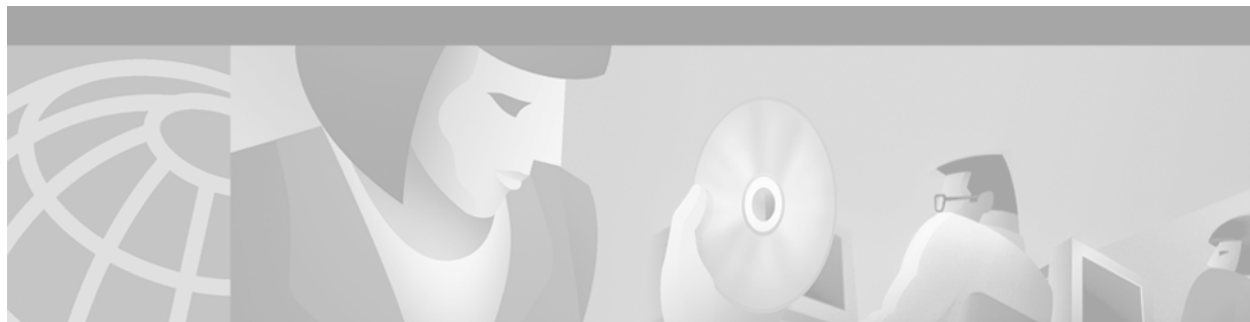




Overview of Modem Interfaces

This chapter describes modem interfaces in the following sections:

- [Cisco Modems and Cisco IOS Modem Features](#)
- [Cisco IOS Modem Components](#)
- [Logical Constructs in Modem Configurations](#)

See the chapter [“Overview of Dial Interfaces, Controllers, and Lines”](#) for more information about Cisco asynchronous serial interfaces.

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 modem support commands in this chapter, refer to the *Cisco IOS Modem Command Reference*. To locate documentation of other commands that appear in this chapter, use the command reference master index or search online.

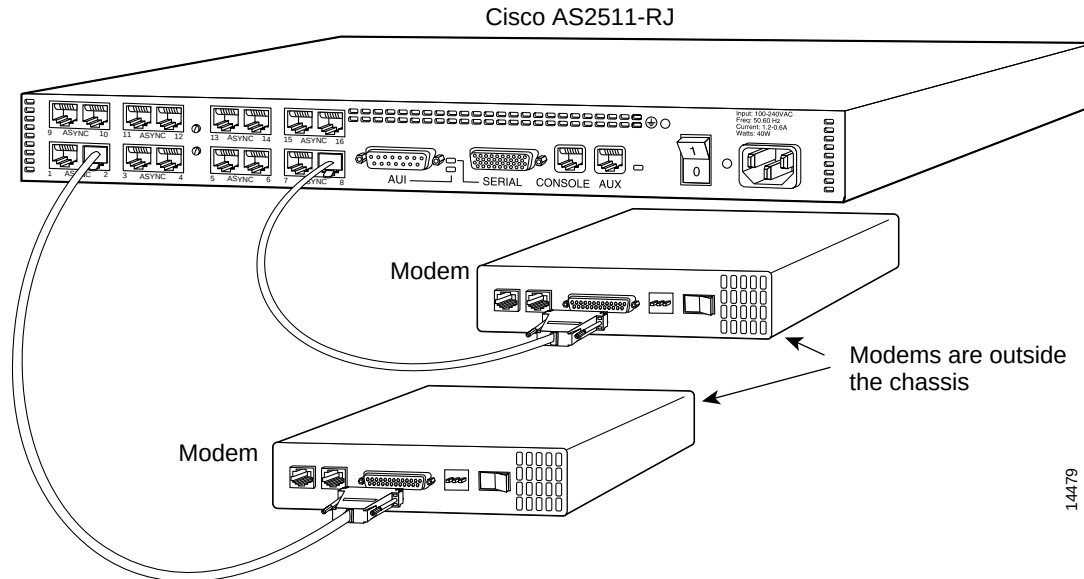
Cisco Modems and Cisco IOS Modem Features

Deciding which asynchronous features to use, to some degree, depends on your hardware configuration. All Cisco access servers must have their asynchronous interfaces and lines configured for network protocol support. Commands entered in asynchronous interface mode configure protocol-specific parameters for asynchronous interfaces, whereas commands entered in line configuration mode configure the physical and logical aspects for the same port.

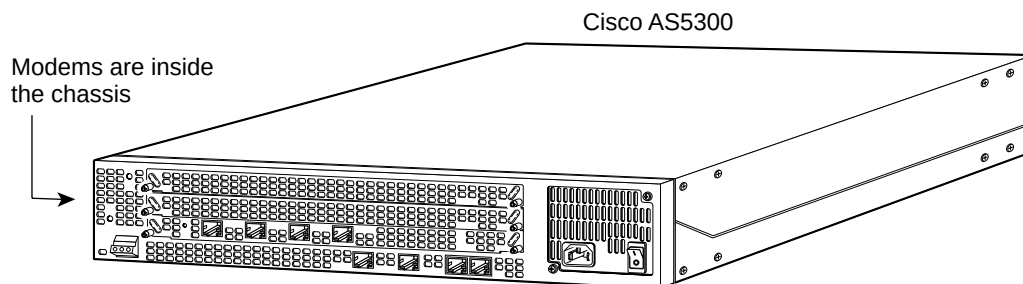
Modems inside high-end access servers need a localized modem country code. This code is projected from the Cisco IOS software to the onboard modems using the **modem country** {**mica** | **microcom_hdms**} *country* command. The following are high-end access servers: Cisco AS5800, Cisco AccessPath, Cisco AS5300, and the Cisco AS5200.

Modems externally attached to low-end access servers need to receive initialization strings from the **modem autoconfigure discovery** command. For troubleshooting tips, see the section [“External Modems on Low-End Access Servers”](#) in the chapter [“Configuring and Managing External Modems.”](#) The following are low-end access servers: Cisco AS2511-RJ, Cisco AS2509-RJ, Cisco 2509, Cisco 2511, and the Cisco 2512.

[Figure 12](#) shows a Cisco AS2511-RJ access server. [Figure 13](#) shows a Cisco AS5300 access server. Notice that modems are either inside or outside the chassis, depending on the product model.

Figure 12 Cisco AS2511-RJ Access Server

14479

Figure 13 Cisco AS5300 Access Server

14480

Cisco IOS Modem Components

Different components inside Cisco IOS software work together to enable remote clients to dial in and send packets. [Figure 14](#) shows one Cisco AS5300 access server that is receiving calls from a remote office, branch office (ROBO); small office, home office (SOHO); and modem client.

Depending on your network scenario, you may encounter all of the components in [Figure 14](#). For example, you might decide to create a virtual IP subnet by using a loopback interface. This step saves address space. Virtual subnets can exist inside devices that you advertise to your backbone. In turn, IP packets get relayed to remote PCs, which route back to the central site.

The diagram illustrates a Cisco AS5300 configuration for ISDN and POTS services. A dashed box on the left indicates the software components: "Cisco IOS software inside a Cisco AS5300".

Legend:

- ▲ = ISDN B channel
- = Modem/POTS

Network Components and Connections:

- Headquarters intranet/Internet:** Connected to the **Routing and switching engine** via a **Fast Ethernet interface** and a **Loopback interface**.
- Virtual access interface:** A central hub for cloning templates.
 - Interface virtual template:** Cloned to the **Virtual access interface**.
 - Interface group-async:** Cloned to the **Virtual access interface**.
- Routing and switching engine:** Connected to the **Virtual access interface**, **Asynchronous interfaces**, **AAA**, and **Interface serial channels**.
- Asynchronous interfaces:** Consist of **Lines** and **Modems**.
- AAA (Authentication, Authorization, and Accounting):** A central triangle connecting the **Routing and switching engine**, **Interface serial channels**, and **TDM bus**.
- Interface serial channels S0:0, S0:1... (B channels):** Connected to the **Routing and switching engine** and **AAA**. A dashed arrow labeled "Cloning" points from the **Interface dialer controlling the D channels** to this interface.
- TDM bus:** Connected to **AAA** and **Controllers**.
- Controllers:** Connected to the **TDM bus** and **E1/T1 PRI ports**.
- E1/T1 PRI ports:** Connected to the **Controllers** and **PRI lines**.
- PRI lines:** Connected to the **E1/T1 PRI ports** and the **PSTN/ISDN** cloud.
- PSTN/ISDN:** A central cloud connecting to **POTS**, **BRI lines**, and **Remote PC**.
- POTS:** Connected to the **PSTN/ISDN** cloud and **Modem**.
- BRI lines:** Connected to the **PSTN/ISDN** cloud and **Cisco 1604 (ROBO)** and **Cisco 766 (SOHO)** routers.
- Remote PC:** Connected to the **Modem** and **POTS line**.

Logical Constructs in Modem Configurations

A logical construct stores core protocol characteristics to assign to physical interfaces. No data packets are forwarded to a logical construct. Cisco uses three types of logical constructs in its access servers and routers. These constructs are described in the following sections:

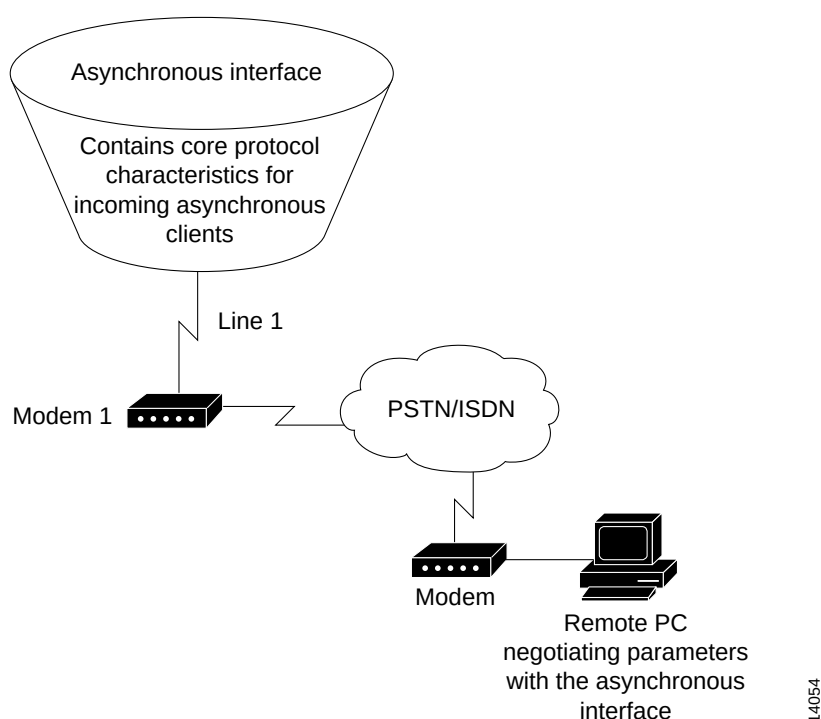
- [Asynchronous Interfaces](#)
- [Group Asynchronous Interfaces](#)
- [Modem Lines and Asynchronous Interfaces](#)

Asynchronous Interfaces

An asynchronous interface assigns network protocol characteristics to remote asynchronous clients that are dialing in through physical terminal lines and modems. (See [Figure 15](#).)

Use the **interface async** command to create and configure an asynchronous interface.

Figure 15 Logical Construct for an Asynchronous Interface



To enable clients to dial in, you must configure two asynchronous components: asynchronous lines and asynchronous interfaces. Asynchronous interfaces correspond to physical terminal lines. For example, asynchronous interface 1 corresponds to tty line 1.

Commands entered in asynchronous interface mode configure protocol-specific parameters for asynchronous interfaces, whereas commands entered in line configuration mode configure the physical aspects for the same port.

Specifically, you configure asynchronous interfaces to support PPP connections. An asynchronous interface on an access server or router can be configured to support the following functions:

- Network protocol support such as IP, Internet Protocol Exchange (IPX), or AppleTalk
- Encapsulation support such as PPP
- IP client addressing options (default or dynamic)
- IPX network addressing options
- PPP authentication
- ISDN BRI and PRI configuration

For additional information about configuring asynchronous interfaces, see the [“Overview of Dial Interfaces, Controllers, and Lines”](#) chapter.

Group Asynchronous Interfaces

A group asynchronous interface is a parent interface that stores core protocol characteristics and projects them to a specified range of asynchronous interfaces. Asynchronous interfaces clone protocol information from group asynchronous interfaces. No data packets arrive in a group asynchronous interface.

By setting up a group asynchronous interface, you also eliminate the need to repeatedly configure identical configuration information across several asynchronous interfaces. For example, on a Cisco AS5300 one group asynchronous interface is used instead of 96 individual asynchronous interfaces. (See [Figure 16.](#))

The following example shows a group asynchronous configuration for a Cisco AS5300 access server loaded with one 4-port ISDN PRI card and 96 MICA modems:

```
Router(config)# interface group-async 1
Router(config-if)# ip unnumbered loopback 0
Router(config-if)# encapsulation ppp
Router(config-if)# async mode interactive
Router(config-if)# peer default ip address pool dialin_pool
Router(config-if)# no cdp enable
Router(config-if)# ppp authentication chap pap dialin
Router(config-if)# group-range 1 96
```

To configure multiple asynchronous interfaces at the same time (with the same parameters), you can assign each asynchronous interface to a group and then configure the group. Configurations throughout this guide configure group asynchronous interfaces, rather than each interface separately.

If you want to configure different attributes on different asynchronous interfaces, do not assign them to the group or assign different interfaces to different groups. After assigning asynchronous interfaces to a group, you cannot configure these interfaces separately. For example, on a Cisco AS5300 access server in a T1 configuration, you could assign asynchronous interfaces 1 to 48 as part of one group (such as group-async1) and asynchronous interfaces 49 to 96 as part of another group (group-async2). You can also use the **member** command to perform a similar grouping function.

Modem Lines and Asynchronous Interfaces

Modems attach to asynchronous lines, which in turn attach to asynchronous interfaces. Depending on the type of access server you have, these components appear outside or inside the physical chassis.

Figure 16 shows the logical relationships among modems, asynchronous lines, asynchronous interfaces, and group asynchronous interfaces. All these components work together to deliver packets as follows:

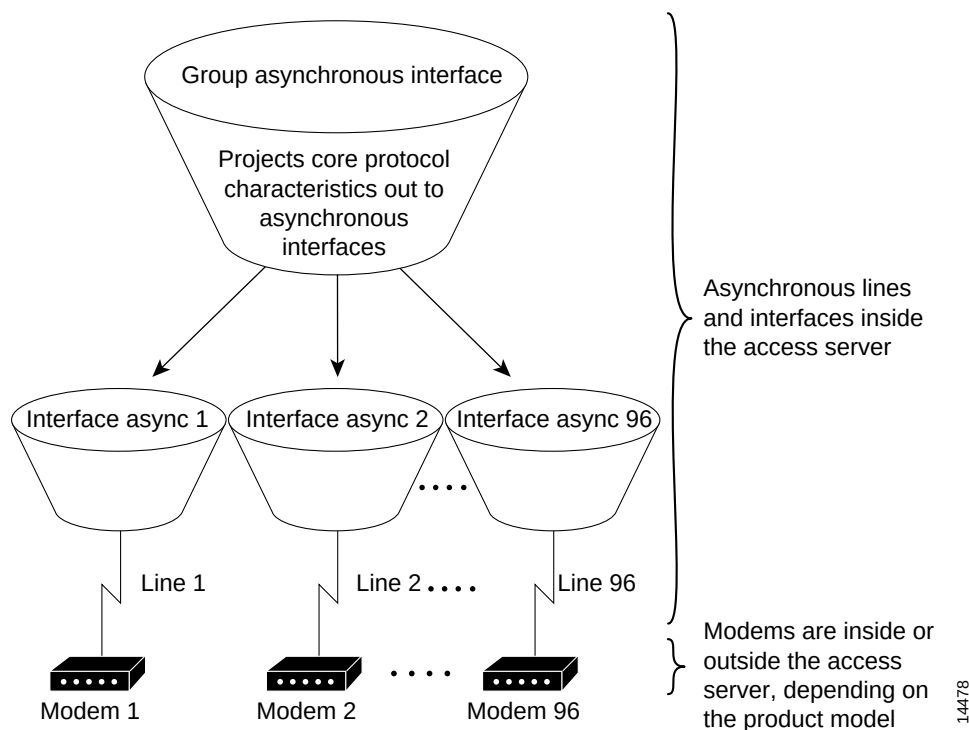
- Asynchronous calls come into the modems from the “plain old telephone service” (POTS) or Public Switched Telephone Network (PSTN).
- Modems pass packets up through asynchronous lines.
- Asynchronous interfaces clone their configuration information from group asynchronous interfaces.



Note

The number of interfaces and modems varies among access server product models.

Figure 16 *Modems, Lines, and Asynchronous Interfaces*



Use the **interface group-async** command to create and configure a group asynchronous interface. The following example shows a group asynchronous configuration for a Cisco AS5300 access server loaded with one 4-port ISDN PRI card and 96 MICA modems:

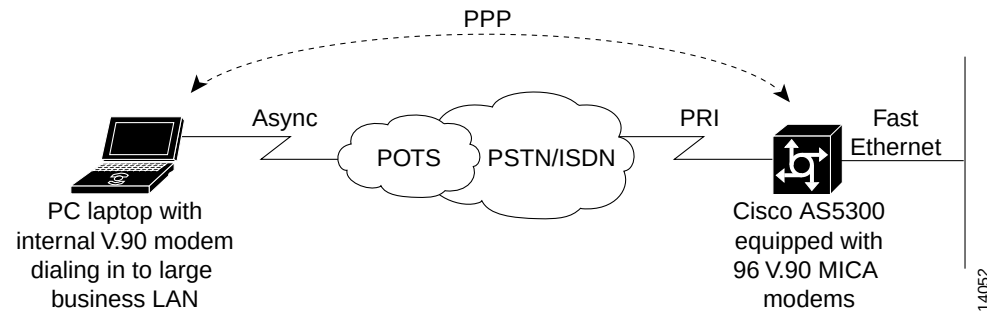
```
Router(config)# interface group-async 1
Router(config-if)# ip unnumbered loopback 0
Router(config-if)# encapsulation ppp
Router(config-if)# async mode interactive
Router(config-if)# peer default ip address pool dialin_pool
Router(config-if)# no cdp enable
Router(config-if)# ppp authentication chap pap dialin
Router(config-if)# group-range 1 96
```

Modem Calls

Modem calls travel through traditional telephone and ISDN lines. Regardless of the media used, these calls are initiated by a modem and terminate on another modem at the remote end.

Figure 17 shows a remote laptop using a V.90 internal modem to dial in to a Cisco AS5300 access server, which is loaded with 96 internal V.90 MICA technologies modems.

Figure 17 Remote Node Dialing In to a Cisco AS5300 Access Server



Asynchronous Line Configuration

Asynchronous line configuration commands configure ports for the following options:

- Physical layer options such as modem configuration
- Security for login in EXEC mode
- AppleTalk Remote Access (ARA) protocol configuration (PPP is configured in interface configuration mode)
- Autoselect to detect incoming protocols (ARA and PPP)

To enter line configuration mode, first connect to the console port of the access server and enter privileged EXEC mode. Then enter global configuration mode and finally enter line configuration mode for the asynchronous lines that you want to configure. The following example shows how you enter line configuration mode for lines 1 through 16:

```
Router> enable
Router# configure terminal
Router(config)# line 1 16
Router(config-line)#
```

Absolute Versus Relative Line Numbers

When you enter line configuration mode, you can specify an absolute line number or a relative line number. For example, absolute line number 20 is vty 2 (line 18 is vty 0). Referring to lines in a relative format is often easier than attempting to recall the absolute number of a line on a large system. Internally, the router uses absolute line numbers.

On all routers except the Cisco AS5350, AS5400, AS5800, AS5850 access servers, you can view all of the absolute and relative line numbers using the **show users all** EXEC command.

In the following sample display, absolute line numbers are listed at the far left. Relative line numbers are in the third column, after the line type. The second virtual terminal line, vty 1, is absolute line number 3. Compare the line numbers in this sample display to the output from the **show line** command.

Line	User	Host(s)	Idle	Location
0	con	0		
1	aux	0		
2	vtty	0	incoming	0 SERVER.COMPANY.COM
3	vtty	1		
4	vtty	2		
5	vtty	3		
6	vtty	4		

On the Cisco AS5350, AS5400, AS5800, AS5850 access servers, you can view the absolute and relative line numbers with the following commands:

- **show users all | exclude tty | interface** to show the non-internal modem lines
- **show controller async | include tty** to show the internal modem lines

The following example shows the information displayed with the **show users all | exclude tty | interface** command:

```
Router# show users all | exclude tty | interface
  Line      User      Host(s)      Idle      Location
*  0 con 0
  1 aux 0
  2 vty 0
  3 vty 1
  4 vty 2
  5 vty 3
  6 vty 4
```

The following example shows the information displayed with the **show controller async | include tty** command:

```
Router# show controller async | include tty
Controller information for Async2/00 (tty324)
Controller information for Async2/01 (tty325)
Controller information for Async2/02 (tty326)
.
.
.
```

Compare the line numbers in this sample display to the output from the **show line** command.

Line and Modem Numbering Issues

The tty line numbering scheme used by your access server or router is specific to your product and its hardware configuration. Refer to the product-specific documentation that came with your product for line numbering scheme information.

For example, the Cisco AS5200 access server has tty lines that map directly to integrated modems, as shown in [Table 5](#). Depending on the shelf, slot, and port physical architecture of the access server, the modem and tty line number schemes will change.

As shown in [Table 5](#), physical terminal lines 1 through 24 directly connect to modems 1/0 through 1/23, which are installed in the first chassis slot in this example. Physical terminal lines 25 through 48 directly connect to modems 2/0 through 2/23, which are installed in the second slot.

Table 5 *tty Lines Associated with Cisco AS5200 Modems*

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

Decimal TCP Port Numbers for Line Connections

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

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

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

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

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

```
line 10
  flowcontrol software
  no exec
```

A host that wants to send data to the printer would connect to the router on TCP port 4008, send the data, and then close the connection. (Remember that line number 10 octal equals 8 decimal.)

Signal and Flow Control Overview

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

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

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

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