dce-terminal-timing enable

To prevent phase shifting of the data with respect to the clock when running the line at high speeds and long distances, use the **dce-terminal-timing enable** command in interface configuration mode. If serial clock transmit external (SCTE) terminal timing is not available from the DTE, use the **no** form of this command; the DCE will use its own clock instead of SCTE from the DTE.

dce-terminal-timing enable

no dce-terminal-timing enable

Syntax Description

This command has no arguments or keywords.

Defaults

DCE uses its own clock.

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

On the Cisco 4000 router, you can specify the serial Network Interface Module timing signal configuration. When the board is operating as a DCE and the DTE provides terminal timing (SCTE or TT), the **dce-terminal-timing enable** command causes the DCE to use SCTE from the DTE.

Examples

The following example prevents phase shifting of the data with respect to the clock:

```
Router(config)# interface serial 0
Router(config-if)# dce-terminal-timing enable
```

default (interface)

To reset the configuration of an interface back to its default values, use the **default** command in global configuration mode.

default *interface-type interface-number*

Syntax Description

interface-type	Type of interface. The interface types that are available to be reset to their default values will vary depending on the available interface types on the networking device and the Cisco IOS release that is installed on the device. Not all possible interface types are documented here. • **async**—Reconfigures the specified async interface to its default value. • **atm**—Reconfigures the specified ATM interface to its default value. • **bvi**—Reconfigures the specified bridge-group virtual interface to its default value. • **dialer**—Reconfigures the specified dialer interface to its default value. • **ethernet**—Reconfigures the specified Ethernet interface to its default value. • **fastethernet**—Reconfigures the specified Fast Ethernet interface to its default value. • **fddi**—Reconfigures the specified FDDI interface to its default value. • **gigabitethernet**—Reconfigures the specified Gigabit Ethernet interface to its default value. • **group-async**—Reconfigures the specified group async interface to its default value. • **loopback**—Reconfigures the specified loopback interface to its default value. • **null**—Reconfigures the specified null interface to its default value. • **pos**—Reconfigures the specified Packet over SONET (POS) interface to its default value. • **serial**—Reconfigures the specified serial interface to its default value. • **tunnel**—Reconfigures the specified tunnel interface to its default value.
interface-number	Number of the interface, slot, router shelf, unit, port, or port adaptor if appropriate for the interface type. Slash marks may be required between elements of this argument.

Defaults

Existing interface configuration values are not reset.

Command Modes

Global configuration

Command History

Release	Modification
11.1	This command was introduced.

Usage Guidelines

The **default** command is a general-purpose command that is not limited to interfaces; it resets defaults based on the command name that follows it. Use the **default** (interface) command when you need to remove any configuration for a specified interface and reset the interface to its default values.

Examples

The following example demonstrates how to reset serial interface 0 to its default values.

```
Router(config)# default serial 0
```

Related Commands

Commands	Description
interface	Enters interface configuration mode.

delay (interface)

To set a delay value for an interface, use the **delay** command in interface configuration mode. To restore the default delay value, use the **no** form of this command.

delay *tens-of-microseconds*

no delay

Syntax Description

tens-of-microseconds	Integer that specifies the delay in tens of microseconds for an interface or network segment. To see the default delay, use the **show interfaces** command.

Defaults

Default delay values may be displayed with the **show interfaces** EXEC command.

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Examples

The following example sets a delay of 30,000-microsecond on serial interface 3:

```
Router(config)# interface serial 3
Router(config-if)# delay 3000
```

Related Commands

Command	Description
show interfaces	Displays the statistical information specific to a serial interface.

description (controller)

To add a description to an E1 or T1 controller or the Channelized T3 Interface Processor (CT3IP) in Cisco 7500 series routers, use the **description** command in controller configuration mode. To remove the description, use the **no** form of this command.

description *string*

no description

Syntax Description

string	Comment or a description (up to 80 characters) to help you remember what is attached to an interface.

Defaults

No description is added.

Command Modes

Controller configuration

Command History

Release	Modification
10.3	This command was introduced.
11.3	This command was modified to include the CT3IP controller.

Usage Guidelines

The **description** command is meant solely as a comment to be put in the configuration to help you remember what certain controllers are used for. The description affects the CT3IP and Multichannel Interface Processor (MIP) interfaces only and appears in the output of the **show controller e1**, **show controller t3**, **show controller t1**, and **more system:running-config** EXEC commands.

Examples

The following example describes a 3174 controller:

```
Router(config)# controller t1
Router(config-controller)# description 3174 Controller for test lab
```

Related Commands

Command	Description
more	Displays a specified file.
more system:running-config	
show controllers e1	Displays information about the E1 links supported by the NPM (Cisco 4000) or MIP (Cisco 7500 series).
show controllers t1	Displays information about the T1 links.
show controllers t3	Displays information about the CT3IP on Cisco 7500 series routers.

down-when-looped

To configure an interface to inform the system that it is down when loopback is detected, use the **down-when-looped** command in interface configuration mode.

down-when-looped

Syntax Description

This command has no arguments or keywords.

Defaults

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

This command is valid for High-Level Data Link Control (HDLC) or PPP encapsulation on serial and High-Speed Serial Interface (HSSI) interfaces.

This command does not have a **no** form.

Backup Interfaces

When an interface has a backup interface configured, it is often desirable that the backup interface be enabled when the primary interface is either down or in loopback. By default, the backup is only enabled if the primary interface is down. By using the **down-when-looped** command, the backup interface will also be enabled if the primary interface is in loopback.

Testing an Interface with the Loopback Command

If testing an interface with the loopback command, or by placing the DCE into loopback, the **down-when-looped** command should not be configured; otherwise, packets will not be transmitted out the interface that is being tested.

Examples

The following example configures interface serial 0 for HDLC encapsulation. It is then configured to let the system know that it is down when in loopback mode.

```
Router(config)# interface serial0
Router(config-if)# encapsulation hdlc
Router(config-if)# down-when-looped
```

Related Commands

Command	Description
backup interface	Configures an interface as a secondary or dial backup interface.
logging-events	Diagnoses equipment malfunctions between an interface and a device.

dsu bandwidth

To specify the maximum allowable bandwidth used by the PA-E3 and PA-T3 port adapters, use the **dsu bandwidth** command in interface configuration mode. To return to the default bandwidth, use the **no** form of this command.

dsu bandwidth *kbps*

no dsu bandwidth

Syntax Description

kbps	Maximum bandwidth in the range of 22 kbps to 44736 kbps. The default values are: • 34010 kbps for PA-E3 • 44736 kbps for PA-T3

Defaults

34010 kbps for PA-E3

44736 kbps for PA-T3

Command Modes

Interface configuration

Command History

Release	Modification
11.1 CA	This command was introduced.

Usage Guidelines

The local interface configuration must match the remote interface configuration. For example, if you reduce the maximum bandwidth to 16000 on the local port, you must also do the same on the remote port.

The **dsu bandwidth** command reduces the bandwidth by padding the E3 and T3 frame.

To verify the data service unit (DSU) bandwidth configured on the interface, use the **show controllers serial** EXEC command.

Examples

The following example sets the DSU bandwidth to 16000 kbps on interface 1/0/0:

```
Router(config)# interface serial 1/0/0
Router(config-if)# dsu bandwidth 16000
```

Related Commands

Command	Description
show controllers serial	Displays information that is specific to the interface hardware.

dsu mode

To specify the interoperability mode used by a PA-E3 or PA-T3 port adapters, use the **dsu mode** command in interface configuration mode. The **dsu mode** command enables and improves interoperability with other DSUs. To return to the default mode, use the **no** form of this command.

dsu mode {**0** | **1** | **2**}

no dsu mode

Syntax Description

0	Sets the interoperability mode to 0. This is the default. Specify mode 0 to connect a PA-E3 port adapter to another PA-E3 port adapter or to a Digital Link DSU (DL3100). Use mode 0 to connect a PA-T3 port adapter to another PA-T3 port adapter or to a Digital Link DSU (DL3100).
1	Sets the interoperability mode to 1. Specify mode 1 to connect a PA-E3 or PA-T3 port adapter to a Kentrox DSU.
2	Sets the interoperability mode to 2. Specify mode 2 to connect a PA-T3 port adapter to a Larscom DSU.

Defaults

0

Command Modes

Interface configuration

Command History

Release	Modification
11.1 CA	This command was introduced.

Usage Guidelines

The local interface configuration must match the remote interface configuration. For example, if you define the data service unit (DSU) interoperability mode as 1 on the local port, you must also do the same on the remote port.

You must know what type of DSU is connected to the remote port to determine if it interoperates with a PA-E3 or a PA-T3 port adapter. Use mode 0 to connect two PA-E3 port adapters or to connect the PA-E3 port adapter to a Digital Link DSU (DL3100). Use mode 1 to connect a PA-E3 or a PA-T3 port adapter to a Kentrox DSU. Use mode 2 to connect a PA-T3 port adapter to a Larscom DSU. The **dsu mode** command enables and improves interoperability with other DSUs.

To verify the DSU mode configured on the interface, use the **show controllers serial** EXEC command.

Examples

The following example sets the DSU mode to 1 on interface 1/0/0:

```
Router(config)# interface serial 1/0/0
Router(config-if)# dsu mode 1
```

Related Commands

Command	Description
show controllers serial	Displays information that is specific to the interface hardware.

dte-invert-txc

On the Cisco 4000 series, you can specify the serial Network Processor Module timing signal configuration. When the board is operating as a DTE, use the **dte-invert-txc** command in interface configuration mode to invert the TXC clock signal received from the DCE. If the DCE accepts serial clock transmit external (SCTE) from the DTE, use the **no** form of this command.

dte-invert-txc

no dte-invert-txc

Syntax Description

This command has no arguments or keywords.

Defaults

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
9.1	This command was introduced.

Usage Guidelines

Use this command if the DCE cannot receive SCTE from the DTE, the data is running at high speeds, and the transmission line is long. The **dte-invert-txc** command prevents phase shifting of the data with respect to the clock.

On the Cisco 4000 series, you can specify the serial Network Processor Module timing signal configuration. When the board is operating as a DTE, the **dte-invert-txc** command inverts the TXC clock signal it gets from the DCE that the DTE uses to transmit data.

If the DCE accepts SCTE from the DTE, use **no dte-invert-txc**.

Examples

The following example inverts the TXC on serial interface 0:

```
Router(config)# interface serial 0
Router(config-if)# dte-invert-txc
```

duplex

To configure duplex operation on an interface, use the **duplex** command in interface configuration mode. To return the system to half-duplex mode, the system default, use the **no** form of this command.

duplex {**full** | **half** | **auto**}

no duplex

Syntax Description

full	Specifies full-duplex operation.
half	Specifies half-duplex operation. This is the default.
auto	Specifies the autonegotiation capability. The interface automatically operates at half or full duplex, depending on environmental factors, such as the type of media and the transmission speeds for the peer routers, hubs, and switches used in the network configuration.

Defaults

Half-duplex mode

Command Modes

Interface configuration

Command History

Release	Modification
11.2(10)P	This command was introduced.

Usage Guidelines

To use the autonegotiation capability (that is, detect speed and duplex modes automatically), you must set both speed and duplex to auto.

Table 9 describes the access server's performance for different combinations of the duplex and speed modes. The specified **duplex** command configured with the specified **speed** command produces the resulting system action.

Table 9 Relationship Between duplex and speed Commands

duplex Command	speed Command	Resulting System Action
duplex auto	**speed auto**	Autonegotiates both speed and duplex modes.
duplex auto	**speed 100** or **speed 10**	Autonegotiates both speed and duplex modes.
duplex half or **duplex full**	**speed auto**	Autonegotiates both speed and duplex modes.
duplex half	**speed 10**	Forces 10 Mbps and half duplex.
duplex full	**speed 10**	Forces 10 Mbps and full duplex.

Table 9 Relationship Between duplex and speed Commands (continued)

duplex Command	speed Command	Resulting System Action
duplex half	**speed 100**	Forces 100 Mbps and half duplex.
duplex full	**speed 100**	Forces 100 Mbps and full duplex.

For the Cisco AS5300, the **duplex** {**full** | **half** | **auto**} command syntax replaces the following two earlier duplex commands:

- **half-duplex**
- **full-duplex**

You will get the following error messages if you try to use these commands on a Cisco AS5300:

```
Router(config)# interface fastethernet 0
Router(config-if)# full-duplex
Please use duplex command to configure duplex mode
Router(config-if)#
Router(config-if)# half-duplex
Please use duplex command to configure duplex mode
```

Examples

The following example shows the different duplex configuration options you can configure on a Cisco AS5300:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface fastethernet 0
Router(config-if)# duplex ?
  auto  Enable AUTO duplex configuration
  full  Force full duplex operation
  half  Force half-duplex operation
```

Related Commands

Command	Description
interface fastethernet	Selects a particular Fast Ethernet interface for configuration.
show controllers fastethernet	Displays information about initialization block information, transmit ring, receive ring, and errors for the Fast Ethernet controller chip on the Cisco 4500, Cisco 7200 series, or Cisco 7500 series routers.
speed	Configures the speed for a Fast Ethernet interface.

dxi interface-dfa

To specify a map command for a point to point serial interface, use the Data Exchange Interface (dxi) command **dxi interface-dfa** in interface configuration mode. To deletct the map command, use the **no** form of this command.

dxi interface-dfa *vpi-number vci*[**snap** | **mux**]

no dxi interface-dfa *vpi-number vci*

Syntax Description

vpi-number	ATM network virtual path identifier (VPI) of the permanent virtual circuit (PVC), in the range from 0 to 15. The VPI is a 4-bit field in the header of the ATM DXI frame. The VPI value is unique only on a single interface, not throughout the ATM network, because it has local significance only. Both *vpi* and *vci* cannot be specified as 0; if one is 0, the other cannot be 0.
vci	ATM network virtual channel identifier (VCI) of this PVC, in the range from 0 to 63. The VCI is a 6-bit field in the header of the ATM DXI frame. The VCI value is unique only on a single interface, not throughout the ATM network, because it has local significance only. Both *vpi* and *vci* cannot be specified as 0; if one is 0, the other cannot be 0.
snap	(Optional) LLC/SNAP encapsulation based on the protocol used in the packet. This keyword defines a PVC that can carry multiple network protocols. This is the default.
mux	(Optional) Enables multiplex (mux) encapsulation.

Defaults

No map command is specified.

Command Modes

Interface configuration

Defaults

No map definition is established.

Command Modes

Interface configuration

Command History

Release	Modification
10.3	This command was introduced.
12.4	This command was integrated into Cisco IOS Release 12.4.

Examples

The following example shows how to specify a map command.

```
Router(config)# interface serial 1
Router(config-if)# dxi interface-dfa 10
```

Related Commands

Command	Description
encapsulation atm-dxi	Enables ATM-DXI encapsulation.
dxi pvc	Configures multiprotocol or single protocol AMT-Data Exchange Interface (dxi) encapsulation.
dxi map	Maps a protocol address to a given virtual path identifier (VPI) and virtual channel identifier (VCI).
show dxi pvc	Displays the PVC statistics for a serial interface.
smds dxi	Enables Data Exchange Interface (dxi) version 2.2 support.
dxi pvc	Configures multiprotocol or single protocol AMT-Data Exchange Interface (dxi) encapsulation.

e2-clockrate

To configure the serial interface 0 for E2 (8 MHZ full duplex) and to shut down the other three serial interfaces (1 to 3), use the **e2-clockrate** command in interface configuration mode. To disable the full duplex E2, use the **no** form of this command.

e2-clockrate

no e2-clockrate

Syntax Description

This command has no arguments or keywords.

Defaults

The interfaces are not affected.

Command Modes

Interface configuration

Command History

Release	Modification
12.0(2)XD	This command was introduced.
12.0(3)T	This command was modified.

Usage Guidelines

The **e2-clockrate** privileged EXEC command is an interface configuration command and is seen only with **interface serial0**. When this command is used, serial interface 0 supports speeds up to E2 (8 MHz full duplex) and the other three serial interfaces (1 to 3) are put in "shutdown" state. Also, running this command displays the following warning message:

```
Serial interface 0 is configured to support E2 rates and serial ports "1-3" are moved to
shutdown state.
```

Examples

The following example shows sample display output for the **e2-clockrate** EXEC command.

```
Router(config-if)# e2-clockrate
Interface Serial 0 is configured to support clockrates up to E2 (8Mbps)
Interfaces serial 1-3 will not be operational
```

Related Commands

Command	Description
clock rate	Configures the clock rate for the hardware connections on serial interfaces such as NIMs and interface processors to an acceptable bit rate.

early-token-release

To enable early token release on Token Ring interfaces, use the **early-token-release** command in interface configuration mode. To disable this feature, use the **no** form of this command.

early-token-release

no early-token-release

Syntax Description

This command has no arguments or keywords.

Defaults

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

Early token release is a method whereby the Token Ring interfaces can release the token back onto the ring immediately after transmitting, rather than waiting for the frame to return. This feature helps increase the total bandwidth of the Token Ring.

The Token Ring Interface Processor (TRIP) on the Cisco 7500 series routers and the Token Ring adapters on the Cisco 7200 series routers all support early token release.

Examples

The following example enables the use of early token release on Token Ring interface 1:

```
Router(config)# interface tokenring 1
Router(config-if)#  early-token-release
```

On the Cisco 7500 series, to enable the use of early token release on your Token Ring interface processor in slot 4 on port 1, issue the following configuration commands:

```
Router(config)# interface tokenring 4/1
Router(config-if)#  early-token-release
```

encapsulation

To set the encapsulation method used by the interface, use the **encapsulation** command in interface configuration mode. To remove the encapsulation use the **no** form of this command.

encapsulation *encapsulation-type*

no encapsulation *encapsulation-type*

Syntax Description

encapsulation-type	Encapsulation type; one of the following keywords: • **atm-dxi**—ATM Mode-Data Exchange Interface. • **bstun**—Block Serial Tunnel. • **frame-relay**—Frame Relay (for serial interface). • **hdlc**—High-Level Data Link Control (HDLC) protocol for serial interface. This encapsulation method provides the synchronous framing and error detection functions of HDLC without windowing or retransmission. This is the default for synchronous serial interfaces. • **isl**—Inter-Switch Link (ISL) (for virtual LANs). • **lapb**—X.25 Link Access Procedure, Balanced. Data link layer protocol (LAPB) DTE operation (for serial interface). • **ppp**—PPP (for serial interface). • **sdlc**—IBM serial Systems Network Architecture (SNA). • **sdlc-primary**—IBM serial SNA (for primary serial interface). • **sdlc-secondary**—IBM serial SNA (for secondary serial interface). • **slip**—Specifies Serial Line Internet Protocol (SLIP) encapsulation for an interface configured for dedicated asynchronous mode or dial-on-demand routing (DDR). This is the default for asynchronous interfaces. • **smds**—Switched Multimegabit Data Services (SMDS) (for serial interface).

Defaults

The default depends on the type of interface. For example, synchronous serial interfaces default to HDLC and asynchronous interfaces default to SLIP.

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

To use SLIP or PPP, the router or access server must be configured with an IP routing protocol or with the **ip host-routing** command. This configuration is done automatically if you are using old-style **slip address** commands. However, you must configure it manually if you configure SLIP or PPP via the **interface async** command.

On lines configured for interactive use, encapsulation is selected by the user when they establish a connection with the **slip** or **ppp** EXEC command.

IP Control Protocol (IPCP) is the part of PPP that brings up and configures IP links. After devices at both ends of a connection communicate and bring up PPP, they bring up the control protocol for each network protocol that they intend to run over the PPP link such as IP or IPX. If you have problems passing IP packets and the **show interface** command shows that line is up, use the **negotiations** command to see if and where the negotiations are failing. You might have different versions of software running, or different versions of PPP, in which case you might need to upgrade your software or turn off PPP option negotiations. All IPCP options as listed in RFC 1332, "PPP Internet Protocol Control Protocol (IPCP)," are supported on asynchronous lines. Only Option 2, TCP/IP header compression, is supported on synchronous interfaces.

PPP echo requests are used as keepalive packets to detect line failure. The **no keepalive** command can be used to disable echo requests. For more information about the **no keepalive** command, refer to the chapter "IP Services Commands" in the *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services* and the chapter "Configuring IP Services" in the *Cisco IOS IP Configuration Guide*.

To use SLIP or PPP, the Cisco IOS software must be configured with an IP routing protocol or with the **ip host-routing** command. This configuration is done automatically if you are using old-style **slip address** commands. However, you must configure it manually if you configure SLIP or PPP via the **interface async** command.

Note Disable software flow control on SLIP and PPP lines before using the **encapsulation** command.

This command does not have a **no** form.

Examples

The following example resets HDLC serial encapsulation on serial interface 1:

```
Router(config)# interface serial 1
Router(config-if)# encapsulation hdlc
```

The following example enables PPP encapsulation on serial interface 0:

```
Router(config)# interface serial 0
Router(config-if)# encapsulation ppp
```

In the following example, async interface 1 is configured for PPP encapsulation:

```
Router# configure terminal
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface async 1
Router(config-if)# encapsulation ppp
```

Related Commands

Command	Description
encapsulation x25	Specifies operation of a serial interface as an X.25 device.
keepalive	Sets the keepalive timer for a specific interface.

Command	Description
ppp	Starts an asynchronous connection using PPP.
ppp authentication	Enables CHAP or PAP or both and specifies the order in which CHAP and PAP authentication are selected on the interface.
ppp bap call	Sets PPP BACP call parameters.
slip	Starts a serial connection to a remote host using SLIP.

fddi burst-count

To allow the FCI card to preallocate buffers to handle bursty FDDI traffic (for example, Network File System (NFS) bursty traffic), use the **fddi burst-count** command in interface configuration mode. To revert to the default value, use the **no** form of this command.

fddi burst-count *number*

no fddi burst-count

Syntax Description

number	Number of preallocated buffers in the range from 1 to 10. The default is 3.

Defaults

3 buffers

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

This command applies to the FCI card only.

Note The microcode software version should *not* be 128.45 or 128.43.

Examples

The following example sets the number of buffers to 5:

```
Router(config)# interface fddi 0
Router(config-if)# fddi burst-count 5
```

fddi c-min

To set the C-Min timer on the pulse code modulation (PCM), use the **fddi c-min** command in interface configuration mode. To revert to the default value, use the **no** form of this command.

fddi c-min *microseconds*

no fddi c-min

Syntax Description

microseconds	Sets the timer value, in microseconds. The default is 1600 microseconds.

Defaults

1600 microseconds

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

This command applies to the processor connection management (CMT) only. You need extensive knowledge of the PCM state machine to tune this timer. Use this command when you run into PCM interoperability problems.

Examples

The following example sets the C-Min timer to 2000 microseconds:

```
Router(config)# interface fddi 0
Router(config-if)# fddi c-min 2000
```

Related Commands

Command	Description
fddi tb-min	Sets the TB-Min timer in the PCM.
fddi tl-min-time	Controls the TL-Min time (the minimum time to transmit a PHY line state before advancing to the PCM state, as defined by the X3T9.5 specification).
fddi t-out	Sets the t-out timer in the PCM.

fddi cmt-signal-bits

To control the information transmitted during the connection management (CMT) signaling phase, use the **fddi cmt-signal-bits** command in interface configuration mode.

fddi cmt-signal-bits *signal-bits* [**phy-a** | **phy-b**]

Syntax Description

signal-bits	A hexadecimal number preceded by 0x; for example, 0x208. The FDDI standard defines 10 bits of signaling information that must be transmitted, as follows: • bit 0—Escape bit. Reserved for future assignment by the FDDI standards committee. • bits 1 and 2—Physical type, as defined in Table 10. • bit 3—Physical compatibility. Set if topology rules include the connection of a physical-to-physical type at the end of the connection. • bits 4 and 5—Link confidence test duration; set as defined in Table 11. • bit 6—MAC available for link confidence test. • bit 7—Link confidence test failed. The setting of bit 7 indicates that the link confidence was failed by the Cisco end of the connection. • bit 8—MAC for local loop. • bit 9—MAC on physical output.
phy-a	(Optional) Selects Physical Sublayer A. The default is 0x008 (hexadecimal) or 00 0000 1000 (binary). Bits 1 and 2 are set to 00 to select Physical A. Bit 3 is set to 1 to indicate “accept any connection.”
phy-b	(Optional) Selects Physical Sublayer B. The default is 0x20c (hexadecimal) or 10 0000 1100 (binary). Bits 1 and 2 are set to 10 to select Physical B. Bit 3 is set to 1 to indicate “accept any connection.” Bit 9 is set to 1 to select MAC on output. The normal data flow on FDDI is input on Physical A and output on Physical B.

Defaults

The default signal bits for the **phy-a** and **phy-b** keywords are as follows:

- **phy-a** is set to 0x008 (hexadecimal) or 00 0000 1000 (binary). Bits 1 and 2 are set to 00 to select Physical A. Bit 3 is set to 1 to indicate “accept any connection.”
- **phy-b** is set to 0x20c (hexadecimal) or 10 0000 1100 (binary). Bits 1 and 2 are set to 10 to select Physical B. Bit 3 is set to 1 to indicate “accept any connection.” Bit 9 is set to 1 to select MAC on output. The normal data flow on FDDI is input on Physical A and output on Physical B.

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

If neither the **phy-a** nor **phy-b** keyword is specified, the signal bits apply to both physical connections.

Caution Use of the **fddi cmt-signal-bits** configuration command is *not* recommended under normal operations. This command is used when debugging specific CMT implementation issues.

Table 10 lists the physical types.

Table 10 FDDI Physical Type Bit Specifications

Bit 2	Bit 1	Physical Type
0	0	Physical A
1	0	Physical B
0	1	Physical S
1	1	Physical M

Table 11 lists the duration bits.

Table 11 FDDI Link Confidence Test Duration Bit Specification

Bit 5	Bit 4	Test Duration
0	0	Short test (default 50 milliseconds)
1	0	Medium test (default 500 milliseconds)
0	1	Long test (default 5 seconds)
1	1	Extended test (default 50 seconds)

This command does not have a **no** form.

Examples

The following example sets the CMT signaling phase to signal bits 0x208 on both physical connections:

```
Router(config)# interface fddi 0
Router(config-if)#  fddi cmt-signal-bits 208
```

fddi duplicate-address-check

To turn on the duplicate address detection capability on the FDDI, use the **fddi duplicate-address-check** command in interface configuration mode. To disable this feature, use the **no** form of this command.

fddi duplicate-address-check

no fddi duplicate-address-check

Syntax Description

This command has no arguments or keywords.

Defaults

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

If you use this command, the Cisco IOS software will detect a duplicate address if multiple stations are sharing the same MAC address. If the software finds a duplicate address, it will shut down the interface.

Examples

The following example enables duplicate address checking on the FDDI:

```
Router(config)# interface fddi 0
Router(config-if)#  fddi duplicate-address-check
```

fddi encapsulate

To specify encapsulating bridge mode on the CSC-C2/FCIT interface card, use the **fddi encapsulate** command in interface configuration mode. To turn off encapsulation bridging and return the FCIT interface to its translational, nonencapsulating mode, use the **no** form of this command.

fddi encapsulate

no fddi encapsulate

Syntax Description

This command has no arguments or keywords.

Defaults

By default, the FDDI interface uses the SNAP encapsulation format defined in RFC 1042, *Standard for the Transmission of IP Datagrams Over IEEE 802 Networks*. It is not necessary to define an encapsulation method for this interface when using the CSC-FCI interface card.

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

The **no fddi encapsulate** command applies only to CSC-C2/FCIT interfaces, because the CSC-FCI interfaces are always in encapsulating bridge mode. The CSC-C2/FCIT interface card fully supports transparent and translational bridging for the following configurations:

- FDDI to FDDI
- FDDI to Ethernet
- FDDI to Token Ring

The **fddi encapsulate** command puts the CSC-C2/FCIT interface into encapsulation mode when doing bridging. In transparent mode, the FCIT interface interoperates with earlier versions of the CSC-FCI encapsulating interfaces when performing bridging functions on the same ring.

Caution Bridging between dissimilar media presents several problems that can prevent communications from occurring. These problems include bit-order translation (or usage of MAC addresses as data), maximum transfer unit (MTU) differences, frame status differences, and multicast address usage. Some or all of these problems might be present in a multimedia bridged LAN and might prevent communication from taking place. These problems are most prevalent when bridging between Token Rings and Ethernets or between Token Rings and FDDI nets. This is because of the different way Token Ring is implemented by the end nodes.

The following protocols have problems when bridged between Token Ring and other media: Novell IPX, DECnet Phase IV, AppleTalk, VINES, XNS, and IP. Furthermore, the following protocols may have problems when bridged between FDDI and other media: Novell IPX and XNS. We recommend that these protocols be routed whenever possible.

Examples

The following example sets FDDI interface 1 on the CSC-C2/FCIT interface card to encapsulating bridge mode:

```
Router(config)# interface fddi 1
Router(config-if)# fddi encapsulate
```

fddi frames-per-token

To specify the maximum number of frames that the FDDI interface will transmit per token capture, use the **fddi frames-per-token** command in interface configuration mode. To revert to the default values, use the **no** form of this command.

fddi frames-per-token *number*

no fddi frames-per-token

Syntax Description

number	Maximum number of frames to transmit per token capture. Valid values are from 1 to 10. The default is 3.

Defaults

3 frames

Command Modes

Interface configuration

Command History

Release	Modification
11.2 P	This command was introduced.

Usage Guidelines

Changing the value will increase or decrease the maximum number of frames that the FDDI interface can transmit when it receives a token. Increasing the value does not necessarily mean more frames will be transmitted on each token capture. This is heavily dependent on the traffic load of the specific interface.

When the interface captures a token, it transmits all of the frames that are queued in the interface's transmit ring, up to a maximum value specified by the **fddi frames-per-token** command.

If there are no frames ready for transmission, the token is passed on, and no frames are transmitted. If there are less than the **fddi frames-per-token** value in the transmit ring, all frames in the transmit ring are transmitted before the token is passed on. If there are more than the **fddi frames-per-token** value in the transmit ring, the specified value is transmitted before the token is passed on. The remaining frames in the transmit ring remain queued until the token is captured again.

Examples

The following example shows how to configure the FDDI interface to transmit four frames per token capture:

```
! Show fddi frames-per-token command options
  4700(config-if)# fddi frames-per-token ?
  <1-10>  Number of frames per token, default = 3
! Specify 4 as the maximum number of frames to be transmitted per token
  4700(config-if)# fddi frames-per-token 4
```

fddi smt-frames

To enable the Station Management (SMT) frame processing capability on the FDDI, use the **fddi smt-frames** command in interface configuration mode. To disable this function and prevent the Cisco IOS software from generating or responding to SMT frames, use the **no** form of this command.

fddi smt-frames

no fddi smt-frames

Syntax Description

This command has no arguments or keywords.

Defaults

Enabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

Use the **no** form of this command to turn off SMT frame processing for diagnosing purposes. Use the **fddi smt-frames** command to reenable the feature.

Examples

The following example disables SMT frame processing:

```
Router(config)# interface fddi 0
Router(config-if)#  no fddi smt-frames
```

fddi tb-min

To set the TB-Min timer in the physical connection management (PCM), use the **fddi tb-min** command in interface configuration mode. To revert to the default value, use the **no** form of this command.

fddi tb-min *milliseconds*

no fddi tb-min

Syntax Description

milliseconds	Number that sets the TB-Min timer value. The range is 0 to 65,535 milliseconds. The default is 100 milliseconds.

Defaults

100 milliseconds

Command Modes

Interface configuration

Command History

Release	Modification
10.3	This command was introduced.

Usage Guidelines

This command applies to the processor connection management (CMT) only. Use this command when you run into PCM interoperability problems.

Note You need extensive knowledge of the PCM state machine to tune this timer.

Examples

The following example sets the TB-Min timer to 200 ms:

```
Router(config)# interface fddi 0
Router(config-if)#  fddi tb-min 200
```

Related Commands

Command	Description
fddi c-min	Sets the C-Min timer on the PCM.
fddi tl-min-time	Controls the TL-Min time (the minimum time to transmit a PHY line state before advancing to the PCM state, as defined by the X3T9.5 specification).
fddi t-out	Sets the t-out timer in the PCM.

fddi tl-min-time

To control the TL-Min time (the minimum time to transmit a Physical Sublayer, or PHY line state, before advancing to the next physical connection management [PCM] state, as defined by the X3T9.5 specification), use the **fddi tl-min-time** command in interface configuration mode.

fddi tl-min-time *microseconds*

Syntax Description

microseconds	Number that specifies the time used during the connection management (CMT) phase to ensure that signals are maintained for at least the value of TL-Min so the remote station can acquire the signal. The range is 0 to 4,294,967,295 microseconds. The default is 30 microseconds.

Defaults

30 microseconds

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

Interoperability tests have shown that some implementations of the FDDI standard need more than 30 microseconds to sense a signal.

This command does not have a **no** form.

Examples

The following example changes the TL-Min time from 30 microseconds to 100 microseconds:

```
Router(config)# interface fddi 0
Router(config-if)#  fddi tl-min-time 100
```

The following example changes the TL-Min time from 30 microseconds to 100 microseconds on a Cisco 7500 series router:

```
Router(config)# interface fddi 3/0
Router(config-if)#  fddi tl-min-time 100
```

Related Commands

Command	Description
fddi c-min	Sets the C-Min timer on the PCM.
fddi tl-min-time	Controls the TL-Min time (the minimum time to transmit a PHY line state before advancing to the PCM state, as defined by the X3T9.5 specification).
fddi t-out	Sets the t-out timer in the PCM.

fddi t-out

To set the t-out timer in the physical connection management (PCM), use the **fddi t-out** command in interface configuration mode. To revert to the default value, use the **no** form of this command.

fddi t-out *milliseconds*

no fddi t-out

Syntax Description

milliseconds	Number that sets the timeout timer. The range is 0 to 65,535 ms. The default is 100 ms.

Defaults

100 milliseconds

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

This command applies to the processor connection management (CMT) only. Use this command when you run into PCM interoperability problems.

Note You need extensive knowledge of the PCM state machine to tune this timer.

Examples

The following example sets the timeout timer to 200 ms:

```
Router(config)# interface fddi 0
Router(config-if)#  fddi t-out 200
```

Related Commands

Command	Description
fddi c-min	Sets the C-Min timer on the PCM.
fddi tb-min	Sets the TB-Min timer in the PCM.
fddi tl-min-time	Controls the TL-Min time (the minimum time to transmit a PHY line state before advancing to the PCM state, as defined by the X3T9.5 specification).

fddi token-rotation-time

To control ring scheduling during normal operation and to detect and recover from serious ring error situations, use the **fddi token-rotation-time** command in interface configuration mode. To revert to the default value, use the **no** form of this command.

fddi token-rotation-time *microseconds*

no fddi token-rotation-time

Syntax Description

microseconds	Number that specifies the token rotation time (TRT). The range is 4000 to 165,000 microseconds. The default is 5000 microseconds.

Defaults

5000 microseconds

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

The FDDI standard restricts the allowed time to be greater than 4000 microseconds and less than 165,000 microseconds. As defined in the X3T9.5 specification, the value remaining in the TRT is loaded into the token holding timer (THT). Combining the values of these two timers provides the means to determine the amount of bandwidth available for subsequent transmissions.

Examples

The following example sets the rotation time to 24,000 microseconds:

```
Router(config)# interface fddi 0
Router(config-if)#  fddi token-rotation-time 24000
```

The following example sets the rotation time to 24,000 microseconds on a Cisco 7500 series router:

```
Router(config)# interface fddi 3/0
Router(config-if)#  fddi token-rotation-time 24000
```

fddi valid-transmission-time

To recover from a transient ring error, use the **fddi valid-transmission-time** command in interface configuration mode. To revert to the default value, use the **no** form of this command.

fddi valid-transmission-time *microseconds*

no fddi valid-transmission-time

Syntax Description

microseconds	Number that specifies the transmission valid timer (TVX) interval. The range is 2500 to 2,147,483,647 microseconds. The default is 2500 microseconds.

Defaults

2500 microseconds

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Examples

The following example changes the transmission timer interval to 3000 microseconds:

```
Router(config)# interface fddi 0
Router(config-if)#  fddi valid-transmission-time 3000
```

The following example changes the transmission timer interval to 3000 microseconds on Cisco 7000 series routers or Cisco 7200 series routers:

```
Router(config)# interface fddi 3/0
Router(config-if)#  fddi valid-transmission-time 3000
```

fdl

To set the Facility Data Link (FDL) exchange standard for CSU controllers or to set the FDL exchange standard for a T1 interface that uses Extended Super Frame (ESF) framing format, use the **fdl** command in controller configuration mode or ATM interface configuration mode. To disable FDL support or to specify that there is no ESF FDL, use the **no** form of this command.

Cisco MC3810 Multiservice Access Concentrator

fdl {**att** | **ansi** | **both**}

no fdl {**att** | **ansi** | **both**}

Cisco 2600 or 3600 Series Routers

fdl {**att** | **ansi** | **all** | **none**}

no fdl {**att** | **ansi** | **all** | **none**}

Syntax Description

att	Selects AT&T technical reference 54016 for ESF FDL exchange support.
ansi	Selects ANSI T1.403 for ESF FDL exchange support.
both	Specifies support for both AT&T technical reference 54016 and ANSI T1.403 for ESF FDL exchange support.
all	Specifies support for both AT&T technical reference 54016 and ANSI T1.403 for ESF FDL exchange support.
none	Specifies that there is no support for ESF FDL exchange.

Defaults

Disabled on the Cisco MC3810 multiservice access concentrator.

The default value is **ansi** on Cisco 2600 or 3600 series routers.

Command Modes

Controller configuration for the Cisco MC3810 multiservice access concentrator.

ATM interface configuration for the Cisco 2600 or 3600 series routers.

Command History

Release	Modification
11.3	This command was introduced.
12.0	This command was modified to add the **both** keyword for the Cisco MC3810.
12.0(5)T and 12.0(5)XK	The command was introduced as an ATM interface configuration command for the Cisco 2600 and 3600 series. The **none** keywordwas added to the original controller command, and the **both** keywordwas changed to **all**.

Usage Guidelines

Cisco MC3810 Multiservice Access Concentrator

You must configure this command on both T1 controllers if you want to support the CSU function on each T1 line. However, you must use the same facilities data link exchange standard as your service provider. You can have a different standard configured on each T1 controller.

Cisco 2600 or 3600 Series Routers

This command is available for T1 links only and sets the standard that will be followed for FDL messaging through a 4-Kbps out-of-band channel that a service provider uses to check for errors on the facility. You must use the same FDL exchange standard as your service provider. If the setting is not correct, the link may fail to come up. You can have a different standard configured on each T1 interface.

Note When using a multiport T1 ATM IMA network module on a Cisco 2600 or 3600 series router, ESF framing and binary eight zero substitution (B8ZS) line encoding are supported. When using a multiport E1 ATM IMA network module on a Cisco 2600 or 3600 series router, CRC4 multiframe framing and HDB3 line encoding are supported. These are the parameters specified by the ATM Forum, and they cannot be changed.

Examples

Cisco MC3810 Multiservice Access Concentrator

The following example configures the ANSI T1.403 standard for both T1 controllers:

```
Router(config)# controller t1 0
Router(config-controller)# fdl ansi
Router(config-controller)# exit
Router(config)# controller t1 1
Router(config-controller)# fdl ansi
```

Cisco 2600 or 3600 Series Routers

In a Cisco 2600 or 3600 series router, the following example specifies both ANSI and AT&T standards for FDL exchange:

```
Router(config)# interface atm 0/2
Router(config-if)#  fdl all
```

frame-relay

To configure Frame Relay payload compression for each Frame Relay port, use the **frame-relay** command in interface configuration mode. To terminate this form of payload compression over Frame Relay, use the **no** form of this command.

frame-relay payload-compression frf9 stac caim [*element-number*]

no frame-relay payload-compression

Syntax Description

payload-compression	Packet-by-packet payload compression, using the Stacker method.
frf9 stac	Enables FRF.9 compression using the Stacker method. If the router contains a data compression Advanced Interface Module (AIM) for the Cisco 2600 series router, compression is performed in the hardware (hardware compression). If the compression Advanced Interface Module (CAIM) is not available, compression is performed in the software installed on the main processor of the router (software compression).
caim *element-number*	Enable the data compression AIM hardware compression daughtercard to do compression, at the element numbered beginning with 0 and incrementing to include all possible elements.

Defaults

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
12.0(2)T	This command was introduced.

Usage Guidelines

Use the **frame-relay payload-compression** command to enable or disable payload compression on a point-to-point interface or subinterface. Use the **frame-relay map** command to enable or disable payload compression on a multipoint interface or subinterface.

Shut down the interface before changing encapsulation types. Although this is not required, shutting down the interface ensures the interface is reset for the new encapsulation.

Examples

The following example shows Frame Relay configured to use payload compression with the frf9 stac algorithm for CAIM hardware compression, using the installed data compression AIM daughtercard as the compression source:

```
Router(config-if)# frame-relay payload-compression frf9 stac caim 0
```

Related Commands

Command	Description
compress stac caim	Specifies the exact hardware compression resource preferred.
encapsulation frame-relay	Enables Frame Relay encapsulation.
frame-relay interface-dlci	Assigns a DLCI to a specified Frame Relay subinterface on the router or access server.
frame-relay map	Defines mapping between a destination protocol address and the DLCI used to connect to the destination address.
show compress	Displays compression statistics.

frame-relay map

To enable Frame Relay compression on a data-link connection (DLC) basis, and to define mapping between a destination protocol address and the data-link connection identifier (DLCI) used to connect to the destination address, use the **frame-relay map** command in interface configuration mode. To deactivate Frame Relay compression, use the **no** form of this command.

frame-relay map {*protocol protocol-address* **dlci**} **payload-compression frf9 stac caim** [*element-number*]

no frame-relay map

Syntax Description

protocol	Supported protocol, bridging, or logical link control keywords: **appletalk**, **decnet**, **dlsw**, **ip**, **ipx**, **llc2**, **rsrb**, **vines**, and **xns**.
protocol-address	Destination protocol address.
dlci	Indicates the DLCI number used to connect to the specified protocol address on the interface.
payload-compression	Packet-by-packet payload compression, using the Stacker method.
frf9	Data compression over Frame Relay.
stac	Specifies that a Stacker (LZS) compression algorithm will be used on LAPB, HDLC, and PPP encapsulation. Compression is implemented in the hardware Advanced Interface Module (AIM) installed in the router.
caim	Compression Advanced Interface Module (CAIM). Enables the data compression AIM hardware compression daughtercard to do compression.
element-number	(Optional) Compression element number, beginning with 0 and including all possible elements.

Defaults

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
12.0(1)T	This command was introduced.

Usage Guidelines

Many DLCIs known by an access server can be used to send data to many different places, but they are all multiplexed over one physical link. The Frame Relay map tells the Cisco IOS software how to get from a specific protocol and address pair to the correct DLCI.

Although you did not specified the IETF keyword during configuration, the map inherits the attributes set with the encapsulation frame-relay command so that all interfaces use IETF encapsulation.

Use the **frame-relay map** command to enable or disable payload compression on multipoint interfaces. Use the **frame-relay payload-compression** command to enable or disable payload compression on point-to-point interfaces.

The broadcast keyword provides two functions: It forwards broadcasts when multicasting is not enabled, and it simplifies the configuration of (Open Shortest Path First (OSPF) for nonbroadcast networks that will use Frame Relay.

The broadcast keyword might also be required for some routing protocols—for example, AppleTalk—that depend on regular routing table updates, especially when the router at the remote site is waiting for a routing update packet to arrive before adding the route. Network broadcasts are necessary if you intend to use routing protocols such as Routing Information Protocol (RIP) or OSPF running across the Frame Relay link.

The **frame-relay map payload-compression frf9 stac caim 0** command enables compression on the Frame Relay link, but requires the **caim 0** portion of the command. To display Frame Relay output, do not use the **show compress** command; use the **show controllers serial 0/0** command.

Examples

The following example shows configuration of the **frame-relay map payload-compression** command using the data compression AIM daughtercard for compression mapping the destination address 1.1.1.2 to DLCI 16:

```
Router(config-if)# frame-relay map ip 10.1.1.2 16 broadcast payload-compression frf9 stac
caim 0
```

Related Commands

Command	Description
compress stac caim	Specifies the exact hardware compression resource preferred.
encapsulation frame-relay	Enables Frame Relay encapsulation.
frame-relay interface-dlci	Assigns a DLCI to a specified Frame Relay subinterface on the router or access server.
frame-relay payload-compress	Enables Stacker payload compression on a specified point-to-point interface or subinterface.
show controllers serial	Displays information that is specific to the interface hardware.

framing

To select the frame type for the T1 or E1 data line, use the **framing** command in controller configuration mode.

Syntax for T1 Lines

framing {**sfadm** | **esfadm**}

Syntax for E1 Lines

framing {**crc4adm** | **pcm30adm** | **clear e1**}

Syntax Description

sfadm	Specifies Super Frame as the T1 channel.
esfadm	Specifies Extended Super Frame as the T1 channel.
crc4adm	Specifies CRC4 frame as the E1 channel.
pcm30adm	Specifies CRC4 disabled framing mode as the E1 channel.
clear e1	Specifies clear-e1 framing mode for the E1 channel.

Defaults

Extended Super Frame for a T1 line

CRC4 disabled framing for an E1 line

Command Modes

Controller configuration

Command History

Release	Modification
11.3	This command was introduced.
12.0(5)XE	The command was enhanced as an ATM interface configuration command.
12.0(7)XE1	Support for Cisco 7100 series routers was added.
12.1(5)T	This command was integrated into Cisco IOS Release 12.1(5)T.

Usage Guidelines

Use this command in configurations in which the router or access server is intended to communicate with T1 or E1 fractional data lines. The service provided determines which framing type, either **sf**, **esf**, or **crc4** is required for your T1 or E1 circuit.

Examples

The following example selects Extended Super Frame as the T1 frame type:

```
framing esfadm
```

Related Commands

Command	Description
lbo	Specifies the distance of the cable from the routers to the network equipment.
linecode	Selects the line-code type for a T1 or E1 line.

framing (E1/T1 controller)

To select the frame type for the E1 or T1 data line, use the **framing** command in controller configuration mode.

T1 Lines

framing {**sf** | **esf**}

E1 Lines

framing {**crc4** | **no-crc4**} [**australia**]

Syntax Description

sf	Specifies Super Frame as the T1 frame type. This is the default.
esf	Specifies extended Super Frame as the T1 frame type.
crc4	Specifies CRC4 frame as the E1 frame type. This is the default for Australia.
no-crc4	Specifies no CRC4 frame as the E1 frame type.
australia	(Optional) Specifies the E1 frame type used in Australia.

Defaults

Super frame is the default on a T1 line.

CRC4 frame is the default on an E1 line.

Command Modes

Controller configuration

Usage Guidelines

Use this command in configurations where the router or access server is intended to communicate with T1 or E1 fractional data lines. The service provider determines the framing type (**sf**, **esf**, or **crc4**) required for your T1/E1 circuit.

This command does not have a **no** form.

Examples

The following example selects extended Super Frame as the T1 frame type:

```
Router(config-controller)# framing esf
```

Related Commands

Command	Description
cablelength	Specifies the distance of the cable from the routers to the network equipment.
linecode	Selects the linecode type for T1 or E1 line.

framing (E3/T3 interface)

To specify E3 or T3 line framing for a PA-E3 or PA-T3 port adapter, use the **framing** command in interface configuration mode. To return to the default G.751 framing or C-bit framing, use the **no** form of this command.

PA-E3

framing {**bypass** | **g751**}

no framing

PA-T3

framing {**c-bit** | **m13** | **bypass**}

no framing

Syntax Description

bypass	Specifies bypass E3 framing.
g751	Specifies G.751 E3 framing. This is the default for the PA-E3.
c-bit	Specifies that C-bit framing is used as the T3 framing type. This is the default for the PA-T3.
m13	Specifies m13 T3 framing.

Defaults

G.751 framing for PA-E3

C-bit framing for PA-T3

Command Modes

Interface configuration

Command History

Release	Modification
11.1 CA	This command was introduced.

Usage Guidelines

The default framing is described in the ITU-T Recommendation G.751.

Note The ITU-T carries out the functions of the former Consultative Committee for International Telegraph and Telephone (CCITT).

When the framing mode is **bypass**, the E3 frame data is not included in the E3 frame, just the data.

When the framing mode is **bypass**, the T3 frame data is not included in the T3 frame, just the data.

If you use the **bypass** keyword, scrambling must be set to the default (disabled), the DSU mode must be set to the default (0), and the DSU bandwidth must be set to the default (44736).

To verify the framing mode configured on the interface, use the **show controllers serial** command in EXEC mode.

Examples

The following example sets the framing mode to bypass on interface 1/0/0:

```
Router(config)# interface serial 1/0/0
Router(config-if)#  framing bypass
```

Related Commands

Command	Description
show controllers serial	Displays information that is specific to the interface hardware.

framing (T3 controller)

To specify T3 line framing used by the CT3 feature board in a Cisco AS5800 universal access server, or by the CT3IP port adapter in Cisco 7500 series routers, use the **framing** command in controller configuration mode. To restore the default framing type, use the **no** form of this command.

Cisco AS5800 Universal Access Server

framing {**c-bit** | **m23**}

no framing

Cisco 7500 Series Routers

framing {**c-bit** | **m23** | **auto-detect**}

no framing

Syntax Description

c-bit	Specifies that C-bit framing is used as the T3 framing type. This is the default for the CT3 in a Cisco AS5800.
m23	Specifies that M23 framing is used as the T3 framing type.
auto-detect	Specifies that the CT3IP detects the framing type it receives from the far-end equipment. This is the default for the CT3IP in a Cisco 7500 series router.

Defaults

c-bit for CT3 in a Cisco AS5800

auto-detect for CT3IP in a Cisco 7500 series router

Command Modes

Controller configuration

Command History

Release	Modification
11.1 CA	This command was introduced.

Usage Guidelines

If you do not specify the **framing** command, the default **auto-detect** is used by the CT3IP to automatically determine the framing type received from the far-end equipment.

Because the CT3IP supports the Application Identification Channel (AIC) signal, the setting for the framing might be overridden by the CT3IP firmware.

You can also set the framing for each T1 channel by using the **t1 framing** controller configuration command.

Examples

The following example sets the framing for the CT3IP to C-bit:

```
Router(config)# controller t3 9/0/0
Router(config-controller)# framing c-bit
```

Related Commands

Command	Description
t1 framing	Specifies the type of framing used by the T1 channels on the CT3IP in Cisco 7500 series routers.

full-duplex

To specify full-duplex mode on full-duplex single-mode and multimode port adapters, use the **full-duplex** command in interface configuration mode. To restore the default half-duplex mode, use the **no** form of this command.

full-duplex

no full-duplex

Syntax Description

This command has no arguments or keywords.

Defaults

Half-duplex mode is the default mode on a Cisco 7500 series router, a Fast Ethernet Interface Processor (FEIP), and for serial interfaces that are configured for bisynchronous tunneling.

Command Modes

Interface configuration

Command History

Release	Modification
11.1	This command was introduced.
11.3	This command was modified to include information on FDDI full-duplex, single-mode and multimode port adapters.

Usage Guidelines

Use this command if the equipment on the other end is capable of full-duplex mode.

This command specifies full-duplex mode on full-duplex single-mode and multimode port adapters available on the following networking devices:

- Cisco 7200 series routers
- Second-generation Versatile Interface Processors (VIP2s) in Cisco 7500 series routers
- FEIP port
- Serial interface port that uses bisynchronous tunneling

Refer to the *Cisco Product Catalog* for hardware compatibility information and for specific model numbers of port adapters.

To enable half-duplex mode, use the **no full-duplex** or **half-duplex** commands.

Caution For the Cisco AS5300, the **duplex** {**full** | **half** | **auto**} command syntax replaces the **full-duplex** and **half-duplex** commands. You will get the following error messages if you try to use the **full-duplex** and **half-duplex** commands on a Cisco AS5300:

```
Router(config)# interface fastethernet 0
Router(config-if)# full-duplex
Please use duplex command to configure duplex mode
Router(config-if)#
Router(config-if)# half-duplex
Please use duplex command to configure duplex mode
```

Support for this Command

Use the question mark (?) command to find out which port adapters support this command. If the interface does not support full-duplex, an informational message similar to the one shown below is displayed, and no changes are made to the interface. To determine if the interface supports full-duplex, use the **show interfaces** command. For example, the following message is displayed if the interface does not support full-duplex:

```
% interface does not support full-duplex.
```

Use on FDDI

Full-duplex on the FDDI full-duplex port adapters allows an FDDI ring with exactly two stations to transform the ring into a full-duplex, point-to-point topology. To operate in full-duplex mode, there must be only two stations on the ring, the two stations must be capable of operating in full-duplex mode, and both stations must complete a full-duplex autoconfiguration protocol. There is no FDDI token in full-duplex mode. Refer to the *Cisco Product Catalog* for specific model numbers of port adapters.

Full-duplex autoconfiguration protocol allows an FDDI station to dynamically and automatically operate in either half-duplex (or ring) or full-duplex mode, and ensures that the stations fall back to ring mode when a configuration change occurs, such as a third station joining the ring.

After booting up, the FDDI stations begin operation in half-duplex mode. While the station performs the full-duplex autoconfiguration protocol, the station continues to provide data-link services to its users. Under normal conditions, the transition between half-duplex mode and full-duplex mode is transparent to the data-link users. The data-link services provided by full-duplex mode are functionally the same as the services provided by half-duplex mode.

If you change the full-duplex configuration (for example from disabled to enabled) on supported interfaces, the interface resets.

Examples

The following example configures full-duplex mode on the Cisco 7000 series routers:

```
Router(config)# interface fastethernet 0/1
Router(config-if)#  full-duplex
```

The following example specifies full-duplex binary synchronous communications (Bisync) mode:

```
Router(config)# interface serial 0
Router(config-if)#  encapsulation bstun
Router(config-if)#  full-duplex
```

The following example enables full-duplex mode on FDDI interface 0:

```
Router(config)# interface fddi 0/1/0
Router(config-if)#  full-duplex
```

Related Commands

Command	Description
half-duplex	Specifies half-duplex mode on an SDLC interface or on the FDDI full-duplex, single-mode port adapter and FDDI full-duplex, multimode port adapter on the Cisco 7200 series and Cisco 7500 series routers.
interface	Configures an interface type and enters interface configuration mode.
interface fastethernet	Selects a particular Fast Ethernet interface for configuration.
interface serial	Specifies a serial interface created on a channelized E1 or channelized T1 controller (for ISDN PRI, CAS, or robbed-bit signaling).
show interfaces	Displays statistics for all interfaces configured on the router or access server.
show interfaces fddi	Displays information about the FDDI interface.

half-duplex

To specify half-duplex mode on an Synchronous Data Link Control (SDLC) interface or on the FDDI full-duplex, single-mode port adapter and FDDI full-duplex, multimode port adapter on the Cisco 7200 series and Cisco 7500 series routers, use the **half-duplex** command in interface configuration mode. To reset the interface to full-duplex mode, use the **no** form of this command.

half-duplex

no half-duplex

Syntax Description

This command has no arguments or keywords.

Defaults

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
11.1	This command was introduced.
11.3	This command was modified to include information on FDDI full-duplex, single-mode and multimode port adapters.

Usage Guidelines

SDLC Interfaces

The **half-duplex** command is used to configure an SDLC interface for half-duplex mode and is used on a variety of port adapters. Use the question mark (?) command to find out which port adapters support this command.

Note The **half-duplex** command replaces the **sdlc hdx** and **media-type half-duplex** commands.

Caution For the Cisco AS5300, the **duplex** {**full** | **half** | **auto**} command syntax replaces the **full-duplex** and **half-duplex** commands. You will get the following error messages if you try to use the **full-duplex** and **half-duplex** commands on a Cisco AS5300:

```
Router(config)# interface fastethernet 0
Router(config-if)# full-duplex
Please use duplex command to configure duplex mode
Router(config-if)#
Router(config-if)# half-duplex
Please use duplex command to configure duplex mode
```

Enabling Full-Duplex Mode

To enable full-duplex mode, use the **no half-duplex** or **full-duplex** commands.

Note The **media-type half-duplex** command exists in Cisco IOS Release 11.0(5). As of Release 11.0(6), the keyword **half-duplex** was removed from the **media-type** command. In Release 11.0(6), the functionality for specifying half-duplex mode is provided by the **half-duplex** command.

Port Adapters

Refer to the *Cisco Product Catalog* for specific model numbers of port adapters.

Examples

The following example configures an SDLC interface for half-duplex mode:

```
Router(config-if)# encapsulation sdlc-primary
Router(config-if)# half-duplex
```

Related Commands

Command	Description
full-duplex	Specifies full-duplex mode on full-duplex single-mode and multimode port adapters.

half-duplex controlled-carrier

To place a low-speed serial interface in controlled-carrier mode, instead of constant-carrier mode, use the **half-duplex controlled-carrier** command in interface configuration mode. To return the interface to constant-carrier mode, use the **no** form of this command.

half-duplex controlled-carrier

no half-duplex controlled-carrier

Syntax Description

This command has no arguments or keywords.

Defaults

Constant-carrier mode, where Data Carrier Detect (DCD) is held constant and asserted by the DCE half-duplex interface.

Command Modes

Interface configuration

Command History

Release	Modification
11.2	This command was introduced.

Usage Guidelines

This command applies only to low-speed serial DCE interfaces in half-duplex mode. Configure a serial interface for half-duplex mode by using the **half-duplex** command. Refer to the *Cisco Product Catalog* for specific model numbers of networking devices which support serial interfaces.

Controlled-carrier operation means that the DCE interface has DCD deasserted in the quiescent state. When the interface has something to transmit, it asserts DCD, waits a user-configured amount of time, then starts the transmission. When the interface has finished transmitting, it waits a user-configured amount of time and then deasserts DCD.

Examples

The following examples place the interface in controlled-carrier mode and back into constant-carrier operation.

This example shows changing to controlled-carrier mode from the default of constant-carrier operation:

```
Router(config)# interface serial 2
Router(config-if)# half-duplex controlled-carrier
```

This example shows changing to constant-carrier operation from controlled-carrier mode:

```
Router(config)# interface serial 2
Router(config-if)# no half-duplex controlled-carrier
```

Related Commands

Command	Description
half-duplex timer	Tunes half-duplex timers.
physical-layer	Specifies the mode of a slow-speed serial interface on a router as either synchronous or asynchronous.

half-duplex timer

To tune half-duplex timers, use the **half-duplex timer** command in interface configuration mode. To return to the default value for that parameter, use the **no** form of this command along with the appropriate keyword.

half-duplex timer {**cts-delay** *value* | **cts-drop-timeout** *value* | **dcd-drop-delay** *value* | **dcd-txstart-delay** *value* | **rts-drop-delay** *value* | **rts-timeout** *value* | **transmit-delay** *value*}

no half-duplex timer {**cts-delay** *value* | **cts-drop-timeout** *value* | **dcd-drop-delay** *value* | **dcd-txstart-delay** *value* | **rts-drop-delay** *value* | **rts-timeout** *value* | **transmit-delay** *value*}

Syntax Description

cts-delay *value*	Specifies the delay introduced by the DCE interface between the time it detects the Request to Send (RTS) to the time it asserts Clear to Send (CTS) in response. The range is dependent on the serial interface hardware. The default **cts-delay** value is 0 ms.
cts-drop-timeout *value*	Determines the amount of time a DTE interface waits for CTS to be deasserted after it has deasserted RTS. If CTS is not deasserted during this time, an error counter is incremented to note this event. The range is 0 to 1,140,000 ms (1140 seconds). The default **cts-drop-timeout** value is 250 ms.
dcd-drop-delay *value*	Applies to DCE half-duplex interfaces operating in controlled-carrier mode (see the **half-duplex controlled-carrier** command). This timer determines the delay between the end of transmission by the DCE and the deassertion of Data Carrier Detect (DCD). The range is 0 to 4400 ms (4.4 seconds). The default **dcd-drop-delay** value is 100 ms.
dcd-txstart-delay *value*	Applies to DCE half-duplex interfaces operating in controlled-carrier mode. This timer determines the time delay between the assertion of DCD and the start of data transmission by the DCE interface. The range is 0 to 1,140,000 ms (1140 seconds). The default **dcd-txstart-delay** value is 100 ms.
rts-drop-delay *value*	Specifies the time delay between the end of transmission by the DTE interface and deassertion of RTS. The range is 0 to 1,140,000 ms (1140 seconds). The default **rts-drop-delay** value is 3 ms.
rts-timeout *value*	Determines the number of milliseconds the DTE waits for CTS to be asserted after the assertion of RTS before giving up on its transmission attempt. If CTS is not asserted in the specified amount of time, an error counter is incremented. The range is dependent on the serial interface hardware. The default **rts-timeout** value is 3 ms.
transmit-delay *value*	Specifies the number of milliseconds a half-duplex interface will delay the start of transmission. In the case of a DTE interface, this delay specifies how long the interface waits after something shows up in the transmit queue before asserting RTS. For a DCE interface, this dictates how long the interface waits after data is placed in the transmit queue before starting transmission. If the DCE interface is in controlled-carrier mode, this delay shows up as a delayed assertion of DCD. This timer enables the transmitter to be adjusted if the receiver is a little slow and is not able to keep up with the transmitter. The range is 0 to 4400 ms (4.4 seconds). The default **transmit-delay** value is 0 ms.

Defaults

The default **cts-delay** value is 0 ms.

The default **cts-drop-timeout** value is 250 ms.

The default **dcd-drop-delay** value is 100 ms.

The default **dcd-txstart-delay** value is 100 ms.

The default **rts-drop-delay** value is 3 ms.

The default **rts-timeout** value is 3 ms.

The default **transmit-delay** value is 0 ms.

Command Modes

Interface configuration

Command History

Release	Modification
11.3	This command was introduced.

Usage Guidelines

Tuning Half-Duplex Timers

The **half-duplex timer** command is used to tune half-duplex timers. With these timer tuning commands you can adjust the timing of the half-duplex state machines to suit the particular needs of their half-duplex installation.

You can configure more than one option using this command, but each option must be specified as a separate command.

Note The **half-duplex timer cts-delay** command replaces the **sdlc cts-delay** command. The **half-duplex timer rts-timeout** command replaces the **sdlc rts-timeout** command.

Value Ranges

The range of values for the **cts-delay** and **rts-timeout** keywords are dependent on the serial interface hardware.

Examples

The following example set the cts-delay timer to 10 ms and the transmit-delay timer to 50 ms:

```
Router(config)# interface serial 2
Router(config-if)# half-duplex timer cts-delay 10
Router(config-if)# half-duplex timer transmit-delay 50
```

Related Commands

Command	Description
half-duplex controlled-carrier	Places a low-speed serial interface in controlled-carrier mode, instead of constant-carrier mode.
physical-layer	Specifies the mode of a slow-speed serial interface on a router as either synchronous or asynchronous.

hold-queue

To limit the size of the IP output queue on an interface, use the **hold-queue** command in interface configuration mode. To restore the default values for an interface, use the **no** form of this command with the appropriate keyword.

hold-queue *length* {**in** | **out**}

no hold-queue {**in** | **out**}

Syntax Description

length	Integer that specifies the maximum number of packets in the queue. The range of allowed values is 0 to 65,535.
in	Specifies the input queue. The default is 75 packets. For asynchronous interfaces, the default is 10 packets.
out	Specifies the output queue. The default is 40 packets. For asynchronous interfaces, the default is 10 packets.

Defaults

The default input hold-queue limit is 75 packets.

The default output hold-queue limit is 40 packets.

For asynchronous interfaces the default is 10 packets.

These limits prevent a malfunctioning interface from consuming an excessive amount of memory. There is no fixed upper limit to a queue size.

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.
11.1	The **no hold-queue** command was added.

Usage Guidelines

Back-to-Back Routing Updates

The default of 10 packets allows the Cisco IOS software to queue a number of back-to-back routing updates. This is the default for asynchronous interfaces only; other media types have different defaults.

Hold Queues and Priority Queueing

The hold queue stores packets received from the network that are waiting to be sent to the client. It is recommended that the queue size not exceed ten packets on asynchronous interfaces. For most other interfaces, queue length should not exceed 100.

The input hold queue prevents a single interface from flooding the network server with too many input packets. Further input packets are discarded if the interface has too many input packets outstanding in the system.

If priority output queueing is being used, the length of the four output queues is set using the **priority-list** global configuration command. The **hold-queue** command cannot be used to set an output hold queue length in this situation.

For slow links, use a small output hold-queue limit. This approach prevents storing packets at a rate that exceeds the transmission capability of the link. For fast links, use a large output hold-queue limit. A fast link may be busy for a short time (and thus require the hold queue), but can empty the output hold queue quickly when capacity returns.

To display the current hold queue setting and the number of packets discarded because of hold queue overflows, use the **show interfaces** command in EXEC mode.

Caution Increasing the hold queue can have detrimental effects on network routing and response times. For protocols that use seq/ack packets to determine round trip times, do not increase the output queue. Dropping packets instead informs hosts to slow down transmissions to match available bandwidth. This is generally better than having duplicate copies of the same packet within the network (which can happen with large hold queues).

Examples

The following example sets a small input queue on a slow serial line:

```
Router(config)# interface serial 0
Router(config-if)#  hold-queue 30 i
```

Related Commands

Command	Description
priority-list	Assigns a priority queue for those packets that do not match any other rule in the priority list.
show interfaces	Displays statistics for all interfaces configured on the router or access server.

hssi external-loop-request

To allow the router to support a CSU/DSU that uses the LC signal to request a loopback from the router, use the **hssi external-loop-request** command in interface configuration mode. To disable the feature, use the **no** form of this command.

hssi external-loop-request

no hssi external-loop-request

Syntax Description

This command has no arguments or keywords.

Defaults

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

The HSA applique on the High Speed Serial Interface (HSSI) contains an LED that indicates the LA, LB, and LC signals transiting through the devices. The CSU/DSU uses the LC signal to request a loopback from the router. The CSU/DSU may want to do this so that its own network management diagnostics can independently check the integrity of the connection between the CSU/DSU and the router.

Use this command to enable a two-way, internal, and external loopback request on HSSI from the CSU/DSU.

Caution If your CSU/DSU does not support this feature, it should not be enabled in the router. Not enabling this feature prevents spurious line noise from accidentally tripping the external loopback request line, which would interrupt the normal data flow.

Examples

The following example enables a CSU/DSU to use the LC signal to request a loopback from the router:

```
Router(config-if)# hssi external-loop-request
```

hssi internal-clock

To convert the High Speed Serial Interface (HSSI) into a clock master, use the **hssi internal-clock** command in interface configuration mode. To disable the clock master mode, use the **no** form of this command.

hssi internal-clock

no hssi internal-clock

Syntax Description

This command has no arguments or keywords.

Defaults

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines

Use this command in conjunction with the HSSI null-modem cable to connect two Cisco routers together with HSSI. You must configure this command at both ends of the link, not just one.

Note HSSI network module provides full-duplex connectivity at SONET OC-1/STS-1 (51.840 Mhz), T3 (44.736 MHZ), and E3 (34.368 MHz) rates in conformance with the EIA/TIA-612 and EIA/TIA-613 specifications. The actual rate of the interface depends on the external data service unit (DSU) and the type of service to which it is connected.

Examples

The following example converts the HSSI interface into a clock master:

```
Router(config-if)# hssi internal-clock
```

hub

To enable and configure a port on an Ethernet hub of a Cisco 2505 or Cisco 2507 router, use the **hub** command in global configuration mode.

hub ethernet *number port* [*end-port*]

Syntax Description

ethernet	Indicates that the hub is in front of an Ethernet interface.
number	Hub number, starting with 0. Because there is only one hub, this number is 0.
port	Port number on the hub. On the Cisco 2505 router, port numbers range from 1 to 8. On the Cisco 2507 router, port numbers range from 1 to 16. If a second port number follows, then the first port number indicates the beginning of a port range.
end-port	(Optional) Last port number of a range.

Defaults

No hub ports are configured.

Command Modes

Global configuration

Command History

Release	Modification
10.3	This command was introduced.

Usage Guidelines

This command does not have a **no** form.

Examples

The following example enables port 1 on hub 0:

```
Router# hub ethernet 0 1
Router(config-hub)# no shutdown
```

The following example enables ports 1 through 8 on hub 0:

```
Router# hub ethernet 0 1 8
Router(config-hub)# no shutdown
```

Related Commands

Command	Description
shutdown (hub)	Shuts down a port on an Ethernet hub of a Cisco 2505 or Cisco 2507 router.

ignore-dcd

To configure the serial interface to monitor the DSR signal instead of the Data Carrier Detect (DCD) signal as the line up/down indicator, use the **ignore-dcd** command in interface configuration mode. To restore the default, use the **no** form of this command.

ignore-dcd

no ignore-dcd

Syntax Description

This command has no arguments or keywords.

Defaults

The serial interface, operating in DTE mode, monitors the DCD signal as the line up/down indicator.

Command Modes

Interface configuration

Command History

Release	Modification
11.0	This command was introduced.

Usage Guidelines

This command applies to Quad Serial NIM interfaces on the Cisco 4000 series routers and Hitachi-based serial interfaces on the Cisco 2500 and Cisco 3000 series routers.

Serial Interfaces in DTE Mode

When the serial interface is operating in DTE mode, it monitors the Data Carrier Detect (DCD) signal as the line up/down indicator. By default, the attached DCE device sends the DCD signal. When the DTE interface detects the DCD signal, it changes the state of the interface to up.

SDLC Multidrop Environments

In some configurations, such as an Synchronous Data Link Control (SDLC) multidrop environment, the DCE device sends the Data Set Ready (DSR) signal instead of the DCD signal, which prevents the interface from coming up. Use this command to tell the interface to monitor the DSR signal instead of the DCD signal as the line up/down indicator.

Examples

The following example configures serial interface 0 to monitor the DSR signal as the line up/down indicator:

```
Router(config)# interface serial 0
Router(config-if)# ignore-dcd
```

ignore-hw local-loopback

To disable the monitoring of the LL pin when in DCE mode, use the **ingnore-hw local-loopback** command in interface configuration mode. To return to the default, use the **no** form of this command.

ignore-hw local-loopback

no ignore-hw local-loopback

Syntax Description

This command has no arguments or keywords.

Defaults

Enabled

Command Modes

Interface configuration

Command History

Release	Modification
11.3	This command was introduced.

Usage Guidelines

Use this command if your system is experiencing spurious modem interrupts, which momentarily causes the interface to enter loopback mode. The end result of this behavior is the loss of SDLLC sessions.

Note This command works only with the low-speed serial interfaces.

Examples

The following example shows how to disable the monitoring of the LL pin when in DCE mode:

```
Router#configure terminal
Router(config)#interface serial 2
```

interface

To configure an interface type and enter interface configuration mode, use the **interface** command in global configuration mode.

interface *type number* [*name-tag*]

Cisco 7200 Series and Cisco 7500 Series with a Packet over SONET Interface Processor

interface *type slot/port*

Cisco 7500 Series with Channelized T1 or E1

interface serial *slot/port***:***channel-group*

Cisco 7500 Series with Ports on VIP Cards

interface *type slot/port-adapter/port* [**ethernet** | **serial**]

Cisco 4000 Series with Channelized T1 or E1 and the Cisco MC3810

interface serial *number***:***channel-group*

To configure a subinterface, use this form of the **interface** global configuration command:

Cisco 7200 Series

interface *type slot/port.subinterface-number* [**multipoint** | **point-to-point**]

Cisco 7500 Series

interface *type slot/port-adapter.subinterface-number* [**multipoint** | **point-to-point**]

Cisco 7500 Series with Ports on VIP Cards

interface *type slot/port-adapter/port.subinterface-number* [**multipoint** | **point-to-point**]

Syntax Description

type	Type of interface to be configured. See Table 12.
number	Port, connector, or interface card number. On a Cisco 4000 series router, specifies the NPM number. The numbers are assigned at the factory at the time of installation or when added to a system, and can be displayed with the **show interfaces** command.
name-tag	(Optional) Specifies the logic name to identify the server configuration so that multiple entries of server configuration can be entered. This optional argument is for use with the RLM feature.
slot	Number of the slot being configured. Refer to the appropriate hardware manual for slot and port information.
port	Number of the port being configured. Refer to the appropriate hardware manual for slot and port information.

port-adapter	Number of the port adapter being configured. Refer to the appropriate hardware manual for information about port adapter compatibility.
ethernet	(Optional) Ethernet IEEE 802.3 interface.
serial	(Optional) Serial interface.
:*channel-group*	Cisco 4000 series routers specify the T1 channel group number in the range of 0 to 23 defined with the **channel-group** controller configuration command. On a dual port card, it is possible to run channelized on one port and primary rate on the other port. Cisco MC3810 specifies the T1/E1 channel group number in the range of 0 to 23 defined with the **channel-group** controller configuration command.
.*subinterface-number*	Subinterface number in the range 1 to 4,294,967,293. The number that precedes the period (.) must match the number to which this subinterface belongs.
multipoint | **point-to-point**	(Optional) Specifies a multipoint or point-to-point subinterface. There is no default.

Defaults

No interface types are configured.

Command Modes

Global configuration

Note To use this command with the RLM feature, you must be in interface configuration mode.

Command History

Release	Modification
10.0	This command was introduced for the Cisco 7000 series routers.
11.0	This command was introduced for the Cisco 4000 series routers.
12.0(3)T	The optional *name-tag* argument was added for the RLM feature.

Usage Guidelines

Subinterfaces can be configured to support partially meshed Frame Relay networks. Refer to the "Configuring Serial Interfaces" chapter in the *Cisco IOS Interface Configuration Guide*.

There is no correlation between the number of the physical serial interface and the number of the logical LAN Extender interface. These interfaces can have the same or different numbers.

Table 12 interface Type Keywords

Keyword	Interface Type
async	Port line used as an asynchronous interface.
atm	ATM interface.

Table 12 interface Type Keywords (continued)

Keyword	Interface Type
bri	ISDN BRI. This interface configuration is propagated to each of the B channels. B channels cannot be individually configured. The interface must be configured with dial-on-demand commands in order for calls to be placed on that interface.
dialer	Dialer interface.
ethernet	Ethernet IEEE 802.3 interface.
fastethernet	100-Mbps Ethernet interface on the Cisco 4500, Cisco 4700, Cisco 7000, and Cisco 7500 series routers.
fddi	FDDI.
group-async	Master asynchronous interface.
hssi	High-Speed Serial Interface (HSSI).
lex	LAN Extender (LEX) interface.
loopback	Software-only loopback interface that emulates an interface that is always up. It is a virtual interface supported on all platforms. The *interface-number* is the number of the loopback interface that you want to create or configure. There is no limit on the number of loopback interfaces you can create.
null	Null interface.
port-channel	Port channel interface
pos	Packet OC-3 interface on the Packet over SONET Interface Processor.
serial	Serial interface.
switch	Switch interface
tokenring	Token Ring interface.
tunnel	Tunnel interface; a virtual interface. The *number* is the number of the tunnel interface that you want to create or configure. There is no limit on the number of tunnel interfaces you can create.
vg-anylan	100VG-AnyLAN port adapter.

This command does not have a **no** form.

Examples

The following example configures serial interface 0 with PPP encapsulation:

```
Router(config)# interface serial 0
Router(config-if)# encapsulation ppp
```

The following example enables loopback mode and assigns an IP network address and network mask to the interface. The loopback interface established here will always appear to be up:

```
Router(config)# interface loopback 0
Router(config-if)# ip address 131.108.1.1 255.255.255.0
```

The following example for the Cisco 7500 series router shows the interface configuration command for Ethernet port 4 on the Ethernet Interface Processor (EIP) that is installed in (or recently removed from) slot 2:

```
Router(config)# interface ethernet 2/4
```

The following example begins configuration on the Token Ring interface processor in slot 1 on port 0 of a Cisco 7500 series routers:

```
Router(config)# interface tokenring 1/0
```

The following example shows how a partially meshed Frame Relay network can be configured. In this example, subinterface serial 0.1 is configured as a multipoint subinterface with three Frame Relay permanent virtual connections (PVCs) associated, and subinterface serial 0.2 is configured as a point-to-point subinterface.

```
Router(config)# interface serial 0
Router(config-if)# encapsulation frame-relay
Router(config)# interface serial 0.1 multipoint
Router(config-if)# ip address 131.108.10.1 255.255.255.0
Router(config-if)# frame-relay interface-dlci 42 broadcast
Router(config-if)# frame-relay interface-dlci 53 broadcast
Router(config)# interface serial 0.2 point-to-point
Router(config-if)# ip address 131.108.11.1 255.255.255.0
Router(config-if)# frame-relay interface-dlci 59 broadcast
```

The following example configures circuit 0 of a T1 link for Point-to-Point Protocol (PPP) encapsulation:

```
Router(config)# controller t1 4/1
Router(config-controller)# circuit 0 1
Router(config)# interface serial 4/1:0
Router(config-if)# ip address 131.108.13.1 255.255.255.0
Router(config-if)# encapsulation ppp
```

The following example configures LAN Extender interface 0:

```
Router(config)# interface lex 0
```

Related Commands

Command	Description
channel-group	Defines the time slots of each T1 or E1 circuit.
clear interface	Resets the hardware logic on an interface.
controller	Configures a T1 or E1 controller and enters controller configuration mode.
mac-address	Sets the MAC layer address of the Cisco Token Ring.
ppp	Starts an asynchronous connection using PPP.
show interfaces	Displays the statistical information specific to a serial interface.
shutdown (RLM)	Shuts down all of the links under the RLM group.
slip	Starts a serial connection to a remote host using SLIP.

interface ctunnel

To create a virtual interface to transport IP over a Connectionless Network Service (CLNS) tunnel (CTunnel), use the **interface ctunnel** command in global configuration mode. To remove the virtual interface, use the **no** form of this command.

interface ctunnel *interface-number*

no interface ctunnel *interface-number*

Syntax Description

interface-number	CTunnel interface number (a number from 0 through 2,147,483,647).

Defaults

No virtual interface is configured.

Command Modes

Global configuration

Command History

Release	Modification
12.1(5)T	This command was introduced.

Usage Guidelines

When configuring an IP over CLNS tunnel, you must first create a virtual interface. In the following example, the **interface ctunnel** command is used to create the virtual interface.

Examples

The following example configures a CTunnel from one router to another and shows the CTunnel destination set to 49.0001.1111.1111.1111.00:

```
interface ctunnel 301
 ip address 10.0.0.3 255.255.255.0
 ctunnel destination 49.0001.1111.1111.1111.00
```

Related Commands

Command	Description
clns routing	Enables routing of CLNS packets.
ctunnel destination	Configures the destination parameter for a CLNS tunnel.
debug ctunnel	Displays debug messages for the IP over a CLNS Tunnel feature.
ip address	Sets a primary or secondary IP address for an interface.
ip routing	Enables IP routing.