



Configuring the Serial Interface

This chapter describes configuring the serial interface for Cisco 800M Series ISRs in the following sections:

- [Configuring the Serial Interface, page 39](#)
- [Features Supported by Serial Module, page 39](#)
- [Information About Configuring Serial Interfaces, page 41](#)
- [How to Configure Serial Interfaces, page 45](#)
- [Configuration Examples, page 47](#)

Configuring the Serial Interface

The Cisco 800M Series Integrated Services Router (ISR) provides serial WAN connectivity to remote sites using Cisco High-Level Data Link Control (HDLC), Point-to-Point Protocol (PPP), or Frame Relay encapsulation through the pluggable, serial WAN interface module. The Cisco 800M Series ISR supports both synchronous and asynchronous modes of communication.

Features Supported by Serial Module

The Cisco 800M Series ISR has 2 WAN slots that can host single-port serial module or multi-mode 3G module and supports the following combinations.

- Serial module in slot 0 and 3G module in slot 1
- 3G module in slot 0 and serial module in slot 1
- Serial module in slot 0 and serial module in slot 1

The features supported by the single-port serial module on Cisco 800M Series ISR is given as follows:

- Supports the following encapsulations :
 - HDLC
 - PPP
 - Frame Relay
 - Serial Line Internet Protocol (SLIP)
- Supports the following serial protocols

- EIA-232
 - EIA-449
 - EIA-530
 - EIA-530A
 - V.35
 - X.21
- Supports synchronous speed of up to 8 Mbps
 - Supports asynchronous speed of up to 115.2 kbps
 - Supports network clock synchronization

Cisco 800M Series ISRs use Cisco smart serial connectors. Information about the cables supported by Cisco 800M Series ISRs are provided in [Table 4-1](#).

Table 4-1 Smart Serial Cabling for Cisco 800M Series ISRs

Product Number	Cable Type	Length	Connector Type
CAB-SS-V35MT	V.35 DTE	10 ft (3m)	Male
CAB-SS-V35FC	V.35 DCE	10 ft (3m)	Female
CAB-SS-232MT	EIA/TIA-232 DTE	10 ft (3m)	Male
CAB-SS-232FC	EIA/TIA-232 DTE	10 ft (3m)	Female
CAB-SS-449MT	EIA/TIA-449 DTE	10 ft (3m)	Male
CAB-SS-449FC	EIA/TIA-449 DTE	10 ft (3m)	Female
CAB-SS-X21MT	X.21 DTE	10 ft (3m)	Male
CAB-SS-X21FC	X.21 DTE	10 ft (3m)	Female
CAB-SS-530MT	EIA/TIA-530 DTE	10 ft (3m)	Male
CAB-SS-530AMT	EIA/TIA-232 DTE	10 ft (3m)	Male

[Table 4-2](#) describes the LEDs on the Cisco 800M series serial WAN module.

Table 4-2 LEDs on the Serial WAN Module

LED Name	Color/ Status	Description
CONN	Green	Indicates the interface status and shows that line protocol is up.
	OFF	Shows that the line protocol is down.
LOOP BACK	Green	Indicates that the hardware loopback status is configured on the serial interface.
	OFF	Indicates that loopback is not configured.

Information About Configuring Serial Interfaces

To configure serial interfaces, you should understand the following concept:

- [Cisco HDLC Encapsulation, page 41](#)
- [PPP Encapsulation, page 41](#)
- [Keepalive Timer, page 43](#)
- [Frame Relay Encapsulation, page 44](#)

Cisco HDLC Encapsulation

Cisco High-Level Data Link Controller (HDLC) is the Cisco proprietary protocol for sending data over synchronous serial links. Cisco HDLC also provides a simple control protocol called Serial Line Address Resolution Protocol (SLARP) to maintain serial link keepalives. Cisco HDLC is the default for data encapsulation at Layer 2 (data link) of the Open System Interconnection (OSI) stack for efficient packet delineation and error control.



Note

Cisco HDLC is the default encapsulation type for the serial interfaces.

When the encapsulation on a serial interface is changed from HDLC to any other encapsulation type, the configured serial subinterfaces on the main interface inherit the newly changed encapsulation and they do not get deleted.

Cisco HDLC uses keepalives to monitor the link state, as described in the [“Keepalive Timer” section on page 4-43](#).

PPP Encapsulation

PPP is a standard protocol used to send data over synchronous serial links. PPP also provides a Link Control Protocol (LCP) for negotiating properties of the link. LCP uses echo requests and responses to monitor the continuing availability of the link.

**Note**

When an interface is configured with PPP encapsulation, a link is declared down and full LCP negotiation is re-initiated after five echo request (ECHOREQ) packets are sent without receiving an echo response (ECHOREP).

PPP provides the following Network Control Protocols (NCPs) for negotiating properties of data protocols that will run on the link:

- IP Control Protocol (IPCP) to negotiate IP properties
- Multiprotocol Label Switching control processor (MPLSCP) to negotiate MPLS properties
- Cisco Discovery Protocol control processor (CDPCP) to negotiate CDP properties
- IPv6CP to negotiate IP Version 6 (IPv6) properties
- Open Systems Interconnection control processor (OSICP) to negotiate OSI properties

PPP uses keepalives to monitor the link state, as described in the “[Keepalive Timer](#)” section on [page 4-43](#).

PPP supports the following authentication protocols, which require a remote device to prove its identity before allowing data traffic to flow over a connection:

- Challenge Handshake Authentication Protocol (CHAP)—CHAP authentication sends a challenge message to the remote device. The remote device encrypts the challenge value with a shared secret and returns the encrypted value and its name to the local router in a response message. The local router attempts to match the remote device’s name with an associated secret stored in the local username or remote security server database; it uses the stored secret to encrypt the original challenge and verify that the encrypted values match.
- Microsoft Challenge Handshake Authentication Protocol (MS-CHAP)—MS-CHAP is the Microsoft version of CHAP. Like the standard version of CHAP, MS-CHAP is used for PPP authentication; in this case, authentication occurs between a personal computer using Microsoft Windows and a Cisco router or access server acting as a network access server.
- Password Authentication Protocol (PAP)—PAP authentication requires the remote device to send a name and a password, which are checked against a matching entry in the local username database or in the remote security server database.

Use the **ppp authentication** command in interface configuration mode to enable CHAP, MS-CHAP, and PAP on a serial interface.

**Note**

Enabling or disabling PPP authentication does not effect the local router’s willingness to authenticate itself to the remote device.

Multilink PPP

Multilink Point-to-Point Protocol (MLPPP) is supported on the Cisco 800M Series ISR serial interface. MLPPP provides a method for combining multiple physical links into one logical link. The implementation of MLPPP combines multiple PPP serial interfaces into one multilink interface. MLPPP performs the fragmenting, reassembling, and sequencing of datagrams across multiple PPP links.

MLPPP provides the same features that are supported on PPP Serial interfaces with the exception of QoS. It also provides the following additional features:

- Fragment sizes of 128, 256, and 512 bytes
- Long sequence numbers (24-bit)
- Lost fragment detection timeout period of 80 ms
- Minimum-active-links configuration option
- LCP echo request/reply support over multilink interface
- Full T1 and E1 framed and unframed links

Keepalive Timer

Cisco keepalives are useful for monitoring the link state. Periodic keepalives are sent to and received from the peer at a frequency determined by the value of the keepalive timer. If an acceptable keepalive response is not received from the peer, the link makes the transition to the down state. As soon as an acceptable keepalive response is obtained from the peer or if keepalives are disabled, the link makes the transition to the up state.

**Note**

The **keepalive** command applies to serial interfaces using HDLC or PPP encapsulation. It does not apply to serial interfaces using Frame Relay encapsulation.

For each encapsulation type, a certain number of keepalives ignored by a peer triggers the serial interface to transition to the down state. For HDLC encapsulation, three ignored keepalives causes the interface to be brought down. For PPP encapsulation, five ignored keepalives causes the interface to be brought down. ECHOREQ packets are sent out only when LCP negotiation is complete (for example, when LCP is open).

Use the **keepalive** command in interface configuration mode to set the frequency at which LCP sends ECHOREQ packets to its peer. To restore the system to the default keepalive interval of 10 seconds, use the **keepalive** command with the **no** keyword. To disable keepalives, use the **keepalive disable** command. For both PPP and Cisco HDLC, a keepalive of 0 disables keepalives and is reported in the **show running-config** command output as **keepalive disable**.

When LCP is running on the peer and receives an ECHOREQ packet, it responds with an ECHOREP packet, regardless of whether keepalives are enabled on the peer.

Keepalives are independent between the two peers. One peer end can have keepalives enabled; the other end can have them disabled. Even if keepalives are disabled locally, LCP still responds with ECHOREP packets to the ECHOREQ packets it receives. Similarly, LCP also works if the period of keepalives at each end is different.

Frame Relay Encapsulation

When Frame Relay encapsulation is enabled on a serial interface, the interface configuration is hierarchical and comprises the following elements:

- The serial main interface comprises the physical interface and port. If you are not using the serial interface to support Cisco HDLC and PPP encapsulated connections, then you must configure subinterfaces with permanent virtual circuits (PVCs) under the serial main interface. Frame Relay connections are supported on PVCs only.
- Serial subinterfaces are configured under the serial main interface. A serial subinterface does not actively carry traffic until you configure a PVC under the serial subinterface. Layer 3 configuration typically takes place on the subinterface.
- When the encapsulation on a serial interface is changed from HDLC to any other encapsulation type, the configured serial subinterfaces on the main interface inherit the newly changed encapsulation and they do not get deleted.
- Point-to-point PVCs are configured under a serial subinterface. You cannot configure a PVC directly under a main interface. A single point-to-point PVC is allowed per subinterface. PVCs use a predefined circuit path and fail if the path is interrupted. PVCs remain active until the circuit is removed from either configuration. Connections on the serial PVC support Frame Relay encapsulation only.

**Note**

The administrative state of a parent interface drives the state of the subinterface and its PVC. When the administrative state of a parent interface or subinterface changes, so does the administrative state of any child PVC configured under that parent interface or subinterface.

To configure Frame Relay encapsulation on serial interfaces, use the **encapsulation (Frame Relay VC-bundle)** command.

Frame Relay interfaces support two types of encapsulated frames:

- Cisco (default)
- IETF

Use the **encap** command in PVC configuration mode to configure Cisco or IETF encapsulation on a PVC. If the encapsulation type is not configured explicitly for a PVC, then that PVC inherits the encapsulation type from the main serial interface.

**Note**

Cisco encapsulation is required on serial main interfaces that are configured for MPLS. IETF encapsulation is not supported for MPLS.

Before you configure Frame Relay encapsulation on an interface, you must verify that all prior Layer 3 configuration is removed from that interface. For example, you must ensure that there is no IP address configured directly under the main interface; otherwise, any Frame Relay configuration done under the main interface will not be viable.

LMI on Frame Relay Interfaces

The Local Management Interface (LMI) protocol monitors the addition, deletion, and status of PVCs. LMI also verifies the integrity of the link that forms a Frame Relay UNI interface. By default, **cisco** LMI is enabled on all PVCs.

If the LMI type is **cisco** (the default LMI type), the maximum number of PVCs that can be supported under a single interface is related to the MTU size of the main interface. Use the following formula to calculate the maximum number of PVCs supported on a card:

$$(MTU - 13) / 8 = \text{maximum number of PVCs}$$

**Note**

The default setting of the **mtu** command for a serial interface is 1504 bytes. Therefore, the default numbers of PVCs supported on a serial interface configured with **cisco** LMI is 186.

How to Configure Serial Interfaces

This section contains the following tasks:

- [Configuring a Synchronous Serial Interface, page 45](#)
- [Configuring Asynchronous Serial Interface, page 46](#)

Configuring a Synchronous Serial Interface

To configure a synchronous serial interface, perform the tasks in the following sections. Each task in the list is identified as either required or optional.

- [Specifying a Synchronous Serial Interface, page 46](#) (Required)
- [Specifying Synchronous Serial Encapsulation, page 46](#) (Optional)

Specifying a Synchronous Serial Interface

To specify a synchronous serial interface and enter interface configuration mode, use the following commands in global configuration mode.

Command	Purpose
Router(config)# interface serial <i>wic/slot/port</i>	Specifies the serial interface and enters interface configuration mode.
Example: Router# interface serial 0/0/0	

Specifying Synchronous Serial Encapsulation

By default, synchronous serial lines use the High-Level Data Link Control (HDLC) serial encapsulation method, which provides the synchronous framing and error detection functions of HDLC without windowing or retransmission. The serial interfaces support the following serial encapsulation methods:

- HDLC
- Frame Relay
- PPP

To define the encapsulation method, use the following command in interface configuration mode.

-f

Command	Purpose
Router(config-if)# encapsulation { hdlc frame-relay ppp }	Configures synchronous serial encapsulation.
Example: Router(config-if)# encapsulation ppp	

Encapsulation methods are set according to the type of protocol or application you configure in the Cisco IOS software.

For configuration examples, see the [“Configuration Examples” section on page 4-47](#).

Configuring Asynchronous Serial Interface

You can use the **physical-layer async** command to change the interface mode from the default synchronous mode to asynchronous mode

SUMMARY STEPS

1. **physical-layer async**

DETAILED STEPS

	Command or Action	Purpose
Step 1	physical-layer async Example: Router(config-if)# physical-layer async	Specifies the mode of a low-speed interface as either synchronous or asynchronous.

**Note**

You cannot use the **physical-layer async** command for frame-relay encapsulation.

When you make a transition from asynchronous mode to synchronous mode in serial interfaces, the interface state becomes down by default. You should then use the **no shutdown** option to bring the interface up.

Configuration Examples

Example: PPP Configuration:

This example shows how to configure PPP encapsulation with CHAP authentication.

```
Router> enable
Router# configure terminal
Router(config)# hostname R1
R1(config)# username R2 password cisco
R1(config)# interface serial 0/0/0
R1(config-if)# encapsulation ppp
R1(config-if)# ppp authentication chap
R1(config-if)# exit
```

```
Router> enable
Router# configure terminal
Router(config)# hostname R2
R2(config)# username R1 password cisco
R2(config)# interface serial 0/0/0
R2(config-if)# encapsulation ppp
R2(config-if)# ppp authentication chap
R2(config-if)# exit
```

This example shows how to configure PPP encapsulation with PAP authentication.

```
Router> enable
Router# configure terminal
Router(config)# hostname R1
R1(config)# username R2 password cisco
R1(config)# interface serial 0/0/0
R1(config-if)# encapsulation ppp
R1(config-if)# ppp authentication PAP
R1(config-if)# ppp pap sent-username R1 password cisco
R1(config-if)# end
```

```
Router> enable
Router#configure terminal
```

```

Router(config)# hostname R2
R2(config)# username R1 password cisco
R2(config)# interface serial 0/0/0
R2(config-if)# encapsulation ppp
R2(config-if)# ppp authentication PAP
R2(config-if)# ppp pap sent-username R2 password cisco
R2(config-if)# end

```

Example: Frame Relay Configuration

This example shows how to configure frame relay encapsulation on a serial interface.

```

Router1>enable
Router1#configure terminal
Router1(config)# interface Serial 0/0/0
Router1(config-if)# ip address 50.50.50.1 255.255.255.0
Router1(config-if)# encapsulation frame-relay
Router1(config-if)# no keepalive
Router1(config-if)# frame-relay interface-dlci 50
Router1(config-if)# end
Router2>enable
Router2#configure terminal
Router2(config)# interface Serial 0/2/0
Router2(config-if)# ip address 50.50.50.2 255.255.255.0
Router2(config-if)# encapsulation frame-relay
Router2(config-if)# no keepalive
Router2(config-if)# clock rate 2000000
Router2(config-if)# frame-relay interface-dlci 50
Router2(config-if)# end

```

Example: MLPPP Configuration

This example shows how to configure Multilink PPP on a serial interface.

```

Router1> enable
Router1# configure terminal
Router1(config)# interface Multilink 1
Router1(config-if)# ip address 120.120.120.1 255.255.255.0
Router1(config-if)# no ip route-cache
Router1(config-if)# ppp multilink
Router1(config-if)# ppp multilink group 1
Router1(config-if)# end
Router1(config)# interface Serial 0/2/0
Router1(config-if)# no ip address
Router1(config-if)# encapsulation ppp
Router1(config-if)# no ip route-cache
Router1(config-if)# ppp multilink
Router1(config-if)# ppp multilink group 1
Router1(config-if)# end
Router1(config)# interface Serial 0/2/1
Router1(config-if)# no ip address
Router1(config-if)# encapsulation ppp
Router1(config-if)# no ip route-cache
Router1(config-if)# ppp multilink
Router1(config-if)# ppp multilink group 1
Router1(config-if)# end

```

Example: Asynchronous Serial Configuration

This example shows how to configure a serial interface on asynchronous mode.

```
Router> enable
Router# configure terminal
Router(config)# interface serial 0/0/0
Router(config-if)# physical-layer async
Router(config-if)# ip address 10.0.0.2 255.0.0.0
Router(config-if)# async mode dedicated
Router(config-if)# end
```

Related Documents

Topic	Document Title
PPP and Multilink PPP	Configuring Media-Independent PPP and Multilink PPP
Serial Interface Configuration	Interface and Hardware Component Configuration Guide, Cisco IOS Release 15M&T
Frame Relay	Wide-Area Networking Configuration Guide: Frame Relay, Cisco IOS Release 15M&T

