X.25 Configuration Commands

Use the commands in this chapter to configure Link Access Procedure, Balanced (LAPB), X.25, DDN X.25, and Blacker Front-end Encryption (BFE). X.25 provides remote terminal access; routing using the IP, DECnet, XNS, ISO CLNS, AppleTalk, Novell IPX, Banyan VINES, and Apollo Domain protocols; and bridging. For X.25 and LAPB configuration information and examples, refer to the "Configuring X.25" chapter of the *Protocol Translator Configuration Guide*.

access-class

To configure an incoming access class on virtual terminals, use the **access-class** line configuration command.

> **access-class** *access-list-number* **in**

Syntax Description

access-list-number	An integer between 1 and 199 that you select for the access list.

Default

None

Command Mode

Line configuration

Usage Guidelines

The access list number is used for both incoming TCP access and incoming PAD access.

In the case of TCP access, the communication server uses the IP access list defined using the **access-list** command.

For incoming PAD connections, the same numbered X.29 access list is referenced. If you only want to have access restrictions on one of the protocols, you can create an access list that permits all addresses for the other protocol.

Example

The following example configures an incoming access class on virtual terminal line 4.

```
line vty 4
access-class 4 in
```

Related Command

x29 access-list
access-list

bfe

To set the protocol translator to participate in emergency mode or to end participation in emergency mode when your system is configured for **x25 bfe-emergency decision** and **x25 bfe-decision ask**, use the **bfe** EXEC command.

> **bfe** {**enter** | **leave**} *interface-type number*

Syntax Description

enter	Causes the protocol translator to send a special address translation packet that includes an "enter emergency mode" command to the BFE if the emergency mode window is open. If the BFE is already in emergency mode, this command enables the sending of address translation information.
leave	Disables the sending of address translation information from the protocol translator to the BFE when the BFE is in emergency mode.
interface-type	Indicates the command mode of interface.
number	Indicates the number of the interface.

Default
None

Command Mode
EXEC

Example
The following example illustrates how to set the protocol translator to participate in BFE mode:

```
bfe enter interface serial 0
```

Related Commands
encapsulation bfex25
x25 bfe-decision
x25 bfe-emergency

clear x25-vc

To clear virtual circuits (VCs), use the **clear x25-vc** privileged EXEC command. This command clears all X.25 virtual circuits at once.

> **clear x25-vc** *interface-type interface-number* [*lcn*]

Syntax Description

interface-type	Specifies the interface type.
interface-number	Specifies the interface unit number.
lcn	(Optional.) Specifies a virtual circuit.

Command Mode
EXEC

Example
The following example illustrates how to clear all VCs:

```
clear x25-vc
```

Related Command
x25 idle

cmns enable

To enable Connection-Mode Network Service (CMNS) on a nonserial interface, use the **cmns enable** interface configuration command. To disable this feature, use the **no** form of this command.

cmns enable
no cmns enable

Syntax Description

This command has no arguments or keywords.

Default

Enabled whenever an X.25 encapsulation command is included with a serial interface configuration.

Command Mode

Interface configuration

Usage Guidelines

After processing this command, all the X.25-related interface configuration commands are made available on the LAN interfaces (Ethernet and Token Ring), as well as on serial interfaces.

Example

The following example illustrates how to enable CMNS on Ethernet interface 0:

```
interface ethernet 0
cmns enable
```

Related Commands

x25 htc
x25 map cmns

encapsulation bfex25

To configure BFE encapsulation on a protocol translator attached to a BFE device, use the **encapsulation bfex25** interface configuration command.

encapsulation bfex25

Syntax Description

This command has no arguments or keywords.

Default

HDLC

Command Mode

Interface configuration

Usage Guidelines

You must choose an X.25 encapsulation method. This encapsulation operates to map between Class A IP addresses and the type of X.121 addresses expected by the BFE encryption device.

Example

The following example sets BFE encapsulation on interface serial 0:

```
interface serial 0
encapsulation bfex25
```

Related Command

bfe

encapsulation ddnx25

To have a protocol translator using DDN X.25 act as a DTE device, use the **encapsulation ddnx25** interface configuration command.

> **encapsulation ddnx25**

Syntax Description

This command has no arguments or keywords.

Default

HDLC

Command Mode

Interface configuration

Usage Guidelines

You must choose an X.25 encapsulation method.

This encapsulation operates to map between IP addresses and the type of X.121 addresses expected by the BFE encryption device.

A protocol translator using DDN X.25 Standard Service can act as either a DTE or a DCE device.

Example

The following example sets DTE DDN X.25 device operation on interface serial 0:

```
interface serial 0
encapsulation ddnx25
```

encapsulation ddnx25-dce

To have a protocol translator using DDN X.25 act as a DCE device, use the **encapsulation ddnx25-dce** interface configuration command.

encapsulation ddnx25-dce

Syntax Description

This command has no arguments or keywords.

Default

HDLC

Command Mode

Interface configuration

Usage Guidelines

You must choose an X.25 encapsulation method. This encapsulation operates to map between IP addresses and the type of X.121 addresses expected by the DDN.

A protocol translator using DDN X.25 Standard Service can act as either a DTE or a DCE device.

Example

The following example sets DCE DDN X.25 device operation on interface serial 0:

```
interface serial 0
encapsulation ddnx25-dce
```

encapsulation lapb

To run datagrams over a DTE serial interface using LAPB encapsulation, use the **encapsulation lapb** interface configuration command.

> **encapsulation lapb**

Syntax Description

This command has no arguments or keywords.

Default

HDLC

Command Mode

Interface configuration

Usage Guidelines

You must choose an X.25 encapsulation method. One end of the link must be DTE, and the other end must be DCE.

Example

The following example sets LAPB DTE encapsulation on interface serial 3:

```
interface serial 3
encapsulation lapb
```

Related Commands

encapsulation lapb-dce
lapb protocol

encapsulation lapb-dce

To run datagrams over a DCE serial interface using LAPB encapsulation, use the **encapsulation lapb-dce** interface configuration command.

> **encapsulation lapb-dce**

Syntax Description

This command has no arguments or keywords.

Default

HDLC

Command Mode

Interface configuration

Usage Guidelines

You must choose an X.25 encapsulation method. One end of the link must be DTE, and the other end must be DCE.

Example

The following example sets LAPB DCE encapsulation on interface serial 3:

```
interface serial 3
encapsulation lapb-dce
```

Related Commands

encapsulation lapb
lapb protocol

encapsulation multi-lapb

For DTE operation, to enable use of multiple local-area network (LAN) protocols on the same line at the same time, use the **encapsulation multi-lapb** interface configuration command.

> **encapsulation multi-lapb**

Syntax Description

This command has no arguments or keywords.

Default

HDLC

Command Mode

Interface configuration

Usage Guidelines

You must choose an X.25 encapsulation method. With the **encapsulation multi-lapb** command, you can use multiple protocols such as IP, DECnet, and XNS at the same time. Both ends of the line must use the same encapsulation; one end of the link must be DCE and the other end DTE.

Example

The following example illustrates how to set multiple protocols on a LAPB line for DTE operation:

```
interface serial 0
encapsulation multi-lapb
```

Related Command

encapsulation multi-lapb-dce

encapsulation multi-lapb-dce

For DCE operation, to enable use of multiple LAN protocols on the same line at the same time, use the **encapsulation multi-lapb-dce** interface configuration command.

encapsulation multi-lapb-dce

Syntax Description

This command has no arguments or keywords.

Default

HDLC

Command Mode

Interface configuration

Usage Guidelines

You must choose an X.25 encapsulation method. With the **encapsulation multi-lapb-dce** command, you can use multiple protocols such as IP, DECnet, and XNS at the same time. Both ends of the line must use the same encapsulation; one end of the link must be DCE and the other end DTE.

Example

The following example illustrates how to set multiple protocols on a LAPB line for DCE operation.

```
interface serial 0
encapsulation multi-lapb-dce
```

Related Command

encapsulation multi-lapb

encapsulation x25

To set X.25 DTE operation, use the **encapsulation x25** interface configuration command.

> **encapsulation x25**

Syntax Description

This command has no arguments or keywords.

Default

HDLC

Command Mode

Interface configuration

Usage Guidelines

You must choose an X.25 encapsulation method. A protocol translator using X.25 Level 3 encapsulation can act as a DTE or DCE device on general X.25 networks.

Example

The following example sets X.25 DTE operation on interface serial 0:

```
interface serial 0
encapsulation x25
```

Related Command

encapsulation x25-dce

encapsulation x25-dce

To set X.25 DCE operation, use the **encapsulation x25-dce** interface configuration command.

encapsulation x25-dce

Syntax Description

This command has no arguments or keywords.

Default

HDLC

Command Mode

Interface configuration

Usage Guidelines

You must choose an X.25 encapsulation method. A protocol translator using X.25 Level 3 encapsulation can act as a DTE or DCE device on general X.25 networks.

Example

The following example sets X.25 DCE operation on interface serial 0:

```
interface serial 0
encapsulation x25-dce
```

Related Command

encapsulation x25

ip tcp header-compression

To implement TCP packet header compression, use the **ip tcp header-compression** interface configuration command. To disable this feature, use the **no** form of this command.

> **ip tcp header-compression** [**passive**]
> **no ip tcp header-compression** [**passive**]

Syntax Description

passive	(Optional.) Outgoing packets are compressed only if incoming TCP packets on the VC for a TCP header compression map are compressed. When the **passive** option is not set, all compressible traffic intended for the TCP header compression address map is compressed.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

The header compression routine complies with the IETF RFC 1144 standard for header compression.

The implementation of Compressed TCP over X.25 uses a separate virtual circuit (VC) to pass the compressed packets from any VCs used for noncompressed packets.

The header compression increases the speed of interactive TCP/IP sessions over serial lines running at 56/64 kilobits per second or slower by caching the 20 bytes or so of the constant part of the IP packet header.

Example

The following example establishes packet compression on interface serial 4:

```
interface serial 4
ip tcp header compression
x25 map compressedtcp 131.08.2.5 000000010300 broadcast
```

Related Command

x25 map compressedtcp

lapb hold-queue

To define the number of packets to be held until they can be sent over the LAPB connection, use the **lapb hold-queue** interface configuration command. Use the **no lapb hold-queue** command without an argument to remove this command from the configuration file and return to the default value.

lapb hold-queue *queue-size*
no lapb hold-queue [*queue-size*]

Syntax Description

queue-size	Number of packets in the hold queue. A hold queue limit of 0 allows an unlimited number of packets in the hold queue.

Default

12 packets

Command Mode

Interface configuration

Example

The following example illustrates how to set the X.25 hold queue to hold 25 packets:

```
interface serial 0
lapb hold-queue 25
```

lapb k

To specify the maximum permissible number of outstanding frames, called the window size (the k parameter), use the **lapb k** interface configuration command.

lapb k *window-size*

Syntax Description

window-size	Number of outstanding frames. It can be a value from 1 to 7.

Default

7 packets

Command Mode

Interface configuration

Example

The following example changes the LAPB window size (the K parameter) to three packets:

```
interface serial 0
lapb k 3
```

lapb n1

To specify the maximum number of bits (n1 bits) a frame can hold, use the **lapb n1** interface configuration command.

lapb n1 *bits*

Syntax Description

bits	Number of bits. This is a decimal integer from 1088 to 32,840. It must be a multiple of eight.

Default

12056 bits (1500 bytes)

Command Mode

Interface configuration

Usage Guidelines

The **lapb n1** command sets the LAPB N1 parameter.

It is not necessary to set N1 to an exact value to support a particular X.25 data packet size, although both ends of a connection should have the same N1 value. The N1 parameter serves to avoid processing of any huge frames that result from a "jabbering" interface, an unlikely event.

The Cisco N1 default value corresponds to the hardware interface buffer size. Any changes to this value must allow for an X.25 data packet and LAPB frame overhead.The software supports an X.25 data packet with a maximum packet size plus three or four bytes of overhead for modulo 8 or 128 operation, respectively, and LAPB frame overhead of two bytes of header for modulo 8 operation plus two bytes of CRC.

In addition, the various standards bodies specify that N1 be given in bits rather than bytes. While some equipment can be configured using bytes or by automatically adjusting for some of the overhead information present, Cisco devices are configured using the true value of N1.

Table 1-1 specifies the *minimum* N1 values needed to support a given X.25 data packet. Note that N1 cannot be set to a value less than what is required to support an X.25 data packet size of 128 bytes under modulo 128 operation. This is because all X.25 implementations must be able to support 128-byte data packets.

Table 1-1   Minimum LAPB N1 Values

Maximum data in X.25 packet	Minimum N1 value for X.25 modulo 8	Minimum N1 value for X.25 modulo 128
128	1088	1088
256	2104	2112
512	4152	4160
1024	8240	8256
2048	16440	16448
4096	32824	32832

Configuring N1 to be less than 2104 will generate a warning message that X.25 may have problems because some nondata packets can use up to 259 bytes.

The N1 parameter cannot be set to a value larger than the default without first increasing the hardware maximum transmission unit (MTU) size.

The X.25 software will accept default packet sizes and CALLs that specify maximum packet sizes greater than what the LAPB layer will support, but will negotiate the CALLs placed on the interface to the largest value that can be supported. For switched CALLs, the packet size negotiation takes place end-to-end through the protocol translator so the CALL will not have a maximum packet size that exceeds the capability of either of the two interfaces involved.

Example

The following example sets the number of bits the frame can hold (the n1 parameter) to 9600:

```
interface serial 0
lapb n1 9600
```

lapb n2

To specify the maximum number of times (the N2 parameter) a data frame can be retransmitted, use the **lapb n2** interface configuration command.

lapb n2 *retries*

Syntax Description

retries	Number of times a data frame can be retransmitted. This can be an integer from 1 to 255.

Default

20 retransmissions

Command Mode

Interface configuration

Example

The following example sets the number of retries (the N2 parameter) to 50.

```
interface serial 0
lapb n2 50
```

lapb protocol

To configure a protocol on the LAPB line, use the **lapb protocol** interface configuration command.

lapb protocol *protocol*

Syntax Description

protocol	Indicates the protocol. Protocol choice: **ip**, **xns**, **decnet**, **appletalk**, **vines**, **clns** (ISO CLNS), **ipx** (Novell IPX), and **apollo**.

Default

IP

Command Mode

Interface configuration

Usage Guidelines

This command is valid only if encapsulation commands are set first.

Example

The following example sets IP as the protocol on the LAPB line:

```
interface serial 1
lapb protocol ip
```

Related Commands

encapsulation lapb
encapsulation lapb-dce

lapb t1

To set the limit retransmission timer period (the t1 parameter), use the **lapb t1** interface configuration command.

> **lapb t1** *milliseconds*

Syntax Description

milliseconds	Length of the retransmission timer, in milliseconds. It can be an integer from 1 to 64,000. The default is 3000 ms.

Default

3000 milliseconds

Command Mode

Interface configuration

Usage Guidelines

The retransmission timer determines how long a transmitted frame can remain unacknowledged before the LAPB software polls for an acknowledgment.

To determine an optimal value for the retransmission timer, use the privileged EXEC command **ping** to measure the round-trip time of a maximum-sized frame on the link. Multiply this time by a safety factor that takes into account the speed of the link, the link quality, and the distance. A typical safety factor is 1.5. Choosing a larger safety factor can result in slower data transfer if the line is noisy. However, this disadvantage is minor compared to the excessive retransmissions and effective bandwidth reduction caused by a timer setting that is too small.

Example

The following example sets the T1 retransmission timer to 20,000 milliseconds.

```
interface serial 0
lapb t1 20000
```

show cmns

To display information about CMNS traffic activity, use the **show cmns** EXEC command.

show cmns [*interface-name*]

Syntax Description

interface-name	(Optional.) Interface about which to display information.

Command Mode

EXEC

Usage Guidelines

You can use this command to display X.25 Level 3 parameters for LAN interfaces (such as Ethernet or Token Ring).

Sample Display

The following is sample output from the **show cmns** command:

```
Ethernet1 is administratively down, line protocol is down
  Hardware address is 0000.0c02.5f4c, (bia 0000.0c2.5f4c), state R1
    Modulo 8, idle 0, timer 0, nvc 1
    Window size: input 2, output 2, Packet size: input 128, output 128
    Timer: TH 0
    Channels: Incoming-only none, Two-way 1-4095, Outgoing-only none
    RESTARTs 0/0 CALLs 0+0/0+0/0+0 DIAGs 0/0
```

Table 1-2 describes the fields shown in the display.

Table 1-2   Show CMNS Field Descriptions

Field	Description
Ethernet1 is ...	Status of the interface. It can be active and inserted into network (up), inactive and not inserted (down), or disabled (administratively down).
line protocol is ...	Indicates whether the software processes that handle the line protocol believes the interface is usable.
Hardware address	MAC address for this interface.
bia	Burned-in address.
state R1	State of the interface. It should always be R1.
modulo 8	Modulo value, which determines the packet sequence numbering scheme used.
idle 0	Number of minutes the protocol translator waits before closing idle virtual circuits.
timer 0	Value of interface timer. It should always be zero.
nvc 1	Maximum number of simultaneous virtual circuits permitted to and from a single host for a particular protocol.
Window size:	Default window size (in packets) for the interface. Use the **x25 facility** interface configuration command to override these default values for the switched virtual circuits originated by the protocol translator.

Field	Description
input 2	Size of the default input window size.
output 2	Size of the default output window size.
Packet size:	Default packet size for the interface. Use the **x25 facility** interface configuration command to override these default values for the switched virtual circuits originated by the protocol translator.
input 128	Input packet size.The default is 1024.
output 128	Output packet size.The default is 1024 bytes.
Timer: TH 0	X.25 hold timer for delayed acknowledgment. It should always be zero.
Channels: Incoming-only none Two-way 1-4095 Outgoing-only none	Channel sequence range for this interface.
RESTARTs 0/0	Number of restarts sent/received.
CALLs 0+0/0+0/0+0	Number of successful calls + failed calls/calls sent + calls failed/calls received + calls failed.
DIAGs 0/0	Number of diagnostic messages sent+received.

Related Command

show interfaces serial

show interfaces serial

To display information about a serial interface, use the **show interfaces serial** EXEC command.

   show interfaces serial *number*

Syntax Description

number	Number of the serial port.

Command Mode

EXEC

Sample Displays

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

```
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 1-3 shows the fields relevant to all LAPB connections.

Table 1-3   Show Interfaces Serial Fields and Descriptions when LAPB is Enabled

Parameter	Description
LAPB state is ...	State of the LAPB protocol.
T1 3000, N1 12000, ...	Current parameter settings.
Window is closed	No more frames can be transmitted until some outstanding frames have been acknowledged.
VS	Modulo 8 frame number to give to the next outgoing I frame.
VR	Modulo 8 frame number to give to the next I frame expected to be received.
RCNT	Number of received I frames that have not yet been acknowledged.
Remote VR	Number of the next I frame the remote expects to receive.
Retransmissions	Count of I frames that have been retransmitted.
IFRAMEs	Count of Information frames in the form of sent/received.
RNRs	Count of Receiver Not Ready frames in the form of sent/received.
REJs	Count of Reject frames in the form of sent/received.
SABMs	Count of Set Asynchronous Balanced Mode commands in the form of sent/received.
FRMRs	Count of Frame Reject frames in the form of sent/received.
DISCs	Count of Disconnect commands in the form of sent/received.

The following is partial sample output from the **show interfaces serial** output for a serial interface that is using X.25 encapsulation:

```
X25 address 000000010100, state R1, modulo 8, idle 0, timer 0, nvc 1
   Window size: input 2, output 2, Packet size: input 128, output 128
   Timers: T20 180, T21 200, T22 180, T23 180, TH 0
   (configuration on RESTART: modulo 8,
   Window size: input 2 output 2, Packet size: input 128, output 128
   Channels: Incoming-only none, Two-way 5-1024, Outgoing-only none)
   RESTARTs 3/2 CALLs 0+0/0+0/0+0/ DIAGs 0/0
```

The stability of the X.25 protocol requires that some parameters not be changed without a RESTART of the protocol. Any change to these parameters will be held until a RESTART is sent or received. If any of these parameters will change, the configuration on RESTART information will be output as well as the values that are currently in effect.

Table 1-4 describes the fields shown in the display.

Table 1-4   Show Interfaces X25 Field Descriptions

Field	Description
X25 address 000000010100	Calling address used in the Call Request packet.
state R1	State of the interface. Possible values include: • R1 is the normal ready state • R2 is the DCE not-ready state •R3 is the DTE not-ready state If the state is R2 or R3, the device is awaiting acknowledgment for a Restart packet.
modulo 8	Modulo value; determines the packet sequence numbering scheme used.
idle 0	Number of minutes the protocol translator waits before closing idle virtual circuits.
timer 0	Value of the interface timer, which is zero unless the interface state is R2 or R3.
nvc 1	Maximum number of simultaneous virtual circuits permitted to and from a single host for a particular protocol.
Window size: input 2, output 2	Default window size (in packets) for the interface. The **x25 facility** interface configuration command can be used to override these default values for the switched virtual circuits originated by the protocol translator.
Packet size: input 128, output 128	Default packet size (in bytes) for the interface. The **x25 facility** interface configuration command can be used to override these default values for the switched virtual circuits originated by the protocol translator.
Timers: T20 180, T21 200, T22 180, T23 180, TH 0	Values of the Request packet timers: • T10 through T13 for a DCE device • T20 through T23 for a DTE device
Channels: Incoming-only none Two-way 5-1024 Outgoing-only none	Channel sequence range for this interface.
RESTARTs 3/2	Shows RESTART packet statistics for the interface using the format Sent/ Received.
CALLs 0+0/0+0/0+0	Shows CALL packet statistics for the interface using these formats: • Successful+Failed / • Sent+Failed sent / • Received+Failed received
DIAGs 0/0	Shows DIAG packet statistics for the interface using the format Forwarded+Failed forwarded.

Related Command
show cmns

show llc2

To display active LLC2 connections, use the **show llc2** EXEC command.

show llc2c

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show llc2** command:

```
TokenRing0 DTE=1000.5A59.04F9,400022224444 SAP=04/04, State=NORMAL
V(S)=5, V(R)=5, Last N(R)=5, Local window=7, Remote Window=127
ack-max=3, n2=8, Next timer in 7768
xid-retry timer 0/60000 ack timer 0/1000
p timer 0/1000 idle timer 7768/10000
rej timer 0/3200 busy timer 0/9600
ack-delay timer 0/3200
CMNS Connections to:
 Address 1000.5A59.04F9 via Ethernet2
 Protocol is up
 Interface type X25-DCE RESTARTS 0/1
 Timers: T10 1 T11 1 T12 1 T13 1
```

The display includes a CMNS addendum, indicating that LLC2 is running with CMNS. When LLC2 is not running with CMNS, the **show llc2** command does not display a CMNS addendum.

Table 1-5 describes significant fields shown in the display.

Table 1-5   Show LLC2 Field Descriptions

Field	Description
TokenRing0	Name of interface on which the session is established.
DTE=1000.5A59.04F9, 400022224444	Address of the station to which ethicist protocol translator is talking on this session. (The protocol translator's address is the MAC address of the interface on which the connection is established, except when Local Acknowledgment or SDLLC is used, in which case the address used by the protocol translator is shown as in this example, following the DTE address and separated by a comma.)
SAP=04/04	Other station's and protocol translator's (remote/local) Service Access Point for this connection. The SAP is analogous to a "port number" on the protocol translator and allows for multiple sessions between the same two stations.

Field	Description
State=	Current state of the LLC2 session, which are any of the following:
ADM	Asynchronous Disconnect Mode—A connection is not established, and either end can begin one.
SETUP	Request to begin a connection has been sent to the remote station, and this station is waiting for a response to that request.
RESET	A previously open connection has been reset because of some error by this station, and this station is waiting for a response to that reset command.
D_CONN	This station has requested a normal, expected, end of communications with the remote, and is waiting for a response to that disconnect request.
ERROR	This station has detected an error in communications and has told the other station about it. This station is waiting for a reply to its posting of this error.
NORMAL	Connection between the two sides is fully established, and normal communication is occurring.
BUSY	Normal communication state exists, except busy conditions on this station make it such that this station cannot receive information frames from the other station at this time.
REJECT	Out-of-sequence frame has been detected on this station, and this station has requested that the other resend this information.
AWAIT	Normal communication exists, but this station has had a timer expire, and is trying to recover from it (usually by resending the frame that started the timer).
AWAIT_BUSY	A combination of the AWAIT and BUSY states.
AWAIT_REJ	A combination of the AWAIT and REJECT states.
V(S)=5	Sequence number of the next Information (I) frame this station will send.
V(R)=5	Sequence number of the next I frame this station expects to receive from the other station.
Last N (R)=5	Last sequence number of this station's transmitted frames acknowledged by the remote station.
Local Window=7	Number of frames this station can send before requiring an acknowledgment from the remote station.
Remote Window=127	Number of frames this station can accept from the remote.
ack-max=3, n2=8	Value of these parameters, as given in the previous configuration section.
Next timer in 7768	Number of milliseconds before the next timer, for any reason, goes off.
xid-retry timer 0/60000	A series of timer values in the form of next-time/time-between, where "next-time" is the next time, in milliseconds, that the timer will wake, and "time-between" is the time, in milliseconds, between each timer wakeup. A "next-time" of zero indicates that the given timer is not enabled, and will never wake.
CMNS Connections to:	CMS addendum when LLC2 is running with the CMNS protocol contains the following:
Address 1000.5A59.04F9 via Ethernet2	MAC address of remote station.
Protocol is up	Up indicates the LLC2 and X.25 protocols are in a state where incoming and outgoing Call Requests can be made on this LLC2 connection.

Field	Description
Interface type X25-DCE	Type of interface. It can be X25-DCE, X25-DTE, or X25-DXE (both DTE and DCE).
RESTARTS 0/1	Restarts sent/received on this LLC2 connection.
Timers:	T10, T11, T12, T13 (or T20, T21, T22, T23 for DTE); these are Request packet timers. These are similar in function to X.25 parameters of the same name.

show x25 map

To display information about configured address maps, use the **show x25 map** EXEC command:

show x25 map

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show x25 map** command.

```
Serial0: IP 131.108.170.1 1311001 PERMANENT BROADCAST, 2 LCN: 3 4*
Serial0: appletalk 128.1 1311005 PERMANENT
Serial1: BRIDGE 1311006 PERMANENT
```

The display shows that three virtual circuits have been configured for the protocol translator, two for the Serial0 interface, and one for the Serial1 interface.

Table 1-6 describes the fields shown in the first line of output in the display.

Table 1-6   Show X.25 Map Field Descriptions

Field	Description
Serial0	Interface for which this X.25 virtual circuit has been configured.
IP	Indicates the type of higher-level address that has been configured for this virtual circuit using the **x25 map** command. The BRIDGE value in this field indicates that all bridged packets go to this X.121 address.
131.108.170.1	Higher-level address that has been configured for this virtual circuit.
1311001	X.121 address that has been configured for this virtual circuit.
PERMANENT	Indicates the address-mapping type that has been configured for the interface in this entry. Possible values include: • CONSTRUCTED—Derived using the DDN address conversion scheme. • PERMANENT—Indicates that the address was entered using the **x25 map** interface configuration. • TEMPORARY—Indicates that the address mapping was not entered using a configuration command, but was dynamically created instead.
BROADCAST	If broadcasts are enabled for an address mapping, the word BROADCAST also appears on the output line.
2 LCN:	If the number of logical circuit numbers (LCNs) is greater than zero, the line of output also includes the LCN numbers.
3 4	Indicates the LCNs, if one or more exists.
*	Marks the current LCN.

show x25 remote-red

To display the one-to-one mapping of the host IP addresses and the remote BFE device's IP addresses, use the **show x25 remote-red** EXEC command.

> **show x25 remote-red**

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show x25 remote-red** command:

```
Entry        REMOTE-RED      REMOTE-BLACK    INTERFACE
1            21.0.0.3        21.0.0.7        serial3
2            21.0.0.10       21.0.0.6        serial1
3            21.0.0.24       21.0.0.8        serial3
```

Table 1-7 describes the fields shown in the display.

Table 1-7   Show X.25 Remote-Red Display Field Descriptions

Field	Description
Entry	Number of the address mapping entry.
REMOTE-RED	Host IP address.
REMOTE-BLACK	IP address of the remote BFE device.
INTERFACE	Name of interface through which communication with the remote BFE device will take place.

show x25 route

To display the X.25 routing table, use the **show x25 route** EXEC command.

> **show x25 route**

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show x25 route** command:

```
Number          X.121       CUD         Forward To
1               1311001                 Serial0, 0 uses
2               1311002                 131.108.170.10, 0 uses
3               1311003     00          alias Serial0, 2 uses
```

Table 1-8 describes the fields shown in the display.

Table 1-8   Show X.25 Route Display Field Descriptions

Field	Description
Number	Number identifying the entry in the X.25 routing table.
X.121	X.121 address pattern associated with this entry.
CUD	Call User Data, if any, that has been configured for this route.
Forward To	Interface or IP address to which the protocol translator will forward a CALL destined for the X.121 address pattern in this entry.
Uses	Number of times this route has been used.

Related Command

x25 route

show x25 vc

To display active X.25 virtual circuit parameters and statistics, use the **show x25 vc** EXEC command.

show x25 vc *interface-type interface-number* [*lcn*]

Syntax Description

interface-type	Specifies the interface type.
interface-number	Specifies the interface unit number.
lcn	(Optional.) Logical channel number (LCN) about which to display active X.25 virtual circuit parameters and statistics.

Command Mode
EXEC

Usage Guidelines

For PVCs, the syntax of the third and sometimes fourth lines of **show x25 vc** output varies depending on whether the PVC is in a connected or disconnected state, and whether the connection is locally switched, or remotely tunneled over a TCP connection.

If the PVC is locally switched and connected, the syntax for the third line of output follows:

```
Switched PVC to interface name PVC #, connected
```

If the PVC is locally switched and not connected, the syntax for the third line of output follows:

```
Switched PVC to interface name PVC #, not connected, PVC state string
```

If the PVC is remotely tunneled and connected, the syntax for the third and fourth lines of output follows:

```
Tunneled PVC to ip address interface name PVC #, connected
via TCP connection from ip address, port to ip address, port, D-bit allowed
```

If the PVC is remotely tunneled and not connected, the syntax for the third line of output follows:

```
Tunneled PVC to ip address interface name PVC #, not connected, PVC state string
```

The PVC state string represents the state of a PVC. Some of these strings only apply to PVCs that are remotely tunneled over a TCP connection. The %X25-3-PVCBAD system error message (as documented in the *System Error Messages* publication), and the **debug x25** all command (as documented in the *Debug Command Reference* publication) also use these PVC state strings. These PVC state strings follow:

```
awaiting PVC-SETUP reply
can't support flow control values
connected
dest. disconnected
dest. interface is not up
dest. PVC configuration mismatch
mismatched flow control values
no such dest. interface
no such dest. PVC
non-X.25 dest. interface
PVC setup protocol error
PVC/TCP connect timed out
PVC/TCP connection refused
PVC/TCP routing error
trying to connect via TCP
waiting to connect
```

Sample Display

The following is sample output from the **show x25 vc** command for an SVC that carries encapsulated IP diagrams:

```
LC1: 1, State: D1, Interface: Serial0
Started 0:55:03, last input 0:54:56, output 0:54:56
Connected to IP [10.4.0.32] <->000000320400 Precedent: 0
Window size input: 7, output: 7
Packet size input: 1024, output: 1024
PS: 2 PR: 6 Remote PR: 2 RCNT: 1 RNR: FALSE
Window is closed
Retransmits: 0   Timer (secs): 0   Reassembly (bytes): 0
Held Fragments/Packets: 0/0
Bytes 1111/588 Packets 18/22 Resets 0/0 RNRs 0/0 REJs 0/0 INTs 0/0
```

Table 1-9 describes significant fields shown in the display.

Table 1-9   Show X25 VC Field Descriptions

Field	Description
LCI	Virtual circuit number.
State	State of the virtual circuit (which is independent of the states of other virtual circuits); D1 is the normal ready state. (See the CCITT X.25 recommendation for a description of virtual circuit states.)
Interface	Interface used for the virtual circuit.
Started	Time elapsed since the virtual circuit was created.
last input	Time of last input.
output	Time of last output.
Connected to	Network-protocol address, in brackets, and the X.121 address of the machine to which the protocol translator is locally connected.
Precedent	IP precedence (appears only if you have specified DDN encapsulation).
Window size	Window size for the virtual circuit.

Field	Description
Packet size	Packet size for the virtual circuit.
PS	Current send sequence number.
PR	Current receive sequence number.
Remote PR	Last PR number received from the other end of the circuit.
RCNT	Count of unacknowledged input packets.
RNR	State of the Receiver Not Ready flag; this field is true if the network sends a receiver-not-ready packet.
Window is closed	protocol translator cannot transmit any more frames until the remote node has acknowledged some outstanding packets.
Retransmits	Number of times a supervisory packet (RESET or CLEAR) has been retransmitted.
Timer (secs)	A nonzero time value if a packet has not been acknowledged or if virtual circuits are being timed for inactivity.
Reassembly (bytes)	Number of bytes received for a partial packet (a packet in which the more data bit is set).
Held Fragments/Packets	Number of X.25 packets being held. (In this case, Fragments refers to the X.25 fragmentation of higher-level data packets.)
Bytes	Total number of bytes sent and received. The Packets, Resets, RNRs, REJs, and INTs fields show the total sent and received packet counts of the indicated types. (RNR is Receiver Not Ready, REJ is Reject, and INT is Interrupt).

Sample Display Showing CMNS Virtual Circuit Parameters and Statistics

When the protocol type used for the connection is CMNS, the display generated with **show x25 vc** differs slightly from the display outlined in the preceding description.

The following is sample output from the **show x25 vc** command for two complementary interfaces, both running CMNS, and transmitting CMNS traffic to each other:

```
LCI: 1,  State: P4,  Interface: Serial1
Started 0:23:00, last input never, output never
Connected to CMNS [37.1111] <--> 313131 via Ethernet1 LCN 4095 to 0000.0c01.487d
Window size input: 6, output: 6
Packet size input: 1024, output: 1024
PS: 0  PR: 0  ACK: 0  Remote PR: 0  RCNT: 0  RNR: FALSE
Retransmits: 0  Timer (secs): 0  Reassembly (bytes): 0
Held Fragments/Packets: 0/0
Bytes 0/0 Packets 0/0 Resets 0/0 RNRs 0/0 REJs 0/0 INTs 0/0
--More--
LCI: 4095,  State: P4,  Interface: Ethernet1
Started 0:23:01, last input never, output never
Connected to CMNS [36.3030.3030.3030.30] <--> 0000.0c01.487d
via Serial1 LCN 1to 313131
Window size input: 6, output: 6
Packet size input: 1024, output: 1024
PS: 0  PR: 0  ACK: 0  Remote PR: 0  RCNT: 0  RNR: FALSE
Retransmits: 0  Timer (secs): 0  Reassembly (bytes): 0
Held Fragments/Packets: 0/0\
Bytes 0/0 Packets 0/0 Resets 0/0 RNRs 0/0 REJs 0/0 INTs 0/0
```

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

Table 1-10   Show X25 VC with CMNS Field Descriptions

Field	Description
LCI	Virtual circuit number; range is 1 to 4095.
State	State of the virtual circuit (which is independent of the states of other virtual circuits); P4 indicates the interface is in the data transfer state (See the CCITT X.25 recommendation for a description of virtual circuit states.)
Interface	Interface used for the virtual circuit. With CMNS, this can indicate Ethernet, Token Ring, and FDDI interfaces, as well as Serial.
Connected to	NSAP address, in brackets, for the device at the indicated X.121 address.
Ethernet1	Logical Channel Number (LCN) used (1 to 4095) and the MAC address of the node to which the interface is connected.

x25 accept-reverse

To instruct the protocol translator to accept all reverse charge calls, use the **x25 accept-reverse** interface configuration command. To disable this facility, use the **no** form of this command.

x25 accept-reverse
no x25 accept-reverse

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This command causes the interface to accept reverse charge calls by default. This behavior also can be configured on a per-peer basis using the **x25 map** interface configuration command.

Example

The following example illustrates how to set acceptance of reverse charge calls:

```
interface serial 0
x25 accept-reverse
```

Related Command

x25 map

x25 address

To set the X.121 address of a particular network interface, use the **x25 address** interface configuration command.

> **x25 address** *X.121-address*

Syntax Description

X.121-address	Variable-length X.121 address. The address is assigned by the X.25 network service provider.

Default

None

Command Mode

Interface configuration

Usage Guidelines

The X.121 must match that assigned by the X.25 network service provider.

Example

The following example sets the X.121 address for the interface:

```
interface serial 0
x25 address 00000123005
```

x25 bfe-decision

To direct how a protocol translator configured for **x25 bfe-emergency decision** will participate in emergency mode, use the **x25 bfe-decision** interface configuration command.

x25 bfe-decision {**no** | **yes** | **ask**}

Syntax Description

no	Prevents the protocol translator from participating in emergency mode and from sending address translation information to the BFE device.
yes	Allows the protocol translator to participate in emergency mode and to send address translation information to the BFE when the BFE enters emergency mode. The protocol translator obtains this information from the table created by the **x25 remote-red** command.
ask	Configures the protocol translator to display an onscreen request to enter the **bfe** EXEC command.

Default

no

Command Mode

Interface configuration

Example

The following example shows how to configure interface serial 0 to require an EXEC command from the administrator before it participates in emergency mode. The host IP address is 21.0.0.12, and the address of the remote BFE unit is 21.0.0.1. When the BFE enters emergency mode, the protocol translator will prompt the administrator for the EXEC command **bfe enter** to direct the protocol translator to participate in emergency mode.

```
interface serial 0
x25 bfe-emergency decision
x25 remote-red 21.0.0.12 remote-black 21.0.0.1
x25 bfe-decision ask
```

Related Commands

bfe
x25 bfe-emergency
x25 remote-red

x25 bfe-emergency

To configure the circumstances under which the protocol translator participate in emergency mode, use the **x25 bfe-emergency** interface configuration command.

> **x25 bfe-emergency** {**never** | **always** | **decision**}

Syntax Description

never	Prevents the protocol translator from sending address translation information to the BFE. If it does not receive address translation information, the BFE cannot open a new connection for which it does not know the address.
always	Allows the protocol translator to pass address translations to the BFE when it enters emergency mode and an address translation table has been created.
decision	Directs the protocol translator to wait until it receives a diagnostic packet from the BFE device indicating that the emergency mode window is open. The window is only open when a condition exists that allows the BFE is to enter emergency mode. When the diagnostic packet is received, the protocol translator's participation in emergency mode depends on how it is configured using the **x25 bfe-decision** command.

Default
never

Command Mode
Interface configuration

Example
The following example shows how to configure interface serial 0 to require an EXEC command from the administrator before it participates in emergency mode. The host IP address is 21.0.0.12, and the address of the remote BFE unit is 21.0.0.1. When the BFE enters emergency mode, the protocol translator will prompt the administrator for the EXEC command **bfe enter** to direct the protocol translator to participate in emergency mode.

```
interface serial 0
x25 bfe-emergency decision
x25 remote-red 21.0.0.12 remote-black 21.0.0.1
x25 bfe-decision ask
```

Related Commands
bfe
x25 bfe-decision

x25 default

To set a default protocol, use the **x25 default** interface configuration command. To remove the specified protocol, use the **no** form of this command.

> **x25 default** *protocol*
> **no x25 default** *protocol*

Syntax Description

protocol	Name of a protocol. It can be **ip** (for IP) or **pad** (for the X.3 PAD protocol).

Default

None

Command Mode

Interface configuration

Usage Guidelines

The **x25 default** command specifies the protocol that the protocol translator assumes to be in use when it receives incoming calls with an unknown Call User Data. If you do not use the **x25 default** command, the protocol translator clears any incoming calls with unknown Call User Data.

Example

The following example illustrates how to establish IP as the default protocol for X.25 calls.

```
interface serial 0
x25 default IP
```

Related Command

x25 map

x25 facility

To override the default facility settings on a per-call basis for calls originated by the protocol translator, use the **x25 facility** interface configuration command. To disable this facility, use the **no** form of this command.

> **x25 facility** *facility-keyword value*
> **no x25 facility** *facility-keyword value*

Syntax Description

facility-keyword	User facility.
value	Facility value. Table 1-11 lists possible values.

Default

No facility sent

Command Mode

Interface configuration

Usage Guidelines

Table 1-11 lists the supported facilities and their values.

Table 1-11   X.25 User Facilities

Option	Description
cug *number*	Number of a closed user group (CUG). It can be a value from 1 to 99 CUGs are allowed. CUGs can be used by a public data network to create a virtual private network within the larger network and to restrict access.
packetsize *in-size out-size*	Input and output packet sizes for flow control parameter negotiation. Both arguments must be one of the following values: 16, 32, 64, 128, 256, 512, 1024, 2048, or 4096.
windowsize *in-size out-size*	Packet count for input and output windows for flow control parameter negotiation. Both arguments must be in the range 1 to 127 and must not be greater than the value set for the **x25 modulo** command.
reverse	Indicates that charges on all calls originated by the interface should be reversed.
throughput *in out*	Requested throughput class negotiation values for input and output throughput across the network. Both arguments can have a value in the range 70 to 48,000 bps.
t**ransit-delay** *value*	Network transit delay for the duration of outgoing calls for networks that support transit delay. The transit delay value can be between 0 and 65334 milliseconds.
rpoa *name*	Name of a list of transit Recognized Private Operation Agencies (RPOAs) as defined by the **x25 rpoa** command to use in outgoing Call Request packets.

Examples

The following example illustrates how to specify a transit delay value in an X.25 configuration:

```
interface serial 0
x25 facility transit-delay 24000
```

The following example illustrates how to set an RPOA name and then send the list via the X.25 user facilities:

```
x25 rpoa green_list 23 35 36
interface serial 0
x25 facility rpoa green_list
x25 map ip 131.108.170.26 10 rpoa green_list
```

x25 hic

To set the highest incoming-only virtual circuit number, use the **x25 hic** interface configuration command.

>   **x25 hic** *circuit-number*

Syntax Description

circuit-number	Virtual circuit number. It can be a value from 0 through 4095. The default is 0, which indicates that there is no incoming-only virtual circuit range.

Default

0

Command Mode

Interface configuration

Example

The following example illustrates how to set a valid incoming-only virtual circuit range of 1 to 5:

```
interface serial 0
x25 lic 1
x25 hic 5
x25 ltc 6
```

Related Command

x25 lic

x25 hoc

To set the highest outgoing-only virtual circuit number, use the **x25 hoc** interface configuration command.

x25 hoc *circuit-number*

Syntax Description

circuit-number	Virtual circuit number from 1 through 4095, or 0 if there is no outgoing-only virtual circuit range.

Default

0

Command Mode

Interface configuration

Example

The following example illustrates how to set a valid outgoing-only virtual circuit range of 2000 to 2005:

```
interface serial 0
x25 loc 2000
x25 hoc 2005
```

Related Command

x25 loc

x25 hold-queue

To modify the maximum number of packets that can be held until a virtual circuit is able to transmit, use the **x25 hold-queue** interface configuration command. To remove this command from the configuration file and restore the default value, use the **no** form of this command.

x25 hold-queue *queue-size*
no x25 hold-queue [*queue-size*]

Syntax Description

queue-size   Number of packets in the hold queue. A value of 0 allows an unlimited number of packets in the hold queue.

Default

10 packets

Command Mode

Interface configuration

Example

The following example illustrates how to set the X.25 hold queue to hold 25 packets.

```
interface serial 0
x25 hold-queue 25
```

x25 hold-vc-timer

To prevent overruns on some X.25 switches caused by Call Request packets, use the **x25 hold-vc-timer** interface configuration command. To restore the default value for the timer, use the **no** form of this command.

> **x25 hold-vc-timer** *minutes*
> **no x25 hold-vc-timer**

Syntax Description

minutes	Number of minutes to prevent calls from going to a previously failed destination. Incoming calls still will be accepted.

Default

0

Command Mode

Interface configuration

Usage Guidelines

The **x25 hold-vc-timer** command uses the Ignore Destination timer to prevent additional calls to a destination for a given period of time.

Only CALLs the protocol translator originates will be held down; routed X.25 CALLs are not affected by this parameter.

Upon receiving a Clear Request for an outstanding Call Request, the X.25 support code immediately tries another Call Request if it has more traffic to send, and this action might cause overrun problems.

Example

The following example illustrates how to set the Ignore Destination timer to three minutes:

```
interface serial 0
x25 hold-vc-timer 3
```

x25 host

To define a static host name-to-address mapping, use the **x25 host** global configuration command. To remove the host name, use the **no** form of this command.

> **x25 host** *name X.121-address* [**cud** *call-user-data*]
> **no x25 host** *name*

Syntax Description

name	Host name.
X.121-address	X.121 address.
cud *call-user-data*	(Optional.) Sets the Call User Data (CUD) field in the X.25 Call Request packet.

Default

None

Command Mode

Global configuration

Examples

The following example illustrates how to specify a static address mapping:

```
x25 host Willard 4085551212
```

The following example illustrates how to remove a static address mapping:

```
no x25 host Willard
```

x25 htc

To set the highest two-way virtual circuit number, use the **x25 htc** interface configuration.

x25 htc *circuit-number*

Syntax Description

circuit-number	Virtual circuit number. It can be a value from 0 to 4095. A value of 0 indicates there is no two-way virtual circuit range.

Default

1024 for X.25 network service interfaces; 4095 for CMNS network service interfaces.

Command Mode

Interface configuration

Example

The following example illustrates how to set a valid two-way virtual circuit range of 5 to 25:

```
interface serial 0
x25 ltc 5
x25 htc 25
```

Related Command

cmns enable
x25 ltc

x25 idle

The protocol translator can clear a switched virtual circuit (SVC) after a period of inactivity. To set this period, use the **x25 idle** interface configuration command.

x25 idle *minutes*

Syntax Description

minutes	Number of minutes in the idle period. The default is 0, which causes the protocol translator to keep the SVC open indefinitely.

Default

0

Command Mode

Interface configuration

Usage Guidelines

Both calls originated and terminated by the protocol translator are cleared; switched virtual circuits are not cleared. To clear one or all virtual circuits at once, use the privileged EXEC command **clear x25-vc**.

Example

The following example illustrates how to set a five minute wait period before an idle circuit is cleared:

```
interface serial 2
x25 idle 5
```

Related Command

clear x25-vc

x25 ip-precedence

To allow a new virtual circuit to open based on the IP precedence value, use the **x25 ip-precedence** interface configuration command. To ignore the precedence value when opening virtual circuits, use the **no** form of this command.

> **x25 ip-precedence**
> **no x25 ip-precedence**

Syntax Description

This command has no arguments or keywords.

Default

The protocol translators open one virtual circuit for all types of service.

Command Mode

Interface configuration

Usage Guidelines

Note that some hosts send nonstandard data in the precedence field, causing multiple, wasteful virtual circuits to be created.

Example

The following example illustrates how to allow new virtual circuits based on the TOS field:

```
interface serial 3
x25 ip-precedence
```

x25 ips

To set the interface default input packet size to match that of the network, use the **x25 ips** interface configuration command.

x25 ips *bytes*

Syntax Description

bytes	Input packet size in bytes. It can be one of the following values: 16, 32, 64, 128, 256, 512, 1024, 2048, or 4096.

Default

128 bytes

Command Mode

Interface configuration

Usage Guidelines

X.25 network connections have a default maximum input packet sizes set by the network administrator. Larger packet sizes require less overhead processing. To send a packet larger than the X.25 packet size over an X.25 virtual circuit, a protocol translator must break the packet into two or more X.25 packets with the M-bit ("more data" bit) set. The receiving device collects all packets with the M-bit set and reassembles them.

Note   Set the **x25 ips** and **x25 ops** commands to the same value unless your network supports asymmetrical input and output packet sizes.

Example

The following example shows how to set the default maximum packet sizes to 512:

```
interface serial 1
x25 ips 512
x25 ops 512
```

Related Command

x25 ops

x25 lic

To set the lowest incoming-only virtual circuit number, use the **x25 lic** interface configuration command.

x25 lic *circuit-number*

Syntax Description

circuit-number	Virtual circuit number. It can be a value from 0 to 4095.The default is 0, which means that there is no incoming-only virtual circuit range.

Default
0

Command Mode
Interface configuration

Example
The following example shows how to set a valid incoming-only virtual circuit range of 1 to 5:

```
interface serial 0
x25 lic 1
x25 hic 5
x25 ltc 6
```

Related Command
x25 hic

x25 linkrestart

To force a packet-level restart when the link level resets, use the **x25 linkrestart** interface configuration command. To disable this facility, use the no form of this command.

> **x25 linkrestart**
> **no x25 linkrestart**

Syntax Description

This command has no arguments or keywords.

Default

Forcing packet-level restarts is the default and is necessary for networks that expect this behavior.

Command Mode

Interface configuration

Usage Guidelines

The **x25 linkrestart** command restarts X.25 Level 3 when errors occur in Level 2 (LAPB).

Example

The following example illustrates how to disable the link level restart.

```
interface serial 3
no x25 linkrestart
```

x25 loc

To set the lowest outgoing-only virtual circuit number, use the **x25 loc** interface configuration command.

x25 loc *circuit-number*

Syntax Description

circuit-number	Virtual circuit number. It can be a value from 0 to 4095.The default is 0, which means that there is no outgoing-only virtual circuit range.

Default

0

Command Mode

Interface configuration

Example

The following example illustrates how to set a valid outgoing-only virtual circuit range of 2000 to 2005:

```
interface serial 0
x25 loc 2000
x25 hoc 2005
```

Related Command

x25 hoc

x25 ltc

To set the lowest two-way virtual circuit number, use the **x25 ltc** interface configuration command.

x25 ltc *circuit-number*

Syntax Description

circuit-number	Virtual circuit number. It can be a value from 0 to 4095. A value of 0 means that there is no two-way virtual circuit range.

Default

1

Command Mode

Interface configuration

Example

The following example illustrates how to set a valid two-way virtual circuit range of 5 to 25:

```
interface serial 0
x25 ltc 5
x25 htc 25
```

Related Command

x25 htc

x25 map

To set up the LAN protocol-to-X.121 address mapping for the host, use the **x25 map** interface configuration command. To retract a network protocol-to-X.121 mapping, use the **no** form of this command.

>   **x25 map** *protocol-keyword protocol-address X.121-address* [*option1... [option6]]*
>   **no x25 map** *protocol-keyword protocol-address X.121-address*

Syntax Description

protocol-keyword	Protocol type. Supported protocol keywords are listed in Table 1-12.
protocol-address	Protocol address.
X.121-address	X.121 address. Both addresses specify the network protocol-to-X.121 mapping.
option 1 ... option 6	(Optional.) Provides additional functionality or the X.25 essential user facilities to the mapping specified. You can specify up to six of the options listed in Table 1-13.

Default

None

Command Mode

Interface configuration

Usage Guidelines

The **broadcast** keyword simplifies the configuration of OSPF for nonbroadcast networks that will use X.25.

Because no defined protocol can dynamically determine such mappings, you must enter a mapping for each host with which the protocol translator will exchange traffic.

Table 1-12 lists the protocols that you can specify for the *protocol-keyword* argument. Table 1-13 lists the options you can specify for the *option* argument.

OSPF treats a nonbroadcast, multiaccess network such as X.25 much the same way it treats a broadcast network in that it requires selection of a designated protocol translator. In previous releases, this required manual assignment in the OSPF configuration using the **neighbor interface** router configuration command. When the **x25 map** command is included in the configuration with the **broadcast** keyword, there is no need to configure any neighbors manually. OSPF will now automatically run over the Frame Relay network as a broadcast network.

Note   The OSPF broadcast mechanism assumes that IP class D addresses are never used for regular traffic over X.25.

Table 1-12   Protocols Supported by X.25

Keyword	Protocol
ip	IP
novell	Novell IPX
compressedtcp	TCP header compression

Table 1-13   X.25 Map Options

Option	Description
reverse	Specifies reverse charging for outgoing calls.
accept-reverse	Accept incoming reverse-charged calls. If this option is not present, the protocol translator clears reverse charge calls.
broadcast	Direct any broadcasts sent through this interface to the specified X.121 address. This option simplifies the configuration of OSPF.
cug *number*	Closed user group number (from 1 to 99) for the mapping in the outgoing call.
nvc *count*	Sets the number of virtual circuits (VCs) for this map/host. The default *count* is the **x25 nvc** setting of the interface. A maximum number of eight VCs can be configured for each single map/host.
packetsize *in-size out-size*	Input and output packet sizes for the PVC. Both arguments must be one of the following values: 16, 32, 64, 128, 256, 512, 1024, 2048, or 4096.
windowsize *in-size out-size*	Packet count for input and output windows for the mapping in an outgoing call. Both arguments should be the same, must be in the range 1 to 127, and must not be greater than the value set for the **x25 modulo** command.
throughput *in out*	Requested throughput class values for input and output (throughput across the network. Values for both arguments can be in the range 70 to 48,000 bps.
transit-delay *number*	Transit delay value in milliseconds (0 to 65334) for the mapping of outgoing calls, for networks that support transit delay.
nuid *username password*	Sends a network ID facility in the outgoing call with the specified TACACS username and password (format determined by Cisco).
nudata *string*	Network user identification in a format determined by the network administrator (as per CCITT recommendation). The string must be enclosed in quotation marks ("") if there are any spaces present.
rpoa *name*	Name defined by the **x25 rpoa** command for a list of transit RPOAs to use in outgoing Call Request packets.

Note   You cannot configure these options with the **x25 map cmns** version of the **x25 map** command.

Examples

The following example illustrates how to map IP address 131.08.2.5 to X.121 address 000000010300. The **broadcast** keyword directs any broadcasts sent through this interface to the specified X.121 address.

```
interface serial 0
x25 map ip 131.08.2.5 000000010300 broadcast
```

The following example illustrates how to set an RPOA name for use in the connection:

```
x25 rpoa green_list 23 35 36
interface serial 0
x25 facility rpoa green_list
x25 map ip 131.108.170.26 10 rpoa green_list
```

The following example shows how to add a network user identifier (NUI) to the address map:

```
interface serial 0
x25 map IP 131.108.174.32 2 nudata "Network User ID 35"
```

Strings can be quoted, but quotes are not required unless embedded blanks are present.

Related Commands
x25 accept-reverse
x25 map cmns
x25 map compressedtcp
x25 map default
x25 map pvc (encapsulated)

x25 map cmns

To map NSAP addresses to either MAC-layer addresses or X.121 addresses after enabling CMNS on a nonserial interface, use the **x25 map cmns** interface configuration command. To retract a mapping, use the **no** form of this command.

x25 map cmns *NSAP MAC-address*
no x25 map cmns *NSAP MAC-address*

x25 map cmns *NSAP* [*X.121-address*]
no x25 map cmns *NSAP* [*X.121-address*]

Syntax Description

NSAP	NSAP address. It can be either the actual DTE NSAP address or the NSAP prefix of the NSAP address. The NSAP prefix is sufficient for a best match to route a call.
MAC-address	MAC-level address.
X.121-address	X.121 address.

Default

None

Command Mode

Interface configuration

Usage Guidelines

The address arguments specify the NSAP address-to-MAC address or NSAP address-to-X.121 address mappings.

Example

The following example shows how to switch traffic intended for any NSAP address with prefix 38.8261.17 to X.121 address 4085551234 over interface Ethernet 0:

```
interface ethernet 0
cmns enable
x25 map cmns 38.8261.17 4085551234
```

Related Command

cmns enable
x25 map

x25 map compressedtcp

To map compressed TCP traffic to X.121 addresses, use the **x25 map compressedtcp** interface configuration command. To disable TCP header compression for the link, use the **no** form of this command.

> **x25 map compressedtcp** *IP-address X.121-address* [*options*]
> **no x25 map compressedtcp** *IP-address X.121-address*

Syntax Description

IP-address	IP address.
X.121-address	X.121 address.
options	(Optional.) The same options as those for the **x25 map** command. See Table 1-13.

Default

None

Command Mode

Interface configuration

Usage Guidelines

The Call User Data of compressed TCP calls is the single byte 0xD8.

TCP header compression is supported over X.25 links. The implementation of compressed TCP over X.25 uses a virtual circuit (VC) to pass the compressed packets. The noncompressed packets use another VC. The NVC map option cannot be used for TCP header compression, as only one virtual circuit can carry compressed TCP header traffic to a given host.

Example

The following example establishes packet compression on interface serial 4:

```
interface serial 4
ip tcp header-compression
x25 map compressedtcp 131.08.2.5 000000010300
```

Related Commands

ip tcp header-compression
x25 map

x25 modulo

To set the window modulus (define the extended packet sequence), use the **x25 modulo** interface configuration command.

> **x25 modulo** *modulus*

Syntax Description

modulus	Size of the modulus. It can be either 8 or 128.

Default

8

Command Mode

Interface configuration

Usage Guidelines

The value of the modulo parameter must agree with that of the device on the other end of the X.25 link. X.25 supports flow control with a sliding window sequence count. The window counter restarts at zero upon reaching the upper limit, which is called the *window modulus*.

Example

The following example illustrates how to set the window modulus to 128:

```
interface serial 0
x25 modulo 128
```

Related Commands

x25 win
x25 wout

x25 nvc

To specify the maximum number of switched virtual circuits that a protocol can have open simultaneously to one host, use the **x25 nvc** interface configuration command.

x25 nvc *count*

Syntax Description

count	Circuit count. It can be a value from 1 to 8. The default is 1. You can configure a maximum of 8 VCs for each protocol/host pair. Protocols that do not tolerate out-of-order delivery, such as encapsulated TCP header compression, will only use one virtual circuit despite this value.

Default

1

Command Mode

Interface configuration

Usage Guidelines

To increase throughput across networks, you can establish up to eight switched virtual circuits to a host.

When the windows and output queues of all existing connections to a host are full, a new virtual circuit will be opened to the designated circuit count. If a new connection cannot be opened, the data is dropped.

Note The *count* value specified for **x25 nvc** affects the default value for the number of SVCs. It does not affect the NVC value for any **x25 map** commands that already have been configured.

Example

The following example illustrates how to set the maximum number of switched virtual circuits that can be open simultaneously to 4:

```
interface serial 0
x25 nvc 4
```

x25 ops

To set the interface default output packet size to match those of the network, use the **x25 ops** interface configuration command.

x25 ops *bytes*

Syntax Description

bytes	Output packet size in bytes. It can be a number in the range. 16 through 1024.

Default

128 bytes

Command Mode

Interface configuration

Usage Guidelines

X.25 networks use maximum input packet sizes set by the network administration. Larger packet sizes are better because smaller packets require more overhead processing. To send a packet larger than the X.25 packet size over an X.25 virtual circuit, a protocol translator must break the packet into two or more X.25 packets with the M-bit ("more data" bit) set. The receiving device collects all packets with the M-bit set and reassembles them.

For optimal throughput, all X.25 interfaces that may carry routed traffic through an IP network should be configured with the same default packet sizes and window sizes.

Note Set the **x25 ips** and **x25 ops** commands to the same value unless your network supports asymmetry between input and output packets.

Example

The following example shows how to set the default maximum packet sizes to 512:

```
interface serial 1
x25 ips 512
x25 ops 512
```

Related Command

x25 ips

x25 pvc (encapsulating)

To establish an encapsulation permanent virtual circuit (PVC), use the **x25 pvc** interface configuration command. To delete the PVC, use the **no** form of this command.

> **x25 pvc** *circuit protocol-keyword protocol-address* [*option*s]
> **no x25 pvc** *circuit protocol-keyword protocol-address*

Syntax Description

circuit	Virtual-circuit channel number and must be less than the virtual circuits assigned to the switched virtual circuits (SVCs).
protocol-keyword	Protocol type. Supported protocols are listed in Table 1-14.
protocol-address	Address of the host at the other end of the PVC.
options	(Optional.) PVC's flow control parameters if they differ from the interface defaults. The *option* arguments add certain features to the mapping specified and can be any of the options listed in Table 1-15.

Default

None

Command Mode

Interface configuration

Usage Guidelines

PVCs are not supported for ISO CMNS. Switched Virtual Circuits (SVCs) are sufficient for CMNS connections over X.25.

You must specify the required network protocol-to-X.121 address mapping with an **x25 map** interface configuration command before you can set up a PVC.

Table 1-14 lists the protocol types you can specify for the *protocol-keyword* argument. Table 1-15 lists the options you can specify for the *option* argument.

Note   When configuring X.25 to use a PVC, you must ensure that no traffic is sent toward a remote terminal server between the time the **x25 map** command is issued and the time that **x25 pvc** command is issued. Otherwise, the local system will create a switched virtual circuit (SVC), and then the **x25 pvc** command will not be allowed.

Note   Map entries with the **broadcast** attribute are particularly likely to get traffic due to routing protocol traffic. The simplest way to ensure that traffic is not sent while configuring an interface to use a PVC is to shut down the interface while configuring it for PVC support.

Table 1-14   Protocols Supported by X.25 PVCs

Keyword	Protocol
ip	IP
ipx or **novell**	Novell IPX
compressedtcp	TCP header compression

Table 1-15   PVC Options

Option	Description
packetsize *in-size out-size*	Input and output packet sizes for the PVC. Both arguments must be one of the following values: 16, 32, 64, 128, 256, 512, 1024, 2048, or 4096.
windowsize *in-size out-size*	Packet counts for input and output windows for the PVC. Both arguments should be the same, must be in the range 1 to 127, and must not be greater than the value set for the **x25 modulo** command.

Example

The following example shows how to establish a PVC on a channel with a VINES host attached:

```
interface serial 0
x25 pvc 2 vines 60002A2D:0001
```

Related Command

x25 map

x25 pvc (switched)

To configure a switched permanent virtual circuit (PVC) for a given interface, use the following variation of the **x25 pvc** interface configuration command:

   x25 pvc *pvc-number1* **interface** *interface-name* **pvc** *pvc-number2* [*option*]

Syntax Description

pvc-number1	PVC number that will be used on the local interface (as defined by the primary interface command).
interface *interface-name*	Remote interface type and unit number (serial 0, for example).
pvc *pvc-number2*	PVC number that will be used on the remote interface.
option	(Optional.) Adds features to the mapping specified. It can be one of the options listed in Table 1-16.

Default

None

Command Mode

Interface configuration

Usage Guidelines

You can configure X.25 permanent virtual circuits (PVCs) in the X.25 switching software. This means that DTEs that require permanent circuits can be connected to the protocol translator acting as an X.25 switch and have a properly functioning connection. X.25 RESETs will be sent to indicate when the circuit comes up or goes down.

The **x25 pvc** command supports remote switching between two routers.

Table 1-16 lists the options you can specify for the *options* argument.

Table 1-16   Switched PVC Options

Option	Description
packetsize *in-size out-size*	Input and output packet sizes for the PVC. Both arguments must be one of the following values: 16, 32, 64, 128, 256, 512, 1024, 2048, or 4096.
windowsize *in-size out-size*	Packet count for input and output windows for the PVC. Both arguments should be the same, must be in the range 1 to 127, and must not be greater than the value set for the **x25 modulo** command.

Example

The following example configures a PVC connected between two serial interfaces on the same protocol translator. In this type of interconnection configuration, the alternate interface must be specified along with the PVC number on that interface. To make a working PVC connection, two commands must be specified, each pointing to the other as this example illustrates:

```
interface serial 0
encapsulation x25
x25 ltc 5
x25 pvc 1 interface serial 1 pvc 1
interface serial 1
encapsulation x25
x25 ltc 5
x25 pvc 1 interface serial 0 pvc 1
```

x25 pvc (tunnel)

To connect two PVCs across a TCP/IP LAN, use the following variation of the **x25 pvc** interface configuration command.

x25 pvc *pvc-number1* **tunnel** *IP-address* **interface serial** *string* **pvc** *pvc-number2* [*option*s]

Syntax Description

pvc-number1	PVC number of the connecting device.
tunnel *IP-address*	IP address of the router to which you are connecting.
interface serial *string*	Serial interface specification that accepts either a number or a string in model 7000 format (number/number) to denote the serial interface.
pvc *pvc-number2*	Remote PVC number of the target device.
options	(Optional.) Adds certain features for the connection; can be either option listed in Table 1-17.

Default

None; the PVC window and packet sizes default to the interface default values.

Command Mode

Interface configuration

Usage Guidelines

Use the PVC tunnel commands to tell the protocol translator to what the far end of the PVC is connected. The incoming and outgoing packet sizes and window sizes must match the remote PVC outgoing and incoming sizes.

See the section "LAPB and X.25 Configuration Examples" in the *Communication Server Configuration Guide* for complete configuration examples.

Table 1-17 lists the options you can specify for the *options* argument.

Table 1-17   X.25 PVC Tunnel Options

Option	Description
packetsize *in-size out-size*	Input and output packet sizes for the PVC. Both arguments must be one of the following values: 16, 32, 64, 128, 256, 512, 1024, 2048, or 4096.
windowsize *in-size out-size*	Packet count for input and output windows for the PVC. Both arguments should be the same, must be in the range 1 to 127, and must not be greater than the value set for the **x25 modulo** command.

x25 remote-red

To set up the table that lists the BFE nodes (host or gateways) to which the protocol translator will send packets, use **x25 remote-red** interface configuration command.

> **x25 remote-red** *host-IP-address* **remote-black** *Blacker-IP-address*

Syntax Description

host-IP-address	IP address of the host or a protocol translator that the packets are being sent to.
remote-black	Delimits the addresses for the table being built.
Blacker-IP-address	IP address of the remote BFE device in front of the host to which the packet is being sent.

Default

None

Command Mode

Interface configuration

Usage Guidelines

The table that results from this command provides the address translation information the protocol translator sends to the BFE when it is in emergency mode.

Example

The following example sets up a short table of BFE nodes for interface serial 0:

```
interface serial 0
x25 remote-red 131.108.9.3 remote-black 131.108.9.13
x25 remote-red 192.108.15.1 remote-black 192.108.15.26
```

Related Command

x25 bfe-decision

x25 route

To create an entry in the X.25 routing table, use the **x25 route** global configuration command. To remove an entry from the table, use the **no** form of this command.

x25 route [# *position*] *X.121-address* [**cud** *pattern*] **interface** *interface number*
no x25 route [# *position*] *X.121-address* [**cud** *pattern*] **interface** *interface number*

x25 route [# *position*] *X.121-address* [**cud** *pattern*] **ip** *IP-address* [*IP-address2 ... IP-address6*]
no x25 route [# *position*] *X.121-address* [**cud** *pattern*] **ip** *IP-address*

x25 route [# *position*] *X.121-address* [**cud** *pattern*] **alias** *interface number*
no x25 route [# *position*] *X.121-address* [**cud** *pattern*] **alias** *interface number*

x25 route [# *position*] *X.121-address* [**substitute-source** *rewrite-pattern*]
    [**substitute-dest** *rewrite-pattern*] [**cud** *pattern*] **interface** *interface number*
no x25 route [# *position*] *X.121-pattern* [**substitute-source** *rewrite-pattern*]
    [**substitute-dest** *rewrite-pattern*] [**cud** *pattern*] **interface** *interface number*

Note For typographical reasons, the last two commands are shown on two lines. When using the optional keywords in this variation of the **x25 route** subcommand, the **substitute-source** keyword must precede the **substitute-dest** keyword, and both must precede the **cud** keyword. The entire command must be on one line.

Syntax Description

# *position*	(Optional.) A pound sign (#) followed by a number to designate a positional parameter after which to insert or delete the new entry in an existing entry. If no *position* parameter is given, the entry is appended to the end of the routing table.
X.121-address	The called X.121 address. This argument can be either an actual X.121 destination address or a regular expression such as 1111*, representing a group of X.121 addresses.
cud *pattern*	(Optional.) Call User Data corresponding to the X.121 address which can be specified as a printable ASCII string. The Call User Data field is matched against the data that follows the protocol identification field, which is 4 bytes.
interface *interface number*	Specifies the interface type followed by the unit or port number, for example, interface Ethernet 0.
ip *IP-address*	Specifies an IP address of the network interface or DTE for connections routed through a LAN. Optionally, up to six IP addresses can be listed and each in turn will be tried in the event that the first address fails, thus allowing alternate routes and decreasing the likelihood of failure.

alias *interface number*	Configures an alias route. Specify the interface type followed by the unit or port number of the destination interface on the destination protocol translator. An alias route is valid only for calls that come in on the named interface. An alias accepts calls for the protocol translator. When calls come in on the specified interface and its destination address fits the X.121 pattern, the call is received on the destination interface.
substitute-source *rewrite-pattern*	(Optional.) Specifies the calling X.121 address to replace in routed X.25 packets. The backslash (\) character is treated specially in the argument *rewrite-pattern*; it indicates that the digit immediately following it selects a portion of the original called address to be inserted in the new called address. The characters \0 are replaced with the entire original address. The characters \1 through \9 are replaced with the strings that matched the first through ninth parenthesized parts of *X.121-pattern*. See Table 1-18 and Table 1-19 for summaries of pattern and character matching, respectively.
substitute-dest *rewrite-pattern*	(Optional.) Specifies the called X.121 address to replace in routed X.25 packets. (For backwards compatibility, the **substitute** keyword will be accepted as **substitute-dest** and written to nonvolatile memory in the new format.) The backslash (\) character is treated specially in the argument *rewrite-pattern*; it indicates that the digit immediately following it selects a portion of the original called address to be inserted in the new called address. The characters \0 are replaced with the entire original address. The characters \1 through \9 are replaced with the strings that matched the first through ninth parenthesized parts of *X.121-pattern*. See Table 1-18 and Table 1-19 for summaries of pattern and character matching, respectively.

Default

None

Command Mode

Global

Usage Guidelines

The X.25 routing table is consulted when an incoming call is received that should be forwarded to its destination. Two fields are used to determine the route: the called X.121 network interface address or the destination host address, and the X.25 packet's Called User Data (CUD) field. When the destination address and the CUD of the incoming packet fit the X.121 and CUD patterns in the routing table, the packet is forwarded.

The order in which X.25 routing table entries are specified is significant; the list is scanned for the first match. The optional argument *# position* (# followed by an integer) designates the line number of an existing entry. The new entry is inserted after the existing entry indicated by the *position* argument. If no *position* parameter is given, the entry is appended to the end of the routing table.

The argument *X.121-pattern* can be either an actual X.121 destination address or a regular expression representing a group of X.121 addresses (for example, 1111.*).

The optional Call User Data pattern also can be specified as a printable ASCII string. Both the X.121 address and Call User Data can be written using UNIX-style, regular expressions. The Call User Data field is matched against the data that follows the protocol identification field, which is four bytes. See Table 1-18 and Table 1-19 for summaries of pattern and character matching. A more complete description of the pattern-matching characters is found in Appendix D.

Note that address substitution is only performed on routes to an interface. When running X.25 over IP, address substitution can be performed on the destination IP system if the destination system is configured with the appropriate X.25 routing commands.

Use the **show x25 route** command to display the X.25 routing table. The interface routes will show up after any routes used for translation commands. Because the interface routes are expected to be less specific, they should come last. This is done automatically.

Table 1-18   Pattern Matching

Pattern	Description
\0	Replaces the entire original address.
\1...9	Replaces strings that match the first through ninth parenthesized part of the X.121 address.
*	Matches 0 or more sequences of the regular expressions.
+	Matches 1 or more sequences of the regular expressions.
?	Matches the regular expression of the null string.

Table 1-19   Character Matching

Character	Description
^	Matches the null string at the beginning of the input string.
$	Matches the null string at the end of the input string.
\char	Matches *char*.
.	Matches any single character.

Examples

The following example illustrates how to use regular expression pattern matching characters to match just the initial portion of the complete X.25 address:

```
x25 route ^3107 interface serial 0
```

In the following example, if a call comes in on interface serial 0 and matches any X.121-address pattern, the call will be accepted for the type of connectivity configured for the interface and the CUD.

```
x25 route .* alias serial 0
```

In the following example, the call will be accepted because both this VAX X.121 address and the address given in the **x25 address** interface command will be treated as local addresses for interface serial 0.

```
x25 route vax-X.121-address alias serial 0
```

The following example illustrates how to configure alternate IP addresses for the routing table. In the event the first address listed is not available, the next address is tried, and so on until a connection is made.

```
x25 route ^3106 ip 131.08.2.5  131.08.7.10  131.08.7.9
```

Related Command
show x25 route

x25 routing

To enable X.25 switching or tunneling, use the **x25 routing** global configuration command. To disable the forwarding of X.25 calls, use the **no** form of this command.

x25 routing
no x25 routing

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global

Usage Guidelines

The **x25 routing** command enables local switching or tunneling, which is used for remote switching to allow routing X.25 traffic through a LAN. X.25 calls will not be forwarded until this command is issued.

Example

The following example illustrates how to enable X.25 switching:

```
x25 routing
```

x25 rpoa

To set the packet network carrier, use the **x25 rpoa** global configuration command. To delete the name of the packet network carrier, use the **no** form of this command.

>**x25 rpoa** *name number...*
>**no x25 rpoa** *name*

Syntax Description

name	Recognized Private Operating Agency (RPOA), which must be unique with respect to all other RPOA names. It is used in the **x25 facility** and **x25 map** interface configuration commands.
number	Number that describes an RPOA. You can enter a list of up to 10 numbers.

Default

None

Command Mode

Global

Usage Guidelines

This command specifies a list of transit Recognized Private Operating Agencies (RPOAs) to use, referenced by name.

Example

The following example illustrates how to set an RPOA name and then send the list via the X.25 user facilities:

```
x25 rpoa green_list 23 35 36
interface serial 0
x25 facility rpoa green_list
x25 map ip 131.108.170.26 10 rpoa green_list
```

Related Commands

x25 facility
x25 map

x25 suppress-called-address

To omit the called (destination) X.121 address in outgoing calls, use the **x25 suppress-called-address** interface configuration command. To return to the default state, use the **no** form of this command.

>**x25 suppress-called-address**
>**no x25 suppress-called-address**

Syntax Description

This command has no arguments or keywords.

Default

The called address is sent by default.

Command Mode

Interface configuration

Usage Guidelines

This command omits the called (destination) X.121 address in Call Request packets and is required for networks that expect only subaddresses in the called address field.

Example

The following example illustrates how to suppress or omit the called address in Call Request packets:

```
interface serial 0
x25 suppress-called-address
```

x25 suppress-calling-address

To omit the calling (source) X.121 address in outgoing calls, use the **x25 suppress-calling-address** interface configuration command. To return to the default state, use the **no** form of this command.

> **x25 suppress-calling-address**
> **no x25 suppress-calling-address**

Syntax Description

This command has no arguments or keywords.

Default

The calling address is sent by default.

Command Mode

Interface configuration

Usage Guidelines

This command omits the calling (source) X.121 address in Call Request packets and is required for networks that expect only subaddresses in the calling address field.

Example

The following example illustrates how to suppress or omit the calling address in Call Request packets:

```
interface serial 0
x25 suppress-calling-address
```

x25 t10

To set the limit for the Restart Request retransmission timer (T10) on DCE devices, use the **x25 t10** interface configuration command.

x25 t10 *seconds*

Syntax Description

seconds	Amount of time in seconds.

Default

60 seconds

Command Mode

Interface configuration

Example

The following example sets the T10 timer to 30 seconds:

```
interface serial 0
x25 t10 30
```

x25 t11

To set the limit for the Call Request Completion timer (T11) on DCE devices, use the **x25 t11** interface configuration command.

x25 t11 *seconds*

Syntax Description

seconds	Amount of time in seconds

Default

180 seconds

Command Mode

Interface configuration

Example

The following example sets the T11 timer to 90 seconds:

```
interface serial 0
x25 t11 90
```

x25 t12

To set the limit for the Reset Request retransmission timer (T12) on DCE devices, use the **x25 t12** interface configuration command.

x25 t12 *seconds*

Syntax Description

seconds   Amount of time in seconds

Default

60 seconds

Command Mode

Interface configuration

Example

The following example sets the T12 timer to 30 seconds:

```
interface serial 0
x25 t12 30
```

x25 t13

To set the limit for the Clear Request retransmission timer (T13) on DCE devices, use the **x25 t13** interface configuration command.

x25 t13 *seconds*

Syntax Description

seconds   Amount of time in seconds

Default

60 seconds

Command Mode

Interface configuration

Example

The following example sets the T13 timer to 30 seconds:

```
interface serial 0
x25 t13 30
```

x25 t20

To set the limit for the Restart Request retransmission timer (T20) on DTE devices, use the **x25 t20** interface configuration command.

x25 t20 *seconds*

Syntax Description

seconds   Amount of time in seconds

Default

180 seconds

Command Mode

Interface configuration

Example

The following example sets the T20 timer to 90 seconds:

```
interface serial 0
x25 t20 90
```

x25 t21

To set the limit for the Call Request Completion timer (T21) on DTE devices, use the **x25 t21** interface configuration command.

x25 t21 *seconds*

Syntax Description

seconds   Amount of time in seconds

Default

200 seconds

Command Mode

Interface configuration

Example

The following example sets the T21 timer to 100 seconds:

```
interface serial 0
x25 t21 100
```

x25 t22

To set the limit for the Reset Request retransmission timer (T22) on DTE devices, use the **x25 t22** interface configuration command.

x25 t22 *seconds*

Syntax Description

seconds	Amount of time in seconds

Default
180 seconds

Command Mode
Interface configuration

Example
The following example sets the T22 timer to 90 seconds:

```
interface serial 0
x25 t22 90
```

x25 t23

To set the limit for the Clear Request retransmission timer (T23) on DTE devices, use the **x25 t23** interface configuration command.

x25 t23 *seconds*

Syntax Description

seconds   Amount of time in seconds

Default

180 seconds

Command Mode

Interface configuration

Example

The following example sets the T23 timer to 90 seconds:

```
interface serial 0
x25 t23 90
```

x25 th

To instruct the protocol translator to send acknowledgment packets when it is not busy sending other packets, even if the number of input packets has not reached the input window size count, use the **x25 th** interface configuration command.

x25 th *delay-count*

Syntax Description

delay-count	A value between 0 and the input window size. A value of 1 sends one Receiver Ready acknowledgment per packet at all times. The default is 0, which disables the delayed acknowledgment strategy.

Default

0

Command Mode

Interface configuration

Usage Guidelines

The **x25 th** command improves line responsiveness at the expense of bandwidth.

The protocol translator sends acknowledgment packets when the number of input packets reaches the count you specify, providing there are no other packets to send. For example, if you specify a count of 1, the protocol translator can send an acknowledgment per input packet.

Example

The following example sends five Receiver Ready acknowledgments per packet as an input packet delay:

```
interface serial 1
x25 th 5
```

Related Commands

x25 win
x25 wout

x25 use-source-address

To update the calling X.121 addresses on outgoing calls that are forwarded over a specific interface, use the **x25 use-source-address** interface configuration command. To prevent updating the source addresses of outgoing calls, use the **no** form of this command.

x25 use-source-address
no x25 use-source-address

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

Some X.25 calls, when forwarded by the X.25 switching support, need the calling (source) X.121 address updated to that of the outgoing interface. This is necessary when forwarding calls from private data networks to public data networks.

Example

The following example shows how to prevent updating the source addresses of outgoing X.25 calls on interface serial 0 once calls have been forwarded:

```
interface serial 0
no x25 use-source-address
```

x25 win

To change the default maximum number of unacknowledged incoming packets for the interface, use the **x25 win** interface configuration command.

x25 win *packets*

Syntax Description

packets	Maximum number of unacknowledged incoming packets. It can be a number in the range 1 to one less than the window modulus.

Default

2 packets

Command Mode

Interface configuration

Usage Guidelines

This command determines how many packets the protocol translator can receive before sending an X.25 acknowledgment. To maintain high bandwidth utilization, assign this limit the largest number that the network allows.

Note   Set **x25 win** and **x25 wout** to the same value unless your network supports asymmetry between input and output window sizes.

Example

The following example specifies that five packets must be received before sending an X.25 acknowledgment:

```
interface serial 1
x25 win 5
```

Related Commands

x25 modulo
x25 th
x25 wout

x25 wout

To change the default maximum number of unacknowledged packets to allow, use the **x25 wout** interface configuration command.

x25 wout *packets*

Syntax Description

packets	Maximum number of unacknowledged packets. It can be a number in the range 1 to the window modulus.

Default

2 packets

Command Mode

Interface configuration

Usage Guidelines

This command determines the default number of packets the protocol translator can send before waiting for an X.25 acknowledgment. To maintain high bandwidth utilization, assign this limit the largest number that the network allows.

Note   Set **x25 win** and **x25 wout** to the same value unless your network supports asymmetry between input and output window sizes.

Example

The following example specifies an upper limit of five for the number of outstanding unacknowledged packets for the output window:

```
interface serial 1
x25 wout 5
```

Related Commands

x25 modulo
x25 th
x25 win

x29 access-list

To limit access to the protocol translator from certain X.25 hosts, use the **x29 access-list** global configuration command. To delete an entire access list, use the **no** form of this command.

> **x29 access-list** *access-list-number* **{deny | permit}** *X.121-address*
> **no x29 access-list** *access-list-number*

Syntax Description

access-list-number	Number of the access list. It can be a value between 1 and 199.
deny	Denies access and clears call requests immediately.
permit	Permits access to the protocol translator.
X.121-address	X.121 address, with or without regular expression pattern-matching characters, with which to compare for access.

Default

None

Command Mode

Global

Usage Guidelines

An access list can contain any number of access list items. The list are processed in the order in which you entered them, with the first match causing the permit or deny condition. If an X.121 address does not match any of the regular expression in the access list, access will be denied.

Access lists take advantage of the message field defined by Recommendation X.29, which describes procedures for exchanging data between two PADs or a PAD and a DTE device.

The UNIX-style regular expression characters allow for pattern matching of characters and character strings in the address. Various pattern-matching constructions are available that will allow many addresses to be matched by a single regular expressions. Refer to Appendix F, "X.3 PAD Parameters," for more information.

Example

The following example permits connections to hosts with addresses beginning with the string 31370:

```
x29 access-list 2 permit ^31370
```

x29 profile

To create a PAD profile script for use by the **translate** command, use the **x29 profile** global configuration command.

> **x29 profile** *name parameter***:***value* [*parameter***:***value*]

Syntax Description

name	Name of the PAD profile script.
*X.parameter***:***value*	X.3 PAD parameter number and value separated by a colon. You can specify multiple parameter-value pairs.

Default

None

Command Mode

Global

Usage Guidelines

When an X.25 connection is established, the protocol translator acts as if an X.29 SET PARAMETER packet had been sent containing the parameters and values set by the **x29 profile** command and sets the protocol translator accordingly.

Example

The following profile script turns local edit mode on when the connection is made and establishes local echo and line termination upon receipt of a Return. The name "linemode" is used with the **translate** global configuration command to effect use of this script.

```
x29 profile linemode 2:1 3:2 15:1
```

Related Command

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

translate [†]