Configuring and Managing Integrated Modems

The Cisco IOS software provides commands that manage modems that reside inside access servers or routers in the form of modem cards. This chapter describes the modem management tasks. It includes the following main sections:

- Modems and Modem Feature Support
- Managing Modems
- Configuration Examples for Modem Management

For additional instructions for configuring Cisco access servers, see the chapter "Configuring and Managing Cisco Access Servers and Dial Shelves" in this publication.

To identify the hardware platform or software image information associated with a feature, use the Feature Navigator on Cisco.com to search for information about the feature or refer to the software release notes for a specific release. For more information, see the "Identifying Supported Platforms" section in the "Using Cisco IOS Software" chapter.

For a complete description of the commands mentioned in this chapter, refer to the *Cisco IOS Dial Technologies Command Reference*. To locate documentation of other commands that appear in this chapter, use the command reference master index or search online.

Modems and Modem Feature Support

The Cisco IOS software supports three types of integrated modems for Cisco access servers and access routers:

- Modem ISDN channel aggregation (MICA) digital modem
- NextPort digital modem
- NM-AM network module analog modem

Table 1 lists device support for each of the Cisco access server hardware platforms.

Table 1 Cisco IOS Modems and Modem Feature Support

Device Support	Cisco AS5300	Cisco AS5350	Cisco AS5400	Cisco AS5800	Cisco 2600/3600 Series Routers
Integrated modems	6- and 12-port MICA	60-port NextPort CSM v6DFC	108-port NextPort CSM v6DFC	72- and 144-port MICA 324-port NextPort CSM v6DFC	6-port, 12-port, 18-port, 24-port, or 30-port MICA NM-DM 8- and 16-port analog NM-AM
V.90	Yes	Yes	Yes	Yes	Yes with NM-DM
V.110	Yes	Yes	Yes	Yes	Yes with NM-DM
V.120	No, CPU only	Yes	Yes	Yes with 324-port NextPort[1] CSM v6DFC	No, CPU only

1. For more detailed information regarding the V.120 functionalities that are supported both by NextPort and Cisco IOS software, see the section "V.120 Bit Rate Adaptation Standard."

Note If the platform is using MICA technologies modems, the V.120 rate adaptation is done by CPU on vty lines like protocol translation sessions.

Note Typically, parity and databits commands have no effect on modem lines without UART. However, since modem lines such as WIC-1AM and WIC-2AM are implemented though UART and use an internal UART, the parity and databits commands can be used to set async parity framing. The following commands may be used on WIC-AM modem cards although changing the DTE speed and stopbits on these lines have no effect: parity, databits, stopbits, flowcontrol, speed, rxspeed, and txspeed.

The following sections summarize the standards supported by modems in the Cisco access servers. See Table 2 through Table 5 for a summary and comparison of the Cisco IOS commands used for the MICA and NextPort modems.

V.90 Modem Standard

Study Group 16 of the International Telecommunication Union Telecommunication Standardization Sector (ITU-T) developed the V.90 modem standard for multimedia systems. The V.90 standard describes a digital modem and analog modem pair for use on the public switched telephone network (PSTN). V.90 modems are designed for connections that are digital at one end and have only one digital-to-analog conversion. The V.90 standard is expected to be widely used for applications such as Internet and online service access. Download speeds of up to 56,000 bits per second (bps) are possible, depending on telephone line conditions, with upload speeds of up to 33,600 bps.

V.110 Bit Rate Adaption Standard

V.110 is a bit rate adaptation standard defined by the ITU that provides a standard method of encapsulating data over global system for mobile telecommunication (GSM) and ISDN networks. V.110 allows for reliable transport of asynchronous or synchronous data. V.110 adapts a low-speed connection to an ISDN B channel allowing the remote station or terminal adapter to use the fast call setup times offered by ISDN. This feature allows V.110 calls to be originated and terminated over ISDN. It also enables GSM wireless connectivity.

V.110, as an alternative to V.120, provides DTE with V-series type interfaces with access to ISDN network by bit stuffing. Many V.110 devices are used in Europe and Japan. In Japan, MICA supports the Personal-Handyphone-System Internet Access Forum Standard (PIAFS) protocol, which is similar to V.110.

The V.110 implementation for calls on MICA modems is managed by special boardware and modem code, along with the appropriate Cisco IOS image, in a manner similar to other modulation standards. This MICA V.110 implementation provides V.110 user rates ranging from 600 bps to 38,400 bps.

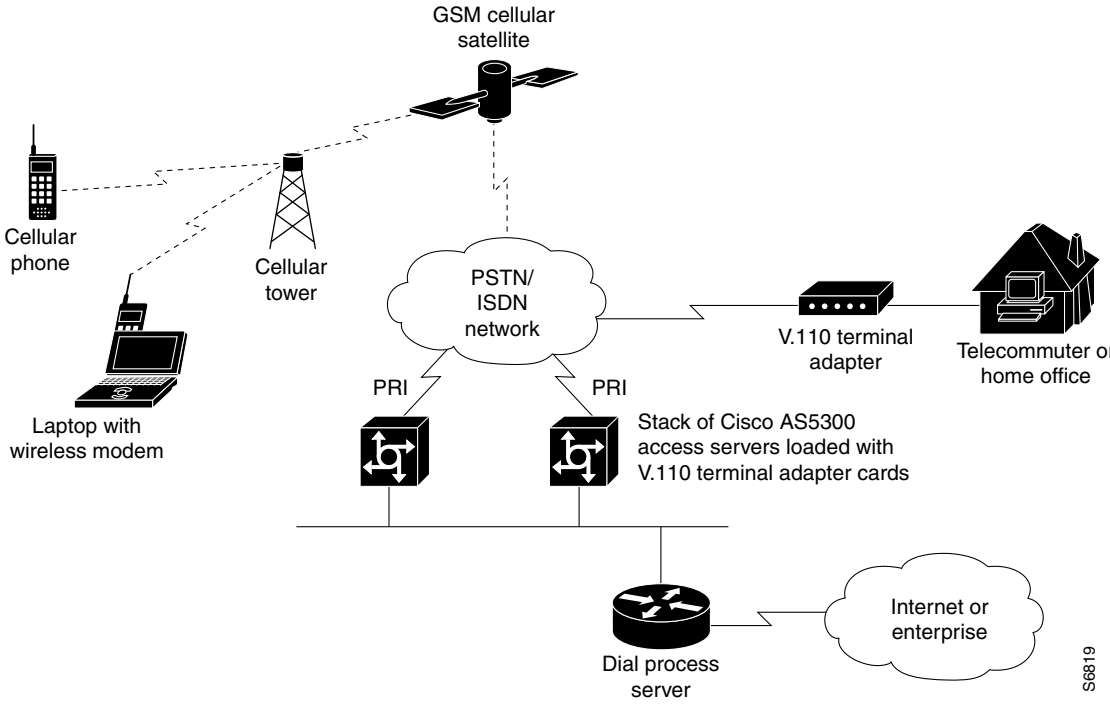
V.110 is supported on the following Cisco devices and network modules:

- Cisco AS5300-series access servers
- Cisco 3620, 3640, and 3660 access routers
- NM-6DM, NM-12DM, NM-18DM, NM-24DM, and NM-30DM network modules

The digital signal processors (DSPs) on the board can function as either modems or V.110 terminal adapters (or V.120 terminal adapters for NextPort DSPs). Based on the ISDN Q.931 bearer capability information element, the Cisco IOS software configures the DSP to treat the incoming call as a modem call, a V.110 call, or a V.120 call.

Figure 1 shows a dial-in scenario for how V.110 technology can be used with a stack of Cisco AS5300-series access servers.

Figure 1 V.110 Dial-In Scenario Using a Stack of Cisco AS5300-Series Access Servers



V.120 Bit Rate Adaptation Standard

ITU-T Recommendation V.120 revised by the ITU-T Study Group 14. V.120 describes a standard that can be used for adapting terminals with non-ISDN standard network interfaces to an ISDN. It is intended to be used between two terminal adapter (TA) functional groups, between two ISDN terminal (TE1) functional groups, between a TA and a TE1, or between either a TA or TE1 and an interworking facility inside a public or private ISDN.

V.120 allows for reliable transport of synchronous, asynchronous, or bit transparent data over ISDN bearer channels. Cisco provides three V.120 support features for terminal adapters that do not send the low-layer compatibility fields or bearer capability V.120 information:

- Answer all incoming calls as V.120—Static configuration used when all remote users have asynchronous terminals and need to connect with a vty on the router.
- Automatically detect V.120 encapsulation—Encapsulation dynamically detected and set.
- Enable V.120 support for asynchronous access over ISDN.

For terminal adapters that send the low-layer compatibility or bearer capability V.120 information, mixed V.120 and ISDN calls are supported. No special configuration is required.

V.120 is a digital rate adaptation and cannot be done on NM-AM network module analog modems. MICA DSP firmware does not have the code to terminate V.120 calls.

NextPort supports only a subset of V.120 functionalities that are supported by Cisco IOS software. Therefore, certain V.120 calls still will need to be terminated on the CPU, even if the chassis has available NextPort modems.

Managing Modems

To manage modems, perform the tasks in the following sections; the tasks you need to perform depend upon the type and needs of your system:

Managing SPE Firmware

You can upgrade your modem firmware to the latest NextPort Service Processing Element (SPE) firmware image available from Cisco. The SPE firmware image is usually retrieved from Cisco.com. You must first copy the SPE image from a TFTP server to flash memory using the **copy tftp flash** command. You then configure the firmware upgrade using the **firmware location** and **firmware upgrade** SPE configuration commands. The **firmware location** command specifies the location of the firmware file and downloads the firmware to an SPE or a range of SPEs, according to the schedule you selected for the firmware upgrade method using the **firmware upgrade** command.

The modem firmware upgrade commands must be saved into the system configuration using the **write memory** command; otherwise, at the next reboot downloading of the specified firmware will not occur.

To upgrade SPE firmware, use the following commands:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters global configuration mode.

Step 2	`AS5400:`
	`Router(config)# spe slot/spe`
	or
	`Router(config)# spe slot/spe slot/spe`
	
	`AS5800:`
	`Router(config)# spe shelf/slot/spe`
	or
	`Router(config)# spe shelf/slot/spe shelf/slot/spe`
		Enters SPE configuration mode. You can choose to configure a range of SPEs by specifying the first and last SPE in the range.

Step 3	`Router(config-spe)# firmware upgrade {busyout | download-maintenance | reboot}`	Specifies the upgrade method. Three methods of upgrade are available. The **busyout** keyword waits until all calls are terminated on an SPE before upgrading the SPE to the designated firmware. The **download-maintenance** keyword upgrades the firmware during the download maintenance time. The **reboot** keyword requests the access server to upgrade firmware at the next reboot.
Step 4	`Router(config-spe)# firmware location [IFS:[/]]filename`	Specifies the SPE firmware file in flash memory to use for the selected SPEs. Allows you to upgrade firmware for SPEs after the new SPE firmware image is copied to your flash memory. The Cisco IOS file specification (IFS) can be any valid IFS on any local file system. Use the **dir all-filesystems** EXEC command to display legal IFSs. Examples of legal IFS specifications include: • **bootflash:**—Loads the firmware from a separate flash memory device. • **flash:**—Loads the firmware from the flash NVRAM located within the router. • **system:/**—Loads the firmware from a built-in file within the Cisco IOS image. The optional forward slash (/) and system path must be entered with this specification. • *filename*—The name of the desired firmware file (for example, mica-modem-pw.2.7.3.0.bin). If the **system** keyword is specified, enter the path to the filename you want to download.
Step 5	`Router(config-spe)# exit`	Exits SPE configuration mode.
Step 6	`Router(config)# exit`	Exits global configuration mode.
Step 7	`Router# copy running-config startup-config`	Saves your changes.

Note As soon as a firmware file is specified, the downloading begins. Do not specify all modems and then go into an upgrade process on a busy router. The modems that are not busy will all be marked busy and the server will wait until all the modems on each of the given cards are free before upgrading the multiple-port cards. The only way to clear this situation is to start disconnecting users with a **clear** command. Normally, groups of modems are specified in scripts with the spe slot/spe_begin and slot/spe_end statements, and upgrades are done in a rolling fashion.

Use the **show modem version** and **show spe version** commands to verify that the modems are running the portware version you specified.

The following example shows how to enter the SPE configuration mode, set the range of SPEs, specify the firmware file location in flash memory, download the file to the SPEs, and display a status report using the **show spe** EXEC command:

```
Router# configure terminal
Router(config)# spe 7/0 7/17
Router(config-spe)# firmware upgrade busyout
Router(config-spe)# firmware location flash:np_6_75
Started downloading firmware flash:np_6_75.spe
Router(config-spe)# exit
Router(config)# exit
Router# show spe 7
.
.
.
                    SPE          SPE     SPE  SPE   Port         Call
SPE#    Port #      State        Busyout Shut Crash State        Type
7/00    0000-0005   ACTIVE             1    0     0 BBBBBB       _______
7/01    0006-0011   DOWNLOAD           1    0     0 bbbbbb       _______
7/02    0012-0017   DOWNLOAD           1    0     0 bbbbbb       _______
7/03    0018-0023   DOWNLOAD           1    0     0 bbbbbb       _______
.
.
.
```

For information about upgrading Cisco 3600 Series and Cisco 3700 modems, see the *Cisco 3600 Series and Cisco 3700 Series Modem Portware Upgrade Configuration Note* at the following URL: http://www.cisco.com/univercd/cc/td/doc/product/access/acs_mod/cis3600/sw_conf/portware/5257d56k.htm .

Configuring Modems in Cisco Access Servers

To configure modem support for access servers such as the Cisco AS5300 and AS5800, perform the following tasks. The list describes which tasks are required and which are optional but recommended.

- Configuring Modem Lines (Required)
- Verifying the Dial-In Connection (Optional but Recommended)
- Troubleshooting the Dial-In Connection (Optional but Recommended)
- Configuring the Modem Using a Modemcap (Required)
- Configuring the Modem Circuit Interface (Required for Digital Modems)

Configuring Modem Lines

You must configure the modem lines and set the country code to enable asynchronous connections into your access server. To configure the modems and line, use the following commands beginning in global configuration mode:

	Command	Purpose

Step 1	**MICA modems**
	`Router(config)# modem country mica country`
	NextPort SPE modems
	`Router(config)# spe country country`
	Microcom modems
	`Router(config)# modem country microcom_hdms country`
		Depending on the type of modems loaded in your access server, specifies the modem vendor and country code.[1] This step is only for the MICA, NextPort SPE, and Microcom modems in the Cisco AS5000 series access servers.
		Table 2 through Table 5 provide a summary and comparison of the Cisco IOS commands used for the MICA and NextPort modems.

Step 2	`Router(config)# line beginning-line-number ending-line-number`	Enters the number of modem lines to configure. Usually this range is equal to the number of modems in the access server. Use the **show line** EXEC command to see which lines are available.
Step 3	`Router(config-line)# transport {input | output} {all | none}`	Specifies that connection protocols can be used when connecting to the line. For outgoing calls, choose the **output** option. For incoming calls, choose the **input** option. If you do not intend to dial out, choose the **none** option.
Step 4	`Router(config-line)# autoselect {arap | ppp | slip}`	Configures the line to automatically startup an AppleTalk Remote Access (ARA), PPP, and Serial Line Internet Protocol (SLIP) session. You can configure more than one protocol by entering multiple **autoselect** commands with the appropriate keyword.
Step 5	`Router(config-line)# autoselect during-login`	Configures the lines to display the username and password prompt as soon as the line is connected, rather than waiting until the user presses the Enter or Return key at the terminal.

Step 6	`Router(config-line)# login authentication dialin`
	or
	`Router(config-line)# login login-name`
	`Router(config-line)# password password`
		Enables authentication across all asynchronous modem logins.
		Use the **login authentication dialin** command when authentication, authorization, and accounting (AAA) authentication has been enabled.
		Use the **login** and **password** commands to configure non-AAA user authentication.

Step 7	`Router(config-line)# modem dialin`	Configures the modem for only incoming calls.
Step 8	`Router(config-line)# exit`	Returns to global configuration mode.

1. For a comprehensive list of modem country codes, see the **modem country mica** command and the **modem country microcom_hdms** command in the *Cisco IOS Dial Technologies Command Reference*.

Verifying the Dial-In Connection

Before configuring any additional protocols for the line such as SLIP, PPP, or ARA, test whether the dial-in connection for the access server and modem are configured correctly for dial-in access,

Note The same configuration issues exist between the client DTE and client modem. Make sure that you have the correct EIA/TIA-232 cabling and modem initialization string for your client modem.

The following is an example of a successful connection from a PC using a known good modem to dial in to a Cisco access server:

```
at
OK
atdt9,5550101
CONNECT 14400/ARQ/V32/LAPM/V42BIS
User Access Verification
Username: user1
Password:
Router>
```

Troubleshooting the Dial-In Connection

Depending upon the problems you experience, take the appropriate action:

- If you are having problems making or receiving calls, make sure that you turned on the protocols for connecting to the lines and configured for incoming and outgoing calls.
- If the calls are not coming up at all, turn on modem debugging. Use the the modem debugging commands as follows:
  - The **debug modem** command enables debugging on the modem line.
  - The **debug modem csm** (or **debug csm modem)** command enables debugging for lines configured for digital modems.
  - The **debug isdn q931** command enables debugging for lines configured for the ISDN and Signaling System 7 (SS7) Q.931 protocols.
  - The **debug cas** command enables debugging for lines configured for channel-associated signaling (CAS).

  Following is a sample of how to enable and then disable Cisco IOS modem debugging commands on a network access server:

```
Router# debug modem
Router# debug modem csm
Router# debug isdn q931
Router# no debug modem
Router# no debug modem csm
Router# no debug isdn q931
```

- Enter the **debug modem ?** command for a list of additional modem debugging commands:

```
Router# debug modem ?
  b2b          Modem Special B2B
  csm          CSM activity
  maintenance  Modem maintenance activity
  mica         MICA Async driver debugging
  oob          Modem out of band activity
  tdm          B2B Modem/PRI TDM
```

```
trace        Call Trace Upload
```

- Turn off the messages by entering the **no debug modem** command.

For more detailed information refer to the TAC Tech Notes document, *Troubleshooting Modems*, at the following URL:
http://www.cisco.com/en/US/partner/tech/tk801/tk36/technologies_tech_note09186a0080094eb9.shtml#

Configuring the Modem Using a Modemcap

Modems are controlled by a series of parameter settings (up to a limit of 128 characters) that are sent to the modem to configure it to interact with a Cisco device in a specified way. The parameter settings are stored in a database called a *modem capability* (modemcap). The Cisco IOS software contains defined modemcaps that have been found to properly initialize internal modems. Following are the names of some modemcaps available in the Cisco IOS software:

- cisco_v110—Cisco (NEC) internal V.110 TA (AS5200)
- mica—Cisco MICA HMM/DMM internal digital modem
- nextport—Cisco NextPort CSMV/6 internal digital modem
- microcom_hdms—Microcom HDMS chassis
- microcom_mimic—Cisco (Microcom) internal analog modem (NM-AM–2600/3600)
- microcom_server—Cisco (Microcom) V.34/56K internal digital modem (AS5200)

Enter these modemcap names with the **modem autoconfigure type** command.

For more information on creating and using modemcaps refer to the TAC Tech Notes documentation, *Recommended Modemcaps for Internal Digital and Analog Modems on Cisco Access Servers*, at the following URL:
http://www.cisco.com/en/US/partner/tech/tk801/tk36/technologies_tech_note09186a008009491b.shtml

If your modem is not on this list and if you know what modem initialization string you need to use with it, you can create your own modemcap; see the following procedure, "Using the Modem Autoconfigure Type Modemcap Feature." To have the Cisco IOS determine what type of modem you have, use the **modem autoconfigure discovery** command to configure it, as described in the procedure "Using the Modem Autoconfigure Discovery Feature."

Note When configuring an internal modem, avoid using the Modem Autoconfigure Discovery feature because the feature can misdetect the internal modem type and cause the modem to start working in an unpredictable and unreproducable manner.

Using the Modem Autoconfigure Type Modemcap Feature

If you know what modem initialization string you need to use with your modem, you can create your own modemcap by performing the following steps.

Step 1 Use the **modemcap edit** command to define your own modemcap entry.

The following example defines modemcap MODEMCAPNAME:

```
Router(config)# modemcap edit MODEMCAPNAME miscellaneous &FS0=1&D3
```

Step 2 Apply the modemcap to the modem lines as shown in the following example:

```
Router# terminal monitor
```

```
Router# debug confmodem
Modem Configuration Database debugging is on
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#line 33 34
Router(config-line)#modem autoconfigure type MODEMCAPNAME
Jan 16 18:12:59.643: TTY34: detection speed (115200) response ---OK---
Jan 16 18:12:59.643: TTY34: Modem command:  --AT&FS0=1&D3--
Jan 16 18:12:59.659: TTY33: detection speed (115200) response ---OK---
Jan 16 18:12:59.659: TTY33: Modem command:  --AT&FS0=1&D3--
Jan 16 18:13:00.227: TTY34: Modem configuration succeeded
Jan 16 18:13:00.227: TTY34: Detected modem speed 115200
Jan 16 18:13:00.227: TTY34: Done with modem configuration
Jan 16 18:13:00.259: TTY33: Modem configuration succeeded
Jan 16 18:13:00.259: TTY33: Detected modem speed 115200
Jan 16 18:13:00.259: TTY33: Done with modem configuration
```

Note The report that is generated by the **debug confmodem** command can be misleading for the MICA and NextPort internal modems because these modems do not have Universal Asynchronous Receiver/Transmitter (UART) and exchange data with the CPU at speeds of hundreds of kbps.

Using the Modem Autoconfigure Discovery Feature

If you prefer that the modem software use its autoconfigure mechanism to configure the modem, use the **modem autoconfigure discovery** command.

The following example shows how to configure modem autoconfigure discovery mode:

```
Router# terminal monitor
Router# debug confmodem
Modem Configuration Database debugging is on
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# line 33 34
Router(config-line)# modem autoconfigure discovery
Jan 16 18:16:17.724: TTY33: detection speed (115200) response ---OK---
Jan 16 18:16:17.724: TTY33: Modem type is default
Jan 16 18:16:17.724: TTY33: Modem command:  --AT&F&C1&D2S0=1H0--
Jan 16 18:16:17.728: TTY34: detection speed (115200) response ---OK---
Jan 16 18:16:17.728: TTY34: Modem type is default
Jan 16 18:16:17.728: TTY34: Modem command:  --AT&F&C1&D2S0=1H0--
Jan 16 18:16:18.324: TTY33: Modem configuration succeeded
Jan 16 18:16:18.324: TTY33: Detected modem speed 115200
Jan 16 18:16:18.324: TTY33: Done with modem configuration
Jan 16 18:16:18.324: TTY34: Modem configuration succeeded
Jan 16 18:16:18.324: TTY34: Detected modem speed 115200
Jan 16 18:16:18.324: TTY34: Done with modem configuration
```

Configuring the Modem Circuit Interface

The next task to complete before using the integrated modem is to configure the modem circuit interface. The basic steps are outlined next:

- If the integrated modem is an analog modem, no further configuration is required; modem characteristics are set on the line.
- If the integrated modem is a digital modem, you can configure either the ISDN or CAS, as appropriate.

- For ISDN BRI and PRI, you need to select the switch type and whether ISDN accepts incoming voice or data calls. If you configure a PRI, you will need to configure the T1 or E1 controller.
- Configuring CAS is described in the chapter "Configuring ISDN PRI" in the Signaling Configuration part of this guide.

If you want to configure SS7, refer to Appendix G, "Configuring the Cisco SS7/C7 Dial Access Solution System," in the *Cisco IOS Voice, Video, and Fax Configuration Guide*.

Comparison of NextPort SPE and MICA Modem Commands

Table 2 through Table 5 compare the MICA and SPE commands.

Table 2 EXEC Commands: NextPort to MICA Command Comparison

NextPort SPE Commands	Purpose	MICA Modem Commands
clear port	Clears specified ports.	**clear modem**
clear port log	Clears all log entries for specified ports.	**clear modem log**
clear spe	Reboots all specified SPEs. All calls will be torn down.	none
clear spe counters	Clears all statistics.	**clear modem counters**
clear spe log	Clears all log entries for specified SPEs.	**clear modem log**
show port config	Displays configuration parameters for the current active session.	**show modem config**
show port modem calltracker	Displays port-level information for an active modem.	**show modem calltracker**
show port modem log	Displays the events generated by the modem sessions.	**show modem log**
show port modem test	Displays port modem test results.	**show modem test**
show port operational-status	Displays statistics for the current active session.	**show modem operational-status**
show spe	Displays the SPE status.	—
show spe log	Displays the SPE system log.	—
show spe modem active	Displays the statistics of all active calls on specified SPEs.	**show modem**
show spe modem csr	Displays the call success rate (CSR) for the specified SPE.	**show modem**
show spe modem disconnect-reason	Displays all modem disconnect reasons for the specified SPEs.	**show modem call-stats**
show spe modem high speed	Displays the total number of connections negotiated within each modulation or coder-decoder (codec) for a specific range of SPEs.	**show modem speed**
show spe modem high standard	Displays the total number of connections negotiated within each high modulation or codec for a specific range of SPEs or for all the SPEs.	—

Table 2 EXEC Commands: NextPort to MICA Command Comparison (continued)

NextPort SPE Commands	Purpose	MICA Modem Commands
show spe modem low speed	Displays the connect-speeds negotiated within each low-speed modulation or codec for a specific range of SPEs or for all the SPEs.	**show modem speed**
show spe modem low standard	Displays the total number of connections negotiated within each low modulation or codec for a specific range of SPEs or for all the SPEs.	—
show spe modem summary	Displays the modem service history statistics for specific SPEs.	**show modem**
show spe version	Displays all MICA and NextPort firmware versions stored in flash memory and the firmware assigned to each SPE.	**show modem mapping**

Table 3 SPE Configuration Commands: NextPort to MICA Command Comparison

NextPort SPE Commands	Purpose	MICA Modem Commands
busyout	Busies out active calls.	**modem busyout**
firmware location *filename*	Specifies the firmware file to be upgraded.	Already implemented on the Cisco AS5300 and Cisco AS5800 platforms.
firmware upgrade	Specifies the upgrade method.	Already implemented on the Cisco AS5300 platform.
port modem autotest[1]	Enables modem autotest.	**modem autotest**
shutdown	Tears down all active calls on the specified SPEs.	**modem shutdown**
spe	Configures the SPE.	Already implemented on the Cisco AS5300 and Cisco AS5800 platforms.
spe call-record	Generates a modem call record at the end of each call.	**modem call-record**
spe country	Sets the system country code.	**modem country**
spe log-size	Sets the maximum log entries for each port.	**modem buffer-size**
spe poll	Sets the statistic polling interval.	**modem poll**

1. Cisco does not recommend the use of the **modem autotest** or **port modem autotest** command. These commands may produce unexpected results including modems being marked out of service and unscheduled reloads. These commands have been removed in Cisco IOS Release 12.3.

Table 4 Port Configuration Commands: NextPort to MICA Command Comparison

NextPort SPE Commands	Purpose	MICA Modem Commands
busyout	Busies out a port.	**modem busyout**
default	Compares the value of the command to its default value.	**default modem**
port	Configures the port range.	**modem range**
shutdown	Shuts down a port.	**modem shutdown**

Table 5 Global Configuration Commands: NextPort to MICA Command Comparison

NextPort SPE CLI Commands	Purpose	MICA Modem CLI Commands
ds0 busyout-threshold	Defines a threshold to maintain a balance between the number of digital signal level 0s (DS0s) and modems.	**modem busyout-threshold**

Configuring Cisco Integrated Modems Using Modem Attention Commands

This section provides information about using modem attention (AT) command sets to modify modem configuration. It contains the following sections:

- Using Modem Dial Modifiers on Cisco MICA Modems (As required)
- Changing Configurations Manually in Integrated Microcom Modems (As required)
- Configuring Leased-Line Support for Analog Modems (As required)

Using Modem Dial Modifiers on Cisco MICA Modems

Dial modifiers permit multistage dialing for outbound modem calling through public and private switched telephone networks (PSTNs).

Note For additional information about dial modifiers for the MICA modems, search Cisco.com for the publication *AT Command Set and Register Summary for MICA Six-Port Modules*.

The Cisco NAS Modem Health feature is enabled by arguments to the **ATD AT** command. The **AT** prefix informs the network access server modem that commands are being sent to it, and the **D** (dial string or dial) suffix dials a telephone number, establishing a connection. With NAS Modem Health feature, you can enter the dial modifiers listed in Table 6 after the **D** in your dial string: **X**, **W**, and the comma (**,**) character. These modifiers had been previously accepted without error but ignored in Cisco MICA modems on Cisco AS5300 and Cisco AS5800 universal access servers.

Table 6 Dial Modifiers for Cisco MICA Modems

Dial Modifier	Definition
X	Switches to in-band dual tone multifrequency (DTMF) mode for any subsequent digits remaining in the **ATD** string. The **X** dial modifier has been added to serve as a delimiter for the host when the dial string is processed. It allows Cisco MICA portware to be used in many environments that do not support DTMF dialing (for example, PRI).
W	Waits for dial tone and then switches to in-band DTMF mode for any subsequent digits remaining in the **ATD** string. The **W** dial modifier also acts as a delimiter between the primary and secondary sections of the dial string, so that no additional **X** modifier is needed. Once either an **X** or a **W** has been parsed in the dial string, any additional **X** modifiers are ignored. Additional **W** modifiers cause Cisco MICA modems to wait for a dial tone.
,	Delay: Number of seconds in S8. Default is 2 seconds. The comma (**,**) dial modifier is treated as a silent DTMF tone for the duration of seconds specified in S8. The comma is acted on only after the call switching module (CSM) has made the transition to DTMF mode, which requires that it either follow an **X** or a **W** in the dial string, or that the T1/E1 be configured for DTMF signaling.

In the following example dial string, the portion of the string before the **X** is dialed for the given line type used in your configuration. All digits after the **X** generate the appropriate DTMF tones.

```
atdT5550101x,,567
```

Changing Configurations Manually in Integrated Microcom Modems

You can change the running configuration of an integrated modem by sending individual modem AT commands. Manageable Microcom modems have an out-of-band feature, which is used to poll modem statistics and send AT commands. The Cisco IOS software uses a direct connect session to transfer information through this out-of-band feature. To send AT commands to a Microcom modem, you must permit a direct connect session for a specified modem, open a direct connect session, send AT commands to a modem, and clear the directly connected session from the modem when you are finished.

Open a direct connect session by entering the **modem at-mode** *slot/port* command in privileged EXEC mode. From here, you can send AT commands directly from your terminal session window to the internal Microcom modems. Most incoming or outgoing calls on the modems are not interrupted when you open a direct connect session and send AT commands. However, some AT commands interrupt a call—for example, the **ATH** command, which hangs up a call. Open and close one direct connect session at a time. Note that multiple open sessions slow down modem performance.

Refer to the AT command set that came with your router for a complete list of AT commands that you can send to the modems.

For Microcom modems, you can clear or terminate an active directly connected session in two ways:

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

The following example illustrates use of the modem commands.

AT Mode Example for Integrated Modems

To establish a direct connect session to an internal or integrated modem (existing inside the router), such as the connection required for Microcom modems in the Cisco AS5200 access server, open a directly connected session with the **modem at-mode** command and then send an AT command to the specified modem. For example, the following example sends the AT command **at%v** to 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:       FFFFFFFFFFFF
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     ATS0=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 modem responds with “OK” when the AT command you send is received.

Configuring Leased-Line Support for Analog Modems

Analog modems on the NM-8AM and NM-16AM network modules in the Cisco 2600 and 3600 series routers provide two-wire leased-line support for enterprise customers who require point-to-point connections between locations and for enterprise customers with medium to high data transfer requirements without access to other technologies or with access to only low-grade phone lines.

This feature works only with leased lines that provide loop current. Each modem used must have an RJ-11 connection to the PSTN.

Several features enhance the analog modem software:

- 2-wire leased-line support.
- Modem speeds up to 33.6 kbps with support for all current analog modem protocols, compression, and error correction techniques.
- Power-on autoconnect and loopback testing.
- Support for the maximum number of leased-line users without data transmission loss at distances up to 2 to 5 km.
- In-band and out-of-band monitoring.

- Support on all Cisco 2600 and Cisco 3600 series platforms and upgradability using Cisco IOS software.
- Compatibility with other major leased-line modem vendors.

To configure this support, configure one modem AT command (**AT&L)** and two AT registers with the **modemcap entry** command for the appropriate leased lines.

For leased line configuration using the **AT&L**{**0** | **1** | **2**}command:

- **0**—Disables the leased line (enables switched line; default).
- **1**—Enables the leased line. The modem initiates a leased line when dial and answer commands (**ATD** and **ATA**) are issued.
- **2**—Enables the leased line. The modem goes off hook automatically after T57 number of seconds in:
  - Originate mode if ATS0 is 0.
  - Answer mode if ATS0 is not equal to 0.

The following AT registers can also be set:

- **AT:T57**—Number of seconds before going off hook in leased-line mode when the command **AT&L2** is used (defaults to 6).
- **AT:T79**—Number of autoretrains before the modem is disconnected (defaults to 3).

For more information about using the AT command set with the modems on the NM-8AM and NM-16AM network modules in the Cisco 2600 and 3600 series routers, search Cisco.com for the publication *AT Command Set and Register Summary for Analog Modem Network Modules.*

To configure a modem for leased-line operation, use the following commands in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# modemcap entry modem-type-name:AA=S0=0&L2`	Sets the modemcap for leased-line operation for the originating modem.
Step 2	`Router(config)# modemcap entry modem-type-name:AA=S0=1&L2`	Sets the modemcap for leased-line operation for the answering modem.

The **show modemcap** command lists all the predefined modem types and any user-defined modemcaps that are currently configured on the router:

- If the leased line has been configured, the modemcap information will be available.
- If the leased line has not been configured, only the predefined modem types will be displayed.

The important setting for leased-line support is what is defined in the modemcap as the key configuration item and its application to the leased line. Consider the following command strings:

```
modemcap entry micro_LL_orig:AA=S0=0&L2
modemcap entry micro_LL_ans:AA=S0=1&L2
```

AA stands for autoanswer:

- The answering modem AA register is set to 1 (AA=S0=1) so that autoanswer is “on”.
- The originating modem AA register is set to 0 (AA=S0=0) so that autoanswer is “off”.

If the AA feature is used, both the originating and answering modem must be put into leased-line mode with the **&L2** AT command.

In the examples, the micro_LL_orig and micro_LL_ans strings are arbitrary text descriptions.

Note For the **modemcap entry** command, one of the predefined modem types may be used or a completely user-defined modemcap may be created. For leased line, no new modem type was added. Users may create their own modemcaps for leased-line functionality.

To configure the modem for leased-line operation, use the **modemcap entry** command. For each connection, each modem must be configured as an originator or answerer.

The following example shows modemcaps for a leased-line originator and answerer and their application to specific ports:

```
modemcap entry micro_LL_orig:AA=S0=0&L2
modemcap entry micro_LL_ans:AA=S0=1&L2
line 73
 no exec
 modem InOut
 modem autoconfigure type micro_LL_ans
 transport input all
line 74
 no exec
 modem InOut
 modem autoconfigure type micro_LL_orig
 transport input all
```

Note When Multilink PPP (MLP) is configured on a dialer interface, the dialer configuration has a default value of 2 minutes for dialer idle timeout. For leased-line connections, set the dialer idle timeout to infinity by adding **dialer idle-timeout 0** to the configuration.

Verifying the Analog Leased-Line Configuration

The following information is important for verifying or troubleshooting your configuration. The **show modem log** command displays the progress of leased-line connections. Here is an example log for a leased-line answerer. Note the “LL Answering” state and “LL Answer” in the “Direction” field of the connection report:

```
00:44:03.884 DTR set high
00:44:02.888 Modem enabled
00:43:57.732 Modem disabled
00:43:52.476 Modem State:LL Answering
00:43:52.476 CSM:event-MODEM_STARTING_CONNECT New
State-CSM_CONNECT_INITIATED_STATE
00:43:51.112 Modem State:Waiting for Carrier
00:43:43.308 Modem State:Connected
00:43:42.304 Connection:TX/RX Speed = 33600/33600,
Modulation = V34
Direction = LL Answer, Protocol = MNP, Compression =
V42bis
00:43:42.304 CSM:event-MODEM_CONNECTED New
State-CONNECTED_STATE
00:43:42.300 RS232:noCTS* DSR* DCD* noRI noRxBREAK
TxBREAK*
00:43:41.892 PPP mode active
00:43:41.892 Modem enabled
00:43:39.888 PPP escape maps set:TX map=00000000 RX
map=FFFFFFFF
00:43:39.724 PPP escape maps set:TX map=00000000 RX
map=000A0000
00:43:34.444 RS232:CTS* DSR DCD noRI noRxBREAK TxBREAK
00:43:11.716 Modem Analog Report:TX = -20, RX = -34,
```

```
Signal to noise = 61
```

Cisco 2600 and 3600 Series Analog Modem Leased-Line Support Examples

In the following examples, one Cisco 3620 router and one Cisco 3640 router are connected back-to-back using leased lines. The Cisco 3620 router has the originating configuration, and the Cisco 3640 router has the answering configuration.

In the dialer interface configuration, the **dialer idle-timeout 0** command is added to set the dialer idle timeout to be infinity. Otherwise the leased line will go down and up every 2 minutes because the default dialer interface idle timeout is 2 minutes.

Note Except for passwords and logins, the Cisco IOS command-line interface (CLI) is case-insensitive. For this document, an uppercase “L” has been used in the command examples to avoid confusion with the numeral “1”.

Leased-Line Originating Configuration

```
version 12.1
service timestamps debug uptime
service timestamps log uptime
!
modemcap entry micro_LL_orig:AA=S0=0&L2
modemcap entry micro_LL_ans:AA=S0=1&L2
!
interface Async33
  no ip address
encapsulation ppp
 no ip route-cache
 no ip mroute-cache
 dialer in-band
 dialer pool-member 1
 async default routing
 async dynamic routing
 async mode dedicated
 no peer default ip address
 no fair-queue
 no cdp enable
 ppp direction callout
 ppp multilink
!
interface Dialer1
 ip address 10.1.24.1 255.255.255.0
 encapsulation ppp
 no ip route-cache
 no ip mroute-cache
 dialer remote-name sara40
 dialer pool 1
 dialer idle-timeout 0
 dialer max-call 4096
 no cdp enable
 ppp direction callout
 ppp multilink
!
dialer-list 1 protocol ip permit
!
line con 0
 exec-timeout 0 0
 transport input none
 line 33
```

```
 no exec
 modem InOut
 modem autoconfigure type micro_LL_orig
 transport input all
line aux 0
 exec-timeout 0 0
line vty 0 4
 exec-timeout 0 0
!
end
```

Leased-Line Answering Configuration

```
version 12.1
service timestamps debug uptime
service timestamps log uptime
!
modemcap entry micro_LL_orig:AA=S0=0&L2
modemcap entry micro_LL_ans:AA=S0=1&L2
!
interface Async73
 no ip address
 encapsulation ppp
 no ip route-cache
 no ip mroute-cache
 dialer in-band
 dialer pool-member 1
 async default routing
 async dynamic routing
 async mode dedicated
 no peer default ip address
 no fair-queue
 no cdp enable
 ppp direction callout
 ppp multilink
!
interface Dialer1
 ip address 10.1.24.2 255.255.255.0
 encapsulation ppp
 no ip route-cache
 no ip mroute-cache
 load-interval 30
 dialer remote-name sara20
 dialer pool 1
 dialer idle-timeout 0
 dialer load-threshold 1 either
 dialer max-call 4096
 no cdp enable
 ppp direction callout
 ppp multilink
!
dialer-list 1 protocol ip permit
line con 0
 exec-timeout 0 0
 transport input none
line 73
 no exec
 modem InOut
 modem autoconfigure type micro_LL_ans
 transport input all
line aux 0
 transport input all
 flowcontrol hardware
line vty 0 4
```

```
 exec-timeout 0 0
!
end
```

Configuring Modem Pooling

Modem pooling allows you to control which modem a call connects to, on the basis of dialed number identification service (DNIS). When modem pooling is not used, incoming and outgoing calls are arbitrarily assigned to modems. For example, consider a Cisco AS5300 access server loaded with a 4-port ISDN PRI card. After an analog modem call comes into the first PRI trunk, the call is greeted by a general pool of B channels and a general pool of modems. Any B channel can be connected to any modem in the access server. A random assignment takes place. Modem resources cannot be controlled.

Modem pooling assigns physical modems to a single DNIS. It enables you to create pools of physical modems in one access server, assign a unique DNIS to each modem pool, and set maximum simultaneous connect limits.

This feature is used for physically partitioning or virtually partitioning modems inside one network access server.

Modem pooling offers these benefits:

- A certain number of modem ports can be guaranteed per DNIS.
- Maximum simultaneous connection limits can be set for each DNIS.

The following restrictions apply:

- Modem pooling is not a solution for large-scale dial access. It cannot be used to create virtual modem pools across multiple access servers that are connected. Modem pooling is physically restricted to one access server.
- MICA and Microcom technology modems support modem pooling. However, only MICA modems support modem pooling for CT1 and CE1 configurations using CAS. To use modem pooling with CT1 or CE1 connections, you must reserve at least two modems in the default modem pool. These reserved modems decode DNIS before handing off calls to the modems assigned to modem pools.

  If you see many call failures appearing on the access server, try assigning more modems to the default pool. Use the **show modem** and **show modem summary** EXEC commands to display the modem call failure and success ratio.
- No MIBs support modem pooling.
- The same DNIS cannot exist in more than one modem pool.

Modem pooling is supported on the Cisco AS5300 access servers. To configure and manage modems, perform the tasks in the following sections; all tasks are optional and depend upon the needs of your system.

- Creating a Modem Pool (Required)
- Verifying Modem Pool Configuration (As required)

Creating a Modem Pool

You must first decide to physically partition or virtually partition your modems. For more information, see the previous section, "Configuring Modem Pooling." After you have made this decision, create a modem pool for a dial-in service or specific customer by using the following commands beginning in global configuration mode.

	Command	Purpose
Step 1	`Router(config)# modem-pool name`	Creates a modem pool and assigns it a name, and starts modem pool configuration mode.
Step 2	`Router(config-modem-pool)# pool-range number-number`	Assigns a range of modems to the pool. A hyphen (-) is required between the two numbers. The range of modems you can choose from is equivalent to the number of modems in your access server that are not currently associated with another modem pool.
Step 3	`Router(config-modem-pool)# called-number number [max-conn number]`	Assigns the DNIS to be used for this modem pool. The **max-conn** option specifies the maximum number of simultaneous connections allowed for this DNIS. If you do not specify a **max-conn** value, the default (total number of modems in the pool) is used.[1]
Step 4	`Router(config-modem-pool)# Ctrl-Z`	Returns to EXEC mode.
Step 5	`Router# show configuration`	Displays the running configuration to verify the modem pool settings. Make changes accordingly.
Step 6	`Router# copy running-config startup-config`	Saves the running configuration to the startup configuration.

1. The DNIS string can have an integer x to indicate a “don’t care” digit for that position, for example, 555010x.

Note If you have active modem calls on the access server before using modem pooling, modem pooling gracefully applies itself to the access server. Modem pooling first waits for active calls to hang up before assigning modems to modem pools and directing calls according to DNIS.

Verifying Modem Pool Configuration

To verify the modem configuration, enter the **show modem-pool** command to display the configuration. This command displays the structure and activity status for all the modem pools in the access server. See Table 7 for a description of each display field.

```
Router# show modem-pool

modem-pool: System-def-Mpool
modems in pool: 0   active conn: 0
0 no free modems in pool

modem-pool: v90service
modems in pool: 48  active conn: 46
 8 no free modems in pool
 called_party_number: 1234
   max conn allowed: 48, active conn: 46
   8 max-conn exceeded, 8 no free modems in pool

modem-pool: v34service
modems in pool: 48  active conn: 35
 0 no free modems in pool
 called_party_number: 5678
   max conn allowed: 48, active conn: 35
   0 max-conn exceeded, 0 no free modems in pool
```

Table 7 show modem-pool Field Descriptions

Field	Description
modem-pool	Name of the modem pool. In the previous example, there are three modem pools configured: System-def-Mpool, v90service, and v34service. To set the modem pool name, refer to the **modem-pool** command. All the modems not assigned to a modem pool are automatically assigned to the system default pool (displayed as System-def-Mpool).
modems in pool	Number of modems assigned to the modem pool. To assign modems to a pool, refer to the display and descriptions for the **pool-range** command.
active conn	Number of simultaneous active connections for the specified modem pool or called party DNIS number.
no free modems in pool	Number of times incoming calls were rejected because there were no more free modems in the pool to accept the call.
called_party_number	Specified called party DNIS number. This is the number that the remote clients use to dial in to the access server. You can have more than one DNIS number per modem pool. To set the DNIS number, refer to the description for the **called-number** command.
max conn allowed	Maximum number of modems that a called party DNIS number can use, which is an overflow protection measure. To set this feature, refer to the description for the **called-number** command.
max-conn exceeded	Number of times an incoming call using this called party DNIS number was rejected because the **max-conn** *number* parameter specified by the **called-number** command was exceeded.

For modem pool configuration examples, see the section "Physical Partitioning with Dial-In and Dial-Out Scenario" later in this chapter.

Check the following if you are having trouble operating your modem:

- Make sure you have not configured the same DNIS for multiple pools.
- Make sure you have not placed the same modem in multiple pools.

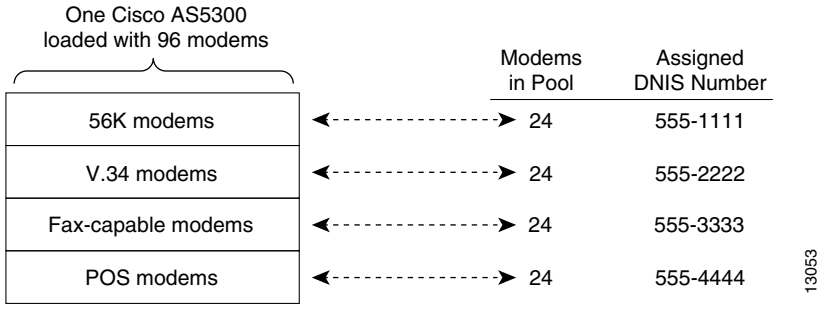
Note Modem pools that use MICA or Microcom modems support incoming analog calls over ISDN PRI. However, only MICA modems support modem pooling for T1 and E1 configurations with CAS.

Configuring Physical Partitioning

You can either physically partition or virtually partition your modems to enable different dial-in and dial-out services. This section provides information about the following optional tasks:

Physical partitioning uses one access server to function as multiple access servers loaded with different types of modem services (for example, V.34 modems, fax-capable modems, and point-of-sale (POS) modems). Each modem service is part of one physical modem pool and is assigned a unique DNIS number. (See Figure 2.)

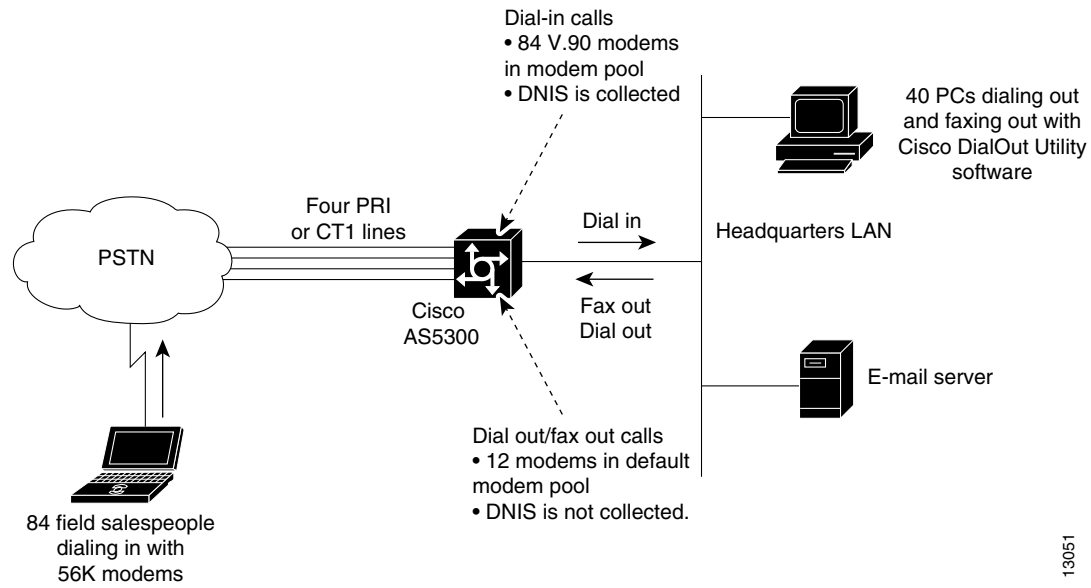
Figure 2 Modem Pooling Using Physical Partitioning



Physical partitioning can also be used to set up an access server for bidirectional dial access. (See Figure 3.)

Figure 3 shows one Cisco AS5300 access server loaded with 96 MICA modems and configured with 2 modem pools. One modem pool has 84 modems and collects DNIS. This pool is shared by 400 salespeople who remotely download e-mail from headquarters. The other modem pool contains 12 fax-capable modems and does not collect DNIS. This pool is shared by 40 employees using PCs on a LAN. Each time an outbound call is initiated by a PC, a modem on the Cisco AS5300 access server is seized and used to fax out or dial out. Not configuring DNIS support in the fax-out modem pool protects the pool from being used by the calls coming in from the field. Regardless of how many salespeople are dialing in or which telephone number they use, the fax-out and dial-out modem pool will always be reserved for the PCs connected to the LAN.

Figure 3 Modem Pooling Used for Bidirectional Dialing



Creating a Physical Partition

The following task creates one V.34 modem pool and one 56K modem pool on a Cisco AS5200. Each modem pool is configured with its own DNIS. Depending on which DNIS the remote clients dial, they connect to a 56K MICA modem or a V.34 Microcom modem.

The following hardware configuration is used on the Cisco AS5200 access server:

- One 2-port T1 PRI card
- One 48-port card containing four 6-port MICA 56K modem modules and two 12-port Microcom V.34 modem modules

To configure basic physical partitioning, perform the following steps:

Step 1 Enter global configuration mode:

```
Router# configure terminal
Router(config)#
```

Step 2 Create the modem pool for the 56K MICA modem services using the **modem-pool** *name* command. The modem pool is called 56kservices, which spans four 6-port MICA 56K modem modules.

```
Router(config)# modem-pool 56kservices
Router(config-modem-pool)#
```

Note The router is in modem pool configuration mode after the prompt changes from Router(config)# to Router(config-modem-pool)#.

Step 3 Assign a range of modems to the modem pool using the **pool-range** *number-number* command. Because all the 56K MICA technologies modems are seated in slot 1, they are assigned TTY line numbers 1 to 24. Use the **show line** EXEC command to determine the TTY line numbering scheme for your access server.

```
Router(config-modem-pool)# pool-range 1-24
```

Step 4 Assign a DNIS to the modem pool using the **called-number** *number* [**max-conn** *number*] command. This example uses the DNIS 5550101 to connect to the 56K modems. The maximum simultaneous connection limit is set to 24. The 25th user who dials 5550101 gets a busy signal.

```
Router(config-modem-pool)# called-number 5550101 max-conn 24
```

Step 5 Return to EXEC mode by entering **Ctrl-Z**. Next, display the modem pool configuration using the **show modem-pool** command. In the following example, 56K modems are in the modem pool called 56kservices. The remaining 24 V.34 Microcom modems are still in the default system pool.

```
Router(config-modem-pool)# ^Z
Router# show modem-pool

modem-pool: System-def-Mpool
modems in pool: 24   active conn: 0
0 no free modems in pool

modem-pool: 56kservices
modems in pool: 24  active conn: 0
 0 no free modems in pool
 called_party_number: 5550101
   max conn allowed: 24, active conn: 0
   0 max-conn exceeded, 0 no free modems in pool
```

Step 6 Create the modem pool for the Microcom physical partition. After the configuration is complete, the **show modem-pool** command shows that there are no remaining modems in the system default modem pool.

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# modem-pool v34services
Router(config-modem-pool)# pool-range 25-48
Router(config-modem-pool)# called-number 5550202 max-conn 24
Router(config-modem-pool)# ^Z
Router# show modem-pool

modem-pool: System-def-Mpool
modems in pool: 0   active conn: 0
0 no free modems in pool

modem-pool: 56kservices
modems in pool: 48  active conn: 0
 0 no free modems in pool
 called_party_number: 5550101
 max conn allowed: 48, active conn: 0
 0 max-conn exceeded, 0 no free modems in pool

modem-pool: v34services
modems in pool: 48  active conn: 0
 0 no free modems in pool
 called_party_number: 5550202
 max conn allowed: 48, active conn: 0
 0 max-conn exceeded, 0 no free modems in pool

Router# copy running-config startup-config
```

Physical Partitioning with Dial-In and Dial-Out Scenario

The following is a bidirectional dial scenario using a Cisco AS5300 access server. Two modem pools are configured. One modem pool contains 84 56K MICA modems, which is shared by 400 remote salespeople who dial in to headquarters. The other modem pool contains 12 fax-capable modems, which are shared by 40 employees who dial out of the headquarters LAN using the Cisco DialOut Utility software. See Figure 3 for the network topology.

The following hardware configuration is used on the Cisco AS5300:

- One 4-port T1 PRI card
- Two 48-port cards containing fourteen 6-port MICA 56K modem modules and two 6-port MICA fax-capable modem modules

To configure physical partitioning with dial-in and dial-out capability, perform the following steps:

Step 1 Create the 56K modem pool for the 400 remote salespeople. This modem pool contains 84 modems, which are reserved for the dial-in calls. To get access, the salespeople dial the DNIS 5550303. The total number of simultaneous calls is limited to 84. The 85th call and those above it are rejected. The **modem dialin** line configuration command is used to prevent modems 1 to 84 from dialing out.

```
Router# configure terminal
Router(config)# modem-pool 56ksalesfolks
Router(config-modem-pool)# pool-range 1-84
Router(config-modem-pool)# called-number 5550303 max-conn 84
Router(config-modem-pool)# exit
Router(config)# line 1 84
```

```
Router(config-line)# modem dialin
Router(config-line)# transport input all
Router(config-line)# rotary 1
Router(config-line)# autoselect ppp
Router(config-line)# exit
Router(config)#
```

Step 2 Create the dial-out/fax-out modem pool for the 40 local employees connected to the headquarters LAN. This modem pool contains 12 fax-capable MICA modems. No DNIS is assigned to the pool. Because lines 85 to 96 are used for the dial-out and fax-out modem services, the asynchronous lines are configured for reverse Telnet. This configuration is needed for the Telnet extensions to work with the dial-out application, which is installed on the LAN PCs.

```
Router(config)# modem-pool dialoutfolks
Router(config-modem-pool)# pool-range 85-96
Router(config-modem-pool)# exit
Router(config)# line 85-96
Router(config-line)# refuse-message z [!NMM!] No Modems Available z
Router(config-line)# exec-timeout 0 0
Router(config-line)# autoselect during-login
Router(config-line)# autoselect ppp
Router(config-line)# modem inout
Router(config-line)# rotary 1
Router(config-line)# transport preferred telnet
Router(config-line)# transport input all
Router(config-line)# exit
Router(config)#
```

Step 3 Configure the group asynchronous interface, which assigns core protocol characteristics to all the asynchronous interfaces in the system. Regardless of the direction that the modems are dialing, all modems in the access server leverage this group asynchronous configuration.

```
Router(config)# interface group-async 1
Router(config-if)# ip unnumbered ethernet 0
Router(config-if)# encapsulation ppp
Router(config-if)# async mode interactive
Router(config-if)# ppp authentication chap pap paplocal
Router(config-if)# peer default ip address pool bidir_dial_pool
Router(config-if)# no cdp enable
Router(config-if)# no ip mroute cache
Router(config-if)# no ip route cache
Router(config-if)# async dynamic routing
Router(config-if)# async dynamic address
Router(config-if)# group range 1-96
Building configuration...
Router(config-if)# exit
```

Step 4 Create an IP address pool for all the dial-in clients and dial-out clients. Both types of clients borrow addresses from this shared pool.

```
Router(config)# ip local pool bidir_dial_pool 10.4.1.1 10.4.1.96
Router(config)# ^z
Router# copy running-config startup-config
```

Step 5 (Optional) If you are using CiscoSecure AAA and a remote TACACS server, include the following security statements on the access server:

```
Router(config)# aaa new-model
Router(config)# aaa authentication login default tacacs+
Router(config)# aaa authentication login noaaa local
Router(config)# aaa authentication login logintac tacacs+
Router(config)# aaa authentication ppp ppptac tacacs+
Router(config)# aaa authentication ppp paplocal local
```

```
Router(config)# aaa authorization exec tacacs+
Router(config)# aaa authorization network tacacs+
Router(config)# aaa authorization reverse-access tacacs+
Router(config)# aaa accounting exec start-stop tacacs+
Router(config)# aaa accounting network start-stop tacacs+
Router(config)# aaa accounting update newinfo
Router(config)# enable password cisco
```

You should also include the host name, timeout interval, and authentication key:

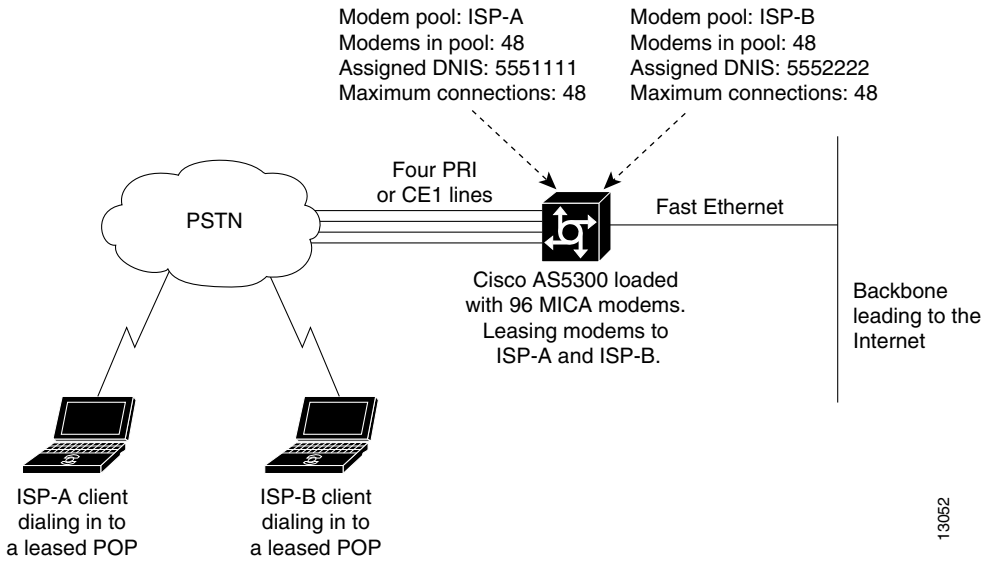
```
Router(config)# tacacs-server host 10.4.1.10
Router(config)# tacacs-server timeout 20
Router(config)# tacacs-server key nas1
```

Configuring Virtual Partitioning

Virtual partitioning creates one large modem pool on one access server, but assigns different DNIS numbers to different customers. Each incoming DNIS consumes resources from the same modem pool, but a maximum connect option is set for each DNIS.

Figure 4 shows two Internet service provider (ISP) customers who are leasing modems from another service provider. Each ISP is assigned its own DNIS number and range of modems. Each ISP is guaranteed a certain number of physical modem ports for simultaneous connections. After an ISP uses up all the modems assigned to its DNIS, a busy signal is issued.

Figure 4 Modem Pooling Using Virtual Partitioning



Virtual partitioning essentially resells modem banks to customers, such as a small-sized ISP. However, remember that modem pooling is a single-chassis solution, not a multichassis solution. Modem pooling is not a solution for reselling ports on a large-scale basis.

The following procedure creates one modem pool on a Cisco AS5300 access server for two ISP customers. The shared modem pool is called isp56kpool. However, both ISP customers are assigned different DNIS numbers and are limited to a maximum number of simultaneous connections.

See Figure 4 for the network topology.

The following hardware configuration is used on the Cisco AS5300 access server:

- One 4-port T1 PRI card
- Two 48-port cards containing sixteen 6-port MICA 56K modem modules

To configure virtual partitioning, perform the following steps:

Step 1 Enter global configuration mode:

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

Step 2 Create the shared modem pool for the 56K MICA modem services. This modem pool is called isp56kpool, which spans sixteen 6-port MICA 56K modem modules.

```
Router(config)# modem-pool isp56kpool
Router(config-modem-pool)#
```

Step 3 Assign all the modems to the modem pool using the **pool-range** *number-number* command. Use the **show line** EXEC command to determine your TTY line numbering scheme.

```
Router(config-modem-pool)# pool-range 1-96
```

Step 4 Assign a unique DNIS to each ISP customer using the **called-number** *number* [**max-conn** *number*] command. In this example, the **max-conn** *number* option limits each ISP to 48 simultaneous connections. The 49th user to dial either DNIS will get a busy signal.

```
Router(config-modem-pool)# called-number 5550101 max-conn 48
Router(config-modem-pool)# called-number 5550202 max-conn 48
```

Step 5 Return to EXEC mode by entering a **Ctrl-Z** sequence. Next, display the modem pool configuration using the **show modem-pool** command. In the following example, all the 56K modems are in the isp56kpool modem pool. The output also shows two DNIS numbers configured: 5550101 and 5550202.

```
Router(config-modem-pool)# ^Z
Router# show modem-pool
modem-pool: System-def-Mpool
modems in pool: 0   active conn: 0
0 no free modems in pool

modem-pool: isp56kpool
modems in pool: 96  active conn: 0
 0 no free modems in pool
 called_party_number: 5550101
   max conn allowed: 48, active conn: 0
   0 max-conn exceeded, 0 no free modems in pool
 called_party_number: 5550202
   max conn allowed: 48, active conn: 0
   0 max-conn exceeded, 0 no free modems in pool

Router# copy running-config startup-config
```

Configuring Call Tracker

The Call Tracker feature captures detailed statistics on the status and progress of active calls and retains historical data for disconnected call sessions. Call Tracker collects session information such as call states and resources, traffic statistics, total bytes transmitted and received, user IP address, and disconnect reason. This data is maintained within the Call Tracker database tables, which are accessible through the Simple Network Management Protocol (SNMP), the CLI, or syslog.

Note The calltracker command, providing Call Tracker services, is supported for dial calls but not voice. Calltracker is supported for dial calls on 5x platforms (5300, 5350, 5400, 5800, and 5850).

Call Tracker is notified of applicable call events by related subsystems such as ISDN, PPP, CSM, Modem, EXEC, or TCP-Clear. SNMP traps are generated at the start of each call, when an entry is created in the active table, and at the end of each call, when an entry is created in the history table. Call Record syslogs are available through configuration that will generate detailed information records for all call terminations. This information can be sent to syslog servers for permanent storage and future analysis.

Additionally, the status and diagnostic data that is routinely collected from MICA modems is expanded to include new link statistics for active calls, such as the attempted transmit and receive rates, the maximum and minimum transmit and receive rates, and locally and remotely issued retrains and speedshift counters. For more detailed information on Call Tracker logs, refer to the TAC Tech Notes document, *Understanding Call Tracker Outputs*, at the following URL: http://www.cisco.com/warp/public/471/calltracker_view.html

To configure Call Tracker, perform the following steps:

	Command	Purpose
Step 1	`Router(config)# calltracker enable`	Enables Call Tracker.
Step 2	`Router(config)# calltracker call-record {terse|verbose} [quiet]`	Enables Call Tracker syslog support for generating detailed Call Records.
Step 3	`Router(config)# calltracker history max-size number`	Sets the maximum number of call entries to store in the Call Tracker history table.
Step 4	`Router(config)# calltracker history retain-mins minutes`	Sets the number of minutes for which calls are stored in the Call Tracker history table.
Step 5	`Router(config)# snmp-server packetsize byte-count`	Sets the maximum packet size allowed for SNMP server requests and replies.
Step 6	`Router(config)# snmp-server queue-length length`	Sets the queue length for SNMP traps.
Step 7	`Router(config)# snmp-server enable traps calltracker`	Enables Call Tracker to send traps whenever a call starts or ends.
Step 8	`Router(config)# snmp-server host host community-string calltracker`	Specifies the name or Internet address of the host to send Call Tracker traps.

Verifying Call Tracker

To verify the operation of Call Tracker, use the the following command in EXEC mode:

Command	Purpose
Router# **show call calltracker summary**	Verifies the Call Tracker configuration and current status.

Enabling Call Tracker

The following example shows how to enable the Call Tracker feature:

```
calltracker enable
calltracker call-record terse
calltracker history max-size 50
calltracker history retain-mins 5000
!
snmp-server engineID local 0012345
snmp-server community public RW
snmp-server community private RW
snmp-server community wxyz123 view v1default RO
snmp-server trap-source FastEthernet0
snmp-server packetsize 17940
snmp-server queue-length 200
snmp-server location SanJose
snmp-server contact Bob
snmp-server enable traps snmp
snmp-server enable traps calltracker
snmp-server enable traps isdn call-information
snmp-server enable traps hsrp
snmp-server enable traps config
snmp-server enable traps entity
snmp-server enable traps envmon
snmp-server enable traps bgp
snmp-server enable traps ipmulticast-heartbeat
snmp-server enable traps rsvp
snmp-server enable traps frame-relay
snmp-server enable traps rtr
snmp-server enable traps syslog
snmp-server enable traps dlsw
snmp-server enable traps dial
snmp-server enable traps dsp card-status
snmp-server enable traps voice poor-qov
snmp-server host 10.255.255.255 wxyz123
snmp-server host 10.0.0.0 xxxyyy calltracker
!
radius-server host 172.16.0.0 auth-port 1645 acct-port 1646 non-standard
radius-server key xyz
!
```

Configuring Polling of Link Statistics on MICA Modems

The status and diagnostic data that is routinely collected from MICA modems is expanded to include new link statistics for active calls, such as the attempted transmit and receive rates, the maximum and minimum transmit and receive rates, and locally and remotely issued retrains and speedshift counters. This connection data is polled from the modem at user-defined intervals and passed to Call Tracker.

To poll modem link statistics, use the following command in global configuration mode:

Command	Purpose
`Router(config)# modem link-info poll time seconds`	Sets the polling interval at which link statistics for active calls are retrieved from the modem.

Note The **modem link-info poll time** command consumes a substantial amount of memory, approximately 500 bytes for each MICA modem call. Use this command only if you require the specific data that it collects; for instance, if you have enabled Call Tracker on your access server.

Configuring MICA In-Band Framing Mode Control Messages

Dial-in Internet connections typically start in character mode to allow the user to log in and select a preferred service. When Cisco IOS software determines that the user wants a framed interface protocol during the call, such as PPP or SLIP, commands are sent to the MICA modem so that it will provide hardware assistance with the framing. This hardware assistance reduces the Cisco IOS processing load. To avoid loss or misinterpretation of framed data during the transition, issue these commands at precise times with respect to the data being sent and received.

MICA modem framing commands can be sent in the data stream itself, which greatly simplifies Cisco IOS tasks in achieving precision timing. For PPP connections, the common way for modems to connect to the Internet, total connect time might typically be improved by 2 to 3 seconds. This functionality reduces timeouts during PPP startup and reduces startup time. If an ASCII banner is sent just before PPP startup, this feature eliminates problems with banner corruption such as truncation and extraneous characters, thus improving the performance of terminal equipment.

In earlier software, the modem interface timing rules were not well understood and were difficult or impossible to implement using the separate command interface of the modem. The practical result is that the MICA in-band framing mode reduces the number of timeouts during PPP startup, and thus reduces startup time. MICA in-band framing is supported on MICA modems in Cisco AS5300 and Cisco AS5800 access servers.

To configure the MICA in-band framing mode control messages, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# line line-number [ending-line-number]`	Specifies the number of modem lines to configure and enters line configuration mode. If a range is entered, it must be equal to the number of modems in the router.
Step 2	`Router(config-line)# no flush-at-activation`	Improves PPP and SLIP startup. Normally a router avoids line and modem noise by clearing the initial data received within the first one or two seconds. However, when the autoselect PPP feature is configured, the router flushes characters initially received and then waits for more traffic. This flush causes timeout problems with applications that send only one carriage return.

The Cisco IOS software offers additional interface commands that can be set to control modem interface timing. Refer to the Cisco IOS command references for more information about the interface commands described in the following paragraphs.

When a link goes down and comes back up before the timer set by the **carrier-delay** command expires, the down state is effectively filtered, and the rest of the software on the switch is not aware that a link-down event occurred. Therefore, a large carrier delay timer results in fewer link-up and link-down events being detected. On the other hand, setting the carrier delay time to 0 means that every link-up and link-down event is detected.

When the link protocol goes down (because of loss of synchronization, for example), the interface hardware is reset and the data terminal ready (DTR) signal is held inactive for at least the specified interval. Setting the **pulse-time** command enable pulsing DTR signal intervals on serial interfaces, and is useful for handling encrypting or other similar devices that toggle the DTR signal to resynchronize.

Use the **modem dtr-delay** command to reduce the time that a DTR signal is held down after an asynchronous line clears and before the DTR signal is raised again to accept new calls. Incoming calls may be rejected in heavily loaded systems, even when modems are unused because the default DTR hold-down interval may be too long. The **modem dtr-delay** command is designed for lines used for an unframed asynchronous session such as Telnet. Lines used for a framed asynchronous session such as PPP should use the **pulse-time** interface command.

Enabling Modem Polling

The following example enables modem status polling through the out-of-band feature, which is associated to line 1:

```
Router# configure terminal
Router(config)# line 1
Router(config-line)# modem status-poll
```

Setting Modem Poll Intervals

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

```
Router# configure terminal
Router(config)# modem poll time 10
```

Setting Modem Poll Retry

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

```
Router# configure terminal
Router(config)# modem poll retry 5
```

Collecting Modem Statistics

Depending upon your modem type, the Cisco IOS software provides several **show** EXEC commands that allow you to display or poll various modem statistics. See Table 2 and Table 3 to find the **show** EXEC command appropriate for your modem type and the task you want to perform.

Logging EIA/TIA Events

To facilitate meaningful analysis of the modem log, turn the storage of specific types of EIA/TIA events on or off. To activate or inactivate the storage of a specific type of EIA/TIA modem event for a specific line or set of lines, use either of the following commands in line configuration mode, as needed:

Command	Purpose
`Router(config-line)# modem log {cts | dcd | dsr | dtr | ri | rs323 | rts | tst}`	Configures the types of EIA/TIA events that are stored in the modem log. The default setting stores no EIA/TIA events.
or	
`Router(config-line)# no modem log {cts | dcd | dsr | dtr | ri | rs323 | rts | tst}`	Turns off the logging of a specific type of EIA/TIA event.

Configuring a Microcom Modem to Poll for Statistics

Manageable Microcom modems have an out-of-band feature, which is used for polling modem statistics. To configure the system to poll for modem statistics, use the following commands in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# modem poll time seconds`	Specifies the number of seconds between statistical modem polling for Microcom modems. The default is 12 seconds. The configuration range is from 2 to 120 seconds.
Step 2	`Router(config)# modem poll retry number`	Sets the maximum number of polling attempts to Microcom modems. The default is three polling attempts. The configuration range is from 0 to 10 attempts.[1]
Step 3	`Router(config)# modem status-poll`	Polls for status and statistics for a Microcom modem through the modem's out-of-band feature.
Step 4	`Router(config)# modem buffer-size number`	Defines the number of modem events that each modem is able to store. The default is 100 events for each modem. Use the **show modem log** command to display modem events.

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

Troubleshooting Using a Back-to-Back Modem Test Procedure

You can manually isolate an internal back-to-back connection and data transfer between two modems for focused troubleshooting purposes. For example, if mobile users cannot dial in to 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 modem known to be functioning, such as modem 2/6. 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 connect and complete sending data in both directions. An operable modem and an inoperable modem do not connect with each other.

To perform the modem test procedure, enter the **test modem back-to-back** *first-slot/port second-slot/port* command, as follows:

Step 1 Perform a back-to-back modem test between two normal functioning modems. This example shows a successful connection between modem 1/1 and modem 1/0, which verifies normal operating conditions between these two modems:

```
Router# test modem back-to-back 1/1 1/0
Repetitions (of 10-byte packets) [1]: 10
Router#
%MODEM-5-B2BCONNECT: Modems (1/1) and (1/0) connected in back-to-back test: CONN
ECT9600/REL-MNP
%MODEM-5-B2BMODEMS: Modems (1/0) and (1/1) completed back-to-back test: success/
packets = 20/20
```

After you enter the **test modem back-to-back** command, you must define the number of packets sent between modems at the Repetitions prompt. The ideal range of packets to send and receive is from 1 to 100. The default is 1 packet that is 10 bytes large. The response message (for example, "success/packets = 20/20") tells you how many packets were 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.

When a known good modem is tested against a known bad modem, the back-to-back modem test fails. In the following example, modem 1/3 is suspected or proven to be inoperable or bad:

```
Router# test modem back-to-back 1/1 1/3
Repetitions (of 10-byte packets) [1]: 10
Router#
%MODEM-5-BADMODEMS: Modems (1/3) and (1/1) failed back-to-back test: NOCARRIER
```

Step 2 You would need to manually mark modem 1/3 as an inoperable or bad modem. You mark the bad modem by determining which line number corresponds with the modem. Use the **show modem 1/3** EXEC command to verify that TTY line number 4 (shown as `TTY4`) is used for modem 1/3:

```
Router# show modem 1/3
Mdm  Typ     Status     Tx/Rx     G  Duration  TX  RX  RTS  CTS  DSR  DCD  DTR
1/3  V34     Idle    28800/28800  0  00:00:00           x    x    x    x    x

Modem 1/3, Microcom MNP10 V34 Modem (Managed), TTY4
Firmware (Boot) Rev: 1.0(23) (1.0(5))
Modem config: Incoming and Outgoing
Protocol: reliable/MNP, Compression: V42bis
Management port config: Status polling and AT session
Management port status: Status polling and AT session
TX signals: -15 dBm, RX signals: -17 dBm


  Last clearing of "show modem" counters never
    0 incoming completes, 0 incoming failures
    0 outgoing completes, 0 outgoing failures
    0 failed dial attempts, 0 ring no answers, 1 busied outs
    0 no dial tones, 0 dial timeouts, 0 watchdog timeouts
    0 no carriers, 0 link failures, 0 resets, 0 recover oob
    0 protocol timeouts, 0 protocol errors, 0 lost events

Transmit Speed Counters:

Connection Speeds         75      300      600     1200     2400     4800
# of connections           0        0        0        0        0        0
Connection Speeds       7200     9600    12000    14400    16800    19200
# of connections           0        0        0        0        0        0
Connection Speeds      21600    24000    26400    28800    31200    32000
```

```
# of connections           0        0        0        1        0        0
Connection Speeds      33600    34000    36000    38000    40000    42000
# of connections           0        0        0        0        0        0
Connection Speeds      44000    46000    48000    50000    52000    54000
# of connections           0        0        0        0        0        0
Connection Speeds      56000
# of connections           0
```

Step 3 Enter line configuration mode and manually remove modem 1/3 from dial services by entering the **modem bad** command on line 4:

```
Router# configure terminal
Router(config)# line 4
Router(config-line)# modem bad
Router(config-line)# exit
Router(config)# exit
```

Step 4 Enter the **show modem** EXEC command or the **show modem** *slot/port* command to display the bad modem status.

Bad modems are marked with the letter B in the Mdm column of the **show modem** command display output.

```
Router# show modem

%SYS-5-CONFIG_I: Configured from console by consolem
                 Inc calls      Out calls   Busied   Failed  No       Succ
  Mdm   Usage    Succ   Fail    Succ   Fail   Out      Dial    Answer   Pct.
  1/0     0%       0      0       0      0       1        0       0      0%
  1/1     0%       0      0       0      0       3        0       0      0%
  1/2     0%       0      0       0      0       1        0       0      0%
B 1/3     0%       0      0       0      0       1        0       0      0%
  1/4     0%       0      0       0      0       1        0       0      0%
  1/5     0%       0      0       0      0       1        0       0      0%
  1/6     0%       0      0       0      0       1        0       0      0%
  1/7     0%       0      0       0      0       1        0       0      0%
  1/8     0%       0      0       0      0       1        0       0      0%
  1/9     0%       0      0       0      0       1        0       0      0%
  1/10    0%       0      0       0      0       1        0       0      0%
  1/11    0%       0      0       0      0       1        0       0      0%
  1/12    0%       0      0       0      0       1        0       0      0%
  1/13    0%       0      0       0      0       1        0       0      0%
  1/14    0%       0      0       0      0       1        0       0      0%
  1/15    0%       0      0       0      0       1        0       0      0%
  1/16    0%       0      0       0      0       1        0       0      0%
  1/17    0%       0      0       0      0       1        0       0      0%
  1/18    0%       0      0       0      0       0        0       0      0%
  1/19    0%       0      0       0      0       0        0       0      0%
  1/20    0%       0      0       0      0       0        0       0      0%
  1/21    0%       0      0       0      0       0        0       0      0%
  1/22    0%       0      0       0      0       0        0       0      0%
  1/23    0%       0      0       0      0       0        0       0      0%
```

Malfunctioning modems are also marked as Bad in the Status column of the **show modem** *slot/port* command display output, as the following example shows:

```
Router# show modem 1/3

Mdm  Typ    Status     Tx/Rx     G  Duration  TX  RX  RTS  CTS  DSR  DCD  DTR
1/3  V34    Bad     28800/28800  0  00:00:00           x    x    x    x    x

Modem 1/3, Microcom MNP10 V34 Modem (Managed), TTY4
Firmware (Boot) Rev: 1.0(23) (1.0(5))
Modem config: Incoming and Outgoing
```

```
Protocol: reliable/MNP, Compression: V42bis
Management port config: Status polling and AT session
Management port status: Status polling and AT session
TX signals: -15 dBm, RX signals: -17 dBm

  Last clearing of "show modem" counters never
    0 incoming completes, 0 incoming failures
    0 outgoing completes, 0 outgoing failures
    0 failed dial attempts, 0 ring no answers, 1 busied outs
    0 no dial tones, 0 dial timeouts, 0 watchdog timeouts
    0 no carriers, 0 link failures, 0 resets, 0 recover oob
    0 protocol timeouts, 0 protocol errors, 0 lost events

Transmit Speed Counters:

Connection Speeds         75      300      600     1200     2400     4800
# of connections           0        0        0        0        0        0
Connection Speeds       7200     9600    12000    14400    16800    19200
# of connections           0        0        0        0        0        0
Connection Speeds      21600    24000    26400    28800    31200    32000
# of connections           0        0        0        1        0        0
Connection Speeds      33600    34000    36000    38000    40000    42000
# of connections           0        0        0        0        0        0
Connection Speeds      44000    46000    48000    50000    52000    54000
# of connections           0        0        0        0        0        0
Connection Speeds      56000
# of connections           0
```

Clearing a Direct Connect Session on a Microcom Modem

The examples in this section are for Microcom modems.

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

```
Router# 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:

```
Router# clear modem at-mode 1/1
clear "modem at-mode" for modem 1/1 [confirm] <press Return>
Router#
```

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

```
Direct connect session cleared by vty0 (172.19.1.164)
```

Displaying Local Disconnect Reasons

To find out why a modem ended its connection or why a modem is not operating at peak performance, use the **show modem call-stats** [*slot*] EXEC command.

Disconnect reasons are described using four hexadecimal digits. The three lower-order digits can be used to identify the disconnect reason. The high-order digit generally indicates the type of disconnect reason or the time at which the disconnect occurred. For detailed information on the meaning of hexadecimal values for MICA modem disconnects, refer to the TAC Tech Notes document, *MICA Modem States and Disconnect Reasons*, at the following URL: http://www.cisco.com/warp/public/76/mica-states-drs.html

For detailed information on the meaning of hexadecimal values for NextPort modem disconnects, refer to the TAC Tech Notes document, *Interpreting NextPort Disconnect Reason Codes*, at the following URL: http://www.cisco.com/warp/public/471/np_disc_code.html .

Local disconnect reasons are listed across the top of the screen display (for example, wdogTimr, compress, retrain, inacTout, linkFail, moduFail, mnpProto, and lapmProt). In the body of the screen display, the number of times each modem disconnected is displayed (see the # column). For a particular disconnect reason, the % column indicates the percent that a modem was logged for the specified disconnect reason with respect to the entire modem pool for that given reason. For example, out of all the times the rmtLink error occurred on all the modems in the system, the rmtLink error occurred 10 percent of the time on modem 0/22.

Malfunctioning modems are detected by an unusually high number of disconnect counters for a particular disconnect reason. For example, if modem 1/0 had a high number of compression errors compared to the remaining modems in system, modem 1/0 would likely be the inoperable modem.

To reset the counters displayed by the **show modem call-stats** command, enter the **clear modem counters** command.

Note For a complete description of each error field displayed by the commands on this page, refer to the *Cisco IOS Dial Technologies Command Reference*. Remote disconnect reasons are not described by the **show modem** command output.

The following example displays output for the **show modem call-stats** command. Because of the screen size limitation of most terminal screen displays, not all possible disconnect reasons are displayed at one time. Only the top eight most frequently experienced disconnect reasons are displayed at one time.

```
Router# show modem call-stats

  dial-in/dial-out call statistics

       lostCarr  dtrDrop  rmtLink wdogTimr compress  retrain inacTout linkFail
  Mdm     #   %    #   %    #   %    #   %    #   %    #   %    #   %    #   %
* 0/0     6   2    2   3    1   0    0   0    0   0    0   0    0   0    0   0
* 0/1     5   2    2   3    2   1    0   0    0   0    0   0    0   0    0   0
  0/2     5   2    2   3    4   3    0   0    0   0    0   0    0   0    0   0
* 0/3     5   2    2   3    2   1    0   0    0   0    0   0    0   0    0   0
* 0/4     5   2    1   1    1   0    0   0    0   0    0   0    0   0    0   0
* 0/5     5   2    2   3    2   1    0   0    0   0    0   0    0   0    0   0
* 0/6     4   1    2   3    2   1    0   0    0   0    0   0    0   0    0   0
* 0/7     4   1    2   3    4   3    0   0    0   0    0   0    0   0    0   0
* 0/8     6   2    1   1    3   2    0   0    0   0    0   0    0   0    0   0
* 0/9     5   2    1   1    1   0    0   0    0   0    0   0    0   0    0   0
* 0/10    5   2    1   1    2   1    0   0    0   0    0   0    0   0    0   0
* 0/11    5   2    1   1    2   1    0   0    0   0    0   0    0   0    0   0
  0/12    5   2    2   3    2   1    0   0    0   0    0   0    0   0    0   0
* 0/13    5   2    1   1    1   0    0   0    0   0    0   0    0   0    0   0
* 0/14    5   2    1   1    1   0    0   0    0   0    0   0    0   0    0   0
* 0/15    5   2    1   1    1   0    0   0    0   0    0   0    0   0    0   0
* 0/16    5   2    1   1    1   0    0   0    0   0    0   0    0   0    0   0
* 0/17    5   2    1   1    2   1    0   0    0   0    0   0    0   0    0   0
* 0/18    5   2    1   1    2   1    0   0    0   0    0   0    0   0    0   0
* 0/19    5   2    1   1    3   2    0   0    0   0    0   0    0   0    0   0
```

```
* 0/20     5    2     1    1     1    0     0    0     0    0     0    0     0    0     0    0
* 0/21     5    2     1    1     1    0     0    0     0    0     0    0     0    0     0    0
* 0/22     5    2     1    1    11   10     0    0     0    0     0    0     0    0     0    0
* 0/23     5    2     1    1     2    1     0    0     0    0     0    0     0    0     0    0
* 2/0      4    1     2    3     2    1     0    0     0    0     0    0     0    0     0    0
* 2/1      5    2     1    1     2    1     0    0     0    0     0    0     0    0     0    0
* 2/2      5    2     2    3     0    0     0    0     0    0     0    0     0    0     0    0
* 2/3      5    2     1    1     2    1     0    0     0    0     0    0     0    0     0    0
* 2/4      5    2     1    1     2    1     0    0     0    0     0    0     0    0     0    0
* 2/5      5    2     1    1     2    1     0    0     0    0     0    0     0    0     0    0
* 2/6      4    1     1    1     1    0     0    0     0    0     0    0     0    0     0    0
* 2/7      5    2     1    1     1    0     0    0     0    0     0    0     0    0     0    0
* 2/8      5    2     1    1     1    0     0    0     0    0     0    0     0    0     0    0
* 2/9      4    1     1    1     2    1     0    0     0    0     0    0     0    0     0    0
* 2/10     5    2     1    1     0    0     0    0     0    0     0    0     0    0     0    0
* 2/11     5    2     1    1     5    4     0    0     0    0     0    0     0    0     0    0
* 2/12     5    2     1    1     2    1     0    0     0    0     0    0     0    0     0    0
* 2/13     5    2     1    1     1    0     0    0     0    0     0    0     0    0     0    0
* 2/14     5    2     1    1     2    1     0    0     0    0     0    0     0    0     0    0
* 2/15     4    1     1    1     3    2     0    0     0    0     0    0     0    0     0    0
* 2/16     4    1     1    1     3    2     0    0     0    0     0    0     0    0     0    0
* 2/17     5    2     2    3     9    8     0    0     0    0     0    0     0    0     0    0
* 2/18     4    1     1    1     1    0     0    0     0    0     0    0     0    0     0    0
* 2/19     3    1     1    1     2    1     0    0     0    0     0    0     0    0     0    0
* 2/20     7    3     1    1     8    7     0    0     0    0     0    0     0    0     0    0
* 2/21     5    2     1    1     1    0     0    0     0    0     0    0     0    0     0    0
* 2/22     4    1     1    1     2    1     0    0     0    0     0    0     0    0     0    0
* 2/23     5    2     1    1     2    1     0    0     0    0     0    0     0    0     0    0
 Total   233        59       110         0         0         0         0         0         0

  dial-out call statistics

          noCarr noDitone      busy     abort dialStrg autoLgon dialTout   rmtHgup
  Mdm      #   %    #   %    #   %    #   %    #   %    #   %    #   %    #   %
* 0/0      1   1    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/1      0   0    0   0    0   0    0   0    0   0    0   0    0   0    0   0
  0/2      0   0    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/3      1   1    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/4      1   1    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/5      0   0    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/6      1   1    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/7      5   5    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/8      0   0    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/9      1   1    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/10     0   0    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/11     5   5    0   0    0   0    0   0    0   0    0   0    0   0    0   0
  0/12     0   0    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/13     0   0    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/14     1   1    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/15     1   1    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/16     2   2    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/17     4   4    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/18     5   5    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/19     1   1    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/20     0   0    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/21     0   0    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/22     5   5    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 0/23     1   1    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 2/0      2   2    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 2/1      3   3    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 2/2      0   0    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 2/3      0   0    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 2/4      0   0    0   0    0   0    0   0    0   0    0   0    0   0    0   0
* 2/5      1   1    0   0    0   0    0   0    0   0    0   0    0   0    0   0
```

```
* 2/6      1   1     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/7      4   4     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/8      7   8     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/9      4   1     1   1     2   1     0   0     0   0     0   0     0   0     0   0
* 2/10     1   1     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/11     1   1     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/12     1   1     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/13     1   1     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/14     4   4     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/15     1   1     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/16     1   1     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/17     5   5     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/18     5   5     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/19     3   3     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/20     0   0     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/21     4   4     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/22     2   2     0   0     0   0     0   0     0   0     0   0     0   0     0   0
* 2/23     0   0     0   0     0   0     0   0     0   0     0   0     0   0     0   0
  Total   84         0         0         0         0         0         0         0
```

Removing Inoperable Modems

To manually remove inoperable modems from dialup services, use the following commands in line configuration mode:

	Command	Purpose
Step 1	`Router(config-line)#` **`modem bad`**	Removes and idles the modem from service and indicates it as suspected or proven to be inoperable.
Step 2	`Router(config-line)#` **`modem hold-reset`**	Resets and isolates the modem hardware for extensive troubleshooting.
Step 3	`Router(config-line)#` **`modem shutdown`**	Abruptly shuts down a modem from dial service.
Step 4	`Router(config-line)#` **`modem recovery-time`** *`minutes`*	Sets the maximum amount of time for which the call-switching module waits for a local modem to respond to a request before it is considered locked in a suspended state. The default is 5 minutes.

If you use the **modem bad** command to remove an idle modem from dial services and mark it as inoperable, the letter B is used to identify the modem as bad. The letter B appears in the Status column in the output of **show modem** *slot/port* command and in the far left column in the output of the **show modem** command. Use the **no modem bad** command to unmark a modem as B and restore it for dialup connection services. If the letter B appears next to a modem number, it means the modem was removed from service with the **modem shutdown** command.

Note Only idle modems can be marked "bad" by the **modem bad** command. If you want to mark a modem bad that is actively supporting a call, first enter the **modem shutdown** command, then enter the **modem bad** command.

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

command also resets 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.

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

```
Router# configure terminal
Router(config)# line 4
Router(config-line)# modem hold-reset
Router(config-line)# no modem hold-reset
```

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.

```
Router# configure terminal
Router(config)# line 1
Router(config)# modem busyout
```

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

```
Router# configure terminal
Router(config)# line 2
Router(config)# modem shutdown
```

In the following example, the modem using TTY line 3 is actively supporting a call (as indicated by the asterisk). However, we want to mark the modem bad because it has poor connection performance. First, abruptly shut down the modem and drop the call with the **modem shutdown** command, and then enter the **modem bad** command to take the modem out of service.

```
Router# show modem

                Inc calls     Out calls     Busied   Failed  No        Succ
  Mdm  Usage    Succ   Fail   Succ   Fail   Out      Dial    Answer    Pct.
  1/0    37%      98      4      0      0       0         0        0     96%
  1/1    38%      98      2      0      0       0         0        0     98%
* 1/2    2%        3     99      0      0       0         0        0     1%
.
.
.

Router# configure terminal
Router(config)# line 3
Router(config)# modem shutdown
Router(config)# modem bad
Router(config)# exit

Router# show modem

                Inc calls     Out calls     Busied   Failed  No        Succ
  Mdm  Usage    Succ   Fail   Succ   Fail   Out      Dial    Answer    Pct.
  1/0    37%      98      4      0      0       0         0        0     96%
  1/1    38%      98      2      0      0       0         0        0     98%
B 1/2    2%        3     99      0      0       0         0        0     1%
```

For more information about modem recovery procedures, refer to TAC Tech Notes *Configuring MICA Modem Recovery* at http://www.cisco.com/warp/public/76/modem-recovery.html and *Configuring NextPort SPE Recovery* at http://www.cisco.com/warp/public/76/spe-recovery.html.

Busying Out a Modem Card

To busy out a modem card in a Cisco access server, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# line shelf/slot/port`	Specifies the line number, by specifying the shelf, slot, and port numbers; you must type in the slashes. This command also begins line configuration mode.
Step 2	`Router(config-line)# modem busyout`	Having specified the modem to be busied out with the **line** command, enter the **modem busyout** command to busy out the modem. The command disables the modem associated with line *shelf/slot/port* from dialing and answering calls.You need not specify a *shelf/slot/port* number again in this command.
Step 3	`Router(config-line)# modem shutdown`	Having specified the modem to be shut down with the **line** command, enter the **modem shutdown** command to shut down the modem, whether or not it has already been busied out. You need not specify a *shelf/slot/port* number again in this command because you have already done so with the **line** command.
Step 4	`Router(config-line)# exit`	Exits line configuration mode and returns to global configuration mode.
Step 5	`Router(config)# modem busyout-threshold number`	Specifies a threshold number using the **modem busyout-threshold** *number* command to balance the number of DS0s with the number of modem lines. For more information, refer to the *Cisco IOS Dial Technologies Command Reference*.
Step 6	`Router(config)# exit`	Exits global configuration mode and returns to privileged EXEC mode.
Step 7	`Router# show busyout`	From privileged EXEC mode, verifies that the line is busied out. If there are active calls, the software waits until the call terminates before the line is busied out.

The **modem busyout** command disables the modem associated with a specified line from dialing and answering calls. The **modem busyout** command can busy out and eventually terminate all 72 ports on the Cisco AS5800 modem card.

Monitoring Resources on Cisco High-End Access Servers

The following tasks enable you to monitor the network access server (NAS) health conditions at the DS0 level, PRI bearer channel level, and modem level. Performing these tasks will benefit network operation with improved visibility into the line status for the NAS for comprehensive health monitoring and notification capability, and improved troubleshooting and diagnostics for large-scale dial networks.

Perform the following tasks to monitor resource availability on the Cisco high-end access servers:

- Enabling DS0 Busyout Traps—DS0 busyout traps are generated when there is a request to busy out a DS0, when there is a request to take a DS0 out of busyout mode, or when busyout completes and the DS0 is out-of-service. DS0 busyout traps are generated at the DS0 level for both CAS and ISDN

configured lines. This feature is enabled and disabled through use of the CLI and MIBs. DS0 busyout traps are disabled by default and are supported on Cisco AS5300, Cisco AS5400, and Cisco AS5800 universal access servers.

- Enabling ISDN PRI Requested Channel Not Available Traps—ISDN PRI channel not available traps are generated when a requested DS0 channel is not available, or when there is no modem available to take the incoming call. This feature is available only for ISDN PRI interfaces. This feature is enabled and disabled through use of CLI for ISDN traps and the CISCO-ISDN-MIB. ISDN PRI channel not available traps are disabled by default and are supported on the Cisco AS5300, Cisco AS5400, and Cisco AS5800.
- Enabling Modem Health Traps—Modem health traps are generated when a modem port is bad, disabled, reflashed, or shut down, or when there is a request to busy out the modem. This feature is enabled and disabled through use of CLI and the CISCO-MODEM-MGMT-MIB. Modem health traps are disabled by default and are supported on the Cisco AS5300, Cisco AS5400, and Cisco AS5800.
- Enabling DS1 Loopback Traps—DS1 loopback traps are generated when a DS1 line goes into loopback mode. This feature is enabled and disabled by CLI and the CISCO-POP-MGMT-MIB. DS1 loopback traps are disabled by default and are supported on the Cisco AS5300 and Cisco AS5400 only.

The CISCO-POP-MGMT-MIB supplies the DS0 busyout traps and the DS1 loopback traps. The CISCO-MODEM-MGMT-MIB supplies additional modem health traps when the modem port becomes non-functional. The CISCO-ISDN-MIB supplies additional traps for ISDN PRI channel not available.

To obtain lists of supported MIBs by platform and Cisco IOS release, and to download MIB modules, go to the Cisco MIB website on Cisco.com at http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml.

See the sections "Verifying Enabled Traps" and "Troubleshooting the Traps" to verify and troubleshoot configuration. The section "NAS Health Monitoring Example" provides output of a configuration with the NAS health monitoring features enabled.

Enabling DS0 Busyout Traps

Before you enable DS0 busyout traps, the SNMP manager must already have been installed on your workstation, and the SNMP agent must be configured on the NAS by entering the **snmp-server community** and **snmp-server host** commands. Refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* for more information on these commands.

To generate DS0 busyout traps, use the following command in global configuration mode:

Command	Purpose
`Router(config)# snmp-server enable traps ds0-busyout`	Generates a trap when there is a request to busy out a DS0 or to indicate when busyout finishes.

Enabling ISDN PRI Requested Channel Not Available Traps

To generate ISDN PRI requested channel not available traps, use the following command in global configuration mode:

Command	Purpose
`Router(config)# snmp-server enable traps isdn chan-not-avail`	Generates a trap when the NAS rejects an incoming call on an ISDN PRI interface because the channel is not available.

Enabling Modem Health Traps

To generate modem health traps, use the following command in global configuration mode:

Command	Purpose
`Router(config)# snmp-server enable traps modem-health`	Generates a trap when a modem port is bad, disabled, or prepared for firmware download; when download fails; when placed in loopback mode for maintenance; or when there is a request to busy out the modem.

Enabling DS1 Loopback Traps

To generate DS1 loopback traps, use the following command in global configuration mode:

Command	Purpose
`Router(config)# snmp-server enable traps ds1-loopback`	Generates a trap when the DS1 line goes into loopback mode.

Verifying Enabled Traps

To verify that the traps are enabled, use the **show run** command. The following output indicates that all the traps are enabled:

```
Router(config)# show run

snmp-server enable traps ds0-busyout
snmp-server enable traps isdn chan-not-avail
snmp-server enable traps modem-health
snmp-server enable traps ds1-loopback
```

Additionally, you can use the **show controllers** command with the **timeslots** keyword to display details about the channel state. This feature shows whether the DS0 channels of a particular controller are in idle, in-service, maintenance, or busyout state. This enhancement applies to both CAS and ISDN PRI interfaces and is supported on the Cisco AS5300 and Cisco AS5400 only.

Troubleshooting the Traps

To troubleshoot the traps, turn on the debug switch for SNMP packets by entering the following command in privileged EXEC mode:

```
Router# debug snmp packets
```

Check the resulting output to see that the SNMP trap information packet is being sent. The output will vary based on the kind of packet sent or received:

```
SNMP: Packet received via UDP from 10.5.4.1 on Ethernet0
```

```
SNMP: Get-next request, reqid 23584, errstat 0, erridx 0
sysUpTime = NULL TYPE/VALUE
 system.1 = NULL TYPE/VALUE
 system.6 = NULL TYPE/VALUE
SNMP: Response, reqid 23584, errstat 0, erridx 0
 sysUpTime.0 = 2217027
 system.1.0 = Cisco Internetwork Operating System Software
 system.6.0 =
SNMP: Packet sent via UDP to 10.5.4.1
```

You can also use trap monitoring and logging tools like snmptrapd, with debugging flags turned on, to monitor output.

NAS Health Monitoring Example

The following is sample configuration output showing all NAS health monitoring traps turned on:

```
Building configuration...

Current configuration:
! Last configuration change at 12:27:30 pacific Thu May 25 2000
version xx.x
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname router
!
aaa new-model
aaa authentication ppp default group radius
enable password <password>
!
spe 1/0 1/7
 firmware location system:/ucode/mica_port_firmware
spe 2/0 2/7
 firmware location system:/ucode/mica_port_firmware
!
resource-pool disable
!
clock timezone PDT -8
clock calendar-valid
no modem fast-answer
modem country mica usa
modem link-info poll time 60
modem buffer-size 300
ip subnet-zero
!
isdn switch-type primary-5ess
isdn voice-call-failure 0
!
controller T1 0
 framing esf
 clock source line primary
 linecode b8zs
 pri-group timeslots 1-24
!
controller T1 1
 framing esf
 linecode b8zs
 ds0-group 0 timeslots 1-24 type e&m-fgb
 cas-custom 0
!
```

```
controller T1 2
 shutdown
 clock source line secondary 2
!
controller T1 3
 shutdown
 clock source line secondary 3
!
controller T1 4
 shutdown
 clock source line secondary 4
!
controller T1 5
 shutdown
 clock source line secondary 5
!
controller T1 6
 shutdown
 clock source line secondary 6
!
controller T1 7
 shutdown
 clock source line secondary 7
!
interface Loopback0
 ip address 10.5.4.1
!
interface Ethernet0
 no ip address
 shutdown
!
interface Serial0
 no ip address
 shutdown
!
interface Serial1
 no ip address
 shutdown
!
interface Serial2
 no ip address
 shutdown
!
interface Serial3
 no ip address
 shutdown
!
interface Serial0:23
 no ip address
 ip mroute-cache
 isdn switch-type primary-5ess
 isdn incoming-voice modem
 no cdp enable
!
interface FastEthernet0
 ip address 10.5.4.1
 duplex full
 speed auto
 no cdp enable
!
interface Group-Async1
 ip unnumbered FastEthernet0
 encapsulation ppp
 ip tcp header-compression passive
```

```
 no ip mroute-cache
 async mode interactive
 peer default ip address pool swattest
 no fair-queue
 ppp authentication chap
 ppp multilink
 group-range 1 192
!
interface Dialer1
 ip unnumbered FastEthernet0
 encapsulation ppp
 ip tcp header-compression passive
 dialer-group 1
 peer default ip address pool swattest
 pulse-time 0
 no cdp enable
!
ip local pool swattest 10.5.4.1
ip default-gateway 10.5.4.1
ip classless
!
dialer-list 1 protocol ip permit
snmp-server engineID local 00000009020000D058890CF0
snmp-server community public RO
snmp-server packetsize 2048
snmp-server enable traps ds0-busyout
snmp-server enable traps isdn chan-not-avail
snmp-server enable traps modem-health
snmp-server enable traps ds1-loopback
snmp-server host 10.5.4.1 public
!
radius-server host 10.5.4.1 auth-port 1645 acct-port 1646
radius-server retransmit 3
radius-server key <password>
!
line con 0
 transport input none
line 1 192
 autoselect ppp
 modem InOut
 transport preferred none
 transport input all
 transport output none
line aux 0
line vty 0 4
end
```

Configuration Examples for Modem Management

This section provides the following examples:

- NextPort Modem Log Example
- Modem Performance Summary Example
- Modem AT-Mode Example
- Connection Speed Performance Verification Example

For additional information and examples about the commands in this chapter, refer to the *Cisco IOS Dial Technologies Command Reference*.

NextPort Modem Log Example

The following is partial sample output for the Cisco AS5400 with the NextPort Distributed forwarding Card (DFC). This example shows the port history event log for slot 5, port 47:

```
Router# show port modem log 5/47

Port 5/47 Events Log
     Service type: DATA_FAX_MODEM
     Service mode: DATA_FAX_MODEM
     Session State: IDLE
  00:02:23: incoming called number: 35160
     Service type: DATA_FAX_MODEM
     Service mode: DATA_FAX_MODEM
     Session State: IDLE
     Service type: DATA_FAX_MODEM
     Service mode: DATA_FAX_MODEM
     Session State: ACTIVE
  00:02:23: Modem State event:
            State: Connect
  00:02:16: Modem State event:
            State: Link
  00:02:13: Modem State event:
            State: Train Up
  00:02:05: Modem State event:
            State: EC Negotiating
  00:02:05: Modem State event:
            State: Steady
  00:02:05: Modem Static event:
    Connect Protocol                          :   LAP-M
    Compression                               :   V.42bis
    Connected Standard                        :   V.34+
    TX,RX Symbol Rate                         :   3429, 3429
    TX,RX Carrier Frequency                   :   1959, 1959
    TX,RX Trellis Coding                      :   16/16
    Frequency Offset                          :   0  Hz
    Round Trip Delay                          :   0  msecs
    TX,RX Bit Rate                            :   33600, 33600
    Robbed Bit Signalling (RBS) pattern       :   0
    Digital Pad                               :   None
    Digital Pad Compensation                  :   None
    4 bytes of link info not formatted        :   0x00 0x00 0x00 0x00 0x00
  00:02:06:Modem Dynamic event:
    Sq Value                                  :   5
    Signal Noise Ratio                        :   40  dB
    Receive Level                             :   -12  dBm
    Phase Jitter Frequency                    :   0  Hz
    Phase Jitter Level                        :   2  degrees
    Far End Echo Level                        :   -90  dBm
    Phase Roll                                :   0  degrees
    Total Retrains                            :   0
    EC Retransmission Count                   :   0
    Characters transmitted, received          :   0, 0
    Characters received BAD                   :   0
    PPP/SLIP packets transmitted, received    :   0, 0
    PPP/SLIP packets received (BAD/ABORTED)   :   0
    EC packets transmitted, received OK       :   0, 0
    EC packets (Received BAD/ABORTED)         :   0
```

Modem Performance Summary Example

You can display a high level summary of the performance of a modem with the **show modem summary** command:

```
Router# show modem summary

         Incoming calls       Outgoing calls      Busied   Failed   No    Succ
Usage  Succ    Fail  Avail    Succ    Fail  Avail    Out      Dial     Ans   Pct.
  14%  2489     123     15       0       0     15        0         3     3     95%
```

Modem AT-Mode Example

The following example shows that modem 1/1 has one open AT directly connected session:

```
Router# show modem at-mode

Active AT-MODE management sessions:
Modem    User's Terminal
1/1 0 cty 0
```

Connection Speed Performance Verification Example

Making sure that your modems are connecting at the correct connection speeds is an important aspect of managing modems. The **show modem connect-speeds** and **show modem** commands provide performance information that allow you to investigate possible inoperable or corrupt modems or T1/E1 lines. For example, suppose you have an access server that is fully populated with V.34 modems. If you notice that modem 1/0 is getting V.34 connections only 50 percent of the time, whereas all the other modems are getting V.34 connections 80 percent of the time, then modem 1/0 is probably malfunctioning. If you are reading low connection speeds across all the modems, you may have a faulty channelized T1 or ISDN PRI line connection.

To display connection speed information for all modems that are running in your system, use the **show modem connect-speeds** *max-speed* EXEC command. Because most terminal screens are not wide enough to display the entire range of connection speeds at one time (for example, 75 to 56,000 bps), the *max-speed* argument is used. This argument specifies the contents of a shifting baud-rate window, which provides you with a snapshot of the modem connection speeds for your system. Replace the *max-speed* argument with the maximum connect speed that you want to display. You can specify from 12,000 to 56,000 bps. If you are interested in viewing a snapshot of lower baud rates, specify a lower connection speed. If you are interested in displaying a snapshot of higher rates, specify a higher connection speed.

The following example displays connection speed information for modems running up to 33,600 bps:

```
Router# show modem connect-speeds 33600

  transmit connect speeds

   Mdm   14400  16800  19200  21600  24000  26400  28800  31200  33600 TotCnt
* 0/0       0      0      0      0      0      0      4      4      1      9
* 0/1       2      0      0      0      0      0      3      3      1      9
  0/2       2      0      0      0      0      1      2      4      1     10
* 0/3       0      0      0      1      0      0      3      4      1      9
* 0/4       1      0      0      0      0      2      2      1      1      7
* 0/5       0      0      0      0      0      0      4      4      1      9
* 0/6       0      0      0      0      0      1      3      3      1      8
* 0/7       0      0      0      2      0      0      4      3      1     10
* 0/8       2      0      0      0      0      0      3      4      1     10
```

```
* 0/9        0       0       0       0       0       0       4       3       0       7
* 0/10       1       0       0       0       0       1       3       2       1       8
* 0/11       0       0       0       0       0       0       4       3       1       8
  0/12       1       0       0       0       0       0       4       2       1       8
* 0/13       0       0       0       0       0       0       4       2       1       7
* 0/14       1       0       0       0       0       1       2       2       1       7
* 0/15       0       0       0       0       0       0       4       2       1       7
* 0/16       0       0       0       1       0       0       3       2       1       7
* 0/17       1       0       0       0       0       0       4       2       1       8
* 0/18       1       0       0       0       0       0       3       3       1       8
* 0/19       0       0       0       0       0       0       5       3       1       9
* 0/20       0       0       0       0       0       0       4       2       1       7
* 0/21       1       0       0       0       0       0       4       2       0       7
* 0/22       0       0       0       0       0       0       7       9       1      17
* 0/23       0       0       0       0       0       2       2       3       1       8
* 2/0        0       0       0       1       0       0       3       3       1       8
* 2/1        0       0       0       0       0       0       5       2       1       8
* 2/2        0       0       0       1       0       0       4       1       1       7
* 2/3        1       0       0       0       0       0       4       2       1       8
* 2/4        0       0       0       0       0       0       5       2       1       8
* 2/5        0       0       0       0       0       0       4       3       1       8
* 2/6        0       0       0       0       0       0       3       2       1       6
* 2/7        1       0       0       0       0       1       3       2       0       7
* 2/8        1       0       0       0       0       0       3       2       1       7
* 2/9        0       0       0       0       0       1       3       2       1       7
* 2/10       2       0       0       0       0       2       1       0       1       6
* 2/11       0       0       0       1       0       1       3       5       1      11
* 2/12       0       0       0       0       0       0       5       2       1       8
* 2/13       1       0       0       0       0       0       5       0       1       7
* 2/14       1       0       0       0       0       0       3       3       1       8
* 2/15       1       0       0       0       0       1       2       3       1       8
* 2/16       0       0       0       0       0       0       4       3       1       8
* 2/17       0       0       0       0       0       0       5      11       0      16
* 2/18       0       0       0       1       0       1       1       2       1       6
* 2/19       0       0       0       0       0       0       2       3       1       6
* 2/20       1       0       0       0       0       2       3       9       1      16
* 2/21       1       0       0       0       0       0       4       1       1       7
* 2/22       0       0       0       1       0       0       2       3       1       7
* 2/23       0       0       0       0       0       1       3       3       1       8
 Tot        23       0       0       9       0      18     165     141      44     400
 Tot %       5       0       0       2       0       4      41      35      11

  receive connect speeds

  Mdm   14400  16800  19200  21600  24000  26400  28800  31200  33600 TotCnt
* 0/0        0       0       0       0       0       4       1       3       1       9
* 0/1        2       0       0       0       0       3       1       2       1       9
  0/2        2       0       0       0       0       3       1       3       1      10
* 0/3        0       0       0       1       0       3       4       0       1       9
* 0/4        1       0       0       0       0       4       0       1       1       7
* 0/5        0       0       0       0       0       4       3       1       1       9
* 0/6        0       0       0       0       0       4       0       3       1       8
* 0/7        0       0       0       2       0       4       1       2       1      10
* 0/8        2       0       0       0       0       3       0       5       0      10
* 0/9        0       0       0       0       0       4       2       0       1       7
* 0/10       1       0       0       0       0       4       0       2       1       8
* 0/11       0       0       0       0       0       4       0       3       1       8
  0/12       1       0       0       0       0       2       2       2       1       8
* 0/13       0       0       0       0       0       4       1       1       1       7
* 0/14       1       0       0       0       0       2       3       0       1       7
* 0/15       0       0       0       0       0       4       1       1       1       7
* 0/16       0       0       0       1       0       3       2       0       1       7
* 0/17       1       0       0       0       0       4       1       1       1       8
* 0/18       1       0       0       0       0       3       2       1       1       8
```

```
* 0/19        0       0       0       0       0       5       1       2       1       9
* 0/20        0       0       0       0       0       4       0       3       0       7
* 0/21        1       0       0       0       0       4       0       1       1       7
* 0/22        0       0       0       0       0       6       6       4       1      17
* 0/23        0       0       0       0       0       4       2       1       1       8
* 2/0         0       0       0       1       0       3       1       2       1       8
* 2/1         0       0       0       0       0       3       3       1       1       8
* 2/2         0       0       0       1       0       4       0       1       1       7
* 2/3         1       0       0       0       0       3       2       1       1       8
* 2/4         0       0       0       0       0       4       2       1       1       8
* 2/5         0       0       0       0       0       4       1       2       1       8
* 2/6         0       0       0       0       0       3       0       3       0       6
* 2/7         1       0       0       0       1       2       2       0       1       7
* 2/8         1       0       0       0       0       3       0       2       1       7
* 2/9         0       0       0       0       0       4       1       1       1       7
* 2/10        2       0       0       0       0       3       0       0       1       6
* 2/11        0       0       0       1       0       3       1       5       1      11
* 2/12        0       0       0       0       0       4       3       0       1       8
* 2/13        1       0       0       0       0       2       3       0       1       7
* 2/14        1       0       0       0       0       3       2       1       1       8
* 2/15        1       0       0       0       0       3       0       3       1       8
* 2/16        0       0       0       0       0       4       0       4       0       8
* 2/17        0       0       0       0       0       5       2       8       1      16
* 2/18        0       0       1       0       0       2       1       1       1       6
* 2/19        0       0       0       0       0       2       2       1       1       6
* 2/20        1       0       0       0       0       4       2       8       1      16
* 2/21        1       0       0       0       0       4       0       1       1       7
* 2/22        0       0       1       0       0       2       0       3       1       7
* 2/23        0       0       0       0       0       4       2       1       1       8
 Tot         23       0       2       7       1     167      64      92      44     400
 Tot %        5       0       0       1       0      41      16      23      11
```