a unique set of IP addresses that are defined by the **subnet mask** for that network. Each side of the router must have a separate network address.

**Standard Subnet Masks** A subnet mask (like an IP address) is 32 bits long, with several contiguous bits (all 1s) that represent the network portion of the IP address and the rest of the bits (all 0s) that represent the host portion of the IP address.

For example, the standard subnet mask for a Class A address is **255.0.0.0** (binary **1111 1111.0000 0000.0000 0000 0000**), so the first eight bits (all 1s) are used for the network portion of the IP address and the remaining 24 bits (all 0s) are used for the host portion of the IP address. This table shows the standard subnet masks for Class A, Class B, and Class C IP addresses.

Class	Standard Subnet Mask	Standard Subnet Mask (Binary Equivalent)
A	255.0.0.0	1111 1111. 0000 0000. 0000 0000. 0000 0000
B	255.255.0.0	1111 1111. 1111 1111. 0000 0000. 0000 0000
C	255.255.255.0	1111 1111. 1111 1111. 1111 1111. 0000 0000

**Custom Subnet Masks** You can create custom subnet masks for Class A, Class B, and Class C IP addresses. For example, for a Class A IP address, by using custom subnet mask **255.255.224.0** you can create 2,046 unique networks with 4,094 unique host on each network.

A custom subnet mask adds more bits for the network portion of the IP address and uses fewer bits for the host portion of the IP address. These additional bits are called **subnet bits**.

#### NOTE

Consult a standard network or TCP/IP book or use a subnet calculator (available on the Web) for creating custom subnets.

## Local and Remote Networks

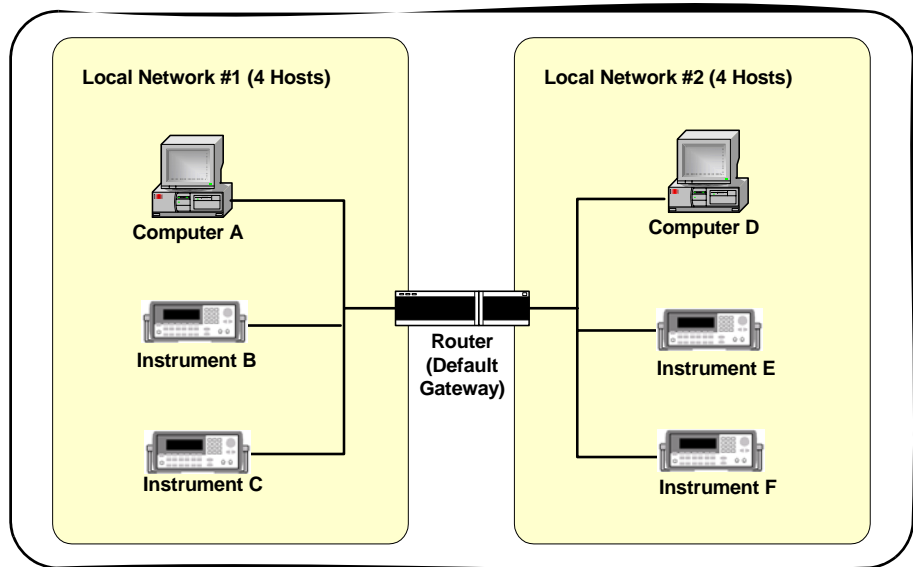
For every packet sent from a source host, IP must determine if the destination host is on the same **local network** or is on a **remote network**. If the destination host is on the same local network as the source host, IP directly gets the hardware address of the destination host. If the destination host is on a remote network, IP sends packets to the **router**.

### Example: Creating Local and Remote Networks

For example, consider the two networks (subnets) in this figure that are created using a router. Computer A, instrument B, instrument C, and one side of the router form one local network, while computer D, instrument E, instrument F, and the other side of the router form another local network.

If computer A wants to send a packet to instrument B or to instrument C, the destination is **local** (all hosts are on the same side of the router). In this case, IP can get the hardware address of the destination host (instrument B or instrument C) and can establish communication directly with the destination host.

However, if computer A wants to send a packet to instrument E or to instrument F, the destination is **remote** (hosts are separated by the router). In this case, IP gets the hardware address of the router. The packet is then sent to the router for transmission to the destination host (instrument E or instrument F).



**Figure 49** Two Networks (Subnets) Using a Router

## IP Address Configuration Methods

This section introduces some IP address configuration methods, including:

- [Configuration Methods Overview](#)
- [Dynamic Host Configuration Protocol \(DHCP\)](#)
- [Auto-IP/ZEROCONF](#)
- [Duplicate IP Address Detection](#)

### Configuration Methods Overview

IP address configuration methods can be divided into two categories: automatic IP configuration and manual IP configuration.

- **Dynamic Host Configuration Protocol (DHCP)** provides **automatic** TCP/IP setup for devices on networks that support DHCP.
- **Auto-IP/ZEROCONF** provides **automatic** TCP/IP setup for devices on networks that do not support DHCP.
- **Manual IP Configuration** provides **manual** TCP/IP setup for devices on manually configured networks.

This table shows the applicability of these methods to various network topologies.

Network Topology	Automatic IP			Manual IP
	DHCP	Auto-IP	ZEROCONF	
Site LAN [with DHCP]	Yes	No	No	Yes
Site LAN [without DHCP]	No	Yes	Yes	Yes
Private LAN [with DHCP]	Yes	No	No	Yes
Private LAN [without DHCP]	No	Yes	Yes	Yes



## Dynamic Host Configuration Protocol (DHCP)

Dynamic Host Configuration Protocol (DHCP) allows each host on the network to be automatically assigned a unique IP address when the device is connected to a network that supports DHCP and is turned ON.

To enable DHCP on a Site LAN, the System Administrator assigns a pool of IP addresses to a **DHCP Server** to be leased to hosts on the network. Each host on the network is called a **DHCP Client**.

## Auto-IP/ZEROCONF

Auto-IP/ZEROCONF can be used to automatically assign IP addresses on networks that do not have DHCP Servers. Auto-IP is a defacto standard for IP address allocation implemented by Microsoft and Apple. ZEROCONF is an Internet standard (RFC) that is currently under development and is expected to be widely implemented in Linux, Windows, etc. Auto-IP and ZEROCONF are similar techniques for IP address allocation and can interoperate together.

Both Auto-IP and ZEROCONF allocate IP addresses from the link-local IP address range (169.254.xxx.xxx). The addresses are allocated using an ARP-based protocol to determine if the desired address is already in use. Auto-IP and ZEROCONF can both be used such that DHCP is tried first and then Auto-IP/ZEROCONF are used after DHCP times out (typically after about 45 - 60 seconds).

Check your instrument's *User's Guide* to determine if the instrument supports Auto-IP/ZEROCONF and, if supported, if Auto-IP/ZEROCONF can be disabled.

## Duplicate IP Address Detection

Duplicate IP address detection ensures that an instrument does not attempt to start operation using an IP address that is already in use on that network. Duplicate IP Address detection provides basic diagnostic information to identify a problem on the network.

Duplicate IP Address detection is described by several Internet standards (RFC 2642/RFC 2131/ZEROCONF). It is implemented using ARP (Address Resolution Protocol) and uses broadcast Ethernet communication. The scope of duplicate IP address detection is the current Ethernet subnet. Most operating systems (Windows, Apple, etc.) implement duplicate IP address detection.

**Duplicate IP Addresses on non-DHCP Networks** Generally, duplicate IP addresses only occur on a manually configured IP address network that does not use DHCP and Auto-IP/ZEROCONF. For example, a user may try to determine an IP address for a host by PINGing IP addresses on the network until an IP address is selected that does not respond.

Based on this non-response, the user may assume the IP address is unused and assign it to their device. However, at a later date, a host that legitimately has that IP address will attempt to use it. As a result, both hosts will experience network problems that can be very difficult to track down.

A duplicate IP address can also happen when a user makes an error during manual entry of an IP address and accidentally configures an IP address already in use elsewhere.

**Duplicate IP Addresses on DHCP Networks** Duplicate IP addresses on DHCP systems are unlikely but they are possible. The DHCP specification (RFC 2131) specifies how a duplicate IP address check should be done within the DHCP Discover/Offer/Request/Acknowledgement protocol sequence. Ideally, the network interface should be disabled when a

duplicate IP address is detected. Windows informs the user of the duplicate IP address, but does NOT disable the network interface.

If two or more devices on the network have the same IP address, the first device that starts using the duplicate IP address will not detect a problem because it is the only device using that IP address. However, when a second device starts and attempts to use the duplicate IP address, a duplicate IP address error is reported.

To recover from a duplicate IP address condition, eliminate the duplicate IP address condition and then restart the affected devices. For example, an instrument can be restarted (to clear up the error) by disconnecting the Ethernet port long enough to be detected by the Ethernet connection monitoring or by cycling power to the instrument.

## Device Hostname Services

This section introduces some device hostname services, including:

- [Device Hostname Services Overview](#)
- [Dynamic DNS Naming](#)
- [RFC NetBIOS Naming](#)
- [Static DNS Naming](#)

### Device Hostname Services Overview

For ease of operation and communication on a TCP/IP network, you can assign a name to a host, called a **hostname**. However, since TCP/IP must use IP addresses to connect a host to another host, hostnames must be resolved (translated) to an IP address before TCP/IP can use the names.

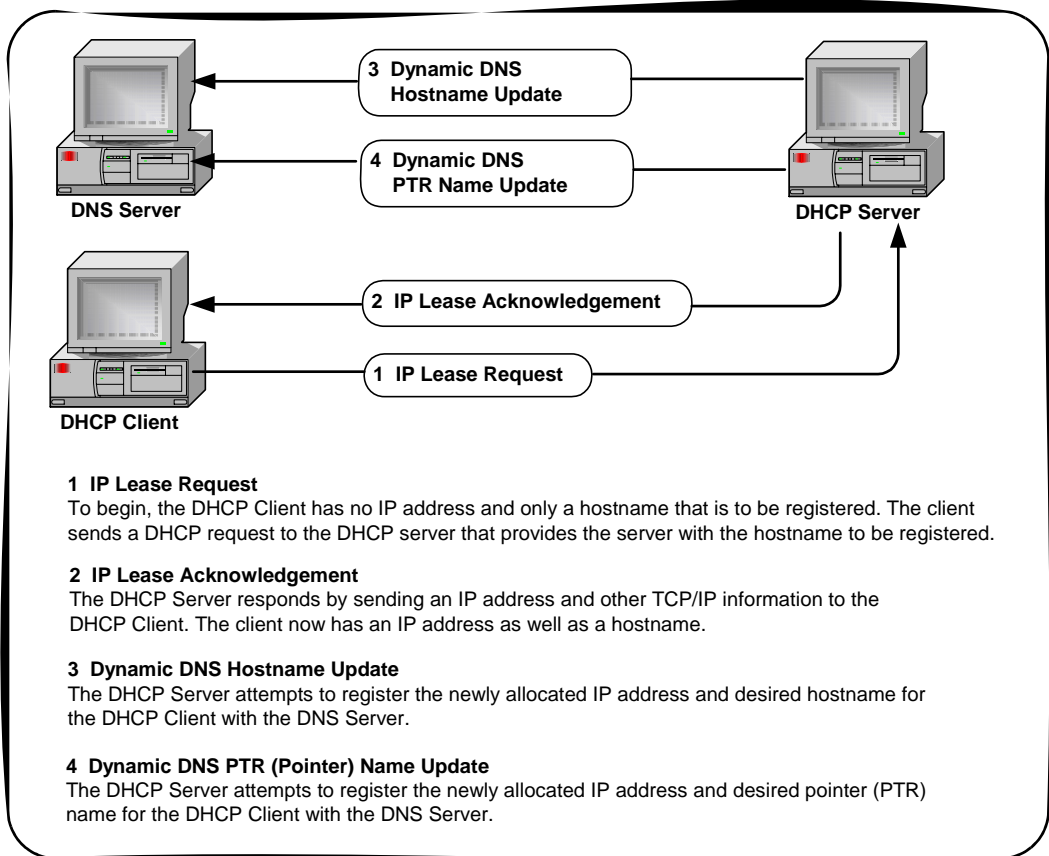
Device hostname services can be divided into two categories: automatic hostname services and manual hostname services. This table shows the applicability of these services to various network topologies.

Network Topology	Automatic Hostname Services		Manual Hostname
	Dynamic DNS	RFC NetBIOS	Static DNS
Site LAN (Direct Connection)	Works with networks that have Dynamic DNS	Works in all network topologies	Works in all network topologies
Site LAN (Router Connection)	Works with networks that have Dynamic DNS	Works in all network topologies	Works in all network topologies
Private LAN (Router Connection)	Works with networks that have Dynamic DNS	Works in all network topologies	Works in all network topologies
Private LAN (Direct Connection)	Typically not used in this topology	Works in all network topologies	Works in all network topologies

## Dynamic DNS Naming

**Dynamic Domain Name System (Dynamic DNS)** is a distributed database of **hostnames** and associated IP addresses on the Internet. All hostnames on the Internet are divided into categories, called **domains**, such as .com, .edu, .org, etc. Dynamic DNS automatically provides hostnames and domain names for devices on networks that support Dynamic DNS. This figure shows a typical process for a DHCP Client to register a hostname and a PTR (pointer) name via a DHCP Server.

If the registration process fails, the DHCP Client hostname may not be assigned to the instrument. If you want to change the hostname after the registration process initiated by the DHCP server, you will probably need to cycle power on the instrument to re-initiate a DHCP request and hostname registration attempt.



**Figure 50** DHCP Client Registering a Hostname and PTR Name

## RFC NetBIOS Naming

**RFC NetBIOS Naming** is a peer-to-peer naming protocol used by Microsoft File/Print Sharing that automatically provides hostnames for devices on networks that support RFC NetBIOS Naming. RFC NetBIOS naming uses a six-step process to resolve an IP address for a specified host name, as shown in the following figure. TCP/IP completes each step

in the sequence shown before returning an error message. RFC NetBIOS naming does require a reboot to force a naming change to take effect.

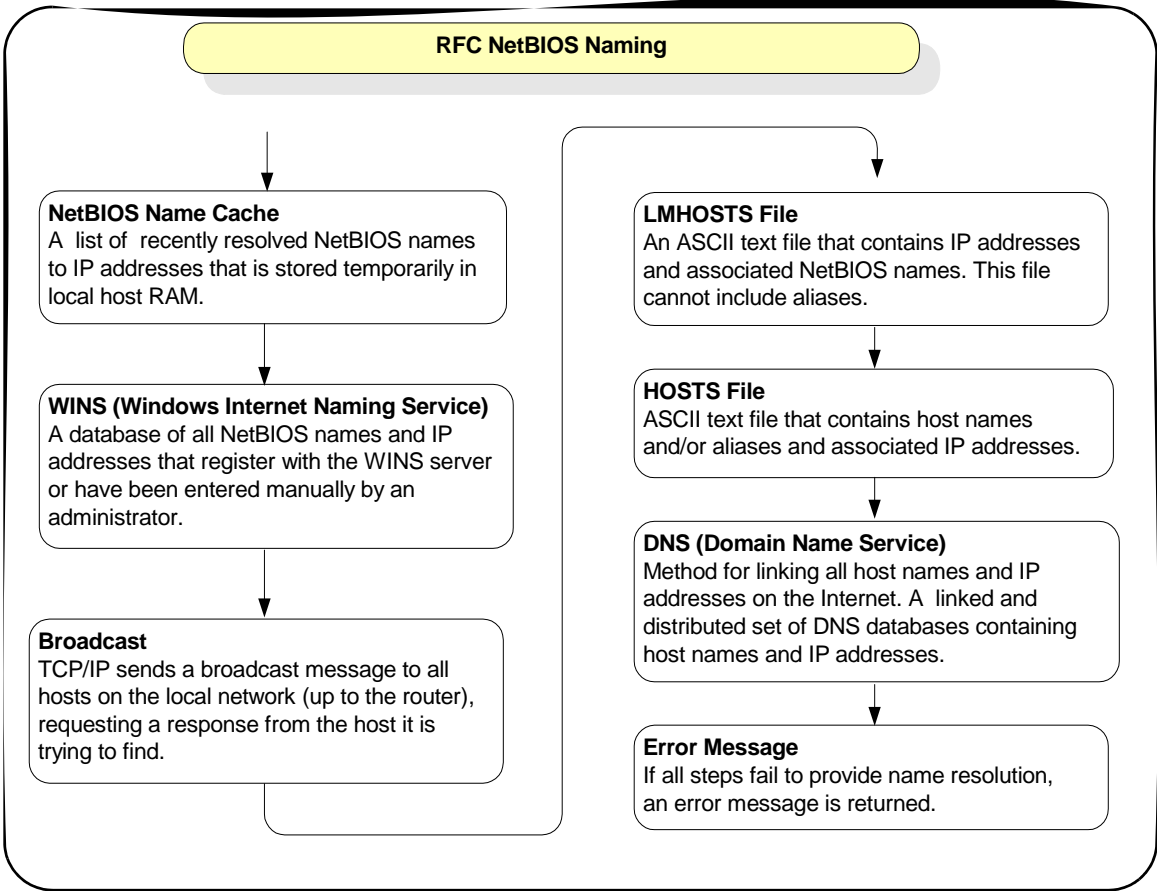


Figure 51 RFC NetBIOS Naming

### Static DNS Naming

In contrast to automatic DNS name resolution, **Static DNS naming** does not require any host functionality to support dynamic methods of hostname resolution. Static DNS Naming

uses a seven-step process to resolve an IP address to a specified hostname. TCP/IP completes each step in the sequence shown before returning an error message. See your instrument's *User's Guide* to determine if your instrument supports Static DNS Naming.

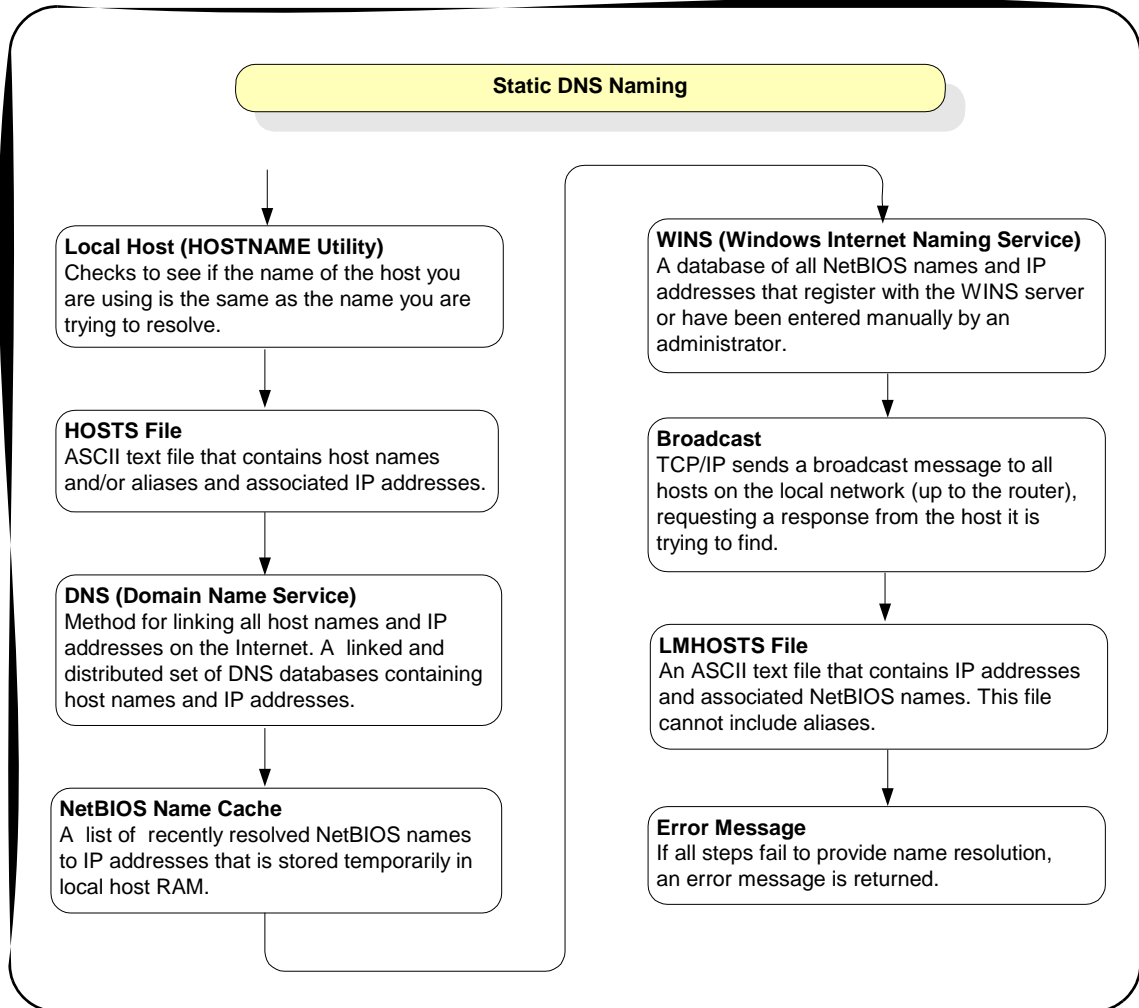


Figure 52 Static DNS Naming



## Configuring Your PC for LAN Operation

This section shows steps to configure your PC, as required, for operation on a private LAN or for operation on site LANs that do not support DHCP, including:

- [Checking PC Settings](#)
- [Installing Network Interface Cards](#)
- [Installing TCP/IP on Your PC](#)
- [Setting PC IP Address](#)
- [Setting PC Hostname](#)

### NOTE

If your PC is part of an existing private LAN or site LAN, you probably do not need to do the steps in this section. If local policies require it, contact your System Administrator before changing PC settings.

---

## Checking PC Settings

To check the current settings for your PC, type **ipconfig /all** from the Command Prompt. The figure below shows a typical display for a Windows 2000 PC. The display for other operating systems may be different. If the settings are correct for your LAN operation, do not do any of the steps in this section. If the settings need to be changed, go to [Installing Network Interface Cards](#).

```

Command Prompt
C:\>ipconfig /all

Windows 2000 IP Configuration

Host Name . . . . . : mypc1
Primary DNS Suffix . . . . . :
Node Type . . . . . : Hybrid
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No
DNS Suffix Search List. . . . . : luld.agilent.com
                                agilent.com

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : luld.agilent.com
    Description . . . . . : Realtek RTL8139(A)-based PCI Fast Et
Ethernet Adapter . . . . . :
    Physical Address. . . . . : 00-10-B5-09-53-30
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes
    IP Address. . . . . : 156.140.104.163
    Subnet Mask . . . . . : 255.255.254.0
    Default Gateway . . . . . : 156.140.104.1
    DHCP Server . . . . . : 156.140.104.102
    DNS Servers . . . . . : 156.140.2.11
                                156.140.2.9
    Primary WINS Server . . . . . : 130.29.152.23
    Secondary WINS Server . . . . . : 130.29.154.21
    Lease Obtained . . . . . : Wednesday, October 02, 2002 4:03:29 AM
    Lease Expires . . . . . : Thursday, October 03, 2002 4:03:29 AM

C:\>_

```

## Installing Network Interface Cards

Network Interface Cards (NIC) provide the hardware interface between your PC and network devices such as routers, hubs, or switches. As required, install NIC(s) in your PC according to the NIC manufacturer's instructions.

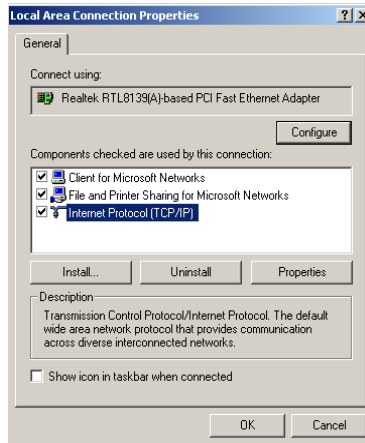
## Installing TCP/IP on Your PC

To configure a network interface card, the TCP/IP protocol must be installed and configured. See the following examples for a Windows 2000 PC or for a Windows XP PC. Modify the steps as required for your operating system.

### Example: Installing TCP/IP (Windows 2000)

- 1 Click **Start > Settings > Network and Dial-up Connections** to display the **Network and Dial-up Connections** dialog box.
- 2 Right-click **Local Area Connection** and then click **Properties** to display the **Local Area Connection Properties** dialog box.

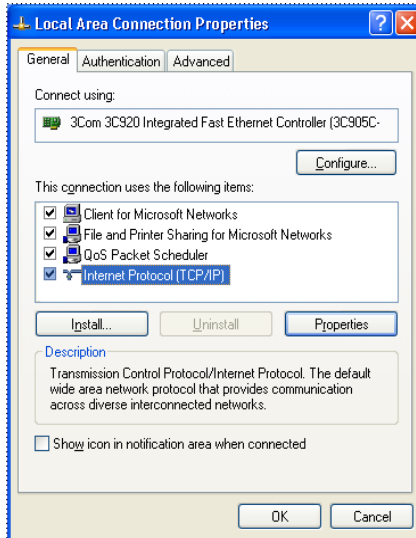
- 3 The General tab should display **Internet Protocol (TCP/IP)**. If not, click **Install...**, then select **Protocol** and click **Add**. Then, select **TCP/IP Protocol** and click **Install**.



### Example: Installing TCP/IP (Windows XP)

- 1 Click **Start > Network > Internet Connections**. From the “or pick a control panel icon”, select **Network Connections**.
- 2 Right-click **Local Area Connection** and then click **Properties** to display the **Local Area Connection Properties** dialog box.

- 3 The General tab should display **Internet Protocol (TCP/IP)**. If not, click **Install**, then select **Protocol** and click **Add**. Then, select **TCP/IP Protocol** and click **Install**.



## Setting PC IP Address

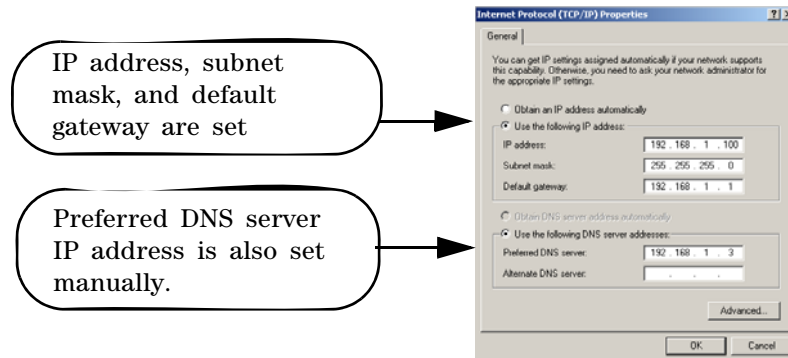
Depending on the LAN capabilities, you can select automatic or manual methods to set the IP address and the DNS Server address on your PC. These examples show how to set an IP address for a Windows 2000 or Windows XP PC. Modify the steps as required for your operating system.

### NOTE

You should *not* change the IP configuration of your PC unless you are sure this is necessary. In almost all cases, the PC should be automatically configured.

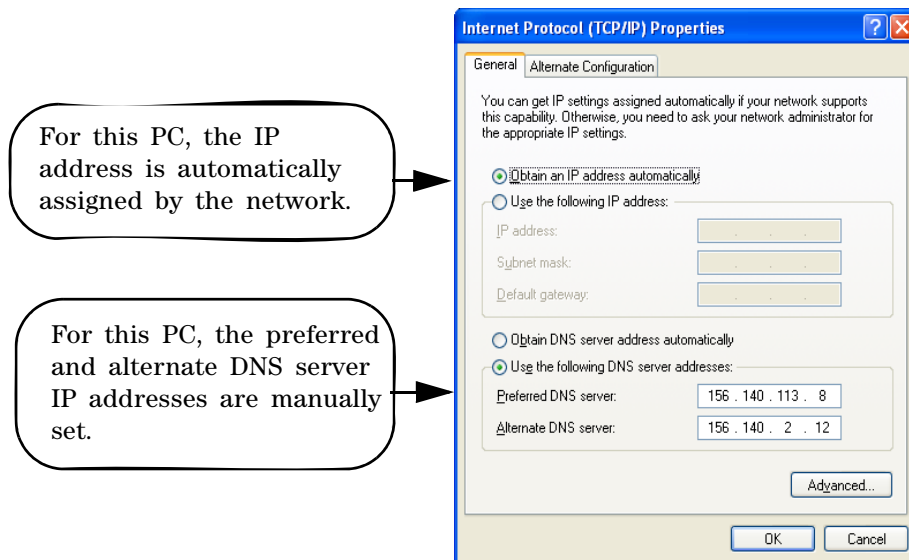
**Example: Setting PC IP Address (Windows 2000)**

- 1 Click **Start** > **Settings** > **Network and Dial-up Connections** to display the **Network and Dial-up Connections** dialog box.
- 2 Right-click **Local Area Connection** and then click **Properties** to display the **Local Area Connection Properties** dialog box.
- 3 From the **General** tab, select **TCP/IP Protocol** and then click **Properties** to display the **Internet Protocol (TCP/IP) Properties** dialog box. Select the method to set the IP address and/or Preferred and Alternate DNS server addresses.
- 4 If you select manual assignments, enter the desired values. When you have finished, click the **OK** box to enable the assignments. This figure shows an **Internet Protocol (TCP/IP) Properties** dialog box to manually assign IP and DNS Server configuration on a Windows 2000 PC.

**Example: Setting PC IP Address (Windows XP)**

- 1 Click **Start** > **Network** > **Internet Connections**.
- 2 From the “or pick a control panel icon”, select **Network Connections**.
- 3 Right-click **Local Area Connection** and then click **Properties** to display the **Local Area Connection Properties** dialog box.
- 4 From the **General** tab, select **TCP/IP Protocol** and then click **Properties**.

- 5 Select **Use the Following IP Address** and then enter the IP address and subnet mask for the network. As required, enter the default gateway and DNS settings. Follow the on-screen instructions and reboot your PC as required.
- 6 This figure shows an **Internet Protocol (TCP/IP) Properties** dialog box to automatically obtain IP addresses and to manually assign DNS Server addresses on a Windows XP PC.

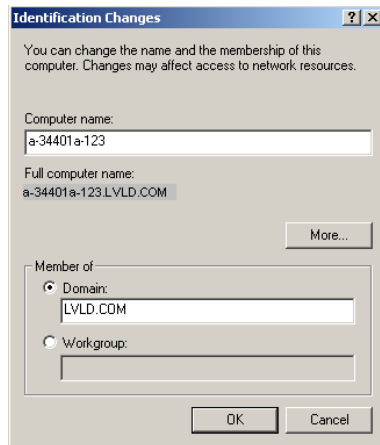


## Setting PC Hostname

As desired, you can set an approved Hostname and a Domain Name (such as *dept.company.com*) for your PC. Examples for Windows 2000 and Windows XP follow. Modify the steps as required for your operating system.

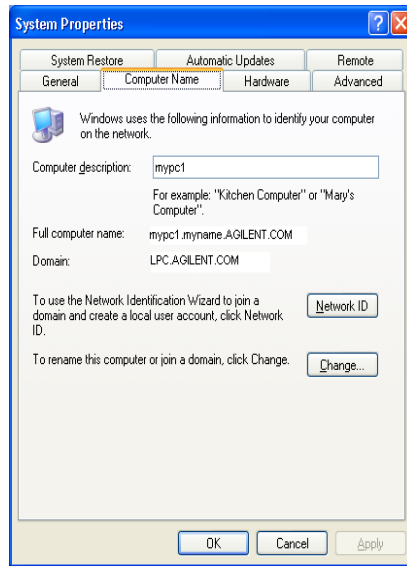
**Example: Setting PC Hostname (Windows 2000)**

- 1 Right-click the **My Computer** icon and then click **Properties** to display the **System Properties** dialog box. Select the **Network Identification** tab and then click **Properties** to display the **Identification Changes** dialog box.
- 2 As required, set/change the Hostname and/or Domain Name. When you have finished, click the **OK** box to enable the assignments. This figure shows an example **Identification Changes** dialog box on a Windows 2000 PC. As desired, you can change the Hostname and/or Domain Name.

**Example: Setting PC Hostname (Windows XP)**

- 1 Right-click the **My Computer** icon and then click **Properties** to display the **System Properties** dialog box.
- 2 Select the **Network Identification** tab and then click **Properties** to display the **Identification Changes** dialog box.
- 3 As required, set/change the Hostname and/or Domain Name. When you have finished, click the **OK** box to enable the assignments.

- 4 This figure shows an example **Identification Changes** dialog box on a Windows XP PC. As desired, you can change the hostname and/or domain name.







## 8 Troubleshooting Guidelines

This chapter includes:

- Troubleshooting LAN Interfaces
- Troubleshooting USB Interfaces
- Troubleshooting GPIB Interfaces



## Troubleshooting LAN Interfaces

This section shows suggested troubleshooting steps for LAN interfaces, including the following items. For information on TCP/IP networks, see [TCP/IP Network Basics](#).

- [LAN Troubleshooting Overview](#)
- [Hardware Checks](#)
- [Communication Checks](#)
- [Web Browser Checks](#)

### LAN Troubleshooting Overview

LAN troubleshooting guidelines follow. However, the first step is to check for normal operation, as shown in [Checking for Normal Operation](#).

**Checking for Normal Operation** Before beginning LAN troubleshooting steps, you should first check to see what symptoms appear when you attempt to connect to instruments via the LAN. To see if your PC has network access to an instrument, open the Web browser on the PC and type in the instrument's IP address (or hostname, if known) on the web browser's address line.

If the instrument's **Welcome Page** appears, this PC has network connection to the instrument. Repeat this step for each instrument on the network. An example for the Agilent 33220A follows. If normal operation cannot be established, go to [Troubleshooting Flowchart](#) for a suggested sequence of troubleshooting steps.

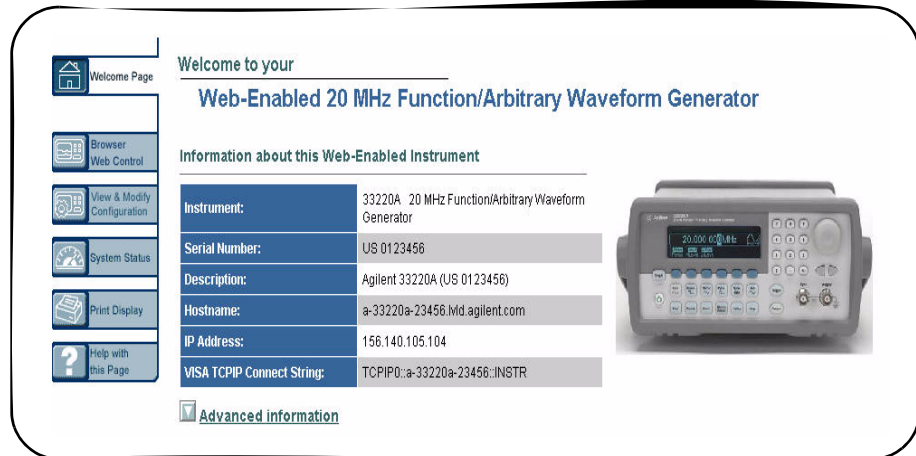
**NOTE**

There are additional troubleshooting tips in the *IO Libraries Suite Online Help*.

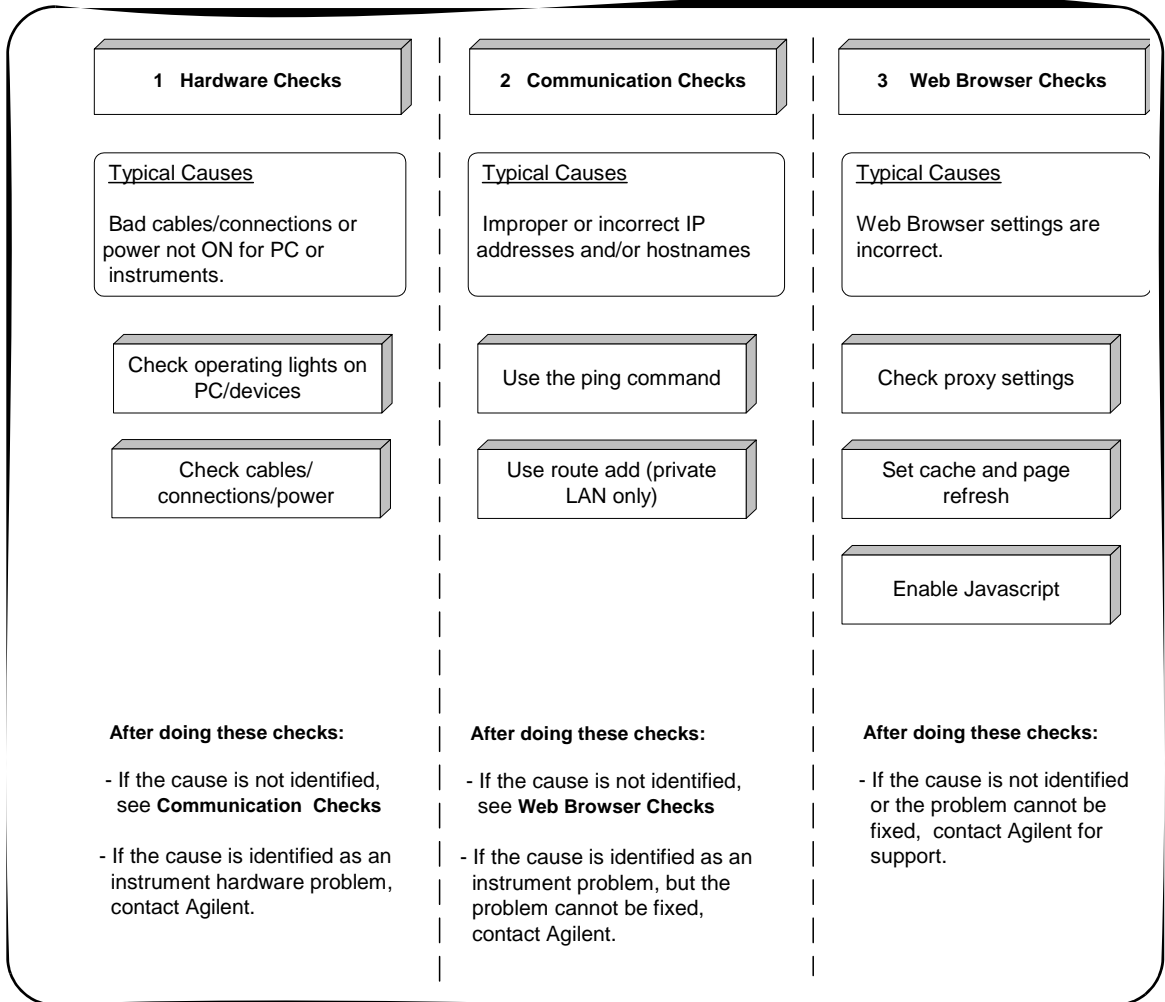
**Example: Using a Web Browser**

Since the 33220A is Web-enabled, to display the 33220A **Welcome Page**, first determine the 33220A IP address from the instrument’s front panel display. Next, open your web browser and type ‘**http://<33220A IP Address>**’, where **<33220A IP Address>** is the IP address displayed on the front panel. Then, press **Enter** to display the 33220A **Welcome page**.

For example, if the current IP address is 169.254.3.2, typing **http://169.254.3.2** and pressing **Enter** displays the 33220A **Welcome** page. This figure shows a portion of an example **Welcome** page display for the Agilent 33220A.



**Troubleshooting Flowchart** If normal operation cannot be established, see this flowchart for suggested troubleshooting steps. We suggest you start at [Hardware Checks](#).



**Hardware Checks** The first step in troubleshooting LAN networks is to check the hardware setup for all devices connected to the LAN. This table shows some possible causes and corrective actions.

After you have made the hardware checks and taken corrective actions as required, again check for normal operation as shown in [Checking for Normal Operation](#). If normal operation has not been established, go to [Communication Checks](#).

Possible Cause(s)	Corrective Action(s)
PC Suspended or Network Interface Card (NIC) not operational.	Verify that the PC is not in a Suspended power state. If the lights on the NIC are ON, the NIC is probably operational.
Ethernet Hub or Switch not ON or not operational.	Verify that the hub/switch is powered ON. If the hub/switch lights are ON, the hub/switch is probably operational.
Instruments are not ON or are not operational.	Verify that all instruments are powered ON. If the power-on light is ON, the instrument is probably operational. Also, check the NIC lights on the instruments. If the NIC lights are ON, the hardware LAN connection to the instrument is OK.
Cables are not properly connected	Check LAN cable connections between PC, hub/switch, and instruments. Tighten connectors or replace cables as required.

## Communication Checks

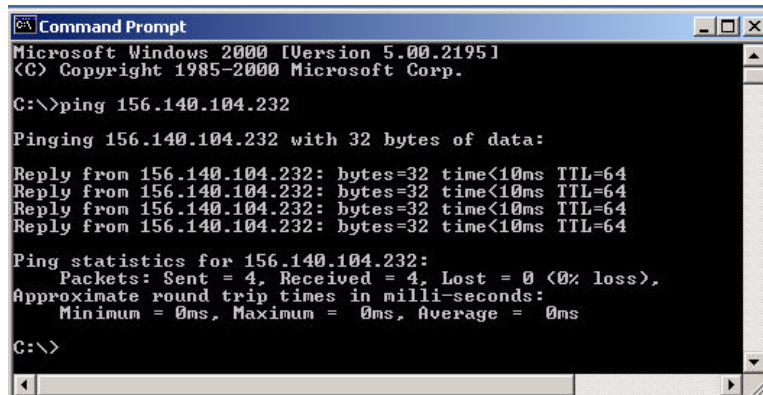
If there are no hardware problems, but normal communication has not been established, the next step is to make communication checks using the **ping** command. In addition, for private LANs only, you may need to use the **route add** command to establish communication. If you can establish communication using these checks, but the web browser display is not correct, go to [Web Browser Checks](#).

**Using the ping Command** An example to verify communication using the **ping** command follows.

**NOTE**

If you cannot verify communication to connected instruments using the ping command, try using **ping** to a device at a known address. If this is successful, the problem is likely with the instrument (for site LANs) or with the hub/switch or instruments (for private LANs).

**Example: Verifying Communication Using the ping Command** For example, on Windows 2000 to send a **ping** command from your PC to an instrument click **Start > Programs > Accessories > Command Prompt**. When the **Command Prompt** window opens, type **ping <IP\_address>**, where **<IP\_address>** is the IP address of the instrument, and then press the **Enter** key. A typical display follows, where **ping** was successful.



```
Command Prompt
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\>ping 156.140.104.232

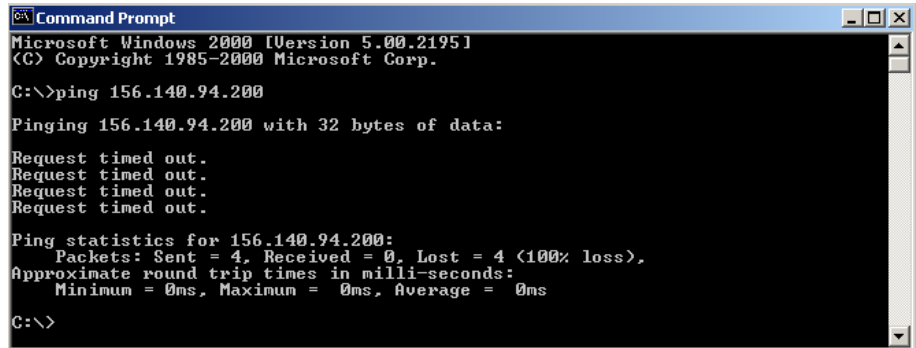
Pinging 156.140.104.232 with 32 bytes of data:

Reply from 156.140.104.232: bytes=32 time<10ms TTL=64
Reply from 156.140.104.232: bytes=32 time<10ms TTL=64
Reply from 156.140.104.232: bytes=32 time<10ms TTL=64
Reply from 156.140.104.232: bytes=32 time<10ms TTL=64

Ping statistics for 156.140.104.232:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

A typical display follows, where **ping** was not successful.



```

C:\> Command Prompt
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\> ping 156.140.94.200

Pinging 156.140.94.200 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 156.140.94.200:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>

```

**Using the route add Command** If you are having difficulty with a private LAN (using a hub, switch, or even a simple crossover cable), a possible cause for no communication between the PC and connected instruments is that the PC has not recognized that it is connected to two different subnets. (See [Local and Remote Networks](#) for details.)

One way to put the PC and the instrument on the same subnet is to use **route add <Instrument IP Address> <PC IP Address>** from the **Command Prompt**. If this action does not correct the problem, go to [Web Browser Checks](#).)

#### NOTE

If your devices are connected to site LAN, do not use the **route add** command.

#### Example: Using the route add Command

For example, on Windows 2000 to send a **route add** command, click **Start > Programs > Accessories > Command Prompt**. When the **Command Prompt** window opens, type **route add <Instrument IP Address> <PC IP Address>**, where **<Instrument IP\_address>** is the IP address of the instrument and **<PC IP Address>** is the IP address of the instrument. Then press the **Enter** key.

For example, if the IP address of your PC is 155.139.103.152 and the instrument address is 169.254.58.10, the two devices are probably on different subnets. To add the instrument to the subnet for the PC, use one of the following:

```
! route is lost when the PC is rebooted  
route add 169.254.58.10 155.139.103.152
```

or

```
! route persists when the PC is rebooted  
route -p add 169.254.58.10 155.139.103.152
```

## Web Browser Checks

If all hardware connections are OK and you have been able to establish communication between your PC and instruments, but the web browser display is incorrect or the web browser display does not appear at all, the problem could be that the web browser settings are incorrect.

Suggested steps follow to check your web browser settings, including the following items. If doing these steps does not correct the problem, contact Agilent for support. See [Contacting Agilent](#) for addresses.

- Checking Proxy Settings
- Setting Cache and Page Refresh
- Enabling Javascript

**Checking Proxy Settings** A **proxy** is a service running on a computer that takes an information request from another computer and passes the request on as if the request were its own. Typically, many companies use proxies to control access from their internal network to the external Internet.

In some situations, you may not be able to communicate with the instrument if you are using a proxy server. Typically, the web page will time out because the requested



address is not accessible by the network, although the correct IP address or hostname is entered into the web browser address field.

To fix this communication problem between the web browser and the instrument web server, the browser must be informed that any requests to the instrument should **not** utilize a proxy.

#### **Example: Setting Proxy Notification (Internet Explorer 5.0 and above)**

Example steps check proxy settings for Internet Explorer versions 5.0 and above follow. The steps are for Windows 2000. Modify the steps as required for other operating systems.

- 1** Click **Start > Settings > Control Panel** to display the **Control Panel**
- 2** From **Control Panel**, double-click **Internet Options** to display the **Internet Properties** dialog box
- 3** From the **Internet Properties** dialog box, select the **Connections** tab
- 4** Click the **LAN Settings...** button under **Local Area Network (LAN) settings**
- 5** Under **Proxy server**, if “**Use a proxy server**” is checked, click the **Advanced...** button
- 6** Under **Exceptions**, add the IP address of the instrument or a fully-qualified domain name (such as **myinstr.example.com**) to the list box called “**Do not use proxy server for addresses beginning with:**” Use a semicolon (;) to separate multiple entries.
- 7** Click the **OK** button on the bottom of the **Internet Options** dialog box to make changes active.

**Setting Cache and Page Refresh** For faster performance, most web browsers cache web pages. If a page is cached, an image of the web page is stored locally. When you navigate

to a page that has already been viewed, the browser will load the page from its cache rather than loading it from the network.

Usually, this process works well for static web pages. However, this process may cause problems for 'dynamically' changing web pages. This problem may occur when you are navigating using the browser's forward/back/refresh options.

Doing this usually causes the browser to first look in its cache to see if the page exists. If the page is cached, the browser displays the page from the cache instead of going to the network to update changes. Since instrument web pages are dynamic in nature, the cache may contain content that is outdated. If you see a problem where dynamic pages are not being updated automatically, the browser should be instructed to check for newer versions of a web page on every page visit.

### **Example: Cache and Page Refresh (Internet Explorer 5.0 and above)**

Example steps for cache and page refresh for Internet Explorer versions 5.0 and above follow. The steps are for Windows 2000. Modify the steps as required for other operating systems.

- 1** Click Start > Settings > Control Panel to display the Control Panel.
- 2** From **Control Panel**, double-click **Internet Options** to display the **Internet Properties** dialog box.
- 3** From the **Internet Properties** dialog box, select the **General** tab.
- 4** Under **Temporary Internet files**, click the **Settings...** button to display the **Settings** dialog box.
- 5** Under **Check for newer versions of stored pages**, click the **"Every visit to the page"** radio button (**"Automatically"** should be okay for Internet Explorer 5.5 and greater).
- 6** Click the **OK** button on the **Settings** page to accept the change.

- 7 Click the **OK** button on the bottom of the **Internet Options** dialog box to make changes active.

**NOTE**

If the previous steps do not resolve the problem, you may want to clear the cache. To do this, repeat steps 1 - 4. Then, in the **Temporary Internet files**, click the **Delete Files...** button to display the **Delete Files** dialog box. Click **OK** to delete the files and clear the cache.

**Enabling Javascript** Instruments that are Web-enabled may generate web pages that depend on Javascript and Frames. For proper operation, you may need to enable Javascript.

**Example: Enabling Javascript (Internet Explorer 5.0 and above)**

Example steps for enabling Javascript for Internet Explorer versions 5.0 and above follow. The steps are for Windows 2000. Modify the steps as required for other operating systems.

- 1 Click **Start > Settings > Control Panel** to display the Control Panel.
- 2 From **Control Panel**, double-click **Internet Options** to display the **Internet Properties** dialog box.
- 3 From the **Internet Properties** dialog box, select the **Securities** tab.
- 4 Click the **Custom Level...** button.
- 5 Under **Settings**, scroll down to the category called **Scripting**.
- 6 Click the **Enable** radio button under **Active Scripting**.
- 7 Click the **Enable** radio button under **Scripting of Java applets**.
- 8 Click the **OK** button on bottom of page to accept security settings.
- 9 When the **Warning!** dialog box appears, click the **YES** button when asked “Are you sure you want to change the security settings for this zone?”
- 10 Click the **OK** button on the bottom of the **Internet Options** dialog box to make the changes active.

## Troubleshooting USB Interfaces

This section shows suggested troubleshooting steps for Universal Serial Bus (USB) interfaces, including:

- [USB Troubleshooting Overview](#)
- [USB Hardware Checks](#)
- [USB Software Checks](#)

### NOTE

The troubleshooting procedures in this section are primarily oriented toward USB instruments and the Agilent IO Libraries Suite. Consult your instrument's User's Guide for troubleshooting details for the instrument.

---

### USB Troubleshooting Overview

This figure shows a typical USB interface with a PC and a USB instrument. A suggested troubleshooting flowchart for USB instruments and the Agilent IO Libraries Suite follows. We suggest you start at [USB Hardware Checks](#) and then go to [USB Software Checks](#), as required.

USB Hardware Checks	USB Software Checks
<p><u>Typical Causes</u></p> <p>No power on USB bus or hub or device turned off by Windows Plug and Play Manager.</p>	<p><u>Typical Causes</u></p> <p>Agilent IO Libraries Suite not installed or USB drivers not installed or improper IO Libraries Suite operation.</p>
<p>Check cable connections/damage</p>	<p>Verify Agilent IO Libraries Suite installation</p>
<p>Check PC/instrument functions</p>	<p>Verify USB driver installation</p>
<p>Check device manager</p>	<p>Check IO Control operation</p>
<p><b>After doing these checks:</b></p> <ul style="list-style-type: none"> <li>- If the cause is not identified, see <b>Software Checks</b>.</li> <li>- If the cause is identified as a hardware problem, contact Agilent to return the device.</li> </ul>	<p><b>After doing these checks:</b></p> <ul style="list-style-type: none"> <li>- If the cause is not identified or the problem cannot be fixed, contact Agilent for support.</li> </ul>

## USB Hardware Checks

This section gives guidelines to make hardware troubleshooting checks for connected USB instruments. Start your troubleshooting sequence by performing the following hardware checks. If performing the hardware checks do not correct the problem, go to [USB Software Checks](#).

**Check Cable Connections/Damage** Check all USB cables for good connection to the USB connector on your PC, to all USB hubs (if installed) and the USB cable connections to connected USB instruments. An improperly attached USB connector can cause the interface to malfunction.

Check all USB cables for cuts/damage and check for bent/misaligned/crushed connector pins. Replace cables as required. If these actions do not solve the problem, try disconnecting and reconnecting (or replacing) USB cables. If the cable checks do not solve the problem, go to [Check PC/Instrument Functions](#).

**Check PC/Instrument Functions** Verify that all USB hubs, and all connected USB instruments are functional and are powered ON. Verify that host computer is not in a Suspended power management state. If these actions do not solve the problem, reboot the PC. If this does not solve the problem, go to [Check Device Manager](#).

**Check Device Manager** You can use the Windows Device Manager to reinstall the USB instrument driver, as required. For example, with Windows 2000, go to **Control Panel** by selecting **Start > Settings > Control Panel**. Then, select **System > Hardware > Device Manager**. From **Device Manager**, select the **USB Test and Measurement Device** and then click **Properties**.

Tab to **Driver** and click **Reinstall Driver**. This will allow the Windows Plug and Play Manager to begin searching for a driver for the instrument. Since Device Manager may have

disabled the instrument, click **Enable** to restart the instrument.

### NOTE

Windows may place a misbehaving USB device in an “Unknown device” category. To see if this has happened, change the View in **Device Manager** by selecting **View > Show Hidden Devices**. Next, right-click the **Universal Serial Bus Controllers Unknown device** and select **Uninstall**. Then, select **Device Manager > Action > Scan** for hardware changes.

## USB Software Checks

This section provides guidelines to make software troubleshooting checks for connected USB instruments. If you have not already done so, we suggest you start your troubleshooting sequence by performing the hardware checks in [USB Hardware Checks](#). If performing the hardware and then the software checks does not correct the problem, contact Agilent for instructions. See [Contacting Agilent](#) for addresses.

### Verify Agilent IO Libraries Suite Installation

- 1 **Check Agilent IO Libraries Version.** If a version of the Agilent IO Libraries or IO Libraries Suite has been installed, a blue IO icon is normally displayed on the Windows taskbar (on the lower right side of the screen).



**Figure 53** IO Control Icons for IO Libraries and IO Libraries Suite

- If the IO icon is displayed, click the icon and click **About Agilent IO Control** to display the version. The version must be M.01.00 or greater. (The next version of the IO Libraries after “M” discontinued this alphabetic naming convention, and uses *IO Libraries Suite 14.0* to designate the version immediately following M.01.01.)
- If the IO icon is not displayed, a version may still be installed. To check this, click **Start > Programs** and look for the **Agilent IO Libraries** or the **Agilent IO Libraries Suite** program group.
- If this group is displayed, click **Agilent IO Libraries > Utilities > IO Control** or **Agilent IO Libraries Suite > Utilities > IO Control** to display the IO icon. Then, click the icon and click **About Agilent IO Control** to display the installed version (must be M.01.00 or greater).
- If neither the IO icon nor the Agilent IO Libraries Suite program group is displayed, no Agilent IO Libraries are installed. In this case, or if the installed version is not M.01.00 or greater, you must install the newer version (see substep 2 following).

**2 Install Agilent IO Libraries Suite (as Required).** If M.01.00 or greater of the Agilent IO Libraries is not installed on your PC, use this substep. Otherwise, skip to [Verify USB Driver Installation](#).

- Remove the USB cable from the USB port on the PC.
- Uninstall the IO Libraries by using **Add/Remove Programs** in the **Control Panel**.
- Install IO Libraries Suite from the *Automation-Ready CD with Agilent IO Libraries Suite*. (Go to [Installing I/O Software](#) for more information.)
- Connect the USB cable to the PC.
- If this does not correct the problem, go to [Verify USB Driver Installation](#).



### Verify USB Driver Installation

**3 Check for USB Driver Files.** Files are listed in their default Windows directories. Verify that the driver files are installed. If they are installed, go to [Check Agilent IO Control Operation](#). If not, go to substep 2 following.

**NOTE**

USB ports ONLY operate properly on Windows XP when the latest Microsoft USB drivers from Windows XP Service Pack 1 or greater have been properly installed. However, upgrading your XP PC to SP1 or SP2 will not automatically update the USB drivers. See [www.agilent.com/find/inst\\_drivers](http://www.agilent.com/find/inst_drivers) to verify if correct USB drivers are installed and to install correct drivers as required.

Windows 2000	
Program Files	C:/Program Files/Agilent/IO Libraries Suite/drivers/agusb48832.dll
Driver Files	C:/Winnt/system32/drivers/ausbtmc.sys
.inf Files	C:/Winnt/inf/ausbtmc.inf
Windows XP	
Program Files	C:/Program Files/Agilent/IO Libraries Suite/drivers/agusb48831.dll
Driver Files	C:/Windows/system32/drivers/ausbtmc.sys
.inf Files	C:/Windows/inf/ausbtmc.inf

**4 Uninstall/Reinstall the Agilent IO Libraries Suite.** If the driver files are not found, uninstall the Agilent IO Libraries Suite by inserting the *Automation-Ready CD with Agilent IO Libraries Suite* and following the instructions to install the libraries. The installation program will detect the existence of the IO Libraries Suite and offer you the option to remove it. Then reinstall the IO Libraries Suite.

**Check Agilent IO Control Operation** When the Agilent IO Libraries Suite was installed, the Agilent IO Control was started. When the IO Control is active, it is displayed as a blue **IO** icon on the Windows taskbar. If the IO Control is deactivated, I/O applications may be unable to open sessions or to communicate with instruments.

By default, the Agilent IO Control is always active after the Agilent IO Libraries Suite is installed and the blue **IO** icon is displayed. However, the IO Control may be active even though the blue **IO** icon is not displayed. There are two ways that the blue **IO** icon can be hidden:

- **Clicking Hide Agilent IO Control.** Clicking the blue **IO** icon and then clicking **Hide Agilent IO Control** hides the blue **IO** icon, but does *not* deactivate the Agilent IO Control. (You can also do this from the Connection Expert **View** menu.)
- **Clicking Exit.** Clicking the blue **IO** icon and then clicking **Exit** causes a dialog box to appear that asks you if you want to terminate the Agilent IO Control. Clicking **Yes** hides the blue **IO** icon *and* deactivates the Agilent IO Control.

If the blue **IO** icon is not displayed, either the blue **IO** icon display has been turned off and/or the Agilent IO Control is not active. In this case, click **Start > Programs > Agilent IO Libraries Suite > Utilities > IO Control** to re-start the Agilent IO Control and to display the blue **IO** icon.

## Troubleshooting GPIB Interfaces

This section shows suggested troubleshooting steps for an Agilent 82350 GPIB interface, including:

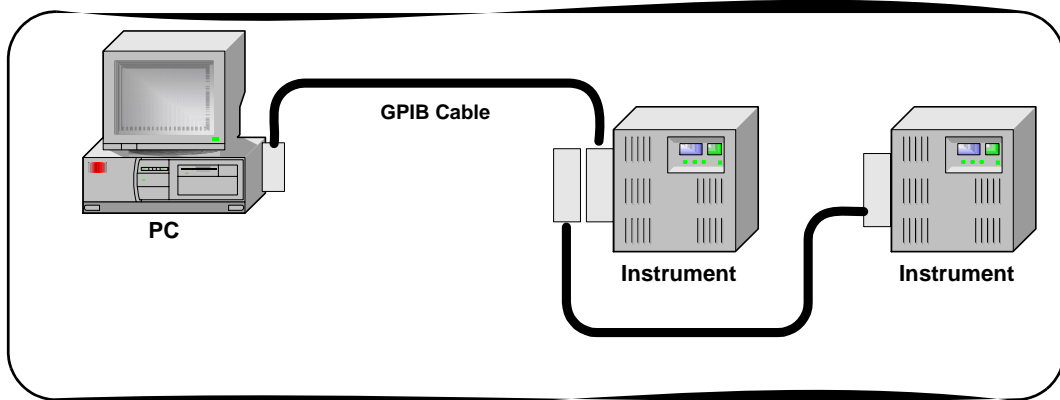
- GPIB Troubleshooting Overview
- GPIB Hardware Checks
- GPIB Software Checks
- Agilent IO Libraries Suite Checks

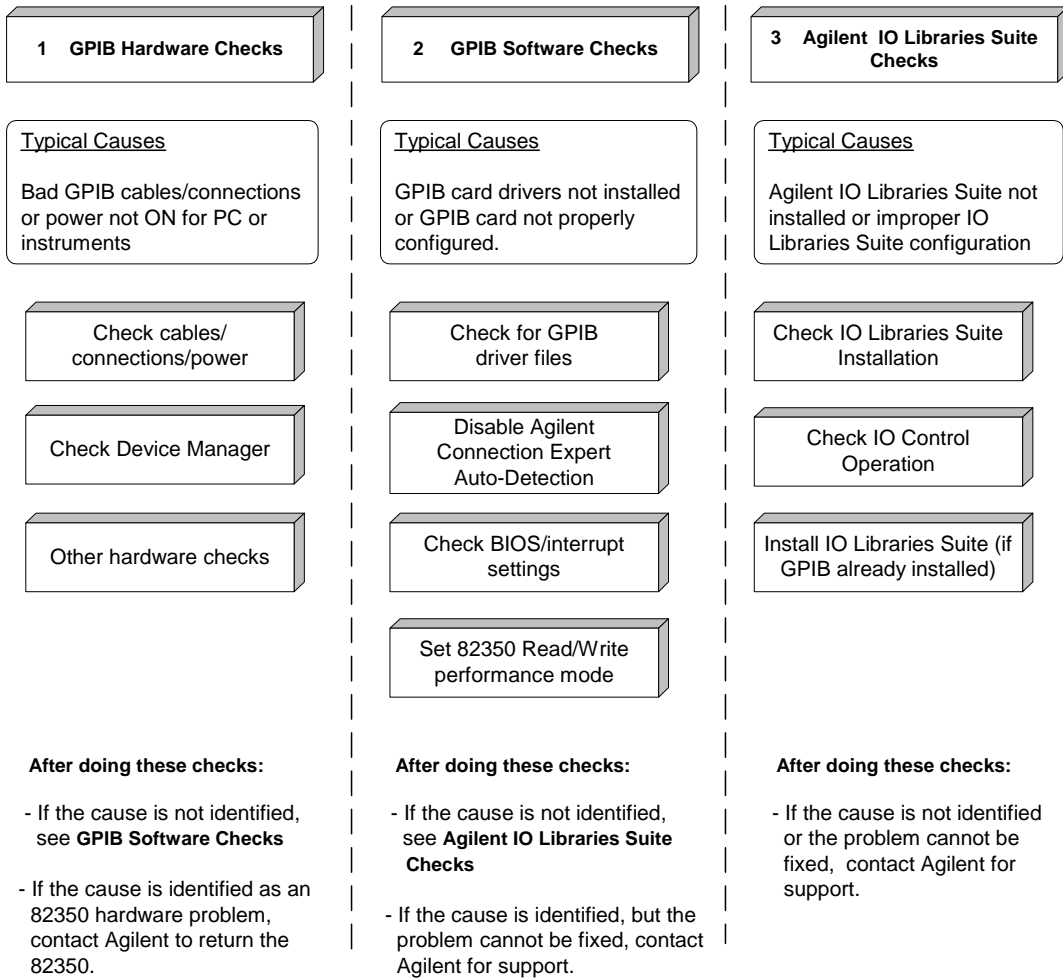
### NOTE

The troubleshooting procedures in this section are primarily oriented toward an Agilent 82350 GPIB card and the Agilent IO Libraries Suite. Consult your instrument's User's Guide for troubleshooting details for the instrument.

## GPIB Troubleshooting Overview

This figure shows a typical GPIB system with a PC and two GPIB instruments. A suggested troubleshooting flowchart for GPIB cards, installed instruments, and the Agilent IO Libraries Suite follows. We suggest you start at Step 1 and then go to Step 2 and then to Step 3, as required.





## GPIB Hardware Checks

This section gives guidelines to make hardware troubleshooting checks for the GPIB card (such as an Agilent 82350), including:

- [Check Cables/Connections/Power](#)
- [Check Device Manager](#)
- [If Sound Card Does Not Work](#)
- [If Data Transfers to Devices Fail](#)
- [If Connection Expert Finds 82350 Card with Serial Number fffffff](#)

Start your troubleshooting sequence by performing the following hardware checks. If the hardware checks do not solve the problem, see [GPIB Software Checks](#).

**NOTE**

There are no user-serviceable parts for the 82350. If you suspect a hardware failure for the 82350, contact Agilent for instructions to return the unit. See [Contacting Agilent](#) for telephone numbers/web site address.

- 1 Check GPIB Cable Connections.** Check all GPIB cables for good connection to the GPIB connector on the 82350 installed in your PC and the GPIB cable connections between all connected GPIB instruments. An improperly attached GPIB connector can cause the bus to malfunction.
- 2 Check GPIB Cables for Damage.** Check all GPIB cables for cuts/damage and check for bent/misaligned/crushed connector pins. Replace cables as required.
- 3 Disconnect/Reconnect GPIB Cables.** If Steps 1 and 2 do not solve the problem, try disconnecting and reconnecting (or replacing) GPIB cables.
- 4 Check PC/Instrument Power-on.** Verify that the PC and all connected GPIB instruments are functional and are powered ON. Verify that host computer is not in a Suspended power management state.
- 5 Reboot Your PC.** If doing Steps 1, 2, 3 or 4 does not solve the problem, reboot the PC. If this does not solve the problem, go to [Check Device Manager](#).

**Check Device Manager** You can use the Windows Device Manager to reinstall the 82350 or equivalent, as required. For example, with Windows 2000, go to **Control Panel** by selecting **Start > Settings > Control Panel**.

Then, select **System > Hardware > Device Manager**. From Device Manager, select **82350** and then **Properties**. Tab to **Driver** and click **Reinstall Driver**. This will allow the Windows Plug and Play Manager to begin searching for a driver for the 82350. Since Device Manager may have disabled the 82350 device, click **Enable** to restart the 82350. If this does not resolve the problem, go to [GPIB Software Checks](#).

If your sound card stops working after configuring interfaces, disable the auto-detection process in Connection Expert. Click **Tools > Options...** . Then de-select **Automatic discovery or refresh of I/O resources** under the **Configuration Settings** tab.

**If Data Transfers to Devices Fail** If devices attached to your ISA (82341) GPIB card are detected, but data transfers to those devices fail, this is usually caused either by an IRQ conflict with another card or by your system having no IRQs available for this card. If no IRQs are available, the IRQ value shown in Connection Expert will be -1.

Try changing the IRQ setting to other values. You must reboot after changing the value to test it. If none work, you will have to free up other IRQs by disabling devices such as serial ports or sound cards.

### **If Connection Expert Finds 82350 Card with Serial Number ffffffff**

If Connection Expert reports finding an 82350 card with serial number ffffffff, this is typically caused by a Windows system configuration that results in failure to properly configure PCI cards. In this case, upgrade your system BIOS to the latest version. Note that even new computers tend to have new BIOS available.

## GPIB Software Checks

This section provides guidelines for 82350 software checks, including:

- [Check for 82350 Driver Files](#)
- [Check Agilent 488 Settings](#)
- [Disable Connection Expert Auto-Detection](#)
- [Check BIOS/Interrupts Settings](#)
- [Set 82350 Read/Write Performance Mode](#)

**Check for 82350 Driver Files** After installing the Agilent IO Libraries Suite, check for installed 82350 driver files.

**1 Check for 82350 Driver Files.** Files are listed in their default directories.

Windows 2000	
Program Files	<i>C:/Program Files/Agilent/IO Libraries Suite/drivers/ag350i32.dll</i>
Driver Files	<i>C:/Winnt/system32/drivers/agt82350.sys</i>
.inf Files	<i>C:/Winnt/inf/agtgpib.inf</i>
Windows XP	
Program Files	<i>C:/Program Files/Agilent/IO Libraries Suite/drivers/ag350i31.dll</i>
Driver Files	<i>C:/Windows/system32/drivers/agt82350.sys</i>
.inf Files	<i>C:/Windows/inf/agtgpib.inf</i>

**2 Uninstall/Reinstall the Agilent IO Libraries Suite.** If the driver files are not found, uninstall the Agilent IO Libraries Suite by inserting the *Automation-Ready CD with Agilent IO Libraries Suite* and following the instructions to remove the libraries. Then, follow the instructions to re-install the libraries.

**Check Agilent 488 Settings** If you are running a program that was written to use NI-488.2, Agilent 488, or another vendor's 488-compatible library, and that program is not operating correctly, check Agilent 488 parameters:

- 1 For Agilent 488 or NI-488.2 code to work with Agilent GPIB interfaces, the Agilent 488 library must be enabled. Click **Tools > Agilent 488...**, then select the check box labeled **Enable Agilent GPIB cards for 488 programs**. This setting allows correct operation of 488 programs with both Agilent and third-party GPIB interfaces (even simultaneously).

If you receive errors from third-party tools (such as National Instruments Measurement & Automation Explorer) when Agilent 488 is enabled, disable Agilent 488, use the third-party tool, then re-enable Agilent 488.

- 2 Most NI-488.2 programs are written to expect the GPIB card at board number 0 (zero). The default board numbers for Agilent GPIB interfaces begin with zero, but *for Agilent GPIB interfaces, the board number is set independently from the VISA interface ID*, so please note that you may need to set the board number explicitly to work with your program. To set the board number, select the GPIB interface (the icon *above* your GPIB instruments) in Connection Expert's explorer view, then click the **Agilent 488 Properties...** button in the detail (right) pane of the Connection Expert window and set the board number in the resulting dialog box.

**Disable Connection Expert Auto-Detection** If your PC sound card stops working after configuring interfaces, disable the auto-detection process in Connection Expert, as the auto-detection process in Connection Expert can cause sound and other cards to stop responding. To disable the auto-detection process, click **Tools > Options...** . Then de-select **Automatic discovery or refresh of I/O resources** under the **Configuration Settings** tab.

**Check BIOS/Interrupts Settings** If Connection Expert reports finding an 82350 card with Serial Number ffffffff, this is typically caused by PCI cards not being configured properly. Try the following steps. If these steps do not work, remove and re-install the 82350 and then reconfigure the card.

- 1 **Check BIOS Setting.**



- 2 **Upgrade your system BIOS to the latest version.** New computers oftentimes have newer BIOSs available.
- 3 **If your computer locks up or freezes after installing.** This is typically caused by interrupt conflicts with other drivers in the system. PCI allows sharing of IRQs, but this also means the drivers for cards which share an IRQ must be handling interrupt chaining properly.

When the IRQ is asserted, the OS calls each ISR in turn until one of them returns TRUE (meaning that it handled the interrupt). The ISR's responsibility is to correctly return TRUE if its device was interrupting or FALSE if not. Drivers that return TRUE, even though they did not service the interrupt, will cause problems. Try the following steps to perform driver workarounds:

- Upgrade the drivers for devices sharing an IRQ with Agilent, including, but not limited to, your video drivers, your LAN drivers, Agilent IDE and/or SCSI drivers, and your sound drivers.
- 4 **Reconfigure Your PC.** Configure your PC so as to not share IRQ lines. Many PCI cards have bugs when sharing IRQ lines. You may or may not be able to do this on all PCs. Many PCs can be configured using the setup option when the PC is first booting.
  - 5 **Upgrade your system BIOS.** New computers may have a newer BIOS available. When installing a new system BIOS:
  - 6 **Make sure the BIOS Installed O/S setting is set correctly.** This determines what software will configure all the *VXIplug&play* cards in your system. Either the BIOS or the O/S can perform the task of querying all the cards to determine their resource needs, picking a valid configuration for all these cards, and telling the cards what their actual resource settings are.
    - If the BIOS **Installed O/S** is set to **Windows 95, Windows 98, Windows 2000** or **Running a PnP O/S**, the BIOS will not perform this task and will leave it to the O/S to do this. If the **Installed O/S** is set to **NOT PnP O/S**, or **Other**, the BIOS will perform this task and the system may not work properly.

**Set 82350 Read/Write Performance Mode** The 82350 card read and write calls use one of two modes:

- **Polling.** Bytes are transferred to/from the card, one at a time. Polling mode is advantageous for transferring a small number of bytes because the setup overhead is very low, but it does require CPU involvement for each byte transferred.
- **Interrupt.** An entire buffer is transferred to/from the card without CPU involvement. Interrupt mode is advantageous for transferring large buffers because the higher per byte transfer rate more than compensates for the relatively long interrupt setup overhead.

The default behavior of the 82350 driver is to use **Polling** mode for transfers of 256 bytes or less and to use **Interrupt** mode for larger transfers. You can modify this default behavior by doing the following:

**VISA:** The VISA function

```
viSetAttribute(vi, VI_ATTR_DMA_ALLOW_EN,
attrValue)
```

can be called to modify the read/write behavior for a VISA session. The VI\_ATTR\_DMA\_ALLOW\_EN values allowed are:

- **VI\_TRUE** (default value) Use Interrupt mode for transfer requests larger than 256 bytes, otherwise, use Polling mode.
- **VI\_FALSE** Use the Polling mode.

**SICL:** The SICL function

```
ihint(id, hint)
```

can be called to modify the read/write behavior for on a SICL session. The hint values allowed are:

- **I\_HINT\_DONTCARE** (default value) Use Interrupt mode for transfer requests larger than 256 bytes, otherwise, use Polling mode.
- **I\_HINT\_USEPOLL** Use the Polling mode.
- **I\_HINT\_IO** Use the Interrupt mode.

Some additional factors to consider are:

- The settings discussed above are per session. This means you can open multiple sessions to a device and set different transfer modes for different sessions. The actual mode used will then depend on which session you are using for the read/write calls.
- In both SICL (with `hint = I_HINT_DONTCARE`) and VISA (with `VI_ATTR_DMA_ALLOW_EN = VI_TRUE`), the size of the read request (as specified by `bufsize` in a SICL `iread()` or `count` in a VISA `viRead()` function call) will determine the mode used even if the number of bytes actually read is less.
- The default formatted I/O read buffer size is 4096 so when using this default size, formatted reads in SICL (with `hint = I_HINT_DONTCARE`) and VISA (with `VI_ATTR_DMA_ALLOW_EN = VI_TRUE`) will use Interrupt mode even when a small number of bytes are expected.
- The default formatted IO write buffer size is 128 so when using this default size, formatted writes in SICL (with `hint = I_HINT_DONTCARE`) and VISA (with `VI_ATTR_DMA_ALLOW_EN = VI_TRUE`) will use Polling mode even when a large number of bytes are being sent.
- In SICL, Polling mode will always be used for the `iread()`, `ifread()` and `iscanf()` regardless of the above settings, when a `termchr` is set (`itermchr()` is not set to -1).
- In VISA, Polling mode will always be used for `viRead()`, `viBufRead()` and `viScanf()` regardless of the above settings, when `VI_ATTR_TERM_CHAR_EN = VI_TRUE`.

The crossover point at which the Interrupt mode becomes faster than the Polling mode depends on the CPU speed, with a faster CPU having a higher crossover point.

## Agilent IO Libraries Suite Checks

This section gives troubleshooting guidelines for the Agilent IO Libraries Suite, including:

- [Check IO Libraries Suite Installation](#)
- [Check Agilent IO Control Operation](#)
- [Install IO Libraries Suite \(if 82350 was installed first\)](#)

**Check IO Libraries Suite Installation** Start your Agilent IO Libraries Suite troubleshooting sequence by verifying IO Libraries Suite installation. If the IO Libraries Suite is installed, go to [Check Agilent IO Control Operation](#).

- 1 Check Agilent IO Libraries Suite Version.** If a version of the Agilent IO Libraries or Agilent IO Libraries Suite has been installed, a blue **IO** icon is normally displayed on the Windows taskbar (on the lower right-hand side of the screen).



**Figure 54** IO Control Icons for IO Libraries and IO Libraries Suite

- If the **IO** icon is displayed, click the icon and click **A**bout Agilent IO Control to display the version. The version must be M.01.00 or greater. (The next version of the IO Libraries after “M” discontinued this naming convention, and uses *IO Libraries Suite 14.0* instead to designate the version immediately following M.01.01.)
- If the **IO** icon is not displayed, a version may still be installed. To check this, click **Start | Programs** and look for the *Agilent IO Libraries Suite* program group.
- If this group is displayed, click **Agilent IO Libraries Suite > Utilities > IO Control** to display the **IO** icon. Then, click the icon and click **A**bout Agilent IO Control to display the installed version (must be M.01.00 or greater).

- If neither the **IO** icon nor the Agilent IO Libraries Suite program group is displayed, no Agilent IO Libraries Suite are installed. In this case, or if the installed version is not M.01.00 or greater, you must install the newer version (see Step 2, following).

**2 Install Agilent IO Libraries Suite (as Required).** If M.01.00 or greater is not installed on your PC, install the IO Libraries Suite. Otherwise, go to [Check Agilent IO Control Operation](#).

**Check Agilent IO Control Operation** When the Agilent IO Libraries Suite was installed, the Agilent IO Control was started. When the IO Control is active, it is displayed as a blue circled **IO** icon on the Windows taskbar. If the Agilent IO Control is deactivated, I/O applications that are running with the 82350 may be unable to open sessions or to communicate with the card and devices.

By default, the Agilent IO Control is always active after the Agilent IO Libraries Suite is installed and the blue **IO** icon is displayed. However, the IO Control may be active even though the blue **IO** icon is not displayed. There are two ways that the blue **IO** icon can be hidden:

- **Clicking Hide Agilent IO Control.** Clicking the blue **IO** icon and then clicking **Hide Agilent IO Control** hides the **IO** icon, but does *not* deactivate the Agilent IO Control.
- **Clicking Exit.** Clicking the blue **IO** icon and then clicking **Exit** causes a dialog box to appear that asks you if you want to terminate the Agilent IO Control. Clicking **Yes** hides the blue **IO** icon *and* deactivates the Agilent IO Control.

If the blue **IO** icon is not displayed, either the blue **IO** icon display has been turned off and/or the Agilent IO Control (and associated `iprocsvr.exe`) is not active. In this case, click **Start > Programs > Agilent IO Libraries** and then click **Agilent IO Control** to re-start the Agilent IO Control and to display the blue **IO** icon.

### Install IO Libraries Suite (if 82350 was Installed First)

**NOTE**

When you install the Agilent IO Libraries Suite, drivers for your 82350 GPIB card will be installed automatically. You should not need to take additional steps to install drivers. The information in this section is provided to assist you in case you have problems with your GPIB card and drivers.

---

- If you installed the 82350 GPIB card before installing the Agilent IO Libraries Suite software, use the applicable step for your operating system to install the Agilent IO Libraries Suite. After the drivers for all new cards are installed, you can run **Setup** from the *Automation-Ready CD with Agilent IO Libraries Suite* to install and configure the Agilent IO Libraries Suite.

**NOTE**

On older operating systems, you may be asked to insert the 'HP I/O Libraries' CD. You should use the CD that contains the Agilent IO Libraries Suite in this case. The card will be identified as a Hewlett-Packard card. This is necessary for backward compatibility.

---

- On Windows 2000, the **Found New Hardware Wizard** will identify this card as a **PCI Simple Communications Controller**. The driver is typically located in the *\Windows 2000* directory on the *instrument* CD.



## 9 Guide Information

This chapter includes:

- [Guide Contents](#)
- [Related Documentation](#)
- [Accessing an Electronic Copy of This Guide](#)
- [Contacting Agilent](#)



## Guide Contents

This *Agilent Technologies USB/LAN/GPIB Interfaces Connectivity Guide* shows how to connect instruments to USB, LAN, and GPIB interfaces and how to configure and troubleshoot these interfaces on PCs with Windows 2000 or Windows XP operating systems. A summary of the guide contents follows.

### NOTE

This guide does not describe LAN networks that include a gateway, such as the Agilent E5810A LAN/GPIB Gateway for Windows. See the applicable gateway documentation for information on gateway systems.

Also, this guide does not provide a detailed description of LAN, USB, or GPIB interfaces or LAN networks. Consult standard reference texts for this information.

Section	Description
<b>CONNECTIVITY GUIDELINES</b>	
<i>Getting Started</i>	Shows steps to get started and how to contact Agilent
<i>Connecting Instruments to LANs</i>	Gives guidelines to connect instruments to LANs
<i>Connecting Instruments to USB</i>	Gives guidelines to connect instruments to USB
<i>Connecting Instruments to GPIB</i>	Gives guidelines to connect instruments to GPIB
<i>Programming Your Instruments</i>	Gives guidelines to program instruments via LAN/USB/GPIB
<b>REFERENCE INFORMATION</b>	
<i>Installing I/O Software</i>	Shows how to install I/O software on your PC
<i>Troubleshooting Guidelines</i>	Troubleshooting guidelines for LAN, USB, and GPIB Interfaces
<i>TCP/IP Network Basics</i>	Provides an overview of TCP/IP networks
<i>Guide Information</i>	Lists general information for this guide
<i>Glossary</i>	Defines some of the terms used in this guide
<i>Index</i>	Index of selected items in this guide



## Related Documentation

Suggested related documentation you can use for interface connectivity operation follows. After Agilent IO Libraries Suite has been installed on your PC, PDF files of the *Agilent IO Libraries Getting Started Guide*, *VISA User's Guide*, and *SICL User's Guide* are available. Click the blue IO icon on the Windows taskbar and then click **Documentation**.

Product / Component	Related Documentation
Agilent IO Libraries Suite	For additional information on the Agilent IO Libraries Suite, see the <i>Agilent IO Libraries Suite Getting Started Guide</i> and <i>IO Libraries Suite Online Help</i> .
VISA	To develop and use VISA applications, see the <i>Agilent VISA User's Guide</i> .
SICL	For SICL I/O applications, see the <i>Agilent SICL User's Guide</i> .
VISA COM	For VISA COM I/O applications, see the <i>Agilent VISA COM Online Help</i> .
Agilent 488	For Agilent 488 I/O applications, see the <i>Agilent 488 Online Help</i> .
T&M Toolkit	To develop programs in Visual Studio.Net, see <i>Agilent T&amp;M Toolkit</i> ( <a href="http://www.agilent.com/find/toolkit">www.agilent.com/find/toolkit</a> ).

## Accessing an Electronic Copy of This Guide

### Access from the Agilent IO Control



After the Agilent IO Libraries Suite is installed, a blue **IO** icon appears on the right side of the Windows tray, as shown to the left. To access an electronic version of this connectivity guide, click the **IO** icon, then click **Documentation > Connectivity Guide**.

### Access from the IO Libraries Suite CD-ROM

You can also access an electronic copy of this guide from the *Automation-Ready CD*. Insert the CD into your CD-ROM drive. Browse into the autorun folder to find the PDF manuals.

## Contacting Agilent

You can reach Agilent Technologies at this telephone number in the United States:

Test and Measurement Contact Center: 1-800-829-4444

A list of contact information for other countries is available on the Agilent Internet site: [www.agilent.com/find/assist](http://www.agilent.com/find/assist). A list of other Agilent Web sites follows.

URL	Description
<a href="http://www.agilent.com/find/assist">www.agilent.com/find/assist</a>	Agilent Technologies "Contact Us" page
<a href="http://www.agilent.com/find/ioilib">www.agilent.com/find/ioilib</a>	Update the Agilent IO Libraries Suite software
<a href="http://www.agilent.com/find/ADN">www.agilent.com/find/ADN</a>	Connectivity resources all in one place
<a href="http://www.agilent.com/find/techsupport">www.agilent.com/find/techsupport</a>	Technical support information, including manuals, application notes, FAQs, and software and firmware downloads
<a href="http://www.agilent.com/find/connectivity">www.agilent.com/find/connectivity</a>	For connection, communication and control of test instruments from your computer, you can find out the latest in connectivity.





## Glossary

### address

A string (or other language construct) that uniquely locates and identifies a resource. VISA defines an ASCII-based grammar that associates address strings with particular physical devices or interfaces and VISA resources.

### Agilent 488

An I/O library provided in Agilent IO Libraries Suite for compatibility with existing test & measurement programs that were developed using National Instruments' NI-488.2 or other similar libraries. Agilent 488 supports communication with GPIB devices and interfaces, but does not support USB, LAN, RS-232, or VXI communications.

### alias

See **VISA alias**.

### API

Application Programming Interface. The interface that a programmer sees when creating an application. For example, the VISA API consists of the sum of all of the operations, attributes, and events of each of the VISA ResourceClasses.

### bridge

In telecommunication networks, a bridge is a product that connects a local area network (LAN) to another local area network that uses the same protocol (for example, Ethernet or token ring). You can envision a bridge as being a device that decides whether a message from you



to someone else is going to the local area network in your building or to someone on the local area network in the building across the street. A bridge examines each message on a LAN, “passing” those known to be within the same LAN, and forwarding those known to be on the other interconnected LAN (or LANs).

**client**

Part of the client/server model used in distributed computing. A client is a computer system that requests services from a server computer system, such as I/O application requests, networking requests, etc.

**Connection Expert**

An Agilent software utility that helps you quickly establish connections between your instruments and your PC. It also helps you troubleshoot connectivity problems. Connection Expert is part of the Agilent IO Libraries Suite product.

**controller**

A computer used to communicate with a remote device such as an instrument. In the communications between the controller and the device, the controller is in charge of and controls the flow of communication that is, it does the addressing and/or other bus management).

**device**

A unit that receives commands from a controller. Typically a device is an instrument but could also be a computer acting in a non-controller role, or another peripheral such as a printer or plotter.

**device driver**

Software code that communicates with a device: for example, a printer driver that communicates with a printer from a PC. A device driver may either

communicate directly with a device by reading to and writing from registers, or it may communicate through an interface driver.

## DHCP

Short for Dynamic Host Configuration Protocol, a protocol for assigning dynamic IP addresses to devices on a network. With dynamic addressing, a device can have a different IP address every time it connects to the network.

In some systems, the device's IP address can even change while it is still connected. DHCP also supports a mix of static and dynamic IP addresses.

Dynamic addressing simplifies network administration because the software keeps track of IP addresses rather than requiring an administrator to manage the task. This means that a new computer can be added to a network without manually assigning it a unique IP address.

## DNS

Short for Domain Name System (or Service), an Internet service that translates domain names into IP addresses. Because domain names are alphabetic, they are easier to remember. The Internet, however, is really based on IP addresses. Every time you use a domain name, therefore, a DNS service must translate the name into the corresponding IP address.

For example, the domain name **www.example.com** might translate to 198.105.232.4. The DNS system is a distributed system. If one DNS server does not know how to translate a particular domain name, it asks another one, and so on, until the correct IP address is returned.

## driver

See **instrument driver** and **device driver**.

**Ethernet (MAC) address**

The Media Access Control Address, also known as the link-level address, the Ethernet (station) address, the LANIC ID, and the hardware address. This is a unique 48-bit address assigned by the manufacturer for each Ethernet device. It is usually displayed as 12 hexadecimal characters, sometimes with colon or dash separators between every two characters, such as “00:03:d3:00:00:17” or “00-03-d3-00-00-17”.

**explorer view**

The tree view within the Connection Expert window that shows all devices connected to a test system.

**gateway**

Hardware that permits a network connection between the LAN that your computer understands and the instrument specific interface that your device understands.

**gateway IP address**

This parameter is the IP address of the default subnet gateway that allows an instrument to communicate with systems that are not on the local subnet. Thus, this is the default subnet gateway where packets are sent that are destined for a device not on the local subnet, as determined by the subnet mask setting. Only one default subnet gateway can be configured. A value of 0.0.0.0 indicates that no subnetting is to be done.

**hub**

A common connection point for devices in a network. Hubs are commonly used to connect segments of a LAN. A hub contains multiple ports. When a packet arrives at one port, it is copied to the other ports so that all segments of the LAN can see all packets.

- **Passive hubs** serve simply as a conduit for the data, enabling it to go from one device (or segment) to another.



- **Intelligent hubs** include additional features that enables an administrator to monitor the traffic passing through the hub and to configure each port in the hub. Intelligent hubs are also called manageable hubs.
- **Switching hubs** actually read the destination address of each packet and then forward the packet to the correct port.

**instrument**

A device that accepts commands and performs a test or measurement function.

**instrument driver**

Software that runs on a computer to allow an application to control a particular instrument.

**Interactive IO**

An Agilent application that allows you to interactively send commands to instruments and read the results. Interactive IO is part of the Agilent IO Libraries Suite product.

**interface**

A connection and communication media between devices and controllers. Interfaces include mechanical, electrical, and protocol connections.

**interface driver**

Software that communicates with an interface. The interface driver also handles commands used to perform communications on an interface.

**interrupt**

An asynchronous event that requires attention and actions that are out of the normal flow of control of a program.

## IO Control

The icon in the Windows notification area (usually the lower right corner of your screen). The IO Control gives you access to Agilent I/O utilities such as Connection Expert, Agilent I/O documentation, and VISA options.

## IO Libraries

Application programming interfaces (APIs) for direct I/O communication between applications and devices. There are four Agilent IO Libraries in the Agilent IO Libraries Suite: VISA, VISA COM, SICL, and Agilent 488.

## IP address

An Internet Protocol (IP) address is an identifier for a computer or device on a TCP/IP network. Networks using the TCP/IP protocol route messages based on the IP address of the destination. The format of an IP address is a 32-bit numeric address written as four numbers separated by periods. Each number can be zero to 255. For example, 1.160.10.240 could be an IP address.

Within an isolated network, you can assign IP addresses at random as long as each one is unique. However, connecting a private network to the Internet requires using registered IP addresses (called Internet addresses) to avoid duplicates. The four numbers in an IP address are used in different ways to identify a particular network and a host on that network. The InterNIC Registration Service assigns Internet addresses from the following three classes.

- Class A - supports 16 million hosts on each of 127 networks
- Class B - supports 65,000 hosts on each of 16,000 networks
- Class C - supports 254 hosts on each of 2 million networks

## LAN

**Local Area Network.** A computer network that spans a relatively small area. Most LANs are confined to a single building or group of buildings. However, one LAN can be connected to other LANs over any distance via telephone lines and radio waves. A system of LANs connected in this way is called a wide-area network (WAN).

Most LANs connect workstations and personal computers. Each node (individual computer) in a LAN has its own CPU with which it executes programs, but it also is able to access data and devices anywhere on the LAN. This means that many users can share expensive devices, such as laser printers, as well as data. Users can also use the LAN to communicate with each other, by sending e-mail or engaging in chat sessions.

There are many different types of LANs, Ethernets being the most common for PCs. Most Apple Macintosh networks are based on Apple's AppleTalk network system, which is built into Macintosh computers. The following characteristics differentiate one LAN from another:

**topology:** The geometric arrangement of devices on the network. For example, devices can be arranged in a ring or in a straight line.

**protocols:** The rules and encoding specifications for sending data. The protocols also determine whether the network uses a peer-to-peer or client/server architecture.

**media:** Devices can be connected by twisted-pair wire, coaxial cables, or fiber optic cables. Some networks do without connecting media altogether, communicating instead via radio waves.

LANs are capable of transmitting data at very fast rates, much faster than data can be transmitted over a telephone line. However, the distances are limited and there is also a limit on the number of computers that can be attached to a single LAN.

**Listener**

A device that can receive data from the bus when instructed (addressed to listen) by the System Controller.

**lock**

A state that prohibits other users from accessing a resource, such as a device or interface.

**logical unit**

A number associated with an interface. A logical unit, in SICL and Agilent VEE, uniquely identifies an interface. Each interface on the controller must have a unique logical unit.

**network**

A group of two or more computer systems linked together. There are many types of computer networks, including:

**local-area networks (LANs):** The computers are geographically close together (that is, in the same building).

**wide-area networks (WANs):** The computers are farther apart and are connected by telephone lines or radio waves.

**campus-area networks (CANs):** The computers are within a limited geographic area, such as a campus or military base.

**metropolitan-area networks (MANs):** A data network designed for a town or city.

**home-area networks (HANs):** A network contained within a user's home that connects a person's digital devices. In addition to these types, the following characteristics are also used to categorize different types of networks:

**topology:** The geometric arrangement of a computer system. Common topologies include a bus, star, and ring. See the Network topology diagrams in the Quick Reference section of Webopedia.

**protocol:** The protocol defines a common set of rules and signals that computers on the network use to communicate. One of the most popular protocols for LANs is called Ethernet. Another popular LAN protocol for PCs is the IBM token-ring network.

**architecture:** Networks can be broadly classified as using either a peer-to-peer or client/server architecture.

Computers on a network are sometimes called nodes. Computers and devices that allocate resources for a network are called servers

### network protocols

The protocol defines a common set of rules and signals that computers on the network use to communicate. One of the most popular protocols for LANs is called *Ethernet*. A description of the ARP, DHCP, DNS, FTP, HTTP, ICMP, and RPC protocols follows.

**ARP:** Short for Address Resolution Protocol, a TCP/IP protocol used to convert an IP address into a physical address (called a DLC address), such as an Ethernet address.

**DHCP:** See DHCP

**DNS:** See DNS

**FTP:** Abbreviation of File Transfer Protocol, the protocol used on the Internet for sending files.

**HTTP:** Short for HyperText Transfer Protocol, the underlying protocol used by the World Wide Web. HTTP defines how messages are formatted and transmitted, and what actions Web servers and browsers should take in response to various commands.

**ICMP:** Short for Internet Control Message Protocol, an extension to the Internet Protocol (IP) defined by RFC 792. ICMP supports packets containing error, control, and informational messages. The ping command, for example, uses ICMP to test an Internet connection.

**RPC:** Abbreviation of remote procedure call, a type of protocol that allows a program on one computer to execute a program on a server computer. Using RPC, a system developer need not develop specific procedures for the server. The client program sends a message to the server with appropriate arguments and the server returns a message containing the results of the program executed.

**non-Controller role**

A computer is in a non-Controller role when it acts as a device communicating with a computer that is in a Controller role.

**notification area**

The area on the Windows taskbar where notifications are posted, typically in the lower right corner of the screen. Also called **taskbar notification area** or **Windows notification area**.

**primary VISA**

The VISA installation that controls the visa32.dll file. The primary VISA will be used by default in VISA applications. See also **secondary VISA**.

**programming alias**

See **VISA alias**.

**proxy server**

A server that sits between a client application, such as a web browser, and a real server. It intercepts all requests to the real server to see if it can fulfill the requests itself. If not, it forwards the request to the real server. Proxy servers have two main purposes:

**Improve performance:** Proxy servers can dramatically improve performance for groups of users, since a proxy server saves the results of all requests for a certain amount of time. Consider the case where both user X and user Y access the Web through a proxy server. First user X requests a certain Web page, which we will call Page 1. Sometime later, user Y requests the same page.

Instead of forwarding the request to the Web server where Page 1 resides, which can be a time-consuming operation, the proxy server returns the Page 1 it already fetched for user X. Since the proxy server is often on the same network as the user, this is a much faster operation. Real proxy servers support hundreds or thousands of users.

**Filter requests:** Proxy servers can also be used to filter requests. For example, a company might use a proxy server to prevent its employees from accessing a specific set of Web sites.

**refresh**

In Connection Expert, the action that invokes the discovery mechanism for detecting interfaces and instruments connected to your computer. The explorer view is then refreshed to show the current, discovered state of your test system.

**register**

An address location that contains a value that represents the state of hardware, or that can be written into to cause hardware to perform a specified action or to enter a specified state.

**router**

A device that connects any number of LANs. Routers use headers and a forwarding table to determine where packets go. They use Internet Control Message Protocol (ICMP) to communicate with each other and configure the best route between any two hosts. Very little filtering of data is done through routers. Routers do not care about the type of data they handle. Routers often have DHCP Server capability.

**SCPI**

Standard Commands for Programmable Instrumentation: a standard set of commands, defined by the SCPI Consortium, to control programmable test and measurement devices in instrumentation systems.

**secondary VISA**

A VISA installation that does not install visa32.dll in the standard VISA location. The secondary VISA installation names its VISA DLL with a different name (agvisa32.dll) so that it can be accessed programmatically. The primary VISA will be used by default in VISA applications. See also **primary VISA**.

**server**

Part of the client/server model used in distributed computing. The server is a computer system designated to act as a main servicer of requests from other client computer systems, such as I/O application requests, networking requests, and so forth.

**session**

VISA term for a communication channel. An instance of a communications path between a software element and a resource. Every communication channel in VISA is unique.



**SICL**

Standard Instrument Control Library. SICL is an Agilent-defined API for instrument I/O. Agilent SICL is one of the IO Libraries installed with Agilent IO Libraries Suite.

**side-by-side**

A side-by-side installation allows two vendors' implementations of VISA to be used on the same computer. See also **primary VISA** and **secondary VISA**.

**SRQ**

An IEEE-488 Service Request. This is an asynchronous request (an interrupt) from a remote device that requires service. In GPIB, an SRQ is implemented by asserting the SRQ line on the GPIB. In VXI, an SRQ is implemented by sending the Request for Service True event (REQT).

**status byte**

A byte of information returned from a remote device that shows the current state and status of the device. If the device follows IEEE-488 (GPIB) conventions, bit 6 of the status byte indicates whether the device is currently requesting service.

**subnet**

A portion of a network that shares a common address component. On TCP/IP networks, subnets are defined as all devices whose IP addresses have the same prefix. For example, all devices with IP addresses that start with 100.100.100. would be part of the same subnet. Dividing a network into subnets is useful for both security and performance reasons. IP networks are divided using a subnet mask.

**subnet mask**

A mask used to determine to what subnet an IP address belongs. An IP address has two components: the network address and the host address. For example, consider the IP address 150.215.017.009. Assuming this is part of a Class B network, the first two numbers (150.215) represent the Class B network address and the second two numbers (017.009) identify a particular host on this network.

Subnetting enables a system administrator to further divide the host part of the address into two or more subnets. In this case, a part of the host address is reserved to identify the particular subnet.

For example, the full address for 150.215.017.009 is 10010110.11010111.00010001.00001001. The Class B network part is 10010110.11010111 and the host address is 00010001.00001001. If this network is divided into 14 subnets, the first four bits of the host address (0001) are reserved for identifying the subnet.

The subnet mask is the network address plus the bits reserved for identifying the subnetwork. (By convention, the bits for the network address are all set to 1, though it would also work if the bits were set exactly as in the network address.) In this case, the subnet mask is 11111111.11111111.11110000.00000000.

It is called a mask because it can be used to identify the subnet to which an IP address belongs by performing a bitwise AND operation on the mask and the IP address. The result is the subnetwork address:

```
Subnet Mask 255.255.240.000
              11111111.11111111.11110000.00000000
IP Address 150.215.017.009
              10010110.11010111.00010001.00001001
Subnet Address 150.215.016.000
              10010110.11010111.00010000.00000000
```

**switch**

A device that filters and forwards packets between LAN segments. Switches operate at the data link layer (layer 2) and sometimes the network layer (layer 3) of the OSI Reference Model and therefore support any packet protocol. LANs that use switches to join segments are called switched LANs or, in the case of Ethernet networks, switched Ethernet LANs. A hub connects all the devices on its “ports” together.

A switch is a bit smarter, as it understands when two devices (out of four, five, eight, sixteen, or even more) want to talk to each other and gives them a switched connection.

**symbolic name**

A name corresponding to a single interface. This name uniquely identifies the interface on a controller or gateway. If there is more than one interface on the controller or gateway, each interface must have a unique symbolic name.

**System Controller**

One Controller on a GPIB is the System Controller. This is a master Controller; it has the ability to demand control and to assert the IFC (Interface Clear) and REN (remote enable) lines.

**system tray**

See **notification area**.

**task guide**

The information and logic represented in the left pane of the Connection Expert window. The task guide provides links to actions and information that help guide you through the most common I/O configuration tasks.

**taskbar notification area**

See **notification area**.

**test system**

An entire test setup including a controller (often a PC), instruments, interfaces, software, and any remote controllers, instruments, and interfaces that are configured to be used as part of the system.

**Universal Plug and Play**

Universal Plug and Play (UPnP) is an open industry standard that uses Internet and Web protocols to enable devices such as PCs, peripherals, intelligent appliances, and wireless devices to be plugged into a network and automatically know about each other.

UPnP is an architecture for pervasive peer-to-peer network connectivity of PCs and intelligent devices or appliances. UPnP builds on Internet standards and technologies, such as TCP/IP, HTTP, and XML, to enable LAN devices to automatically connect with one another and work together.

With UPnP, when a user plugs a device into the network, the device will configure itself, acquire a TCP/IP address, and use a discovery protocol based on the Internet's Hypertext Transfer Protocol (HTTP) to announce its presence on the network to other devices. UPnP devices use Extensible Markup Language (XML) to establish a common language, or “protocol negotiation”, to talk to each other and determine capabilities.

**VEE**

The Agilent Visual Engineering Environment, which is software used for I/O application programming.

**virtual instrument**

A name given to the grouping of software modules (such as VISA resources with any associated or required hardware) to give them the functionality of a traditional stand-alone instrument. Within VISA, a virtual instrument is the logical grouping of any of the VISA resources. The VISA Instrument Control Resources Organizer serves as a means to group any number of any type of VISA Instrument Control Resources within a VISA system.

**VISA**

The Agilent Virtual Instrument Software Architecture library, which is software used for I/O application programming. Agilent VISA is part of the IO Libraries Suite product.

**VISA address**

A resource descriptor that can be used to open a VISA session.

**VISA alias**

A string that can be used instead of a resource descriptor in VISA programs. Using VISA aliases rather than hard-coded resource descriptors makes your programs more portable. You can define VISA aliases for your instruments in Connection Expert.

**VISA COM**

The *VXIplug&play* specification for a COM-compliant VISA I/O library and its implementation. Agilent VISA COM is part of the Agilent IO Libraries Suite.

**Windows notification area**

See **notification area**.



# Index

## Symbols

.NET Framework, 132

## A

address, 213

Address Resolution Protocol (ARP), 148

Agilent 488, 98, 111, 131, 135, 199, 213

Agilent IO Control icon, 87

Agilent Web sites, 211

Auto-IP/ZEROCONF, 161

## B

board number, 200

Bridge, 213

browser, 124

## C

client, 214

Configuring the Instrument page, 33

connecting instruments to LANs, 13

Connection Expert, 131

connections

testing, 131

controller, 214

copyright information, 210

crossover cable, 141

crossover point, 203

Current Configuration page, 33

## D

default gateways, 157

device, 214

DHCP, 161, 215

disk space

required for IO Libraries Suite

installation, 124

DNS, 215

domain, 165

dot-notation IP addresses, 156

dynamic host configuration protocol, 161

## E

error 1935, 132

Ethernet (MAC) Address, 216

examples

Addressing GPIB Instruments Using  
VISA/SICL, 112

Addressing LAN Instruments Using  
VISA/SICL, 107

Cache and Page Refresh, 186

Class A, B, and C IP Addresses, 156

Connecting Multiple GPIB

Instruments, 84

Creating Local and Remote  
Networks, 159

Direct Connection (Private LAN), 29

Direct Connection (Site LAN), 27

Enabling Javascript, 187

Installing TCP/IP (Windows 2000), 170

Installing TCP/IP (Windows XP), 171

Setting PC Hostname (Windows  
2000), 175

Setting PC Hostname (Windows  
XP), 175

Setting PC IP Address (Windows  
2000), 173

Setting PC IP Address (Windows  
XP), 173

Setting Proxy Notification, 185

Simple Sine Waveform (Visual Basic  
and VISA-COM), 114

Simple Sine Waveform (Visual C++ and  
IVI-COM), 116

Simple Sine Waveform (Visual C++ and  
VISA), 120

Switch Connections (Private LAN), 30

Switch Connections (Site LAN), 28

Using Network Protocols, 105

Using ping for Echo Request, 149

Using the route add Command, 183

Verifying Communication Using the ping  
Command, 182

Verifying Connection Using a Web  
Browser, 179

## F

File Transfer Protocol (FTP), 153

firewall, 141

Found New Hardware wizard, 64, 206

## G

gateway, 216

gateway IP address, 216

getting started, 9, 11

glossary, 213

GPIB

addressing instruments, 110

Agilent IO Libraries checks, 204

cable connection guidelines, 82

check for driver files, 199

communicate with instruments, 91

configuration files, installing, 87

configure GPIB cards, 86

connect instruments, 82

crossover point, 203

definition, 72

Found New Hardware wizard, 206

hardware checks, 196

install GPIB cards, 77

install software, 76

interrupt mode, 202

polling mode, 202

program your instruments, 95

quick start, 72

software checks, 199

troubleshooting overview, 195

guide

contents, 208

copyright information, 210

electronic copies, accessing, 210

related documentation, 209

using, 10

## H

hardware address, 147

Help, online, 135

host, 141

host portion, 155

hub, [141, 216](#)  
hyperlink, [154](#)  
Hypertext Transfer Protocol (HTTP), [154](#)

## I

ihint, [202](#)  
installation  
    location, [129](#)  
    upgrading, [126](#)  
installation error, [132](#)  
installing software  
    selecting Agilent IO Libraries, [133](#)  
    selecting application software, [133](#)  
instrument Welcome page, [32](#)  
instrument, definition, [217](#)  
instruments  
    configuring, [131](#)  
interface, [217](#)  
Interface not found, error in 488  
    programs, [200](#)  
interfaces  
    configuring, [131](#)  
Internet Control Message Protocol (ICMP), [149](#)  
IO Config, [127](#)  
iopen, [108, 113](#)  
IP addresses, [141, 156, 218](#)  
IVI components, [130](#)  
    location, [129](#)

## L

LAN, [219](#)  
    Auto-IP/ZEROCONF, [161](#)  
    cache and page refresh, [185](#)  
    checking PC Settings, [169](#)  
    checking proxy settings, [184](#)  
    client-server model, [142](#)  
    configuring PC for LAN operation, [169](#)  
    connecting instruments, [13](#)  
    DHCP, [161](#)  
    duplicate IP address detection, [162](#)  
    Dynamic DNS Naming, [165](#)  
    enabling Javascript, [187](#)  
    hardware architecture, [142](#)  
    installing TCP/IP, [170](#)  
    interface overview, [140](#)  
    IP address classes, [155](#)

IP addressing, [155](#)  
local and remote networks, [158](#)  
Network Interface cards, installing, [170](#)  
    quick start, [14](#)  
    request for comments (RFC), [146](#)  
    RFC NetBIOS Naming, [166](#)  
    setting PC hostname, [174](#)  
    setting PC IP address, [172](#)  
    SICL-LAN protocol, [104](#)  
    Static DNS Naming, [167](#)  
    subnets and subnet masks, [157](#)  
    TCP/IP Application Layer, [151](#)  
    TCP/IP Network Interface Layer, [147](#)  
    TCP/IP network model, [145](#)  
    TCP/IP protocols, [104, 145](#)  
    TCP/IP Transport Layer, [150](#)  
    troubleshooting overview, [178](#)  
    typical network topologies, [140](#)  
    VXI-11 protocol, [104](#)  
license, software, [126](#)  
lock, [220](#)  
logical address, [148](#)  
logical unit, [220](#)

## M

Media Access Control (MAC) address, [147](#)  
memory  
    required for IO Libraries Suite  
        installation, [124](#)  
Microsoft .NET Framework, [130](#)  
Microsoft Windows, [124](#)

## N

National Instruments VISA (NI-VISA), [127, 128](#)  
network classes, [218](#)  
Network Interface cards, installing, [170](#)  
network portion, [155](#)  
network protocols, [221](#)  
NI MAX, errors, [200](#)  
NI tools, errors, [200](#)  
NI-488.2, [98, 135, 199, 213](#)

## O

operating system, [124](#)  
operating systems, supported, [10](#)

## P

packet switching, [143](#)  
paths  
    choosing, [129](#)  
ports, [152](#)  
private LAN  
    connecting devices, [29](#)  
    definition, [14, 20](#)  
    network info card, [24](#)  
processor, [124](#)  
protocol stack, [145](#)  
protocol suite, [145](#)  
protocols, [145](#)  
proxy server, [223](#)

## R

request for comments (RFC), [146](#)  
RFC NetBIOS Naming, [166](#)  
routers, [141, 224](#)  
rsrcName, [109](#)

## S

server, [224](#)  
servers, [141](#)  
SICL, [225](#)  
SICL-LAN protocol, [104](#)  
site LAN  
    connecting instruments, [27](#)  
    definition, [14, 19](#)  
    information, gathering, [22](#)  
    network info card, [23](#)  
sockets, [152](#)  
SRQ, [225](#)  
Static DNS Naming, [167](#)  
subnet, [225](#)  
subnet mask, [155, 226](#)  
switch, [141, 227](#)  
symbolic name, [227](#)

## T

TCP/IP  
    network basics, [139](#)  
    network model, [145](#)  
    protocols, [104, 145](#)  
Transmission Control Protocol (TCP), [150](#)



## troubleshooting

- Agilent IO Libraries checks, [204](#)
- cache and page refresh, [185](#)
- check GPIB driver files, [199](#)
- check USB driver files, [193](#)
- checking proxy settings, [184](#)
- enabling Javascript, [187](#)
- GPIB crossover point, [203](#)
- GPIB hardware checks, [196](#)
- GPIB interrupt mode, [202](#)
- GPIB overview, [195](#)
- GPIB polling mode, [202](#)
- GPIB software checks, [199](#)
- LAN overview, [178](#)
- USB hardware checks, [190](#)
- USB overview, [188](#)
- USB software checks, [191](#)

## U

- Uniform Resource Locator (URL), [154](#)
- Universal Plug and Play, [228](#)
- upgrading to IO Libraries Suite, [126](#)
- USB

- addressing instruments, [109](#)
- assign Alias name, [62](#)
- check instrument ID, [66](#)
- check USB driver files, [193](#)
- communicate with instruments, [69](#)
- connecting instruments, [59](#)
- definition, [54](#)
- hardware checks, [190](#)
- quick start, [54](#)
- software checks, [191](#)
- supported OS, [54](#)
- troubleshooting overview, [188](#)

- User Datagram Protocol (UDP), [151](#)
- Using the ping Command, [182](#)
- Using the route add Command, [183](#)
- using this guide, [10](#)

## V

- VEE, [228](#)
- video requirement, [124](#)
- viOpen, [108, 113](#)

- VISA, [229](#)
  - location, [129](#)
  - National Instruments, [127](#)
  - primary, [127, 128](#)
  - secondary, [127, 128](#)
  - side-by-side, [127, 128](#)
- VISA rsrcName, [109](#)
- viSetAttribute, [202](#)
- VXI-11 protocol, [104](#)

## W

- Web Browser, [32](#)
  - configuring the instrument page, [33](#)
  - current configuration page, [33](#)
  - instrument Welcome page, [32](#)
  - set/change TCP/IP parameters, [34](#)
  - TCP/IP parameters, setting, [32](#)
- Web site
  - installing from, [129](#)
- Web sites, Agilent, [211](#)
- Windows version, [124](#)

