Configuring the PA-4T

To continue your PA-4T installation, you must configure the *serial* interfaces. The instructions that follow apply to all supported platforms. Minor differences between the platforms—with Cisco IOS software commands—are noted.

This chapter contains the following sections:

Using the EXEC Command Interpreter

You modify the configuration of your router through the software command interpreter called the *EXEC* (also called enable mode). You must enter the privileged level of the EXEC command interpreter with the **enable** command before you can use the **configure** command to configure a new interface or change the existing configuration of an interface. The system prompts you for a password if one has been set.

The system prompt for the privileged level ends with a pound sign (#) instead of an angle bracket (>). At the console terminal, use the following procedure to enter the privileged level:

Step 1 At the user-level EXEC prompt, enter the **enable** command. The EXEC prompts you for a privileged-level password as follows:

```
Router> enable

Password:
```

Step 2 Enter the password (the password is case sensitive). For security purposes, the password is not displayed.

When you enter the correct password, the system displays the privileged-level system prompt (#):

```
Router#
```

To configure the new interfaces, proceed to the "Configuring the Interfaces" section on page 4-2.

Configuring the Interfaces

After you verify that the new PA-4T is installed correctly (the enabled LED goes on), use the privileged-level **configure** command to configure the new interfaces. Have the following information available:

- Protocols you plan to route on each new interface
- IP addresses, if you plan to configure the interfaces for IP routing
- Bridging protocols you plan to use
- *Clock timing source you plan to use for each new interface and clock speeds for external timing*

If you installed a new PA-4T or if you want to change the configuration of an existing interface, you must enter configuration mode to configure the new interfaces. If you replaced a PA-4T that was previously configured, the system recognizes the new interfaces and brings each of them up in their existing configuration.

For a summary of the configuration options available and instructions for configuring interfaces on a PA-4T, refer to the appropriate configuration publications listed in the "Related Documentation" section on page vi.

You execute configuration commands from the privileged level of the EXEC command interpreter, which usually requires password access. Contact your system administrator, if necessary, to obtain password access. (See the "Using the EXEC Command Interpreter" section on page 4-1 for an explanation of the privileged level of the EXEC.)

This section contains the following subsections:

Shutting Down an Interface

Before you remove an interface that you will not replace, **replace a serial cable**, or replace port adapters, use the **shutdown** command to shut down (disable) the interfaces to prevent anomalies when you reinstall the new or reconfigured
port adapter. When you shut down an interface, it is designated *administratively down* in the **show** command displays.

Follow these steps to shut down an interface:

Step 1 Enter the privileged level of the EXEC command interpreter (also called enable mode). (See the "Using the EXEC Command Interpreter" section on page 4-1 for instructions.)

Step 2 At the privileged-level prompt, enter configuration mode and specify that the console terminal is the source of the configuration subcommands, as follows:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
```

Step 3 Shut down interfaces by entering the **interface serial** subcommand (followed by the interface address of the interface), and then enter the **shutdown** command. Table 4-1 shows the command syntax.

When you have finished, press **Ctrl-Z**—hold down the **Control** key while you press **Z**—or enter **end** or **exit** to exit configuration mode and return to the EXEC command interpreter.

Table 4-1 Syntax of the shutdown Command

Platform	Command	Example

Cisco 7200 series routers	**interface**, followed by the *type* (**serial**) and *slot/port* (port-adapter-slot-number/ interface-port-number)	The example is for interface 0 and interface 1 on a port adapter in port adapter slot 6.
		`Router(config-if)# interface serial 6/0`
		`Router(config-if)# shutdown`
		`Router(config-if)# interface serial 6/1`
		`Router(config-if)# shutdown`
		`Ctrl-Z`
		`Router#`

VIP in Cisco 7500 series routers	**interface**, followed by the *type* (**serial**) and *slot/port adapter/port* (interface-processor-slot-number/ port-adapter-slot-number/ interface-port-number)	The example is for interface 1 and interface 0 on a port adapter in port adapter slot 1 of a VIP installed in interface processor slot 1.
		`Router(config-if)# interface serial 1/1/1`
		`Router(config-if)# shutdown`
		`Router(config-if)# interface serial 1/1/0`
		`Router(config-if)# shutdown`
		`Ctrl-Z`
		`Router#`

Note If you need to shut down additional interfaces, enter the **interface serial** command (followed by the interface address of the interface) for each of the interfaces on your port adapter. Use the **no shutdown** command to enable the interface.

Step 4 Write the new configuration to NVRAM as follows:

```
Router# copy running-config startup-config
[OK]
Router#
```

The system displays an OK message when the configuration has been stored in NVRAM.

Step 5 Verify that new interfaces are now in the correct state (shut down) using the **show interfaces** command (followed by the interface type and interface address of the interface) to display the specific interface. Table 4-2 provides examples.

Table 4-2 Examples of the show interfaces Command

Platform	Command	Example

Cisco 7200 series routers	**show interfaces serial**, followed by *slot/port* (port-adapter-slot-number/ interface-port-number)	The example is for interface 0 on a port adapter in port adapter slot 6.
		`Router# show interfaces serial 6/0`
		`Serial 6/0 is administratively down, line protocol is down`
		`[Additional display text omitted from this example]`

VIP in Cisco 7500 series routers	**show interfaces serial**, followed by *slot/port adapter/port* (interface-processor-slot-number/ port-adapter-slot-number/ interface-port-number)	The example is for interface 0 on a port adapter in port adapter slot 1 of a VIP in interface processor slot 1.
		`Router# show interfaces serial 1/1/0`
		`Serial 1/1/0 is administratively down, line protocol is down`
		`[Additional display text omitted from this example]`

Step 6 Reenable interfaces by doing the following:

a. Repeat Step 3 to reenable an interface. Substitute the **no shutdown** command for the **shutdown** command.

b. Repeat Step 4 to write the new configuration to memory. Use the **copy running-config startup-config** command.

c. Repeat Step 5 to verify that the interfaces are in the correct state. Use the **show interfaces** command followed by the interface type and interface address of the interface.

For complete descriptions of software configuration commands, refer to the publications listed in the "Related Documentation" section on page vi.

Performing a Basic Configuration

Following are instructions for a basic configuration: enabling an interface, specifying IP routing, and **setting the clock rate**. You might also need to enter other configuration subcommands, depending on the requirements for your system configuration and the protocols you plan to route on the interface. For complete descriptions of configuration subcommands and the configuration options available for *serial* interfaces, refer to the appropriate software documentation.

In the following procedure, press the **Return** key after each step unless otherwise noted. At any time you can exit the privileged level and return to the user level by entering **disable** at the prompt as follows:

```
Router# disable

Router>
```

Step 1 Enter configuration mode and specify that the console terminal is the source of the configuration subcommands, as follows:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
```

Step 2 Specify the first interface to configure by entering the **interface serial** subcommand, followed by the interface address of the interface you plan to configure. (The command for your port adapter may be different, for example, **interface atm**.) Table 4-3 provides examples.

Table 4-3 Examples of the interface serial Subcommand

Platform	Command	Example

Cisco 7200 series routers	**interface serial**, followed by *slot/port* (port-adapter-slot-number/ interface-port-number)	The example is for the first interface of a port adapter in port adapter slot 6.
		`Router(config)# interface serial 6/0`
		`Router(config-if)#`

VIP in Cisco 7500 series routers	**interface serial**, followed by *slot/port adapter/port* (interface-processor-slot-number/ port-adapter-slot-number/ interface-port-number)	The example is for the first interface of a port adapter in port adapter slot 1 of a VIP in interface processor slot 1.
		`Router(config)# interface serial 1/1/0`
		`Router(config-if)#`

Step 3 Assign an IP address and subnet mask to the interface (if IP routing is enabled on the system) by using the **ip address** subcommand, as in the following example:

```
Router(config-if)# ip address 10.0.0.0 10.255.255.255
```

Step 4 Add any additional configuration subcommands required to enable routing protocols and set the interface characteristics.

Note If you are configuring a DTE interface, proceed to Step 6. If you are configuring a DCE interface, you need to configure the external clock signal, which is described in Step 5.

The example in Step 5 applies to all systems in which the PA-4T is supported.

Step 5 Set the clock rate using the **clock rate** command. (See the next section, "Configuring Timing [Clock] Signals.")

```
Router(config-if)# clock rate 72000
```

Step 6 Reenable the interfaces using the **no shutdown** command. (See the "Shutting Down an Interface" section on page 4-2.)

Step 7 Configure all additional port adapter interfaces as required.

Step 8 After including all of the configuration subcommands to complete your configuration, press **Ctrl-Z**—hold down the **Control** key while you press **Z**—or enter **end** or **exit** to exit configuration mode and return to the EXEC command interpreter prompt.

Step 9 Write the new configuration to NVRAM as follows:

```
Router# copy running-config startup-config
[OK]
```

```
Router#
```

This completes the procedure for creating a basic configuration.

Configuring Timing (Clock) Signals

All EIA/TIA-232 interfaces support both DTE and DCE mode, depending on the mode of the compact serial cable attached to the port. To use a port as a DTE interface, you need only connect a DTE compact serial cable to the port. When the system detects the DTE mode cable, it automatically uses the external timing signal. To use a port in DCE mode, you must connect a DCE compact serial cable and set the clock speed with the **clock rate** configuration command. You must also set the clock rate to perform a loopback test. This section describes how to set the clock rate on a DCE port and, if necessary, how to invert the clock to correct a phase shift between the data and clock signals. Table 4-4 summarizes some of the commands used to configure the clock rate. See the specific sections that follow for further details.

Table 4-4 Clock Rate Configuration Commands

Purpose	Command	Example	Additional Information

Set standard clock rate.	**clock rate**	The example is for a serial interface with a standard clock rate of 72 kbps.
		`Router(config)# interface serial 3/0`
		`Router(config-if)# clock rate 7200`
			"Setting the Clock Rate"

Set nonstandard clock rate.	**clock rate**	The example is for a serial interface with a nonstandard clock rate of 1234567 kbps.
		`Router(config)# interface serial 3/0`
		`Router(config-if)# clock rate 1234567`
			"Setting the Clock Rate"

Remove a clock rate that has been set.	**no clock rate**	The example is for a serial interface and removes a standard clock rate of 72 kbps.
		`Router(config)# interface serial 3/0`
		`Router(config-if)# clock rate 7200`
		`Router(config-if)# no clock rate`
			"Setting the Clock Rate"

Invert the transmit clock signal.	**invert-txc**	The example inverts the transmit clock signal for a serial interface.
		`Router(config)# interface serial 3/0`
		`Router(config-if)# invert-txc`
			"Inverting the Clock Signal"

Change the clock signal back to its original phase.	**no invert-txc**	The example sets the transmit clock signal for a serial interface back to its original phase.
		`Router(config)# interface serial 3/0`
		`Router(config-if)# no invert-txc`
			"Inverting the Clock Signal"

Invert the data signal.	**invert data**	The example inverts the data stream for both transmit and receive for a serial interface:
		`Router(config)# interface serial 3/0`
		`Router(config-if)# invert data`
			"Configuring NRZI Format"

Setting the Clock Rate

The default operation on a PA-4T DCE interface is for the DCE device to generate its own clock signal (TxC) and send it to the remote DTE. The remote DTE device returns the clock signal to the DCE (the PA-4T). Set the clock rate of an interface using the **clock rate** subcommand, which specifies the clock rate as a bits-per-second value. This subcommand functions in the same way on all supported platforms.

Before you can assign a clock rate, you must use the **interface serial** command (followed by the interface address of the interface) to select the interface to which you want to assign the clock rate value.

In the following example, the clock rate is specified as 72 kbps:

```
Router(config-if)# clock rate 72000
```

The preceding command example applies to all systems in which the PA-4T is supported. Use the **no clock rate** command to remove the clock rate.

Following are the standard clock rates:

1200, 2400, 4800, 9600, 19200 38400, 56000, 64000,
72000, 125000 148000, 250000, 500000, 800000, 1000000,
1300000, 2000000, 4000000, 8000000

When you have finished, press **Ctrl-Z—**hold down the **Control** key while you press **Z**—or enter **end** or **exit** to exit configuration mode and return to the EXEC command interpreter prompt. Then write the new configuration to NVRAM using the **copy running-config startup-config** command.

Inverting the Clock Signal

Systems that use long cables or cables that are not transmitting the TxC (clock) signal might experience high error rates when operating at higher transmission speeds. If a PA-4T DCE port is reporting a high number of error packets, a phase shift might be the problem: inverting the clock might correct this phase shift.

When the EIA/TIA-232 interface is a DTE, the **invert-transmit-clock** command inverts the TxC signal the DTE receives from the remote DCE. When the EIA/TIA-232 interface is a DCE, the **invert-transmit-clock** command inverts the clock signal to the remote DTE port. Use the **no invert-transmit-clock** command to change the clock signal back to its original phase.

Configuring NRZI Format

Table 4-5 summarizes NRZI format commands. For more information, see the remainder of this section.

Table 4-5 NRZI Format Commands

Purpose	Command	Example	Further Information

Enable NRZI encoding.	**nrzi-encoding** [**mark**][1]	The example is for a serial interface with NRZI mark encoding specified:
		`Router(config)# interface serial 3/0`
		`Router(config-if)# nrzi-encoding mark`
		The example is for a serial interface with NRZI space encoding specified:
		`Router(config)# interface serial 3/0`
		`Router(config-if)# nrzi-encoding`
			"Configuring NRZI Format"

Disable NRZI encoding.	**no nrzi-encoding**	The example disables NRZI encoding on a serial interface:
		`Router(config)# interface serial 3/0`
		`Router(config-if)# no nrzi-encoding`
			"Configuring NRZI Format"

1. *Mark* is an optional argument. When *mark* is used, it means there is no signal transition; there is data (a mark) at the beginning of a bit interval. When *mark* is not used, it means there is a signal transition; there is no data (a space) at the beginning of a bit interval.

All EIA/TIA-232 interfaces on the PA-4T support non-return-to-zero (NRZ) and non-return-to-zero inverted (NRZI) formats. Both formats use two different voltage levels for transmission. NRZ signals maintain constant voltage levels with no signal transitions—no return to a zero voltage level—during a bit interval and are decoded using absolute values: 0 and 1. NRZI uses the same constant signal levels but interprets the absence of data—a space—at the beginning of a bit interval as a signal transition and the presence of data—a mark—as no signal transition. NRZI uses relational encoding to interpret signals rather than determining absolute values.

NRZ format—the factory default on all interfaces—is more common. NRZI format is commonly used with EIA/TIA-232 connections in IBM environments.

Enable NRZI encoding on any interface using the **nrzi-encoding** [**mark**] command, where no argument after the command is interpreted as a signal transition, and **mark** is interpreted as no signal transition. This command functions in the same way on all supported platforms. Before you can enable NRZI encoding, you must use the **interface serial** command (followed by the interface address of the interface) to select the interface on which you want to enable NRZI encoding.

In the example that follows, NRZI encoding with a signal transition—no argument—is specified:

```
Router(config-if)# nrzi-encoding
```

In the example that follows, NRZI encoding with no signal transition—with argument—is specified:

```
Router(config-if)# nrzi-encoding mark
```

The preceding command examples apply to all systems in which the PA-4T is supported.
Use the **no nrzi-encoding** command to disable NRZI encoding.

When you have finished, press **Ctrl-Z**—hold down the **Control** key while you press **Z**—or enter **end** or **exit** to exit configuration mode and return to the EXEC command interpreter prompt. Then write the new configuration to NVRAM using the **copy running-config startup-config** command.

For complete command descriptions and instructions, refer to the *Configuration Fundamentals Configuration Guide* publication. For more information, see the "Obtaining Documentation" section on page vii and the.

Configuring Cyclic Redundancy Checks

Table 4-6 summarizes cyclic redundancy check (CRC) commands. For more information, see the remainder of this section.

Table 4-6 CRC Commands

Purpose	Command	Example	Further Information

Enable 32-bit CRC.	**crc** *size*	The example enables 32-bit CRD on a serial interface:
		`Router(config)# interface serial 3/0`
		`Router(config-if)# crc 32`
			"Configuring Cyclic Redundancy Checks"

Return to default 16-bit CRC.	**no crc** ***size***	The example disables 32-bit CRD on a serial interface and returns to the default 16-bit CRC:
		`Router(config)# interface serial 3/0`
		`Router(config-if)# no crc 32`
			"Configuring Cyclic Redundancy Checks"

CRC is an error-checking technique that uses a calculated numeric value to detect errors in transmitted data. All interfaces use a 16-bit CRC (CRC-CITT) by default but also support a 32-bit CRC. The sender of a data frame calculates the frame check sequence (FCS). Before it sends a frame, the sender appends the FCS value to the message. The receiver recalculates the FCS and compares its calculation to the FCS from the sender. If there is a difference between the two calculations, the receiver assumes that a transmission error occurred and sends a request to the sender to resend the frame.

Enable 32-bit CRC using the **crc 32** command. Before you can enable 32-bit CRC, you must use the **interface serial** command (followed by the interface address of the interface) to select the interface on which you want to enable 32-bit CRC. This command functions in the same way on all supported platforms.

In the example that follows, the first serial port on a PA-4T, installed on a VIP in interface processor slot 3 is configured for 32-bit CRC:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface serial 3/1/0
Router(config-int)# crc 32
Ctrl-Z
Router#
```

The preceding command example applies to all systems in which the PA-4T is supported.
Use the **no crc 32** command to disable CRC-32 and return the interface to the default CRC-16 (CRC-CITT) setting.

When you have finished, press **Ctrl-Z**—hold down the **Control** key while you press **Z**—or enter **end** or **exit** to exit configuration mode and return to the EXEC command interpreter prompt. Then write the new configuration to NVRAM using the **copy running-config startup-config** command.

For command descriptions, refer to the *Configuration Fundamentals Configuration Guide* publication. For more information, see the "Obtaining Documentation" section on page vii and the "Obtaining Technical Assistance" section on page viii.

Checking the Configuration

After configuring the new interface, use the **show** commands to display the status of the new interface or all interfaces, and use the **ping** and **loopback** commands to check connectivity. This section includes the following subsections:

Using show Commands to Verify the New Interface Status

Table 4-7 demonstrates how you can use the **show** commands to verify that new interfaces are configured and operating correctly and that the *PA-4T* appears in them correctly. Sample displays of the output of selected **show** commands appear in the sections that follow. For complete command descriptions and examples, refer to the publications listed in the "Related Documentation" section on page vi.

Note The outputs that appear in this document may not match the output you receive when running these commands. The outputs in this document are examples only.

Table 4-7 Using show Commands

Command	Function	Example
show version or **show hardware**	Displays system hardware configuration, the number of each interface type installed, Cisco IOS software version, names and sources of configuration files, and boot images	`Router# show version`
show controllers	Displays all the current interface processors and their interfaces	`Router# show controllers`
show diag *slot*	Displays types of port adapters installed in your system and information about a specific port adapter slot, interface processor slot, or chassis slot	`Router# show diag 2`
show interfaces *type port-adapter-slot-number/ interface-port-number*	Displays status information about a specific type of interface (for example, serial) in a Cisco 7200 series router	`Router# show interfaces serial 1/0`
show interfaces *type interface-processor-slot-number/port-adapter-slot-number /interface-port-number*	Displays status information about a specific type of interface (for example, serial) on a VIP in a Cisco 7500 series router	`Router# show interfaces serial 3/1/0`
show protocols	Displays protocols configured for the entire system and for specific interfaces	`Router# show protocols`
show running-config	Displays the running configuration file	`Router# show running-config`

Table 4-7 Using show Commands (continued)

Command	Function	Example
show startup-config	Displays the configuration stored in NVRAM	`Router# show startup-config`

If an interface is shut down and you configured it as up, or if the displays indicate that the hardware is not functioning properly, ensure that the interface is properly connected and terminated. If you still have problems bringing up the interface, contact a service representative for assistance. This section includes the following subsections:

- Using the show version or show hardware Commands
- Using the show diag Command
- Using the show interfaces Command

Choose the subsection appropriate for your system. Proceed to the "Using the ping Command to Verify Network Connectivity" section on page 4-15 when you have finished using the **show** commands.

Using the show version or show hardware Commands

Display the configuration of the system hardware, the number of each interface type installed, the Cisco IOS software version, the names and sources of configuration files, and the boot images, using the **show version** (or **show hardware**) command.

Note The ouputs that appear in this document may not match the output you receive when running these commands. The outputs in this document are examples only.

Cisco 7200 Series Routers

Following is an example of the **show version** command from a Cisco 7200 series router with the *PA-4T*:

```
Router# show version

Cisco Internetwork Operating System Software
IOS (tm) 7200 Software (C7200-J-M), Version 11.1(472) [biff 105]
Copyright (c) 1986-1996 by cisco Systems, Inc.
Compiled Fri 06-Oct-95 12:22 by mpo
Image text-base: 0x600088A0, data-base: 0x605A4000

ROM: System Bootstrap, Version 11.1(10979) RELEASED SOFTWARE

Router uptime is 4 hours, 22 minutes
System restarted by reload
System image file is "slot0:c7200-j-mz.960421", booted via slot0

cisco 7200 (R4700) processor with 22528K/10240K bytes of memory.
R4700 processor, Implementation 33, Revision 1.0
Last reset from power-on
Bridging software.
X.25 software, Version 2.0, NET2, BFE and GOSIP compliant.
Chassis Interface.
3 Ethernet/IEEE 802.3 interfaces.
3 Network Serial interfaces.
125K bytes of non-volatile configuration memory.
```

```
20480K bytes of Flash PCMCIA card at slot 0 (Sector size 128K).
8192K bytes of Flash internal SIMM (Sector size 256K).
Configuration register is 0x2
```

VIP in Cisco 7500 Series Routers

Following is an example of the **show version** command from a Cisco 7500 series router with the *PA-4T*:

```
Router# show version

Cisco Internetwork Operating System Software
IOS (tm) GS Software (RSP-A), Version 11.1(471) [mpo 105]
Copyright (c) 1986-1995 by cisco Systems, Inc.
Compiled Fri 06-Oct-95 12:22 by mpo
Image text-base: 0x600088A0, data-base: 0x605A4000

ROM: System Bootstrap, Version 5.3(16645) [biff 571], INTERIM SOFTWARE
ROM: GS Bootstrap Software (RSP-BOOT-M), Version 11.1(1.2), MAINTENANCE INTERIME

honda uptime is 4 hours, 22 minutes
System restarted by reload
System image file is "slot0:rsp-a111.471", booted via slot0

cisco RSP2 (R4600) processor with 32768K bytes of memory.
R4600 processor, Implementation 32, Revision 2.0
Last reset from power-on
G.703/E1 software, Version 1.0.
Bridging software.
X.25 software, Version 2.0, NET2, BFE and GOSIP compliant.
Chassis Interface.
1 VIP2 controller (4 Ethernet)(4 Serial).
4 Ethernet/IEEE 802.3 interfaces.
4 Network Serial interfaces.
125K bytes of non-volatile configuration memory.

20480K bytes of Flash PCMCIA card at slot 0 (Sector size 128K).
8192K bytes of Flash internal SIMM (Sector size 256K).
No slave installed in slot 6.
Configuration register is 0x2
```

Using the show diag Command

Display the types of port adapters installed in your system (and specific information about each) using the **show diag** *slot* command, where *slot* is the *port adapter slot* in a Cisco 7200 series router and the *interface processor slot* in a Cisco 7500 series router with a VIP.

Note The ouputs that appear in this document may not match the output you receive when running these commands. The outputs in this document are examples only.

Cisco 7200 Series Routers

Following is an example of the **show diag** *slot* command that shows a PA-4T in port adapter slot 1 of a Cisco 7200 series router:

```
Router# show diag 1

Slot 1:
```

```
Serial port adapter, 4 ports
Port adapter is analyzed
Port adapter insertion time 2d09h ago
Hardware revision 1.1           Board revision A0
Serial number     4294967295    Part number    73-1556-04
Test history      0x0           RMA number     00-00-00
EEPROM format version 1
EEPROM contents (hex):
0x20: 01 02 01 01 FF FF FF FF 49 06 14 04 00 00 00 00
0x30: 50 00 00 00 FF FF FF FF FF FF FF FF FF FF FF FF
```

VIP in Cisco 7500 Series Routers

Following is an example of the **show diag** *slot* command that shows a PA-4T in port adapter slot 0 on a VIP in interface processor slot 9:

```
Router# show diag 9
Slot 9:
        Physical slot 9, ~physical slot 0x6, logical slot 9, CBus 0
        Microcode Status 0xC
        Master Enable, LED, WCS Loaded
        Board is analyzed
        Pending I/O Status: Console I/O
        EEPROM format version 1
        VIP2 controller, HW rev 2.2, board revision UNKNOWN
        Serial number: 03508066  Part number: 73-1684-02
        Test history: 0x00        RMA number: 00-00-00
        Flags: cisco 7000 board; 7500 compatible

        EEPROM contents (hex):
          0x20: 01 15 02 02 00 35 87 62 49 06 94 02 00 00 00 00
          0x30: 12 2B 00 2A 1A 00 00 00 00 00 00 00 00 00 00 00

        Slot database information:
        Flags: 0x4      Insertion time: 0x5314 (01:20:55 ago)

        Controller Memory Size: 8 MBytes

        PA Bay 0 Information:
                Fast-Serial PA, 4 ports
                EEPROM format version 1
                HW rev 1.0, Board revision 4
                Serial number: 02827523  Part number: 73-3417-04
```

Using the show interfaces Command

The **show interfaces** command displays status information (including the physical slot and interface address) for the interfaces you specify. All of the examples that follow specify ***serial*** interfaces.

For complete descriptions of interface subcommands and the configuration options available for Cisco 7200 series and Cisco 7500 VIP interfaces, refer to the publications listed in the "Related Documentation" section on page vi.

Note The ouputs that appear in this document may not match the output you receive when running these commands. The outputs in this document are examples only.

Cisco 7200 Series Routers

Following is an example of the **show interfaces** command for Cisco 7200 series routers. In this example, the four serial interfaces (0 to 3) are on a port adapter in port adapter slot 1; also, most of the status information for each interface is omitted. (Interfaces are administratively shut down until you enable them.)

```
Router# sh int serial 1/0
Serial1/0 is administratively down, line protocol is down
  Hardware is 4T/MC68360
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load 1/255
  Encapsulation HDLC, loopback not set, keepalive set (10 sec)
(display text omitted]

Router# sh int serial 1/1
Serial1/1 is administratively down, line protocol is down
  Hardware is 4T/MC68360
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load 1/255
  Encapsulation HDLC, loopback not set, keepalive set (10 sec)
(display text omitted]

Router# sh int serial 1/2
Serial1/2 is administratively down, line protocol is down
  Hardware is 4T/MC68360
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load 1/255
  Encapsulation HDLC, loopback not set, keepalive set (10 sec)
(display text omitted]

Router# sh int serial 1/3
Serial1/3 is administratively down, line protocol is down
  Hardware is 4T/MC68360
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load 1/255
  Encapsulation HDLC, loopback not set, keepalive set (10 sec)
(display text omitted]
```

Following is an example of the **show interfaces serial** command, which shows all of the information specific to interface port 0 on a PA-4T installed in port adapter slot 1:

```
Router# sh int serial 1/0
Serial1/0 is administratively down, line protocol is down
  Hardware is 4T/MC68360
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load 1/255
  Encapsulation HDLC, loopback not set, keepalive set (10 sec)
  Last input never, output 1d17h, output hang never
  Last clearing of "show interface" counters never
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     24 packets output, 5137 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions     DCD=down  DSR=down  DTR=down  RTS=down  CTS=down
```

VIP in Cisco 7500 Series Routers

Following is an example of the **show interfaces** command used with the VIP. In this example, the four serial interfaces (0 to 3) are on a port adapter in port adapter slot 1 of a VIP in interface processor slot 3; also, most of the status information for each interface is omitted. (Interfaces are administratively shut down until you enable them.)

```
Router# sh int serial 3/1/0
Serial3/1/0 is administratively down, line protocol is down
  Hardware is cyBus Serial, address is 0000.0ca5.2300 (bia 0000.0ca5.2389)
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set, keepalive set (10 sec)

Router# sh int serial 3/1/1
Serial3/1/1 is administratively down, line protocol is down
  Hardware is cyBus Serial, address is 0000.0ca5.2300 (bia 0000.0ca5.238a)
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set, keepalive set (10 sec)

Router# sh int serial 3/1/2
Serial3/1/2 is administratively down, line protocol is down
  Hardware is cyBus Serial, address is 0000.0ca5.2300 (bia 0000.0ca5.238b)
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set, keepalive set (10 sec)

Router# sh int serial 3/1/3
Serial3/1/3 is administratively down, line protocol is down
  Hardware is cyBus Serial, address is 0000.0ca5.2300 (bia 0000.0ca5.238b)
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set, keepalive set (10 sec)
```

Following is an example of the **show interfaces serial** command, which shows all of the information specific to interface 0 on a port adapter in port adapter slot 1 of a VIP in interface processor slot 3:

```
Router# sh int serial 3/1/0
Serial3/1/0 is administratively down, line protocol is down
  Hardware is cyBus Serial, address is 0000.0ca5.2300 (bia 0000.0ca5.2388)
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set, keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 4:00:00
  Last input never, output never, output hang never
  Last clearing of "show interface" counters 2:56:26
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 input packets with dribble condition detected
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets, 0 restarts
     0 output buffer failures, 0 output buffers swapped out
```

Proceed to the next section, "Using the ping Command to Verify Network Connectivity," to check network connectivity of the *PA-4T* and switch or router.

Using the ping Command to Verify Network Connectivity

Using the **ping** command, you can verify that an interface port is functioning properly. This section provides a brief description of this command. Refer to the publications listed in the "Related Documentation" section on page vi for detailed command descriptions and examples.

The **ping** command sends echo request packets out to a remote device at an IP address that you specify. After sending an echo request, the system waits a specified time for the remote device to reply. Each echo reply is displayed as an exclamation point (!) on the console terminal; each request that is not

returned before the specified timeout is displayed as a period (.). A series of exclamation points (!!!!!) indicates a good connection; a series of periods (.....) or the messages [timed out] or [failed] indicate a bad connection.

Following is an example of a successful **ping** command to a remote server with the address 10.0.0.10:

```
Router# ping 10.0.0.10 <Return>
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 10.0.0.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/15/64 ms
Router#
```

If the connection fails, verify that you have the correct IP address for the destination and that the device is active (powered on), and repeat the **ping** command.

Proceed to the next section, "Using loopback Commands," to finish checking network connectivity.

Using loopback Commands

With the loopback test, you can detect and isolate equipment malfunctions by testing the connection between the PA-4T interface and a remote device such as a modem or a CSU/DSU. The **loopback** subcommand places an interface in loopback mode, which enables test packets that are generated from the **ping** command to loop through a remote device or compact serial cable. If the packets complete the loop, the connection is good. If not, you can isolate a fault to the remote device or compact serial cable in the path of the loopback test.

Note You must configure a clock rate on the port *before* performing a loopback test. However, if no cable is attached to the port, the port is administratively up, and the port is in loopback mode; you do not have to configure a clock rate on the port *before* performing a loopback test.

Depending on the mode of the port, issuing the **loopback** command checks the following path:

- When no compact serial cable is attached to the PA-4T interface port, or if a DCE cable is attached to a port that is configured as line protocol up, the **loopback** command tests the path between the network processing engine and the interface port only (without leaving the network processing engine and port adapter).
- When a DTE cable is attached to the port, the **loopback** command tests the path between the network processing engine and the near (network processing engine) side of the DSU or modem to test the PA-4T interface and compact serial cable. (The X.21 DTE interface cable does not support this loopback test; see the following Note.)

Note The X.21 interface definition does not include a loopback definition. On the 4T port adapter, the X.21 DTE interface does not support the loopback function. Because of the internal clock signal present on the PA-4T interfaces, loopback will function on an X.21 DCE interface. This completes the configuration procedure for the new 4T port adapter serial interfaces.