CHAPTER 4

Configuring Asynchronous Connections with Modems

To successfully configure an asynchronous modem connection, the following must occur:

1. The modem itself must be configured to respond correctly to the telephone company circuit.
2. The physical aspects of the router link to the modem must be correctly defined to match the modem parameters.
3. The logical parameters must be established to provide a network-layer end-to-end connection.

The modem must be configured so that it understands the signalling on both the telephone-line side and the router-connection side. This information includes the line rate and the number of bits used for data and other physical settings for the modem. The particulars for the modem are discussed in the body of this chapter.

The second and third pieces of an asynchronous modem connection are configured on the router and provide both physical and logical aspects for a connection. The physical properties are configured on the *line*. These parameters include the line rate, the data link-layer protocols supported on the line, and so on. These parameters are needed for the router line to communicate with the attached modem.

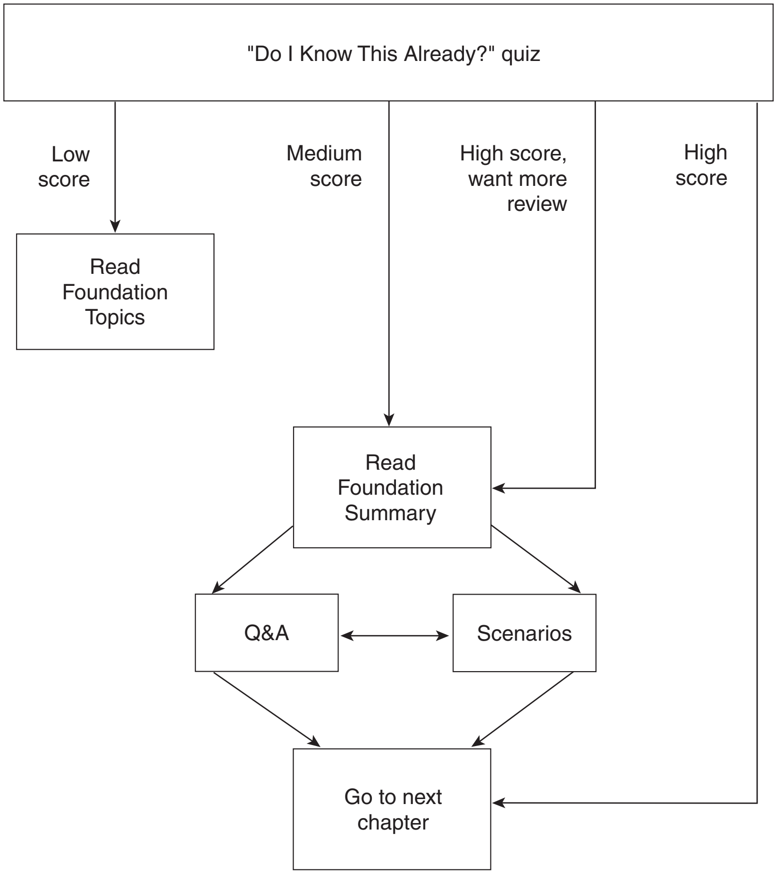
The last piece of an asynchronous modem connection is configuring the logical information on the router *interface*. The logical information includes the Layer 3 addresses, the network-layer protocol, the authentication methods, and so forth.

How to Best Use This Chapter

By taking the following steps, you can make better use of your study time:

- Keep your notes and answers for all your work with this book in one place for easy reference.
- Take the “Do I Know This Already?” quiz and write down your answers. Studies show retention is significantly increased through writing facts and concepts down, even if you never look at the information again.
- Use the diagram in Figure 4-1 to guide you to the next step.

Figure 4-1 *How to Use This Chapter*



"Do I Know This Already?" Quiz

The purpose of the "Do I Know This Already?" quiz is to help you decide what parts of this chapter to use. If you already intend to read the entire chapter, you do not necessarily need to answer these questions now.

The twelve-question quiz helps you determine how to spend your limited study time. The quiz is sectioned into smaller, two-question "quizlets," each of which corresponds to the six major topic headings in the chapter. Use the scoresheet in Table 4-1 to record your scores.

Table 4-1 *Scoresheet for Quizlets and Quiz*

Quizlet Number	Foundation Topics Section Covered by These Questions	Questions	Score
1	Modem Signaling	1–2	
2	Modem Configuration Using Reverse Telnet	3–4	
3	Router Line Numbering	5–6	
4	Basic Asynchronous Configuration	7–8	
5	Configuration of the Attached Modem	9–10	
6	Chat Scripts to Control Modem Connections	11–12	
All questions		1–12	

1 What pins are used for modem control?

2 What is the standard for DCE/DTE signaling?

3 In character mode using reverse Telnet, what is the command to connect to the first async port on a 2509 router that has a loopback interface of 192.168.1.1?

4 What port range is reserved for accessing an individual port using binary mode?

5 If a four-port serial (A/S) module is in the second slot on a 3640 router, what are the line numbers for each port?

6 What is the AUX port line number on a 3620 series router?

7 What does the **physical-line async** command do and on what interfaces would you apply it?

8 In what configuration mode must you be to configure the physical properties of an asynchronous interface?

9 When should **modem autoconfigure discovery** be used? What happens when you use it?

10 Which of the following commands would you use to add an entry to a modemcap database called newmodem?

a. **edit modemcap newmodem**

b. **modemcap edit newmodem**

c. **modemcap edit type newmodem**

d. **modemcap add newmodem**

11 List four reasons why you would use a chat script.

____

____

____

____

12 Which of the following would trigger a chat script start?

a. Line reset

b. DDR

c. Line activation

d. Manual

The answers to the "Do I Know This Already?" quiz are found in Appendix A, "Answers to the 'Do I Know This Already?' Quizzes and Q&A," on page 397. The suggested choices for your next step are as follows:

- **6 or fewer overall score**—Read the chapter. This includes the "Foundation Topics," the "Foundation Summary," Q&A, and scenarios at the end of the chapter.
- **7, 8, or 9 overall score**—Begin with the "Foundation Summary," then go to the Q&A and scenarios at the end of the chapter.
- **10 or more overall score**—If you want more review on these topics, skip to the "Foundation Summary," then go to the Q&A and scenarios at the end of the chapter. Otherwise, move to the next chapter.

Foundation Topics

Modem Signaling

This chapter covers the signaling of the modem and the configurations for a Remote Access Server (RAS) connection. The successful CCNP or CCDP candidate should be able to describe the signaling and pins used by the cabling and not just the syntax that is required for the connection. The signaling is just as important because it provides the basis for the physical-layer troubleshooting that can be needed to establish a connection.

Asynchronous data communications technology occurs when an end device, such as a PC, calls another end device, such as a server, to exchange data. In asynchronous data communications, end devices are called data terminal equipment (DTE). These devices communicate through data circuit-terminating equipment (DCE). DCE devices clock the flow of information. In our case, the modem provides the DCE function to the PC and server.

The Electronic Industries Association/Telecommunications Industry Association (EIA/TIA) defines a standard for the interface between DCE and DTE devices. This standard is the EIA/TIA-232 and was previously referred to as the RS-232-C standard (where the RS stood for "recommended standard").

It is unwise to think of a PC-to-server connection that uses asynchronous communications as a single circuit. The PC using a modem is one DTE to DCE path end. The far end DCE to DTE (modem to server) is another path. Each DTE–DCE or DCE–DTE connection must be made prior to data transfer.

With asynchronous communication, eight pins are used in a DB25 to transfer data and control the modem, as listed in Table 4-2. The table shows the pins and their definitions. As you read the table, note the direction of the signal and whether DCE or DTE controls or signals on the pin.

Table 4-2 *Standard EIA/TIA-232 Definitions and Codes*

Pin Number	Designation	Definition	Description
2	TD	Transmits data	DTE-to-DCE data transfer
3	RD	Receives data	DCE-to-DTE data transfer
4	RTS	Request to send	DTE signal buffer available
5	CTS	Clear to send	DCE signal buffer available
6	DSR	Data set ready	DCE is ready.
7	GRD	Signal ground	
8	CD	Carrier detect	DCE senses carrier.
20	DTR	Data terminal ready	DTE is ready.

Pins 2, 3, and 7 enable data transfer, pins 4 and 5 enable flow control of data, and pins 6, 8, and 20 provide modem control.

Data Transfer

The pins used for data transfer are pin 2, 3, and 7. The DTE device raises the voltage on the RTS when it has buffer space available to receive from the DCE device. Once a call is established and the DTE device sees the DCE raise the voltage on the CTS, the DTE device transmits data on pin 2. Conversely, the DTE device will raise the voltage on the RTS when it has buffer space available to receive from the DCE device. The need for the ground pin is such that a positive or negative voltage can be discerned.

Data Flow Control

The RTS pin and the CTS pin control the flow of information. The DTE device controls the RTS pin (as shown in Tabel 4-2), which, when seen by the DCE, alerts the DCE that it can receive data. It might help you to think of the RTS as the ready-to-receive pin. The DCE device controls the CTS pin, which in turn signals the DTE that it has buffer available. These definitions are critical to a CCNP or CCDP candidate.

Modem Control

DSR and DTR are signal pins used to control how the modem operates. The DSR pin is raised when the modem is powered on. This raising lets the DTE device know that the modem is ready for use. The DTR pin is raised when the DTE device is powered and ready to receive information from the DCE.

In most cases, when the DTE device is powered on, the DTR pin is raised; however, there are cases in which the DTR pin is raised only if a software package begins to run. This might sound like a minor point, but when you are troubleshooting, it is important to know if the DTE has signaled the modem that it is ready. In fact, just because the PC is on does not necessarily mean that DTR is asserted, and whether your DTE device raises the DTR when powering up or when you turn on your communication software, DTR is needed for a two-way conversation between the DCE and DTE device.

Note that the CD pin is also a signal pin. When two DCE devices establish a connection, the CD pin is asserted to indicate that a carrier signal has been established between the DCE devices. Note also that because two devices constitute the DTE (PC) and DCE (modem) connection, either must be allowed to terminate the connection.

DTE Call Termination

When the DTE is ready to terminate the connection because the user has completed the call and signaled the PC to go back on-hook, the DTR is dropped. For this to happen, the modem must be configured to interpret the loss of the DTR as the end of a conversation. When the DTE drops the DTR, the modem is alerted that the carrier is no longer needed.

This configuration is done when the modem is first installed. This can be manually done for each call, or it can be scripted in a chat script that is sent to the modem each time a call is terminated. Each time a call is terminated, the router resets (rescripts) the modem. This low level configuration is done on the modem to prepare the modem for reuse. In many cases, accepting the default configuration for a modem allows it to function properly.

Even accepting the default configuration provides a "configuration" to the modem. The details of each modem parameter are discussed in the section, "Configuration of an Attached Modem," later in this chapter.

DCE Call Termination

If a far-end modem drops the CD because the remote DTE has ended the transmission, the near-end modem must signal the near-end DTE that the transmission has been terminated. The modem must be programmed to understand and signal this termination. In other words, the modem must be told how to handle the loss of carrier detection. By default, most modems understand that this signal loss is an indication that the call is to be terminated. However, it is a configuration parameter that the modem must understand.

Modem Configuration Using Reverse Telnet

In order to configure a modem, a router must be set up to talk to it. Cisco refers to this as a *reverse Telnet connection*. A host that is connected to a router can Telnet to a Cisco reserved port address on the router and establish an 8-N-1 connection to a specific asynchronous port. An *8-N-1* connection declares the physical signaling characteristics for a line.

Table 4-3 shows reserved port addresses. The router must have a valid IP address on an interface and an asynchronous port. To establish a connection to the modem connected to the asynchronous port, you can Telnet to any valid IP address on the router and declare the Cisco reserved port number for the asynchronous interface. You can do this only, however, from the router console or a remote device that has Telnet access to the router.

Most modem consoles operate using eight data bits, zero parity bits, and one stop bit. In addition, the use of reverse Telnet enables the administrator to configure locally attached devices. For example, suppose you want to set up an 8-N-1 connection to the first asynchronous interface on a router, which has the 123.123.123.123 address assigned to its E0 port. To connect in character mode using Telnet, you would issue the following command:

```
telnet 123.123.123.123 2001
```

where **123.123.123.123** is the router's E0 port and **2001** is the Cisco reserved port number for the first asynchronous port on the router. Table 4-3 shows the Cisco reserved port numbers for all port ranges.

Table 4-3 *Reverse Telnet Cisco Reserved Port Numbers*

Connection Service	Reserved Port Range For Individual Ports	Reserved Port Range For Rotary Groups
Telnet (character mode)	2000–2*xxx*	3000–3*xxx*
TCP (line mode)	4000–4*xxx*	5000–5*xxx*
Telnet (binary mode)	6000–6*xxx*	7000–7*xxx*
Xremote	9000–9*xxx*	10000–10*xxx*

The use of the rotary group reserved port number connects to the first available port that is in the designated rotary group. If a specific individual port is desired, the numbers from the first column of Table 4-3 are used.

You can establish a session with an attached modem using reverse Telnet and the standard **AT** command set (listed later in Table 4-4) to set the modem configuration. This, however, is the hard way because once a modem connection has been established using reverse Telnet, you must disconnect from the line for the modem to be usable again. In addition, to exit the connection, you would have to press Ctrl+Shift+6 and then x to suspend the session, and then issue the **disconnect** command from the router prompt. It is important to remember this simple sequence because the modem does not understand the **exit** command as does a router!

Router Line Numbering

The line numbers on a router are obtained in a methodical manner. The console port is line 0. Each asynchronous (TTY) port is then numbered 1 through the number of TTY ports on the router. The auxiliary port is given the line number LAST TTY + 1, and the virtual terminal (vty) ports are numbered starting at LAST TTY + 2.

Example 4-1 has the **show line** output for a Cisco 2511 router, which has eight asynchronous ports available. Notice that the AUX port is labeled in line 17 and the vty ports are labeled in lines 18–22.

Example 4-1 **show line** *Output for Cisco 2511 Router*

```
2511Router>show line
 Tty Typ     Tx/Rx    A Modem  Roty AccO AccI   Uses   Noise  Overruns   Int
*  0 CTY              -    -      -    -    -      0       1      0/0      -
*  1 TTY   9600/9600  -    -      -    -    -      7      23      0/0      -
*  2 TTY   9600/9600  -    -      -    -    -      5       1      0/0      -
*  3 TTY   9600/9600  -    -      -    -    -     14      63      0/0      -
*  4 TTY   9600/9600  -    -      -    -    -      4       3      0/0      -
```

continues

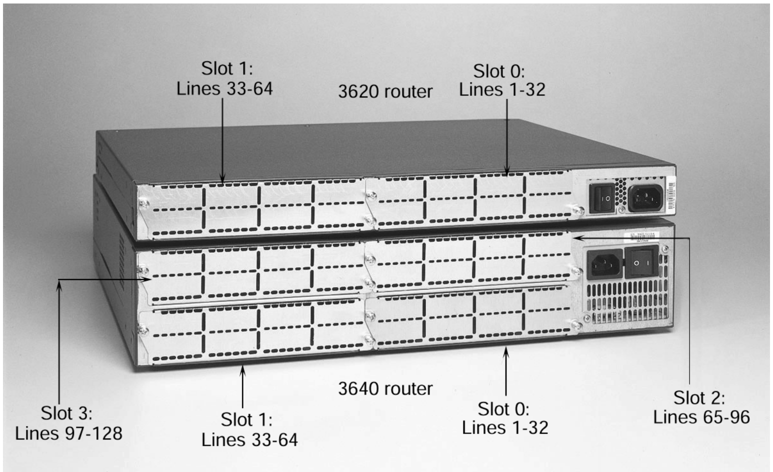
Example 4-1 **show line** *Output for Cisco 2511 Router (Continued)*

```
*  5 TTY   9600/9600  -    -      -    -    -     16      6     0/0        -
*  6 TTY   9600/9600  -    -      -    -    -     12      7     0/0        -
   7 TTY   9600/9600  -    -      -    -    -      3      1     0/0        -
   8 TTY   9600/9600  -    -      -    -    -      0      9     0/0        -
*  9 TTY   9600/9600  -    -      -    -    -     12      0     0/0        -
* 10 TTY   9600/9600  -    -      -    -    -     16      0     0/0        -
* 11 TTY   9600/9600  -    -      -    -    -     25      2     0/0        -
* 12 TTY   9600/9600  -    -      -    -    -      5      0     0/0        -
* 13 TTY   9600/9600  -    -      -    -    -      0      0     0/0        -
  14 TTY   9600/9600  -    -      -    -    -      0      2     0/0        -
  15 TTY   9600/9600  -    -      -    -    -      0      0     0/0        -
  16 TTY   9600/9600  -    -      -    -    -      3      0     0/0        -
  17 AUX   9600/9600  -    -      -    -    -      0      0     0/0        -
  18 VTY              -    -      -    -    -      0      0     0/0        -
  19 VTY              -    -      -    -    -      0      0     0/0        -
  20 VTY              -    -      -    -    -      0      0     0/0        -
  21 VTY              -    -      -    -    -      0      0     0/0        -
  22 VTY              -    -      -    -    -      0      0     0/0        -
```

The numbering scheme for interfaces was expanded for the 3600 series routers. The console is still line 0 and the vty ports are similarly counted after the TTYs. However, Cisco chose to use reserved numbering for the available slots. Thus, slot 0 has reserved lines 1–32, slot 1 has reserved lines 33–64, slot 2 has reserved lines 65–97, and so on. Each slot is given a range of 32 line numbers, whether they are used or not.

Figure 4-2 shows the rear of the chassis for a 3620 and 3640 router and the line numbers associated with each slot.

Figure 4-2 *Line Numbers for 3620 and 3640 Routers*



The line-numbering scheme is important when configuring a router. In the case of the 3600 and 2600 routers with the new modular interfaces, the line numbers are based on the slot that the feature card is in. For illustration, consider the output in Example 4-2, which is from a 3640 series router with a modem card in slot 2. Notice that the line numbers for the internal modems are 65–70 because only one MICA card is installed in the slot.

Example 4-2 **show line** *Output from a 3640 Series Router with a Modem Card in Slot 2*

```
router#show line
 Tty Typ     Tx/Rx    A Modem  Roty AccO AccI   Uses   Noise  Overruns   Int
*  0 CTY              -    -      -    -    -      0       0     0/0       -
I 65 TTY              - inout     -    -    -      0       0     0/0       -
I 66 TTY              - inout     -    -    -      0       0     0/0       -
I 67 TTY              - inout     -    -    -      0       0     0/0       -
I 68 TTY              - inout     -    -    -      0       0     0/0       -
I 69 TTY              - inout     -    -    -      0       0     0/0       -
I 70 TTY              - inout     -    -    -      0       0     0/0       -
I 97 TTY 115200/115200- inout     -    -    -      0       0     0/0     Se3/0
*129 AUX   9600/9600  -    -      -    -    -      0       0     0/0       -
 130 VTY              -    -      -    -    -      0       0     0/0       -
 131 VTY              -    -      -    -    -      0       0     0/0       -
 132 VTY              -    -      -    -    -      0       0     0/0       -
 133 VTY              -    -      -    -    -      0       0     0/0       -
 134 VTY              -    -      -    -    -      0       0     0/0       -
The following lines are not in asynchronous mode or are without hardware support:
1-64, 71-96, and 98-128.
```

To properly configure a router, you must know the association between the line and interface numbers. The AUX port on the modular routers is the last line number, which would be the number of slots multiplied by 32, plus 1. In the case of the 3640 router shown in Example 4-2, the AUX port number is 129, and the vty ports are 130–134 by default.

In Example 4-3, the configuration for a 3640 router has physical characteristics configured on line 97 for the asynchronous interface in slot 3/0. The remaining IOS commands are discussed in detail later in this chapter, but are presented here for completeness.

Example 4-3 *3640 Router Configuration*

```
interface Serial3/0
 physical-layer async
 ip unnumbered Ethernet0/0
 no ip directed-broadcast
 encapsulation ppp
 async mode interactive
 peer default ip address pool TESTPOOL
 no cdp enable
 ppp authentication chap
!
line 97
 password cisco
```

continues

Example 4-3 *3640 Router Configuration (Continued)*

```
 autoselect during-login
 autoselect ppp
 login local
 modem InOut
 transport input all
 stopbits 1
 speed 115200
 flowcontrol hardware
line aux 0
line vty 0 4
 login local
!
```

Basic Asynchronous Configuration

To configure the modem (the DCE) from the router (the DTE), you must set up the logical and physical parameters for the connection. The logical parameters include the protocol addressing, the authentication method, and the encapsulation, all of which are configured on the asynchronous interface. The physical configuration is done on the line. The physical parameters include the flow control, the DTE-DCE speed, and the login request. It is important for the successful CCNP or CCDP to be aware of the command mode needed for configuration.

The configuration in Example 4-4 demonstrates which commands are used on each line or interface.

Example 4-4 *Configuration for a Serial Interface in Asynchronous Mode*

```
interface Serial3/0     !logical parameters go on the interface
    physical-layer async
     ip unnumbered Ethernet0/0
      no ip directed-broadcast
      encapsulation ppp
      async mode interactive
      peer default ip address pool remaddpool
      no cdp enable
      ppp authentication chap
line 97   !physical parameters go on the line
      autoselect during-login
      autoselect ppp
      login
      modem InOut
      modem autoconfigure type usr_sportster
      transport input all
      stopbits 1
      rxspeed 115200
```

Example 4-4 shows the distinction between the physical and logical parameters and where they are defined in the router configuration file.

Three types of router interfaces can be configured for serial communication:

- Asynchronous interfaces
- Synchronous/asynchronous interfaces (A/S)
- Synchronous interfaces

Router interfaces that are synchronous only cannot be used for modem or asynchronous communication. On the router models with A/S ports, the serial ports default to synchronous, and the interface must be declared for asynchronous usage using the **physical-layer async** command.

The configuration in Example 4-4 is for the first (port 0) synchronous/asynchronous interface on a four-port A/S card in the third slot of a 3600. The **physical-layer async** is needed because this device has A/S ports. Hence, the **physical-layer async** command is entered at the **router(config-if)#** prompt for Serial 3/0. On the other hand, in the case of those routers that have ports designated as asynchronous, only the **physical-layer async** command is not used.

Logical Considerations on the Router

Logical considerations are configured on the interface of the router. These include the network-layer addressing, the encapsulation method, the authentication, and so on. The configuration in Example 4-5 is for a serial interface that is used to receive an inbound call.

Example 4-5 *Router Configuration for Serial Interface Receiving Inbound Calls*

```
interface Serial2
  physical-layer async
  ip unnumbered Ethernet0
  ip tcp header-compression passive
   encapsulation ppp
   bandwidth 38
   async mode interactive
   peer default ip address pool remaddpool
   no cdp enable
   ppp authentication chap
```

In Example 4-5, the **physical-layer async** command places the serial 2 interface in asynchronous mode. Once this command is issued, the router treats the interface as an asynchronous port. This can be done on ONLY those interfaces that are defined as A/S.

The **ip unnumbered Ethernet0** command declares that the interface assume the address of the E0 interface. This enables the saving of IP addresses but makes the interface non-SNMP manageable. This command could be replaced with the desired IP address of the interface (refer to the discussion in this section that covers **ip address pool**). Note that it is quite common for a large number of asynchronous interfaces to a common physical interface to be unnumbered and to use an address pool to assign the network-layer addresses to the dial-up users.

The **ip tcp header-compression passive** command states that if the other DCE device sends packets with header-compression, the interface understands and sends in kind but does not initiate the compression.

The **encapsulation ppp** command declares the encapsulation method for the interface.

The **bandwidth 38** command tells the routing protocol and the router (for statistics) the speed of the line. This command has no affect on the actual negotiated speed of the modem or the speed at which the DTE talks to the modem.

The **async mode interactive** command enables, once a connection is made, the dial-up user access to the EXEC prompt.

The **peer default ip address pool remaddpool** command specifies that the IP address assigned to the dial-up user be from the address grouping or pool defined by the label **remaddpool**. The syntax for the pool definition, defined in global configuration mode, is as follows:

```
ip local pool remaddpool low-ip-pool-address high-ip-pool-address.
```

A unique address from the pool of addresses is given to a dial-up user for the duration of the session. The address is returned to the pool when the dial-up user disconnects the session. In this fashion, it is not necessary to associate an IP address with each asynchronous interface. Each asynchronous interface to another interface on the router is unnumbered and the pool is created from part of that interface's subnet. For more information and examples on the use of address pools and unnumbering, refer to Chapter 6, "Using ISDN and DDR Technologies."

The **no cdp enable** command turns off the Cisco Discovery Protocol for the interface. By default, this protocol is on, and because the interface is likely connected to a dial-up user who does not understand CDP, the bandwidth it would use is saved.

The **ppp authentication chap** command specifies that the Challenge Handshake Authentication Protocol (CHAP) be used on this link. Failure of the client to honor CHAP results in the link not being established.

Physical Considerations on the Router

Physical characteristics are configured in line mode. These include the speed, the direction of the call, modem setup, and so on. Example 4-6 shows a configuration used to connect to a USR Sportster modem on physical line 2.

Example 4-6 *Router Configuration Connecting USR Sportster Modem on Physical Line 2*

```
line 2
      autoselect during-login
      autoselect ppp
      login local
      modem InOut
      modem autoconfigure type usr_sportster
      transport input all
      stopbits 1
      rxspeed 115200
      txspeed 115200
      flowcontrol hardware
```

The **login local** command is the same for this line as it is for the console and AUX ports. The **Login local** command tells the physical line to request a username/password pair when a connection is made and to look locally on the router for a matching **username** *xxxx* **password** *yyyy* pair that has been configured in global mode (*xxxx* and *yyyy* represent a freely chosen username and password combination).

The **autoselect during-login** and **autoselect ppp** commands automatically start the PPP protocol and issue a carriage return so that the user is prompted for the login. This feature became available in IOS Software Release 11.0. Prior to this "during-login" feature, the dial-up user was required to issue an exec command or press the Enter key to start the session.

The **modem InOut** command enables both incoming and outgoing calls. The alternative to this command is the default **no modem inout** command, which yields no control over the modem.

The **modem autoconfigure type usr_sportster** command uses the **modemcap database usr_sportster** entry to initialize the modem. We further discuss this initialization later in the chapter.

The **transport input all** command enables the processing of any protocols on the line. This command defines which protocols to use to connect to a line. The default command prior to 11.1 was **all**; the default with 11.1 is **none**.

In the router configuration, the number of **stopbits** must be the same for both communicating DCE devices. Remember that the physical-layer parameters must match for the physical layer to be established. Failure to do so prevents the upper layers from beginning negotiation.

In Example 4-6, **rxspeed** and **txspeed** are shown as separate commands. The **speed** command**,** however, sets both transmit and receive speeds and locks the speed between the modem and the DTE device. Failure to lock or control the DTE-to-DCE speed allows the speed of local communication to vary with the line speed negotiated between the DCE devices. This limits the capability of the DTE-to-DCE flow control.

The **flowcontrol hardware** command specifies that the RTS and CTS be honored for flow control.

Example 4-6 provides the basic configuration for an asynchronous line. Once the DTE device has been configured, you must set the DCE device to communicate with the modem by using the AT commands.

Configuration of the Attached Modem

In the early modem days, the Hayes command set was the de facto standard; however, there was never a ratified industry command set. Today, rather than converging to a general standard, the modem industry has actually diverged. Nonetheless, the AT commands documented in Table 4-4 are considered "standard" and should work on most modems.

Table 4-4 *Standard* AT *Commands*

COMMAND	Result
AT&F	Loads factory default settings
ATS0=n	Auto answers
AT&C1	CD reflects the line state
AT&D2	Hangs up on low DTR
ATE0	Turns off local echo
ATM0	Turns off the speaker

A CCNP or CCDP should be familiar with these commands. For many modems on the market today, commands not in this table are used to configure the modem fall into the category of not standard.

The correct initialization string must be sent to the modem for proper operation. You can do this by using a chat script or the **modem autoconfigure** command. The former method is the most common.

Modem Autoconfiguration and the Modem Capabilities Database

Modem autoconfiguration is a Cisco IOS software feature that enables the router to issue the modem configuration commands, which frees the administrator from creating and maintaining scripts for each modem. The general syntax for modem autoconfiguration is as follows:

```
modem autoconfigure [discovery | type modemcap-entry-name]
```

The two command options for the **modem autoconfigure** command are as follows:

- **type**—This option configures modems without using modem commands, or so it is implied. The **type** argument declares the modem type that is defined in the modem capabilities database so that that the administrator does not have to create the modem commands.

- **discovery**—Autodiscover modem also uses the modem capabilities database, but in the case of **discover**, it tries each modem type in the database as it looks for the proper response to its query.

As you can see, the **modem autoconfigure** command relies on the modem capabilities database, also known as the *modemcap*. The modem capabilities database has a listing of modems and a generic initialization string for the modem type. The discovery of a modem using the **autoconfigure** feature uses the initialization strings from each modem in the modem capabilities database to discover the installed modem. If the modem is not in the database, it fails, and the administrator has to manually add the modem to the database.

The use of the discovery feature is not recommended because of the overhead on the router. Each time the line is reset, the modem is rediscovered. However, the discovery feature can be used to initially learn the modem type if you are not geographically near the router and cannot gather the information any other way. After discovery has taken place, the administrator should use the **type** option to specify the entry in the modem capabilities database to use.

To discover a modem, the syntax would be as follows:

```
modem autoconfigure discovery
```

Again, once the modem type is determined, the final configuration for the router interface should be as follows:

```
modem autoconfigure type entry_name_from_modemcap
```

This configuration eliminates unnecessary overhead on the router.

Use the **show modemcap** command to see the entries in the modemcap database. Example 4-7 demonstrates the output from the **show modemcap** command.

Example 4-7 **show modemcap** *Command Output Reveals Modemcap Database Entries*

```
BCRANrouter#show modemcap

default
codex_3260
usr_courier
usr_sportster
hayes_optima
global_village
viva
telebit_t3000
microcom_hdms
microcom_server
nec_v34
nec_v110
nec_piafs
cisco_v110
mica
```

To view the detailed settings for a particular entry in the modem capabilities database, the entry name is added as an argument to the **show modemcap** command. The database has most models of modems. If your entry is not in the database, it can be added by editing the database.

Editing the database requires creating your own entry name and specifying the AT commands for the initialization string. This must be done for any modem that is not in the database. This might sound time-consuming or tedious, but it has to be done only once. The added information to the database is stored in NVRAM as part of the router configuration and can be copied to other routers that have the same modems.

Common practice dictates that multiple modem types not be used at a single RAS facility. Instead, the administrator should use a single modem type and maintain spares of that particular type so that constant manipulation of the modem capabilities database is not necessary.

Let's take a look at how a modem is added to the database. If an attached modem is a Viva plus that is not listed in the database, but another Viva modem is in the database, you could create a new entry and name it whatever you want. The AT commands that are unique to the Viva plus modem would be added to the local configuration in NVRAM and the additional AT commands that are the same for all Viva modems would be obtained from the database.

To add the modem, you would use the following global commands:

```
modemcap edit viva_plus speed &B1
modemcap edit viva_plus autoanswer s0=2
modemcap edit viva_plus template viva
```

These commands use the initialization string from the entry **viva** and enable the administrator to alter the newly created **viva_plus**. All changes and additions to the modemcap are stored in the configuration file for the router. Because of this, Cisco can add to the modemcap at any release because the local NVRAM changes override the modemcap.

The overview of all this is that you bought some modems that you, as the administrator, feel are the best for your application. The modemcap database may, or may not, have these particular modems defined. If the modem is defined in the modemcap then you can simply use the **type** option to the **modem autoconfigure** command. If the modem is not in the database then it must be added. Once it is added, all future modem connections on this router can simply point to the added entry.

Chat Scripts to Control Modem Connections

Chat scripts enable us to talk to or through a modem to a remote system using whatever character strings or syntax is needed. A chat script takes the form of

Expect-string - send-string - expect-string - send-string

where the *expect* strings are character strings sent from or through the modem to the DTE device and the *send* strings are character strings sent from the DTE device to or through the modem.

Reasons for Using a Chat Script

As a CCNP or CCDP, you should be aware that chat scripts are used for the following goals:

- **Initialization**—To initialize the modem
- **Dial string**—To provide the modem with a dial string
- **Logon**—To log in to a remote system
- **Command execution**—To execute a set of commands on a remote system

Reasons for a Chat Script Starting

A chat script can be manually started on a line using the **start-chap** command; they can also be configured to start for the following events:

- **Line activation**—CD trigger (incoming traffic)
- **Line connection**—DTR trigger (outgoing traffic)
- **Line reset**—Asynchronous line reset
- **Startup of an active call**—Access server trigger
- **Dialer startup**—From a dial-on-demand trigger

Using a Chat Script

The primary use of a chat script is to provide the dial number for the connection. The following line shows an example of this chat script:

```
Router(config)#chat-script REMDEVICE ABORT ERROR ABORT BUSY "" "ATZ" OK "ATDT \T"
TIMEOUT 30 CONNECT \c
```

Care should be taken with the character case used in this command. **ABORT ERROR** and **ABORT BUSY** cause the modem to abort if it sees **ERROR** or **BUSY**. Both arguments might be easier understood if read as "abort if you see ERROR" and "abort if you see BUSY," respectively. If **error** or **abort** are entered in lowercase, the modem never sees these conditions because its search is case-sensitive. The **\T** inserts the called number from the **dial string** or **map** command into the chat script. A **\t** causes the script to look for a "table character"; hence, case is important here as well.

NOTE Detailed information on the **dial string** and **map** commands are provided in Chapter 6.

The **REMDEVICE** chat script has been configured to drop the connection if the modem declares a busy or error condition. If no busy or error condition is declared, the router does not wait for anything except string = " ". The router then issues the **ATZ**, or modem reset, command, using a send string. The router waits for the modem to respond OK, which is the normal modem response to **ATZ**. The router then sends the **ATDT** command and replaces the **\T** with the phone number to make the call. Last, the **TIMEOUT 30** declares that the call is considered "not answered" if no carrier is obtained in 30 seconds. Once the connection is made, the chat script sends a **c**, which is a carriage return.

Provided that the router, the modem, and the phone number are correct, the physical layer should now be established! Congratulations! You can now move on to the upper layer protocols, such as PPP (see Chapter 5, "Configuring PPP and Controlling Network Access") and advanced uses (see Chapter 6).

Foundation Summary

The Foundation Summary is a collection of tables and figures that provides a convenient review of many key concepts in this chapter. For those of you already comfortable with the topics in this chapter, this summary could help you recall a few details. For those of you who just read this chapter, this review should help solidify some key facts. For any of you doing your final preparation before the exam, these tables and figures will hopefully be a convenient way to review the day before the exam.

Table 4-5 *Standard EIA/TIA-232 Definitions and Codes*

Pin Number	Designation	Definition	Description
2	TD	Transmits data	DTE-to-DCE data transfer
3	RD	Receives data	DCE-to-DTE data transfer
4	RTS	Request to send	DTE signal buffer available
5	CTS	Clear to send	DCE signal buffer available
6	DSR	Data set ready	DCE is ready.
7	GRD	Signal ground	
8	CD	Carrier detect	DCE senses carrier.
20	DTR	Data terminal ready	DTE is ready.

Table 4-6 *Cisco Reserved Port Numbers Used with Reverse Telnet*

Connection Service	Reserved Port Range for Individual Ports	Reserved Port Range for Rotary Groups
Telnet (character mode)	2000–2*xxx*	3000–3*xxx*
TCP (line mode)	4000–4*xxx*	5000–5*xxx*
Telnet (binary mode)	6000–6*xxx*	7000–7*xxx*
Xremote	9000–9*xxx*	10000–10*xxx*

Figure 4-3 *3600 Line Numbers*

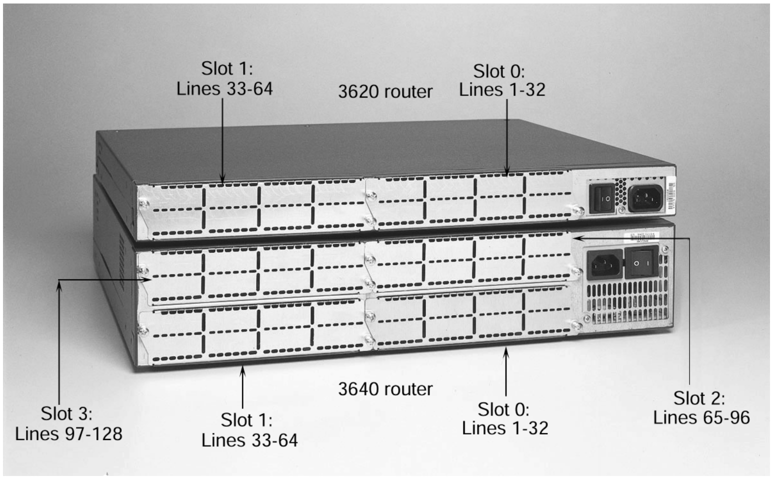


Table 4-7 **modem autoconfigure** *Commands*

Command	What It Does
modem autoconfigure discovery	Discovers the modem
modem autoconfigure type *entry_name_from_modemcap*	Creates the final configuration for the router interface, which eliminates unnecessary overhead on the router
show modemcap	Displays the entries in the modemcap database

Table 4-8 *Standard* AT *Commands*

Command	Result
AT&F	Loads factory default settings
ATS0=n	Auto answers
AT&C1	CD reflects the line state
AT&D2	Hangs up on low DTR
ATE0	Turns off local echo
ATM0	Turns off the speaker

Reasons for using a chat script:

- **Initialization**—To initialize the modem
- **Dial string**—To provide the modem with a dial string
- **Logon**—To log in to a remote system
- **Command Execution**—To execute a set of commands on a remote system

A chat script can be manually started on a line using the **start-chap** command; they can also be configured to start for the following events:

- **Line activation**—CD trigger (incoming traffic)
- **Line connection**—DTR trigger (outgoing traffic)
- **Line reset**—Asynchronous line reset
- **Startup of an active call**—Access server trigger
- **Dialer startup**—From a dial-on-demand trigger

Q&A

The questions and scenarios in this book are more difficult than what you will experience on the actual exam. The questions do not attempt to cover more breadth or depth than the exam; however, they are designed to make sure that you know the answer. Rather than enabling you to derive the answer from clues hidden inside the question itself, the questions challenge your understanding and recall of the subject.

Questions from the "Do I Know This Already?" quiz from the beginning of the chapter are repeated here to ensure that you have mastered the chapter's topic areas. Hopefully, mastering these questions will help you limit the number of exam questions on which you narrow your choices to two options and then guess.

The answers to these questions can be found in Appendix A, on page 397.

1 What pins are used for modem control?

2 What is the standard for DCE/DTE signaling?

3 If the user wants to terminate a call, what pin does the DTE device drop to signal the modem?

4 What must be done to terminate a reverse Telnet session with an attached modem?

5 In character mode using reverse Telnet, what is the command to connect to the first async port on a 2509 router that has a loopback interface of 192.168.1.1?

____

____

____

6 Which interface is line 97 on a 3640 series router?

a. S 0/97

b. S 3/1

c. S 2/1

d. S 097

7 What port range is reserved for accessing an individual port using binary mode?

____

____

____

8 When flow control is enabled, which pins are used?

____

____

____

9 If a four-port serial (A/S) module is in the second slot on a 3640 router, what are the line numbers for each port?

____

____

____

10 What is the **AT** command to return a router to factory default settings?

a. **AT Default**

b. **AT@F**

c. **AT&F**

d. **ATZ**

11 What is the AUX port line number on a 3620 series router?

____

____

____

12 Which of the following commands configure a router for use with a Viva modem?

a. **modem autoconfigure viva**

b. **modem configure type viva**

c. **modem autoconfigure type viva**

d. **modem autoconfigure discovery type viva**

13 What does the **physical-line async** command do and on what interfaces would you apply it?

____

____

____

14 In what configuration mode must you be to configure the physical properties of an asynchronous interface?

____

____

____

15 What does it mean when the signal pin RTS is asserted?

____

____

____

16 What is the command to manually begin a chat script named remcon?

17 When should **modem autoconfigure discovery** be used, and what are the ramifications of doing so?

18 What command would you use to add an entry to the modemcap database called newmodem?

19 Which interface type provides clocking for a line?

20 List four reasons why you would use a chat script.

21 What command can be used to determine whether Serial 0 is the DCE or DTE?

22 What command lists the transmit and receive speeds for the asynchronous ports on the router?

____

____

____

23 On which pins does the DTE device send and receive?

____

____

____

24 Which of the following would trigger a chat script start?

a. Line reset

b. DDR

c. Line activation

d. Manual

Scenarios

There are no scenarios for this particular chapter. The key issues and concepts here are syntax, syntax, and syntax. For further review, you should practice creating a configuration for a router and include all parts necessary for an asynchronous setup. The parts should include:

- Line configuration (physical)
- Interface configuration (logical)
- A new modemcap entry (your choice)
- An alias to address the modem locally (Reverse Telnet)
- A chat script for the connection (no phone number needed!)