CHAPTER 6

Interface Commands

This chapter contains the commands used to configure nonprotocol-specific interface features. The commands are in alphabetical order. For hardware technical descriptions, and for information about installing the router interfaces, refer to the hardware installation and maintenance publication for your particular product.

This chapter also includes Serial Line Internet Protocol (SLIP) and Point-to-Point (PPP) asynchronous connection commands. The router supports asynchronous connections via SLIP and PPP on the auxiliary port only.

For interface configuration tasks and examples, refer to the "Configuring Interfaces" chapter of the *Router Products Configuration Guide*.

For a conversion table of the modular products and Cisco 7000 processors, see Appendix E.

async-bootp

Use the **async-bootp** global configuration command to enable support for extended BOOTP requests as defined in RFC 1084 when the router is configured for SLIP. Use the **no async-bootp** global configuration command to restore the default.

async-bootp *tag* [*:hostname*] *data*
no async-bootp

Syntax Description

tag	Item being requested; expressed as filename, integer, or IP dotted-decimal address. See Table 1-1 for possible values.
:hostname	(Optional.) This entry applies only to the host specified. The argument *:hostname* accepts both an IP address and a logical host name.
data	List of IP addresses entered in dotted-decimal notation or as logical host names, a number, or a quoted string.

Table 1-1 Async-BOOTP Tag Keywords

Keyword	Description
bootfile	Specifies use of a server boot file from which to download the boot program. Use the optional *:hostname* and *data* arguments to specify the filename.
subnet-mask *mask*	Dotted-decimal address specifying the network and local subnetwork mask (as defined by RFC 950).
time-offset *offset*	Signed 32-bit integer specifying the time offset of the local subnetwork in seconds from Universal Coordinated Time (UTC).
gateway *address*	Dotted-decimal address specifying the IP addresses of gateways for this subnetwork. A preferred gateway should be listed first.
time-server *address*	Dotted-decimal address specifying the IP address of time servers (as defined by RFC 868).
IEN116-server *address*	Dotted-decimal address specifying the IP address of name servers (as defined by IEN 116).
DNS-server *address*	Dotted-decimal address specifying the IP address of Domain Name Servers (as defined by RFC 1034).
log-server *address*	Dotted-decimal address specifying the IP address of an MIT-LCS UDP log server.
quote-server *address*	Dotted-decimal address specifying the IP address of Quote of the Day servers (as defined in RFC 865).
lpr-server *address*	Dotted-decimal address specifying the IP address of Berkeley UNIX Version 4 BSD servers.
impress-server *address*	Dotted-decimal address specifying the IP address of Impress network image servers.

Keyword	Description
rlp-server *address*	Dotted-decimal address specifying the IP address of Resource Location Protocol (RLP) servers (as defined in RFC 887).
hostname *name*	The name of the client, which may or may not be domain qualified, depending upon the site.
bootfile-size *value*	A two-octet value specifying the number of 512-octet (byte) blocks in the default boot file.

Default

If no extended BOOTP commands are entered, the router software generates a gateway and subnet mask appropriate for the local network.

Command Mode

Global configuration

Usage Guidelines

Use the EXEC command **show async-bootp** to list the configured parameters. Use the **no async-bootp** command to clear the list.

Examples

The following example illustrates how to specify different boot files: one for a PC, and one for a Macintosh.

```
async-bootp bootfile :128.128.1.1 “pcboot”
async-bootp bootfile :mac "macboot"
```

With this configuration, a BOOTP request from the host on 128.128.1.1 results in a reply listing the boot filename as *pcboot*. A BOOTP request from the host named *mac* results in a reply listing the boot filename as *macboot*.

The following example specifies a subnet mask of 255.255.0.0.

```
async-bootp subnet-mask 255.255.0.0
```

The following example specifies a negative time offset of the local subnetwork of -3600 seconds.

```
async-bootp time-offset -3600
```

The following example specifies the IP address of a time server.

```
async-bootp time-server 128.128.1.1
```

Related Command

show async-bootp

async default ip address

To assign the interface address that is used by the device connecting to the router via PPP or SLIP, unless you override the address at the command line, use the **async default ip address** interface configuration command. Use the **no** form of the command to remove the address from your configuration.

async default ip address *ip address*
no async default ip address

Syntax Description

ip address	Address of the client interface.

Default

None

Command Mode

Interface configuration

Example

The following example specifies address 182.32.7.51 on asynchronous interface 1:

```
interface async 1
async default ip address 182.32.7.51
```

Related Command

async dynamic address

async dynamic address

To specify an address on an asynchronous interface (rather than using the default address), use the **async dynamic address** interface configuration command. Use the **no** form of this command to disable dynamic addressing.

async dynamic address
no async dynamic address

Syntax Description

This command has no arguments or keywords.

Default

None

Command Mode

Interface configuration

Example

The following example shows dynamic addressing assigned to an interface:

```
interface async 1
async dynamic address
```

Related Commands

ppp
slip

async dynamic routing

To implement asynchronous routing on an interface, use the **async dynamic routing** interface configuration command. The **no** form of this command disables use of routing protocols; static routing will still be used.

async dynamic routing
no async dynamic routing

Syntax Description

This command has no arguments or keywords.

Default

None

Command Mode

Interface configuration

Example

The following example shows how to enable asynchronous routing on asynchronous interface 1. The **ip tcp header-compression passive** command enables Van Jacobson TCP header compression and prevents transmission of compressed packets until a compressed packet arrives from the asynchronous link.

```
interface async 1
async dynamic routing
async dynamic address
async default ip address 1.1.1.2
ip tcp header-compression passive
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

async dynamic address
ip tcp header-compression †

async mode dedicated

To place a line into network mode using SLIP or PPP encapsulation, use the **async mode dedicated** interface configuration command. The **no** form of this command returns the line to interactive mode.

async mode dedicated
no async mode

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

With dedicated asynchronous mode, the interface will use either SLIP or PPP encapsulation, depending on which **encapsulation** method is configured for the interface. An EXEC prompt does not appear, and the line is not available for normal interactive use.

If you configure a line for dedicated mode, you will not be able to use **async dynamic address**, because there is no user prompt. You must configure either **async default ip address** and **ip unnumbered** or **ip address**.

Example

The following example assigns an Internet address to an asynchronous line and places the line into network mode. Setting the stop bits to 1 enhances performance.

```
!
interface async 1
async default ip address 182.32.7.51
async mode dedicated
encapsulation slip
```

Related Command

async mode interactive

async modeinteractive

To enable the **slip** and **ppp** EXEC commands, use the **async mode interactive** line configuration command. Use the **no** form of this command to prevent users from implementing SLIP and PPP at the EXEC level.

async mode interactive
no async mode

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Example

The following example enables the **ppp** and **slip** EXEC commands:

```
interface async 1
async mode interactive
```

Related Commands

async mode dedicated
ppp
slip

atm-dxi map

To map a given VPI and VCI to a FrameRelay DLCI, use the **atm-dxi map** interface configuration command. Use the **no** form of this command to remove the definition.

atm-dxi map *protocol address VPI VCI* [**broadcast**]
no atm-dxi map *protocol address*

Syntax Description

protocol	Specifies the protocol: **apollo**, **appletalk**, **bridge**, **clns**, **decnet**, **ip**, **novell**, **vines**, **xns**.
address	Protocol-specific address.
VPI	Specifies the Virtual Path Identifier in the range 0 to 15.
VCI	Specifies the Virtual Circuit Identifier in the range 0 to 63.
broadcast	(Optional.) Broadcasts should be forwarded to this address.

Default

None

Command Mode

Interface configuration

Usage Guidelines

This command is used in configurations where the router is intended to communicate with an ATM network through an ATM Data Service Unit (ADSU). Given the circuit identifier parameters (VPI/VCI) for the ATM virtual circuit, the DXI Frame Address (DFA) that is used for communication between the router and the ADSU is computed and used.

Example

In the following example, all IP packets intended for the host with IP address 131.108.170.49 are converted into ATM cells identified with a VPI of 2 (binary 0000 0010) and a VCI of 46 (binary 0000 0000 0010 1110) by the ADSU. Using the mapping defined in Annex A of the ATM DXI Specification, this combination of VPI and VCI maps into to a DFA of 558 (binary 1000101110). The ADSU will then extract the VPI and VCI information from the DFA of the incoming Frame Relay frame when formulating ATM cells.

```
interface serial 0
atm-dxi map ip 131.108.170.49 2 46 broadcast
```

Related Command

encapsulation atm-dxi

backup delay

To define how much time should elapse before a secondary line is set up or taken down after a primary line transition, use the **backup delay** interface configuration command. Use the **no backup delay** command to remove the definition.

backup delay {*enable-delay* | **never**} {*disable-delay* | **never**}
no backup delay {*enable-delay* | **never**} {*disable-delay* | **never**}

Syntax Description

enable-delay	Integer argument that specifies the delay in seconds after the primary line goes down before the secondary line is activated.
disable-delay	Integer argument that specifies the delay in seconds after the primary line goes up before the secondary line is deactivated.
never	Keyword that is specified to prevent the secondary line from being activated or deactivated.

Default

never

Command Mode

Interface configuration

Usage Guidelines

When a primary line goes down, the router delays the amount of seconds defined by the *enable-delay* argument before enabling the secondary line. If, after the delay period, the primary line is still down, the secondary line is activated.

When a primary line comes back up, the router will delay the amount of seconds defined by the *disable-delay* argument.

Note In cases where there are spurious signal disruptions that may appear as intermittent lost carrier signals, it is recommended that some delay be enabled before activating and deactivating a secondary.

Note The interval configured with the **backup delay** command does not affect the operation of the **backup load** command.

Examples

The following example sets a 10-second delay on deactivating the secondary line; however, the line is activated immediately:

```
interface serial 0
backup delay 0 10
```

The same example on the Cisco 7000 requires the following commands:

```
interface serial 1/1
backup delay 0 10
```

backup interface

To configure the serial interface as a secondary, or dial backup line, use the **backup interface** interface configuration command. Use the **no backup** command with the appropriate serial port designation to turn this feature off.

backup interface *interface-name*
backup interface *type slot/port* (for the Cisco 7000)
no backup interface *interface-name*

Syntax Description

interface-name or type	Serial port to be set as the secondary interface line.
slot	On the Cisco 7000, specifies the slot number.
port	On the Cisco 7000, specifies the port number.

Default

None

Command Mode

Interface configuration

Examples

The following example sets serial 1 as the backup line to serial 0:

```
interface serial 0
backup interface serial 1
```

The following example on the Cisco 7000 set serial 2 as the backup line to serial 1:

```
interface serial 1/1
backup interface serial 2/2
```

Related Command

down-when-looped

backup load

To set the traffic load thresholds for dial backup service, use the **backup load** interface configuration command. Use the **no backup load** command to remove the setting.

backup load {*enable-threshold* | **never**} {*disable-load* | **never**}
no backup load {*enable-threshold* | **never**} {*disable-load* | **never**}

Syntax Description

enable-threshold	Integer argument that specifies a percentage of the primary line's available bandwidth.
disable-load	Integer argument that specifies a percentage of the primary line's available bandwidth.
never	Keyword that sets the secondary line to never be activated due to load.

Default

never

Command Mode

Interface configuration

Usage Guidelines

When the transmitted or received load on the primary line is greater than the value assigned to the *enable-threshold* argument, the secondary line is enabled.

When the transmitted load on the primary line plus the transmitted load on the secondary line is less than the value entered for the *disable-load* argument, and the received load on the primary line plus the received load on the secondary line is less than the value entered for the *disable-load* argument, the secondary line is disabled.

If the **never** keyword is used instead of an *enable-threshold* value, the secondary line is never activated because of load. If the **never** keyword is used instead of an *disable-load* value, the secondary line is never deactivated because of load.

Examples

The following example sets the traffic load threshold to 60 percent on the primary line. When that load is exceeded, the secondary line is activated, and will not be deactivated until the combined load is less than 5 percent of the primary bandwidth.

```
interface serial 0
backup load 60 5
```

The same example on the Cisco 7000 requires the following commands:

```
interface serial 1/1
backup load 60 5
```

bandwidth

To set a bandwidth value for an interface, use the **bandwidth** interface configuration command. Use the **no bandwidth** command to restore the default values.

bandwidth *kilobits*
no bandwidth

Syntax Description

kilobits	Intended bandwidth in kilobits per second. For a full bandwidth DS3, enter the value 44736.

Default

Default bandwidth values are set during startup and can be displayed with the EXEC command **show interfaces**.

Command Mode

Interface configuration

Usage Guidelines

The **bandwidth** command sets an informational parameter only; you cannot adjust the actual bandwidth of an interface with this command. For some media, such as Ethernet, the bandwidth is fixed; for other media, such as serial lines, you can change the actual bandwidth by adjusting hardware. For both classes of media, you can use the **bandwidth** configuration command to communicate the current bandwidth to the higher-level protocols.

Additionally, IGRP uses the minimum path bandwidth to determine a routing metric. The TCP protocol adjusts initial retransmission parameters based on the apparent bandwidth of the outgoing interface.

At higher bandwidths, the value you configure with the **bandwidth** command is not what is displayed by the **show interface** command. The value shown is that used in IGRP updates and also used in computing load.

Note This is a routing parameter only; it does not affect the physical interface.

Example

The following example sets the full bandwidth for DS3 transmissions:

```
interface serial 0
bandwidth 44736
```

Related Command

A dagger (†) indicates that the command is documented in another chapter.

vines metric †

clear counters

To clear the interface counters, use the **clear counters** EXEC command.

clear counters [*type number*]
clear counters [*type slot/port*] (for the Cisco 7000)

Syntax Description

type (Optional.) Specifies the interface type; it is one of the keywords listed in Table 1-2.

number (Optional.) Specifies the interface counter displayed with the **show interfaces** command.

slot (Optional.) On the Cisco 7000, specifies the backplane slot number. Value can be 0, 1, 2, 3, or 4.

port (Optional.) On the Cisco 7000, specifies the port number of the interface. Value can be 0, 1, 2, or 3 for the serial interface.

Table 1-2 Clear Counters Interface Type Keywords

Keyword	Interface Type
async	Async interface
bri	Integrated Services Digital Network (ISDN) Basic Rate Interface (BRI)
dialer	Dialer interface
ethernet	Ethernet interface
fddi	Fiber Distributed Data Interface (FDDI)
hssi	High-Speed Serial Interface (HSSI)
loopback	Loopback interface
null	Null interface
serial	Synchronous serial interface
tokenring	Token Ring interface
tunnel	Tunnel interface

Command Mode

EXEC

Usage Guidelines

This command clears all the current interface counters from the interface unless the optional arguments *type* and *number* are specified to clear only a specific interface type (serial, Ethernet, Token Ring, and so on).

Note This command will not clear counters retrieved using SNMP, but only those seen with the EXEC **show interface** command.

Example

The following example illustrates how to clear all interface counters:

```
Router# clear counters
```

Related Command

show interfaces

clear interface

To reset the hardware logic on an interface, use the **clear interface** EXEC command.

clear interface *type number*
clear interface *type slot/port* (on a Cisco 7000)

Syntax Description

type Specifies the interface type; it is one of the keywords listed in Table 1-3.

number Specifies the port, connector, or interface card number.

slot On a Cisco 7000, specifies the backplane slot number and can be 0, 1, 2, 3, or 4.

port On a Cisco 7000, specifies the port number of the interface and can be 0, 1, 2, 3, 4 or 5 depending on the type of interface, as follows:

- EIP (Ethernet Interface Processor) 0, 1, 2, 3, 4, or 5
- FIP (FDDI Interface Processor) 0
- HIP (HSSI Interface Processor) 0
- TRIP (Token Ring Interface Processor) 0, 1, 2, or 3

Table 1-3 Clear Interface Type Keywords

Keyword	Interface Type
async	Async interface
bri	Integrated Services Digital Network (ISDN) Basic Rate Interface (BRI)
ethernet	Ethernet interface
fddi	Fiber Distributed Data Interface (FDDI)
hssi	High-Speed Serial Interface (HSSI)
loopback	Loopback interface
null	Null interface
serial	Synchronous serial interface
tokenring	Token Ring interface
tunnel	Tunnel interface

Command Mode

EXEC

Note Under normal circumstances, you do not need to clear the hardware logic on interfaces.

Example

The following example resets the interface logic on interface HSSI 1:

```
Router# clear interface hssi 1
```

clear rif-cache

To clear entries from the Routing Information Field (RIF) cache, use the **clear rif-cache** EXEC command.

clear rif-cache

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Example

The following example illustrates how to clear the RIF cache:

```
Router# clear rif-cache
```

Related Command

A dagger (†) indicates that the command is documented in another chapter.

multiring †

clockrate

To configure the clock rate for appliques (connector hardware) on the serial interface of the MCI and SCI cards to an acceptable bit rate, use the **clockrate** interface configuration command. Use the **no clockrate** command to remove the clock rate if you change the interface from a DCE to a DTE device.

clockrate *bps*
no clockrate

Syntax Description

bps	Desired clock rate in bits per second: 1200, 2400, 4800, 9600, 19200, 34800, 56000, 64000, 72000, 125000, 148000, 500000, 800000, 1000000, 1300000, 2000000, or 4000000.

Default

No clock rate

Command Mode

Interface configuration

Usage Guidelines

Be aware that the fastest speeds might not work if your cable is too long, and that speeds faster than 148,000 bits per second are too fast for RS-232 signaling. It is recommended that you only use the synchronous serial RS-232 signal at speeds up to 64,000 bits per second. To permit a faster speed, use an RS-449 or V.35 applique.

Example

The following example sets the clock rate on the first serial interface to 64,000 bits per second:

```
interface serial 0
clockrate 64000
```

cmt connect

To start the processes that perform the connection management (CMT) function and allow the ring on one fiber to be started, use the **cmt connect** EXEC command.

cmt connect [*interface-name* [**phy-a** | **phy-b**]]

Syntax Description

interface-name	(Optional.) Specifies the FDDI interface.
phy-a	(Optional.) Selects Physical Sublayer A.
phy-b	(Optional.) Selects Physical Sublayer B.

Command Mode

EXEC

Usage Guidelines

In normal operation, the FDDI interface is operational once the interface is connected and configured. The **cmt connect** command allows the operator to start the processes that perform the CMT function.

The **cmt connect** command is not needed in the normal operation of FDDI; this command is used mainly in interoperability tests.

Examples

The following examples demonstrate use of the **cmt connect** command for starting the CMT processes on the FDDI ring.

This command starts all FDDI interfaces:

```
Router# cmt connect
```

This command starts both fibers on the FDDI interface unit zero:

```
Router# cmt connect fddi 0
```

This command on the Cisco 7000 starts both fibers on the FDDI interface unit zero:

```
Router# cmt connect fddi 1/0
```

This command starts only Physical Sublayer A on the FDDI interface unit 0 (zero):

```
Router# cmt connect fddi 0 phy-a
```

This command on the Cisco 7000 starts only Physical Sublayer A on the FDDI interface unit 0 (zero):

```
Router# cmt connect fddi 1/0 phy-a
```

cmt disconnect

To stop the processes that perform the connection management (CMT) function and allow the ring on one fiber to be stopped, use the **cmt disconnect** EXEC command.

cmt disconnect [*interface-name* [**phy-a** | **phy-b**]]

Syntax Description

interface-name	(Optional.) Specifies the FDDI interface.
phy-a	(Optional.) Selects Physical Sublayer A.
phy-b	(Optional.) Selects Physical Sublayer B.

Command Mode

EXEC

Usage Guidelines

In normal operation, the FDDI interface is operational once the interface is connected and configured, and is turned off using the **shutdown** interface configuration command. The **cmt disconnect** command allows the operator to stop the processes that perform the CMT function and allow the ring on one fiber to be stopped.

The **cmt disconnect** command is not needed in the normal operation of FDDI; this command is used mainly in interoperability tests.

Examples

The following examples demonstrate use of the **cmt disconnect** command for stopping the CMT processes on the FDDI ring.

This command stops all FDDI interfaces:

```
Router# cmt disconnect
```

This command stops both fibers on the FDDI interface unit zero:

```
Router# cmt disconnect fddi 0
```

This command on the Cisco 7000 stops both fibers on the FDDI interface unit zero:

```
Router# cmt disconnect fddi 1/0
```

This command stops only Physical Sublayer A on the FDDI interface unit 0 (zero). This command causes the FDDI media to go into a wrapped state so that the ring will be broken.

```
Router# cmt disconnect fddi 0 phy-a
```

This command on the Cisco 7000 stops only Physical Sublayer A on the FDDI interface unit 0 (zero). This command causes the FDDI media to go into a wrapped state so that the ring will be broken.

```
Router# cmt disconnect fddi 1/0 phy-a
```

crc

To set the length of the cyclic redundancy check (CRC) on a Fast Serial Interface Processor (FSIP) of a Cisco 7000, use the **crc** interface configuration command. To set the CRC length to 16 bits, use the **no** form of the command.

crc *size*
no crc

Syntax Description

size	CRC size (16 or 32 bits).

Default

16 bits

Command Mode

Interface configuration

Usage Guidelines

All interfaces use a 16-bit cyclic redundancy check (CRC) by default, but also support a 32-bit CRC. CRC is an error-checking technique that uses a calculated numeric value to detect errors in transmitted data. The designators 16 and 32 indicate the length (in bits) of the frame check sequence (FCS). A CRC of 32 bits provides more powerful error detection, but adds overhead. Both the sender and receiver must use the same setting.

CRC-16, the most widely used throughout the United States and Europe, is used extensively with wide-area networks (WANs). CRC-32 is specified by IEEE 802 and as an option by some point-to-point transmission standards. It is often used on SMDS networks and LANs.

Example

In the following example, the 32-bit CRC is enabled on serial interface 3/0:

```
interface serial 3/0
crc 32
```

custom-queue-list

To assign a custom queue list to an interface, use the **custom-queue-list** interface configuration command. To remove a specific list or all list assignments, use the **no** form of the command.

custom-queue-list *list*
no custom-queue-list [*list*]

Syntax Description

list	Number of the custom queue list you want to assign to the interface. An integer from 1 to 10.

Default

None

Command Mode

Interface configuration

Usage Guidelines

Only one queue list can be assigned per interface. Use this command in place of the **priority-list** command (not in addition to it). Custom queuing allows a fairness not provided with priority queuing. With custom queueing, you can control the interfaces' available bandwidth when it is unable to accommodate the aggregate traffic enqueued. Associated with each output queue is a configurable byte count, which specifies how many bytes of data should be delivered from the current queue by the system before the system moves on to the next queue. When a particular queue is being processed, packets are sent until the number of bytes sent exceeds the queue byte count or until the queue is empty.

Example

In the following example, custom queue list number 3 is assigned to interface serial 0:

```
Router(config)# interface serial 0
Router(config-if)# custom-queue-list 3
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

queue-list
queue-list default †
queue-list interface †
queue-list protocol †
queue-list queue byte-count †
queue-list queue limit †
queue-list stun †

dce-terminal-timing enable

When running the line at high speeds and long distances, use the **dce-terminal-timing enable** interface configuration command to prevent phase shifting of the data with respect to the clock. If SCTE is not available from the DTE, use **no dce-terminal-timing-enable**, which causes the DCE to 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 keywords or arguments.

Default

no dce-terminal-timing enable

Command Mode

Interface configuration

Usage Guidelines

On the Cisco 4000 platform, 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.

Example

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

```
interface serial 0
dce-terminal-timing enable
```

delay

To set a delay value for an interface, use the **delay** interface configuration command. Use the **no delay** command to restore the default delay value

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.

Default

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

Command Mode

Interface configuration

Example

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

```
interface serial 3
delay 30000
```

Related Command

show interfaces

description

To add a description to an interface configuration, use the **description** interface configuration command. Use the **no description** command to remove the description.

description *string*
no description

Syntax Description

string	Comment or a description to help you remember what is attached to this interface.

Default

None

Command Mode

Interface configuration

Usage Guidelines

The **description** command is meant solely as a comment to be put in the configuration to help you remember what certain interfaces are used for. The description appears in the output of the following EXEC commands: **show configuration, show interfaces**, and **write terminal.**

Example

The following example describes a 3174 controller on interface serial 0.

```
interface serial 0
description 3174 Controller for test lab
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

show configuration †
show interfaces
write terminal †

down-when-looped

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

down-when-looped

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This command is valid for HDLC or PPP encapsulation on serial and HSSI 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.

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

Example

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

```
interface serial0
encapsulation hdlc
down-when-looped
```

Related Commands

backup interface
loopback

dte-invert-txc

On the Cisco 4000 platform, you can specify the serial Network Interface 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. Use the **no** form of this command if the DCE accepts SCTE from the DTE.

dte-invert-txc
no dte-invert-txc

Syntax Description

This command has no arguments or keywords.

Default

Off

Command Mode

Interface configuration

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. This prevents phase shifting of the data with respect to the clock.

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

Example

The following example inverts the TXC on interface serial 0:

```
interface serial 0
dte-invert-txc
```

early-token-release

To enable *early token release*, 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, use the **early-token-release** interface configuration command. This feature helps to increase the total bandwidth of the Token Ring.

The CSC-C2CTR, CSC-R16 (or CSC-R16M), CSC-2R, and CSC-1R cards and the Token Ring Interface Processor (TRIP) on the Cisco 7000 all support early token release. Once enabled, use the **no early-token-release** command to disable this feature.

early-token-release
no early-token-release

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Examples

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

```
interface tokenring 1
early-token-release
```

On a Cisco 7000, 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:

```
interface tokenring 4/1
early-token-release
```

encapsulation

To set the encapsulation method used by the interface, use the **encapsulation** interface configuration command.

encapsulation *encapsulation-type*

Syntax Description

encapsulation-type	Encapsulation type. See Table 1-4 for a list of supported encapsulation types.

Table 1-4 Encapsulation Types

Keyword	Encapsulation Type
atm-dxi	Asynchronous Transfer Mode-Data Exchange Interface.
bfex25	Blacker Front End Encryption X.25 operation (for serial interface).
ddnx25-dce	DDN X.25 DCE operation (for serial interface).
ddnx25	DDN X.25 DTE operation (for serial interface).
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.
lapb-dce	X.25 LAPB DCE operation (for serial interface).
lapb	X.25 LAPB DTE operation (for serial interface).
multi-lapb-dce	X.25 LAPB multiprotocol DCE operation (for serial interface).
multip-lapb	X.25 LAPB multiprotocol DTE operation (for serial interface).
ppp	Point-to-Point Protocol (PPP) (for serial interface).
sdlc-primary	IBM serial SNA (for serial interface).
sdlc-secondary	IBM serial SNA (for serial interface).
smds	Switched Multimegabit Data Services (SMDS) (for serial interface).
stun	Cisco Serial Tunnel (STUN) protocol functions (for serial interface).
x25-dce	X.25 DCE operation (for serial interface).
x25	X.25 DTE operation (for serial interface).

Default

The default depends on the type of interface. For example, a synchronous serial interface defaults to HDLC. An Ethernet interface defaults to ARPA.

Command Mode

Interface configuration

Usage Guidelines

To change the encapsulation on LAN media, do not use the **encapsulation** command. Instead, change the encapsulation on a per-protocol basis.

Examples

The following example reenables standard Ethernet Version 2.0 encapsulation on interface Ethernet 0:

```
interface ethernet 0
encapsulation arpa
```

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

```
interface serial 1
encapsulation hdlc
```

The following example sets IEEE 802.3 encapsulation on Ethernet 1:

```
interface ethernet 1
encapsulation sap
```

The following example enables PPP encapsulation on interface serial 0:

```
interface serial 0
encapsulation ppp
```

The following example sets IEEE 802.2 encapsulation on Ethernet 1:

```
interface ethernet 1
encapsulation snap
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

keepalive †
ppp
ppp authentication chap
slip

fddi burst-count

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

fddi burst-count *number*
no fddi burst-count

Syntax Description

number	Number of preallocated buffers in the range from 1 to 10.

Default

3 buffers

Command Mode

Interface configuration

Usage Guidelines

This command applies to the FCI card only. The microcode software version should *not* be 128.45 or 128.43.

Example

The following example sets the number of buffers to 5:

```
interface fddi 0
fddi burst-count 5
```

fddi c-min

To set the C-Min timer on the PCM, use the **fddi c-min** interface configuration command. Use the **no** form of this command to revert to the default value.

fddi c-min *microseconds*
no fddi c-min

Syntax Description

microseconds	Sets the timer value in microseconds.

Default

1600 microseconds

Command Mode

Interface configuration

Usage Guidelines

This command applies to the processor 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.

Example

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

```
interface fddi 0
fddi c-min 2000
```

Related Commands

fddi tb-min
fddi tl-min
fddi t-out

fddi cmt-signal-bits

To control the information transmitted during the connection management (CMT) signaling phase, use the **fddi cmt-signal-bits** interface configuration command. If neither the **phy-a** nor **phy-b** keyword is specified, the signal bits apply to both physical connections.

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

Note 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.

Syntax Description

signal-bits A hexadecimal number preceded by 0x; for example, 0x208. The FDDI standard defines ten 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 1-5.
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 1-6.
bit 6—Media Access Control (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.

phy-b (Optional.) Selects Physical Sublayer B.

Default

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 Mode

Interface configuration

Usage Guidelines

Use the following tables to set the physical type and duration bits.

Table 1-5 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 1-6 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)

Example

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

```
interface fddi 0
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** interface configuration command. Use the **no** form of this command to turn off this feature.

fddi duplicate-address-check
no fddi duplicate-address-check

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

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

Example

The following example enables duplicate address checking on the FDDI:

```
interface fddi 0
fddi duplicate-address-check
```

fddi encapsulate

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

fddi encapsulate
no fddi encapsulate

Syntax Description

This command has no arguments or keywords.

Default

The FDDI interface by default uses the SNAP encapsulation format defined in RFC 1042. It is not necessary to define an encapsulation method for this interface when using the CSC-FCI interface card.

Command Mode

Interface configuration

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 command **fddi encapsulate** 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 b it-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 may be present in a multimedia bridged LAN and preventing 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. Further, 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.

Example

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

```
interface fddi 1
fddi encapsulate
```

fddi if-cmt

To turn on the connection management (CMT) onboard functions on the CSC-FCI and CSC-C2/FCIT interface cards, use the **fddi if-cmt** interface configuration command. Use the **no fddi if-cmt** command to turn these functions off.

fddi if-cmt
no fddi if-cmt

Syntax Description

This command has no arguments or keywords.

Default

CMT functions on

Command Mode

Interface configuration

Usage Guidelines

The CSC-C2/FCIT and CSC-FCI interface cards provide CMT functions in microcode. These functions are separate from those provided on the processor card and accessed through EXEC commands.

A typical use of the **no fddi if-cmt** command is when you work with new FDDI equipment and have problems bringing up the ring. When you use this command to disable the CMT microcode, the following actions occur:

- The FCI and FCIT CMT microcode is disabled.
- The main system code performs the CMT function while debugging output is generated.

Example

The following example disables CMT functions on the FDDI interface cards:

```
interface fddi 0
no fddi if-cmt
```

fddi smt-frames

To enable the SMT frame processing capability on the FDDI, use the **fddi smt-frames** interface configuration command. Use the **no** form of this command to disable this feature, in which case the router will not generate or respond to SMT frames.

fddi smt-frames
no fddi smt-frames

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

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.

Example

The following example disables SMT frame processing:

```
interface fddi 0
no fddi smt-frames
```

fddi t-out

To set the t-out timer in the PCM, use the **fddi t-out** interface configuration command. Use the **no** form of this command to revert to the default value.

fddi t-out *milliseconds*
no fddi t-out

Syntax Description

milliseconds	Sets the timeout timer.

Default

100 milliseconds

Command Mode

Interface configuration

Usage Guidelines

This command applies to the processor 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.

Example

The following example sets the timeout timer to 200 milliseconds:

```
interface fddi 0
fddi t-out 200
```

Related Commands

fddi c-min
fddi tb-min
fddi tl-min

fddi tb-min

To set the TB-Min timer in the PCM, use the **fddi tb-min** interface configuration command. Use the **no** form of this command to revert to the default value.

fddi tb-min *milliseconds*
no fddi tb-min

Syntax Description

milliseconds	Sets the TM-Min timer.

Default

100 milliseconds

Command Mode

Interface configuration

Usage Guidelines

This command applies to the processor 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.

Example

The following example sets theTB-Min timer to 200 milliseconds:

```
interface fddi 0
fddi tb-min 200
```

Related Commands

fddi c-min
fddi tl-min-time
fddi t-out

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** interface configuration command.

fddi tl-min-time *microseconds*

Syntax Description

microseconds	Integer 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.

Default

30 microseconds

Command Mode

Interface configuration

Usage Guidelines

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

Examples

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

```
interface fddi 0
fddi tl-min-time 100
```

The following example changes the TL-Min time from 30 microseconds to 100 microseconds on a Cisco 7000:

```
interface fddi 3/0
fddi tl-min-time 100
```

Related Commands

fddi c-min
fddi tb-min
fddi t-out

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** interface configuration command.

fddi token-rotation-time *microseconds*

Syntax Description

microseconds	Integer that specifies the token rotation time (TRT).

Default

5000 microseconds

Command Mode

Interface configuration

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:

```
interface fddi 0
fddi token-rotation-time 24000
```

The following example sets the rotation time to 24,000 microseconds on a Cisco 7000:

```
interface fddi 3/0
fddi token-rotation-time 24000
```

fddi valid-transmission-time

To recover from a transient ring error, use the **fddi valid-transmission-time** interface configuration command.

fddi valid-transmission-time *microseconds*

Syntax Description

microseconds Integer that specifies the transmission valid timer (TVX) interval.

Default

2500 microseconds

Command Mode

Interface configuration

Examples

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

```
interface fddi 0
fddi valid-transmission-time 3000
```

The following example changes the transmission timer interval to 3000 microseconds on a Cisco 7000:

```
interface fddi 3/0
fddi valid-transmission-time 3000
```

hold-queue

To specify the hold-queue limit of an interface, use the **hold-queue** interface configuration command. Use the **no hold-queue** command with the appropriate keyword to restore the default values for an interface.

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

Syntax Description

length	An integer that specifies the maximum number of packets in the queue.
in	A keyword that specifies the input queue.
out	A keyword that specifies the output queue.

Default

The default input hold-queue limit is 75 packets. The default output hold-queue limit is 40 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 Mode

Interface configuration

Usage Guidelines

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 EXEC command **show interfaces**.

Note 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).

Example

The following example illustrates how to set a small input queue on a slow serial line:

```
interface serial 0
hold-queue 30 in
```

Related Command

show interfaces

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** interface configuration command. Use the **no** form of this command to remove the feature if it had been enabled.

hssi external-loop-request
no hssi external-loop-request

Syntax Description

This command has no arguments or keywords.

Default

no hssi external-loop-request

Command Mode

Interface configuration

Usage Guidelines

The HSA applique (on the 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.

Note 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.

Example

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

```
hssi external-loop-request
```

hssi internal-clock

To convert the HSSI interface into a 45 MHz clock master, use the **hssi internal-clock** interface configuration command. Use the **no** form of this command to disable the clock master mode.

hssi internal-clock
no hssi internal-clock

Syntax Description

This command has no arguments or keywords.

Default

no hssi internal-clock

Command Mode

Interface configuration

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.

Example

The following example converts the HSSI interface into a 45 MHz clock master:

```
hssi internal-clock
```

interface

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

interface *interface-type interface-number*
interface *interface-type slot/port* (for the Cisco 7000)

To configure a subinterface, use the **interface** global configuration command.

interface *interface-type interface-number.subinterface-number* [**multipoint** | **point-to-point**]
interface *interface-type slot/port.subinterface-number* [**multipoint** | **point-to-point**] (for the Cisco 7000)

Syntax Description

interface-type	Specifies the type of interface to be configured. See Table 1-7.
interface-number	Specifies the port, connector, or interface card 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.
.subinterface-number	Specifies the subinterface number in the range 1 to 4294967293. The *interface-number* that precedes the period (.) must match the *interface-number* this subinterface belongs to.
multipoint | **point-to-point**	(Optional.) Specifies a multipoint or point-to-point subinterface. Default is **multipoint**.
slot	On the Cisco 7000, specifies the backplane slot number; can be 0, 1, 2, 3, or 4. The slots are numbered from left to right.
port	On the Cisco 7000, specifies the port number of the interface. It can be 0, 1, 2, 3, 4, or 5 depending on the type of interface, as follows: • EIP (Ethernet Interface Processor) 0, 1, 2, 3, 4, or 5 • FIP (FDDI Interface Processor) 0 • FSIP (Fast Serial Interface Processor) 0, 1, 2, or 3 • HIP (HSSI Interface Processor) 0 • TRIP (Token Ring Interface Processor) 0, 1, 2, or 3 Ports on each interface processor are numbered from the top down.

Table 1-7 Interface Type Keywords

Keyword	Interface Type
async	Auxiliary port line used as an asynchronous interface.
bri	Integrated Services Digital Network (ISDN) Basic Rate Interface (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.

Keyword	Interface Type
fddi	Fiber Distributed Data Interface (FDDI).
hssi	High-Speed Serial Interface (HSSI).
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.
serial	Serial interface.
tokenring	Token Ring interface.
tunnel	Tunnel interface; a virtual interface. The *interface-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.

Default

The default mode for subinterfaces is **multipoint**.

Command Mode

Global configuration

Usage Guidelines

Subinterfaces can be configured to support partially meshed Frame Relay networks and multiple IPX encapsulations on LAN media (refer to Chapter 6 of the *Router Products Configuration Guide*).

Examples

In the following example, serial interface 0 is configured with PPP encapsulation:

```
interface serial 0
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:

```
interface loopback 0
ip address 131.108.1.1 255.255.255.0
```

The following example for the Cisco 7000 shows the interface configuration command for Ethernet port 4 on the EIP that is installed in (or recently removed from) slot 2:

```
interface ethernet 2/4
```

The following example begins configuration on the Token Ring interface processor in slot 1 on port 0 of a Cisco 7000:

```
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 PVCs associated, and subinterface serial 0.2 is configured as a point-to-point subinterface.

```
interface serial 0
encapsulation frame-relay
interface serial 0.1 multipoint
ip address 131.108.10.1 255.255.255.0
frame-relay interface-dlci 42 broadcast
frame-relay interface-dlci 53 broadcast
interface serial 0.2 point-to-point
ip address 131.108.11.1 255.255.0
frame-relay interface-dlci 59 broadcast
```

Related Commands

ppp
show interfaces
slip

invert-transmit-clock

Delays between the SCTE clock and data transmission indicate that the transmit clock signal might not be appropriate for the interface rate and length of cable being used. Different ends of the wire may have variances that differ slightly. Invert the clock signal to compensate for these factors, using the **invert-transmit-clock** interface configuration command. This command applies to a Cisco 7000.

invert-transmit-clock
no invert-transmit-clock

Syntax Description

This command has no keywords or arguments.

Command Mode

Interface configuration

Example

In the following example, the clock signal on serial interface 3/0 is inverted.

```
interface serial 3/0
invert-transmit-clock
```

isdn spid1

Use the **isdn spid1** interface configuration command to define at the router the SPID number that has been assigned by the ISDN service provider for the B1 channel. Use the **no isdn spid1** command to disable the specified SPID, thereby preventing access to the switch. If you include the LDN in the **no** form of this command, the access to the switch is permitted, but the other B channel may not be able to receive incoming calls.

isdn spid1 *spid-number* [*ldn*]
no isdn spid1 *spid-number* [*ldn*]

Syntax Description

spid-number	Number identifying the service to which you have subscribed. This value is assigned by the ISDN service provider and is usually a ten-digit telephone number with some extra digits.
ldn	(Optional.) Local directory number. This is a seven-digit number also assigned by the service provider.

Default

None

Command Mode

Interface configuration

Usage Guidelines

You must define the LDN if you want to receive any incoming calls on the B2-channel. The ISDN switch (DMS-100) checks for the LDN to determine whether both channels can be used to transmit and receive data. If the LDN is not present, then only the B1-channel can be used for full duplex communication. However, the other channel can still be used for making outgoing calls.

Example

The following example defines, on the router, a SPID and LDN for the B1 channel:

```
isdn spid1 415555121301 5551215
```

isdn spid2

Use the **isdn spid2** interface configuration command to define at the router the SPID number that has been assigned by the ISDN service provider for the B2 channel. Use the **no isdn spid2** command to disable the specified SPID, thereby preventing access to the switch. If you include the LDN in the **no** form of this command, the access to the switch is permitted, but the other B channel may not be able to receive incoming calls.

isdn spid2 *spid-number* [*ldn*]
no isdn spid2 *spid-number* [*ldn*]

Syntax Description

spid-number	Number identifying the service to which you have subscribed. This value is assigned by the ISDN service provider and is usually a ten-digit telephone number with some extra digits.
ldn	(Optional.) Local directory number. This is a seven-digit number also assigned by the service provider.

Default

None

Command Mode

Interface configuration

Usage Guidelines

You must define the LDN if you want to receive any incoming calls on the B1-channel. The ISDN switch (DMS-100) checks for the LDN to determine whether both channels can be used to transmit and receive data. If the LDN is not present, then only the B2-channel can be used for full duplex communication. However, the other channel can still be used for making outgoing calls.

Example

The following example defines, on the router, a SPID and LDN for the B2 channel:

```
isdn spid2 415555121202 5551214
```

isdn switch-type

To configure a central office switch on the ISDN interface, use the **isdn switch-type** global configuration command.

isdn switch-type *switch-type*

Syntax Description

switch-type Central office switch type; see Table 1-8 for a list of supported switches.

Table 1-8 ISDN Office Switch Types

Keyword	Switch Type
basic-1tr6	German switch standard
basic-5ess	ATT 5ESS
basic-dms100	NT DMS-100
basic-net3	Switch type for NET3 in UK and Europe
basic-ni1	National ISDN-1
ntt	NTT ISDN switch (Japan)
vn2	French VN2 standard
vn3	French VN3 standard

Default

None

Command Mode

Global configuration

Example

The following example configures the French VN2 ISDN switch type:

```
isdn switch-type vn2
```

isdn tei

To configure when ISDN terminal endpoint identifier (TEI) negotiation should occur, use the **isdn tei** global configuration command. Use the **no** form of this command to restore the default.

isdn tei [**first-call** | **powerup**]
no isdn tei

Syntax Description

first-call	ISDN TEI negotiation should occur when the first ISDN call is placed or received.
powerup	ISDN TEI negotiation should occur when the router is powered on.

Default

powerup

Command Mode

Global configuration

Usage Guidelines

Use this command with care.

Example

The following example configures the router to negotiate TEI when the first ISDN call is placed or received:

```
isdn tei first-call
```

loopback

To diagnose equipment malfunctions between interface and device, use the **loopback** interface configuration command. The **no loopback** command disables the test.

loopback
no loopback

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

On HSSI serial interface cards, the loopback function configures a two-way internal and external loop on the HSA applique of the specific interface.

On MCI and SCI serial interface cards, the loopback functions when a CSU/DSU or equivalent device is attached to the router. The **loopback** command loops the packets through the CSU/DSU to configure a CSU loop, when the device supports this feature.

On the MCI and MEC Ethernet cards, the interface receives back every packet it sends when the **loopback** command is enabled. Loopback operation has the additional effect of disconnecting network server functionality from the network.

On the CSC-FCI FDDI card, the interface receives back every packet it sends when the **loopback** command is enabled. Loopback operation has the additional effect of disconnecting network server functionality from the network.

On all Token Ring interface cards (except the 4-megabit CSC-R card), the interface receives back every packet it sends when the **loopback** command is enabled. Loopback operation has the additional effect of disconnecting network server functionality from the network.

Note Loopback does not work on an X.21 DTE because the X.21 interface definition does not include a loopback definition. The loopback tests do not work on the Ethernet interface of the IGS router product.

To show interfaces currently in loopback operation, use the **show interfaces loopback** EXEC command.

Example

The following example configures the loopback test on interface Ethernet 4:

```
interface ethernet 4
loopback
```

Related Commands

down-when-looped
show interfaces loopback

loopback applique

To configure an internal loop on the HSSI applique, use the **loopback** interface configuration command. To remove the loop, use the **no** form of the command.

loopback applique
no loopback applique

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This command loops the packets within the applique, thus providing a way to test for communication within the router. It is useful for sending pings to yourself to check functionality of the applique.

To show interfaces currently in loopback operation, use the **show interfaces loopback** EXEC command.

Example

The following example configures the loopback test on the HSSI applique:

```
interface serial 1
loopback applique
```

Related Command

show interfaces loopback

loopback dte

To loop packets to DTE internally within the CSU/DSU at the DTE interface, when the device supports this feature, use the **loopback** interface configuration command. To remove the loop, use the **no** form of the command.

loopback dte
no loopback dte

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This command is useful for testing the DTE-to-DCE cable.

To show interfaces currently in loopback operation, use the **show interfaces loopback** EXEC command.

Example

The following example configures the loopback test on the DTE interface:

```
interface serial 1
loopback dte
```

Related Command

show interfaces loopback

loopback line

To loop packets completely through the CSU/DSU to configure the CSU loop, when the device supports this feature, use the **loopback line** interface configuration command. To remove the loop, use the **no** form of the command.

loopback line
no loopback line

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This command is useful for testing the DCE device (CSU/DSU) itself.

To show interfaces currently in loopback operation, use the **show interfaces loopback** EXEC command.

Example

The following example configures the loopback test on the DCE device:

```
interface serial 1
loopback line
```

Related Command

show interfaces loopback

loopback remote

To loop packets completely through the CSU/DSU, over the DS3 link, to the remote CSU/DSU and back, use the **loopback remote** interface configuration command. To remove the loop, use the **no** form of the command.

loopback remote
no loopback remote

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This command applies only when the device supports the remote function. It is used for testing the data communication channels. The loopback usually is performed at the line port, rather than the DTE port, of the remote CSU/DSU.

To show interfaces currently in loopback operation, use the **show interfaces loopback** EXEC command.

Example

The following example configures a remote loopback test:

```
interface serial 0
loopback remote
```

Related Command

show interfaces loopback

media-type

To specify the Ethernet Network Interface Module configuration on the Cisco 4000 platform, use the **media-type** interface configuration command.

media-type [**aui** | **10baset**]
no media-type [**aui** | **10baset**]

Syntax Description

aui	(Optional.) Selects a 15-pin physical connection.
10baset	(Optional.) Selects an RJ45 10BaseT physical connection.

Default

AUI 15-pin physical connection

Command Mode

Interface configuration

Example

The following example selects an RJ45 10BaseT physical connection on interface Ethernet 1:

```
interface ethernet 1
media-type 10baset
```

mop enabled

To enable an interface to support the Maintenance Operation Protocol (MOP), use the **mop enabled** interface configuration command. To disable MOP on an interface, use the **no mop enabled** command.

mop enabled
no mop enabled

Syntax Description

This command has no arguments or keywords.

Default

Enabled by default on Ethernet interfaces and disabled on all other interfaces.

Command Mode

Interface configuration

Example

In the following example, MOP is enabled for serial interface 0:

```
Router(config)# interface serial0
Router(config-if)# mop enabled
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

mop sysid
mop retransmit-timer †
mop retries †

mop sysid

To enable an interface to send out period Maintenance Operation Protocol (MOP) system identification messages, use the **mop sysid** interface configuration command. To disable MOP message support on an interface, use the **no mop sysid** command.

mop sysid
no mop sysid

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Usage Guidelines

You can still run MOP without having the background system ID messages sent. This lets you use the MOP remote console, but does not generate messages used by the configurator.

Example

In the following example, serial interface 0 is enabled to send MOP system identification messages:

```
Router(config)# interface serial0
Router(config-if)# mop sysid
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

mop device-code †
mop enabled

mtu

To adjust the maximum packet size or maximum transmission unit (MTU) size, use the **mtu** interface configuration command. Use the **no mtu** command to restore the MTU value to its original default value.

mtu *bytes*
no mtu

Syntax Description

bytes	Desired size in bytes.

Default

Table 1-9 lists default MTU values according to media type.

Table 1-9 Default Media MTU Values

Media Type	Default MTU
Ethernet	1500
Serial	1500
Token Ring	4464
FDDI	4470
HSSI (HSA)	4470

Command Mode

Interface configuration

Usage Guidelines

Each interface has a default maximum packet size or maximum transmission unit (MTU) size. This number generally defaults to the largest size possible for that type interface. On serial interfaces, the MTU size varies, but cannot be set smaller than 64 bytes.

Note Changing the MTU value with the **mtu** interface configuration command can affect values for the protocol-specific versions of the command (**ip mtu** for example). If the values specified with the **ip mtu** interface configuration command is the same as the value specified with the **mtu** interface configuration command, and you change the value for the **mtu** interface configuration command, the **ip mtu** value automatically matches the new **mtu** interface configuration command value. However, changing the values for the **ip mtu** configuration commands has no effect on the value for the **mtu** interface configuration command.

Example

The following example specifies an MTU of 1000 bytes:

```
interface serial 1
mtu 1000
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

encapsulation smds †
ip mtu †

nrzi-encoding

To enable non-return to zero inverted (NRZI) line coding format, use the **nrzi-encoding** interface configuration command. Use the **no** form of the command to disable this capability.

nrzi-encoding
no nrzi-encoding

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

All FSIP interface types support nonreturn to zero (NRZ) and nonreturn to zero inverted (NRZI) format. This is a line coding format that is required for serial connections in some environments. NRZ encoding is most common. NRZI encoding is used primarily with RS-232 connections in IBM environments.

Example

In the following example, serial interface 1 is configured for NRZI encoding:

```
interface serial 1
nrzi-encoding
```

ntp broadcast

To specify that a specific interface should send Network Time Protocol (NTP) broadcast packets, use the **ntp broadcast** interface configuration command. Use the **no** form of the command to disable this capability.

ntp broadcast [**version** *number*]
no ntp broadcast

Syntax Description

version *number*	(Optional.) Number from 1 to 3 indicating the NTP version.

Default

Disabled

Command Mode

Interface configuration

Examples

In the following example, interface Ethernet0 is configured to send NTP version 2 packets:

```
Router(config)# interface ethernet0
Router(config-if)# ntp broadcast version 2
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

ntp broadcast client
ntp broadcastdelay †

ntp broadcast client

To allow the system to receive NTP broadcast packets on an interface, use the **ntp broadcast client** command. Use the **no** form of the command to disable this capability.

ntp broadcast client
no ntp broadcast client

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

Use this command to allow the system to listen to broadcast packets on an interface-by-interface basis.

Example

In the following example, the router synchronizes to NTP packets broadcasted on interface Ethernet1:

```
interface ethernet1
ntp broadcast client
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

ntp broadcast
ntp broadcastdelay †

ntp disable

To prevent an interface from receiving NTP packets, use the **ntp disable** interface configuration command. To enable receipt of NTP packets on an interface, use the **no ntp disable** command.

ntp disable
no ntp disable

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Usage Guidelines

This command provides a simple method of access control.

Example

In the following example, interface Ethernet 1 is prevented from receiving NTP packets:

```
Router(config)# interface ethernet0
Router(config-if)# ntp disable
```

ppp

To make an asynchronous connection from the auxiliary port using the PPP protocol, enter the **ppp** EXEC command.

ppp [**default** | *client* [**@***tacacs-server*]] [**/routing**]

Syntax Description

default	(Optional.) Makes PPP connection when a default address has been configured.
client	(Optional.) IP address or the name of the client workstation or PC.
tacacs-server	(Optional.) IP address or IP host name of the TACACS server to which the user's TACACS authentication request is to be sent.
/routing	(Optional.) Indicates asynchronous routing is enabled.

Default

None

Command Mode

EXEC

Usage Guidelines

The IP address (indicated by the *client* and *tacacs-server* arguments) is the IP address of the system dialing in. This parameter can only be specified if the line is set for dynamic addresses using the line configuration command **async dynamic address**.

If you do not specify an address or enter **default**, you will be prompted for an IP address or host name. You can enter **default** at this point to use the default address configured for the line.

Using the *tacacs-server* argument, you can specify a TACACS server. If you do not specify a TACACS server for PPP address authentication, the TACACS server specified at login (if any) will be used for the PPP address query.

The **/routing** switch indicates that the remote system is a router and that routing messages should be exchanged over the link. The **/routing** switch can only be used if the line is configured for **async dynamic routing**. Static routing is always used.

You cannot configure PPP on the console port.

Example

The following example shows a PPP EXEC command that specifies a TACACS authentication server and configured the line for routing:

```
Router> ppp ntpc@server1 /routing
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

async default ip address
async dynamic address
async dynamic routing
async mode dedicated
async mode interactive
encapsulation ppp
interface async 1
tacacs-server †

ppp authentication chap

To enable Challenge Handshake Authentication Protocol (CHAP) on a serial interface, use the **ppp authentication chap** interface configuration command. Use the **no ppp authentication chap** command to disable this encapsulation.

ppp authentication chap [**if-needed**]
no ppp authentication chap

Syntax Description

if-needed	(Optional.) Indicates that the system will not perform CHAP authentication if the user has already been authenticated. This option applies only to asynchronous and virtual asynchronous interfaces.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

Once you have enabled CHAP, the local router requires a password from remote devices. If the remote device does not support CHAP, no traffic will be passed to that device.

Example

The following example enables CHAP on serial interface 4:

```
interface serial 4
encapsulation ppp
ppp authentication chap
```

Related Command

encapsulation ppp

ppp authentication pap

To enable the Password Authentication Protocol (PAP) on a serial interface, use the **ppp authenticate pap** interface configuration command. To disable this feature, use the **no** form of this command.

ppp authentication pap [**if-needed**]
no ppp authentication pap

Syntax Description

if-needed	(Optional.) Indicates that the system will not perform PAP authentication if the user has already been authenticated. This option applies only to asynchronous and virtual asynchronous interfaces.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

When you enable PAP, it forces the router to negotiate with the remote device for the Password Authentication Protocol. If the remote device does not support PAP, no traffic will be passed to that device.

If both sides of the serial link are running PPP and PAP, both username entries and their passwords must be present on both sides of the link. You define these with the **username** *name* **password** *secret* global configuration command.

Example

The following example enables PAP on serial interface 4:

```
interface serial 4
ppp authentication pap
```

Related Commands

encapsulation ppp
username password

ppp quality

To enable Link Quality Monitoring (LQM) on a serial interface, use the **ppp quality** interface configuration command. Use the **no** form of this command to disable LQM.

ppp quality *percentage*
no ppp quality

Syntax Description

percentage	Specifies the link quality threshold. Range is 1 to 100.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

The percentages are calculated for both incoming and outgoing directions. The outgoing quality is calculated by comparing the total number of packets and bytes sent to the total number of packets and bytes received by the peer. The incoming quality is calculated by comparing the total number of packets and bytes received to the total number of packets and bytes sent by the peer.

If the link quality percentage is not maintained, the link is deemed to be of poor quality and is taken down. The policy implements a time lag so that the link does not bounce up and down.

Example

The following example enables LQM on interface serial 4:

```
interface serial 4
encapsulation ppp
ppp quality 80
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

encapsulation ppp
keepalive †

priority-group

To assign the specified priority list to an interface, use the **priority-group** interface configuration command. Use the **no priority-group** command to remove the specified **priority-group** assignment.

priority-group *list*
no priority-group

Syntax Description

list Priority list number assigned to the interface.

Default

None

Command Mode

Interface configuration

Usage Guidelines

Only one list can be assigned per interface. Priority output queueing provides a mechanism to prioritize packets transmitted on an interface.

Example

The following example causes packets on interface serial 0 to be classified by priority list 1:

```
interface serial 0
priority-group 1
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

priority-list †
priority-list interface †
priority-list queue-limit †
priority-list stun †

pulse-time

To enable pulsing DTR signal intervals on the serial interfaces, use the **pulse-time** interface configuration command. Use the **no pulse-time** command to restore the default interval.

pulse-time *seconds*
no pulse-time

Syntax Description

seconds	Integer that specifies the DTR signal interval in seconds.

Default

0 seconds

Command Mode

Interface configuration

Usage Guidelines

When the serial line protocol goes down (for example, because of loss of synchronization) the interface hardware is reset and the DTR signal is held inactive for at least the specified interval. This function is useful for handling encrypting or other similar devices that use the toggling of the DTR signal to resynchronize.

Example

The following example enables DTR pulse signals for three seconds on interface serial:

```
interface serial 2
pulse-time 3
```

ring-speed

To set the ring speed for the CSC-1R, CSC-2R, and IGS/TR Token Ring interfaces, use the **ring-speed** interface configuration command.

ring-speed *speed*

Syntax Description

speed	Integer that specifies the ring speed, either 4 for 4-Mbps or 16 for 16-Mbps operation.

Default

16-Mbps operation

Caution Configuring a ring speed that is wrong or incompatible with the connected Token Ring will cause the ring to beacon, which effectively takes the ring down and makes it nonoperational.

Command Mode

Interface configuration

Example

The following example sets a Token Ring interface ring speed to 4 Mbps:

```
interface tokenring 0
ring-speed 4
```

scheduler-interval

To control the maximum amount of time that can elapse without running the lowest priority system processes, use the **scheduler-interval** global configuration command. Use the **no** form of this command to restore the default.

scheduler-interval *milliseconds*
no scheduler-interval

Syntax Description

milliseconds	An integer that specifies the interval in milliseconds. The minimum interval that you can specify is 500 milliseconds; there is no maximum value.

Default

The default is to allow high-priority operations to use as much of the central processor as needed.

Command Mode

Global configuration

Usage Guidelines

The normal operation of the network server allows the switching operations to use as much of the central processor as is required. If the network is running unusually heavy loads that do not allow the processor the time to handle the routing protocols, give priority to the system process scheduler.

Example

The following example changes the low-priority process schedule to an interval of 750 milliseconds:

```
scheduler-interval 750
```

show async-bootp

Use the **show async-bootp** EXEC command to display the parameters that have been configured for SLIP extended BOOTP requests.

show async-bootp

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is a sample output of the **show async-bootp** command:

```
Router# show async-bootp

The following extended data will be sent in BOOTP responses:

bootfile (for address 128.128.1.1) "pcboot"
bootfile (for address 131.108.1.111) "dirtboot"
subnet-mask 255.255.0.0
time-offset -3600
time-server 128.128.1.1
```

Table 1-10 describes significant fields shown in the display.

Table 1-10 Show Async-Bootp Field Descriptions

Field	Description
bootfile... “pcboot”	Boot file for address 128.128.1.1 is named pcboot.
subnet-mask 255.255.0.0	Subnet mask.
time-offset -3600	Local time is one hour (3600 seconds) earlier than UTC time.
time-server 128.128.1.1	Address of the time server for the network.

Related Command

async-bootp

show async status

To list the status of the asynchronous interface 1 associated with the router auxiliary port, use the **show async status** EXEC command:

show async status

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

Shows all asynchronous sessions, whether they are using SLIP or PPP encapsulation.

Sample Display

The following is sample output from the **show async status** command:

```
Router> show async status
Async protocol statistics:
  Rcvd: 5448 packets, 7682760 bytes
        1 format errors, 0 checksum errors, 0 overrun, 0 no buffer
  Sent: 5455 packets, 7682676 bytes, 0 dropped


 Int           Local          Remote Qd InPack OutPac Inerr  Drops  MTU Qsz
   1     192.31.7.84         Dynamic  0      0      0     0      0 1500  10
```

Table 1-11 describes significant fields shown in the display.

Table 1-11 Asynchronous Statistics Display Field Descriptions

Field	Description
Rcvd:	Statistics on packets received.
5548 packets	Packets received.
7682760 bytes	Total number of bytes.
1 format errors	Packets with a bad IP header, even before the checksum is calculated.
0 checksum errors	Count of checksum errors.
0 overrun	Number of giants received.
0 no buffer	Number of packets received when no buffer was available.
Sent:	Statistics on packets sent.
5455 packets	Packets sent.
7682676 bytes	Total number of bytes.
0 dropped	Number of packets dropped.
Int	Interface number.
*	Line currently in use.

Field	Description
Local	Local IP address on the link.
Remote	Remote IP address on the link; “Dynamic” indicates that a remote address is allowed but has not been specified; “None” indicates that no remote address is assigned or being used.
Qd	Number of packets on hold queue (Qsz is max).
InPack	Number of packets received.
OutPac	Number of packets sent.
Inerr	Number of total input errors; sum of format errors, checksum errors, overruns and no buffers.
Drops	Number of packets received that would not fit on the hold queue.
MTU	Current maximum transmission unit size.
Qsz	Current output hold queue size.

Related Commands

async
interface async

show controllers bri

To display information about the ISDN Basic Rate Interface (BRI) on a Cisco 3000, use the **show controllers bri** EXEC command.

show controllers bri

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show controllers bri** command:

```
Router# sho cont b 0
BRI unit 0
D Chan Info:
Layer 1 is ACTIVATED
idb 0x32089C, ds 0x3267D8, reset_mask 0x2
buffer size 1524
RX ring with 2 entries at 0x2101600 : Rxhead 0
00 pak=0x4122E8 ds=0x412444 status=D000 pak_size=0
01 pak=0x410C20 ds=0x410D7C status=F000 pak_size=0
TX ring with 1 entries at 0x2101640: tx_count = 0, tx_head = 0, tx_tail = 0
00 pak=0x000000 ds=0x000000 status=7C00 pak_size=0
0 missed datagrams, 0 overruns, 0 bad frame addresses
0 bad datagram encapsulations, 0 memory errors
0 transmitter underruns
B1 Chan Info:
Layer 1 is ACTIVATED
idb 0x3224E8, ds 0x3268C8, reset_mask 0x0
buffer size 1524
RX ring with 8 entries at 0x2101400 : Rxhead 0
00 pak=0x421FC0 ds=0x42211C status=D000 pak_size=0
01 pak=0x4085E8 ds=0x408744 status=D000 pak_size=0
02 pak=0x422EF0 ds=0x42304C status=D000 pak_size=0
03 pak=0x4148E0 ds=0x414A3C status=D000 pak_size=0
04 pak=0x424D50 ds=0x424EAC status=D000 pak_size=0
05 pak=0x423688 ds=0x4237E4 status=D000 pak_size=0
06 pak=0x41AB98 ds=0x41ACF4 status=D000 pak_size=0
07 pak=0x41A400 ds=0x41A55C status=F000 pak_size=0
TX ring with 4 entries at 0x2101440: tx_count = 0, tx_head = 0, tx_tail = 0
00 pak=0x000000 ds=0x000000 status=5C00 pak_size=0
01 pak=0x000000 ds=0x000000 status=5C00 pak_size=0
02 pak=0x000000 ds=0x000000 status=5C00 pak_size=0
03 pak=0x000000 ds=0x000000 status=7C00 pak_size=0
0 missed datagrams, 0 overruns, 0 bad frame addresses
0 bad datagram encapsulations, 0 memory errors
0 transmitter underruns
B2 Chan Info:
Layer 1 is ACTIVATED
idb 0x324520, ds 0x3269B8, reset_mask 0x2
buffer size 1524
RX ring with 8 entries at 0x2101500 : Rxhead 0
00 pak=0x40FCF0 ds=0x40FE4C status=D000 pak_size=0
01 pak=0x40E628 ds=0x40E784 status=D000 pak_size=0
```

```
02 pak=0x40F558 ds=0x40F6B4 status=D000 pak_size=0
03 pak=0x413218 ds=0x413374 status=D000 pak_size=0
04 pak=0x40EDC0 ds=0x40EF1C status=D000 pak_size=0
05 pak=0x4113B8 ds=0x411514 status=D000 pak_size=0
06 pak=0x416ED8 ds=0x417034 status=D000 pak_size=0
07 pak=0x416740 ds=0x41689C status=F000 pak_size=0
TX ring with 4 entries at 0x2101540: tx_count = 0, tx_head = 0, tx_tail = 0
00 pak=0x000000 ds=0x000000 status=5C00 pak_size=0
01 pak=0x000000 ds=0x000000 status=5C00 pak_size=0
02 pak=0x000000 ds=0x000000 status=5C00 pak_size=0
03 pak=0x000000 ds=0x000000 status=7C00 pak_size=0
0 missed datagrams, 0 overruns, 0 bad frame addresses
0 bad datagram encapsulations, 0 memory errors
0 transmitter underruns
```

Table 1-12 describes the significant fields in the display.

Table 1-12 Show Controllers BRI Field Descriptions

Field	Description
BRI unit 0	Interface type and unit number.
Chan Info	D and B channel numbers
Layer 1 is ACTIVATED	Status can be DEACTIVATED, PENDING ACTIVATION, or ACTIVATED.
idb ds reset_mask	Information about internal data structures and parameters.
buffer size	Number of bytes allocated for buffers
RX ring with - entries at -	Information about the Receiver Queue.
Rxhead	Start of the Receiver Queue.
pak ds status pak_size	Information about internal data structures and parameters.
TX ring with - entries at -	Information about the Transmitter Queue.
tx_count	Number of packets to transmit.
tx_head	Start of the transmit list.
tx_tail	End of the transmit list.
missed datagrams	Incoming packets missed due to internal errors.
overruns	Number of times the receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver's ability to handle the data.
bad frame addresses	Frames received with a CRC error and noninteger number of octets.
bad datagram encapsulations	Packets received with bad encapsulation.
memory errors	Internal DMA memory errors.
transmitter underruns	Number of times that the transmitter has been running faster than the router can handle.

show controllers cbus

Use the **show controllers cbus** EXEC command on the AGS+ to display all information under the CiscoBus controller card. This command also shows the capabilities of the card and reports controller-related failures.

show controllers cbus

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show controllers cbus** command:

```
Router# show controllers cbus
cBus 1, controller type 3.0, microcode version 2.0
  128 Kbytes of main memory, 32 Kbytes cache memory
  40 1520 byte buffers, 14 4484 byte buffers
  Restarts: 0 line down, 0 hung output, 0 controller error
 --More--
HSCI 1, controller type 10.0, microcode version 129.3
  Interface 6 - Hssi0, electrical interface is Hssi DTE
    5 buffer RX queue threshold, 7 buffer TX queue limit, buffer size 1520
    ift 0004, rql 2, tq 0000 0000, tql 7
    Transmitter delay is 0 microseconds
 MEC 3, controller type 5.1, microcode version 130.6
  Interface 18 - Ethernet2, station address 0000.0c02.a03c (bia 0000.0c02.a03c)
    10 buffer RX queue threshold, 7 buffer TX queue limit, buffer size 1520
    ift 0000, rql 10, tq 0000 0000, tql 7
    Transmitter delay is 0 microseconds
  Interface 19 - Ethernet3, station address 0000.0c02.a03d (bia 0000.0c02.a03d)
    10 buffer RX queue threshold, 7 buffer TX queue limit, buffer size 1520
    ift 0000, rql 10, tq 0000 0000, tql 7
    Transmitter delay is 0 microseconds
```

Table 1-13 describes the fields shown in the following lines of output from the display.

```
cBus 1, controller type 3.0, microcode version 2.0
  128 Kbytes of main memory, 32 Kbytes cache memory
  40 1520 byte buffers, 14 4484 byte buffers
  Restarts: 0 line down, 0 hung output, 0 controller error
```

Table 1-13 Show Controllers cBus Field Descriptions—Part 1

Field	Description
cBus 1	Card type and unit number (varies depending on card).
controller type 3.0	Version number of the card.
microcode version 2.0	Version number of the card's internal software (in read-only memory).
128 Kbytes of main memory	Amount of main memory on the card.
32 Kbytes cache memory	Amount of cache memory on the card.
40 1520 byte buffers	Number of buffers of this size on the card.
14 4484 byte buffers	Number of buffers of this size on the card.
Restarts 0 line down 0 hung output 0 controller error	Count of restarts due to the following conditions: Communication line down Output unable to transmit Internal error

Table 1-14 describes the fields shown in the following lines of output from the display:

```
HSCI 1, controller type 10.0, microcode version 129.3
  Interface 6 - Hssi0, electrical interface is Hssi DTE
    5 buffer RX queue threshold, 7 buffer TX queue limit, buffer size 1520
    ift 0004, rql 2, tq 0000 0000, tql 7
    Transmitter delay is 0 microseconds
```

Table 1-14 Show Controllers cBus Field Descriptions—Part 2

Field	Description
HSCI 1	Card type and unit number (varies depending on card).
controller type 10.0	Version number of the card.
microcode version 129.3	Version number of the card's internal software (in read-only memory).
Interface 6	Physical interface number.
Hssi 0	Logical name for this interface.
electrical interface is Hssi DTE	Self-explanatory.
5 buffer RX queue threshold	Maximum number of buffers allowed in the receive queue.
7 buffer TX queue limit	Maximum number of buffers allowed in the transmit queue.
buffer size 1520	Size of the buffers on this card (in bytes).
ift 0004	Interface type code.
rql 2	Current number of buffers allowed for the receive queue.
tq 0000 0000	Transmit queue head/tail pointers.
tql 7	Current number of buffers allowed for transmit queue.
Transmitter delay is 0 microseconds	Transmitter delay between the packets.

show controllers cxbus

The **show controllers cxbus** EXEC command displays information about the switch processor (SP) CxBus controller on the Cisco 7000. This command displays information the system uses for bridging and routing that is specific to the interface hardware. The information displayed is generally useful for diagnostic tasks performed by technical support personnel only.

show controllers cxbus

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output on the Cisco 7000 from the **show controllers cxbus** command:

```
Router# show controllers cxbus
Switch Processor 5, hardware version 11.1, microcode version 130.2
  512 Kbytes of main memory, 128 Kbytes cache memory
  120 1520 byte buffers, 70 4484 byte buffers, 212 byte system buffer
Restarts: 0 line down, 0 hung output, 0 controller error
 FIP 3, hardware version 6.1, microcode version 141.7
  Interface 24 - Fddi3/0, station addr 0000.0c02.adf1 (bia 0000.0c02.adf1)
    70 buffer RX queue threshold, 71 buffer TX queue limit, buffer size 4484
    ift 0006, rql 66, tq 0000 0000, tql 70
 EIP 4, hardware version 5.1, microcode version 128.10
  Interface 32 - Ethernet4/0, station addr 0000.0c02.d0cc (bia 0000.0c02.d0cc)
    20 buffer RX queue threshold, 28 buffer TX queue limit, buffer size 1520
    ift 0000, rql 20, tq 0000 0000, tql 28
    Transmitter delay is 0 microseconds
  Interface 33 - Ethernet4/1, station addr 0000.0c02.d0cd (bia 0000.0c02.d0cd)
    20 buffer RX queue threshold, 28 buffer TX queue limit, buffer size 1520
    ift 0000, rql 20, tq 0000 0000, tql 28
    Transmitter delay is 0 microseconds
  Interface 34 - Ethernet4/2, station addr 0000.0c02.d0ce (bia 0000.0c02.d0ce)
    20 buffer RX queue threshold, 28 buffer TX queue limit, buffer size 1520
    ift 0000, rql 20, tq 0000 0000, tql 28
    Transmitter delay is 0 microseconds
  Interface 35 - Ethernet4/3, station addr 0000.0c02.d0cf (bia 0000.0c02.d0cf)
    20 buffer RX queue threshold, 28 buffer TX queue limit, buffer size 1520
    ift 0000, rql 20, tq 0000 0000, tql 28
    Transmitter delay is 0 microseconds
  Interface 36 - Ethernet4/4, station addr 0000.0c02.d0d0 (bia 0000.0c02.d0d0)
    20 buffer RX queue threshold, 28 buffer TX queue limit, buffer size 1520
    ift 0000, rql 20, tq 0000 0000, tql 28
    Transmitter delay is 0 microseconds
  Interface 37 - Ethernet 4/5, station addr 0000.0c02.d0d1 (bia0000.0c02.d0d1)
    20 buffer RX queue threshold, 28 buffer TX queue limit, buffer size 1520
    ift 0000, rql 20, tq 0000 0000, tql 28
Transmitter delay is 0 microseconds
```

Table 1-15 describes the **show controllers cxbus** display fields.

Table 1-15 Show Controllers CxBus Field Descriptions

Field	Description
IP type, slot number	Unit type and slot number.
hardware version	Version number of the controller.
microcode version	Version number of the controller's internal software (in read-only memory).
main memory cache memory	Amount of main and cache memory on the processor .
Restarts line down hung output controller error	Number of restarts due to the following conditions: Communication line down Output unable to transmit Internal error
Interface # -	Names of interfaces by CxBus interface type, slot, and port number.
RX buffers	Number of buffers for received packets.
TX queue limit	Maximum number of buffers in transmit queue.
Transmitter delay	Delay between outgoing frames.
Station address	The hardware address of the interface.

show controllers ethernet

To display information about the Ethernet controller cards in a Cisco 2500, Cisco 3000 series, or Cisco 4000 router, use the **show controllers ethernet** EXEC command:

show controllers ethernet

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show controllers ethernet** command on the Cisco 4000:

```
router# show controller ethernet
LANCE unit 0, NIM slot 1, NIM type code 4, NIM version 1
Media Type is 10BaseT, Link State is Up, Squelch is Normal
idb 0x4060, ds 0x5C80, regaddr = 0x8100000
IB at 0x600D7AC: mode=0x0000, mcfilter 0000/0001/0000/0040
station address 0000.0c03.a14f  default station address 0000.0c03.a14f
buffer size 1524
RX ring with 32 entries at 0xD7E8
Rxhead = 0x600D8A0 (12582935), Rxp = 0x5CF0(23)
00 pak=0x60336D0 ds=0x6033822 status=0x80 max_size=1524 pak_size=98
01 pak=0x60327C0 ds=0x6032912 status=0x80 max_size=1524 pak_size=98
02 pak=0x6036B88 ds=0x6036CDA status=0x80 max_size=1524 pak_size=98
03 pak=0x6041138 ds=0x604128A status=0x80 max_size=1524 pak_size=98
04 pak=0x603FAA0 ds=0x603FBF2 status=0x80 max_size=1524 pak_size=98
05 pak=0x600DC50 ds=0x600DDA2 status=0x80 max_size=1524 pak_size=98
06 pak=0x6023E48 ds=0x6023F9A status=0x80 max_size=1524 pak_size=1506
07 pak=0x600E3D8 ds=0x600E52A status=0x80 max_size=1524 pak_size=1506
08 pak=0x6020990 ds=0x6020AE2 status=0x80 max_size=1524 pak_size=386
09 pak=0x602D4E8 ds=0x602D63A status=0x80 max_size=1524 pak_size=98
10 pak=0x603A7C8 ds=0x603A91A status=0x80 max_size=1524 pak_size=98
11 pak=0x601D4D8 ds=0x601D62A status=0x80 max_size=1524 pak_size=98
12 pak=0x603BE60 ds=0x603BFB2 status=0x80 max_size=1524 pak_size=98
13 pak=0x60318B0 ds=0x6031A02 status=0x80 max_size=1524 pak_size=98
14 pak=0x601CD50 ds=0x601CEA2 status=0x80 max_size=1524 pak_size=98
15 pak=0x602C5D8 ds=0x602C72A status=0x80 max_size=1524 pak_size=98
16 pak=0x60245D0 ds=0x6024722 status=0x80 max_size=1524 pak_size=98
17 pak=0x6008328 ds=0x600847A status=0x80 max_size=1524 pak_size=98
18 pak=0x601EB70 ds=0x601ECC2 status=0x80 max_size=1524 pak_size=98
19 pak=0x602DC70 ds=0x602DDC2 status=0x80 max_size=1524 pak_size=98
20 pak=0x60163E0 ds=0x6016532 status=0x80 max_size=1524 pak_size=98
21 pak=0x602CD60 ds=0x602CEB2 status=0x80 max_size=1524 pak_size=98
22 pak=0x6037A98 ds=0x6037BEA status=0x80 max_size=1524 pak_size=98
23 pak=0x602BE50 ds=0x602BFA2 status=0x80 max_size=1524 pak_size=98
24 pak=0x6018988 ds=0x6018ADA status=0x80 max_size=1524 pak_size=98
25 pak=0x6033E58 ds=0x6033FAA status=0x80 max_size=1524 pak_size=98
26 pak=0x601BE40 ds=0x601BF92 status=0x80 max_size=1524 pak_size=98
27 pak=0x6026B78 ds=0x6026CCA status=0x80 max_size=1524 pak_size=98
```

```
28 pak=0x6024D58 ds=0x6024EAA status=0x80 max_size=1524 pak_size=74
29 pak=0x602AF40 ds=0x602B092 status=0x80 max_size=1524 pak_size=98
30 pak=0x601FA80 ds=0x601FBD2 status=0x80 max_size=1524 pak_size=98
31 pak=0x6038220 ds=0x6038372 status=0x80 max_size=1524 pak_size=98
TX ring with 8 entries at 0xDA20, tx_count = 0
tx_head = 0x600DA58 (12582919), head_txp = 0x5DC4 (7)
tx_tail = 0x600DA58 (12582919), tail_txp = 0x5DC4 (7)
00 pak=0x000000 ds=0x600CF12 status=0x03 status2=0x0000 pak_size=118
01 pak=0x000000 ds=0x602126A status=0x03 status2=0x0000 pak_size=60
02 pak=0x000000 ds=0x600CF12 status=0x03 status2=0x0000 pak_size=118
03 pak=0x000000 ds=0x600CF12 status=0x03 status2=0x0000 pak_size=118
04 pak=0x000000 ds=0x600CF12 status=0x03 status2=0x0000 pak_size=118
05 pak=0x000000 ds=0x600CF12 status=0x03 status2=0x0000 pak_size=118
06 pak=0x000000 ds=0x600CF12 status=0x03 status2=0x0000 pak_size=118
07 pak=0x000000 ds=0x6003ED2 status=0x03 status2=0x0000 pak_size=126
0 missed datagrams, 0 overruns, 2 late collisions, 2 lost carrier events
0 transmitter underruns, 0 excessive collisions,  0 tdr, 0 babbles
0 memory errors, 0 spurious initialization done interrupts
0 no enp status, 0 buffer errors, 0 overflow errors
10 one_col, 10 more_col, 22 deferred, 0 tx_buff
0 throttled, 0 enabled
Lance csr0 = 0x73
```

show controllers fddi

Use the **show controllers fddi** EXEC command to display all information under the FDDI controller card on the AGS+ or FDDI Interface Processor (FIP) on the Cisco 7000.

show controllers fddi

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

This command reflects the internal state of the chips and information the system uses for bridging and routing that is specific to the interface hardware. The information displayed is generally useful for diagnostic tasks performed by technical support personnel only.

Sample Display

The following is sample output from the **show controllers fddi** command on the Cisco 7000:

```
Router> show controllers fddi
Fddi2/0 - hardware version 2.2, microcode version 1.2
  Phy-A registers:
    cr0 4, cr1 0, cr2 0, status 3, cr3 0
  Phy-B registers:
    cr0 4, cr1 4, cr2 0, status 3, cr3 0
  FORMAC registers:
    irdtlb  71C2, irdtneg F85E, irdthtt F5D5, irdmir  FFFF0BDC
    irdtrth F85F, irdtmax FBC5, irdtvxt 5959, irdstmc 0810
    irdmode 6A20, irdimsk 0000, irdstat 8060, irdtpri 0000
  FIP registers
    ccb:   002C  cmd:   0006  fr:   000F  mdptr: 0000  mema: 0000
    icb:   00C0  arg:   0003  app:  0004  mdpg:  0000  af:   0603
    clm:   E002  bcn:   E016  clbn: 0198  rxoff: 002A  en:   0001
    clmbc: 8011  bcnbc: 8011  robn: 0004  park:  0000  fop:  8004

    txchn: 0000  pend:  0000  act:  0000  tail:  0000  cnt:  0000
    state: 0003  check: 0000  eof:  0000  tail:  0000  cnt:  0000
    rxchn: 0000  buf0:  0534  nxt0: 0570  eof:   0000  tail: 0000
    eofch: 0000  buf1:  051C  nxt1: 0528  pool:  0050  err:  005C

    head:  0984  cur:   0000  t0:   0030  t1:    0027  t2:   000F
    tail:  0984  cnt:   0001  t3:   0000  rxlft: 000B  used: 0000
    txq_s: 0018  txq_f: 0018  Aarm: 0000  Barm:  1388  fint: 8004

 Total LEM: phy-a 6, phy-b 13
```

The last line of output indicates how many LEM events occurred on the specific PHY.

show controllers mci

Use the **show controllers mci** EXEC command to display all information under the SCI or the Multiport Communications Interface card. This command displays information the system uses for bridging and routing that is specific to the interface hardware. The information displayed is generally useful for diagnostic tasks performed by technical support personnel only.

show controllers mci

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show controllers mci** command:

```
Router# show controllers mci

MCI 0, controller type 1.1, microcode version 1.8
   128 Kbytes of main memory, 4 Kbytes cache memory
22 system TX buffers, largest buffer size 1520
   Restarts: 0 line down, 0 hung output, 0 controller error
Interface 0 is Ethernet0, station address 0000.0c00.d4a6
   15 total RX buffers, 11 buffer TX queue limit, buffer size 1520
   Transmitter delay is 0 microseconds
Interface 1 is Serial0, electrical interface is V.35 DTE
   15 total RX buffers, 11 buffer TX queue limit, buffer size 1520
   Transmitter delay is 0 microseconds
   High speed synchronous serial interface
Interface 2 is Ethernet1, station address aa00.0400.3be4
   15 total RX buffers, 11 buffer TX queue limit, buffer size 1520
   Transmitter delay is 0 microseconds
Interface 3 is Serial1, electrical interface is V.35 DCE
   15 total RX buffers, 11 buffer TX queue limit, buffer size 1520
   Transmitter delay is 0 microseconds
   High speed synchronous serial interface
```

Table 1-16 describes significant fields shown in the display.

Table 1-16 Show Controllers MCI Field Descriptions

Field	Description
MCI 0	Card type and unit number (varies depending on card).
controller type 1.1	Version number of the card.
microcode version 1.8	Version number of the card's internal software (in read-only memory).
128 Kbytes of main memory	Amount of main memory on the card.
4 Kbytes cache memory	Amount of cache memory on the card.
22 system TX buffers	Number of buffers that hold packets to be transmitted.
largest buffer size 1520	Largest size of these buffers (in bytes).
Restarts 0 line down 0 hung output 0 controller error	Count of restarts due to the following conditions: Communication line down Output unable to transmit Internal error
Interface 0 is Ethernet0	Names of interfaces, by number.
electrical interface is V.35 DTE	Line interface type for serial connections.
15 total RX buffers	Number of buffers for received packets.
11 buffer TX queue limit	Maximum number of buffers in transmit queue.
Transmitter delay is 0 microseconds	Delay between outgoing frames.
Station address 0000.0c00.d4a6	Hardware address of the interface.

Note The interface type is only queried at startup. If the hardware changes *subsequent* to initial startup, then the wrong type is reported. This has *no* adverse effect on the operation of the software. For instance, if a DCE cable is connected to a dual-mode V.35 applique after the unit has been booted, then the display presented for **show interfaces** incorrectly reports attachment to a DTE device although the software recognizes the DCE interface and behaves accordingly.

Related Command

tx-queue-limit

show controllers serial

Use the **show controllers serial** EXEC command on the Cisco 7000 to display information about the Fast Serial Interface Processor (FSIP). This command displays information the system uses for bridging and routing that is specific to the interface hardware. The information displayed is generally useful for diagnostic tasks performed by technical support personnel only.

show controllers serial

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

Sample output of the **show controllers serial** command on the Cisco 4000 follows:

```
Router# show controllers serial

MK5 unit 0, NIM slot 1, NIM type code 7, NIM version 1
idb = 0x60A8, driver structure at 0x31CE20, regaddr = 0x8100300
IB at 0x6045B30: mode=0x0108, local_addr=0, remote_addr=0
N1=1524, N2=1, scaler=100, T1=1000, T3=2000, TP=1
buffer size 1524
DTE V.35 serial cable attached
RX ring with 32 entries at 0x45B90 : RLEN=5, Rxhead 0
00 pak=0x60453A0  ds=0x6045500 status=80 max_size=1524 pak_size=0
01 pak=0x6044C10  ds=0x6044D70 status=80 max_size=1524 pak_size=0
02 pak=0x6044480  ds=0x60445E0 status=80 max_size=1524 pak_size=0
03 pak=0x6043CF0  ds=0x6043E50 status=80 max_size=1524 pak_size=0
04 pak=0x6043560  ds=0x60436C0 status=80 max_size=1524 pak_size=0
05 pak=0x6042DD0  ds=0x6042F30 status=80 max_size=1524 pak_size=0
06 pak=0x6042640  ds=0x60427A0 status=80 max_size=1524 pak_size=0
07 pak=0x6041EB0  ds=0x6042010 status=80 max_size=1524 pak_size=0
08 pak=0x6041720  ds=0x6041880 status=80 max_size=1524 pak_size=0
09 pak=0x6040F90  ds=0x60410F0 status=80 max_size=1524 pak_size=0
10 pak=0x6040800  ds=0x6040960 status=80 max_size=1524 pak_size=0
11 pak=0x6040070  ds=0x60401D0 status=80 max_size=1524 pak_size=0
12 pak=0x603F8E0  ds=0x603FA40 status=80 max_size=1524 pak_size=0
13 pak=0x603F150  ds=0x603F2B0 status=80 max_size=1524 pak_size=0
14 pak=0x603E9C0  ds=0x603EB20 status=80 max_size=1524 pak_size=0
15 pak=0x603E230  ds=0x603E390 status=80 max_size=1524 pak_size=0
16 pak=0x603DAA0  ds=0x603DC00 status=80 max_size=1524 pak_size=0
17 pak=0x603D310  ds=0x603D470 status=80 max_size=1524 pak_size=0
18 pak=0x603CB80  ds=0x603CCE0 status=80 max_size=1524 pak_size=0
19 pak=0x603C3F0  ds=0x603C550 status=80 max_size=1524 pak_size=0
20 pak=0x603BC60  ds=0x603BDC0 status=80 max_size=1524 pak_size=0
21 pak=0x603B4D0  ds=0x603B630 status=80 max_size=1524 pak_size=0
22 pak=0x603AD40  ds=0x603AEA0 status=80 max_size=1524 pak_size=0
23 pak=0x603A5B0  ds=0x603A710 status=80 max_size=1524 pak_size=0
24 pak=0x6039E20  ds=0x6039F80 status=80 max_size=1524 pak_size=0
25 pak=0x6039690  ds=0x60397F0 status=80 max_size=1524 pak_size=0
26 pak=0x6038F00  ds=0x6039060 status=80 max_size=1524 pak_size=0
27 pak=0x6038770  ds=0x60388D0 status=80 max_size=1524 pak_size=0
28 pak=0x6037FE0  ds=0x6038140 status=80 max_size=1524 pak_size=0
29 pak=0x6037850  ds=0x60379B0 status=80 max_size=1524 pak_size=0
30 pak=0x60370C0  ds=0x6037220 status=80 max_size=1524 pak_size=0
```

```
31 pak=0x6036930  ds=0x6036A90 status=80 max_size=1524 pak_size=0
TX ring with 8 entries at 0x45DC0 : TLEN=3, TWD=7
tx_count = 0, tx_head = 2, tx_tail = 2
00 pak=0x000000 ds=0x600D786 status=0x38 max_size=1524 pak_size=2
01 pak=0x000000 ds=0x600D98A status=0x38 max_size=1524 pak_size=2
02 pak=0x000000 ds=0x600D786 status=0x38 max_size=1524 pak_size=2
03 pak=0x000000 ds=0x600D786 status=0x38 max_size=1524 pak_size=2
04 pak=0x000000 ds=0x600D786 status=0x38 max_size=1524 pak_size=2
05 pak=0x000000 ds=0x600D786 status=0x38 max_size=1524 pak_size=2
06 pak=0x000000 ds=0x600D786 status=0x38 max_size=1524 pak_size=2
07 pak=0x000000 ds=0x600D786 status=0x38 max_size=1524 pak_size=2
XID/Test TX desc at 0xFFFFFF, status=0x30, max_buffer_size=0, packet_size=0
XID/Test RX desc at 0xFFFFFF, status=0x0, max_buffer_size=0, packet_size=0
Status Buffer at 0x6045FF8: rcv=0, tcv=0, local_state=0, remote_state=0
phase=0, tac=0, currd=0x00000, curxd=0x00000
bad_frames=0, frmrs=0, T1_timeouts=0, rej_rxs=0, runts=0
0 missed datagrams, 0 overruns, 0 bad frame addresses
0 bad datagram encapsulations, 0 user primitive errors
0 provider primitives lost, 0 unexpected provider primitives
0 spurious primitive interrupts, 0 memory errors, 0 transmitter underruns
mk5025 registers: csr0 = 0x0E00, csr1 = 0x0302, csr2 = 0x0704
                  csr3 = 0x5B30, csr4 = 0x0214, csr5 = 0x0008
```

show controllers token

This command displays information about memory management, error counters, and the CSC-R, CSC-1R, CSC-2R, C2CTR, and CSC-R16 (or CSC-R16M) Token Ring interface cards or Token Ring Interface Processor (TRIP), in the case of the Cisco 7000. Depending on the board being used, the output can vary. This command also displays information that is proprietary to Cisco Systems. Thus, the information that **show controllers token** displays is of primary use to Cisco technical personnel. Information that is useful to users can be obtained using the **show interfaces tokenring** command, described later in this chapter.

show controllers token

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output on the AGS+ from the **show controllers token** command:

```
Router# show controllers token
TR Unit 0 is board 0 - ring 0

 state 3, dev blk: 0x1D2EBC, mailbox: 0x2100010, sca: 0x2010000
   current address: 0000.3080.6f40, burned in address: 0000.3080.6f40
   current TX ptr: 0xBA8, current RX ptr: 0x800

   Last Ring Status: none

 Stats: soft:0/0, hard:0/0, sig loss:0/0
        tx beacon: 0/0, wire fault 0/0, recovery: 0/0
        only station: 0/0, remote removal: 0/0
   Bridge: local 3330, bnum 1, target 3583
     max_hops 7, target idb: 0x0, not local
   Interface failures: 0  -- Bkgnd Ints: 0
   TX shorts 0, TX giants 0

   Monitor state: (active)
     flags 0xC0, state 0x0, test 0x0, code 0x0, reason 0x0
 f/w ver: 1.0, chip f/w: '000000.ME31100', [bridge capable]
     SMT versions: 1.01 kernel, 4.02 fastmac
     ring mode: F00, internal enables:  SRB REM RPS CRS/NetMgr
     internal functional: 0000011A (0000011A), group: 00000000 (00000000)
     if_state: 1, ints: 0/0, ghosts: 0/0, bad_states: 0/0
     t2m fifo purges: 0/0
     t2m fifo current: 0, t2m fifo max: 0/0, proto_errs: 0/0
     ring: 3330, bridge num: 1, target: 3583, max hops: 7
Packet counts:
       receive total:  298/6197, small: 298/6197, large 0/0
               runts: 0/0, giants: 0/0
               local: 298/6197, bridged: 0/0, promis: 0/0
             bad rif: 0/0, multiframe: 0/0
       ring num mismatch 0/0, spanning violations 0
       transmit total: 1/25, small: 1/25, large 0/0
                runts: 0/0, giants: 0/0, errors 0/0
```

```
bad fs: 0/0, bad ac: 0
congested: 0/0, not present: 0/0
     Unexpected interrupts: 0/0,  last unexp. int: 0

     Internal controller counts:
   line errors: 0/0,  internal errors: 0/0
   burst errors: 0/0,  ari/fci errors: 0/0
   abort errors: 0/0, lost frame: 0/0
   copy errors: 0/0, rcvr congestion: 0/0
   token errors: 0/0, frequency errors: 0/0
   dma bus errors: -/-, dma parity errors: -/-
     Internal controller smt state:
   Adapter MAC:     0000.3080.6f40, Physical drop:     00000000
   NAUN Address:    0000.a6e0.11a6, NAUN drop:         00000000
   Last source:     0000.a6e0.11a6, Last poll:         0000.3080.6f40
   Last MVID:       0006,           Last attn code:    0006
   Txmit priority:  0006,           Auth Class:        7FFF
   Monitor Error:   0000,           Interface Errors:  FFFF
   Correlator:      0000,           Soft Error Timer:  00C8
   Local Ring:      0000,           Ring Status:       0000
   Beacon rcv type: 0000,           Beacon txmit type: 0000
   Beacon type:     0000,           Beacon NAUN:       0000.a6e0.11a6
```

Table 1-17 describes the fields shown in the following line of sample output:

```
TR Unit 0 is board 0 - ring 0
```

Table 1-17 Show Controllers Token Field Descriptions—Part 1

Field	Description
TR Unit 0	Unit number assigned to the Token Ring interface associated with this output.
is board 0	Board number assigned to the Token Ring controller board associated with this interface.
ring 0	Number of the Token Ring associated with this board.

In the following output line, state 3 indicates the state of the board. The rest of this output line displays memory mapping that is of primary use to Cisco engineers.

```
state 3, dev blk: 0x1D2EBC, mailbox: 0x2100010, sca: 0x2010000
```

The following line also appears in **show interface token** output as the address and burned in address, respectively:

```
current address: 0000.3080.6f40, burned in address: 0000.3080.6f40
```

The following line of output displays buffer management pointers that change by board:

```
current TX ptr: 0xBA8, current RX ptr: 0x800
```

The following line of output indicates the ring status from the controller chip set. This information is used by LAN Network Manager:

```
Last Ring Status: none
```

The following lines of output show Token Ring statistics. See the Token Ring specification for more information.

```
Stats: soft:0/0, hard:0/0, sig loss:0/0
       tx beacon: 0/0, wire fault 0/0, recovery: 0/0
       only station: 0/0, remote removal: 0/0
```

The following line of output indicates that Token Ring communication has been enabled on the interface. If this line of output appears, the message “Source Route Bridge capable” should appear in the **show interfaces tokenring** display.

```
Bridge: local 3330, bnum 1, target 3583
```

Table 1-18 describes the fields shown in this line of sample output:

```
max_hops 7, target idb: 0x0, not local
```

Table 1-18 Show Controllers Token Field Descriptions—Part 2

Field	Description
max_hops 7	Maximum number of bridges.
target idb: 0x0	Destination interface definition.
not local	Indicates whether the interface has been defined as a local or remote bridge.

The following line of output is specific to the hardware:

```
Interface failures: 0  -- Bkgnd Ints: 0
```

In the following line of output, TX shorts are the number of packets the interface transmits that are discarded because they are smaller than the medium’s minimum packet size. TX giants are the number of packets the interface transmits that are discarded because they exceed the medium’s maximum packet size.

```
TX shorts 0, TX giants 0
```

The following line of output indicates the state of the controller. Possible values include active, failure, inactive, and reset:

```
Monitor state: (active)
```

The following line of output displays detailed information relating to the monitor state shown in the previous line of output. This information relates to the firmware on the controller. This information is relevant to Cisco engineers only if the monitor state is something other than active.

```
flags 0xC0, state 0x0, test 0x0, code 0x0, reason 0x0
```

Table 1-19 describes the fields in the following line of output:

```
f/w ver: 1.0 expr 0, chip f/w: '000000.ME31100', [bridge capable]
```

Table 1-19 Show Controllers Token Field Descriptions—Part 3

Field	Description
f/w ver: 1.0	Version of the Cisco firmware on the board.
chip f/w: '000000.ME31100'	Firmware on the chip set.
[bridge capable]	Interface has not been configured for bridging, but that it has that capability.

The following line of output displays the version numbers for the kernel and the accelerator microcode of the Madge firmware on the board; this firmware is the LLC interface to the chip set:

```
SMT versions: 1.01 kernel, 4.02 fastmac
```

The following line of output displays LAN Network Manager information that relates to ring status:

```
ring mode: F00, internal enables:  SRB REM RPS CRS/NetMgr
```

The following line of output corresponds to the functional address and the group address shown in **show interfaces tokenring** output:

```
  internal functional: 0000011A (0000011A), group: 00000000 (00000000)
```

The following line of output displays interface board state information that is proprietary to Cisco Systems:

```
if_state: 1, ints: 0/0, ghosts: 0/0, bad_states: 0/0
```

The following output lines display information that is proprietary to Cisco Systems. Cisco engineers use this information for debugging purposes.

```
t2m fifo purges: 0/0
t2m fifo current: 0, t2m fifo max: 0/0, proto_errs: 0/0
```

Each of the fields in the following line of output maps to a field in the **show source bridge** display, as follows: ring maps to srn; bridge num maps to bn; target maps to trn; and max hops maps to max:

```
      ring: 3330, bridge num: 1, target: 3583, max hops: 7
```

In the following lines of output, the number preceding the slash (/) indicates the count since the value was last displayed; the number following the slash (/) indicates count since the system was last booted:

```
Packet counts:
       receive total:  298/6197, small: 298/6197, large 0/0
```

In the following line of output, the number preceding the slash (/) indicates the count since the value was last displayed; the number following the slash (/) indicates count since the system was last booted. The runts and giants values that appear here correspond to the runts and giants values that appear in **show interfaces tokenring** output.

```
                runts: 0/0, giants: 0/0
```

The following lines of output are receiver-specific information that Cisco engineers can use for debugging purposes:

```
                local: 298/6197, bridged: 0/0, promis: 0/0
              bad rif: 0/0, multiframe: 0/0
        ring num mismatch 0/0, spanning violations 0
        transmit total: 1/25, small: 1/25, large 0/0
                 runts: 0/0, giants: 0/0, errors 0/0
```

The following output lines include very specific statistics that are not relevant in most cases, but exist for historical purposes. In particular, the internal errors, burst errors, ari/fci, abort errors, copy errors, frequency errors, dma bus errors, and dma parity errors fields are not relevant.

```
      Internal controller counts:
line errors: 0/0,  internal errors: 0/0
burst errors: 0/0,  ari/fci errors: 0/0
abort errors: 0/0, lost frame: 0/0
copy errors: 0/0, rcvr congestion: 0/0
token errors: 0/0, frequency errors: 0/0
dma bus errors: -/-, dma parity errors: -/-
```

The following lines of output are low-level Token Ring interface statistics relating to the state and status of the Token Ring with respect to all other Token Rings on the line:

```
  Internal controller smt state:
Adapter MAC:     0000.3080.6f40, Physical drop:     00000000
NAUN Address:    0000.a6e0.11a6, NAUN drop:         00000000
Last source:     0000.a6e0.11a6, Last poll:         0000.3080.6f40
Last MVID:       0006,           Last attn code:    0006
Txmit priority:  0006,           Auth Class:        7FFF
Monitor Error:   0000,           Interface Errors:  FFFF
Correlator:      0000,           Soft Error Timer:  00C8
Local Ring:      0000,           Ring Status:       0000
Beacon rcv type: 0000,           Beacon txmit type: 0000
```

Sample Display

Sample output for the **show controllers token** command on the Cisco 7000 follows:

```
Router> show controllers token
Tokenring4/0: state administratively down
  current address: 0000.3040.8b4a, burned in address: 0000.3040.8b4a
  Last Ring Status: none
    Stats: soft: 0/0, hard: 0/0, sig loss: 0/0
           tx beacon: 0/0, wire fault 0/0, recovery: 0/0
           only station: 0/0, remote removal: 0/0
  Monitor state: (active), chip f/w: '000000........', [bridge capable]
    ring mode: 0”
    internal functional: 00000000 (00000000), group: 00000000 (00000000)
    internal addrs: SRB: 0000, ARB: 0000, EXB 0000, MFB: 0000
                    Rev: 0000, Adapter: 0000, Parms 0000
    Microcode counters:
      MAC giants 0/0, MAC ignored 0/0
      Input runts 0/0, giants 0/0, overrun 0/0
      Input ignored 0/0, parity 0/0, RFED 0/0
      Input REDI 0/0, null rcp 0/0, recovered rcp 0/0
      Input implicit abort 0/0, explicit abort 0/0
      Output underrun 0/0, tx parity 0/0, null tcp 0/0
      Output SFED 0/0, SEDI 0/0, abort 0/0
      Output False Token 0/0, PTT Expired 0/0
    Internal controller counts:
      line errors: 0/0,  internal errors: 0/0
      burst errors: 0/0,  ari/fci errors: 0/0
      abort errors: 0/0, lost frame: 0/0
      copy errors: 0/0, rcvr congestion: 0/0
      token errors: 0/0, frequency errors: 0/0
    Internal controller smt state:
      Adapter MAC:     0000.0000.0000, Physical drop:     00000000
      NAUN Address:    0000.0000.0000, NAUN drop:         00000000
      Last source:     0000.0000.0000, Last poll:         0000.0000.0000
      Last MVID:       0000,           Last attn code:    0000
      Txmit priority:  0000,           Auth Class:        0000
      Monitor Error:   0000,           Interface Errors:  0000
      Correlator:      0000,           Soft Error Timer:  0000
      Local Ring:      0000,           Ring Status:       0000
      Beacon rcv type: 0000,           Beacon txmit type: 0000
      Beacon type:     0000,           Beacon NAUN:       0000.0000.0000
      Beacon drop:     00000000,       Reserved:          0000
      Reserved2:       0000
```

Table 1-20 describes key **show controllers token** display fields.

Table 1-20 Show Controllers Token Field Descriptions

Field	Description
Tokenring4/0	Interface processor type, slot, and port.
Last Ring Status	Last abnormal ring condition. Can be any of the following: Signal Loss HW Removal Remote Removal Counter Overflow Only station Ring Recovery

show interfaces

Use the **show interfaces** EXEC command to display statistics for all interfaces configured on the router. The resulting output varies, depending on the network for which an interface has been configured.

show interfaces [*type* {*unit*}] [*first*] [*last*] [**accounting**]
show interfaces [*type slot/port*] [**accounting**] (for the Cisco 7000)

Syntax Description

type unit	(Optional.) Specify that information for a particular interface controller be displayed. Allowed values for type include **async**, **bri0**, **ethernet**, **fddi**, **hssi**, **loopback**, **null**, **serial**, **tokenring**, and **tunnel**. For the Cisco 7000, *type* can be **ethernet**, **fddi**, **serial**, or **tokenring**. The argument *unit* must match a port number on the selected interface controller.
first last	(Optional.) The Cisco 3000 supports the ISDN Basic Rate Interface (BRI). The argument *first* can be either 1 or 2. The argument *last* can only be 2, indicating B channels 1 and 2. D-channel information is obtained by using the command without the optional arguments.
accounting	(Optional.) Displays the number of packets of each protocol type that has been sent through the interface. You can show these numbers for all interfaces, or you can specify a specific *type* and *unit*.
slot	Specifies the backplane slot number and can be 0, 1, 2, 3, or 4.
port	Specifies the port number of the interface and can be 0, 1, 2, 3, 4, or 5 depending on the type of interface, as follows: • EIP (Ethernet Interface Processor) 0, 1, 2, 3, 4, or 5 • FIP (FDDI Interface Processor) 0 • FSIP (Fast Serial Interface Processor) 0, 1, 2, or 3 • HIP (HSSI Interface Processor) 0 • TRIP (Token Ring Interface Processor) 0, 1, 2, or 3

Command Mode

EXEC

Usage Guidelines

The **show interfaces** command displays statistics for the network interfaces. The resulting display on a Cisco 7000 will show the interface processors in slot order. If you add interface processors after booting the system, they will appear at the end of the list, in the order in which they were inserted.

If you use the **show interfaces** command in a Cisco 7000 without the *slot/port* arguments, information for all interface types will be shown. For example, if you type **show interfaces ethernet** you will receive information for all ethernet, serial, Token Ring, and FDDI interfaces. Only by adding the *type slot/port* argument can you specify a particular interface.

If you enter a **show interfaces** command for an interface type that has been removed from the router, interface statistics will be displayed accompanied by the following text: "Hardware has been removed."

You will use the **show interfaces** command frequently while configuring and monitoring routers. The various forms of the **show interfaces** commands are described in detail in the sections immediately following this command.

Sample Display

The following is sample output from the **show interfaces** command. Because your display will depend on the type and number of interface cards in your router, only a portion of the display is shown.

```
Router# show interfaces
Ethernet 0 is up, line protocol is up
  Hardware is MCI Ethernet, address is 0000.0c00.750c (bia 0000.0c00.750c)
  Internet address is 131.108.28.8, subnet mask is 255.255.255.0
  MTU 1500 bytes, BW 10000 Kbit, DLY 100000 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set, keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 4:00:00
  Last input 0:00:00, output 0:00:00, output hang never
  Last clearing of "show interface" counters 0:00:00
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  Five minute input rate 0 bits/sec, 0 packets/sec
  Five minute output rate 2000 bits/sec, 4 packets/sec
     1127576 packets input, 447251251 bytes, 0 no buffer
     Received 354125 broadcasts, 0 runts, 0 giants
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     5332142 packets output, 496316039 bytes, 0 underruns
     0 output errors, 432 collisions, 0 interface resets, 0 restarts
---More---
```

Sample Display with Accounting Option

To display the number of packets of each protocol type that have been sent through all configured interfaces, use the **show interfaces accounting** EXEC command. When you use the **accounting** option, only the accounting statistics are displayed.

Note Except for protocols that are encapsulated inside other protocols, such as IP over X.25, the accounting option also shows the total of all bytes sent and received, including the MAC header. For example, it totals the size of the Ethernet packet or the size of a packet that includes HDLC encapsulation.

Table 1-21 lists the protocols for which per-packet accounting information is kept.

Table 1-21 Per-Packet Counted Protocols

Protocol	Notes
Apollo	no note
AppleTalk	no note
ARP	For IP, Apollo, Frame Relay, SMDS.
CLNS	no note

Protocol	Notes
DEC MOP	The routers use MOP packets to advertise their existence to DEC machines that use the MOP protocol. A router periodically broadcasts MOP packets to identify itself as a MOP host. This results in MOP packets being counted, even when DECnet is not being actively used.
DECnet	no note
HP Probe	no note
IP	no note
Lan Manager	LAN Network Manager and IBM Network Manager.
Novell	no note
Serial Tunnel	SDLC.
Spanning Tree	no note
SR Bridge	no note
Transparent Bridge	no note
VINES	no note
XNS	no note

Sample Display

The following is sample output from the **show interfaces accounting** command:

```
Router# show interfaces accounting
Interface TokenRing0 is disabled

Ethernet0
               Protocol    Pkts In   Chars In   Pkts Out  Chars Out
                     IP     873171  735923409      34624    9644258
                 Novell     163849   12361626      57143    4272468
                DEC MOP          0          0          1         77
                    ARP      69618    4177080       1529      91740
Interface Serial0 is disabled

Ethernet1
               Protocol    Pkts In   Chars In   Pkts Out  Chars Out
                     IP          0          0         37      11845
                 Novell          0          0       4591     275460
                DEC MOP          0          0          1         77
                    ARP          0          0          7        420

Interface Serial1 is disabled
Interface Ethernet2 is disabled
Interface Serial2 is disabled
Interface Ethernet3 is disabled
Interface Serial3 is disabled
Interface Ethernet4 is disabled
Interface Ethernet5 is disabled
Interface Ethernet6 is disabled
Interface Ethernet7 is disabled
Interface Ethernet8 is disabled
Interface Ethernet9 is disabled

Fddi0
```

```
Protocol    Pkts In   Chars In   Pkts Out  Chars Out
  Novell          0          0        183      11163
     ARP          1         49          0          0
```

When the output indicates an interface is “disabled,” the router has received excessive errors (over 5000 in a keepalive period).

show interfaces async

Use the **show interfaces async** EXEC command to display information about the serial interface.

show interfaces async [*unit*] [**accounting**]

Syntax Description

unit	(Optional.) Must be 1.
accounting	(Optional.) Displays the number of packets of each protocol type that have been sent through the interface.

Command Mode

EXEC

Sample Display

The following is sample output from the **show interfaces async** command:

```
Router# show interfaces async 1

Async 1 is up, line protocol is up
   Hardware is Async Serial
Internet address is 1.0.0.1, subnet mask is 255.0.0.0
MTU 1500 bytes, BW 9 Kbit, DLY 100000 usec, rely 255/255, load 56/255
Encapsulation SLIP, keepalive set (0 sec)
Last input 0:00:03, output 0:00:03, output hang never
Last clearing of “show interface” counters never
Output queue 0/3, 2 drops; input queue 0/0, 0 drops
Five minute input rate 0 bits/sec, 1 packets/sec
Five minute output rate 2000 bits/sec, 1 packets/sec
273 packets input, 13925 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
221 packets output, 41376 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets, 0 restarts
0 carrier transitions
```

Table 1-22 describes the fields shown in the display.

Table 1-22 Show Interfaces Async Field Descriptions

Field	Description
Async... is {up | down} ...is administratively down	Indicates whether the interface hardware is currently active (whether carrier detect is present) and if it has been taken down by an administrator.
line protocol is {up | down | administratively down}	Indicates whether the software processes that handle the line protocol think the line is usable (that is, whether keepalives are successful).
Hardware is	Hardware type.
Internet address is	Internet address and subnet mask, followed by packet size.
MTU	Maximum Transmission Unit of the interface.
BW	Bandwidth of the interface in kilobits per second.
DLY	Delay of the interface in microseconds.
rely	Reliability of the interface as a fraction of 255 (255/255 is 100% reliability), calculated as an exponential average over 5 minutes.
load	Load on the interface as a fraction of 255 (255/255 is completely saturated), calculated as an exponential average over 5 minutes. The calculation uses the value from the **bandwidth** interface configuration command.
Encapsulation	Encapsulation method assigned to interface.
keepalive	Indicates whether keepalives are set or not.
Last input	Number of hours, minutes, and seconds since the last packet was successfully received by an interface. Useful for knowing when a dead interface failed.
Last output	Number of hours, minutes, and seconds since the last packet was successfully transmitted by an interface.
output hang	Number of hours, minutes, and seconds (or never) since the interface was last reset because of a transmission that took too long. When the number of hours in any of the "last" fields exceeds 24 hours, the number of days and hours is printed. If that field overflows, asterisks are printed.
Last clearing	The time at which the counters that measure cumulative statistics (such as number of bytes transmitted and received) shown in this report were last reset to zero. Note that variables that might affect routing (for example, load and reliability) are not cleared when the counters are cleared. *** indicates the elapsed time is too large to be displayed. 0:00:00 indicates the counters were cleared more than 2^{31}ms (and less than 2^{32}ms) ago.
Output queue, drops input queue, drops	Number of packets in output and input queues. Each number is followed by a slash, the maximum size of the queue, and the number of packets dropped due to a full queue.
Five minute input rate, Five minute output rate	Average number of bits and packets transmitted per second in the last 5 minutes.
packets input	Total number of error-free packets received by the system.
bytes input	Total number of bytes, including data and MAC encapsulation, in the error free packets received by the system.

Field	Description
no buffers	Number of received packets discarded because there was no buffer space in the main system. Compare with ignored count. Broadcast storms on Ethernets and bursts of noise on serial lines are often responsible for no input buffer events.
broadcasts	Total number of broadcast or multicast packets received by the interface.
runts	Number of packets that are discarded because they are smaller than the medium's minimum packet size.
giants	Number of packets that are discarded because they exceed the medium's maximum packet size.
input errors	Total number of no buffer, runts, giants, CRCs, frame, overrun, ignored, and abort counts. Other input-related errors can also increment the count, so that this sum may not balance with the other counts.
CRC	Cyclic redundancy checksum generated by the originating LAN station or far end device does not match the checksum calculated from the data received. On a LAN, this usually indicates noise or transmission problems on the LAN interface or the LAN bus itself. A high number of CRC's is usually the result of collisions or a station transmitting bad data. On a serial link, CRC's usually indicate noise, gain hits or other transmission problems on the data link.
frame	Number of packets received incorrectly having a CRC error and a noninteger number of octets. On a serial line, this is usually the result of noise or other transmission problems.
overrun	Number of times the serial receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver's ability to handle the data.
ignored	Number of received packets ignored by the interface because the interface hardware ran low on internal buffers. These buffers are different than the system buffers mentioned previously in the buffer description. Broadcast storms and bursts of noise can cause the ignored count to be incremented.
abort	Illegal sequence of one bits on a serial interface. This usually indicates a clocking problem between the serial interface and the data link equipment.
input packets with dribble condition detected	Dribble bit condition indicates Carrier Sense did not go inactive on a receive data byte boundary.
packets output	Total number of messages transmitted by the system.
bytes	Total number of bytes, including data and MAC encapsulation, transmitted by the system.
output errors	Sum of all errors that prevented the final transmission of datagrams out of the interface being examined. Note that this may not balance with the sum of the enumerated output errors, as some datagrams may have more than one error, and others may have errors that do not fall into any of the specifically tabulated categories.

Field	Description
interface resets	Number of times an interface has been completely reset. This can happen if packets queued for transmission were not sent within several seconds. On a serial line, this can be caused by a malfunctioning modem that is not supplying the transmit clock signal, or by a cable problem. If the system notices that the carrier detect line of a serial interface is up, but the line protocol is down, it periodically resets the interface in an effort to restart it. Interface resets can also occur when an interface is looped back or shut down.
restarts	Number of times the controller was restarted because of errors.
carrier transitions	Number of times the carrier detect signal of a serial interface has changed state. Indicates modem or line problems if the carrier detect line is changing state often.
Protocol	Protocol that is operating on the interface.
Pkts In	Number of packets received for that protocol.
Chars In	Number of characters received for that protocol.
Pkts Out	Number of packets transmitted for that protocol.
Chars Out	Number of characters transmitted for that protocol.

Sample Display with Accounting Option

The following is a sample display from the **show interfaces async accounting** command:

```
Router# show interfaces async 0 accounting

Async 0
   Protocol  Pkts In   Chars In  Pkts Out   Chars Out
   IP        7344      4787842   1803       1535774
   DEC MOP   0         0         127        9779
   ARP       7         420       39         2340
```

The **show line** and **show slip** commands can also be useful in monitoring asynchronous interfaces.

show interfaces bri

Use the **show interfaces bri** EXEC command to display information about the BRI D and B channels.

show interfaces bri [*first*] [*last*] [**accounting**]

Syntax Description

first last	(Optional.) The argument *first* can be either 1 or 2. The argument *last* can only be 2, indicating B channels 1 and 2. D-channel information is obtained by using the command without the optional arguments.
accounting	(Optional.) Displays the number of packets of each protocol type that have been sent through the interface.

Command Mode

EXEC

Sample Display

The following is sample output from the **show interfaces** command for BRI interfaces:

```
Router# show interfaces bri 0

BRI0 is up, line protocol is up (spoofing)
   Hardware is BRI
   Internet address is 150.136.190.203, subnet mask is 255.255.255.0
   MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec, rely 255/255, load 1/255
   Encapsulation HDLC, loopback not set, keepalive set (10 sec)
   Last input 0:00:07, output 0:00:00, output hang never
   Output queue 0/40, 0 drops; input queue 0/75, 0 drops
   Five minute input rate 0 bits/sec, 0 packets/sec
   Five minute output rate 0 bits/sec, 0 packets/sec
       16263 packets input, 1347238 bytes, 0 no buffer
       Received 13983 broadcasts, 0 runts, 0 giants
       2 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 2 abort
       22146 packets output, 2383680 bytes, 0 underruns
       0 output errors, 0 collisions, 2 interface resets, 0 restarts
       1 carrier transitions
```

Table 1-23 describes the fields shown in the display.

Table 1-23 Show Interfaces ISDN BRI Field Descriptions

Field	Description
BRI ... is {up | down} ...is administratively down	Indicates whether the interface hardware is currently active (whether line signal is present) and if it has been taken down by an administrator.
line protocol is {up | down | administratively down}	Indicates whether the software processes that handle the line protocol consider the line usable (that is, whether keepalives are successful).
Hardware is	Hardware type.
Internet address is	Internet address and subnet mask, followed by packet size.

Field	Description
MTU	Maximum Transmission Unit of the interface.
BW	Bandwidth of the interface in kilobits per second.
DLY	Delay of the interface in microseconds.
rely	Reliability of the interface as a fraction of 255 (255/255 is 100% reliability), calculated as an exponential average over 5 minutes.
load	Load on the interface as a fraction of 255 (255/255 is completely saturated), calculated as an exponential average over 5 minutes.
Encapsulation	Encapsulation method assigned to interface.
loopback	Indicates whether loopback is set or not.
keepalive	Indicates whether keepalives are set or not.
Last input	Number of hours, minutes, and seconds since the last packet was successfully received by an interface. Useful for knowing when a dead interface failed.
Last output	Number of hours, minutes, and seconds since the last packet was successfully transmitted by an interface.
output hang	Number of hours, minutes, and seconds (or never) since the interface was last reset because of a transmission that took too long. When the number of hours in any of the "last" fields exceeds 24 hours, the number of days and hours is printed. If that field overflows, asterisks are printed.
Output queue, drops Input queue, drops	Number of packets in output and input queues. Each number is followed by a slash, the maximum size of the queue, and the number of packets dropped due to a full queue.
Five minute input rate Five minute output rate	Average number of bits and packets transmitted per second in the last 5 minutes.
packets input	Total number of error-free packets received by the system.
bytes input	Total number of bytes, including data and MAC encapsulation, in the error free packets received by the system.
no buffers	Number of received packets discarded because there was no buffer space in the main system. Compare with ignored count. Broadcast storms on Ethernets and bursts of noise on serial lines are often responsible for no input buffer events.
broadcasts	Total number of broadcast or multicast packets received by the interface.
runts	Number of packets that are discarded because they are smaller than the medium's minimum packet size.
giants	Number of packets that are discarded because they exceed the medium's maximum packet size.
input errors	Total number of no buffer, runts, giants, CRCs, frame, overrun, ignored, and abort counts. Other input-related errors also can increment the count, so this sum may not balance with the other counts.
CRC	Cyclic redundancy checksum generated by the originating station or far-end device does not match the checksum calculated from the data received. On a serial link, CRCs usually indicate noise, gain hits, or other transmission problems on the data link.

Field	Description
frame	Number of packets received incorrectly having a CRC error and a noninteger number of octets. On a serial line, this is usually the result of noise or other transmission problems.
overrun	Number of times the serial receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver's ability to handle the data.
ignored	Number of received packets ignored by the interface because the interface hardware ran low on internal buffers. Broadcast storms and bursts of noise can cause the ignored count to be increased.
abort	Illegal sequence of one bits on a serial interface. This usually indicates a clocking problem between the serial interface and the data link equipment.
input packets with dribble condition detected	Dribble bit condition indicates Carrier Sense did not go inactive on a receive data byte boundary.
packets output	Total number of messages transmitted by the system.
bytes	Total number of bytes, including data and MAC encapsulation, transmitted by the system.
underruns	Number of times that the transmitter has been running faster than the router can handle. This may never be reported on some interfaces.
output errors	Sum of all errors that prevented the final transmission of datagrams out of the interface being examined. Note that this may not balance with the sum of the enumerated output errors, as some datagrams may have more than one error, and others may have errors that do not fall into any of the specifically tabulated categories.
interface resets	Number of times an interface has been completely reset. This can happen if packets queued for transmission were not sent within several seconds. On a serial line, this can be caused by a malfunctioning modem that is not supplying the transmit clock signal or by a cable problem. If the system notices that the carrier detect line of a serial interface is up, but the line protocol is down, it periodically resets the interface in an effort to restart it. Interface resets can also occur when an interface is looped back or shut down.
restarts	Number of times the controller was restarted because of errors.
carrier transitions	Number of times the carrier detect signal of a serial interface has changed state. Indicates modem or line problems if the carrier detect line is changing state often.
Protocol	Protocol that is operating on the interface.
Pkts In	Number of packets received for that protocol.
Chars In	Number of characters received for that protocol.
Pkts Out	Number of packets transmitted for that protocol.
Chars Out	Number of characters transmitted for that protocol.

show interfaces brief

To list a brief summary of an interface IP information and status, use the **show interfaces brief** EXEC command.

show interfaces brief

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show interfaces brief** command:

```
Router# show interfaces brief
Any interface listed with OK? value "NO" does not have a valid configuration
Interface    IP-Address      OK?  Method     Status                 Protocol
TokenRing0   unassigned      YES  not set    administratively down  down
Ethernet0    131.108.160.22  YES  NVRAM      up                     up
Serial0      unassigned      YES  not set    administratively down  down
Ethernet1    unassigned      YES  not set    up                     up
Serial1      unassigned      YES  not set    administratively down  down
Ethernet2    unassigned      YES  not set    administratively down  down
Serial2      unassigned      YES  not set    administratively down  down
Ethernet3    unassigned      YES  not set    administratively down  down
Serial3      unassigned      YES  not set    administratively down  down
Ethernet4    unassigned      YES  not set    administratively down  down
Ethernet5    unassigned      YES  not set    administratively down  down
Ethernet6    unassigned      YES  not set    administratively down  down
Ethernet7    unassigned      YES  not set    administratively down  down
Ethernet8    unassigned      YES  not set    administratively down  down
Ethernet9    unassigned      YES  not set    administratively down  down
Fddi0        unassigned      YES  not set    up                     up
Tunnel12     unassigned      YES  not set    up                     down
Tunnel4      unassigned      YES  not set    up                     down
```

Related Command

show interfaces

show interfaces ethernet

Use the **show interfaces ethernet** EXEC command to display information about an Ethernet interface on the router.

show interfaces ethernet *unit* [**accounting**]
show interfaces ethernet *slot/port* [**accounting**] (on a Cisco 7000)

Syntax Description

unit	Must match a port number on the selected interface.
accounting	(Optional.) Displays the number of packets of each protocol type that have been sent through the interface.
slot	On a Cisco 7000, optional slot location of the interface processor.
port	On a Cisco 7000, optional port number on interface.

Command Mode

EXEC

Usage Guidelines

If you do not provide values for the argument *unit* (or *slot* and *port* on a Cisco 7000), the command will display statistics for all network interfaces. The optional keyword **accounting** displays the number of packets of each protocol type that have been sent through the interface.

Sample Display

The following is sample output from the **show interfaces** command for the Ethernet 0 interface:

```
Router# show interfaces ethernet 0

Ethernet 0 is up, line protocol is up
   Hardware is MCI Ethernet, address is aa00.0400.0134 (bia 0000.0c00.4369)
   Internet address is 131.108.1.1, subnet mask is 255.255.255.0
   MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load 1/255
   Encapsulation ARPA, loopback not set, keepalive set (10 sec)
   ARP type: ARPA, PROBE, ARP Timeout 4:00:00
   Last input 0:00:00, output 0:00:00, output hang never
   Output queue 0/40, 0 drops; input queue 0/75, 2 drops
   Five minute input rate 61000 bits/sec, 4 packets/sec
   Five minute output rate 1000 bits/sec, 2 packets/sec
       2295197 packets input, 305539992 bytes, 0 no buffer
       Received 1925500 broadcasts, 0 runts, 0 giants
       3 input errors, 3 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
       0 input packets with dribble condition detected
       3594664 packets output, 436549843 bytes, 0 underruns
       8 output errors, 1790 collisions, 10 interface resets, 0 restarts
```

Table 1-24 describes significant fields shown in the display.

Table 1-24 Show Interfaces Ethernet Field Descriptions

Field	Description
Ethernet ... is up ...is administratively down	Indicates whether the interface hardware is currently active and if it has been taken down by an administrator. “Disabled” indicates the router has received over 5000 errors in a keepalive interval, which is 10 seconds by default.
line protocol is {up | down | administratively down}	Indicates whether the software processes that handle the line protocol believe the interface is usable (that is, whether keepalives are successful) or if it has been taken down by an administrator.
Hardware	Hardware type (for example, MCI Ethernet, SCI, cBus Ethernet) and address.
Internet address	Internet address followed by subnet mask.
MTU	Maximum Transmission Unit of the interface.
BW	Bandwidth of the interface in kilobits per second.
DLY	Delay of the interface in microseconds.
rely	Reliability of the interface as a fraction of 255 (255/255 is 100% reliability), calculated as an exponential average over 5 minutes.
load	Load on the interface as a fraction of 255 (255/255 is completely saturated), calculated as an exponential average over 5 minutes.
Encapsulation	Encapsulation method assigned to interface.
ARP type:	Type of Address Resolution Protocol assigned.
loopback	Indicates whether loopback is set or not.
keepalive	Indicates whether keepalives are set or not.
Last input	Number of hours, minutes, and seconds since the last packet was successfully received by an interface. Useful for knowing when a dead interface failed.
Last output	Number of hours, minutes, and seconds since the last packet was successfully transmitted by an interface.
output	Number of hours, minutes, and seconds since the last packet was successfully transmitted by the interface. Useful for knowing when a dead interface failed.
output hang	Number of hours, minutes, and seconds (or never) since the interface was last reset because of a transmission that took too long. When the number of hours in any of the “last” fields exceeds 24 hours, the number of days and hours is printed. If that field overflows, asterisks are printed.
Last clearing	Time at which the counters that measure cumulative statistics (such as number of bytes transmitted and received) shown in this report were last reset to zero. Note that variables that might affect routing (for example, load and reliability) are not cleared when the counters are cleared. *** indicates the elapsed time is too large to be displayed. 0:00:00 indicates the counters were cleared more than 2^{31}ms (and less than 2^{32}ms) ago.
Output queue, input queue, drops	Number of packets in output and input queues. Each number is followed by a slash, the maximum size of the queue, and the number of packets dropped due to a full queue.

Field	Description
Five minute input rate, Five minute output rate	Average number of bits and packets transmitted per second in the last 5 minutes. If the interface is not in promiscuous mode, it senses network traffic it sends and receives (rather than all network traffic). The 5-minute input and output rates should be used only as an approximation of traffic per second during a given 5-minute period. These rates are exponentially weighted averages with a time constant of 5 minutes. A period of four time constants must pass before the average will be within two percent of the instantaneous rate of a uniform stream of traffic over that period.
packets input	Total number of error-free packets received by the system.
bytes input	Total number of bytes, including data and MAC encapsulation, in the error free packets received by the system.
no buffers	Number of received packets discarded because there was no buffer space in the main system. Compare with ignored count. Broadcast storms on Ethernets and bursts of noise on serial lines are often responsible for no input buffer events.
Received ... broadcasts	Total number of broadcast or multicast packets received by the interface.
runts	Number of packets that are discarded because they are smaller than the medium's minimum packet size. For instance, any Ethernet packet that is less than 64 bytes is considered a runt.
giants	Number of packets that are discarded because they exceed the medium's maximum packet size. For example, any Ethernet packet that is greater than 1,518 bytes is considered a giant.
input error	Includes runts, giants, no buffer, CRC, frame, overrun, and ignored counts. Other input-related errors can also cause the input errors count to be increased, and some datagrams may have more than one error; therefore, this sum may not balance with the sum of enumerated input error counts.
CRC	Cyclic redundancy checksum generated by the originating LAN station or far-end device does not match the checksum calculated from the data received. On a LAN, this usually indicates noise or transmission problems on the LAN interface or the LAN bus itself. A high number of CRCs is usually the result of collisions or a station transmitting bad data.
frame	Number of packets received incorrectly having a CRC error and a noninteger number of octets. On a LAN, this is usually the result of collisions or a malfunctioning Ethernet device.
overrun	Number of times the receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver's ability to handle the data.
ignored	Number of received packets ignored by the interface because the interface hardware ran low on internal buffers. These buffers are different than the system buffers mentioned previously in the buffer description. Broadcast storms and bursts of noise can cause the ignored count to be increased.
dribble condition detected	Dribble bit condition indicates that carrier sense did not go inactive on a receive data byte boundary.
packets output	Total number of messages transmitted by the system.

Field	Description
bytes	Total number of bytes, including data and MAC encapsulation, transmitted by the system.
underruns	Number of times that the transmitter has been running faster than the router can handle. This may never be reported on some interfaces.
output errors	Sum of all errors that prevented the final transmission of datagrams out of the interface being examined. Note that this may not balance with the sum of the enumerated output errors, as some datagrams may have more than one error, and others may have errors that do not fall into any of the specifically tabulated categories.
collisions	Number of messages retransmitted due to an Ethernet collision. This is usually the result of an overextended LAN (Ethernet or transceiver cable too long, more than two repeaters between stations, or too many cascaded multiport transceivers). A packet that collides is counted only once in output packets.
interface resets	Number of times an interface has been completely reset. This can happen if packets queued for transmission were not sent within several seconds. On a serial line, this can be caused by a malfunctioning modem that is not supplying the transmit clock signal, or by a cable problem. If the system notices that the carrier detect line of a serial interface is up, but the line protocol is down, it periodically resets the interface in an effort to restart it. Interface resets can also occur when an interface is looped back or shut down.
restarts	Number of times a Type 2 Ethernet controller was restarted because of errors.

Sample Display on Cisco 7000

The following sample output illustrates the **show interfaces ethernet** command on the Cisco 7000:

```
> show interfaces ethernet 4/2
Ethernet4/2 is up, line protocol is up
  Hardware is cxBus Ethernet, address is 0000.0c02.d0ce (bia 0000.0c02.d0ce)
  Internet address is 131.108.7.1, subnet mask is 255.255.255.0
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set, keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 4:00:00
  Last input 0:00:00, output 0:00:09, output hang never
  Last clearing of "show interface" counters 0:56:40
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  Five minute input rate 3000 bits/sec, 4 packets/sec
  Five minute output rate 0 bits/sec, 0 packets/sec
     4961 packets input, 715381 bytes, 0 no buffer
     Received 2014 broadcasts, 0 runts, 0 giants
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     567 packets output, 224914 bytes, 0 underruns
     0 output errors, 168 collisions, 0 interface resets, 0 restarts
```

Sample Display with Accounting Option

The following is sample output from the **show interfaces ethernet** command with the **accounting** option on the Cisco 7000:

```
Router# show interfaces ethernet 4/2 accounting
Ethernet4/2
       Protocol    Pkts In   Chars In   Pkts Out  Chars Out
             IP       7344    4787842       1803    1535774
      Appletalk      33345    4797459      12781    1089695
        DEC MOP          0          0        127       9779
            ARP          7        420         39       2340
```

show interfaces fddi

Use the **show interfaces fddi** EXEC command to display information about the FDDI interface.

show interfaces fddi *unit* [**accounting**]
show interfaces fddi *slot/port* [**accounting**] (for the Cisco 7000)

Syntax Description

unit	Must match a port number on the selected interface.
accounting	(Optional.) Displays the number of packets of each protocol type that have been sent through the interface.
slot	On a Cisco 7000, optional slot location of the interface processor.
port	On a Cisco 7000, optional port number on interface.

Command Mode

EXEC

Sample Display

The following is a sample partial display of FDDI-specific data from the **show interfaces fddi** command on a Cisco 7000:

```
> show interfaces fddi 3/0
Fddi3/0 is up, line protocol is up
  Hardware is cxBus Fddi, address is 0000.0c02.adf1 (bia 0000.0c02.adf1)
  Internet address is 131.108.33.14, subnet mask is 255.255.255.0
  MTU 4470 bytes, BW 100000 Kbit, DLY 100 usec, rely 255/255, load 1/255
  Encapsulation SNAP, loopback not set, keepalive not set
  ARP type: SNAP, ARP Timeout 4:00:00
  Phy-A state is  active, neighbor is   B, cmt signal bits 008/20C, status ILS
  Phy-B state is  active, neighbor is   A, cmt signal bits 20C/008, status ILS
  CFM is thru A, token rotation 5000 usec, ring operational 21:32:34
  Upstream neighbor 0000.0c02.ba83, downstream neighbor 0000.0c02.ba83
  Last input 0:00:05, output 0:00:00, output hang never
  Last clearing of "show interface" counters 0:59:10
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  Five minute input rate 69000 bits/sec, 44 packets/sec
  Five minute output rate 0 bits/sec, 1 packets/sec
     113157 packets input, 21622582 bytes, 0 no buffer
     Received 276 broadcasts, 0 runts, 0 giants
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     4740 packets output, 487346 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets, 0 restarts
     0 transitions, 2 traces, 3 claims, 2 beacons
```

The following is an example that includes the **accounting** option. When you use the **accounting** option, only the accounting statistics are displayed.

```
> show interfaces fddi 3/0 accounting
Fddi3/0
       Protocol    Pkts In   Chars In   Pkts Out  Chars Out
             IP       7344    4787842       1803    1535774
      Appletalk      33345    4797459      12781    1089695
        DEC MOP          0          0        127       9779
            ARP          7        420         39       2340
```

Table 1-25 describes the **show interfaces fddi** display fields.

Table 1-25 Show Interfaces FDDI Field Descriptions

Field	Description
Fddi is {up |down} ...is administratively down	Gives the interface processor unit number and tells whether the interface hardware is currently active and can transmit and receive or if it has been taken down by an administrator. “Disabled” indicates the router has received over 5000 errors in a keepalive interval, which is 10 seconds by default.
line protocol is {up | down | administratively down}	Indicates whether the interface hardware is currently active and can transmit and receive or if it has been taken down by an administrator.
Hardware	Provides the hardware type, followed by the hardware address.
Internet address	Internet address, followed by subnet mask.
MTU	Maximum Transmission Unit of the interface.
BW	Bandwidth of the interface in kilobits per second.
DLY	Delay of the interface in microseconds.
rely	Reliability of the interface as a fraction of 255 (255/255 is 100% reliability), calculated as an exponential average over 5 minutes.
load	Load on the interface as a fraction of 255 (255/255 is completely saturated), calculated as an exponential average over 5 minutes.
Encapsulation	Encapsulation method assigned to interface.
loopback	Indicates whether or not loopback is set.
keepalive	Indicates whether or not keepalives are set.
ARP type:	Type of Address Resolution Protocol assigned.
Phy-{A | B}	Lists the state the Physical A or Physical B connection is in; one of: off, active, trace, connect, next, signal, join, verify, or break.

neighbor	State of the neighbor: • A—Indicates that the CMT process has established a connection with its neighbor. The bits received during the CMT signaling process indicate that the neighbor is a Physical A type dual-attachment station or concentrator that attaches to the primary ring IN and the secondary ring OUT when attaching to the dual ring. • S—Indicates that the CMT process has established a connection with its neighbor and that the bits received during the CMT signaling process indicate that the neighbor is one Physical type in a single-attached station (SAS). • B—Indicates that the CMT process has established a connection with its neighbor and that the bits received during the CMT signaling process indicate that the neighbor is a Physical B dual-attached station or concentrator that attaches to the secondary ring IN and the primary ring OUT when attaching to the dual ring. • M—Indicates that the CMT process has established a connection with its neighbor and that the bits received during the CMT signaling process indicate that the router's neighbor is a Physical M-type concentrator that serves as a Master to a connected station or concentrator. • unk—Indicates that the network server has not completed the CMT process, and as a result, does not know about its neighbor. See the section "Setting Bit Control" for an explanation of the bit patterns.
cmt signal bits	Shows the transmitted/received CMT bits. The transmitted bits are 0x008 for a Physical A type and 0x20C for Physical B type. The number after the slash (/) is the received signal bits. If the connection is not active, the received bits are zero (0); see the line beginning Phy-B earlier in this display.
status	Status value displayed is the actual status on the fiber. The FDDI standard defines the following values: • LSU—Line State Unknown, the criteria for entering or remaining in any other line state have not been met. • NLS—Noise Line State is entered upon the occurrence of 16 potential noise events without satisfying the criteria for entry into another line state. • MLS—Master Line State is entered upon the reception of eight or nine consecutive HQ or QH symbol pairs. • ILS—Idle Line State is entered upon receipt of four or five idle symbols. • HLS—Halt Line State is entered upon the receipt of 16 or 17 consecutive H symbols. • QLS—Quiet Line State is entered upon the receipt of 16 or 17 consecutive Q symbols or when carrier detect goes low. • ALS—Active Line State is entered upon receipt of a JK symbol pair when carrier detect is high. • OVUF—Elasticity buffer Overflow/Underflow. The normal states for a connected Physical type are ILS or ALS. If the report displays the QLS status, this indicates that the fiber is disconnected from Physical B, or that it is not connected to another Physical type, or that the other station is not running.
Off	Indicates that the CMT is not running on the Physical Sublayer. The state will be off if the interface has been shutdown or if the **cmt disconnect** command has been issued for Physical A or Physical B.
Brk	Break State is the entry point in the start of a PCM connection.
Tra	Trace State localizes a stuck beacon condition.

Con	Connect State is used to synchronize the ends of the connection for the signaling sequence.
Nxt	Next State separates the signaling performed in the Signal State and transmits Protocol Data Units (PDUs) while MAC Local Loop is performed.
Sig	Signal State is entered from the Next State when a bit is ready to be transmitted.
Join	Join State is the first of three states in a unique sequence of transmitted symbol streams received as line states—the Halt Line State, Master Line State, and Idle Line State, or HLS-MLS-ILS—that leads to an active connection.
Vfy	Verify State is the second state in the path to the Active State and will not be reached by a connection that is not synchronized.
Act	Active State indicates that the CMT process has established communications with its physical neighbor. The transition states are defined in the X3T9.5 specification. You are referred to the specification for details about these states.
CFM is ...	Contains information about the current state of the MAC connection. The Configuration Management (CFM) state can be one of the following: • Isolated—The MAC is not attached to any Physical type. • Wrap A—The MAC is attached to Physical A. Data is received on Physical A and transmitted on Physical A. • Wrap B—The MAC is attached to Physical B. Data is received on Physical B and transmitted on Physical B. • Thru A—The MAC is attached to Physical A and B. Data is received on Physical A and transmitted on Physical B. This is the normal mode for a dual attachment station (DAS) with one MAC. The ring has been operational for 1 minute and 42 seconds.
token rotation	Token rotation value is the default or configured rotation value as determined by the fddi token rotation-time command. This value is used by all stations on the ring. The default is 5000 microseconds.
ring operational	When the ring is operational, the displayed value will be the negotiated token rotation time of all stations on the ring. Operational times are displayed by the number of hours:minutes:seconds the ring has been up. If the ring is not operational, the message “ring not operational” is displayed.
Upstream | downstream neighbor	Displays the canonical MAC address of outgoing upstream and downstream neighbors. If the interface is not up, these values will be zero (0).
Last input	Number of hours, minutes, and seconds since the last packet was successfully received by an interface. Useful for knowing when a dead interface failed.
Last output	Number of hours, minutes, and seconds since the last packet was successfully transmitted by an interface.
output hang	Number of hours, minutes, and seconds (or never) since the interface was last reset because of a transmission that took too long. When the number of hours in any of the “last” fields exceeds 24 hours, the number of days and hours is printed. If that field overflows, asterisks are printed.

Last clearing	Time at which the counters that measure cumulative statistics (such as number of bytes transmitted and received) shown in this report were last reset to zero. Note that variables that might affect routing (for example, load and reliability) are not cleared when the counters are cleared. *** indicates the elapsed time is too large to be displayed. 0:00:00 indicates the counters were cleared more than 2^{31}ms (and less than 2^{32}ms) ago.
Output queue, Input Queue, drops	Number of packets in output and input queues. Each number is followed by a slash, the maximum size of the queue, and the number of packets dropped due to a full queue.
Five-minute input rate Five-minute output rate	Average number of bits and packets transmitted per second in the last 5 minutes. The five-minute input and output rates should be used only as an approximation of traffic per second during a given 5-minute period. These rates are exponentially weighted averages with a time constant of 5 minutes. A period of four time constants must pass before the average will be within two percent of the instantaneous rate of a uniform stream of traffic over that period.
packets input	Total number of error-free packets received by the system.
bytes input	Total number of bytes, including data and MAC encapsulation, in the error free packets received by the system.
no buffers	Number of received packets discarded because there was no buffer space in the main system. Compare with ignored count. Broadcast storms on Ethernets and bursts of noise on serial lines are often responsible for no input buffer events.
broadcasts	Total number of broadcast or multicast packets received by the interface.
runts	Number of packets that are discarded because they are smaller than the medium's minimum packet size.
giants	Number of packets that are discarded because they exceed the medium's maximum packet size.
CRC	Cyclic redundancy checksum generated by the originating FDDI station or far-end device does not match the checksum calculated from the data received. This means that the data got corrupted somewhere between the sender and receiver. However, this field gives not indication of what caused the corruption. This usually indicates noise or transmission problems on the FDDI interface or the physical ring. On Cisco 2000, Cisco 2500, Cisco 3000, and Cisco 4000 series routers, it can also indicate hardware problems on either the sender or receiver FDDI units. A high number of CRCs is usually the result of collisions or a station transmitting bad data.
frame	Number of packets received incorrectly that have a CRC error and a noninteger number of octets. This is usually the result of collisions, but may also be the result of a failing fiber (due to cracks) or a hardware malfunction. It may also be the result of a corner-case bug in the FDDI chipset used in the AGS+. In general, these "errors" are spurious counts. However, if you are also seeing extra claims on the same FDDI interface, you should consider upgrading the FDDI applique.
overrun	Number of times the serial receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver's ability to handle the data.

ignored	Number of received packets ignored by the interface because the interface hardware ran low on internal buffers. These buffers are different than the system buffers mentioned previously in the buffer description. Broadcast storms and bursts of noise can cause the ignored count to be increased.
packets output	Total number of messages transmitted by the system.
bytes output	Total number of bytes, including data and MAC encapsulation, transmitted by the system.
underruns	Number of packets that caused a transmit abort. This indicates that the interface cannot transfer packets to the transmitter fast enough.
output errors	Sum of all errors that prevented the final transmission of datagrams out of the interface being examined. Note that this may not balance with the sum of the enumerated output errors, because some datagrams may have more than one error, and others may have errors that do not fall into any of the specifically tabulated categories.
collisions	Because an FDDI ring cannot have collisions, this statistic is always zero.
interface resets	Number of times an interface has been reset. The interface may be reset by the administrator or automatically when an internal error occurs.
restarts	Should always be zero for FDDI interfaces.
transitions	The number of times the ring made a transition from ring operational to ring nonoperational, or vice versa. A large number of transitions indicates a problem with the ring or the interface.
traces	Trace count applies to both the FCI, FCIT, and FIP. Indicates the number of times this interface started a trace.
claims	Pertains to FCIT and FIP only. Indicates the number of times this interface has been in claim state.
beacons	Pertains to FCIT and FIP only. Indicates the number of times the interface has been in beacon state.
Protocol	Protocol that is operating on the interface.
Pkts In	Number of packets received for that protocol.
Chars In	Number of characters received for that protocol.
Pkts Out	Number of packets transmitted for that protocol.
Chars Out	Number of characters transmitted for that protocol.

show interfaces hssi

Use the **show interfaces hssi** EXEC command to display information about the HSSI interface.

show interfaces hssi *unit* [**accounting**]
show interfaces hssi *slot/port* [**accounting**] (for the Cisco 7000)

Syntax Description

unit	Must match a port number on the selected interface.
accounting	(Optional.) Displays the number of packets of each protocol type that have been sent through the interface.
slot	On a Cisco 7000, optional slot location of the interface processor.
port	On a Cisco 7000, optional port number on interface.

Command Mode

EXEC

Sample Display

The following is sample output from the **show interfaces hssi** command when HSSI is enabled:

```
Router# show interfaces hssi 0

HSSI 0 is up, line protocol is up
Hardware is cBus HSSI
Internet address is 150.136.67.190, subnet mask is 255.255.255.0
MTU 4470 bytes, BW 45045 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation HDLC, loopback not set, keepalive set (10 sec)
Last input 0:00:03, output 0:00:00, output hang never
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
Five minute input rate 0 bits/sec, 0 packets/sec
Five minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 no buffer
    Received 0 broadcasts, 0 runts, 0 giants
              0 parity, 0 rx disabled
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
17 packets output, 994 bytes, 0 underruns
0 output errors, 0 applique, 4 interface resets, 0 restarts
2 carrier transitions
```

Table 1-26 describes significant fields shown in the display.

Table 1-26 Show Interfaces HSSI Field Descriptions

Field	Description
HSSI is {up | down} ...is administratively down	Indicates whether the interface hardware is currently active (whether carrier detect is present) and if it has been taken down by an administrator. "Disabled" indicates the router has received over 5000 errors in a keepalive interval, which is 10 seconds by default.
line protocol is {up | down | administratively down}	Indicates whether the software processes that handle the line protocol considers the line usable (that is, whether keepalives are successful).
Hardware	Specifies the hardware type.
Internet address	Lists the Internet address followed by subnet mask.
MTU	Maximum Transmission Unit of the interface.
BW	Bandwidth of the interface in kilobits per second.
DLY	Delay of the interface in microseconds.
rely	Reliability of the interface as a fraction of 255 (255/255 is 100% reliability), calculated as an exponential average over 5 minutes.
load	Load on the interface as a fraction of 255 (255/255 is completely saturated), calculated as an exponential average over 5 minutes.
Encapsulation	Encapsulation method assigned to interface.
loopback	Indicates whether loopback is set and type of loopback test.
keepalive	Indicates whether keepalives are set or not.
Last input	Number of hours, minutes, and seconds since the last packet was successfully received by an interface. Useful for knowing when a dead interface failed.
Last output	Number of hours, minutes, and seconds since the last packet was successfully transmitted by an interface.
output hang	Number of hours, minutes, and seconds (or never) since the interface was last reset because of a transmission that took too long. When the number of hours in any of the "last" fields exceeds 24 hours, the number of days and hours is printed. If that field overflows, asterisks are printed.
Last clearing	Time at which the counters that measure cumulative statistics (such as number of bytes transmitted and received) shown in this report were last reset to zero. Note that variables that might affect routing (for example, load and reliability) are not cleared when the counters are cleared. *** indicates the elapsed time is too large to be displayed. 0:00:00 indicates the counters were cleared more than 2^{31}ms (and less than 2^{32}ms) ago.
Output queue, drops Input queue, drops	Number of packets in output and input queues. Each number is followed by a slash, the maximum size of the queue, and the number of packets dropped due to a full queue.
Five minute input rate, Five minute output rate	Average number of bits and packets transmitted per second in the last 5 minutes.
packets input	Total number of error-free packets received by the system.

Field	Description
bytes input	Total number of bytes, including data and MAC encapsulation, in the error free packets received by the system.
no buffers	Number of received packets discarded because there was no buffer space in the main system. Compare with ignored count. Broadcast storms on Ethernets and bursts of noise on serial lines are often responsible for no input buffer events.
broadcasts	Total number of broadcast or multicast packets received by the interface.
runts	Number of packets that are discarded because they are smaller than the medium's minimum packet size.
giants	Number of packets that are discarded because they exceed the medium's maximum packet size.
parity	Report of the parity errors on the HSSI.
rx disabled	Indicates the HSSI could not find a free buffer on the ciscoBus controller to reserve for use for the HSSI receiver. When this happens, the HSSI shuts down its receiver and waits until a buffer is available. Data is not lost unless a packet comes in and overflows the HSSI FIFO. Usually, the receive disables are frequent but do not last for long, and the number of dropped packets is less than the count in the "rx disabled" field. A receive disabled condition can happen in systems that are under heavy traffic load and that have shorter packets. In this situation, the number of buffers available on the ciscoBus controller is at a premium. One way to alleviate this problem is to reduce the mtu on the HSSI interface from 4500 (FDDI size) to 1500 (Ethernet size). Doing so allows the software to take the fixed memory of the ciscoBus controller and divide it into a larger number of smaller buffers, rather than a small number of large buffers. Receive disables are not errors, so they are not included in any error counts.
input errors	Sum of all errors that prevented the receipt of datagrams on the interface being examined. This may not balance with the sum of the enumerated output errors, because some datagrams may have more than one error and others may have errors that do not fall into any of the specifically tabulated categories.
CRC	Cyclic redundancy checksum generated by the originating LAN station or far-end device does not match the checksum calculated from the data received. On a LAN, this usually indicates noise or transmission problems on the LAN interface or the LAN bus itself. A high number of CRCs is usually the result of collisions or a station transmitting bad data. On a serial link, CRCs usually indicate noise, gain hits, or other transmission problems on the data link. CRC errors are also reported when a far-end abort occurs, and when the idle flag pattern is corrupted. This makes it possible to get CRC errors even when there is no data traffic.
frame	Number of packets received incorrectly having a CRC error and a noninteger number of octets. On a serial line, this is usually the result of noise or other transmission problems.

Field	Description
overrun	Number of times the serial receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver's ability to handle the data.
ignored	Number of received packets ignored by the interface because the interface hardware ran low on internal buffers. These buffers are different than the system buffers mentioned previously in the buffer description. Broadcast storms and bursts of noise can cause the ignored count to be increased.
abort	Number of packets whose receipt was aborted.
input packets with dribble condition detected	Dribble bit condition indicates Carrier Sense did not go inactive on a receive data byte boundary.
packets output	Total number of messages transmitted by the system.
bytes output	Total number of bytes, including data and MAC encapsulation, transmitted by the system.
underruns	Number of times that the far-end transmitter has been running faster than the near-end router's receiver can handle. This may never happen (be reported) on some interfaces.
congestion drop	Number of messages discarded because the output queue on an interface grew too long. This can happen on a slow, congested serial link.
output errors	Sum of all errors that prevented the final transmission of datagrams out of the interface being examined. Note that this may not balance with the sum of the enumerated output errors, as some datagrams may have more than one error, and others may have errors that do not fall into any of the specifically tabulated categories.
applique	Indicates an unrecoverable error has occurred on the HSA applique. The system then invokes an interface reset.
interface resets	Number of times an interface has been completely reset. This can happen if packets queued for transmission were not sent within several seconds time. On a serial line, this can be caused by a malfunctioning modem that is not supplying the transmit clock signal, or by a cable problem. If the system notices that the carrier detect line of a serial interface is up, but the line protocol is down, it periodically resets the interface in an effort to restart it. Interface resets can also occur when an interface is looped back or shut down.
restarts	Number of times the controller was restarted because of errors.
carrier transitions	Number of times the carrier detect signal of a serial interface has changed state. Indicates modem or line problems if the carrier detect line is changing state often.
Protocol	Protocol that is operating on the interface.
Pkts In	Number of packets received for that protocol.
Chars In	Number of characters received for that protocol.
Pkts Out	Number of packets transmitted for that protocol.
Chars Out	Number of characters transmitted for that protocol.

The following is an example of the **show interfaces hssi** command on a Cisco 7000:

```
Router# show in hssi 1/0

Hssi1/0 is up, line protocol is up
  Hardware is cxBus HSSI
  Internet address is 131.108.38.14, subnet mask is 255.255.255.0
  MTU 1500 bytes, BW 45045 Kbit, DLY 1000000 usec, rely 255/255, load 1/255
  Encapsulation HDLC, loopback not set, keepalive set (10 sec)
  Last input 0:00:00, output 0:00:08, output hang never
  Last clearing of "show interface" counters never
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  Five minute input rate 1000 bits/sec, 2 packets/sec
  Five minute output rate 0 bits/sec, 0 packets/sec
     630573548 packets input, 2077237628 bytes, 0 no buffer
     Received 2832063 broadcasts, 0 runts, 0 giants
              0 parity, 1970 rx disabled
     113 input errors, 20 CRC, 93 frame, 0 overrun, 0 ignored, 0 abort
     629721628 packets output, 1934313295 bytes, 0 underruns
     0 output errors, 0 applique, 62 interface resets, 0 restarts
     309 carrier transitions
```

The following is an example of the **show interfaces hssi** command with the **accounting** option on a Cisco 7000:

```
> show interfaces hssi 1/0 accounting
HIP1/0
       Protocol    Pkts In   Chars In   Pkts Out  Chars Out
             IP       7344    4787842       1803    1535774
      Appletalk      33345    4797459      12781    1089695
        DEC MOP          0          0        127       9779
            ARP          7        420         39       2340
```

show interfaces loopback

Use the **show interfaces loopback** EXEC command to display information about the dialer interface.

show interfaces loopback [*unit*] [**accounting**]

Syntax Description

unit	(Optional.) Must match a port number on the selected interface.
accounting	(Optional.) Displays the number of packets of each protocol type that have been sent through the interface.

Command Mode

EXEC

Sample Displays

The following is sample output from the **show interfaces loopback** command:

```
Router# show int loop 0
Loopback0 is up, line protocol is up
  Hardware is Loopback
  MTU 1500 bytes, BW 1 Kbit, DLY 50 usec, rely 255/255, load 1/255
  Encapsulation UNKNOWN, loopback not set, keepalive set (10 sec)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Output queue 0/0, 0 drops; input queue 0/75, 0 drops
  Five minute input rate 0 bits/sec, 0 packets/sec
  Five minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets, 0 restarts
```

The following is sample output when the **accounting** keyword is included:

```
Router# show int loop 0 acc
Loopback0
               Protocol    Pkts In   Chars In   Pkts Out  Chars Out
No traffic sent or received on this interface.
Router#
```

Table 1-27 describes significant fields shown in the displays.

Table 1-27 Show Interfaces Loopback Descriptions

Field	Description
Loopback is {up | down} ...is administratively down	Indicates whether the interface hardware is currently active (whether carrier detect is present) and if it has been taken down by an administrator. "Disabled" indicates the router has received over 5000 errors in a keepalive interval, which is 10 seconds by default.
line protocol is {up | down | administratively down}	Indicates whether the software processes that handle the line protocol considers the line usable (that is, whether keepalives are successful).
Hardware	Hardware is Loopback.
MTU	Maximum Transmission Unit of the interface.
BW	Bandwidth of the interface in kilobits per second.
DLY	Delay of the interface in microseconds.
rely	Reliability of the interface as a fraction of 255 (255/255 is 100% reliability), calculated as an exponential average over 5 minutes.
load	Load on the interface as a fraction of 255 (255/255 is completely saturated), calculated as an exponential average over 5 minutes.
Encapsulation	Encapsulation method assigned to interface.
loopback	Indicates whether loopback is set and type of loopback test.
keepalive	Indicates whether keepalives are set or not.
Last input	Number of hours, minutes, and seconds since the last packet was successfully received by an interface. Useful for knowing when a dead interface failed.
Last output	Number of hours, minutes, and seconds since the last packet was successfully transmitted by an interface.
output hang	Number of hours, minutes, and seconds (or never) since the interface was last reset because of a transmission that took too long. When the number of hours in any of the "last" fields exceeds 24 hours, the number of days and hours is printed. If that field overflows, asterisks are printed.
Last clearing	Time at which the counters that measure cumulative statistics (such as number of bytes transmitted and received) shown in this report were last reset to zero. Note that variables that might affect routing (for example, load and reliability) are not cleared when the counters are cleared. *** indicates the elapsed time is too large to be displayed. 0:00:00 indicates the counters were cleared more than 2^{31}ms (and less than 2^{32}ms) ago.
Output queue, drops Input queue, drops	Number of packets in output and input queues. Each number is followed by a slash, the maximum size of the queue, and the number of packets dropped due to a full queue.
Five minute input rate, Five minute output rate	Average number of bits and packets transmitted per second in the last 5 minutes.
packets input	Total number of error-free packets received by the system.

Field	Description
bytes input	Total number of bytes, including data and MAC encapsulation, in the error free packets received by the system.
no buffer	Number of received packets discarded because there was no buffer space in the main system. Compare with ignored count. Broadcast storms on Ethernets and bursts of noise on serial lines are often responsible for no input buffer events.
broadcasts	Total number of broadcast or multicast packets received by the interface.
runts	Number of packets that are discarded because they are smaller than the medium's minimum packet size.
giants	Number of packets that are discarded because they exceed the medium's maximum packet size.
input errors	Sum of all errors that prevented the receipt of datagrams on the interface being examined. This may not balance with the sum of the enumerated output errors, because some datagrams may have more than one error and others may have errors that do not fall into any of the specifically tabulated categories.
CRC	Cyclic redundancy checksum generated by the originating LAN station or far-end device does not match the checksum calculated from the data received. On a LAN, this usually indicates noise or transmission problems on the LAN interface or the LAN bus itself. A high number of CRCs is usually the result of collisions or a station transmitting bad data. On a serial link, CRCs usually indicate noise, gain hits, or other transmission problems on the data link. CRC errors are also reported when a far-end abort occurs, and when the idle flag pattern is corrupted. This makes it possible to get CRC errors even when there is no data traffic.
frame	Number of packets received incorrectly having a CRC error and a noninteger number of octets. On a serial line, this is usually the result of noise or other transmission problems.
overrun	Number of times the serial receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver's ability to handle the data.
ignored	Number of received packets ignored by the interface because the interface hardware ran low on internal buffers. These buffers are different than the system buffers mentioned previously in the buffer description. Broadcast storms and bursts of noise can cause the ignored count to be increased.
abort	Number of packets whose receipt was aborted.
input packets with dribble condition detected	Dribble bit condition indicates Carrier Sense did not go inactive on a receive data byte boundary.
packets output	Total number of messages transmitted by the system.
bytes output	Total number of bytes, including data and MAC encapsulation, transmitted by the system.
underruns	Number of times that the far-end transmitter has been running faster than the near-end router's receiver can handle. This may never happen (be reported) on some interfaces.

Field	Description
output errors	Sum of all errors that prevented the final transmission of datagrams out of the interface being examined. Note that this may not balance with the sum of the enumerated output errors, as some datagrams may have more than one error, and others may have errors that do not fall into any of the specifically tabulated categories.
collisions	A loopback interface does not have collisions.
interface resets	Number of times an interface has been completely reset. This can happen if packets queued for transmission were not sent within several seconds time. On a serial line, this can be caused by a malfunctioning modem that is not supplying the transmit clock signal, or by a cable problem. If the system notices that the carrier detect line of a serial interface is up, but the line protocol is down, it periodically resets the interface in an effort to restart it. Interface resets can also occur when an interface is looped back or shut down.
restarts	Number of times the controller was restarted because of errors.
Protocol	Protocol that is operating on the interface.
Pkts In	Number of packets received for that protocol.
Chars In	Number of characters received for that protocol.
Pkts Out	Number of packets transmitted for that protocol.
Chars Out	Number of characters transmitted for that protocol.

show interfaces serial

Use the **show interfaces serial** command to display information about a serial interface.

show interfaces serial *unit* [**accounting**]
show interfaces serial *slot/port* [**accounting**] (for the Cisco 7000)

Syntax Description

unit	Must match the interface port number.
accounting	(Optional.) Displays the number of packets of each protocol type that have been sent through the interface.
slot	On a Cisco 7000, optional slot location of the interface processor.
port	On a Cisco 7000, optional port number on interface.

Command Mode

EXEC

Sample Display

The following is sample output from the **show interfaces** command for a synchronous serial interface:

```
Router# show interfaces serial
Serial 0 is up, line protocol is up
   Hardware is MCI Serial
   Internet address is 150.136.190.203, subnet mask is 255.255.255.0
   MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load 1/255
   Encapsulation HDLC, loopback not set, keepalive set (10 sec)
   Last input 0:00:07, output 0:00:00, output hang never
   Output queue 0/40, 0 drops; input queue 0/75, 0 drops
   Five minute input rate 0 bits/sec, 0 packets/sec
   Five minute output rate 0 bits/sec, 0 packets/sec
       16263 packets input, 1347238 bytes, 0 no buffer
       Received 13983 broadcasts, 0 runts, 0 giants
       2 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 2 abort
1 carrier transitions

     22146 packets output, 2383680 bytes, 0 underruns
     0 output errors, 0 collisions, 2 interface resets, 0 restarts
```

Table 1-28 describes significant fields shown in the display.

Table 1-28 Show Interfaces Serial Field Descriptions

Field	Description
Serial ... is {up | down} ...is administratively down	Indicates whether the interface hardware is currently active (whether carrier detect is present) and if it has been taken down by an administrator. "Disabled" indicates the router has received over 5000 errors in a keepalive interval, which is 10 seconds by default. "(spoofing)," which may be present for lines configured for DDR, indicates that spoofing is enabled. When spoofing is enabled the interface pretends to dial on demand in response to packets being routed to it when the interface is idle. Normally, no packets are routed to interfaces that are down, so the router interface must pretend to be up (that is, spoof) so that packets will be routed to it when it is not connected.
line protocol is {up | down}	Indicates whether the software processes that handle the line protocol consider the line usable (that is, whether keepalives are successful) or if it has been taken down by an administrator.
Hardware is	Specifies the hardware type.
Internet address is	Specifies the Internet address and subnet mask.
MTU	Maximum Transmission Unit of the interface.
BW 1544 Kbit	Indicates the value of the bandwidth parameter that has been configured for the interface (in kilobits per second). The bandwidth parameter is used to compute IGRP metrics only. If the interface is attached to a serial line with a line speed that does not match the default (1536 or 1544 for T1 and 56 for a standard synchronous serial line), use the **bandwidth** command to specify the correct line speed for this serial line.
DLY	Delay of the interface in microseconds.
rely	Reliability of the interface as a fraction of 255 (255/255 is 100% reliability), calculated as an exponential average over 5 minutes.
load	Load on the interface as a fraction of 255 (255/255 is completely saturated), calculated as an exponential average over 5 minutes.
Encapsulation	Encapsulation method assigned to interface.
loopback	Indicates whether loopback is set or not.
keepalive	Indicates whether keepalives are set or not.
Last input	Number of hours, minutes, and seconds since the last packet was successfully received by an interface. Useful for knowing when a dead interface failed.
Last output	Number of hours, minutes, and seconds since the last packet was successfully transmitted by an interface.
output hang	Number of hours, minutes, and seconds (or never) since the interface was last reset because of a transmission that took too long. When the number of hours in any of the "last" fields exceeds 24 hours, the number of days and hours is printed. If that field overflows, asterisks are printed.
Output queue, drops input queue, drops	Number of packets in output and input queues. Each number is followed by a slash, the maximum size of the queue, and the number of packets dropped due to a full queue.

Field	Description
Five minute input rate Five minute output rate	Average number of bits and packets transmitted per second in the last 5 minutes. The 5-minute input and output rates should be used only as an approximation of traffic per second during a given 5-minute period. These rates are exponentially weighted averages with a time constant of 5 minutes. A period of four time constants must pass before the average will be within two percent of the instantaneous rate of a uniform stream of traffic over that period.
packets input	Total number of error-free packets received by the system.
bytes input	Total number of bytes, including data and MAC encapsulation, in the error free packets received by the system.
no buffers	Number of received packets discarded because there was no buffer space in the main system. Compare with ignored count. Broadcast storms on Ethernets and bursts of noise on serial lines are often responsible for no input buffer events.
Received ... broadcasts	Total number of broadcast or multicast packets received by the interface.
runts	Number of packets that are discarded because they are smaller than the medium's minimum packet size.
giants	Number of packets that are discarded because they exceed the medium's maximum packet size.
input error	Total number of no buffer, runts, giants, CRCs, frame, overrun, ignored, and abort counts. Other input-related errors can also increment the count, so that this sum may not balance with the other counts.
CRC	Cyclic redundancy checksum generated by the originating station or far-end device does not match the checksum calculated from the data received. On a serial link, CRCs usually indicate noise, gain hits, or other transmission problems on the data link.
frame	Number of packets received incorrectly having a CRC error and a noninteger number of octets. On a serial line, this is usually the result of noise or other transmission problems.
overrun	Number of times the serial receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver's ability to handle the data.
ignored	Number of received packets ignored by the interface because the interface hardware ran low on internal buffers. Broadcast storms and bursts of noise can cause the ignored count to be increased.
abort	Illegal sequence of one bits on a serial interface. This usually indicates a clocking problem between the serial interface and the data link equipment.
input packets with dribble condition detected	Dribble bit condition indicates Carrier Sense did not go inactive on a receive data byte boundary.
packets output	Total number of messages transmitted by the system.
bytes output	Total number of bytes, including data and MAC encapsulation, transmitted by the system.
underruns	Number of times that the transmitter has been running faster than the router can handle. This may never be reported on some interfaces.

Field	Description
output errors	Sum of all errors that prevented the final transmission of datagrams out of the interface being examined. Note that this may not balance with the sum of the enumerated output errors, as some datagrams may have more than one error, and others may have errors that do not fall into any of the specifically tabulated categories.
collisions	Number of messages retransmitted due to an Ethernet collision. This usually is the result of an overextended LAN (Ethernet or transceiver cable too long, more than two repeaters between stations, or too many cascaded multiport transceivers). Some collisions are normal. However, if your collision rate climbs to around 4 or 5%, you should consider verifying that there is no faulty equipment on the segment and/or moving some existing stations to a new segment. A packet that collides is counted only once in output packets.
interface resets	Number of times an interface has been completely reset. This can happen if packets queued for transmission were not sent within several seconds' time. On a serial line, this can be caused by a malfunctioning modem that is not supplying the transmit clock signal, or by a cable problem. If the system notices that the carrier detect line of a serial interface is up, but the line protocol is down, it periodically resets the interface in an effort to restart it. Interface resets can also occur when an interface is looped back or shut down.
restarts	Number of times the controller was restarted because of errors.
carrier transitions	Number of times the carrier detect signal of a serial interface has changed state. Indicates modem or line problems if the carrier detect line is changing state often.

The following is sample output of the **show interfaces serial** command for the HDLC synchronous serial interface on a Cisco 7000:

```
> show interfaces serial 1/0
Serial1/0 is up, line protocol is up
  Hardware is cxBus Serial
  Internet address is 150.136.190.203, subnet mask is 255.255.255.0
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load 1/255
  Encapsulation HDLC, loopback not set, keepalive set (10 sec)
  Last input 0:00:07, output 0:00:00, output hang never
  Last clearing of "show interface" counters 2w4d
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  Five minute input rate 0 bits/sec, 0 packets/sec
  Five minute output rate 0 bits/sec, 0 packets/sec
     16263 packets input, 1347238 bytes, 0 no buffer
     Received 13983 broadcasts, 0 runts, 0 giants
     2 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 2 abort
     0 input packets with dribble condition detected
     22146 packets output, 2383680 bytes, 0 underruns
     0 output errors, 0 collisions, 2 interface resets, 0 restarts
     1 carrier transitions
```

Sample Display with Frame Relay Encapsulation

When using the Frame Relay encapsulation, use the **show interfaces** command to display information on the multicast DLCI, the DLCI of the interface, and the LMI DLCI used for the local management interface.

The multicast DLCI and the local DLCI can be set using the **frame-relay multicast-dlci** and the **frame-relay local-dlci** configuration commands, or provided through the local management interface. The status information is taken from the LMI, when active.

The following is sample output from the **show interfaces serial** command when using Frame Relay encapsulation.

```
Router# show interfaces serial

Serial 2 is up, line protocol is up
   Hardware type is MCI Serial
   Internet address is 131.108.122.1, subnet mask is 255.255.255.0
   MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load 1/255
   Encapsulation FRAME-RELAY, loopback not set, keepalive set (10 sec)
   multicast DLCI 1022,  status defined, active
   source DLCI    20, status defined, active
   LMI DLCI 1023, LMI sent 10, LMI stat recvd 10, LMI upd recvd 2
   Last input 7:21:29, output 0:00:37, output hang never
   Output queue 0/100, 0 drops; input queue 0/75, 0 drops
   Five minute input rate 0 bits/sec, 0 packets/sec
   Five minute output rate 0 bits/sec, 0 packets/sec
       47 packets input, 2656 bytes, 0 no buffer
       Received 5 broadcasts, 0 runts, 0 giants
       5 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 57 abort
       518 packets output, 391205 bytes
       0 output errors, 0 collisions, 0 interface resets, 0 restarts
       1 carrier transitions
```

In this display, the multicast DLCI has been changed to 1022 with the **frame-relay multicast-dlci** interface configuration command.

The display shows the statistics for the LMI are the number of status inquiry messages sent (LMI sent), the number of status messages received (LMI recvd), and the number of status updates received (upd recvd). See the *Frame Relay Interface* specification for additional explanations of this output.

Sample Display with ANSI LMI

For a serial interface with the ANSI LMI enabled, use the **show interfaces** command to determine the LMI type implemented.

The following is a sample display from the **show interfaces** output for a serial interface with the ANSI LMI enabled.

```
Router# show interfaces serial

Serial 1 is up, line protocol is up
   Hardware is MCI Serial
   Internet address is 131.108.121.1, subnet mask is 255.255.255.0
   MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load 1/255
   Encapsulation FRAME-RELAY, loopback not set, keepalive set
   LMI DLCI    0, LMI sent 10, LMI stat recvd 10
   LMI type is ANSI Annex D
   Last input 0:00:00, output 0:00:00, output hang never
   Output queue 0/40, 0 drops; input queue 0/75, 0 drops
   Five minute input rate 0 bits/sec, 1 packets/sec
   Five minute output rate 1000 bits/sec, 1 packets/sec
       261 packets input, 13212 bytes, 0 no buffer
       Received 33 broadcasts, 0 runts, 0 giants
       0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
       238 packets output, 14751 bytes, 0 underruns
       0 output errors, 0 collisions, 0 interface resets, 0 restarts
```

Notice that the **show interfaces** output for a serial interface with ANSI LMI shown in this display is very similar to that for encapsulation set to Frame Relay, as shown in the previous display. Table 1-29 describes the few differences that exist.

Table 1-29 Show Interfaces Serial Field Description with ANSI LMI

Field	Description
LMI DLCI 0	Identifies the DLCI used by the LMI for this interface. Default: 1023.
LMI sent 10	Number of LMI packets the router sent.
LMI type is ANSI Annex D	Indicates that the interface is configured for the ANSI-adopted Frame Relay specification T1.617 Annex D.

Sample Display with LAPB Encapsulation

Use the **show interfaces** command to display operation statistics for an interface using LAPB encapsulation.

The following is sample output from the **show interfaces** command for a serial interface using LAPB encapsulation:

```
Router# show interfaces

LAPB state is DISCONNECT, T1 3000, N1 12000, N2 20, K7, TH 3000
Window is closed
IFRAMEs 12/28 RNRs 0/1 REJs 13/1 SABMs 1/13 FRMRs 3/0 DISCs 0/11
```

Table 1-30 shows the fields relevant to all LAPB connections.

Table 1-30 Show Interfaces Serial Fields and Descriptions when LAPB is Enabled

Parameter	Description
LAPB state is DISCONNECT	State of the LAPB protocol.
T1 3000, N1 12000, ...	Current parameter settings.
Window is closed	Indicates that no more frames can be transmitted until some outstanding frames have been acknowledged.
IFRAMEs 12/28 RNRs 0/1 ...	Count of the different types of frames in the form of sent/ received.

Show Interfaces Serial with PPP

An interface configured for synchronous PPP encapsulation differs from the standard **show interface serial** output. An interface configured for PPP might include the following information.

```
lcp state = OPEN
ncp ipcp state = OPEN   ncp osicp state = NOT NEGOTIATED
ncp ipxcp state = NOT NEGOTIATED   ncp xnscp state = NOT NEGOTIATED
ncp vinescp state = NOT NEGOTIATED   ncp deccp state = NOT NEGOTIATED
ncp bridgecp state = NOT NEGOTIATED   ncp atalkcp state = NOT NEGOTIATED
```

Table 1-31 show the fields relevant to PPP connections.

Table 1-31 Show Interfaces Serial Fields and Descriptions with PPP Encapsulation

Field	Description
lcp state	Link Control Protocol.
ncp ipcp state	Network Control Protocol Internet Protocol Control Protocol
ncp osicp state	Network Control Protocol OSI (CLNS) Control Protocol
ncp ipxcp state	Network Control Protocol IPX (Novell) Control Protocol
ncp xnscp state	Network Control Protocol XNS Control Protocol
ncp vinescp state	Network Control Protocol VINES Control Protocol
ncp deccp state	Network Control Protocol DECnet Control Protocol
ncp bridgecp state	Network Control Protocol Bridging Control Protocol
ncp atalkcp state	Network Control Protocol AppleTalk Control Protocol

Sample Display with SDLC Connections

Use the **show interfaces** command to display the SDLC information for a given SDLC interface. The following is sample output from the **show interfaces** command for an SDLC primary interface supporting the SDLLC function.

```
Router# show interfaces

Serial 0 is up, line protocol is up
Hardware is MCI Serial
MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation SDLC-PRIMARY, loopback not set
    Timers (msec): poll pause 100 fair poll 500. Poll limit 1
    [T1 3000, N1 12016, N2 20, K 7] timer: 56608 Last polled device: none
    SDLLC [ma: 0000.0C01.14--, ring: 7 bridge: 1, target ring: 10
             largest token ring frame 2052]
SDLC addr C1 state is CONNECT
     VS 6, VR 3, RCNT 0, Remote VR 6, Current retransmit count 0
     Hold queue: 0/12 IFRAMEs 77/22 RNRs 0/0 SNRMs 1/0 DISCs 0/0
     Poll: clear, Poll count: 0, chain: p: C1 n: C1
     SDLLC [largest SDLC frame: 265, XID: disabled]
 Last input 00:00:02, output 00:00:01, output hang never
 Output queue 0/40, 0 drops; input queue 0/75, 0 drops
 Five minute input rate 517 bits/sec, 30 packets/sec
 Five minute output rate 672 bits/sec, 20 packets/sec
     357 packets input, 28382 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     926 packets output, 77274 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets, 0 restarts
     2 carrier transitions
```

Table 1-32 shows the fields relevant to all SDLC connections.

Table 1-32 Show Interfaces Serial Fields and Descriptions when SDLC is Enabled

Parameter	Description
Timers (msec): poll pause, fair poll, Poll limit	Current values of these timers, as described in the configuration section, for this interface.
T1, N1, N2, K	Values for these parameters, as described in the configuration section, for this interface.

Table 1-33 shows other data given for each SDLC secondary configured to be attached to this interface.

Table 1-33 SDLC Secondary Descriptions

SDLC Secondary	Description
addr	Address of this secondary.
state is	Current state of this connection, which is one of the following:
DISCONNECT	No communication is being attempted to this secondary.
CONNECT	A normal connect state exists between this router and this secondary.
DISCSENT	This router has sent a disconnect request to this secondary and is awaiting its response.
SNRMSENT	This router has sent a connect request (SNRM) to this secondary and is awaiting its response.
THEMBUSY	This secondary has told this router that it is temporarily unable to receive any more information frames.
USBUSY	This router has told this secondary that it is temporarily unable to receive any more information frames.
BOTHBUSY	Both sides have told each other that they are temporarily unable to receive any more information frames.
ERROR	This router has detected an error and is waiting for a response from the secondary acknowledging this.
VS	Sequence number of the next information frame this station sends.
VR	Sequence number of the next information frame from this secondary that this station expects to receive.
Remote VR	Last frame transmitted by this station that has been acknowledged by the other station.
Current retransmit count:	Number of times the current I-frame or sequence of I-frames has been retransmitted.
Hold Queue	Number of frames in hold queue/Maximum size of hold queue.
IFRAMEs, RNRs, SNRMs, DISCs	Sent/received count for these frames.

SDLC Secondary	Description
Poll	"Set" if this router has a poll outstanding to the secondary; "clear" if it does not.
Poll Count	Number of polls in a row that have been given to this secondary at this time.
Chain	Shows the previous (p) and next (n) secondary address on this interface in the *round robin loop* of polled devices.

Sample Display with SDLLC

Use the **show interfaces serial** command to display the SDLLC statistics for SDLLC configured interfaces.

The following is sample output from the **show interfaces serial** command for an a serial interface configured for SDLLC:

```
Router# show interfaces serial

Serial 0 is up, line protocol is up
   Hardware is MCI Serial
   MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load 1/255
   Encapsulation SDLC-PRIMARY, loopback not set
       Timers (msec): poll pause 100 fair poll 500. Poll limit 1
       [T1 3000, N1 12016, N2 20, K 7] timer: 56608 Last polled device: none
       SDLLC [ma: 0000.0C01.14--, ring: 7 bridge: 1, target ring: 10
              largest token ring frame 2052]
   SDLC addr C1 state is CONNECT
       VS 6, VR 3, RCNT 0, Remote VR 6, Current retransmit count 0
       Hold queue: 0/12 IFRAMEs 77/22 RNRs 0/0 SNRMs 1/0 DISCs 0/0
       Poll: clear, Poll count: 0, chain: p: C1 n: C1
       SDLLC [largest SDLC frame: 265, XID: disabled]
   Last input 00:00:02, output 00:00:01, output hang never
   Output queue 0/40, 0 drops; input queue 0/75, 0 drops
   Five minute input rate 517 bits/sec, 30 packets/sec
   Five minute output rate 672 bits/sec, 20 packets/sec
       357 packets input, 28382 bytes, 0 no buffer
       Received 0 broadcasts, 0 runts, 0 giants
       0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
       926 packets output, 77274 bytes, 0 underruns
       0 output errors, 0 collisions, 0 interface resets, 0 restarts
       6608 Last polled device: none
       SDLLC [ma: 0000.0C01.14--, ring: 7 brid2 carrier transitions
```

Most of the output shown in the display is generic to all SDLC encapsulated interfaces and is described in Chapter 23, "LLC2 and SDLC Commands." Table 1-34 shows the parameters specific to SDLLC.

Table 1-34 SDLLC Parameters

Parameter	Description
SDLLC ma	Lists the MAC address configured for this interface. The last byte is shown as "--" to indicate that it is filled in with the SDLC address of the connection.
ring, bridge, target ring	Lists the parameters as configured by the **sdllc traddr** command.
largest token ring frame	Shows the largest Token Ring frame that is accepted on the LLC2 side of the connection.
largest SDLC frame	Shows the largest SDLC frame that is accepted and will be generated on the SDLC side of the connection.
XID	Enabled or disabled: Shows whether XID processing is enabled on the SDLC side of the connection. If enabled, it will show the XID value for this address.

Sample Display with Accounting Option

The following example illustrates the **show interfaces serial** command with the **accounting** option on a Cisco 7000:

```
> show interfaces serial 1/0 accounting
Serial1/0
       Protocol    Pkts In   Chars In   Pkts Out  Chars Out
             IP       7344    4787842       1803    1535774
      Appletalk      33345    4797459      12781    1089695
        DEC MOP          0          0        127       9779
            ARP          7        420         39       2340
```

show interfaces tokenring

Use the **show interfaces tokenring** EXEC command to display information about the Token Ring interface and the state of source route bridging.

show interfaces tokenring *unit* [**accounting**]
show interfaces tokenring *slot/port* [**accounting**] (for the Cisco 7000)

Syntax Description

unit	Must match the interface port line number.
accounting	(Optional.) Displays the number of packets of each protocol type that have been sent through the interface.
slot	On a Cisco 7000, optional slot location of the interface processor.Value can be 0, 1, 2, 3, or 4.
port	On a Cisco 7000, optional port number on interface. Value can be 0, 1, 2, or 3.

Command Mode

EXEC

Usage Guidelines

If you do not provide values for the parameters *slot*, and *port*, the command will display statistics for all the network interfaces. The optional keyword **accounting** displays the number of packets of each protocol type that have been sent through the interface.

Sample Display

The following is sample output from the **show interfaces tokenring** command:

```
Router# show interfaces tokenring

TokenRing 0 is up, line protocol is up
Hardware is 16/4 Token Ring, address is 5500.2000.dc27 (bia 0000.3000.072b)
   Internet address is 150.136.230.203, subnet mask is 255.255.255.0
   MTU 8136 bytes, BW 16000 Kbit, DLY 630 usec, rely 255/255, load 1/255
   Encapsulation SNAP, loopback not set, keepalive set (10 sec)
   ARP type: SNAP, ARP Timeout 4:00:00
   Ring speed: 16 Mbps
   Single ring node, Source Route Bridge capable
   Group Address: 0x00000000, Functional Address: 0x60840000
   Last input 0:00:01, output 0:00:01, output hang never
   Output queue 0/40, 0 drops; input queue 0/75, 0 drops
   Five minute input rate 0 bits/sec, 0 packets/sec
   Five minute output rate 0 bits/sec, 0 packets/sec
   16339 packets input, 1496515 bytes, 0 no buffer
        Received 9895 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     32648 packets output, 9738303 bytes, 0 underruns
0 output errors, 0 collisions, 2 interface resets, 0 restarts
     5 transitions
```

Table 1-35 describes significant fields shown in the display.

Table 1-35 Show Interfaces Tokenring Field Descriptions

Field	Description
Token Ring is up | down	Interface is either currently active and inserted into ring (up) or inactive and not inserted (down). On the Cisco 7000, gives the interface processor type, slot number, and port number. "Disabled" indicates the router has received over 5000 errors in a keepalive interval, which is 10 seconds by default.
Token Ring is Reset	Hardware error has occurred.
Token Ring is Initializing	Hardware is up, in the process of inserting the ring.
Token Ring is Administratively Down	Hardware has been taken down by an administrator.
line protocol is {up | down | administratively down}	Indicates whether the software processes that handle the line protocol believe the interface is usable (that is, whether keepalives are successful).
Hardware	Hardware type. "Hardware is Token Ring" indicates that the board is a CSC-R board. "Hardware is 16/4 Token Ring" indicates that the board is a CSC-R16 board. Also shows the address of the interface.
Internet address	Lists the Internet address followed by subnet mask.
MTU	Maximum Transmission Unit of the interface.
BW	Bandwidth of the interface in kilobits per second.
DLY	Delay of the interface in microseconds.
rely	Reliability of the interface as a fraction of 255 (255/255 is 100% reliability), calculated as an exponential average over 5 minutes.
load	Load on the interface as a fraction of 255 (255/255 is completely saturated), calculated as an exponential average over 5 minutes.
Encapsulation	Encapsulation method assigned to interface.
loopback	Indicates whether loopback is set or not.
keepalive	Indicates whether keepalives are set or not.
ARP type:	Type of Address Resolution Protocol assigned.
Ring speed:	Speed of Token Ring—4 or 16 Mbps.
{Single ring/multiring node}	Indicates whether a node is enabled to collect and use source routing information (RIF) for routable Token Ring protocols.
Group Address:	Interface's group address, if any. The group address is a multicast address; any number of interfaces on the ring may share the same group address. Each interface may have at most one group address.
Last input	Number of hours, minutes, and seconds since the last packet was successfully received by an interface. Useful for knowing when a dead interface failed.
Last output	Number of hours, minutes, and seconds since the last packet was successfully transmitted by an interface.

Field	Description
output hang	Number of hours, minutes, and seconds (or never) since the interface was last reset because of a transmission that took too long. When the number of hours in any of the "last" fields exceeds 24 hours, the number of days and hours is printed. If that field overflows, asterisks are printed.
Last clearing	Time at which the counters that measure cumulative statistics (such as number of bytes transmitted and received) shown in this report were last reset to zero. Note that variables that might affect routing (for example, load and reliability) are not cleared when the counters are cleared. *** indicates the elapsed time is too large to be displayed. 0:00:00 indicates the counters were cleared more than 2^{31}ms (and less than 2^{32}ms) ago.
Output queue, drops Input queue, drops	Number of packets in output and input queues. Each number is followed by a slash, the maximum size of the queue, and the number of packets dropped due to a full queue.
Five minute input rate, Five minute output rate	Average number of bits and packets transmitted per second in the last 5 minutes. The 5-minute input and output rates should be used only as an approximation of traffic per second during a given 5-minute period. These rates are exponentially weighted averages with a time constant of 5 minutes. A period of four time constants must pass before the average will be within two percent of the instantaneous rate of a uniform stream of traffic over that period.
packets input	Total number of error-free packets received by the system.
bytes input	Total number of bytes, including data and MAC encapsulation, in the error free packets received by the system.
no buffers	Number of received packets discarded because there was no buffer space in the main system. Compare with ignored count. Broadcast storms on Ethernets and bursts of noise on serial lines are often responsible for no input buffer events.
broadcasts	Total number of broadcast or multicast packets received by the interface.
runts	Number of packets that are discarded because they are smaller than the medium's minimum packet size.
giants	Number of packets that are discarded because they exceed the medium's maximum packet size.
CRC	Cyclic redundancy checksum generated by the originating LAN station or far-end device does not match the checksum calculated from the data received. On a LAN, this usually indicates noise or transmission problems on the LAN interface or the LAN bus itself. A high number of CRCs is usually the result of a station transmitting bad data.
frame	Number of packets received incorrectly having a CRC error and a noninteger number of octets.
overrun	Number of times the serial receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver's ability to handle the data.

Field	Description
ignored	Number of received packets ignored by the interface because the interface hardware ran low on internal buffers. These buffers are different than the system buffers mentioned previously in the buffer description. Broadcast storms and bursts of noise can cause the ignored count to be increased.
packets output	Total number of messages transmitted by the system.
bytes output	Total number of bytes, including data and MAC encapsulation, transmitted by the system.
underruns	Number of times that the far-end transmitter has been running faster than the near-end router's receiver can handle. This may never be reported on some interfaces.
output errors	Sum of all errors that prevented the final transmission of datagrams out of the interface being examined. Note that this may not balance with the sum of the enumerated output errors, as some datagrams may have more than one error, and others may have errors that do not fall into any of the specifically tabulated categories.
collisions	Since a Token Ring cannot have collisions, this statistic is nonzero only if an unusual event occurred when frames were being queued or dequeued by the system software.
interface resets	Number of times an interface has been reset. The interface may be reset by the administrator or automatically when an internal error occurs.
Restarts	Should always be zero for Token Ring interfaces.
transitions	Number of times the ring made a transition from up to down, or vice versa. A large number of transitions indicates a problem with the ring or the interface.

The following is sample output from the **show interfaces tokenring** command on a Cisco 7000:

```
Router> show interfaces tokenring 2/0

TokenRing2/0 is administratively down, line protocol is down
  Hardware is cxBus Token Ring, address is 0000.3040.8b4a (bia 0000.3040.8b4a)
  MTU 8136 bytes, BW 16000 Kbit, DLY 630 usec, rely 255/255, load 1/255
  Encapsulation SNAP, loopback not set, keepalive set (10 sec)
  ARP type: SNAP, ARP Timeout 4:00:00
  Ring speed: 0 Mbps
  Single ring node, Source Route Transparent Bridge capable
  Ethernet Transit OUI: 0x0000F8
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  Five minute input rate 0 bits/sec, 0 packets/sec
  Five minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 1 interface resets, 0 restarts
     1 transitions
```

The following example on the Cisco 70000 includes the **accounting** option.When you use the accounting option, only the accounting statistics are displayed.

```
> show interfaces tokenring 2/0 accounting
TokenRing2/0
       Protocol    Pkts In   Chars In   Pkts Out  Chars Out
             IP       7344    4787842       1803    1535774
      Appletalk      33345    4797459      12781    1089695
        DEC MOP          0          0        127       9779
            ARP          7        420         39       2340
```

show interfaces tunnel

To list tunnel interface information, use the **show interfaces tunnel** EXEC command.

show interfaces tunnel *unit* [**accounting**]

Syntax Description

unit	Must match the interface port line number.
accounting	(Optional.) Displays the number of packets of each protocol type that have been sent through the interface.

Command Mode

EXEC

Sample Display

The following is sample output from the **show interface tunnel** command:

```
Router# show interfaces tunnel 4
Tunnel4 is up, line protocol is down
  Hardware is Routing Tunnel
  MTU 1500 bytes, BW 9 Kbit, DLY 500000 usec, rely 255/255, load 1/255
  Encapsulation TUNNEL, loopback not set, keepalive set (10 sec)
  Tunnel source 0.0.0.0, destination 0.0.0.0
  Tunnel protocol/transport GRE/IP, key disabled, sequencing disabled
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Output queue 0/0, 0 drops; input queue 0/75, 0 drops
  Five minute input rate 0 bits/sec, 0 packets/sec
  Five minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets, 0 restarts
```

Table 1-36 describes significant fields shown in the display.

Table 1-36 Show Interfaces Tunnel Field Descriptions

Field	Description
Tunnel is up | down	Interface is currently active and inserted into ring (up) or inactive and not inserted (down). On the Cisco 7000, gives the interface processor type, slot number, and port number.
line protocol is {up | down | administratively down}	Shows line protocol up if a valid route is available to the tunnel destination. Shows line protocol down if no route is available, or if the route would be recursive.
Hardware	Specifies the hardware type.
MTU	Maximum Transmission Unit of the interface.
BW	Bandwidth of the interface in kilobits per second.
DLY	Delay of the interface in microseconds.

Field	Description
rely	Reliability of the interface as a fraction of 255 (255/255 is 100% reliability), calculated as an exponential average over 5 minutes.
load	Load on the interface as a fraction of 255 (255/255 is completely saturated), calculated as an exponential average over 5 minutes.
Encapsulation	Encapsulation method is always TUNNEL for tunnels.
loopback	Indicates whether loopback is set or not.
keepalive	Indicates whether keepalives are set or not.
Tunnel source	IP address used as the source address for packets in the tunnel.
destination	IP address of the host destination.
Tunnel protocol	Tunnel transport protocol (the protocol the tunnel is using). This is based on the **tunnel mode** command, which defaults to GRE.
key	ID key for the tunnel interface, unless disabled.
sequencing	Indicates whether the tunnel interface drops datagrams that arrive out of order. Can be disabled.
Last input	Number of hours, minutes, and seconds since the last packet was successfully received by an interface. Useful for knowing when a dead interface failed.
Last output	Number of hours, minutes, and seconds since the last packet was successfully transmitted by an interface.
output hang	Number of hours, minutes, and seconds (or never) since the interface was last reset because of a transmission that took too long. When the number of hours in any of the "last" fields exceeds 24 hours, the number of days and hours is printed. If that field overflows, asterisks are printed.
Last clearing	Time at which the counters that measure cumulative statistics (such as number of bytes transmitted and received) shown in this report were last reset to zero. Note that variables that might affect routing (for example, load and reliability) are not cleared when the counters are cleared. *** indicates the elapsed time is too large to be displayed. 0:00:00 indicates the counters were cleared more than 2^{31}ms (and less than 2^{32}ms) ago.
Output queue, drops Input queue, drops	Number of packets in output and input queues. Each number is followed by a slash, the maximum size of the queue, and the number of packets dropped due to a full queue.
Five minute input rate, Five minute output rate	Average number of bits and packets transmitted per second in the last 5 minutes. The 5-minute input and output rates should be used only as an approximation of traffic per second during a given 5-minute period. These rates are exponentially weighted averages with a time constant of 5 minutes. A period of four time constants must pass before the average will be within two percent of the instantaneous rate of a uniform stream of traffic over that period.
packets input	Total number of error-free packets received by the system.
bytes input	Total number of bytes, including data and MAC encapsulation, in the error free packets received by the system.

Field	Description
no buffers	Number of received packets discarded because there was no buffer space in the main system. Compare with ignored count. Broadcast storms on Ethernets and bursts of noise on serial lines are often responsible for no input buffer events.
broadcasts	Total number of broadcast or multicast packets received by the interface.
runts	Number of packets that are discarded because they are smaller than the medium's minimum packet size.
giants	Number of packets that are discarded because they exceed the medium's maximum packet size.
CRC	Cyclic redundancy checksum generated by the originating LAN station or far-end device does not match the checksum calculated from the data received. On a LAN, this usually indicates noise or transmission problems on the LAN interface or the LAN bus itself. A high number of CRCs is usually the result of a station transmitting bad data.
frame	Number of packets received incorrectly having a CRC error and a noninteger number of octets.
overrun	Number of times the serial receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver's ability to handle the data.
ignored	Number of received packets ignored by the interface because the interface hardware ran low on internal buffers. These buffers are different than the system buffers mentioned previously in the buffer description. Broadcast storms and bursts of noise can cause the ignored count to be increased.
abort	Illegal sequence of one bits on a serial interface. This usually indicates a clocking problem between the serial interface and the data link equipment.
input packets with dribble condition detected	Dribble bit condition indicates Carrier Sense did not go inactive on a receive data byte boundary.
packets output	Total number of messages transmitted by the system.
bytes output	Total number of bytes, including data and MAC encapsulation, transmitted by the system.
underruns	Number of times that the far-end transmitter has been running faster than the near-end router's receiver can handle. This may never be reported on some interfaces.
output errors	Sum of all errors that prevented the final transmission of datagrams out of the interface being examined. Note that this may not balance with the sum of the enumerated output errors, as some datagrams may have more than one error, and others may have errors that do not fall into any of the specifically tabulated categories.

Field	Description
collisions	Number of messages retransmitted due to an Ethernet collision. This usually is the result of an overextended LAN (Ethernet or transceiver cable too long, more than two repeaters between stations, or too many cascaded multiport transceivers). Some collisions are normal. However, if your collision rate climbs to around 4 or 5%, you should consider verifying that there is no faulty equipment on the segment and/or moving some existing stations to a new segment. A packet that collides is counted only once in output packets.
interface resets	Number of times an interface has been reset. The interface may be reset by the administrator or automatically when an internal error occurs.
Restarts	Number of times the controller was restarted because of errors.

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

show interfaces
show ip route †
show route †

show queueing

To list the current state of the queue lists, use the **show queueing** EXEC command.

show queueing [**custom** | **priority**]

Syntax Description

custom	Shows status of custom queue lists.
priority	Shows status of priority lists.

Command Mode

EXEC

Usage Guidelines

If no keyword is entered, this command show the status of both custom and priority queue lists.

Sample Display

The following is sample output from the **show queueing custom** EXEC command:

```
Router# show queueing custom
Current custom queue configuration:

List   Queue  Args
3      10     default
3      3      interface Tunnel3
3      3      protocol ip
3      3      byte-count 444 limit 3
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

custom-queue-list
priority-group
priority-list interface †
priority-list queue-limit †
priority-list stun †
queue-list default †
queue-list interface †
queue-list protocol †
queue-list queue byte-count †
queue-list queue limit †
queue-list stun †

shutdown

To disable an interface, use the **shutdown** interface configuration command. To restart a disabled interface, use the **no shutdown** command.

shutdown
no shutdown

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Usage Guidelines

The **shutdown** command disables all functions on the specified interface. On serial interfaces, this command causes the DTR signal to be dropped. On Token Ring interfaces, this command causes the interface to be deinserted from the ring. On FDDI interfaces, this command causes the optical bypass switch, if present, to go into bypass mode.

This command also marks the interface as unavailable. To check whether an interface is disabled, use the EXEC command **show interfaces**. An interface that has been shut down is shown as administratively down in the display from this command.

Examples

The following example turns off interface Ethernet 0:

```
interface ethernet 0
shutdown
```

The following example turns the interface back on:

```
interface ethernet 0
no shutdown
```

Related Command

show interfaces

slip

To make a SLIP connection on the auxiliary port, use the **slip** EXEC command.

slip [**default** | *client* [**@***tacacs-server*]] [**/routing**] [**/compressed**]

Syntax Description

default	(Optional.) Makes a SLIP connection when a default address has been configured.
client	(Optional.) IP address or the name of the client workstation or PC.
@*tacacs-server*	(Optional.) IP address or IP hostname of the TACACS server to which the user's TACACS authentication request is sent.
/routing	(Optional.) Indicates routing is enabled. Interface async 1 must be configured for **async dynamic routing**.
/compressed	(Optional.) Indicates IP header compression should be used on the link.

Default

None

Command Mode

EXEC

Usage Guidelines

The IP address (indicated by the *client* and *tacacs-server* arguments) is the IP address of the system dialing in. This parameter can only be specified if the line is set for dynamic addresses using the line configuration command **async address dynamic**.

If you do not specify an address or enter **default**, you will be prompted for an IP address or host name. You can enter **default** at this point to use the default address configured for the line.

Using the *tacacs-server* argument, you can specify a TACACS server. If you do not specify a TACACS server for SLIP address authentication, the TACACS server specified at login (if any) will be used for the SLIP address query.

The **/routing** switch indicates that the remote system is a router and that routing messages should be exchanged over the link. The **/routing** switch can only be used if the line is configured for **async dynamic routing**.

The **/compressed** switch indicates that IP header compression should be used on the link. This switch can only be specified if the underlying asynchronous interface has been configured for **ip tcp header-compression passive**. Compression is always used if the interface is configured for **ip tcp header-compression on** and it is never used if the interface is configured for **ip tcp header-compression off**.

You cannot configure SLIP on the console port.

Examples

The following example illustrates how to make a connection when a permanent SLIP address has been assigned. Once a correct password is entered, you are placed in SLIP mode, and the IP address is displayed.

```
Router> slip
Password:
Entering SLIP mode.
Your IP address is 192.31.7.28, MTU is 1524 bytes
```

The following example illustrates the prompts displayed and the response required when dynamic addressing is used to assign the SLIP address:

```
Router> slip
IP address or hostname? 192.31.6.15
Password:
Entering SLIP mode
Your IP address is 192.31.6.15, MTU is 1524 bytes
```

In the following example, the address 192.31.6.15 has been assigned as the default. Password verification is still required before SLIP mode can be enabled.

```
Router> slip default
Password:
Entering SLIP mode
Your IP address is 192.31.6.15, MTU is 1524 bytes
```

The following example illustrates the implementation of header compression on the interface with the IP address 128.66.2.1:

```
Router> slip /compressed 128.66.2.1
Password:
Entering SLIP mode.
Interface IP address is 128.66.2.1, MTU is 1500 bytes.
Header compression is On.
```

When a line is configured for **slip header-compression passive**, and you use the **slip** EXEC command to enter SLIP mode, you will see that the interface is set to match compression status indicated at the EXEC level.

The message “Header compression will match your system” indicates that the router starts compressing when it first receives a compressed packet. You can override this and force compression with /**compression**. If the line was configured for **slip header-compression on**, this line would read “Header compression is On.” TACACS server check is specified for address authentication.

```
Router> slip 1.0.0.1@check
Password:
Entering SLIP mode.
Interface IP address is 1.0.0.1, MTU is 1500 bytes
Header compression will match your system.
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

async default ip address
async dynamic address
async dynamic routing
async mode dedicated
async mode interactive

encapsulation ppp
interface async 1
tacacs-server †

smt-queue-threshold

To set the maximum number of unprocessed FDDI station management (SMT) frames that will be held for processing, use the **smt-queue-threshold** global configuration command. Use the **no smt-queue-threshold** command to restore the queue to the default.

smt-queue-threshold *number*
no smt-queue-threshold

Syntax Description

number Number of buffers used to store unprocessed SMT messages that are to be queued for processing. Acceptable values are positive integers.

Default

The default threshold value is equal to the number of FDDI interfaces installed in the router.

Command Mode

Global configuration

Usage Guidelines

This command helps ensure that the routers keep track of FDDI *upstream* and *downstream* neighbors, particularly when a router includes more than one FDDI interface.

In FDDI, upstream and downstream neighbors are determined by transmitting and receiving SMT Neighbor Information Frames (NIFs). The router can appear to lose track of neighbors when it receives an SMT frame and the queue currently contains an unprocessed frame. This occurs because the router discards incoming SMT frames if the queue is full. Discarding SMT NIF frames can cause the router to lose its upstream or downstream neighbor.

Note Use this command carefully, because the SMT buffer is charged to the inbound interface (input hold queue) until the frame is completely processed by the system. Setting this value to a high limit can impact buffer usage and the ability of the router to receive routable packets or routing updates.

Example

The following example specifies that the SMT queue can hold ten messages. As SMT frames are processed by the system, the queue is decreased by one:

```
smt-queue-threshold 10
```

squelch

To extend the Ethernet twisted-pair 10BaseT capability beyond the standard 100 meters on the Cisco 4000 platform, use the **squelch** interface configuration command.

squelch {**normal** | **reduced**}
no squelch {**normal** | **reduced**}

Syntax Description

normal Allows normal capability.

reduced Allows extended 10BaseT capability.

Default

Normal range

Command Mode

Interface configuration

Example

The following example extends the twisted-pair 10BaseT capability on the cable attached to interface Ethernet 2:

```
interface ethernet 2
squelch reduced
```

systat

To display information about the active ports of the router, enter the **systat** EXEC command.

systat [**all**]

Syntax Description

all (Optional.) Displays information for both active and inactive ports.

Command Mode

EXEC

Example

The following example shows how to use the **systat** command:

```
Router> systat

  Line     User      Host(s)              Idle Location
   0 con 0                                    charnel console
   1 tty 1                                    T2500 #1-1
   2 tty 2                                    T2500 #1-2
   3 tty 3                                    T2500 #1-3
   4 tty 4   xyz       LANE                  56 T2500 #1-4
   5 tty 5                                    T2500 #A1
   6 tty 6                                    3262 #B1
   7 tty 7   train     ABC                    0 3262 #B2
   8 tty 8                                    3262 #A2
   9 tty 9   pzwt      XRemote:  6 clients    0 3262 #B2
```

The information displayed includes the line number, connection name, idle time, and terminal location.

transmit-clock-internal

When a DTE does not return a transmit clock, use the **transmit-clock-internal** interface command to enable the internally generated clock on a serial interface on a Cisco 7000. Use the **no** form of this command to disable the feature.

transmit-clock-internal
no transmit-clock-internal

Syntax Description

This command has no keywords or arguments.

Default

no transmit-clock-internal

Command Mode

Interface configuration

Example

In the following example, the internally generated clock is enabled on serial interface 3/0:

```
interface serial 3/0
transmit-clock-internal
```

transmitter-delay

To specify a minimum dead-time after transmitting a packet, use the **transmitter-delay** interface configuration command. The **no transmitter-delay** command restores the default.

transmitter-delay {*microseconds* | *hdlc-flags*}
no transmitter-delay

Syntax Description

microseconds	Approximate number of microseconds of minimum delay after transmitting a packet on the MCI and SCI interface cards.
hdlc-flags	Minimum number of HDLC flags to be sent between each packet on the IGS serial interface, HIP, HSCI, FSIP, or HSSI. The valid range on the IGS is 2 to 62; the valid range on the HSSI is 2 to 128000.

Default

0 microseconds

Command Mode

Interface configuration

Usage Guidelines

This command is especially useful for serial interfaces that can send back-to-back data packets over serial interfaces faster than some hosts can receive them.

The transmitter delay feature is implemented for the following Token Ring cards: CSC-R16, CSC-R16M, CSC-1R, CSC-2R, and CSC-CTR. For the first four cards, the command syntax is the same as the existing command and specifies the number of milliseconds to delay between sending frames that are generated by the router. Transmitter delay for the CSC-CTR uses the same syntax, but specifies a relative time interval to delay between transmission of all frames.

Example

The following example specifies a delay of 300 microseconds on interface serial 0:

```
interface serial 0
transmitter-delay 300
```

tunnel checksum

To enable encapsulator-to-decapsulator checksumming of packets on a tunnel interface, use the **tunnel checksum** interface configuration command. To disable checksumming, use the **no** form of the command.

tunnel checksum
no tunnel checksum

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This command currently applies to generic route encapsulation (GRE) only. Some passenger protocols rely on media checksums to provide data integrity. By default, the tunnel does not guarantee packet integrity. By enabling end-to-end checksums, the routers will drop corrupted packets.

Example

In the following example, all protocols will have encapsulator-to-decapsulator checksumming of packets on the tunnel interface:

```
tunnel checksum
```

tunnel destination

To specify a tunnel interface's destination, use the **tunnel destination** interface configuration command. To remove the destination, use the **no** form of this command.

tunnel destination {*host name* | *ip address*}
no tunnel destination

Syntax Description

host name	Name of the host destination.
ip address	IP address of the host destination.

Default

None

Command Mode

Interface configuration

Usage Guidelines

You cannot have two tunnels using the same encapsulation mode with exactly the same source and destination address. The workaround is to create a loopback interface and source packets off of the loopback interface.

Example

In the following example, the tunnel destination is 131.222.111.234:

```
tunnel destination 131.222.111.234
```

Related Command

tunnel source

tunnel key

To enable an ID key for a tunnel interface, use the **tunnel key** interface configuration command. To remove the ID key, use the **no** form of this command.

tunnel key *key-number*
no tunnel key

Syntax Description

key-number	Integer from 0 to 4294967295.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This command currently applies to generic route encapsulation (GRE) only. Tunnel ID keys can be used as a form of *weak* security to prevent misconfiguration or injection of packets from a foreign source.

Note When using GRE, the ID key is carried in each packet. We do *not* recommend relying on this key for security purposes.

Example

In the following example, the tunnel key is set to 3:

```
tunnel key 3
```

tunnel mode

To set the encapsulation mode for the tunnel interface, use the **tunnel mode** interface configuration command. To set to the default, use the **no** form of the command.

tunnel mode {**cayman** | **eon** | **gre** | **nos**}
no tunnel mode

Syntax Description

cayman	Cayman TunnelTalk AppleTalk encapsulation.
eon	EON compatible CLNS tunnel.
gre	Generic route encapsulation protocol over IP.
nos	KA9Q/NOS compatible IP over IP.

Default

gre

Command Mode

Interface configuration

Example

In the following example, the tunnel mode is set to EON:

```
tunnel mode eon
```

tunnel sequence-datagrams

To configure a tunnel interface to drop datagrams that arrive out of order, use the **tunnel sequence-datagrams** interface configuration command. To disable this function, use the **no** form of this command.

tunnel sequence-datagrams
no tunnel sequence-datagrams

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This command currently applies to generic route encapsulation (GRE) only. This command is useful when carrying passenger protocols that behave poorly when they receive packets out of order (for example, LLC2-based protocols).

Example

In the following example, the tunnel is configured to drop datagrams that arrive out of order:

```
tunnel sequence-datagrams
```

tunnel source

To set a tunnel interface's source address, use the **tunnel source** interface configuring command. To remove the source address, use the **no** form of the command.

tunnel source *ip address* | *interface-type interface-number*
no tunnel source

Syntax Description

ip address	IP address to use as the source address for packets in the tunnel.
interface-type	All types.
interface-number	Specifies the port, connector, or interface card 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.

Default

None

Command Mode

Interface configuration

Usage Guidelines

You cannot have two tunnels using the same encapsulation mode with exactly the same source and destination address. The workaround is to create a loopback interface and source packets off of the loopback interface.

When using tunnels to Cayman boxes, you must set the **tunnel source** to an explicit IP address on the same subnet as the Cayman box, not the tunnel itself.

Example

In the following example, the tunnel source is set to the IP address assigned to ethernet 0:

```
Router(config-if)# tunnel source ethernet 0
```

Related Command

tunnel destination

tx-queue-limit

To control the number of transmit buffers available to a specified interface on the MCI and SCI cards, use the **tx-queue-limit** interface configuration command.

tx-queue-limit *number*

Syntax Description

number	Maximum number of transmit buffers that the specified interface can subscribe.

Default

Defaults depend on the total transmit buffer pool size and the traffic patterns of all the interfaces on the card. Defaults and specified limits are displayed with the **show controllers mci** EXEC command.

Command Mode

Interface configuration

Usage Guidelines

This command should be used only under the guidance of a technical support representative.

Example

The following example sets the maximum number of transmit buffers on the interface to 5:

```
interface ethernet 0
tx-queue-limit 5
```

Related Command

show controllers mci