

Configuring Terminal Lines and Modem Support

This chapter describes how to configure the Cisco IOS software for line, terminal, and modem connections. Cisco devices have four types of lines: console, auxiliary, asynchronous, and virtual terminal lines. Different routers have different numbers of these line types. Refer to the hardware user guide shipped with your device for exact configurations.

For a complete description of the commands mentioned in this chapter, refer to the “Terminal Line and Modem Support Commands” chapter in the *Access Services Command Reference*.

For more information about making connections to network hosts through your router, refer to the “Making Connections to Network Devices” chapter later in this publication.

Note Some commands in this chapter have been replaced by new commands. Although they continue to perform their normal functions in the current release, they are no longer documented and support for these commands will cease in a future release. See the *Access Services Command Reference* for detailed command information.

Cisco Line and Interface Paradigm

This section describes the different line types available on Cisco routers. It also describes the relationship between lines and interfaces.

Line Types

Table 1 shows the types of lines that can be configured on Cisco routers.

Table 1 Line Types Available on Cisco Routers

Line Type	Port	Description	Numbering Rules
CON or CTY	Console	Typically used to log in to the router for configuration purposes.	Line 0
AUX	Auxiliary	RS-232 DTE port used as a backup asynchronous port (TTY). Cannot be used as a second console port.	Last TTY line number plus 1

Table 1 Line Types Available on Cisco Routers (Continued)

Line Type	Port	Description	Numbering Rules
TTY	Asynchronous	Same as asynchronous interface. Available on access server models only (Cisco 2509, 10, 11, 12, AS5100, and Cisco 1001). Used typically for remote-node dial-in sessions that use such protocols as SLIP, PPP, and XRemote.	Line 1 through 8 (Cisco 2509, Cisco 2510) or 1 through 16.
VTY	Virtual terminal	Used for incoming Telnet, LAT, X.25 PAD, and protocol translation connections into synchronous ports (such as Ethernet and serial interfaces) on the router.	Last TTY line number plus 2 through the maximum number of VTY lines specified. ¹

1. Increase the number of VTY lines on a router using the **line vty** command.

Relationship between Lines and Interfaces on Cisco Routers

This section describes the relationship between lines and interfaces on Cisco routers. The following sections describe the functions:

- Asynchronous Interfaces and TTY Lines
- Synchronous Interfaces and VTY Lines

Asynchronous Interfaces and TTY Lines

Asynchronous interfaces correspond to physical terminal (TTY) lines. Commands entered in asynchronous interface mode enable you to configure protocol-specific parameters for asynchronous interfaces; commands entered in line configuration mode permit you to configure the physical aspects of the line's port.

For example, to enable IP resources to dial in to a network, configure the lines and asynchronous interfaces as follows:

Step 1 On lines, you configure the physical aspect of a port. You might enter the following commands to configure lines 1 through 16, all asynchronous TTY lines on a Cisco 2511 access server:

```
line 1 16
 login local
 modem inout
 speed 115200
 flowcontrol hardware
 ! configures the line to autosense PPP; physical line attribute
 autoselect ppp
```

Step 2 On asynchronous interface 1, you configure your protocol-specific commands. You might enter the following commands:

```
interface async 1
 encapsulation ppp
 async mode interactive
 async dynamic address
 async dynamic routing
 async default ip address 198.192.16.132
 ppp authentication chap
```

The remote node services SLIP, PPP, and XRemote are configured in asynchronous interface mode. AppleTalk Remote Access (ARA) is configured in line configuration mode on virtual terminal (VTY) lines or TTY lines.

For more information about configuring interfaces, refer to the “Configuring Interfaces” chapter in the *Configuration Fundamentals Configuration Guide*.

Only Cisco access server products have multiple asynchronous interfaces.

Synchronous Interfaces and VTY Lines

Virtual terminal (VTY) lines provide access to the router through a synchronous interface. VTY lines do not correspond to synchronous interfaces in the same way that TTY lines correspond to asynchronous interfaces. This is because VTY lines are created dynamically on the router, whereas TTY lines are static physical ports. When a user connects to the router on a VTY line, that user is connecting into a *virtual* port on an interface. You can have multiple virtual ports for each synchronous interfaces.

For example, several Telnet connections can be made to an interface (such as an Ethernet or serial interface).

The number of VTY lines available on a router are defined using the **line vty** *number-of-lines* global configuration command.

For more information about the relationship between asynchronous interfaces and the AUX port, refer to the “Configuring Interfaces” chapter in the *Configuration Fundamentals Configuration Guide*.

Use the **show line** command, to see the status of each of the lines available on a router (see Figure 8).

Figure 8 Sample Show Line Output Showing CTY, TTY, , and VTY Line Statistics

Autoselect state	Rotary group #	Access class in/out									
sankara> show line											
	Tty Typ	Tx/Rx	A	Modem	Roty	ACCO	ACCI	Uses	Noise	Overruns	
	* 0	CTY	-	-	-	-	-	0	0	0/0	
	* 1	TTY	115200/115200	-	inout	-	4	-	31	26	0/0
	* 2	TTY	115200/115200	-	inout	-	21630	-	37	23	0/0
Absolute line number	A 3	TTY	115200/115200	-	inout	-	25	-	10	24	1/0
	* 4	TTY	115200/115200	-	inout	-	4	-	20	63	1/0
	* 5	TTY	115200/115200	-	inout	-	32445	-	18	325	22/0
	A 6	TTY	115200/115200	-	inout	-	25	-	7	0	0/0
Line speed	I 7	TTY	115200/115200	-	inout	-	6	-	6	36	1/0
	I 8	TTY	115200/115200	-	inout	-	-	-	3	25	3/0
	* 9	TTY	115200/115200	-	inout	-	4	-	2	0	0/0
	A 10	TTY	115200/115200	-	inout	-	56	-	2	470	216/0
	I 11	TTY	115200/115200	-	inout	-	4	-	31	26	0/0
	I 12	TTY	115200/115200	-	inout	-	4	-	31	26	0/0
	I 13	TTY	115200/115200	-	inout	-	4	-	31	26	0/0
	I 14	TTY	115200/115200	-	inout	-	4	-	31	26	0/0
	I 15	TTY	115200/115200	-	inout	-	4	-	31	26	0/0
	I 16	TTY	115200/115200	-	inout	-	4	-	31	26	0/0
	17	AUX	9600/9600	-	-	-	-	-	2	1	2/104800
	* 18	VTY	9600/9600	-	-	-	-	-	103	0	0/0
	19	VTY	9600/9600	-	-	-	-	-	6	0	0/0
This is VTY2 (3rd VTY) line 20	20	VTY	9600/9600	-	-	-	-	-	1	0	0/0
	21	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	22	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	23	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	24	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	25	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	26	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	27	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	28	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	29	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	30	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	31	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	32	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	33	VTY	9600/9600	-	-	-	-	-	0	0	0/0

Absolute versus Relative Line Numbers

When you enter line configuration mode, as described in the “Enter and Exit Line Configuration Mode” section later in this chapter, you can specify an *absolute line number* or a *relative line number*. For example, in Figure 8, absolute line 20 is VTY2 (line 18 is VTY0). Referring to lines in a relative format is often easier than attempting to recall the absolute number of a line on a large system. Internally, the router uses absolute line numbers.

You can view all of the absolute and relative line numbers with the **show users all EXEC** command. In the following sample display, absolute line numbers are listed at the far left under the heading “Line.” Relative line numbers are in the third column, after the line type. In this example, the second virtual terminal line, vty 1, is absolute line number 3.

Line	User	Host(s)	Idle Location
0	con	0	
1	aux	0	
2	vtty	0	
3	vtty	1	
4	vtty	2	
5	vtty	3	
6	vtty	4	

Compare the line numbers in this sample display to the output from the **show line** command, as shown in Figure 8.

Configure Modem Support

This section contains information to help you to configure most popular modems to work with asynchronous interfaces on Cisco routers (including the Cisco 2509 through 2512, the Cisco 2520 through 2523, and the Cisco AS5100 and AS5200) or with the console or auxiliary port on Cisco routers that do not have multiple asynchronous interfaces.

Fundamental Line Configuration Tasks

When configuring TTY lines for modem support, most customers typically configure a set of core commands. These commands are performed in line configuration mode and set the following parameters:

- Line speed
- Flow control on the line
- Enable the modem to initiate outgoing calls or accept incoming calls
- Automatically configure your modem to communicate with the router
- Enable the Cisco IOS software to automatically detect the incoming protocol

To configure lines for modem support, perform the following steps, beginning in global configuration mode:

Task	Command
Step 1 Specify the TTY line number. Refer to Table 1 earlier in this chapter for line numbering rules.	line tty <i>line-number</i>
Step 2 Set the line speed to the highest common speed for the modem and the router port. ¹	speed 115200
Step 3 Set RTS/CTS flow control on the line.	flowcontrol hardware
Step 4 Configure the line to drop the connection when CD is lost (cycle DTR to close the connection). ²	modem inout
Step 5 Discover the type of modem attached to your router.	modem autoconfigure discovery <i>terminal</i>
Step 6 Automatically configure the modem.	modem autoconfigure <i>type</i>
Step 7 Configure a line to automatically start an ARA, PPP, or SLIP session when it detects the appropriate start packet.	autoselect { arap ppp slip during login }

1. If you are configuring the AUX port, the maximum speed is 38400.

2. For commands used in other modem scenarios, see Table 1.

The previous configuration parameters assume the following:

- The modem always communicates with the router at the speed you establish in Step 2. You must therefore LOCK SPEED on the modem.
- The carrier detect (CD) signal of the modem reflects the real state of the carrier, and the modem will hang up when a Cisco device drops the data terminal ready (DTR) signal.

You can set many other parameters. These are the most common commands necessary for basic modem communications.

Note If possible, avoid using the **autobaud** command. You might experience problems with the router if you use it.

Table 2 lists the different modem commands you can substitute in Step 4 of the preceding task table, depending on the task you are performing.

Table 2 Cisco IOS Modem Commands used in Different Scenarios

Scenario	Command
Connect the access server directly to a terminal.	No modem needed, so no modem command needed.
Connect the access server directly to a terminal, but close the session when the terminal is shut down (no reverse TCP allowed).	modem dialin
Connect the access server directly to a terminal, but close the session when the terminal is shut down (reverse TCP is allowed).	modem inout
Attach a modem to an access server for incoming calls only.	modem dialin
Attach a modem to an access server for both incoming and outgoing calls.	modem inout
Attach a printer device to an access server; connections refused when the printer is off.	modem host
Access a host device through an access server; DTR is established while the connection is established. Drop the connection when the user logs out.	modem host
Automatically dial a modem to a remote site.	modem host

Automatically Configure Modems

The Cisco IOS software can issue initialization strings automatically for most types of modems. A modem initialization string is a series of parameter settings that are sent to your modem to configure it to interact with the access server in a specified way. The Cisco IOS software defines seven initialization strings that have been found to properly initialize most modems so that the modems function properly with Cisco access servers. These initialization scripts have the following names:

- Codex_3260
- Usr_courier
- Usr_sportster
- Hayes_optima
- Global_village
- Viva
- Telebit_t3000

If you do not know which of these modem strings is appropriate for your modems, refer to the “Discover the Modem Automatically” section.

If you know that your modem can be configured using an initialization string from one of these scripts, refer to the “Initialize the Modem Automatically” section.

You can also manually create modem scripts (called chat scripts). For more information, refer to the “Configure Chat Scripts for Asynchronous Lines” section later in this chapter.

Discover the Modem Automatically

The Cisco IOS software contains a database of modem capabilities for most modems. You can configure a router to automatically attempt to discover what kind of modem is connected to the line and then to configure that modem. To automatically discover which of the supported modem strings properly initializes your modem and then initialize the modem, perform the following task in line configuration mode:

Task	Command
Discover the type of modem attached to your router.	modem autoconfigure discovery <i>terminal</i>

The Cisco IOS software first tries the first of the supported modem strings listed earlier in this section to see if the modem initializes properly. If not, the Cisco IOS software cycles to the next string and repeats the process until the appropriate string is found. If none of the strings properly initializes the modem, you must manually initialize the modem, as described in the “Manually Configure Modems” section later in this chapter.

Note If you know which string properly initializes your modem, refer to the following section “Initialize the Modem Automatically,” which describes how to specify a modem type. If you specify a modem type, initialization proceeds more quickly than if you require the Cisco IOS software to discover the appropriate string before it initializes the modem.

Initialize the Modem Automatically

After you discover which modem string properly initializes your modems, you can initialize the modems automatically. To initialize one or more attached modems, perform the following task in line configuration mode:

Task	Command
Automatically initialize the modem.	modem autoconfigure type <i>modem-string</i>

Display the List of Known Modems

To display the list of modems for which the router has entries, perform the following task in EXEC mode:

Task	Command
Display the list of modems for which the router has entries.	show modemcap <i>modem-name</i>

Change the Values in a Modem Setting

You can change a modem value that was returned from the **show modemcap** command. For example, you might want to add the factory default, **&F**, entry to the configuration file. To change the values in a modem setting, perform the following task in line configuration mode:

Task	Command
Change a modem value that was returned from the show modemcap command.	modemcap edit attribute value

Configure one attribute of one modem at a time. See the modem-capability values defined by the **show modemcap** command.

Manually Configure Modems

You must ensure that a modem always communicates with the router at a set speed, regardless of the speed of any incoming modem connection. To do so, issue commands to the modem starting with “**AT**” to specify the speed at which you want the modem to communicate with the router.

Using **AT** modem commands also requires that you set the modem to lock on the data terminal equipment (DTE) speed. The following sections describe how to accomplish this task. Specifically, this section describes bit-rate maximum speeds for Cisco devices, then describes the following tasks:

- Configure a Reverse Connection
- Issue a Modem String

For information about modem settings to be used in strings, refer to the appendix “Configuring Modem Support and Chat Scripts” in the *Access Services Command Reference*.

Bit-Rate Maximums for Cisco Devices

Bit-rate information for Cisco access servers and routers is as follows:

- 38400 is the maximum speed for AUX ports.
- 115200 is the maximum speed for the Cisco routers with multiple asynchronous interfaces.

If flow control is not available on your modem, use 9600 baud as the maximum speed.

Note If you are routing through the AUX port, each character generates a processor interrupt. An abnormally high load on the CPU can be resolved by using a lower AUX port speed.

Configure a Reverse Connection

If possible, configure your modems by establishing a reverse connection. To make a reverse connection, issue this Telnet command from anywhere on the network that can ping *x.x.x.x*:

```
telnet x.x.x.x 20yy
```

where *x.x.x.x* is any active, connected, and up interface on the router, and *yy* is the decimal line number to which you want to connect.

Note The router AUX port is 01. The AUX port on a router with asynchronous interfaces is the last TTY line plus 1. For example, on a 16-port access server (such as a Cisco 2511), the AUX port is port 17.

If you receive a connection refused message, refer to the “Modem Troubleshooting Tips” section in the “Configuring Modem Support and Chat Scripts” appendix in the *Access Services Command Reference* for a likely cause and resolution.

Issue a Modem String

Establish a connection to the modem and specify a modem string by performing the following steps:

Step 1 Connect to the modem using the same speed at which the Cisco router port will be set, as described previously in the “Configure a Reverse Connection” section. This ensures that you are at the same line speed as the Cisco router.

Step 2 Issue an **AT** command configured with the appropriate information for your modem. (See the “Configuring Modem Support and Chat Scripts” appendix in the *Access Services Command Reference*.)

Start with commands listed as required for all modems and then add the EC/Compression pair (either “BEST,” or “NO”) that best suits your need. For applications that are primarily file transfer, the “BEST” pair is recommended. For connections that are primarily ARA, Xremote, or interactive packet-protocol (SLIP/PPP) traffic, the “NO” pair is recommended. This is the minimum configuration required to connect the modem.

Step 3 If you have an AUX port (or no modem control), add commands listed under the column Settings for Use with AUX Port in the table “AUX and Platform Specific Settings” in the appendix “Configuring Modem Support and Chat Scripts” in the *Access Services Command Reference*. Remember to limit the line speed to 9600 bps if you have no flow control.

Step 4 End the string with the **&w** string.

Example String

A Microcom modem with best error correction and compression is configured as follows:

```
AT&FS0=1&C1&D3\Q3\J0\N6%C1\Q2&W
```

After having configured your modem to function with the router, refer to the section “Prepare to Configure Lines” later in this chapter, then the “Automatically Configure Modems” chapter earlier in this chapter.

Line Configuration Task List

The following sections describe line configuration tasks. One of the first things you need to do for line configuration is to set up the lines for the terminals or other asynchronous devices attached to them. The parameters for each line are configured next. However, the tasks you perform and the order in which you perform them are determined entirely by the requirements of your network environment.

- Prepare to Configure Lines
- Configure Protocols to Start Up Automatically

- Establish and Control the EXEC Process
- Configure Flow Control
- Configure Modem Control
- Define Terminal Operation Characteristics
- Define a Command String for Automatic Execution
- Connections to an Individual Line
- Configure Rotary Groups
- LPD Protocol Support
- Configure Chat Scripts for Asynchronous Lines
- Call Back Asynchronous Clients
- Enable NASI Clients to Access Network Resources
- Display Terminal Banner Messages
- Enable Modem Services Specific to the Cisco AS5200
- Configure Virtual Asynchronous Interfaces
- Line Configuration Examples
- Cisco AS5200 Configuration Examples
- Callback Examples

See the end of this chapter for configuration examples. See the Cisco IOS command references for information about the commands listed in this chapter.

Prepare to Configure Lines

You set terminal-specific parameters in line configuration mode. Line configuration commands can configure physical terminal (TTY) lines, virtual terminal (VTY) lines, the auxiliary (AUX) port, or the console (CON) port. This section describes how to enter line configuration mode so that you can configure these line types. Specifically, this section describes the following tasks:

- Enter and Exit Line Configuration Mode
- Enable the Auxiliary Port
- Create Additional Virtual Terminal Lines
- Eliminate VTY Lines

Enter and Exit Line Configuration Mode

To enter line configuration mode, perform the following steps:

Task	Command
Step 1 At the privileged EXEC prompt, enter configuration mode from the terminal.	configure ¹ [terminal]
Step 2 From global configuration mode, begin to configure a line by entering line configuration mode.	line [aux console tty vty] line-number [ending-line-number]

Task	Command
Step 3 Enter commands listed in this chapter to configure the line.	Use the commands listed in this chapter.
Step 4 Exit line configuration mode and return to EXEC mode.	exit
Step 5 Save the configuration changes to nonvolatile random access memory (NVRAM).	copy running-config startup-config¹

1. These commands are documented in the “System Image and Configuration File Load Commands” chapter in the *Configuration Fundamentals Command Reference*.

Once a line is configured, you can check its status by entering the **show users all** EXEC command. To view configuration parameters before saving them to your startup configuration, enter the **show running config** EXEC command. To view configuration parameters after you saved them, enter the **show startup-config** EXEC command.

To leave line configuration mode, enter **exit**. You can also enter another configuration mode by issuing the mode-specific command (such as **interface type number**).

Enable the Auxiliary Port

The AUX port is typically configured as an asynchronous serial interface on routers without built-in asynchronous interfaces. To configure the AUX port as an asynchronous interface, configure it first as an auxiliary line with the **line aux 1** global configuration command.

The AUX port sends a DTR signal only when a Telnet connection is established. To understand the differences between standard asynchronous interfaces and AUX ports configured as an asynchronous interface, refer to Table 3. To enable the auxiliary port, perform the following task:

Task	Command
Enable the auxiliary serial DTE port.	line aux line-number

You cannot use the auxiliary (AUX) port as a second console port. To use the AUX port as a console port, you must order a special cable from your technical support personnel.

On an access server, you can configure any of the available asynchronous interfaces (1 through 8, 16, or 48). The auxiliary port (labeled AUX on the back of the product) can also be configured as an asynchronous serial interface, although performance on the AUX port is much slower than on standard asynchronous interfaces and does not support some features. Table 3 illustrates why asynchronous interfaces permit substantially better performance than AUX ports configured as asynchronous interfaces.

Table 3 Differences between the Auxiliary (AUX) Port and the Asynchronous Port

Feature	Asynchronous Interface	Auxiliary Port
Maximum speed	115200 kbps	38400 kbps
Supported Platforms	Cisco 2509, 2510, 2511, 2512, AS5100, AS5200	All Cisco routers
Supports DMA buffering ¹	Yes	No
PPP framing on chip ²	Yes	No

Table 3 Differences between the Auxiliary (AUX) Port and the Asynchronous Port

Feature	Asynchronous Interface	Auxiliary Port
IP fast switching ³	Yes	No

1. Direct Memory Access (DMA) buffering moves data packets directly to and from system memory without interrupting the main central processing unit (CPU). This process removes overhead from the CPU and increases overall system performance.
2. PPP framing on a hardware chip removes overhead from the router's CPU, which enables the router to sustain 115.2 kbps throughput on all asynchronous ports simultaneously.
3. After the destination of the first IP packet is added to the fast switching cache, it is fast switched to and from other interfaces with minimal involvement from the main processor.

On routers without built-in asynchronous interfaces, only the AUX port can be configured as an asynchronous serial interface. To configure the AUX port as an asynchronous interface, you must also configure it as an auxiliary line with the **line aux 1** command. Access servers do not have this restriction. Use the line command with the appropriate line configuration commands for modem control, such as speed.

Only IP packets can be sent across lines configured for SLIP. PPP supports transmission of IP, IPX, and AppleTalk packets on an asynchronous serial interface.

Create Additional Virtual Terminal Lines

The Cisco IOS software permits you to create as many virtual terminal (VTY) lines as you need. To understand what a VTY line is, refer to the section “Cisco Line and Interface Paradigm” earlier in this chapter.

The **line vty** command accepts any *line number* larger than 5 up to the maximum number of lines supported by your router with its current configuration. The Cisco IOS software dynamically creates all of the new VTY lines between the current highest-numbered line and the number that you specify with the **line vty** command. You can then configure those lines with additional line configuration commands.

See the “Creating Additional Virtual Terminal Lines Example” section at the end of this chapter for an example of how to add virtual terminal lines.

Eliminate VTY Lines

To delete VTY lines, perform the following steps:

Task	Command
Step 1 At the privileged EXEC prompt, enter configuration mode from the terminal.	configure¹ terminal
Step 2 From global configuration mode, delete VTY lines.	no line vty line-number

1. This command is documented in the “System Image and Configuration File Load Commands” chapter in the *Configuration Fundamentals Command Reference*.

The Cisco IOS software deletes all VTY lines greater than the *line-number* variable that you specified.

You cannot delete VTY lines that are in use; attempting to do so results in a warning message. VTY lines should be deleted on an idle system only. See the “Eliminating Virtual Terminal Lines Example” section at the end of this chapter for an example of how to eliminate VTY lines.

Configure Protocols to Start Up Automatically

To configure the Cisco IOS software to allow an AppleTalk Remote Access (ARA), Point-to-Point Protocol (PPP), or Serial Line Internet Protocol (SLIP) session to start automatically, perform the following task in line configuration mode:

Task	Command
Configure a line to automatically start an ARA, PPP, or SLIP session.	autoselect { arap ppp slip during login }

The **autoselect** command enables the Cisco IOS software to start a process automatically when a start character is received. The Cisco IOS software detects either a Return character (which is the start character for an EXEC session, or the start character for the ARA protocol).

The **autoselect** command bypasses the login prompt and enables the specified session to begin automatically. However, by entering the **autoselect** command with the **during login** keyword, the username or password prompt appears without pressing the Return key. While the username or password prompt is displayed, you can choose either to answer these prompts or to send packets from an autoselected protocol. For a line configuration example, see the “Basic Line Configuration Example” section at the end of this chapter.

Note When you use the **autoselect** command, the activation character should be set to the default Return, and `exec-character-bits` to 7. If you change these defaults, the application cannot recognize the activation request.

Establish and Control the EXEC Process

By default, the Cisco IOS software starts an EXEC process on all lines. However, you can control EXEC processes, as follows:

- Turn the EXEC process on or off.

A serial printer, for example, should not have an EXEC session started.

- Set the idle terminal timeout interval.

The EXEC command interpreter waits for a specified amount of time to receive user input. If no input is detected, the EXEC facility resumes the current connection. If no connections exist, it returns the terminal to the idle state and disconnects the incoming connection. To control the EXEC process, perform the following tasks in line configuration mode:

Task	Command
Turn on EXEC processes.	exec
Set the idle terminal timeout interval.	exec-timeout <i>minutes</i> [<i>seconds</i>]

Configure Flow Control

Dial-up modems that operate over normal dial-up telephone lines at speeds of 28800 bits per second and higher are now available. These modems do not operate at a guaranteed throughput; instead, they operate at a speed determined by the quality of the line, the effectiveness of data compression algorithms of the transmitted data, and other variables.

These modems use hardware flow control to stop the data from reaching the host by toggling an RS-232 signal when the modems cannot accept any more data. For information about setting up the RS-232 line for hardware flow control, see the hardware installation and maintenance manual for your product.

In addition to hardware flow control, dial-up modems require special software handling. For example, they must be configured to create an EXEC session when a user dials in and to hang up when the user exits the EXEC. These modems also must be configured to close any existing network connections if the telephone line hangs up in the middle of a session.

You can set both hardware and software flow control between the router and devices attached to it. Both types of flow control are bidirectional. When you specify the **software** flow control option, an additional keyword specifies the direction: **in** causes the router to listen to flow control from the attached device, and **out** causes the router to send flow control information to the attached device. If you do not specify a direction, the router enables software flow control in both directions.

For software flow control, the default stop and start characters are Ctrl-S and Ctrl-Q (XOFF and XON), respectively. However, you can define other characters or character sequences to signal the start and end of data transmission when software flow control is in effect. This capability is useful for providing control of data over an asynchronous serial line.

To configure flow control between the router and attached device, perform one or more of the following tasks in line configuration mode:

Task	Command
Set the terminal flow control.	flowcontrol { none software [in out] hardware [in out] } ¹
(Optional.) In EXEC mode, display informational messages about modem control events (such as signal transitions and autobaud progress) on the console terminal.	debug modem ²
(Optional.) In EXEC mode, display the status of a line. In the detailed command output, a status line with “Idle” identifies inactive modem dialin lines and all other modem lines; a status line with “Ready” identifies lines in use.	show line ³
Set the flow control start character.	start-character <i>ascii-number</i> ⁴
Set the flow control stop character.	stop-character <i>ascii-number</i> ¹

1. If you want to enable outgoing hardware flow control based on the CTS input, specify the **hardware** option.
2. See the *Debug Command Reference*.
3. These commands are documented in the “User Interface Commands” chapter in the *Configuration Fundamentals Command Reference*.
4. This command is seldom used. Typically, you only need to use the **flowcontrol** command.

Configure Modem Control

Cisco routers use six EIA/TIA-232 signals for each port, so one 50-pin Telco, RJ-11, or RJ-45 connector can support eight sessions. The router can support the most popular forms of modem control and hardware flow control, as well as high-speed dial-up modems.

The EIA/TIA-232 output signals are Transmit Data (TXDATA), Data Terminal Ready (DTR), and Ready To Send (RTS, 2500 only). The input signals are Receive Data (RXDATA), Clear to Send (CTS), and RING. The sixth signal is ground. Depending on the type of modem control your modem uses, these names may or may not correspond to the standard EIA/TIA-232 signals.

Dial-up modems that operate over normal dial-up telephone lines at speeds of 28800 bits per second (bps) use hardware flow control to stop the data from reaching the host by toggling an EIA/TIA-232 signal when their limit is reached.

In addition to hardware flow control, dial-up modems require special software configuring. For example, they must be configured to create an EXEC session when a user dials in and to hang up when the user exits the EXEC. These modems also must be configured to close any existing network connections if the telephone line hangs up in the middle of a session.

The Cisco IOS software supports hardware flow control on its CTS input signal, which is also used by the normal modem handshake.

The following modem line characteristics and modem features are discussed in the following sections:

- Configure Automatic Dialing
- Automatically Answer a Modem
- Support Dial-In and Dial-Out Modems
- Configure a Line Timeout Interval
- Close Modem Connections
- Configure Automatic Line Disconnect
- Support Old-Style Dial-In Modems
- Support Reverse Modem Connections and Prevent Incoming Calls

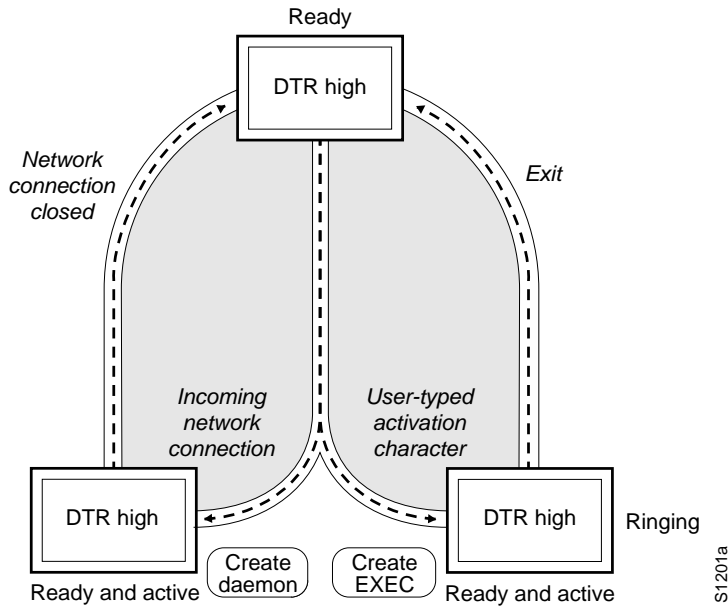
Signal and line state diagrams accompany some of the tasks in the following sections to illustrate how the modem control works. The diagrams show two processes:

- The “create daemon” process creates a TTY daemon that handles the incoming network connection.
- The “create EXEC” process creates the process that interprets user commands. (Refer to Figure 9 through Figure 14.)

In the diagrams, the current signal state and the signal the line is watching are listed inside each box. The state of the line (as displayed by the **show line EXEC** command) is listed next to the box. Events that change that state appear in italics along the event path, and actions that the software performs are described within the ovals.

Figure 9 illustrates line states when no modem control is set. The DTR output is always high, and CTS and RING are completely ignored. The Cisco IOS software starts an EXEC session when the user types the activation character. Incoming TCP connections occur instantly if the line is not in use and can be closed only by the remote host.

Figure 9 EXEC and Daemon Creation on a Line with No Modem Control



Configure Automatic Dialing

With the dial-up capability, you can set a modem to dial the phone number of a remote router automatically. This feature offers cost savings because phone line connections are made only when they are needed—you only pay for using the phone line when there is data to be received or sent. To configure a line for automatic dialing, perform the following task in line configuration mode:

Task	Command
Configure a line to initiate automatic dialing.	modem dtr-active

Using the **modem dtr-active** command causes a line to raise DTR signal only when there is an outgoing connection (such as reverse Telnet, NASI, or DDR), rather than leave DTR raised all the time. When raised, DTR potentially tells the modem that the router is ready to accept a call.

Automatically Answer a Modem

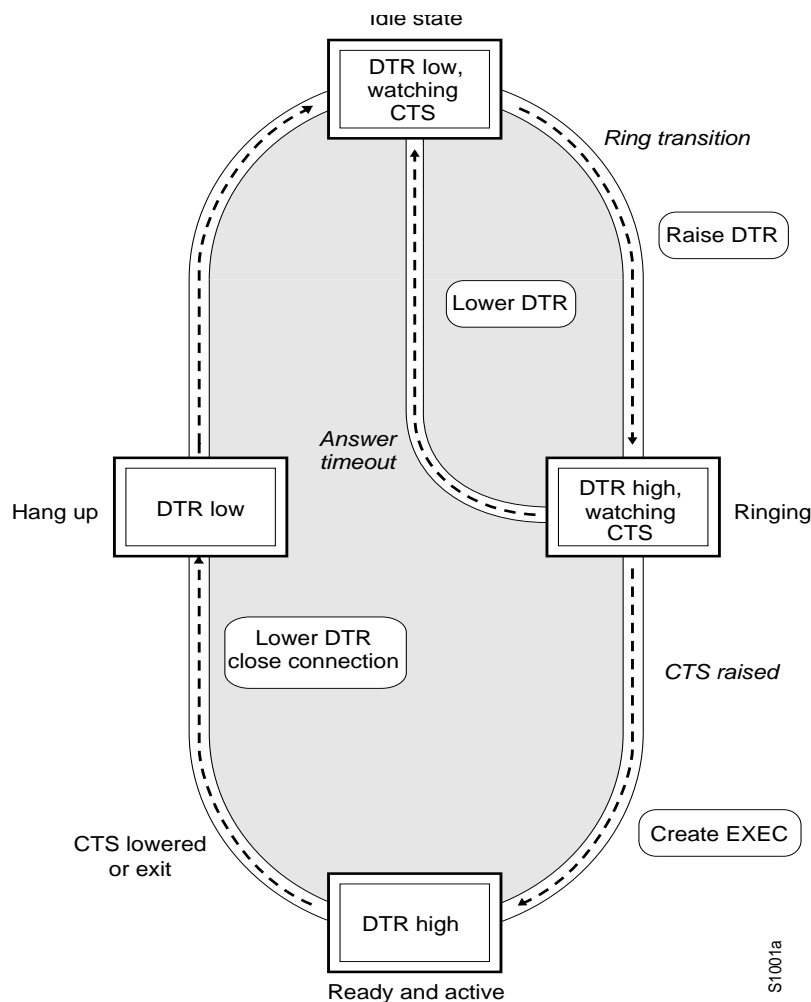
You can configure a line to answer a modem automatically. You also can configure the modem to answer the telephone on its own (as long as DTR is high), drop connections when DTR is low, and use its Carrier Detect (CD) signal to accurately reflect the presence of carrier. (Configuring the modem is a modem-dependent process.) Wire the modem’s CD signal (generally pin-8) to the router’s RING input (pin-22), and perform the following task in line configuration mode:

Task	Command
Configure a line to automatically answer a modem.	modem dialin

You can turn on the modem’s hardware flow control independently to respond to the status of the router’s CTS input. Wire CTS to whatever signal the modem uses for hardware flow control. If the modem expects to control hardware flow in both directions, you might also need to wire the modem’s flow control input to some other signal that the router always has high (such as the DTR signal).

Figure 10 illustrates the **modem dialin** process with a high-speed dial-up modem. When the Cisco IOS software detects a signal on the RING input of an idle line, it starts an EXEC or autobaud process on that line. If the RING signal disappears on an active line, the Cisco IOS software closes any open network connections and terminates the EXEC facility. If the user exits the EXEC or the software terminates because of no user input, the line makes the modem hang up by lowering the DTR signal for 5 seconds. After 5 seconds, the modem is ready to accept another call.

Figure 10 EXEC Creation on a Line Configured for a High-Speed Dial-up Modem



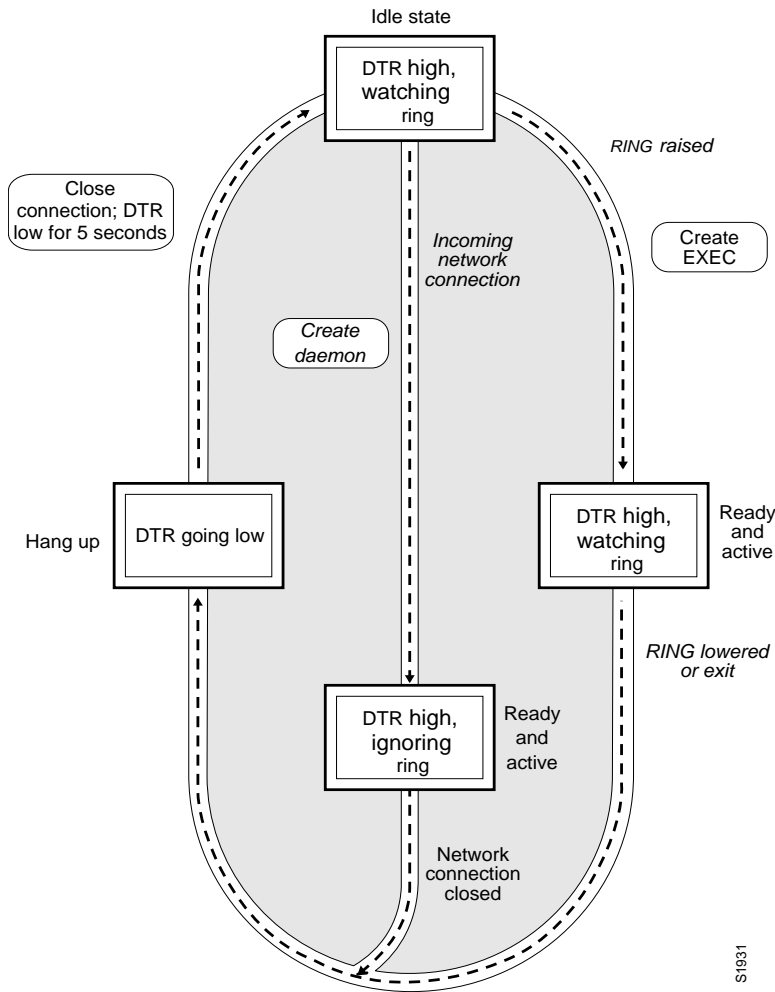
Support Dial-In and Dial-Out Modems

You can configure a line for both incoming and outgoing calls by performing the following task in line configuration mode:

Task	Command
Configure a line for both incoming and outgoing calls.	modem inout

Figure 11 illustrates the **modem inout** command. If the line is activated by raising the data set ready (DSR) signal, it functions exactly as a line configured with the **modem dialin** line configuration command described in the “Automatically Answer a Modem” section earlier in this chapter. If the line is activated by an incoming TCP connection, the line functions similarly to lines not used with modems.

Figure 11 EXEC and Daemon Creation on a Line Configured for Incoming and Outgoing Calls



Note If your system incorporates dial-out modems, consider using access lists to prevent unauthorized use.

Configure a Line Timeout Interval

You can change the interval that the Cisco IOS software waits for the CTS signal after raising the DTR signal in response to the DSR (the default is 15 seconds). To do so, perform the following task in line configuration mode. The timeout applies to the **modem callin** command only.

Task	Command
Configure modem line timing.	modem answer-timeout <i>seconds</i>

Note The DSR signal is called RING on older ASM-style chassis.

Close Modem Connections

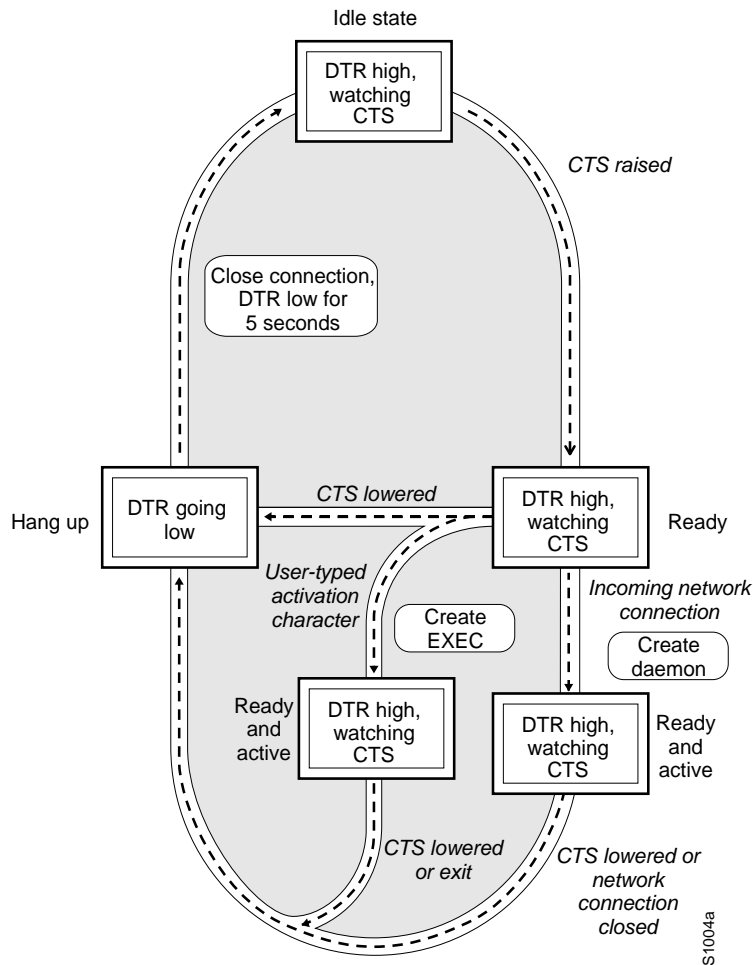
You can configure a line to close connections from a user's terminal when the terminal is turned off and prevent inbound connections to devices that are out of service. To do so, perform the following task in line configuration mode:

Task	Command
Configure a line to close connections.	modem cts-required

Figure 12 illustrates the **modem cts-required** command operating in the context of a continuous CTS signal. This form of modem control requires that the CTS signal be high for the entire session. If CTS is not high, the user's input is ignored and incoming connections are refused (or sent to the next line in a rotary group).

Note For the Cisco IOS software to reliably detect a CTS signal change, the CTS signal must remain in the new state for at least 1 full second.

Figure 12 EXEC and Daemon Creation on a Line Configured for Continuous CTS



Configure Automatic Line Disconnect

You can configure automatic line disconnect by performing the following task in line configuration mode:

Task	Command
Configure automatic line disconnect.	autohangup

The **autohangup** command causes the EXEC facility to issue the **exit** command when the last connection closes. This feature is useful for UNIX-to-UNIX copy program (UUCP) applications, because UUCP scripts cannot issue a command to hang up the telephone. This feature is not often used at this point.

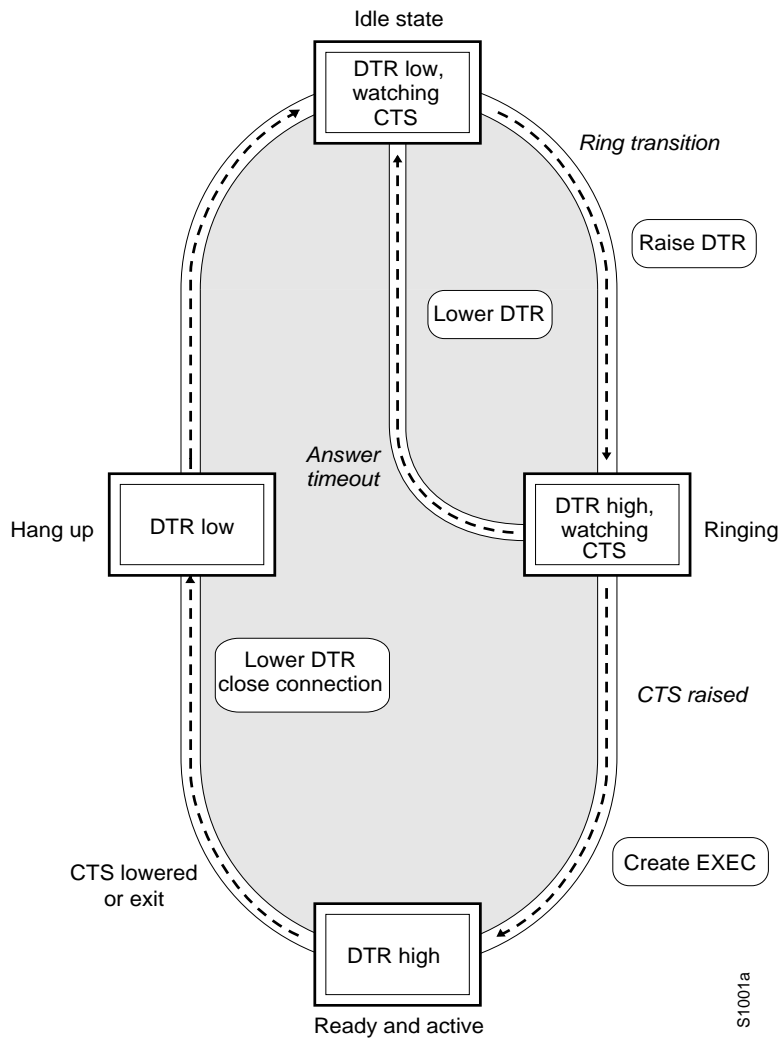
Support Old-Style Dial-In Modems

The Cisco IOS software supports dial-in modems that use DTR to control the off-hook status of the telephone line. This feature is supported primarily on old-style modems, especially those in Europe. To configure the line to support this feature, perform the following task in line configuration mode:

Task	Command
Configure a line for a dial-in modem.	modem callin

Figure 13 illustrates the **modem callin** command. When a modem dialing line is idle, it has its DTR signal at a low state and waits for a transition to occur on the DSR (RING) input. This transition causes the line to raise the DTR signal and start watching the CTS signal from the modem. After the modem raises CTS, the Cisco IOS software creates an EXEC session on the line. If the timeout interval (set with the **modem answer-timeout** command) passes before the modem raises the CTS signal, the line lowers the DTR signal and returns to the idle state.

Figure 13 EXEC Creation on a Line Configured for Modem Call-in



Note The **modem callin** and **modem cts-required** line configuration commands are useful for SLIP operation. These commands ensure that when the line is hung up or the CTS signal drops, the line reverts from SLIP mode to normal interactive mode. These commands do not work if you put the line in network mode permanently.

Although you can use the **modem callin** line configuration command with newer modems, the **modem dialin** line configuration command described in this section is more appropriate. The **modem dialin** command frees up CTS input for hardware flow control. Modern modems do not require the assertion of DTR to answer a phone line (that is, to take the line off-hook).

Support Reverse Modem Connections and Prevent Incoming Calls

In addition to initiating connections, the Cisco IOS software can receive incoming connections. This capability allows you to attach serial and parallel printers, modems, and other shared peripherals to the router and drive them remotely from other modem-connected systems. The Cisco IOS software supports reverse TCP, XRemote, and LAT connections.

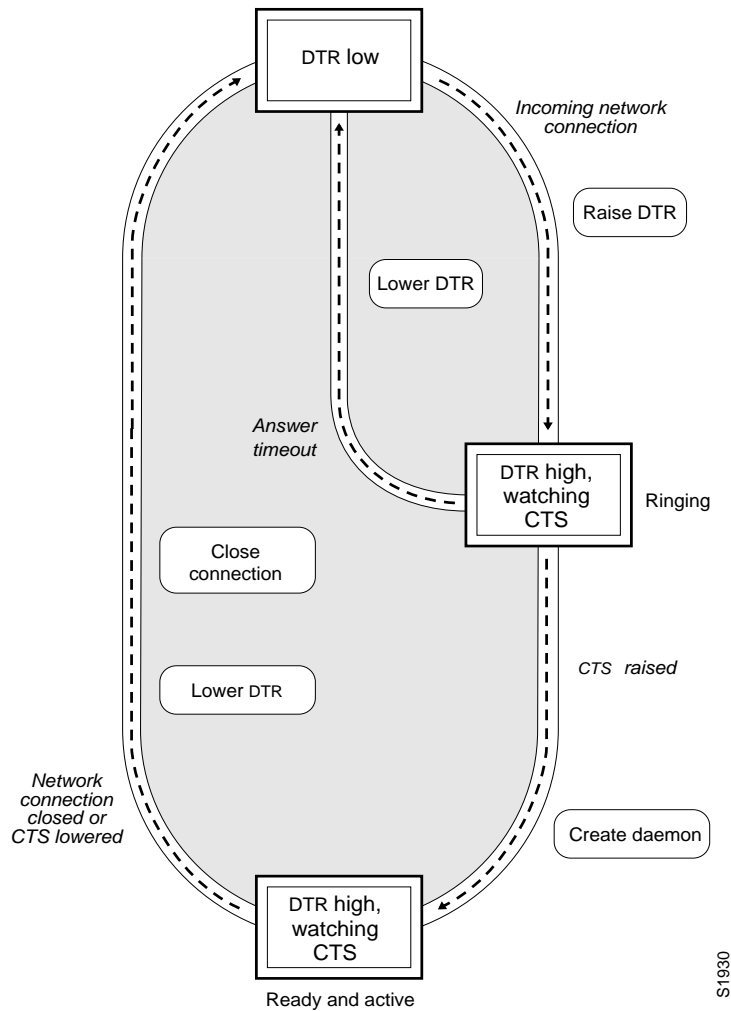
The specific TCP port, or socket, to which you attach the device determines the type of service that the Cisco IOS software provides on a line. When you attach the serial lines of a computer system or a data terminal switch to the serial lines of the router, the router can act as a network front-end device for a host that does not support the TCP/IP protocols. This arrangement is sometimes called *front-ending*, or *reverse connection mode*.

The Cisco IOS software supports ports connected to computers that are connected to modems. You can configure the Cisco IOS software to function somewhat like a modem by performing the following task in line configuration mode. This command also prevents incoming calls.

Task	Command
Configure a line for reverse connections and prevent incoming calls.	modem callout

Figure 14 illustrates the **modem callout** process. When the Cisco IOS software receives an incoming connection, it raises the DTR signal and waits to see if the CTS signal is raised to indicate that the host has noticed the router's DTR signal. If the host does not respond within the interval set by the **modem answer-timeout** line configuration command, the software lowers the DTR signal and drops the connection.

Figure 14 Daemon Creation on a Line Configured for Modem Call-out



Define Terminal Operation Characteristics

In line configuration mode, you can set terminal operation characteristics that will be in operation for that line until the next time you change the line parameters. Alternatively, you can change the line settings temporarily with the **terminal EXEC** commands described in the “Configuring Connections to Network Devices” chapter.

The most commonly used terminal operation characteristics are described in the following sections:

- Select the Transport Protocol for a Specific Line
- Change the Default Privilege Level for Lines
- Enable Password Checking at Login
- Configure Device Communication Parameters
- Define Escape Character and Other Key Sequences

- Configure Data Transparency
 - Set a Line to Act as a Pipe for File Transfers
 - Specify the International Character Display
- Establish Terminal Session Limits
- Disable Enhanced Editing Mode
- Record the Device Location
- Change the Retry Interval for a Terminal Port Queue

The following sections describe less commonly used terminal operating characteristics:

- Set Character Padding
- Set a Terminal-Locking Mechanism
- Specify the Terminal and Keyboard Type
- Configure Automatic Baud Rate Detection
- Set the Terminal Screen Length and Width
- Display Line Connection Information after the Login Prompt
- Save Local Settings between Sessions
- Set a Line as Insecure
- Set Pending Output Notification
- Create Character and Packet Dispatch Sequences

Select the Transport Protocol for a Specific Line

Use the **transport preferred** command to specify which transport protocol used on connections. Use the **transport input** and **transport output** commands to explicitly specify the protocols allowed on individual lines for both incoming and outgoing connections.



Caution Cisco routers do not accept incoming network connections to asynchronous ports (TTY lines) by default. You have to specify an incoming transport protocol, or specify **transport input all** before the line will accept incoming connections. For example, if you are using your router as a terminal server to make console-port connections to routers or other devices, you will not be able to use Telnet to connect to these devices. You will receive the message “Connection Refused.” This behavior is new as of Cisco IOS Software release 11.1. Previous to release 11.1, the default was **transport input all**. If you are upgrading to Cisco IOS software version 11.1(1) or later from Cisco IOS Software release 11.0 or earlier, you must add the **transport input** {*protocol* | **all**} command, or you will be locked out of your router.

The process of using Telnet to make a connection out of an asynchronous port is referred to as reverse Telnet. For more information, refer to the section “Configure a Reverse Connection” earlier in this chapter.

For routers that support LAT, the default protocol for outgoing connections is LAT. For those that do not support LAT, the default protocol for outgoing connections is Telnet. For incoming connections, no supported network protocols are accepted (the default protocol is **none**).

Perform one or more of the following tasks in line configuration mode to specify transport protocols:

Task	Command
Define which protocols can be used to connect to a specific line.	transport input {all lat mop nasi none pad rlogin telnet v120}
Determine the protocols that can be used for outgoing connections from a line.	transport output {all lat mop nasi none pad rlogin telnet v120}
Specify the protocol for the router to use if the user did not specify a protocol.	transport preferred {all lat mop nasi pad rlogin telnet v120}
Prevent errant connection attempts.	transport preferred none

The router accepts a host name entry at the EXEC system prompt as a Telnet command. If you enter the host name incorrectly, the router interprets the entry as an incorrect Telnet command and provides an error message indicating that the host does not exist. The **transport preferred none** command disables this option so that if you enter a command incorrectly at the EXEC prompt, the router does not attempt to make a Telnet connection to a host that it cannot find.

The **preferred transport** setting specifies a search order when attempting to resolve names that might be valid for multiple protocols. If the address or service does not match the preferred protocol, all other valid output protocols are searched to find a valid match.

To change the preferred transport protocol for a line during the current session, refer to the “Select a Preferred Connection Protocol for a Session” section in the “Making Connections to Network Devices” chapter of this publication.

Change the Default Privilege Level for Lines

To change the default privilege level for a given line or a group of lines, perform the following task in line configuration mode:

Task	Command
Specify a default privilege level for a line.	privilege level <i>level</i> ¹

1. This command is documented in the *Security Command Reference*.

Enable Password Checking at Login

You can enable password checking on a particular line so that the user is prompted to enter a password at the system login screen. You must then also specify a password. To do so, perform the following steps in line configuration mode:

Task	Command
Step 1 Enable password checking on a per-line basis using the password specified with the password command.	login
Step 2 Assign a password to a particular line.	password <i>password</i>

You can enable password checking on a per-user basis, in which case authentication is based on the username specified with the **username** global configuration command (see the “Managing the System” chapter in the *Security Configuration Guide*). To enable password checking on a per-user basis, perform one of the following tasks in line configuration mode:

Task	Command
Enable password checking on a per-user basis using the username and password specified with the username global configuration command.	login local
Select the TACACS-style user ID and password-checking mechanism.	login tacacs or login authentication { default list-name }

Use the **login tacacs** command with Terminal Access Controller Access Control System (TACACS) and Extended TACACS. Use the **login authentication** command with AAA/TACACS+.

By default, virtual terminals require passwords. If you do not set a password for a virtual terminal, the router displays an error message and closes the attempted connection. Use the **no login** command to disable this function and allow connections without a password.

For other access control tasks and password restrictions, including the **enable password** global configuration command that restricts access to privileged mode, see the *Security Configuration Guide*. For an example of enabling password checking, see the “Password Checking Examples” section at the end of this chapter.

Configure Device Communication Parameters

The Cisco IOS software supplies the following default serial communication parameters for terminal and other serial device operation:

- 9600 bits per second (bps) line speed
- 8 data bits
- 2 stop bits
- No parity bit

You can change these parameters as necessary to meet the requirements of the terminal or host to which you are connected. To do so, perform one or more of the following tasks in line configuration mode:

Task	Command
Set the line speed. Choose from line speed, transmit speed, or receive speed.	speed bps txspeed bps rxspeed bps
Set the data bits.	databits { 5 6 7 8 }
Set the stop bits.	stopbits { 1 1.5 2 }
Set the parity bit.	parity { none even odd space mark }

To change the transport protocols for a line during the current session, refer to the “Set Communication Parameters for the Current Session” section in the “Making Connections to Network Devices” chapter.

Define Escape Character and Other Key Sequences

You can define or modify the default key sequences to execute functions for system escape, terminal activation, disconnect, and terminal pause. To define or change the default sequence for any of these functions, perform one or more of the following tasks in line configuration mode:

Task	Command
Change the system escape sequence. The escape sequence indicates that the codes that follow have special meaning. The default escape sequence is Ctrl- ¹ .	escape-character <i>ascii-number</i>
Define a session activation sequence or character. Entering this sequence at a vacant terminal begins a terminal session. The default activation sequence is the Return key.	activation-character <i>ascii-number</i>
Define the session disconnect sequence or character. Entering this sequence at a terminal ends the session with the router. There is no default disconnect sequence.	disconnect-character <i>ascii-number</i>
Define the hold sequence or character that causes output to the terminal screen to pause. To continue the output, enter any character after the hold character. To use the hold character in normal communications, precede it with the escape character. There is no default sequence.	hold-character <i>ascii-number</i>

1. Pressing **Ctrl** displays a caret (^) character. The escape sequence is **Ctrl-Shift-6**.

You can reinstate the default value for the escape character or activation character by using the **no** form of the command. For example, issuing the **no escape-character** line configuration command returns the escape character to Ctrl-¹.

Note If you are using the **autoselect** function, the activation character should not be changed from the default value of Return. If you change this default, the **autoselect** feature may not function immediately.

To define escape characters for a line during the current session, refer to the “Define Special Characters for the Current Session” section in the “Making Connections to Network Devices” chapter of this publication.

Configure Data Transparency

Data transparency enables the Cisco IOS software to pass data on a terminal connection without the data being interpreted as a control character.

During terminal operations, some characters are reserved for special functions. For example, **Ctrl-Shift-6-X** (^X) suspends a session. When transferring files over a terminal connection (using the ZMODEM or Kermit protocols, for example), you must suspend the recognition of these special characters to allow a successful file transfer. This process is called *data transparency*.

Also, the classic U.S. ASCII character set is limited to 7 bits (128 characters) which adequately represents most displays in the U.S. Most defaults on the modem router work best on a 7-bit path. However, international character sets and special symbol display can require an 8-bit wide path and other handling. The “Specify the International Character Display” section later in this chapter describes how to reconfigure your router for international terminals.

Set a Line to Act as a Pipe for File Transfers

You can set a line to act as a transparent pipe so that programs such as Kermit, XMODEM, or CrossTalk can download a file across a terminal line. To temporarily configure a line to act as a pipe for file transfers, perform the following task in EXEC mode:

Task	Command
Set up the terminal line to act as a transparent pipe for file transfers.	terminal download

The **terminal download** command is equivalent to entering all the following commands, which are documented in the “Connection Commands” chapter of the *Access Services Command Reference* publication.

- **terminal telnet transparent**
- **terminal no escape-character**
- **terminal no hold-character**
- **terminal no pad 0**
- **terminal no pad 128**
- **terminal parity none**
- **terminal databits 8**

Specify the International Character Display

You can use a 7-bit character set (such as ASCII), or you can enable a full 8-bit international character set (such as ISO 8859). This allows special graphical and international characters for use in banners and prompts, and adds special characters such as software flow control. Character settings can be configured globally, by interface, and locally at the user level. Use the following criteria for determining which configuration mode to use when you set this international character display:

- If a large number of connected terminals support nondefault ASCII bit settings, use the global configuration commands.
- If only a few of the connected terminals support nondefault ASCII bit settings, use line configuration commands or the EXEC local terminal setting commands.

Note Setting the EXEC character width to an 8-bit character set can cause failures. If a user on a terminal that is sending parity enters the **help** command, an “unrecognized command” message appears because the system is reading all eight bits, although the eighth bit is not needed for **help**.

To specify a character set for all lines, perform one or both of the following tasks in global configuration mode:

Task	Command
Specify the character set used in EXEC and configuration command characters.	default-value exec-character-bits {7 8}
Specify the character set used in special characters such as software flow control, hold, escape, and disconnect characters.	default-value special-character-bits {7 8}

To specify a character set based on hardware, software, or on a per-line basis, perform the appropriate task in line configuration mode, as follows:

Task	Command
Set the number of data bits per character that are generated and interpreted by hardware.	databits {5 6 7 8}
Set the number of data bits per character that are generated and interpreted by software.	data-character-bits {7 8}
Specify the character set used in EXEC and configuration command characters on a per-line basis.	exec-character-bits {7 8}
Specify the character set used in special characters such as software flow control, hold, escape, and disconnect characters on per-line basis.	special-character-bits {7 8}

Note If you are using the **autoselect** function, the activation character should be set to the default Return, and the EXEC character bit should be set to 7. If you change these defaults, the application does not recognize the activation request.

To select an international character display for the current session, refer to the “Specify an International Character Display for the Current Session” section in the “Making Connections to Network Devices” chapter.

Establish Terminal Session Limits

You might need to control terminal sessions in high-traffic areas to provide resources for all users. You can define the following limitations for terminal sessions:

- The maximum number of sessions
- The idle session timeout interval or the absolute timeout interval

To establish terminal session limits, perform one of the following tasks in line configuration mode:

Task	Command
Set the maximum number of simultaneous sessions.	session-limit <i>session-number</i> ¹
Set the idle session timeout interval.	session-timeout <i>minutes</i> [output]
or	or
Set the absolute timeout interval.	absolute-timeout <i>minutes</i>
Warn users of impending timeouts set with the absolute-timeout command.	logout-warning [<i>seconds</i>]

1. There is no inherent upper limit to the number of sessions you can create.

Note The **absolute-timeout** command overrides any timeouts set through AppleTalk Remote Access (ARA).

Disable Enhanced Editing Mode

To disable enhanced editing mode and revert to the editing mode of previous software releases, perform the following task in line configuration mode:

Task	Command
Disable the enhanced editing features for a particular line.	no editing ¹

1. This command is documented in the “User Interface Commands” chapter of the *Configuration Fundamentals Command Reference*.

For example, you might disable enhanced editing if you have prebuilt scripts that conflict when enhanced editing is enabled. You can re-enable enhanced editing mode with the **editing** command.

Record the Device Location

You can record the location of a serial device. The text provided for the location appears in the output of the EXEC monitoring commands. To record the device location, perform the following task in line configuration mode:

Task	Command
Record the location of a serial device.	location <i>text</i>

Change the Retry Interval for a Terminal Port Queue

If you attempt to connect to a remote device (such as a printer) to the device that is busy, the connection attempt is placed in a terminal port queue. If the retry interval is set too high, and several routers or other devices are connected to the remote device, your connection attempt can have long delays. To change the retry interval for a terminal port queue, perform the following task in global configuration mode:

Task	Command
Change the retry interval for a terminal port queue.	terminal-queue entry-retry-interval <i>interval</i>

Set Character Padding

Character padding adds a number of null bytes to the end of the string and can be used to make a string an expected length for conformity. You can change the character padding on a specific output character. To set character padding, perform the following task in line configuration mode:

Task	Command
Set padding on a specific output character for the specified line.	padding <i>ascii-number count</i>

To set character padding for the current session, refer to the “Change Character Padding for the Current Session” section in the “Making Connections to Network Devices” chapter in this publication.

Set a Terminal-Locking Mechanism

You can enable a terminal-locking mechanism that allows a terminal to be temporarily locked by performing the following task in line configuration mode:

Task	Command
Enable a temporary terminal locking mechanism.	lockable

After you configure the line as lockable, you must still issue the **lock EXEC** command to lock the keyboard.

Specify the Terminal and Keyboard Type

You can specify the type of terminal connected to a line. This feature has two benefits: it provides a record of the type of terminal attached to a line, and it can be used in Telnet terminal negotiations to inform the remote host of the terminal type for display management. To specify the terminal type, perform the following task in line configuration mode:

Task	Command
Specify the terminal type.	terminal-type { <i>terminal-name</i> <i>terminal-type</i> }

This feature is used by TN3270 terminal to identify the keymap and ttycap passed by the Telnet protocol to the end host.

To specify the terminal or keyboard type for the current session, refer to the “Change the Terminal and Keyboard Type” section in the “Making Connections to Network Devices” chapter of this publication.

Configure Automatic Baud Rate Detection

You can configure a terminal to detect the baud rate being used over an asynchronous serial line automatically. To set up automatic baud detection, perform the following task in line configuration mode:

Task	Command
Set the terminal to automatically detect the baud rate.	autobaud

Note Do not use the **autobaud** command with the **autoselect** command.

To start communications using automatic baud detection, enter multiple Returns at the terminal. A 600-, 1800-, or 19200- baud line requires three Returns to detect the baud rate. A line at any other baud rate requires only two Returns. If you enter extra Returns after the baud rate is detected, the EXEC facility simply displays another system prompt.

Set the Terminal Screen Length and Width

By default, the Cisco IOS software provides a screen display of 24 lines by 80 characters. You can change these values if they do not meet the requirements of your terminal. The screen values you set are passed during rsh and rlogin sessions. To set the terminal screen length and width, perform the following tasks in line configuration mode:

Task	Command
Set the screen length.	length <i>screen-length</i>
Set the screen width.	width <i>characters</i>

The screen values set can be learned by some host systems that use this type of information in terminal negotiation. To disable pausing between screens of output, set the screen length to a zero.

The screen length specified can be learned by remote hosts. For example, the rlogin protocol uses the screen length to set up terminal parameters on a remote UNIX host. The width specified also can be learned by remote hosts.

To change the terminal screen length or width for the current session, refer to the “Change the Terminal Screen Length and Width” section in the “Making Connections to Network Devices” chapter of this publication.

Display Line Connection Information after the Login Prompt

You can display the host name, line number, and location of the host each time an EXEC session is started or an incoming connection is made. The line number banner appears immediately after the EXEC banner or incoming banner. This feature is useful for tracking problems with modems because it lists the host and line for the modem connection. Modem type information is also included if applicable.

To provide line information, perform the following task in global configuration mode:

Task	Command
Provide service line number information after the EXEC banner or incoming banner.	service linenum

Save Local Settings between Sessions

You can configure the Cisco IOS software to save local parameters set with **terminal EXEC** commands between sessions. Saving local settings ensures that the parameters the user sets will remain in effect between terminal sessions. This function is useful for servers in private offices. To save local settings between sessions, perform the following task in line configuration mode:

Task	Command
Save local settings between sessions.	private

By default, user-set terminal parameters are cleared when the session ends with either the **exit EXEC** command, or when the interval set with the **exec-timeout** line configuration command has passed.

Set a Line as Insecure

You can set up a terminal line to appear as an insecure dial-up line. The information is used by the LAT software, which reports such dial-up connections to remote systems.

To set a line as insecure, perform the following tasks in line configuration mode:

Task	Command
Set the line as a dial-up line.	insecure

In the previous releases of Cisco IOS software, any line that used modem control was reported as dial-up connection through the LAT protocol; this feature allows more direct control of your line.

Set Pending Output Notification

You can set up a line to inform a user who has multiple, concurrent Telnet connections when output is pending on a connection other than the active one. For example, you might want to know when another connection receives mail or a message. To set pending output notification, perform the following task in line configuration mode:

Task	Command
Set up a line to notify a user of pending output.	notify

To change pending output notification for the current session, refer to the “Change Packet Output Notification” section in the “Making Connections to Network Devices” chapter of this publication.

Create Character and Packet Dispatch Sequences

The Cisco IOS software supports dispatch sequences and TCP state machines that transmit data packets only when they receive a defined character or sequence of characters. You can set up dispatch characters that allow packets to be buffered, then transmitted upon receipt of a character. You can set up a state machine that allows packets to be buffered, then transmitted upon receipt of a sequence of characters. This feature enables packet transmission when the user presses a function key, which is typically defined as a sequence of characters, such as “Esc I C.”

TCP state machines can control TCP processes with a set of predefined character sequences. The current state of the device determines what happens next, given an expected character sequence. The state-machine commands configure the server to search for and recognize a particular sequence of characters, then cycle through a set of states. The user defines these states—up to eight states can be defined. (Think of each state as a task that the server performs based on the assigned configuration commands and the type of character sequences received.)

The Cisco IOS software supports user-specified state machines for determining whether data from an asynchronous port should be sent to the network. This functionality extends the concept of the dispatch character and allows the equivalent of multicharacter dispatch strings.

Up to eight states can be set up for the state machine. Data packets are buffered until the appropriate character or sequence triggers the transmission. Delay and timer metrics allow for more efficient use of system resources. Characters defined in the TCP state machine take precedence over those defined for a dispatch character.

Perform the following tasks in line configuration mode, as needed, for your particular system needs:

Task	Command
Specify the transition criteria for the states in a TCP state machine.	state-machine <i>name state firstchar lastchar [nextstate transmit]</i>
Specify the state machine for TCP packet dispatch.	dispatch-machine <i>name</i>

Task	Command
Define a character that triggers packet transmission.	dispatch-character <i>ASCII-number</i> [<i>ASCII-number2</i> . . . <i>ASCII-number</i>]
Set the dispatch timer.	dispatch-timeout <i>milliseconds</i>

To change character and packet dispatch sequence for the current session, refer to the “Change the Packet Dispatch Character for the Current Session” section in the “Making Connections to Network Devices” chapter of this publication.

Define a Command String for Automatic Execution

You can set up a command to execute automatically when the router connects to another host. The Cisco IOS can execute any appropriate EXEC command and any switch or host name that occurs with the EXEC command. To define a command, perform the following task in line configuration mode:

Task	Command
Define a command to be automatically executed.	autocommand <i>command</i>

Connections to an Individual Line

Connections to an individual line are most useful when a dial-out modem, parallel printer, or serial printer is attached to that line. To connect to an individual line, the remote host or terminal must specify a particular TCP port on the router.

If reverse XRemote is required, that port is 9000 (decimal) plus the decimal value of the line number.

If a raw TCP stream is required, the port is 4000 (decimal) plus the decimal line number. The raw TCP stream is usually the required mode for sending data to a printer.

If Telnet protocols are required, that port is 2000 (decimal) plus the decimal value of the line number. The Telnet protocol might require that Return characters be translated into Return and line-feed character pairs. You can turn off this translation by specifying the Telnet binary mode option. To specify this option, connect to port 6000 (decimal) plus the decimal line number.

For example, a laser printer is attached to line 10 of a Cisco 2511 router. Such a printer usually uses XON/XOFF software flow control. Because the Cisco IOS software cannot receive an incoming connection if the line already has a process, you must ensure that an EXEC session is not accidentally started. You must, therefore, configure it as follows:

```
line 10
flowcontrol software
no exec
transport input all
```

A host that wants to send data to the printer would connect to the router on TCP port 4010, send the data, and then close the connection.

Configure Rotary Groups

Connections can be made to the next free line in a group of lines, also called a *rotary group* or *hunt group*. A line can be in only one rotary group; a rotary group can consist of a single line or several contiguous lines. The console line (line 0) cannot be in a rotary group.

To configure a rotary group, perform the following task in line configuration mode:

Task	Command
Add a line to the specified rotary group.	rotary <i>group</i>

LPD Protocol Support

The Cisco IOS software supports a subset of the Berkeley UNIX Line Printer Daemon (LPD) protocol used to send print jobs between UNIX systems. This subset of the LPD protocol permits:

- Improved status information
- Cancellation of print jobs
- Confirmation of successful printing and automatic retry for common print failures
- Use of standard UNIX software

The Cisco implementation of LPD permits you to configure a printer to allow several types of data to be sent as print jobs (for example, PostScript or raw text).

To configure a printer for the LPD protocol, perform the following task in global configuration mode:

Task	Command
Configure printer and specify a TTY line (or lines) for the device.	printer <i>printername</i> { line number rotary number } [newline-convert]

If you use the **printer** command, you also must modify the */etc/printcap* file on the UNIX system to include the definition of the remote printer on the router. Use the optional **newline-convert** keyword on UNIX systems that do not handle single character line terminators to convert a new line to a character Return, line-feed sequence.

The following example includes the configuration of the printer Saturn on the host Memphis:

```
comm1pt|Printer on cisco AccessServer:\
:rm=memphis:rp+saturday:\
:sd+/usr/spool/lpd/comm1pt:\
:lf=?var/log/lpd/comm1pt:
```

The content of the actual file may differ, depending on the configuration of your UNIX system.

Configure Chat Scripts for Asynchronous Lines

Chat scripts are strings of text used to send commands for modem dialing, logging onto remote systems, and initializing asynchronous devices connected to an asynchronous line. On a router, chat scripts can be configured on the auxiliary port only. A chat script must be configured to dial out on asynchronous lines. You also can configure chat scripts so that they are executed automatically for other specific events on a line, or so that they are executed manually. Each chat script is defined for a different event. These events can include the following:

- Line activation
- Incoming connection initiation
- Asynchronous dial-on-demand routing (see the “Configuring DDR” chapter in the *Wide-Area Networking Configuration Guide*)

- Line resets
- Startup

To use a chat script, perform the following steps:

Step 1 Define the chat script in global configuration mode using the **chat-script** command.

Step 2 Configure the line so that a chat script is activated when a specific event occurs (using the **script** line configuration command), or start a chat script manually (using the **start-chat** privileged EXEC command).

Create a Chat Script

To define a chat script, perform the following task in global configuration mode:

Task	Command
Create a script that will place a call on a modem, log on to a remote system, or initialize an asynchronous device on a line.	chat-script <i>script-name expect send...</i> ¹

1. This command is described in the “DDR Commands” chapter in the *Wide-Area Networking Command Reference*.

A limited list of keywords are supported, along with expect/send pairs. Send strings can have special escape modifiers. For an example of how to write scripts, refer to section “DDR Configuration Examples” in the “Configuring DDR” chapter in the *Wide-Area Networking Configuration Guide*.

Cisco recommends that one chat script (a “modem” chat script) be written for placing a call and another chat script (a “system” or “login” chat script) be written to log onto remote systems, where required.

Chat scripts are not supported on lines where modem control is set for inbound activity that only uses the **modem dialin** command.

Suggested Chat Script Naming Conventions (for Dial Scripts only)

When you create a script name, include the modem vendor, type, and modulation, separated by hyphens. For example, if you have a Telebit t3000 modem that uses V.32bis modulation, your script name would be *telebit-t3000-v32bis*.

A suggested naming convention for chat scripts used to dial is as follows:

vendor-type-modulation

In other words, the syntax of the **chat-script** command becomes the following:

chat-script *vendor-type-modulation expect send...*

For example, if you have a Telebit t3000 modem that uses V.32bis modulation, you would name your chat script as follows:

telebit-t3000-v32bis

The chat-script command could become the following:

```
Router(config)# chat-script telebit-t3000-v32bis ABORT ERROR ABORT BUSY ABORT
"NO ANSWER" "" "ATH" OK "ATDT\T" TIMEOUT 30 CONNECT
```

Adhering to this naming convention allows you to specify a range of chat scripts using partial chat script names with regular expressions. This is particularly useful for dialer rotary groups and is explained further in the “Configure an Interface to Receive Calls” section in the “Configuring DDR” chapter in the *Wide-Area Networking Configuration Guide*.

Configure the Line to Activate Chat Scripts

Chat scripts can be activated by any of five events, each corresponding to a different version of the **script** line configuration command. To start a chat script manually at any point, refer to the following section, “Start a Chat Script Manually on an Asynchronous Line.”

To define a chat script to start automatically when a specific event occurs, perform the following tasks in line configuration mode:

Task	Command
Start a chat script on a line when the line is activated (every time a command EXEC is started on the line).	script activation <i>regexp</i> ¹
Start a chat script on a line when a network connection is made to the line.	script connection <i>regexp</i>
Specify a modem script for DDR on a line.	script dialer ² <i>regexp</i>
Start a chat script on a line whenever the line is reset.	script reset <i>regexp</i>
Start a chat script on a line whenever the system is started up.	script startup <i>regexp</i>

1. The argument *regexp* is a regular expression that is matched to a script name that has already been defined using the **chat-script** command.
2. To use a chat script for dial-on-demand routing (DDR), refer to “Create Chat Scripts for Asynchronous Interfaces” in the “Configuring DDR” chapter in the *Wide-Area Networking Configuration Guide*.

Note Outbound chat scripts are not supported on lines where modem control is set for inbound activity only (using the **modem dialin** command).

Start a Chat Script Manually on an Asynchronous Line

You can start a chat script manually on any line that is currently not active by performing the following task in privileged EXEC mode:

Task	Command
Start a chat script manually on any asynchronous line.	start-chat <i>regexp</i> [<i>line-number</i> [<i>dialer-string</i>]]

If you do not specify the line number, the script runs on the current line. If the line specified is already in use, you cannot start the chat script. A message appears indicating that the line is already in use.

Call Back Asynchronous Clients

You can configure the Cisco IOS software to call back an asynchronous device that dials in and requests a callback from the router, then disconnects. Asynchronous callback is supported for the following protocols:

- Any device calling in and connecting to the router at the EXEC level
- AppleTalk Remote Access (ARA)
- Point-to-Point Protocol (PPP)

Callback is also supported on other interface types for PPP, including ISDN.

All callback sessions are returned on physical terminal (TTY) lines. ARA is supported on VTY lines, but also is supported on TTY lines if the **vty-arap** command is used. PPP, however, is supported on interfaces. Therefore, to enable PPP callback, you must issue the **autoselect ppp** command on the callback lines.

All current security mechanisms supported in the Cisco IOS software are supported by the callback facility, including the following:

- TACACS+
- CHAP and PAP for PPP
- Per-user authentication for EXEC callback and ARA callback

The call originator must have the appropriate permissions set on the router before it can initiate a callback session.

Callback is useful for two purposes:

- Cost savings on toll calls

For example, suppose it costs more to call from clients in Zone A to devices in Zone D than to call from Zone D to Zone A—costs are lower when devices in Zone D call back clients in Zone A.
- Consolidation and centralization of phone billing

For example, if a corporation has 64 dial-in clients, enabling the corporation's routers to call back these clients consolidates billing. Instead of 64 phone bills, the corporation receives one bill.

Call Back Clients Dialing In and Connecting to the EXEC Prompt

You can call back clients that dial in to a TTY line and connect to the EXEC prompt. To enable callback, perform the following tasks, starting in global configuration mode:

Task	Command
Step 1 Enable EXEC callback.	service exec-callback
Step 2 Define a chat script to be applied when clients dial in to the EXEC prompt.	chat-script <i>script-name expect-send</i> ¹
Step 3 Specify a per-username callback dial string.	username ² <i>name</i> [callback-dialstring <i>telephone-number</i>]
Step 4 Specify a per-username rotary group for callback.	username ² <i>name</i> [callback-rotary <i>rotary-group-number</i>]
Step 5 Specify a per-username line or set of lines for callback.	username ² <i>name</i> [callback-line [aux tty] <i>line-number</i> [<i>ending-line-number</i>]]

Task	Command
Step 6 Do not require authentication on EXEC callback.	username ² <i>name</i> [nocallback-verify]
Step 7 Enter line configuration mode.	line [tty] <i>line-number</i> [<i>ending-line-number</i>]
Step 8 Apply a chat script to the line or a set of lines.	script callback <i>regexp</i>
Step 9 Delay the callback for client modems that require a rest period before receiving a callback.	callback forced-wait <i>number-of-seconds</i>

1. This command is described in the “DDR Commands” chapter in the *Wide-Area Networking Command Reference*.

2. This command is described in the *Security Command Reference*.

The recommended EXEC chat script is as follows:

```
chat-script name ABORT ERROR ABORT BUSY "" "ATZ" OK "ATDT \T" TIMEOUT 30 CONNECT \c
```

For an example of calling back clients connecting to the EXEC facility, see the “Call Back Clients Connecting to the EXEC Prompt Example” section at the end of this chapter.

Call Back ARA Clients

You can call back ARA clients. Perform the following steps, starting in global configuration mode. These steps assume you have already enabled AppleTalk routing and enabled ARA.

Task	Command
Step 1 Enable callback to an ARA client.	arap callback
Step 2 Define a chat script to be applied when an ARA client connects to a TTY line and requests callback.	chat-script <i>script-name expect-send</i> ¹
Step 3 Enter line configuration mode.	line [tty] <i>line-number</i> [<i>ending-line-number</i>]
Step 4 Enable ARA on the line.	arap enable ²
Step 5 Configure automatic protocol startup on the line.	autoselect arap ²
Step 6 Enable authentication on the line.	login { authentication local } ³
Step 7 Apply an ARA-specific chat script to a line or set of lines.	script arap-callback <i>regexp</i>
Step 8 Delay the callback for client modems that require a rest period before receiving a callback.	callback forced-wait <i>number-of-seconds</i>
Step 9 Exit to global configuration mode.	exit
Step 10 Specify a per-username callback dial string.	username ³ <i>name</i> [callback-dialstring <i>telephone-number</i>]
Step 11 Specify a per-username rotary group for callback.	username ³ <i>name</i> [callback-rotary <i>rotary-group-number</i>]
Step 12 Specify a per-username line or set of lines for callback.	username ³ <i>name</i> [callback-line [tty] <i>line-number</i> [<i>ending-line-number</i>]]

1. This command is described in the “DDR Commands” chapter in the *Wide-Area Networking Command Reference*.

2. This command is documented in the “AppleTalk Remote Access Commands” chapter in the *Access Services Command Reference*.

3. This command is described in the *Security Command Reference*.

The recommended ARA chat script follows. The parts of the string that are **bolded** are vendor-specific extensions on the Telebit 3000 modem to disable error control. Refer to the manual for your modem for the specific commands to disable error correction for ARA.

```
chat-script name ABORT ERROR ABORT BUSY "" "ATZ" OK "ATS180=0" OK "ATS181=1" OK "ATDT \T" TIMEOUT 60 CONNECT \c
```

For an example of calling back a PPP client, see the “Call Back a PPP Client Example” section at the end of this chapter.

Call Back PPP Clients

You can call back PPP clients that dial in to asynchronous interfaces. You can enable callback to the following two types of PPP clients:

- Clients that implement PPP callback per RFC 1570 (as an LCP negotiated extension).
- Clients that do not negotiate callback but can put themselves in answer-mode, whereby a callback from the router is accepted.

This section describes how to enable callback to each of these types of PPP clients.

Accept Callback Requests from RFC-Compliant PPP Clients

To accept a callback request from a RFC 1370-PPP compliant client, perform the following task, in interface (asynchronous) configuration mode:

Task	Command
Step 1 Enable callback requests from RFC1570-compliant PPP clients on an asynchronous interface.	ppp callback accept ¹

1. This command is described in the “SLIP and PPP Commands” chapter in the *Access Services Command Reference*.

To configure the Cisco IOS software to call back the originating PPP client, refer to the section “Enable PPP Callback on Outgoing Lines” later in this chapter.

Accept Callback Requests from Non-RFC-Compliant PPP Clients Placing Themselves in Answer Mode

A PPP client can put itself in answer-mode and can still be called back by the router, even though it cannot specifically request callback. To enable callback on the router to this type of client, perform the following task in interface (asynchronous) configuration mode:

Task	Command
Initiate callback requests from non-RFC 1570-compliant PPP clients on an asynchronous interface.	ppp callback initiate ¹

1. This command is described in the “SLIP and PPP Commands” chapter in the *Access Services Command Reference*. It is supported on access server product models only. The **ppp callback request** command enables routers to place callback requests to peer routers. Refer to the “DDR Commands” chapter in the *Wide-Area Networking Command Reference*.

To configure the Cisco IOS software to call back the originating PPP client, refer to the next section, “Enable PPP Callback on Outgoing Lines.”

Enable PPP Callback on Outgoing Lines

After enabling PPP clients to connect to an asynchronous interface and wait for a callback, you must place one or more TTY lines in PPP mode. Although calls from PPP clients enter through an asynchronous interface, the calls exit the client on a line placed in PPP mode.

To enable PPP client callback on outgoing TTY lines, perform the following steps, beginning in global configuration mode:

Task	Command
Step 1 Define a chat script to be applied when a PPP client requests callback.	chat-script ¹ <i>script-name expect-send</i>
Step 2 Specify a per-username callback dial string.	username ² <i>name [callback-dialstring telephone-number]</i>
Step 3 Specify a per-username rotary group for callback.	username ³ <i>name [callback-rotary rotary-group-number]</i>
Step 4 Specify a per-username line or set of lines for callback.	username ³ <i>name [callback-line [tty] line-number [ending-line-number]]</i>
Step 5 Enter line configuration mode.	line [tty] line-number [ending-line-number]
Step 6 Configure automatic PPP startup on a line or set of lines.	autoselect ppp ³
Step 7 Enable authentication on the line.	login { authentication local } ³
Step 8 Apply a chat script to a line or set of lines.	script callback regexp
Step 9 Delay the callback for client modems that require a rest period before receiving a callback.	callback forced-wait number-of-seconds

1. This command is described in the “DDR Commands” chapter in the *Wide-Area Networking Command Reference* publication.

2. This command is described in the *Security Command Reference* publication.

3. This command is documented in the “SLIP and PPP Commands” chapter in the *Access Services Command Reference* publication.

A client can issue a callback dial string; that dial string is used *only* if the dial string on the router is specified as NULL, or is not defined.

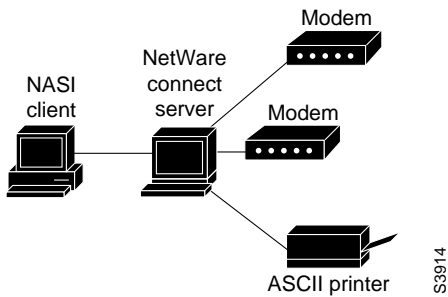
The recommended PPP chat script follows.

```
chat-script name ABORT ERROR ABORT BUSY "" "ATZ" OK "ATDT \T" TIMEOUT 30 CONNECT \c
```

Enable NASI Clients to Access Network Resources

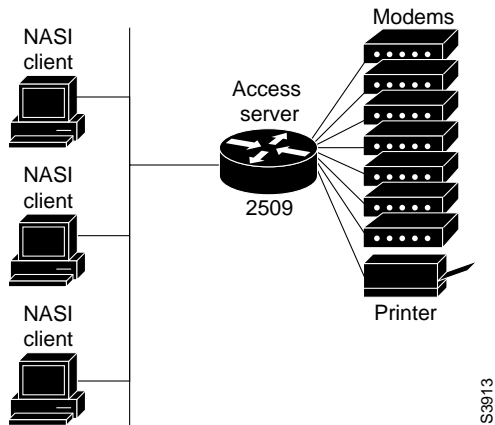
You can enable your router to function as a NetWare Asynchronous Services Interface (NASI) server. A NASI server enables a NASI client to connect to asynchronous network resources (such as modems) without having these resources located on the client’s desktop, as shown in Figure 15.

Figure 15 NASI Setup in a NetWare Environment



You can configure the Cisco IOS software to enable NASI clients to connect to asynchronous resources attached to your router. The NASI client can connect to any port on the router other than the console port to access network resources. (See Figure 16.) The NASI clients are connected to the Ethernet 0 interface on the router. When the user on the NASI client uses the Windows or DOS application to connect to the router, a list of available TTY and VTY lines appears, beginning with TTY1. The user selects the desired outgoing TTY or VTY port. You also can configure TACACS+ security on the router so that after the user selects a TTY or VTY port, a username and password prompt appear for authentication, authorization, and accounting purposes.

Figure 16 NASI Clients Accessing Asynchronous Resources through an Access Server



Note The Cisco IOS implementation of NASI functions best with NASI client software version 2.0 and later.

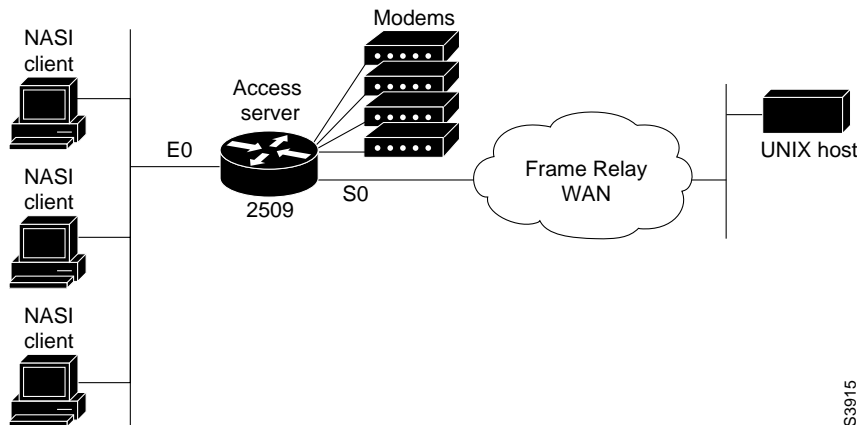
The NASI client can be on a local LAN or can also be on a remote LAN. If it is on a remote LAN, the following two requirements must be met:

- A router routing IPX forwards NetWare Connect Server SAP advertisements from the remote LAN to the LAN to which the local router is connected.

- The same router routing IPX spoofs Get Nearest Server (GNS) replies for the GNS requests that the client sends out.

The fact that you can connect to many different ports on the router means that you can provide access to more than one asynchronous device. When the user accesses the VTY line, the connection to the user EXEC facility and can issue a Telnet or NASI command to access a remote network (see Figure 17). Only the first available VTY line appears in the list of available ports on the router (and it is titled *RCONSOLE*).

Figure 17 NASI Clients Gaining Access to IP Hosts on a Remote Network



To configure your router as a NASI server, perform the following tasks, beginning in global configuration mode:

Task	Command
Step 1 Enable IPX routing on the router.	ipx routing ¹
Step 2 Define an internal IPX network number.	ipx internal-network ¹
Step 3 Enter interface configuration mode.	interface <i>type number</i> ²
Step 4 Enable IPX routing on an interface.	ipx network [<i>network</i> unnumbered] ¹
Step 5 Exit to global configuration mode.	exit
Step 6 Enable NASI.	ipx nasi-server enable
Step 7 Configure TACACS+ security on all lines on the router (optional).	aaa authentication nasi { <i>list-name</i> default } { methods list } ³
Step 8 Enter line configuration mode.	line [aux tty vty] <i>line-number</i> [<i>ending-line-number</i>]
Step 9 Configure TACACS+ security on a per-line basis (optional).	login authentication nasi { <i>list-name</i> default } ³

1. This command is documented in the *Network Protocols Command Reference, Part 2*.

2. This command is documented in the *Configuration Fundamentals Command Reference*.

3. This command is documented in the *Security Command Reference*.

You also can configure SAP filters to filter SAP updates, and access lists to filter NASI traffic between interfaces on the router.

Note If a NASI server is already on the LAN segment connected to the router, the router cannot respond to Get Next Server (GNS) requests for NASI services.

Display Terminal Banner Messages

The types of messages that can be displayed to terminal users who connect to the router are described in the following sections:

- Configure a Message-of-the-Day (MOTD) Banner
- Configure a Login Banner
- Configure a Line-Activation Banner
- Configure an Incoming Banner
- Configure an Idle Terminal Message
- Display a “Line in Use” Message
- Display a “Host Failed” Message
- Enable or Disable the Display of Banners

You also can turn off message displays.

For an example of displaying terminal banner messages, see the “Banner Example” section at the end of this chapter.

Configure a Message-of-the-Day (MOTD) Banner

You can configure a message-of-the-day (MOTD) banner to be displayed on all connected terminals. This banner is displayed at login and is useful for sending messages that affect all network users (such as impending system shutdowns). To do so, perform the following task in global configuration mode:

Task	Command
Configure a MOTD banner.	banner motd <i>d message d</i>

Configure a Login Banner

You can configure a login banner to be displayed on all connected terminals. This banner is displayed after the MOTD banner and before the login prompts.

To configure a login banner, perform the following task in global configuration mode:

Task	Command
Configure a login banner.	banner login <i>d message d</i>

The login banner cannot be disabled on a per-line basis. To globally disable the login banner, you must delete the login banner with the **no banner login** command.

Configure a Line-Activation Banner

You can configure a line-activation banner to be displayed when an EXEC process (such as a line-activation or incoming connection to a VTY line) is created. To do so, perform the following task in global configuration mode:

Task	Command
Configure a banner to be displayed on terminals with an interactive EXEC session.	banner exec <i>d message d</i>

Configure an Incoming Banner

You can configure a banner to be displayed on terminals connected to reverse Telnet lines. This banner is useful for providing instructions to users of these types of connections.

To configure a banner that is sent on incoming connections, perform the following task in global configuration mode:

Task	Command
Configure a banner to display on terminals connected to reverse Telnet lines.	banner incoming <i>d message d</i>

Configure an Idle Terminal Message

You can configure messages to be displayed on a console or terminal not in use. Also called a *vacant message*, this message is different from the banner message displayed when an EXEC process is activated. To configure an idle terminal message, perform the following task in line configuration mode:

Task	Command
Display an idle terminal message.	vacant-message [<i>d message d</i>]

Display a “Line in Use” Message

You can display a “line in use” message when an incoming connection is attempted and all rotary group or other lines are in use. Perform the following task in line configuration mode:

Task	Command
Display a “line in use” message.	refuse-message <i>d message d</i>

If you do not define such a message, the user receives a system-generated error message when all lines are in use. You also can use this message to provide the user with further instructions.

Display a “Host Failed” Message

You can display a “host failed” message when a Telnet connection with a specific host fails. Perform the following task in line configuration mode:

Task	Command
Display a “host failed” message.	busy-message <i>hostname d message d</i>

Enable or Disable the Display of Banners

You can control display of the message-of-the-day (MOTD) and line-activation (EXEC) banners. By default, these banners are displayed on all lines. To suppress or reinstate the display of such banners, perform one of the following tasks in line configuration mode:

Task	Command
Suppress MOTD and EXEC banner display.	no exec-banner
Reinstate the display of the EXEC or MOTD banners.	exec-banner
Suppress MOTD banner display only.	no motd-banner
Reinstate the display of the MOTD banners.	motd-banner

These commands determine whether the router will display the EXEC banner and the message-of-the-day (MOTD) banner when an EXEC session is created. These banners are defined with the **banner motd** and **banner exec** commands. By default, the MOTD banner and the EXEC banner are enabled on all lines.

Disable the EXEC and MOTD banners using the **no exec-banner** command.

The MOTD banners can also be disabled by the **no motd-banner** line configuration command, which disables MOTD banners on a line. If the **no exec-banner** command is configured on a line, the MOTD banner will be disabled regardless of whether the **motd-banner** command is enabled or disabled. Table 4 summarizes the effects of the **exec-banner** command and the **motd-banner** command.

Table 4 Banners Displayed

	exec-banner (default)	no exec-banner
motd-banner (default)	MOTD banner EXEC banner	None
no motd-banner	EXEC banner	None

For reverse Telnet connections, the EXEC banner is never displayed. Instead, the incoming banner is displayed. The MOTD banner is displayed by default, but it is disabled if either the **no exec-banner** command or **no motd-banner** command is configured. Table 5 summarizes the effects of the **exec-banner** command and the **motd-banner** command for reverse Telnet connections.

Table 5 Banners Displayed—Reverse Telnet Session to Async Lines

	exec-banner (default)	no exec-banner
motd-banner (default)	MOTD banner incoming banner	incoming banner
no motd-banner	incoming banner	incoming banner

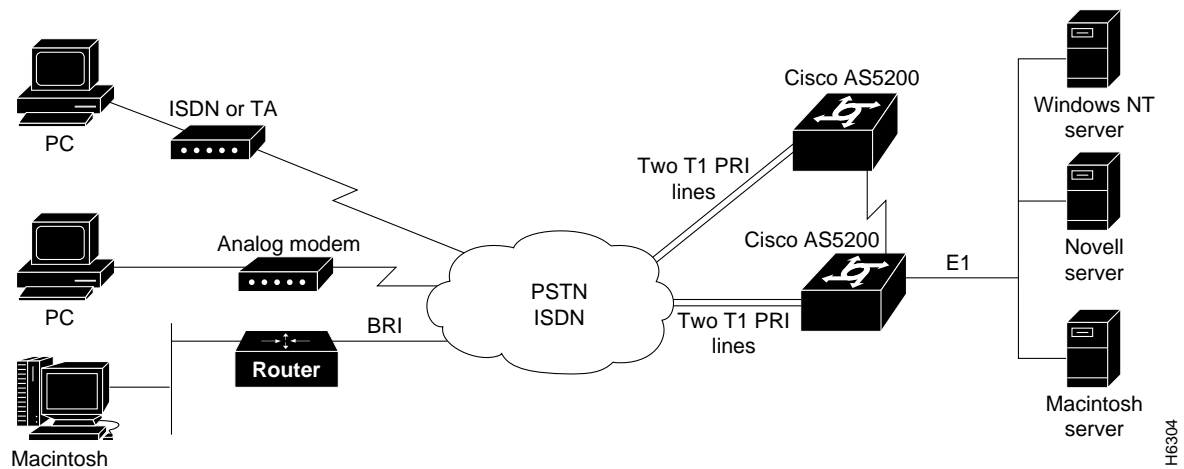
Enable Modem Services Specific to the Cisco AS5200

The Cisco AS5200 access server supports the needs of the following users:

- Small office or home users who use analog modems to dial in from disparate locations. These incoming analog calls enter the T1 Primary Rate Interface (PRI), connect to an integrated Cisco AS5200 modem, and access network resources.
- Telecommuters and central offices who are increasingly using ISDN Basic Rate Interface (BRI), ISDN Primary Rate Interface (PRI), and ISDN digital modems to access enterprise networks. These incoming digital data connections and calls enter the T1 PRI interface and directly connect to network resources.

Figure 18 shows how the Cisco AS5200 services these types of users.

Figure 18 Cisco AS5200 Access Server Connections



The following sections describe some of the asynchronous services you can perform with the Cisco AS5200 access server:

- Monitor Modems and Enable Modem Events
- Troubleshoot and Manage Modems
- Send AT Commands to Manageable Modems

- Poll Manageable Modems
- Download Modem Firmware

The Cisco AS5200 access server contains integrated V.34 modems that are *manageable* (also known as *select*) or *nonmanageable* (also known as *reliable*). Each manageable modem has one out-of-band port, which is used for polling modem statistics and creating a directly connected session for transmitting attention (AT) commands. Nonmanageable modems do not have out-of-band ports. As you read through this documentation, note which commands apply to manageable versus nonmanageable modems.

Incoming ISDN and analog calls access the Cisco AS5200 through dual T1 PRIs. Unlike digital calls, incoming analog calls are first terminated and then converted to digital data at the modem card.

Each TTY line is directly mapped to an integrated Cisco AS5200 modem, as shown in Table 6. The TTY lines 1 through 24 directly connect to modems 1/0 through 1/23, which are installed in the first chassis slot. The TTY lines 25 through 48 directly connect to modems 2/0 through 2/23, which are installed in the second chassis slot.

Table 6 TTY Lines Associated with Integrated Cisco AS5200 Modems

TTY Line	Slot/Modem Port Number	TTY Line	Slot/Modem Port Number
1	1/0	25	2/0
2	1/1	26	2/1
3	1/2	27	2/2
4	1/3	28	2/3
5	1/4	29	2/4
6	1/5	30	2/5
7	1/6	31	2/6
8	1/7	32	2/7
9	1/8	33	2/8
10	1/9	34	2/9
11	1/10	35	2/10
12	1/11	36	2/11
13	1/12	37	2/12
14	1/13	38	2/13
15	1/14	39	2/14
16	1/15	40	2/15
17	1/16	41	2/16
18	1/17	42	2/17
19	1/18	43	2/18
20	1/19	44	2/19
21	1/20	45	2/20
22	1/21	46	2/21
23	1/22	47	2/22
24	1/23	48	2/23

Note Table 6 reflects the modem mapping for a Cisco AS5200 access server using 48 integrated modems. If the first chassis slot in the access server is not occupied by a modem card, the TTY lines 1 to 24 connect to modems 2/0 through 2/23.

See the “Cisco AS5200 Startup Sample Configuration Example” section at the end of this chapter.

Monitor Modems and Enable Modem Events

You can view various modem statistics and configure modem events using the Cisco IOS software with the Cisco AS5200 access server.

Show Modem Performance

To show modem performance statistics, perform the following tasks in EXEC mode:

Task	Command
Show various performance statistics for a modem or group of modems.	show modem [<i>slot/modem-port</i> group number]
Show the call-switching module status for a modem or group of modems.	show modem csm [<i>slot/modem-port</i> group number]
Show the event log status for a modem or group of modems.	show modem log [<i>slot/modem-port</i> group number] ¹
Display the cumulative system statistics for all modems installed in the access server.	show modem summary ¹
Display all directly connected AT sessions active on the access server.	show modem at-mode ¹

1. This command does not apply to nonmanageable modems.

Enable Incoming and Outgoing Analog Calls

To enable the Cisco AS5200 modems to accept and send incoming and outgoing analog calls through the T1 PRIs, you must configure the following interface configuration and controller configuration command by performing the following tasks in interface configuration mode:

Task	Command
Enable ISDN voice calls to dial into and dial out of the Cisco AS5200.	isdn incoming-voice modem ¹
Enable incoming and outgoing channelized T1 voice calls to dial into and dial out of the Cisco AS5200.	cas-group <i>channel-number</i> [timeslots range] ²

1. Without configuring this command, the modem calls cannot be routed to the internal modems.

2. The default value for this command configures 24 time slots with the channel associated signal called E&M (Ear and Mouth), which is the default signal type. Switched 56 digital calls are not supported under this new feature.

See the “Cisco AS5200 Configuration Examples” section at the end of this chapter for the “ISDN Analog Calls Example.”

Set Modem Recovery Time

To set the maximum amount of time the call-switching module waits for a local modem to respond to a request before it is considered locked in a suspended state, perform the following task in global configuration mode:

Task	Command
Set maximum time local modems will wait for a response. The default is 5 minutes.	modem recovery-time <i>minutes</i>

This command does not apply to nonmanageable modems.

After the call-switching module resets a suspended modem, the module recovers to a default call-switching module state.

Set Modem Event Buffer

To configure the size of the history event queue buffer for manageable modems in the access server, perform the following task in global configuration mode:

Task	Command
Define the number of modem events that each modem is able to store. The default is 100 events for each modem.	modem buffer-size <i>number</i>

This command does not apply to nonmanageable modems.

Use the **show modem log** command to view modem events.

Troubleshoot and Manage Modems

This section describes how to troubleshoot the integrated modems and remove them from dial-up connection services.

Perform a Modem Startup Test

To perform diagnostic testing on all the installed modems during the system’s initial startup or rebooting process, perform the following task in global configuration mode:

Task	Command
Perform diagnostic testing for all modems.	modem startup-test

The results of the modem startup test are displayed in the *Status* column of the **show modem** command’s output. Modems that pass the diagnostic test are marked as *Idle*, *Busy*, *Downloading*, and *Reset*. Modems that fail the diagnostic test are marked as *Bad**. These modems cannot be used for call connections. Depending on how many modems are installed, this diagnostic test may take from 5 to 15 minutes to complete.

Perform additional testing on an inoperative modem by executing the **test modem back-to-back** command. The **no modem startup-test** command disables startup testing.

See the “Cisco AS5200 Configuration Examples” section at the end of this chapter for the “Modem Startup Test Example.”

Test Two Modems Back-to-Back

Perform additional testing on a modem suspected of being inoperable by conducting a series of internal back-to-back connections and data transfers between two modems. All modem test connections occur inside the access server. For example, if mobile users cannot dial into modem 2/5 (which is the sixth modem port on the modem board in the second chassis slot), attempt a back-to-back test with modem 2/5 and a known-functioning modem such as modem 2/6.

Use the following command in EXEC mode to perform internal back-to-back modem tests between two modems:

Task	Command
Perform internal back-to-back modem tests between two modems.	test modem back-to-back <i>first-slot/modem-number second-slot/modem-number</i>

You might need to enable this command on several different combinations of modems to determine which one is not functioning properly. A pair of operable modems successfully connect and complete transmitting data in both directions. An operable modem and an inoperable modem do not successfully connect with each other.

See the “Cisco AS5200 Configuration Examples” section at the end of this chapter for the “Back-to-Back Modem Test Example.”

Remove Inoperable Modems from Service

To remove modems from service and indicate them as suspected or proven to be inoperable, perform the following task in line configuration mode:

Task	Command
Specify a modem as inoperable.	modem bad

If you mark a *single* modem as inoperable using this command, it appears as *Bad*—without the asterisk (*)—in the *Status* column of the **show modem** command’s output for that particular modem. A modem marked inoperable by the **modem startup-test** command appears as *Bad** in the **show modem** command output for that particular modem. Use the **no modem bad** command to unmark a modem as *Bad** or *Bad* and restore it for dial-up connection services.

See the “Cisco AS5200 Configuration Examples” section at the end of this chapter for the “Inoperable Modem Example.”

Hold and Reset a Modem

To reset and isolate the modem hardware for extensive troubleshooting, perform the following task in line configuration mode:

Task	Command
Reset and isolate the modem hardware.	modem hold-reset

Use this command if you are experiencing extreme modem behavior (for example, if the modem is uncontrollably dialing into the network). This command prevents the modem from establishing software relationships such as those created by the **test back-to-back modem** command and the **modem startup-test** command. The modem is unusable while the **modem hold-reset** command is configured.

This command is also used to reset a modem that is frozen in a suspended state. Disable the suspended modem with the **modem hold-reset** command, and then restart hardware initialization with the **no modem hold-reset** command.

A modem decommissioned by the **modem hold-reset** command does not accept modem firmware upgrades using the **copy modem** command.

See the “Cisco AS5200 Configuration Examples” section at the end of this chapter for the “Hold and Reset Modem Example.”

Disable a Modem from Dial-Up Services

To disable modems from dialing or answering calls, perform one of the following tasks in line configuration mode:

Task	Command
Gracefully disable a modem from dial-up services.	modem busyout
Abruptly shut down a modem from dial-up services.	modem shutdown

The **modem busyout** command is not executed until the active modem is idle. No active connections are interrupted when you use this command. In contrast, the **modem shutdown** command immediately terminates all active connections on the specified modem. The resulting modem status for both these commands is the same.

Enable the **no** form of these commands to restore a modem for dial-up services.

You can still configure the following commands on a disabled modem:

- **test modem back-to-back**
- **clear modem**
- **modem bad**
- **copy modem**

See the “Cisco AS5200 Configuration Examples” section at the end of this chapter for the “Disable Modem Examples.”

Debug a Modem

To debug a modem or group of modems, perform the following tasks in EXEC mode:

Task	Command
Debug a modem’s out-of-band port, which is used to poll modem events.	debug modem oob [<i>slot/modem-port</i> group <i>group-number</i>]
Debug a call-switching module, which is used to connect calls.	debug modem csm [<i>slot/modem-port</i> group <i>group-number</i>]
Debug the call trace, which determines why calls are terminated. Use this keyword only with manageable modems. Upload the call trace on normal , abnormal , or all call terminations.	debug modem trace [normal abnormal all] [<i>slot/modem-port</i> group <i>group-number</i>]

Use the **debug modem** command to do the following:

- Debug the out-of-band port, which is used for polling modem events
- Debug the call-switching module, which is used to connect calls
- Debug the uploaded modem call trace to the syslog server, which determines why calls are terminated

To create an asynchronous interface and use it as a group interface associated with a group of modems, refer to the “Configuring Interfaces” chapter in the *Configuration Fundamentals Configuration Guide*.

See the “Cisco AS5200 Configuration Examples” section at the end of this chapter for the “Debug Modem Example.”

Send AT Commands to Manageable Modems

Each Cisco AS5200 manageable modem has one out-of-band port, which is used to poll modem statistics and transmit AT commands. The Cisco IOS software uses a directly connected session to transfer information to an out-of-band port. To transfer AT commands, you must permit a directly connected session on a modem, open a directly connected session and send AT commands to a modem, and clear a directly connected session from a modem.

Note This section does not apply to nonmanageable modems, which do not have out-of-band ports.

Permit a Directly Connected Session

To permit a manageable modem to accept a directly connected session, which is enabled by default on all modems, perform the following task in line configuration mode:

Task	Command
Permit a modem to accept a directly connected session.	modem at-mode-permit

The **no modem at-mode-permit** command disables a modem from accepting a direct connection, which is useful for ensuring modem security.

See the “Cisco AS5200 Configuration Examples” section for the “Directly Connected Session Example.”

Open a Directly Connected Session and Transmit AT Commands

To open a directly connected session and enable AT command mode (which is needed to transmit to a manageable modem), perform the following command in EXEC mode:

Task	Command
Open a directly connected session and enter AT command mode.	modem at-mode slot/modem-port

Once you enable this command, you can transmit AT commands directly from your terminal session. Most incoming or outgoing calls on the modem are not interrupted when you open a directly connected session and transmit AT commands. However, some AT commands interrupt a call—for example, the **ATH** command, which hangs up a call.

Open and close one directly connected session at a time. Multiple open directly connect sessions slow down modem performance.

Refer to the *12-Port Modem AT Command Set and Register Summary* publication (part of the Cisco AS5200 documentation set) for a complete list of AT commands that you can transmit.

See the “Cisco AS5200 Configuration Examples” section at the end of this chapter for the “Transmit AT Command Example.”

Clear a Directly Connected Session

You can clear or terminate an active directly connected session in two ways:

- Press **Ctrl-C** after transmitting all AT commands as instructed by the system when you enter AT command mode.
- Enter a second Telnet session and enable the **clear modem at-mode slot/modem-port** command in EXEC configuration mode. This method is used for closing a directly connected session that may have been mistakenly left open by the first Telnet session.

See the “Cisco AS5200 Configuration Examples” section at the end of this chapter for the “Clear Session from a Second Telnet Session Example.”

Poll Manageable Modems

Each manageable modem has one out-of-band port, which is used for polling modem statistics.

Note This section does not apply to nonmanageable modems, which do not have out-of-band ports.

Set Time Interval between Polls

To set the time interval between the polls that are sent to the local modems for reporting modem status and statistics, perform the following task in global configuration mode:

Task	Command
Specify the number of seconds between polls. The default is 12 seconds. The configuration range is 2 to 120 seconds.	modem poll time <i>seconds</i>

See the “Cisco AS5200 Configuration Examples” section at the end of this chapter for the “Time Interval Example.”

Poll for Modem Statistics

To poll for a modem’s status and statistics through its out-of-band port, perform the following task in line configuration mode:

Task	Command
Poll for a modem’s status and statistics.	modem status-poll

The **no modem status-poll** command disables status polling through the out-of-band port for a specified modem.

See the “Cisco AS5200 Configuration Examples” section at the end of this chapter for the “Modem Polling Example.”

Set Polling Attempts

To set the maximum number of polling attempts used to retrieve a local modem’s status or statistics, perform the following task in global configuration mode:

Task	Command
Set maximum number of polling attempts. The default is three polling attempts. The configuration range is from 0 to 10 attempts.	modem poll retry <i>number</i>

If the number of attempts to retrieve modem status or statistics exceeds the *number* you define, the out-of-band port is removed from operation. In this case, you must reset the modem hardware using the **clear modem** command.

See the “Cisco AS5200 Configuration Examples” section at the end of this chapter for the “Polling Attempts Example.”

Download Modem Firmware

To download firmware to modems in the access server, perform one of the following tasks in EXEC mode:

Task	Command
Copy modem firmware from Flash memory to a modem.	copy flash modem
Copy modem firmware from a TFTP server to a modem.	copy tftp modem
Copy modem firmware remotely from a network server to a modem.	copy rcp modem

After you enter a command, you are prompted for the download destination, the remote host name, and the path leading to the source modem firmware, as requested by the system software.

If a modem that you want to upgrade is busy with a call when the **copy modem** command is enabled, the upgrade for that modem yields until the active call is dropped. All other idle modems in the upgrade range proceed with the downloading operation.

See the “Cisco AS5200 Configuration Examples” section at the end of this chapter for the “Download Firmware Examples.”

Configure Virtual Asynchronous Interfaces

Typically, asynchronous protocol features (such as PPP) are configured on asynchronous interfaces in the Cisco IOS software. However, the Cisco IOS software also enables you to configure asynchronous protocol features on VTY lines.

Refer to the “Enable SLIP and PPP on Virtual Asynchronous Interfaces” section in the “Configuring SLIP and PPP” chapter in this publication for more information about configuring virtual asynchronous interfaces.

Line Configuration Examples

The following sections provide line configuration examples:

- Creating Additional Virtual Terminal Lines Example
- Eliminating Virtual Terminal Lines Example
- Basic Line Configuration Example
- Password Checking Examples
- Banner Example

Creating Additional Virtual Terminal Lines Example

In the following example, the user creates and configures the maximum 100 virtual terminal lines with the “no login” feature:

```
line vty 0 99
no login
```

Eliminating Virtual Terminal Lines Example

In the following example, the user eliminates virtual terminal line number 5 and all higher-numbered virtual terminal lines. Only virtual terminal lines 0 to 4 will remain.

```
no line vty 5
```

Basic Line Configuration Example

In the following example, the user configures console line 0, auxiliary line 0, and virtual terminal lines 0 through 4:

```
line vty 0 4
login
line con 0
password baskerville
line aux 0
password Mypassword
no exec
access-class 1 in
speed 19200
line vty 0
exec-timeout 0 0
password Mypassword
line vty 1
exec-timeout 0 0
password Mypassword
line vty 2
exec-timeout 0 0
password Mypassword
line vty 3
password Mypassword
line vty 4
password Mypassword
```

Password Checking Examples

The following example shows how to enable password checking for a virtual terminal line 1:

```
line vty 1
 login
 password letmein
```

The following example shows how to enable password checking on a user basis:

```
username jksmith password 0 letmein
username lmjones password 0 littlerock
...
line vty 1
 login local
```

Banner Example

The following example shows how to use the **banner** global configuration commands and the **no exec-banner** line configuration command to notify your users that the server is going to be reloaded with new software:

```
! The EXEC and MOTD banners are inappropriate for the VTYS.
line vty 0 4
 no exec-banner
!
banner exec /
 This is Cisco Systems training group router.

Unauthorized access prohibited.
/
!
banner incoming /
 You are connected to a Hayes-compatible modem.

Enter the appropriate AT commands.
Remember to reset anything to change before disconnecting.
/
!
banner motd /
 The router will go down at 6pm for a software upgrade
/
```

When someone connects to the router, the MOTD banner appears before the login prompt. After the user successfully logs in to the router, the EXEC banner or incoming banner will be displayed, depending on the type of connection. For a reverse Telnet login, the incoming banner will be displayed. For all other connections, the router will display the EXEC banner.

Callback Examples

The following sections provide examples for callback:

- Call Back Clients Connecting to the EXEC Prompt Example
- Call Back an ARA Client Example
- Call Back a PPP Client Example

Call Back Clients Connecting to the EXEC Prompt Example

The following example shows the process to configure an outgoing callback on the same line as the incoming request. The **login local** command enables local username authentication on lines 4 and 7. Re-authentication is required upon reconnection.

```
Router(config)# service exec-callback
Router(config)# username milarepa callback-dialstring "" password letmein
Router(config)# line 4
Router(config-line)# login local
Router(config-line)# exit
Router(config)# line 7
Router(config-line)# login local
```

Call Back an ARA Client Example

The following example shows the process of configuring callback to an ARA client on line 7. The **login local** command enables local username authentication on lines 4 and 7. Line 7 will always be used for ARA callback, whether the incoming call enters line 4, 7, or 8.

```
Router(config)# appletalk routing
Router(config)# arap callback
Router(config)# arap network 422 router test
Router(config)# username excalibur callback-dialstring "123456" callback-line 7
password guenivere
Router(config)# line 4
Router(config-line)# login local
Router(config-line)# modem InOut
Router(config-line)# autoselect arap
Router(config-line)# arap enable
Router(config-line)# exit
Router(config)# line 7
Router(config-line)# login local
Router(config-line)# modem InOut
Router(config-line)# autoselect arap
Router(config-line)# arap enable
Router(config-line)# exit
Router(config)# line 8
Router(config-line)# login local
Router(config-line)# modem InOut
Router(config-line)# autoselect arap
Router(config-line)# arap enable
```

Call Back a PPP Client Example

The following example shows the process of configuring callback to a PPP client on rotary 77. PAP authentication is enabled for PPP on the asynchronous interfaces. The **login local** command enables local username authentication on lines 7, 8, and 9. The remote PPP client's host name is Ted, and the callback number is fixed at 1234567.

```
Router(config)# username Ted callback-dialstring "1234567" callback-rotary 77
password Rhoda
Router(config)# interface async7
Router(config-if)# ip unnumbered ethernet0
Router(config-if)# encapsulation ppp
Router(config-if)# no keepalive
Router(config-if)# async default ip address 1.1.1.1
Router(config-if)# async mode interactive
Router(config-if)# ppp callback accept
Router(config-if)# ppp authentication pap
Router(config-if)# exit
```

```

Router(config)# interface async8
Router(config-if)# ip unnumbered ethernet0
Router(config-if)# encapsulation ppp
Router(config-if)# no keepalive
Router(config-if)# async default ip address 1.1.1.2
Router(config-if)# async mode interactive
Router(config-if)# ppp callback accept
Router(config-if)# ppp authentication pap
Router(config-if)# exit

Router(config)# interface async9
Router(config-if)# ip unnumbered ethernet0
Router(config-if)# encapsulation ppp
Router(config-if)# no keepalive
Router(config-if)# async default ip address 1.1.1.3
Router(config-if)# async mode interactive
Router(config-if)# ppp callback accept
Router(config-if)# ppp authentication pap
Router(config-if)# exit

Router(config)# line 7
Router(config-line)# login local
Router(config-line)# modem InOut
Router(config-line)# rotary 77
Router(config-line)# autoselect ppp
Router(config-line)# exit

Router(config)# line 8
Router(config-line)# login local
Router(config-line)# modem InOut
Router(config-line)# rotary 77
Router(config-line)# autoselect ppp
Router(config-line)# exit

Router(config)# line 9
Router(config-line)# login local
Router(config-line)# modem InOut
Router(config-line)# rotary 77
Router(config-line)# autoselect ppp

```

Cisco AS5200 Configuration Examples

This section provides the following example Cisco AS5200 configurations, including a sample startup configuration:

- Cisco AS5200 Startup Sample Configuration Example
- ISDN Analog Calls Example
- Channelized T1 Analog Calls Example
- Modem Startup Test Example
- Back-to-Back Modem Test Example
- Inoperable Modem Example
- Hold and Reset Modem Example
- Disable Modem Examples
- Debug Modem Example
- Directly Connected Session Example

- Transmit AT Command Example
- Clear Session from a Second Telnet Session Example
- Time Interval Example
- Modem Polling Example
- Polling Attempts Example
- Download Firmware Examples

Cisco AS5200 Startup Sample Configuration Example

Take the following steps to configure the Cisco AS5200 access server.

Step 1 Configure a base security and local database with the following commands:

```
version 11.1
service udp-small-servers
service tcp-small-servers

hostname brasil4_brasil
!
aaa new-model
aaa authentication login default radius
aaa authentication login console none
aaa authentication ppp default radius local
aaa authorization exec radius if-authenticated none
aaa authorization network radius if-authenticated none
aaa accounting exec start-stop radius
aaa accounting network start-stop radius
!
username Brasil4_1003 password 7 060A0E23
username guest password 7 021201481F
ip address-pool local
```

Step 2 Set the ISDN switch type.

```
isdn switch-type primary-5ess
```

Step 3 Set the parameters for the two T1 controller interfaces (which are the primary ISDN lines), to accept incoming calls and send outgoing calls through each T1 Primary Rate Interface (PRI).

```
controller T1 0
framing esf
clock source line primary
linecode b8zs
pri-group timeslots 1-24
!
controller T1 1
framing esf
clock source line secondary
linecode b8zs
pri-group timeslots 1-24
```

Step 4 Configure the Ethernet and serial interfaces.

```
interface Ethernet 0
ip address 10.1.1.1 255.0.0.0
!
interface Serial 0
ip address 100.1.1.1 255.0.0.0
encapsulation ppp
```

```

!
interface Serial 1
 ip address 101.1.1.1 255.0.0.0
 encapsulation frame-relay IETF
 clockrate 2000000
 frame-relay lmi-type ansi
 frame-relay map ip 101.1.1.2 101 broadcast

```

- Step 5** Once you create the T1 controllers, two corresponding D channel serial interfaces are instantly created. Serial interface 0:23 is the D channel for controller T1 0, and serial interface 1:23 is the D channel for controller T1 1. You must configure each serial interface to receive incoming and send outgoing calls.

```

interface Serial 0:23
 ip unnumbered Ethernet0
 encapsulation ppp
 no keepalive
 isdn incoming-voice modem
 dialer idle-timeout 4000
 dialer map ip 10.1.1.10 name Brasil4_1003 broadcast 1111111
 dialer-group 1
 ppp authentication chap
!
interface Serial 1:23
 ip unnumbered Ethernet 0
 encapsulation ppp
 no keepalive
 isdn incoming-voice modem
 dialer idle-timeout 4000

```

- Step 6** Configure one group asynchronous interface to correspond with the 48 integrated modems.

```

interface Group-Async 1
 ip unnumbered Ethernet 0
 encapsulation ppp
 async default routing
 async mode interactive
 peer default ip address pool local
 no cdp enable
 ppp authentication chap
 group-range 1 48

```

- Step 7** Enable a routing protocol to run on the access server.

```

router igrp 100
 network 100.0.0.0
 network 101.0.0.0
 network 10.0.0.0

```

- Step 8** Define a range of IP addresses and configure the integrated modems for remote access clients using the Cisco AS5200. For example, the first remote client dialing in uses IP address 10.1.1.100, the second remote client uses 10.1.1.101, and so on. The 48 integrated modems are configured to transmit and receive data at 115.2 Kbps.

```

ip local pool local 10.1.1.100 10.1.1.148
no ip classless
tacacs-server host 223.255.254.250
tacacs-server key arcticfr32
!
dialer-list 1 protocol ip permit
!
line con 0
 exec-timeout 0 0

```

```

line 1 48
  exec-timeout 0 0
  autoselect ppp
  modem InOut
  transport input all
  stopbits 1
  rxspeed 115200
  txspeed 115200
  flowcontrol hardware
line aux 0
line vty 0 4
  password test
  login
!
end

```

ISDN Analog Calls Example

The following example configures incoming and outgoing ISDN analog calls at the controller T1 0 interface:

```

AS5200(config)# interface serial 0:23
AS5200(config-if)# isdn incoming-voice modem
AS5200(config-if)#

```

Channelized T1 Analog Calls Example

The following example shows how the **cas-group** command interrelates with the **pri-group** command and the **channel-group** command. The range of timeslots that you allocate to a PRI group, channel group, and channel associated signaling group must match the timeslot allocations that your central office chooses to use.

The following configuration configures one ISDN PRI group using timeslots 1 to 10:

```

AS5200(config-controller)# pri-group timeslots 1-10
AS5200(config-controller)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1:0, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1:1, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1:2, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1:3, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1:4, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1:5, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1:6, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1:7, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1:8, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1:9, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1:23, changed state to up
%LINK-3-UPDOWN: Interface Serial1:23, changed state to up

```

Channelized T1 data is transmitted over timeslots 11 through 16, which are assigned to channel group 11 in the next configuration example. However, notice how the earlier attempt to configure channel group 1 is denied because timeslot 1 is used by the previous ISDN PRI group configuration.

```

AS5200(config-controller)# channel-group 1 timeslots 11-16
%Channel-group 1 is already an isdn channel
AS5200(config-controller)# channel-group 11 timeslots 11-16
AS5200(config-controller)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1:11, changed state to down
AS5200(config-controller)#
%LINK-3-UPDOWN: Interface Serial1:11, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1:11, changed state to up

```

The channel associated signal E&M is configured on the remaining 17 to 23 timeslots, which are used for incoming and outgoing analog calls. Notice how channel number 12 cannot be used to configure these timeslots, because it is used in the previously configured channel group range, which is timeslots 11 to 16.

```
AS5200(config-controller)# cas-group 12 timeslots 17-23
The channel has been assigned to pri or channel-group
AS5200(config-controller)# cas-group 17 timeslots 17-23
AS5200(config-controller)#
```

Modem Startup Test Example

The following example shows how to perform a startup test on the integrated Cisco AS5200 modems:

```
configure terminal
modem startup-test
```

Display the results of the modem-startup test after you reboot the system by enabling the **show modem** command, as shown in the following example:

```
AS5200# show modem 2/3
Mdm Typ Status Tx/Rx G Duration TX RX RTS CTS DSR DCD DTR
2/3 VFC Bad* 19200/19200 0 00:17:11 x x x x x x x

Modem 2/3, AS5200 Manageable Modem, TTY4
Firmware (Boot) Rev: 1.0.23(1.0.5)
Modem config: Incoming and Outgoing
Protocol: Normal, Compression: None
Management config: status and AT session polling
TX signals: -17 dBm, RX signals: -33 dBm
```

Back-to-Back Modem Test Example

The following example shows how to perform a back-to-back modem test between modems on a Cisco AS5200 Universal Access Server.

The first part of the example shows a successful connection between modem 2/1 and modem 2/0, which verifies normal operating conditions between these two modems. However, when modem 2/1 is tested against modem 2/3, the back-to-back modem test fails. Therefore, modem 2/3 is suspected or proven to be inoperable. Modem 2/3 is removed from dial-up services through the use of the **modem bad** command on line 28.

```
AS5200# test modem back-to-back 2/1 2/0
Repetitions (of 10-byte packets) [1]: 10
AS5200#
%MODEM-5-B2BCONNECT: Modems (2/1) and (2/0) connected in back-to-back test: CONN
ECT9600/REL-MNP
%MODEM-5-B2BMODEMS: Modems (2/0) and (2/1) completed back-to-back test: success/
packets = 20/20
AS5200# test modem back-to-back 2/1 2/3
Repetitions (of 10-byte packets) [1]: 10
AS5200#
%MODEM-5-BADMODEMS: Modems (2/3) and (2/1) failed back-to-back test: NOCARRIER
AS5200# configure terminal
AS5200(config)# line 28
AS5200(config-line)# modem bad
AS5200(config-line)# end
```

Once you enter the **test modem back-to-back** command, you must define the number of packets transmitted between modems at the *Repetitions* prompt. The ideal range of packets to transmit and receive is from 1 to 100. The default is 1 packet. The response message (for example, “success/packets = 2/2”) tells you how many packets were successfully sent in *both* directions compared to the total number of packets attempted to be sent in both directions. Because the software reports the packet total in both directions, the reported numbers are *two times* the number you originally specify.

Inoperable Modem Example

The first part of the following example shows a successful connection between modem 2/1 and modem 2/0, which verifies normal operating conditions between these two modems. However, when modem 2/1 is tested against modem 2/3, the back-to-back modem test fails. Therefore, modem 2/3 is suspected or proven to be inoperable. Modem 2/3 is removed from dial-up services through the use of the **modem bad** command on line 28 (see Table 6).

```
AS5200# test modem back-to-back 2/1 2/0
Repetitions (of 10-byte packets) [1]: 10
AS5200#
%MODEM-5-B2BCONNECT: Modems (2/1) and (2/0) connected in back-to-back test: CONN
ECT9600/REL-MNP
%MODEM-5-B2BMODEMS: Modems (2/0) and (2/1) completed back-to-back test: success/
packets = 20/20
AS5200# test modem back-to-back 2/1 2/3
Repetitions (of 10-byte packets) [1]: 10
AS5200#
%MODEM-5-BADMODEMS: Modems (2/3) and (2/1) failed back-to-back test: NOCARRIER
AS5200# configure terminal
AS5200(config)# line 28
AS5200(config-line)# modem bad
AS5200(config-line)# end
```

Hold and Reset Modem Example

The following example disables a suspended modem and resets its hardware initialization:

```
configure terminal
line 4
modem hold-reset
no modem hold-reset
```

Disable Modem Examples

The following example gracefully disables the modem associated with line 1 from dialing and answering calls. The modem is disabled only after all active calls on the modem are dropped.

```
configure terminal
line 1
modem busyout
```

The following example abruptly shuts down the modem associated with line 2. All active calls on the modem are dropped immediately.

```
configure terminal
line 2
modem shutdown
```

Note You do not specify a *slot/modem-port* number with the **modem busyout** or **modem shutdown** commands. Instead you configure the asynchronous line associated with the modem.

Debug Modem Example

The following example is sample output from the **debug modem trace abnormal** command:

```
AS5200# debug modem trace abnormal 1/14

Modem 1/14 Abnormal End of Connection Trace. Caller 123-4567
Start-up Response: AS5200 Modem, Firmware 1.0
Control Reply: 0x7C01
DC session response: brasil firmware 1.0
RS232 event:
DSR=On, DCD=On, RI=Off, TST=Off
changes: RTS=No change, DTR=No change, CTS=No change
changes: DSR=No change, DCD=No change, RI=No change, TST=No change
Modem State event: Connected
Connection event: Speed = 19200, Modulation = VFC
Direction = Originate, Protocol = reliable/LAPM, Compression = V42bis
DTR event: DTR On
Modem Activity event: Data Active
Modem Analog signal event: TX = -10, RX = -24, Signal to noise = -32
End connection event: Duration = 10:34-11:43,
Number of xmit char =          67, Number of rcvd char = 88, Reason: Watchdog Time-out.
```

Directly Connected Session Example

The following example permits modem 1/1 on TTY line 1 to accept a directly connected session using the **modem at-mode-permit** command:

```
configure terminal
line 1
modem at-mode-permit
```

Transmit AT Command Example

The following example opens a directly connected session on modem 1/1, enters AT command mode on modem 1/1, and transmits the **ATH** command through the out-of-band port on modem 1/1:

```
AS5200# modem at-mode 1/1
You are now entering AT command mode on modem (slot 1 / port 1).
Please type CTRL-C to exit AT command mode.
at%v

MNP Class 10 V.34/V.FC Modem Rev 1.0/85

OK
at\s

IDLE          000:00:00
LAST DIAL

NET ADDR:      FFFFFFFF
MODEM HW: SA 2W United States
4 RTS 5 CTS 6 DSR - CD 20 DTR - RI
MODULATION    IDLE
MODEM BPS     28800  AT%G0
MODEM FLOW    OFF    AT\G0
MODEM MODE    AUT    AT\N3
V.23 OPR.    OFF    AT%F0
AUTO ANS.    ON     ATSO=1
SERIAL BPS    115200 AT%U0
BPS ADJUST    OFF    AT\J0
SPT BPS ADJ.  0     AT\W0
ANSWER MESSGS ON    ATQ0
```

```
SERIAL FLOW    BHW    AT\Q3
PASS XON/XOFF  OFF    AT\X0
PARITY        8N     AT
```

The manageable modem returns “OK” if the AT command you transmit is successfully enabled.

Clear Session from a Second Telnet Session Example

The following example shows how to execute the **modem at-mode** command from a Telnet session:

```
AS5200# modem at-mode 1/1
```

The following example shows how to execute the **clear modem at-mode** command from a second Telnet session while the first Telnet session is connected to the modem:

```
AS5200# clear modem at-mode 1/1
clear "modem at-mode" for modem 1/1 [confirm]
AS5200#
```

The following output is displayed in the first Telnet session once the modem is cleared by the second Telnet session:

```
Direct connect session cleared by vty0 (171.69.1.164)
```

Time Interval Example

The following example sets the time interval between polls to 10 seconds using the **modem poll time** configuration command:

```
configure terminal
modem poll time 10
```

Modem Polling Example

The following example enables modem status polling through the out-of-band port connected to line 1:

```
configure terminal
line 1
modem status-poll
```

Polling Attempts Example

The following example configures the server to attempt to retrieve statistics from a local modem up to five times before discontinuing the polling effort:

```
configure terminal
modem poll retry 5
```

Download Firmware Examples

The following example shows how to copy the modem firmware file called *modem_upgrade* from the TFTP server called *Modem_Server* to modem 2/0, which is installed in the Cisco AS5200 access server:

```
AS5200# copy tftp modem
Modem Firmware Download Modem Numbers? 2/0
Address or name of remote host [UNKNOWN]? Modem_Server
Source file name? dirt/elem/modem_upgrade
```

```

Accessing file 'dirt/elem/modem_upgrade on Modem_Server...
Loading dirt/elem/modem_upgrade .from 223.255.254.254 (via Ethernet0): ! [OK]

Loading dirt/elem/modem_upgrade from 223.255.254.254 (via Ethernet0):
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[OK - 237503/278528 bytes]

AS5200#
%MODEM-5-DL_START: Modem (2/0) started firmware download
%MODEM-5-DL_GOOD: Modem (2/0) completed firmware download:
MNPCClass10V.34/V.FCModemRev1.0.23/85.23/85

```

Once the modem firmware successfully downloads, a response message reports the new version number of the installed modem firmware—for example, Rev1.0.23/85.23/85—as shown the last line of this example.

The next time you use the **copy tftp modem** command, the interactive display shows the name of the previously accessed remote server as the default setting. For example, *Modem_Server* replaces *UNKNOWN* as shown in this example.

You might want to copy the file to one modem first for testing before copying the file to all the modems in the access server.

The following example shows how to download the same modem firmware file from the TFTP server *Modem_Server* to all the modems in the Cisco AS5200 access server:

```

AS5200# copy tftp modem
Modem Firmware Download Modem Numbers? all
Address or name of remote host [UNKNOWN]? Modem_Server
Source file name? dirt/elem/modem_upgrade
Accessing file 'dirt/elem/modem_upgrade on Modem_Server...
Loading dirt/elem/modem_upgrade .from 223.255.254.254 (via Ethernet0): ! [OK]

Loading dirt/elem/modem_upgrade from 223.255.254.254 (via Ethernet0):
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[OK - 237503/278528 bytes]

AS5200#
%MODEM-5-DL_START: Modem (2/0) started firmware download
%MODEM-5-DL_START: Modem (2/1) started firmware download
%MODEM-5-DL_START: Modem (2/2) started firmware download
%MODEM-5-DL_START: Modem (2/3) started firmware download
%MODEM-5-DL_START: Modem (2/4) started firmware download
%MODEM-5-DL_START: Modem (2/5) started firmware download
%MODEM-5-DL_START: Modem (2/6) started firmware download
%MODEM-5-DL_START: Modem (2/7) started firmware download
%MODEM-5-DL_START: Modem (2/8) started firmware download
%MODEM-5-DL_START: Modem (2/9) started firmware download
%MODEM-5-DL_START: Modem (2/10) started firmware download
%MODEM-5-DL_START: Modem (2/11) started firmware download
%MODEM-5-DL_START: Modem (2/12) started firmware download
%MODEM-5-DL_START: Modem (2/13) started firmware download
%MODEM-5-DL_START: Modem (2/14) started firmware download
%MODEM-5-DL_START: Modem (2/15) started firmware download
%MODEM-5-DL_START: Modem (2/16) started firmware download
%MODEM-5-DL_START: Modem (2/17) started firmware download
%MODEM-5-DL_START: Modem (2/18) started firmware download
%MODEM-5-DL_START: Modem (2/19) started firmware download
%MODEM-5-DL_START: Modem (2/20) started firmware download
%MODEM-5-DL_START: Modem (2/21) started firmware download
%MODEM-5-DL_START: Modem (2/22) started firmware download
%MODEM-5-DL_START: Modem (2/23) started firmware download
%MODEM-5-DL_GOOD: Modem (2/2) completed firmware download:
MNPCClass10V.34/V.FCModemRev1.0.23/85.23/85

```

```
%MODEM-5-DL_GOOD: Modem (2/10) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/4) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/6) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/7) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/12) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/11) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/13) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/1) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/14) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/19) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/22) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/5) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/8) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/9) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/17) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/0) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/3) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/21) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/16) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/15) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/18) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/20) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
%MODEM-5-DL_GOOD: Modem (2/23) completed firmware download:
MNPClass10V.34/V.FCModemRev1.0.23/85.23/85
```

