Introducing COM/IP®

Software Modem to Internet-Enable Windows PCs

COM/IP is client software from Tactical Software that essentially replaces a hardware modem and phone line with an IP connection. With it, a Windows COM port can communicate through any TCP/IP network, including the Internet. Applications that are built to expect a modem on a COM port can use the COM/IP redirector to "dial" using network addresses and "answer" incoming connections on your network or over the Internet.

COM/IP works by creating new virtual COM ports that appear to be connected to a Hayes-compatible modem. Instead of using a telephone line, the new ports transfer data on the LAN utilizing COM/IP's software modem technology. Applications that use modem connections can continue to assume the existence of a modem while leveraging the performance and cost benefits of the Internet.

This document is updated for Version 3.0 of COM/IP. You can find the latest revision of this information on the Tactical Software website.

Note: This user's guide is provided in both HTML/online and Adobe Acrobat PDF formats. In the Acrobat version, all of the links work the same as they do online. However, the drop-down menu for navigating the chapters do not work. Use the bookmarks feature in Acrobat instead to move from chapter to chapter.

Quick Links to Chapters

How to Use this Online Manual

Information in this user's guide is organized by chapter and section. To navigate to a chapter, use the drop-down menu box appearing in the upper left-hand corner of this page or the quick links below. A series of related section links will appear at the top of the page. Click on these links to jump directly to sections of interest. Alternatively, you can browse through sections by clicking on the right and left arrow keys appearing at the bottom of the screen.

1. Welcome — COM/IP Overview

What's Inside

This chapter provides useful background information that will help you decide when and how to apply COM/IP in your computing environment. This information will be most helpful if you are new to using pooled modems on modem-sharing servers or you are a network administrator who has not yet configured your modem-sharing server for use with COM/IP. If you are more interested in getting COM/IP installed, refer to the next chapter, Installation.

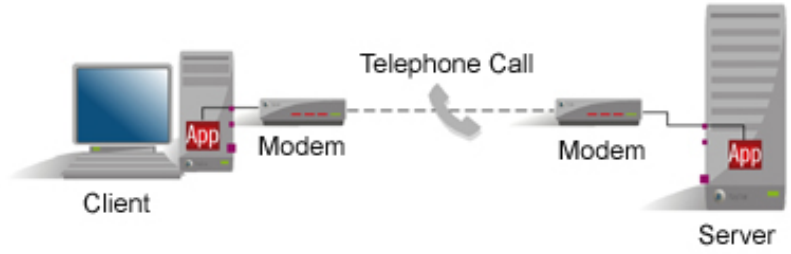
What's Inside the Chapter Sections

1. Introduction to COM/IP
   Basic background information on the product
2. Typical Applications
   Descriptions of common uses of COM/IP
3. Pre-Install Checklist
   What to know and check before installing COM/IP

1. Welcome — Introduction to COM/IP

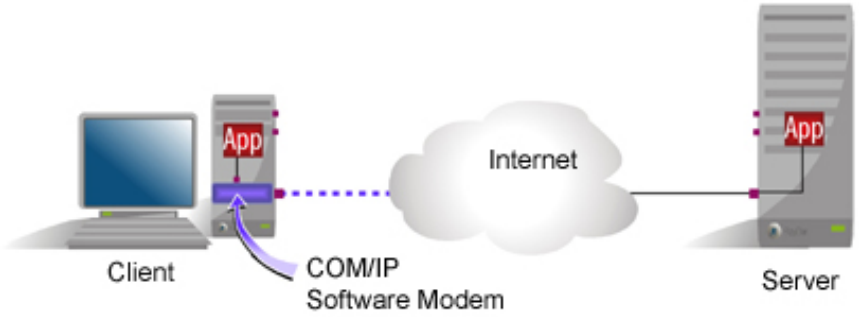
Background

COM/IP is a software package that lets your modem communications applications use TCP/IP connections. Your applications "see" what appears to be a modem on a COM port, but no hardware is required. The only necessary change to your applications is to configure them to "dial" IP network addresses instead of telephone numbers. In effect, your applications place a call using TCP/IP networking. In this User's Guide, we refer to these connections as "network calls." For example, here is a diagram showing a typical scenario using traditional hardware modems:



With modems, the client PC makes a telephone call over standard telephone lines, which in turn is answered by another modem.

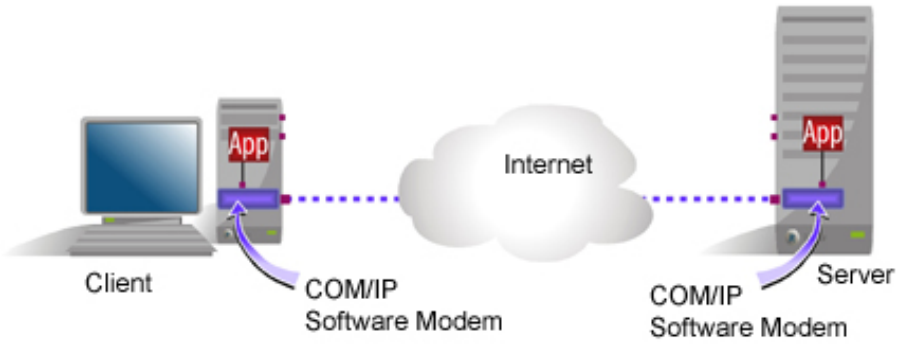
Making a modem connection over a computer network is a little different. The phone lines are replaced with a network connection, typically the Internet, and the connections that are completed are between a client and a server that can both accept TCP/IP network socket connections. This is where COM/IP comes into the picture. Here is a diagram showing a typical use of COM/IP to make network calls:



In practical terms, the receiver of the network call usually means a Telnet server, SMTP server or any other server software that "listens" for TCP/IP socket connections. The latter includes servers as diverse as bulletin board systems, financial information services and custom-built server applications. The bottom line is this: when considering your potential use of COM/IP, you must also determine whether the software at the other end is also network-capable.

Receiving Network Calls

In addition to placing network calls, COM/IP can also receive network calls. This means that you can use COM/IP to network-enable a client application that usually waits for a modem to indicate "ring" with an incoming call. A typical use of this would be to enable BBS software to accept connections over the Internet. Here is a diagram that shows this type of arrangement:



COM/IP includes a Hayes-compatible modem emulator that responds to standard modem commands and S-register settings. Additionally, the modem emulator supports a set of special COM/IP S-registers used for configuring COM/IP itself. There are sections in Chapter Three, Configuration, describing the operation of all supported S-registers.

How COM/IP Works

Here's a summary of how COM/IP provides access to networked modem pools:

When you install COM/IP, you create one or more new virtual COM ports. Unlike your other COM ports, these ports don't control serial hardware — they just appear to do so. You can also create Windows modem devices that use virtual COM ports instead of real COM ports. These modem devices are not to be confused with the modem emulator built into COM/IP. Rather, Windows modem devices are often used by certain Windows applications to simplify their interactions with various modem brands.

Here are some other points to keep in mind:

- For each virtual COM port, you adjust its configuration with an initialization string (called the Init String), much like that of a real modem. The special COM/IP S-registers are modified by the Init String, if necessary. One of the key settings of the COM/IP S-registers is the selection of Telnet protocol versus a raw socket connection.
- You then configure Windows client programs to use a virtual COM port directly or through a Windows modem device that uses a virtual COM port.
- When your client program uses the modem or the virtual COM port, COM/IP sees that the COM port has been opened, processes the Init String, and begins processing input from the client application. The first action of the client application is usually to initiate a network call.
- When your client program closes the COM port, COM/IP terminates the network call. The application at the other end of the network call may also terminate the connection.

Intro to COM/IP | Typical Applications | Pre-Install Checklist

1. Welcome — COM/IP Typical Applications

Overview

With COM/IP your PC applications can initiate (and receive) network calls using a virtual modem and TCP/IP network instead of a hardware modem and telephone lines. The types of applications that work with COM/IP are those that already use modems for dial-out or dial-in purposes. Often these applications are older software products that cannot be readily adapted to use TCP/IP networks directly.

Why make network calls instead of telephone calls? The obvious reason for using COM/IP network calls is economic: when an existing network replaces the use of modems and telephone lines there can be significant savings as individual desktop phone lines and modems are eliminated.

The increased performance of a network call is another compelling benefit. Not only can data flow faster, but network call setup time is significantly less than that of a telephone call and modem negotiation.

Compatible Applications

1. **Tested by Tactical Software**
   COM/IP has been thoroughly tested with the following popular packages during development:

- Windows HyperTerminal
- TurboCom ViP from Pacific Commware

2. **Popular Applications Employed by COM/IP users**
   COM/IP is used by thousands of people daily with applications such as:

- Procomm Plus for Windows v2 and later
- Crosstalk v2 and later
- Telix for Windows
- Quicklink II
- WinCIM v2 and later
- WinComm Pro, v7 and later
- Smartcom

More information about settings for common communications software is found in Chapter Three, Application Configuration Tips.

Note: Tactical Software does not have detailed information on all communications software. This information must be obtained from the software manufacturer.

COM/IP Compatibility Limitations

1. **Dial-Up Networking (DUN)**
   Due to fundamental Windows networking issues, COM/IP cannot be used to establish dial-up networking connections. It is possible, however, to make COM/IP network calls over a pre-existing dial-up networking connection.

2. **Fax Applications**
   The modem emulator in COM/IP emulates a data modem only. It does not support the additional fax-

related commands. Therefore it is not possible to use COM/IP with Class 1 or Class 2 fax packages, such as WinFax, or the fax software provided with Windows Exchange.

3. **Support for DOS Applications**
   Under Windows 98 and Me — the 3.0 release of COM/IP does not support Windows 95 — DOS applications that access the UART (universal asynchronous receiver-transmitter) directly are not supported directly by COM/IP. If support for such applications is required, there is a third-party software product that often solves the problem when used with COM/IP. Try TurboCom ViP from Pacific Commware.

   Under Windows XP, NT and 2000, DOS applications that manipulate the UART hardware directly are able to access both COM/IP and standard ports in the COM 1-4 range using the NTVDM driver (NT Virtual DOS Machine) that ships with these versions of Windows. The NTVDM is a library that maps 16-bit APIs into 32-bit APIs. It is installed by default.

   DOS applications will not be able to access ports COM5 and above using this mechanism, however. Also be aware that Tactical Software's testing has shown that not all DOS applications perform well under NTVDM, even with the standard COM port driver.

   More information is provided in Chapter Three, DOS Application Support.

Intro to COM/IP | Typical Applications | Pre-Install Checklist

1. Welcome — Pre-Installation Checklist

Confirm the Following

Before installing the COM/IP on your PC, you should verify the following information. Your answers to the following questions now will save installation time later.

- What is the IP address of the server to which you will place network calls?
- Does the server application use the Telnet protocol?
- How many simultaneous network calls will you be making from this PC?
- What COM ports on your PC correspond to real hardware?
- Where on your disk do you want the COM/IP files to be located?

PC Checklist

Before installing COM/IP, make sure that the following are true of your PC:

1. **Is the target PC able to run COM/IP? Confirm the following:**

- You are using an Intel-compatible PC running Windows XP, Windows 2000, Windows NT 4.0 SP5, Windows Me or Windows 98. The Citrix Metaframe and Windows Terminal Services multi-user environments are also supported.

  Note: COM/IP does not run under Windows NT 3.51 or on older versions of Citrix Metaframe software based on NT 3.51, and is no longer supported on Windows 95.

- Your PC has at least 4 megabytes of local disk storage for a complete installation of COM/IP software and documentation. The COM/IP installation program will place several small files in the Windows system folder. All other files will be located in the installation directory that you designate during installation.

- Your PC is connected to a TCP/IP network that includes the server(s) that you plan to contact with a network call.

  Note: This connection must permit the computer to communicate with the server using a Winsock socket. Almost all TCP/IP-based local area networks satisfy this requirement.

- You have the IP address of the server to which you will place network calls and the server application uses the Telnet protocol.

- The Windows Notepad accessory tool (notepad.exe) is accessible from the default installation directory.

  Note: The Microsoft Windows operating systems include Microsoft's TCP/IP networking software, and COM/IP is tested only with this software. If the Microsoft TCP/IP networking software is missing or has been replaced, COM/IP may not function properly.

2. **Installing the Software**

   Like most software products for Windows, COM/IP is installed with a setup program that asks a few preliminary questions, then places files on your disk and makes the requisite adjustments to Windows that will permit the software to run properly. Experienced Windows users will likely find all they need to proceed in Chapter Two's Quick Start Guide. More details are provided in the full installation instructions.

Intro to Shared Modems | COM/IP COM Ports | Pre-Install Checklist

2. Installation — Overview

Overview

Chapter Two provides instructions for both expert and less experienced Windows users on how to install and configure COM/IP. Additional information is also provided in this chapter on setting up and using Windows Modem Devices.

Power users will likely find sufficient information in the Quick Start section to install and operate the software. Those requiring more detailed instructions or background information should read each of the other sections in this chapter in the order presented.

Technical Support

If you require additional assistance, you can contact Tactical Software's support group by filling out the online support request form. A support representative will respond via e-mail within eight business hours of the request being logged.

Why Read the Manual?

Even those experienced with communications software for Windows should read and follow the installation procedures presented in this chapter. The information has been carefully developed by Tactical Software support professionals to reflect their experience in helping users get up and running quickly and easily.

What's Inside the Chapter Sections

1. Quick Start Guide
   Fastest route to installing and configuring COM/IP (for power users)
2. Install Software
   Detailed instructions for installing COM/IP on Windows operating systems
3. Uninstall Software
   How to uninstall COM/IP
4. Windows Modem Devices
   Create the modem devices that some applications use instead of directly accessing the virtual COM ports

2. Installation — Quick Start Guide

Quick Install and Configuration Information

Users with a solid understanding of Windows-based PCs can follow the basic installation and configuration information below to get up and running quickly with COM/IP. If more information is required or a step does not go as expected, refer to the more detailed information in this chapter such as the complete software installation section.

1. **First, make some basic checks:**

- You are using an Intel-compatible PC running Windows XP, Windows 2000, Windows NT 4.0 SP5, Windows Me or Windows 98.
- Your PC is networked to a modem-sharing server on a TCP/IP network.

2. **Next, install COM/IP on your PC:**

- Run the COM/IP installer program and follow its simple instructions.
- In the COM/IP Manager window, create new COM/IP COM ports on your PC.
- Run the Windows modem installation wizard to install the correct modem driver (modem definition file) for each new COM port used.

3. **Reconfigure your applications to use COM/IP COM ports and modems instead of local ports and modems:**

- Change your application modem settings to use a previously-installed Windows modem or a COM/IP COM port.
  Note: In most cases, you should select the modem device name (such as U.S. Robotics 56K Voice) and *not* a COM port number.
- Your application will now detect the COM/IP virtual modem just as if it were a hardware modem.
  Note: More specific instructions for this step appear in Chapter Three, Application Configuration Tips.

2. Installation — Software Installation Guide

Follow these Steps

Like most software products for Windows, COM/IP is installed by running an installer program that asks some questions and places the COM/IP software on your local disk. This section provides detailed instructions for installing the COM/IP software and setting up the virtual COM ports needed for operation.

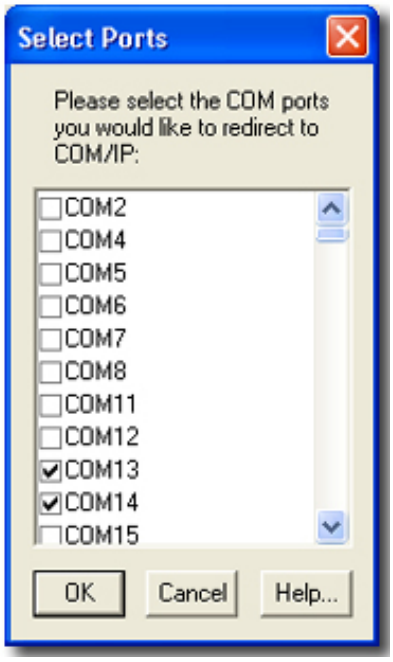
1. **Install COM/IP on your PC:**

- Log in as a user with administration privileges. (It is not necessary to be an admin to use COM/IP.)
- Quit all Windows programs that use COM ports.
- Run the COM/IP installer program and read the release notes. (The displayed readme.txt file will be placed in the installation directory for future reference.)
- Review the license information and indicate whether you agree to continue the installation under the terms presented. If you stop the installation, your PC will be unaffected.
- Select the destination directory where you want COM/IP to be installed or go with the default setting. Then select which components you wish to install: COM/IP, the HTML docs or INT14 Support.
  Note: Select INT14 Support if you plan to run any legacy INT14/FOSSIL applications in DOS windows.
- Enter your name, company and serial number in the boxes provided. Then click Finish.
  Note: Inserting no serial number will invoke a 30-day evaluation. If you have previously installed an evaluation copy of COM/IP, or have an existing older version, you will be prompted to run the Uninstall procedure.

Create and Configure the COM/IP COM Ports

COM/IP virtual COM ports are easy to create. The COM/IP Select Ports window will automatically appear after the installation completes. This window can also be used later to modify the list of virtual COM ports.

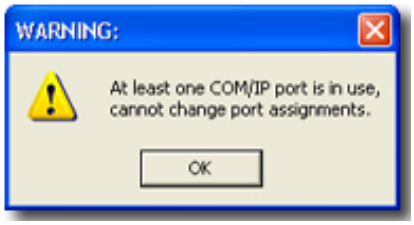
1. **Create the COM/IP COM ports:**



- After the installer runs, the Select Ports Window appears. Check one or more new COM ports for COM/IP to create and click OK. Because some older client programs do not display COM ports higher than COM4, consider selecting ports in the COM 1-4 range if you will be using such applications.
  Note: The list of COM ports will not include those already in use by Windows on this PC.

- You can later change the set of ports defined as COM/IP COM ports by right clicking on the COM/IP icon in the system tray and choosing Configure. Click the Select Ports button to change the port settings.

  Note: If you attempt to change the ports setting while one or more COM/IP COM ports are still in use, COM/IP will display a warning message that the port assignments cannot be changed.



- A window appears asking if you want to launch the Windows modem installation wizard. Click Add Modems.

- The Windows modem installation wizard should now launch. Because many applications use modem devices (instead of COM ports directly), it's very important to install the correct modem driver (modem definition file) for each new COM port used. Windows typically lets you install the same driver for multiple ports in one step — this saves time leading to a faster installation. If the wizard does not launch automatically, open the Windows Control Panel and activate the wizard manually.
  Note: Do not ask Windows to automatically detect the modem or be concerned if a "modem test" fails.

- The Wizard should display "Tactical Software, LLC" as the manufacturer and "COM-IP Modem" in the list of modem models presented. If it does not, direct the wizard to the directory where you installed COM/IP. The driver is an INF file with a name ending in ".INF." Standard modems and generic drivers should be avoided.
  Note: *This is a critical step. If no driver — or the wrong driver — is installed for a COM port, COM/IP will not work*.

- After installing the driver, it is highly recommended that you reboot your PC.

2. **Configure and verify COM/IP COM ports:**

- The main COM/IP manager window should now be open. If not, right click on the COM/IP icon in the System Tray and choose Configure. Then configure each COM/IP virtual COM port with the Init String field setting that will activate COM/IP's modem emulator when this virtual COM port is opened. For more information on init strings, click here.

Quick Start Guide | Install Software | Uninstall Software
Windows Modem Devices

2. Installation — Uninstall Software Guide

Uninstall the Software

The COM/IP uninstall procedure automatically shuts down COM/IP if necessary, removes all COM/IP files from your disk, and removes all COM/IP virtual COM ports for the PC. Follow these steps:

- Run Add/Remove Programs in the Control Panel.
- Select COM/IP Version 3.0 in the list of installed software.
- Click the Add/Remove button to start the uninstall wizard.
- In the wizard's Select Uninstall Method dialog, choose the default choice, **Automatic**.
- Click **Next**, then **Finish**.
- Restart the PC.

2. Installation — Create Windows Modem Devices

Windows modem devices allow Windows applications to more easily use different types of modems. Windows HyperTerminal and many other applications can — and sometimes must — use Windows modem devices. If your application needs a specific modem device, as opposed to a COM port, follow the instructions here to create Windows modem devices for COM/IP virtual COM ports.

Creating a Windows Modem Device

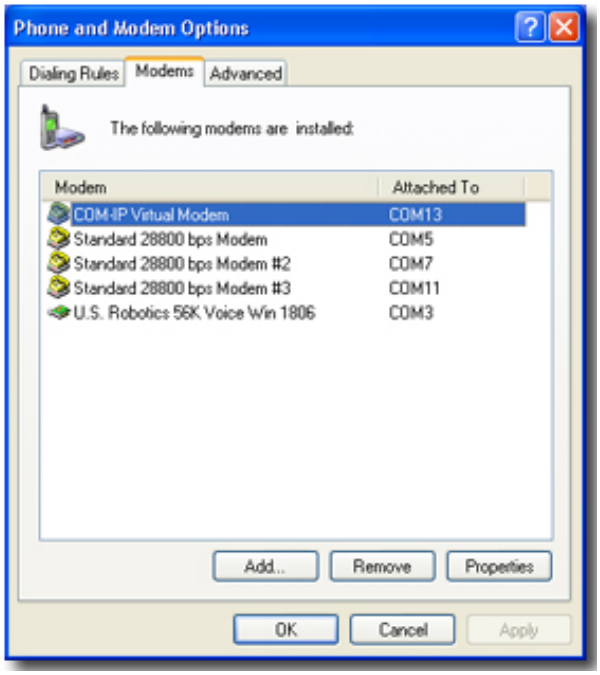
All versions of Windows include a hardware wizard which simplifies the process of creating a modem device for a standard modem on a modem-sharing server. This wizard is accessed through the Windows Control Panel. Because the wizard is implemented slightly different in each Windows version, the following instructions are generalized for all versions of Windows. Refer to your Windows documentation if additional details are needed.

1. Open the Windows Control Panel and click on the Modems file or icon.
2. Click Add to begin adding a new modem.
3. If Windows offers to detect the modem, check the box that prevents automatic detection.
4. When you are presented with a list of modems directly supported by Windows, choose instead to direct Windows to the directory where you have installed COM/IP. This directory will have two ".INF" files as choices.
5. Select the modem named **COM-IP**. This is a Hayes-compatible modem device definition supplied with COM/IP that reflects the capabilities of the modem emulator built into COM/IP.
6. When you are asked to identify the COM port for the modem, you will see COM/IP virtual COM ports included in the list.
7. When you have completed this procedure, Windows will automatically show the new modem in installed modem lists.

   Note: After installing the driver, it is highly recommended that you reboot your PC.

Recognition from Windows

In the example here, the PC originally has a U.S. Robotics 56K hardware modem physically attached to COM3. Most importantly, it now has a COM/IP modem device using the COM/IP virtual COM port, COM13. (The PC also has a number of generic drivers associated with other virtual COM ports.)



Quick Start Guide | Install Software | Uninstall Software
Windows Modem Devices

3. Configuration — Overview

Overview

Once COM/IP is installed, it is necessary to configure its virtual COM ports and to direct client programs to use virtual COM ports. Carefully review the sections of this chapter for configuration procedures and suggestions.

What's Inside the Chapter Sections

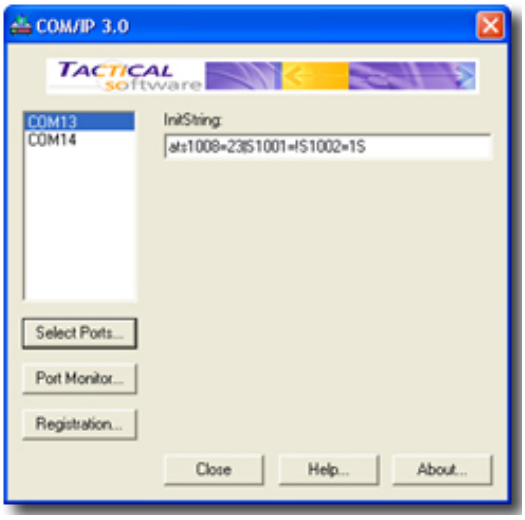
1. Manager Window Tour
   A look inside COM/IP's main "desktop dashboard"
2. Port Monitor Window
   How to set or change the port settings in COM/IP
3. Initiating a Call
   To place a network call, use these steps
4. Receiving a Call
   To receive a network call, use these steps
5. Application Configuration Tips
   Many COM/IP problems with applications can often be traced to improper configuration settings
6. AT Commands
   The COM/IP modem emulator and its supported Hayes-compatible "AT" command set
7. S-Registers Overview
   Detailing the S-registers supported by COM/IP
8. DOS Application Support
   How COM/IP provides limited support of DOS applications in Windows

3. Configuration — Manager Window Tour

The Manager window serves as the primary "desktop dashboard" for accessing most features in COM/IP. For example, COM/IP ports are configured from this window, which is also used to access the current list of ports, change their configuration and settings, and check the real-time status of COM port activity. This section provides a brief overview of all these features.

Following installation, you can open the Manager window at any time by right clicking on the system tray icon and choosing Configure. Alternatively, you can open it by selecting COM/IP Configuration from the COM/IP menu selection of the Windows Start menu.

This screen shot shows a typical Manager window display. This is just an example, and your Manager window likely will show different settings.



The vertical box on the left displays the list of DialOut/IP COM ports that have been created — in this case, COM13 and COM12. When you click on a COM port and highlight it, the InitString field changes to show the configuration of that port.

The Select Ports Button

This button displays the COM/IP port selection window described in the Software Installation section of Chapter Two.

The Port Monitor Button

In COM/P, it's easy to view the data activity present on any COM port. Just click on the Port Monitor button. More detailed information is presented in Chapter Three's next section, The Port Monitor Window.

The Registration Button

The Registration button displays the license dialog. This allows you to enter company information and a serial number, thereby upgrading your evaluation version of DialOut/IP to a standard license at any time.

The Init String Field

The Init String is used to initialize the COM/IP modem emulator when the COM port is opened and before COM/IP processes commands from the client program. Usually, this string is used to set COM/IP S-register values. Because the Init String is being delivered to a modem emulator, it takes the familiar form of a modem command string, such as:

```
AT S1001=2 S1002=1
```

The COM/IP modem emulator responds to basic "AT" commands and two sets of S-registers: Hayes-compatible standard S-registers and COM/IP S-registers.

The Help Button

The Help button opens COM/IP's Quick Start guide, which is provided in PDF format. Viewing the guide requires Adobe's Acrobat Reader® software, which can be downloaded free of charge from the Adobe website.

The About Button

The About button displays the COM/IP About window providing the version of COM/IP you are running. If you are running an evaluation version, the expiration date will be noted.

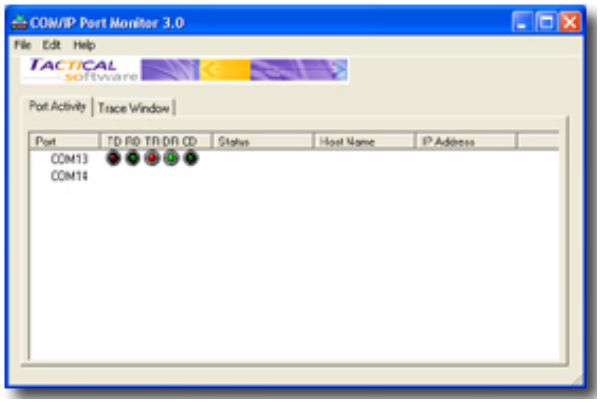
Manager Window Tour | Port Monitor Window | Initiating a Call | Receiving a Call
Application Configuration Tips | AT Commands | S-Registers Overview | DOS Application Support

3. Configuration — Port Monitor Window

Accessing the Window

In COM/IP, it's easy to view the data activity present on any COM port. From the Manager window, click on the Port Monitor button. Alternatively, open the Port Monitor window directly by right clicking on the system tray icon and choosing Port Monitor. This window defaults to the Port Activity tab.

Immediately to the right of each COM label is a space that contains the status indicator lights. These appear when an application has opened a COM/IP COM port. Some of the names displayed in the window are abbreviated due to space constraints. The more common names for the signals are listed in parentheses below:



- **TD** is on when transmitting data to the modem or network connection.
- **RD** is on when receiving data from the modem or network connection.
- **TR** (DTR) is the signal to the modem emulator that the PC is online and ready to communicate. The most frequent use of DTR is to signal the modem emulator to disconnect by lowering the DTR line.
- **DR** (DSR) is the signal to the PC that the modem emulator is online and ready to communicate.
- **CD** (DCD) is the signal from the modem emulator to the PC that it has successfully negotiated a connection over the network.

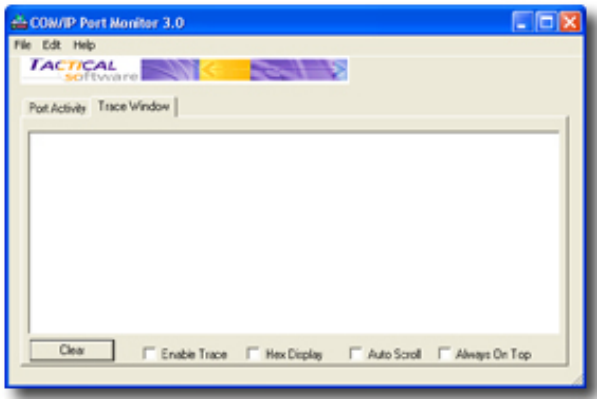
These indicators only appear when an application has opened a COM/IP COM port. At other times, the indicator area is blank.

Current Connection Display

Further right of the status indicator lights is a column header labeled IP Address. When there is an active connection to a remote server, the IP Address of that server is displayed here. This is useful because it confirms that the TCP/IP "dialing" has succeeded. BBS operators also find it useful for determining the originating IP Address of the incoming network calls.

The Trace Window Tab

The Trace Window tab appears in the Port Monitor window. Click this tab to see a detailed trace of the data moving between COM/IP and the connected server software. Further details about the Trace Window is located in Chapter Four (Debugging with Trace Window).



Manager Window Tour | Port Monitor Window | Initiating a Call | Receiving a Call
Application Configuration Tips | AT Commands | S-Registers Overview | DOS Application Support

3. Configuration — Initiating a Network Call

Background

Before initiating a network call to a remote system using COM/IP, confirm the following:

- COM/IP is correctly installed and configured.
- Your communications software is correctly configured to use COM/IP.
- If you are using dial-up network access, your connection is online and active at the time.
- The remote system is accessible on the network.

  Note: For most communications software, you must know the IP address of the system where you intend to connect. For some communications software, you can use the hostname of the remote system.

Specifying the Remote System's IP Address

Although COM/IP can process the hostname of the remote system, many communications packages fail to provide the name to COM/IP. Therefore, the only certain way to specify the remote system is with its IP address. Even then, the IP address must usually be expressed in a form that will not be altered by the communications software.

Fortunately, there is a simple 12-digit format for the IP address that will pass through all communications software correctly to COM/IP. To translate an IP address into this format, follow this procedure:

1. For each of the four numbers in the IP address, make sure that each one is exactly three digits long. Do this by adding one or zeros to the beginning of each number as necessary.
2. Remove the dots between the numbers in the IP address. This will leave you with a single, 12-digit number. For example, the IP address "10.182.50.3" translates to the 12-digit number "010182050003." As another example, the IP address "10.182.2.130" translates to "010182002130." IP addresses specified in this format look like a phone number to the communications software, but are transformed into the correct IP format by COM/IP when attempting to connect to the remote system.

In the likely case that you are forced by the communications software to use either a numeric IP address or the 12-digit equivalent, try using the communications software's phonebook capabilities to ease the burden of remembering these obscure numbers. Refer to the Application Configuration Tips section in Chapter Three for a listing of which communications software packages require what phone number format.

TCP/IP Port Numbers

By default, COM/IP is configured to use port 23 (the Telnet port) on outgoing connections. This setting is determined by the S1009 register (see the S-Registers Overview section for details). If you wish to connect to a port that is different from the S1009 setting — for example, to connect to an SMTP server on port 25 — then you need to explicitly specify the port number in the phone number of the remote host. This can be done in one of two ways:

1. If you are using hostnames or IP addresses, add the port number to the end of the hostname or address, enclosed in square brackets "[ ]." For example, to specify port 6000 you would use "telnet.xyz.com[6000]" for a hostname or "10.182.50.3[6000]" for an IP address.

2. If you are using a 12-digit equivalent IP address, you would add an additional five digits containing the port number, using leading zeros if necessary. For example, "10.182.50.3[6000]" would translate into the following 17-digit equivalent: "01018205000306000."

Why You Cannot Dial the Remote Systems Hostname

Ideally, you would simply substitute a hostname where your communications software asks for a phone number, and COM/IP would then connect to that network address rather than dialing a phone number. For example, you should be able to just tell the communications software to dial "telnet.xyz.com" rather than "1-800-555-1212." Here's why this doesn't usually work in practice.

First, the communications software may add an area code to the phone number. Most software packages can be configured to automatically detect when a phone number needs a long-distance access code (e.g., "1" plus the phone number) and the area code to be added to the phone number. Because area codes and long-distance access codes have no meaning in IP host addresses, this feature would need to be disabled in the communications software.

Secondly, many communications software packages attempt to error-check the phone number you enter, making it impossible to use a hostname in place of a phone number. For example, if you instructed your communications software to dial "telnet.xyz.com," one of three things may occur:

1. The communications software may substitute the touch-tone equivalents of the letters, transforming this call into 32-999-0266, which is obviously not the intent.

2. The communications software may simply remove characters from the phone number which it deems to be invalid, which in this case would be no phone number at all.

3. The software may report back that it determines the phone number to be invalid, either when you entered it or when it attempts to dial. As a workaround to this problem, it should be possible to specify the numeric IP address rather than a hostname.

   For example, assume that telnet.xyz.com has an IP address of "10.182.50.3." Although this looks more like a phone number than does telnet.xyz.com, in many cases the communications software may consider the dots between the numbers to be invalid characters and simply remove them, again leaving you with an invalid address (in this case, 1-018-2503).

   The foolproof way to specify the remote host is to use the 12-digit numeric format described above.

Manager Window Tour | Port Monitor Window | Initiating a Call | Receiving a Call
Application Configuration Tips | AT Commands | S-Registers Overview | DOS Application Support

3. Configuration — Receiving a Network Call

Background

Before receiving a network call using COM/IP, confirm the following:

- COM/IP is correctly installed and configured.
- Your communications software is correctly configured to use COM/IP.
- Your communications software is set to automatically answer incoming data calls. You should set the software to answer calls on the first ring.
- If you are using dial-up network access, your connection should be online and active at the time.

Besides the above, there are three other issues you must consider: the IP address of your PC; the network accessibility of your PC; and the TCP/IP port number you will use for incoming connections. More information is provided below

Fixed Versus Dynamic IP Addresses and Dial-Up Accounts

It is easier to receive network calls with COM/IP if your computer has a *fixed* IP address rather than a *dynamic* IP address.

Dynamically-assigned IP addresses are often the type provided by dial-up Internet Access accounts, such as a single-user PPP account with an Internet Service Provider, or with online services such as CompuServe or America Online. These dial-up accounts work by taking a small number of IP addresses (typically much smaller than the total number of potential users), and assigns one to the user's computer at the time it dials in. As this IP address is later reused, the IP address provided during a subsequent dial-up connection is typically not the same.

The problem that this presents to COM/IP is that there is no way to know what IP address you will be assigned until after you dial in to your service. On one call you might be assigned 10.182.50.3; the next call you might be assigned 10.182.44.23.

Dynamically-assigned IP addresses effectively mean that someone else cannot determine which host address to use to connect to your system. If you intend to use your computer to receive incoming network calls on a regular basis, consider asking your Internet Service Providers for a fixed IP address. Many ISP's can accommodate this, although others cannot. Currently, none of the major online services such as CompuServe or America Online offers this service.

Note: this is only an issue if you intend to receive incoming calls with COM/IP. If you will be using your PC only to place outgoing calls, it does not matter if your IP address is dynamically assigned.

How to Handle Incoming Calls

1. **Network Accessibility**

If you have a dial-up account, you must be connected to receive incoming network calls. This is the equivalent of having your modem turned on (computer dialed-in) or having it turned off (computer not online).

2. **Enabling Incoming Calls**
   By default COM/IP is not configured to listen for incoming calls. If want to enable incoming calls, it is recommended that you do so by adding "S1008=23" to your communications software's modem init string so that COM/IP will listen for incoming connections on port 23 (the standard port for Telnet connections). Most PC's do not have software to accept incoming Telnet connections, so this port should not conflict with your existing software. In the case that port 23 is a problem, we recommend using one of the following ports:

- Port 80, the HTTP Server port
- Port 119, the NNTP News Server port
- Any port number higher than 4096

3. **Disabling Incoming Calls**
   If you do not intend to receive calls on your PC using COM/IP, you should configure your communications software to not automatically answer incoming calls. Additionally, you should add "S1008=0" to your communications software's modem init string or simply not initialize the S1008 register.

4. **Supporting Multiple Incoming Calls**
   COM/IP can support more than one COM port to await incoming calls on a single TCP/IP port number. For example, you can specify "S1008=23" on multiple COM/IP COM ports, such as COM5, COM6, COM7 and COM8. This feature allows non-network-aware programs — such as BBS programs — to service multiple incoming connections much the same way they would with multiple phone lines.

   When more than one COM port listens on a single TCP/IP port number, and a call comes in on that port number, COM/IP routes the call to the lowest-numbered COM port that is not currently servicing a network call. Continuing the above example, if COM5 and COM7 were currently active on other network calls, an incoming call on port 23 would be routed to COM6, which would emit the familiar “RING” message until the client software answered the call.

Manager Window Tour | Port Monitor Window | Initiating a Call | Receiving a Call
Application Configuration Tips | AT Commands | S-Registers Overview | DOS Application Support

3. Configuration — Application Configuration Tips

Background

COM/IP communication problems can often be traced to improper configuration settings. The following guidelines will help you set up COM/IP for several popular communications packages and environments.

Communication Settings

1. **Terminal Emulation**
   If you want your remote host to automatically recognize a particular terminal emulation, add a section to your modem initialization string which reads S1000="terminal." In this case, terminal refers to the terminal type you are using. See the S-Registers Overview section for more details on S-registers.

2. **Baud Rate**
   You can specify any baud rate in your communications software, but be aware that COM/IP will inform the remote Telnet server of the baud rate you choose. The recommended approach is to choose either a very high baud rate (e.g., 38400 or 57600) for TCP/IP connections through a network or choose the actual baud rate you are using for your SLIP or PPP connection.
   Note: The actual throughput you see through the COM/IP port depends solely on the bandwidth of your TCP/IP connection, not the Baud Rate setting.

3. **Transferring Binary Files**
   Transferring binary files through a Telnet connection can be difficult because the interpretation of Binary Mode and Carriage-Return padding varies widely among different Telnet implementations. To get binary file transfer to work properly you may need to experiment with the S1002 and S1005 registers to find the best settings.

   On at least one UNIX-based Telnet server tested by Tactical Software, S1002=1 and S1005=0 work best with binary files, although many files will transfer fine with S1002=0 and S1005=3. Also try switching to the YMODEM protocol rather than ZMODEM protocol, as in some cases better results occur using YMODEM. See the S-Registers Overview section for more details on S-registers.

Application Software Settings

1. **HyperTerminal**
   When configuring Windows HyperTerminal, you should have installed one or more COM//IP modem drivers, and you should select one of the COM-IP modems from the list of modems on your system. If you wish to access one of the COM/IP ports directly, you will need to have installed the COM/IP port as a low-numbered port, as HyperTerminal only allows you direct access to ports COM 1-4.

2. **WinCIM**
   COM/IP can be made to work with CompuServe's WinCIM, although you either need WinCIM version 1.3 or the latest support files. Of key importance is the ability to dial using the network setting of "Internet." If that option is not available in your copy of WinCIM, you need to either upgrade to version 1.3 or obtain the latest support files from CompuServe.

   To use COM/IP with WinCIM, change the Session Settings as follows:

   Phone: gateway.compuserve.com[4144]

Baud Rate: 9600
Network: Internet
Connector: (one of the Telnet Modems)

In addition, set the Modem Control String as follows:

Initialize: AT&F &C1 &D2 S1006=4^M

Note: The above settings apply only if you intend to use WinCIM directly through the Internet. If you are going to use WinCIM over a networked modem, then you need to add "AT S1006=4" to the Auto Init box in the COM/IP Manager Property Sheet.

3. **Smartcom for Windows**
   COM/IP can be made to work with Hayes' Smartcom for Windows. To do so, you need to either check the "Use reduced command set" box in the Modem Settings or configure it to use "Generic Hayes Compatible" for the modem type. Failure to do so will prevent dialing using COM/IP.

4. **COMit for Windows**
   COMit has trouble dialing with COM/IP if you configure it to use 9600 or 19200 baud. Instead, use 38400 or 57600 baud.

5. **Reflection**
   Reflection uses its own driver, bypassing COM/IP and the Windows drivers. In order to get COM/IP to work with Reflection, issue the following command in the Reflection Command Window:

   SET ENHANCED-SERIAL-DATACOMM NO

Manager Window Tour | Port Monitor Window | Initiating a Call | Receiving a Call
Application Configuration Tips | AT Commands | S-Registers Overview | DOS Application Support

3. Configuration — AT Commands

Background

The COM/IP modem emulator provides a Hayes-compatible "AT" command set. Most AT commands are ignored because they are relevant only to a real modem. The following section lists the AT commands that are relevant to COM/IP.

Modem Emulator Standard AT Command Set

1. **A — Answer**
   When an incoming connection is pending, this command instructs the modem emulator to open the connection, returning a CONNECT response string. If there is no pending incoming connection, the word ERROR is output.

2. **D — Dial Command**
   This command instructs the modem emulator to attempt to establish a connection to the named-host or IP address following the D command. This works similarly to a modem, except that a network address is specified instead of a phone number. The "T" and "P" (tone or pulse) modifiers are allowed to follow the D command, and they are simply ignored. The modem emulator attempts to connect to the specified host, returning a CONNECT response string on success or NO CARRIER upon failure.

   If the connection fails, an additional error message beginning with WINSOCK: is output, providing more detail as to why the connection failed. As with a normal modem, if you press any key while the connection attempt is in progress, the connection will be aborted and NO CARRIER will be output.

3. **E0 or E1 — Command-Mode Echo**
   This command enables or disables character echoing while the modem emulator is in "command mode" (i.e., accepting "AT" commands). E1 enables echo and E0 disables echo. The default is E1.

4. **F0 or F1 — Online Echo**
   This command enables or disables character echoing while the modem emulator is in "online mode" (i.e., actually connected to a remote host). F1 enables echo and F0 disables echo. The default is F0.

5. **H0 — Go Off-Hook**
   This command is valid only when the modem emulator has established a connection, and it causes the emulator to close the current connection.

6. **O — Return to Online Mode**
   This command is valid only when the modem emulator has established a connection, and it causes the emulator to exit "command mode" and returns control of the serial line to the current Telnet connection.

7. **Sr=n — Write to an S-Register**
   This command writes a value "n" into the S-register numbered "r." The list of S-registers supported is listed later. Any write to an unsupported S-register is simply ignored.

8. **Sr? — Query an S-Register**
   This command outputs the current value of the S-register numbered "r." The list of S-registers supported is listed later. Any write to an unsupported S-register causes the word ERROR to be output and parsing of the current command line is terminated.

9. **V0, V1**

This command sets the modem emulator to use text or numeric responses. V0 enables numeric responses (e.g., "0" is the equivalent of "OK"), and V1 enables text responses (e.g., OK, NO CARRIER, etc.). The default is V1.

10. **W0, W1, W2 — Negotiation Progress Messages**
    In addition to the CONNECT response string, this command can be used to enable the CARRIER, PROTOCOL and COMPRESSION messages that are available with some MNP or v.42 modems. W0 and W2 disable the extended progress messages, and W1 enables the CARRIER and PROTOCOL messages.

    Note: Any value other than zero in S95 will override this setting. Also note that since no compression or transmission protocols are actually being used, they are always reported as NONE. The default is W0.

11. **Z — Reset**
    This command resets the modem emulator to use the default settings, and will close any currently open connection.

12. **&C0, &C1, &C2 — Data Carrier Detect Options**
    This command controls how the Data Carrier Detect (DCD) line is reported by the modem emulator. &C0 causes the emulator to always report the DCD line as ON.

    &C1 causes the emulator to report the DCD line as OFF while there is no connection, is switched to ON immediately after the CONNECT message and OFF again when the connection is lost.

    &C2 causes the emulator to report the DCD line as ON while there is no connection, and as OFF while a connection is being established, then ON again once the connection is made. The default is &C1.

13. **&D0, &D1, &D2, &D3 — Data Terminal Ready Options**
    The command controls how control of the Data Terminal Ready (DTR) line by an application is interpreted by the modem emulator. &D0 causes the emulator to ignore the DTR line.

    &D1 causes an ON-to-OFF transition of the DTR line to simply put the emulator in command mode while maintaining any currently open connection.

    &D2 or &D3 causes an ON-to-OFF transition of the DTR line to close any currently open connections (it issues a NO CARRIER response if one was open) and puts the emulator in command mode. The default is &D0.

14. **&F — Factory Defaults**
    This command acts exactly like the "Z" command.

15. **&S0, &S1, &S2 — Data Set Ready Options**
    This command controls how the Data Set Ready (DSR) line is reported by the modem emulator. &S0 causes the emulator to always report the DSR line as ON.

    &S1 causes the emulator to report the DSR line as OFF while there is no connection opened, switches it to ON when attempting to open a connection, then OFF again once the connection is closed.

    &S2 causes the emulator to report the DSR line as OFF while there is no connection opened, switches it to ON when a connection is opened, then OFF again once the connection is closed. The default is &S0.

16. **&V — View current settings**
    This command displays the current settings being uses by the modem emulator. A sample output (using the default settings) is shown below:

    ```
    ACTIVE PROFILE:

    E1 F1 W0 &C1 &D0 &S0

    S0:000 S02:043 S03:013 S04:010 S05:008 S12:050 S95:000

    S1000:000

    S1001:002 S1002:000 S1003:001 S1004:001 S1005:003

    S1006:000 S1007:243 S1008:023
    ```

Manager Window Tour | Port Monitor Window | Initiating a Call | Receiving a Call
Application Configuration Tips | AT Commands | S-Registers Overview | DOS Application Support

3. Configuration — S-Registers Overview

Standard S-Registers

1. **S0 — Auto-Answer Ring Count**
   This register, when set to a number greater than zero, enables support for automatically answering incoming TCP/IP connections on the port specified by the S1008 register (listed later). When an incoming connection is received, the modem emulator will output a single RING message followed by a CONNECT message. At this point it will switch to online mode and the port will be connected to the other host.

2. **S2 — Escape Sequence Character**
   This register holds the ASCII value of the escape character used in the escape sequence. The escape sequence is defined by a period of silence (as defined in S12) followed by three escape characters, followed by another period of silence. This causes the modem emulator to switch from online mode to command mode while a connection is open. The default is 43 (the "+" character).

3. **S3 — Carriage Return Character**
   This register holds the ASCII value of the character which is both understood and output for a carriage return. The default is 13.

4. **S4 — Line Feed Character**
   This register holds the ASCII value of the character which is both understood and output for a line feed. The default is 10.

5. **S5 — Backspace Character**
   This register holds the ASCII value of the character which is both understood and output for a backspace. The default is 8.

6. **S12 — Escape Sequence Guard Time**
   This register holds the value of the silence period required when issuing an escape sequence, in 1/50th of a second increments. The default value is 50 (1 second).

7. **S95 — Negotiation Progress Messages**
   This register, when set to anything other than zero, overrides the W setting in how the emulator reports connection progress. Setting bit 2 of the register enables the CARRIER message. Setting bit 3 of the register enables the PROTOCOL message. Setting bit 5 of the register enables the COMPRESSION message. The default is 0.

Specialized COM/IP S-Registers

For the purpose of setting Telnet-specific options, the modem emulator supports an extended set of S-registers, numbered 1000 through 1010. Note that since COM/IP does not normally use the Telnet protocol, only registers 1004, 1006, 1008, 1009 and 1010 have any significant effect.

1. **S1000 — Terminal Type**
   This register is unique in that it can accept both a numeric and a string value. By setting this register to a value other than 0, any Terminal-Type query by the remote Telnet server is answered with the current value of this register. By setting this register to a string value (e.g., ATS1000="vt100"), the verbatim string is reported. By setting this register to a numeric value (e.g., ATS1000=5), the numeric value (presumably a well-known terminal id) is reported. The default is 0.

2. **S1001 — Interpret IAC Characters**
   The Telnet protocol is based on an escape character (ASCII 255) known as the IAC character. This register controls whether COM/IP will interpret and deliver these escape sequences. Setting this register to 0 causes the client to treat the IAC character as any other. Setting this register to 1 causes COM/IP to always interpret IAC. Setting this register to 2 causes COM/IP to interpret/ignore based upon the requested port number.

   If a connection to port 23 is made (the default in the D command, and the standard Telnet server port), then IAC escapes are interpreted. If a connection to any other port number is made by using [nn] in the D command, then IAC characters are treated like any other. The default is 2.

3. **S1002 — Request Binary Connection**
   The Telnet connection can be used either in Terminal or Binary modes. This register controls whether a Binary connection is requested by COM/IP. Setting this register to 0 causes COM/IP to never request a Binary connection. Setting this register to 1 causes COM/IP to always request a Binary connection. Setting this register to 2 causes COM/IP to request a Binary connection only if the port is opened using a byte length of 8 bits. The default is 0.

4. **S1003 — Telnet Echo**
   The Telnet server can be operated in either Echo or Non-Echo modes. This register controls whether an Echo mode connection is requested by COM/IP. Setting this register to 0 causes COM/IP to never request an Echo mode connection. Setting this register to 1 causes COM/IP to request that the remote host perform echo. Setting this register to 2 causes COM/IP to request permission to echo data back to the remote host (and COM/IP will respond affirmatively to such requests by the remote host). If the remote host agrees, then COM/IP will echo all received data back to the remote host. Setting this register to 3 causes both local and remote echo behavior. The default setting is 1.

   This register also supports a "delegated echo" mode, which is enabled by setting bit-2 to one (i.e., adding 4 to the values in the previous paragraph). When delegated echo is enabled, COM/IP assumes that any echoing will be done by the client program, most often a BBS server. Therefore, if this register is set to 6 (i.e., enable local echo but also use delegated echo), COM/IP will inform the remote host that it will perform echo but will not in fact do the echoing. Rather, it will assume that the client program will be performing the echoing on its behalf.

   If COM/IP is used to host a BBS server, and it is found that certain programs are seeing double-echo when connecting to the BBS server, then setting S1003=6 will often remedy the situation.

5. **S1004 — Verbose Connection Failure**
   This register controls whether the additional WINSOCK: error message is reported before the NO CARRIER response. Setting this register to 0 disables the WINSOCK message, and setting it to 1 enables the message. The default is 1.

6. **S1005 — Carriage-Return Padding**
   The Telnet protocol specifies that all Carriage-Return characters (ASCII 13) which are sent without a Line-Feed character (ASCII 10) should be sent as "CR NUL" (ASCII 13 followed by ASCII 0). Setting this register to 1 causes COM/IP to strip away any NUL character received which follows a Carriage-Return. Setting this register to 2 causes COM/IP to add NUL characters to any outbound Carriage-Return which is not directly followed by a Line-Feed. Setting this register to 3 enables both inbound and outbound translations, and setting it to 0 disables both.

   Note: S1001 will override this setting, such that if IAC characters are not being processed, neither will Carriage Return characters be processed. The default is 3.

7. **S1006 — Deadlock Timer**
   This register takes a value from 0 to 100, and specifies (in 1/10ths of seconds) how often COM/IP should check for incoming data. Use this to control applications that cause deadlock by continually checking for incoming data from the modem without allowing Windows to process messages.

8. **S1007 — BREAK Handling**
   This register takes a value of 0, or a value from 243 through 248 inclusive. When COM/IP sees a "break" request from the application, it either ignores it (if this register is set to zero) or sends an IAC byte followed by the value of this register. This will be interpreted by the Telnet server according to the following table:

   S1007 Interpretation

```
==========================

243BRK (break)

244IP (Interrupt Process)

245AO (Abort)

246AYT (Are You There)

247EC (Erase Character)

248EL (Erase Line)
```

The default value is 243, the Break character.

9. **S1008 — TCP/IP IN port number**
   This register takes a value from 0 to 65535, and specifies which TCP/IP port number to listen on. If the value is 0, then incoming connections are disabled. If the value is not 0, COM/IP will listen for incoming connections on the specified TCP/IP port. Note that the S1000 registers (except S1003) affect incoming as well as outgoing connections. The default is 0, incoming connections disabled.

10. **S1009 — TCP/IP OUT port number**
    This register takes a value from 0 to 65535, and specifies the default TCP/IP port number to connect to on outbound connections. If the value is 0, then port 23 is used. If the value is not 0, COM/IP attempts to make outbound connections on the specified TCP/IP port unless overridden by a port number in the dial string. The default is 23, listen on the Telnet port.

11. **S1010 — Immediate Modem Response (Windows 3.1 Applications Only)**
    This register takes a value of 0 or 1, controlling whether modem response strings are delayed by 0 or 1 second. This behavior is needed by certain Windows 3.1 client programs that require the delay in order to be satisfied with modem-like behavior. Setting this register to 0 disables the delay. The default value is 1, which enables the delay.

Manager Window Tour | Port Monitor Window | Initiating a Call | Receiving a Call
Application Configuration Tips | AT Commands | S-Registers Overview | DOS Application Support

3. Configuration — DOS Application Support

Overview

COM/IP primarily supports Windows applications that interact with COM ports. However, it also contains limited support for DOS applications running under Windows. This section explains how that support works.

There are two types of DOS applications that use COM ports. They are:

1. **Direct COM port access.** This common type of DOS application uses a COM port by directly accessing the COM port hardware (also known as the UART).
2. **System BIOS access.** Found less often, this type of DOS application may also use a COM port through the system BIOS that leverages an interface known as INT14. Many programs intended for use with bulletin board systems (BBSs) use an enhanced version of this interface known as FOSSIL.

UART-based Applications

Under Windows 98 and Me, DOS applications that access the UART (universal asynchronous receiver-transmitter) directly are not supported directly by COM/IP. If support for such applications is required, there is a third-party software product that often solves the problem when used with COM/IP. Try TurboCom ViP from Pacific Commware.

Under Windows XP, NT and 2000, DOS applications that manipulate the UART hardware directly are able to access both DialOut and standard ports in the COM 1-4 range using the NTVDM driver (NT Virtual DOS Machine) that ships with these versions of Windows. The NTVDM is a library that maps 16-bit APIs into 32-bit APIs. It is installed by default.

DOS applications will not be able to access ports COM5 and above using this mechanism, however. Also be aware that Tactical Software's testing has shown that not all DOS applications perform well under NTVDM, even with the standard COM port driver.

INT14 / FOSSIL Applications

COM/IP supports INT14/FOSSIL applications on all Windows platforms. This includes FOSSIL-5 with the X00 extensions as well as the simpler INT14 BIOS interface. A DOS application that supports INT14 or FOSSIL will be able to use any of the COM/IP COM ports.

Note: In order to use INT14 applications, the INT14 support feature must be selected during the COM/IP installation process covered in Chapter Two. This loads a driver which enables the DOS sessions to take place.

Under Windows 98 and Me, the INT14/FOSSIL driver is loaded automatically at boot time. However, under Windows XP, NT and 2000, this driver support is by default loaded only on demand. For these operating systems, installing COM/IP adds a new command called "tsnt14." For any given DOS window, be sure to run the tsnt14 command prior to running any DOS applications which require INT14 or FOSSIL support. This loads the INT14/FOSSIL driver support *for that DOS Window only.*

Note: *This command is required to be run for each DOS Window which requires INT14/FOSSIL support to be activated.*

It is also possible to unload the driver by rerunning the command using a /u option. For example, to run a BBS which requires FOSSIL — and then unload the FOSSIL driver — use the following commands:

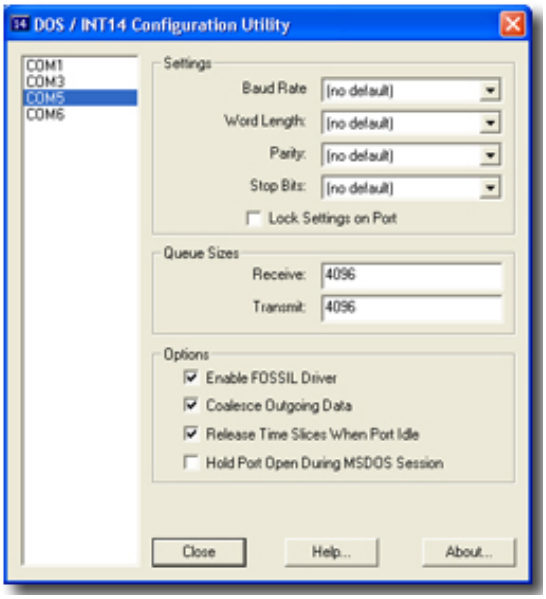
```
tsnt14
bbs
tsnt14 /u
```

In certain situations it may be desirable to have the FOSSIL driver enabled for all DOS sessions. One example is hosting a BBS that spawns external programs. In this case add the tsnt14 command to the end of the autoexec.nt file. To edit the autoexec.nt file, simply use the Run command in the Start menu and enter the following command:

```
notepad %SYSTEMROOT%\SYSTEM32\autoexec.nt
```

INT14/FOSSIL Options

The DOS/INT14 settings for each COM port can be changed through the DOS / INT14 Configuration Utility that is optionally installed when the INT14 support feature is selected during the COM/IP installation process. To open the utility, go to the Windows Control panel and click on the DOS / INT-14 Configuration Utility icon. This opens the DOS / INT14 Configuration Utility window.



A list of all Windows COM ports appears in the box on the left. For each selected port, the right section is divided into three sections: Settings, Queue Sizes and Options values.

1. **Settings.** The default COM port settings are applied to the COM port when it is first opened via INT14.

   Note: The default value for each setting is "no default." This means that no extra COM port configuration is done by the INT14 driver. The example shown here is keeping the original default settings.

- **Baud Rate.** Use this if a DOS application needs to operate at a particular modem speed. In most cases the "no default" choice works fine.
- **Word Length.** Use this if a DOS application needs to operate with a particular word length. In most cases the "no default" choice works fine.
- **Parity.** Use this if a DOS application needs to operate with a particular parity setting. In most cases the "no default" choice works fine.
- **Stop Bits.** Use this if a DOS application needs to operate with a particular stop bit setting. In most cases the "no default" choice works fine.
- **Lock Settings on Port.** When checked, the default COM port settings are "locked" to the COM port. Subsequent attempts to change the COM port settings via INT14 are ignored by the driver. When unchecked, a client application is free to override the default settings.

2. **Queue Sizes.** These two fields set the number of bytes that the INT14 driver will request for the transmit and receive queues on the highlighted COM port. The default is 4096, and the value can range from 16 to 16384. For most applications 4096 will be adequate, given that TCP/IP connections implement their own flow control.

3. **Options**. The following checkbox options are presented:

- **Enable Fossil Driver.** When checked (the default setting), the built-in INT14 redirector will respond to

requests on the highlighted COM port. When unchecked, requests on this COM port are passed along to the next driver, if one is present.

- **Coalesce Outgoing Data.** To minimize ring transitions and to optimize throughput, the built-in INT14 redirector will buffer small amounts of outgoing data (from the client application to the COM port) for up to 50ms before releasing it. If this option is checked (the default setting), this buffering is enabled. If unchecked, the buffering is disabled and outgoing data is always handed over immediately to the COM port.

- **Release Time Slices When Port Idle.** When checked (the default setting), the INT14 driver will release the time slice for the DOS session when it detects the port is idle and the session has requested a data-related INT14 service. The port is considered idle when DCD is low and there is no data in the transmit or receive queues. If a client application is suffering from performance or throughput issues, try disabling this option.

- **Hold Port Open During MSDOS Session.** The default behavior (unchecked) is for the INT14 driver to close the COM port 6 to 10 seconds after seeing a FOSSIL close request, or after 30 seconds of inactivity when the DOS application is using the minimal INT14 BIOS interface. Checking this option will cause the INT14 driver to never close the COM port during the DOS session in which it was opened.

  Note: If checked, the INT14 BIOS applications will not timeout, and the FOSSIL close request will be ignored. This option is mainly used for BBS programs with "doors" that open FOSSIL ports in a non-standard way.

The TSLOCK Utility

The TSLOCK command allows the parameters found in the Settings area of the DOS / INT14 Configuration Utility window to be changed with a DOS command. The TSLOCK command line and options are:

```
tslock COMx [baud,length,parity,stop] [ locked |
unlocked ]
```

The first parameter is the COM port to be adjusted. Following that can be the port settings and/or the locking status for those settings. For example, to set COM4 to a 57600 8-N-1 and lock those settings, use the following:

```
tslock COM4 57600,8,N,1 locked
```

To then unlock COM4's settings without affecting their current value, use the following:

```
tslock COM4 unlocked
```

To change COM4 to 38400 7-E-1 without affecting the locking status, use the following:

```
tslock COM4 38400,7,E,1
```

Note: *This command changes the settings as seen in the DOS / INT14 Configuration Utility window interface, where you can see the results of running this command.* These settings will only take effect upon the following conditions:

1. A FOSSIL application opens or reopens the COM port.

2. A FOSSIL application attempts to change the settings of the port .

Application Summary

For most client applications such as terminal programs, the default DOS port settings should be sufficient. For most BBS-hosting programs, the recommended settings are to disable the Release Time Slices option and to enable the Host Port Open option.

Note: All DOS-related settings have no effect on Windows applications. It is counterproductive to experiment with

these settings when troubleshooting COM/IP in conjunction with non-DOS applications.

Manager Window Tour | Port Monitor Window | Initiating a Call | Receiving a Call
Application Configuration Tips | AT Commands | S-Registers Overview | DOS Application Support

4. Troubleshooting — Overview

This chapter contains supplementary information that can assist you in deciding how to approach problems when configuring COM/IP.

What's Inside the Chapter Sections

1. Frequently Asked Questions
   Answers to questions about installing and using COM/IP

   Note: This section links to the FAQ section on Tactical Software's website, which contains comprehensive information pertaining to COM/IP and other products in the Tactical Software family

2. Common Problems
   Helps identify the causes and solutions for problems related to COM/IP configuration

3. Debugging with Trace Data
   Explains how this mode collects detailed information about the activity of the COM/IP software modem

4. InitString Overview
   Useful examples for setting COM/IP's InitString field

4. Troubleshooting — Common Problems

In most cases, installing COM/IP is easy and straightforward. However, when trouble arises, pinpointing the exact cause can take a little more work. The following list of common problems are among those most frequently reported by COM/IP users.

1. **Telnet protocol setting mismatch.**
   COM/IP is unable to automatically determine whether the network call receiver expects to use the Telnet protocol. You *must* make sure that both caller and receiver are set the same way — with or without the Telnet protocol.

2. **Client program isn't set to use the right modem and/or COM port.**
   It is surprisingly easy to forget to set client programs to use a COM/IP modem device or virtual COM port.

3. **Network call receiver is requesting a username/password login.**
   COM/IP cannot automatically respond to a username/password prompt. It will be relayed directly to the client program instead.

4. **Trying to use COM/IP as a modem pool.**
   Try using Tactical Software's DialOut/IP product instead. This is designed to work with modem pools.

5. **Using the Telnet protocol for binary transfers but without binary mode enabled.**
   If Telnet is enabled and the network call will be transporting binary data, binary mode must also be enabled with S-register 1002.

6. **The software reports "modem not ready" or "modem not responding."**
   Make sure that the COM/IP Manager is running. If you do not see its icon in the System Tray then it is not running and you will need to restart it.

7. **The software reports "unable to initialize modem."**
   Make sure that you have configured the communications software for the correct modem type. See the Application Configuration Tips section in Chapter Three for more details. If you have configured the software to receive incoming calls, try changing the S1009 register value in COM/IP's InitString field then restarting the communications software. The value you have selected may be conflicting with other software running on your PC.

8. **The software reports failure when dialing.**
   If you are on a dial-up Internet account (i.e., you use a modem to connect to the Internet) then make sure that you are dialed in at the time you attempt to place a call. Check that you have specified the correct IP address and port number of the remote host. See the Initiating a Call section in Chapter Three for more details.

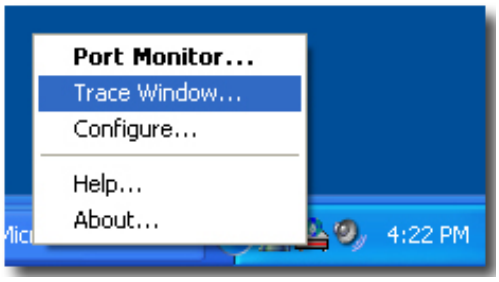
9. **The software is unable to receive incoming TCP/IP calls.**
   If you are on a dial-up Internet account (i.e., you use a modem to connect to the Internet) then make sure that you are dialed in at the time you wish to receive incoming calls. Ensure that you have configured the communications software to auto answer incoming calls, preferably on the first ring. See the Receiving a Network Call section in Chapter Three for more details. Try changing the S1008 register in the communications software's Modem Init String setting then restarting the communications software. The value you have selected may be conflicting with other software running on your PC.

Common Problems | Debugging with Trace Data | InitString Overview

4. Troubleshooting — Debugging with the Trace Window

COM/IP's Trace Window feature helps to solve difficult configuration problems. This window lets you monitor all interactions between the client application and the remote server. The window works by recording the interactions both on-screen and in a file that can be examined off-line or sent to others for analysis. The Trace Window is available by right clicking on the COM/IP system tray icon.

Selecting the **Trace Window** menu option opens the COM/IP Port Monitor 3.0 window, with the Trace Window tab enabled. To begin tracing, check the **Enable Trace** box at the bottom of this window.



Using Tracing

To collect trace data, follow these four steps:

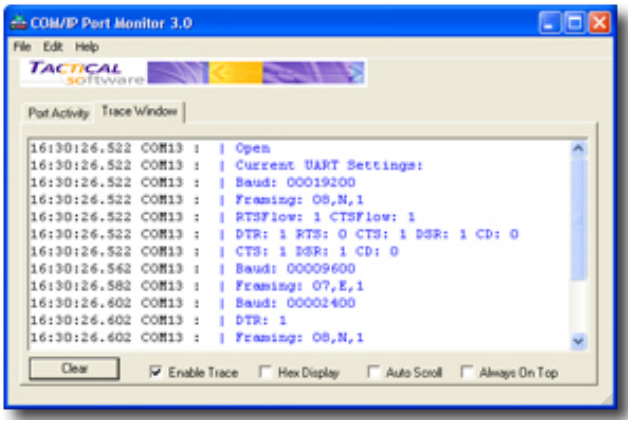
1. Be sure you are logged in as administrator.
2. Enable tracing as described above.
3. Recreate the problem condition that you are debugging.
4. Allow the Trace Window to show and record the history of communications activity between COM/IP and the remote server.
5. To save the trace data for future review, use the File pull-down menu option and select "Save As." Give the file a name and directory destination. COM/IP then saves trace data automatically in its native *.cilog binary file format. Files saved in this format open natively in the COM/IP Port Monitor window.

   You can also save files for examination in text editors, such as Windows Notepad, by first running a trace and then by selecting "Copy" from the Edit pull-down menu option. Copy places the text content into the Windows clipboard, where it can then be pasted into any text editor program.

   If a support representative requests a log file during the course of a support incident, it will be necessary to email the file as an attachment to the regular message.

Overview of the Trace Window

In addition to the Trace Window's primary data screen, the window provides the following user options:



1. **Clear button.**
   This clears any data captured from a previous trace session.
2. **Enable Trace box.**
   Check this box to begin recording data in a new trace session. Be sure to check this before attempting to dial out.
3. **Hex Display check box.**
   Check this box to toggle the recorded data between the default ASCII text and hexadecimal format.
   Note: Do not send send a Hex trace to Tactical Software's support team unless requested.

4. **Auto Scroll box.**
   Check this box to allow the display to automatically scroll forward as new session data arrives in real-time in the data screen.
   Note: This is useful for analyzing a COM/IP session as it is underway.

5. **Always On Top box.**
   Check this box to keep the DialOut Port Monitor window always on top of other open applications in Windows.

The Trace Window displays a series of events, one event per line. Every event is tagged with the current time (shown in hour:minute:second.millisecond format), and the COM port where the event occurred. There are three types of events:

1. **Transmit events.**
   Shown in green and preceded by "»", these indicate that the application software transmitted data to the COM port. The remainder of the line shows the data transmitted, either in ASCII or hexadecimal format, depending upon the current display mode.

2. **Receive events.**
   Shown in red and preceded by "«", these indicate that the application software received data from the COM port. The remainder of the line shows the data received, either in ASCII or hexadecimal format, depending upon the current display mode.

3. **Control events.**
   Shown in blue and preceded by "|", these indicate non-data events. These events can include the setting of session parameters (such as Telnet, pcAnywhere compatibility, etc); opening and closing a port; connecting to the modem-sharing server; setting a modem control or status line (DTR, DSR, DCD, etc); and the configuration of baud rate and framing parameters.

   Note: The trace display is updated only once per second to avoid introducing large changes in system timing; as a result, there may be noticeable delays in seeing changes. Do not run COM/IP with tracing enabled except for special situations because of the extra memory (1MB) required for the trace data.

Common Problems | Debugging with Trace Data | InitString Overview

4. Troubleshooting — InitString Overview

Most users of COM/IP will never need to adjust the InitString field found in the main manager window. In many cases, the need for adjustment arises only for incoming calls — for example, BBS systems. Even in these cases, it is often better to configure the init strings in the communications software rather than in the manager window itself.

A Few Examples

The following InitString settings have proven useful for other users of COM/IP. If you are having trouble with your application, you may want to try them.

1. **BBS operators**
   Try this string:

   ```
   AT S1001=1 S1002=1 S1003=7 S1005=0 S1008=23 &D2
   ```

2. **Other applications**
   For other applications besides BBS software, it is often a good idea to enable binary mode, allowing ZMODEM file transfers to work properly. This is done with the following string:

   ```
   AT S1002=1
   ```

3. **Remote port connection at boot-up**
   Occasionally you may want COM/IP to connect to a remote port immediately after your PC boots up. To enable this functionality, put an InitString in the COM/IP manager to "dial" the desired IP and port as soon as COM/IP is loaded. Use this string:

   ```
   ATDT 123.321.123.321[6000]
   ```